

RA0E3 Group

User's Manual: Hardware

32-Bit MCU

Renesas Advanced (RA) Family
Renesas RA0 Series

All information contained in these materials, including products and product specifications, represents information on the product at the time of publication and is subject to change by Renesas Electronics Corp. without notice. Please review the latest information published by Renesas Electronics Corp. through various means, including the Renesas Electronics Corp. website (<http://www.renesas.com>).

RA0E3组

用户手册：硬件

32位微控制器

瑞萨先进（RA）系列
瑞萨 RA0 系列

本资料中包含的所有信息，包括产品及产品规格，均代表发布时的产品信息，瑞萨电子株式会社保留随时更改的权利，恕不另行通知。请查阅瑞萨电子株式会社通过各种渠道（包括瑞萨电子株式会社的官方网站）发布的最新信息。
网站（<http://www.renesas.com>）。

Notice

- Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
- Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other claims involving patents, copyrights, or other intellectual property rights of third parties, by or arising from the use of Renesas Electronics products or technical information described in this document, including but not limited to, the product data, drawings, charts, programs, algorithms, and application examples.
- No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
- You shall be responsible for determining what licenses are required from any third parties, and obtaining such licenses for the lawful import, export, manufacture, sales, utilization, distribution or other disposal of any products incorporating Renesas Electronics products, if required.
- You shall not alter, modify, copy, or reverse engineer any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copying or reverse engineering.
- Renesas Electronics products are classified according to the following two quality grades: “Standard” and “High Quality”. The intended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below.

“Standard”: Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; industrial robots; etc.

“High Quality”: Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.

- Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems; surgical implantations; etc.), or may cause serious property damage (space system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user’s manual or other Renesas Electronics document.
- No semiconductor product is absolutely secure. Notwithstanding any security measures or features that may be implemented in Renesas Electronics hardware or software products, Renesas Electronics shall have absolutely no liability arising out of any vulnerability or security breach, including but not limited to any unauthorized access to or use of a Renesas Electronics product or a system that uses a Renesas Electronics product. RENESAS ELECTRONICS DOES NOT WARRANT OR GUARANTEE THAT RENESAS ELECTRONICS PRODUCTS, OR ANY SYSTEMS CREATED USING RENESAS ELECTRONICS PRODUCTS WILL BE INVULNERABLE OR FREE FROM CORRUPTION, ATTACK, VIRUSES, INTERFERENCE, HACKING, DATA LOSS OR THEFT, OR OTHER SECURITY INTRUSION (“Vulnerability Issues”). RENESAS ELECTRONICS DISCLAIMS ANY AND ALL RESPONSIBILITY OR LIABILITY ARISING FROM OR RELATED TO ANY VULNERABILITY ISSUES. FURTHERMORE, TO THE EXTENT PERMITTED BY APPLICABLE LAW, RENESAS ELECTRONICS DISCLAIMS ANY AND ALL WARRANTIES, EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT AND ANY RELATED OR ACCOMPANYING SOFTWARE OR HARDWARE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE.
 - When using Renesas Electronics products, refer to the latest product information (data sheets, user’s manuals, application notes, “General Notes for Handling and Using Semiconductor Devices” in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.
 - Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a failure or malfunction of Renesas Electronics products, such as safety design for hardware and software, including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult and impractical, you are responsible for evaluating the safety of the final products or systems manufactured by you.
 - Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
 - Renesas Electronics products and technologies shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You shall comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
 - It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
 - This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
 - Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.

(Note2) “Renesas Electronics product(s)” means any product developed or manufactured by or for Renesas Electronics.

(Rev.5.0-1 October 2020)

注意

- 本文档中对电路、软件及其他相关信息的描述仅用于说明半导体产品的运行原理和应用示例。您须对产品或系统设计中电路、软件及信息的整合或任何其他用途承担全部责任。瑞萨电子对您或第三方因使用这些电路、软件或信息而遭受的任何损失和损害不承担任何责任。

- 瑞萨电子特此明确声明，对于因使用瑞萨电子产品或本文件中描述的技术信息（包括但不限于产品数据、图纸、图表、程序、算法和应用示例）而引起的或与之相关的侵犯第三方专利、版权或其他知识产权的索赔，瑞萨电子不承担任何保证和责任。

- 本协议未授予瑞萨电子或其他方的任何专利、版权或其他知识产权的任何明示、默示或其他许可。

- 您应负责确定需要从任何第三方获得哪些许可，并负责获得合法进口、出口、制造、销售、使用、分销或以其他方式处置任何包含瑞萨电子产品的产品所需的此类许可（如有需要）。

- 您不得对瑞萨电子的任何产品（无论全部或部分）进行更改、修改、复制或反向工程。对于因上述更改、修改、复制或反向工程而导致您或第三方遭受的任何损失或损害，瑞萨电子概不承担任何责任。

- 瑞萨电子产品根据以下两个质量等级进行分类：“标准”和“高品质”。每款瑞萨电子产品的预期用途取决于其质量等级，如下所示。

“标准”： 计算机；办公设备；通讯设备；测试测量设备；音视频设备；家用电器；机床；个人电子设备；工业机器人；等等。

“高质量”： 运输设备（汽车、火车、轮船等）；交通管制（交通信号灯）；大型通信设备；关键金融终端系统；安全控制设备；等等。

除非瑞萨电子数据手册或其他瑞萨电子文档中明确指定为高可靠性产品或适用于恶劣环境的产品，否则瑞萨电子产品不适用于或授权用于可能对人身安全构成直接威胁（例如人工生命维持装置或系统、外科植入物等）或可能造成严重财产损失（例如航天系统、海底中继器、核电控制系统、飞机控制系统、关键工厂系统、军事装备等）的产品或系统。对于您或任何第三方因使用任何不符合瑞萨电子数据手册、用户手册或其他瑞萨电子文档规定的瑞萨电子产品而遭受的任何损害或损失，瑞萨电子概不承担任何责任。

- 任何半导体产品都无法做到绝对安全。尽管瑞萨电子的硬件或软件产品中可能实施了任何安全措施或功能，但对于任何漏洞或安全漏洞，包括但不限于任何未经授权访问或使用瑞萨电子产品或使用瑞萨电子产品的系统，瑞萨电子概不承担任何责任。瑞萨电子不保证其产品或使用瑞萨电子产品创建的任何系统不会受到攻击、损坏、病毒、干扰、黑客攻击、数据丢失或被盗或其他安全入侵（“漏洞问题”）的影响。瑞萨电子对因任何漏洞问题引起或与之相关的任何及所有责任概不负责。此外，在适用法律允许的范围内，瑞萨电子对本文件及任何相关或随附的软件或硬件不作任何明示或暗示的保证，包括但不限于适销性或特定用途适用性的暗示保证。

- 使用瑞萨电子产品时，请参考最新的产品信息（数据手册、用户手册、应用笔记、可靠性手册中的“半导体器件处理和使用通用注意事项”等），并确保使用条件在瑞萨电子规定的范围内，包括最大额定值、工作电源电压范围、散热特性、安装等。对于因在超出上述规定范围使用瑞萨电子产品而导致的任何故障、失效或事故，瑞萨电子概不承担任何责任。

- 尽管瑞萨电子致力于提升其产品的质量和可靠性，但半导体产品具有其固有特性，例如在特定条件下会发生故障或失效。除非在瑞萨电子的数据手册或其他文件中明确标明为高可靠性产品或适用于恶劣环境的产品，否则瑞萨电子产品不进行抗辐射设计。您有责任采取安全措施，防止瑞萨电子产品发生故障或失效时造成人身伤害、火灾或其他损害，以及/或对公众造成危险。这些措施包括硬件和软件的安全设计，例如冗余设计、防火和故障预防、针对老化退化的适当处理或其他任何适当的措施。由于仅对微计算机软件进行评估非常困难且不切实际，因此您有责任评估您制造的最终产品或系统的安全性。

- 有关瑞萨电子各产品的环境兼容性等环境相关事宜的详细信息，请联系瑞萨电子销售办事处。您有责任仔细、充分地了解有关管制物质添加或使用的适用法律法规，包括但不限于欧盟RoHS指令，并确保瑞萨电子产品的使用符合所有适用法律法规。对于因您违反适用法律法规而导致的任何损害或损失，瑞萨电子概不承担任何责任。

- 瑞萨电子的产品和技术不得用于或集成到任何根据适用的国内或国外法律法规禁止制造、使用或销售的产品或系统中。您应遵守任何对交易双方或交易行使管辖权的国家政府颁布和管理的适用出口管制法律法规。

- 瑞萨电子产品的买方或分销商，或任何其他分销、处置或以其他方式出售或转让该产品给第三方的各方，有责任事先通知该第三方本文件中规定的内容和条件。

- 未经瑞萨电子事先书面同意，不得以任何形式全部或部分地转载、复制或复印本文件。
- 如果您对本文档中包含的信息或瑞萨电子产品有任何疑问，请联系瑞萨电子销售办事处。

（注1）本文档中使用的“瑞萨电子”是指瑞萨电子株式会社，也包括其直接或间接控制的子公司。

（注2）“瑞萨电子产品”是指由瑞萨电子开发或制造或为瑞萨电子开发或制造的任何产品。

（2020年10月修订版5.0-1）

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,

Koto-ku, Tokyo 135-0061, Japan

www.renesas.com

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.

Contact Information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit:

www.renesas.com/contact/

公司总部

丰洲 FORESIA, 3-2-24 丰洲,

日本东京江东区 135-0061

www.renesas.com

商标

Renesas 和 Renesas 标志是瑞萨电子株式会社的商标。所有商标和注册商标均为其各自所有者的财产。

联系信息

如需了解产品、技术、文档最新版本或离您最近的销售办事处的更多信息，请访问：

www.renesas.com/contact/

General Precautions in the Handling of Microprocessing Unit and Microcontroller Unit Products

The following usage notes are applicable to all Microprocessing unit and Microcontroller unit products from Renesas. For detailed usage notes on the products covered by this document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

1. Precaution against Electrostatic Discharge (ESD)

A strong electrical field, when exposed to a CMOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop the generation of static electricity as much as possible, and quickly dissipate it when it occurs. Environmental control must be adequate. When it is dry, a humidifier should be used. This is recommended to avoid using insulators that can easily build up static electricity.

Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work benches and floors must be grounded. The operator must also be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions must be taken for printed circuit boards with mounted semiconductor devices.

2. Processing at power-on

The state of the product is undefined at the time when power is supplied. The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the time when power is supplied. In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the time when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the time when power is supplied until the power reaches the level at which resetting is specified.

3. Input of signal during power-off state

Do not input signals or an I/O pull-up power supply while the device is powered off. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements. Follow the guideline for input signal during power-off state as described in your product documentation.

4. Handling of unused pins

Handle unused pins in accordance with the directions given under handling of unused pins in the manual. The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of the LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible.

5. Clock signals

After applying a reset, only release the reset line after the operating clock signal becomes stable. When switching the clock signal during program execution, wait until the target clock signal is stabilized. When the clock signal is generated with an external resonator or from an external oscillator during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Additionally, when switching to a clock signal produced with an external resonator or by an external oscillator while program execution is in progress, wait until the target clock signal is stable.

6. Voltage application waveform at input pin

Waveform distortion due to input noise or a reflected wave may cause malfunction. If the input of the CMOS device stays in the area between V_{IL} (Max.) and V_{IH} (Min.) due to noise, for example, the device may malfunction. Take care to prevent chattering noise from entering the device when the input level is fixed, and also in the transition period when the input level passes through the area between V_{IL} (Max.) and V_{IH} (Min.).

7. Prohibition of access to reserved addresses

Access to reserved addresses is prohibited. The reserved addresses are provided for possible future expansion of functions. Do not access these addresses as the correct operation of the LSI is not guaranteed.

8. Differences between products

Before changing from one product to another, for example to a product with a different part number, confirm that the change will not lead to problems. The characteristics of a microprocessing unit or microcontroller unit products in the same group but having a different part number might differ in terms of internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

微处理器和微控制器产品处理的一般注意事项

以下使用说明适用于瑞萨电子的所有微处理器和微控制器产品。有关本文档所涵盖产品的详细使用说明，请参阅本文档的相关章节以及针对这些产品发布的任何技术更新。

1. 防止静电放电 (ESD)

强电场作用于CMOS器件时，会破坏栅极氧化层，最终导致器件性能下降。必须采取措施尽可能防止静电产生，并在静电产生时迅速将其消散。环境控制必须到位。干燥时，应使用加湿器。建议避免使用容易积聚静电的绝缘材料。

半导体器件必须存放在防静电容器、防静电屏蔽袋或导电材料中，并进行运输。所有测试和测量工具，包括工作台和地面，都必须接地。操作人员也必须佩戴接地腕带。半导体

devices must not be touched with bare hands. Similar precautions must be taken for printed circuit boards with mounted semiconductor devices.

2. 开机处理

通电时，产品的状态未定义。LSI内部电路的状态不确定，寄存器设置和引脚的状态在通电时也未定义。在成品中，如果复位信号施加到外部复位引脚，则从通电到复位过程完成期间，引脚的状态无法保证。同样，对于通过片上上电复位功能复位的产品，从通电到电源达到复位所需的电平期间，引脚的状态也无法保证。

3. 断电状态下的信号输入

设备断电时，请勿输入信号或 I/O 上拉电源。输入此类信号或 I/O 上拉电源产生的电流注入可能导致设备故障，此时流经设备的异常电流可能会损坏内部元件。请遵循产品文档中关于断电状态下输入信号的指导原则。

4. 未使用引脚的处理

请按照手册中“未使用引脚的处理”部分所述步骤处理未使用的引脚。CMOS 产品的输入引脚通常处于高阻抗状态。如果将未使用的引脚置于开路状态，则会在 LSI 附近感应出额外的电磁噪声，导致内部产生直通电流，并且由于引脚状态被错误识别为输入信号，可能会引发故障。

5. 时钟信号

复位后，只有在工作时钟信号稳定后才能释放复位线。在程序执行期间切换时钟信号时，请等待目标时钟信号稳定。复位期间，如果时钟信号由外部谐振器或外部振荡器产生，请确保在时钟信号完全稳定后才能释放复位线。此外，在程序执行过程中切换到由外部谐振器或外部振荡器产生的时钟信号时，也请等待目标时钟信号稳定。

6. 输入引脚上的电压施加波形

输入噪声或反射波引起的波形失真可能导致故障。例如，如果由于噪声导致CMOS器件的输入电压停留在 V_{IL} （最大值）和 V_{IH} (Min.) 之间的区域，则器件可能会发生故障。当输入电平固定时，以及输入电平在 V_{IL} （最大值）和 V_{IH} （最小值）之间的过渡期间，应注意防止抖动噪声进入器件。

7. 禁止访问保留地址

禁止访问保留地址。保留地址仅供未来功能扩展之用。请勿访问这些地址，否则无法保证LSI的正常运行。

8. 产品之间的差异

在更换产品之前，例如更换为不同部件号的产品，请务必确认更换不会导致任何问题。即使是同一组但部件号不同的微处理器或微控制器单元产品，其内部存储容量、布局模式和其他因素也可能存在差异，这些差异会影响电气特性范围，例如特性值、工作裕度、抗噪声能力和辐射噪声量。更换为不同部件号的产品时，请对该产品进行系统评估测试。

Preface

1. About this document

This manual is generally organized into an overview of the product, descriptions of the CPU, system control functions, peripheral functions, electrical characteristics, and usage notes. This manual describes the product specification of the microcontroller (MCU) superset. Depending on your product, some pins, registers, or functions might not exist. Address space that store unavailable registers are reserved.

2. Audience

This manual is written for system designers who are designing and programming applications using the Renesas Microcontroller. The user is expected to have basic knowledge of electrical circuits, logic circuits, and the MCU.

3. Renesas Publications

Renesas provides the following documents. Before using any of these documents, visit www.renesas.com for the most up-to-date version of the document.

Component	Document Type	Description
Microcontrollers	Data sheet	Features, overview, and electrical characteristics of the MCU
	User's Manual: Hardware	MCU specifications such as pin assignments, memory maps, peripheral functions, electrical characteristics, timing diagrams, and operation descriptions
	Application Notes	Technical notes, board design guidelines, and software migration information
	Technical Update (TU)	Preliminary reports on product specifications such as restriction and errata
Software	User's Manual: Software	API reference and programming information
	Application Notes	Project files, guidelines for software programming, and application examples to develop embedded software applications
Tools & Kits, Solutions	User's Manual: Development Tools	User's manual and quick start guide for developing embedded software applications with Development Kits (DK), Starter Kits (SK), Promotion Kits (PK), Product Examples (PE), and Application Examples (AE)
	User's Manual: Software	
	Quick Start Guide	
	Application Notes	Project files, guidelines for software programming, and application examples to develop embedded software applications
Other	Application Notes	Supplement of EMI (Electro Magnetic Interference) design.

前言

1.关于本文档

本手册大致分为产品概述、CPU 说明、系统控制功能、外设功能、电气特性和使用注意事项五个部分。本手册描述了微控制器 (MCU) 超集的产品规格。根据您的产品，某些引脚、寄存器或功能可能不存在。存储不可用寄存器的地址空间已被保留。

2. 受众

本手册面向使用瑞萨单片机设计和编程应用程序的系统设计人员。用户应具备电路、逻辑电路和单片机的基本知识。

3.瑞萨出版公司

瑞萨电子提供以下文档。在使用任何文档之前，请访问 www.renesas.com 获取最新版本。

成分	文档类型	描述
微控制器	数据表	MCU 的特性、概述和电气特性
	用户手册：硬件	MCU规格，例如引脚分配、存储器映射、外设功能、电气特性、时序图和操作说明
	应用说明	技术说明、电路板设计指南和软件迁移信息
	技术更新（TU）	产品规格的初步报告，例如限制和勘误
软件	用户手册：软件	API 参考和编程信息
	应用说明	项目文件、软件编程指南和嵌入式软件开发应用示例
工具和套件，解决方案	用户手册：开发工具	嵌入式开发用户手册和快速入门指南 软件应用程序，包括开发套件 (DK)、入门套件 (SK)、推广套件 (PK)、产品示例 (PE) 和应用程序。 示例（AE）
	用户手册：软件	
	快速入门指南	
	应用说明	项目文件、软件编程指南和嵌入式软件开发应用示例
其他	应用说明	电磁干扰（EMI）设计补充。

4. Numbering Notation

The following numbering notation is used throughout this manual:

Example	Description
011b	Binary number. For example, the binary equivalent of the number 3 is 011b.
0x1F	Hexadecimal number. For example, the hexadecimal equivalent of the number 31 is described 0x1F. In some cases, a hexadecimal number is shown with the suffix "h".
1234	Decimal number. A decimal number is followed by this symbol only when the possibility of confusion exists. Decimal numbers are generally shown without a suffix.

5. Typographic Notation

The following typographic notation is used throughout this manual:

Example	Description
AAA.BBB.CCC	Periods separated a function module symbol (AAA), register symbol (BBB), and bit field symbol (CCC).
AAA.BBB	A period separated a function module symbol (AAA) and register symbol (BBB).
BBB.DDD	A period separated a register symbol (BBB) and bit field symbol (DDD).
EEE[3:0]	Numbers in brackets expresses a bit number. For example, EEE[3:0] occupies bits 3 to 0.

6. Unit and Unit Prefix

The following units and unit prefixes are sometimes misleading. Those unit prefixes are described throughout this manual with the following meaning:

Symbol	Name	Description
b	Binary Digit	Single 0 or 1
B	Byte	This unit is generally used for memory specification of the MCU and address space.
k	kilo-	1000 = 10 ³ . k is also used to denote 1024 (2 ¹⁰) but this unit prefix is used to denote 1000 (10 ³) throughout this manual.
K	Kilo-	1024 = 2 ¹⁰ . This unit prefix is used to denote 1024 (2 ¹⁰) not 1000 (10 ³) throughout this manual.

7. Special Terms

The following terms have special meanings.

Term	Description
NC	Not connected pin. This pin should be left floating unless specified otherwise.
Hi-Z	High impedance.
x	Don't care or undefined.

4.编号符号

本手册通篇采用以下编号符号：

例子	描述
011b	二进制数。例如，数字 3 的二进制表示是 011b。
0x1F	十六进制数。例如，数字 31 的十六进制表示为 0x1F。在某些情况下，十六进制数会带有后缀“h”。
1234	小数。只有当可能造成混淆时，小数后才加此符号。小数通常不加后缀。

5.印刷符号

本手册通篇采用以下印刷符号：

例子	描述
AAA.BBB.CCC	句点分隔功能模块符号（AAA）、寄存器符号（BBB）和位域符号（CCC）。
AAA.BBB	功能模块符号 (AAA) 和寄存器符号 (BBB) 之间用句点隔开。
BBB.DDD	句点分隔寄存器符号 (BBB) 和位域符号 (DDD)。
EEE[3:0]	括号内的数字表示比特位编号。例如，EEE[3:0] 占用比特位 3 到 0。

6.单位和单位前缀

以下单位和单位前缀有时会产生误导。本手册中对这些单位前缀的含义进行了如下说明：

象征	Name	描述
b	二进制数字	单个 0 或 1
B	字节	该单元通常用于MCU的存储器规格和地址空间。
k	公斤	1000 = 10 ³ . k 也用于表示 1024(2 ¹⁰)，但本手册中，该单位前缀用于表示 10 00(10 ³)。
K	公斤	1024 = 2 ¹⁰ . 本手册中，此单位前缀用于表示 1024(2 ¹⁰) 而不是 1000(10 ³)。

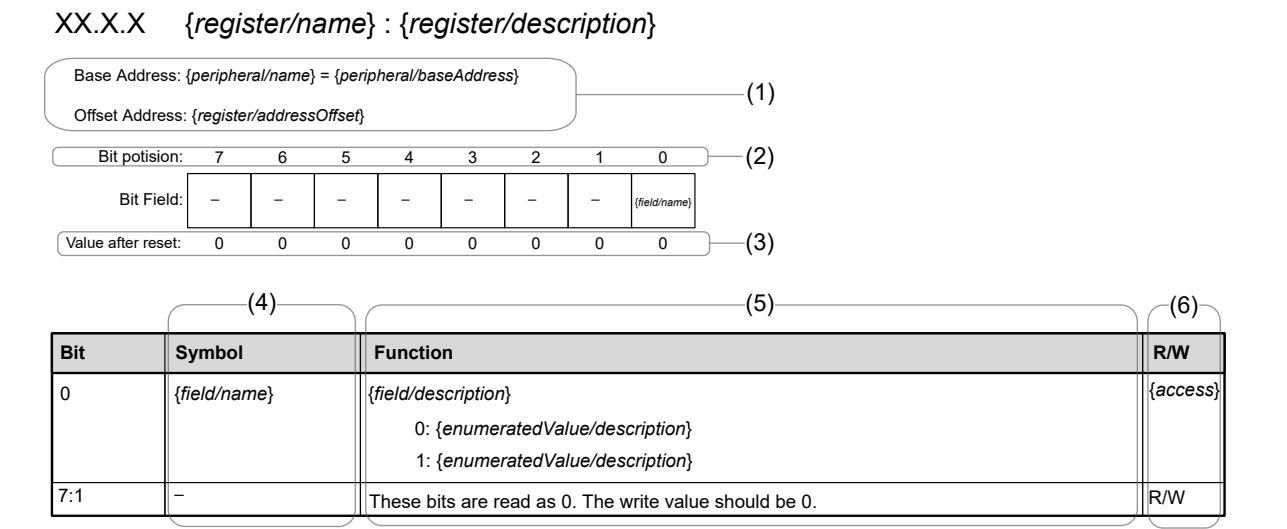
7.特殊条款

下列术语具有特殊含义。

学期	描述
NC	未连接引脚。除非另有规定，否则此引脚应保持悬空状态。
高阻抗	高阻抗。
x	不在乎或未定义。

8. Register Description

Each register description includes both a register diagram that shows the bit assignments and a register bit table that describes the content of each bit. The example of symbols used in these tables are described in the sections that follow. The following is an example of a register description and associated bit field definition.



(1) Function module symbol, register symbol, and address assignment

Function module symbol, {pheripheral/name}, register symbol, {register/name}, and address assignment of this register are generally expressed. Base Address and Offset Address mean {regiser/name} : {register/description} of {peripheral/name} is assigned to address {peripheral/baseAddress} + {register/addressOffset}.

(2) Bit number

This number indicates the bit number. This bits are shown in order from bits 31 to 0 for 32-bit register, from bits 15 to 0 for 16-bit register, and from bits 7 to 0 for 8-bit register.

(3) Value after reset

This symbol or number indicate the value of each bit after a hard reset. The value is shown in binary unless specified otherwise.

- 0: Indicates that the value is 0 after a reset.
- 1: Indicates that the value is 1 after a reset.
- x: Indicates that the value is undefined after a reset.

(4) Symbol

{filed/name} indicates the short name of bit field. Reserved bit is expressed with a —.

(5) Function

Function indicates the full name of the bit field, {field/description}, and enumerated values.

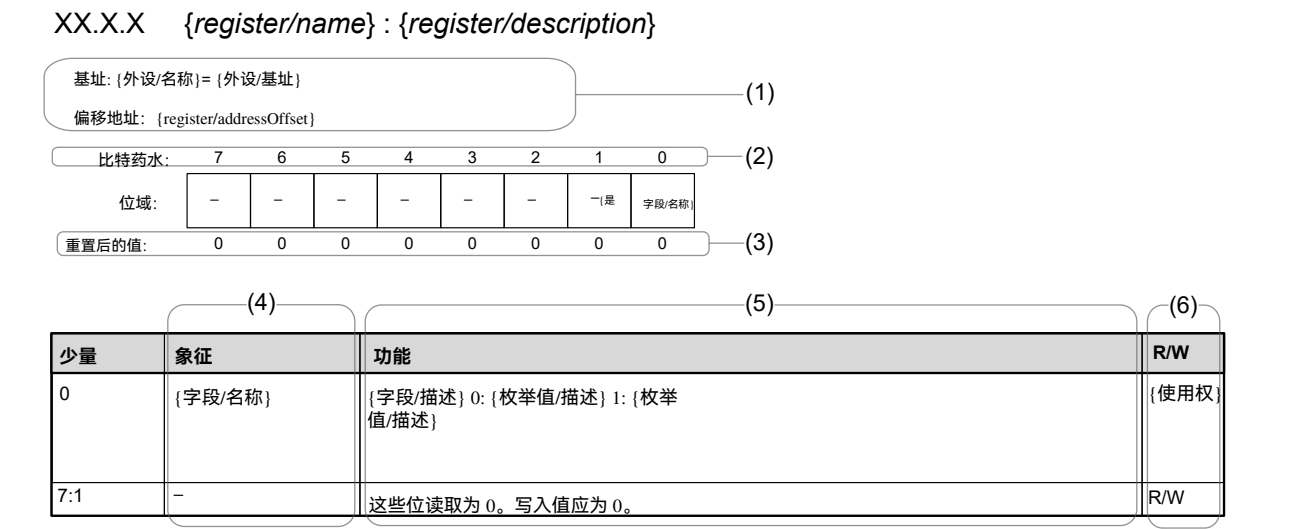
(6) R/W

The R/W column indicates access type whether the bit field is readable or writable.

- R/W: The bit field is readable and writable.
- R: The bit field is readable only. Writing to this bit field has no effect.
- W: The bit field is writable only. The read value is the same as after a reset unless specified otherwise.

8.注册说明

每个寄存器描述都包含一个寄存器图（显示位分配）和一个寄存器位表（描述每个位的内容）。这些表中使用的符号示例将在后续章节中进行描述。以下是寄存器描述及其相关位域定义的示例。



(1) 功能模块符号、寄存器符号和地址分配

通常表示功能模块符号 {外设/名称}、寄存器符号 {寄存器/名称} 以及该寄存器的地址分配。基地址和偏移地址表示 {寄存器/名称} : {外设/名称} 的 {寄存器/描述} 被分配到地址 {外设/基地址}+ {寄存器/地址偏移}。

(2) 位编号

该数字表示位序号。对于 32 位寄存器，这些位按顺序从第 31 位到第 0 位显示；对于 16 位寄存器，这些位按顺序从第 15 位到第 0 位显示；对于 8 位寄存器，这些位按顺序从第 7 位到第 0 位显示。

(3) 重置后的值

此符号或数字表示硬复位后每一位的值。除非另有说明，否则该值以二进制形式显示。

- 0: 表示重置后值为 0。
- 1: 表示重置后值为 1。
- x: 表示重置后该值未定义。

(4) 符号

{filed/name} 表示位字段的简称。保留位用—表示。

(5) 功能

函数表示位字段的完整名称，{字段/描述}，以及枚举值。

(6) R/W

R/W 列指示访问类型，即位字段是可读还是可写。

- R/W: 该位域可读写。
- R: 该位域只读。写入该位域无效。
- W: 该位域仅可写。除非另有规定，读取值与复位后的值相同。

9. Abbreviations

Abbreviations used in this document are shown in the following table.

Abbreviation	Description
AHB	Advanced High-performance Bus
AHB-AP	AHB Access Port
APB	Advanced Peripheral Bus
APB-AP	APB Access Port
BCD	Binary Coded Decimal
HMI	Human Machine Interface
IrDA	Infrared Data Association
LSB	Least Significant Bit
MSB	Most Significant Bit
MTB	Micro Trace Buffer
NVIC	Nested Vector Interrupt Controller
PC	Program Counter
PFS	Port Function Select
POR	Power-on reset
PWM	Pulse Width Modulation
S/H	Sample and Hold
SP	Stack Pointer
SWD	Serial Wire Debug
SW-DP	Serial Wire-Debug Port
SWJ-DP	Serial Wire JTAG Debug Port
UART	Universal Asynchronous Receiver/Transmitter
WFI	Wait for interrupt

9. 缩写

本文档中使用的缩写如下表所示。

缩写	描述
AHB	先进高性能总线
AHB-AP	AHB接入端口
APB	高级外围总线
APB-AP	APB接入端口
BCD	二进制编码十进制
HMI	人机界面
红外光谱	红外数据协会
LSB	最低有效位
MSB	最重要位
MTB	微量缓冲液
NVIC	嵌套向量中断控制器
PC	程序计数器
PFS	端口功能选择
POR	上电重置
PWM	脉冲宽度调制
S/H	采样和保持
SP	栈指针
SWD	串行线调试
SW-DP	串行线调试端口
SWJ-DP	串行线 JTAG 调试端口
UART	通用异步收发器
WFI	等待中断

10. Proprietary Notice

All text, graphics, photographs, trademarks, logos, artwork and computer code, collectively known as content, contained in this document is owned, controlled or licensed by or to Renesas, and is protected by trade dress, copyright, patent and trademark laws, and other intellectual property rights and unfair competition laws. Except as expressly provided herein, no part of this document or content may be copied, reproduced, republished, posted, publicly displayed, encoded, translated, transmitted or distributed in any other medium for publication or distribution or for any commercial enterprise, without prior written consent from Renesas.

Arm® and Cortex® are registered trademarks of Arm Limited. CoreSight™ is a trademark of Arm Limited.

CoreMark® is a registered trademark of the Embedded Microprocessor Benchmark Consortium.

SuperFlash® is a registered trademark of Silicon Storage Technology, Inc. in several countries including the United States and Japan.

Other brands and names mentioned in this document may be the trademarks or registered trademarks of their respective holders.

11. Feedback on the product

If you have any comments or suggestions about this product, go to [Contact Us](#).

10. 所有权声明

本文件中包含的所有文字、图形、照片、商标、标识、艺术作品和计算机代码（统称为“内容”）均归瑞萨电子所有、控制或授权使用，并受商业外观、版权、专利和商标法以及其他知识产权和反不正当竞争法的保护。除本文件明确规定外，未经瑞萨电子事先书面同意，不得以任何方式复制、再版、发布、公开展示、编码、翻译、传输或分发本文件或其任何内容，用于出版、发行或任何商业用途。

Arm®和 Cortex® 是 Arm Limited 的注册商标。CoreSight™是 Arm Limited 的商标。

CoreMark® 是嵌入式微处理器基准测试联盟的注册商标。

SuperFlash® 是 Silicon Storage Technology, Inc. 在美国和日本等多个国家的注册商标。

本文档中提及的其他品牌和名称可能是其各自所有者的商标或注册商标。

11. 产品反馈

如果您对本产品有任何意见或建议，请前往“联系我们”页面。

Contents

Features	21
1. Overview.....	22
1.1 Function Outline	22
1.2 Block Diagram	25
1.3 Part Numbering	25
1.4 Function Comparison	27
1.5 Pin Functions	28
1.6 Pin Assignments	29
1.7 Pin Lists	30
2. CPU	31
2.1 Overview.....	31
2.1.1 CPU.....	31
2.1.2 Debug.....	31
2.1.3 Operating Frequency	31
2.1.4 Block Diagram	31
2.2 Implementation Options.....	32
2.3 SWD Interface	33
2.4 Debug Function	33
2.4.1 Debug Mode Definition.....	33
2.4.2 Debug Mode Effects.....	33
2.4.3 Trace Control (for the MTB)	34
2.4.4 CoreSight (for MTB)	34
2.5 Programmers Model.....	35
2.5.1 Address Spaces	35
2.5.2 Cortex-M23 Peripheral Address Map	35
2.5.3 External Debug Address Map	35
2.5.4 CoreSight ROM Table	36
2.5.5 DBGREG Module.....	36
2.5.6 OCDREG Module.....	38
2.6 SysTick Timer	41
2.7 OCD Emulator Connection	41
2.7.1 ID Code	41
2.7.2 DBGEN	41
2.7.3 Restrictions on Connecting an OCD emulator	41
2.8 References	43
2.9 Usage Notes.....	43
3. Operating Modes	44
3.1 Overview.....	44
3.2 Operating Modes Transitions.....	44
3.2.1 Operating Mode Transitions	44
4. Address Space.....	45
4.1 Address Space	45
5. Resets.....	46
5.1 Overview.....	46
5.2 Register Descriptions	47
5.2.1 RESF : Reset Status Flag Register.....	47
5.2.2 PORSR : Power-On Reset Status Register	48
5.3 Operation.....	48

内容

特征	21
1. 概述	22
1.1 功能概述	22
1.2框图	25
1.3零件编号	25
1.4 功能比较	27
1.5针脚功能	28
1.6引脚分配	29
1.7引脚列表	30
2. CPU	31
2.1 概述	31
2.1.1 CPU.....	31
2.1.2 调试	31
2.1.3 工作频率	31
2.1.4 框图	31
2.2 实施方案	32
2.3SWD接口	33
2.4调试功能	33
2.4.1 调试模式定义	33
2.4.2 调试模式效果	33
2.4.3 追踪控制（适用于山地自行车）.....	34
2.4.4 CoreSight（适用于山地自行车）.....	34
2.5 程序员模型	35
2.5.1 地址空间	35
2.5.2 Cortex-M23 外围地址映射	35
2.5.3 外部调试地址映射	35
2.5.4 CoreSight ROM 表	36
2.5.5 DBGREG 模块	36
2.5.6 OCDREG 模块	38
2.6 系统滴答定时器	41
2.7OCD模拟器连接	41
2.7.1 ID 代码	41
2.7.2 DBGEN	41
2.7.3 连接强迫症模拟器的限制	41
2.8 参考文献	43
2.9 使用说明	43
3.操作模式	44
3.1 概述	44
3.2 操作模式转换	44
3.2.1 运行模式转换	44
4.地址空间	45
4.1地址空间	45
5. 重置	46
5.1 概述	46
5.2寄存器描述	47
5.2.1 RESF：重置状态标志寄存器	47
5.2.2 PORSR：上电复位状态寄存器	48
5.3 操作 5.	48

5.3.1	RES Pin Reset	48
5.3.2	Power-On Reset.....	49
5.3.3	Voltage Monitor Reset.....	49
5.3.4	Independent Watchdog Timer Reset.....	50
5.3.5	Software Reset.....	50
6.	Option-Setting Memory	51
6.1	Overview.....	51
6.2	Register Descriptions	51
6.2.1	OFS0 : Option Function Select Register	51
6.2.2	OFS1 : Option Function Select Register	53
6.2.3	AWS : Access Window Setting Register	55
6.2.4	OSIS : OCD/Serial programmer ID Setting Register.....	56
6.3	Setting Option-Setting Memory	57
6.3.1	Allocation of Data in Option-Setting Memory	57
6.3.2	Setting Data for Programming Option-Setting Memory.....	57
6.4	Usage Notes.....	58
6.4.1	Data for Programming Reserved Areas and Reserved Bits in the Option-Setting Memory..	58
6.4.2	Note on FSPR Bit.....	58
7.	Low Voltage Detection (LVD).....	59
7.1	Overview.....	59
7.2	Register Descriptions	60
7.2.1	LVD1CR : Voltage Monitor 1 Circuit Control Register	60
7.2.2	LVD1MKR : Voltage Monitor 1 Circuit Mask Register	61
7.2.3	LVD1SR : Voltage Monitor 1 Circuit Status Register.....	61
7.3	VCC Input Voltage Monitor	62
7.3.1	Monitoring V _{det0}	62
7.3.2	Monitoring V _{det1}	62
7.4	Reset from Voltage Monitor 0	62
7.5	Interrupt and Reset from Voltage Monitor 1.....	63
8.	Clock Generation Circuit	65
8.1	Overview.....	65
8.2	Register Descriptions	66
8.2.1	FSUBSCR : FSUB Clock Source Control Register	66
8.2.2	ICLKSCR : ICLK Clock Source Control Register	67
8.2.3	LOCOCR : Low-speed On-chip Oscillator Control Register.....	67
8.2.4	HOCOCR : High-speed On-chip Oscillator Control Register	68
8.2.5	OSCSF : Oscillation Stabilization Flag Register	69
8.2.6	HOCODIV : High-speed On-chip Oscillator Frequency Select Register	70
8.2.7	OSMC : Subsystem Clock Supply Mode Control Register.....	70
8.2.8	CKS0 : Clock Out Control Register 0	70
8.2.9	LIOTRM : Low-speed On-chip Oscillator Trimming Register	71
8.2.10	HIOTRM : High-speed On-chip Oscillator Trimming Register.....	72
8.3	Internal Clock.....	72
8.3.1	System Clock (ICLK).....	73
8.3.2	IWDT Clock (IWDTCLK)	74
8.3.3	SysTick Timer-dedicated Clock (SYSTICCLK).....	74
8.3.4	External Pin Output Clock (CLKOUT)	74
8.4	Usage Notes.....	74
8.4.1	Register Access	74
8.4.2	Notes on Clock Generation Circuit.....	75

5.3.1	RES引脚复位	48
5.3.2	上电复位	49
5.3.3	电压监控器复位	49
5.3.4	独立看门狗定时器复位	50
5.3.5	软件重置	50
6.	选项设置内存	51
6.1	概述	51
6.2	寄存器描述	51
6.2.1	OFS0: 选项功能选择寄存器 0	51
6.2.2	OFS1: 选项功能选择寄存器 1	53
6.2.3	AWS: 访问窗口设置注册表	55
6.2.4	OSIS: OCD/串行编程器 ID 设置寄存器	56
6.3	设置选项-设置内存	57
6.3.1	选项设置内存中的数据分配	57
6.3.2	编程选项设置存储器的设置数据	57
6.4	使用说明	58
6.4.1	用于对选项设置存储器中的保留区域和保留位进行编程的数据..	58
6.4.2	关于 FSPR 位的说明.....	58
7.	低电压检测 (LVD)	59
7.1	概述	59
7.2	寄存器描述	60
7.2.1	LVD1CR: 电压监视器 1 电路控制寄存器	60
7.2.2	LVD1MKR: 电压监视器 1 电路掩码寄存器	61
7.2.3	LVD1SR: 电压监视器 1 电路状态寄存器	61
7.3	VCC 输入电压监控器	62
7.3.1	监测 V _{det0}	62
7.3.2	Monitoring V _{det1}	62
7.4	从电压监视器 0 重置	62
7.5	电压监视器 1 的中断和复位	63
8.	时钟生成电路	65
8.1	概述	65
8.2	寄存器描述	66
8.2.1	FSUBSCR: FSUB 时钟源控制寄存器	66
8.2.2	ICLKSCR: ICLK 时钟源控制寄存器	67
8.2.3	LOCOCR: 低速片上振荡器控制寄存器	67
8.2.4	HOCOCR: 高速片上振荡器控制寄存器	68
8.2.5	OSCSF: 振荡稳定标志寄存器	69
8.2.6	HOCODIV: 高速片上振荡器频率选择寄存器	70
8.2.7	OSMC: 子系统时钟供电模式控制寄存器	70
8.2.8	CKS0: 时钟输出控制寄存器 0	70
8.2.9	LIOTRM: 低速片上振荡器微调寄存器	71
8.2.10	HIOTRM: 高速片上振荡器微调寄存器	72
8.3	内部时钟	72
8.3.1	系统时钟 (ICLK)	73
8.3.2	IWDT时钟 (IWDTCLK)	74
8.3.3	SysTick 定时器专用时钟 (SYSTICCLK)	74
8.3.4	外部引脚输出时钟 (CLKOUT)	74
8.4	使用说明	74
8.4.1	注册访问权限	74
8.4.2	关于时钟发生电路的说明	75

9. Low Power Modes	76
9.1 Overview	76
9.2 Register Descriptions	78
9.2.1 SBYCR : Standby Control Register	78
9.2.2 MSTPCRA : Module Stop Control Register A	79
9.2.3 MSTPCRB : Module Stop Control Register B	79
9.2.4 MSTPCRC : Module Stop Control Register C	80
9.2.5 MSTPCRD : Module Stop Control Register D	80
9.2.6 FLMODE : Flash Operating Mode Control Register	81
9.2.7 FLMWRP : Flash Operating Mode Protect Register	82
9.2.8 SYOCDRC : System Control OCD Control Register	82
9.3 Reducing Power Consumption by Switching Clock Signals	82
9.4 Module-stop Function	82
9.5 Function for Lower Operating Power Consumption	83
9.5.1 Setting Operating Power Control Mode	83
9.5.2 Operating Range	84
9.6 Sleep Mode	86
9.6.1 Transitioning to Sleep Mode	86
9.6.2 Canceling Sleep Mode	86
9.7 Software Standby Mode	87
9.7.1 Transition to Software Standby Mode	87
9.7.2 Canceling Software Standby Mode	87
9.7.3 Example of Software Standby Mode Application	88
9.8 Snooze Mode	88
9.8.1 Transition to Snooze Mode	88
9.8.2 Canceling Snooze Mode	89
9.8.3 Returning from Snooze Mode to Software Standby Mode	90
9.8.4 Snooze Operation Example	90
9.9 Usage Notes	91
9.9.1 Register Access	91
9.9.2 I/O Port pin states	92
9.9.3 Module-stop State of DTC	92
9.9.4 Internal Interrupt Sources	92
9.9.5 Transitioning to Low Power Modes	92
9.9.6 Timing of WFI Instruction	93
9.9.7 Writing to the IWDTC Registers by DTC in Sleep Mode or Snooze Mode	93
9.9.8 Oscillators in Snooze Mode	93
9.9.9 Using SAU0 in Snooze Mode	93
9.9.10 Using UART0 in Snooze Mode	93
9.9.11 Sleep-on-exit Function	93
10. Register Write Protection	94
10.1 Overview	94
10.2 Register Descriptions	94
10.2.1 PRCR : Protect Register	94
11. Interrupt Controller Unit (ICU)	95
11.1 Overview	95
11.2 Register Descriptions	96
11.2.1 IRQCRI : IRQ Control Register i (i = 0 to 5)	96
11.2.2 NMISR : Non-maskable Interrupt Status Register	97
11.2.3 NMIER : Non-maskable Interrupt Enable Register	98

9.低功耗模式	76
9.1 概述	76
9.2 寄存器描述	78
9.2.1 SBYCR: 待机控制寄存器	78
9.2.2 MSTPCRA: 模块停止控制寄存器 A	79
9.2.3 MSTPCRB: 模块停止控制寄存器 B	79
9.2.4 MSTPCRC: 模块停止控制寄存器 C	80
9.2.5 MSTPCRD: 模块停止控制寄存器 D	80
9.2.6 FLMODE: 闪存操作模式控制寄存器	81
9.2.7 FLMWRP: 闪存操作模式保护寄存器	82
9.2.8 SYOCDRC: 系统控制OCD控制寄存器	82
9.3 通过切换时钟信号降低功耗	82
9.4 模块停止功能	82
9.5 降低运行功耗的功能	83
9.5.1 设置操作电源控制模式	83
9.5.2 操作范围	84
9.6睡眠模式	86
9.6.1 过渡到睡眠模式	86
9.6.2 取消睡眠模式	86
9.7 软件待机模式	87
9.7.1 过渡到软件待机模式	87
9.7.2 取消软件待机模式	87
9.7.3 软件待机模式应用示例	88
9.8 贪睡模式	88
9.8.1 切换到睡眠模式	88
9.8.2 取消贪睡模式	89
9.8.3 从休眠模式返回软件待机模式	90
9.8.4 贪睡操作示例	90
9.9 使用说明	91
9.9.1 注册访问权限	91
9.9.2 I/O端口引脚状态	92
9.9.3 DTC模块停止状态	92
9.9.4 内部中断源	92
9.9.5 过渡到低功耗模式	92
9.9.6 WFI 指令的时机	93
9.9.7 在睡眠模式或睡眠模式下, DTC 向 IWDTC 寄存器写入数据	93
9.9.8 振荡器处于休眠模式	93
9.9.9 在睡眠模式下使用 SAU0	93
9.9.10 在休眠模式下使用 UART0	93
9.9.11退出时睡眠功能	93
10.寄存器写保护	94
10.1 概述	94
10.2 寄存器描述	94
10.2.1PRCR: 保护寄存器	94
11.中断控制器单元 (ICU)	95
11.1 概述	95
11.2 寄存器描述	96
11.2.1 IRQCRI: IRQ 控制寄存器 i(i = 0至 5)	96
11.2.2 NMISR: 不可屏蔽中断状态寄存器	97
11.2.3NMIER: 不可屏蔽中断使能寄存器	98

11.2.4	NMICLR : Non-maskable Interrupt Status Clear Register	99
11.2.5	NMICR : NMI Pin Interrupt Control Register	99
11.2.6	DTCENST0 : DTC Enable Status Register 0	100
11.2.7	DTCENST1 : DTC Enable Status Register 1	101
11.2.8	DTCENSET0 : DTC Enable Set Register 0	101
11.2.9	DTCENSET1 : DTC Enable Set Register 1	102
11.2.10	DTCENCLR0 : DTC Enable Clear Register 0	103
11.2.11	DTCENCLR1 : DTC Enable Clear Register 1	103
11.2.12	INTFLAG0 : Interrupt Request Flag Monitor Register 0	104
11.2.13	INTFLAG1 : Interrupt Request Flag Monitor Register 1	105
11.2.14	SBYEDCR0 : Software Standby/Snooze End Control Register 0	106
11.2.15	SBYEDCR1 : Software Standby/Snooze End Control Register 1	107
11.3	Vector Table	108
11.3.1	Interrupt Vector Table	108
11.3.2	Event Number	110
11.4	Interrupt Operation	111
11.4.1	Detecting Interrupts	112
11.5	Interrupt setting procedure	112
11.5.1	Enabling Interrupt Requests	112
11.5.2	Disabling Interrupt Requests	112
11.5.3	Polling for Interrupts	112
11.5.4	Selecting Interrupt Request Destinations	113
11.5.5	External Pin Interrupts	113
11.6	Non-maskable Interrupt Operation	113
11.7	Return from Low Power Modes	114
11.7.1	Return from Sleep Mode	114
11.7.2	Return from Software Standby Mode	114
11.7.3	Return from Snooze Mode	114
11.8	Using the WFI Instruction with Non-maskable Interrupts	114
11.9	Reference	114
12.	Buses	115
12.1	Overview	115
12.2	Description of Buses	116
12.2.1	Main Buses	116
12.2.2	Slave Interface	116
12.2.3	Parallel Operations	116
12.2.4	Restriction on Endianness	116
12.2.5	Restriction on Exclusive Access	116
12.3	Register Descriptions	117
12.3.1	BUSMCNTx : Master Bus Control Register x (x = SYS, DMA)	117
12.3.2	BUSnERRADD : Bus Error Address Register n (n = 3, 4)	117
12.3.3	BUSnERRSTAT : BUS Error Status Register n (n = 3, 4)	118
12.4	Bus Error Monitoring Section	118
12.4.1	Error Type that Occurs by Bus	118
12.4.2	Operation when a Bus Error Occurs	118
12.4.3	Conditions for issuing illegal Address Access Errors	118
12.5	References	119
13.	Flash Read Protection (FRP)	120
13.1	Overview	120
13.1.1	Memory Protection	120

11.2.4	NMICLR: 不可屏蔽中断状态清除寄存器	99
11.2.5	NMICR: NMI引脚中断控制寄存器	99
11.2.6	DTCENST0: DTC 使能状态寄存器 0	100
11.2.7	DTCENST1: DTC 使能状态寄存器 1	101
11.2.8	DTCENSET0: DTC 使能设置寄存器 0	101
11.2.9	DTCENSET1: DTC 使能设置寄存器 1	102
11.2.10	DTCENCLR0: DTC 使能清除寄存器 0	103
11.2.11	DTCENCLR1: DTC 使能清除寄存器 1	103
11.2.12	INTFLAG0: 中断请求标志监视器寄存器 0	104
11.2.13	INTFLAG1: 中断请求标志监视器寄存器 1	105
11.2.14	SBYEDCR0: 软件待机/休眠结束控制寄存器 0	106
11.2.15	SBYEDCR1: 软件待机/休眠结束控制寄存器 1	107
11.3	向量表	108
11.3.1	中断向量表	108
11.3.2	事件编号	110
11.4	中断操作	111
11.4.1	检测中断	112
11.5	中断设置程序	112
11.5.1	启用中断请求	112
11.5.2	禁用中断请求	112
11.5.3	轮询中断	112
11.5.4	选择中断请求目标	113
11.5.5	外部引脚中断	113
11.6	不可屏蔽中断操作	113
11.7	从低功耗模式返回	114
11.7.1	从睡眠模式恢复	114
11.7.2	从软件待机模式返回	114
11.7.3	从休眠模式返回	114
11.8	将 WFI 指令与不可屏蔽中断一起使用	114
11.9	参考	114
12.	公交车	115
12.1	概述	115
12.2	公交车描述	116
12.2.1	主要公交车	116
12.2.2	从设备接口	116
12.2.3	并行操作	116
12.2.4	对字节序的限制	116
12.2.5	专属访问限制	116
12.3	寄存器描述	117
12.3.1	BUSMCNTx: 主总线控制寄存器x (x = SYS, DMA)	117
12.3.2	BUSnERRADD: 总线错误地址寄存器 n(n = 3, 4)	117
12.3.3	BUSnERRSTAT: 总线错误状态寄存器 n(n = 3, 4)	118
12.4	总线错误监控部分	118
12.4.1	总线发生的错误类型	118
12.4.2	总线故障时的操作	118
12.4.3	发出非法地址访问错误的条件	118
12.5	参考文献	119
13.	闪存读取保护 (FRP)	120
13.1	概述	120
13.1.1	内存保护	120

13.2	Usage Notes.....	121
13.2.1	Notes on the Use of a Debugger.....	121
13.2.2	Compiler Settings.....	121
13.2.3	Protection of OFS1 Register	121
14.	Data Transfer Controller (DTC).....	122
14.1	Overview.....	122
14.2	Register Descriptions	123
14.2.1	MRA : DTC Mode Register A	123
14.2.2	MRB : DTC Mode Register B	124
14.2.3	SAR : DTC Transfer Source Register	125
14.2.4	DAR : DTC Transfer Destination Register.....	126
14.2.5	CRA : DTC Transfer Count Register A.....	126
14.2.6	CRB : DTC Transfer Count Register B.....	127
14.2.7	DTCCR : DTC Control Register	127
14.2.8	DTCVBR : DTC Vector Base Register	127
14.2.9	DTCST : DTC Module Start Register	128
14.2.10	DTCSTS : DTC Status Register	128
14.3	Activation Sources.....	129
14.3.1	Allocating Transfer Information and DTC Vector Table	129
14.4	Operation.....	131
14.4.1	Transfer Information Read Skip Function.....	133
14.4.2	Transfer Information Write-Back Skip Function.....	133
14.4.3	Normal Transfer Mode	134
14.4.4	Repeat Transfer Mode	135
14.4.5	Block Transfer Mode	136
14.4.6	Chain Transfer.....	137
14.4.7	Operation Timing.....	138
14.4.8	Execution Cycles of DTC	140
14.4.9	DTC Bus Mastership Release Timing	141
14.5	DTC Setting Procedure	141
14.6	Examples of DTC Usage	142
14.6.1	Normal Transfer	142
14.6.2	Chain Transfer.....	142
14.6.3	Chain Transfer when Counter = 0	143
14.7	Interrupt	145
14.7.1	Interrupt Sources.....	145
14.8	Low Power Consumption Function.....	145
14.9	Usage Notes.....	146
14.9.1	Transfer Information Start Address	146
15.	I/O Ports.....	147
15.1	Overview.....	147
15.2	Register Descriptions	148
15.2.1	PODRm : Pmn Output Data Register (m = 0 to 3, n = 00 to 15).....	148
15.2.2	PDRm : Pmn Direction Register (m = 0 to 3, n = 00 to 15).....	149
15.2.3	PIDRm : Pmn State Register (m = 0 to 3, n = 00 to 15).....	149
15.2.4	PORRm : Pmn Output Reset Register (m = 0 to 3, n = 00 to 15)	150
15.2.5	POSRm : Pmn Output Set Register (m = 0 to 3, n = 00 to 15)	150
15.2.6	PmnPFS_A : Port mn Pin Function Select Register (m = 1 to 3, n = 00 to 15).....	151
15.2.7	P0nPFS_A : Port 0n Pin Function Select Register (n = 10 to 13).....	153
15.2.8	PWPR : Write-Protect Register	153

13.2	使用说明	121
13.2.1	关于调试器使用的注意事项	121
13.2.2	编译器设置	121
13.2.3	OFS1 注册表的保护	121
14.	数据传输控制器 (DTC)	122
14.1	概述	122
14.2	寄存器描述	123
14.2.1	MRA: DTC模式寄存器A	123
14.2.2	MRB: DTC模式寄存器B	124
14.2.3	SAR: DTC 传输源寄存器	125
14.2.4	DAR: DTC 转账目的地寄存器	126
14.2.5	CRA: DTC 转账计数寄存器 A	126
14.2.6	CRB: DTC 传输计数寄存器 B	127
14.2.7	DTCCR: DTC 控制寄存器	127
14.2.8	DTCVBR: DTC矢量基址寄存器	127
14.2.9	DTCST: DTC模块启动寄存器	128
14.2.10	DTCSTS: DTC 状态寄存器	128
14.3	激活源	129
14.3.1	分配转移信息和 DTC 向量表	129
14.4	操作	131
14.4.1	传输信息读取跳过功能	133
14.4.2	传输信息写回跳过功能	133
14.4.3	正常传输模式	134
14.4.4	重复传输模式	135
14.4.5	块传输模式	136
14.4.6	链式转移	137
14.4.7	操作时序	138
14.4.8	DTC的执行周期	140
14.4.9	DTC 总线主控权发布时间	141
14.5	DTC 设置程序	141
14.6	DTC 使用示例	142
14.6.1	正常转移	142
14.6.2	链式转移	142
14.6.3	当计数器= 0时链式转移	143
14.7	中断	145
14.7.1	中断源	145
14.8	低功耗功能	145
14.9	使用说明	146
14.9.1	转账信息起始地址	146
15.	I/O 端口	147
15.1	概述	147
15.2	寄存器描述	148
15.2.1	PODRm: Pmn 输出数据寄存器(m = 0 至 3, n = 00至 15)	148
15.2.2	PDRm: PMN 方向寄存器(m = 0 至 3, n = 00 至 15)	149
15.2.3	PIDRm: PMN 状态寄存器(m = 0至 3, n = 00至 15)	149
15.2.4	PORRm: Pmn 输出复位寄存器(m = 0至 3, n = 00至 15)	150
15.2.5	POSRm: PMN 输出设置寄存器(m = 0设置为 3, n = 00设置为 15)	150
15.2.6	PmnPFS_A: 端口 mn 引脚功能选择寄存器(m = 1至 3, n = 00 至 15)	151
15.2.7	P0nPFS_A: 端口 0n 引脚功能选择寄存器(n = 10至 13)	153
15.2.8	PWPR: 写保护寄存器	153

15.3	Operation	154
15.3.1	General I/O Ports	154
15.3.2	Port Function Select	154
15.4	Handling of Unused Pins	154
15.5	Usage Notes	155
15.5.1	Procedure for Specifying the Pin Functions	155
15.5.2	Port Output Data Register (PODR) Summary	155
15.5.3	Notes on Register Settings and Port Pin State	155
15.5.4	Notes on Using Analog Functions	156
15.5.5	Notes on Using Alternate Functions	156
15.5.6	Notes on Communications with Devices Operating at a Different Voltage (1.8 V, 2.5 V, or 3 V) by Switching I/O Buffers	160
15.5.7	Restriction on P206 Usage	161
15.6	Peripheral Select Settings for Each Product	161
16.	Timer Array Unit (TAU)	163
16.1	Overview	163
16.2	Register Descriptions	170
16.2.1	TCR0n : Timer Counter Register 0n (n = 0 to 7)	170
16.2.2	TDR0n/TDR01x/TDR03x : Timer Data Register 0n (n = 0 to 7) (x = L, H)	172
16.2.3	TPS0 : Timer Clock Select Register 0	172
16.2.4	TMR0n : Timer Mode Register 0n (n = 0, 2, 4, 5, 6, 7)	176
16.2.5	TMR0n : Timer Mode Register 0n (n = 1, 3)	178
16.2.6	TSR0n : Timer Status Register 0n (n = 0 to 7)	180
16.2.7	TE0 : Timer Channel Enable Status Register 0	181
16.2.8	TS0 : Timer Channel Start Register 0	181
16.2.9	TT0 : Timer Channel Stop Register 0	182
16.2.10	TIS0 : Timer Input Select Register 0	183
16.2.11	TOE0 : Timer Output Enable Register 0	183
16.2.12	TO0 : Timer Output Register 0	184
16.2.13	TOL0 : Timer Output Level Register 0	184
16.2.14	TOM0 :Timer Output Mode Register 0	185
16.2.15	ISC : Input Switch Control Register	186
16.2.16	TNFEN : TAU Noise Filter Enable Register	186
16.2.17	Registers Controlling Port Functions of Pins to be Used for Timer I/O	187
16.3	Basic Rules of Timer Array Unit	188
16.3.1	Basic Rules of Simultaneous Channel Operation Function	188
16.3.2	Basic Rules of 8-bit Timer Operation Function (Channels 1 and 3 only)	189
16.4	Operations of Counters	190
16.4.1	Count Clock (f _{TCLK})	190
16.4.2	Timing of the Start of Counting	191
16.4.3	Operations of Counters	192
16.5	Channel Output (TO0n Pin) Control	196
16.5.1	TO0n Pin Output Circuit Configuration	196
16.5.2	TO0n Pin Output Setting	197
16.5.3	Cautions on Channel Output Operation	198
16.5.4	Collective Manipulation of TO0.TO[n] Bit	201
16.5.5	Timer Interrupts and TO0n Outputs When Counting is Started	201
16.6	Timer Input (TI0n) Control	202
16.6.1	TI0n Input Circuit Configuration	202
16.6.2	Noise Filter	203

15.3	操作	154
15.3.1	通用 I/O 端口	154
15.3.2	端口功能选择	154
15.4	未使用引脚的处理	154
15.5	使用说明	155
15.5.1	引脚功能指定程序	155
15.5.2	端口输出数据寄存器 (PODR) 概述	155
15.5.3	关于寄存器设置和端口引脚状态的说明	155
15.5.4	关于使用模拟函数的注意事项	156
15.5.5	关于使用替代函数的说明	156
15.5.6	关于与工作电压不同的设备 (1.8 V、2.5 V 或 10 V) 进行通信的注意事项 3 V) 通过切换 I/O 缓冲区	160
15.5.7	对 P206 使用的限制	161
15.6	各产品的外围设备选择设置	161
16.	定时器阵列单元 (TAU)	163
16.1	概述	163
16.2	寄存器描述	170
16.2.1	TCR0n: 定时器计数器寄存器 0n(n = 0至 7)	170
16.2.2	TDR0n/TDR01x/TDR03x: 定时器数据寄存器 0n(n = 0至 7)(x = L, H)	172
16.2.3	TPS0: 定时器时钟选择寄存器 0	172
16.2.4	TMR0n: 定时器模式寄存器 0n(n = 0, 2, 4, 5, 6, 7)	176
16.2.5	TMR0n: 定时器模式寄存器 0n(n = 1, 3)	178
16.2.6	TSR0n: 定时器状态寄存器 0n(n = 0至 7)	180
16.2.7	TE0: 定时器通道使能状态寄存器 0	181
16.2.8	TS0: 定时器通道启动寄存器 0	181
16.2.9	TT0: 定时器通道停止寄存器 0	182
16.2.10	TIS0: 定时器输入选择寄存器 0	183
16.2.11	TOE0: 定时器输出使能寄存器 0	183
16.2.12	TO0: 定时器输出寄存器 0	184
16.2.13	TOL0: 定时器输出电平寄存器 0	184
16.2.14	TOM0: 定时器输出模式寄存器 0	185
16.2.15	ISC: 输入开关控制寄存器	186
16.2.16	TNFEN: TAU 噪声滤波器使能寄存器	186
16.2.17	寄存器控制用于定时器 I/O 的引脚的端口功能	187
16.3	定时器阵列单元的基本规则	188
16.3.1	同时信道操作功能的基本规则	188
16.3.2	8 位定时器操作功能的基本规则 (仅限通道 1 和 3)	189
16.4	计数器的操作	190
16.4.1	计数时钟 (f _{TCLK})	190
16.4.2	开始计数的时间	191
16.4.3	计数器的操作	192
16.5	通道输出 (TO0n引脚) 控制	196
16.5.1	TO0n 引脚输出电路配置	196
16.5.2	TO0n 引脚输出设置	197
16.5.3	通道输出操作注意事项	198
16.5.4	对 TO0.TO[n] 位进行集体操作	201
16.5.5	定时器中断和计数开始时的 TO0n 输出	201
16.6	定时器输入 (TI0n) 控制	202
16.6.1	TI0n 输入电路配置	202
16.6.2	噪声滤波器	203

16.6.3	Cautions on Channel Input Operation	203
16.7	Independent Channel Operation Function of Timer Array Unit	203
16.7.1	Operation as an Interval Timer or for Square Wave Output	203
16.7.2	Operation as an External Event Counter	207
16.7.3	Operation as a Frequency Divider (Channel 0 of Unit 0 only).....	210
16.7.4	Operation for Input Pulse Interval Measurement	213
16.7.5	Operation for Input Signal High- or Low-Level Width Measurement.....	216
16.7.6	Operation as a Delay Counter.....	220
16.8	Simultaneous Channel Operation Function of Timer Array Unit.....	223
16.8.1	Operation for the One-shot Pulse Output Function.....	223
16.8.2	Operation for the PWM Function.....	229
16.8.3	Operation for the Multiple PWM Output Function	235
16.9	Usage Notes.....	242
16.9.1	Cautions when Using Timer Output	242
17.	32-bit Interval Timer (TML32).....	243
17.1	Overview.....	243
17.2	Register Descriptions	245
17.2.1	ITLCMP0n/ITLCMP0n_L/ITLCMP0n_H : Interval Timer Compare Registers 0n (n = 0, 1). 245	
17.2.2	ITLCAP00 : Interval Timer Capture Register 00.....	245
17.2.3	ITLCTL0 : Interval Timer Control Register	246
17.2.4	ITLCSEL0 : Interval Timer Clock Select Register 0	247
17.2.5	ITLFDIV00 : Interval Timer Frequency Division Register 0	248
17.2.6	ITLFDIV01 : Interval Timer Frequency Division Register 1	249
17.2.7	ITLCC0 : Interval Timer Capture Control Register 0	250
17.2.8	ITLS0 : Interval Timer Status Register	250
17.2.9	ITLMKF0 : Interval Timer Match Detection Mask Register.....	252
17.3	Operation.....	252
17.3.1	Counter Mode Settings	252
17.3.2	Capture Mode Settings	254
17.3.3	Timer Operation	255
17.3.4	Capture Operation.....	255
17.3.5	Interrupt.....	256
17.3.6	Interval Timer Setting Procedures.....	258
18.	Independent Watchdog Timer (IWDT).....	260
18.1	Overview.....	260
18.2	Register Descriptions	261
18.2.1	IWDTRR : IWDT Refresh Register.....	261
18.2.2	IWDTSR : IWDT Status Register	262
18.2.3	OFS0 : Option Function Select Register 0	263
18.3	Operation.....	265
18.3.1	Auto Start Mode	265
18.3.2	Refresh Operation.....	266
18.3.3	Status Flags	267
18.3.4	Reset Output	268
18.3.5	Interrupt Sources.....	268
18.3.6	Reading the Down-Counter Value.....	268
18.4	Usage Notes.....	268
18.4.1	Refresh Operations	268
18.4.2	Clock Division Ratio Setting	269
19.	Serial Array Unit (SAU)	270

16.6.3	通道输入操作注意事项	203
16.7	定时器阵列单元的独立通道操作功能	203
16.7.1	作为间隔定时器或方波输出运行	203
16.7.2	作为外部事件计数器运行	207
16.7.3	作为分频器运行（仅限 0 号单元的 0 号通道）	210
16.7.4	输入脉冲间隔测量操作	213
16.7.5	输入信号高低电平宽度测量操作	216
16.7.6	作为延迟计数器运行	220
16.8	定时器阵列单元的同步通道操作功能	223
16.8.1	单次脉冲输出功能的操作	223
16.8.2	PWM功能的操作	229
16.8.3	多路PWM输出功能的操作	235
16.9	使用说明	242
16.9.1	使用定时器输出时的注意事项	242
17.32	位间隔定时器 (TML32)	243
17.1	概述	243
17.2	寄存器描述	245
17.2.1	ITLCMP0n/ITLCMP0n_L/ITLCMP0n_H: 间隔定时器比较寄存器 0n(n = 0, 1) 。 245	
17.2.2	ITLCAP00: 间隔定时器捕获寄存器 00	245
17.2.3	ITLCTL0: 间隔定时器控制寄存器	246
17.2.4	ITLCSEL0: 间隔定时器时钟选择寄存器 0	247
17.2.5	ITLFDIV00: 间隔定时器频率分频寄存器 0	248
17.2.6	ITLFDIV01: 间隔定时器频率分频寄存器 1	249
17.2.7	ITLCC0: 间隔定时器捕获控制寄存器 0	250
17.2.8	ITLS0: 间隔定时器状态寄存器	250
17.2.9	ITLMKF0: 间隔定时器匹配检测掩码寄存器	252
17.3	操作	252
17.3.1	计数器模式设置	252
17.3.2	捕获模式设置	254
17.3.3	定时器操作	255
17.3.4	捕获操作	255
17.3.5	中断	256
17.3.6	间隔定时器设置程序	258
18.	独立看门狗定时器 (IWDT)	260
18.1	概述	260
18.2	寄存器描述	261
18.2.1	IWDTRR: IWDT 刷新寄存器	261
18.2.2	IWDTSR: IWDT 状态寄存器	262
18.2.3	OFS0: 选项功能选择寄存器 0	263
18.3	操作	265
18.3.1	自动启动模式	265
18.3.2	刷新操作	266
18.3.3	状态标志	267
18.3.4	复位输出	268
18.3.5	中断源	268
18.3.6	读取向下计数器值	268
18.4	使用说明	268
18.4.1	刷新操作	268
18.4.2	时钟分频比设置	269
19.	串行阵列单元 (SAU)	270

19.1	Overview	270
19.1.1	Simplified SPI	270
19.1.2	UART	271
19.1.3	Simplified I ² C	271
19.2	Configuration of Serial Array Unit	272
19.3	Register Descriptions	274
19.3.1	SPSm : Serial Clock Select Register m (m = 0)	274
19.3.2	SMRmn : Serial Mode Register mn (mn = 00, 02)	275
19.3.3	SMRmn : Serial Mode Register mn (mn = 01, 03)	276
19.3.4	SCRm0 : Serial Communication Operation Setting Register m0 (m = 0)	277
19.3.5	SCRm1 : Serial Communication Operation Setting Register m1 (m = 0)	279
19.3.6	SCR02 : Serial Communication Operation Setting Register 02	281
19.3.7	SCR03 : Serial Communication Operation Setting Register 03	282
19.3.8	SDRmn : Serial Data Register mn (mn = 00, 01, 02, 03)	284
19.3.9	SIRmn : Serial Flag Clear Trigger Register mn (mn = 00, 02)	285
19.3.10	SIRmn : Serial Flag Clear Trigger Register mn (mn = 01, 03)	285
19.3.11	SSRmn : Serial Status Register mn (mn = 00, 02)	286
19.3.12	SSRmn : Serial Status Register mn (mn = 01, 03)	287
19.3.13	SS0 : Serial Channel Start Register 0	289
19.3.14	ST0 : Serial Channel Stop Register 0	289
19.3.15	SE0 : Serial Channel Enable Status Register 0	290
19.3.16	SOE0 : Serial Output Enable Register 0	290
19.3.17	SO0 : Serial Output Register 0	291
19.3.18	SOL0 : Serial Output Level Register 0	291
19.3.19	SSC0 : Serial Standby Control Register 0	292
19.3.20	ISC : Input Switch Control Register	293
19.3.21	SNFEN : SAU Noise Filter Enable Register	294
19.3.22	ULBS : UART Loopback Select Register	295
19.4	Operation Stop Mode	295
19.5	Operation of Simplified SPI	296
19.5.1	Master Transmission	297
19.5.2	Master Reception	304
19.5.3	Master Transmission and Reception	313
19.5.4	Slave Transmission	320
19.5.5	Slave Reception	328
19.5.6	Slave Transmission and Reception	333
19.5.7	Snooze Mode Function	342
19.5.8	Calculating Transfer Clock Frequency	345
19.5.9	Procedure for Processing Errors that Occurred During Simplified SPI Communication	347
19.6	Operation of UART Communication	347
19.6.1	UART Transmission	348
19.6.2	UART Reception	355
19.6.3	Snooze Mode Function	361
19.6.4	Calculating Baud Rate	367
19.6.5	Procedure for Processing Errors that Occurred During UART Communication	369
19.7	Operation of LIN Communication	370
19.7.1	LIN Transmission	370
19.7.2	LIN Reception	373
19.8	Operation of Simplified I ² C Communication	376
19.8.1	Address Field Transmission	377

19.1	概述	270
19.1.1	简化的SPI	270
19.1.2	UART	271
19.1.3	Simplified I ² C	271
19.2	串行阵列单元的配置	272
19.3	寄存器描述	274
19.3.1	SPSm: 串行时钟选择寄存器m (m = 0)	274
19.3.2	SMRmn: 串行模式寄存器 mn(mn = 00, 02)	275
19.3.3	SMRmn: 串行模式寄存器 mn(mn = 01, 03)	276
19.3.4	SCRm0: 串行通信操作设置寄存器m0 (m = 0)	277
19.3.5	SCRm1: 串行通信操作设置寄存器 m1(m = 0)	279
19.3.6	SCR02: 串行通信操作设置寄存器02	281
19.3.7	SCR03: 串行通信操作设置寄存器03	282
19.3.8	SDRmn: 串行数据寄存器 mn(mn = 00, 01, 02, 03)	284
19.3.9	SIRmn: 串行标志清除触发寄存器 mn(mn = 00, 02)	285
19.3.10	SIRmn: 串行标志清除触发寄存器 mn(mn = 01, 03)	285
19.3.11	SSRmn: 串行状态寄存器 mn(mn = 00, 02)	286
19.3.12	SSRmn: 串行状态寄存器 mn(mn = 01, 03)	287
19.3.13	SS0: 串行通道起始寄存器 0	289
19.3.14	ST0: 串行通道停止寄存器 0	289
19.3.15	SE0: 串行通道使能状态寄存器 0	290
19.3.16	SOE0: 串行输出使能寄存器 0	290
19.3.17	SO0: 串行输出寄存器 0	291
19.3.18	SOL0: 串行输出电平寄存器 0	291
19.3.19	SSC0: 串行待机控制寄存器 0	292
19.3.20	ISC: 输入开关控制寄存器	293
19.3.21	SNFEN: SAU 噪声滤波器使能寄存器	294
19.3.22	ULBS: UART 回环选择寄存器	295
19.4	操作停止模式	295
19.5	简化SPI的操作	296
19.5.1	主变速器	297
19.5.2	主接待处	304
19.5.3	主传输和接收	313
19.5.4	从动变速器	320
19.5.5	奴隶接收	328
19.5.6	从设备传输和接收	333
19.5.7	贪睡模式功能	342
19.5.8	计算传输时钟频率	345
19.5.9	简化SPI通信过程中发生的错误的处理程序	347
19.6	UART通信的操作	347
19.6.1	UART 传输	348
19.6.2	UART接收	355
19.6.3	贪睡模式功能	361
19.6.4	计算波特率	367
19.6.5	UART 通信过程中发生的错误的处理程序	369
19.7	LIN通信的操作	370
19.7.1	ILIN 变速器	370
19.7.2	LIN 接待处	373
19.8	简化通信的操作	376
19.8.1	地址字段传输	377

19.8.2	Data Transmission	381
19.8.3	Data Reception	384
19.8.4	Stop Condition Generation.....	389
19.8.5	Calculating Transfer Rate.....	390
19.8.6	Procedure for Processing Errors that Occurred during Simplified I ² C Communication	390
20.	I²C Bus Interface (IICA)	392
20.1	Overview.....	392
20.2	Register Descriptions	395
20.2.1	IICA0 : IICA Shift Register 0.....	395
20.2.2	SVA0 : Slave Address Register 0.....	396
20.2.3	IICCTL00 : IICA Control Register 00.....	396
20.2.4	IICS0 : IICA Status Register 0.....	400
20.2.5	IICF0 : IICA Flag Register 0.....	403
20.2.6	IICCTL01 : IICA Control Register 01.....	405
20.2.7	IICWL0 : IICA Low-level Width Setting Register 0	407
20.2.8	IICWH0 : IICA High-level Width Setting Register 0.....	407
20.2.9	Registers to Control the Port Function Multiplexed with the I ² C I/O Pins	408
20.3	I ² C Bus Definitions and Control Methods.....	408
20.3.1	Pin Configuration.....	408
20.3.2	Setting Transfer Clock Using IICWL0 and IICWH0 Registers.....	408
20.3.3	Start Conditions.....	409
20.3.4	Address	410
20.3.5	Transfer Direction Specification	410
20.3.6	Acknowledge (ACK)	411
20.3.7	Stop Condition.....	412
20.3.8	Clock Stretching.....	412
20.3.9	Release from Clock Stretching.....	414
20.3.10	Timing of Generation of the Interrupt Request Signal (IICA0_TXRXI) and Control of Clock Stretching.....	415
20.3.11	Address Match Detection Method	416
20.3.12	Error Detection	416
20.3.13	Extension Code.....	416
20.3.14	Arbitration.....	417
20.3.15	Wakeup Function	418
20.3.16	Communication Reservation	419
20.3.17	Usage Notes	421
20.3.18	Communication Operations.....	422
20.3.19	Timing of I ² C Interrupt Request Signal (IICA0_TXRXI) Occurrence	429
20.4	Timing Charts	444
21.	Cyclic Redundancy Check (CRC)	459
21.1	Overview.....	459
21.2	Register Descriptions	459
21.2.1	CRCCR0 : CRC Control Register 0	459
21.2.2	CRCDIR/CRCDIR_BY : CRC Data Input Register.....	460
21.2.3	CRCDOR/CRCDOR_HA : CRC Data Output Register	460
21.3	Operation.....	461
21.3.1	Basic Operation.....	461
21.4	Usage Notes.....	462
21.4.1	Settings for the Module-Stop State	462
21.4.2	Note on Transmission	462

19.8.2	数据传输	381
19.8.3	数据接收	384
19.8.4	停止条件生成	389
19.8.5	计算转账费率	390
19.8.6	简化通信过程中发生的错误的处理程序	390
20.	I²C总线接口 (IICA)	392
20.1	概述	392
20.2	寄存器描述	395
20.2.1	IICA0: IICA 班次登记簿 0	395
20.2.2	SVA0: 从机地址寄存器 0	396
20.2.3	IICCTL00: IICA 控制寄存器 00	396
20.2.4	IICS0: IICA 状态寄存器 0	400
20.2.5	IICF0: IICA 标志寄存器 0	403
20.2.6	IICCTL01: IICA 控制寄存器 01	405
20.2.7	IICWL0: IICA 低级宽度设置寄存器 0	407
20.2.8	IICWH0: IICA 高级宽度设置寄存器 0	407
20.2.9	用于控制与I ² C I/O 引脚复用的端口功能的寄存器	408
20.3	I2C总线定义和控制方法	408
20.3.1	引脚配置	408
20.3.2	使用 IICWL0 和 IICWH0 寄存器设置传输时钟	408
20.3.3	起始条件	409
20.3.4	地址	410
20.3.5	传输方向规范	410
20.3.6	确认 (ACK)	411
20.3.7	停止条件	412
20.3.8	点钟方向的拉伸	412
20.3.9	时钟拉伸的释放	414
20.3.10	中断请求信号 (IICA0_TXRXI) 的生成时序及对的控制 时钟拉伸	415
20.3.11	地址匹配检测方法	416
20.3.12	错误检测	416
20.3.13	扩展代码	416
20.3.14	仲裁	417
20.3.15	唤醒功能	418
20.3.16	通信预留	419
20.3.17	使用说明	421
20.3.18	通信操作	422
20.3.19	I ² C中断请求信号 (IICA0_TXRXI) 发生的时间	429
20.4	时序图	444
21.	循环冗余校验 (CRC)	459
21.1	概述	459
21.2	寄存器描述	459
21.2.1	CRCCR0: CRC 控制寄存器 0	459
21.2.2	CRCDIR/CRCDIR_BY: CRC 数据输入寄存器	460
21.2.3	CRCDOR/CRCDOR_HA: CRC 数据输出寄存器	460
21.3	操作	461
21.3.1	基本操作	461
21.4	使用说明	462
21.4.1	模块停止状态的设置	462
21.4.2	关于传输的说明	462

22. 10-bit A/D Converter (ADC10).....464

22.1 Overview.....464

22.2 Registers to Control the A/D Converter 468

22.2.1 ADM0 : A/D Converter Mode Register 0 468

22.2.2 ADM1 : A/D Converter Mode Register 1 477

22.2.3 ADM2 : A/D Converter Mode Register 2 478

22.2.4 ADCR: 10-bit A/D Conversion Result Register 480

22.2.5 ADCRH : 8-bit A/D Conversion Result Register.....480

22.2.6 ADS : Analog Input Channel Specification Register 481

22.2.7 ADUL : Conversion Result Comparison Upper Limit Setting Register.....482

22.2.8 ADLL : Conversion Result Comparison Lower Limit Setting Register 482

22.2.9 ADTES : A/D Test Register 482

22.3 A/D Converter Operations 483

22.4 Input Voltage and Conversion Results 484

22.5 A/D Converter Operation Modes 485

22.5.1 Software Trigger Mode (Select Mode, Sequential Conversion Mode) 485

22.5.2 Software Trigger Mode (Select Mode, One-shot Conversion Mode) 486

22.5.3 Hardware Trigger No-wait Mode (Select Mode, Sequential Conversion Mode).....487

22.5.4 Hardware Trigger No-wait Mode (Select Mode, One-shot Conversion Mode).....488

22.5.5 Hardware Trigger Wait Mode (Select Mode, Sequential Conversion Mode).....489

22.5.6 Hardware Trigger Wait Mode (Select Mode, One-shot Conversion Mode).....490

22.6 A/D Converter Setup Procedure.....491

22.6.1 Setting up Software Trigger Mode.....491

22.6.2 Setting up Hardware Trigger No-wait Mode 492

22.6.3 Setting up Hardware Trigger Wait Mode 493

22.6.4 Example of Using the ADC10 when Selecting the Temperature Sensor Output Voltage or Internal Reference Voltage, and Software Trigger Mode and One-shot Conversion Mode 494

22.6.5 Setting Up Test Mode 495

22.7 Snooze Mode Function.....496

22.7.1 A/D Conversion by Inputting a Hardware Trigger.....496

22.8 Testing of the A/D Converter 500

22.9 How to Read A/D Converter Characteristics Table.....500

22.10 Usage Notes.....503

23. Temperature Sensor (TSN) 507

23.1 Overview.....507

23.2 Using the Temperature Sensor.....507

23.2.1 Preparation for Using the Temperature Sensor.....507

23.2.2 Procedures for Using the Temperature Sensor.....508

24. SRAM.....509

24.1 Overview.....509

24.2 Operation.....509

24.2.1 Access Cycle.....509

24.3 Usage Notes.....509

24.3.1 Instruction Fetch from the SRAM Area 509

24.3.2 SRAM Store Buffer.....509

25. Flash Memory 510

25.1 Overview.....510

25.2 Memory Structure 511

25.3 Register Descriptions 511

25.3.1 FENTRYR : Flash P/E Mode Entry Register 511

10 位模数转换器 (ADC10) 464

22.1 概述 464

22.2 控制模数转换器的寄存器 468

22.2.1ADM0: A/D转换器模式寄存器0 468

22.2.2ADM1: A/D转换器模式寄存器1 477

22.2.3ADM2: A/D转换器模式寄存器2 478

22.2.4ADCR: 10 位 A/D 转换结果寄存器 480

22.2.5ADCRH: 8 位 A/D 转换结果寄存器 480

22.2.6ADS: 模拟输入通道规范寄存器 481

22.2.7ADUL: 转换结果比较上限设置寄存器 482

22.2.8ADLL: 转换结果比较下限设置寄存器 482

22.2.9ADTES: A/D 测试寄存器 482

22.3A/D转换器操作 483

22.4 输入电压和转换结果 484

22.5A/D转换器工作模式 485

22.5.1软件触发模式（选择模式、顺序转换模式） 485

22.5.2 软件触发模式（选择模式、单次转换模式） 486

22.5.3硬件触发无等待模式（选择模式、顺序转换模式） 487

22.5.4硬件触发无等待模式（选择模式、单次转换模式） 488

22.5.5硬件触发等待模式（选择模式、顺序转换模式） 489

22.5.6硬件触发等待模式（选择模式、单次转换模式） 490

22.6A/D转换器设置步骤 491

22.6.1 设置软件触发模式 491

22.6.2 设置硬件触发无等待模式 492

22.6.3 设置硬件触发等待模式 493

22.6.4 选择温度传感器输出电压时使用 ADC10 的示例
内部参考电压、软件触发模式和单次转换模式 494

22.6.5 设置测试模式 495

22.7 贪睡模式功能 496

22.7.1 通过输入硬件触发信号进行A/D转换 496

22.8 A/D转换器的测试 500

22.9 如何解读模数转换器特性表 500

22.10 使用说明 503

23.温度传感器（TSN） 507

23.1 概述 507

23.2 使用温度传感器 507

23.2.1 温度传感器使用前的准备工作 507

23.2.2 温度传感器的使用步骤 508

24. SRAM 509

24.1 概述 509

24.2 操作 509

24.2.1访问周期 509

24.3 使用说明 509

24.3.1 从 SRAM 区域获取指令 509

24.3.2 SRAM 存储缓冲区 509

25.闪存 510

25.1 概述 510

25.2 内存结构 511

25.3 寄存器描述 511

25.3.1FENTRYR: 闪存 P/E 模式输入寄存器 511

25.3.2	FPR : Protection Unlock Register	512	25.3.2FPR: 保护解锁寄存器	512
25.3.3	FPSR : Protection Unlock Status Register	512	25.3.3FPSR: 保护解锁状态寄存器	512
25.3.4	FPMCR : Flash P/E Mode Control Register	513	25.3.4FPMCR: 闪存 P/E 模式控制寄存器	513
25.3.5	FISR : Flash Initial Setting Register	514	25.3.5FISR: 闪存初始设置寄存器	514
25.3.6	FRESETR : Flash Reset Register	515	25.3.6FRESETR: 闪存复位寄存器	515
25.3.7	FASR : Flash Area Select Register	515	25.3.7FASR: 闪存区域选择寄存器	515
25.3.8	FCR : Flash Control Register	516	25.3.8FCR: 闪存控制寄存器	516
25.3.9	FEXCR : Flash Extra Area Control Register	517	25.3.9FEXCR: 闪存额外区域控制寄存器	517
25.3.10	FSARH : Flash Processing Start Address Register H.....	519	25.3.10FSARH: 闪存处理起始地址寄存器 H	519
25.3.11	FSARL : Flash Processing Start Address Register L.....	519	25.3.11FSARL: 闪存处理起始地址寄存器 L	519
25.3.12	FEARH : Flash Processing End Address Register H.....	520	25.3.12FEARH: 闪存处理结束地址寄存器 H	520
25.3.13	FEARL : Flash Processing End Address Register L.....	520	25.3.13FEARL: 闪存处理结束地址寄存器 L	520
25.3.14	FWBL0 : Flash Write Buffer Register L0	520	25.3.14FWBL0: 闪存写入缓冲寄存器 L0	520
25.3.15	FWBH0 : Flash Write Buffer Register H0	521	25.3.15FWBH0: 闪存写入缓冲区寄存器 H0	521
25.3.16	FSTATR1 : Flash Status Register 1	521	25.3.16FSTATR1: 闪存状态寄存器 1	521
25.3.17	FSTATR2 : Flash Status Register 2	522	25.3.17FSTATR2: 闪存状态寄存器 2	522
25.3.18	FEAMH : Flash Error Address Monitor Register H.....	523	25.3.18FEAMH: 闪存错误地址监视器寄存器 H	523
25.3.19	FEAML : Flash Error Address Monitor Register L.....	523	25.3.19FEAML: 闪存错误地址监视器寄存器 L	523
25.3.20	FSCMR : Flash Startup Setting Monitor Register	523	25.3.20FSCMR: Flash 启动设置监视器寄存器	523
25.3.21	FAWSMR : Flash Access Window Start Address Monitor Register	524	25.3.21FAWSMR: 闪存访问窗口起始地址监视器寄存器	524
25.3.22	FAWEMR : Flash Access Window End Address Monitor Register	524	25.3.22FAWEMR: 闪存访问窗口结束地址监视器寄存器	524
25.3.23	UIDRn : Unique ID Registers n (n = 0 to 3).....	525	25.3.23 UIDRn: 唯一 ID 寄存器 n(n = 0至 3)	525
25.3.24	PNRn : Part Numbering Register n (n = 0 to 3).....	525	25.3.24PNRn: 零件编号寄存器n (n = 0 至 3)	525
25.3.25	MCUVER : MCU Version Register	526	25.3.25MCUVER: MCU 版本寄存器	526
25.4	Operating Modes Associated with the Flash Memory	526	25.4 与闪存相关的操作模式	526
25.4.1	ID Code Protection	526	25.4.1 ID 代码保护	526
25.5	Overview of Functions	527	25.5 函数概述	527
25.5.1	Configuration Area Bit Map	528	25.5.1 配置区域位图	528
25.5.2	Startup Area Select	529	25.5.2 启动区域选择	529
25.5.3	Protection by Access Window	529	25.5.3 通过访问窗口进行保护	529
25.6	Programming Commands.....	530	25.6 编程命令	530
25.7	Suspend Operation.....	530	25.7 暂停操作	530
25.8	Protection	530	25.8 保护	530
25.8.1	Startup Program Protection.....	530	25.8.1 初创企业保护	530
25.8.2	Area Protection	531	25.8.2 区域保护	531
25.9	Self-programming	531	25.9 自编程	531
25.9.1	Overview	531	25.9.1 概述	531
25.10	Programming and Erasure	532	25.10 编程与擦除	532
25.10.1	Sequencer Modes	532	25.10.1 音序器模式	532
25.10.2	Software Commands.....	533	25.10.2 软件命令	533
25.10.3	Software Command Usage	533	25.10.3 软件命令用法	533
25.11	Reading the Flash Memory	539	25.11 读取闪存	539
25.11.1	Reading the Code Flash Memory	539	25.11.1 读取代码闪存	539
25.12	Usage Notes.....	539	25.12 使用说明	539
25.12.1	Erase Suspended Area	539	25.12.1 擦除悬浮区域	539
25.12.2	Constraints on Additional Writes	540	25.12.2 额外写入的限制	540
25.12.3	Reset during Programming and Erasure.....	540	25.12.3 编程和擦除过程中的重置	540
25.12.4	Non-Maskable Interrupt Disabled during Programming and Erasure	540	25.12.4 编程和擦除期间禁用不可屏蔽中断	540
25.12.5	Location of Interrupt Vectors during Programming and Erasure	540	25.12.5 编程和擦除期间中断向量的位置	540
25.12.6	Programming and Erasure in Subosc-speed Operating Mode	540	25.12.6 亚振荡器速度操作模式下的编程和擦除	540

25.12.7	Abnormal Termination during Programming and Erasure	540
25.12.8	Actions Prohibited during Programming and Erasure	540
25.12.9	Flash-IF clock (ICLK) during Program/Erase	540
26.	Internal Voltage Regulator	541
26.1	Overview.....	541
26.2	Operation.....	541
27.	Electrical Characteristics.....	542
27.1	Absolute Maximum Ratings.....	542
27.1.1	Tj/Ta Definition	543
27.2	Oscillators Characteristics	544
27.2.1	On-chip Oscillators Characteristics	544
27.3	DC Characteristics.....	544
27.3.1	Pin Characteristics	544
27.3.2	Operating and Standby Current	548
27.3.3	Thermal Characteristics	551
27.4	AC Characteristics.....	551
27.4.1	Reset Timing	554
27.4.2	Wakeup Time	555
27.5	Peripheral Function Characteristics.....	558
27.5.1	Serial Array Unit (SAU)	558
27.5.2	I ² C Bus Interface (IICA)	580
27.6	Analog Characteristics.....	582
27.6.1	A/D Converter Characteristics for T _A = -40 to +85°C	582
27.6.2	A/D Converter Characteristics for T _A = -40 to +125°C	586
27.6.3	Temperature Sensor/Internal Reference Voltage Characteristics	590
27.6.4	POR Characteristics.....	590
27.6.5	LVD Characteristics.....	591
27.6.6	Power Supply Voltage Rising Slope Characteristics	593
27.7	RAM Data Retention Characteristics.....	593
27.8	Flash Memory Programming Characteristics	593
27.9	Serial Wire Debug (SWD).....	594
Appendix 1.	Port States in each Processing Mode.....	596
Appendix 2.	Package Dimensions	598
Appendix 3.	I/O Registers	599
3.1	Peripheral Base Addresses	599
3.2	Access Cycles	599
Appendix 4.	Peripheral Variant	601

25.12.7	编程和擦除过程中的异常终止	540
25.12.8	编程和擦除期间禁止的操作	540
25.12.9	编程/擦除期间的闪存-IF时钟 (ICLK)	540
26.	内部电压调节器	541
26.1	概述	541
26.2	操作	541
27.	电气特性	542
27.1	绝对最大额定值	542
27.1.1	Tj/Ta 定义	543
27.2	振荡器特性	544
27.2.1	片上振荡器特性	544
27.3	直流特性	544
27.3.1	引脚特性	544
27.3.2	工作电流和待机电流	548
27.3.3	热特性	551
27.4	AC特性	551
27.4.1	复位时序	554
27.4.2	起床时间	555
27.5	外周功能特征	558
27.5.1	串行阵列单元 (SAU)	558
27.5.2	I ² C 总线接口 (IICA)	580
27.6	模拟特性	582
27.6.1	A/D转换器特性 T _A = -40 to +85°C	582
27.6.2	T _A = -40 to +125°C的 A/D 转换器特性	586
27.6.3	温度传感器/内部参考电压特性	590
27.6.4	POR特性	590
27.6.5	LVD特性	591
27.6.6	电源电压上升斜率特性	593
27.7	RAM 数据保持特性	593
27.8	闪存编程特性	593
27.9	串行线调试 (SWD)	594
附录 1.	各处理模式下的端口状态	596
附录 2.	包装尺寸	598
附录 3.	I/O 寄存器	599
3.1	外围设备基地址	599
3.2	访问周期	599
附录 4.	外周变异	601

Ultra low power 32 MHz Arm® Cortex®-M23 core, 16-KB code flash memory, 2-KB SRAM, 10-bit A/D Converter, Serial interfaces and Safety features.

Features

■ Arm Cortex-M23 Core

- Armv8-M architecture
- Maximum operating frequency: 32 MHz
- Debug and Trace: DWT, FPB, CoreSight™ MTB-M23
- CoreSight Debug Port: SW-DP

■ Memory

- 16-KB code flash memory
- 2-KB SRAM
- Flash read protection (FRP)
- 128-bit unique ID

■ Connectivity

- Serial Array Unit (SAU)
 - Simplified SPI × 2
 - Simplified IIC × 2
 - UART × 1
 - UART (LIN-bus supported) × 1
- I²C Bus interface (IICA) × 1

■ Analog

- 10-bit A/D Converter (ADC10)
- Temperature Sensor (TSN)

■ Timers

- 16-bit Timer Array Unit (TAU) × 8
- 32-bit interval timer (TML32) × 1
 - 1 channel in 32-bit counter mode
 - 2 channels in 16-bit counter mode
 - 4 channels in 8-bit counter mode

■ Safety

- Flash area protection
- ADC self-diagnosis function
- Cyclic Redundancy Check (CRC)
- Independent Watchdog Timer (IWDT)
- GPIO read-back level detection
- Register write protection
- Illegal memory access detection

■ System and Power Management

- Low power modes
- Data Transfer Controller (DTC)
- Power-on reset
- Low Voltage Detection (LVD) with voltage settings

■ Multiple Clock Sources

- High-speed on-chip oscillator (HOCO) (32 MHz)
- Low-speed on-chip oscillator (LOCO) (32.768 kHz)
- Clock trim function for HOCO/LOCO
- Clock out support

■ Up to 17 pins for general I/O ports

- 5-V tolerance, open drain, input pull-up

■ Operating Voltage

- VCC: 1.6 to 5.5 V

■ Operating Temperature and Packages

- Ta = -40°C to +125°C
 - 20-pin TSSOP (4.4 mm × 6.5 mm, 0.65 mm pitch)

超低功耗 32 MHz Arm® Cortex®-M23 内核，16 KB 代码闪存，2 KB SRAM，10 位 A/D 转换器，串行接口和安全功能。

特征

■ Arm Cortex-M23核心

- Armv8-M架构
- 最大工作频率：32 MHz
- 调试和跟踪：DWT、FPB、CoreSight™ MTB-M23
- CoreSight调试端口：SW-DP

■记忆

- 16KB 代码闪存
- 2-KB SRAM
- 闪存读取保护（FRP）
- 128 位唯一标识符

■连接性

- 串行阵列单元（SAU）
 - 简化版SPI × 2
 - 简化的 IIC × 2
 - UART × 1
 - UART（支持 LIN 总线）× 1
- I²C 总线接口(IICA) × 1

■模拟

- 10位模数转换器（ADC10）
- 温度传感器（TSN）

■定时器

- 16 位定时器阵列单元(TAU) × 8
- 32位间隔定时器(TML32) × 1
 - 1 个通道，32 位计数器模式
 - 2 个通道，16 位计数器模式
 - 4 个通道，8 位计数器模式

■安全

- 闪光区域保护
- ADC自诊断功能
- 循环冗余校验（CRC）
- 独立看门狗定时器（IWDT）
- GPIO 读取回读电平检测
- 寄存器写保护
- 非法内存访问检测

■系统和电源管理

- 低功耗模式
- 数据传输控制器（DTC）
- 上电重置
- 低电压检测（LVD）及电压设置

■多个时钟源

- 高速片上振荡器（HOCO）（32 MHz）
- 低速片上振荡器（LOCO）（32.768 kHz）
- HOCO/LOCO 的时钟微调功能
- 下班支持

■最多可使用17个引脚的通用I/O端口

- 5V容差，开漏，输入上拉

■工作电压

- VCC: 1.6 至5.5 V

■工作温度和封装

- Ta = -40°C 至+125°C
 - 20引脚 TSSOP(4.4 mm × 6.5 mm,0.65 mm 间距)

1. Overview

The MCU integrates multiple series of software- and pin-compatible Arm®-based 32-bit cores that share a common set of Renesas peripherals to facilitate design scalability.

The MCU in this series incorporates an energy-efficient Arm Cortex®-M23 32-bit core, that is particularly well suited for cost-sensitive and low-power applications, with the following features:

- 16-KB code flash memory
- 2-KB SRAM
- Serial Interface (SAU, IICA)
- General Purpose Timer (TAU, TML32)
- 10-bit A/D Converter (ADC10)

1.1 Function Outline

Table 1.1 Arm core

Feature	Functional description
Arm Cortex-M23 core	● Maximum operating frequency: up to 32 MHz ● Arm Cortex-M23 core: – Revision: r1p0-00rel0 – Armv8-M architecture profile – Single-cycle integer multiplier – 19-cycle integer divider ● SysTick timer: – Driven by SYSTICCLK (LOCO) or ICLK

Table 1.2 Memory

Feature	Functional description
Code flash memory	16-KB of code flash memory. See section 25, Flash Memory .
Option-setting memory	The option-setting memory determines the state of the MCU after a reset. See section 6, Option-Setting Memory .
SRAM	On-chip SRAM See section 24, SRAM .

Table 1.3 System (1 of 2)

Feature	Functional description
Operating modes	Operating mode: ● Single-chip mode See section 3, Operating Modes .
Resets	The MCU provides 6 resets (RES pin reset, power-on reset, independent watchdog timer reset, voltage monitor 0/1 resets, software reset). See section 5, Resets .
Low Voltage Detection (LVD)	The Low Voltage Detection (LVD) module monitors the voltage level input to the VCC pin. The detection level can be selected by register settings. The LVD module consists of two separate voltage level detectors (LVD0, LVD1). LVD0 and LVD1 measure the voltage level input to the VCC pin. LVD registers allow your application to configure detection of VCC changes at various voltage thresholds. See section 7, Low Voltage Detection (LVD) .
Clocks	● High-speed on-chip oscillator (HOCO) ● Low-speed on-chip oscillator (LOCO) ● Clock output / Buzzer output support See section 8, Clock Generation Circuit .
Interrupt Controller Unit (ICU)	The Interrupt Controller Unit (ICU) controls which event signals are linked to the Nested Vector Interrupt Controller (NVIC), and the Data Transfer Controller (DTC) modules. The ICU also controls non-maskable interrupts. See section 11, Interrupt Controller Unit (ICU) .

1. 概述

MCU 集成了多个系列的软件和引脚兼容的Arm®32 位内核，这些内核共享一组通用的瑞萨外设，以方便设计的可扩展性。

该系列MCU采用节能型Arm Cortex®-M23 32位内核，特别适用于对成本敏感和低功耗的应用，具有以下特点：

- 16KB 代码闪存
- 2-KB SRAM
- 串行接口（SAU、IICA）
- 通用定时器（TAU、TML32）
- 10位模数转换器（ADC10）

1.1 功能概述

表 1.1 臂核心

特征	功能描述
Arm Cortex-M23 核心	●最高工作频率：高达 32 MHz ●Arm Cortex-M23 核心： –修订版： r1p0-00rel0 –Armv8-M架构配置文件 单周期整数乘法器 -19周期整数除法器 ●系统滴答定时器： –由 SYSTICCLK (LOCO) 或 ICLK 驱动

表 1.2 内存

特征	功能描述
代码闪存	16KB 代码闪存。参见第 25 节，闪存。
选项设置内存	选项设置存储器决定MCU复位后的状态。 请参阅第 6 节“选项设置内存”。
SRAM	片上 SRAM 参见第 24 节，SRAM。

表 1.3 系统 (1/2)

特征	功能描述
操作模式	操作模式： ●单芯片模式参见第3节，操作模式。
重置	MCU 提供 6 种复位方式（RES 引脚复位、上电复位、独立看门狗定时器复位、电压监视器0/1复位、软件复位）。 请参阅第 5 节“重置”。
低电压检测（LVD）	低电压检测 (LVD) 模块监控输入到 VCC 引脚的电压电平。检测电平可通过寄存器设置进行选择。LVD 模块包含两个独立的电压电平检测器 (LVD0 和 LVD1)。LVD0 和 LVD1 测量输入到 VCC 引脚的电压电平。LVD 寄存器允许您的应用程序配置在不同的电压阈值下检测 VCC 变化。 参见第 7 节，低电压检测 (LVD)。
时钟	●高速片上振荡器（HOCO） ●低速片上振荡器（LOCO） ●时钟输出/蜂鸣器输出支持请参阅第 8 节，时钟生成电路。
中断控制器单元（ICU）	中断控制器单元 (ICU) 控制哪些事件信号链接到嵌套向量中断控制器 (NVIC) 和数据传输控制器 (DTC) 模块。ICU 还控制不可屏蔽中断。 请参阅第 11 节，中断控制器单元 (ICU)。

Table 1.3 System (2 of 2)

Feature	Functional description
Low power modes	Power consumption can be reduced in multiple ways, including setting clock dividers, stopping modules, selecting power control mode in normal operation, and transitioning to low power modes. See section 9, Low Power Modes .
Register write protection	The register write protection function protects important registers from being overwritten due to software errors. The registers to be protected are set with the Protect Register (PRCR). See section 10, Register Write Protection .
Flash Read Protection	The MCU incorporates the flash read protection with one secure regions that include the code flash. The secure region can be protected from non-secure program accesses. A non-secure program cannot access a protected region. See section 13, Flash Read Protection (FRP) .
Independent Watchdog Timer (IWDT)	The Independent Watchdog Timer (IWDT) consists of a 14-bit down counter that must be serviced periodically to prevent counter underflow. The IWDT provides functionality to reset the MCU or to generate a non-maskable interrupt or an underflow interrupt. Because the timer operates with the LOCO, it is particularly useful in returning the MCU to a known state as a fail-safe mechanism when the system runs out of control. The IWDT can be triggered automatically by a reset, underflow, refresh error, or a refresh of the count value in the registers. See section 18, Independent Watchdog Timer (IWDT) .

Table 1.4 Direct memory access

Feature	Functional description
Data Transfer Controller (DTC)	A Data Transfer Controller (DTC) module is provided for transferring data when activated by an interrupt request. See section 14, Data Transfer Controller (DTC) .

Table 1.5 Timers

Feature	Functional description
Timer Array Unit (TAU)	The timer array unit has eight 16-bit timers. Each 16-bit timer is called a channel and can be used as an independent timer. In addition, two or more channels can be used to create a High functional timer. See section 16, Timer Array Unit (TAU) .
32-bit Interval Timer (TML32)	The 32-bit interval timer is made up of four 8-bit interval timers (referred to as channels 0 to 3). Each is capable of operating independently and in that case they all have the same functions. Two 8-bit interval timer channels can be connected to operate as a 16-bit interval timer. Four 8-bit interval timer channels can be connected to operate as a 32-bit interval timer. See section 17, 32-bit Interval Timer (TML32) .

Table 1.6 Communication interfaces

Feature	Functional description
Serial Array Unit (SAU)	A Serial Array Unit (SAU) has one unit. Unit0 has four channels. Each channel can achieve simplified SPI, UART, or simplified IIC. See section 19, Serial Array Unit (SAU) .
I ² C Bus Interface (IICA)	The I ² C Bus Interface (IICA) has 1 channel. The IICA module conforms I ² C (Inter-Integrated Circuit) Bus Interface functions. See section 20, I²C Bus Interface (IICA) .

Table 1.7 Analog (1 of 2)

Feature	Functional description
10-bit A/D Converter (ADC10)	A 10-bit successive approximation A/D converter is provided. Up to 6 analog input channels are selectable. Temperature sensor output and internal reference voltage are selectable for conversion. See section 22, 10-bit A/D Converter (ADC10) .

表 1.3 系统 (2/2)

特征	功能描述
低功耗模式	可以通过多种方式降低功耗，包括设置时钟分频器、停止模块、在正常运行中选择电源控制模式以及过渡到低功耗模式。 请参阅第 9 节“低功耗模式”。
寄存器写保护	寄存器写保护功能可防止重要寄存器因软件错误而被写入。需要保护的寄存器通过保护寄存器（PRCR）进行设置。 请参阅第 10 节，寄存器写保护。
闪存读取保护	该MCU集成了闪存读取保护功能，并包含一个安全区域，该安全区域存储着代码闪存。安全区域可防止非安全程序访问。非安全程序无法访问受保护的区域。 请参阅第 13 节，闪存读取保护 (FRP)。
独立看门狗定时器（IWDT）	独立看门狗定时器 (IWDT) 由一个 14 位递减计数器组成，必须定期进行维护以防止计数器下溢。IWDT 提供复位 MCU、生成不可屏蔽中断或下溢中断的功能。由于该定时器与 LOCO 协同工作，因此在系统失控时，它作为一种故障保护机制，能够有效地将 MCU 恢复到已知状态。IWDT 可以由复位、下溢、刷新错误或寄存器中计数值的刷新自动触发。 请参阅第 18 节，独立看门狗定时器 (IWDT)。

表 1.4 直接内存访问

特征	功能描述
数据传输控制器（DTC）	提供数据传输控制器（DTC）模块，用于在中断请求激活时传输数据。 请参阅第 14 节“数据传输控制器 (DTC)”。

表 1.5 计时器

特征	功能描述
定时器阵列单元（TAU）	定时器阵列单元包含八个 16 位定时器。每个 16 位定时器称为一个通道，可以作为独立的定时器使用。此外，两个或多个通道可以组合成一个高电平功能定时器。 请参见第 16 节“定时器阵列单元 (TAU)”。
32 位间隔定时器 (TML32)	32 位间隔定时器由四个 8 位间隔定时器（称为通道 0 到 3）组成。每个定时器都可以独立运行，此时它们的功能相同。两个 8 位间隔定时器通道可以连接起来，组成一个 16 位间隔定时器。四个 8 位间隔定时器通道可以连接起来，组成一个 32 位间隔定时器。 请参阅第 17 节，32 位间隔定时器 (TML32)。

表 1.6 通信接口

特征	功能描述
串行阵列单元（SAU）	串行阵列单元（SAU）包含一个单元。单元0有四个通道。每个通道可以实现简化的SPI、UART或简化的IIC。 参见第 19 节，串行阵列单元 (SAU) 。
I ² C 总线接口 (IICA)	I ² C总线接口 (IICA) 有 1 个通道。IICA 模块符合I ² C(集成电路间) 总线接口功能。 请参阅第 20 节，I²C总线接口 (IICA)。

表 1.7 模拟 (1/2)

特征	功能描述
10 位模数转换器 (ADC10)	本产品配备一个10位逐次逼近型模数转换器。最多可选择6个模拟输入通道。温度传感器输出和内部参考电压均可用于转换。 请参阅第 22 节，10 位 A/D 转换器 (ADC10)。

Table 1.7 Analog (2 of 2)

Feature	Functional description
Temperature Sensor (TSN)	The on-chip Temperature Sensor (TSN) determines and monitors the die temperature for reliable operation of the device. The sensor outputs a voltage directly proportional to the die temperature, and the relationship between the die temperature and the output voltage is fairly linear. The output voltage is provided to the ADC10 for conversion and can be further used by the end application. See section 23, Temperature Sensor (TSN) .

Table 1.8 Data processing

Feature	Functional description
Cyclic Redundancy Check (CRC) calculator	The Cyclic Redundancy Check (CRC) generates CRC codes to detect errors in the data. Two CRC-generation polynomials (CRC-CCITT, CRC-32) are available. See section 21, Cyclic Redundancy Check (CRC) .

Table 1.9 I/O ports

Feature	Functional description
I/O ports	- I/O ports for the 20-pin TSSOP - I/O pins: 16 - Input pins: 1 - Pull-up resistors: 12 - N-ch open-drain outputs: 11

表 1.7 模拟 (2/2)

特征	功能描述
温度传感器 (TSN)	片上温度传感器 (TSN) 用于检测和监控芯片温度，以确保器件可靠运行。该传感器输出的电压与芯片温度成正比，且芯片温度与输出电压之间的关系基本呈线性。输出电压被送至 ADC10 进行转换，转换后的信号可供最终应用使用。 参见第 23 节，温度传感器 (TSN)。

表 1.8 数据处理

特征	功能描述
循环冗余校验 (CRC) 计算器	循环冗余校验 (CRC) 生成CRC码以检测数据中的错误。目前有两种CRC生成多项式 (CRC-CCITT和CRC-32) 可用。 参见第 21 节，循环冗余校验 (CRC)。

表 1.9 I/O 端口

特征	功能描述
I/O端口	20引脚TSSOP封装的I/O端口 - I/O引脚：16个 - 输入引脚：1 - 上拉电阻：12 - N沟道开漏输出：11

1.2 Block Diagram

Figure 1.1 shows a block diagram of the MCU superset. Some individual devices within the group have a subset of the features.

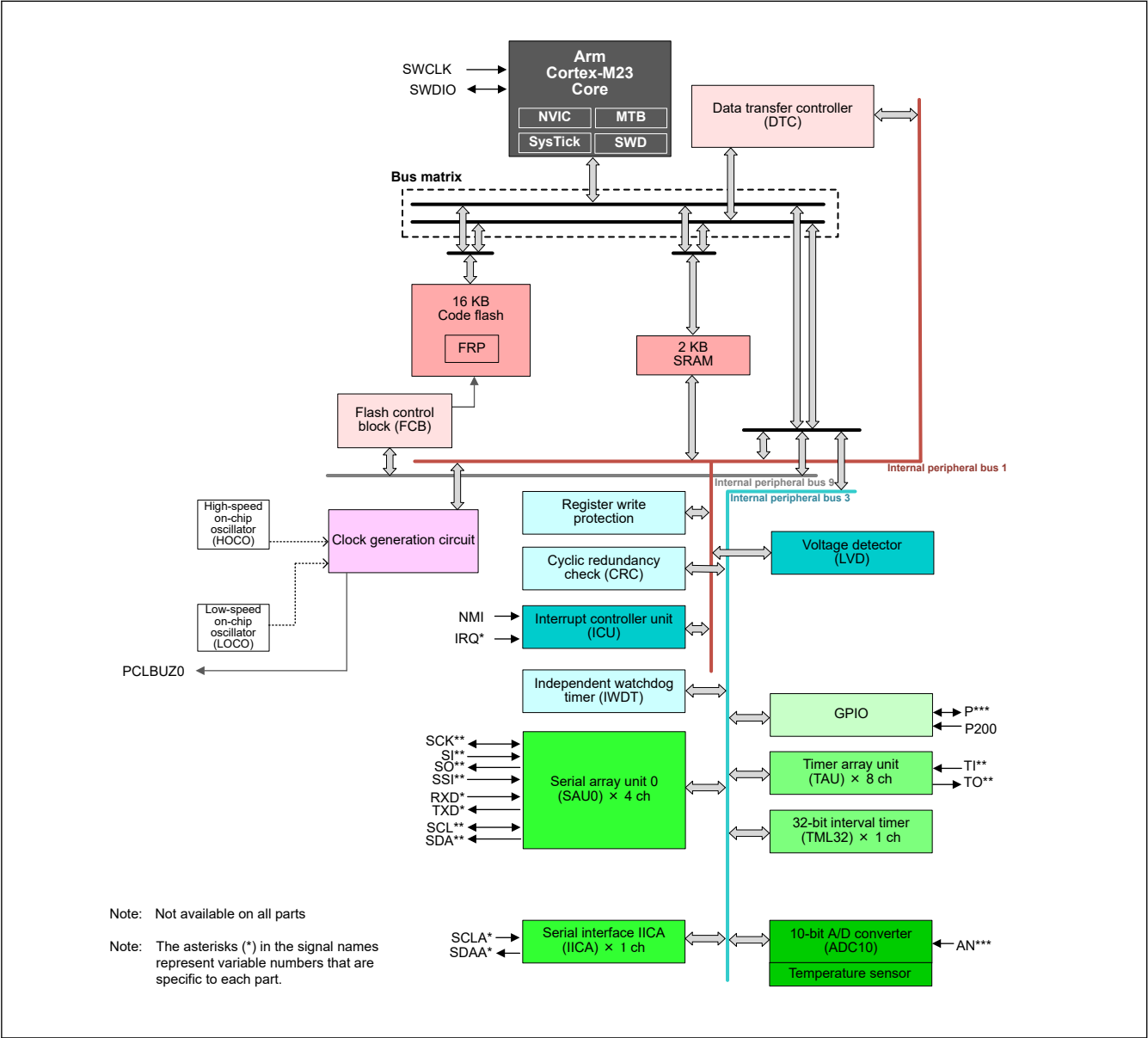


Figure 1.1 Block diagram

1.3 Part Numbering

Figure 1.2 shows the product part number information, including memory capacity and package type. Table 1.10 shows a list of products.

1.2框图

图 1.1 显示了 MCU 超集的框图。该组内的一些单个器件具有部分功能。

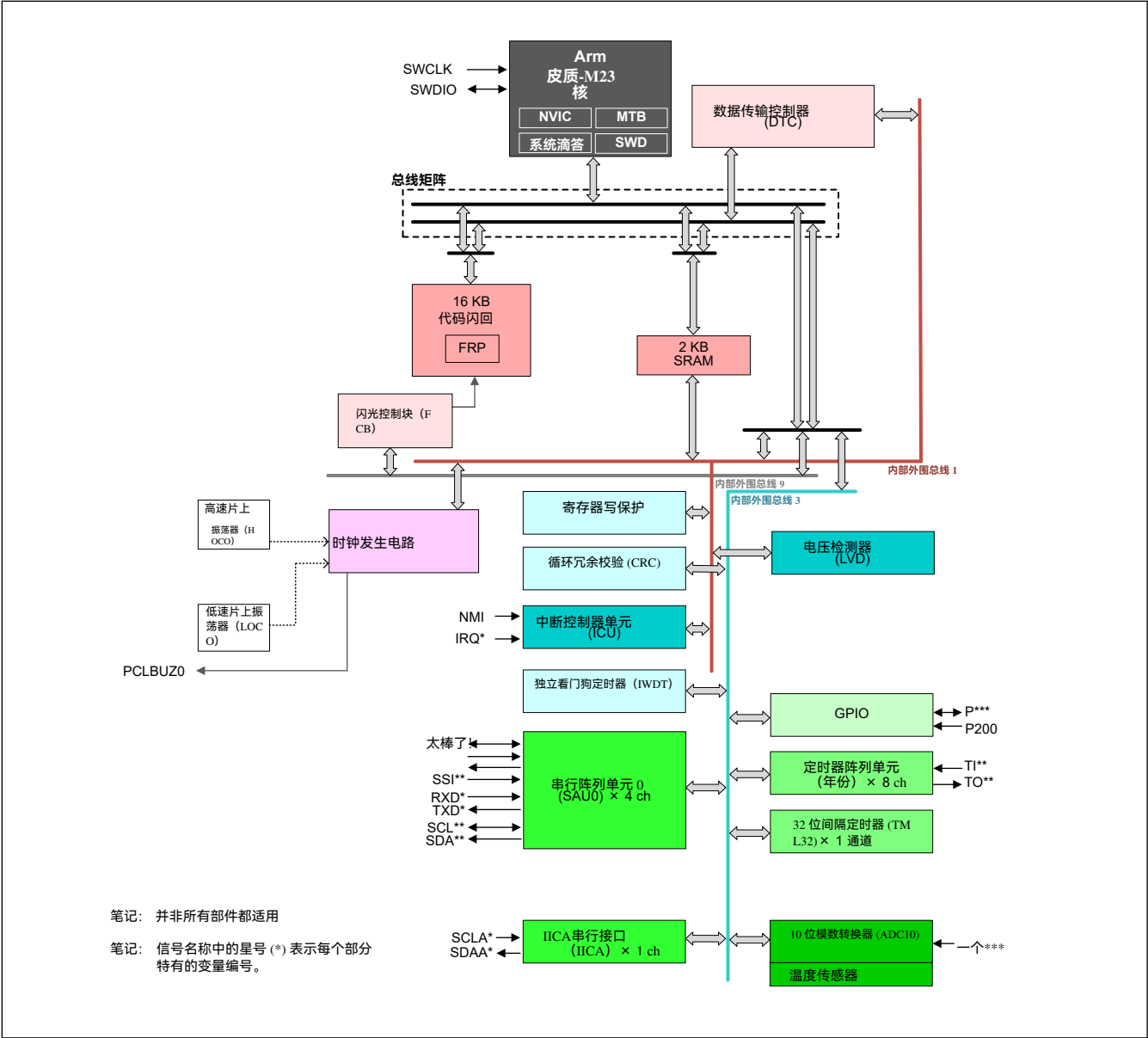


图 1.1 方框图

1.3零件编号

图 1.2 显示了产品部件号信息，包括内存容量和封装类型。表 1.10 列出了产品清单。

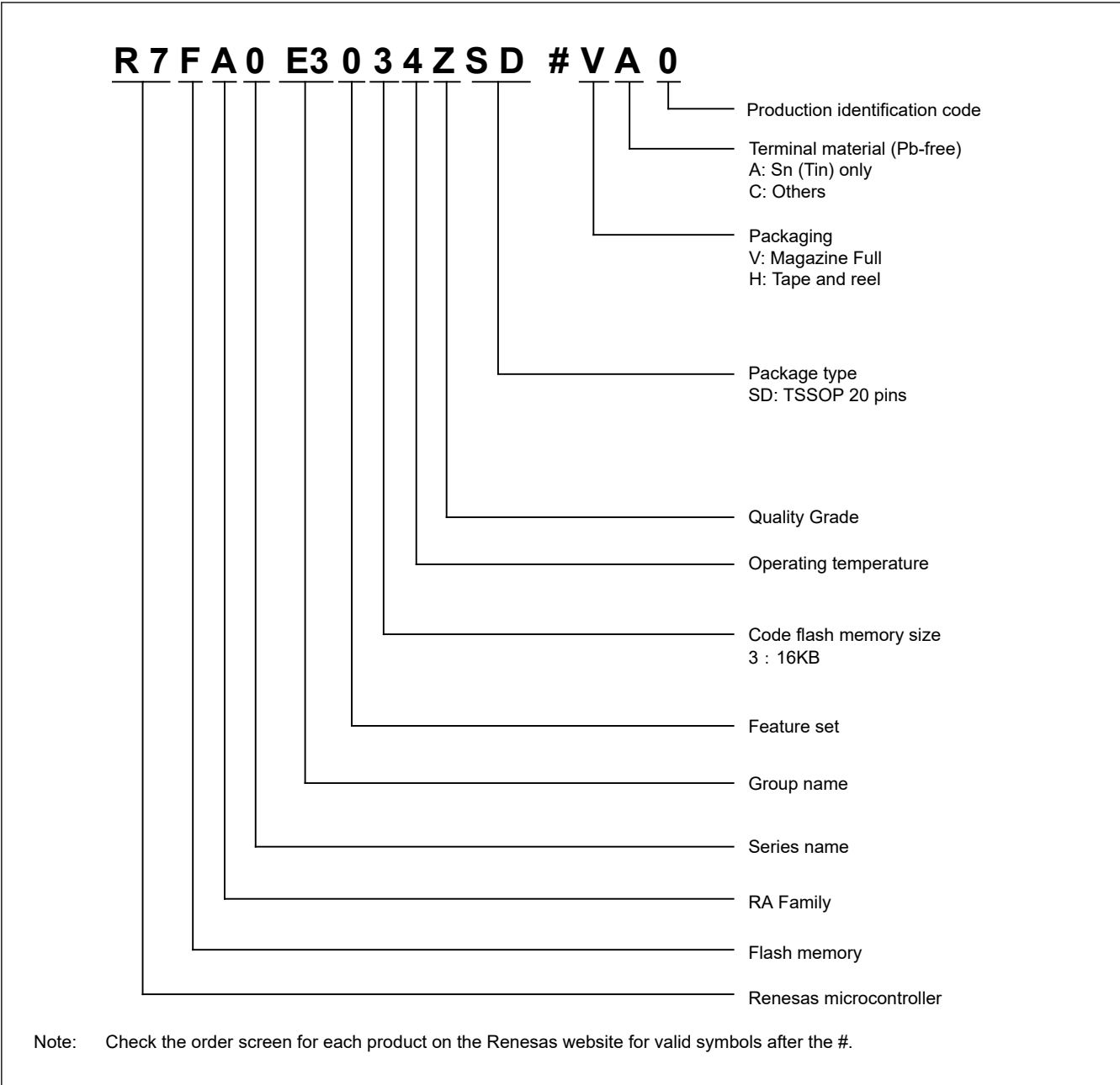


Figure 1.2 Part numbering scheme

Table 1.10 Product list

Product part number	Package code	Code flash	SRAM	Operating temperature
R7FA0E3034ZSD	PTSP0020JI-A	16 KB	2 KB	-40 to +125°C

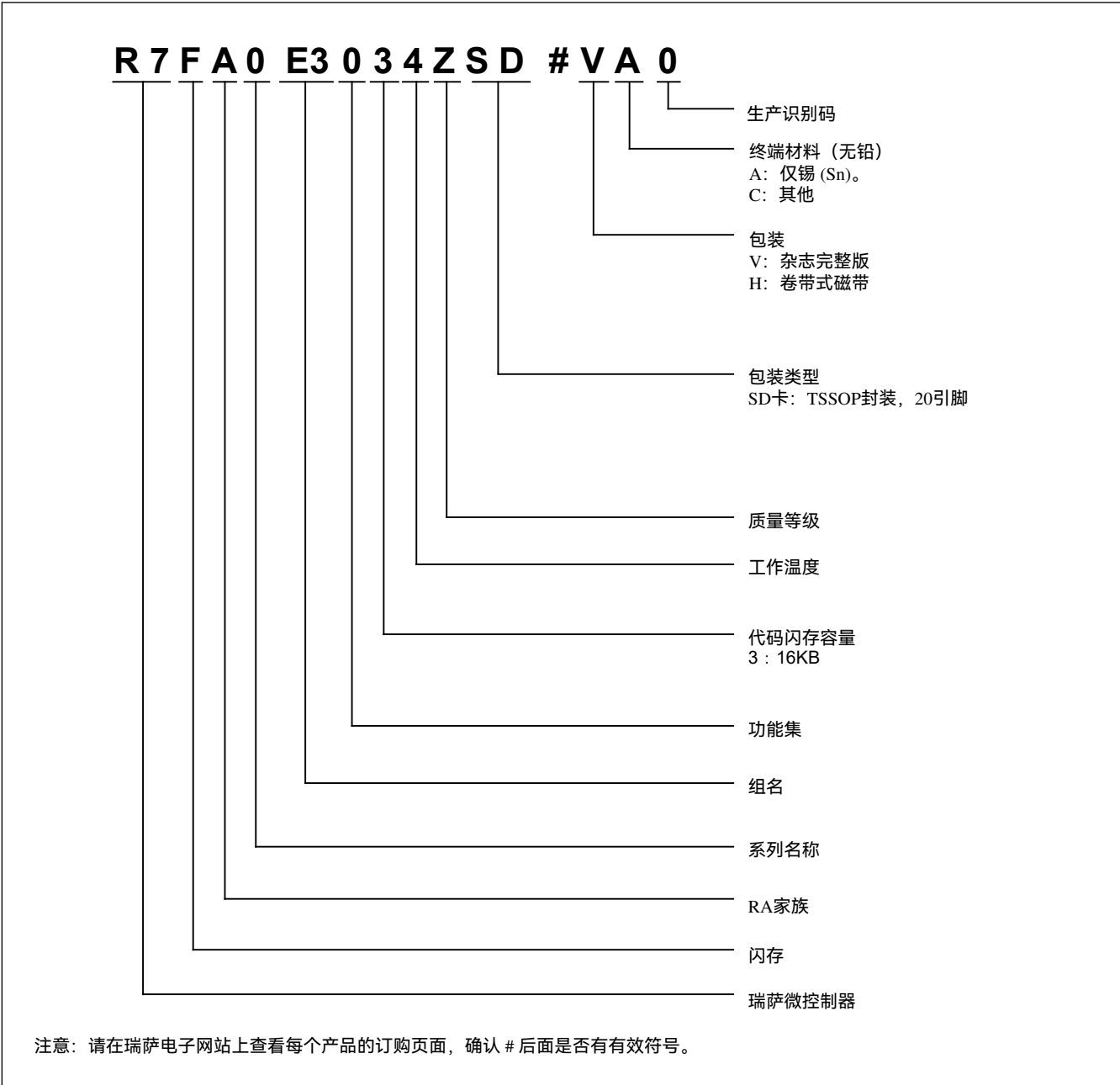


图 1.2 零件编号方案

表 1.10 产品清单

产品零件编号	包装代码	代码闪存	SRAM	工作温度
R7FA0E3034ZSD	PTSP0020JI-A	16 KB	2 KB	-40 to +125°C

1.4 Function Comparison

Table 1.11 Function comparison

Parts number		R7FA0E3034ZSD
Pin count		20
Package		TSSOP
Code flash memory		16 KB
SRAM		2 KB
System	CPU clock	32 MHz
	ICU	Yes
DMA	DTC	Yes
Timers	TAU	8 (PWM outputs: 7)
	TML32	1 (32-bit counter mode), 2 (16-bit counter mode), 4 (8-bit counter mode)
	IWDT	Yes
Communication	SAU	2 (simplified SPI)* ¹ , 2 (simplified IIC)* ¹ , 1 (UART)* ¹ , 1 (UART supporting LIN-bus)* ¹
	IICA	1
Analog	ADC10	6
	TSN	Yes
Data processing	CRC	Yes
I/O ports	I/O pins	16
	Input pins	1
	Pull-up resistors	12
	N-ch open-drain outputs	11
	5-V tolerance	—

Note 1. SAU consists of several channels. Each channel can be assigned only one function at a time.

1.4 功能比较

表 1.11 功能比较

零件编号		R7FA0E3034ZSD
保龄球		20
包裹		TSSOP
代码闪存		16 KB
SRAM		2 KB
系统	CPU时钟	32 MHz
	ICU	Yes
DMA	DTC	Yes
定时器	TAU	8 (PWM 输出: 7)
	TML32	1 (32 位计数器模式), 2 (16 位计数器模式), 4 (8 位计数器模式)
	IWDT	Yes
沟通	SAU	2 个 (简化 SPI) *1、2 个 (简化 IIC) *1、1 个 (UART) *1、1 个 (支持 LIN 总线的 UART) *1
	IICA	1
模拟	ADC10	6
	TSN	Yes
数据处理	CRC	Yes
I/O端口	I/O引脚	16
	输入引脚	1
	上拉电阻	12
	N沟道开漏输出	11
	5V容差	—

注1: SAU由多个通道组成。每个通道一次只能分配一个功能。

1.5 Pin Functions

Table 1.12 Pin functions

Function	Signal	I/O	Description
Power supply	VCC	Input	Power supply pin. Connect it to the system power supply. Connect this pin to VSS by a 0.1-μF capacitor. Place the capacitor close to the pin.
	VCL	I/O	Connect this pin to the VSS pin by the smoothing capacitor used to stabilize the internal power supply. Place the capacitor close to the pin.
	VSS	Input	Ground pin. Connect it to the system power supply (0 V).
Clock	PCLBUZ0	Output	Clock output / Buzzer output
System control	RES	Input	Reset signal input pin. The MCU enters the reset state when this signal goes low.
On-chip debug	SWDIO	I/O	Serial wire debug data input/output pin
	SWCLK	Input	Serial wire clock pin
Interrupt	NMI	Input	Non-maskable interrupt request pin
	IRQ0 to IRQ5	Input	Maskable interrupt request pins
TAU	TI00 to TI07	Input	Pins for inputting an external counting clock/capture trigger to 16-bit timers 00 to 07
	TO00 to TO07	I/O	Timer output pins for 16-bit timers 00 to 07
IICA	SCLAn (n = 0)	I/O	Input/output pins for the clock
	SDAAn (n = 0)	I/O	Input/output pins for data
SAU	SCK00, SCK11	I/O	Serial clock I/O pins for serial interfaces SPI00, SPI11
	SI00, SI11	Input	Serial data input pins for serial interfaces SPI00, SPI11
	SO00, SO11	Output	Serial data output pins for serial interfaces SPI00, SPI11
	SSI00	Input	Chip select pin for serial interfaces SPI00
	SCL00, SCL11	Output	Serial clock output pins for serial interfaces IIC00, IIC11
	SDA00, SDA11	I/O	Serial data I/O pins for serial interfaces IIC00, IIC11
	RXD0, RXD1	Input	Serial data input pins for serial interfaces UART0, UART1
	TXD0, TXD1	Output	Serial data output pins for serial interfaces UART0, UART1
Analog power supply	VREFH0	Input	Analog reference voltage supply pin for the ADC10. Connect this pin to external reference voltage or VCC.
	VREFL0	Input	Analog reference ground pin for the ADC10. Connect this pin to external reference voltage or VSS.
ADC10	AN000, AN001, AN004, AN005, AN021 to AN022	Input	Input pins for the analog signals to be processed by the A/D converter.
I/O ports	P010 to P013	I/O	General-purpose input/output pins
	P100 to P102, P108 to P110, P112	I/O	General-purpose input/output pins
	P200	Input	General-purpose input pin
	P201, P206, P212, P213	I/O	General-purpose input/output pins
	P300	I/O	General-purpose input/output pins

1.5针脚功能

表 1.12 引脚功能

功能	信号	I/O	描述
电源	VCC	输入	电源引脚。将其连接到系统电源。使用一个0.1-μF 电容将此引脚连接到VSS。将电容靠近引脚放置。
	VCL	I/O	将此引脚通过用于稳定内部电源的滤波电容连接到 VSS 引脚。将电容靠近引脚放置。
	VSS	输入	接地引脚。将其连接到系统电源(0 V)。
钟	PCLBUZ0	输出	时钟输出/蜂鸣器输出
系统控制	RES	输入	复位信号输入引脚。当此信号为低电平时，MCU 进入复位状态。
片上调试	SWDIO	I/O	串行线调试数据输入/输出引脚
	SWCLK	输入	串行线时钟引脚
打断	NMI	输入	不可屏蔽中断请求引脚
	IRQ0 至 IRQ5	输入	可屏蔽中断请求引脚
TAU	TI00 至 TI07	输入	用于向 16 位定时器 00 至 07 输入外部计数时钟/捕获触发信号的引脚
	TO00 至 TO07	I/O	16 位定时器的定时器输出引脚 00 至 07
IICA	SCLAn (n = 0)	I/O	时钟的输入/输出引脚
	SDAAn (n = 0)	I/O	数据输入/输出引脚
SAU	SCK00, SCK11	I/O	串行接口 SPI00、SPI11 的串行时钟 I/O 引脚
	SI00, SI11	输入	串行接口 SPI00、SPI11 的串行数据输入引脚
	SO00, SO11	输出	串行接口 SPI00、SPI11 的串行数据输出引脚
	SSI00	输入	串行接口 SPI00 的片选引脚
	SCL00、SCL11	输出	串行接口 IIC00、IIC11 的串行时钟输出引脚
	SDA00, SDA11	I/O	串行接口 IIC00、IIC11 的串行数据 I/O 引脚
	RXD0, RXD1	输入	串行接口 UART0、UART1 的串行数据输入引脚
	TXD0, TXD1	输出	串行接口 UART0、UART1 的串行数据输出引脚
模拟电源	VREFH0	输入	ADC10 的模拟参考电压供电引脚。将此引脚连接到外部参考电压或 VCC。
	VREFL0	输入	ADC10 的模拟参考接地引脚。将此引脚连接到外部参考电压或 VSS。
ADC10	AN000、AN001、AN004、AN005、AN021 至 AN022	输入	用于输入模拟信号的引脚，这些信号将由 A/D 转换器处理。
I/O端口	P010 至 P013	I/O	通用输入/输出引脚
	P100 至 P102, P108 至 P110、P112	I/O	通用输入/输出引脚
	P200	输入	通用输入引脚
	P201, P206, P212, P213	I/O	通用输入/输出引脚
	P300	I/O	通用输入/输出引脚

1.6 Pin Assignments

Figure 1.3 shows the pin assignment from the top view.

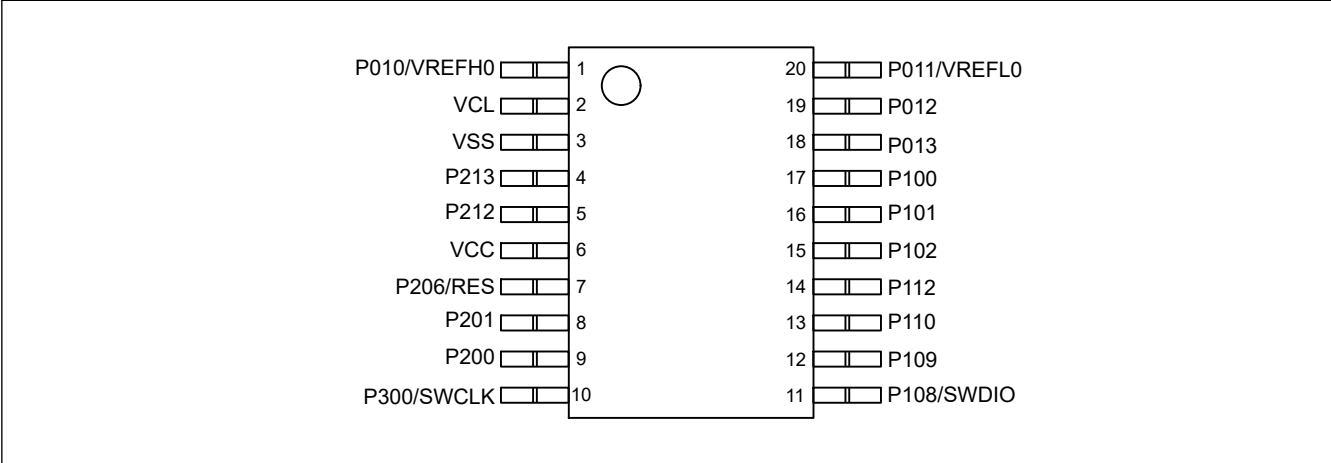


Figure 1.3 Pin assignment for TSSOP 20-pin (top view)

1.6引脚分配

图 1.3 显示了从俯视图看到的引脚分配情况。

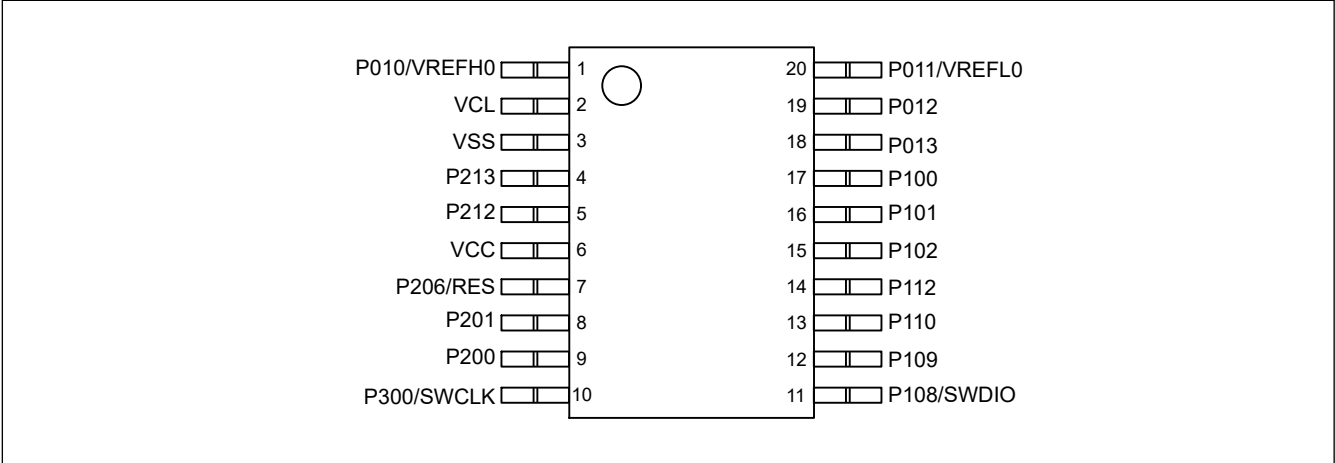


图 1.3 TSSOP 20 引脚封装的引脚分配（俯视图）

1.7 Pin Lists

Table 1.13 Pin list

Pin number				Timers	Communication interfaces		Analogs
				TAU	SAU	IICA	ADC
1	VREFH0	P010	—	—	—	—	AN000
2	VCL	—	—	—	—	—	—
3	VSS	—	—	—	—	—	—
4	—	P213	IRQ0_B	TI00_A/TI02_B/TO02_B	TXD1_A/SO11_A	SDAA0_B	—
5	—	P212	IRQ1_B	TO00_A/TI03_C/TO03_C	RXD1_A/SDA11_A/SI11_A	SCLA0_B	—
6	VCC	—	—	—	—	—	—
7	RES	P206	IRQ1_C	—	—	—	—
8	PCLBUZ0_A	P201	IRQ5_B	TI05_B/TO05_B	SCL11_B/SSI00_B/SCK11_B	—	—
9	—	P200	IRQ0_A/NMI	TI06_B	RXD1_B/SSI00_D	—	—
10	SWCLK	P300	—	TI04_B/TO04_B	TXD1_B	SDAA0_E	—
11	SWDIO/PCLBUZ0_D	P108	—	TI03_B/TO03_B	SCK00_C	SCLA0_E	—
12	—	P109	IRQ4_B	TI02_A/TO02_A	TXD0_B/SO00_B	SDAA0_C	—
13	—	P110	IRQ3_B	TI01_A/TO01_A	RXD0_B/SDA00_B/SI00_B	SCLA0_C	—
14	—	P112	IRQ2_B	TI03_A/TO03_A	SCL00_B/SCK00_B/SSI00_C	—	—
15	PCLBUZ0_B	P102	IRQ4_A	TI06_A/TO06_A/TO00_C	SCL00_A/SCK00_A	—	—
16	—	P101	IRQ3_A	TI07_A/TO07_A/TI00_C	TXD0_A/SO00_A	SDAA0_D	AN021
17	—	P100	IRQ2_A	TI04_A/TO04_A/TI01_B/TO01_B	RXD0_A/SDA00_A/SI00_A	SCLA0_D	AN022
18	—	P013	—	—	—	—	AN005
19	—	P012	—	—	—	—	AN004
20	VREFL0	P011	—	—	—	—	AN001

Note: Some signal names have _A, _B, _C, _D, or _E suffixes, but these suffixes can be ignored when assigning functionality, except for SAU and IICA. For SAU and IICA, only signals, except for SCL11, SCK11, SCK00, and SSI00, bearing the same suffix can be selected. Assigning the same function to two or more pins simultaneously is prohibited.

1.7引脚列表

表 1.13 引脚列表

Pin number				定时器	通信接口		类似物
				TAU	SAU	IICA	ADC
1	VREFH0	P010	—	—	—	—	AN000
2	VCL	—	—	—	—	—	—
3	VSS	—	—	—	—	—	—
4	—	P213	IRQ0_B	TI00_A/TI02_B/TO02_B	TXD1_A/SO11_A	SDAA0_B	—
5	—	P212	IRQ1_B	TO00_A/TI03_C/TO03_C	RXD1_A/SDA11_A/SI11_A	SCLA0_B	—
6	VCC	—	—	—	—	—	—
7	RES	P206	IRQ1_C	—	—	—	—
8	PCLBUZ0_A	P201	IRQ5_B	TI05_B/TO05_B	SCL11_B/SSI00_B/SCK11_B	—	—
9	—	P200	IRQ0_A/NMI	TI06_B	RXD1_B/SSI00_D	—	—
10	SWCLK	P300	—	TI04_B/TO04_B	TXD1_B	SDAA0_E	—
11	SWDIO/PCLBUZ0_D	P108	—	TI03_B/TO03_B	SCK00_C	SCLA0_E	—
12	—	P109	IRQ4_B	TI02_A/TO02_A	TXD0_B/SO00_B	SDAA0_C	—
13	—	P110	IRQ3_B	TI01_A/TO01_A	RXD0_B/SDA00_B/SI00_B	SCLA0_C	—
14	—	P112	IRQ2_B	TI03_A/TO03_A	SCL00_B/SCK00_B/SSI00_C	—	—
15	PCLBUZ0_B	P102	IRQ4_A	TI06_A/TO06_A/TO00_C	SCL00_A/SCK00_A	—	—
16	—	P101	IRQ3_A	TI07_A/TO07_A/TI00_C	TXD0_A/SO00_A	SDAA0_D	AN021
17	—	P100	IRQ2_A	TI04_A/TO04_A/TI01_B/TO01_B	RXD0_A/SDA00_A/SI00_A	SCLA0_D	AN022
18	—	P013	—	—	—	—	AN005
19	—	P012	—	—	—	—	AN004
20	VREFL0	P011	—	—	—	—	AN001

注意：部分信号名称带有 _A、_B、_C、_D 或 _E 后缀，但在分配功能时可以忽略这些后缀，SAU 和 IICA 除外。对于 SAU 和 IICA，只能选择具有相同后缀的信号（SCL11、SCK11、SCK00 和 SSI00 除外）。禁止同时为两个或多个引脚分配相同的功能。

2. CPU

The MCU is based on the Arm® Cortex®-M23 core.

2.1 Overview

2.1.1 CPU

- Arm Cortex-M23
 - Revision: r1p0-00rel0
 - Armv8-M architecture profile
 - Main Extension is not implemented
 - Single-cycle integer multiplier
 - 19-cycle integer divider
- SysTick timer
 - Driven by SYSTICCLK (LOCO) or ICLK

See reference 1. and reference 2. in [section 2.8. References](#) for details.

2.1.2 Debug

- Arm® CoreSight™ MTB-M23
 - Revision: r0p0-00rel0
 - Buffer size: 1 KB of 2-KB MTB SRAM
- Data Watchpoint Unit (DWT)
 - 2 comparators for watchpoints
- Flash Patch and Break point Unit (FPB)
 - 4 instruction comparators
- CoreSight Debug Access Port (DAP)
 - Serial Wire-Debug Port (SW-DP)
- Debug Register Module (DBGREG)
 - Reset control
 - Halt control

See reference 1. and reference 2. in [section 2.8. References](#) for details.

2.1.3 Operating Frequency

The operating frequencies for the MCU are as follows:

- CPU: maximum 32 MHz
- Serial Wire Debug (SWD) interface: maximum 12.5 MHz

2.1.4 Block Diagram

[Figure 2.1](#) shows a block diagram of the Cortex-M23 core.

2. CPU

该MCU基于Arm®Cortex®-M23内核。

2.1 概述

2.1.1 CPU

- 手臂 Cortex-M23
 - 版本: r1p0-00rel0
 - Armv8-M架构简介
 - 主扩展尚未实现
 - 单周期整数乘法器
 - 19周期整数除法器
- 系统滴答定时器
 - 由 SYSTICCLK (LOCO) 或 ICLK 驱动

有关详细信息, 请参阅第 2.8 节“参考文献”中的参考文献 1 和参考文献 2。

2.1.2 调试

- Arm® CoreSight™ MTB-M23
 - 版本: r0p0-00rel0
 - 缓冲区大小: 1 KB (2 KB MTB SRAM)
- 数据观察点单元 (DWT)
 - 2 个监测点比较器
- 闪存补丁和断点单元 (FPB)
 - 4 个指令比较器
- CoreSight调试访问端口 (DAP)
 - 串行线调试端口 (SW-DP)
- 调试寄存器模块 (DBGREG)
 - 重置控制
 - 停止控制

有关详细信息, 请参阅第 2.8 节“参考文献”中的参考文献 1 和参考文献 2。

2.1.3 工作频率

MCU 的工作频率如下:

- CPU: 最大值32 MHz
- 串行线调试 (SWD) 接口: 最大值12.5 MHz

2.1.4框图

图 2.1 显示了 Cortex-M23 核心的框图。

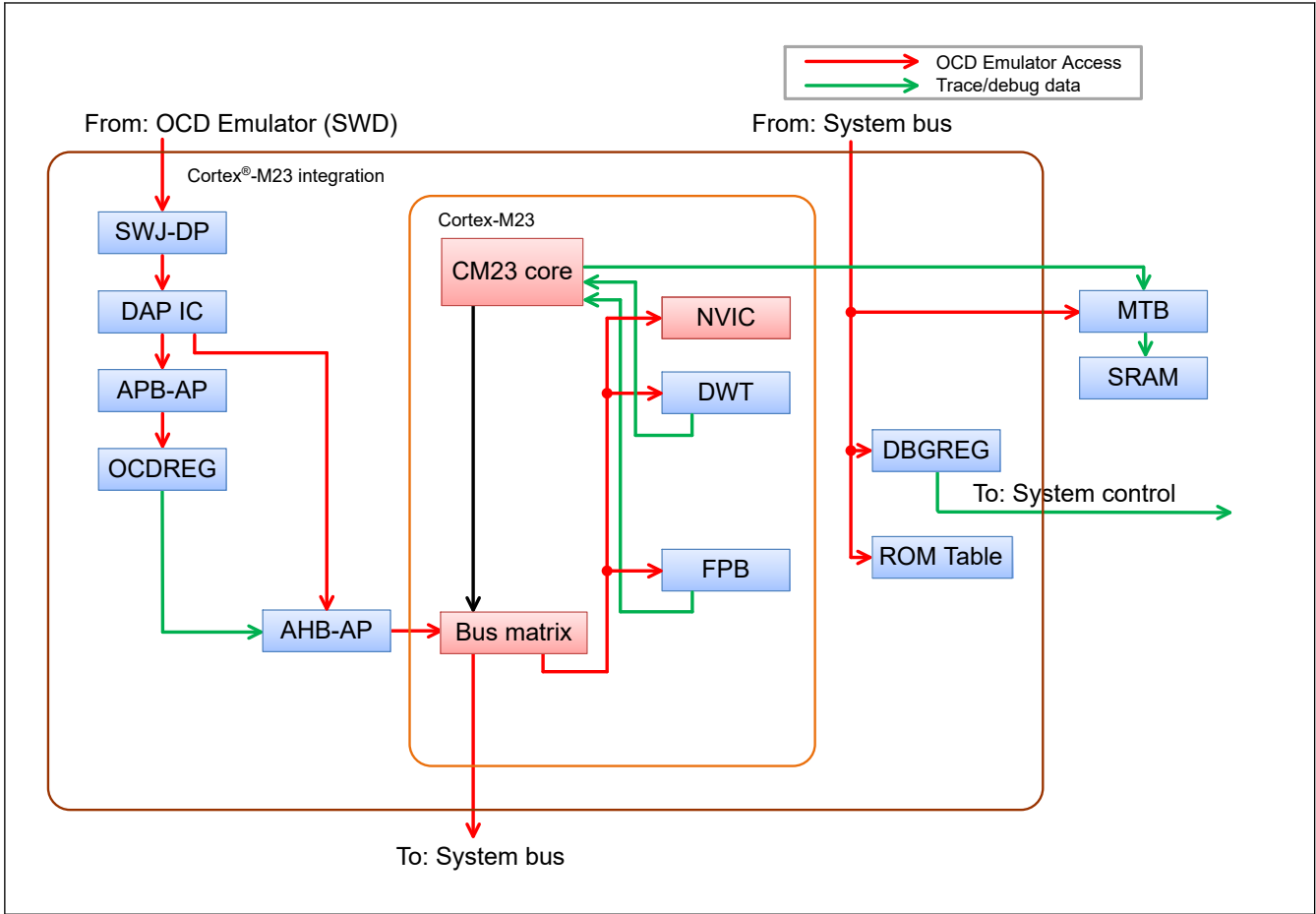


Figure 2.1 Cortex-M23 block diagram

2.2 Implementation Options

Table 2.1 shows the implementation options of the MCU.

Table 2.1 Implementation options (1 of 2)

Option	Implementation
Non-secure MPU	Not included
Secure MPU	Not included
Security extension	Not included
Single-cycle multiplier	Included
Divider	Included, 19 cycles
Number of interrupts	64*1
Number of Wakeup Interrupt Controllers (WIC)	Not included
Cross Trigger Interface (CTI)	Not included
Micro Trace Buffer (MTB)	Included
Embedded Trace Macrocell (ETM)	Not included
Multi-drop support for serial wire	Not supported
Sleep mode power saving	Sleep mode and other low power modes are supported. For more details, see section 9, Low Power Modes . Note: SCB.SCR.SLEEPDEEP is ignored.
Endianness	Little-endian
SysTick timer	Included

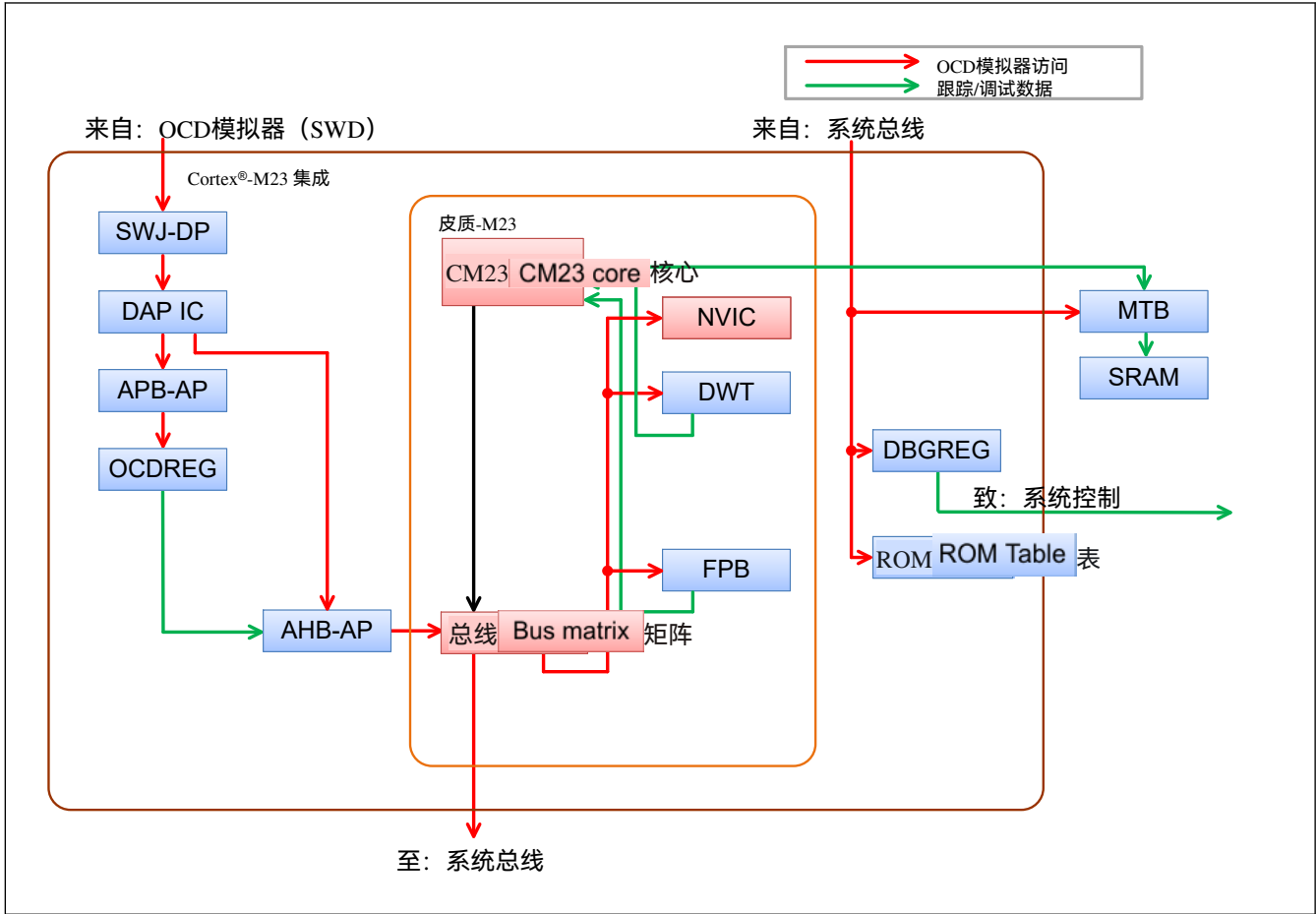


图 2.1 Cortex-M23 框图

2.2 实施方案

表 2.1 显示了 MCU 的实现选项。

表 2.1 实施方案 (1/2)

选项	执行
非安全 MPU	不包含
安全 MPU	不包含
安全扩展	不包含
单周期乘法器	包括
分隔线	包括 19 个循环
中断次数	64*1
唤醒中断控制器数量 (WIC)	不包含
交叉触发接口 (CTI)	不包含
微量缓冲液 (MTB)	包括
嵌入式跟踪宏单元 (ETM)	不包含
支持串行线多点连接	不支持
睡眠模式省电	支持睡眠模式和其他低功耗模式。更多详情，请参阅第 9 节“低功耗模式”。 注意：SCB.SCR.SLEEPDEEP 将被忽略。
字节序	小端字节序
系统滴答定时器	包括

Table 2.1 Implementation options (2 of 2)

Option	Implementation
SYST_CALIB register (0x4000_0147)	Bit [31] = 0 Bit [30] = 1 Bits [29:24] = 0x00 Bits [23:0] = 0x000147 Reference clock provided TENMS value is inexact Reserved TENMS: (32768 × 10 ms) - 1/32.768 kHz = 326.66 decimal = 327 with skew = 0x000147
Event input/output	Not implemented
System reset request output	The SYSRESETREQ bit in Application Interrupt and Reset Control Register causes a CPU reset
Auxiliary fault inputs (AUXFAULT)	Not implemented

Note 1. Some of them are reserved. For the detail of the interrupt sources, [section 11.3.1. Interrupt Vector Table](#)

2.3 SWD Interface

[Table 2.2](#) shows the SWD pins.

Table 2.2 SWD pins

Name	I/O	Function	What to do when not in use
SWCLK*1	Input	Serial wire clock pin	Pull-up
SWDIO*1	I/O	Serial wire data I/O pin	Pull-up

Note 1. After the reset, the built-in pull-up is enabled for that pin.

2.4 Debug Function

2.4.1 Debug Mode Definition

[Table 2.3](#) shows the CPU debug modes and usage conditions.

Table 2.3 CPU debug mode and conditions

Conditions		Mode	
OCD connect*1	SWD authentication	Debug mode	Debug authentication*2
Not connected	—	User mode	Disabled
Connected	Failed	User mode	Disabled
Connected	Passed	OCD mode	Enabled

Note: When CSYSPWRUPREQ bit is set, OFS1.PORTSELB value is masked and fixed to the RES input.

Note: Do not use on-chip debugging with products designated as part of mass production, because using this function may cause the guaranteed number of times the flash memory is rewritten to be exceeded, and product reliability therefore cannot be guaranteed.

Note 1. OCD connect is determined by the CDBGPWRUPREQ bit and CSYSPWRUPREQ bit output in the SWJ-DP register. The bit can only be written by the OCD.
However, the level of the bit can be confirmed by reading the DBGSTR.CDBGPWRUPREQ bit.

Note 2. Debug authentication is defined by the Armv8-M Architecture. Enabled means that both invasive and non-invasive CPU debugging are permitted. Disabled means that both are not permitted.

2.4.2 Debug Mode Effects

This section describes the effects of debug mode, which occur both internally and externally to the CPU.

2.4.2.1 Low power mode

All CoreSight debug components can retain the register settings even when the CPU enters Software Standby, or Snooze mode. However, AHB-AP cannot respond to On-Chip Debug (OCD) emulator access in these low power modes. The OCD emulator must wait for cancellation of the low power mode to access the CoreSight debug components. To request low power mode cancellation, the OCD emulator can set the DBIRQ bit in the MCUCTRL register. For details, see [section 2.5.6.3. MCUCTRL : MCU Control Register](#).

表 2.1 实施方案 (2/2)

选项	执行
SYST_CALIB 寄存器 (0x4000_0147)	位[31] = 0 位[30] = 1 位[29:24] = 0x00 位[23:0] = 0x000147 参考时钟提供的 TENMS 值 不准确, 已保留 TENMS: (32768× 10 毫秒) 1/32.768 kH z= 326.66十进制= 327带偏斜 = 0x000147
事件输入/输出	未实施
系统重置请求输出	应用程序中断和复位控制寄存器中的 SYSRESETREQ 位会导致 CPU 复位
辅助故障输入 (AUXFAULT)	未实施

注1: 其中一些是保留的。有关中断源的详细信息, 请参阅11.3.1节“中断向量表”。

2.3SWD接口

表 2.2 显示了 SWD 引脚。

表 2.2SWD 引脚

Name	I/O	功能	不使用时该如何处理
SWCLK*1	输入	串行线时钟引脚	引体向上
SWDIO*1	I/O	串行线数据 I/O 引脚	引体向上

注 1. 复位后, 该引脚的内置上拉电阻将被启用。

2.4调试功能

2.4.1调试模式定义

表 2.3 显示了 CPU 调试模式和使用条件。

表 2.3 CPU 调试模式和条件

状况		模式	
OCD 连接*1	SWD认证	调试模式	调试身份验证*2
未连接	—	用户模式	已禁用
已连接	失败的	用户模式	已禁用
已连接	通过	强迫症模式	已启用

注意: 当 CSYSPWRUPREQ 位设置时, OFS1.PORTSELB 值被屏蔽并固定为 RES 输入。

注意: 请勿对指定为批量生产产品的一部分的产品使用片上调试功能, 因为使用此功能可能会导致闪存重写次数超过保证次数, 因此无法保证产品的可靠性。

注1: OCD连接状态由SWJ-DP寄存器中的CDBGPWRUPREQ位和CSYSPWRUPREQ位输出决定。该位只能由OCD写入。

但是, 可以通过读取 DBGSTR.CDBGPWRUPREQ 位来确认该位的级别。

注2: 调试认证由Armv8-M架构定义。启用表示允许侵入式和非侵入式CPU调试。禁用表示两种调试方式均不允许。

2.4.2调试模式效果

本节描述调试模式的影响, 这些影响既发生在 CPU 内部, 也发生在 CPU 外部。

2.4.2.1 低功耗模式

即使 CPU 进入软件待机或休眠模式, 所有 CoreSight 调试组件也能保留寄存器设置。但是, 在这些低功耗模式下, AHB-AP 无法响应片上调试 (OCD) 仿真器的访问。OCD 仿真器必须等待低功耗模式取消才能访问 CoreSight 调试组件。要请求取消低功耗模式, OCD 仿真器可以设置 MCUCTRL 寄存器中的 DBIRQ 位。详情请参见 2.5.6.3 节。MCUCTRL: MCU 控制寄存器。

2.4.2.2 Reset

In OCD mode, some resets depend on the CPU status and the DBGSTOPCR register setting.

Table 2.4 Reset or interrupt and mode setting

Reset or interrupt name	Control in On-Chip Debug (OCD) mode	
	OCD break mode	OCD run mode
RES pin reset	Same as user mode	
Power-on reset	Same as user mode	
Independent watchdog timer reset/interrupt	Does not occur*1	Depends on DBGSTOPCR setting
Voltage monitor reset/interrupt	Depends on DBGSTOPCR setting	
Software reset	Same as user mode	

Note: In OCD break mode, the CPU is halted. In OCD run mode, the CPU is in OCD mode and the CPU is not halted.
Note 1. The IWDT always stop in this mode.

2.4.3 Trace Control (for the MTB)

The Micro Trace Buffer (MTB) has programmable registers to control the behavior of the trace features and the POSITION, MASTER, FLOW, and BASE registers. Table 2.5 shows the registers in offset order from the base address.

Table 2.5 Address of MTB registers

Address	Register	Value on reset
MTB_BASE + 0x000	MTB_POSITION	Bits [31:0] = UNKNOWN
MTB_BASE + 0x004	MTB_MASTER	Bits [31] = 0, Bits [30:10] = UNKNOWN, Bits [9:8] = 0, Bits [7] = 1, Bits [6:5] = 0, Bits [4:0] = UNKNOWN
MTB_BASE + 0x008	MTB_FLOW	Bits [31:2] = UNKNOWN, Bits [1:0] = 0
MTB_BASE + 0x00C	MTB_BASE	Bits [31:0] = 0x2000_4000

Note: MTB_BASE = 0x4001_9000

For more information on these registers, see the *ARM® CoreSight™ MTB-M23 Technical Reference Manual (ARM DDI 0564C)*.

Note: Do not attempt to access reserved or unused address locations.

The MTB for trace is limited from 0x2000_4000 to 0x2000_47FF.

2.4.4 CoreSight™ (for MTB)

See the *ARM® CoreSight™ Architecture Specification* for more information about the registers and access types. Table 2.6 shows the registers in offset order from the base address.

Table 2.6 Address of CoreSight

Address	Register
MTB_BASE + 0xFF0 to 0xFFC	Component ID
MTB_BASE + 0xFE0 to 0xFDC	Peripheral ID
MTB_BASE + 0xFCC	Device Type Identifier
MTB_BASE + 0xFC8	Device Configuration
MTB_BASE + 0xFBC	Device Architecture
MTB_BASE + 0xFB8	Authentication Status
MTB_BASE + 0xFB4	Lock Status
MTB_BASE + 0xFB0	Lock Access

Note: MTB_BASE = 0x4001_9000

2.4.2.2 重置

在 OCD 模式下，某些复位取决于 CPU 状态和 DBGSTOPCR 寄存器设置。

表 2.4 复位或中断和模式设置

复位或中断名称	片上调试 (OCD) 模式下的控制	
	强迫症中断模式	强迫症运行模式
RES引脚复位	与用户模式相同	
上电重置	与用户模式相同	
独立看门狗定时器复位/中断	不会发生*1	取决于 DBGSTOPCR 设置
电压监控器复位/中断	取决于 DBGSTOPCR 设置	
软件重置	与用户模式相同	

注意：在OCD中断模式下，CPU会停止运行。在OCD运行模式下，CPU处于OCD模式，但不会停止运行。
注1. IWDT 在此模式下始终停止。

2.4.3轨迹控制（适用于山地自行车）

微型跟踪缓冲器 (MTB) 具有可编程寄存器，用于控制跟踪功能的行为以及 POSITION、MASTER、FLOW 和 BASE 寄存器。表 2.5 按相对于基地址的偏移顺序显示了这些寄存器。

表 2.5 MTB 寄存器的地址

地址	登记	重置值
MTB_BASE + 0x000	MTB_POSITION	位[31:0] = 未知
MTB_BASE + 0x004	MTB_MASTER	位 [31] = 0, 位[30:10] = 未知, 位[9:8] = 0, 位 [7] = 1 , 位[6:5] = 0, 位[4:0] =未知
MTB_BASE + 0x008	MTB_FLOW	位[31:2] = 未知, 位[1:0] = 0
MTB_BASE + 0x00C	MTB_BASE	位[31:0] = 0x2000_4000

注意: MTB_BASE = 0x4001_9000

有关这些寄存器的更多信息，请参阅*ARM®CoreSight™ MTB-M23 技术参考手册 (ARM DDI 0564C)*。

注意：请勿尝试访问已保留或未使用的地址位置。

跟踪的 MTB 范围限定在0x2000_4000到0x2000_47FF之间。

2.4.4CoreSight™（适用于山地自行车）

有关寄存器和访问类型的更多信息，请参阅 CoreSight™架构规范（*ARM®*）。表 2.6 按偏移量顺序显示了寄存器相对于基地址的顺序。

表 2.6 CoreSight 的地址

地址	登记
MTB_BASE + 0xFF0 to 0xFFC	组件 ID
MTB_BASE + 0xFE0 to 0xFDC	外围设备 ID
MTB_BASE + 0xFCC	设备类型标识符
MTB_BASE + 0xFC8	设备配置
MTB_BASE + 0xFBC	设备架构
MTB_BASE + 0xFB8	身份验证状态
MTB_BASE + 0xFB4	锁定状态
MTB_BASE + 0xFB0	锁门

注意: MTB_BASE= 0x4001_9000

For more information on these registers, see the *ARM® CoreSight™ MTB-M23 Technical Reference Manual (ARM DDI 0564C)*.

Note: Do not attempt to access reserved or unused address locations.

2.5 Programmers Model

2.5.1 Address Spaces

The MCU debug system includes two CoreSight Access Ports (AP):

- AHB-AP, which is connected to the CPU bus matrix and has the same access to the system address space as the CPU
- APB-AP, which has a dedicated address space (OCD address space) and is connected to the OCDREG registers.

Figure 2.2 shows a block diagram of the AP connection and address spaces.

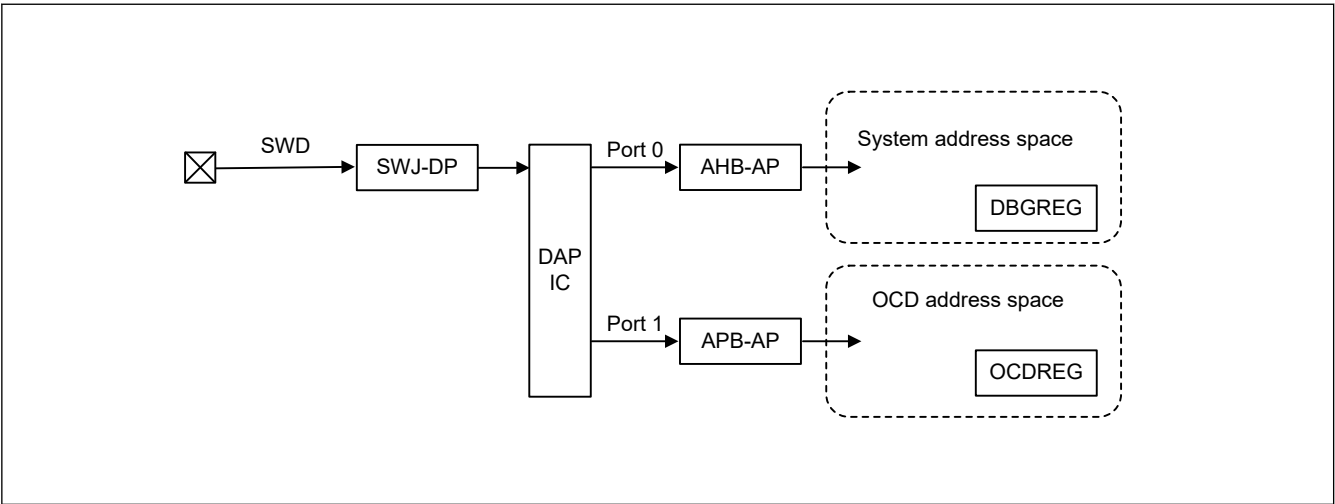


Figure 2.2 SWD authentication block diagram

For debugging purposes, there are two register modules, DBGREG and OCDREG. DBGREG is located in the system address space and can be accessed from the OCD emulator, the CPU, and other bus masters in the MCU. OCDREG is located in the OCD address space and can only be accessed from the OCD tool. The CPU and other bus masters cannot access OCDREG.

2.5.2 Cortex-M23 Peripheral Address Map

In the system address space, the Cortex-M23 core has a Private Peripheral Bus (PPB) that can only be accessed from the CPU and OCD emulator. Table 2.7 shows the address map of the MCU.

Table 2.7 Cortex-M23 peripheral address map

Component name	Start address	End address	Note
DWT	0xE000_1000	0xE000_1FFF	See reference 2. in section 2.8. References
FPB	0xE000_2000	0xE000_2FFF	See reference 2. in section 2.8. References
SCS	0xE000_E000	0xE000_EFFF	See reference 2. in section 2.8. References

2.5.3 External Debug Address Map

In the system address space, the Cortex-M23 core has external debug components. These components can be accessed from the CPU and other bus masters through the system bus. Table 2.8 shows the address map of the Cortex-M23 external debug components.

有关这些寄存器的更多信息，请参阅*ARM®CoreSight™ MTB-M23 技术参考手册 (ARM DDI 0564C)*。

注意：请勿尝试访问已保留或未使用的地址位置。

2.5 程序员模型

2.5.1地址空间

MCU调试系统包括两个CoreSight访问端口（AP）：

- AHB-AP，它连接到CPU总线矩阵，并与CPU一样拥有对系统地址空间的访问权限。
- APB-AP 具有专用地址空间（OCD 地址空间），并连接到 OCDREG 寄存器。

图 2.2 显示了 AP 连接和地址空间的框图。

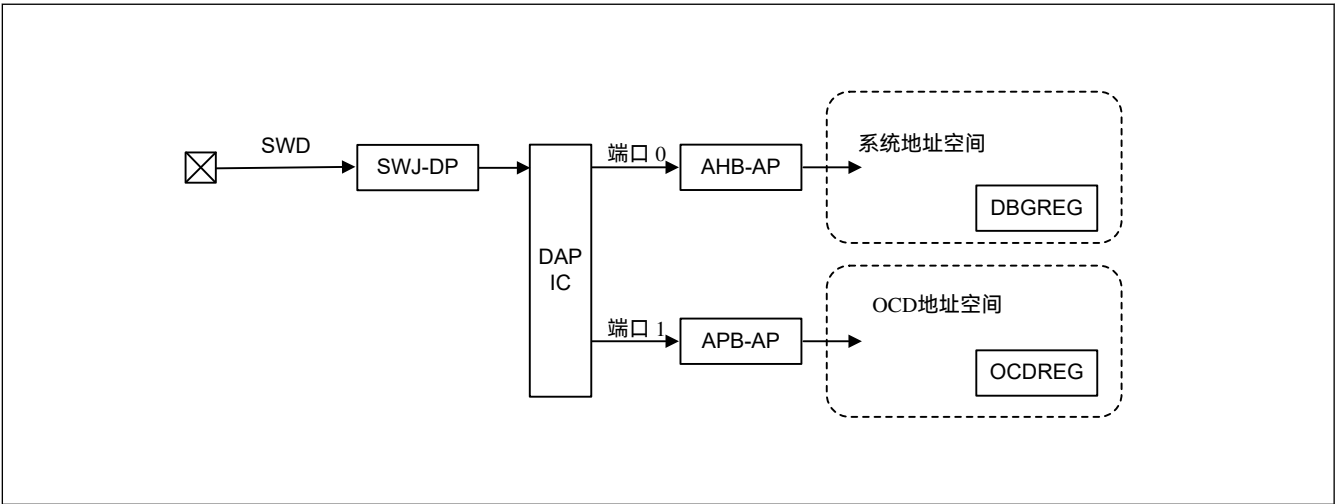


图 2.2SWD 认证框图

为了便于调试，MCU 中有两个寄存器模块：DBGREG 和 OCDREG。DBGREG 位于系统地址空间，可由 OCD 仿真器、CPU 以及 MCU 中的其他总线主控器访问。OCDREG 位于 OCD 地址空间，只能由 OCD 工具访问。CPU 和其他总线主控器无法访问 OCDREG。

2.5.2 Cortex-M23 外围地址映射

在系统地址空间中，Cortex-M23 内核具有一条专用外设总线 (PPB)，该总线只能从 CPU 和 OCD 仿真器访问。表 2.7 显示了 MCU 的地址映射。

表 2.7 Cortex-M23 外设地址映射

组件名称	起始地址	结束地址	Note
DWT	0xE000_1000	0xE000_1FFF	参见第 2.8 节中的参考文献 2。参考文献
FPB	0xE000_2000	0xE000_2FFF	参见第 2.8 节中的参考文献 2。参考文献
SCS	0xE000_E000	0xE000_EFFF	参见第 2.8 节中的参考文献 2。参考文献

2.5.3 外部调试地址映射

在系统地址空间中，Cortex-M23 内核包含外部调试组件。这些组件可以通过系统总线从 CPU 和其他总线主控器访问。表 2.8 显示了 Cortex-M23 外部调试组件的地址映射。

Table 2.8 External debug address map

Component name	Start address	End address	Note
MTB (SRAM area)	0x2000_4000	0x2000_47FF	MTB uses up to 1 KB as trace buffer See reference 6. in section 2.8. References .
MTB (SFR area)	0x4001_9000	0x4001_9FFF	See reference 6. in section 2.8. References .
ROM Table	0x4001_A000	0x4001_AFFF	See reference 6. in section 2.8. References .

2.5.4 CoreSight ROM Table

The MCU contains one CoreSight ROM Table, which lists all components implemented in the user area.

2.5.4.1 ROM entries

[Table 2.9](#) shows the ROM entries. The OCD emulator can use the ROM entries to determine which components are implemented in a system. See reference 4. in [section 2.8. References](#) for details.

Table 2.9 ROM entries

#	Address	Access size	R/W	Value	Target module pointer
0	0x4001_A000	32 bits	R	0x9FFF4003	SCS
1	0x4001_A004	32 bits	R	0x9FFE7003	DWT
2	0x4001_A008	32 bits	R	0x9FFE8003	FPB
3	0x4001_A00C	32 bits	R	0xFFFFF003	MTB
4	0x4001_A010	32 bits	R	0x00000000	End of entries

2.5.4.2 CoreSight component registers

The CoreSight ROM Table lists the CoreSight component registers defined in the Arm CoreSight architecture.

[Table 2.10](#) shows the registers. See reference 5. in [section 2.8. References](#) for details of each register.

Table 2.10 CoreSight component registers in the CoreSight ROM Table

Name	Address	Access size	R/W	Initial value
MEMTYPE	0x4001_AFCC	32 bits	R	0x00000001
PIDR4	0x4001_AFD0	32 bits	R	0x00000004
PIDR5	0x4001_AFD4	32 bits	R	0x00000000
PIDR6	0x4001_AFD8	32 bits	R	0x00000000
PIDR7	0x4001_AFDC	32 bits	R	0x00000000
PIDR0	0x4001_AFE0	32 bits	R	0x0000007C
PIDR1	0x4001_AFE4	32 bits	R	0x00000030
PIDR2	0x4001_AFE8	32 bits	R	0x0000000A
PIDR3	0x4001_AFEC	32 bits	R	0x00000000
CIDR0	0x4001_AFF0	32 bits	R	0x0000000D
CIDR1	0x4001_AFF4	32 bits	R	0x00000010
CIDR2	0x4001_AFF8	32 bits	R	0x00000005
CIDR3	0x4001_AFFC	32 bits	R	0x000000B1

2.5.5 DBGREG Module

The DBGREG module controls the debug functionalities and is implemented as a CoreSight-compliant component.

[Table 2.11](#) shows the DBGREG registers other than the CoreSight component registers.

表 2.8 外部调试地址映射

组件名称	起始地址	结束地址	Note
山地车（SRAM 区域）	0x2000_4000	0x2000_47FF	MTB 使用最多 1 KB 作为跟踪缓冲区。请参阅第 2.8 节中的参考文献 6。参考文献。
山地自行车（SFR 区域）	0x4001_9000	0x4001_9FFF	参见第 2.8 节“参考文献”中的参考文献 6。
ROM表	0x4001_A000	0x4001_AFFF	参见第 2.8 节“参考文献”中的参考文献 6。

2.5.4 CoreSight ROM 表

MCU 包含一个 CoreSight ROM 表，其中列出了用户区域中实现的所有组件。

2.5.4.1 ROM条目

表 2.9 显示了 ROM 条目。OCD 模拟器可以使用 ROM 条目来确定系统中实现了哪些组件。详情请参见 2.8 节“参考文献”中的参考文献 4。

表 2.9 ROM 条目

#	地址	访问规模	R/W	价值	目标模块指针
0	0x4001_A000	32 位	R	0x9FFF4003	SCS
1	0x4001_A004	32 位	R	0x9FFE7003	DWT
2	0x4001_A008	32 位	R	0x9FFE8003	FPB
3	0x4001_A00C	32 位	R	0xFFFFF003	MTB
4	0x4001_A010	32 位	R	0x00000000	条目结束

2.5.4.2 CoreSight 组件寄存器

CoreSight ROM 表列出了 Arm CoreSight 架构中定义的 CoreSight 组件寄存器。

表 2.10 列出了登记册。有关每个登记册的详细信息，请参见第 2.8 节中的参考文献 5。

表 2.10 CoreSight ROM 表中的 CoreSight 组件寄存器

Name	地址	访问规模	R/W	初始值
MEMTYPE	0x4001_AFCC	32 位	R	0x00000001
PIDR4	0x4001_AFD0	32 位	R	0x00000004
PIDR5	0x4001_AFD4	32 位	R	0x00000000
PIDR6	0x4001_AFD8	32 位	R	0x00000000
PIDR7	0x4001_AFDC	32 位	R	0x00000000
PIDR0	0x4001_AFE0	32 位	R	0x0000007C
PIDR1	0x4001_AFE4	32 位	R	0x00000030
PIDR2	0x4001_AFE8	32 位	R	0x0000000A
PIDR3	0x4001_AFEC	32 位	R	0x00000000
CIDR0	0x4001_AFF0	32 位	R	0x0000000D
CIDR1	0x4001_AFF4	32 位	R	0x00000010
CIDR2	0x4001_AFF8	32 位	R	0x00000005
CIDR3	0x4001_AFFC	32 位	R	0x000000B1

2.5.5DBGREG 模块

DBGREG 模块控制调试功能，并以 CoreSight 兼容组件的形式实现。

表 2.11 显示了除 CoreSight 组件寄存器之外的 DBGREG 寄存器。

Table 2.11 Non-CoreSight DBGREG registers

Name		DAP port	Address	Access size	R/W
Debug Status Register		Port 0	0x4001_B000	32 bits	R
Debug Stop Control Register		Port 0	0x4001_B010	32 bits	R/W

2.5.5.1 DBGSTR : Debug Status Register

Base address: DBG = 0x4001_B000															
Offset address: 0x0000															
Bit position: 31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16															
Bit field:		—	—	CDBG PWRU PACK	CDBG PWRU PREQ	—	—	—	—	—	—	—	—	—	—
Value after reset:		0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit position: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0															
Bit field:		—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:		0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
27:0	—	These bits are read as 0.	R
28	CDBGPWRUPREQ	Debug Power-up Request 0: OCD emulator is not requesting debug power up 1: OCD emulator is requesting debug power up	R
29	CDBGPWRUPACK	Debug Power-up Acknowledge 0: Debug power-up request is not acknowledged 1: Debug power-up request is acknowledged	R
31:30	—	These bits are read as 0.	R

The DBGSTR register is a status register which indicates the state of the debug power-up request to the MCU from the emulator.

2.5.5.2 DBGSTOPCR : Debug Stop Control Register

Base address: DBG = 0x4001_B000															
Offset address: 0x0010															
Bit position: 31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16															
Bit field:		—	—	—	—	—	—	—	—	—	—	—	—	DBGS TOP_L VD1	DBGS TOP_L VD0
Value after reset:		0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit position: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0															
Bit field:		DBGS TOP_SIR	DBGS TOP_TIM	—	—	—	—	—	—	—	—	—	—	—	DBGS TOP_I WDT
Value after reset:		0	0	0	0	0	0	0	0	0	0	0	0	1	1

Bit	Symbol	Function	R/W
0	DBGSTOP_IWDT	Mask Bit for IWDT Reset/Interrupt in the OCD Run Mode In the OCD break mode, the reset/interrupt is masked and IWDT counter is stopped, regardless of this bit value. 0: Enable IWDT reset/interrupt 1: Mask IWDT reset/interrupt and stop IWDT counter	R/W

表 2.11 非 CoreSight DBGREG 寄存器

Name		DAP端口	地址	访问规模	R/W
调试状态寄存器		端口 0	0x4001_B000	32 位	R
调试停止控制寄存器		端口 0	0x4001_B010	32 位	R/W

2.5.5.1 DBGSTR: 调试状态寄存器

基地址: DBG = 0x4001_B000															
Offset address: 0x0000															
位位置: 31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16															
位域:		—	—	CDBG PWRU PACK	CDBG PWRU PREQ	—	—	—	—	—	—	—	—	—	—
重置后的值:		0	0	0	0	0	0	0	0	0	0	0	0	0	0
位位置: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0															
位域:		—	—	—	—	—	—	—	—	—	—	—	—	—	—
重置后的值:		0	0	0	0	0	0	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
27:0	—	这些位读取为 0。	R
28	CDBGPWRUPREQ	调试启动请求 0: OCD模拟器未请求调试启动 1: OCD模拟器正在请求调试启动	R
29	CDBGPWRUPACK	调试启动确认 0: 调试启动请求未得到确认 1: 调试启动请求已确认	R
31:30	—	这些位读取为 0。	R

DBGSTR 寄存器是一个状态寄存器，它指示从仿真器到 MCU 的调试上电请求的状态。

2.5.5.2DBGSTOPCR: 调试停止控制寄存器

基地址: DBG = 0x4001 _B000															
偏移地址: 0x0010															
位位置: 31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16															
位域:		—	—	—	—	—	—	—	—	—	—	—	—	DBGS TOP_L VD1	DBGS TOP_L VD0
重置后的值:		0	0	0	0	0	0	0	0	0	0	0	0	0	0
位位置: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0															
位域:		DBGS TOP_SIR	DBGS TOP_TIM	—	—	—	—	—	—	—	—	—	—	—	DBGS TOP_I WDT
重置后的值:		0	0	0	0	0	0	0	0	0	0	0	0	1	1

少量	象征	功能	R/W
0	DBGSTOP_IWDT	OCD 运行模式下 IWDT 复位/中断的掩码位在 OCD 中断模式下，无论此位值如何，复位/中断都会被掩码，并且 IWDT 计数器都会停止。 0: 启用 IWDT 复位/中断 1: 屏蔽 IWDT 复位/中断并停止 IWDT 计数器	R/W

Bit	Symbol	Function	R/W
1	—	This bit is read as 1. The write value should be 1.	R/W
13:2	—	These bits are read as 0. The write value should be 0.	R/W
14	DBGSTOP_TIM	Mask Bit for TAU, TML32 Reset/Interrupt in the OCD brake mode. In the OCD brake mode, the reset/interrupt is masked and each operation is stopped. 0: Enable TAU, TML32 reset/interrupt 1: Mask TAU, TML32 reset/interrupt	R/W
15	DBGSTOP_SIR	Mask Bit for SAU, IICA, PORT_IRQ0-5 Reset/Interrupt in the OCD brake mode. In the OCD brake mode, the reset/interrupt is masked and each operation is stopped. 0: Enable SAU, IICA, PORT_IRQ0 to 5 reset/interrupt 1: Mask SAU, IICA, PORT_IRQ0 to 5 reset/interrupt	R/W
16	DBGSTOP_LVD0	Mask Bit for LVD0 Reset 0: Enable LVD0 reset 1: Mask LVD0 reset	R/W
17	DBGSTOP_LVD1	Mask Bit for LVD1 Reset/Interrupt 0: Enable LVD1 reset/interrupt 1: Mask LVD1 reset/interrupt	R/W
31:18	—	These bits are read as 0. The write value should be 0.	R/W

The Debug Stop Control Register (DBGSTOPCR) controls the functional stop in OCD mode. All bits in the register are regarded as 0 when the MCU is not in OCD mode. In OCD break mode, the states of the CPU and peripheral functions may deviate. Also, when released OCD break mode and run the program again, the state of the CPU and peripheral functions may be different and the program may be executed.

2.5.5.3 DBGREG CoreSight component registers

The DBGREG module provides the CoreSight component registers defined in the Arm CoreSight architecture.

Table 2.12 shows the registers. See reference 4. in section 2.8. References for details of each register.

Table 2.12 DBGREG CoreSight component registers

Name	Address	Access size	R/W	Initial value
PIDR4	0x4001_BFD0	32 bits	R	0x00000004
PIDR5	0x4001_BFD4	32 bits	R	0x00000000
PIDR6	0x4001_BFD8	32 bits	R	0x00000000
PIDR7	0x4001_BFDC	32 bits	R	0x00000000
PIDR0	0x4001_BFE0	32 bits	R	0x00000005
PIDR1	0x4001_BFE4	32 bits	R	0x00000030
PIDR2	0x4001_BFE8	32 bits	R	0x0000001A
PIDR3	0x4001_BFEC	32 bits	R	0x00000000
CIDR0	0x4001_BFF0	32 bits	R	0x0000000D
CIDR1	0x4001_BFF4	32 bits	R	0x000000F0
CIDR2	0x4001_BFF8	32 bits	R	0x00000005
CIDR3	0x4001_BFFC	32 bits	R	0x000000B1

2.5.6 OCDREG Module

The OCDREG module are only accessible by the On-Chip Debug (OCD) emulator. OCDREG is implemented as a CoreSight-compliant component.

Table 2.13 lists the OCDREG registers.

少量	象征	功能	R/W
1	—	该位读取值为 1。写入值应为 1。	R/W
13:2	—	这些位读取为 0。写入值应为 0。	R/W
14	DBGSTOP_TIM	TAU、TML32复位/中断的掩码位，用于OCD中断模式。在OCD中断模式下，复位/中断信号被屏蔽，所有操作都会停止。 0: 启用 TAU、TML32 复位/中断； 1: 屏蔽 TAU、TML32 复位/中断	R/W
15	DBGSTOP_SIR	OCD 中断模式下，SAU、IICA 和 PORT_IRQ0-5 复位/中断的掩码位。在 OCD 中断模式下，复位/中断信号被屏蔽，所有操作都会停止。 0: 启用 SAU、IICA 和 PORT_IRQ0 至 5 的复位/中断； 1: 屏蔽 SAU、IICA 和 PORT_IRQ0 至 5 的复位/中断	R/W
16	DBGSTOP_LVD0	LVD0复位掩码位 0: 启用 LVD0 复位 1: 掩码 LVD0 重置	R/W
17	DBGSTOP_LVD1	LVD1复位/中断掩码位 0: 启用 LVD1 复位/中断 1: 屏蔽 LVD1 复位/中断	R/W
31:18	—	这些位读取为 0。写入值应为 0。	R/W

调试停止控制寄存器 (DBGSTOPCR) 控制 OCD 模式下的功能停止。当 MCU 不在 OCD 模式下时，寄存器中的所有位都被视为 0。在 OCD 中断模式下，CPU 和外设功能的状态可能会发生偏差。此外，当解除 OCD 中断模式并再次运行程序时，CPU 和外设功能的状态可能不同，程序可能会被执行。

2.5.5.3DBGREG CoreSight 组件寄存器

DBGREG 模块提供 Arm CoreSight 架构中定义的 CoreSight 组件寄存器。

表 2.12 列出了登记册。有关每个登记册的详细信息，请参见第 2.8 节中的参考文献 4。

表 2.12DBGREG CoreSight 组件寄存器

Name	地址	访问规模	R/W	初始值
PIDR4	0x4001_BFD0	32 位	R	0x00000004
PIDR5	0x4001_BFD4	32 位	R	0x00000000
PIDR6	0x4001_BFD8	32 位	R	0x00000000
PIDR7	0x4001_BFDC	32 位	R	0x00000000
PIDR0	0x4001_BFE0	32 位	R	0x00000005
PIDR1	0x4001_BFE4	32 位	R	0x00000030
PIDR2	0x4001_BFE8	32 位	R	0x0000001A
PIDR3	0x4001_BFEC	32 位	R	0x00000000
CIDR0	0x4001_BFF0	32 位	R	0x0000000D
CIDR1	0x4001_BFF4	32 位	R	0x000000F0
CIDR2	0x4001_BFF8	32 位	R	0x00000005
CIDR3	0x4001_BFFC	32 位	R	0x000000B1

2.5.6OCDREG 模块

OCDREG 模块只能通过On芯片调试 (OCD) 仿真器访问。OCDREG 是作为 CoreSight 兼容组件实现的。

表 2.13 列出了 OCDREG 寄存器。

Table 2.13 OCDREG registers

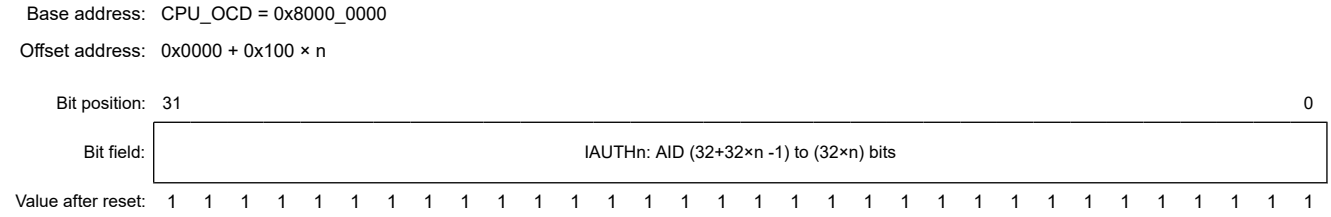
Name		DAP port	Address	Access size	R/W
ID Authentication Code Register 0	IAUTH0	Port 1	0x8000_0000	32 bits	W
ID Authentication Code Register 1	IAUTH1	Port 1	0x8000_0100	32 bits	W
ID Authentication Code Register 2	IAUTH2	Port 1	0x8000_0200	32 bits	W
ID Authentication Code Register 3	IAUTH3	Port 1	0x8000_0300	32 bits	W
MCU Status Register	MCUSTAT	Port 1	0x8000_0400	32 bits	R
MCU Control Register	MCUCTRL	Port 1	0x8000_0410	32 bits	R/W

Note: OCDREG is located in the dedicated OCD address space. This address map is independent of the system address map. See [section 2.5.2. Cortex-M23 Peripheral Address Map](#).

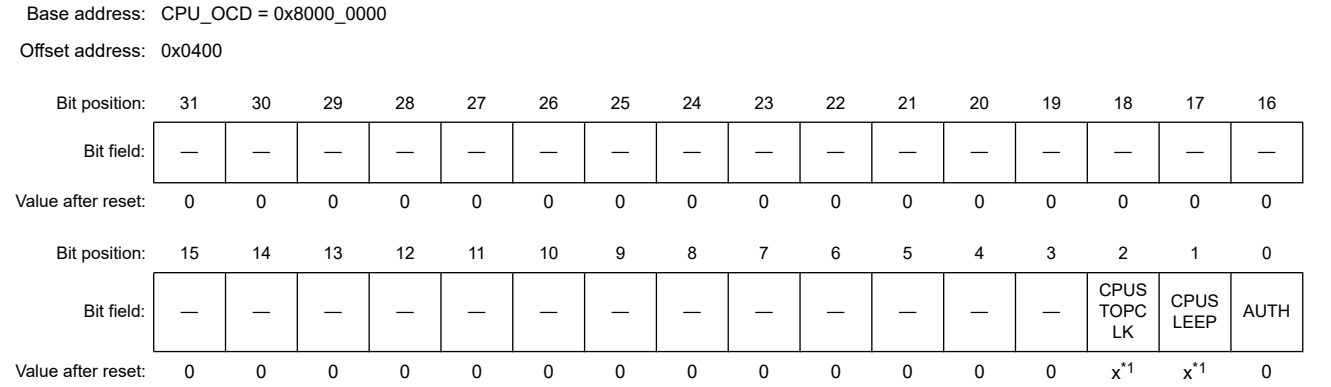
2.5.6.1 IAUTHn : ID Authentication Code Register (n = 0 to 3)

Four authentication registers are provided for writing the 128-bit key. These registers must be written in sequential order from IAUTHn (n = 0 to 3).

The initial value of the registers is all 0xFFFFFFFF. This means that SWD access is initially permitted when the ID code in the OSIS register has the initial value. See [section 2.7.1. ID Code](#).



2.5.6.2 MCUSTAT : MCU Status Register



Bit	Symbol	Function	R/W
0	AUTH	Authentication Status 0: Authentication failed 1: Authentication succeeded	R
1	CPUSLEEP	Sleep Mode Status 0: CPU is not in Sleep mode 1: CPU in Sleep mode	R
2	CPUSTOPCLK	CPU Clock Status 0: CPU clock is not stopped. The MCU is in neither Software Standby nor Snooze mode. 1: CPU clock is stopped. The MCU is in Software Standby or Snooze mode.	R
31:3	—	These bits are read as 0.	R

表 2.13OCDREG 寄存器

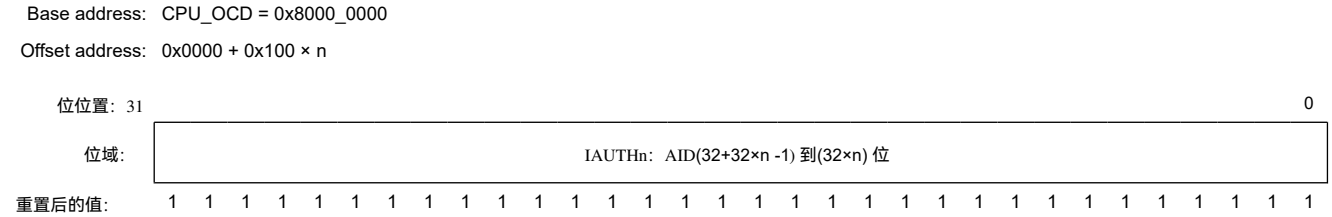
Name		DAP端口	地址	访问规模	R/W
ID认证码注册 0	IAUTH0	端口 1	0x8000_0000	32 位	W
身份认证码注册 1	IAUTH1	端口 1	0x8000_0100	32 位	W
ID认证码注册 2	IAUTH2	端口 1	0x8000_0200	32 位	W
身份验证码注册 3	IAUTH3	端口 1	0x8000_0300	32 位	W
MCU状态寄存器	MCUSTAT	端口 1	0x8000_0400	32 位	R
MCU 控制寄存器	MCUCTRL	端口 1	0x8000_0410	32 位	R/W

注意：OCDREG 位于专用的 OCD 地址空间中。此地址映射独立于系统地址映射。请参阅 2.5.2 节 Cortex-M23 外设地址映射。

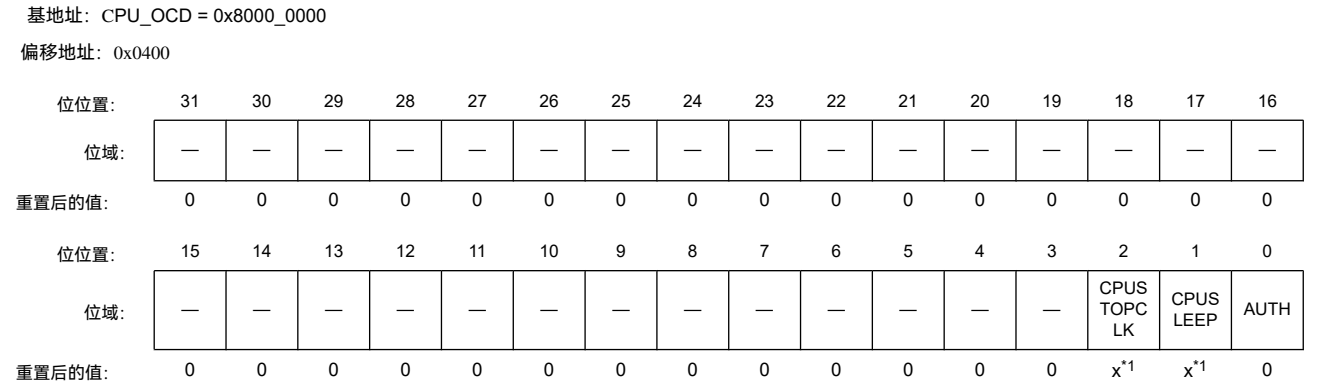
2.5.6.1IAUTHn：ID 认证码注册(n = 0至 3)

提供了四个认证寄存器用于写入 128 位密钥。这些寄存器必须按顺序写入，从 IAUTHn(n = 0到 3) 。

寄存器的初始值均为 0xFFFFFFFF。这意味着当 OSIS 寄存器中的 ID 代码具有初始值时，SWD 访问初始时是允许的。参见 2.7.1 节“ID 代码”。



2.5.6.2MCUSTAT：MCU 状态寄存器



少量	象征	功能	R/W
0	AUTH	身份验证状态 0: 身份验证失败 1: 身份验证成功	R
1	CPUSLEEP	睡眠模式状态 0: CPU 未处于睡眠模式 1: CPU处于睡眠模式	R
2	CPUSTOPCLK	CPU时钟状态0: CPU时钟未停止。 MCU既不处于软件待机模式，也不处于休眠模式。 1: CPU时钟停止。 MCU处于软件待机或休眠模式。	R
31:3	—	这些位读取为 0。	R

Note 1. Depends on the MCU status.

2.5.6.3 MCUCTRL : MCU Control Register

Base address: CPU_OCD = 0x8000_0000

Offset address: 0x0410

Bit position:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Bit field:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	—	—	—	—	—	—	—	DBIRQ	—	—	—	—	—	—	—	EDBG RQ
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
0	EDBGRQ	External Debug Request Writing 1 to the bit causes a CPU halt. When the EDBGRQ bit is set to 0 or the CPU is halted, the EDBGRQ bit is cleared. 0: Debug event not requested 1: Debug event requested	R/W
7:1	—	These bits are read as 0. The write value should be 0.	R/W
8	DBIRQ	Debug Interrupt Request Writing 1 to the bit wakes up the MCU from low power mode. The condition can be cleared by writing 0 to the DBIRQ bit. 0: Debug interrupt not requested 1: Debug interrupt requested	R/W
31:9	—	These bits are read as 0. The write value should be 0.	R/W

Note: Set DBIRQ and EDBGRQ to the same value.

2.5.6.4 OCDREG CoreSight component registers

The OCDREG module provides the CoreSight component registers defined in the Arm CoreSight architecture.

Table 2.14 shows the registers. See reference 4. in [section 2.8. References](#) for details of each register.

Table 2.14 OCDREG CoreSight component registers

Name	Address	Access size	R/W	Initial value
PIDR4	0x8000_0FD0	32 bits	R	0x00000004
PIDR5	0x8000_0FD4	32 bits	R	0x00000000
PIDR6	0x8000_0FD8	32 bits	R	0x00000000
PIDR7	0x8000_0FDC	32 bits	R	0x00000000
PIDR0	0x8000_0FE0	32 bits	R	0x00000004
PIDR1	0x8000_0FE4	32 bits	R	0x00000030
PIDR2	0x8000_0FE8	32 bits	R	0x0000000A
PIDR3	0x8000_0FEC	32 bits	R	0x00000000
CIDR0	0x8000_0FF0	32 bits	R	0x0000000D
CIDR1	0x8000_0FF4	32 bits	R	0x000000F0
CIDR2	0x8000_0FF8	32 bits	R	0x00000005
CIDR3	0x8000_0FFC	32 bits	R	0x000000B1

注1.取决于MCU状态。

2.5.6.3MCUCTRL： MCU 控制寄存器

基地址： CPU_OCD = 0x8000_0000

偏移地址： 0x0410

位位置：	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域：	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
重置后的值：	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位位置：	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域：	—	—	—	—	—	—	—	DBIRQ	—	—	—	—	—	—	—	EDBG RQ
重置后的值：	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
0	EDBGRQ	外部调试请求 向 EDBGRQ 位写入 1 会导致 CPU 停止运行。当 EDBGRQ 位被设置为 0 或 CPU 停止运行时，EDBGRQ 位将被清除。 0: 未请求调试事件；1: 已请求调试事件	R/W
7:1	—	这些位读取为 0。写入值应为 0。	R/W
8	DBIRQ	调试中断请求 向该位写入 1 可唤醒 MCU，使其脱离低功耗模式。向 DBIRQ 位写入 0 可清除该状态。 0: 未请求调试中断；1: 已请求调试中断	R/W
31:9	—	这些位读取为 0。写入值应为 0。	R/W

注意：将 DBIRQ 和 EDBGRQ 设置为相同的值。

2.5.6.4OCDREG CoreSight 组件寄存器

OCDREG 模块提供 Arm CoreSight 架构中定义的 CoreSight 组件寄存器。

表 2.14 显示了登记册。有关每个登记册的详细信息，请参见第 2.8 节中的参考文献 4。

表 2.14OCDREG CoreSight 组件寄存器

Name	地址	访问规模	R/W	初始值
PIDR4	0x8000_0FD0	32 位	R	0x00000004
PIDR5	0x8000_0FD4	32 位	R	0x00000000
PIDR6	0x8000_0FD8	32 位	R	0x00000000
PIDR7	0x8000_0FDC	32 位	R	0x00000000
PIDR0	0x8000_0FE0	32 位	R	0x00000004
PIDR1	0x8000_0FE4	32 位	R	0x00000030
PIDR2	0x8000_0FE8	32 位	R	0x0000000A
PIDR3	0x8000_0FEC	32 位	R	0x00000000
CIDR0	0x8000_0FF0	32 位	R	0x0000000D
CIDR1	0x8000_0FF4	32 位	R	0x000000F0
CIDR2	0x8000_0FF8	32 位	R	0x00000005
CIDR3	0x8000_0FFC	32 位	R	0x000000B1

2.6 SysTick Timer

The SysTick timer provides a simple 24-bit down counter. The reference clock for the timer can be selected as the CPU clock (ICLK) or SysTick timer clock (SYSTICCLK). See [section 8, Clock Generation Circuit](#) and reference 1. in [section 2.8. References](#) for details.

2.7 OCD Emulator Connection

A SWD authentication mechanism checks access permission for debug and MCU resources. To obtain full debug functionality, a pass result of the authentication mechanism is required.

[Figure 2.3](#) shows a block diagram of the authentication mechanism.

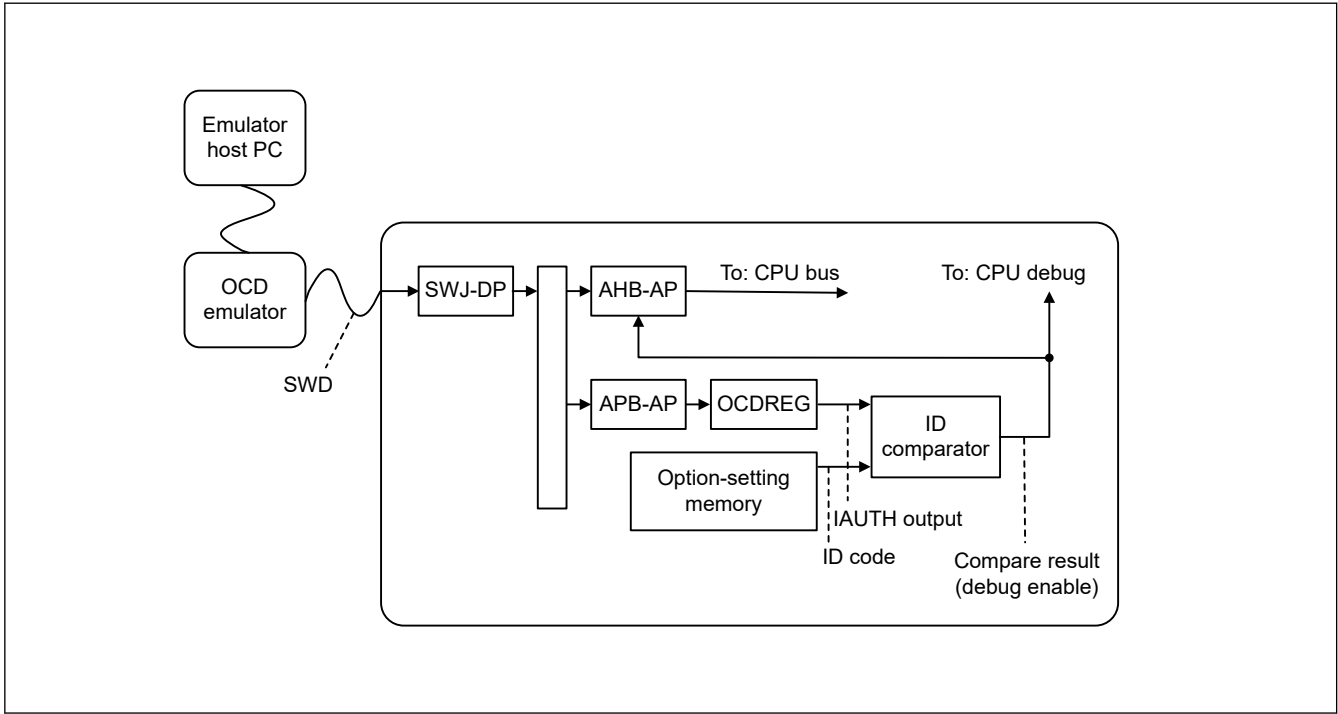


Figure 2.3 SWD authentication mechanism block diagram

An ID comparator is available in the MCU for authentication. The comparator compares the 128-bit IAUTH output from the OCDREG and the 128-bit ID code from the option-setting memory. When the two outputs are identical, the CPU debug functions and system bus access from the OCD emulator are permitted.

After the OCD emulator gets access permission, the OCD emulator must set the DBGEN bit in the System Control OCD Control Register (SYOCDRCR). In addition, the OCD emulator must clear the DBGEN bit before disconnecting. See [section 9.2.8. SYOCDRCR : System Control OCD Control Register](#).

2.7.1 ID Code

The ID code is used for checking permission for debug and access to on-chip resources. If the ID code matches the 128-bit data written in the ID Authentication Code Registers 0 to 3, the SWD debugger obtains access permission. ID code is written in the OCD/Serial Programmer ID Setting Register (OSIS) in the option-setting memory. The initial value of the ID code is all 1s (0xFFFFFFFF_FFFFFFFF_FFFFFFFF_FFFFFFFF). See [section 6, Option-Setting Memory](#) for details.

2.7.2 DBGEN

After the OCD emulator gets access permission, the OCD emulator must set the DBGEN bit in the System Control OCD Control Register (SYOCDRCR). In addition, the OCD emulator must clear the DBGEN bit before disconnecting it. See [section 9, Low Power Modes](#) for details.

2.7.3 Restrictions on Connecting an OCD emulator

This section describes the restrictions on emulator access.

2.6 系统滴答定时器

SysTick 定时器提供一个简单的 24 位递减计数器。定时器的参考时钟可以选择为 CPU 时钟 (ICLK) 或 SysTick 定时器时钟 (SYSTICCLK)。详情请参见第 8 节“时钟生成电路”和第 2.8 节“参考文献”中的参考文献 1。

2.7OCD模拟器连接

SWD认证机制用于检查对调试和MCU资源的访问权限。要获得完整的调试功能，必须通过认证机制。

图 2.3 显示了认证机制的框图。

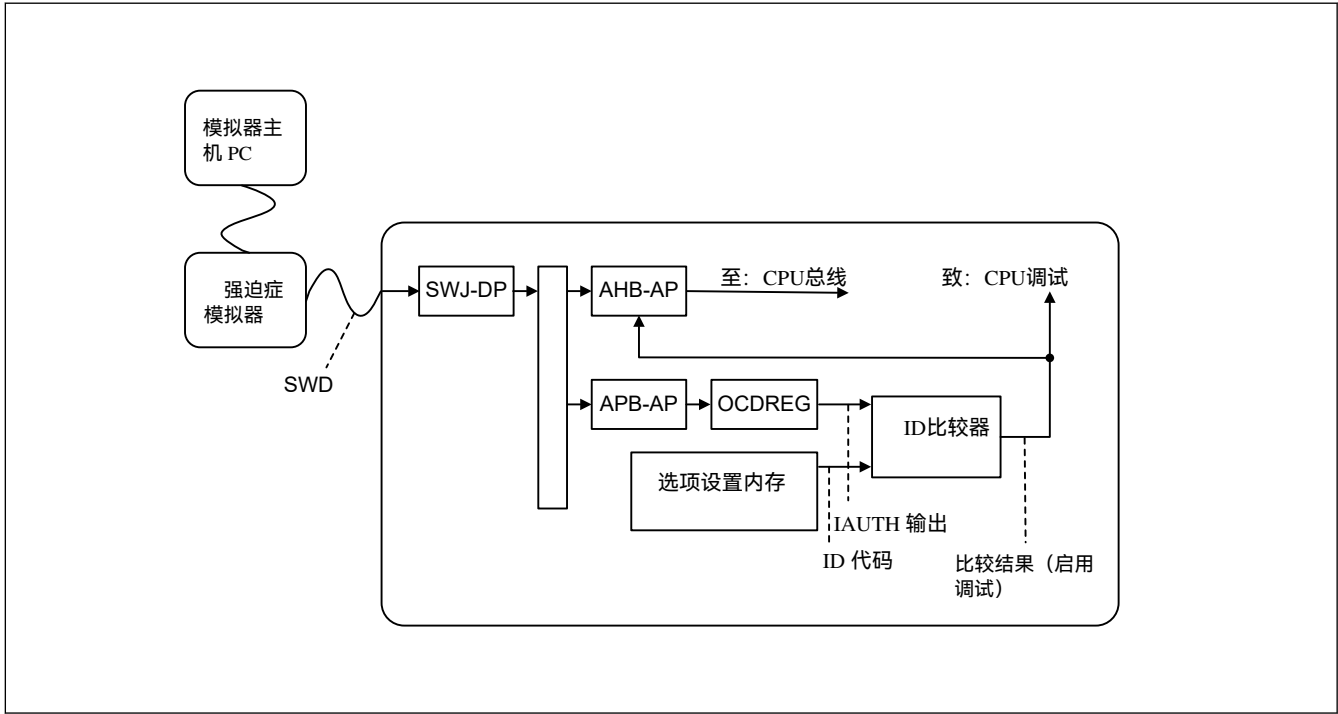


图 2.3SWD 认证机制框图

MCU 中有一个用于身份验证的 ID 比较器。该比较器将 OCDREG 输出的 128 位 IAUTH 数据与选项设置存储器中的 128 位 ID 代码进行比较。当这两个输出完全一致时，OCD 仿真器才能执行 CPU 调试功能并访问系统总线。

OCD 模拟器获得访问权限后，必须设置系统控制 OCD 控制寄存器 (SYOCDRCR) 中的 DBGEN 位。此外，OCD 模拟器在断开连接前必须清除 DBGEN 位。参见 9.2.8 节。SYOCDRCR: 系统控制 OCD 控制寄存器。

2.7.1 ID 代码

ID 代码用于检查调试权限和片上资源访问权限。如果 ID 代码与写入 ID 认证码寄存器 0 到 3 的 128 位数据匹配，则 SWD 调试器获得访问权限。ID 代码写入选项设置存储器中的 OCD/串行编程器 ID 设置寄存器 (OSIS)。ID 代码的初始值全部为 1 (0xFFFFFFFF_FFFFFFFF_FFFFFFFF_FFFFFFFF)。有关详细信息，请参阅第 6 节“选项设置存储器”。

2.7.2 DBGEN

OCD 模拟器获得访问权限后，必须在系统控制 OCD 控制寄存器 (SYOCDRCR) 中设置 DBGEN 位。此外，OCD 模拟器在断开连接前必须清除 DBGEN 位。详情请参见第 9 节“低功耗模式”。

2.7.3 连接强迫症模拟器的限制

本节介绍对模拟器访问的限制。

2.7.3.1 Starting connection while in low power mode

When starting a SWD connection from an OCD emulator, the MCU must be in Normal or Sleep mode. If the MCU is in Software Standby or Snooze mode, the OCD emulator can cause the MCU to hang.

2.7.3.2 Changing low power mode while in OCD mode

When the MCU is in OCD mode, the low power mode can be changed. However, system bus access from AHB-AP is prohibited in Software Standby or Snooze mode. Only SWJ-DP, APB-AP, and OCDREG can be accessed from the OCD emulator in these modes. Table 2.15 shows the restrictions.

Table 2.15 Restrictions by mode

Active mode	Start OCD emulator connection	Change low power mode	Access AHB-AP and system bus	Access APB-AP and OCDREG
Normal	Yes	Yes	Yes	Yes
Sleep	Yes	Yes	Yes	Yes
Software Standby	No	Yes	No	Yes
Snooze	No	Yes	No	Yes

If system bus access is required in Software Standby or Snooze mode, set the MCUCTRL.DBIRQ bit in OCDREG to wake up the MCU from the low power modes. Simultaneously, by asserting the MCUCTRL.DBIRQ bit in OCDREG, the OCD emulator can wake up the MCU without starting CPU execution by using a CPU break.

2.7.3.3 Modify the ID code in OSIS

After modifying the ID code in the OSIS, the OCD emulator must reset the MCU by asserting the RES pin or setting the SYSRESETREQ bit of the Application Interrupt and Reset Control Register in the system control block to 1. The modified ID code is reflected after reset. For the system control block, see reference 2. listed in section 2.8. References.

The emulator must set the modified ID code in the IAUTH0 to IAUTH3 registers immediately before the MCU is placed in the reset state. When the IAUTH0 to IAUTH3 registers have been overwritten, writing to the SYSRESETREQ bit is not possible. Place the MCU in the reset state by asserting the signal on the RES pin.

2.7.3.4 Connecting sequence and SWD authentication

Because the OCD emulator is protected by the SWD authentication mechanism, the OCD might be required to input the ID code to the SWD authentication registers. The OSIS value in the option-setting memory determines whether the code is required. After negation of the RES pin, a wait time is required before comparing the OSIS value at cold start. See section 27.4.1. Reset Timing. The SWD authentication process is described in detail below.

(1) When MSB of the OSIS register is 0 (bit [127] = 0)

The ID code is always a mismatch and connection to the OCD is prohibited.

(2) When bits in the OSIS register is all 1s (initial value)

ID authentication is not required and the OCD can use the AHB-AP without authentication. For details of the settings for using the AHB-AP, see reference 4. in section 2.8. References.

1. Connect the OCD emulator to the MCU through the SWD interface.
2. Set up SWJ-DP to access the DAP bus. In the setup, the OCD emulator must assert CDBGPWRUPREQ in the SWJ-DP Control Status Register, then wait until CDBGPWRUPACK in the same register is asserted.
3. Set up the AHB-AP to access the system address space. The AHB-AP is connected to the DAP bus port 0.
4. Start accessing the CPU debug resources using the AHB-AP.

(3) When OSIS[127:126] = 10b

ID authentication is required and the OCD must write the unlock code to the IAUTH registers 0 to 3 in OCDREG before using the AHB-AP.

1. Connect the OCD debugger to the MCU through the SWD interface.

2.7.3.1 低功耗模式下的启动连接

从 OCD 模拟器启动 SWD 连接时，MCU 必须处于正常模式或睡眠模式。如果 MCU 处于软件待机或休眠模式，OCD 模拟器可能会导致 MCU 死机。

2.7.3.2 在 OCD 模式下更改低功耗模式

当MCU处于OCD模式时，可以更改低功耗模式。但是，在软件待机或休眠模式下，禁止通过AHB-AP访问系统总线。在这些模式下，OCD仿真器只能访问SWJ-DP、APB-AP和OCDREG。表2.15列出了这些限制。

表 2.15 按模式划分的限制

活动模式	启动强迫症模拟器连接	更改低功耗模式	接入 AHB-AP 和系统总线	访问 APB-AP 和 OCDREG
普通的	Yes	Yes	Yes	Yes
睡觉	Yes	Yes	Yes	Yes
软件待机	No	Yes	No	Yes
打盹	No	Yes	No	Yes

如果在软件待机或休眠模式下需要访问系统总线，请设置 OCDREG 中的 MCUCTRL.DBIRQ 位，以将 MCU 从低功耗模式唤醒。同时，通过置位 OCDREG 中的 MCUCTRL.DBIRQ 位，OCD 仿真器可以使用 CPU 中断来唤醒 MCU，而无需启动 CPU 执行。

2.7.3.3 修改 OSIS 中的 ID 代码

在 OSIS 中修改 ID 代码后，OCD 仿真器必须通过置位 RES 引脚或将系统控制块中应用中断和复位控制寄存器的 SYSRESETREQ 位设置为 1 来复位 MCU。复位后，修改后的 ID 代码才会生效。有关系统控制块，请参阅 2.8 节“参考文献”中列出的参考文献 2。

仿真器必须在MCU进入复位状态之前，立即将修改后的ID代码写入IAUTH0至IAUTH3寄存器。如果IAUTH0至IAUTH3寄存器已被覆盖，则无法写入SYSRESETREQ位。通过置位RES引脚上的信号，使MCU进入复位状态。

2.7.3.4 连接顺序和SWD认证

由于 OCD 仿真器受 SWD 认证机制保护，因此 OCD 可能需要将 ID 代码输入到 SWD 认证寄存器中。选项设置存储器中的 OSIS 值决定是否需要输入代码。冷启动时，在 RES 引脚取反后，需要等待一段时间才能比较 OSIS 值。请参阅 27.4.1 节“复位时序”。SWD 认证过程将在下文详细描述。

(1) 当 OSIS 寄存器的最高有效位为 0 时（位 [127] = 0）

ID 代码始终不匹配，禁止与 OCD 连接。

(2) 当 OSIS 寄存器中的位全部为 1 时（初始值）

无需身份验证，OCD 无需身份验证即可使用 AHB-AP。有关使用 AHB-AP 的设置详情，请参阅第 2.8 节“参考资料”中的参考文献 4。

1. 通过 SWD 接口将 OCD 模拟器连接到 MCU。
2. 设置 SWJ-DP 以访问 DAP 总线。在设置过程中，OCD 仿真器必须将 SWJ-DP 控制状态寄存器中的 CDBGPWRUPREQ 置位，然后等待同一寄存器中的 CDBGPWRUPACK 置位。
3. 配置 AHB-AP 以访问系统地址空间。AHB-AP 连接到 DAP 总线端口 0。
4. 使用 AHB-AP 开始访问 CPU 调试资源。

(3) When OSIS[127:126] = 10b

使用 AHB-AP 之前，需要进行身份认证，并且 OCD 必须将解锁码写入 OCDREG 中的 IAUTH 寄存器 0 到 3。

1. 通过 SWD 接口将 OCD 调试器连接到 MCU。

- 2. Set up SWJ-DP to access the DAP bus. In the setup, the OCD emulator must assert CDBGPWRUPREQ in SWJ-DP Control Status Register, then wait until CDBGPWRUPACK in the same register is asserted.
- 3. Set up the APB-AP to access OCDREG. The APB-AP is connected to the DAP bus port 1.
- 4. Write the 128-bit ID code to IAUTH registers 0 to 3 in OCDREG using the APB-AP.
- 5. If the 128-bit ID code matches the OSIS value, the AHB-AP is authorized to issue an AHB transaction. The authorization result can be confirmed by the AUTH bit in the MCUSTAT Register or the DbgStatus bit in the AHB-AP Control Status Word Register.
 - When the DbgStatus bit is 1, the 128-bit ID code is a match with the OSIS value. AHB transfers are permitted.
 - When the DbgStatus bit is 0, the 128-bit ID code is not a match with the OSIS value. AHB transfers are not permitted.
- 6. Set up the AHB-AP to access the system address space. The AHB-AP is connected to the DAP bus port 0.
- 7. Start accessing the CPU debug resources using the AHB-AP.

(4) When OSIS[127:126] = 11b

OCD authentication is required and the OCD emulator must write the ID code to the IAUTH registers 0 to 3 in OCDREG. The connection sequence is the same when OSIS[127:126] is 10b except for “ALeRASE” capability.

When IATUH registers 0 to 3 are “ALeRASE” in ASCII code, the contents of the code flash, and configuration area are erased at once. See [section 25, Flash Memory](#) for details.

The ALeRASE sequence is as follows:

- 1. Connect the OCD debugger to the MCU through the SWD interface.
- 2. Set up SWJ-DP to access the DAP bus. In the setup, the OCD emulator must assert CDBGPWRUPREQ in the SWJ-DP Control Status Register, then wait until CDBGPWRUPACK in the same register is asserted.
- 3. Set up the APB-AP to access OCDREG. The APB-AP is connected to the DAP bus port 1.
- 4. Write the 128-bit ID code to IAUTH registers 0 to 3 in OCDREG using the APB-AP.
- 5. If the 128-bit ID code is “ALeRASE” in ASCII code (0x414C_6552_4153_45FF_FFFF_FFFF_FFFF_FFFF), the contents of the code flash, and configuration area are erased. Thereafter, the MCU transitions to Sleep mode.

2.8 References

- 1. *ARM®v8-M Architecture Reference Manual* (ARM DDI 0553B.a)
- 2. *ARM®Cortex®-M23 Processor Technical Reference Manual* (ARM DDI 0550C)
- 3. *ARM®Cortex®-M23 Device Generic User Guide* (ARM DUI 1095A)
- 4. *ARM®CoreSight™SoC-400 Technical Reference Manual* (ARM DDI 0480G)
- 5. *ARM®CoreSight™Architecture Specification* (ARM IHI 0029E)
- 6. *ARM®CoreSight™MTB-M23 Technical Reference Manual* (ARM DDI 0564C)

2.9 Usage Notes

In TBLOFF field of the Vector Table Offset Register (VTOR), TBLOFF[31:8] are valid and writing 1 to TBLOFF[7] bit is ignored. For the detail of Vector Table Offset Register, see reference 3. listed in [section 2.8. References](#).

- 2. 设置 SWJ-DP 以访问 DAP 总线。在设置过程中，OCD 仿真器必须将 SWJ-DP 控制状态寄存器中的 CDBGPWRUPREQ 置位，然后等待同一寄存器中的 CDBGPWRUPACK 置位。
- 3. 设置 APB-AP 以访问 OCDREG。APB-AP 连接到 DAP 总线端口 1。
- 4. 使用 APB-AP 将 128 位 ID 代码写入 OCDREG 中的 IAUTH 寄存器 0 到 3。
- 5. 如果 128 位 ID 代码与 OSIS 值匹配，则 AHB-AP 被授权发出 AHB 事务。授权结果可通过 MCUSTAT 寄存器中的 AUTH 位或 AHB-AP 控制状态字寄存器中的 DbgStatus 位进行确认。
 - 当 DbgStatus 位为 1 时，128 位 ID 代码与 OSIS 值匹配。允许进行 AHB 传输。
 - 当 DbgStatus 位为 0 时，128 位 ID 代码与 OSIS 值不匹配。不允许进行 AHB 传输。

- 6. 配置 AHB-AP 以访问系统地址空间。AHB-AP 连接到 DAP 总线端口 0。
- 7. 使用 AHB-AP 开始访问 CPU 调试资源。

(4) When OSIS[127:126] = 11b

需要进行 OCD 身份验证，并且 OCD 模拟器必须将 ID 代码写入 OCDREG 中的 IAUTH 寄存器 0 到 3。当 OSIS[127:126] 为 10b 时，连接顺序相同，但“ALeRASE”功能除外。

当 IATUH 寄存器 0 到 3 的 ASCII 码为“ALeRASE”时，代码闪存和配置区域的内容将被立即擦除。详情请参见第 25 节“闪存”。

ALeRASE序列如下：

- 1. 通过 SWD 接口将 OCD 调试器连接到 MCU。
- 2. 设置 SWJ-DP 以访问 DAP 总线。在设置过程中，OCD 仿真器必须将 SWJ-DP 控制状态寄存器中的 CDBGPWRUPREQ 置位，然后等待同一寄存器中的 CDBGPWRUPACK 置位。
- 3. 设置 APB-AP 以访问 OCDREG。APB-AP 连接到 DAP 总线端口 1。
- 4. 使用 APB-AP 将 128 位 ID 代码写入 OCDREG 中的 IAUTH 寄存器 0 到 3。
- 5. 如果 128 位 ID 代码的 ASCII 码为“ALeRASE”（0x414C_6552_4153_45FF_FFFF_FFFF_FFFF_FFFF），则代码闪存和配置区域的内容将被擦除。之后，MCU 将进入睡眠模式。

2.8 参考文献

- 1.*ARM®v8M 架构参考手册* (ARM DDI 0553B.a)
- 2.*ARM®Cortex®-M23 处理器技术参考手册* (ARM DDI 0550C)
- 3.*ARM®Cortex®-M23 设备通用用户指南* (ARM DUI 1095A)
- 4.*ARM®CoreSight™SoC-400 技术参考手册* (ARM DDI 0480G)
- 5.*ARM®CoreSight™架构规范* (ARM IHI 0029E)
- 6.*ARM®CoreSight™MTB-M23 技术参考手册* (ARM DDI 0564C)

2.9 使用说明

在向量表偏移寄存器 (VTOR) 的 TBLOFF 字段中，TBLOFF[31:8] 为有效值，并且向 TBLOFF [7] 位写入 1 将被忽略。有关向量表偏移寄存器的详细信息，请参阅第 2.8 节“参考文献”中列出的参考文献 3。

3. Operating Modes

3.1 Overview

The MCU starts in single-chip mode and the on-chip flash is enabled when a reset is released. In single-chip mode, all I/O pins are available for use as input or output port, inputs or outputs for peripheral functions, or as interrupt inputs.

3.2 Operating Modes Transitions

3.2.1 Operating Mode Transitions

Figure 3.1 shows operating mode transitions.

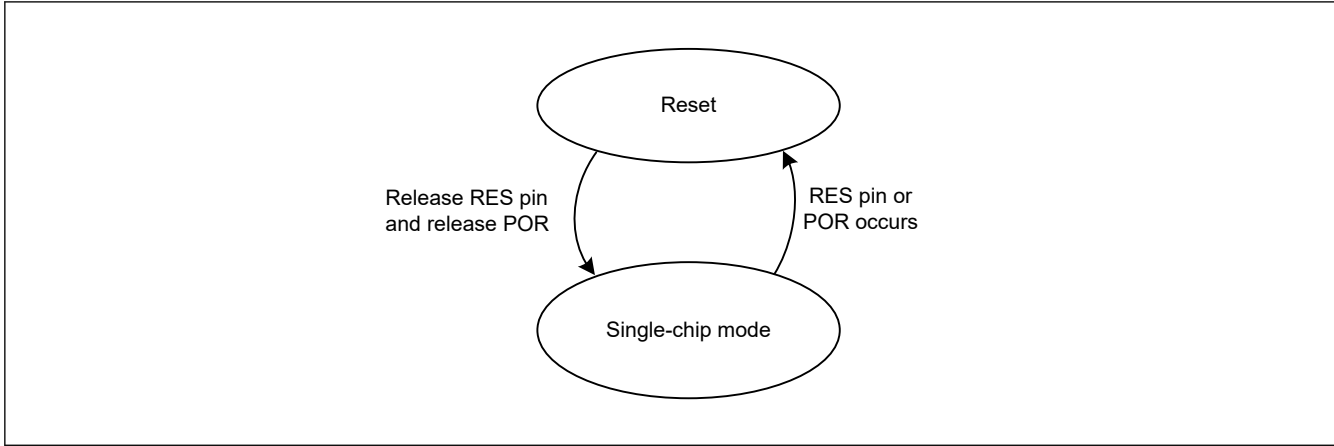


Figure 3.1 Operating mode

3.操作模式

3.1 概述

MCU 以单芯片模式启动，复位信号释放后，片上闪存被启用。在单芯片模式下，所有 I/O 引脚均可用作输入/输出端口、外设功能的输入/输出，或用作中断输入。

3.2 操作模式转换

3.2.1 运行模式转换

图 3.1 显示了运行模式转换。

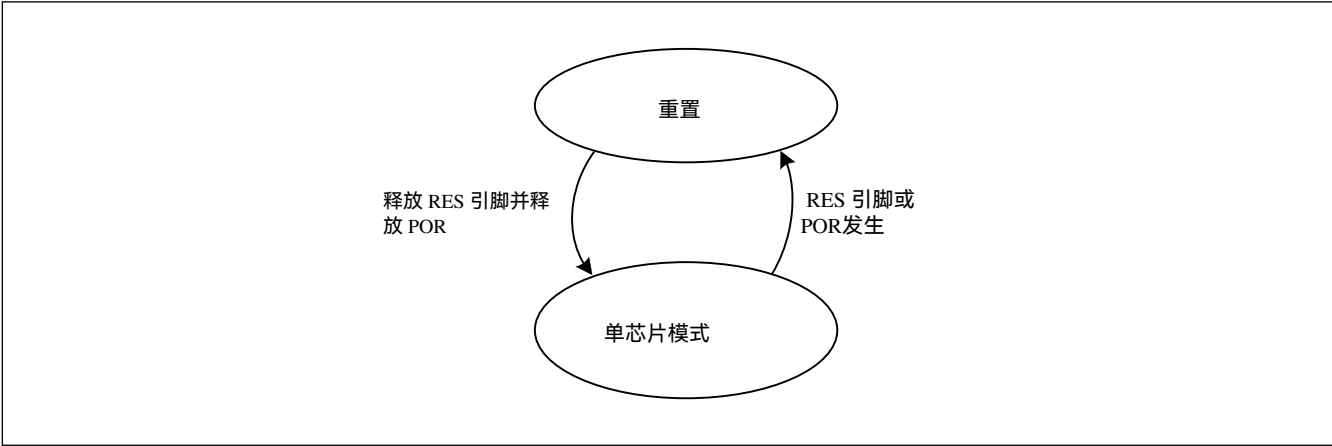


图 3.1 运行模式

4. Address Space

4.1 Address Space

The MCU supports a 4-GB linear address space ranging from 0x0000_0000 to 0xFFFF_FFFF that can contain both program and data. [Figure 4.1](#) shows the memory map of a 16-KB flash product.

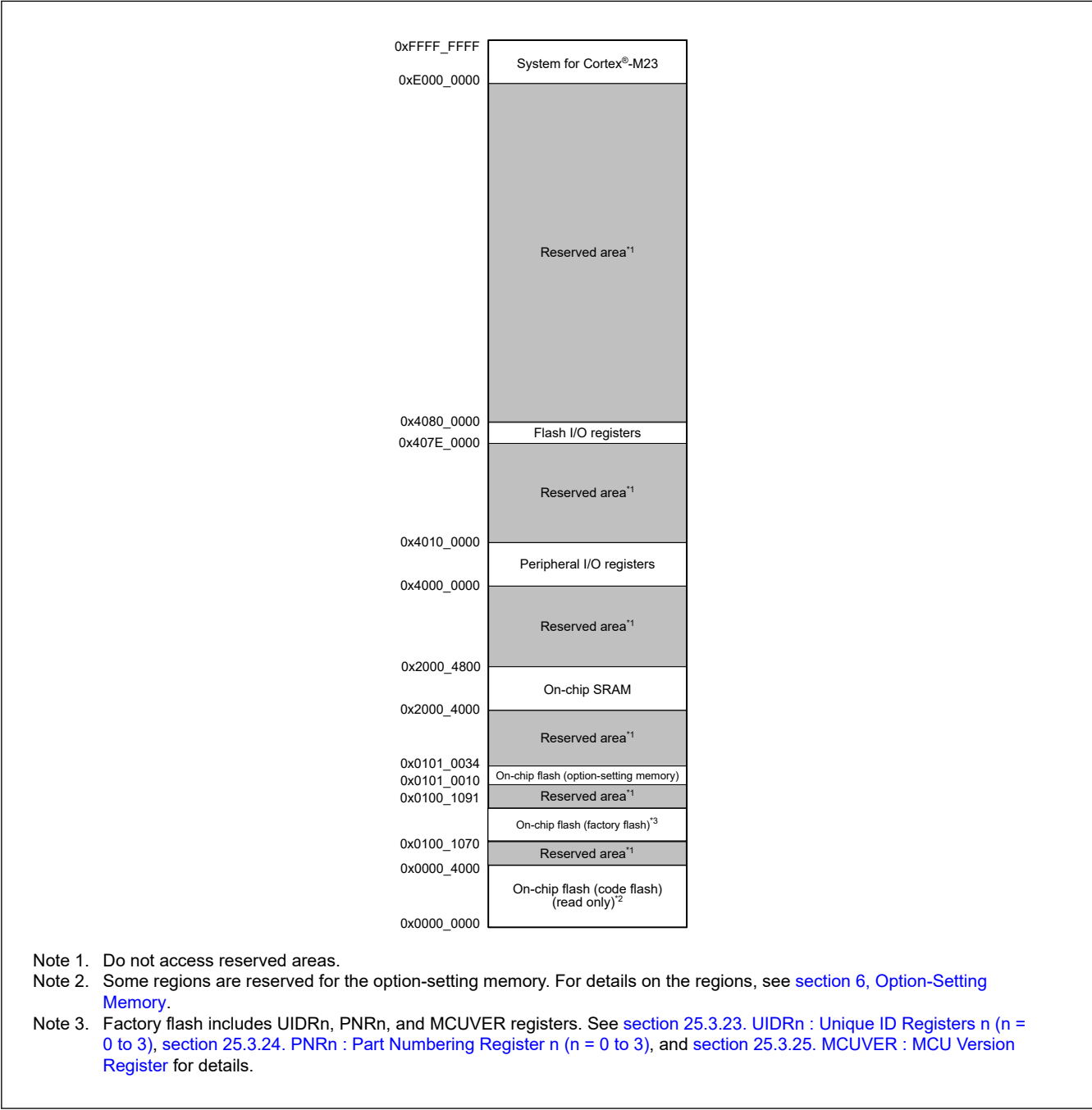


Figure 4.1 Memory map

4.地址空间

4.1地址空间

该MCU支持从0x0000_0000 到0xFFFF_FFFF的4GB线性地址空间，可同时容纳程序和数据。图4.1显示了一个16KB闪存产品的内存映射图。

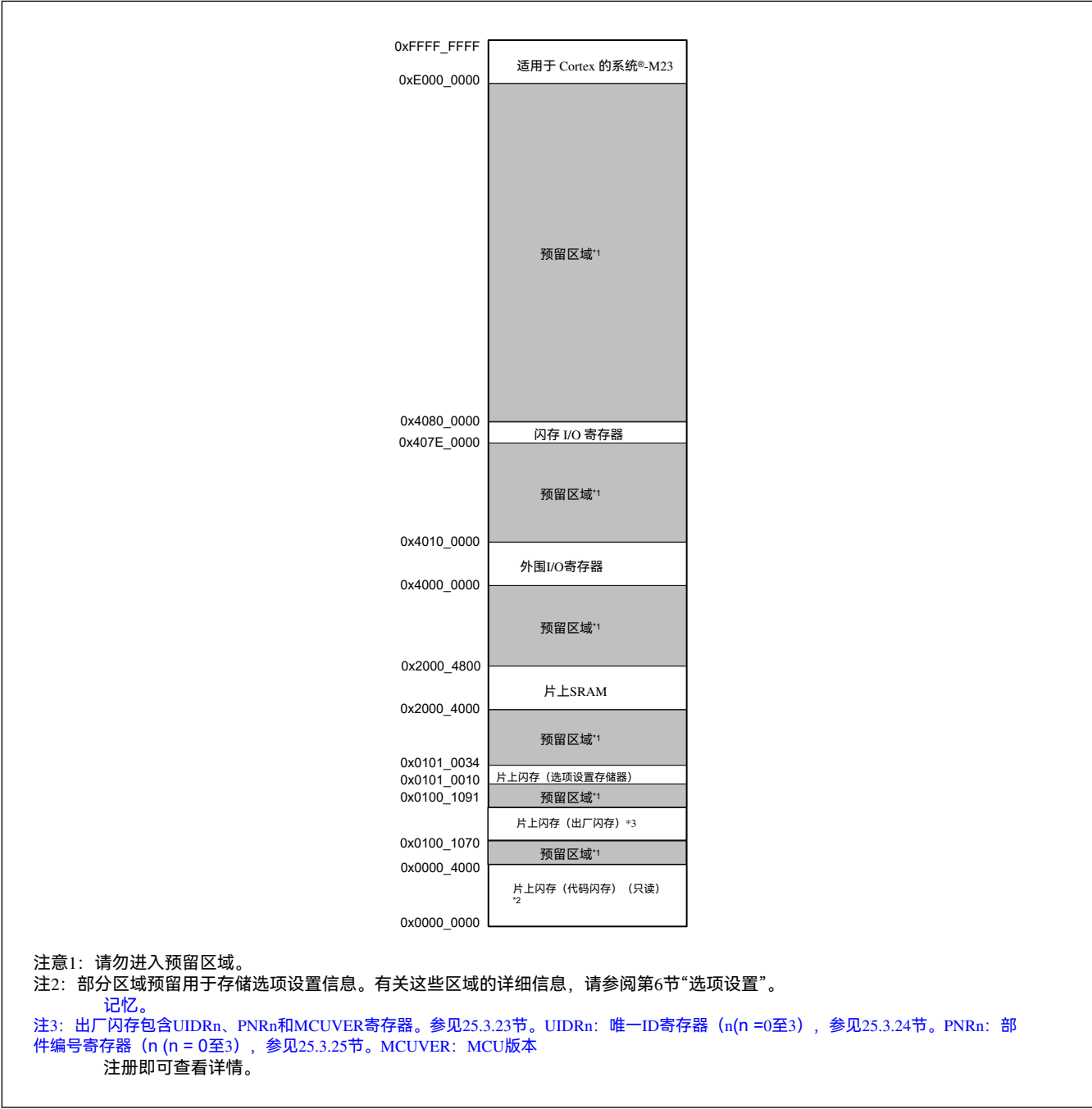


图 4.1 内存映射

5. Resets

5.1 Overview

The MCU provides 6 resets. [Table 5.1](#) lists the reset names and sources.

Table 5.1 Reset names and sources

Reset name	Source
RES pin reset	Voltage input to the RES pin is driven low
Power-on reset	VCC rise (voltage detection V _{POR}) and VCC fall (voltage detection V _{PDR})*1
Independent watchdog timer reset	IWDT underflow or refresh error
Voltage monitor 0 reset	VCC fall (voltage detection V _{det0})*1
Voltage monitor 1 reset	VCC rise/fall (voltage detection V _{det1})*1
Software reset	Register setting (use the Arm® software reset bit AIRCR.SYSRESETREQ)

Note 1. For details on the voltages to be monitored (V_{POR}, V_{det0}, and V_{det1}), see [section 7, Low Voltage Detection \(LVD\)](#) and [section 27, Electrical Characteristics](#).

The internal state and pins are initialized by a reset. [Table 5.2](#) and [Table 5.3](#) list the targets initialized by resets.

Table 5.2 Reset detect flags initialized by each reset source (1 of 2)

Flags to be initialized	Reset source			
	RES pin reset	Power-on reset	Voltage monitor 0 reset	Independent watchdog timer reset
Power-On Reset Detect Flag (PORSR.PORF)	—	✓	—	—
Voltage Monitor 0/1 Reset Detect Flag (RESF.LVIRF)	✓	✓	—	—
Independent Watchdog Timer Reset Detect Flag (RESF.IWDTRF)	✓	✓	—	—
Software Reset Detect Flag (RESF.SWRF)	✓	✓	—	—

Table 5.2 Reset detect flags initialized by each reset source (2 of 2)

Flags to be initialized	Reset source	
	Voltage monitor 1 reset	Software reset
Power-On Reset Detect Flag (PORSR.PORF)	—	—
Voltage Monitor 0/1 Reset Detect Flag (RESF.LVIRF)	—	—
Independent Watchdog Timer Reset Detect Flag (RESF.IWDTRF)	—	—
Software Reset Detect Flag (RESF.SWRF)	—	—

Note: ✓ : Initialized to 0
 — : Not initialized

Table 5.3 Module-related registers initialized by each reset source (1 of 3)

Registers to be initialized		Reset source			
		RES pin reset	Power-on reset	Voltage monitor 0 reset	Independent watchdog timer reset
Registers related to the IWDT	IWDTRR, IWDTSR	✓	✓	✓	✓

5. 重置

5.1 概述

MCU 提供 6 种复位方式。表 5.1 列出了复位名称和复位源。

表 5.1 重置名称和来源

重置名称	来源
RES引脚复位	RES引脚的输入电压被拉低
上电重置	VCC 上升（电压检测V _{POR} ）和 VCC 下降（电压检测V _{PDR} ）*1
独立看门狗定时器复位	IWDT 下溢或刷新错误
电压监控器 0 重置	VCC 下降（电压检测 V _{det0} ）*1
电压监控器 1 重置	VCC 上升/下降（电压检测 V _{det1} ）*1
软件重置	寄存器设置（使用Arm®软件复位位 AIRCR.SYSRESETREQ）

注 1. 有关要监测的电压(V_{POR}, V_{det0}, 和V_{det1})的详细信息，请参阅第 7 节“低电压检测 (LVD)”和第 27 节“电气特性”。

内部状态和引脚通过复位进行初始化。表 5.2 和表 5.3 列出了通过复位初始化的目标。

表 5.2 各个复位源初始化的复位检测标志（1/2）

待初始化的标志	重置源			
	RES引脚复位	上电重置	电压监控器 0 重置	独立看门狗定时器复位
上电复位检测标志 (PORSR.PORF)	—	✓	—	—
电压监视器 0/1 复位检测标志 (RESF.LVIRF)	✓	✓	—	—
独立看门狗定时器复位检测标志 (RESF.IWDTRF)	✓	✓	—	—
软件复位检测标志 (RESF.SWRF)	✓	✓	—	—

表 5.2 各个复位源初始化的复位检测标志（2/2）

待初始化的标志	重置源	
	电压监控器 1 重置	软件重置
上电复位检测标志 (PORSR.PORF)	—	—
电压监视器 0/1 复位检测标志 (RESF.LVIRF)	—	—
独立看门狗定时器复位检测标志 (RESF.IWDTRF)	—	—
软件复位检测标志 (RESF.SWRF)	—	—

注意: : 已初始化为 0
 —: 未初始化

表 5.3 各复位源初始化的模块相关寄存器（1/3）

待初始化的寄存器		重置源			
		RES引脚复位	上电重置	电压监控器 0 重置	独立看门狗定时器复位
与以下相关的登记册 IWDT	IWDTRR, IWDTSR	✓	✓	✓	✓

Table 5.3 Module-related registers initialized by each reset source (2 of 3)

Registers to be initialized		Reset source			
		RES pin reset	Power-on reset	Voltage monitor 0 reset	Independent watchdog timer reset
Registers related to the voltage monitor function 1	LVD1CR, LVD1MKR, LVD1SR	✓	✓	✓	✓
Pin state		✓	✓	✓	✓
Registers other than those shown, CPU, and internal state		✓	✓	✓	✓

Table 5.3 Module-related registers initialized by each reset source (3 of 3)

Registers to be initialized		Reset source	
		Voltage monitor 1 reset	Software reset
Registers related to the IWDTRR, IWDTSR		✓	✓
Registers related to the voltage monitor function 1	LVD1CR, LVD1MKR, LVD1SR	✓	✓
Pin state		✓	✓
Registers other than those shown, CPU, and internal state		✓	✓

Note: ✓ : Initialized
 — : Not initialized

Table 5.4 show the states of LOCO when a reset occurs.

Table 5.4 States of LOCO when a reset occurs

		Reset source	
		POR/LVD0/LVD1	Other
LOCO	Enable or disable	Initialized to disable. However, during IWDTRR operation, LOCO oscillates regardless of the value of LCSTP.	

When a reset is canceled, reset exception handling starts.

Table 5.5 lists the pin related to the reset function.

Table 5.5 Pin related to reset

Pin name	I/O	Function
RES	Input	Reset pin

5.2 Register Descriptions

5.2.1 RESF : Reset Status Flag Register

Base address:	SYSC = 0x4001_E000							
Offset address:	0x0830							
Bit position:	7	6	5	4	3	2	1	0
Bit field:	—	—	SWRF	IWDTRF	—	—	—	LVIRF
Value after reset:	0	0	x ¹	x ¹	0	0	0	x ¹

表 5.3 各复位源初始化的模块相关寄存器 (2/3)

待初始化的寄存器		重置源			
		RES引脚复位	上电重置	电压监控器 0 重置	独立看门狗定时器复位
与电压监控功能相关的寄存器 1	LVD1CR、LVD1MKR、LVD1SR	✓	✓	✓	✓
引脚状态		✓	✓	✓	✓
除所示寄存器外的其他寄存器、CPU 和内部状态		✓	✓	✓	✓

表 5.3 各复位源初始化的模块相关寄存器 (3/3)

待初始化的寄存器		重置源	
		电压监控器 1 重置	软件重置
与以下相关的登记册 IWDTRR, IWDTSR		✓	✓
与电压监控功能相关的寄存器 1	LVD1CR、LVD1MKR、LVD1SR	✓	✓
引脚状态		✓	✓
除所示寄存器外的其他寄存器、CPU 和内部状态		✓	✓

注意： : 已初始化
 — : 未初始化

表 5.4 显示了 LOCO 在复位时的状态。

表 5.4 复位时 LOCO 的状态

		重置源	
		POR/LVD0/LVD1	其他
LOCO	启用或禁用	初始化为禁用状态。然而，在 IWDTRR 操作期间，无论 LCSTP 的值如何，LOCO 都会振荡。	

当重置操作被取消时，将启动重置异常处理。

表 5.5 列出了与复位功能相关的引脚。

表 5.5 复位引脚

引脚名称	I/O	功能
RES	输入	复位引脚

5.2寄存器描述

5.2.1RESF： 复位状态标志寄存器

基地址: SYSC = 0x4001E000

偏移地址: 0x0830

位位置:	7	6	5	4	3	2	1	0
位域:	—	—	SWRF	IWDTRF	—	—	—	LVIRF
重置后的值:	0	0	x ¹	x ¹	0	0	0	x ¹

Bit	Symbol	Function	R/W
0	LVIRF	Internal Reset Request by Voltage Detector (LVD0 or LVD1) 0: Internal reset request is not generated, or the RESF register is cleared. 1: Internal reset request is generated.	R
3:1	—	These bits are read as 0.	R
4	IWDTRF	Internal Reset Request by Independent Watchdog Timer (IWDT) 0: Internal reset request is not generated, or the RESF register is cleared. 1: Internal reset request is generated.	R
5	SWRF	Internal Reset Request by Software Reset 0: Internal reset request is not generated, or the RESF register is cleared. 1: Internal reset request is generated.	R
7:6	—	These bits are read as 0.	R

Note 1. The value after reset varies depending on the reset source.

The RESF register indicates which reset source generated the reset request.

This register is cleared after it is read, in addition to when a reset occurs as described in [Table 5.2](#).

5.2.2 PORSR : Power-On Reset Status Register

Base address:	SYSC = 0x4001_E000							
Offset address:	0x0831							
Bit position:	7	6	5	4	3	2	1	0
Bit field:	—	—	—	—	—	—	—	PORF
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
0	PORF	Checking Occurrence of Power-on Reset 0: A value 1 has not been written, or a power-on reset has occurred. 1: No power-on reset has occurred.*1	R/W
7:1	—	These bits are read as 0. The write value should be 0.	R/W

Note 1. If the PORF bit is set to 1, it guarantees that no power-on reset has occurred, but it does not guarantee that the RAM value is retained.

The PORSR register is used to check the occurrence of a Power-on reset.

This register is reset only by a Power-on reset; it retains the value when a reset caused by another source occurs.

PORF bit (Checking Occurrence of Power-on Reset)

The PORF bit indicates whether a power-on reset has occurred.

This bit is valid only for writing 1; writing 0 is invalid.

When checking whether a power-on reset has occurred, write 1 to this bit in advance.

5.3 Operation

5.3.1 RES Pin Reset

The RES pin generates this reset. When the RES pin is driven low, all the processing in progress is aborted and the MCU enters a reset state. To successfully reset the MCU, the RES pin must be held low for the power supply stabilization time specified at power-on.

When the RES pin is driven high from low, the internal reset is canceled after the post-RES cancellation wait time (t_{RESWT}, t_{RESWT2}) elapses. The CPU then starts the reset exception handling.

For details, see [section 27, Electrical Characteristics](#).

少量	象征	功能	R/W
0	LVIRF	电压检测器 (LVD0 或 LVD1) 的内部复位请求 0: 未生成内部复位请求, 或者 RESF 寄存器已清除。 1: 生成内部重置请求。	R
3:1	—	这些位读取为 0。	R
4	IWDTRF	独立看门狗定时器 (IWDT) 内部复位请求 0: 未生成内部复位请求, 或者 RESF 寄存器已清除。 1: 生成内部重置请求。	R
5	SWRF	软件重置请求内部重置 0: 未生成内部复位请求, 或者 RESF 寄存器已清除。 1: 生成内部重置请求。	R
7:6	—	这些位读取为 0。	R

注1. 复位后的值取决于复位源。

RESF寄存器指示哪个复位源生成了复位请求。

除了表 5.2 中所述的复位操作外, 该寄存器在读取后也会被清除。

5.2.2 PORSR: 上电复位状态寄存器

基地址: SYSC = 0x4001E000

偏移地址: 0x0831

位位置:	7	6	5	4	3	2	1	0
位域:	—	—	—	—	—	—	—	PORF
重置后的值:	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
0	PORF	检查上电复位是否发生 0: 尚未写入值 1, 或者发生了上电复位。 1: 未发生上电重置。*1	R/W
7:1	—	这些位读取为 0。写入值应为 0。	R/W

注 1. 如果 PORF 位设置为 1, 则保证不会发生上电复位, 但不能保证 RAM 值得以保留。

PORSR 寄存器用于检查上电复位是否发生。

该寄存器仅通过上电复位进行复位; 当发生由其他原因引起的复位时, 它会保持其值。

PORF 位 (检查上电复位事件发生情况)

PORF 位指示是否发生了上电复位。

该位仅对写入 1 有效; 写入 0 无效。

检查是否发生上电复位时, 预先将 1 写入该位。

5.3 操作

5.3.1 RES 引脚复位

RES 引脚产生此复位信号。当 RES 引脚被拉低时, 所有正在进行的处理都会中止, MCU 进入复位状态。要成功复位 MCU, 必须在开机时指定的电源稳定时间内保持 RES 引脚为低电平。

当 RES 引脚由低电平拉高后, 经过 RES 取消等待时间 (t_{RESWT}, t_{RESWT2}) 后, 内部复位将被取消。然后 CPU 开始处理复位异常。

有关详细信息, 请参阅第 27 节“电气特性”。

5.3.2 Power-On Reset

The power-on reset (POR) is an internal reset generated by the power-on reset circuit. A power-on reset is generated under the following conditions.

1. If the RES pin is in a high level state when power is supplied
2. If the RES pin is in a high level state when VCC is below V_{POR}

After VCC exceeds V_{POR} and the specified power-on reset time (t_{POR}) elapses, the CPU starts the reset exception handling. The power-on reset time is a stabilization period of the external power supply and the MCU circuit.

After a power-on reset is generated, the PORF flag in the PORSR is clear to 0. When VCC falls below V_{POR} , a power-on reset state is occurred.

Figure 5.1 shows example of operations during a power-on reset.

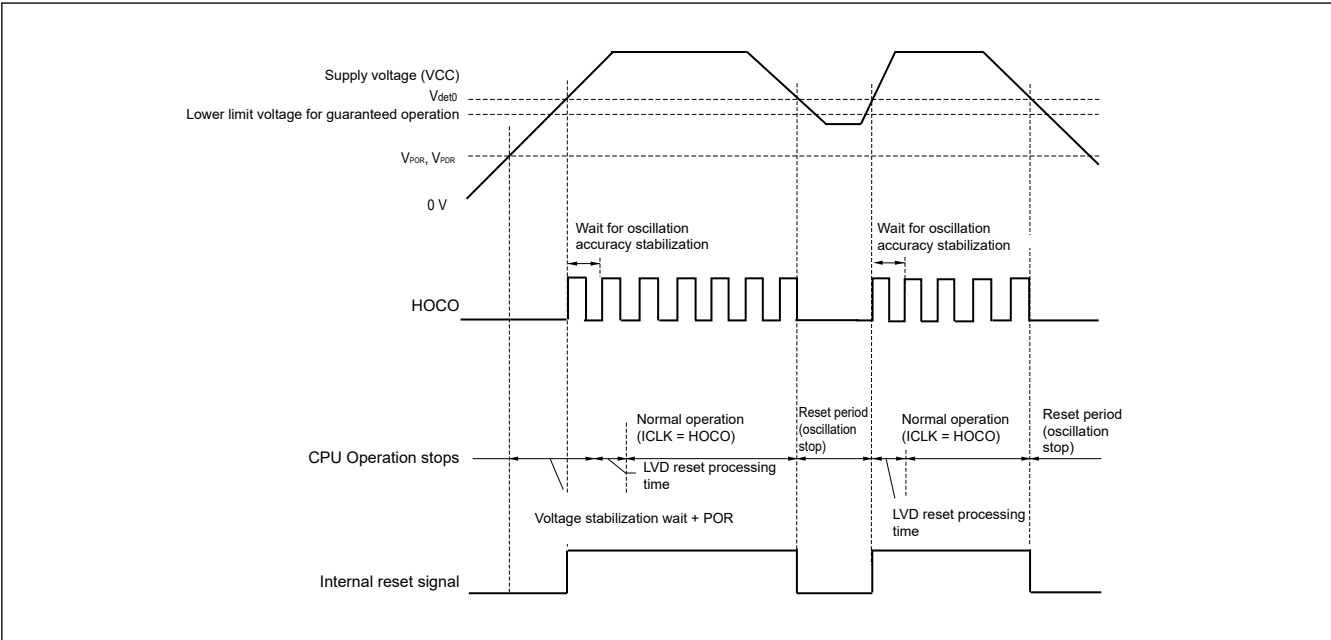


Figure 5.1 Example of operations during a power-on reset

5.3.3 Voltage Monitor Reset

The voltage monitor i ($i = 0, 1$) reset is an internal reset generated by the voltage monitor i circuit. If the Voltage Detection 0 Circuit Start (LVDAS) bit in the Option Function Select Register 1 (OFS1) is 0 (voltage monitor 0 reset is enabled after a reset) and VCC falls below V_{det0} , the RESF.LVIRF flag becomes 1 and the voltage detection circuit generates voltage monitor 0 reset. Clear the OFS1.LVDAS bit to 0 if the voltage monitor 0 reset is to be used. After VCC exceeds V_{det0} and the voltage monitor 0 reset time (t_{LVD0}) elapses, the internal reset is canceled and the CPU starts the reset exception handling.

When the Enabling Operation of LVD1 bit (LVD1EN) is set to 1 (enabling generation of a reset or interrupt by the voltage detection circuit) and the LVD1 Operation Mode Select bit (LVD1SEL) is set to 1 (selecting generation of a reset in response to detection of a low voltage) in Voltage Monitor 1 Circuit Control register (LVD1CR), the RESF.LVIRF flag is set to 1 and the voltage detection circuit generates a voltage monitor 1 reset if VCC falls to or below V_{det1} ^{*1}.

If operation of LVD1 is enabled while VCC is lower than the voltage detection level (V_{det1}), it generates an internal reset at the time its operation is enabled.

Detection levels V_{det1} can be changed in the LVD1CR.LVD1V[4:0] bit.

Note 1. The LVD1MKR.MK bit should be 0. (If MK = 1, reset and interrupt generation by LVD1 are masked.)

Figure 5.2 shows example of operation during voltage monitor 1 reset. For details on the voltage monitor 1 reset, see section 7, Low Voltage Detection (LVD).

5.3.2 开机复位

上电复位 (POR) 是由上电复位电路产生的内部复位。在以下情况下会产生上电复位。

1. 如果供电时 RES 引脚处于高电平状态
2. 如果当 VCC 低于 V_{POR} 时, RES 引脚处于高电平状态

当 VCC 超过 V_{POR} 且指定的上电复位时间 (t_{POR}) 到期后, CPU 将开始复位异常处理。
上电复位时间是外部电源和 MCU 电路的稳定期。

上电复位后, PORSR 中的 PORF 标志被清除为 0。当 VCC 低于 V_{POR} 时, 就会发生上电复位状态。

图 5.1 显示了上电复位期间的操作示例。

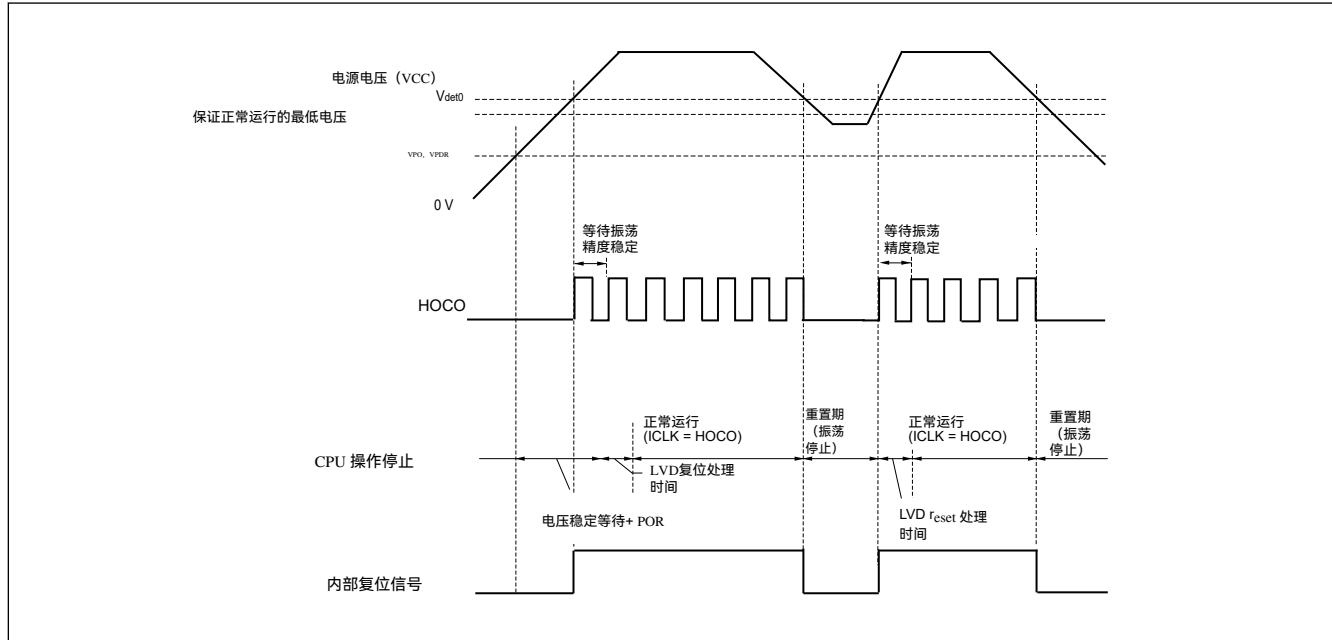


图 5.1 上电复位期间的操作示例

5.3.3 电压监控器复位

电压监视器 i ($i = 0, 1$) 复位是由电压监视器 i 电路产生的内部复位。如果选项功能选择寄存器 1 (OFS1) 中的电压检测 0 电路启动 (LVDAS) 位为 0 (复位后启用电压监视器 0 复位), 且 VCC 低于 V_{det0} , 则 RESF.LVIRF 标志变为 1, 电压检测电路产生电压监视器 0 复位。如果需要使用电压监视器 0 复位, 请将 OFS1.LVDAS 位清除为 0。当 VCC 超过 V_{det0} 且电压监视器 0 复位时间 (t_{LVD0}) 到期后, 内部复位将被取消, CPU 开始执行复位异常处理。

当电压监视器 1 电路控制寄存器 (LVD1CR) 中的 LVD1 使能操作位 (LVD1EN) 设置为 1 (使能电压检测电路生成复位或中断) 且 LVD1 操作模式选择位 (LVD1SEL) 设置为 1 (选择响应低电压检测而生成复位) 时, RESF.LVIRF 标志设置为 1, 并且如果 VCC 下降到或低于 V_{det1} ^{*1}, 电压检测电路将生成电压监视器 1 复位。

如果在 VCC 低于电压检测电平 (V_{det1}) 时启用 LVD1 的操作, 则会在其操作启用时产生内部复位。

检测级别 V_{det1} 可在 LVD1CR.LVD1V[4:0] 位中进行更改。

注1: LVD1MKR.MK 位应为 0。(如果 MK = 1, 则 LVD1 的复位和中断生成将被屏蔽。)

图 5.2 显示了电压监视器 1 复位期间的操作示例。有关电压监视器 1 复位的详细信息, 请参阅第 7 节“低电压检测 (LVD)”。

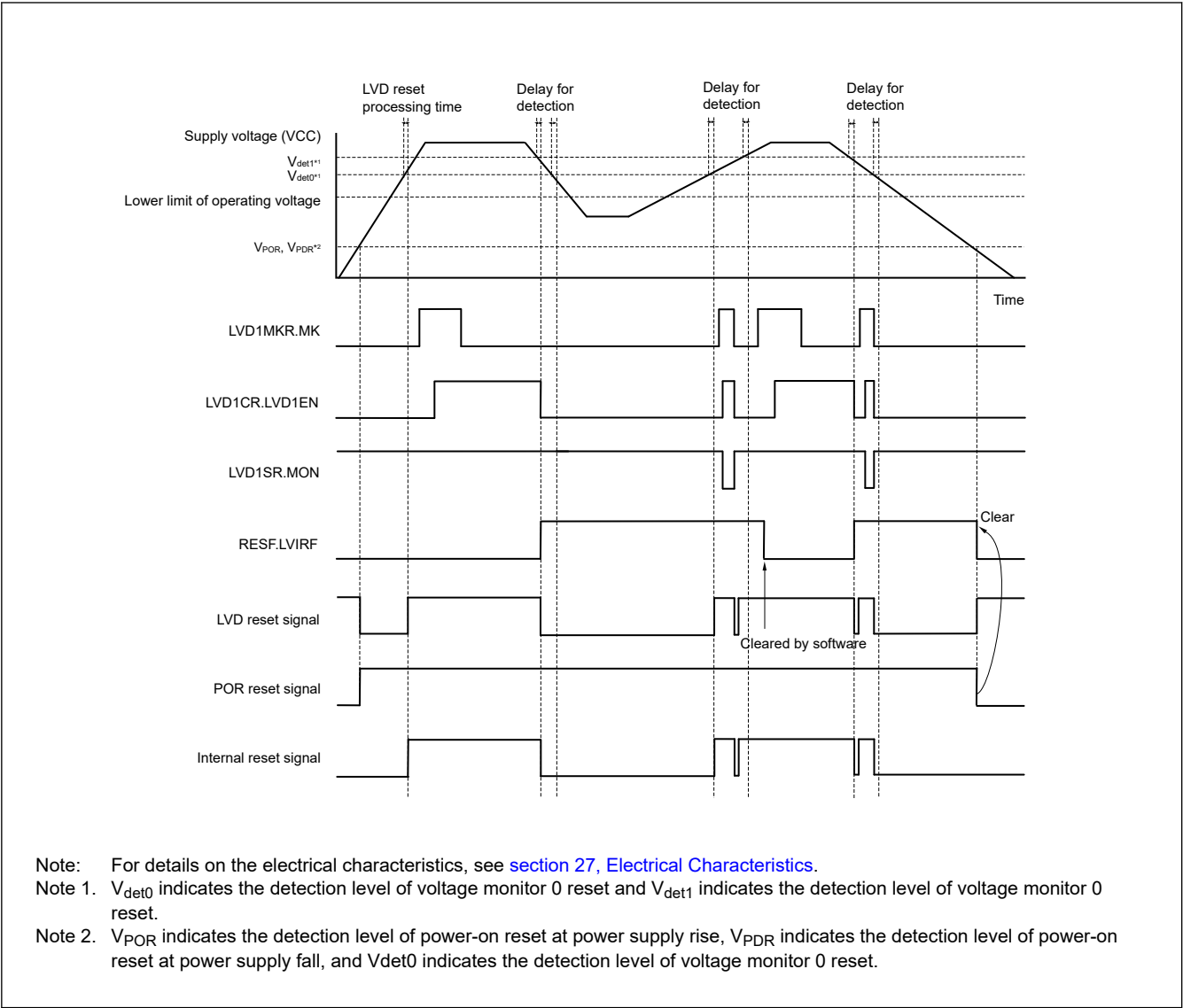


Figure 5.2 Example of operation during voltage monitor 1 reset

5.3.4 Independent Watchdog Timer Reset

The independent watchdog timer reset is an internal reset generated from the Independent Watchdog Timer (IWDG). Output of the reset from the IWDG can be selected in the Option Function Select Register 0 (OFS0).

When output of the independent watchdog timer reset is selected, the reset is generated if the IWDG underflows, or if data is written when refresh operation is disabled. When the internal reset time (t_{RESW2}) elapses after the independent watchdog timer reset is generated, the internal reset is canceled and the CPU starts the reset exception handling.

For details on the independent watchdog timer reset, see [section 18, Independent Watchdog Timer \(IWDG\)](#).

5.3.5 Software Reset

The software reset is an internal reset generated by a software setting of the SYSRESETREQ bit in the AIRCR register in the Arm core. When the SYSRESETREQ bit is set to 1, a software reset is generated. When the internal reset time (t_{RESW2}) elapses after the software reset is generated, the internal reset is canceled and the CPU starts the reset exception handling.

For details on the SYSRESETREQ bit, see the *ARM® Cortex®-M23 Technical Reference Manual*.

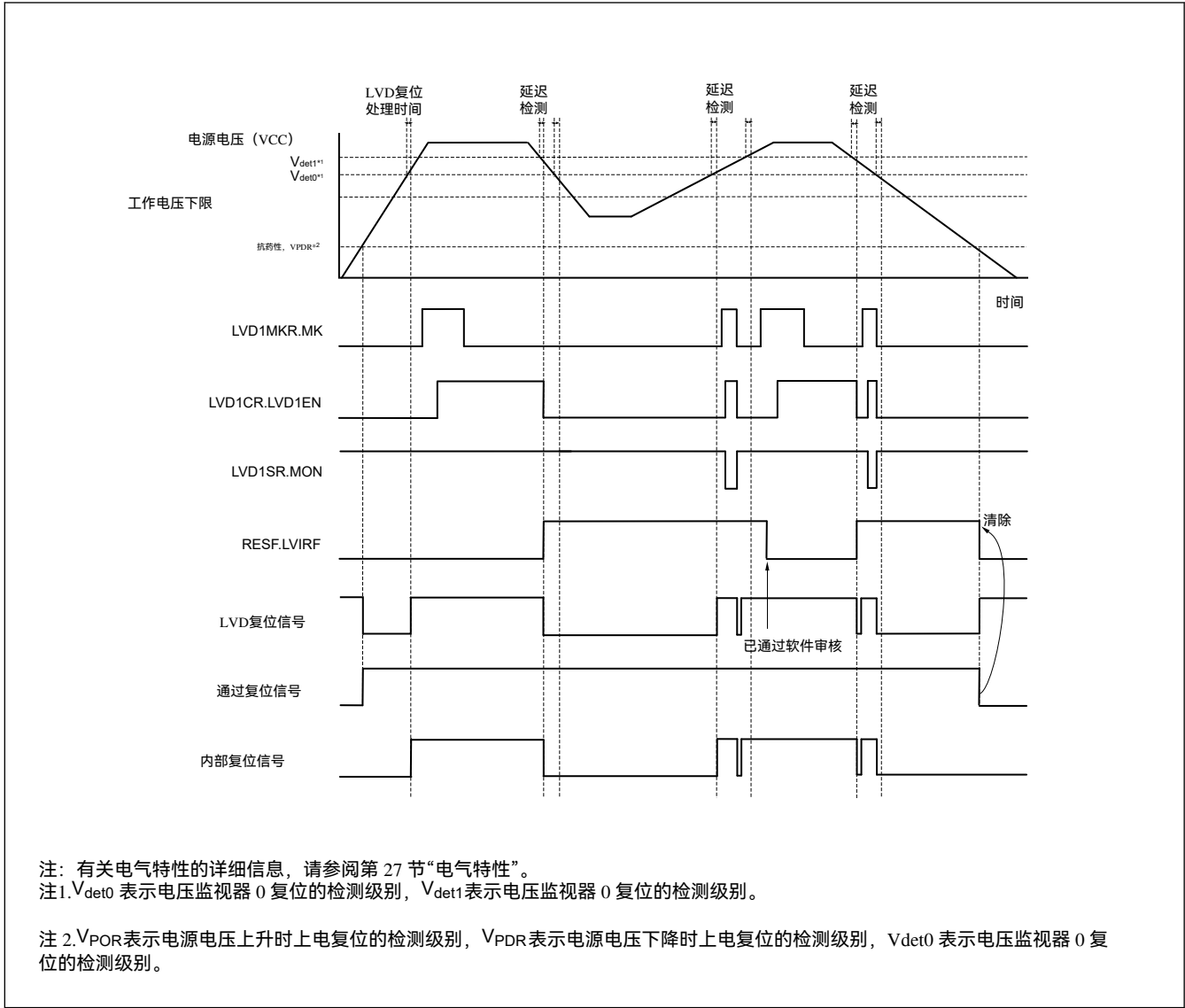


图 5.2 电压监控器 1 复位期间的操作示例

5.3.4 独立看门狗定时器复位

独立看门狗定时器复位信号是由独立看门狗定时器 (IWDG) 产生的内部复位信号。IWDG 的复位输出可在选项功能选择寄存器 0 (OFS0) 中进行选择。

当独立看门狗定时器复位输出被选中时, 若独立看门狗定时器发生下溢, 或在刷新操作被禁用时写入数据, 则会触发复位。当独立看门狗定时器复位信号生成后, 经过内部复位时间 (t_{RESW2}) 后, 内部复位信号将被取消, CPU 开始执行复位异常处理。

有关独立看门狗定时器复位的详细信息, 请参阅第 18 节“独立看门狗定时器 (IWDG)”。

5.3.5 软件重置

软件复位是由软件设置 Arm 内核 AIRCR 寄存器中的 SYSRESETREQ 位而产生的内部复位。当 SYSRESETREQ 位设置为 1 时, 就会产生软件复位。当软件复位产生后经过内部复位时间 (t_{RESW2}) 时, 内部复位将被取消, CPU 开始执行复位异常处理。

有关 SYSRESETREQ 位的详细信息, 请参阅 *ARM® Cortex®-M23 技术参考手册*。

6. Option-Setting Memory

6.1 Overview

The option-setting memory determines the state of the MCU after a reset. The Option-setting memory is allocated to the configuration setting area and the code flash area of the flash memory. The available methods of setting are different for the two areas.

Figure 6.1 shows the option-setting memory area.

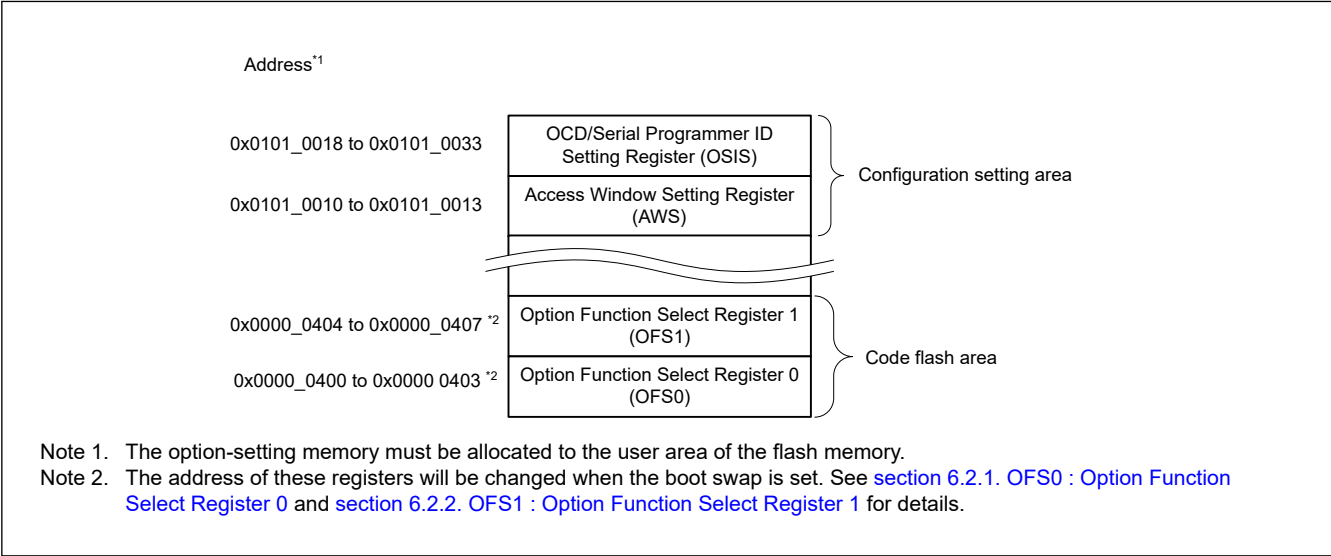
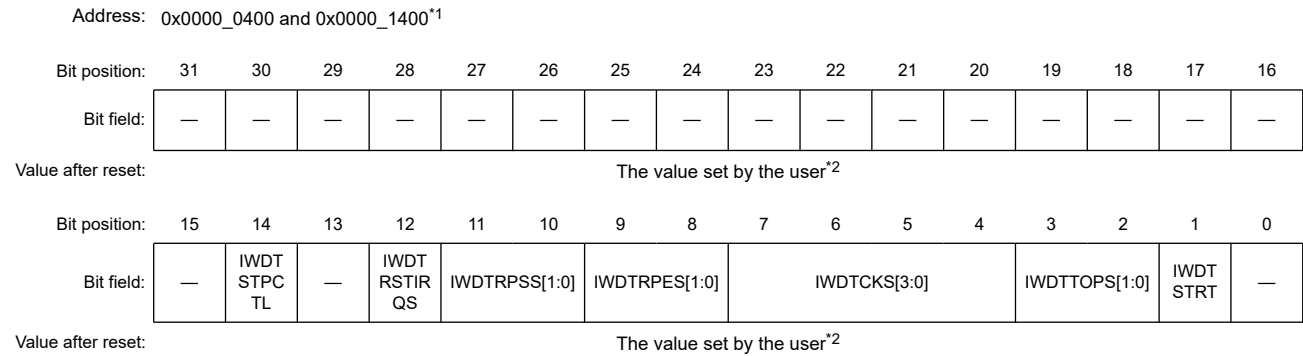


Figure 6.1 Option-setting memory area

6.2 Register Descriptions

6.2.1 OFS0 : Option Function Select Register 0



Bit	Symbol	Function	R/W
0	—	When read, this bit returns the written value.	R
1	IWDSTRT	IWDT Start Mode Select 0: Automatically activate IWDT after a reset (auto start mode) 1: Disable IWDT after a reset	R
3:2	IWDTTOPS[1:0]	IWDT Timeout Period Select 0 0: 128 cycles (0x007F) 0 1: 512 cycles (0x01FF) 1 0: 1024 cycles (0x03FF) 1 1: 2048 cycles (0x07FF)	R

6.选项设置内存

6.1 概述

选项设置存储器决定了MCU复位后的状态。选项设置存储器分配给闪存的配置设置区和代码闪存区。这两个区域的设置方法不同。

图 6.1 显示了选项设置存储区域。

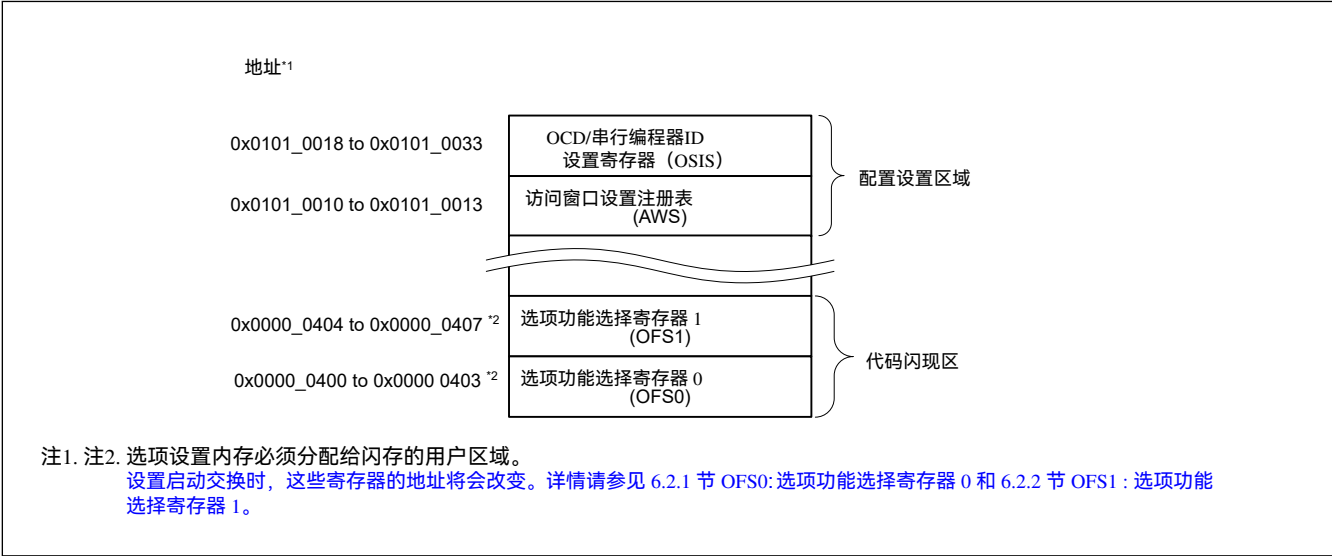
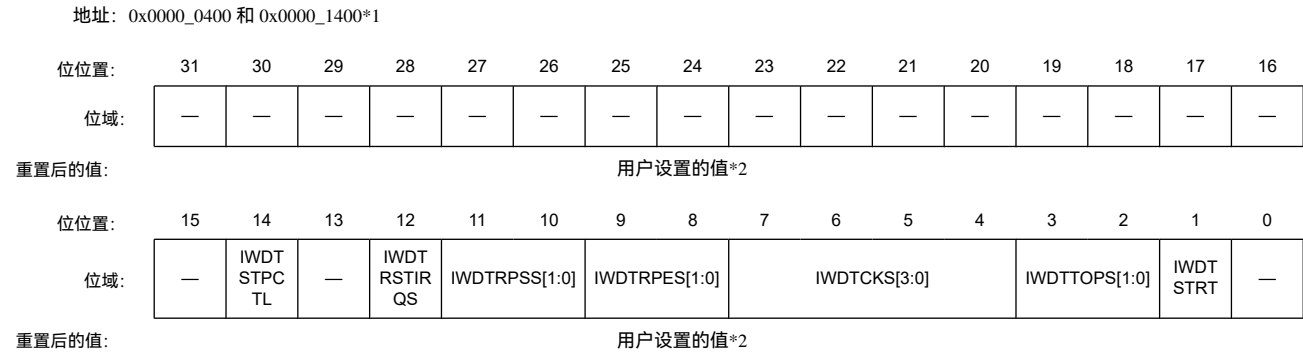


图 6.1 选项设置存储区

6.2 寄存器描述

6.2.1OFS0：选项功能选择寄存器 0



少量	象征	功能	R/W
0	—	读取此位时，返回写入的值。	R
1	IWDSTRT	IWDT 启动模式选择 0: 重置后自动激活 IWDT（自动启动模式） 1: 重置后禁用 IWDT	R
3:2	IWDTTOPS[1:0]	IWDT 超时周期选择 0 0: 128 个周期 (0x007F) 0 1: 512 个周期 (0x01FF) 1 0: 1024 个周期 (0x03FF) 1 1: 2048 个周期 (0x07FF)	R

Bit	Symbol	Function	R/W
7:4	IWDTCKS[3:0]	IWDT Clock Frequency Division Ratio Select 0x0: × 1 0x2: × 1/16 0x3: × 1/32 0x4: × 1/64 0xF: × 1/128 0x5: × 1/256 Others: Setting prohibited	R
9:8	IWDRPES[1:0]	IWDT Window End Position Select 0 0: 75% 0 1: 50% 1 0: 25% 1 1: 0% (no window end position setting)	R
11:10	IWDRPSS[1:0]	IWDT Window Start Position Select 0 0: 25% 0 1: 50% 1 0: 75% 1 1: 100% (no window start position setting)	R
12	IWDRSTIRQS	IWDT Reset Interrupt Request Select 0: Non-maskable interrupt or Maskable interrupt 1: Reset	R
13	—	When read, this bit returns the written value.	R
14	IWDTSTPCTL	IWDT Stop Control 0: Continue counting 1: Stop counting when in Sleep, Snooze, or Software Standby mode	R
31:15	—	When read, these bits return the written value.	R

Note 1. When the boot swap is set, the address of this register changes. Therefore, set 0x0000_1400 and 0x0000_0400 to the same value if boot swap is used.
Note 2. The value in a blank product is 0xFFFFFFFF. It can be set with a flash writer or self-programming.

IWDTSTRT bit (IWDT Start Mode Select)

The IWDTSTRT bit selects the mode in which the IWDT is activated after a reset (stopped state or activated state).

IWDTTOPS[1:0] bits (IWDT Timeout Period Select)

The IWDTTOPS[1:0] bits specify the timeout period, that is, the time it takes for the down counter to underflow, as 128, 512, 1024, or 2048 cycles of the frequency-divided clock set in the IWDTCCKS[3:0] bits. The time it takes for the counter to underflow after a refresh operation is determined by the combination of the IWDTCCKS[3:0] and IWDTTOPS[1:0] bits.

For details, see [section 18, Independent Watchdog Timer \(IWDT\)](#).

IWDTCCKS[3:0] bits (IWDT Clock Frequency Division Ratio Select)

The IWDTCCKS[3:0] bits specify the division ratio of the prescaler for dividing the frequency of the clock for the IWDT as 1/1, 1/16, 1/32, 1/64, 1/128, and 1/256. Using this setting combined with the IWDTTOPS[1:0] bits setting, the IWDT counting period can be set from 128 to 524288 IWDT clock cycles.

For details, see [section 18, Independent Watchdog Timer \(IWDT\)](#).

IWDRPES[1:0] bits (IWDT Window End Position Select)

The IWDRPES[1:0] bits specify the position where the window for the down counter ends as 0%, 25%, 50%, or 75% of the count value. The value of the window end position must be smaller than the value of the window start position, otherwise only the value for the window start position is valid.

The counter values associated with the settings for the start and end positions of the window in the IWDRPSS[1:0] and IWDRPES[1:0] bits vary with the setting in the IWDTTOPS[1:0] bits.

For details, see [section 18, Independent Watchdog Timer \(IWDT\)](#).

少量	象征	功能	R/W
7:4	IWDTCCKS[3:0]	IWDT 时钟频率分频比选择 0x0: × 1 0x2: × 1/16 0x3: × 1/32 0x4: × 1/64 0xF: × 1/128 0x5: × 1/256 其他: 设置禁止	R
9:8	IWDRPES[1:0]	IWDT 窗口结束位置选择 0 0:75%0 1:50%1 0:25% 1 1:0% (未设置窗口结束位置)	R
11:10	IWDRPSS[1:0]	IWDT 窗口起始位置选择 0 0:25% 1 :50% 1 0: 75% 1 1:100% (未设置窗口起始位置)	R
12	IWDRSTIRQS	IWDT 复位中断请求选择 0: 不可屏蔽中断或可屏蔽中断 1: 重置	R
13	—	读取此位时, 返回写入的值。	R
14	IWDTSTPCTL	IWDT 停止控制 0: 继续计数 1: 在睡眠、小睡或软件待机模式下停止计数	R
31:15	—	读取这些位时, 会返回写入的值。	R

注1: 启用引导交换后, 此寄存器的地址会发生变化。因此, 如果启用引导交换, 则应将0x0000_1400和0x0000_0400设置为相同的值。

注2: 空白产品的值为0xFFFFFFFF。可通过闪存写入器或自行编程进行设置。

IWDTSTRT 位 (IWDT 启动模式选择)

IWDTSTRT 位选择 IWDT 在复位后激活的模式 (停止状态或激活状态)。

IWDTTOPS[1:0] 位 (IWDT 超时周期选择)

IWDTTOPS[1:0] 位指定超时周期, 即向下计数器下溢所需的时间, 其值为 IWDTCCKS[3:0] 位设置的分频时钟的 128、512、1024 或 2048 个周期。刷新操作后计数器下溢所需的时间由 IWDTCCKS[3:0] 和 IWDTTOPS[1:0] 位的组合决定。

有关详细信息, 请参阅第 18 节“独立看门狗定时器 (IWDT)”。

IWDTCCKS[3:0] 位 (IWDT 时钟频率分频比选择)

IWDTCCKS[3:0] 位指定预分频器的分频比, 用于将 IWDT 的时钟频率分频, 分别为 1/1、1/16、1/32、1/64、1/128 和 1/256。结合 IWDTTOPS[1:0] 位设置, 可以将 IWDT 计数周期设置为 128 到 524288 个 IWDT 时钟周期。

有关详细信息, 请参阅第 18 节“独立看门狗定时器 (IWDT)”。

IWDRPES[1:0] 位 (IWDT 窗口结束位置选择)

IWDRPES 的[1:0] 位指定递减计数器窗口的结束位置, 即计数值的0%、25%、50%或75%。窗口结束位置的值必须小于窗口起始位置的值, 否则只有窗口起始位置的值有效。

与 IWDTTOPS[1:0] 和 IWDRPES[1:0] 位中窗口起始位置和结束位置的设置相关的计数器值, 会随着 IWDTTOPS[1:0] 位中的设置而变化。

有关详细信息, 请参阅第 18 节“独立看门狗定时器 (IWDT)”。

IWDTRPSS[1:0] bits (IWDT Window Start Position Select)

The IWDTRPSS[1:0] bits specify the position where the window for the down counter starts as 25%, 50%, 75%, or 100% of the counted value. The point at which counting starts is 100% and the point at which an underflow occurs is 0%. The interval between the window starts and ends positions becomes the period in which a refresh is possible. Refresh is not possible outside this period.

For details, see [section 18, Independent Watchdog Timer \(IWDT\)](#).

IWDRSTIRQS bit (IWDT Reset Interrupt Request Select)

The IWDRSTIRQS bit selects the operation on an underflow of the down counter or generation of a refresh error. The operation is selectable to an independent watchdog timer reset, a non-maskable interrupt request, or an interrupt request.

For details, see [section 18, Independent Watchdog Timer \(IWDT\)](#).

IWDTSTPCTL bit (IWDT Stop Control)

The IWDTSTPCTL bit specifies whether to stop counting when entering Sleep mode, Snooze mode, or Software Standby mode.

[Table 6.1](#) shows the count stop control by the IWDTSTPCTL bit.

Table 6.1 Count stop control by the IWDTSTPCTL bit

IWDTSTPCTL	Mode	Counting of IWDT
0	Sleep / snooze/ software standby mode	Continue counting
1	Sleep / snooze / software standby mode	Stop counting

For details, see [section 18, Independent Watchdog Timer \(IWDT\)](#).

6.2.2 OFS1 : Option Function Select Register 1

Address: 0x0000_0404 and 0x0000_1404*1																
Bit position:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Bit field:	—	—	—	FRPDI S	FRPE[5:0]						FRPS[5:0]					
Value after reset: The value set by the user*2																
Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	PORT SELB	HOCOFRQ1[2:0]			—	—	—	—	—	—	VDSEL0[2:0]			LVDA S	—	—
Value after reset: The value set by the user*2																

Bit	Symbol	Function	R/W
1:0	—	When read, these bits return the written value.	R
2	LVDA S	Voltage Detection 0 Circuit Start 0: Enable voltage monitor 0 reset after a reset 1: Disable voltage monitor 0 reset after a reset	R
5:3	VDSEL0[2:0]	Voltage Detection 0 Level Select*3 0 1 0: V _{det0_0} 0 1 1: V _{det0_1} 1 0 0: V _{det0_2} 1 0 1: V _{det0_3} 1 1 0: V _{det0_4} 1 1 1: V _{det0_5} Others: Setting prohibited	R
11:6	—	When read, these bits return the written value.	R

IWDTRPSS[1:0] 位 (IWDT 窗口起始位置选择)

IWDTRPSS 的[1:0] 位指定递减计数器窗口的起始位置，即计数值的25%, 50%, 75%或100%。计数开始的位置是100%，下溢发生的位置是0%。窗口起始位置和结束位置之间的间隔即为刷新周期。在此周期之外，无法进行刷新。

有关详细信息，请参阅第 18 节“独立看门狗定时器 (IWDT)”。

IWDRSTIRQS 位 (IWDT 复位中断请求选择)

IWDRSTIRQS 位用于选择计数器下溢或刷新错误发生时的操作。该操作可选择为独立看门狗定时器复位、不可屏蔽中断请求或中断请求。

有关详细信息，请参阅第 18 节“独立看门狗定时器 (IWDT)”。

IWDTSTPCTL 位 (IWDT 停止控制)

IWDTSTPCTL 位指定进入睡眠模式、贪睡模式或软件待机模式时是否停止计数。

表 6.1 显示了 IWDTSTPCTL 位对计数停止的控制。

表 6.1 通过 IWDTSTPCTL 位进行计数停止控制

IWDTSTPCTL	模式	IWDT计数
0	睡眠/贪睡/软件待机模式	继续计数
1	睡眠/贪睡/软件待机模式	停止计数

有关详细信息，请参阅第 18 节“独立看门狗定时器 (IWDT)”。

6.2.2 OFS1: 选项功能选择寄存器 1

地址: 0x0000_0404 和 0x0000_1404*1

位位置:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域:	—	—	—	FRPDI S	FRPE[5:0]						FRPS[5:0]					

重置后的值: The value set by the user*2

位位置:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域:	PORT SELB	HOCOFRQ1[2:0]			—	—	—	—	—	—	VDSEL0[2:0]			LVDA S	—	—

重置后的值: 用户设置的值*2

少量	象征	功能	R/W
1:0	—	读取这些位时，会返回写入的值。	R
2	LVDA S	电压检测 0 电路启动 0: 复位后启用电压监视器 0 复位 1: 重置后禁用电压监视器 0 重置	R
5:3	VDSEL0[2:0]	电压检测 0 级选择*3 0 1 0: V _{det0_0} 0 1 1: V _{det0_1} 1 0 0: V _{det0_2} 1 0 1: V _{det0_3} 1 1 0: V _{det0_4} 1 1 1: V _{det0_5} 其他: 禁止设置	R
11:6	—	读取这些位时，会返回写入的值。	R

Bit	Symbol	Function	R/W
14:12	HOCOFRQ1[2:0]	HOCO Frequency Setting 1 0 1 0: 32 MHz 1 1 1: 32 MHz Others: Setting prohibited	R
15	PORTSELB	P206/RES Terminal Selection 0: Port (P206) 1: RES input (internal pull-up register is always active.)	R
21:16	FRPS[5:0]	Flash Read Protection Starting Address FRPS[5:0] specifies the bit[16:11] of the starting address of a protected region. The bit[31:17] and bit[10:0] of the starting address are filled with 0s. The value range of FRPS[5:0] is from 0x01 to 0x3F, excluding reserved areas.	R
27:22	FRPE[5:0]	Flash Read Protection Ending Address FRPE[5:0] specifies the bit[16:11] of the ending address of a protected region. The bit[31:17] of the ending address are filled with 0s and the bit[10:0] of the ending address are filled with 1s. The value range of FRPE[5:0] is from 0x01 to 0x3F, excluding reserved areas.	R
28	FRPDIS	Flash Read Protection Disable 0: Enable flash read protection 1: Disable flash read protection	R
31:29	—	When read, these bits return the written value.	R

Note 1. When the boot swap is set, the address of this register changes. Therefore, set 0x0000_1404 and 0x0000_0404 to the same value if boot swap is used.
Note 2. The value in a blank product is 0xFFFFFFFF. It can be set with a flash writer or self-programming.
Note 3. See [section 27, Electrical Characteristics](#) for the voltage levels to be detected. Set to 010b if LVD0 is not used.

LVDAS bit (Voltage Detection 0 Circuit Start)

The LVDAS bit selects whether the voltage monitor 0 reset is enabled or disabled after a reset.

VDSEL0[2:0] bits (Voltage Detection 0 Level Select)

The VDSEL0[2:0] bits select the voltage detection level of the voltage detection 0 circuit.

HOCOFRQ1[2:0] bits (HOCO Frequency Setting 1)

The HOCOFRQ1[2:0] bits select the HOCO frequency after reset. Set either 010b or 111b for the HOCOFRQ1[2:0] setting.

PORTSELB bit (P206/RES Terminal Selection)

The PORTSELB bit defines whether this port operates as port (P206) or reset.

FRPS[5:0] bits (Flash Read Protection Starting Address)

FRPE[5:0] bits (Flash Read Protection Ending Address)

The FRPS[5:0] and FRPE[5:0] bits specify the starting and ending address of a protected region of the FRP. [Figure 6.2](#) shows the starting and ending address of a protected region. Both the starting and ending address are included in a protected region (starting address ≤ protected region ≤ ending address). When the FRP is enabled, the memory space defined by the FRPS[5:0] and FRPE[5:0] bits is accessible only by instruction fetch. Since a protected region is not readable, it is prohibited that a protected region includes a vector table. Therefore, do not set FRPS[5:0] to 0x00 when using the FRP. Setting FRPS[5:0] to 0x00 causes a protected region to include the initial vector table.

For details of the FRP, see [section 13, Flash Read Protection \(FRP\)](#).

FRPDIS bit (Flash Read Protection Disable)

The FRPDIS bit enables or disables the FRP. To enable the FRP, FRPDIS must be set to 0. To disable the FRP, FRPDIS must be set to 1.

For details of the FRP, see [section 13, Flash Read Protection \(FRP\)](#).

少量	象征	功能	R/W
14:12	HOCOFRQ1[2:0]	HOCO 频率设置 1 0 1 0: 32 MHz 1 1 1: 32 MHz 其他: 禁止设置	R
15	波特塞尔布	P206/RES 端子选择 0: 端口 (P206) 1: RES 输入 (内部上拉寄存器始终处于激活状态。)	R
21:16	FRPS[5:0]	闪存读取保护起始地址 FRPS[5:0] 指定受保护区域起始地址的位[16:11]。 起始地址的第 17 位[31和第 0 位] 以及第 0 位[10和第 4 位] 填充为 0。 FRPS[5:0] 的值范围为0x01 至0x3F, , 不包括保留区域。	R
27:22	FRPE[5:0]	闪存读取保护结束地址 FRPE[5:0] 指定受保护区域的结束地址的位[16:11]。 结束地址的位[31:17] 填充为 0, 结束地址的位[10:0] 填充为 1。 FRPE[5:0] 的值范围为0x01 至0x3F, , 不包括保留区域。	R
28	FRPDIS	禁用闪存读取保护 0: 启用闪存读取保护 1: 禁用闪存读取保护	R
31:29	—	读取这些位时, 会返回写入的值。	R

注1: 启用引导交换后, 此寄存器的地址会发生变化。因此, 如果启用引导交换, 则应将0x0000_1404和0x0000_0404设置为相同的值。

注2: 空白产品的值为0xFFFFFFFF。可通过闪存写入器或自行编程进行设置。
注3: 有关要检测的电压等级, 请参见第27节“电气特性”。如果未使用LVD0, 则设置为010b。

LVDAS 位 (电压检测 0 电路启动)

LVDAS 位用于选择复位后是否启用或禁用电压监视器 0 复位。

VDSEL0[2:0] 位 (电压检测 0 电平选择)

VDSEL0[2:0] 位选择电压检测 0 电路的电压检测级别。

HOCOFRQ1[2:0] 位 (HOCO 频率设置 1)

HOCOFRQ1[2:0] 位用于选择复位后的 HOCO 频率。HOCOFRQ1[2:0] 设置应为 010b 或 111b。

PORTSELB 位 (P206/RES 终端选择)

PORTSELB 位定义此端口是作为端口 (P206) 还是复位端口运行。

FRPS[5:0] 位 (闪存读取保护起始地址)

FRPE[5:0] 位 (闪存读取保护结束地址)

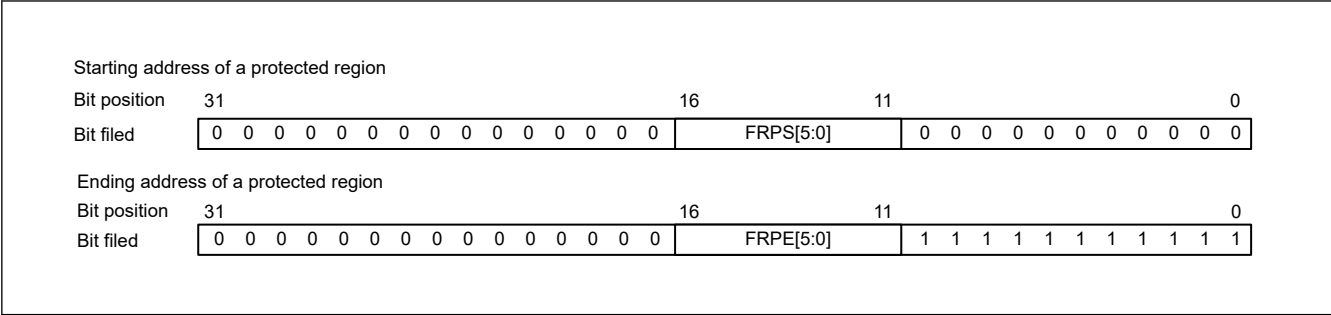
FRPS[5:0] 和 FRPE[5:0] 位分别指定 FRP 保护区域的起始地址和结束地址。图 6.2 显示了保护区域的起始地址和结束地址。起始地址和结束地址均包含在保护区域内 (起始地址≤保护区域≤结束地址)。启用 FRP 后, 由 FRPS[5:0] 和 FRPE[5:0] 位定义的内存储器空间只能通过指令提取访问。由于保护区域不可读, 因此禁止在保护区域内包含向量表。因此, 在使用 FRP 时, 不要将 FRPS[5:0] 设置为0x00 。将 FRPS[5:0] 设置为0x00会使受保护区域包含初始向量表。

[有关 FRP 的详细信息, 请参阅第 13 节“闪存读取保护 \(FRP\)”。](#)

FRPDIS 位 (闪存读取保护禁用)

FRPDIS 位用于启用或禁用 FRP。要启用 FRP, FRPDIS 必须设置为 0。要禁用 FRP, FRPDIS 必须设置为 1。

[有关 FRP 的详细信息, 请参阅第 13 节“闪存读取保护 \(FRP\)”。](#)



The access window is specified in both the FAWS[10:0] bits and the FAWE[10:0] bits. The settings for the FAWS[10:0] and FAWE[10:0] bits are as follows:

FAWE[10:0] = FAWS[10:0]: The P/E command is allowed to execute in the full code flash area.

FAWE[10:0] > FAWS[10:0]: The P/E command is only allowed to execute in the window from the block pointed to by the FAWS[10:0] bits to the block one lower than the block pointed to by the FAWE[10:0] bits.

FAWE[10:0] < FAWS[10:0]: The P/E command is not allowed to execute in the code flash area.

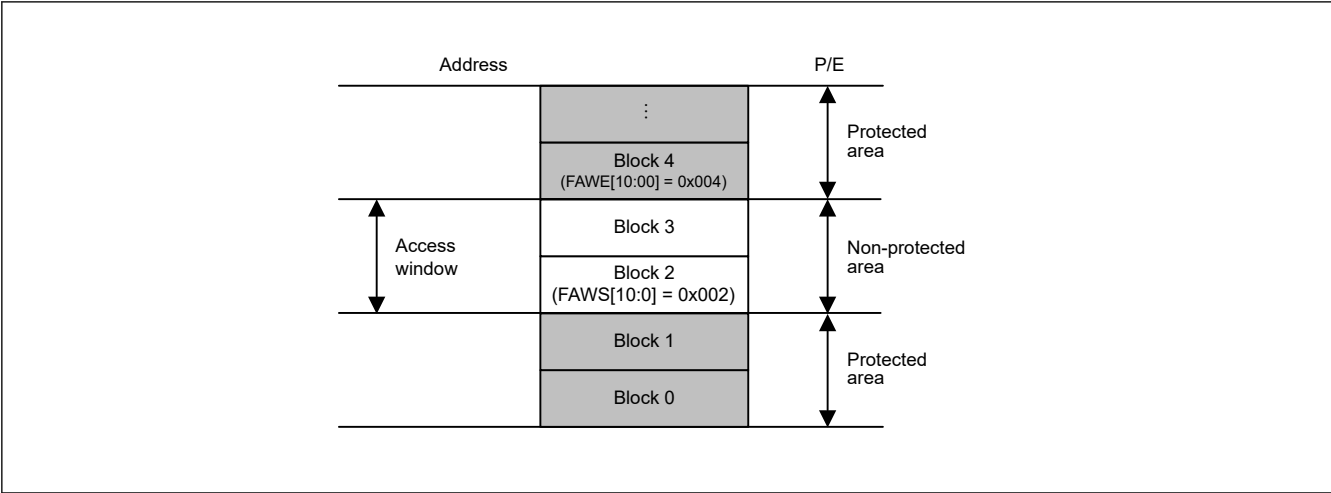


Figure 6.3 Access window overview

6.2.4 OSIS : OCD/Serial programmer ID Setting Register

The OSIS register stores the ID for ID code protection of the OCD emulator/serial programmer. When connecting the OCD emulator/serial programmer, write values so that the MCU can determine whether to permit the connection. Use this register to check whether a code transmitted from the OCD emulator/serial programmer matches the ID code in the option-setting memory. When the ID code matches, connection with the OCD emulator/serial programmer is permitted, if not, connection with the OCD emulator/serial programmer is not possible. The OSIS register must be set in 32-bit words.

Address: 0x0101_0018, 0x0101_0020, 0x0101_0028, 0x0101_0030

Bit position: 31 0

Bit field:

Value after reset: The value set by the user

These fields hold the ID for use in ID authentication for the OCD emulator/serial programmer.

ID code bits [127] and [126] on the 32-bit word at the address 0x0101_0018 determine whether the ID code protection is enabled, and the authentication method. Table 6.2 shows how the ID code determines the authentication method.

访问窗口在 FAWS[10:0] 位和 FAWE[10:0] 位中均有指定。FAWS[10:0] 和 FAWE[10:0] 位如下：

FAWE[10:0] = FAWS[10:0]：允许在完整代码闪存区域中执行P/E命令。

FAWE[10:0] > FAWS[10:0]:P/E命令仅允许在 FAWS[10:0] 位指向的块到 FAWE[10:0] 位指向的块低一位的块的窗口中执行。

FAWE[10:0] < FAWS[10:0]：不允许在代码闪存区域执行P/E命令。

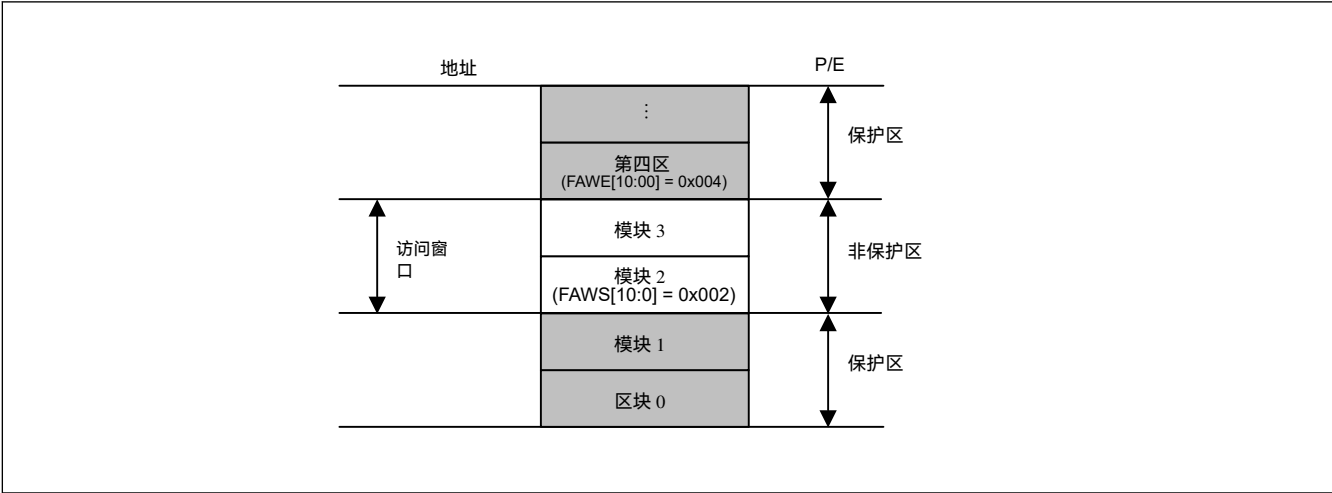


图 6.3 访问窗口概览

6.2.4 OSIS: OCD/串行编程器 ID 设置寄存器

OSIS 寄存器存储 OCD 仿真器/串行编程器的 ID 代码，用于 ID 代码保护。连接 OCD 仿真器/串行编程器时，需要写入相应的值，以便 MCU 判断是否允许连接。使用此寄存器检查从 OCD 仿真器/串行编程器发送的代码是否与选项设置存储器中的 ID 代码匹配。如果 ID 代码匹配，则允许与 OCD 仿真器/串行编程器连接；否则，不允许连接。OSIS 寄存器必须以 32 位字为单位进行设置。

地址: 0x0101_0018、0x0101_0020、0x0101_0028、0x0101_0030

位位置: 31 0

位域:

重置后的值: 用户设置的值

这些字段保存用于 OCD 模拟器/串行编程器 ID 认证的 ID。

地址0x0101_0018处的32位字上的ID码位[127]和[126]决定了是否启用ID码保护以及认证方法。表6.2展示了ID码如何确定认证方法。

Table 6.2 Specifications for ID code protection

Operating mode on boot up	ID code	State of protection	Operations on connection to programmer or on-chip debugger
On-chip debug mode (SWD boot mode)	0xFF, ..., 0xFF (all bytes are 0xFF)	Protection disabled	The ID code is not checked, the ID code always matches, and the connection to the on-chip debugger or serial programmer is permitted.
	Bit [127] = 1, bit [126] = 1, and at least one of the 16 bytes is not 0xFF	Protection enabled	Matching ID code indicates that authentication is complete and connection to the on-chip debugger or serial programmer is permitted. Mismatching ID code indicates transition to the ID code protection wait state. When the ID code sent from the on-chip debugger or serial programmer is ALeRASE in ASCII code (0x414C_6552_4153_45FF_FFFF_FFFF_FFFF_FFFF), the content of the user flash area is erased and all bits in the OSIS register are 1. However, when the AWS.FSPR bit is 0, the content of the user flash area is not erased.
	Bit [127] = 1 and bit [126] = 0	Protection enabled	Matching ID code indicates that authentication is complete and connection to the on-chip debugger or serial programmer is permitted. Mismatching ID code indicates transition to the ID code protection wait state.
	Bit [127] = 0	Protection enabled	The ID code is not checked, the ID code is always mismatching, the connection to the on-chip debugger or serial programmer is prohibited.

6.3 Setting Option-Setting Memory

6.3.1 Allocation of Data in Option-Setting Memory

Programming data is allocated to the addresses in the option-setting memory shown in Figure 6.1. The allocated data is used by tools such as a flash programming software or an on-chip debugger.

Note: Programming formats vary depending on the compiler. See the compiler manual for details.

6.3.2 Setting Data for Programming Option-Setting Memory

Allocating data according to the procedure described in section 6.3.1. Allocation of Data in Option-Setting Memory, alone does not actually write the data to the option-setting memory. You must also follow one of the actions described in this section.

(1) Changing the option-setting memory by self-programming

Use the programming command to write data to the code flash area. Use the configuration setting command to write data to the option-setting memory in the configuration setting area. In addition, use the startup area select function to safely update the boot program that includes the option-setting memory.

For details of the programming command, the configuration setting command, and the startup area select function, see section 25, Flash Memory.

(2) Debugging through an OCD or programming by a flash writer

This procedure depends on the tool in use, see the tool manual for details.

The MCU provides two setting procedures:

- Read the data allocated as described in section 6.3.1. Allocation of Data in Option-Setting Memory, from an object file or Motorola S-format file generated by the compiler, and write the data to the MCU
- Use the GUI interface of the tool to program the same data as allocated in section 6.3.1. Allocation of Data in Option-Setting Memory.

表 6.2 ID 代码保护规范

启动时的操作模式	ID 代码	保护状态	对连接到编程器或片上调试器的操作
片上调试模式 (SWD启动模式)	0xFF, ..., 0xFF (所有字节均为 0xFF)	保护已禁用	不检查 ID 代码, ID 代码始终匹配, 允许连接到片上调试器或串行编程器。
	位[127] = 1, 位[126] = 1, 且16个字节中至少有一个不为0xFF。	已启用保护	匹配的 ID 代码表示身份验证完成, 并且允许连接到片上调试器或串行编程器。 ID 代码不匹配表示进入 ID 代码保护等待状态。 当片上调试器或串行编程器发送的 ID 代码为 ASCII 码 ALeRASE (0x414C_6552_4153_45FF_FFFF_FFFF_FFFF_FF FF) 时, 用户闪存区域的内容将被擦除, OSIS 寄存器中的所有位均为 1。 但是, 当 AWS.FSPR 位为 0 时, 用户闪存区域的内容不会被擦除。
	位[127] = 1 且位[126] = 0	已启用保护	匹配的 ID 代码表示身份验证完成, 并且允许连接到片上调试器或串行编程器。 ID 代码不匹配表示进入 ID 代码保护等待状态。
	位[127] = 0	已启用保护	ID 代码未进行检查, ID 代码始终不匹配, 禁止连接到片上调试器或串行编程器。

6.3 设置选项-设置内存

6.3.1 选项设置内存中的数据分配

编程数据被分配到图 6.1 所示的选项设置存储器中的地址。分配的数据可供闪存编程软件或片上调试器等工具使用。

注意: 编程格式因编译器而异。详情请参阅编译器手册。

6.3.2 设置编程选项的数据 - 设置存储器

仅按照 6.3.1 节“选项设置内存中的数据分配”中描述的步骤分配数据, 并不会真正将数据写入选项设置内存。您还必须执行本节中描述的操作之一。

(1) 通过自编程更改选项设置存储器

使用编程命令将数据写入代码闪存区域。使用配置设置命令将数据写入配置设置区域中的选项设置存储器。此外, 使用启动区域选择功能安全地更新包含选项设置存储器的引导程序。

有关编程命令、配置设置命令和启动区域选择功能的详细信息, 请参阅第 25 节“闪存”。

(2) 通过 OCD 进行调试或通过闪存写入器进行编程

此操作步骤取决于所使用的工具, 详情请参阅工具手册。

MCU 提供两种设置程序:

- 按照 6.3.1 节“选项设置存储器中的数据分配”所述, 从编译器生成的目标文件或 Motorola S 格式文件中读取已分配的数据, 并将数据写入 MCU。
- 使用该工具的 GUI 界面, 对 6.3.1 节中分配的相同数据进行编程。选项设置内存中的数据分配。

6.4 Usage Notes

6.4.1 Data for Programming Reserved Areas and Reserved Bits in the Option-Setting Memory

When reserved areas and reserved bits in the option-setting memory are within the scope of programming, write 1 to all bits of reserved areas and all reserved bits. If 0 is written to these bits, normal operation cannot be guaranteed.

6.4.2 Note on FSPR Bit

The AWS.FSPR bit cannot be changed to 1 once it is set to 0. At that time, access window and startup area selection cannot be set again.

6.4 使用说明

6.4.1 选项设置存储器中保留区域和保留位的编程数据

当选项设置存储器中的保留区域和保留位处于编程范围内时，应将所有保留区域和保留位的位都写入 1。如果写入 0，则无法保证正常运行。

6.4.2 关于 FSPR 位的说明

AWS.FSPR 位一旦设置为 0，就无法再更改为 1。此时，将无法再次设置访问窗口和启动区域选择。

7. Low Voltage Detection (LVD)

7.1 Overview

The Low Voltage Detection (LVD) module monitors the voltage level input to the VCC pin. The detection level can be selected by register settings. The LVD module consists of two separate voltage level detectors (LVD0, LVD1). LVD0 and LVD1 measure the voltage level input to the VCC pin. LVD registers allow your application to configure detection of VCC changes at various voltage thresholds.

Voltage monitor registers are used to configure the LVD to trigger an interrupt, or reset when the thresholds are crossed.

Table 7.1 lists the LVD specifications. Figure 7.1 shows a block diagram of the voltage monitor 0 reset generation circuit and Figure 7.2 shows a block diagram of the voltage monitor 1 interrupt and reset circuit.

Table 7.1 LVD specifications

Parameter		Voltage monitor 0	Voltage monitor 1
Means for setting up operation		OFS1 register	Registers
Target for monitoring		VCC pin input voltage	VCC pin input voltage
Monitored voltage		V_{det0}	V_{det1}
Detected event		Voltage falls past V_{det0}	Voltage rises or falls past V_{det1}
Detection voltage		Selectable from 6 different levels in the OFS1.VDSEL0[2:0] bits	Selectable from 18 different levels in the LVD1CR.LVD1V[4:0] bits
Monitoring flag		None	LVD1SR.MON flag: Monitors whether voltage is higher or lower than V_{det1}
			LVD1SR.DET flag: V_{det1} passage detection
Process on voltage detection	Reset	Voltage monitor 0 reset Deasserts an internal reset signal on detecting $VCC \geq V_{det0}$. Generates an internal reset on detecting $VCC < V_{det0}$ and retains the reset state until $VCC \geq V_{det0}$ is detected.	Voltage monitor 1 reset Deasserts an internal reset signal on detecting $VCC \geq V_{det1}$. Generates an internal reset on detecting $VCC < V_{det1}$ and retains the reset. The reset is deasserted after specified time.
	Interrupt	No interrupt	Voltage monitor 1 interrupt
			Non-maskable or maskable interrupt selectable Interrupt request is issued when VCC crosses V_{det1}

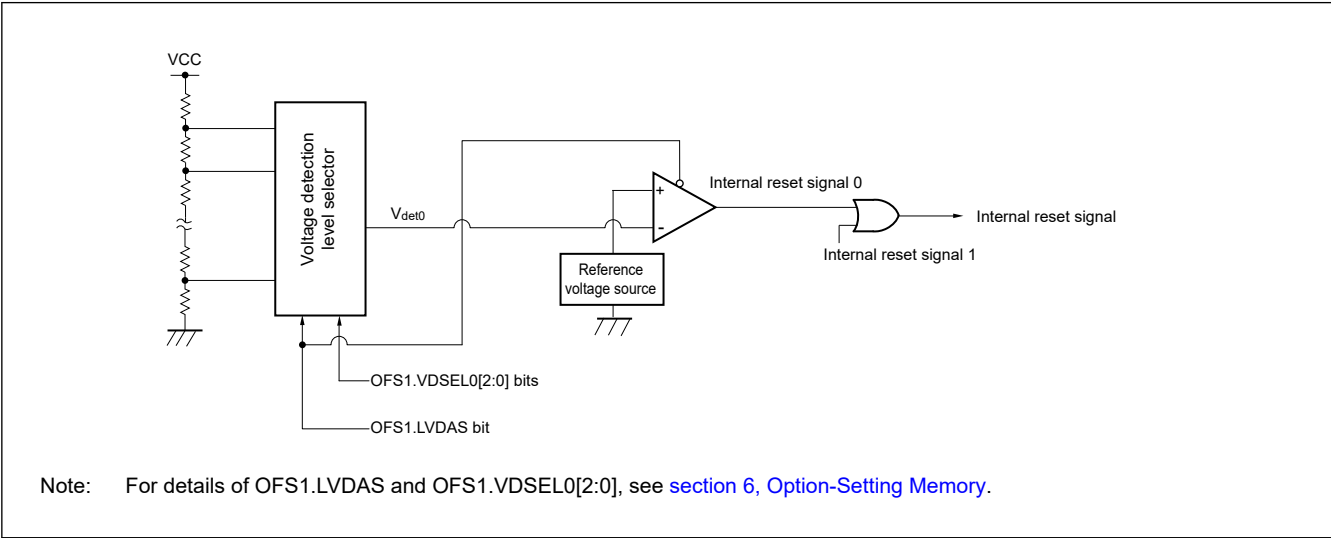


Figure 7.1 Block diagram of voltage monitor 0 reset generation circuit

7.低电压检测（LVD）

7.1 概述

低电压检测 (LVD) 模块监控输入到 VCC 引脚的电压电平。检测电平可通过寄存器设置进行选择。LVD 模块包含两个独立的电压电平检测器 (LVD0 和 LVD1)。LVD0 和 LVD1 测量输入到 VCC 引脚的电压电平。LVD 寄存器允许您的应用程序配置在不同的电压阈值下检测 VCC 变化。

电压监控寄存器用于配置低电压检测器 (LVD)，使其在超过阈值时触发中断或复位。表 7.1 列出了 LVD 的规格。图 7.1 显示了电压监控器 0 复位生成电路的框图，图 7.2 显示了电压监控器 1 中断和复位电路的框图。

表 7.1 LVD 规格

范围		电压监测器 0	电压监测器 1
建立操作的方法		OFS1 寄存器	登记册
监测目标		VCC 引脚输入电压	VCC 引脚输入电压
监测电压		V_{det0}	V_{det1}
检测到事件		电压降低至 V_{det0}	电压上升或下降超过 V_{det1}
检测电压		可从 6 个不同级别中选择 OFS1.VDSEL0[2:0] bits	可从 18 个不同级别中选择 LVD1CR.LVD1V[4:0] bits
监控标志		None	LVD1SR.MON 标志：监控电压是否高于或低于 V_{det1}
			LVD1SR.DET 标志： V_{det1} 通道检测
电压检测过程	重置	电压监控器 0 重置 检测到 $VCC \geq V_{det0}$ 时取消内部复位信号；检测到 $VCC < V_{det0}$ 时产生内部复位信号，并保持复位状态，直到检测到 $VCC \geq V_{det0}$ 为止。	电压监控器 1 重置 检测到 $VCC \geq V_{det1}$ 时，取消内部复位信号。检测到 $VCC < V_{det1}$ 时，生成内部复位信号并保持复位状态。复位信号在指定时间后取消。
	打断	无中断	电压监控器 1 中断 可选择不可屏蔽或可屏蔽中断 当 VCC 越过 V_{det1} 时，发出中断请求。

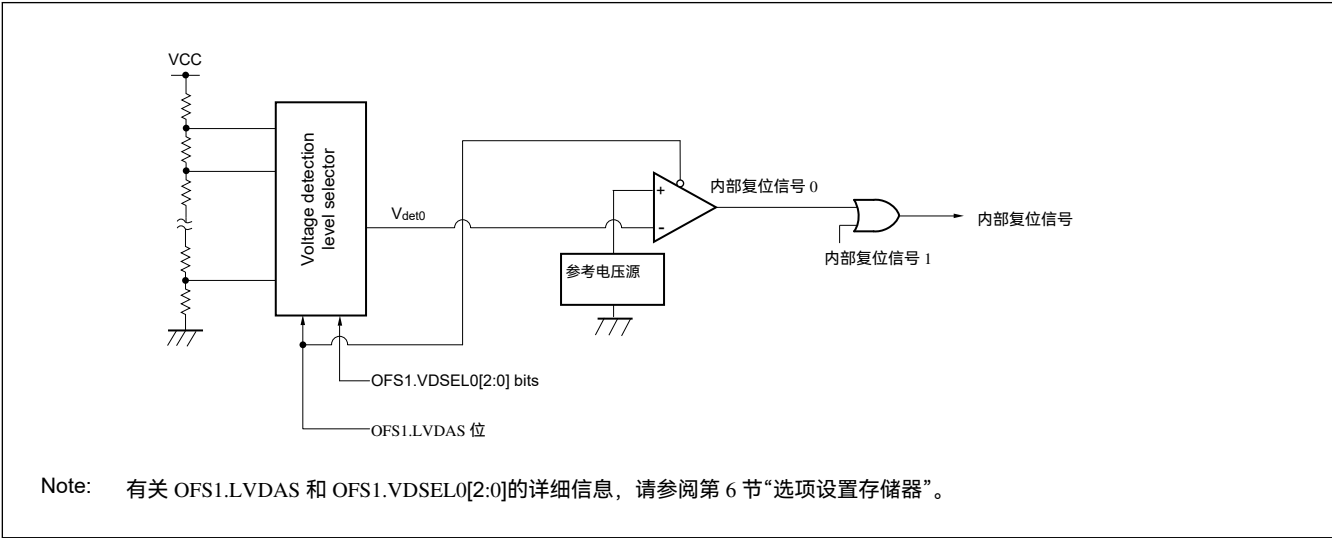


图 7.1 电压监测器 0 复位生成电路框图

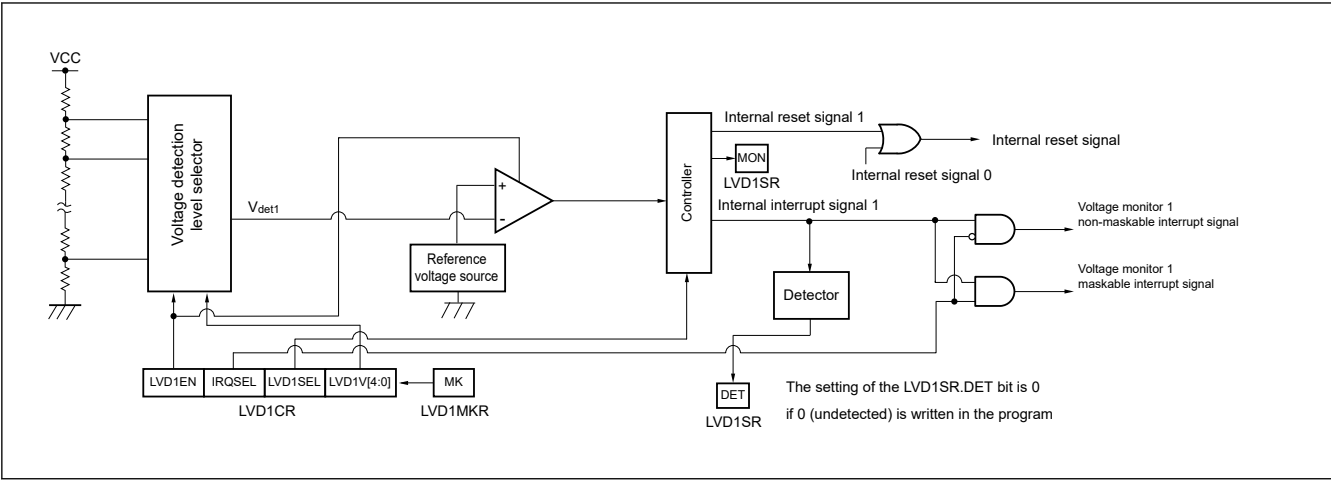


Figure 7.2 Block diagram of voltage monitor 1 interrupt and reset circuit

7.2 Register Descriptions

7.2.1 LVD1CR : Voltage Monitor 1 Circuit Control Register

Base address: SYSC = 0x4001_E000
Offset address: 0x0840

Bit position:	7	6	5	4	3	2	1	0
Bit field:	LVD1E N	LVD1S EL	IRQSE L	LVD1V[4:0]				
Value after reset:	0	0	0	1	1	0	0	1

Bit	Symbol	Function	R/W
4:0	LVD1V[4:0]	Voltage Detection 1 Level Select*1 *2 0x0E: V _{det1_0} 0x0F: V _{det1_1} 0x10: V _{det1_2} 0x11: V _{det1_3} 0x12: V _{det1_4} 0x13: V _{det1_5} 0x14: V _{det1_6} 0x15: V _{det1_7} 0x16: V _{det1_8} 0x17: V _{det1_9} 0x18: V _{det1_A} 0x19: V _{det1_B} 0x1A: V _{det1_C} 0x1B: V _{det1_D} 0x1C: V _{det1_E} 0x1D: V _{det1_F} 0x1E: V _{det1_10} 0x1F: V _{det1_11} Others: Setting prohibited	R/W
5	IRQSEL	Voltage Monitor 1 Interrupt Type Select 0: Non-maskable interrupt 1: Maskable interrupt	R/W
6	LVD1SEL	Operation mode of LVD1 0: Interrupt mode 1: Reset mode	R/W

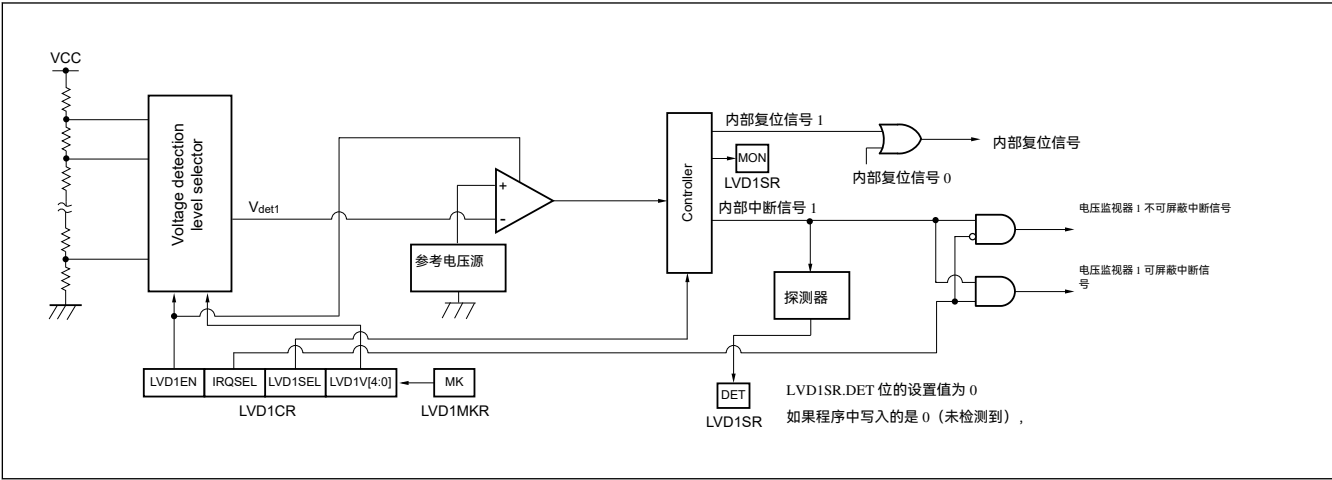


图 7.2 电压监控器 1 中断和复位电路框图

7.2寄存器描述

7.2.1 LVD1CR: 电压监视器 1 电路控制寄存器

基地址: SYSC = 0x4001E000
偏移地址: 0x0840

位位置:	7	6	5	4	3	2	1	0
位域:	LVD1E N	LVD1S EL	IRQSE L	LVD1V[4:0]				
重置后的值:	0	0	0	1	1	0	0	1

少量	象征	功能	R/W
4:0	LVD1V[4:0]	电压检测 1 级选择*1 *2 0x0E: V _{det1_0} 0x0F: V _{det1_1} 0x10: V _{det1_2} 0x11: V _{det1_3} 0x12: V _{det1_4} 0x13: V _{det1_5} 0x14: V _{det1_6} 0x15: V _{det1_7} 0x16: V _{det1_8} 0x17: V _{det1_9} 0x18: V _{det1_A} 0x19: V _{det1_B} 0x1A: V _{det1_C} 0x1B: V _{det1_D} 0x1C: V _{det1_E} 0x1D: V _{det1_F} 0x1E: V _{det1_10} 0x1F: V _{det1_11} 其他: 禁止设置	R/W
5	IRQSEL	电压监视器 1 中断类型选择 0: 不可屏蔽中断 1: 可屏蔽中断	R/W
6	LVD1SEL	LVD1 的工作模式 0: 中断模式 1: 重置模式	R/W

Bit	Symbol	Function	R/W
7	LVD1EN	Enabling Operation of LVD1 When using voltage detection 1 interrupt/reset or the LVD1SR.MON flag, set the LVD1EN bit to 1. The voltage detection 1 circuit starts when LVD1 operation stabilization time ($t_{d(E-A)}$) elapses after the LVD1EN bit value is changed from 0 to 1. For details on $t_{d(E-A)}$, see section 27, Electrical Characteristics . 0: Operation stopped 1: Operation enabled	R/W

Note:

- Set the PRCR.PRC3 bit to 1 (write enabled) before rewriting this register.
- Set the LVD1MKR.MK bit to 1 (write enabled) before rewriting this register.

Note 1. See [section 27, Electrical Characteristics](#) for the voltage levels to be detected.

Note 2. When using LVD0, set the detection voltage of LVD1 higher than the detection voltage of LVD0.

7.2.2 LVD1MKR : Voltage Monitor 1 Circuit Mask Register

Base address:	SYSC = 0x4001_E000						
Offset address:	0x0841						
Bit position:	7	6	5	4	3	2	1 0
Bit field:	—	—	—	—	—	—	MK
Value after reset:	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
0	MK	Specification of Whether to Enable or Disable Rewriting th LVD1CR Register The MK bit enables or disables the rewriting of the Voltage Monitor 1 Circuit Control Register (LVD1CR). While the MK bit is 1, the reset and interrupt generation by LVD1 are masked. Therefore, clear the MK bit to 0 after having written a new value to the LVD1CR register. 0: Rewriting of the LVD1CR register is disabled. 1: Rewriting of the LVD1CR register is enabled (reset and interrupt generation by LVD1 are masked)	R/W
7:1	—	These bits are read as 0. The write value should be 0.	R/W

Note: Set the PRCR.PRC3 bit to 1 (write enabled) before rewriting this register.

7.2.3 LVD1SR : Voltage Monitor 1 Circuit Status Register

Base address:	SYSC = 0x4001_E000						
Offset address:	0x0843						
Bit position:	7	6	5	4	3	2	1 0
Bit field:	—	—	—	—	—	—	MON DET ^{*1}
Value after reset:	0	0	0	0	0	0	1 0

Bit	Symbol	Function	R/W
0	DET	Voltage Monitor 1 Voltage Variation Detection Flag 0: Not detected 1: V_{det1} crossing is detected	R/W ^{*2}
1	MON	Voltage Monitor 1 Signal Monitor Flag 0: $VCC < V_{det1}$ 1: $VCC \geq V_{det1}$ or MON is disabled	R
7:2	—	These bits are read as 0. The write value should be 0.	R/W

Note: Set the PRCR.PRC3 bit to 1 (write enabled) before rewriting this register.

Note 1. The DET bit is invalid when reset mode is selected.

Note 2. Only 0 can be written to this bit.

少量	象征	功能	R/W
7	LVD1EN	启用LVD1的运行 使用电压检测 1 中断/复位或 LVD1SR.MON 标志时，将 LVD1EN 位设置为 1。电压检测 1 电路在 LVD1EN 位值从 0 变为 1 后，经过 LVD1 操作稳定时间($t_{d(E-A)}$)时启动。有关 $t_{d(E-A)}$ 的详细信息，请参阅第 27 节“电气特性”。 0: 操作已停止；1: 操作已启用	R/W

注意:

- 在重写此寄存器之前，请将 PRCR.PRC3 位设置为 1（使能写入）。
- 在重写此寄存器之前，请将 LVD1MKR.MK 位设置为 1（使能写入）。

注 1. 有关要检测的电压等级，请参阅第 27 节“电气特性”。

注2. 使用LVD0时，应将LVD1的检测电压设置得高于LVD0的检测电压。

7.2.2LVD1MKR：电压监视器 1 电路掩码寄存器

基地址:	SYSC = 0x4001E000						
偏移地址:	0x0841						
位位置:	7	6	5	4	3	2	1 0
位域:	—	—	—	—	—	—	MK
重置后的值:	0	0	0	0	0	0	0

少量	象征	功能	R/W
0	MK	LVD1CR 寄存器重写启用/禁用规范：MK 位用于启用或禁用电压监控器 1 电路控制寄存器 (LVD1CR) 的重写功能。当 MK 位为 1 时，LVD1 的复位和中断生成功能将被屏蔽。因此，在向 LVD1CR 寄存器写入新值后，应将 MK 位清除为 0。 0: LVD1CR 寄存器的重写功能已禁用。 1: LVD1CR寄存器重写功能已启用（LVD1的复位和中断生成功能已屏蔽）	R/W
7:1	—	这些位读取为 0。写入值应为 0。	R/W

注意：在重写此寄存器之前，请将 PRCR.PRC3 位设置为 1（使能写入）。

7.2.3LVD1SR：电压监控器 1 电路状态寄存器

基地址:	SYSC = 0x4001E000						
偏移地址:	0x0843						
位位置:	7	6	5	4	3	2	1 0
位域:	—	—	—	—	—	—	MON DET ^{*1}
重置后的值:	0	0	0	0	0	0	1 0

少量	象征	功能	R/W
0	DET	电压监控器 1 电压变化检测标志 0: 未检测到 1: 检测到 V_{det1} 交叉	R/W ^{*2}
1	MON	电压监视器 1 信号监视器标志 0: $VCC < V_{det1}$ 1: $VCC \geq V_{det1}$ 或 MON 已禁用	R
7:2	—	这些位读取为 0。写入值应为 0。	R/W

注意：在重写此寄存器之前，请将 PRCR.PRC3 位设置为 1（使能写入）。

注1. 选择复位模式时，DET 位无效。

注2: 此位只能写入0。

DET flag (Voltage Monitor 1 Voltage Variation Detection Flag)

The DET flag is enabled when LVD1CR.LVD1EN = 1 (LVD1 operation enabled) and LVD1CR.LVD1SEL = 0 (LVD1 is set to interrupt mode).

MON flag (Voltage Monitor 1 Signal Monitor Flag)

The MON flag is enabled when LVD1CR.LVD1EN = 1 (LVD1 operation enabled).

7.3 VCC Input Voltage Monitor

7.3.1 Monitoring V_{det0}

The comparison results from voltage monitor 0 are not available for reading.

7.3.2 Monitoring V_{det1}

Table 7.2 shows the procedures to set up monitoring against V_{det1} . After the settings are complete, the comparison results from voltage monitor 1 can be monitored with the LVD1SR.MON flag.

Table 7.2 Procedures to set up monitoring against V_{det1}

Step	Monitoring the comparison results from voltage monitor 1
1	Set the LVD1MKR.MK bit to 1 to enable writing to the LVD1CR register.
2	LVD1CR register setting - Set LVD1V[4:0] bits to set the LVD1 detection voltage - Set LVD1EN bit = 1 to enable LVD1 operation
3	LVD1 is enabled after $t_{d(E-A)}$ elapses as stability wait time.
4	Set the LVDMKR.MK bit to 0 to disable writing to the LVD1CR register.

7.4 Reset from Voltage Monitor 0

When using the reset from voltage monitor 0, clear the OFS1.LVDAS bit to 0 to enable the voltage monitor 0 reset after a reset.

Figure 7.3 shows the timing of the internal reset signal generated in the LVD0 reset mode.

DET 标志（电压监视器 1 电压变化检测标志）

当 LVD1CR.LVD1EN = 1（LVD1 操作已启用）和 LVD1CR.LVD1SEL = 0（LVD1 设置为中断模式）时，DET 标志启用。

MON 标志（电压监视器 1 信号监视器标志）

当 LVD1CR.LVD1EN = 1 时，MON 标志启用（LVD1 操作已启用）。

7.3VCC 输入电压监控器

7.3.1 监控 V_{det0}

电压监测器 0 的比较结果无法读取。

7.3.2 监控 V_{det1}

表 7.2 显示了设置对 V_{det1} 进行监控的步骤。设置完成后，可以使用 LVD1SR.MON 标志监控电压监控器 1 的比较结果。

表 7.2 设置针对 V_{det1} 的监控的步骤

步	监测电压监测器 1 的对比结果
1	将 LVD1MKR.MK 位设置为 1，以启用对 LVD1CR 寄存器的写入。
2	LVD1CR 寄存器设置 ● 设置 LVD1V[4:0] 位以设置 LVD1 检测电压 ● 设置 LVD1EN 位 = 1 以启用 LVD1 操作
3	LVD1 在 $t_{d(E-A)}$ 稳定性等待时间结束后启用。
4	将 LVDMKR.MK 位设置为 0 可禁用对 LVD1CR 寄存器的写入。

7.4 从电压监视器 0 重置

使用电压监视器 0 复位时，将 OFS1.LVDAS 位清除为 0，以便在复位后启用电压监视器 0 复位。

图 7.3 显示了 LVD0 复位模式下产生的内部复位信号的时序。

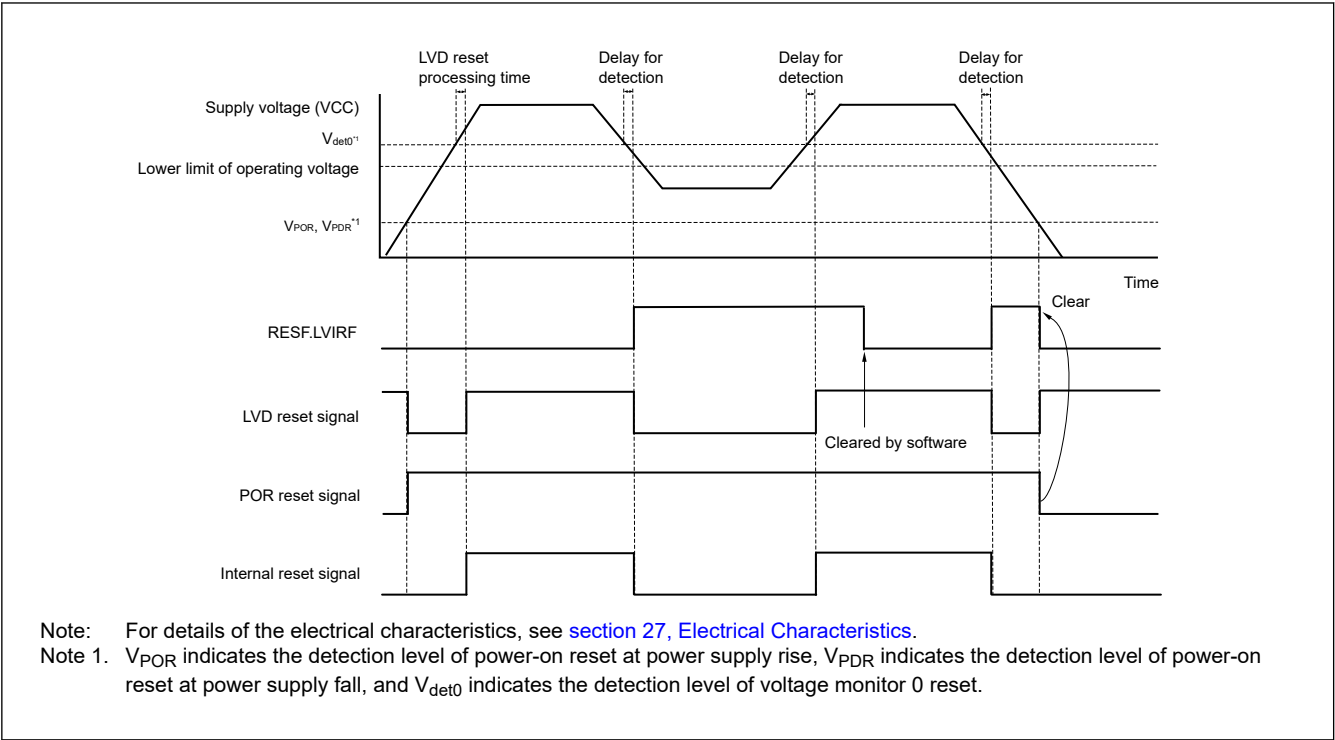


Figure 7.3 Timing of LVD0 internal reset signal generation

7.5 Interrupt and Reset from Voltage Monitor 1

An interrupt or reset can be generated in response to the comparison results from the voltage monitor 1 circuit.

[Table 7.3](#) shows the procedures for setting bits related to the voltage monitor 1 interrupt/reset so that voltage monitoring occurs. [Table 7.4](#) shows the procedures for setting bits related to the voltage monitor 1 interrupt/reset so that voltage monitoring stops. [Figure 7.4](#) shows an example of operations for a voltage monitor 1 interrupt. For the operation of the voltage monitor 1 reset, see [Figure 5.2](#) in [section 5, Resets](#).

Table 7.3 Procedures for setting bits related to voltage monitor 1 interrupt and voltage monitor 1 reset so that voltage monitoring occurs

Step	Voltage monitor 1 interrupt	Voltage monitor 1 reset
1	Set the LVD1MKR.MK bit to 1 to enable writing to the LVD1CR register.	
2	LVD1CR register setting - Set LVD1V[4:0] bits to set the LVD1 detection voltage - Select the interrupt request condition in the IRQSEL bit - Set LVD1SEL = 0 to set LVD1 operation mode to interrupt mode - Set LVD1EN = 1 to enable LVD1 operation	LVD1CR register setting - Set LVD1V[4:0] bits to set the LVD1 detection voltage - Set LVD1SEL = 1 to set LVD1 operation mode to reset mode - Set LVD1EN = 1 to enable LVD1 operation
3	LVD1 is enabled after $t_{d(E-A)}$ elapses as stability wait time.	
4	Set the LVD1MKR.MK bit to 0 to disable writing to the LVD1CR register.	

Table 7.4 Procedures for setting bits related to voltage monitor 1 interrupt and voltage monitor 1 reset so that voltage monitoring stops

Step	Voltage monitor 1 interrupt, voltage monitor 1 reset
1	Set the LVD1MKR.MK bit to 1 to enable writing to the LVD1CR register.
2	Set LVD1CR.LVD1EN = 0 to disable LVD1 operation.
3	Set the LVD1MKR.MK bit to 0 to disable writing to the LVD1CR register.

[Figure 7.4](#) shows the timing of the interrupt request signal generated in the LVD1 interrupt mode.

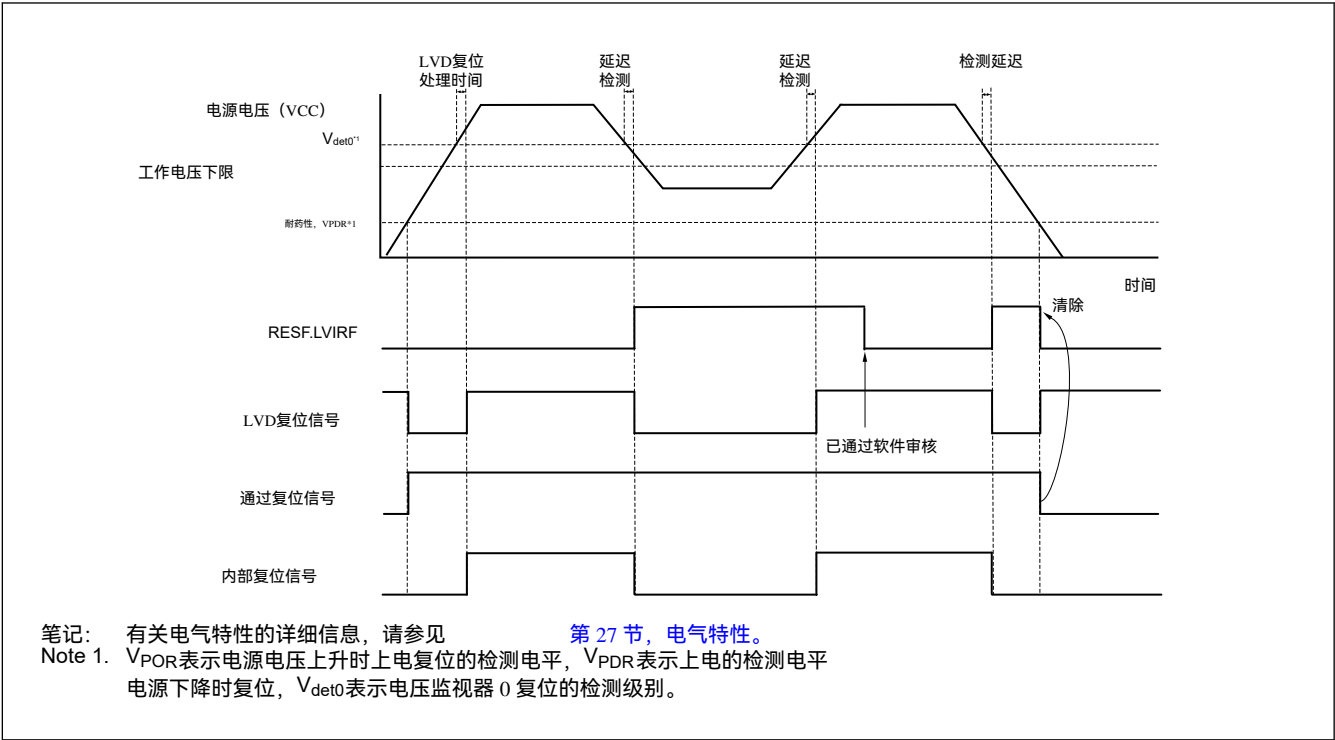


图 7.3 LVD0 内部复位信号生成的时序

7.5 电压监视器 1 的中断和复位

可以根据电压监视器 1 电路的比较结果生成中断或复位信号。

表 7.3 显示了设置与电压监控器 1 中断/复位相关的位以启动电压监控的步骤。表 7.4 显示了设置与电压监控器 1 中断/复位相关的位以停止电压监控的步骤。图 7.4 显示了电压监控器 1 中断操作的示例。有关电压监控器 1 复位的操作, 请参见第 5 节“复位”中的图 5.2。

表 7.3 设置与电压监视器 1 中断和电压监视器 1 复位相关的位的步骤, 以便进行电压监视

步	电压监控器 1 中断	电压监控器 1 重置
1	将 LVD1MKR.MK 位设置为 1, 以启用对 LVD1CR 寄存器的写入。	
2	LVD1CR 寄存器设置 ● 设置 LVD1V[4:0] 位以设置 LVD1 检测电压 ● 在 IRQSEL 位中选择中断请求条件 ● 设置 LVD1SEL= 0 将 LVD1 操作模式设置为中断模式 ● 设置 LVD1EN= 1 以启用 LVD1 操作	LVD1CR 寄存器设置 ● 设置 LVD1V[4:0] 位以设置 LVD1 检测电压 ● 设置 LVD1SEL= 1 将 LVD1 操作模式设置为复位模式 ● 设置 LVD1EN= 1 以启用 LVD1 操作
3	LVD1 在 $t_{d(E-A)}$ 稳定性等待时间结束后启用。	
4	将 LVD1MKR.MK 位设置为 0 以禁用对 LVD1CR 寄存器的写入。	

表 7.4 设置与电压监视器 1 中断和电压监视器 1 复位相关的位以停止电压监视的步骤

步	电压监视器 1 中断, 电压监视器 1 复位
1	将 LVD1MKR.MK 位设置为 1, 以启用对 LVD1CR 寄存器的写入。
2	设置 LVD1CR.LVD1EN= 0 以禁用 LVD1 操作。
3	将 LVD1MKR.MK 位设置为 0 以禁用对 LVD1CR 寄存器的写入。

图 7.4 显示了 LVD1 中断模式下生成的中断请求信号的时序。

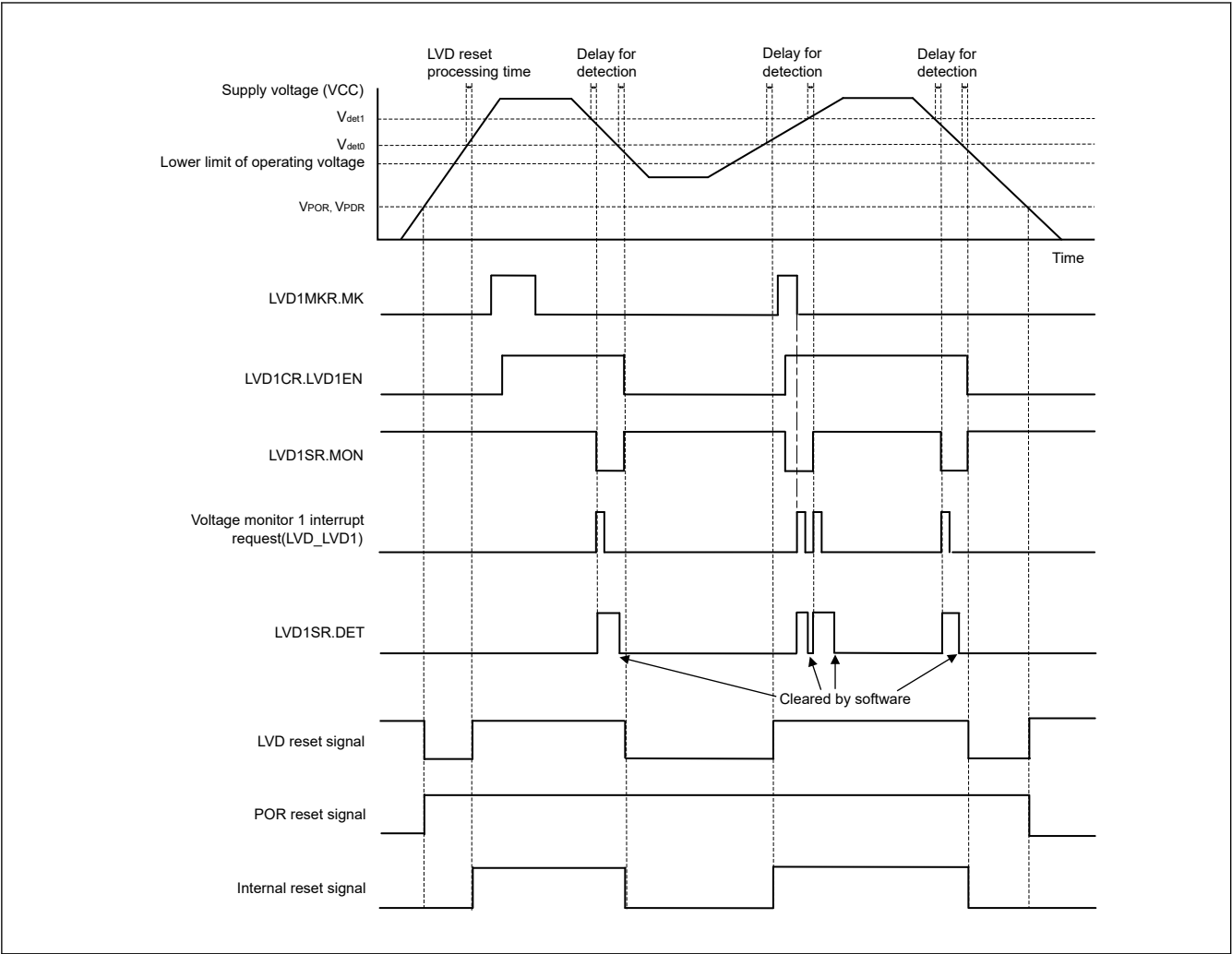


Figure 7.4 Timing of LVD1 interrupt request signal generation

Note: If operation of LVD1 is enabled while VCC is lower than the voltage detection level (V_{det1}), it generates an interrupt request signal (LVD_LVD1) at the time its operation is enabled.

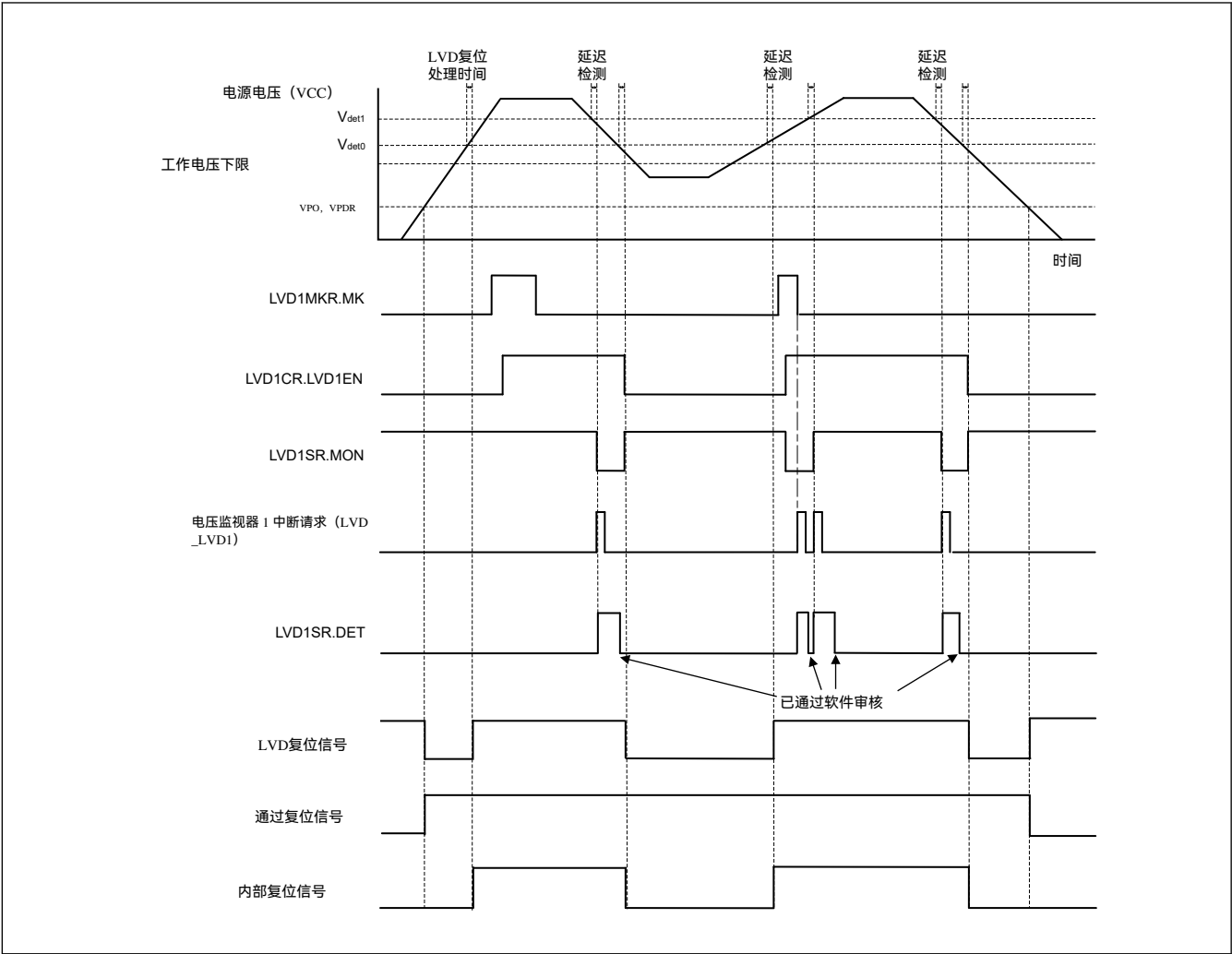


图 7.4 LVD1 中断请求信号生成的时序

注意：如果 LVD1 的操作在 VCC 低于电压检测电平(V_{det1}) 时启用，则在启用 LVD1 操作时会生成中断请求信号 (LVD_LVD1)。

8. Clock Generation Circuit

8.1 Overview

The MCU provides a clock generation circuit. Table 8.1 and Table 8.2 list the clock generation circuit specifications. Figure 8.1 shows a block diagram, and Table 8.3 lists the I/O pins.

Table 8.1 Clock generation circuit specifications for the clock sources

Clock source	Description	Specification
High-speed on-chip oscillator (HOCO)	Oscillation frequency	32 MHz
	User trimming	Available
Low-speed on-chip oscillator (LOCO)	Oscillation frequency	32.768 kHz
	User trimming	Available
External clock input for SWD (SWCLK)	Input clock frequency	Up to 12.5 MHz

Table 8.2 Clock generation circuit specifications for the internal clocks

Item	Clock source	Clock supply	Specification
System clock (ICLK)	HOCO/LOCO	CPU, DTC, FLASH, Flash-IF, SRAM	Up to 32 MHz 1 MHz to 32 MHz (P/E)
Peripheral module clock (PCLKB)	HOCO/LOCO	Peripheral modules	Up to 32 MHz
SysTick timer clock (SYSTICCLK)	LOCO	SysTick timer	32.768 kHz
Clock/buzzer output (CLKOUT)	HOCO/LOCO	PCLBUZ0 pin	Up to 16 MHz Division ratios: 1/2/4/8/16/2048/4096/8192 (HOCO) 1/2/4/8/16/32/64/128 (LOCO)
Serial wire clock (SWCLK)	SWCLK pin	OCD	Up to 12.5 MHz

8.时钟生成电路

8.1 概述

MCU 提供时钟生成电路。表 8.1 和表 8.2 列出了时钟生成电路的规格。图 8.1 显示了框图，表 8.3 列出了 I/O 引脚。

表 8.1 时钟源的时钟生成电路规格

时钟源	描述	规格
高速片上振荡器（HOCO）	振荡频率	32 MHz
	用户修剪	可用的
低速片上振荡器（LOCO）	振荡频率	32.768 kHz
	用户修剪	可用的
SWD 的外部时钟输入 (SWCLK)	输入时钟频率	最高可达 12.5 MHz

表 8.2 内部时钟的时钟生成电路规格

物品	时钟源	时钟电源	规格
系统时钟（ICLK）	HOCO/LOCO	CPU、DTC、闪存、闪存接口、SRAM	最高可达 32 MHz 1 MHz to 32 MHz (P/E)
外围模块时钟 (PCLKB)	HOCO/LOCO	外围模块	最高可达 32 MHz
SysTick 定时器时钟（系统时钟）	LOCO	系统滴答定时器	32.768 kHz
时钟/蜂鸣器输出 (CLKOUT)	HOCO/LOCO	PCLBUZ0 引脚	最高可达 16 MHz 划分比例: 1/2/4/8/16/2048/4096/8192 (HOCO) 1/2/4/8/16/32/64/128 (LOCO)
串行线时钟（SWCLK）	SWCLK引脚	OCD	最高可达 12.5 MHz

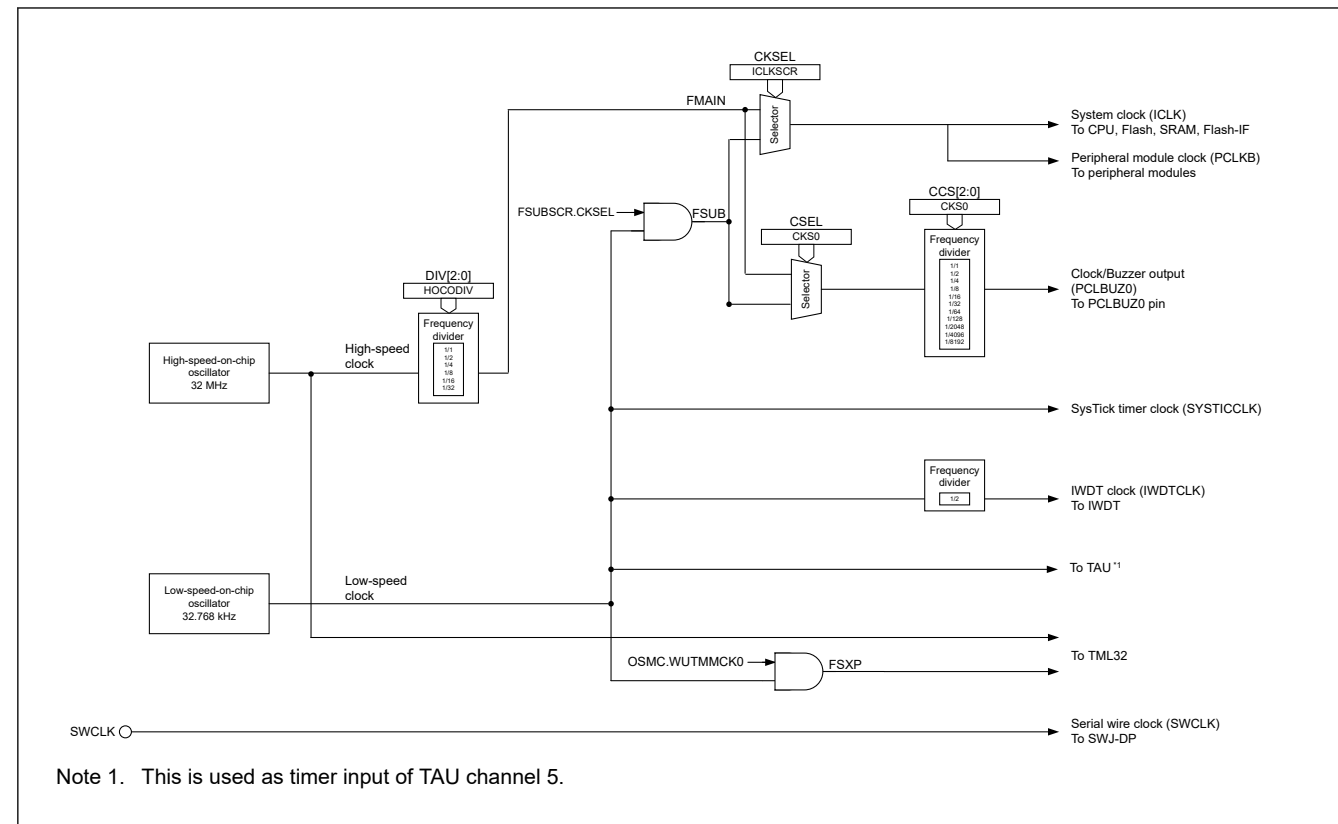


Figure 8.1 Clock generation circuit block diagram (20-pin)

Table 8.3 Clock generation circuit input/output pins

Pin name	I/O	Description
PCLBUZ0	Output	This pin is used to output the CLKOUT/BUZZER clock
SWCLK	Input	This pin is used to input from the SWD

8.2 Register Descriptions

8.2.1 FSUBSCR : FSUB Clock Source Control Register

Base address: SYSC = 0x4001_E000

Offset address: 0x0822

Bit position:	7	6	5	4	3	2	1	0
Bit field:	—	—	—	—	—	—	—	CKSE L
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
0	CKSEL	FSUB Clock Source Select 0: No clock source selected. 1: LOCO	R/W
7:1	—	These bits are read as 0. The write value should be 0.	R/W

Note: Set the PRCR.PRC0 bit to 1 (write enabled) before rewriting this register.

CKSEL bit (FSUB Clock Source Select)

The CKSEL bit selects the source for the following clock:

- Sub System clock (FSUB)

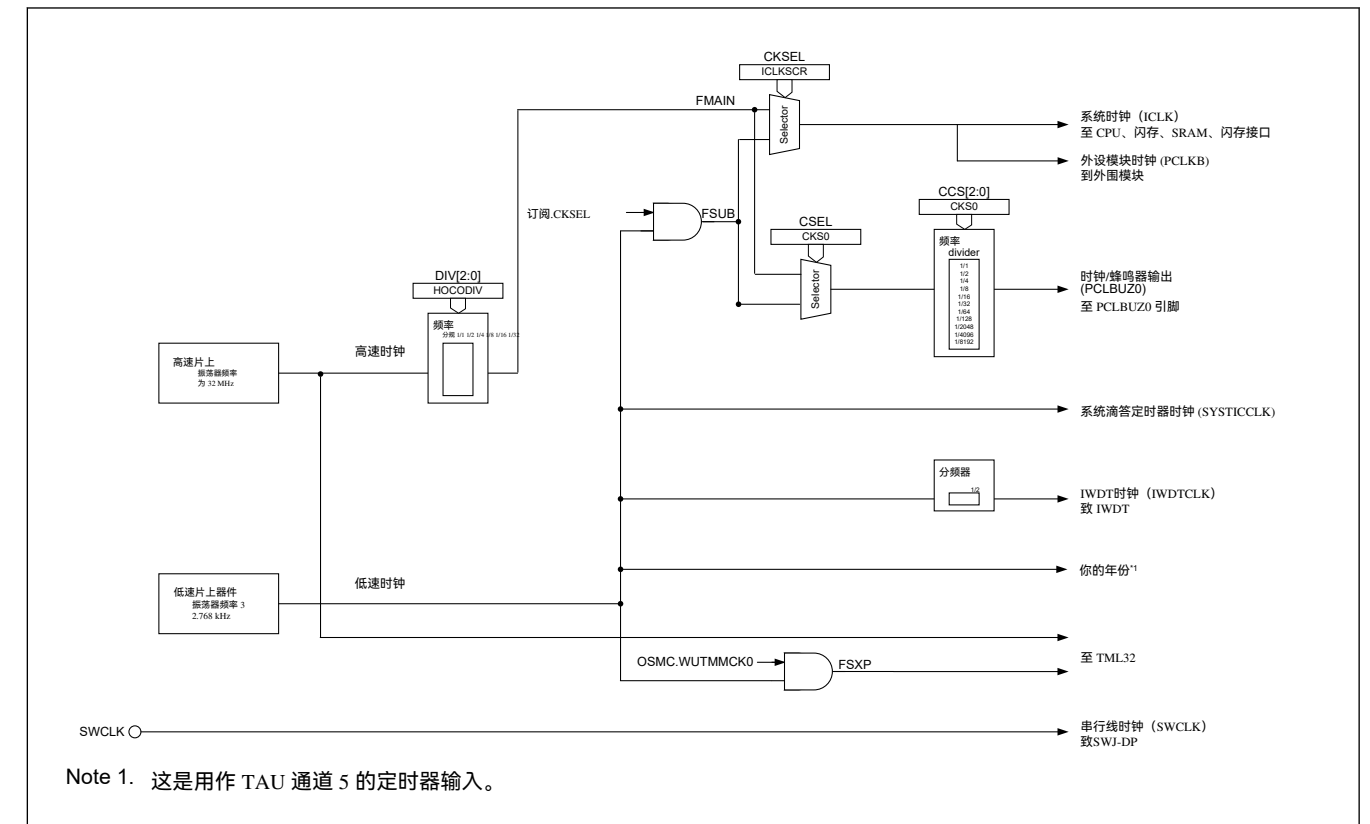


图 8.1 时钟生成电路框图 (20 引脚)

表 8.3 时钟发生电路输入/输出引脚

引脚名称	I/O	描述
PCLBUZ0	输出	此引脚用于输出 CLKOUT/BUZZER 时钟信号
SWCLK	输入	此引脚用于接收来自SWD的输入信号。

8.2 寄存器描述

8.2.1FSUBSCR: FSUB 时钟源控制寄存器

基地址: SYSC = 0x4001E000

偏移地址: 0x0822

位位置:	7	6	5	4	3	2	1	0
位域:	—	—	—	—	—	—	—	CKSFL
重置后的值:	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
0	CKSEL	FSUB 时钟源选择 0: 未选择时钟源。 1: LOCO	R/W
7:1	—	这些位读取为 0。写入值应为 0。	R/W

注意：在重写此寄存器之前，请将 PRCR.PRC0 位设置为 1（使能写入）。

CKSEL 位 (FSUB 时钟源选择)

CKSEL 位选择下一时钟源:

- 子系统时钟 (FSUB)

Writing to FSUBSCR.CKSEL is prohibited while MCU is under the following conditions:

- 1. ICLKSCR.CKSEL = 1
- 2. ICLKSCR.CKST = 1

The bit selects the following source:

- Low-speed on-chip oscillator clock (LOCO)

8.2.2 ICLKSCR : ICLK Clock Source Control Register

Base address:	SYSC = 0x4001_E000							
Offset address:	0x0823							
Bit position:	7	6	5	4	3	2	1	0
Bit field:	—	—	—	—	—	—	CKST	CKSEL
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
0	CKSEL	ICLK Clock Source Select 0: FMAIN 1: FSUB	R/W
1	CKST	ICLK Clock Source Status 0: FMAIN 1: FSUB	R
7:2	—	These bits are read as 0. The write value should be 0.	R/W

Note: Set the PRCR.PRC0 bit to 1 (write enabled) before rewriting this register.

The ICLKSCR register selects the clock source for the system clock.

CKSEL bit (ICLK Clock Source Select)

The CKSEL bit selects the source for the following clock:

- System clock (ICLK)

The bit selects one of the following sources:

- Main System clock (FMAIN)
- Sub System clock (FSUB)

CKST bit (ICLK Clock Source Status)

The CKST flag indicates the source for the following clock:

- System clock (ICLK)

8.2.3 LOCOCR : Low-speed On-chip Oscillator Control Register

Base address:	SYSC = 0x4001_E000							
Offset address:	0x080A							
Bit position:	7	6	5	4	3	2	1	0
Bit field:	—	—	—	—	—	—	—	LCSTP
Value after reset:	0	0	0	0	0	0	0	1

当 MCU 处于以下条件时，禁止写入 FSUBSCR.CKSEL：

- 1. ICLKSCR.CKSEL = 1
- 2. ICLKSCR.CKST = 1

该位选择以下信号源：

- 低速片上振荡器时钟（LOCO）

8.2.2 ICLKSCR：ICLK 时钟源控制寄存器

基本地址：	SYSC = 0x4001_E000							
偏移地址：	0x0823							
位位置：	7	6	5	4	3	2	1	0
位域：	—	—	—	—	—	—	CKST	CKSEL
重置后的值：	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
0	CKSEL	ICLK 时钟源选择 0: FMAIN 1: FSUB	R/W
1	CKST	ICLK 时钟源状态 0: FMAIN 1: FSUB	R
7:2	—	这些位读取为 0。写入值应为 0。	R/W

注意：在重写此寄存器之前，请将 PRCR.PRC0 位设置为 1（使能写入）。

ICLKSCR寄存器用于选择系统时钟的时钟源。

CKSEL 位（ICLK 时钟源选择）

CKSEL 位选择下一时钟源：

- 系统时钟（ICLK）

该位选择以下来源之一：

- 主系统时钟（FMAIN）
- 子系统时钟（FSUB）

CKST 位（ICLK 时钟源状态）

CKST 标志指示以下时钟的来源：

- 系统时钟（ICLK）

8.2.3 LOCOCR: 低速片上振荡器控制寄存器

基本地址：	SYSC = 0x4001_E000							
偏移地址：	0x080A							
位位置：	7	6	5	4	3	2	1	0
位域：	—	—	—	—	—	—	—	LCSTP
重置后的值：	0	0	0	0	0	0	0	1

Bit	Symbol	Function	R/W
0	LCSTP	LOCO Stop 0: Operate the LOCO clock 1: Stop the LOCO clock	R/W
7:1	—	These bits are read as 0. The write value should be 0.	R/W

Note: Set the PRCR.PRC0 bit to 1 (write enabled) before rewriting this register.

The LOCOCR register controls the LOCO clock.

LCSTP bit (LOCO Stop)

The LCSTP bit starts or stops the LOCO clock.

After setting the LCSTP bit to 0 to start the LOCO clock, only use the clock after the LOCO clock-oscillation stabilization wait time (t_{LOCO}) elapses. A fixed stabilization wait time is required after setting the LOCO clock to start operation. A fixed wait time is also required after setting the LOCO clock to stop.

The following restrictions apply when starting and stopping operation:

- After stopping the LOCO clock, allow a stop interval of at least 5 LOCO clock cycles before restarting it
- Confirm that LOCO oscillation is stable before stopping the LOCO clock
- Regardless of whether the LOCO is selected as the system clock, confirm that LOCO oscillation is stable before executing a WFI instruction to place the MCU in Software Standby mode
- When a transition to Software Standby mode is to follow the setting to stop the LOCO clock, wait for at least 3 LOCO cycles before executing the WFI instruction.

Writing 1 to LCSTP is prohibited under the following condition:

- FSUBSCR.CKSEL = 1 (Sub system clock source = LOCO).

During IWDT operation, LOCO oscillates regardless of the value of LCSTP.

8.2.4 HOCOCCR : High-speed On-chip Oscillator Control Register

Base address:	SYSC = 0x4001_E000							
Offset address:	0x0808							
Bit position:	7	6	5	4	3	2	1	0
Bit field:	—	—	—	—	—	—	—	HCST P
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
0	HCSTP	HOCO Stop 0: Operate the HOCO clock 1: Stop the HOCO clock	R/W
7:1	—	These bits are read as 0. The write value should be 0.	R/W

Note: Set the PRCR.PRC0 bit to 1 (write enabled) before rewriting this register.

The HOCOCCR register controls the HOCO clock.

HCSTP bit (HOCO Stop)

The HCSTP bit starts or stops the HOCO clock.

After setting the HCSTP bit to 0 to start the HOCO clock, confirm that the OSCSF.HOCOSF is set to 1 before using the clock. When OFS1.HOCOEN is set to 1, confirm that OSCSF.HOCOSF is also set to 1 before using the HOCO clock. A fixed stabilization wait time is required after setting the HOCO clock to start operation. A fixed wait time is also required after setting the HOCO clock to stop.

The following limitations apply when starting and stopping operation:

- After stopping the HOCO clock, confirm that the OSCSF.HOCOSF bit is 0 before restarting the HOCO clock.

少量	象征	功能	R/W
0	LCSTP	疯狂停车 0: 操作机车时钟 1: 停止 LOCO 时钟	R/W
7:1	—	这些位读取为 0。写入值应为 0。	R/W

注意：在重写此寄存器之前，请将 PRCR.PRC0 位设置为 1（使能写入）。

LOCOCR寄存器控制LOCO时钟。

LCSTP 位（LOCO 停止）

LCSTP 位用于启动或停止 LOCO 时钟。

将 LCSTP 位设置为 0 以启动 LOCO 时钟后，只有在 LOCO 时钟振荡稳定等待时间 (t_{LOCO}) 结束后才能使用该时钟。启动 LOCO 时钟后需要固定的稳定等待时间。停止 LOCO 时钟后也需要固定的等待时间。

启动和停止操作时，须遵守以下限制：

- 停止 LOCO 时钟后，至少等待 5 个 LOCO 时钟周期后再重新启动。
- 停止 LOCO 时钟之前，请确认 LOCO 振荡稳定。
- 无论是否选择 LOCO 作为系统时钟，在执行 WFI 指令将 MCU 置于软件待机模式之前，请确认 LOCO 振荡稳定。
- 当切换到软件待机模式时，要按照设置停止 LOCO 时钟，至少等待 3 个 LOCO 周期后再执行 WFI 指令。

在下列情况下，禁止向 LCSTP 发送第 1 行：

- FSUBSCR.CKSEL= 1(子系统时钟源= LOCO)

在 IWDT 运行期间，无论 LCSTP 的值如何，LOCO 都会振荡。

8.2.4 HOCOCCR: 高速片上振荡器控制寄存器

基地址: SYSC = 0x4001E000

偏移地址: 0x0808

位位置:	7	6	5	4	3	2	1	0
位域:	—	—	—	—	—	—	—	HCST P
重置后的值:	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
0	HCSTP	HOCO 停止 0: 操作 HOCO 时钟 1: 停止返校节计时器	R/W
7:1	—	这些位读取为 0。写入值应为 0。	R/W

注意：在重写此寄存器之前，请将 PRCR.PRC0 位设置为 1（使能写入）。

HOCOCCR寄存器控制HOCO时钟。

HCSTP 位（HOCO 停止）

HCSTP 位用于启动或停止 HOCO 时钟。

将 HCSTP 位设置为 0 以启动 HOCO 时钟后，请确认 OSCSF.HOCOSF 已设置为 1 后再使用该时钟。当 OFS1.HOCOEN 设置为 1 时，也请确认 OSCSF.HOCOSF 已设置为 1 后再使用该 HOCO 时钟。启动 HOCO 时钟后需要一段固定的稳定等待时间。停止 HOCO 时钟后也需要一段固定的等待时间。

启动和停止操作时，需遵守以下限制：

- 停止 HOCO 时钟后，确认 OSCSF.HOCOSF 位为 0，然后再重新启动 HOCO 时钟。

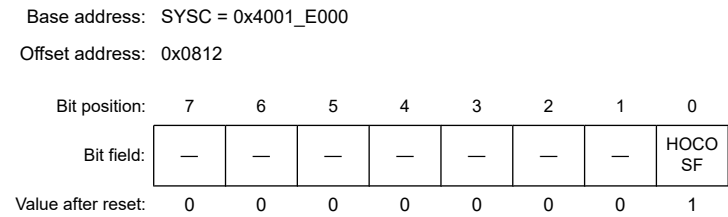
- Confirm that the HOCO clock operates and that the OSCSF.HOCOSF bit is 1 before stopping the HOCO clock.
- Regardless of whether the HOCO clock is selected as the system clock, confirm that the OSCSF.HOCOSF bit is set to 1 before executing a WFI instruction to place the MCU in Software Standby mode after setting HOCO operation with the HCSTP bit.
- When a transition to Software Standby mode is to follow the setting of the HOCO clock to stop, confirm that the OSCSF.HOCOSF bit is set to 0 after setting the HOCO clock and before executing the WFI instruction.

Writing 1 to HCSTP is prohibited under the following condition:

- ICLKSCR.CKST = 0 (system clock source = HOCO).

During On chip Debug operation, HOCO oscillates regardless of the value of HCSTP.

8.2.5 OSCSF : Oscillation Stabilization Flag Register



Bit	Symbol	Function	R/W
0	HOCOSF	HOCO Clock Oscillation Stabilization Flag 0: The HOCO clock is stopped, or the HOCO clock is being started and waiting for the precision of its oscillation to become stable is in progress. 1: The HOCO clock is operating with high precision.	R
7:1	—	These bits are read as 0.	R

The OSCSF register contains flags to indicate the operating status of the counters in the oscillation stabilization wait circuits for the individual oscillators. After oscillation starts, these counters measure the wait time until each oscillator output clock is supplied to the internal circuits. An overflow of a counter indicates that the clock supply is stable and available for the associated circuit.

HOCOSF flag (HOCO Clock Oscillation Stabilization Flag)

The HOCOSF flag indicates the operating status of the counter that measures the wait time for the high-speed clock oscillator (HOCO).

[Setting condition]

- When the HOCO oscillation stabilization time count is completed after the HOCOCR.HCSTP bit becomes 0 while HOCO is stopped. For the HOCO oscillation stabilization time, see [section 27, Electrical Characteristics](#).
- When the HOCO oscillation stabilization time has been counted after Software Standby mode is released.

[Clearing condition]

- When the HOCO clock is operating and then is deactivated because the HOCOCR.HCSTP bit is set to1.
- When the MCU enters Software Standby mode due to the WFI instruction.

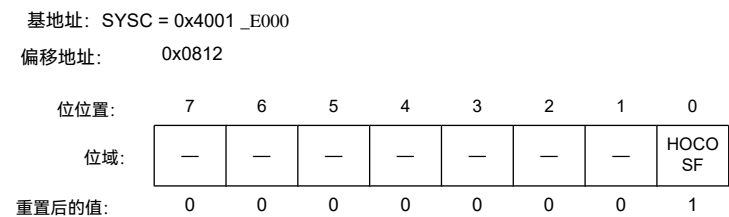
- 确认 HOCO 时钟运行正常，并且在停止 HOCO 时钟之前 OSCSF.HOCOSF 位为 1。
- 无论是否选择 HOCO 时钟作为系统时钟，在使用 HCSTP 位设置 HOCO 操作后，在执行 WFI 指令将 MCU 置于软件待机模式之前，请确认 OSCSF.HOCOSF 位设置为 1。
- 当软件待机模式的转换要按照 HOCO 时钟的设置停止时，请在设置 HOCO 时钟之后、执行 WFI 指令之前，确认 OSCSF.HOCOSF 位设置为 0。

在下列情况下，禁止向 HCSTP 发送第 1 封邮件：

- ICLKSCR.CKST= 0 （系统时钟源= HOCO）

在片上调试操作期间，无论 HCSTP 的值如何，HOCO 都会振荡。

8.2.5OSCSF：振荡稳定标志寄存器



少量	象征	功能	R/W
0	HOCOSF	HOCO时钟振荡稳定标志 0: HOCO 时钟已停止，或者 HOCO 时钟正在启动，正在等待其振荡精度稳定。 1: HOCO 时钟运行精度很高。	R
7:1	—	这些位读取为 0。	R

OSCSF寄存器包含一些标志位，用于指示各个振荡器振荡稳定等待电路中计数器的工作状态。振荡开始后，这些计数器会测量等待时间，直到每个振荡器的输出时钟信号被提供给内部电路。计数器溢出表示时钟信号稳定，可供相关电路使用。

HOCOSF 标志（HOCO 时钟振荡稳定标志）

HOCOSF 标志指示测量高速时钟振荡器 (HOCO) 等待时间的计数器的运行状态。

[设置条件]

- 当 HOCO 振荡停止且 HOCOCR.HCSTP 位变为 0 时，HOCO 振荡稳定时间计数完成。有关 HOCO 振荡稳定时间，请参见第 27 节“电气特性”。
- 当软件待机模式解除后，HOCO 振荡稳定时间已计算完毕。

[清除条件]

- 当 HOCO 时钟正在运行，然后由于 HOCOCR.HCSTP 位设置为 1 而被停用时。
- 当MCU由于WFI指令进入软件待机模式时。

8.2.6 HOCODIV : High-speed On-chip Oscillator Frequency Select Register

Base address: SYSC = 0x4001_E000

Offset address: 0x0818

Bit position:

76543210

Bit field:

—	—	—	—	—	DIV[2:0]		

Value after reset:

00000011

Bit	Symbol	Function	R/W
2:0	DIV[2:0]	High-speed On-chip Oscillator Clock Division Ratio 0 0 0: × 1/1 0 0 1: × 1/2 0 1 0: × 1/4 0 1 1: × 1/8 1 0 0: × 1/16 1 0 1: × 1/32 Others: Setting prohibited	R/W
7:3	—	These bits are read as 0. The write value should be 0.	R/W

Note: Set the PRCR.PRC0 bit to 1 (write enabled) before rewriting this register.

8.2.7 OSMC : Subsystem Clock Supply Mode Control Register

Base address: SYSC = 0x4001_E000

Offset address: 0x0824

Bit position:

76543210

Bit field:

—	—	—	WUTM MCK0	—	—	—	—

Value after reset:

00000000

Bit	Symbol	Function	R/W
3:0	—	These bits are read as 0. The write value should be 0.	R/W
4	WUTMMCK0	Selection of the Operating clock source for the 32-bit Interval Timer 0: No clock source selected. 1: LOCO	R/W
7:5	—	These bits are read as 0. The write value should be 0.	R/W

Note: Set the PRCR.PRC0 bit to 1 (write enabled) before rewriting this register.

8.2.8 CKS0 : Clock Out Control Register 0

Base address: PCLBUZ = 0x400A_3B00

Offset address: 0x0001

Bit position:

76543210

Bit field:

PCLO E	—	—	—	CSEL	CCS[2:0]		

Value after reset:

00000000

8.2.6 HOCODIV: 高速片上振荡器频率选择寄存器

基地址: SYSC= 0x4001E000

Offset address: 0x0818

位位置:

76543210

位域:

—	—	—	—	—	DIV[2:0]		

重置后的值:

00000011

少量	象征	功能	R/W
2:0	DIV[2:0]	高速片上振荡器时钟分频比 0 0 0: × 1/1 0 0 1: × 1/2 0 1 0: × 1/4 0 1 1: × 1/8 1 0 0: × 1/16 1 0 1: × 1/32 其他: 禁止设置	R/W
7:3	—	这些位读取为 0。写入值应为 0。	R/W

注意: 在重写此寄存器之前, 请将 PRCR.PRC0 位设置为 1 (使能写入) 。

8.2.7OSMC: 子系统时钟供电模式控制寄存器

基地址: SYSC = 0x4001E000

偏移地址: 0x0824

位位置:

76543210

位域:

—	—	—	WUTM MCK0	—	—	—	—

重置后的值:

00000000

少量	象征	功能	R/W
3:0	—	这些位读取为 0。写入值应为 0。	R/W
4	WUTMMCK0	32 位间隔定时器 0 的工作时钟源选择: 未选择时钟源。 1: LOCO	R/W
7:5	—	这些位读取为 0。写入值应为 0。	R/W

注意: 在重写此寄存器之前, 请将 PRCR.PRC0 位设置为 1 (使能写入) 。

8.2.8CKS0: 时钟输出控制寄存器 0

基地址: PCLBUZ = 0x400A_3B00

偏移地址: 0x0001

位位置:

76543210

位域:

PCLO E	—	—	—	CSEL	CCS[2:0]		

重置后的值:

00000000

Bit	Symbol	Function	R/W
2:0	CCS[2:0]	Clock Out Divide Select 0 0 0: value after reset FMAIN (When CKS0.CSEL = 0) FSUB (When CKS0.CSEL = 1) 0 0 1: FMAIN × 1/2 (When CKS0.CSEL = 0) FSUB × 1/2 (When CKS0.CSEL = 1) 0 1 0: FMAIN × 1/2 ² (When CKS0.CSEL = 0) FSUB × 1/2 ² (When CKS0.CSEL = 1) 0 1 1: FMAIN × 1/2 ³ (When CKS0.CSEL = 0) FSUB × 1/2 ³ (When CKS0.CSEL = 1) 1 0 0: FMAIN × 1/2 ⁴ (When CKS0.CSEL = 0) FSUB × 1/2 ⁴ (When CKS0.CSEL = 1) 1 0 1: FMAIN × 1/2 ¹¹ (When CKS0.CSEL = 0) FSUB × 1/2 ⁵ (When CKS0.CSEL = 1) 1 1 0: FMAIN 1/2 ¹² (When CKS0.CSEL = 0) FSUB × 1/2 ⁶ (When CKS0.CSEL = 1) 1 1 1: FMAIN × 1/2 ¹³ (When CKS0.CSEL = 0) FSUB × 1/2 ⁷ (When CKS0.CSEL = 1)	R/W
3	CSEL	Clock Out Select 0: FMAIN 1: FSUB	R/W
6:4	—	These bits are read as 0. The write value should be 0.	R/W
7	PCLOE	Clock Out Enable 0: Disable clock out 1: Enable clock out	R/W

CCS[2:0] bits (Clock Out Divide Select)

The CCS[2:0] bits specify the clock division ratio. Set the PCLOE bit to 0 when changing the division ratio. The division ratio of the output clock frequency must be set to a value no higher than the characteristics of the PCLBUZ0 pin output frequency. For details on the characteristics of the PCLBUZ0 pin, see [section 27, Electrical Characteristics](#).

CSEL bit (Clock Out Select)

The CSEL bit selects the source of the clock to be output from the PCLBUZ0 pin. When changing the clock source, set the PCLOE bit to 0.

PCLOE bit (Clock Out Enable)

The PCLOE enables output from the PCLBUZ0 pin.

When this bit is set to 1, the selected clock is output. When this bit is set to 0, low is output. When changing this bit, confirm that the clock out source clock selected in the CSEL and CCS[2:0] bits is stable. Otherwise, a glitch might be generated in the output.

Clear this bit before entering Software Standby mode if the selecting clock out source clock is stopped in that mode.

8.2.9 LIOTRM : Low-speed On-chip Oscillator Trimming Register

Base address:	SYSC = 0x4001_E000						
Offset address:	0x0805						
Bit position:	7	6	5	4	3	2	1 0
Bit field:	LIOTRM[7:0]						
Value after reset:	1	0	0	0	0	0	0 0

少量	象征	功能	R/W
2:0	CCS[2:0]	时钟输出除法选择0 0 0:重置后的值 FMAIN (当 CKS0.CSEL= 0) FSUB (当 CKS0.CSEL= 1) 0 0 1: FMAIN × 1/2 (When CKS0.CSEL = 0) FSUB× 1/2 (当 CKS0.CSEL= 1) 0 1 0: FMAIN × 1/2 ² (When CKS0.CSEL = 0) FSUB × 1/2 ² (当 CKS0.CSEL= 1) 0 1 1: FMAIN × 1/2 ³ (When CKS0.CSEL = 0) FSUB × 1/2 ³ (当 CKS0.CSEL= 1) 1 0 0: FMAIN × 1/2 ⁴ (When CKS0.CSEL = 0) FSUB × 1/2 ⁴ (当 CKS0.CSEL= 1) 1 0 1: FMAIN × 1/2 ¹¹ (When CKS0.CSEL = 0) FSUB × 1/2 ⁵ (当 CKS0.CSEL= 1) 1 1 0: FMAIN 1/2 ¹² (When CKS0.CSEL = 0) FSUB× 1/2 ⁶ (当 CKS0.CSEL= 1) 1 1 1: FMAIN × 1/2 ¹³ (When CKS0.CSEL = 0) FSUB × 1/2 ⁷ (当 CKS0.CSEL= 1)	R/W
3	CSEL	下班选择 0: FMAIN 1: FSUB	R/W
6:4	—	这些位读取为 0。写入值应为 0。	R/W
7	PCLOE	时钟输出使能 0: 禁用时钟输出 1: 启用时钟输出	R/W

CCS[2:0] 位 (时钟输出分频选择)

CCS[2:0] 位指定时钟频比。更改分频比时，请将 PCLOE 位设置为 0。输出时钟频率的分频比必须设置为不高于 PCLBUZ0 引脚输出频率特性的值。有关 PCLBUZ0 引脚特性的详细信息，请参阅第 27 节“电气特性”。

CSEL 位 (时钟输出选择)

CSEL 位用于选择从 PCLBUZ0 引脚输出的时钟源。更改时钟源时，请将 PCLOE 位设置为 0。

PCLOE 位 (时钟输出使能)

PCLOE 使 PCLBUZ0 引脚能够输出信号。

当此位设置为 1 时，输出选定的时钟信号。当此位设置为 0 时，输出低电平。更改此位时，请确认 CSEL 和 CCS[2:0] 位中选择的时钟输出源时钟信号稳定。否则，输出中可能会出现毛刺。

如果选择时钟输出源时钟在软件待机模式下停止，则在进入软件待机模式之前清除此位。

8.2.9 LIOTRM: 低速片上振荡器微调寄存器

基本地址:	SYSC = 0x4001_E000						
偏移地址:	0x0805						
位位置:	7	6	5	4	3	2	1 0
位域:	LIOTRM[7:0]						
重置后的值:	1	0	0	0	0	0	0 0

Bit	Symbol	Function	R/W
7:0	LIOTRM[7:0]	LOCO User Trimming 0x00: Minimum speed 0x01: ⋮ 0x80: Initial value ⋮ 0xFE: 0xFF: Maximum speed	R/W

Note: Set the PRCR.PRC0 bit to 1 (write enabled) before rewriting this register.

The frequency of the low-speed on-chip oscillator can be trimmed by setting the LIOTRM register. When the LIOTRM register is modified, the frequency stabilization time corresponds to the frequency stabilization time at the start of the MCU operation. For details on the resolution of frequency trimming and the stabilization time of the low-speed on-chip oscillator clock, see [Table 27.4](#).

The frequency of the low-speed oscillator can be verified using a timer (such as a Timer Array Unit or a 32-bit interval timer) with a high-accuracy external clock input, or by other means.

Note: The frequency of the oscillator may vary due to changes in temperature and power supply voltage after frequency trimming. In such cases, regular trimming is essential, especially when high-frequency accuracy is required.

8.2.10 HIOTRM : High-speed On-chip Oscillator Trimming Register

Base address:	FLCN = 0x407E_C000							
Offset address:	0x0200							
Bit position:	7	6	5	4	3	2	1	0
Bit field:	—	—	HIOTRM[5:0]					
Value after reset:	0	0	x	x	x	x	x	x

Bit	Symbol	Function	R/W
5:0	HIOTRM[5:0]	HOCO User Trimming 0x00: Minimum speed ⋮ 0x3F: Maximum speed	R/W
7:6	—	These bits are read as 0. The write value should be 0.	R/W

Note: After a reset, the HIOTRM register is initialized with the trimming value set at the time of shipment.

The frequency of the high-speed on-chip oscillator can be trimmed by setting the HIOTRM register. When the HIOTRM register is modified, the frequency stabilization time corresponds to the frequency stabilization time at the start of the MCU operation. For details on the resolution of frequency trimming and the stabilization time of the high-speed on-chip oscillator clock, see [Table 27.4](#).

The frequency of the high-speed oscillator can be verified using a timer (such as a Timer Array Unit or a 32-bit interval timer) with a high-accuracy external clock input, or by other means.

Note: The frequency of the oscillator may vary due to changes in temperature and power supply voltage after frequency trimming. In such cases, regular trimming is essential, especially when high-frequency accuracy is required.

8.3 Internal Clock

Clock sources for the internal clock signals include:

- HOCO clock
- LOCO clock

The following internal clocks are produced from these sources.

- Operating clock of the CPU, DTC, Flash, Flash-IF, SRAM, and peripheral modules — system clock (ICLK)
- Operating clock for the IWDTC — IWDTCCLK

少量	象征	功能	R/W
7:0	LIOTRM[7:0]	LOCO 用户修剪 0x00: 最低速度 0x01: ⋮ 0x80: 初始值 ⋮ 0xFE: 0xFF: 最大速度	R/W

注意: 在重写此寄存器之前, 请将 PRCR.PRC0 位设置为 1 (使能写入)。

可以通过设置 LIOTRM 寄存器来微调片上低速振荡器的频率。当 LIOTRM 寄存器被修改时, 频率稳定时间对应于 MCU 启动时的频率稳定时间。有关频率微调分辨率和片上低速振荡器时钟稳定时间的详细信息, 请参见表 27.4。

低速振荡器的频率可以通过具有高精度外部时钟输入的定时器 (例如定时器阵列单元或 32 位间隔定时器) 或其他方式进行验证。

注意: 振荡器的频率可能会因温度和电源电压的变化而改变, 即使频率调整后也是如此。在这种情况下, 定期进行频率调整至关重要, 尤其是在需要高精度频率时。

8.2.10 HIOTRM: 高速片上振荡器微调寄存器

基地址:		FLCN= 0x407E_C000							
偏移地址:		0x0200							
位位置:		7	6	5	4	3	2	1	0
位域:		—	—	HIOTRM[5:0]					
重置后的值:		0	0	x	x	x	x	x	x

少量	象征	功能	R/W
5:0	HIOTRM[5:0]	HOCO 用户修剪 0x00: 最低速度 ⋮ 0x3F: 最大速度	R/W
7:6	—	这些位读取为 0。写入值应为 0。	R/W

注意: 复位后, HIOTRM 寄存器将初始化为发货时设置的修剪值。

可以通过设置 HIOTRM 寄存器来微调片上高速振荡器的频率。修改 HIOTRM 寄存器后, 频率稳定时间将与 MCU 启动时的频率稳定时间相对应。有关频率微调分辨率和片上高速振荡器时钟稳定时间的详细信息, 请参见表 27.4。

可以使用具有高精度外部时钟输入的定时器 (例如定时器阵列单元或 32 位间隔定时器) 或其他方法来验证高速振荡器的频率。

注意: 振荡器的频率可能会因温度和电源电压的变化而改变, 即使频率调整后也是如此。在这种情况下, 定期进行频率调整至关重要, 尤其是在需要高精度频率时。

8.3 内部时钟

内部时钟信号的时钟源包括:

- HOCO时钟
- LOCO时钟

以下内部时钟均由这些源生成。

- CPU、DTC、Flash、Flash-IF、SRAM 和外围模块的工作时钟——系统时钟 (ICLK)
- IWDTC 的工作时钟— IWDTCCLK

- Clock for external pin output — CLKOUT

For details of the registers used to set the frequencies of the internal clocks, see [section 8.3.1. System Clock \(ICLK\)](#) to [section 8.3.4. External Pin Output Clock \(CLKOUT\)](#).

If the value of these bits is changed, subsequent operation is at the frequency determined by the new value.

8.3.1 System Clock (ICLK)

The system clock, ICLK, is the operating clock for the CPU, DTC, Flash, Flash-IF, SRAM, and the peripheral modules. The frequency of the given clock is specified in the following bits:

- HOCOFRQ1[2:0] in OFS1
- DIV[2:0] in HOCODIV
- CKSEL in FSUBSCR
- CKSEL in ICLKSCR

When the clock source of ICLK is being switched, the duration of ICLK clock cycle become longer during the clock source transition period. See [Figure 8.2](#) and [Figure 8.3](#).

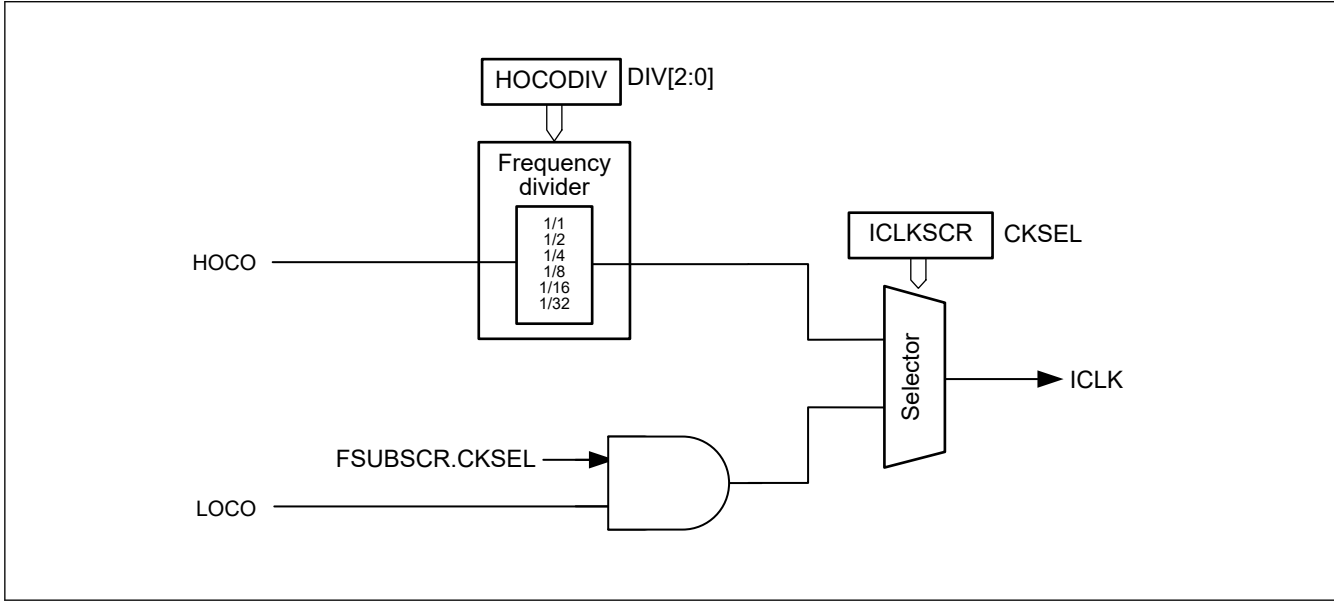


Figure 8.2 Block diagram of clock source selector

- 外部引脚输出时钟— CLKOUT

有关用于设置内部时钟频率的寄存器的详细信息，请参阅第 8.3.1 节“系统时钟 (ICLK)”至第 8.3.4 节“外部引脚输出时钟 (CLKOUT)”。

如果这些位的值发生变化，后续操作将以新值确定的频率进行。

8.3.1 系统时钟 (ICLK)

系统时钟 ICLK 是 CPU、DTC、Flash、Flash-IF、SRAM 和外围模块的运行时钟。

给定时钟的频率由以下几位指定：

- HOCOFRQ1[2:0] in OFS1
- DIV[2:0] in HOCODIV
- FSUBSCR 中的 CKSEL
- CKSEL 在 ICLKSCR 中

当 ICLK 的时钟源切换时，在时钟源切换期间，ICLK 时钟周期的持续时间会延长。参见图 8.2 和图 8.3。

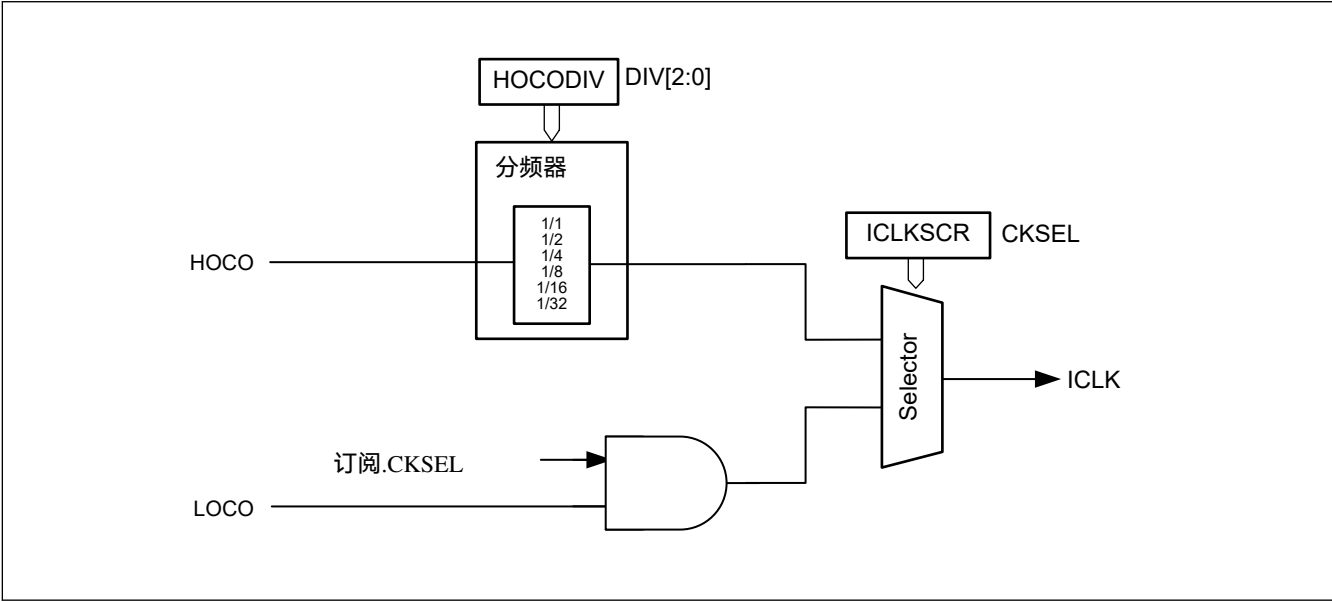


图 8.2 时钟源选择器的框图

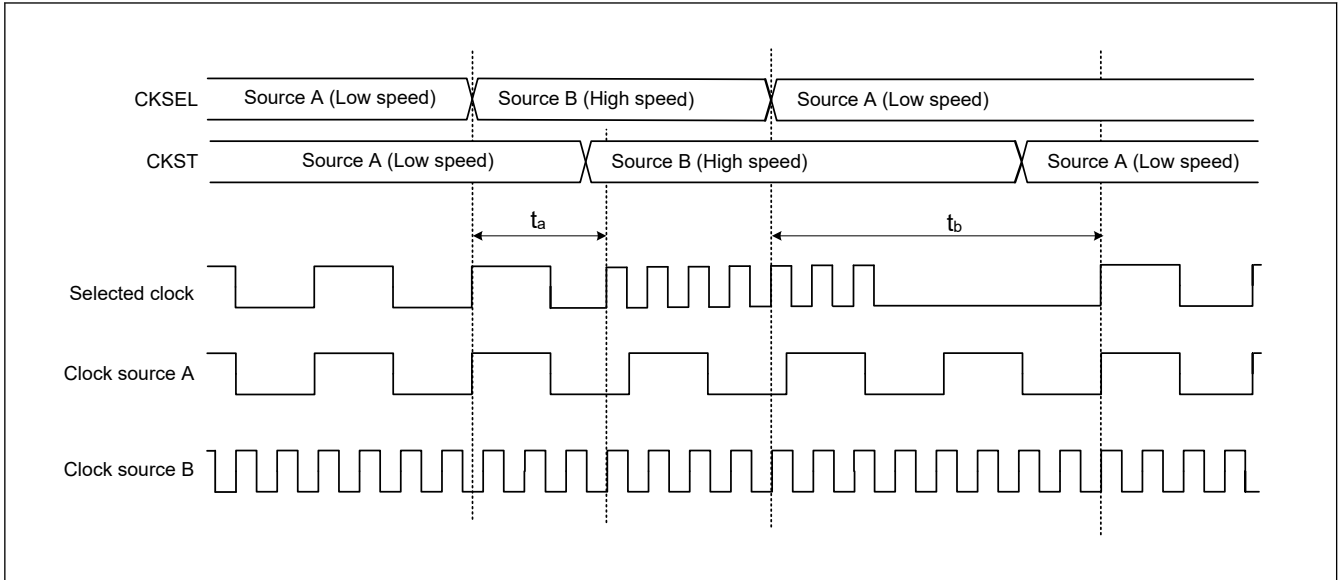


Figure 8.3 Timing of clock source switching

- Maximum Number of Clock Cycles Required for FMAIN ↔ FSUB
 t_a : 3 cycles
 t_b : $1 + 2 \times (\text{FMAIN frequency}) / (\text{FSUB frequency})$ cycles

8.3.2 IWDTClock (IWDTCCLK)

The IWDTClock (IWDTCCLK) is the operating clock for the IWDTC. IWDTCCLK is internally generated by the LOCO clock.

8.3.3 SysTick Timer-dedicated Clock (SYSTICCLK)

The SysTick timer-dedicated clock, SYSTICCLK, is the operating clock for the SysTick timer. SYSTICCLK is generated by the LOCO clock.

8.3.4 External Pin Output Clock (CLKOUT)

The CLKOUT is output externally for the clock or buzzer output. CLKOUT is output to the PCLBUZ0 pin when CKS0.PCLOE is set to 1. Only change the value in the CSEL or CCS[2:0] bits in CKS0 when the CKS0.PCLOE bit is 0.

The CLKOUT clock frequency is specified in the following bits:

- CSEL or CCS[2:0] in CKS0
- HOCOFQR1[2:0] in OFS1
- DIV[2:0] in HOCODIV
- CKSEL in FSUBSCR

8.4 Usage Notes

8.4.1 Register Access

- Do not write to registers listed in this section for the following condition:
[Registers]
 - FSUBSCR, HOCODIV
[Condition]
 - ICLKSCR.CKSEL = 1 (ICLK = LOCO)

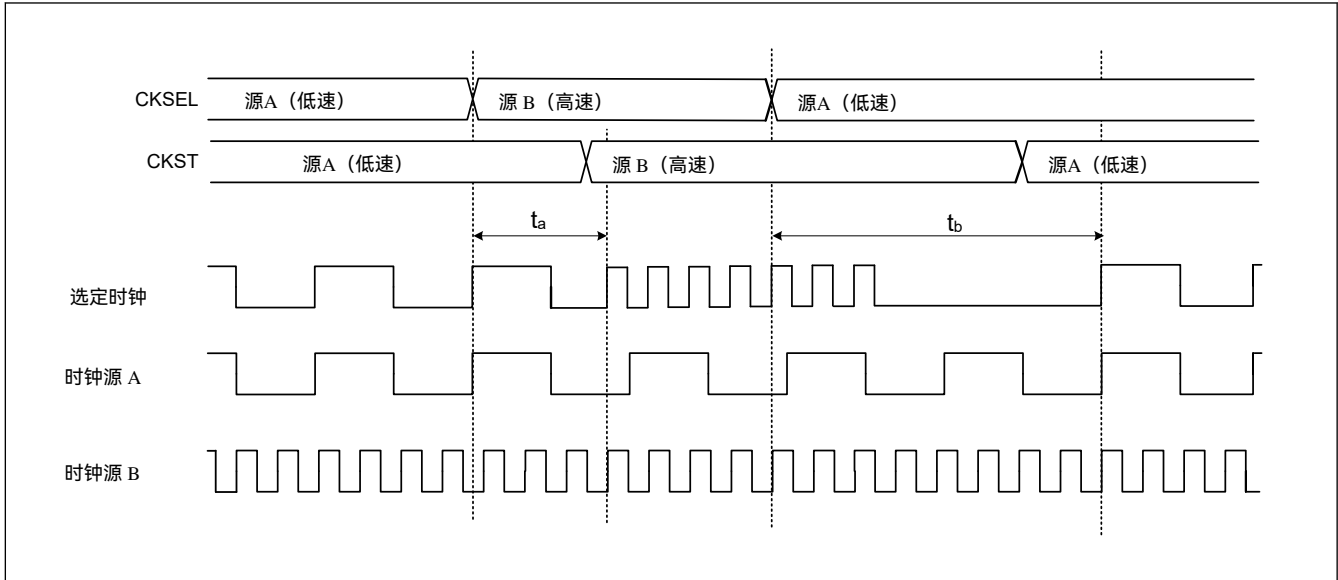


图 8.3 时钟源切换时序

- FMAIN↔FSUB t_a : 3 周期 t_b : $1 + 2 \times (\text{FMAIN 频率}) / (\text{FSUB 频率})$
周期所需的最大时钟周期数

8.3.2 IWDTClock (IWDTCCLK)

IWDTClock (IWDTCCLK) 是IWDTC的工作时钟。IWDTCCLK由LOCO时钟内部产生。

8.3.3 系统滴答定时器专用时钟 (SYSTICCLK)

SysTick定时器专用时钟SYSTICCLK是SysTick定时器的时钟。SYSTICCLK由LOCO时钟产生。

8.3.4 外部引脚输出时钟 (CLKOUT)

CLKOUT 输出到外部，用作时钟或蜂鸣器输出。当 CKS0.PCLOE 设置为 1 时，CLKOUT 输出到 PCLBUZ0 引脚。仅当 CKS0.PCLOE 位为 0 时，才更改 CKS0 中的 CSEL 或 CCS[2:0] 位的值。

CLKOUT 时钟频率由以下几位指定：

- CSEL 或 CCS[2:0] 在 CKS0 中
- HOCOFQR1[2:0] 在 OFS1
- DIV[2:0] 在 HOCODIV
- FSUBSCR 中的 CKSEL

8.4 使用说明

8.4.1 注册访问

- 在下列情况下，请勿写入本节列出的寄存器：
[寄存器]
 - FSUBSCR, HOCODIV
[条件]
 - ICLKSCR.CKSEL = 1 (ICLK = LOCO)

8.4.2 Notes on Clock Generation Circuit

To ensure correct processing after the clock frequency changes, first modify to the relevant Clock Control register to change the frequency, then read the value from the register, and finally perform the subsequent processing.

8.4.2 时钟生成电路的注释

为确保时钟频率改变后能够正确处理，首先修改相关的时钟控制寄存器以改变频率，然后从寄存器中读取该值，最后执行后续处理。

9. Low Power Modes

9.1 Overview

The MCU provides several functions for reducing power consumption, such as setting clock dividers, stopping modules, selecting power control mode in normal mode, and transitioning to low power modes.

Table 9.1 lists the specifications of the low power mode functions. Table 9.2 lists the conditions to transition to low power modes, the states of the CPU and peripheral modules, and the method for canceling each mode. After a reset, the MCU enters the program execution state, but only the DTC and SRAM operate.

Table 9.1 Specifications of the low power mode functions

Item	Specification
Reducing power consumption by switching clock signals	The frequency division ratio can be selected for HOCO*1
Module stop	Functions can be stopped independently for each peripheral module
Low power modes	- Sleep mode - Software Standby mode - Snooze mode
Power control modes	Power consumption can be reduced in Normal, Sleep, and Snooze mode by selecting an appropriate operating power control mode according to the operating frequency and voltage. Four operating power control modes are available: - High-speed mode - Middle-speed mode - Low-speed mode - Subosc-speed mode

Note 1. For details, see section 8, Clock Generation Circuit.

Table 9.2 Operating conditions of each low power mode (1 of 2)

Item	Sleep mode	Software Standby mode	Snooze mode*1
Transition condition	When [Condition 1] or [Condition 2] while SBYCR.SSBY = 0 [Condition 1] - WFI instruction - A valid interrupt request*9 is not accepted by the CPU. (including a transition from the time WFI instruction is executed to the time the transition to Sleep mode is completed) [Condition 2] - SCR.SLEEPONEXIT = 1 - Complete execution of all exception handlers - A valid interrupt request*9 is not accepted by the CPU. (including a transition from the time WFI instruction is executed to the time the transition to Sleep mode is completed)	When [Condition 1] or [Condition 2] while SBYCR.SSBY = 1 [Condition 1] - WFI instruction - A valid interrupt request*9 is not accepted by the CPU. (including a transition from the time WFI instruction is executed to the time the transition to Software Standby mode is completed) [Condition 2] - SCR.SLEEPONEXIT = 1 - Complete execution of all exception handlers - A valid interrupt request*9 is not accepted by the CPU. (including a transition from the time WFI instruction is executed to the time the transition to Software Standby mode is completed)	Snooze request in Software Standby mode.
Canceling method	All interrupts. Any reset available in the mode.	Interrupts shown in Table 9.3. Any reset available in the mode.	Interrupts shown in Table 9.3. Any reset available in the mode.
State after cancellation by an interrupt	Program execution state	Program execution state	Program execution state
State after cancellation by a reset	Reset state	Reset state	Reset state
High-speed on-chip oscillator	Selectable	Stop	Operating
Low-speed on-chip oscillator	Selectable	Selectable	Selectable

9.低功耗模式

9.1 概述

MCU 提供了多种降低功耗的功能，例如设置时钟分频器、停止模块、在正常模式下选择电源控制模式以及过渡到低功耗模式。

表 9.1 列出了低功耗模式功能的规格。表 9.2 列出了切换到低功耗模式的条件、CPU 和外设模块的状态以及取消每种模式的方法。复位后，MCU 进入程序执行状态，但只有 DTC 和 SRAM 工作。

表 9.1 低功耗模式功能的规格

物品	规格
通过切换时钟信号来降低功耗	HOCO*1 的频率分频比可以选择
模块停止	可以针对每个外围模块独立停止功能。
低功耗模式	- 睡眠模式 - 软件待机模式 - 贪睡模式
电源控制模式	在正常模式、睡眠模式和贪睡模式下，根据工作频率和电压选择合适的工作功率控制模式，可以降低功耗。 提供四种运行功率控制模式： - 高速模式 - 中速模式 - 低速模式 - 子振荡器速度模式

注1：详情请参见第8节“时钟生成电路”。

表 9.2 各低功耗模式的运行条件（1/2）

物品	睡眠模式	软件待机模式	贪睡模式*1
过渡条件	当[条件 1] 或[条件 2] 时，SBYCR.SSBY= 0[条件 1] - WFI指令 - CPU 不接受有效的中断请求*9。 （包括从执行 WFI 指令到完成睡眠模式转换的整个过程）[条件 2] - SCR.SLEEPONEXIT = 1 - 所有异常处理程序完全执行完毕 - CPU 不接受有效的中断请求*9。 （包括从执行 WFI 指令到完成进入睡眠模式的转换过程）	当[条件 1] 或[条件 2] 时，SBYCR.SSBY= 1[条件 1] - WFI指令 - CPU 不接受有效的中断请求*9。 （包括从执行 WFI 指令到完成向软件待机模式转换的整个过程）[条件 2] - SCR.SLEEPONEXIT = 1 - 所有异常处理程序完全执行完毕 - CPU 不接受有效的中断请求*9。 （包括从执行 WFI 指令到完成向软件待机模式转换的整个过程）	软件待机模式下的延迟请求。
取消方法	所有中断。此模式下可用的任何复位。	表 9.3 中所示的中断。该模式下有任何可用的重置方法。	表 9.3 中所示的中断。该模式下有任何可用的重置方法。
中断取消后的状态	程序执行状态	程序执行状态	程序执行状态
重置后的状态	重置状态	重置状态	重置状态
高速片上振荡器	可选	停止	操作
低速片上振荡器	可选	可选	可选

Table 9.2 Operating conditions of each low power mode (2 of 2)

Item	Sleep mode	Software Standby mode	Snooze mode*1
Clock/buzzer output function	Selectable	Selectable*2	Selectable*2
CPU	Stop (Retained)	Stop (Retained)	Stop (Retained)
SRAM	Selectable	Stop (Retained)	Selectable
Flash memory	Selectable*5	Stop (Retained)	Selectable*5
Data Transfer Controller (DTC)	Selectable	Stop (Retained)	Selectable
Independent Watchdog Timer (IWDT)	Selectable*3	Selectable*3	Selectable*3
32-bit Interval Timer (TML32)	Selectable	Selectable*4	Selectable*4
10-bit A/D Converter (ADC10)	Selectable	Stop (Retained)	Selectable*8
Serial Array Unit (SAU0)	Selectable	Stop (Retained)	Selectable*6
I2C Bus Interface (IICA0)	Selectable	Selectable*7	Selectable*7
NMI, IRQn (n = 0 to 5) pin interrupt	Selectable	Selectable	Selectable
Low Voltage Detection (LVD)	Selectable	Selectable	Selectable
Power-on reset circuit	Operating	Operating	Operating
Other peripheral modules	Selectable	Stop (Retained)	Operation prohibited
I/O ports	Operating	Retained	Operating

Note: Selectable means that operating or not operating can be selected by the control registers.
Stop (Retained) means that the contents of the internal registers are retained but the operations are suspended. Operation prohibited means that the function must be stopped before entering Software Standby mode.
Otherwise, proper operation is not guaranteed in Snooze mode.

Note 1. All modules whose module-stop bits are 0 start as soon as ICLK are supplied after entering Snooze mode.
To avoid an increasing power consumption in Snooze mode, set the module-stop bit of modules that are not required in Snooze mode to 1 before entering Software Standby mode.

Note 2. Stopped when the Clock Output Source Select bit (CKS0.CSEL) is set to a value other than 1 (FSUB).

Note 3. IWDT, operating or stopping is selected by setting the IWDT Stop Control bit (IWDTSTPCTL) in Option Function Select register 0 (OFS0) in IWDT auto start mode.

Note 4. TML32 operation is possible when 100b (FSXP) is selected in the ITLCSEL0.CSEL[2:0] bits.

Note 5. Flash, operating or stopping is selected by setting the Flash Stop Control bit (SBYCR.FLSTP).

Note 6. When using the Serial Array Unit 0 (SAU0) in Snooze mode, the SSC0.SWC bit must be 1.

Note 7. Only wakeup interrupt is available.

Note 8. When using the 10-bit A/D Converter (ADC10) in Snooze mode, the ADM2.AWC bit must be 1.

Note 9. Valid interrupt requests are any interrupts or exceptions that are not masked by the priority level of current exception and the priority level set by BASEPRI. In addition, if the interrupt request is based on IELSRn, the interrupt must be enabled by NVIC_ISERn.

Table 9.3 Available interrupt sources to transition to Normal mode from Snooze mode and Software Standby mode

Interrupt source	Name	Software Standby mode	Snooze mode
NMI		Yes	Yes
Port	PORT_IRQn (n = 0 to 5)	Yes	Yes
LVD	LVD_LVD1	Yes	Yes
IWDT	IWDT_NMIUNDF	Yes	Yes
TML32	TML32_ITL_OR	Yes	Yes
IICA	IICA0_TXRXI	Yes	Yes
ADC10	ADC10_ADI	No	Yes
SAU0	SAU0_SPI_TXRXI00	No	Yes
	SAU0_UART_ERRI0	No	Yes
	SAU0_UART_RXI0	No	Yes
DTC	DTC_COMPLETE	No	Yes

表 9.2 各低功耗模式的运行条件 (2/2)

物品	睡眠模式	软件待机模式	贪睡模式*1
时钟/蜂鸣器输出功能	可选	可选*2	可选*2
CPU	停止（保留）	停止（保留）	停止（保留）
SRAM	可选	停止（保留）	可选
闪存	可选*5	停止（保留）	可选*5
数据传输控制器（DTC）	可选	停止（保留）	可选
独立看门狗定时器 (IWDT)	可选*3	可选*3	可选*3
32 位间隔定时器 (TML32)	可选	可选*4	可选*4
10 位模数转换器 (ADC10)	可选	停止（保留）	可选*8
串行阵列单元 (SAU0)	可选	停止（保留）	可选*6
I2C总线接口（IICA0）	可选	可选*7	可选*7
NMI, IRQn(n = 0 至 5) 引脚中断	可选	可选	可选
低电压检测（LVD）	可选	可选	可选
上电复位电路	操作	操作	操作
其他外围模块	可选	停止（保留）	禁止操作
I/O端口	操作	保留	操作

注：可选择是指可以通过控制寄存器选择是否运行。
“停止（保留）”表示内部寄存器的内容将被保留，但操作将被暂停。“禁止操作”表示必须先停止该功能才能进入软件待机模式。

否则，在睡眠模式下无法保证正常运行。

注 1. 所有模块停止位为 0 的模块在进入休眠模式后，一旦提供 ICLK，就会立即启动。
为避免在睡眠模式下功耗增加，在进入软件待机模式之前，将睡眠模式下不需要的模块的模块停止位设置为 1。

注 2. 当时钟输出源选择位 (CKS0.CSEL) 设置为 1 以外的值 (FSUB) 时停止。

注 3. IWDT 的运行或停止是通过在 IWDT 自动启动模式下设置选项功能选择寄存器 0 (OFS0) 中的 IWDT 停止控制位 (IWDTSTPCTL) 来选择的。

注 4. 当在 ITLCSEL0.CSEL[2:0] 位中选择 100b (FSXP) 时，TML32 操作是可能的。

注 5. 闪光灯的运行或停止是通过设置闪光灯停止控制位 (SBYCR.FLSTP) 来选择的。

注 6. 当使用串行阵列单元 0 (SAU0) 处于休眠模式时，SSC0.SWC 位必须为 1。

注 7: 仅唤醒中断可用。

注 8. 当在休眠模式下使用 10 位 A/D 转换器 (ADC10) 时，ADM2.AWC 位必须为 1。

注 9: 有效的中断请求是指任何未被当前异常的优先级和BASEPRI设置的优先级所屏蔽的中断或异常。此外，如果中断请求基于IELSRn，则必须通过NVIC_ISERn启用该中断。

表 9.3 从休眠模式和软件待机模式转换到正常模式的可用中断源

中断源	Name	软件待机模式	贪睡模式
NMI		Yes	Yes
港口	PORT_IRQn (n = 0 to 5)	Yes	Yes
LVD	LVD_LVD1	Yes	Yes
IWDT	IWDT_NMIUNDF	Yes	Yes
TML32	TML32_ITL_OR	Yes	Yes
IICA	IICA0_TXRXI	Yes	Yes
ADC10	ADC10_ADI	No	Yes
SAU0	SAU0_SPI_TXRXI00	No	Yes
	SAU0_UART_ERRI0	No	Yes
	SAU0_UART_RXI0	No	Yes
DTC	DTC_COMPLETE	No	Yes

Figure 9.1 shows the transition between Normal mode to low power mode.

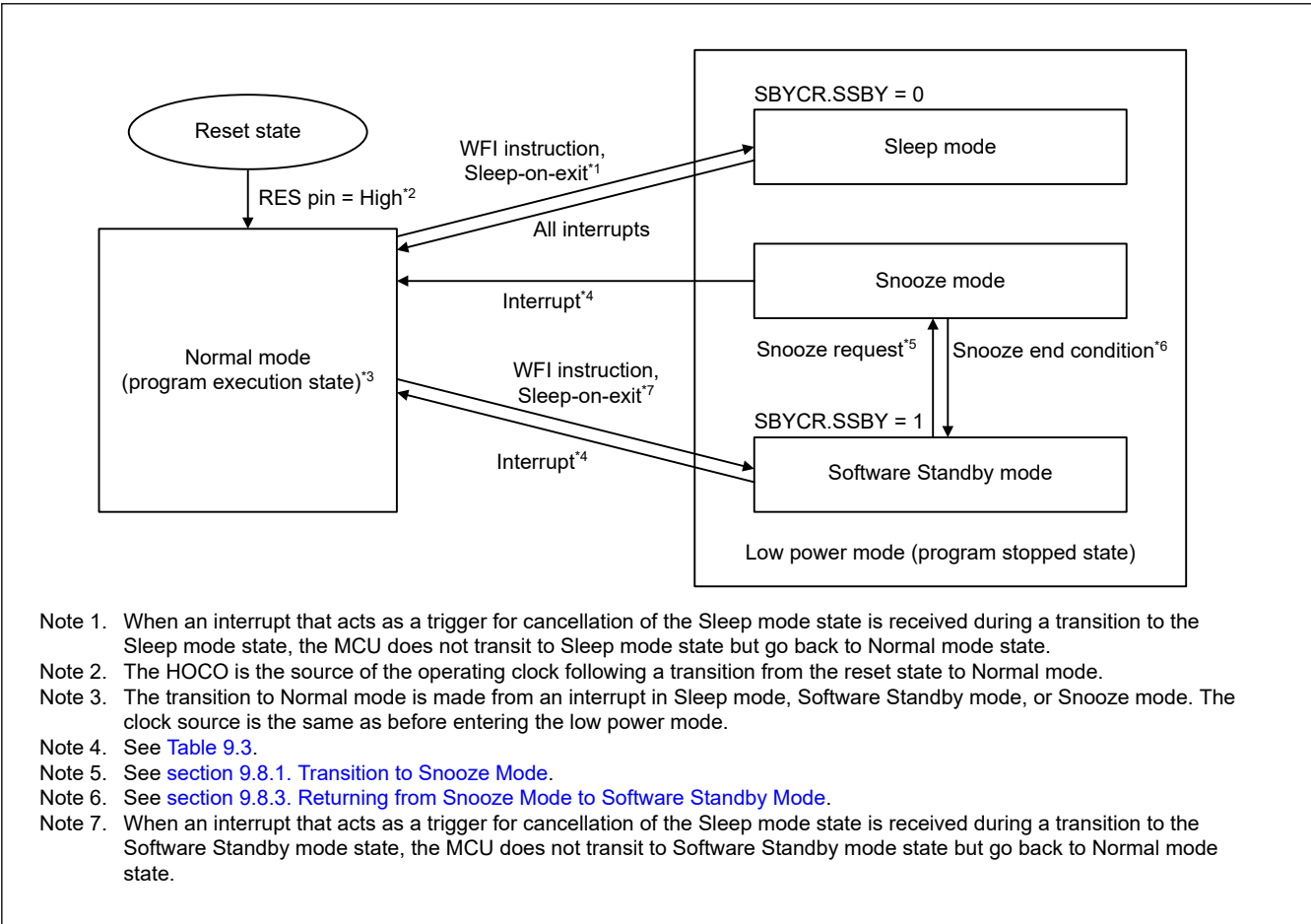


Figure 9.1 Low power mode transitions

9.2 Register Descriptions

9.2.1 SBYCR : Standby Control Register

Base address: SYSC = 0x4001_E000																
Offset address: 0x0860																
Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	SSBY	—	—	—	—	—	—	—	FLSTP	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
6:0	—	These bits are read as 0. The write value should be 0.	R/W
7	FLSTP	Flash Mode in Sleep Mode or in Snooze Mode 0: Flash active 1: Flash stop	R/W
14:8	—	These bits are read as 0. The write value should be 0.	R/W
15	SSBY	Software Standby Mode Select 0: Sleep mode 1: Software Standby mode.	R/W

Note: Set the PRCR.PRC1 bit to 1 (write enabled) before rewriting this register.

图 9.1 显示了正常模式到低功耗模式的转换。

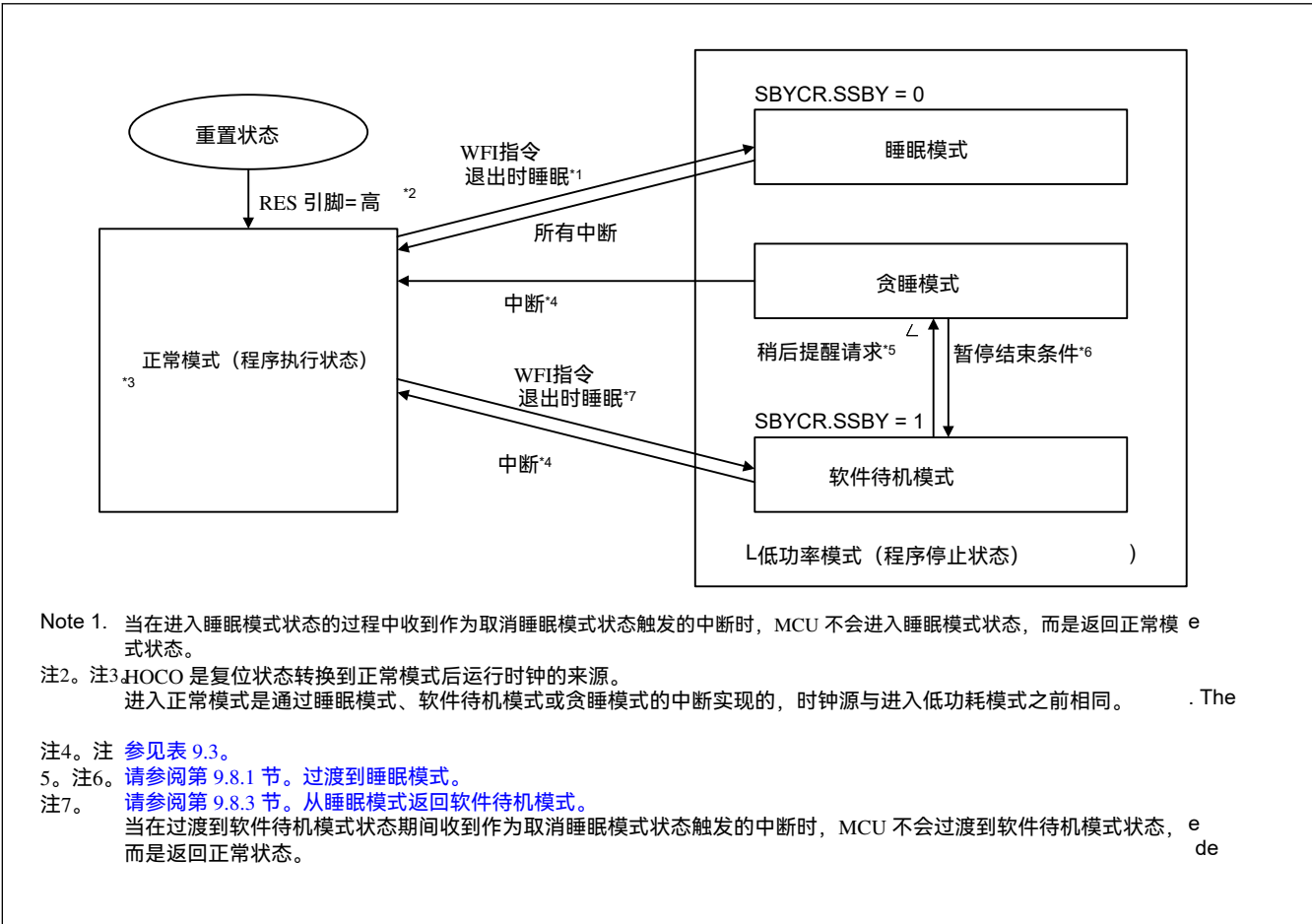


图 9.1 低功耗模式转换

9.2 寄存器描述

9.2.1 SBYCR：待机控制寄存器

基地址: SYSC = 0x4001_E000

偏移地址: 0x0860

位位置:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域:	SSBY	—	—	—	—	—	—	—	FLSTP	—	—	—	—	—	—	—
重置后的值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
6:0	—	这些位读取为 0。写入值应为 0。	R/W
7	FLSTP	睡眠模式或贪睡模式下的闪光模式 0: 闪光灯已激活 1: 停止闪光	R/W
14:8	—	这些位读取为 0。写入值应为 0。	R/W
15	SSBY	软件待机模式选择 0: 睡眠模式 1: 软件待机模式。	R/W

注意: 在重写此寄存器之前，请将 PRCR.PRC1 位设置为 1（使能写入）。

FLSTP bit (Flash Mode in Sleep Mode or in Snooze Mode)

When the FLSTP bit is set to 1, power consumption can be reduced because Flash stops during Sleep mode or Snooze mode. Instead, the Sleep mode release or Snooze mode release time will be extended. Also flash programming/erasure is not possible.

SSBY bit (Software Standby Mode Select)

The SSBY bit specifies the transition destination after a WFI instruction is executed.

When the SSBY bit is set to 1, the MCU enters Software Standby mode after execution of a WFI instruction. When the MCU returns to Normal mode from Software Standby mode by an interrupt, the SSBY bit remains 1. The SSBY bit can be cleared by writing 0 to it.

While the FENTRYR.FENTRY0 bit is 1 setting of the SSBY bit is ignored. Even if SSBY bit is 1, the MCU enters Sleep mode on execution of a WFI instruction.

9.2.2 MSTPCRA : Module Stop Control Register A

Base address:	SYSC = 0x4001_E000															
Offset address:	0x0C02															
Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	—	—	—	—	—	—	—	—	—	MSTP A22	—	—	—	—	—	—
Value after reset:	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1

Bit	Symbol	Function	R/W
5:0	—	These bits are read as 1. The write value should be 1.	R/W
6	MSTPA22	DTC Module Stop*1 0: Cancel the module-stop state 1: Enter the module-stop state	R/W
15:7	—	These bits are read as 1. The write value should be 1.	R/W

Note 1. When rewriting the MSTPA22 bit from 0 to 1, disable the DTC before setting the MSTPA22 bit.

9.2.3 MSTPCRB : Module Stop Control Register B

Base address:	MSTP = 0x4004_7000															
Offset address:	0x0000															
Bit position:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Bit field:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	—	—	—	—	—	MSTP B10	—	—	—	MSTP B6	—	—	—	—	—	—
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit	Symbol	Function	R/W
5:0	—	These bits are read as 1. The write value should be 1.	R/W
6	MSTPB6	Serial Array Unit 0 Module Stop Target module: SAU0 0: Cancel the module-stop state 1: Enter the module-stop state	R/W
9:7	—	These bits are read as 1. The write value should be 1.	R/W

FLSTP 位（睡眠模式或贪睡模式下的闪存模式）

当 FLSTP 位设置为 1 时，由于闪存存在睡眠模式或暂停模式下停止工作，因此可以降低功耗。取而代之的是，睡眠模式或暂停模式的释放时间将会延长。此外，闪存的编程/擦除操作也将无法进行。

SSBY 位（软件待机模式选择）

SSBY 位指定执行 WFI 指令后的转换目标。

当 SSBY 位设置为 1 时，MCU 在执行 WFI 指令后进入软件待机模式。当 MCU 通过中断从软件待机模式返回正常模式时，SSBY 位仍保持为 1。可以通过向 SSBY 位写入 0 来清除该位。

当 FENTRYR.FENTRY0 位为 1 时，SSBY 位的设置将被忽略。即使 SSBY 位为 1，MCU 在执行 WFI 指令时也会进入睡眠模式。

9.2.2 MSTPCRA: 模块停止控制寄存器 A

Base address:	SYSC = 0x4001_E000															
Offset address:	0x0C02															
位位置:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域:	—	—	—	—	—	—	—	—	—	MSTP A22	—	—	—	—	—	—
重置后的值:	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1

少量	象征	功能	R/W
5:0	—	这些位读取为 1。写入值应为 1。	R/W
6	MSTPA22	DTC模块停止*1 0: 取消模块停止状态 1: 进入模块停止状态	R/W
15:7	—	这些位读取为 1。写入值应为 1。	R/W

注 1. 当将 MSTPA22 位从 0 重写为 1 时，在设置 MSTPA22 位之前禁用 DTC。

9.2.3 MSTPCRB: 模块停止控制寄存器 B

基地址:	MSTP = 0x4004_7000															
偏移地址:	0x0000															
位位置:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
重置后的值:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
位位置:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域:	—	—	—	—	—	MSTP B10	—	—	—	MSTP B6	—	—	—	—	—	—
重置后的值:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

少量	象征	功能	R/W
5:0	—	这些位读取为 1。写入值应为 1。	R/W
6	MSTPB6	串行阵列单元 0 模块停止 目标模块: SAU0 0: 取消模块停止状态 1: 进入模块停止状态	R/W
9:7	—	这些位读取为 1。写入值应为 1。	R/W

Bit	Symbol	Function	R/W
10	MSTPB10	I ² C Bus Interface Module Stop Target module: IICA0 0: Cancel the module-stop state 1: Enter the module-stop state	R/W
31:11	—	These bits are read as 1. The write value should be 1.	R/W

9.2.4 MSTPCRC : Module Stop Control Register C

Base address: MSTP = 0x4004_7000																
Offset address: 0x0004																
Bit position:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Bit field:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MSTP C1	—
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit	Symbol	Function	R/W
0	—	This bit is read as 1. The write value should be 1.	R/W
1	MSTPC1	Cyclic Redundancy Check Calculator Module Stop Target module: CRC 0: Cancel the module-stop state 1: Enter the module-stop state	R/W
31:2	—	These bits are read as 1. The write value should be 1.	R/W

9.2.5 MSTPCRD : Module Stop Control Register D

Base address: MSTP = 0x4004_7000																																	
Offset address: 0x0008																																	
Bit position: 31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16																																	
Bit field: <table><tr><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>MSTP D16</td><td></td></tr></table>																	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MSTP D16	
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MSTP D16																		
Value after reset: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1																																	
Bit position: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																	
Bit field: <table><tr><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>MSTP D4</td><td>—</td><td>—</td><td>—</td><td>MSTP D0</td><td></td></tr></table>																	—	—	—	—	—	—	—	—	—	—	—	MSTP D4	—	—	—	MSTP D0	
—	—	—	—	—	—	—	—	—	—	—	MSTP D4	—	—	—	MSTP D0																		
Value after reset: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1																																	

Bit	Symbol	Function	R/W
0	MSTPD0	Timer Array Unit 0 Module Stop Target module: TAU0 0: Cancel the module-stop state 1: Enter the module-stop state	R/W
3:1	—	These bits are read as 1. The write value should be 1.	R/W
4	MSTPD4	32-bit Interval Timer Module Stop Target module: TML32 0: Cancel the module-stop state 1: Enter the module-stop state	R/W

少量	象征	功能	R/W
10	MSTPB10	I2C总线接口模块停止 目标模块: IICA0 0: 取消模块停止状态 1: 进入模块停止状态	R/W
31:11	—	这些位读取为 1。写入值应为 1。	R/W

9.2.4 MSTPCRC: 模块停止控制寄存器 C

基地址: MSTP = 0x4004_7000																																	
偏移地址: 0x0004																																	
位位置: 31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16																																	
位域: <table><tr><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td></tr></table>																	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—																	
重置后的值: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1																																	
位位置: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																	
位域: <table><tr><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>MSTP C1</td><td>—</td></tr></table>																	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MSTP C1	—
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MSTP C1	—																	
重置后的值: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1																																	

少量	象征	功能	R/W
0	—	该位读取值为 1。写入值应为 1。	R/W
1	MSTPC1	循环冗余校验计算器模块停止 目标模块: CRC 0: 取消模块停止状态 1: 进入模块停止状态	R/W
31:2	—	这些位读取为 1。写入值应为 1。	R/W

9.2.5 MSTPCRD: 模块停止控制寄存器 D

基地址: MSTP= 0x4004_7000																																	
偏移地址: 0x0008																																	
位位置: 31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16																																	
位域: <table><tr><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>MSTP D16</td></tr></table>																	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MSTP D16
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MSTP D16																	
重置后的值: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1																																	
位位置: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																																	
位域: <table><tr><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>MSTP D4</td><td>—</td><td>—</td><td>—</td><td>MSTP D0</td></tr></table>																	—	—	—	—	—	—	—	—	—	—	—	—	MSTP D4	—	—	—	MSTP D0
—	—	—	—	—	—	—	—	—	—	—	—	MSTP D4	—	—	—	MSTP D0																	
重置后的值: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1																																	

少量	象征	功能	R/W
0	MSTPD0	定时器阵列单元 0 模块停止 目标模块: TAU0 0: 取消模块停止状态 1: 进入模块停止状态	R/W
3:1	—	这些位读取为 1。写入值应为 1。	R/W
4	MSTPD4	32 位间隔定时器模块停止目标模块: TML32 0: 取消模块停止状态 1: 进入模块停止状态	R/W

Bit	Symbol	Function	R/W
15:5	—	These bits are read as 1. The write value should be 1.	R/W
16	MSTPD16	A/D Converter Module Stop Target module: ADC10 0: Cancel the module-stop state 1: Enter the module-stop state	R/W
31:17	—	These bits are read as 1. The write value should be 1.	R/W

9.2.6 FLMODE : Flash Operating Mode Control Register

Base address:	FLCN = 0x407E_C000							
Offset address:	0x020A							
Bit position:	7	6	5	4	3	2	1	0
Bit field:	MODE[1:0]	—	—	—	—	—	—	—
Value after reset:	1	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
5:0	—	These bits are read as 0. The write value should be 0.	R/W
7:6	MODE[1:0]	Operating Mode Select 0 0: Setting prohibited 0 1: Low-speed mode 1 0: Middle-speed mode 1 1: High-speed mode	R/W

Note: Direct transition between High-speed mode and Low-speed mode is prohibited (Writing 0x40 (0xC0) over 0xC0 (0x40) is ignored). The transition between High-speed mode and Low-speed mode should be made through the Middle-speed mode.

The FLMODE register is used to reduce power consumption in Normal mode, Sleep mode, and Snooze mode. Power consumption can be reduced according to the operating frequency and operating voltage used by the FLMODE setting. For the procedure to change the operating power control modes, see [section 9.5. Function for Lower Operating Power Consumption](#).

MODE[1:0] bits (Operating Mode Select)

The MODE[1:0] bits select the operating power control mode in Normal mode, Sleep mode, and Snooze mode.

Table 9.4 shows the relationship between the operating power control modes, the MODE[1:0], and ICLKSCR.CKSEL bits settings. Writing to FLMODE.MODE[1:0] is prohibited while MCU is under the following conditions:

1. The value of the FLMODE register can be changed when the FLMWEN bit in the flash operation mode protect register (FLMWRP) is 1. After the value of the FLMODE register is changed, set the FLMWEN bit to 0.
2. The MCU is in Sleep or Snooze mode, the MCU transitions to Normal mode from Sleep or Snooze mode, the MCU transitions to Sleep, Snooze, or Software Standby mode from Normal mode, or the MCU is in transfer state when in operating power mode.
3. Flash is in programming mode.
4. The MCU is in Subosc-speed mode (ICLKSCR.CKST bit and CKSEL bit is 1).

Table 9.4 Operating power control mode

Operating power control mode	MODE[1:0] bits	ICLKSCR.CKST bit	Power consumption
High-speed mode	11b	0	High ↓ ↓
Middle-speed mode	10b	0	
Low-speed mode	01b	0	
Subosc-speed mode	xxb	1	Low

少量	象征	功能	R/W
15:5	—	这些位读取为 1。写入值应为 1。	R/W
16	MSTPD16	A/D转换器模块停止 目标模块: ADC10 0: 取消模块停止状态 1: 进入模块停止状态	R/W
31:17	—	这些位读取为 1。写入值应为 1。	R/W

9.2.6FLMODE：闪存操作模式控制寄存器

基地址:		FLCN= 0x407E_C000													
偏移地址:		0x020A													
位位置:		7	6	5	4	3	2	1	0						
位域:		<table border="1"><tr><td>MODE[1:0]</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td></tr></table>							MODE[1:0]	—	—	—	—	—	—
MODE[1:0]	—	—	—	—	—	—									
重置后的值:		1	0	0	0	0	0	0	0						

少量	象征	功能	R/W
5:0	—	这些位读取为 0。写入值应为 0。	R/W
7:6	MODE[1:0]	操作模式选择 0 0: 设置禁止 0 1: 低速模式 1 0: 中速模式 1.1: 高速模式	R/W

注意：禁止在高速模式和低速模式之间直接转换（将0x40 (0xC0) 写入0xC0 (0x40) 将被忽略）。高速模式和低速模式之间的转换应该通过中速模式进行。

FLMODE寄存器用于降低正常模式、睡眠模式和贪睡模式下的功耗。功耗可根据FLMODE设置所使用的工作频率和工作电压进行降低。有关更改工作功耗控制模式的步骤，请参见9.5节“降低工作功耗的功能”。

模式[1:0] 位（操作模式选择）

MODE[1:0] 位用于选择正常模式、睡眠模式和贪睡模式下的电源控制模式。

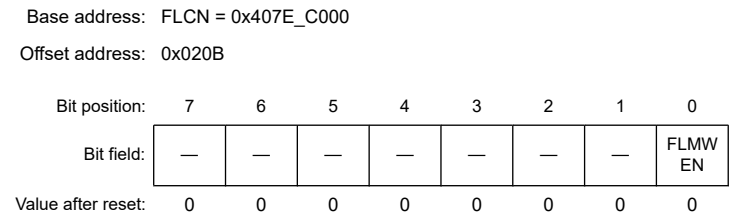
表 9.4 显示了工作功率控制模式、MODE[1:0]和 ICLKSCR.CKSEL 位设置之间的关系。当 MCU 处于以下条件时，禁止写入 FLMODE.MODE[1:0]：

1. 当闪存操作模式保护寄存器（FLMWRP）中的 FLMWEN 位为 1 时，可以更改 FLMODE 寄存器的值。更改 FLMODE 寄存器的值后，将 FLMWEN 位设置为 0。
2. MCU 处于睡眠或贪睡模式，MCU 从睡眠或贪睡模式过渡到正常模式，MCU 从正常模式过渡到睡眠、贪睡或软件待机模式，或者 MCU 在工作电源模式下处于传输状态。
3. Flash 处于编程模式。
4. MCU 处于子振荡器速度模式（ICLKSCR.CKST 位和 CKSEL 位为 1）。

表 9.4 运行功率控制模式

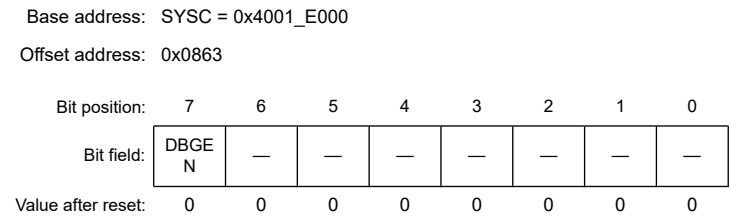
运行功率控制模式	模式[1:0] 位	ICLKSCR.CKST 位	功耗
高速模式	11b	0	高的 ↓ ↓
中速模式	10b	0	
低速模式	01b	0	
子振荡器速度模式	xxb	1	低的

9.2.7 FLMWRP : Flash Operating Mode Protect Register



Bit	Symbol	Function	R/W
0	FLMWEN	Control of Flash Operation Mode Select Register 0: Rewriting the FLMODE register is disabled 1: Rewriting the FLMODE register is enabled	R/W
7:1	—	These bits are read as 0. The write value should be 0.	R/W

9.2.8 SYOCDCCR : System Control OCD Control Register



Bit	Symbol	Function	R/W
6:0	—	These bits are read as 0. The write value should be 0.	R/W
7	DBGEN	Debugger Enable bit Set to 1 first in on-chip debug mode. 0: On-chip debugger is disabled 1: On-chip debugger is enabled	R/W

Note: Set the PRCR.PRC1 bit to 1 (write enabled) before rewriting this register.

DBGEN bit (Debugger Enable bit)

The DBGEN bit enables the on-chip debug mode. This bit must be set to 1 first in the on-chip debugger mode.

[Setting condition]

- Writing 1 to the bit when the debugger is connected.

[Clearing condition]

- Power-on reset is generated
- Writing 0 to the bit.

Note: Certain restrictions apply in terms of the MCU states in which the DBGEN bit can be set to 1. For details, see [section 2.7.3. Restrictions on Connecting an OCD emulator](#).

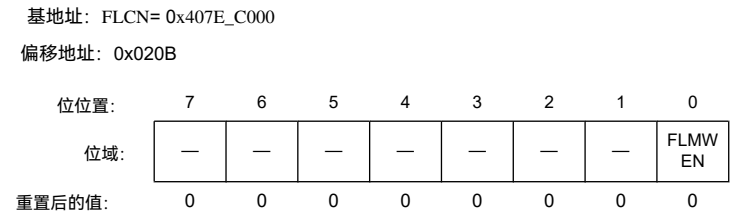
9.3 Reducing Power Consumption by Switching Clock Signals

The clock frequency changes when the HOCODIV register is set.

9.4 Module-stop Function

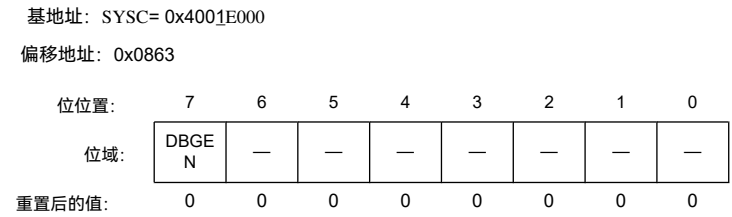
The module stop function can stop the clock supply set for each peripheral module.

9.2.7 FLMWRP: Flash 操作模式保护寄存器



少量	象征	功能	R/W
0	FLMWEN	闪存操作模式选择寄存器的控制 0: 已禁用重写 FLMODE 寄存器。 1: 已启用重写 FLMODE 寄存器。	R/W
7:1	—	这些位读取为 0。写入值应为 0。	R/W

9.2.8 SYOCDCCR: 系统控制 OCD 控制寄存器



少量	象征	功能	R/W
6:0	—	这些位读取为 0。写入值应为 0。	R/W
7	DBGEN	在片上调试模式下，首先将调试器使能位设置为 1。 0: 片上调试器已禁用；1: 片上调试器已启用。	R/W

注意: 在重写此寄存器之前，请将 PRCR.PRC1 位设置为 1（使能写入）。

DBGEN 位（调试器使能位）

DBGEN 位用于启用片上调试模式。在片上调试器模式下，必须首先将此位设置为 1。

[Setting condition]

- 当调试器连接时，向该位写入 1。

[清除条件]

- 上电复位信号
- 向该位写入0。

注意: 对于可以将 DBGEN 位设置为 1 的 MCU 状态，存在一些限制。有关详细信息，请参阅第 2.7.3 节“连接 OCD 模拟器的限制”。

9.3 通过切换时钟信号降低功耗

当 HOCODIV 寄存器置位时，时钟频率会发生变化。

9.4 模块停止功能

模块停止功能可以停止每个外围模块的时钟电源。

When the MSTPmi bit (m = A to D, i = 31 to 0) in MSTPCRn (n = A to D) is set to 1, the specified module stops operating and enters the module-stop state, but the CPU continues to operate independently. Setting the MSTPmi bit to 0 cancels the module-stop state, allowing the module to resume operation at the end of the bus cycle.

After a reset is canceled, all modules other than the DTC modules are placed in the module-stop state. Do not access the module while the corresponding MSTPmi bit is 1. Additionally, do not set 1 to the MSTPmi bit while the corresponding module is accessed.

9.5 Function for Lower Operating Power Consumption

By selecting an appropriate operating power consumption control mode according to the operating frequency, power consumption can be reduced in Normal mode, Sleep mode, and Snooze mode.

9.5.1 Setting Operating Power Control Mode

Ensure the operating condition such as the frequency range is always within the specified range before and after switching the operating power control modes.

This section provides example procedures for switching operating power control modes.

Table 9.5 Available oscillators in each mode

Mode	Oscillator	
	High-speed on-chip oscillator	Low-speed on-chip oscillator
High-speed	Available	Available
Middle-speed	Available	Available
Low-speed	Available	Available
Subosc-speed	N/A	Available

(1) Switching from a higher power mode to a lower power mode

Example 1: From High-speed mode to Middle-speed mode:

(Operation begins in High-speed mode)

1. Change the oscillator to what is used in Middle-speed mode. Set the frequency of each clock lower than or equal to the maximum operating frequency in Middle-speed mode.
2. Turn off the oscillator that is not required in Middle-speed mode.
3. Set the FLMWRP.FLMWEN bit to 1 (Rewriting the FLMODE register is enabled.).
4. Set the FLMODE.MODE[1:0] bits to 10b (Middle-speed mode).
5. Set the FLMWRP.FLMWEN bit to 0 (Rewriting the FLMODE register is disabled.).

(Operation is now in Middle-speed mode)

Example 2: From High-speed mode to Subosc-speed mode

(Operation begins in High-speed mode)

1. Switch the clock source to LOCO.
2. Turn off HOCO.

(Operation is now in Subosc-speed mode)

(2) Switching from a lower power mode to a higher power mode

Example 1: From Subosc-speed mode to High-speed mode*1

(Operation begins in Subosc-speed mode)

1. Turn on the required oscillator in High-speed mode.
2. Set the frequency of each clock lower than or equal to the maximum operating frequency for High-speed mode.

当MSTPCRn中的MSTPmi位 ((m = A至D、i = 31至0) 以及(n = A至D) 被设置为1时，指定的模块停止运行并进入模块停止状态，但CPU仍独立运行。将MSTPmi位设置为0可取消模块停止状态，允许模块在总线周期结束时恢复运行。

复位取消后，除DTC模块外的所有模块都将进入模块停止状态。当相应的MSTPmi位为1时，请勿访问该模块。此外，在访问相应模块时，请勿将MSTPmi位设置为1。

9.5 降低运行功耗的功能

通过根据运行频率选择合适的运行功耗控制模式，可以降低正常模式、睡眠模式和贪睡模式下的功耗。

9.5.1 设置运行功率控制模式

在切换运行功率控制模式前后，确保频率范围等运行条件始终在规定的范围内。

本节提供切换运行功率控制模式的示例步骤。

表 9.5 各模式下的可用振荡器

模式	振荡器	
	高速片上振荡器	低速片上振荡器
高速	可用的	可用的
中速	可用的	可用的
低速	可用的	可用的
亚振荡器速度	N/A	可用的

(1) 从大功率模式切换到低功率模式

示例 1：从高速模式切换到中速模式：

(以高速模式开始运行)

1. 将振荡器更换为中速模式下使用的振荡器。将每个时钟的频率设置为低于或等于中速模式下的最大工作频率。
2. 关闭中速模式下不需要的振荡器。
3. 将 FLMWRP.FLMWEN 位设置为 1（启用重写 FLMODE 寄存器）。
4. 将 FLMODE.MODE[1:0] 位设置为 10b（中速模式）。
5. 将 FLMWRP.FLMWEN 位设置为 0（禁用重写 FLMODE 寄存器）。

(当前运行模式为中速)

示例 2：从高速模式到次振荡器速度模式

(以高速模式开始运行)

1. 将时钟源切换到 LOCO。
2. 关闭 HOCO。

(当前操作处于亚振荡器速度模式)

(2) 从低功耗模式切换到更高功率模式

示例 1：从子振荡器速度模式切换到高速模式*1

(操作以亚振荡器速度模式开始)

1. 将所需的振荡器设置为高速模式。
2. 将每个时钟的频率设置为低于或等于高速模式下的最大工作频率。

(Operation is now in High-speed mode)

Note: When transitioning from Sub-speed mode to a higher speed mode, it can only revert to the mode that was active before entering Sub-speed mode.

Note 1. When the mode before entering Sub-speed mode was High-speed mode.

Example 2: From Middle-speed mode to High-speed mode

(Operation begins in Middle-speed mode)

1. Set the FLMWRP.FLMWEN bit to 1 (Rewriting the FLMODE register is enabled.).
2. Set the FLMODE.MODE[1:0] bits to 11b (High-speed mode).
3. Set the FLMWRP.FLMWEN bit to 0 (Rewriting the FLMODE register is disabled.).
4. Turn on any required oscillator in High-speed mode.
5. Set the frequency of each clock lower than or equal to the maximum operating frequency for High-speed mode.

(Operation is now in High-speed mode)

Note: There is an automatic wait time.

- Middle-speed to High-speed: 225 clocks
- Low-speed to Middle_speed: 10 clocks
- Middle-speed to Low-speed: 10 clocks
- High-speed to Middle-speed: 30 clocks
- other to Subosc-speed: 1+2 fMAIN/fSUB clock
- Subosc-speed to other: 3 clocks

9.5.2 Operating Range

Figure 9.2 to Figure 9.5 show the ICLK operating voltages and frequencies.

High-speed mode

The maximum operating frequency during a flash read is 32 MHz for ICLK. The operating voltage range during a flash read is 1.6 to 5.5 V. However, the maximum operating frequency during a flash read is 4 MHz when the operating voltage is 1.6 to 1.8 V.

During flash programming/erasure (P/E), the operating frequency range is 1 to 32 MHz and the operating voltage range is 1.8 to 5.5 V.

Figure 9.2 shows the operating voltages and frequencies in High-speed mode.

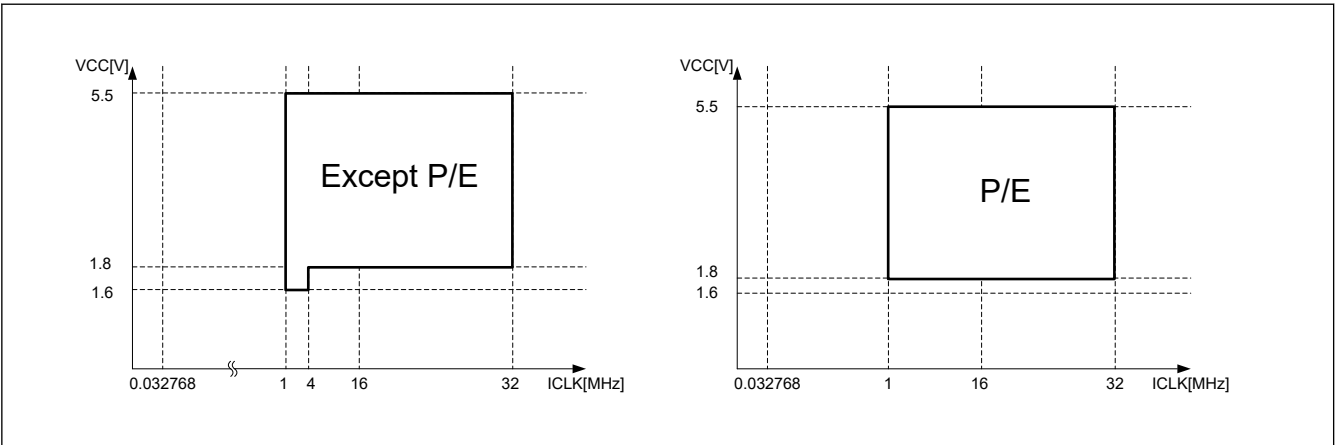


Figure 9.2 Operating voltages and frequencies in High-speed mode

Middle-speed mode

The power consumption of this mode is lower than that of High-speed mode under the same conditions.

(当前运行模式为高速模式)

注意：从低速模式过渡到高速模式时，只能恢复到进入低速模式之前所激活的模式。

注1. 进入降速模式之前的模式为高速模式。

示例 2：从中速模式到高速模式

(以中速模式运行)

1. 将 FLMWRP.FLMWEN 位设置为 1（启用重写 FLMODE 寄存器）。
2. 将 FLMODE.MODE[1:0] 位设置为 11b（高速模式）。
3. 将 FLMWRP.FLMWEN 位设置为 0（禁用重写 FLMODE 寄存器）。
4. 将所需的振荡器设置为高速模式。
5. 将每个时钟的频率设置为低于或等于高速模式下的最大工作频率。

(当前运行模式为高速模式)

注意：系统会自动设置等待时间。

- 中速至高速：225 时钟周期
- 低速到中速：10 个时钟周期
- 中速至低速：10 个时钟周期
- 高速到中速：30 个时钟周期
- 其他子振荡器速度：1+2 fMAIN/fSUB 时钟
- 子振荡器速度与其他频率：3 个时钟周期

9.5.2 工作范围

图 9.2 至图 9.5 显示了 ICLK 的工作电压和频率。

高速模式

闪存读取期间，ICLK 的最大工作频率为 32 MHz。闪存读取期间的工作电压范围为 1.6 至 5.5 V。但是，当工作电压为 1.6 至 1.8 V 时，闪存读取期间的最大工作频率为 4 MHz。

在闪存编程/擦除 (P/E) 过程中，工作频率范围为 1 至 32 MHz，工作电压范围为 1.8 至 5.5 V。

图 9.2 显示了高速模式下的工作电压和频率。

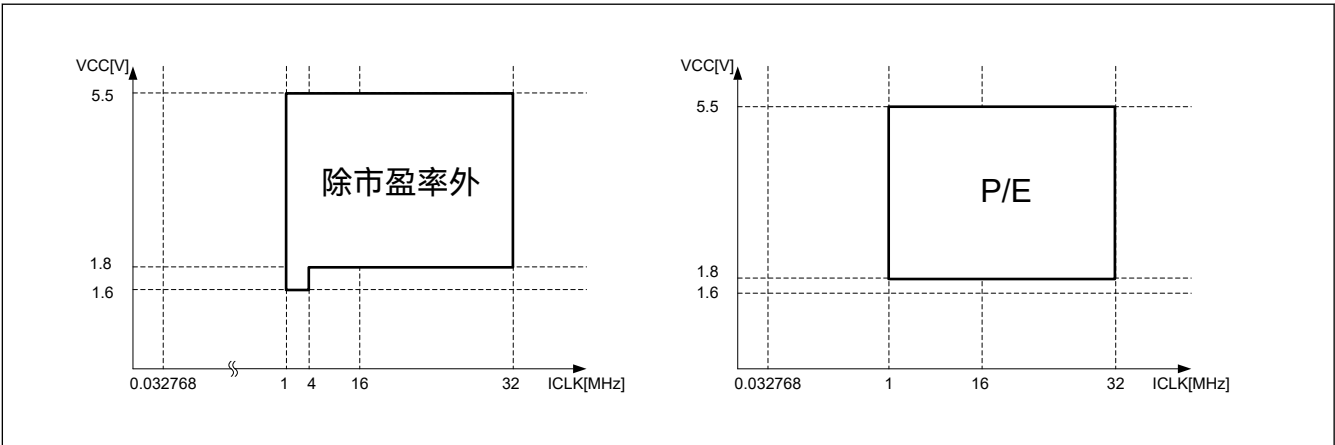


图 9.2 高速模式下的工作电压和频率

中速模式

在相同条件下，该模式的功耗低于高速模式。

The maximum operating frequency during a flash read is 16 MHz for ICLK. The operating voltage range during a flash read is 1.6 to 5.5 V. However, the maximum operating frequency during a flash read is 4 MHz when the operating voltage is 1.6 to 1.8 V.

During flash programming/erasure (P/E), the operating frequency range is 1 to 16 MHz and the operating voltage range is 1.8 to 5.5 V.

Figure 9.3 shows the operating voltages and frequencies in Middle-speed mode.

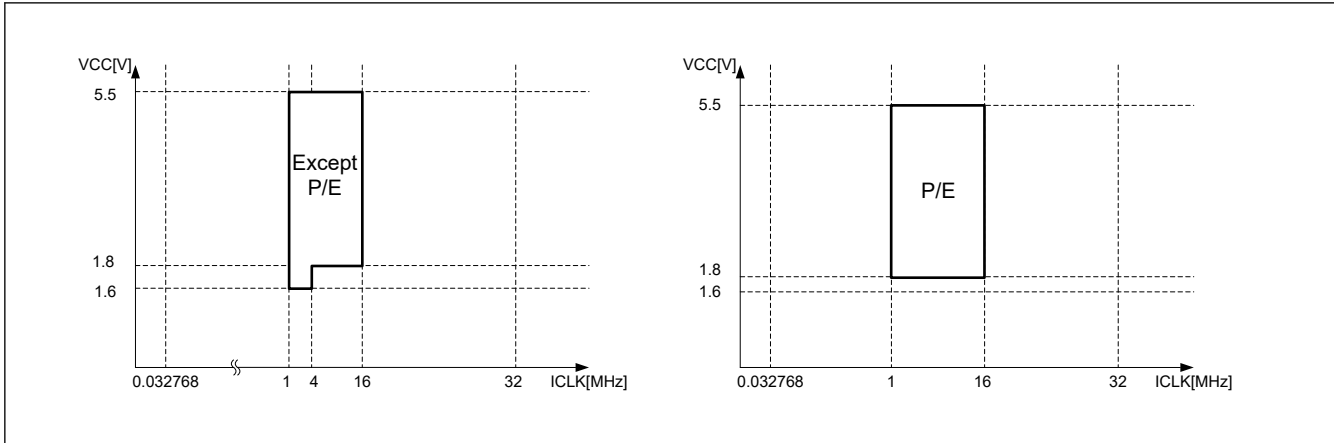


Figure 9.3 Operating voltages and frequencies in Middle-speed mode

Low-speed mode

The maximum operating frequency during a flash read is 2 MHz for ICLK. The operating voltage range during a flash read is 1.6 to 5.5 V. P/E operations for flash memory are prohibited.

Figure 9.4 shows the operating voltages and frequencies in Low-speed mode.

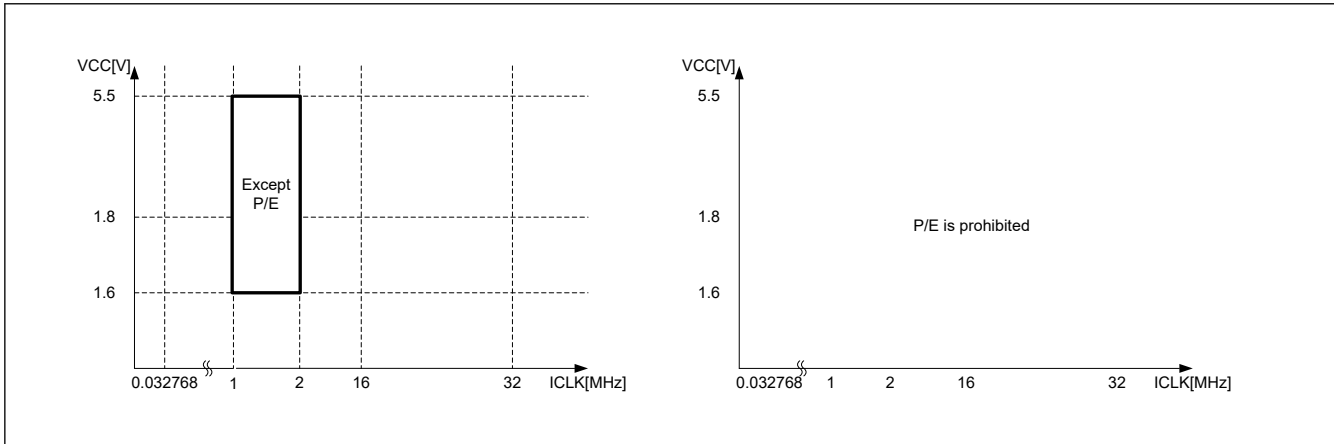


Figure 9.4 Operating voltages and frequencies in Low-speed mode

Subosc-speed mode

The maximum operating frequency during a flash read is 37.6832 kHz for ICLK. The operating voltage range during a flash read is 1.6 to 5.5 V. P/E operations for flash memory are prohibited.

Using the oscillators other than low-speed on-chip oscillator is prohibited.

Figure 9.5 shows the operating voltages and frequencies in Subosc-speed mode.

闪存读取期间，ICLK 的最大工作频率为16 MHz。闪存读取期间的工作电压范围为 1.6 至5.5 V。但是，当工作电压为 1.6 至1.8 V时，闪存读取期间的最大工作频率为4 MHz。

在闪存编程/擦除 (P/E) 过程中，工作频率范围为 1 至16 MHz，工作电压范围为 1.8 至 5.5 V。

图 9.3 显示了中速模式下的工作电压和频率。

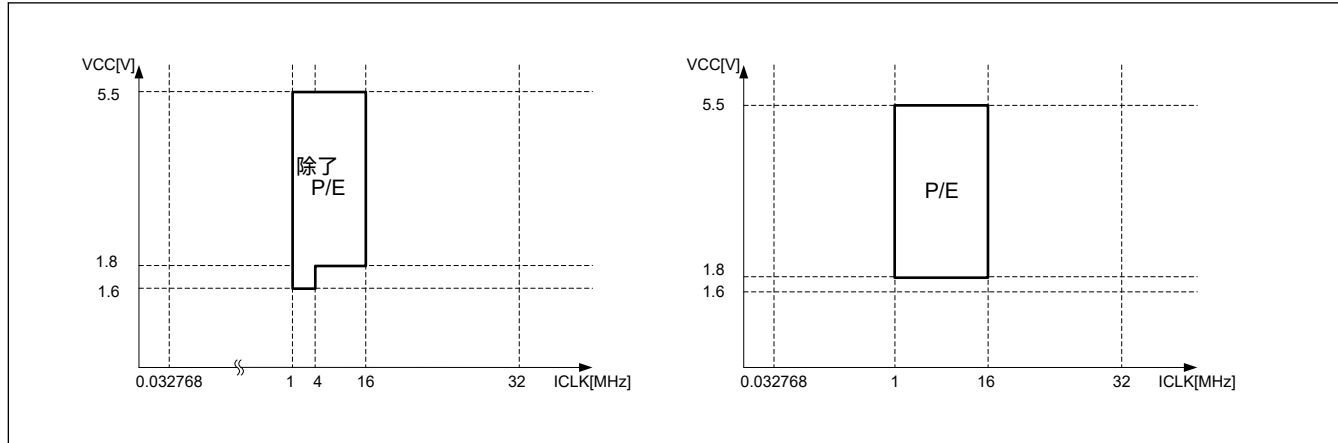


图 9.3 中速模式下的工作电压和频率

低速模式

闪存读取期间 ICLK 的最大工作频率为2 MHz。闪存读取期间的工作电压范围为 1.6 至5.5 V。禁止对闪存进行 P/E 操作。

图 9.4 显示了低速模式下的工作电压和频率。

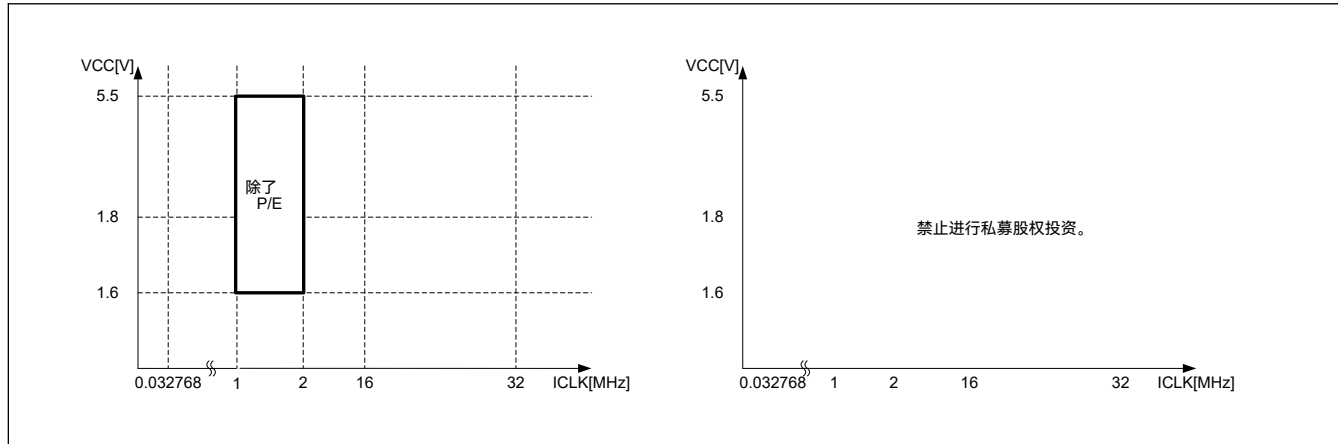


图 9.4 低速模式下的工作电压和频率

子振荡器速度模式

闪存读取期间 ICLK 的最大工作频率为37.6832 kHz。闪存读取期间的工作电压范围为 1.6 至5.5 V. P/E，禁止对闪存进行此类操作。

禁止使用除低速片上振荡器以外的振荡器。

图 9.5 显示了 Subosc 速度模式下的工作电压和频率。

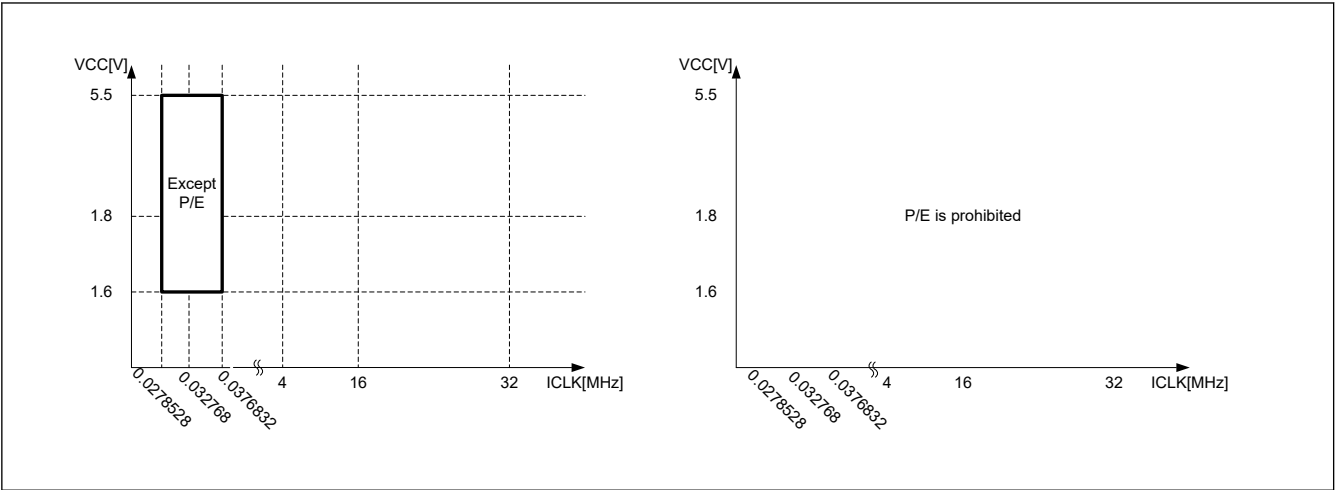


Figure 9.5 Operating voltages and frequencies in Subosc-speed mode

9.6 Sleep Mode

9.6.1 Transitioning to Sleep Mode

When a WFI instruction is executed while SBYCR.SSBY bit is 0, the MCU enters Sleep mode. In this mode, the CPU stops operating but the contents of its internal registers are retained. Other peripheral functions do not stop. Available resets or interrupts in Sleep mode cause the MCU to cancel Sleep mode. All interrupt sources are available.

Counting by IWDT stops when the MCU enters Sleep mode while the IWDT is in auto start mode and the OFS0.IWDTSTPCTL bit is 1 (IWDT stops in Sleep mode, Software Standby mode, or Snooze mode).

Counting by IWDT continues when the MCU enters Sleep mode while the IWDT is in auto start mode and the OFS0.IWDTSTPCTL bit is 0 (IWDT does not stop in Sleep mode, Software Standby mode, or Snooze mode).

9.6.2 Canceling Sleep Mode

Sleep mode is canceled by:

- An interrupt
- A RES pin reset
- A power-on reset
- A voltage monitor reset
- A reset caused by an IWDT underflow

The operations are as follows:

1. Canceling by an interrupt
When an interrupt request is generated, Sleep mode is canceled and the MCU starts the interrupt handling.
2. Canceling by RES pin reset
When the RES pin is driven low, the MCU enters the reset state. Be sure to keep the RES pin low for the time period specified in [section 27, Electrical Characteristics](#). When the RES pin is driven high after the specified time period, the CPU starts the reset exception handling.
3. Canceling by IWDT reset
 - Sleep mode is canceled by an internal reset generated by an IWDT underflow and the MCU starts the reset exception handling. However, IWDT stops in Sleep mode and an internal reset for canceling Sleep mode is not generated in the following conditions:
 - OFS0.IWDTSTRT = 0 and OFS0.IWDTSTPCTL = 1.
4. Canceling by other resets available in Sleep mode
Sleep mode is canceled by other resets and the MCU starts the reset exception handling.

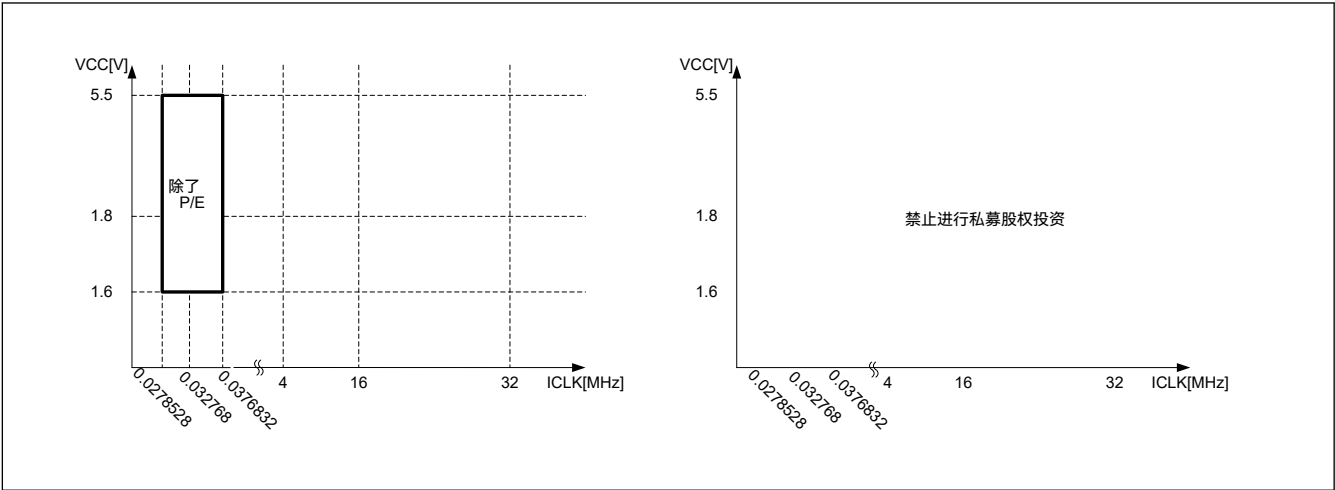


图 9.5 子振荡器速度模式下的工作电压和频率

9.6睡眠模式

9.6.1 过渡到睡眠模式

当 SBYCR.SSBY 位为 0 时执行 WFI 指令，MCU 将进入睡眠模式。在此模式下，CPU 停止运行，但其内部寄存器的内容会被保留。其他外设功能不会停止。睡眠模式下的复位或中断操作会将 MCU 取消睡眠模式。所有中断源均可用。

当 MCU 进入睡眠模式，而 IWDT 处于自动启动模式且 OFS0.IWDTSTPCTL 位为 1 时，IWDT 计数停止（IWDT 在睡眠模式、软件待机模式或贪睡模式下停止）。

当 MCU 进入睡眠模式时，IWDT 会继续计数，同时 IWDT 处于自动启动模式，并且 OFS0.IWDTSTPCTL 位为 0（IWDT 在睡眠模式、软件待机模式或贪睡模式下不会停止）。

9.6.2 取消睡眠模式

睡眠模式可通过以下方式取消：

- 中断
- 复位引脚 RES
- 上电重置
- 电压监控器复位
- 由 IWDT 下溢引起的重置

具体操作如下：

1. 通过中断取消
当产生中断请求时，睡眠模式被取消，MCU 开始处理中断。
2. Canceling by RES pin reset
当 RES 引脚被拉低时，MCU 进入复位状态。请务必将 RES 引脚保持低电平，持续时间如第 27 节“电气特性”中所述。当 RES 引脚在指定时间段后被拉高时，CPU 将启动复位异常处理。
3. 通过 IWDT 重置取消
 - 睡眠模式由 IWDT 下溢产生的内部复位取消，MCU 开始执行复位异常处理。但是，在以下情况下，IWDT 会在睡眠模式下停止，并且不会产生用于取消睡眠模式的内部复位：
 - OFS0.IWDTSTRT = 0 and OFS0.IWDTSTPCTL = 1.
4. 通过睡眠模式下的其他重置方法取消
睡眠模式被其他复位操作取消，MCU 开始复位异常处理。

Note: For details on proper setting of the interrupts, see [section 11, Interrupt Controller Unit \(ICU\)](#).

9.7 Software Standby Mode

9.7.1 Transition to Software Standby Mode

When a WFI instruction is executed while SBYCR.SSBY bit is 1, the MCU enters Software Standby mode. In this mode, the CPU, most of the on-chip peripheral functions and oscillators stop. However, the contents of the CPU internal registers and SRAM data, the states of on-chip peripheral functions and the I/O ports are retained. Software Standby mode allows a significant reduction in power consumption because most of the oscillators stop in this mode. [Table 9.2](#) shows the status of each on-chip peripheral functions and oscillators. Available resets or interrupts in Software Standby mode cause the MCU to cancel Software Standby mode. See [Table 9.3](#) for available interrupt sources and [section 11.2.14. SBYEDCR0 : Software Standby/Snooze End Control Register 0](#) and [section 11.2.15. SBYEDCR1 : Software Standby/Snooze End Control Register 1](#) for information on how to wake up the MCU from Software Standby mode. For details, see [section 11, Interrupt Controller Unit \(ICU\)](#).

Counting by IWDT stops when the MCU enters Software Standby mode while the IWDT is in auto start mode and the OFS0.IWDTSTPCTL bit is 1 (IWDT stops in Sleep mode, Software Standby mode, and Snooze mode). Counting by IWDT continues if the MCU enters Software Standby mode while the IWDT is in auto start mode and the OFS0.IWDTSTPCTL bit is 0 (IWDT does not stop in Sleep mode, Software Standby mode, and Snooze mode).

Do not enter Software Standby mode while the flash memory performs a programming or erasing procedure. To enter Software Standby mode, execute a WFI instruction after the programming or erasing procedure completes.

9.7.2 Canceling Software Standby Mode

Software Standby mode is canceled by:

- An available interrupt shown in [Table 9.3](#)
- A RES pin reset
- A power-on reset
- A voltage monitor reset
- A reset caused by an IWDT underflow.

After exiting Software Standby mode, the oscillators that were operating before the transition restart. After the oscillator set as the ICLK source clock has stabilized, the MCU returns from Software standby mode to normal mode. Oscillators that are not set as the ICLK source should wait for stabilization before using them. See [section 11.2.14. SBYEDCR0 : Software Standby/Snooze End Control Register 0](#) and [section 11.2.15. SBYEDCR1 : Software Standby/Snooze End Control Register 1](#) for information on how to wake up the MCU from Software Standby mode.

You can cancel Software Standby mode in any of the following ways:

1. Canceling by an interrupt
When an available interrupt request (see [Table 9.3](#)) is generated, an oscillator that operates before the transition to Software Standby mode restarts. After all the oscillators are stabilized, the MCU returns to Normal mode from Software Standby mode and starts the interrupt handling.
2. Canceling by a RES pin reset
When the RES pin is driven low, the MCU enters the reset state, and the oscillators whose default status is operating, start the oscillation. Be sure to keep the RES pin low for the time period specified in [section 27, Electrical Characteristics](#). When the RES pin is driven high after the specified time period, the CPU starts the reset exception handling.
3. Canceling by a power-on reset
Software Standby mode is canceled by a power-on reset and the MCU starts the reset exception handling.
4. Canceling by a voltage monitor reset
Software Standby mode is canceled by a voltage monitor reset from the voltage detection circuit and the MCU starts the reset exception handling.
5. Canceling by IWDT reset

注：有关中断的正确设置的详细信息，请参阅第 11 节“中断控制器单元 (ICU)”。

9.7 软件待机模式

9.7.1 过渡到软件待机模式

当 SBYCR.SSBY 位为 1 时执行 WFI 指令，MCU 将进入软件待机模式。在此模式下，CPU、大多数片上外设功能和振荡器将停止工作。但是，CPU 内部寄存器和 SRAM 数据的内容、片上外设功能的状态以及 I/O 端口的状态将被保留。由于大多数振荡器在此模式下停止工作，因此软件待机模式可以显著降低功耗。表 9.2 显示了每个片上外设功能和振荡器的状态。在软件待机模式下，可用的复位或中断将使 MCU 退出软件待机模式。有关可用的中断源，请参见表 9.3 和 11.2.14 节。SBYEDCR0：[有关如何从软件待机模式唤醒 MCU 的信息，请参阅第 11 节“中断控制器单元 \(ICU\)”。](#)[有关软件待机/休眠结束控制寄存器 0 和第 11.2.15 节“SBYEDCR1: 软件待机/休眠结束控制寄存器 1”的更多信息。](#)[有关详细信息，请参阅第 11 节“中断控制器单元 \(ICU\)”。](#)

当 MCU 进入软件待机模式且 IWDT 处于自动启动模式，并且 OFS0.IWDTSTPCTL 位为 1 时，IWDT 计数停止（IWDT 在睡眠模式、软件待机模式和贪睡模式下停止）。如果 MCU 进入软件待机模式且 IWDT 处于自动启动模式，并且 OFS0.IWDTSTPCTL 位为 0，则 IWDT 计数继续（IWDT 在睡眠模式、软件待机模式和贪睡模式下不停止）。

闪存执行编程或擦除操作时，请勿进入软件待机模式。要进入软件待机模式，请在编程或擦除操作完成后执行 WFI 指令。

9.7.2 取消软件待机模式

软件待机模式可通过以下方式取消：

- 表9.3中显示了一个可用的中断。
- 复位引脚 RES
- 上电重置
- 电压监控器复位
- 由 IWDT 下溢引起的复位。

退出软件待机模式后，转换前正在运行的振荡器将重新启动。当设置为 ICLK 时钟源的振荡器稳定后，MCU 将从软件待机模式返回正常模式。未设置为 ICLK 时钟源的振荡器应等待其稳定后再使用。有关如何从软件待机模式唤醒 MCU 的信息，请参阅 11.2.14 节“SBYEDCR0: 软件待机/休眠结束控制寄存器 0”和 11.2.15 节“SBYEDCR1: 软件待机/休眠结束控制寄存器 1”。

您可以通过以下任一方式取消软件待机模式：

1. **通过中断取消**
当生成可用中断请求（参见表 9.3）时，在进入软件待机模式之前运行的振荡器会重新启动。所有振荡器稳定后，MCU 从软件待机模式返回正常模式并开始处理中断。
2. **通过复位 RES 引脚取消**
当 RES 引脚被拉低时，MCU 进入复位状态，默认处于工作状态的振荡器开始振荡。请务必将 RES 引脚保持低电平，持续时间如第 27 节“电气特性”中所述。当 RES 引脚在指定时间段后被拉高时，CPU 将启动复位异常处理。
3. **通过开机重启取消**
软件待机模式通过上电复位取消，MCU 开始复位异常处理。
4. **通过电压监控器复位取消**
软件待机模式通过电压检测电路的电压监控复位取消，MCU 开始复位异常处理。

5. 通过 IWDT 重置取消

Software Standby mode is canceled by an internal reset generated by an IWDT underflow and the MCU starts the reset exception handling. However, IWDT stops in Software Standby mode and an internal reset for canceling Software Standby mode is not generated in the following condition:

- OFS0.IWDTSTRT = 0 and OFS0.IWDTSTPCTL = 1.

9.7.3 Example of Software Standby Mode Application

Figure 9.6 shows an example of entry to Software Standby mode on detection of a falling edge of the IRQn pin, and exit from Software Standby mode by a rising edge of the IRQn pin.

In this example, an IRQn pin interrupt is accepted with the IRQCRi.IRQMD[1:0] bits of the ICU set to 00b (falling edge) in Normal mode, and the IRQCRi.IRQMD[1:0] bits are set to 01b (rising edge). After that, the SBYCR.SSBY bit is set to 1 and a WFI instruction is executed. As a result, entry to Software Standby mode completes and exit from Software Standby mode is initiated by a rising edge of the IRQn pin.

Setting the ICU is also required to exit Software Standby mode. For details, see section 11, Interrupt Controller Unit (ICU). The oscillation stabilization time in Figure 9.6 is specified in section 27, Electrical Characteristics.

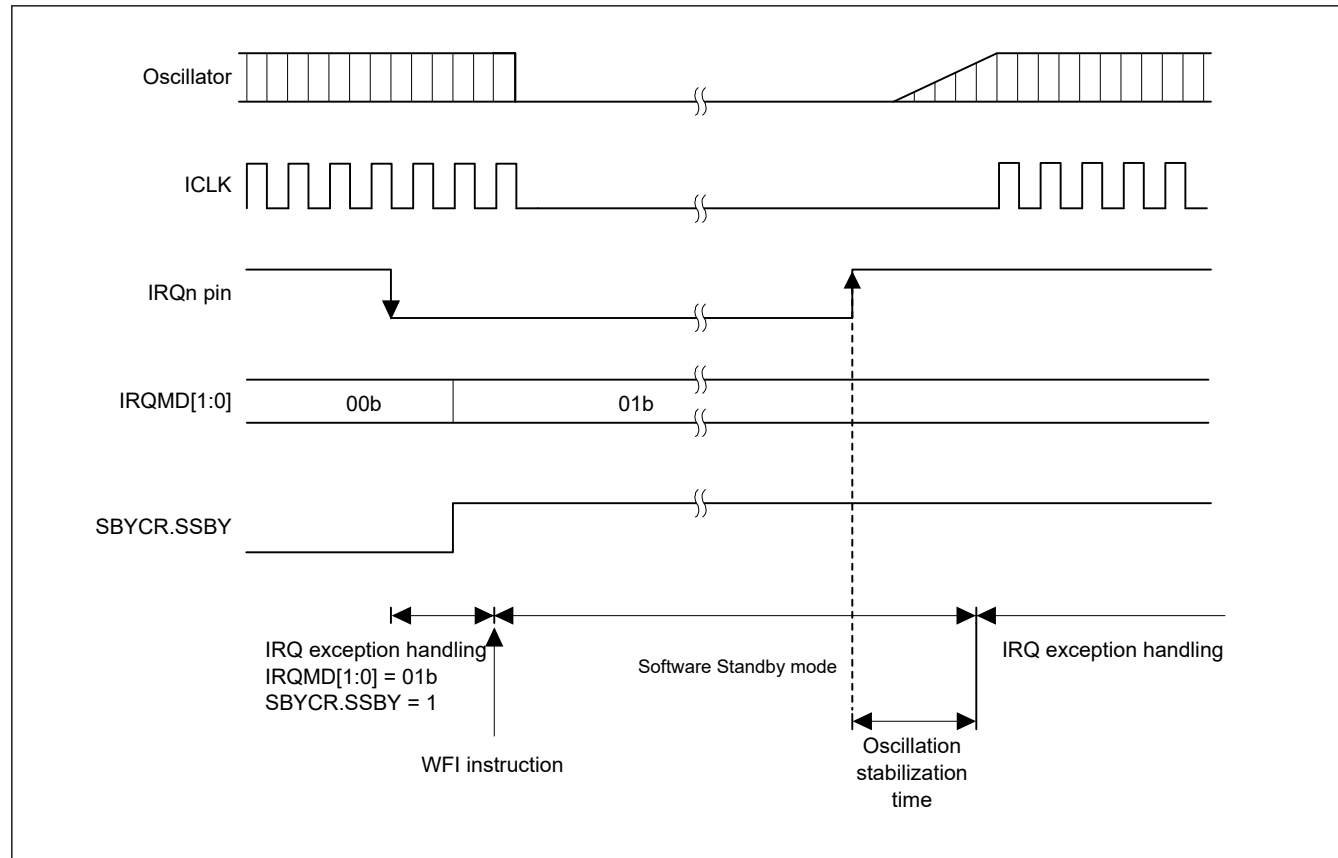


Figure 9.6 Example of Software Standby mode application

9.8 Snooze Mode

9.8.1 Transition to Snooze Mode

Figure 9.7 shows Snooze mode entry configuration. When the Snooze control circuit receives a Snooze request in Software Standby mode, the MCU transfers to Snooze mode. In this mode, some peripheral modules operate without waking up the CPU. Table 9.2 shows the peripheral modules that can operate in Snooze mode.

This can only be specified when the high-speed on-chip oscillator is selected for the CPU/peripheral hardware clock (ICLK).

软件待机模式因 IWDT 下溢产生的内部复位而取消，MCU 开始执行复位异常处理。然而，在以下情况下，IWDT 会在软件待机模式下停止，并且不会产生用于取消软件待机模式的内部复位：

- OFS0.IWDTSTRT = 0 and OFS0.IWDTSTPCTL = 1.

9.7.3 软件待机模式应用示例

图 9.6 显示了通过检测 IRQn 引脚的下降沿进入软件待机模式，以及通过 IRQn 引脚的上升沿退出软件待机模式的示例。

在本例中，IRQn 引脚中断被接收，此时 ICU 的 IRQCRi.IRQMD[1:0] 位在正常模式下被设置为 00b（下降沿），IRQCRi.IRQMD[1:0] 位被设置为 01b（上升沿）。之后，SBYCR.SSBY 位被设置为 1，并执行一条 WFI 指令。因此，进入软件待机模式完成，而退出软件待机模式则由 IRQn 引脚的上升沿触发。

要退出软件待机模式，还需要设置中断控制器单元 (ICU)。详情请参见第 11 节“中断控制器单元 (ICU)”。图 9.6 中的振荡稳定时间在第 27 节“电气特性”中规定。

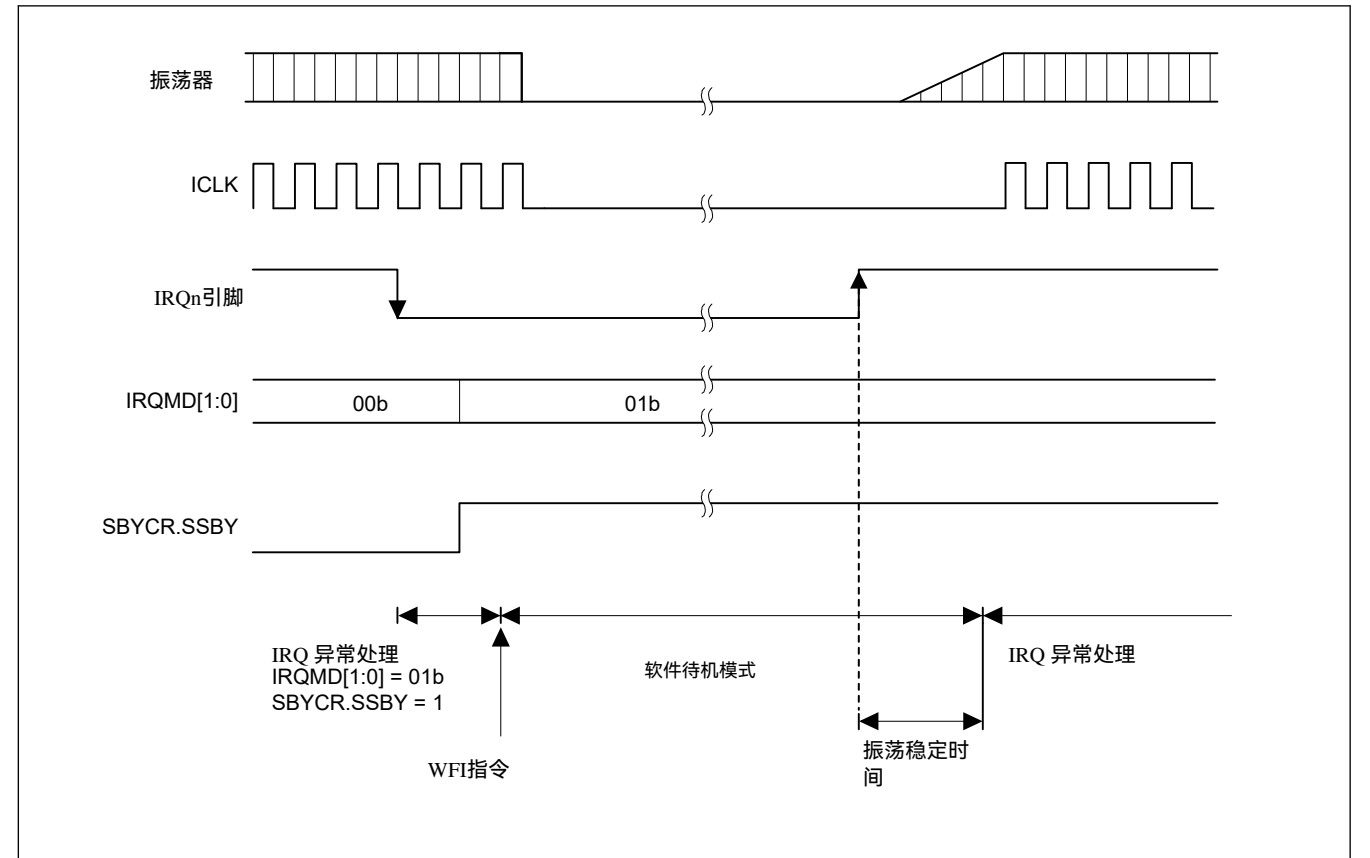


图 9.6 软件待机模式应用示例

9.8 贪睡模式

9.8.1 过渡到睡眠模式

图 9.7 显示了休眠模式的进入配置。当休眠控制电路在软件待机模式下接收到休眠请求时，MCU 将切换到休眠模式。在此模式下，部分外设模块无需唤醒 CPU 即可运行。表 9.2 列出了可在休眠模式下运行的外设模块。

只有当选择高速片上振荡器作为 CPU/外围硬件时钟 (ICLK) 时，才能指定这一点。

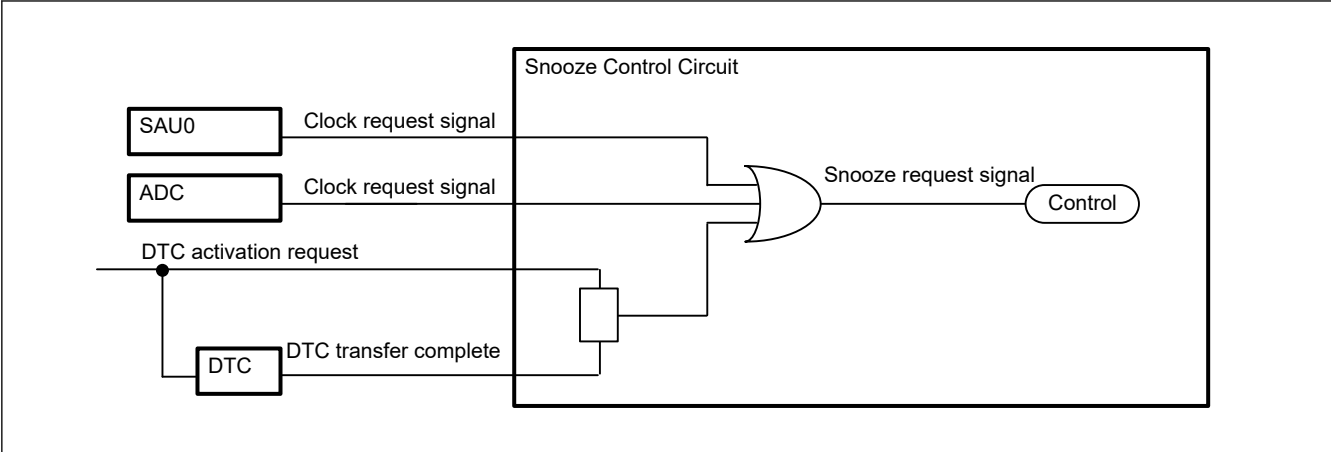


Figure 9.7 Snooze mode entry configuration

Table 9.6 shows the Snooze requests to switch the MCU from Software Standby mode to Snooze mode. To use the listed Snooze requests as a trigger to switch to Snooze mode.

Note: Do not enable multiple Snooze requests at the same time.

Table 9.6 Available Snooze requests to switch to Snooze mode

Snooze request output source	Control Register	
	Register	Bit
SAU0	SSC0	SWC
ADC10	ADM2	AWC
ICU (for DTC)	DTCENSTn	STm

Clear the DTCST.DTCST bit to 0 before executing a WFI instruction except when using DTC in Snooze mode. If DTC is required in Snooze mode, set the DTCST.DTCST bit to 1 before executing a WFI instruction.

9.8.2 Canceling Snooze Mode

Snooze mode is canceled by an interrupt request that is available in Software Standby mode or a reset. Table 9.3 shows the requests that can be used to exit each mode. After canceling the Snooze mode, the MCU enters Normal mode and proceeds with exception processing for the given interrupt or reset.

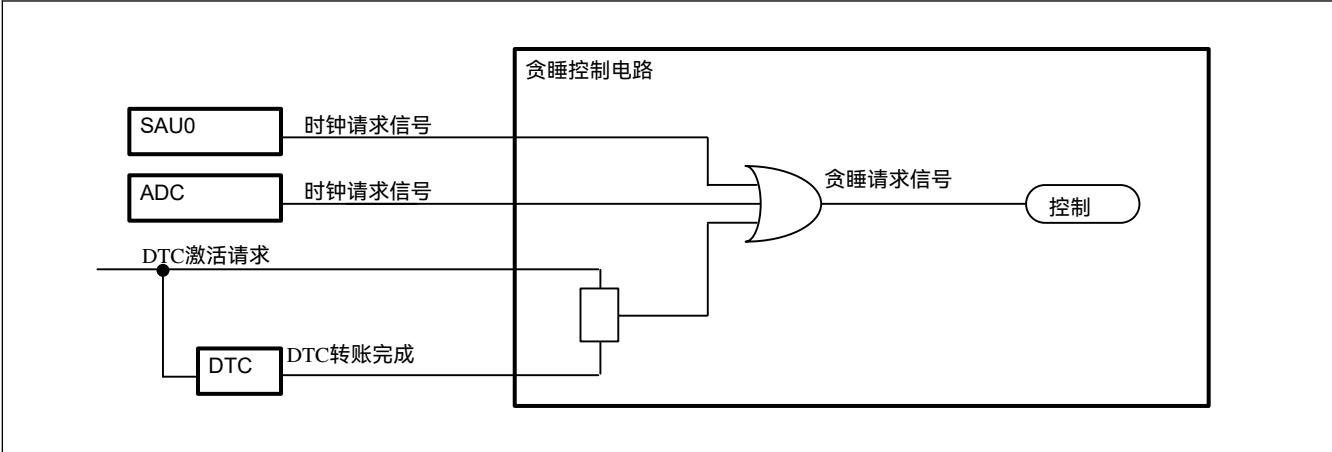


图 9.7 贪睡模式入口配置

表 9.6 显示了将 MCU 从软件待机模式切换到休眠模式的休眠请求。要使用列出的休眠请求作为切换到休眠模式的触发器，请执行以下操作。

注意：请勿同时启用多个延迟提醒请求。

表 9.6 可用的“稍后提醒”模式切换请求

暂停请求输出源	控制寄存器	
	登记	少量
SAU0	SSC0	SWC
ADC10	ADM2	AWC
ICU (适用于DTC)	DTCENSTn	STm

执行 WFI 指令前，除在休眠模式下使用 DTC 外，应将 DTCST.DTCST 位清零。如果休眠模式下需要使用 DTC，则在执行 WFI 指令前，应将 DTCST.DTCST 位设置为 1。

9.8.2 取消贪睡模式

软件待机模式下的中断请求或复位操作可以取消休眠模式。表 9.3 列出了可用于退出各种模式的请求。取消休眠模式后，MCU 进入正常模式，并继续处理给定中断或复位操作的异常。

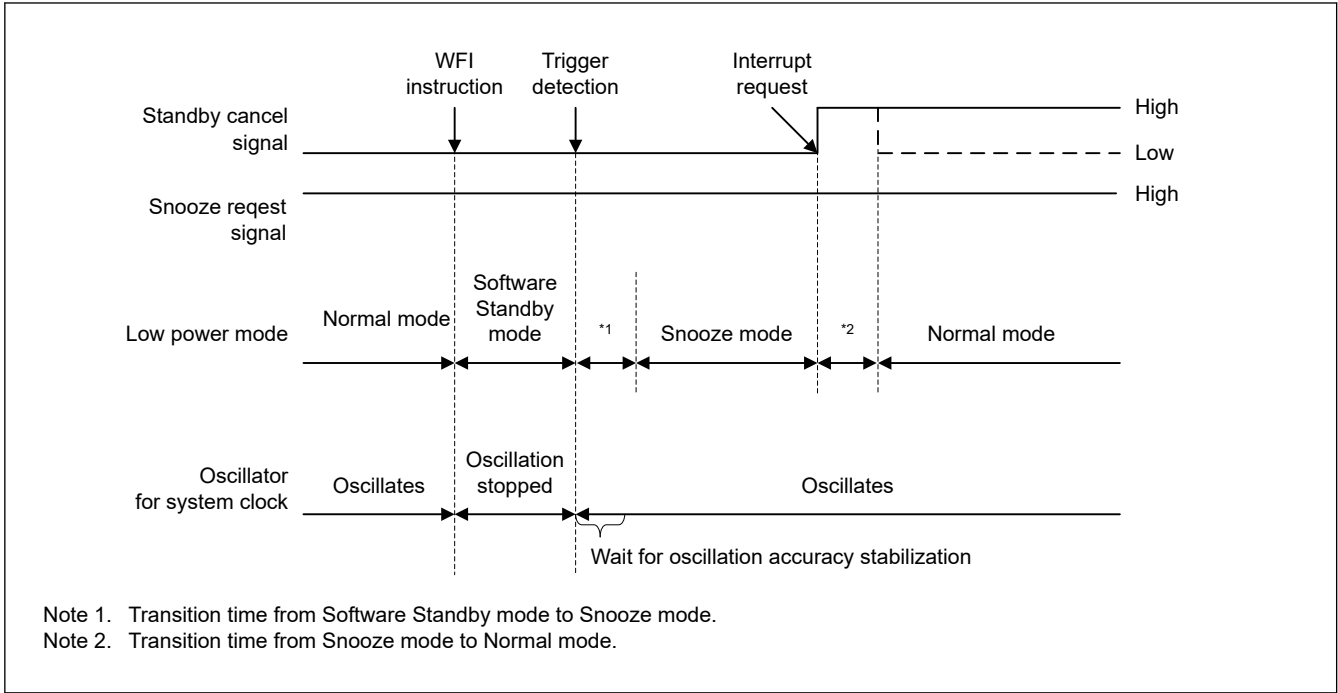


Figure 9.8 Canceling of Snooze mode when an interrupt request signal is generated

9.8.3 Returning from Snooze Mode to Software Standby Mode

Figure 9.9 shows the timing diagram for the transition from Snooze mode to Software Standby mode. This mode transition occurs cleared a Snooze request

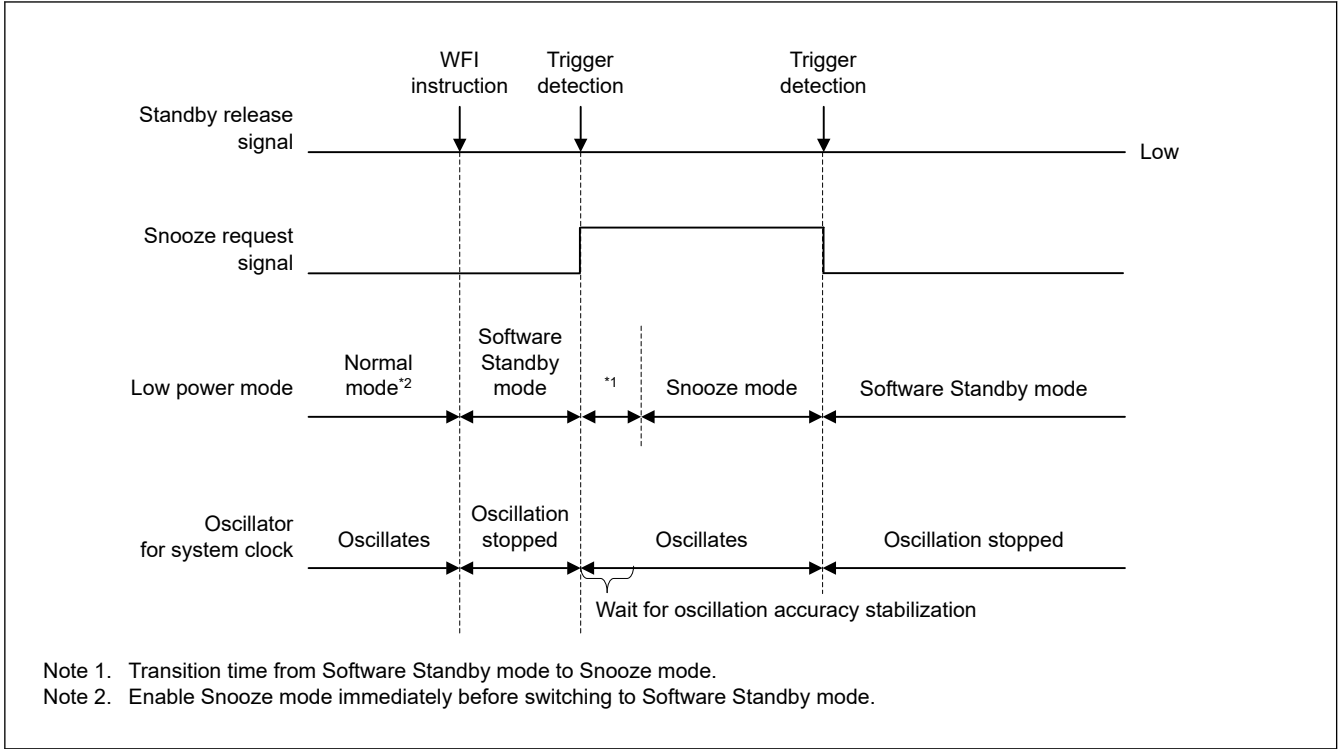


Figure 9.9 Canceling of Snooze mode when an interrupt request signal is not generated

9.8.4 Snooze Operation Example

Figure 9.10 shows an example setting for using DTC in Snooze mode.

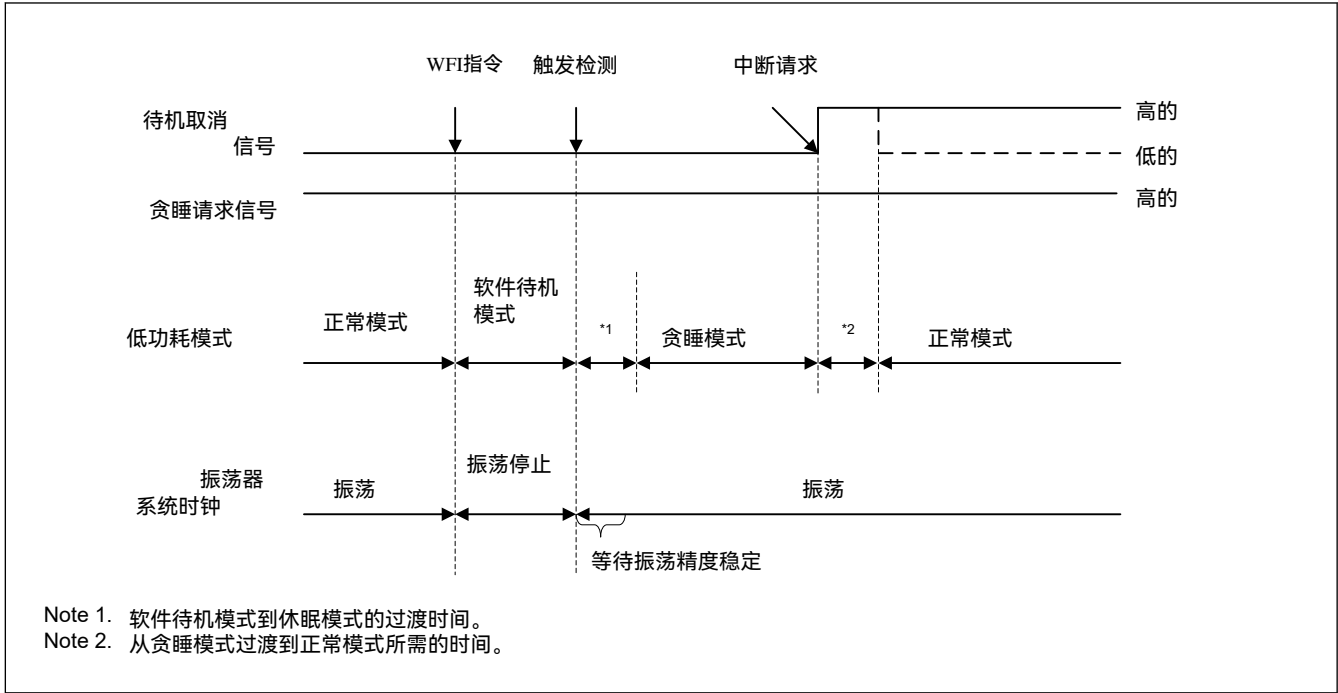


图 9.8 当产生中断请求信号时取消休眠模式

9.8.3 从休眠模式返回软件待机模式

图 9.9 显示了从休眠模式到软件待机模式转换的时序图。此模式转换发生在休眠请求被清除之后。

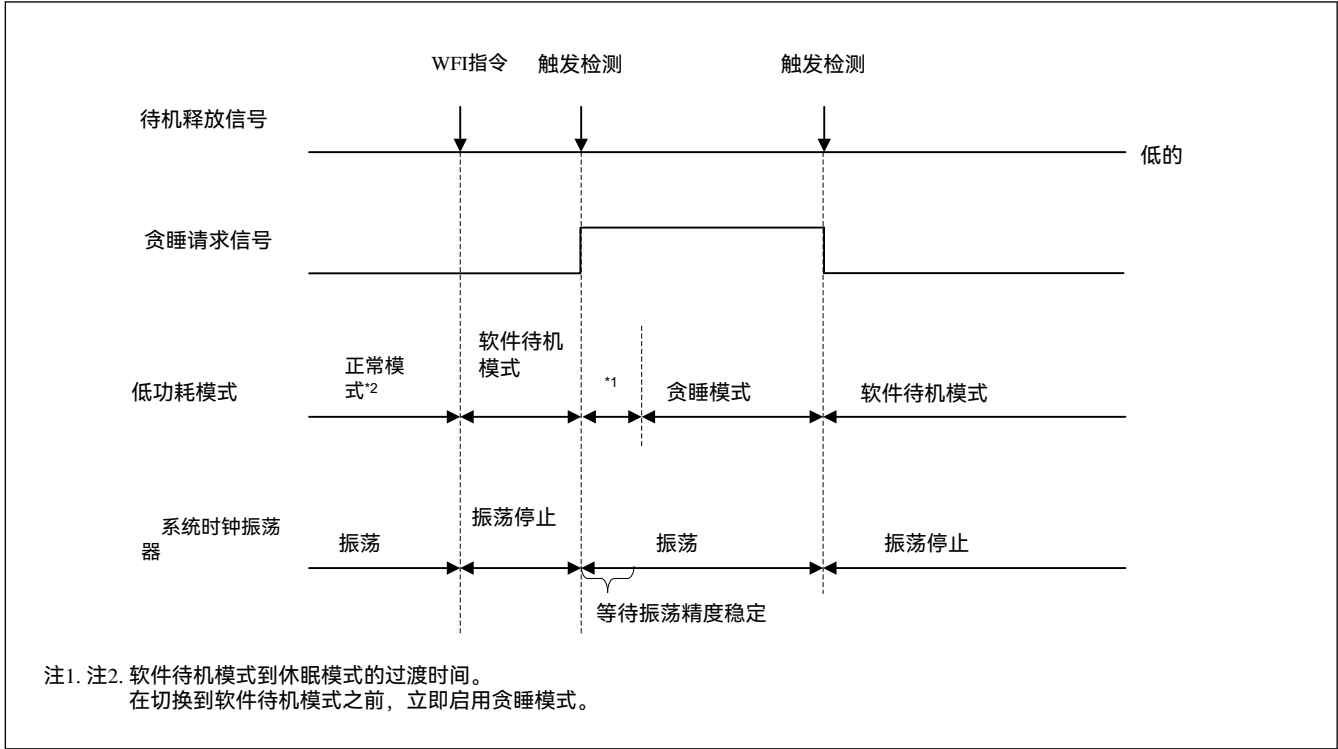


图 9.9: 当未生成中断请求信号时取消休眠模式

9.8.4 贪睡操作示例

图 9.10 显示了在睡眠模式下使用 DTC 的示例设置。

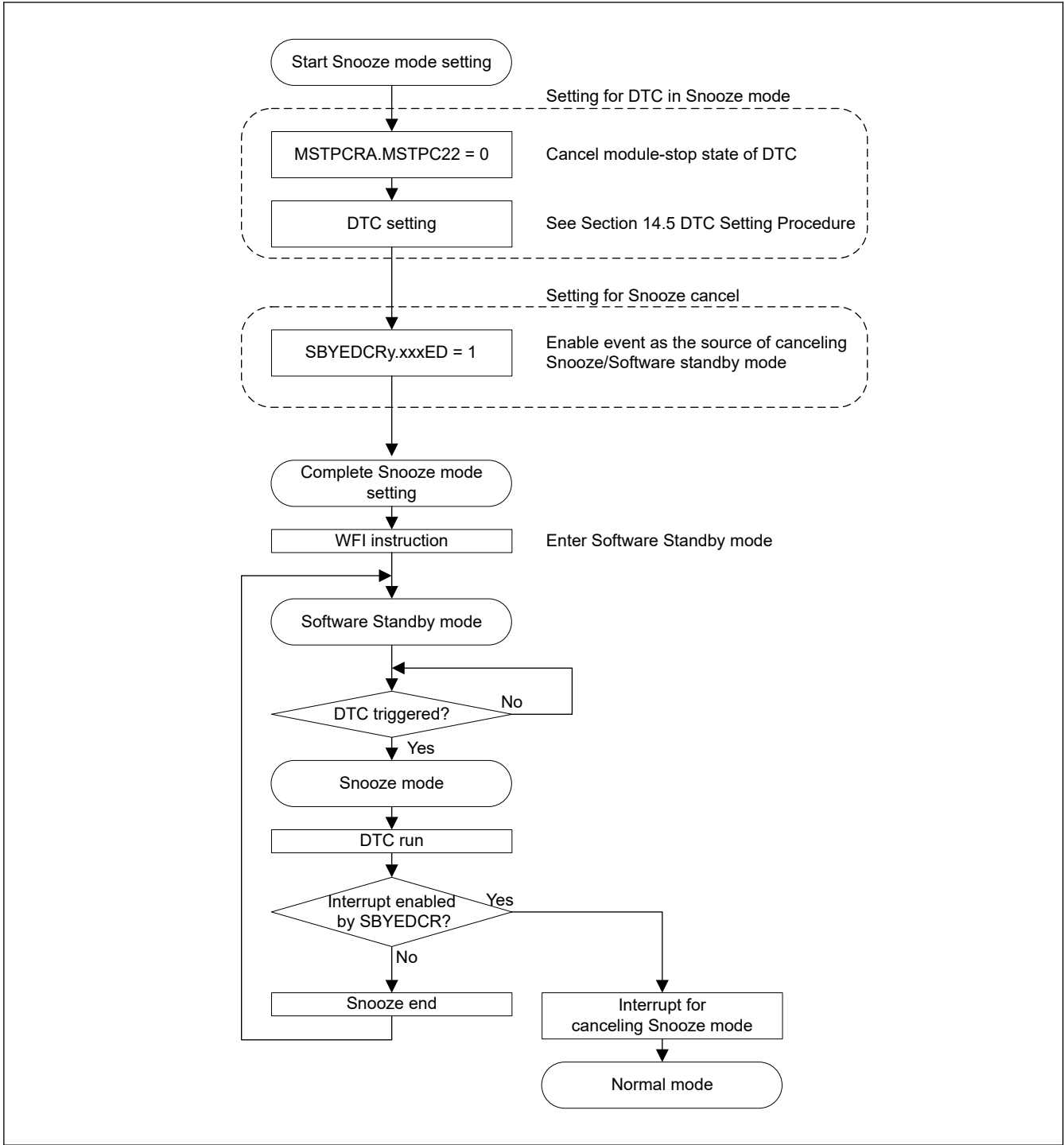


Figure 9.10 Setting example of using DTC in Snooze mode

9.9 Usage Notes

9.9.1 Register Access

(1) Do not write to registers listed in this section in any of the following conditions:

[Registers]

- All registers with a peripheral name of SYSTEM and the FLMODE register

[Conditions]

- During the time period from executing a WFI instruction to returning to Normal mode

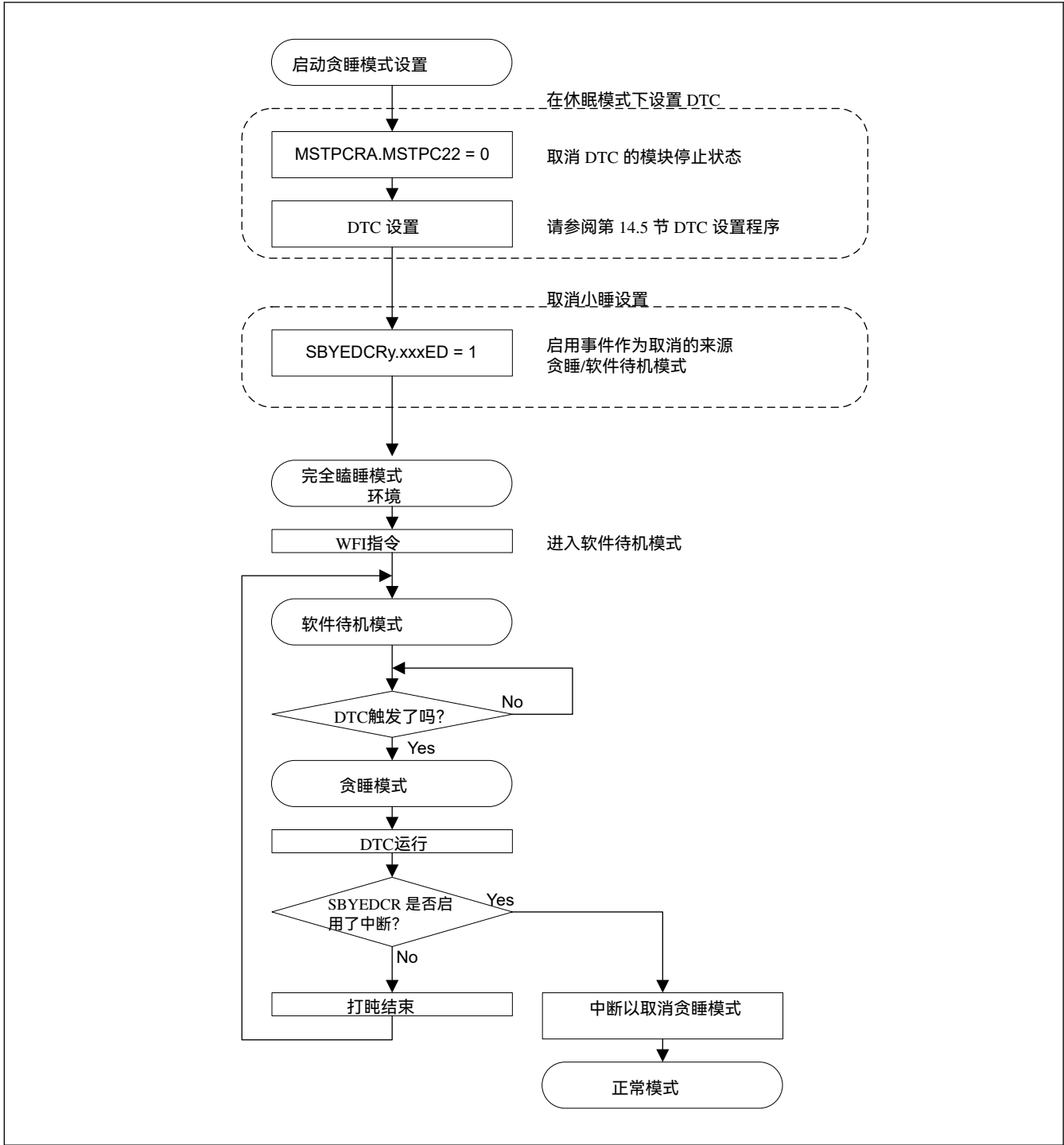


图 9.10 设置示例：在休眠模式下使用 DTC

9.9 使用说明

9.9.1 注册访问

(1) 在下列任何情况下，不得向本节所列寄存器写入数据：

[寄存器]

- 所有外设名称为 SYSTEM 的寄存器以及 FLMODE 寄存器

[条件]

- 从执行 WFI 指令到返回正常模式的这段时间内

- Flash P/E mode

(2) Valid setting for the clock-related registers

Table 9.7 shows the valid settings for the clock-related registers in each operating power control mode. Do not write any value other than the valid setting, otherwise it is ignored. Additionally, each register has some prohibited settings under certain conditions other than those related to the operating power control modes. See [section 8, Clock Generation Circuit](#) for these other conditions of each register.

Table 9.7 Valid settings for the clock-related registers

Mode	Valid settings		
	ICKSCR.CKSEL	HOCOCR.HCSTP	LOCOCR.LCSTP
Subosc-speed	1 (LOCO)	1 (stopped)	0 (operating) 1 (stopped)

(3) Do not write to registers listed in this section for the following condition:

[Registers]

- FLMODE

[Condition]

- ICKSCR.CKSEL = 1 (ICLK = LOCO).

(4) Do not write to registers listed in this section by DTC:

[Registers]

- MSTPCRA, MSTPCRB, MSTPCRC, MSTPCRD, FLMODE

(5) Write access to registers listed in this section is invalid when PRCR.PRC1 bit is 0:

[Registers]

- SBYCR, SYOCDRC.

9.9.2 I/O Port pin states

The I/O port pin states in Software Standby mode and Snooze mode, unless modifying in Snooze mode, are the same before entering the modes.

9.9.3 Module-stop State of DTC

Before writing 1 to MSTPCRA.MSTPA22, clear the DTCST.DTCST bit of the DTC to 0. For details, see [section 14, Data Transfer Controller \(DTC\)](#).

9.9.4 Internal Interrupt Sources

Interrupts do not operate in the module-stop state. If setting the module-stop bit while an interrupt request is generated, a CPU interrupt source or a DTC startup source cannot be cleared. Always disable the associated interrupts before setting the module-stop bits.

9.9.5 Transitioning to Low Power Modes

Because the MCU does not support wakeup by events, do not enter the low power modes such as Sleep mode, Software Standby mode by executing a WFE instruction. Also, do not set the SLEEPDEEP bit of the System Control Register in the Cortex-M23 core because the MCU does not support low power modes by SLEEPDEEP.

- 闪光灯 P/E 模式

(2) 时钟相关寄存器的有效设置

表 9.7 显示了各工作电源控制模式下时钟相关寄存器的有效设置。请勿写入任何非有效值，否则该值将被忽略。此外，除工作电源控制模式外，每个寄存器在某些特定情况下还有一些禁止设置。有关各寄存器的这些其他情况，请参阅第 8 节“时钟生成电路”。

表 9.7 时钟相关寄存器的有效设置

模式	有效设置		
	ICKSCR.CKSEL	HOCOCR.HCSTP	LOCOCR.LCSTP
亚振荡器速度	1 (LOCO)	1 (已停止)	0 (运行中) 1 (停止)

(3) 对于以下情况，请勿写入本节列出的寄存器：

[寄存器]

- FLMODE

[条件]

- ICKSCR.CKSEL = 1 (ICLK = LOCO)。

(4) DTC 不得写入本节列出的寄存器：

[寄存器]

- MSTPCRA、MSTPCRB、MSTPCRC、MSTPCRD、FLMODE

(5) 当 PRCR.PRC1 位为 0 时，对本节所列寄存器的写入访问无效：

[寄存器]

- SBYCR，SYOCDRC。

9.9.2 I/O 端口引脚状态

除非在休眠模式下进行修改，否则软件待机模式和休眠模式下的I/O端口引脚状态在进入这些模式之前是相同的。

9.9.3 DTC模块停止状态

在向 MSTPCRA.MSTPA22 写入 1 之前，将 DTC 的 DTCST.DTCST 位清除为 0。有关详细信息，请参阅第 14 节“数据传输控制器 (DTC)”。

9.9.4 内部中断源

中断在模块停止状态下无法工作。如果在生成中断请求时设置了模块停止位，则无法清除 CPU 中断源或 DTC 启动源。务必在设置模块停止位之前禁用相关的中断。

9.9.5 过渡到低功耗模式

由于MCU不支持事件唤醒，请勿通过执行WFE指令进入低功耗模式，例如睡眠模式或软件待机模式。此外，请勿设置Cortex-M23内核中系统控制寄存器的SLEEPDEEP位，因为MCU不支持通过SLEEPDEEP位进入低功耗模式。

9.9.6 Timing of WFI Instruction

It is possible for the WFI instruction to be executed before I/O register write is completed, in which case operation might not be as intended. This can happen if the WFI is placed immediately after a write to an I/O register. To avoid this problem, read back the register that was written to confirm that the write completed.

9.9.7 Writing to the IWDTC Registers by DTC in Sleep Mode or Snooze Mode

Do not write to the IWDTC registers by the DTC while IWDTC is stopped after entering Sleep mode or Snooze mode.

9.9.8 Oscillators in Snooze Mode

ICLK sourced oscillator that stop on entering Software Standby mode automatically restart when a trigger for switching to Snooze mode is generated. Other oscillators maintain their state in Software standby mode.

9.9.9 Using SAU0 in Snooze Mode

A transition to Software Standby mode must not occur during an SAU0 communication.

9.9.10 Using UART0 in Snooze Mode

When using UART0 in Snooze mode, ensure that the Snooze request (RXD0 falling edge) does not conflict with the wakeup requests set by the SBYEDCRn register, otherwise UART cannot be guaranteed.

When using UART in Snooze mode, the following conditions must be satisfied:

- The clock source must be HOCO.
- The RXD0 pin must be kept high before entering Software Standby mode.

9.9.11 Sleep-on-exit Function

There are 2 ways to transition to low power modes. One is WFI instruction and the other is Sleep-on-exit. When Sleep-on-exit is used for transition to low power modes, the comments described in the User’s Manual Hardware for WFI instruction are also applicable to Sleep-on-exit.

9.9.6 WFI 指令的时机

WFI 指令有可能在I/O寄存器写入完成之前执行，在这种情况下，操作可能无法按预期进行。如果 WFI 指令紧随I/O寄存器的写入之后，则可能会发生这种情况。为避免此问题，请回读已写入的寄存器以确认写入已完成。

9.9.7 睡眠模式或休眠模式下 DTC 对 IWDTC 寄存器的写入

在进入睡眠模式或贪睡模式后，当 IWDTC 停止运行时，DTC 不得向 IWDTC 寄存器写入数据。

9.9.8 振荡器处于休眠模式

进入软件待机模式后停止的 ICLK 振荡器，在触发切换到休眠模式的信号时会自动重启。其他振荡器在软件待机模式下保持其状态。

9.9.9 在睡眠模式下使用 SAU0

在 SAU0 通信期间，不得切换到软件待机模式。

9.9.10 在休眠模式下使用 UART0

在睡眠模式下使用 UART0 时，请确保睡眠请求（RXD0 下降沿）与 SBYEDCRn 寄存器设置的唤醒请求不冲突，否则无法保证 UART 正常工作。

使用 UART 的休眠模式时，必须满足以下条件：

- 时钟源必须是 HOCO。
- 进入软件待机模式前，必须将 RXD0 引脚保持高电平。

9.9.11退出时睡眠功能

有两种方法可以切换到低功耗模式。一种是WFI指令，另一种是退出睡眠。当使用退出睡眠切换到低功耗模式时，用户手册硬件部分中关于WFI指令的说明也适用于退出睡眠。

10. Register Write Protection

10.1 Overview

The register write protection function protects important registers from being overwritten due to software errors. The registers to be protected are set with the Protect Register (PRCR).

Table 10.1 lists the association between the bits in the PRCR register and the registers to be protected.

Table 10.1 Association between the bits in the PRCR register and registers to be protected

PRCR bit	Register to be protected
PRC0	● Registers related to the clock generation circuit: FSUBSCR, ICLKSCR, LOCOCR, HOCOCR, HOCODIV, OSMC, LIOTRM
PRC1	● Registers related to the low power modes: SBYCR, SYOCDCCR
PRC3	● Registers related to the LVD: LVD1CR, LVD1MKR, LVD1SR

10.2 Register Descriptions

10.2.1 PRCR : Protect Register

Base address: SYSC = 0x4001_E000															
Offset address: 0x08FE															
Bit position: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0															
Bit field: PRKEY[7:0] — — — — PRC3 — PRC1 PRC0															
Value after reset: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0															

Bit	Symbol	Function	R/W
0	PRC0	Enable writing to the registers related to the clock generation circuit 0: Disable writes 1: Enable writes	R/W
1	PRC1	Enable writing to the registers related to the low power modes 0: Disable writes 1: Enable writes	R/W
2	—	This bit is read as 0. The write value should be 0.	R/W
3	PRC3	Enable writing to the registers related to the LVD 0: Disable writes 1: Enable writes	R/W
7:4	—	These bits are read as 0. The write value should be 0.	R/W
15:8	PRKEY[7:0]	PRC Key Code These bits control the write access to the PRCR register. To modify the PRCR register, write 0xA5 to the upper 8 bits and the target value to the lower 8 bits as a 16-bit unit.	W

PRCn bits (Protect bit n) (n = 0, 1, 3)

The PRCn bits enable or disable writing to the protected registers listed in Table 10.1. Setting the PRCn bits to 1 or 0 enables or disables writing, respectively.

10.寄存器写保护

10.1 概述

寄存器写保护功能可防止重要寄存器因软件错误而被写入。需要保护的寄存器通过保护寄存器（PRCR）进行设置。

表 10.1 列出了 PRCR 寄存器中的位与要保护的寄存器之间的关联。

表 10.1 PRCR 寄存器中的位与待保护寄存器之间的关联

PRCR 位	注册即可获得保护
PRC0	●与时钟生成电路相关的寄存器： FSUBSCR、ICLKSCR、LOCOCR、HOCOCR、HOCODIV、OSMC、LIOTRM
PRC1	●与低功耗模式相关的寄存器： SBYCR, SYOCDCCR
PRC3	●与LVD相关的寄存器： LVD1CR、LVD1MKR、LVD1SR

10.2 寄存器描述

10.2.1PRCR：保护寄存器

基地址: SYSC = 0x4001_E000															
偏移地址: 0x08FE															
位位置: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0															
位域: PRKEY[7:0] — — — — PRC3 — PRC1 PRC0															
重置后的值: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0															

少量	象征	功能	R/W
0	PRC0	允许写入与时钟生成电路相关的寄存器 0: 禁用写入 1: 启用写入	R/W
1	PRC1	启用对低功耗模式相关寄存器的写入 0: 禁用写入 1: 启用写入	R/W
2	—	该位读取值为 0。写入值应为 0。	R/W
3	PRC3	启用对与LVD相关的寄存器的写入功能 0: 禁用写入 1: 启用写入	R/W
7:4	—	这些位读取为 0。写入值应为 0。	R/W
15:8	PRKEY[7:0]	中国密钥代码 这些位控制对 PRCR 寄存器的写入访问。要修改 PRCR 寄存器，请将 0xA5 写入高 8 位，并将目标值写入低 8 位，作为一个 16 位单元。	W

PRCn 位（保护位 n）(n = 0, 1, 3)

PRCn位用于启用或禁用对表 10.1 中列出的受保护寄存器的写入。将 PRCn 位设置为 1 或 0 分别用于启用或禁用写入。

11. Interrupt Controller Unit (ICU)

11.1 Overview

The Interrupt Controller Unit (ICU) controls which event signals are linked to the Nested Vector Interrupt Controller (NVIC), and the Data Transfer Controller (DTC) modules. The ICU also controls non-maskable interrupts.

Table 11.1 lists the ICU specifications, Figure 11.1 shows a block diagram, and Table 11.2 lists the I/O pins.

Table 11.1 ICU specifications

Item		Description
Maskable interrupts	Peripheral function interrupts	- Interrupts from peripheral modules - Number of sources: 23
	External pin interrupts	- Interrupt detection on falling edge, rising edge, rising and falling edges. One of these detection methods can be set for each source 6 sources, with interrupts from IRQi (i = 0 to 5) pins.
	Interrupt requests to CPU (NVIC)	29 interrupt requests are output to NVIC.
	DTC control	- The DTC can be activated using interrupt sources*1 - The method for selecting an interrupt source is the same as that of the interrupt request to NVIC.
Non-maskable interrupts*2	NMI pin interrupt	- Interrupt from the NMI pin - Interrupt detection on falling edge or rising edge
	IWDT underflow/refresh error*3	Interrupt on an underflow of the down-counter or occurrence of a refresh error
	Low voltage detection 1*3	Voltage monitor 1 interrupt of the voltage monitor 1 circuit (LVD_LVD1)
Low power modes		- Sleep mode: return is initiated by non-maskable interrupts or any other interrupt source - Software Standby mode: return is initiated by non-maskable interrupts or any other interrupt source. Interrupt can be selected in the SBYEDCRn register. - Snooze mode: return is initiated by non-maskable interrupts or any other interrupt source. Interrupt can be selected in the SBYEDCRn register. See section 11.2.14. SBYEDCR0 : Software Standby/Snooze End Control Register 0 and section 11.2.15. SBYEDCR1 : Software Standby/Snooze End Control Register 1.

Note 1. For the DTC activation sources, see Table 11.5.

Note 2. Non-maskable interrupts can be enabled only once after a reset release.

Note 3. These non-maskable interrupts can also be used as maskable interrupts. When used as maskable interrupts, do not change the value of the NMIER register from the reset state. To enable voltage monitor 1 interrupts, set the LVD1CR.IRQSEL bits to 1.

Figure 11.1 shows the ICU block diagram.

11.中断控制器单元 (ICU)

11.1 概述

中断控制器单元 (ICU) 控制哪些事件信号链接到嵌套向量中断控制器 (NVIC) 和数据传输控制器 (DTC) 模块。ICU 还控制不可屏蔽中断。

表 11.1 列出了 ICU 规格，图 11.1 显示了框图，表 11.2 列出了 I/O 引脚。

表 11.1 ICU 规格

物品		描述
可屏蔽中断	外围功能中断	●外围模块中断 来源数量：23
	外部引脚中断	●下降沿、上升沿、上升沿和下降沿进行中断检测。每个信号源可以设置其中一种检测方法。 ●6 个中断源，中断来自 IRQi(i = 0至 5) 引脚。
	向 CPU 发出中断请求 (NVIC)	●向 NVIC 输出 29 个中断请求。
	DTC 控制	●DTC 可通过中断源*1 激活 ●选择中断源的方法与向 NVIC 发出中断请求的方法相同。
不可遮盖中断*2	NMI引脚中断	●来自 NMI 引脚的中断 ●下降沿或上升沿中断检测
	IWDT 下溢/刷新错误*3	计数器下溢或刷新错误发生时中断
	低电压检测1*3	电压监控器 1 电路 (LVD_LVD1) 的电压监控器 1 中断
低功耗模式		●睡眠模式：由不可屏蔽中断或任何其他中断源触发返回。 ●软件待机模式：由不可屏蔽中断或其他任何中断源触发返回。中断可在SBYEDCRn寄存器中选择。 ●休眠模式：由不可屏蔽中断或任何其他中断源触发返回。 可以在 SBYEDCRn 寄存器中选择中断。 请参阅第 11.2.14 节。SBYEDCR0: 软件待机/休眠结束控制寄存器 0 和第 11.2.15 节。SBYEDCR1: 软件待机/休眠结束控制寄存器 1。

注1.有关DTC激活源，请参见表11.5。

注 2. 不可屏蔽中断只能在复位释放后启用一次。

注3：这些不可屏蔽中断也可以用作可屏蔽中断。用作可屏蔽中断时，请勿将NMIER寄存器的值从复位状态更改。要启用电压监视器1中断，请将LVD1CR.IRQSEL位设置为1。

图 11.1 显示了 ICU 框图。

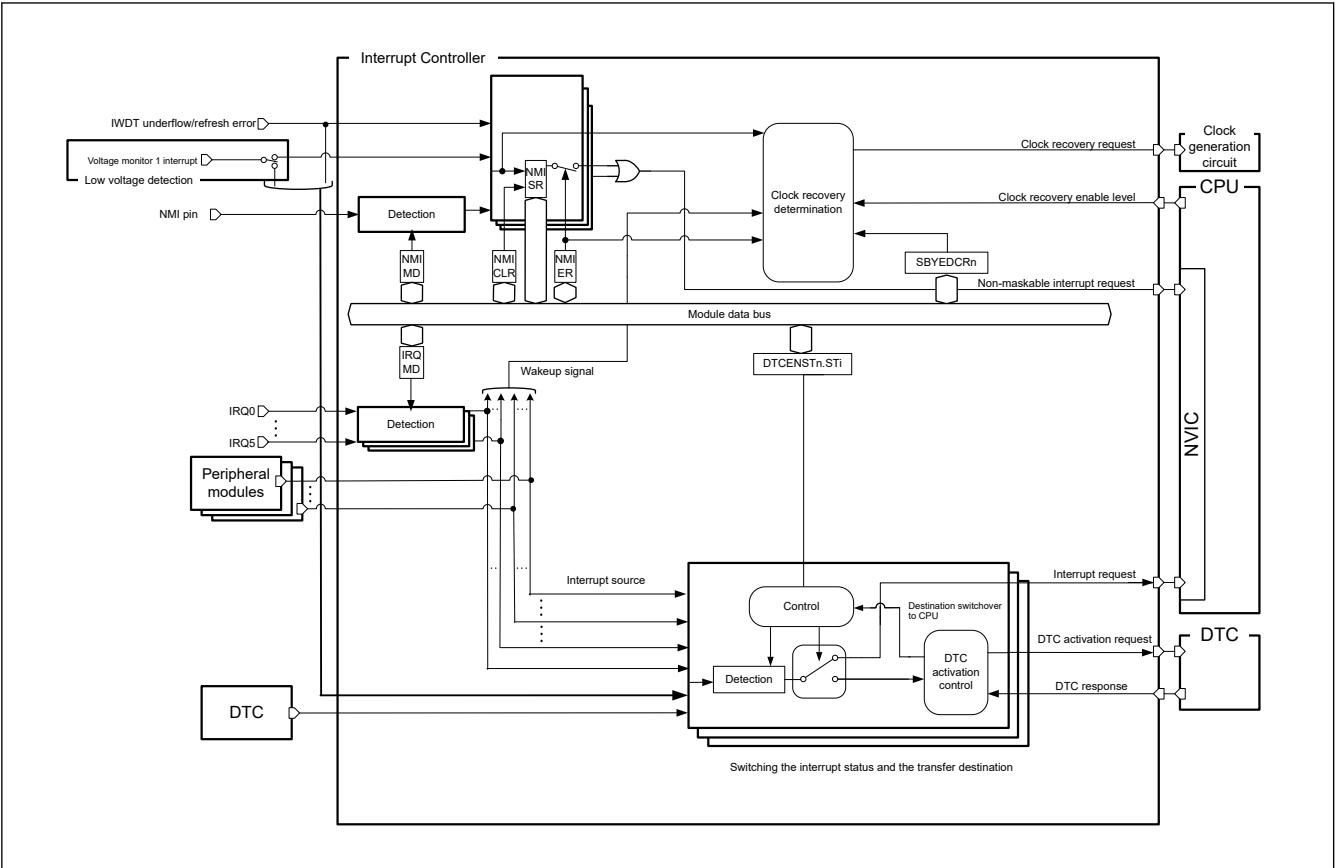


Figure 11.1 ICU block diagram

Table 11.2 lists the ICU input/output pins.

Table 11.2 ICU I/O pins

Pin name	I/O	Description
NMI	Input	Non-maskable interrupt request pin
IRQi (i = 0 to 5)	Input	External interrupt request pins

11.2 Register Descriptions

This chapter does not describe the Arm® NVIC internal registers. For information about these registers, see ARM® Cortex®-M23 Processor Technical Reference Manual (ARM DDI 0550C).

11.2.1 IRQCRi : IRQ Control Register i (i = 0 to 5)

Base address: ICU = 0x4000_6000							
Offset address: 0x0000 + i							
Bit position: 7 6 5 4 3 2 1 0							
Bit field: — — — — — — IRQMD[1:0]							
Value after reset: 0 0 0 0 0 0 0 0							

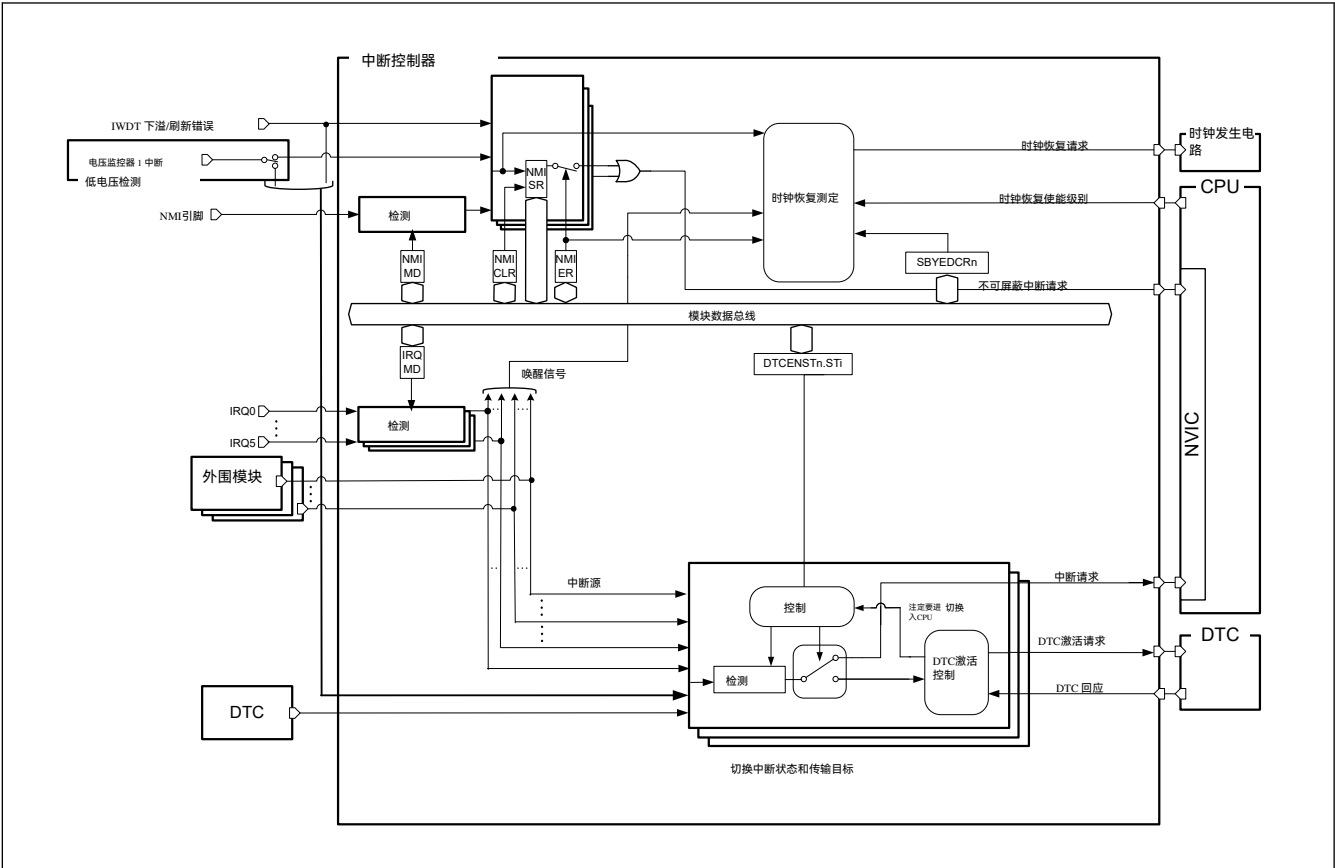


图 11.1 ICU 框图

表 11.2 列出了 ICU 输入/输出引脚。

表 11.2 ICU I/O 引脚

引脚名称	I/O	描述
NMI	输入	不可屏蔽中断请求引脚
IRQi(i = 0至 5)	输入	外部中断请求引脚

11.2 寄存器描述

本章不描述Arm® NVIC内部寄存器。有关这些寄存器的信息，请参阅ARM®Cortex®-M23处理器技术参考手册（ARM DDI 0550C）。

11.2.1IRQCRi: IRQ 控制寄存器 i(i = 0至 5)

基地址: ICU = 0x4000_6000							
偏移地址: 0x0000 + i							
位位置: 7 6 5 4 3 2 1 0							
位域: — — — — — — IRQMD[1:0]							
重置后的值: 0 0 0 0 0 0 0 0							

Bit	Symbol	Function	R/W
1:0	IRQMD[1:0]	IRQi Detection Sense Select 0 0: Falling edge 0 1: Rising edge 1 0: Rising and falling edges 1 1: Setting prohibited	R/W
7:2	—	These bits are read as 0. The write value should be 0.	R/W

IRQCRi register changes must satisfy the following conditions:

- For a wakeup enable signal:
Change the IRQCRi register setting before setting the target SBYEDCR0.IRQnED (n = 0 to 5). The register value should be changed when the target SBYEDCR0.IRQnED is 0.

IRQMD[1:0] bits (IRQi Detection Sense Select)

The IRQMD[1:0] bits set the detection sensing method for the IRQi external pin interrupt sources. For setting method when using external pin interrupt, see [section 11.5.5. External Pin Interrupts](#).

11.2.2 NMISR : Non-maskable Interrupt Status Register

Base address:	ICU = 0x4000_6000															
Offset address:	0x0140															
Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	—	—	—	—	—	—	—	—	NMIST	—	—	—	—	LVD1S T	—	IWDT ST
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
0	IWDTST	IWDT Underflow/Refresh Error Interrupt Status Flag 0: Interrupt not requested 1: Interrupt requested	R
1	—	This bit is read as 0.	R
2	LVD1ST	Voltage Monitor 1 Interrupt Status Flag 0: Interrupt not requested 1: Interrupt requested	R
6:3	—	These bits are read as 0.	R
7	NMIST	NMI Pin Interrupt Status Flag 0: Interrupt not requested 1: Interrupt requested	R
15:8	—	These bits are read as 0.	R

The NMISR register monitors the status of non-maskable interrupt sources. Writes to the NMISR register are ignored. The setting in the Non-Maskable Interrupt Enable Register (NMIER) does not affect the status flags in this register. Before the end of the non-maskable interrupt handler, check that all bits in this register are set to 0 to confirm that no other NMI requests are generated during handler processing.

IWDTST flag (IWDT Underflow/Refresh Error Interrupt Status Flag)

The IWDTST flag indicates an IWDT underflow/refresh error interrupt request. It is read-only and cleared by the NMICLR.IWDTCLR bit.

[Setting condition]

When the IWDT underflow/refresh error interrupt is generated and this interrupt source is enabled.

[Clearing condition]

When 1 is written to the NMICLR.IWDTCLR bit.

少量	象征	功能	R/W
1:0	IRQMD[1:0]	IRQi 检测感应选择 0 0: 下降沿 0 1: 上升沿 1 0: 上升沿和下降沿 1.1: 禁止设置	R/W
7:2	—	这些位读取为 0。写入值应为 0。	R/W

IRQCRi寄存器更改必须满足以下条件:

- 唤醒使能信号:
在将目标 SBYEDCR0.IRQnED(n = 0 设置为 5 之前, 请更改 IRQCRi 寄存器设置。当目标 SBYEDCR0.IRQnED 为 0 时, 应更改寄存器值。

IRQMD[1:0] 位（IRQi 检测感应选择）

IRQMD[1:0] 位设置 IRQi 外部引脚中断源的检测方法。有关使用外部引脚中断时的设置方法, 请参阅 11.5.5 节“外部引脚中断”。

11.2.2 NMISR: 不可屏蔽中断状态寄存器

基地址: ICU = 0x4000_6000																	
偏移地址: 0x0140																	
位位置:	1514131211109876543210																
位域:	<table><tr><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>NMIST</td><td>—</td><td>—</td><td>—</td><td>—</td><td>LVD1S T</td><td>—</td><td>IWDT ST</td></tr></table>	—	—	—	—	—	—	—	—	NMIST	—	—	—	—	LVD1S T	—	IWDT ST
—	—	—	—	—	—	—	—	NMIST	—	—	—	—	LVD1S T	—	IWDT ST		
重置后的值:	0000000000000000																

少量	象征	功能	R/W
0	IWDTST	IWDT 下溢/刷新错误中断状态标志 0: 未请求中断 1: 请求中断	R
1	—	该位读取值为 0。	R
2	LVD1ST	电压监视器 1 中断状态标志 0: 未请求中断 1: 请求中断	R
6:3	—	这些位读取为 0。	R
7	NMIST	NMI引脚中断状态标志 0: 未请求中断 1: 请求中断	R
15:8	—	这些位读取为 0。	R

NMISR 寄存器用于监视不可屏蔽中断源的状态。对 NMISR 寄存器的写入操作将被忽略。不可屏蔽中断使能寄存器 (NMIER) 中的设置不会影响此寄存器中的状态标志。在不可屏蔽中断处理程序结束之前, 请检查此寄存器中的所有位是否都设置为 0, 以确认在处理程序处理期间没有生成其他 NMI 请求。

IWDTST 标志（IWDT 下溢/刷新错误中断状态标志）

IWDTST 标志指示 IWDT 下溢/刷新错误中断请求。它是只读的, 由 NMICLR.IWDTCLR 位清除。

[设置条件]

当 IWDT 下溢/刷新错误中断生成且此中断源已启用时。

[清除条件]

当 1 被写入 NMICLR.IWDTCLR 位时。

LVD1ST flag (Voltage Monitor 1 Interrupt Status Flag)

The LVD1ST flag indicates a request for voltage monitor 1 interrupt. It is read-only and cleared by the NMICLR.LVDCLR bit.

[Setting condition]

When the voltage monitor 1 interrupt is generated and this interrupt source is enabled.

[Clearing condition]

When 1 is written to the NMICLR.LVD1CLR bit.

NMIST flag (NMI Pin Interrupt Status Flag)

The NMIST flag indicates an NMI pin interrupt request. It is read-only and cleared by the NMICLR.NMICLR bit.

[Setting condition]

When an edge specified by the NMICR.NMIMD bit is input to the NMI pin.

[Clearing condition]

When 1 is written to the NMICLR.NMICLR bit.

11.2.3 NMIER : Non-maskable Interrupt Enable Register

Base address: ICU = 0x4000_6000																
Offset address: 0x0120																
Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	—	—	—	—	—	—	—	—	NMIE N	—	—	—	—	LVD1E N	—	IWDT EN
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
0	IWDTEN	IWDT Underflow/Refresh Error Interrupt Enable 0: Disabled 1: Enabled.	R/W*1 *2
1	—	This bit is read as 0. The write value should be 0.	R/W
2	LVD1EN	Voltage Monitor 1 Interrupt Enable 0: Disabled 1: Enabled	R/W*1 *2
6:3	—	These bits are read as 0. The write value should be 0.	R/W
7	NMIEN	NMI Pin Interrupt Enable 0: Disabled 1: Enabled	R/W*1
15:8	—	These bits are read as 0. The write value should be 0.	R/W

Note 1. You can write 1 to this bit only once after reset. Subsequent write accesses are invalid. Writing 0 to this bit is invalid.

Note 2. Do not write 1 to this bit when the source is used as an event signal.

IWDTEN bit (IWDT Underflow/Refresh Error Interrupt Enable)

The IWDTEN bit enables IWDT underflow/refresh error interrupt as an NMI trigger.

LVD1EN bit (Voltage Monitor 1 Interrupt Enable)

The LVD1EN bit enables voltage monitor 1 interrupt as an NMI trigger.

NMIEN bit (NMI Pin Interrupt Enable)

The NMIEN bit enables NMI pin interrupt as an NMI trigger.

LVD1ST 标志（电压监视器 1 中断状态标志）

LVD1ST 标志位表示请求电压监视器 1 中断。它是只读的，由 NMICLR.LVDCLR 位清除。

[设置条件]

当电压监视器 1 中断生成且此中断源启用时。

[清除条件]

当 1 被写入 NMICLR.LVD1CLR 位时。

NMIST 标志（NMI 引脚中断状态标志）

NMIST 标志指示 NMI 引脚中断请求。它是只读的，由 NMICLR.NMICLR 位清除。

[设置条件]

当 NMICR.NMIMD 位指定的边沿输入到 NMI 引脚时。

[清除条件]

当 1 被写入 NMICLR.NMICLR 位时。

11.2.3NMIER：不可屏蔽中断使能寄存器

基地址： ICU = 0x4000_6000

Offset address: 0x0120

位位置:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域:	—	—	—	—	—	—	—	—	NMIE N	—	—	—	—	LVD1E N	—	IWDT EN
重置后的值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
0	IWDTEN	IWDT 下溢/刷新错误中断使能 0：禁用 1：启用。	R/W*1 *2
1	—	该位读取值为 0。写入值应为 0。	R/W
2	LVD1EN	电压监视器 1 中断使能 0：已禁用 1：已启用	R/W*1 *2
6:3	—	这些位读取为 0。写入值应为 0。	R/W
7	NMIEN	NMI引脚中断使能 0：已禁用 1：已启用	R/W*1
15:8	—	这些位读取为 0。写入值应为 0。	R/W

注1：复位后，此位只能写入1一次。后续写入操作无效。向此位写入0也是无效的。

注 2. 当源用作事件信号时，不要将 1 写入此位。

IWDTEN 位（IWDT 下溢/刷新错误中断使能）

IWDTEN 位使 IWDT 下溢/刷新错误中断能够作为 NMI 触发器。

LVD1EN 位（电压监视器 1 中断使能）

LVD1EN 位使电压监视器 1 中断能够作为 NMI 触发器。

NMIEN 位（NMI 引脚中断使能）

NMIEN 位使 NMI 引脚中断能够作为 NMI 触发器。

11.2.4 NMICLR : Non-maskable Interrupt Status Clear Register

Base address: ICU = 0x4000_6000																
Offset address: 0x0130																
Bit position: 1514131211109876543210																
Bit field: — — — — — — — NMICLR — — — — LVD1CLR — IWDTCCLR																
Value after reset: 0000000000000000																

Bit	Symbol	Function	R/W
0	IWDTCCLR	IWDT Underflow/Refresh Error Interrupt Status Flag Clear 0: No effect 1: Clear the NMISR.IWDTST flag	R/W ^{*1}
1	—	This bit is read as 0. The write value should be 0.	R/W
2	LVD1CLR	Voltage Monitor 1 Interrupt Status Flag Clear 0: No effect 1: Clear the NMISR.LVD1ST flag	R/W ^{*1}
6:3	—	These bits are read as 0. The write value should be 0.	R/W
7	NMICLR	NMI Pin Interrupt Status Flag Clear 0: No effect 1: Clear the NMISR.NMIST flag	R/W ^{*1}
15:8	—	These bits are read as 0. The write value should be 0.	R/W

Note 1. Only write 1 to this bit.

IWDTCCLR bit (IWDT Underflow/Refresh Error Interrupt Status Flag Clear)

Writing 1 to the IWDTCCLR bit clears the NMISR.IWDTST flag. This bit is read as 0.

LVD1CLR bit (Voltage Monitor 1 Interrupt Status Flag Clear)

Writing 1 to the LVD1CLR bit clears the NMISR.LVD1ST flag. This bit is read as 0.

NMICLR bit (NMI Pin Interrupt Status Flag Clear)

Writing 1 to the NMICLR bit clears the NMISR.NMIST flag. This bit is read as 0.

11.2.5 NMICR : NMI Pin Interrupt Control Register

Base address: ICU = 0x4000_6000							
Offset address: 0x0100							
Bit position: 76543210							
Bit field: — — — — — — — NMIMD							
Value after reset: 00000000							

Bit	Symbol	Function	R/W
0	NMIMD	NMI Detection Set 0: Falling edge 1: Rising edge	R/W
7:1	—	These bits are read as 0. The write value should be 0.	R/W

Change the NMICR register settings before enabling NMI pin interrupts, that is, before setting NMIER.NMIEN to 1.

NMIMD bit (NMI Detection Set)

The NMIMD bit selects the detection sensing method for the NMI pin interrupts.

11.2.4 NMICLR: 不可屏蔽中断状态清除寄存器

Base address: ICU = 0x4000_6000																
Offset address: 0x0130																
位位置: 1514131211109876543210																
位域: — — — — — — — NMICLR — — — — LVD1CLR — IWDTCCLR																
重置后的值: 0000000000000000																

少量	象征	功能	R/W
0	IWDTCCLR	IWDT 下溢/刷新错误中断状态标志清除 0: 无效果 1: 清除 NMISR.IWDTST 标志	R/W ^{*1}
1	—	该位读取值为 0。写入值应为 0。	R/W
2	LVD1CLR	电压监视器 1 中断状态标志清除 0: 无效果 1: 清除 NMISR.LVD1ST 标志	R/W ^{*1}
6:3	—	这些位读取为 0。写入值应为 0。	R/W
7	NMICLR	NMI引脚中断状态标志清除 0: 无效果 1: 清除 NMISR.NMIST 标志	R/W ^{*1}
15:8	—	这些位读取为 0。写入值应为 0。	R/W

注1: 此位只能写入1。

IWDTCCLR 位 (IWDT 下溢/刷新错误中断状态标志清除)

将 1 写入 IWDTCCLR 位会清除 NMISR.IWDTST 标志。该位读取时为 0。

LVD1CLR 位 (电压监视器 1 中断状态标志清除)

将 LVD1CLR 位写入 1 会清除 NMISR.LVD1ST 标志。该位读取时为 0。

NMICLR 位 (NMI 引脚中断状态标志清除)

将 NMICLR 位写入 1 会清除 NMISR.NMIST 标志。该位读取时为 0。

11.2.5 NMICR: NMI 引脚中断控制寄存器

基地址: ICU = 0x4000_6000							
Offset address: 0x0100							
位位置: 76543210							
位域: — — — — — — — NMIMD							
重置后的值: 00000000							

少量	象征	功能	R/W
0	NMIMD	NMI检测集 0: 下降沿 1: 上升沿	R/W
7:1	—	这些位读取为 0。写入值应为 0。	R/W

在启用 NMI 引脚中断之前，即在将 NMIER.NMIEN 设置为 1 之前，更改 NMICR 寄存器设置。

NMIMD 位 (NMI 检测集)

NMIMD 位选择 NMI 引脚中断的检测传感方法。

11.2.6 DTCENST0 : DTC Enable Status Register 0

Base address: ICU = 0x4000_6000

Offset address: 0x0300

Bit position:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Bit field:	ST31	ST30	ST29	ST28	ST27	—	—	—	ST23	ST22	—	—	ST19	ST18	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	—	—	—	—	—	—	—	—	ST7	ST6	ST5	ST4	ST3	ST2	ST1	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
0	—	This bit is read as 0.	R
7:1	ST1 to ST7	DTC Enable Status by Event Number i The suffix number of each bit symbol corresponds to the DTC vector number i. 0: DTC Disable by Event number i 1: DTC Enable by Event number i	R
17:8	—	These bits are read as 0.	R
19:18	ST18 to ST19	DTC Enable Status by Event Number i The suffix number of each bit symbol corresponds to the DTC vector number i. 0: DTC Disable by Event number i 1: DTC Enable by Event number i	R
21:20	—	These bits are read as 0.	R
23:22	ST22 to ST23	DTC Enable Status by Event Number i The suffix number of each bit symbol corresponds to the DTC vector number i. 0: DTC Disable by Event number i 1: DTC Enable by Event number i	R
26:24	—	These bits are read as 0.	R
31:27	ST27 to ST31	DTC Enable Status by Event Number i The suffix number of each bit symbol corresponds to the DTC vector number i. 0: DTC Disable by Event number i 1: DTC Enable by Event number i	R

STi bits (DTC Enable Status by Event Number i) (i = 1 to 7, 18 to 19, 22 to 23, 27 to 31)

The STi bit indicates whether the corresponding event is disabled or enabled as a DTC activation factor. This register is read-only and is set by the DTCENSETn.SETi bit and cleared by the DTCENCLRn.CLRi bit.

After check the DTC transfer end (DTCENSTn.STi = 0), stop the DTC module by setting MSTPCRA.MSTPA22 or DTCST.DTCST register.

[Setting condition]

- When 1 is written to the DTCENSETn.SETi bit.

[Clearing condition]

- When 1 is written to the DTCENCLRn.CLRi bit.
- When the specified number of transfers is complete. For chain transfers, when the specified number of transfers for the last chain transfer is complete.

11.2.6 DTCENST0: DTC 使能状态寄存器 0

Base address: ICU = 0x4000_6000

Offset address: 0x0300

位位置:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域:	ST31	ST30	ST29	ST28	ST27	—	—	—	ST23	ST22	—	—	ST19	ST18	—	—
重置后的值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位位置:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域:	—	—	—	—	—	—	—	—	ST7	ST6	ST5	ST4	ST3	ST2	ST1	—
重置后的值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
0	—	该位读取值为 0。	R
7:1	ST1 至 ST7	DTC 启用状态（按事件编号 i） 每个比特符号的后缀编号对应于 DTC 向量编号 i。 0: 根据事件编号 i 禁用 DTC 1: 根据事件编号 i 启用 DTC	R
17:8	—	这些位读取为 0。	R
19:18	ST18 至 ST19	DTC 启用状态（按事件编号 i） 每个比特符号的后缀编号对应于 DTC 向量编号 i。 0: 根据事件编号 i 禁用 DTC 1: 根据事件编号 i 启用 DTC	R
21:20	—	这些位读取为 0。	R
23:22	ST22 至 ST23	DTC 启用状态（按事件编号 i） 每个比特符号的后缀编号对应于 DTC 向量编号 i。 0: 根据事件编号 i 禁用 DTC 1: 根据事件编号 i 启用 DTC	R
26:24	—	这些位读取为 0。	R
31:27	ST27 至 ST31	DTC 启用状态（按事件编号 i） 每个比特符号的后缀编号对应于 DTC 向量编号 i。 0: 根据事件编号 i 禁用 DTC 1: 根据事件编号 i 启用 DTC	R

STi 位（DTC 使能状态，按事件编号 i）(i = 1至 7、18 至 19、22 至 23、27 至 31)

STi 位指示相应事件是否作为 DTC 激活因子被禁用或启用。此寄存器为只读寄存器，由 DTCENSETn.SETi 位置位，由 DTCENCLRn.CLRi 位清除。

检查 DTC 传输结束 (DTCENSTn.STi = 0) 后，通过设置 MSTPCRA.MSTPA22 或 DTCST.DTCST 寄存器来停止 DTC 模块。

[设置条件]

- 当 1 被写入 DTCENSETn.SETi 位时。

[清除条件]

- 当 DTCENCLRn.CLRi 位写入 1 时。
- 当指定次数的转账完成时。对于链式转账，当最后一次链式转账的指定次数完成时。

11.2.7 DTCENST1 : DTC Enable Status Register 1

Base address: ICU = 0x4000_6000

Offset address: 0x0304

Bit position:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Bit field:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	—	—	—	—	—	—	—	—	—	ST38	ST37	ST36	ST35	ST34	ST33	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
0	—	This bit is read as 0.	R
6:1	ST33 to ST38	DTC Enable Status by Event Number i The suffix number of each bit symbol corresponds to the DTC vector number i. 0: DTC Disable by Event number i 1: DTC Enable by Event number i	R
31:7	—	These bits are read as 0.	R

STi bits (DTC Enable Status by Event Number i) (i = 33 to 38)

The STi bit indicates whether the corresponding event is disabled or enabled as a DTC activation factor. This register is read-only and is set by the DTCENSETn.SETi bit and cleared by the DTCENCLRn.CLRi bit.

After check the DTC transfer end (DTCENSTn.STi = 0), stop the DTC module by setting MSTPCRA.MSTPA22 or DTCST.DTCST register.

[Setting condition]

- When 1 is written to the DTCENSETn.SETi bit.

[Clearing condition]

- When 1 is written to the DTCENCLRn.CLRi bit.
- When the specified number of transfers is complete. For chain transfers, when the specified number of transfers for the last chain transfer is complete.

11.2.8 DTCENSET0 : DTC Enable Set Register 0

Base address: ICU = 0x4000_6000

Offset address: 0x0310

Bit position:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Bit field:	SET31	SET30	SET29	SET28	SET27	—	—	—	SET23	SET22	—	—	SET19	SET18	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	—	—	—	—	—	—	—	—	SET7	SET6	SET5	SET4	SET3	SET2	SET1	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
0	—	This bit is read as 0. The write value should be 0.	R/W

11.2.7 DTCENST1:DTC 使能状态寄存器 1

基地址: ICU = 0x4000_6000

偏移地址: 0x0304

位位置:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
重置后的值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
位位置:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域:	—	—	—	—	—	—	—	—	—	ST38	ST37	ST36	ST35	ST34	ST33	—
重置后的值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
0	—	该位读取值为 0。	R
6:1	ST33 至 ST38	DTC 启用状态（按事件编号 i） 每个比特符号的后缀编号对应于 DTC 向量编号 i。 0: 根据事件编号 i 禁用 DTC 1: 根据事件编号 i 启用 DTC	R
31:7	—	这些位读取为 0。	R

STi 位 (DTC 使能状态, 按事件编号 i) (i = 33 至 38)

STi 位指示相应事件是否作为 DTC 激活因子被禁用或启用。此寄存器为只读寄存器，由 DTCENSETn.SETi 位置位，由 DTCENCLRn.CLRi 位清除。

检查 DTC 传输结束 (DTCENSTn.STi = 0) 后，通过设置 MSTPCRA.MSTPA22 或 DTCST.DTCST 寄存器来停止 DTC 模块。

[设置条件]

- 当 1 被写入 DTCENSETn.SETi 位时。

[清除条件]

- 当 DTCENCLRn.CLRi 位写入 1 时。
- 当指定次数的转账完成时。对于链式转账，当最后一次链式转账的指定次数完成时。

11.2.8DTCENSET0: DTC 使能设置寄存器 0

基地址: ICU = 0x4000_6000

偏移地址: 0x0310

位位置:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域:	SET31	SET30	SET29	SET28	SET27	—	—	—	SET23	SET22	—	—	SET19	SET18	—	—
重置后的值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
位位置:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域:	—	—	—	—	—	—	—	—	SET7	SET6	SET5	SET4	SET3	SET2	SET1	—
重置后的值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
0	—	该位读取值为 0。写入值应为 0。	R/W

Bit	Symbol	Function	R/W
7:1	SET1 to SET7	DTC Enable Set by Event Number i The suffix number of each bit symbol corresponds to the DTC vector number i. 0: No effect 1: DTC Enable by Event number i	R/W
17:8	—	These bits are read as 0. The write value should be 0.	R/W
19:18	SET18 to SET19	DTC Enable Set by Event Number i The suffix number of each bit symbol corresponds to the DTC vector number i. 0: No effect 1: DTC Enable by Event number i	R/W
21:20	—	These bits are read as 0. The write value should be 0.	R/W
23:22	SET22 to SET23	DTC Enable Set by Event Number i The suffix number of each bit symbol corresponds to the DTC vector number i. 0: No effect 1: DTC Enable by Event number i	R/W
26:24	—	These bits are read as 0. The write value should be 0.	R/W
31:27	SET27 to SET31	DTC Enable Set by Event Number i The suffix number of each bit symbol corresponds to the DTC vector number i. 0: No effect 1: DTC Enable by Event number i	R/W

SETi bits (DTC Enable Set by Event Number i) (i = 1 to 7, 18 to 19, 22 to 23, 27 to 31)

By writing 1 to the SETi bit, the corresponding event is selected as the DTC activation source. Writing 0 has no effect. It reads as 0.

11.2.9 DTCENSET1 : DTC Enable Set Register 1

Base address: ICU = 0x4000_6000																
Offset address: 0x0314																
Bit position:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Bit field:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	—	—	—	—	—	—	—	—	—	SET38	SET37	SET36	SET35	SET34	SET33	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
0	—	This bit is read as 0. The write value should be 0.	R/W
6:1	SET33 to SET38	DTC Enable Set by Event Number i The suffix number of each bit symbol corresponds to the DTC vector number i. 0: No effect 1: DTC Enable by Event number i	R/W
31:7	—	These bits are read as 0. The write value should be 0.	R/W

SETi bits (DTC Enable Set by Event Number i) (i = 33 to 38)

By writing 1 to the SETi bit, the corresponding event is selected as the DTC activation source. Writing 0 has no effect. It reads as 0.

少量	象征	功能	R/W
7:1	第1组至第7组	DTC 启用设置（按事件编号 i） 每个比特符号的后缀编号对应于 DTC 向量编号 i。 0: 无影响 1: 通过事件编号 i 启用 D TC	R/W
17:8	—	这些位读取为 0。写入值应为 0。	R/W
19:18	第18组至第19组	DTC 启用设置（按事件编号 i） 每个比特符号的后缀编号对应于 DTC 向量编号 i。 0: 无影响 1: 通过事件编号 i 启用 D TC	R/W
21:20	—	这些位读取为 0。写入值应为 0。	R/W
23:22	SET22 至 SET23	DTC 启用设置（按事件编号 i） 每个比特符号的后缀编号对应于 DTC 向量编号 i。 0: 无影响 1: 通过事件编号 i 启用 D TC	R/W
26:24	—	这些位读取为 0。写入值应为 0。	R/W
31:27	SET27 至 SET31	DTC 启用设置（按事件编号 i） 每个比特符号的后缀编号对应于 DTC 向量编号 i。 0: 无影响 1: 通过事件编号 i 启用 D TC	R/W

SETi 位（DTC 使能由事件编号 i 设置）(i = 1 至 7、18 至 19、22 至 23、27 至 31)

将 SETi 位写入 1，即可选择相应的事件作为 DTC 激活源。写入 0 无效，读取结果为 0。

11.2.9DTCENSET1: DTC 使能设置寄存器 1

基地址: ICU = 0x4000_6000																
偏移地址: 0x0314																
位位置:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
重置后的值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
位位置:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域:	—	—	—	—	—	—	—	—	—	SET38	SET37	SET36	SET35	SET34	SET33	—
重置后的值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
0	—	该位读取值为 0。写入值应为 0。	R/W
6:1	SET33 至 SET38	DTC 启用设置（按事件编号 i） 每个比特符号的后缀编号对应于 DTC 向量编号 i。 0: 无影响 1: 通过事件编号 i 启用 D TC	R/W
31:7	—	这些位读取为 0。写入值应为 0。	R/W

SETi 位（DTC 使能由事件编号 i 设置）(i = 33至 38)

将 SETi 位写入 1，即可选择相应的事件作为 DTC 激活源。写入 0 无效，读取结果为 0。

11.2.10 DTCENCLR0 : DTC Enable Clear Register 0

Base address: ICU = 0x4000_6000

Offset address: 0x0320

Bit position:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Bit field:	CLR31	CLR30	CLR29	CLR28	CLR27	—	—	—	CLR23	CLR22	—	—	CLR19	CLR18	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	—	—	—	—	—	—	—	—	CLR7	CLR6	CLR5	CLR4	CLR3	CLR2	CLR1	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
0	—	This bit is read as 0. The write value should be 0.	R/W
7:1	CLR1 to CLR7	DTC Enable Clear by Event Number i The suffix number of each bit symbol corresponds to the DTC vector number i. 0: No effect 1: DTC Disable by Event number i	R/W
17:8	—	These bits are read as 0. The write value should be 0.	R/W
19:18	CLR18 to CLR19	DTC Enable Clear by Event Number i The suffix number of each bit symbol corresponds to the DTC vector number i. 0: No effect 1: DTC Disable by Event number i	R/W
21:20	—	These bits are read as 0. The write value should be 0.	R/W
23:22	CLR22 to CLR23	DTC Enable Clear by Event Number i The suffix number of each bit symbol corresponds to the DTC vector number i. 0: No effect 1: DTC Disable by Event number i	R/W
26:24	—	These bits are read as 0. The write value should be 0.	R/W
31:27	CLR27 to CLR31	DTC Enable Clear by Event Number i The suffix number of each bit symbol corresponds to the DTC vector number i. 0: No effect 1: DTC Disable by Event number i	R/W

CLRi bits (DTC Enable Clear by Event Number i)(i = 1 to 7, 18 to 19, 22 to 23, 27 to 31)

Writing 1 to the CLRi bit disables DTC activation by the corresponding event. Writing 0 has no effect. It reads as 0.

11.2.11 DTCENCLR1 : DTC Enable Clear Register 1

Base address: ICU = 0x4000_6000

Offset address: 0x0324

Bit position:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Bit field:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	—	—	—	—	—	—	—	—	—	CLR38	CLR37	CLR36	CLR35	CLR34	CLR33	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

11.2.10 DTCENCLR0: DTC 使能清除寄存器 0

基地址: ICU = 0x4000_6000

偏移地址: 0x0320

位位置:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域:	CLR31	CLR30	CLR29	CLR28	CLR27	—	—	—	CLR23	CLR22	—	—	CLR19	CLR18	—	—
重置后的值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
位位置:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域:	—	—	—	—	—	—	—	—	—	CLR7	CLR6	CLR5	CLR4	CLR3	CLR2	CLR1
重置后的值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
0	—	该位读取值为 0。写入值应为 0。	R/W
7:1	CLR1 至 CLR7	DTC 启用清除（按事件编号 i） 每个比特符号的后缀编号对应于 DTC 向量编号 i。 0: 无影响 1: 按事件编号 i 禁用 D TC	R/W
17:8	—	这些位读取为 0。写入值应为 0。	R/W
19:18	CLR18 至 CLR19	DTC 启用清除（按事件编号 i） 每个比特符号的后缀编号对应于 DTC 向量编号 i。 0: 无影响 1: 按事件编号 i 禁用 D TC	R/W
21:20	—	这些位读取为 0。写入值应为 0。	R/W
23:22	CLR22 至 CLR23	DTC 启用清除（按事件编号 i） 每个比特符号的后缀编号对应于 DTC 向量编号 i。 0: 无影响 1: 按事件编号 i 禁用 D TC	R/W
26:24	—	这些位读取为 0。写入值应为 0。	R/W
31:27	CLR27 至 CLR31	DTC 启用清除（按事件编号 i） 每个比特符号的后缀编号对应于 DTC 向量编号 i。 0: 无影响 1: 按事件编号 i 禁用 D TC	R/W

CLRi 位（DTC 使能清除，按事件编号 i）(i = 1 至 7、18 至 19、22 至 23、27 至 31)

向 CLRi 位写入 1 会禁用相应事件触发的 DTC。写入 0 无效，读取结果为 0。

11.2.11 DTCENCLR1: DTC 使能清除寄存器 1

基本地址: ICU = 0x4000_6000

偏移地址: 0x0324

位位置:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
重置后的值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
位位置:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域:	—	—	—	—	—	—	—	—	—	CLR38	CLR37	CLR36	CLR35	CLR34	CLR33	—
重置后的值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
0	—	This bit is read as 0. The write value should be 0.	R/W
6:1	CLR33 to CLR38	DTC Enable Clear by Event Number i The suffix number of each bit symbol corresponds to the DTC vector number i. 0: No effect 1: DTC Disable by Event number i	R/W
31:7	—	These bits are read as 0. The write value should be 0.	R/W

CLRi bits (DTC Enable Clear by Event Number i) (i = 33 to 38)

Writing 1 to the CLRi bit disables DTC activation by the corresponding event. Writing 0 has no effect. It reads as 0.

11.2.12 INTFLAG0 : Interrupt Request Flag Monitor Register 0

Base address: ICU = 0x4000_6000																
Offset address: 0x0330																
Bit position:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Bit field:	IF31	IF30	IF29	IF28	IF27	IF26	IF25	IF24	IF23	IF22	IF21	IF20	IF19	IF18	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	—	—	—	—	IF11	IF10	—	—	IF7	IF6	IF5	IF4	IF3	IF2	IF1	IF0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
7:0	IF0 to IF7	Interrupt Request Flag Monitor 0: Interrupt request of event number i is not being accepted by the ICU 1: Interrupt request of event number i is being accepted by the ICU	R
9:8	—	These bits are read as 0.	R
11:10	IF10 to IF11	Interrupt Request Flag Monitor 0: Interrupt request of event number i is not being accepted by the ICU 1: Interrupt request of event number i is being accepted by the ICU	R
17:12	—	These bits are read as 0.	R
31:18	IF18 to IF31	Interrupt Request Flag Monitor 0: Interrupt request of event number i is not being accepted by the ICU 1: Interrupt request of event number i is being accepted by the ICU	R

IFi flags (Interrupt Request Flag Monitor) (i = 0 to 7, 10 to 11, 18 to 31)

The IFi flag indicates whether an interrupt request or a DTC request of the event i is being accepted by the ICU or not. This register is read-only.

[Setting condition]

- When an interrupt request or a DTC activation request occurs.

[Clearing condition]

- When the ICU notifies the interrupt request to the NVIC.
- When a DTC transfer that does not notify the CPU of an interrupt started.
 - MRB.DISEL = 0 and Remaining transfer operations ≠ 0
- When a DTC transfer finished.
 - MRB.DISEL = 0 and Remaining transfer operations = 0
 - MRB.DISEL = 1
- When 1 is written to the DTCENSETn.SETi bit or DTCENCLRn.CLRi bit.

少量	象征	功能	R/W
0	—	该位读取值为 0。写入值应为 0。	R/W
6:1	CLR33 至 CLR38	DTC 启用清除（按事件编号 i） 每个比特符号的后缀编号对应于 DTC 向量编号 i。 0: 无影响 1: 按事件编号 i 禁用 D TC	R/W
31:7	—	这些位读取为 0。写入值应为 0。	R/W

CLRi 位（通过事件编号 i 清除 DTC 使能）(i = 33至 38)

向 CLRi 位写入 1 会禁用相应事件触发的 DTC。写入 0 无效，读取结果为 0。

11.2.12 INTFLAG0: 中断请求标志监视器寄存器 0

基地址: ICU = 0x4000_6000																
偏移地址: 0x0330																
位位置:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域:	IF31	IF30	IF29	IF28	IF27	IF26	IF25	IF24	IF23	IF22	IF21	IF20	IF19	IF18	—	—
重置后的值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
位位置:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域:	—	—	—	—	IF11	IF10	—	—	IF7	IF6	IF5	IF4	IF3	IF2	IF1	IF0
重置后的值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
7:0	IF0 至 IF7	中断请求标志监视器 0: ICU 不接受事件编号 i 的中断请求 1: ICU 正在接受事件编号为 i 的中断请求。	R
9:8	—	这些位读取为 0。	R
11:10	IF10 至 IF11	中断请求标志监视器 0: ICU 不接受事件编号 i 的中断请求 1: ICU 正在接受事件编号为 i 的中断请求。	R
17:12	—	这些位读取为 0。	R
31:18	IF18 至 IF31	中断请求标志监视器 0: ICU 不接受事件编号 i 的中断请求 1: ICU 正在接受事件编号为 i 的中断请求。	R

IFi 标志（中断请求标志监视器）(i = 0至 7、10 至 11、18 至 31)

IFi 标志指示 ICU 是否接受事件 i 的中断请求或 DTC 请求。此寄存器为只读寄存器。

[设置条件]

- 当发生中断请求或DTC激活请求时。

[清除条件]

- 当ICU向NVIC发出中断请求时。
- 当 DTC 传输未通知 CPU 中断时开始。
 - MRB.DISEL= 0和剩余的转移操作≠ 0
- 当DTC转账完成后。
 - MRB.DISEL= 0和剩余的转移操作= 0
 - MRB.DISEL = 1
- 当 1 写入 DTCENSETn.SETi 位或 DTCENCLRn.CLRi 位时。

To clear the interrupt request flag, write 1 to the DTCENSETn.SETi bit or DTCENCLRn.CLRi bit.

By using the INTFLAG0 register and the DTCENST0 register, pending DTC requests can be confirmed. If the DTCENST0.STi = 1, the INTFLAG0.IFi = 1 and the DTCSTS.VECN[7:0] ≠ i, the DTC request of the event i is pending.

11.2.13 INTFLAG1 : Interrupt Request Flag Monitor Register 1

Base address: ICU = 0x4000_6000

Offset address: 0x0334

Bit position:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Bit field:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	—	—	—	—	—	—	—	—	—	IF38	IF37	IF36	IF35	IF34	IF33	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
0	—	This bit is read as 0.	R
6:1	IF33 to IF38	Interrupt Request Flag Monitor 0: Interrupt source of event number i is not accepted by the ICU 1: Interrupt source of event number i is accepted by the ICU	R
31:7	—	These bits are read as 0.	R

IFi flags (Interrupt Request Flag Monitor) (i = 33 to 38)

The IFi flag indicates whether an interrupt request or a DTC request of the event i is being accepted by the ICU or not. This register is read-only.

[Setting condition]

- When an interrupt request or the DTC activation request occurs.

[Clearing condition]

- When the ICU notifies the interrupt request to the NVIC.
- When a DTC transfer that does not notify the CPU of an interrupt started.
 - MRB.DISEL = 0 and Remaining transfer operations ≠ 0
- When a DTC transfer finished.
 - MRB.DISEL = 0 and Remaining transfer operations = 0
 - MRB.DISEL = 1
- When 1 is written to the DTCENSETn.SETi bit or DTCENCLRn.CLRi bit.

To clear the interrupt request flag, write 1 to the DTCENSETn.SETi bit or DTCENCLRn.CLRi bit.

By using the INTFLAG1 register and the DTCENST1 register, pending DTC requests can be confirmed. If the DTCENST1.STi = 1, the INTFLAG1.IFi = 1 and the DTCSTS.VECN[7:0] ≠ i, the DTC request of the event i is pending.

要清除中断请求标志，请向 DTCENSETn.SETi 位或 DTCENCLRn.CLRi 位写入 1。

通过使用 INTFLAG0 寄存器和 DTCENST0 寄存器，可以确认待处理的 DTC 请求。如果 DTCENST0.STi = 1、INTFLAG0.IFi = 1和 DTCSTS.VECN[7:0] ≠ i, 都为真，则事件 i 的 DTC 请求处于待处理状态。

11.2.13INTFLAG1：中断请求标志监视器寄存器 1

基地址：ICU = 0x4000_6000																
偏移地址：0x0334																
位位置：	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域：	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
重置后的值：	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
位位置：	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域：	—	—	—	—	—	—	—	—	—	IF38	IF37	IF36	IF35	IF34	IF33	—
重置后的值：	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
0	—	该位读取值为 0。	R
6:1	IF33 至 IF38	中断请求标志监视器 0: ICU 不接受事件编号 i 的中断源 1: ICU 接受了事件编号 i 的中断源。	R
31:7	—	这些位读取为 0。	R

IFi 标志（中断请求标志监视器）(i = 33至 38)

IFi 标志指示 ICU 是否接受事件 i 的中断请求或 DTC 请求。此寄存器为只读寄存器。

[设置条件]

- 当发生中断请求或DTC激活请求时。

[清除条件]

- 当ICU向NVIC发出中断请求时。
- 当 DTC 传输未通知 CPU 中断时开始。
 - MRB.DISEL = 0和剩余的转账操作≠ 0
- 当DTC转账完成后。
 - MRB.DISEL= 0和剩余的转移操作= 0
 - MRB.DISEL = 1
- 当 1 写入 DTCENSETn.SETi 位或 DTCENCLRn.CLRi 位时。

要清除中断请求标志，请向 DTCENSETn.SETi 位或 DTCENCLRn.CLRi 位写入 1。

通过使用 INTFLAG1 寄存器和 DTCENST1 寄存器，可以确认待处理的 DTC 请求。如果 DTCENST1.STi = 1、INTFLAG1.IFi = 1和DTCSTS.VECN[7:0] ≠ i, 都为真，则事件 i 的 DTC 请求处于待处理状态。

11.2.14 SBYEDCR0 : Software Standby/Snooze End Control Register 0

Base address: ICU = 0x4000_6000

Offset address: 0x0340

Bit position:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Bit field:	ADC10ED	—	—	—	UART0RXED	IICA0ED	—	—	—	—	—	UART0ERRED	—	SPI00RXED	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	—	—	—	—	—	DTCED	—	—	IRQ5ED	IRQ4ED	IRQ3ED	IRQ2ED	IRQ1ED	IRQ0ED	LVD1ED	IWDTED
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
0	IWDTED	IWDT Interrupt Software Standby/Snooze Mode Returns Enable 0: Software Standby/Snooze Mode returns by IWDT interrupt disabled 1: Software Standby/Snooze Mode returns by IWDT interrupt enabled	R/W
1	LVD1ED	LVD1 Interrupt Software Standby/Snooze Mode Returns Enable 0: Software Standby/Snooze Mode returns by LVD1 interrupt disabled 1: Software Standby/Snooze Mode returns by LVD1 interrupt enabled	R/W
7:2	IRQiED (i = 0 to 5)	IRQ Interrupt Software Standby/Snooze Mode Returns Enable The suffix number of each bit symbol corresponds to the IRQ number i. 0: Software Standby/Snooze Mode returns by IRQi interrupt disabled 1: Software Standby/Snooze Mode returns by IRQi interrupt enabled	R/W
9:8	—	These bits are read as 0. The write value should be 0.	R/W
10	DTCED	DTC Transfer Complete Interrupt Snooze Mode Returns Enable 0: Snooze Mode returns by DTC transfer complete interrupt disabled 1: Snooze Mode returns by DTC transfer complete interrupt enabled	R/W
17:11	—	These bits are read as 0. The write value should be 0.	R/W
18	SPI00RXED	SPI00 Transfer End or Buffer Empty Interrupt Snooze Mode Returns Enable 0: Snooze Mode returns by SPI00 transfer end or buffer empty interrupt disabled 1: Snooze Mode returns by SPI00 transfer end or buffer empty interrupt enabled	R/W
19	—	This bit is read as 0. The write value should be 0.	R/W
20	UART0ERRED	UART0 Reception Communication Error Occurrence Interrupt Snooze Mode Returns Enable 0: Snooze Mode returns by UART0 reception communication error occurrence interrupt disabled 1: Snooze Mode returns by UART0 reception communication error occurrence interrupt enabled	R/W
25:21	—	These bits are read as 0. The write value should be 0.	R/W
26	IICA0ED	IICA0 Address Match Interrupt Software Standby/Snooze Mode Returns Enable 0: Software Standby/Snooze Mode returns by IICA0 address match interrupt disabled 1: Software Standby/Snooze Mode returns by IICA0 address match interrupt enabled	R/W
27	UART0RXED	UART0 Reception Transfer End Interrupt Snooze Mode Returns Enable 0: Snooze Mode returns by UART0 reception transfer end interrupt disabled 1: Snooze Mode returns by UART0 reception transfer end interrupt enabled	R/W
30:28	—	These bits are read as 0. The write value should be 0.	R/W
31	ADC10ED	End of A/D Conversion Interrupt Snooze Mode Returns Enable 0: Snooze Mode returns by End of A/D conversion interrupt disabled 1: Snooze Mode returns by End of A/D conversion interrupt enabled	R/W

The bits in this register control whether the associated interrupt can wake up the CPU from Software Standby/Snooze Mode.

11.2.14 SBYEDCR0: 软件待机/休眠结束控制寄存器 0

基地址: ICU = 0x4000_6000

偏移地址: 0x0340

位位置:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域:	ADC10ED	—	—	—	UART0RXED	IICA0ED	—	—	—	—	—	UART0ERRED	—	SPI00RXED	—	—
重置后的值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
位位置:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域:	—	—	—	—	—	DTCED	—	—	IRQ5ED	IRQ4ED	IRQ3ED	IRQ2ED	IRQ1ED	IRQ0ED	LVD1ED	IWDTED
重置后的值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
0	IWDTED	IWDT中断软件待机/休眠模式返回使能 0: 软件待机/休眠模式通过 IWDT 中断返回已禁用 1: 软件待机/休眠模式通过 IWDT 中断返回已启用	R/W
1	LVD1ED	LVD1 中断软件待机/休眠模式返回使能 0: 通过 LVD1 中断返回软件待机/休眠模式已禁用 1: 通过 LVD1 中断使能返回软件待机/休眠模式	R/W
7:2	IRQiED(i = 0至 5)	IRQ 中断软件待机/休眠模式返回使能。每个位符号的后缀编号对应于 IRQ 编号 i。 0: 禁用通过 IRQi 中断返回软件待机/休眠模式; 1: 启用通过 IRQi 中断返回软件待机/休眠模式。	R/W
9:8	—	这些位读取为 0。写入值应为 0。	R/W
10	DTCED	DTC传输完成中断休眠模式返回使能 0: DTC传输完成中断后, 睡眠模式返回已禁用 1: DTC传输完成中断使能返回睡眠模式	R/W
17:11	—	这些位读取为 0。写入值应为 0。	R/W
18	SPI00RXED	SPI00 传输结束或缓冲区为空中断休眠模式返回使能 0: 休眠模式通过 SPI00 传输结束或缓冲区为空返回, 中断已禁用 1: 当 SPI00 传输结束或缓冲区为空时, 休眠模式返回, 中断已启用	R/W
19	—	该位读取值为 0。写入值应为 0。	R/W
20	UART0ERRED	UART0 接收通信错误发生中断休眠模式返回使能 0: 睡眠模式通过 UART0 接收通信错误返回, 中断已禁用 1: 通过 UART0 接收通信错误发生中断返回休眠模式	R/W
25:21	—	这些位读取为 0。写入值应为 0。	R/W
26	IICA0ED	IICA0 地址匹配中断软件待机/休眠模式返回使能 0: 通过 IICA0 地址匹配中断返回的软件待机/休眠模式已禁用 1: 软件待机/休眠模式通过 IICA0 地址匹配中断返回, 已启用	R/W
27	UART0RXED	UART0 接收传输结束中断休眠模式返回使能 0: 睡眠模式通过 UART0 接收传输结束中断返回, 已禁用 1: 通过 UART0 接收传输结束中断返回休眠模式	R/W
30:28	—	这些位读取为 0。写入值应为 0。	R/W
31	ADC10ED	A/D转换结束中断休眠模式返回使能 0: 暂停模式在 A/D 转换结束时返回, 中断已禁用 1: A/D转换结束中断使能返回休眠模式	R/W

该寄存器中的位控制相关中断是否可以将 CPU 从软件待机/休眠模式唤醒。

IWDTED bit (IWDT Interrupt Software Standby/Snooze Mode Returns Enable)

The IWDTED bit enables the use of IWDT interrupts to cancel Software Standby/Snooze Mode.

LVD1ED bit (LVD1 Interrupt Software Standby/Snooze Mode Returns Enable)

The LVD1ED bit enables the use of LVD1 interrupts to cancel Software Standby/Snooze Mode.

IRQiED bits (IRQ Interrupt Software Standby/Snooze Mode Returns Enable) (i = 0 to 5)

The IRQiED bits enable the use of IRQn interrupts to cancel Software Standby/Snooze Mode.

DTCED bit (DTC Transfer Complete Interrupt Snooze Mode Returns Enable)

The DTCED bits enable the use of DTC transfer complete interrupts to cancel Snooze Mode.

SPI00RXED bit (SPI00 Transfer End or Buffer Empty Interrupt Snooze Mode Returns Enable)

The SPI00RXED bits enable the use of SPI00 transfer end or buffer empty interrupts to cancel Snooze Mode.

UART0ERRED bit (UART0 Reception Communication Error Occurrence Interrupt Snooze Mode Returns Enable)

The UART0ERRED bits enable the use of UART0 reception communication error occurrence interrupts to cancel Snooze Mode.

IICA0ED bit (IICA0 Address Match Interrupt Software Standby/Snooze Mode Returns Enable)

The IICA0ED bit enables the use of IICA0 interrupts to cancel Software Standby/Snooze Mode.

UART0RXED bit (UART0 Reception Transfer End Interrupt Snooze Mode Returns Enable)

The UART0RXED bits enable the use of UART0 reception transfer end interrupts to cancel Snooze Mode.

ADC10ED bit (End of A/D Conversion Interrupt Snooze Mode Returns Enable)

The ADC10ED bits enable the use of End of A/D conversion interrupts to cancel Snooze Mode.

11.2.15 SBYEDCR1 : Software Standby/Snooze End Control Register 1

Base address: ICU = 0x4000_6000															
Offset address: 0x0344															
Bit position: 31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16															
Bit field: — — — — — — — — — — — — — — —															
Value after reset: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0															
Bit position: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0															
Bit field: — — — — — — — — — — — — ITLED —															
Value after reset: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0															

Bit	Symbol	Function	R/W
0	—	This bit is read as 0. The write value should be 0.	R/W
1	ITLED	Interval Signal of 32-bit Interval Timer Interrupt Software Standby/Snooze Mode Returns Enable 0: Software Standby/Snooze Mode returns by Interval signal of 32-bit interval timer interrupt disabled 1: Software Standby/Snooze Mode returns by Interval signal of 32-bit interval timer interrupt enabled	R/W
31:2	—	These bits are read as 0. The write value should be 0.	R/W

The bits in this register control whether the associated interrupt can wake up the CPU from Software Standby/Snooze Mode.

IWDTED 位 (IWDT 中断软件待机/休眠模式返回使能)

IWDTED 位允许使用 IWDT 中断来取消软件待机/休眠模式。

LVD1ED 位 (LVD1 中断软件待机/休眠模式返回使能)

LVD1ED 位允许使用 LVD1 中断来取消软件待机/休眠模式。

IRQiED 位 (IRQ 中断软件待机/休眠模式返回使能) (i = 0至 5)

IRQiED 位允许使用 IRQn 中断来取消软件待机/休眠模式。

DTCED 位 (DTC 传输完成中断休眠模式返回使能)

DTCED 位允许使用 DTC 传输完成中断来取消睡眠模式。

SPI00RXED 位 (SPI00 传输结束或缓冲区为空中断休眠模式返回使能)

SPI00RXED 位允许使用 SPI00 传输结束或缓冲区空中断来取消休眠模式。

UART0ERRED 位 (UART0 接收通信错误发生中断休眠模式返回使能够)

UART0ERRED 位允许使用 UART0 接收通信错误发生中断来取消休眠模式。

IICA0ED 位 (IICA0 地址匹配中断软件待机/休眠模式返回使能)

IICA0ED 位允许使用 IICA0 中断取消软件待机/休眠模式。

UART0RXED 位 (UART0 接收传输结束中断休眠模式返回使能)

UART0RXED 位允许使用 UART0 接收传输结束中断来取消休眠模式。

ADC10ED 位 (A/D 转换中断结束休眠模式返回使能)

ADC10ED 位允许使用 A/D 转换结束中断来取消睡眠模式。

11.2.15SBYEDCR1： 软件待机/休眠结束控制寄存器 1

基地址: ICU = 0x4000_6000															
Offset address: 0x0344															
位位置: 31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16															
位域: — — — — — — — — — — — — — — —															
重置后的值: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0															
位位置: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0															
位域: — — — — — — — — — — — — ITLED —															
重置后的值: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0															

少量	象征	功能	R/W
0	—	该位读取值为 0。写入值应为 0。	R/W
1	ITLED	32 位间隔定时器中断软件待机/贪睡模式的间隔信号返回使能 0: 软件待机/休眠模式通过 32 位间隔定时器中断的间隔信号返回，已禁用 1: 软件待机/贪睡模式通过 32 位间隔定时器中断的间隔信号返回，已启用	R/W
31:2	—	这些位读取为 0。写入值应为 0。	R/W

该寄存器中的位控制相关中断是否可以将 CPU 从软件待机/休眠模式唤醒。

ITLED bit (Interval Signal of 32-bit Interval Timer Interrupt Software Standby/Snooze Mode Returns Enable)

The ITLED bit enables the use of Interval signal of 32-bit interval timer interrupts to cancel Software Standby/Snooze Mode.

11.3 Vector Table

The ICU detects maskable and non-maskable interrupts. Interrupt priorities are set up in the Arm NVIC. For information about these registers, see [section 11.9. Reference](#).

11.3.1 Interrupt Vector Table

[Table 11.3](#) describes the interrupt vector table. The interrupt vector addresses conform to the NVIC specifications.

Table 11.3 Interrupt vector table (1 of 3)

Exception number	IRQ number	Vector offset	Source	Description
0	—	0x000	Arm	Initial stack pointer
1	—	0x004	Arm	Initial program counter (reset vector)
2	—	0x008	Arm	Non-Maskable Interrupt (IWDT Underflow/Refresh Error Interrupt, Voltage monitor 1 Interrupt, NMI Pin Interrupt)
3	—	0x00C	Arm	Hard Fault
4	—	0x010	Arm	Reserved
5	—	0x014	Arm	Reserved
6	—	0x018	Arm	Reserved
7	—	0x01C	Arm	Reserved
8	—	0x020	Arm	Reserved
9	—	0x024	Arm	Reserved
10	—	0x028	Arm	Reserved
11	—	0x02C	Arm	Supervisor Call (SVCall)
12	—	0x030	Arm	Reserved
13	—	0x034	Arm	Reserved
14	—	0x038	Arm	Pendable request for system service (PendableSrvReq)
15	—	0x03C	Arm	System Tick Timer (SysTick)
16	0	0x040	IWDT_NMIUNDF	Watchdog timer interval
17	1	0x044	LVD_LVD1	Low voltage detection 1
18	2	0x048	PORT_IRQ0	External pin interrupt 0
19	3	0x04C	PORT_IRQ1	External pin interrupt 1
20	4	0x050	PORT_IRQ2	External pin interrupt 2
21	5	0x054	PORT_IRQ3	External pin interrupt 3
22	6	0x058	PORT_IRQ4	External pin interrupt 4
23	7	0x05C	PORT_IRQ5	External pin interrupt 5
24	8	0x060	Reserved	Reserved
25	9	0x064	Reserved	Reserved
26	10	0x068	DTC_COMPLETE	DTC transfer complete
27	11	0x06C	FLASH_FRDYI	Flash ready interrupt
28	12	0x070	Reserved	Reserved
29	13	0x074	Reserved	Reserved

ITLED 位（32 位间隔定时器中断软件待机/贪睡模式返回使能的间隔信号）

ITLED 位允许使用 32 位间隔定时器中断的间隔信号来取消软件待机/休眠模式。

11.3 向量表

ICU 检测可屏蔽中断和不可屏蔽中断。中断优先级在 Arm NVIC 中设置。有关这些寄存器的信息，请参阅第 11.9 节。参考。

11.3.1 中断向量表

表 11.3 描述了中断向量表。中断向量地址符合 NVIC 规范。

表 11.3 中断向量表 (1/3)

异常编号	IRQ 编号	矢量偏移	来源	描述
0	—	0x000	Arm	初始堆栈指针
1	—	0x004	Arm	初始程序计数器（复位向量）
2	—	0x008	Arm	不可屏蔽中断（IWDT 下溢/刷新错误中断），电压监视器 1 中断，NMI 引脚中断）
3	—	0x00C	Arm	硬故障
4	—	0x010	Arm	预订的
5	—	0x014	Arm	预订的
6	—	0x018	Arm	预订的
7	—	0x01C	Arm	预订的
8	—	0x020	Arm	预订的
9	—	0x024	Arm	预订的
10	—	0x028	Arm	预订的
11	—	0x02C	Arm	主管呼叫 (SVCall)
12	—	0x030	Arm	预订的
13	—	0x034	Arm	预订的
14	—	0x038	Arm	待处理的系统服务请求（PendableSrvReq）
15	—	0x03C	Arm	系统滴答定时器（SysTick）
16	0	0x040	IWDT_NMIUNDF	看门狗定时器间隔
17	1	0x044	LVD_LVD1	低电压检测 1
18	2	0x048	端口中断0	外部引脚中断 0
19	3	0x04C	端口中断1	外部引脚中断 1
20	4	0x050	PORT_IRQ2	外部引脚中断 2
21	5	0x054	端口中断3	外部引脚中断 3
22	6	0x058	端口中断4	外部引脚中断 4
23	7	0x05C	PORT_IRQ5	外部引脚中断 5
24	8	0x060	预订的	预订的
25	9	0x064	预订的	预订的
26	10	0x068	DTC_COMPLETE	DTC转账完成
27	11	0x06C	FLASH_FRDYI	闪存就绪中断
28	12	0x070	预订的	预订的
29	13	0x074	预订的	预订的

Table 11.3 Interrupt vector table (2 of 3)

Exception number	IRQ number	Vector offset	Source	Description
30	14	0x078	Reserved	Reserved
31	15	0x07C	Reserved	Reserved
32	16	0x080	Reserved	Reserved
33	17	0x084	Reserved	Reserved
34	18	0x088	SAU0_UART_TXI0/ SAU0_SPI_TXRXI00/ SAU0_IIC_TXRXI00	UART0 transmission transfer end or buffer empty interrupt/SPI00 transfer end or buffer empty interrupt/IIC00 transfer end
35	19	0x08C	TAU0_TMI00	End of timer channel 00 count or capture
36	20	0x090	SAU0_UART_ERRI0	UART0 reception communication error occurrence
37	21	0x094	TAU0_TMI01H	End of timer channel 01 count or capture (at higher 8-bit timer operation)
38	22	0x098	SAU0_UART_TXI1	UART1 transmission transfer end or buffer empty interrupt
39	23	0x09C	SAU0_UART_RXI1/ SAU0_SPI_TXRXI11/ SAU0_IIC_TXRXI11	UART1 reception transfer end/SPI11 transfer end or buffer empty interrupt/IIC11 transfer end
40	24	0x0A0	SAU0_UART_ERRI1	UART1 reception communication error occurrence
41	25	0x0A4	TAU0_TMI03H	End of timer channel 03 count or capture (at higher 8-bit timer operation)
42	26	0x0A8	IICA0_TXRXI	End of IICA0 communication
43	27	0x0AC	SAU0_UART_RXI0	UART0 reception transfer end
44	28	0x0B0	TAU0_TMI01	End of timer channel 01 count or capture (at 16-bit/lower 8-bit timer operation)
45	29	0x0B4	TAU0_TMI02	End of timer channel 02 count or capture
46	30	0x0B8	TAU0_TMI03	End of timer channel 03 count or capture (at 16-bit/lower 8-bit timer operation)
47	31	0x0BC	ADC10_ADI	End of A/D conversion
48	32	0x0C0	Reserved	Reserved
49	33	0x0C4	TML32_ITL_OR	Interval signal of 32-bit interval timer detection
50	34	0x0C8	Reserved	Reserved
51	35	0x0CC	TAU0_TMI04	End of timer channel 04 count or capture
52	36	0x0D0	TAU0_TMI05	End of timer channel 05 count or capture
53	37	0x0D4	TAU0_TMI06	End of timer channel 06 count or capture
54	38	0x0D8	TAU0_TMI07	End of timer channel 07 count or capture
55	39	0x0DC	Reserved	Reserved
56	40	0x0E0	Reserved	Reserved
57	41	0x0E4	Reserved	Reserved
58	42	0x0E8	Reserved	Reserved
59	43	0x0EC	Reserved	Reserved
60	44	0x0F0	Reserved	Reserved
61	45	0x0F4	Reserved	Reserved
62	46	0x0F8	Reserved	Reserved
63	47	0x0FC	Reserved	Reserved
64	48	0x100	Reserved	Reserved
65	49	0x104	Reserved	Reserved

表 11.3 中断向量表 (2/3)

异常编号	IRQ 编号	矢量偏移	来源	描述
30	14	0x078	预订的	预订的
31	15	0x07C	预订的	预订的
32	16	0x080	预订的	预订的
33	17	0x084	预订的	预订的
34	18	0x088	SAU0_UART_TXI0/ SAU0_SPI_TXRXI00/ SAU0_IIC_TXRXI00	UART0 传输结束或缓冲区为空中断/SPI00 传输结束或缓冲区为空中断/IIC00 传输结束
35	19	0x08C	TAU0_TMI00	定时器通道 00 计数或捕获结束
36	20	0x090	SAU0_UART_ERRI0	UART0接收通信错误发生
37	21	0x094	TAU0_TMI01H	定时器通道 01 计数或捕获结束（在高 8 位定时器操作时）
38	22	0x098	SAU0_UART_TXI1	UART1 传输结束或缓冲区为空中断
39	23	0x09C	SAU0_UART_RXI1/ SAU0_SPI_TXRXI11/ SAU0_IIC_TXRXI11	UART1 接收传输结束/SPI11 传输结束或缓冲区空中断/IIC11 传输结束
40	24	0x0A0	SAU0_UART_ERRI1	UART1接收通信错误发生
41	25	0x0A4	TAU0_TMI03H	定时器通道 03 计数或捕获结束（在高 8 位定时器操作时）
42	26	0x0A8	IICA0_TXRXI	IICA0 通信结束
43	27	0x0AC	SAU0_UART_RXI0	UART0 接收传输端
44	28	0x0B0	TAU0_TMI01	定时器通道 01 计数或捕获结束（16 位/低 8 位定时器操作）
45	29	0x0B4	TAU0_TMI02	定时器通道 02 计数或捕获结束
46	30	0x0B8	TAU0_TMI03	定时器通道 03 计数或捕获结束（16 位/低 8 位定时器操作）
47	31	0x0BC	ADC10_ADI	A/D转换结束
48	32	0x0C0	预订的	预订的
49	33	0x0C4	TML32_ITL_OR	32位间隔定时器检测的间隔信号
50	34	0x0C8	预订的	预订的
51	35	0x0CC	TAU0_TMI04	定时器通道 04 计数或捕获结束
52	36	0x0D0	TAU0_TMI05	定时器通道 05 计数或捕获结束
53	37	0x0D4	TAU0_TMI06	定时器通道 06 计数或捕获结束
54	38	0x0D8	TAU0_TMI07	定时器通道 07 计数或捕获结束
55	39	0x0DC	预订的	预订的
56	40	0x0E0	预订的	预订的
57	41	0x0E4	预订的	预订的
58	42	0x0E8	预订的	预订的
59	43	0x0EC	预订的	预订的
60	44	0x0F0	预订的	预订的
61	45	0x0F4	预订的	预订的
62	46	0x0F8	预订的	预订的
63	47	0x0FC	预订的	预订的
64	48	0x100	预订的	预订的
65	49	0x104	预订的	预订的

Table 11.3 Interrupt vector table (3 of 3)

Exception number	IRQ number	Vector offset	Source	Description
66	50	0x108	Reserved	Reserved
67	51	0x10C	Reserved	Reserved
68	52	0x110	Reserved	Reserved
69	53	0x114	Reserved	Reserved
70	54	0x118	Reserved	Reserved
71	55	0x11C	Reserved	Reserved
72	56	0x120	Reserved	Reserved
73	57	0x124	Reserved	Reserved
74	58	0x128	Reserved	Reserved
75	59	0x12C	Reserved	Reserved
76	60	0x130	Reserved	Reserved
77	61	0x134	Reserved	Reserved
78	62	0x138	Reserved	Reserved
79	63	0x13C	Reserved	Reserved

11.3.2 Event Number

The following table lists heading details for Table 11.5, which describes each event number.

Table 11.4 Event number

Heading	Description
Interrupt request source	Name of the source generating the interrupt request
Name	Name of the interrupt
Connect to NVIC	“✓” indicates the interrupt can be used as a CPU interrupt
Invoke DTC	“✓” indicates the interrupt can be used to request DTC activation
Canceling Snooze mode	“✓” indicates the interrupt can be used to request a return from Snooze mode
Canceling Software Standby mode	“✓” indicates the interrupt can be used to request a return from Software Standby mode

Table 11.5 Event table (1 of 2)

Event number	Interrupt request source	Name	Connect to NVIC	Invoke DTC	Canceling Snooze	Canceling Software Standby
0	IWDT	IWDT_NMIUNDF	✓	—	✓	✓
1	LVD	LVD_LVD1	✓	✓	✓	✓
2	PORT	PORT_IRQ0	✓	✓	✓	✓
3		PORT_IRQ1	✓	✓	✓	✓
4		PORT_IRQ2	✓	✓	✓	✓
5		PORT_IRQ3	✓	✓	✓	✓
6		PORT_IRQ4	✓	✓	✓	✓
7		PORT_IRQ5	✓	✓	✓	✓
8		Reserved	—	—	—	—
9		Reserved	—	—	—	—
10	DTC	DTC_COMPLETE	✓	—	✓	—
11	FLASH	FLASH_FRDYI	✓	—	—	—

表 11.3 中断向量表 (3/3)

异常编号	IRQ 编号	矢量偏移	来源	描述
66	50	0x108	预订的	预订的
67	51	0x10C	预订的	预订的
68	52	0x110	预订的	预订的
69	53	0x114	预订的	预订的
70	54	0x118	预订的	预订的
71	55	0x11C	预订的	预订的
72	56	0x120	预订的	预订的
73	57	0x124	预订的	预订的
74	58	0x128	预订的	预订的
75	59	0x12C	预订的	预订的
76	60	0x130	预订的	预订的
77	61	0x134	预订的	预订的
78	62	0x138	预订的	预订的
79	63	0x13C	预订的	预订的

11.3.2 事件编号

下表列出了表 11.5 的标题详情，该表描述了每个事件编号。

表 11.4 事件编号

标题	描述
中断请求源	生成中断请求的源的名称
Name	中断名称
连接到NVIC	“✓”表示该中断可用作 CPU 中断。
调用DTC	“”表示该中断可用于请求DTC激活
取消贪睡模式	“✓”表示该中断可用于请求从休眠模式返回
取消软件待机模式	“✓”表示该中断可用于请求从软件待机模式返回

表 11.5 事件表 (1/2)

事件编号	中断请求源	Name	连接到 NVIC	调用DTC	取消打盹	取消软件支持
0	IWDT	IWDT_NMIUNDF	✓	—	✓	✓
1	LVD	LVD_LVD1	✓	✓	✓	✓
2	PORT	端口中断0	✓	✓	✓	✓
3		端口中断1	✓	✓	✓	✓
4		PORT_IRQ2	✓	✓	✓	✓
5		端口中断3	✓	✓	✓	✓
6		端口中断4	✓	✓	✓	✓
7		PORT_IRQ5	✓	✓	✓	✓
8		预订的	—	—	—	—
9		预订的	—	—	—	—
10	DTC	DTC_COMPLETE	✓	—	✓	—
11	FLASH	FLASH_FRDYI	✓	—	—	—

Table 11.5 Event table (2 of 2)

Event number	Interrupt request source	Name	Connect to NVIC	Invoke DTC	Canceling Snooze	Canceling Software Standby
12	Reserved	Reserved	—	—	—	—
13	Reserved	Reserved	—	—	—	—
14	Reserved	Reserved	—	—	—	—
15	Reserved	Reserved	—	—	—	—
16	Reserved	Reserved	—	—	—	—
17	Reserved	Reserved	—	—	—	—
18	SAU0	SAU0_UART_TXI0/ SAU0_SPI_TXRXI00/ SAU0_IIC_TXRXI00	✓	✓	✓	—
19	TAU0	TAU0_TMI00	✓	✓	—	—
20	SAU0	SAU0_UART_ERRI0	✓	—	✓	—
21	TAU0	TAU0_TMI01H	✓	—	—	—
22	SAU0	SAU0_UART_TXI1	✓	✓	—	—
23		SAU0_UART_RXI1/ SAU0_SPI_TXRXI11/ SAU0_IIC_TXRXI11	✓	✓	—	—
24		SAU0_UART_ERRI1	✓	—	—	—
25	TAU0	TAU0_TMI03H	✓	—	—	—
26	IICA0	IICA0_TXRXI	✓	—	✓	✓
27	SAU0	SAU0_UART_RXI0	✓	✓	✓	—
28	TAU0	TAU0_TMI01	✓	✓	—	—
29		TAU0_TMI02	✓	✓	—	—
30		TAU0_TMI03	✓	✓	—	—
31	ADC10	ADC10_ADI	✓	✓	✓	—
32	Reserved	Reserved	—	—	—	—
33	TML32	TML32_ITL_OR	✓	✓	✓	✓
34		TML32_ITL0	—	✓	—	—
35	TAU0	TAU0_TMI04	✓	✓	—	—
36		TAU0_TMI05	✓	✓	—	—
37		TAU0_TMI06	✓	✓	—	—
38		TAU0_TMI07	✓	✓	—	—
39	Reserved	Reserved	—	—	—	—
40		Reserved	—	—	—	—
41		Reserved	—	—	—	—

11.4 Interrupt Operation

The ICU performs the following functions:

- Detecting interrupts
- Enabling and disabling interrupts
- Selecting interrupt request destinations such as CPU interrupt, DTC activation.

表 11.5 事件表 (2/2)

事件编号	中断请求源	Name	连接到 NVIC	调用DTC	取消打盹	取消软件支持
12	预订的	预订的	—	—	—	—
13	预订的	预订的	—	—	—	—
14	预订的	预订的	—	—	—	—
15	预订的	预订的	—	—	—	—
16	预订的	预订的	—	—	—	—
17	预订的	预订的	—	—	—	—
18	SAU0	SAU0_UART_TXI0/ SAU0_SPI_TXRXI00/ SAU0_IIC_TXRXI00	✓	✓	✓	—
19	TAU0	TAU0_TMI00	✓	✓	—	—
20	SAU0	SAU0_UART_ERRI0	✓	—	✓	—
21	TAU0	TAU0_TMI01H	✓	—	—	—
22	SAU0	SAU0_UART_TXI1	✓	✓	—	—
23		SAU0_UART_RXI1/ SAU0_SPI_TXRXI11/ SAU0_IIC_TXRXI11	✓	✓	—	—
24		SAU0_UART_ERRI1	✓	—	—	—
25	TAU0	TAU0_TMI03H	✓	—	—	—
26	IICA0	IICA0_TXRXI	✓	—	✓	✓
27	SAU0	SAU0_UART_RXI0	✓	✓	✓	—
28	TAU0	TAU0_TMI01	✓	✓	—	—
29		TAU0_TMI02	✓	✓	—	—
30		TAU0_TMI03	✓	✓	—	—
31	ADC10	ADC10_ADI	✓	✓	✓	—
32	预订的	预订的	—	—	—	—
33	TML32	TML32_ITL_OR	✓	✓	✓	✓
34		TML32_ITL0	—	✓	—	—
35	TAU0	TAU0_TMI04	✓	✓	—	—
36		TAU0_TMI05	✓	✓	—	—
37		TAU0_TMI06	✓	✓	—	—
38		TAU0_TMI07	✓	✓	—	—
39	预订的	预订的	—	—	—	—
40		预订的	—	—	—	—
41		预订的	—	—	—	—

11.4 中断操作

重症监护室 (ICU) 履行以下职能:

- 检测中断
- 启用和禁用中断
- 选择中断请求目的地，例如 CPU 中断、DTC 激活。

11.4.1 Detecting Interrupts

The ICU accepted interrupt source from a peripheral function interrupt or an external pin interrupt and sends an interrupt request to the NVIC.

External pin interrupt requests are detected by either:

- Edges (falling edge, rising edge, or rising and falling edges)

Set the IRQCn.IRQMD[1:0] bits to select the detection mode for the IRQi pins. For interrupt sources associated with peripheral modules, see Table 11.3. Events must be accepted by the NVIC before an interrupt occurs and is accepted by the CPU.

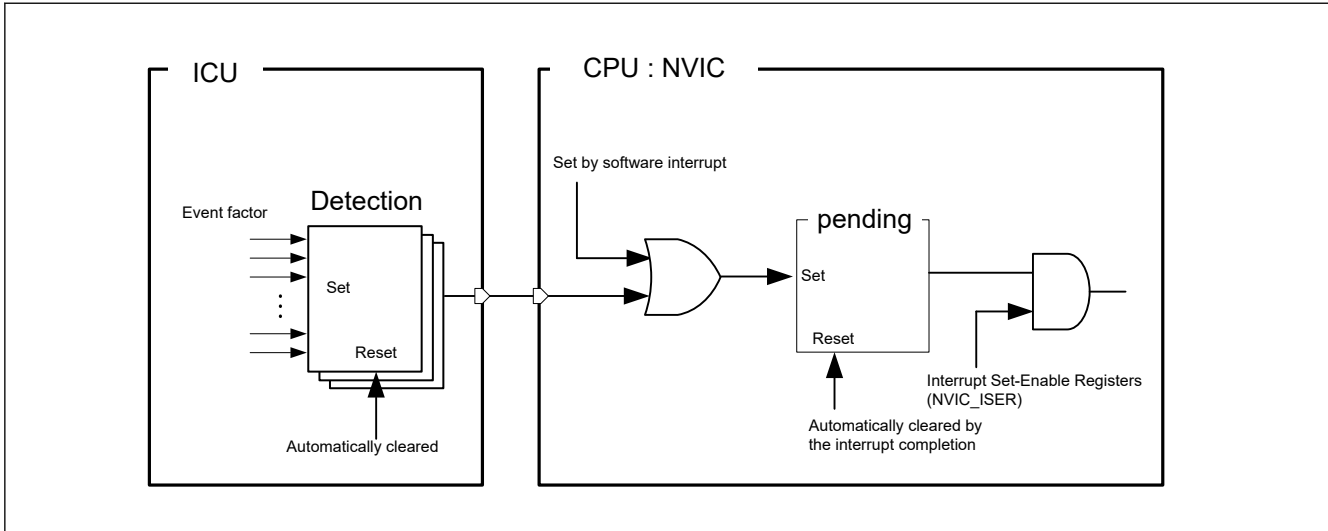


Figure 11.2 Interrupt path of the ICU and CPU (NVIC)

11.5 Interrupt setting procedure

11.5.1 Enabling Interrupt Requests

The procedure for enabling an interrupt request is as follows:

- Set the Interrupt Set-Enable register (NVIC_ISER).
- If you use the DTC, write 1 to the DTCENSETn.SETi bit to set DTCENSTn.STi bit.
- Specify the operation settings for the event source, snooze mode or software standby mode cancellation (SBYEDCRn register setting).

11.5.2 Disabling Interrupt Requests

The procedure to disable the interrupt request is as follows:

- Disable the operation settings for the event source, snooze mode or software standby mode cancellation (SBYEDCRn register setting).
- If the DTCENSTn.STi bit is set, write 1 to the DTCENCLRn.CLRi bit to clear DTCENSTn.STi bit.
- Clear the interrupt Clear-Enable register (NVIC_ICER) and interrupt Clear-Pending register (NVIC_ICPR).

11.5.3 Polling for Interrupts

The procedure for polling for interrupt requests is as follows:

- Set the Interrupt Clear-Enable register (NVIC_ICER).
- Specify the operation settings for the event source, snooze mode or software standby mode cancellation (SBYEDCRn register setting).
- Poll the interrupt Set-Pending register (NVIC_ISPR).

11.4.1 检测中断

ICU 接受来自外围功能中断或外部引脚中断的中断源，并向 NVIC 发送中断请求。

外部引脚中断请求可通过以下两种方式检测：

- 边缘（下降边缘、上升边缘或上升和下降边缘）

设置 IRQCn.IRQMD[1:0] 位以选择 IRQi 引脚的检测模式。有关与外设模块关联的中断源，请参见表 11.3。事件必须先被 NVIC 接收，然后才会发生中断并被 CPU 接收。

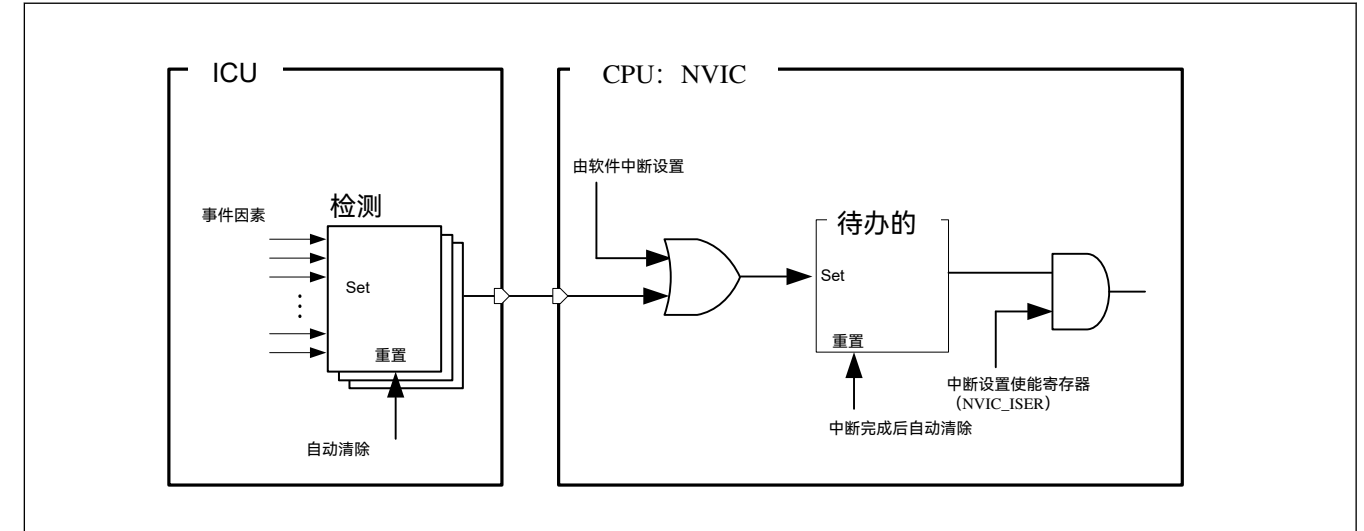


图 11.2 ICU 和 CPU 的中断路径 (NVIC)

11.5 中断设置程序

11.5.1 启用中断请求

启用中断请求的步骤如下：

- 设置中断设置使能寄存器 (NVIC_ISER)。
- 如果使用 DTC，请将 1 写入 DTCENSETn.SETi 位以设置 DTCENSTn.STi 位。
- 指定事件源、贪睡模式或软件待机模式取消的操作设置 (SBYEDCRn 寄存器设置)。

11.5.2 禁用中断请求

禁用中断请求的步骤如下：

- 禁用事件源、贪睡模式或软件待机模式取消的操作设置 (SBYEDCRn 寄存器设置)。
- 如果 DTCENSTn.STi 位已设置，则向 DTCENCLRn.CLRi 位写入 1 以清除 DTCENSTn.STi 位。
- 清除中断使能寄存器 (NVIC_ICER) 和中断挂起寄存器 (NVIC_ICPR)。

11.5.3 轮询中断

轮询中断请求的步骤如下：

- 设置中断清除使能寄存器 (NVIC_ICER)。
- 指定事件源、贪睡模式或软件待机模式取消的操作设置 (SBYEDCRn 寄存器设置)。
- 轮询中断设置挂起寄存器 (NVIC_ISPR)。

11.5.4 Selecting Interrupt Request Destinations

The available destinations are fixed for each interrupt, as described in [Table 11.3](#) and [Table 11.5](#). The interrupt output destination, CPU, or DTC can be independently selected for each interrupt source.

11.5.4.1 CPU interrupt request

When DTCENSTn.STi = 0, the interrupt is output to the NVIC.

11.5.4.2 DTC activation

When DTCENSTn.STi = 1, the interrupt is output to the DTC. Use the following procedure:

1. Write 1 to the DTCENSETn.SETi bit to DTC enable the DTCENSTn.STi bit.
2. Set the DTC Module Start bit (DTCST.DTCST) to 1.

[Table 11.6](#) shows operation when the DTC is the interrupt request destination.

Table 11.6 Operation when DTC becomes interrupt request destination

Interrupt request destination	DISEL *1	Remaining transfer operations	Operation per request	Interrupt request destination after transfer
DTC*2	1	≠ 0	DTC transfer → CPU interrupt	DTC
		= 0	DTC transfer → CPU interrupt	CPU (DTCENSTn.STi bit is automatically cleared)
	0	≠ 0	DTC transfer	DTC
		= 0	DTC transfer → CPU interrupt	CPU (DTCENSTn.STi bit is automatically cleared)

Note: When the INTFLAGn.IFi is 1, an interrupt request (DTC activation request) that occurs again is ignored.
Note 1. MRB.DISEL bit controls the interrupt generates timing from DTC to CPU.
Note 2. For chain transfers, DTC transfer continues until the last chain transfer ends. The MRB.DISEL bit state and the remaining transfer count determine whether a CPU interrupt occurs, the INTFLAGn.IFi clear timing, and the interrupt request destination after transfer. See [Table 14.2](#) in [section 14, Data Transfer Controller \(DTC\)](#).

11.5.5 External Pin Interrupts

To use external pin interrupts:

1. Configure I/O ports settings.
2. Set the IRQMD[1:0] bits of the given IRQCRi register (i = 0 to 5) to select the senses of detection.
3. Select the IRQ pin as follows:
 - If the IRQ pin is to be used for CPU interrupt requests, write 1 to the DTCENCLRn.CLRi bit to DTC disable the DTCENSTn.STi bit.
 - If the IRQ pin is to be used for DTC activation, write 1 to the DTCENSETn.SETi bit to DTC enable the DTCENSTn.STi bit.

11.6 Non-maskable Interrupt Operation

The following sources can trigger a non-maskable interrupt:

- NMI pin interrupt
- IWDT underflow/refresh error interrupt
- Voltage monitor 1 interrupt

Non-maskable interrupts can only be used with the CPU, not to activate the DTC. Non-maskable interrupts take precedence over all other interrupts. The non-maskable interrupt states can be verified in the Non-Maskable Interrupt Status Register (NMISR). Confirm that all bits in the NMISR are 0 before returning from the NMI handler.

Non-maskable interrupts are disabled by default. To use non-maskable interrupts:

11.5.4 选择中断请求目标

如表 11.3 和表 11.5 所述，每个中断的可用目标都是固定的。中断输出目标（CPU 或 DTC）可以针对每个中断源独立选择。

11.5.4.1 CPU中断请求

当 DTCENSTn.STi = 0时，中断输出到 NVIC。

11.5.4.2DTC激活

当 DTCENSTn.STi = 1时，中断信号将输出到 DTC。请使用以下步骤：

1. 将 1 写入 DTCENSETn.SETi 位，使 DTC 启用 DTCENSTn.STi 位。
2. 将 DTC 模块启动位 (DTCST.DTCST) 设置为 1。

表 11.6 显示了当 DTC 为中断请求目的地时的操作。

表 11.6 当 DTC 成为中断请求目的地时的操作

中断请求目标	DISEL *1	剩余转移操作	按请求操作	中断请求目的地在传输后
DTC*2	1	≠ 0	DTC 传输→CPU 中断	DTC
		= 0	DTC 传输→CPU 中断	CPU（DTCENSTn.STi 位自动清除）
	0	≠ 0	DTC转账	DTC
		= 0	DTC 传输→CPU 中断	CPU（DTCENSTn.STi 位自动清除）

注意：当 INTFLAGn.IFi 为 1 时，再次发生的中断请求（DTC 激活请求）将被忽略。
注1. MRB.DISEL 位控制从 DTC 到 CPU 的中断生成时序。
注2：对于链式传输，DTC传输会持续进行，直到最后一个链式传输结束。MRB.DISEL位状态和剩余传输计数决定是否发生CPU中断、INTFLAGn.IFi清除时序以及传输后的中断请求目标。
[请参阅第 14 节中的表14.2，数据传输控制器 \(DTC\)。](#)

11.5.5 外部引脚中断

要使用外部引脚中断：

1. 配置 I/O 端口设置。
2. 将给定 IRQCRi 寄存器(i = 0的 IRQMD[1:0] 位设置为 5) 以选择检测方向。
3. 按如下方式选择 IRQ 引脚：
 - 如果 IRQ 引脚用于 CPU 中断请求，则将 1 写入 DTCENCLRn.CLRi 位，以禁用 DTCENSTn.STi 位。
 - 如果 IRQ 引脚用于 DTC 激活，则将 1 写入 DTCENSETn.SETi 位以启用 DTCENSTn.STi 位。

11.6 不可屏蔽中断操作

以下来源可以触发不可屏蔽中断：

- NMI引脚中断
- IWDT 下溢/刷新错误中断
- 电压监控器 1 中断

不可屏蔽中断只能用于 CPU，不能用于激活 DTC。不可屏蔽中断的优先级高于所有其他中断。可以在不可屏蔽中断状态寄存器 (NMISR) 中验证不可屏蔽中断的状态。在从 NMI 处理程序返回之前，请确认 NMISR 中的所有位均为 0。

默认情况下，不可屏蔽中断处于禁用状态。要使用不可屏蔽中断：

1. Set the NMIMD bit of NMICR register.
2. Write 1 to the NMICLR.NMICLR bit to clear the NMISR.NMIST flag to 0.
3. Enable the non-maskable interrupt by writing 1 to the associated bit in the Non-Maskable Interrupt Enable Register (NMIER).

After 1 is written to the NMIER register, subsequent write access to the NMIEEN bit in NMIER is ignored. An NMI cannot be disabled when enabled, except by a reset.

11.7 Return from Low Power Modes

Table 11.5 lists the interrupt sources that can be used to exit Sleep or Software Standby mode. For more information, see section 9, Low Power Modes.

11.7.1 Return from Sleep Mode

To return from Sleep mode in response to an interrupt:

non-maskable interrupt

- Use the NMIER register to enable the target interrupt request.

maskable interrupt

- Select the CPU as the interrupt request destination.
- Enable the interrupt in the NVIC.

11.7.2 Return from Software Standby Mode

The ICU returns from Software Standby mode using a non-maskable interrupt or a maskable interrupt. For maskable interrupt of canceling source, see Table 11.5.

To return from Software Standby mode:

1. Select the interrupt source that enables return from Software Standby:
 - For non-maskable interrupts, use the NMIER register to enable the target interrupt request
 - For maskable interrupts, use the SBYEDCRn register to enable the target interrupt request.
2. Select the CPU as the interrupt request destination
3. Enable the interrupt in the NVIC.

Interrupt requests through the IRQn pins that do not satisfy these conditions are not detected while the clock is stopped in Software Standby mode.

11.7.3 Return from Snooze Mode

The ICU can return to Normal mode from Snooze mode using the interrupts provided for this mode.

To return to Normal mode from Snooze mode:

1. Select the CPU as the interrupt request destination.
2. Enable the interrupt in the NVIC.

Note: In Snooze mode, a clock is supplied to the ICU. If an interrupt is detected, the CPU acknowledges the interrupt after returning to Normal mode from Software Standby mode.

11.8 Using the WFI Instruction with Non-maskable Interrupts

Whenever a WFI instruction is executed, confirm that all status flags in the NMISR register are 0.

11.9 Reference

- ARM® Cortex®-M23 Processor Technical Reference Manual (ARM DDI 0550C)

1. 设置 NMICR 寄存器的 NMIMD 位。
2. 将 1 写入 NMICLR.NMICLR 位，将 NMISR.NMIST 标志清除为 0。
3. 通过向不可屏蔽中断使能寄存器 (NMIER) 中的相应位写入 1 来启用不可屏蔽中断。

当 NMIER 寄存器写入 1 后，后续对 NMIER 寄存器中 NMIEEN 位的写入操作将被忽略。NMI 启用后无法禁用，除非通过复位操作。

11.7 从低功耗模式返回

表 11.5 列出了可用于退出睡眠或软件待机模式的中断源。更多信息，请参阅第 9 节“低功耗模式”。

11.7.1 从睡眠模式恢复

响应中断而从睡眠模式返回：

不可屏蔽中断

- 使用 NMIER 寄存器启用目标中断请求。

可屏蔽中断

- 选择CPU作为中断请求目标。
- 启用NVIC中断。

11.7.2 从软件待机模式返回

ICU 通过不可屏蔽中断或可屏蔽中断从软件待机模式返回。有关取消源的可屏蔽中断，请参见表 11.5。

从软件待机模式返回：

1. 选择能够从软件待机状态返回的中断源：
 - 对于不可屏蔽中断，请使用 NMIER 寄存器启用目标中断请求
 - 对于可屏蔽中断，使用 SBYEDCRn 寄存器启用目标中断请求。
2. 选择 CPU 作为中断请求目标
3. 在 NVIC 中启用中断。

在软件待机模式下，时钟停止时，通过 IRQn 引脚发出的不满足这些条件的中断请求将不会被检测到。

11.7.3 从休眠模式返回

ICU 可以使用为该模式提供的中断从休眠模式返回正常模式。

从稍后提醒模式返回正常模式：

1. 选择 CPU 作为中断请求目标。
2. 在 NVIC 中启用中断。

注：在休眠模式下，CPU 会收到时钟信号。如果检测到中断，CPU 会在从软件待机模式返回正常模式后确认该中断。

11.8 将 WFI 指令与不可屏蔽中断一起使用

每当执行 WFI 指令时，确认 NMISR 寄存器中的所有状态标志均为 0。

11.9 参考

- ARM® Cortex®-M23 处理器技术参考手册 (ARM DDI 0550C)

12. Buses

12.1 Overview

Table 12.1 lists the bus specifications, Figure 12.1 shows the bus configuration, and Table 12.2 lists the addresses assigned for each bus.

Table 12.1 Bus specifications

Bus type		Description
Main bus	System bus (CPU)	- Connected to CPU - Connected to on-chip memory and internal peripheral bus
	DMA bus	- Connected to DTC - Connected to on-chip memory and internal peripheral bus
Slave interface	Memory bus 1	Connected to code flash memory
	Memory bus 4	Connected to SRAM
	Internal peripheral bus 1	Connected to system control related to peripheral modules
	Internal peripheral bus 3	- Connected to peripheral modules (IWDTC, MSTP and CRC) - Connected to peripheral modules (I/O Ports, ADC10, SAU0, TAU, IICA, TML32, and PCLBUZ)
	Internal peripheral bus 9	Connected to code flash memory (in P/E (Programming/Erase))

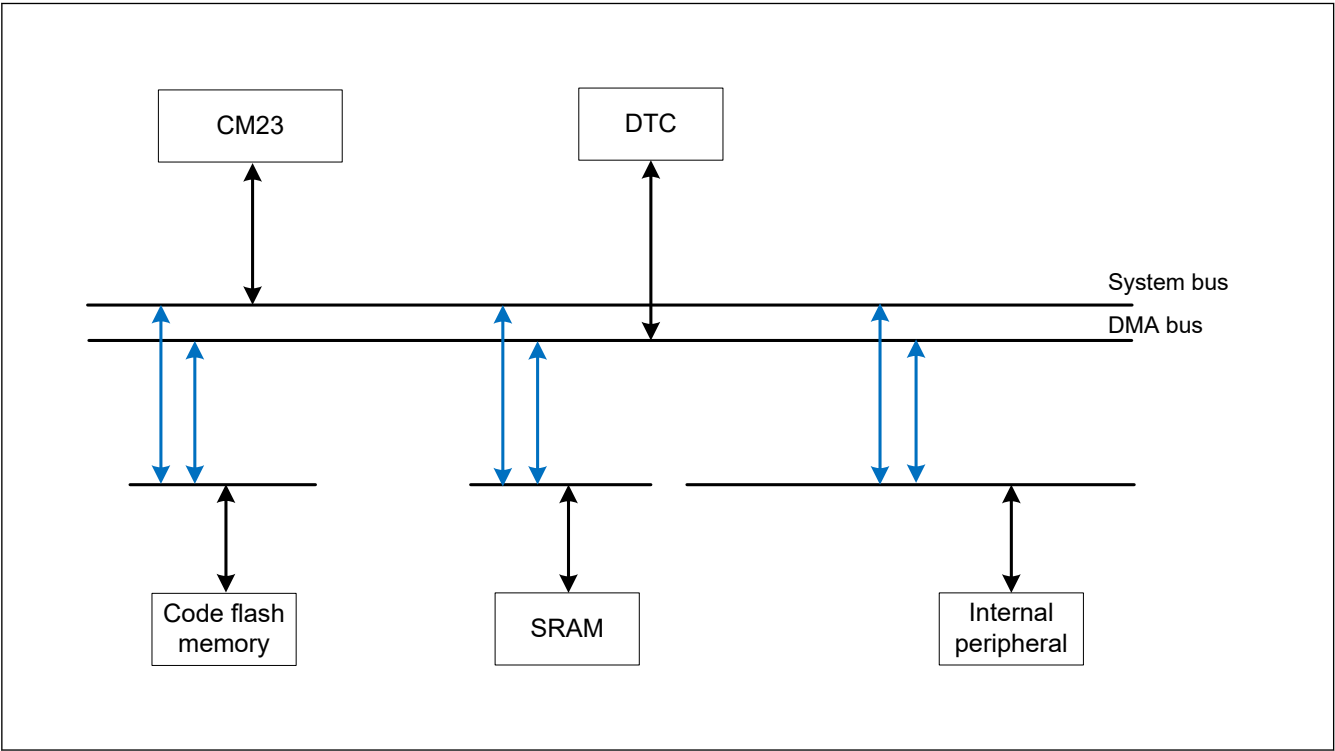


Figure 12.1 Bus configuration

Table 12.2 Addresses assigned for each bus (1 of 2)

Address	Bus	Area
0x0000_0000 to 0x01FF_FFFF	Memory bus 1	Code flash memory
0x2000_0000 to 0x2000_7FFF	Memory bus 4	SRAM
0x4000_0000 to 0x4001_8FFF	Internal peripheral bus 1	Peripheral I/O registers
0x4001_9000 to 0x4001_9FFF	Memory bus 4	MTB I/O registers

12. 公交车

12.1 概述

表 12.1 列出了总线规格，图 12.1 显示了总线配置，表 12.2 列出了分配给每条总线的地址。

表 12.1 总线规格

公交车类型		描述
主干道	系统总线（CPU）	●连接至CPU ●连接至片上存储器和内部外设总线
	DMA总线	●已连接至DTC ●连接至片上存储器和内部外设总线
从接口	内存总线 1	连接到代码闪存
	内存总线 4	连接到 SRAM
	内部外围总线 1	与外围模块相关的系统控制相连
	内部外围总线 3	●连接至外围模块（IWDTC、MSTP 和 CRC） ●连接至外围模块（I/O端口、ADC10、SAU0、TAU、IICA、TML32和PCLBUZ）
	内部外围总线 9	连接到代码闪存（在 P/E（编程/擦除）中）

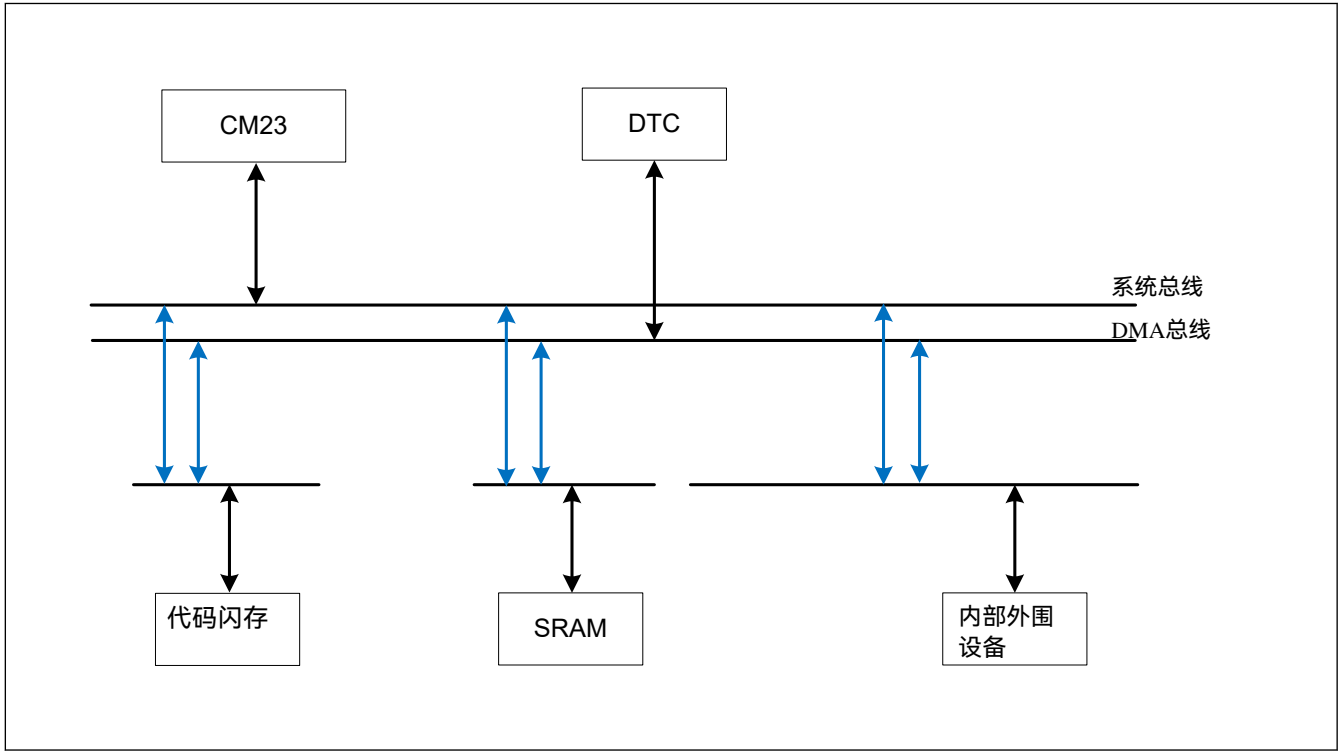


图 12.1 总线配置

表 12.2 每辆公交车的分配地址（1/2）

地址	公共汽车	Area
0x0000_0000 to 0x01FF_FFFF	内存总线 1	代码闪存
0x2000_0000 to 0x2000_7FFF	内存总线 4	SRAM
0x4000_0000 to 0x4001_8FFF	内部外围总线 1	外围I/O寄存器
0x4001_9000 to 0x4001_9FFF	内存总线 4	MTB I/O寄存器

Table 12.2 Addresses assigned for each bus (2 of 2)

Address	Bus	Area
0x4001_A000 to 0x4001_FFFF	Internal peripheral bus 1	Peripheral I/O registers
0x4004_0000 to 0x400B_FFFF	Internal peripheral bus 3	
0x4010_0000 to 0x407F_FFFF	Internal peripheral bus 9	Code flash memory (in P/E)

12.2 Description of Buses

12.2.1 Main Buses

The main buses consist of the system bus and DMA bus. The system bus and DMA bus are connected to the following:

- Code flash memory
- SRAM
- Internal peripheral bus

The system bus is used for instruction and data accesses to the CPU.

Different master and slave transfer combinations can proceed simultaneously. In addition, requests for bus access from masters other than the DTC are not accepted during reads of transfer control information for the DTC.

12.2.2 Slave Interface

For connections from the main bus to the slave interfaces, see the slave interfaces in [section 12.1. Overview](#).

Bus access from the system bus and DMA bus is arbitrated and has the following fixed priority order:

DMA bus > system bus

Different master and slave transfer combinations can proceed simultaneously.

12.2.3 Parallel Operations

Parallel operations are possible when different bus masters request access to different slave modules. [Figure 12.2](#) shows an example of parallel operations. In this example, the CPU uses the instruction and operand buses for simultaneous access to the flash and SRAM, respectively. Additionally, the DTC simultaneously uses the DMA bus for access to a peripheral bus during access to the flash and SRAM by the CPU.

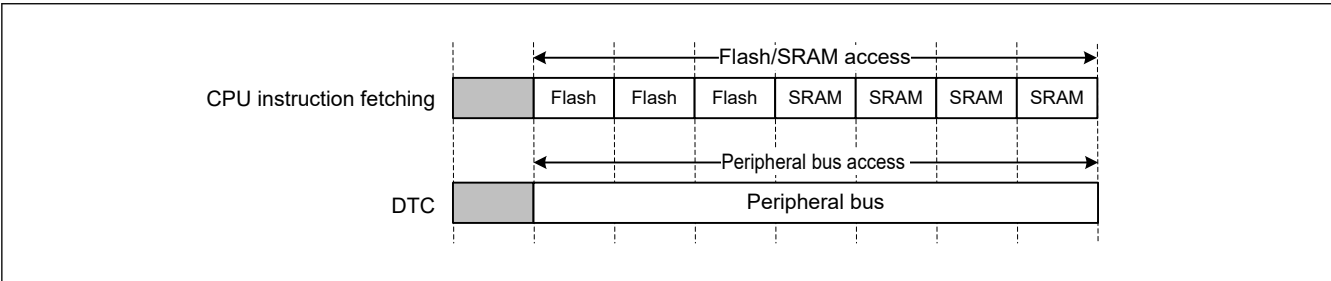


Figure 12.2 Example of parallel operations

12.2.4 Restriction on Endianness

Memory space must be little-endian to execute code on the Cortex®-M23 core.

12.2.5 Restriction on Exclusive Access

The main bus does not support exclusive transfers and there is no global monitor in the MCU. The main bus always deasserts the HEXOKAY signal (a signal in the AHB-Lite protocol) to the CPU. This means that a store exclusive instruction such as STREX instruction always gets a failed status. When an exclusive write operation is performed by the CPU, the main bus always writes the data successfully.

表 12.2 每辆公交车的分配地址 (2/2)

地址	公共汽车	Area
0x4001_A000 to 0x4001_FFFF	内部外围总线 1	外围I/O寄存器
0x4004_0000 to 0x400B_FFFF	内部外围总线 3	
0x4010_0000 to 0x407F_FFFF	内部外围总线 9	闪存代码 (P/E)

12.2 公交车描述

12.2.1 主要公交车

主总线包括系统总线和DMA总线。系统总线和DMA总线连接到以下设备:

- 代码闪存
- SRAM
- 内部外围总线

系统总线用于向 CPU 发送指令和数据。

不同的主从传输组合可以同时进行。此外，在读取DTC传输控制信息期间，不接受来自除DTC以外的其他主设备的总线访问请求。

12.2.2 从设备接口

有关从主总线到从接口的连接，请参阅第 12.1 节中的从接口概述。

系统总线和 DMA 总线的总线访问由仲裁机制决定，并具有以下固定的优先级顺序:

DMA总线>系统总线

不同的主从传输组合可以同时进行。

12.2.3 并行操作

当不同的总线主控器请求访问不同的从模块时，可以进行并行操作。图 12.2 展示了一个并行操作的示例。在该示例中，CPU 使用指令总线和操作数总线分别同时访问闪存和 SRAM。此外，在 CPU 访问闪存和 SRAM 期间，DTC 同时使用 DMA 总线访问外设总线。

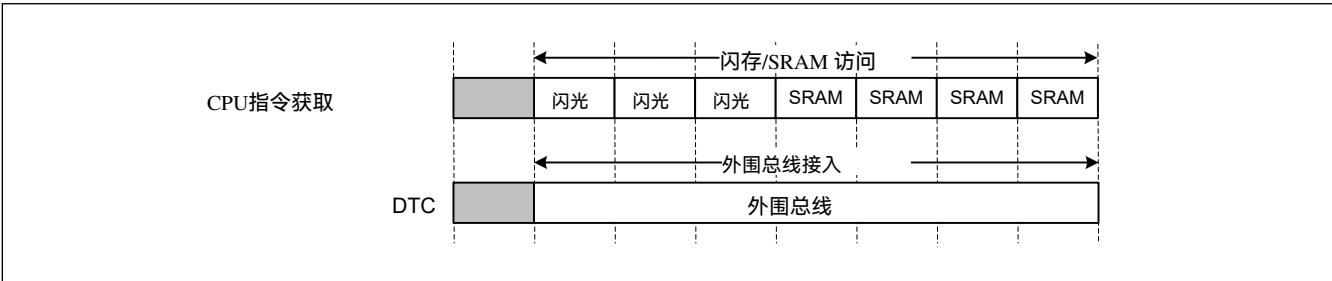


图 12.2 并行操作示例

12.2.4 对字节序的限制

要在 Cortex®-M23 内核上执行代码，内存空间必须采用小端字节序。

12.2.5 专属访问限制

主总线不支持独占传输，且MCU中没有全局监视器。主总线始终对CPU的HEXOKAY信号（AHB-Lite协议中的一个信号）保持无效状态。这意味着诸如STREX指令之类的存储独占指令始终会返回失败状态。当CPU执行独占写入操作时，主总线始终会成功写入数据。

12.3 Register Descriptions

12.3.1 BUSMCNTx : Master Bus Control Register x (x = SYS, DMA)

Base address:	BUS = 0x4000_3000															
Offset address:	0x1008 (BUSMCNTSYS) 0x100C (BUSMCNTDMA)															
Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	IERES	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
14:0	—	These bits are read as 0. The write value should be 0.	R/W
15	IERES	Ignore Error Responses 0: A bus error is reported. 1: A bus error is not reported.	R/W

Note: Changing reserved bits from the initial value of 0 is prohibited. Operation during the change is not guaranteed.

IERES bit (Ignore Error Responses)

The IERES bit enables or disables an error response of the AHB-Lite protocol.

Table 12.3 lists the registers associated with each bus type.

Table 12.3 Associations between bus types and registers

Bus type	Master Bus Control Register	Bus Error Address Register	Bus Error Status Register
System bus (CPU)	BUSMCNTSYS	BUS3ERRADD	BUS3ERRSTAT
DMA bus	BUSMCNTDMA	BUS4ERRADD	BUS4ERRSTAT

12.3.2 BUSnERRADD : Bus Error Address Register n (n = 3, 4)

Base address:	BUS = 0x4000_3000																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																												
Offset address:	0x1820 (n = 3) 0x1830 (n = 4)																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																												
Bit position:	31																															0																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																													
Bit field:	BERAD[31:0]																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																												
Value after reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Bit	Symbol	Function	R/W
31:0	BERAD[31:0]	Bus Error Address When a bus error occurs, these bits store the error address.	R

Note: BUSnERRADD is only cleared by resets. For more information, see section 5, Resets.

Table 12.3 lists the registers associated with each bus type.

BERAD[31:0] bits (Bus Error Address)

The BERAD[31:0] bits store the accessed address when a bus error occurred. For more information, see the description of the ERRSTAT flag in section 12.3.3. BUSnERRSTAT : BUS Error Status Register n (n = 3, 4) and section 12.4. Bus Error Monitoring Section.

The value of the BUSnERRADD.BERAD[31:0] bits (n = 3, 4) is valid only when the BUSnERRSTAT.ERRSTAT flag (n = 3, 4) is set to 1.

12.3 寄存器描述

12.3.1 BUSMCNTx: 主总线控制寄存器x (x = SYS, DMA)

基本地址:	BUS = 0x4000_3000															
偏移地址:	0x1008 (BUSMCNTSYS) 0x100C (BUSMCNTDMA)															
位位置:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域:	IERES	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
重置后的值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
14:0	—	这些位读取为 0。写入值应为 0。	R/W
15	IERES	忽略错误响应 0: 报告总线错误。 1: 未报告总线错误。	R/W

注意: 禁止更改保留位的初始值 (0)。更改期间的操作无法保证成功。

IERES 位 (忽略错误响应)

IERES 位启用或禁用 AHB-Lite 协议的错误响应。

表 12.3 列出了与每种总线类型相关的寄存器。

表 12.3 总线类型与寄存器之间的关联

公交车类型	主总线控制寄存器	总线错误地址寄存器	总线错误状态寄存器
系统总线 (CPU)	BUSMCNTSYS	BUS3ERRADD	BUS3ERRSTAT
DMA总线	BUSMCNTDMA	BUS4ERRADD	BUS4ERRSTAT

12.3.2BUSnERRADD: 总线错误地址寄存器 n(n = 3, 4)

基本地址:	BUS = 0x4000_3000																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																
偏移地址:	0x1820 (n = 3) 0x1830 (n = 4)																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																
位位置:	31																															0																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																	
位域:	BERAD[31:0]																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																
重置后的值:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

少量	象征	功能	R/W
31:0	BERAD[31:0]	总线错误地址 当总线发生错误时，这些位存储错误地址。	R

注意: BUSnERRADD 只能通过复位清除。更多信息，请参阅第 5 节“复位”。

表 12.3 列出了与每种总线类型相关的寄存器。

BERAD[31:0] 位 (总线错误地址)

BERAD[31:0] 位存储总线发生错误时访问的地址。有关更多信息，请参阅 12.3.3 节中 ERRSTAT 标志的描述。BUSnERRSTAT:总线错误状态寄存器 n(n = 3, 4) 和 12.4 节。总线错误监控部分。

BUSnERRADD.BERAD[31:0] 位(n = 3, 4) 的值仅在 BUSnERRSTAT.ERRSTAT 标志(n = 3, 4) 设置为 1 时有效。

12.3.3 BUSnERRSTAT : BUS Error Status Register n (n = 3, 4)

Base address:	BUS = 0x4000_3000						
Offset address:	0x1824 (n = 3) 0x1834 (n = 4)						
Bit position:	7	6	5	4	3	2	1 0
Bit field:	ERRS TAT	—	—	—	—	—	ACCS TAT
Value after reset:	0	0	0	0	0	0	x

Bit	Symbol	Function	R/W
0	ACCSTAT	Error Access Status flag Access status when the error occurred: 0: Read access 1: Write access	R
6:1	—	These bits are read as 0.	R
7	ERRSTAT	Bus Error Status flag 0: No bus error occurred. 1: Bus error occurred.	R

Note: BUSnERRSTAT is only cleared by resets. For more information, see [section 5, Resets](#).

[Table 12.3](#) lists the registers associated with each bus type.

ACCSTAT flag (Error Access Status flag)

The ACCSTAT flag indicates the access status, write or read access, when a bus error occurs. For more information, see the description of the BUSnERRSTAT.ERRSTAT flag and [section 12.4. Bus Error Monitoring Section](#).

The value is valid only when the BUSnERRSTAT.ERRSTAT flag (n = 3, 4) is set to 1.

ERRSTAT flag (Bus Error Status flag)

The ERRSTAT flag indicates whether a bus error occurred. When a bus error occurs, the access address and status of write or read access are stored. The BUSnERRSTAT.ERRSTAT flag (n = 3, 4) is set to 1.

For more information on bus errors, see [section 12.4. Bus Error Monitoring Section](#).

12.4 Bus Error Monitoring Section

The monitoring system monitors each individual area, and whenever it detects an error, it returns the error to the requesting master IP using the AHB-Lite error response protocol.

12.4.1 Error Type that Occurs by Bus

One type of errors can occur on each bus:

- Illegal address access

[section 12.4.3. Conditions for issuing illegal Address Access Errors](#) lists the address ranges where access leads to illegal address access errors. The reserved area in the slave does not trigger an illegal address access error.

12.4.2 Operation when a Bus Error Occurs

When a bus error occurs, operation is not guaranteed and the error is returned to the requesting master IP. The bus error information occurred in each master is stored in the BUSnERRADD and BUSnERRSTAT registers. These registers must be cleared by reset only. For more information, see [section 12.3.2. BUSnERRADD : Bus Error Address Register n \(n = 3, 4\)](#) and [section 12.3.3. BUSnERRSTAT : BUS Error Status Register n \(n = 3, 4\)](#).

Note: DTC does not receive bus errors. If the DTC accesses the bus, the transfer continues.

12.4.3 Conditions for issuing illegal Address Access Errors

[Table 12.4](#) lists the address spaces for each bus that issue illegal address access errors.

12.3.3BUSnERRSTAT：总线错误状态寄存器 n(n = 3, 4)

基地址：	BUS = 0x4000_3000						
Offset address:	0x1824 (n = 3) 0x1834 (n = 4)						
位位置：	7	6	5	4	3	2	1 0
位域：	ERRS TAT	—	—	—	—	—	ACCS TAT
重置后的值：	0	0	0	0	0	0	x

少量	象征	功能	R/W
0	ACCSTAT	错误访问状态标志 发生错误时的访问状态： 0：读取权限 1：写入权限	R
6:1	—	这些位读取为 0。	R
7	ERRSTAT	总线错误状态标志 0：未发生总线错误。 1：发生总线错误。	R

注意：BUSnERRSTAT 只能通过复位操作清除。更多信息，请参阅第 5 节“复位”。

表 12.3 列出了与每种总线类型相关的寄存器。

ACCSTAT 标志（错误访问状态标志）

ACCSTAT 标志指示总线发生错误时的访问状态（写入或读取访问）。有关更多信息，请参阅 BUSnERRSTAT.ERRSTAT 标志的说明和第 12.4 节“总线错误监控”部分。

仅当 BUSnERRSTAT.ERRSTAT 标志(n = 3, 4) 设置为 1 时，该值才有效。

ERRSTAT 标志（总线错误状态标志）

ERRSTAT 标志指示是否发生总线错误。发生总线错误时，会存储访问地址以及写入或读取访问的状态。BUSnERRSTAT.ERRSTAT 标志(n = 3, 4) 被设置为 1。

[有关总线错误的更多信息，请参阅第 12.4 节“总线错误监控”部分。](#)

12.4 总线错误监控部分

监控系统监控每个单独的区域，每当检测到错误时，它都会使用 AHB-Lite 错误响应协议将错误返回给请求的主 IP。

12.4.1 总线发生的错误类型

每辆公交车上都可能出现一种类型的错误：

- 非法地址访问

第 12.4.3 节“发出非法地址访问错误的条件”列出了访问会导致非法地址访问错误的地址范围。从服务器中的保留区域不会触发非法地址访问错误。

12.4.2 总线故障时的操作

当发生总线错误时，操作无法保证，并且错误信息会返回给请求主IP。每个主设备中发生的总线错误信息存储在BUSnERRADD和BUSnERRSTAT寄存器中。这些寄存器只能通过复位来清除。更多信息请参见12.3.2节。BUSnERRADD: 总线错误地址寄存器 n(n = 3, 4)和12.3.3节。BUSnERRSTAT: 总线错误状态寄存器 n(n = 3, 4)。

注意：DTC 不接收总线错误信息。如果 DTC 连接到总线，传输将继续进行。

12.4.3 发出非法地址访问错误的条件

表 12.4 列出了每条总线发出非法地址访问错误的地址空间。

Table 12.4 Conditions leading to illegal address access errors

Address	Slave bus name	Main buses	
		System bus(CPU)	DMA bus
0x0000_0000 to 0x01FF_FFFF	Memory bus 1	—	—
0x0200_0000 to 0x1FFF_FFFF	Reserved	E	E
0x2000_0000 to 0x2000_7FFF	Memory bus 4	—	—
0x2000_8000 to 0x3FFF_FFFF	Reserved	E	E
0x4000_0000 to 0x4001_8FFF	Internal peripheral bus 1	—	—
0x4001_9000 to 0x4001_9FFF	Memory bus 4	—	—
0x4001_A000 to 0x4001_FFFF	Internal peripheral bus 1	—	—
0x4002_0000 to 0x4003_FFFF	Reserved	E	E
0x4004_0000 to 0x400B_FFFF	Internal peripheral bus 3	—	—
0x400C_0000 to 0x400F_FFFF	Reserved	E	E
0x4010_0000 to 0x407F_FFFF	Internal peripheral bus 9	—	—
0x4080_0000 to 0xDFFF_FFFF	Reserved	E	E
0xE000_0000 to 0xFFFF_FFFF	System for Cortex®-M23	—	E

Note: E indicates the path where an illegal address access error occurs.
— indicates the path where an illegal address access error does not occur.

Note: The bus module detects an access error resulting from access to reserved area, for example if no area is assigned for the slave.
0x0200_0000 to 0x1FFF_FFFF: Access error detection.
0x0000_0000 to 0x01FF_FFFF: Memory bus 1 no access error detection.

12.5 References

- 1. *ARM®v8-M Architecture Reference Manual* (ARM DDI0553B.a)
- 2. *ARM® Cortex®-M23 Processor User Guide* (ARM DUI0963B)
- 3. *ARM® AMBA® 5 AHB-Lite Protocol Specification* (ARM IHI0033B.b)

表 12.4 导致非法地址访问错误的条件

地址	奴隶巴士名称	主要巴士	
		系统总线 (CPU)	DMA总线
0x0000_0000 to 0x01FF_FFFF	内存总线 1	—	—
0x0200_0000 to 0x1FFF_FFFF	预订的	E	E
0x2000_0000 to 0x2000_7FFF	内存总线 4	—	—
0x2000_8000 to 0x3FFF_FFFF	预订的	E	E
0x4000_0000 to 0x4001_8FFF	内部外围总线 1	—	—
0x4001_9000 to 0x4001_9FFF	内存总线 4	—	—
0x4001_A000 to 0x4001_FFFF	内部外围总线 1	—	—
0x4002_0000 to 0x4003_FFFF	预订的	E	E
0x4004_0000 to 0x400B_FFFF	内部外围总线 3	—	—
0x400C_0000 to 0x400F_FFFF	预订的	E	E
0x4010_0000 to 0x407F_FFFF	内部外围总线 9	—	—
0x4080_0000 to 0xDFFF_FFFF	预订的	E	E
0xE000_0000 to 0xFFFF_FFFF	适用于 Cortex 的系统®-M23	—	E

注: E 表示发生非法地址访问错误的路径。
—表示不会发生非法地址访问错误的路径。

注意: 总线模块检测到因访问保留区域而导致的访问错误, 例如, 如果未为从设备分配区域。
0x0200_0000 至 0x1FFF_FFFF: 访问错误检测。0x0000_0000至 0x01FF_FFFF:
内存总线 1 无访问错误检测。

12.5 参考文献

- 1.*ARM®v8M 架构参考手册* (ARM DDI0553B.a)
- 2.*ARM®Cortex®-M23 处理器用户指南* (ARM DUI0963B)
- 3.*ARM® AMBA® 5 AHB-Lite 协议规范* (ARM IHI0033B.b)

13. Flash Read Protection (FRP)

13.1 Overview

The MCU incorporates the Flash Read Protection (FRP) function with one secure region that includes the code flash. The secure region can be protected from read accesses. Program cannot read data in a protected region.

Table 13.1 lists the specifications of the FRP and Figure 13.1 shows a block diagram.

Table 13.1 Flash read protection specifications

Specifications	Description
Secure regions	Code flash
Address range that can be specified as protected regions	0x0000_0800 to 0x0001_FFFF
Number of protected regions	One region
How to specify the address of each protected region	Setting the address where regions start and end
Enable or disable setting of the protection	Settings enabled or disabled for the associated region

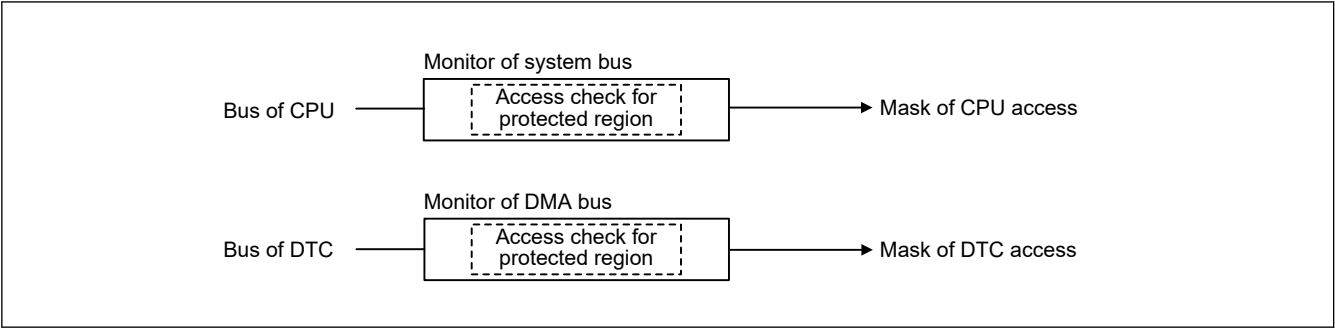


Figure 13.1 Flash read protection block diagram

13.1.1 Memory Protection

All the FRP registers are option-setting memory. Option-setting memory refers to a set of registers that are available for selecting the state of the microcontroller after a reset. The option-setting memory is allocated in the code flash. The FRP uses FRPS, FRPE, and FRPDIS bits of OFS1 register of option-setting memory. OFS1.FRPS and OFS1.FRPE define starting and ending address of a protected region. OFS1.FRPDIS disable or enable the FRP. For details of OFS1 register, see section 6, Option-Setting Memory.

The FRP protects the secured regions (the code flash memory) from being read by programs. If read access to a protected region is detected, the FRP invalidates it.

In each of the system bus and the DMA bus, there is a monitor that checks access on the bus. When OFS1.FRPDIS is set to 0, the monitor checks if access on the bus is not instruction fetch and if its address is within a protected region defined by OFS1.FRPS and OFS1.FRPE. If the access is not instruction fetch and its address is within a protected region, the monitor returns 0xFFFF_FFFF as a read value to the bus master instead of a value read from the address. If the access is instruction fetch or its address is outside a protected region, the monitor returns a value read from the address. When OFS1.FRPDIS is set to 1, the monitor does not check access on the bus.

The FRP provides access protection in the following conditions:

- Secure data is read from the CPU
- Secure data is read from other than the CPU (DTC)
- Secure data is read from the debugger.
- Secure data is accessible only by instruction fetch.

Note: Secure data:
Code flash region within the limits set up by OFS1.FRPS and OFS1.FRPE.

13.闪存读取保护（FRP）

13.1 概述

MCU 集成了闪存读取保护 (FRP) 功能，其中包含一个安全区域，该区域存储代码闪存。安全区域可防止读取访问。程序无法读取受保护区域中的数据。

表 13.1 列出了 FRP 的规格，图 13.1 显示了框图。

表 13.1 闪存读取保护规范

规格	描述
安全区域	代码闪存
可指定为受保护区域的地址范围	0x0000_0800 to 0x0001_FFFF
保护区数量	一个地区
如何指定每个受保护区域的地址	设置区域的起始和结束地址
启用或禁用保护设置	已启用或已禁用关联区域的设置

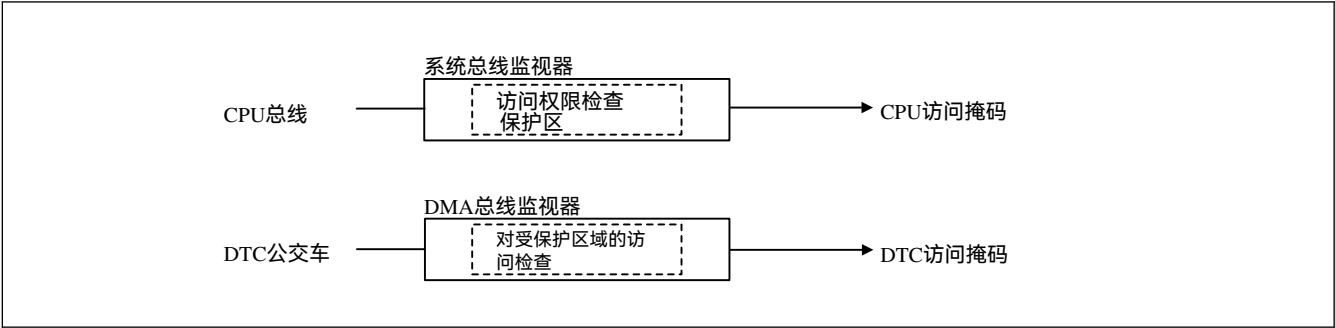


图 13.1 闪存读取保护框图

13.1.1 内存保护

所有FRP寄存器均为选项设置存储器。选项设置存储器是指一组寄存器，用于在复位后选择微控制器的状态。选项设置存储器分配在代码闪存中。FRP使用选项设置存储器OFS1寄存器的FRPS、FRPE和FRPDIS位。OFS1.FRPS和OFS1.FRPE定义了受保护区域的起始地址和结束地址。OFS1.FRPDIS用于禁用或启用FRP。有关OFS1寄存器的详细信息，请参阅第6节“选项设置存储器”。

FRP（闪存保护机制）保护安全区域（代码闪存）免受程序读取。如果检测到对受保护区域的读取访问，FRP 会使其失效。

在系统总线和DMA总线上，各有一个监视器用于检查总线访问。当OFS1.FRPDIS设置为0时，监视器会检查总线访问是否为指令提取，以及其地址是否位于OFS1.FRPS和OFS1.FRPE定义的保护区域内。如果访问不是指令提取且其地址位于保护区域内，则监视器会向总线主控返回0xFFFF_FFFF作为读取值，而不是从该地址读取的值。如果访问是指令提取或其地址位于保护区域外，则监视器会返回从该地址读取的值。当OFS1.FRPDIS设置为1时，监视器不会检查总线访问。

FRP 在以下情况下提供访问保护：

- 安全数据从CPU读取
- 安全数据是从CPU以外的其他位置读取的（DTC）
- 从调试器读取安全数据。
- 安全数据只能通过指令获取来访问。

注意：数据安全：
在 OFS1.FRPS 和 OFS1.FRPE 设定的范围内，代码闪现区域。

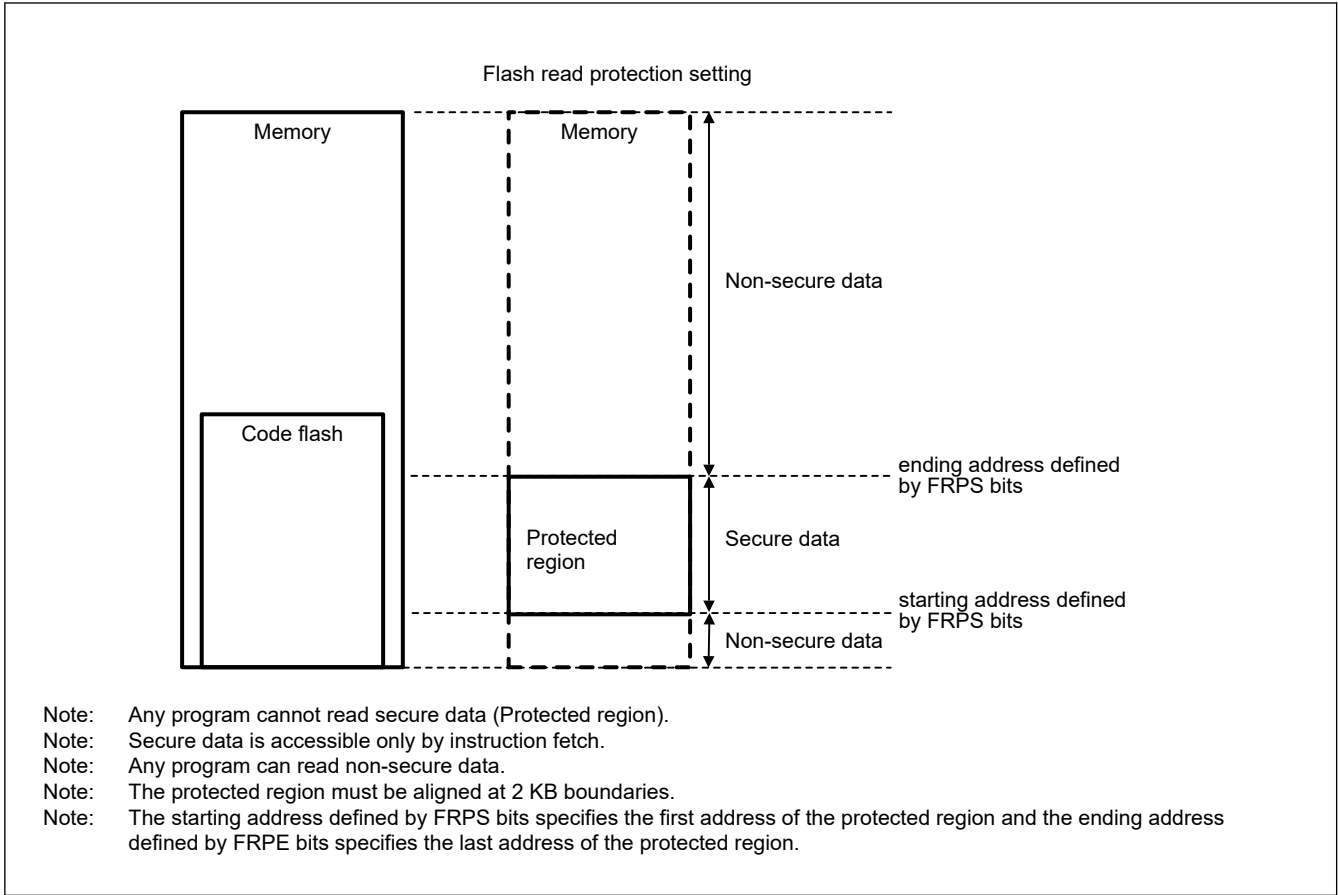


Figure 13.2 Use case of flash read protection

13.2 Usage Notes

13.2.1 Notes on the Use of a Debugger

The memory cannot be debugged if the FRP is enabled. Disable the flash read protection when debug a program, OCD debug only valid when OFS1.FRPDIS bit is 1.

13.2.2 Compiler Settings

The FRP is a kind of execute-only memory (XOM). Since data in a protected region is not readable, a protected region cannot have constant data such as literal pool. Therefore, appropriate compiler settings are required.

13.2.3 Protection of OFS1 Register

Because overwriting OFS1.FRPS[5:0], OFS1.FRPE[5:0], or OFS1.FRPDIS bits can disable the protection of FRP, the OFS1 register (address = 0x0000_0404) must be protected by the access window function. However, setting the access window function, which includes setting AWS.FSPR bit to 0, also disables the changing of AWS.BTFLG and FISR.SAS[1:0] bits used for the startup area select function. Therefore, the startup area select function is not available when using the FRP function. See for the startup area select function and for the access window function.

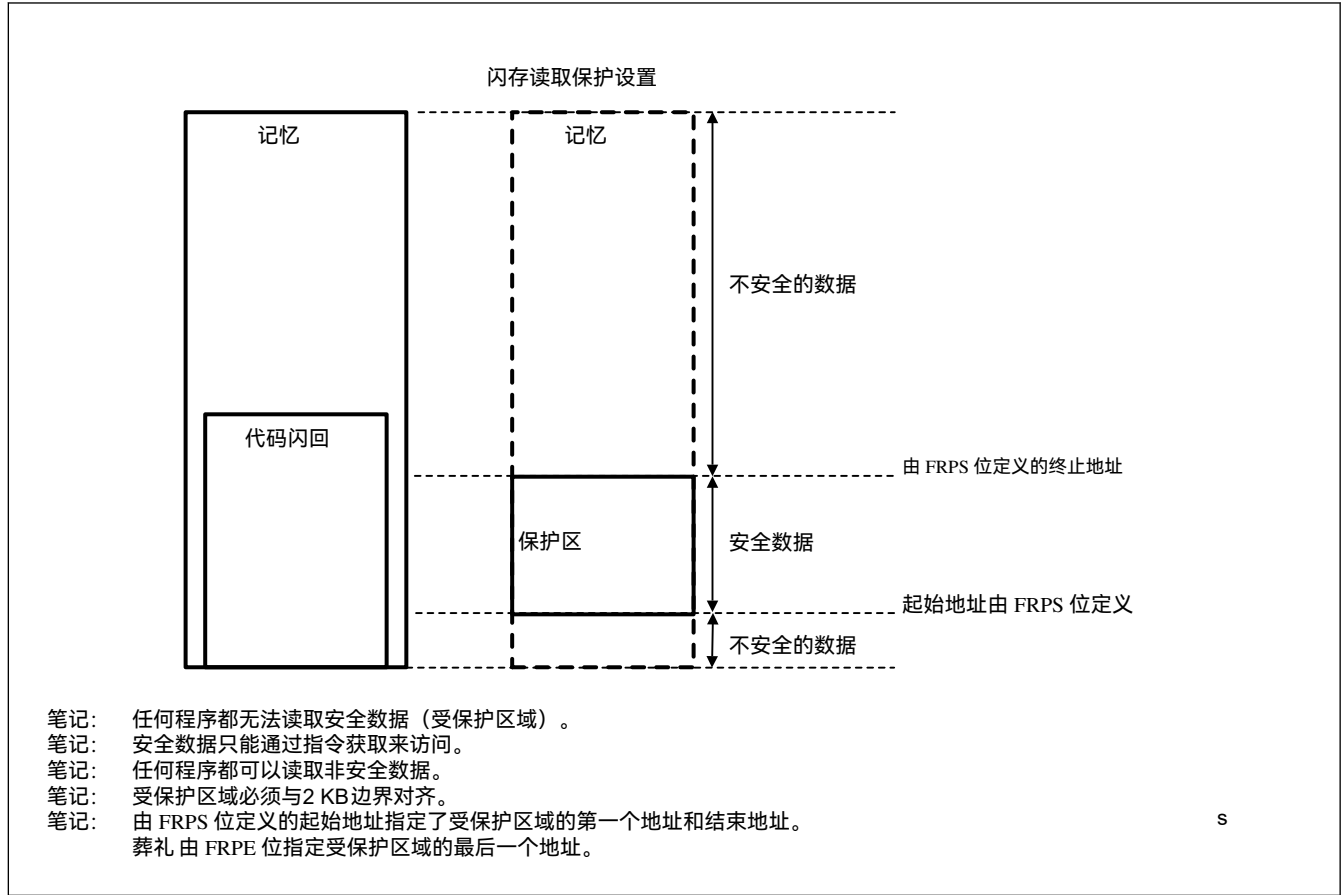


图 13.2 闪存读取保护的应用案例

13.2 使用说明

13.2.1 关于调试器使用的注意事项

如果启用了闪存读取保护 (FRP)，则无法调试存储器。调试程序时，请禁用闪存读取保护；仅当 OFS1.FRPDIS 位为 1 时，OCD 调试才有效。

13.2.2 编译器设置

FRP 是一种只执行内存 (XOM)。由于受保护区域中的数据不可读，因此受保护区域不能包含常量数据，例如字面量池。因此，需要进行相应的编译器设置。

13.2.3 OFS1 注册表的保护

由于覆盖 OFS1.FRPS[5:0]、OFS1.FRPE[5:0]或 OFS1.FRPDIS 位会禁用 FRP 保护，因此 OFS1 寄存器（地址= 0x000 0_0404）必须通过访问窗口功能进行保护。但是，设置访问窗口功能（包括将 AWS.FSPR 位设置为 0）也会禁用用于启动区域选择功能的 AWS.BTFLG 和 FISR.SAS[1:0] 位的更改。因此，使用 FRP 功能时，启动区域选择功能不可用。有关启动区域选择功能和访问窗口功能的更多信息，请参阅相关文档。

14. Data Transfer Controller (DTC)

14.1 Overview

A Data Transfer Controller (DTC) module is provided for transferring data when activated by an interrupt request.

Table 14.1 lists the DTC specifications and Figure 14.1 shows DTC block diagram.

Table 14.1 DTC specifications

Parameter	Description
Transfer modes	- Normal transfer mode A single activation leads to a single data transfer. - Repeat transfer mode A single activation leads to a single data transfer. The transfer address returns to the start address after the number of data transfers reaches the specified repeat size. The maximum number of repeat transfers is 256 and the maximum data transfer size is 256 × 32 bits (1024 bytes) - Block transfer mode A single activation leads to a transfer of a single block. The maximum block size is 256 × 32 bits = 1024 bytes.
Transfer channel	- Channel transfer can be associated with the interrupt source (transferred by a DTC activation request from the ICU) - Multiple data units can be transferred on a single activation source (chain transfer) - Chain transfers are selectable to either execute when the counter is 0, or always execute.
Transfer space	4 GB area from 0x0000_0000 to 0xFFFF_FFFF, excluding reserved areas
Data transfer units	- Single data unit: 1 byte (8 bits), 1 halfword (16 bits), 1 word (32 bits) - Single block size: 1 to 256 data units.
CPU interrupt source	- An interrupt request can be generated to the CPU on a DTC activation interrupt - An interrupt request can be generated to the CPU after a single data transfer - An interrupt request can be generated to the CPU after a data transfer of a specified volume.
Read skip	Read of transfer information can be skipped
Write-back skip	When the transfer source or destination address is specified as fixed, a write-back of transfer information can be skipped
Module-stop function	Module-stop state can be set to reduce power consumption

14.数据传输控制器 (DTC)

14.1 概述

提供数据传输控制器 (DTC) 模块，用于在中断请求激活时传输数据。

表 14.1 列出了 DTC 规格，图 14.1 显示了 DTC 框图。

表 14.1DTC 规格

范围	描述
传输模式	●普通传输模式：单次激活导致单次数据传输。 ●重复传输模式：单次激活只会导致一次数据传输。 当数据传输次数达到指定的重复大小后，传输地址将返回到起始地址。 重复传输的最大次数为 256，最大数据传输大小为 32 位（1024 字节）。 ●块传输模式 单次激活会导致单个数据块的传输。最大数据块大小为256 × 32位= 1024 字节。
传输通道	●通道转移可以与中断源关联（由 ICU 发出的 DTC 激活请求转移） ●多个数据单元可以通过单个激活源进行传输（链式传输） ●链式转账可以选择在计数器为 0 时执行，或者始终执行。
转移空间	●从 0x0000_0000 到 0xFFFF_FFFF 的 4 GB 区域，不包括保留区域
数据传输单元	●单个数据单元：1 字节（8 位）、1 半字（16 位）、1 字（32 位） ●单个数据块大小：1 至 256 个数据单元。
CPU中断源	●在DTC激活中断时，可以向CPU生成中断请求 ● 单次数据传输后，即可向 CPU 生成中断请求。 ●在传输指定量的数据传输后，可以向 CPU 生成中断请求。
阅读跳过	可以跳过读取转账信息这一步。
回写跳过	当传输源地址或目标地址被指定为固定地址时，可以跳过传输信息的写回操作。
模块停止功能	可以设置模块停止状态以降低功耗

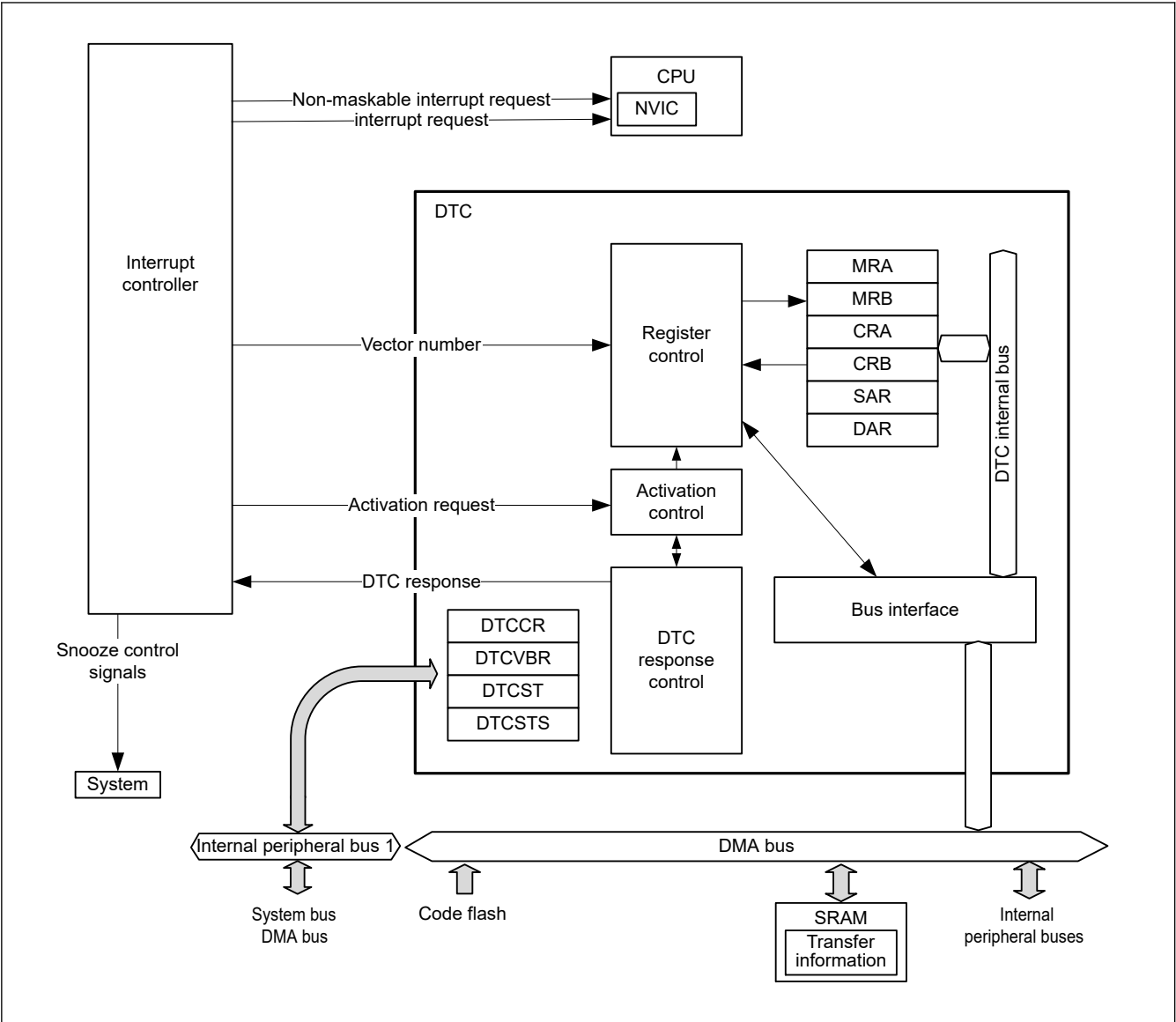


Figure 14.1 DTC block diagram

See [section 11, Interrupt Controller Unit \(ICU\)](#) for the connections between the DTC and NVIC in the CPU.

14.2 Register Descriptions

MRA, MRB, SAR, DAR, CRA, and CRB are all DTC internal registers that cannot be directly accessed from the CPU. Values to be set in these DTC internal registers are placed in the SRAM area as transfer information. When an activation request is generated, the DTC reads the transfer information from the SRAM area and sets it in its internal registers. After the data transfer ends, the internal register contents are written back to the SRAM area as transfer information.

14.2.1 MRA : DTC Mode Register A

Base address:	DTCVBR							
Offset address:	0x0003 + 0x4 × Vector number (Inaccessible directly from the CPU. See section 14.3.1. Allocating Transfer Information and DTC Vector Table)							
Bit position:	7	6	5	4	3	2	1	0
Bit field:	MD[1:0]	SZ[1:0]	SM[1:0]	—	—	—	—	—
Value after reset:	x	x	x	x	x	x	x	x

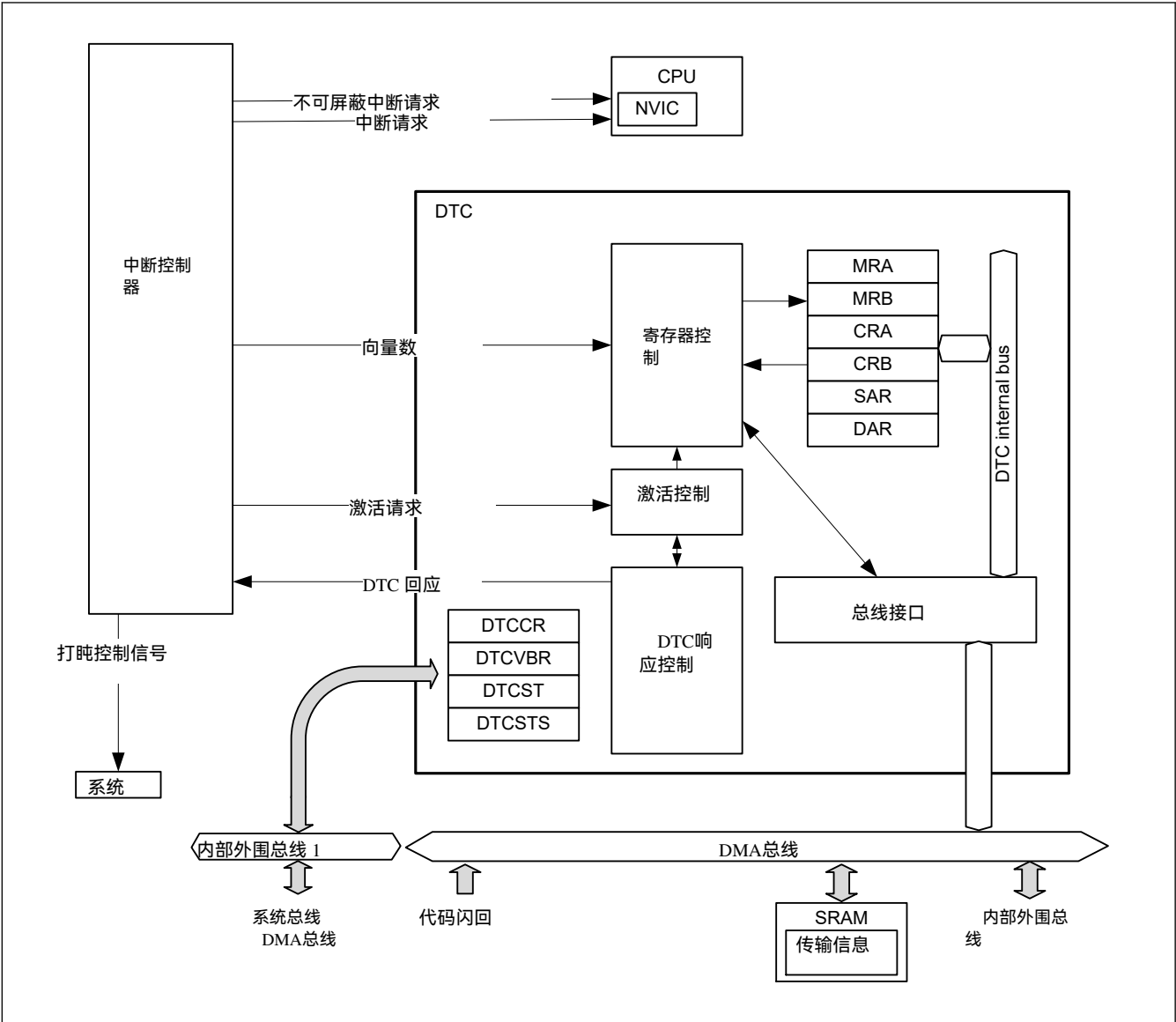


图 14.1DTC 框图

有关 CPU 中 DTC 和 NVIC 之间的连接，请参阅第 11 节“中断控制器单元 (ICU)”。

14.2 寄存器描述

MRA、MRB、SAR、DAR、CRA 和 CRB 均为 DTC 内部寄存器，CPU 无法直接访问。这些 DTC 内部寄存器中的待设置值会作为传输信息存储在 SRAM 区域。当生成激活请求时，DTC 从 SRAM 区域读取传输信息并将其设置到其内部寄存器中。数据传输结束后，内部寄存器的内容会作为传输信息写回 SRAM 区域。

14.2.1MRA：DTC模式寄存器A

基本地址:	DTCVBR							
偏移地址:	0x0003 + 0x4 × V扇形编号 (访问量可能的 直接来自 CP U. 参见第 1 节 4.3.1. 允许猫咪 转移信息和DTC向量表)							
位位置:	7	6	5	4	3	2	1	0
位域:	MD[1:0]	SZ[1:0]	SM[1:0]	—	—	—	—	—
重置后的值:	x	x	x	x	x	x	x	x

Bit	Symbol	Function	R/W
1:0	—	The read values are undefined. The write value should be 0.	—
3:2	SM[1:0]	Transfer Source Address Addressing Mode 0 0: Address in the SAR register is fixed (write-back to SAR is skipped.) 0 1: Address in the SAR register is fixed (write-back to SAR is skipped.) 1 0: SAR value is incremented after data transfer: +1 when SZ[1:0] = 00b +2 when SZ[1:0] = 01b +4 when SZ[1:0] = 10b 1 1: SAR value is decremented after data transfer: -1 when SZ[1:0] = 00b -2 when SZ[1:0] = 01b -4 when SZ[1:0] = 10b	—
5:4	SZ[1:0]	DTC Data Transfer Size 0 0: Byte (8-bit) transfer 0 1: Halfword (16-bit) transfer 1 0: Word (32-bit) transfer 1 1: Setting prohibited	—
7:6	MD[1:0]	DTC Transfer Mode Select 0 0: Normal transfer mode 0 1: Repeat transfer mode 1 0: Block transfer mode 1 1: Setting prohibited	—

The MRA register cannot be accessed directly from the CPU. However, the CPU can access the SRAM area (transfer information (n) start address + 0x03) and DTC transfers it automatically to and from the MRA register. See [section 14.3.1. Allocating Transfer Information and DTC Vector Table](#).

14.2.2 MRB : DTC Mode Register B

Base address:	DTCVBR						
Offset address:	0x0002 + 0x4 × Vector number (Inaccessible directly from the CPU. See section 14.3.1. Allocating Transfer Information and DTC Vector Table)						
Bit position:	7	6	5	4	3	2	1 0
Bit field:	CHNE	CHNS	DISEL	DTS	DM[1:0]	—	—
Value after reset:	x	x	x	x	x	x	x

Bit	Symbol	Function	R/W
1:0	—	The read values are undefined. The write value should be 0.	—
3:2	DM[1:0]	Transfer Destination Address Addressing Mode 0 0: Address in the DAR register is fixed (write-back to DAR is skipped) 0 1: Address in the DAR register is fixed (write-back to DAR is skipped) 1 0: DAR value is incremented after data transfer: +1 when MRA.SZ[1:0] = 00b +2 when SZ[1:0] = 01b +4 when SZ[1:0] = 10b 1 1: DAR value is decremented after data transfer: -1 when MRA.SZ[1:0] = 00b -2 when SZ[1:0] = 01b -4 when SZ[1:0] = 10b	—
4	DTS	DTC Transfer Mode Select 0: Select transfer destination as repeat or block area. 1: Select transfer source as repeat or block area.	—
5	DISEL	DTC Interrupt Select 0: Generate an interrupt request to the CPU when specified data transfer is complete. 1: Generate an interrupt request to the CPU each time DTC data transfer is performed.	—

少量	象征	功能	R/W
1:0	—	读取的值未定义。写入的值应为0。	—
3:2	SM[1:0]	传输源地址寻址模式0 0: SAR 寄存器中的地址固定（跳过写回 SAR）。0 1: SAR 寄存器中的地址固定（跳过写回 SAR）。1 0: 数据传输后 SAR 值递增: +1当SZ[1:0] = 00b+2当SZ[1:0] = 01b+4当SZ[1:0] = 10b1 1: 数据传输后 SA R 值递减: -1 当SZ[1:0] = 00b-2当SZ[1:0] = 01b-4当SZ[1:0] = 10b	—
5:4	SZ[1:0]	DTC数据传输大小 0 0: 字节（8位）传输 0 1: 半字（16 位）传输 1 0: 字（32 位）传输 1.1: 禁止设置	—
7:6	MD[1:0]	DTC传输模式选择 0 0: 正常传输模式 0 1: 重复传输模式 1 0:块传输模式 1.1: 禁止设置	—

CPU 无法直接访问 MRA 寄存器。但是，CPU 可以访问 SRAM 区域（传输信息 (n) 起始地址+ 0x03)），DTC 会自动将其传输到 MRA 寄存器或从 MRA 寄存器传输信息。参见第 14.3.1 节。
[分配转移信息和DTC向量表](#)。

14.2.2 MRB: DTC 模式寄存器 B

基地址:	DTCVBR						
偏移地址:	0x0002 + 0x4 × 向量编号 (无法直接从 CPU 访问。参见 14.3.1 节“分配传输信息和 DTC 向量表”)						
位位置:	7	6	5	4	3	2	1 0
位域:	CHNE	CHNS	DISEL	DTS	DM[1:0]	—	—
重置后的值:	x	x	x	x	x	x	x

少量	象征	功能	R/W
1:0	—	读取的值未定义。写入的值应为0。	—
3:2	DM[1:0]	传输目标地址寻址模式 0 0: DAR寄存器中的地址固定（跳过写回DAR）0 1: DAR寄存器中的地址固定（跳过写回DAR）1 0: 数据传输后DAR值递增: +1 当MRA.SZ[1:0] = 00b+2当SZ[1:0] = 01b+4当SZ[1:0] = 10b1 1: 数据传输后DAR值递减: -1 当 MRA.SZ[1:0] = 00b-2当SZ[1:0] = 0 1b-4当SZ[1:0] = 10b	—
4	DTS	DTC传输模式选择 0: 选择转账目的地为重复区域或封锁区域。 1: 选择传输源为重复区域或块区域。	—
5	DISEL	DTC中断选择 0: 当指定的数据传输完成时，向 CPU 生成中断请求。 1: 每次执行 DTC 数据传输时，向 CPU 生成中断请求。	—

Bit	Symbol	Function	R/W
6	CHNS	DTC Chain Transfer Select 0: Chain transfer is continuous. 1: Chain transfer occurs only when the transfer counter changes from 1 to 0 or 1 to CRAH.	—
7	CHNE	DTC Chain Transfer Enable 0: Chain transfer is disabled. 1: Chain transfer is enabled.	—

The MRB register cannot be accessed directly from the CPU. However, the CPU can access the SRAM area (transfer information (n) start address + 0x02) and DTC transfers it automatically to and from the MRB register. See [section 14.3.1. Allocating Transfer Information and DTC Vector Table](#).

DM[1:0] bits (Transfer Destination Address Addressing Mode)

The DM[1:0] bits are to fix the address of the DAR register or specify increment/decrement of the DAR register after transfer.

DTS bit (DTC Transfer Mode Select)

The DTS bit specifies whether the transfer source or destination is the repeat or block area in repeat or block transfer mode.

DISEL bit (DTC Interrupt Select)

The DISEL bit specifies the condition for generating an interrupt request to the CPU.

CHNS bit (DTC Chain Transfer Select)

The CHNS bit selects the chain transfer condition. When CHNE is 0, the CHNS setting is ignored. For details on the conditions for chain transfer, see [Table 14.3](#).

When the next transfer is chain transfer, completion of the specified number of transfers is not determined, the activation source flag is not cleared, and an interrupt request to the CPU is not generated.

CHNE bit (DTC Chain Transfer Enable)

The CHNE bit enables chain transfer. The chain transfer condition is selected by the CHNS bit. For details on chain transfer, see [section 14.4.6. Chain Transfer](#).

14.2.3 SAR : DTC Transfer Source Register

[illegible]

The SAR sets the transfer source start address and cannot be accessed directly from the CPU. However, the CPU can access the SRAM area (transfer information (n) start address + 0x04) and DTC transfers it automatically to and from the SAR register. See [section 14.3.1. Allocating Transfer Information and DTC Vector Table](#).

Misalignment is prohibited for DTC transfers. Bit[0] must be 0 when MRA.SZ[1:0] = 01b, and bit[1] and bit[0] must be 0 when MRA.SZ[1:0] = 10b.

少量	象征	功能	R/W
6	CHNS	DTC 链式传输选择 0: 链式传输是连续的。 1: 链式转移仅当转移计数器从 1 变为 0 或从 1 变为 CRAH 时发生。	—
7	CHNE	DTC 链式传输启用: 0: 链式传输已禁用。1: 链式传输已启用。	—

CPU 无法直接访问 MRB 寄存器。但是，CPU 可以访问 SRAM 区域（传输信息 (n) 起始地址+0x02），DTC 会自动将其传输到 MRB 寄存器或从 MRB 寄存器传输信息。参见第 14.3.1 节。

[分配转移信息和DTC向量表。](#)

DM[1:0] 位（传输目标地址寻址模式）

DM[1:0] 位用于固定 DAR 寄存器的地址，或指定传输后 DAR 寄存器的增量/减量。

DTS 位 (DTC 传输模式选择)

DTS 位指定传输源或目标是重复传输模式还是块传输模式下的重复区域或块区域。

DISEL 位 (DTC 中断选择)

DISEL 位指定向 CPU 生成中断请求的条件。

CHNS 位 (DTC 链传输选择)

CHNS 位用于选择链转移条件。当 CHNE 为 0 时, CHNS 设置将被忽略。有关链转移条件的详细信息, 请参见表 14.3。

当下一次传输是链式传输时，未确定是否完成了指定次数的传输，未清除激活源标志，也未向 CPU 生成中断请求。

CHNE 位 (DTC 链传输使能)

CHNE 位启用链式传输。链式传输条件由 CHNS 位选择。有关链式传输的详细信息，请参见第 14.4.6 节“链式传输”。

14.2.3 SAR: DTC 传输源寄存器

基地址: DTCVBR

偏移地址: $0x0004 + 0x4 \times \text{向量编号}$
 (无法直接从 CPU 访问。参见 14.3.1 节“分配传输信息和 DTC 向量表”)

位位置: 31

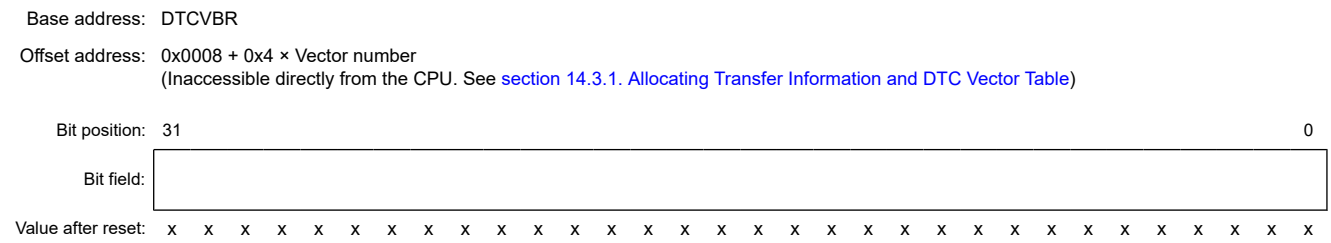
位域:

重置后的值: XXX XX XX XX XX XX XX XX X XX XX XX

SAR 设置传输源起始地址，CPU 无法直接访问它。但是，CPU 可以访问 SRAM 区域（传输信息 (n) 起始地址+ 0x04），DTC 会自动将其传输到 SAR 寄存器或从 SAR 寄存器传输。参见第 14.3.1 节“分配传输信息和 DTC 向量表”。

DTC 传输禁止错位。当 MRA.SZ[1:0] = 01b 时，位 [0] 必须为 0；当 MRA.SZ[1:0] = 10b 时，位 [1] 和位 [0] 必须为 0。

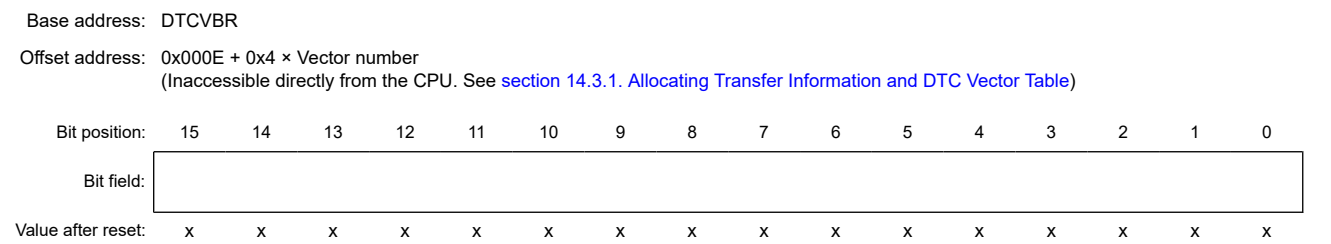
14.2.4 DAR : DTC Transfer Destination Register



The DAR sets the transfer destination start address and cannot be accessed directly from the CPU. However, the CPU can access the SRAM area (transfer information (n) start address + 0x08) and DTC transfers it automatically to and from the DAR register. See [section 14.3.1. Allocating Transfer Information and DTC Vector Table](#).

Misalignment is prohibited for DTC transfers. Bit[0] must be 0 when MRA.SZ[1:0] = 01b, and bit[1] and bit[0] must be 0 when MRA.SZ[1:0] = 10b.

14.2.5 CRA : DTC Transfer Count Register A



Bit	Symbol	Function	R/W
7:0	CRAL	Transfer Counter A Lower Register Specify the transfer count.	—
15:8	CRAH	Transfer Counter A Upper Register Specify the transfer count.	—

Note: The function depends on the transfer mode.
Note: Set CRAH and CRAL to the same value in repeat transfer mode and block transfer mode.

The CRA register consists of 16 bits. CRAL is the lower 8 bits and CRAH is the upper 8 bits. CRA is used in normal mode. CRAL and CRAH are used in repeat transfer mode and block transfer mode.

The CRA register cannot be accessed directly from the CPU. However, the CPU can access the SRAM area (transfer information (n) start address + 0x0E) and DTC transfers it automatically to and from the CRA register. See [section 14.3.1. Allocating Transfer Information and DTC Vector Table](#).

(1) Normal transfer mode (MRA.MD[1:0] = 00b)

In normal transfer mode, CRA functions as a 16-bit transfer counter. The transfer count is 1, 65535, and 65536 when the set value is 0x0001, 0xFFFF, and 0x0000, respectively. The CRA value is decremented (-1) on each data transfer.

(2) Repeat transfer mode (MRA.MD[1:0] = 01b)

In repeat transfer mode, the CRAH register holds the transfer count and the CRAL register functions as an 8-bit transfer counter. The transfer count is 1, 255, and 256 when the set value is 0x01, 0xFF, and 0x00, respectively. The CRAL value is decremented (-1) on each data transfer. When it reaches 0x00, the CRAH value is transferred to CRAL.

(3) Block transfer mode (MRA.MD[1:0] = 10b)

In block transfer mode, the CRAH register holds the block size and the CRAL register functions as an 8-bit block size counter. The transfer count is 1, 255, and 256 when the set value is 0x01, 0xFF, and 0x00, respectively. The CRAL value is decremented (-1) on each data transfer. When it reaches 0x00, the CRAH value is transferred to CRAL.

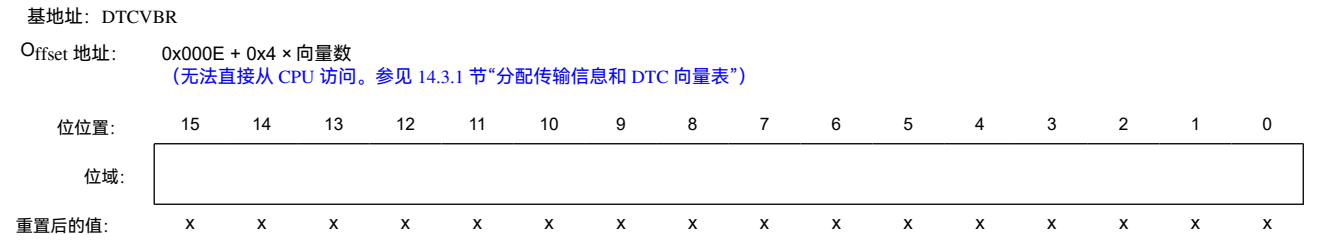
14.2.4 DAR: DTC 转账目的地寄存器



DAR 设置传输目标起始地址，CPU 无法直接访问。但是，CPU 可以访问 SRAM 区域（传输信息 (n) 起始地址+ 0x08）），DTC 会自动将其传输到 DAR 寄存器或从 DAR 寄存器传输。参见 [14.3.1 节“分配传输信息和 DTC 向量表”](#)。

DTC 传输禁止错位。当 MRA.SZ[1:0] = 01b时，位 [0] 必须为 0；当 MRA.SZ[1:0] = 10b时，位 [1] 和位 [0] 必须为 0。

14.2.5CRA： DTC 转账计数寄存器 A



少量	象征	功能	R/W
7:0	CRAL	传输计数器 A 低寄存器指定传输计数。	—
15:8	CRAH	传输计数器 A 高位寄存器指定传输计数。	—

注: 此功能取决于传输模式。
Note: Set CRAH and CRAL to the same value in repeat transfer mode and block transfer mode.

CRA寄存器由16位组成。CRAL是低8位，CRAH是高8位。CRA在正常模式下使用。CRAL 和 CRAH 用于重复传输模式和块传输模式。CPU 无法直接访问 CRA 寄存器。但是，CPU 可以访问 SRAM 区域（传输信息 (n) 起始地址+ 0x0E）），DTC 会自动将其传输到 CRA 寄存器或从 CRA 寄存器传输信息。参见第 [14.3.1 节。分配转移信息和DTC向量表。](#)

(1) 正常传输模式(MRA.MD[1:0] = 00b)

在正常传输模式下，CRA 用作 16 位传输计数器。当其值分别为0x0001、0xFFFF 和0x0000, 时，传输计数分别为 1、65 535 和 65536。每次数据传输后，CRA 值都会减 1。

(2) 重复传输模式(MRA.MD[1:0] = 01b)

在重复传输模式下，CRAH 寄存器保存传输计数，CRAL 寄存器用作 8 位传输计数器。当 CRAL 寄存器的设置值分别为0x01、0xFF 和 0x00 时，传输计数分别为 1、255 和 256。每次数据传输时，CRAL 寄存器的值都会减 1。当 CRAL 寄存器的值达到0x00时，CRAH 寄存器的值会被传递给 CRAL 寄存器。

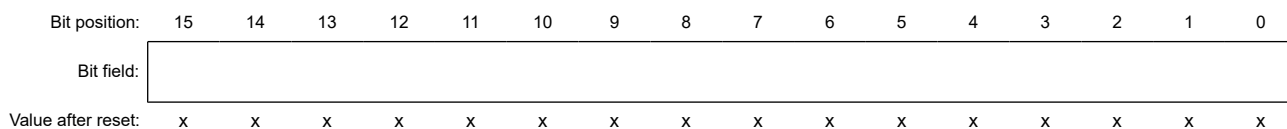
(3) 块传输模式(MRA.MD[1:0] = 10b)

在块传输模式下，CRAH 寄存器保存块大小，CRAL 寄存器用作 8 位块大小计数器。当 CRAH 寄存器的值分别为0x01、0xFF, 和0x00时，传输计数分别为 1、255 和 256。每次数据传输时，CRAL 寄存器的值都会减 1。当 CRAL 寄存器的值达到0x00, 时，CRAH 寄存器的值会被传递给 CRAL 寄存器。

14.2.6 CRB : DTC Transfer Count Register B

Base address: DTCVBR

Offset address: $0x000C + 0x4 \times \text{Vector number}$
(Inaccessible directly from the CPU. See [section 14.3.1. Allocating Transfer Information and DTC Vector Table](#))



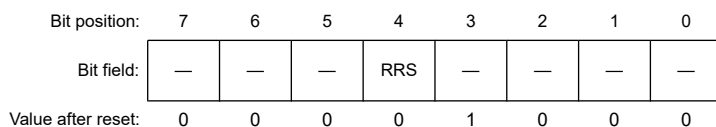
The CRB sets the block transfer count for block transfer mode. The transfer count is 1, 65535, and 65536 when the set value is 0x0001, 0xFFFF, and 0x0000, respectively. The CRB value is decremented (-1) when the final data of a single block size is transferred. When normal transfer mode or repeat transfer mode is selected, this register is not used, and the set value is ignored.

The CRB cannot be accessed directly from the CPU. However, the CPU can access the SRAM area (transfer information (n) start address + 0x0C) and DTC transfers it automatically to and from the CRB register. See [section 14.3.1. Allocating Transfer Information and DTC Vector Table](#).

14.2.7 DTCCR : DTC Control Register

Base address: DTC = 0x4000_5400

Offset address: 0x0000



Bit	Symbol	Function	R/W
2:0	—	These bits are read as 0. The write value should be 0.	R/W
3	—	This bit is read as 1. The write value should be 1.	R/W
4	RRS	DTC Transfer Information Read Skip Enable 0: Transfer information read is not skipped 1: Transfer information read is skipped when vector numbers match	R/W
7:5	—	These bits are read as 0. The write value should be 0.	R/W

RRS bit (DTC Transfer Information Read Skip Enable)

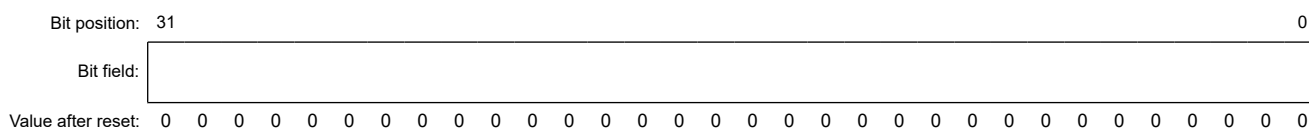
The RRS bit enables skipping of transfer information reads when vector numbers match. The DTC vector number is compared with the vector number in the previous activation process. When these vector numbers match and the RRS bit is set to 1, DTC data transfer is performed without reading the transfer information. However, when the previous transfer is a chain transfer, the transfer information is read regardless of the RRS bit.

When the transfer counter (CRA register) becomes 0 during the previous normal transfer and when the transfer counter (CRB register) becomes 0 during the previous block transfer, the transfer information is read regardless of the RRS bit value.

14.2.8 DTCVBR : DTC Vector Base Register

Base address: DTC = 0x4000 5400

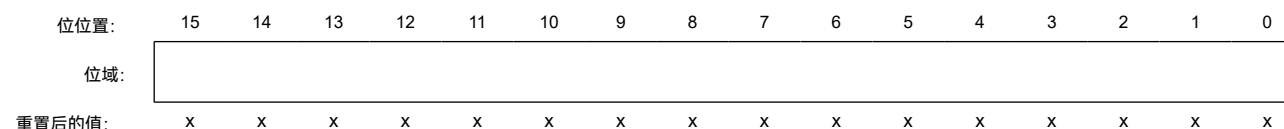
Offset address: 0x0004



14.2.6 CRB: DTC 传输计数寄存器 B

基地址: DTCVBR

偏移地址: $0x000C + 0x4 \times \text{向量编号}$
(无法直接从 CPU 访问。参见 14.3.1 节“分配传输信息和 DTC 向量表”)



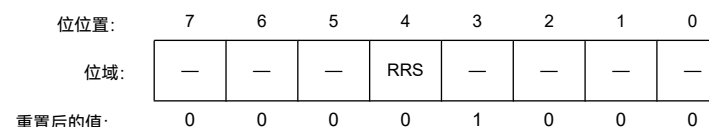
CRB 寄存器用于设置块传输模式下的块传输计数。当其值分别为0x0001、0xFFFF 和0x0000, 时, 传输计数分别为 1、65535 和 65536。当单个块大小的最后一个数据被传输完毕后, CRB 寄存器的值会减 1。当选择普通传输模式或重复传输模式时, 该寄存器将不被使用, 其设置值也会被忽略。

CPU 无法直接访问 CRB。但是，CPU 可以访问 SRAM 区域（传输信息 (n) 起始地址+ 0x0C），DTC 会自动将其传输到 CRB 寄存器或从 CRB 寄存器传输信息。参见第 14.3.1 节“分配传输信息和 DTC 向量表”。

14.2.7DTCCR: DTC 控制寄存器

基本地址: DTC = 0x4000_5400

偏移地址: 0x0000



少量	象征	功能	R/W
2:0	—	这些位读取为 0。写入值应为 0。	R/W
3	—	该位读取值为 1。写入值应为 1。	R/W
4	RRS	DTC 传输信息读取跳过启用 0: 传输信息读取不会被跳过 1: 当向量编号匹配时, 跳过读取传输信息	R/W
7:5	—	这些位读取为 0。写入值应为 0。	R/W

RRS 位 (DTC 传输信息读取跳过使能)

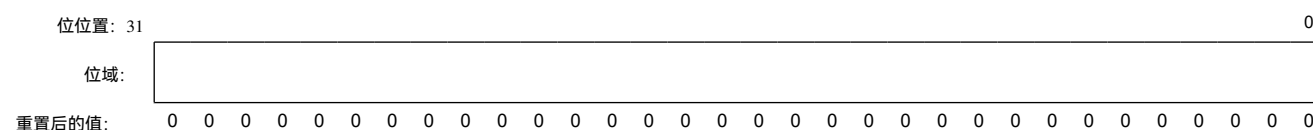
当向量编号匹配时，RRS 位允许跳过传输信息读取。DTC 向量编号会与上一次激活过程中的向量编号进行比较。如果这两个向量编号匹配且 RRS 位设置为 1，则执行 DTC 数据传输，而无需读取传输信息。但是，如果上一次传输是链式传输，则无论 RRS 位如何，都会读取传输信息。

当上一次正常传输期间传输计数器（CRA 寄存器）变为 0，且上一次块传输期间传输计数器（CRB 寄存器）变为 0 时，无论 RRS 位值如何，都会读取传输信息。

14.2.8DTCVBR: DTC矢量基址寄存器

Base address: DTC = 0x4000 5400

Offset address: 0x0004



Bit	Symbol	Function	R/W
31:0	n/a	DTC Vector Base Address Set the DTC vector base address. The lower 10 bits should be 0.	R/W

The DTCVBR sets the base address for calculating the DTC vector table address, which can be set in the range of 0x0000_0000 to 0xFFFF_FFFF (4 GB) in 1-KB units.

14.2.9 DTCST : DTC Module Start Register

Base address: DTC = 0x4000_5400															
Offset address: 0x000C															
Bit position: 7 6 5 4 3 2 1 0															
Bit field: <table><tr><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>DTCST</td></tr></table>								—	—	—	—	—	—	—	DTCST
—	—	—	—	—	—	—	DTCST								
Value after reset: 0 0 0 0 0 0 0 0															

Bit	Symbol	Function	R/W
0	DTCST	DTC Module Start 0: DTC module stopped. 1: DTC module started.	R/W
7:1	—	These bits are read as 0. The write value should be 0.	R/W

DTCST bit (DTC Module Start)

Set the DTCST bit to 1 to enable the DTC to accept transfer requests. When this bit is set to 0, transfer requests are no longer accepted. If this bit is set to 0 during a data transfer, the accepted transfer request is active until processing completes.

DTCST must be set to 0 before transitioning to one of the following state or mode:

- Module-stop state
- Software Standby mode without Snooze mode transition

For details on these transitions, see [section 14.8. Low Power Consumption Function](#) and [section 9, Low Power Modes](#).

14.2.10 DTCSTS : DTC Status Register

Base address: DTC = 0x4000_5400																															
Offset address: 0x000E																															
Bit position: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																															
Bit field: <table><tr><td>ACT</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td colspan="8">VECN[7:0]</td></tr></table>																ACT	—	—	—	—	—	—	—	VECN[7:0]							
ACT	—	—	—	—	—	—	—	VECN[7:0]																							
Value after reset: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0																															

Bit	Symbol	Function	R/W
7:0	VECN[7:0]	DTC-Activating Vector Number Monitoring These bits indicate the vector number for the activation source when a DTC transfer is in progress. The value is only valid if a DTC transfer is in progress (ACT flag is 1).	R
14:8	—	These bits are read as 0.	R
15	ACT	DTC Active Flag 0: DTC transfer operation is not in progress. 1: DTC transfer operation is in progress.	R

少量	象征	功能	R/W
31:0	n/a	DTC矢量基地址 设置DTC向量基地址。低10位应为0。	R/W

DTCVBR 设置用于计算 DTC 向量表地址的基地址，该地址可以设置在0x0000_0000 到 0xFFFF_FFFF (4 GB) 的范围内，以 1-KB 为单位。

14.2.9DTCST：DTC模块启动寄存器

基地址：DTC = 0x4000_5400															
偏移地址：0x000C															
位位置： 7 6 5 4 3 2 1 0															
位域： <table><tr><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>DTCST</td></tr></table>								—	—	—	—	—	—	—	DTCST
—	—	—	—	—	—	—	DTCST								
重置后的值： 0 0 0 0 0 0 0 0															

少量	象征	功能	R/W
0	DTCST	DTC模块启动状态：0：DTC模块停止。1：DTC模块启动。	R/W
7:1	—	这些位读取为 0。写入值应为 0。	R/W

DTCST 位（DTC 模块启动）

将 DTCST 位设置为 1 可使 DTC 接受传输请求。当此位设置为 0 时，将不再接受传输请求。如果在数据传输期间此位设置为 0，则已接受的传输请求将保持活动状态，直到处理完成。

在转换到以下状态或模式之一之前，必须将 DTCST 设置为 0:

- 模块停止状态
- 软件待机模式，无需进入休眠模式即可过渡

有关这些转换的详细信息，请参阅第 14.8 节“低功耗功能”和第 9 节“低功耗模式”。

14.2.10DTCSTS：DTC 状态寄存器

基地址：DTC = 0x4000_5400																
偏移地址：0x000E																
位位置：	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域：	ACT	—	—	—	—	—	—	—	VECN[7:0]							
重置后的值：	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
7:0	VECN[7:0]	DTC激活向量编号监测 这些位指示 DTC 传输进行时激活源的向量编号。 仅当 DTC 转账正在进行时（ACT 标志为 1）该值才有效。	R
14:8	—	这些位读取为 0。	R
15	ACT	DTC激活标志 0: DTC传输操作未进行中。 1: DTC传输操作正在进行中。	R

VECN[7:0] bits (DTC-Activating Vector Number Monitoring)

While transfer by the DTC is in progress, the VECN[7:0] bits indicate the vector number associated with the activation source for the transfer. The value read from the VECN[7:0] bits is valid if the ACT flag is 1, indicating a DTC transfer in progress, and invalid if the ACT flag is 0, indicating no DTC transfer is in progress.

ACT flag (DTC Active Flag)

The ACT flag indicates the state of the DTC transfer operation.

[Setting condition]

- When the DTC is activated by a transfer request.

[Clearing condition]

- When transfer by the DTC, in response to a transfer request, is complete.

14.3 Activation Sources

The DTC is activated by an interrupt request. Setting the ICU.DTCENSTx.STy (x = 0, 1, y = 0 to 41) bit to 1 enables activation of the DTC by the associated interrupt, where y indicates the event number of the interrupt request.

The interrupt vector number is equivalent to the DTC vector table number. After the DTC accepted an activation request, it does not accept another activation request until the transfer for that single request is complete, regardless of the priority of the requests. When multiple activation requests are generated during a DTC transfer, the highest priority request is accepted on completion of the transfer. When multiple activation requests are generated while the DTC Module Start bit (DTCST.DTCST) is 0, the DTC accepts the highest priority request when DTCST.DTCST is subsequently set to 1. The smaller interrupt vector number has higher priority.

The DTC performs the following operations at the start of a single data transfer or for a chain transfer, after the last of the consecutive transfers:

- On completion of a specified round of data transfer, the ICU.DTCENSTx.STy bit is set to 0, and an interrupt request is sent to the CPU.
- If the MRB.DISEL bit is 1, an interrupt request is sent to the CPU on completion of a data transfer.

14.3.1 Allocating Transfer Information and DTC Vector Table

The DTC reads the start address of the transfer information associated with each activation source from the vector table and reads the transfer information starting at that address.

The vector table must be located so that the lower 10 bits of the base address (start address) are 0. Use the DTC Vector Base Register (DTCVBR) to set the base address of the DTC vector table. Transfer information is allocated in the SRAM area. In the SRAM area, the start address of the transfer information *n* with vector number *n* must be 4*n* added to the base address in the vector table.

Figure 14.2 shows the relationship between the DTC vector table and transfer information. Figure 14.3 shows the allocation of transfer information in the SRAM area.

VECN[7:0] 位（DTC-激活矢量编号监控）

当DTC传输正在进行时，VECN[7:0] 位指示与传输激活源关联的向量编号。如果ACT标志为1，则从VECN[7:0] 位读取的值有效，表示DTC传输正在进行；如果ACT标志为0，则读取的值无效，表示DTC传输未进行。

ACT 标志（DTC 激活标志）

ACT 标志指示 DTC 传输操作的状态。

[设置条件]

- 当 DTC 因转账请求而被激活时。

[清除条件]

- 当 DTC 响应转账请求完成转账时。

14.3激活源

DTC 由中断请求激活。将 ICU.DTCENSTx.STy(x = 0, 1, y = 0设置为 41) 位为 1 可启用由关联中断激活 DTC，其中 y 表示中断请求的事件编号。

中断向量编号等同于DTC向量表编号。DTC接受激活请求后，无论请求优先级如何，在单个请求的传输完成之前，不会接受其他激活请求。如果在DTC传输过程中生成多个激活请求，则在传输完成后接受优先级最高的请求。如果在DTC模块起始位（DTCST.DTCST）为0时生成多个激活请求，则当DTCST.DTCST随后被设置为1时，DTC接受优先级最高的请求。中断向量编号越小，优先级越高。

DTC 在单次数据传输开始时或在链式传输的最后一次连续传输完成后执行以下操作：

- 完成指定轮次的数据传输后，ICU.DTCENSTx.STy 位设置为 0，并向 CPU 发送中断请求。
- 如果 MRB.DISEL 位为 1，则在数据传输完成后向 CPU 发送中断请求。

14.3.1 分配转移信息和 DTC 向量表

DTC 从向量表中读取与每个激活源关联的传输信息的起始地址，并从该地址开始读取传输信息。

向量表的放置位置必须保证基地址（起始地址）的低 10 位为 0。使用 DTC 向量基址寄存器 (DTCVBR) 设置 DTC 向量表的基址。传输信息分配在 SRAM 区域中。在 SRAM 区域中，向量号为*n*的传输信息*n*的起始地址必须加到向量表的基址上。

图 14.2 显示了 DTC 向量表与传输信息之间的关系。图 14.3 显示了传输信息在 SRAM 区域中的分配情况。

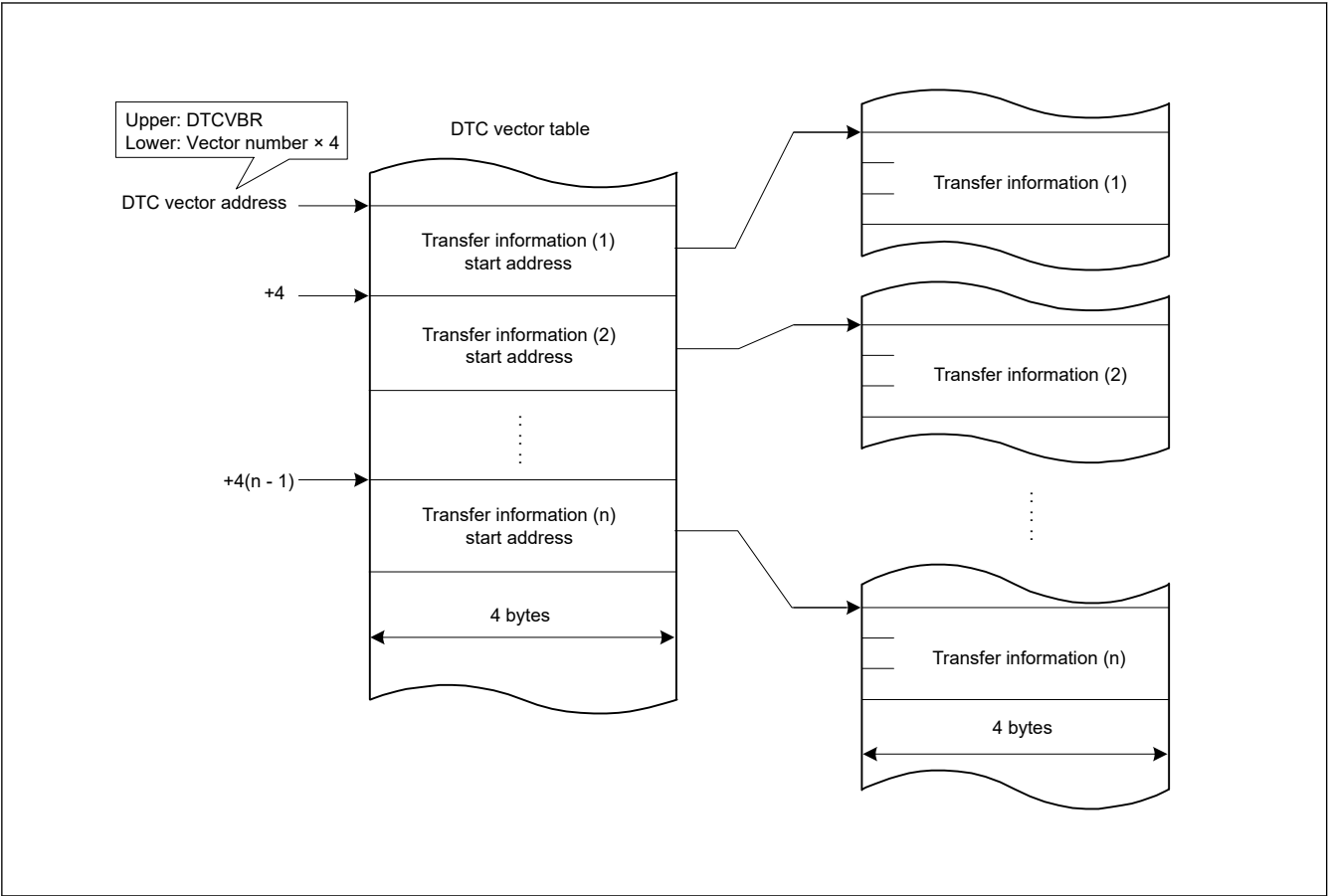


Figure 14.2 DTC vector table and transfer information

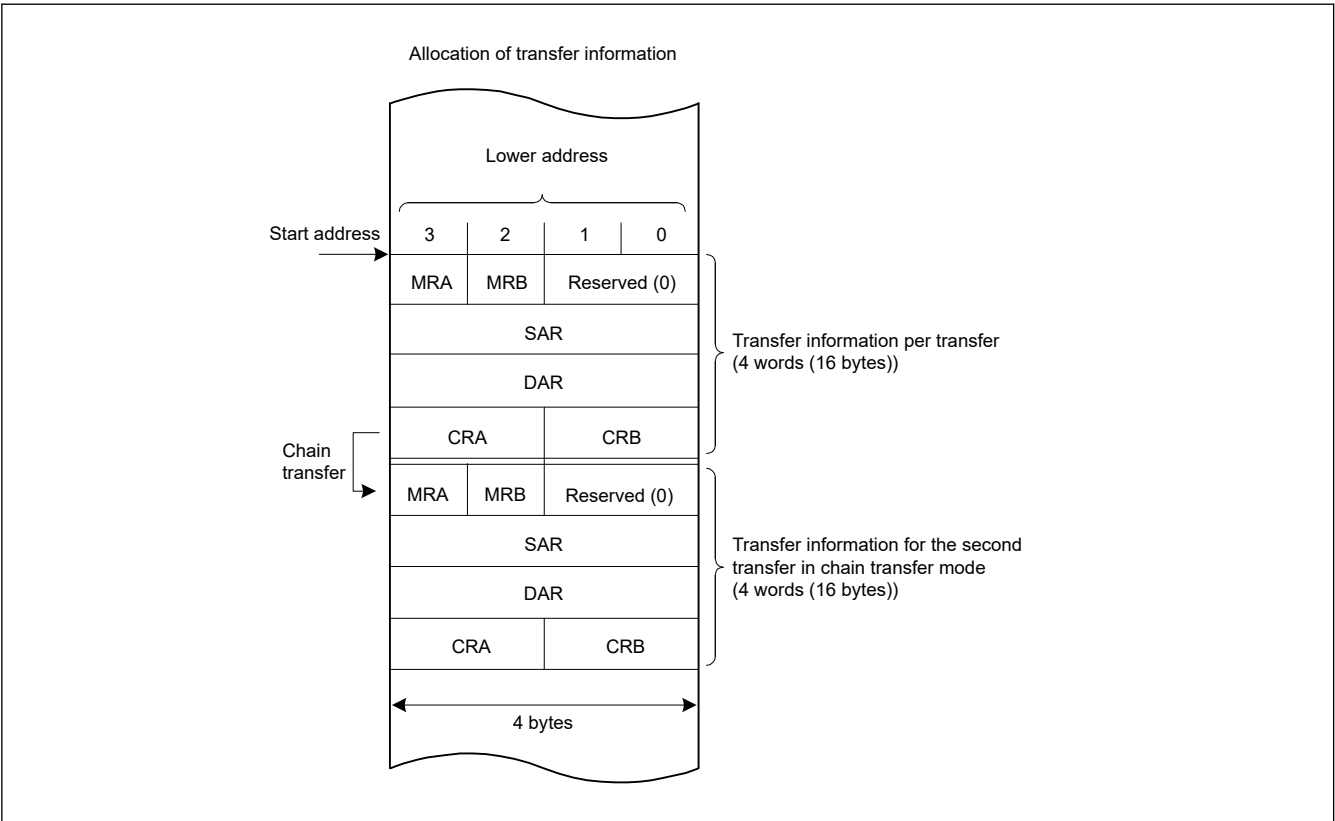


Figure 14.3 Allocation of transfer information in the SRAM area

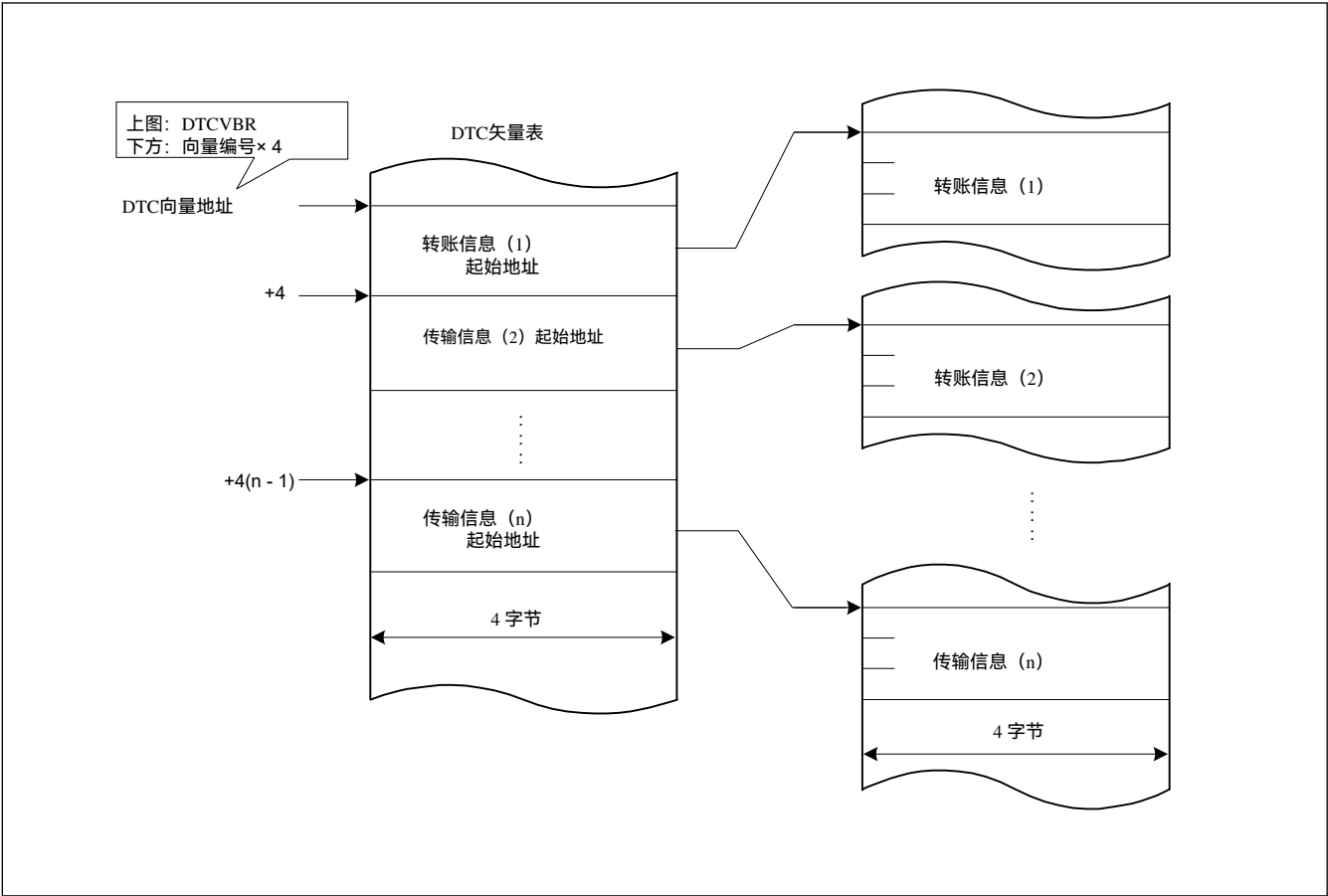


图 14.2 DTC 向量表和传输信息

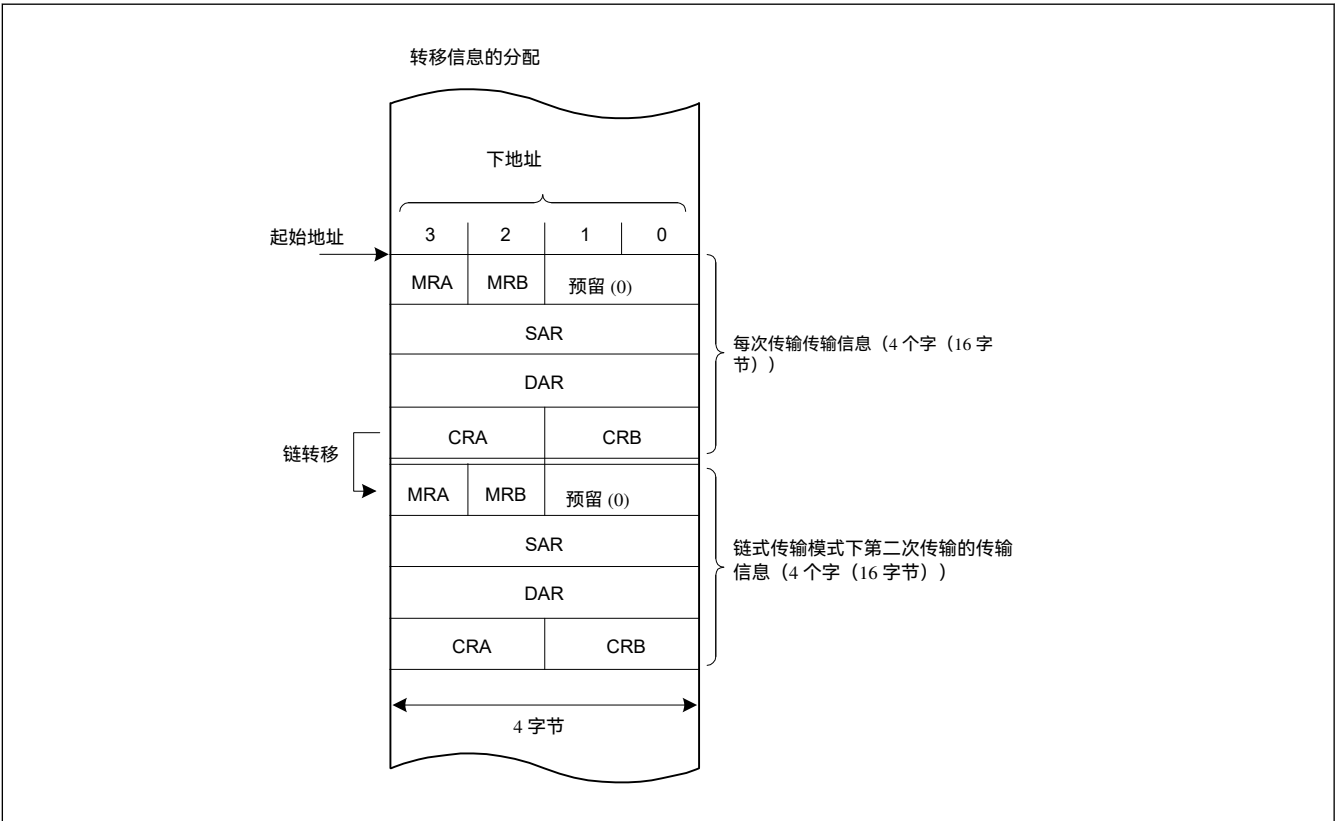


图 14.3 SRAM 区域中传输信息的分配

14.4 Operation

The DTC transfers data according to the transfer information. Storage of the transfer information in the SRAM area is required before a DTC operation. When the DTC is activated, it reads the DTC vector associated with the vector number. The DTC reads the transfer information from the transfer information store address referenced by the DTC vector and transfers the data. After the data transfer, the DTC writes back the transfer information. Storing the transfer information in the SRAM area allows data transfer of any number of channels.

The transfer modes include:

- Normal transfer mode
- Repeat transfer mode
- Block transfer mode.

The DTC specifies a transfer source address in the SAR register and a transfer destination address in the DAR register. The values of these registers are incremented, decremented, or address-fixed independently after the data transfer.

Table 14.2 lists the DTC transfer modes.

Table 14.2 DTC transfer modes

Transfer mode	Data size transferred on single transfer request	Increment or decrement of memory address	Settable transfer count
Normal transfer mode	1 byte (8 bit), 1 halfword (16 bit), 1 word (32 bit)	Incremented or decremented by 1, 2, or 4 or address-fixed	1 to 65536
Repeat transfer mode*1	1 byte (8 bit), 1 halfword (16 bit), 1 word (32 bit)	Incremented or decremented by 1, 2, or 4 or address-fixed	1 to 256*3
Block transfer mode*2	Block size specified in CRAH (1 to 256 bytes, 1 to 256 halfwords (2 to 512 bytes), or 1 to 256 words (4 to 1024 bytes))	Incremented or decremented by 1, 2, or 4 or address-fixed	1 to 65536

Note 1. Set the transfer source or transfer destination as the repeat area.

Note 2. Set the transfer source or transfer destination as the block area.

Note 3. After a data transfer of the specified count, the initial state is restored and operation restarts.

Setting the MRB.CHNE bit to 1 allows multiple transfers or chain transfer on a single activation source. It also enables a chain transfer when the specified data transfer is complete.

Figure 14.4 shows the operation flow of the DTC. Table 14.3 lists the chain transfer conditions. The combination of control information for the second and subsequent transfers are omitted in this table.

14.4 操作

DTC根据传输信息传输数据。DTC操作前需要将传输信息存储在SRAM区域中。DTC激活时，它会读取与向量编号关联的DTC向量。

DTC 从 DTC 向量指向的传输信息存储地址读取传输信息，并传输数据。数据传输完成后，DTC 将传输信息写回。将传输信息存储在 SRAM 区域中，可以实现任意数量通道的数据传输。

传输方式包括：

- 普通传输模式
- 重复传输模式
- 块传输模式。

DTC 在 SAR 寄存器中指定传输源地址，在 DAR 寄存器中指定传输目标地址。数据传输完成后，这些寄存器的值会独立地递增、递减或固定。

表 14.2 列出了 DTC 传输模式。

表 14.2DTC 传输模式

传输模式	单次传输请求中传输的数据大小	内存地址的递增或递减	可设置转移计数
正常传输模式	1 字节（8 位），1 半字（16 位），1 字（32 位）	递增或递减 1、2 或 4，或地址固定	1 to 65536
重复传输模式*1	1 字节（8 位），1 半字（16 位），1 字（32 位）	递增或递减 1、2 或 4，或地址固定	1 to 256*3
块传输模式*2	CRAH 中指定的块大小（1 到 256 字节，1 到 256 个半字（2 到 512 字节），或 1 到 256 个字（4 到 1024 字节））	递增或递减 1、2 或 4，或地址固定	1 to 65536

注1. 将传输源或传输目标设置为重复区域。

注2. 将传输源或传输目标设置为块区域。

注3. 传输指定数量的数据后，恢复初始状态并重新开始操作。

将 MRB.CHNE 位设置为 1 允许在单个激活源上进行多次传输或链式传输。它还会在指定的数据传输完成后启用链式传输。

图 14.4 显示了 DTC 的运行流程。表 14.3 列出了链式传递条件。该表中省略了第二次及后续传递的控制信息组合。

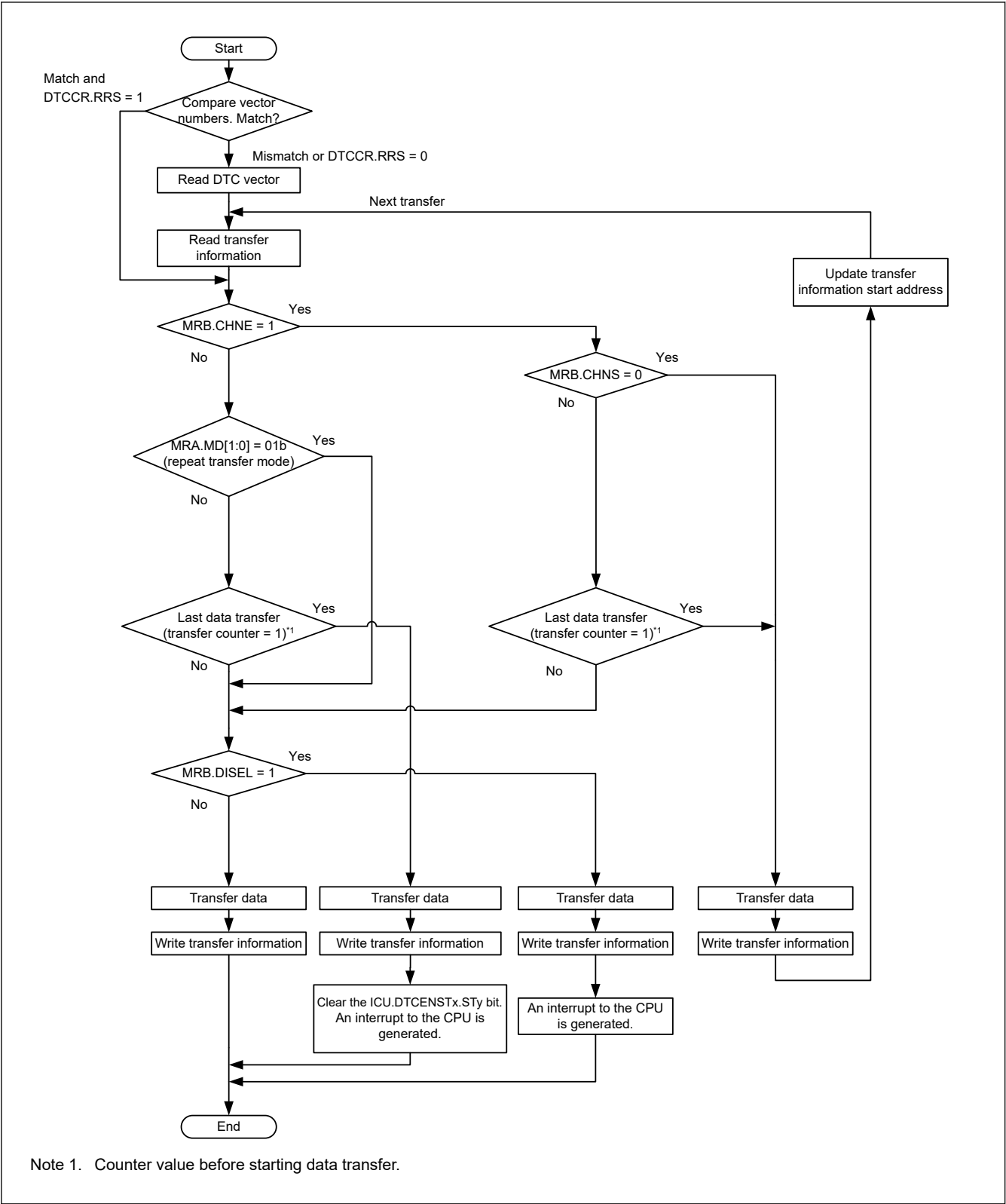


Figure 14.4 DTC operation flow

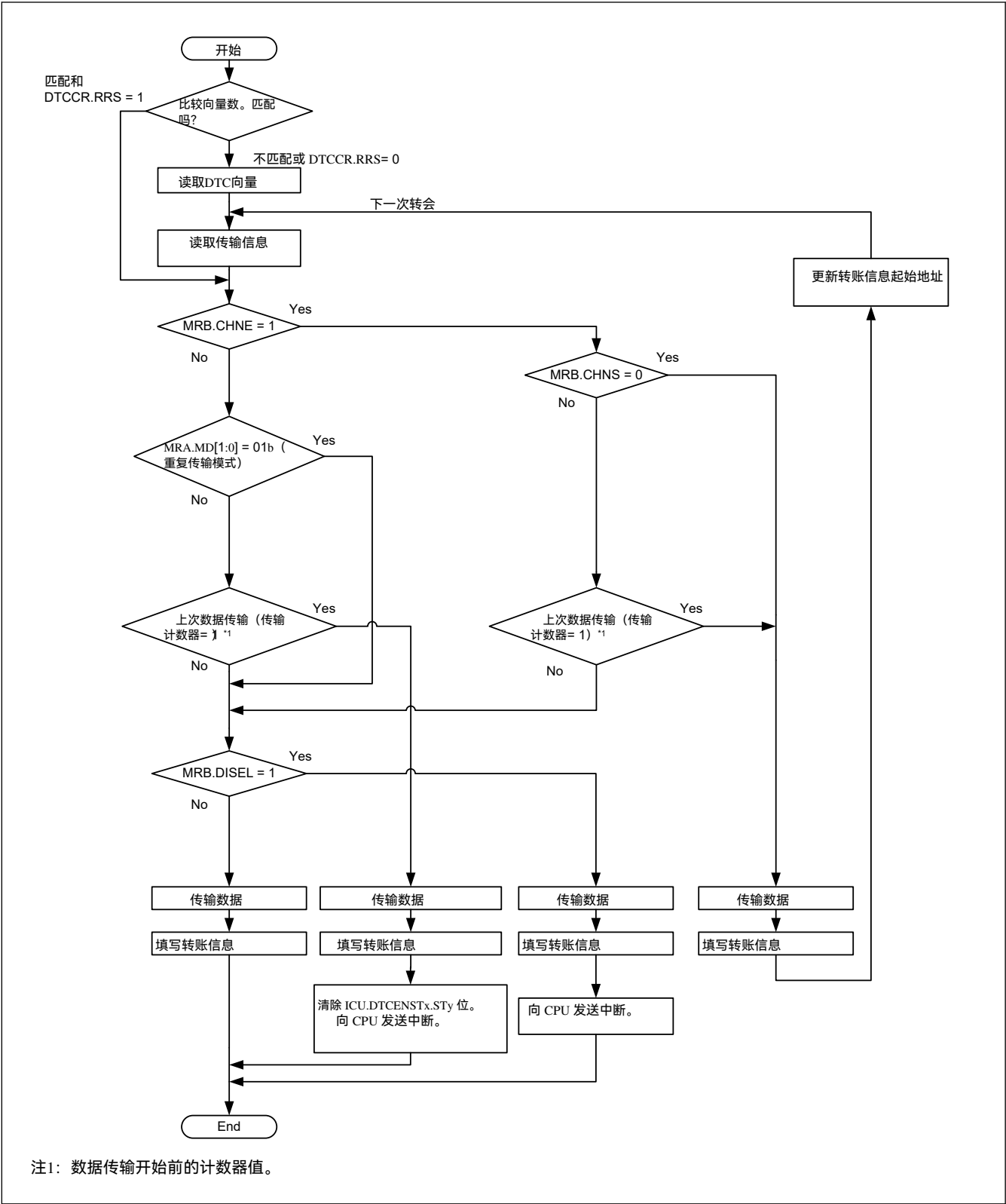


图 14.4 DTC 操作流程

Table 14.3 Chain transfer conditions

First transfer				Second transfer*3				DTC transfer
CHNE bit	CHNS bit	DISEL bit	Transfer counter*1 *2	CHNE bit	CHNS bit	DISEL bit	Transfer counter*1 *2	
0	—	0	Other than (1 → 0)	—	—	—	—	Ends after the first transfer
0	—	0	(1 → 0)	—	—	—	—	Ends after the first transfer with an interrupt request to the CPU
0	—	1	—	—	—	—	—	
1	0	—	—	0	—	0	Other than (1 → 0)	Ends after the second transfer
				0	—	0	(1 → 0)	Ends after the second transfer with an interrupt request to the CPU
				0	—	1	—	
1	1	0	Other than (1 → *)	—	—	—	—	Ends after the first transfer
1	1	—	(1 → *)	0	—	0	Other than (1 → 0)	Ends after the second transfer
				0	—	0	(1 → 0)	Ends after the second transfer with an interrupt request to the CPU
				0	—	1	—	
1	1	1	Other than (1 → *)	—	—	—	—	Ends after the first transfer with an interrupt request to the CPU

Note 1. The transfer counter used depends on the transfer modes as follows:
Normal transfer mode — CRA register
Repeat transfer mode — CRAL register
Block transfer mode — CRB register

Note 2. On completion of a data transfer, the counters operate as follows:
1 → 0 in normal and block transfer modes
1 → CRAH in repeat transfer mode
(1 → *) in the table indicates both of these two operations, depending on the mode.

Note 3. Chain transfer can be selected for the second or subsequent transfers. The conditions for the combination of the second transfer and CHNE = 1 is omitted.

14.4.1 Transfer Information Read Skip Function

Reading of vector addresses and transfer information can be skipped by setting the DTCCR.RRS bit. When a DTC activation request is generated, the current DTC vector number is compared with the DTC vector number in the previous activation process. When these vector numbers match and the RRS bit is set to 1, the DTC data transfer is performed without reading the vector address and transfer information. However, when the previous transfer is a chain transfer, the vector address and transfer information are read. Additionally, when the transfer counter (CRA register) becomes 0 during the previous normal transfer, and when the transfer counter (CRB register) becomes 0 during the previous block transfer, transfer information is read regardless of the RRS bit. Figure 14.12 shows an example when reading the transfer information is skipped.

To update the vector table and transfer information, set the RRS bit to 0, update the vector table and transfer information, then set the RRS bit to 1. The stored vector number is discarded by setting the RRS bit to 0. The updated DTC vector table and transfer information are read in the next activation process.

14.4.2 Transfer Information Write-Back Skip Function

When the MRA.SM[1:0] bits or the MRB.DM[1:0] bits are set to address fixed, a part of the transfer information is not written back. Table 14.4 lists the transfer information write-back skip conditions and the associated registers. The CRA and CRB registers are written back, and the write-back of the MRA and MRB registers is skipped.

表 14.3 链转移条件

首次转会				第二次转移*3				DTC转账
CHNE 比特	CHNS 位	柴油钻头	转账计数器*1*2	CHNE 比特	CHNS 位	柴油钻头	转账计数器*1 *2	
0	—	0	除了(1 → 0)	—	—	—	—	首次转账后结束
0	—	0	(1 → 0)	—	—	—	—	第一次传输结束后，会向 CPU 发出中断请求，传输过程结束。
0	—	1	—	—	—	—	—	
1	0	—	—	0	—	0	除了(1 → 0)	第二次转会后结束
				0	—	0	(1 → 0)	第二次传输结束后，向 CPU 发出中断请求，传输结束。
				0	—	1	—	
1	1	0	除了(1 → *)	—	—	—	—	首次转账后结束
1	1	—	(1 → *)	0	—	0	除了(1 → 0)	第二次转会后结束
				0	—	0	(1 → 0)	第二次传输结束后，向 CPU 发出中断请求，传输结束。
				0	—	1	—	
1	1	1	除了(1 → *)	—	—	—	—	第一次传输结束后，会向 CPU 发出中断请求，传输过程结束。

注1. 所使用的传输计数器取决于传输模式，具体如下：
正常转账模式— CRA 注册
重复传输模式— CRAL寄存器
块传输模式— CRB 寄存器

注2：数据传输完成后，计数器按如下方式运行：
1 → 0在正常和块传输模式下
1 → CRAH重复传输模式
表格中的(1 → *)表示这两种操作，具体取决于模式。

注3. 链式转移可用于第二次或后续转移。第二次转移的组合条件如下。
并且省略了CHNE = 1。

14.4.1 传输信息读取跳过功能

通过设置 DTCCR.RRS 位，可以跳过读取向量地址和传输信息。生成 DTC 激活请求时，会将当前 DTC 向量号与上一次激活过程中的 DTC 向量号进行比较。如果这两个向量号匹配且 RRS 位设置为 1，则执行 DTC 数据传输，而无需读取向量地址和传输信息。但是，如果上一次传输是链式传输，则会读取向量地址和传输信息。此外，如果在上一次正常传输期间传输计数器（CRA 寄存器）变为 0，或者在上一次块传输期间传输计数器（CRB 寄存器）变为 0，则无论 RRS 位如何设置，都会读取传输信息。图 14.12 显示了跳过读取传输信息的示例。

要更新矢量和传输信息，请将 RRS 位设置为 0，更新矢量和传输信息，然后将 RRS 位设置为 1。将 RRS 位设置为 0 会丢弃存储的向量号。更新后的 DTC 矢量和传输信息将在下次激活过程中读取。

14.4.2 传输信息写回跳过功能

当 MRA.SM[1:0] 位或 MRB.DM[1:0] 位被设置为地址固定时，部分传输信息不会被写回。表 14.4 列出了传输信息写回跳过条件以及相关的寄存器。CRA 和 CRB 寄存器会被写回，而 MRA 和 MRB 寄存器的写回会被跳过。

Table 14.4 Transfer information write-back skip conditions and applicable registers

MRA.SM[1:0] bits		MRB.DM[1:0] bits		SAR register	DAR register
b3	b2	b3	b2		
0	0	0	0	Skip	Skip
0	0	0	1		
0	1	0	0		
0	1	0	1		
0	0	1	0	Skip	Write-back
0	0	1	1		
0	1	1	0		
0	1	1	1		
1	0	0	0	Write-back	Skip
1	0	0	1		
1	1	0	0		
1	1	0	1		
1	0	1	0	Write-back	Write-back
1	0	1	1		
1	1	1	0		
1	1	1	1		

14.4.3 Normal Transfer Mode

The normal transfer mode allows a 1-byte (8 bit), 1-halfword (16 bit), 1-word (32 bit) data transfer on a single activation source. The transfer count can be set from 1 to 65536. Transfer source and destination addresses can be independently set to increment, decrement, or fixed. This mode enables an interrupt request to the CPU to be generated at the end of a specified-count transfer.

Table 14.5 lists register functions in normal transfer mode, and Figure 14.5 shows the memory map of normal transfer mode.

Table 14.5 Register functions in normal transfer mode

Register	Description	Value written back by writing transfer information
SAR	Transfer source address	Increment, decrement, or fixed*1
DAR	Transfer destination address	Increment, decrement, fixed*1
CRA	Transfer counter A	CRA - 1
CRB	Transfer counter B	Not updated

Note 1. Write-back operation is skipped in address-fixed mode.

表 14.4 传输信息写回跳过条件及适用寄存器

MRA.SM[1:0] bits		MRB.DM[1:0] bits		SAR登记册	DAR登记册
b3	b2	b3	b2		
0	0	0	0	跳过	跳过
0	0	0	1		
0	1	0	0		
0	1	0	1		
0	0	1	0	跳过	写回
0	0	1	1		
0	1	1	0		
0	1	1	1		
1	0	0	0	写回	跳过
1	0	0	1		
1	1	0	0		
1	1	0	1		
1	0	1	0	写回	写回
1	0	1	1		
1	1	1	0		
1	1	1	1		

14.4.3 正常传输模式

正常传输模式允许在单个激活源上进行 1 字节（8 位）、1 个半字（16 位）或 1 个字（32 位）的数据传输。传输计数可设置为 1 到 65536。传输源地址和目标地址可以独立设置为递增、递减或固定。此模式允许在指定计数的传输结束后向 CPU 发送中断请求。

表 14.5 列出了正常传输模式下的寄存器功能，图 14.5 显示了正常传输模式下的内存映射。

表 14.5 正常传输模式下的寄存器功能

登记	描述	通过写入转账信息写回的价值
SAR	转移源地址	增量、减量或固定值*1
DAR	转运目的地地址	增量、减量、固定值*1
CRA	转账计数器 A	CRA - 1
CRB	转账计数器 B	未更新

注1. 在地址固定模式下，会跳过写回操作。

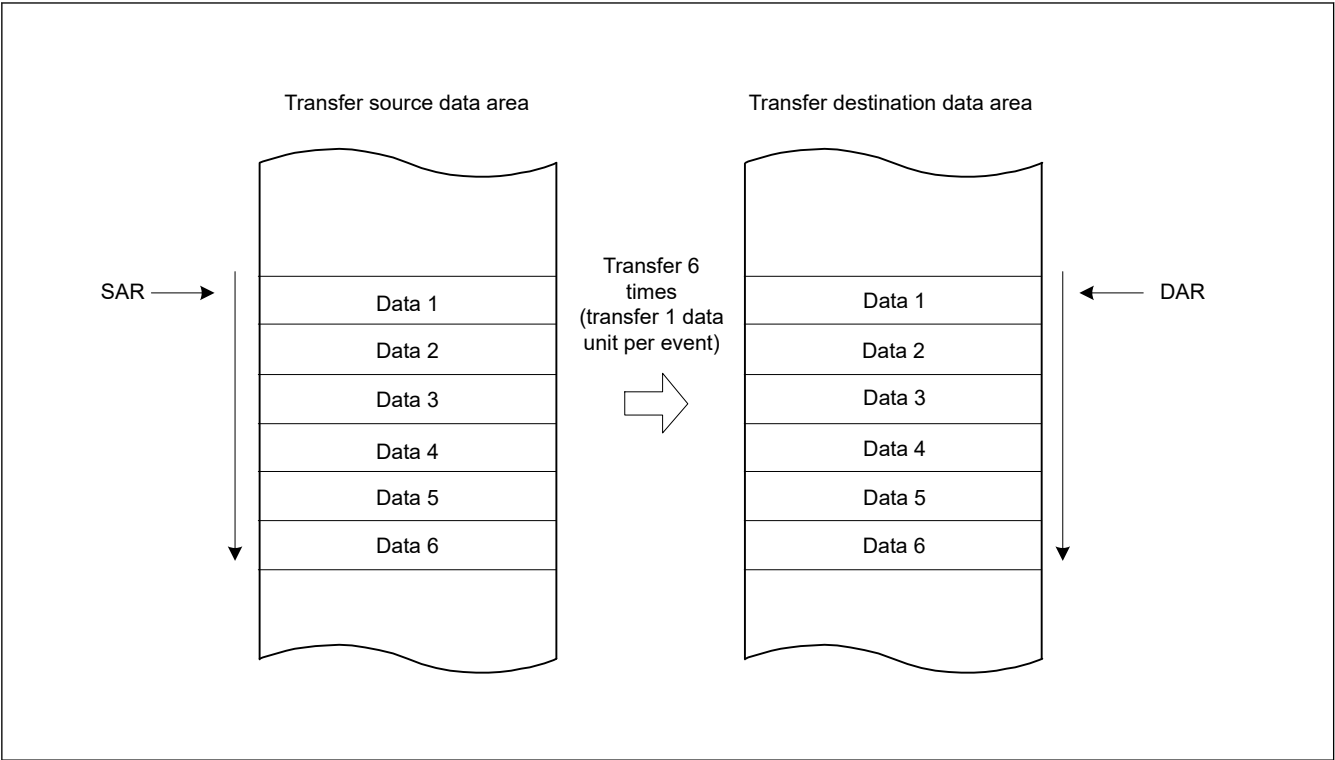


Figure 14.5 Memory map of normal transfer mode (MRA.SM[1:0] = 10b, MRB.DM[1:0] = 10b, CRA = 0x0006)

14.4.4 Repeat Transfer Mode

The repeat transfer mode allows a 1-byte (8-bit), 1-halfword (16-bit), or 1-word (32-bit) data transfer on a single activation source. Transfer source or transfer destination for the repeat area must be specified in the MRB.DTS bit. The transfer count can be set from 1 to 256. When the specified transfer count is complete, the initial value of the address register specified in the repeat area is restored, the initial value of the transfer counter is restored, and transfer is repeated. The other address register is incremented or decremented continuously or remains unchanged.

When the transfer counter CRAL decrements to 0x00 in repeat transfer mode, the CRAL value is updated to the value set in the CRAH register. As a result, the transfer counter does not clear to 0x00, which disables interrupt requests to the CPU when the MRB.DISEL bit is set to 0. An interrupt request to the CPU is generated when the specified data transfer completes.

Table 14.6 lists the register functions in repeat transfer mode, and Figure 14.6 shows the memory map of repeat transfer mode.

Table 14.6 Register functions in repeat transfer mode

Register	Description	Value written back by writing transfer information	
		When CRAL is not 1	When CRAL is 1
SAR	Transfer source address	Increment, decrement, fixed*1	- When the MRB.DTS bit is 0 Increment, decrement, or fixed*1 - When the MRB.DTS bit is 1 SAR register initial value
DAR	Transfer destination address	Increment, decrement, or fixed*1	- When the MRB.DTS bit is 0 DAR register initial value - When the MRB.DTS bit is 1 Increment, decrement, or fixed*1
CRAH	Retains transfer counter	CRAH	CRAH
CRAL	Transfer counter A	CRAL - 1	CRAH
CRB	Transfer counter B	Not updated	Not updated

Note 1. Write-back is skipped in address-fixed mode.

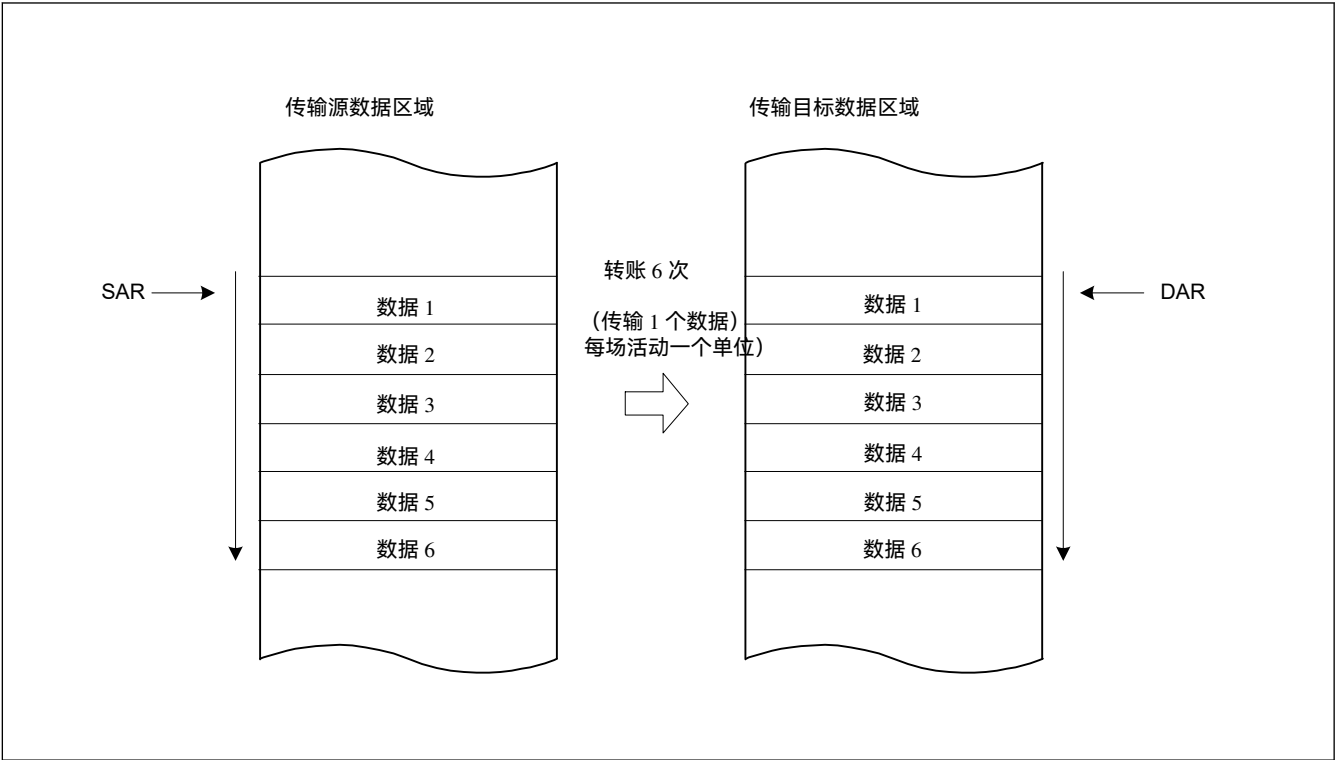


图 14.5 正常传输模式的内存映射 (MRA.SM[1:0] = 10b, MRB.DM[1:0] = 10b, CRA = 0x0006)

14.4.4 重复传输模式

重复传输模式允许在单个激活源上进行 1 字节（8 位）、1 个半字（16 位）或 1 个字（32 位）的数据传输。重复区域的传输源或传输目标必须在 MRB.DTS 位中指定。传输计数可以设置为 1 到 256。当指定的传输计数完成后，重复区域中指定的地址寄存器的初始值将被恢复，传输计数器的初始值也将恢复，然后重复传输。另一个地址寄存器的值将持续递增或递减，或者保持不变。

在重复传输模式下，当传输计数器 CRAL 递减至 0x00 时，CRAL 值会被更新为 CRAH 寄存器中设置的值。因此，传输计数器不会清零至 0x00，从而在 MRB.DISEL 位设置为 0 时禁用对 CPU 的中断请求。当指定的数据传输完成时，会生成一个对 CPU 的中断请求。

表 14.6 列出了重复传输模式下的寄存器功能，图 14.6 显示了重复传输模式的内存映射。

表 14.6 重复传输模式下的寄存器功能

登记	描述	通过写入转账信息写回的价值	
		当 CRAL 不为 1 时	当 CRAL 为 1 时
SAR	转移源地址	增量、减量、固定值*1	●当 MRB.DTS 位为 0 时 增量、减量或固定值*1 ●当 MRB.DTS 位为 1 时 SAR 寄存器初始值
DAR	转运目的地地址	增量、减量或固定值*1	●当 MRB.DTS 位为 0 时 DAR 寄存器初始值 ●当 MRB.DTS 位为 1 时 增量、减量或固定值*1
CRAH	保留转账计数器	CRAH	CRAH
CRAL	转账计数器 A	CRAL - 1	CRAH
CRB	转账计数器 B	未更新	未更新

注1. 在地址固定模式下，会跳过写回操作。

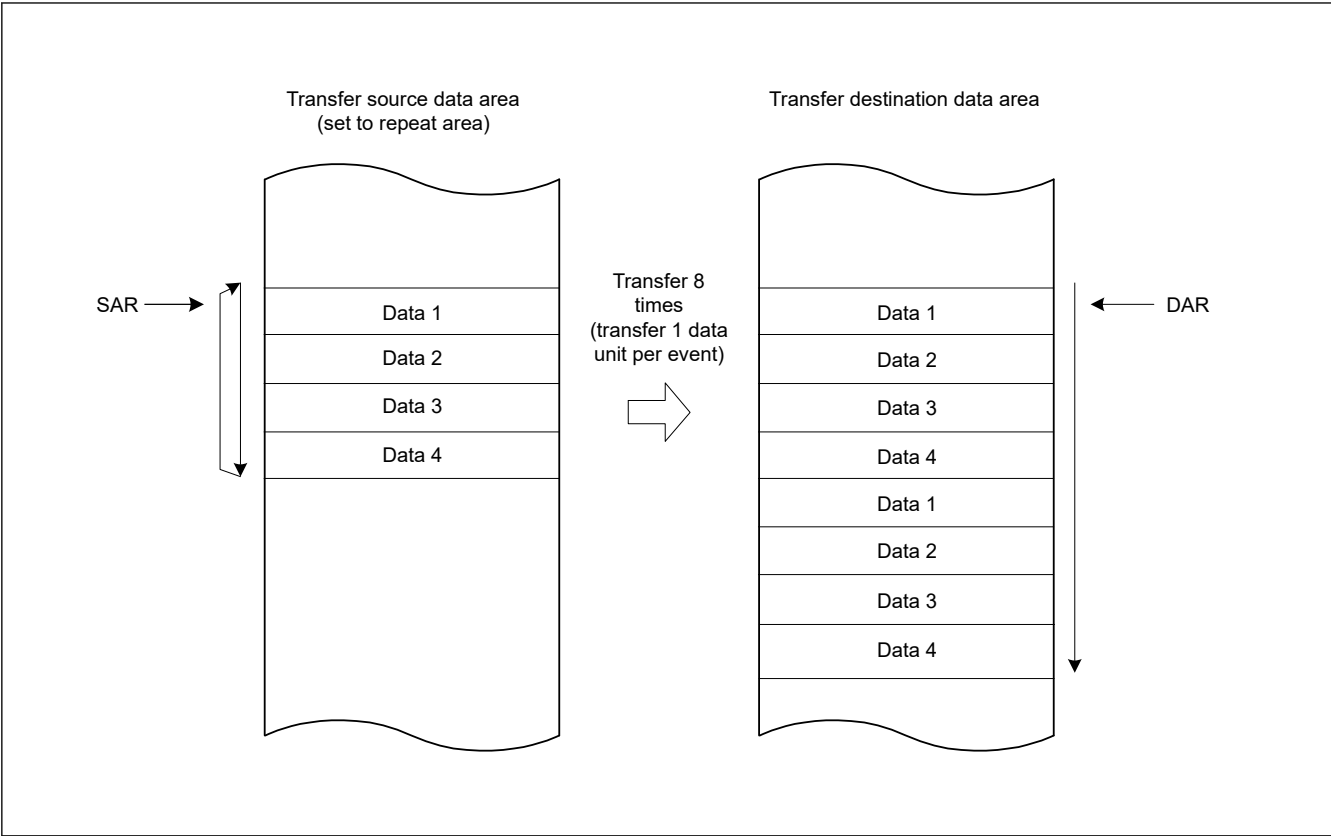


Figure 14.6 Memory map of repeat transfer mode when transfer source is a repeat area (MRA.SM[1:0] = 10b, MRB.DM[1:0] = 10b, CRAH = 0x04)

14.4.5 Block Transfer Mode

The block transfer mode allows single-block data transfer on a single activation source. Transfer source or transfer destination for the block area must be specified in the MRB.DTS bit. The block size can be set from 1 to 256 bytes, 1 to 256 halfwords (2 to 512 bytes), or 1 to 256 words (4 to 1024 bytes). When transfer of the specified block completes, the initial values of the block size counter CRAL and the address register (the SAR register when the MRB.DTS = 1 or the DAR register when the DTS = 0) specified in the block area are restored. The other address register is incremented or decremented continuously or remains unchanged.

The transfer count (block count) can be set from 1 to 65536. This mode enables an interrupt request to the CPU to be generated at the end of the specified-count block transfer.

Table 14.7 lists the register functions in block transfer mode, and Figure 14.7 shows the memory map for block transfer mode.

Table 14.7 Register functions in block transfer mode

Register	Description	Value written back by writing transfer information
SAR	Transfer source address	- When MRB.DTS bit is 0 Increment, decrement, or fixed*1 - When MRB.DTS bit is 1 SAR register initial value.
DAR	Transfer destination address	- When MRB.DTS bit is 0 DAR register initial value - When MRB.DTS bit is 1 Increment, decrement, or fixed*1.
CRAH	Holds block size	CRAH
CRAL	Block size counter	CRAH
CRB	Block transfer counter	CRB - 1

Note 1. Write-back is skipped in address-fixed mode.

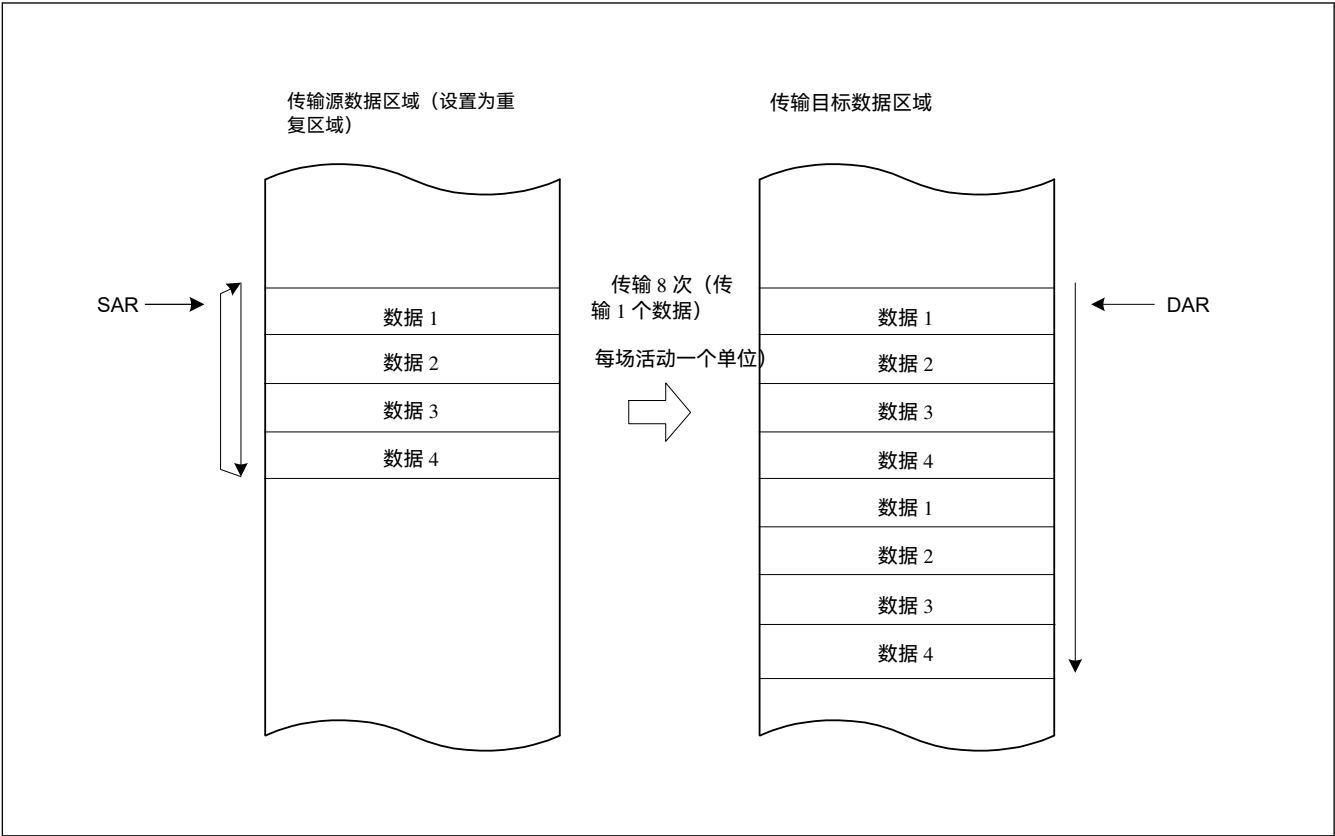


图 14.6 当传输源为重复区域 (MRA.SM[1:0] = 10b, MRB.DM[1:0] = 10b, CRAH = 0x04)

14.4.5 块传输模式

块传输模式允许在单个激活源上进行单块数据传输。必须在 MRB.DTS 位中指定块区域的传输源或传输目标。块大小可以设置为 1 到 256 字节、1 到 256 个半字 (2 到 512 字节) 或 1 到 256 个字 (4 到 1024 字节)。当指定的块传输完成后，块大小计数器 CRAL 和地址寄存器 (当 MRB.DTS = 1 为块区域中指定的地址时，为 SAR 寄存器；当 MRB.DTS = 0) 为块区域中指定的地址时，为 DAR 寄存器) 的初始值将被恢复。另一个地址寄存器的值将连续递增或递减，或者保持不变。

传输计数 (块计数) 可以设置为 1 到 65536。此模式允许在指定计数块传输结束时向 CPU 生成中断请求。

表 14.7 列出了块传输模式下的寄存器功能，图 14.7 显示了块传输模式下的内存映射。

表 14.7 块传输模式下的寄存器功能

登记	描述	通过写入转账信息写回的价值
SAR	转移源地址	- 当 MRB.DTS 位为 0 时，递增、递减或固定值*1 - 当 MRB.DTS 位为 1 时，SAR 寄存器初始值。
DAR	转运目的地地址	- 当 MRB.DTS 位为 0 时，DAR 寄存器初始值为 - 当 MRB.DTS 位为 1 时，递增、递减或固定*1。
CRAH	保存块大小	CRAH
CRAL	区块大小计数器	CRAH
CRB	区块传输计数器	CRB - 1

注1. 在地址固定模式下，会跳过写回操作。

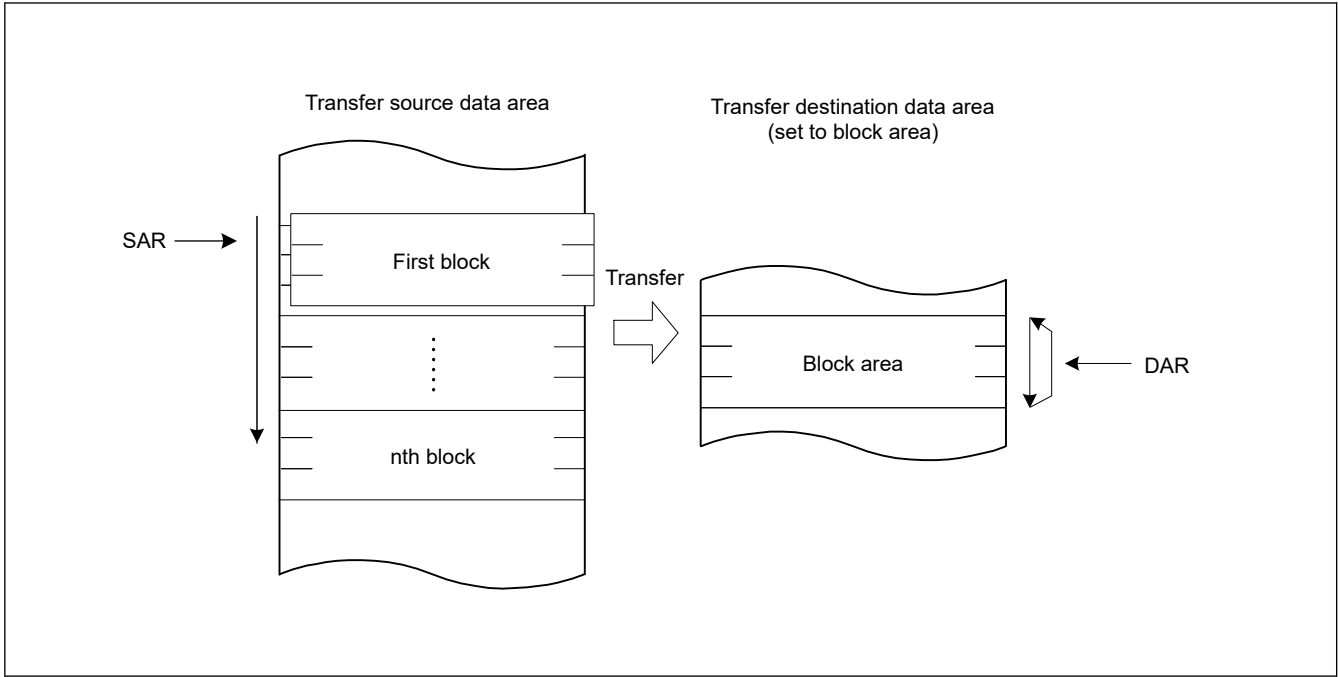


Figure 14.7 Memory map of block transfer mode

14.4.6 Chain Transfer

Setting the MRB.CHNE bit to 1 allows chain transfer to be performed continuously on a single activation source. If the MRB.CHNE is set to 1 and CHNS to 0, an interrupt request to the CPU is not generated on completion of the specified number of rounds of transfer or by setting the MRB.DISEL bit to 1. An interrupt request is sent to the CPU each time DTC data transfer is performed.

The SAR, DAR, CRA, CRB, MRA, and MRB registers can be set independently of each other to define the data transfer. Figure 14.8 shows a chain transfer operation.

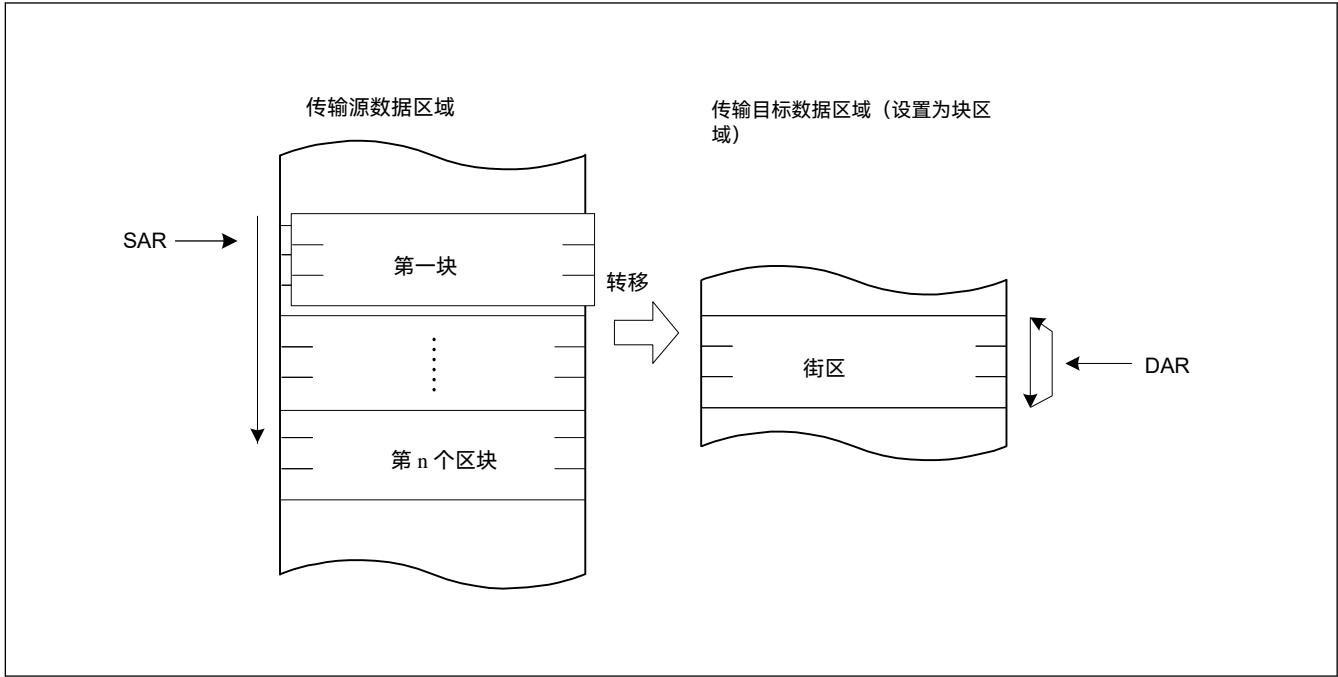


图 14.7 块传输模式的内存映射

14.4.6 链式转移

将 MRB.CHNE 位设置为 1 允许在单个激活源上连续执行链式传输。如果 MRB.CHNE 设置为 1 且 CHNS 设置为 0, 则在完成指定轮数的传输或将 MRB.DISEL 位设置为 1 时, 不会向 CPU 发送中断请求。每次执行 DTC 数据传输时, 都会向 CPU 发送中断请求。

SAR、DAR、CRA、CRB、MRA 和 MRB 寄存器可以彼此独立设置, 以定义数据传输。图 14.8 显示了一个链转移操作。

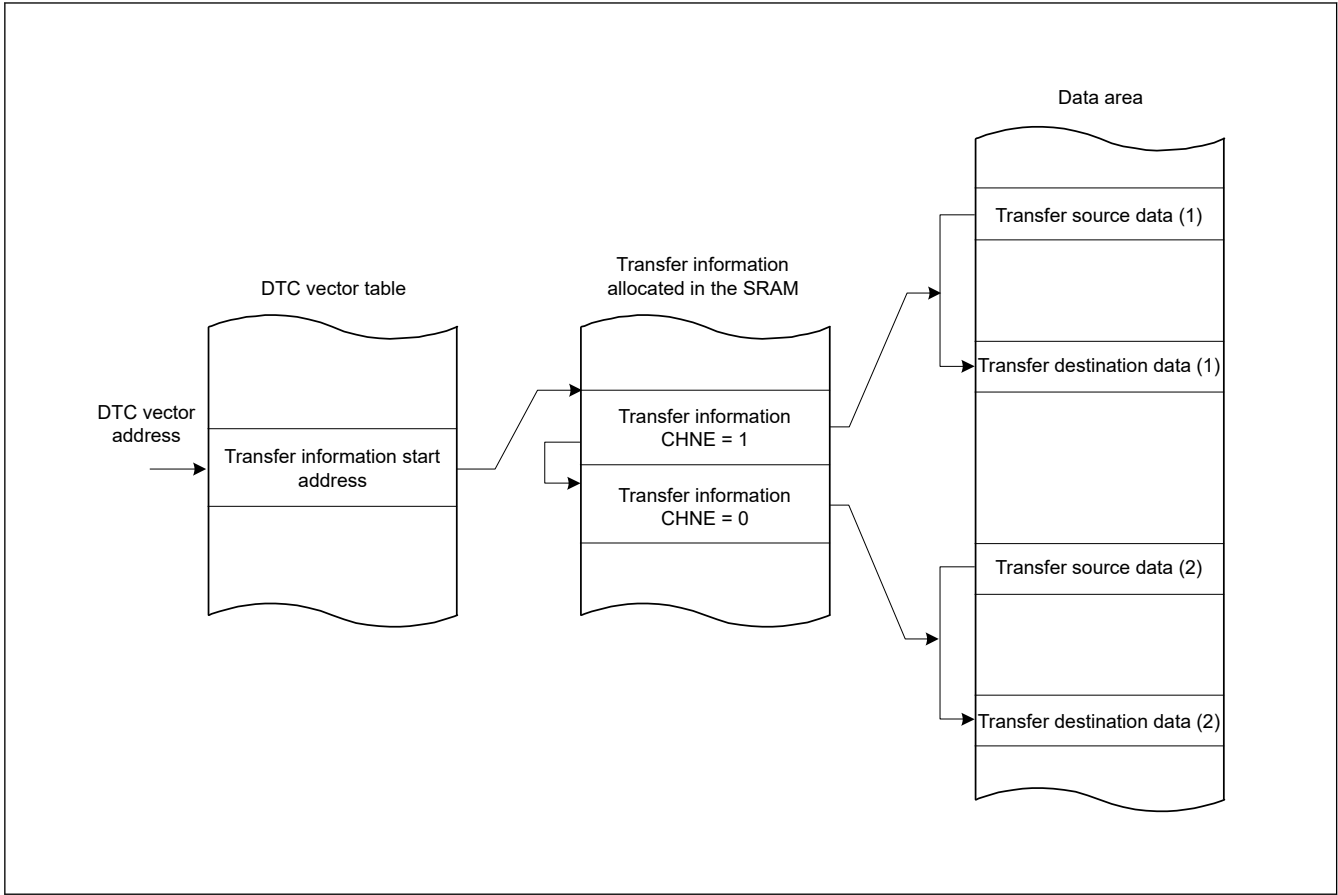


Figure 14.8 Chain transfer operation

Writing 1 to the MRB.CHNE and CHNS bits enables chain transfer to be performed only after completion of the specified data transfer. In repeat transfer mode, chain transfer is performed after completion of the specified data transfer. For details on chain transfer conditions, see [Table 14.3](#).

14.4.7 Operation Timing

[Figure 14.9](#) to [Figure 14.12](#) are timing diagrams that show the minimum number of execution cycles.

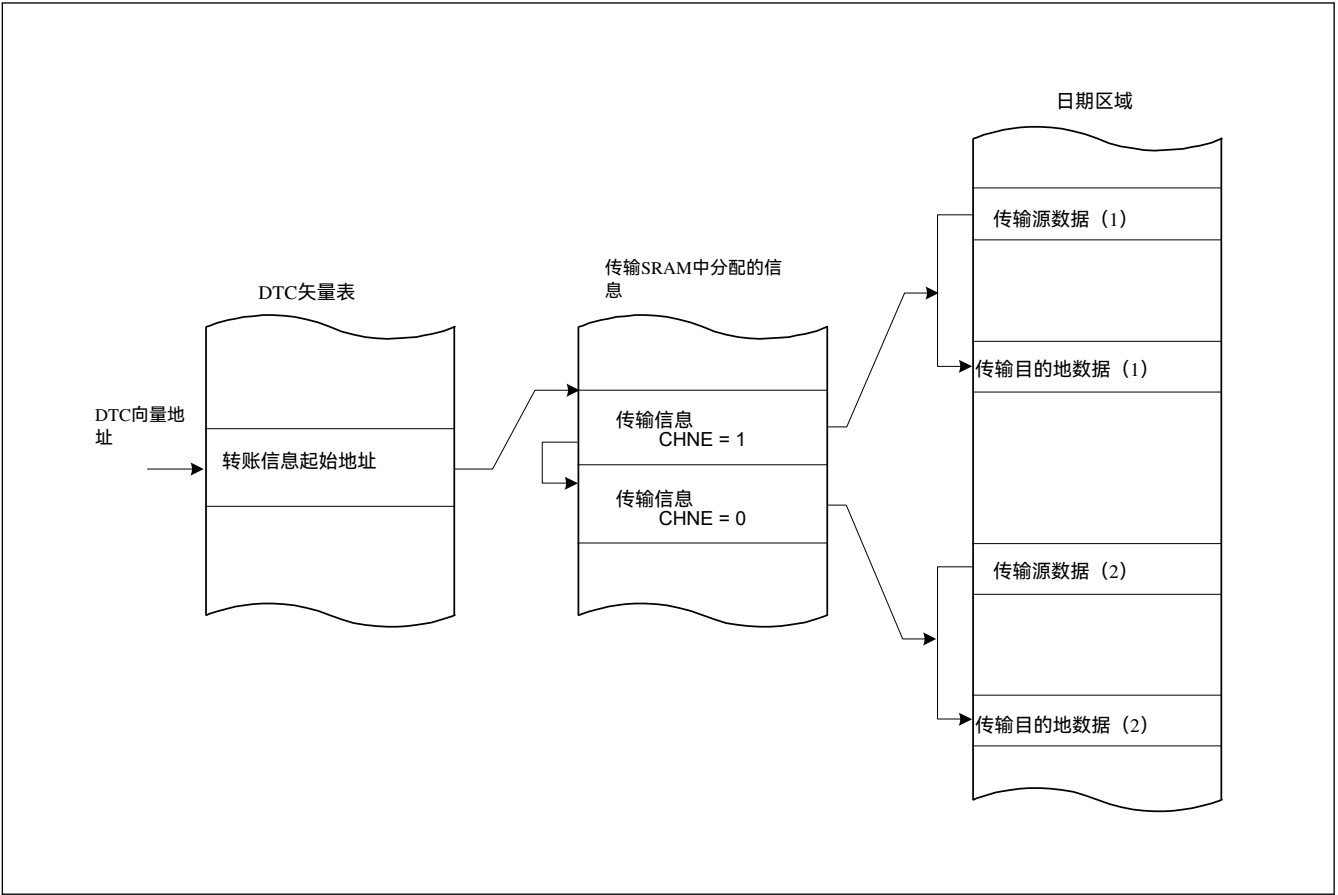


图 14.8 链转移操作

将 MRB.CHNE 和 CHNS 位写入 1 可使链式传输仅在指定的数据传输完成后执行。在重复传输模式下，链式传输在指定的数据传输完成后执行。有关链式传输条件的详细信息，请参见表 14.3。

14.4.7 操作时序

图 14.9 至图 14.12 是时序图，显示了最小执行周期数。

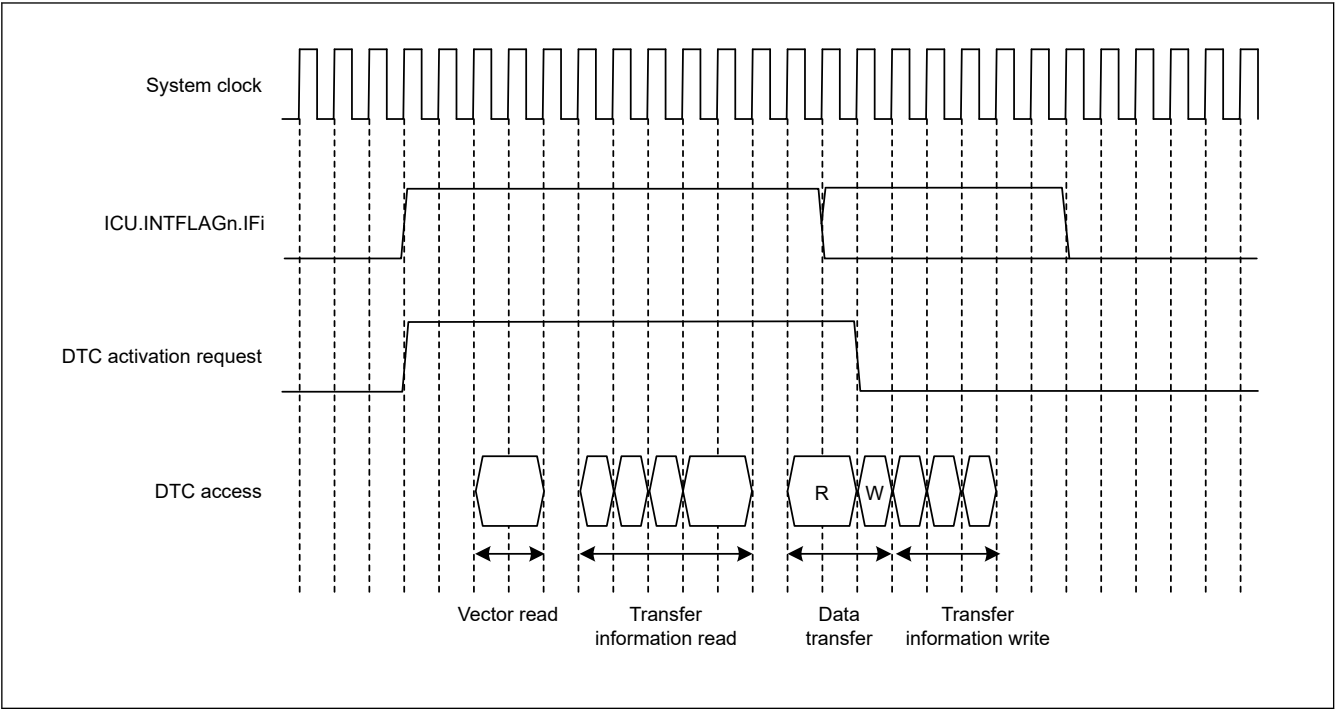


Figure 14.9 Example 1 of DTC operation timing in normal transfer and repeat transfer modes

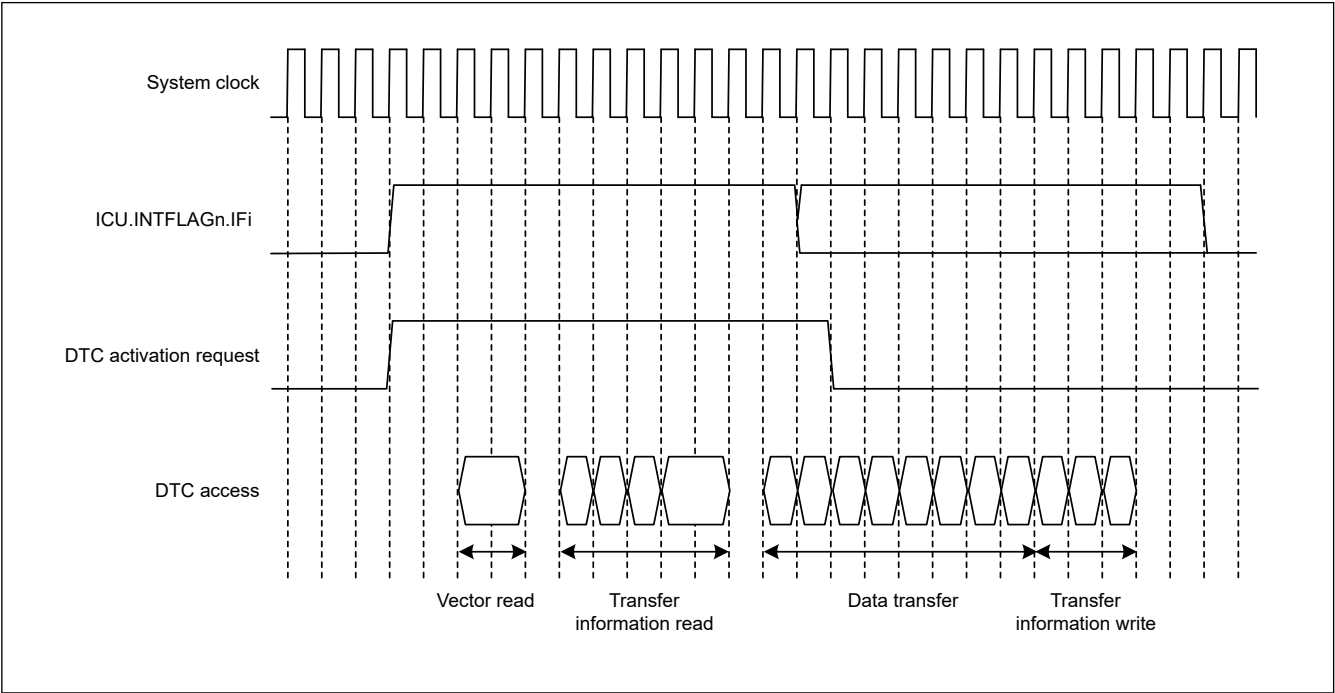


Figure 14.10 Example 2 of DTC operation timing in block transfer mode when the block size = 4

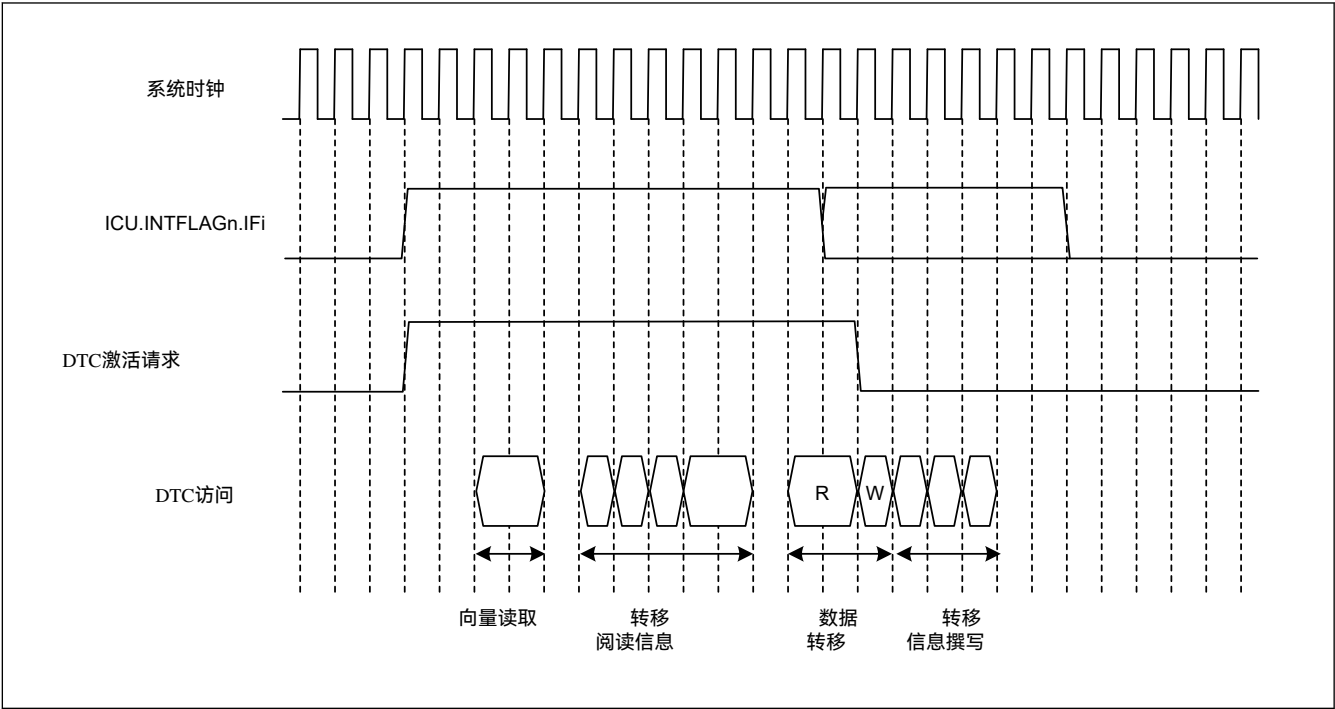


图 14.9 正常传输和重复传输模式下 DTC 操作时序示例 1

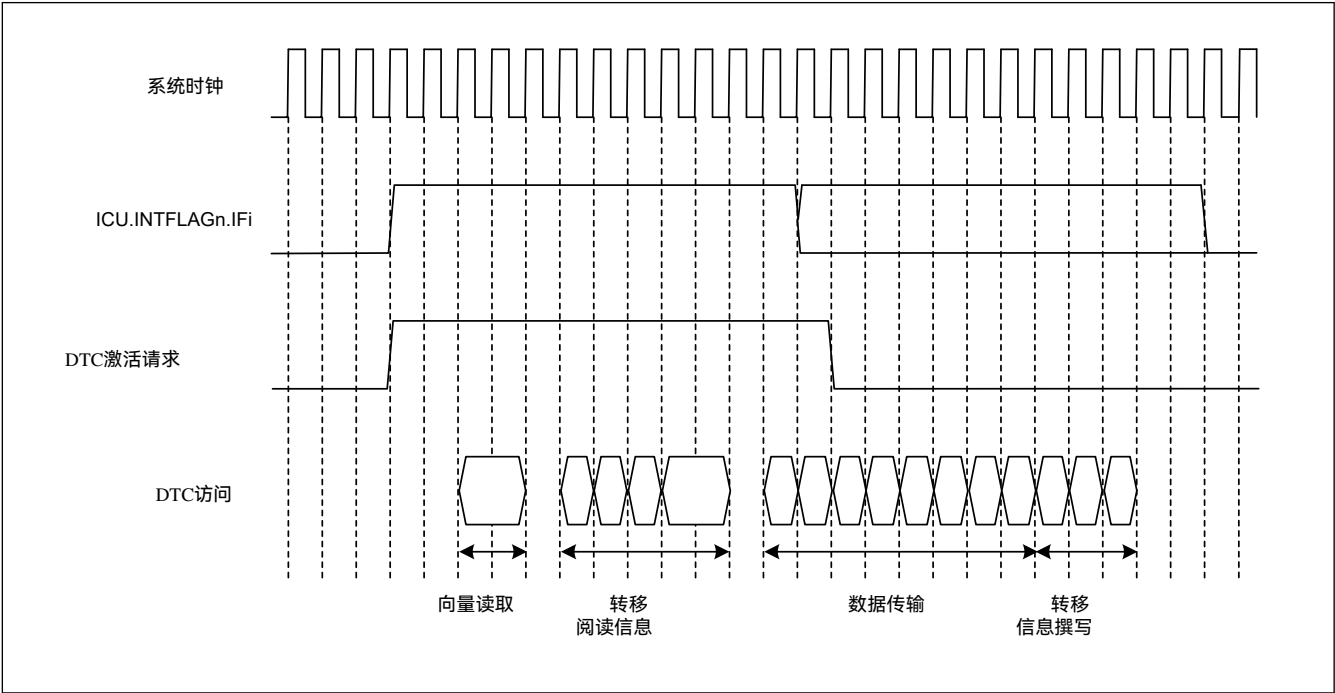


图 14.10 示例 2: 块传输模式下 DTC 操作时序, 当块大小为= 4

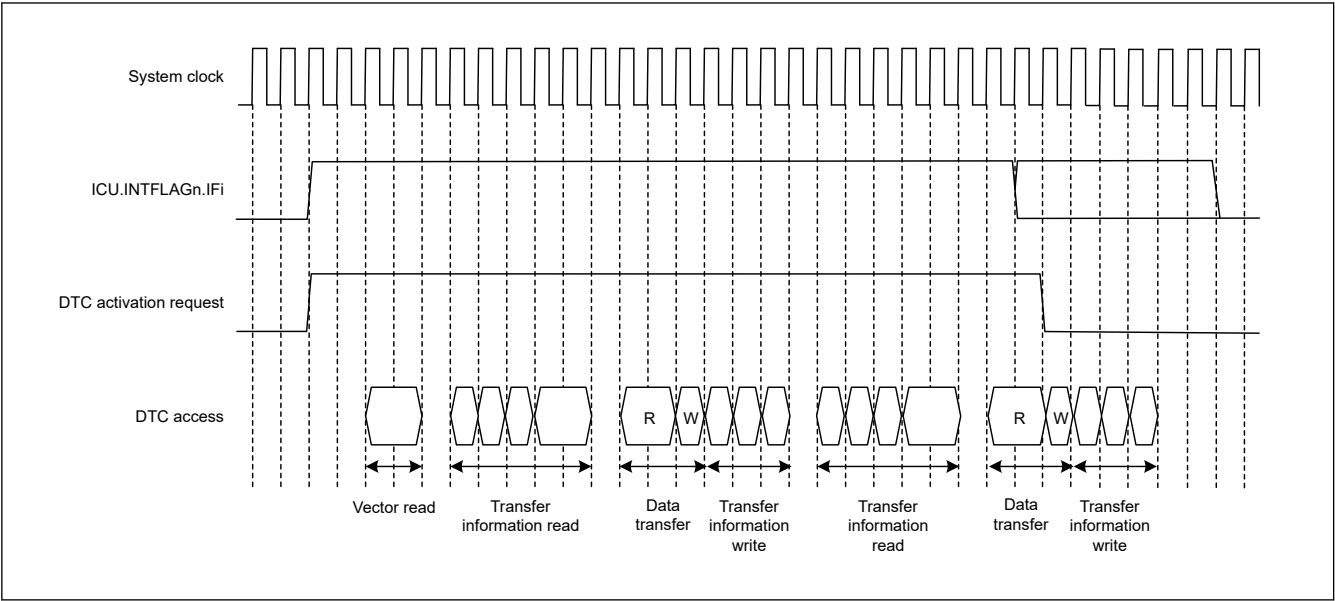


Figure 14.11 Example 3 of DTC operation timing for chain transfer

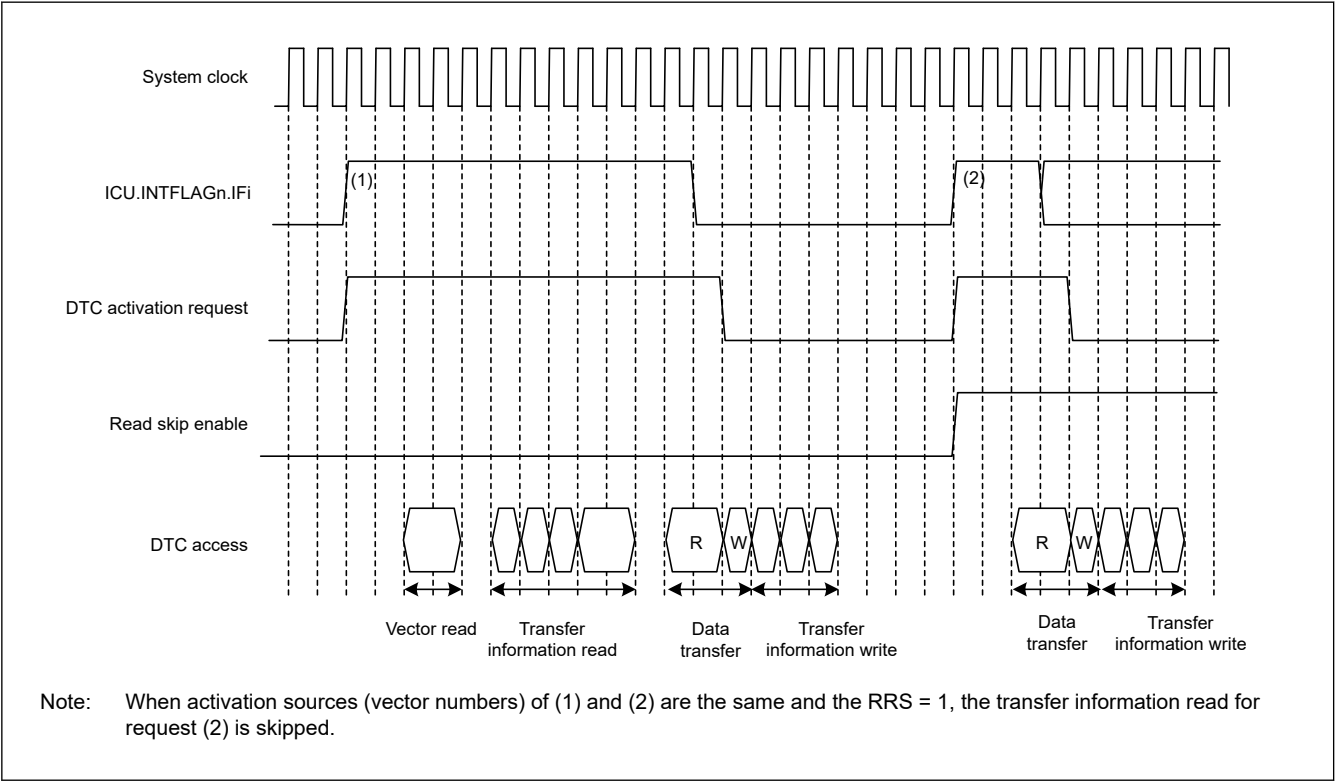


Figure 14.12 Example of operation when a transfer information read is skipped with the vector, transfer information, and transfer destination data on the SRAM, and the transfer source data on the peripheral module

14.4.8 Execution Cycles of DTC

Table 14.8 lists the execution cycles of single data transfer of the DTC. For the order of the execution states, see [section 14.4.7. Operation Timing](#).

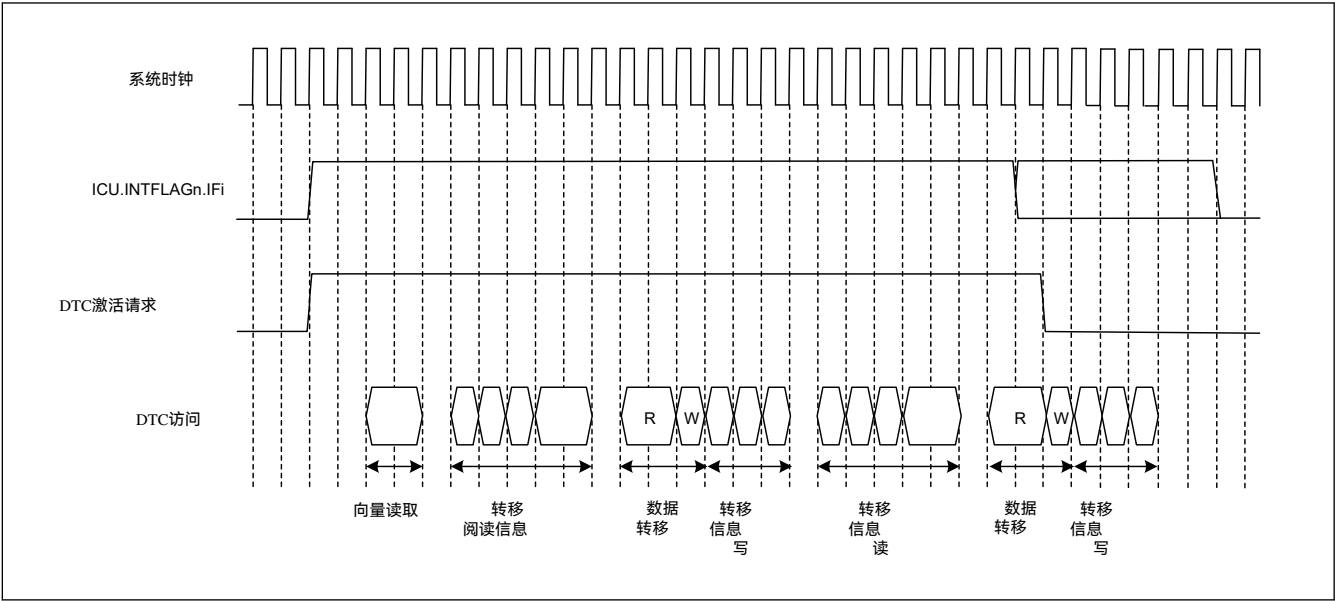


图 14.11 链式传动 DTC 操作时序示例 3

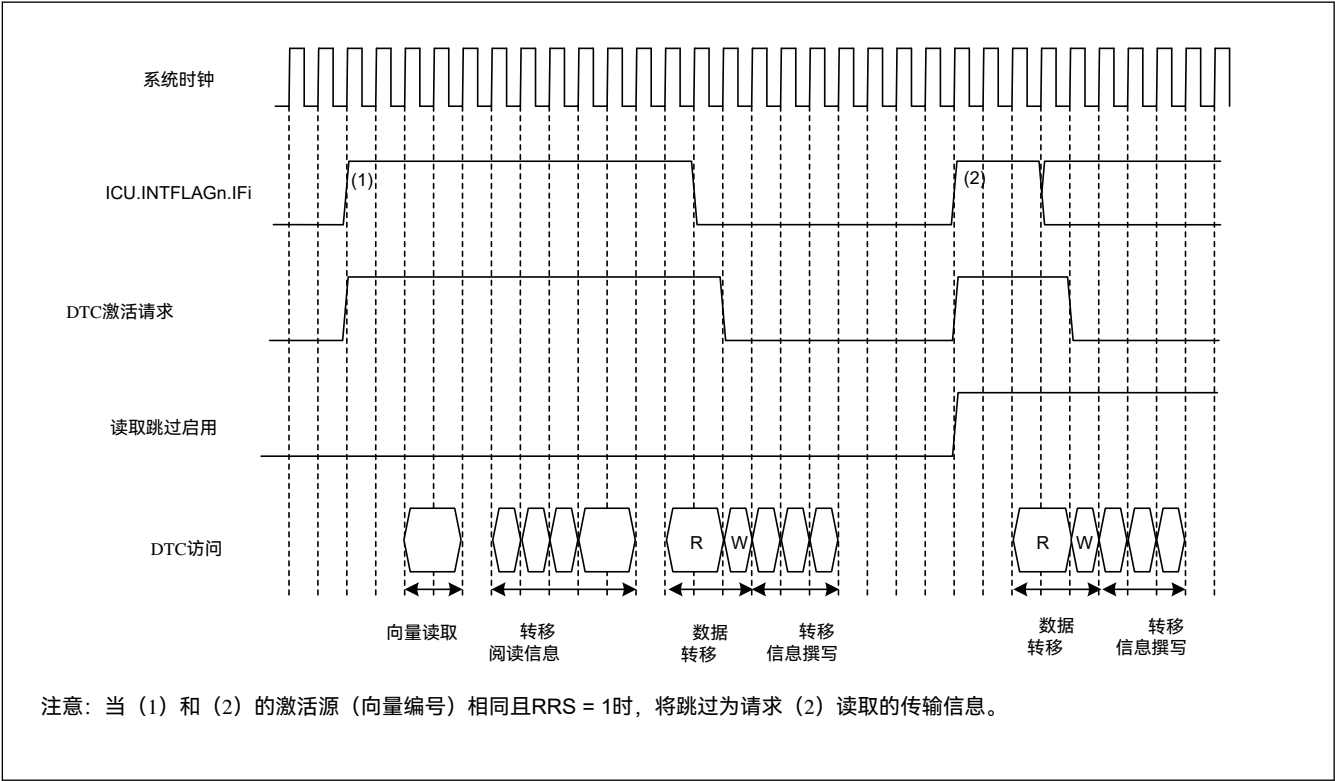


图 14.12 示例：当 SRAM 中存储有向量、传输信息和传输目标数据，而外设模块中存储有传输源数据时，跳过一次传输信息读取的操作。

14.4.8 DTC的执行周期

表 14.8 列出了 DTC 的单次数据传输的执行周期。有关执行状态的顺序，请参见第 14.4.7 节“操作时序”。

Table 14.8 Execution cycles of DTC

P: Block size (initial settings of CRAH and CRAL)
Cv: Cycles for access to vector transfer information storage destination
Ci: Cycles for access to transfer information storage destination address
Cr: Cycles for access to data read destination
Cw: Cycles for access to data write destination
The unit is for system clocks (ICLK) + 1 in the Vector read, Transfer information read, and Data transfer read columns and 2 in the Internal operation column.
Cv, Ci, Cr, and Cw vary depending on the corresponding access destination. For the number of cycles for respective access destinations, see [section 24, SRAM](#), [section 25, Flash Memory](#), and [section 12, Buses](#).
The frequency ratio of the system clock and peripheral clock is also taken into consideration.
The DTC response time is the time from when the DTC activation source is detected until DTC transfer starts.
[Table 14.8](#) does not include the time until DTC data transfer starts after the DTC activation source becomes active.

Transfer mode	Vector read		Transfer information read		Transfer information write			Data transfer		Internal operation	
								Read	Write		
Normal	Cv + 1	0*1	4 × Ci + 1	0*1	3 × Ci + 1*2	2 × Ci + 1*3	Ci*4	Cr + 1	Cw + 1	2	0*1
Repeat								Cr + 1	Cw + 1		
Block*5								P × Cr	P × Cw		

- Note 1. When transfer information read is skipped.
Note 2. When neither SAR nor DAR is set to address-fixed mode.
Note 3. When SAR or DAR is set to address-fixed mode.
Note 4. When SAR and DAR are set to address-fixed mode.
Note 5. When the block size is 2 or more. If the block size is 1, the cycle number for normal transfer applies.

14.4.9 DTC Bus Mastership Release Timing

The DTC does not release the bus mastership during transfer information reads. Before the transfer information is read or written, the bus is arbitrated according to the priority determined by the bus master arbitrator. For bus arbitration, see [section 12, Buses](#).

14.5 DTC Setting Procedure

Before using the DTC, set the DTC Vector Base Register (DTCVBR). Set to disable the interrupt in the NVIC and follow the procedure in [Table 14.9](#) to set the DTC.

Table 14.9 DTC setting procedure

No.	Step Name	Description
1	Set the DTCCR.RRS bit to 0	Set the DTCCR.RRS bit to 0 to reset the transfer information read skip flag. After that, the transfer information read is not skipped while the DTC is activated. Be sure to specify this setting when the transfer information is updated.
2	Set transfer information (MRA, MRB, SAR, DAR, CRA, and CRB)	Allocate transfer information (MRA, MRB, SAR, DAR, CRA, and CRB) in the data area. To set transfer information, see section 14.2. Register Descriptions . To allocate transfer information, see section 14.3.1. Allocating Transfer Information and DTC Vector Table .
3	Set transfer information start addresses in the DTC vector table	Set the transfer information start addresses in the DTC vector table. To set the DTC vector table, see section 14.3.1. Allocating Transfer Information and DTC Vector Table .
4	Set the DTCCR.RRS bit to 1	Set the DTCCR.RRS bit to 1 to enable skipping of the second and subsequent transfer information read cycles for continuous DTC activation from the same interrupt source. The RRS bit can be set to 1, but if this is set during DTC transfer, it becomes valid from the next transfer.
5	Set the ICU.DTCENSTx.STy bit to 1. The interrupt should be enabled in the NVIC.	Set the ICU.DTCENSTx.STy bit to 1. The interrupt must be enabled in the NVIC.
6	Set the enable bit for an activation source interrupt	Set the enable bit for the activation source interrupts to 1. When a source interrupt is generated, the DTC is activated. To set the interrupt source enable bit, see the settings for the modules that are to be the activation sources.
7	Set the DTCST.DTCST bit to 1	Set the DTC Module Start bit (DTCST.DTCST) to 1.

Note: The DTCST.DTCST bit can be set even if the setting for each activation source is not completed.

表 14.8 DTC 的执行周期

P: 块大小（CRAH 和 CRAL 的初始设置）
Cv: 用于访问矢量传输信息存储目标的周期
Ci: 访问传输信息存储目标地址所需的周期
Cr: 访问数据读取目标所需的周期
Cw: 访问数据写入目标的周期
该单位用于向量读取、传输信息读取和数据传输读取列中的系统时钟(ICLK) + 1 以及内部操作列中的2。

Cv、Ci、Cr 和 Cw 的值取决于相应的访问目标。有关各个访问目标的周期数，请参见第 24 节 SRAM、第 25 节闪存和第 12 节总线。

系统时钟与外设时钟的频率比也被考虑在内。
DTC响应时间是指从检测到DTC激活源到DTC传输开始所需的时间。
表 14.8 不包括 DTC 激活源激活后 DTC 数据传输开始的时间。

传输模式	向量读取		已读取传输信息		传输信息写入			数据传输		内部运作	
								读	写		
普通的	Cv + 1	0*1	4 × Ci + 1	0*1	3 × Ci + 1*2	2 × Ci + 1*3	Ci*4	Cr + 1	Cw + 1	2	0*1
重复								Cr + 1	Cw + 1		
Block*5								P × Cr	P × Cw		

- 注1. 跳过传输信息读取。
注 2. 当 SAR 和 DAR 均未设置为地址固定模式时。
注 3. 当 SAR 或 DAR 设置为地址固定模式时。
注 4. 当 SAR 和 DAR 设置为地址固定模式时。
注 5. 当块大小为 2 或更大时。如果块大小为1，则适用正常传输的周期数。

14.4.9DTC 总线主控权发布时间

在读取换乘信息期间，DTC 不会释放总线主控权。在读取或写入换乘信息之前，总线会根据总线主控仲裁员确定的优先级进行仲裁。有关总线仲裁，请参阅第 12 节“总线”。

14.5DTC 设置程序

使用DTC之前，请设置DTC向量基址寄存器（DTCVBR）。将其设置为禁用NVIC中的中断，并按照表14.9中的步骤设置DTC。

表 14.9 DTC 设置程序

No.	步骤名称	描述
1	将 DTCCR.RRS 位设置为 0	将 DTCCR.RRS 位设置为 0 可重置传输信息读取跳过标志。之后，当 DTC 激活时，传输信息读取将不再跳过。更新传输信息时，请务必指定此设置。
2	设置转移信息（MRA、MRB、SAR、DAR、CRA 和 CRB）	在数据区分配传输信息（MRA、MRB、SAR、DAR、CRA 和 CRB）。要设置传输信息，请参阅第 14.2 节“寄存器描述”。要分配传输信息，请参阅第 14.3.1 节“分配传输信息和 DTC 向量表”。
3	在 DTC 向量表中设置传输信息起始地址	在 DTC 向量表中设置传输信息起始地址。要设置 DTC 向量表，请参阅第 14.3.1 节“分配传输信息和 DTC 向量表”。
4	将 DTCCR.RRS 位设置为 1	将 DTCCR.RRS 位设置为 1，可跳过第二次及后续传输信息读取周期，以便从同一中断源连续激活 DTC。RRS 位可以设置为 1，但如果在 DTC 传输期间设置此位，则从下一次传输开始生效。
5	将 ICU.DTCENSTx.STy 位设置为 1。应在 NVIC 中启用中断。	将 ICU.DTCENSTx.STy 位设置为 1。必须在 NVIC 中启用中断。
6	设置激活源中断的使能位	将激活源中断的使能位设置为 1。当产生源中断时，DTC 将被激活。要设置中断源使能位，请参阅要作为激活源的模块的设置。
7	将 DTCST.DTCST 位设置为 1	将 DTC 模块启动位 (DTCST.DTCST) 设置为 1。

注意：即使每个激活源的设置尚未完成，也可以设置 DTCST.DTCST 位。

14.6 Examples of DTC Usage

14.6.1 Normal Transfer

This section provides an example of DTC usage and its application when consecutively capturing A/D conversion results 40 times.

(1) Transfer information settings

In the MRA register, select a fixed source address (MRA.SM[1:0] = 00b), normal transfer mode (MRA.MD[1:0] = 00b), and halfword-sized transfer (MRA.SZ[1:0] = 01b). In the MRB register, specify incrementation of the destination address (MRB.DM[1:0] = 10b) and single data transfer by a single interrupt (MRB.CHNE = 0 and MRB.DISEL = 0). The MRB.DTS bit can be set to any value. Set the ADCR register address of the ADC10 in the SAR register, the start address of the SRAM area for data storage in the DAR register, and 40 (0x0028) in the CRA register. The CRB register can be set to any value.

(2) DTC vector table settings

The start address of the transfer information for the ADC10_ADI interrupt is set in the vector table for the DTC.

(3) ICU settings and DTC module activation

Set the ICU.DTCENST0.ST31 bit to 1. The interrupt must be enabled in the NVIC. Set the DTCST.DTCST bit to 1.

(4) ADC10 settings

Set the ADC10.ADUL, the ADC10.ADLL, and the ADC10.ADM2 register appropriately to enable ADC10_ADI. If ADC10_ADI does not occur, the DTC do not start the transfer. For the details of setting of the ADC10, please refer to [section 22.6. A/D Converter Setup Procedure](#).

(5) DTC transfer

Each time A/D conversion by the ADC10 is completed, an ADC10_ADI interrupt is generated to activate the DTC. The DTC transfers the A/D conversion result from the ADCR of the ADC10 to the SRAM, after which the DAR register is incremented and the CRA register is decremented.

(6) Interrupt handling

After 40 rounds of data transfer are complete and the value in the CRA register becomes 0, an ADC10_ADI interrupt request is generated for the CPU. Complete the process in the handling routine for this interrupt.

14.6.2 Chain Transfer

This section provides an example of chain transfer by the DTC and describes its use in consecutively capturing A/D conversion result and then transmitting it by UART0 of the SAU. You can use chain transfer to transfer A/D conversion results to the SRAM area and transmit it using UART0 of the SAU.

For the first of the chain transfers, normal transfer mode is specified for transfer from the ADC10.ADCR register to the SRAM area. For the second transfer, normal transfer mode is specified for transfer from the A/D conversion result in the SRAM area to the SAU0.SDR00 register. This is because clearing of the activation source and generation of an interrupt on completion of the specified number of transfers are restricted to the second of the chain transfers, that is, transfer while MRB.CHNE = 0.

The following example shows how to use the A/D conversion end interrupt as an activating source for the DTC.

(1) First transfer information setting

Set up transfer the ADC10.ADCR register to the SRAM area.

1. In the MRA register, select the source address as fixed (MRA.SM[1:0] = 00b).
2. Set the transfer to normal transfer mode (MRA.MD[1:0] = 00b) and halfword-sized transfer (MRA.SZ[1:0] = 01b).
3. In the MRB register, select incrementation of the destination address (MRB.DM = 10b) and set up chain transfer (MRB.CHNE = 1 and MRB.CHNS = 0).
4. Set the SAR register to the address of the ADC10.ADCR register.

14.6 DTC 使用示例

14.6.1正常转移

本节提供了一个 DTC 使用示例及其在连续捕获 40 次A/D 转换结果时的应用。

(1) 传输信息设置

在 MRA 寄存器中，选择固定源地址(MRA.SM[1:0] = 00b)、正常传输模式(MRA.MD[1:0] = 00b和半字大小的传输(M RA.SZ[1:0] = 01b)。在 MRB 寄存器中，指定目标地址的递增方式(MRB.DM[1:0] = 10b) 和单次中断单次数据传输(M RB.CHNE = 0，以及 MRB.DISEL= 0)。MRB.DTS 位可以设置为任意值。在 SAR 寄存器中设置 ADC10 的 ADCR 寄存器地址，在 DAR 寄存器中设置用于数据存储的 SRAM 区域的起始地址，并在 CRA 寄存器中设置 40 (0x0028)。CRB 寄存器可以设置为任意值。

(2) DTC矢量表设置

ADC10_ADI 中断的传输信息的起始地址在 DTC 的向量表中设置。

(3) ICU 设置和 DTC 模块激活

将 ICU.DTCENST0.ST31 位设置为 1。必须在 NVIC 中启用中断。将 DTCST.DTCST 位设置为 1。

(4) ADC10 设置

正确设置 ADC10.ADUL、ADC10.ADLL 和 ADC10.ADM2 寄存器以启用 ADC10_ADI。如果未启用 ADC10_ADI，则 DTC 不会启动传输。有关 ADC10 设置的详细信息，请参阅第 22.6 节“A/D 转换器设置步骤”。

(5) DTC转账

每次ADC10完成A/D转换时，都会产生一个ADC10_ADI中断来激活DTC。DTC将ADC10的ADCR中的A/D转换结果传输到SRAM，之后DAR寄存器递增，CRA寄存器递减。

(6) 中断处理

完成 40 轮数据传输后，CRA 寄存器中的值变为 0，此时会向 CPU 生成 ADC10_ADI 中断请求。请在处理此中断的例程中完成后续处理。

14.6.2 链式转移

本节提供了一个DTC链式传输的示例，并描述了其在连续捕获A/D转换结果并通过SAU的UART0发送时的用法。您可以使用链式传输将A/D转换结果传输到SRAM区域，并通过SAU的UART0发送。

对于链式传输中的第一次传输，指定了从 ADC10.ADCR 寄存器到 SRAM 区域的正常传输模式。对于第二次传输，指定了从 SRAM 区域中的 A/D 转换结果到 SAU0.SDR00 寄存器的正常传输模式。这是因为清除激活源并在完成指定次数的传输后生成中断的操作仅限于链式传输中的第二次传输，即在MRB.CHNE = 0期间进行传输。

以下示例展示了如何使用 A/D 转换结束中断作为 DTC 的激活源。

(1) 首次传输信息设置

将 ADC10.ADCR 寄存器传输到 SRAM 区域。

1. 在 MRA 寄存器中，选择源地址为固定地址(MRA.SM[1:0] = 00b)。
2. 将传输设置为正常传输模式（MRA.MD[1:0] = 00b) 和半字大小的传输(MRA.SZ[1:0] = 01b)）。
3. 在 MRB 寄存器中，选择目标地址(MRB.DM = 10b) 的递增，并设置链式转账(MRB.CHNE = 1和 MRBCHNS = 0)。
4. 将 SAR 寄存器设置为 ADC10.ADCR 寄存器的地址。

- Set the DAR register to the address of the data table in the SRAM area where to store the A/D conversion result.
- Set the CRA registers to the size of the data table. The CRB register can be set to any value.

(2) Second transfer information setting

Set up for transfer to the SAU0.SDR00 register.

- In the MRA register, select incrementation of the source address (MRA.SM[1:0] = 10b).
- Set the transfer to normal transfer mode (MRA.MD[1:0] = 00b) and halfword-sized transfer (MRA.SZ[1:0] = 01b).
- In the MRB register, select the destination address as fixed (MRB.DM[1:0] = 00b) and set up single data transfer per interrupt(MRB.CHNE = 0, MRB.DISEL = 0).
- Set the SAR register to the first address of the data table.
- Set the DAR register to the address of the SAU0.SDR00 register.
- Set the CRA registers to the size of the data table. The CRB register can be set to any value.

(3) Transfer information assignment

Place the transfer information for use in the transfer to the SAU0.SDR00 immediately after the transfer control information for use in the ADC10.ADCR registers.

(4) DTC vector table

In the DTC vector table, set the address where the transfer control information for use in transfer to the ADC10.ADCR register starts.

(5) ICU setting and DTC module activation

- Set the ICU.DTCENST0.ST31 bit associated with the A/D conversion end interrupt.
- Set the DTCST.DTCST bit to 1.

(6) ADC10 settings

Set the ADC10.ADUL, the ADC10.ADLL, and the ADC10.ADM2 register appropriately to enable ADC10_ADI. If ADC10_ADI does not occur, the DTC do not start the transfer. For details of setting the ADC10, see [section 22.6. A/D Converter Setup Procedure](#).

(7) SAU settings

See [Figure 19.31](#) for the detailed settings of UART using the SAU.

(8) DTC transfer

Each time an A/D conversion end interrupt is generated, the value of the ADCR register is transferred to the SRAM area and the SAU0.SDR00 register. When the SAU0.SDR00 register is written, the transmission automatically starts.

(9) Interrupt handling

After the specified rounds of data transfer are complete, for example when the value in the CRA register for the ADC10 transfer becomes 0, an A/D conversion end interrupt is issued for the CPU. Complete the process for this interrupt in the handling routine.

14.6.3 Chain Transfer when Counter = 0

The second data transfer is performed only when the transfer counter is set to 0 in the first data transfer, and the first data transfer information is repeatedly changed in the second transfer. Chain transfer enables transfers to be repeated 256 times or more.

The following procedure shows an example of configuring a 1-KB input buffer, where the input buffer is set so that its lower address starts with 0x00. [Figure 14.13](#) shows a chain transfer when the counter = 0.

- Set the normal transfer mode to input data for the first data transfer. Set the following:
 - Transfer source address = fixed.

- 将 DAR 寄存器设置为 SRAM 区域中用于存储A/D转换结果的数据表的地址。
- 将 CRA 寄存器设置为数据表的大小。CRB 寄存器可以设置为任意值。

(2) 二次传输信息设置

设置为传输到 SAU0.SDR00 寄存器。

- 在 MRA 寄存器中, 选择源地址(MRA.SM[1:0] = 10b)的递增。
- 将传输设置为正常传输模式(MRA.MD[1:0] = 00b) 和半字大小传输(MRA.SZ[1:0] = 01b)。
- 在 MRB 寄存器中, 选择目标地址为固定地址 (MRB.DM[1:0] = 00b)), 并设置每次中断进行一次数据传输 ((MRB.CHNE = 0, MRB.DISEL = 0)) 。
- 将 SAR 寄存器设置为数据表的第一个地址。
- 将 DAR 寄存器设置为 SAU0.SDR00 寄存器的地址。
- 将 CRA 寄存器设置为数据表的大小。CRB 寄存器可以设置为任意值。

(3) 转移信息分配

将传输信息用于传输到 SAU0.SDR00, 紧随传输控制信息用于 ADC10.ADCR 寄存器之后。

(4) DTC矢量表

在 DTC 向量表中, 设置传输控制信息用于传输到 ADC10.ADCR 寄存器的起始地址。

(5) ICU 设置和 DTC 模块激活

- 设置与 A/D 转换结束中断关联的 ICU.DTCENST0.ST31 位。
- 将 DTCST.DTCST 位设置为 1。

(6) ADC10 设置

正确设置 ADC10.ADUL、ADC10.ADLL 和 ADC10.ADM2 寄存器以启用 ADC10_ADI。如果未发生 ADC10_ADI, 则 DTC 不会启动传输。有关设置 ADC10 的详细信息, 请参阅第 22.6 节“A/D 转换器设置步骤”。

(7) SAU 设置

有关使用 SAU 的 UART 的详细设置, 请参见图 19.31。

(8) DTC转移

每次产生A/D 转换结束中断时, ADCR寄存器的值会被传输到SRAM区域和SAU0.SDR00寄存器。当SAU0.SDR00寄存器被写入时, 传输会自动开始。

(9) 中断处理

例如, 当指定的数据传输轮次完成后, 当 ADC10 传输的 CRA 寄存器值变为 0 时, 会向 CPU 发出A/D转换结束中断。在处理例程中完成此中断的处理。

14.6.3当计数器= 0时链式转移

仅当第一次数据传输中的传输计数器被设置为 0, 且第二次数据传输中重复修改了第一次数据传输的信息时, 才会执行第二次数据传输。链式传输允许传输重复 256 次或更多次。

以下步骤展示了如何配置一个 1 KB 的输入缓冲区, 其中输入缓冲区的低地址以0x00. 开头。图 14.13 展示了当计数器= 0为时发生的链式传输。

- 将首次数据传输的正常传输模式设置为输入数据。设置如下:
 - 转移源地址=固定。

- (b) CRA register = 0x0200 (512) times.
 - (c) MRB.CHNE bit = 1 (chain transfer is enabled).
 - (d) MRB.CHNS bit = 1 (chain transfer is performed only when the transfer counter is 0).
 - (e) MRB.DISEL bit = 0 (an interrupt request to the CPU is generated when the specified data transfer completes).
2. Prepare the upper 8-bit address of the start address at every 512 times of the transfer destination address for the first data transfer in different area such as the flash. For example, when setting the input buffer to 0x8000 to 0x83FF, prepare 0x82 and 0x80.
3. For the second data transfer:
- (a) Set the repeat transfer mode (with transfer source and destination address = fixed.) to reset the transfer counter of the first data transfer.
 - (b) Specify the CRA register in the first transfer information area for the transfer destination.
 - (c) Set the MRB.CHNE bit = 1 (chain transfer is enabled).
 - (d) Set the MRB.CHNS bit = 0 (select continuous chain transfer).
 - (e) Set the MRB.DISEL bit = 0 (an interrupt request to the CPU is generated when the specified data transfer completes).
 - (f) CRA register = 0x0101 (The transfer count is 1).
4. For the third data transfer:
- (a) Set the repeat transfer mode (with the source as the repeat area) to reset the transfer destination address of the first data transfer.
 - (b) Specify the upper 8 bits of the DAR register in the first transfer information area for the transfer destination.
 - (c) Set the MRB.CHNE bit = 0 (chain transfer is disabled).
 - (d) Set the MRB.DISEL bit = 0 (an interrupt request to the CPU is generated when the specified data transfer completes).
 - (e) When setting the input buffer to 0x8000 to 0x83FF, also set the transfer counter to 2.
5. The first data transfer is performed by an interrupt 512 times. When the transfer counter of the first data transfer becomes 0, the second data transfer starts. Set the transfer counter of the first data transfer to 0x0200. The lower 8 bits of the transfer destination address and the transfer counter of the first data transfer becomes 0x0200.
6. The second data transfer is performed by an interrupt 1 time. When the transfer counter of the first data transfer becomes 0, the third data transfer starts. Set the upper 8 bits of the transfer destination address of the first data transfer to 0x82. The lower 8 bits of the transfer destination address becomes 0x00 and the transfer counter of the first data transfer becomes 0x0200.
7. In succession, the first data transfer is performed by an interrupt 512 times as specified for the first data transfer. When the transfer counter of the first data transfer becomes 0, the second data transfer starts. Set the transfer counter of the first data transfer to 0x0200. The lower 8 bits of the transfer destination address and the transfer counter of the first data transfer becomes 0x0200.
8. The second data transfer is performed by an interrupt 1 time. When the transfer counter of the first data transfer becomes 0, the third data transfer starts. Set the upper 8 bits of the transfer destination address of the first data transfer to 0x80. The lower 8 bits of the transfer destination address becomes 0x00 and the transfer counter of the first data transfer becomes 0x0200.
9. Steps 5 to 8 are repeated indefinitely. Because the second data transfer is in repeat transfer mode, no interrupt request to the CPU is generated.

- (b) CRA 注册= 0x0200 (512) 次。
 - (c) MRB.CHNE 位= 1 (链式传输已启用)。
 - (d) MRB.CHNS 位= 1 (仅当传输计数器为 0 时才执行链传输)。
 - (e) MRB.DISEL 位= 0 (当指定的数据传输完成时, 会生成向 CPU 发出的中断请求)。
2. 准备每隔 512 个传输目标地址的起始地址的高 8 位地址, 用于在不同区域 (例如闪存) 进行首次数据传输。例如, 当设置输入缓冲区0x8000到0x83FF, 时, 准备 0x82 和0x80。
- 3. 第二次数据传输:**
- (a) 设置重复传输模式 (传输源地址和目标地址=固定。)以重置第一次数据传输的传输计数器。
 - (b) 在转账目的地的第一个转账信息区域中指定 CRA 注册表。
 - (c) 设置 MRB.CHNE 位= 1 (链式传输已启用)。
 - (d) 设置 MRB.CHNS 位= 0 (选择连续链传输)。
 - (e) 设置 MRB.DISELbit = 0 (当指定的数据传输完成时, 会生成一个向 CPU 发出的中断请求)。
 - (f) CRA 登记= 0x0101 (转账计数为 1)。
- 4. 第三次数据传输:**
- (a) 设置重复传输模式 (以源作为重复区域) 以重置第一次数据传输的传输目标地址。
 - (b) 指定传输目的地的第一个传输信息区域中 DAR 寄存器的高 8 位。
 - (c) 设置 MRB.CHNE 位= 0 (链式传输已禁用)。
 - (d) 设置 MRB.DISEL 位= 0 (当指定的数据传输完成时, 会生成向 CPU 发出的中断请求)。
 - (e) 当将输入缓冲区设置为0x8000到0x83FF, 时, 也应将传输计数器设置为 2。
5. 第一次数据传输通过中断执行 512 次。当第一次数据传输的传输计数器变为 0 时, 第二次数据传输开始。将第一次数据传输的传输计数器设置为0x0200。传输目标地址的低 8 位和第一次数据传输的传输计数器变为0x0200。
6. 第二次数据传输通过中断 1 次执行。当第一次数据传输的传输计数器变为 0 时, 第三次数据传输开始。将第一次数据传输的传输目标地址的高 8 位设置为0x82。传输目标地址的低 8 位变为0x00, 第一次数据传输的传输计数器变为0x0200。
7. 按照第一次数据传输的指定方式, 通过中断连续执行 512 次第一次数据传输。当第一次数据传输的传输计数器变为 0 时, 第二次数据传输开始。将第一次数据传输的传输计数器设置为0x0200。传输目标地址的低 8 位和第一次数据传输的传输计数器变为0x0200。
8. 第二次数据传输通过中断 1 次执行。当第一次数据传输的传输计数器变为 0 时, 第三次数据传输开始。将第一次数据传输的传输目标地址的高 8 位设置为0x80。传输目标地址的低 8 位变为0x00, 第一次数据传输的传输计数器变为0x0200。
9. 步骤5至8无限重复。由于第二次数据传输处于重复传输模式, 因此不会向CPU发出中断请求。

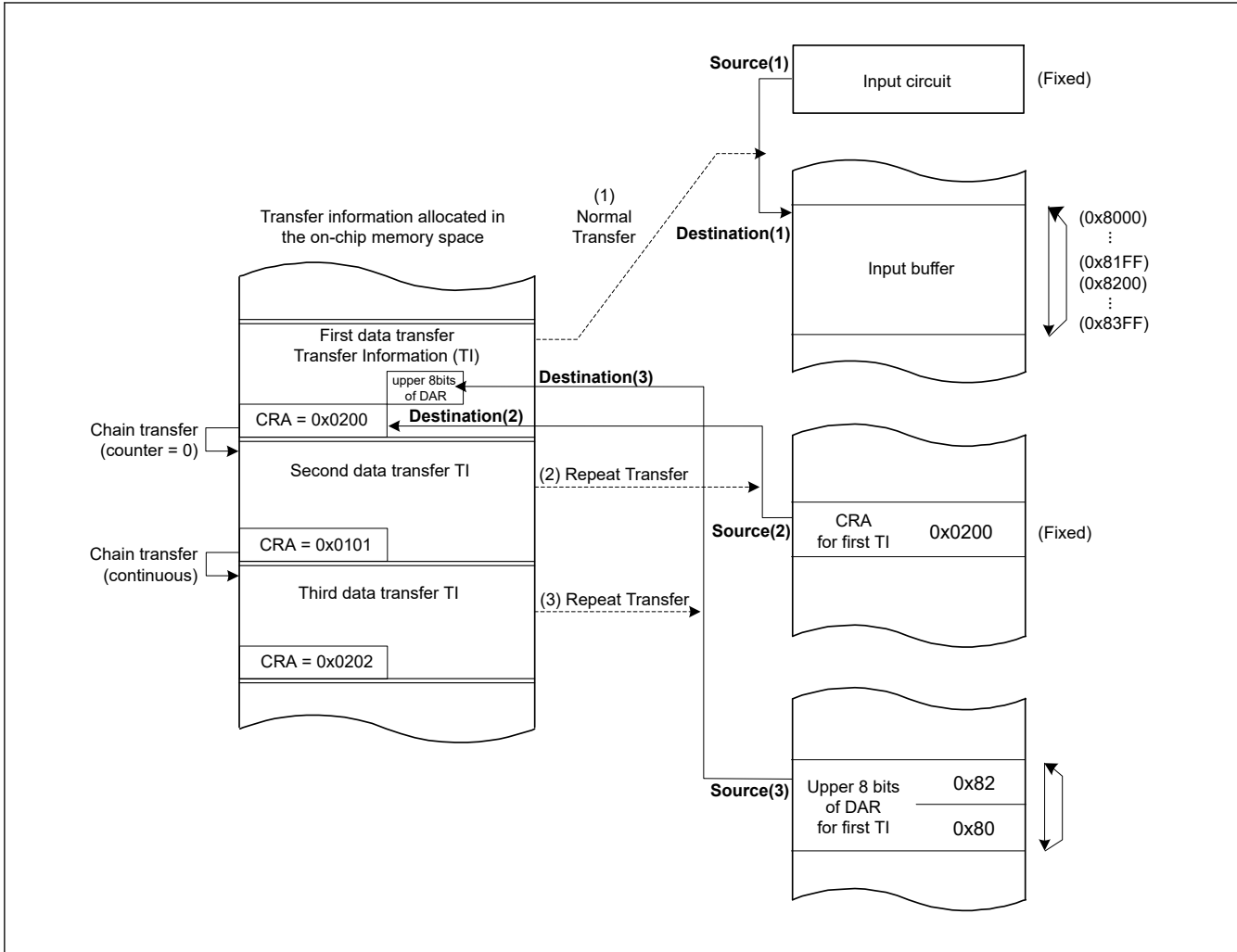


Figure 14.13 Chain transfer when counter = 0

14.7 Interrupt

14.7.1 Interrupt Sources

When the DTC completes data transfer of the specified count or when data transfer with MRB.DISEL set to 1 is complete, a DTC activation source generates an interrupt to the CPU. Two types of interrupt are available: interrupts triggered by a DTC activation (per channel) and an interrupt triggered by the event signal DTC_COMPLETE (common to all channels). Interrupts to the CPU are controlled according to the settings in the NVIC. See [section 11, Interrupt Controller Unit \(ICU\)](#). The DTC prioritizes activation sources by granting the smaller interrupt vector numbers higher priority. The priority of interrupts to the CPU is determined by the NVIC priority.

14.8 Low Power Consumption Function

Before transitioning to the module-stop state, or Software Standby mode without Snooze mode transition, set the DTCST.DTCST bit to 0, and then perform the operations described in the following sections.

(1) Module-Stop Function

Writing 1 to the MSTPCRA.MSTPA22 bit enables the module-stop function of the DTC. If a DTC transfer is in progress when 1 is written to the MSTPCRA.MSTPA22 bit, the transition to the module-stop state proceeds after the DTC transfer ends. While the MSTPCRA.MSTPA22 bit is 1, accessing the DTC registers is prohibited. Writing 0 to the MSTPCRA.MSTPA22 bit releases the DTC from the module-stop state.

(2) Software Standby Mode

Use the settings described in [section 9.7.1. Transition to Software Standby Mode](#).

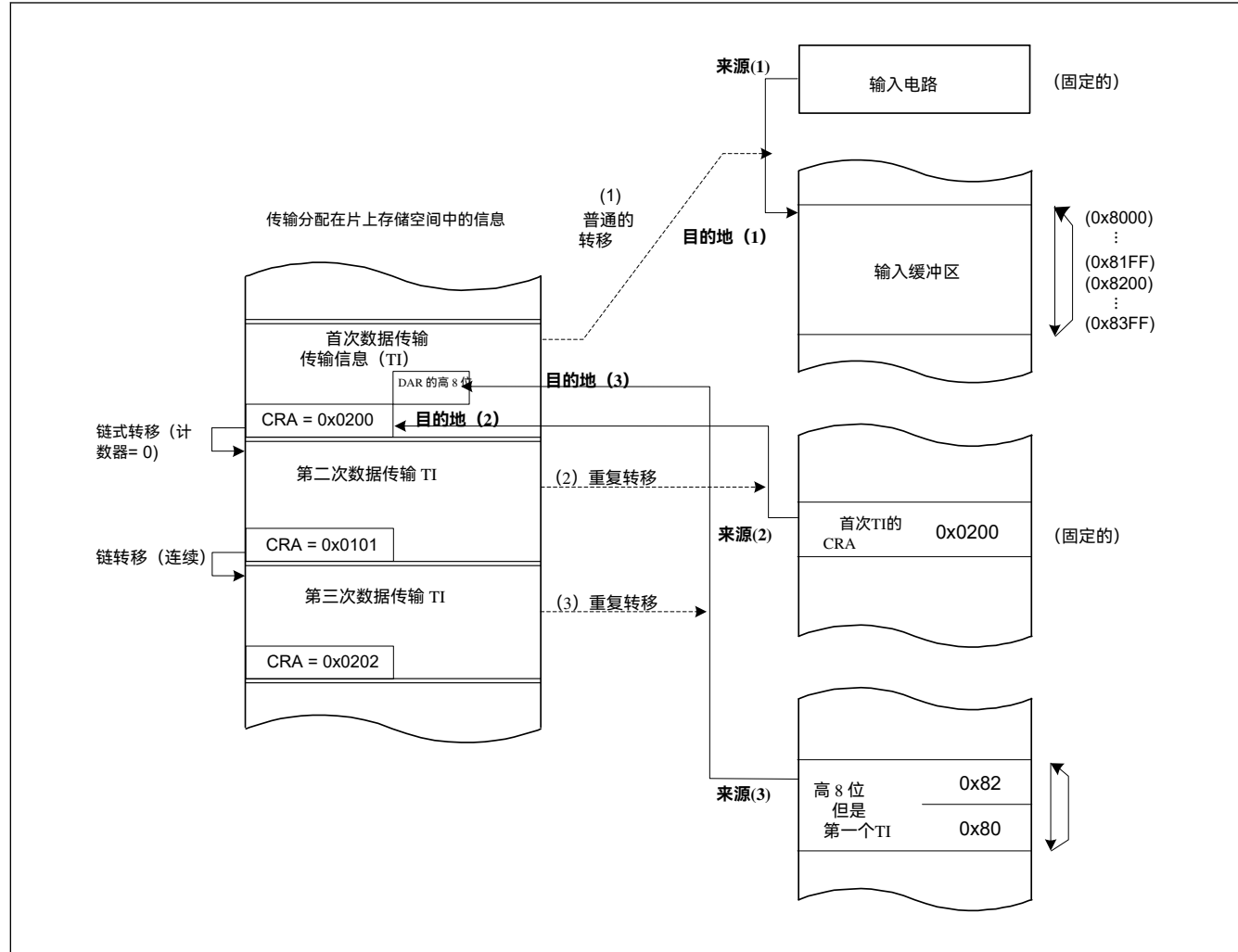


图 14.13 当计数器=0时链转移

14.7 中断

14.7.1 中断源

当DTC完成指定计数的数据传输，或MRB.DISEL设置为1的数据传输完成时，DTC激活源会向CPU发送中断。有两种类型的中断：由DTC激活触发的中断（每个通道一个）和由事件信号DTC_COMPLETE触发的中断（所有通道通用）。CPU中断的控制取决于NVIC中的设置。请参阅第11节“中断控制器单元（ICU）”。DTC 通过赋予较小的中断向量编号更高的优先级来确定激活源的优先级。CPU 中断的优先级由 NVIC 优先级决定。

14.8 低功耗功能

在切换到模块停止状态或软件待机模式（不进行休眠模式转换）之前，将 DTCST.DTCST 位设置为 0，然后执行以下各节中描述的操作。

(1) 模块停止功能

将 MSTPCRA.MSTPA22 位写入 1 可启用 DTC 的模块停止功能。如果在将 1 写入 MSTPCRA.MSTPA22 位时 DTC 传输正在进行，则在 DTC 传输结束后才会进入模块停止状态。当 MSTPCRA.MSTPA22 位为 1 时，禁止访问 DTC 寄存器。将 0 写入 MSTPCRA.MSTPA22 位可解除 DTC 的模块停止状态。

(2) 软件待机模式

使用第 9.7.1 节中描述的设置。过渡到软件待机模式。

If DTC transfer operations are in progress when the WFI instruction is executed, the transition to Software Standby mode is executed after the completion of the DTC transfer.

(3) Snooze Mode

When the snooze control circuit receives a snooze request in Software Standby mode, the MCU transitions to Snooze mode. See [section 9.8.1. Transition to Snooze Mode](#). If DTC operation is enabled in Snooze mode, before transitioning to Software Standby mode, set the DTCST.DTCST bit to 1.

Set the ICU.SBYEDCRn.xxxED bit (the bit corresponding to the interrupt that caused the DTC to activate) to 1 when transitioning to normal mode when the DTC transfer is completed.

- Note:

When a transfer end interrupt from the simplified SPI in SNOOZE mode is being used as the DTC activation source, use the transfer end interrupt to release the chip from the SNOOZE mode and start processing by the CPU after completion of DTC transfer, or use a chain transfer to make the settings for reception by the simplified SPI (writing 1 to the ST[0] bit, writing 0 to the SWC bit, setting the SSC0 register, and writing 1 to the SS[0] bit) again.
- Note:

When a transfer end interrupt from the UART in SNOOZE mode is being used as the DTC activation source, use the transfer end interrupt to release the chip from the SNOOZE mode and start processing by the CPU after completion of DTC transfer, or use a chain transfer to make the settings for reception by the UART (writing 1 to the ST[1] bit, writing 0 to the SWC bit, setting the SSC0 register, and writing 1 to the SS[1] bit) again.
- Note:

When an A/D conversion end interrupt from the A/D converter in SNOOZE mode is being used as the DTC activation source, use the A/D conversion end interrupt to release the chip from the SNOOZE mode and start processing by the CPU after completion of DTC transfer, or use a chain transfer to make the settings for the SNOOZE mode function of the A/D converter (writing 1 to the AWC bit after having written 0 to it) again.

(4) Notes on Low Power Consumption Function

For the WFI instruction and the register setting procedure, see [section 9, Low Power Modes](#).

To perform a DTC transfer after returning from a low power mode without a Snooze mode transition, set the DTCST.DTCST bit to 1 again.

To use a request that is generated in Software Standby mode as an interrupt request to the CPU but not as a DTC activation request, specify the CPU as the interrupt request destination as described in [section 11.4.1. Detecting Interrupts](#), then execute the WFI instruction. If DTC operation is enabled in Snooze mode, do not use the module-stop function of the DTC.

14.9 Usage Notes

14.9.1 Transfer Information Start Address

You must set multiples of 4 for the transfer information start addresses in the vector table. Otherwise, such addresses are accessed with their lowest 2 bits regarded as 00b.

如果在执行 WFI 指令时 DTC 传输操作正在进行中，则在 DTC 传输完成后执行向软件待机模式的转换。

(3) 贪睡模式

当睡眠控制电路在软件待机模式下收到睡眠请求时，MCU 将转换到睡眠模式。参见 9.8.1 节“转换到睡眠模式”。如果在睡眠模式下启用了 DTC 操作，则在转换到软件待机模式之前，应将 DTCST.DTCST 位设置为 1。

当 DTC 传输完成并转换到正常模式时，将 ICU.SBYEDCRn.xxxED 位（对应于导致 DTC 激活的中断的位）设置为 1。

注意：当使用 SNOOZE 模式下简化 SPI 的传输结束中断作为 DTC 激活源时，应使用传输结束中断将芯片从 SNOOZE 模式中释放出来，并在 DTC 传输完成后开始由 CPU 进行处理，或者使用链式传输再次对简化 SPI 的接收进行设置（将 1 写入 ST [0] 位，将 0 写入 SWC 位，设置 SSC0 寄存器，并将 1 写入 SS [0] 位）。

注意：当 SNOOZE 模式下 UART 的传输结束中断用作 DTC 激活源时，应使用传输结束中断将芯片从 SNOOZE 模式中释放出来，并在 DTC 传输完成后由 CPU 开始处理，或者使用链式传输再次对 UART 接收进行设置（将 1 写入 ST [1] 位，将 0 写入 S WC 位，设置 SSC0 寄存器，并将 1 写入 SS [1] 位）。

注意：当 SNOOZE 模式下 A/D 转换器的 A/D 转换结束中断用作 DTC 激活源时，应使用 A/D 转换结束中断将芯片从 S NOOZE 模式中释放出来，并在 DTC 传输完成后由 CPU 开始处理，或者使用链式传输再次设置 A/D 转换器的 SNOOZE 模式功能（在向 AWC 位写入 0 后向其写入 1）。

(4) 低功耗功能说明

有关 WFI 指令和寄存器设置过程，请参阅第 9 节“低功耗模式”。

要在从低功耗模式返回后执行 DTC 传输（无需休眠模式转换），请再次将 DTCST.DTCST 位设置为 1。

要将软件待机模式下生成的请求用作 CPU 中断请求，而不是 DTC 激活请求，请按照 11.4.1 节“检测中断”中的说明，将 CPU 指定为中断请求目标，然后执行 WFI 指令。如果 DTC 操作在休眠模式下启用，请勿使用 DTC 的模块停止功能。

14.9 使用说明

14.9.1 转账信息起始地址

您必须将向量表中传输信息起始地址设置为 4 的倍数。否则，访问此类地址时，其最低 2 位将被视为 00b。

15. I/O Ports

15.1 Overview

The I/O port pins operate as general I/O port pins, I/O pins for peripheral modules, interrupt input pins, analog I/O.

All pins operate as input pins immediately after a reset, and pin functions are switched by register settings. The I/O ports and peripheral modules for each pin are specified in the associated registers.

Figure 15.1 shows a connection diagram for the I/O port registers. The configuration of the I/O ports differs depending on the package. Table 15.1 lists the I/O port specifications by package, and Table 15.2 lists the port functions.

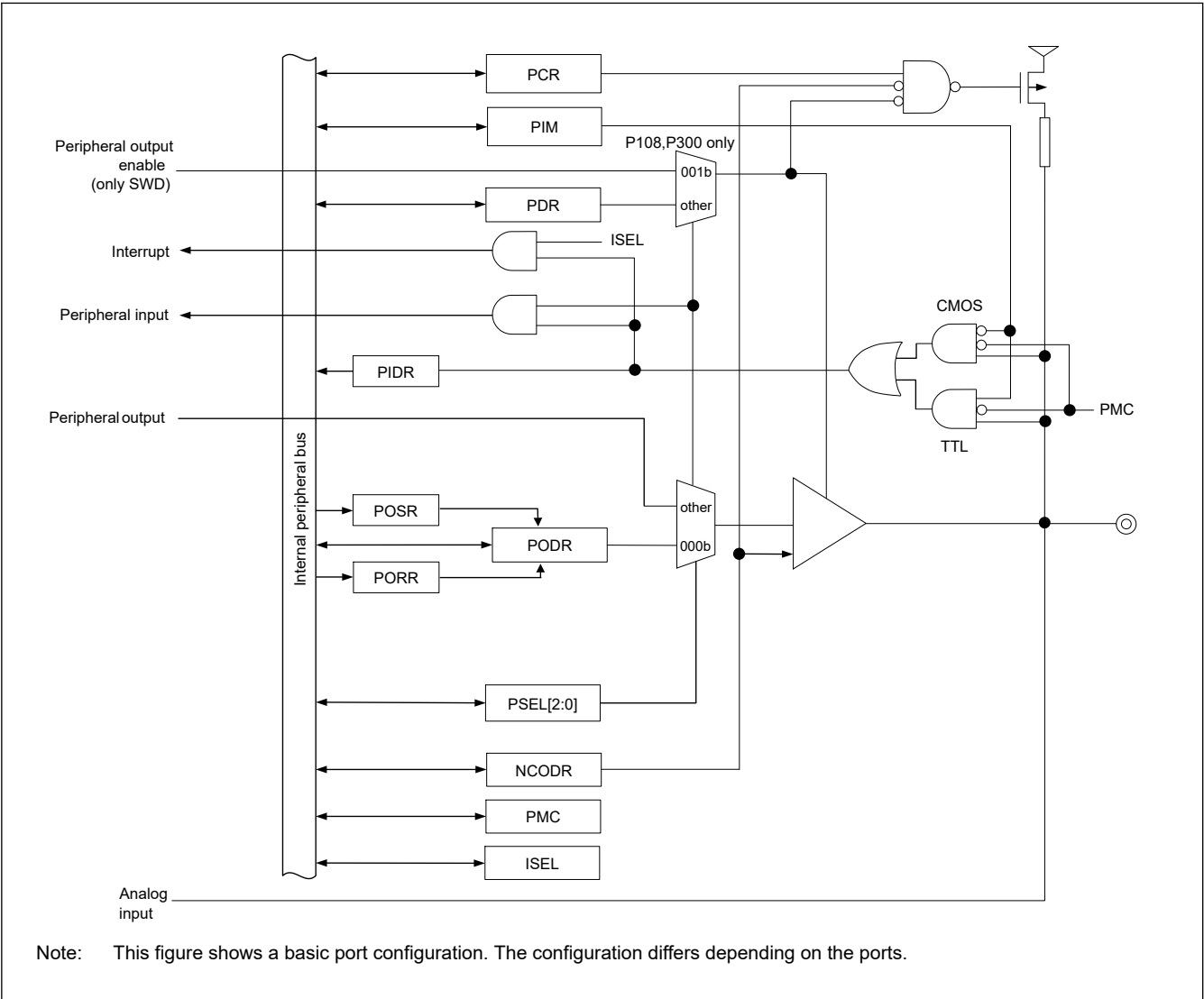


Figure 15.1 Connection diagram for I/O port registers

Table 15.1 shows the I/O port specifications and Table 15.2 shows the port functions.

Table 15.1 I/O port specifications

Port	Package	
	20 pins	Number of pins
Port 0	P010 to P013	4
Port 1	P100 to P102, P108 to P110, P112	7
Port 2	P200, P201, P206, P212, P213	5
Port 3	P300	1

15. I/O 端口

15.1 概述

I/O端口引脚用作通用I/O端口引脚、外围模块的I/O引脚、中断输入引脚、模拟I/O。

复位后所有引脚立即作为输入引脚工作，引脚功能通过寄存器设置进行切换。每个引脚的I/O端口和外设模块均在相应的寄存器中指定。

图 15.1 显示了 I/O 端口寄存器的连接图。I/O 端口的配置因封装而异。表 15.1 列出了按封装分类的I/O端口规格，表 15.2 列出了端口功能。

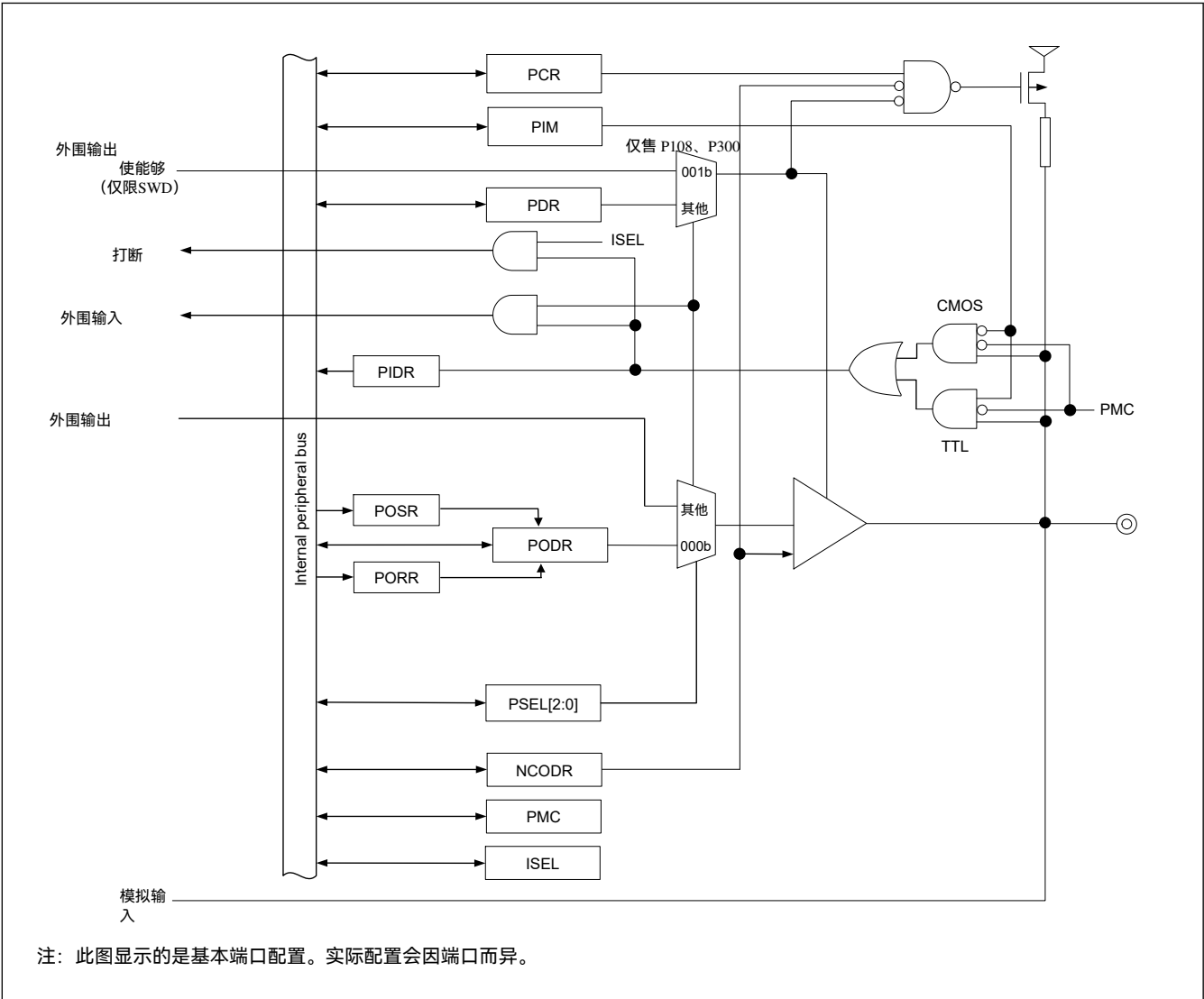


图 15.1 I/O 端口寄存器的连接图

表 15.1 显示了 I/O 端口规格，表 15.2 显示了端口功能。

表 15.1 I/O 端口规格

港口	包裹	
	20个引脚	引脚数量
端口 0	P010 至 P013	4
端口 1	P100 至 P102、P108 至 P110、P112	7
端口 2	P200、P201、P206、P212、P213	5
端口 3	P300	1

Note: —: Setting prohibited

Table 15.2 I/O port functions

Port	Port name	Input pull-up	Input mode switching	Open drain output	5V tolerant	I/O
Port 0	P010 to P013	—	—	—	—	Input/Output
Port 1	P100 to P102, P108 to P110, P112	✓	CMOS/TTL	✓	—	Input/Output
Port 2	P200	—	—	—	—	Input
	P201	✓	CMOS/TTL	✓	—	Input/Output
	P206	✓	—	—	—	Input/Output
	P212, P213	✓	—	✓	—	Input/Output
Port 3	P300	✓	CMOS/TTL	✓	—	Input/Output

Note: ✓: Available
 —: Setting prohibited

15.2 Register Descriptions

15.2.1 PODRm : Pmn Output Data Register (m = 0 to 3, n = 00 to 15)

Base address: PORTm = 0x400A_0000 + 0x20 × m (m = 0 to 3)
Offset address: 0x0000

Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	PODR 15	PODR 14	PODR 13	PODR 12	PODR 11	PODR 10	PODR 09	PODR 08	PODR 07	PODR 06	PODR 05	PODR 04	PODR 03	PODR 02	PODR 01	PODR 00
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
15:0	PODR15 to PODR00	Pmn Output Data 0: Low output 1: High output	R/W

The Pmn Output Data Register (PODRm) is a 16-bit read/write register and is accessed in 16-bit units.

Note: The existent ports in this product.

- PODR0: P0n Output Data Register (n = 10 to 13)
- PODR1: P1n Output Data Register (n = 00 to 02, 08 to 10, 12)
- PODR2: P2n Output Data Register (n = 01, 06, 12, 13)
- PODR3: P3n Output Data Register (n = 00)

PODRn bits (Pmn Output Data)

The PODRn bits hold data to be output from the general I/O pins. Bits of non-existent port m are reserved. Reserved bits are read as 0. The write value should be 0. P200 is input only, so PODR2.PODR00 bit is reserved. The PODRn bit in the PODRm register serves the same function as the PODR bit in the PmnPFS_A register.

Note: When the RES pin (OFS1.PORTSELB = 1) is selected, the PODR2.PODR06 bit (P206) is always read as 0.

注意： 一： 禁止设置

表 15.2 I/O 端口功能

港口	端口名称	输入上拉	输入模式切换	开漏输出	耐5V电压	I/O
端口 0	P010 至 P013	—	—	—	—	输入/输出
端口 1	P100 至 P102、P108 至 P110、P112	✓	CMOS/TTL	✓	—	输入/输出
端口 2	P200	—	—	—	—	输入
	P201	✓	CMOS/TTL	✓	—	输入/输出
	P206	✓	—	—	—	输入/输出
	P212, P213	✓	—	✓	—	输入/输出
端口 3	P300	✓	CMOS/TTL	✓	—	输入/输出

注意：： 有货
 —: 设置禁止

15.2 寄存器描述

15.2.1PODRm：PMN 输出数据寄存器(m = 0至 3， n = 00至 15)

基地址: PORTm = 0x400A_0000 + 0x20 × m (m = 0至 3)
偏移地址: 0x0000

位位置:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域:	PODR 15	PODR 14	PODR 13	PODR 12	PODR 11	PODR 10	PODR 09	PODR 08	PODR 07	PODR 06	PODR 05	PODR 04	PODR 03	PODR 02	PODR 01	PODR 00
重置后的值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
15:0	PODR15 至 PODR00	PMN 输出数据 0: 低输出 1: 高输出	R/W

Pmn输出数据寄存器 (PODRm) 是一个 16 位读/写寄存器， 以 16 位为单位进行访问。

注意： 本产品中已存在的端口。

- PODR0: P0n 输出数据寄存器(n = 10 至 13)
- PODR1: P1n 输出数据寄存器(n = 00 至 02、08 至 10、12)
- PODR2: P2n 输出数据寄存器(n = 01, 06, 12, 13)
- PODR3: P3n 输出数据寄存器(n = 00)

PODRn 位(Pmn输出数据)

PODRn 位保存要从通用 I/O 引脚输出的数据。不存在的端口 m 的位被保留。保留位读取为 0。写入值应为 0。P200 仅用于输入， 因此 PODR2.PODR00 位被保留。PODRm 寄存器中的 PODRn 位与 PmnPFS_A 寄存器中的 PODR 位功能相同。

注意： 当选择 RES 引脚（OFS1.PORTSELB= 1） 时， PODR2.PODR06 位（P206） 始终读取为 0。

15.2.2 PDRm : Pmn Direction Register (m = 0 to 3, n = 00 to 15)

Base address:	PORTm = 0x400A_0000 + 0x20 × m (m = 0 to 3)															
Offset address:	0x0002															
Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	PDR1 5	PDR1 4	PDR1 3	PDR1 2	PDR11	PDR1 0	PDR0 9	PDR0 8	PDR0 7	PDR0 6	PDR0 5	PDR0 4	PDR0 3	PDR0 2	PDR0 1	PDR0 0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
15:0	PDR15 to PDR00	Pmn Direction 0: Input (functions as an input pin) 1: Output (functions as an output pin)	R/W

The Pmn Direction Register (PDRm) is a 16-bit read/write register and is accessed in 16-bit units.

- Note: The existent ports in this product.
- PDR0: P0n Direction Register (n = 10 to 13)
 - PDR1: P1n Direction Register (n = 00 to 02, 08 to 10, 12)
 - PDR2: P2n Direction Register (n = 01, 06, 12, 13)
 - PDR3: P3n Direction Register (n = 00)

PDRn bits (Pmn Direction)

The PDRn bits select the input or output direction for individual pins on the associated port when the pins are configured as general I/O pins. Each pin on port m is associated with a PDRm.PDRn bit. The I/O direction can be specified in 1-bit units. Bits associated with non-existent pins are reserved. Reserved bits are read as 0. The write value should be 0.

P200 is input only, so PDR2.PDR00 bit is reserved. The PDRn bit in the PDRm register serves the same function as the PDR bit in the PmnPFS_A register.

Note: When the RES pin (OFS1.PORTSELB = 1) is selected, the PDR2.PDR06 bit (P206) is always read as 0.

15.2.3 PIDRm : Pmn State Register (m = 0 to 3, n = 00 to 15)

Base address:	PORTm = 0x400A_0000 + 0x20 × m (m = 0 to 3)															
Offset address:	0x0006															
Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	PIDR1 5	PIDR1 4	PIDR1 3	PIDR1 2	PIDR1 1	PIDR1 0	PIDR0 9	PIDR0 8	PIDR0 7	PIDR0 6	PIDR0 5	PIDR0 4	PIDR0 3	PIDR0 2	PIDR0 1	PIDR0 0
Value after reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Bit	Symbol	Function	R/W
15:0	PIDR15 to PIDR00	Pmn State 0: Low level 1: High level	R

The Pmn State Register (PIDRm) is a 16-bit write register and is accessed in 16-bit units.

- Note: The existent ports in this product.
- PIDR0: P0n State Register (n = 10 to 13)
 - PIDR1: P1n State Register (n = 00 to 02, 08 to 10, 12)
 - PIDR2: P2n State Register (n = 00, 01, 06, 12, 13)
 - PIDR3: P3n State Register (n = 00)

15.2.2 PDRm: Pmn 方向寄存器(m = 0至 3, n = 00至 15)

基本地址:	PORTm = 0x400A_0000 + 0x20 × m (m = 0至 3)															
偏移地址:	0x0002															
位位置:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域:	PDR1 5	PDR1 4	PDR1 3	PDR1 2	PDR11	PDR1 0	PDR0 9	PDR0 8	PDR0 7	PDR0 6	PDR0 5	PDR0 4	PDR0 3	PDR0 2	PDR0 1	PDR0 0
重置后的值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
15:0	PDR15 至 PDR00	PMN方向 0: 输入（用作输入引脚） 1: 输出（用作输出引脚）	R/W

Pmn方向寄存器 (PDRm) 是一个 16 位读/写寄存器，以 16 位为单位进行访问。

- 注意: 本产品中已存在的端口。
- PDR0: P0n 方向寄存器(n = 10 至 13)
 - PDR1: P1n 方向寄存器(n = 00 至 02、08 至 10、12)
 - PDR2: P2n 方向寄存器(n = 01, 06, 12, 13)
 - PDR3: P3n 方向寄存器(n = 00)

PDRn 位(Pmn方向)

当引脚配置为通用 I/O 引脚时，PDRn 位用于选择关联端口上各个引脚的输入或输出方向。端口 m 上的每个引脚都与一个 PDRm.PDRn 位关联。I/O 方向可以以 1 位为单位指定。与不存在的引脚关联的位会被保留。保留位读取为 0。写入值应为 0。

P200 仅供输入，因此 PDR2.PDR00 位被保留。PDRm 寄存器中的 PDRn 位与 PmnPFS_A 寄存器中的 PDR 位功能相同。

注意: 当选择 RES 引脚（OFS1.PORTSELB= 1）时，PDR2.PDR06 位（P206）始终读取为 0。

15.2.3 PIDRm: Pmn 状态寄存器(m = 0至 3, n = 00至 15)

Base address:	PORTm = 0x400A_0000 + 0x20 × m (m = 0 to 3)															
Offset address:	0x0006															
位位置:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域:	PIDR1 5	PIDR1 4	PIDR1 3	PIDR1 2	PIDR1 1	PIDR1 0	PIDR0 9	PIDR0 8	PIDR0 7	PIDR0 6	PIDR0 5	PIDR0 4	PIDR0 3	PIDR0 2	PIDR0 1	PIDR0 0
重置后的值:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

少量	象征	功能	R/W
15:0	PIDR15 至 PIDR00	巴基斯坦国家 0: 低级别 1: 高水平	R

Pmn状态寄存器 (PIDRm) 是一个 16 位写寄存器，以 16 位为单位进行访问。

- 注意: 本产品中已存在的端口。
- PIDR0: P0n 状态寄存器(n = 10 至 13)
 - PIDR1: P1n 状态寄存器(n = 00 至 02、08 至 10、12)
 - PIDR2: P2n 状态寄存器(n = 00, 01, 06, 12, 13)
 - PIDR3: P3n 状态寄存器(n = 00)

PIDRn bits (Pmn State)

The PIDRn bits reflect the individual pin states of the port, regardless of the values set in PDRm.PDRn. The PIDRn bit in the PIDRm register serves the same function as the PIDR bit in the PmnPFS_A register.

A pin state cannot be reflected in PIDRn when one of the following functions is enabled:

- Analog function (PmnPFS_A.PMC = 1)
- Input to the input buffer is disabled (PmnPFS_A.PMC = 1)

Note: When the RES pin (OFS1.PORTSELB = 1) is selected, the PIDR2.PIDR06 bit (P206) is always read as 1.

15.2.4 PORRm : Pmn Output Reset Register (m = 0 to 3, n = 00 to 15)

Base address:	PORTm = 0x400A_0000 + 0x20 × m (m = 0 to 3)															
Offset address:	0x0008															
Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	PORR 15	PORR 14	PORR 13	PORR 12	PORR 11	PORR 10	PORR 09	PORR 08	PORR 07	PORR 06	PORR 05	PORR 04	PORR 03	PORR 02	PORR 01	PORR 00
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
15:0	PORR15 to PORR00	Pmn Output Reset 0: No effect on output 1: Low output	W

The Pmn Output Reset Register (PORRm) is a 16-bit write register and is accessed in 16-bit units.

Note: The existent ports in this product.

- PORR0: P0n Output Reset Register (n = 10 to 13)
- PORR1: P1n Output Reset Register (n = 00 to 02, 08 to 10, 12)
- PORR2: P2n Output Reset Register (n = 01, 06, 12, 13)
- PORR3: P3n Output Reset Register (n = 00)

PORRn bits (Pmn Output Reset)

PORRn changes PODRn when reset by a software write. For example, for P100, when PORR1.PORR00 = 1, PODR1.PODR00 outputs 0. Bits associated with non-existent pins are reserved. The write value should always be 0.

P200 is input only, so PORR2.PORR00 bit is reserved.

15.2.5 POSRm : Pmn Output Set Register (m = 0 to 3, n = 00 to 15)

Base address:	PORTm = 0x400A_0000 + 0x20 × m (m = 0 to 3)															
Offset address:	0x000A															
Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	POSR 15	POSR 14	POSR 13	POSR 12	POSR 11	POSR 10	POSR 09	POSR 08	POSR 07	POSR 06	POSR 05	POSR 04	POSR 03	POSR 02	POSR 01	POSR 00
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
15:0	POSR15 to POSR00	Pmn Output Set 0: No effect on output 1: High output	W

The Pmn Output Set Register (POSRm) is a 16-bit write register and is accessed in 16-bit units.

Note: The existent ports in this product.

PIDRn 位 (PMN 状态)

无论 PDRm.PDRn 中设置的值如何，PIDRn 位都反映端口的各个引脚状态。PIDRm 寄存器中的 PIDRn 位与 PmnPFS_A 寄存器中的 PIDR 位功能相同。

当启用下列功能之一时，引脚状态无法反映在 PIDRn 中：

- 模拟函数(PmnPFS_A.PMC = 1)
- 输入缓冲区已禁用(PmnPFS_A.PMC = 1)

注意：当选择 RES 引脚 (OFS1.PORTSELB= 1) 时，PIDR2.PIDR06 位 (P206) 始终读取为 1。

15.2.4 PORRm: Pmn 输出复位寄存器(m = 0至 3， n = 00至 15)

基本地址:	采购订单RTm = 0x400A_0000 + 0x20 × m (m = 0 至3)															
偏移地址:	0x0008															
位位置:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域:	PORR 15	PORR 14	PORR 13	PORR 12	PORR 11	PORR 10	PORR 09	PORR 08	PORR 07	PORR 06	PORR 05	PORR 04	PORR 03	PORR 02	PORR 01	PORR 00
重置后的值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
15:0	PORR15 至 PORR00	PMN 输出复位 0: 对输出无影响 1: 低输出	W

Pmn输出复位寄存器 (PORRm) 是一个 16 位写寄存器，以 16 位为单位进行访问。

注意：本产品中已存在的端口。

- PORR0: P0n 输出复位寄存器(n = 10至 13)
- PORR1: P1n 输出复位寄存器(n = 00至 02、08 至 10、12)
- PORR2: P2n 输出复位寄存器(n = 01, 06, 12, 13)
- PORR3: P3n 输出复位寄存器(n = 00)

PORRn 位 (PMN 输出复位)

当软件写入复位时，PORRn 会改变 PODRn 的值。例如，对于 P100，当 PORR1.PORR00 = 1时，PODR1.PODR0 0 输出 0。与不存在的引脚关联的位被保留。写入值应始终为0。

P200 仅供输入，因此 PORR2.PORR00 位保留。

15.2.5POSRm： PMN 输出设置寄存器(m = 0设置为 3， n = 00设置为 15)

基本地址:	PORTm = 0x400A_0000 + 0x20 × m (m = 0 to 3)															
偏移地址:	0x000A															
位位置:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域:	POSR 15	POSR 14	POSR 13	POSR 12	POSR 11	POSR 10	POSR 09	POSR 08	POSR 07	POSR 06	POSR 05	POSR 04	POSR 03	POSR 02	POSR 01	POSR 00
重置后的值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
15:0	POSR15 至 POSR00	PMN 输出集 0: 对输出无影响 1: 高输出	W

Pmn输出集寄存器 (POSRm) 是一个 16 位写寄存器，以 16 位为单位进行访问。

注意：本产品中已存在的端口。

- POSR0: P0n Output Set Register (n = 10 to 13)
- POSR1: P1n Output Set Register (n = 00 to 02, 08 to 10, 12)
- POSR2: P2n Output Set Register (n = 01, 06, 12, 13)
- POSR3: P3n Output Set Register (n = 00)

POSRn bits (Pmn Output Set)

POSRn changes PODRn when set by a software write. For example, for P100, when POSR1.POSR00 = 1, PODR1.PODR00 outputs 1. Bits associated with non-existent pins are reserved. The write value should always be 0. P200 is input only, so POSR2.POSR00 bit is reserved.

15.2.6 PmnPFS_A : Port mn Pin Function Select Register (m = 1 to 3, n = 00 to 15)

Base address: PFS_A = 0x400A_0200															
Offset address: 0x0000 + 0x20 × m + 0x2 × n															
Bit position: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0															
Bit field: PMC ISEL — — — PSEL[2:0] — NCOD R PIM PCR — PDR PIDR PODR															
Value after reset: 0*1 0 0 0 0 0 0 0 0 0 0 0*1 0 0 0 x 0															

Bit	Symbol	Function	R/W
0	PODR	Pmn Output Data 0: Low output 1: High output	R/W
1	PIDR	Pmn State 0: Low level 1: High level	R
2	PDR	Pmn Direction 0: Input (functions as an input pin) 1: Output (functions as an output pin)	R/W
3	—	This bit is read as 0. The write value should be 0.	R/W
4	PCR	Pull-up Control 0: Disable input pull-up 1: Enable input pull-up	R/W
5	PIM	Pin Input Buffer Selection 0: Normal input buffer 1: TTL input buffer	R/W
6	NCODR	N-channel Open-drain Control 0: CMOS output 1: NMOS open-drain output	R/W
7	—	This bit is read as 0. The write value should be 0.	R/W
10:8	PSEL[2:0]	Peripheral Select These bits select the peripheral function. For individual pin functions, see the associated tables in this chapter.	R/W
13:11	—	These bits are read as 0. The write value should be 0.	R/W
14	ISEL	IRQ Input Enable 0: Not used as an IRQn input pin 1: Used as an IRQn input pin	R/W
15	PMC	Pin Mode Control 0: Digital I/O 1: Analog input function. Input to the input buffer is disabled	R/W

Note 1. The initial value of P100, P101, P108, P206, and P300 is not 0x0000. The initial value of P100 is 0x8000, P101 is 0x8000, P108 is 0x0110, P206 is 0x0010, and P300 is 0x0110.

Port mn Pin Function Select Register (PmnPFS_A) is a 16-bit read/write control register that selects the port mn pin function, and is accessed in 16-bit units.

- POSR0: P0n 输出集寄存器(n = 10 至 13)
- POSR1: P1n 输出设置寄存器(n = 00 至 02、08 至 10、12)
- POSR2: P2n 输出集寄存器(n = 01, 06, 12, 13)
- POSR3: P3n 输出集寄存器(n = 00)

POSRN 位 (PMN 输出集)

当通过软件写入设置 POSRn 时, POSRn 会更改 PODRn。例如, 对于 P100, 当 POSR1.POSR00 = 1时, PODR1.PODR00 输出 1。与不存在的引脚关联的位被保留。写入值应始终为 0。

P200 仅供输入, 因此 POSR2.POSR00 位被保留。

15.2.6 PmnPFS_A: 端口 mn 引脚功能选择寄存器(m = 1 至 3, n = 00至 15)

Base address: PFS_A = 0x400A_0200															
Offset address: 0x0000 + 0x20 × m + 0x2 × n															
位位置: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0															
位域: PMC ISEL — — — PSEL[2:0] — NCOD R PIM PCR — PDR PIDR PODR															
重置后的值: 0*1 0 0 0 0 0 0 0 0 0 0 0*1 0 0 0 x 0															

少量	象征	功能	R/W
0	PODR	PMN 输出数据 0: 低输出 1: 高输出	R/W
1	PIDR	巴基斯坦国家 0: 低级别 1: 高水平	R
2	PDR	PMN方向 0: 输入 (用作输入引脚) 1: 输出 (用作输出引脚)	R/W
3	—	该位读取值为 0。写入值应为 0。	R/W
4	PCR	拉起控制 0: 禁用输入上拉电阻 1: 启用输入上拉电阻	R/W
5	PIM	引脚输入缓冲器选择 0: 普通输入缓冲区 1: TTL输入缓冲器	R/W
6	NCODR	N沟道开漏控制 0: CMOS 输出 1: NMOS 开漏输出	R/W
7	—	该位读取值为 0。写入值应为 0。	R/W
10:8	PSEL[2:0]	外围设备选择 这些位用于选择外设功能。有关各个引脚的功能, 请参阅本章中的相关表格。	R/W
13:11	—	这些位读取为 0。写入值应为 0。	R/W
14	ISEL	IRQ 输入使能 0: 未用作 IRQn 输入引脚 1: 用作 IRQn 输入引脚	R/W
15	PMC	引脚模式控制 0: 数字 I/O 1: 模拟输入功能。输入缓冲器的输入已禁用。	R/W

注1: P100、P101、P108、P206 和 P300 的初始值不是0x0000。P100 的初始值为0x8000, P101 的初始值为0x8000, P206 的初始值为P108, P300 的初始值为0x0110, 。

端口 mn 引脚功能选择寄存器(PmnPFS_A) 是一个 16 位读/写控制寄存器, 用于选择端口 mn 引脚功能, 以 16 位为单位进行访问。

Note: The existent ports in this product:

- P1nPFS_A: Port 1n Pin Function Select Register (n = 00 to 02, 08 to 10, 12)
- P2nPFS_A: Port 2n Pin Function Select Register (n = 00, 01, 06, 12, 13)
- P3nPFS_A: Port 3n Pin Function Select Register (n = 00)

PODR bit (Pmn Output Data), PIDR bit (Pmn State), PDR bit (Pmn Direction)

The PDR, PIDR, and PODR bits serve the same function as the PDRm, PIDRm, PODRm. When these bits are read, the PDRm, PIDRm, PODRm value is read.

PCR bit (Pull-up Control)

The PCR bit enables or disables an input pull-up resistor on the individual port pins. When a pin is in the input state with the associated bit in PmnPFS_A.PCR set to 1, the pull-up resistor connected to the pin is enabled. When a pin is set as a general port output pin, the pull-up resistor for the pin is disabled regardless of the PCR setting. The pull-up resistor is also disabled in the reset state. Bits associated with non-existent pins are reserved. Reserved bits are read as 0. The write value should be 0.

P200 does not have the PCR bit, so the PCR bits in these Port Pin Function Select registers are reserved.

Note: PCR can be selected when P206 (OFS1.PORTSELB = 0) is selected. When the RES pin (OFS1.PORTSELB = 1) is selected, the input pull-up resistor is enabled.

PIM bit (Pin Input Buffer Selection)

The PIM bit sets the input buffer. TTL input buffer can be selected during serial communication with an external device operating at a different voltage. Bits associated with non-existent pins are reserved. Reserved bits are read as 0. The write value should be 0.

P200, P206, P212, and P213 do not have the PIM bit, so the PIM bits in these Port Pin Function Select registers are reserved.

NCODR bit (N-channel Open-drain Control)

The NCODR bit specifies the output type for the port pins. N-ch open drain output (withstand voltage of VCC) mode can be selected during serial communication with an external device operating at a different voltage, and the SDA00, SDA11 pins during simplified I2C communication with an external device operating at the same voltage. In addition, NCODR are used in combination with PCR to specify whether to use on-chip pull-up resistors. An on-chip pull-up resistor is not connected with a bit for which N-ch open drain output (withstand voltage of VCC) mode (NCODR = 1) is set. Bits associated with non-existent pins are reserved. Reserved bits are read as 0. The write value should be 0.

P200 and P206 do not have the NCODR bit, so the NCODR bits in these Port Pin Function Select registers are reserved.

PSEL[2:0] bits (Peripheral Select)

The PSEL[2:0] bits assign the peripheral function. For details on the peripheral settings for each product, see [section 15.6. Peripheral Select Settings for Each Product](#).

Bits associated with non-existent pins are reserved. Reserved bits are read as 0. The write value should be 0.

P206 does not have the PSEL[2:0] bits, so the PSEL[2:0] bits in these Port Pin Function Select registers are reserved.

ISEL bit (IRQ Input Enable)

The ISEL bit specifies IRQ input pins. This setting can be used in combination with the peripheral functions, although an IRQn (external pin interrupt) of the same number must only be enabled for one pin.

Bits associated with non-existent pins are reserved. Reserved bits are read as 0. The write value should be 0.

P108 and P300 do not have the ISEL bit, so the ISEL bits in these Port Pin Function Select registers are reserved.

PMC bit (Pin Mode Control)

The PMC bit specifies the digital I/O or analog input function. When a pin is set as an analog pin by this bit:

1. Specify it as a general I/O port in the PSEL[2:0] bits.
2. Disable the pull-up resistor in the Pull-up Control bit (PmnPFS_A.PCR)

注意: 本产品现有接口:

- P1nPFS_A: 端口 1n 引脚功能选择寄存器(n = 00至 02、08 至 10、12)
- P2nPFS_A: 端口 2n 引脚功能选择寄存器(n = 00, 01, 06, 12, 13)
- P3nPFS_A: 端口 3n 引脚功能选择寄存器(n = 00)

PODR 位 (PMN 输出数据)、PIDR 位 (Pmn状态)、PDR 位 (Pmn方向)

PDR、PIDR 和 PODR 位与 PDRm、PIDRm 和 PODRm 的功能相同。读取这些位时，即读取 PDRm、PIDRm 和 PODRm 的值。

PCR 位点 (上拉对照)

PCR 位用于启用或禁用各个端口引脚上的输入上拉电阻。当引脚处于输入状态且 PmnPFS_A.PCR 中对应的位设置为 1 时，连接到该引脚的上拉电阻将被启用。当引脚设置为通用端口输出引脚时，无论 PCR 设置如何，该引脚的上拉电阻都将被禁用。复位状态下，上拉电阻也被禁用。与不存在的引脚关联的位将被保留。保留位读取为 0。写入值也应为 0。

P200 没有 PCR 位，因此这些端口引脚功能选择寄存器中的 PCR 位是保留的。

注意: 当选择 P206 (OFS1.PORTSELB= 0)) 时，可以选择 PCR。当选择 RES 引脚 (OFS1.PORTSELB= 1) 时，输入上拉电阻将被启用。

PIM 位 (引脚输入缓冲器选择)

PIM 位用于设置输入缓冲器。在与工作电压不同的外部设备进行串行通信时，可以选择 TTL 输入缓冲器。与不存在的引脚关联的位被保留。保留位读取为 0。写入值应为 0。

P200、P206、P212 和 P213 没有 PIM 位，因此这些端口引脚功能选择寄存器中的 PIM 位是保留的。

NCODR 位 (N 沟道开漏控制)

NCODR 位指定端口引脚的输出类型。在与工作电压不同的外部设备进行串行通信时，可以选择 N 通道开漏输出（耐受电压为 VCC）模式；在与工作电压相同的外部设备进行简化的 I2C 通信时，可以选择 SDA00 和 SDA11 引脚。此外，NCODR 与 PCR 结合使用，以指定是否使用片上上拉电阻。如果 N 通道开漏输出（耐受电压为 VCC）模式(NCODR = 1) 被置位，则片上上拉电阻不与该位连接。与不存在的引脚关联的位被保留。保留位读取为 0。写入值也应为 0。

P200 和 P206 没有 NCODR 位，因此这些端口引脚功能选择寄存器中的 NCODR 位是保留的。

PSEL[2:0] 位 (外设选择)

PSEL[2:0] 位用于分配外设功能。有关各产品外设设置的详细信息，请参见 15.6 节。[每个产品的外围设备选择设置](#)。

与不存在的引脚关联的位会被保留。保留位读取为 0。写入值应为 0。

P206 没有 PSEL[2:0] 位，因此这些端口引脚功能选择寄存器中的 PSEL[2:0] 位是保留的。

ISEL 位 (IRQ 输入使能)

ISEL 位指定 IRQ 输入引脚。此设置可与外设功能结合使用，但相同编号的 IRQn（外部引脚中断）只能为一个引脚启用。

与不存在的引脚关联的位会被保留。保留位读取为 0。写入值应为 0。

P108 和 P300 没有 ISEL 位，因此这些端口引脚功能选择寄存器中的 ISEL 位是保留的。

PMC 位 (引脚模式控制)

PMC 位指定数字输入或模拟输入功能。当此位将引脚设置为模拟引脚时:

1. 在 PSEL[2:0] 位中将其指定为通用端口I/O。
2. 禁用上拉控制位 (PmnPFS_A.PCR) 中的上拉电阻

3. Specify the input in the Port Direction bit (PmnPFS_A.PDR). The pin state cannot be read at this point. The PmnPFS_A register is protected by the Write-Protect Register (PWPR). Release write-protect before modifying the register.

In addition, the PMC bit is used to prevent through-current flowing into input buffers.

When N-ch open drain output is selected for serial communications with an external device operating at a different voltage or an input port is not used, low power consumption can be achieved by setting the corresponding PMC bit to 1.

Bits associated with non-existent pins are reserved. Reserved bits are read as 0. The write value should be 0.

P206 does not have the PMC bit, so the PMC bits in these Port Pin Function Select registers are reserved.

The ISEL bit for an unspecified IRQn is reserved.

15.2.7 P0nPFS_A : Port 0n Pin Function Select Register (n = 10 to 13)

Base address: PFS_A = 0x400A_0200																
Offset address: 0x0000 + 0x2 × n																
Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	PMC	ISEL	—	—	—	PSEL[2:0]		—	—	—	—	—	—	PDR	PIDR	PODR
Value after reset:	1	0	0	0	0	0	0	0	0	0	0	0	0	0	x	0

Bit	Symbol	Function	R/W
0	PODR	P0n Output Data 0: Low output 1: High output	R/W
1	PIDR	P0n State 0: Low level 1: High level	R
2	PDR	P0n Direction 0: Input (functions as an input pin) 1: Output (functions as an output pin)	R/W
7:3	—	These bits are read as 0. The write value should be 0.	R/W
10:8	PSEL[2:0]	Peripheral Select These bits select the peripheral function. For individual pin functions, see the associated tables in this chapter.	R/W
13:11	—	These bits are read as 0. The write value should be 0.	R/W
14	ISEL	IRQ Input Enable 0: Not used as an IRQn input pin 1: Used as an IRQn input pin	R/W
15	PMC	Pin Mode Control 0: Digital I/O 1: Analog input function. Input to the input buffer is disabled	R/W

15.2.8 PWPR : Write-Protect Register

Base address: PFS_A = 0x400A_0200							
Offset address: 0x0140							
Bit position:	7	6	5	4	3	2	1 0
Bit field:	B0WI	PFSW E	—	—	—	—	—
Value after reset:	1	0	0	0	0	0	0

Bit	Symbol	Function	R/W
5:0	—	These bits are read as 0. The write value should be 0.	R/W

3. 在端口方向位 (PmnPFS_A.PDR) 中指定输入。此时无法读取引脚状态。PmnPFS_A 寄存器受写保护寄存器 (PWPR) 保护。修改寄存器前，请先解除写保护。

此外，PMC 位用于防止电流流入输入缓冲器。

当选择 N 沟道开漏输出与工作电压不同的外部设备进行串行通信，或者未使用输入端口时，可以通过将相应的 PMC 位设置为 1 来实现低功耗。

与不存在的引脚关联的位会被保留。保留位读取为 0。写入值应为 0。

P206 没有 PMC 位，因此这些端口引脚功能选择寄存器中的 PMC 位是保留的。

未指定 IRQn 的 ISEL 位已保留。

15.2.7P0nPFS_A： 端口 0n 引脚功能选择寄存器(n = 10至 13)

基地址： PFS_A = 0x400A_0200																
偏移地址： 0x0000 + 0x2 × n																
位位置：	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域：	PMC	ISEL	—	—	—	PSEL[2:0]		—	—	—	—	—	PDR	PIDR	PODR	
重置后的值：	1	0	0	0	0	0	0	0	0	0	0	0	0	0	x	0

少量	象征	功能	R/W
0	PODR	P0n 输出数据 0: 低输出 1: 高输出	R/W
1	PIDR	P0n 州 0: 低级别 1: 高水平	R
2	PDR	P0n 方向 0: 输入（用作输入引脚） 1: 输出（用作输出引脚）	R/W
7:3	—	这些位读取为 0。写入值应为 0。	R/W
10:8	PSEL[2:0]	外围设备选择 这些位用于选择外设功能。有关各个引脚的功能，请参阅本章中的相关表格。	R/W
13:11	—	这些位读取为 0。写入值应为 0。	R/W
14	ISEL	IRQ 输入使能 0: 未用作 IRQn 输入引脚 1: 用作 IRQn 输入引脚	R/W
15	PMC	引脚模式控制 0: 数字 I/O 1: 模拟输入功能。输入缓冲器的输入已禁用。	R/W

15.2.8 PWPR: 写保护寄存器

基地址： PFS_A = 0x400A_0200							
偏移地址： 0x0140							
位位置：	7	6	5	4	3	2	1 0
位域：	B0WI	PFSW E	—	—	—	—	—
重置后的值：	1	0	0	0	0	0	0

少量	象征	功能	R/W
5:0	—	这些位读取为 0。写入值应为 0。	R/W

Bit	Symbol	Function	R/W
6	PFSWE	PmnPFS_A Register Write Enable 0: Writing to the PmnPFS_A register is disabled 1: Writing to the PmnPFS_A register is enabled	R/W
7	B0WI	PFSWE Bit Write Disable 0: Writing to the PFSWE bit is enabled 1: Writing to the PFSWE bit is disabled	R/W

PFSWE bit (PmnPFS_A Register Write Enable)

Writing to the PmnPFS_A register is enabled only when the PFSWE bit is set to 1. You must first write 0 to the B0WI bit before setting PFSWE to 1.

B0WI bit (PFSWE Bit Write Disable)

Writing to the PFSWE bit is enabled only when the B0WI bit is set to 0.

15.3 Operation

15.3.1 General I/O Ports

All pins except P108 and P300 operate as general I/O ports after reset. General I/O ports are organized as 16 bits per port and can be accessed by PODRm, PDRm, PIDRm, PORRm, POSRm or by individual pins with the Port mn Pin Function Select register. For details on these registers, see [section 15.2. Register Descriptions](#).

Each port has the following bits:

- Pmn Direction bit (PDRn), which selects input or output direction
- Pmn Output Data bit (PODRn), which holds data for output
- Pmn Input Data bit (PIDRn), which indicates the pin states
- Pmn Output Set bit (POSRn), which indicates the output value when a software write occurs
- Pmn Output Reset bit (PORRn), which indicates the output value when a software write occurs

15.3.2 Port Function Select

The following port functions are available for configuring each pin:

- I/O configuration: CMOS output or NMOS open-drain output, pull-up control
- General I/O port: Port direction, output data setting, and read input data
- Alternate function: Configured function mapping to the pin.

Each pin is associated with a Port mn Pin Function Select register (PmnPFS_A), which includes the associated PODR, PIDR, and PDR bits. In addition, the PmnPFS_A register includes the following:

- PCR: Pull-up resistor control bit that turns the input pull-up MOS on or off
- NCODR: N-channel open-drain control bit that selects the output type for each pin
- PIM: Pin input buffer selection bit to specify a normal input or TTL input buffer
- ISEL: IRQ input enable bit to specify an IRQ input pin
- PMC: Pin mode control bit to specify an analog pin and prevent through-current flowing into input buffers
- PSEL[2:0]: Port function select bits to select the associated peripheral function.

These configurations can be made by a single-register access to the Port mn Pin Function Select register. For details, see [section 15.2.6. PmnPFS_A : Port mn Pin Function Select Register \(m = 1 to 3, n = 00 to 15\)](#), [section 15.2.7. P0nPFS_A : Port 0n Pin Function Select Register \(n = 10 to 13\)](#).

15.4 Handling of Unused Pins

[Table 15.3](#) shows how to handle unused pins.

少量	象征	功能	R/W
6	PFSWE	PMN PFS_A 寄存器写使能 0: 已禁用对 PmnPFS_A 寄存器的写入操作。 1: 已启用对 PmnPFS_A 寄存器的写入操作。	R/W
7	B0WI	PFSWE 位写禁用 0: 已启用对 PFSWE 位的写入。 1: 已禁用对 PFSWE 位的写入。	R/W

PFSWE 位 (PmnPFS_A 寄存器写使能)

只有当 PFSWE 位设置为 1 时，才能写入PmnPFS_A寄存器。必须先将 0 写入B0WI位，然后才能将 PFSWE 设置为 1。

B0WI 位 (PFSWE 位写禁用)

只有当 B0WI 位设置为 0 时，才能写入 PFSWE 位。

15.3 操作

15.3.1 通用 I/O 端口

除 P108 和 P300 外，所有引脚在复位后均作为通用 I/O 端口运行。通用 I/O 端口每个端口 16 位，可通过 PODRm、PDRm、PIDRm、PORRm、POSRm 或使用端口 mn 引脚功能选择寄存器访问单个引脚。有关这些寄存器的详细信息，请参阅 15.2 节“寄存器描述”。

每个端口包含以下位：

- PMN方向位（PDRn）， 用于选择输入或输出方向
- PMN 输出数据位 (PODRn)，用于保存输出数据
- PMN 输入数据位 (PIDRn)，指示引脚状态
- PMN 输出置位 (POSRn)，指示软件写入发生时的输出值
- PMN 输出复位位 (PORRn)，指示软件写入发生时的输出值

15.3.2端口功能选择

以下端口功能可用于配置每个引脚：

- I/O配置：CMOS输出或NMOS开漏输出，上拉控制
- 通用I/O端口：端口方向、输出数据设置和读取输入数据
- 备用功能：配置的功能映射到引脚。

每个引脚都与一个端口 mn 引脚功能选择寄存器 (PmnPFS_A) 相关联，该寄存器包含相关的 PODR，PIDR 和 PDR 位。此外，PmnPFS_A 寄存器还包括以下内容：

- PCR：上拉电阻控制位，用于开启或关闭输入上拉MOS管
- NCODR：N沟道开漏控制位，用于选择每个引脚的输出类型
- PIM：引脚输入缓冲器选择位，用于指定普通输入缓冲器或TTL输入缓冲器。
- ISEL：IRQ 输入使能位，用于指定 IRQ 输入引脚
- PMC：引脚模式控制位，用于指定模拟引脚并防止电流流入输入缓冲器。
- PSEL[2:0]: 端口功能选择位，用于选择关联的外设功能。

这些配置可以通过对端口 mn 引脚功能选择寄存器的单寄存器访问来实现。详情请参见 15.2.6 节。PmnPFS_A: 端口 m n 引脚功能选择寄存器(m = 1至 3， n = 00 至 15)，15.2.7 节。P0nPFS_A: 端口 0n 引脚功能选择寄存器(n = 10至 13)。

15.4 未使用引脚的处理

表 15.3 显示了如何处理未使用的引脚。

Table 15.3 Handling of unused pins

Pin name	Description
P206/RES	PORTSELB = 0: - If the direction setting is for input (PDR2.PDR06 = 0), connect the associated pin to VCC (pulled up) through a resistor.*2 - If the direction setting is for output (PDR2.PDR06 = 1), keep pin open. PORTSELB = 1: Keep pin open, or connect to VCC.
P200/NMI	Set in Pin Mode Control bit (PmnPFS_A.PMC) to 1, and keep pin open. Alternatively, connect the associated pin to VCC (pulled up) through a resistor or to VSS (pulled down) through a resistor.
P0x to P3x	- If the direction setting is for input (PDRm.PDRn = 0), connect the associated pin to VCC (pulled up) through a resistor or to VSS (pulled down) through a resistor*1,*2 - If the direction setting is for output (PDRm.PDRn = 1), keep pin open*1

Note 1. Clear the PmnPFS_A.PSEL[2:0], PmnPFS_A.ISEL, PmnPFS_A.PCR, and PmnPFS_A.PMC bits to 0.
Note 2. P108, P206, and P300 are recommended to be pulled up to VCC through a resistor, as they are configured with input pull-up enabled (PmnPFS_A.PCR = 1) by default.

15.5 Usage Notes

15.5.1 Procedure for Specifying the Pin Functions

To specify the I/O pin functions:

- Clear the B0WI bit in the PWPR register. This enables writing to the PFSWE bit in the PWPR register.
- Set 1 to the PFSWE bit in the PWPR register. This enables writing to the PmnPFS_A register.
- Specify the I/O function for the pin through the PSEL[2:0] bits settings in the PmnPFS_A register.
- Clear the PFSWE bit in the PWPR register. This disables writing to the PmnPFS_A register.
- Set 1 to the B0WI bit in the PWPR register. This disables writing to the PFSWE bit in the PWPR register.

15.5.2 Port Output Data Register (PODR) Summary

This register outputs data as follows:

- Outputs 0 if PORRm.PORRn is set to 1.
- Outputs 1 if POSRm.POSRn is set to 1.
- Outputs 0 or 1 because PODRm.PODRn is set.
- Outputs 0 or 1 because PmnPFS_A.PODRn is set.

Numbers in this list correspond to the priority for writing to the PODRn. For example, if 1. and 3. from the list occur at the same time, the higher priority event 1. is executed.

15.5.3 Notes on Register Settings and Port Pin State

The correspondence between register settings and port pin state is shown in [Table 15.4](#).

Table 15.4 Correspondence between register settings and port pin state (1 of 2)

PDR	PMC	PCR	PODR	Pin State
0	1	0	x	Analog input /digital input disable
0	1	1	x	Pulled up
0	0	0	x	Digital Input
0	0	1	x	Pulled up
1	1	x	1	High-level port output/digital input disable
1	0	x	1	High-level port output
1	1	x	0	Low-level port output/digital input disable

表 15.3 未使用引脚的处理

引脚名称	描述
P206/RES	PORTSELB = 0: ●如果方向设置是针对输入(PDR2.PDR06 = 0) ，则通过电阻将相关引脚连接到VCC（上拉）。*2 ●如果方向设置为输出（PDR2.PDR06= 1），则保持引脚开路。 PORTSELB = 1: 保持引脚断开，或连接到 VCC。
P200/NMI	将引脚模式控制位 (PmnPFS_A.PMC) 设置为 1，并保持引脚打开。 或者，通过电阻器将相关引脚连接到 VCC（上拉）或 VSS（下拉）。
P0x 至 P3x	●如果方向设置针对输入(PDRm.PDRn = 0)，则将相关引脚通过电阻连接到VCC（上拉）或通过电阻*1、*2连接到VSS（下拉）。 ●如果方向设置为输出（PDRm.PDRn= 1）保持引脚开路*1），则引脚保持开路*1

注 1. 将 PmnPFS_A.PSEL[2:0]、PmnPFS_A.ISEL、PmnPFS_A.PCR 和 PmnPFS_A.PMC 位清除为 0。
注 2. 建议通过电阻器将 P108、P206 和 P300上拉至 VCC，因为默认情况下它们配置为启用输入上拉 (PmnPFS_A.PCR= 1)。

15.5 使用说明

15.5.1 引脚功能指定程序

指定 I/O 引脚功能:

- 清除 PWPR 寄存器中的 B0WI 位。这将允许写入 PWPR 寄存器中的 PFSWE 位。
- 将 PWPR 寄存器中的 PFSWE 位设置为 1。这将启用对 PmnPFS_A 寄存器的写入操作。
- 通过 PmnPFS_A 寄存器中的 PSEL[2:0] 位设置，为引脚指定I/O功能。
- 清除 PWPR 寄存器中的 PFSWE 位。这将禁用对 PmnPFS_A 寄存器的写入操作。
- 将 PWPR 寄存器中的 B0WI 位设置为 1。这将禁用对 PWPR 寄存器中 PFSWE 位的写入操作。

15.5.2端口输出数据寄存器（PODR）概述

该寄存器输出的数据如下:

- 如果 PORRm.PORRn 设置为 1，则输出 0。
- 如果 POSRm.POSRn 设置为 1，则输出 1。
- 输出 0 或 1，因为 PODRm.PODRn 已设置。
- 输出 0 或 1，因为 PmnPFS_A.PODRn 已设置。

此列表中的数字对应于写入 PODRn 的优先级。例如，如果列表中的 1 和 3 同时发生，则执行优先级更高的事件 1。

15.5.3 关于寄存器设置和端口引脚状态的说明

寄存器设置与端口引脚状态的对应关系如表 15.4 所示。

表 15.4 寄存器设置与端口引脚状态的对应关系（1/2）

PDR	PMC	PCR	PODR	引脚状态
0	1	0	x	模拟输入/数字输入禁用
0	1	1	x	拉起来
0	0	0	x	数字输入
0	0	1	x	拉起来
1	1	x	1	高电平端口输出/数字输入禁用
1	0	x	1	高电平端口输出
1	1	x	0	低电平端口输出/数字输入禁用

Table 15.4 Correspondence between register settings and port pin state (2 of 2)

PDR	PMC	PCR	PODR	Pin State
1	0	x	0	Low-level port output

15.5.4 Notes on Using Analog Functions

To use an analog function, set the Pin Mode Control bit (PMC) to 1, the N-Channel Open-Drain Control bit (NCODR) to 0, and the Port Direction bit (PDR) to 0 in the Port mn Pin Function Select Register PmnPFS_A.

15.5.5 Notes on Using Alternate Functions

To use alternate functions, set PmnPFS_A register. Examples of register settings for port and alternate functions are shown in below table. The registers used to control the port functions should be set as shown in below table.

Table 15.5 Examples of register settings for port and alternate functions (1/5)

Pin Name	Function Used		PSEL[2:0]	ISEL	PMC	PDR	PODR
	Function Name	I/O					
P010	P010	Input	000b	—	0	0	x
		Output	000b	—	x	1	0/1
	AN000	Analog Input	000b	—	1	0	x
P011	P011	Input	000b	—	0	0	x
		Output	000b	—	x	1	0/1
	AN001	Analog Input	000b	—	1	0	x
P012	P012	Input	000b	—	0	0	x
		Output	000b	—	x	1	0/1
	AN004	Analog Input	000b	—	1	0	x
P013	P013	Input	000b	—	0	0	x
		Output	000b	—	x	1	0/1
	AN005	Analog Input	000b	—	1	0	x

Table 15.6 Examples of register settings for port and alternate functions (2/5) (1 of 4)

Pin Name	Function Used		PSEL[2:0]	ISEL	NCODR	PMC	PDR	PODR
	Function Name	I/O						
P100	P100	Input	000b	0	0	0	0	x
		Output	000b	0	0	x	1	0/1
		N-ch open drain output	000b	0	1	x	1	0/1
	AN022	Analog Input	000b	0	0	1	0	x
	TI04	Input	001b	0	0	0	0	x
	TO04	Output	001b	0	0	x	1	x
	TI01	Input	010b	0	0	0	0	x
	TO01	Output	010b	0	0	x	1	x
	RXD0	Input	011b	0	x	0	0	x
	SI00	Input	011b	0	x	0	0	x
	SDA00	I/O	011b	0	1	0	1	x
	SCLA0	I/O	100b	0	1	0	1	x
	IRQ2	Input	000b	1	0	0	0	x

表 15.4 寄存器设置与端口引脚状态的对应关系 (2/2)

PDR	PMC	PCR	PODR	引脚状态
1	0	x	0	低电平端口输出

15.5.4 关于使用模拟函数的注意事项

要使用模拟功能，请在端口 mn 引脚功能选择寄存器 PmnPFS_A 中，将引脚模式控制位 (PMC) 设置为 1，将 N 沟道开漏控制位 (NCODR) 设置为 0，将端口方向位 (PDR) 设置为 0。

15.5.5 关于使用替代函数的说明

要使用备用功能，请设置PmnPFS_A寄存器。端口和备用功能的寄存器设置示例如下表所示。用于控制端口功能的寄存器应按如下表所示进行设置。

表 15.5 端口和备用功能的寄存器设置示例(1/5)

别针名称	使用的功能		PSEL[2:0]	ISEL	PMC	PDR	PODR
	函数名称	I/O					
P010	P010	输入	000b	—	0	0	x
		输出	000b	—	x	1	0/1
	AN000	模拟输入	000b	—	1	0	x
P011	P011	输入	000b	—	0	0	x
		输出	000b	—	x	1	0/1
	AN001	模拟输入	000b	—	1	0	x
P012	P012	输入	000b	—	0	0	x
		输出	000b	—	x	1	0/1
	AN004	模拟输入	000b	—	1	0	x
P013	P013	输入	000b	—	0	0	x
		输出	000b	—	x	1	0/1
	AN005	模拟输入	000b	—	1	0	x

表 15.6 端口和备用功能的寄存器设置示例 (2/5) (1/4)

别针名称	使用的功能		PSEL[2:0]	ISEL	NCODR	PMC	PDR	PODR
	函数名称	I/O						
P100	P100	输入	000b	0	0	0	0	x
		输出	000b	0	0	x	1	0/1
		N沟道开漏输出	000b	0	1	x	1	0/1
	AN022	模拟输入	000b	0	0	1	0	x
	TI04	输入	001b	0	0	0	0	x
	TO04	输出	001b	0	0	x	1	x
	TI01	输入	010b	0	0	0	0	x
	TO01	输出	010b	0	0	x	1	x
	RXD0	输入	011b	0	x	0	0	x
	SI00	输入	011b	0	x	0	0	x
	SDA00	I/O	011b	0	1	0	1	x
	SCLA0	I/O	100b	0	1	0	1	x
	IRQ2	输入	000b	1	0	0	0	x

Table 15.6 Examples of register settings for port and alternate functions (2/5) (2 of 4)

Pin Name	Function Used		PSEL[2:0]	ISEL	NCODR	PMC	PDR	PODR
	Function Name	I/O						
P101	P101	Input	000b	0	0	0	0	x
		Output	000b	0	0	x	1	0/1
		N-ch open drain output	000b	0	1	x	1	0/1
	AN021	Analog Input	000b	0	0	1	0	x
	TI07	Input	001b	0	0	0	0	x
	TO07	Output	001b	0	0	x	1	x
	TI00	Input	010b	0	0	0	0	x
	TXD0	Output	011b	0	0/1	0/1	1	x
	SO00	Output	011b	0	0/1	0/1	1	x
	SDAA0	I/O	100b	0	1	0	1	x
	IRQ3	Input	000b	1	0	0	0	x
P102	P102	Input	000b	0	0	0	0	x
		Output	000b	0	0	x	1	0/1
		N-ch open drain output	000b	0	1	x	1	0/1
	TI06	Input	001b	0	0	0	0	x
	TO06	Output	001b	0	0	x	1	x
	TO00	Output	010b	0	0	x	1	x
	SCK00	Input	011b	0	x	0	0	x
		Output	011b	0	0/1	0/1	1	x
	SCL00	Output	011b	0	0/1	0/1	1	x
	PCLBUZ	Output	101b	0	0	x	1	x
	IRQ4	Input	000b	1	0	0	0	x
P108	P108	Input	000b	—	0	0	0	x
		Output	000b	—	0	x	1	0/1
		N-ch open drain output	000b	—	1	x	1	0/1
	SWDIO	I/O	001b	—	x	x	x	x
	TI03	Input	010b	—	0	0	0	x
	TO03	Output	010b	—	0	x	1	x
	SCK00	Input	011b	0	x	0	0	x
		Output	011b	0	0/1	0/1	1	x
	SCLA0	I/O	100b	0	1	0	1	x
	PCLBUZ	Output	101b	0	0	x	1	x

表 15.6 端口和备用功能的寄存器设置示例 (2/5) (2/4)

别针名称	使用的功能		PSEL[2:0]	ISEL	NCODR	PMC	PDR	PODR
	函数名称	I/O						
P101	P101	输入	000b	0	0	0	0	x
		输出	000b	0	0	x	1	0/1
		N沟道开漏输出	000b	0	1	x	1	0/1
	AN021	模拟输入	000b	0	0	1	0	x
	TI07	输入	001b	0	0	0	0	x
	TO07	输出	001b	0	0	x	1	x
	TI00	输入	010b	0	0	0	0	x
	TXD0	输出	011b	0	0/1	0/1	1	x
	SO00	输出	011b	0	0/1	0/1	1	x
	SDAA0	I/O	100b	0	1	0	1	x
	IRQ3	输入	000b	1	0	0	0	x
P102	P102	输入	000b	0	0	0	0	x
		输出	000b	0	0	x	1	0/1
		N沟道开漏输出	000b	0	1	x	1	0/1
	TI06	输入	001b	0	0	0	0	x
	TO06	输出	001b	0	0	x	1	x
	TO00	输出	010b	0	0	x	1	x
	SCK00	输入	011b	0	x	0	0	x
		输出	011b	0	0/1	0/1	1	x
	SCL00	输出	011b	0	0/1	0/1	1	x
	PCLBUZ	输出	101b	0	0	x	1	x
	IRQ4	输入	000b	1	0	0	0	x
P108	P108	输入	000b	—	0	0	0	x
		输出	000b	—	0	x	1	0/1
		N沟道开漏输出	000b	—	1	x	1	0/1
	SWDIO	I/O	001b	—	x	x	x	x
	TI03	输入	010b	—	0	0	0	x
	TO03	输出	010b	—	0	x	1	x
	SCK00	输入	011b	0	x	0	0	x
		输出	011b	0	0/1	0/1	1	x
	SCLA0	I/O	100b	0	1	0	1	x
	PCLBUZ	输出	101b	0	0	x	1	x

Table 15.6 Examples of register settings for port and alternate functions (2/5) (3 of 4)

Pin Name	Function Used		PSEL[2:0]	ISEL	NCODR	PMC	PDR	PODR
	Function Name	I/O						
P109	P109	Input	000b	0	0	0	0	x
		Output	000b	0	0	x	1	0/1
		N-ch open drain output	000b	0	1	x	1	0/1
	TI02	Input	001b	0	0	0	0	x
	TO02	Output	001b	0	0	x	1	x
	TXD0	Output	010b	0	0/1	0/1	1	x
	SO00	Output	010b	0	0/1	0/1	1	x
	SDAA0	I/O	011b	0	1	0	1	x
	IRQ4	Input	000b	1	0	0	0	x
P110	P110	Input	000b	0	0	0	0	x
		Output	000b	0	0	x	1	0/1
		N-ch open drain output	000b	0	1	x	1	0/1
	TI01	Input	001b	0	0	0	0	x
	TO01	Output	001b	0	0	x	1	x
	RXD0	Input	010b	0	x	0	0	x
	SI00	Input	010b	0	x	0	0	x
	SDA00	I/O	010b	0	1	0	1	x
	SCLA0	I/O	011b	0	1	0	1	x
P112	P112	Input	000b	0	0	0	0	x
		Output	000b	0	0	x	1	0/1
		N-ch open drain output	000b	0	1	x	1	0/1
	TI03	Input	001b	0	0	0	0	x
	TO03	Output	001b	0	0	x	1	x
	SCK00	Input	010b	0	x	0	0	x
		Output	010b	0	0/1	0/1	1	x
	SCL00	Output	010b	0	0/1	0/1	1	x
	SSI00	Input	011b	0	0	0	0	x
P200	P200	Input	000b	0	—	0	—	—
	IRQ0	Input	000b	1	—	0	—	—
	TI06	Input	001b	0	—	0	—	—
	SSI00	Input	010b	0	—	0	—	—
	RXD1	Input	011b	0	—	0	—	—

表 15.6 端口和备用功能的寄存器设置示例 (2/5) (3/4)

别针名称	使用的功能		PSEL[2:0]	ISEL	NCODR	PMC	PDR	PODR
	函数名称	I/O						
P109	P109	输入	000b	0	0	0	0	x
		输出	000b	0	0	x	1	0/1
		N沟道开漏输出	000b	0	1	x	1	0/1
	TI02	输入	001b	0	0	0	0	x
	TO02	输出	001b	0	0	x	1	x
	TXD0	输出	010b	0	0/1	0/1	1	x
	SO00	输出	010b	0	0/1	0/1	1	x
	SDAA0	I/O	011b	0	1	0	1	x
	IRQ4	输入	000b	1	0	0	0	x
P110	P110	输入	000b	0	0	0	0	x
		输出	000b	0	0	x	1	0/1
		N沟道开漏输出	000b	0	1	x	1	0/1
	TI01	输入	001b	0	0	0	0	x
	TO01	输出	001b	0	0	x	1	x
	RXD0	输入	010b	0	x	0	0	x
	SI00	输入	010b	0	x	0	0	x
	SDA00	I/O	010b	0	1	0	1	x
	SCLA0	I/O	011b	0	1	0	1	x
P112	P112	输入	000b	0	0	0	0	x
		输出	000b	0	0	x	1	0/1
		N沟道开漏输出	000b	0	1	x	1	0/1
	TI03	输入	001b	0	0	0	0	x
	TO03	输出	001b	0	0	x	1	x
	SCK00	输入	010b	0	x	0	0	x
		输出	010b	0	0/1	0/1	1	x
	SCL00	输出	010b	0	0/1	0/1	1	x
	SSI00	输入	011b	0	0	0	0	x
P200	P200	输入	000b	0	—	0	—	—
	IRQ0	输入	000b	1	—	0	—	—
	TI06	输入	001b	0	—	0	—	—
	SSI00	输入	010b	0	—	0	—	—
	RXD1	输入	011b	0	—	0	—	—

Table 15.6 Examples of register settings for port and alternate functions (2/5) (4 of 4)

Pin Name	Function Used		PSEL[2:0]	ISEL	NCODR	PMC	PDR	PODR
	Function Name	I/O						
P201	P201	Input	000b	0	0	0	0	x
		Output	000b	0	0	x	1	0/1
		N-ch open drain output	000b	0	1	x	1	0/1
	TI05	Input	001b	0	0	0	0	x
	TO05	Output	001b	0	0	x	1	x
	SSI00	Input	010b	0	0	0	0	x
	SCK11	Input	011b	0	x	0	0	x
		Output	011b	0	0/1	0/1	1	x
	SCL11	Output	011b	0	0/1	0/1	1	x
	PCLBUZ	Output	101b	0	0	x	1	x
	IRQ5	Input	000b	1	0	0	0	x

Table 15.7 Examples of register settings for port and alternate functions (3/5)

Pin Name	Function Used		ISEL	PORTSELB	PDR	PODR
	Function Name	I/O				
P206	P206	Input	0	0	0	0
		Output	0	0	1	0/1
	RES	Input	0	1	—	—
	IRQ1	Input	1	0	0	0

Table 15.8 Examples of register settings for port and alternate functions (4/5) (1 of 2)

Pin Name	Function Used		PSEL[2:0]	ISEL	NCODR	PMC	PDR	PODR
	Function Name	I/O						
P212	P212	Input	000b	0	0	0	0	x
		Output	000b	0	0	x	1	0/1
		N-ch open drain output	000b	0	1	x	1	0/1
	TO00	Output	001b	0	0	x	1	x
	TI03	Input	010b	0	0	0	0	x
	TO03	Output	010b	0	0	x	1	x
	RXD1	Input	011b	0	x	0	0	x
	SI11	Input	100b	0	x	0	0	x
	SDA11	I/O	100b	0	1	0	1	x
	SCLA0	I/O	101b	0	1	0	1	x
	IRQ1	Input	000b	1	0	0	0	x

表 15.6 端口和备用功能的寄存器设置示例 (2/5) (4/4)

别针名称	使用的功能		PSEL[2:0]	ISEL	NCODR	PMC	PDR	PODR
	函数名称	I/O						
P201	P201	输入	000b	0	0	0	0	x
		输出	000b	0	0	x	1	0/1
		N沟道开漏输出	000b	0	1	x	1	0/1
	TI05	输入	001b	0	0	0	0	x
	TO05	输出	001b	0	0	x	1	x
	SSI00	输入	010b	0	0	0	0	x
	SCK11	输入	011b	0	x	0	0	x
		输出	011b	0	0/1	0/1	1	x
	SCL11	输出	011b	0	0/1	0/1	1	x
	PCLBUZ	输出	101b	0	0	x	1	x
	IRQ5	输入	000b	1	0	0	0	x

表 15.7 端口和备用功能的寄存器设置示例(3/5)

别针名称	使用的功能		ISEL	波特塞尔布	PDR	PODR
	函数名称	I/O				
P206	P206	输入	0	0	0	0
		输出	0	0	1	0/1
	RES	输入	0	1	—	—
	IRQ1	输入	1	0	0	0

表 15.8 端口和备用功能的寄存器设置示例 (4/5) (1/2)

别针名称	使用的功能		PSEL[2:0]	ISEL	NCODR	PMC	PDR	PODR
	函数名称	I/O						
P212	P212	输入	000b	0	0	0	0	x
		输出	000b	0	0	x	1	0/1
		N沟道开漏输出	000b	0	1	x	1	0/1
	TO00	输出	001b	0	0	x	1	x
	TI03	输入	010b	0	0	0	0	x
	TO03	输出	010b	0	0	x	1	x
	RXD1	输入	011b	0	x	0	0	x
	SI11	输入	100b	0	x	0	0	x
	SDA11	I/O	100b	0	1	0	1	x
	SCLA0	I/O	101b	0	1	0	1	x
	IRQ1	输入	000b	1	0	0	0	x

Table 15.8 Examples of register settings for port and alternate functions (4/5) (2 of 2)

Pin Name	Function Used		PSEL[2:0]	ISEL	NCODR	PMC	PDR	PODR
	Function Name	I/O						
P213	P213	Input	000b	0	0	0	0	x
		Output	000b	0	0	x	1	0/1
		N-ch open drain output	000b	0	1	x	1	0/1
	TI00	Input	001b	0	0	0	0	x
	TI02	Input	010b	0	0	0	0	x
	TO02	Output	010b	0	0	x	1	x
	TXD1	Output	011b	0	0/1	0/1	1	x
	SO11	Output	100b	0	0/1	0/1	1	x
	SDAA0	I/O	101b	0	1	0	1	x
	IRQ0	Input	000b	1	0	0	0	x

Table 15.9 Examples of register settings for port and alternate functions (5/5)

Pin Name	Function Used		PSEL[2:0]	ISEL	NCODR	PMC	PDR	PODR
	Function Name	I/O						
P300	P300	Input	000b	—	0	0	0	x
		Output	000b	—	0	x	1	0/1
		N-ch open drain output	000b	—	1	x	1	0/1
	SWCLK	Input	001b	—	x	x	x	x
	TI04	Input	010b	—	0	0	0	x
	TO04	Output	010b	—	0	x	1	x
	TXD1	Output	011b	—	0/1	0/1	1	x
	SDAA0	I/O	100b	—	1	0	1	x

15.5.6 Notes on Communications with Devices Operating at a Different Voltage (1.8 V, 2.5 V, or 3 V) by Switching I/O Buffers

The pin input buffer selection bits (PIM) and the N-Channel Open-Drain Control bits (NCODR) can be used to switch the I/O buffers to enable communications with external devices that have different operating voltages (1.8 V, 2.5 V, or 3 V) to this device.

- (1) Procedure for setting input pins of UART0 to UART1, SPI00, SPI11 for use with the TTL input buffers
1. Pull up the input pin to be used to the voltage of the target device via an external resistor. The on-chip pull-up resistor cannot be used for this purpose.

2. Set the PIM bit to 1 to switch to the TTL input buffer. For V_{IH} and V_{IL} , refer to the DC characteristics when the TTL input buffer is selected.

3. Enable the operation of the serial array unit and set the mode to the UART/simplified SPI mode.
- (2) Procedure for setting output pins of UART0 to UART1, SPI00, SPI11 for use with the N-ch open-drain output mode
1. Pull up the input pin to be used to the voltage of the target device via an external resistor. The on-chip pull-up resistor cannot be used for this purpose.

2. The port pins are set for input (Hi-Z) after the reset state is released.

3. Set the PMC bits to 1 to disable input to the input buffer.

表 15.8 端口和备用功能的寄存器设置示例 (4/5) (2/2)

别针名称	使用的功能		PSEL[2:0]	ISEL	NCODR	PMC	PDR	PODR
	函数名称	I/O						
P213	P213	输入	000b	0	0	0	0	x
		输出	000b	0	0	x	1	0/1
		N沟道开漏输出	000b	0	1	x	1	0/1
	TI00	输入	001b	0	0	0	0	x
	TI02	输入	010b	0	0	0	0	x
	TO02	输出	010b	0	0	x	1	x
	TXD1	输出	011b	0	0/1	0/1	1	x
	SO11	输出	100b	0	0/1	0/1	1	x
	SDAA0	I/O	101b	0	1	0	1	x
	IRQ0	输入	000b	1	0	0	0	x

表 15.9 端口和备用功能的寄存器设置示例 (5/5)

别针名称	使用的功能		PSEL[2:0]	ISEL	NCODR	PMC	PDR	PODR
	函数名称	I/O						
P300	P300	输入	000b	—	0	0	0	x
		输出	000b	—	0	x	1	0/1
		N沟道开漏输出	000b	—	1	x	1	0/1
	SWCLK	输入	001b	—	x	x	x	x
	TI04	输入	010b	—	0	0	0	x
	TO04	输出	010b	—	0	x	1	x
	TXD1	输出	011b	—	0/1	0/1	1	x
	SDAA0	I/O	100b	—	1	0	1	x

15.5.6 关于与工作电压不同的设备（1.8 V、2.5 V 或3 V）通过切换I/O缓冲器）进行通信的说明

引脚输入缓冲器选择位 (PIM) 和N通道开漏控制位 (NCODR) 可用于切换I/O缓冲器，从而实现与具有不同工作电压(1.8 V, 2.5 V, 或3 V)的外部设备的通信。

- (1) 将UART0的输入引脚设置为UART1、SPI00、SPI11以与TTL输入缓冲器一起使用的步骤
1. 通过外部电阻将要使用的输入引脚上拉至目标器件的电压。片上上拉电阻不能用于此目的。

2. 将 PIM 位设置为 1 以切换到 TTL 输入缓冲器。对于 V_{IH} 和 V_{IL} ，请参考选择 TTL 输入缓冲器时的直流特性。

3. 启用串行阵列单元的操作，并将模式设置为 UART/简化 SPI 模式。
- (2) 将 UART0 的输出引脚设置为 UART1、SPI00 和 SPI11 以用于 N 沟道开漏输出模式的步骤
1. 通过外部电阻将要使用的输入引脚上拉至目标器件的电压。片上上拉电阻不能用于此目的。

2. 复位状态释放后，端口引脚设置为输入（高阻抗）。

3. 将 PMC 位设置为 1，以禁用输入缓冲区的输入。

- 4. Set the NCODR bits to 1 to set the N-ch open drain output (withstand voltage of VCC) mode.
- 5. Enable the operation of the serial array unit and set the mode to the UART/simplified SPI mode.
- 6. Set the PDR bits to the output mode. At this time, the output data is high level, so the pin is in the Hi-Z state.

(3) Procedure for setting I/O pins of IIC00 for use in connection with a device operating at a different voltage (1.8 V, 2.5 V, or 3 V)

- 1. Pull up the input pin to be used to the voltage of the target device via an external resistor. The on-chip pull-up resistor cannot be used for this purpose.
- 2. The port pins are set for input (Hi-Z) after the reset state is released.
- 3. Set the NCODR bits to 1 to set the N-ch open drain output (withstand voltage of VCC) mode.
- 4. Set the PIM bit to 1 to switch to the TTL input buffer. For V_{IH} and V_{IL}, refer to the DC characteristics when the TTL input buffer is selected.
- 5. Enable the operation of the serial array unit and set the mode to the simplified I²C mode.
- 6. Set the PDR bits to the output mode (data I/O is possible in the output mode). At this time, the output data is high level, so the pin is in the Hi-Z state.

Note: The input buffer is enabled even if the pin is operating as an output when the N-ch open-drain output mode is selected by the corresponding bit in the N-channel open-drain control bit (NCODR). This may lead to a through current flowing through the pin when the voltage level on this pin is intermediate. However, setting the corresponding bit of the given PMC bit to 1 prevents the flow of a through-current.

Note: When the pin is set to TTL input buffer by the port input buffer selection bit (PIMx) and is driven high, a through-current may flow through the pin due to the configuration of the TTL input buffer. Drive the pin low to prevent the through current.

Note: Communications by using P212 and P213 with devices operating at different voltage levels are not possible since P212PFS_A and P213PFS_A registers do not have PIM bit.

15.5.7 Restriction on P206 Usage

Once the power is turned on, P206 functions as the RES input. The PORTSELB bit of the option select register1 (OFS1) defines whether this port operates as P206 or RES. When this pin is set to P206, do not input the low level to this pin during a reset by the power-on-reset (POR) circuit and during the period from release from the reset by the POR circuit to the start of normal operation. If input of the low level continues during this period, the chip will remain in the reset state in response to the external reset. The on-chip pull-up resistor is enabled after power is turned on.

15.6 Peripheral Select Settings for Each Product

This section describes the pin function select configuration by the PmnPFS_A register. Some signal names have _A, _B, _C, _D, or _E suffixes, but these suffixes can be ignored when assigning functionality, except for SAU and IICA. For SAU and IICA, only signals, except for SCL11, SCK11, SCK00, and SSI00, bearing the same suffix can be selected. Assigning the same function to two or more pins simultaneously is prohibited. Only the allowed values (functions) should be specified in the PSEL bits of PmnPFS_A. If a value that is not allowed for the register is specified, the correct operation is not guaranteed.

Table 15.10 Register settings for the pin function select configuration (PORT0)

PSEL[2:0] settings	Pin			
	P010	P011	P012	P013
000b	P0n Output Data (initial)			
PMC bit	✓ (AN000/VREFH0)	✓ (AN001/VREFL0)	✓ (AN004)	✓ (AN005)
ISEL bit	—	—	—	—
NCODR bit	—	—	—	—
PIM bit	—	—	—	—
PCR bit	—	—	—	—
20-pin product	✓	✓	✓	✓

✓: Available

- 4. 将 NCODR 位设置为 1，以设置 N 沟道开漏输出（耐压 VCC）模式。
- 5. 启用串行阵列单元的操作，并将模式设置为 UART/简化 SPI 模式。
- 6. 将 PDR 位设置为输出模式。此时，输出数据为高电平，因此引脚处于高阻态。

(3) 设置 IIC00 的 I/O 引脚以连接工作电压不同的设备（1.8 V、2.5 V 或 3 V）的步骤

- 1. 通过外部电阻将要使用的输入引脚上拉至目标器件的电压。片上上拉电阻不能用于此目的。
- 2. 复位状态释放后，端口引脚设置为输入（高阻抗）。
- 3. 将 NCODR 位设置为 1，以设置 N 沟道开漏输出（耐压 VCC）模式。
- 4. 将 PIM 位设置为 1 以切换到 TTL 输入缓冲器。对于 V_{IH}和 V_{IL}，请参考选择 TTL 输入缓冲器时的直流特性。
- 5. 启用串行阵列单元的操作，并将模式设置为简化的 I²C 模式。
- 6. 将 PDR 位设置为输出模式（输出模式下可以输出数据 I/O）。此时，输出数据为高电平，因此引脚处于高阻态。

注意：即使引脚作为输出工作，当 N 沟道开漏控制位 (NCODR) 中的相应位选择 N 沟道开漏输出模式时，输入缓冲器也会被启用。这可能导致当该引脚上的电压电平处于中间值时，有电流流过该引脚。但是，将给定 PMC 位的相应位设置为 1 可以防止电流流过。

注意：当端口输入缓冲器选择位 (PIMx) 将引脚设置为 TTL 输入缓冲器并使其拉高时，由于 TTL 输入缓冲器的配置，可能会有电流流过该引脚。将引脚拉低以防止电流流过。

注意：由于 P212PFS_A 和 P213PFS_A 寄存器没有 PIM 位，因此无法使用 P212 和 P213 与工作电压不同的设备进行通信。

15.5.7 对 P206 使用的限制

上电后，P206 用作 RES 输入。选项选择寄存器 1 (OFS1) 的 PORTSELB 位定义此端口是作为 P206 还是 RES 工作。当此引脚设置为 P206 时，在上电复位 (POR) 电路复位期间以及从 POR 电路复位解除到正常工作开始的这段时间内，请勿向此引脚输入低电平。如果在此期间持续输入低电平，芯片将响应外部复位信号而保持复位状态。上电后，片上上拉电阻将被启用。

15.6 各产品的外围设备选择设置

本节介绍通过 PmnPFS_A 寄存器进行引脚功能选择配置。某些信号名称带有 _A、_B、_C、_D、或 _E 后缀，但除 SAU 和 IICA 外，在分配功能时可以忽略这些后缀。对于 SAU 和 IICA，只能选择具有相同后缀的信号（SCL11、SCK11、SCK00 和 SSI00 除外）。禁止同时为两个或多个引脚分配相同的功能。PmnPFS_A 的 PSEL 位中只能指定允许的值（功能）。如果指定了寄存器不允许的值，则无法保证操作正确。

表 15.10 引脚功能选择配置 (PORT0) 的寄存器设置

PSEL[2:0] 设置	别针			
	P010	P011	P012	P013
000b	P0n 输出数据（初始）			
PMC 比特	✓ (AN000/VREFH0)	✓ (AN001/VREFL0)	✓ (AN004)	✓ (AN005)
低位	—	—	—	—
NCODR 位	—	—	—	—
PIM 位	—	—	—	—
PCR 位	—	—	—	—
20 针产品	✓	✓	✓	✓

：有货

—: Setting prohibited

Table 15.11 Register settings for the pin function select configuration (PORT1)

PSEL[2:0] settings	Pin						
	P100	P101	P102	P108	P109	P110	P112
000b	P1n Output Data (initial)			P1n Output Data	P1n Output Data (initial)		
001b	TI04_A/TO04_A	TI07_A/TO07_A	TI06_A/TO06_A	SWDIO (initial)	TI02_A/TO02_A	TI01_A/TO01_A	TI03_A/TO03_A
010b	TI01_B/TO01_B	TI00_C	TO00_C	TI03_B/TO03_B	TXD0_B/ SO00_B	RXD0_B/SI00_B/ SDA00_B	SCK00_B/SCL00_B
011b	RXD0_A/SI00_A/ SDA00_A	TXD0_A/ SO00_A	SCK00_A/SCL00_A	SCK00_C	SDAA0_C	SCLA0_C	SSI00_C
100b	SCLA0_D	SDAA0_D	—	SCLA0_E	—	—	—
101b	—	—	PCLBUZ0_B	PCLBUZ0_D	—	—	—
PMC bit	✓ (AN022)	✓ (AN021)	✓	✓	✓	✓	✓
ISEL bit	IRQ2_A	IRQ3_A	IRQ4_A	—	IRQ4_B	IRQ3_B	IRQ2_B
NCODR bit	✓	✓	✓	✓	✓	✓	✓
PIM bit	✓	✓	✓	✓	✓	✓	✓
PCR bit	✓	✓	✓	✓	✓	✓	✓
20-pin product	✓	✓	✓	✓	✓	✓	✓

✓: Available
—: Setting prohibited

Table 15.12 Register settings for the pin function select configuration (PORT2)

PSEL[2:0] settings	Pin				
	P200	P201	P206	P212	P213
000b	Hi-Z (initial)	P2n Output Data (initial)			
001b	TI06_B	TI05_B/TO05_B	—	TO00_A	TI00_A
010b	SSI00_D	SSI00_B	—	TI03_C/TO03_C	TI02_B/TO02_B
011b	RXD1_B	SCK11_B/SCL11_B	—	RXD1_A	TXD1_A
100b	—	—	—	SI11_A /SDA11_A	SO11_A
101b	—	PCLBUZ0_A	—	SCLA0_B	SDAA0_B
PMC bit	✓	✓	—	✓	✓
ISEL bit	IRQ0_A	IRQ5_B	IRQ1_C	IRQ1_B	IRQ0_B
NCODR bit	—	✓	—	✓	✓
PIM bit	—	✓	—	—	—
PCR bit	—	✓	✓	✓	✓
20-pin product	✓	✓	✓	✓	✓

✓: Available
—: Setting prohibited

Table 15.13 Register settings for the pin function select configuration (PORT3)

PSEL[2:0] settings	Pin
	P300
000b	P300 Output Data
001b	SWCLK (initial)
010b	TI04_B/TO04_B
011b	TXD1_B
100b	SDAA0_E
PMC bit	✓
ISEL bit	—
NCODR bit	✓
PIM bit	✓
PCR bit	✓
20-pin product	✓

✓: Available
—: Setting prohibited

—: 设置禁止

表 15.11 引脚功能选择配置的寄存器设置（PORT1）

PSEL[2:0] 设置	别针						
	P100	P101	P102	P108	P109	P110	P112
000b	P1n 输出数据（初始）			P1n 输出数据	P1n 输出数据（初始）		
001b	TI04_A/TO04_A	TI07_A/TO07_A	TI06_A/TO06_A	SWDIO（初始）	TI02_A/TO02_A	TI01_A/TO01_A	TI03_A/TO03_A
010b	TI01_B/TO01_B	TI00_C	TO00_C	TI03_B/TO03_B	TXD0_B/ SO00_B	RXD0_B/SI00_B/ SDA00_B	SCK00_B/SCL00_B
011b	RXD0_A/SI00_A/ SDA00_A	TXD0_A/ SO00_A	SCK00_A/SCL00_A	SCK00_C	SDAA0_C	SCLA0_C	SSI00_C
100b	SCLA0_D	SDAA0_D	—	SCLA0_E	—	—	—
101b	—	—	PCLBUZ0_B	PCLBUZ0_D	—	—	—
PMC 比特	✓ (AN022)	✓ (AN021)	✓	✓	✓	✓	✓
低位	IRQ2_A	IRQ3_A	IRQ4_A	—	IRQ4_B	IRQ3_B	IRQ2_B
NCODR 位	✓	✓	✓	✓	✓	✓	✓
PIM 位	✓	✓	✓	✓	✓	✓	✓
PCR 位	✓	✓	✓	✓	✓	✓	✓
20针产品	✓	✓	✓	✓	✓	✓	✓

✓: 可用
—: 设置禁止

表 15.12 引脚功能选择配置的寄存器设置（PORT2）

PSEL[2:0] 设置	别针				
	P200	P201	P206	P212	P213
000b	高阻抗（初始）	P2n 输出数据（初始）			
001b	TI06_B	TI05_B/TO05_B	—	TO00_A	TI00_A
010b	SSI00_D	SSI00_B	—	TI03_C/TO03_C	TI02_B/TO02_B
011b	RXD1_B	SCK11_B/SCL11_B	—	RXD1_A	TXD1_A
100b	—	—	—	SI11_A /SDA11_A	SO11_A
101b	—	PCLBUZ0_A	—	SCLA0_B	SDAA0_B
PMC 比特	✓	✓	—	✓	✓
低位	IRQ0_A	IRQ5_B	IRQ1_C	IRQ1_B	IRQ0_B
NCODR 位	—	✓	—	✓	✓
PIM 位	—	✓	—	—	—
PCR 位	—	✓	✓	✓	✓
20针产品	✓	✓	✓	✓	✓

✓: 可用
—: 设置禁止

表 15.13 引脚功能选择配置的寄存器设置（PORT3）

PSEL[2:0] 设置	别针
	P300
000b	P300 输出数据
001b	SWCLK（初始）
010b	TI04_B/TO04_B
011b	TXD1_B
100b	SDAA0_E
PMC 比特	✓
低位	—
NCODR 位	✓
PIM 位	✓
PCR 位	✓
20针产品	✓

✓: 可用
—: 设置禁止

16. Timer Array Unit (TAU)

16.1 Overview

The timer array unit of this device has one unit with eight 16-bit timers.

Each 16-bit timer is called a channel and can be used as an independent timer. In addition, two or more channels can be used to create a High functional timer.

Figure 16.1 shows the channel configuration per unit in timer array unit.

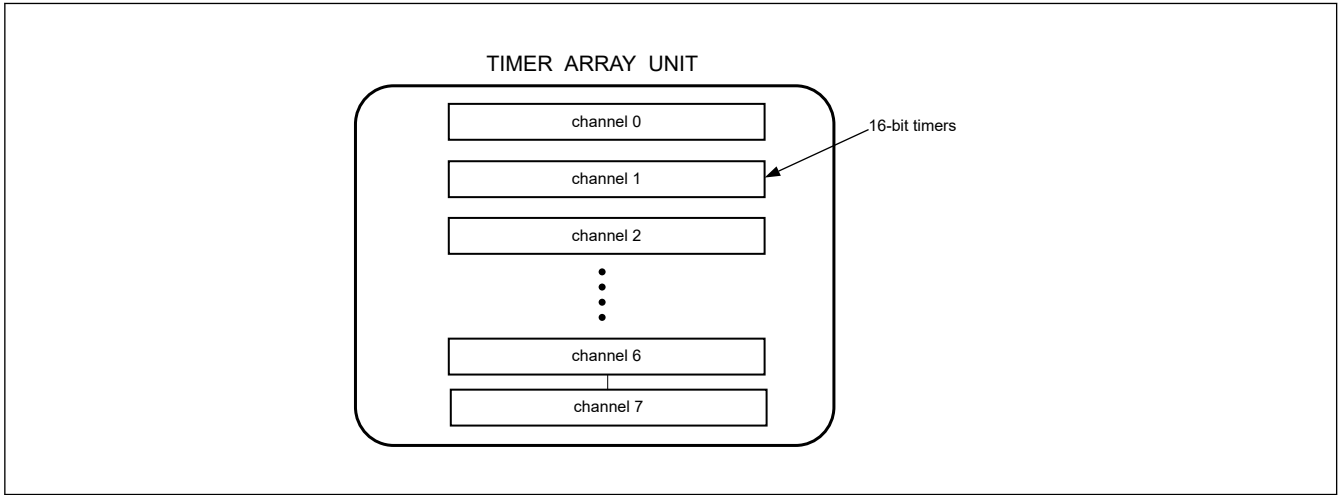


Figure 16.1 Channel configuration per unit

It is possible to use the 16-bit timer of channels 1 and 3 of unit 0 as two 8-bit timers (higher and lower). The functions that can use channels 1 and 3 as 8-bit timers are as follows:

- Interval timer (upper and lower 8-bit timer) and square wave output (lower 8-bit timer only)
- External event counter (lower 8-bit timer only)
- Delay counter (lower 8-bit timer only)

Channel 7 of unit 0 can be used to realize LIN-bus communication operating in combination with UART0 of the serial array unit.

Table 16.1 lists the TAU functions and Figure 16.2 to Figure 16.11 show each functional image.

16.定时器阵列单元（TAU）

16.1 概述

该设备的定时器阵列单元包含一个单元，该单元有八个 16 位定时器。

每个 16 位定时器称为一个通道，可以作为独立的定时器使用。此外，两个或多个通道可以组合成一个高阶功能定时器。

图 16.1 显示了定时器阵列单元中每个单元的通道配置。

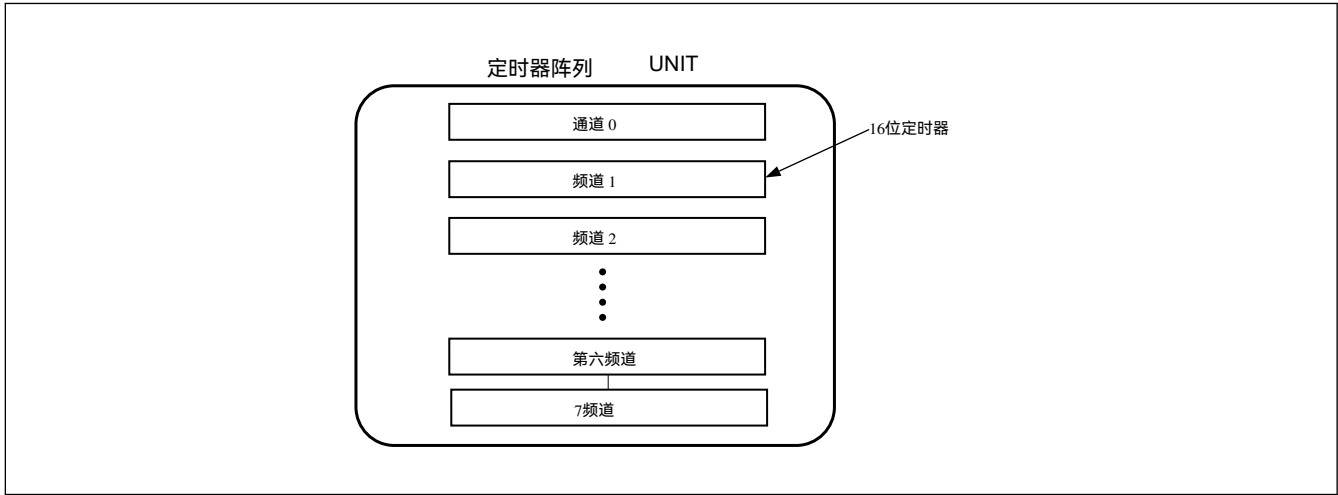


图 16.1 每单元通道配置

可以将单元 0 的通道 1 和通道 3 的 16 位定时器用作两个 8 位定时器（高电平和低电平）。可以使用通道 1 和通道 3 作为 8 位定时器的函数如下：

- 间隔定时器（高8位和低8位定时器）和方波输出（仅低8位定时器）
- 外部事件计数器（仅限低 8 位定时器）
- 延迟计数器（仅限低 8 位定时器）

单元 0 的通道 7 可用于与串行阵列单元的 UART0 配合使用，实现 LIN 总线通信。

表 16.1 列出了 TAU 函数，图 16.2 至图 16.11 显示了每个功能图像。

Table 16.1 TAU functions

Parameter		Description
Independent channel operation function *1	Interval timer	Each timer of a unit can be used as a reference timer that generates an interrupt (TAU0_TMI0n) at fixed intervals.
	Square wave output	A toggle operation is performed each time TAU0_TMI0n interrupt is generated and a square wave with a duty factor of 50% is output from a timer output pin (TO0n).
	External event counter	Each timer of a unit can be used as an event counter that generates an interrupt when the number of the valid edges of a signal input to the timer input pin (TI0n) has reached a specific value.
	Divider function (only channel 0 of unit 0)	A clock input from a timer input pin (TI00) is divided and output from an output pin (TO00).
	Input pulse interval measurement	Counting is started by the valid edge of a pulse signal input to a timer input pin (TI0n). The count value of the timer is captured at the valid edge of the next pulse. In this way, the interval of the input pulse can be measured.
	Measurement of high- or low-level width of input signal	Counting is started by a single edge of the signal input to the timer input pin (TI0n), and the count value is captured at the other edge. In this way, the high-level or low-level width of the input signal can be measured.
	Delay counter	Counting is started at the valid edge of the signal input to the timer input pin (TI0n), and an interrupt is generated after any delay period.
Simultaneous channel operation function *2	One-shot pulse output	Two channels are used as a set to generate a one-shot pulse with a specified output timing and a specified pulse width.
	PWM (Pulse Width Modulation) output	Two channels are used as a set to generate a pulse with a specified period and a specified duty factor.
	Multiple PWM (Pulse Width Modulation) output	By extending the PWM function and using one master channel and two or more slave channels, up to seven types of PWM signals that have a specific period and a specified duty factor can be generated.
8-bit timer operation function (channels 1 and 3 only) *3		The 8-bit timer operation function makes it possible to use a 16-bit timer channel in a configuration consisting of two 8-bit timer channels.
LIN-bus supporting function (channel 7 of unit 0 only) *4	Detection of wakeup signal	The timer starts counting at the falling edge of a signal input to the serial data input pin (RXD0) of UART0 and the count value of the timer is captured at the rising edge. In this way, a low-level width can be measured. If the low-level width is greater than a specific value, it is recognized as a wakeup signal.
	Detection of break field	The timer starts counting at the falling edge of a signal input to the serial data input pin (RXD0) of UART0 after a wakeup signal is detected, and the count value of the timer is captured at the rising edge. In this way, a low-level width is measured. If the low-level width is greater than a specific value, it is recognized as a break field.
	Measurement of pulse width of sync field	After a break field is detected, the low-level width and high-level width of the signal input to the serial data input pin (RXD0) of UART0 are measured. From the bit interval of the sync field measured in this way, a baud rate is calculated.

- Note 1. This function can be used without being affected by the operation mode of other channels. For details, see [section 16.7. Independent Channel Operation Function of Timer Array Unit](#).
- Note 2. This function can be used to combine a master channel (a reference timer mainly controlling the cycle) and slave channels (timers operating according to the master channel). For details, see [section 16.8. Simultaneous Channel Operation Function of Timer Array Unit](#).
- Note 3. There are several rules for using 8-bit timer operation function. For details, see [section 16.3.2. Basic Rules of 8-bit Timer Operation Function \(Channels 1 and 3 only\)](#).

表 16.1TAU 函数

范围		描述
独立通道操作功能*1	间隔定时器	单元中的每个定时器都可以用作参考定时器，以固定的时间间隔产生中断（TAU0_TMI0n）。
	方波输出	每次产生 TAU0_TMI0n 中断时，都会执行翻转操作，并从定时器输出引脚（TO0n）输出占空比为50% 的方波。
	外部事件计数器	单元中的每个定时器都可以用作事件计数器，当输入到定时器输入引脚（TI0n）的信号的有效边沿数达到特定值时，该计数器会产生中断。
	分频器功能（仅限单元的通道 0）	来自定时器输入引脚（TI00）的时钟输入信号被分频后从输出引脚（TO00）输出。
	输入脉冲间隔测量	计数由输入到定时器输入引脚（TI0n）的脉冲信号的有效边沿启动。定时器的计数值在下一个脉冲的有效边沿被捕获。通过这种方式，可以测量输入脉冲的间隔。
	测量输入信号的高电平或低电平宽度	计数由定时器输入引脚（TI0n）输入信号的单个边沿启动，计数值在下一个边沿被捕获。通过这种方式，可以测量输入信号的高电平或低电平宽度。
	延迟计数器	计数从定时器输入引脚（TI0n）的信号输入的有效边沿开始，并在任何延迟期后产生中断。
同时通道操作功能*2	单脉冲输出	两个通道组合在一起，可以产生具有指定输出时间和指定脉冲宽度的单次脉冲。
	PWM（脉冲宽度调制）输出	两个通道组成一个系统，用于生成具有指定周期和指定占空比的脉冲。
	多频 PWM（脉冲宽度调制）调制）输出	通过扩展 PWM 功能，并使用一个主通道和两个或多个从通道，可以生成多达七种具有特定周期和指定占空比的 PWM 信号。
8 位定时器操作功能（仅限通道 1 和 3） *3		8 位定时器操作功能使得在由两个 8 位定时器通道组成的配置中使用 16 位定时器通道成为可能。
LIN总线支持功能（仅限0号单元的通道7） *4	唤醒信号检测	定时器在输入到 UART0 串行数据输入引脚（RXD0）的信号下降沿开始计数，并在上升沿捕获计数值。这样就可以测量低电平宽度。如果低电平宽度大于特定值，则将其识别为唤醒信号。
	检测断裂场	定时器在检测到唤醒信号后，从输入到 UART0 串行数据输入引脚（RXD0）的信号下降沿开始计数，并在上升沿捕获计数值。这样就测量出了低电平宽度。如果低电平宽度大于特定值，则判定为断点。
	同步场脉冲宽度的测量	检测到中断场后，测量输入到 UART0 串行数据输入引脚（RXD0）的信号的低电平宽度和高电平宽度。根据以此方式测量的同步场的位置间隔，计算波特率。

- 注1：此功能的使用不受其他通道运行模式的影响。详情请参见16.7节。[定时器阵列单元的独立通道操作功能](#)。
- 注2. 此功能可用于组合主通道（主要控制周期的参考定时器）和从通道（根据主通道运行的定时器）。详情请参见第16.8节“定时器阵列单元的同步通道操作功能”。
- 注3. 使用8位定时器操作功能有若干规则。详情请参见16.3.2节。8位定时器操作功能的基本规则（仅限通道1和3）。

Note 4. Timer array unit is used to check whether signals received in LIN-bus communication match the LIN-bus communication format. For details about setting up the operations used to implement the LIN-bus, see [section 16.2.15. ISC : Input Switch Control Register](#) and [section 16.7.5. Operation for Input Signal High- or Low-Level Width Measurement](#).

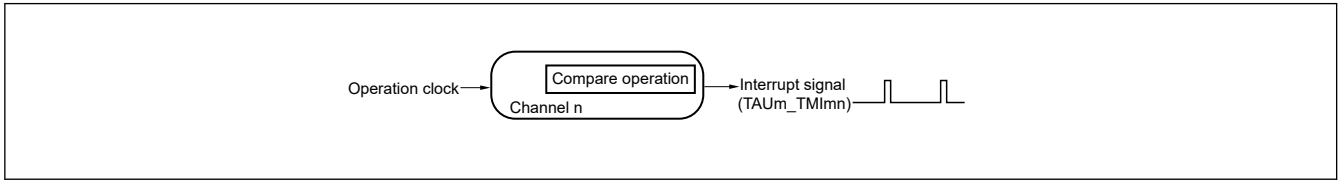


Figure 16.2 Functional image of interval timer

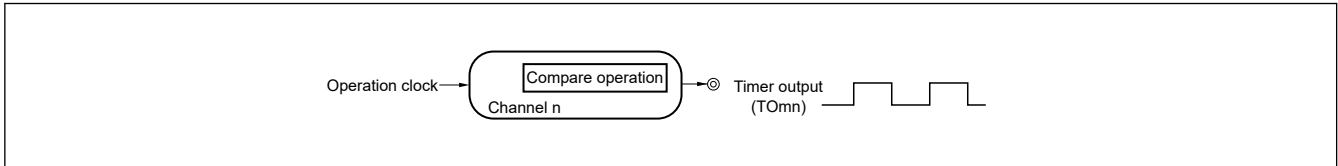


Figure 16.3 Functional image of square wave output

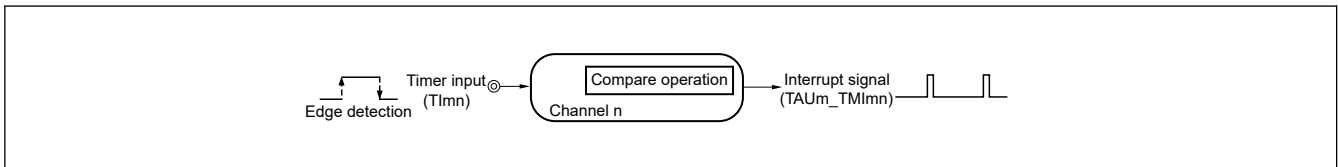


Figure 16.4 Functional image of external event counter

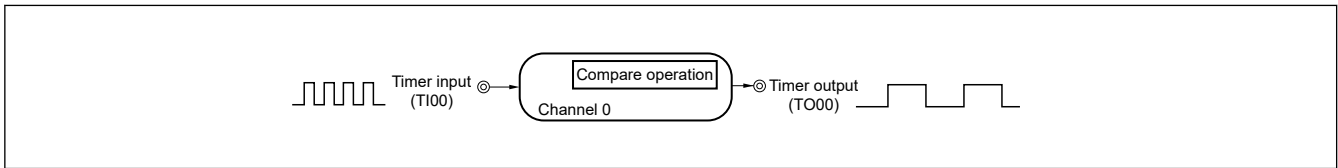


Figure 16.5 Functional image of divider function

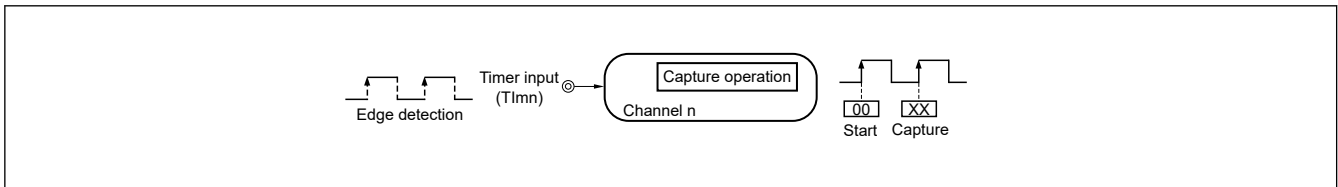


Figure 16.6 Functional image of input pulse interval measurement

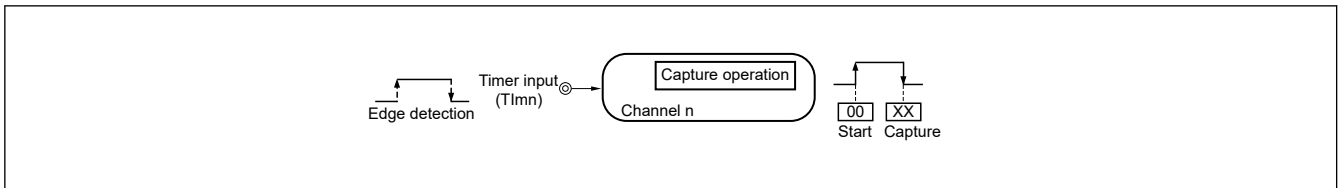


Figure 16.7 Functional image of measurement of high- or low-level width of input signal

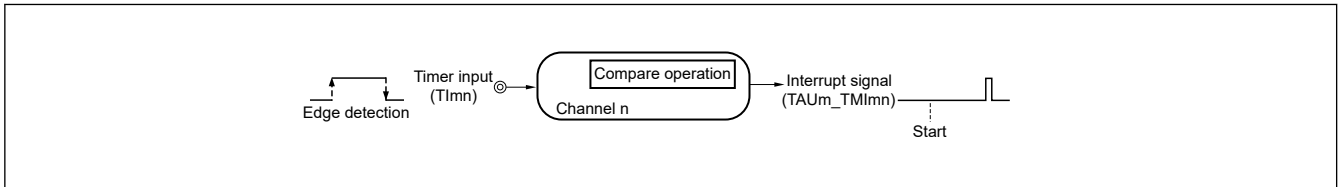


Figure 16.8 Functional image of delay counter

注4. 定时器阵列单元用于检查LIN总线通信中接收到的信号是否符合LIN总线通信格式。有关LIN总线实现操作的详细信息，请参见16.2.15节“ISC：输入开关控制寄存器”和16.7.5节“输入信号高电平或低电平宽度测量操作”。

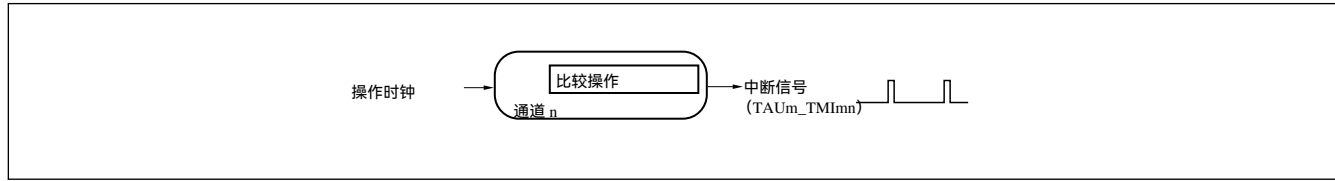


图 16.2 间隔定时器的功能示意图

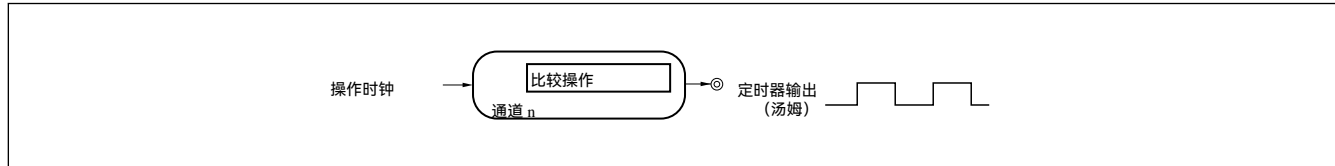


图 16.3 方波输出的功能图像

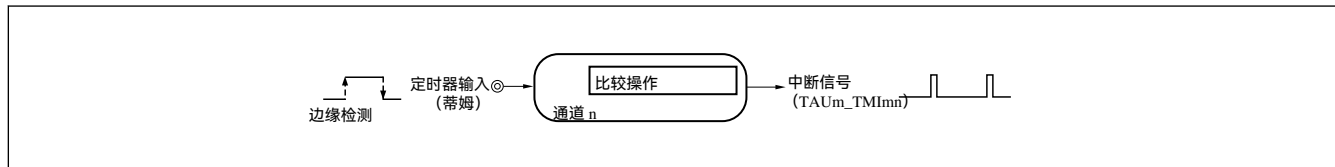


图 16.4 外部事件计数器的功能图

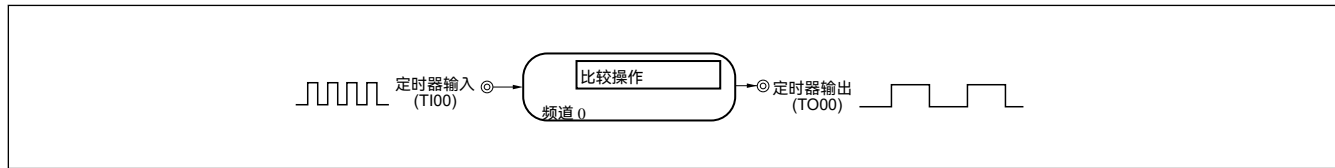


图 16.5 除法器函数的功能图像

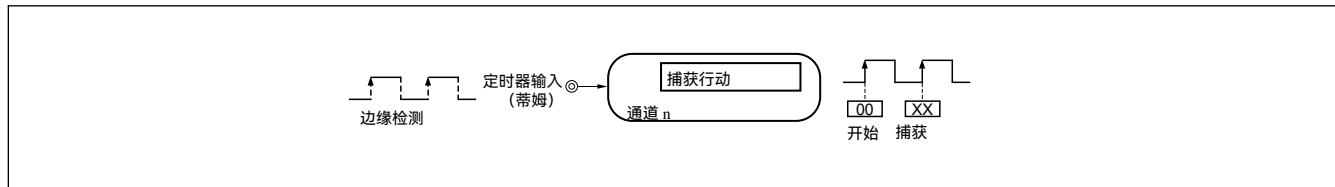


图 16.6 输入脉冲间隔测量的功能图像

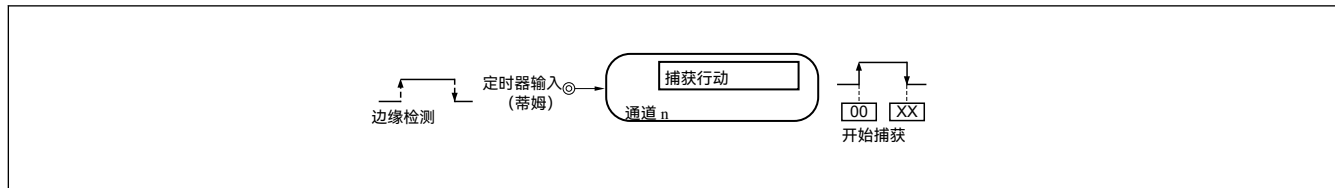


图 16.7 输入信号高低电平宽度测量的功能图像

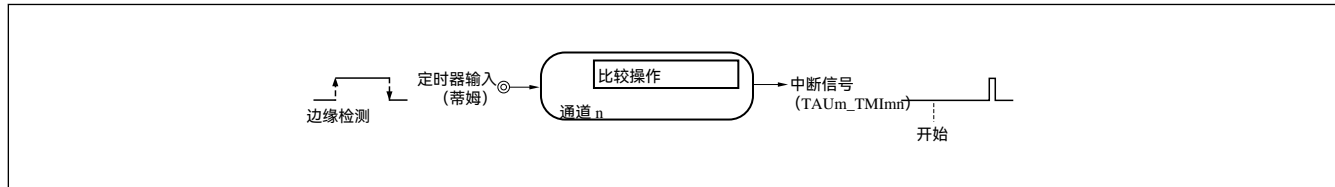


图 16.8 延迟计数器的功能图像

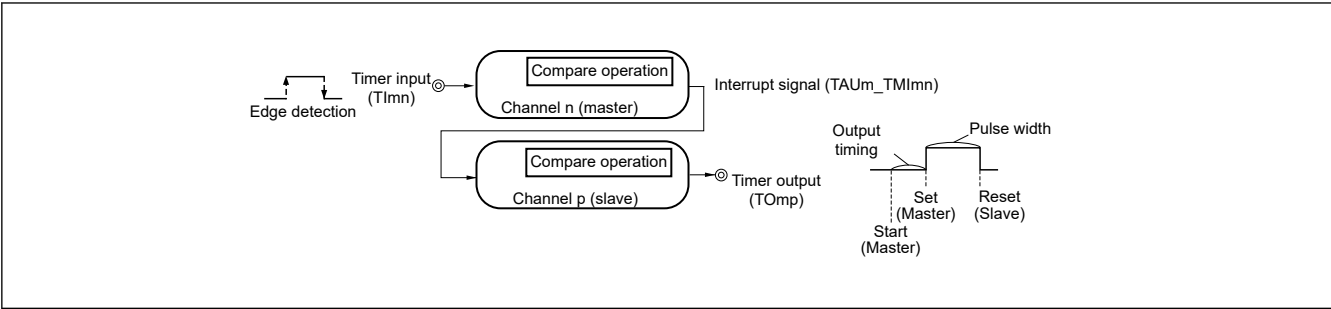


Figure 16.9 Functional image of one-shot pulse output

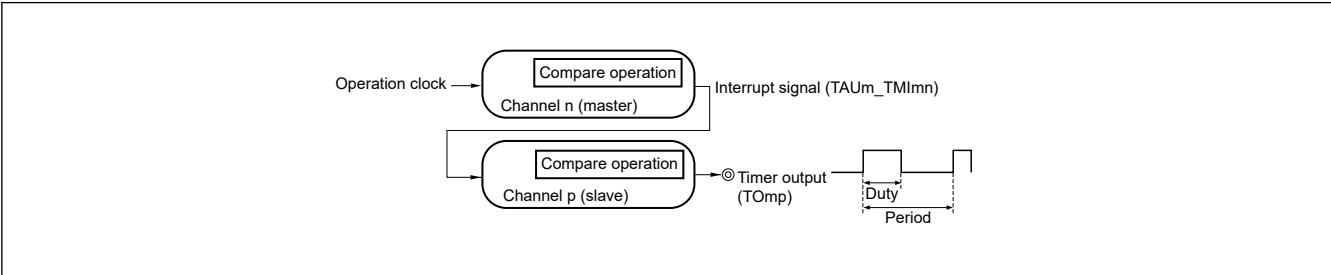


Figure 16.10 Functional image of PWM output

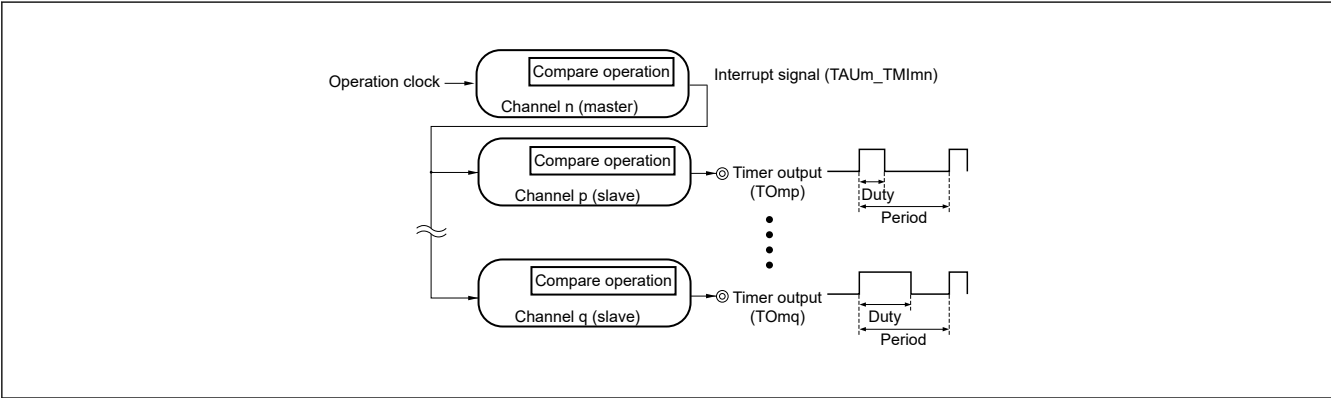


Figure 16.11 Functional image of multiple PWM output

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 7), p, q: Slave channel number (n < p < q ≤ 7)

Timer array unit includes the hardware shown in Table 16.2.

Table 16.2 Configuration of timer array unit

Item	Configuration
Timer/counter	Timer counter register 0n (TCR0n)
Register	Timer data register 0n (TDR0n)
Timer input	TI00 to TI07 pins, RXD0 pin (for LIN-bus)
Timer output	TO00 to TO07 pins, output controller

Figure 16.12 shows the block diagram of the timer array unit. Figure 16.13 to Figure 16.18 show a block diagram for each channel.

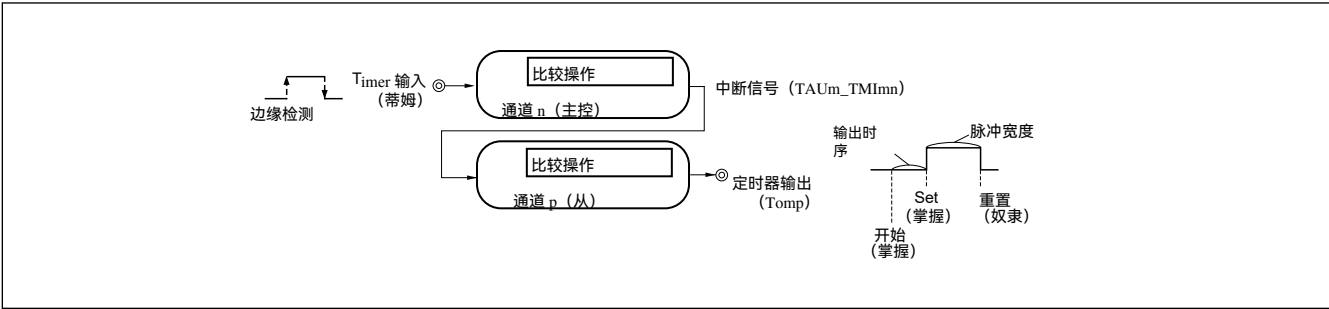


图 16.9 单脉冲输出的功能图像

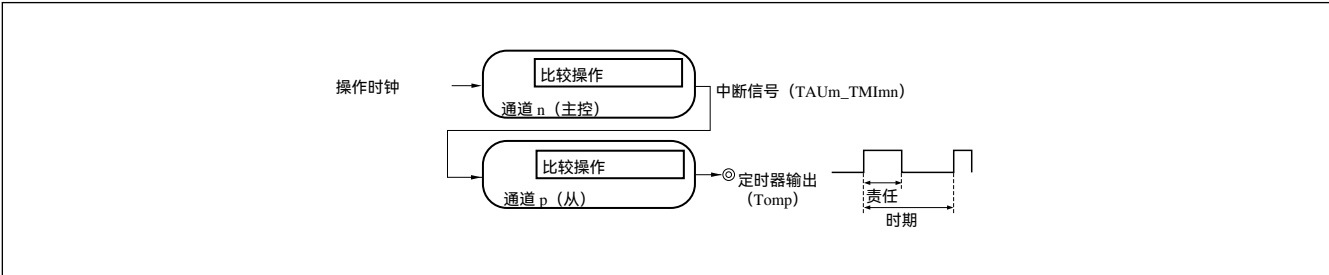


图 16.10 PWM 输出的功能图像

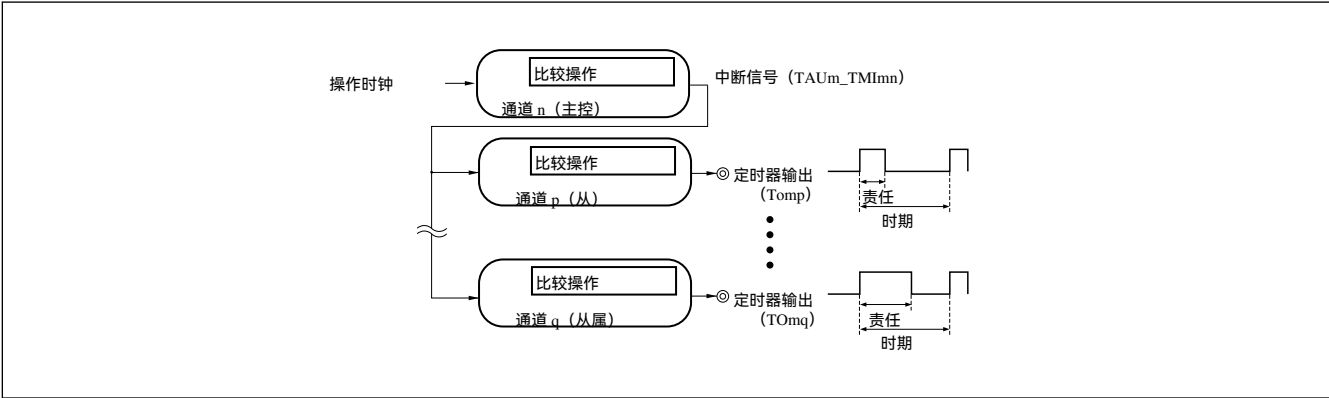


图 16.11 多路 PWM 输出的功能示意图

注: m: 单元编号(m = 0) n: 通道编号(n = 0至7) , p, q: 从通道编号(n < p < q ≤ 7)

定时器阵列单元包括表 16.2 所示的硬件。

表 16.2 定时器阵列单元配置

物品	配置
计时器/计数器	定时器计数器寄存器 0n (TCR0n)
登记	定时器数据寄存器 0n (TDR0n)
定时器输入	TI00 至 TI07 引脚, RXD0 引脚 (用于 LIN 总线)
定时器输出	TO00 至 TO07 引脚, 输出控制器

图 16.12 显示了定时器阵列单元的框图。图 16.13 至图 16.18 显示了每个通道的框图。

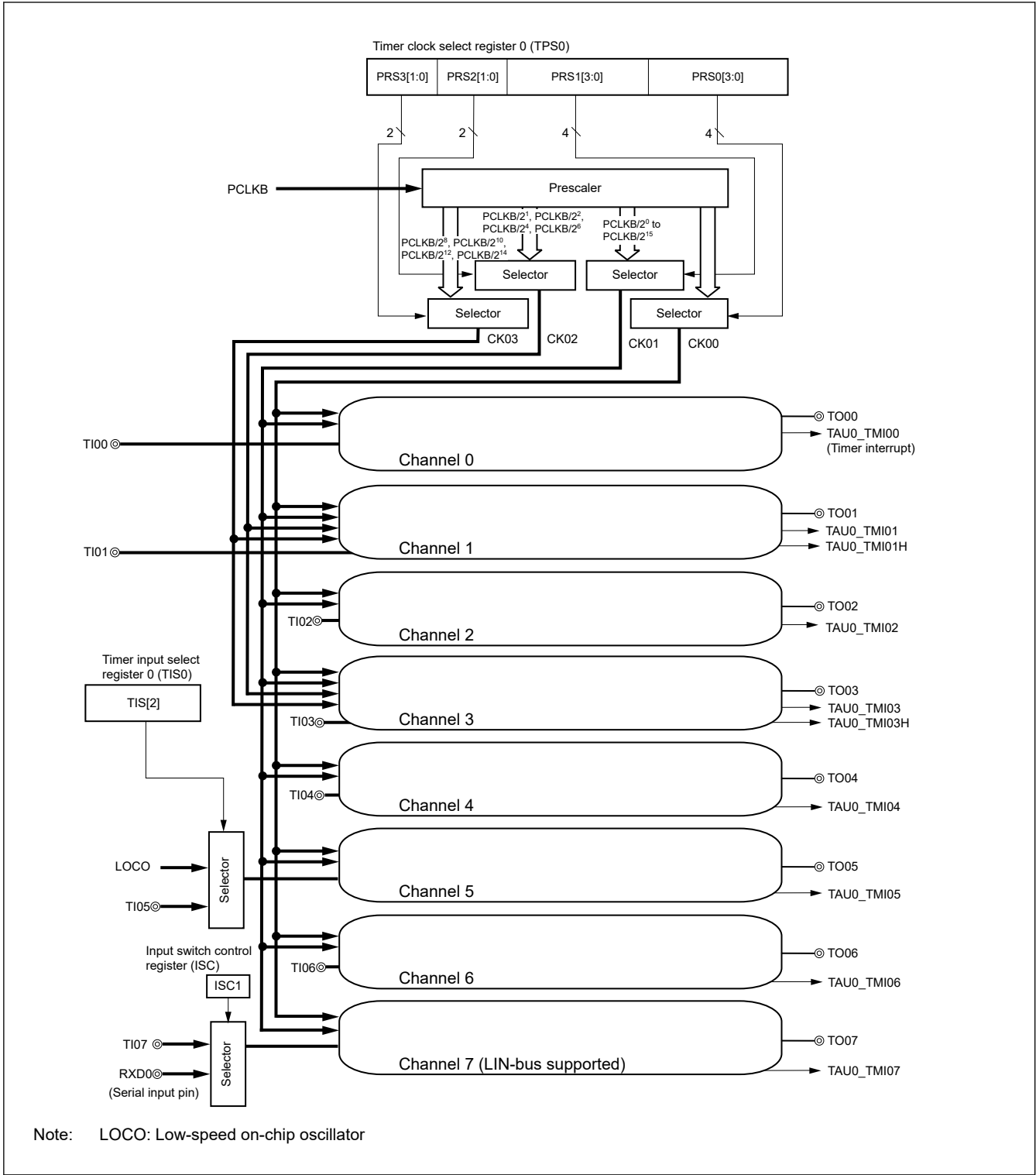


Figure 16.12 Entire configuration of timer array unit 0

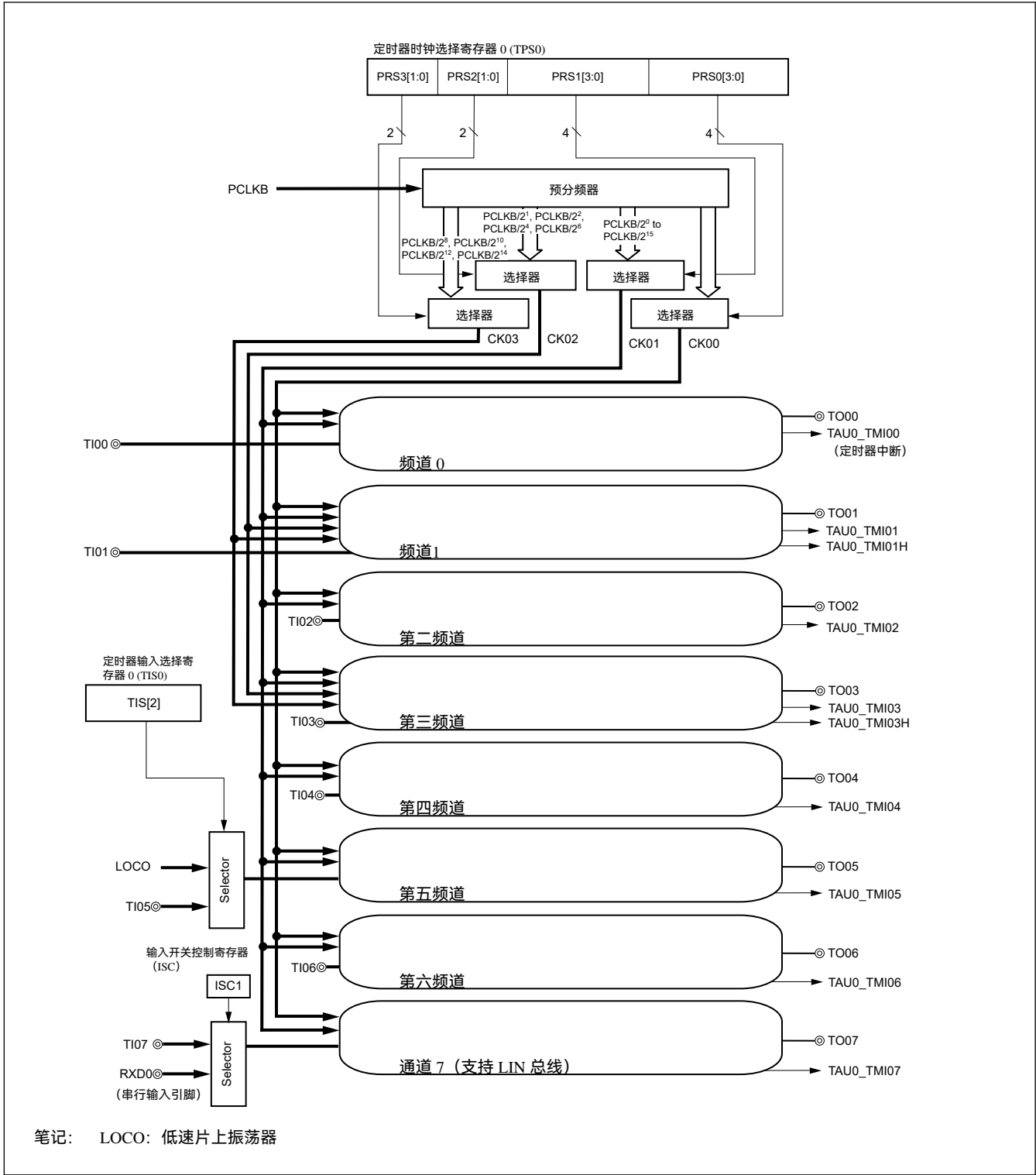


图 16.12 定时器阵列单元 0 的完整配置

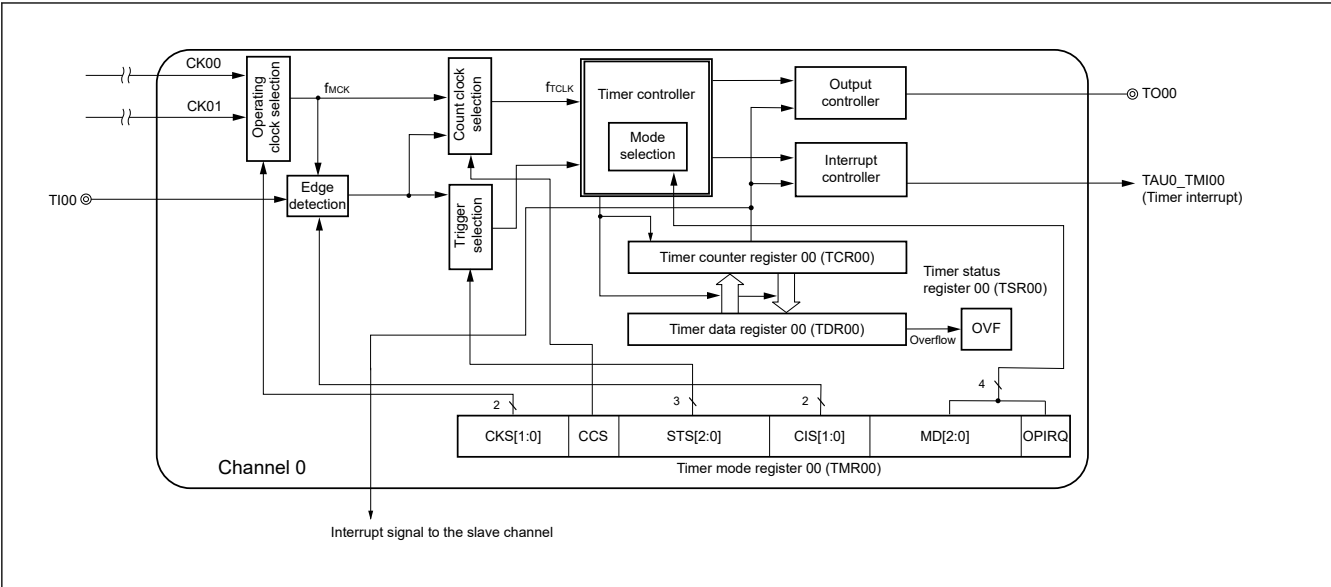


Figure 16.13 Internal block diagram of channel 0 of timer array unit 0

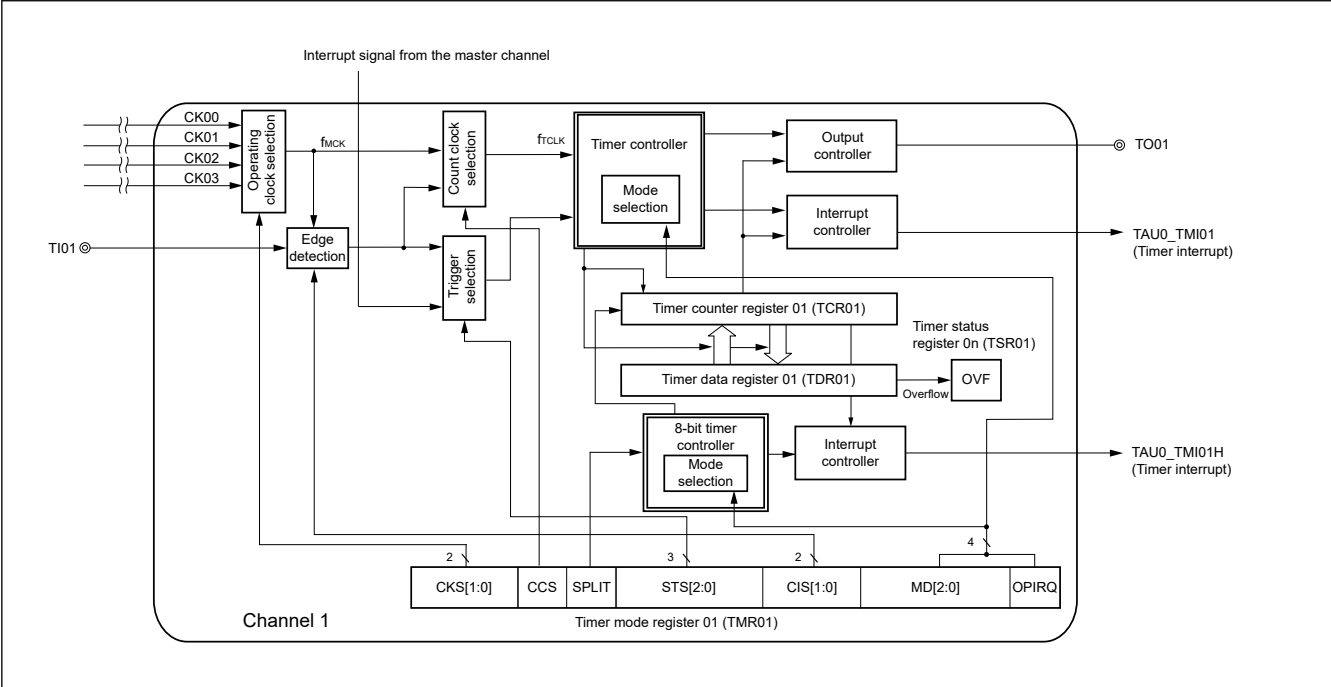


Figure 16.14 Internal block diagram of channel 1 of timer array unit 0

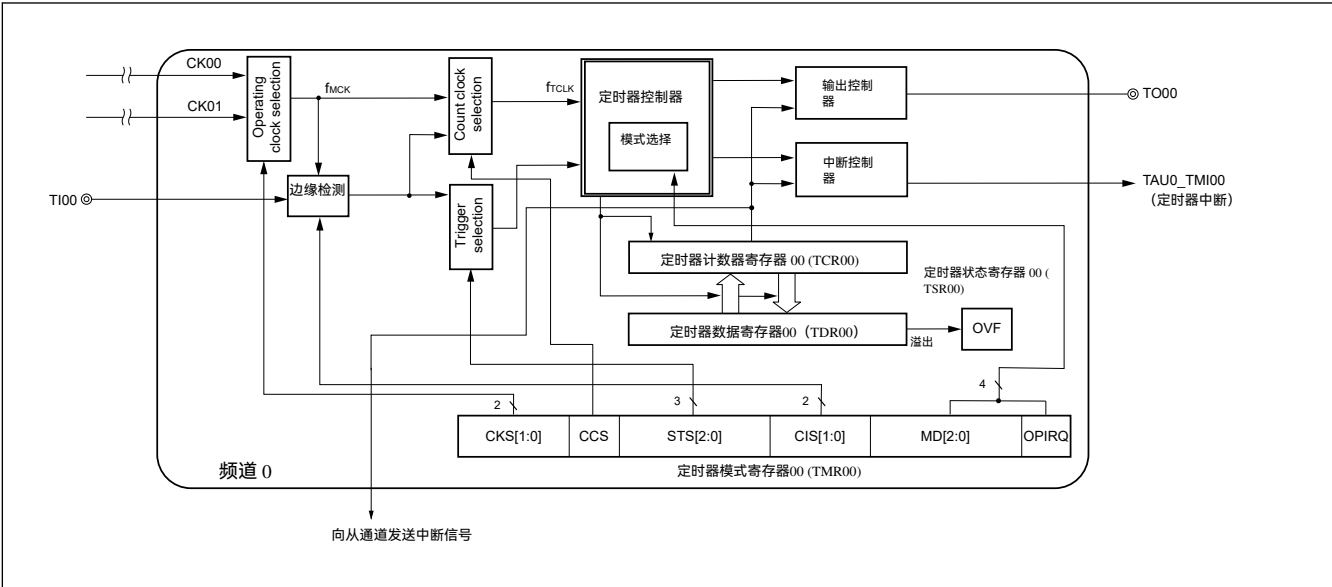


图 16.13 定时器阵列单元 0 的通道 0 的内部框图

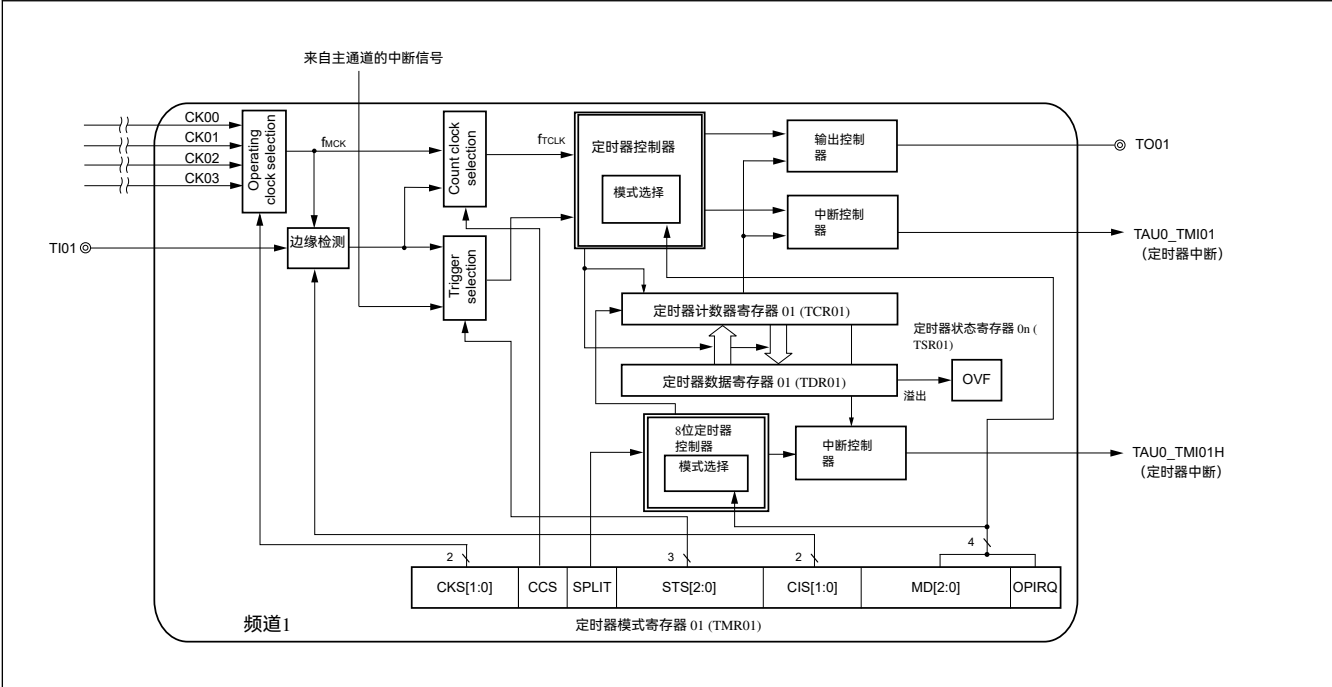


图 16.14 定时器阵列单元 0 的通道 1 的内部框图

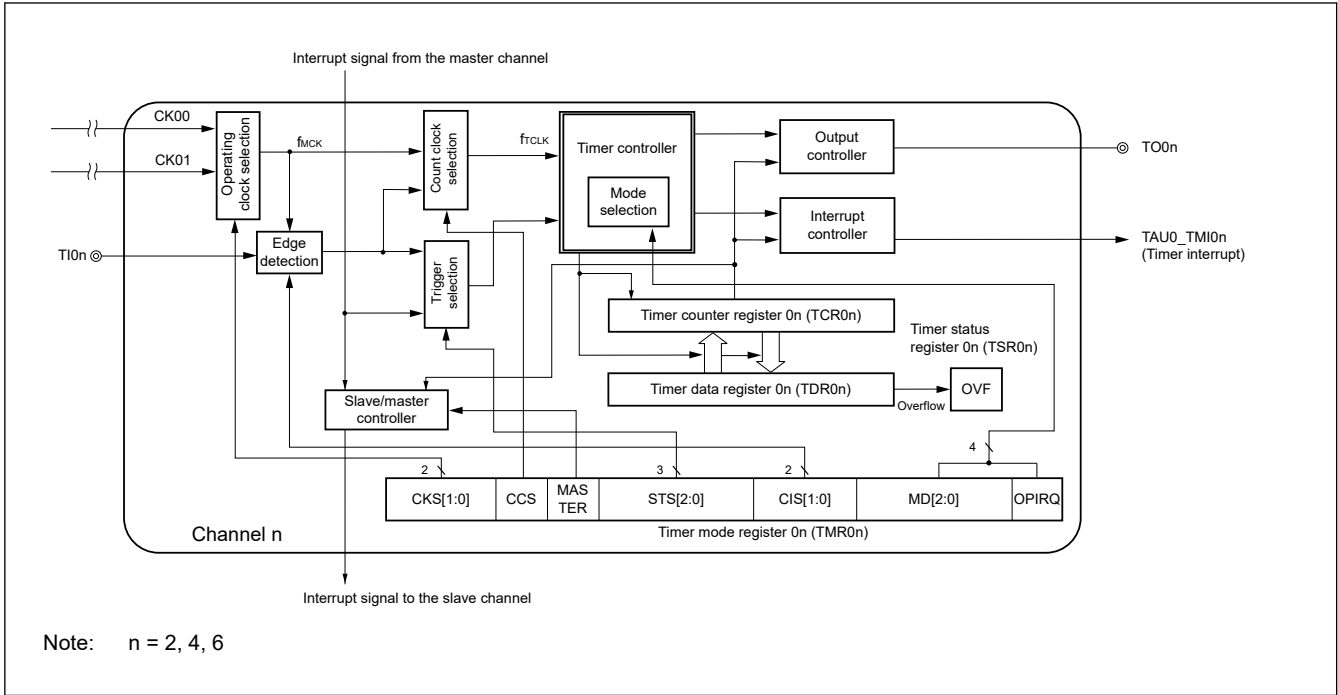


Figure 16.15 Internal block diagram of channels 2, 4, and 6 of timer array unit 0

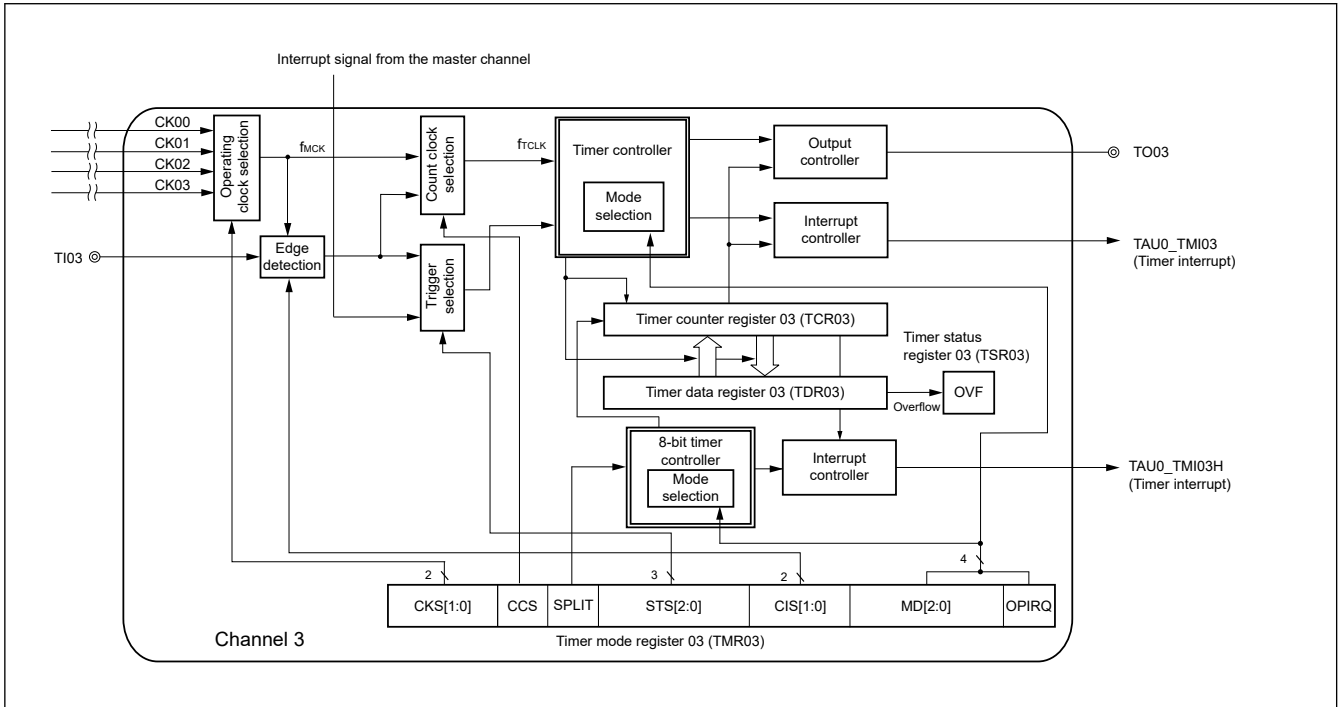


Figure 16.16 Internal block diagram of channel 3 of timer array unit 0

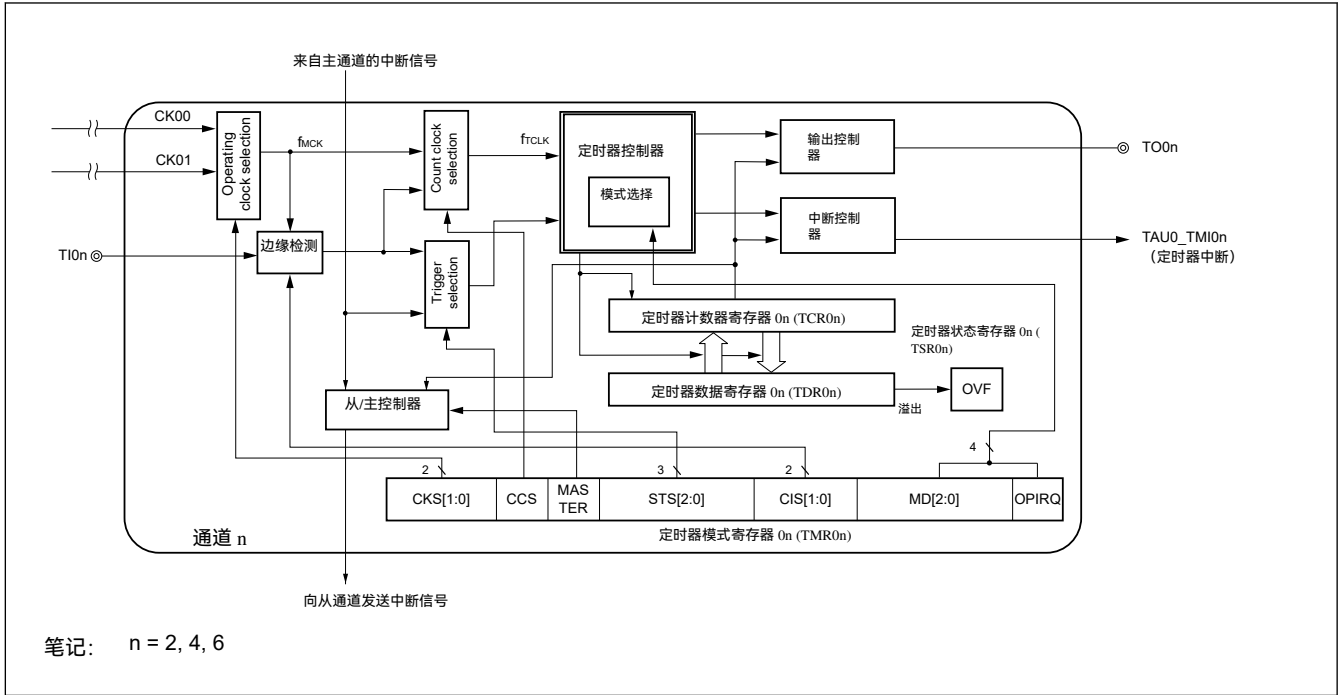


图 16.15 定时器阵列单元 0 的通道 2、4 和 6 的内部框图

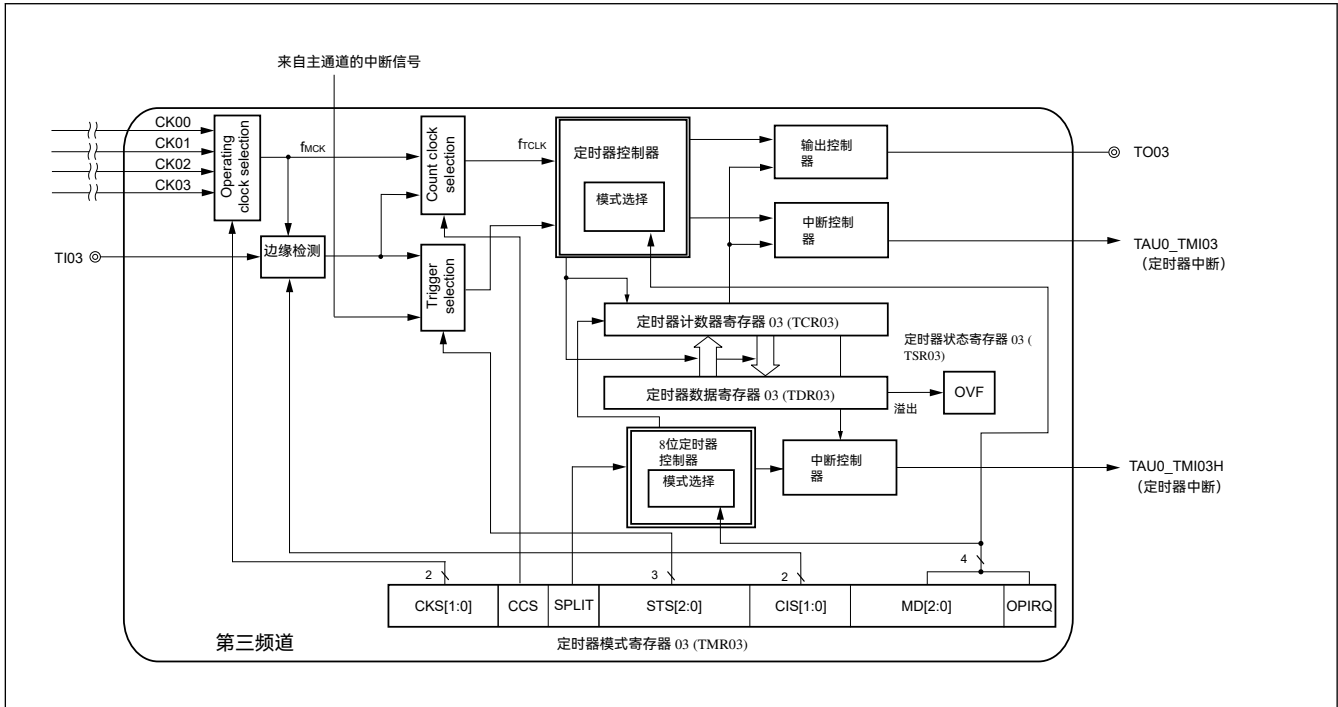


图 16.16 定时器阵列单元 0 的通道 3 的内部框图

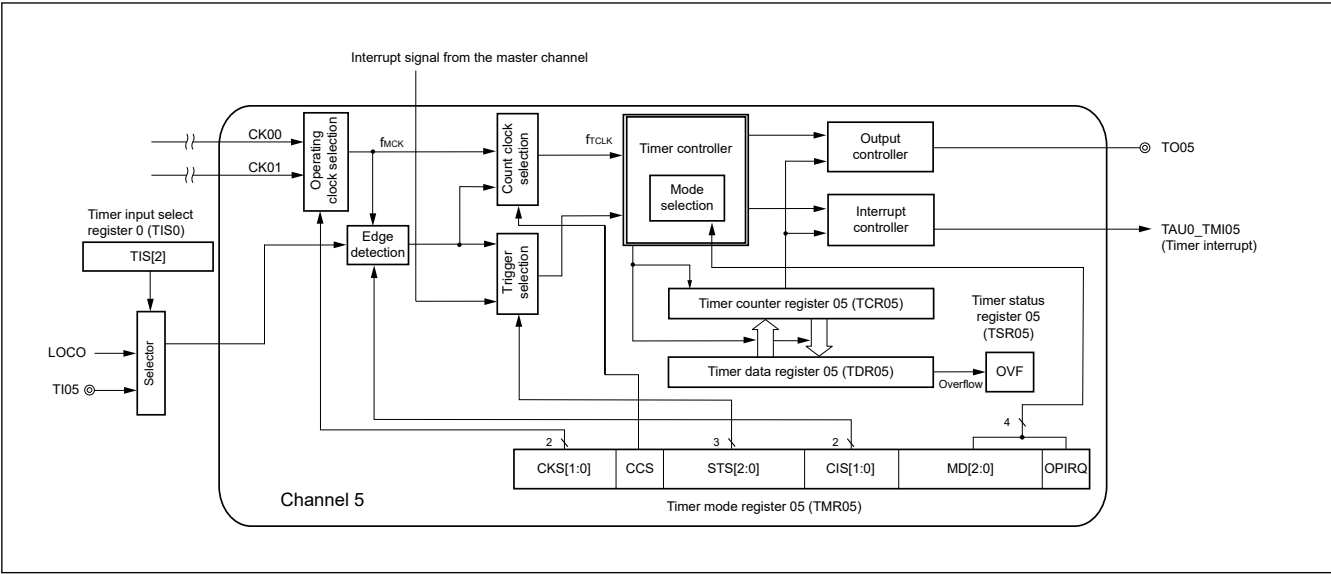


Figure 16.17 Internal block diagram of channel 5 of timer array unit 0

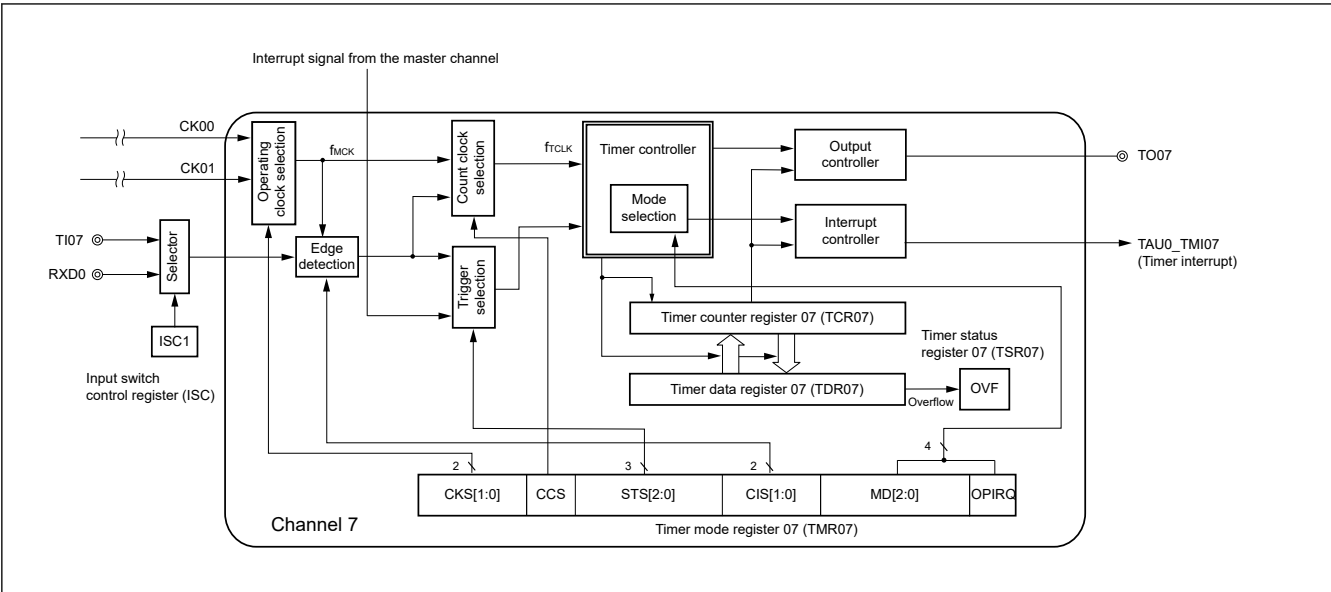


Figure 16.18 Internal block diagram of channel 7 of timer array unit 0

16.2 Register Descriptions

16.2.1 TCR0n : Timer Counter Register 0n (n = 0 to 7)

Base address: TAU = 0x400A_2600
Offset address: 0x0100 + 0x2 × n

Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	n/a															
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit	Symbol	Function	R/W
15:0	n/a	16-bit Clock Count Result for Unit m and Channel n	R

The TCR0n register is a 16-bit read-only register used to count clocks.

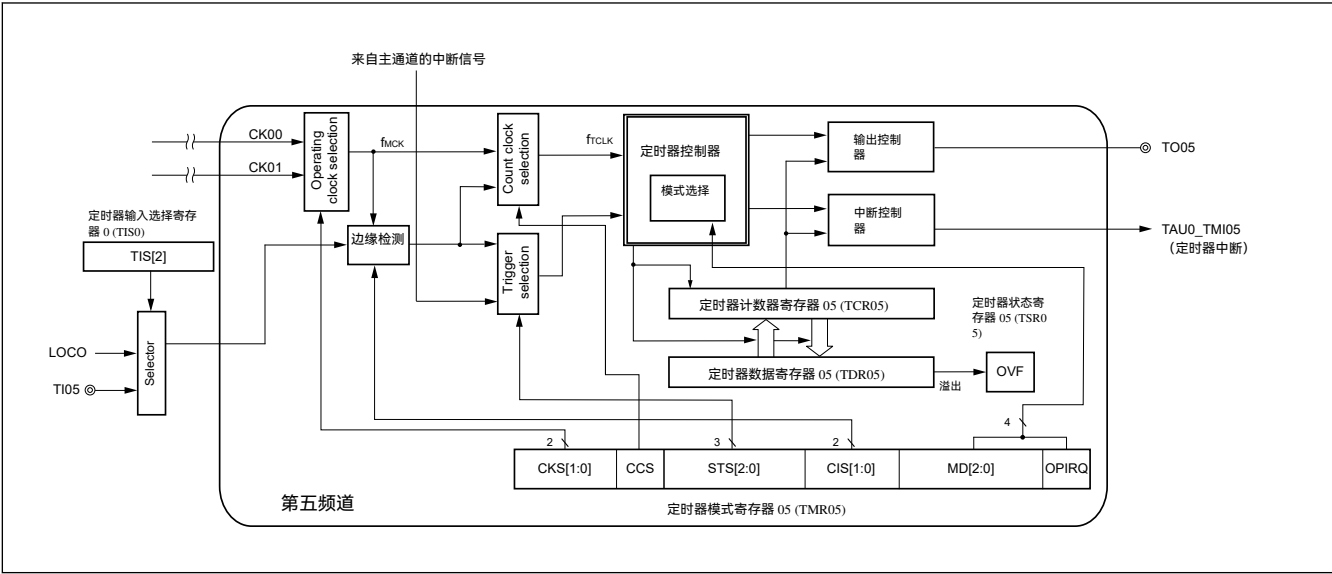


图 16.17 定时器阵列单元 0 的通道 5 的内部框图

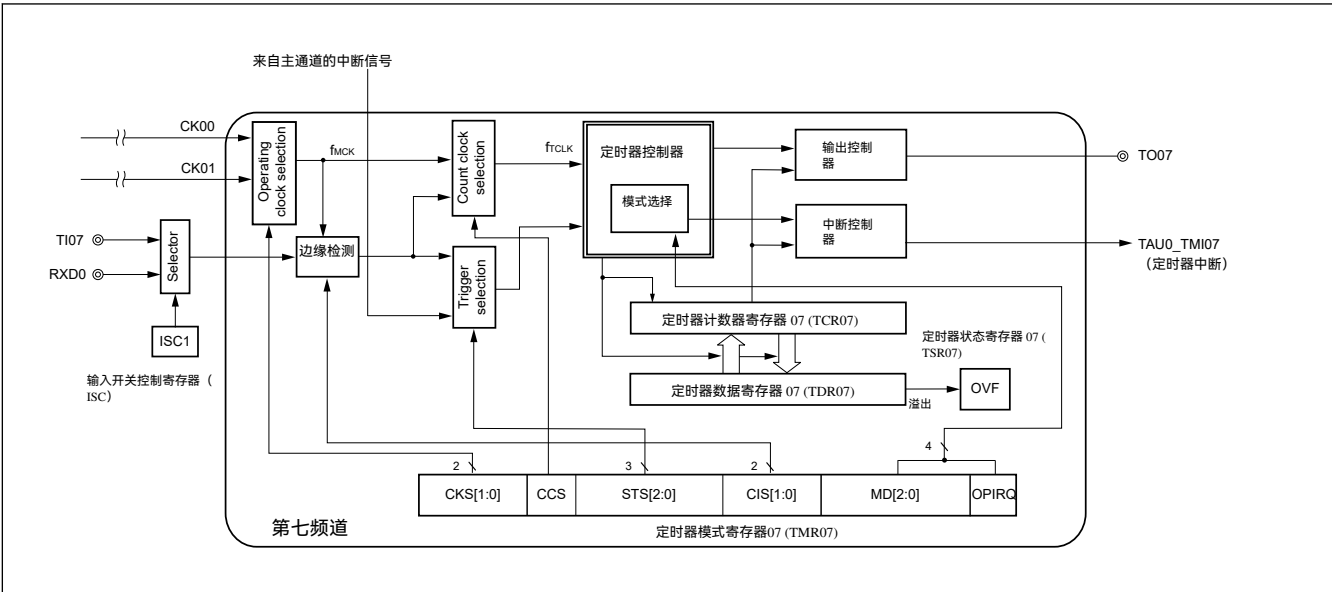


图 16.18 定时器阵列单元 0 的通道 7 的内部框图

16.2 寄存器描述

16.2.1 TCR0n: 定时器计数器寄存器 0n(n = 0至 7)

Base address: TAU = 0x400A_2600
Offset address: 0x0100 + 0x2 × n

位位置:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域:	n/a															
重置后的值:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

少量	象征	功能	R/W
15:0	n/a	单元 m 和通道 n 的 16 位时钟计数结果	R

TCR0n 寄存器是一个 16 位只读寄存器，用于计数时钟。

The value of this counter is incremented or decremented in synchronization with the rising edge of a count clock. Whether the counter is incremented or decremented depends on the operation mode that is selected by the MD[2:0] bits and OPIRQ bit of timer mode register 0n (TMR0n) (see [section 16.2.4. TMR0n : Timer Mode Register 0n \(n = 0, 2, 4, 5, 6, 7\)](#) and [section 16.2.5. TMR0n : Timer Mode Register 0n \(n = 1, 3\).](#))

The count value can be read by reading timer counter register 0n (TCR0n). The count value is set to 0xFFFF in the following cases.

- When the reset signal is generated
- When counting of the slave channel has been completed in the PWM output mode
- When counting of the slave channel has been completed in the delay count mode
- When counting of the master/slave channel has been completed in the one-shot pulse output mode
- When counting of the slave channel has been completed in the multiple PWM output mode

The count value is cleared to 0x0000 in the following cases.

- When the start trigger is input in the capture mode
- When capturing has been completed in the capture mode

Note: The count value is not captured to timer data register 0n (TDR0n) even when the TCR0n register is read.

Note: When reading the TCR0n register, it is necessary to access the TCR0n register with 16-bit width which is the same as its counter size to prevent from mistaking the count value.

The value read from the TCR0n register varies depending on the change to the operation mode and operating state as shown in the [Table 16.3](#).

Table 16.3 Timer counter register 0n (TCR0n) read value in various operation modes

Operation mode	Count mode	Value read from the timer counter register 0n (TCR0n)*1			
		Value when the operation mode is changed after releasing reset	Value when count operation is temporarily stopped (TT0.TT[n] = 1)	Value when the operation mode is changed after count operation was temporarily stopped (TT0.TT[n] = 1)	Value when waiting for a start trigger after one count
Interval timer mode	Countdown	0xFFFF	Value when counting is stopped	Undefined	—
Capture mode	Count-up	0x0000	Value when counting is stopped	Undefined	—
Event counter mode	Countdown	0xFFFF	Value when counting is stopped	Undefined	—
One-count mode	Countdown	0xFFFF	Value when counting is stopped	Undefined	0xFFFF
Capture & one-count mode	Count-up	0x0000	Value when counting is stopped	Undefined	Captured value of TRD0n register + 1

Note 1. This indicates the value read from the TCR0n register when channel n has stopped operating as a timer (TE0.TE[n] = 0) and has been enabled to operate as a counter (TS0.TS[n] = 1). The read value is held in the TCR0n register until the count operation starts.

该计数器的值与计数时钟的上升沿同步递增或递减。计数器是递增还是递减取决于定时器模式寄存器[2的 MD] 位和0n(TMR0n) 的 OPIRQ 位所选择的操作模式（参见 16.2.4 节 TMR0n: 定时器模式寄存器0n (n = 0, 2, 4, 5, 6, 7) 和 16.2.5 节 TMR0n: 定时器模式寄存器0n (n = 1, 3)）。

可以通过读取定时器计数器寄存器 0n (TCR0n) 来读取计数值。在以下情况下，计数值被设置为 0xFFFF。

- 当产生复位信号时
- 当PWM输出模式下从通道计数完成后
- 在延迟计数模式下，当从通道计数完成后
- 当单次脉冲输出模式下主/从通道计数完成后
- 在多通道PWM输出模式下，当从通道计数完成后

在下列情况下，计数值将被清除为0x0000。

- 在捕获模式下输入启动触发信号时
- 在捕获模式下完成捕获后

注意：即使读取了 TCR0n 寄存器，计数值也不会捕获到定时器数据寄存器 0n (TDR0n) 中。

注意：读取 TCR0n 寄存器时，必须以 16 位宽度访问 TCR0n 寄存器，这与它的计数器大小相同，以防止计数值出错。

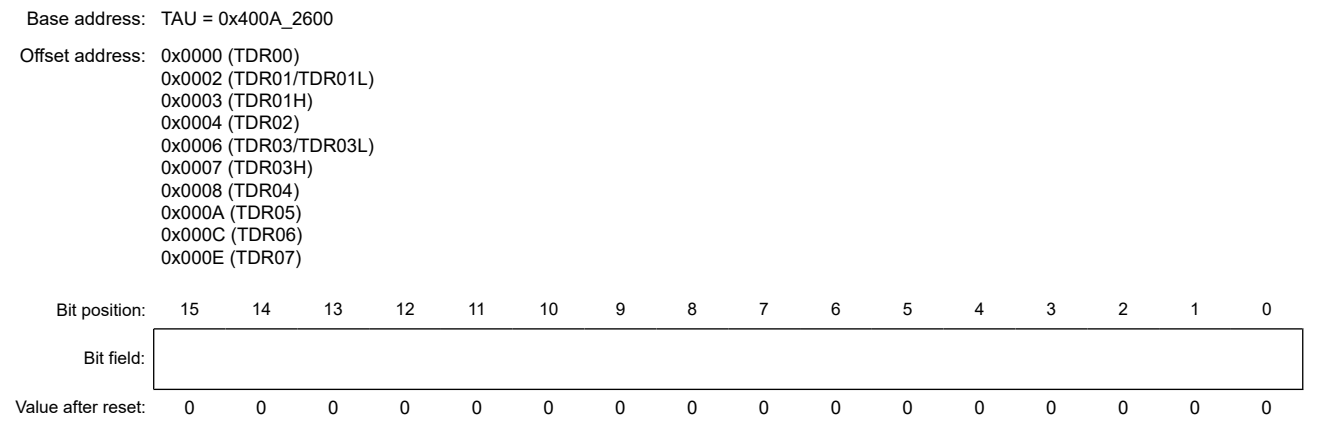
根据表 16.3 所示，从 TCR0n 寄存器读取的值会随着操作模式和操作状态的变化而变化。

表 16.3 定时器计数器寄存器 0n (TCR0n) 在各种操作模式下的读取值

操作模式	计数模式	从定时器计数器寄存器 0n (TCR0n)*1 读取的值			
		复位释放后操作模式改变时的值	计数操作暂时停止时的值(TT0.TT[n] = 1)	计数操作暂时停止后，操作模式改变时的值 (TT0.TT[n] = 1)	等待启动触发器一次计数后的值
间隔定时器模式	倒计时	0xFFFF	计数停止时的值	不明确的	—
捕获模式	计数	0x0000	计数停止时的值	不明确的	—
事件计数器模式	倒计时	0xFFFF	计数停止时的值	不明确的	—
单次计数模式	倒计时	0xFFFF	计数停止时的值	不明确的	0xFFFF
捕获和单次计数模式	计数	0x0000	计数停止时的值	不明确的	的捕获值 TRD0n 寄存器+ 1

注1. 这表示当通道n停止作为定时器(TE0.TE[n] = 0) 运行并被启用作为计数器(TS0.TS[n] = 1)运行时，从TCR0n寄存器读取的值。读取的值将保存在TCR0n寄存器中，直到计数操作开始。

16.2.2 TDR0n/TDR01x/TDR03x : Timer Data Register 0n (n = 0 to 7) (x = L, H)



Bit	Symbol	Function	R/W
15:0	n/a	16-bit Timer Capture Result or Setting Compare Data for Unit m and Channel n	R/W

This is a 16-bit register from which a capture function and a compare function can be selected.

The capture or compare function can be switched by selecting an operation mode by using the TMR0n.MD[2:0] bits and TMR0n.OPIRQ bit of timer mode register 0n (TMR0n).

The value of the TDR0n register can be changed at any time. This register can be read or written in 16-bit units.

In addition, for the TDR01 and TDR03 registers, while in the 8-bit timer mode (when the SPLIT01, SPLIT03 bits of timer mode registers 01 and 03 (TMR01, TMR03) are 1), it is possible to read and write the data in 8-bit units, with TDR01H and TDR03H used as the higher 8 bits, and TDR01L and TDR03L used as the lower 8 bits.

(i) When timer data register 0n (TDR0n) is used as compare register

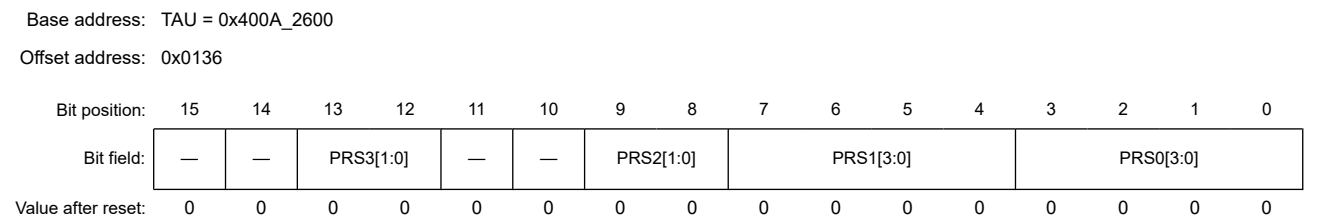
Counting down is started from the value set to the TDR0n register. When the count value reaches 0x0000, an interrupt signal (TAU0_TMI0n) is generated. The TDR0n register holds its value until it is rewritten.

Note: The TDR0n register does not perform a capture operation even if a capture trigger is input, when it is set to the compare function.

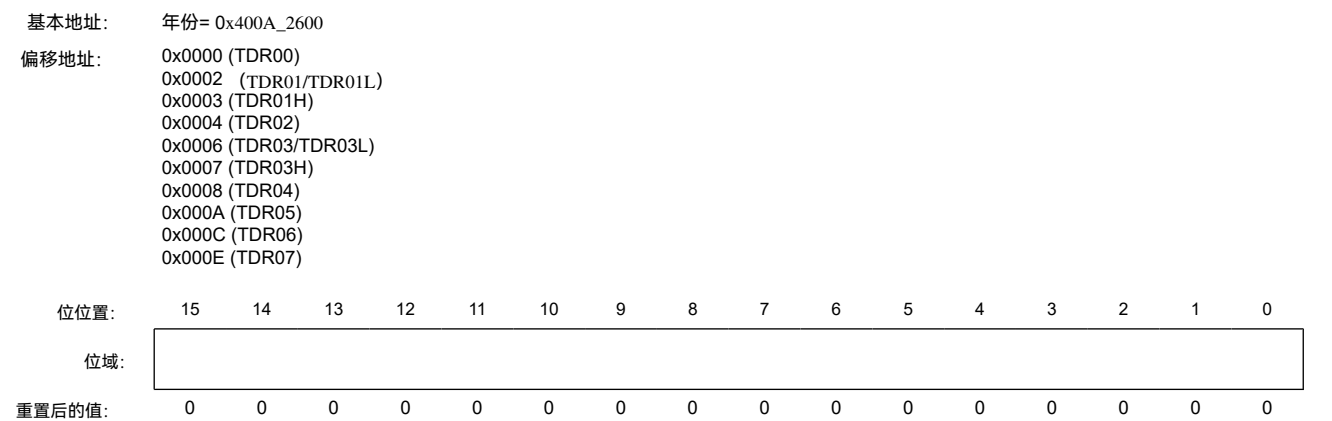
(ii) When timer data register 0n (TDR0n) is used as capture register

The count value of timer counter register 0n (TCR0n) is captured to the TDR0n register when the capture trigger is input. A valid edge of the TI0n pin can be selected as the capture trigger. This selection is made by timer mode register 0n (TMR0n).

16.2.3 TPS0 : Timer Clock Select Register 0



16.2.2 TDR0n/TDR01x/TDR03x: 定时器数据寄存器 0n(n = 0至 7)(x = L, H)



少量	象征	功能	R/W
15:0	n/a	16 位定时器捕获结果或设置比较单元 m 和通道 n 的数据	R/W

这是一个 16 位寄存器，可以从中选择捕获功能和比较功能。

可以通过使用定时器模式寄存器 0n (TMR0n) 的 TMR0n.MD[2:0] 位和 TMR0n.OPIRQ 位来选择操作模式，从而切换捕获或比较功能。

TDR0n寄存器的值可以随时更改。该寄存器可以以16位为单位进行读写操作。

此外，对于 TDR01 和 TDR03 寄存器，在 8 位定时器模式下（定时器模式寄存器 01 和 03 (TMR01, TMR03) 的 SPLIT01、SPLIT03 位为 1 时），可以以 8 位为单位读取和写入数据，其中 TDR01H 和 TDR03H 用作高 8 位，TDR01L 和 TDR03L 用作低 8 位。

(i) 当定时器数据寄存器 0n (TDR0n) 用作比较寄存器时

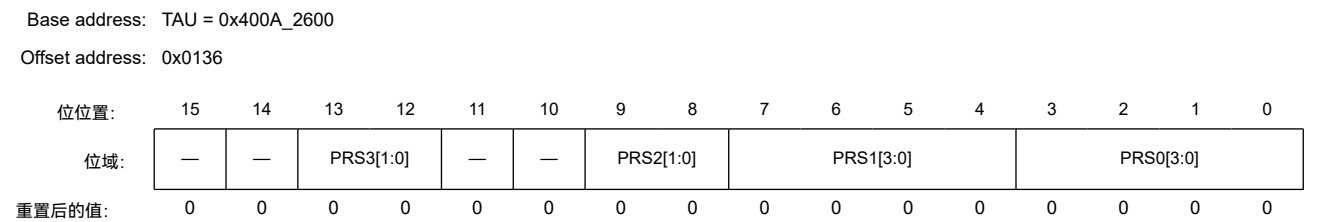
倒计时从TDR0n寄存器的值开始。当计数达到0x0000时，会产生一个中断信号（TAU0_TMI0n）。TDR0n寄存器会保持其值，直到被重新写入。

注意：当 TDR0n 寄存器设置为比较功能时，即使输入捕获触发信号，它也不会执行捕获操作。

(ii) 当定时器数据寄存器 0n (TDR0n) 用作捕获寄存器时

当输入捕获触发信号时，定时器计数器寄存器 0n (TCR0n) 的计数值被捕获到 TDR0n 寄存器中。可以选择 TI0n 引脚的有效边沿作为捕获触发信号。此选择由定时器模式寄存器0n(TMR0n) 完成。

16.2.3 TPS0: 定时器时钟选择寄存器 0



Bit	Symbol	Function	R/W
3:0	PRS0[3:0]	Selection of Operation Clock (CK00)*1 *3 *4 0x0: PCLKB 0x1: PCLKB/2 0x2: PCLKB/2 ² 0x3: PCLKB/2 ³ 0x4: PCLKB/2 ⁴ 0x5: PCLKB/2 ⁵ 0x6: PCLKB/2 ⁶ 0x7: PCLKB/2 ⁷ 0x8: PCLKB/2 ⁸ 0x9: PCLKB/2 ⁹ 0xA: PCLKB/2 ¹⁰ 0xB: PCLKB/2 ¹¹ 0xC: PCLKB/2 ¹² 0xD: PCLKB/2 ¹³ 0xE: PCLKB/2 ¹⁴ 0xF: PCLKB/2 ¹⁵	R/W
7:4	PRS1[3:0]	Selection of Operation Clock (CK01)*1 *3 *4 0x0: PCLKB 0x1: PCLKB/2 0x2: PCLKB/2 ² 0x3: PCLKB/2 ³ 0x4: PCLKB/2 ⁴ 0x5: PCLKB/2 ⁵ 0x6: PCLKB/2 ⁶ 0x7: PCLKB/2 ⁷ 0x8: PCLKB/2 ⁸ 0x9: PCLKB/2 ⁹ 0xA: PCLKB/2 ¹⁰ 0xB: PCLKB/2 ¹¹ 0xC: PCLKB/2 ¹² 0xD: PCLKB/2 ¹³ 0xE: PCLKB/2 ¹⁴ 0xF: PCLKB/2 ¹⁵	R/W
9:8	PRS2[1:0]	Selection of Operation Clock (CK02)*1 *2 0x0: PCLKB/2 0x1: PCLKB/2 ² 0x2: PCLKB/2 ⁴ 0x3: PCLKB/2 ⁶	R/W
11:10	—	These bits are read as 0. The write value should be 0.	R/W
13:12	PRS3[1:0]	Selection of Operation Clock (CK03)*1 *2 0x0: PCLKB/2 ⁸ 0x1: PCLKB/2 ¹⁰ 0x2: PCLKB/2 ¹² 0x3: PCLKB/2 ¹⁴	R/W
15:14	—	These bits are read as 0. The write value should be 0.	R/W

- Note 1. When changing the clock selected for PCLKB, stop timer array unit (TT0 = 0x00FF).
- Note 2. The timer array unit must also be stopped if the operating clock (f_{MCK}) or the valid edge of the signal input from the TI0n pin is selected.
- Note 3. If PCLKB (undivided) is selected as the operation clock (CK00, CK01) and TDR0n is set to 0x0000 (n = 0 to 7), interrupt requests output from timer array units cannot be used.
- Note 4. Waveform of the clock to be selected in the TPS0 register which becomes high level for one period of PCLKB from its rising edge. For details, see [section 16.4.1. Count Clock \(f_{TCLK}\)](#).

Note: PCLKB: CPU and peripheral hardware clock frequency.

The TPS0 register is a 16-bit register used to select two types or four types of operation clocks (CK00, CK01, CK02, CK03) that are commonly supplied to each channel. CK00 is selected by using bits 3 to 0 of the TPS0 register, and CK01

少量	象征	功能	R/W
3:0	PRS0[3:0]	选择操作时钟 (CK00)*1 *3 *4 0x0: PCLKB 0x1: PCLKB/2 0x2: PCLKB/22 0x3: PCLKB/23 0x4: PCLKB/24 0x5: PCLKB/25 0x6: PCLKB/26 0x7: PCLKB/27 0x8: PCLKB/28 0x9: PCLKB/29 0xA: PCLKB/210 0xB: PCLKB/211 0xC: PCLKB/212 0xD: PCLKB/213 0xE: PCLKB/214 0xF: PCLKB/215	R/W
7:4	PRS1[3:0]	选择操作时钟 (CK01)*1 *3 *4 0x0: PCLKB 0x1: PCLKB/2 0x2: PCLKB/22 0x3: PCLKB/23 0x4: PCLKB/24 0x5: PCLKB/25 0x6: PCLKB/26 0x7: PCLKB/27 0x8: PCLKB/28 0x9: PCLKB/29 0xA: PCLKB/210 0xB: PCLKB/211 0xC: PCLKB/212 0xD: PCLKB/213 0xE: PCLKB/214 0xF: PCLKB/215	R/W
9:8	PRS2[1:0]	选择操作时钟 (CK02)*1 *2 0x0: PCLKB/2 0x1: PCLKB/22 0x2: PCLKB/2 ⁴ 0x3: PCLKB/2 ⁶	R/W
11:10	—	这些位读取为 0。写入值应为 0。	R/W
13:12	PRS3[1:0]	选择操作时钟 (CK03)*1 *2 0x0: PCLKB/28 0x1: PCLKB/210 0x2: PCLKB/212 0x3: PCLKB/214	R/W
15:14	—	这些位读取为 0。写入值应为 0。	R/W

- 注 1. 当更改为PCLKB选择的时钟时，停止定时器阵列单元(TT0 = 0x00FF)。
- 注 2. 如果选择了工作时钟(f_{MCK}) 或来自 TI0n 引脚的信号输入的有效边沿，则定时器阵列单元也必须停止。
- 注 3. 如果选择 PCLKB（未分频）作为操作时钟（CK00、CK01），并且 TDR0n 设置为0x0000 (n = 0 至 7，则无法使用定时器阵列单元的中断请求输出。
- 注 4. 要在 TPS0 寄存器中选择的时钟波形，从 PCLKB 的上升沿开始，在 PCLKB 的一个周期内变为高电平。
[有关详细信息，请参阅第 16.4.1 节。计数时钟 \(f_{TCLK}\)。](#)

注：PCLKB：CPU及外围硬件时钟频率。

TPS0 寄存器是一个 16 位寄存器，用于选择两种或四种类型的操作时钟（CK00、CK01、CK02、CK03）通常提供给每个通道。CK00 通过 TPS0 寄存器的第 3 位到第 0 位选择，CK01 通过 TPS0 寄存器的第 7 位到第 4 位选择。

is selected by using bits 7 to 4 of the TPS0 register. In addition, only for channels 1 and 3, CK02 and CK03 can be also selected. CK02 is selected by using bits 9 and 8 of the TPS0 register, and CK03 is selected by using bits 13 and 12 of the TPS0 register.

Rewriting of the TPS0 register during timer operation is possible only in the following cases.

- If the PRS0[3:0] bits can be rewritten (n = 0 to 7):
All channels for which CK00 is selected as the operation clock (TMR0n.CKS[1:0] = 00b) are stopped (TE0.TE[n] = 0).
- If the PRS1[3:0] bits can be rewritten (n = 0 to 7):
All channels for which CK01 is selected as the operation clock (TMR0n.CKS[1:0] = 01b) are stopped (TE0.TE[n] = 0).
- If the PRS2[1:0] bits can be rewritten (n = 1, 3):
All channels for which CK02 is selected as the operation clock (TMR0n.CKS[1:0] = 10b) are stopped (TE0.TE[n] = 0).
- If the PRS3[1:0] bits can be rewritten (n = 1, 3):
All channels for which CK03 is selected as the operation clock (TMR0n.CKS[1:0] = 11b) are stopped (TE0.TE[n] = 0).

PRS0[3:0] bits (Selection of Operation Clock (CK00))

The input sources that can be selected with the PRS0[3:0] bits are shown in Table 16.4.

PRS1[3:0] bits (Selection of Operation Clock (CK01))

The input sources that can be selected with the PRS1[1:0] bits are shown in Table 16.4.

Table 16.4 Selection of operation clock (PRSk (k = 0, 1))

PRSk[3:0]	Selection of operation clock (CK0k)*1 (k = 0, 1)					
		PCLKB = 2 MHz	PCLKB = 5 MHz	PCLKB = 10 MHz	PCLKB = 20 MHz	PCLKB = 32 MHz
0000b	PCLKB	2 MHz	5 MHz	10 MHz	20 MHz	32 MHz
0001b	PCLKB/2	1 MHz	2.5 MHz	5 MHz	10 MHz	16 MHz
0010b	PCLKB/2 ²	500 kHz	1.25 MHz	2.5 MHz	5 MHz	8 MHz
0011b	PCLKB/2 ³	250 kHz	625 kHz	1.25 MHz	2.5 MHz	4 MHz
0100b	PCLKB/2 ⁴	125 kHz	313 kHz	625 kHz	1.25 MHz	2 MHz
0101b	PCLKB/2 ⁵	62.5 kHz	156 kHz	313 kHz	625 kHz	1 MHz
0110b	PCLKB/2 ⁶	31.3 kHz	78.1 kHz	156 kHz	313 kHz	500 kHz
0111b	PCLKB/2 ⁷	15.6 kHz	39.1 kHz	78.1 kHz	156 kHz	250 kHz
1000b	PCLKB/2 ⁸	7.81 kHz	19.5 kHz	39.1 kHz	78.1 kHz	125 kHz
1001b	PCLKB/2 ⁹	3.91 kHz	9.77 kHz	19.5 kHz	39.1 kHz	62.5 kHz
1010b	PCLKB/2 ¹⁰	1.95 kHz	4.88 kHz	9.77 kHz	19.5 kHz	31.3 kHz
1011b	PCLKB/2 ¹¹	977 Hz	2.44 kHz	4.88 kHz	9.77 kHz	15.6 kHz
1100b	PCLKB/2 ¹²	488 Hz	1.22 kHz	2.44 kHz	4.88 kHz	7.81 kHz
1101b	PCLKB/2 ¹³	244 Hz	610 Hz	1.22 kHz	2.44 kHz	3.91 kHz
1110b	PCLKB/2 ¹⁴	122 Hz	305 Hz	610 Hz	1.22 kHz	1.95 kHz
1111b	PCLKB/2 ¹⁵	61.0 Hz	153 Hz	305 Hz	610 Hz	977 Hz

Note 1. When changing the clock selected for PCLKB, stop timer array unit (TT0 = 0x00FF).

Note: If PCLKB (undivided) is selected as the operation clock (CK0k) and TDR0n is set to 0x0000 (n = 0 to 7), interrupt requests output from timer array units cannot be used.

Note: PCLKB: CPU and peripheral hardware clock frequency.

Note: Waveform of the clock to be selected in the TPS0 register which becomes high level for one period of PCLKB from its rising edge. For details, see section 16.4.1. Count Clock (f_{TCLK}).

此外，仅对于通道 1 和 3，还可以选择 CK02 和 CK03。CK02 通过 TPS0 寄存器的第 9 位和第 8 位选择，CK03 通过 TPS0 寄存器的第 13 位和第 12 位选择。

定时器运行期间，只有在以下情况下才能重写 TPS0 寄存器。

- 如果 PRS0[3:0] 位可以重写为(n = 0至 7):
凡选择 CK00 作为操作时钟的通道(TMR0n.CKS[1:0] = 00b) 均停止(TE0.TE[n] = 0)。
- 如果 PRS1[3:0] 位可以重写为(n = 0至 7):
凡选择 CK01 作为操作时钟的通道(TMR0n.CKS[1:0] = 01b) 均停止(TE0.TE[n] = 0)。
- If the PRS2[1:0] bits can be rewritten (n = 1, 3):
所有选择 CK02 作为操作时钟的通道(TMR0n.CKS[1:0] = 10b) 都停止(TE0.TE[n] = 0) 。
- 如果 PRS3[1:0] 位可以重写(n = 1, 3)
凡选择 CK03 作为操作时钟的通道(TMR0n.CKS[1:0] = 11b) 均停止(TE0.TE[n] = 0)。

PRS0[3:0] 位（操作时钟 (CK00) 的选择）

表 16.4 显示了可使用 PRS0[3:0] 位选择的输入源。

PRS1[3:0] 位（操作时钟 (CK01) 的选择）

表 16.4 显示了可使用 PRS1[1:0] 位选择的输入源。

表 16.4 操作时钟的选择 (PRSk(k = 0, 1))

PRSk[3:0]	选择操作时钟(CK0k)*1 (k = 0, 1)					
		PCLKB = 2 MHz	PCLKB = 5 MHz	PCLKB = 10 MHz	PCLKB = 20 MHz	PCLKB = 32 MHz
0000b	PCLKB	2 MHz	5兆赫	10 MHz	20 MHz	32 MHz
0001b	PCLKB/2	1 MHz	2.5 MHz	5兆赫	10 MHz	16 MHz
0010b	PCLKB/2 ²	500 kHz	1.25 MHz	2.5 MHz	5兆赫	8 MHz
0011b	PCLKB/2 ³	250 kHz	625 kHz	1.25 MHz	2.5 MHz	4 MHz
0100b	PCLKB/2 ⁴	125 kHz	313 kHz	625 kHz	1.25 MHz	2 MHz
0101b	PCLKB/2 ⁵	62.5 kHz	156 kHz	313 kHz	625 kHz	1 MHz
0110b	PCLKB/2 ⁶	31.3 kHz	78.1 kHz	156 kHz	313 kHz	500 kHz
0111b	PCLKB/2 ⁷	15.6 kHz	39.1 kHz	78.1 kHz	156 kHz	250 kHz
1000b	PCLKB/2 ⁸	7.81 kHz	19.5 kHz	39.1 kHz	78.1 kHz	125 kHz
1001b	PCLKB/2 ⁹	3.91 kHz	9.77 kHz	19.5 kHz	39.1 kHz	62.5 kHz
1010b	PCLKB/2 ¹⁰	1.95 kHz	4.88 kHz	9.77 kHz	19.5 kHz	31.3 kHz
1011b	PCLKB/2 ¹¹	977 Hz	2.44 kHz	4.88 kHz	9.77 kHz	15.6 kHz
1100b	PCLKB/2 ¹²	488 Hz	1.22 kHz	2.44 kHz	4.88 kHz	7.81 kHz
1101b	PCLKB/2 ¹³	244 Hz	610 Hz	1.22 kHz	2.44 kHz	3.91 kHz
1110b	PCLKB/2 ¹⁴	122 Hz	305 Hz	610 Hz	1.22 kHz	1.95 kHz
1111b	PCLKB/2 ¹⁵	61.0 Hz	153 Hz	305 Hz	610 Hz	977 Hz

注1. 当更改PCLKB所选时钟时，停止定时器阵列单元(TT0 = 0x00FF) 。

注意: 如果选择 PCLKB（未分频）作为操作时钟 (CK0k) 并且 TDR0n 设置为0x0000 (n = 0 至 7，则无法使用定时器阵列单元的中断请求输出。

注: PCLKB: CPU及外围硬件时钟频率。

注: 待选时钟波形存储在 TPS0 寄存器中，该寄存器从 PCLKB 的上升沿开始，在一个周期内保持高电平。详情请参见 16.4.1 节。计数时钟 (f_{TCLK})。

PRS2[1:0] bits (Selection of Operation Clock (CK02))

The input sources that can be selected with the PRS2[1:0] bits are shown in Table 16.5.

Table 16.5 Selection of operation clock (PRS2[1:0])

PRS2[1:0]	Selection of operation clock (CK02)*1					
		PCLKB = 2 MHz	PCLKB = 5 MHz	PCLKB = 10 MHz	PCLKB = 20 MHz	PCLKB = 32 MHz
00b	PCLKB/2	1 MHz	2.5 MHz	5 MHz	10 MHz	16 MHz
01b	PCLKB/2 ²	500 kHz	1.25 MHz	2.5 MHz	5 MHz	8 MHz
10b	PCLKB/2 ⁴	125 kHz	313 kHz	625 kHz	1.25 MHz	2 MHz
11b	PCLKB/2 ⁶	31.3 kHz	78.1 kHz	156 kHz	313 kHz	500 kHz

Note 1. When changing the clock selected for PCLKB, stop timer array unit (TT0 = 0x00FF).
The timer array unit must also be stopped if the operating clock (f_{MCK}) or the valid edge of the signal input from the TI0n pin is selected.

PRS3[1:0] bits (Selection of Operation Clock (CK03))

The input sources that can be selected with the PRS3[1:0] bits are shown in Table 16.6.

Table 16.6 Selection of operation clock (PRS3[1:0])

PRS3[1:0]	Selection of operation clock (CK03)*1					
		PCLKB = 2 MHz	PCLKB = 5 MHz	PCLKB = 10 MHz	PCLKB = 20 MHz	PCLKB = 32 MHz
00b	PCLKB/2 ⁸	7.81 kHz	19.5 kHz	39.1 kHz	78.1 kHz	125 kHz
01b	PCLKB/2 ¹⁰	1.95 kHz	4.88 kHz	9.77 kHz	19.5 kHz	31.3 kHz
10b	PCLKB/2 ¹²	488 Hz	1.22 kHz	2.44 kHz	4.88 kHz	7.81 kHz
11b	PCLKB/2 ¹⁴	122 Hz	305 Hz	610 Hz	1.22 kHz	1.95 kHz

Note 1. When changing the clock selected for PCLKB, stop timer array unit (TT0 = 0x00FF).
The timer array unit must also be stopped if the operating clock (f_{MCK}) or the valid edge of the signal input from the TI0n pin is selected.

By using channels 1 and 3 in the 8-bit timer mode and specifying CK02 or CK03 as the operation clock, the interval times shown in Table 16.7 can be achieved by using the interval timer function.

Table 16.7 Interval times available for operation clock CK02 or CK03

Clock		Interval time*1 (PCLKB = 32 MHz)			
		10 μs	100 μs	1 ms	10 ms
CK02	PCLKB/2	✓	—	—	—
	PCLKB/2 ²	✓	—	—	—
	PCLKB/2 ⁴	✓	✓	—	—
	PCLKB/2 ⁶	✓	✓	—	—
CK03	PCLKB/2 ⁸	—	✓	✓	—
	PCLKB/2 ¹⁰	—	✓	✓	—
	PCLKB/2 ¹²	—	—	✓	✓
	PCLKB/2 ¹⁴	—	—	✓	✓

Note 1. The margin is within 5%.

Note: PCLKB: CPU and peripheral hardware clock frequency

Note: For details of a signal of PCLKB/2^j selected with the TPS0 register, see section 16.4.1. Count Clock (f_{TCLK}).

PRS2[1:0] 位（操作时钟 (CK02) 选择）

表 16.5 显示了可使用 PRS2[1:0] 位选择的输入源。

表 16.5 操作时钟的选择 (PRS2[1:0])

PRS2[1:0]	选择操作时钟 (CK02) *1					
		PCLKB = 2 MHz	PCLKB = 5 MHz	PCLKB = 10 MHz	PCLKB = 20 MHz	PCLKB = 32 MHz
00b	PCLKB/2	1 MHz	2.5 MHz	5兆赫	10 MHz	16 MHz
01b	PCLKB/2 ²	500 kHz	1.25 MHz	2.5 MHz	5兆赫	8 MHz
10b	PCLKB/2 ⁴	125 kHz	313 kHz	625 kHz	1.25 MHz	2 MHz
11b	PCLKB/2 ⁶	31.3 kHz	78.1 kHz	156 kHz	313 kHz	500 kHz

注1. 当更改为PCLKB选择的时钟时，停止定时器阵列单元(TT0 = 0x00FF)。
如果选择了工作时钟(f_{MCK}) 或来自TI0n引脚的信号输入的有效边沿，则定时器阵列单元也必须停止。

PRS3[1:0] 位（操作时钟 (CK03) 选择）

表 16.6 显示了可使用 PRS3[1:0] 位选择的输入源。

表 16.6 操作时钟的选择 (PRS3[1:0])

PRS3[1:0]	选择操作时钟 (CK03)*1					
		PCLKB = 2 MHz	PCLKB = 5 MHz	PCLKB = 10 MHz	PCLKB = 20 MHz	PCLKB = 32 MHz
00b	PCLKB/2 ⁸	7.81 kHz	19.5 kHz	39.1 kHz	78.1 kHz	125 kHz
01b	PCLKB/2 ¹⁰	1.95 kHz	4.88 kHz	9.77 kHz	19.5 kHz	31.3 kHz
10b	PCLKB/2 ¹²	488 Hz	1.22 kHz	2.44 kHz	4.88 kHz	7.81 kHz
11b	PCLKB/2 ¹⁴	122 Hz	305 Hz	610 Hz	1.22 kHz	1.95 kHz

注1. 当更改PCLKB所选时钟时，停止定时器阵列单元(TT0 = 0x00FF)。
如果选择了工作时钟(f_{MCK}) 或来自TI0n引脚的信号输入的有效边沿，则定时器阵列单元也必须停止。

通过在 8 位定时器模式下使用通道 1 和 3，并指定 CK02 或 CK03 作为操作时钟，可以使用间隔定时器功能实现表 16.7 中所示的间隔时间。

表 16.7 可用于操作时钟 CK02 或 CK03 的间隔时间

钟		间隔时间*1(PCLKB = 32 MHz)			
		10 μs	100 μs	1 ms	10 ms
CK02	PCLKB/2	✓	—	—	—
	PCLKB/2 ²	✓	—	—	—
	PCLKB/2 ⁴	✓	✓	—	—
	PCLKB/2 ⁶	✓	✓	—	—
CK03	PCLKB/2 ⁸	—	✓	✓	—
	PCLKB/2 ¹⁰	—	✓	✓	—
	PCLKB/2 ¹²	—	—	✓	✓
	PCLKB/2 ¹⁴	—	—	✓	✓

注1.边距在5%内。

注: PCLKB: CPU 和外围硬件时钟频率

注意: 有关使用 TPS0 寄存器选择的 PCLKB/2^j信号的详细信息，请参阅第 16.4.1 节。计数时钟 (f_{TCLK})。

16.2.4 TMR0n : Timer Mode Register 0n (n = 0, 2, 4, 5, 6, 7)

Base address: TAU = 0x400A_2600

Offset address: 0x0110 (TMR00)
0x0114 (TMR02)
0x0118 (TMR04)
0x011A (TMR05)
0x011C (TMR06)
0x011E (TMR07)

Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	CKS[1:0]	—	CCS	MAST ER	STS[2:0]	CIS[1:0]	—	—	MD[2:0]	OPIR Q						
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
0	OPIRQ	Setting of Starting Count and Interrupt	R/W
3:1	MD[2:0]	Selection of Operation Mode at Channel n 0 0 0: Interval timer mode 0 1 0: Capture mode 0 1 1: Event counter mode 1 0 0: One-count mode*1 1 1 0: Capture and one-count mode Others: Setting prohibited	R/W
5:4	—	These bits are read as 0. The write value should be 0.	R/W
7:6	CIS[1:0]	Selection of TI0n Pin Input Valid Edge 0 0: Falling edge 0 1: Rising edge 1 0: Both edges (when low-level width is measured) Start trigger: Falling edge, Capture trigger: Rising edge 1 1: Both edges (when high-level width is measured) Start trigger: Rising edge, Capture trigger: Falling edge	R/W
10:8	STS[2:0]	Setting of Start Trigger or Capture Trigger of Channel n 0 0 0: Only software trigger start is valid (other trigger sources are unselected). 0 0 1: Valid edge of the TI0n pin input is used as both the start trigger and capture trigger. 0 1 0: Both the edges of the TI0n pin input are used as a start trigger and a capture trigger. 1 0 0: Interrupt signal of the master channel is used (when the channel is used as a slave channel with the simultaneous channel operation function). Others: Setting prohibited	R/W
11	MASTER*2	Selection Between Using Channel n Independently or Simultaneously with Another Channel (as a Slave or Master) 0: Operates in independent channel operation function or as slave channel in simultaneous channel operation function. 1: Operates as master channel in simultaneous channel operation function.	R/W
12	CCS	Selection of Counter Clock (f _{TCLK}) of Channel n 0: Operating clock (f _{MCK})specified by the CKS[1:0] bits 1: Valid edge of input signal input from the TI0n pin. - In channel 5, valid edge of input signal selected by the TIS0 register - In channel 7, valid edge of input signal selected by the ISC register	R/W
13	—	This bit is read as 0. The write value should be 0.	R/W
15:14	CKS[1:0]	Selection of Operation Clock (f _{MCK}) of Channel n 0 0: Operation clock CK00 set by timer clock select register 0 (TPS0) 0 1: Operation clock CK02 set by timer clock select register 0 (TPS0) 1 0: Operation clock CK01 set by timer clock select register 0 (TPS0) 1 1: Operation clock CK03 set by timer clock select register 0 (TPS0)	R/W

Note 1. In one-count mode, interrupt output (TAU0_TMI0n) when starting a count operation and TO0n output are not controlled.
Note 2. Not supported (Bit 11 is a read-only bit and fixed to 0) when n = 0, 5, 7. Writing to this bit is ignored.

Note: The bit function assigned to bit 11 of the TMR0n register depends on the channel.

16.2.4 TMR0n: 定时器模式寄存器 0n(n = 0, 2, 4, 5, 6, 7)

基地址: TAU = 0x400A_2600

偏移地址: 0x0110 (TMR00)
0x0114 (TMR02)
0x0118 (TMR04)
0x011A (TMR05)
0x011C (TMR06)
0x011E (TMR07)

位位置:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域:	CKS[1:0]	—	CCS	MAST ER	STS[2:0]	CIS[1:0]	—	—	MD[2:0]	OPIR Q						
重置后的值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
0	OPIRQ	设置起始计数和中断	R/W
3:1	MD[2:0]	通道 n 的操作模式选择 0 0 0: 间隔定时器模式 0 1 0: 捕获模式 0 1 1: 事件计数器模式 1 0 0: 单次计数模式*1 1 1 0: 捕获和单次计数模式 其他: 禁止设置	R/W
5:4	—	这些位读取为 0。写入值应为 0。	R/W
7:6	CIS[1:0]	选择TI0n引脚输入有效边沿 0 0: 下降沿 0 1: 上升沿 1 0:两条边（当测量低级宽度时） 启动触发: 下降沿, 捕获触发: 上升沿 1 1: 两侧边缘（测量高级宽度时） 启动触发: 上升沿, 捕获触发: 下降沿	R/W
10:8	STS[2:0]	通道 n 0 0 0 的启动触发或捕获触发设置: 仅软件触发启动有效（其他触发源未选择）。 0 0 1: TI0n 引脚输入的有效边沿同时用作启动触发和捕获触发。 0 1 0: TI0n 引脚输入的两个边沿均用作启动触发和捕获触发。1 0 0: 使用主通道的 中断信号（当通道用作具有同步通道操作功能的从通道时）。 其他: 禁止设置	R/W
11	MASTER*2	选择是独立使用通道 n, 还是同时与另一个通道（作为从属通道或主通道）使用通道 n 0: 以独立通道操作功能运行, 或作为同步通道操作功能中的从属通道运行。 1: 在同步通道操作功能中作为主通道运行。	R/W
12	CCS	通道 n 的计数器时钟 (f _{TCLK}) 的选择 0: 由 CKS[1:0] 位 1 指定 的运行时钟(f _{MCK})01: 来自 TI0n 引脚的输入信号的有效边沿。 ●在通道 5 中, 输入信号的有效边沿由 TIS0 寄存器选择 ●在通道 7 中, 输入信号的有效边沿由 ISC 寄存器选择	R/W
13	—	该位读取值为 0。写入值应为 0。	R/W
15:14	CKS[1:0]	选择通道 n 的操作时钟(f _{MCK}) 0 0: 操作时钟 CK00 由定时器时钟选择寄存器 0 (TPS0) 设置 0 1: 操作时钟 CK02 由定时器时钟选择寄存器 0 (TPS0) 设置 1 0: 操作时钟 CK01 由定时器时钟选择寄存器 0 (TPS0) 设置 1 1: 操作时钟CK03由定时器时钟选择寄存器 0 (TPS0) 设置	R/W

Note 1. In one-count mode, interrupt output (TAU0_TMI0n) when starting a count operation and TO0n output are not controlled.
Note 2. Not supported (Bit 11 is a read-only bit and fixed to 0) when n = 0, 5, 7. Writing to this bit is ignored.

注意: TMR0n 寄存器第 11 位所分配的位功能取决于通道。

TMR00, TMR05, TMR07: Fixed to 0

The TMR0n register sets an operation mode of channel n. This register is used to select the operation clock (f_{MCK}), select the count clock, select the master or slave, select the 16-bit timer, specify the start trigger and capture trigger, select the valid edge of the timer input, and specify the operation mode (interval, capture, event counter, one-count, or capture and one-count).

Rewriting the TMR0n register is prohibited while the corresponding timer is in operation (when $TE0.TE[n] = 1$). However, bits 7 and 6 (CIS[1:0]) can be rewritten even while the corresponding timer is operating with some functions (when $TE0.TE[n] = 1$). For details, see [section 16.7. Independent Channel Operation Function of Timer Array Unit](#) and [section 16.8. Simultaneous Channel Operation Function of Timer Array Unit](#).

Note: The timer array unit must be stopped ($TT0 = 0x00FF$) if the clock selected for PCLKB is changed even if the operating clock specified by using the CKS[1:0] bits (f_{MCK}) or the valid edge of the signal input from the TI0n pin is selected as the count clock(f_{TCLK}).

OPIRQ bit (Setting of Starting Count and Interrupt)

Table 16.8 lists operation mode that can selected with MD[2:0] bits and OPIRQ bit.

Table 16.8 OPIRQ operation mode selected with OPIRQ bit

Operation mode (MD[2:0])	OPIRQ	Setting of starting count and interrupt
Interval timer mode (000b) Capture mode (010b)	0	Timer interrupt is not generated when counting is started (timer output does not change, either).
	1	Timer interrupt is generated when counting is started (timer output also changes).
Event counter mode (011b)	0	Timer interrupt is not generated when counting is started (timer output does not change, either).
One-count mode (100b)	0	Start trigger is invalid during counting operation. At that time, interrupt is not generated.
	1	Start trigger is valid during counting operation *1. At that time, interrupt is not generated.
Capture & one-count mode (110b)	0	Timer interrupt is not generated when counting is started (timer output does not change, either). Start trigger is invalid during counting operation. At that time interrupt is not generated.
Other than above		Setting prohibited

Note 1. If the start trigger ($TS0.TS[n] = 1$) is issued during operation, the counter is initialized, and recounting is started (interrupt request does not occur).

MD[2:0] bits (Selection of Operation Mode at Channel n)

The operation in each mode varies depending on the TMR0n.OPIRQ bit (see Table 16.8).

Table 16.9 lists operation mode that can be selected with MD[2:0] bits.

Table 16.9 Operation mode selected with MD[2:0] bits (1 of 2)

MD[2:0]	Operation mode of channel n	Corresponding function	Count operation of TCR
000b	Interval timer mode	Interval timer or Square wave output or Divider function or PWM output (master)	Counting down
010b	Capture mode	Input pulse interval measurement	Counting up
011b	Event counter mode	External event counter	Counting down

TMR00、TMR05、TMR07：固定为 0

TMR0n 寄存器设置通道 n 的操作模式。该寄存器用于选择操作时钟(f_{MCK})、选择计数时钟、选择主控或从控、选择 16 位定时器、指定起始触发和捕获触发、选择定时器输入的有效边沿，以及指定操作模式（间隔、捕获、事件计数器、单次计数或捕获和单次计数）。

当相应的定时器正在运行时（ $TE0.TE[n] = 1$ ），禁止重写 TMR0n 寄存器。但是，即使相应的定时器正在执行某些功能（ $TE0.TE[n] = 1$ ），也可以重写第 7 位和第 6 位（CIS[1:0]）。有关详细信息，请参阅第 16.7 节“定时器阵列单元的独立通道操作功能”和第 16.8 节“定时器阵列单元的同步通道操作功能”。

注意：即使使用 CKS [1:0] 位(f_{MCK}) 指定的运行时钟或从 TI0n 引脚输入的信号的有效边沿被选作计数时钟 (f_{TCLK})，如果为 PCLKB 选择的时钟发生改变，定时器阵列单元也必须停止($TT0 = 0x00FF$)。

OPIRQ 位（起始计数和中断设置）

表 16.8 列出了可使用MD[2:0]位和 OPIRQ 位选择的操作模式。

表 16.8 使用 OPIRQ 位选择 OPIRQ 操作模式

操作模式（MD[2:0]）	OPIRQ	设置起始计数和中断
间隔定时器模式（000b） 捕获模式（010b）	0	开始计数时不会产生定时器中断（定时器输出也不会改变）。
	1	开始计数时会产生定时器中断（定时器输出也会发生变化）。
事件计数器模式（011b）	0	开始计数时不会产生定时器中断（定时器输出也不会改变）。
单次计数模式（100b）	0	计数操作期间启动触发器无效。 此时不会产生中断。
	1	计数操作期间启动触发器有效 *1。 此时不会产生中断。
捕获和单次计数模式（110b）	0	开始计数时不会产生定时器中断（定时器输出也不会改变）。 计数操作期间启动触发器无效。 此时不会产生中断。
除上述内容外		设置禁止

注 1. 如果在运行期间发出启动触发器($TS0.TS[n] = 1$)，则初始化计数器并开始重新计数（不会发生中断请求）。

MD[2:0] 位（通道 n 的操作模式选择）

每种模式下的操作都根据 TMR0n.OPIRQ 位而有所不同（参见表 16.8）。

表 16.9 列出了可使用 MD[2:0] 位选择的操作模式。

表 16.9 使用 MD[2:0] 位选择的操作模式（1/2）

MD[2:0]	通道 n 的工作模式	对应函数	TCR 的计数操作
000b	间隔定时器模式	间隔定时器或方波输出或除法函数或 PWM 输出（主输出）	倒计时
010b	捕获模式	输入脉冲间隔测量	计数
011b	事件计数器模式	外部事件计数器	倒计时

Table 16.9 Operation mode selected with MD[2:0] bits (2 of 2)

MD[2:0]	Operation mode of channel n	Corresponding function	Count operation of TCR
100b	One-count mode	Delay counter or One-shot pulse output or PWM output (slave)	Counting down
110b	Capture & one-count mode	Measurement of high- or low- level width of input signal	Counting up
Other than above	Setting prohibited		

CIS[1:0] bits (Selection of TI0n Pin Input Valid Edge)

If both the edges are specified when the value of the STS[2:0] bits is other than 010b, set the CIS[1:0] bits to 10b.

STS[2:0] bits (Setting of Start Trigger or Capture Trigger of Channel n)

These bits are used for setting the start trigger or capture trigger of channel n.

MASTER bit (Selection Between Using Channel n Independently or Simultaneously with Another Channel (as a Slave or Master))

Only the channel 2, 4, 6 can be set as a master channel (MASTER = 1).

Be sure to use channel 0, 5, 7 are fixed to 0 (Regardless of the bit setting, channel 0 operates as master, because it is the highest channel).

Clear the MASTER bit to 0 for a channel that is used with the independent channel operation function.

CCS bits (Selection of Counter Clock (f_{TCCLK}) of Channel n)

Counter clock (f_{TCCLK}) is used for the counter, output controller, and interrupt controller.

CKS[1:0] bits (Selection of Operation Clock (f_{MCK}) of Channel n)

Operation clock (f_{MCK}) is used by the edge detector. A count clock (f_{TCLK}) and a sampling clock are generated depending on the setting of the CCS bit.

The operation clocks CK02 and CK03 can only be selected for channels 1 and 3.

16.2.5 TMR0n : Timer Mode Register 0n (n = 1, 3)

Base address: TAU = 0x400A_2600																
Offset address: 0x0112 (TMR01) 0x0116 (TMR03)																
Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	CKS[1:0]		—	CCS	SPLIT	STS[2:0]		CIS[1:0]		—	—	MD[2:0]			OPIR Q	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
0	OPIRQ	Setting of Starting Count and Interrupt	R/W
3:1	MD[2:0]	Selection of Operation Mode at Channel n 0 0 0: Interval timer mode 0 1 0: Capture mode 0 1 1: Event counter mode 1 0 0: One-count mode 1 1 0: Capture & one-count mode Others: Setting prohibited	R/W
5:4	—	These bits are read as 0. The write value should be 0.	R/W

表 16.9 使用 MD[2:0] 位选择的操作模式 (2/2)

MD[2:0]	通道 n 的工作模式	对应函数	TCR的计数操作
100b	单次计数模式	延迟计数器或 单次脉冲输出或 PWM 输出（从机）	倒计时
110b	捕获和单次计数模式	测量输入信号的高电平或低电平宽度	计数
除上述内容外	设置禁止		

CIS[1:0] 位（TI0n 引脚输入有效边沿的选择）

如果当 STS[2:0] 位的值不为 010b 时，两个边都指定了，则将 CIS[1:0] 位设置为10b。

STS[2:0] 位（通道 n 的起始触发或捕获触发设置）

这些位用于设置通道 n 的起始触发或捕获触发。

主控位（选择独立使用通道 n 还是与另一个通道同时使用）
频道（作为奴隶或主人）

只有通道 2、4、6 可以设置为主通道(MASTER = 1)。

务必使用通道 0、5、7，它们固定为 0（无论位设置如何，通道 0 都作为主通道运行，因为它是最高的通道）。

将使用独立通道操作功能的通道的 MASTER 位清除为 0。

CCS 位（通道 n 的计数器时钟(f_{TCCLK})的选择）

计数器时钟(f_{TCCLK}) 用于计数器、输出控制器和中断控制器。

CKS[1:0] 位（通道 n 的操作时钟 (f_{MCK}) 选择）

边沿检测器使用操作时钟(f_{MCK})。计数时钟（f_{TCLK}）和采样时钟的生成取决于CCS位的设置。

操作时钟 CK02 和 CK03 只能为通道 1 和 3 选择。

16.2.5TMR0n： 定时器模式寄存器 0n(n = 1, 3)

基地址: TAU = 0x400A_2600																
偏移地址: 0x0112 (TMR01) 0x0116 (TMR03)																
位位置:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域:	CKS[1:0]		—	CCS	SPLIT	STS[2:0]		CIS[1:0]		—	—	MD[2:0]			OPIR Q	
重置后的值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
0	OPIRQ	设置起始计数和中断	R/W
3:1	MD[2:0]	通道 n 的操作模式选择 0 0 0: 间隔定时器模式 0 1 0: 捕获模式 0 1 1: 事件计数器模式 1 0 0: 单次计数模式 1 1 0: 捕获和单次计数模式 其他: 禁止设置	R/W
5:4	—	这些位读取为 0。写入值应为 0。	R/W

Bit	Symbol	Function	R/W
7:6	CIS[1:0]	Selection of TI0n Pin Input Valid Edge 0 0: Falling edge 0 1: Rising edge 1 0: Both edges (when low-level width is measured) Start trigger: Falling edge, Capture trigger: Rising edge 1 1: Both edges (when high-level width is measured) Start trigger: Rising edge, Capture trigger: Falling edge	R/W
10:8	STS[2:0]	Setting of Start Trigger or Capture Trigger of Channel n 0 0 0: Only software trigger start is valid (other trigger sources are unselected). 0 0 1: Valid edge of the Ti0n pin input is used as both the start trigger and capture trigger. 0 1 0: Both the edges of the Ti0n pin input are used as a start trigger and a capture trigger. 1 0 0: Interrupt signal of the master channel is used (when the channel is used as a slave channel with the simultaneous channel operation function). Others: Setting prohibited	R/W
11	SPLIT	Selection of 8 or 16-bit Timer Operation for Channels 1 and 3 0: Operates as 16-bit timer (Operates in independent channel operation function or as slave channel in simultaneous channel operation function.) 1: Operates as 8-bit timer	R/W
12	CCS	Selection of Counter Clock (f _{TCLK}) of Channel n 0: Operating clock (f _{MCK}) specified by the CKS[1:0] bits 1: Valid edge of input signal input from the Ti0n pin	R/W
13	—	This bit is read as 0. The write value should be 0.	R/W
15:14	CKS[1:0]	Selection of Operation Clock (f _{MCK}) of Channel n 0 0: Operation clock CK00 set by timer clock select register 0 (TPS0) 0 1: Operation clock CK02 set by timer clock select register 0 (TPS0) 1 0: Operation clock CK01 set by timer clock select register 0 (TPS0) 1 1: Operation clock CK03 set by timer clock select register 0 (TPS0)	R/W

Note: The bit function assigned to bit 11 of the TMR0n register depends on the channel.
TMR01, TMR03: SPLIT bit (n = 1, 3)

The TMR0n register sets an operation mode of channel n. This register is used to select the operation clock (f_{MCK}), select the count clock, select the master or slave, select the 16 or 8-bit timer (only for channels 1 and 3), specify the start trigger and capture trigger, select the valid edge of the timer input, and specify the operation mode (interval, capture, event counter, one-count, or capture & one-count).

Rewriting the TMR0n register is prohibited when the register is in operation (when TE0.TE[n] = 1). However, bits 7 and 6 (CIS[1:0]) can be rewritten even while the register is operating with some functions (when TE0.TE[n] = 1). For details, see [section 16.7. Independent Channel Operation Function of Timer Array Unit](#) and [section 16.8. Simultaneous Channel Operation Function of Timer Array Unit](#).

Note: The timer array unit must be stopped (TT0 = 0x00FF) if the clock selected for PCLKB is changed, even if the operating clock specified by using the CKS[1:0] bits (f_{MCK}) or the valid edge of the signal input from the TI0n pin is selected as the count clock(f_{TCLK}).

OPIRQ bit (Setting of Starting Count and Interrupt)

Table 16.8 lists operation mode that can be selected with MD[2:0] bits and OPIRQ bit.

MD[2:0] bits (Selection of Operation Mode at Channel n)

The operation in each mode varies depending on the TMR0n.OPIRQ bit (see Table 16.8).

Table 16.9 lists operation mode that can be selected with MD[2:0] bits.

CIS[1:0] bits (Selection of TI0n Pin Input Valid Edge)

If both the edges are specified when the value of the STS[2:0] bits is other than 010b, set the CIS[1:0] bits to 10b.

STS[2:0] bits (Setting of Start Trigger or Capture Trigger of Channel n)

These bits are used for setting the start trigger or capture trigger of channel n.

少量	象征	功能	R/W
7:6	CIS[1:0]	选择TI0n引脚输入有效边沿 0 0: 下降沿 0 1: 上升沿 1 0: 两侧边缘（测量低级宽度时） 启动触发: 下降沿, 捕获触发: 上升沿 1 1: 两侧边缘（测量高级宽度时） 启动触发: 上升沿, 捕获触发: 下降沿	R/W
10:8	STS[2:0]	设置通道 n 0 0 0 的启动触发或捕获触发: 仅软件触发启动有效（其他触发源未选择）。 0 0 1: Ti0n引脚输入的有效边沿同时用作起始触发和捕获触发。0 1 0: Ti0n引脚输入的两个边沿均用作起始触发和捕获触发。 1 0 0: 使用主通道的中断信号（当通道用作具有同步通道操作功能的从通道时）。 其他: 禁止设置	R/W
11	SPLIT	通道 1 和 3 的 8 位或 16 位定时器操作选择: 0: 作为 16 位定时器运行 （可作为独立通道运行, 也可作为同步通道运行中的从属通道。）1: 作为 8 位定时器运行	R/W
12	CCS	选择通道 n 的计数器时钟 (f _{TCLK}) 0: 由 CKS[1:0] 位指定的运行时钟 (f _{MCK}) 1: 来自 Ti0n 引脚的输入信号的有效边沿	R/W
13	—	该位读取值为 0。写入值应为 0。	R/W
15:14	CKS[1:0]	选择通道 n 的操作时钟(f _{MCK}) 0 0: 操作时钟 CK00 由定时器时钟选择寄存器 0 (TPS0) 设置 0 1: 操作时钟 CK02 由定时器时钟选择寄存器 0 (TPS0) 设置 1 0:操作时钟 CK01 由定时器时钟选择寄存器 0 (TPS0) 设置 1 1: 操作时钟 CK03 由定时器时钟选择寄存器 0 (TPS0) 设置	R/W

注意: TMR0n 寄存器第 11 位所分配的位功能取决于通道。
TMR01、TMR03: 拆分位(n = 1, 3)

TMR0n 寄存器设置通道 n 的操作模式。该寄存器用于选择操作时钟(f_{MCK})、选择计数时钟、选择主控或从控、选择 16 位或 8 位定时器（仅适用于通道 1 和 3）、指定起始触发和捕获触发、选择定时器输入的有效边沿, 以及指定操作模式（间隔、捕获、事件计数器、单次计数或捕获&单次计数）。

当 TMR0n 寄存器处于工作状态（TE0.TE[n] = 1时）时, 禁止重写该寄存器。但是, 即使寄存器正在执行某些功能（TE0.TE[n] = 1时）, 也可以重写第 7 位和第 6 位（CIS[1:0]）。有关详细信息, 请参阅第 16.7 节“定时器阵列单元的独立通道操作功能”和第 16.8 节“定时器阵列单元的同步通道操作功能”。

注意: 如果为 PCLKB 选择的时钟发生改变, 即使使用 CKS {1:0} 位(f_{MCK})指定的运行时钟或从 TI0n 引脚输入的信号的有效边沿被选作计数时钟 (f_{TCLK}), 定时器阵列单元也必须停止(TT0 = 0x00FF)。

OPIRQ 位（起始计数和中断设置）

表 16.8 列出了可使用 MD[2:0] 位和 OPIRQ 位选择的操作模式。

MD[2:0] 位（通道 n 的操作模式选择）

每种模式下的操作取决于 TMR0n.OPIRQ 位（见表 16.8）。

表 16.9 列出了可使用 MD[2:0] 位选择的操作模式。

CIS[1:0] 位（TI0n 引脚输入有效边沿的选择）

如果当 STS[2:0] 位的值不为 010b 时, 两个边都指定了, 则将 CIS[1:0] 位设置为10b.

STS[2:0] 位（通道 n 的起始触发或捕获触发设置）

这些位用于设置通道 n 的起始触发或捕获触发。

SPLIT bit (Selection of 8 or 16-bit Timer Operation for Channels 1 and 3)

This bit is used to select 8 or 16-bit timer operation for channels 1 and 3.

CCS bit (Selection of Counter Clock (f_{TCLK}) of Channel n)

Counter clock (f_{TCLK}) is used for the counter, output controller, and interrupt controller.

CKS[1:0] bits (Selection of Operation Clock (f_{MCK}) of Channel n)

Operation clock (f_{MCK}) is used by the edge detector. A count clock (f_{TCLK}) and a sampling clock are generated depending on the setting of the CCS bit.

16.2.6 TSR0n : Timer Status Register 0n (n = 0 to 7)

Base address:	TAU = 0x400A_2600															
Offset address:	0x0120 + 0x2 × n															
Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	OVF
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
0	OVF	Counter Overflow State of Channel n 0: Overflow does not occur 1: Overflow occurs	R
15:1	—	These bits are read as 0. The write value should be 0.	R

The TSR0n register indicates the overflow state of the counter of channel n.

The TSR0n register is valid only in the capture mode (TMR0n.MD[2:0] = 010b) and capture & one-count mode (TMR0n.MD[2:0] = 110b). See [Table 16.10](#) for the operation of the OVF bit in each operation mode and set or clear conditions.

The 8 lower-order bits of a TSR0n register can be handled as TSR0nL, which can be read by an 8-bit memory manipulation instruction.

OVF bit (Counter Overflow State of Channel n)

When OVF = 1, this flag is cleared (OVF = 0) when the next value is captured without overflow.

[Table 16.10](#) shows the OVF bit operation, set, and clear conditions in each operation mode.

Table 16.10 OVF bit operation, set and clear conditions in each operation mode

Timer operation mode	OVF bit	Set and clear conditions
● Capture mode ● Capture & one-count mode	clear	When no overflow has occurred upon capturing
	set	When an overflow has occurred upon capturing
● Interval timer mode ● Event counter mode ● One-count mode	clear	— (Use prohibited)
	set	

Note: The OVF bit does not change immediately after the counter has overflowed, but changes upon the subsequent capture.

分割位（选择通道 1 和 3 的 8 位或 16 位定时器操作）

该位用于选择通道 1 和 3 的 8 位或 16 位定时器操作。

CCS 位（通道 n 的计数器时钟 (f_{TCLK}) 选择）

计数器时钟 (f_{TCLK}) 用于计数器、输出控制器和中断控制器。

CKS[1:0] 位（通道 n 的操作时钟 (f_{MCK}) 选择）

边沿检测器使用操作时钟 (f_{MCK})。计数时钟 (f_{TCLK}) 和采样时钟的生成取决于 CCS 位的设置。

16.2.6 TSR0n: 定时器状态寄存器 0n(n = 0至 7)

基本地址:	TAU = 0x400A_2600															
偏移地址:	0x0120 + 0x2 × n															
位位置:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	OVF
重置后的值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
0	OVF	通道 n 的计数器溢出状态 0: 不会发生溢出。 1: 发生溢出	R
15:1	—	这些位读取为 0。写入值应为 0。	R

TSR0n 寄存器指示通道 n 的计数器的溢出状态。

TSR0n 寄存器仅在捕获模式(TMR0n.MD[2:0] = 010b) 和捕获& 单计数模式(TMR0n.MD[2:0] = 110b)下有效。

有关每种操作模式下 OVF 位的操作以及置位或清除条件，请参见表 16.10。

TSR0n 寄存器的低 8 位可以作为 TSR0nL 处理，可以通过 8 位内存操作指令读取。

OVF 位（通道 n 的计数器溢出状态）

当 OVF = 1 时，当捕获到下一个值而没有溢出时，此标志将被清除(OVF = 0)。

表 16.10 显示了每种操作模式下 OVF 位的操作、置位和清除条件。

表 16.10OVF 位操作，每种操作模式下的置位和清除条件

定时器操作模式	OVF 位	设置并清除条件
● 捕获模式 ● 捕捉与单次计数模式	清除	当捕获过程中未发生溢出时
	set	当捕获过程中发生溢出时
● 间隔定时器模式 ● 事件计数器模式 ● 单次计数模式	清除	— (禁止使用)
	set	

注意：OVF 位在计数器溢出后不会立即改变，而是在随后的捕获过程中改变。

16.2.7 TE0 : Timer Channel Enable Status Register 0

Base address: TAU = 0x400A_2600															
Offset address: 0x0130															
Bit position: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0															
Bit field: — — — — TEH3 — TEH1 — TE[7:0]															
Value after reset: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0															

Bit	Symbol	Function	R/W
7:0	TE[7:0]	Indication of Operation Enabled or Stopped State of Channel n 0: Operation is stopped 1: Operation is enabled	R
8	—	This bit is read as 0.	R
9	TEH1	Indication of whether Operation of the Higher 8-bit Timer is Enabled or Stopped when Channel 1 is in the 8-bit Timer Mode 0: Operation is stopped 1: Operation is enabled	R
10	—	This bit is read as 0.	R
11	TEH3	Indication of whether Operation of the Higher 8-bit Timer is Enabled or Stopped when Channel 3 is in the 8-bit Timer Mode 0: Operation is stopped 1: Operation is enabled	R
15:12	—	These bits are read as 0.	R

The TE0 register is used to enable or stop the timer operation of each channel.

Each bit of the TE0 register corresponds to each bit of the timer channel start register 0 (TS0) and the timer channel stop register 0 (TT0). When a bit of the TS0 register is set to 1, the corresponding bit of this register is set to 1. When a bit of the TT0 register is set to 1, the corresponding bit of this register is cleared to 0.

The lower 8 bits of the TE0 register can be set with an 8-bit memory manipulation instruction.

16.2.8 TS0 : Timer Channel Start Register 0

Base address: TAU = 0x400A_2600															
Offset address: 0x0132															
Bit position: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0															
Bit field: — — — — TSH3 — TSH1 — TS[7:0]															
Value after reset: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0															

Bit	Symbol	Function	R/W
7:0	TS[7:0]	Operation Enable (Start) Trigger of Channel n 0: No trigger operation 1: The TE0.TE[n] bit is set to 1 and the count operation becomes enabled	R/W
8	—	This bit is read as 0. The write value should be 0.	R/W
9	TSH1	Trigger to Enable Operation (Start Operation) of the Higher 8-bit Timer when Channel 1 is in the 8-bit Timer Mode 0: No trigger operation 1: The TE0.TEH1 bit is set to 1 and the count operation becomes enabled	R/W
10	—	This bit is read as 0. The write value should be 0.	R/W

16.2.7TE0： 定时器通道使能状态寄存器 0

基地址: TAU = 0x400A_2600															
偏移地址: 0x0130															
位位置: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0															
位域: — — — — TEH3 — TEH1 — TE[7:0]															
重置后的值: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0															

少量	象征	功能	R/W
7:0	TE[7:0]	指示通道 n 的运行启用或停止状态 0: 操作已停止 1: 操作已启用	R
8	—	该位读取值为 0。	R
9	TEH1	指示高位 8 位定时器的操作是启用还是停止 通道 1 处于 8 位定时器模式 0: 操作已停止 1: 操作已启用	R
10	—	该位读取值为 0。	R
11	TEH3	指示高位 8 位定时器的操作是启用还是停止 通道 3 处于 8 位定时器模式 0: 操作已停止 1: 操作已启用	R
15:12	—	这些位读取为 0。	R

TE0 寄存器用于启用或停止每个通道的定时器操作。

TE0 寄存器的每一位都对应于定时器通道起始寄存器 0 (TS0) 和定时器通道停止寄存器 0 (TT0) 的每一位。当 TS0 寄存器的某一位被设置为 1 时，该寄存器的对应位也被设置为 1。当 TT0 寄存器的某一位被设置为 1 时，该寄存器的对应位被清零。

TE0 寄存器的低 8 位可以通过 8 位内存操作指令进行设置。

16.2.8TS0： 定时器通道启动寄存器 0

基地址: TAU = 0x400A_2600															
偏移地址: 0x0132															
位位置: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0															
位域: — — — — TSH3 — TSH1 — TS[7:0]															
重置后的值: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0															

少量	象征	功能	R/W
7:0	TS[7:0]	通道 n 的操作使能（启动）触发 0: 无触发操作 1: TE0.TE[n] 位设置为 1，计数操作启用。	R/W
8	—	该位读取值为 0。写入值应为 0。	R/W
9	TSH1	当通道 1 处于 8 位定时器模式 0 时，触发启用高位 8 位定时器的操作（启动操作）：无触发操作 1: TE0.TEH1 位设置为 1，计数操作启用。	R/W
10	—	该位读取值为 0。写入值应为 0。	R/W

Bit	Symbol	Function	R/W
11	TSH3	Trigger to Enable Operation (Start Operation) of the Higher 8-bit Timer when Channel 3 is in the 8-bit Timer Mode 0: No trigger operation 1: The TE0.TEH3 bit is set to 1 and the count operation becomes enabled	R/W
15:12	—	These bits are read as 0. The write value should be 0.	R/W

Note: When switching from a function that does not use TI0n pin input to one that does, the following wait period is required from when timer mode register 0n (TMR0n) is set until the TS[n] (TSH1, TSH3) bit is set to 1.
When the TI0n pin noise filter is enabled (TNFEN.TNFEN0n = 1): Four cycles of the operation clock (f_{MCK})
When the TI0n pin noise filter is disabled (TNFEN.TNFEN0n = 0): Two cycles of the operation clock (f_{MCK})

Note: When the TS0 register is read, 0 is always read.

The TS0 register is a trigger register that is used to initialize timer counter register 0n (TCR0n) and start the counting operation of each channel.

When a bit of this register is set to 1, the corresponding bit of timer channel enable status register 0 (TE0) is set to 1. The TS[n], TSH1, TSH3 bits are immediately cleared when operation is enabled (TE0.TE[n], TEH1, TEH3), because they are trigger bits.

TS[7:0] bits (Operation Enable (Start) Trigger of Channel n)

The TCR0n register count operation start in the count operation enabled state varies depending on each operation mode (see Table 16.11 in section 16.4.2. Timing of the Start of Counting).

This bit is the trigger to enable operation (start operation) of the lower 8-bit timer for TS01 and TS03 when channel 1 or 3 is in the 8-bit timer mode.

TSH1 bit (Trigger to Enable Operation (Start Operation) of the Higher 8-bit Timer when Channel 1 is in the 8-bit Timer Mode)

The TCR01 register count operation start in the interval timer mode in the count operation enabled state (see Table 16.11 in section 16.4.2. Timing of the Start of Counting).

TSH3 bit (Trigger to Enable Operation (Start Operation) of the Higher 8-bit Timer when Channel 3 is in the 8-bit Timer Mode)

The TCR03 register count operation start in the interval timer mode in the count operation enabled state (see Table 16.11 in section 16.4.2. Timing of the Start of Counting).

16.2.9 TT0 : Timer Channel Stop Register 0

Base address:	TAU = 0x400A_2600															
Offset address:	0x0134															
Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	—	—	—	—	TTH3	—	TTH1	—	TT[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
7:0	TT[7:0]	Operation Stop Trigger of Channel n 0: No trigger operation 1: The TE0.TE[n] bit is cleared to 0 and the count operation is stopped	R/W
8	—	This bit is read as 0. The write value should be 0.	R/W
9	TTH1	Trigger to Stop Operation of the Higher 8-bit Timer when Channel 1 is in the 8-bit Timer Mode 0: No trigger operation 1: The TE0.TEH1 bit is cleared to 0 and the count operation is stopped	R/W
10	—	This bit is read as 0. The write value should be 0.	R/W

少量	象征	功能	R/W
11	TSH3	当通道 3 处于 8 位定时器模式 0 时，触发启用高位 8 位定时器的操作（启动操作）：无触发操作 1: TE0.TEH3 位设置为 1，计数操作启用。	R/W
15:12	—	这些位读取为 0。写入值应为 0。	R/W

注意：当从不使用 TI0n 引脚输入的功能切换到使用 TI0n 引脚输入的功能时，从定时器模式寄存器 0n (TMR0n) 设置到 TS[n] (TSH1, TSH3) 位设置为 1 之间需要等待一段时间。
当 TI0n 引脚噪声滤波器启用时 (TNFEN.TNFEN0n= 1): 四个工作时钟周期(f_{MCK}) 当 TI0n 引脚噪声滤波器禁用时 (TNFEN.TNFEN0n= 0) 两个工作时钟周期 (f_{MCK}) 注意：读取 TS0 寄存器时，始终读取 0。

TS0 寄存器是一个触发寄存器，用于初始化定时器计数器寄存器 0n (TCR0n) 并启动每个通道的计数操作。

当此寄存器的一位设置为 1 时，定时器通道使能状态寄存器 0 (TE0) 的相应位设置为 1。当操作使能 (TE0.TE[n], TSH1, TSH3 位) 时，TS[n], TEH1, TEH3 位立即被清除，因为它们是触发位。

TS[7:0] 位（通道 n 的操作使能（启动）触发）

在计数操作使能状态下，TCR0n 寄存器计数操作的开始时间因操作模式而异（参见 16.4.2 节中的表 16.11。计数开始时间）。

当通道 1 或 3 处于 8 位定时器模式时，此位触发 TS01 和 TS03 的低 8 位定时器的运行（启动运行）。

TSH1 位（当通道 1 处于 8 位定时器模式时，触发启用高 8 位定时器的操作（启动操作））

TCR01 寄存器计数操作在计数操作使能状态下以间隔定时器模式开始（参见 16.4.2 节中的表 16.11。计数开始的计时）。

TSH3 位（当通道 3 处于 8 位定时器模式时，触发启用高 8 位定时器的操作（启动操作））

TCR03 寄存器计数操作在计数操作使能状态下以间隔定时器模式启动（参见 16.4.2 节中的表 16.11。计数开始的定时）。

16.2.9TT0: 定时器通道停止寄存器 0

基地址:	TAU = 0x400A _2600															
偏移地址:	0x0134															
位位置:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域:	—	—	—	—	TTH3	—	TTH1	—	TT[7:0]							
重置后的值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
7:0	TT[7:0]	通道 n 的操作停止触发 0: 无触发操作 1: TE0.TE[n] 位被清零，计数操作停止。	R/W
8	—	该位读取值为 0。写入值应为 0。	R/W
9	TTH1	当通道 1 处于 8 位定时器状态时，触发停止高位 8 位定时器的运行。模式 0: 无触发操作 1: TE0.TEH1 位被清零，计数操作停止。	R/W
10	—	该位读取值为 0。写入值应为 0。	R/W

Bit	Symbol	Function	R/W
11	TTH3	Trigger to Stop Operation of the Higher 8-bit Timer when Channel 3 is in the 8-bit Timer Mode 0: No trigger operation 1: The TE0.TEH3 bit is cleared to 0 and the count operation is stopped	R/W
15:12	—	These bits are read as 0. The write value should be 0.	R/W

Note: When the TT0 register is read, 0 is always read.

The TT0 register is a trigger register that is used to stop the counting operation of each channel.

When a bit of this register is set to 1, the corresponding bit of timer channel enable status register 0 (TE0) is cleared to 0. The TT0.TT[n], TTH1, TTH3 bits are immediately cleared when operation is stopped (TE0.TE[n], TEH1, TEH3), because they are trigger bits.

16.2.10 TIS0 : Timer Input Select Register 0

Base address:	PORG = 0x400A_1000							
Offset address:	0x0004							
Bit position:	7	6	5	4	3	2	1	0
Bit field:	—	—	—	—	—	TIS[2]	—	—
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
1:0	—	These bits are read as 0. The write value should be 0.	R/W
2	TIS[2]	Selection of Timer Input Used with Channel 5 0: Input signal of timer input pin (TI05) 1: Low-speed on-chip oscillator (LOCO)	R/W
7:3	—	These bits are read as 0. The write value should be 0.	R/W

The TIS0 register is used to select the channel 5 of unit 0 timer input.

Note: Make sure that both the high-level and low-level widths of timer input to be selected are no less than 1/f_{MCK} + 10 ns. Therefore, when selecting FSUB as PCLKB, the TIS[2] bit cannot be set to 1.

16.2.11 TOE0 : Timer Output Enable Register 0

Base address:	TAU = 0x400A_2600															
Offset address:	0x013A															
Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	—	—	—	—	—	—	—	—	TOE[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
7:0	TOE[7:0]	Enabling or Disabling Timer Output for Channel n 0: Disables timer output. 1: Enables timer output.	R/W
15:8	—	These bits are read as 0. The write value should be 0.	R/W

The TOE0 register is used to enable or disable timer output of each channel.

Channel n for which timer output has been enabled becomes unable to rewrite the value of the TO0.TO[n] bit of timer output register 0 (TO0) described later by software, and the value reflecting the setting of the timer output function through the count operation is output from the timer output pin (TO0n).

The lower 8 bits of the TOE0 register can be set with an 8-bit memory manipulation instruction.

少量	象征	功能	R/W
11	TTH3	当通道 3 处于 8 位定时器状态时，触发停止高位 8 位定时器的运行。 模式 0: 无触发操作 1: TE0.TEH3 位被清零，计数操作停止。	R/W
15:12	—	这些位读取为 0。写入值应为 0。	R/W

注意：读取 TT0 寄存器时，始终读取 0。

TT0 寄存器是一个触发寄存器，用于停止每个通道的计数操作。

当此寄存器的某个位设置为 1 时，定时器通道使能状态寄存器 0 (TE0) 的相应位将被清除为 0。当操作停止时，TT 0.TT[n], TTH1, TTH3 位会立即被清除 ((TE0.TE[n], TEH1, TEH3))，因为它们是触发位。

16.2.10TIS0：定时器输入选择寄存器 0

基地址:	PORG = 0x400A_1000							
偏移地址:	0x0004							
位位置:	7	6	5	4	3	2	1	0
位域:	—	—	—	—	—	TIS[2]	—	—
重置后的值:	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
1:0	—	这些位读取为 0。写入值应为 0。	R/W
2	TIS[2]	选择与通道 5 一起使用的定时器输入。 0: 定时器输入引脚 (TI05) 的输入信号 1: 低速片上振荡器 (LOCO)	R/W
7:3	—	这些位读取为 0。写入值应为 0。	R/W

TIS0 寄存器用于选择单元 0 定时器输入的通道 5。

注意：确保所选定时器输入的高电平和低电平宽度均不小于 1/f_{MCK} + 10 ns。因此，当选择 FSUB 作为 PCLKB 时，TIS [2] 位不能设置为 1。

16.2.11 TOE0: 定时器输出使能寄存器 0

基地址:	TAU = 0x400A_2600															
偏移地址:	0x013A															
位位置:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域:	—	—	—	—	—	—	—	—	TOE[7:0]							
重置后的值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
7:0	TOE[7:0]	启用或禁用通道 n 的定时器输出 0: 禁用定时器输出。1: 启用定时器输出。	R/W
15:8	—	这些位读取为 0。写入值应为 0。	R/W

TOE0 寄存器用于启用或禁用每个通道的定时器输出。

已启用定时器输出的通道 n 将无法重写稍后由软件描述的定时器输出寄存器 0 (TO0) 的TO0.TO[n]位的值，并且通过计数操作反映定时器输出功能设置的值将从定时器输出引脚 (TO0n) 输出。

TOE0 寄存器的低 8 位可以通过 8 位内存操作指令进行设置。

TOE[7:0] bits (Enabling or Disabling Timer Output for Channel n)

When set TOE[n] = 0

The corresponding TO0.TO[n] bit does not reflect timer operation with this setting, so the output level of a TO0.TO[n] bit is fixed to the level written to the TO0 register.

Writing to the TO0.TO[n] bit is enabled and the level set in the TO0.TO[n] bit is output from the TO0n pin.

When set TOE[n] = 1

The corresponding TO0.TO[n] bit reflects timer operation with this setting, so the output waveform is generated.

Writing to the TO0.TO[n] bit is ignored.

Note: n: Channel number (n = 0 to 7).

16.2.12 TO0 : Timer Output Register 0

Base address:	TAU = 0x400A_2600															
Offset address:	0x0138															
Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	—	—	—	—	—	—	—	—	TO[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
7:0	TO[7:0]	Timer Output of Channel n 0: Timer output value is 0 1: Timer output value is 1	R/W
15:8	—	These bits are read as 0. The write value should be 0.	R/W

The TO0 register is a buffer register of timer output of each channel.

The value of each bit in this register is output from the timer output pin (TO0n) of each channel.

The TO0n bit on this register can be rewritten by software only when timer output is disabled (TOE0.TO[n] = 0). When timer output is enabled (TOE0.TO[n] = 1), rewriting this register by software is ignored, and the value is changed only by the timer operation.

To use the port functions multiplexed with the inputs and outputs of timer array units, select the function by setting PSEL[2:0] bits.

The lower 8 bits of the TO0 register can be set with an 8-bit memory manipulation instruction.

Note: n: Channel number (n = 0 to 7)

16.2.13 TOL0 : Timer Output Level Register 0

Base address:	TAU = 0x400A_2600															
Offset address:	0x013C															
Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	—	—	—	—	—	—	—	—	TOL[6:0]							—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
0	—	This bit is read as 0. The write value should be 0.	R/W
7:1	TOL[6:0]	Control of Timer Output of Channel n 0: Positive logic output (active-high) 1: Negative logic output (active-low)	R/W

TOE[7:0] 位（启用或禁用通道 n 的定时器输出）

当设置TOE[n] = 0

在此设置下，相应的TO0.TO[n]位不反映定时器操作，因此TO0.TO[n] 位的输出电平固定为写入 TO0 寄存器的电平。

写入 TO0.TO[n] 位被启用，TO0.TO[n] 位设置的电平从 TO0n 引脚输出。

当设置TOE[n] = 1

相应的TO0.TO[n]位反映了此设置下的定时器操作，因此会生成输出波形。

写入 TO0.TO[n] 位将被忽略。

注: n: 通道号(n = 0至7) 。

16.2.12TO0：定时器输出寄存器 0

基本地址:	TAU = 0x400A_2600															
偏移地址:	0x0138															
位位置:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域:	—	—	—	—	—	—	—	—	TO[7:0]							
重置后的值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
7:0	TO[7:0]	通道 n 的定时器输出 0: 定时器输出值为 0 1: 定时器输出值为 1	R/W
15:8	—	这些位读取为 0。写入值应为 0。	R/W

TO0 寄存器是每个通道定时器输出的缓冲寄存器。

该寄存器中每一位的值由每个通道的定时器输出引脚（TO0n）输出。

只有当定时器输出被禁用时，该寄存器的TO0n位才能被软件重写(TOE0.TO[n] = 0)位。当定时器输出被启用时，软件重写该寄存器的操作将被忽略，其值仅由定时器操作改变。

要使用与定时器阵列单元的输入和输出复用的端口功能，请通过设置 PSEL[2:0] 位来选择功能。

TO0 寄存器的低 8 位可以通过 8 位内存操作指令进行设置。

注: n: 频道号(n = 0至 7)

16.2.13TOL0：定时器输出电平寄存器 0

基地址:	TAU = 0x400A_2600															
偏移地址:	0x013C															
位位置:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域:	—	—	—	—	—	—	—	—	TOL[6:0]							—
重置后的值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
0	—	该位读取值为 0。写入值应为 0。	R/W
7:1	TOL[6:0]	控制通道 n 的定时器输出 0: 正逻辑输出（高电平有效） 1: 负逻辑输出（低电平有效）	R/W

Bit	Symbol	Function	R/W
15:8	—	These bits are read as 0. The write value should be 0.	R/W

The TOL0 register controls the timer output level of each channel.

The setting of the inverted output of channel n by this register is reflected at the timing of set or reset of the timer output signal while the timer output is enabled (TOE0.TOE[n] = 1) in the Slave channel output mode (TOM0.TOM[n] = 1). In the master channel output mode (TOM0.TOM[n] = 0), this register setting is invalid.

The lower 8 bits of the TOL0 register can be set with an 8-bit memory manipulation instruction.

Note: If the value of this register is rewritten during timer operation, the timer output logic is inverted when the timer output signal changes next, instead of immediately after the register value is rewritten.

Note: n: Channel number (n = 0 to 7)

16.2.14 TOM0 :Timer Output Mode Register 0

Base address:	TAU = 0x400A_2600															
Offset address:	0x013E															
Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	—	—	—	—	—	—	—	—	TOM[6:0]							—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
0	—	This bit is read as 0. The write value should be 0.	R/W
7:1	TOM[6:0]	Control of Timer Output Mode of Channel n 0: Master channel output mode (to produce toggled output by timer interrupt request signal (TAU0_TMI0n)) 1: Slave channel output mode (output is set by the timer interrupt request signal (TAU0_TMI0n) of the master channel, and reset by the timer interrupt request signal (TAU0_TMI0p) of the slave channel)	R/W
15:8	—	These bits are read as 0. The write value should be 0.	R/W

The TOM0 register is used to control the timer output mode of each channel.

When a channel is used for the independent channel operation function, set the corresponding bit of the channel to be used to 0.

When a channel is used for the simultaneous channel operation function (PWM output, one-shot pulse output, or multiple PWM output), set the corresponding bit of the master channel to 0 and the corresponding bit of the slave channel to 1.

The setting of each channel n by this register is reflected at the timing when the timer output signal is set or reset while the timer output is enabled (TOE0.TOE[n] = 1).

The lower 8 bits of the TOM0 register can be set with an 8-bit memory manipulation instruction.

Note: n: Channel number

n = 0 to 7 (n = 0, 2, 4, 6 for master channel)

p: Slave channel number

n < p ≤ 7

For details of the relation between the master channel and slave channel, see [section 16.3.1. Basic Rules of Simultaneous Channel Operation Function](#).

少量	象征	功能	R/W
15:8	—	这些位读取为 0。写入值应为 0。	R/W

TOL0 寄存器控制每个通道的定时器输出电平。

当定时器输出在从通道输出模式(TOM0.TOM[n] = 1)中使能(TOE0.TOE[n] = 1) 时，该寄存器对通道n反相输出的设置会反映在定时器输出信号的置位或复位时。在主通道输出模式(TOM0.TOM[n] = 0)中，此寄存器设置无效。

可以使用 8 位内存操作指令设置 TOL0 寄存器的低 8 位。

注意：如果在定时器操作期间重写此寄存器的值，则定时器输出逻辑会在定时器输出信号下次变化时反转，而不是在寄存器值被重写后立即反转。

注：n: 频道号(n = 0至 7)

16.2.14TOM0：定时器输出模式寄存器 0

基地址：	TAU = 0x400A _2600															
偏移地址：	0x013E															
位位置：	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域：	—	—	—	—	—	—	—	—	TOM[6:0]							—
重置后的值：	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
0	—	该位读取值为 0。写入值应为 0。	R/W
7:1	TOM[6:0]	通道 n 的定时器输出模式控制 0: 主通道输出模式（通过定时器中断请求信号（TAU0_TMI0n）产生切换输出） 1: 从通道输出模式（输出由主通道的定时器中断请求信号（TAU0_TMI0n）设置，由从通道的定时器中断请求信号（TAU0_TMI0p）复位）	R/W
15:8	—	这些位读取为 0。写入值应为 0。	R/W

TOM0寄存器用于控制每个通道的定时器输出模式。

当一个通道用于独立通道操作功能时，将要使用的通道的相应位设置为 0。

当一个通道用于同时通道操作功能（PWM 输出、单次脉冲输出或多个 PWM 输出）时，将主通道的相应位设置为 0，将从通道的相应位设置为 1。

该寄存器对每个通道 n 的设置反映在定时器输出信号置位或复位时，而定时器输出使能(TOE0.TOE[n] = 1)的时序上。

TOM0 寄存器的低 8 位可以通过 8 位内存操作指令进行设置。

注：n: 频道号

n = 0至 7(n = 0, 2, 4, 6（主通道）

p: 从属通道编号

n < p ≤ 7

有关主信道和从信道之间关系的详细信息，请参阅第 16.3.1 节“同时信道操作功能的基本规则”。

16.2.15 ISC : Input Switch Control Register

Base address: PORGA = 0x400A_1000							
Offset address: 0x0003							
Bit position:	7	6	5	4	3	2	1 0
Bit field:	—	—	—	—	—	SSIE0 0	ISC1 ISC0
Value after reset:	0	0	0	0	0	0	0 0

Bit	Symbol	Function	R/W
0	ISC0	Switching External Interrupt (IRQ0) Input 0: Uses the input signal of the IRQ0 pin as an external interrupt (normal operation) 1: Uses the input signal of the RXD0 pin as an external interrupt (wakeup signal detection)	R/W
1	ISC1	Switching Channel 7 Input of Timer Array Unit 0: Uses the input signal of the TI07 pin as a timer input (normal operation) 1: Input signal of the RXD0 pin is used as timer input (detects the wakeup signal and measures the low width of the break field and the pulse width of the sync field).	R/W
2	SSIE00	Setting of the SSI00 Input of Channel 0 in the Communications Through SPI00 in the Slave Mode 0: The SSI00 input is disabled. 1: The SSI00 input is enabled.	R/W
7:3	—	These bits are read as 0. The write value should be 0.	R/W

Note: When the LIN-bus communication function is used, select the input signal of the RXD0 pin by setting ISC1 to 1.

The ISC1 and ISC0 bits of the ISC register are used to implement LIN-bus communication operation by using TAU channel 7 in association with the serial array unit. When bit 0 is set to 1, the input signal of the serial data input (RXD0) pin is selected as an external interrupt (IRQ0) that can be used to detect a wakeup signal. When bit 1 is set to 1, the input signal of the serial data input (RXD0) pin is selected as a timer input, so that wake up signal can be detected, the low width of the break field, and the pulse width of the sync field can be measured by the timer.

The SSIE00 bit is used to control the SSI00 input of channel 0 in the communications through SPI00 in the slave mode. Reception and transmission do not proceed even if the serial clock is input while the SSI00 pin is being driven high. Reception and transmission proceed in response to an input of the serial clock according to the mode setting while the SSI00 pin is being driven low.

16.2.16 TNFEN : TAU Noise Filter Enable Register

Base address: PORGA = 0x400A_1000							
Offset address: 0x0001							
Bit position:	7	6	5	4	3	2	1 0
Bit field:	TNFE N07	TNFE N06	TNFE N05	TNFE N04	TNFE N03	TNFE N02	TNFE N01 TNFE N00
Value after reset:	0	0	0	0	0	0	0 0

Bit	Symbol	Function	R/W
0	TNFEN00	Enabling or Disabling Use of the Noise Filter for the TI00 Pin 0: Turns the noise filter off 1: Turns the noise filter on	R/W
1	TNFEN01	Enabling or Disabling Use of the Noise Filter for the TI01 Pin 0: Turns the noise filter off 1: Turns the noise filter on	R/W
2	TNFEN02	Enabling or Disabling Use of the Noise Filter for the TI02 Pin 0: Turns the noise filter off 1: Turns the noise filter on	R/W

16.2.15 ISC: 输入开关控制寄存器

基地址: PORGA= 0x400A_1000							
Offset address: 0x0003							
位位置:	7	6	5	4	3	2	1 0
位域:	—	—	—	—	—	SSIE0 0	ISC1 ISC0
重置后的值:	0	0	0	0	0	0	0 0

少量	象征	功能	R/W
0	ISC0	切换外部中断 (IRQ0) 输入 0: 使用 IRQ0 引脚的输入信号作为外部中断（正常运行） 1: 使用 RXD0 引脚的输入信号作为外部中断（唤醒信号检测）	R/W
1	ISC1	切换定时器阵列单元的通道 7 输入 0: 使用 TI07 引脚的输入信号作为定时器输入（正常操作） 1: 使用 RXD0 引脚的输入信号作为定时器输入（检测唤醒信号并测量断点低电平宽度和同步场脉冲宽度）。	R/W
2	SSIE00	在从模式下，通过 SPI00 进行通信时，设置通道 0 的 SSI00 输入 0: SSI00 输入已禁用。1: SSI00 输入已启用。	R/W
7:3	—	这些位读取为 0。写入值应为 0。	R/W

注意：当使用 LIN 总线通信功能时，将 ISC1 设置为 1 来选择 RXD0 引脚的输入信号。

ISC寄存器的ISC1和ISC0位用于通过TAU通道7与串行阵列单元配合实现LIN总线通信操作。当位0设置为1时，串行数据输入（RXD0）引脚的输入信号被选为外部中断（IRQ0），可用于检测唤醒信号。当位1设置为1时，串行数据输入（RXD0）引脚的输入信号被选为定时器输入，以便定时器可以检测唤醒信号、测量断点低电平的宽度以及同步电平的脉冲宽度。

SSIE00 位用于在从模式下通过 SPI00 进行通信时控制通道 0 的 SSI00 输入。即使在 SSI00 引脚被拉高时输入串行时钟，接收和发送也不会进行。当 SSI00 引脚被拉低时，接收和发送会根据模式设置响应串行时钟的输入而进行。

16.2.16TNFEN：TAU噪声滤波器使能寄存器

基地址: PORGA= 0x400A_1000							
Offset address: 0x0001							
位位置:	7	6	5	4	3	2	1 0
位域:	TNFE N07	TNFE N06	TNFE N05	TNFE N04	TNFE N03	TNFE N02	TNFE N01 TNFE N00
重置后的值:	0	0	0	0	0	0	0 0

少量	象征	功能	R/W
0	TNFEN00	启用或禁用 TI00 引脚的噪声滤波器 0: 关闭噪声滤波器 1: 开启噪声滤波器	R/W
1	TNFEN01	启用或禁用 TI01 引脚的噪声滤波器 0: 关闭噪声滤波器 1: 开启噪声滤波器	R/W
2	TNFEN02	启用或禁用 TI02 引脚的噪声滤波器 0: 关闭噪声滤波器 1: 开启噪声滤波器	R/W

Bit	Symbol	Function	R/W
3	TNFEN03	Enabling or Disabling Use of the Noise Filter for the TI03 Pin 0: Turns the noise filter off 1: Turns the noise filter on	R/W
4	TNFEN04	Enabling or Disabling Use of the Noise Filter for the TI04 Pin 0: Turns the noise filter off 1: Turns the noise filter on	R/W
5	TNFEN05	Enabling or Disabling Use of the Noise Filter for the TI05 Pin 0: Turns the noise filter off 1: Turns the noise filter on	R/W
6	TNFEN06	Enabling or Disabling Use of the Noise Filter for the TI06 Pin 0: Turns the noise filter off 1: Turns the noise filter on	R/W
7	TNFEN07	Enabling or Disabling Use of the Noise Filter for the TI07 Pin*1 0: Turns the noise filter off 1: Turns the noise filter on	R/W

Note 1. For the TI07 pin, it can be switched by setting the ISC1 bit of the ISC register.
ISC.ISC1 = 0: Whether to use the noise filter of the TI07 pin can be selected.
ISC.ISC1 = 1: Whether to use the noise filter of the RXD0 pin can be selected.

Note: The presence or absence of timer I/O pins of channel 0 to 7 depends on the product.

The TNFEN registers are used to set whether the noise filter can be used for the timer input signal to each channel.

Enable the noise filter by setting the corresponding bits to 1 on the pins in need of noise removal.

When the noise filter is enabled, after synchronization with the operating clock (f_{MCK})for the target channel, whether the signal keeps the same value for two clock cycles is detected.

When the noise filter is disabled, the input signal is only synchronized with the operating clock (f_{MCK}) for the target channel*1.

Note 1. For details, see (2) When valid edge of input signal via the TI0n pin is selected (TMR0n.CCS = 1), section 16.4.2. Timing of the Start of Counting, and section 16.6. Timer Input (TI0n) Control.

16.2.17 Registers Controlling Port Functions of Pins to be Used for Timer I/O

Set the following registers to control the port functions multiplexed with the inputs and outputs of timer array units.

- Pmn Direction Register (PDRm) or PDR bit of Port mn Pin Function Select Register (PmnPFS_A)
- PSEL[2:0] bits of Port mn Pin Function Select Register (PmnPFS_A)
- Do not use the analog input (ex. AN0xx)

For details, see the following sections.

- [section 15.2.2. PDRm : Pmn Direction Register \(m = 0 to 3, n = 00 to 15\)](#)
- [section 15.2.6. PmnPFS_A : Port mn Pin Function Select Register \(m = 1 to 3, n = 00 to 15\)](#)
- [section 15.2.7. P0nPFS_A : Port 0n Pin Function Select Register \(n = 10 to 13\)](#)
- [section 15.6. Peripheral Select Settings for Each Product](#)

When the pins multiplexed with TO01 to TO07 are to be used for outputs of timers, set the following registers.

- Set the PDRxx bit of Pmn Direction Register (PDRm) or PDR bit of Port mn Pin Function Select Register (PmnPFS_A)
Set the PDRm.PDRxx bit to 1. Or set the PmnPFS_A.PDR bit to 1.
- PSEL[2:0] bits of Port mn Pin Function Select Register (PmnPFS_A)
Select the TO0x function by setting PSEL[2:0] bits of Port mn Pin Function Select Register (PmnPFS_A)

When the pins multiplexed with TI01 to TI07 are to be used for inputs of timers, set the following registers.

- Set the PDRxx bit of Pmn Direction Register (PDRm) or PDR bit of Port mn Pin Function Select Register (PmnPFS_A)
Set the PDRm.PDRxx bit to 0. Or set the PmnPFS_A.PDR bit to 0.

少量	象征	功能	R/W
3	TNFEN03	启用或禁用 TI03 引脚的噪声滤波器 0: 关闭噪声滤波器 1: 开启噪声滤波器	R/W
4	TNFEN04	启用或禁用 TI04 引脚的噪声滤波器 0: 关闭噪声滤波器 1: 开启噪声滤波器	R/W
5	TNFEN05	启用或禁用 TI05 引脚的噪声滤波器 0: 关闭噪声滤波器 1: 开启噪声滤波器	R/W
6	TNFEN06	启用或禁用 TI06 引脚的噪声滤波器 0: 关闭噪声滤波器 1: 开启噪声滤波器	R/W
7	TNFEN07	启用或禁用 TI07 引脚*1 的噪声滤波器 0: 关闭噪声滤波器 1: 开启噪声滤波器	R/W

注1. 对于TI07引脚，可以通过设置ISC寄存器的ISC1位来切换。
ISC.ISC1 = 0: 可以选择是否使用TI07引脚的噪声滤波器。
ISC.ISC1 = 1: 可以选择是否使用 RXD0 引脚的噪声滤波器。

注意: 通道 0 至 7 的定时器I/O引脚是否存在取决于产品。

TNFEN 寄存器用于设置噪声滤波器是否可用于每个通道的定时器输入信号。

将需要消除噪声的引脚上的相应位设置为 1，即可启用噪声滤波器。

启用噪声滤波器后，在与目标通道的工作时钟(f_{MCK})同步后，检测信号是否在两个时钟周期内保持相同的值。

当噪声滤波器被禁用时，输入信号仅与目标通道*1 的工作时钟 (f_{MCK}) 同步。

注 1. 有关详细信息，请参阅 (2) 当通过 TI0n 引脚选择输入信号的有效边沿时 (TMR0n.CCS= 1)，第 16.4.2 节。计数开始的计时，以及第 16.6 节定时器输入 (TI0n) 控制。

16.2.17 寄存器控制用于定时器 I/O 的引脚的端口功能

设置以下寄存器以控制与定时器阵列单元的输入和输出复用的端口功能。

- Pmn方向寄存器（PDRm）或端口mn引脚功能选择寄存器（PmnPFS_A）的PDR位
- PSEL[2:0] 端口 mn 引脚功能选择寄存器 (PmnPFS_A) 的位
- 请勿使用模拟输入（例如 AN0xx）

详情请见以下章节。

- [第 15.2.2 节。PDRm:Pmn 方向寄存器\(m = 0至 3, n = 00 至 15\)](#)
- [第 15.2.6 节。PmnPFS_A: 端口 mn 引脚功能选择寄存器\(m = 1至 3, n = 00 至 15\)](#)
- [第 15.2.7 节。P0nPFS_A: 端口 0n 引脚功能选择寄存器\(n = 10至 13\)](#)
- [第 15.6 节各产品的外围设备选择设置](#)

当使用 TO01 至 TO07 复用的引脚作为定时器的输出时，设置以下寄存器。

- 设置Pmn方向寄存器（PDRm）的PDRxx位或端口mn引脚功能选择寄存器（PmnPFS_A）的PDR位。将PDRm.PDRxx位设置为1。或者将PmnPFS_A.PDR位设置为1。
- PSEL[2:0] 端口 mn 引脚功能选择寄存器 (PmnPFS_A) 的位
通过设置端口 mn 引脚功能选择寄存器 (PmnPFS_A) 的 PSEL[2:0] 位来选择 TO0x 功能。

当与 TI01 至 TI07 复用的引脚要用作定时器的输入时，设置以下寄存器。

- 设置Pmn方向寄存器（PDRm）的PDRxx位或端口mn引脚功能选择寄存器（PmnPFS_A）的PDR位。将PDRm.PDRxx位设置为0。或者将PmnPFS_A.PDR位设置为0。

- PSEL[2:0] bits of Port mn Pin Function Select Register (PmnPFS_A)
Select the TI0x function by setting PSEL[2:0] bits of Port mn Pin Function Select Register (PmnPFS_A)

16.3 Basic Rules of Timer Array Unit

16.3.1 Basic Rules of Simultaneous Channel Operation Function

When simultaneously using multiple channels, namely, a combination of a master channel (a reference timer mainly counting the cycle) and slave channels (timers operating according to the master channel), the following rules apply.

1. Only an even channel (channels 0, 2, 4, etc.) can be set as a master channel.
2. Any channel, except channel 0, can be set as a slave channel.
3. The slave channel must be lower than the master channel.
Example: If channel 2 is set as a master channel, channel 3 or those that follow (channels 3, 4, 5, etc.) can be set as a slave channel.
4. Two or more slave channels can be set for one master channel.
5. When two or more master channels are to be used, slave channels with a master channel between them may not be set.
Example: If channels 0 and 4 are set as master channels, channels 1 to 3 can be set as the slave channels of master channel 0. Channels 5 to 7 cannot be set as the slave channels of master channel 0.
6. The operating clock for a slave channel in combination with a master channel must be the same as that of the master channel. The CKS[1:0] bits (bit 15, 14 of timer mode register 0n (TMR0n)) of the slave channel that operates in combination with the master channel must be the same value as that of the master channel.
7. A master channel can transmit TAU0_TMI0n (interrupt), start software trigger, and count clock to the lower channels.
8. A slave channel can use TAU0_TMI0n (interrupt), a start software trigger, or the count clock of the master channel as a source clock, but cannot transmit its own TAU0_TMI0n (interrupt), start software trigger, or count clock to channels with lower channel numbers.
9. A master channel cannot use TAU0_TMI0n (interrupt), a start software trigger, or the count clock from the other higher master channel as a source clock.
10. To simultaneously start channels that operate in combination, the channel start trigger bit (TS0.TS[n]) of the channels in combination must be set at the same time.
11. During the counting operation, a TS0.TS[n] bit of a master channel or TS0.TS[n] bit of all channels which are operating simultaneously can be set. It cannot be applied to TS0.TS[n] bit of slave channels alone.
12. To stop the channels in combination simultaneously, the channel stop trigger bit (TT0.TT[n]) of the channels in combination must be set at the same time.
13. CK02 and CK03 cannot be selected while channels are operating simultaneously, because the operating clocks of master channels and slave channels have to be synchronized.
14. Timer mode register 00 (TMR00) has no master bit (it is fixed to 0). However, as channel 0 is the highest channel, it can be used as a master channel during simultaneous operation.

The rules of the simultaneous channel operation function are applied in a channel group (a master channel and slave channels forming one simultaneous channel operation function).

If two or more channel groups that do not operate in combination are specified, the basic rules of the simultaneous channel operation function in this section do not apply to the channel groups.

Note: n: Channel number (n = 0 to 7)

Figure 16.19 shows an example of how TAU can be used.

- PSEL[2:0] 端口 mn 引脚功能选择寄存器 (PmnPFS_A) 的位
通过设置端口 mn 引脚功能选择寄存器 (PmnPFS_A) 的 PSEL[2:0] 位来选择 TI0x 功能。

16.3 定时器阵列单元的基本规则

16.3.1 同时信道操作功能的基本规则

当同时使用多个通道时，即主通道（主要计数周期的参考定时器）和从通道（根据主通道运行的定时器）的组合，适用以下规则。

1. 只有偶数通道（通道 0、2、4 等）才能设置为主通道。
2. 除通道 0 外，任何通道都可以设置为从属通道。
3. 从通道必须低于主通道。
例如：如果将通道 2 设置为主通道，则可以将通道 3 或后续通道（通道 3、4、5 等）设置为从通道。
4. 一个主通道可以设置两个或多个从属通道。
5. 当使用两个或多个主通道时，不能设置与主通道位于同一位置的从通道。
例如：如果将通道 0 和 4 设置为主通道，则通道 1 至 3 可以设置为主通道 0 的从通道。通道 5 至 7 不能设置为主通道 0 的从通道。
6. 从通道与主通道组合运行时，其工作时钟必须与主通道的工作时钟相同。与主通道组合运行时，从通道的 CKS[1:0] 位（定时器模式寄存器 0n (TMR0n) 的第 15 位和第 14 位）的值必须与主通道的相应值相同。
7. 主通道可以向低级通道发送 TAU0_TMI0n（中断）、启动软件触发和计数时钟。
8. 从通道可以使用 TAU0_TMI0n（中断）、启动软件触发或主通道的计数时钟作为源时钟，但不能将自己的 TAU0_TMI0n（中断）、启动软件触发或计数时钟发送给通道号较低的通道。
9. 主通道不能使用 TAU0_TMI0n（中断）、启动软件触发或来自其他更高主通道的计数时钟作为源时钟。
10. 要同时启动组合运行的通道，必须同时设置组合通道的通道启动触发位(TS0.TS[n])。
11. 在计数操作期间，可以设置主通道的TS0.TS[n]位或同时运行的所有通道的 TS0.TS[n] 位。但不能单独应用于从通道的 TS0.TS[n] 位。
12. 要同时停止组合通道，必须同时设置组合通道的通道停止触发位(TT0.TT[n])。
13. 当通道同时运行时，不能选择 CK02 和 CK03，因为主通道和从通道的工作时钟必须同步。
14. 定时器模式寄存器 00 (TMR00) 没有主位（固定为 0）。但是，由于通道 0 是最高通道，因此在同时操作期间可以将其用作主通道。

同时通道操作功能的规则适用于通道组（一个主通道和多个从通道组成一个同时通道操作功能）。

如果指定了两个或多个不组合运行的通道组，则本节中同时通道运行功能的基本规则不适用于这些通道组。

注: n: 频道号(n = 0至 7)

图 16.19 显示了 TAU 的使用示例。

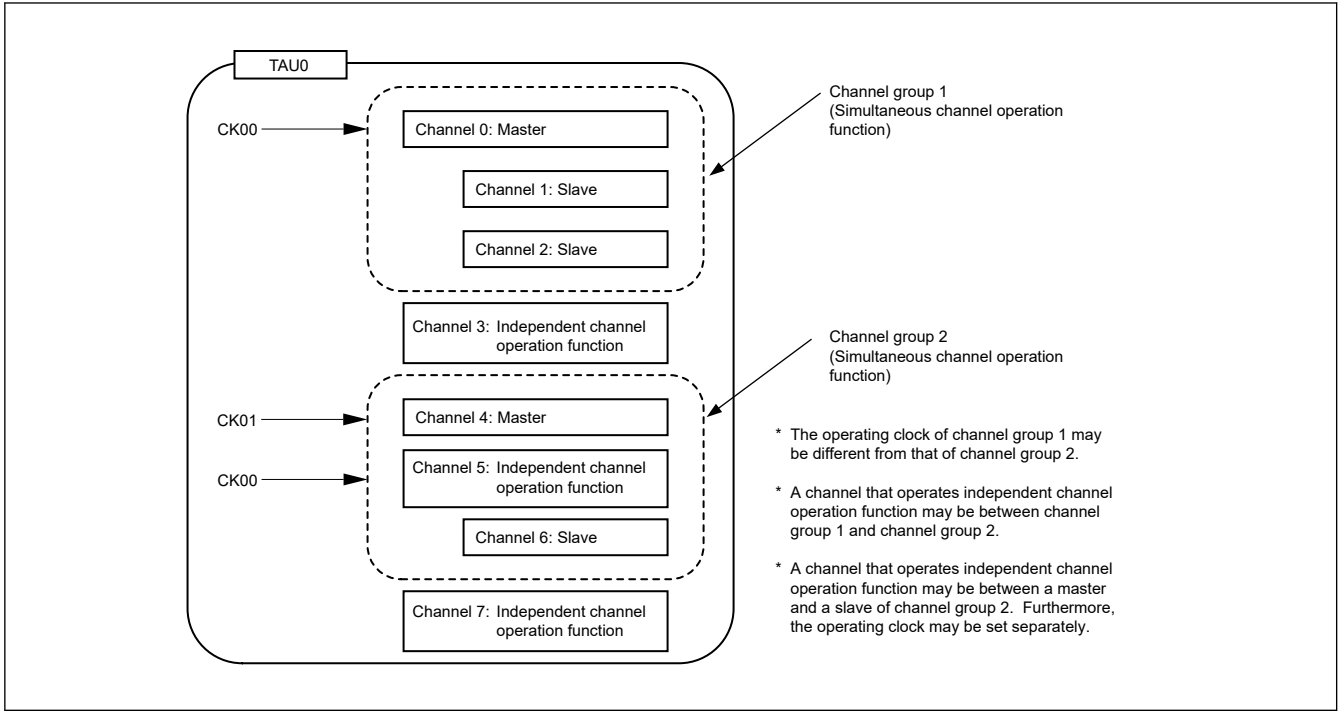


Figure 16.19 TAU utilization example

16.3.2 Basic Rules of 8-bit Timer Operation Function (Channels 1 and 3 only)

The 8-bit timer operation function makes it possible to use a 16-bit timer channel in a configuration consisting of two 8-bit timer channels.

This function can only be used for channels 1 and 3, and there are several rules for using it.

The basic rules for this function are as follows:

1. The 8-bit timer operation function applies only to channels 1 and 3.
2. When using 8-bit timers, set the SPLIT bit of timer mode register 0n (TMR0n) to 1.
3. The higher 8 bits can be operated as the interval timer function.
4. At the start of operation, the higher 8 bits output TAU0_TMI01H and TAU0_TMI03H (an interrupt) (which is the same operation performed when the TMR0n.OPIRQ bit is set to 1).
5. The operation clock of the higher 8 bits is selected according to the CKS[1:0] bits of the lower-bit TMR0n register.
6. For the higher 8 bits, the TS0.TSH1 and TSH3 bits are manipulated to start channel operation and the TT0.TTH1 and TTH3 bits are manipulated to stop channel operation. The channel state can be checked using the TE0.TEH1 and TEH3 bits.
7. The lower 8 bits operate according to the TMR0n register settings. The following three functions support operation of the lower 8 bits:
 - Interval timer function and square wave output function
 - External event counter function
 - Delay count function
8. For the lower 8 bits, the TS0.TS[1] and TS[3] bits are manipulated to start channel operation and the TT0.TT[1] and TT[3] bits are manipulated to stop channel operation. The channel state can be checked using the TE0.TE[1] and TE[3] bits.
9. During 16-bit operation, manipulating the TS0.TSH1, TSH3, TT0.TTH1, and TTH3 bits are invalid. The TS0.TS[1], TS[3], TT0.TT[1], and TT[3] bits are manipulated to operate channels 1 and 3. The TE0.TEH1, and TEH3 bits are not changed.
10. For the 8-bit timer function, the simultaneous operation functions (one-shot pulse, PWM, and multiple PWM) cannot be used.

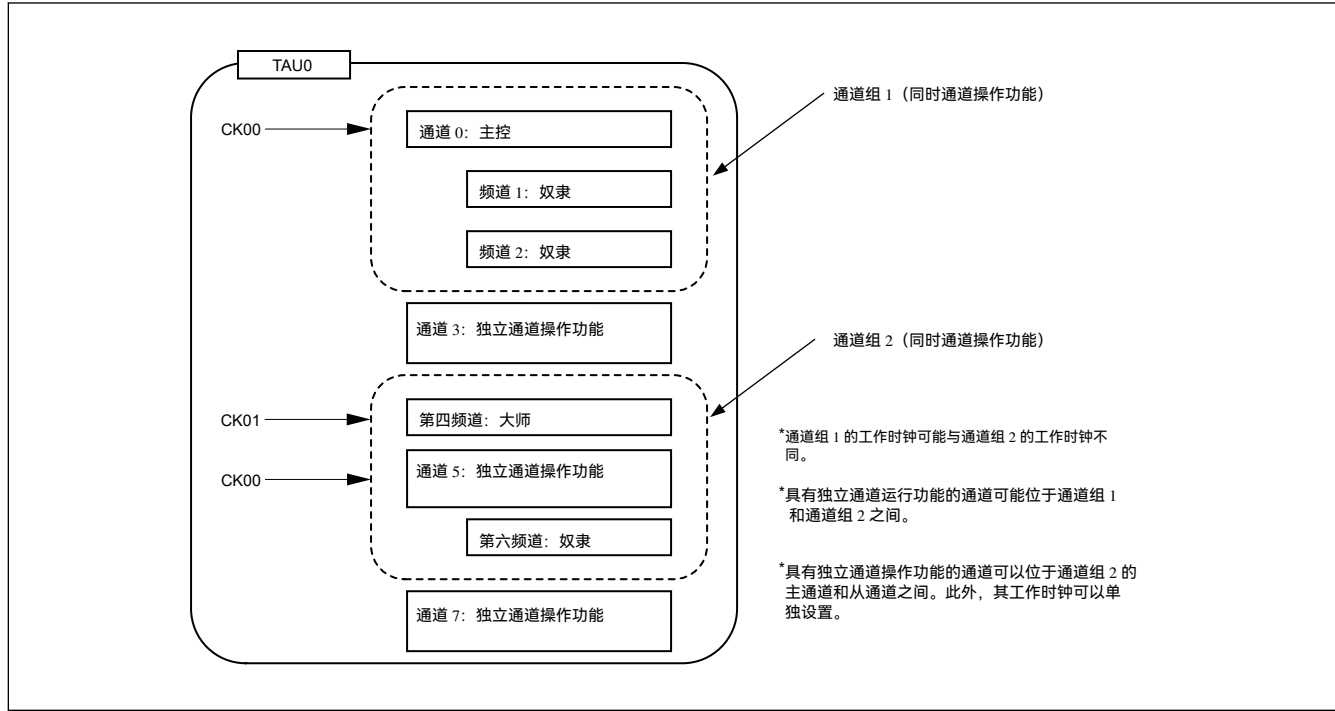


图 16.19 TAU 利用率示例

16.3.2 8 位定时器操作功能的基本规则（仅限通道 1 和 3）

8 位定时器操作功能使得在由两个 8 位定时器通道组成的配置中使用 16 位定时器通道成为可能。

此功能只能用于通道 1 和 3，并且使用时有几条规则。

该函数的基本规则如下：

1. 8 位定时器操作功能仅适用于通道 1 和 3。
2. 使用 8 位定时器时，将定时器模式寄存器 0n (TMR0n) 的 SPLIT 位设置为 1。
3. 高 8 位可以作为间隔定时器功能。
4. 在操作开始时，较高的 8 位输出 TAU0_TMI01H 和 TAU0_TMI03H（中断）（这与 TMR0n.OPIRQ 位设置为 1 时执行的操作相同）。
5. 根据低位 TMR0n 寄存器的 CKS[1:0] 位选择高 8 位的操作时钟。
6. 对于高 8 位，TS0、TSH1 和 TSH3 位用于启动信道操作，TT0、TTH1 和 TTH3 位用于停止信道操作。信道状态可以通过 TE0、TEH1 和 TEH3 位进行检查。
7. 低 8 位根据 TMR0n 寄存器的设置进行操作。以下三个功能支持低 8 位的操作：
 - 间隔定时器功能和方波输出功能
 - 外部事件计数器功能
 - 延迟计数功能
8. 对于低 8 位，TS0、TS[1] 和 TS[3] 位用于启动信道操作，TT0、TT[1] 和 TT[3] 位用于停止信道操作。信道状态可以通过 TE0、TE[1] 和 TE[3] 位进行检查。
9. 在 16 位操作期间，对 TS0.TSH1、TSH3、TT0.TTH1 和 TTH3 位进行操作无效。TS0.TS[1]、TS[3]、TT0.TT[1] 和 TT[3] 位用于操作通道 1 和通道 3。TE0.TEH1 和 TEH3 位保持不变。
10. 对于 8 位定时器功能，不能同时使用操作功能（单脉冲、PWM 和多 PWM）。

Note: n: Channel number (n = 1, 3)

16.4 Operations of Counters

16.4.1 Count Clock (f_{TCLK})

The count clock (f_{TCLK}) of the timer array unit can be selected between following by CCS bit of timer mode register 0n (TMR0n).

- Operation clock (f_{MCK}) specified by the TMR0n.CKS[1:0] bits
- Valid edge of input signal input from the TI0n pin

Because the timer array unit is designed to operate in synchronization with PCLKB, the timings of the count clock (f_{TCLK}) are shown below.

(1) When operation clock (f_{MCK}) specified by the TMR0n.CKS[1:0] bits is selected (TMR0n.CCS = 0)

The count clock (f_{TCLK}) is between PCLKB to PCLKB/2¹⁵ by setting of timer clock select register 0 (TPS0). When a divided PCLKB is selected, however, the clock selected in TPS0n register, but a signal which becomes high level for one period of PCLKB from its rising edge. When a PCLKB is selected, fixed to high level

Counting of timer counter register 0n (TCR0n) delayed by one period of PCLKB from rising edge of the count clock, because of synchronization with PCLKB. But, this is described as "counting at rising edge of the count clock", as a matter of convenience.

Figure 16.20 shows the count clock (f_{TCLK}) timing from PCLKB when TMR0n.CCS = 0.

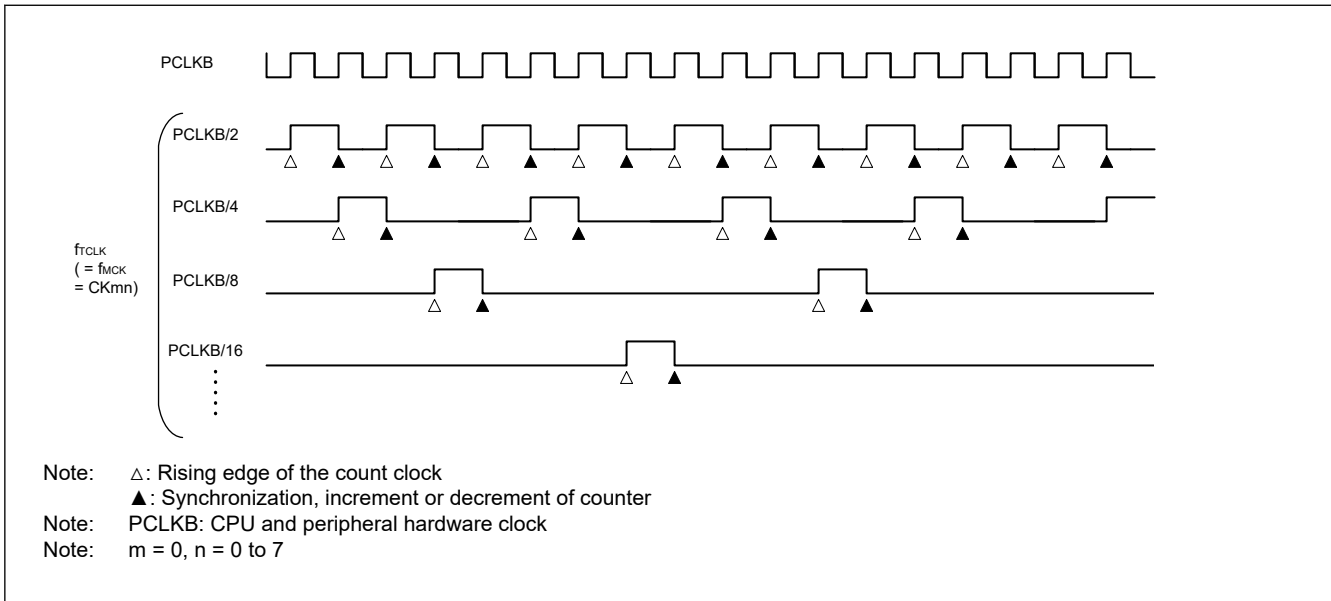


Figure 16.20 Timing of PCLKB and count clock (f_{TCLK}) (when TMR0n.CCS = 0)

(2) When valid edge of input signal via the TI0n pin is selected (TMR0n.CCS = 1)

The count clock (f_{TCLK}) becomes the signal that detects valid edge of input signal via the TI0n pin and synchronizes next rising f_{MCK} . The count clock (f_{TCLK}) is delayed for 1 to 2 period of f_{MCK} from the input signal via the TI0n pin (when a noise filter is used, the delay becomes 3 to 4 clock).

Counting of timer counter register 0n (TCR0n) delayed by one period of PCLKB from rising edge of the count clock, because of synchronization with PCLKB. But, this is described as "counting at valid edge of input signal via the TI0n pin", as a matter of convenience.

Figure 16.21 shows the count clock (f_{TCLK}) timing from PCLKB when TMR0n.CCS = 1 and noise filter unused.

注: n: 频道号(n = 1, 3)

16.4 计数器的操作

16.4.1 计数时钟 (f_{TCLK})

定时器阵列单元的计数时钟 (f_{TCLK}) 可通过定时器模式寄存器 0n (TMR0n) 的 CCS 位进行选择。

- 操作时钟 (f_{MCK}) 由 TMR0n.CKS[1:0] 位指定
- 来自 TI0n 引脚的输入信号的有效边沿

由于定时器阵列单元设计为与 PCLKB 同步运行，因此计数时钟 (f_{TCLK}) 的时序如下所示。

(1) 当选择由 TMR0n.CKS[1:0] 位指定的操作时钟 (f_{MCK})(TMR0n.CCS = 0)

计数时钟 (f_{TCLK}) 介于 PCLKB 和 PCLKB/2¹⁵ 之间，由定时器时钟选择寄存器 0 (TPS0) 的设置决定。然而，当选择分频 PCLKB 时，TPS0n 寄存器中选择的时钟信号，会在其上升沿之后的一个周期内保持高电平。当选择 PCLKB 时，该信号会固定为高电平。

由于与 PCLKB 同步，定时器计数器寄存器 0n (TCR0n) 的计数比计数时钟的上升沿延迟一个 PCLKB 周期。但为了方便起见，这里将其描述为“在计数时钟的上升沿计数”。

图 16.20 显示了 TMR0n 时 PCLKB 的计数时钟 (f_{TCLK}) 时序。CCS = 0。

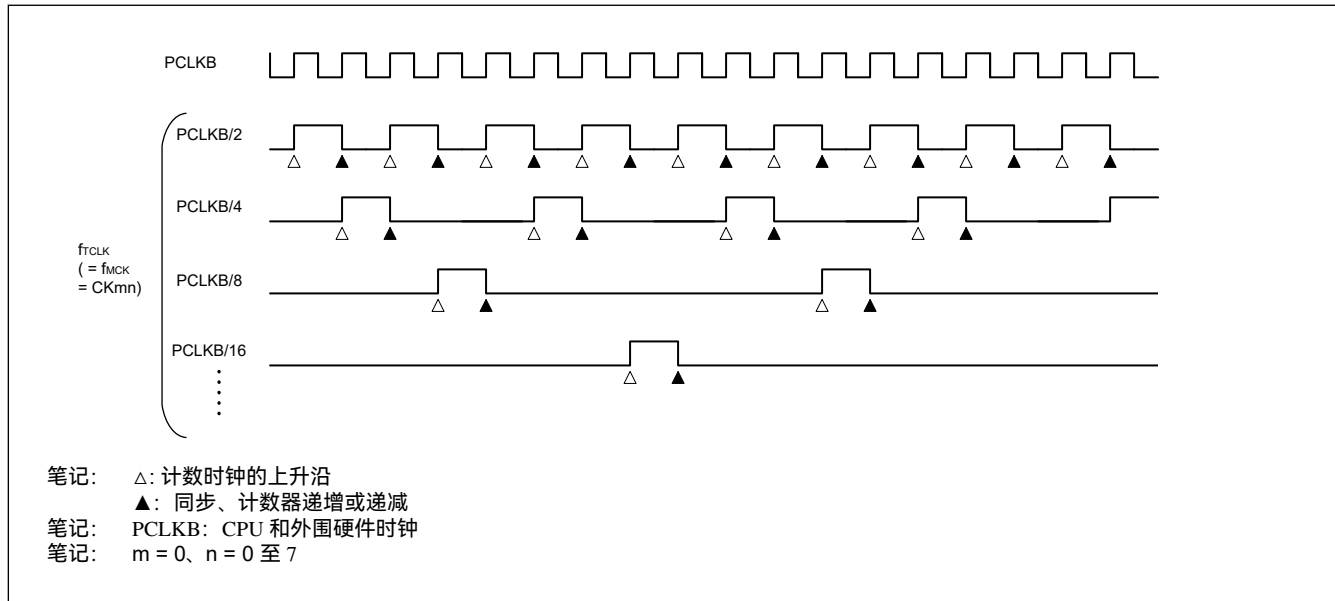


图 16.20 PCLKB 和计数时钟 (f_{TCLK}) 的时序 (当 TMR0n.CCS= 0 时)

(2) 当通过 TI0n 引脚的输入信号的有效边沿被选中时 (TMR0n.CCS = 1)

计数时钟 (f_{TCLK}) 通过 TI0n 引脚检测输入信号的有效边沿，并同步下一个上升沿 f_{MCK} 。计数时钟 (f_{TCLK}) 从 TI0n 引脚的输入信号延迟 1 到 2 个周期的 f_{MCK} (当使用噪声滤波器时，延迟变为 3 到 4 个时钟周期)。

由于与 PCLKB 同步，定时器计数器寄存器 0n (TCR0n) 的计数比计数时钟的上升沿延迟一个 PCLKB 周期。但为了方便起见，这里将其描述为“通过 TI0n 引脚在输入信号的有效边沿进行计数”。

图 16.21 显示了当 TMR0n 时，来自 PCLKB 的计数时钟 (f_{TCLK}) 的时序。CCS = 1 和噪声滤波器未使用。

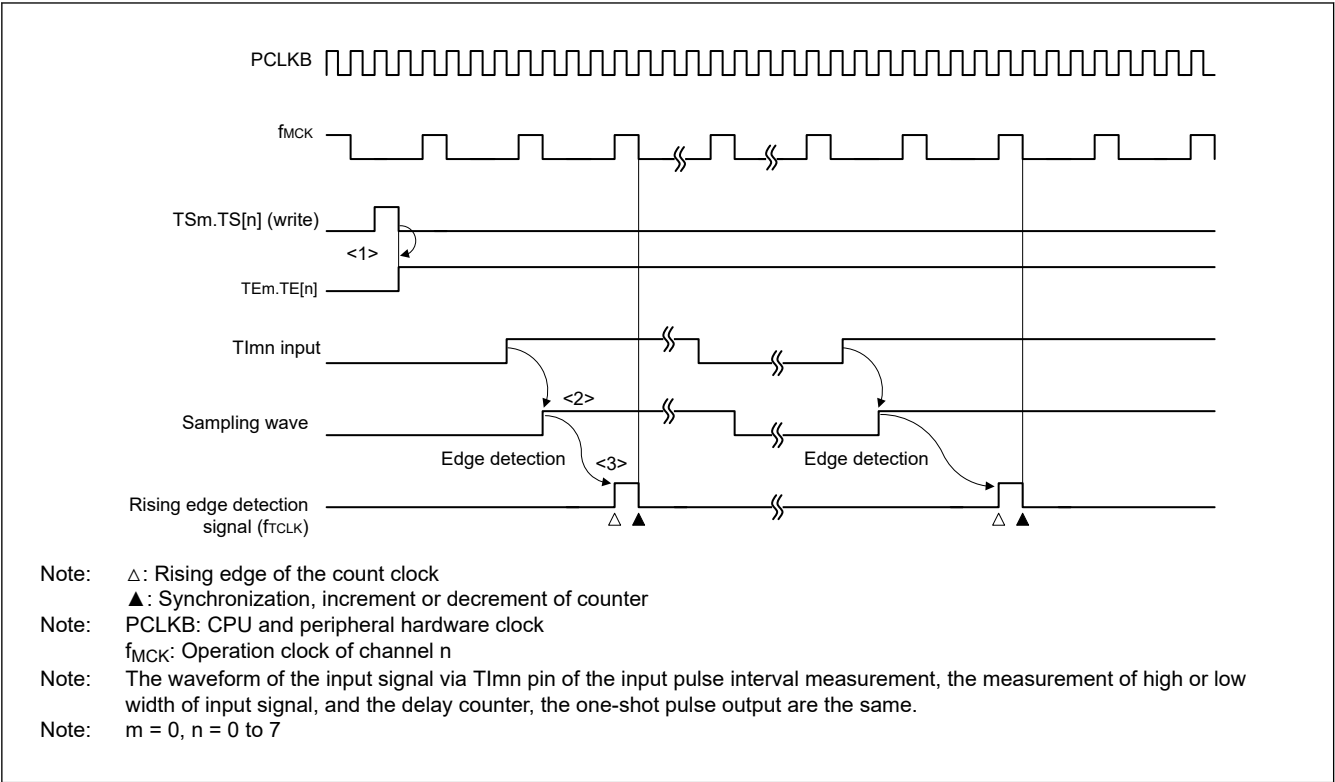


Figure 16.21 Timing of PCLKB and count clock (f_{TCLK}) (when TMR0n.CCS = 1, noise filter unused)

- <1> Setting TS0.TS[n] bit to 1 enables the timer to be started and to become wait state for valid edge of input signal via the TI0n pin.
- <2> The rise of input signal via the TI0n pin is sampled by f_{MCK} .
- <3> The edge is detected by the rising of the sampled signal and the detection signal (count clock) is output.

16.4.2 Timing of the Start of Counting

Timer counter register 0n (TCR0n) becomes enabled to operation by setting of TS[n] bit of timer channel start register 0 (TS0).

Operations from count operation enabled state to timer counter register 0n (TCR0n) count start is shown in Table 16.11.

Table 16.11 Operations from the count operation enabled state to the start of counting by a Timer Counter Register 0n (TCR0n) (1 of 2)

Timer operation mode	Operation when TS0.TS[n] = 1 is set
● Interval timer mode	No operation is carried out from start trigger detection (TS0.TS[n] = 1) until count clock generation. The first count clock loads the value of the TDR0n register to the TCR0n register and the subsequent count clock performs count down operation (see (1) Operation in interval timer mode).
● Event counter mode	Writing 1 to the TS0.TS[n] bit loads the value of the TDR0n register to the TCR0n register. If detect edge of TI0n input. The subsequent count clock performs count down operation (see (2) Operation in event counter mode).
● Capture mode	No operation is carried out from start trigger detection (TS0.TS[n] = 1) until count clock generation. The first count clock loads 0x0000 to the TCR0n register and the subsequent count clock performs count up operation (see (3) Operation in capture mode (input pulse interval measurement)).
● One-count mode	The waiting-for-start-trigger state is entered by writing 1 to the TS0.TS[n] bit while the timer is stopped (TE0.TE[n] = 0). No operation is carried out from start trigger detection until count clock generation. The first count clock loads the value of the TDR0n register to the TCR0n register and the subsequent count clock performs count down operation (see (4) Operation in one-count mode).

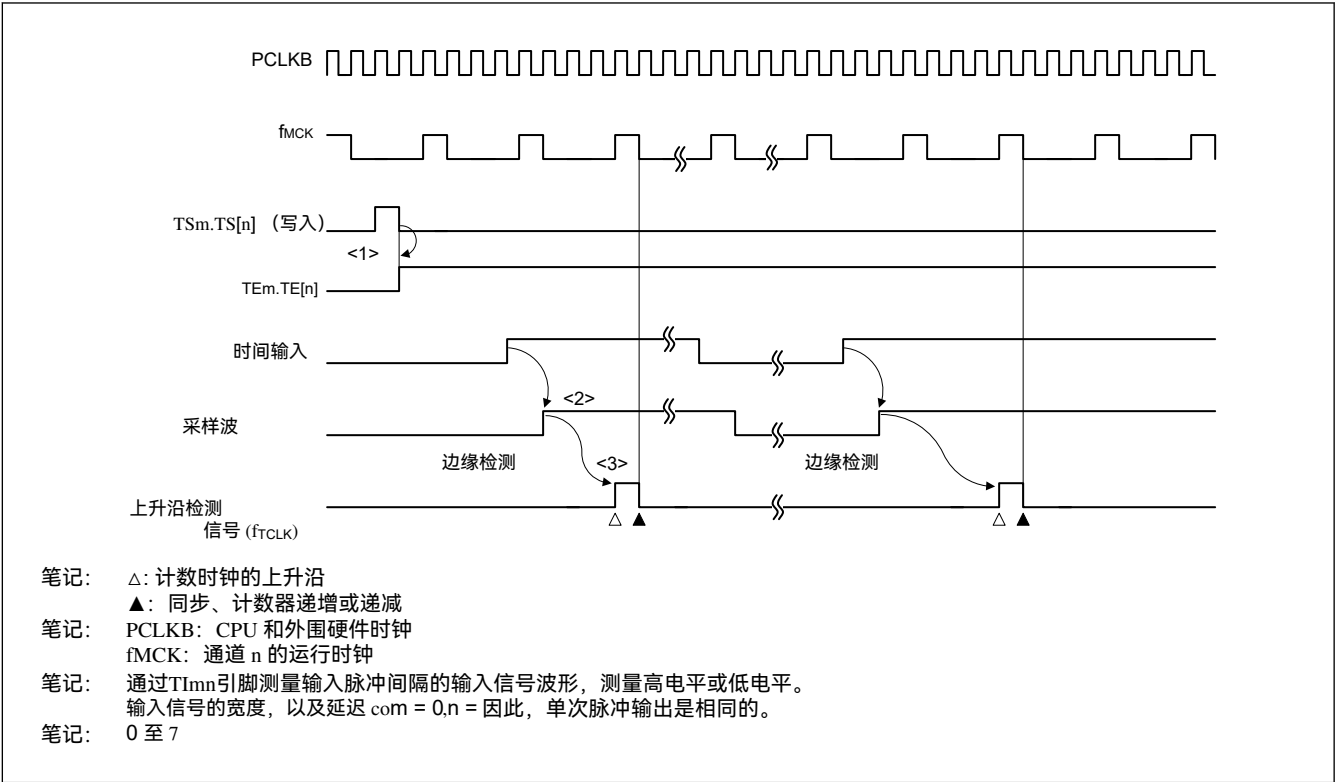


图 16.21 PCLKB 和计数时钟 (f_{TCLK}) 的时序 (当 TMR0n.CCS= 1时，未使用噪声滤波器)

将<1>位设置为 1 可使定时器启动，并使其进入等待状态，等待通过 TI0n 引脚输入信号的有效边沿。

- <2>通过 TI0n 引脚的输入信号上升沿由 f_{MCK} 采样。
- <3>通过采样信号的上升沿检测边沿，并输出检测信号（计数时钟）。

16.4.2 开始计数的时间

通过设置定时器通道起始寄存器 0 (TS0) 的TS[n] 位，定时器计数器寄存器 0n (TCR0n) 即可启用。

表 16.11 显示了从计数操作使能状态到定时器计数器寄存器 0n (TCR0n) 计数开始的操作。

表 16.11 定时器计数器从计数操作启用状态到开始计数的操作
寄存器 0n (TCR0n) (1/2)

定时器操作模式	当TS0.TS[n] = 1 设置时的操作
● 间隔定时器模式	从开始触发检测(TS0.TS[n] = 1) 到计数时钟生成期间，不执行任何操作。第一个计数时钟将 TDR0n 寄存器的值加载到 TCR0n 寄存器，随后的计数时钟执行倒计时操作（参见 (1) 间隔定时器模式下的操作）。
● 事件计数器模式	将 1 写入 TS0.TS[n] 位，即可将 TDR0n 寄存器的值加载到 TCR0n 寄存器。如果检测到 TI0n 输入的边沿，则后续的计数时钟将执行倒计时操作（参见 (2) 事件计数器模式下的操作）。
● 捕获模式	从开始触发检测(TS0.TS[n] = 1) 到计数时钟生成期间，不执行任何操作。第一个计数时钟将 0x0000 加载到 TCR0n 寄存器，随后的计数时钟执行向上计数操作（参见 (3) 捕获模式下的操作（输入脉冲间隔测量））。
● 单次计数模式	当定时器停止时，通过向 TS0.TS[n] 位写入 1 进入等待启动触发状态 (TE0.TE[n] = 0) 从开始触发检测到计数时钟生成期间，不执行任何操作。第一个计数时钟将 TDR0n 寄存器的值加载到 TCR0n 寄存器，随后的计数时钟执行倒计时操作（参见 (4) 单次计数模式下的操作）。

Table 16.11 Operations from the count operation enabled state to the start of counting by a Timer Counter Register 0n (TCR0n) (2 of 2)

Timer operation mode	Operation when TS0.TS[n] = 1 is set
● Capture & one-count mode	The waiting-for-start-trigger state is entered by writing 1 to the TS0.TS[n] bit while the timer is stopped (TE0.TE[n] = 0). No operation is carried out from start trigger detection until count clock generation. The first count clock loads 0x0000 to the TCR0n register and the subsequent count clock performs count up operation (see (5) Operation in capture & one-count mode (high-level width measurement)).

16.4.3 Operations of Counters

Here, the counter operation in each mode is explained.

(1) Operation in interval timer mode

- <1> Operation is enabled (TE0.TE[n] = 1) by writing 1 to the TS0.TS[n] bit. Timer counter register 0n (TCR0n) holds the initial value until count clock generation.
- <2> A start trigger is generated at the first count clock after operation is enabled.
- <3> When the TMR0n.OPIRQ bit is set to 1, TAU0_TMI0n is generated by the start trigger.
- <4> By the first count clock after the operation enable, the value of timer data register 0n (TDR0n) is loaded to the TCR0n register and counting starts in the interval timer mode.
- <5> When the TCR0n register counts down and its count value is 0x0000, TAU0_TMI0n is generated and the value of timer data register 0n (TDR0n) is loaded to the TCR0n register and counting keeps on.

Figure 16.22 shows the timing during operation in interval timer mode.

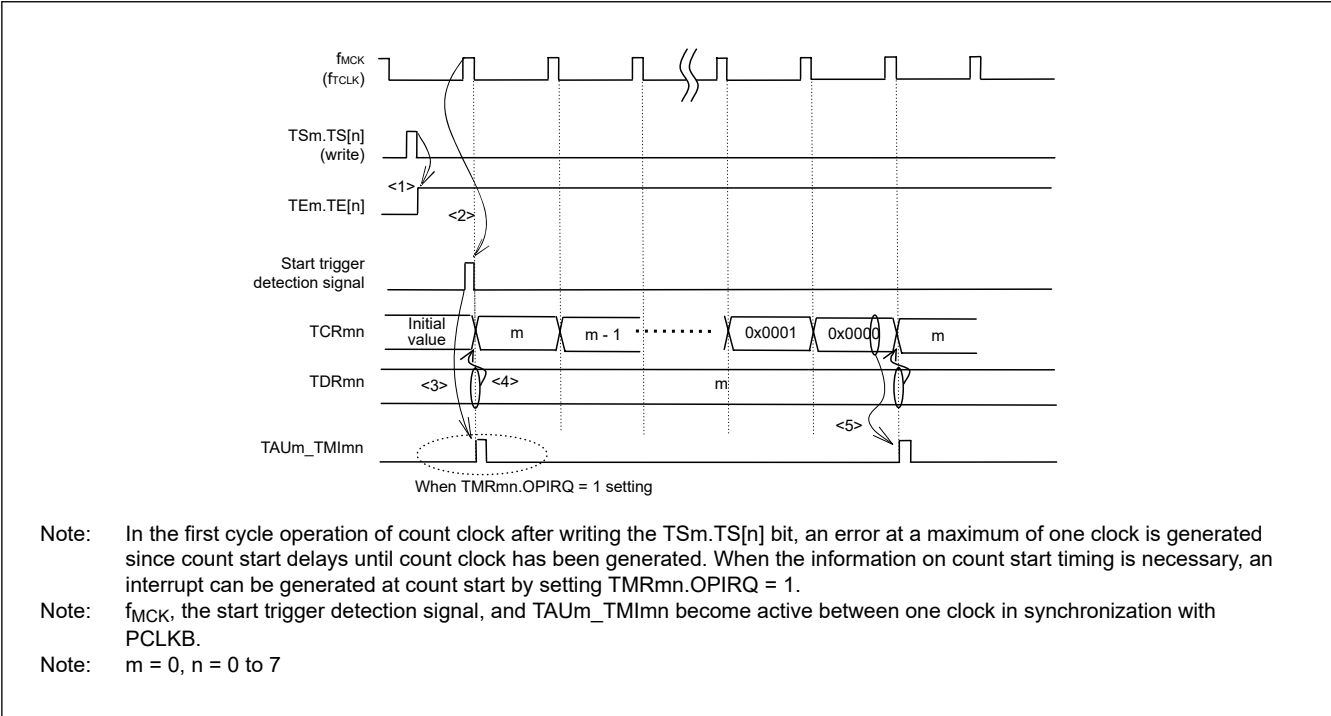


Figure 16.22 Timing during operation in interval timer mode

(2) Operation in event counter mode

- <1> Timer counter register 0n (TCR0n) holds its initial value while operation is stopped (TE0.TE[n] = 0).
- <2> Operation is enabled (TE0.TE[n] = 1) by writing 1 to the TS0.TS[n] bit.
- <3> As soon as 1 has been written to the TS0.TS[n] bit and 1 has been set to the TE0.TE[n] bit, the value of timer data register 0n (TDR0n) is loaded to the TCR0n register to start counting.
- <4> After that, the TCR0n register value is counted down according to the count clock of the valid edge of the TI0n input.

表 16.11 定时器计数器从计数操作启用状态到开始计数的操作 寄存器 0n (TCR0n) (2/2)

定时器操作模式	当TS0.TS[n] = 1 设置时的操作
● 捕捉与单次计数模式	当定时器停止时，通过向 TS0.TS[n] 位写入 1 进入等待启动触发状态 (TE0.TE[n] = 0) 从开始触发检测到计数时钟生成期间，不执行任何操作。第一个计数时钟将0x0000加载到 TCR0n 寄存器，随后的计数时钟执行计数加一操作（参见 (5) 捕获和单次计数模式下的操作（高电平宽度测量））。

16.4.3 计数器的操作

这里解释了每种模式下的计数器操作。

(1) 间隔定时器模式操作

- <1>操作通过向TS0.TS[n]位写入 1 来启用。定时器计数器寄存器 0n (TCR0n) 保存初始值，直到计数时钟生成。
- <2> A启动触发是在操作启用后的第一个计数时钟处生成的。
- <3>当 TMR0n.OPIRQ 位设置为 1 时，启动触发器会生成 TAU0_TMI0n。
- <4>在操作使能后的第一个计数时钟周期，定时器数据寄存器 0n (TDR0n) 的值被加载到 TCR0n 寄存器中，并在间隔定时器模式下开始计数。
- <5>当 TCR0n 寄存器倒数且其计数值为0x0000, 时，生成 TAU0_TMI0n，并将定时器数据寄存器0n(TDR0n) 的值加载到 TCR0n 寄存器中，计数继续进行。

图 16.22 显示了间隔定时器模式下的运行时间。

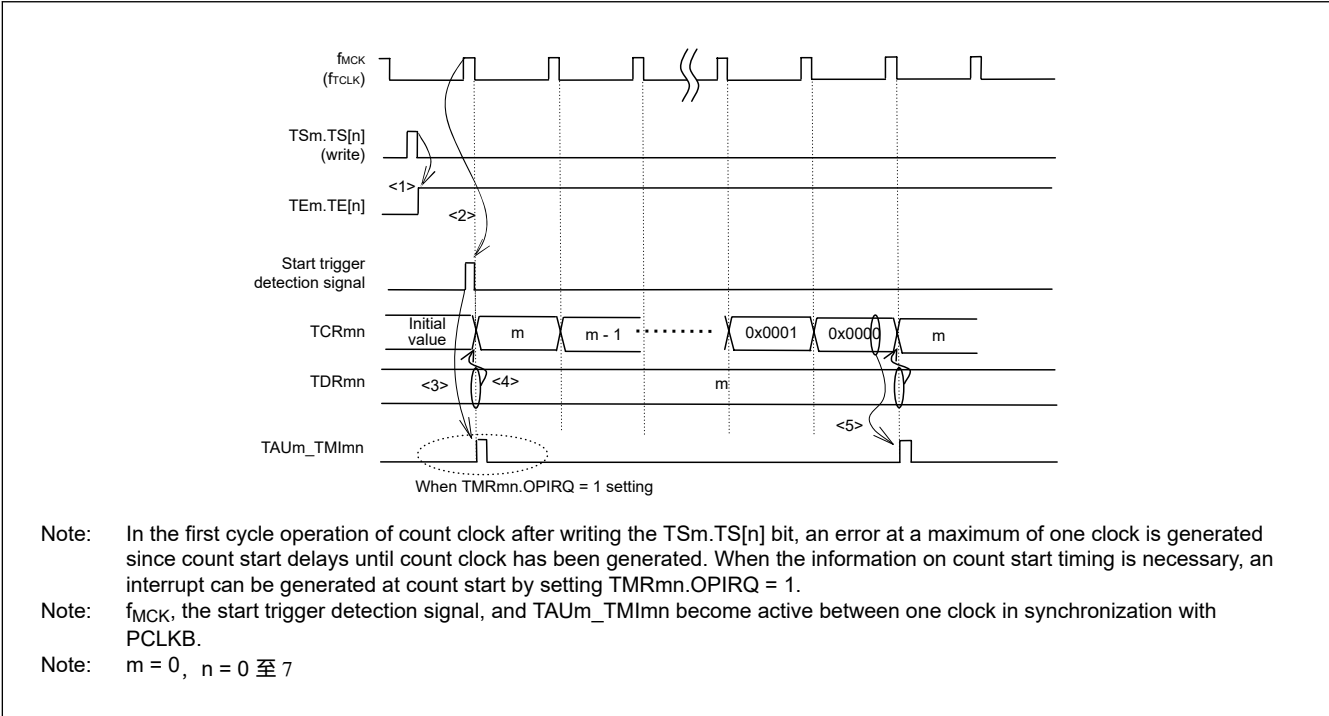


图 16.22 间隔定时器模式下的运行时间

(2) 事件计数器模式下的操作

- <1>定时器计数器寄存器0n(TCR0n) 在操作停止时保持其初始值(TE0.TE[n] = 0)。
- <2>操作通过向TS0.TS[n]位写入 1 来启用。
- <3>一旦 1 被写入TS0.TS[n]位，并且 1 被设置为TE0.TE[n] 位，定时器数据寄存器 0n (TDR0n) 的值就会被加载到 TCR0n 寄存器中以开始计数。
- <4>之后，根据 TI0n 输入有效边沿的计数时钟，TCR0n 寄存器值会递减。

Figure 16.23 shows the timing during operation in event counter mode.

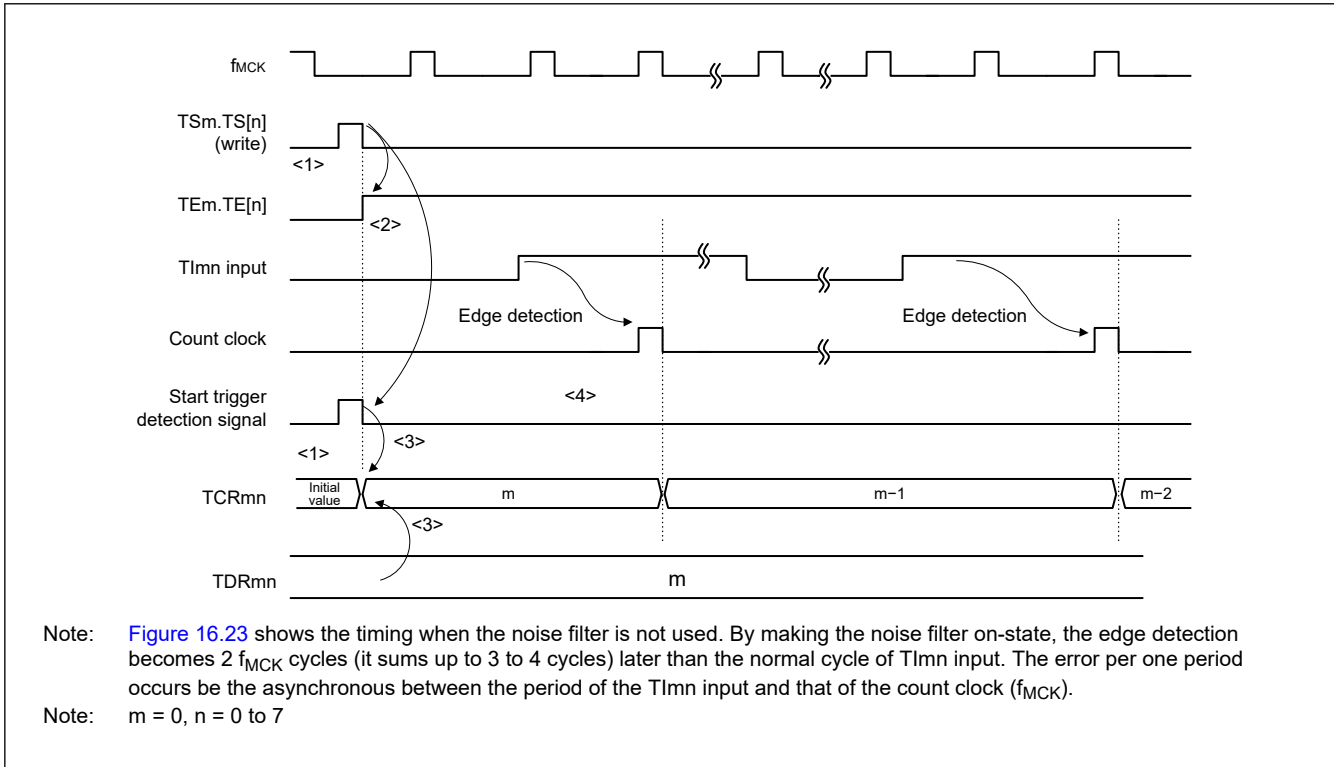


Figure 16.23 Timing during operation in event counter mode

(3) Operation in capture mode (input pulse interval measurement)

- <1> Operation is enabled ($TE0.TE[n] = 1$) by writing 1 to the $TS0.TS[n]$ bit.
- <2> Timer counter register $0n$ ($TCR0n$) holds the initial value until count clock generation.
- <3> A start trigger is generated at the first count clock after operation is enabled. And the value of $0x0000$ is loaded to the $TCR0n$ register and counting starts in the capture mode. (when the $TMR0n.OPIRQ$ bit is set to 1, $TAU0\_TMI0n$ is generated by the start trigger.)
- <4> On detection of the valid edge of the $TI0n$ input, the value of the $TCR0n$ register is captured to timer data register $0n$ ($TDR0n$) and $TAU0\_TMI0n$ is generated. However, this captured value is meaningless. The $TCR0n$ register keeps on counting from $0x0000$.
- <5> On next detection of the valid edge of the $TI0n$ input, the value of the $TCR0n$ register is captured to timer data register $0n$ ($TDR0n$) and $TAU0\_TMI0n$ is generated.

Figure 16.24 shows the timing during operation in capture mode (Input Pulse Interval Measurement).

图 16.23 显示了事件计数器模式下的运行时间。

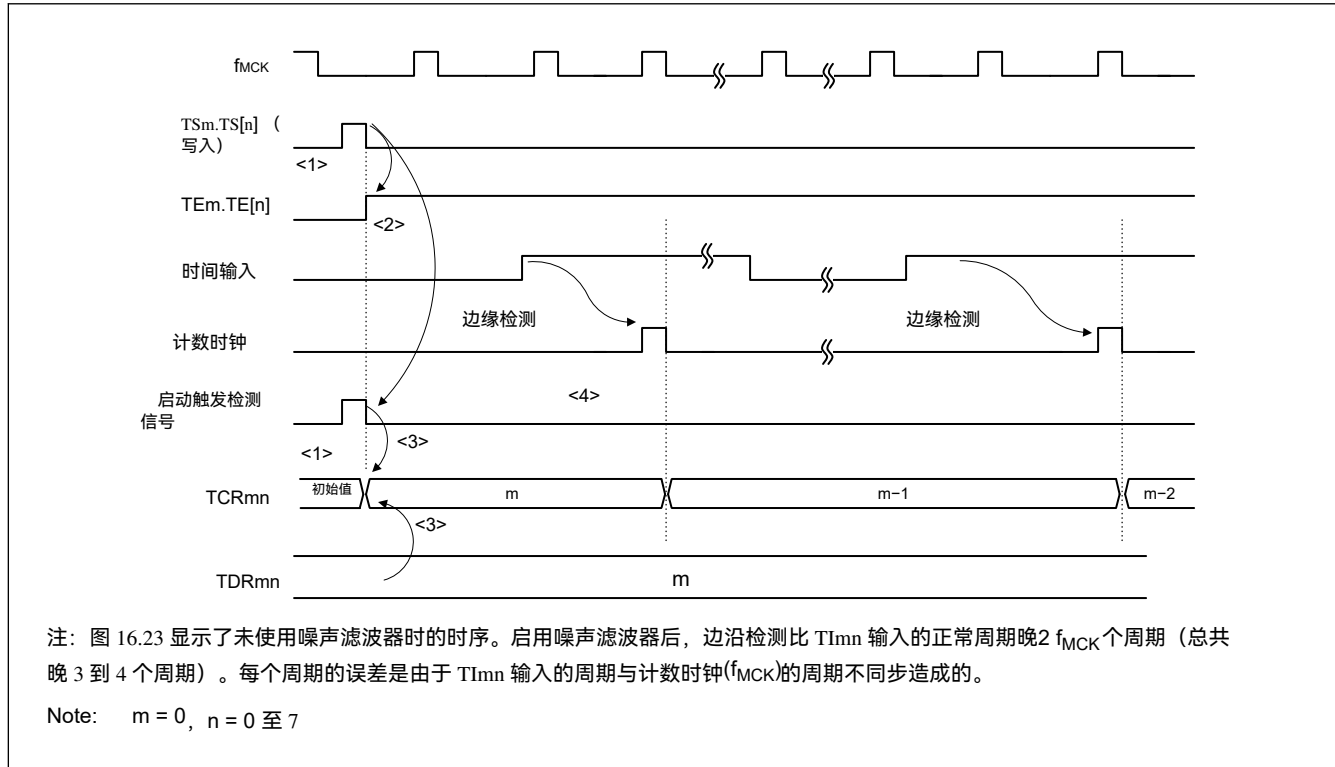


图 16.23 事件计数器模式下的运行时间

(3) 捕获模式下的操作（输入脉冲间隔测量）

- <1>操作通过向 $TS0.TS(TE0.TE[n] = 1)$ n 位写入 1 启用。
- <2>定时器计数器寄存器 $0n(TCR0n)$ 保存初始值，直到计数时钟生成。
- <3> A 启动触发信号在操作使能后的第一个计数时钟周期生成。 $0x0000$ 的值被加载到 $TCR0n$ 寄存器，并在捕获模式下开始计数。（当 $TMR0n.OPIRQ$ 位设置为 1 时，启动触发信号会生成 $TAU0\_TMI0n$ 。）
- <4>检测到 $TI0n$ 输入的有效边沿时， $TCR0n$ 寄存器的值被捕获到定时器数据寄存器 $0n(TDR0n)$ ，并生成 $TAU0\_TMI0n$ 。然而，这个捕获的值是无意义的。 $TCR0n$ 寄存器从 $0x0000$ 开始持续计数。
- <5>下次检测到 $TI0n$ 输入的有效边沿时，将 $TCR0n$ 寄存器的值捕获到定时器数据寄存器 $0n$ ($TDR0n$) 中，并生成 $TAU0\_TMI0n$ 。

图 16.24 显示了捕获模式下的操作时序（输入脉冲间隔测量）。

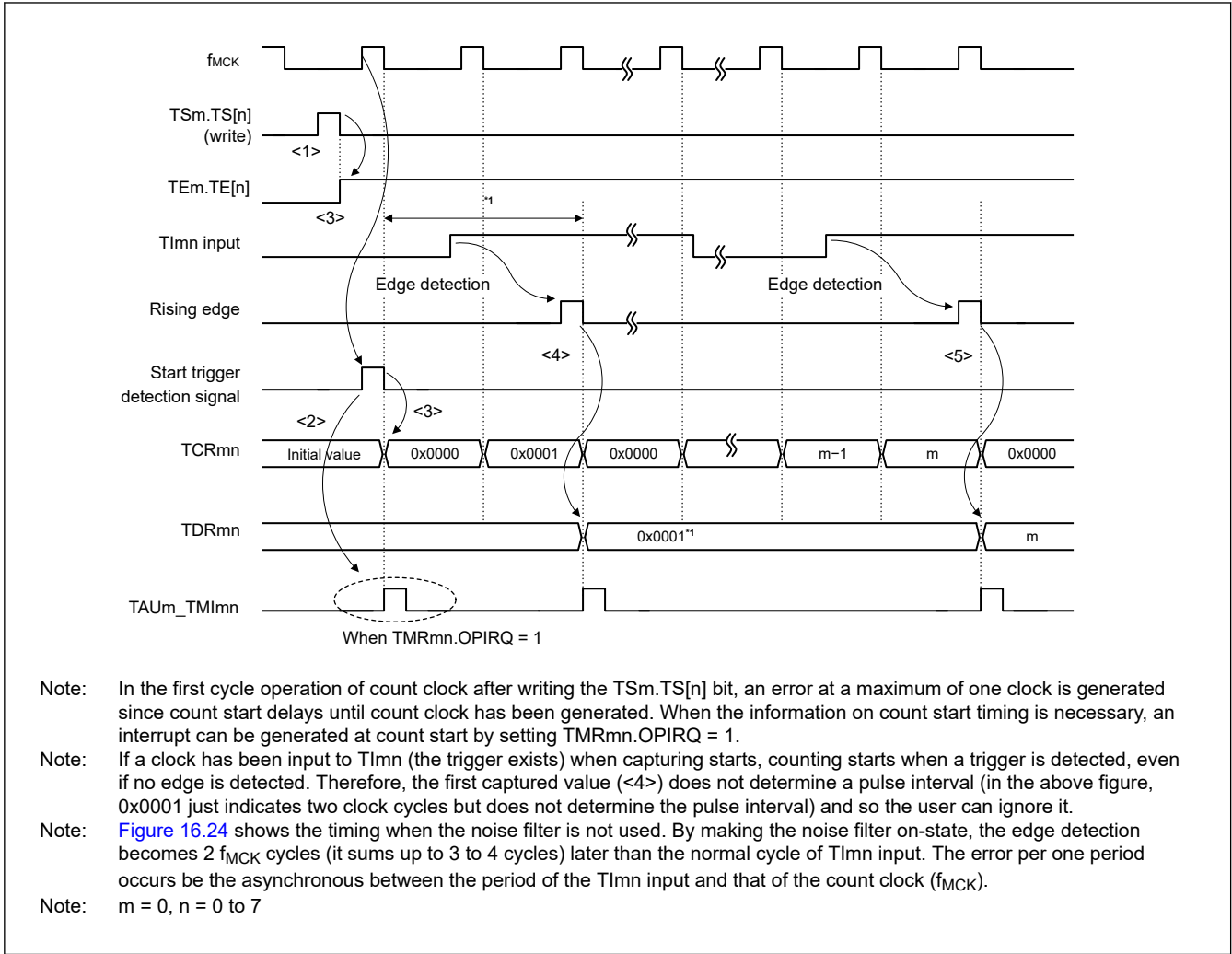


Figure 16.24 Timing during operation in capture mode (input pulse interval measurement)

(4) Operation in one-count mode

- <1> Operation is enabled (TE0.TE[n] = 1) by writing 1 to the TS0.TS[n] bit.
- <2> Timer counter register 0n (TCR0n) holds the initial value until start trigger generation.
- <3> Rising edge of the TI0n input is detected.
- <4> On start trigger detection, the value of timer data register 0n (TDR0n) is loaded to the TCR0n register and count starts.
- <5> When the TCR0n register counts down and its count value is 0x0000, TAU0_TMI0n is generated and the value of the TCR0n register becomes 0xFFFF and counting stops.

Figure 16.25 shows the timing during operation in one-count mode.

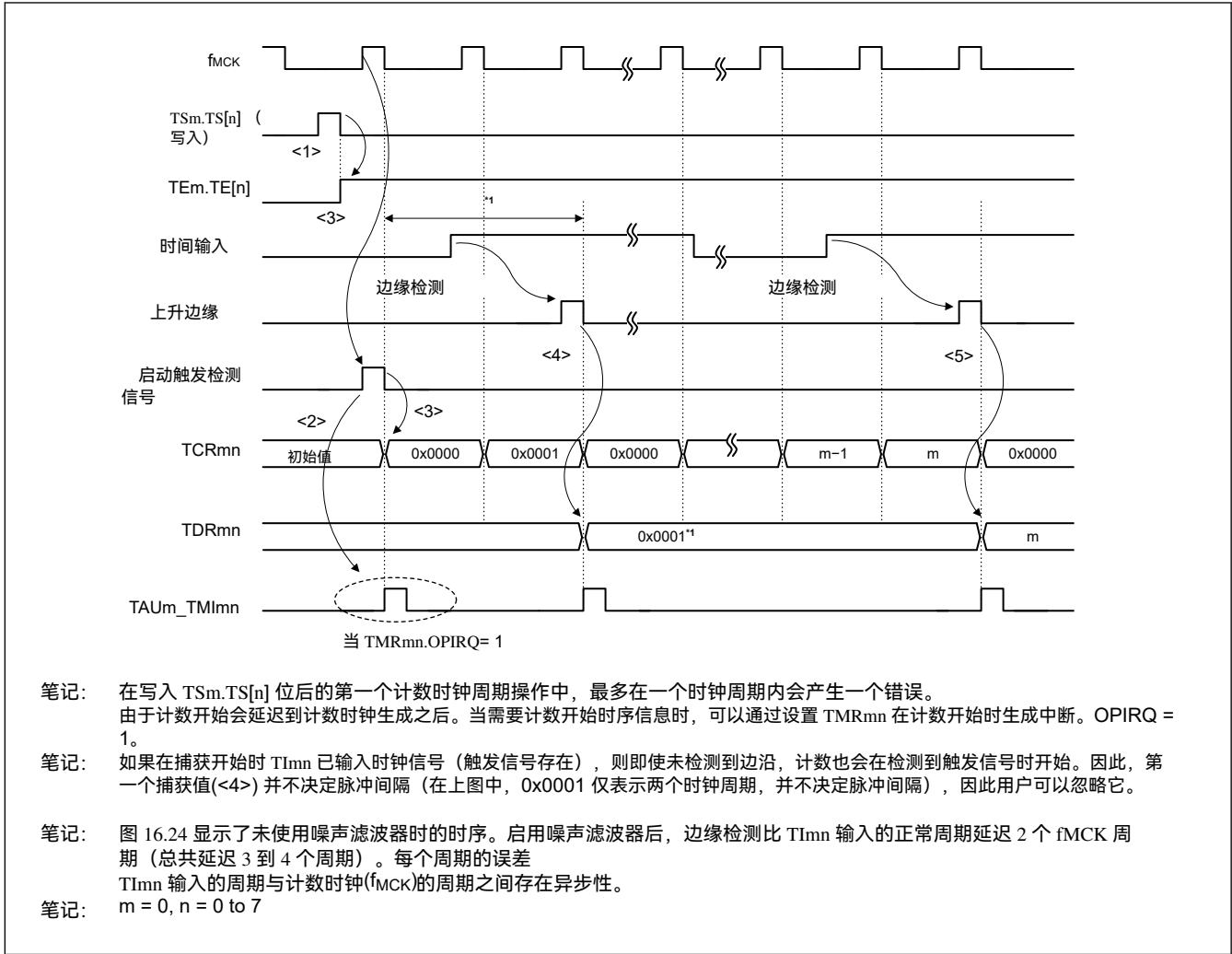


图 16.24 捕获模式下的操作时序 (输入脉冲间隔测量)

(4) 单次计数模式操作

- <1>操作通过向TS0.TS[n]位写入 1 来启用。
- <2>定时器计数器寄存器0n(TCR0n) 保存初始值, 直到开始触发生成。
- <3>检测到 TI0n 输入的上升沿。
- <4>检测到启动触发时, 定时器数据寄存器 0n (TDR0n) 的值被加载到 TCR0n 寄存器中, 计数开始。
- <5>当 TCR0n 寄存器递减计数且其计数值为0x0000, 时, 生成 TAU0_TMI0n, TCR0n 寄存器的值变为 0xFFFF, 计数停止。

图 16.25 显示了单计数模式下的操作时间。

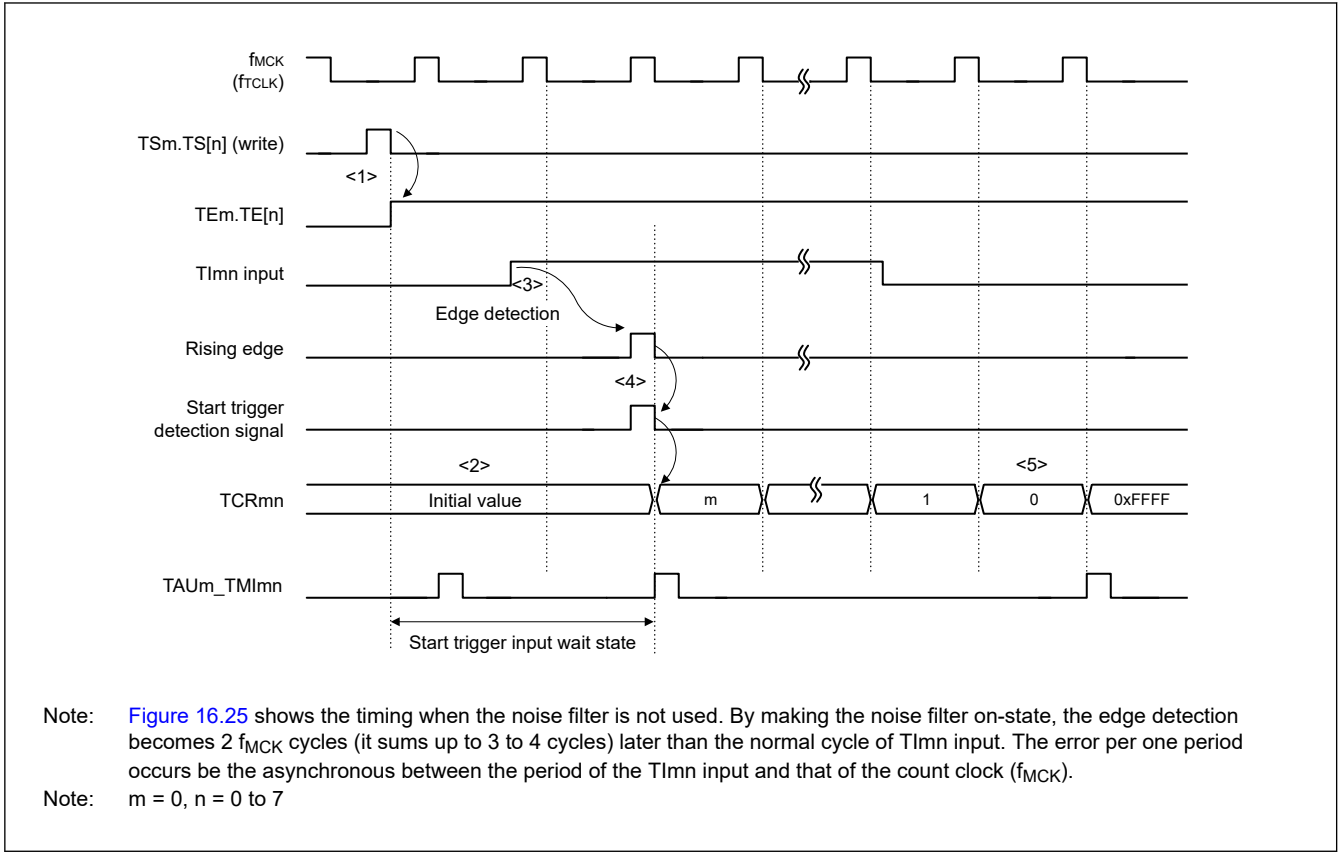


Figure 16.25 Timing during operation in one-count mode

(5) Operation in capture & one-count mode (high-level width measurement)

- <1> Operation is enabled ($TE0.TE[n] = 1$) by writing 1 to the $TS[n]$ bit of timer channel start register 0 ($TS0$).
- <2> Timer counter register 0n ($TCR0n$) holds the initial value until start trigger generation.
- <3> Rising edge of the $TI0n$ input is detected.
- <4> On start trigger detection, the value of 0x0000 is loaded to the $TCR0n$ register and count starts.
- <5> On detection of the falling edge of the $TI0n$ input, the value of the $TCR0n$ register is captured to timer data register 0n ($TDR0n$) and $TAU0\_TMI0n$ is generated.

Figure 16.26 shows the timing during operation in capture & one-count mode (High-Level Width Measurement).

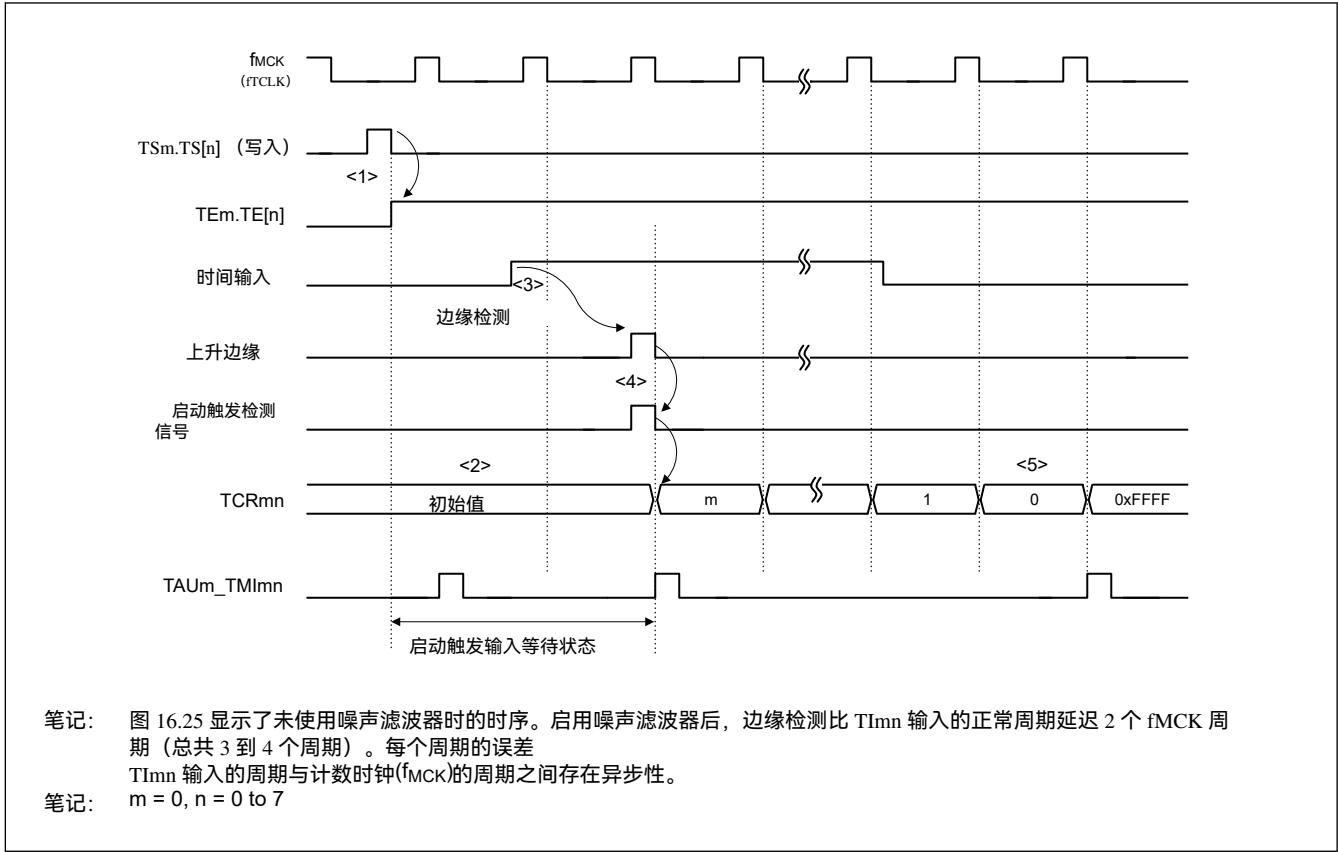


图 16.25 单计数模式下的操作时序

(5) 捕获操作&单计数模式 (高级宽度测量)

- <1>操作通过向定时器通道起始寄存器 0 ($TS0$) 的 $TS[n]$ 位写入 1 来启用。
- <2>定时器计数器寄存器0n($TCR0n$) 保存初始值, 直到触发生成开始。
- <3>检测到 $TI0n$ 输入的上升沿。
- <4>检测到启动触发时, 0x0000的值被加载到 $TCR0n$ 寄存器中, 计数开始。
- <5>当检测到 $TI0n$ 输入的下降沿时, $TCR0n$ 寄存器的值被捕获到定时器数据寄存器 0n ($TDR0n$) 中, 并生成 $TAU0\_TMI0n$ 。

图 16.26 显示了捕获&单计数模式 (高电平宽度测量) 操作期间的计时。

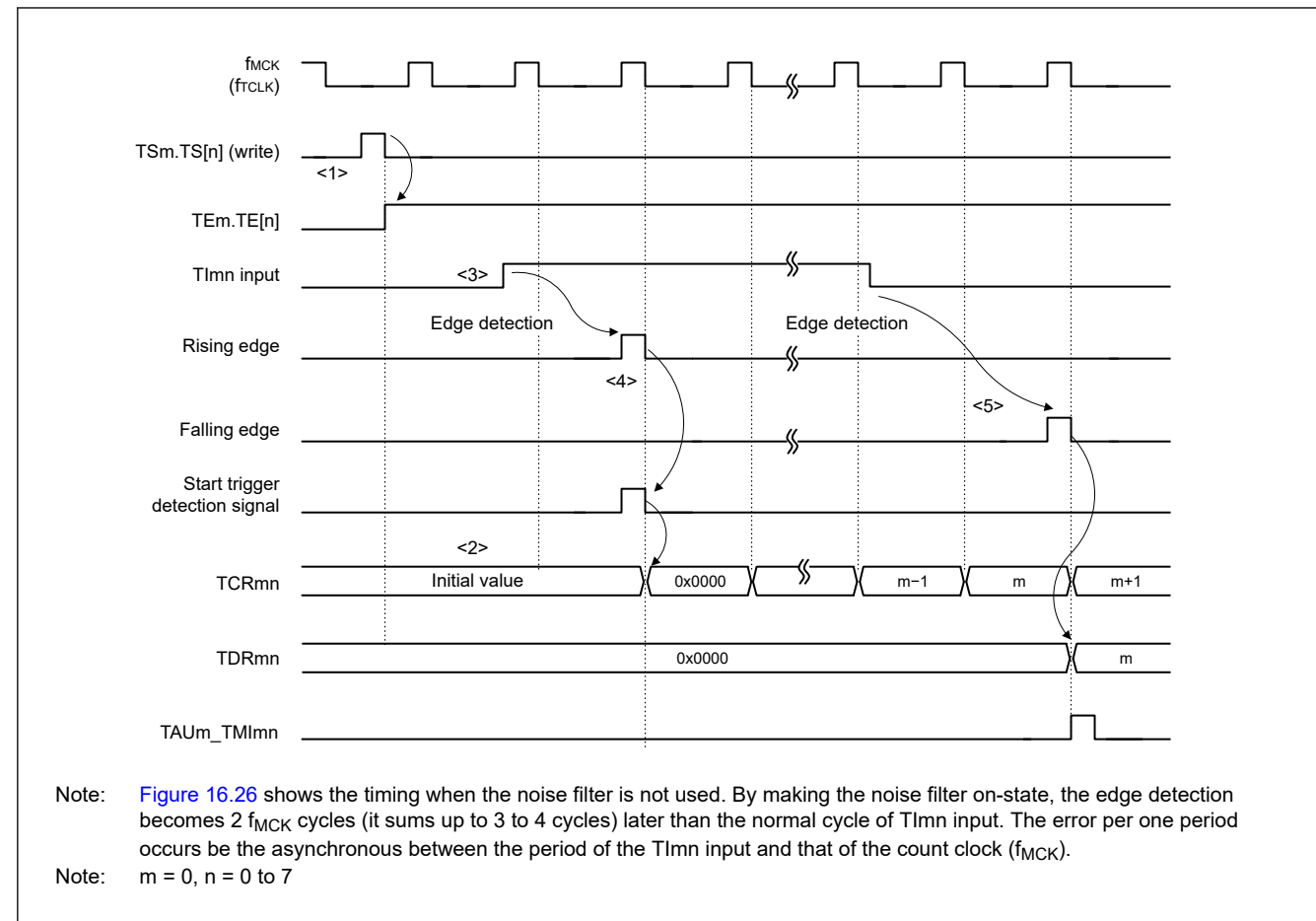


Figure 16.26 Timing during operation in capture & one-count mode (high-level width measurement)

16.5 Channel Output (TO0n Pin) Control

16.5.1 TO0n Pin Output Circuit Configuration

Figure 16.27 shows the TO0n pin output circuit.

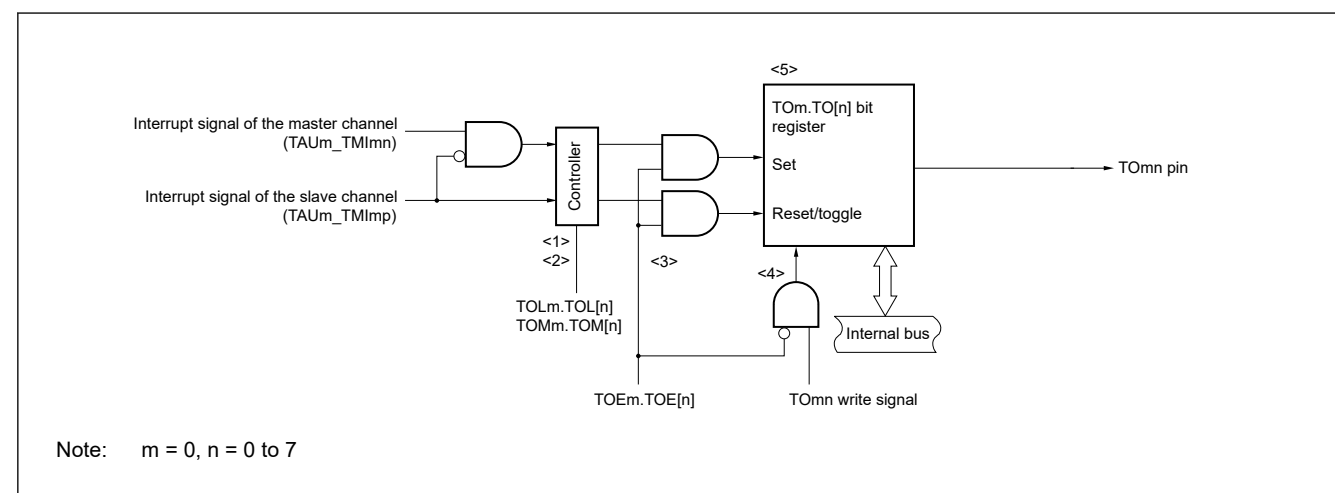


Figure 16.27 **Output circuit configuration**

The following describes the TO0n pin output circuit.

<1> When TOM0.TOM[n] = 0 (master channel output mode), the set value of timer output level register 0 (TOL0) is ignored and only TAU0 TMI0p (slave channel timer interrupt) is transmitted to timer output register 0 (TO0).

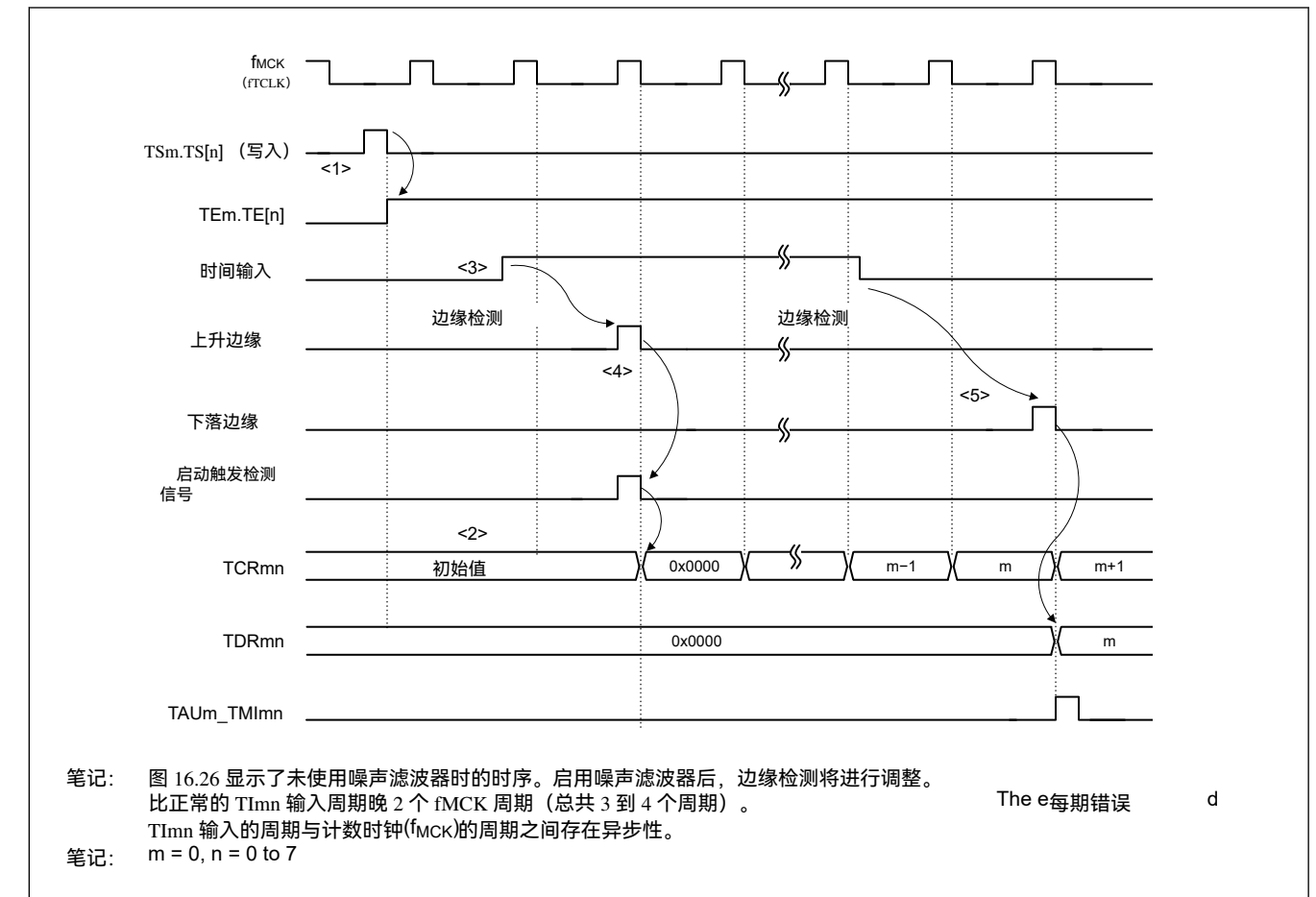


图 16.26 捕获和单次计数模式下的操作时序（高电平宽度测量）

16.5通道输出（TO0n引脚）控制

16.5.1T00n 引脚输出电路配置

图 16.27 显示了 TO0n 引脚输出电路。

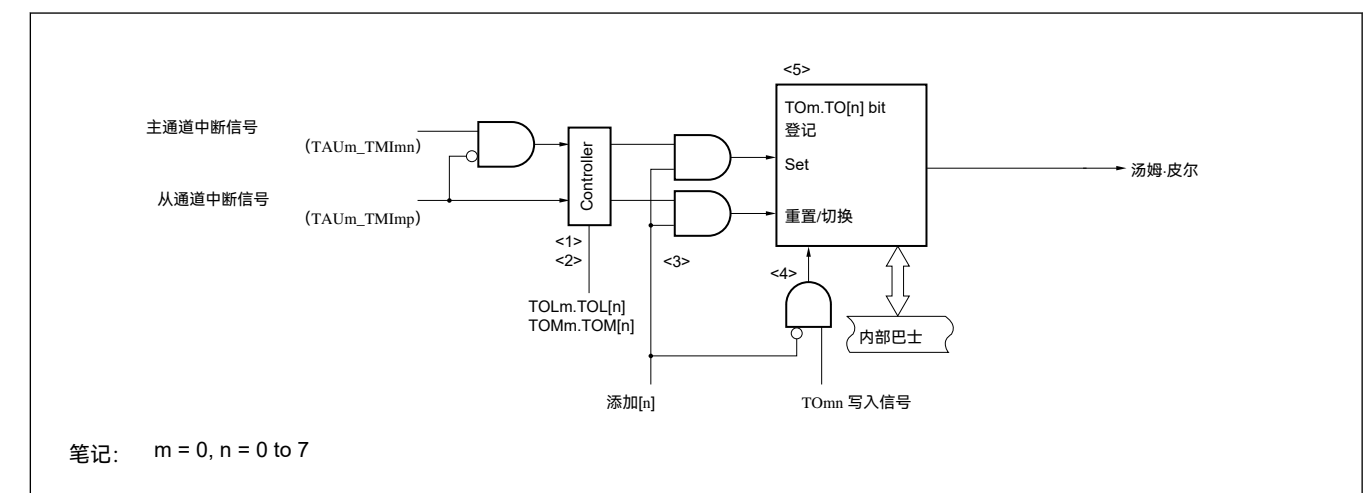


图 16.27 输出电路配置

下面描述 TO0n 引脚输出电路。

<1> When TOM0.TOM[n] = 0 (主通道输出模式)，定时器输出电平寄存器 0 (TOL0) 的设定值将被忽略，只有 TAU0.TMI0p (从通道定时器中断) 被发送到定时器输出寄存器 0 (TO0)。

<2> When $TOM0.TOM[n] = 1$ (slave channel output mode), both $TAU0\_TMI0n$ (master channel timer interrupt) and $TAU0\_TMI0p$ (slave channel timer interrupt) are transmitted to the $TO0$ register.

At this time, the $TOL0$ register becomes valid and the signals are controlled as follows:

When $TOL0.TOL[n] = 0$: Positive logic output ($TAU0\_TMI0n \rightarrow \text{set}$, $TAU0\_TMI0p \rightarrow \text{reset}$)

When $TOL0.TOL[n] = 1$: Negative logic output ($TAU0\_TMI0n \rightarrow \text{reset}$, $TAU0\_TMI0p \rightarrow \text{set}$)

When $TAU0\_TMI0n$ and $TAU0\_TMI0p$ are simultaneously generated, (0% output of PWM), $TAU0\_TMI0p$ (reset signal) takes priority, and $TAU0\_TMI0n$ (set signal) is masked.

<3> While timer output is enabled ($TOE0.TOE[n] = 1$), $TAU0\_TMI0n$ (master channel timer interrupt) and $TAU0\_TMI0p$ (slave channel timer interrupt) are transmitted to the $TO0$ register. Writing to the $TO0$ register ($TO0.TO[n]$ write signal) becomes invalid.

When $TOE0.TOE[n] = 1$, the $TO0n$ pin output never changes with signals other than interrupt signals.

To initialize the $TO0n$ pin output level, it is necessary to set timer operation is stopped ($TOE0.TOE[n] = 0$) and to write a value to the $TO0$ register.

<4> While timer output is disabled ($TOE0.TOE[n] = 0$), writing to the $TO0.TO[n]$ bit to the target channel ($TO0.TO[n]$ write signal) becomes valid. When timer output is disabled ($TOE0.TOE[n] = 0$), neither $TAU0\_TMI0n$ (master channel timer interrupt) nor $TAU0\_TMI0p$ (slave channel timer interrupt) is transmitted to the $TO0$ register.

<5> The $TO0$ register can always be read, and the $TO0n$ pin output level can be checked.

Note: n: Channel number
 $n = 0$ to 7 ($n = 0, 2, 4, 6$ for master channel)
 p: Slave channel number
 $n < p \leq 7$

16.5.2 $TO0n$ Pin Output Setting

Figure 16.28 shows the procedure and state transitions from the initial settings of a $TO0n$ output pin to the start of timer operation.

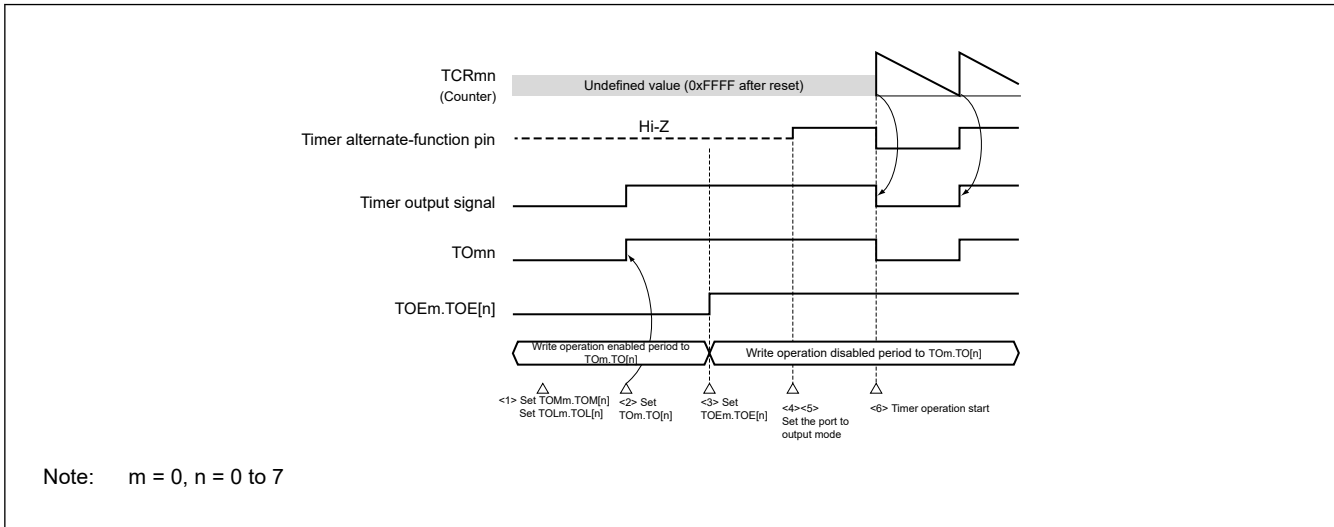


Figure 16.28 State transitions from the settings for timer output to the start of timer operation

<1> The operation mode of timer output is set.

- $TOM0.TOM[n]$ bit (0: Master channel output mode, 1: Slave channel output mode)
- $TOL0.TOL[n]$ bit (0: Positive logic output, 1: Negative logic output)

<2> The timer output signal is set to the initial state by setting timer output register 0 ($TO0$).

<3> The timer output operation is enabled by writing 1 to the $TOE0.TOE[n]$ bit (writing to the $TO0$ register is disabled).

当<2>为 $TOM0.TOM[n] = 1$ （从通道输出模式）时， $TAU0\_TMI0n$ （主通道定时器中断）和 $TAU0\_TMI0p$ （从通道定时器中断）都会发送到 $TO0$ 寄存器。

此时， $TOL0$ 寄存器生效，信号控制方式如下：

当 $TOL0.TOL[n] = 0$: 为正逻辑输出（ $TAU0\_TMI0n \rightarrow \text{set}$, $TAU0\_TMI0p \rightarrow \text{reset}$ ）

当 $TOL0.TOL[n] = 1$: 负逻辑输出($TAU0\_TMI0n \rightarrow \text{复位时}$, $TAU0\_TMI0p \rightarrow \text{set}$)

当 $TAU0\_TMI0n$ 和 $TAU0\_TMI0p$ 同时生成时，(0%为 PWM 输出， $TAU0\_TMI0p$ (复位信号) 优先，而 $TAU0\_TMI0n$ (置位信号) 被屏蔽。

<3>当定时器输出使能时($TOE0.TOE[n] = 1$)， $TAU0\_TMI0n$ （主通道定时器中断）和 $TAU0\_TMI0p$ （从通道定时器中断）被发送到 $TO0$ 寄存器。写入 $TO0$ 寄存器（($TO0.TO[n]$ 写入信号）将失效。

当 $TOE0.TOE[n] = 1$ 时，除中断信号外， $TO0n$ 引脚输出永远不会因其他信号而改变。

要初始化 $TO0n$ 引脚输出电平，需要设置定时器操作停止($TOE0.TOE[n] = 0$) 并向 $TO0$ 寄存器写入一个值。

当定时器输出被禁用时，写入目标通道的位（即写入信号）将变为有效。当定时器输出被禁用时，主通道定时器中断信号（ $TAU0\_TMI0n$ ）和从通道定时器中断信号（ $TAU0\_TMI0p$ ）都不会发送到 $TO0$ 寄存器。

<5> $TO0$ 寄存器始终可以读取，并且可以检查 $TO0n$ 引脚输出电平。

注: n: 频道号
 $n = 0$ 至 7 ($n = 0, 2, 4, 6$ （主通道）
 p: 从属通道编号
 $n < p \leq 7$

16.5.2 $TO0n$ 引脚输出设置

图 16.28 显示了从 $TO0n$ 输出引脚的初始设置到定时器开始运行的过程和状态转换。

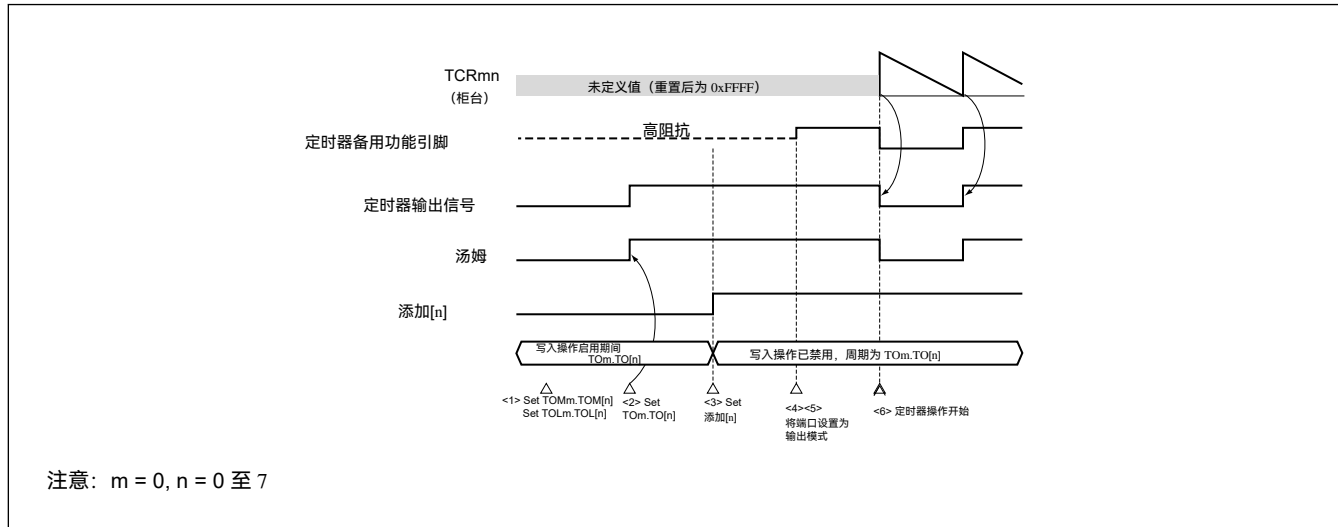


图 16.28 定时器输出设置到定时器开始运行的状态转换

<1>定时器输出的操作模式已设置。

- $TOM0.TOM[n]$ 位（0: 主通道输出模式，1: 从通道输出模式）
- $TOL0.TOL[n]$ 位（0: 正逻辑输出，1: 负逻辑输出）

<2>将定时器输出寄存器 0 ($TO0$) 设置为定时器输出信号初始状态。

<3>通过向 $TOE0.TOE[n]$ 位写入 1 来启用定时器输出操作（禁止写入 $TO0$ 寄存器）。

<4> The port is set to Peripheral output by the PSEL[2:0] bits of Port mn Pin Function Select Register (PmnPFS_A) (see section 15.2.6. PmnPFS_A : Port mn Pin Function Select Register (m = 1 to 3, n = 00 to 15), section 15.2.7. P0nPFS_A : Port 0n Pin Function Select Register (n = 10 to 13)).

<5> The port I/O setting is set to output by the PDR bit of PmnPFS_A register (see section 16.2.17. Registers Controlling Port Functions of Pins to be Used for Timer I/O).

<6> The timer operation is enabled (TS0.TS[n] = 1).

Note: n: Channel number (n = 0 to 7)

16.5.3 Cautions on Channel Output Operation

(1) Changing values set in the registers TO0, TOE0, and TOL0 during timer operation

Since the timer operations (operations of timer counter register 0n (TCR0n) and timer data register 0n (TDR0n)) are independent of the TO0n output circuit and changing the values set in timer output register 0 (TO0), timer output enable register 0 (TOE0), and timer output level register 0 (TOL0) does not affect the timer operation, the values can be changed during timer operation. To output an expected waveform from the TO0n pin by timer operation, however, set the TO0, TOE0, TOL0, and TOM0 registers to the values stated in the register setting example of each operation shown in section 16.6. Timer Input (TI0n) Control and section 16.7. Independent Channel Operation Function of Timer Array Unit.

When the values set to the TOE0, and TOM0 registers (but not the TO0 register) are changed close to the occurrence of the timer interrupt (TAU0_TMI0n) of each channel, the waveform output to the TO0n pin might differ, depending on whether the values are changed immediately before or immediately after the timer interrupt (TAU0_TMI0n) occurs.

Note: n: Channel number (n = 0 to 7)

(2) Default level of TO0n pin and output level after timer operation start

The change in the output level of the TO0n pin when timer output register 0 (TO0) is written while timer output is disabled (TOE0.TOE[n] = 0), the initial level is changed, and then timer output is enabled (TOE0.TOE[n] = 1) before port output is enabled, is shown below.

a. When operation starts with master channel output mode (TOM0.TOM[n] = 0) setting

The setting of timer output level register 0 (TOL0) is invalid when master channel output mode (TOM0.TOM[n] = 0). When the timer operation starts after setting the default level, the toggle signal is generated and the output level of the TO0n pin is inverted.

Figure 16.29 shows the output state of the TO0n pin with the output toggled (TOM0.TOM[n] = 0).

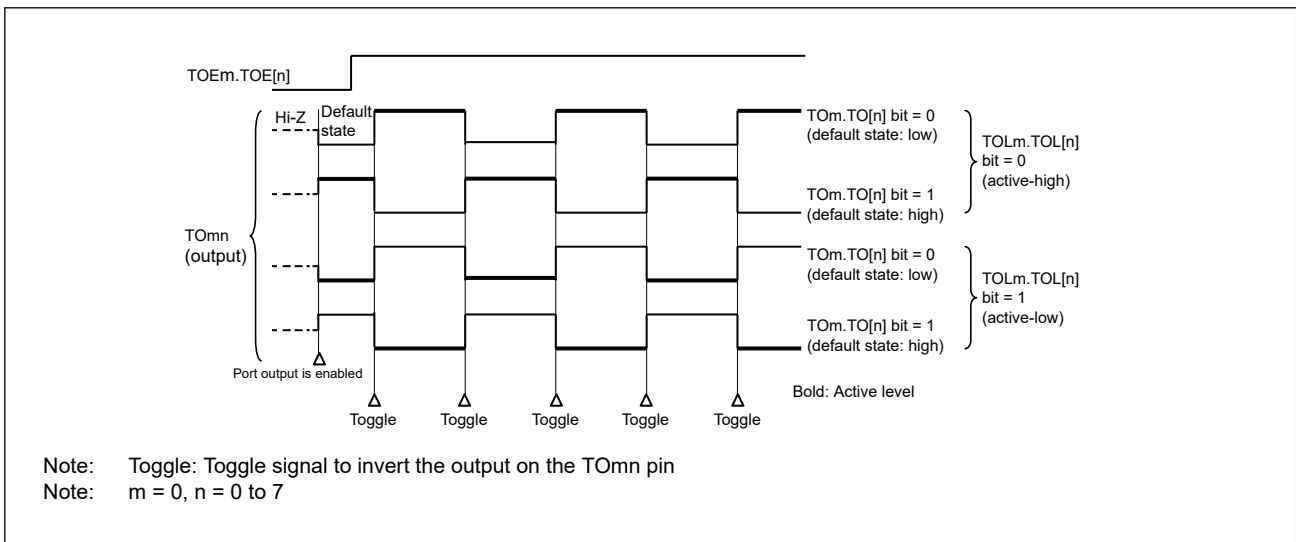


Figure 16.29 T0mn pin output states with toggled output (TOM0.TOM[n] = 0)

b. When operation starts with slave channel output mode (TOM0.TOM[p] = 1) setting (PWM output)

When slave channel output mode (TOM0.TOM[p] = 1), the active level is determined by timer output level register 0 (TOL0) setting.

Figure 16.30 shows the output state of the TO0p pin with PWM output (TOM0.TOM[p] = 1).

<4>端口由 PSEL 设置为外设输出[2:0] 端口 mn 引脚功能选择寄存器 (PmnPFS_A) 的位 (参见 15.2.6 节 PmnPFS_A:端口 mn 引脚功能选择寄存器(m = 1至 3, n = 00 至 15) , 15.2.7 节 P0nPFS_A : 端口0n引脚功能选择寄存器(n = 10至 13))。

<5>端口I/O设置由 PmnPFS_A 寄存器的 PDR 位设置为输出 (参见 16.2.17 节。用于定时器 I/O 的引脚端口功能的寄存器)。

<6>定时器操作已启用(TS0.TS[n] = 1)。

注: n: 频道号(n = 0至 7)

16.5.3 通道输出操作注意事项

(1) 在定时器操作期间更改寄存器 TO0、TOE0 和 TOL0 中设置的值

由于定时器操作 (定时器计数器寄存器 0n (TCR0n) 和定时器数据寄存器 0n (TDR0n) 的操作) 独立于 TO0n 输出电路, 并且更改定时器输出寄存器 0 (TO0)、定时器输出使能寄存器 0 (TOE0) 和定时器输出电平寄存器 0 (TOL0) 中设置的值不会影响定时器操作, 因此可以在定时器操作期间更改这些值。但是, 要通过定时器操作从 TO0n 引脚输出预期波形, 请将 TO0、TOE0、TOL0 和 TOM0 寄存器设置为第 16.6 节“定时器输入 (TI0n) 控制”和第 16.7 节“定时器阵列单元的独立通道操作功能”中所示的每个操作的寄存器设置示例中所述的值。

当在每个通道的定时器中断 (TAU0_TMI0n) 发生附近更改 TOE0 和 TOM0 寄存器 (但不更改 TO0 寄存器) 的值时, 输出到 TO0n 引脚的波形可能会有所不同, 具体取决于这些值是在定时器中断 (TAU0_TMI0n) 发生之前立即更改还是在发生之后立即更改。

注: n: 频道号(n = 0至 7)

(2) 定时器开始工作后 TO0n 引脚和输出电平的默认电平

下面显示了定时器输出寄存器 0 (TO0) 在定时器输出禁用(TOE0.TOE[n] = 0)时写入 TO0n 引脚时输出电平的变化, 初始电平发生变化, 然后在端口输出启用之前使能定时器输出(TOE0.TOE[n] = 1)。

a. 当操作以主通道输出模式(TOM0.TOM[n] = 0) 设置启动时

当主通道输出模式为(TOM0.TOM[n] = 0)时, 定时器输出电平寄存器0 (TOL0) 的设置无效。设置默认电平后, 定时器开始工作时, 会产生翻转信号, 并且TO0n引脚的输出电平会被反转。

图 16.29 显示了 TO0n 引脚的输出状态, 输出翻转(TOM0.TOM[n] = 0)。

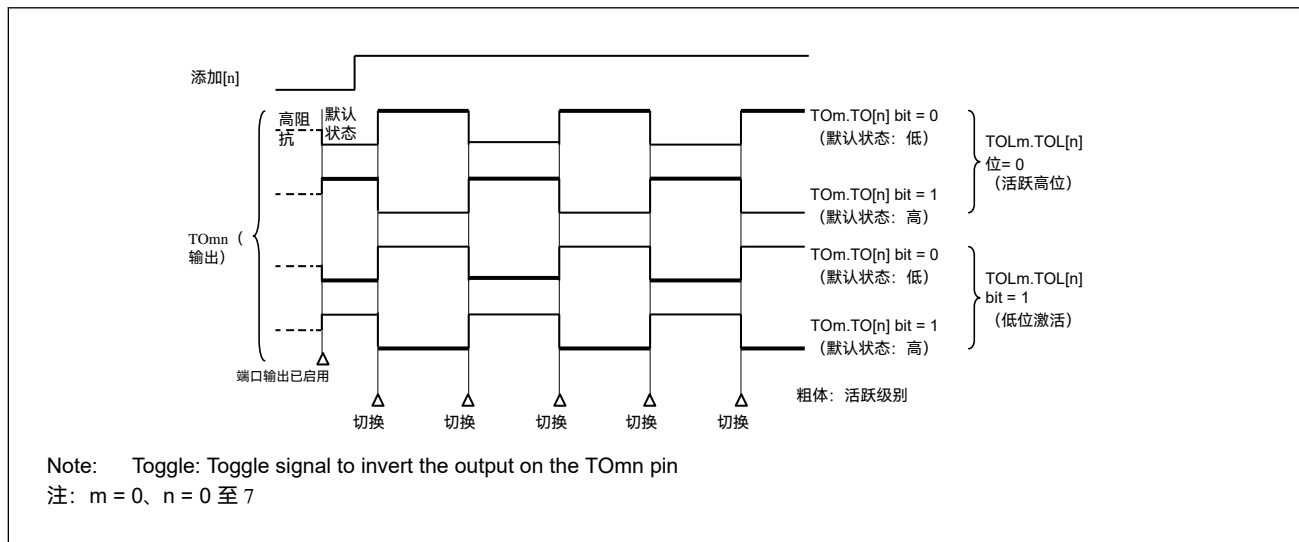


图 16.29 T0mm 引脚输出状态 (输出切换) (TOM0.TOM[n] = 0)

b. 当操作开始时, 从通道输出模式(TOM0.TOM[p] = 1) 设置 (PWM 输出)

当从通道输出模式为(TOM0.TOM[p] = 1)时, 有效电平由定时器输出电平寄存器0 (TOL0) 设置决定。

图 16.30 显示了TO0p引脚的输出状态, 其中(TOM0.TOM[p] = 1)为 PWM 输出。

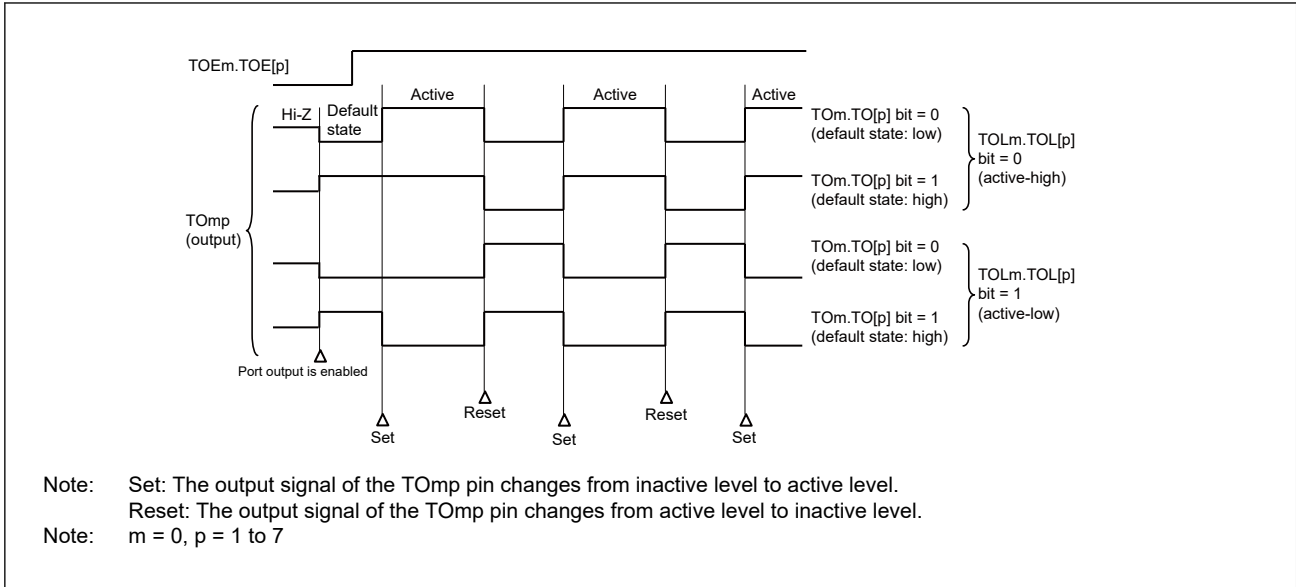


Figure 16.30 TO0p pin output states with PWM output (TOM0.TOM[p] = 1)

(3) Operation of TO0n pin in slave channel output mode (TOM0.TOM[n] = 1)

- a. When the relevant bit of timer output level register 0 (TOL0) is changed during timer operation
- When the TOL0 register setting has been changed during timer operation, the setting becomes valid at the generation timing of the TO0n pin change condition. Rewriting the TOL0 register does not change the output level of the TO0n pin. The operation when TOM0.TOM[n] is set to 1 and the value of the TOL0 register is changed while the timer is operating (TE0.TE[n] = 1) is shown Figure 16.31.

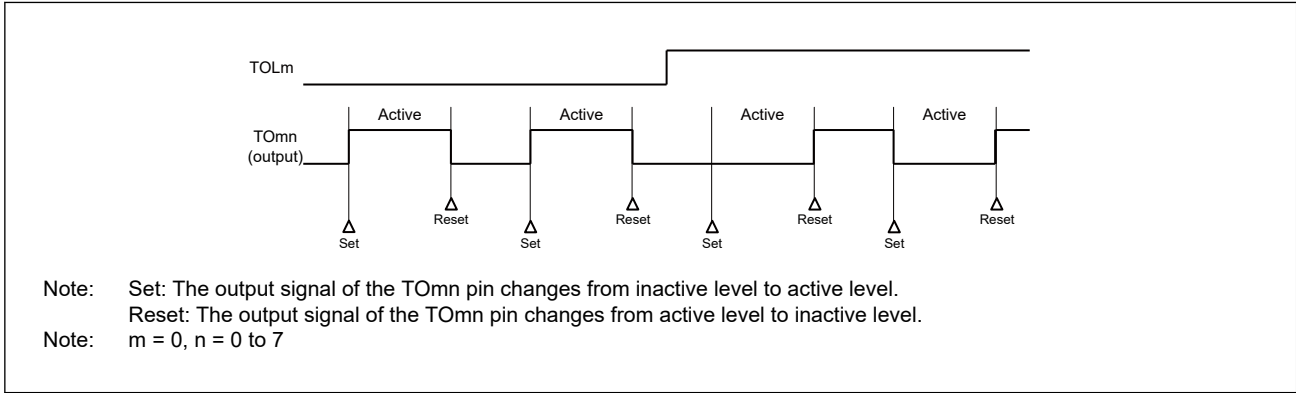


Figure 16.31 Operation when the relevant bit of the TOL0 register is changed during timer operation

- b. Set and reset timing
- To realize 0% and 100% output at PWM output, the TO0n pin and TO0.TO[n] bit set timing at master channel timer interrupt (TAU0_TMI0n) generation is delayed by 1 count clock by the slave channel.
- If the set condition and reset condition are generated at the same time, a higher priority is given to the latter.
- Figure 16.32 shows the states of operation following set and reset signals when the master and slave channels are set as follows.

Master channel: TOE0.TOE[n] = 1, TOM0.TOM[n] = 0, TOL0.TOL[n] = 0

Slave channel: TOE0.TOE[p] = 1, TOM0.TOM[p] = 1, TOL0.TOL[p] = 0

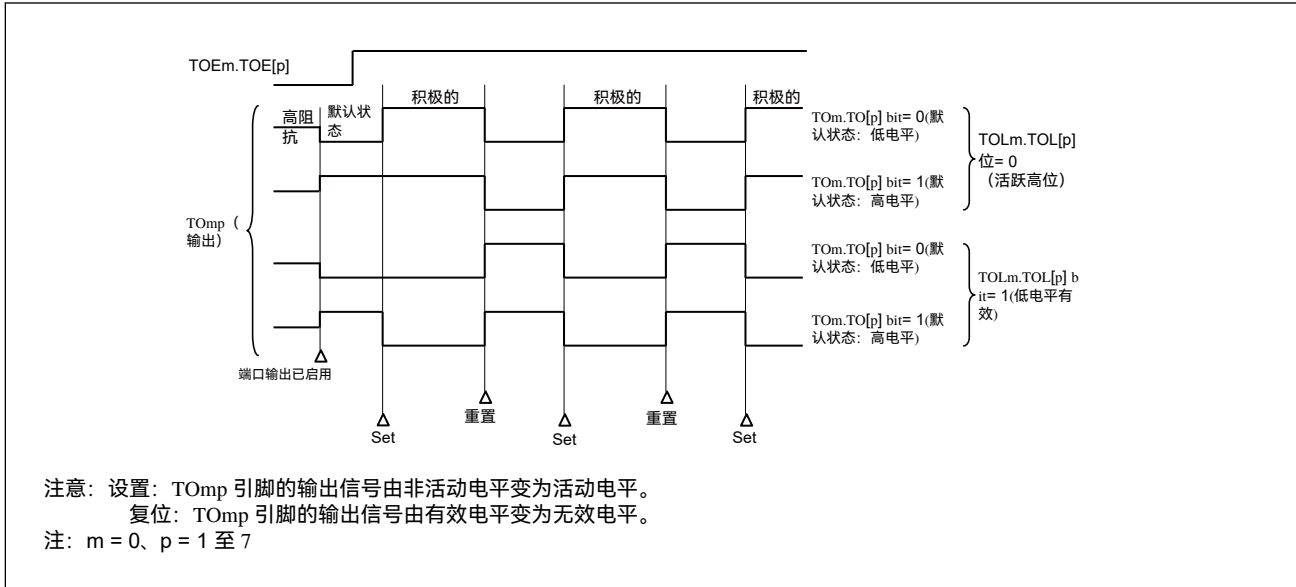


图 16.30 TO0p 引脚输出状态，带 PWM 输出(TOM0.TOM[p] = 1)

(3) TO0n 引脚在从通道输出模式下的操作(TOM0.TOM[n] = 1)

- a. 当定时器工作期间定时器输出电平寄存器0 (TOL0) 的相关位发生变化时，该变化会在TO0n 引脚变化条件生成时生效。重写TOL0寄存器不会改变TO0n 引脚的输出电平。

图 16.31 显示了当TOM0.TOM[n]设置为 1 且定时器(TE0.TE[n] = 1) 运行时 TOL0 寄存器的值发生变化时的操作。

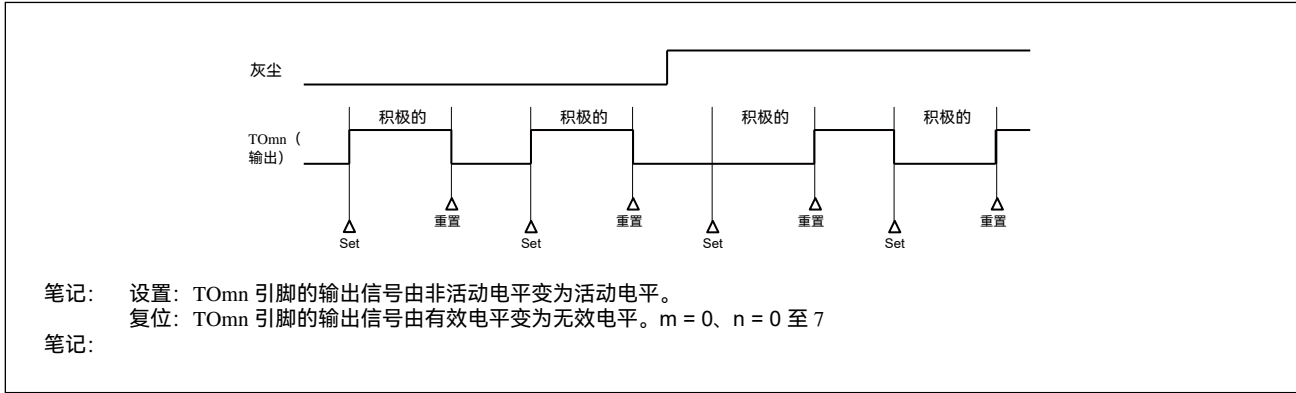


Figure 16.31 Operation when the relevant bit of the TOL0 register is changed during timer operation

- b. 设置和重置定时
- 为了在 PWM 输出端实现0%和100%输出，从通道将主通道定时器中断 (TAU0_TMI0n) 生成时 TO0n 引脚和TO0.TO[n] 位设置的时序延迟 1 个时钟周期。
- 如果设置条件和复位条件同时产生，则后者具有更高的优先级。
- 图 16.32 显示了当主通道和从通道设置如下时，在设置和复位信号之后的操作状态。

主通道: TOE0.TOE[n] = 1、TOM0.TOM[n] = 0、TOL0.TOL[n] = 0

从通道: TOE0.TOE[p] = 1, TOM0.TOM[p] = 1, TOL0.TOL[p] = 0

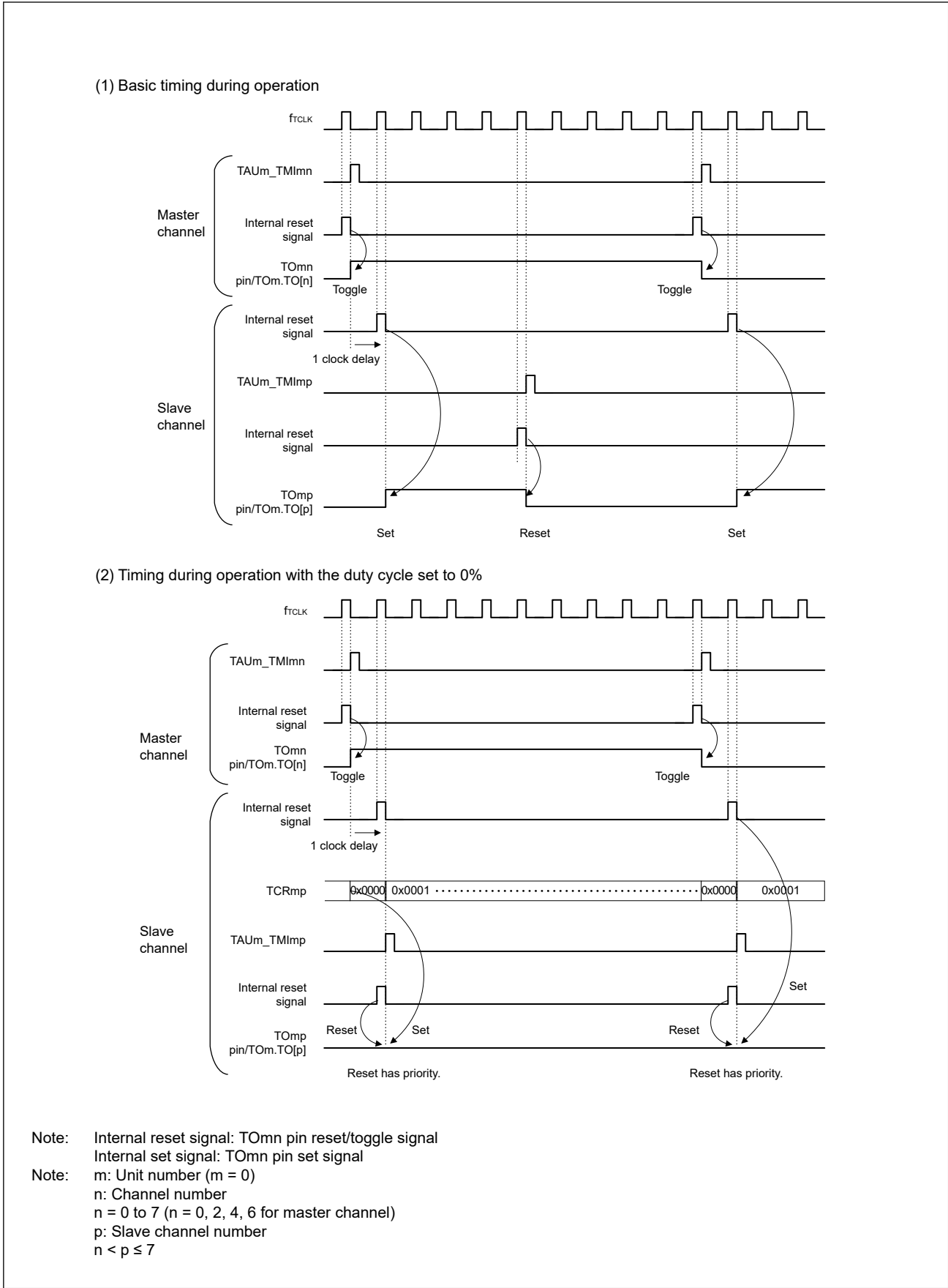


Figure 16.32 States of operation following set and reset signals

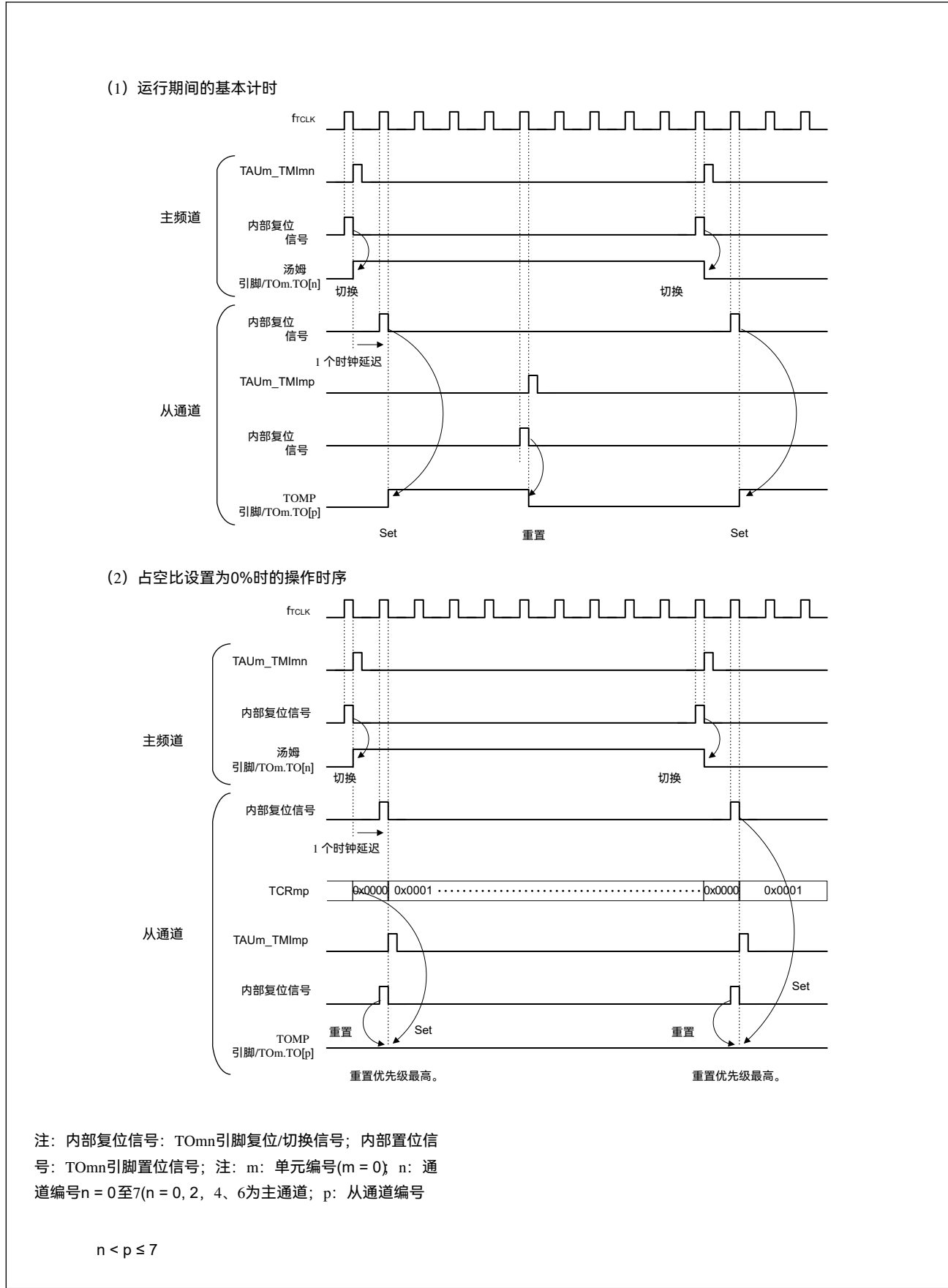


图 16.32 设置和复位信号后的操作状态

16.5.4 Collective Manipulation of TO0.TO[n] Bit

In timer output register 0 (TO0), the setting bits for all the channels are located in one register in the same way as timer channel start register 0 (TS0). Therefore, the TO0.TO[n] bit of all the channels can be manipulated collectively.

Only the desired bits can also be manipulated by enabling writing only to the TO0.TO[n] bit (TOE0.TOE[n] = 0) that correspond to the relevant bits of the channel used to perform output (TO0n).

Table 16.12 shows an example of a TO0n bit collective manipulation.

Table 16.12 Example of TO0.TO[n] bit collective manipulation

Bit position:		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Before writing	TO0	0	0	0	0	0	0	0	0	TO[7] 0	TO[6] 0	TO[5] 1	TO[4] 0	TO[3] 0	TO[2] 0	TO[1] 1	TO[0] 0
	TOE0	0	0	0	0	0	0	0	0	TOE[7] 0	TOE[6] 0	TOE[5] 1	TOE[4] 0	TOE[3] 1	TOE[2] 1	TOE[1] 1	TOE[0] 1
Data to be written	TO0	0	0	0	0	0	0	0	0	1 ↓	1 ↓	0 ↓ x	0 ↓	0 ↓ x	0 ↓ x	1 ↓ x	1 ↓ x
After writing	TO0	0	0	0	0	0	0	0	0	TO[7] 1	TO[6] 1	TO[5] 1	TO[4] 0	TO[3] 0	TO[2] 0	TO[1] 1	TO[0] 0

Writing is done only to the TO0.TO[n] bit with TOE0.TOE[n] = 0, and writing to the TO0.TO[n] bit with TOE0.TOE[n] = 1 is ignored.

TO0n (channel output) to which TOE0.TOE[n] = 1 is set is not affected by the write operation. Even if the write operation is done to the TO0.TO[n] bit, it is ignored and the output change by timer operation is normally done.

Figure 16.33 shows an example of a TO0.TO[n] bit collective manipulation.

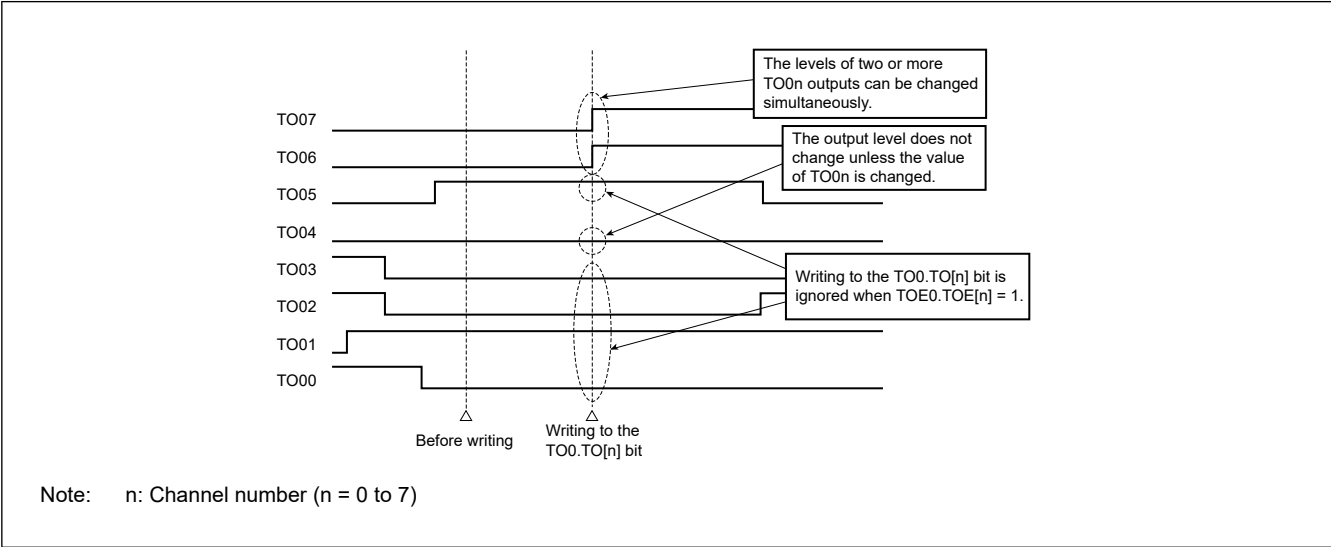


Figure 16.33 TO0n pin states by collective manipulation of TO0.TO[n] bit

16.5.5 Timer Interrupts and TO0n Outputs When Counting is Started

In the interval timer mode or capture mode, the TMR0n.OPIRQ bit in timer mode register 0n (TMR0n) sets whether to generate a timer interrupt at count start.

When the TMR0n.OPIRQ bit is set to 1, the count operation start timing can be known by the timer interrupt (TAU0_TMI0n) generation. In the other modes, neither timer interrupt at count operation start nor TO0n output is controlled.

Figure 16.34 shows operation examples when the interval timer mode (TOE0.TOE[n] = 1, TOM0.TOM[n] = 0) is set.

16.5.4 对 TO0.TO[n] 位进行集体操作

在定时器输出寄存器 0 (TO0) 中，所有通道的设置位都位于同一个寄存器中，与定时器通道起始寄存器 0 (TS0) 相同。因此，可以对所有通道的 TO0.TO[n] 位进行统一操作。

也可以通过仅写入与用于执行输出 (TO0n) 的通道的相关位对应的 TO0.TO[n] 位 (TOE0.TOE[n] = 0) 来操作所需的位。

表 16.12 显示了 TO0n 位集体操作的示例。

表 16.12 TO0.TO[n] 位集体操纵示例

位位置:		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
写	TO0	0	0	0	0	0	0	0	0	TO[7] 0	TO[6] 0	TO[5] 1	TO[4] 0	TO[3] 0	TO[2] 0	TO[1] 1	TO[0] 0
	TOE0	0	0	0	0	0	0	0	0	TOE[7] 0	TOE[6] 0	TOE[5] 1	TOE[4] 0	TOE[3] 1	TOE[2] 1	TOE[1] 1	TOE[0] 1
待写入的数据	TO0	0	0	0	0	0	0	0	0	1 ↓	1 ↓	0 ↓ x	0 ↓	0 ↓ x	0 ↓ x	1 ↓ x	1 ↓ x
写完之后	TO0	0	0	0	0	0	0	0	0	TO[7] 1	TO[6] 1	TO[5] 1	TO[4] 0	TO[3] 0	TO[2] 0	TO[1] 1	TO[0] 0

仅对 TO0.TO[n] 位进行写入，使用 OE0.TOE[n] = 0 进行写入，而对 TO0.TO[n] 位进行写入（使用 TOE0.TOE[n] = 1）将被忽略。

置位为 TOE0.TOE[n] = 1 的 TO0n（通道输出）不受写入操作的影响。即使对 TO0.TO[n] 位执行写入操作，也会被忽略，定时器操作引起的输出变化仍会正常执行。

图 16.33 显示了 TO0.TO[n] 位集体操作的示例。

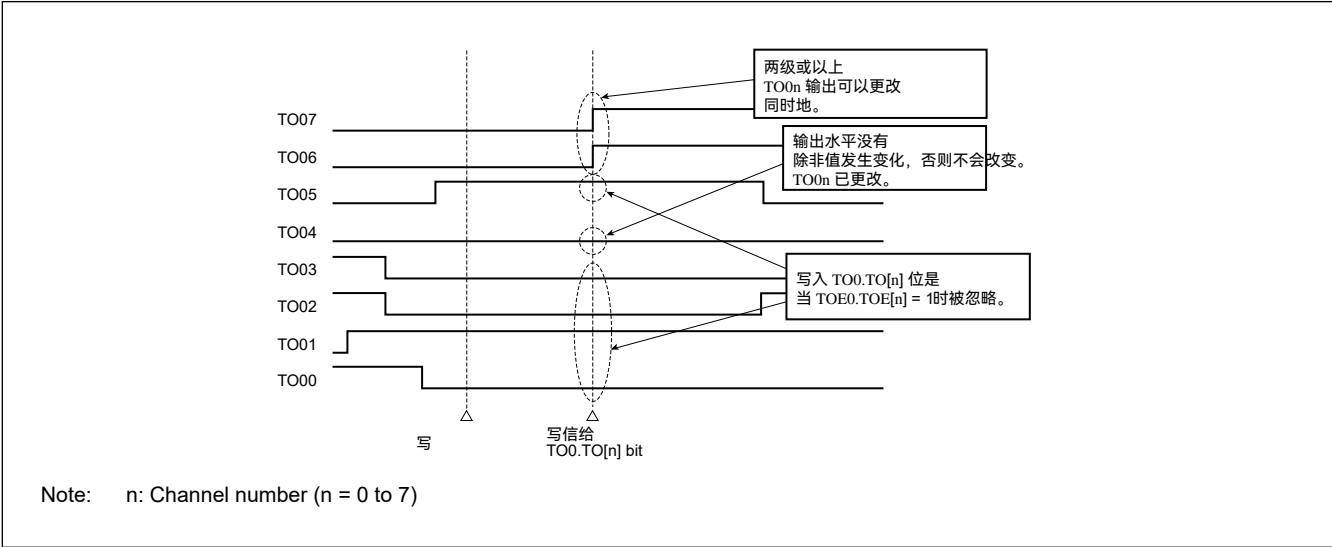


图 16.33 通过对 TO0.TO[n] 位进行集体操作，TO0n 引脚状态

16.5.5 定时器中断和计数开始时的 TO0n 输出

在间隔定时器模式或捕获模式下，定时器模式寄存器 0n (TMR0n) 中的 TMR0n.OPIRQ 位设置是否在计数开始时生成定时器中断。

当 TMR0n.OPIRQ 位设置为 1 时，可以通过定时器中断 (TAU0_TMI0n) 的生成来确定计数操作的开始时间。在其他模式下，计数操作开始时的定时器中断和 TO0n 输出均不受控制。

图 16.34 显示了设置间隔定时器模式 (TOE0.TOE[n] = 1、TOM0.TOM[n] = 0) 时的操作示例。

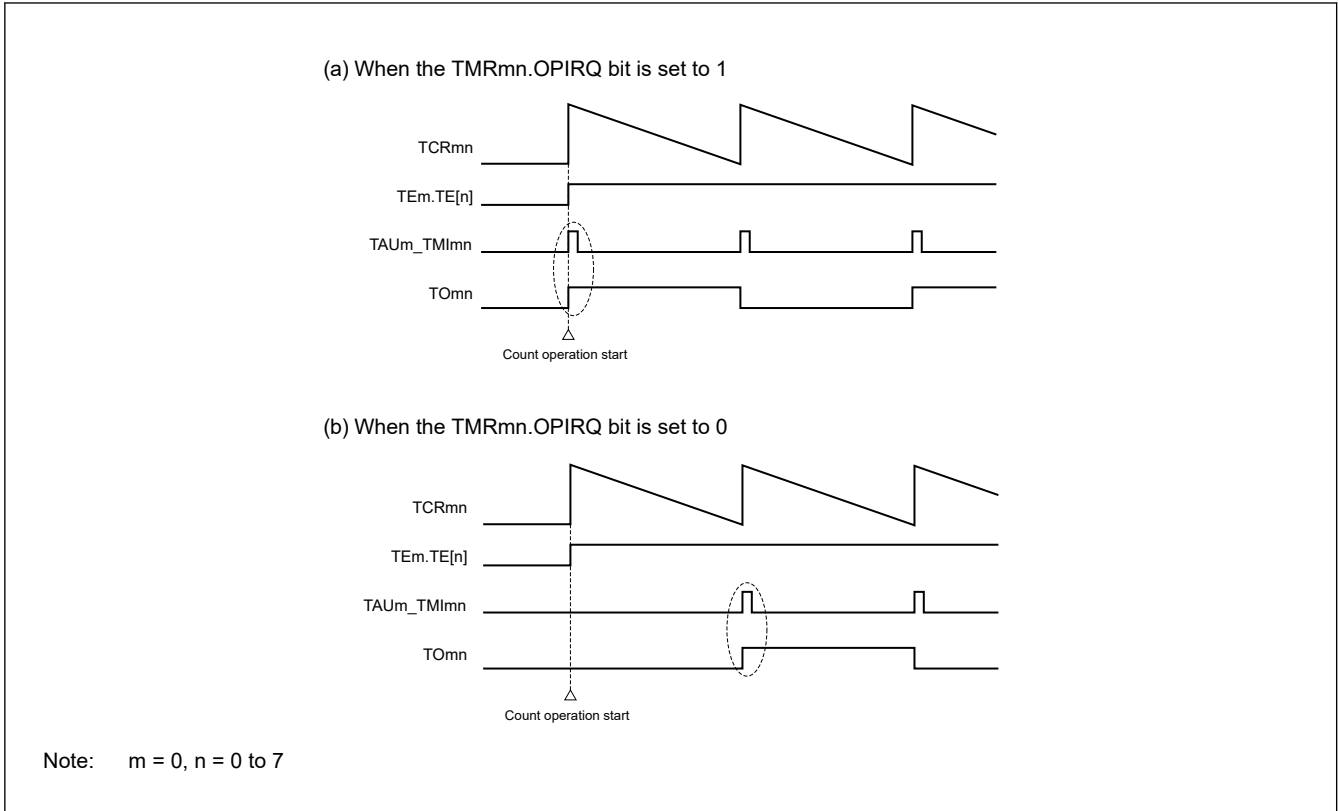


Figure 16.34 Examples of the operation of timer interrupts and TOMn outputs when counting is started

When the TMR0n.OPIRQ bit is set to 1, a timer interrupt (TAU0_TMI0n) is output at count operation start, and TO0n performs a toggle operation.

When the TMR0n.OPIRQ bit is set to 0, a timer interrupt (TAU0_TMI0n) is not output at count operation start, and TO0n does not change either. After counting one cycle, AUA0_TMI0n is output and TO0n performs a toggle operation.

16.6 Timer Input (TI0n) Control

16.6.1 TI0n Input Circuit Configuration

A signal is input from a timer input pin, goes through a noise filter and an edge detector, and is sent to a timer controller. Enable the noise filter for the pin in need of noise removal. Figure 16.35 shows the configuration of the input circuit.

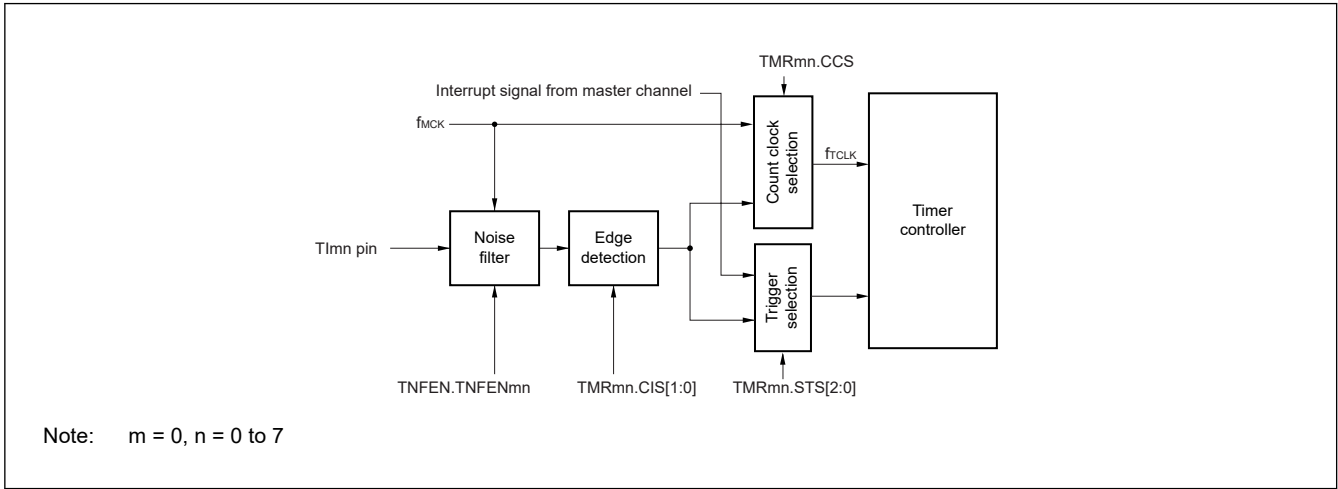


Figure 16.35 Input circuit configuration

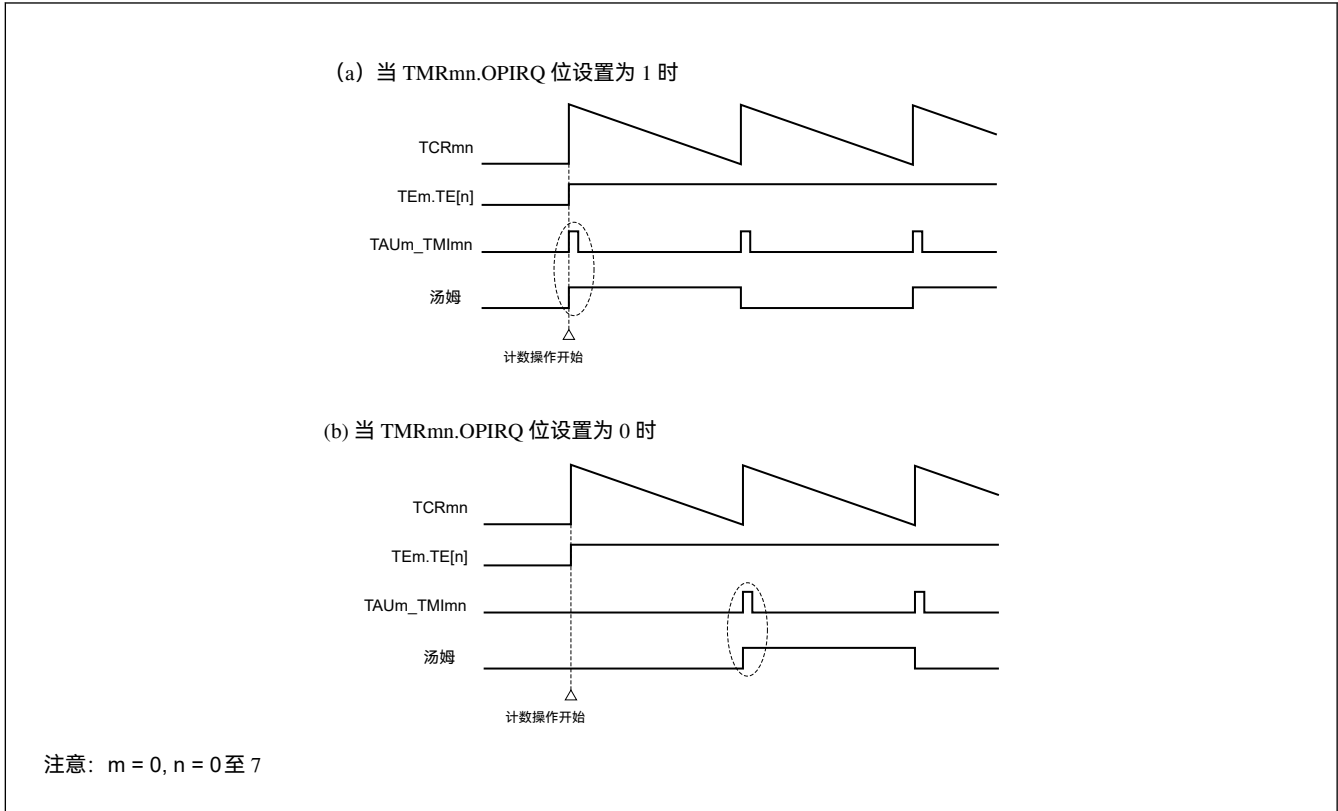


图 16.34 计数开始时定时器中断和 TOMn 输出的操作示例

当 TMR0n.OPIRQ 位设置为 1 时，计数操作开始时输出定时器中断 (TAU0_TMI0n)，并且 TO0n 执行翻转操作。

当 TMR0n.OPIRQ 位设置为 0 时，计数操作开始时不会输出定时器中断 (TAU0_TMI0n)，且 TO0n 也不会改变。计数一个周期后，输出 AUA0_TMI0n，TO0n 执行翻转操作。

16.6 定时器输入 (TI0n) 控制

16.6.1 TI0n 输入电路配置

信号从定时器输入引脚输入，经过噪声滤波器和边沿检测器，然后发送到定时器控制器。启用需要消除噪声的引脚的噪声滤波器。图 16.35 显示了输入电路的配置。

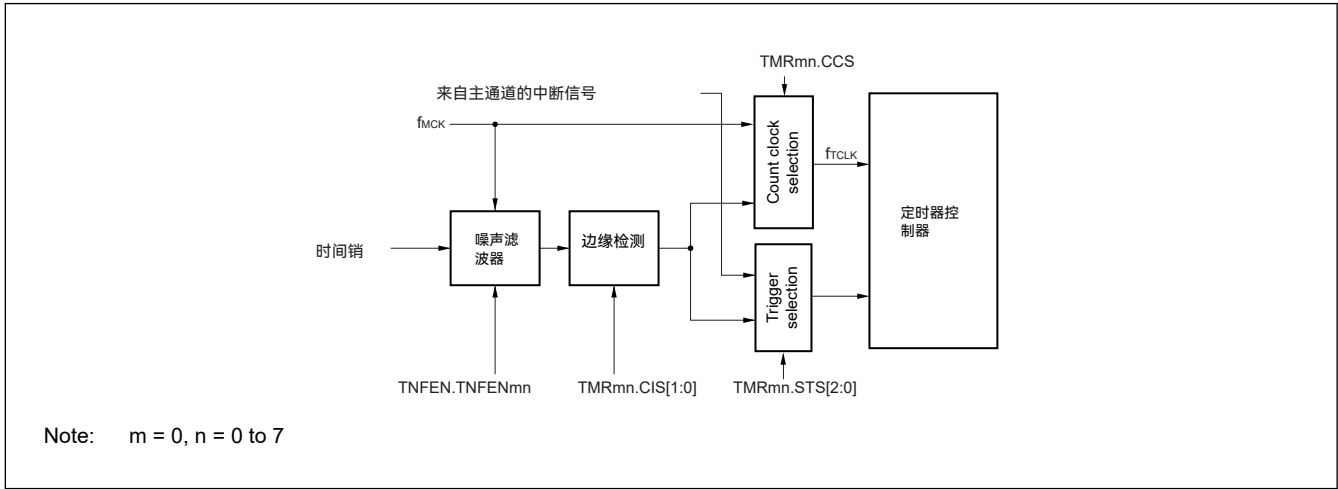


图 16.35 输入电路配置

16.6.2 Noise Filter

When the noise filter is disabled, the input signal is only synchronized with the operating clock (f_{MCK}) for channel n . When the noise filter is enabled, after synchronization with the operating clock (f_{MCK}) for channel n , whether the signal keeps the same value for two clock cycles is detected. Figure 16.36 shows differences in waveforms output from the noise filter between when the noise filter is enabled and disabled.

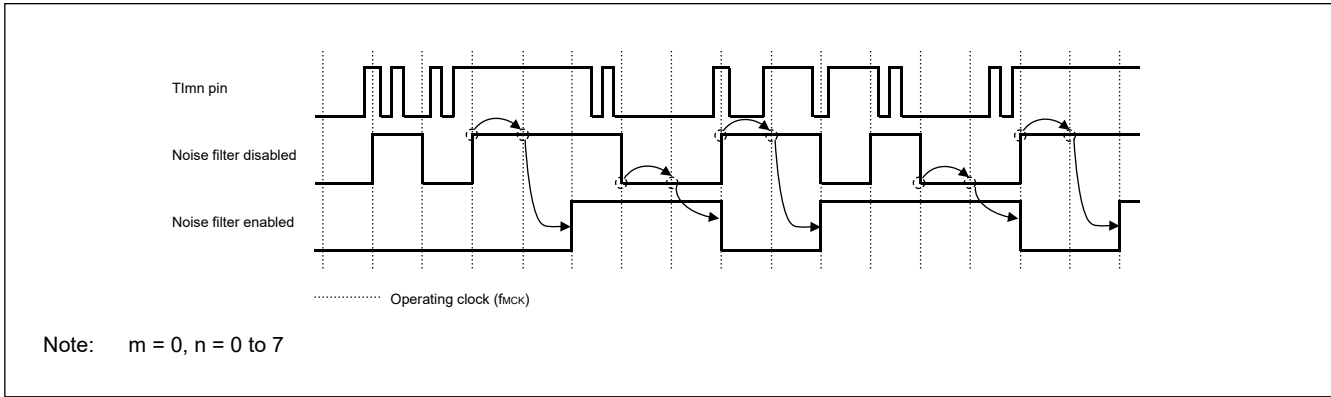


Figure 16.36 Sampling waveforms through TI0n input pin with noise filter enabled and disabled

16.6.3 Cautions on Channel Input Operation

When a timer input pin is set as unused, the operating clock is not supplied to the noise filter. Therefore, after settings are made to use the timer input pin, the following wait time is necessary before a trigger is specified to enable operation of the channel corresponding to the timer input pin.

1. Noise filter is disabled
When bits 12 (CCS), 9 and 8 (STS[1:0]) in the timer mode register 0n (TMR0n) are 0 and then one of them is set to 1, wait for at least two cycles of the operating clock (f_{MCK}), and then set the operation enable trigger bit in the timer channel start register (TS0).
2. Noise filter is enabled
When bits 12 (CCS), 9 and 8 (STS[1:0]) in the timer mode register 0n (TMR0n) are all 0 and then one of them is set to 1, wait for at least four cycles of the operating clock (f_{MCK}), and then set the operation enable trigger bit in the timer channel start register (TS0).

16.7 Independent Channel Operation Function of Timer Array Unit

16.7.1 Operation as an Interval Timer or for Square Wave Output

(1) Interval timer

The timer array unit can be used as a reference timer that generates TAU0_TMI0n (timer interrupt) at fixed intervals. The interrupt generation period can be calculated by the following expression.

Generation period of TAU0_TMI0n (timer interrupt) = Period of count clock $\times$ (Set value of TDR0n + 1)

(2) Operation for square wave output

TO0n performs a toggle operation as soon as TAU0_TMI0n has been generated, and outputs a square wave with a duty factor of 50%.

The period and frequency for outputting a square wave from TO0n can be calculated by the following expressions.

- Period of square wave output from TO0n = Period of count clock $\times$ (Set value of TDR0n + 1) $\times$ 2
- Frequency of square wave output from TO0n = Frequency of count clock / {(Set value of TDR0n + 1) $\times$ 2}

Timer counter register 0n (TCR0n) operates as a down counter in the interval timer mode.

The TCR0n register loads the value of timer data register 0n (TDR0n) at the first count clock after the channel start trigger bit (TS[n], TSH1, TSH3) of timer channel start register 0 (TS0) is set to 1. If the TMR0n.OPIRQ bit of timer mode register

16.6.2 噪声滤波器

当噪声滤波器禁用时，输入信号仅与通道 n 的工作时钟 (f_{MCK}) 同步。当噪声滤波器启用时，在与通道 n 的工作时钟 (f_{MCK}) 同步后，检测信号在两个时钟周期内是否保持不变。图 16.36 显示了噪声滤波器启用和禁用时输出波形的差异。

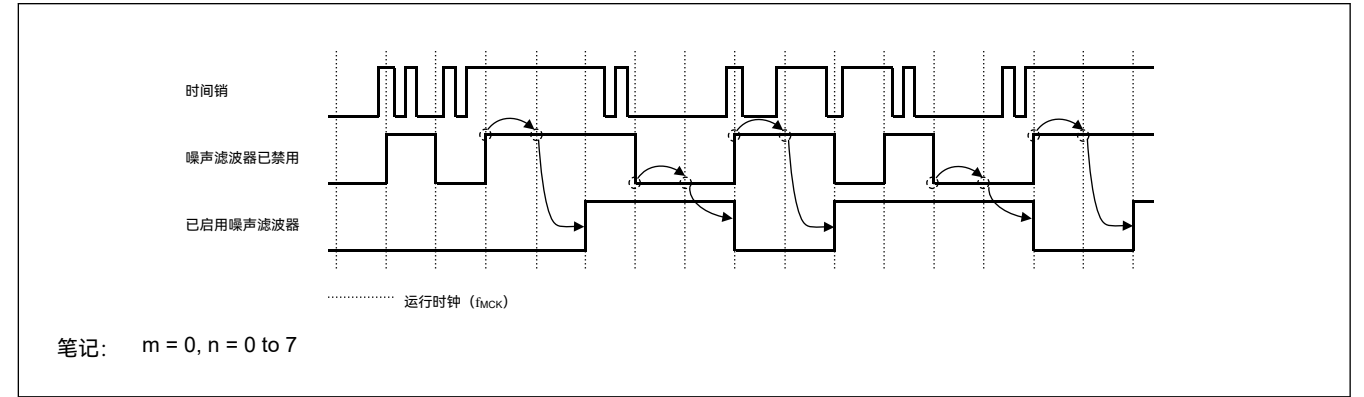


图 16.36: 启用和禁用噪声滤波器时，通过 TI0n 输入引脚采样波形

16.6.3 通道输入操作注意事项

当定时器输入引脚设置为未使用时，噪声滤波器将不会获得工作时钟信号。因此，在设置使用定时器输入引脚后，需要等待一段时间，然后才能指定触发信号以启用与该定时器输入引脚对应的通道。

1. 噪声滤波器已禁用
当定时器模式寄存器 0n (TMR0n) 中的第 12 位 (CCS)、第 9 位和第 8 位 (STS[1:0]) 为 0，然后其中一个被设置为 1 时，等待至少两个工作时钟周期 (f_{MCK})，然后设置定时器通道起始寄存器 (TS0) 中的操作使能触发位。
2. 已启用噪声滤波器
当定时器模式寄存器 0n (TMR0n) 中的第 12 位 (CCS)、第 9 位和第 8 位 (STS[1:0]) 全部为 0，然后其中一个被设置为 1 时，等待至少四个工作时钟周期 (f_{MCK})，然后设置定时器通道起始寄存器 (TS0) 中的操作使能触发位。

16.7 定时器阵列单元的独立通道操作功能

16.7.1 作为间隔定时器或方波输出运行

(1) 间隔定时器

定时器阵列单元可用作参考定时器，以固定间隔生成 TAU0_TMI0n（定时器中断）。中断生成周期可通过以下表达式计算。

TAU0_TMI0n（定时器中断）的生成周期 = 计数时钟周期 $\times$ (TDR0n 的设置值 + 1)

(2) 方波输出操作

TO0n 在 TAU0_TMI0n 生成后立即执行切换操作，并输出占空比为 50% 的方波。

TO0n 输出方波的周期和频率可以通过以下表达式计算。

- 来自 TO0n = 的方波输出周期；计数时钟 $\times$ 的周期 (TDR0n + 1) $\times$ 2 的设置值
- 方波输出频率 (来自 TO0n =) 计数时钟频率 / {(TDR0n + 1) $\times$ 2} 的设置值

定时器计数器寄存器 0n (TCR0n) 在间隔定时器模式下用作递减计数器。

TCR0n 寄存器在定时器通道起始寄存器 0 (TS0) 的通道起始触发位 (TS[n], TSH1, TSH3) 设置为 1 后的第一个计数时钟周期，加载定时器数据寄存器 0n (TDR0n) 的值。如果 TMR0n.

0n (TMR0n) is 0 at this time, TAU0_TMI0n is not output and the output on TO0n is not toggled. If the TMR0n.OPIRQ bit of the TMR0n register is 1, TAU0_TMI0n is output and the output on TO0n is toggled.

After that, the TCR0n register count down in synchronization with the count clock.

When TCR0n = 0x0000, TAU0_TMI0n is output and the output on TO0n is toggled at the next count clock. At the same time, the TCR0n register loads the value of the TDR0n register again. After that, the same operation is repeated.

The TDR0n register can be rewritten at any time. The new value of the TDR0n register becomes valid from the next period.

Note: n: Channel number (n = 0 to 7)

Figure 16.37 shows a block diagram of the operation as an interval timer or a square wave output.

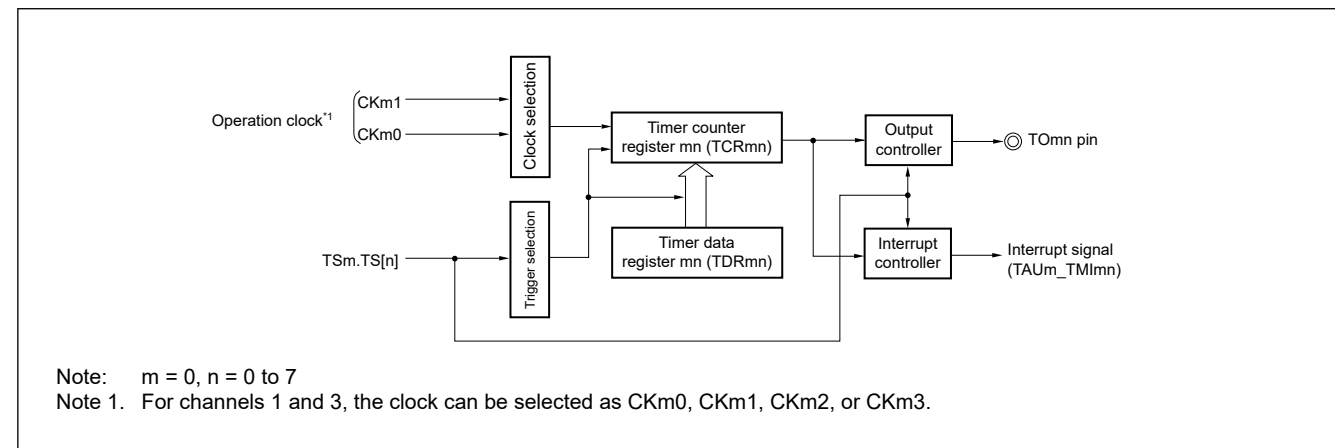


Figure 16.37 Block diagram for operation as an interval timer or for square wave output

Figure 16.38 shows an example of basic timing during operation as an Interval timer or for square wave output (TMR0n.OPIRQ = 1).

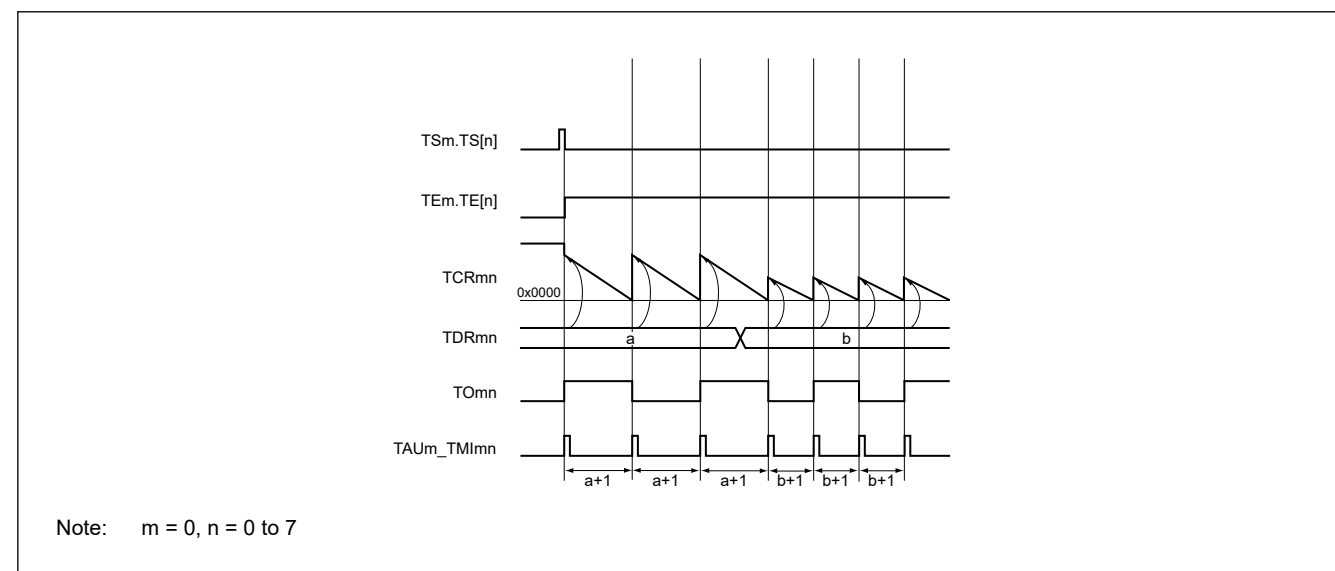


Figure 16.38 Example of basic timing during operation as an interval timer or for square wave output (TMR0n.OPIRQ = 1)

Table 16.13 to Table 16.18 show register settings and procedure for operation when the interval timer or square wave output.

此时定时器模式寄存器 0n (TMR0n) 的 OPIRQ 位为 0, TAU0_TMI0n 不输出, TO0n 的输出也不会翻转。如果 TMR0n 寄存器的 TMR0n.OPIRQ 位为 1, 则 TAU0_TMI0n 输出, TO0n 的输出也会翻转。

之后，TCR0n 寄存器与计数时钟同步倒计时。

当TCR0n = 0x0000时, TAU0_TMI0n 输出, TO0n 的输出在下一个时钟周期翻转。同时, TCR0n 寄存器再次加载 TDR0n 寄存器的值。之后, 重复上述操作。

TDR0n寄存器可以随时被重写。TDR0n寄存器的新值从下一个周期开始生效。

注: n: 频道号(n = 0至7)

图 16.37 显示了作为间隔定时器或方波输出的操作框图。

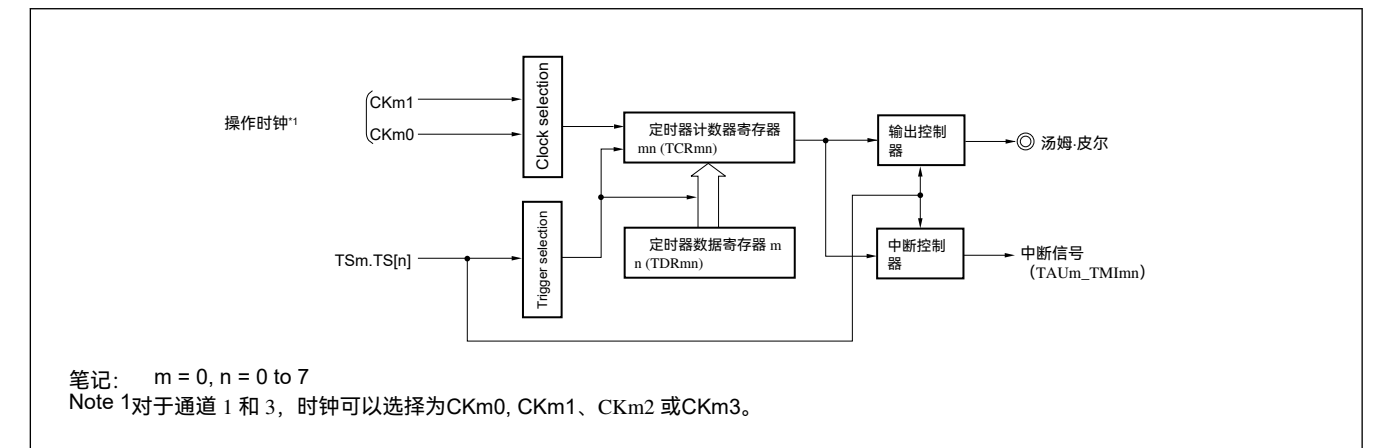


图 16.37 用作间隔定时器或方波输出的框图

图 16.38 显示了作为间隔定时器或方波输出(TMR0n.OPIRQ = 1)运行时的基本计时示例。

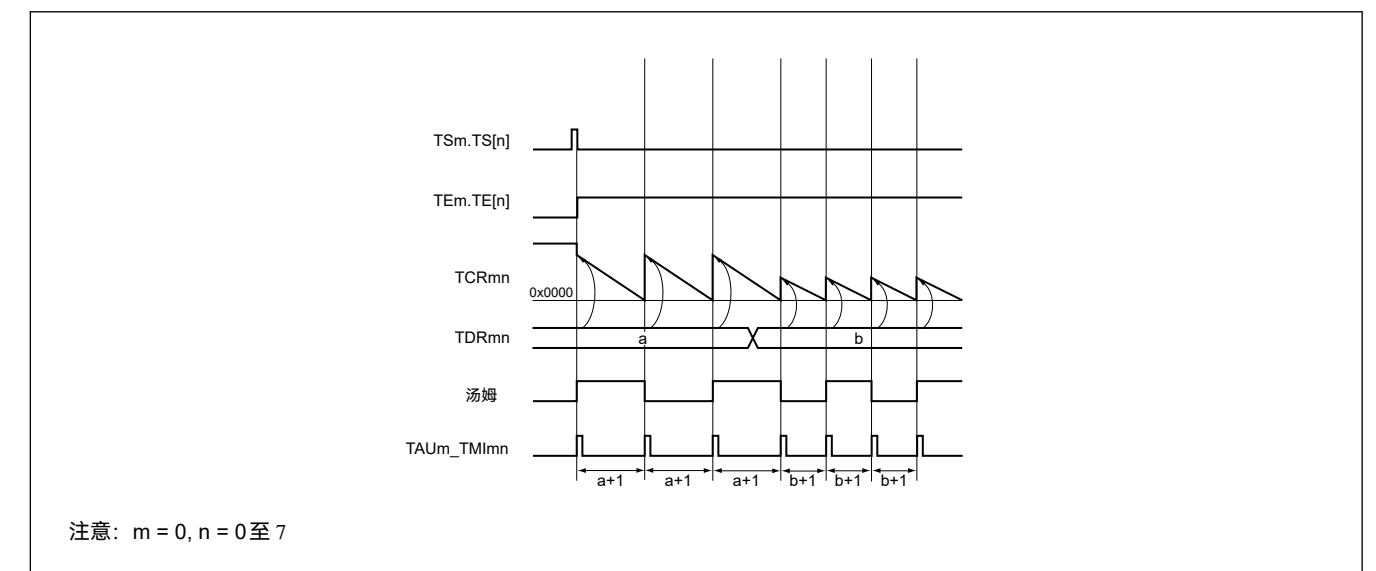


图 16.38 间隔定时器或方波输出运行时的基本定时示例
(TMR0n.OPIRQ = 1)

表 16.13 至表 16.18 显示了间隔定时器或方波输出时的寄存器设置和操作步骤。

Table 16.13 Example of TMR0n settings for operation as an interval timer or for square wave output

Bit	Symbol	Set value	Function
0	OPIRQ	1/0	Setting of operation when counting is started 0: Neither generates TAU0_TMI0n nor inverts timer output when counting is started. 1: Generates TAU0_TMI0n and inverts timer output when counting is started.
3:1	MD[2:0]	000b	Operation mode of channel n 0 0 0: Interval timer mode
5:4	—	00b	Fixed to 0
7:6	CIS[1:0]	00b	Selection of TI0n pin input edge 0 0: Set to 00b because the TI0n input pin is not to be used.
10:8	STS[2:0]	000b	Start trigger selection 0 0 0: Selects only software start.
11	— (n = 0, 5, 7)	0	Fixed to 0 (channels 0, 5, 7)
	SPLIT (n = 1, 3)	1/0	Setting of SPLIT bit (channels 1, 3) 0: 16-bit timer mode 1: 8-bit timer mode
	MASTER (n = 2, 4, 6)	0	Setting of MASTER bit (channels 2, 4, 6) 0: Independent channel operation function.
12	CCS	0	Count clock selection 0: Selects operation clock (f _{MCK}).
13	—	0	Fixed to 0.
15:14	CKS[1:0]	00b to 11b	Selection of the operating clock (f _{MCK}) 0 0: Selects CK00 as the operating clock for channel n. 0 1: Selects CK02 as the operating clock (this can only be selected for channels 1 and 3). 1 0: Selects CK01 as the operating clock for channel n. 1 1: Selects CK03 as the operating clock (this can only be selected for channels 1 and 3).

Table 16.14 Example of TO0 settings for operation as an interval timer or for square wave output

Bit	Symbol	Set value	Function
n	TO[n]	1/0	Timer output of channel n 0: Outputs 0 from TO0n. 1: Outputs 1 from TO0n.

Table 16.15 Example of TOE0 settings for operation as an interval timer or for square wave output

Bit	Symbol	Set value	Function
n	TOE[n]	1/0	Enabling or disabling timer output for channel n 0: Stops the TO0n output operation by counting operation. 1: Enables the TO0n output operation by counting operation.

表 16.13 TMR0n 设置示例，可用于间隔定时器或方波输出

少量	象征	设置值	功能
0	OPIRQ	1/0	计数开始时的操作设置 0: 计数开始时既不生成 TAU0_TMI0n，也不反转定时器输出。 1: 计数开始时生成 TAU0_TMI0n 并反转定时器输出。
3:1	MD[2:0]	000b	通道 n 的工作模式 0 0 0:间隔定时器模式
5:4	—	00b	固定为 0
7:6	CIS[1:0]	00b	选择 TI0n 引脚输入边沿0 0:设置为 00b，因为 TI0n 输入引脚不使用。
10:8	STS[2:0]	000b	启动触发器选择0 0 0:仅选择软件启动。
11	— (n = 0, 5, 7)	0	固定为 0（通道 0、5、7）
	SPLIT (n = 1, 3)	1/0	SPLIT 位设置（通道 1、3）：0: 16 位定时器模式；1: 8 位定时器模式
	MASTER (n = 2, 4, 6)	0	MASTER 位（通道 2、4、6）设置为 0: 独立通道操作功能。
12	CCS	0	计数时钟选择 0: 选择操作时钟(f _{MCK})。
13	—	0	固定值为0。
15:14	CKS[1:0]	00b 至 11b	选择工作时钟 (f _{MCK}) 0 0: 选择 CK00 作为通道 n 的工作时钟。0 1: 选择 CK02 作为工作时钟（只能为通道 1 和 3 选择）。 1 0: 选择 CK01 作为通道 n 的工作时钟。 1 1: 选择 CK03 作为工作时钟（只能为通道 1 和 3 选择）。

表 16.14 TO0 设置示例，可用于间隔定时器或方波输出

少量	象征	设置值	功能
n	TO[n]	1/0	通道 n 的定时器输出 0: 从 TO0n 输出 0。 1: 从 TO0n 输出 1。

表 16.15 TOE0 设置示例，可用于间隔定时器或方波输出

少量	象征	设置值	功能
n	TOE[n]	1/0	启用或禁用通道 n 的定时器输出 0: 通过计数操作停止 TO0n 输出操作。 1: 通过计数操作启用 TO0n 输出操作。

Table 16.16 Example of TOL0 settings for operation as an interval timer or for square wave output

Bit	Symbol	Set Value	Function
n	— (n = 0)	0	Fixed to 0 (channels 0)
	TOL[n] (n = 1 to 7)		Control of timer output of channel n (channels 1 to 7) 0: Set this bit to 0 when TOM0.TOM[n] = 0 (master channel output mode).

Table 16.17 Example of TOM0 settings for operation as an interval timer or for square wave output

Bit	Symbol	Set value	Function
n	— (n = 0)	0	Fixed to 0 (channels 0)
	TOM[n] (n = 1 to 7)		Control of timer output mode of channel n (channels 1 to 7) 0: Sets master channel output mode.

Table 16.18 Procedure for operations when the interval timer or square wave output function is to be used (1 of 2)

	Step	Software operation	Hardware state
TAU default setting		—	Power-off state (Clock supply is stopped and writing to each register is disabled.)
	<1>	Sets the MSTPD0 bit of Module Stop Control Register D (MSTPCRD) to 0.	→ Power-on state. Each channel stops operating. (Clock supply is started and writing to each register is enabled.)
	<2>	Sets timer clock select register 0 (TPS0). Determines clock frequencies of CK00 to CK03.	—
Channel default setting	<3>	Sets timer mode register 0n (TMR0n) (determines operation mode of channel). Sets interval (period) value to timer data register 0n (TDR0n).	Channel stops operating. (Clock is supplied and some power is consumed.)
	<4>	To use the TO0n output Clears the TOM0.TOM[n] bit of timer output mode register 0 (TOM0) to 0 (master channel output mode). Clears the TOL0.TOL[n] bit to 0.	The TO0n pin goes into Hi-Z output state.
		Sets the TO0.TO[n] bit and determines default level of the TO0n output.	→ The TO0n default setting level is output when the Ppq Direction Register (PDRp) is in the output mode.
		Sets the TOE0.TOE[n] bit to 1 and enables operation of TO0n. Sets the Ppq Direction Register (PDRp) to 1.	→ TO0n does not change because channel stops operating. → The TO0n pin outputs the TO0n set level.
Operation start	<5>	(Sets the TOE0.TOE[n] bit to 1 only if using TO0n output and resuming operation.). Sets the TS0.TS[n] (TSH1, TSH3) bit to 1. The TS0.TS[n] (TSH1, TSH3) bit automatically returns to 0 because it is a trigger bit.	→ TE0.TE[n] (TEH1, TEH3) = 1, and count operation starts. Value of the TDR0n register is loaded to timer counter register 0n (TCR0n). TAU0_TMI0n is generated and TO0n performs toggle operation if the TMR0n.OPIRQ bit of the TMR0n register is 1.
During operation	<6>	Set values of the TMR0n register, TOM0.TOM[n], and TOL0.TOL[n] bits cannot be changed. Set value of the TDR0n register can be changed. The TCR0n register can always be read. The TSR0n register is not used. Set values of the TO0 and TOE0 registers can be changed.	Counter (TCR0n) counts down. When count value reaches 0x0000, the value of the TDR0n register is loaded to the TCR0n register again and the count operation is continued. By detecting TCR0n = 0x0000, TAU0_TMI0n is generated and TO0n performs toggle operation. After that, the above operation is repeated.

表 16.16 TOL0 设置示例，适用于间隔定时器或方波输出

少量	象征	设置值	功能
n	— (n = 0)	0	固定为 0（通道 0）
	TOL[n] (n = 1 to 7)		控制通道 n（通道 1 至 7）的定时器输出 0: 当 TOM0.TOM[n] = 0 时（主通道输出模式），将此位设置为 0。

表 16.17 TOM0 设置示例，可用于间隔定时器或方波输出

少量	象征	设置值	功能
n	— (n = 0)	0	固定为 0（通道 0）
	汤姆[n] (n = 1 至 7)		控制通道 n（通道 1 至 7）的定时器输出模式 0: 设置主通道输出模式。

表 16.18 使用间隔定时器或方波输出功能时的操作步骤（1/2）

	步	软件操作	硬件状态
TAU 默认设置		—	关机状态 (时钟信号停止，所有寄存器的写入操作均被禁用。)
	<1>	将模块停止控制寄存器 D (MSTPCRD) 的 MSTP D0 位设置为 0。	→ 上电状态。各通道停止工作。 (时钟电源已启动，并且对每个寄存器的写入操作已启用。)
	<2>	设置定时器时钟选择寄存器 0 (TPS0)。确定 CK00 至 CK03 的时钟频率。	—
通道默认设置	<3>	设置定时器模式寄存器 0n (TMR0n)（确定通道的操作模式）。 将定时器数据寄存器 0n (TDR0n) 的间隔（周期）值设置到定时器数据寄存器 0n 中。	频道停止运行。 (时钟信号被提供，并消耗一定功率。)
	<4>	要使用 TO0n 输出，请将定时器输出模式寄存器 0 (TOM0) 的 TOM[n] 位清除为 0 (主通道输出模式)。 将 TOL0.TOL[n] 位清除为 0。	TO0n 引脚进入高阻抗输出状态。
		设置 TO0.TO[n] 位，并确定 TO0n 输出的默认电平。	→ 当 Ppq 方向寄存器 (PDRp) 处于输出模式时，输出 TO0n 默认设置电平。
		将 TOE0.TOE[n] 位设置为 1，使 TO0n 能够运行。 将 Ppq 方向寄存器 (PDRp) 设置为 1。	→ TO0n 不会改变，因为通道停止工作。 → TO0n 引脚输出 TO0n 设定电平。
操作开始	<5>	(仅当使用 TO0n 输出并恢复操作时，才将 TOE0.TOE[n] 位设置为 1。) 将 TS0.TS[n] (TSH1, TSH3) 位设置为 1。 TS0.TS[n] (TSH1, TSH3) 位会自动返回 0，因为它是一个触发位。	→ TE0.TE[n] (TEH1, TEH3) = 1，) 开始计数操作。TDR0n 寄存器的值加载到定时器计数器寄存器 0n (TCR0n)。生成 TAU0_TMI0n，如果 TMR0n 寄存器的 TMR0n.OPIRQ 位为 1，则 TO0n 执行翻转操作。
运行期间	<6>	TMR0n 寄存器的设置值、TOM0.TOM[n] 和 TOL0.TOL[n] 位不能更改。 TDR0n 寄存器的设定值可以更改。TCR0n 寄存器始终可以读取。 TSR0n 寄存器未使用。 TO0 和 TOE0 寄存器的设置值可以更改。	计数器 (TCR0n) 递减计数。当计数达到 0x0000 时，TDR0n 寄存器的值再次加载到 TCR0n 寄存器，计数继续进行。通过检测 TCR0n = 0x0000，生成 TAU0_TMI0n，TO0n 执行翻转操作。 之后，重复上述操作。

Table 16.18 Procedure for operations when the interval timer or square wave output function is to be used (2 of 2)

	Step	Software operation	Hardware state
Operation stop	<7>	The TT0.TT[n] (TTH1, TTH3) bit is set to 1. The TT0.TT[n] (TTH1, TTH3) bit automatically returns to 0 because it is a trigger bit.	→ TE0.TE[n] (TEH1, TEH3) = 0, and count operation stops. The TCR0n register holds count value and stops. The TO0n output is not initialized and retains its current state.
	<8>	The TOE0.TOE[n] bit is cleared to 0 and value is set to the TO0.TO[n] bit. To resume operation, go to step <5>. To terminate the operation, go to step <9>	→ The TO0n pin outputs the TO0.TO[n] bit set level.
TAU stop	<9>	To hold the TO0n pin output level Set PSEL[2:0] bits to 000b after the value to be held is set to the Ppq Output Data Register (PODRp). When holding the TO0n pin output level is not necessary Setting not required.	→ The TO0n pin output level is held by port function.
	<10>	Sets the MSTPD0 bit of Module Stop Control Register D (MSTPCRD) to 1.	→ This stops supply of the input clock to timer array unit 0. Power-off state.

Note: n = 0 to 7, p = 0 to 9, q = 00 to 15

16.7.2 Operation as an External Event Counter

The timer array unit can be used as an external event counter that counts the number of times the valid input edge (external event) is detected in the TI0n pin. When a specified count value is reached, the event counter generates an interrupt. The specified number of counts can be calculated by the following expression.

Specified number of counts = Set value of TDR0n + 1

Timer counter register 0n (TCR0n) operates as a down counter in the event counter mode.

The TCR0n register loads the value of timer data register 0n (TDR0n) by setting any TS0.TS[n] bit to 1.

The TCR0n register counts down each time the valid input edge of the TI0n pin has been detected. When TCR0n = 0x0000, the TCR0n register loads the value of the TDR0n register again, and outputs TAU0_TMI0n.

After that, the above operation is repeated.

An irregular waveform that depends on external events is output from the TO0n pin. Stop the output by setting the TOE[n] bit of timer output enable register 0 (TOE0) to 0.

The TDR0n register can be rewritten at any time. The new value of the TDR0n register becomes valid during the next count period.

Instead of using the TI0n pin input, a channel specified for the external event counter function can also use the timer input selected in the TIS0 register as its input source to drive counting.

Figure 16.39 shows a block diagram of the operation as an external event counter.

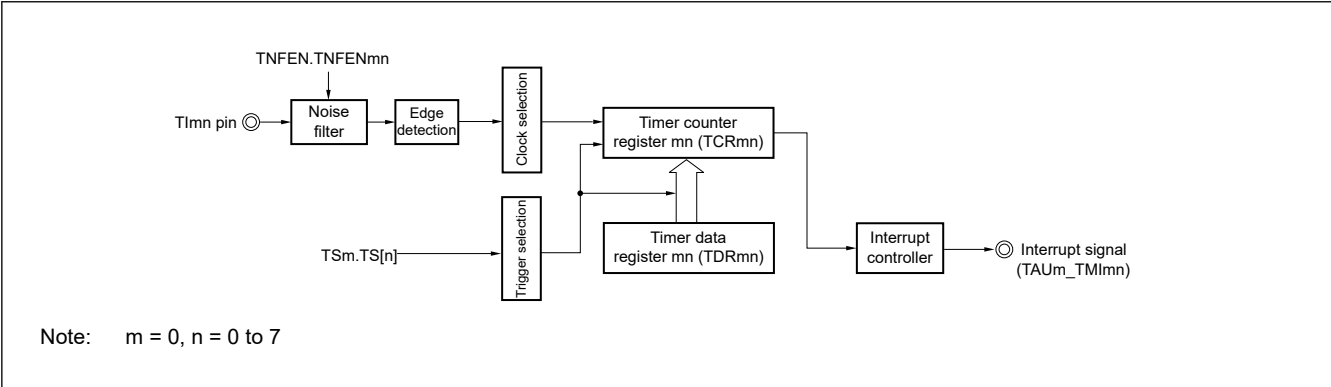


Figure 16.39 Block diagram for operation as an external event counter

表 16.18 使用间隔定时器或方波输出功能时的操作步骤 (2/2)

	步	软件操作	硬件状态
停止操作	<7>	TT0.TT[n] (TTH1, TTH3) 位设置为 1。TT0.TT[n] (TTH1, TTH3) 位会自动返回 0，因为它是一个触发位。	→ TE0.TE[n] (TEH1, TEH3) = 0，计数操作停止。TCR0n 寄存器保持计数值并停止。TO0n 输出未初始化，并保持其当前状态。
	<8>	TOE0.TOE[n] 位被清除为 0，并且 TO0.TO[n] 位的值被设置。要恢复操作，请转到步骤<5>。要终止操作，请转到步骤<9>	→ TO0n 引脚输出 TO0.TO[n] 位设置电平。
停止！	<9>	要保持 TO0n 引脚输出电平，在将要保持的值设置到 Ppq 输出数据寄存器 (PODRp) 之后，将 PSEL[2:0] 位设置为 000b。保持 TO0n 引脚输出电平时，无需设置。	→ TO0n 引脚的输出电平由端口功能保持。
	<10>	将模块停止控制寄存器 D (MSTPCRD) 的 MSTPD0 位设置为 1。	→ 这将停止向定时器阵列单元 0 提供输入时钟。关机状态。

注意: n = 0 至 7, p = 0 至 9, q = 00 至 15

16.7.2 作为外部事件计数器运行

定时器阵列单元可用作外部事件计数器，用于统计TI0n引脚上检测到的有效输入边沿（外部事件）的次数。当达到指定的计数值时，事件计数器会产生中断。指定的计数值可通过以下表达式计算。

指定计数= 设置TDR0n + 1的值

定时器计数器寄存器 0n (TCR0n) 在事件计数器模式下作为递减计数器运行。

TCR0n 寄存器通过将任意 TS0.TS[n] 位设置为 1 来加载定时器数据寄存器 0n (TDR0n) 的值。

每当检测到 TI0n 引脚的有效输入边沿时，TCR0n 寄存器就会递减计数。当 TCR0n = 0x0000时，TCR0n 寄存器会再次加载 TDR0n 寄存器的值，并输出 TAU0_TMI0n。

之后，重复上述操作。

TO0n 引脚输出一个取决于外部事件的不规则波形。将定时器输出使能寄存器 0 (TOE0) 的 TOE[n]位设置为 0 即可停止输出。

TDR0n寄存器可以随时被重写。TDR0n寄存器的新值在下一个计数周期内生效。

除了使用 TI0n 引脚输入外，为外部事件计数器功能指定的通道还可以使用 TI0 寄存器中选择的定时器输入作为其输入源来驱动计数。

图 16.39 显示了作为外部事件计数器的操作框图。

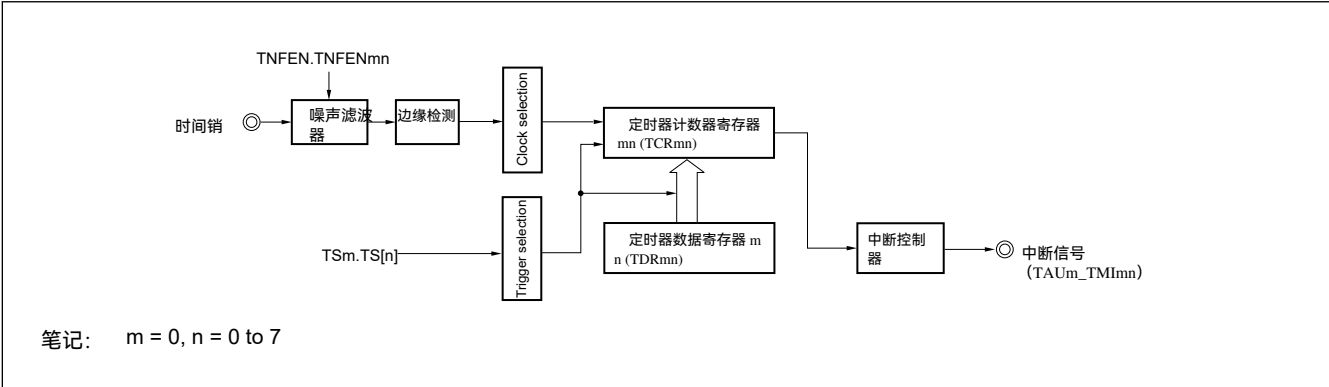


图 16.39 外部事件计数器运行框图

Figure 16.40 shows an example of basic timing during operation as an external event counter.

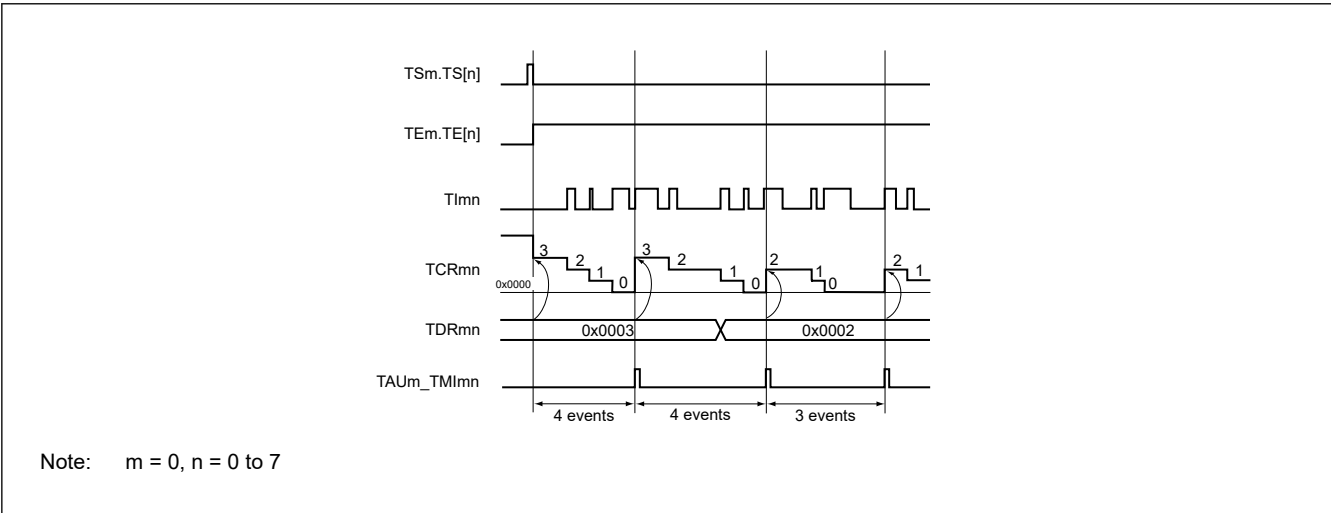


Figure 16.40 Example of basic timing during operation as an external event counter

Table 16.19 to Table 16.24 show register settings and procedure for operation when the external event counter.

Table 16.19 Example of TMR0n settings in external event counter mode (1 of 2)

Bit	Symbol	Set value	Function
0	OPIRQ	0	Setting of operation when counting is started 0: Neither generates TAU0_TMI0n nor inverts timer output when counting is started.
3:1	MD[2:0]	011b	Operation mode of channel n 0 1 1: Event count mode
5:4	—	00b	Fixed to 0
7:6	CIS[1:0]	00b to 10b	Selection of TI0n pin input edge 0 0: Detects falling edge. 0 1: Detects rising edges. 1 0: Detects both edge. Others: Setting prohibited.
10:8	STS[2:0]	000b	Start trigger selection 0 0 0: Selects only software start.
11	— (n = 0, 5, 7)	0	Fixed to 0 (channels 0, 5, 7)
	SPLIT (n = 1, 3)	1/0	Setting of SPLIT bit (channels 1, 3) 0: 16-bit timer mode 1: 8-bit timer mode
	MASTER (n = 2, 4, 6)	0	Setting of MASTER bit (channels 2, 4, 6) 0: Independent channel operation function
12	CCS	1	Count clock selection 1: Selects the TI0n pin input valid edge.
13	—	0	Fixed to 0.

图 16.40 显示了作为外部事件计数器运行时的基本计时示例。

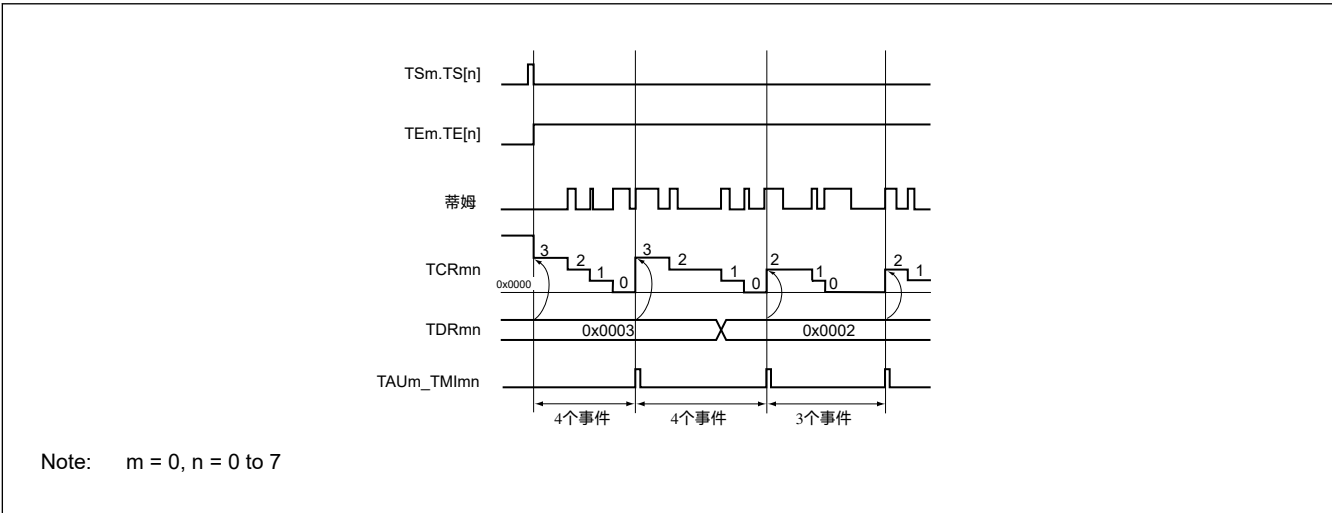


图 16.40 外部事件计数器运行期间的基本计时示例

表 16.19 至表 16.24 显示了外部事件计数器的寄存器设置和操作程序。

表 16.19 外部事件计数器模式下 TMR0n 设置示例 (1/2)

少量	象征	设置值	功能
0	OPIRQ	0	计数开始时的操作设置 0: 计数开始时既不生成 TAU0_TMI0n, 也不反转定时器输出。
3:1	MD[2:0]	011b	通道 n 的工作模式 0 1 1: 事件计数模式
5:4	—	00b	固定为 0
7:6	CIS[1:0]	00b 至 10b	TI0n引脚输入边沿选择0 0:检测下降沿。0 1:检测上升沿。1 0:检测上升沿和下降沿。 其他: 禁止设置。
10:8	STS[2:0]	000b	启动触发选择 0 0 0: 仅选择软件启动。
11	— (n = 0, 5, 7)	0	固定为 0 (通道 0、5、7)
	SPLIT (n = 1, 3)	1/0	SPLIT 位设置 (通道 1、3) : 0: 16 位定时器模式; 1: 8 位定时器模式
	MASTER (n = 2, 4, 6)	0	主控位设置 (通道 2、4、6) 0: 独立通道操作功能
12	CCS	1	计数时钟选择 1: 选择 TI0n 引脚输入有效边沿。
13	—	0	固定值为0。

Table 16.19 Example of TMR0n settings in external event counter mode (2 of 2)

Bit	Symbol	Set value	Function
15:14	CKS[1:0]	00b to 11b	Selection of the operating clock (f_{MCK}) 0 0: Selects CK00 as the operating clock for channel n. 0 1: Selects CK02 as the operating clock (this can only be selected for channels 1 and 3). 1 0: Selects CK01 as the operating clock for channel n. 1 1: Selects CK03 as the operating clock (this can only be selected for channels 1 and 3).

Table 16.20 Example of TO0 settings in external event counter mode

Bit	Symbol	Set value	Function
n	TO[n]	0	Timer output of channel n 0: Outputs 0 from TO0n.

Table 16.21 Example of TOE0 settings in external event counter mode

Bit	Symbol	Set value	Function
n	TOE[n]	0	Enabling or disabling timer output for channel n 0: Stops the TO0n output operation by counting operation.

Table 16.22 Example of TOL0 settings in external event counter mode

Bit	Symbol	Set value	Function
n	— (n = 0)	0	Fixed to 0 (channels 0)
	TOL[n] (n = 1 to 7)		Control of timer output of channel n (channels 1 to 7) 0: Set this bit to 0 when TOM0.TOM[n] = 0 (master channel output mode).

Table 16.23 Example of TOM0 settings in external event counter mode

Bit	Symbol	Set value	Function
n	— (n = 0)	0	Fixed to 0 (channels 0)
	TOM[n] (n = 1 to 7)		Control of timer output mode of channel n (channels 1 to 7) 0: Sets master channel output mode.

Table 16.24 Procedure for operations when the external event counter function is to be used (1 of 2)

	Step	Software operation	Hardware state
TAU default setting		—	Power-off state (Clock supply is stopped and writing to each register is disabled.)
	<1>	Sets the MSTPD0 bit of Module Stop Control Register D (MSTPCRD) to 0.	→ Power-on state. Each channel stops operating. (Clock supply is started and writing to each register is enabled.)
	<2>	Sets timer clock select register 0 (TPS0). Determines clock frequencies of CK00 to CK03.	—
Channel default setting	<3>	Sets the corresponding bit of the TAU noise filter enable register (TNFEN) to 0 (off) or 1 (on). Sets timer mode register 0n (TMR0n) (determines operation mode of channel). Sets number of counts to timer data register 0n (TDR0n). Clears the TOE[n] bit of timer output enable register 0 (TOE0) to 0.	Channel stops operating. (Clock is supplied and some power is consumed.)

表 16.19 外部事件计数器模式下 TMR0n 设置示例 (2/2)

少量	象征	设置值	功能
15:14	CKS[1:0]	00b to 11b	选择工作时钟 (f) 0 0: 选择 CK00 作为通道 n 的工作时钟。 0 1: 选择 CK02 作为工作时钟 (仅通道 1 可以选择此项) 以及 3)。 1 0: 选择 CK01 作为通道 n 的工作时钟。 1 1: 选择 CK03 作为工作时钟 (只能为通道 1 和 3 选择)。

表 16.20 外部事件计数器模式下 TO0 设置示例

少量	象征	设置值	功能
n	TO[n]	0	通道 n 的定时器输出 0: 从 TO0n 输出 0。

表 16.21 外部事件计数器模式下 TOE0 设置示例

少量	象征	设置值	功能
n	TOE[n]	0	启用或禁用通道 n 的定时器输出 0: 通过计数操作停止 TO0n 输出操作。

表 16.22 外部事件计数器模式下 TOL0 设置示例

少量	象征	设置值	功能
n	— (n = 0)	0	固定为 0 (通道 0)
	TOL[n] (n = 1 to 7)		控制通道 n (通道 1 至 7) 的定时器输出 0: 当 TOM0.TOM[n] = 0 时 (主通道输出模式)，将此位设置为 0。

表 16.23 外部事件计数器模式下 TOM0 设置示例

少量	象征	设置值	功能
n	— (n = 0)	0	固定为 0 (通道 0)
	汤姆[n] (n = 1 至 7)		控制通道 n (通道 1 至 7) 的定时器输出模式 0: 设置主通道输出模式。

表 16.24 使用外部事件计数器功能时的操作步骤 (1/2)

	步	软件操作	硬件状态
TAU 默认设置		—	关机状态 (时钟信号停止，所有寄存器的写入操作均被禁用。)
	<1>	将模块停止控制寄存器 D (MSTPCRD) 的 MSTPD0 位设置为 0。	→ 上电状态。各通道停止工作。 (时钟电源已启动，并且对每个寄存器的写入操作已启用。)
	<2>	设置定时器时钟选择寄存器 0 (TPS0)。确定 CK00 到 CK03 的时钟频率	—
通道默认设置	<3>	将 TAU 噪声滤波器使能寄存器 (TNFEN) 的相应位设置为 0 (关闭) 或 1 (开启)。 设置定时器模式寄存器 0n (TMR0n) (确定通道的操作模式)。 设置定时器数据寄存器 0n (TDR0n) 的计数次数。 将定时器输出使能寄存器 0 (TOE0) 的 TOE[n] 位清除为 0。	频道停止运行。 (时钟信号被提供，并消耗一定功率。)

Table 16.24 Procedure for operations when the external event counter function is to be used (2 of 2)

	Step	Software operation		Hardware state
Operation start	<4>	Sets the TS0.TS[n] bit to 1. The TS0.TS[n] bit automatically returns to 0 because it is a trigger bit.	→	TE0.TE[n] = 1 and count operation starts. Value of the TDR0n register is loaded to timer counter register 0n (TCR0n) and detection of the TI0n pin input edge is awaited.
During operation	<5>	Set value of the TDR0n register can be changed. The TCR0n register can always be read. The TSR0n register is not used. Set values of the TMR0n register, TOM0.TOM[n], TOL0.TOL[n], TO0.TO[n], and TOE0.TOE[n] bits cannot be changed.		Counter (TCR0n) counts down each time input edge of the TI0n pin has been detected. When count value reaches 0x0000, the value of the TDR0n register is loaded to the TCR0n register again, and the count operation is continued. By detecting TCR0n = 0x0000, the TAU0_TMI0n output is generated. After that, the above operation is repeated.
Operation stop	<6>	The TT0.TT[n] bit is set to 1. The TT0.TT[n] bit automatically returns to 0 because it is a trigger bit. To resume operation, go to step <4>. To terminate the operation, go to step <7>	→	TE0.TE[n] = 0, and count operation stops. The TCR0n register holds count value and stops.
TAU stop	<7>	Sets the MSTPD0 bit of Module Stop Control Register D (MSTPCRD) to 1.	→	This stops supply of the input clock to timer array unit 0. Power-off state.

Note: n = 0 to 7

16.7.3 Operation as a Frequency Divider (Channel 0 of Unit 0 only)

The timer array unit can be used as a frequency divider that divides a clock input to the TI00 pin and outputs the result from the TO00 pin.

The divided clock frequency output from TO00 can be calculated by the following expression.

- When rising edge/falling edge is selected:
Divided clock frequency = Input clock frequency / {(Set value of TDR00 + 1) × 2}
- When both edges are selected:
Divided clock frequency ≈ Input clock frequency / (Set value of TDR00 + 1)

Timer counter register 00 (TCR00) operates as a down counter in the interval timer mode.

After the channel start trigger bit (TS[0]) of timer channel start register 0 (TS0) is set to 1, the TCR00 register loads the value of timer data register 00 (TDR00) when the TI00 valid edge is detected.

If the OPIRQ bit of timer mode register 00 (TMR00) is 0 at this time, TAU0_TMI00 is not output and the output on TO00 is not toggled. If the OPIRQ bit of timer mode register 00 (TMR00) is 1, TAU0_TMI00 is output and the output on TO00 is toggled.

After that, the TCR00 register counts down at the valid edge of the TI00 pin. When TCR00 = 0x0000, it toggles the output on TO00. At the same time, the TCR00 register loads the value of the TDR00 register again, and continues counting.

If detection of both the edges of the TI00 pin is selected, the duty factor error of the input clock affects the divided clock period of the TO00 output.

The period of the TO00 output clock includes a sampling error of one period of the operation clock.

Clock period of TO00 output = Ideal TO00 output clock period ± Operation clock period (error)

The TDR00 register can be rewritten at any time. The new value of the TDR00 register becomes valid during the next count period.

Figure 16.41 shows a block diagram for operation as a frequency divider.

表 16.24 使用外部事件计数器功能时的操作步骤 (2/2)

	步	软件操作		硬件状态
操作开始	<4>	将 TS0.TS[n] 位设置为 1。 TS0.TS[n] 位会自动返回 0，因为它是一个触发位。	→	TE0.TE[n] = 1 计数操作开始。 将 TDR0n 寄存器的值加载到定时器计数器寄存器 0n (TCR0n) 中，并等待检测 TI0n 引脚输入边沿。
运行期间	<5>	TDR0n寄存器的设定值可以更改。 TCR0n寄存器始终可以读取。 TSR0n 寄存器未使用。 TMR0n 寄存器的设置值，TOM0.TOM[n]，TOL0.TOL[n]，TO0.TO[n]和 TOE0.TOE[n] 位不能更改。		计数器 (TCR0n) 在每次输入边沿时递减一次已检测到TI0n引脚的计数。当计数值达到0x0000时，TDR0n寄存器的值再次加载到TCR0n寄存器，计数操作继续进行。通过检测TCR0n =和0x0000，生成TAU0_TMI0n输出。 之后，重复上述操作。
停止操作	<6>	TT0.TT[n] 位设置为 1。 TT0.TT[n] 位会自动返回 0，因为它是一个触发位。 要恢复操作，请转到步骤<4>。 要终止操作，请转到步骤<7>	→	TE0.TE[n] = 0，计数操作停止。 TCR0n寄存器保存计数值并停止。
停止!	<7>	将模块停止控制寄存器 D (MSTPCRD) 的 MSTPD0 位设置为 1。	→	这将停止向定时器阵列单元 0 提供输入时钟。 关机状态。

注意: n = 0 至 7

16.7.3 作为分频器运行（仅限 0 号单元的 0 号通道）

定时器阵列单元可用作分频器，将输入到 TI00 引脚的时钟信号分频，并将结果从TO00 pin输出。

TO00 输出的分频时钟频率可以通过以下表达式计算。

- 选择上升沿/下降沿时：
分频时钟频率= 输入时钟频率 / {(TDR00 的设置值+ 1)× 2}
- 当两条边都被选中时：
分频时钟频率≈输入时钟频率 / (TDR00 的设置值+ 1)

定时器计数器寄存器 00 (TCR00) 在间隔定时器模式下用作递减计数器。

当定时器通道启动寄存器 0 (TS0) 的通道启动触发位 (TS [0]) 设置为 1 时，检测到 TI00 有效边沿时，TCR00 寄存器会将定时器数据寄存器 00 (TDR00) 的值加载到寄存器中。

如果此时定时器模式寄存器 00 (TMR00) 的 OPIRQ 位为 0，则不输出 TAU0_TMI00，并且 TO00 的输出也不会翻转。如果此时定时器模式寄存器 00 (TMR00) 的 OPIRQ 位为 1，则输出 TAU0_TMI00，并且 TO00 的输出会翻转。

之后，TCR00 寄存器在TI00 pin. 的有效边沿处开始倒计时。当 TCR00 = 0x0000时，它会翻转 TO00 的输出。同时，TCR00 寄存器再次加载 TDR00 寄存器的值，并继续计数。

如果选择检测 TI00 引脚的两个边沿，则输入时钟的占空比误差会影响 TO00 输出的分频时钟周期。

TO00 输出时钟的周期包含一个工作时钟周期的采样误差。

TO00 输出时钟周期=理想 TO00 输出时钟周期±工作时钟周期（误差）

TDR00寄存器可以随时被重写。TDR00寄存器的新值在下一个计数周期内生效。

图 16.41 显示了作为分频器工作的框图。

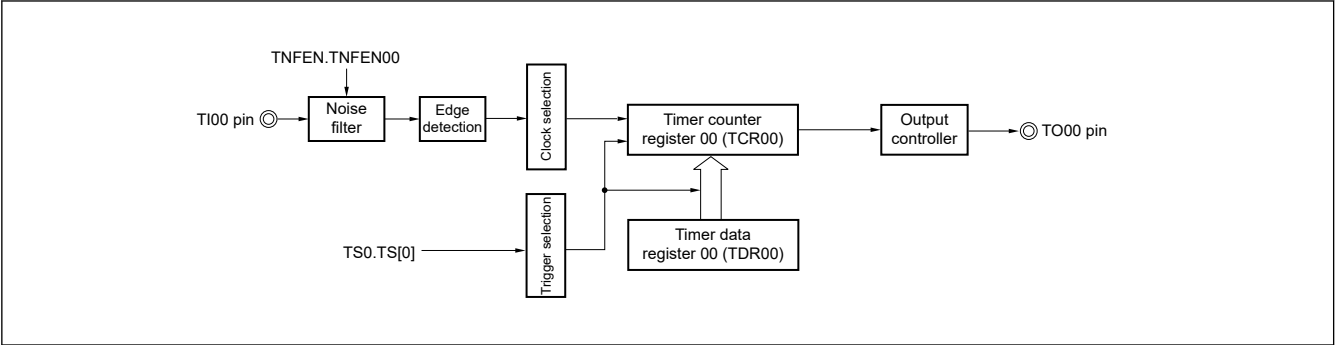


Figure 16.41 Block diagram for operation as a frequency divider

Figure 16.42 shows an example of basic timing during operation as a frequency divider (TMR00.OPIRQ = 1).

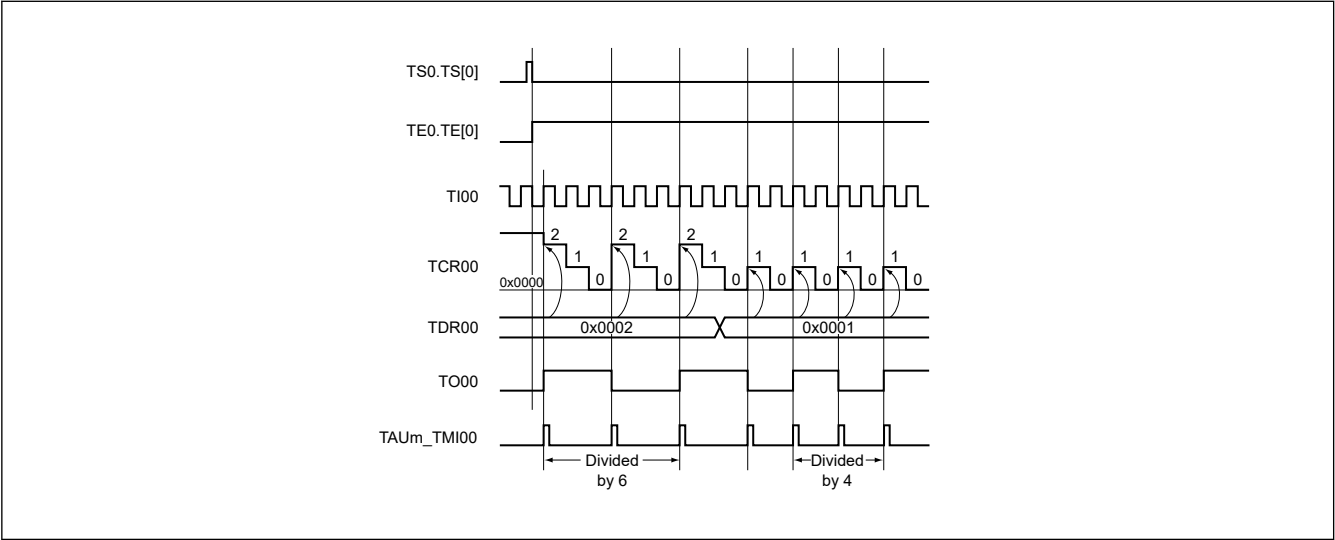


Figure 16.42 Example of basic timing during operation as a frequency divider (TMR00.OPIRQ = 1)

Table 16.25 to Table 16.30 show register settings and procedure for operation when a frequency divider.

Table 16.25 Example of TMR00 settings for operation as a frequency divider (1 of 2)

Bit	Symbol	Set value	Function
0	OPIRQ	1/0	Setting of operation when counting is started 0: Neither generates TAU0_TMI00 nor inverts timer output when counting is started. 1: Generates TAU0_TMI00 and inverts timer output when counting is started.
3:1	MD[2:0]	000b	Operation mode of channel 0 0 0 0: Interval timer mode
5:4	—	00b	Fixed to 0
7:6	CIS[1:0]	00b to 10b	Selection of TImn pin input edge 0 0: Detects falling edge. 0 1: Detects rising edges. 1 0: Detects both edge. Others: Setting prohibited.
10:8	STS[2:0]	000b	Start trigger selection 0 0 0: Selects only software start.
11	—	0	Fixed to 0

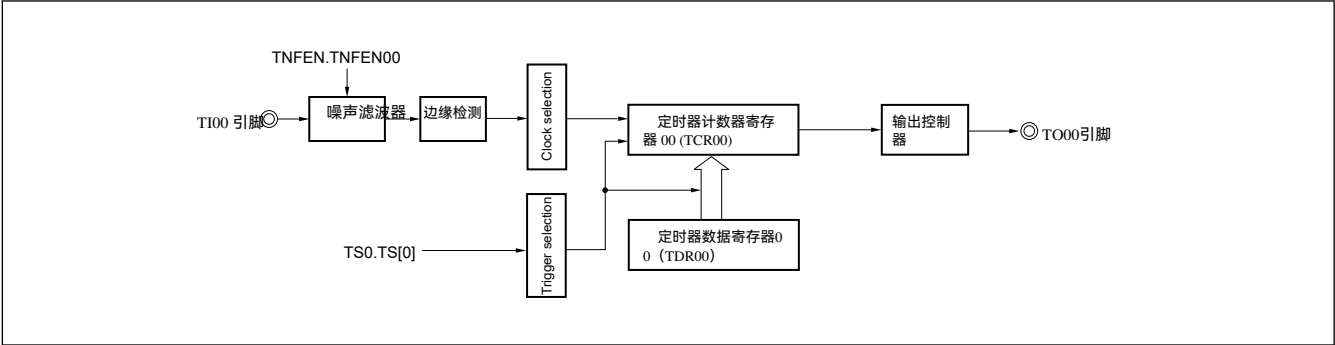


图 16.41 频率分频器工作原理框图

图 16.42 显示了作为分频器(TMR00.OPIRQ = 1)运行时的基本定时示例。

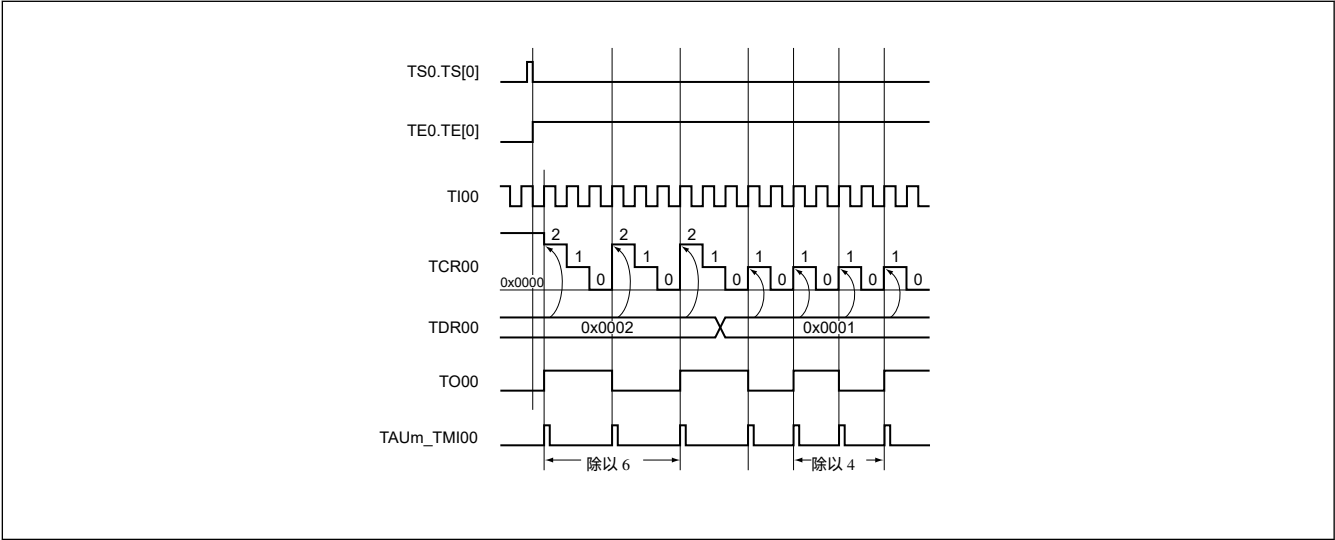


图 16.42 频率分频器工作期间的基本定时示例 (TMR00.OPIRQ= 1)

表 16.25 至表 16.30 显示了分频器运行时的寄存器设置和操作程序。

表 16.25 TMR00 作为分频器运行时的设置示例 (1/2)

少量	象征	设置值	功能
0	OPIRQ	1/0	计数开始时的操作设置 0: 计数开始时既不生成 TAU0_TMI00, 也不反转定时器输出。 1: 计数开始时生成 TAU0_TMI00 并反转定时器输出。
3:1	MD[2:0]	000b	通道 0 的工作模式 0 0 0: 间隔定时器模式
5:4	—	00b	固定为 0
7:6	CIS[1:0]	00b 至 10b	TImn 引脚输入边沿选择 0 0: 检测下降沿。0 1: 检测上升沿。1 0: 检测下降沿和上升沿。 其他: 禁止设置。
10:8	STS[2:0]	000b	启动触发选择 0 0 0: 仅选择软件启动。
11	—	0	固定为 0

Table 16.25 Example of TMR00 settings for operation as a frequency divider (2 of 2)

Bit	Symbol	Set value	Function
12	CCS	1	Count clock selection 1: Selects the TI00 pin input valid edge.
13	—	0	Fixed to 0.
15:14	CKS[1:0]	00b or 10b	Selection of the operating clock (f _{MCK}) 0 0: Selects CK00 as the operating clock for channel 0. 1 0: Selects CK01 as the operating clock for channel 0.

Table 16.26 Example of TO0 settings for operation as a frequency divider

Bit	Symbol	Set value	Function
0	TO[0]	1/0	Timer output of channel 0 0: Outputs 0 from TO00. 1: Outputs 1 from TO00.

Table 16.27 Example of TOE0 settings for operation as a frequency divider

Bit	Symbol	Set value	Function
0	TOE[0]	1/0	Enabling or disabling timer output for channel 0 0: Stops the TO00 output operation by counting operation. 1: Enables the TO00 output operation by counting operation.

Table 16.28 Example of TOL0 settings for operation as a frequency divider

Bit	Symbol	Set value	Function
0	TOL[0]	0	Control of timer output of channel 0 0: Set this bit to 0 when TOM0.TOM[0] = 0 (master channel output mode).

Table 16.29 Example of TOM0 settings for operation as a frequency divider

Bit	Symbol	Set value	Function
0	TOM[0]	0	Control of timer output mode of channel 0 0: Sets master channel output mode.

Table 16.30 Procedure for operations when the frequency divider function is to be used (1 of 2)

	Step	Software operation		Hardware state
TAU default setting		—	→	Power-off state (Clock supply is stopped and writing to each register is disabled.)
	<1>	Sets the MSTPD0 bit of Module Stop Control Register D (MSTPCRD) to 0.		Power-on state. Each channel stops operating. (Clock supply is started and writing to each register is enabled.)
	<2>	Sets timer clock select register 0 (TPS0). Determines clock frequencies of CK00 to CK03.		—

表 16.25 TMR00 作为分频器运行时的设置示例 (2/2)

少量	象征	设置值	功能
12	CCS	1	计数时钟选择 1: 选择 TI00 引脚输入有效边沿。
13	—	0	固定值为0。
15:14	CKS[1:0]	00b 或 10b	选择工作时钟 (f _{MCK}) 0 0: 选择 CK00 作为通道 0 的工作时钟。 1 0: 选择 CK01 作为通道 0 的工作时钟。

表 16.26 用作分频器时 TO0 设置示例

少量	象征	设置值	功能
0	TO[0]	1/0	通道 0 的定时器输出 0: 从 TO00 输出 0。 1: 从 TO00 输出 1。

表 16.27 用作分频器的 TOE0 设置示例

少量	象征	设置值	功能
0	TOE[0]	1/0	启用或禁用通道 0 的定时器输出 0: 通过计数操作停止 TO00 输出操作。 1: 通过计数操作启用 TO00 输出操作。

表 16.28 用作分频器时 TOL0 设置示例

少量	象征	设置值	功能
0	TOL[0]	0	控制通道 0 的定时器输出 0: 当 TOM0.TOM [0] = 0 时, 将此位设置为 0 (主通道输出模式)。

表 16.29 TOM0 设置为分频器的示例

少量	象征	设置值	功能
0	TOM[0]	0	控制通道 0 的定时器输出模式 0: 设置主通道输出模式。

表 16.30 使用分频器功能时的操作步骤 (1/2)

	步	软件操作		硬件状态
TAU 默认设置		—	→	关机状态 (时钟信号停止, 所有寄存器的写入操作均被禁用。)
	<1>	将模块停止控制寄存器 D (MSTPCRD) 的 MSTPD0 位设置为 0。		上电状态。各通道停止工作。 (时钟电源已启动, 并且对每个寄存器的写入操作已启用。)
	<2>	设置定时器时钟选择寄存器 0 (TPS0)。确定 CK00 至 CK03 的时钟频率。		—

Table 16.30 Procedure for operations when the frequency divider function is to be used (2 of 2)

	Step	Software operation		Hardware state
Channel default setting	<3>	Sets the corresponding bit of the TAU noise filter enable register (TNFEN) to 0 (off) or 1 (on). Sets timer mode register 00 (TMR00) (determines operation mode of channel and selects the detection edge). Sets interval (period) value to timer data register 00 (TDR00).		Channel stops operating. (Clock is supplied and some power is consumed.)
	<4>	Sets the TO0.TO[0] bit and determines default level of the TO00 output. Sets the TOE0.TOE[0] bit to 1 and enables operation of TO00. Clears the Pmn Output Data Register and Pmn Direction Register to 0.	→ → →	The TO00 pin goes into Hi-Z output state. The TO00 default setting level is output when the Ppq Direction Register (PDRp) is in output mode. TO00 does not change because channel stops operating. The TO00 pin outputs the TO00 set level.
Operation start	<5>	Sets the TOE0.TOE[0] bit to 1 (only when operation is resumed). Sets the TS0.TS[0] bit to 1. The TS0.TS[0] bit automatically returns to 0 because it is a trigger bit.	→	TE0.TE[0] = 1, and count operation starts. Value of the TDR00 register is loaded to timer counter register 00 (TCR00). TAU0_TMI00 is generated and TO00 performs toggle operation if the MD[0] bit of the TMR00 register is 1.
During operation	<6>	Set value of the TDR00 register can be changed. The TCR00 register can always be read. The TSR00 register is not used. Set values of the TO0 and TOE0 registers can be changed. Set values of the TMR00 register cannot be changed.		Counter (TCR00) counts down. When count value reaches 0x0000, the value of the TDR00 register is loaded to the TCR00 register again, and the count operation is continued. By detecting TCR00 = 0x0000, TAU0_TMI00 is generated and TO00 performs toggle operation. After that, the above operation is repeated.
Operation stop	<7>	The TT0.TT[0] bit is set to 1. The TT0.TT[0] bit automatically returns to 0 because it is a trigger bit.	→	TE0.TE[0] = 0, and count operation stops. The TCR00 register holds count value and stops. The TO00 output is not initialized and retains its current state.
	<8>	The TOE0.TOE[0] bit is cleared to 0 and value is set to the TO0.TO[0] bit. To resume operation, go to step <5>. To terminate the operation, go to step <9>	→	The TO00 pin outputs the TO00 set level.
TAU stop	<9>	To hold the TO0n pin output level Set PSEL[2:0] bits to 000b after the value to be held is set to the Ppq Output Data Register (PODRp). When holding the TO0n pin output level is not necessary Setting not required.	→	The TO0n pin output level is held by port function
	<10>	Sets the MSTPD0 bit of Module Stop Control Register D (MSTPCRD) to 1.	→	This stops supply of the input clock to timer array unit 0. Power-off state.

Note: p = 0 to 9, q = 00 to 15

16.7.4 Operation for Input Pulse Interval Measurement

The count value can be captured at the TI0n valid edge and the interval of the pulse input to TI0n can be measured. In addition, the count value can be captured by using software operation (TS0.TS[n] = 1) as a capture trigger while the TE0.TE[n] bit is set to 1.

The pulse interval can be calculated by the following expression.

TI0n input pulse interval = Period of count clock × ((0x10000 × TSR0n.OVF) + (Captured value of TDR0n + 1))

Note: The TI0n pin input is sampled using the operating clock selected with the CKS[1:0] bits of timer mode register 0n (TMR0n), so an error equivalent to one operation clock occurs.

Timer counter register 0n (TCR0n) operates as an up counter in the capture mode.

表 16.30 使用分频器功能时的操作步骤 (2/2)

	步	软件操作		硬件状态
通道默认设置	<3>	将 TAU 噪声滤波器使能寄存器 (TNFEN) 的相应位设置为 0 (关闭) 或 1 (开启)。 设置定时器模式寄存器 00 (TMR00) (确定通道的操作模式并选择检测边沿)。 将定时器数据寄存器 00 (TDR00) 的间隔 (周期) 值设置到定时器数据寄存器 00 (TDR00)。		频道停止运行。 (时钟信号被提供, 并消耗一定功率。)
	<4>	设置 TO0.TO [0] 位, 并确定 TO00 输出的默认电平。 将 TOE0.TOE [0] 位设置为 1, 使 TO00 能够运行。 将 Pmn 输出数据寄存器和 Pmn 方向寄存器清零。	→ → →	TO00 引脚进入高阻抗输出状态。 当 Ppq 方向寄存器 (PDRp) 处于输出模式时, 输出 T O00 默认设置电平。 TO00 不会改变, 因为通道停止运行。 TO00 引脚输出 TO00 设定电平。
操作开始	<5>	将 TOE0.TOE [0] 位设置为 1 (仅当操作恢复时)。 将 TS0.TS [0] 位设置为 1。 因为 TS0.TS [0] 位会自动返回到 0。 这是一个触发位。	→	TE0.TE[0] = 1, 计数操作开始。 TDR00 寄存器的值被加载到定时器计数器寄存器 00 (TCR00) 中。如果 TMR00 寄存器的 MD[0] 位为 1, 则生成 TAU0_TMI00, 并且 TO00 执行翻转操作。
运行期间	<6>	TDR00 寄存器的设定值可以更改。TCR00 寄存器始终可以读取。 TSR00 寄存器未使用。 TO0 和 TOE0 寄存器的设置值可以更改。 TMR00 寄存器的设定值无法更改。		计数器 (TCR00) 递减计数。当计数达到 0x0000 时, 将 TDR00 寄存器的值重新加载到 TCR00 寄存器, 并继续计数。通过检测 TCR00= 0x0000, 生成 TAU0_TMI00, TO00 执行翻转操作。 之后, 重复上述操作。
停止操作	<7>	TT0.TT [0] 位设置为 1。 TT0.TT [0] 位会自动返回 0, 因为它是一个触发位。	→	TE0.TE[0] = 0, 计数操作停止。 TCR00 寄存器保存计数值并停止工作。TO00 输出未初始化, 保持其当前状态。
	<8>	TOE0.TOE [0] 位被清除为 0, TO0.TO [0] 位的值被设置。 要恢复操作, 请转到步骤 <5>。 要终止操作, 请转到步骤 <9>	→	TO00 引脚输出 TO00 设定电平。
停止!	<9>	要保持 TO0n 引脚输出电平, 在将要保持的值设置到 Pp q 输出数据寄存器 (PODRp) 之后, 将 PSEL[2:0] 位设置为 000b。 保持 TO0n 引脚输出电平时, 无需设置。	→	TO0n 引脚输出电平由端口功能保持
	<10>	将模块停止控制寄存器 D (MSTPCRD) 的 MSTP D0 位设置为 1。	→	这将停止向定时器阵列单元 0 提供输入时钟。 关机状态。

注意: p = 0 至 9, q = 00 至 15

16.7.4 输入脉冲间隔测量操作

可以在 TI0n 的有效边沿捕获计数值, 并测量输入到 TI0n 的脉冲间隔。此外, 当 TE0.TE[n] 位设置为 1 时, 可以使用软件操作 (TS0.TS[n] = 1) 作为捕获触发器来捕获计数值。

脉冲间隔可由以下表达式计算。

TI0n 输入脉冲间隔 = 计数时钟周期 × ((0x10000 × TSR0n.OVF) + (TDR0n 的捕获值 + 1))

注意: TI0n 引脚输入使用定时器模式寄存器 0n (TMR0n) 的 CKS[1:0] 位选择的工作时钟进行采样, 因此会产生相当于一个工作时钟的错误。

定时器计数器寄存器 0n (TCR0n) 在捕获模式下用作递增计数器。

When the channel start trigger bit (TS[n]) of timer channel start register 0 (TS0) is set to 1, the TCR0n register counts up from 0x0000 in synchronization with the count clock.

When the TI0n pin input valid edge is detected, the count value of the TCR0n register is transferred (captured) to timer data register 0n (TDR0n) and, at the same time, the TCR0n register is cleared to 0x0000, and the TAU0_TMI0n is output. If the counter overflows at this time, the OVF bit of timer status register 0n (TSR0n) is set to 1. If the counter does not overflow, the OVF bit is cleared. After that, the above operation is repeated.

As soon as the count value has been captured to the TDR0n register, the OVF bit of the TSR0n register is updated depending on whether the counter overflows during the measurement period. Therefore, the overflow state of the captured value can be checked.

If the counter reaches a full count for two or more periods, it is judged to be an overflow occurrence, and the OVF bit of the TSR0n register is set to 1. However, a normal interval value cannot be measured for the OVF bit, if two or more overflows occur.

Set the STS[2:0] bits of the TMR0n register to 001b to use the valid edges of TI0n as a start trigger and a capture trigger.

Instead of using the TI0n pin input, an input pulse interval can also be measured by using the timer input selected in the TIS0 register or the software operation (TS0.TS[n] = 1) as a start trigger and a capture trigger.

Figure 16.43 shows a block diagram for operation for input pulse interval measurement.

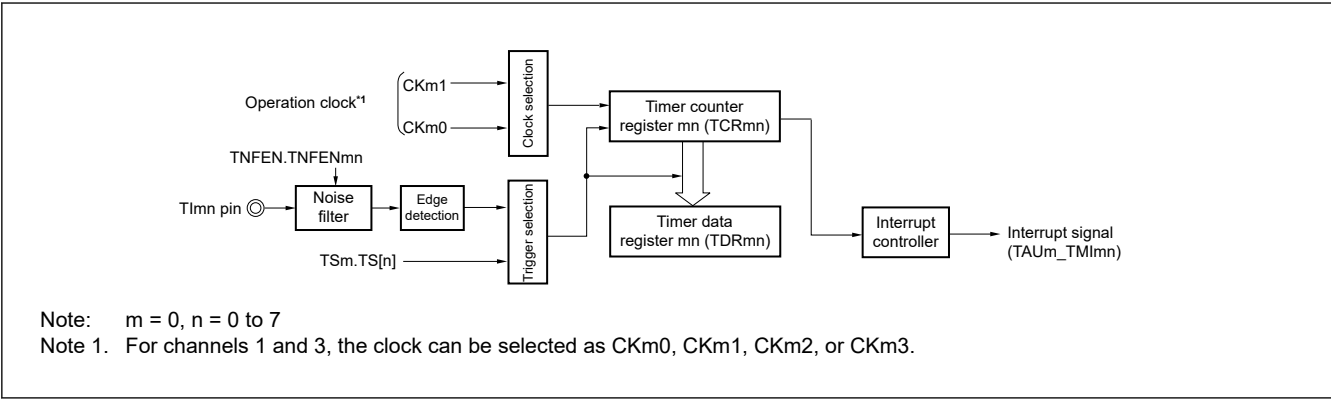


Figure 16.43 Block diagram for operation for input pulse interval measurement

Figure 16.44 shows an example of basic timing during operation for input pulse interval measurement (TMR0n.OPIRQ = 0).

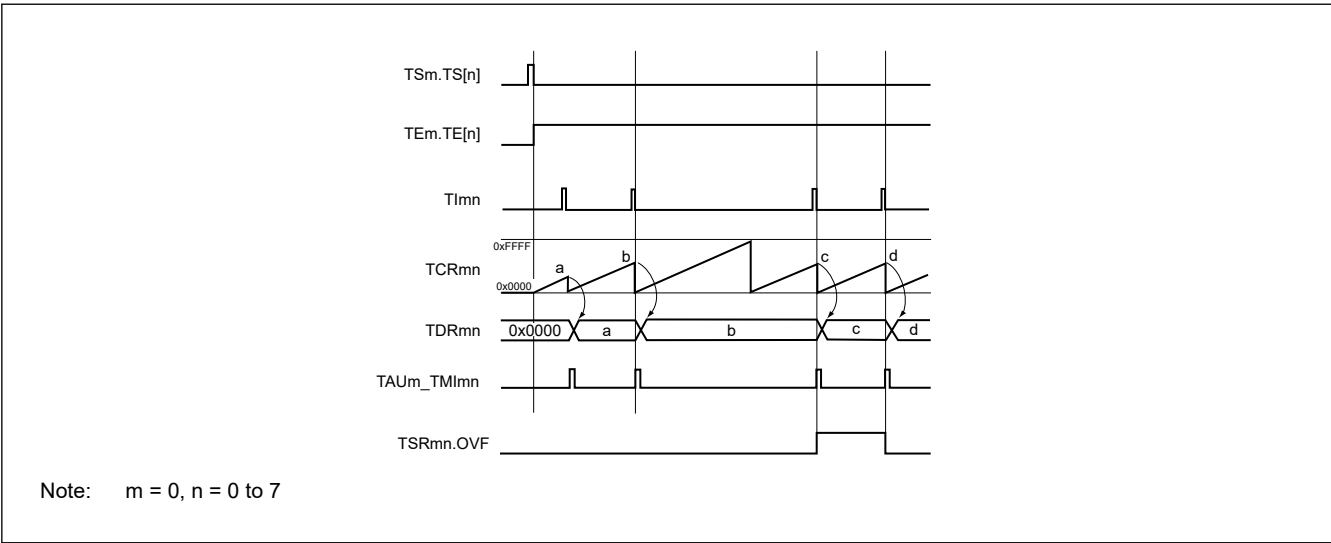


Figure 16.44 Example of basic timing during operation for input pulse interval measurement (TMR0n.OPIRQ = 0)

Table 16.31 to Table 16.36 show register settings and procedure for operation for input pulse interval measurement.

当定时器通道启动寄存器 0 (TS0) 的通道启动触发位(TS[n]) 设置为 1 时, TCR0n 寄存器与计数时钟同步从0x0000 开始向上计数。

当检测到TI0n引脚输入有效边沿时, 将TCR0n寄存器的计数值传输(捕获)到定时器数据寄存器0n (TDR0n), 同时将TCR0n寄存器清零至0x0000, 并输出TAU0_TMI0n。如果此时计数器溢出, 则将定时器状态寄存器0n (TSR0n) 的OVF位置1。如果计数器未溢出, 则清零OVF位。之后, 重复上述操作。

一旦计数值被捕获到 TDR0n 寄存器, TSR0n 寄存器的 OVF 位就会根据测量期间计数器是否溢出而更新。因此, 可以检查捕获值的溢出状态。

如果计数器连续两个或多个周期达到满计数, 则判定为溢出事件, 并将 TSR0n 寄存器的 OVF 位设置为 1。但是, 如果发生两次或多次溢出, 则无法测量 OVF 位的正常间隔值。

将 TMR0n 寄存器的 STS[2:0] 位设置为 001b, 即可使用 TI0n 的有效边沿作为起始触发和捕获触发。除了使用 TI0n 引脚输入外, 还可以使用 TIS0 寄存器中选择的定时器输入或软件操作(TS0.TS[n] = 1)作为起始触发和捕获触发来测量输入脉冲间隔。

图 16.43 显示了输入脉冲间隔测量的操作框图。

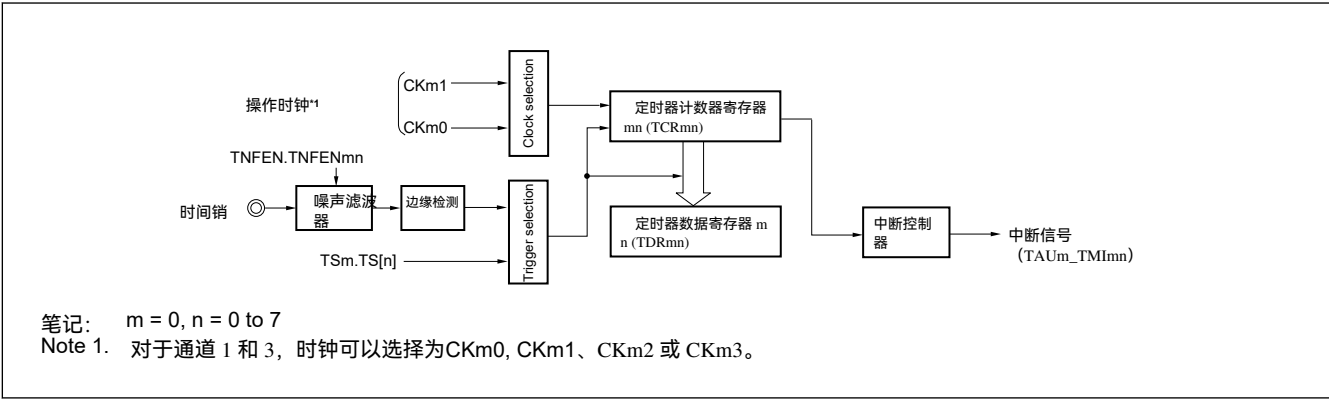


图 16.43 输入脉冲间隔测量操作框图

图 16.44 显示了输入脉冲间隔测量(TMR0n.OPIRQ =0 操作期间的基本定时示例)。

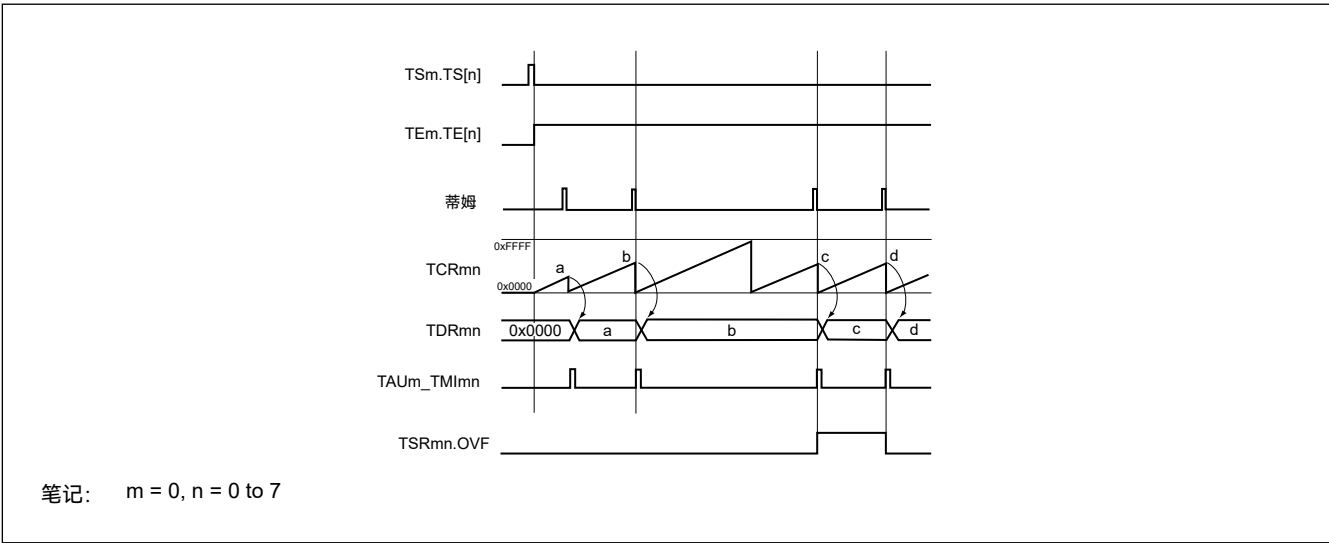


图 16.44 输入脉冲间隔测量操作期间的基本时序示例 (TMR0n.OPIRQ = 0)

表 16.31 至表 16.36 显示了输入脉冲间隔测量的寄存器设置和操作步骤。

Table 16.31 Example of TMR0n settings for operation for input pulse interval measurement

Bit	Symbol	Set value	Function
0	OPIRQ	1/0	Setting of operation when counting is started 0: Neither generates TAU0_TMI0n nor inverts timer output when counting is started. 1: Generates TAU0_TMI0n and inverts timer output when counting is started.
3:1	MD[2:0]	010b	Operation mode of channel n 0 1 0: Capture mode
5:4	—	00b	Fixed to 0
7:6	CIS[1:0]	00b to 10b	Selection of TI0n pin input edge 0 0: Detects falling edge. 0 1: Detects rising edge. 1 0: Detects both edges. Others: Setting prohibited
10:8	STS[2:0]	001b	Start trigger selection 0 0 1: Selects the TI0n pin input valid edge.
11	— (n = 0, 5, 7)	0	Fixed to 0 (channels 0, 5, 7)
	SPLIT (n = 1, 3)	0	Setting of SPLIT bit (channels 1, 3) 0: 16-bit timer mode
	MASTER (n = 2, 4, 6)	0	Setting of MASTER bit (channels 2, 4, 6) 0: Independent channel operation function.
12	CCS	0	Count clock selection 0: Selects operation clock (f _{MCK}).
13	—	0	Fixed to 0.
15:14	CKS[1:0]	00b to 11b	Selection of the operating clock (f _{MCK}) 0 0: Selects CK00 as the operating clock for channel n. 0 1: Selects CK02 as the operating clock (this can only be selected for channels 1 and 3). 1 0: Selects CK01 as the operating clock for channel n. 1 1: Selects CK03 as the operating clock (this can only be selected for channels 1 and 3).

Table 16.32 Example of TO0 settings for operation for input pulse interval measurement

Bit	Symbol	Set value	Function
n	TO[n]	0	Timer output of channel n 0: Outputs 0 from TO0n.

Table 16.33 Example of TOE0 settings for operation for input pulse interval measurement

Bit	Symbol	Set value	Function
n	TOE[n]	0	Enabling or disabling timer output for channel n 0: Stops the TO0n output operation by counting operation.

表 16.31 输入脉冲间隔测量操作的 TMR0n 设置示例

少量	象征	设置值	功能
0	OPIRQ	1/0	计数开始时的操作设置 0: 计数开始时既不生成 TAU0_TMI0n，也不反转定时器输出。 1: 计数开始时生成 TAU0_TMI0n 并反转定时器输出。
3:1	MD[2:0]	010b	通道 n 的工作模式 0 1 0: 捕获模式
5:4	—	00b	固定为 0
7:6	CIS[1:0]	00b 至 10b	TI0n引脚输入边沿选择0 0:检测下降沿。0 1:检测上升沿。1 0:检测上升沿和下降沿。 其他: 禁止设置
10:8	STS[2:0]	001b	开始触发选择0 0 1:选择 TI0n 引脚输入有效边沿。
11	— (n = 0, 5, 7)	0	固定为 0 (通道 0、5、7)
	SPLIT (n = 1, 3)	0	SPLIT 位 (通道 1、3) 设置: 0: 16 位定时器模式
	MASTER (n = 2, 4, 6)	0	MASTER 位 (通道 2、4、6) 设置为 0: 独立通道操作功能。
12	CCS	0	计数时钟选择 0: 选择操作时钟(f _{MCK})。
13	—	0	固定值为0。
15:14	CKS[1:0]	00b 至 11b	选择工作时钟 (f _{MCK}) 0 0: 选择 CK00 作为通道 n 的工作时钟。0 1: 选择 CK02 作为工作时钟 (只能为通道 1 和 3 选择)。 1 0: 选择 CK01 作为通道 n 的工作时钟。 1 1: 选择 CK03 作为工作时钟 (只能为通道 1 和 3 选择)。

表 16.32 输入脉冲间隔测量操作的 TO0 设置示例

少量	象征	设置值	功能
n	TO[n]	0	通道 n 的定时器输出 0: 从 TO0n 输出 0。

表 16.33 输入脉冲间隔测量操作的 TOE0 设置示例

少量	象征	设置值	功能
n	TOE[n]	0	启用或禁用通道 n 的定时器输出 0: 通过计数操作停止 TO0n 输出操作。

Table 16.34 Example of TOL0 settings for operation for input pulse interval measurement

Bit	Symbol	Set value	Function
n	— (n = 0)	0	Fixed to 0 (channels 0)
	TOL[n] (n = 1 to 7)		Control of timer output of channel n (channels 1 to 7) 0: Set this bit to 0 when TOM0.TOM[n] = 0 (master channel output mode).

Table 16.35 Example of TOM0 settings for operation for input pulse interval measurement

Bit	Symbol	Set value	Function
n	— (n = 0)	0	Fixed to 0 (channels 0)
	TOM[n] (n = 1 to 7)		Control of timer output mode of channel n (channels 1 to 7) 0: Sets master channel output mode.

Table 16.36 Procedure for operations when the input pulse interval measurement function is to be used

	Step	Software operation		Hardware state
TAU default setting		—	→	Power-off state (Clock supply is stopped and writing to each register is disabled.)
	<1>	Sets the MSTPD0 bit of Module Stop Control Register D (MSTPCRD) to 0.		Power-on state. Each channel stops operating. (Clock supply is started and writing to each register is enabled.)
	<2>	Sets timer clock select register 0 (TPS0). Determines clock frequencies of CK00 to CK03.		—
Channel default setting	<3>	Sets the corresponding bit of the TAU noise filter enable register (TNFEN) to 0 (off) or 1 (on). Sets timer mode register 0n (TMR0n) (determines operation mode of channel).		Channel stops operating. (Clock is supplied and some power is consumed.)
Operation start	<4>	Sets TS0.TS[n] bit to 1. The TS0.TS[n] bit automatically returns to 0 because it is a trigger bit.	→	TE0.TE[n] = 1 and count operation starts. Timer counter register 0n (TCR0n) is cleared to 0x0000. When the OPIRQ bit of the TMR0n register is 1, TAU0_TMI0n is generated.
During operation	<5>	Set values of only the CIS[1:0] bits of the TMR0n register can be changed. The TDR0n register can always be read. The TCR0n register can always be read. The TSR0n register can always be read. Set values of the TOM0.TOM[n], TOL0.TOL[n], TO0.TO[n], and TOE0.TOE[n] bits cannot be changed.		Counter (TCR0n) counts up from 0x0000. When the valid edge of the TIOn pin input is detected or the TS0.TS[n] bit is set to 1, the count value is transferred (captured) to timer data register 0n (TDR0n). At the same time, the TCR0n register is cleared to 0x0000, and the TAU0_TMI0n signal is generated. If an overflow occurs at this time, the OVF bit of timer status register 0n (TSR0n) is set; if an overflow does not occur, the OVF bit is cleared. After that, the above operation is repeated.
Operation stop	<6>	The TT0.TT[n] bit is set to 1. The TT0.TT[n] bit automatically returns to 0 because it is a trigger bit. To resume operation, go to step <4>. To terminate the operation, go to step <7>.	→	TE0.TE[n] = 0, and count operation stops. The TCR0n register holds count value and stops. The OVF bit of the TSR0n register is also held.
TAU stop	<7>	Sets the MSTPD0 bit of Module Stop Control Register D (MSTPCRD) to 1.	→	This stops supply of the input clock to timer array unit 0. Power-off state.

Note: n = 0 to 7

16.7.5 Operation for Input Signal High- or Low-Level Width Measurement

Note: When using a channel to implement the LIN-bus, set bit 1 (ISC1) of the input switch control register (ISC) to 1. In the following descriptions, read TImn as RXD0.

表 16.34 输入脉冲间隔测量操作的 TOL0 设置示例

少量	象征	设置值	功能
n	— (n = 0)	0	固定为 0（通道 0）
	TOL[n] (n = 1 to 7)		控制通道 n（通道 1 至 7）的定时器输出 0: 当 TOM0.TOM[n] = 0 时（主通道输出模式），将此位设置为 0。

表 16.35 输入脉冲间隔测量操作的 TOM0 设置示例

少量	象征	设置值	功能
n	— (n = 0)	0	固定为 0（通道 0）
	汤姆[n] (n = 1 至 7)		控制通道 n（通道 1 至 7）的定时器输出模式 0: 设置主通道输出模式。

表 16.36 输入脉冲间隔测量功能使用时的操作步骤

	步	软件操作		硬件状态
TAU 默认设置		—	→	关机状态 (时钟信号停止，所有寄存器的写入操作均被禁用。)
	<1>	将模块停止控制寄存器 D (MSTPCRD) 的 MSTPD0 位设置为 0。		上电状态。各通道停止工作。 (时钟电源已启动，并且对每个寄存器的写入操作已启用。)
	<2>	设置定时器时钟选择寄存器 0 (TPS0)。 确定 CK00 到CK03的时钟频率		—
通道默认设置	<3>	将 TAU 噪声滤波器使能寄存器 (TNFEN) 的相应位设置为 0（关闭）或 1（开启）。 设置定时器模式寄存器 0n (TMR0n)（确定通道的操作模式）。		频道停止运行。 (时钟信号被提供，并消耗一定功率。)
操作开始	<4>	将 TS0.TS[n] 位设置为 1。 TS0.TS[n] 位会自动返回 0，因为它是一个触发位。	→	TE0.TE[n] = 1 计数操作开始。 定时器计数器寄存器 0n (TCR0n) 被清零 0x0000。 当 TMR0n 寄存器的 OPIRQ 位为 1 时，生成 TAU0_TMI0n。
运行期间	<5>	只能更改 TMR0n 寄存器的 CIS[1:0] 位的值。 TDR0n寄存器始终可读。TCR0n寄存器始终可读。TSR0n寄存器始终可读。 TOM0.TOM[n]、TOL0.TOL[n]、TO0.TO[n]和 T OE0.TOE[n] 位的值不能更改。		计数器 (TCR0n) 从0x0000. 开始计数。当检测到 T IOn 引脚输入的有效边沿或TS0.TS[n]位被置为 1 时，计数值被传输（捕获）到定时器数据寄存器 0n (TDR0n)。同时，TCR0n 寄存器被清零至0x0000，并生成 TAU0_TMI0n 信号。 如果此时发生溢出，则定时器状态寄存器 0n (TSR0n) 的 OVF 位将被置位；如果没有发生溢出，则 OVF 位将被清除。 之后，重复上述操作。
停止操作	<6>	TT0.TT[n] 位设置为 1。 TT0.TT[n] 位会自动返回 0，因为它是一个触发位。 要恢复操作，请转到步骤<4>。 要终止操作，请转到步骤<7>。	→	TE0.TE[n] = 0，计数操作停止。 TCR0n 寄存器保存计数值并停止。TSR0n 寄存器的 OVF 位也保持保存状态。
停止!	<7>	将模块停止控制寄存器 D (MSTPCRD) 的 MSTPD0 位设置为 1。	→	这将停止向定时器阵列单元 0 提供输入时钟。 关机状态。

注意: n = 0 至 7

16.7.5 输入信号高低电平宽度测量操作

注意: 当使用通道实现 LIN 总线时，将输入开关控制寄存器 (ISC) 的第 1 位 (ISC1) 设置为 1。在以下描述中，将 TImn 读取为 RXD0。

By starting counting at one edge of the TI0n pin input and capturing the number of counts at another edge, the signal width (high-level width or low-level width) of TI0n can be measured. The signal width of TI0n can be calculated by the following expression.

Signal width of TI0n input = Period of count clock × ((0x10000 × TSR0n.OVF) + (Captured value of TDR0n + 1))

Note: The TI0n pin input is sampled using the operating clock selected with the CKS[1:0] bits of timer mode register 0n (TMR0n), so an error equivalent to one operation clock occurs.

Timer counter register 0n (TCR0n) operates as an up counter in the capture & one-count mode.

When the channel start trigger bit (TS[n]) of timer channel start register 0 (TS0) is set to 1, the TE0.TE[n] bit is set to 1 and the TI0n pin start edge detection wait state is set.

When the TI0n pin input start edge (rising edge of the TI0n pin input when the high-level width is to be measured) is detected, the counter counts up from 0x0000 in synchronization with the count clock. When the valid capture edge (falling edge of the TI0n pin input when the high-level width is to be measured) is detected later, the count value is transferred to timer data register 0n (TDR0n) and, at the same time, TAU0_TMI0n is output. If the counter overflows at this time, the OVF bit of timer status register 0n (TSR0n) is set to 1. If the counter does not overflow, the OVF bit is cleared. The TCR0n register stops at the value "value transferred to the TDR0n register + 1", and the TI0n pin start edge detection wait state is set. After that, the above operation is repeated.

As soon as the count value has been captured to the TDR0n register, the OVF bit of the TSR0n register is updated depending on whether the counter overflows during the measurement period. Therefore, the overflow state of the captured value can be checked.

If the counter reaches a full count for two or more periods, it is judged to be an overflow occurrence, and the OVF bit of the TSR0n register is set to 1. However, a normal interval value cannot be measured for the OVF bit, if two or more overflows occur.

Whether the high-level width or low-level width of the TI0n pin is to be measured can be selected by using the CIS[1:0] bits of the TMR0n register.

Because this function is used to measure the signal width of the TI0n pin input, the TS0.TS[n] bit cannot be set to 1 while the TE0.TE[n] bit is 1.

Instead of the TI0n pin input, the timer input selected in the TIS0 register can also be used as a start edge and a capture edge.

CIS[1:0] of TMR0n register = 10b: Low-level width is measured.

CIS[1:0] of TMR0n register = 11b: High-level width is measured.

Figure 16.45 shows a block diagram for operation for Input Signal high- or low-level width measurement.

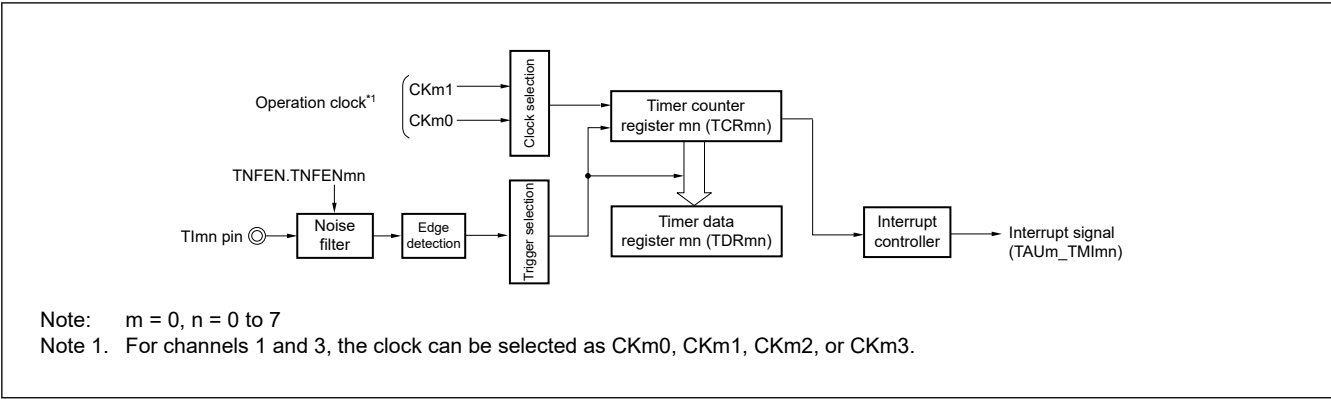


Figure 16.45 Block diagram for operation for input signal high- or low-level width measurement

Figure 16.46 shows an example of basic timing during operation for input signal high- or low-level width measurement.

通过从TI0n引脚输入的一个边沿开始计数，并在另一个边沿捕获计数结果，可以测量TI0n的信号宽度（高电平宽度或低电平宽度）。TI0n的信号宽度可以通过以下表达式计算。

TI0n 输入信号宽度= 计数时钟周期× ((0x10000 × TSR0n.OVF) + (TDR0n 的捕获值+ 1))

注意: TI0n 引脚输入使用定时器模式寄存器 0n (TMR0n) 的 CKS[1:0] 位选择的工作时钟进行采样，因此会产生相当于一个工作时钟的错误。

定时器计数器寄存器 0n (TCR0n) 在捕获& 单计数模式下作为递增计数器运行。

当定时器通道启动寄存器 0 (TS0) 的通道启动触发位(TS[n]) 设置为 1 时，TE0.TE[n] 位设置为 1，并且 TI0n 引脚启动边沿检测等待状态被设置。

当检测到TI0n引脚输入起始边沿（待测量高电平宽度时TI0n引脚输入的上升沿）时，计数器与计数时钟同步从0x0000开始递增计数。随后，当检测到有效捕获边沿（待测量高电平宽度时TI0n引脚输入的下降沿）时，计数值被传输到定时器数据寄存器0n (TDR0n)，同时输出TAU0_TMI0n。如果此时计数器溢出，则定时器状态寄存器0n (TSR0n)的OVF位被置为1。如果计数器未溢出，则OVF位被清零。TCR0n寄存器停止在“传输到TDR0n寄存器的值+ 1”，并且TI0n引脚起始边沿检测等待状态被置位。之后，重复上述操作。

一旦计数值被捕获到 TDR0n 寄存器，TSR0n 寄存器的 OVF 位就会根据测量期间计数器是否溢出而更新。因此，可以检查捕获值的溢出状态。

如果计数器连续两个或多个周期达到满计数，则判定为溢出事件，并将 TSR0n 寄存器的 OVF 位设置为 1。但是，如果发生两次或多次溢出，则无法测量 OVF 位的正常间隔值。

可以通过使用 TMR0n 寄存器的 CIS[1:0] 位来选择要测量 TI0n 引脚的高电平宽度还是低电平宽度。

因为该函数用于测量 TI0n 引脚输入的信号宽度，所以当 TE0.TE[n] 位为 1 时，TS0.TS[n] 位不能设置为 1。

除了 TI0n 引脚输入外，TIS0 寄存器中选择的定时器输入也可以用作起始边沿和捕获边沿。

CIS[1:0] 的 TMR0n 寄存器= 10b: 测量低电平宽度。

CIS[1:0] TMR0n 寄存器= 11b: 测量高电平宽度。

图 16.45 显示了输入信号高电平或低电平宽度测量的操作框图。

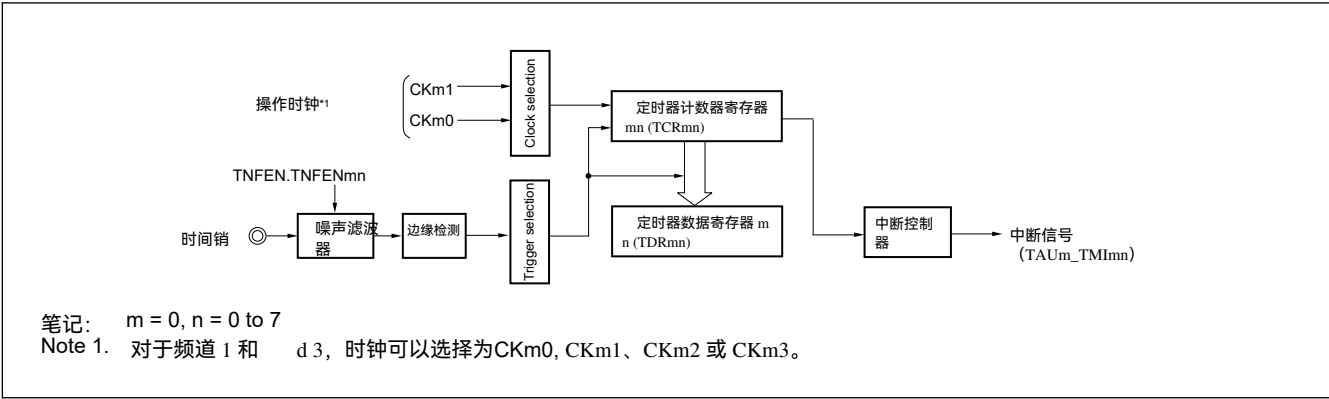


图 16.45 输入信号高电平或低电平宽度测量的操作框图

图 16.46 显示了输入信号高电平或低电平宽度测量操作期间的基本时序示例。

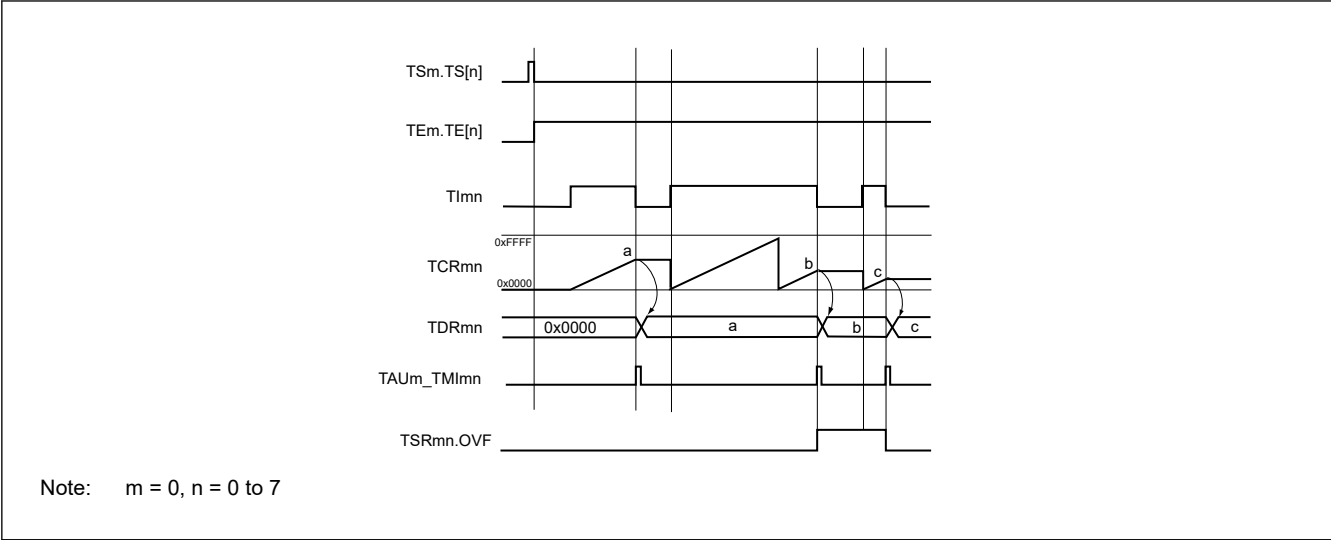


Figure 16.46 Example of basic timing during operation for input signal high- or low-level width measurement

Table 16.37 to Table 16.42 show register settings and procedure for operation for input signal high- or low-level width measurement.

Table 16.37 Example of TMR0n settings for operation for input signal high- or low-level width measurement (1 of 2)

Bit	Symbol	Set value	Function
0	OPIRQ	0	Setting of operation when counting is started 0: Neither generates TAU0_TMI0n nor inverts timer output when counting is started.
3:1	MD[2:0]	110b	Operation mode of channel n 1 1 0: Capture & one-count.
5:4	—	00b	Fixed to 0
7:6	CIS[1:0]	10b to 11b	Selection of TI0n pin input edge 1 0: Both edges (to measure low-level width) 1 1: Both edges (to measure high-level width) Others: Setting prohibited
10:8	STS[2:0]	010b	Start trigger selection 0 1 0: Selects the TI0n pin input valid edges.
11	— (n = 0, 5, 7)	0	Fixed to 0 (channels 0, 5, 7)
	SPLIT (n = 1, 3)	0	Setting of SPLIT bit (channels 1, 3) 0: 16-bit timer mode
	MASTER (n = 2, 4, 6)	0	Setting of MASTER bit (channels 2, 4, 6) 0: Independent channel operation function.
12	CCS	0	Count clock selection 0: Selects operation clock (fMCK).
13	—	0	Fixed to 0.

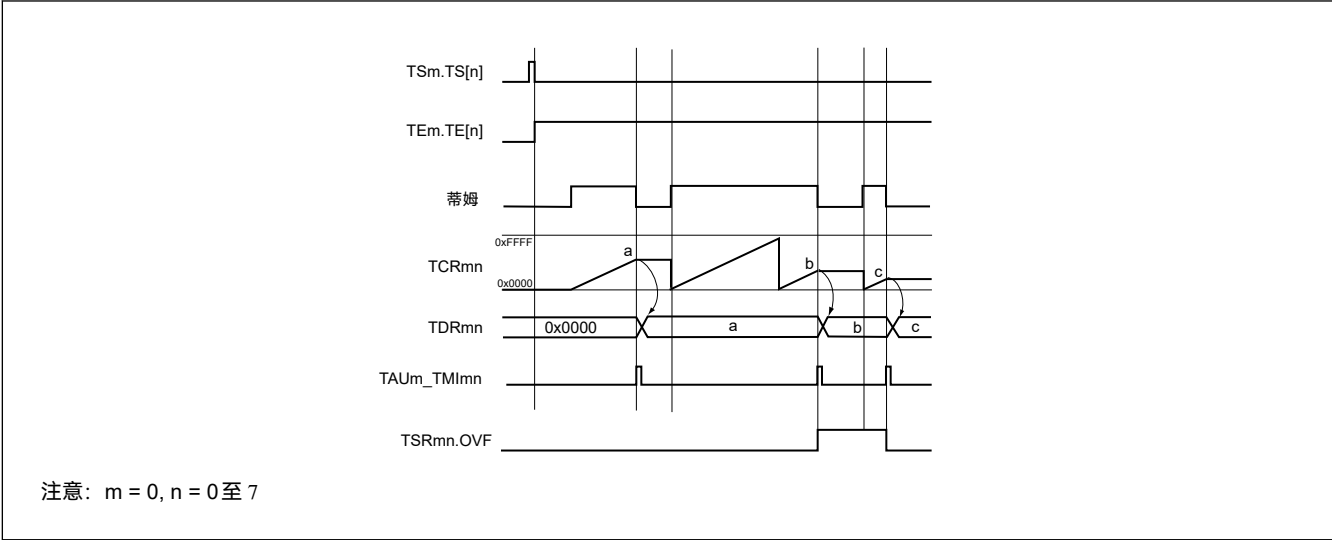


图 16.46 输入信号高电平或低电平宽度测量期间的基本时序示例

表 16.37 至表 16.42 显示了输入信号高电平或低电平宽度测量的寄存器设置和操作步骤。

表 16.37 输入信号高电平或低电平宽度测量的 TMR0n 设置示例 (1/2)

少量	象征	设置值	功能
0	OPIRQ	0	计数开始时的操作设置 0: 计数开始时既不生成 TAU0_TMI0n, 也不反转定时器输出。
3:1	MD[2:0]	110b	通道 n 1 1 0 的操作模式: 捕获和计数。
5:4	—	00b	固定为 0
7:6	CIS[1:0]	10b 至 11b	选择 TI0n 引脚输入边沿 1 0: 两侧边缘 (用于测量低级宽度) 1 1: 两条边 (用于测量高层宽度) 其他: 禁止设置
10:8	STS[2:0]	010b	开始触发选择 0 1 0: 选择 TI0n 引脚输入有效边沿。
11	— (n = 0, 5, 7)	0	固定为 0 (通道 0、5、7)
	SPLIT (n = 1, 3)	0	SPLIT 位 (通道 1、3) 设置: 0: 16 位定时器模式
	MASTER (n = 2, 4, 6)	0	MASTER 位 (通道 2、4、6) 设置为 0: 独立通道操作功能。
12	CCS	0	计数时钟选择 0: 选择操作时钟 (fMCK)。
13	—	0	固定值为 0。

Table 16.37 Example of TMR0n settings for operation for input signal high- or low-level width measurement (2 of 2)

Bit	Symbol	Set value	Function
15:14	CKS[1:0]	00b to 11b	Selection of the operating clock (f_{MCK}) 0 0: Selects CK00 as the operating clock for channel n. 0 1: Selects CK02 as the operating clock (this can only be selected for channels 1 and 3). 1 0: Selects CK01 as the operating clock for channel n. 1 1: Selects CK03 as the operating clock (this can only be selected for channels 1 and 3).

Table 16.38 Example of TO0 settings for operation for input signal high- or low-level width measurement

Bit	Symbol	Set value	Function
n	TO[n]	0	Timer output of channel n 0: Outputs 0 from TO0n.

Table 16.39 Example of TOE0 settings for operation for input signal high- or low-level width measurement

Bit	Symbol	Set value	Function
n	TOE[n]	0	Enabling or disabling timer output for channel n 0: Stops the TO0n output operation by counting operation.

Table 16.40 Example of TOL0 settings for operation for input signal high- or low-level width measurement

Bit	Symbol	Set value	Function
n	— (n = 0)	0	Fixed to 0 (channels 0)
	TOL[n] (n = 1 to 7)		Control of timer output of channel n (channels 1 to 7) 0: Set this bit to 0 when TOM0.TOM[n] = 0 (master channel output mode).

Table 16.41 Example of TOM0 settings for operation for input signal high- or low-level width measurement

Bit	Symbol	Set value	Function
n	— (n = 0)	0	Fixed to 0 (channels 0)
	TOM[n] (n = 1 to 7)		Control of timer output mode of channel n (channels 1 to 7) 0: Sets master channel output mode.

Table 16.42 Procedure for operations when the input signal high- or low-level width measurement function is to be used (1 of 2)

	Step	Software operation	Hardware state
TAU default setting		—	Power-off state (Clock supply is stopped and writing to each register is disabled.)
	<1>	Sets the MSTPD0 bit of Module Stop Control Register D (MSTPCRD) to 0.	→ Power-on state. Each channel stops operating. (Clock supply is started and writing to each register is enabled.)
	<2>	Sets timer clock select register 0 (TPS0). Determines clock frequencies of CK00 to CK03.	—
Channel default setting	<3>	Sets the corresponding bit of the TAU noise filter enable register (TNFEN) to 0 (off) or 1 (on). Sets timer mode register 0n (TMR0n) (determines operation mode of channel). Clears the TOE0.TOE[n] bit to 0 and stops operation of TO0n.	Channel stops operating. (Clock is supplied and some power is consumed.)

表 16.37 输入信号高电平或低电平宽度测量的 TMR0n 设置示例 (2/2)

少量	象征	设置值	功能
15:14	CKS[1:0]	00b to 11b	选择工作时钟 (f) 0 0: 选择 CK00 作为通道 n 的工作时钟。 0 1:选择 CK02 作为工作时钟 (这只能为通道 1 和 3 选择)。 1 0: 选择 CK01 作为通道 n 的工作时钟。 1 1: 选择 CK03 作为工作时钟 (只能为通道 1 和 3 选择)。

表 16.38 输入信号高电平或低电平宽度测量操作的 TO0 设置示例

少量	象征	设置值	功能
n	TO[n]	0	通道 n 的定时器输出 0: 从 TO0n 输出 0。

表 16.39 输入信号高电平或低电平宽度测量操作的 TOE0 设置示例

少量	象征	设置值	功能
n	TOE[n]	0	启用或禁用通道 n 的定时器输出 0: 通过计数操作停止 TO0n 输出操作。

表 16.40 输入信号高电平或低电平宽度测量操作的 TOL0 设置示例

少量	象征	设置值	功能
n	— (n = 0)	0	固定为 0 (通道 0)
	TOL[n] (n = 1 to 7)		控制通道 n (通道 1 至 7) 的定时器输出 0: 当 TOM0.TOM[n] = 0 时 (主通道输出模式)，将此位设置为 0。

表 16.41 输入信号高电平或低电平宽度测量操作的 TOM0 设置示例

少量	象征	设置值	功能
n	— (n = 0)	0	固定为 0 (通道 0)
	汤姆[n] (n = 1 至 7)		控制通道 n (通道 1 至 7) 的定时器输出模式 0: 设置主通道输出模式。

表 16.42 输入信号高电平或低电平宽度测量功能使用时的操作步骤 (1/2)

	步	软件操作	硬件状态
TAU 默认设置		—	关机状态 (时钟信号停止，所有寄存器的写入操作均被禁用。)
	<1>	将模块停止控制寄存器 D (MSTPCRD) 的 MSTPD0 位设置为 0。	→ 上电状态。各通道停止工作。 (时钟电源已启动，并且对每个寄存器的写入操作已启用。)
	<2>	设置定时器时钟选择寄存器 0 (TPS0)。确定 CK00 到CK03的时钟频率	—
通道默认设置	<3>	将 TAU 噪声滤波器使能寄存器 (TNFEN) 的相应位设置为 0 (关闭) 或 1 (开启)。 设置定时器模式寄存器 0n (TMR0n) (确定通道的操作模式)。 将 TOE0.TOE[n] 位清除为 0，并停止 TO0n 的操作。	频道停止运行。 (时钟信号被提供，并消耗一定功率。)

Table 16.42 Procedure for operations when the input signal high- or low-level width measurement function is to be used (2 of 2)

	Step	Software operation		Hardware state
Operation start	<4>	Sets TS0.TS[n] bit to 1. The TS0.TS[n] bit automatically returns to 0 because it is a trigger bit.	→	TE0.TE[n] = 1, and the TI0n pin start edge detection wait state is set.
	<5>	Detects the TI0n pin input count start valid edge.	→	Clears timer counter register 0n (TCR0n) to 0x0000 and starts counting up.
During operation	<6>	Set value of the TDR0n register can always be read. The TCR0n register can always be read. The TSR0n register can always be read. Set values of the TMR0n register, TOM0.TOM[n], TOL0.TOL[n], TO0.TO[n], and TOE0.TOE[n] bits cannot be changed.		When the TI0n pin start edge is detected, the counter (TCR0n) counts up from 0x0000. If a capture edge of the TI0n pin is detected, the count value is transferred to timer data register 0n (TDR0n) and TAU0_TMI0n is generated. If an overflow occurs at this time, the OVF bit of timer status register 0n (TSR0n) is set; if an overflow does not occur, the OVF bit is cleared. The TCR0n register stops the count operation until the next TI0n pin start edge is detected.
Operation stop	<7>	The TT0.TT[n] bit is set to 1. The TT0.TT[n] bit automatically returns to 0 because it is a trigger bit. To resume operation, go to step <4>. To terminate the operation, go to step <8>	→	TE0.TE[n] = 0, and count operation stops. The TCR0n register holds count value and stops. The OVF bit of the TSR0n register is also held.
TAU stop	<8>	Sets the MSTPD0 bit of Module Stop Control Register D (MSTPCRD) to 1.	→	This stops supply of the input clock to timer array unit 0. Power-off state.

Note: n = 0 to 7

16.7.6 Operation as a Delay Counter

It is possible to start counting down when the valid edge of the TI0n pin input is detected (an external event), and then generate TAU0_TMI0n (a timer interrupt) after any specified interval.

It is also possible to start counting down and generate TAU0_TMI0n (timer interrupt) at any interval by setting TS0.TS[n] to 1 by software while TE0.TE[n] = 1.

The interrupt generation period can be calculated by the following expression.

Generation period of TAU0_TMI0n (timer interrupt) = Period of count clock × (Set value of TDR0n + 1)

Timer counter register 0n (TCR0n) operates as a down counter in the one-count mode.

When the channel start trigger bit (TS[n], TSH1, TSH3) of timer channel start register 0 (TS0) is set to 1, the TE0.TE[n], TEH1, and TEH3 bits are set to 1 and the TI0n pin input valid edge detection wait state is set.

Timer counter register 0n (TCR0n) starts operating upon TI0n pin input valid edge detection and loads the value of timer data register 0n (TDR0n). The TCR0n register counts down from the value of the TDR0n register it has loaded, in synchronization with the count clock. When TCR0n = 0x0000, it outputs TAU0_TMI0n and stops counting until the next TI0n pin input valid edge is detected.

The TDR0n register can be rewritten at any time. The new value of the TDR0n register becomes valid from the next period. Instead of using the TI0n pin input, a channel specified for the delay counter function can also use the timer input selected in the TIS0 register or the software operation (TS0.TS[n] = 1) as a start trigger for the function.

Figure 16.47 shows a block diagram for operation as a delay counter.

表 16.42 输入信号高电平或低电平宽度测量功能使用时的操作步骤 (2/2)

	步	软件操作		硬件状态
操作开始	<4>	将 TS0.TS[n] 位设置为 1。 TS0.TS[n] 位会自动返回 0，因为它是一个触发位。	→	TE0.TE[n] = 1，并且 TI0n 引脚起始边沿检测等待状态已设置。
	<5>	检测 TI0n 引脚输入计数开始有效边沿。	→	将定时器计数器寄存器 0n (TCR0n) 清零至 0x0000 并开始向上计数。
运行期间	<6>	TDR0n 寄存器的设定值始终可以读取。 TCR0n 寄存器始终可读取。TSR0n 寄存器始终可读取。 TMR0n 寄存器的设置值，TOM0.TOM[n]，TOL0.TOL[n]，TO0.TO[n] 和 TOE0.TOE[n] 位不能更改。		当检测到 TI0n 引脚的起始边沿时，计数器 (TCR0n) 从 0x0000 开始计数。如果检测到 TI0n 引脚的捕获边沿，则计数值将传输到定时器数据寄存器 0n (TDR0n)，并生成 TAU0_TMI0n。 如果此时发生溢出，则定时器状态寄存器 0n (TSR0n) 的 OVF 位被置位；如果未发生溢出，则 OVF 位被清除。TCR0n 寄存器停止计数操作，直到检测到下一个 TI0n 引脚起始边沿。
停止操作	<7>	TT0.TT[n] 位设置为 1。 TT0.TT[n] 位会自动返回 0，因为它是一个触发位。 要恢复操作，请转到步骤<4>。 要终止操作，请转到步骤<8>	→	TE0.TE[n] = 0，计数操作停止。 TCR0n 寄存器保存计数值并停止。TSR0n 寄存器的 OVF 位也保持保存状态。
停止!	<8>	将模块停止控制寄存器 D (MSTPCRD) 的 MSTPD0 位设置为 1。	→	这将停止向定时器阵列单元 0 提供输入时钟。 关机状态。

注意: n = 0 至 7

16.7.6 作为延迟计数器运行

当检测到 TI0n 引脚输入的有效边沿（外部事件）时，可以开始倒计时，然后在任何指定的时间间隔后生成 TAU0_TMI0n（定时器中断）。

也可以通过软件将 TS0.TS[n] 设置为 1，同时将 TE0.TE[n] = 1 设置为 1，从而开始倒计时并以任意间隔生成 TAU0_TMI0n（定时器中断）。

中断生成周期可由以下表达式计算。

TAU0_TMI0n（定时器中断）的生成周期 = 计数时钟周期 × （TDR0n 的设置值 + 1）

定时器计数器寄存器 0n (TCR0n) 在计数模式下作为递减计数器运行。

当定时器通道启动寄存器 0 (TS0) 的通道启动触发位 (TS[n], TSH1, TSH3) 设置为 1 时，TE0.TE[n], TEH1 和 TEH3 位设置为 1，并且 TI0n 引脚输入有效边沿检测等待状态被设置。

定时器计数器寄存器 0n (TCR0n) 在检测到 TI0n 引脚输入有效边沿时开始工作，并将定时器数据寄存器 0n (TDR0n) 的值加载到寄存器中。TCR0n 寄存器从加载的 TDR0n 寄存器的值开始倒计时，并与计数时钟同步。当 TCR0n = 0x0000 时，它输出 TAU0_TMI0n 并停止计数，直到检测到下一个 TI0n 引脚输入有效边沿。

TDR0n 寄存器可以随时重写。TDR0n 寄存器的新值从下一个周期开始生效。除了使用 TI0n 引脚输入外，为延迟计数器功能指定的通道还可以使用 TIS0 寄存器中选择的定时器输入或软件操作 (TS0.TS[n] = 1) 作为该功能的启动触发信号。

图 16.47 显示了作为延迟计数器的运行框图。

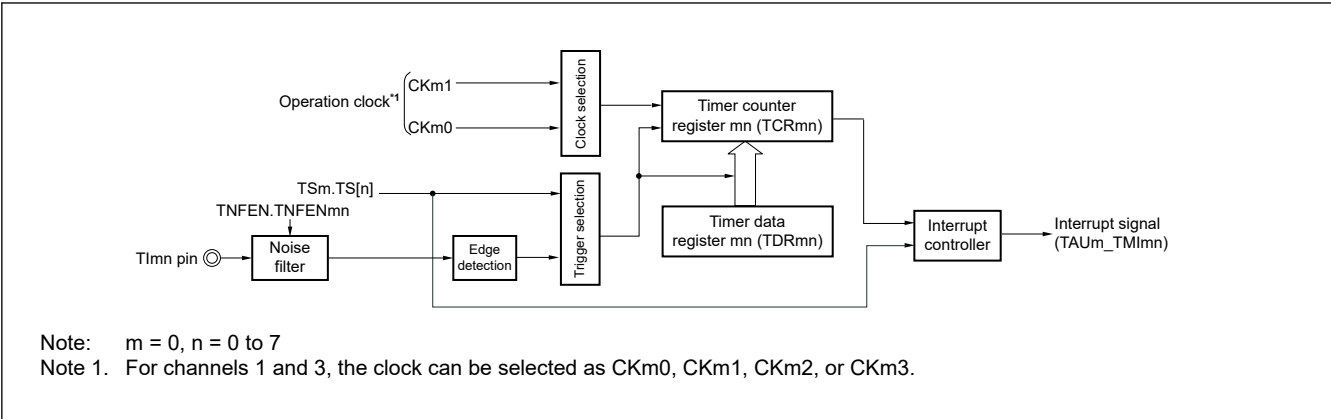


Figure 16.47 Block diagram for operation as a delay counter

Figure 16.48 shows an example of basic timing during operation as a delay counter.

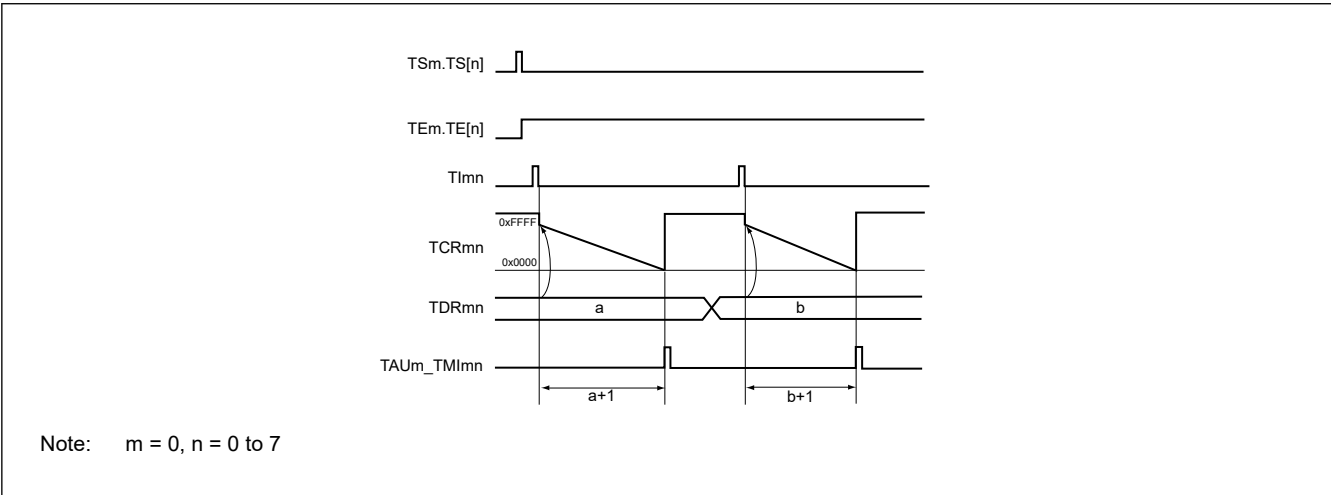


Figure 16.48 Example of basic timing during operation as a delay counter

Table 16.43 to Table 16.48 show register settings and procedure for operation as a delay counter.

Table 16.43 Example of TMR0n settings for operation as a delay counter (1 of 2)

Bit	Symbol	Set value	Function
0	OPIRQ	1/0	Start trigger during operation 0: Trigger input is invalid 1: Trigger input is valid
3:1	MD[2:0]	100b	Operation mode of channel n 1 0 0: One-count mode
5:4	—	00b	Fixed to 0
7:6	CIS[1:0]	00b to 10b	Selection of TI0n pin input edge 0 0: Detects falling edge 0 1: Detects rising edge 1 0: Detects both edges Others: Setting prohibited
10:8	STS[2:0]	001b	Start trigger selection 0 0 1: Selects the TI0n pin input valid edge.

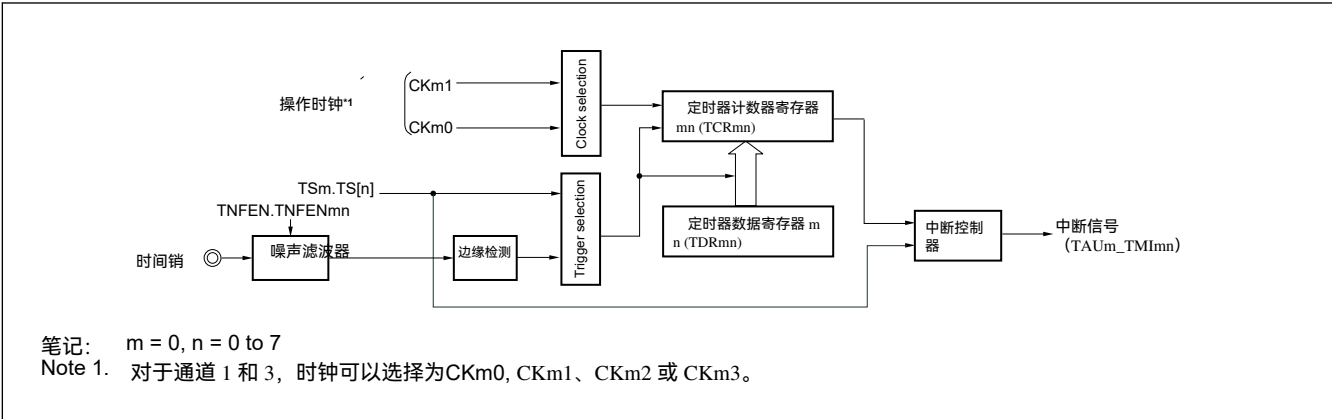


图 16.47 延迟计数器的运行框图

图 16.48 显示了作为延迟计数器运行时的基本计时示例。

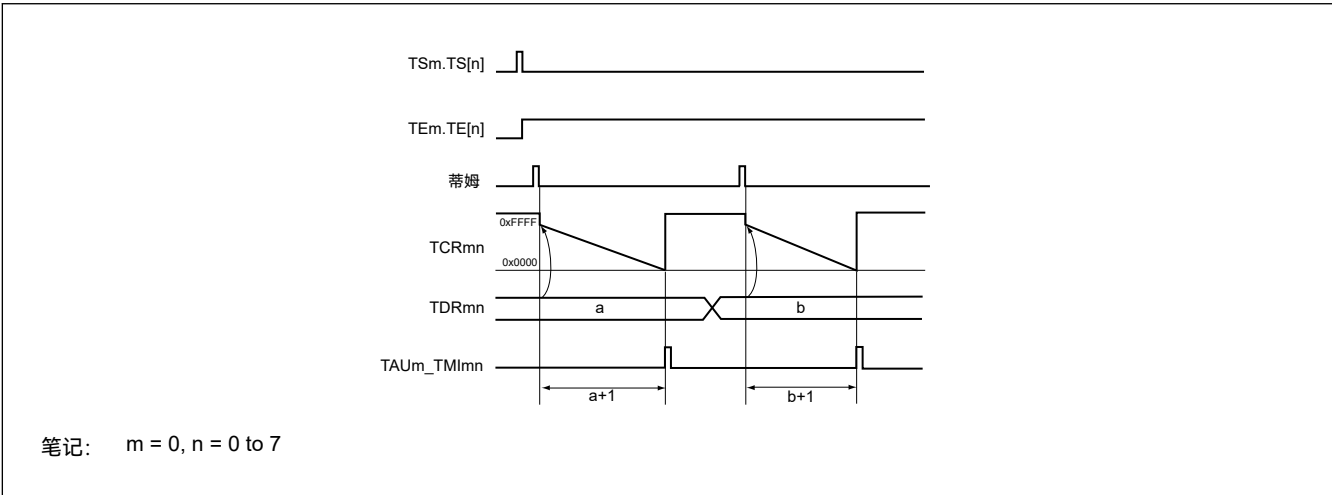


图 16.48 延迟计数器工作期间的基本计时示例

表 16.43 至表 16.48 显示了作为延迟计数器的寄存器设置和操作程序。

表 16.43 TMR0n 设置为延迟计数器的示例 (1/2)

少量	象征	设置值	功能
0	OPIRQ	1/0	操作期间启动触发器 0: 触发输入无效 1: 触发输入有效
3:1	MD[2:0]	100b	通道 n 的工作模式 1 0 0:单计数模式
5:4	—	00b	固定为 0
7:6	CIS[1:0]	00b 至 10b	选择TI0n引脚输入边沿 0 0: 检测下降沿 0 1: 检测上升沿 1 0: 检测两个边缘 其他: 禁止设置
10:8	STS[2:0]	001b	开始触发选择0 0 1:选择 TI0n 引脚输入有效边沿。

Table 16.43 Example of TMR0n settings for operation as a delay counter (2 of 2)

Bit	Symbol	Set value	Function
11	— (n = 0, 5, 7)	0	Fixed to 0 (channels 0, 5, 7)
	SPLIT (n = 1, 3)	1/0	Setting of SPLIT bit (channels 1, 3) 0: 16-bit timer mode 1: 8-bit timer mode
	MASTER (n = 2, 4, 6)	0	Setting of MASTER bit (channels 2, 4, 6) 0: Independent channel operation function.
12	CCS	0	Count clock selection 0: Selects operation clock (f _{MCK}).
13	—	0	Fixed to 0.
15:14	CKS[1:0]	00b to 11b	Selection of the operating clock (f _{MCK}) 0 0: Selects CK00 as the operating clock for channel n 0 1: Selects CK02 as the operating clock (this can only be selected for channels 1 and 3) 1 0: Selects CK01 as the operating clock for channel n 1 1: Selects CK03 as the operating clock (this can only be selected for channels 1 and 3)

Table 16.44 Example of TO0 settings for operation as a delay counter

Bit	Symbol	Set value	Function
n	TO[n]	0	Timer output of channel n 0: Outputs 0 from TO0n

Table 16.45 Example of TOE0 settings for operation as a delay counter

Bit	Symbol	Set value	Function
n	TOE[n]	0	Enabling or disabling timer output for channel n 0: Stops the TO0n output operation by counting operation.

Table 16.46 Example of TOL0 settings for operation as a delay counter

Bit	Symbol	Set value	Function
n	— (n = 0)	0	Fixed to 0 (channels 0)
	TOL[n] (n = 1 to 7)		Control of timer output of channel n (channels 1 to 7) 0: Set this bit to 0 when TOM0.TOM[n] = 0 (master channel output mode)

Table 16.47 Example of TOM0 settings for operation as a delay counter

Bit	Symbol	Set value	Function
n	— (n = 0)	0	Fixed to 0 (channels 0)
	TOM[n] (n = 1 to 7)		Control of timer output mode of channel n (channels 1 to 7) 0: Sets master channel output mode.

表 16.43 TMR0n 设置为延迟计数器的示例 (2/2)

少量	象征	设置值	功能
11	— (n = 0, 5, 7)	0	固定为 0 (通道 0、5、7)
	SPLIT (n = 1, 3)	1/0	SPLIT 位设置 (通道 1、3) : 0: 16 位定时器模式; 1: 8 位定时器模式
	MASTER (n = 2, 4, 6)	0	MASTER 位 (通道 2、4、6) 设置为 0: 独立通道操作功能。
12	CCS	0	计数时钟选择 0: 选择操作时钟(f _{MCK})。
13	—	0	固定值为0。
15:14	CKS[1:0]	00b 至 11b	选择工作时钟 (f _{MCK}) 0 0: 选择 CK00 作为通道 n 的工作时钟 1 1: 选择 CK02 作为工作时钟 (只能为通道 1 和 3 选择) 1 0: 选择 CK01 作为通道 n 的工作时钟 1 1: 选择 CK03 作为工作时钟 (只能为通道 1 和 3 选择)

表 16.44 用作延迟计数器的 TO0 设置示例

少量	象征	设置值	功能
n	TO[n]	0	通道 n 的定时器输出 0: 从 TO0n 输出 0

表 16.45 用作延迟计数器的 TOE0 设置示例

少量	象征	设置值	功能
n	TOE[n]	0	启用或禁用通道 n 的定时器输出 0: 通过计数操作停止 TO0n 输出操作。

表 16.46 延迟计数器操作的 TOL0 设置示例

少量	象征	设置值	功能
n	— (n = 0)	0	固定为 0 (通道 0)
	TOL[n] (n = 1 to 7)		控制通道 n (通道 1 至 7) 的定时器输出 0: 当 TOM0.TOM[n] = 0 时 (主通道输出模式)，将此位设置为 0。

表 16.47 TOM0 设置为延迟计数器的示例

少量	象征	设置值	功能
n	— (n = 0)	0	固定为 0 (通道 0)
	汤姆[n] (n = 1 至 7)		控制通道 n (通道 1 至 7) 的定时器输出模式 0: 设置主通道输出模式。

Table 16.48 Procedure for operations when the delay counter function is to be used

	Step	Software operation		Hardware state
TAU default setting		—		Power-off state (Clock supply is stopped and writing to each register is disabled.)
	<1>	Sets the MSTPD0 bit of Module Stop Control Register D (MSTPCRD) to 0.	→	Power-on state. Each channel stops operating. (Clock supply is started and writing to each register is enabled.)
	<2>	Sets timer clock select register 0 (TPS0). Determines clock frequencies of CK00 to CK03.		—
Channel default setting	<3>	Sets the corresponding bit of the TAU noise filter enable register (TNFEN) to 0 (off) or 1 (on). Sets timer mode register 0n (TMR0n) (determines operation mode of channel). TAU0_TMI0n output delay is set to timer data register 0n (TDR0n). Clears the TOE0.TOE[n] bit to 0 and stops operation of TO0n.		Channel stops operating. (Clock is supplied and some power is consumed.)
Operation start	<4>	Sets TS0.TS[n] bit to 1. The TS0.TS[n] bit automatically returns to 0 because it is a trigger bit.	→	TE0.TE[n] = 1, and the start trigger detection (the valid edge of the TI0n pin input is detected or the TS0.TS[n] bit is set to 1) wait state is set.
	<5>	The counter starts counting down by the next start trigger detection. • Detects the TI0n pin input valid edge. • Sets the TS0.TS[n] bit to 1 by the software.	→	Value of the TDR0n register is loaded to the timer counter register 0n (TCR0n).
During operation	<6>	Set value of the TDR0n register can be changed. The TCR0n register can always be read. The TSR0n register is not used.		The counter (TCR0n) counts down. When the count value of TCR0n reaches 0x0000, the TAU0_TMI0n output is generated, and the count operation stops until the next start trigger detection (the valid edge of the TI0n pin input is detected or the TS0.TS[n] bit is set to 1).
Operation stop	<7>	The TT0.TT[n] bit is set to 1. The TT0.TT[n] bit automatically returns to 0 because it is a trigger bit. To resume operation, go to step <4>. To terminate the operation, go to step <8>	→	TE0.TE[n] = 0, and count operation stops. The TCR0n register holds count value and stops.
TAU stop	<8>	Sets the MSTPD0 bit of Module Stop Control Register D (MSTPCRD) to 1.	→	This stops supply of the input clock to timer array unit 0. Power-off state.

Note: n = 0 to 7

16.8 Simultaneous Channel Operation Function of Timer Array Unit

16.8.1 Operation for the One-shot Pulse Output Function

By using two channels as a set, a one-shot pulse having any delay pulse width can be generated from the signal input to the TI0n pin.

The delay time and pulse width can be calculated by the following expressions.

Delay time = {Set value of TDR0n (master) + 2} × Count clock period

Pulse width = {Set value of TDR0p (slave)} × Count clock period

The master channel operates in the one-count mode and counts the delays. Timer counter register 0n (TCR0n) of the master channel starts operating upon start trigger detection and loads the value of timer data register 0n (TDR0n).

The TCR0n register counts down from the value of the TDR0n register it has loaded, in synchronization with the count clock. When TCR0n = 0x0000, it outputs TAU0_TMI0n and stops counting until the next start trigger is detected.

The slave channel operates in the one-count mode and counts the pulse width. The TCR0p register of the slave channel starts operation using TAU0_TMI0n of the master channel as a start trigger, and loads the value of the TDR0p register. The TCR0p register counts down from the value of the TDR0p register it has loaded, in synchronization with the count value. When count value = 0x0000, it outputs TAU0_TMI0p and stops counting until the next start trigger (TAU0_TMI0n of the

表 16.48 使用延迟计数器功能时的操作步骤

	步	软件操作		硬件状态
TAU 默认设置		—		关机状态 (时钟信号停止，所有寄存器的写入操作均被禁用。)
	<1>	将模块停止控制寄存器 D (MSTPCRD) 的 MSTPD0 位设置为 0。	→	上电状态。各通道停止工作。 (时钟电源已启动，并且对每个寄存器的写入操作已启用。)
	<2>	设置定时器时钟选择寄存器 0 (TPS0)。 确定 CK00 至 CK03 的时钟频率。		—
通道默认设置	<3>	将 TAU 噪声滤波器使能寄存器 (TNFEN) 的相应位设置为 0 (关闭) 或 1 (开启)。 设置定时器模式寄存器 0n (TMR0n) (确定通道的操作模式)。 TAU0_TMI0n 输出延迟设置为定时器数据寄存器 0n (TDR0n)。 将 TOE0.TOE[n] 位清除为 0，并停止 TO0n 的操作。		频道停止运行。 (时钟信号被提供，并消耗一定功率。)
操作开始	<4>	将 TS0.TS[n] 位设置为 1。 TS0.TS[n] 位会自动返回 0，因为它是一个触发位。	→	TE0.TE[n] = 1，并且启动触发检测 (检测到 TI0n 引脚输入的有效边沿或 TS0.TS[n] 位设置为 1) 等待状态已设置。
	<5>	计数器从下一次触发检测开始倒计时。 ● 检测 TI0n 引脚输入有效边沿。 ● 通过软件将 TS0.TS[n] 位设置为 1。	→	TDR0n 寄存器的值加载到定时器计数器寄存器 0n (TCR0n) 中。
运行期间	<6>	TDR0n寄存器的设定值可以更改。TCR0n寄存器始终可以读取。 TSR0n 寄存器未使用。		计数器 (TCR0n) 递减计数。当 TCR0n 的计数值达到0x0000时，生成 TAU0_TMI0n 输出，计数操作停止，直到检测到下一个起始触发信号 (检测到 TI0n 引脚输入的有效边沿或 TS0.TS[n] 位设置为 1) 。
停止操作	<7>	TT0.TT[n] 位设置为 1。 TT0.TT[n] 位会自动返回 0，因为它是一个触发位。 要恢复操作，请转到步骤<4>。 要终止操作，请转到步骤<8>	→	TE0.TE[n] = 0，计数操作停止。 TCR0n寄存器保存计数值并停止。
停止!	<8>	将模块停止控制寄存器 D (MSTPCRD) 的 MSTPD0 位设置为 1。	→	这将停止向定时器阵列单元 0 提供输入时钟。 关机状态。

注意: n = 0 至 7

16.8 定时器阵列单元的同步通道操作功能

16.8.1 单次脉冲输出功能的操作

通过将两个通道组合在一起，可以从输入到 TI0n 引脚的信号生成具有任意延迟脉冲宽度的单次脉冲。

延迟时间和脉冲宽度可以通过以下表达式计算。

延迟时间= {设置 TDR0n (主控) 的值+ 2} × 计数时钟周期

脉冲宽度= {设置 TDR0p (从设备) 的值} × 计数时钟周期

主通道以单次计数模式运行，并对延迟进行计数。主通道的定时器计数器寄存器 0n (TCR0n) 在检测到启动触发时开始工作，并将定时器数据寄存器 0n (TDR0n) 的值加载到该寄存器中。

TCR0n 寄存器从其加载的 TDR0n 寄存器的值开始倒数，并与计数时钟同步。当TCR0n = 0x0000时，它输出 TAU0_TMI0n 并停止计数，直到检测到下一个起始触发信号。

从通道以单次计数模式运行，并对脉冲宽度进行计数。从通道的TCR0p 寄存器以主通道的TAU0_TMI0n作为启动触发信号开始工作，并加载TDR0p寄存器的值。TCR0p寄存器从其加载的TDR0p寄存器的值开始计数，并与计数值同步。当计数值为= 0x0000时，它输出TAU0_TMI0p并停止计数，直到检测到下一个启动触发信号（主通道的TAU0_TMI0n）。

master channel) is detected. The output level of TO0p becomes active one count clock after generation of TAU0_TMI0n from the master channel, and inactive when TCR0p = 0x0000.

Instead of using the TI0n pin input, a one-shot pulse can also be output using the software operation (TS0.TS[n] = 1) as a start trigger.

Note: The timing of loading of timer data register 0n (TDR0n) of the master channel is different from that of the TDR0p register of the slave channel. If the TDR0n and TDR0p registers are rewritten during operation, therefore, an illegal waveform is output. Rewrite the TDR0n register after TAU0_TMI0n is generated and the TDR0p register after TAU0_TMI0p is generated.

Note: n = 0, 2, 4, 6 (master channel number)
n < p ≤ 7 (Slave channel number)

Figure 16.49 shows a block diagram for operation for the one-shot pulse output function.

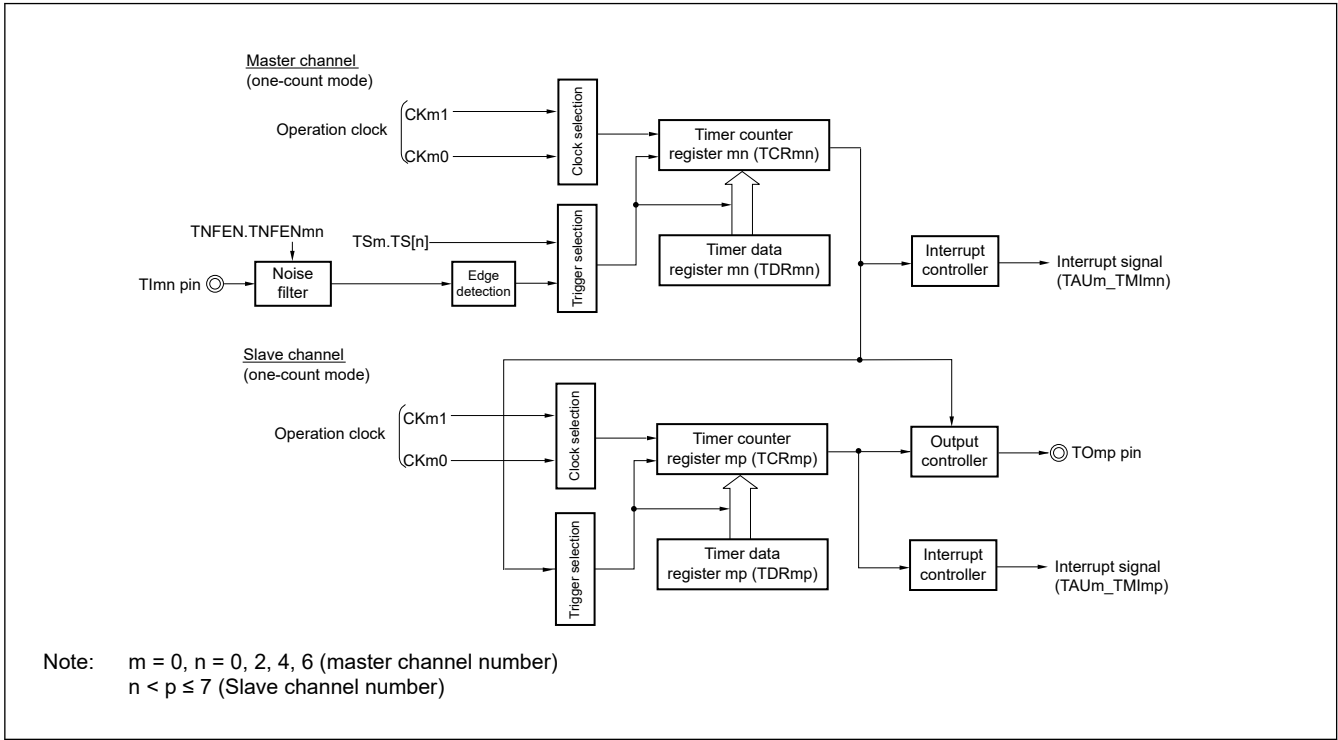


Figure 16.49 Block diagram for operation for the one-shot pulse output function

Figure 16.50 shows an example of basic timing during operation for the one-shot pulse output function.

TO0p的输出电平在主通道生成 TAU0_TMI0n 后经过一个时钟周期变为有效，而当 TCR0p = 0x0000出现时则变为无效。

除了使用 TI0n 引脚输入外，还可以使用软件操作(TS0.TS[n] = 1) 作为启动触发来输出单次脉冲。

注意：主通道定时器数据寄存器 0n (TDR0n) 的加载时序与从通道的TDR0p寄存器的加载时序不同。因此，如果在运行期间重写 TDR0n 和 TDR0p 寄存器，则会输出非法波形。请在生成 TAU0_TMI0n 后重写 TDR0n 寄存器，并在生成 TAU0_TMI0p 后重写 TDR0p 寄存器。

注: n = 0, 2, 4, 6 (主通道号) n < p ≤ 7 (从通道号)

图 16.49 显示了单次脉冲输出功能的运行框图。

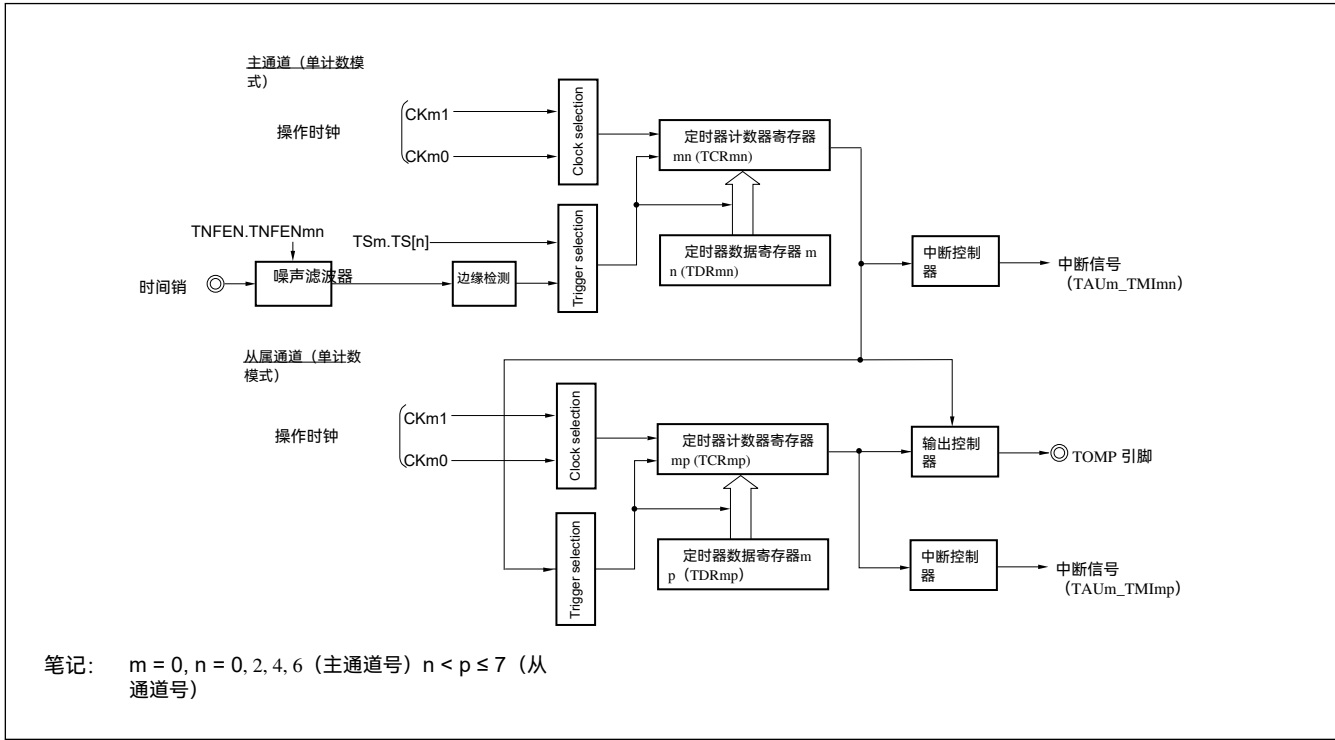


图 16.49 单次脉冲输出函数的运行框图

图 16.50 显示了单次脉冲输出功能运行期间的基本时序示例。

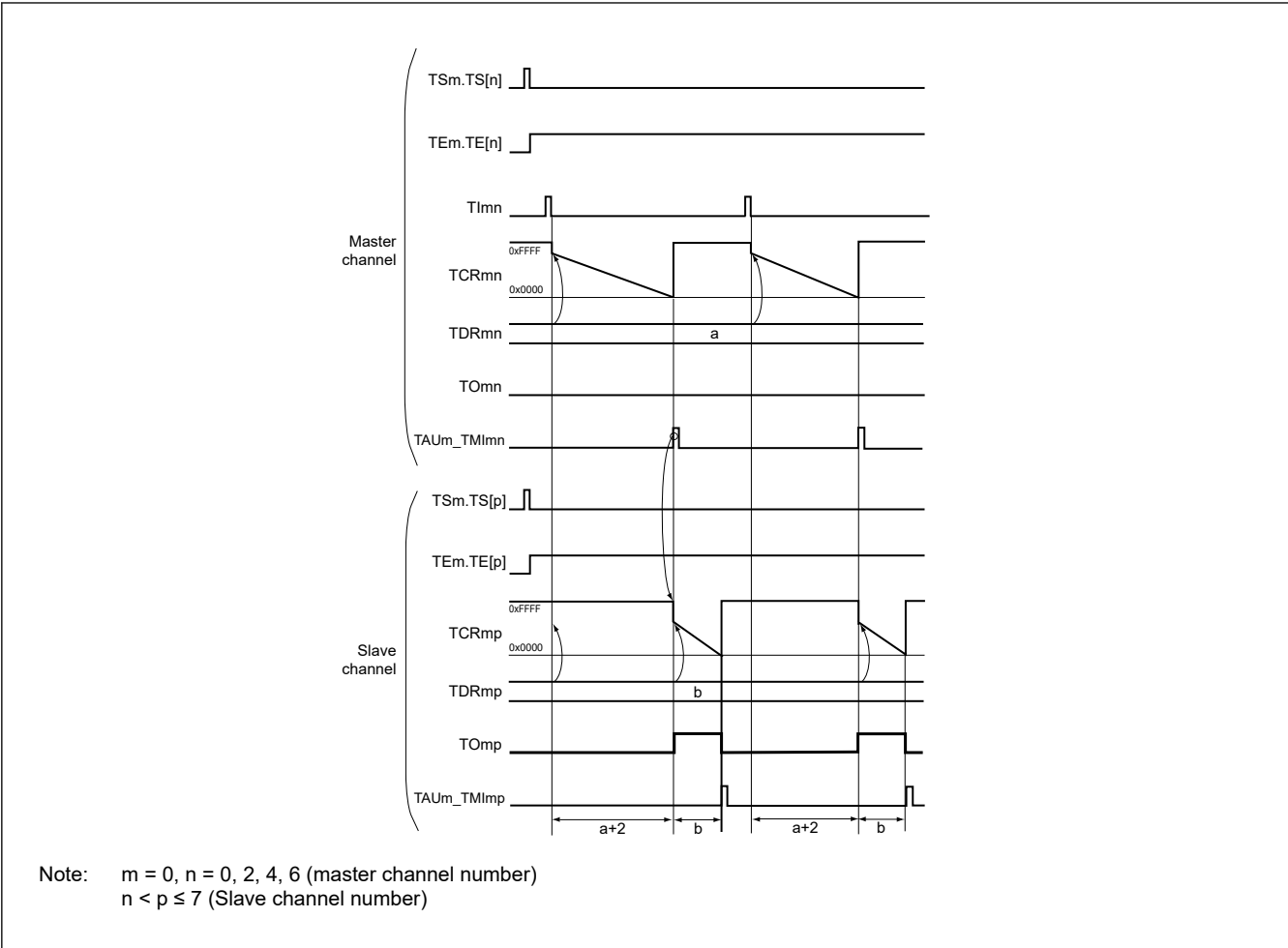


Figure 16.50 Example of basic timing during operation for the one-shot pulse output function

Table 16.49 to Table 16.53 show register settings for the master channel when the one-shot pulse output function is to be used.

Table 16.49 Example of TMR0n settings for the master channel when the one-shot pulse output function is to be used (1 of 2)

Bit	Symbol	Set value	Function
0	OPIRQ	0	Start trigger during operation 0: Trigger input is invalid
3:1	MD[2:0]	100b	Operation mode of channel n 1 0 0: One-count mode
5:4	—	00b	Fixed to 0
7:6	CIS[1:0]	00b to 10b	Selection of TI0n pin input edge 0 0: Detects falling edge 0 1: Detects rising edge 1 0: Detects both edges Others: Setting prohibited
10:8	STS[2:0]	001b	Start trigger selection 0 0 1: Selects the TI0n pin input valid edge

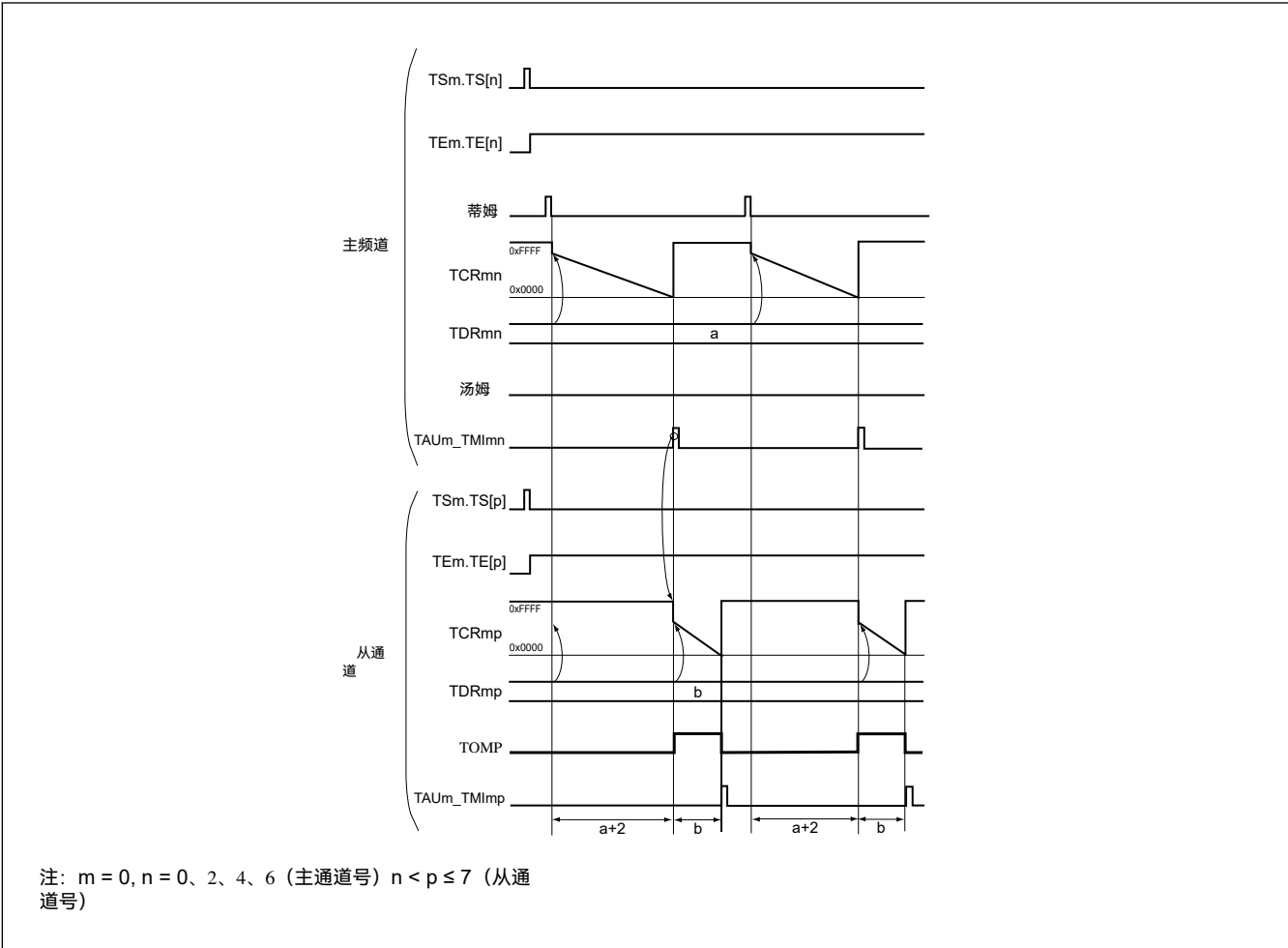


图 16.50 单脉冲输出函数运行期间的基本时序示例

表 16.49 至表 16.53 显示了使用单次脉冲输出功能时主通道的寄存器设置。

表 16.49 当使用单次脉冲输出功能时，主通道的 TMR0n 设置示例 (1/2)

少量	象征	设置值	功能
0	OPIRQ	0	操作期间启动触发器 0: 触发输入无效
3:1	MD[2:0]	100b	通道 n 的工作模式 1 0 0: 单计数模式
5:4	—	00b	固定为 0
7:6	CIS[1:0]	00b 至 10b	选择 TI0n 引脚输入边沿 0 0: 检测下降沿 0 1: 检测上升沿 1 0: 检测两个边缘 其他: 禁止设置
10:8	STS[2:0]	001b	开始触发选择 0 0 1: 选择 TI0n 引脚输入有效边沿

Table 16.49 Example of TMR0n settings for the master channel when the one-shot pulse output function is to be used (2 of 2)

Bit	Symbol	Set value	Function
11	— (n = 0)	0	Fixed to 0 (channels 0)
	MASTER (n = 2, 4, 6)	1	Setting of MASTER bit (channels 2, 4, 6) 1: Master channel
12	CCS	0	Count clock selection 0: Selects operation clock (f _{MCK}).
13	—	0	Fixed to 0.
15:14	CKS[1:0]	00b or 10b	Selection of the operating clock (f _{MCK}) 0 0: Selects CK00 as the operating clock for channel n 1 0: Selects CK01 as the operating clock for channel n

Table 16.50 Example of TO0 settings for the master channel when the one-shot pulse output function is to be used

Bit	Symbol	Set value	Function
n	TO[n]	0	Timer output of channel n 0: Outputs 0 from TO0n

Table 16.51 Example of TOE0 settings for the master channel when the one-shot pulse output function is to be used

Bit	Symbol	Set value	Function
n	TOE[n]	0	Enabling or disabling timer output for channel n 0: Stops the TO0n output operation by counting operation.

Table 16.52 Example of TOL0 settings for the master channel when the one-shot pulse output function is to be used

Bit	Symbol	Set value	Function
n	— (n = 0)	0	Fixed to 0 (channels 0)
	TOL[n] (n = 2, 4, 6)		Control of timer output of channel n (channels 2, 4, 6) 0: Set this bit to 0 when TOM0.TOM[n] = 0 (master channel output mode)

Table 16.53 Example of TOM0 settings for the master channel when the one-shot pulse output function is to be used

Bit	Symbol	Set value	Function
n	— (n = 0)	0	Fixed to 0 (channels 0)
	TOM[n] (n = 2, 4, 6)		Control of timer output mode of channel n (channels 2, 4, 6) 0: Sets master channel output mode

Table 16.54 to Table 16.58 show register settings for the slave channel when the one-shot pulse output function is to be used.

Table 16.54 Example of TMR0p settings for the slave channel when the one-shot pulse output function is to be used (1 of 2)

Bit	Symbol	Set value	Function
0	OPIRQ	0	Start trigger during operation 0: Trigger input is invalid.
3:1	MD[2:0]	100b	Operation mode of channel p 1 0 0: One-count mode

表 16.49 当使用单次脉冲输出功能时，主通道的 TMR0n 设置示例 (2/2)

少量	象征	设置值	功能
11	— (n = 0)	0	固定为 0 (通道 0)
	MASTER (n = 2, 4, 6)	1	主控位设置 (通道 2、4、6) 1: 主通道
12	CCS	0	计数时钟选择 0: 选择操作时钟(f _{MCK})。
13	—	0	固定值为0。
15:14	CKS[1:0]	00b 或 10b	选择工作时钟 (f _{MCK}) 0 0: 选择 CK00 作为通道 n 的工作时钟 1 0: 选择 CK01 作为通道 n 的工作时钟

表 16.50 使用单次脉冲输出功能时，主通道 TO0 设置示例

少量	象征	设置值	功能
n	TO[n]	0	通道 n 的定时器输出 0: 从 TO0n 输出 0

表 16.51 使用单次脉冲输出功能时，主通道的 TOE0 设置示例

少量	象征	设置值	功能
n	TOE[n]	0	启用或禁用通道 n 的定时器输出 0: 通过计数操作停止 TO0n 输出操作。

表 16.52 当使用单次脉冲输出功能时，主通道的 TOL0 设置示例

少量	象征	设置值	功能
n	— (n = 0)	0	固定为 0 (通道 0)
	TOL[n] (n = 2, 4, 6)		控制通道 n (通道 2、4、6) 的定时器输出 0: 当 TOM0.TOM[n] = 0 时 (主通道输出模式)，将此位设置为 0。

表 16.53 使用单次脉冲输出功能时，主通道 TOM0 设置示例

少量	象征	设置值	功能
n	— (n = 0)	0	固定为 0 (通道 0)
	TOM[n] (n = 2, 4, 6)		控制通道 n (通道 2、4、6) 的定时器输出模式 0: 设置主通道输出模式

表 16.54 至表 16.58 显示了使用单次脉冲输出功能时从通道的寄存器设置。

表 16.54 当使用单脉冲输出功能时，从通道的 TMR0p 设置示例 (1/2)

少量	象征	设置值	功能
0	OPIRQ	0	操作 0 期间触发启动: 触发输入无效。
3:1	MD[2:0]	100b	通道 p 的工作模式 1 0 0: 单次计数模式

Table 16.54 Example of TMR0p settings for the slave channel when the one-shot pulse output function is to be used (2 of 2)

Bit	Symbol	Set value	Function
5:4	—	00b	Fixed to 0
7:6	CIS[1:0]	00b	Selection of TI0p pin input edge 0 0: Set to 00b because the TI0p input pin is not to be used
10:8	STS[2:0]	100b	Start trigger selection 1 0 0: Selects TAU0_TMI0n of master channel
11	— (p = 5, 7)	0	Fixed to 0 (channels 5, 7)
	SPLIT (p = 1, 3)	0	Setting of SPLIT bit (channels 1, 3) 0: 16-bit timer mode
	MASTER (p = 2, 4, 6)	0	Setting of MASTER bit (channels 2, 4, 6) 0: Slave channel operation function
12	CCS	0	Count clock selection 0: Selects operation clock (f _{MCK})
13	—	0	Fixed to 0.
15:14	CKS[1:0]	00b or 10b	Selection of the operating clock (f _{MCK}) (Make the same setting as master channel.) 0 0: Selects CK00 as the operating clock for channel. 1 0: Selects CK01 as the operating clock for channel p

Table 16.55 Example of TO0 settings for the slave channel when the one-shot pulse output function is to be used

Bit	Symbol	Set value	Function
p	TO[p]	1/0	Timer output of channel p 0: Outputs 0 from TO0p 1: Outputs 1 from TO0p

Table 16.56 Example of TOE0 settings for the slave channel when the one-shot pulse output function is to be used

Bit	Symbol	Set value	Function
p	TOE[p]	1/0	Enabling or disabling timer output for channel p 0: Stops the TO0p output operation by counting operation. 1: Enables the TO0p output operation by counting operation.

Table 16.57 Example of TOL0 settings for the slave channel when the one-shot pulse output function is to be used

Bit	Symbol	Set value	Function
p	TOL[p]	1/0	Control of timer output of channel p 0: Positive logic output (active-high) 1: Negative logic output (active-low)

Table 16.58 Example of TOM0 settings for the slave channel when the one-shot pulse output function is to be used

Bit	Symbol	Set value	Function
p	TOM[p]	1	Control of timer output mode of channel p (channels 1 to 7) 1: Sets the slave channel output mode.

Table 16.59 show procedure for operations when the one-shot pulse output function is to be used.

表 16.54 当使用单脉冲输出功能时，从通道的 TMR0p 设置示例 (2/2)

少量	象征	设置值	功能
5:4	—	00b	固定为 0
7:6	CIS[1:0]	00b	选择 TI0p 引脚输入边沿 0 0: 设置为 00b，因为 TI0p 输入引脚不使用。
10:8	STS[2:0]	100b	开始触发选择 1 0 0:选择主通道的 TAU0_TMI0n
11	— (p = 5, 7)	0	固定为 0 (通道 5、7)
	SPLIT (p = 1, 3)	0	SPLIT 位 (通道 1、3) 设置: 0: 1 6 位定时器模式
	MASTER (p = 2, 4, 6)	0	主控位设置 (通道 2、4、6) 0: 从通道操作功能
12	CCS	0	计数时钟选择 0: 选择操作时钟 (f _{MCK})
13	—	0	固定值为0。
15:14	CKS[1:0]	00b 或 10b	选择工作时钟(f _{MCK}) (与主通道设置相同。) 0 0: 选择 CK00 作为通道的工作时钟。 1 0: 选择 CK01 作为通道 p 的工作时钟

表 16.55 当使用单次脉冲输出功能时，从通道的 TO0 设置示例

少量	象征	设置值	功能
p	TO[p]	1/0	通道 p 的定时器输出 0: 从 TO0p 输出 0 1: 从 TO0p 输出 1

表 16.56 当使用单次脉冲输出功能时，从通道的 TOE0 设置示例

少量	象征	设置值	功能
p	TOE[p]	1/0	启用或禁用通道 p 的定时器输出 0: 通过计数操作停止TO0p输出操作。 1: 通过计数操作启用TO0p输出操作。

表 16.57: 使用单次脉冲输出功能时，从通道的 TOL0 设置示例

少量	象征	设置值	功能
p	TOL[p]	1/0	控制通道 p 的定时器输出 0: 正逻辑输出 (高电平有效) 1: 负逻辑输出 (低电平有效)

表 16.58 当使用单次脉冲输出功能时，从通道的 TOM0 设置示例

少量	象征	设置值	功能
p	TOM[p]	1	控制通道 p (通道 1 至 7) 的定时器输出模式 1: 设置从通道输出模式。

表 16.59 显示了使用单脉冲输出功能时的操作步骤。

Table 16.59 Procedure for operations when the one-shot pulse output function is to be used (1 of 2)

	Step	Software operation	Hardware state
TAU default setting		—	Power-off state (Clock supply is stopped and writing to each register is disabled.)
	<1>	Sets the MSTPD0 bit of Module Stop Control Register D (MSTPCRD) to 0.	→ Power-on state. Each channel stops operating. (Clock supply is started and writing to each register is enabled.)
	<2>	Sets timer clock select register 0 (TPS0). Determines clock frequencies of CK00 and CK01.	—
Channel default setting	<3>	Sets the corresponding bit of the TAU noise filter enable register (TNFEN) to 1. Sets timer mode register 0n, 0p (TMR0n, TMR0p) of two channels to be used (determines operation mode of channels). An output delay is set to timer data register 0n (TDR0n) of the master channel, and a pulse width is set to the TDR0p register of the slave channel.	Channel stops operating. (Clock is supplied and some power is consumed.)
	<4>	Sets slave channel. The TOM0.TOM[p] bit of timer output mode register 0 (TOM0) is set to 1 (slave channel output mode). Sets the TOL0.TOL[p] bit. Sets the TO0.TO[p] bit and determines default level of the TO0p output. Sets the TOE0.TOE[p] bit to 1 and enables operation of TO0p. Sets the Prs Direction Register (PDRr) to 1.	The TO0p pin goes into Hi-Z output state. → The TO0p default setting level is output when the Prs Direction Register (PDRr) is in output mode. → TO0p does not change because channel stops operating. → The TO0p pin outputs the TO0p set level.
Operation start	<5>	Sets the TOE0.TOE[p] bit (slave) to 1 (only when operation is resumed). The TS0.TS[n] (master) and TS0.TS[p] (slave) bits of timer channel start register 0 (TS0) are set to 1 at the same time. The TS0.TS[n] and TS[p] bits automatically return to 0 because they are trigger bits.	→ The TE0.TE[n] and TE[p] bits are set to 1 and the master channel enters the start trigger detection (the valid edge of the TI0n pin input is detected or the TS0.TS[n] bit of the master channel is set to 1) wait state. Counter stops operating.
	<6>	Count operation of the master channel is started by start trigger detection of the master channel. • Detects the TI0n pin input valid edge. • Sets the TS0.TS[n] bit of the master channel to 1 by software*1.	Master channel starts counting.
During operation	<7>	Set values of only the CIS[1:0] bits of the TMR0n register can be changed. Set values of the TMR0p, TDR0n, TDR0p registers, TOM0.TOM[n], TOM[p], TOL0.TOL[n], and TOL[p] bits cannot be changed. The TCR0n and TCR0p registers can always be read. The TSR0n and TSR0p registers are not used. Set values of the TO0 and TOE0 registers by slave channel can be changed.	Master channel loads the value of the TDR0n register to timer counter register 0n (TCR0n) by the start trigger detection (the valid edge of the TI0n pin input is detected or the TS0.TS[n] bit of the master channel is set to 1), and the counter starts counting down. When the count value reaches TCR0n = 0x0000, the TAU0_TMI0n output is generated, and the counter stops until the next start trigger detection. The slave channel, triggered by TAU0_TMI0n of the master channel, loads the value of the TDR0p register to the TCR0p register, and the counter starts counting down. The output level of TO0p becomes active one count clock after generation of TAU0_TMI0n from the master channel. It becomes inactive when TCR0p = 0x0000, and the counting operation is stopped. After that, the above operation is repeated.

表 16.59 使用单脉冲输出功能时的操作步骤 (1/2)

	步	软件操作	硬件状态
TAU 默认设置		—	关机状态 (时钟信号停止, 所有寄存器的写入操作均被禁用。)
	<1>	将模块停止控制寄存器 D (MSTPCRD) 的 MSTPD0 位设置为 0。	→ 上电状态。各通道停止工作。 (时钟电源已启动, 并且对每个寄存器的写入操作已启用。)
	<2>	设置定时器时钟选择寄存器 0 (TPS0)。 确定 CK00 和 CK01 的时钟频率。	—
通道默认设置	<3>	将 TAU 噪声滤波器使能寄存器 (TNFEN) 的相应位设置为 1。 设置要使用的两个通道的定时器模式寄存器 0n、0p (TMR0n、TMR0p) (确定通道的操作模式)。 将输出延迟设置到主通道的定时器数据寄存器 0n (TDR0n), 并将脉冲宽度设置到从通道的 TDR0p 寄存器。	频道停止运行。 (时钟信号被提供, 并消耗一定功率。)
	<4>	设置从属通道。 定时器输出模式寄存器 0 (TOM0) 的 TOM0.TOM[p] 位设置为 1 (从通道输出模式)。 设置 TOL0.TOL[p] 位。 设置 TO0.TO[p] 位, 并确定 TO0p 输出的默认电平。 将 TOE0.TOE[p] 位设置为 1, 使 TO0p 能够运行。 将 PRS 方向寄存器 (PDRr) 设置为 1。	TO0p引脚进入高阻抗输出状态。 → 当 Prs 方向寄存器 (PDRr) 处于输出模式时, 输出 TO0p 默认设置电平。 → TO0p 不会改变, 因为通道停止运行。 → TO0p引脚输出TO0p置位电平。
操作开始	<5>	将 TOE0.TOE[p] 位 (从设备) 设置为 1 (仅当操作恢复时)。 定时器通道起始寄存器 0 (TS0) 的 TS0.TS[n] (主) 位和 TS0.TS[p] (从) 位同时设置为 1。 TS0.TS[n] 和 TS[p] 位会自动返回 0, 因为它们是触发位。	→ TE0.TE[n] 和 TE[p] 位设置为 1, 主通道进入启动触发检测 (检测到 TI0n 引脚输入的有效边沿或主通道的 TS0.TS[n] 位设置为 1) 等待状态。 计数器停止运行。
	<6>	主通道的计数操作由主通道的启动触发检测启动。 ●检测TI0n引脚输入有效边沿。 ●通过软件*1将主通道的TS0.TS[n] 位设置为1。	主频道开始计数。
运行期间	<7>	只能更改 TMR0n 寄存器的 CIS[1:0] 位的值。 TMR0p、TDR0n、TDR0p 寄存器、TOM0.TOM[n]、TOM[p]、TOL0.TOL[n]和 TOL[p]位的值不能更改。 TCR0n 和 TCR0p 寄存器始终可读。TSR0n 和 TSR0p 寄存器未使用。从通道可以更改 TO0 和 TOE0 寄存器的设置值。	主通道通过启动触发检测 (检测到TI0n引脚输入的有效边沿或主通道的TS0.TS[n] 位设置为1) 将TDR0n寄存器的值加载到定时器计数器寄存器0n (TCR0n) 中, 计数器开始倒计时。当计数值达到TCR0n =时, 计数器停止计数。 当值为 0x0000 时, 生成 TAU0_TMI0n 输出, 计数器停止计数, 直到检测到下一次启动触发。 从通道在主通道的 TAU0_TMI0n 信号触发下, 将TDR0p寄存器的值加载到 TCR0p 寄存器, 计数器开始倒计时。在主通道生成 TAU0_TMI0n 信号一个时钟周期后, TO0p寄存器的输出电平变为有效。当TCR0p = 0x0000, 寄存器的值变为无效时, 计数操作停止。 之后, 重复上述操作。

Table 16.59 Procedure for operations when the one-shot pulse output function is to be used (2 of 2)

	Step	Software operation		Hardware state
Operation stop	<8>	The TT0.TT[n] (master) and TT[p] (slave) bits are set to 1 at the same time. The TT0.TT[n] and TT[p] bits automatically return to 0 because they are trigger bits.	→	TE0.TE[n], TE[p] = 0, and count operation stops. The TCR0n and TCR0p registers hold count value and stop. The TO0p output is not initialized and retains its current state.
	<9>	The TOE0.TOE[p] bit of slave channel is cleared to 0 and value is set to the TO0.TO[p] bit. To resume operation, go to step <5>. To terminate the operation, go to step <10>	→	The TO0p pin outputs the TO0p set level.
TAU stop	<10>	To hold the TO0p pin output level Set PSEL[2:0] bits to 000b after the value to be held is set to the Prs Output Data Register (PODRr). When holding the TO0p pin output level is not necessary Setting not required.	→	The TO0p pin output level is held by port function
	<11>	Sets the MSTPD0 bit of Module Stop Control Register D (MSTPCRD) to 1.	→	This stops supply of the input clock to timer array unit 0. Power-off state.

Note 1. Do not set the TS0.TS[n] bit of the slave channel to 1.

Note: n = 0, 2, 4, 6 (Master channel number)
 n < p ≤ 7 (Slave channel number)
 r = 0 to 9
 s = 00 to 15

16.8.2 Operation for the PWM Function

Two channels can be used as a set to generate a pulse of any period and duty factor.

The period and duty factor of the output pulse can be calculated by the following expressions.

Pulse period = {Set value of TDR0n (master) + 1} × Count clock period
Duty factor [%] = {Set value of TDR0p (slave)} / {Set value of TDR0n (master) + 1} × 100
0% output: Set value of TDR0p (slave) = 0x0000
100% output: Set value of TDR0p (slave) ≥ {Set value of TDR0n (master) + 1}

Note: The duty factor exceeds 100% if the set value of TDR0p (slave) > (set value of TDR0n (master) + 1), it summarizes to 100% output.

The master channel operates in the interval timer mode. If the channel start trigger bit (TS[n]) of timer channel start register 0 (TS0) is set to 1, an interrupt (TAU0_TMI0n) is output, the value set to timer data register 0n (TDR0n) is loaded to timer counter register 0n (TCR0n), and the counter counts down in synchronization with the count clock. When the counter reaches 0x0000, TAU0_TMI0n is output, the value of the TDR0n register is loaded again to the TCR0n register, and the counter counts down. This operation is repeated until the channel stop trigger bit (TT[n]) of timer channel stop register 0 (TT0) is set to 1.

If two channels are used to output a PWM waveform, the period until the master channel counts down to 0x0000 is the PWM output (TO0p) cycle.

The slave channel operates in one-count mode. By using TAU0_TMI0n from the master channel as a start trigger, the TCR0p register loads the value of the TDR0p register and the counter counts down to 0x0000. When the counter reaches 0x0000, it outputs TAU0_TMI0p and waits until the next start trigger (TAU0_TMI0n from the master channel) is generated.

If two channels are used to output a PWM waveform, the period until the slave channel counts down to 0x0000 is the PWM output (TO0p) duty.

PWM output (TO0p) goes to the active level one clock after the master channel generates TAU0_TMI0n and goes to the inactive level when the TCR0p register of the slave channel becomes 0x0000.

表 16.59 使用单次脉冲输出功能时的操作步骤 (2/2)

	步	软件操作		硬件状态
停止操作	<8>	TT0.TT[n]（主设备）和 TT[p]（从设备）位同时设置为 1。 TT0.TT[n] 和 TT[p] 位会自动返回 0，因为它们是触发位。	→	TE0.TE[n].TE[p] = 0，计数操作停止。TCR0n 和 TC R0p 寄存器保存计数值并停止。 TO0p 输出未初始化，并保持其当前状态。
	<9>	从通道的 TOE0.TOE[p] 位被清零，其值被设置为 TO0.TO[p] 位。要恢复操作，请转到步骤<5>。 要终止操作，请转到步骤<10>	→	TO0p引脚输出TO0p置位电平。
停止！	<10>	要保持 TO0p 引脚的输出电平，请在将要保持的值设置到 Prs 输出数据寄存器 (PODRr) 之后，将 PSEL[2:0 和] 位设置为 000b。保持TO0p引脚的输出电平时，无需进行设置。	→	TO0p引脚输出电平由端口功能保持
	<11>	将模块停止控制寄存器 D (MSTPCRD) 的 MSTP D0 位设置为 1。	→	这将停止向定时器阵列单元 0 提供输入时钟。 关机状态。

注意 1. 不要将从通道的 TS0.TS[n] 位设置为 1。

注: n = 0, 2, 4, 6（主通道号）
 n < p ≤ 7（从通道号）
 r = 0 to 9
 s = 00 to 15

16.8.2 PWM功能的操作

两个通道可以组合使用，生成任意周期和占空比的脉冲。

输出脉冲的周期和占空比可以通过以下表达式计算。

脉冲周期= {设置 TDR0n（主控）的值+ 1} × 计数时钟周期
占空比[%] = {设置 TDR0p（从机）的值} / {设置 TDR0n 的值(master) + 1} × 100
0%输出：设置 TDR0p（从设备）的值= 0x0000
100%输出：设置值 o f TDR0p (slave) ≥ {Set value of TDR0n (master) + 1}

注意：如果 TDR0p（从设备）的设定值>（TDR0n（主设备）的设定值+ 1）超过占空比100%，则汇总输出100%。

主通道工作在间隔定时器模式。如果定时器通道起始寄存器 0 (TS0) 的通道起始触发位(TS[n]) 被置为 1，则输出中断 (TAU0_TMI0n)，定时器数据寄存器 0n (TDR0n) 的值被加载到定时器计数器寄存器 0n (TCR0n) 中，计数器与计数时钟同步递减计数。当计数器达到0x0000时，输出 TAU0_TMI0n，TDR0n 寄存器的值再次被加载到 TCR0n 寄存器中，计数器继续递减计数。重复此操作，直到定时器通道停止寄存器 0 (TT0) 的通道停止触发位 (TT[n]) 被置为 1。

如果使用两个通道输出 PWM 波形，则主通道倒数到0x0000的周期即为 PWM 输出 (TO0p) 周期。

从通道以单次计数模式运行。它使用来自主通道的 TAU0_TMI0n 作为启动触发信号，将 TDR0p 寄存器的值加载到 TCR 0p 寄存器，计数器开始计数至0x0000。当计数器达到0x0000时，它输出 TAU0_TMI0p，并等待下一个启动触发信号（来自主通道的 TAU0_TMI0n）的出现。

如果使用两个通道输出 PWM 波形，则从通道倒数到0x0000的周期是 PWM 输出(TO0p) 的占空比。

PWM 输出 (TO0p) 在主通道生成 TAU0_TMI0n 后一个时钟周期变为有效电平，当从通道的 TCR0p 寄存器变为0x000 0时变为无效电平。

In the 16-pin products, this function can be used by setting channels 0 and 2 as the master channel and the slave channel, respectively.

Note: To rewrite both timer data register 0n (TDR0n) of the master channel and the TDR0p register of the slave channel, a write access is necessary two times. The timing at which the values of the TDR0n and TDR0p registers are loaded to the TCR0n and TCR0p registers is upon occurrence of TAU0_TMI0n of the master channel. Thus, when rewriting is performed split before and after occurrence of TAU0_TMI0n of the master channel, the TO0p pin cannot output the expected waveform. To rewrite both the TDR0n register of the master and the TDR0p register of the slave, therefore, be sure to rewrite both the registers immediately after TAU0_TMI0n is generated from the master channel.

Note: n = 0, 2, 4, 6 (master channel number)
n < p ≤ 7 (Slave channel number)

Figure 16.51 shows a block diagram for operation for the PWM function

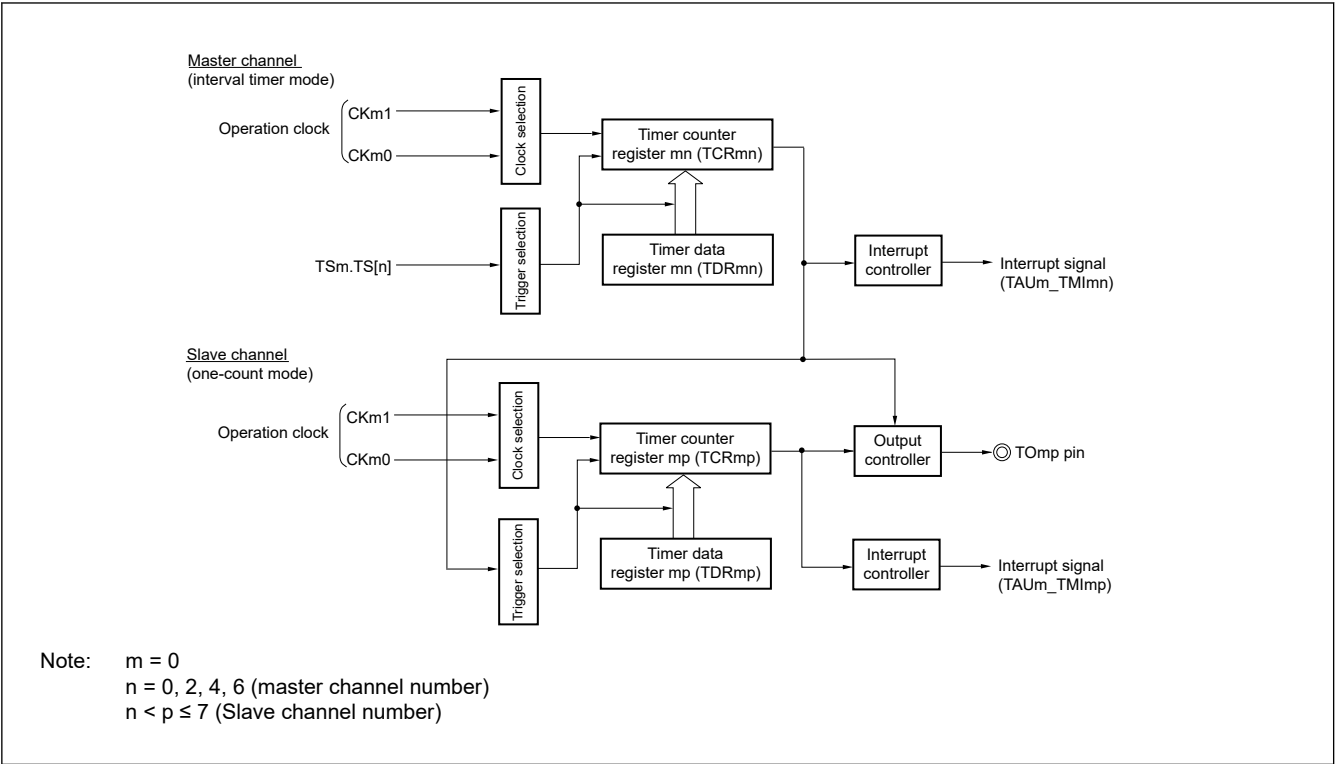


Figure 16.51 Block diagram for operation for the PWM function

Figure 16.52 shows an example of basic timing during operation for the PWM function.

在 16 针产品中，可以通过将通道 0 和 2 分别设置为主通道和从通道来使用此功能。

注意：要同时重写主通道的定时器数据寄存器 0n (TDR0n) 和从通道的定时器数据寄存器 0p (TDR0p)，需要进行两次写入操作。TDR0n 和 TDR0p 寄存器的值加载到 TCR0n 和 TCR0p 寄存器的时机是在主通道发生 TAU0_TMI0n 事件时。因此，如果重写操作在主通道发生 TAU0_TMI0n 事件前后分别进行，则 TO0p 引脚将无法输出预期波形。所以，要同时重写主通道的 TDR0n 寄存器和从通道的 TDR0p 寄存器，务必在主通道生成 TAU0_TMI0n 事件后立即重写这两个寄存器。

注：n = 0, 2, 4, 6 (主频道号)
n < p ≤ 7 (从通道号)

图 16.51 显示了 PWM 功能的操作框图

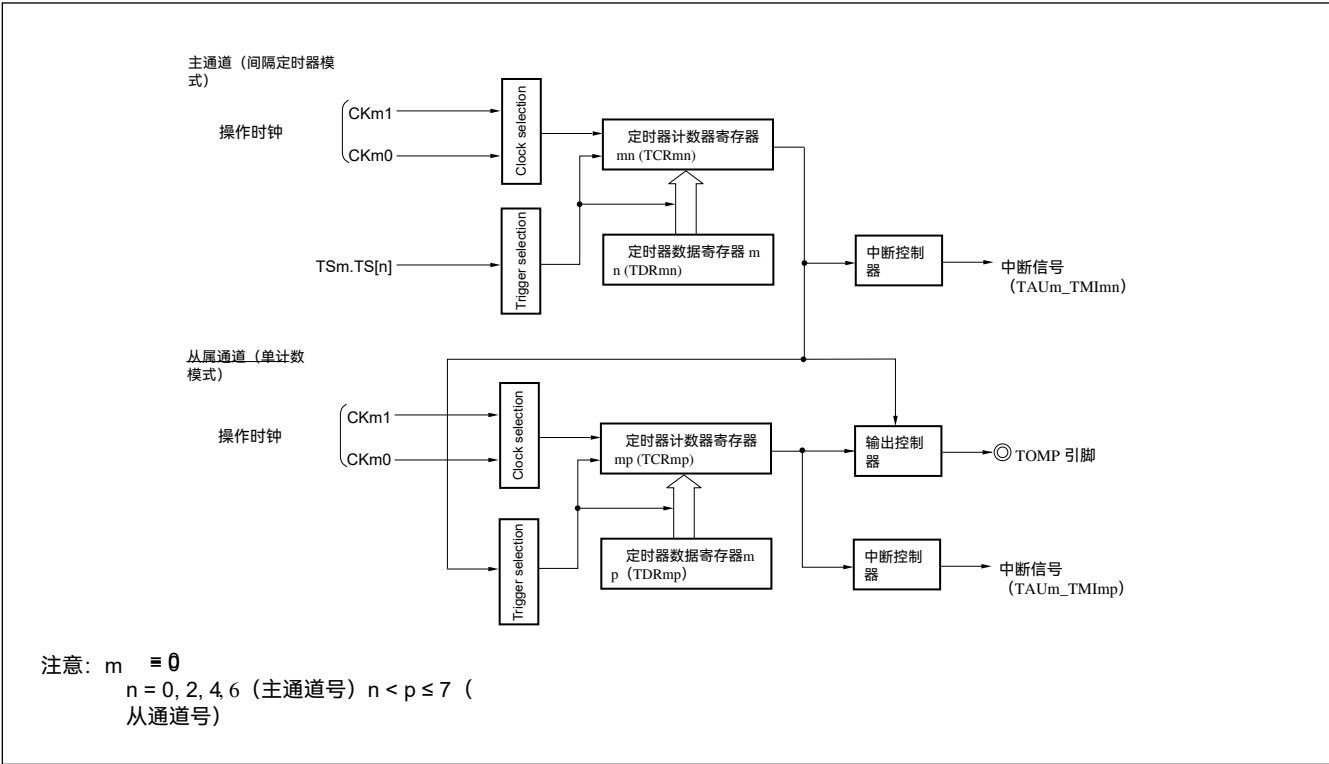


图 16.51 PWM 功能的操作框图

图 16.52 显示了 PWM 功能运行期间的基本时序示例。

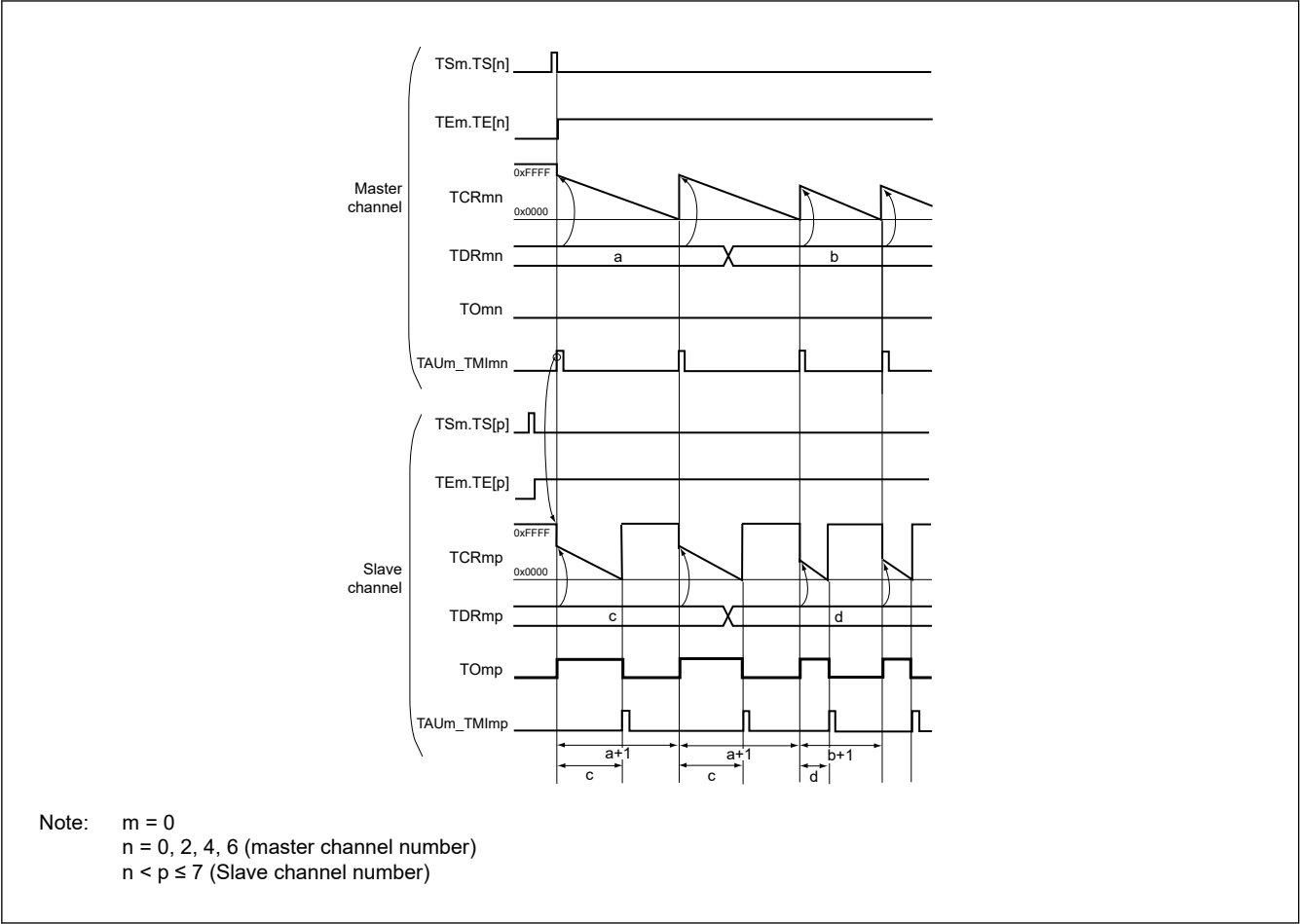


Figure 16.52 Example of basic timing during operation for the PWM function

Table 16.60 to Table 16.64 show register settings for the master channel when the PWM function is to be used.

Table 16.60 Example of TMR0n settings for the master channel when the PWM function is to be used (1 of 2)

Bit	Symbol	Set value	Function
0	OPIRQ	1	Setting of operation when counting is started 1: Generates TAU0_TMI0n when counting is started
3:1	MD[2:0]	000b	Operation mode of channel n 0 0 0: Interval timer
5:4	—	00b	Fixed to 0
7:6	CIS[1:0]	00b	Selection of TI0n pin input edge 0 0: Set to 00b because the TI0n input pin is not to be used
10:8	STS[2:0]	000b	Start trigger selection 0 0 0: Selects only software start
11	— (n = 0)	0	Fixed to 0 (channels 0)
	MASTER (n = 2, 4, 6)	1	Setting of MASTER bit (channels 2, 4, 6) 1: Master channel
12	CCS	0	Count clock selection 0: Selects operation clock (fMCK)
13	—	0	Fixed to 0.

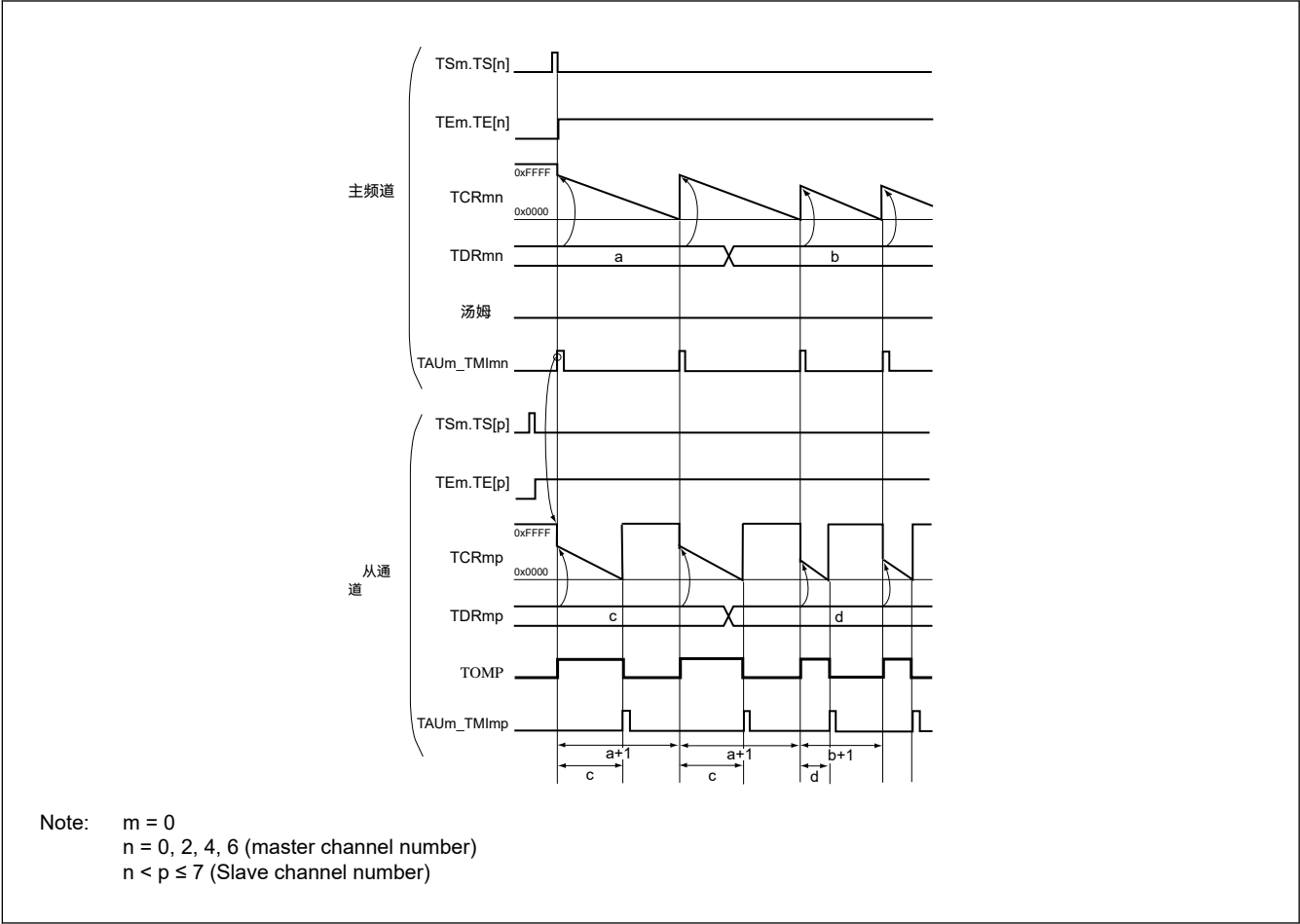


图 16.52 PWM 功能运行期间的基本时序示例

表 16.60 至表 16.64 显示了使用 PWM 功能时主通道的寄存器设置。

表 16.60 使用 PWM 功能时主通道的 TMR0n 设置示例 (1/2)

少量	象征	设置值	功能
0	OPIRQ	1	计数开始时的操作设置 1: 计数开始时生成 TAU0_TMI0n
3:1	MD[2:0]	000b	通道 n 的工作模式 0 0 0: 间隔定时器
5:4	—	00b	固定为 0
7:6	CIS[1:0]	00b	选择 TI0n 引脚输入边沿 0 0: 设置为 00b, 因为 TI0n 输入引脚不使用。
10:8	STS[2:0]	000b	开始触发选择 0 0 0: 仅选择软件启动
11	— (n = 0)	0	固定为 0 (通道 0)
	MASTER (n = 2, 4, 6)	1	主控位设置 (通道 2、4、6) 1: 主通道
12	CCS	0	计数时钟选择 0: 选择操作时钟 (fMCK)
13	—	0	固定值为 0。

Table 16.60 Example of TMR0n settings for the master channel when the PWM function is to be used (2 of 2)

Bit	Symbol	Set value	Function
15:14	CKS[1:0]	00b or 10b	Selection of the operating clock (f _{MCK}) 0 0: Selects CK00 as the operating clock for channel n 1 0: Selects CK01 as the operating clock for channel n

Table 16.61 Example of TO0 settings for the master channel when the PWM function is to be used

Bit	Symbol	Set value	Function
n	TO[n]	0	Timer output of channel n 0: Outputs 0 from TO0n.

Table 16.62 Example of TOE0 settings for the master channel when the PWM function is to be used

Bit	Symbol	Set value	Function
n	TOE[n]	0	Enabling or disabling timer output for channel n 0: Stops the TO0n output operation by counting operation.

Table 16.63 Example of TOL0 settings for the master channel when the PWM function is to be used

Bit	Symbol	Set value	Function
n	— (n = 0)	0	Fixed to 0 (channels 0)
	TOL[n] (n = 2, 4, 6)		Control of timer output of channel n (channels 2, 4, 6) 0: Set this bit to 0 when TOM0.TOM[n] = 0 (master channel output mode).

Table 16.64 Example of TOM0 settings for the master channel when the PWM function is to be used

Bit	Symbol	Set value	Function
n	— (n = 0)	0	Fixed to 0 (channels 0)
	TOM[n] (n = 2, 4, 6)		Control of timer output mode of channel n (channels 2, 4, 6) 0: Sets master channel output mode.

Table 16.65 to Table 16.69 show register settings for the slave channel when the PWM function is to be used.

Table 16.65 Example of TMR0p settings for the slave channel when the PWM function is to be used (1 of 2)

Bit	Symbol	Set value	Function
0	OPIRQ	1	Start trigger during operation 1: Trigger input is valid
3:1	MD[2:0]	100b	Operation mode of channel p 1 0 0: One-count mode
5:4	—	00b	Fixed to 0
7:6	CIS[1:0]	00b	Selection of TImp pin input edge 0 0: Set to 00b because the TI0p input pin is not to be used
10:8	STS[2:0]	100b	Start trigger selection 1 0 0: Selects TAU0_TMI0n of master channel
11	— (p = 5, 7)	0	Fixed to 0 (channels 5, 7)
	SPLIT (p = 1, 3)	0	Setting of SPLIT bit (channels 1, 3) 0: 16-bit timer mode
	MASTER (p = 2, 4, 6)	0	Setting of MASTER bit (channels 2, 4, 6) 0: Slave channel

表 16.60 使用 PWM 功能时主通道的 TMR0n 设置示例 (2/2)

少量	象征	设置值	功能
15:14	CKS[1:0]	00b or 10b	选择工作时钟 (f _{MCK}) 0 0: 选择 CK00 作为通道 n 的工作时钟 1 0: 选择 CK01 作为通道 n 的工作时钟

表 16.61 使用 PWM 功能时主通道的 TO0 设置示例

少量	象征	设置值	功能
n	TO[n]	0	通道 n 的定时器输出 0: 从 TO0n 输出 0。

表 16.62 使用 PWM 功能时主通道的 TOE0 设置示例

少量	象征	设置值	功能
n	TOE[n]	0	启用或禁用通道 n 的定时器输出 0: 通过计数操作停止 TO0n 输出操作。

表 16.63 使用 PWM 功能时主通道的 TOL0 设置示例

少量	象征	设置值	功能
n	— (n = 0)	0	固定为 0 (通道 0)
	TOL[n] (n = 2, 4, 6)		控制通道 n (通道 2、4、6) 的定时器输出 0: 当 TOM0.TOM[n] = 0 时 (主通道输出模式)，将此位设置为 0。

表 16.64 使用 PWM 功能时主通道的 TOM0 设置示例

少量	象征	设置值	功能
n	— (n = 0)	0	固定为 0 (通道 0)
	TOM[n] (n = 2, 4, 6)		控制通道 n (通道 2、4、6) 的定时器输出模式 0: 设置主通道输出模式。

表 16.65 至表 16.69 显示了使用 PWM 功能时从通道的寄存器设置。

表 16.65 使用 PWM 功能时从通道的 TMR0p 设置示例 (1/2)

少量	象征	设置值	功能
0	OPIRQ	1	操作期间启动触发器 1: 触发输入有效
3:1	MD[2:0]	100b	通道 p 的工作模式 1 0 0: 单次计数模式
5:4	—	00b	固定为 0
7:6	CIS[1:0]	00b	选择 TImp 引脚输入边沿 0 0: 设置为 00b，因为 TI0p 输入引脚不使用。
10:8	STS[2:0]	100b	开始触发选择 1 0 0: 选择主通道的 TAU0_TMI0n
11	— (p = 5, 7)	0	固定为 0 (通道 5、7)
	SPLIT (p = 1, 3)	0	SPLIT 位 (通道 1、3) 设置: 0: 1 6 位定时器模式
	MASTER (p = 2, 4, 6)	0	主控位设置 (通道 2、4、6) 0: 从属通道

Table 16.65 Example of TMR0p settings for the slave channel when the PWM function is to be used (2 of 2)

Bit	Symbol	Set value	Function
12	CCS	0	Count clock selection 0: Selects operation clock (f _{MCK})
13	—	0	Fixed to 0.
15:14	CKS[1:0]	00b or 10b	Selection of the operating clock (f _{MCK}) (Make the same setting as master channel.) 0 0: Selects CK00 as the operating clock for channel p 1 0: Selects CK01 as the operating clock for channel p

Table 16.66 Example of TO0 settings for the slave channel when the PWM function is to be used

Bit	Symbol	Set value	Function
p	TO[p]	1/0	Timer output of channel p 0: Outputs 0 from TO0p 1: Outputs 1 from TO0p

Table 16.67 Example of TOE0 settings for the slave channel when the PWM function is to be used

Bit	Symbol	Set value	Function
p	TOE[p]	1/0	Enabling or disabling timer output for channel p 0: Stops the TO0p output operation by counting operation. 1: Enables the TO0p output operation by counting operation.

Table 16.68 Example of TOL0 settings for the slave channel when the PWM function is to be used

Bit	Symbol	Set value	Function
p	TOL[p]	1/0	Control of timer output of channel p 0: Positive logic output (active-high) 1: Negative logic output (active-low)

Table 16.69 Example of TOM0 settings for the slave channel when the PWM function is to be used

Bit	Symbol	Set value	Function
p	TOM[p]	1	Control of timer output mode of channel p (channels 1 to 7) 1: Sets the slave channel output mode.

Table 16.70 show procedure for operations when the PWM function is to be used.

Table 16.70 Procedure for operations when the PWM function is to be used (1 of 2)

	Step	Software operation	Hardware state
TAU default setting		—	Power-off state (Clock supply is stopped and writing to each register is disabled.)
	<1>	Sets the MSTPD0 bit of Module Stop Control Register D (MSTPCRD) to 0.	→ Power-on state. Each channel stops operating. (Clock supply is started and writing to each register is enabled.)
	<2>	Sets timer clock select register 0 (TPS0). Determines clock frequencies of CK00 and CK01.	—

表 16.65 使用 PWM 功能时从通道的TMR0p设置示例 (2/2)

少量	象征	设置值	功能
12	CCS	0	计数时钟选择 0: 选择操作时钟 (f _{MCK})
13	—	0	固定值为0。
15:14	CKS[1:0]	00b 或 10b	选择工作时钟(f _{MCK}) (设置与主通道相同。) 0 0: 选择 CK00 作为通道 p 的工作时钟 1 0: 选择 CK01 作为通道 p 的工作时钟

表 16.66 当使用 PWM 功能时，从通道的 TO0 设置示例

少量	象征	设置值	功能
p	TO[p]	1/0	通道 p 的定时器输出 0: 从 TO0p 输出 0 1: 从 TO0p 输出 1

表 16.67 当使用 PWM 功能时，从通道的 TOE0 设置示例

少量	象征	设置值	功能
p	TOE[p]	1/0	启用或禁用通道 p 的定时器输出 0: 通过计数操作停止 TO0p 输出操作。 1: 通过计数操作启用TO0p 输出操作。

表 16.68 使用 PWM 功能时从通道的 TOL0 设置示例

少量	象征	设置值	功能
p	TOL[p]	1/0	控制通道 p 的定时器输出 0: 正逻辑输出 (高电平有效) 1: 负逻辑输出 (低电平有效)

表 16.69 使用 PWM 功能时，从通道的 TOM0 设置示例

少量	象征	设置值	功能
p	TOM[p]	1	控制通道 p (通道 1 至 7) 的定时器输出模式 1: 设置从通道输出模式。

表 16.70 显示了使用 PWM 功能时的操作步骤。

表 16.70 使用 PWM 功能时的操作步骤 (1/2)

	步	软件操作	硬件状态
TAU 默认设置		—	关机状态 (时钟信号停止，所有寄存器的写入操作均被禁用。)
	<1>	将模块停止控制寄存器 D (MSTPCRD) 的 MSTPD0 位设置为 0。	→ 上电状态。各通道停止工作。 (时钟电源已启动，并且对每个寄存器的写入操作已启用。)
	<2>	设置定时器时钟选择寄存器 0 (TPS0)。确定 CK00 和 CK01 的时钟频率。	—

Table 16.70 Procedure for operations when the PWM function is to be used (2 of 2)

	Step	Software operation	Hardware state
Channel default setting	<3>	Sets timer mode register 0n, 0p (TMR0n, TMR0p) of two channels to be used (determines operation mode of channels). An interval (period) value is set to timer data register 0n (TDR0n) of the master channel, and a duty factor is set to the TDR0p register of the slave channel.	Channel stops operating. (Clock is supplied and some power is consumed.)
	<4>	Sets slave channel. The TOM0.TOM[p] bit of timer output mode register 0 (TOM0) is set to 1 (slave channel output mode). Sets the TOL0.TOL[p] bit. Sets the TO0.TO[p] bit and determines default level of the TO0p output. Sets the TOE0.TOE[p] bit to 1 and enables operation of TO0p. Sets the Prs Direction Register (PDRr) to 1.	The TO0p pin goes into Hi-Z output state. → The TO0p default setting level is output when the Prs Direction Register (PDRr) is in output mode. → TO0p does not change because channel stops operating. → The TO0p pin outputs the TO0p set level.
Operation start	<5>	Sets the TOE0.TOE[p] bit (slave) to 1 (only when operation is resumed). The TS[n] (master) and TS[p] (slave) bits of timer channel start register 0 (TS0) are set to 1 at the same time. The TS0.TS[n] and TS[p] bits automatically return to 0 because they are trigger bits.	→ TE0.TE[n] = 1, TE0.TE[p] = 1 When the master channel starts counting, TAU0_TMI0n is generated. Triggered by this interrupt, the slave channel also starts counting.
During operation	<6>	Set values of the TMR0n and TMR0p registers, TOM0.TOM[n], TOM[p], TOL0.TOL[n], and TOL[p] bits cannot be changed. Set values of the TDR0n and TDR0p registers can be changed after TAU0_TMI0n of the master channel is generated. The TCR0n and TCR0p registers can always be read. The TSR0n and TSR0p registers are not used.	The counter of the master channel loads the TDR0n register value to timer counter register 0n (TCR0n), and counts down. When the count value reaches TCR0n = 0x0000, TAU0_TMI0n output is generated. At the same time, the value of the TDR0n register is loaded to the TCR0n register, and the counter starts counting down again. At the slave channel, the value of the TDR0p register is loaded to the TCR0p register, triggered by TAU0_TMI0n of the master channel, and the counter starts counting down. The output level of TO0p becomes active one count clock after generation of the TAU0_TMI0n output from the master channel. It becomes inactive when TCR0p = 0x0000, and the counting operation is stopped. After that, the above operation is repeated.
Operation stop	<7>	The TT0.TT[n] (master) and TT[p] (slave) bits are set to 1 at the same time. The TT0.TT[n] and TT[p] bits automatically return to 0 because they are trigger bits.	→ TE0.TE[n], TE[p] = 0, and count operation stops. The TCR0n and TCR0p registers hold count value and stop. The TO0p output is not initialized and retains its current state.
	<8>	The TOE0.TOE[p] bit of slave channel is cleared to 0 and value is set to the TOM0.TOM[p] bit. To resume operation, go to step <5>. To terminate the operation, go to step <9>	→ The TO0p pin outputs the TO0p set level.
TAU stop	<9>	To hold the TO0p pin output level Set PSEL[2:0] bits to 000b after the value to be held is set to the Prs Output Data Register (PODRr). When holding the TO0p pin output level is not necessary Setting not required.	→ The TO0p pin output level is held by port function
	<10>	Sets the MSTPD0 bit of Module Stop Control Register D (MSTPCRD) to 1.	→ This stops supply of the input clock to timer array unit 0. Power-off state.

Note: n = 0, 2, 4, 6 (Master channel number)
n < p ≤ 7 (Slave channel number)
r = 0 to 9
s = 00 to 15

表 16.70 使用 PWM 功能时的操作步骤 (2/2)

	步	软件操作	硬件状态
通道默认设置	<3>	设置两个通道的定时器模式寄存器 0n、0p (TMR0n、TMR0p) 的值 (确定工作模式)。 通道数)。 将间隔 (周期) 值设置到主通道的定时器数据寄存器 0n (TDR0n), 并将占空比设置到从通道的 TDR0p 寄存器。	频道停止运行。 (时钟信号被提供, 并消耗一定功率。)
	<4>	设置从属通道。 定时器输出模式寄存器 0 (TOM0) 的 TOM0.TOM[p] 位设置为 1 (从通道输出模式)。 设置 TOL0.TOL[p] 位。 设置 TO0.TO[p] 位, 并确定 TO0p 输出的默认电平。 将 TOE0.TOE[p] 位设置为 1, 使 TO0p 能够运行。 将 PRS 方向寄存器 (PDRr) 设置为 1。	TO0p引脚进入高阻抗输出状态。 → 当 Prs 方向寄存器 (PDRr) 处于输出模式时, 输出 TO0p 默认设置电平。 → TO0p 不会改变, 因为通道停止运行。 → TO0p 引脚输出TO0p 设置电平。
操作开始	<5>	将 TOE0.TOE[p] 位 (从设备) 设置为 1 (仅当操作恢复时)。 定时器通道起始寄存器 0 (TS0) 的 TS[n] (主设备) 和 TS[p] (从设备) 位在以下情况下被设置为 1: 同时。 TS0.TS[n] 和 TS[p] 位会自动返回 0, 因为它们是触发位。	→ TE0.TE[n] = 1, TE0.TE[p] = 1 当主通道开始计数时, 会生成 TAU0_TMI0n 中断。受此中断触发, 从通道也开始计数。
运行期间	<6>	设置TMR0n和TMR0p寄存器的值, TOM0.TOM[n], TOM[p], TOL0.TOL[n], 和 TOL[p]位不能更改。 主通道的 TAU0_TMI0n 生成后, 可以更改 TDR0n 和 TDR0p 寄存器的设置值。 TCR0n 和 TCR0p 寄存器始终可读。TSR0n 和 TSR0p 寄存器未使用。	主通道的计数器加载 TDR0n 将寄存器值加载到定时器计数器寄存器 0n (TCR0n) 中, 并开始倒计时。当计数达到TCR0n = 0x0000时, 产生 TAU0_TMI0n 输出。同时, 将 TDR0n 寄存器的值加载到 TCR0n 寄存器中, 计数器再次开始倒计时。 在从通道, 主通道的 TAU0_TMI0n 信号触发后, TDR0p寄存器的值被加载到TCR0p 寄存器, 计数器开始倒计时。在主通道产生 TAU0_TMI0n 输出后一个时钟周期, TO0p寄存器的输出电平变为有效。当TCR0p = 0x0000寄存器变为无效时, 计数操作停止。 之后, 重复上述操作。
停止操作	<7>	TT0.TT[n] (主设备) 和 TT[p] (从设备) 位同时设置为 1。 TT0.TT[n] 和 TT[p] 位会自动返回 0, 因为它们是触发位。	→ TE0.TE[n],TE[p] = 0, 计数操作停止。TCR0n 和 TCR0p 寄存器保存计数值并停止。 TO0p 输出未初始化, 并保持其当前状态。
	<8>	从通道的 TOE0.TOE[p] 位被清零, 其值被设置为 TOM0.TOM[p] 位。要恢复操作, 请转到步骤<5>。 要终止操作, 请转到步骤<9>	→ TO0p引脚输出TO0p置位电平。
停止!	<9>	要保持 TO0p 引脚的输出电平, 请在将要保持的值设置到 Prs 输出数据寄存器 (PODRr) 之后, 将 PSEL[2:0] 位设置为 000b。保持TO0p引脚的输出电平时, 无需进行设置。	→ TO0p引脚输出电平由端口功能保持
	<10>	将模块停止控制寄存器 D (MSTPCRD) 的 MSTPD0 位设置为 1。	→ 这将停止向定时器阵列单元 0 提供输入时钟。 关机状态。

注: n = 0、2、4、6 (主通道号)
n < p ≤ 7 (从通道号)
r = 0 to 9
s = 00 to 15

16.8.3 Operation for the Multiple PWM Output Function

By extending the PWM function and using multiple slave channels, many PWM waveforms with different duty values can be output.

For example, when using two slave channels, the period and duty factor of an output pulse can be calculated by the following expressions.

Pulse period = {Set value of TDR0n (master) + 1} × Count clock period

Duty factor 1 [%] = {Set value of TDR0p (slave 1)} / {Set value of TDR0n (master) + 1} × 100

Duty factor 2 [%] = {Set value of TDR0q (slave 2)} / {Set value of TDR0n (master) + 1} × 100

Note: Although the duty factor exceeds 100% if the set value of TDR0p (slave 1) > {set value of TDR0n (master) + 1} or if the {set value of TDR0q (slave 2)} > {set value of TDR0n (master) + 1}, it is summarized into 100% output.

Timer counter register 0n (TCR0n) of the master channel operates in the interval timer mode and counts the periods. The TCR0p register of the slave channel 1 operates in one-count mode, counts the duty factor, and outputs a PWM waveform from the TO0p pin. The TCR0p register loads the value of timer data register 0p (TDR0p), using TAU0_TMI0n of the master channel as a start trigger, and starts counting down. When TCR0p = 0x0000, TCR0p outputs TAU0_TMI0p and stops counting until the next start trigger (TAU0_TMI0n of the master channel) has been input. The output level of TO0p becomes active one count clock after generation of TAU0_TMI0n from the master channel, and inactive when TCR0p = 0x0000.

In the same way as the TCR0p register of the slave channel 1, the TCR0q register of the slave channel 2 operates in one-count mode, counts the duty factor, and outputs a PWM waveform from the TO0q pin. The TCR0q register loads the value of the TDR0q register, using TAU0_TMI0n of the master channel as a start trigger, and starts counting down.

When TCR0q = 0x0000, the TCR0q register outputs TAU0_TMI0q and stops counting until the next start trigger (TAU0_TMI0n of the master channel) has been input. The output level of TO0q becomes active one count clock after generation of TAU0_TMI0n from the master channel, and inactive when TCR0q = 0x0000.

When channel 0 is used as the master channel as above, up to seven types of PWM signals can be output at the same time.

Note: To rewrite both timer data register 0n (TDR0n) of the master channel and the TDR0p register of the slave channel 1, write access is necessary at least twice. Since the values of the TDR0n and TDR0p registers are loaded to the TCR0n and TCR0p registers after TAU0_TMI0n is generated from the master channel, if rewriting is performed separately before and after generation of TAU0_TMI0n from the master channel, the TO0p pin cannot output the expected waveform. To rewrite both the TDR0n register of the master and the TDR0p register of the slave, be sure to rewrite both the registers immediately after TAU0_TMI0n is generated from the master channel (This applies also to the TDR0q register of the slave channel 2).

Note: n = 0, 2, 4 (Master channel number)
p, q : Slave channel number
n < p < q ≤ 7 (Where p and q are integers greater than n)

Figure 16.53 shows a block diagram for operation for the multiple PWM output function (for two types of PWM output).

16.8.3 多路PWM输出功能的操作

通过扩展 PWM 功能并使用多个从通道，可以输出许多具有不同占空比的 PWM 波形。

例如，当使用两个从属通道时，输出脉冲的周期和占空比可以通过以下表达式计算。

每脉冲 周期 = {Set value of TDR0n (master) + 1} × Count clock period

职责事实 1 [%] = {Set value of TDR0p (slave 1)} / {Set value of TDR0n (master) + 1} × 100

职责事实 2 [%] = {Set value of TDR0q (slave 2)} / {Set value of TDR0n (master) + 1} × 100

注意：即使占空比超过100%，如果 TDR0p（从机）的设定值1)>{TDR0n（主机）的设定值+ 1}或 {TDR0q（从机）的设定值2}> {TDR0n（主机）的设定值+ 1}，则汇总到100% 输出中。

主通道的定时器计数器寄存器 0n (TCR0n) 工作在间隔定时器模式，用于计数周期。从通道 1 的寄存器 TCR0p 工作在单次计数模式，用于计数占空比，并从TO0p引脚输出 PWM 波形。TCR0p 寄存器使用主通道的 TAU0_TMI0n 作为起始触发信号，加载定时器数据寄存器0p(TDR0p) 的值，并开始倒计时。当TCR0p = 0x0000时，TCR0p 输出 TAU0_TMI0p 并停止计数，直到接收到下一个起始触发信号（主通道的 TAU0_TMI0n）。TO0p的输出电平在主通道生成 TAU0_TMI0n 后经过一个时钟周期变为有效，当TCR0p =0x0000时变为无效。

与从通道 1 的TCR0p寄存器类似，从通道 2 的TCR0q寄存器以单次计数模式运行，对占空比进行计数，并从 TO0q 引脚输出 PWM 波形。TCR0q 寄存器加载 TDR0q 寄存器的值，以主通道的 TAU0_TMI0n 作为起始触发信号，并开始倒计时。

当TCR0q = 0x0000时，TCR0q 寄存器输出 TAU0_TMI0q 并停止计数，直到输入下一个起始触发信号（主通道的 TAU0_TMI0n）。TO0q的输出电平在主通道生成 TAU0_TMI0n 后一个时钟周期变为有效，当TCR0q = 0x0000时变为无效。

当通道 0 用作上述主通道时，最多可以同时输出七种类型的 PWM 信号。

注意：要同时重写主通道的定时器数据寄存器 0n (TDR0n) 和从通道 1 的TDR0p寄存器，至少需要两次写入操作。由于 TDR0n 和 TDR0p 寄存器的值是在主通道生成 TAU0_TMI0n 之后加载到 TCR0n 和TCR0p 寄存器的，因此如果在主通道生成 TAU0_TMI0n 之前和之后分别执行重写操作，则TO0p 引脚将无法输出预期的波形。要同时重写主通道的 TDR0n 寄存器和从通道的 TDR0p 寄存器，请务必在主通道生成 TAU0_TMI0n 之后立即重写这两个寄存器（这也适用于从通道 2 的 TDR0q 寄存器）。

注：n = 0, 2, 4 （主通道号）
p、q：从属通道号
n < p < q ≤ 7 （其中p和 q 是大于 n 的整数）

图 16.53 显示了多路 PWM 输出功能的操作框图（适用于两种类型的 PWM 输出）。

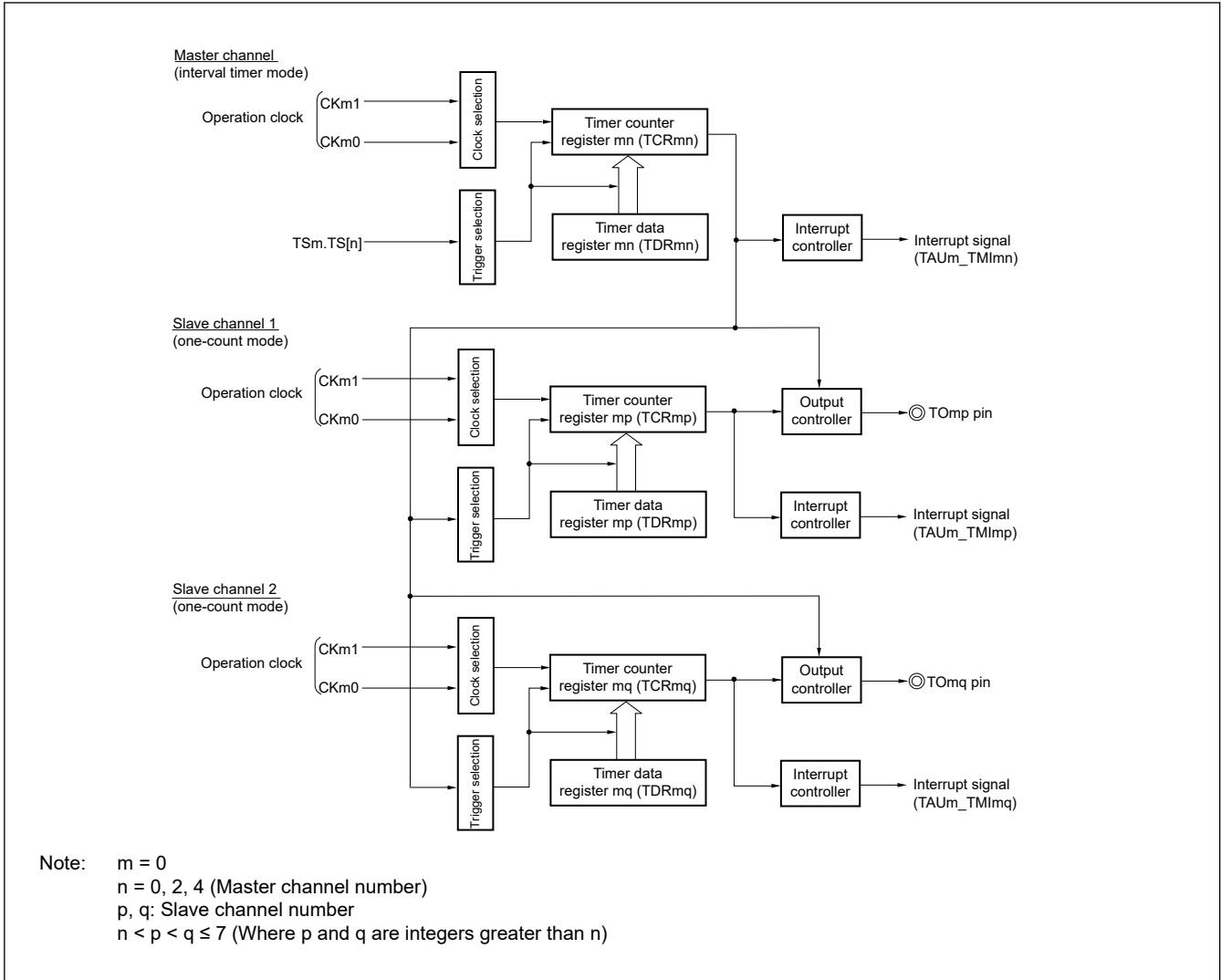


Figure 16.53 Block diagram for the multiple PWM output function (for two types of PWM output)

Figure 16.54 shows an example of basic timing during operation for the multiple PWM output function (for two types of PWM output).

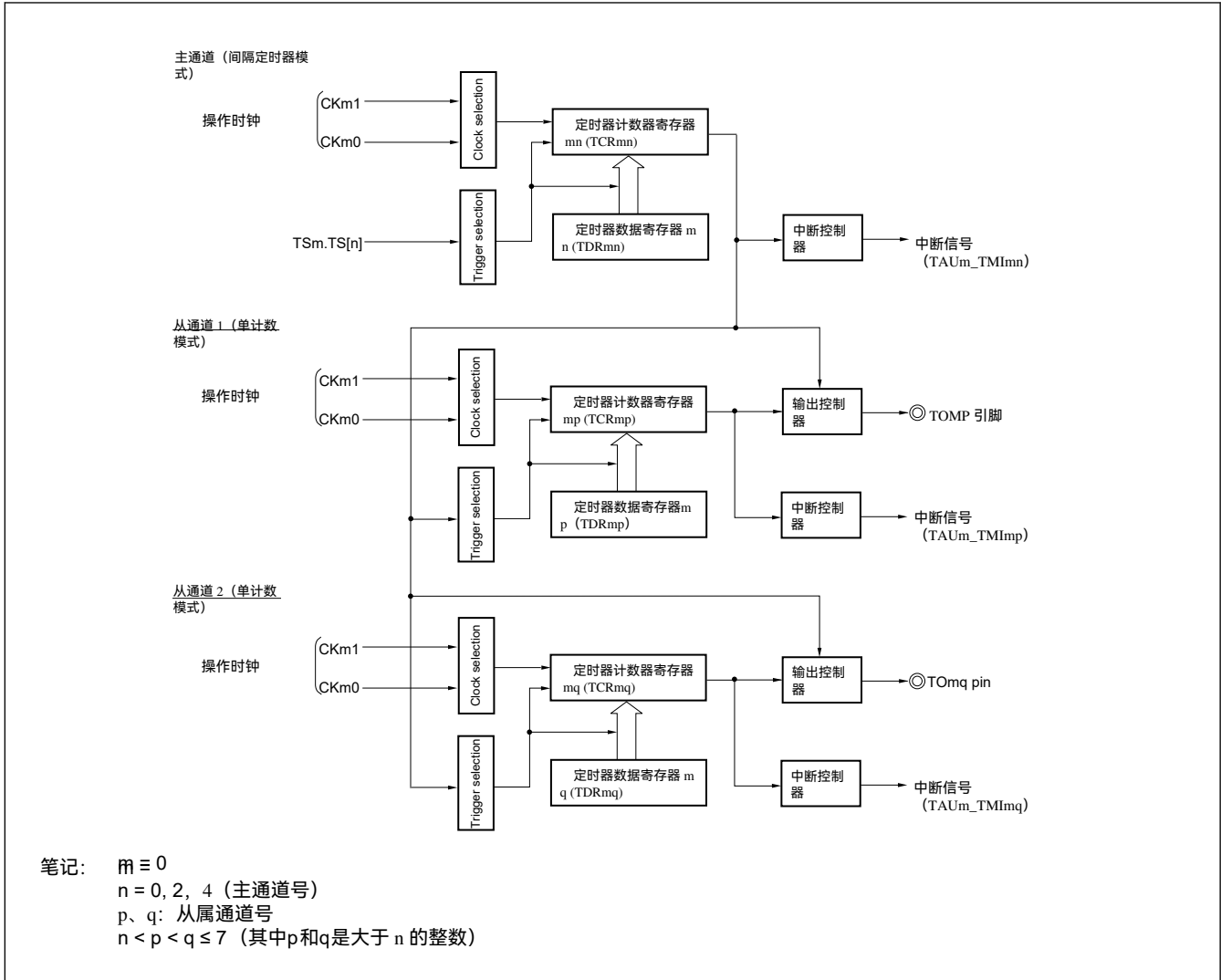


图 16.53 多路 PWM 输出功能的框图 (两种类型的 PWM 输出)

图 16.54 显示了多 PWM 输出功能 (两种类型的 PWM 输出) 运行期间的基本时序示例。

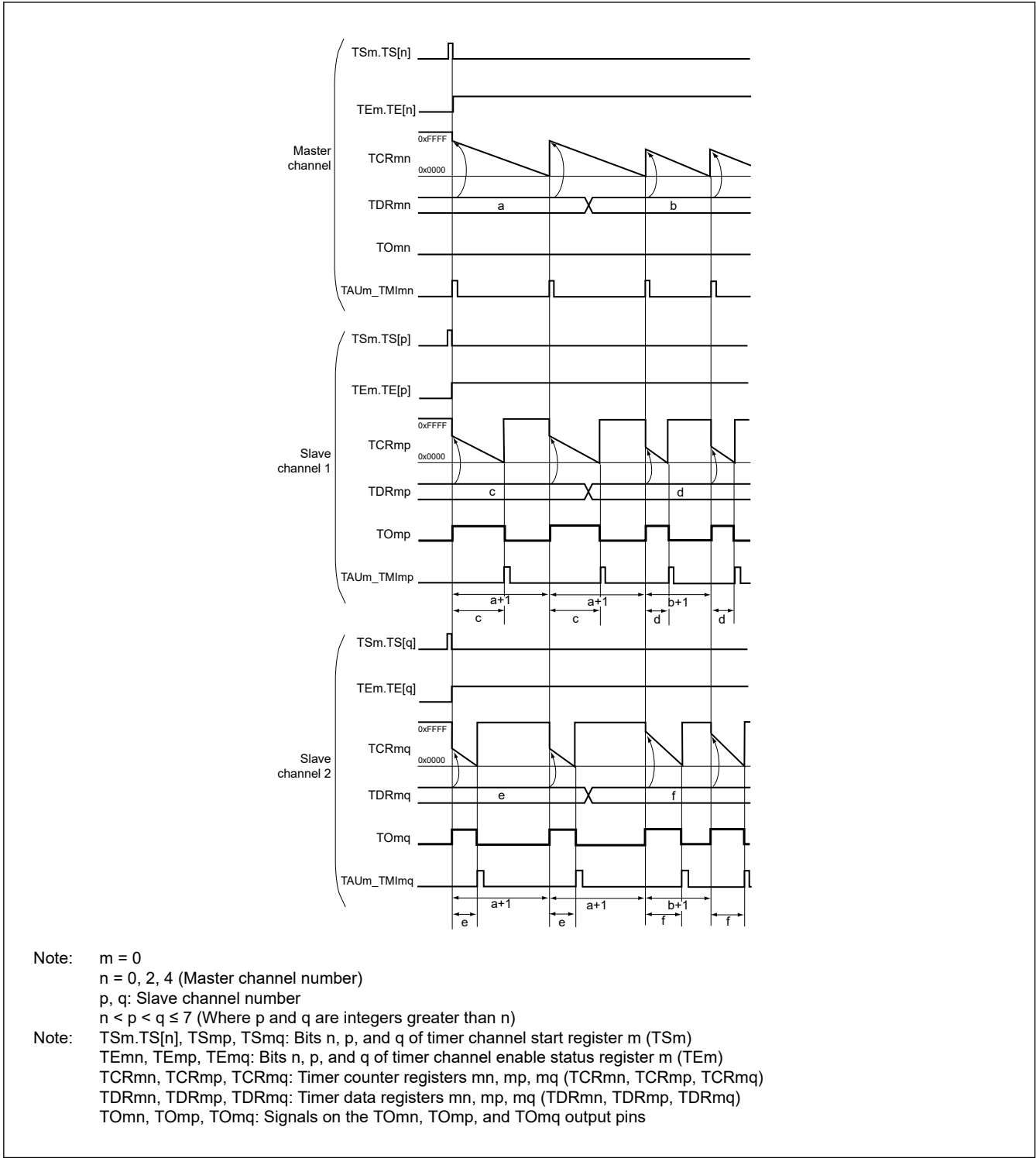


Figure 16.54 Example of basic timing during operation for the multiple PWM output function (for two types of PWM output)

Table 16.71 to Table 16.75 show register settings for the master channel when the multiple PWM output function is to be used.

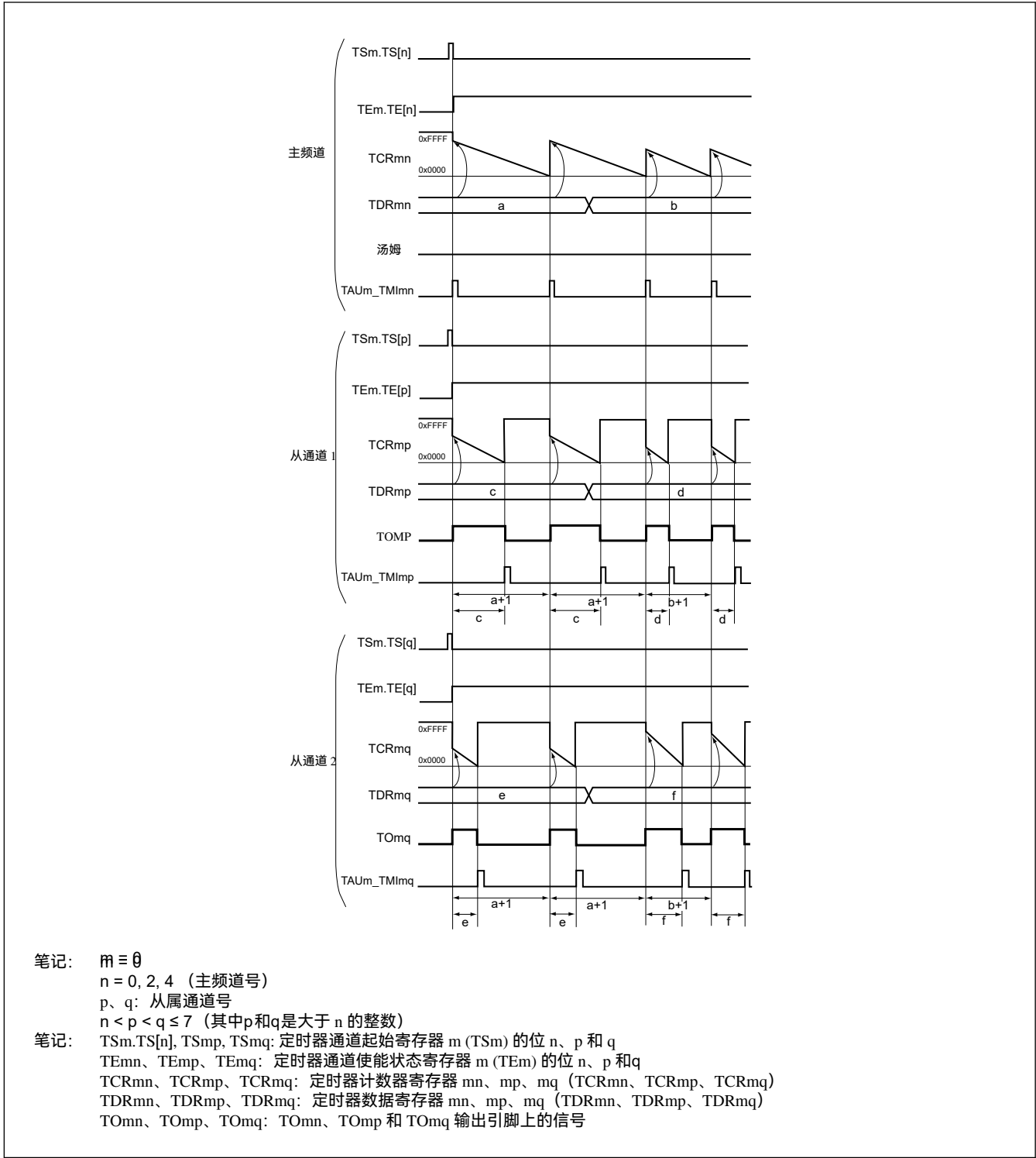


图 16.54 多路 PWM 输出功能运行期间的基本时序示例 (两种类型的 PWM 输出)

表 16.71 至表 16.75 显示了使用多通道 PWM 输出功能时主通道的寄存器设置。

Table 16.71 Example of TMR0n settings for the master channel when the multiple PWM output function is to be used

Bit	Symbol	Set value	Function
0	OPIRQ	1	Setting of operation when counting is started 1: Generates TAU0_TMI0n when counting is started
3:1	MD[2:0]	000b	Operation mode of channel n 0 0 0: Interval timer
5:4	—	00b	Fixed to 0
7:6	CIS[1:0]	00b	Selection of TI0n pin input edge 0 0: Set to 00b because the TI0n input pin is not to be used
10:8	STS[2:0]	000b	Start trigger selection 0 0 0: Selects only software start
11	— (n = 0)	0	Fixed to 0 (channels 0)
	MASTER (n = 2, 4)	1	Setting of MASTER bit (channels 2, 4) 1: Master channel
12	CCS	0	Count clock selection 0: Selects operation clock (fMCK)
13	—	0	Fixed to 0.
15:14	CKS[1:0]	00b or 10b	Selection of the operating clock (fMCK) 0 0: Selects CK00 as the operating clock for channel n 1 0: Selects CK01 as the operating clock for channel n

Table 16.72 Example of TO0 settings for the master channel when the multiple PWM output function is to be used

Bit	Symbol	Set value	Function
n	TO[n]	0	Timer output of channel n 0: Outputs 0 from TO0n

Table 16.73 Example of TOE0 settings for the master channel when the multiple PWM output function is to be used

Bit	Symbol	Set value	Function
n	TOE[n]	0	Enabling or disabling timer output for channel n 0: Stops the TO0n output operation by counting operation.

Table 16.74 Example of TOL0 settings for the master channel when the multiple PWM output function is to be used

Bit	Symbol	Set value	Function
n	— (n = 0)	0	Fixed to 0 (channels 0)
	TOL[n] (n = 2, 4)		Control of timer output of channel n (channels 2, 4) 0: Set this bit to 0 when TOM0.TOM[n] = 0 (master channel output mode).

Table 16.75 Example of TOM0 settings for the master channel when the multiple PWM output function is to be used

Bit	Symbol	Set value	Function
n	— (n = 0)	0	Fixed to 0 (channels 0)
	TOM[n] (n = 2, 4)		Control of timer output mode of channel n (channels 2, 4) 0: Sets master channel output mode.

表 16.71 使用多通道 PWM 输出功能时，主通道的 TMR0n 设置示例

少量	象征	设置值	功能
0	OPIRQ	1	计数开始时的操作设置 1: 计数开始时生成 TAU0_TMI0n
3:1	MD[2:0]	000b	通道 n 的工作模式 0 0 0: 间隔定时器
5:4	—	00b	固定为 0
7:6	CIS[1:0]	00b	选择TI0n引脚输入边沿 0 0: 设置为 00b，因为 TI0n 输入引脚不使用。
10:8	STS[2:0]	000b	开始触发选择 0 0 0: 仅选择软件启动
11	— (n = 0)	0	固定为 0（通道 0）
	MASTER (n = 2, 4)	1	主控位设置（通道 2、4） 1: 主通道
12	CCS	0	计数时钟选择 0: 选择操作时钟（fMCK）
13	—	0	固定值为0。
15:14	CKS[1:0]	00b 或 10b	选择工作时钟（fMCK） 0 0: 选择 CK00 作为通道 n 的工作时钟 1 0: 选择 CK01 作为通道 n 的工作时钟

表 16.72 使用多通道 PWM 输出功能时，主通道 TO0 设置示例

少量	象征	设置值	功能
n	TO[n]	0	通道 n 的定时器输出 0: 从 TO0n 输出 0

表 16.73 使用多通道 PWM 输出功能时，主通道的 TOE0 设置示例

少量	象征	设置值	功能
n	TOE[n]	0	启用或禁用通道 n 的定时器输出 0: 通过计数操作停止 TO0n 输出操作。

表 16.74 使用多通道 PWM 输出功能时，主通道的 TOL0 设置示例

少量	象征	设置值	功能
n	— (n = 0)	0	固定为 0（通道 0）
	TOL[n] (n = 2, 4)		控制通道 n（通道 2、4）的定时器输出 0: 当 TOM0.TOM[n] = 0时（主通道输出模式），将此位设置为 0。

表 16.75 使用多通道 PWM 输出功能时，主通道 TOM0 设置示例

少量	象征	设置值	功能
n	— (n = 0)	0	固定为 0（通道 0）
	TOM[n] (n = 2, 4)		控制通道 n（通道 2、4）的定时器输出模式 0: 设置主通道输出模式。

Table 16.76 to Table 16.81 show register settings for the slave channel when the multiple PWM output function is to be used (for two types of PWM output).

Table 16.76 Example of TMR0p settings for the slave channel when the multiple PWM output function is to be used (for two types of PWM output)

Bit	Symbol	Set value	Function
0	OPIRQ	1	Start trigger during operation 1: Trigger input is valid
3:1	MD[2:0]	100b	Operation mode of channel p 1 0 0: One-count mode
5:4	—	00b	Fixed to 0
7:6	CIS[1:0]	00b	Selection of TI0p pin input edge 0 0: Set to 00b because the TI0p input pins are not to be used
10:8	STS[2:0]	100b	Start trigger selection 1 0 0: Selects TAU0_TMI0n of master channel
11	— (p = 5)	0	Fixed to 0 (channel 5)
	SPLIT (p = 1, 3)	0	Setting of SPLIT bit (channels 1, 3) 0: 16-bit timer mode
	MASTER (p = 2, 4, 6)	0	Setting of MASTER bit (channels 2, 4, 6) 0: Slave channel
12	CCS	0	Count clock selection 0: Selects operation clock (f _{MCK})
13	—	0	Fixed to 0.
15:14	CKS[1:0]	00b or 10b	Selection of the operating clock (f _{MCK}) (make the same setting as master channel.) 0 0: Selects CK00 as the operating clock for channel p 1 0: Selects CK01 as the operating clock for channel p

Table 16.77 Example of TMR0q settings for the slave channel when the multiple PWM output function is to be used (for two types of PWM output) (1 of 2)

Bit	Symbol	Set value	Function
0	OPIRQ	1	Start trigger during operation 1: Trigger input is valid
3:1	MD[2:0]	100b	Operation mode of channel q 1 0 0: One-count mode
5:4	—	00b	Fixed to 0
7:6	CIS[1:0]	00b	Selection of TI0q pin input edge 0 0: Set to 00b because the TI0q input pins are not to be used
10:8	STS[2:0]	100b	Start trigger selection 1 0 0: Selects TAU0_TMI0n of master channel
11	— (q = 5, 7)	0	Fixed to 0 (channels 5, 7)
	SPLIT (q = 3)	0	Setting of SPLIT bit (channel 3) 0: 16-bit timer mode
	MASTER (q = 2, 4, 6)	0	Setting of MASTER bit (channels 2, 4, 6) 0: Slave channel
12	CCS	0	Count clock selection 0: Selects operation clock (f _{MCK})

表 16.76 至表 16.81 显示了使用多 PWM 输出功能时从通道的寄存器设置（适用于两种类型的 PWM 输出）。

表 16.76 当使用多 PWM 输出功能时，从通道的TMR0p设置示例（针对两种类型的 PWM 输出）

少量	象征	设置值	功能
0	OPIRQ	1	操作期间启动触发器 1: 触发输入有效
3:1	MD[2:0]	100b	通道 p 的工作模式 1 0 0: 单次计数模式
5:4	—	00b	固定为 0
7:6	CIS[1:0]	00b	选择 TI0p 引脚输入边沿 0 0: 设置为 00b，因为 TI0p 输入引脚不使用。
10:8	STS[2:0]	100b	开始触发选择 1 0 0:选择主通道的 TAU0_TMI0n
11	— (p = 5)	0	固定为 0（通道 5）
	SPLIT (p = 1, 3)	0	SPLIT 位（通道 1、3）设置: 0: 1 6 位定时器模式
	MASTER (p = 2, 4, 6)	0	主控位设置（通道 2、4、6） 0: 从属通道
12	CCS	0	计数时钟选择 0: 选择操作时钟（f _{MCK} ）
13	—	0	固定值为0。
15:14	CKS[1:0]	00b 或 10b	选择工作时钟(f _{MCK})（设置与主通道相同。） 0 0: 选择 CK00 作为通道 p 的工作时钟 1 0: 选择 CK01 作为通道 p 的工作时钟

表 16.77 当使用多 PWM 输出功能时，从通道的 TMR0q 设置示例（针对两种类型的 PWM 输出）（1/2）

少量	象征	设置值	功能
0	OPIRQ	1	操作期间启动触发器 1: 触发输入有效
3:1	MD[2:0]	100b	通道 q 的工作模式 1 0 0: 单次计数模式
5:4	—	00b	固定为 0
7:6	CIS[1:0]	00b	选择TI0q引脚输入边沿 0 0: 设置为 00b，因为 TI0q 输入引脚不使用。
10:8	STS[2:0]	100b	开始触发选择 1 0 0:选择主通道的 TAU0_TMI0n
11	— (q = 5, 7)	0	固定为 0（通道 5、7）
	SPLIT (q = 3)	0	SPLIT 位（通道 3）设置为 0: 16 位定时器模式
	MASTER (q = 2, 4, 6)	0	主控位设置（通道 2、4、6） 0: 从属通道
12	CCS	0	计数时钟选择 0: 选择操作时钟(f _{MCK})

Table 16.77 Example of TMR0q settings for the slave channel when the multiple PWM output function is to be used (for two types of PWM output) (2 of 2)

Bit	Symbol	Set value	Function
13	—	0	Fixed to 0.
15:14	CKS[1:0]	00b or 10b	Selection of the operating clock (f_{MCK}) (make the same setting as master channel.) 0 0: Selects CK00 as the operating clock for channel q 1 0: Selects CK01 as the operating clock for channel q

Table 16.78 Example of TO0 settings for the slave channel when the multiple PWM output function is to be used (for two types of PWM output)

Bit	Symbol	Set value	Function
p	TO[p]	1/0	Timer output of channel p 0: Outputs 0 from TO0p. 1: Outputs 1 from TO0p.
q	TO[q]	1/0	Timer output of channel q 0: Outputs 0 from TO0q. 1: Outputs 1 from TO0q.

Table 16.79 Example of TOE0 settings for the slave channel when the multiple PWM output function is to be used (for two types of PWM output)

Bit	Symbol	Set value	Function
p	TOE[p]	1/0	Enabling or disabling timer output for channel p 0: Stops the TO0p output operation by counting operation. 1: Enables the TO0p output operation by counting operation.
q	TOE[q]	1/0	Enabling or disabling timer output for channel q 0: Stops the TO0q output operation by counting operation. 1: Enables the TO0q output operation by counting operation.

Table 16.80 Example of TOL0 settings for the slave channel when the multiple PWM output function is to be used (for two types of PWM output)

Bit	Symbol	Set value	Function
p	TOL[p]	1/0	Control of timer output of channel p 0: Positive logic output (active-high) 1: Negative logic output (active-low)
q	TOL[q]	1/0	Control of timer output of channel q 0: Positive logic output (active-high) 1: Negative logic output (active-low)

Table 16.81 Example of TOM0 settings for the slave channel when the multiple PWM output function is to be used (for two types of PWM output)

Bit	Symbol	Set value	Function
p	TOM[p]	1	Control of timer output mode of channel p (channels 1 to 6) 1: Sets the slave channel output mode.
q	TOM[q]	1	Control of timer output mode of channel q (channels 2 to 7) 1: Sets the slave channel output mode.

Table 16.82 show procedure for operations when the PWM function is to be used.

Table 16.77 当使用多PWM输出功能时，从通道的TMR0q设置示例（适用于两种类型的PWM输出）（2/2）

少量	象征	设置值	功能
13	—	0	固定值为0。
15:14	CKS[1:0]	00b 或 10b	选择工作时钟(f_{MCK})（设置与主通道相同。） 0 0: 选择 CK00 作为通道 q 的工作时钟 1 0: 选择 CK01 作为通道 q 的工作时钟

表 16.78 当使用多路 PWM 输出功能时，从通道的 TO0 设置示例（针对两种类型的 PWM 输出）

少量	象征	设置值	功能
p	TO[p]	1/0	通道 p 的定时器输出 0: 从 TO0p 输出 0。1: 从 TO0p 输出 1。
q	TO[q]	1/0	通道 q 的定时器输出 0: 从 TO0q 输出 0。1: 从 TO0q 输出 1。

表 16.79 当使用多 PWM 输出功能时，从通道的 TOE0 设置示例（针对两种类型的 PWM 输出）

少量	象征	设置值	功能
p	TOE[p]	1/0	启用或禁用通道 p 0 的定时器输出：通过计数操作停止 TO0p 输出操作。 1: 通过计数操作启用 TO0p 输出操作。
q	TOE[q]	1/0	启用或禁用通道 q 0 的定时器输出：通过计数操作停止 TO0q 输出操作。 1: 通过计数操作启用 TO0q 输出操作。

表 16.80 当使用多 PWM 输出功能时，从通道的 TOL0 设置示例（针对两种类型的 PWM 输出）

少量	象征	设置值	功能
p	TOL[p]	1/0	控制通道 p 的定时器输出 0: 正逻辑输出（高电平有效） 1: 负逻辑输出（低电平有效）
q	TOL[q]	1/0	控制通道 q 的定时器输出 0: 正逻辑输出（高电平有效） 1: 负逻辑输出（低电平有效）

表 16.81 当使用多 PWM 输出功能时，从通道的 TOM0 设置示例（针对两种类型的 PWM 输出）

少量	象征	设置值	功能
p	TOM[p]	1	控制通道 p（通道 1 至 6）的定时器输出模式 1: 设置从通道输出模式。
q	TOM[q]	1	控制通道 q（通道 2 至 7）的定时器输出模式 1: 设置从通道输出模式。

表 16.82 显示了使用 PWM 功能时的操作步骤。

Table 16.82 Procedure for operations when the multiple PWM output function is to be used (for two types of PWM output) (1 of 2)

	Step	Software operation	Hardware state
TAU default setting		—	Power-off state (Clock supply is stopped and writing to each register is disabled.)
	<1>	Sets the MSTPD0 bit of Module Stop Control Register D (MSTPCRD) to 0.	→ Power-on state. Each channel stops operating. (Clock supply is started and writing to each register is enabled.)
	<2>	Sets timer clock select register 0 (TPS0). Determines clock frequencies of CK00 and CK01.	—
Channel default setting	<3>	Sets timer mode register 0n, 0p, 0q (TMR0n, TMR0p, TMR0q) of each channel to be used (determines operation mode of channels). An interval (period) value is set to timer data register 0n (TDR0n) of the master channel, and a duty factor is set to the TDR0p and TDR0q registers of the slave channels.	Channel stops operating. (Clock is supplied and some power is consumed.)
	<4>	Sets slave channels. The TOM[p] and TOM[q] bits of timer output mode register 0 (TOM0) are set to 1 (slave channel output mode). Sets the TOL0.TOL[p] and TOL[q] bits and determines default level of the TO0p and TO0q outputs. Sets the TOE0.TOE[p] and TOE[q] bits to 1 and enables operation of TO0p and TO0q. Sets the Prs Direction Register (PDRr) to 1.	→ The TO0p and TO0q pins go into Hi-Z output state. → The TO0p and TO0q default setting levels are output when Prs Direction Register (PDRr) is in output mode. → TO0p and TO0q do not change because channels stop operating. → The TO0p and TO0q pins output the TO0p and TO0q set levels.
Operation start	<5>	(Sets the TOE0.TOE[p], TOE[q] (slave) bits to 1 only when resuming operation.) The TS[n] bit (master), and TS[p], TS[q] (slave) bits of timer channel start register 0 (TS0) are set to 1 at the same time. The TS0.TS[n], TS[p], and TS[q] bits automatically return to 0 because they are trigger bits.	→ TE0.TE[n] = 1, TE0.TE[p], TE[q] = 1 When the master channel starts counting, TAU0_TMI0n is generated. Triggered by this interrupt, the slave channel also starts counting.
During operation	<6>	Set values of the TMR0n, TMR0p, TMR0q registers, TOM0.TOM[n], TOM[p], TOM[q], OL0.TOL[n], TOL[p], and TOL[q] bits cannot be changed. Set values of the TDR0n, TDR0p and TDR0q registers can be changed after TAU0_TMI0n of the master channel is generated. The TCR0n, TCR0p and TCR0q registers can always be read. The TSR0n, TSR0p and TSR0q registers are not used.	The counter of the master channel loads the TDR0n register value to timer counter register 0n (TCR0n) and counts down. When the count value reaches TCR0n = 0x0000, TAU0_TMI0n output is generated. At the same time, the value of the TDR0n register is loaded to the TCR0n register, and the counter starts counting down again. At the slave channel 1, the values of the TDR0p register are transferred to the TCR0p register, triggered by TAU0_TMI0n of the master channel, and the counter starts counting down. The output levels of TO0p become active one count clock after generation of the TAU0_TMI0n output from the master channel. It becomes inactive when TCR0p = 0x0000, and the counting operation is stopped. At the slave channel 2, the values of the TDR0q register are transferred to TCR0q register, triggered by TAU0_TMI0n of the master channel, and the counter starts counting down. The output levels of TO0q become active one count clock after generation of the TAU0_TMI0n output from the master channel. It becomes inactive when TCR0q = 0x0000, and the counting operation is stopped. After that, the above operation is repeated.

表 16.82 使用多路 PWM 输出功能时的操作步骤（针对两种类型的 PWM 输出）（1/2）

	步	软件操作	硬件状态
TAU 默认设置		—	关机状态 (时钟信号停止，所有寄存器的写入操作均被禁用。)
	<1>	将模块停止控制寄存器 D (MSTPCRD) 的 MSTPD0 位设置为 0。	→ 上电状态。各通道停止工作。 (时钟电源已启动，并且对每个寄存器的写入操作已启用。)
	<2>	设置定时器时钟选择寄存器 0 (TPS0)。 确定 CK00 和 CK01 的时钟频率。	—
通道默认设置	<3>	设置每个通道的定时器模式寄存器 0n、0p、0q (TMR0n、TMR0p、TMR0q) 以决定通道的操作模式。 将间隔（周期）值设置到主通道的定时器数据寄存器 0n (TDR0n)，并将占空比设置到从通道的 TDR0p 和 TDR0q 寄存器。	频道停止运行。 (时钟信号被提供，并消耗一定功率。)
	<4>	设置从属通道。 定时器输出模式寄存器 0 (TOM0) 的 TOM[p] 和 TOM[q] 位设置为 1（从通道输出模式）。 设置 TOL0.TOL[p] 和 TOL[q] 位，并确定 TO0p 和 TO0q 输出的默认设置级别。 将 TOE0.TOE[p] 和 TOE[q] 位设置为 1，并使能 TO0p 和 TO0q 的操作。 将 PRS 方向寄存器 (PDRr) 设置为 1。	→ TO0p 和 TO0q 引脚进入高阻抗输出状态。 → 当 Prs 方向寄存器 (PDRr) 处于输出模式时，将输出 TO0p 和 TO0q 的默认设置级别。 → TO0p 和 TO0q 不会改变，因为通道停止运行。 → TO0p 和 TO0q 引脚输出 TO0p 和 TO0q 的设定电平。
操作开始	<5>	（仅在恢复操作时，将 TOE0.TOE[p]、TOE[q]（从机）位设置为 1。）定时器通道起始寄存器 0 (TS0) 的 TS[n] 位（主机）和 TS[p]、TS[q]（从机）位同时设置为 1。 TS0.TS[n]、TS[p] 和 TS[q] 位会自动返回 0，因为它们是触发位。	→ TE0.TE[n] = 1, TE0.TE[p], TE[q] = 1 当主通道开始计数时，会生成 TAU0_TMI0n 中断。受此中断触发，从通道也开始计数。
运行期间	<6>	TMR0n、TMR0p、TMR0q 寄存器、TOM0.TOM[n]、TOM[p]、TOM[q]、OL0.TOL[n]、TOL[p] 和 TOL[q] 位的值不能更改。 主通道的 TAU0_TMI0n 生成后，可以更改 TDR0n、TDR0p 和 TDR0q 寄存器的设置值。 TCR0n、TCR0p 和 TCR0q 寄存器始终可以读取。 TSR0n、TSR0p 和 TSR0q 寄存器未使用。	主通道的计数器将 TDR0n 寄存器的值加载到定时器计数器寄存器 0n (TCR0n) 并开始倒计时。当计数达到 TCR0n = 0x0000 时，产生 TAU0_TMI0n 输出。同时，将 TDR0n 寄存器的值加载到 TCR0n 寄存器，计数器再次开始倒计时。 在从通道 1，主通道的 TAU0_TMI0n 触发后，TDR0p 寄存器的值被传输到 TCR0p 寄存器，计数器开始倒计时。主通道的 TAU0_TMI0n 输出产生后一个时钟周期，TO0p 的输出电平变为有效。当 TCR0p = 0x0000 时，TO0p 的输出电平变为无效，计数操作停止。 在从通道 2，主通道的 TAU0_TMI0n 触发后，TDR0q 寄存器的值被传输到 TCR0q 寄存器，计数器开始倒计时。主通道的 TAU0_TMI0n 输出产生后一个时钟周期，TO0q 的输出电平变为有效。当 TCR0q = 0x0000，为无效时，TO0q 的输出电平变为无效，计数操作停止。 之后，重复上述操作。

Table 16.82 Procedure for operations when the multiple PWM output function is to be used (for two types of PWM output) (2 of 2)

	Step	Software operation		Hardware state
Operation stop	<7>	The TT0.TT[n] bit (master), TT[p], and TT[q] (slave) bits are set to 1 at the same time. The TT0.TT[n], TT[p] and TT[q] bits automatically return to 0 because they are trigger bits.	→	TE0.TE[n], TE[p], TE[q] = 0, and count operation stops. The TCR0n, TCR0p and TCR0q registers hold count value and stop. The TO0p and TO0q outputs are not initialized and retain their current states.
	<8>	The TOE0.TOE[p], and TOE[q] bits of slave channels are cleared to 0 and value is set to the TO0.TO[p] and TO[q] bits. To resume operation, go to step <5>. To terminate the operation, go to step <9>	→	The TO0p and TO0q pins output the TO0p and TO0q set levels.
TAU stop	<9>	To hold the TO0p pin and TO0q pin output levels Set PSEL[2:0] bits to 000b after the value to be held is set to the Prs Output Data Register (PODRr). When holding the TO0p pin output level is not necessary Setting not required.	→	The TO0p pin and TO0q pin output levels are held by port function
	<10>	Sets the MSTPD0 bit of Module Stop Control Register D (MSTPCRD) to 1.	→	This stops supply of the input clock to timer array unit 0. Power-off state.

Note: n = 0, 2, 4 (Master channel number)
p, q: Slave channel number
n < p < q ≤ 7 (Where p and q are integers greater than n)
r = 0 to 9
s = 00 to 15

16.9 Usage Notes

16.9.1 Cautions when Using Timer Output

Pins may be assigned multiplexed timer output and other alternate functions. The assignment depends on the product. If you intend to use a timer output, set the outputs from all other multiplexed pin functions to their initial values.

For details, see [section 15, I/O Ports](#).

表 16.82 使用多路 PWM 输出功能时的操作步骤（针对两种类型的 PWM 输出）（2/2）

	步	软件操作		硬件状态
停止操作	<7>	TT0.TT[n] 位（主设备）、TT[p]和 TT[q] 位（从设备）同时设置为 1。 TT0.TT[n]、TT[p] 和 TT[q] 位会自动返回 0，因为它们是触发位。	→	TE0.TE[n], TE[p],TE[q] = 0，计数操作停止。TCR0n、TCR0p 和 TCR0q 寄存器保存计数值并停止。 TO0p 和 TO0q 输出未初始化，并保留其当前状态。
	<8>	从通道的 TOE0.TOE[p]和 TOE[q] 位被清除为 0，并将值设置为 TO0.TO[p]和 TO[q] 位。 要恢复操作，请转到步骤<5>。 要终止操作，请转到步骤<9>	→	TO0p 和 TO0q 引脚输出 TO0p 和 TO0q 的设定电平。
停止！	<9>	要保持 TO0p 引脚和 TO0q 引脚的输出电平，请在将要保持的值设置到 Prs 输出数据寄存器 (PODRr) 之后，将 PSEL[2:0] 位设置为 000b。保持 TO0p 引脚的输出电平时，无需这样做。 无需设置。	→	TO0p 引脚和 TO0q 引脚的输出电平由端口功能保持
	<10>	将模块停止控制寄存器 D (MSTPCRD) 的 MSTP D0 位设置为 1。	→	这将停止向定时器阵列单元 0 提供输入时钟。 关机状态。

注：n = 0、2、4（主通道号）
p、q：从属通道号
n < p < q ≤ 7 （其中p和 q 是大于 n 的整数）
r = 0 to 9
s = 00 to 15

16.9 使用说明

16.9.1 使用定时器输出时的注意事项

引脚可以分配多路复用定时器输出和其他备用功能。具体分配取决于产品。如果您打算使用定时器输出，请将所有其他多路复用引脚功能的输出设置为其初始值。

有关详细信息，请参阅[15, I/O 端口](#)部分。

17. 32-bit Interval Timer (TML32)

17.1 Overview

The 32-bit interval timer is made up of four 8-bit interval timers (referred to as channels 0 to 3). Each is capable of operating independently and in that case they all have the same functions. Two 8-bit interval timer channels can be connected to operate as a 16-bit interval timer. Four 8-bit interval timer channels can be connected to operate as a 32-bit interval timer.

The 32-bit interval timer operates with the HOCO, LOCO, which is asynchronous with the CPU operation. Table 17.1 lists the 32-bit interval timer functions. Figure 17.1 to Figure 17.4 show images of each timer function, and Figure 17.5 shows a block diagram of the 32-bit interval timer.

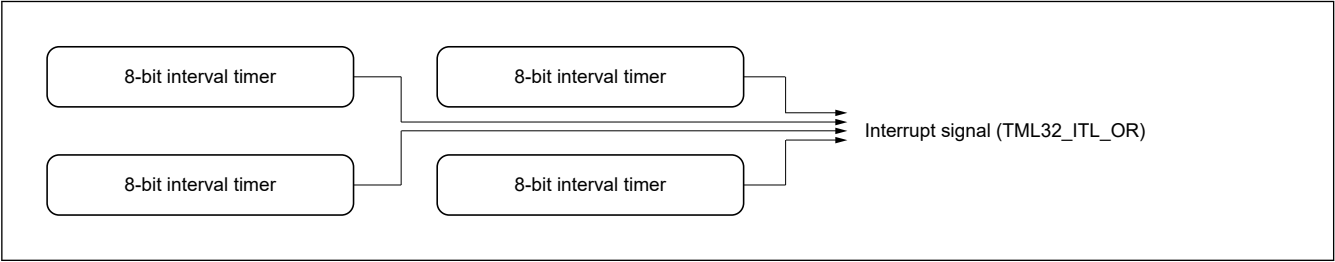


Figure 17.1 Image of four 8-bit interval timer function

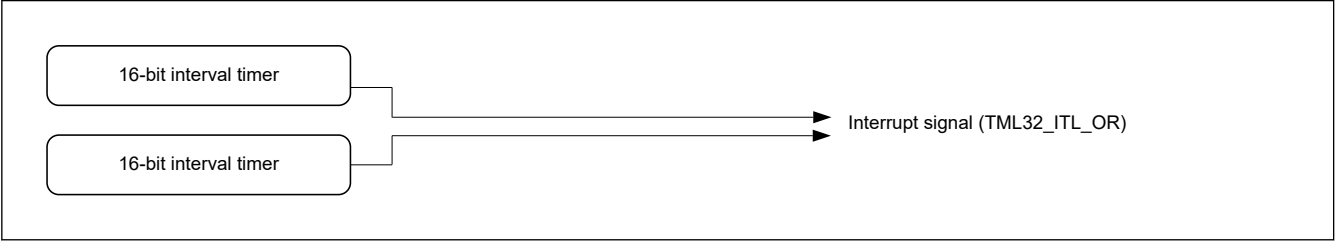


Figure 17.2 Image of two 16-bit Interval timer function

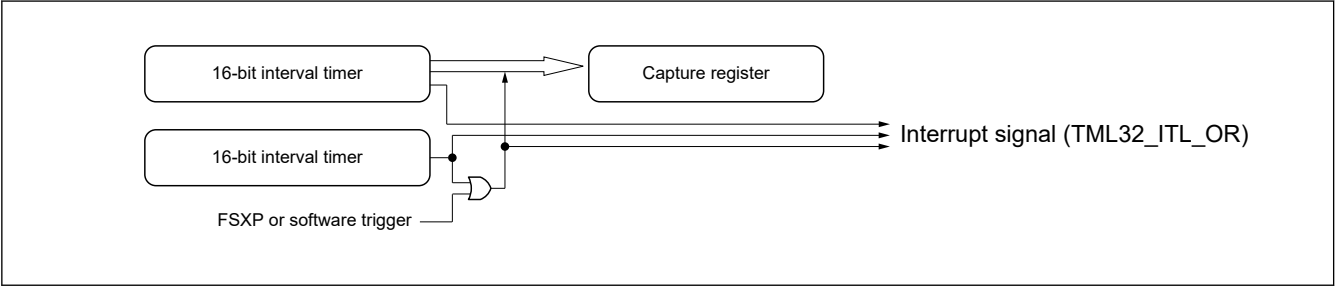


Figure 17.3 Image of the 16-bit interval timer and 16-bit capture function

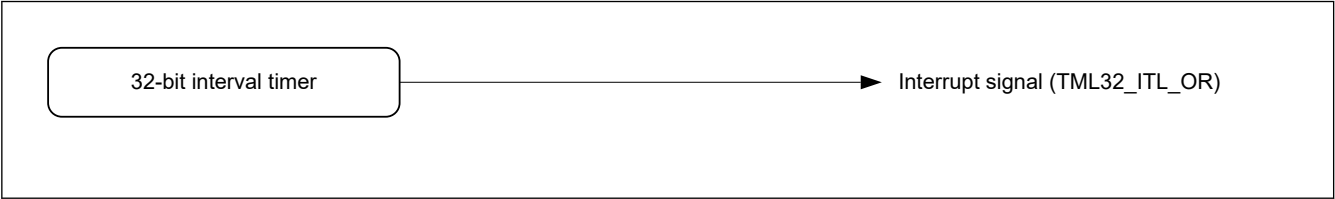


Figure 17.4 Image of a 32-bit interval timer function

Table 17.1 Specifications of 32-bit interval timer operations (1 of 2)

Item	Description
Count source (operating clock)	• HOCO • LOCO*1

17.32 位间隔定时器 (TML32)

17.1 概述

32 位间隔定时器由四个 8 位间隔定时器（称为通道 0 到 3）组成。每个定时器都可以独立运行，此时它们的功能相同。两个 8 位间隔定时器通道可以连接起来，组成一个 16 位间隔定时器。四个 8 位间隔定时器通道可以连接起来，组成一个 32 位间隔定时器。

32 位间隔定时器与 HOCO 和 LOCO 配合工作，与 CPU 操作异步。表 17.1 列出了 32 位间隔定时器的功能。图 17.1 至图 17.4 分别展示了每个定时器功能的图像，图 17.5 则展示了 32 位间隔定时器的框图。

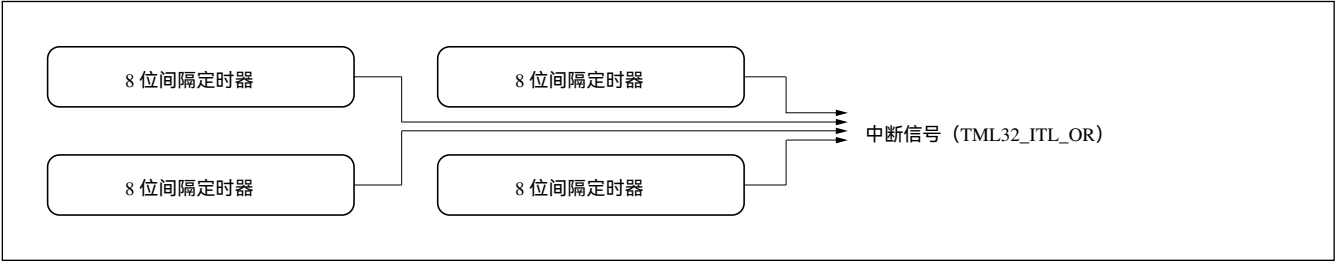


图 17.1 四个 8 位间隔定时器函数的图像

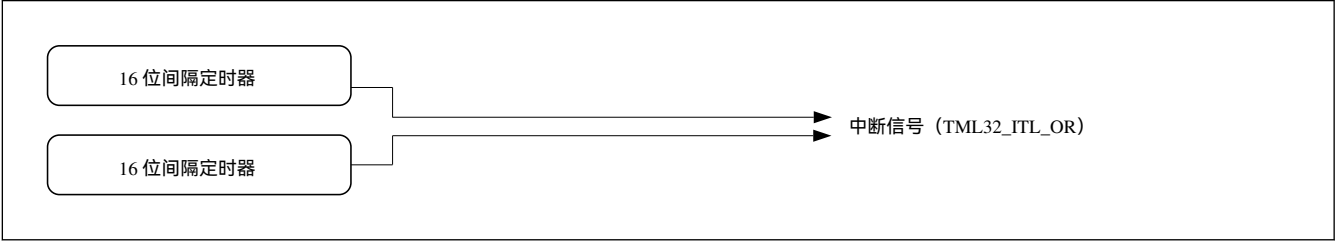


图 17.2 两个 16 位间隔定时器函数的图像

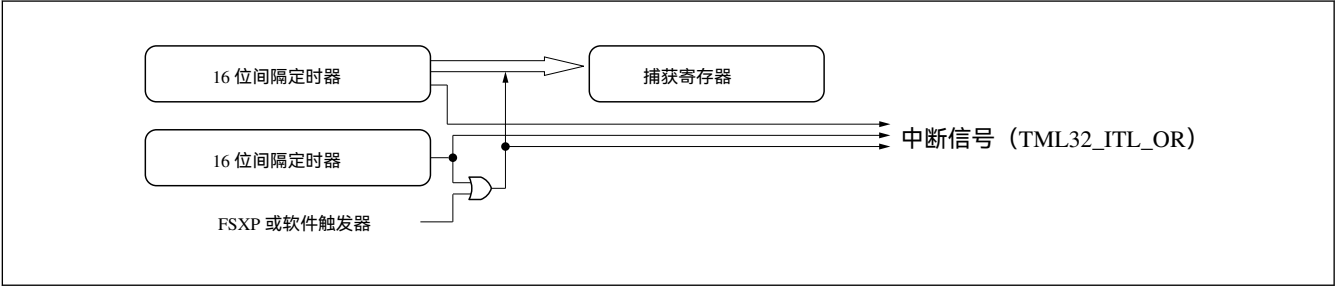


图 17.3 16 位间隔定时器和 16 位捕获功能的图像

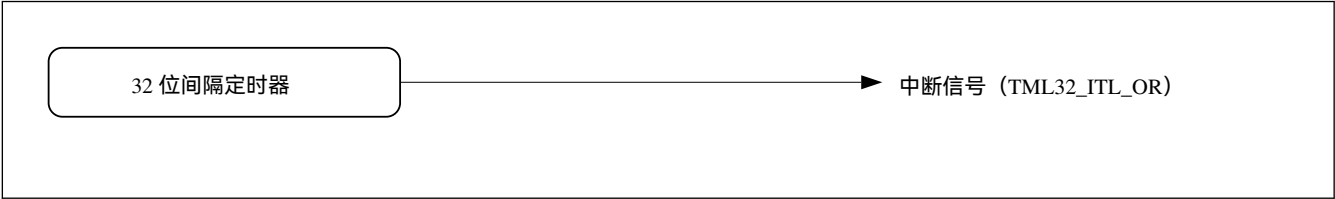


图 17.4 32 位间隔定时器函数的图像

表 17.1 32 位间隔定时器操作规范 (1/2)

物品	描述
计数源 (运行时钟)	• HOCO ● 位置*1

Table 17.1 Specifications of 32-bit interval timer operations (2 of 2)

Item	Description
Capture clock (Selectable sources for counting by the timer which can generate a capture trigger)	• HOCO • LOCO*1
Frequency division ratio	• 1/1, 1/2, 1/4, 1/8, 1/16, 1/32, 1/64, 1/128
Operating mode	• 8-bit counter mode Channels 0 to 3 independently operate as 8-bit counters. • 16-bit counter mode The combinations of channels 0 and 1 and channels 2 and 3 are cascade-connectable to operate as two 16-bit counters. • 32-bit counter mode Channels 0 to 3 are connected to operate as a 32-bit counter. • 16-bit capture mode Channels 0 and 1 are connected to operate as a 16-bit counter using the count source, channels 2 and 3 are connected to operate as a 16-bit counter using the capture clock, and the connected counters are used for capture operation.
Interrupt	• Five interrupt sources are integrated into one interrupt signal and output as the TML32_ITL_OR signal. – Output when the counter value in any of channels 0 to 3 matches the compare value. – Output when the capturing of the counter value is completed in capture mode.

Note 1. Select LOCO as the FSXP by setting the OSMC.WUTMMCK0 bit.

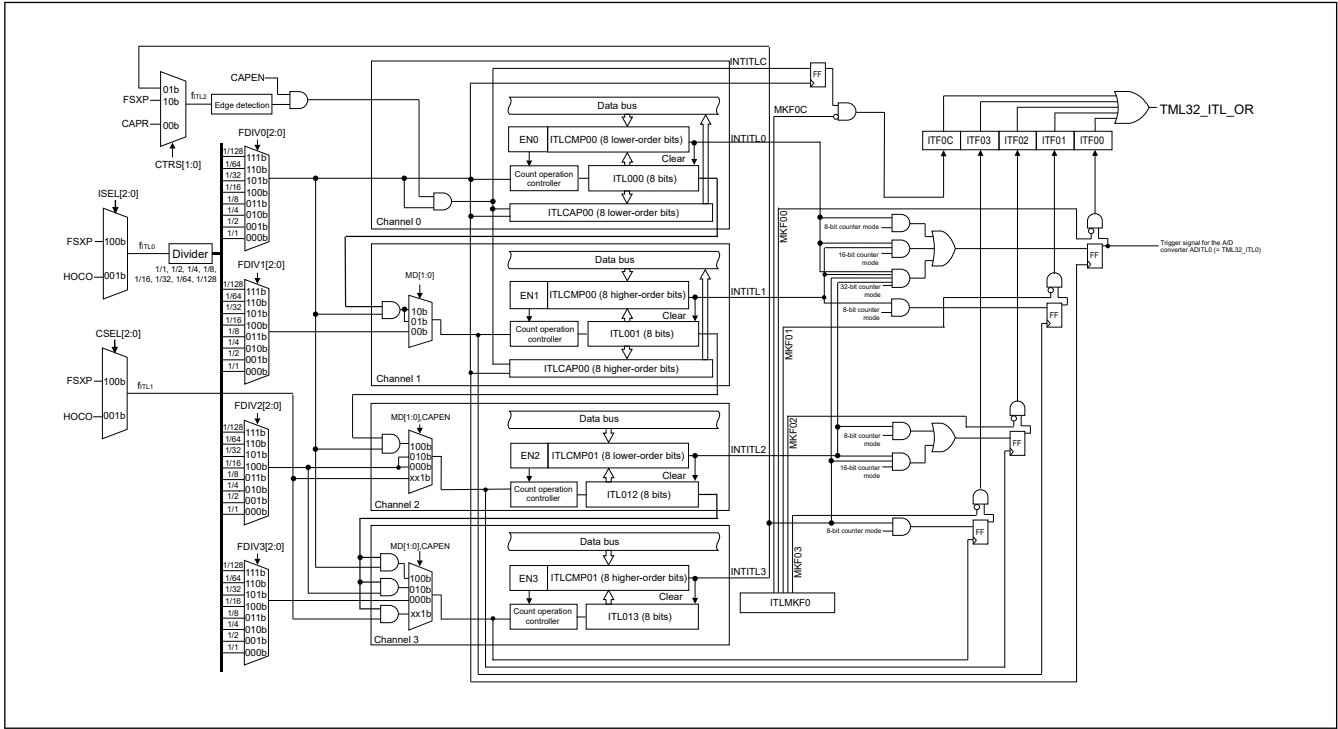


Figure 17.5 Block diagram of 32-bit interval timer

ITL000, ITL001, ITL012, ITL013: 8-bit counters

Note: In 16-bit counter mode, the counters in channels 0 and 1 are connected (ITL000 + ITL001) and the counters in channels 2 and 3 are connected (ITL012 + ITL013).
In 32-bit counter mode, the counters in channels 0 to 3 are connected (ITL000 + ITL001 + ITL012 + ITL013).

表 17.1 32 位间隔定时器操作规范 (2/2)

物品	描述
捕获时钟（定时器计数的可选源，可生成捕获触发信号）	• HOCO • 位置*1
分频比	• 1/1、1/2、1/4、1/8、1/16、1/32、1/64、1/128
操作模式	• 8位计数器模式 通道 0 至 3 独立作为 8 位计数器运行。 • 16位计数器模式 通道 0 和 1 以及通道 2 和 3 可以级联连接，作为两个 16 位计数器运行。 • 32位计数器模式 通道 0 至 3 连接在一起，用作 32 位计数器。 • 16 位捕获模式 通道 0 和 1 连接起来，使用计数源作为 16 位计数器运行；通道 2 和 3 连接起来，使用捕获时钟作为 16 位计数器运行；连接的计数器用于捕获操作。
打断	• 五个中断源集成到一个中断信号中，并作为 TML32_ITL_OR 信号输出。 – 当通道 0 到 3 中的任何一个通道中的计数器值与比较值匹配时输出。 – 在捕获模式下完成计数器值捕获时输出。

注 1. 通过设置 OSMC.WUTMMCK0 位，选择 LOCO 作为 FSXP。

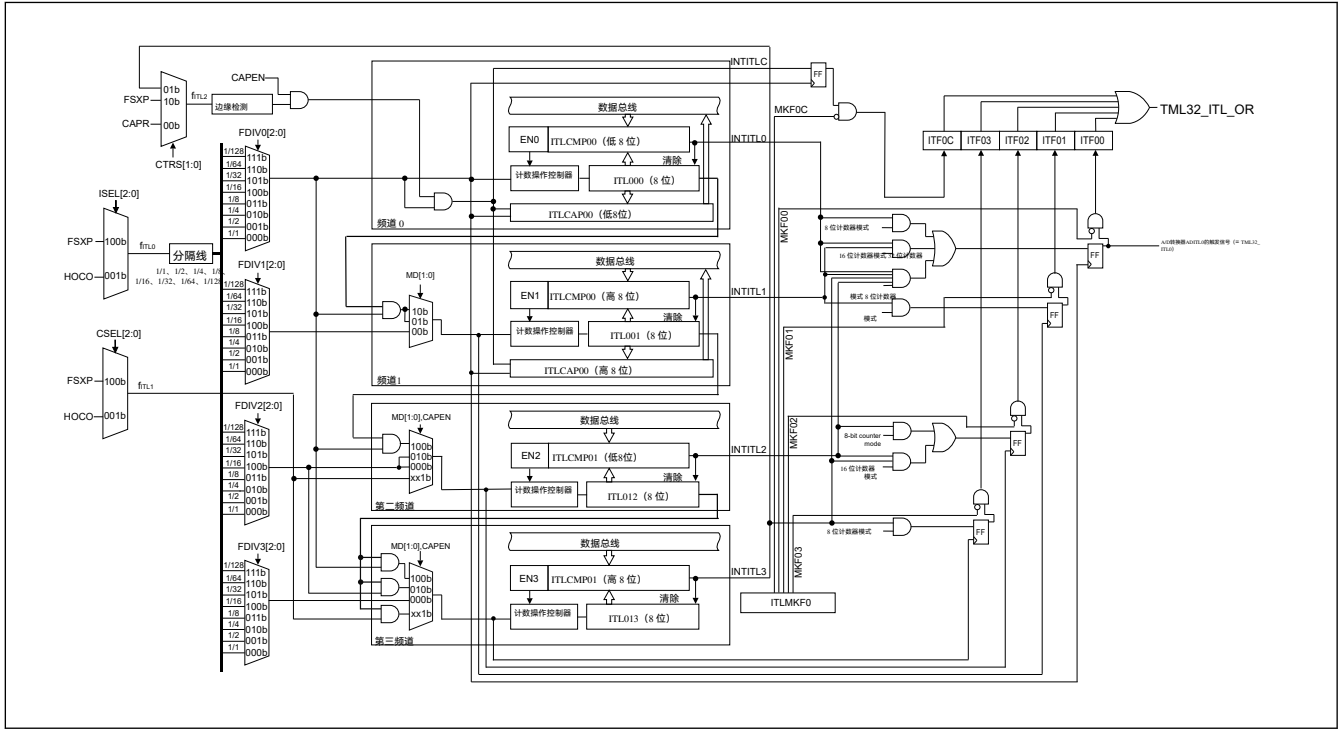


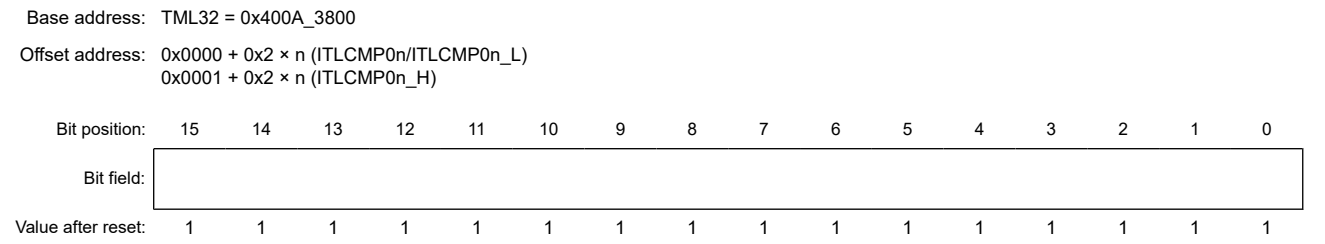
图 17.5 32 位间隔定时器的框图

ITL000、ITL001、ITL012、ITL013：8 位计数器

注意：在 16 位计数器模式下，通道 0 和 1 中的计数器连接为(ITL000 + ITL001)，通道 2 和 3 中的计数器连接为(ITL012 + ITL013)。
In 32-bit counter mode, the counters in channels 0 to 3 are connected (ITL000 + ITL001 + ITL012 + ITL013).

17.2 Register Descriptions

17.2.1 ITLCMP0n/ITLCMP0n_L/ITLCMP0n_H : Interval Timer Compare Registers 0n (n = 0, 1)



Bit	Symbol	Function	R/W
15:0	n/a	Comparison Data of 16-bit Counter (ITL0n) and 8-bit Counter (ITL0n_H/ITL0n_L) are Stored The ITLCMP00 is compared with ITL00 (ITL000 + ITL001). The ITLCMP01 is compared with ITL01 (ITL012 + ITL013).	R/W

Note: Write to the ITLCMP0n_H and ITLCMP0n_L registers while the settings of the EN0 to EN3 bits in the ITLCTL0 register are 0, respectively.

Note: Write to the ITLCMP00 register while the IEN0 bit in the ITLCTL0 register is 0. Write to the ITLCMP01 register while the EN2 bit in the ITLCTL0 register is 0 in 16-bit counter mode or while the EN0 bit in the ITLCTL0 register is 0 in 32-bit counter mode.

Interval Timer Compare Registers (ITLCMP0n/ITLCMP0n_L/ITLCMP0n_H) is compare value registers used in 8-bit, 16-bit, or 32-bit counter mode. ITLCMP0n_L (ITLCMP0n[7:0]) and ITLCMP0n_H (ITLCMP0n[15:8]) used in 8-bit counter mode.

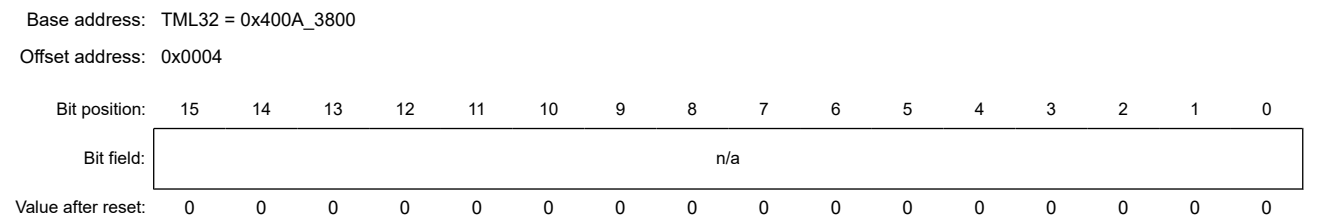
A value from 0x0001 to 0xFFFF can be specified. Setting these registers to 0x0000 is prohibited.

These registers hold values to be compared with the ITL0n counter values.

When the ITLCTL0.MD[1:0] bits are set to 10b, these registers are used as compare registers in 32-bit counter mode.

Specify the upper 16-bit compare value in the ITLCMP01 register and the lower 16-bit compare value in the ITLCMP00 register.

17.2.2 ITLCAP00 : Interval Timer Capture Register 00



Bit	Symbol	Function	R/W
15:0	n/a	Capture Result of 16-bit Counter (ITL00) is Stored.	R

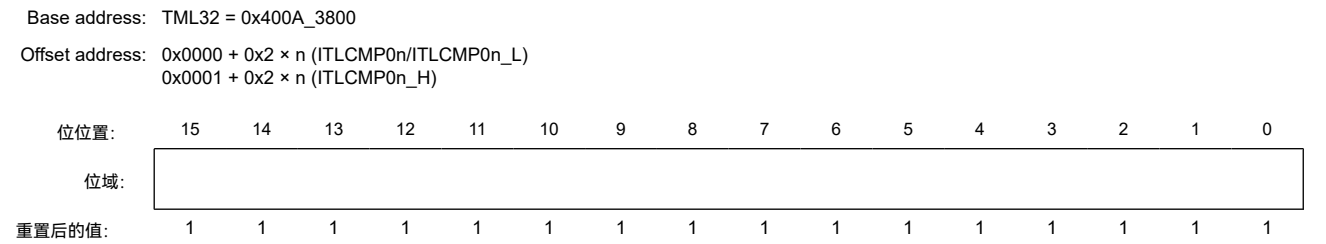
This register holds 16-bit captured values when the interval timers are operating in 16-bit capture mode.

The values of the 16-bit counters (ITL000 + ITL001) are stored in the ITLCAP00 register in response to the capture trigger selected in the ITLCC0 register when the CAPEN bit in the ITLCC0 register is 1.

When an interrupt on compare match with the ITLCMP01 register is to be used, select the counter clock in the ITLCSEL0 register and set the comparison value in the ITLCMP01 register.

17.2 寄存器描述

17.2.1 ITLCMP0n/ITLCMP0n_L/ITLCMP0n_H: 间隔定时器比较寄存器 0n(n = 0, 1)



少量	象征	功能	R/W
15:0	n/a	存储 16 位计数器 (ITL0n) 和 8 位计数器 (ITL0n_H/ITL0n_L) 的比较数据。将 ITLCMP00 与 ITL00 进行比较 ((ITL000 + ITL001)。将 ITLCMP01 与 ITL01 进行比较 (ITL012+ ITL013)。	R/W

注意：当 ITLCTL0 寄存器中的 EN0 到 EN3 位分别设置为 0 时，写入 ITLCMP0n_H 和 ITLCMP0n_L 寄存器。

注意：当 ITLCTL0 寄存器中的 IEN0 位为 0 时，写入 ITLCMP00 寄存器。在 16 位计数器模式下，当 ITLCTL0 寄存器中的 EN2 位为 0 时，或当 ITLCTL0 寄存器中的 EN0 位为 0 时，写入 ITLCMP01 寄存器。

间隔定时器比较寄存器（ITLCMP0n/ITLCMP0n_L/ITLCMP0n_H）是用于 8 位、16 位或 32 位计数器模式的比较值寄存器。ITLCMP0n_L（ITLCMP0n[7:0]）和 ITLCMP0n_H（ITLCMP0n[15:8]）用于 8 位计数器模式。

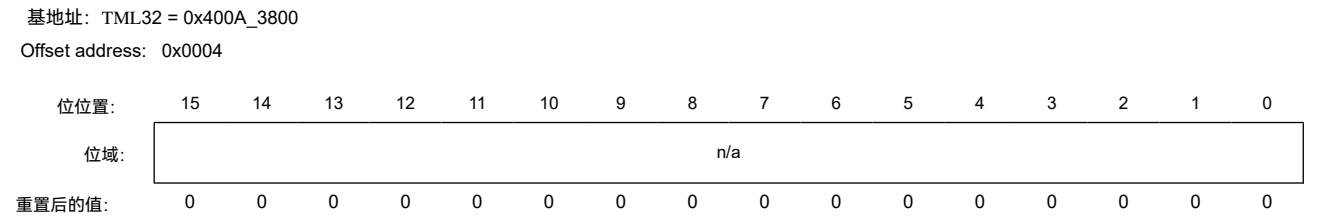
可以指定从0x0001 到0xFFFF之间的值。禁止将这些寄存器设置为0x0000。

这些寄存器保存着要与 ITL0n 计数器值进行比较的值。

当 ITLCTL0.MD[1:0] 位设置为 10b 时，这些寄存器在 32 位计数器模式下用作比较寄存器。

在 ITLCMP01 寄存器中指定高 16 位比较值，在 ITLCMP00 寄存器中指定低 16 位比较值。

17.2.2 ITLCAP00: 间隔定时器捕获寄存器 00



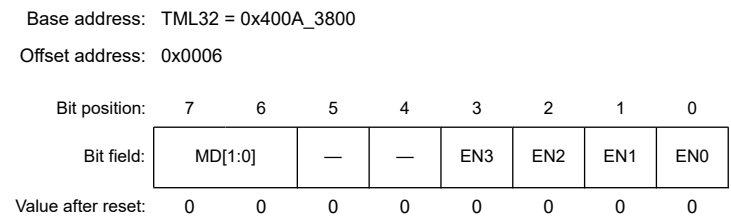
少量	象征	功能	R/W
15:0	n/a	16 位计数器 (ITL00) 的捕获结果已存储。	R

当间隔定时器以 16 位捕获模式运行时，此寄存器保存 16 位捕获值。

当 ITLCC0 寄存器中的 CAPEN 位为 1 时，响应于 ITLCC0 寄存器中选择的捕获触发，16 位计数器(ITL000 + ITL001) 的值存储在 ITLCAP00 寄存器中。

当要使用与ITLCMP01 寄存器比较匹配的中断时，选择 ITLCSEL0 寄存器中的计数器时钟，并在 ITLCMP01 寄存器中设置比较值。

17.2.3 ITLCTL0 : Interval Timer Control Register



Bit	Symbol	Function	R/W
0	EN0	8-bit Counter Mode: ITL000 Count Enable*1 16-bit Counter Mode: ITL000 + ITL001 Count Enable*1 32-bit Counter Mode: ITL000 + ITL001 + ITL012 + ITL013 Count Enable*1 0: Counting stops 1: Counting begins	R/W
1	EN1	8-bit Counter Mode: ITL001 Count Enable*1 0: Counting stops 1: Counting begins	R/W
2	EN2	8-bit Counter Mode: ITL012 Count Enable*1 16-bit Counter Mode: ITL012 + ITL013 Count Enable*1 0: Counting stops 1: Counting begins	R/W
3	EN3	8-bit Counter Mode: ITL013 Count Enable*1 0: Counting stops 1: Counting begins	R/W
5:4	—	These bits are read as 0. The write value should be 0.	R/W
7:6	MD[1:0]	Selection of 8-bit, 16-bit, or 32-bit Counter Mode*2 0 0: The interval timer operates in 8-bit counter mode. 0 1: The interval timer operates in 16-bit counter mode (channel 0 is connected with channel 1 and channel 2 is connected with channel 3). 1 0: The interval timer operates in 32-bit counter mode (channels 0 to 3 are connected). 1 1: Setting prohibited.	R/W

Note 1. When one of the EN3 to EN0 bits is cleared to 0, the corresponding counter is cleared to 0 without synchronization with the counter clock.

Note 2. To change the timer mode, be sure to write to the MD[1:0] bits only while the EN0, EN1, EN2, and EN3 bits are all 0.

This register is used to start or stop counting by the interval timer and to select 8-bit, 16-bit, or 32-bit counter mode.

EN0 bit (8-bit Counter Mode: ITL000 Count Enable, 16-bit Counter Mode: ITL000 + ITL001 Count Enable, 32-bit Counter Mode: ITL000 + ITL001 + ITL012 + ITL013 Count Enable)

In 8-bit counter mode, writing 1 to this bit starts up-counting in the ITL000 counter and writing 0 stops it.

In 16-bit counter mode, writing 1 to this bit starts up-counting in the ITL000 + ITL001 counter and writing 0 stops it.

In 32-bit counter mode, writing 1 to this bit starts up-counting in the ITL000 + ITL001 + ITL012 + ITL013 counter and writing 0 stops it.

EN1 bit (8-bit Counter Mode: ITL001 Count Enable)

In 8-bit counter mode, writing 1 to this bit starts up-counting in the ITL001 counter and writing 0 stops it.

In 16-bit counter mode, set this bit to 0.

In 32-bit counter mode, set this bit to 0.

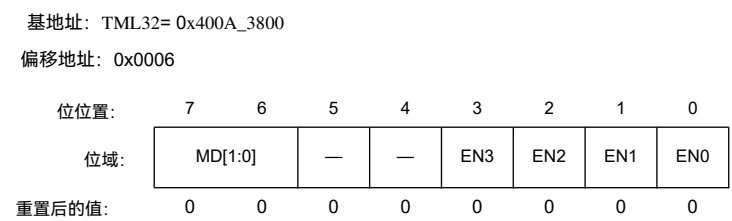
EN2 bit (8-bit Counter Mode: ITL012 Count Enable, 16-bit Counter Mode: ITL012 + ITL013 Count Enable)

In 8-bit counter mode, writing 1 to this bit starts up-counting in the ITL012 counter and writing 0 stops it.

In 16-bit counter mode, writing 1 to this bit starts up-counting in the ITL012 + ITL013 counter and writing 0 stops it.

In 32-bit counter mode, set this bit to 0.

17.2.3 ITLCTL0: 间隔定时器控制寄存器



少量	象征	功能	R/W
0	EN0	8 位计数器模式: ITL000 计数使能*1 16 位计数器模式: ITL000+ITL001 计数使能*1 32 位计数器模式: ITL000+ITL001+ITL012+ITL013 计数使能*1 0: 计数停止 1: 计数开始	R/W
1	EN1	8 位计数器模式: ITL001 计数使能*1 0: 计数停止 1: 计数开始	R/W
2	EN2	8 位计数器模式: ITL012 计数使能*1 16 位计数器模式: ITL012+ITL013 计数使能*1 0: 计数停止 1: 计数开始	R/W
3	EN3	8 位计数器模式: ITL013 计数使能*1 0: 计数停止 1: 计数开始	R/W
5:4	—	这些位读取为 0。写入值应为 0。	R/W
7:6	MD[1:0]	选择 8 位、16 位或 32 位计数器模式*20 0:间隔定时器以 8 位计数器模式运行。 0 1: 间隔定时器以 16 位计数器模式运行（通道 0 与通道 1 连接，通道 2 与通道 3 连接）。 1 0: 间隔定时器以 32 位计数器模式运行（通道 0 到 3 已连接）。 1 1: 禁止设置。	R/W

注1. 当EN3至EN0位中的一位被清除为0时，相应的计数器将被清除为0，而无需与计数器时钟同步。

注 2. 要更改定时器模式，请确保仅写入 MD[1:0] 位，同时 EN0、EN1、EN2 和 EN3 位均为 0。

该寄存器用于启动或停止间隔定时器的计数，以及选择 8 位、16 位或 32 位计数器模式。

EN0 位（8 位计数器模式: ITL000 计数使能，16 位计数器模式: ITL000 +ITL001 计数使能，32 位计数器模式: ITL000 + ITL001 + ITL012 + ITL013 C单位使能）

在 8 位计数器模式下，向此位写入 1 会开始在ITL000计数器中递增计数，写入 0 会停止计数。

In 16-bit counter mode, writing 1 to this bit starts up-counting in the ITL000 + ITL001 counter and writing 0 stops it.

在 32 位计数器模式下，向此位写入 1 会开始在 ITL000 + ITL001 + ITL012 + ITL013计数器中向上计数，写入 0 会停止计数。

EN1 位（8 位计数器模式: ITL001 计数使能）

在 8 位计数器模式下，向此位写入 1 会开始 ITL001 计数器的递增计数，写入 0 则会停止计数。

在 16 位计数器模式下，将此位设置为 0。

在 32 位计数器模式下，将此位设置为 0。

EN2 位（8 位计数器模式: ITL012 计数使能，16 位计数器模式: ITL012+ITL013 计数使能）

在 8 位计数器模式下，向此位写入 1 会开始 ITL012 计数器的递增计数，写入 0 则会停止计数。

在 16 位计数器模式下，向此位写入 1 会开始在ITL012 + ITL013 计数器中向上计数，写入 0 会停止计数。

在 32 位计数器模式下，将此位设置为 0。

EN3 bit (8-bit Counter Mode: ITL013 Count Enable)

In 8-bit counter mode, writing 1 to this bit starts up-counting in the ITL013 counter and writing 0 stops it.

In 16-bit counter mode, set this bit to 0.

In 32-bit counter mode, set this bit to 0.

MD[1:0] bits (Selection of 8-bit, 16-bit, or 32-bit Counter Mode)

Table 17.2 lists the target counters that can be enabled in the MD[1:0] bits and EN0 to EN3 bit settings.

Table 17.2 Target counter setting

Mode	MD[1:0]	EN3	EN2	EN1	EN0	Target counter
8-bit mode	00b	—	—	—	✓	ITL000
		—	—	✓	—	ITL001
		—	✓	—	—	ITL012
		✓	—	—	—	ITL013
16-bit mode	01b	Always set to 0.	—	Always set to 0.	✓	ITL000 + ITL001
		Always set to 0.	✓	Always set to 0.	—	ITL012 + ITL013
32-bit mode	10b	Always set to 0.	Always set to 0.	Always set to 0.	✓	ITL000 + ITL001 + ITL012 + ITL013

Note: ✓: Enables counting in the target counter.

Note: In 8-bit counter mode, two or more bits of EN3 to EN0 can be set to 1 or 0 at the same time.

Note: In 16-bit counter mode, the EN2 and EN0 bits can be set to 1 or 0 at the same time.

17.2.4 ITLCSEL0 : Interval Timer Clock Select Register 0

Base address: TML32 = 0x400A_3800							
Offset address: 0x0007							
Bit position:	7	6	5	4	3	2	1 0
Bit field:	—	CSEL[2:0]			—	ISEL[2:0]	
Value after reset:	0	0	0	0	0	0	0 0

Bit	Symbol	Function	R/W
2:0	ISEL[2:0]	Selection of Interval Timer Count Clock (f_{ITL0})*1 0 0 0: Counting stops 0 0 1: HOCO 1 0 0: FSXP Others: Setting prohibited	R/W
3	—	This bit is read as 0. The write value should be 0.	R/W
6:4	CSEL[2:0]	Selection of Interval Timer Count Clock for Capturing (f_{ITL1})*1 0 0 0: Counting stops 0 0 1: HOCO 1 0 0: FSXP Others: Setting prohibited	R/W
7	—	This bit is read as 0. The write value should be 0.	R/W

Note 1. Be sure to write to the CSEL[2:0] bits and ISEL[2:0] bits only while the ITLCTL0.EN3 to ITLCTL0.EN0 bits are all 0.

This register is used to select the count source for the interval timer.

EN3 位（8 位计数器模式：ITL013 计数使能）

在 8 位计数器模式下，向此位写入 1 会开始 ITL013 计数器的递增计数，写入 0 则会停止计数。

在 16 位计数器模式下，将此位设置为 0。

在 32 位计数器模式下，将此位设置为 0。

MD[1:0] 位（选择 8 位、16 位或 32 位计数器模式）

表 17.2 列出了可在MD[1:0] 位和 EN0 到 EN3 位设置中启用的目标计数器。

表 17.2 目标计数器设置

模式	MD[1:0]	EN3	EN2	EN1	EN0	目标计数器
8 位模式	00b	—	—	—	✓	ITL000
		—	—	✓	—	ITL001
		—	✓	—	—	ITL012
		✓	—	—	—	ITL013
16 位模式	01b	始终设置为 0。	—	始终设置为 0。	✓	ITL000 + ITL001
		始终设置为 0。	✓	始终设置为 0。	—	ITL012 + ITL013
32 位模式	10b	始终设置为 0。	始终设置为 0。	始终设置为 0。	✓	ITL000 + ITL001 + ITL012 + ITL013

注意: ✓: 启用目标计数器中的计数。

注意: 在 8 位计数器模式下，EN3 到 EN0 的两位或多位可以同时设置为 1 或 0。

注意: 在 16 位计数器模式下，EN2 和 EN0 位可以同时设置为 1 或 0。

17.2.4ITLCSEL0：间隔定时器时钟选择寄存器 0

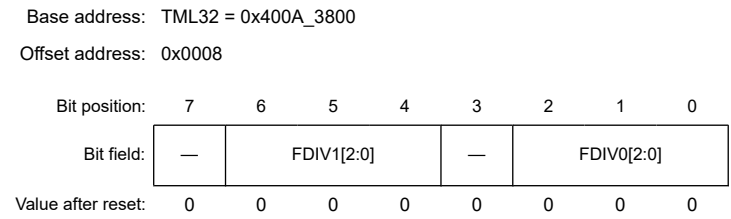
基地址: TML32 = 0x400A_3800							
偏移地址: 0x0007							
位位置:	7	6	5	4	3	2	1 0
位域:	—	CSEL[2:0]			—	ISEL[2:0]	
重置后的值:	0	0	0	0	0	0	0 0

少量	象征	功能	R/W
2:0	ISEL[2:0]	选择间隔定时器计数时钟 (f_{ITL0})*1 0 0 0: 计数停止 0 0 1: HOCO 1 0 0: FSXP 其他: 禁止设置	R/W
3	—	该位读取值为 0。写入值应为 0。	R/W
6:4	CSEL[2:0]	选择用于捕获(f_{ITL1})*1 的间隔定时器计数时钟 0 0 0: 计数停止 0 0 1: HOCO 1 0 0: FSXP 其他: 禁止设置	R/W
7	—	该位读取值为 0。写入值应为 0。	R/W

注意 1. 确保仅写入 CSEL[2:0] 位和 ISEL[2:0] 位，而 ITLCTL0.EN3 到 ITLCTL0.EN0 位均为 0。

该寄存器用于选择间隔定时器的计数源。

17.2.5 ITLFDIV00 : Interval Timer Frequency Division Register 0



Bit	Symbol	Function	R/W
2:0	FDIV0[2:0]	8-bit Counter Mode: Counter Clock for ITL000* ¹ 16-bit Counter Mode: Counter Clock for ITL000 + ITL001* ¹ 32-bit Counter Mode: Counter Clock for ITL000 + ITL001 + ITL012 + ITL013* ¹ 0 0 0: f _{ITL0} 0 0 1: f _{ITL0} /2 0 1 0: f _{ITL0} /4 0 1 1: f _{ITL0} /8 1 0 0: f _{ITL0} /16 1 0 1: f _{ITL0} /32 1 1 0: f _{ITL0} /64 1 1 1: f _{ITL0} /128	R/W
3	—	This bit is read as 0. The write value should be 0.	R/W
6:4	FDIV1[2:0]	8-bit Counter Mode: Counter Clock for ITL001* ² 0 0 0: f _{ITL0} 0 0 1: f _{ITL0} /2 0 1 0: f _{ITL0} /4 0 1 1: f _{ITL0} /8 1 0 0: f _{ITL0} /16 1 0 1: f _{ITL0} /32 1 1 0: f _{ITL0} /64 1 1 1: f _{ITL0} /128	R/W
7	—	This bit is read as 0. The write value should be 0.	R/W

Note 1. Be sure to write to the FDIV0[2:0] bits only while the ITLCTL0.EN0 bit is 0.
Note 2. In 8-bit counter mode, be sure to write to the FDIV1[2:0] bits only while the ITLCTL0.EN1 bit is 0.

This register is used to select the counter clock for the interval timer.

FDIV0[2:0] bits (8-bit Counter Mode: Counter Clock for ITL000, 16-bit Counter Mode: Counter Clock for ITL000 + ITL001, 32-bit Counter Mode: Counter Clock for ITL000 + ITL001 + ITL012 + ITL013)

In 8-bit counter mode, ITL000 counts cycles of the counter clock specified in the FDIV0[2:0] bits.

In 16-bit counter mode, ITL000 + ITL001 counts cycles of the counter clock specified in the FDIV0[2:0] bits.

In 32-bit counter mode, ITL000 + ITL001 + ITL012 + ITL013 counts cycles of the counter clock specified in the FDIV0[2:0] bits.

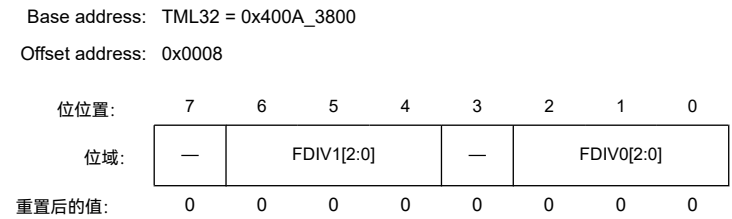
FDIV1[2:0] bits (8-bit Counter Mode: Counter Clock for ITL001)

In 8-bit counter mode, ITL001 counts cycles of the counter clock specified in the FDIV1[2:0] bits.

In 16-bit counter mode, set these bits to 000b.

In 32-bit counter mode, set these bits to 000b.

17.2.5 ITLFDIV00: 间隔定时器频率分频寄存器 0



少量	象征	功能	R/W
2:0	FDIV0[2:0]	8 位计数器模式: ITL000*1 的计数器时钟 16 位计数器模式: ITL000 + ITL001* ¹ 的计数器时钟 32 位计数器模式: ITL000+ITL001+ITL012+ITL013*1 的计数器时钟 0 0 0: f _{ITL0} 0 0 1: f _{ITL0} /2 0 1 0: f _{ITL0} /4 0 1 1: f _{ITL0} /8 1 0 0: f _{ITL0} /16 1 0 1: f _{ITL0} /32 1 1 0: f _{ITL0} /64 1 1 1: f _{ITL0} /128	R/W
3	—	该位读取值为 0。写入值应为 0。	R/W
6:4	FDIV1[2:0]	8 位计数器模式: ITL001*2 的计数器时钟 0 0 0: f _{ITL0} 0 0 1: f _{ITL0} /2 0 1 0: f _{ITL0} /4 0 1 1: f _{ITL0} /8 1 0 0: f _{ITL0} /16 1 0 1: f _{ITL0} /32 1 1 0: f _{ITL0} /64 1 1 1: f _{ITL0} /128	R/W
7	—	该位读取值为 0。写入值应为 0。	R/W

注意 1. 确保仅在 ITLCTL0.EN0 位为 0 时才写入 FDIV0[2:0] 位。
注 2. 在 8 位计数器模式下, 请确保仅在 ITLCTL0.EN1 位为 0 时才写入 FDIV1[2:0] 位。

该寄存器用于选择间隔定时器的计数器时钟。

FDIV0[2:0] 位 (8 位计数器模式: ITL000 的计数器时钟, 16 位计数器模式: 计数器时钟) ITL000 + ITL001, 32 位计数器模式: ITL000 + ITL001 + ITL012 + ITL013)的计数器时钟

In 8-位计数器模式, ITL000对 FDIV0[2:0] 位中指定的计数器时钟周期进行计数。

在 16 位计数器模式下, ITL000 + ITL001对 FDIV0[2:0] 位中指定的计数器时钟进行计数。

在 32 位计数器模式下, ITL000 + ITL001 + ITL012 + ITL013 对 FDIV0[2:0] 位中指定的计数器时钟进行计数。

FDIV1[2:0] 位 (8 位计数器模式: ITL001 的计数器时钟)

在 8 位计数器模式下, ITL001对 FDIV1[2:0] 位中指定的计数器时钟进行计数。

在 16 位计数器模式下, 将这些位设置为000b。

In 32- 位计数器模式, 将这些位设置为000b。

17.2.6 ITLFDIV01 : Interval Timer Frequency Division Register 1

Base address: TML32 = 0x400A_3800

Offset address: 0x0009

Bit position: 76543210

Bit field:

—	FDIV3[2:0]	—	FDIV2[2:0]

Value after reset: 00000000

Bit	Symbol	Function	R/W
2:0	FDIV2[2:0]	8-bit Counter Mode: Counter Clock for ITL012 ^{*1} 16-bit Counter Mode: Counter Clock for ITL012 + ITL013 ^{*1} 0 0 0: f _{ITL0} 0 0 1: f _{ITL0} /2 0 1 0: f _{ITL0} /4 0 1 1: f _{ITL0} /8 1 0 0: f _{ITL0} /16 1 0 1: f _{ITL0} /32 1 1 0: f _{ITL0} /64 1 1 1: f _{ITL0} /128	R/W
3	—	This bit is read as 0. The write value should be 0.	R/W
6:4	FDIV3[2:0]	8-bit Counter Mode: Counter Clock for ITL013 ^{*2} 0 0 0: f _{ITL0} 0 0 1: f _{ITL0} /2 0 1 0: f _{ITL0} /4 0 1 1: f _{ITL0} /8 1 0 0: f _{ITL0} /16 1 0 1: f _{ITL0} /32 1 1 0: f _{ITL0} /64 1 1 1: f _{ITL0} /128	R/W
7	—	This bit is read as 0. The write value should be 0.	R/W

Note 1. In 8-bit or 16-bit counter mode, be sure to write to the FDIV2[2:0] bits only while the ITLCTL0.EN2 bit is 0.
Note 2. In 8-bit counter mode, be sure to write to the FDIV3[2:0] bits only while the ITLCTL0.EN3 bit is 0.

This register is used to select the counter clock for the interval timer.

FDIV2[2:0] bits (8-bit Counter Mode: Counter Clock for ITL012, 16-bit Counter Mode: Counter Clock for ITL012 + ITL013)

In 8-bit counter mode, ITL012 counts cycles of the counter clock specified in the FDIV2[2:0] bits.
In 16-bit counter mode, ITL012 + ITL013 counts cycles of the counter clock specified in the FDIV2[2:0] bits.
In 32-bit counter mode, these bits are not used; write 000b to them.

FDIV3[2:0] bits (8-bit Counter Mode: Counter Clock for ITL013)

In 8-bit counter mode, ITL013 counts cycles of the counter clock specified in the FDIV3[2:0] bits.
In 16-bit counter mode, set these bits to 000b.
In 32-bit counter mode, set these bits to 000b.

17.2.6 ITLFDIV01: 间隔定时器频率分频寄存器 1

Base address: TML32 = 0x400A_3800

Offset address: 0x0009

位位置: 76543210

位域:

—	FDIV3[2:0]	—	FDIV2[2:0]

重置后的值: 00000000

少量	象征	功能	R/W
2:0	FDIV2[2:0]	8 位计数器模式: ITL012*1 的计数器时钟 16 位计数器模式: ITL012+ITL013*1 的计数器时钟 0 0 0: f _{ITL0} 0 0 1: f _{ITL0} /2 0 1 0: f _{ITL0} /4 0 1 1: f _{ITL0} /8 1 0 0: f _{ITL0} /16 1 0 1: f _{ITL0} /32 1 1 0: f _{ITL0} /64 1 1 1: f _{ITL0} /128	R/W
3	—	该位读取值为 0。写入值应为 0。	R/W
6:4	FDIV3[2:0]	8 位计数器模式: ITL013*2 的计数器时钟 0 0 0: f _{ITL0} 0 0 1: f _{ITL0} /2 0 1 0: f _{ITL0} /4 0 1 1: f _{ITL0} /8 1 0 0: f _{ITL0} /16 1 0 1: f _{ITL0} /32 1 1 0: f _{ITL0} /64 1 1 1: f _{ITL0} /128	R/W
7	—	该位读取值为 0。写入值应为 0。	R/W

注 1. 在 8 位或 16 位计数器模式下, 请确保仅在 ITLCTL0.EN2 位为 0 时才写入 FDIV2[2:0] 位。
注 2. 在 8 位计数器模式下, 请确保仅在 ITLCTL0.EN3 位为 0 时才写入 FDIV3[2:0] 位。

该寄存器用于选择间隔定时器的计数器时钟。

FDIV2[2:0] 位 (8 位计数器模式: ITL012 的计数器时钟, 16 位计数器模式: ITL012 + ITL013的计数器时钟)

在 8 位计数器模式下, ITL012 对 FDIV2[2:0] 位中指定的计数器时钟周期进行计数。
在 16 位计数器模式下, ITL012 + ITL013对 FDIV2[2:0] 位中指定的计数器时钟进行计数。
在 32 位计数器模式下, 这些位未使用; 请向其中写入 000b。

FDIV3[2:0] 位 (8 位计数器模式: ITL013 的计数器时钟)

在 8 位计数器模式下, ITL013 对 FDIV3[2:0] 位中指定的计数器时钟周期进行计数。
在 16 位计数器模式下, 将这些位设置为000b。
在 32 位计数器模式下, 将这些位设置为000b。

17.2.7 ITLCC0 : Interval Timer Capture Control Register 0

Base address:	TML32 = 0x400A_3800							
Offset address:	0x000A							
Bit position:	7	6	5	4	3	2	1	0
Bit field:	CAPE N	CAPF CR	CAPF	CAPR	CAPC CR	—	CTRS[1:0]	
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
1:0	CTRS[1:0]	Selection of Capture Trigger*1 *2 0 0: Software trigger 0 1: Interrupt on compare match with ITLCMP01 1 0: FSXP (rising edge) 1 1: Setting prohibited	R/W
2	—	This bit is read as 0. The write value should be 0.	R/W
3	CAPCCR	Selection of Capture Counter Clearing After Capturing*3 0: The capture counter value is held after the completion of capturing. 1: The capture counter value is cleared after the completion of capturing.	R/W
4	CAPR	Software Capture Trigger*2 *4 0: Trigger operation does not proceed. 1: A software trigger for capturing is generated.	R/W
5	CAPF	Capture Completion Flag This flag is set to 1 after a capture trigger selected in the CTRS[1:0] bits is generated and the captured data is stored in ITLCAP00. Writing 1 to the CAPFCR bit clears this flag to 0. 0: Capturing has not been completed. 1: Capturing has been completed.	R
6	CAPFCR	Capture Completion Flag Clear*5 0: The value of the capture completion flag CAPF is held. 1: The value of the capture completion flag CAPF is cleared.	R/W
7	CAPEN	Capture Enable*6 0: Capturing is disabled. 1: Capturing is enabled.	R/W

Note 1. Be sure to write to the CTRS[1:0] bits only while the ITLCTL0.EN3 to ITLCTL0.EN0 bits are all 0.
Note 2. In the capture operation, the interval at which the capture trigger is generated should be two or more cycles of the counter clock.
Note 3. Be sure to write to the CAPCCR bit only while the ITLCTL0.EN3 to ITLCTL0.EN0 bits are all 0.
Note 4. The CAPR bit is always read as 0.
Note 5. The CAPFCR bit is always read as 0.
Note 6. Be sure to write to the CAPEN bit only while the ITLCTL0.EN3 to ITLCTL0.EN0 bits are all 0.

This register is used to enable or disable the capture function of the interval timer, specify whether to hold or clear the capture completion flag, set up the software trigger, and select the capture trigger.

17.2.8 ITLS0 : Interval Timer Status Register

Base address:	TML32 = 0x400A_3800							
Offset address:	0x000B							
Bit position:	7	6	5	4	3	2	1	0
Bit field:	—	—	—	ITF0C	ITF03	ITF02	ITF01	ITF00
Value after reset:	0	0	0	0	0	0	0	0

17.2.7 ITLCC0: 间隔定时器捕获控制寄存器 0

基地址:	TML32= 0x400A_3800							
Offset address:	0x000A							
位位置:	7	6	5	4	3	2	1	0
位域:	CAPE N	CAPF CR	CAPF	CAPR	CAPC CR	—	CTRS[1:0]	
重置后的值:	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
1:0	CTRS[1:0]	选择捕获触发器*1 *2 0 0: 软件触发器 0 1: 与 ITLCMP01 比较匹配时中断 1 0: FSXP（上升边缘） 1.1: 禁止设置	R/W
2	—	该位读取值为 0。写入值应为 0。	R/W
3	CAPCCR	选择捕获计数器清除后捕获*3 0: 捕获完成后，捕获计数器值将被保留。 1: 捕获完成后，捕获计数器值将被清除。	R/W
4	CAPR	软件捕获触发器*2 *4 0: 触发操作不进行。 1: 生成用于捕获的软件触发器。	R/W
5	CAPF	捕获完成标志 在 CTRS[1:0] 位中选择捕获触发信号后，该标志位被设置为 1，捕获的数据存储在 ITLCAP00 中。 将 1 写入 CAPFCR 位会将该标志清除为 0。 0: 采集尚未完成。 1: 采集已完成。	R
6	CAPFCR	捕获完成标志清除*5 0: 捕获完成标志 CAPF 的值被保留。 1: 捕获完成标志 CAPF 的值被清除。	R/W
7	CAPEN	捕获启用*6 0: 已禁用捕获。1: 已启用捕获。	R/W

注意 1. 确保仅写入 CTRS[1:0] 位，而 ITLCTL0.EN3 到 ITLCTL0.EN0 位均为 0。
注 2. 在捕获操作中，生成捕获触发的间隔应为计数器时钟的两个或更多周期。
注 3. 务必仅在 ITLCTL0.EN3 到 ITLCTL0.EN0 位均为 0 时才写入 CAPCCR 位。
注 4. CAPR 位始终读取为 0。
注 5. CAPFCR 位始终读取为 0。
注 6. 务必仅在 ITLCTL0.EN3 到 ITLCTL0.EN0 位均为 0 时才写入 CAPEN 位。

该寄存器用于启用或禁用间隔定时器的捕获功能，指定是否保持或清除捕获完成标志，设置软件触发器，以及选择捕获触发器。

17.2.8ITLS0: 间隔定时器状态寄存器

基地址:	TML32 = 0x400A_3800							
偏移地址:	0x000B							
位位置:	7	6	5	4	3	2	1	0
位域:	—	—	—	ITF0C	ITF03	ITF02	ITF01	ITF00
重置后的值:	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
0	ITF00	Compare Match Detection Flag for Channel 0 0: A compare match signal has not been detected in channel 0. 1: A compare match signal has been detected in channel 0.	R/W
1	ITF01	Compare Match Detection Flag for Channel 1 0: A compare match signal has not been detected in channel 1. 1: A compare match signal has been detected in channel 1.	R/W
2	ITF02	Compare Match Detection Flag for Channel 2 0: A compare match signal has not been detected in channel 2. 1: A compare match signal has been detected in channel 2.	R/W
3	ITF03	Compare Match Detection Flag for Channel 3 0: A compare match signal has not been detected in channel 3. 1: A compare match signal has been detected in channel 3.	R/W
4	ITF0C	Capture Detection Flag 0: Completion of capturing has not been detected. 1: Completion of capturing has been detected.	R/W
7:5	—	These bits are read as 0. The write value should be 0.	R/W

Note: Writing 1 to each bit is ignored. To clear the ITF0C or ITF0i bit (i = 0, 1, 2, 3), write 0 to the desired bit and 1 to the other bits.

Note: If clearing any of the ITF0C, ITF03, ITF02, ITF01, ITF00 flag bits to 0 does not lead to the value of the ITLS0 register becoming 0x00, an interrupt request (TML32_ITL_OR) is generated.

Note: To clear a flag bit in the ITLS0 register to 0, only write 0 to a bit that has the setting 1. This is because writing 0 to a bit that has the setting 0 may make detecting a compare match signal or capture detection signal generated at the same time as the writing of 0 impossible. For example, when the ITF01 flag bit is set to 1, write 00011101b to the ITLS0 register to clear the ITF01 flag bit.

This is a status register for the interval timer.

When the value of the ITL0mn counter (mn = 00, 01, 12, 13) matches the value specified in the ITLCMP00 and ITLCMP01 registers, the compare-match flag for the corresponding channel is set.

When a capture trigger is generated while the CAPEN bit in the ITLCC0 register is 1, the capture detection flag is set after the value of the ITL0mn counter is stored in the ITLCAP00 register.

The values of the ITF0C and ITF03 to ITF00 bits in this register are ORed and output as the TML32_ITL_OR interrupt signal. Table 17.3 shows the conditions for setting the status flags in each timer mode selected by the ITLCTL0.MD[1:0] bits.

Table 17.3 Conditions for setting the status flags in each timer mode

Mode	ITLCTL0.MD[1:0]	ITLCC0.CAPEN	Status flag	Conditions for setting status flag
8-bit mode	00b	x	ITF00	The next rising edge of the counter clock following a match between the ITLCMP00_L and ITL000 values
		x	ITF01	The next rising edge of the counter clock following a match between the ITLCMP00_H and ITL001 values
		x	ITF02	The next rising edge of the counter clock following a match between the ITLCMP01_L and ITL012 values
		x	ITF03	The next rising edge of the counter clock following a match between the ITLCMP01_H and ITL013 values
16-bit mode	01b	x	ITF00	The next rising edge of the counter clock following a match between the ITLCMP00 and ITL000 + ITL001 values
		x	ITF02	The next rising edge of the counter clock following a match between the ITLCMP01 and ITL012 + ITL013 values
		1	ITF0C	The ITL000 + ITL001 value is stored in ITLCAP00 after a capture trigger is generated.
32-bit mode	10b	—	ITF00	The next rising edge of the counter clock following a match between the ITLCMP00 + ITLCMP01 and ITL000 + ITL001 + ITL012 + ITL013 values

少量	象征	功能	R/W
0	ITF00	通道 0 的比较匹配检测标志：通道 0 中未检测到比较匹配信号。 1：在通道 0 中检测到比较匹配信号。	R/W
1	ITF01	通道 1 的比较匹配检测标志 0：通道 1 中未检测到比较匹配信号。 1：在通道 1 中检测到比较匹配信号。	R/W
2	ITF02	通道 2 的比较匹配检测标志 0：通道 2 中未检测到比较匹配信号。 1：在通道 2 中检测到比较匹配信号。	R/W
3	ITF03	通道 3 的比较匹配检测标志 0：通道 3 中未检测到比较匹配信号。 1：在通道 3 中检测到比较匹配信号。	R/W
4	ITF0C	捕获检测标志 0：未检测到捕获完成。 1：已检测到捕获完成。	R/W
7:5	—	这些位读取为 0。写入值应为 0。	R/W

注意：向每个位写入 1 的操作将被忽略。要清除 ITF0C 或 ITF0i 位 (i = 0, 1、2、3)， 请向所需位写入 0，向其他位写入 1。

注意：如果将 ITF0C、ITF03、ITF02、ITF01、ITF00 标志位中的任何一个清除为 0，却没有使 ITLS0 寄存器的值变为 0x00，，则会生成中断请求 (TML32_ITL_OR)。

注意：要将 ITLS0 寄存器中的某个标志位清零，只能将 0 写入当前值为 1 的位。这是因为将 0 写入当前值为 0 的位可能会导致无法检测到与写入 0 同时产生的比较匹配信号或捕获检测信号。例如，当 ITF01 标志位设置为 1 时，应将 00011101b 写入 ITLS0 寄存器以清除 ITF01 标志位。

这是间隔定时器的状态寄存器。

当 ITL0mn 计数器(mn = 00 (01、12、13) 的值与 ITLCMP00 和 ITLCMP01 寄存器中指定的值匹配时，相应通道的比较匹配标志将被设置。

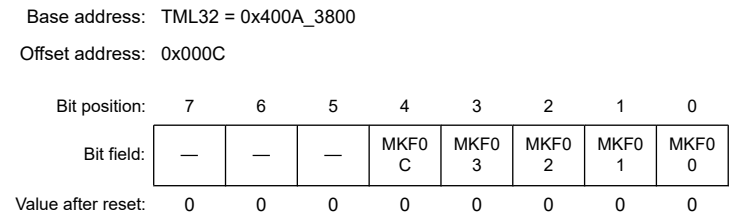
当 ITLCC0 寄存器中的 CAPEN 位为 1 时生成捕获触发信号，在将 ITL0mn 计数器的值存储到 ITLCAP00 寄存器之后，捕获检测标志将被设置。

该寄存器中 ITF0C 和 ITF03 至 ITF00 位的值进行按位或运算，并作为 TML32_ITL_OR 中断信号输出。表 17.3 显示了 ITLCTL0.MD[1:0] 位选择的每个定时器模式下设置状态标志的条件。

表 17.3 各定时器模式下设置状态标志的条件

模式	ITLCTL0.MD[1:0]	ITLCC0.CAPEN	状态标志	设置状态标志的条件
8 位模式	00b	x	ITF00	ITLCMP00_L 和 ITL000 值匹配后，计数器时钟的下一个上升沿。
		x	ITF01	ITLCMP00_H 和 ITL001 值匹配后，计数器时钟的下一个上升沿。
		x	ITF02	ITLCMP01_L 和 ITL012 值匹配后，计数器时钟的下一个上升沿。
		x	ITF03	ITLCMP01_H 和 ITL013 值匹配后，计数器时钟的下一个上升沿出现。
16 位模式	01b	x	ITF00	ITLCMP00 和 ITL000 + ITL001 值匹配后，计数器时钟的下一个上升沿
		x	ITF02	ITLCMP01 和 ITL012 + ITL013 值匹配后，计数器时钟的下一个上升沿。
		1	ITF0C	捕获触发器生成后，ITL000+ ITL001 值存储在 ITLCAP00 中。
32 位模式	10b	—	ITF00	ITLCMP00+ ITLCMP01 和 ITL000 + ITL001 + ITL012 + ITL013 值匹配后，计数器时钟的下一个上升沿。

17.2.9 ITLMKF0 : Interval Timer Match Detection Mask Register



Bit	Symbol	Function	R/W
0	MKF00	Mask for Compare Match Status Flag for Channel 0 ^{*1} 0: ITLS0.ITF00 is not masked 1: ITLS0.ITF00 is masked	R/W
1	MKF01	Mask for Compare Match Status Flag for Channel 1 ^{*1} 0: ITLS0.ITF01 is not masked 1: ITLS0.ITF01 is masked	R/W
2	MKF02	Mask for Compare Match Status Flag for Channel 2 ^{*1} 0: ITLS0.ITF02 is not masked 1: ITLS0.ITF02 is masked	R/W
3	MKF03	Mask for Compare Match Status Flag for Channel 3 ^{*1} 0: ITLS0.ITF03 is not masked 1: ITLS0.ITF03 is masked	R/W
4	MKF0C	Mask for Capture Detection Status Flag ^{*1} 0: ITLS0.ITF0C is not masked 1: ITLS0.ITF0C is masked	R/W
7:5	—	These bits are read as 0. The write value should be 0.	R/W

Note 1. Setting all functional bits to 1 for masking prevents setting of the corresponding bits in the ITLS0 register. This in turn prevents software detection of compare matches and completion of capture. When compare match for any of channels 0 to 3 is to be used, be sure to set the bit corresponding to the given status flag to 0 so that the flag is not masked. For the state of completion of capture, on the other hand, the CAPF flag in the interval timer capture control register 0 (ITLCC0) can be used to detect this even when the MKF0C bit is set to 1 to mask the ITLS0.ITF0C flag.

This register is used to enable or disable setting of each valid bit in the interval timer status register (ITLS0) to 1.

Setting an MKF0C or MKF0i (i = 0 to 3) bit to 1 masks the corresponding status flag among ITF0C and ITF0i (i = 0 to 3), after which the given flag is not set to 1 even if a compare match with a compare register or capture completion is detected. Since the status flag is not set to 1, masking also prevents generation of the interval detection interrupt (TML32_ITL_OR).

17.3 Operation

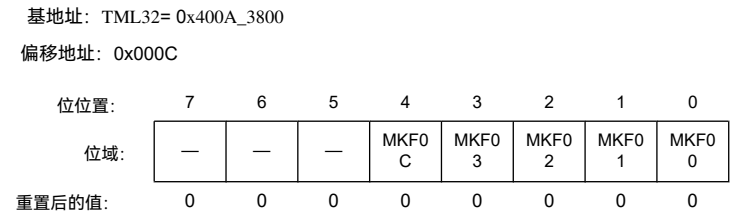
17.3.1 Counter Mode Settings

The 32-bit interval timer has three counter modes: 8-bit counter mode, 16-bit counter mode, and 32-bit counter mode. [Table 17.4](#) to [Table 17.6](#) show the registers and settings for use in 8-bit counter mode, 16-bit counter mode, and 32-bit counter mode, respectively.

Table 17.4 Registers and settings used in 8-bit counter mode (1 of 2)

Register name (symbol)	Bit	Setting
Interval timer compare registers 000 (ITLCMP00)	Bits 7 to 0	Specify 8-bit compare values for channels 0.
Interval timer compare registers 001 (ITLCMP00)	Bits 7 to 0	Specify 8-bit compare values for channels 1.
Interval timer compare registers 012 (ITLCMP01)	Bits 7 to 0	Specify 8-bit compare values for channels 2.
Interval timer compare registers 013 (ITLCMP01)	Bits 7 to 0	Specify 8-bit compare values for channels 3.

17.2.9 ITLMKF0: 间隔定时器匹配检测掩码寄存器



少量	象征	功能	R/W
0	MKF00	用于比较匹配状态标志的通道掩码0 ^{*1} 0: ITLS0.ITF00 未被屏蔽 1: ITLS0.ITF00 已被屏蔽	R/W
1	MKF01	通道 1*1 的比较匹配状态标志掩码 0: ITLS0.ITF01 未被屏蔽 1: ITLS0.ITF01 已被屏蔽	R/W
2	MKF02	用于比较匹配状态标志的通道掩码2 ^{*1} 0: ITLS0.ITF02 未被屏蔽 1: ITLS0.ITF02 已被屏蔽	R/W
3	MKF03	用于比较匹配状态标志的通道掩码3 ^{*1} 0: ITLS0.ITF03 未被屏蔽 1: ITLS0.ITF03 已被屏蔽	R/W
4	MKF0C	捕获检测状态标志掩码*1 0: ITLS0.ITF0C 未被屏蔽 1: ITLS0.ITF0C 已被屏蔽	R/W
7:5	—	这些位读取为 0。写入值应为 0。	R/W

注1: 将所有功能位设置为1进行掩码会阻止ITLS0寄存器中相应位的设置。这反过来会阻止软件检测比较匹配和捕获完成。当需要使用通道0到3中任意通道的比较匹配时, 请务必将与给定状态标志对应的位设置为0, 以避免该标志被掩码。另一方面, 对于捕获完成状态, 即使MKF0C位设置为1以掩码ITLS0.ITF0C标志, 也可以使用间隔定时器捕获控制寄存器0 (ITLCC0) 中的CAPF标志来检测捕获完成状态。

该寄存器用于启用或禁用将间隔定时器状态寄存器 (ITLS0) 中的每个有效位设置为 1。

将MKF0C或MKF0i(i=0至3)位设置为1会屏蔽ITF0C和ITF0i(i=0至3)中对应的状态标志。此后, 即使检测到与比较寄存器的比较匹配或捕获完成, 该标志也不会被设置为1。由于状态标志未被设置为1, 屏蔽操作还会阻止间隔检测中断 (TML32_ITL_OR) 的生成。

17.3 操作

17.3.1 计数器模式设置

32 位间隔定时器有三种计数模式: 8 位计数模式、16 位计数模式和 32 位计数模式。表 17.4 至表 17.6 分别列出了 8 位计数模式、16 位计数模式和 32 位计数模式下使用的寄存器和设置。

表 17.4 8 位计数器模式下使用的寄存器和设置 (1/2)

注册名称 (符号)	少量	环境
间隔定时器比较寄存器 000 (ITLCMP00)	第 7 位到第 0 位	为通道 0 指定 8 位比较值。
间隔定时器比较寄存器 001 (ITLCMP00)	第 7 位到第 0 位	为通道 1 指定 8 位比较值。
间隔定时器比较寄存器 012 (ITLCMP01)	第 7 位到第 0 位	为通道 2 指定 8 位比较值。
间隔定时器比较寄存器 013 (ITLCMP01)	第 7 位到第 0 位	为通道 3 指定 8 位比较值。

Table 17.4 Registers and settings used in 8-bit counter mode (2 of 2)

Register name (symbol)	Bit	Setting
Interval timer control register 0 (ITLCTL0)	EN0	Specify whether to start or stop counting in channel 0.
	EN1	Specify whether to start or stop counting in channel 1.
	EN2	Specify whether to start or stop counting in channel 2.
	EN3	Specify whether to start or stop counting in channel 3.
	MD[1:0]	Set to 00b.
Interval timer frequency division registers 0 (ITLFDIV00)	FDIV0[2:0]	Select the count clock for channel 0.
	FDIV1[2:0]	Select the count clock for channel 1.
Interval timer frequency division registers 1 (ITLFDIV01)	FDIV2[2:0]	Select the count clock for channel 2.
	FDIV3[2:0]	Select the count clock for channel 3.
Interval timer clock select register 0 (ITLCSEL0)	ISEL[2:0]	Select the count clock for the interval timer.
	CSEL[2:0]	Set to 000b.
Interval timer capture control register 0 (ITLCC0)	Bits 7 to 0	Set to 0.

Table 17.5 Registers and settings used in 16-bit counter mode

Register name (symbol)	Bit	Setting
Interval timer compare registers 00 (ITLCMP00)	Bits 15 to 0	Specify 16-bit compare values for channels 0 and 1
Interval timer compare registers 01 (ITLCMP01)	Bits 15 to 0	Specify 16-bit compare values for channels 2 and 3.
Interval timer control register 0 (ITLCTL0)	EN0	Specify whether to start or stop counting in channels 0 and 1.
	EN1	Set to 0.
	EN2	Specify whether to start or stop counting in channels 2 and 3.
	EN3	Set to 0.
	MD[1:0]	Set to 01b.
Interval timer frequency division registers 0 (ITLFDIV00)	FDIV0[2:0]	Select the count clock for channels 0 and 1.
	FDIV1[2:0]	Set to 000b.
Interval timer frequency division registers 1 (ITLFDIV01)	FDIV2[2:0]	Select the count clock for channels 2 and 3.
	FDIV3[2:0]	Set to 000b.
Interval timer clock select register 0 (ITLCSEL0)	ISEL[2:0]	Select the count clock for the interval timer.
	CSEL[2:0]	Set to 000b.
Interval timer capture control register 0 (ITLCC0)	Bits 7 to 0	Set to 0.

Table 17.6 Registers and settings used in 32-bit counter mode (1 of 2)

Register name (symbol)	Bit	Setting
Interval timer compare registers 00 (ITLCMP00)	Bits 15 to 0	Specify a compare value in 32-bit counter mode. Specify the lower 16 bits of the compare value in channels 0 and 1 (ITLCMP00).
Interval timer compare registers 01 (ITLCMP01)	Bits 15 to 0	Specify a compare value in 32-bit counter mode. Specify the upper 16 bits of the compare value in channels 2 and 3 (ITLCMP01).

表 17.4 8 位计数器模式中使用的寄存器和设置 (2/2)

注册名称 (符号)	少量	环境
间隔定时器控制寄存器 0 (ITLCTL0)	EN0	指定是否开始或停止通道 0 的计数。
	EN1	指定是否开始或停止通道 1 的计数。
	EN2	指定是否开始或停止通道 2 的计数。
	EN3	指定是否开始或停止通道 3 的计数。
	MD[1:0]	设置为 00b。
间隔定时器分频寄存器 0 (ITLFDIV00)	FDIV0[2:0]	选择通道 0 的计数时钟。
	FDIV1[2:0]	选择通道 1 的计数时钟。
间隔定时器分频寄存器 1 (ITLFDIV01)	FDIV2[2:0]	选择通道 2 的计数时钟。
	FDIV3[2:0]	选择通道 3 的计数时钟。
间隔定时器时钟选择寄存器 0 (ITLCSEL0)	ISEL[2:0]	选择间隔定时器的计数时钟。
	CSEL[2:0]	设置为 000b。
间隔定时器捕获控制寄存器 0 (ITLCC0)	第 7 位到第 0 位	设置为 0。

表 17.5 16 位计数器模式下使用的寄存器和设置

注册名称 (符号)	少量	环境
间隔定时器比较寄存器 00 (ITLCMP00)	第 15 位到第 0 位	为通道 0 和 1 指定 16 位比较值
间隔定时器比较寄存器 01 (ITLCMP01)	第 15 位到第 0 位	为通道 2 和 3 指定 16 位比较值。
间隔定时器控制寄存器 0 (ITLCTL0)	EN0	指定是否开始或停止通道 0 和 1 的计数。
	EN1	设置为 0。
	EN2	指定是否开始或停止通道 2 和 3 的计数。
	EN3	设置为 0。
	MD[1:0]	设置为 01b。
间隔定时器分频寄存器 0 (ITLFDIV00)	FDIV0[2:0]	选择通道 0 和 1 的计数时钟。
	FDIV1[2:0]	设置为 000b。
间隔定时器分频寄存器 1 (ITLFDIV01)	FDIV2[2:0]	选择通道 2 和 3 的计数时钟。
	FDIV3[2:0]	设置为 000b。
间隔定时器时钟选择寄存器 0 (ITLCSEL0)	ISEL[2:0]	选择间隔定时器的计数时钟。
	CSEL[2:0]	设置为 000b。
间隔定时器捕获控制寄存器 0 (ITLCC0)	第 7 位到第 0 位	设置为 0。

表 17.6 32 位计数器模式下使用的寄存器和设置 (1/2)

注册名称 (符号)	少量	环境
间隔定时器比较寄存器 00 (ITLCMP00)	第 15 位到第 0 位	在 32 位计数器模式下指定比较值。在通道 0 和 1 中指定比较值的低 16 位 (ITLCMP00)。
间隔定时器比较寄存器 01 (ITLCMP01)	第 15 位到第 0 位	在 32 位计数器模式下指定比较值。在通道 2 和 3 中指定比较值的高 16 位 (ITLCMP01)。

Table 17.6 Registers and settings used in 32-bit counter mode (2 of 2)

Register name (symbol)	Bit	Setting
Interval timer control register 0 (ITLCTL0)	EN0	Specify whether to start or stop counting in channels 0 to 3.
	EN1	Set to 0.
	EN2	Set to 0.
	EN3	Set to 0.
	MD[1:0]	Set to 10b.
Interval timer frequency division registers 0 (ITLFDIV00)	FDIV0[2:0]	Select the count clock for channels 0 to 3.
	FDIV1[2:0]	Set to 000b.
Interval timer frequency division registers 1 (ITLFDIV01)	FDIV2[2:0]	Set to 000b.
	FDIV3[2:0]	Set to 000b.
Interval timer clock select register 0 (ITLCSEL0)	ISEL[2:0]	Select the count clock for the interval timer.
	CSEL[2:0]	Set to 000b.
Interval timer capture control register 0 (ITLCC0)	Bits 7 to 0	Set to 0.

17.3.2 Capture Mode Settings

When the 16-bit capture mode is to be used for channels 0 and 1, the counter value is stored in interval timer capture register 00 (ITLCAP00) in response to a selected capture trigger.

Table 17.7 show the registers and settings for use in 16-bit capture mode.

Table 17.7 Registers and settings used in 16-bit capture mode

Register name (symbol)	Bit	Setting
Interval timer compare register 00 (ITLCMP00)	Bits 15 to 0	Specify 16-bit compare values for channels 0 and 1.
Interval timer compare register 01 (ITLCMP01)*1	Bits 15 to 0	Specify 16-bit compare values for channels 2 and 3.
Interval timer control register 0 (ITLCTL0)	EN0	Specify whether to start or stop counting in channels 0 and 1.
	EN1	Set to 0.
	EN2	Specify whether to start or stop counting in channels 2 and 3.
	EN3	Set to 0.
	MD[1:0]	Set to 01b.
Interval timer frequency division registers 0 (ITLFDIV00)	FDIV0[2:0]	Select the count clock for channel 0.
	FDIV1[2:0]	Set to 000b.
Interval timer frequency division registers 1 (ITLFDIV01)	FDIV2[2:0]	Set to 000b.
	FDIV3[2:0]	Set to 000b.
Interval timer clock select register 0 (ITLCSEL0)	ISEL[2:0]	Select the count clock for the interval timer in channels 0 and 1.
	CSEL[2:0]	Select the count clock for the interval timer for capturing in channels 2 and 3.
Interval timer capture control register 0 (ITLCC0)	CAPEN	Set to 1.
	CAPCCR	Specify whether to clear or hold the counter value in channels 0 and 1 after the completion of capturing.
	CTRS[1:0]	Select a capture trigger.

Note 1. Channels 2 and 3 can only be used in 16-bit counter mode when an interrupt on compare match with ITLCMP01 is not to be used as a capture trigger.

表 17.6 32 位计数器模式中使用的寄存器和设置 (2/2)

注册名称（符号）	少量	环境
间隔定时器控制寄存器 0 (ITLCTL0)	EN0	指定是否开始或停止通道 0 到的计数 3.
	EN1	设置为0。
	EN2	设置为0。
	EN3	设置为0。
	MD[1:0]	设置为 10b。
间隔定时器分频寄存器 0 (ITLFDIV00)	FDIV0[2:0]	选择通道 0 至 3 的计数时钟。
	FDIV1[2:0]	设置为 000b。
间隔定时器分频寄存器 1 (ITLFDIV01)	FDIV2[2:0]	设置为 000b。
	FDIV3[2:0]	设置为 000b。
间隔定时器时钟选择寄存器 0 (ITLCSEL0)	ISEL[2:0]	选择间隔定时器的计数时钟。
	CSEL[2:0]	设置为 000b。
间隔定时器捕获控制寄存器 0 (ITLCC0)	第 7 位到第 0 位	设置为0。

17.3.2 捕获模式设置

当对通道 0 和 1 使用 16 位捕获模式时，计数器值会根据选定的捕获触发存储在间隔定时器捕获寄存器 00 (ITLCAP00) 中。

表 17.7 显示了 16 位捕获模式下使用的寄存器和设置。

表 17.7 16 位捕获模式中使用的寄存器和设置

注册名称（符号）	少量	环境
间隔定时器比较寄存器 00 (ITLCMP00)	第 15 位到第 0 位	为通道 0 和 1 指定 16 位比较值。
间隔定时器比较寄存器 01 (ITLCMP01)*1	第 15 位到第 0 位	为通道 2 和 3 指定 16 位比较值。
间隔定时器控制寄存器 0 (ITLCTL0)	EN0	指定是否开始或停止通道 0 和 1 的计数。
	EN1	设置为0。
	EN2	指定是否开始或停止通道 2 和 3 的计数。
	EN3	设置为0。
	MD[1:0]	设置为 01b。
间隔定时器分频寄存器 0 (ITLFDIV00)	FDIV0[2:0]	选择通道 0 的计数时钟。
	FDIV1[2:0]	设置为 000b。
间隔定时器分频寄存器 1 (ITLFDIV01)	FDIV2[2:0]	设置为 000b。
	FDIV3[2:0]	设置为 000b。
间隔定时器时钟选择寄存器 0 (ITLCSEL0)	ISEL[2:0]	选择通道 0 和 1 中间隔定时器的计数时钟。
	CSEL[2:0]	选择通道 2 和 3 的间隔定时器的计数时钟，用于捕获。
间隔定时器捕获控制寄存器 0 (ITLCC0)	CAPEN	设置为 1。
	CAPCCR	指定捕获完成后是清除还是保留通道 0 和 1 中的计数器值。
	CTRS[1:0]	选择一个捕获触发器。

注 1. 通道 2 和 3 只能在 16 位计数器模式下使用，当与 ITLCMP01 的比较匹配中断不用作捕获触发时。

17.3.3 Timer Operation

The ITL0mn counter counts up cycles of the counting clock specified in the interval timer frequency division registers (ITLFDIV00 and ITLFDIV01). An interrupt request signal (TML32_ITL_OR) is generated on the counting of the next clock cycle after the value of the counter matches the comparison value. The interrupt request signal (TML32_ITL_OR) remains high until the value of the ITLS0 register becomes 0x00.

While the interrupt request signal (TML32_ITL_OR) is high, the generation of an additional interrupt request (TML32_ITL_OR) does not proceed even if a compare match or capture completion is detected for an operating channel.

Clearing the ITLCTL0.EN0 to EN3 bits to 0 clears the counter value.

Figure 17.6 shows an example of timer operation.

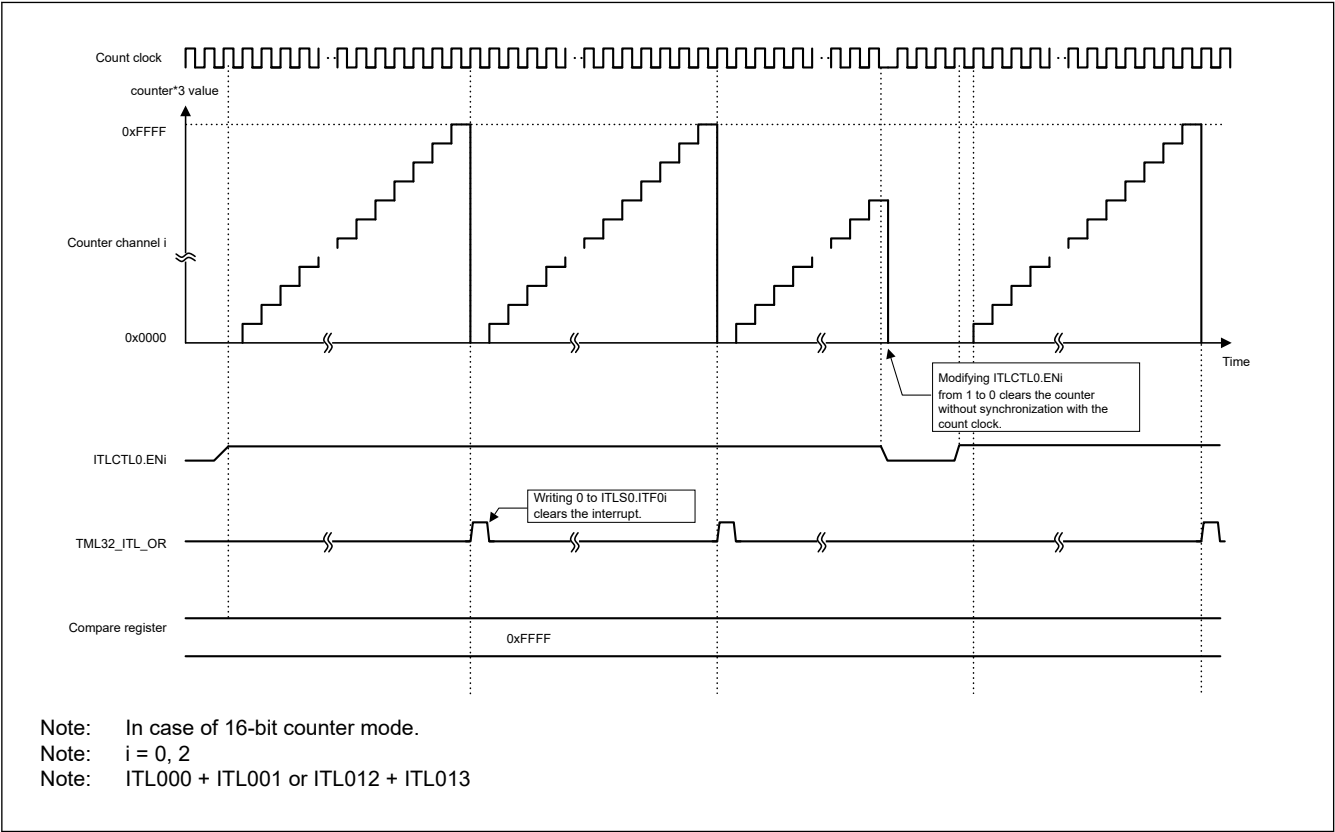


Figure 17.6 Example of timer operation

17.3.4 Capture Operation

When the setting of the CAPEN bit in the interval timer capture control register 0 (ITLCC0) is 1, the values in the 16-bit counters (ITL000 and ITL001) are stored in interval timer capture register 00 (ITLCAP00) in response to the capture trigger specified in the ITLCC0 register.

The capture trigger is selectable from among the interrupt on compare match with the ITLCMP01 register, FSXP, and a software trigger (setting the ITLCC0.CAPR bit to 1). To use the interrupt on compare match with the ITLCMP01 register as the capture trigger, set interval timer clock select register 0 (ITLCSEL0) to select the clock for counting, and set interval timer compare register 01 (ITLCMP01) to specify the comparison value. When using FSXP, or a software trigger (setting the ITLCC0.CAPR bit to 1) as a capture trigger, channels 2 and 3 can be used in 16-bit counter mode.

After a capture trigger is input and the counter value is stored in the interval timer capture register, the interrupt request signal (TML32_ITL_OR) is output, the capture completion flag (ITLCC0.CAPF) and capture detection flag (ITLS0.ITF0C) are set to 1, and the flag values are retained until they are explicitly cleared *1. The ITLCC0.CAPF flag can be cleared by setting the ITLCC0.CAPFCR bit to 1. The ITF0C flag in the ITLS0 register can be cleared by writing 0 to it. Since capture operations operate with the counter clock, the interval at which the capture trigger is generated should be at least 5 cycles of the counter clock. If a capture trigger is generated again within 2 cycles of the counter clock after an earlier capture trigger was generated, the ITLCC0.CAPF bit may not be set.

17.3.3 定时器操作

ITL0mn计数器根据间隔定时器分频寄存器 (ITLFDIV00 和 ITLFDIV01) 中指定的计数时钟周期进行计数。当计数器值与比较值匹配后，在下一个时钟周期计数时，会生成中断请求信号(TML32_ITL_OR)。中断请求信号 (TML32_ITL_OR) 保持高电平，直到 ITLS0 寄存器的值变为0x00为止。

当中断请求信号 (TML32_ITL_OR) 为高电平时，即使检测到操作通道的比较匹配或捕获完成，也不会生成额外的中断请求 (TML32_ITL_OR)。

将 ITLCTL0.EN0 到 EN3 位清除为 0 可清除计数器值。

图 17.6 显示了定时器操作的示例。

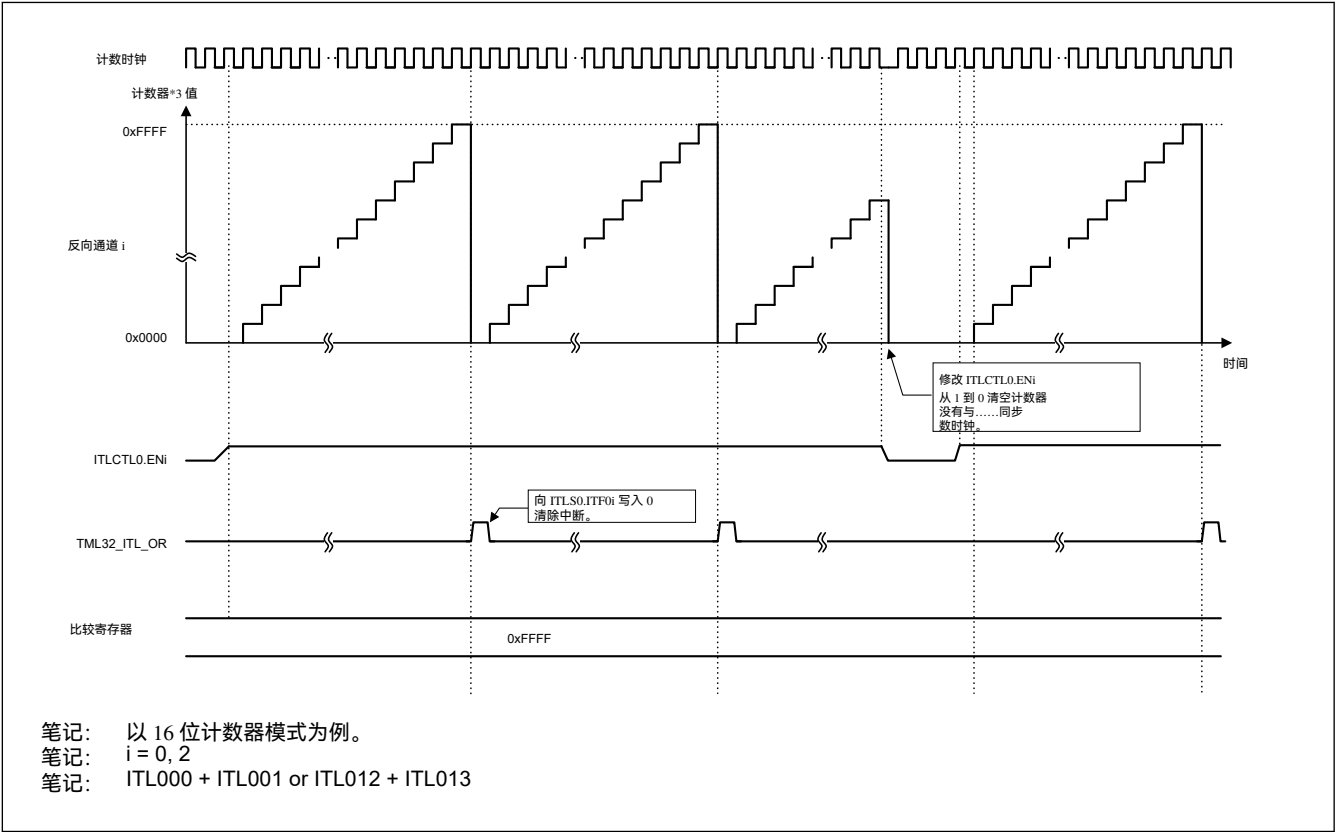


图 17.6 定时器操作示例

17.3.4 捕获操作

当间隔定时器捕获控制寄存器 0 (ITLCC0) 中的 CAPEN 位设置为 1 时，16 位计数器 (ITL000 和 ITL001) 中的值将根据 ITLCC0 寄存器中指定的捕获触发存储在间隔定时器捕获寄存器 00 (ITLCAP00) 中。

捕获触发方式可从以下选项中选择：与 ITLCMP01 寄存器比较匹配中断、FSXP 以及软件触发（将 ITLCC0.CAPR 位设置为 1）。要使用与 ITLCMP01 寄存器比较匹配中断作为捕获触发，请将间隔定时器时钟选择寄存器 0 (ITLCSEL0) 设置为计数时钟，并将间隔定时器比较寄存器 01 (ITLCMP01) 设置为比较值。当使用 FSXP 或软件触发（将 ITLCC0.CAPR 位设置为 1）作为捕获触发时，通道 2 和 3 可以以 16 位计数器模式使用。

输入捕获触发信号并将计数器值存储到间隔定时器捕获寄存器后，输出中断请求信号 (TML32_ITL_OR)，捕获完成标志 (ITLCC0.CAPF) 和捕获检测标志 (ITLS0.ITF0C) 被置为 1，并且这些标志值会一直保持，直到被显式清除为止 *1。可以通过将 ITLCC0.CAPFCR 位置为 1 来清除 ITLCC0.CAPF 标志。可以通过向 ITLS0 寄存器中的 ITF0C 标志写入 0 来清除它。由于捕获操作与计数器时钟同步运行，因此生成捕获触发信号的间隔应至少为计数器时钟周期的 5 个周期。如果在前一次生成捕获触发信号后的 2 个计数器时钟周期内再次生成捕获触发信号，则 ITLC0.CAPF 位可能不会被置位。

Note 1. If the value of the ITLS0 register is other than 0x00, interrupt operation does not proceed even when the capture detection flag (ITF0C) is set to 1 because the interrupt request signal (TML32_ITL_OR) is kept at the high level.

Figure 17.7 shows an example of capture operation.

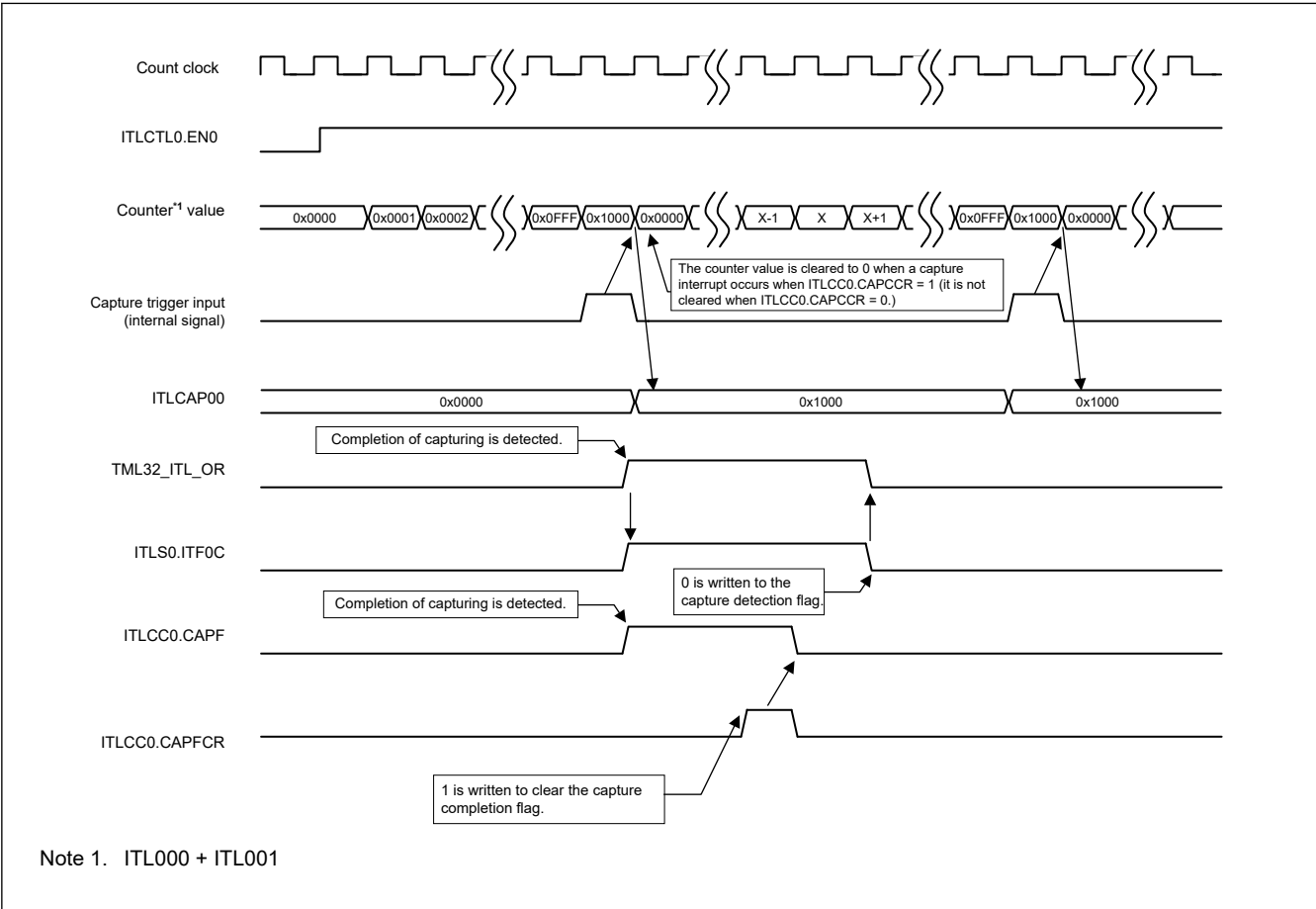


Figure 17.7 Example of capture operation

When the counter value matches the comparison value while the ITLCC0.CAPCCR bit in the ITLCC0 register is set to 1 (mode where the capture counter value is cleared following the completion of capture), counting of the next clock cycle clears the counter value. The counter value is not cleared in this way if the ITLCC0.CAPCCR bit is set to 0 (mode where the capture counter retains its value after the completion of capture).

17.3.5 Interrupt

Table 17.8 shows the interrupt sources in 8-bit, 16-bit, and 32-bit counter modes.

The ITF00 to ITF03 and ITF0C bits are interrupt status flags in the ITLS0 register. When any of the interrupt status flag is set, an interrupt request is output as the TML32_ITL_OR signal.

Table 17.8 Interrupt sources in 8-bit, 16-bit, and 32-bit counter modes (1 of 2)

Interrupt source	Interrupt condition in 8-bit counter mode	Interrupt condition in 16-bit counter mode	Interrupt condition in 32-bit counter mode
ITLS0.ITF00	Next rising edge of the counter clock after a compare match in channel 0	Next rising edge of the counter clock after a compare match in channels 0 and 1	Next rising edge of the counter clock after a compare match
ITLS0.ITF01	Next rising edge of the counter clock after a compare match in channel 1	Not generated	Not generated
ITLS0.ITF02	Next rising edge of the counter clock after a compare match in channel 2	Next rising edge of the counter clock after a compare match in channels 2 and 3	Not generated

注 1. 如果 ITLS0 寄存器的值不是 0x00，即使捕获检测标志 (ITF0C) 设置为 1，中断操作也不会进行，因为中断请求信号 (TML32_ITL_OR) 保持高电平。

图 17.7 显示了一个捕获操作的示例。

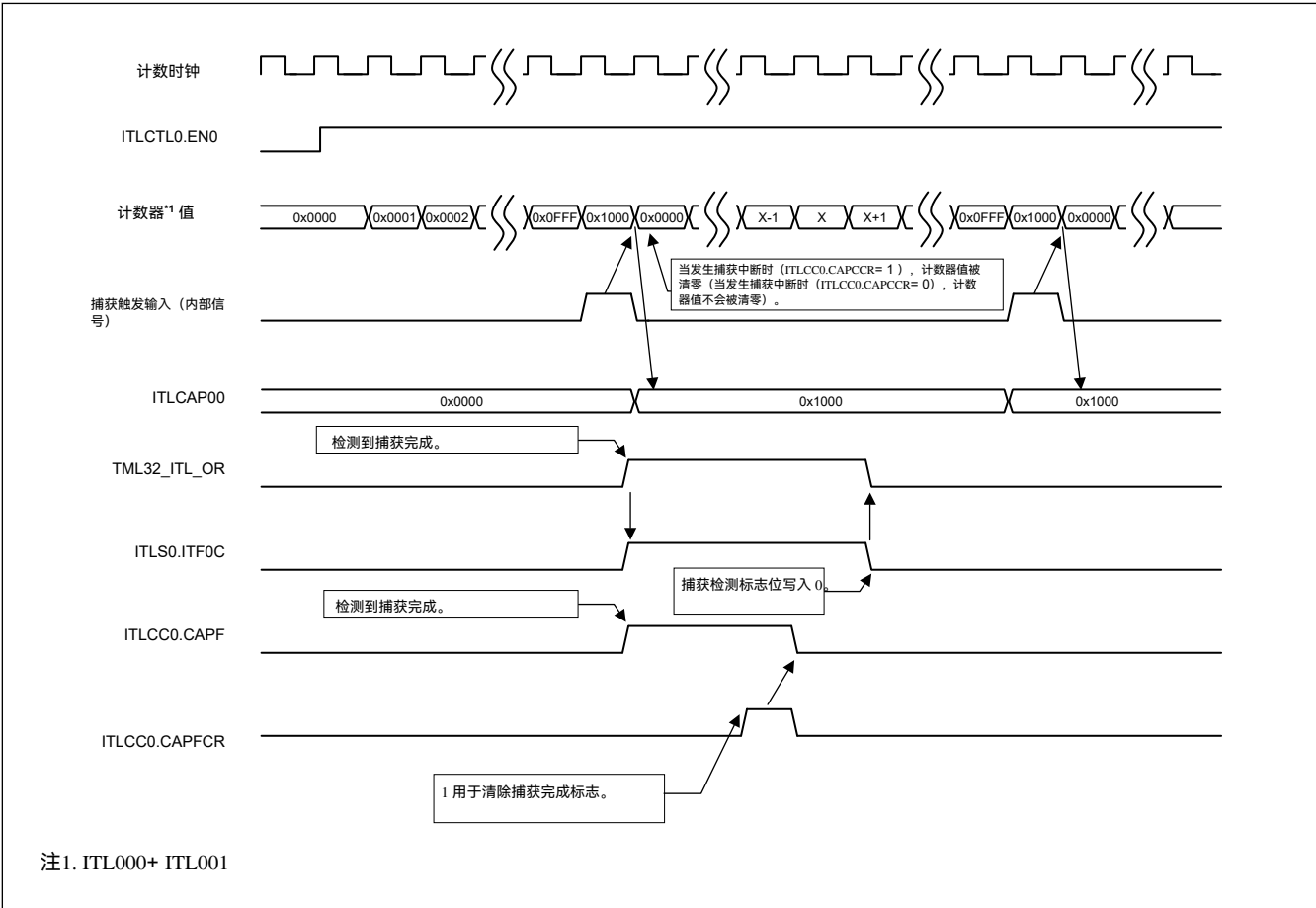


图 17.7 捕获操作示例

当计数器值与比较值匹配，且 ITLCC0 寄存器中的 ITLCC0.CAPCCR 位设置为 1（捕获完成后清除捕获计数器值）时，下一个时钟周期的计数会清除计数器值。如果 ITLCC0.CAPCCR 位设置为 0（捕获完成后保持捕获计数器值），则不会以这种方式清除计数器值。

17.3.5 中断

表 17.8 显示了 8 位、16 位和 32 位计数器模式下的中断源。

ITLS0 寄存器中的 ITF00 至 ITF03 和 ITF0C 位是中断状态标志。当任何一个中断状态标志被置位时，就会输出 TML32_ITL_OR 信号作为中断请求。

表 17.8 8 位、16 位和 32 位计数器模式下的中断源 (1/2)

中断源	8 位计数器模式下的中断条件	16 位计数器模式下的中断条件	32 位计数器模式下的中断条件
ITLS0.ITF00	通道 0 中比较匹配后，计数器时钟的下一个上升沿。	通道 0 和 1 中比较匹配后，计数器时钟的下一个上升沿	比较匹配后，倒计时时钟的下一个上升沿
ITLS0.ITF01	通道 1 中比较匹配后，计数器时钟的下一个上升沿。	未生成	未生成
ITLS0.ITF02	通道 2 中比较匹配后，计数器时钟的下一个上升沿。	通道 2 和 3 中比较匹配后，计数器时钟的下一个上升沿	未生成

Table 17.8 Interrupt sources in 8-bit, 16-bit, and 32-bit counter modes (2 of 2)

Interrupt source	Interrupt condition in 8-bit counter mode	Interrupt condition in 16-bit counter mode	Interrupt condition in 32-bit counter mode
ITLS0.ITF03	Next rising edge of the counter clock after a compare match in channel 3	Not generated	Not generated
ITLS0.ITF0C	Not generated; this is the case when the setting of the ITLCC0 register is 0x00.	Timing of storing the counter value in the capture register after a capture trigger is input.	Not generated; this is the case when the setting of the ITLCC0 register is 0x00.

If the value of the ITLS0 register is other than 0x00, the interrupt request signal (TML32_ITL_OR) is kept at the high level. Accordingly, the generation of an additional interrupt request (TML32_ITL_OR) does not proceed, even when a compare match or completion of capture is detected for an operating channel.

However, if the value of the ITLS0 register is not 0x00 after any bit in the ITLS0 register is set to 0 by an 8-bit memory manipulation instruction, a low-level pulse signal is output on the TML32_ITL_OR pin is set to 1. Accordingly, clearing a status flag in the ITLS0 register to 0 during interrupt processing or other processing enables the detection of an interrupt in response to another status bit having the setting 1. Figure 17.8 shows the relationship between clearing of the detection flags and the interval detection interrupt signal.

The following describes the operation shown in Figure 17.8.

When a compare match in channel 1 is detected while the value of the ITLS0 register is 0x00, the ITF01 flag is set to 1 and the interval detection interrupt signal (TML32_ITL_OR) is driven high. While the interval detection interrupt signal (TML32_ITL_OR) is kept at the high level, the generation of an additional interrupt request (TML32_ITL_OR) does not proceed even when a compare match or completion of capture is detected for an operating channel.

Note that if another detection flag is set to 1 immediately before clearing the ITLS0.ITF0x (x = 0, 1, 2, 3, C) flag bit to 0, the output of the TML32_ITL_OR pin temporarily goes to the low level after clearing of the given ITLS0.ITF0x flag bit.

<1> The ITLS0.ITF01 flag is set to 1 in response to a compare match in channel 1 and the interval detection interrupt signal (TML32_ITL_OR) are driven high. The interval detection interrupt processing is executed.

<2> Check which detection flag in the ITLS0 register is set to 1 from within the interval detection interrupt processing. In the case shown in Figure 17.8, the ITLS0.ITF01 and ITF00 flags being set to 1 can be confirmed.

<3> Clear the ITLS0.ITF01 and ITF00 flags detected in step 2 and write 00011100b to the ITLS0 register so that its value becomes 0x00.*1

<4> The respective processing sequences in response to the ITLS0.ITF01 and ITF00 flags being set to 1 are then executed.*1

Note 1. Missing an interrupt source can also be prevented by repeating the processing for clearing an interrupt source per flag.

<5> The ITLS0.ITF01 flag is set to 1 in response to a further compare match in channel 1 and the interval detection interrupt signal (TML32_ITL_OR) is driven high. The interval detection interrupt processing is executed.

<6> Check which detection flag in the ITLS0 register is set to 1 from within the interval detection interrupt processing. In the case shown in Figure 17.8, the ITLS0.ITF01 flag being set to 1 can be confirmed.

<7> Clear the ITLS0.ITF01 flag detected in step 6 and write 00011101b to the ITLS0 register so that its value becomes 0x00. Though the ITLS0.ITF00 flag is also set to 1 in response to the compare match in channel 0 at this time, the ITLS0.ITF00 flag is not cleared because the processing for the flag does not proceed.

<8> As the ITLS0.ITF00 flag is set to 1 at the time the ITLS0.ITF01 flag is cleared to 0 in step 7, the TML32_ITL_OR signal is temporarily driven low.

<9> The processing in response to the ITLS0.ITF00 flag being set to 1 is then executed.

<10> Check which detection flag in the ITLS0 register is set to 1 from within the interval detection interrupt processing. In the case shown in Figure 17.8, the ITLS0.ITF00 flag being set to 1 can be confirmed.

<11> Clear the ITLS0.ITF00 flag detected in step 11 and write 00011101b to the ITLS0 register so that its value becomes 0x00.

<12> The processing in response to the ITLS0.ITF00 flag being set to 1 is then executed.

表 17.8 8 位、16 位和 32 位计数器模式下的中断源 (2/2)

中断源	8 位计数器模式下的中断条件	16 位计数器模式下的中断条件	32 位计数器模式下的中断条件
ITLS0.ITF03	频道 3 中比较匹配后，倒计时时钟的下一个上升沿。	未生成	未生成
ITLS0.ITF0C	未生成；当 ITLCC0 寄存器的设置为 0 x00 时，就会出现这种情况。	输入捕获触发信号后，将计数器值存储到捕获寄存器中的时序。	未生成；当 ITLCC0 寄存器的设置未满足以下条件时，就会出现这种情况：0x00。

如果 ITLS0 寄存器的值不是 0x00，，则中断请求信号 (TML32_ITL_OR) 保持高电平。因此，即使检测到工作通道的比较匹配或捕获完成，也不会生成额外的中断请求 (TML32_ITL_OR)。

然而，如果在通过 8 位内存操作指令将 ITLS0 寄存器中的任何一位设置为 0 后，ITLS0 寄存器的值仍然不是 0x00，则会在 TML32_ITL_OR 引脚上输出一个低电平脉冲信号，并将该引脚设置为 1。因此，在中断处理或其他处理过程中，将 ITLS0 寄存器中的某个状态标志清除为 0，可以检测到响应于另一个状态位设置为 1 的中断。图 17.8 显示了检测标志清除与间隔检测中断信号之间的关系。

下面描述了图 17.8 所示的操作。

当通道 1 中检测到比较匹配且 ITLS0 寄存器值为 0x00 时，ITF01 标志位被置为 1，并且间隔检测中断信号 (TML32_ITL_OR) 被拉高。只要间隔检测中断信号 (TML32_ITL_OR) 保持高电平，即使检测到工作通道的比较匹配或捕获完成，也不会生成额外的中断请求 (TML32_ITL_OR)。

注意，如果在将 ITLS0.ITF0x (x = 0, 1, 2, 3, C) 标志位清除为 0 之前立即将另一个检测标志设置为 1，则在清除给定的 ITLS0.ITF0x 标志位之后，TML32_ITL_OR 引脚的输出会暂时变为低电平。

<1>当通道 1 中比较匹配时，ITLS0.ITF01 标志位被置为 1，并且间隔检测中断信号 (TML32_ITL_OR) 被拉高。执行间隔检测中断处理。

<2>检查 ITLS0 寄存器中哪个检测标志在间隔检测中断处理过程中被设置为 1。如图 17.8 所示，可以确认 ITLS0.ITF01 和 ITF00 标志被设置为 1。

<3>清除步骤 2 中检测到的 ITLS0.ITF01 和 ITF00 标志，并将 00011100b 写入 ITLS0 寄存器，使其值变为 0x00。*1

<4>当 ITLS0.ITF01 和 ITF00 标志设置为 1 时，相应的处理序列将执行。*1

注 1. 通过对每个标志重复清除中断源的处理，也可以防止丢失中断源。

<5>响应于通道 1 中进一步的比较匹配，ITLS0.ITF01 标志被设置为 1，并且间隔检测中断信号 (TML32_ITL_OR) 被拉高。执行间隔检测中断处理。

<6>检查在间隔检测中断处理过程中，ITLS0 寄存器中哪个检测标志被设置为 1。如图 17.8 所示，可以确认 ITLS0.ITF01 标志被设置为 1。

<7>清除步骤 6 中检测到的 ITLS0.ITF01 标志，并将 00011101b 写入 ITLS0 寄存器，使其值变为 0x00。尽管此时通道 0 中的比较匹配也使 ITLS0.ITF00 标志置为 1，但由于该标志的处理并未继续进行，因此 ITLS0.ITF00 标志不会被清除。

<8>在步骤 7 中，当 ITLS0.ITF01 标志被清除为 0 时，ITLS0.ITF00 标志被设置为 1，因此 TML32_ITL_OR 信号暂时被拉低。

<9>然后执行对 ITLS0.ITF00 标志设置为 1 的响应处理。

<10>在间隔检测中断处理过程中，检查 ITLS0 寄存器中的哪个检测标志设置为 1。

如图 17.8 所示，可以确认 ITLS0.ITF00 标志设置为 1。

<11>清除步骤 11 中检测到的 ITLS0.ITF00 标志，并将 00011101b 写入 ITLS0 寄存器，使其值变为 0x00。

<12>当 ITLS0.ITF00 标志设置为 1 时，将执行相应的处理。

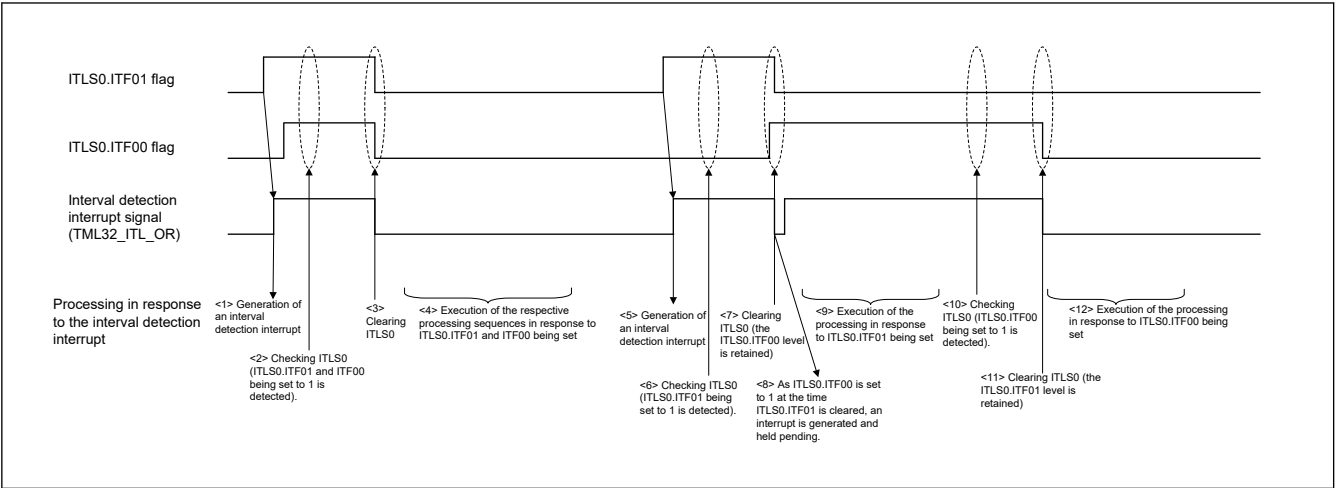


Figure 17.8 Example of clearing the detected flags

17.3.6 Interval Timer Setting Procedures

Table 17.9 shows the procedure for setting up the 32-bit interval timer.

Table 17.9 Procedure for starting the 32-bit interval timer

Step		Process	Detail
Starting the 32-bit interval timer	<1>	Start operation	—
	<2>	Select 8-bit, 16-bit, or 32-bit counter mode.	Set up the ITLCTL0.MD[1:0] bits.
	<3>	Select the count clock for the interval timer. Select the frequency division ratio for the count source. Specify a compare value.	Set up the ITLCSEL0.ISEL[2:0] bits. Set up the ITLFDIV0n register. Set up the ITLCMP0n register.
	<4>	When using the capture function - Enable capturing. - Clear the capture completion flag. - Select the count clock for the capture timer. - Specify the clearing of the counter values in channels 0 and 1 after completion of capturing. - Select the capture trigger.	Set the ITLCC0.CAPEN bit. Clear the ITLCC0.CAPFCR bit. Set up the ITLCSEL0.CSEL[2:0] bits. Clear the ITLCC0.CAPCCR bit. Set up the ITLCC0.CTRS[1:0] bits.
	<5>	When using an interrupt*1 - Clear the ITLS0.ITF0i interrupt status flags. - Set up masks for the ITLS0.ITF0i status flags.	Clear the ITLCC0.CAPFCR bit. Set up the ITLCSEL0.CSEL[2:0] bits.
	<6>	Start the 32-bit interval timer.	Set the ITLCTL0.ENi bit.
	<7>	Set the ITLCC0.CAPR bit to 1 when using the software capture trigger.	Set the ITLCC0.CAPR bit.
	<8>	Wait for an interrupt.	—

Note: n = 0, 1, i = 0 to 3

Note 1. When using this timer as an interval timer, do not mask the interrupts. When selecting a compare match in channels 2 and 3 as the capture trigger in 16-bit counter mode, set the ITLMKF0.MKF02 bit to 1 to specify a mask.

Table 17.10 shows the procedure for stopping the 32-bit interval timer.

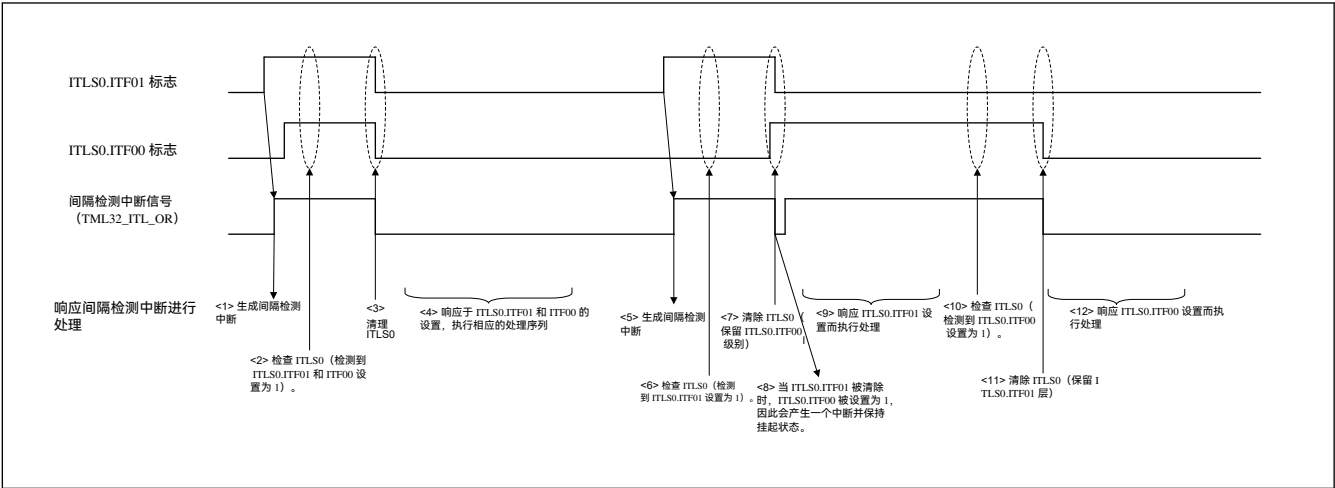


图 17.8 清除检测到的标志的示例

17.3.6 间隔定时器设置程序

表 17.9 显示了设置 32 位间隔定时器的步骤。

表 17.9 启动 32 位间隔定时器的步骤

步		过程	细节
开始 32 位间隔定时器	<1>	开始操作	—
	<2>	选择 8 位、16 位或 32 位计数器模式。	设置 ITLCTL0.MD[1:0] 位。
	<3>	选择间隔定时器的计数时钟。 选择计数源的频率分割比。 指定比较值。	设置 ITLCSEL0.ISEL[2:0] 位。 设置 ITLFDIV0n 寄存器。设置 I TLCMP0n 寄存器。
	<4>	使用捕获功能时 - 启用捕获功能。 - 清除捕获完成标志。 - 选择捕获计时器的计数时钟。 - 指定在捕获完成后清除通道 0 和 1 中的计数数值。 - 选择拍摄触发器。	设置 ITLCC0.CAPEN 位。 清除 ITLCC0.CAPFCR 位。 设置 ITLCSEL0.CSEL[2:0] 位。 清除 ITLCC0.CAPCCR 位。 设置 ITLCC0.CTRS[1:0] 位。
	<5>	使用中断时*1 - 清除 ITLS0.ITF0i 中断状态标志。 - 设置 ITLS0.ITF0i 状态标志的掩码。	清除 ITLCC0.CAPFCR 位。 设置 ITLCSEL0.CSEL[2:0] 位。
	<6>	启动 32 位间隔定时器。	设置 ITLCTL0.ENi 位。
	<7>	使用软件捕获触发器时, 将 ITLCC0.CAPR 位设置为 1。	设置 ITLCC0.CAPR 位。
	<8>	等待中断。	—

注意: n = 0, 1, i = 0 至 3

注 1: 当将此定时器用作间隔定时器时, 请勿屏蔽中断。在 16 位计数器模式下, 选择通道 2 和 3 中的比较匹配作为捕获触发时, 请将 ITLMKF0.MKF02 位设置为 1 以指定掩码。

表 17.10 显示了停止 32 位间隔定时器的步骤。

Table 17.10 Procedure for stopping the 32-bit interval timer

Step		Process	Detail
Stopping the 32-bit interval timer	<1>	Starting to stop the counter	—
	<2>	Set up masks for the ITLS0.ITF0i status flags. Clear the ITLS0.ITF0i interrupt status flags.	Set the ITLMKF0.MKF0i bits. Clear the ITLS0.ITF0i bits.
	<3>	When the capture function is in use Set up a mask for the ITLS0.ITF0C status flag. Clear the ITLS0.ITF0C interrupt status flag.	Set the ITLMKF0.MKF0C bit. Clear the ITLS0.ITF0C bit.
	<4>	Stop the 32-bit interval timer. Counting stops after one cycle of the source clock.	Clear the ITLCTL0.ENi bit.
	<5>	Completion of stopping the counter.	—

Note: n = 0, 1, i = 0 to 3

Table 17.11 shows the procedure for changing the operating mode of the 32-bit interval timer.

Table 17.11 Procedure for changing the operating mode of the 32-bit interval timer

Step		Process	Detail
Changing the operating mode of the 32-bit interval timer	<1>	Starting to change the operating mode	—
	<2>	Set up masks for the ITLS0.ITF0i status flags. Clear the ITLS0.ITF0i interrupt status flags.	Set the ITLMKF0.MKF0i bits. Clear the ITLS0.ITF0i bits.
	<3>	When the capture function is in use - Set up a mask for the ITLS0.ITF0C status flag. - Clear the ITF0C interrupt status flag.	Set the ITLMKF0.MKF0C bit. Clear the ITLS0.ITF0C bit.
	<4>	Disable all counters in the 32-bit interval timer.	Clear the ITLCTL0.EN0 to EN3 bits.
	<5>	Wait for at least one cycle of the count source until the timer is stopped.	Wait for stopping.
	<6>	Change the operating mode of the 32-bit interval timer. (see Table 17.9)	Make the setting to change the operating mode.
	<7>	Completion of changing the operating mode	—

Note: i = 0 to 3

表 17.10 停止 32 位间隔定时器的步骤

步		过程	细节
停止 32 位间隔定时器	<1>	开始停止计数	—
	<2>	设置 ITLS0.ITF0i 状态标志的掩码。清除 ITLS0.ITF0i 中断状态标志。	设置 ITLMKF0.MKF0i 位。清除 ITLS0.ITF0i 位。
	<3>	当捕获功能使用时，为 ITLS0.ITF0C 状态标志设置掩码。 清除 ITLS0.ITF0C 中断状态标志。	设置 ITLMKF0.MKF0C 位。清除 ITLS0.ITF0C 位。
	<4>	停止 32 位间隔定时器。 计数在源时钟一个周期后停止。	清除 ITLCTL0.ENi 位。
	<5>	停止计数器操作完成。	—

注意: n = 0, 1, i = 0至 3

表 17.11 显示了更改 32 位间隔定时器操作模式的步骤。

表 17.11 32 位间隔定时器工作模式变更步骤

步		过程	细节
更改 32 位间隔定时器的运行模式	<1>	开始改变操作模式	—
	<2>	设置 ITLS0.ITF0i 状态标志的掩码。清除 ITLS0.ITF0i 中断状态标志。	设置 ITLMKF0.MKF0i 位。清除 ITLS0.ITF0i 位。
	<3>	当使用捕获功能时 - 为 ITLS0.ITF0C 状态标志设置掩码。 - 清除 ITF0C 中断状态标志。	设置 ITLMKF0.MKF0C 位。清除 ITLS0.ITF0C 位。
	<4>	禁用 32 位间隔定时器中的所有计数器。	清除 ITLCTL0.EN0 到 EN3 位。
	<5>	至少等待计数源完成一个周期，直到计时器停止。	等待停止。
	<6>	更改 32 位间隔定时器的运行模式。 (见表17.9)	进行设置以更改操作模式。
	<7>	操作模式变更完成	—

注意: i = 0至 3

18. Independent Watchdog Timer (IWDT)

18.1 Overview

The Independent Watchdog Timer (IWDT) consists of a 14-bit down counter that must be serviced periodically to prevent counter underflow. The IWDT provides functionality to reset the MCU or to generate a non-maskable interrupt or an underflow interrupt. Because the timer operates with LOCO, it is particularly useful in returning the MCU to a known state as a fail-safe mechanism when the system runs out of control. The IWDT can be triggered automatically by a reset, underflow, refresh error, or a refresh of the count value in the registers.

Table 18.1 lists the IWDT specifications and Figure 18.1 shows a block diagram.

Table 18.1 IWDT specifications

Parameter	Description
Count source	IWDT clock (IWDTCLK) = The divided LOCO by 2
Clock division ratio	Division by 1, 16, 32, 64, 128, or 256
Counter operation	Counting down using a 14-bit down-counter
Condition for starting the counter	Counting automatically starts after a reset
Conditions for stopping the counter	- Reset (the down-counter and other registers return to their initial values) - Counter underflow or a refresh error is generated (counting restarts automatically).
Window function	Window start and end positions can be specified (refresh-permitted and refresh-prohibited periods)
Reset output sources	- Down-counter underflow - Refreshing outside the refresh-permitted period (refresh error).
Non-maskable interrupt/interrupt sources	- Down-counter underflow - Refreshing outside the refresh-permitted period (refresh error).
Reading the counter value	The down-counter value can be read by the IWDTSR register
Output signal (internal signal)	- Reset output - Interrupt request output - Sleep-mode count stop control output.
Auto start mode	Configurable to the following triggers: - Clock frequency division ratio after a reset (OFS0.IWDTCKS[3:0] bits) - Timeout period of the Independent Watchdog Timer (OFS0.IWDTTOPS[1:0] bits) - Window start position in the Independent Watchdog Timer (OFS0.IWDRPSS[1:0] bits) - Window end position in the Independent Watchdog Timer (OFS0.IWDRPES[1:0] bits) - Reset output or interrupt request output (OFS0.IWDRSTIRQS bit) - Down-count stop function at transition to Sleep, Snooze, or Software Standby mode (OFS0.IWDTSTPCTL bit).

The bus interface and registers operate with PCLKB, and the 14-bit counter and control circuits operate with IWDTCLK.

18.独立看门狗定时器（IWDT）

18.1 概述

独立看门狗定时器 (IWDT) 由一个 14 位递减计数器组成，必须定期进行维护以防止计数器下溢。IWDT 提供复位 MCU、生成不可屏蔽中断或下溢中断的功能。由于该定时器采用 LOCO 机制，因此在系统失控时，它作为一种故障安全机制，能够将 MCU 恢复到已知状态，从而发挥特别重要的作用。IWDT 可以由复位、下溢、刷新错误或寄存器中计数值的刷新自动触发。

表 18.1 列出了 IWDT 规格，图 18.1 显示了框图。

表 18.1 IWDT 规格

范围	描述
计数来源	IWDT 时钟 (IWDTCLK)=LOCO 除以 2
分频比	除以 1、16、32、64、128 或 256
反制行动	使用14位递减计数器进行倒计时
启动计数器的条件	●重置后自动开始计数
停止计数器的条件	●复位（递减计数器和其他寄存器恢复到其初始值） ●计数器下溢或刷新错误发生（自动计数重启）。
窗口函数	可以指定窗口的起始位置和结束位置（允许刷新和禁止刷新的时间段）
重置输出源	●向下反向底流 ●在允许刷新时间之外刷新（刷新错误）。
不可屏蔽中断/中断源	●向下反向底流 ●在允许刷新时间之外刷新（刷新错误）。
读取计数器值	可以通过 IWDTSR 寄存器读取向下计数器的值。
输出信号（内部信号）	●复位输出 ●中断请求输出 ●睡眠模式计数停止控制输出。
自动启动模式	可配置为以下触发器： ●复位后的时钟频率分频比（OFS0.IWDTCKS[3:0] 位） ●独立看门狗定时器的超时周期（OFS0.IWDTTOPS[1:0] 位） ●独立看门狗定时器中的窗口起始位置（OFS0.IWDRPSS[1:0] 位） ●独立看门狗定时器中的窗口结束位置（OFS0.IWDRPES[1:0] 位） ●复位输出或中断请求输出（OFS0.IWDRSTIRQS 位） ●在过渡到睡眠、贪睡或软件待机模式时停止倒计时功能（OFS0.IWDTSTPCTL 位）。

总线接口和寄存器使用 PCLKB 时钟，14 位计数器和控制电路使用 IWDTCLK 时钟。

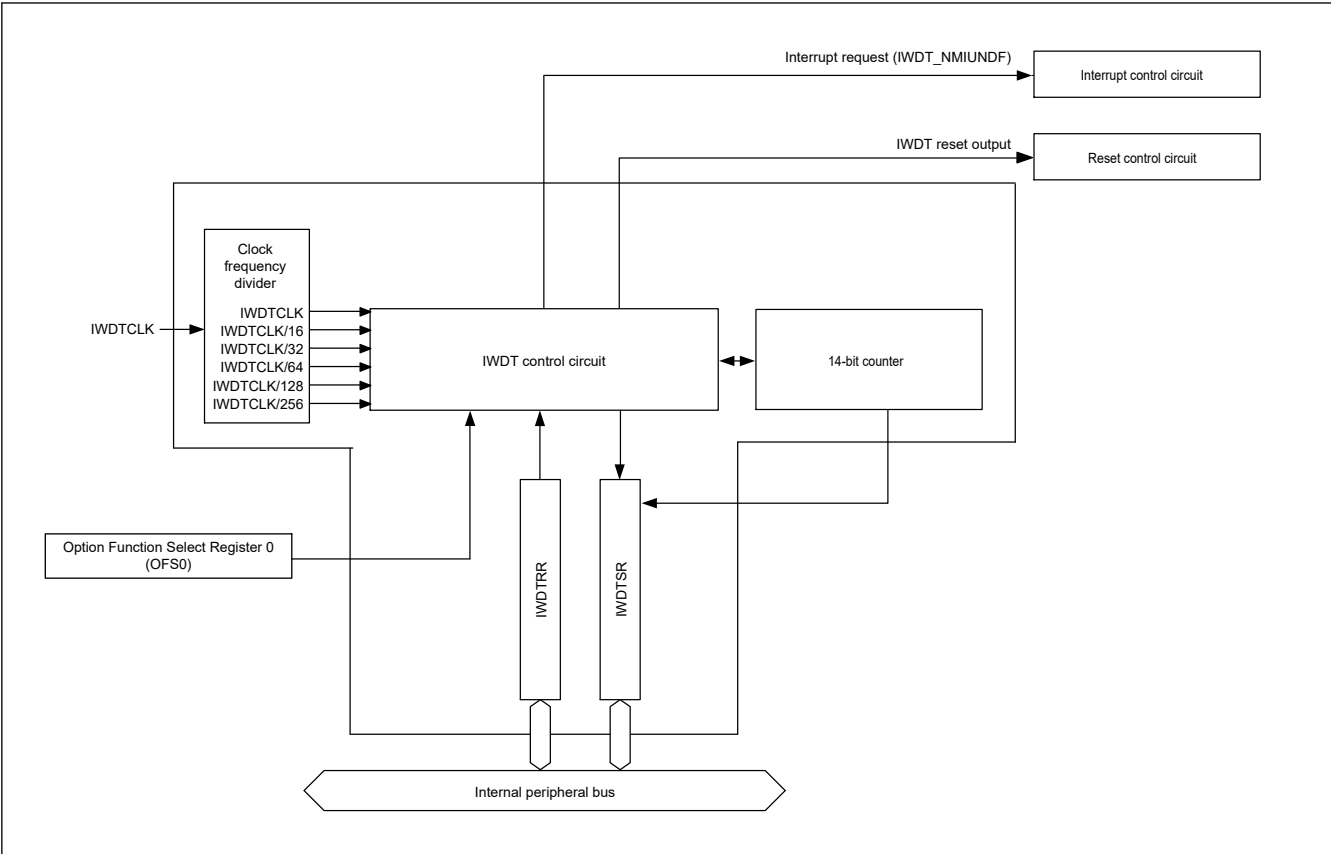
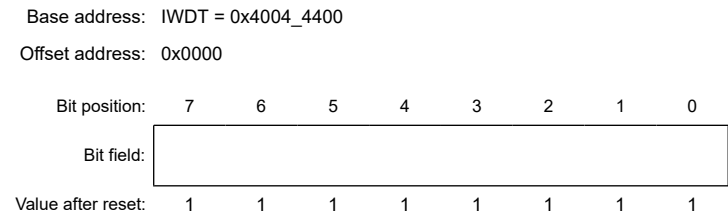


Figure 18.1 IWDT block diagram

18.2 Register Descriptions

18.2.1 IWDTRR : IWDT Refresh Register



Bit	Symbol	Function	R/W
7:0	n/a	The down-counter is refreshed by writing 0x00 and then writing 0xFF to this register	R/W

The IWDTRR register refreshes the down-counter of the IWDT. The down-counter of the IWDT is refreshed by writing 0x00 and then writing 0xFF to IWDTRR (refresh operation) within the refresh-permitted period. After the down-counter is refreshed, it starts counting down from the value selected in the IWDT Timeout Period Select bits (OFS0.IWDTTOPS[1:0]) in the Option Function Select Register 0 (OFS0).

When 0x00 is written, the read value is 0x00. When a value other than 0x00 is written, the read value is 0xFF. For details of the refresh operation, see [section 18.3.2. Refresh Operation](#).

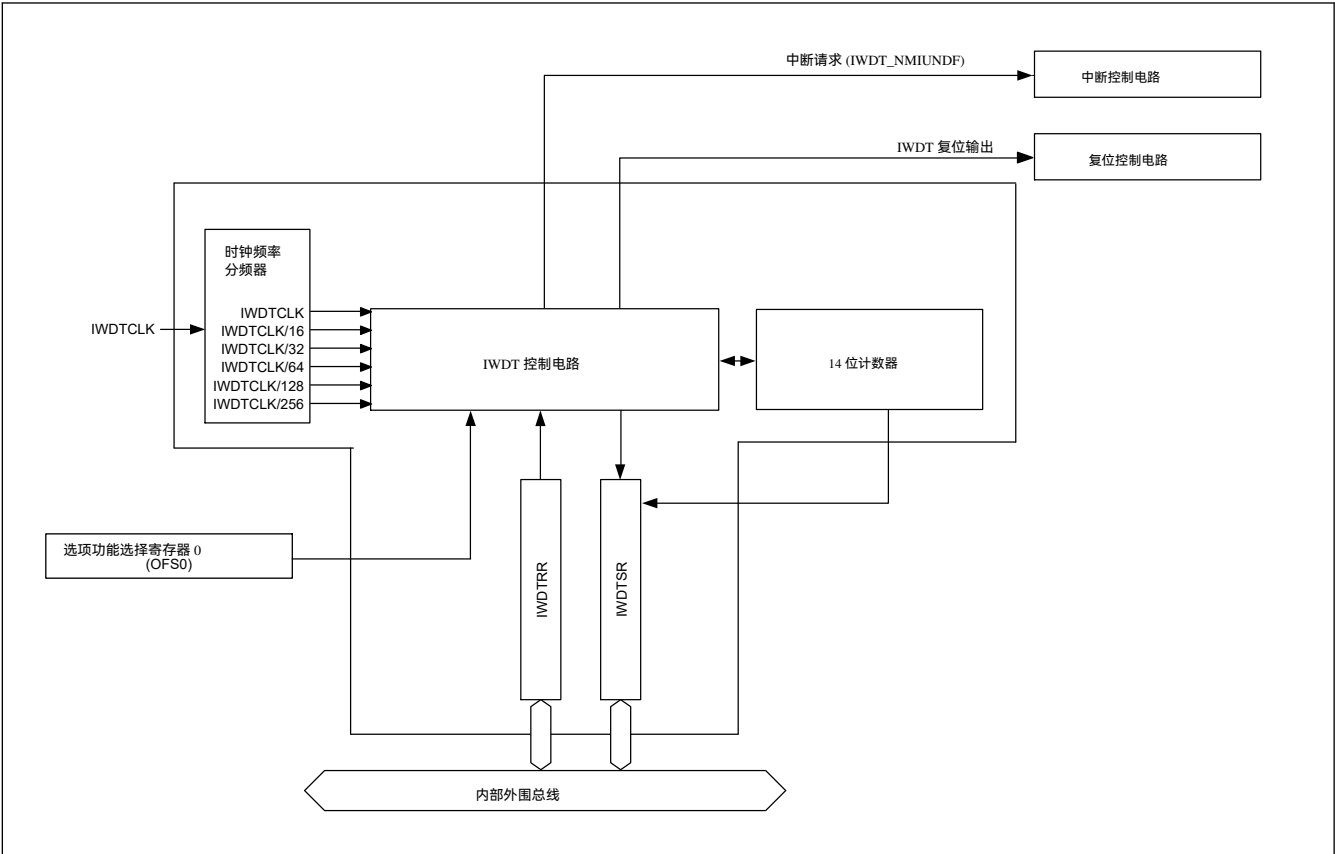
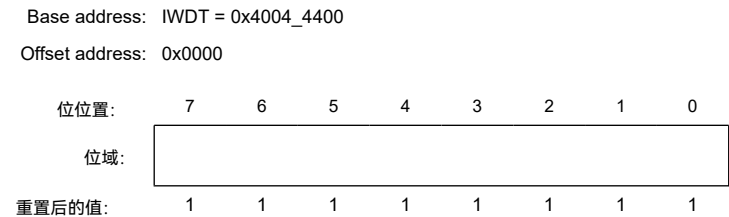


图 18.1 IWDT 框图

18.2 寄存器描述

18.2.1IWDTRR: IWDT 刷新寄存器

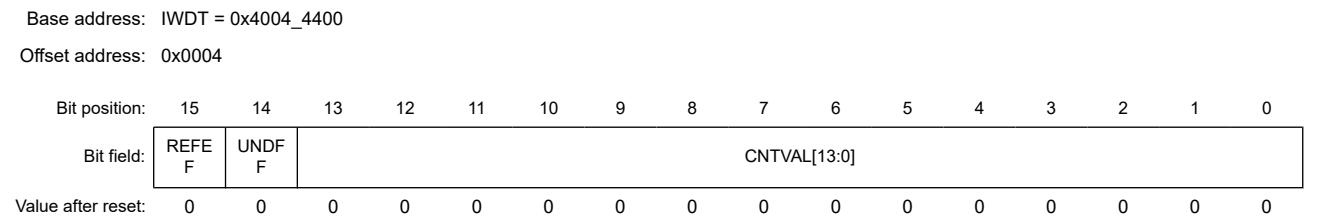


少量	象征	功能	R/W
7:0	n/a	通过向该寄存器写入0x00并写入 0xFF 来刷新递减计数器。	R/W

IWDTRR 寄存器刷新 IWDT 的递减计数器。在允许的刷新周期内，通过向 IWDTRR 写入0x00和 0xFF（刷新操作）来刷新 IWDT 的递减计数器。递减计数器刷新后，从选项功能选择寄存器 0 (OFS0) 中 IWDT 超时周期选择位 (OFS0.IWDTTOPS[1:0]) 所选择的值开始递减计数。

当写入0x00时，读取值为0x00。当写入的值不是0x00时，读取值为 0xFF。有关刷新操作的详细信息，请参见第 18.3.2 节“刷新操作”。

18.2.2 IWDTSR : IWDT Status Register



Bit	Symbol	Function	R/W
13:0	CNTVAL[13:0]	Down-counter Value Value counted by the down-counter	R
14	UNDF	Underflow Flag 0: No underflow occurred 1: Underflow occurred	R/W ^{*1}
15	REFE	Refresh Error Flag 0: No refresh error occurred 1: Refresh error occurred	R/W ^{*1}

Note 1. Only 0 can be written to clear the flag.

The IWDTSR register indicates the counter value of the down-counter and whether an underflow or refresh error occurred in the down-counter.

CNTVAL[13:0] bits (Down-counter Value)

Read the CNTVAL[13:0] bits to confirm the value of the down-counter. The read value might differ from the actual count by 1.

UNDF flag (Underflow Flag)

Read the UNDF flag to confirm whether an underflow occurred in the down-counter. The value 1 indicates that the down-counter underflowed. Write 0 to the UNDF flag to set the value to 0. Writing 1 has no effect.

Clearing of the UNDF flag takes (N + 2) IWDTCLK cycles and 2 PCLKB cycles. In addition, clearing of this flag is ignored for (N + 2) IWDTCLK cycles after an underflow. N is specified in the IWDTCKS[3:0] bits as follows:

- When OFS0.IWDTCKS[3:0] = 0x0, N = 1
- When OFS0.IWDTCKS[3:0] = 0x2, N = 16
- When OFS0.IWDTCKS[3:0] = 0x3, N = 32
- When OFS0.IWDTCKS[3:0] = 0x4, N = 64
- When OFS0.IWDTCKS[3:0] = 0xF, N = 128
- When OFS0.IWDTCKS[3:0] = 0x5, N = 256.

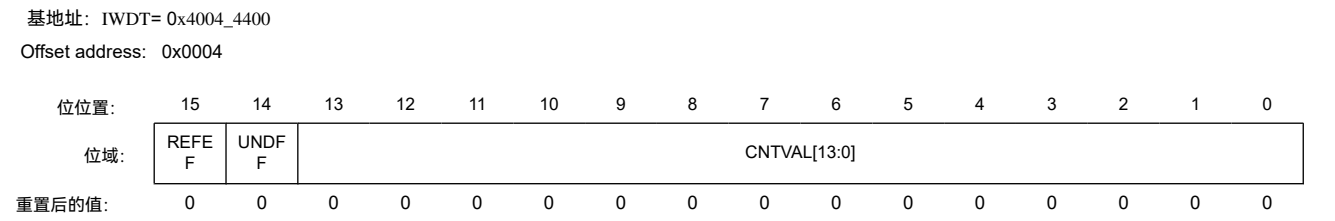
REFE flag (Refresh Error Flag)

Read the REFE flag to confirm whether a refresh error occurred. This indicates that a refresh operation was performed during a prohibited period. The value 1 indicates that a refresh error occurred. Write 0 to the REFE flag to set the value to 0. Writing 1 has no effect.

Clearing of the REFE flag takes (N + 2) IWDTCLK cycles and 2 PCLKB cycles. In addition, clearing of this flag is ignored for (N + 2) IWDTCLK cycles following a refresh error. N is specified in the IWDTCKS[3:0] bits as follows:

- When OFS0.IWDTCKS[3:0] = 0x0, N = 1
- When OFS0.IWDTCKS[3:0] = 0x2, N = 16
- When OFS0.IWDTCKS[3:0] = 0x3, N = 32
- When OFS0.IWDTCKS[3:0] = 0x4, N = 64
- When OFS0.IWDTCKS[3:0] = 0xF, N = 128
- When OFS0.IWDTCKS[3:0] = 0x5, N = 256.

18.2.2 IWDTSR: IWDT 状态寄存器



少量	象征	功能	R/W
13:0	CNTVAL[13:0]	向下计数价值 向下计数器计数的值	R
14	UNDF	下溢标志 0: 未发生下溢 1: 发生下溢	R/W ^{*1}
15	REFE	刷新错误标志 0: 未发生刷新错误 1: 发生刷新错误	R/W ^{*1}

注1: 只能写入0来清除标志。

IWDTSR 寄存器指示递减计数器的计数器值，以及递减计数器是否发生下溢或刷新错误。

CNTVAL[13:0] 位（向下计数器值）

读取 CNTVAL[13:0] 位以确认递减计数器的值。读取的值可能与实际计数相差 1。

UNDF 标志（下溢标志）

读取 UNDF 标志位以确认向下计数器是否发生下溢。值为 1 表示向下计数器发生下溢。向 UNDF 标志位写入 0 可将其值设置为 0。写入 1 无效。

清除 UNDF 标志需要(N + 2) 个 IWDTCLK 周期和 2 个 PCLKB 周期。此外，在发生下溢后的(N + 2)个 IWDTCLK 周期内，该标志的清除操作将被忽略。N在 IWDTCKS[3:0] 位中指定如下：

- When OFS0.IWDTCKS[3:0] = 0x0, N = 1
- When OFS0.IWDTCKS[3:0] = 0x2, N = 16
- When OFS0.IWDTCKS[3:0] = 0x3, N = 32
- When OFS0.IWDTCKS[3:0] = 0x4, N = 64
- When OFS0.IWDTCKS[3:0] = 0xF, N = 128
- When OFS0.IWDTCKS[3:0] = 0x5, N = 256.

REFE 标志（刷新错误标志）

读取 REFE 标志以确认是否发生刷新错误。这表示在禁止刷新期间执行了刷新操作。值为 1 表示发生刷新错误。向 REFE 标志写入 0 可将其值设置为 0。写入 1 无效。

清除 REFE 标志需要(N + 2)个 IWDTCLK 周期和 2 个 PCLKB 周期。此外，在刷新错误后的(N + 2)个 IWDTCLK 周期内，该标志的清除操作将被忽略。N在 IWDTCKS[3:0] 位中指定如下：

- When OFS0.IWDTCKS[3:0] = 0x0, N = 1
- When OFS0.IWDTCKS[3:0] = 0x2, N = 16
- When OFS0.IWDTCKS[3:0] = 0x3, N = 32
- When OFS0.IWDTCKS[3:0] = 0x4, N = 64
- When OFS0.IWDTCKS[3:0] = 0xF, N = 128
- When OFS0.IWDTCKS[3:0] = 0x5, N = 256.

18.2.3 OFS0 : Option Function Select Register 0

For information on the Option Function Select Register 0 (OFS0), see [section 6.2.1. OFS0 : Option Function Select Register 0](#).

IWDTTOPS[1:0] bits (IWDT Timeout Period Select)

The IWDTTOPS[1:0] bits select the timeout period, that is, the period until the down-counter underflows, from 128, 512, 1024, or 2048 cycles, taking the divided clock specified in the IWDTCKS[3:0] bits as 1 cycle.

After the down-counter is refreshed, the combination of the IWDTCKS[3:0] and IWDTTOPS[1:0] bits determines the number of IWDTCLK cycles until the counter underflows.

[Table 18.2](#) lists the relationship between the IWDTCKS[3:0] and IWDTTOPS[1:0] bit settings, the timeout period, and the number of IWDTCLK cycles.

Table 18.2 Timeout period settings

IWDTCKS[3:0] bits				IWDTTOPS [1:0] bits		Clock division ratio	Timeout period (number of cycles)	IWDTCLK cycles
b7	b6	b5	b4	b3	b2			
0	0	0	0	0	0	IWDTCLK	128	128
				0	1		512	512
				1	0		1024	1024
				1	1		2048	2048
0	0	1	0	0	0	IWDTCLK/16	128	2048
				0	1		512	8192
				1	0		1024	16384
				1	1		2048	32768
0	0	1	1	0	0	IWDTCLK/32	128	4096
				0	1		512	16384
				1	0		1024	32768
				1	1		2048	65536
0	1	0	0	0	0	IWDTCLK/64	128	8192
				0	1		512	32768
				1	0		1024	65536
				1	1		2048	131072
1	1	1	1	0	0	IWDTCLK/128	128	16384
				0	1		512	65536
				1	0		1024	131072
				1	1		2048	262144
0	1	0	1	0	0	IWDTCLK/256	128	32768
				0	1		512	131072
				1	0		1024	262144
				1	1		2048	524288

IWDTCKS[3:0] bits (IWDT Clock Frequency Division Ratio Select)

The IWDTCKS[3:0] bits specify the division ratio of the clock used for the down-counter. The division ratio can be selected from the IWDT clock (IWDTCLK) divided by 1, 16, 32, 64, 128, and 256. Combined with the IWDTTOPS[1:0] bit setting, the IWDT can be configured to a count period between 128 and 524,288 IWDTCLK cycles.

18.2.3 OFS0: 选项功能选择寄存器 0

有关选项功能选择寄存器 0 (OFS0) 的信息，请参阅 6.2.1 节。OFS0: 选项功能选择寄存器 0。

IWDTTOPS[1:0] 位 (IWDT 超时周期选择)

IWDTTOPS[1:0] 位选择超时周期，即向下计数器下溢之前的周期，取值为 128、512、1024 或 2048 个周期，以 IWD TCKS[3:0] 位中指定的分频时钟为 1 个周期。

递减计数器刷新后，IWDTCKS[3:0] 和 IWDTTOPS[1:0] 位的组合决定了计数器下溢之前 IWDTCLK 周期的数量。

表 18.2 列出了 IWDTCKS[3:0] 和 IWDTTOPS[1:0] 位设置、超时周期和 IWDTCLK 周期数之间的关系。

表 18.2 超时时间设置

IWDTCKS[3:0] bits				IWDTTOPS [1:0] bits		分频比	超时时间（周期数）	IWDTCLK周期
b7	b6	b5	b4	b3	b2			
0	0	0	0	0	0	IWDTCLK	128	128
				0	1		512	512
				1	0		1024	1024
				1	1		2048	2048
0	0	1	0	0	0	IWDTCLK/16	128	2048
				0	1		512	8192
				1	0		1024	16384
				1	1		2048	32768
0	0	1	1	0	0	IWDTCLK/32	128	4096
				0	1		512	16384
				1	0		1024	32768
				1	1		2048	65536
0	1	0	0	0	0	IWDTCLK/64	128	8192
				0	1		512	32768
				1	0		1024	65536
				1	1		2048	131072
1	1	1	1	0	0	IWDTCLK/128	128	16384
				0	1		512	65536
				1	0		1024	131072
				1	1		2048	262144
0	1	0	1	0	0	IWDTCLK/256	128	32768
				0	1		512	131072
				1	0		1024	262144
				1	1		2048	524288

IWDTCKS[3:0] 位 (IWDT 时钟频率分频比选择)

IWDTCKS[3:0] 位指定用于递减计数器的时钟分频比。分频比可从 IWDT 时钟 (IWDTCLK) 除以 1、16、32、64、128 和 256 中选择。结合 IWDTTOPS[1:0] 位设置，IWDT 的计数周期可配置为 128 到 524,288 个 IWDTCLK 周期。

IWDTRPES[1:0] bits (IWDT Window End Position Select)

The IWDTRPES[1:0] bits specify the window end position that indicates the refresh-permitted period. 75%, 50%, 25%, or 0% of the timeout period can be selected for the window end position. Set the window end position to a value less than the window start position (window start position > window end position). If the window start position is set to a value less than or equal to the window end position, the window start position setting is enabled and the window end position is set to 0%.

IWDTRPSS[1:0] bits (IWDT Window Start Position Select)

The IWDTRPSS[1:0] bits specify the window start position that indicates the refresh-permitted period. 100%, 75%, 50%, or 25% of the timeout period can be selected for the window start position. Set the window start position to a value greater than the window end position. If the window start position is set to a value less than or equal to the window end position, the window start position setting is enabled and the window end position is set to 0%.

Table 18.3 lists the counter values for the window start and end positions, and Figure 18.2 shows the refresh-permitted period set in the IWDTRPSS[1:0], IWDTRPES[1:0], and IWDTTOPS[1:0] bits.

Table 18.3 Relationship between the timeout period and window start and end counter values

IWDTTOPS[1:0] bits		Timeout period		Window start and end counter value			
b3	b2	Cycles	Counter value	100%	75%	50%	25%
0	0	128	0x007F	0x007F	0x005F	0x003F	0x001F
0	1	512	0x01FF	0x01FF	0x017F	0x00FF	0x007F
1	0	1024	0x03FF	0x03FF	0x02FF	0x01FF	0x00FF
1	1	2048	0x07FF	0x07FF	0x05FF	0x03FF	0x01FF

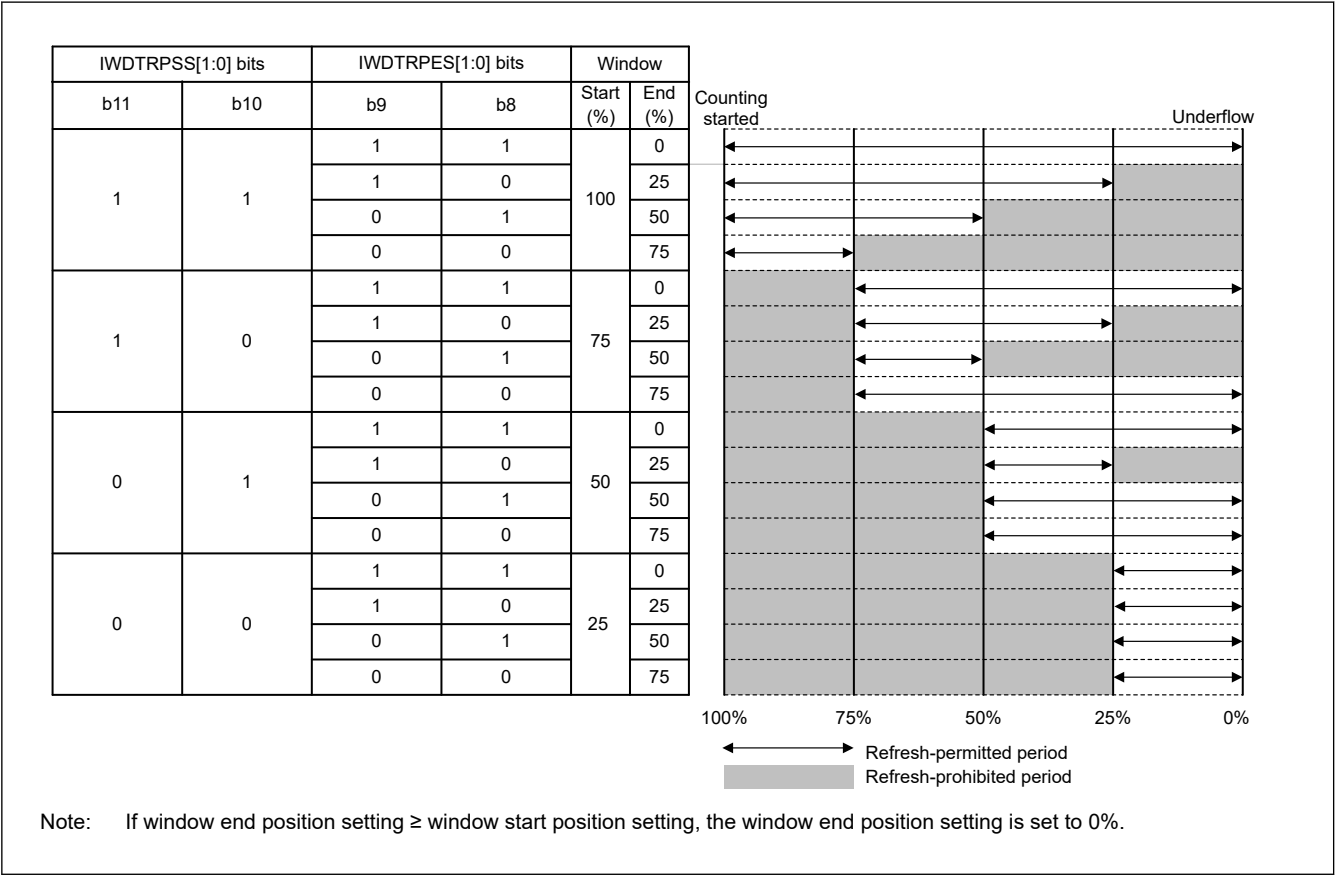


Figure 18.2 IWDTRPSS[1:0] and IWDTRPES[1:0] bit settings and refresh-permitted period

IWDTRSTIRQS bit (IWDT Reset Interrupt Request Select)

The IWDTRSTIRQS bit specifies the behavior when an underflow or a refresh error occurs. Setting 1 selects reset output. Setting 0 selects interrupt.

IWDTRPES[1:0] 位 (IWDT 窗口结束位置选择)

IWDTRPES[1:0] 位指定窗口结束位置，该位置指示允许刷新的周期。超时周期的75%、50%、25%或0%位可作为窗口结束位置。将窗口结束位置设置为小于窗口起始位置的值（窗口起始位置>窗口结束位置）。如果窗口起始位置设置为小于或等于窗口结束位置的值，则启用窗口起始位置设置，并将窗口结束位置设置为0%。

IWDTRPSS[1:0] 位 (IWDT 窗口起始位置选择)

IWDTRPSS 的[1:0] 位指定窗口起始位置，该位置指示允许刷新的周期。超时周期的100%、75%、50%或25%位可作为窗口起始位置。将窗口起始位置设置为大于窗口结束位置的值。如果将窗口起始位置设置为小于或等于窗口结束位置的值，则启用窗口起始位置设置，并将窗口结束位置设置为0%。

表 18.3 列出了窗口起始位置和结束位置的计数器值，图 18.2 显示了在 IWDTRPSS[1:0]、IWDTRPES[1:0]和 IWDT TOPS[1:0] 位中设置的允许刷新周期。

表 18.3 超时时间与窗口开始和结束计数器值之间的关系

IWDTTOPS[1:0] bits		超时时间		窗口起始和结束计数器值			
b3	b2	循环	计数器值	100%	75%	50%	25%
0	0	128	0x007F	0x007F	0x005F	0x003F	0x001F
0	1	512	0x01FF	0x01FF	0x017F	0x00FF	0x007F
1	0	1024	0x03FF	0x03FF	0x02FF	0x01FF	0x00FF
1	1	2048	0x07FF	0x07FF	0x05FF	0x03FF	0x01FF

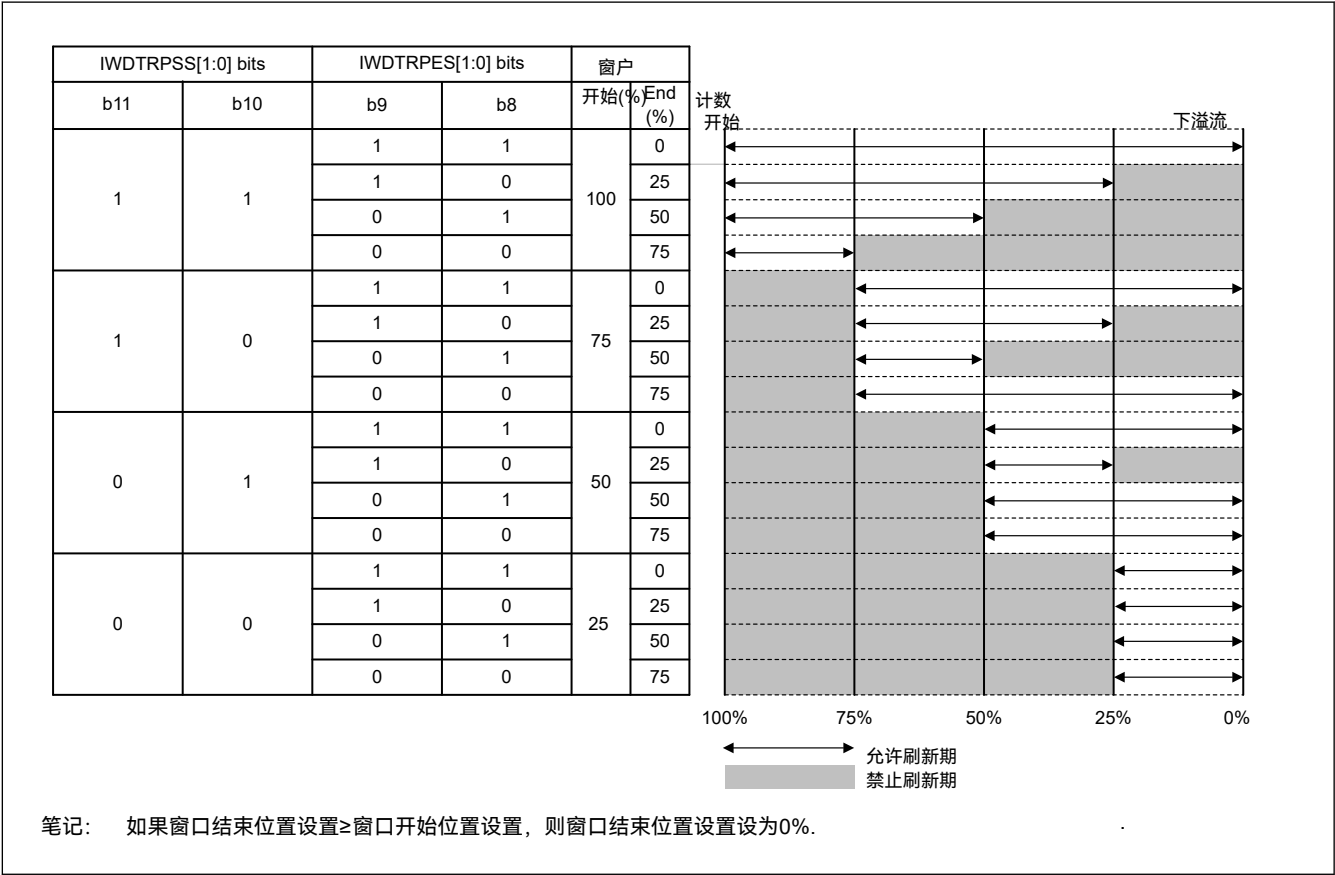


图 18.2 IWDTRPSS[1:0] 和 IWDTRPES[1:0] 位设置和允许刷新周期

IWDTRSTIRQS 位 (IWDT 复位中断请求选择)

IWDTRSTIRQS 位指定发生下溢或刷新错误时的行为。设置为 1 选择复位输出。设置 0 为中断。

IWDTSTPCTL bit (IWDT Stop Control)

The IWDTSTPCTL bit selects whether to stop counting on transition to Sleep, Snooze, or Software Standby mode.

18.3 Operation

18.3.1 Auto Start Mode

When the IWDT Start Mode Select bit (OFS0.IWDTSTRT) in the Option Function Select Register 0 is 0, auto start mode is selected, otherwise the IWDT is disabled.

Within the reset state, the setting values for the following in the Option Function Select Register 0 (OFS0) are set in the IWDT registers:

- Clock division ratio (OFS0.IWDTCKS[3:0])
- Window start and end positions (OFS0.IWDTRPSS[1:0], OFS0.IWDTRPES[1:0])
- Timeout period (OFS0.IWDTTOPS[1:0])
- Reset output or interrupt request (OFS0.IWDTRSTIRQS)

When the reset state is released, the counter automatically starts counting down from the value selected in the IWDT Timeout Period Select bits (OFS0.IWDTTOPS[1:0]).

After that, as long as the program continues normal operation and the counter is refreshed within the refresh-permitted period, the value in the counter is reset each time the counter is refreshed and down-counting continues. The IWDT does not output the reset signal as long as this procedure continues. However, if the counter underflows because the program crashed or because a refresh error occurred when an attempt is made to refresh outside the refresh-permitted period, the IWDT asserts the reset signal or non-maskable interrupt request/interrupt request (IWDT_NMIUNDF).

After the reset signal or non-maskable interrupt request/interrupt request is generated, the counter reloads the timeout period after counting for 1 cycle, the value of the timeout period is set in the down-counter and counting starts. The reset output or interrupt request output can be selected with the IWDT Reset Interrupt Request Select bit (OFS0.IWDTRSTIRQS). The interrupt enabled for operating the NMI can be selected with the IWDT Underflow/Refresh Error Interrupt Enable bit (NMIER.IWDTEN).

Figure 18.3 shows an example of operation under the following conditions:

- Auto start mode (OFS0.IWDTSTRT = 0)
- IWDT behavior selection: interrupt (OFS0.IWDTRSTIRQS = 0)
- Non-maskable Interrupt: IWDT Underflow/Refresh Error Interrupt Enabled (NMIER.IWDTEN = 1)
- The window start position is 75% (OFS0.IWDTRPSS[1:0] = 10b)
- The window end position is 25% (OFS0.IWDTRPES[1:0] = 10b).

IWDTSTPCTL 位 (IWDT 停止控制)

IWDTSTPCTL 位用于选择在过渡到睡眠、小睡或软件待机模式时是否停止计数。

18.3 操作

18.3.1 自动启动模式

当选项功能选择寄存器 0 中的 IWDT 启动模式选择位 (OFS0.IWDTSTRT) 为 0 时，选择自动启动模式；否则，禁用 IWDT。

在复位状态下，选项功能选择寄存器 0 (OFS0) 中以下各项的设置值将被设置：
IWDT注册：

- 时钟分频比（OFS0.IWDTCKS[3:0]）
- 窗口起始和结束位置（OFS0.IWDTRPSS[1:0], OFS0.IWDTRPES[1:0]）
- 超时时间(OFS0.IWDTTOPS[1:0])
- 复位输出或中断请求（OFS0.IWDTRSTIRQS）

当复位状态释放时，计数器会自动从 IWDT 超时周期选择位 (OFS0.IWDTTOPS[1:0]) 中选择的值开始倒计时。

此后，只要程序继续正常运行，且计数器在允许刷新的时间段内刷新，每次刷新计数器时，计数器的值都会被重置，从而继续向下计数。在此过程中，IWDT 不会输出复位信号。但是，如果由于程序崩溃或在允许刷新时间段之外尝试刷新时发生刷新错误而导致计数器下溢，则 IWDT 会输出复位信号或不可屏蔽中断请求/中断请求 (IWDT_NMIUNDF)。

复位信号或不可屏蔽中断请求/中断请求生成后，计数器计数一个周期后重新加载超时周期，并将超时周期的值置入递减计数器，然后重新开始计数。复位输出或中断请求输出可通过 IWDT 复位中断请求选择位 (OFS0.IWDTRSTIRQS) 进行选择。
可以通过 IWDT 下溢/刷新错误中断使能位 (NMIER.IWDTEN) 选择用于操作 NMI 的中断。

图 18.3 显示了在以下条件下运行的示例：

- 自动启动模式 (OFS0.IWDTSTRT= 0))
- IWDT 行为选择：中断 (OFS0.IWDTRSTIRQS= 0)
- 不可屏蔽中断：IWDT 下溢/刷新错误中断已启用 (NMIER.IWDTEN= 1)
- 窗口起始位置为75%(OFS0.IWDTRPSS[1:0] = 10b)
- 窗口结束位置为25% (OFS0.IWDTRPES[1:0] = 10b)。

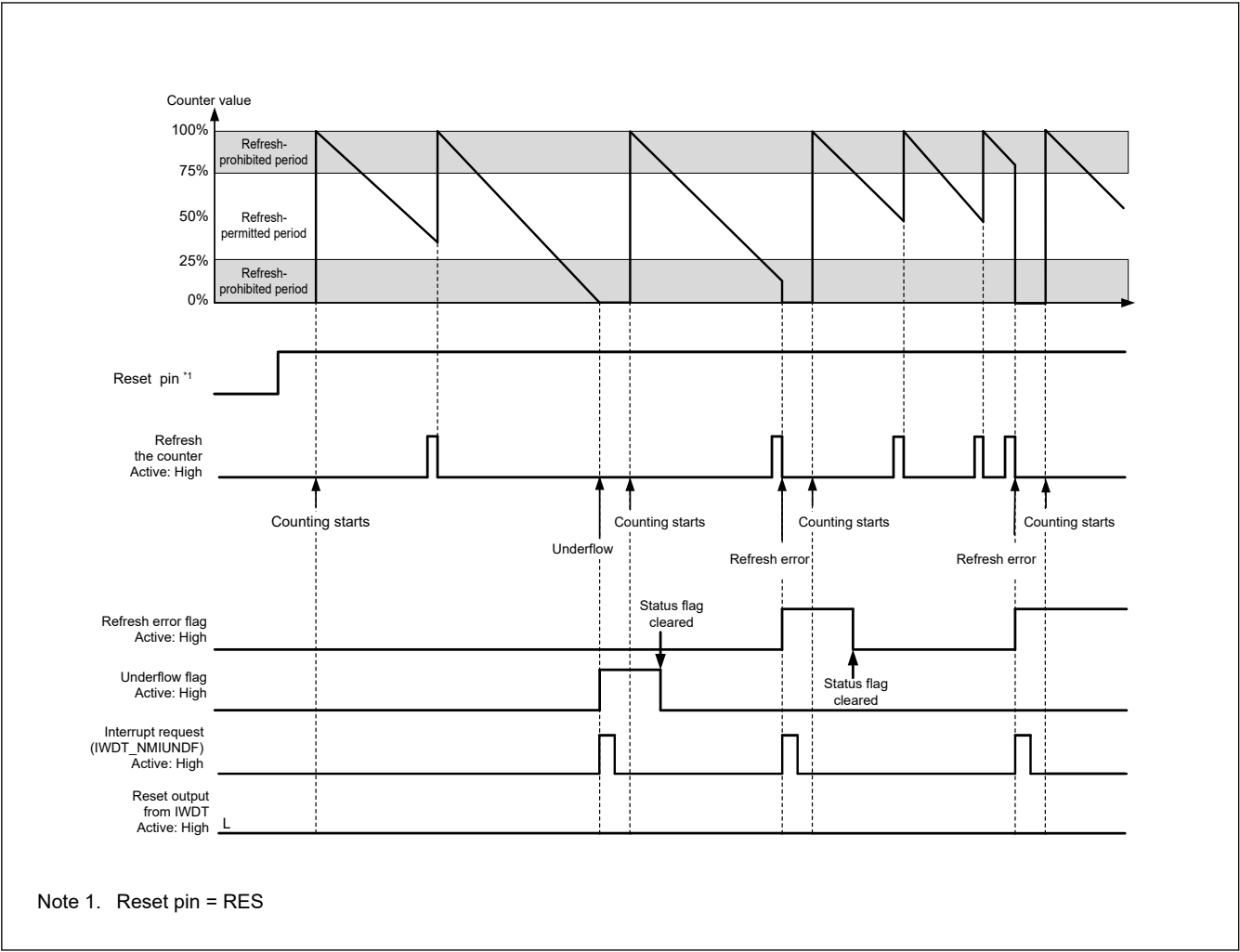


Figure 18.3 Operation example in auto start mode

18.3.2 Refresh Operation

To refresh the down counter and start the counting operation, write to the IWDT Refresh Register (IWDTRR) in the order of values from 0x00 to 0xFF. If a value other than 0xFF is written after 0x00, the down-counter is not refreshed. If an invalid value is written, refreshing is performed normally by writing to the IWDTRR register in the order of values from 0x00 to 0xFF.

When writes are made in the order of 0x00 (first time) → 0x00 (second time), and if 0xFF is written after that, the writing order 0x00 → 0xFF is satisfied. Writing 0x00 ((n - 1)th time) → 0x00 (nth time) → 0xFF is valid, and the refresh is performed correctly. Even when the first value written before 0x00 is not 0x00, correct refreshing is performed as long as the operation contains the write sequence of 0x00 → 0xFF.

Correct refreshing is also performed regardless of whether a register other than IWDTRR is accessed or IWDTRR is read between writing 0x00 and writing 0xFF to IWDTRR. Writes to refresh the counter must be made within the refresh-permitted period. Whether writing is done within the refresh-permitted period is determined when 0xFF is written. For this reason, correct refreshing is performed even when 0x00 is written outside the refresh-permitted period.

[Example write sequences that are valid to refresh the counter]

- 0x00 → 0xFF
- 0x00 ((n - 1)th time) → 0x00 (nth time) → 0xFF
- 0x00 → access to another register or read from IWDTRR → 0xFF.

[Example write sequences that are not valid to refresh the counter]

- 0x23 (a value other than 0x00) → 0xFF

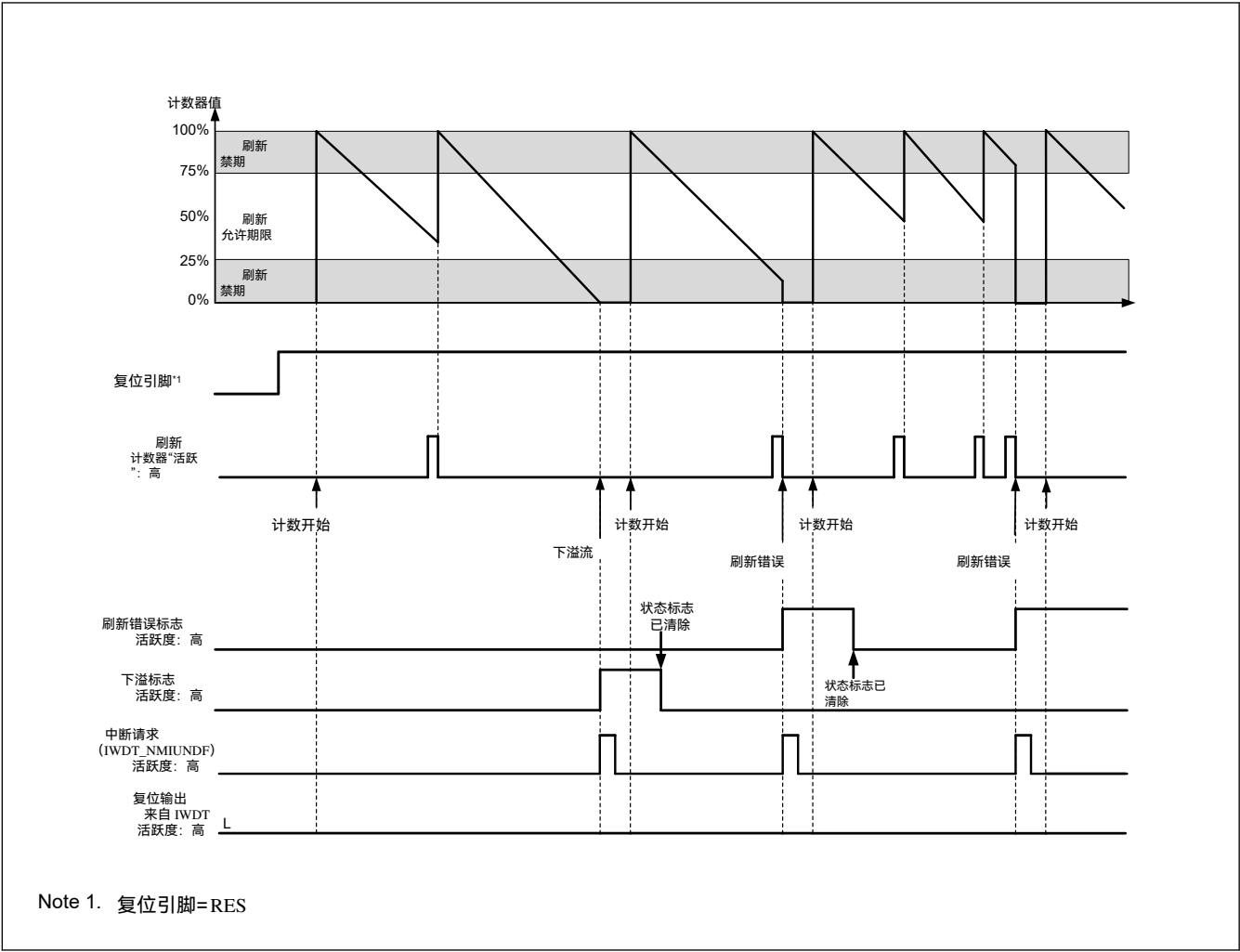


图 18.3 自动启动模式下的操作示例

18.3.2 刷新操作

要刷新递减计数器并开始计数操作，请按从0x00到0xFF的顺序写入IWDT刷新寄存器(IWDTRR)。如果在0x00之后写入的值不是0xFF，则递减计数器不会刷新。如果写入了无效值，则会按从0x00到0xFF的顺序正常写入IWDTRR寄存器来刷新计数器。

当写入顺序为0x00（第一次）→ 0x00（第二次）时，如果之后写入了0xFF，则满足写入顺序0x00 → 0xFF。写入0x00 ((n - 1)th time) → 0x00（第n次）→ 0xFF是有效的，刷新操作正确执行。即使在0x00之前写入的第一个值0x00，不正确，只要操作包含0x00 → 0xFF的写入序列，刷新操作也会执行。

无论在向IWDTRR写入0x00和写入0xFF之间访问了除IWDTRR之外的其他寄存器，还是读取了IWDTRR，都会执行正确的刷新操作。刷新计数器的写入操作必须在允许刷新的时间段内完成。写入操作是否在允许刷新的时间段内完成，取决于写入0xFF的操作。因此，即使在允许刷新的时间段之外写入0x00，也会执行正确的刷新。

[用于刷新计数器的有效写入序列示例]

- 0x00 → 0xFF
- 0x00 ((n - 1)th time) → 0x00 (nth time) → 0xFF
- 0x00 → 访问另一个寄存器或从IWDTRR读取数据 → 0xFF.

[示例写入序列无效，无法刷新计数器]

- 0x23（除0x00）→ 0xFF之外的值)

- 0x00 → 0x54 (a value other than 0xFF)
- 0x00 → 0xAA (0x00 and a value other than 0xFF) → 0xFF.

After 0xFF is written to the IWDTRR register, refreshing the counter requires up to 4 cycles of the signal for counting the IWDT Clock Frequency Division Ratio Select bits (OFS0.IWDTCKS[3:0]) to determine how many cycles of the IWDT clock (IWDTCCLK) make up 1 cycle for counting. To meet this requirement, writing 0xFF to the IWDTRR must be completed 4 count cycles before the end of the refresh-permitted period or a down-counter underflow. The value of the counter can be checked with the counter bits (IWDTSR.CNTVAL[13:0]).

[Example refreshing timings]

- When the window start position is set to 0x01FF, even if 0x00 is written to IWDTRR before 0x01FF is reached at (0x0202, for example), refreshing occurs if 0xFF is written to IWDTRR after the value of the IWDTSR.CNTVAL[13:0] bits reaches 0x01FF
- When the window end position is set to 0x01FF, refreshing occurs if 0x0203 (4 count cycles before 0x01FF) or a greater value is read from the IWDTSR.CNTVAL[13:0] bits immediately after writing 0x00 → 0xFF to IWDTRR
- When the refresh-permitted period continues until count 0x0000, refreshing can be performed immediately before an underflow. In this case, if 0x0003 (4 count cycles before an underflow) or a greater value is read from the IWDTSR.CNTVAL[13:0] bits immediately after writing 0x00 → 0xFF to IWDTRR, no underflow occurs and refreshing is performed.

Figure 18.4 shows the IWDT refresh-operation waveforms when PCLKB > IWDTCCLK and the clock division ratio is IWDTCCLK.

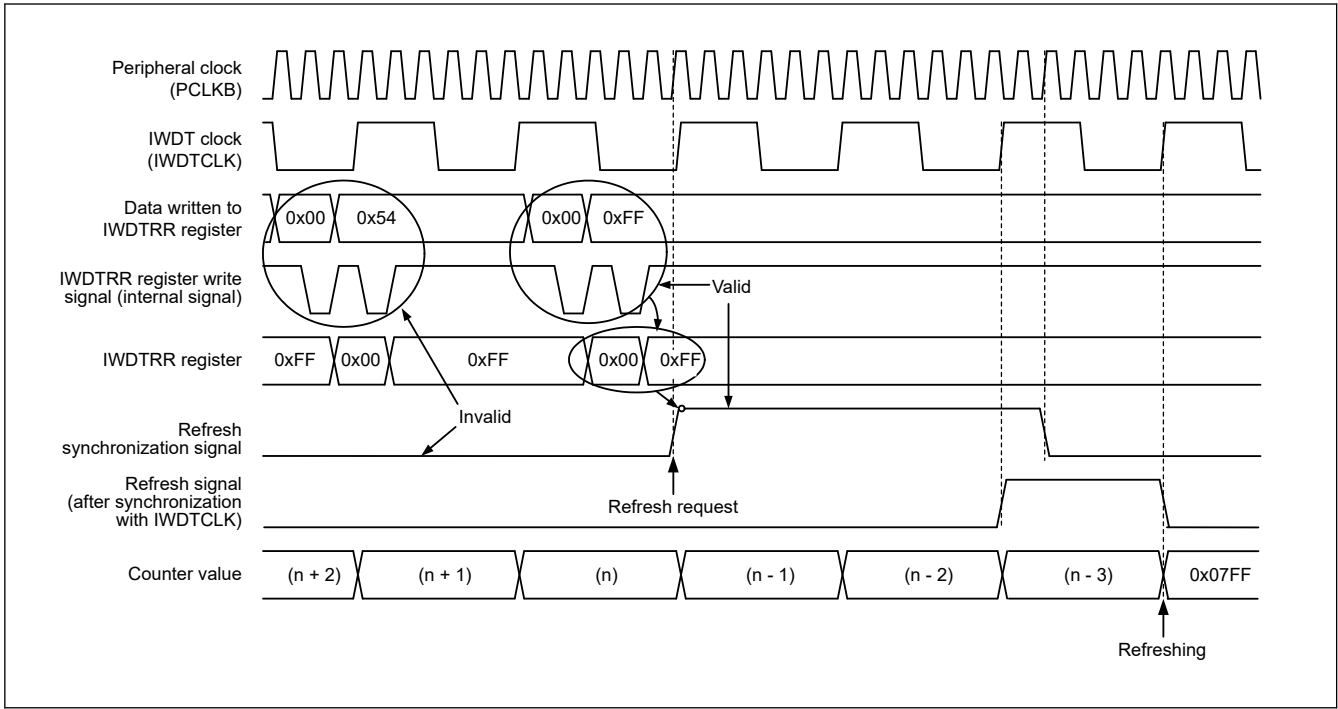


Figure 18.4 IWDT refresh operation waveforms when OFS0.IWDTCKS[3:0] = 0000b, OFS0.IWDTTOPS[1:0] = 11b

18.3.3 Status Flags

The refresh error (IWDTSR.REFEF) and underflow (IWDTSR.UNDF) flags retain the source of the interrupt request from the IWDT. Therefore, after a release from the interrupt request generation, read the IWDTSR.REFEF and UNDF flags to check for the interrupt source. For each flag, writing 0 clears the bit and writing 1 has no effect.

Leaving the status flags unchanged does not affect operation. If the flags are not cleared at the time of the next interrupt request from the IWDT, the earlier interrupt source is cleared and the new interrupt source is written. For the time period between when 0 is written in each flag and when its value is reflected, see [section 18.2.2. IWDTSR: IWDT Status Register](#).

- 0x00 → 0x54 (非 0xFF 值)
- 0x00 → 0xAA (0x00 和除 0xFF) → 0xFF 之外的值

在将 0xFF 写入 IWDTRR 寄存器后，刷新计数器需要最多 4 个周期的信号来计数 IWDT 时钟频率分频比选择位 (OFS0.IWDTCKS[3:0])，以确定 IWDT 时钟 (IWDTCCLK) 的多少个周期构成一个计数周期。为了满足此要求，必须在允许刷新周期结束或计数器下溢之前 4 个计数周期内完成向 IWDTRR 写入 0xFF 的操作。可以使用计数器位 (IWDTSR.CNTVAL[13:0]) 检查计数器的值。

[刷新时间示例]

- 当窗口起始位置设置为 0x01FF 时 (例如，即使在到达 0x0202 之前，0x00 已写入 IWDTRR)，如果在 IWDTRR.CNTVAL[13:0] 位的值达到 0x01FF 之后，向 IWDTRR 写入 0xFF，则会发生刷新。
- 当窗口结束位置设置为 0x01FF 时，如果从 IWDTSR.CNTVAL 读取到 0x0203 (在 0x01FF 之前 4 个计数周期) 或更大的值，则刷新发生。[13:0] 位紧随写入 0x00 → 0xFF 到 IWDTRR 之后。
- 当允许刷新的周期持续到计数 0x0000 时，可以在发生下溢之前立即执行刷新。在这种情况下，如果在向 IWDTRR 写入 0x00 → 0xFF 之后立即从 IWDTSR.CNTVAL[13:0] 位读取 0x0003 (下溢前 4 个计数周期) 或更大的值，则不会发生下溢，并且可以执行刷新。

图 18.4 显示了当 PCLKB > IWDTCCLK 且时钟分频比为 IWDTCCLK 时 IWDT 刷新操作波形。

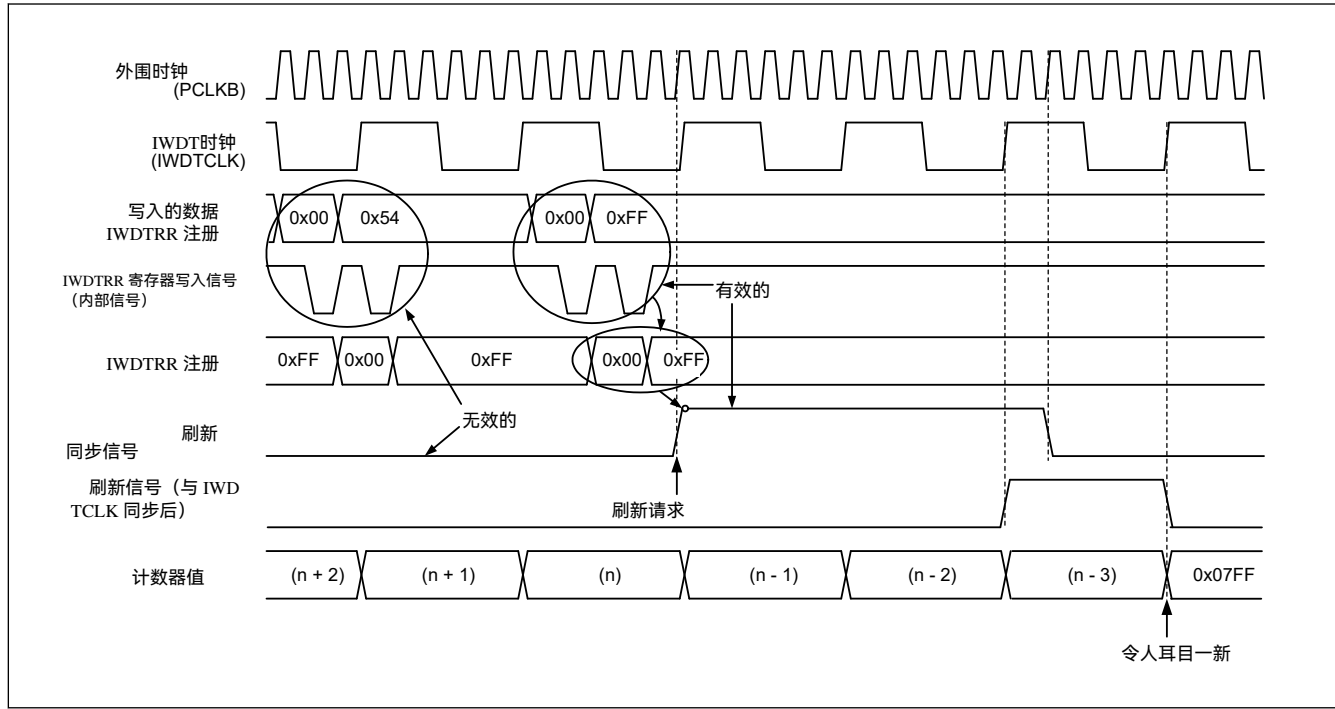


图 18.4 当 OFS0.IWDTCKS[3:0] = 0000b, OFS0.IWDTTOPS[1:0] = 11b 时，IWDT 刷新操作波形

18.3.3 状态标志

刷新错误标志 (IWDTSR.REFEF) 和下溢标志 (IWDTSR.UNDF) 用于保存来自 IWDT 的中断请求源。因此，在中断请求生成结束后，读取 IWDTSR.REFEF 和 UNDF 标志以检查中断源。对于每个标志，写入 0 会清除该位，写入 1 则无效。

保持状态标志不变不会影响操作。如果在 IWDT 发出下一个中断请求时标志未被清除，则清除先前的中断源并写入新的中断源。有关每个标志写入 0 到其值被反映的时间间隔，请参见 18.2.2 节。IWDTSR: IWDT 状态寄存器。

18.3.4 Reset Output

When the IWDT Reset Interrupt Request Select bit (OFS0.IWDTRSTIRQS) in the Option Function Select Register 0 (OFS0) is set to 1, a reset signal is output when an underflow in the counter or a refresh error occurs. Counting down automatically starts after the reset output.

18.3.5 Interrupt Sources

When the IWDT Reset Interrupt Request Select bit (OFS0.IWDTRSTIRQS) in the Option Function Select Register 0 (OFS0) is set to 0, an interrupt (IWDT_NMIUNDF) signal occurs when an underflow in the counter or a refresh error occurs. This interrupt can be used as a non-maskable interrupt or an interrupt. For details, see [section 11, Interrupt Controller Unit \(ICU\)](#).

Table 18.4 IWDT interrupt source

Name	Interrupt source	Interrupt to CPU	Start DTC
IWDT_NMIUNDF	- Down-counter underflow - Refresh error	Possible	Not possible

18.3.6 Reading the Down-Counter Value

As the counter is a IWDT clock (IWDTCCLK), the counter value cannot be read directly. The IWDT synchronizes the counter value with the peripheral clock (PCLKB) and stores it in the down-counter value bits (IWDTSR.CNTVAL[13:0]) of the IWDT Status Register. Check these bits to obtain the counter value indirectly.

Reading the counter value requires multiple PCLKB clock cycles (up to 4 clock cycles), and the read counter value might differ from the actual counter value by a value of one count.

Figure 18.5 shows the processing for reading the IWDT counter value when PCLKB > IWDTCCLK and the clock division ratio is IWDTCCLK.

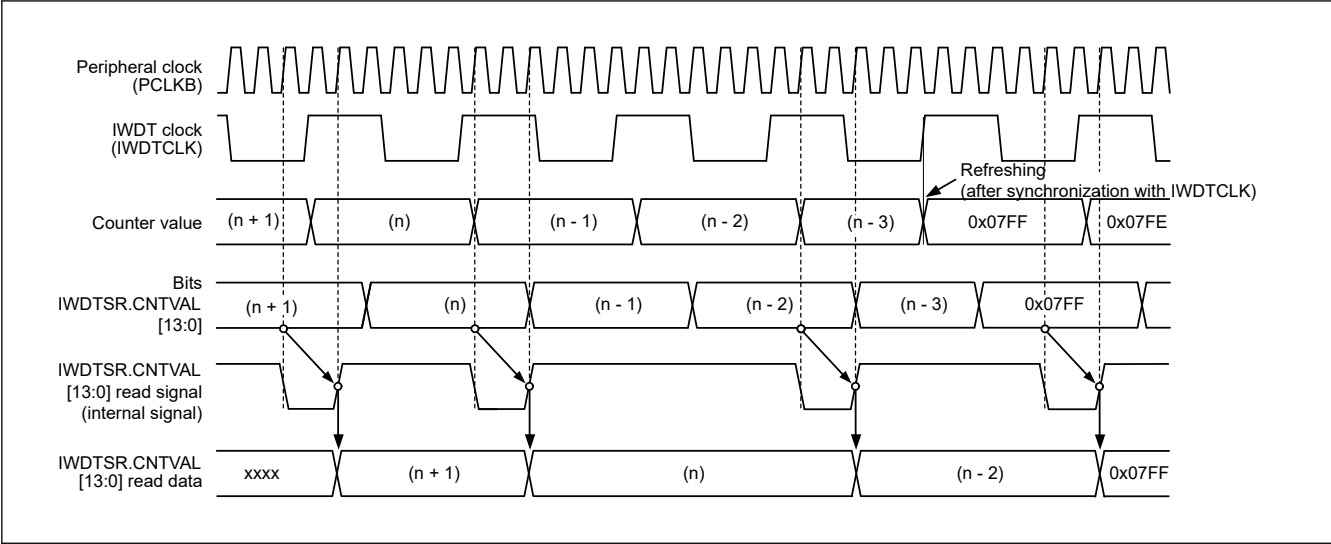


Figure 18.5 Processing for reading IWDT counter value when OFS0.IWDTCKS[3:0] = 0000b, OFS0.IWDTTOPS[1:0] = 11b

18.4 Usage Notes

18.4.1 Refresh Operations

While configuring the refresh time, consider variations in the range of errors given the accuracy of PCLKB and IWDTCCLK. Set values that ensure refreshing is possible.

18.3.4 复位输出

当选项功能选择寄存器 0 (OFS0) 中的 IWDT 复位中断请求选择位 (OFS0.IWDTRSTIRQS) 设置为 1 时，如果计数器发生下溢或刷新错误，则会输出复位信号。复位输出后，倒计时自动开始。

18.3.5 中断源

当选项功能选择寄存器 0 (OFS0) 中的 IWDT 复位中断请求选择位 (OFS0.IWDTRSTIRQS) 设置为 0 时，如果计数器发生下溢或刷新错误，则会产生中断 (IWDT_NMIUNDF) 信号。此中断可用作不可屏蔽中断或普通中断。有关详细信息，请参阅第 11 节“中断控制器单元 (ICU)”。

表 18.4 IWDT 中断源

Name	中断源	中断到 CPU	启动 DTC
IWDT_NMIUNDF	- 向下反向底流 - 刷新错误	可能的	不可能

18.3.6 读取向下计数器值

由于计数器是 IWDT 时钟 (IWDTCCLK)，因此无法直接读取计数器值。IWDT 将计数器值与外设时钟 (PCLKB) 同步，并将其存储在 IWDT 状态寄存器的减计数器值位 (IWDTSR.CNTVAL[13:0]) 中。通过检查这些位可以间接获得计数器值。

读取计数器值需要多个 PCLKB 时钟周期（最多 4 个时钟周期），读取的计数器值可能与实际计数器值相差 1 个计数。

图 18.5 显示了当 PCLKB>IWDTCCLK 且时钟分频比为 IWDTCCLK 时读取 IWDT 计数器值的处理过程。

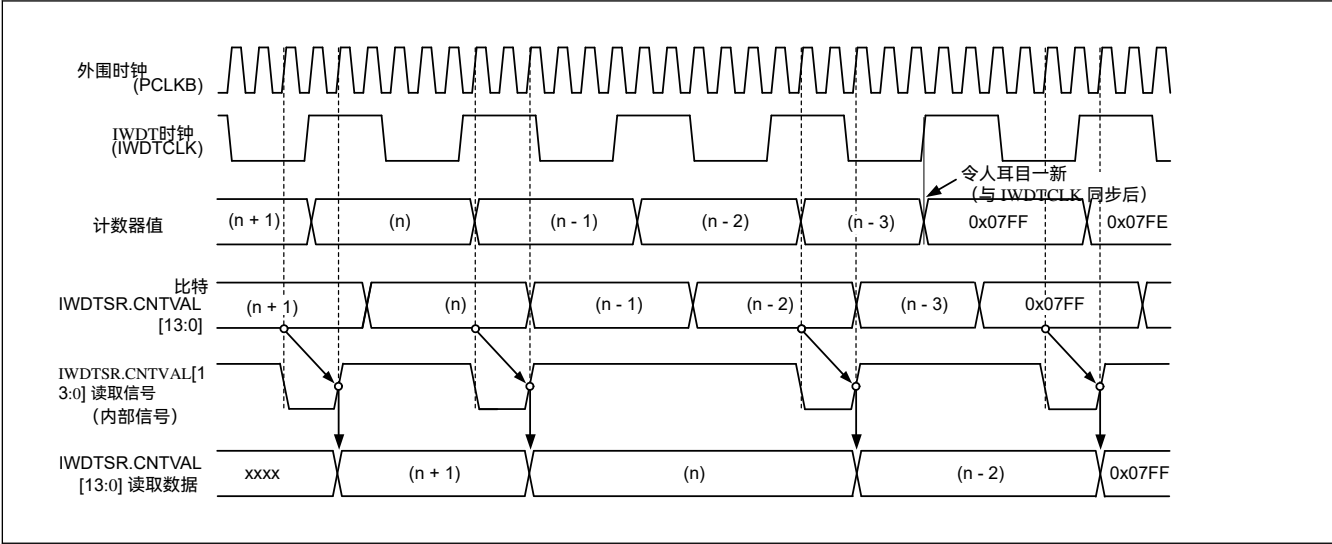


图 18.5 当 OFS0.IWDTCKS[3:0] = 0000b 时读取 IWDT 计数器值的处理
OFS0.IWDTTOPS[1:0] = 11b

18.4 使用说明

18.4.1 刷新操作

配置刷新时间时，考虑到 PCLKB 和 IWDTCCLK 的精度，请考虑误差范围的变化。设置确保可以刷新的值。

18.4.2 Clock Division Ratio Setting

Satisfy the frequency of the peripheral module clock (PCLKB) $\geq 4 \times$ (the frequency of the count clock source after division).

18.4.2 时钟分频比设置

满足外围模块时钟频率(PCLKB) $\geq 4 \times$ (除以计数时钟源频率后的频率) 。

19. Serial Array Unit (SAU)

19.1 Overview

A Serial Array Unit (SAU) has one unit. Unit 0 has four channels. Each channel can operate as 3-wire serial (simplified SPI), UART, or simplified IIC. However, these functions cannot be assigned to the same channel simultaneously.

Function assignment of each channel supported by the MCU is as shown in [Table 19.1](#).

Table 19.1 Function assignment

Unit	Channel	Used as simplified SPI	Used as UART	Used as simplified I ² C
0	0	SPI00	UART0 (supporting LIN-bus)	IIC00
	1	—		—
	2	—	UART1	—
	3	SPI11		IIC11

19.1.1 Simplified SPI

Data is transmitted or received in synchronization with the serial clock (SCK) output from the master.

3-wire serial communication is clocked communication performed by using three communication lines: one for the serial clock (SCK), one for transmitting serial data (SO), one for receiving serial data (SI).

For details about the settings, see [section 19.5. Operation of Simplified SPI](#).

[Data transmission and reception]

- Data length of 7 or 8 bits
- Phase control of transmit and receive data
- MSB- or LSB-first selectable

[Clock control]

- Master or slave selection
- Phase control of I/O clock
- Setting of transfer period by prescaler and internal counter of each channel
- Maximum transfer rate ^{*1}
During master communication: Max. PCLKB/2 (SPI00 only), Max. PCLKB/4
During slave communication: Max. f_{MCK}/6 ^{*2}

[Interrupt function]

- Transfer end interrupt or buffer empty interrupt (SAU0_SPI_TXRXI00/SAU0_SPI_TXRXI11)

[Error detection flag]

- Overrun error

In addition, simplified SPIs of following channels support the Snooze mode. In the Snooze mode, data can be received without CPU processing upon detecting SCK input in the Software Standby mode. The Snooze mode is only available in the following simplified SPIs, which support asynchronous reception.

- SPI00

Note 1. Set up the transfer rate within a range satisfying the SCK cycle time (tKCY). For details, see [section 27, Electrical Characteristics](#) with T_a = -40 to +125°C.

Note 2. f_{MCK} is a clock divided by PCLKB with a prescaler.

Note: Use a general-purpose port pin to send a chip select signal when required.

19. 串行阵列单元 (SAU)

19.1 概述

串行阵列单元 (SAU) 由一个单元组成。单元 0 有四个通道。每个通道可以作为三线串行（简化 SPI）、UART 或简化 IIC 运行。但是，这些功能不能同时分配给同一个通道。

MCU支持的每个通道的功能分配如表19.1所示。

表 19.1 函数分配

单元	渠道	用作简化的SPI	用作 UART	用作简化版I ² C
0	0	SPI00	UART0（支持 LIN） 公共汽车）	IIC00
	1	—		—
	2	—	UART1	—
	3	SPI11		IIC11

19.1.1简化的SPI

数据传输或接收与主控端的串行时钟 (SCK) 输出同步。

三线串行通信是使用三条通信线执行的时钟通信：一条用于串行时钟 (SCK)，一条用于发送串行数据 (SO)，一条用于接收串行数据 (SI)。

有关设置的详细信息，[请参阅第 19.5 节“简化 SPI 的操作”。](#)

[数据传输和接收]

- 数据长度为7或8位
- 发射和接收数据的相位控制
- MSB或LSB优先可选

[时钟控制]

- 主人或奴隶的选择
- I/O时钟的相位控制
- 通过预分频器和各通道的内部计数器设置传输周期
- 最高转账速率*1
主控通信期间：最大 PCLKB/2（仅限 SPI00），最大 PCLKB/4
从属通信期间：最大值f_{MCK}/6 ^{*2}

[中断功能]

- 传输结束中断或缓冲区空中断（SAU0_SPI_TXRXI00/SAU0_SPI_TXRXI11）

[错误检测标志]

- 超限错误

此外，以下通道的简化版 SPI 支持休眠模式。在休眠模式下，当软件待机模式下检测到 SCK 输入时，无需 CPU 处理即可接收数据。休眠模式仅适用于以下支持异步接收的简化版 SPI。

- SPI00

注 1. 将传输速率设置在满足 SCK 周期时间 (tKCY) 的范围内。有关详细信息，[请参阅第 27 节“电气特性（T_a = -40至+125°C）”。](#)

注 2. f_{MCK} 是由 PCLKB 分频并带预分频器的时钟。

注意：必要时，可以使用通用端口引脚发送片选信号。

19.1.2 UART

This is a start-stop synchronization communication function using two lines: serial data transmission (TXD) and serial data reception (RXD) lines. By using these two communication lines, each data frame, which consist of a start bit, data, parity bit, and stop bit, is transferred asynchronously (using the internal baud rate) between the microcontroller and the other communication party. Full-duplex UART communication can be performed by using a channel dedicated to transmission (even-numbered channel) and a channel dedicated to reception (odd-numbered channel). The LIN-bus can be implemented by using timer array unit with an external interrupt (IRQ0).

For details about the settings, see [section 19.6. Operation of UART Communication](#).

[Data Transmission and reception]

- Data length of 7, 8, or 9 bits*¹
- MSB or LSB first selectable
- Level setting of transmit and receive data and select of reverse
- Parity bit appending and parity check functions
- Stop bit appending

[Interrupt function]

- Transfer end interrupt and buffer empty interrupt (SAU0_UART_TXI0/SAU0_UART_RXI0/SAU0_UART_TXI1/SAU0_UART_RXI1)
- Error interrupt in case of framing error, parity error, or overrun error (SAU0_UART_ERRI0/SAU0_UART_ERRI1)

[Error detection flag]

- Framing error, parity error, or overrun error

In addition, UART reception of following channels supports the Snooze mode. In the Snooze mode, data can be received without CPU processing upon detecting RXD input in the Software Standby mode. The Snooze mode is only available in the following UARTs, which support the reception baud rate adjustment function.

- UART0

The LIN-bus is accepted in UART0 (channels 0 and 1 of unit 0).

[LIN-bus function]

- Wakeup signal detection
- Break field (BF) detection
- Sync field measurement, baud rate calculation

Note 1. Only UART0 supports the 9-bit data length.

19.1.3 Simplified I²C

This is a clocked communication function to communicate with two or more devices by using two lines: serial clock (SCL) and serial data (SDA). This simplified I²C is designed for single communication with a device such as EEPROM, flash memory, or A/D converter, and therefore, it functions only as a master.

Make sure by using software, as well as operating the control registers, that the AC specifications of the start and stop conditions are observed.

For details about the settings, see [section 19.8. Operation of Simplified I²C Communication](#).

[Data transmission and reception]

- Master transmission, master reception (only master function with a single master)
- ACK output function*¹ and ACK detection function
- Data length of 8 bits
(When an address is transmitted, the address is specified by the higher 7 bits, and the least significant bit is used for R/W control.)

19.1.2 UART

这是一个使用两条线路的启动-停止同步通信功能：串行数据传输 (TXD) 线和串行数据接收 (RXD) 线。通过这两条通信线路，每个数据帧（包含起始位、数据、奇偶校验位和停止位）在微控制器和通信另一方之间异步传输（使用内部波特率）。全双工 UART 通信可以通过使用专用于发送的通道（偶数通道）和专用于接收的通道（奇数通道）来实现。LIN 总线可以通过使用带有外部中断 (IRQ0) 的定时器阵列单元来实现。

[有关设置的详细信息，请参阅第 19.6 节“UART 通信的操作”。](#)

[数据传输和接收]

- 数据长度为7、8或9位*1
- 可优先选择最高有效位 (MSB) 或最低有效位 (LSB)。
- 设置发送和接收数据的电平以及反向选择
- 奇偶校验位附加和奇偶校验功能
- 停止位附加

[中断功能]

- 传输结束中断和缓冲区空中断（SAU0_UART_TXI0/SAU0_UART_RXI0/SAU0_UART_TXI1/SAU0_UART_RXI1）
- 帧错误、奇偶校验错误或溢出错误时的错误中断（SAU0_UART_ERRI0/SAU0_UART_ERRI1）

[错误检测标志]

- 帧错误、奇偶校验错误或溢出错误

此外，以下通道的 UART 接收支持休眠模式。在休眠模式下，当软件待机模式下检测到 RXD 输入时，无需 CPU 处理即可接收数据。休眠模式仅适用于支持接收波特率调节功能的以下 UART。

- UART0

LIN 总线在 UART0（单元 0 的通道 0 和 1）中被接受。

[LIN总线功能]

- 唤醒信号检测
- 断场（BF）检测
- 同步场测量，波特率计算

注1：只有UART0支持9位数据长度。

19.1.3 简化I²C

这是一个时钟通信功能，它使用两条线路：串行时钟 (SCL) 和串行数据 (SDA)，与两个或多个设备进行通信。这个简化的I²C专为与单个设备（例如 EEPROM、闪存或 A/D 转换器）通信而设计，因此它仅作为主设备运行。

通过使用软件以及操作控制寄存器，确保满足启动和停止条件的交流电规格。

[有关设置的详细信息，请参阅第 19.8 节。简化的I²C通信的操作。](#)

[数据传输和接收]

- 主控发送、主控接收（仅限单主控功能）
- ACK 输出功能*1 和 ACK 检测功能
- 数据长度为 8 位
(当传输地址时，地址由高 7 位指定，最低有效位用于读写控制。)

- Manual generation of start condition and stop condition

[Interrupt function]

- Transfer end interrupt (SAU0_IIC_TXRXI00/SAU0_IIC_TXRXI11)

[Error detection flag]

- ACK error, or overrun error

[Functions not supported by simplified I²C]

- Slave transmission, slave reception
- Arbitration loss detection function
- Clock stretch detection

Note 1. When receiving the last data, ACK is not output if 0 is written to the SOE bit (serial output enable register m (SOEm)) and serial communication data output is stopped. See (2) [Processing flow](#) for details.

Note: To use an I²C bus of full function, see [section 20, I²C Bus Interface \(IICA\)](#).

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3)

19.2 Configuration of Serial Array Unit

The serial array unit includes the registers, and input and output pins shown in [Table 19.2](#).

Table 19.2 Configuration of serial array

Item	Configuration
Shift register	8 or 9 bits*1
Buffer register	Lower 8 bits or 9 bits of serial data register mn (SDRmn)*1
Serial clock I/O	SCK00, SCK11 pins (for simplified SPI) SCL00, SCL11 pins (for simplified I ² C)
Serial data input	SI00, SI11 pins (for simplified SPI) RXD0 pins (for UART supporting LIN-bus), RXD1 pins (for UART)
Serial data output	SO00, SO11 pins (for simplified SPI) TXD0 pin (for UART supporting LIN-bus), TXD1 pins (for UART)
Serial data I/O	SDA00, SDA11 pins (for simplified I ² C)
Chip select input	SSI00 pin (for SPI00)
Control registers	<Registers of unit setting block> ● Serial clock select register m (SPSm) ● Serial channel enable status register m (SEm) ● Serial channel start register m (SSm) ● Serial channel stop register m (STm) ● Serial output enable register m (SOEm) ● Serial output register m (SOM) ● Serial output level register m (SOLm) ● Serial standby control register 0 (SSC0) ● Input switch control register (ISC) ● SAU Noise filter enable register (SNFEN) <Registers of each channel> ● Serial data register mn (SDRmn) ● Serial mode register mn (SMRmn) ● Serial communication operation setting register mn (SCRmn) ● Serial status register mn (SSRmn) ● Serial flag clear trigger register mn (SIRmn) ● UART loopback select register (ULBS)

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3)

Note 1. The number of bits used as the shift register and buffer register differs depending on the unit and channel.

- mn = 00, 01: lower 9 bits
- Other than above: lower 8 bits

- 手动生成启动条件和停止条件

[中断功能]

- 传输结束中断（SAU0_IIC_TXRXI00/SAU0_IIC_TXRXI11）

[错误检测标志]

- ACK错误或溢出错误

[简化版不支持的函数I²C]

- 从属传输，从属接收
- 仲裁损失检测功能
- 时钟拉伸检测

注1: 接收到最后一条数据时，如果将SOE位（串行输出使能寄存器m (SOEm)）写入0，则不会输出ACK，并且串行通信数据输出已停止。详情请参见(2) 处理流程。

注意：要使用具有完整功能的I²C总线，请参阅第20节，[I²C总线接口（IICA）](#)。

注: m: 单元编号(m = 0) n: 通道编号(n = 0 至3)

19.2 串行阵列单元的配置

串行阵列单元包括表 19.2 所示的寄存器、输入引脚和输出引脚。

表 19.2 串行阵列配置

物品	配置
值班登记表	8 或 9 位*1
缓冲寄存器	串行数据寄存器 mn 的低 8 位或 9 位 (SDRmn)*1
串行时钟 I/O	SCK00、SCK11 引脚（用于简化的 SPI） SCL00、SCL11 引脚（简化为I ² C）
串行数据输入	SI00、SI11 引脚（用于简化的 SPI） RXD0 引脚（用于支持 LIN 总线的 UART），RXD1 引脚（用于 UART）
串行数据输出	SO00、SO11 引脚（用于简化的 SPI） TXD0 引脚（用于支持 LIN 总线的 UART），TXD1 引脚（用于 UART）
串行数据 I/O	SDA00、SDA11 引脚（简化为I ² C）
芯片选择输入	SSI00 引脚（用于 SPI00）
控制寄存器	<单元设置块寄存器> ●串行时钟选择寄存器m（SPSm） ●串行通道使能状态寄存器m（SEm） ●串行通道起始寄存器m（SSm） ●串行通道停止寄存器m（STm） ●串行输出使能寄存器m（SOEm） ●串行输出寄存器m（SOM） ●串行输出电平寄存器m（SOLm） ●串行待机控制寄存器0（SSC0） ●输入开关控制寄存器（ISC） ●SAU噪声滤波器使能寄存器（SNFEN） <各通道寄存器> ●串行数据寄存器 mn (SDRmn) ●串行模式寄存器 mn (SMRmn) ●串行通信操作设置寄存器mn（SCRmn） ●串行状态寄存器 mn (SSRmn) ●串行标志清除触发寄存器 mn (SIRmn) ●UART 循环选择寄存器 (ULBS)

注: m: 单元编号(m = 0) n: 通道编号(n = 0 至3)

注1. 移位寄存器和缓冲寄存器所用的位数因单元和通道而异。

- mn = 00 ,01: 低9位
- 除上述以外: 低8位

Figure 19.1 shows the block diagram of serial array unit 0.

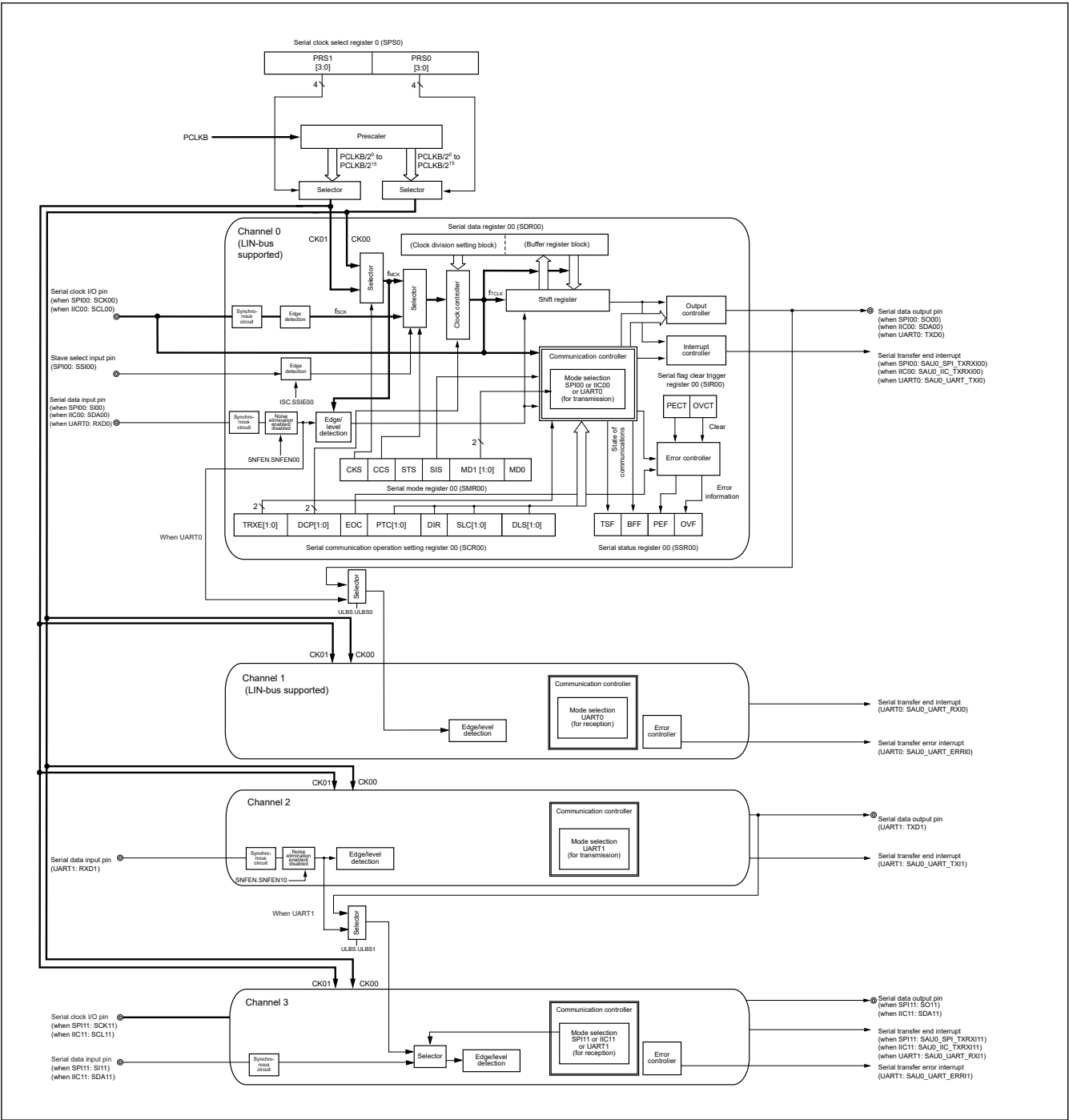


Figure 19.1 Block diagram of serial array unit 0

图 19.1 显示了串行阵列单元 0 的框图。

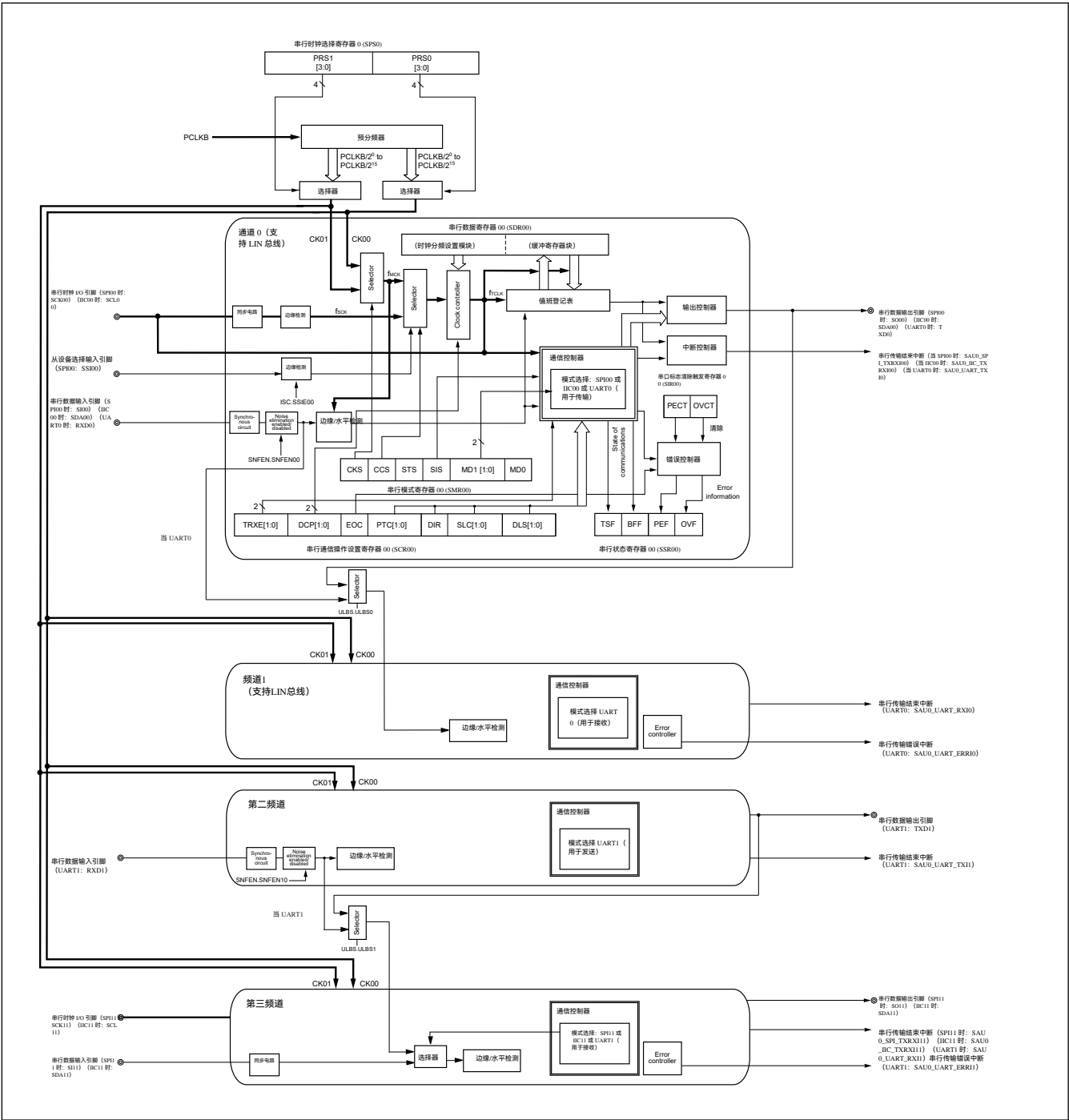


图 19.1 串行阵列单元 0 的框图

19.3 Register Descriptions

19.3.1 SPSm : Serial Clock Select Register m (m = 0)

Base address: SAUm = 0x400A_2000 + 0x200 × m (m = 0)

Offset address: 0x0126

Bit position:

1514131211109876543210

Bit field:

—	—	—	—	—	—	—	—	PRS1[3:0]				PRS0[3:0]			

Value after reset:

0000000000000000

Bit	Symbol	Function	R/W
3:0	PRS0[3:0]	Selection of Operation Clock (CKm0)*1 0x0: PCLKB 0x1: PCLKB/2 0x2: PCLKB/2 ² 0x3: PCLKB/2 ³ 0x4: PCLKB/2 ⁴ 0x5: PCLKB/2 ⁵ 0x6: PCLKB/2 ⁶ 0x7: PCLKB/2 ⁷ 0x8: PCLKB/2 ⁸ 0x9: PCLKB/2 ⁹ 0xA: PCLKB/2 ¹⁰ 0xB: PCLKB/2 ¹¹ 0xC: PCLKB/2 ¹² 0xD: PCLKB/2 ¹³ 0xE: PCLKB/2 ¹⁴ 0xF: PCLKB/2 ¹⁵	R/W
7:4	PRS1[3:0]	Selection of Operation Clock (CKm1)*1 0x0: PCLKB 0x1: PCLKB/2 0x2: PCLKB/2 ² 0x3: PCLKB/2 ³ 0x4: PCLKB/2 ⁴ 0x5: PCLKB/2 ⁵ 0x6: PCLKB/2 ⁶ 0x7: PCLKB/2 ⁷ 0x8: PCLKB/2 ⁸ 0x9: PCLKB/2 ⁹ 0xA: PCLKB/2 ¹⁰ 0xB: PCLKB/2 ¹¹ 0xC: PCLKB/2 ¹² 0xD: PCLKB/2 ¹³ 0xE: PCLKB/2 ¹⁴ 0xF: PCLKB/2 ¹⁵	R/W
15:8	—	These bits are read as 0. The write value should be 0.	R/W

Note 1. When changing the clock selected for PCLKB, do so after having stopped (serial channel stop register m (STm) = 0x000F) the operation of the serial array unit (SAU).

The SPSm is a 16-bit register that is used to select two types of operation clocks (CKm0, CKm1) that are commonly supplied to each channel. CKm1 is selected by the PRS1[3:0] bits, and CKm0 is selected by the PRS0[3:0] bits. Rewriting the SPSm register is prohibited when the register is in operation (when SEm.SE[n] = 1).

The input sources that can be selected with the PRS0[3:0] and PRS1[3:0] bits are shown in [Table 19.3](#).

19.3 寄存器描述

19.3.1 SPSm: 串行时钟选择寄存器 m(m = 0)

基地址: SAUm = 0x400A_2000 + 0x200 × m (m = 0)

偏移地址: 0x0126

位位置:

1514131211109876543210

位域:

—	—	—	—	—	—	—	—	PRS1[3:0]				PRS0[3:0]			

重置后的值:

0000000000000000

少量	象征	功能	R/W
3:0	PRS0[3:0]	选择操作时钟 (CKm0)*1 0x0: PCLKB 0x1: PCLKB/2 0x2: PCLKB/2 ² 0x3: PCLKB/2 ³ 0x4: PCLKB/2 ⁴ 0x5: PCLKB/2 ⁵ 0x6: PCLKB/2 ⁶ 0x7: PCLKB/2 ⁷ 0x8: PCLKB/2 ⁸ 0x9: PCLKB/2 ⁹ 0xA: PCLKB/2 ¹⁰ 0xB: PCLKB/2 ¹¹ 0xC: PCLKB/2 ¹² 0xD: PCLKB/2 ¹³ 0xE: PCLKB/2 ¹⁴ 0xF: PCLKB/2 ¹⁵	R/W
7:4	PRS1[3:0]	选择操作时钟 (CKm1)*1 0x0: PCLKB 0x1: PCLKB/2 0x2: PCLKB/2 ² 0x3: PCLKB/2 ³ 0x4: PCLKB/2 ⁴ 0x5: PCLKB/2 ⁵ 0x6: PCLKB/2 ⁶ 0x7: PCLKB/2 ⁷ 0x8: PCLKB/2 ⁸ 0x9: PCLKB/2 ⁹ 0xA: PCLKB/2 ¹⁰ 0xB: PCLKB/2 ¹¹ 0xC: PCLKB/2 ¹² 0xD: PCLKB/2 ¹³ 0xE: PCLKB/2 ¹⁴ 0xF: PCLKB/2 ¹⁵	R/W
15:8	—	这些位读取为 0。写入值应为 0。	R/W

注 1. 当更改为 PCLKB 选择的时钟时，请在停止串行阵列单元 (SAU) 的操作后进行（串行通道停止寄存器 m(STm) = 0x000F）。

SPSm是一个 16 位寄存器，用于选择两个常用的时钟类型(CKm0和CKm1，它们通常提供给每个通道。CKm1 由 PRS 1 的[3:0 和] 位选择，CKm0由 PRS0 的[3:0 和] 位选择。当寄存器处于工作状态时（即SEm.SE[n] = 1)处于工作状态时），禁止重写 SPSm 寄存器。

表 19.3 显示了可使用 PRS0[3:0] 和 PRS1[3:0] 位选择的输入源。

Table 19.3 Selection of operation clock (PRSk[3:0](k = 0, 1))

PRSk[3:0]		Selection of operation clock (CKmk) (k = 0, 1)				
		PCLKB = 2 MHz	PCLKB = 5 MHz	PCLKB = 10 MHz	PCLKB = 20 MHz	PCLKB = 32 MHz
0x0	PCLKB	2 MHz	5 MHz	10 MHz	20 MHz	32 MHz
0x1	PCLKB/2	1 MHz	2.5 MHz	5 MHz	10 MHz	16 MHz
0x2	PCLKB/2 ²	500 kHz	1.25 MHz	2.5 MHz	5 MHz	8 MHz
0x3	PCLKB/2 ³	250 kHz	625 kHz	1.25 MHz	2.5 MHz	4 MHz
0x4	PCLKB/2 ⁴	125 kHz	313 kHz	625 kHz	1.25 MHz	2 MHz
0x5	PCLKB/2 ⁵	62.5 kHz	156 kHz	313 kHz	625 kHz	1 MHz
0x6	PCLKB/2 ⁶	31.3 kHz	78.1 kHz	156 kHz	313 kHz	500 kHz
0x7	PCLKB/2 ⁷	15.6 kHz	39.1 kHz	78.1 kHz	156 kHz	250 kHz
0x8	PCLKB/2 ⁸	7.81 kHz	19.5 kHz	39.1 kHz	78.1 kHz	125 kHz
0x9	PCLKB/2 ⁹	3.91 kHz	9.77 kHz	19.5 kHz	39.1 kHz	62.5 kHz
0xA	PCLKB/2 ¹⁰	1.95 kHz	4.88 kHz	9.77 kHz	19.5 kHz	31.3 kHz
0xB	PCLKB/2 ¹¹	977 Hz	2.44 kHz	4.88 kHz	9.77 kHz	15.6 kHz
0xC	PCLKB/2 ¹²	488 Hz	1.22 kHz	2.44 kHz	4.88 kHz	7.81 kHz
0xD	PCLKB/2 ¹³	244 Hz	610 Hz	1.22 kHz	2.44 kHz	3.91 kHz
0xE	PCLKB/2 ¹⁴	122 Hz	305 Hz	610 Hz	1.22 kHz	1.95 kHz
0xF	PCLKB/2 ¹⁵	61.0 Hz	153 Hz	305 Hz	610 Hz	977 Hz

Note: When changing the clock selected for PCLKB, do so after having stopped (serial channel stop register m (STm) = 0x000F) the operation of the serial array unit (SAU).

19.3.2 SMRmn : Serial Mode Register mn (mn = 00, 02)

Base address: SAUm = 0x400A_2000 + 0x200 × m

Offset address: 0x0110 + 0x2 × n

Bit position: 1514131211109876543210

Bit field:

CKS	CCS	—	—	—	—	—	—	—	—	—	—	—	—	MD1[1:0]	MD0

Value after reset: 000000000001000000

Bit	Symbol	Function	R/W
0	MD0	Selection of Channel n Interrupt Source 0: Transfer end interrupt 1: Buffer empty interrupt (Occurs when data is transferred from the SDRmn register to the shift register.)	R/W
2:1	MD1[1:0]	Setting of Channel n Operation Mode 0 0: Simplified SPI mode 0 1: UART mode 1 0: Simplified I ² C mode 1 1: Setting prohibited	R/W
4:3	—	These bits are read as 0. The write value should be 0.	R/W
5	—	This bit is read as 1. The write value should be 1.	R/W
13:6	—	These bits are read as 0. The write value should be 0.	R/W
14	CCS	Selection of Transfer Clock (f _{TCLK}) of Channel n 0: Divided operation clock f _{MCK} specified by the CKS bit 1: Clock input f _{SCK} from the SCKp pin (slave transfer in simplified SPI mode)	R/W

表 19.3 操作时钟的选择(PRSk[3:0](k = 0, 1))

PRSk[3:0]		选择操作时钟 (CKmk)(k = 0, 1)				
		PCLKB = 2 MHz	PCLKB = 5兆赫	PCLKB = 10 MHz	PCLKB = 20 MHz	PCLKB = 32 MHz
0x0	PCLKB	2 MHz	5兆赫	10 MHz	20 MHz	32 MHz
0x1	PCLKB/2	1 MHz	2.5 MHz	5兆赫	10 MHz	16 MHz
0x2	PCLKB/2 ²	500 kHz	1.25 MHz	2.5 MHz	5兆赫	8 MHz
0x3	PCLKB/2 ³	250 kHz	625 kHz	1.25 MHz	2.5 MHz	4 MHz
0x4	PCLKB/2 ⁴	125 kHz	313 kHz	625 kHz	1.25 MHz	2 MHz
0x5	PCLKB/2 ⁵	62.5 kHz	156 kHz	313 kHz	625 kHz	1 MHz
0x6	PCLKB/2 ⁶	31.3 kHz	78.1 kHz	156 kHz	313 kHz	500 kHz
0x7	PCLKB/2 ⁷	15.6 kHz	39.1 kHz	78.1 kHz	156 kHz	250 kHz
0x8	PCLKB/2 ⁸	7.81 kHz	19.5 kHz	39.1 kHz	78.1 kHz	125 kHz
0x9	PCLKB/2 ⁹	3.91 kHz	9.77 kHz	19.5 kHz	39.1 kHz	62.5 kHz
0xA	PCLKB/2 ¹⁰	1.95 kHz	4.88 kHz	9.77 kHz	19.5 kHz	31.3 kHz
0xB	PCLKB/2 ¹¹	977 Hz	2.44 kHz	4.88 kHz	9.77 kHz	15.6 kHz
0xC	PCLKB/2 ¹²	488 Hz	1.22 kHz	2.44 kHz	4.88 kHz	7.81 kHz
0xD	PCLKB/2 ¹³	244 Hz	610 Hz	1.22 kHz	2.44 kHz	3.91 kHz
0xE	PCLKB/2 ¹⁴	122 Hz	305 Hz	610 Hz	1.22 kHz	1.95 kHz
0xF	PCLKB/2 ¹⁵	61.0 Hz	153 Hz	305 Hz	610 Hz	977 Hz

注意: 更改 PCLKB 选择的时钟时, 请在停止串行阵列单元 (SAU) 的操作 (串行通道停止寄存器m (STm) = 0x000F)) 之后进行。

19.3.2 SMRmn: 串行模式寄存器 mn(mn = 00, 02)

基地址: SAUm = 0x400A_2000 + 0x200 × m

偏移地址: 0x0110 + 0x2 × n

位位置: 1514131211109876543210

位域:

CKS	CCS	—	—	—	—	—	—	—	—	—	—	—	—	MD1[1:0]	MD0

重置后的值: 000000000001000000

少量	象征	功能	R/W
0	MD0	选择通道 n 中断源 0: 传输结束中断 1: 缓冲区空中断 (当数据从 SDRmn 寄存器传输到移位寄存器时发生。)	R/W
2:1	MD1[1:0]	设置通道 n 操作模式 0 0: 简化的 SPI 模式 0 1:UART 模式 1 0: 简化I ² C模式 1.1: 禁止设置	R/W
4:3	—	这些位读取为 0。写入值应为 0。	R/W
5	—	该位读取值为 1。写入值应为 1。	R/W
13:6	—	这些位读取为 0。写入值应为 0。	R/W
14	CCS	通道 n 的传输时钟 (f _{TCLK}) 选择 0: 由 CKS 位指定的分频操作时钟 f _{MCK} 1: 时钟输入f _{sck} 来自SCKp引脚 (简化 SPI 模式下的从机传输)	R/W

Bit	Symbol	Function	R/W
15	CKS	Selection of Operation Clock (f _{MCK}) of Channel n 0: Operation clock CKm0 set by the SPSm register 1: Operation clock CKm1 set by the SPSm register	R/W

The SMRmn register is used to set an operation mode of channel n. It is also used to select an operation clock (f_{MCK}), specify whether the serial clock (f_{SCK}) may be input or not, set a start trigger, the operating mode (as simplified SPI, UART or simplified I²C), and an interrupt source. This register is also used to invert the level of the receive data only in the UART mode.

Rewriting the SMRmn register is prohibited when the register is in operation (when SEm.SE[n] = 1). However, the MD0 bit can be rewritten during operation.

MD0 bit (Selection of Channel n Interrupt Source)

For continuous transmission, set this bit to 1 and write the next transmit data when SDRmn data has run out.

MD1[1:0] bits (Setting of Channel n Operation Mode)

The MD1[1:0] bits are used for setting of channel n operation mode.

CCS bit (Selection of Transfer Clock (f_{TCLK}) of Channel n)

Transfer clock (f_{TCLK}) is used for the shift register, communication controller, output controller, interrupt controller, and error controller. When CCS = 0, the division ratio of operation clock (f_{MCK}) is set by the higher 7 bits of the SDRmn register.

CKS bit (Selection of Operation Clock (f_{MCK}) of Channel n)

Operation clock (f_{MCK}) is used by the edge detector. In addition, depending on the setting of the CCS bit and the higher 7 bits of the SDRmn register, a transfer clock (f_{TCLK}) is generated.

19.3.3 SMRmn : Serial Mode Register mn (mn = 01, 03)

Base address:	SAUm = 0x400A_2000 + 0x200 × m															
Offset address:	0x0110 + 0x2 × n															
Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	CKS	CCS	—	—	—	—	—	STS	—	SIS0	—	—	—	MD1[1:0]		MD0
Value after reset:	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0

Bit	Symbol	Function	R/W
0	MD0	Selection of Channel n Interrupt Source 0: Transfer end interrupt 1: Buffer empty interrupt (Occurs when data is transferred from the SDRmn register to the shift register.)	R/W
2:1	MD1[1:0]	Setting of Channel n Operation Mode 0 0: Simplified SPI mode 0 1: UART mode 1 0: Simplified I ² C mode 1 1: Setting prohibited	R/W
4:3	—	These bits are read as 0. The write value should be 0.	R/W
5	—	This bit is read as 1. The write value should be 1.	R/W
6	SIS0	Controls Inversion of Level of Channel n Receive Data in UART Mode 0: Falling edge is detected as the start bit. The input communication data is captured as is. 1: Rising edge is detected as the start bit. The input communication data is inverted and captured.	R/W

少量	象征	功能	R/W
15	CKS	选择通道 n 的操作时钟(f _{MCK}) 0: 操作时钟 CKm0 由 SPSm 寄存器设置 1: 操作时钟 CKm1 由 SPSm 寄存器设置	R/W

SMRmn 寄存器用于设置通道 n 的工作模式。它还用于选择工作时钟(f_{MCK})，指定是否允许输入串行时钟(f_{SCK})，设置启动触发信号、工作模式（例如简化 SPI、UART 或简化I²C)）以及中断源。此外，该寄存器还用于在 UART 模式下反转接收数据的电平。

当寄存器处于运行状态时（当SEm.SE[n] = 1时），禁止重写 SMRmn 寄存器。但是，MD0 位可以在运行期间重写。

MD0 位（通道 n 中断源的选择）

对于连续传输，当 SDRmn 数据用完时，将此位设置为 1 并写入下一个传输数据。

MD1[1:0] 位（通道 n 操作模式的设置）

MD1[1:0] 位用于设置通道 n 的操作模式。

CCS 位（通道 n 的传输时钟 (f_{TCLK}) 选择）

传输时钟(f_{TCLK}) 用于移位寄存器、通信控制器、输出控制器、中断控制器和错误控制器。当CCS = 0时，操作时钟(f_{MCK}) 的分频比由SDRmn寄存器的高7位设置。

CKS 位（通道 n 的操作时钟 (f_{MCK}) 选择）

边沿检测器使用操作时钟(f_{MCK})。此外，根据CCS位和SDRmn寄存器高7位的设置，生成传输时钟(f_{TCLK})。

19.3.3SMRmn： 串行模式寄存器 mn(mn = 01, 03)

基地址: SAUm = 0x400A_2000 + 0x200 × m

偏移地址: 0x0110 + 0x2 × n

位位置:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域:	CKS	CCS	—	—	—	—	—	STS	—	SIS0	—	—	—	MD1[1:0]	MD0	
重置后的值:	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0

少量	象征	功能	R/W
0	MD0	选择通道 n 中断源 0: 传输结束中断 1: 缓冲区空中断（当数据从 SDRmn 寄存器传输到移位寄存器时发生。）	R/W
2:1	MD1[1:0]	设置通道 n 操作模式 0 0: 简化的 SPI 模式 0 1: UART模式 1 0: 简化I ² C模式 1.1: 禁止设置	R/W
4:3	—	这些位读取为 0。写入值应为 0。	R/W
5	—	该位读取值为 1。写入值应为 1。	R/W
6	SIS0	控制通道 n 的电平反转，在 UART 模式下接收数据: 0: 检测到下降沿作为起始位。 输入通信数据按原样捕获。 1: 检测到上升沿作为起始位。 输入通信数据被反转并捕获。	R/W

Bit	Symbol	Function	R/W
7	—	This bit is read as 0. The write value should be 0.	R/W
8	STS	Selection of Start Trigger Source 0: Only software trigger is valid (selected for simplified SPI, UART transmission, and simplified I ² C) 1: Valid edge of the RXDq pin (selected for UART reception)	R/W
13:9	—	These bits are read as 0. The write value should be 0.	R/W
14	CCS	Selection of Transfer Clock (f _{TCLK}) of Channel n 0: Divided operation clock f _{MCK} specified by the CKS bit 1: Clock input f _{SCK} from the SCKp pin (slave transfer in simplified SPI mode)	R/W
15	CKS	Selection of Operation Clock (f _{MCK}) of Channel n 0: Operation clock CKm0 set by the SPSm register 1: Operation clock CKm1 set by the SPSm register	R/W

The SMRmn register is used to set an operation mode of channel n. It is also used to select an operation clock (f_{MCK}), specify whether the serial clock (f_{SCK}) may be input or not, set a start trigger, the operating mode (as simplified SPI, UART, or simplified I²C), and an interrupt source. This register is also used to invert the level of the receive data only in the UART mode.

Rewriting the SMRmn register is prohibited when the register is in operation (when SEm.SE[n] = 1). However, the MD0 bit can be rewritten during operation.

MD0 bit (Selection of Channel n Interrupt Source)

For continuous transmission, set this bit to 1 and write the next transmit data when SDRmn data has run out.

MD1[1:0] bits (Setting of Channel n Operation Mode)

The MD1[1:0] bits are used for setting of channel n operation mode.

SIS0 bit (Controls Inversion of Level of Channel n Receive Data in UART Mode)

The SIS0 bit is used for control inversion of the level of channel n receive data in UART mode.

STS bit (Selection of Start Trigger Source)

Transfer is started when a valid edge on the software trigger or the RxDq pin is detected after 1 is set to the SSm register.

CCS bit (Selection of Transfer Clock (f_{TCLK}) of Channel n)

Transfer clock (f_{TCLK}) is used for the shift register, communication controller, output controller, interrupt controller, and error controller. When CCS = 0, the division ratio of operation clock (f_{MCK}) is set by the higher 7 bits of the SDRmn register.

CKS bit (Selection of Operation Clock (f_{MCK}) of Channel n)

Operation clock (f_{MCK}) is used by the edge detector. In addition, depending on the setting of the CCS bit and the higher 7 bits of the SDRmn register, a transfer clock (f_{TCLK}) is generated.

19.3.4 **SCRm0 : Serial Communication Operation Setting Register m0 (m = 0)**

Base address:	SAUm = 0x400A_2000 + 0x200 × m (m = 0)															
Offset address:	0x0118															
Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	TRXE[1:0]		DCP[1:0]		—	—	PTC[1:0]		DIR	—	SLC[1:0]		—	—	DLS[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	1	0	0	0	0	1	1	1

少量	象征	功能	R/W
7	—	该位读取值为 0。写入值应为 0。	R/W
8	STS	选择启动触发源 0: 仅软件触发有效（选择用于简化 SPI、UART 传输和简化I ² C） 1: RXDq引脚的有效边沿（选择用于UART接收）	R/W
13:9	—	这些位读取为 0。写入值应为 0。	R/W
14	CCS	通道 n 的传输时钟 (f _{TCLK}) 选择 0: 由 CKS 位指定的分频操作时钟f _{MCK} 1: 时钟输入 f _{SCK} 来自 SCKp 引脚（简化 SPI 模式下的从机传输）	R/W
15	CKS	选择通道 n 的操作时钟(f _{MCK}) 0: 操作时钟 CKm0 由 SPSm 寄存器设置 1: 操作时钟 CKm1 由 SPSm 寄存器设置	R/W

SMRmn 寄存器用于设置通道 n 的工作模式。它还用于选择工作时钟(f_{MCK})，指定是否允许输入串行时钟(f_{SCK})，设置启动触发信号、工作模式（简化 SPI、UART 或简化I²C）以及中断源。此外，该寄存器还用于在 UART 模式下反转接收数据的电平。

当寄存器处于运行状态时（当SEm.SE[n] = 1时），禁止重写 SMRmn 寄存器。但是，MD0 位可以在运行期间重写。

MD0 位（通道 n 中断源的选择）

对于连续传输，当 SDRmn 数据用完时，将此位设置为 1 并写入下一个传输数据。

MD1[1:0] 位（通道 n 操作模式的设置）

MD1[1:0] 位用于设置通道 n 的操作模式。

SIS0 位（控制 UART 模式下通道 n 接收数据的电平反转）

SIS0 位用于控制 UART 模式下通道 n 接收数据的电平反转。

STS 位（启动触发源选择）

当 SSm 寄存器设置为 1 后，检测到软件触发或 RxDq 引脚上的有效边沿时，传输开始。

CCS 位（通道 n 的传输时钟(f_{TCLK}) 的选择）

传输时钟(f_{TCLK}) 用于移位寄存器、通信控制器、输出控制器、中断控制器和错误控制器。当CCS = 0时，操作时钟(f_{MCK}) 的分频比由SDRmn寄存器的高7位设置。

CKS 位（通道 n 的操作时钟(f_{MCK}) 的选择）

边沿检测器使用操作时钟(f_{MCK})。此外，根据CCS位和SDRmn寄存器高7位的设置，生成传输时钟(f_{TCLK})。

19.3.4**SCRm0: 串行通信操作设置寄存器 m0(m = 0)**

Base address:	SAUm = 0x400A_2000 + 0x200 × m (m = 0)															
偏移地址:	0x0118															
位位置:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域:	TRXE[1:0]		DCP[1:0]		—	—	PTC[1:0]		DIR	—	SLC[1:0]		—	—	DLS[1:0]	
重置后的值:	0	0	0	0	0	0	0	0	1	0	0	0	0	1	1	1

Bit	Symbol	Function	R/W
1:0	DLS[1:0]	Setting of Data Length in Simplified SPI and UART Modes 0 0: Setting prohibited 0 1: 9-bit data length (stored in bits 0 to 8 of the SDRm0 register) (settable in UART mode only) 1 0: 7-bit data length (stored in bits 0 to 6 of the SDRm0 register) 1 1: 8-bit data length (stored in bits 0 to 7 of the SDRm0 register)	R/W
2	—	This bit is read as 1. The write value should be 1.	R/W
3	—	This bit is read as 0. The write value should be 0.	R/W
5:4	SLC[1:0]	Setting of Stop Bit in UART Mode 0 0: No stop bit 0 1: Stop bit length = 1 bit 1 0: Stop bit length = 2 bits 1 1: Setting prohibited	R/W
6	—	This bit is read as 0. The write value should be 0.	R/W
7	DIR	Selection of Data Transfer Sequence in Simplified SPI and UART Modes 0: Inputs or outputs data with MSB first 1: Inputs or outputs data with LSB first	R/W
9:8	PTC[1:0]	Setting of Parity Bit in UART Mode 0 0: Transmission: Does not output the parity bit Reception: Receives without parity 0 1: Transmission: Outputs 0 parity*1 Reception: No parity determination 1 0: Transmission: Outputs even parity Reception: Determines as even parity 1 1: Transmission: Outputs odd parity Reception: Determines as odd parity	R/W
11:10	—	These bits are read as 0. The write value should be 0.	R/W
13:12	DCP[1:0]	Selection of Data and Clock Phase in Simplified SPI Mode 0 0: Type1 (SCK: inverted, Input timing: rising edge) 0 1: Type2 (SCK: non-inverted, Input timing: falling edge) 1 0: Type3 (SCK: inverted, Input timing: falling edge) 1 1: Type4 (SCK: non-inverted, Input timing: rising edge)	R/W
15:14	TRXE[1:0]	Setting of Channel 0 Operation Mode 0 0: Disable communication 0 1: Reception only 1 0: Transmission only 1 1: Transmission and reception	R/W

Note 1. 0 is always added regardless of the data contents.

The SCRm0 is a communication operation setting register of channel 0. It is used to set a data transmission and reception mode, phase of data and clock, whether an error signal is to be masked or not, parity bit, start bit, stop bit, and data length.

Rewriting the SCRm0 register is prohibited when the register is in operation (when SE_m.SE[0] = 1).

DLS[1:0] bits (Setting of Data Length in Simplified SPI and UART Modes)

Be sure to set DLS[1:0] = 11b in the simplified I²C mode.

SLC[1:0] bits (Setting of Stop Bit in UART Mode)

When the transfer end interrupt is selected, the interrupt is generated when all stop bits have been completely transferred.

Set 1 bit (SLC[1:0]= 01b) during UART reception or in the simplified I²C mode. Set no stop bit (SLC[1:0] = 00b) in the simplified SPI mode.

Set 1 bit (SLC[1:0] = 01b) or 2 bits (SLC[1:0] = 10b) during UART transmission.

DIR bit (Selection of Data Transfer Sequence in Simplified SPI and UART Modes)

Be sure to clear DIR = 0 in the simplified I²C mode.

少量	象征	功能	R/W
1:0	DLS[1:0]	简化 SPI 和 UART 模式下数据长度的设置 0 0: 禁止设置 0 1: 9 位数据长度 (存储在 SDRm0 寄存器的第 0 位到第 8 位) (仅在 U ART 模式下可设置) 1 0: 7 位数据长度 (存储在 SDRm0 寄存器的第 0 位到第 6 位) 1 1: 8 位数据长度 (存储在 SDRm0 寄存器的第 0 位到第 7 位)	R/W
2	—	该位读取值为 1。写入值应为 1。	R/W
3	—	该位读取值为 0。写入值应为 0。	R/W
5:4	SLC[1:0]	UART模式下的停止位设置 0 0: 无停止位 0 1: 停止位长度= 1 位 1 0: 停止位长度= 2 位 1.1: 禁止设置	R/W
6	—	该位读取值为 0。写入值应为 0。	R/W
7	DIR	简化SPI和UART模式下数据传输顺序的选择 0: 输入或输出数据, 最高有效位在前。 1: 输入或输出数据以最低有效位 (LSB) 为先。	R/W
9:8	PTC[1:0]	UART模式下奇偶校验位的设置 0 0: 传输: 不输出奇偶校验位 接收: 接收无奇偶性 0 1:传输: 输出 0 个奇偶校验位*1 接收: 无平移判定 1 0: 传输: 输出偶校验 接收: 判定为均等 1.1: 传输: 输出奇校验 接收: 判定为奇偶校验	R/W
11:10	—	这些位读取为 0。写入值应为 0。	R/W
13:12	DCP[1:0]	简化SPI模式下的数据和时钟相位选择 0 0: 类型 1 (SCK: 反相, 输入时序: 上升沿) 0 1: 类型 2 (SCK: 非反相, 输入时序: 下降沿) 1 0: 类型 3 (SCK: 反相, 输入时序: 下降沿) 1 1: Type4 (SCK: 非反相, 输入时序: 上升沿)	R/W
15:14	TRXE[1:0]	设置通道 0 操作模式 0 0: 禁用通信 0 1: 仅限接待处 1 0: 仅限变速器 1.1: 发射和接收	R/W

注1. 无论数据内容如何, 始终会添加0。

SCRm0是通道 0 的通信操作设置寄存器。它用于设置数据传输和接收模式、数据和时钟的相位、是否屏蔽错误信号、奇偶校验位、起始位、停止位和数据长度。

当寄存器正在运行时 (当SE_m.SE[0] = 1处于工作状态时), 禁止重写SCRm0 寄存器。

DLS[1:0] 位 (简化 SPI 和 UART 模式下的数据长度设置)

务必将DLS[1:0] = 11b设置为简化的I²C模式。

SLC[1:0] 位 (UART 模式下的停止位设置)

选择传输结束中断时, 当所有停止位完全传输完毕后, 将产生中断。在 UART 接收期间或简化的I²C模式下, 设置 1 位(SLC[1:0]= 01b)。在简化的 SPI 模式下, 不设置停止位(SLC[1:0] = 00b)。

在 UART 传输期间设置 1 位(SLC[1:0] = 01b) 或 2 位(SLC[1:0] = 10b)。

DIR 位 (简化 SPI 和 UART 模式下的数据传输顺序选择)

务必在简化的I²C模式下清除DIR = 0。

PTC[1:0] bits (Setting of Parity Bit in UART Mode)

Be sure to set PTC[1:0] = 00b in the simplified SPI mode and simplified I²C mode

DCP[1:0] bits (Selection of Data and Clock Phase in Simplified SPI Mode)

See Figure 19.2.

Be sure to set DCP[1:0] = 00b in the UART mode and simplified I²C mode.

Figure 19.2 shows the data and clock phase in simplified SPI mode.

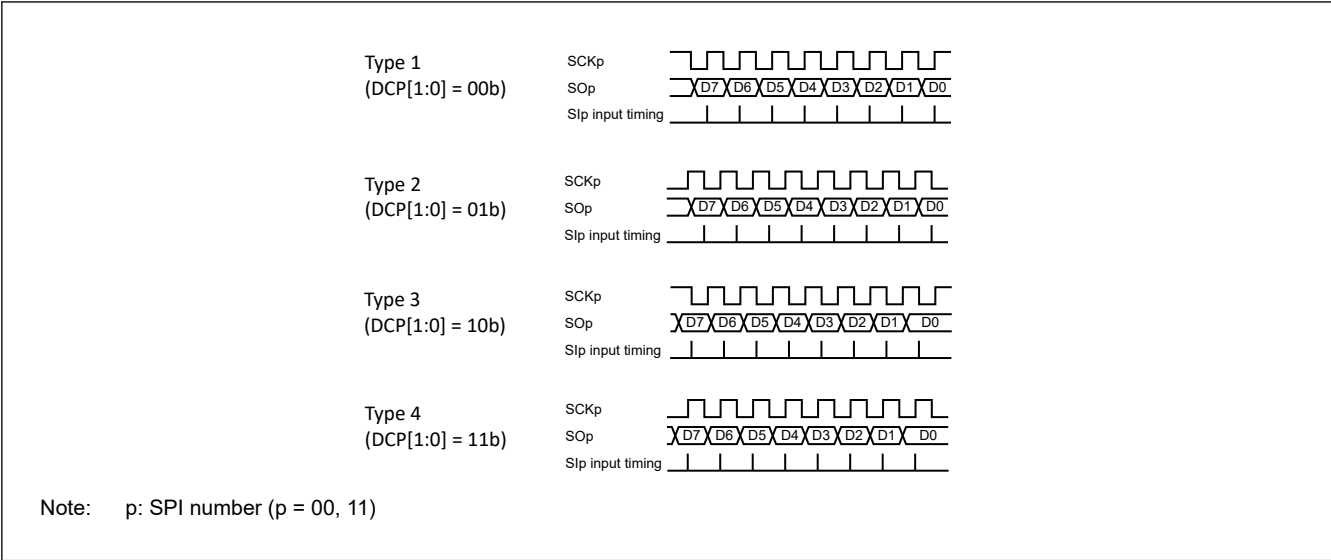


Figure 19.2 Data and clock phase in simplified SPI mode

TRXE[1:0] bits (Setting of Channel 0 Operation Mode)

The TRXE[1:0] bits are used for setting of channel 0 operation mode.

19.3.5 SCRM1 : Serial Communication Operation Setting Register m1 (m = 0)

Base address: SAUm = 0x400A_2000 + 0x200 × m (m = 0)													
Offset address: 0x011A													
Bit position: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0													
Bit field: TRXE[1:0] DCP[1:0] — EOC PTC[1:0] DIR — — SLC — — DLS[1:0]													
Value after reset: 0 0 0 0 0 0 0 0 1 0 0 0 0 1 1 1													

Bit	Symbol	Function	R/W
1:0	DLS[1:0]	Setting of Data Length in Simplified SPI and UART Modes 0 0: Setting prohibited 0 1: 9-bit data length (stored in the DAT[8:0] bits of the SDRm1 register) (settable in UART mode only) 1 0: 7-bit data length (stored in the DAT[6:0] bits of the SDRm1 register) 1 1: 8-bit data length (stored in the DAT[7:0] bits of the SDRm1 register)	R/W
2	—	This bit is read as 1. The write value should be 1.	R/W
3	—	This bit is read as 0. The write value should be 0.	R/W
4	SLC	Setting of Stop Bit in UART Mode 0: No stop bit 1: Stop bit length = 1 bit	R/W
6:5	—	These bits are read as 0. The write value should be 0.	R/W

PTC[1:0] 位 (UART 模式下奇偶校验位的设置)

请务必将PTC[1:0] = 00b设置为简化的 SPI 模式和简化的I²C模式

DCP[1:0] 位 (简化 SPI 模式下的数据和时钟相位选择) 参见图 19.2。

务必将DCP[1:0] = 00b设置为 UART 模式，并将I²C设置为简化模式。

图 19.2 显示了简化 SPI 模式下的数据和时钟相位。

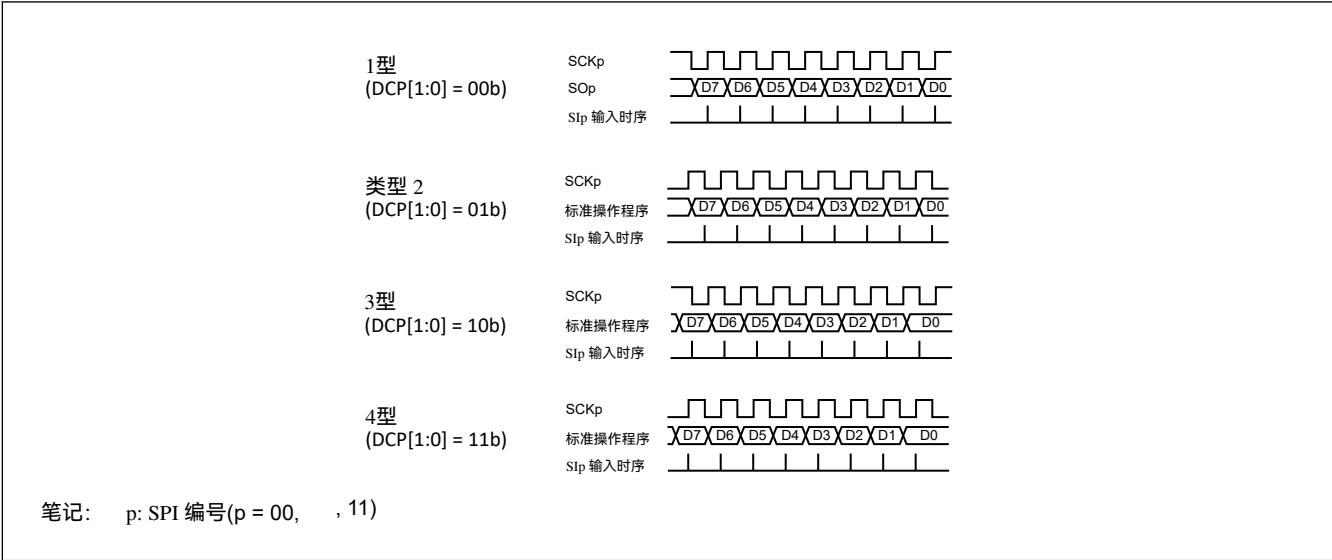


图 19.2 简化 SPI 模式下的数据和时钟相位

TRXE[1:0] 位 (设置通道 0 操作模式)

TRXE[1:0] 位用于设置通道 0 的操作模式。

19.3.5SCRM1: 串行通信操作设置寄存器 m1(m = 0)

基地址: SAUm = 0x400A_2000 + 0x200 × m (m = 0)													
偏移地址: 0x011A													
位位置: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0													
位域: TRXE[1:0] DCP[1:0] — EOC PTC[1:0] DIR — — SLC — — DLS[1:0]													
重置后的值: 0 0 0 0 0 0 0 0 1 0 0 0 0 1 1 1													

少量	象征	功能	R/W
1:0	DLS[1:0]	简化 SPI 和 UART 模式下数据长度的设置 0 0: 禁止设置 0 1: 9 位数据长度 (存储在 SDRm1 寄存器的 DAT[8:0] 位中) (仅在 UART 模式下可设置) 1 0: 7 位数据长度 (存储在 SDRm1 寄存器的 DAT[6:0] 位中) 1 1: 8 位数据长度 (存储在 SDRm1 寄存器的 DAT[7:0] 位中)	R/W
2	—	该位读取值为 1。写入值应为 1。	R/W
3	—	该位读取值为 0。写入值应为 0。	R/W
4	SLC	UART模式下的停止位设置 0: 无停止位 1: 停止位长度= 1 位	R/W
6:5	—	这些位读取为 0。写入值应为 0。	R/W

Bit	Symbol	Function	R/W
7	DIR	Selection of Data Transfer Sequence in Simplified SPI and UART Modes 0: Inputs or outputs data with MSB first 1: Inputs or outputs data with LSB first	R/W
9:8	PTC[1:0]	Setting of Parity Bit in UART Mode 0 0: Transmission: Does not output the parity bit Reception: Receives without parity 0 1: Transmission: Outputs 0 parity*1 Reception: No parity judgment 1 0: Transmission: Outputs even parity Reception: Determines as even parity 1 1: Transmission: Outputs odd parity Reception: Determines as odd parity	R/W
10	EOC	Mask Control of Error Interrupt Signal SAU0_UART_ERRI0 (m = 0) 0: Disables generation of error interrupt SAU0_UART_ERRI0 (m = 0) (SAUm_UART_RXIq is generated) 1: Enables generation of error interrupt SAU0_UART_ERRI0 (m = 0) (SAUm_UART_RXIq is not generated if an error occurs)	R/W
11	—	This bit is read as 0. The write value should be 0.	R/W
13:12	DCP[1:0]	Selection of Data and Clock Phase in Simplified SPI Mode 0 0: Type1 (SCK: inverted, Input timing: rising edge) 0 1: Type2 (SCK: non-inverted, Input timing: falling edge) 1 0: Type3 (SCK: inverted, Input timing: falling edge) 1 1: Type4 (SCK: non-inverted, Input timing: rising edge)	R/W
15:14	TRXE[1:0]	Setting of Channel 1 Operation Mode 0 0: Disable communication 0 1: Reception only 1 0: Transmission only 1 1: Transmission and reception	R/W

Note 1. 0 is always added regardless of the data contents.

The SCRM1 is a communication operation setting register of channel 1. It is used to set a data transmission and reception mode, phase of data and clock, whether an error signal is to be masked or not, parity bit, start bit, stop bit, and data length.

Rewriting the SCRM1 register is prohibited when the register is in operation (when SE_m.SE[1] = 1).

DLS[1:0] bits (Setting of Data Length in Simplified SPI and UART Modes)

Be sure to set DLS[1:0] = 11b in the simplified I²C mode.

SLC bit (Setting of Stop Bit in UART Mode)

When the transfer end interrupt is selected, the interrupt is generated when all stop bits have been completely transferred.

Set 1 bit (SLC = 1) during UART reception or in the simplified I²C mode. Set no stop bit (SLC = 0) in the simplified SPI mode.

Set 1 bit (SLC = 0) during UART transmission.

DIR bit (Selection of Data Transfer Sequence in Simplified SPI and UART Modes)

Be sure to clear DIR = 0 in the simplified I²C mode.

PTC[1:0] bits (Setting of Parity Bit in UART Mode)

Be sure to set PTC[1:0] = 00b in the simplified SPI mode and simplified I²C mode.

EOC bit (Mask Control of Error Interrupt Signal SAU0_UART_ERRI0 (m = 0))

Set EOC = 0 in the simplified SPI mode, simplified I²C mode, and during UART transmission.

DCP[1:0] bits (Selection of Data and Clock Phase in Simplified SPI Mode)

See Figure 19.2.

Be sure to set DCP[1:0] = 00b in the UART mode and simplified I²C mode.

少量	象征	功能	R/W
7	DIR	简化SPI和UART模式下数据传输顺序的选择 0: 输入或输出数据，最高有效位在前。 1: 输入或输出数据以最低有效位 (LSB) 为先。	R/W
9:8	PTC[1:0]	UART模式下奇偶校验位的设置 0 0: 传输: 不输出奇偶校验位 接收: 接收无奇偶性 0 1:传输:输出 0 个奇偶校验位*1 评价: 无平等判断 1 0: 传输: 输出偶校验 接收: 判定为均等 1.1: 传输: 输出奇校验 接收: 判定为奇偶校验	R/W
10	EOC	错误中断信号 SAU0_UART_ERRI0 的掩码控制(m = 0) 0: 禁用错误中断 SAU0_UART_ERRI0 的生成(m = 0) (已生成 SAUm_UART_RXIq) 1: 启用错误中断 SAU0_UART_ERRI0 的生成(m = 0) (如果发生错误, 则不会生成 SAUm_UART_RXIq)	R/W
11	—	该位读取值为 0。写入值应为 0。	R/W
13:12	DCP[1:0]	简化SPI模式下的数据和时钟相位选择 0 0: 类型 1 (SCK: 反相, 输入时序: 上升沿) 0 1: Type2 (SCK: 非反相, 输入时序: 下降沿) 1 0: 类型 3 (SCK: 反相, 输入时序: 下降沿) 1 1: Type4 (SCK: 非反相, 输入时序: 上升沿)	R/W
15:14	TRXE[1:0]	设置通道 1 操作模式 0 0: 禁用通信 0 1: 仅限接待处 1 0: 仅传输 1.1: 发射和接收	R/W

注1. 无论数据内容如何，始终会添加0。

SCRM1 是通道 1 的通信操作设置寄存器。它用于设置数据传输和接收模式、数据和时钟的相位、是否屏蔽错误信号、奇偶校验位、起始位、停止位和数据长度。

当寄存器正在运行时（当SE_m.SE[1] = 1时），禁止重写 SCRM1 寄存器。

DLS[1:0] 位（简化 SPI 和 UART 模式下的数据长度设置）

务必将DLS[1:0] = 11b设置为简化的I²C模式。

SLC 位（UART 模式下的停止位设置）

选择传输结束中断时，当所有停止位完全传输完毕后，将产生中断。在 UART 接收期间或简化的I²C模式下，设置 1 位(SLC = 1)。在简化的 SPI 模式下，不设置停止位(SLC = 0)。

在 UART 传输期间设置 1 位(SLC = 0)。

DIR 位（简化 SPI 和 UART 模式下的数据传输顺序选择）

务必在简化的I²C模式下清除DIR = 0。

PTC[1:0] 位（UART 模式下奇偶校验位的设置）

务必将PTC[1:0] = 00b设置为简化的 SPI 模式和简化的I²C模式。

EOC 位（错误中断信号 SAU0_UART_ERRI0(m = 0)的掩码控制）

在简化的 SPI 模式、简化的EOC = 0 模式和 UART 传输期间设置I²C。

DCP[1:0] 位（简化 SPI 模式下的数据和时钟相位选择）参见图 19.2。

务必将DCP[1:0] = 00b设置为 UART 模式，并将I²C设置为简化模式。

TRXE[1:0] bits (Setting of Channel 1 Operation Mode)

The TRXE[1:0] bits are used for setting of channel 1 operation mode.

19.3.6 SCR02 : Serial Communication Operation Setting Register 02

Base address: SAU0 = 0x400A_2000

Offset address: 0x011C

Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	TRXE[1:0]		DCP[1:0]		—	—	PTC[1:0]		DIR	—	SLC[1:0]		—	—	—	DLS
Value after reset:	0	0	0	0	0	0	0	0	1	0	0	0	0	1	1	1

Bit	Symbol	Function	R/W
0	DLS	Setting of Data Length in Simplified SPI and UART Modes 0: 7-bit data length (stored in the DAT[6:0] bits of the SDR02 register) 1: 8-bit data length (stored in the DAT[7:0] bits of the SDR02 register)	R/W
2:1	—	These bits are read as 1. The write value should be 1.	R/W
3	—	This bit is read as 0. The write value should be 0.	R/W
5:4	SLC[1:0]	Setting of Stop Bit in UART Mode 0 0: No stop bit 0 1: Stop bit length = 1 bit 1 0: Stop bit length = 2 bits 1 1: Setting prohibited	R/W
6	—	This bit is read as 0. The write value should be 0.	R/W
7	DIR	Selection of Data Transfer Sequence in Simplified SPI and UART Modes 0: Inputs or outputs data with MSB first 1: Inputs or outputs data with LSB first	R/W
9:8	PTC[1:0]	Setting of Parity Bit in UART Mode 0 0: Transmission: Does not output the parity bit Reception: Receives without parity 0 1: Transmission: Outputs 0 parity*1 Reception: No parity judgment 1 0: Transmission: Outputs even parity Reception: Determines as even parity 1 1: Transmission: Outputs odd parity Reception: Determines as odd parity	R/W
11:10	—	These bits are read as 0. The write value should be 0.	R/W
13:12	DCP[1:0]	Selection of Data and Clock Phase in Simplified SPI Mode 0 0: Type1 (SCK: inverted, Input timing: rising edge) 0 1: Type2 (SCK: non-inverted, Input timing: falling edge) 1 0: Type3 (SCK: inverted, Input timing: falling edge) 1 1: Type4 (SCK: non-inverted, Input timing: rising edge)	R/W
15:14	TRXE[1:0]	Setting of Channel 2 Operation Mode 0 0: Disables communication 0 1: Reception only 1 0: Transmission only 1 1: Transmission and reception	R/W

Note 1. 0 is always added regardless of the data contents.

The SCR02 is a communication operation setting register of channel 2. It is used to set a data transmission and reception mode, phase of data and clock, whether an error signal is to be masked or not, parity bit, start bit, stop bit, and data length.

Rewriting the SCR02 register is prohibited when the register is in operation (when SE0.SE[2] = 1).

DLS bit (Setting of Data Length in Simplified SPI and UART Modes)

Be sure to set DLS = 1 in the simplified I²C mode.

TRXE[1:0] 位（通道 1 操作模式的设置）

TRXE[1:0] 位用于设置通道 1 的操作模式。

19.3.6SCR02： 串行通信操作设置寄存器02

基地址: SAU0 = 0x400A_2000

偏移地址: 0x011C

位位置:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域:	TRXE[1:0]		DCP[1:0]		—	—	PTC[1:0]		DIR	—	SLC[1:0]		—	—	—	DLS
重置后的值:	0	0	0	0	0	0	0	0	1	0	0	0	0	1	1	1

少量	象征	功能	R/W
0	DLS	简化 SPI 和 UART 模式下的数据长度设置：0： 7 位数据长度（存储在 SDR02 寄存器的 DAT[6:0] 位中） 1： 8 位数据长度（存储在 SDR02 寄存器的 DAT[7:0] 位中）	R/W
2:1	—	这些位读取为 1。写入值应为 1。	R/W
3	—	该位读取值为 0。写入值应为 0。	R/W
5:4	SLC[1:0]	UART模式下的停止位设置 0 0： 无停止位 0 1:停止位长度= 1 位 1 0： 停止位长度= 2 位 1.1： 禁止设置	R/W
6	—	该位读取值为 0。写入值应为 0。	R/W
7	DIR	简化SPI和UART模式下数据传输顺序的选择 0： 输入或输出数据，最高有效位在前。 1： 输入或输出数据以最低有效位 (LSB) 为先。	R/W
9:8	PTC[1:0]	UART模式下奇偶校验位的设置 0 0： 传输： 不输出奇偶校验位 接收： 接收无奇偶性 0 1:传输: 输出 0 个奇偶校验位*1 评价： 无平等判断 1 0： 传输： 输出偶校验 接收： 判定为均等 1.1： 传输： 输出奇校验 接收： 判定为奇偶校验	R/W
11:10	—	这些位读取为 0。写入值应为 0。	R/W
13:12	DCP[1:0]	简化SPI模式下的数据和时钟相位选择 0 0： 类型 1（SCK： 反相，输入时序： 上升沿） 0 1： 类型 2（SCK： 非反相，输入时序： 下降沿） 1 0： 类型 3（SCK： 反相，输入时序： 下降沿） 1 1： Type4（SCK： 非反相，输入时序： 上升沿）	R/W
15:14	TRXE[1:0]	设置通道 2 操作模式 0 0： 禁用通信 0 1： 仅限接待处 1 0： 仅限变速器 1.1： 发射和接收	R/W

注1. 无论数据内容如何，始终会添加0。

SCR02 是通道 2 的通信操作设置寄存器。它用于设置数据传输和接收模式、数据和时钟的相位、是否屏蔽错误信号、奇偶校验位、起始位、停止位和数据长度。

当寄存器正在运行时（当SE0.SE[2] = 1时），禁止重写SCR02寄存器。

DLS 位（简化 SPI 和 UART 模式下的数据长度设置）

务必将DLS = 1 设置为简化的I²C模式。

SLC[1:0] bits (Setting of Stop Bit in UART Mode)

When the transfer end interrupt is selected, the interrupt is generated when all stop bits have been completely transferred.

Set 1 bit (SLC[1:0] = 01b) during UART reception or in the simplified I²C mode. Set no stop bit (SLC[1:0] = 00b) in the simplified SPI mode.

Set 1 bit (SLC[1:0] = 01b) or 2 bits (SLC[1:0] = 10b) during UART transmission.

DIR bit (Selection of Data Transfer Sequence in Simplified SPI and UART Modes)

Be sure to clear DIR = 0 in the simplified I²C mode.

PTC[1:0] bits (Setting of Parity Bit in UART Mode)

Be sure to set PTC[1:0] = 00b in the simplified SPI mode and simplified I²C mode.

DCP[1:0] bits (Selection of Data and Clock Phase in Simplified SPI Mode)

See [Figure 19.2](#).

Be sure to set DCP[1:0] = 00b in the UART mode and simplified I²C mode.

TRXE[1:0] bits (Setting of Channel 2 Operation Mode)

The TRXE[1:0] bits are used for setting of channel 2 operation mode.

19.3.7 SCR03 : Serial Communication Operation Setting Register 03

Base address: SAU0 = 0x400A_2000															
Offset address: 0x011E															
Bit position: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0															
Bit field: TRXE[1:0] DCP[1:0] — EOC PTC[1:0] DIR — — SLC — — — DLS															
Value after reset: 0 0 0 0 0 0 0 0 1 0 0 0 0 1 1 1															

Bit	Symbol	Function	R/W
0	DLS	Setting of Data Length in Simplified SPI and UART Modes 0: 7-bit data length (stored in the DAT[6:0] bits of the SDR03 register) 1: 8-bit data length (stored in the DAT[7:0] bits of the SDR03 register)	R/W
2:1	—	These bits are read as 1. The write value should be 1.	R/W
3	—	This bit is read as 0. The write value should be 0.	R/W
4	SLC	Setting of Stop Bit in UART Mode 0: No stop bit 1: Stop bit length = 1 bit	R/W
6:5	—	These bits are read as 0. The write value should be 0.	R/W
7	DIR	Selection of Data Transfer Sequence in Simplified SPI and UART Modes 0: Inputs or outputs data with MSB first 1: Inputs or outputs data with LSB first	R/W
9:8	PTC[1:0]	Setting of Parity Bit in UART Mode 0 0: Transmission: Does not output the parity bit Reception: Receives without parity 0 1: Transmission: Outputs 0 parity*1 Reception: No parity determination 1 0: Transmission: Outputs even parity Reception: Determines as even parity 1 1: Transmission: Outputs odd parity Reception: Determines as odd parity	R/W

SLC[1:0] 位 (UART 模式下的停止位设置)

选择传输结束中断时，当所有停止位完全传输完毕后，就会产生中断。

在 UART 接收期间或在简化的I²C模式下，设置 1 位(SLC[1:0] = 01b)。在简化的 SPI 模式下，不设置停止位(SLC[1:0] = 00b)。

在 UART 传输期间设置 1 位(SLC[1:0] = 01b) 或 2 位(SLC[1:0] = 10b)。

DIR 位 (简化 SPI 和 UART 模式下的数据传输顺序选择)

务必在简化的I²C模式下清除DIR = 0。

PTC[1:0] 位 (UART 模式下奇偶校验位的设置)

务必将PTC[1:0] = 00b设置为简化 SPI 模式和简化I²C模式。

DCP[1:0] 位 (简化 SPI 模式下的数据和时钟相位选择)

参见图 19.2。

务必将DCP[1:0] = 00b设置为 UART 模式，并将I²C设置为简化模式。

TRXE[1:0] 位 (通道 2 操作模式的设置)

TRXE[1:0] 位用于设置通道 2 的操作模式。

19.3.7SCR03： 串行通信操作设置寄存器03

基地址: SAU0 = 0x400A_2000															
偏移地址: 0x011E															
位位置: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0															
位域: TRXE[1:0] DCP[1:0] — EOC PTC[1:0] DIR — — SLC — — — DLS															
重置后的值: 0 0 0 0 0 0 0 0 1 0 0 0 0 1 1 1															

少量	象征	功能	R/W
0	DLS	简化 SPI 和 UART 模式下的数据长度设置: 0: 7 位数据长度 (存储在 SDR03 寄存器的 DAT[6:0] 位中) 1: 8 位数据长度 (存储在 SDR03 寄存器的 DAT[7:0] 位中)	R/W
2:1	—	这些位读取为 1。写入值应为 1。	R/W
3	—	该位读取值为 0。写入值应为 0。	R/W
4	SLC	UART模式下的停止位设置 0: 无停止位 1: 停止位长度= 1 位	R/W
6:5	—	这些位读取为 0。写入值应为 0。	R/W
7	DIR	简化SPI和UART模式下数据传输顺序的选择 0: 输入或输出数据，最高有效位在前。 1: 输入或输出数据以最低有效位 (LSB) 为先。	R/W
9:8	PTC[1:0]	UART模式下奇偶校验位的设置 0 0: 传输: 不输出奇偶校验位 接收: 接收无奇偶性 0 1:传输: 输出 0 个奇偶校验位*1 接收: 无平移判定 1 0: 传输: 输出偶校验 接收: 判定为均等 1.1: 传输: 输出奇校验 接收: 判定为奇偶校验	R/W

Bit	Symbol	Function	R/W
10	EOC	Mask Control of Error Interrupt Signal SAU0_UART_ERRI1 0: Disables generation of error interrupt SAU0_UART_ERRI1 (SAU0_UART_RXI1 is generated) 1: Enables generation of error interrupt SAU0_UART_ERRI1 (SAU0_UART_RXI1 is not generated if an error occurs)	R/W
11	—	This bit is read as 0. The write value should be 0.	R/W
13:12	DCP[1:0]	Selection of Data and Clock Phase in Simplified SPI Mode 0 0: Type1 (SCK: inverted, Input timing: rising edge) 0 1: Type2 (SCK: non-inverted, Input timing: falling edge) 1 0: Type3 (SCK: inverted, Input timing: falling edge) 1 1: Type4 (SCK: non-inverted, Input timing: rising edge)	R/W
15:14	TRXE[1:0]	Setting of Operation Mode of Channel 3 0 0: Disable communication 0 1: Reception only 1 0: Transmission only 1 1: Transmission and reception	R/W

Note 1. 0 is always added regardless of the data contents.

The SCR03 is a communication operation setting register of channel 3. It is used to set a data transmission and reception mode, phase of data and clock, whether an error signal is to be masked or not, parity bit, start bit, stop bit, and data length.

Rewriting the SCR03 register is prohibited when the register is in operation (when SE0.SE[3] = 1).

DLS bit (Setting of Data Length in Simplified SPI and UART Modes)

Be sure to set DLS = 1 in the simplified I²C mode.

SLC bit (Setting of Stop Bit in UART Mode)

When the transfer end interrupt is selected, the interrupt is generated when all stop bits have been completely transferred.

Set 1 bit (SLC = 1) during UART reception or in the simplified I²C mode. Set no stop bit (SLC = 0) in the simplified SPI mode.

Set 1 bit (SLC = 0) during UART transmission.

DIR bit (Selection of Data Transfer Sequence in Simplified SPI and UART Modes)

Be sure to clear DIR = 0 in the simplified I²C mode.

PTC[1:0] bits (Setting of Parity Bit in UART Mode)

Be sure to set PTC[1:0] = 00b in the simplified SPI mode and simplified I²C mode.

EOC bit (Mask Control of Error Interrupt Signal SAU0_UART_ERRI1)

Set EOC = 0 in the simplified SPI mode, simplified I²C mode, and during UART transmission.*1

DCP[1:0] bits (Selection of Data and Clock Phase in Simplified SPI Mode)

See [Figure 19.2](#).

Be sure to set DCP[1:0] = 00b in the UART mode and simplified I²C mode.

TRXE[1:0] bits (Setting of Operation Mode of Channel 3)

The TRXE[1:0] bits are used for setting of channel 3 operation mode.

Note 1. When using SPI11 not with EOC = 0, error interrupt SAU0_UART_ERRI1 may be generated.

少量	象征	功能	R/W
10	EOC	错误中断信号 SAU0_UART_ERRI1 的掩码控制 0: 禁用错误中断 SAU0_UART_ERRI1 的生成（生成 SAU0_UART_RXI1） 1: 启用错误中断 SAU0_UART_ERRI1 的生成（如果发生错误，则不会生成 SAU0_UART_RXI1）	R/W
11	—	该位读取值为 0。写入值应为 0。	R/W
13:12	DCP[1:0]	简化SPI模式下的数据和时钟相位选择 0 0: 类型 1（SCK：反相，输入时序：上升沿） 0 1: 类型 2（SCK：非反相，输入时序：下降沿） 1 0: 类型 3（SCK：反相，输入时序：下降沿） 1 1: Type4（SCK：非反相，输入时序：上升沿）	R/W
15:14	TRXE[1:0]	通道 3 操作模式设置 0 0: 禁用通信 0 1: 仅限接待处 1 0: 仅限变速器 1.1: 发射和接收	R/W

注1. 无论数据内容如何，始终会添加0。

SCR03 是通道 3 的通信操作设置寄存器。它用于设置数据传输和接收模式、数据和时钟的相位、是否屏蔽错误信号、奇偶校验位、起始位、停止位和数据长度。

当寄存器正在运行时（当SE0.SE[3] = 1时），禁止重写SCR03寄存器。

DLS 位（简化 SPI 和 UART 模式下的数据长度设置）

务必将DLS = 1 设置为简化的I²C模式。

SLC 位（UART 模式下的停止位设置）

选择传输结束中断时，当所有停止位完全传输完毕后，就会产生中断。

在 UART 接收期间或在简化的I²C模式下，设置 1 位(SLC = 1)。在简化的 SPI 模式下，不设置停止位(SLC = 0)。

在 UART 传输期间设置 1 位(SLC = 0)。

DIR 位（简化 SPI 和 UART 模式下的数据传输顺序选择）

务必在简化的I²C模式下清除DIR = 0。

PTC[1:0] 位（UART 模式下奇偶校验位的设置）

务必将PTC[1:0] = 00b设置为简化 SPI 模式和简化I²C模式。

EOC 位（错误中断信号 SAU0_UART_ERRI1 的掩码控制）

在简化的 SPI 模式、简化的I²C模式以及 UART 传输期间设置EOC = 0。*1

DCP[1:0] 位（简化 SPI 模式下的数据和时钟相位选择）

参见图 19.2。

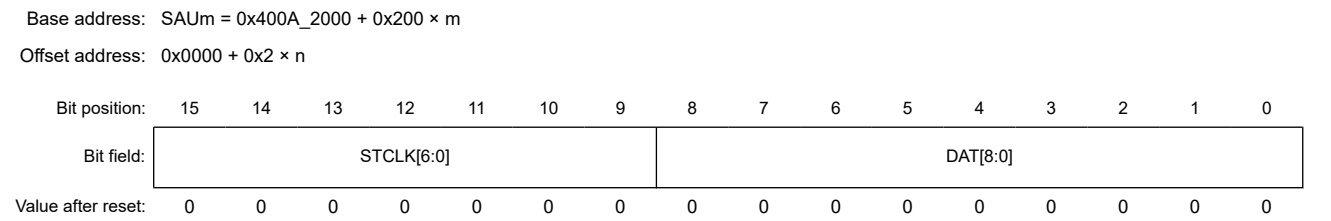
务必将DCP[1:0] = 00b设置为 UART 模式，并将I²C设置为简化模式。

TRXE[1:0] 位（通道 3 的操作模式设置）

TRXE[1:0] 位用于设置通道 3 的操作模式。

注1. 当使用SPI11时，如果未使用EOC = 0, 错误中断，可能会生成SAU0_UART_ERRI1。

19.3.8 SDRmn : Serial Data Register mn (mn = 00, 01, 02, 03)



Bit	Symbol	Function	R/W
8:0	DAT[8:0]	Data Buffer for Transmit and Receive	R/W ^{*1}
15:9	STCLK[6:0]	Transfer Clock Setting by Dividing the Operation Clock 0x00: f _{MCK} /2 (Setting prohibited for UART and simplified I ² C) 0x01: f _{MCK} /4 (Setting prohibited for UART) 0x02: f _{MCK} /6 0x03: f _{MCK} /8 ⋮ 0x7C: f _{MCK} /250 0x7D: f _{MCK} /252 0x7E: f _{MCK} /254 0x7F: f _{MCK} /256	R/W

Note 1. The SDR02.DAT[8] and SDR03.DAT[8] bits are reserved bits. These bits are read as 0. The write value should be 0.

The SDRmn is the transmit and receive data register (16 bits) of unit m and channel n.

The DAT[8:0] bits of SDR00, SDR01 or the DAT[7:0] bits of SDR02^{*1} and SDR03^{*1}, function as a transmit and receive buffer register, and the STCLK[6:0] bits are used as a register that sets the division ratio of the operation clock (f_{MCK}).

If the CCS bit of serial mode register mn (SMRmn) is cleared to 0, the clock set by dividing the operation clock by the STCLK[6:0] bits is used as the transfer clock.

If the CCS bit of serial mode register mn (SMRmn) is set to 1, set the STCLK[6:0] bits of SDR00, SDR01 to 0000000b. The input clock f_{SCK} (slave transfer in simplified SPI mode) from the SCKp pin is used as the transfer clock.

The DAT[7:0] or DAT[8:0] bits function as a transmit and receive buffer register. During reception, the parallel data converted by the shift register is stored in the DAT[7:0] or DAT[8:0] bits, and during transmission, the data to be transmitted to the shift register is set to the DAT[7:0] or DAT[8:0] bits.

The SDRmn register can be read or written in 16-bit access.

However, the STCLK[6:0] bits can only be written or read when the operation is stopped (SEm.SE[n] = 0). During operation (SEm.SE[n] = 1), a value is written only to the DAT[7:0] or DAT[8:0] bits. When the SDRmn register is read during operation, the STCLK[6:0] bits are always read as 0.

The data stored in the DAT[7:0] or DAT[8:0] bits is as follows, depending on the setting of the DLS[1:0] bits of serial communication operation setting register mn (SCRmn), regardless of the output sequence of the data.

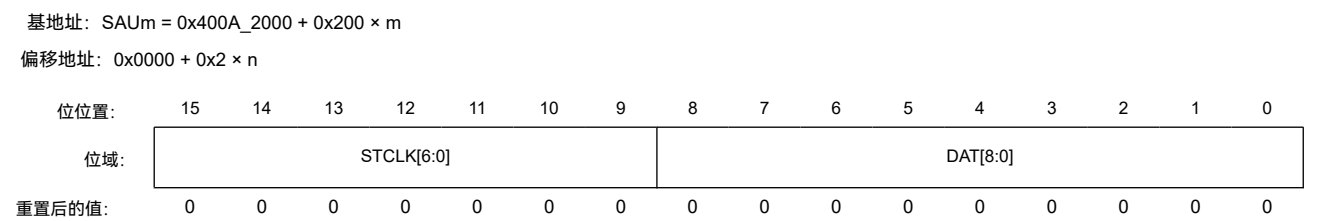
- 7-bit data length (stored in the DAT[6:0] bits)
- 8-bit data length (stored in the DAT[7:0] bits)
- 9-bit data length (stored in the DAT[8:0] bits)^{*1}

Note 1. Only the following UARTs support the 9-bit data length.

- UART0

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3)

19.3.8 SDRmn: 串行数据寄存器 mn(mn = 00, 01, 02, 03)



少量	象征	功能	R/W
8:0	DAT[8:0]	用于发送和接收的数据缓冲区	R/W ^{*1}
15:9	STCLK[6:0]	通过分频操作时钟设置传输时钟0x00: f _{MCK} /2 (UART 禁止设置, 简化为I ² C) 0x01: f _{MCK} /4 (UART 禁止设置) 0x02: f _{MCK} /60x0 3: f _{MCK} /8 ⋮ 0x7C: f _{MCK} /250 0 x7D: f _{MCK} /252 0 x7E: f _{MCK} /254 0 x7F: f _{MCK} /256	R/W

注1: SDR02.DAT[8]和SDR03.DAT[8]位为保留位, 读取时为0, 写入值也应为0。

SDRmn 是单元m和信道 n 的发送和接收数据寄存器 (16 位)。

SDR00、SDR01 的 DAT[8:0] 位或SDR02^{*1} 和SDR03^{*1}的 DAT[7:0] 位用作发送和接收缓冲寄存器, 而 STCLK[6:0] 位用作设置操作时钟(f_{MCK})分频比的寄存器。

如果串行模式寄存器 mn (SMRmn) 的 CCS 位被清除为 0, 则通过将操作时钟除以 STCLK[6:0] 位而设置的时钟将用作传输时钟。

如果串行模式寄存器 mn (SMRmn) 的 CCS 位设置为 1, 则将 SDR00 和 SDR01 的 STCLK[6:0] 位设置为 0000000b。来自 SCKp 引脚的输入时钟 f_{SCK} (简化 SPI 模式下的从设备传输) 用作传输时钟。

DAT[7:0] 或 DAT[8:0] 位用作发送和接收缓冲寄存器。接收期间, 移位寄存器转换后的并行数据存储在 DAT[7:0] 或 DAT[8:0] 位中; 发送期间, 要发送到移位寄存器的数据被设置为 DAT[7:0] 或 DAT[8:0] 位。

SDRmn 寄存器可以进行 16 位访问的读取或写入。

然而, STCLK[6:0] 位只能在操作停止时(SEm.SE[n] = 0)进行写入或读取。在操作期间(SEm.SE[n] = 1) 仅向 D AT[7:0] 或 DAT[8:0] 位写入值。在操作期间读取 SDRmn 寄存器时, STCLK[6:0] 位始终读取为 0。

根据串行通信操作设置寄存器 mn (SCRmn) 的 DLS[1:0] 或 DAT[8:0] 位的设置, 存储在 DAT[7:0或 DAT:0位中的数据如下, 与数据的输出顺序无关。

- 7-位数据长度 (存储在 DAT 中[6:0] 位)
- 8 位数据长度 (存储在 DAT[7:0] 位中)
- 9 位数据长度 (存储在 DAT[8:0] 位中) *1

注1: 只有以下UART支持9位数据长度。

- UART0

注: m: 单元编号(m = 0), n: 通道编号(n = 0至3)

19.3.9 SIRmn : Serial Flag Clear Trigger Register mn (mn = 00, 02)

Base address: SAUm = 0x400A_2000 + 0x200 × m

Offset address: 0x0108 + 0x2 × n

Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PECT	OVCT
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
0	OVCT	Clear Trigger of Overrun Error Flag of Channel n 0: Not cleared 1: Clears the OVF bit of the SSRmn register to 0	R/W
1	PECT	Clear Trigger of Parity Error Flag of Channel n 0: Not cleared 1: Clears the PEF bit of the SSRmn register to 0.	R/W
15:2	—	These bits are read as 0. The write value should be 0.	R/W

The SIRmn is a trigger register that is used to clear each error flag of channel n.

When each bit (PECT, OVCT) of this register is set to 1, the corresponding bit (PEF, OVF) of serial status register mn is cleared to 0. Because the SIRmn is a trigger register, it is cleared immediately when the corresponding bit of the SSRmn register is cleared. When the SIRmn register is read, 0x0000 is always read.

19.3.10 SIRmn : Serial Flag Clear Trigger Register mn (mn = 01, 03)

Base address: SAUm = 0x400A_2000 + 0x200 × m

Offset address: 0x0108 + 0x2 × n

Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	—	—	—	—	—	—	—	—	—	—	—	—	—	FECT	PECT	OVCT
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
0	OVCT	Clear Trigger of Overrun Error Flag of Channel n 0: Not cleared 1: Clears the OVF bit of the SSRmn register to 0	R/W
1	PECT	Clear Trigger of Parity Error Flag of Channel n 0: Not cleared 1: Clears the PEF bit of the SSRmn register to 0	R/W
2	FECT	Clear Trigger of Framing Error Flag of Channel n 0: Not cleared 1: Clears the FEF bit of the SSRmn register to 0	R/W
15:3	—	These bits are read as 0. The write value should be 0.	R/W

The SIRmn is a trigger register that is used to clear each error flag of channel n.

When each bit (FECT, PECT, OVCT) of this register is set to 1, the corresponding bit (FEF, PEF, OVF) of serial status register mn is cleared to 0. Because the SIRmn is a trigger register, it is cleared immediately when the corresponding bit of the SSRmn register is cleared. When the SIRmn register is read, 0x0000 is always read.

19.3.9 SIRmn: 串口标志清除触发寄存器 mn(mn = 00, 02)

基地址: SAUm = 0x400A_2000 + 0x200 × m

偏移地址: 0x0108 + 0x2 × n

位位置:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PECT	OVCT
重置后的值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
0	OVCT	清除通道 n 的溢出错误标志触发器 0: 未清除 1: 将 SSRmn 寄存器的 OVF 位清除为 0	R/W
1	PECT	通道 n 0 的奇偶校验错误标志清除触发器: 未清除 1: 将 SSRmn 寄存器的 PEF 位清除为 0。	R/W
15:2	—	这些位读取为 0。写入值应为 0。	R/W

SIRmn 是一个触发寄存器，用于清除通道 n 的每个错误标志。

当该寄存器的每一位（PECT、OVCT）被置为 1 时，串行状态寄存器mn的对应位（PEF、OVF）被清零。由于 SIRmn 是一个触发寄存器，因此当SSRmn寄存器的对应位被清零时，SIRmn 也会立即被清零。读取 SIRmn 寄存器时，总是会读取0x0000。

19.3.10SIRmn： 串行标志清除触发寄存器 mn(mn = 01, 03)

基地址: SAUm = 0x400A_2000 + 0x200 × m

偏移地址: 0x0108 + 0x2 × n

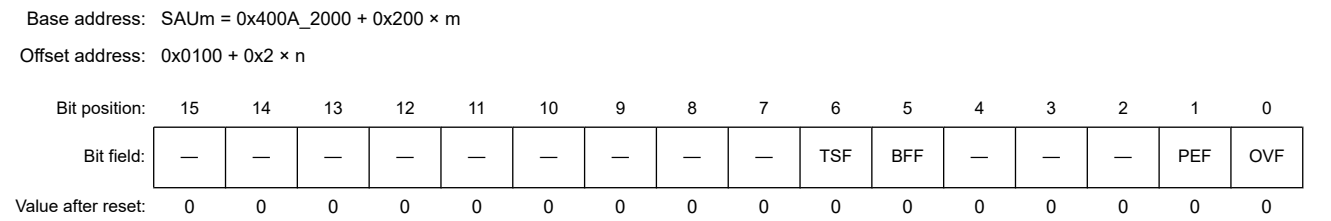
位位置:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域:	—	—	—	—	—	—	—	—	—	—	—	—	—	FECT	PECT	OVCT
重置后的值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
0	OVCT	清除通道 n 的溢出错误标志触发器 0: 未清除 1: 将 SSRmn 寄存器的 OVF 位清除为 0	R/W
1	PECT	清除通道 n 的奇偶校验错误标志触发 0: 未清除 1: 将 SSRmn 寄存器的 PEF 位清零。	R/W
2	FECT	清除信道 n 的帧错误标志触发 0: 未清除 1: 将 SSRmn 寄存器的 FEF 位清零。	R/W
15:3	—	这些位读取为 0。写入值应为 0。	R/W

SIRmn 是一个触发寄存器，用于清除通道 n 的每个错误标志。

当该寄存器的每一位（FECT、PECT、OVCT）被置为 1 时，串行状态寄存器mn的对应位（FEF、PEF、OVF）将被清零。由于 SIRmn 是一个触发寄存器，因此当 SSRmn 寄存器的对应位被清零时，SIRmn 也会立即被清零。读取 SIRmn 寄存器时，总是会读取0x0000。

19.3.11 SSRmn : Serial Status Register mn (mn = 00, 02)



Bit	Symbol	Function	R/W
0	OVF	Overrun Error Detection Flag of Channel n 0: No error occurs 1: An error occurs	R
1	PEF	Parity or ACK Error Detection Flag of Channel n 0: No error occurs 1: Parity error occurs (during UART reception) or ACK is not detected (during I ² C transmission)	R
4:2	—	These bits are read as 0.	R
5	BFF	Flag Indicating the State of the Buffer Register for Channel n 0: Valid data is not stored in the SDRmn register 1: Valid data is stored in the SDRmn register	R
6	TSF	Flag Indicating the State of Communications for Channel n 0: Communication is stopped or suspended 1: Communication is in progress	R
15:7	—	These bits are read as 0.	R

Note: When the simplified SPI is handling reception in the Snooze mode (SSC0.SWC = 1), the OVF flag and the BFF flag do not change.

Note: If data is written to the SDRmn register when BFF = 1, the transmit or receive data stored in the register is discarded and an overrun error (OVF = 1) is detected.

The SSRmn register indicates the state of communications and occurrence of errors for channel n. The errors indicated by this register are a framing error, parity error, and overrun error.

OVF bit (Overrun Error Detection Flag of Channel n)

<Clearing condition>

- 1 is written to the OVCT bit of the SIRmn register.

<Setting condition>

- Even though receive data is stored in the SDRmn register, that data is not read and transmit data or the next receive data is written while the TRXE[0] bit of the SCRmn register is set to 1 (reception or transmission and reception mode in each communication mode).
- Transmit data is not ready for slave transmission or transmission and reception in simplified SPI mode.

PEF bit (Parity or ACK Error Detection Flag of Channel n)

<Clearing condition>

- 1 is written to the PECT bit of the SIRmn register.

<Setting condition>

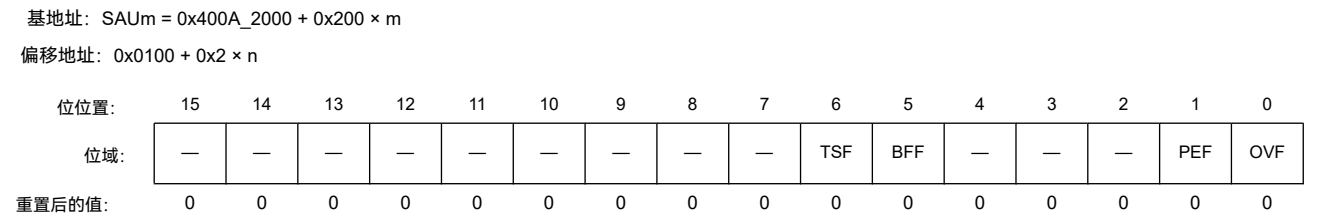
- The parity of the transmit data and the parity bit do not match when UART reception ends (parity error).
- No ACK signal is returned from the slave at the ACK reception timing during I²C transmission (ACK is not detected).

BFF bit (Flag Indicating the State of the Buffer Register for Channel n)

<Clearing condition>

- Transferring transmit data from the SDRmn register to the shift register ends during transmission.

19.3.11 SSRmn: 串行状态寄存器 mn(mn = 00, 02)



少量	象征	功能	R/W
0	OVF	通道 n 的溢出错误检测标志 0: 未发生错误 1: 发生错误	R
1	PEF	通道 n 的奇偶校验或 ACK 错误检测标志: 0: 未发生错误 1: 发生奇偶校验错误（在 UART 接收期间）或未检测到 ACK（在I ² C传输期间）	R
4:2	—	这些位读取为 0。	R
5	BFF	指示通道 n 的缓冲寄存器状态的标志 0: SDRmn 寄存器中未存储有效数据。 1: 有效数据存储在 SDRmn 寄存器中。	R
6	TSF	指示信道 n 通信状态的标志 0: 通信已停止或暂停 1: 沟通正在进行中	R
15:7	—	这些位读取为 0。	R

注意: 当简化的 SPI 在 Snooze 模式(SSC0.SWC = 1下处理接收时, OVF 标志和 BFF 标志不会改变。

注意: 如果在BFF = 1, 时向 SDRmn 寄存器写入数据, 则存储在寄存器中的发送或接收数据将被丢弃, 并且检测到溢出错误(OVF = 1)。

SSRmn 寄存器指示信道 n 的通信状态和错误发生情况。该寄存器指示的错误包括帧错误、奇偶校验错误和溢出错误。

OVF 位（通道 n 的溢出错误检测标志）

<清除条件>

- 1 被写入 SIRmn 寄存器的 OVCT 位。

<设置条件>

- 即使接收数据存储在 SDRmn 寄存器中, 当 SCRmn 寄存器的 TRXE [0] 位设置为 1 时（在每种通信模式下, 接收或发送和接收模式）, 也不会读取该数据并发送数据或写入下一个接收数据。
- 传输数据尚未准备好进行从设备传输或在简化的SPI模式下进行传输和接收。

PEF 位（通道 n 的奇偶校验或 ACK 错误检测标志）

<清除条件>

- 1 被写入 SIRmn 寄存器的 PECT 位。

<设置条件>

- 发送数据的奇偶校验位与 UART 接收结束时的奇偶校验位不匹配（奇偶校验错误）。
- 在I²C传输期间, ACK接收时从设备未返回ACK信号（未检测到ACK）。

BFF 位（指示通道 n 的缓冲寄存器状态的标志）

<清除条件>

- 传输过程中, 将发送数据从 SDRmn 寄存器传输到移位寄存器的过程结束。

- Reading receive data from the SDRmn register ends during reception.
- The ST[n] bit of the STm register is set to 1 (communication is stopped) or the SS[n] bit of the SSm register is set to 1 (communication is enabled).

<Setting condition>

- Transmit data is written to the SDRmn register while the TRXE[1] bit of the SCRmn register is set to 1 (transmission or transmission and reception mode in each communication mode).
- Receive data is stored in the SDRmn register while the TRXE[0] bit of the SCRmn register is set to 1 (reception or transmission and reception mode in each communication mode).
- A reception error occurs.

TSF bit (Flag Indicating the State of Communications for Channel n)

<Clearing condition>

- The ST[n] bit of the STm register is set to 1 (communication is stopped) or the SS[n] bit of the SSm register is set to 1 (communication is suspended).
- Communication ends.

<Setting condition>

- Communication starts.

19.3.12 SSRmn : Serial Status Register mn (mn = 01, 03)

Base address:	SAUm = 0x400A_2000 + 0x200 × m															
Offset address:	0x0100 + 0x2 × n															
Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	—	—	—	—	—	—	—	—	—	TSF	BFF	—	—	FEF	PEF	OVF
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
0	OVF	Overrun Error Detection Flag of Channel n 0: No error occurs 1: An error occurs	R
1	PEF	Parity or ACK Error Detection Flag of Channel n 0: No error occurs 1: Parity error occurs (during UART reception) or ACK is not detected (during I ² C transmission)	R
2	FEF	Framing Error Detection Flag of Channel n 0: No error occurs 1: An error occurs (during UART reception)	R
4:3	—	These bits are read as 0.	R
5	BFF	Flag Indicating the State of the Buffer Register for Channel n 0: Valid data is not stored in the SDRmn register 1: Valid data is stored in the SDRmn register	R
6	TSF	Flag Indicating the State of Communications for Channel n 0: Communication is stopped or suspended 1: Communication is in progress	R
15:7	—	These bits are read as 0. The write value should be 0.	R

Note: When the simplified SPI is handling reception in the Snooze mode (SSC0.SWC = 1), the OVF flag and the BFF flag do not change.
Note: If data is written to the SDRmn register when BFF = 1, the transmit or receive data stored in the register is discarded and an overrun error (OVF = 1) is detected.

- 从 SDRmn 寄存器读取接收数据在接收过程中结束。
- STm 寄存器的ST[n]位设置为 1（通信停止）或 SSm 寄存器的 SS[n] 位设置为 1（通信启用）。

<设置条件>

- 发送数据写入 SDRmn 寄存器，同时 SCRmn 寄存器的 TRXE [1] 位设置为 1（在每种通信模式下，发送或发送和接收模式）。
- 接收数据存储在 SDRmn 寄存器中，同时 SCRmn 寄存器的 TRXE [0] 位设置为 1（接收或发送以及每种通信模式下的接收模式）。
- 接收错误。

TSF 位（指示信道 n 通信状态的标志）

<清除条件>

- STm 寄存器的 ST[n] 位设置为 1（通信停止）或 SSm 寄存器的SS[n] 位设置为 1（通信暂停）。

- 通讯结束。

<设置条件>

- 沟通开始。

19.3.12SSRmn： 串行状态寄存器 mn(mn = 01, 03)

基地址: SAUm = 0x400A_2000 + 0x200 × m																		
偏移地址: 0x0100 + 0x2 × n																		
位位置:	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																	
位域:	<table><tr><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>TSF</td><td>BFF</td><td>—</td><td>—</td><td>FEF</td><td>PEF</td><td>OVF</td></tr></table>	—	—	—	—	—	—	—	—	—	—	TSF	BFF	—	—	FEF	PEF	OVF
—	—	—	—	—	—	—	—	—	—	TSF	BFF	—	—	FEF	PEF	OVF		
重置后的值:	<table><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		

少量	象征	功能	R/W
0	OVF	通道 n 的溢出错误检测标志 0: 未发生错误 1: 发生错误	R
1	PEF	通道 n 的奇偶校验或 ACK 错误检测标志: 0: 未发生错误 1: 发生奇偶校验错误（在 UART 接收期间）或未检测到 ACK（在I ² C传输期间）	R
2	FEF	信道 n 的帧错误检测标志 0: 未发生错误 1: 发生错误（UART接收期间）	R
4:3	—	这些位读取为 0。	R
5	BFF	指示通道 n 的缓冲寄存器状态的标志 0: SDRmn 寄存器中未存储有效数据。 1: 有效数据存储在 SDRmn 寄存器中。	R
6	TSF	指示信道 n 通信状态的标志 0: 通信已停止或暂停 1: 沟通正在进行中	R
15:7	—	这些位读取为 0。写入值应为 0。	R

注意: 当简化的 SPI 在 Snooze 模式(SSC0.SWC = 1)下处理接收时，OVF 标志和 BFF 标志不会改变。
注意: 如果在BFF = 1, 时向 SDRmn 寄存器写入数据，则存储在寄存器中的发送或接收数据将被丢弃，并且检测到溢出错误(OVF = 1)。

OVF bit (Overflow Error Detection Flag of Channel n)

<Clearing condition>

- 1 is written to the OVCT bit of the SIRmn register.

<Setting condition>

- Even though receive data is stored in the SDRmn register, that data is not read and transmit data or the next receive data is written while the TRXE[0] bit of the SCRmn register is set to 1 (reception or transmission and reception mode in each communication mode).
- Transmit data is not ready for slave transmission or transmission and reception in simplified SPI mode.

PEF bit (Parity or ACK Error Detection Flag of Channel n)

<Clearing condition>

- 1 is written to the PECT bit of the SIRmn register.

<Setting condition>

- The parity of the transmit data and the parity bit do not match when UART reception ends (parity error).
- No ACK signal is returned from the slave at the ACK reception timing during I²C transmission (ACK is not detected).

FEF bit (Framing Error Detection Flag of Channel n)

<Clearing condition>

- 1 is written to the FECT bit of the SIRmn register.

<Setting condition>

- A stop bit is not detected when UART reception ends.

BFF bit (Flag Indicating the State of the Buffer Register for Channel n)

<Clearing condition>

- Transferring transmit data from the SDRmn register to the shift register ends during transmission.
- Reading receive data from the SDRmn register ends during reception.
- The ST[n] bit of the STm register is set to 1 (communication is stopped) or the SS[n] bit of the SSm register is set to 1 (communication is enabled).

<Setting condition>

- Transmit data is written to the SDRmn register while the TRXE[1] bit of the SCRmn register is set to 1 (transmission or transmission and reception mode in each communication mode).
- Receive data is stored in the SDRmn register while the TRXE[0] bit of the SCRmn register is set to 1 (reception or transmission and reception mode in each communication mode).
- A reception error occurs.

TSF bit (Flag Indicating the State of Communications for Channel n)

<Clearing condition>

- The ST[n] bit of the STm register is set to 1 (communication is stopped) or the SS[n] bit of the SSm register is set to 1 (communication is suspended).
- Communication ends.

<Setting condition>

- Communication starts.

OVF 位（通道 n 的溢出错误检测标志）

<清除条件>

将● 1 写入 SIRmn 寄存器的 OVCT 位。

<设置条件>

- 即使接收数据存储在 SDRmn 寄存器中，当 SCRmn 寄存器的 TRXE [0] 位设置为 1 时（在每种通信模式下，接收或发送和接收模式），也不会读取该数据并发送数据或写入下一个接收数据。
- 传输数据尚未准备好进行从设备传输或在简化的SPI模式下进行传输和接收。

PEF 位（通道 n 的奇偶校验或 ACK 错误检测标志）

<清除条件>

- 1 被写入 SIRmn 寄存器的 PECT 位。

<设置条件>

- 发送数据的奇偶校验位与 UART 接收结束时的奇偶校验位不匹配（奇偶校验错误）。
- 在I²C传输期间，ACK接收时从从设备未返回ACK信号（未检测到ACK）。

FEF 位（信道 n 的帧错误检测标志）

<清除条件>

- 1 被写入 SIRmn 寄存器的 FECT 位。

<设置条件>

- UART接收结束时未检测到停止位。

BFF 位（指示通道 n 的缓冲寄存器状态的标志）

<清除条件>

- 传输过程中，将发送数据从 SDRmn 寄存器传输到移位寄存器的过程结束。
- 从 SDRmn 寄存器读取接收数据在接收过程中结束。
- STm 寄存器的 ST[n] 位设置为 1（通信停止）或 SSm 寄存器的 SS[n] 位设置为 1（通信启用）。

<设置条件>

- 发送数据写入 SDRmn 寄存器，同时 SCRmn 寄存器的 TRXE [1] 位设置为 1（在每种通信模式下，发送或发送和接收模式）。
- 接收数据存储在 SDRmn 寄存器中，同时 SCRmn 寄存器的 TRXE [0] 位设置为 1（接收或发送以及每种通信模式下的接收模式）。
- 接收错误。

TSF 位（指示信道 n 通信状态的标志）

<清除条件>

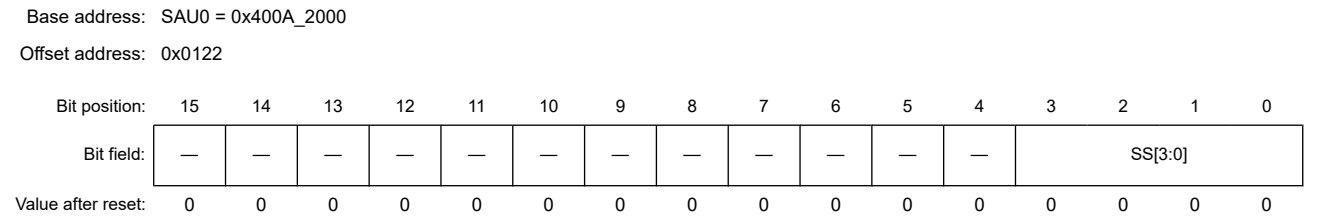
- STm 寄存器的 ST[n] 位设置为 1（通信停止）或 SSm 寄存器的 SS[n] 位设置为 1（通信暂停）。

- 通讯结束。

<设置条件>

- 沟通开始。

19.3.13 SS0 : Serial Channel Start Register 0



Bit	Symbol	Function	R/W
3:0	SS[3:0]	Operation Start Trigger of Channel n 0: No trigger operation 1: Set the SE0.SE[n] bit to 1 to place the channel in communications waiting state*1	R/W
15:4	—	These bits are read as 0. The write value should be 0.	R/W

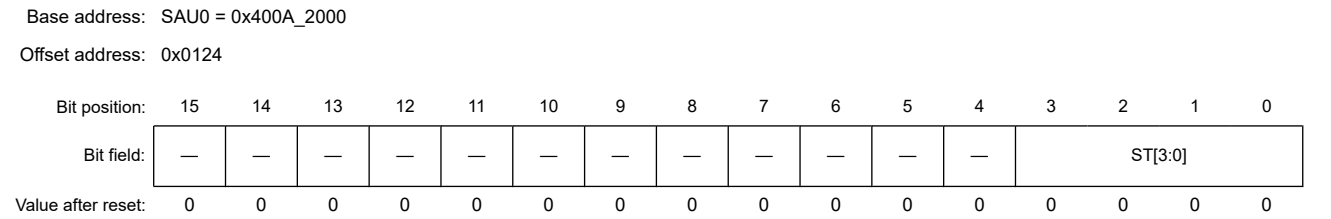
Note: For the UART reception, set the TRXE[0] bit of SCR0n register to 1, and then be sure to set the SS[n] bit to 1 after at least 4 f_{MCK} clock cycles have elapsed.

Note 1. Setting an SS[n] bit to 1 during communications stops communications through channel n and places the channel in the waiting state. At this time, the values of the control registers and shift register, the states of the SCKp and SOp pins, and the values of the SSR0n.FEF, PEF, and OVF flags are retained.

The SS0 is a trigger register that is used to enable starting communication or count by each channel of serial array unit 0.

When 1 is written to a bit (SS[n]) of this register, the corresponding bit (SE[n]) of serial channel enable status register 0 (SE0) is set to 1 (operation is enabled). Because the SS[n] bit is a trigger bit, it is cleared immediately when SE0.SE[n] = 1. When the SS0 register is read, 0x0000 is always read.

19.3.14 ST0 : Serial Channel Stop Register 0



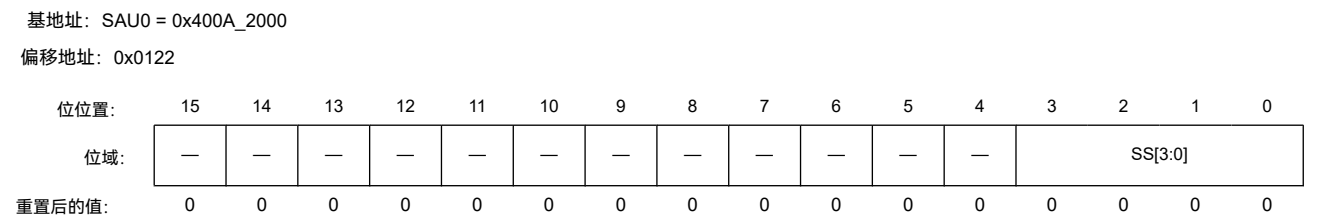
Bit	Symbol	Function	R/W
3:0	ST[3:0]	Operation Stop Trigger of Channel n 0: No trigger operation 1: Clears the SE0.SE[n] bit to 0 and stops the communication operation*1	R/W
15:4	—	These bits are read as 0. The write value should be 0.	R/W

Note 1. The values of the control registers and shift register, the states of the SCKp and SOp pins, and the values of the SSR0n.FEF, PEF, and OVF flags are retained.

The ST0 register is a trigger register that is used to enable stopping communication or count by each channel of serial array unit 0.

When 1 is written to a bit (ST[n]) of this register, the corresponding bit (SE[n]) of serial channel enable status register 0 (SE0) is cleared to 0 (operation is stopped). Because the ST[n] bit is a trigger bit, it is cleared immediately when SE0.SE[n] = 0. When the ST0 register is read, 0x0000 is always read.

19.3.13SS0： 串行通道起始寄存器 0



少量	象征	功能	R/W
3:0	SS[3:0]	通道 n 的操作启动触发 0: 无触发操作 1: 将 SE0.SE[n] 位设置为 1，使通道处于通信等待状态*1	R/W
15:4	—	这些位读取为 0。写入值应为 0。	R/W

注意：对于 UART 接收，将 SCR0n 寄存器的 TRXE [0] 位设置为 1，然后确保在至少经过 4 个 fMCK 时钟周期后将 SS[n] 位设置为 1。

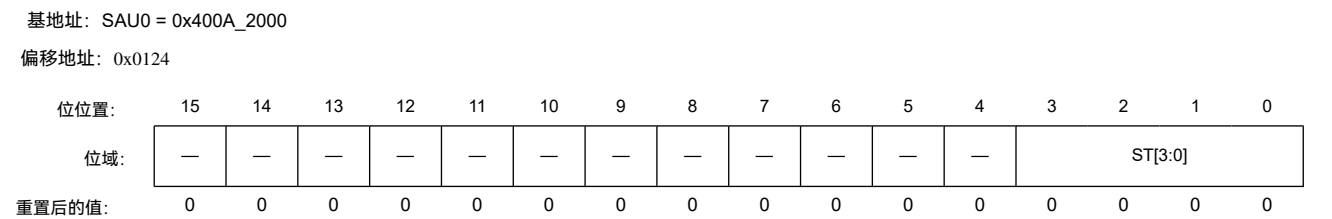
注1：在通信过程中，将SS[n] 位设置为1会停止通过通道n的通信，并将该通道置于等待状态。此时，控制寄存器和移位寄存器的值、SCKp 和 SOp 引脚的状态以及SSR0n.FEF、PEF和OVF标志的值均保持不变。

SS0 是一个触发寄存器，用于使串行阵列单元 0 的每个通道能够开始通信或计数。

当向该寄存器的位(SS[n]) 写入1时，串行通道使能状态寄存器0 (SE0) 的对应位(SE[n])被置为1（操作使能）。由于SS[n] 位是触发位，因此当SE0.SE[n] = 1被写入1时，它立即被清除。

读取 SS0 寄存器时，总是会读取0x0000。

19.3.14ST0： 串行通道停止寄存器 0



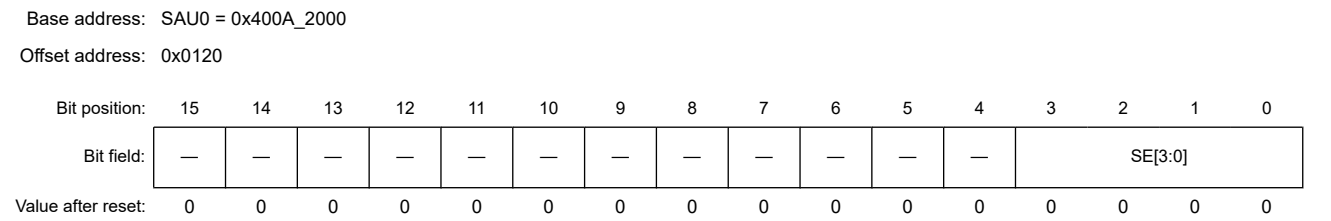
少量	象征	功能	R/W
3:0	ST[3:0]	通道 n 的操作停止触发 0: 无触发操作 1: 将 SE0.SE[n] 位清除为 0，并停止通信操作*1	R/W
15:4	—	这些位读取为 0。写入值应为 0。	R/W

注 1. 控制寄存器和移位寄存器的值、SCKp 和 SOp 引脚的状态以及 SSR0n.FEF、PEF 和 OVF 标志的值均被保留。

ST0 寄存器是一个触发寄存器，用于使串行阵列单元 0 的每个通道停止通信或计数。

当向该寄存器的(ST[n]) 位写入 1 时，串行通道使能状态寄存器 0 (SE0) 的对应位 ([位、]位) 将被清零（操作停止）。由于ST[n] 位是触发位，因此当SE0.SE[n]位和= 0位被写入时，位也会立即被清零。读取 ST0 寄存器时，始终会读取0x000 0位。

19.3.15 SE0 : Serial Channel Enable Status Register 0



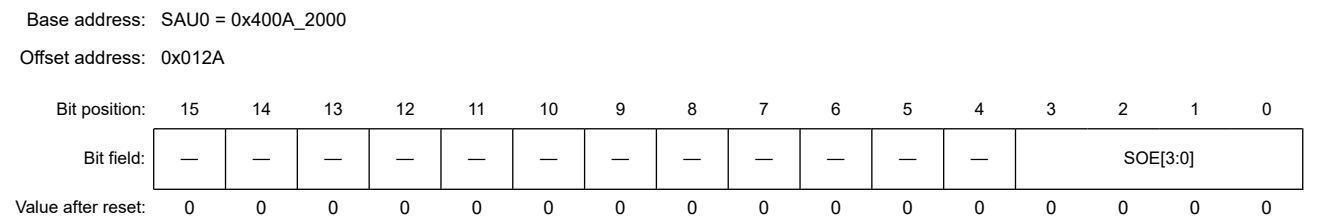
Bit	Symbol	Function	R/W
3:0	SE[3:0]	Indication of whether Operation of Channel n is Enabled or Stopped. 0: Operation stops 1: Operation is enabled	R
15:4	—	These bits are read as 0.	R

The SE0 register indicates whether data transmission and reception operation of each channel of serial array unit 0 is enabled or stopped. When 1 is written to a bit of serial channel start register 0 (SS0), the corresponding bit of this register is set to 1. When 1 is written to a bit of serial channel stop register 0 (ST0), the corresponding bit is cleared to 0.

For channel n whose operation is enabled, the value of the CKO[n] bit of serial output register 0 (SO0) to be described later cannot be rewritten by software, and a value reflected by a communication operation is output from the serial clock pin.

For channel n whose operation is stopped, the value of the CKO[n] bit of the SO0 register can be set by software and is output from the serial clock pin. In this way, any waveform, such as that of a start condition or stop condition, can be created by software.

19.3.16 SOE0 : Serial Output Enable Register 0



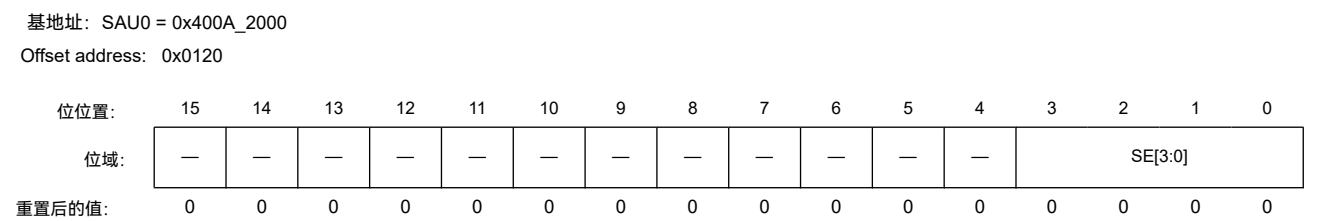
Bit	Symbol	Function	R/W
3:0	SOE[3:0]	Serial Output Enable or Stop of Channel n 0: Stops output by serial communication operation 1: Enables output by serial communication operation	R/W
15:4	—	These bits are read as 0. The write value should be 0.	R/W

The SOE0 register is used to enable or stop output of the serial communication operation of each channel of serial array unit 0.

For channel n whose serial output is enabled, the value of the SO[n] bit of serial output register 0 (SO0) to be described later cannot be rewritten by software, and a value reflected by a communication operation is output from the serial data output pin.

For channel n, whose serial output is stopped, the SO[n] bit value of the SO0 register can be set by software, and that value can be output from the serial data output pin. In this way, any waveform, such as that of a start condition or stop condition, can be created by software.

19.3.15SE0： 串行通道使能状态寄存器 0



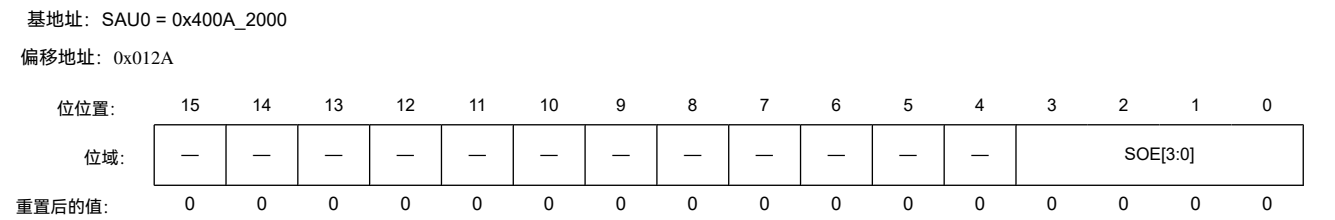
少量	象征	功能	R/W
3:0	SE[3:0]	指示通道 n 的操作是启用还是停止。 0：操作停止；1：操作已启用	R
15:4	—	这些位读取为 0。	R

SE0寄存器指示串行阵列单元0各通道的数据传输和接收操作是否启用或停止。当串行通道起始寄存器0（SS0）的某一位写入1时，该寄存器的对应位被置为1。当串行通道停止寄存器0（ST0）的某一位写入1时，该寄存器的对应位被清零。

对于已启用操作的通道 n，稍后将描述的串行输出寄存器 0 (SO0) 的CKO[n] 位的值不能通过软件重写，并且通过通信操作反映的值将从串行时钟引脚输出。

对于停止运行的通道 n，可以通过软件设置 SO0 寄存器的CKO[n]位的值，并将其从串行时钟引脚输出。这样，就可以利用软件生成任何波形，例如启动条件波形或停止条件波形。

19.3.16SOE0： 串行输出使能寄存器 0



少量	象征	功能	R/W
3:0	SOE[3:0]	通道 n 的串行输出使能或停止 0：停止串行通信操作的输出 1：启用串行通信操作的输出	R/W
15:4	—	这些位读取为 0。写入值应为 0。	R/W

SOE0 寄存器用于启用或停止串行阵列单元 0 的每个通道的串行通信操作的输出。

对于启用串行输出的通道 n，稍后将描述的串行输出寄存器 0 (SO0) 的SO[n] 位的值不能通过软件重写，并且通信操作反映的值将从串行数据输出引脚输出。

对于串行输出停止的通道 n，可以通过软件设置 SO0 寄存器的SO[n]位值，并将该值从串行数据输出引脚输出。这样，就可以利用软件生成任意波形，例如起始条件波形或停止条件波形。

19.3.17 SO0 : Serial Output Register 0

Base address:	SAU0 = 0x400A_2000															
Offset address:	0x0128															
Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	—	—	—	—	CKO[3:0]				—	—	—	—	SO[3:0]			
Value after reset:	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1

Bit	Symbol	Function	R/W
3:0	SO[3:0]	Serial Data Output of Channel n 0: Serial data output value is 0 1: Serial data output value is 1	R/W
7:4	—	These bits are read as 0. The write value should be 0.	R/W
11:8	CKO[3:0]	Serial Clock Output of Channel n 0: Serial clock output value is 0 1: Serial clock output value is 1	R/W
15:12	—	These bits are read as 0. The write value should be 0.	R/W

The SO0 is a buffer register for serial output of each channel of serial array unit 0.

The value of the SO[n] bit of this register is output from the serial data output pin of channel n.

The value of the CKO[n] bit of this register is output from the serial clock output pin of channel n.

The SO[n] bit of this register can be rewritten by software only when serial output is disabled (SOE0.SOE[n] = 0). When serial output is enabled (SOE0.SOE[n] = 1), rewriting by software is ignored, and the value of the register can be changed only by a serial communication operation.

The CKO[n] bit of this register can be rewritten by software only when the channel operation is stopped (SE0.SE[n] = 0). While channel operation is enabled (SE0.SE[n] = 1), rewriting by software is ignored, and the value of the CKO[n] bit can be changed only by a serial communication operation.

19.3.18 SOL0 : Serial Output Level Register 0

Base address:	SAU0 = 0x400A_2000															
Offset address:	0x0134															
Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	—	—	—	—	—	—	—	—	—	—	—	—	SOL2	—	SOL0	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
0	SOL0	Selects Inversion of the Level of the Transmit Data of Channel 0 in UART Mode 0: Communication data is output as is 1: Communication data is inverted and output	R/W
1	—	This bit is read as 0. The write value should be 0.	R/W
2	SOL2	Selects Inversion of the Level of the Transmit Data of Channel 2 in UART Mode 0: Communication data is output as is 1: Communication data is inverted and output	R/W
15:3	—	These bits are read as 0. The write value should be 0.	R/W

The SOL0 register is used to set inversion of the data output level of each channel of serial array unit 0.

This register can be set only in the UART mode. Be sure to set 0 for the bit corresponding the channel used in the simplified SPI mode or simplified I²C mode.

Inverting channel n by using this register is reflected on pin output only when serial output is enabled (SOE0.SOE[n] = 1).

19.3.17 SO0: 串行输出寄存器 0

基地址: SAU0 = 0x400A_2000																	
Offset address: 0x0128																	
位位置:	1514131211109876543210																
位域:	<table><tr><td>—</td><td>—</td><td>—</td><td>—</td><td colspan="4">CKO[3:0]</td><td>—</td><td>—</td><td>—</td><td>—</td><td colspan="4">SO[3:0]</td></tr></table>	—	—	—	—	CKO[3:0]				—	—	—	—	SO[3:0]			
—	—	—	—	CKO[3:0]				—	—	—	—	SO[3:0]					
重置后的值:	0000111100001111																

少量	象征	功能	R/W
3:0	SO[3:0]	通道 n 的串行数据输出 0: 串口数据输出值为 0 1: 串口数据输出值为 1	R/W
7:4	—	这些位读取为 0。写入值应为 0。	R/W
11:8	CKO[3:0]	通道 n 的串行时钟输出 0: 串行时钟输出值为 0 1: 串行时钟输出值为 1	R/W
15:12	—	这些位读取为 0。写入值应为 0。	R/W

SO0 是串行阵列单元 0 的每个通道的串行输出的缓冲寄存器。

该寄存器的SO[n] 位的值由通道n的串行数据输出引脚输出。

该寄存器的CKO[n]位的值由通道n的串行时钟输出引脚输出。

只有当串行输出被禁用时，该寄存器的SO[n]位才能通过软件进行重写(SOE0.SOE[n] = 0)。当串行输出被启用时，软件重写将被忽略，寄存器的值只能通过串行通信操作进行更改。

只有当通道操作停止时，该寄存器的CKO[n] 位才能通过软件重写(SE0.SE[n] = 0)位。当通道操作启用时，软件重写将被忽略，并且CKO[n] 位的值只能通过串行通信操作进行更改。

19.3.18SOL0: 串行输出电平寄存器 0

基地址: SAU0 = 0x400A_2000

Offset address: 0x0134

位位置:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域:	—	—	—	—	—	—	—	—	—	—	—	—	—	SOL2	—	SOL0
重置后的值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
0	SOL0	选择反转 UART 模式下通道 0 的发送数据电平 0: 通信数据按原样输出 1: 通信数据被反转并输出	R/W
1	—	该位读取值为 0。写入值应为 0。	R/W
2	SOL2	选择反转 UART 模式下通道 2 的发送数据电平 0: 通信数据按原样输出 1: 通信数据被反转并输出	R/W
15:3	—	这些位读取为 0。写入值应为 0。	R/W

SOL0 寄存器用于设置串行阵列单元 0 的每个通道的数据输出电平的反转。

此寄存器只能在 UART 模式下设置。请确保将与简化 SPI 模式或简化I²C模式下使用的通道对应的位设置为 0。

使用此寄存器反转通道 n 仅在串行输出启用时才会反映在引脚输出上(SOE0.SOE[n] = 1)。

When serial output is disabled (SOE0.SOE[n] = 0), the value of the SO0.SO[n] bit is output as is.
Rewriting the SOL0 register is prohibited when the channel n is in operation (when SE0.SE[n] = 1).

Figure 19.3 shows examples in which the level of transmit data is reversed during UART transmission.

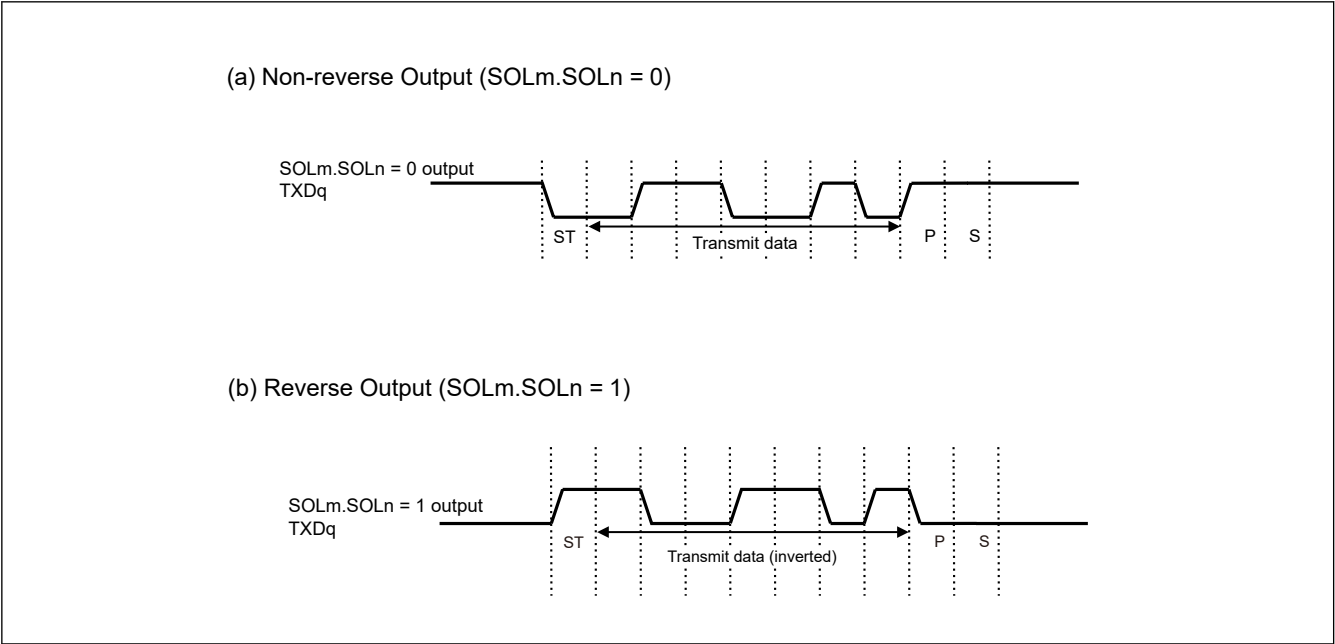


Figure 19.3 Examples of reverse transmit data

Note: m: Unit number (m = 0), n: Channel number (n = 0, 2), q: UART number (q = 0 to 1)

19.3.19 SSC0 : Serial Standby Control Register 0

Base address: SAU0 = 0x400A_2000
Offset address: 0x0138

Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SSEC	SWC
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
0	SWC	Setting of the Snooze Mode 0: Do not use the Snooze mode function 1: Use the Snooze mode function	R/W
1	SSEC	Selection of whether to Enable or Disable the Generation of Communication Error Interrupts in the Snooze Mode 0: Enable the generation of error interrupts SAU0_UART_ERRI0 1: Disable the generation of error interrupts SAU0_UART_ERRI0	R/W
15:2	—	These bits are read as 0. The write value should be 0.	R/W

The SSC0 register is used to control the startup of reception (the Snooze mode) while in the Software Standby mode when receiving SPI00 or UART0 serial data.

Note: The maximum transfer rate in the Snooze mode is as follows.

- When using SPI00: Up to 1 Mbps
- When using UART0: Up to 9600 bps (when setting the PCLKB = HOCO (32 MHz))

当串行输出被禁用时(SOE0.SOE[n] = 0), SO0.SO[n] 位的值将按原样输出。
当通道n正在运行时 (当SE0.SE[n] = 1处于工作状态时), 禁止重写 SOL0 寄存器。
图 19.3 显示了一些示例, 其中在 UART 传输期间发送数据的电平被反转。

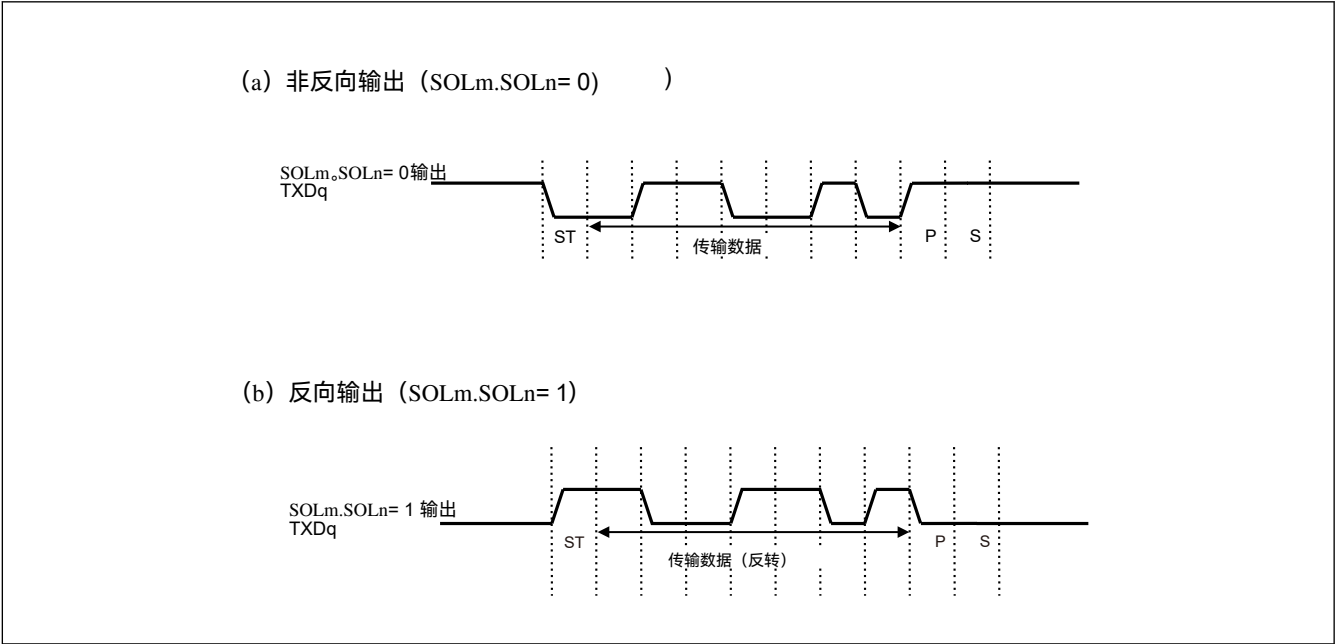


图 19.3 反向传输数据示例

注: m: 单元编号(m = 0), n: 通道编号(n = 0, 2), q: UART编号(q = 0至1)

19.3.19SSC0: 串行待机控制寄存器 0

基地址: SAU0 = 0x400A_2000
Offset address: 0x0138

位位置:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SSEC	SWC
重置后的值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
0	SWC	设置贪睡模式 0: 请勿使用“稍后提醒”模式功能 1: 使用贪睡模式功能	R/W
1	SSEC	选择是否在休眠模式下启用或禁用通信错误中断的生成 0: 启用错误中断 SAU0_UART_ERRI0 的生成; 1: 禁用错误中断 SAU0_UART_ERRI0 的生成。	R/W
15:2	—	这些位读取为 0。写入值应为 0。	R/W

当处于软件待机模式并接收 SPI00 或 UART0 串行数据时, SSC0 寄存器用于控制接收的启动 (休眠模式)。

注意: 在睡眠模式下, 最大传输速率如下。

- 使用SPI00时: 最高可达1 Mbps
- 使用 UART0 时: 最高可达 9600 bps (当设置 PCLKB= HOCO (32 MHz) 时)

SWC bit (Setting of the Snooze Mode)

- When there is a hardware trigger signal in the Software Standby mode, the Software Standby mode is exited, and simplified SPI or UART reception is performed without operating the CPU (the Snooze mode).
- The Snooze mode function can only be specified when the high-speed on-chip oscillator clock or medium-speed on-chip oscillator clock is selected for the CPU and peripheral hardware clock (PCLKB). If any other clock is selected, specifying this mode is prohibited.
- Even when using Snooze mode, be sure to set the SWC bit to 0 in normal operation mode and change it to 1 just before shifting to Software Standby mode. Also, after returning from software standby mode to normal operation mode, be sure to set the SWC bit to 0.

SSEC bit (Selection of whether to Enable or Disable the Generation of Communication Error Interrupts in the Snooze Mode)

- The SSEC bit can be set to 1 or 0 only when both the SWC and SCRmn.EOC bits are set to 1 during UART reception in the Snooze mode. In other cases, clear the SSEC bit to 0.
- Setting SSEC, SWC = 1, 0 is prohibited.

Table 19.4 shows the interrupt in UART reception operation in Snooze mode.

Table 19.4 Interrupt in UART reception operation in Snooze mode

SCRmn.EOC bit	SSEC bit	Reception ended successfully	Reception ended in an error
0	0	SAU0_UART_RXI0 is generated.	SAU0_UART_RXI0 is generated.
0	1	SAU0_UART_RXI0 is generated.	SAU0_UART_RXI0 is generated.
1	0	SAU0_UART_RXI0 is generated.	SAU0_UART_ERRI0 is generated.
1	1	SAU0_UART_RXI0 is generated.	No interrupt is generated.

19.3.20 ISC : Input Switch Control Register

Base address:	PORG_A = 0x400A_1000							
Offset address:	0x0003							
Bit position:	7	6	5	4	3	2	1	0
Bit field:	—	—	—	—	—	SSIE0 0	ISC1	ISC0
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
0	ISC0	Switching External Interrupt (IRQ0) Input 0: Uses the input signal of the IRQ0 pin as an external interrupt (normal operation) 1: Uses the input signal of the RXD0 pin as an external interrupt (wakeup signal detection)	R/W
1	ISC1	Switching Channel 7 Input of Timer Array Unit 0: Uses the input signal of the TI07 pin as a timer input (normal operation) 1: Input signal of the RXD0 pin is used as timer input (detects the wakeup signal and measures the low width of the break field and the pulse width of the sync field).	R/W
2	SSIE00	Setting of the SSIE00 Input of Channel 0 in the Communications Through SPI00 in the Slave Mode 0: The SSIE00 input is disabled. 1: The SSIE00 input is enabled.	R/W
7:3	—	These bits are read as 0. The write value should be 0.	R/W

Note: When the LIN-bus communication function is used, select the input signal of the RXD0 pin by setting ISC1 to 1.

The ISC1 and ISC0 bits of the ISC register are used to implement LIN-bus communication operation by using TAU channel 7 in association with the serial array unit. When bit 0 is set to 1, the input signal of the serial data input (RXD0) pin is selected as an external interrupt (IRQ0) that can be used to detect a wakeup signal. When bit 1 is set to 1, the input signal

SWC 位（设置贪睡模式）

- 当软件待机模式下有硬件触发信号时，软件待机模式退出，并执行简化的SPI或UART接收，而不操作CPU（休眠模式）。
- 仅当CPU和外设硬件时钟（PCLKB）选择高速片上振荡器时钟或中速片上振荡器时钟时，才能指定“休眠模式”功能。如果选择其他时钟，则禁止指定此模式。
- 即使在使用休眠模式时，也务必在正常运行模式下将SWC位设置为0，并在切换到软件待机模式之前将其更改为1。此外，从软件待机模式返回正常运行模式后，也务必将SWC位设置为0。

SSEC 位（选择是否在休眠模式下启用或禁用通信错误中断的生成）

- 仅当在休眠模式下 UART 接收时，如果 SWC 和 SCRmn.EOC 位均设置为 1，则 SSEC 位才能设置为 1 或 0。其他情况下，应将 SSEC 位清除为 0。
- 禁止设置 SSEC，SWC = 1, 0。

表 19.4 显示了 UART 接收操作在休眠模式下的中断。

表 19.4 休眠模式下 UART 接收操作中断

SCRmn.EOC 位	SSEC 位	接待工作圆满结束	接待工作以错误告终。
0	0	已生成 SAU0_UART_RXI0。	已生成 SAU0_UART_RXI0。
0	1	已生成 SAU0_UART_RXI0。	已生成 SAU0_UART_RXI0。
1	0	已生成 SAU0_UART_RXI0。	生成了 SAU0_UART_ERRI0。
1	1	已生成 SAU0_UART_RXI0。	未产生中断。

19.3.20 ISC: 输入开关控制寄存器

基本地址: PORGA = 0x400A_1000

偏移地址: 0x0003

位位置:	7	6	5	4	3	2	1	0
位域:	—	—	—	—	—	SSIE0 0	ISC1	ISC0
重置后的值:	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
0	ISC0	切换外部中断 (IRQ0) 输入 0: 使用 IRQ0 引脚的输入信号作为外部中断（正常运行） 1: 使用 RXD0 引脚的输入信号作为外部中断（唤醒信号检测）	R/W
1	ISC1	切换定时器阵列单元的通道 7 输入 0: 使用 TI07 引脚的输入信号作为定时器输入（正常操作） 1: 使用 RXD0 引脚的输入信号作为定时器输入（检测唤醒信号并测量断点低电平宽度和同步场脉冲宽度）。	R/W
2	SSIE00	在从模式下，通过 SPI00 进行通信时，设置通道 0 的 SSIE00 输入 0: SSIE00 输入已禁用。1: SSIE00 输入已启用。	R/W
7:3	—	这些位读取为 0。写入值应为 0。	R/W

注意: 当使用 LIN 总线通信功能时，将 ISC1 设置为 1 来选择 RXD0 引脚的输入信号。

ISC寄存器的ISC1和ISC0位用于通过与串行阵列单元关联的TAU通道7实现LIN总线通信操作。当位0设置为1时，串行数据输入（RXD0）引脚的输入信号被选为外部中断（IRQ0），可用于检测唤醒信号。

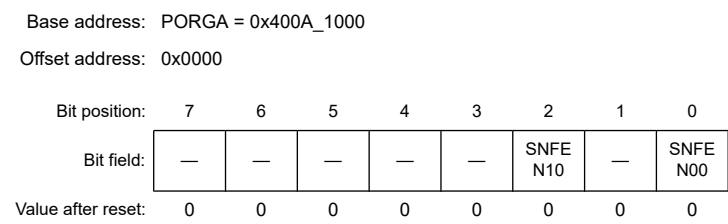
of the serial data input (RXD0) pin is selected as a timer input, so that wake up signal can be detected, the low width of the break field, and the pulse width of the sync field can be measured by the timer.

The SSIE00 bit is used to control the SSI00 input of channel 0 in the communications through SPI00 in the slave mode.

Reception and transmission do not proceed even if the serial clock is input while the SSI00 pin is being driven high.

Reception and transmission proceed in response to an input of the serial clock according to the mode setting while the SSI00 pin is being driven low.

19.3.21 SNFEN : SAU Noise Filter Enable Register



Bit	Symbol	Function	R/W
0	SNFEN00	Use of Noise Filter of RXD0 Pin 0: Noise filter OFF 1: Noise filter ON	R/W
1	—	This bit is read as 0. The write value should be 0.*1	R/W
2	SNFEN10	Use of Noise Filter of RXD1 Pin 0: Noise filter OFF 1: Noise filter ON	R/W
7:3	—	These bits are read as 0. The write value should be 0.*1	R/W

Note 1. Be sure to clear bits [7:3] and bit [1].

The SNFEN register is used to set whether the noise filter can be used for the input signal from the serial data input pin to each channel.

Disable the noise filter of the pin used for simplified SPI or simplified I²C communication, by clearing the corresponding bit of this register to 0.

Enable the noise filter of the pin used for UART communication, by setting the corresponding bit of this register to 1. When the noise filter is enabled, after synchronization is performed with the operation clock (f_{MCK}) of the target channel, 2-clock match detection is performed. When the noise filter is disabled, only synchronization is performed with the operation clock (f_{MCK}) of the target channel.

SNFEN00 bit (Use of Noise Filter of RXD0 Pin)

Set SNFEN00 to 1 to use the RXD0 pin.

Clear SNFEN00 to 0 to use the other than RXD0 pin.

SNFEN10 bit (Use of Noise Filter of RXD1 Pin)

Set SNFEN10 to 1 to use the RXD1 pin.

Clear SNFEN10 to 0 to use the other than RXD1 pin.

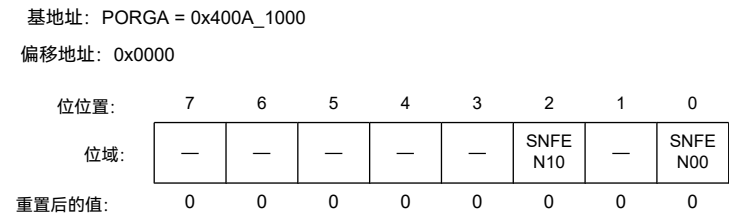
当位 1 设置为 1 时，串行数据输入 (RXD0) 引脚的输入信号被选作定时器输入，以便检测唤醒信号，定时器可以测量断路场的低电平宽度和同步场的脉冲宽度。

SSIE00 位用于在从模式下通过 SPI00 进行通信时控制通道 0 的 SSI00 输入。

即使在 SSI00 引脚被拉高时输入串行时钟，接收和发送也不会进行。

当 SSI00 引脚被拉低时，接收和发送会根据模式设置响应串行时钟的输入而进行。

19.3.21SNFEN：SAU 噪声滤波器使能寄存器



少量	象征	功能	R/W
0	SNFEN00	使用 RXD0 引脚的噪声滤波器 0: 降噪功能关闭 1: 噪声滤波器开启	R/W
1	—	该位读取值为 0。写入值应为 0.*1	R/W
2	SNFEN10	使用 RXD1 引脚的噪声滤波器 0: 降噪功能关闭 1: 噪声滤波器开启	R/W
7:3	—	这些位读取为 0。写入值应为 0.*1	R/W

注意 1. 务必清除位[7:3] 和位 [1] 。

SNFEN 寄存器用于设置是否可以使用噪声滤波器对来自串行数据输入引脚到每个通道的输入信号进行滤波。

通过将寄存器的相应位清除为 0，禁用用于简化 SPI 或简化I²C通信的引脚的噪声滤波器。

将该寄存器的相应位设置为 1，即可启用用于 UART 通信引脚的噪声滤波器。启用噪声滤波器后，在与目标通道的工作时钟(f_{MCK}) 同步后，将执行双时钟匹配检测。禁用噪声滤波器后，仅与目标通道的工作时钟(f_{MCK}) 进行同步。

SNFEN00 位（使用 RXD0 引脚的噪声滤波器）

将 SNFEN00 设置为 1 以使用 RXD0 引脚。

将 SNFEN00 清零以使用 RXD0 引脚以外的引脚。

SNFEN10 位（使用 RXD1 引脚的噪声滤波器）

将 SNFEN10 设置为 1 以使用 RXD1 引脚。

将 SNFEN10 清零以使用 RXD1 引脚以外的引脚。

19.3.22 ULBS : UART Loopback Select Register

Base address: PORGA = 0x400A_1000

Offset address: 0x0009

Bit position: 76543210

Bit field:

—	—	—	—	—	—	ULBS1	ULBS0

Value after reset: 00000000

Bit	Symbol	Function	R/W
0	ULBS0	Selection of the UART0 Loopback Function 0: Inputs the states of the RXD0 pin of serial array unit UART0 to the reception shift register. 1: Loops back output from the transmission shift register to the reception shift register.	R/W
1	ULBS1	Selection of the UART1 Loopback Function 0: Inputs the states of the RXD1 pin of serial array unit UART1 to the reception shift register. 1: Loops back output from the transmission shift register to the reception shift register.	R/W
7:2	—	These bits are read as 0. The write value should be 0.	R/W

The ULBS register is used to enable the UART loopback function. This register has bits to individually control UART channels. When the bit corresponding to each channel is set to 1, the UART loopback function is selected, and output from the transmission shift register is looped back to the reception shift register.

19.4 Operation Stop Mode

Each serial interface of serial array unit has the operation stop mode.

In this mode, serial communication cannot be executed, thus reducing the power consumption. In addition, the pin for serial interface can be used as port function pins in this mode.

The stopping of the operation by channels is set using each of the following registers.

Table 19.5 to Table 19.8 show each register setting when stopping the operation by channels.

(a) Serial channel stop register m (STm)

The STm is a trigger register that is used to enable stopping communication or count by each channel.

Table 19.5 Setting of serial channel stop register m (STm) when stopping the operation by channels

Bit	Symbol	Set value	Function
n	ST[n]	1	Operation stop trigger of channel n Because the ST[n] bit is a trigger bit, it is cleared immediately when SEm.SE[n] = 0. 1: Clears the SEm.SE[n] bit to 0 and stops the communication operation

(b) Serial channel enable status register m (SEm)

This register indicates whether data transmission and reception operation of each channel is enabled or stopped.

19.3.22 ULBS: UART 回环选择寄存器

基地址: PORGA= 0x400A_1000

Offset address: 0x0009

位位置: 76543210

位域:

—	—	—	—	—	—	ULBS1	ULBS0

重置后的值: 00000000

少量	象征	功能	R/W
0	ULBS0	选择 UART0 回环功能 0: 将串行阵列单元 UART0 的 RXD0 引脚的状态输入到接收移位寄存器。 1: 将发送移位寄存器的输出环回接收移位寄存器。	R/W
1	ULBS1	选择 UART1 回环功能 0: 将串行阵列单元 UART1 的 RXD1 引脚的状态输入到接收移位寄存器。 1: 将发送移位寄存器的输出环回接收移位寄存器。	R/W
7:2	—	这些位读取为 0。写入值应为 0。	R/W

ULBS寄存器用于启用UART回环功能。该寄存器包含用于单独控制各个UART通道的位。当每个通道对应的位都设置为1时，UART回环功能被选中，发送移位寄存器的输出将环回接收移位寄存器。

19.4 操作停止模式

串行阵列单元的每个串行接口都有操作停止模式。

在此模式下，无法执行串行通信，从而降低功耗。此外，串行接口引脚在此模式下可用作端口功能引脚。

通过以下每个寄存器设置按通道停止操作。

表 19.5 至表 19.8 显示了按通道停止操作时每个寄存器的设置。

(a) 串行通道停止寄存器m (STm)

STm 是一个触发寄存器，用于使每个通道能够停止通信或计数。

表 19.5 串行通道停止操作时串行通道停止寄存器m(STm) 的设置

少量	象征	设置值	功能
n	ST[n]	1	通道 n 的操作停止触发 因为 ST[n] 位是触发位，所以当SEm.SE[n] = 0时，它会立即被清除。 1: 将 SEm.SE[n] 位清除为 0 并停止通信操作

(b) 串行通道使能状态寄存器m (SEm)

该寄存器指示每个通道的数据传输和接收操作是否启用或停止。

Table 19.6 Status of serial channel enable status register m (SEm) when stopping the operation by channels

Bit	Symbol	Read value	Function
n	SE[n]	1 or 0	Indication of whether operation of channel n is enabled or stopped. With a channel whose operation is stopped, the value of the CKO[n] bit of the SOm register can be set by software. The SEm is a read-only status register, whose operation is stopped by using the STm register. 0: Operation stops 1: Operation is enabled.

(c) Serial output enable register m (SOEm)

This register is used to enable or stop output of the serial communication operation of each channel.

Table 19.7 Setting of serial output enable register m (SOEm) when stopping the operation by channels

Bit	Symbol	Set value	Function
n	SOE[n]	0	Serial output enable or stop of channel n For channel n, whose serial output is stopped, the SO[n] bit value of the SOm register can be set by software. 0: Stops output by serial communication operation

(d) Serial output register m (SOM)

The SOm is a buffer register for serial output of each channel.

Table 19.8 Setting of serial output register m (SOM) when sopping the operation by channels

Bit	Symbol	Set value	Function
n	SO[n]	1	Serial data output of channel n When using pins corresponding to each channel as port function pins, set the corresponding SO[n] bit to 1. 1: Serial data output value is 1
n+8	CKO[n]	1	Serial clock output of channel n When using pins corresponding to each channel as port function pins, set the corresponding CKO[n] bit to 1. 1: Serial clock output value is 1

Note: Unit number (m = 0), n: Channel number (n = 0 to 3)

19.5 Operation of Simplified SPI

This is a clocked communication function that uses three lines: serial clock (SCK) and serial data (SI and SO) lines.

[Data transmission and reception]

- Data length of 7 or 8 bits
- Phase control of transmit and receive data
- MSB- or LSB-first selectable

[Clock control]

- Master or slave selection
- Phase control of I/O clock
- Setting of transfer period by prescaler and internal counter of each channel
- Maximum transfer rate*1
 - During master communication:
 - Max. PCLKB/2 (SPI00 only)

表 19.6 串行通道使能状态寄存器m(SEm) 在通道停止操作时的状态

少量	象征	读取值	功能
n	SE[n]	1 or 0	指示通道 n 的操作是否已启用或已停止。 对于停止运行的通道，可以通过软件设置 SOm 寄存器的 CKO[n] 位的值。 SEm 是一个只读状态寄存器，可以通过使用 STm 寄存器来停止其操作。 0: 操作停止 1: 操作已启用。

(c) 串行输出使能寄存器 m (SOEm)

该寄存器用于启用或停止每个通道的串行通信操作的输出。

表 19.7 停止操作时，按通道设置串行输出使能寄存器m(SOEm)

少量	象征	设置值	功能
n	SOE[n]	0	通道 n 的串行输出启用或停止 对于串行输出停止的通道 n，可以通过软件设置 SOm 寄存器的 SO[n] 位值。 0: 停止串行通信操作的输出

(d) 串行输出寄存器 m(SOM)

SOm是每个通道串行输出的缓冲寄存器。

表 19.8 串行输出寄存器m (SOM) 的设置（通过通道进行操作时）

少量	象征	设置值	功能
n	所以[n]	1	通道 n 的串行数据输出 当使用与每个通道对应的引脚作为端口功能引脚时，将相应的 SO[n] 位设置为 1。 1: 串口数据输出值为 1
n+8	CKO[n]	1	通道 n 的串行时钟输出 当使用与每个通道对应的引脚作为端口功能引脚时，将相应的 CKO[n] 位设置为 1。 1: 串行时钟输出值为 1

注：单元编号(m = 0) n: 通道编号(n = 0至3)

19.5 简化SPI的操作

这是一个时钟通信功能，使用三条线：串行时钟线（SCK）和串行数据线（SI 和 SO）。

[数据传输和接收]

- 数据长度为7或8位
- 发射和接收数据的相位控制
- MSB或LSB优先可选

[时钟控制]

- 主人或奴隶的选择
- I/O时钟的相位控制
- 通过预分频器和各通道的内部计数器设置传输周期
- 最高转账费率*1
 - 在主控沟通期间:
 - 最大值 PCLKB/2（仅限 SPI00）

- Max. PCLKB/4
- During slave communication:
 - Max. $f_{MCK}/6$

[Interrupt function]

- Transfer end interrupt or buffer empty interrupt (SAU0_SPI_TXRXI00/SAU0_SPI_TXRXI11)

[Error detection flag]

- Overrun error

In addition, simplified SPIs of following channels support the Snooze mode. In the Snooze mode, data can be received without CPU processing upon detecting SCK input in the Software Standby mode. The Snooze mode is only available in SPI00, which support asynchronous reception.

Note 1. Set up the transfer rate within a range satisfying the SCK cycle time (tKCY). For details, see [section 27, Electrical Characteristics](#).

Note: Use a general-purpose port pin to send a chip select signal when required.

The channels supporting simplified SPI are channels 0 and 3 of SAU0. See [Table 19.1](#).

Simplified SPI performs the following seven types of communication operations.

- Master transmission (See [section 19.5.1. Master Transmission.](#))
- Master reception (See [section 19.5.2. Master Reception.](#))
- Master transmission and reception (See [section 19.5.3. Master Transmission and Reception.](#))
- Slave transmission (See [section 19.5.4. Slave Transmission.](#))
- Slave reception (See [section 19.5.5. Slave Reception.](#))
- Slave transmission and reception (See [section 19.5.6. Slave Transmission and Reception.](#))
- Snooze mode function (See [section 19.5.7. Snooze Mode Function.](#))

19.5.1 Master Transmission

Master transmission is when a microcontroller outputs a transfer clock and transmits data to another device.

[Table 19.9](#) shows the specification of master transmission of Simplified SPI.

Table 19.9 Specification of master transmission of simplified SPI (1 of 2)

Simplified SPI	SPI00	SPI11
Target channel	Channel 0 of SAU0	Channel 3 of SAU0
Pins used	SCK00, SO00	SCK11, SO11
Interrupt	SAU0_SPI_TXRXI00	SAU0_SPI_TXRXI11
	Transfer end interrupt (in single-transfer mode) or buffer empty interrupt (in continuous transfer mode) can be selected.	
Error detection flag	None	
Transfer data length	7 or 8 bits	
Transfer rate *1	Max. PCLKB/2 [Hz] (SPI00 only), PCLKB/4 [Hz] Min. PCLKB/(2 × 2 ¹⁵ × 128) [Hz] PCLKB: System clock frequency	
Data phase	Selectable by the DCP[1] bit of the SCRmn register • DCP[1] = 0: Data output starts from the start of the operation of the serial clock. • DCP[1] = 1: Data output starts half a clock cycle before the start of the serial clock operation.	
Clock phase	Selectable by the DCP[0] bit of the SCRmn register • DCP[0] = 0: Non-reverse • DCP[0] = 1: Reverse	

- 最大值 PCLKB/4
- 在奴隶交流期间:
 - 最大值 fMCK/6

[中断功能]

- 传输结束中断或缓冲区空中断（SAU0_SPI_TXRXI00/SAU0_SPI_TXRXI11）

[错误检测标志]

- 超限错误

此外，以下通道的简化版 SPI 支持休眠模式。在休眠模式下，当软件待机模式下检测到 SCK 输入时，无需 CPU 处理即可接收数据。休眠模式仅适用于支持异步接收的 SPI00。

注1. 将传输速率设置在满足SCK周期时间(tKCY)的范围内。详情请参见第27节“电气特性”。

注意：必要时，可以使用通用端口引脚发送片选信号。

支持简化 SPI 的通道是 SAU0 的通道 0 和 3。参见表 19.1。

简化的 SPI 执行以下七种类型的通信操作。

- 主变速器（参见第19.5.1节主变速器。）
- 主接待处（参见第19.5.2节主接待处。）
- 主控发射和接收（参见第19.5.3节主控发射和接收。）
- 从属变速器（参见第19.5.4节从属变速器。）
- 奴隶接收（参见第19.5.5节奴隶接收。）
- 从属设备的传输和接收（参见第19.5.6节从属设备的传输和接收。）
- 贪睡模式功能（参见19.5.7节贪睡模式功能。）

19.5.1主变速器

主传输是指微控制器输出传输时钟并将数据传输到另一个设备。

表 19.9 显示了简化 SPI 的主传输规范。

表 19.9 简化 SPI 主传输规范（1/2）

简化的SPI	SPI00	SPI11
目标频道	SAU0 的 0 通道	SAU0 的 3 频道
使用的引脚	SCK00, SO00	SCK11, SO11
打断	SAU0_SPI_TXRXI00	SAU0_SPI_TXRXI11
	可以选择传输结束中断（单次传输模式）或缓冲区空中断（连续传输模式）。	
错误检测标志	None	
传输数据长度	7 位或 8 位	
转账费率*1	最大值 PCLKB/2[Hz]（仅限 SPI00），PCLKB/4[Hz] 最小 PCLKB/(2 × 2 ¹⁵ × 128) [Hz] PCLKB: 系统时钟频率	
数据阶段	可通过 SCRmn 寄存器的 DCP [1] 位进行选择 • DCP[1] = 0 : 数据输出从串行时钟开始运行时开始。 • DCP[1] = 1: 数据输出在串行时钟操作开始前半时钟周期开始。	
时钟相位	可通过 SCRmn 寄存器的 DCP[0] 位进行选择 • DCP[0] = 0 : 非反向 • DCP[0] = 1: 反向	

Table 19.9 Specification of master transmission of simplified SPI (2 of 2)

Simplified SPI	SPI00	SPI11
Data direction	MSB or LSB first	

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3)

Note 1. Use this operation within a range that satisfies the conditions above and the peripheral functions characteristics specified in the electrical characteristics. For details, see [section 27, Electrical Characteristics](#).

(1) Register setting

Table 19.10 to Table 19.15 show examples of the register contents for master transmission of simplified SPI.

(a) Serial mode register mn (SMRmn)

Table 19.10 Example of serial mode register mn (SMRmn) contents for master transmission of simplified SPI

Bit	Symbol	Set value	Function
0	MD0	0/1	Interrupt source of channel n 0: Transfer end interrupt 1: Buffer empty interrupt
2:1	MD1[1:0]	00b	Setting of operation mode of channel n 0 0: Simplified SPI mode
5:3	—	100b	Setting disabled (set to the initial value)
6	SIS0	0	Setting is fixed in the simplified SPI mode
7	—	0	Setting disabled (set to the initial value)
8	STS	0	Selection of start trigger source 0: Only software trigger is valid (selected for simplified SPI, UART transmission, and simplified I ² C).
13:9	—	00000b	Setting disabled (set to the initial value)
14	CCS	0	Selection of transfer clock (f _{TCLK}) of channel n 0: Divided operation clock f _{MCK} specified by the CKS bit
15	CKS	0/1	Operation clock (f _{MCK}) of channel n 0: Prescaler output clock CKm0 set by the SPSm register 1: Prescaler output clock CKm1 set by the SPSm register

(b) Serial communication operation setting register mn (SCRmn)

Table 19.11 Example of serial communication operation setting register mn (SCRmn) contents for master transmission of simplified SPI (1 of 2)

Bit	Symbol	Set Value	Function
1:0	DLS[1:0]	10b or 11b	Setting of data length 1 0: 7-bit data length 1 1: 8-bit data length
3:2	—	01b	Setting disabled (set to the initial value)
5:4	SLC[1:0]	00b	Since this bit is dedicated to UART mode, it is fixed in the simplified SPI mode.
6	—	0	Setting disabled (set to the initial value)
7	DIR	0/1	Selection of data transfer sequence in simplified SPI and UART modes 0: Input or output data with MSB first 1: Input or output data with LSB first
9:8	PTC[1:0]	00b	Since this bit is dedicated to UART mode, it is fixed in the simplified SPI mode.

表 19.9 简化 SPI 主传输规范 (2/2)

简化的SPI	SPI00	SPI11
数据方向	先 MSB 还是 LSB	

注: m: 单元编号(m = 0) n: 通道编号(n = 0 至3)

注1: 此操作应在满足上述条件和电气特性中规定的外围功能特性的范围内使用。详情请参见第27节“电气特性”。

(1) 寄存器设置

表 19.10 至表 19.15 显示了简化 SPI 主传输的寄存器内容示例。

(a) 串行模式寄存器 mn (SMRmn)

表 19.10 简化主传输的串行模式寄存器 mn (SMRmn) 内容示例
SPI

少量	象征	设置值	功能
0	MD0	0/1	通道 n 的中断源 0: 传输结束中断 1: 缓冲区为空中断
2:1	MD1[1:0]	00b	设置通道 n 的工作模式 0 0: 简化的 SPI 模式
5:3	—	100b	设置已禁用 (已设置为初始值)
6	SIS0	0	简化 SPI 模式下的设置是固定的
7	—	0	设置已禁用 (已设置为初始值)
8	STS	0	选择启动触发源 0: 仅软件触发有效 (选择用于简化 SPI、UART 传输和简化I ² C)。
13:9	—	00000b	设置已禁用 (已设置为初始值)
14	CCS	0	选择通道 n 的传输时钟 (f _{TCLK}) 0: 由 CKS 位指定的分频操作时钟 f _{MCK}
15	CKS	0/1	通道 n 的运行时钟(f _{MCK}) 0: 预分频器输出时钟 CKm0 由 SPSm 寄存器设置 1: 预分频器输出时钟 CKm1 由 SPSm 寄存器设置

(b) 串行通信操作设置寄存器 mn (SCRmn)

表 19.11 简化 SPI 主传输的串行通信操作示例, 设置寄存器 mn (SCRmn) 的内容 (1/2)

少量	象征	设置值	功能
1:0	DLS[1:0]	10b 或 11b	数据长度设置: 1.0: 7 位数 据长度; 1.1: 8 位数数据长度
3:2	—	01b	设置已禁用 (已设置为初始值)
5:4	SLC[1:0]	00b	由于该位专用于 UART 模式, 因此在简化的 SPI 模式下它是固定的。
6	—	0	设置已禁用 (已设置为初始值)
7	DIR	0/1	简化 SPI 和 UART 模式下数据传输顺序的选择 0: 输入或输出数据, 最高有效位在前。 1: 输入或输出数据时, 最低有效位 (LSB) 优先。
9:8	PTC[1:0]	00b	由于该位专用于 UART 模式, 因此在简化的 SPI 模式下它是固定的。

Table 19.11 Example of serial communication operation setting register mn (SCRmn) contents for master transmission of simplified SPI (2 of 2)

Bit	Symbol	Set Value	Function
10	EOC	0	Since this bit is dedicated to UART receive modes, it is fixed in the simplified SPI mode.
11	—	0	Setting disabled (set to the initial value)
13:12	DCP[1:0]	00b to 11b	Selection of data and clock phase in simplified SPI mode For details about the setting, see section 19.3. Register Descriptions .
15:14	TRXE[1:0]	10b	Setting TRXE[1:0] = 10b is fixed in the simplified SPI master transmission mode

(c) Serial data register mn (SDRmn)

Table 19.12 Example of serial data register mn (SDRmn) contents for master transmission of simplified SPI

Bit	Symbol	Set value	Function
7:0	DAT[7:0]	0x00 to 0xFF	Transmit data (Transmit data setting)
8	DAT[8]	0	0 Fixed
15:9	STCLK[6:0]	0x00 to 0x7F	Baud rate setting (Operation clock (f _{MCK}) division setting)

(d) Serial output register m (SOM)

Set only the bit of the target channel.

Table 19.13 Example of serial output register m (SOM) contents for master transmission of simplified SPI

Bit	Symbol	Set value	Function
n	SO[n]	0/1	Serial data output of channel n 0: Serial data output value is 0 1: Serial data output value is 1
n+8	CKO[n]	0/1	Communication starts when a bit is 1 if the clock phase is non reverse (the SCRmn.DCP[0] = 0). If the clock phase is reversed (SCRmn.DCP[0] = 1), communication starts when a bit is 0

(e) Serial output enable register m (SOEm)

Set only the bit of the target channel to 1.

Table 19.14 Example of serial output enable register m (SOEm) contents for master transmission of simplified SPI

Bit	Symbol	Set value	Function
n	SOE[n]	1	Serial output enable or stop of channel n 1: Enable output by serial communication operation.

(f) Serial channel start register m (SSm)

Set only the bit of the target channel to 1.

Table 19.15 Example of serial channel start register m (SSm) contents for master transmission of simplified SPI

Bit	Symbol	Set value	Function
n	SS[n]	1	Operation start trigger of channel n 1: Set the SEm.SE[n] bit to 1 to place the channel in communication waiting state

表 19.11 简化 SPI 主传输的串行通信操作示例，设置寄存器 mn (SCRmn) 的内容 (2/2)

少量	象征	设置值	功能
10	EOC	0	由于该位专用于 UART 接收模式，因此在简化的 SPI 模式下它是固定的。
11	—	0	设置已禁用（已设置为初始值）
13:12	DCP[1:0]	00b 至 11b	简化 SPI 模式下的数据和时钟相位选择。有关设置的详细信息，请参阅第 19.3 节“寄存器描述”。
15:14	TRXE[1:0]	10b	在简化的 SPI 主传输模式下，设置TRXE[1:0] = 10b是固定的。

(c) 串行数据寄存器 mn (SDRmn)

表 19.12 简化主传输的串行数据寄存器 mn (SDRmn) 内容示例

少量	象征	设置值	功能
7:0	DAT[7:0]	0x00 到 0xFF	传输数据 (传输数据设置)
8	DAT[8]	0	0 固定
15:9	STCLK[6:0]	0x00 至 0x7F	波特率设置 (操作时钟(f _{MCK}) 分频设置)

(d) 串行输出寄存器 m (SOM)

仅设置目标通道的位。

表 19.13 简化主传输的串行输出寄存器m (SOM) 内容示例

少量	象征	设置值	功能
n	所以[n]	0/1	通道 n 的串行数据输出 0: 串口数据输出值为 0 1: 串口数据输出值为 1
n+8	CKO[n]	0/1	当时钟相位未反转（SCRmn.DCP[0]= 0) 时，通信在某个位为 1 时开始。如果时钟相位反转（SCRmn.DCP[0] = 1），则当某个位为 0 时通信开始。

(e) 串行输出使能寄存器 m (SOEm)

仅将目标通道的位设置为 1。

表 19.14 简化主传输的串行输出使能寄存器m(SOEm) 内容示例

少量	象征	设置值	功能
n	SOE[n]	1	通道 n 的串行输出启用或停止 1: 通过串口通信操作启用输出。

(f) 串行通道起始寄存器m (SSm)

仅将目标通道的位设置为 1。

表 19.15 简化主传输的串行通道起始寄存器m (SSm) 内容示例

少量	象征	设置值	功能
n	SS[n]	1	通道 n 的操作启动触发 1: 将 SEm.SE[n] 位设置为 1，使通道处于通信等待状态

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3), mn = 00 to 03

Note: 0/1: Set to 0 or 1 depending on the usage of the user

(2) Operation procedure

Table 19.16 shows the procedure for initial setting of master transmission.

Table 19.16 Initial setting procedure for master transmission

Step		Process	Detail
Procedure for initial setting of master transmission	<1>	Starting initial setting	—
	<2>	Setting the SPSm register	Set the operation clock.
	<3>	Setting the SMRmn register	Set an operation mode.
	<4>	Setting the SCRmn register	Set a communication format.
	<5>	Setting the SDRmn register	Set a transfer baud rate (setting the transfer clock by dividing the operation clock (f_{MCK})).
	<6>	Setting the SOm register	Set the initial output level of the serial clock (SOm.CKO[n]) and serial data (SOm.SO[n]).
	<7>	Setting of the SOEm register	Set the SOEm.SOE[n] bit to 1 and enable data output of the target channel.
	<8>	Setting port	Set a Peripheral Select register, a Pmn Output Data Register and a Pmn Direction Register (enable data output and clock output of the target channel)
	<9>	Writing to the SSm register	Set the SSm.SS[n] bit of the target channel to 1 and set SEm.SE[n] bit = 1 to enable operation.
	<10>	Completing initial setting	Setting of SAU is completed. Write transmit data to the SDRmn.DAT[7:0] bits and start communication.

Table 19.17 shows the procedure for stopping master transmission.

Table 19.17 Procedure for stopping master transmission

Step		Process	Detail
Procedure for stopping master transmission	<1>	Starting setting to stop	—
	<2>	Wait until SSRmn.TSF is cleared (optional)	If there is any data being transferred, wait for their completion. If there is an urgent requirement to stop, do not wait.
	<3>	Writing the STm register	Write 1 to the STm.ST[n] bit of the target channel (stopping operation by setting SEm.SE[n] = 0).
	<4>	Changing setting of the SOEm register	Set the SOEm.SOE[n] bit to 0 and stop the output of the target channel.
	<5>	Changing setting of the SOm register (optional)	The levels of the serial clock (SOm.CKO[n]) and serial data (SOm.SO[n]) on the target channel can be changed if required.
	<6>	Stop setting is completed	The master transmission is stopped. Go to the next processing.

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3)

Table 19.18 shows the procedure for resuming master transmission.

注: m: 单元编号(m = 0) n: 通道编号(n = 0至3) , mn = 00至03

注意: 0/1: 根据用户使用情况设置为 0 或 1

(2) 操作流程

表 19.16 显示了主传输的初始设置程序。

表 19.16 主传输初始设置程序

步		过程	细节
主变速器初始设置程序	<1>	初始设置	—
	<2>	设置 SPSm 寄存器	设置运行时钟。
	<3>	设置 SMRmn 寄存器	设置操作模式。
	<4>	设置 SCRmn 寄存器	制定沟通模式。
	<5>	设置 SDRmn 寄存器	设置传输波特率（通过分频操作时钟来设置传输时钟(f_{MCK})）。
	<6>	设置 SOm 寄存器	设置串行时钟(SOm.CKO[n])和串行数据(SOm.SO[n])的初始输出电平。
	<7>	设置 SOEm 寄存器	将 SOEm.SOE[n] 位设置为 1，使能目标通道的数据输出。
	<8>	设置端口	设置外设选择寄存器、Pmn输出数据寄存器和Pmn 方向寄存器（使能目标通道的数据输出和时钟输出）
	<9>	致函SSm登记处	将目标通道的 SSm.SS[n] 位设置为 1，并将 SEm.SE[n] 位= 1 设置为启用操作。
	<10>	完成初始设置	SAU设置已完成。 将发送数据写入 SDRmn.DAT[7:0] 位并开始通信。

表 19.17 显示了停止主传输的步骤。

表 19.17 停止主传输的步骤

步		过程	细节
程序 停止主变速器	<1>	开始设置停止	—
	<2>	等待 SSRmn.TSF 被清除（可选）	如果有数据传输，请等待传输完成。如果有紧急需要停止，不要等待。
	<3>	写入 STm 寄存器	将 1 写入目标通道的 STm.ST[n] 位（通过设置 SEm.SE[n] = 0来停止操作）。
	<4>	更改 SOEm 寄存器的设置	将 SOEm.SOE[n] 位设置为 0，停止目标通道的输出。
	<5>	更改 SOm 寄存器的设置（可选）	如果需要，可以更改目标通道上的串行时钟(SOm.CKO[n]) 和串行数据(SOm.SO[n])的电平。
	<6>	停止设置完成	主变速器已停止。 进入下一处理阶段。

注: m: 单元编号(m = 0) n: 通道编号(n = 0 至3)

表 19.18 显示了恢复主传输的步骤。

Table 19.18 Procedure for resuming master transmission

Step		Process	Detail
Procedure for resuming master transmission	<1>	Starting setting for resumption	—
	<2>	Wait until Slave is ready	Wait until stop the communication target (slave) or communication operation completed.
	<3>	Port manipulation	Disable data output and clock output of the target channel.
	<4>	Changing setting of the SPSm register (optional)	Reset the register to change the operation clock setting.
	<5>	Changing setting of the SDRmn register (optional)	Reset the register to change the transfer baud rate setting (setting the transfer clock by dividing the operation clock (f _{MCK})).
	<6>	Changing setting of the SMRmn register (optional)	Reset the register to change serial mode register mn (SMRmn) setting.
	<7>	Changing setting of the SCRmn register (optional)	Reset the register to change serial communication operation setting register mn (SCRmn) setting.
	<8>	Changing setting of the SOEm register (optional)	Set the SOEm.SOE[n] bit to 0 to stop output from the target channel.
	<9>	Changing setting of the SOm register (optional)	Set the initial output level of the serial clock (SOm.CKO[n]) and serial data (SOm.SO[n]).
	<10>	Changing setting of the SOEm register	Set the SOEm.SOE[n] bit to 1 and enable output from the target channel.
	<11>	Port manipulation	Enable data output and clock output of the target channel.
	<12>	Writing to the SSm register	Set the SSm.SS[n] bit of the target channel to 1 and set SEm.SE[n] = 1 to enable operation.
	<13>	Completing resumption setting	Setting is completed. Set transmit data to the SDRmn.DAT[7:0] bits and start communication.

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3)

(3) Processing flow (in single transmission mode)

Figure 19.4 shows the timing of master transmission (in single transmission mode) (Type 1: SCRmn.DCP[1:0] = 00b).

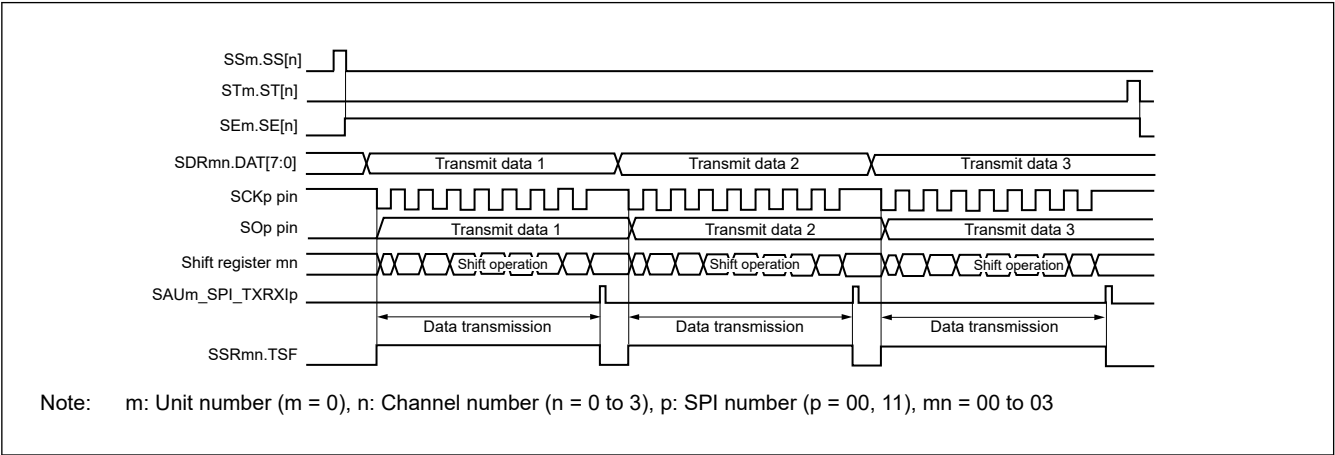


Figure 19.4 Timing of master transmission (in single transmission mode) (type 1: SCRmn.DCP[1:0] = 00b)

Figure 19.5 shows the flowchart of master transmission (in single transmission mode).

表 19.18 恢复主传输的步骤

步		过程	细节
恢复主传输的步骤	<1>	恢复的初始设置	—
	<2>	等待奴隶准备就绪	等待通信目标（从设备）停止或通信操作完成。
	<3>	端口操作	禁用目标通道的数据输出和时钟输出。
	<4>	更改 SPSm 寄存器的设置（可选）	重置寄存器以更改操作时钟设置。
	<5>	更改 SDRmn 寄存器的设置（可选）	重置寄存器以更改传输波特率设置（通过分频操作时钟 (f _{MCK}) 来设置传输时钟）。
	<6>	更改 SMRmn 寄存器的设置（可选）	重置寄存器以更改串行模式寄存器 mn (SMRmn) 设置。
	<7>	更改 SCRmn 寄存器的设置（可选）	重置寄存器以更改串行通信操作设置寄存器 mn (SCRmn) 设置。
	<8>	更改 SOEm 寄存器的设置（可选）	将 SOEm.SOE[n] 位设置为 0 以停止目标通道的输出。
	<9>	更改 SOm 寄存器的设置（可选）	设置串行时钟 (SOm.CKO[n]) 和串行数据(SOm.SO[n])的初始输出电平。
	<10>	更改 SOEm 寄存器的设置	将 SOEm.SOE[n] 位设置为 1，并启用目标通道的输出。
	<11>	端口操作	启用目标通道的数据输出和时钟输出。
	<12>	致函SSm登记处	将目标通道的 SSm.SS[n] 位设置为 1，并将SEm.SE[n] = 1 设置为启用操作。
	<13>	完成恢复设置	设置完成。 将发送数据设置为 SDRmn.DAT[7:0] 位并开始通信。

注: m: 单元编号(m = 0) n: 通道编号(n = 0 至3)

(3) 处理流程（单次传输模式）

图 19.4 显示了主传输的时序（单传输模式）（类型 1: SCRmn. DCP[1:0] = 00b）。

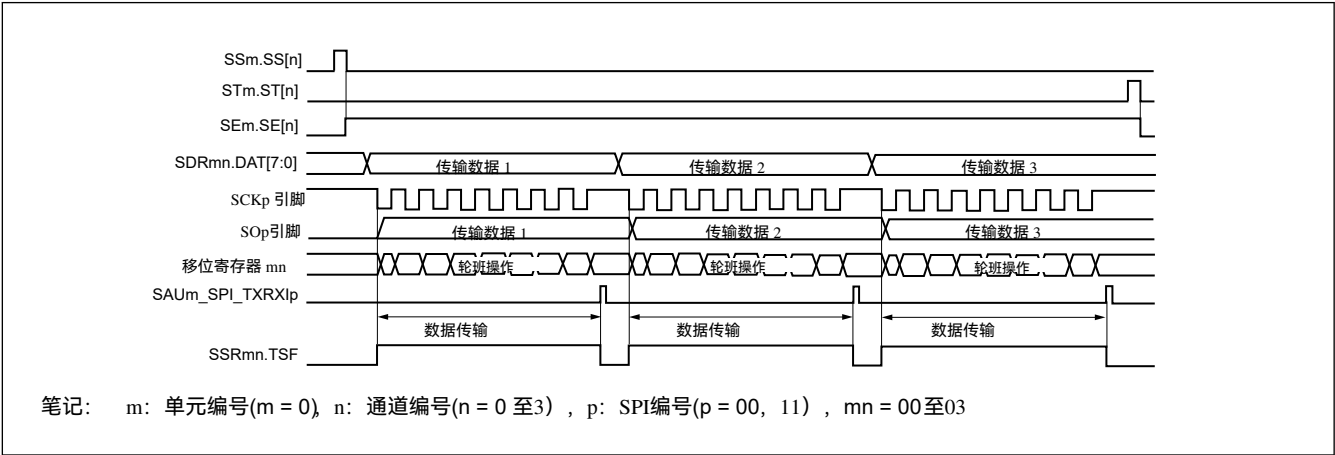


图 19.4 主传输时序（单传输模式）（类型 1: SCRmn.DCP[1:0] = 00b)

图 19.5 显示了主传输（单传输模式）的流程图。

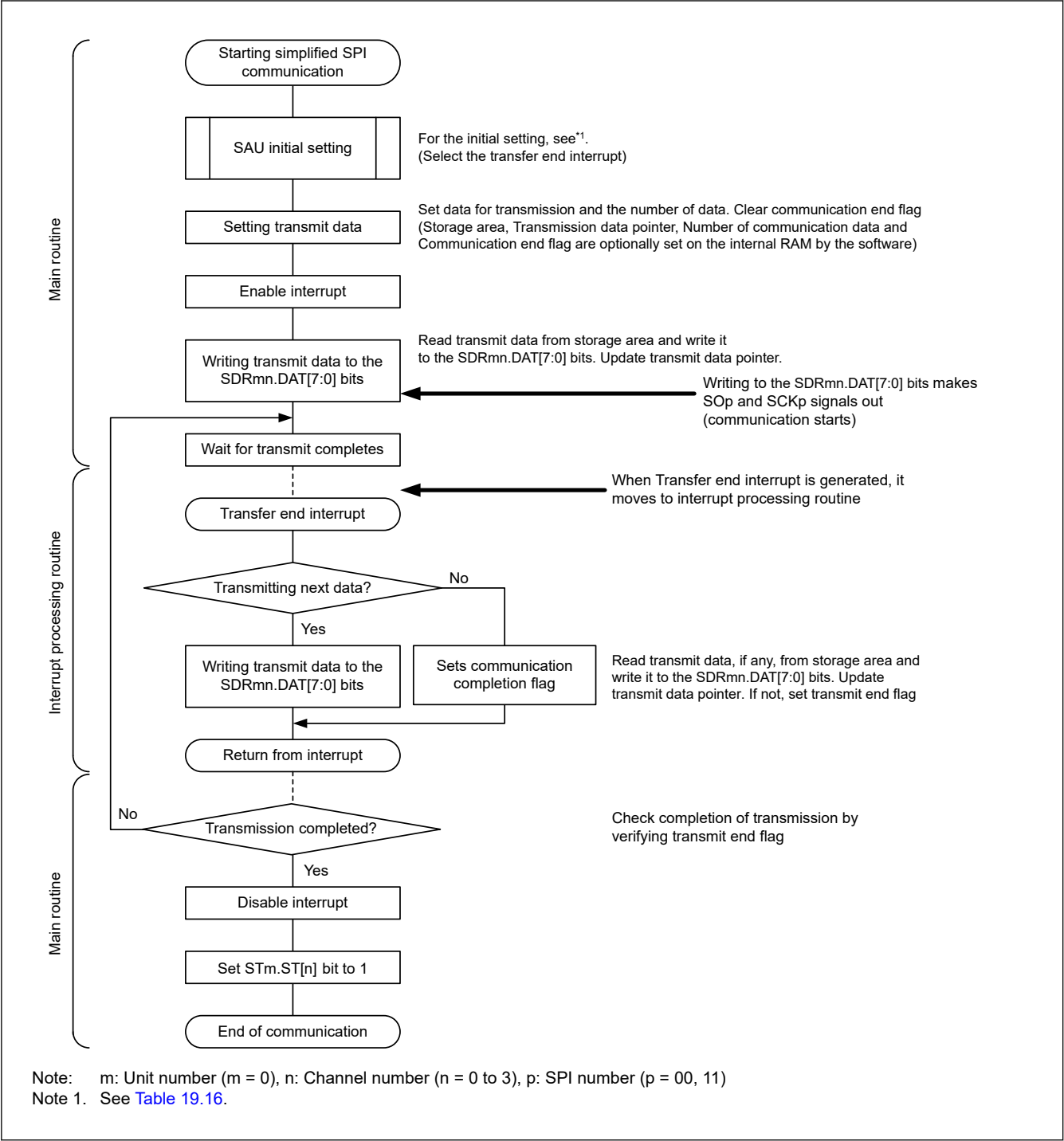


Figure 19.5 Flowchart of master transmission (in single transmission mode)

(4) Processing flow (in continuous transmission mode)

Figure 19.6 shows the timing of master transmission (in continuous transmission mode) (Type 1: SCRmn.DCP[1:0] = 00b).

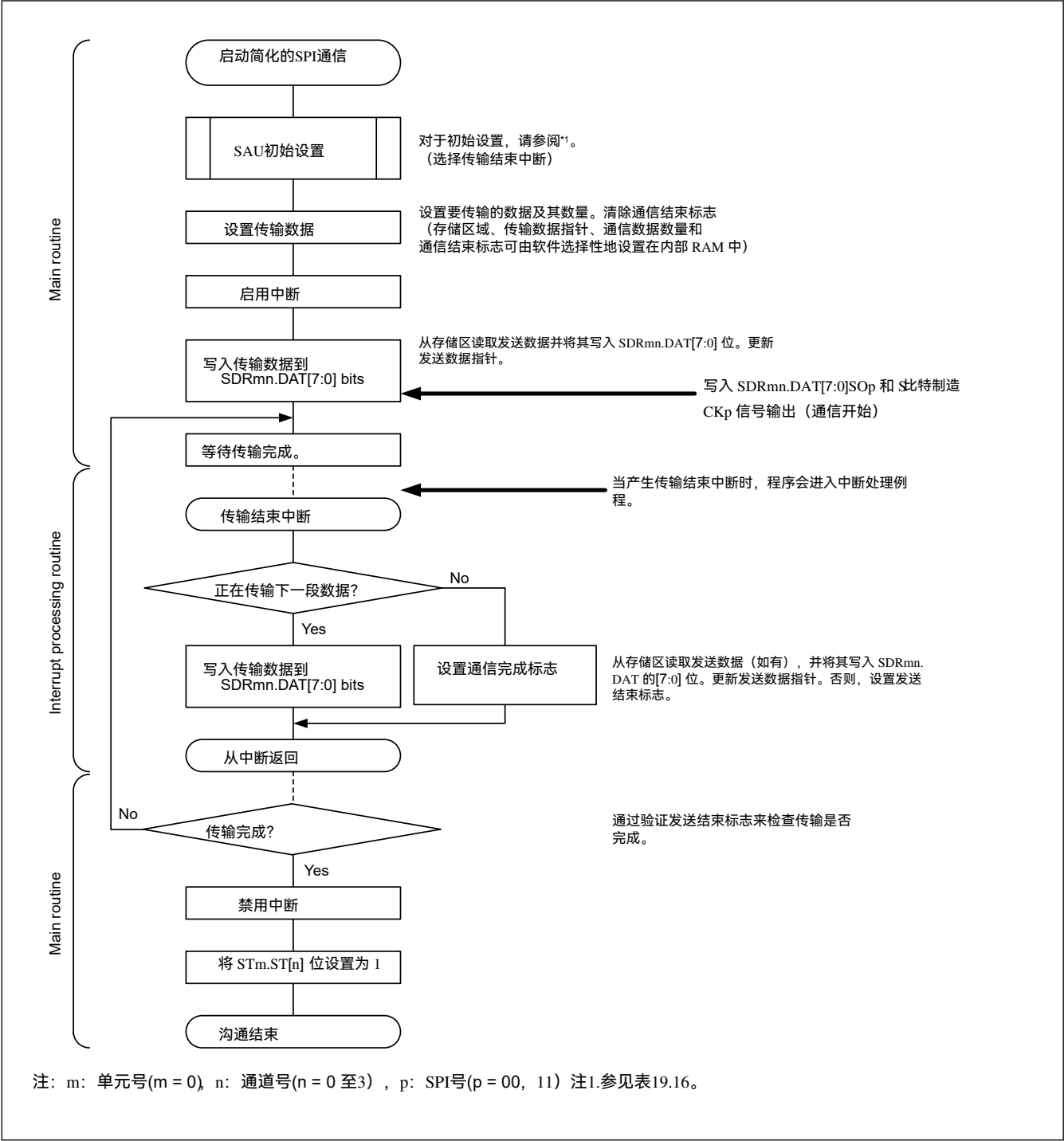


图 19.5 主传输流程图（单传输模式）

(4) 处理流程（连续传输模式）

图 19.6 显示了主传输的时序（在连续传输模式下）（类型 1: SCRmn.DCP[1:0] = 00b）。

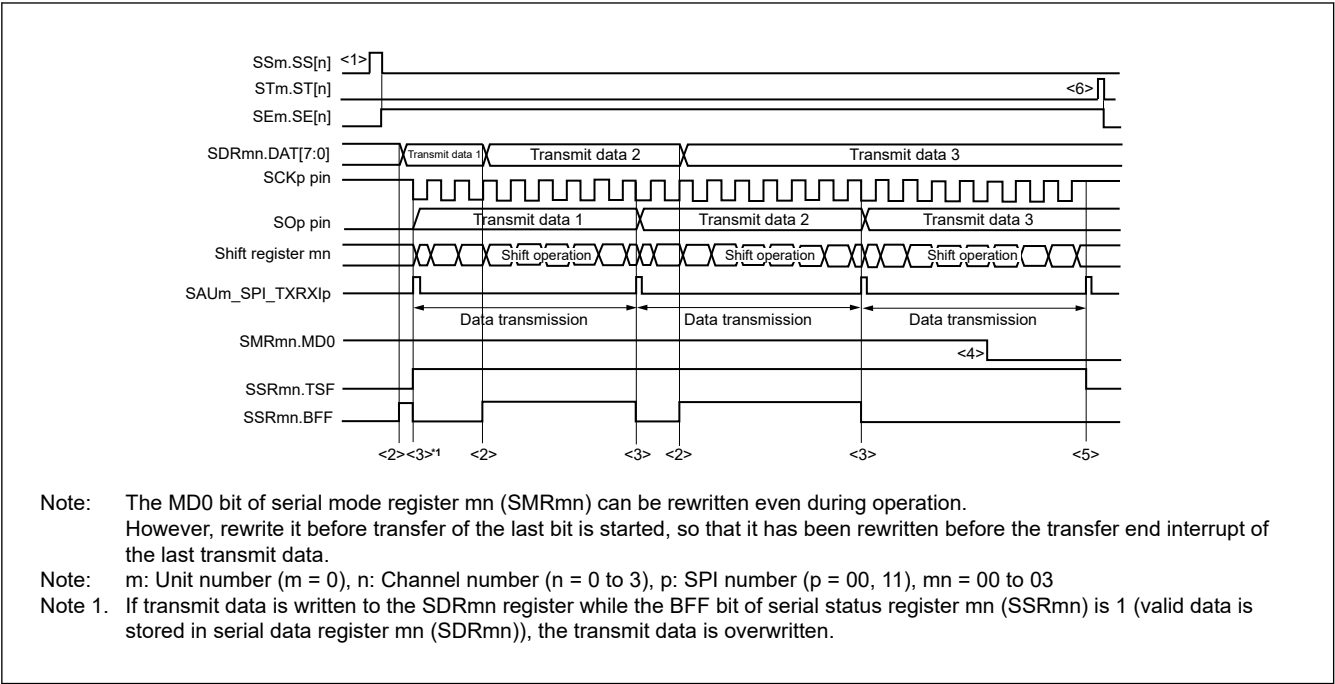


Figure 19.6 Timing of master transmission (in continuous transmission mode) (type 1: SCRmn.DCP[1:0] = 00b)

Figure 19.7 shows the flowchart of master transmission (in continuous transmission mode).

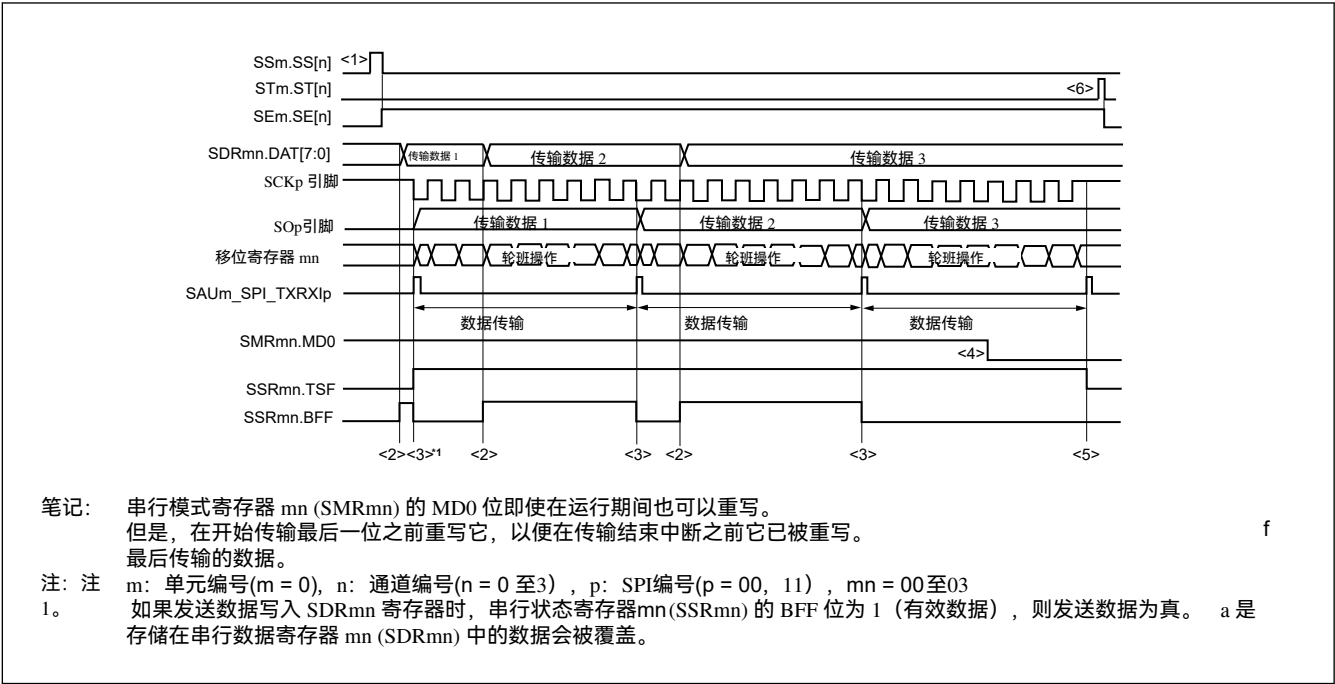


图 19.6 主传输时序 (连续传输模式) (类型 1: SCRmn.DCP[1:0] = 00b)

图 19.7 显示了主传输流程图 (在连续传输模式下)。

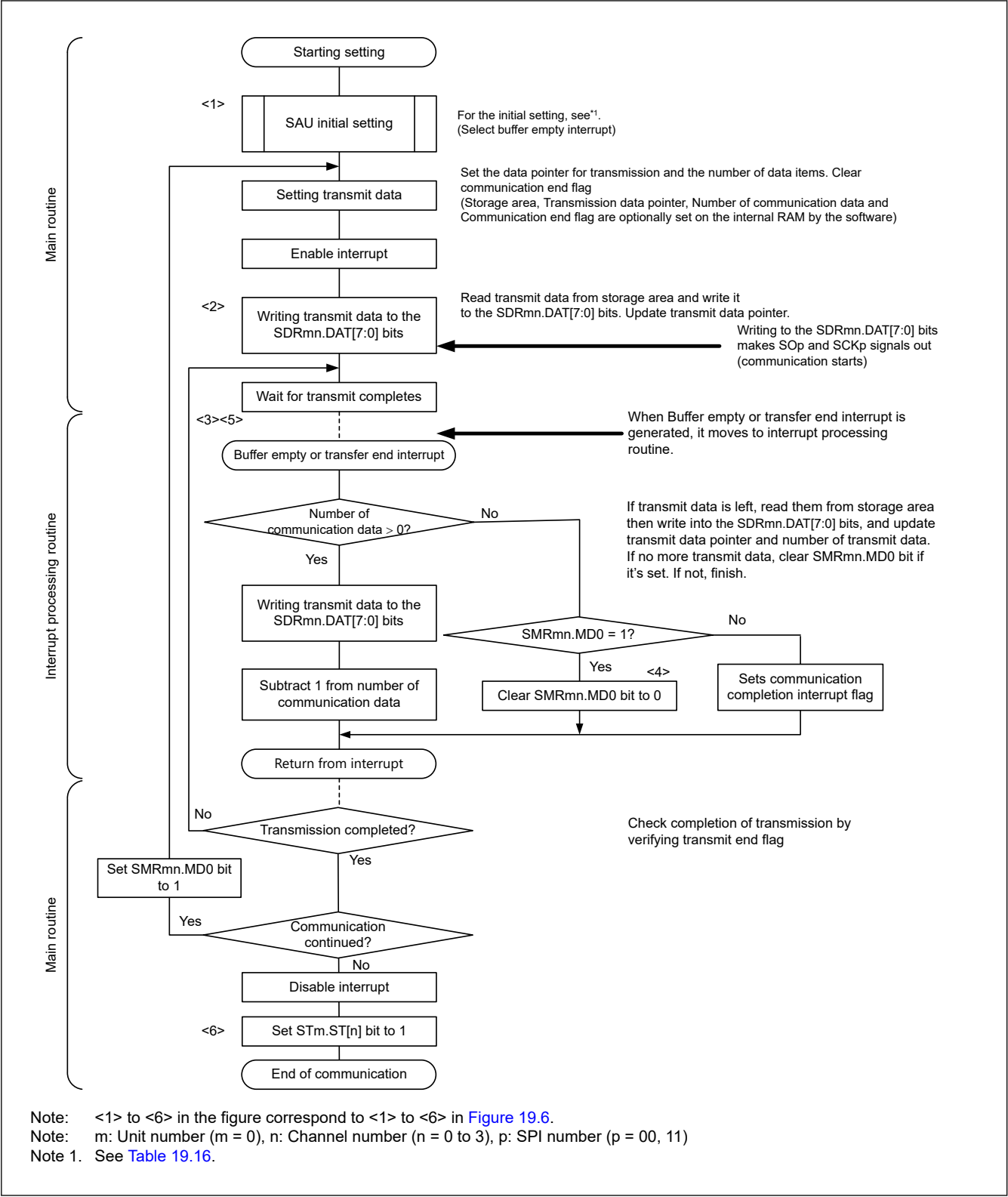


Figure 19.7 Flowchart of master transmission (in continuous transmission mode)

19.5.2 Master Reception

Master reception is when a microcontroller outputs a transfer clock and receives data from another device.

Table 19.19 shows the specification of master reception of Simplified SPI.

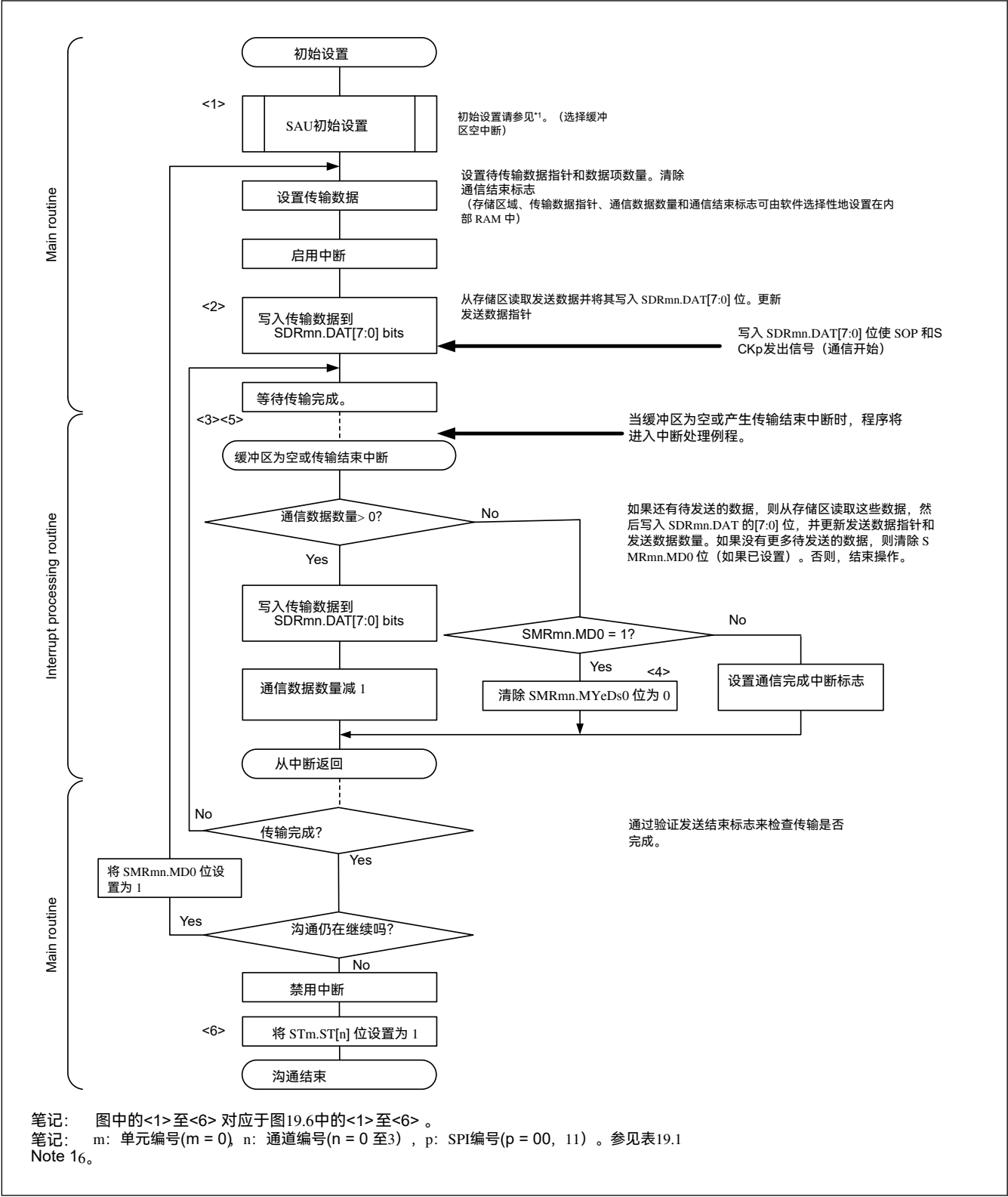


图 19.7 主传输流程图（连续传输模式）

19.5.2 主接待处

主接收是指微控制器输出传输时钟信号并从另一个设备接收数据。

表 19.19 显示了简化 SPI 的主接收规范。

Table 19.19 Specification of master reception of simplified SPI

Simplified SPI	SPI00	SPI11
Target channel	Channel 0 of SAU0	Channel 3 of SAU0
Pins used	SCK00, SI00	SCK11, SI11
Interrupt	SAU0_SPI_TXRXI00	SAU0_SPI_TXRXI11
	Transfer end interrupt (in single-transfer mode) or buffer empty interrupt (in continuous transfer mode) can be selected.	
Error detection flag	Overrun error detection flag (SSRmn.OVF) only	
Transfer data length	7 or 8 bits	
Transfer rate*1	Max. PCLKB/2 [Hz] (SPI00 only), PCLKB/4 [Hz] Min. PCLKB/(2 × 2 ¹⁵ × 128)[Hz]	
Data phase	Selectable by the DCP[1] bit of the SCRmn register - DCP[1] = 0: Data input starts from the start of the operation of the serial clock. - DCP[1] = 1: Data input starts half a clock cycle before the start of the serial clock operation.	
Clock phase	Selectable by the DCP[0] bit of the SCRmn register - DCP[0] = 0: Non-reverse - DCP[0] = 1: Reverse	
Data direction	MSB or LSB first	

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3), mn = 00 to 03

Note 1. See [section 27, Electrical Characteristics](#).

(1) Register setting

[Table 19.20](#) to [Table 19.25](#) show examples of the register contents for master reception of Simplified SPI.

(a) Serial mode register mn (SMRmn)

Table 19.20 Example of serial mode register mn (SMRmn) contents for master reception of simplified SPI

Bit	Symbol	Set value	Function
0	MD0	0/1	Interrupt source of channel n 0: Transfer end interrupt 1: Buffer empty interrupt
2:1	MD1[1:0]	00b	Setting of operation mode of channel n 0 0: Simplified SPI mode
5:3	—	100b	Setting disabled (set to the initial value)
6	SIS0	0	Setting is fixed in the simplified SPI mode
7	—	0	Setting disabled (set to the initial value)
8	STS	0	Selection of start trigger source 0: Only software trigger is valid (selected for simplified SPI, UART transmission, and simplified I ² C).
13:9	—	00000b	Setting disabled (set to the initial value)
14	CCS	0	Selection of transfer clock (f _{TCLK}) of channel n 0: Divided operation clock f _{MCK} specified by the CKS bit
15	CKS	0/1	Operation clock (f _{MCK}) of channel n 0: Prescaler output clock CKm0 set by the SPSm register 1: Prescaler output clock CKm1 set by the SPSm register

表 19.19 简化 SPI 主接收规范

简化的SPI	SPI00	SPI11
目标频道	SAU0 的 0 通道	SAU0 的 3 频道
使用的引脚	SCK00, SI00	SCK11, SI11
打断	SAU0_SPI_TXRXI00	SAU0_SPI_TXRXI11
	可以选择传输结束中断（单次传输模式）或缓冲区空中断（连续传输模式）。	
错误检测标志	仅溢出错误检测标志 (SSRmn.OVF)	
传输数据长度	7 位或 8 位	
转账费率*1	最大值 PCLKB/2[Hz]（仅限 SPI00），PCLKB/4[Hz] 最小 PCLKB/(2 × 2 ¹⁵ × 128)[Hz]	
数据阶段	可通过 SCRmn 寄存器的 DCP [1] 位进行选择 - DCP[1] = 0: 数据输入从串行时钟开始运行。 - DCP[1] = 1: 数据输入在串行时钟操作开始前半时钟周期开始。	
时钟相位	可通过 SCRmn 寄存器的 DCP[0] 位进行选择 - DCP[0] = 0: 非反向 - DCP[0] = 1: 反向	
数据方向	先 MSB 还是 LSB	

注: m: 单元编号(m = 0), n: 通道编号(n = 0 至3) , mn = 00至03

注1.参见第27节，电气特性。

(1) 寄存器设置

表 19.20 至表 19.25 显示了简化 SPI 主接收寄存器内容的示例。

(a) 串行模式寄存器 mn (SMRmn)

表 19.20 简化 SPI 主接收的串行模式寄存器 mn (SMRmn) 内容示例

少量	象征	设置值	功能
0	MD0	0/1	通道 n 的中断源 0: 传输结束中断 1: 缓冲区为空中断
2:1	MD1[1:0]	00b	设置通道 n 的工作模式 0 0: 简化的 SPI 模式
5:3	—	100b	设置已禁用（已设置为初始值）
6	SIS0	0	简化 SPI 模式下的设置是固定的
7	—	0	设置已禁用（已设置为初始值）
8	STS	0	选择启动触发源 0: 仅软件触发有效（选择用于简化 SPI、UART 传输和简化 I ² C）。
13:9	—	00000b	设置已禁用（已设置为初始值）
14	CCS	0	选择通道 n 的传输时钟 (f _{TCLK}) 0: 由 CKS 位指定的分频操作时钟 f _{MCK}
15	CKS	0/1	通道 n 的运行时钟(f _{MCK}) 0: 预分频器输出时钟CKm0由 SPSm 寄存器设置 1: 预分频器输出时钟 CKm1 由 SPSm 寄存器设置

(b) Serial communication operation setting register mn (SCRmn)

Table 19.21 Example of serial communication operation setting register mn (SCRmn) contents for master reception of simplified SPI

Bit	Symbol	Set value	Function
1:0	DLS[1:0]	10b or 11b	Setting of data length 1 0: 7-bit data length 1 1: 8-bit data length
3:2	—	01b	Setting disabled (set to the initial value)
5:4	SLC[1:0]	00b	Since this bit is dedicated to UART mode, it is fixed in the simplified SPI mode.
6	—	0	Setting disabled (set to the initial value)
7	DIR	0/1	Selection of data transfer sequence in simplified SPI and UART modes 0: Input or output data with MSB first 1: Inputs or outputs data with LSB first
9:8	PTC[1:0]	00b	Since this bit is dedicated to UART mode, it is fixed in the simplified SPI mode.
10	EOC	0	Since this bit is dedicated to UART receive modes, it is fixed in the simplified SPI mode.
11	—	0	Setting disabled (set to the initial value)
13:12	DCP[1:0]	00b to 11b	Selection of data and clock phase in simplified SPI mode For details about the setting, see section 19.3. Register Descriptions .
15:14	TRXE[1:0]	01b	Setting TRXE[1:0] = 01b is fixed in the simplified SPI master reception mode

(c) Serial data register mn (SDRmn)

Table 19.22 Example of serial data register mn (SDRmn) contents for master reception of simplified SPI

Bit	Symbol	Set value	Function
7:0	DAT[7:0]	0xFF	Receive data (Write 0xFF as dummy data)
8	DAT[8]	0	0 Fixed
15:9	STCLK[6:0]	0x00 to 0x7F	Baud rate setting (Operation clock (f _{MCK}) division setting)

(d) Serial output register m (SOM)

Set only the bit of the target channel.

Table 19.23 Example of serial output register m (SOM) contents for master reception of simplified SPI

Bit	Symbol	Set value	Function
n	SO[n]	0/1	Serial data output of channel n 0: Serial data output value is 0 1: Serial data output value is 1
n+8	CKO[n]	0/1	Communication starts when a bit is 1 if the clock phase is non reverse (SCRmn.DCP[0] = 0). If the clock phase is reversed (SCRmn.DCP[0] = 1), communication starts when a bit is 0

(e) Serial output enable register m (SOEm)

This register is not used in this mode.

(b) 串行通信操作设置寄存器 mn (SCRmn)

表 19.21 串行通信操作示例：设置寄存器 mn (SCRmn) 的内容，以便主接收简化的 SPI 信号

少量	象征	设置值	功能
1:0	DLS[1:0]	10b 或 11b	数据长度设置：1.0： 7 位数 据长度；1.1： 8 位数据长度
3:2	—	01b	设置已禁用（已设置为初始值）
5:4	SLC[1:0]	00b	由于该位专用于 UART 模式，因此在简化的 SPI 模式下它是固定的。
6	—	0	设置已禁用（已设置为初始值）
7	DIR	0/1	简化 SPI 和 UART 模式下数据传输顺序的选择 0： 输入或输出数据，最高有效位在前。 1： 输入或输出数据以最低有效位 (LSB) 为先。
9:8	PTC[1:0]	00b	由于该位专用于 UART 模式，因此在简化的 SPI 模式下它是固定的。
10	EOC	0	由于该位专用于 UART 接收模式，因此在简化的 SPI 模式下它是固定的。
11	—	0	设置已禁用（已设置为初始值）
13:12	DCP[1:0]	00b 至 11b	简化 SPI 模式下的数据和时钟相位选择。有关设置的详细信息，请参阅第 1 9.3 节“寄存器描述”。
15:14	TRXE[1:0]	01b	在简化的 SPI 主接收模式下，TRXE[1:0] = 01b 的设置是固定的。

(c) 串行数据寄存器 mn (SDRmn)

表 19.22 简化 SPI 主接收的串行数据寄存器 mn (SDRmn) 内容示例

少量	象征	设置值	功能
7:0	DAT[7:0]	0xFF	接收数据 （写入 0xFF 作为虚拟数据）
8	DAT[8]	0	0 固定
15:9	STCLK[6:0]	0x00 至 0x7F	波特率设置 （操作时钟(f _{MCK}) 分频设置）

(d) 串行输出寄存器 m (SOM)

仅设置目标通道的位。

表 19.23 简化 SPI 主接收的串行输出寄存器m(SOM) 内容示例

少量	象征	设置值	功能
n	所以[n]	0/1	通道 n 的串行数据输出 0： 串口数据输出值为 0 1： 串口数据输出值为 1
n+8	CKO[n]	0/1	当时钟相位非反相时（SCRmn.DCP[0] = ），通信在比特为1时开始。 0）。如果时钟相位反转(SCRmn.DCP[0] = 1)，则当一个比特为 0

(e) 串行输出使能寄存器m (SOEm)

此寄存器在此模式下不使用。

Table 19.24 Example of serial output enable register m (SOEm) contents for master reception of simplified SPI

Bit	Symbol	Set value	Function
n	SOE[n]	x	Bit that cannot be used in this mode (set to the initial value when not used in any mode)

(f) Serial channel start register m (SSm)

Set only the bit of the target channel to 1.

Table 19.25 Example of serial channel start register m (SSm) contents for master reception of simplified SPI

Bit	Symbol	Set value	Function
n	SS[n]	1	Operation start trigger of channel n 1: Set the SEm.SE[n] bit to 1 to place the channel in the communications waiting state.

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3), mn = 00 to 03

Note: x: Bit that cannot be used in this mode (set to the initial value when not used in any mode)
0/1: Set to 0 or 1 depending on the usage of the user

(2) Operation procedure

Table 19.26 shows the procedure for initial setting of master reception.

Table 19.26 Initial setting procedure for master reception

Step		Process	Detail
Procedure for initial setting for master reception	<1>	Starting initial setting	—
	<2>	Setting the SPSm register	Set the operation clock.
	<3>	Setting the SMRmn register	Set an operation mode, etc.
	<4>	Setting the SCRmn register	Set a communication format.
	<5>	Setting the SDRmn register	Set a transfer baud rate (setting the transfer clock by dividing the operation clock (f_{MCK})).
	<6>	Setting the SOm register	Set the initial output level of the serial clock (SOm.CKO[n]).
	<7>	Setting port	Enable clock output of the target channel.
	<8>	Writing to the SSm register	Set the SSm.SS[n] bit of the target channel to 1 and set SEm.SE[n] = 1 to enable operation.
	<9>	Completing initial setting	Initial setting is completed. Set dummy data to the SDRmn.DAT[7:0] bits and start communication.

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3)

Table 19.27 shows the procedure for stopping master reception.

Table 19.24 简化版主接收串行输出使能寄存器m(SOEm) 内容示例

少量	象征	设置值	功能
n	SOE[n]	x	此模式下无法使用的位（在任何模式下未使用时，设置为初始值）

(f) 串行通道起始寄存器m（SSm）

仅将目标通道的位设置为 1。

表 19.25 简化主接收串行通道起始寄存器m (SSm) 内容示例

少量	象征	设置值	功能
n	SS[n]	1	通道 n 的操作启动触发 1: 将 SEm.SE[n] 位设置为 1，将信道置于通信等待状态状态。

注：m：单元编号(m = 0), n：通道编号(n = 0至3) ， mn = 00至03

注意：x: 此模式下不能使用的位（在任何模式下未使用时设置为初始值）
0/1: 根据用户使用情况设置为 0 或 1

(2) 操作流程

表 19.26 显示了主接收机的初始设置程序。

表 19.26 主接收机初始设置程序

步		过程	细节
初始程序 主接待处设置	<1>	初始设置	—
	<2>	设置 SPSm 寄存器	设置运行时钟。
	<3>	设置 SMRmn 寄存器	设置操作模式等。
	<4>	设置 SCRmn 寄存器	制定沟通模式。
	<5>	设置 SDRmn 寄存器	设置传输波特率（通过分频操作时钟来设置传输时钟(f_{MCK})）。
	<6>	设置 SOm 寄存器	设置串行时钟的初始输出电平（SOm.CKO[n]）。
	<7>	设置端口	启用目标通道的时钟输出。
	<8>	致函SSm登记处	将目标通道的 SSm.SS[n] 位设置为 1，并将SEm.SE[n] = 1 设置为启用操作。
	<9>	完成初始设置	初始设置完成。 将虚拟数据设置到 SDRmn.DAT[7:0] 位并开始通信。

注：m：单元编号(m = 0), n：通道编号(n = 0 至3)

表 19.27 显示了停止主接收的步骤。

Table 19.27 Procedure for stopping master reception

Step		Process	Detail
Procedure for stopping master reception	<1>	Starting setting to stop	—
	<2>	Wait until SSRmn.TSF is cleared (optional)	If there is any data being transferred, wait for their completion. If there is a requirement to stop, do not wait.
	<3>	Writing the STm register	Write 1 to the STm.ST[n] bit of the target channel (stopping operation by setting SEm.SE[n] = 0).
	<4>	Changing setting of the SOEm register	Set the SOEm.SOE[n] bit to 0 and stop the output of the target channel.
	<5>	Changing setting of the SOm register (optional)	The levels of the serial clock (SOm.CKO[n]) and serial data (SOm.SO[n]) on the target channel can be changed if required.
	<6>	Stop setting is completed	After the stop setting is completed, go to the next processing.

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3)

Table 19.28 shows the procedure for resuming master reception.

Table 19.28 Procedure for resuming master reception

Step		Process	Detail
Procedure for resuming master transmission	<1>	Starting setting for resumption	—
	<2>	Wait until completing slave preparations	Wait until the communication target (slave) stops or communication operation completed.
	<3>	Port manipulation	Disable data output and clock output of the target channel.
	<4>	Changing setting of the SPSm register (optional)	Reset the register to change the operation clock setting.
	<5>	Changing setting of the SDRmn register (optional)	Reset the register to change the transfer baud rate setting (setting the transfer clock by dividing the operation clock (fMCK)).
	<6>	Changing setting of the SMRmn register (optional)	Reset the register to change serial mode register mn (SMRmn) setting.
	<7>	Changing setting of the SCRmn register (optional)	Reset the register to change serial communication operation setting register mn (SCRmn) setting.
	<8>	Changing setting of the SOm register (optional)	Set the initial output level of the serial clock (SOm.CKO[n]).
	<9>	Clearing error flag	If the SSRmn.OVF flag remains set, clear this using serial flag clear trigger register mn (SIRmn).
	<10>	Port manipulation	Enable clock output of the target channel.
	<11>	Writing to the SSm register	Set the SSm.SS[n] bit of the target channel to 1 and set SEm.SE[n] = 1 to enable operation.
	<12>	Completing resumption setting	Setting is completed. Set dummy data to the SDRmn.DAT[7:0] bits and start communication.

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3)

(3) Processing flow (in single reception mode)

Figure 19.8 shows the timing of master reception (in single reception mode) (Type 1: SCRmn.DCP[1:0] = 00b).

表 19.27 停止主接收程序

步		过程	细节
停止主接收程序	<1>	开始设置停止	—
	<2>	等待 SSRmn.TSF 被清除（可选）	如果有数据传输，请等待传输完成。如果需要停车，不要等待。
	<3>	写入 STm 寄存器	将 1 写入目标通道的 STm.ST[n] 位（通过设置SEm.SE[n] = 0停止操作）。
	<4>	更改 SOEm 寄存器的设置	将 SOEm.SOE[n] 位设置为 0，停止目标通道的输出。
	<5>	更改 SOm 寄存器的设置（可选）	如果需要，可以更改目标通道上的串行时钟(SOm.CKO[n]) 和串行数据(SOm.SO[n])的电平。
	<6>	停止设置完成	停止设置完成后，进行下一步处理。

注: m: 单元编号(m = 0), n: 通道编号(n = 0 至3)

表 19.28 显示了恢复主接收的步骤。

表 19.28 恢复主接收程序

步		过程	细节
恢复主传输的步骤	<1>	恢复的初始设置	—
	<2>	等奴隶准备工作完成后再做决定。	等待通信目标（从设备）停止或通信操作完成。
	<3>	端口操作	禁用目标通道的数据输出和时钟输出。
	<4>	更改 SPSm 寄存器的设置（可选）	重置寄存器以更改操作时钟设置。
	<5>	更改 SDRmn 寄存器的设置（可选）	重置寄存器以更改传输波特率设置（通过分频操作时钟 (fMCK) 来设置传输时钟）。
	<6>	更改 SMRmn 寄存器的设置（可选）	重置寄存器以更改串行模式寄存器 mn (SMRmn) 设置。
	<7>	更改 SCRmn 寄存器的设置（可选）	重置寄存器以更改串行通信操作设置寄存器 mn (SCRmn) 设置。
	<8>	更改 SOm 寄存器的设置（可选）	设置串行时钟的初始输出电平（SOm.CKO[n]）。
	<9>	清除错误标志	如果 SSRmn.OVF 标志仍然设置，则使用串行标志清除触发寄存器 mn (SIRmn) 清除此标志。
	<10>	端口操作	启用目标通道的时钟输出。
	<11>	致函SSm登记处	将目标通道的 SSm.SS[n] 位设置为 1，并将SEm.SE[n] = 1 设置为启用操作。
	<12>	完成恢复设置	设置完成。将虚拟数据设置到 SDRmn.DAT[7:0] 位并开始通信。

注: m: 单元编号(m = 0) n: 通道编号(n = 0 至3)

(3) 处理流程（单次接收模式）

图 19.8 显示了主接收的时序（在单接收模式下）（类型 1: SCRmn。DCP[1:0] = 00b）。

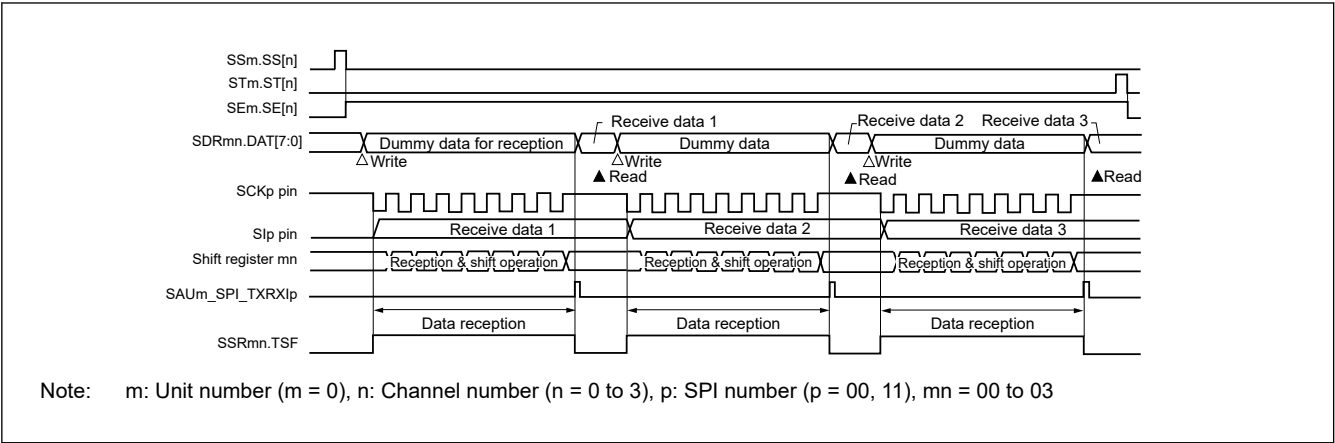


Figure 19.8 Timing of master reception (in single reception mode) (type 1: SCRmn.DCP[1:0] = 00b)

Figure 19.9 shows the flowchart of master reception (in single reception mode).

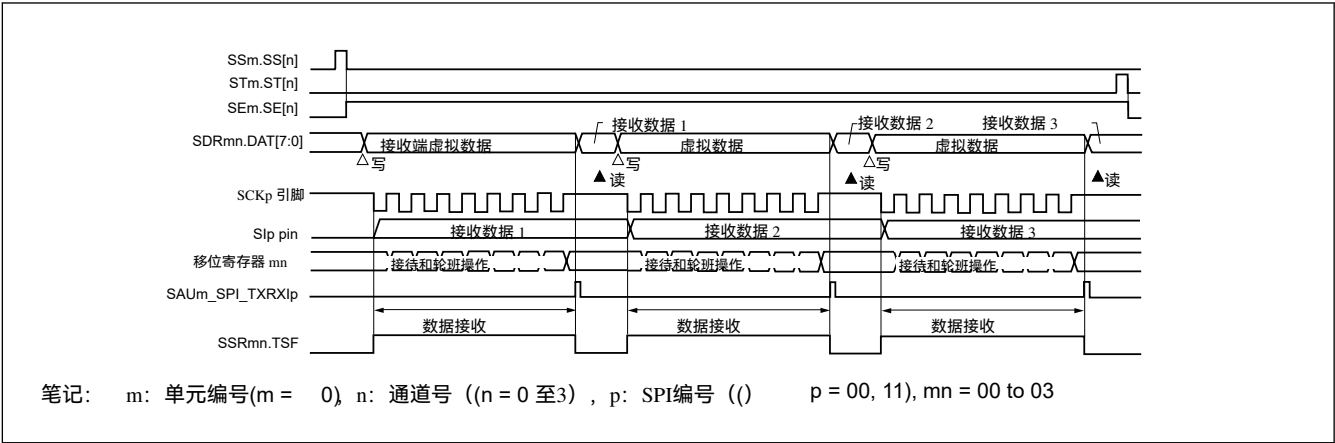


图 19.8 主接收时序（单接收模式）（类型 1: SCRmn.DCP[1:0] = 00b）

图 19.9 显示了主接收（单接收模式）的流程图。

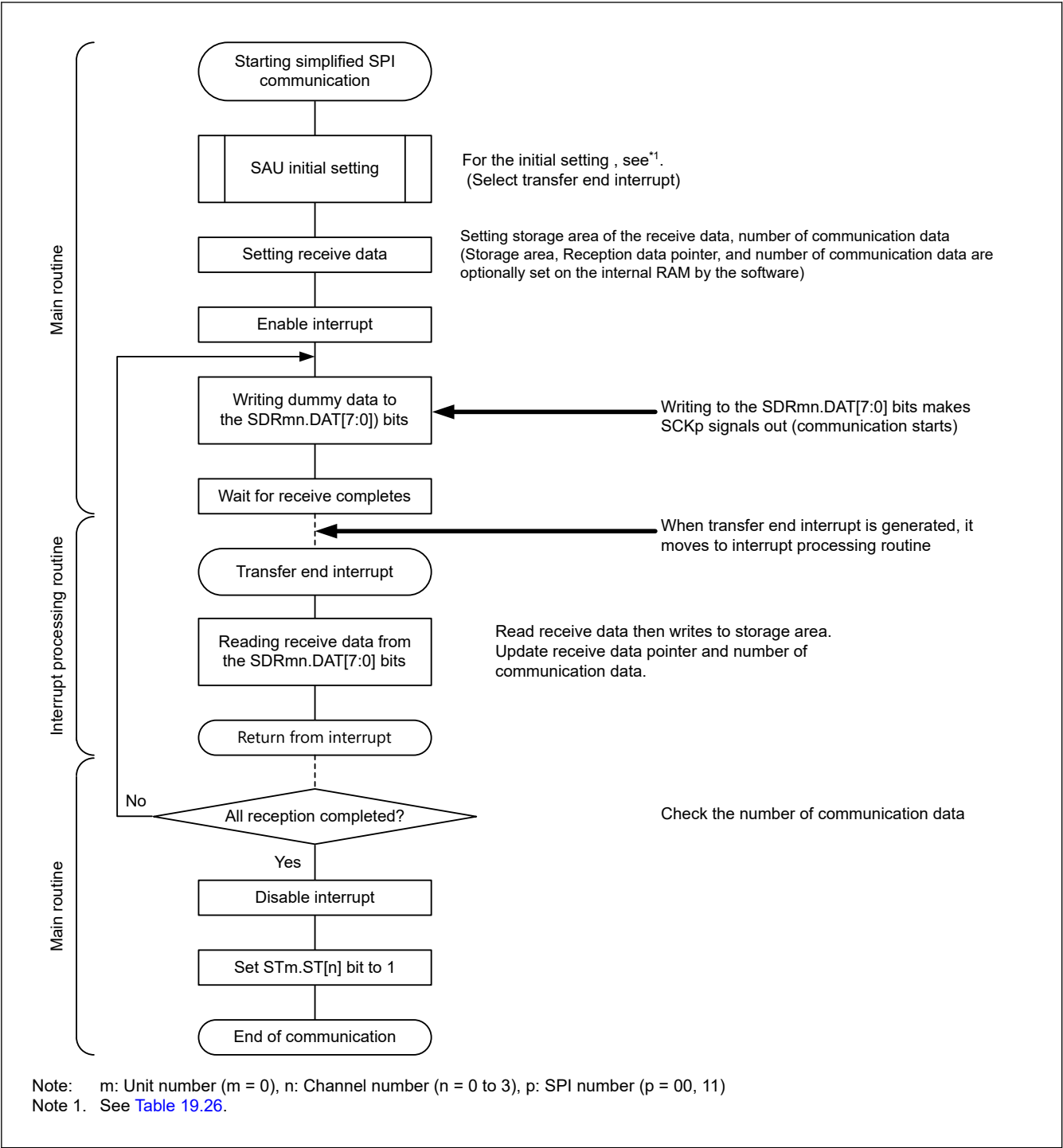


Figure 19.9 Flowchart of master reception (in single reception mode)

(4) Processing flow (in continuous reception mode)

Figure 19.10 shows the timing of master reception (in continuous reception mode) (Type 1: SCRmn.DCP[1:0] = 00b).

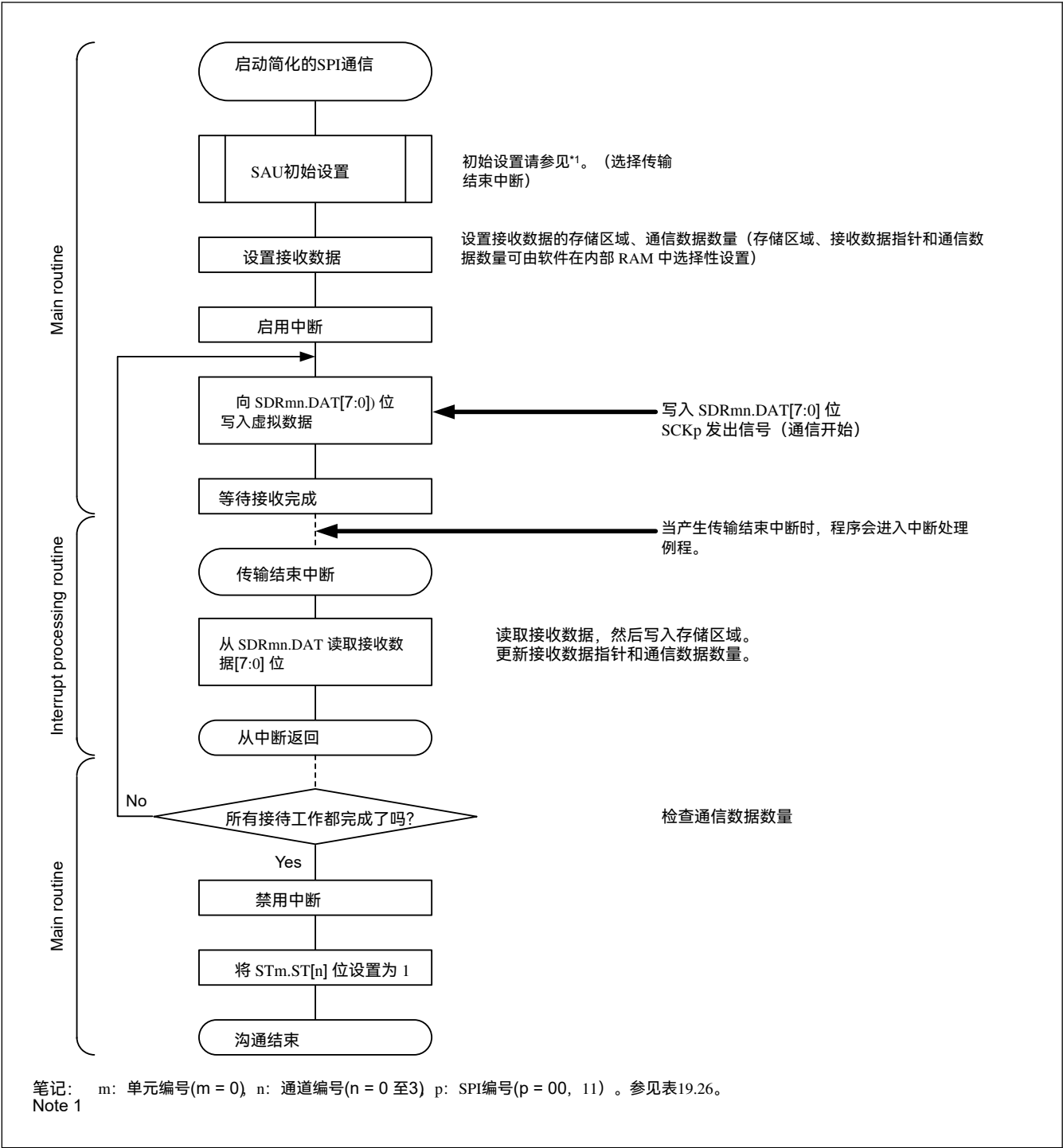


图 19.9 主接收流程图（单接收模式）

(4) 处理流程（连续接收模式）

图 19.10 显示了主接收的时序（在连续接收模式下）（类型 1: SCRmn.DCP[1:0] = 00b）。

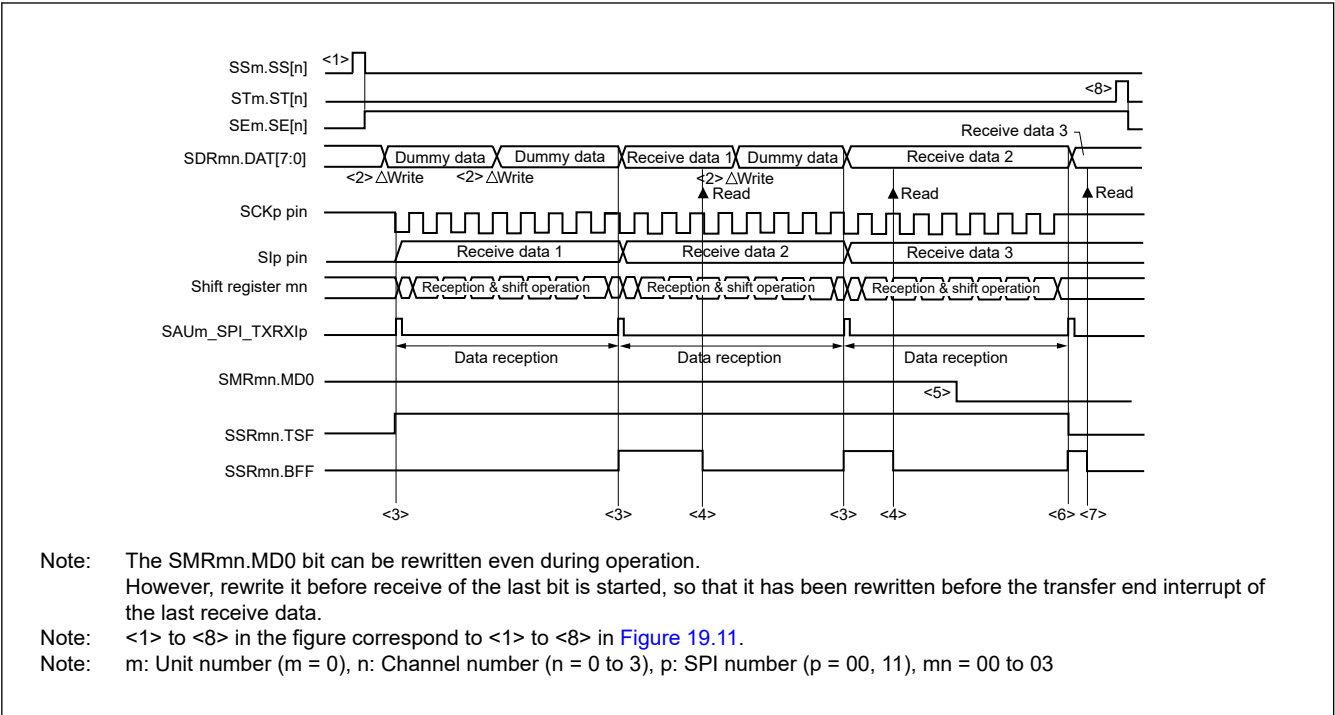


Figure 19.10 Timing of master reception (in continuous reception mode) (type 1: DCPmn[1:0] = 00b)

Figure 19.11 shows the flowchart of master reception (in continuous reception mode).

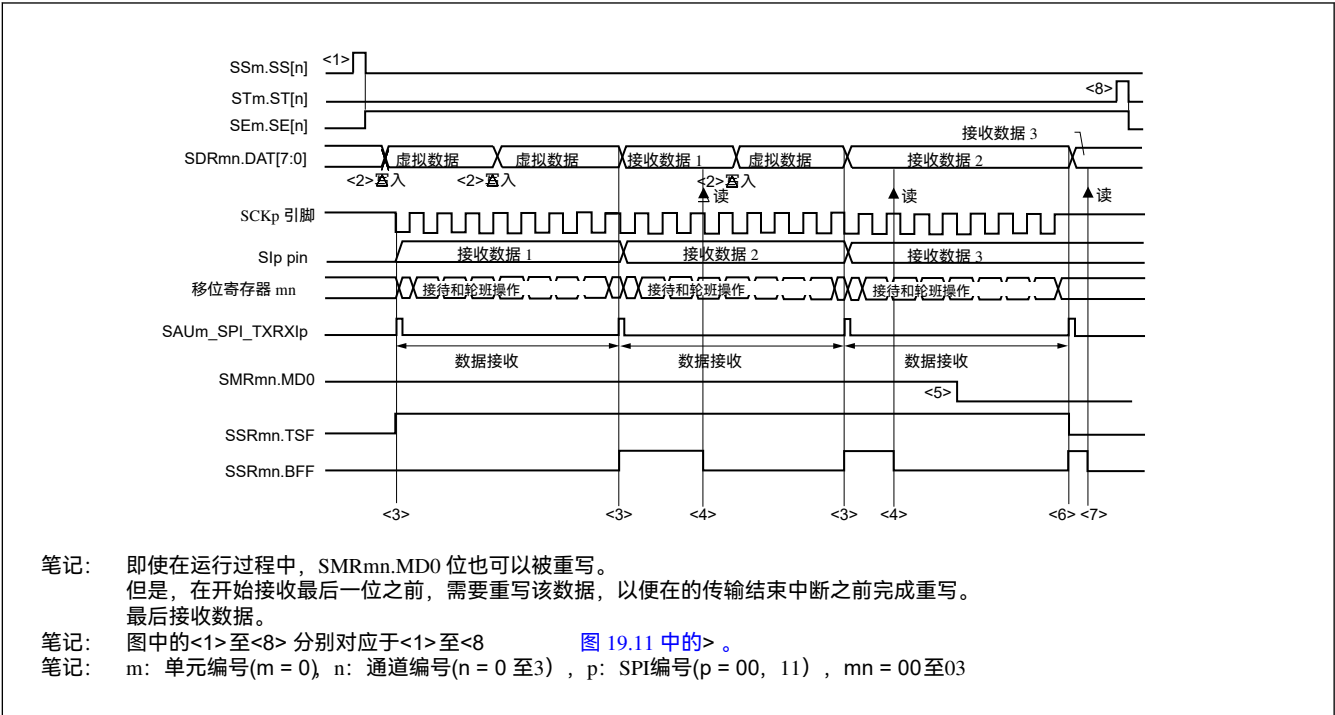


图 19.10 主接收时序 (连续接收模式) (类型 1: DCPmn[1:0] = 00b)

图 19.11 显示了主接收流程图 (在连续接收模式下)。

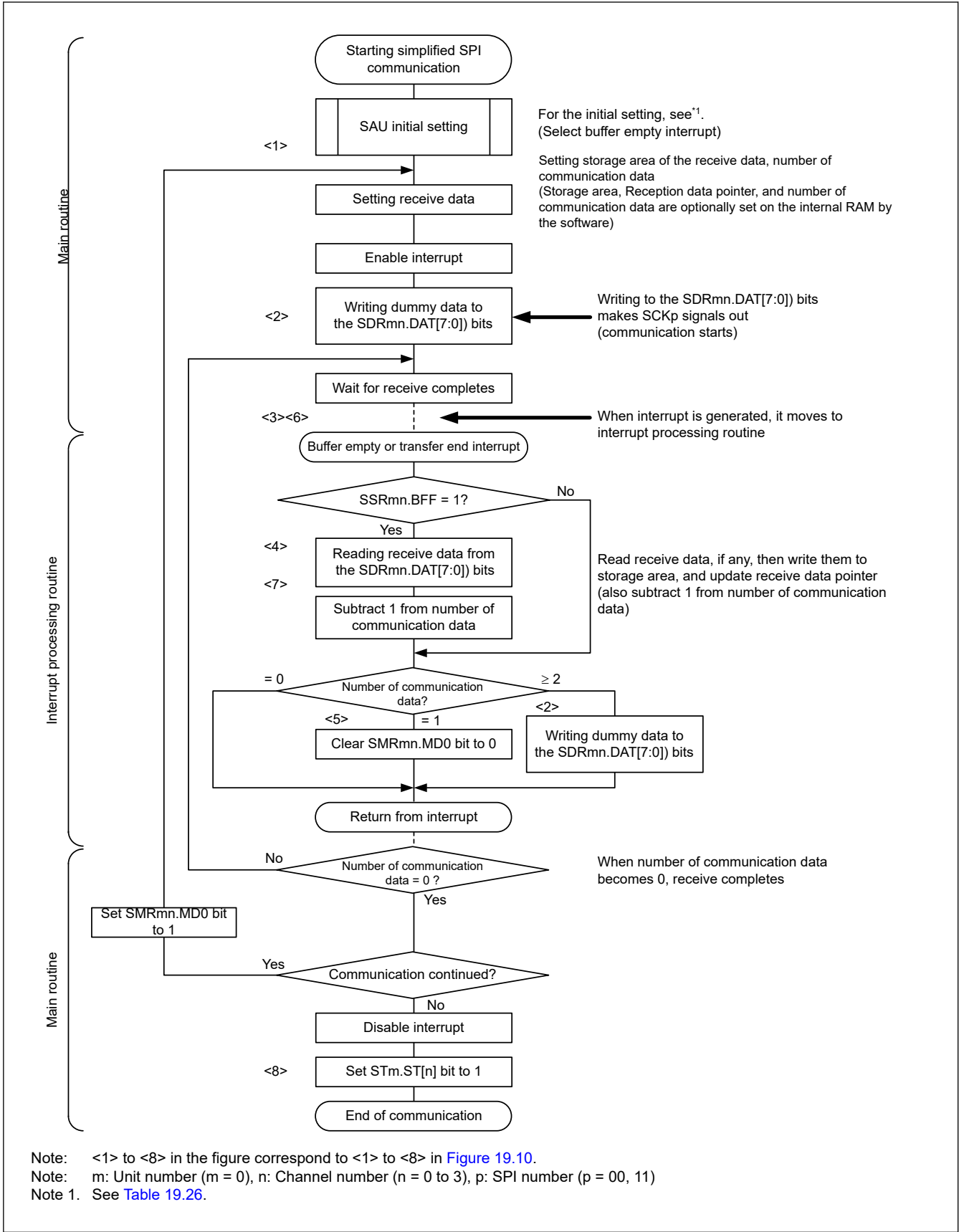


Figure 19.11 Flowchart of master reception (in continuous reception mode)

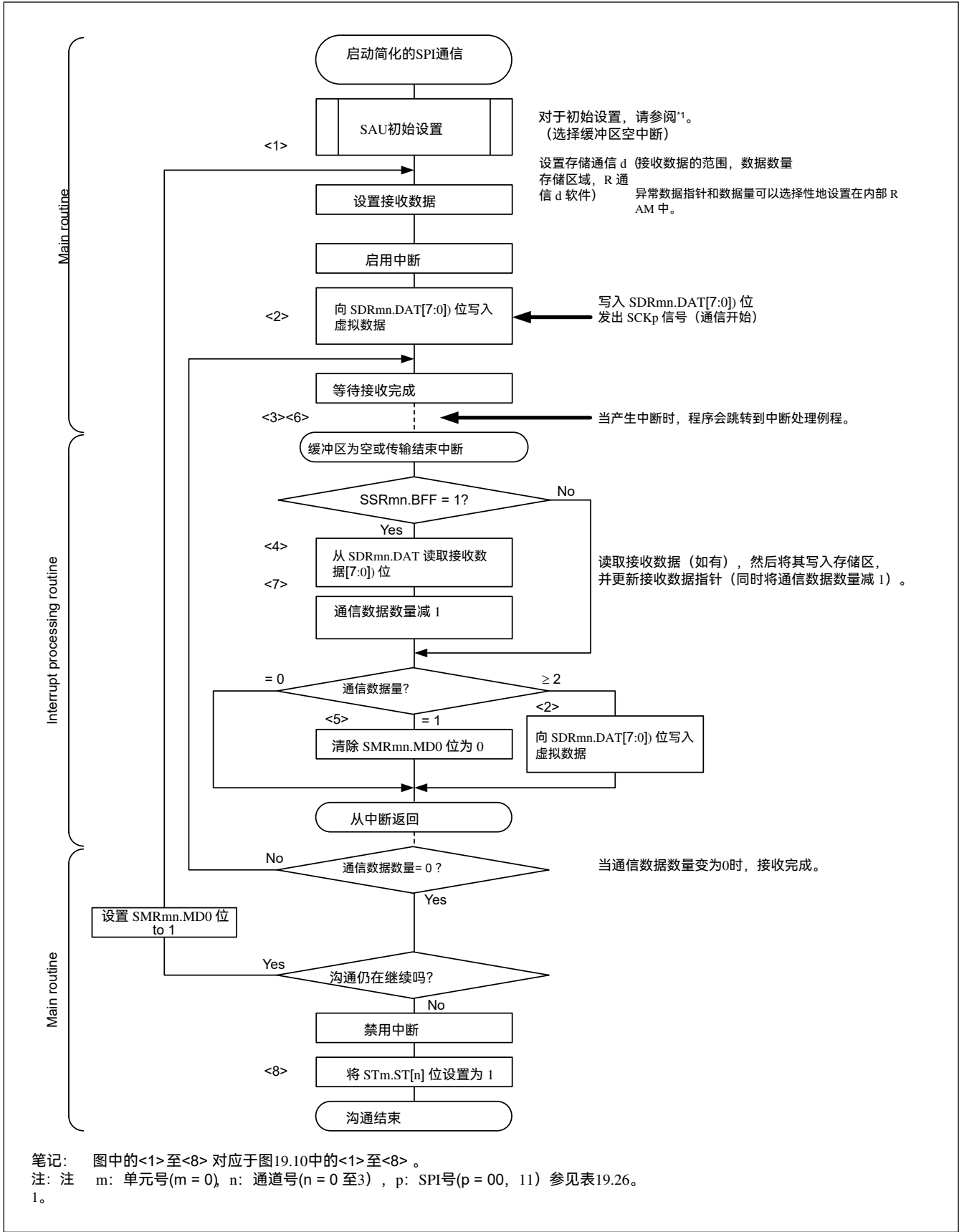


图 19.11 主接收流程图 (连续接收模式)

19.5.3 Master Transmission and Reception

Master transmission and reception is when a microcontroller outputs a transfer clock and transmits and receives data to and from other device.

Table 19.29 shows the specification for master transmission and reception of Simplified SPI.

Table 19.29 Specification for master transmission and reception of Simplified SPI

Simplified SPI	SPI00	SPI11
Target channel	Channel 0 of SAU0	Channel 3 of SAU0
Pins used	SCK00, SI00, SO00	SCK11, SI11, SO11
Interrupt	SAU0_SPI_TXRXI00	SAU0_SPI_TXRXI11
	Transfer end interrupt (in single-transfer mode) or buffer empty interrupt (in continuous transfer mode) can be selected.	
Error detection flag	Overrun error detection flag (SSRmn.OVF) only	
Transfer data length	7 or 8 bits	
Transfer rate*1	Max. PCLKB /2 [Hz] (SPI00 only), PCLKB /4 [Hz] Min. PCLKB/(2 × 2 ¹⁵ × 128)[Hz]	
Data phase	Selectable by the DCP[1] bit of the SCRmn register - DCP[1] = 0: Data I/O starts at the start of the operation of the serial clock. - DCP[1] = 1: Data I/O starts half a clock cycle before the start of the serial clock operation.	
Clock phase	Selectable by the DCP[0] bit of the SCRmn register - DCP[0] = 0: Non-reverse - DCP[0] = 1: Reverse	
Data direction	MSB or LSB first	

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3), mn = 00 to 03

Note 1. Use this operation within a range that satisfies the conditions above and the peripheral functions characteristics specified in the electrical characteristics. For details, see [section 27, Electrical Characteristics](#).

(1) Register setting

Table 19.30 to Table 19.35 show examples of the register contents for master transmission and reception of simplified SPI.

(a) Serial mode register mn (SMRmn)

Table 19.30 Example of serial mode register mn (SMRmn) contents for master transmission and reception of simplified SPI (1 of 2)

Bit	Symbol	Set value	Function
0	MD0	0/1	Interrupt source of channel n 0: Transfer end interrupt 1: Buffer empty interrupt
2:1	MD1[1:0]	00b	Setting of operation mode of channel n 0 0: Simplified SPI mode
5:3	—	100b	Setting disabled (set to the initial value)
6	SIS0	0	Setting is fixed in the simplified SPI mode
7	—	0	Setting disabled (set to the initial value)
8	STS	0	Selection of start trigger source 0: Only software trigger is valid (selected for simplified SPI, UART transmission, and simplified I ² C).
13:9	—	00000b	Setting disabled (set to the initial value)

19.5.3 主传输和接收

主控收发是指微控制器输出传输时钟，并向其他设备发送和接收数据。

表 19.29 显示了简化 SPI 的主发送和接收规范。

表 19.29 简化 SPI 主传输和接收规范

简化的SPI	SPI00	SPI11
目标频道	SAU0 的 0 通道	SAU0 的 3 频道
使用的引脚	SCK00、SI00、SO00	SCK11、SI11、SO11
打断	SAU0_SPI_TXRXI00	SAU0_SPI_TXRXI11
	可以选择传输结束中断（单次传输模式）或缓冲区空中断（连续传输模式）。	
错误检测标志	仅溢出错误检测标志 (SSRmn.OVF)	
传输数据长度	7 位或 8 位	
转账费率*1	最大值 PCLKB /2[Hz]（仅限 SPI00）， PCLKB /4[Hz] 最小值PCLKB/(2 × 2 ¹⁵ × 128)[Hz]	
数据阶段	可通过 SCRmn 寄存器的 DCP [1] 位进行选择 - DCP[1] = 0: 数据 I/O 在串行时钟开始运行时启动。 - DCP[1] = 1: 数据 I/O 比串行时钟操作开始提前半个时钟周期开始。	
时钟相位	可通过 SCRmn 寄存器的 DCP[0] 位进行选择 - DCP[0] = 0: 非反向 - DCP[0] = 1: 反转	
数据方向	先 MSB 还是 LSB	

注: m: 单元编号(m = 0) n: 通道编号(n = 0 至3) , mn = 00至03

注1: 此操作应在满足上述条件和电气特性中规定的外围功能特性的范围内使用。详情请参见第27节“电气特性”。

(1) 寄存器设置

表 19.30 至表 19.35 显示了简化 SPI 主发送和接收的寄存器内容示例。

(a) 串行模式寄存器 mn (SMRmn)

表 19.30 简化 SPI 主发送和接收的串行模式寄存器 mn (SMRmn) 内容示例 (1/2)

少量	象征	设置值	功能
0	MD0	0/1	通道 n 的中断源 0: 传输结束中断 1: 缓冲区为空中断
2:1	MD1[1:0]	00b	设置通道 n 的工作模式 0 0: 简化的 SPI 模式
5:3	—	100b	设置已禁用（已设置为初始值）
6	SIS0	0	简化 SPI 模式下的设置是固定的
7	—	0	设置已禁用（已设置为初始值）
8	STS	0	选择启动触发源 0: 仅软件触发有效（选择用于简化 SPI、UART 传输和简化I ² C）。
13:9	—	00000b	设置已禁用（已设置为初始值）

Table 19.30 Example of serial mode register mn (SMRmn) contents for master transmission and reception of simplified SPI (2 of 2)

Bit	Symbol	Set value	Function
14	CCS	0	Selection of transfer clock (f _{TCLK}) of channel n 0: Divided operation clock f _{MCK} specified by the CKS bit
15	CKS	0/1	Operation clock (f _{MCK}) of channel n 0: Prescaler output clock CKm0 set by the SPSm register 1: Prescaler output clock CKm1 set by the SPSm register

(b) Serial communication operation setting register mn (SCRmn)

Table 19.31 Example of serial communication operation setting register mn (SCRmn) contents for master transmission and reception of simplified SPI

Bit	Symbol	Set value	Function
1:0	DLS[1:0]	10b or 11b	Setting of data length 1 0: 7-bit data length 1 1: 8-bit data length
3:2	—	01b	Setting disabled (set to the initial value)
5:4	SLC[1:0]	00b	Since this bit is dedicated to UART mode, it is fixed in the simplified SPI mode.
6	—	0	Setting disabled (set to the initial value)
7	DIR	0/1	Selection of data transfer sequence in simplified SPI and UART modes 0: Inputs or outputs data with MSB first. 1: Inputs or outputs data with LSB first.
9:8	PTC[1:0]	00b	Since this bit is dedicated to UART mode, it is fixed in the simplified SPI mode.
10	EOC	0	Since this bit is dedicated to UART receive modes, it is fixed in the simplified SPI mode.
11	—	0	Setting disabled (set to the initial value)
13:12	DCP [1:0]	00b to 11b	Selection of data and clock phase in simplified SPI mode For details about the setting, see section 19.3. Register Descriptions .
15:14	TRXE[1:0]	11b	Setting TRXE[1:0] = 11b is fixed in the simplified SPI master transmission and reception mode

(c) Serial data register mn (SDRmn)

Table 19.32 Example of serial data register mn (SDRmn) contents for master transmission and reception of simplified SPI

Bit	Symbol	Set value	Function
7:0	DAT[7:0]	0xFF	Transmit data or Receive data (Transmit data setting and receive data read)
8	DAT[8]	0	0 Fixed
15:9	STCLK[6:0]	0x00 to 0x7F	Baud rate setting Operation clock (f _{MCK}) division setting

(d) Serial output register m (SOM)

Set only the bit of the target channel.

表 19.30 简化 SPI 主发送和接收的串行模式寄存器 mn (SMRmn) 内容示例 (2/2)

少量	象征	设置值	功能
14	CCS	0	选择通道 n 的传输时钟 (f _{TCLK}) 0: 由 CKS 位指定的分频操作时钟 f _{MCK}
15	CKS	0/1	通道 n 的运行时钟 (f _{MCK}) 0: 预分频器输出时钟 CKm0 由 SPSm 寄存器设置 1: 预分频器输出时钟 CKm1 由 SPSm 寄存器设置

(b) 串行通信操作设置寄存器 mn (SCRmn)

表 19.31 串行通信操作示例：设置寄存器 mn (SCRmn) 的内容，用于简化 SPI 的主发送和接收

少量	象征	设置值	功能
1:0	DLS[1:0]	10b 或 11b	数据长度设置：1.0：7 位数 据长度；1.1：8 位数数据长度
3:2	—	01b	设置已禁用（已设置为初始值）
5:4	SLC[1:0]	00b	由于该位专用于 UART 模式，因此在简化的 SPI 模式下它是固定的。
6	—	0	设置已禁用（已设置为初始值）
7	DIR	0/1	简化 SPI 和 UART 模式下数据传输顺序的选择：0：以最高有效位 (MS B) 为先进行输入或输出数据。1：以最低有效位 (LSB) 为先进行输入或输出数据。
9:8	PTC[1:0]	00b	由于该位专用于 UART 模式，因此在简化的 SPI 模式下它是固定的。
10	EOC	0	由于该位专用于 UART 接收模式，因此在简化的 SPI 模式下它是固定的。
11	—	0	设置已禁用（已设置为初始值）
13:12	DCP [1:0]	00b 至 11b	简化 SPI 模式下的数据和时钟相位选择。有关设置的详细信息，请参阅第 1 9.3 节“寄存器描述”。
15:14	TRXE[1:0]	11b	在简化的 SPI 主控收发模式下，TRXE[1:0] = 11b 的设置是固定的。

(c) 串行数据寄存器 mn (SDRmn)

表 19.32 简化 SPI 主控发送和接收的串行数据寄存器 mn (SDRmn) 内容示例

少量	象征	设置值	功能
7:0	DAT[7:0]	0xFF	发送数据或接收数据 (发送数据设置并接收数据读取)
8	DAT[8]	0	0 固定
15:9	STCLK[6:0]	0x00 至 0x7F	波特率设置 操作时钟 (f _{MCK}) 分频设置

(d) 串行输出寄存器 m (SOM)

仅设置目标通道的位。

Table 19.33 Example of serial output register m (SOm) contents for master transmission and reception of simplified SPI

Bit	Symbol	Set value	Function
n	SO[n]	0/1	Serial data output of channel n (n = 0 to 3) 0: Serial data output value is 0 1: Serial data output value is 1
n+8	CKO[n]	0/1	Communication starts when a bit is 1 if the clock phase is non-reversed (SCRmn.DCP[0] = 0). If the clock phase is reversed (SCRmn.DCP[0] = 1), communication starts when a bit is 0

(e) Serial output enable register m (SOEm)

Set only the bit of the target channel to 1.

Table 19.34 Example of serial output enable register m (SOEm) contents for master transmission and reception of simplified SPI

Bit	Symbol	Set value	Function
n	SOE[n]	1	Serial output enable or stop of channel n 1: Enable output by serial communication operation

(f) Serial channel start register m (SSm)

Set only the bit of the target channel to 1.

Table 19.35 Example of serial channel start register m (SSm) contents for master transmission and reception of simplified SPI

Bit	Symbol	Set value	Function
n	SS[n]	1	Operation start trigger of channel n 1: Set the SEm.SE[n] bit to 1 to place the channel in communication waiting state

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3), mn = 00 to 03

Note: 0/1: Set to 0 or 1 depending on the usage of the user

(2) Operation procedure

Table 19.36 shows the procedure for initial setting of master transmission and reception.

Table 19.36 Initial setting procedure for master transmission and reception

Step		Process	Detail
Procedure for initial setting of master transmission and reception	<1>	Starting initial setting	—
	<2>	Setting the SPSm register	Set the operation clock.
	<3>	Setting the SMRmn register	Set an operation mode.
	<4>	Setting the SCRmn register	Set a communication format.
	<5>	Setting the SDRmn register	Set a transfer baud rate (setting the transfer clock by dividing the operation clock (f _{MCK})).
	<6>	Setting the SOm register	Set the initial output level of the serial clock (SOm.CKO[n]) and serial data (SOm.SO[n]).
	<7>	Changing setting of the SOEm register	Set the SOEm.SOE[n] bit to 1 and enable data output of the target channel
	<8>	Setting port	Enable data output and clock output of the target channel.
	<9>	Writing to the SSm register	Set the SSm.SS[n] bit of the target channel to 1 and set SEm.SE[n] bit to 1 to enable operation.
	<10>	Completing initial setting	Initial setting is completed. Set transmit data to the SDRmn.DAT[7:0] bits and start.

表 19.33 简化 SPI 主发送和接收的串行输出寄存器m (SOm) 内容示例

少量	象征	设置值	功能
n	所以[n]	0/1	通道 n(n = 0至 3 的串行数据输出) 0: 串口数据输出值为 0 1: 串口数据输出值为 1
n+8	CKO[n]	0/1	当时钟相位未反转时, 如果某个位为 1 (SCRmn.DCP[0] = 0)。如果时钟相位反转(SCRmn.DCP[0] = 1) 则当一个比特为 0

(e) 串行输出使能寄存器 m (SOEm)

仅将目标通道的位设置为 1。

表 19.34 简化 SPI 主控发送和接收的串行输出使能寄存器m(SOEm) 内容示例

少量	象征	设置值	功能
n	SOE[n]	1	通道 n 的串行输出启用或停止 1: 通过串行通信操作启用输出

(f) 串行通道起始寄存器m (SSm)

仅将目标通道的位设置为 1。

表 19.35 简化 SPI 主发送和接收的串行通道起始寄存器m (SSm) 内容示例

少量	象征	设置值	功能
n	SS[n]	1	通道 n 的操作启动触发 1: 将 SEm.SE[n] 位设置为 1, 使通道处于通信等待状态

注: m: 单元编号(m = 0) n: 通道编号(n = 0至3) , mn = 00至03

注意: 0/1: 根据用户使用情况设置为 0 或 1

(2) 操作流程

表 19.36 显示了主发射和接收的初始设置程序。

表 19.36 主发射和接收的初始设置程序

步		过程	细节
主发射和接收的初始设置程序	<1>	初始设置	—
	<2>	设置 SPSm 寄存器	设置运行时钟。
	<3>	设置 SMRmn 寄存器	设置操作模式。
	<4>	设置 SCRmn 寄存器	制定沟通模式。
	<5>	设置 SDRmn 寄存器	设置传输波特率 (通过分频操作时钟 (f _{MCK}) 来设置传输时钟)。
	<6>	设置 SOm 寄存器	设置串行时钟 (SOm.CKO[n]) 和串行数据 (SOm.SO[n]) 的初始输出电平。
	<7>	更改 SOEm 寄存器的设置	将 SOEm.SOE[n] 位设置为 1, 并启用目标通道的数据输出
	<8>	设置端口	启用目标通道的数据输出和时钟输出。
	<9>	致函SSm登记处	将目标通道的 SSm.SS[n] 位设置为 1, 并将 SEm.SE[n] 位设置为 1 以启用操作。
	<10>	完成初始设置	初始设置完成。 将发送数据设置为 SDRmn.DAT[7:0] 位并开始。

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3)

Table 19.37 shows the procedure for stopping master transmission and reception.

Table 19.37 Procedure for stopping master transmission and reception

Step		Process	Detail
Procedure for stopping master transmission and reception	<1>	Starting setting to stop	—
	<2>	Wait until SSRmn.TSF is cleared (optional)	If there is any data being transferred, wait for their completion. If there is a requirement to stop, do not wait.
	<3>	Writing the STm register	Write 1 to the STm.ST[n] bit of the target channel and set SEm.SE[n] = 0 to stop operation.
	<4>	Changing setting of the SOEm register	Set the SOEm.SOE[n] bit to 0 and stop the output of the target channel.
	<5>	Changing setting of the SOm register (optional)	The levels of the serial clock (SOm.CKO[n]) and serial data (SOm.SO[n]) on the target channel can be changed if required.
	<6>	Stop setting is completed	After the stop setting is completed, go to the next processing.

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3)

Table 19.38 shows the procedure for resuming master transmission and reception.

Table 19.38 Procedure for resuming master transmission and reception

Step		Process	Detail
Procedure for Resuming Master Transmission and Reception	<1>	Starting setting for resumption	—
	<2>	Check completing slave preparations	Wait until the communication target (slave) stops or communication operation completed.
	<3>	Port manipulation	Disable data output and clock output of the target channel.
	<4>	Changing setting of the SPSm register (optional)	Reset the register to change the operation clock setting.
	<5>	Changing setting of the SDRmn register (optional)	Reset the register to change the transfer baud rate setting (setting the transfer clock by dividing the operation clock (fMCK)).
	<6>	Changing setting of the SMRmn register (optional)	Reset the register to change serial mode register mn (SMRmn) setting.
	<7>	Changing setting of the SCRmn register (optional)	Reset the register to change serial communication operation setting register mn (SCRmn) setting.
	<8>	Clearing error flag (optional)	If the SSRmn.OVF flag remains set, clear this using serial flag clear trigger register mn (SIRmn).
	<9>	Changing setting of the SOEm register (optional)	Set the SOEm.SOE[n] bit to 0 to stop output from the target channel.
	<10>	Changing setting of the SOm register (optional)	Set the initial output level of the serial clock (SOm.CKO[n]) and serial data (SOm.SO[n]).
	<11>	Changing setting of the SOEm register (optional)	Set the SOEm.SOE[n] bit to 1 and enable output from the target channel.
	<12>	Port manipulation	Enable data output and clock output of the target channel.
	<13>	Writing to the SSm register	Set the SSm.SS[n] bit of the target channel to 1 and set the SEm.SE[n] bit to 1 to enable operation.
	<14>	Completing resumption setting	—

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3)

(3) Processing flow (in single transmission and reception mode)

Figure 19.12 shows the timing of master transmission and reception (in single transmission and reception mode) (type 1: SCRmn.DCP[1:0] = 00b).

注: m: 单元编号(m = 0) n: 通道编号(n = 0 至3)

表 19.37 显示了停止主传输和接收的步骤。

表 19.37 停止主传输和接收的程序

步		过程	细节
停止主传输和接收的程序	<1>	开始设置停止	—
	<2>	等待 SSRmn.TSF 被清除 (可选)	如果有数据传输, 请等待传输完成。如果需要停车, 不要等待。
	<3>	写入 STm 寄存器	将 1 写入目标通道的 STm.ST[n] 位, 并将 SEm.SE[n] = 0 设置为停止操作。
	<4>	更改 SOEm 寄存器的设置	将 SOEm.SOE[n] 位设置为 0, 停止目标通道的输出。
	<5>	更改 SOm 寄存器的设置 (可选)	如果需要, 可以更改目标通道上的串行时钟(SOm.CKO[n]) 和串行数据(SOm.SO[n])的电平。
	<6>	停止设置完成	停止设置完成后, 进行下一步处理。

注: m: 单元编号(m = 0) n: 通道编号(n = 0 至3)

表 19.38 显示了恢复主传输和接收的步骤。

表 19.38 恢复主传输和接收的步骤

步		过程	细节
程序恢复主控传输和接待	<1>	恢复的初始设置	—
	<2>	检查奴隶准备工作是否完成。	等待通信目标 (从设备) 停止或通信操作完成。
	<3>	端口操作	禁用目标通道的数据输出和时钟输出。
	<4>	更改 SPSm 寄存器的设置 (可选)	重置寄存器以更改操作时钟设置。
	<5>	更改 SDRmn 寄存器的设置 (可选)	重置寄存器以更改传输波特率设置 (通过分频操作时钟 (fMCK) 来设置传输时钟) 。
	<6>	更改 SMRmn 寄存器的设置 (可选)	重置寄存器以更改串行模式寄存器 mn (SMRmn) 设置。
	<7>	更改 SCRmn 寄存器的设置 (可选)	重置寄存器以更改串行通信操作设置寄存器 mn (SCRmn) 设置。
	<8>	清除错误标志 (可选)	如果 SSRmn.OVF 标志仍然设置, 则使用串行标志清除触发寄存器 mn (SIRmn) 清除此标志。
	<9>	更改 SOEm 寄存器的设置 (可选)	将 SOEm.SOE[n] 位设置为 0 以停止目标通道的输出。
	<10>	更改 SOm 寄存器的设置 (可选)	设置串行时钟 (SOm.CKO[n]) 和串行数据 (SOm.SO[n]) 的初始输出电平。
	<11>	更改 SOEm 寄存器的设置 (可选)	将 SOEm.SOE[n] 位设置为 1, 并启用目标通道的输出。
	<12>	端口操作	启用目标通道的数据输出和时钟输出。
	<13>	致函SSm登记处	将目标通道的 SSm.SS[n] 位设置为 1, 并将 SEm.SE[n] 位设置为 1 以启用操作。
	<14>	完成恢复设置	—

注: m: 单元编号(m = 0) n: 通道编号(n = 0 至3)

(3) 处理流程（单次发送和接收模式）

图 19.12 显示了主传输和接收的时序（在单次传输和接收模式下）（类型 1: SCRmn.DCP[1:0] = 00b),

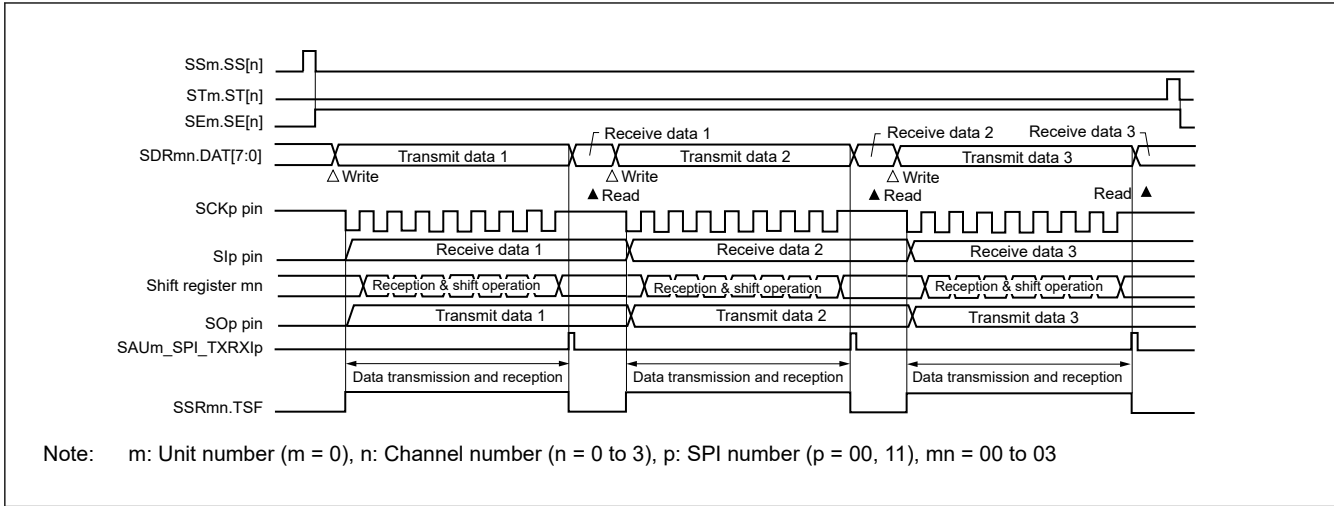


Figure 19.12 Timing of master transmission and reception (in single transmission and reception mode) (type 1: SCRmn.DCP[1:0] = 00b)

Figure 19.13 shows the flowchart of master transmission and reception (in single transmission and reception mode).

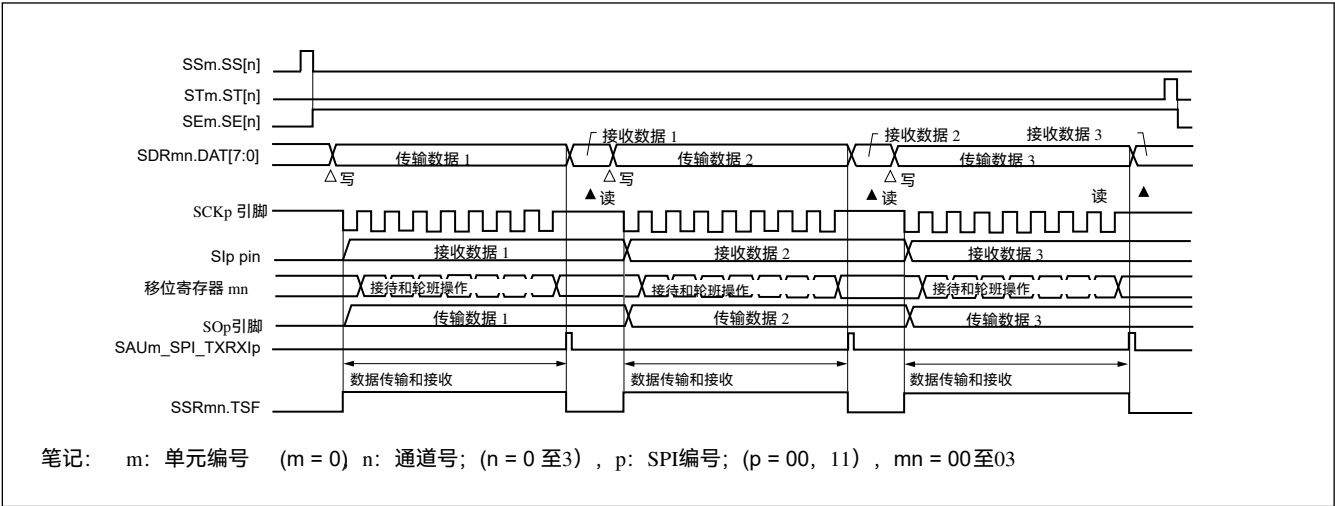


图 19.12 主传输和接收的时序（单传输和接收模式）（类型 1: SCRmn.DCP[1:0] = 00b）

图 19.13 显示了主传输和接收的流程图（在单次传输和接收模式下）。

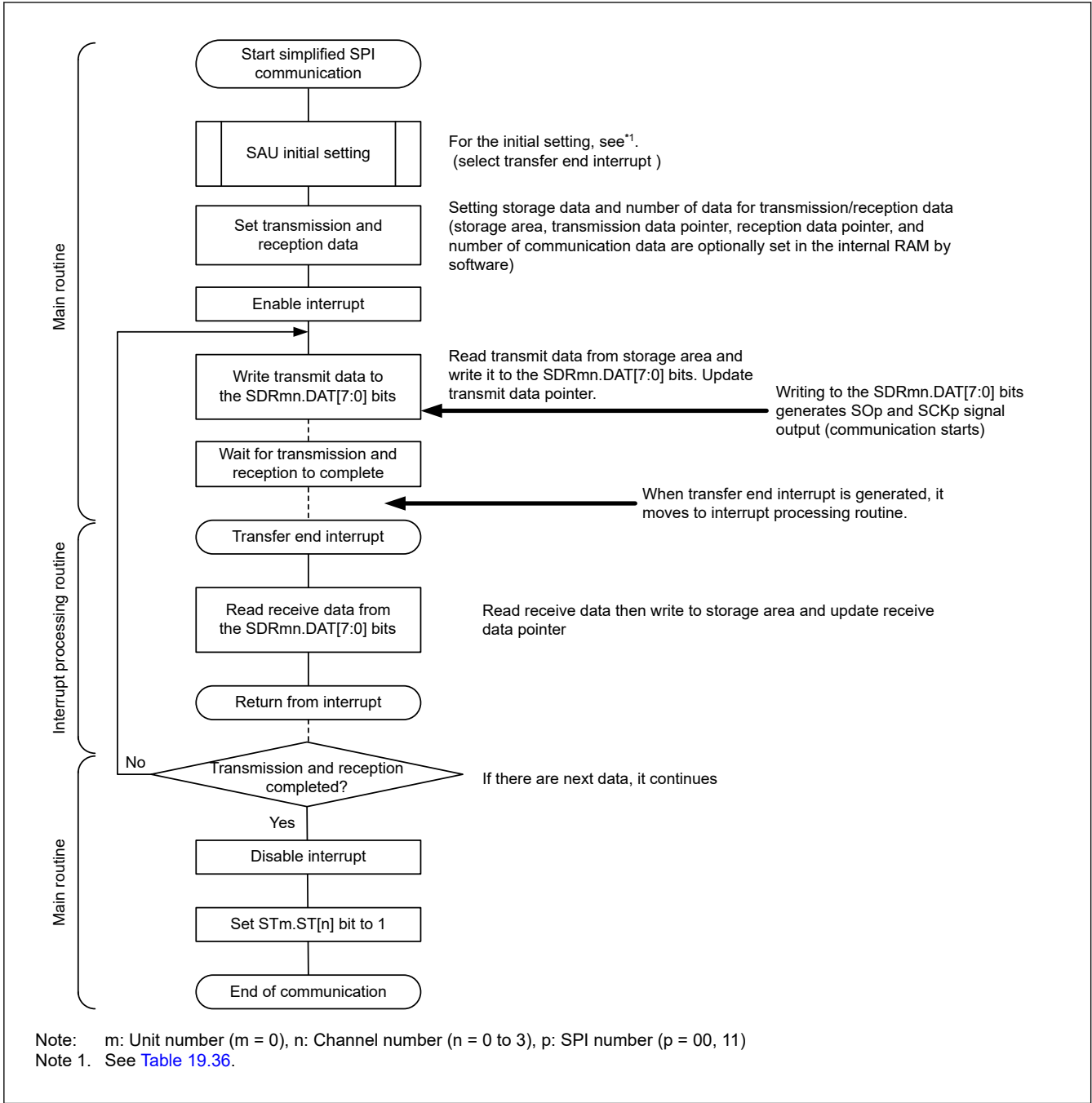


Figure 19.13 Flowchart of master transmission and reception (in single transmission and reception mode)

(4) Processing flow (in continuous transmission and reception mode)

Figure 19.14 shows the timing of master transmission and reception (in continuous transmission and reception mode) (type 1: SCRmn.DCP[1:0] = 00b).

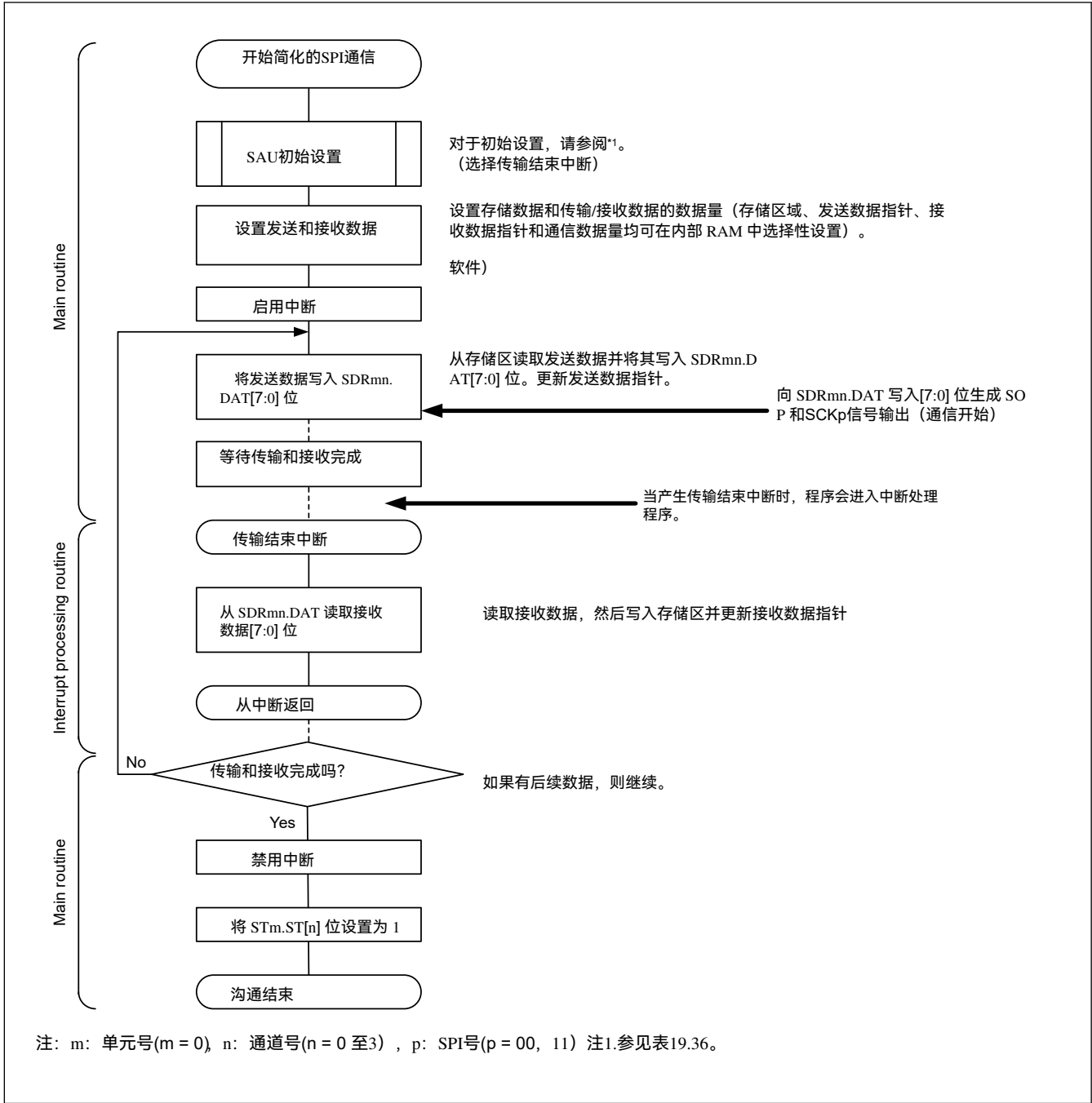


图 19.13 主控发送和接收流程图 (单次发送和接收模式)

(4) 处理流程 (连续发送和接收模式)

图 19.14 显示了主发送和接收的时序 (在连续发送和接收模式下) (类型 1: SCRmn.DCP[1:0] = 00b)。

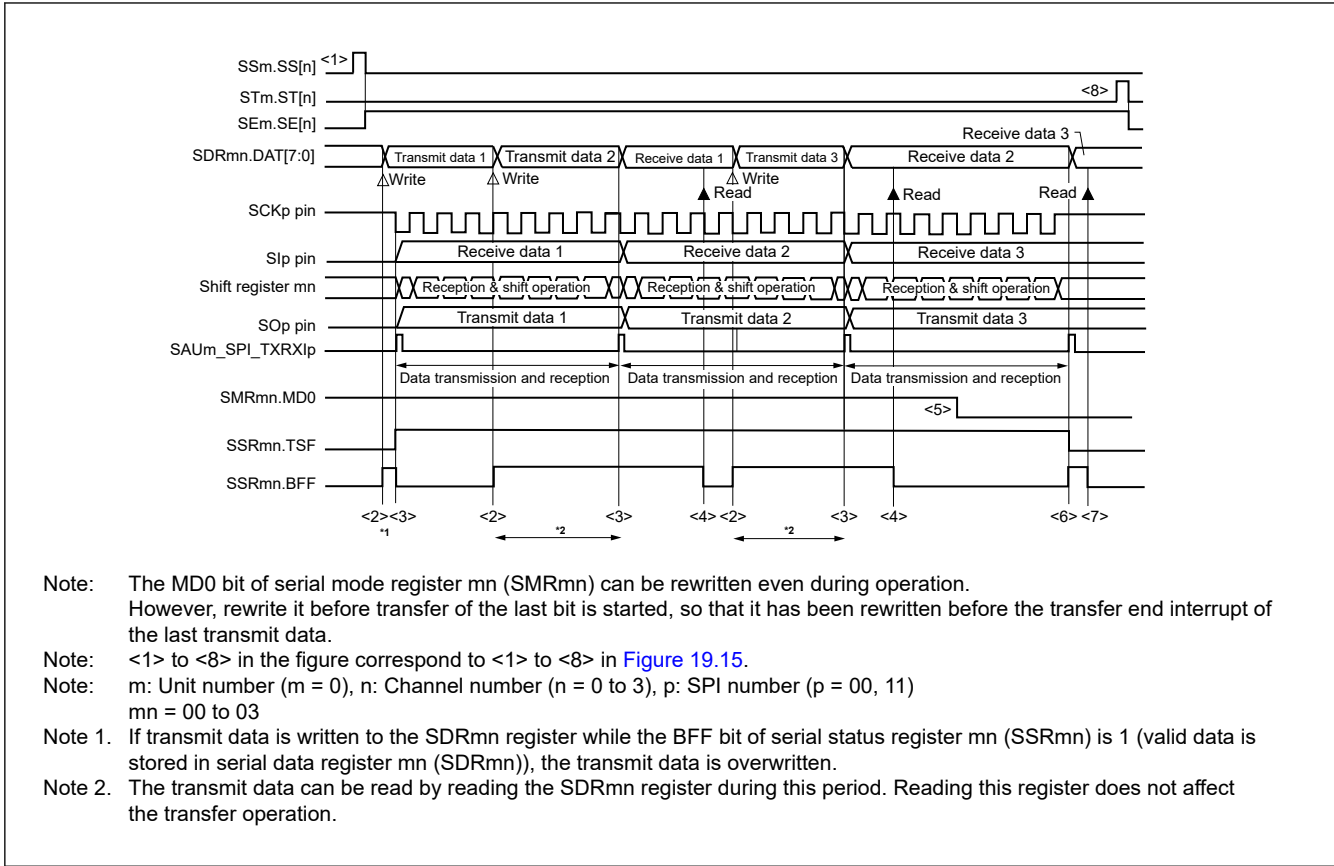


Figure 19.14 Timing of master transmission and reception (in continuous transmission and reception mode) (type 1: SCRmn.DCP[1:0] = 00b)

Figure 19.15 shows the flowchart of master transmission and reception (in continuous transmission and reception mode)

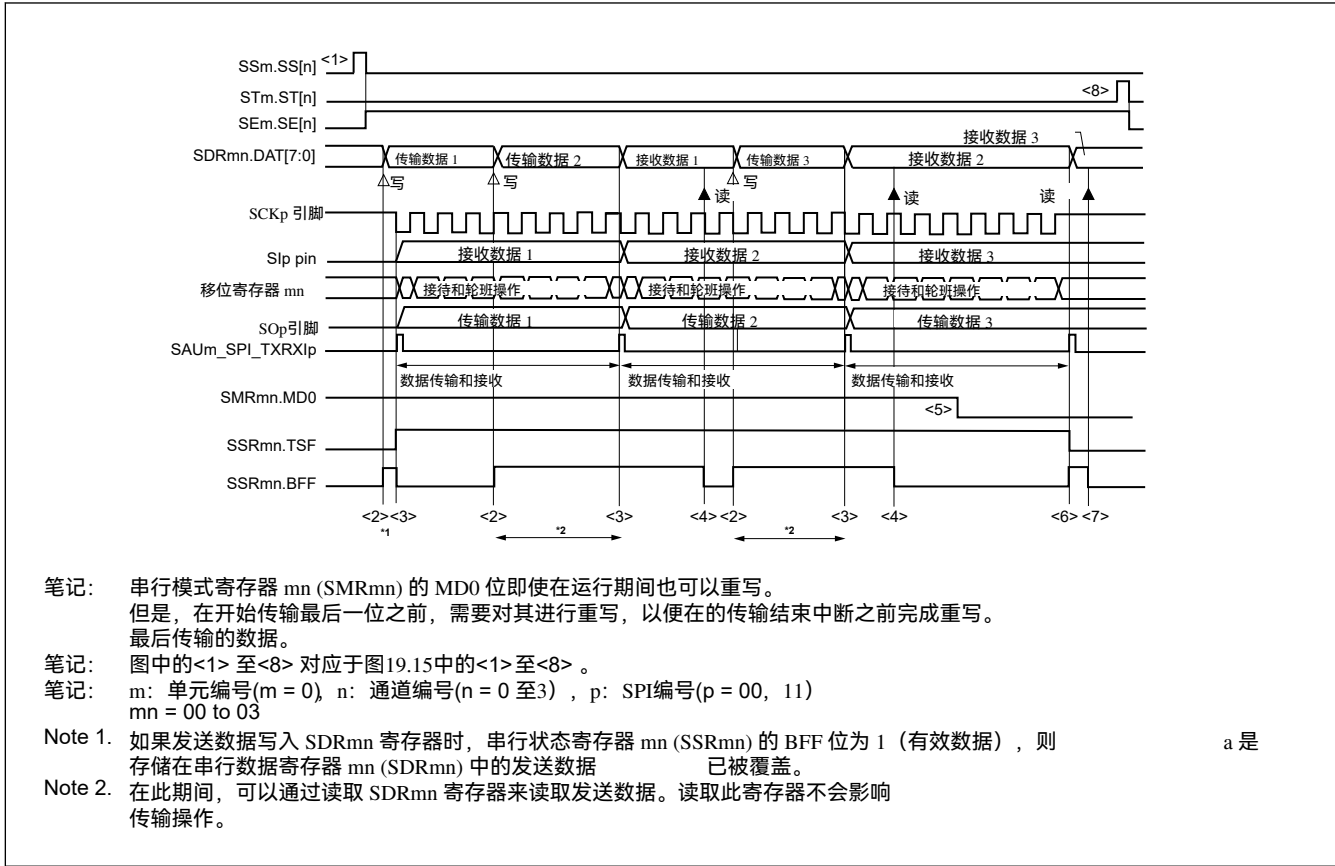


图 19.14 主发送和接收的时序 (连续发送和接收模式) (类型 1: SCRmn.DCP[1:0] = 00b)

图 19.15 显示了主控发送和接收的流程图 (连续发送和接收模式)。

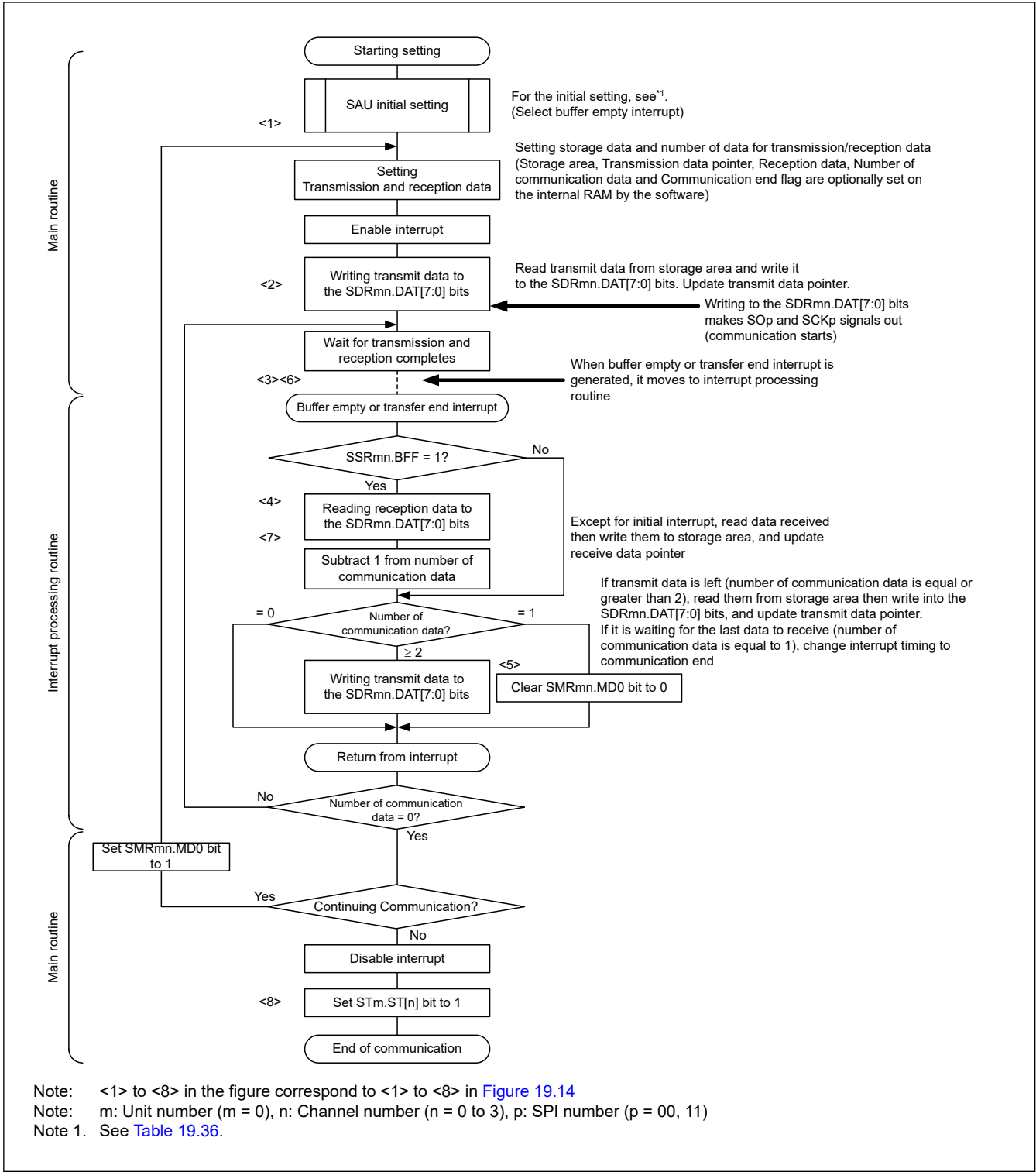


Figure 19.15 Flowchart of master transmission and reception (in continuous transmission and reception mode)

19.5.4 Slave Transmission

Slave transmission is when a microcontroller transmits data to another device in the state of a transfer clock being input from another device.

Table 19.39 shows the specification of slave transmission of simplified SPI.

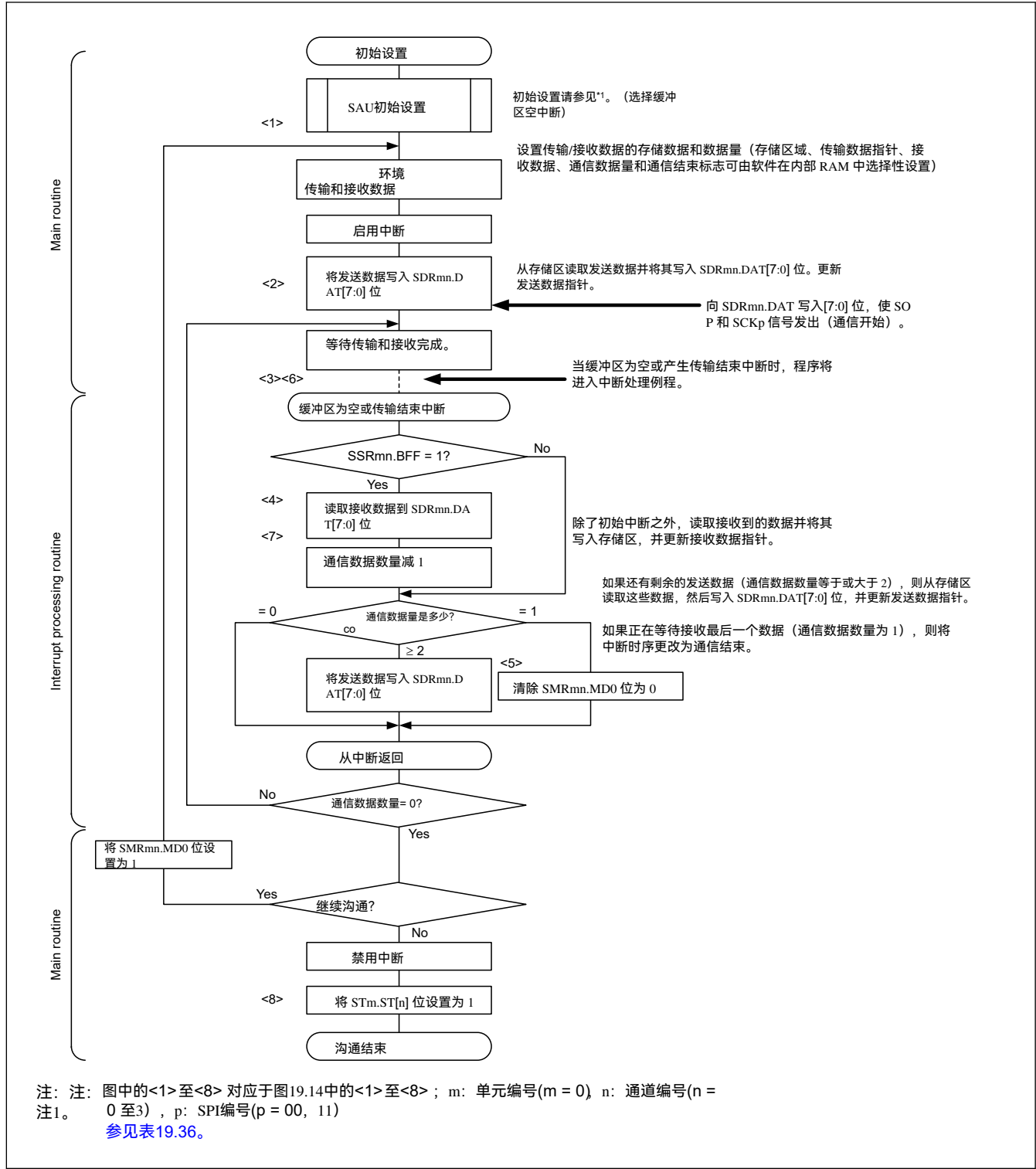


图 19.15 主控发送和接收流程图（连续发送和接收模式）

19.5.4 从动变速器

从设备传输是指当一个微控制器在另一个设备输入传输时钟信号时，向另一个设备传输数据。

表 19.39 显示了简化 SPI 从设备传输的规范。

Table 19.39 Specification of slave transmission of simplified SPI

Simplified SPI	SPI00	SPI11
Target channel	Channel 0 of SAU0	Channel 3 of SAU0
Pins used	SCK00, SO00	SCK11, SO11
Interrupt	SAU0_SPI_TXRXI00	SAU0_SPI_TXRXI11
	Transfer end interrupt (in single-transfer mode) or buffer empty interrupt (in continuous transfer mode) can be selected.	
Error detection flag	Overrun error detection flag (SSRmn.OVF) only	
Transfer data length	7 or 8 bits	
Transfer rate	Max. $f_{MCK}/6$ [Hz] ^{*1 *2}	
Data phase	Selectable by the DCP[1] bit of the SCRmn register - DCP[1] = 0: Data output starts from the start of the operation of the serial clock - DCP[1] = 1: Data output starts half a clock cycle before the start of the serial clock operation	
Clock phase	Selectable by the DCP[0] bit of the SCRmn register - DCP[0] = 0: Non-reverse - DCP[0] = 1: Reverse	
Data direction	MSB or LSB first	

Note: f_{MCK} : Operation clock frequency of target channel

f_{SCK} : Serial clock frequency

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3), mn = 00 to 03

Note 1. Because the external serial clock input to the SCK00 and SCK11 pins is sampled internally and used, the fastest transfer rate is $f_{MCK}/6$ [Hz].

Note 2. Use this operation within a range that satisfies the conditions above and the peripheral functions characteristics specified in the electrical characteristics. For details, see [section 27, Electrical Characteristics](#).

(1) Register setting

[Table 19.40](#) to [Table 19.45](#) show examples of the register contents for slave transmission of simplified SPI.

(a) Serial mode register mn (SMRmn)

Table 19.40 Example of serial mode register mn (SMRmn) contents for slave transmission of simplified SPI (1 of 2)

Bit	Symbol	Set value	Function
0	MD0	0/1	Interrupt source of channel n 0: Transfer end interrupt 1: Buffer empty interrupt
2:1	MD1[1:0]	00b	Setting of operation mode of channel n 0 0: Simplified SPI mode
5:3	—	100b	Setting disabled (set to the initial value)
6	SIS0	0	Setting is fixed in the simplified SPI mode
7	—	0	Setting disabled (set to the initial value)
8	STS	0	Selection of start trigger source 0: Only software trigger is valid (selected for simplified SPI, UART transmission, and simplified I ² C).
13:9	—	00000b	Setting disabled (set to the initial value)
14	CCS	1	Selection of transfer clock (f_{TCLK}) of channel n 1: Clock input f_{SCK} from the SCKp pin (slave transfer in simplified SPI mode)

表 19.39 简化 SPI 从站传输规范

简化的SPI	SPI00	SPI11
目标频道	SAU0 的 0 通道	SAU0 的 3 频道
使用的引脚	SCK00, SO00	SCK11, SO11
打断	SAU0_SPI_TXRXI00	SAU0_SPI_TXRXI11
	可以选择传输结束中断（单次传输模式）或缓冲区空中断（连续传输模式）。	
错误检测标志	仅溢出错误检测标志 (SSRmn.OVF)	
传输数据长度	7 位或 8 位	
转账费率	Max. $f_{MCK}/6$ [Hz] ^{*1 *2}	
数据阶段	可通过 SCRmn 寄存器的 DCP [1] 位进行选择 - DCP[1] = 0 : 数据输出从串行时钟开始运行起。 - DCP[1] = 1 : 数据输出在串行时钟操作开始前半个时钟周期开始	
时钟相位	可通过 SCRmn 寄存器的 DCP[0] 位进行选择 - DCP[0] = 0 : 非反向 - DCP[0] = 1: 反向	
数据方向	先 MSB 还是 LSB	

注: f_{MCK} : 目标通道的工作时钟频率

f_{SCK} : 串行时钟频率

注: m: 单元编号(m = 0), n: 通道编号(n = 0 至 3), mn = 00至03

注1. 由于输入到SCK00和SCK11引脚的外部串行时钟在内部进行采样和使用, 因此最快的传输速率为 $f_{MCK}/6$ [Hz]。

注2: 此操作应在满足上述条件和电气特性中规定的外围功能特性的范围内使用。详情请参见第27节“电气特性”。

(1) 寄存器设置

表 19.40 至表 19.45 显示了简化 SPI 从设备传输的寄存器内容示例。

(a) 串行模式寄存器 mn (SMRmn)

表 19.40 简化从设备传输的串行模式寄存器 mn (SMRmn) 内容示例
SPI (1/2)

少量	象征	设置值	功能
0	MD0	0/1	通道 n 的中断源 0: 传输结束中断 1: 缓冲区为空中断
2:1	MD1[1:0]	00b	设置通道 n 的工作模式 0 0: 简化的 SPI 模式
5:3	—	100b	设置已禁用（已设置为初始值）
6	SIS0	0	简化 SPI 模式下的设置是固定的
7	—	0	设置已禁用（已设置为初始值）
8	STS	0	选择启动触发源 0: 仅软件触发有效（选择用于简化 SPI、UART 传输和简化 I ² C）。
13:9	—	00000b	设置已禁用（已设置为初始值）
14	CCS	1	选择通道 n 的传输时钟 (f_{TCLK}) 1: 时钟输入 f_{SCK} 来自 SCKp 引脚（简化 SPI 模式下的从机传输）

Table 19.40 Example of serial mode register mn (SMRmn) contents for slave transmission of simplified SPI (2 of 2)

Bit	Symbol	Set value	Function
15	CKS	0/1	Operation clock (f _{MCK}) of channel n 0: Prescaler output clock CKm0 set by the SPSm register 1: Prescaler output clock CKm1 set by the SPSm register

(b) Serial communication operation setting register mn (SCRmn)

Table 19.41 Example of serial communication operation setting register mn (SCRmn) contents for slave transmission of simplified SPI

Bit	Symbol	Set value	Function
1:0	DLS[1:0]	10b or 11b	Setting of data length 1 0: 7-bit data length 1 1: 8-bit data length
3:2	—	01b	Setting disabled (set to the initial value)
5:4	SLC[1:0]	00b	Since this bit is dedicated to UART mode, it is fixed in the simplified SPI mode.
6	—	0	Setting disabled (set to the initial value)
7	DIR	0/1	Selection of data transfer sequence in simplified SPI and UART modes 0: Input or output data with MSB first 1: Input or output data with LSB first
9:8	PTC[1:0]	00b	Since this bit is dedicated to UART mode, it is fixed in the simplified SPI mode.
10	EOC	0	Since this bit is dedicated to UART receive modes, it is fixed in the simplified SPI mode.
11	—	0	Setting disabled (set to the initial value)
13:12	DCP[1:0]	00b to 11b	Selection of data and clock phase in simplified SPI mode. Selection of the data and clock phase (For details about the setting, see section 19.3. Register Descriptions.)
15:14	TRXE[1:0]	10b	Setting TRXE[1:0] = 10b is fixed in the simplified SPI slave transmission mode

(c) Serial data register mn (SDRmn)

Table 19.42 Example of serial data register mn (SDRmn) contents for slave transmission of simplified SPI

Bit	Symbol	Set value	Function
7:0	DAT[7:0]	0x00 to 0xFF	Transmit data Transmit data setting
8	DAT[8]	0	0 Fixed
15:9	STCLK[6:0]	0x00	Baud rate setting (do not use in any slave mode)

(d) Serial output register m (SOM)

Set only the bit of the target channel.

Table 19.43 Example of serial output register m (SOM) contents for slave transmission of simplified SPI

Bit	Symbol	Set value	Function
n	SO[n]	0/1	Serial data output of channel n 0: Serial data output value is 0 1: Serial data output value is 1
n+8	CKO[n]	x	Bit that cannot be used in this mode (set to the initial value when not used in any mode)

Table 19.40 简化的从机传输串行模式寄存器 mn (SMRmn) 内容示例
SPI (2/2)

少量	象征	设置值	功能
15	CKS	0/1	通道 n 的运行时钟 (f _{MCK}) 0: 预分频器输出时钟CKm0由 SPSm 寄存器设置 1: 预分频器输出时钟 CKm1 由 SPSm 寄存器设置

(b) 串行通信操作设置寄存器 mn (SCRmn)

表 19.41 简化 SPI 从机传输的串行通信操作示例，设置寄存器 mn (SCRmn) 的内容

少量	象征	设置值	功能
1:0	DLS[1:0]	10b 或 11b	数据长度设置: 1.0: 7 位数 据长度; 1.1: 8 位数据长度
3:2	—	01b	设置已禁用 (已设置为初始值)
5:4	SLC[1:0]	00b	由于该位专用于 UART 模式，因此在简化的 SPI 模式下它是固定的。
6	—	0	设置已禁用 (已设置为初始值)
7	DIR	0/1	简化 SPI 和 UART 模式下数据传输顺序的选择 0: 输入或输出数据，最高有效位在前。 1: 输入或输出数据时，最低有效位 (LSB) 优先。
9:8	PTC[1:0]	00b	由于该位专用于 UART 模式，因此在简化的 SPI 模式下它是固定的。
10	EOC	0	由于该位专用于 UART 接收模式，因此在简化的 SPI 模式下它是固定的。
11	—	0	设置已禁用 (已设置为初始值)
13:12	DCP[1:0]	00b 至 11b	简化SPI模式下的数据和时钟相位选择。 数据和时钟相位的选择 (有关设置的详细信息，请参阅第 19.3 节。 注册说明。)
15:14	TRXE[1:0]	10b	在简化的 SPI 从机传输模式下，设置 TRXE[1:0] = 10b 是固定的。

(c) 串行数据寄存器 mn (SDRmn)

表 19.42 简化从设备传输的串行数据寄存器 mn (SDRmn) 内容示例
SPI

少量	象征	设置值	功能
7:0	DAT[7:0]	0x00 到 0xFF	传输数据 传输数据设置
8	DAT[8]	0	0 固定
15:9	STCLK[6:0]	0x00	波特率设置 (请勿在任何从模式下使用)

(d) 串行输出寄存器 m (SOM)

仅设置目标通道的位。

表 19.43 简化 SPI 从设备传输的串行输出寄存器m (SOM) 内容示例

少量	象征	设置值	功能
n	所以[n]	0/1	通道 n 的串行数据输出 0: 串口数据输出值为 0 1: 串口数据输出值为 1
n+8	CKO[n]	x	此模式下无法使用的位 (在任何模式下未使用时，设置为初始值)

(e) Serial output enable register m (SOEm)

Set only the bit of the target channel to 1.

Table 19.44 Example of serial output enable register m (SOEm) contents for slave transmission of simplified SPI

Bit	Symbol	Set value	Function
n	SOE[n]	1	Serial output enable or stop of channel n 1: Enable output by serial communication operation

(f) Serial channel start register m (SSm)

Set only the bit of the target channel to 1.

Table 19.45 Example of serial channel start register m (SSm) contents for slave transmission of simplified SPI

Bit	Symbol	Set value	Function
n	SS[n]	1	Operation start trigger of channel n 1: Set the SEm.SE[n] bit to 1 to place the channel in communication waiting state

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3), mn = 00 to 03

Note: ×: Bit that cannot be used in this mode (set to the initial value when not used in any mode)
0/1: Set to 0 or 1 depending on the usage of the user

(2) Operation procedure

Table 19.46 shows the procedure for initial setting of slave transmission.

Table 19.46 Initial setting procedure for slave transmission

Step		Process	Detail
Procedure for initial setting of slave transmission	<1>	Starting initial setting	—
	<3>	Setting the SPSm register	Set the operation clock.
	<4>	Setting the SMRmn register	Set an operation mode.
	<5>	Setting the SCRmn register	Set a communication format.
	<6>	Setting the SDRmn register	Set the SDRmn.STCLK[6:0] bits to 0x00 for baud rate setting.
	<7>	Setting the SOm register	Set the initial output level of the serial data (SOm.SO[n]).
	<8>	Changing setting of the SOEm register	Set the SOEm.SOE[n] bit to 1 and enable data output of the target channel.
	<9>	Setting port	Enable data output of the target channel.
	<10>	Writing to the SSm register	Set the SSm.SS[n] bit of the target channel to 1 and SEm.SE[n] bit to 1 to enable operation.
	<11>	Completing initial setting	Initial setting is completed. Set transmit data to the SDRmn.DAT[7:0] bits and wait for a clock from the master

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3)

Table 19.47 shows the procedure for stopping slave transmission.

(e) 串行输出使能寄存器 m (SOEm)

仅将目标通道的位设置为 1。

表 19.44 简化从设备传输的串行输出使能寄存器m(SOEm) 内容示例

少量	象征	设置值	功能
n	SOE[n]	1	通道 n 的串行输出启用或停止 1: 通过串行通信操作启用输出

(f) 串行通道起始寄存器m (SSm)

仅将目标通道的位设置为 1。

表 19.45 简化从设备传输的串行通道起始寄存器m (SSm) 内容示例

少量	象征	设置值	功能
n	SS[n]	1	通道 n 的操作启动触发 1: 将 SEm.SE[n] 位设置为1，使通道处于通信等待状态

注: m: 单元编号(m = 0), n: 通道编号(n = 0至3) , mn = 00至03

注意: ×: 此模式下不能使用的位（在任何模式下未使用时设置为初始值）
0/1: 根据用户使用情况设置为 0 或 1

(2) 操作流程

表 19.46 显示了从属变速器的初始设置程序。

表 19.46 从属变速器的初始设置程序

步		过程	细节
从变速器初始设置程序	<1>	初始设置	—
	<3>	设置 SPSm 寄存器	设置运行时钟。
	<4>	设置 SMRmn 寄存器	设置操作模式。
	<5>	设置 SCRmn 寄存器	制定沟通模式。
	<6>	设置 SDRmn 寄存器	设置 SDRmn.STCLK[6:0] 位为0x00位，以设置波特率。
	<7>	设置 SOm 寄存器	设置串行数据的初始输出电平(SOm.SO[n])。
	<8>	更改 SOEm 寄存器的设置	将 SOEm.SOE[n] 位设置为 1，使能目标通道的数据输出。
	<9>	设置端口	启用目标通道的数据输出。
	<10>	致函SSm登记处	将目标通道的 SSm.SS[n] 位设置为 1，并将 SEm.SE[n] 位设置为 1 以启用操作。
	<11>	完成初始设置	初始设置完成。 将发送数据设置为 SDRmn.DAT[7:0] 位，并等待主时钟信号。

注: m: 单元编号(m = 0), n: 通道编号(n = 0 至3)

表 19.47 显示了停止从属传输的步骤。

Table 19.47 Procedure for stopping slave transmission

Step		Process	Detail
Procedure for stopping slave transmission	<1>	Starting setting to stop	—
	<2>	Wait until the SSRmn.TSF bit is cleared (optional)	If there is any data being transferred, wait for their completion. If there is a requirement to stop, do not wait.
	<3>	Writing the STm register	Write 1 to the STm.ST[n] bit of the target channel and set the SEm.SE[n] bit to 0 to stop operation.
	<4>	Changing setting of the SOEm register	Set the SOEm.SOE[n] bit to 0 and stop the output of the target channel.
	<5>	Changing setting of the SOm register (optional)	The levels of the serial data (SOm.SO[n]) on the target channel can be changed if required.
	<6>	Stop setting is completed	After the stop setting is completed, go to the next processing.

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3)

Table 19.48 shows the procedure for resuming slave transmission.

Table 19.48 Procedure for resuming slave transmission

Step		Process	Detail
Procedure for resuming slave transmission	<1>	Starting setting for resumption	—
	<2>	Wait until completing master preparations	Wait until the communication target (master) stops or communication operation completed.
	<3>	Port manipulation	Disable data output and clock output of the target channel.
	<4>	Changing setting of the SPSm register (optional)	Reset the register to change the operation clock setting.
	<5>	Changing setting of the SDRmn register (optional)	Reset the register to change the transfer baud rate setting (setting the transfer clock by dividing the operation clock (fMCK)).
	<6>	Changing setting of the SMRmn register (optional)	Reset the register to change serial mode register mn (SMRmn) setting.
	<7>	Changing setting of the SCRmn register (optional)	Reset the register to change serial communication operation setting register mn (SCRmn) setting.
	<8>	Clearing error flag (optional)	If the SSRmn.OVF flag remains set, clear this using serial flag clear trigger register mn (SIRmn).
	<9>	Changing setting of the SOEm register (optional)	Set the SOEm.SOE[n] bit to 0 to stop output from the target channel.
	<10>	Changing setting of the SOm register	Set the initial output level of the serial data (SOm.SO[n]).
	<11>	Changing setting of the SOEm register	Set the SOEm.SOE[n] bit to 1 and enable output from the target channel.
	<12>	Port manipulation	Enable data output of the target channel.
	<13>	Writing to the SSm register	Set the SSm.SS[n] bit of the target channel to 1 and set SEm.SE[n] to 1 to enable operation.
	<14>	Starting communication	Sets transmit data to set the SEm.SE[n] bit and wait for a clock from the master.
	<15>	Completing resumption setting	—

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3)

(3) Processing flow (in single transmission mode)

Figure 19.16 shows the timing of slave transmission (in single transmission mode) (type 1: SCRmn.DCP[1:0] = 00b).

表 19.47 停止从属传输的程序

步		过程	细节
停止从属传动装置的程序	<1>	开始设置停止	—
	<2>	等待 SSRmn.TSF 位被清除（可选）	如果有数据传输，请等待传输完成。如果需要停车，不要等待。
	<3>	写入 STm 寄存器	将 1 写入目标通道的 STm.ST[n] 位，并将 SEm.SE[n] 位设置为 0 以停止操作。
	<4>	更改 SOEm 寄存器的设置	将 SOEm.SOE[n] 位设置为 0，停止目标通道的输出。
	<5>	更改 SOm 寄存器的设置（可选）	如果需要，可以更改目标通道上的串行数据(SOm.SO[n]) 的电平。
	<6>	停止设置完成	停止设置完成后，进行下一步处理。

注: m: 单元编号(m = 0) n: 通道编号(n = 0 至3)

表 19.48 显示了恢复从属传输的步骤。

表 19.48 恢复从属传输的步骤

步		过程	细节
恢复从属传输的步骤	<1>	恢复的初始设置	—
	<2>	等完成硕士准备工作后再做决定。	等待通信目标（主设备）停止或通信操作完成。
	<3>	端口操作	禁用目标通道的数据输出和时钟输出。
	<4>	更改 SPSm 寄存器的设置（可选）	重置寄存器以更改操作时钟设置。
	<5>	更改 SDRmn 寄存器的设置（可选）	重置寄存器以更改传输波特率设置（通过分频操作时钟 (fMCK) 来设置传输时钟）。
	<6>	更改 SMRmn 寄存器的设置（可选）	重置寄存器以更改串行模式寄存器 mn (SMRmn) 设置。
	<7>	更改 SCRmn 寄存器的设置（可选）	重置寄存器以更改串行通信操作设置寄存器 mn (SCRmn) 设置。
	<8>	清除错误标志（可选）	如果 SSRmn.OVF 标志仍然设置，则使用串行标志清除触发寄存器 mn (SIRmn) 清除此标志。
	<9>	更改 SOEm 寄存器的设置（可选）	将 SOEm.SOE[n] 位设置为 0 以停止目标通道的输出。
	<10>	更改 SOm 寄存器的设置	设置串行数据的初始输出级别 (SOm.SO[n]) 。
	<11>	更改 SOEm 寄存器的设置	将 SOEm.SOE[n] 位设置为 1，并启用目标通道的输出。
	<12>	端口操作	启用目标通道的数据输出。
	<13>	致函SSm登记处	将目标通道的 SSm.SS[n] 位设置为 1，并将 SEm.SE[n] 设置为 1 以启用操作。
	<14>	开始沟通	设置发送数据以设置 SEm.SE[n] 位，并等待来自主控的时钟。
	<15>	完成恢复设置	—

注: m: 单元编号(m = 0) n: 通道编号(n = 0 至3)

(3) 处理流程（单次传输模式）

图 19.16 显示了从属传输（在单次传输模式下）的时序（类型1: SCRmn.DCP[1:0] = 00b）。

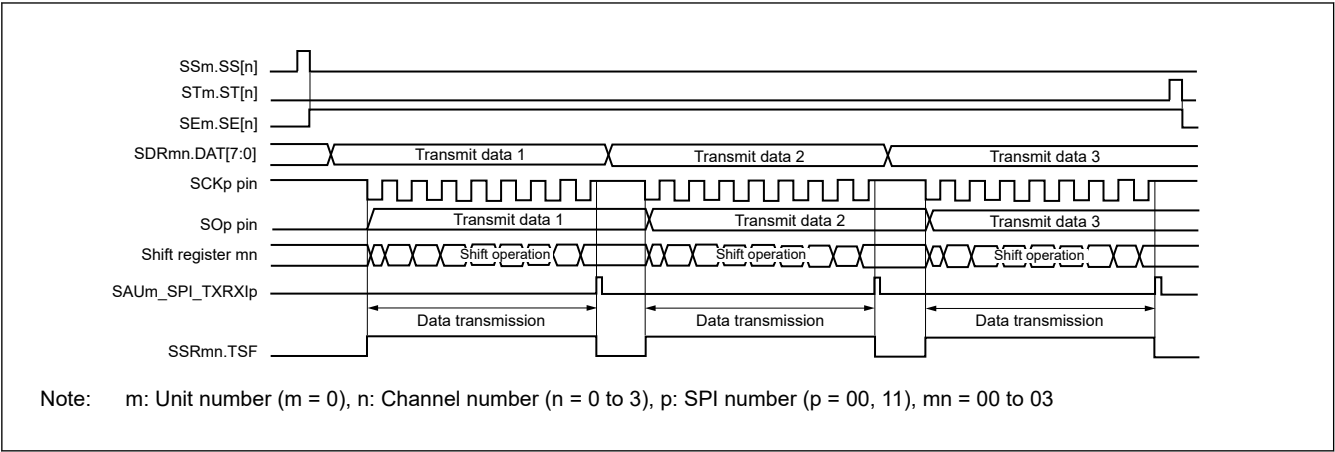


Figure 19.16 Timing of slave transmission (in single transmission mode) (type 1: SCRmn.DCP[1:0] = 00b)

Figure 19.17 shows the flowchart of slave transmission (in single transmission mode).

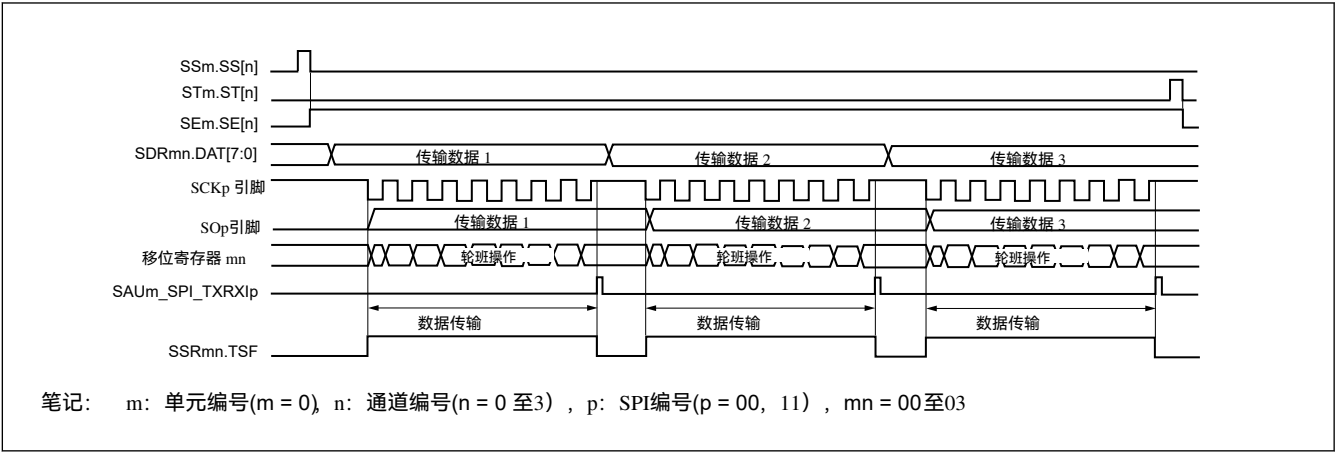


图 19.16 从设备传输的时序（单次传输模式）（类型 1: SCRmn.DCP[1:0] = 00b）图 19.17 显示了从设备传输的流程（单次传输模式）。

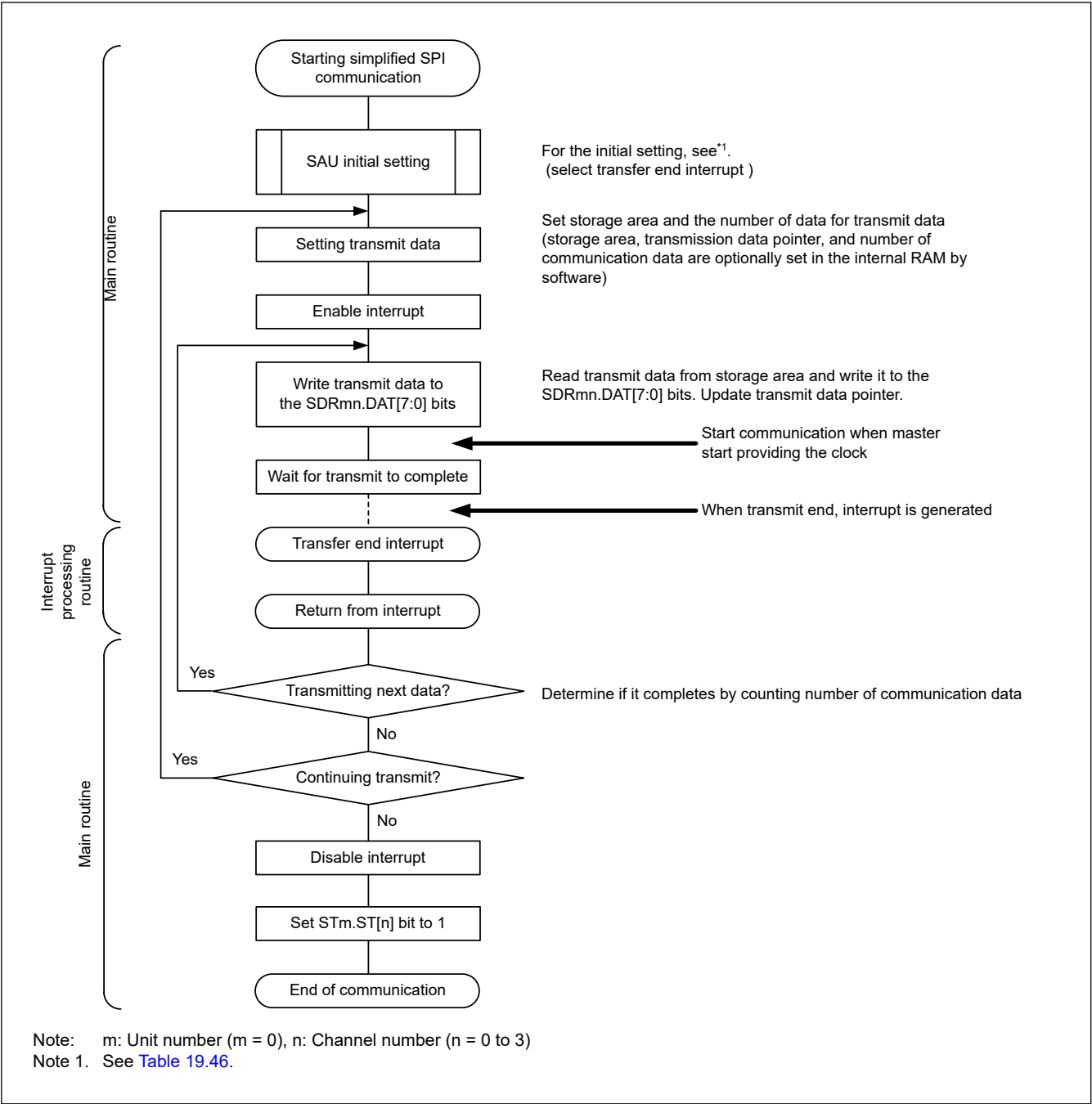


Figure 19.17 Flowchart of slave transmission (in single transmission mode)

(4) Processing flow (in continuous transmission mode)

Figure 19.18 shows the timing of slave transmission (in continuous transmission mode) (type 1: SCRmn.DCP[1:0] = 00b).

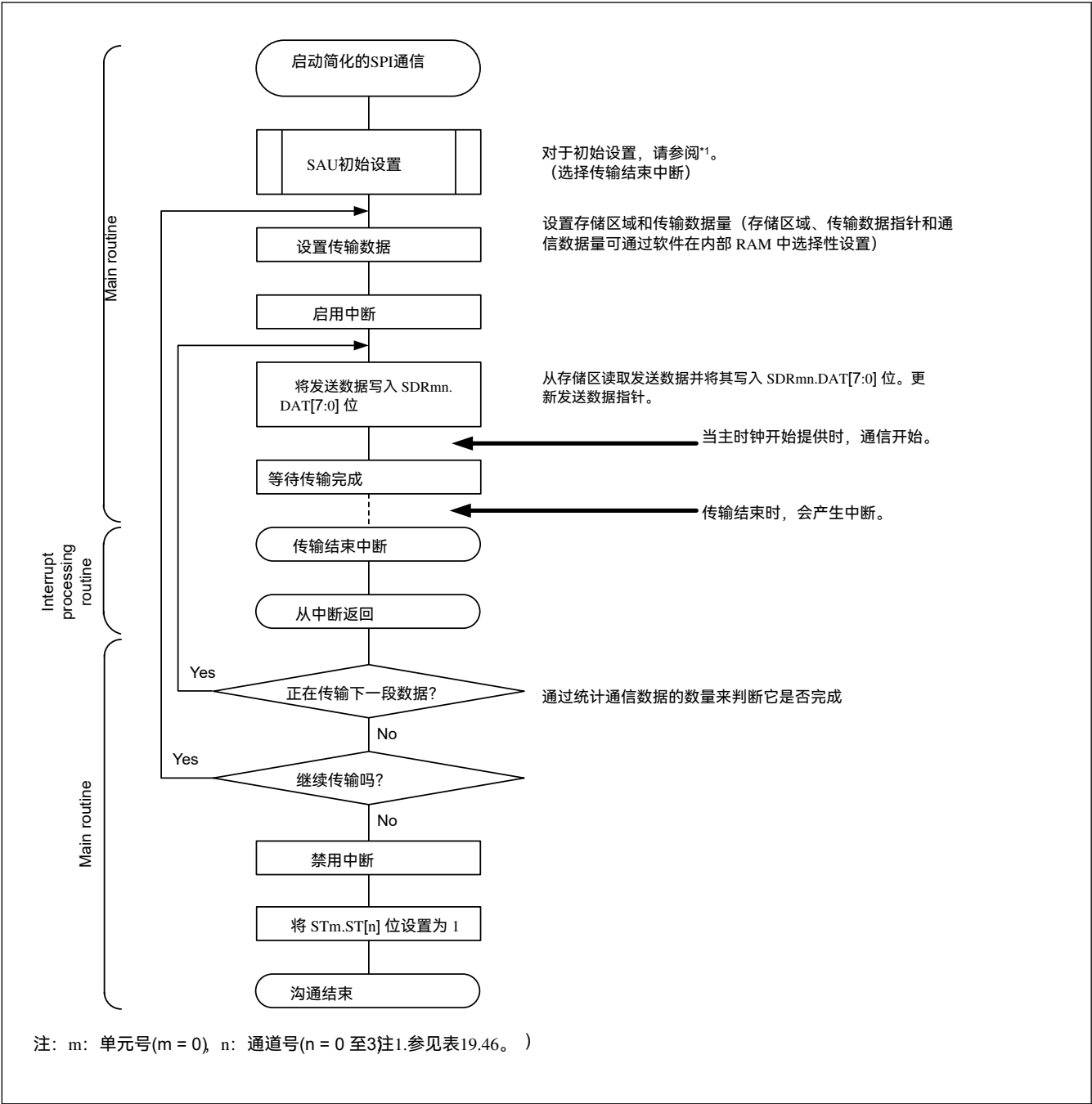


图 19.17 从设备传输流程图（单传输模式）

(4) 处理流程（连续传输模式）

图 19.18 显示了从设备传输的时序（在连续传输模式下）（类型 1: SCRmn.DCP[1:0] = 00b）。

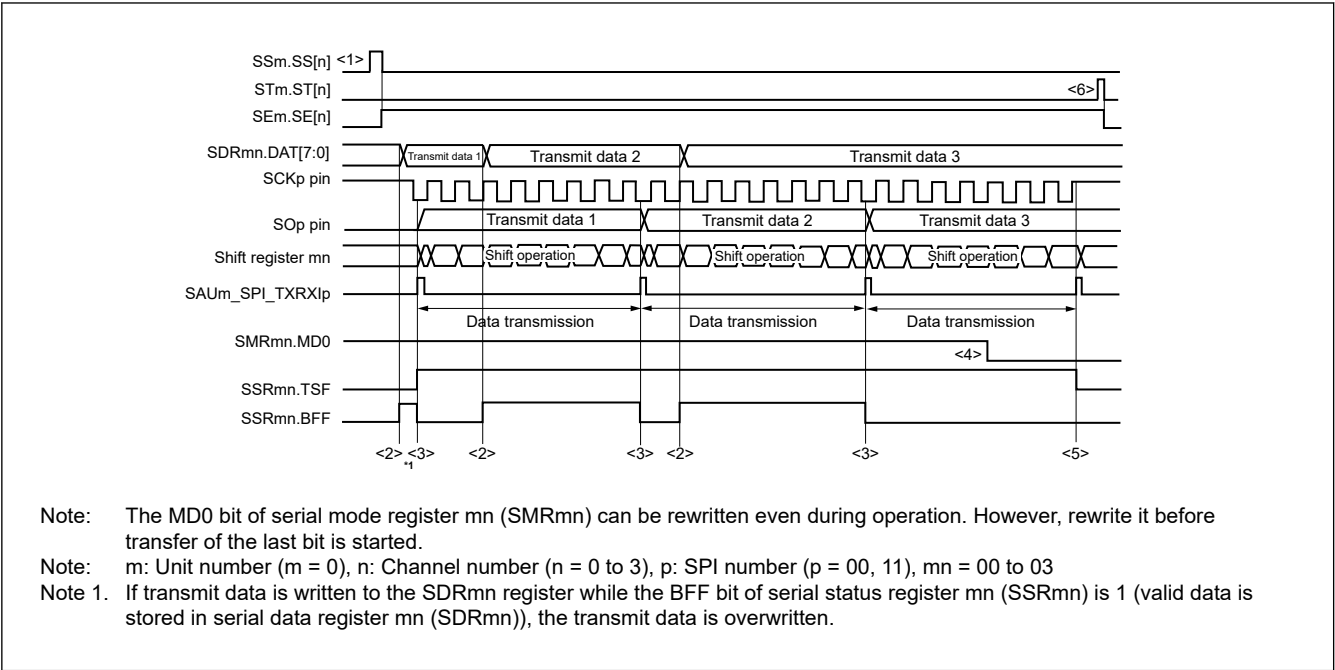


Figure 19.18 Timing of slave transmission (in continuous transmission mode) (type 1: SCRmn.DCP[1:0] = 00b)

Figure 19.19 shows the flowchart of slave transmission (in continuous transmission mode).

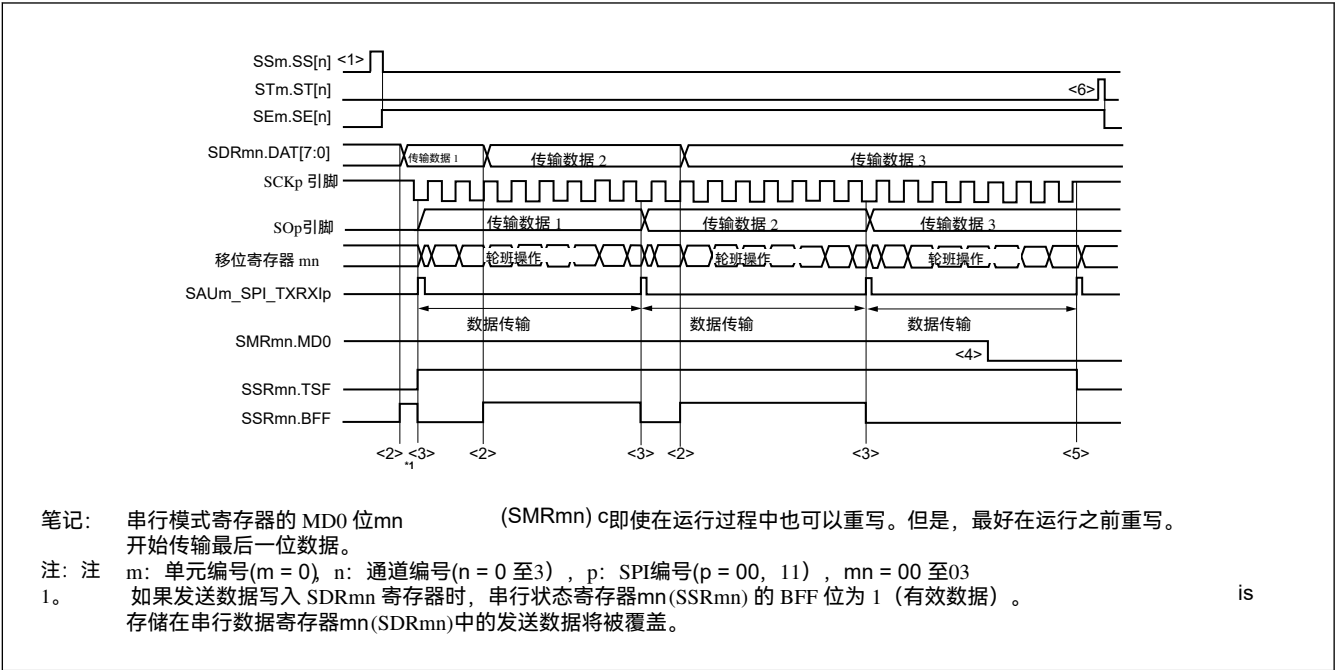


图 19.18 从设备传输时序 (连续传输模式) (类型 1: SCRmn.DCP[1:0] = 00b)

图 19.19 显示了从设备传输流程图 (连续传输模式)。

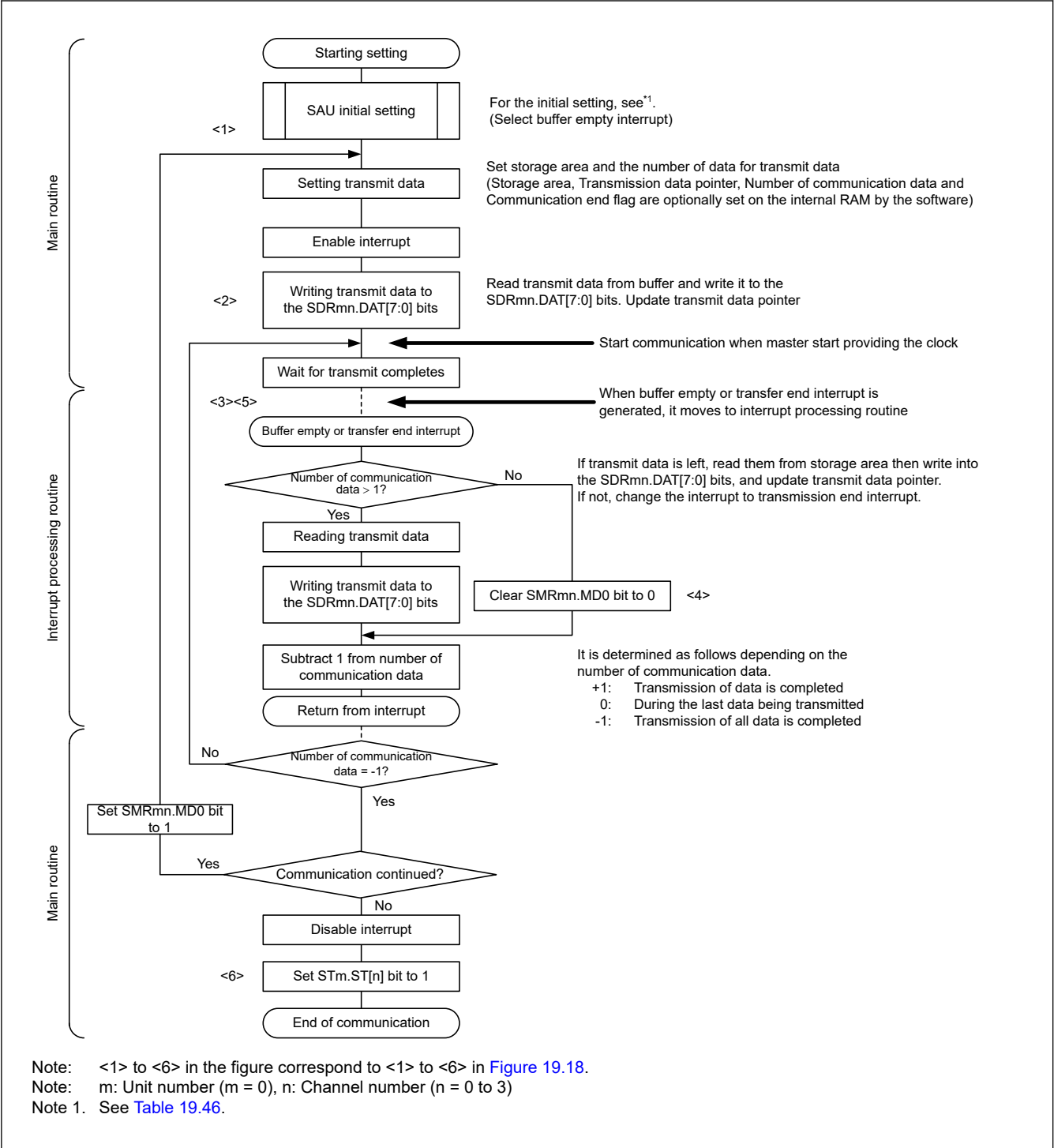


Figure 19.19 Flowchart of slave transmission (in continuous transmission mode)

19.5.5 Slave Reception

Slave reception is when a microcontroller receives data from another device in the state of a transfer clock being input from another device.

Table 19.49 shows the specification of slave reception of simplified SPI.

Table 19.49 Specification of slave reception of simplified SPI (1 of 2)

Simplified SPI	SPI00	SPI11
Target channel	Channel 0 of SAU0	Channel 3 of SAU0

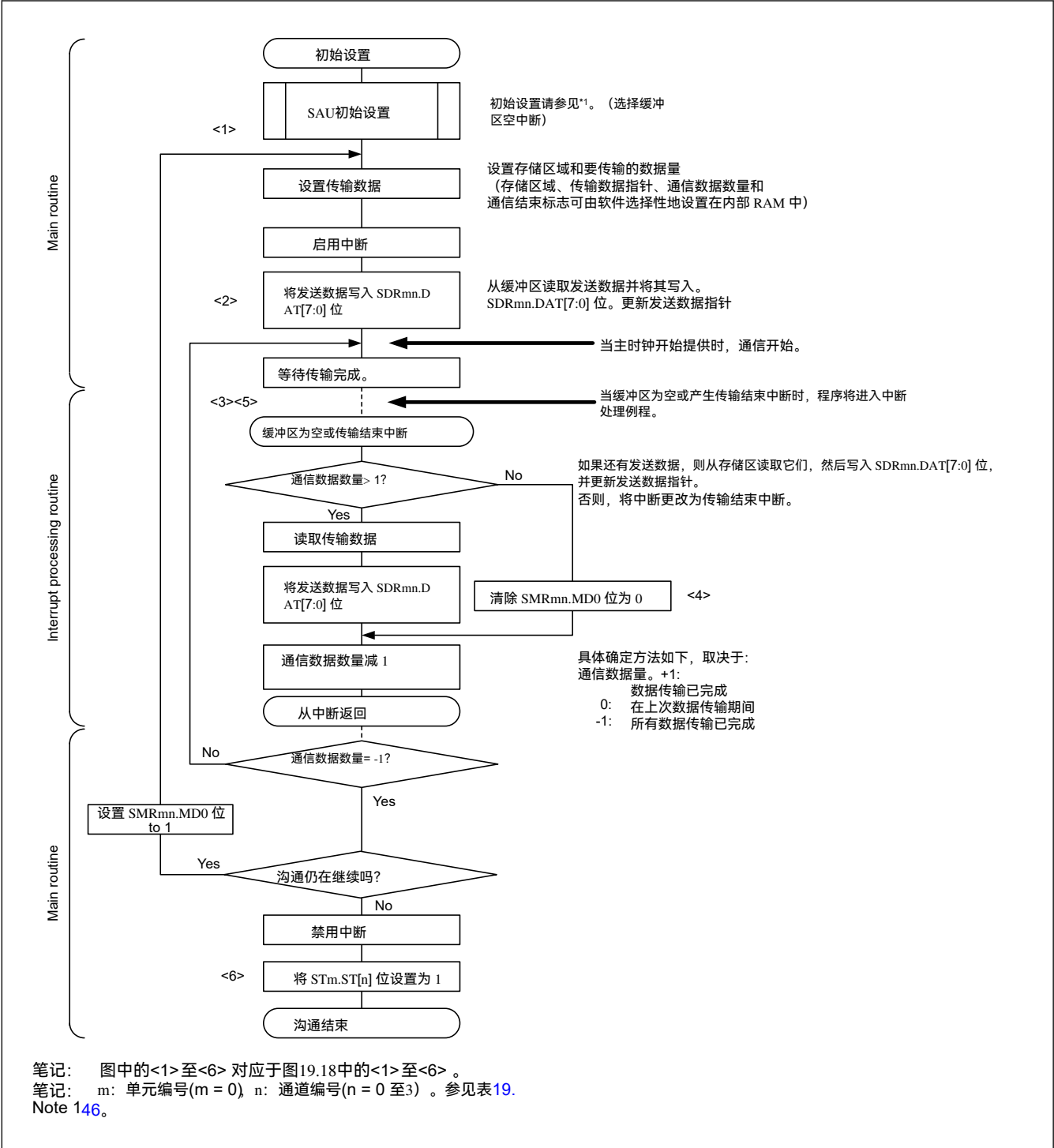


图 19.19 从站传输流程图（连续传输模式）

19.5.5 奴隶接收

从设备接收是指微控制器在传输时钟从另一设备输入的情况下，从另一设备接收数据。

表 19.49 显示了简化 SPI 从接收的规范。

表 19.49 简化 SPI 从设备接收规范（1/2）

简化的SPI	SPI00	SPI11
目标频道	SAU0 的 0 通道	SAU0 的 3 频道

Table 19.49 Specification of slave reception of simplified SPI (2 of 2)

Simplified SPI	SPI00	SPI11
Pins used	SCK00, SI00	SCK11, SI11
Interrupt	SAU0_SPI_TXRXI00	SAU0_SPI_TXRXI11
	Transfer end interrupt only (Setting the buffer empty interrupt is prohibited.)	
Error detection flag	Overrun error detection flag (SSRmn.OVF) only	
Transfer data length	7 or 8 bits	
Transfer rate	Max. $f_{MCK}/6$ [Hz] ^{*1 *2}	
Data phase	Selectable by the DCP[1] bit of the SCRmn register - DCP[1] = 0: Data input starts from the start of the operation of the serial clock. - DCP[1] = 1: Data input starts half a clock cycle before the start of the serial clock operation.	
Clock phase	Selectable by the DCP[0] bit of the SCRmn register - DCP[0] = 0: Non-reverse - DCP[0] = 1: Reverse	
Data direction	MSB or LSB first	

Note: f_{MCK} : Operation clock frequency of target channel
 f_{SCK} : Serial clock frequency

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3), mn = 00 to 03

Note 1. Because the external serial clock input to the SCK00, SCK11 pins is sampled internally and used, the fastest transfer rate is $f_{MCK}/6$ [Hz].

Note 2. Use this operation within a range that satisfies the conditions above and the peripheral functions characteristics specified in the electrical characteristics. For details, see [section 27, Electrical Characteristics](#) .

(1) Register setting

[Table 19.50](#) to [Table 19.55](#) show examples of the register contents for slave reception of simplified SPI.

(a) Serial mode register mn (SMRmn)

Table 19.50 Example of serial mode register mn (SMRmn) contents for slave reception of simplified SPI

Bit	Symbol	Set value	Function
0	MD0	0	Interrupt source of channel n 0: Transfer end interrupt
2:1	MD1[1:0]	00b	Setting of operation mode of channel n 0 0: Simplified SPI mode
5:3	—	100b	Setting disabled (set to the initial value)
6	SIS0	0	Setting is fixed in the simplified SPI mode
7	—	0	Setting disabled (set to the initial value)
8	STS	0	Selection of start trigger source 0: Only software trigger is valid (selected for simplified SPI, UART transmission, and simplified I ² C).
13:9	—	00000b	Setting disabled (set to the initial value)
14	CCS	1	Selection of transfer clock (f_{TCLK}) of channel n 1: Clock input f_{SCK} from the SCKp pin (slave transfer in simplified SPI mode)
15	CKS	0/1	Operation clock (f_{MCK}) of channel n 0: Prescaler output clock CKm0 set by the SPSm register 1: Prescaler output clock CKm1 set by the SPSm register

表 19.49 简化 SPI 从设备接收规范 (2/2)

简化的SPI	SPI00	SPI11
使用的引脚	SCK00, SI00	SCK11, SI11
打断	SAU0_SPI_TXRXI00	SAU0_SPI_TXRXI11
	仅传输结束中断（禁止设置缓冲区空中断）。	
错误检测标志	仅溢出错误检测标志 (SSRmn.OVF)	
传输数据长度	7 位或 8 位	
转账费率	最大值. $f_{MCK}/6$ [Hz] ^{*1 *2}	
数据阶段	可通过 SCRmn 寄存器的 DCP [1] 位进行选择 - DCP[1] = 0 ：数据输入从串行时钟开始运行时开始。 - DCP[1] = 1: 数据输入在串行时钟操作开始前半个时钟周期开始。	
时钟相位	可通过 SCRmn 寄存器的 DCP[0] 位进行选择 - DCP[0] = 0: 非反向 - DCP[0] = 1 ：反向	
数据方向	先 MSB 还是 LSB	

注: f_{MCK} : 目标通道的工作时钟频率
 f_{SCK} : 串行时钟频率

注: m: 单元编号(m = 0), n: 通道编号(n = 0 至3) , mn = 00至03

注1. 由于SCK00、SCK11引脚的外部串行时钟输入在内部进行采样和使用，因此最快的传输速率为 $f_{MCK}/6$ [Hz]。

注2: 此操作应在满足上述条件和电气特性中规定的外围功能特性的范围内使用。详情请参见第27节“电气特性”。

(1) 寄存器设置

表 19.50 至表 19.55 显示了简化 SPI 从设备接收寄存器内容的示例。

(a) 串行模式寄存器 mn (SMRmn)

表 19.50 简化 SPI 从设备接收串行模式寄存器 mn (SMRmn) 内容示例

少量	象征	设置值	功能
0	MD0	0	通道 n 的中断源 0: 传输结束中断
2:1	MD1[1:0]	00b	设置通道 n 的工作模式 0 0: 简化的 SPI 模式
5:3	—	100b	设置已禁用（已设置为初始值）
6	SIS0	0	简化 SPI 模式下的设置是固定的
7	—	0	设置已禁用（已设置为初始值）
8	STS	0	选择启动触发源 0: 仅软件触发有效（选择用于简化 SPI、UART 传输和简化I ² C）。
13:9	—	00000b	设置已禁用（已设置为初始值）
14	CCS	1	选择通道 n 的传输时钟 (f_{TCLK}) 1: 时钟输入 f_{SCK} 来自 SCKp 引脚（简化 SPI 模式下的从机传输）
15	CKS	0/1	通道 n 的运行时钟(f_{MCK}) 0: 预分频器输出时钟CKm0由 SPSm 寄存器设置 1: 预分频器输出时钟 CKm1 由 SPSm 寄存器设置

(b) Serial communication operation setting register mn (SCRmn)

Table 19.51 Example of serial communication operation setting register mn (SCRmn) contents for slave reception of simplified SPI

Bit	Symbol	Set value	Function
1:0	DLS [1:0]	10b or 11b	Setting of data length 1 0: 7-bit data length 1 1: 8-bit data length
3:2	—	01b	Setting disabled (set to the initial value)
5:4	SLC[1:0]	00b	Since this bit is dedicated to UART mode, it is fixed in the simplified SPI mode.
6	—	0	Setting disabled (set to the initial value)
7	DIR	0/1	Selection of data transfer sequence in simplified SPI and UART modes 0: Inputs or outputs data with MSB first. 1: Inputs or outputs data with LSB first.
9:8	PTC[1:0]	00b	Since this bit is dedicated to UART mode, it is fixed in the simplified SPI mode.
10	EOC	0	Since this bit is dedicated to UART receive modes, it is fixed in the simplified SPI mode.
11	—	0	Setting disabled (set to the initial value)
13:12	DCP[1:0]	00b to 11b	Selection of data and clock phase in simplified SPI mode For details about the setting, see section 19.3. Register Descriptions .
15:14	TRXE[1:0]	01b	Setting TRXE[1:0] = 01b is fixed in the simplified SPI slave reception mode

(c) Serial data register mn (SDRmn)

Read-only.

Table 19.52 Example of serial data register mn (SDRmn) contents for slave reception of simplified SPI

Bit	Symbol	Set value	Function
7:0	DAT[7:0]	0xFF	Receive data
8	DAT[8]	0	0 Fixed
15:9	STCLK[6:0]	0x00	Baud rate setting (Do not used in any slave mode)

(d) Serial output register m (SOM)

This register is not used in this mode.

Table 19.53 Example of serial output register m (SOM) contents for slave reception of simplified SPI

Bit	Symbol	Set value	Function
n	SO[n]	x	Bit that cannot be used in this mode (set to the initial value when not used in any mode)
n+8	CKO[n]	x	Bit that cannot be used in this mode (set to the initial value when not used in any mode)

(e) Serial output enable register m (SOEm)

This register is not used in this mode.

Table 19.54 Example of serial output enable register m (SOEm) contents for slave reception of simplified SPI

Bit	Symbol	Set value	Function
n	SOE[n]	x	Bit that cannot be used in this mode (set to the initial value when not used in any mode)

(f) Serial channel start register m (SSm)

Set only the bit of the target channel to 1.

(b) 串行通信操作设置寄存器 mn (SCRmn)

表 19.51 串行通信操作示例：设置寄存器 mn (SCRmn) 的内容，以便从机接收简化的 SPI 信号

少量	象征	设置值	功能
1:0	DLS [1:0]	10b 或 11b	数据长度设置：1.0： 7 位数 据长度；1.1： 8 位数据长度
3:2	—	01b	设置已禁用（已设置为初始值）
5:4	SLC[1:0]	00b	由于该位专用于 UART 模式，因此在简化的 SPI 模式下它是固定的。
6	—	0	设置已禁用（已设置为初始值）
7	DIR	0/1	简化 SPI 和 UART 模式下数据传输顺序的选择：0：以最高有效位 (MS B) 为先进行输入或输出数据。1：以最低有效位 (LSB) 为先进行输入或输出数据。
9:8	PTC[1:0]	00b	由于该位专用于 UART 模式，因此在简化的 SPI 模式下它是固定的。
10	EOC	0	由于该位专用于 UART 接收模式，因此在简化的 SPI 模式下它是固定的。
11	—	0	设置已禁用（已设置为初始值）
13:12	DCP[1:0]	00b 至 11b	简化 SPI 模式下的数据和时钟相位选择。有关设置的详细信息，请参阅第 1 9.3 节“寄存器描述”。
15:14	TRXE[1:0]	01b	在简化的 SPI 从机接收模式下，TRXE[1:0] = 01b 的设置是固定的。

(c) 串行数据寄存器 mn (SDRmn)

只读。

表 19.52 简化 SPI 从设备接收的串行数据寄存器 mn (SDRmn) 内容示例

少量	象征	设置值	功能
7:0	DAT[7:0]	0xFF	接收数据
8	DAT[8]	0	0 固定
15:9	STCLK[6:0]	0x00	波特率设置（请勿在任何从模式下使用）

(d) 串行输出寄存器 m (SOM)

此寄存器在此模式下不使用。

表 19.53 简化 SPI 从设备接收的串行输出寄存器m(SOM) 内容示例

少量	象征	设置值	功能
n	所以[n]	x	此模式下无法使用的位（在任何模式下未使用时，设置为初始值）
n+8	CKO[n]	x	此模式下无法使用的位（在任何模式下未使用时，设置为初始值）

(e) 串行输出使能寄存器 m (SOEm)

此寄存器在此模式下不使用。

表 19.54 简化从设备接收的串行输出使能寄存器m(SOEm) 内容示例
SPI

少量	象征	设置值	功能
n	SOE[n]	x	此模式下无法使用的位（在任何模式下未使用时，设置为初始值）

(f) 串行通道起始寄存器m (SSm)

仅将目标通道的位设置为 1。

Table 19.55 Example of serial channel start register m (SSm) contents for slave reception of simplified SPI

Bit	Symbol	Set value	Function
n	SS[n]	1	Operation start trigger of channel n 1: Set the SEm.SE[n] bit to 1 to place the channel in communication waiting state

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3), mn = 00 to 03

Note: ×: Bit that cannot be used in this mode (set to the initial value when not used in any mode)
0/1: Set to 0 or 1 depending on the usage of the user

(2) Operation procedure

Table 19.56 shows the procedure for initial setting of slave reception.

Table 19.56 Initial setting procedure for slave reception

Step		Process	Detail
Procedure for initial setting of slave reception	<1>	Starting initial setting	—
	<2>	Setting the SPSm register	Set the operation clock.
	<3>	Setting the SMRmn register	Set an operation mode.
	<4>	Setting the SCRmn register	Set a communication format.
	<5>	Setting the SDRmn register	Set the SDRmn.STCLK[6:0] bits to 0x00 for baud rate setting.
	<6>	Setting port	Enable data input and clock input of the target channel.
	<7>	Writing to the SSm register	Set the SSm.SS[n] bit of the target channel to 1 and set SEm.SE[n] bit to 1 to enable operation. Wait for a clock from the master.
	<8>	Completing initial setting	—

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3)

Table 19.57 shows the procedure for stopping slave reception.

Table 19.57 Procedure for stopping slave reception

Step		Process	Detail
Procedure for stopping slave reception	<1>	Start setting to stop	—
	<2>	Wait until the SSRmn.TSF bit is cleared (optional)	If there is any data being transferred, wait for their completion. If there is a requirement to stop, do not wait.
	<3>	Write the STm register	Write 1 to the STm.ST[n] bit of the target channel and set SEm.SE[n] = 0 to stop operation.
	<4>	Change setting of the SOEm register	Set the SOEm.SOE[n] bit to 0 and stop the output of the target channel.
	<5>	Stop setting is completed	After the stop setting is completed, go to the next processing.

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3)

Table 19.58 shows the procedure for resuming slave reception.

表 19.55 简化从设备接收的串行通道起始寄存器m (SSm) 内容示例 SPI

少量	象征	设置值	功能
n	SS[n]	1	通道 n 的操作启动触发 1: 将 SEm.SE[n] 位设置为1，使通道处于通信等待状态

注: m: 单元编号(m = 0) n: 通道编号(n = 0至3) , mn = 00至03

注意: ×: 此模式下不能使用的位（在任何模式下未使用时设置为初始值）
0/1: 根据用户使用情况设置为 0 或 1

(2) 操作流程

表 19.56 显示了从属接收的初始设置程序。

表 19.56 从属接收的初始设置程序

步		过程	细节
从属接收的初始设置程序	<1>	初始设置	—
	<2>	设置 SPSm 寄存器	设置运行时钟。
	<3>	设置 SMRmn 寄存器	设置操作模式。
	<4>	设置 SCRmn 寄存器	制定沟通模式。
	<5>	设置 SDRmn 寄存器	设置 SDRmn.STCLK[6:0] 位为0x00位，以设置波特率。
	<6>	设置端口	启用目标通道的数据输入和时钟输入。
	<7>	致函SSm登记处	将目标通道的 SSm.SS[n] 位设置为 1，并将 SEm.SE[n] 位设置为 1 以启用操作。等待主时钟信号。
	<8>	完成初始设置	—

注: m: 单元编号(m = 0) n: 通道编号(n = 0 至3)

表 19.57 显示了停止从属接收的步骤。

表 19.57 停止从属接收程序

步		过程	细节
程序停止奴隶接收	<1>	开始设置停止	—
	<2>	等待 SSRmn.TSF 位被清除（可选）	如果有数据传输，请等待传输完成。如果需要停车，不要等待。
	<3>	写入 STm 寄存器	将 1 写入目标通道的 STm.ST[n] 位，并设置SEm.SE[n] = 0 以停止操作。
	<4>	更改 SOEm 寄存器的设置	将 SOEm.SOE[n] 位设置为 0，停止目标通道的输出。
	<5>	停止设置完成	停止设置完成后，进行下一步处理。

注: m: 单元编号(m = 0) n: 通道编号(n = 0 至3)

表 19.58 显示了恢复从属接收的步骤。

Table 19.58 Procedure for resuming slave transmission

Step		Process	Detail
Procedure for resuming slave reception	<1>	Start setting for resumption	—
	<2>	Wait until completing master preparations	Wait until the communication target (master) stops or communication operation completed.
	<3>	Port manipulation	Disable data output and clock output of the target channel.
	<4>	Change setting of the SPSm register (optional)	Reset the register to change the operation clock setting.
	<5>	Change setting of the SMRmn register (optional)	Reset the register to change serial mode register mn (SMRmn) setting.
	<6>	Change setting of the SCRmn register (optional)	Reset the register to change serial communication operation setting register mn (SCRmn) setting.
	<7>	Clearing error flag (optional)	If the SSRmn.OVF flag remains set, clear this using serial flag clear trigger register mn (SIRmn).
	<8>	Port manipulation	Enable clock output of the target channel.
	<9>	Writing to the SSm register	Set the SSm.SS[n] bit of the target channel to 1 and set SEm.SE[n] bit = 1 to enable operation. Wait for a clock from the master.
	<10>	Completing resumption setting	—

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3)

(3) Processing flow (in single reception mode)

Figure 19.20 shows the timing of slave reception (in single reception mode) (type 1: SCRmn.DCP[1:0] = 00b).

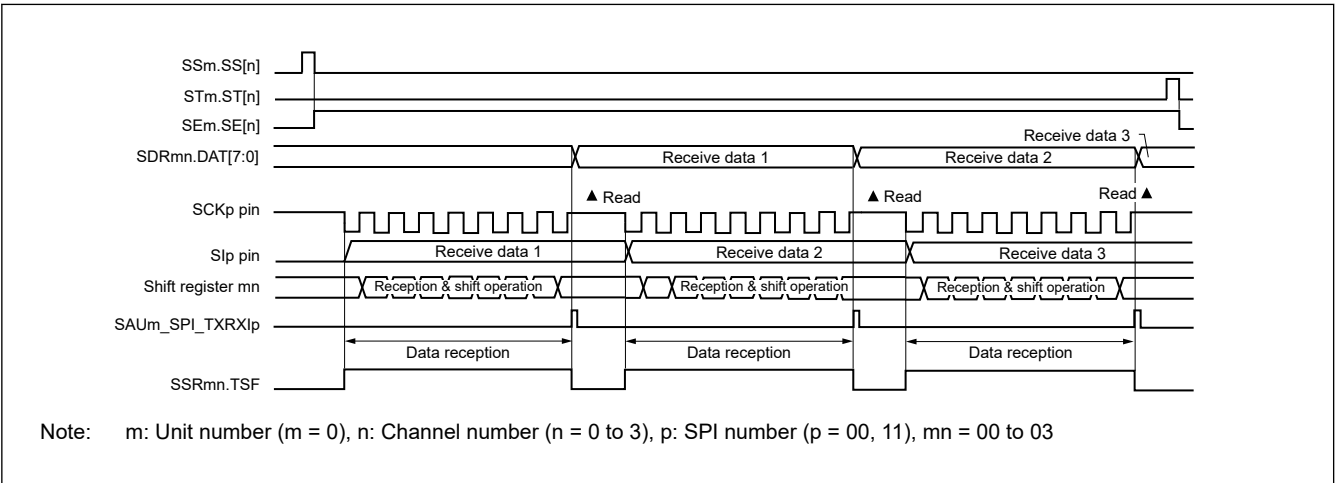


Figure 19.20 Timing of slave reception (in single reception mode) (type 1: SCRmn.DCP[1:0] = 00b)

Figure 19.21 shows the flowchart of slave reception (in single reception mode).

表 19.58 恢复从属传输的步骤

步		过程	细节
恢复从属接收程序	<1>	开始恢复设置	—
	<2>	等完成硕士准备工作后再做决定。	等待通信目标（主设备）停止或通信操作完成。
	<3>	端口操作	禁用目标通道的数据输出和时钟输出。
	<4>	更改 SPSm 寄存器的设置（可选）	重置寄存器以更改操作时钟设置。
	<5>	更改 SMRmn 寄存器的设置（可选）	重置寄存器以更改串行模式寄存器 mn (SMRmn) 设置。
	<6>	更改 SCRmn 寄存器的设置（可选）	重置寄存器以更改串行通信操作设置寄存器 mn (SCRmn) 设置。
	<7>	清除错误标志（可选）	如果 SSRmn.OVF 标志仍然设置，则使用串行标志清除触发寄存器 mn (SIRmn) 清除此标志。
	<8>	端口操作	启用目标通道的时钟输出。
	<9>	致函SSm登记处	将目标通道的 SSm.SS[n] 位设置为 1，并将 SEm.SE[n] 位= 1 设置为使能操作。等待主时钟信号。
	<10>	完成恢复设置	—

注: m: 单元编号(m = 0), n: 通道编号(n = 0 至3)

(3) 处理流程（单次接收模式）

图 19.20 显示了从接收的时序（在单接收模式下）（类型 1: SCRmn.DCP[1:0] = 00b）。

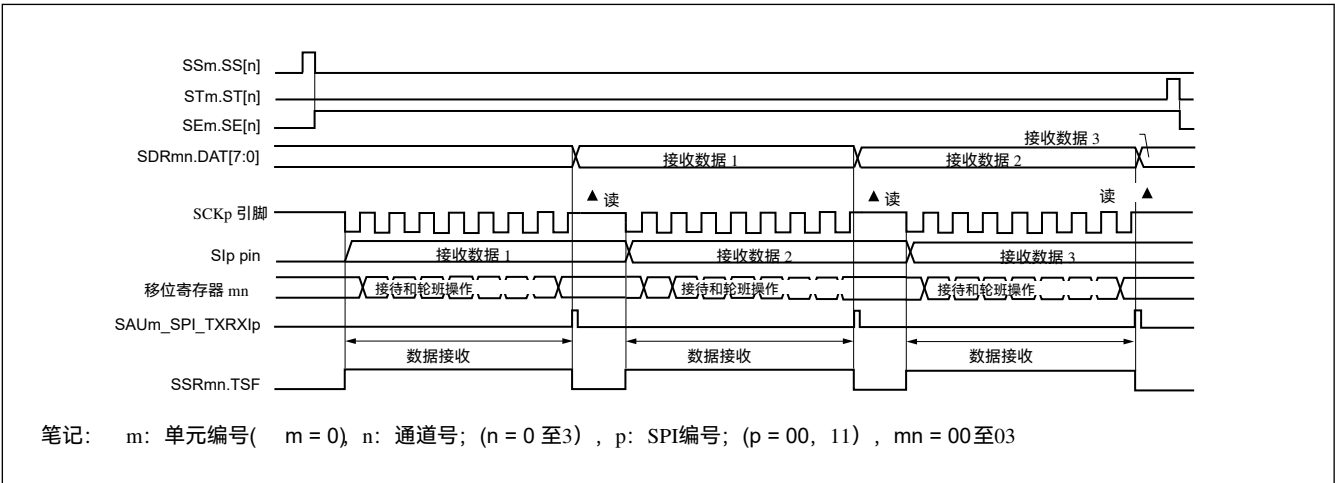


图 19.20 从接收时序（单接收模式）（类型 1: SCRmn.DCP[1:0] = 00b）

图 19.21 显示了从属接收（单接收模式）的流程图。

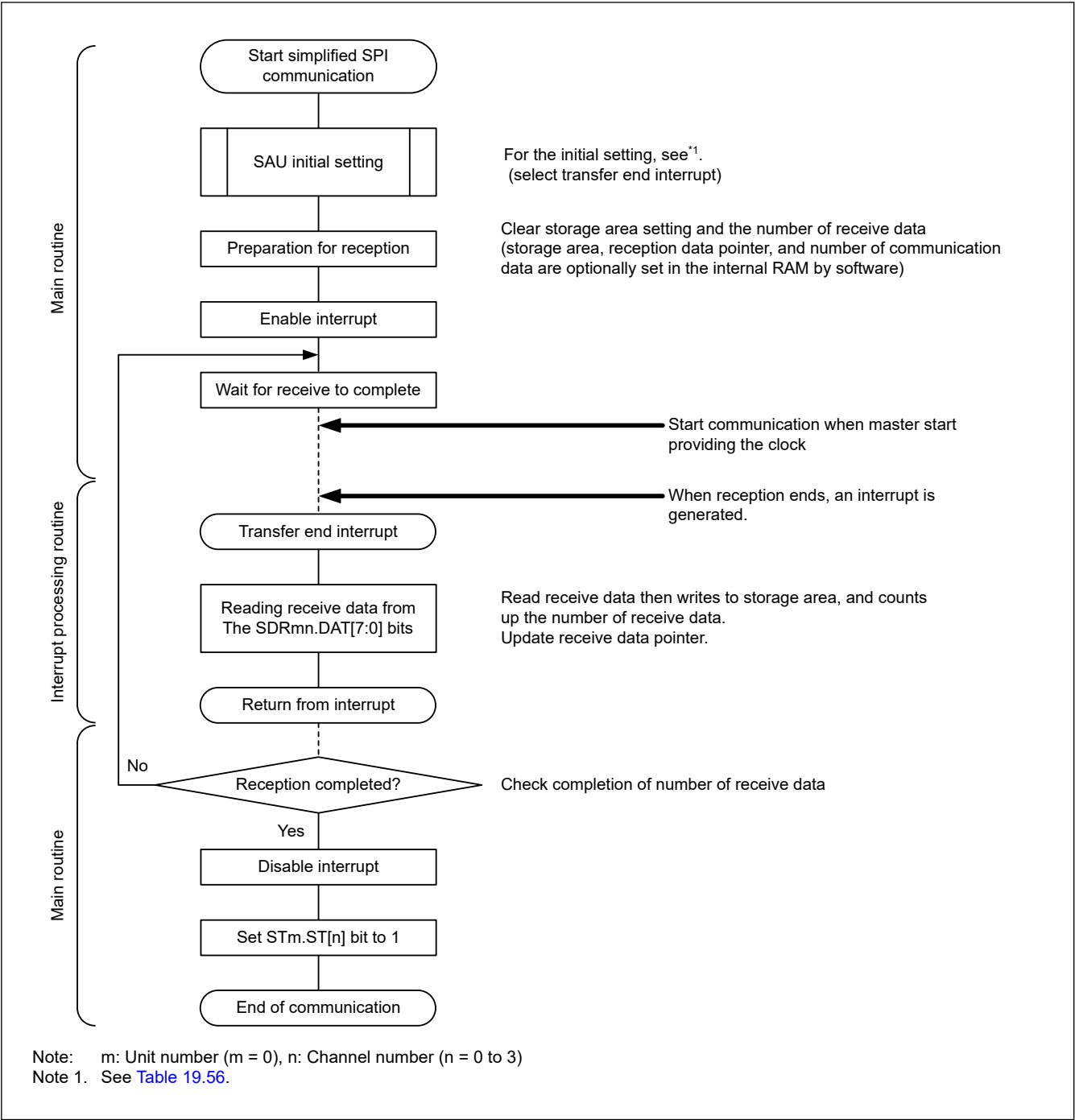


Figure 19.21 Flowchart of slave reception (in single reception mode)

19.5.6 Slave Transmission and Reception

Slave transmission and reception is when a microcontroller transmits and receives data to and from another device in the state of a transfer clock being input from another device.

Table 19.59 shows the specification of slave transmission and reception of simplified SPI.

Table 19.59 Specification of slave transmission and reception of simplified SPI (1 of 2)

Simplified SPI	SPI00	SPI11
Target channel	Channel 0 of SAU0	Channel 3 of SAU0
Pins used	SCK00, SI00, SO00	SCK11, SI11, SO11

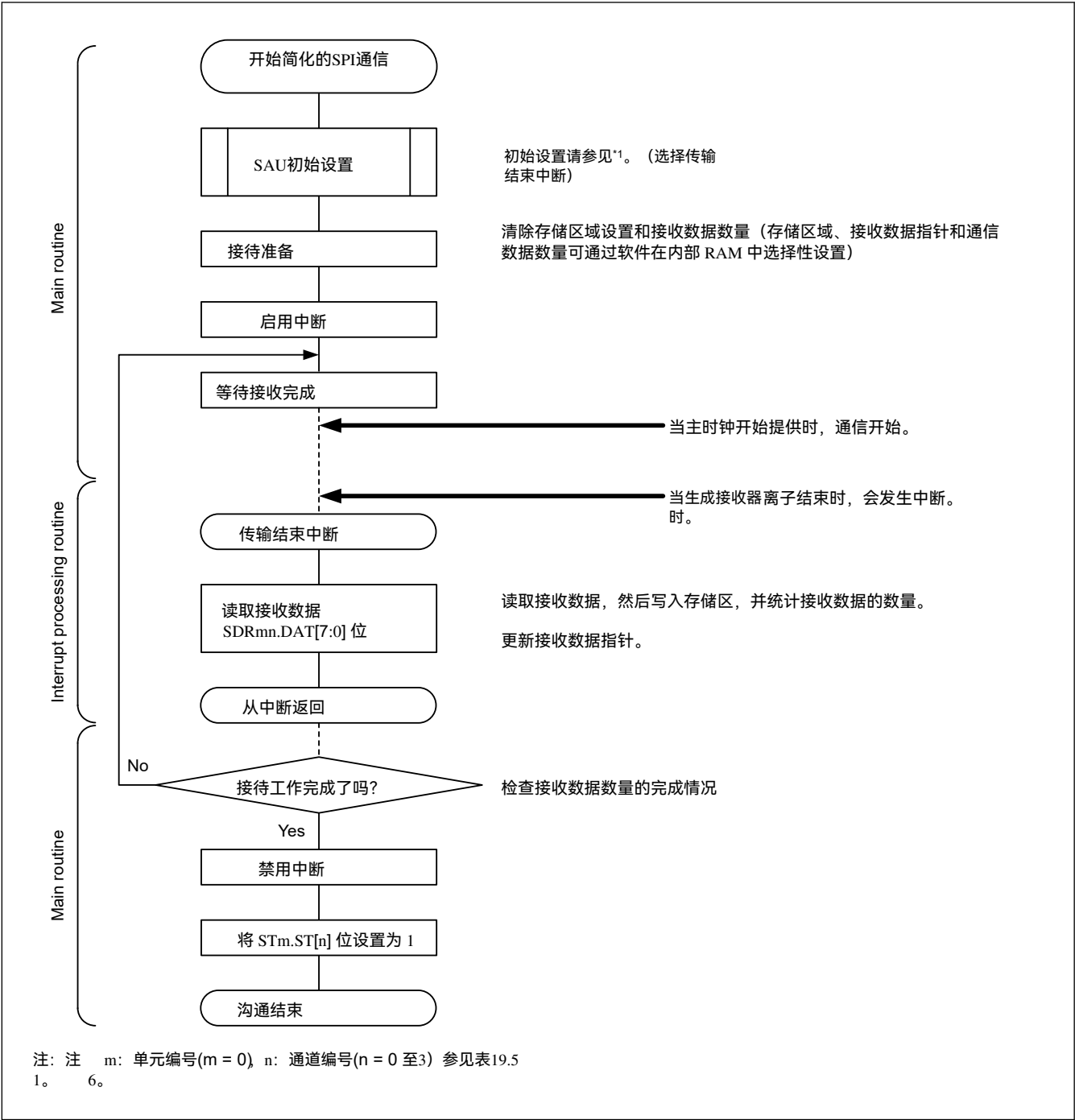


图 19.21 从设备接收流程图 (单接收模式)

19.5.6 从设备传输和接收

从设备传输和接收是指微控制器在从另一设备输入传输时钟的状态下, 向另一设备发送和接收数据。

表 19.59 显示了简化 SPI 的从站发送和接收规范。

表 19.59 简化 SPI 从站发送和接收规范 (1/2)

简化的SPI	SPI00	SPI11
目标频道	SAU0 的 0 通道	SAU0 的 3 频道
使用的引脚	SCK00、SI00、SO00	SCK11、SI11、SO11

Table 19.59 Specification of slave transmission and reception of simplified SPI (2 of 2)

Simplified SPI	SPI00	SPI11
Interrupt	SAU0_SPI_TXRXI00	SAU0_SPI_TXRXI11
	Transfer end interrupt (in single-transfer mode) or buffer empty interrupt (in continuous transfer mode) can be selected.	
Error detection flag	Overrun error detection flag (SSRmn.OVF) only	
Transfer data length	7 or 8 bits	
Transfer rate	Max. $f_{MCK}/6$ [Hz]*1 *2	
Data phase	Selectable by the DCP[1] bit of the SCRmn register - DCP[1] = 0: Data I/O starts at the start of the operation of the serial clock. - DCP[1] = 1: Data I/O starts half a clock cycle before the start of the serial clock operation.	
Clock phase	Selectable by the DCP[0] bit of the SCRmn register - DCP[0] = 0: Non-reverse - DCP[0] = 1: Reverse	
Data direction	MSB or LSB first	

Note: f_{MCK} : Operation clock frequency of target channel
 f_{SCK} : Serial clock frequency

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3), mn = 00 to 03

Note 1. Because the external serial clock input to the SCK00, SCK11 pins is sampled internally and used, the fastest transfer rate is $f_{MCK}/6$ [Hz].

Note 2. Use this operation within a range that satisfies the conditions above and the peripheral functions characteristics specified in the electrical characteristics. For details, see [section 27, Electrical Characteristics](#).

(1) Register setting

Table 19.60 to Table 19.65 show examples of the register contents for slave transmission and reception of simplified SPI.

(a) Serial mode register mn (SMRmn)

Table 19.60 Example of serial mode register mn (SMRmn) contents for slave transmission and reception of simplified SPI

Bit	Symbol	Set value	Function
0	MD0	0/1	Interrupt source of channel n 0: Transfer end interrupt 1: Buffer empty interrupt
2:1	MD1[1:0]	00b	Setting of operation mode of channel n 0 0: Simplified SPI mode
5:3	—	100b	Setting disabled (set to the initial value)
6	SIS0	0	Setting is fixed in the simplified SPI mode
7	—	0	Setting disabled (set to the initial value)
8	STS	0	Selection of start trigger source 0: Only software trigger is valid (selected for simplified SPI, UART transmission, and simplified I ² C).
13:9	—	00000b	Setting disabled (set to the initial value)
14	CCS	1	Selection of transfer clock (f_{TCLK}) of channel n 1: Clock input f_{SCK} from the SCKp pin (slave transfer in simplified SPI mode)
15	CKS	0/1	Operation clock (f_{MCK}) of channel n 0: Prescaler output clock CKm0 set by the SPSm register 1: Prescaler output clock CKm1 set by the SPSm register

表 19.59 简化 SPI 从站发送和接收规范 (2/2)

简化的SPI	SPI00	SPI11
打断	SAU0_SPI_TXRXI00	SAU0_SPI_TXRXI11
	可以选择传输结束中断（单次传输模式）或缓冲区空中断（连续传输模式）。	
错误检测标志	仅溢出错误检测标志 (SSRmn.OVF)	
传输数据长度	7 位或 8 位	
转账费率	最大限度。 $f_{MCK}/6$ [赫兹]*1*2	
数据阶段	可通过 SCRmn 寄存器的 DCP [1] 位进行选择 - DCP[1] = 0 : 数据I/O 从串行时钟开始运行时开始。 - DCP[1] = 1: 数据I/O在串行时钟操作开始前半个时钟周期开始。	
时钟相位	可通过 SCRmn 寄存器的 DCP[0] 位进行选择 - DCP[0] = 0 : 非反向 - DCP[0] = 1 : 反向	
数据方向	先 MSB 还是 LSB	

注: f_{MCK} : 目标通道的工作时钟频率
 f_{SCK} : 串行时钟频率

注: m: 单元编号(m = 0), n: 通道编号(n = 0 至3) , mn = 00至03

注1. 由于SCK00、SCK11引脚的外部串行时钟输入在内部进行采样和使用, 因此最快的传输速率为 $f_{MCK}/6$ [Hz]。

注2: 此操作应在满足上述条件和电气特性中规定的外围功能特性的范围内使用。详情请参见第27节“电气特性”。

(1) 寄存器设置

表 19.60 至表 19.65 显示了简化 SPI 从设备发送和接收的寄存器内容示例。

(a) 串行模式寄存器 mn (SMRmn)

表 19.60 简化 SPI 从设备发送和接收的串行模式寄存器 mn (SMRmn) 内容示例

少量	象征	设置值	功能
0	MD0	0/1	通道 n 的中断源 0: 传输结束中断 1: 缓冲区为空中断
2:1	MD1[1:0]	00b	设置通道 n 的工作模式 0 0: 简化的 SPI 模式
5:3	—	100b	设置已禁用（已设置为初始值）
6	SIS0	0	简化 SPI 模式下的设置是固定的
7	—	0	设置已禁用（已设置为初始值）
8	STS	0	选择启动触发源 0: 仅软件触发有效（选择用于简化 SPI、UART 传输和简化I ² C）。
13:9	—	00000b	设置已禁用（已设置为初始值）
14	CCS	1	选择通道 n 的传输时钟 (f_{TCLK}) 1: 时钟输入 f_{SCK} 来自 SCKp 引脚（简化 SPI 模式下的从机传输）
15	CKS	0/1	通道 n 的运行时钟(f_{MCK}) 0: 预分频器输出时钟CKm0由 SPSm 寄存器设置 1: 预分频器输出时钟 CKm1 由 SPSm 寄存器设置

(b) Serial communication operation setting register mn (SCRmn)

Table 19.61 Example of serial communication operation setting register mn (SCRmn) contents for slave transmission and reception of simplified SPI

Bit	Symbol	Set value	Function
1:0	DLS[1:0]	10b or 11b	Setting of data length 1 0: 7-bit data length 1 1: 8-bit data length
3:2	—	01b	Setting disabled (set to the initial value)
5:4	SLC[1:0]	00b	Since this bit is dedicated to UART mode, it is fixed in the simplified SPI mode.
6	—	0	Setting disabled (set to the initial value)
7	DIR	0/1	Selection of data transfer sequence in simplified SPI and UART modes 0: Inputs or outputs data with MSB first 1: Inputs or outputs data with LSB first
9:8	PTC[1:0]	00b	Since this bit is dedicated to UART mode, it is fixed in the simplified SPI mode.
10	EOC	0	Since this bit is dedicated to UART receive modes, it is fixed in the simplified SPI mode.
11	—	0	Setting disabled (set to the initial value)
13:12	DCP[1:0]	00b to 11b	Selection of data and clock phase in simplified SPI mode. For details about the setting, see section 19.3. Register Descriptions .
15:14	TRXE[1:0]	11b	Setting TRXE[1:0] = 11b is fixed in the simplified SPI master transmission and reception mode

(c) Serial data register mn (SDRmn)

Table 19.62 Example of serial data register mn (SDRmn) contents for slave transmission and reception of simplified SPI

Bit	Symbol	Set value	Function
7:0	DAT[7:0]	0xFF	Transmit data or Receive data (Transmit data setting and receive data read)
8	DAT[8]	0	0 Fixed
15:9	STCLK[6:0]	0x00	Baud rate setting (do not used in any slave mode)

(d) Serial output register m (SOM)

Set only the bit of the target channel.

Table 19.63 Example of serial output register m (SOM) contents for slave transmission and reception of simplified SPI

Bit	Symbol	Set value	Function
n	SO[n]	0/1	Serial data output of channel n (n = 0 to 3) 0: Serial data output value is 0 1: Serial data output value is 1
n+8	CKO[n]	x	Bit that cannot be used in this mode (set to the initial value when not used in any mode)

(e) Serial output enable register m (SOEm)

Set only the bit of the target channel to 1.

(b) 串行通信操作设置寄存器 mn (SCRmn)

表 19.61 串行通信操作示例：设置寄存器 mn (SCRmn) 的内容，用于简化 SPI 的从机发送和接收

少量	象征	设置值	功能
1:0	DLS[1:0]	10b 或 11b	数据长度设置：1.0： 7 位数 据长度；1.1： 8 位数数据长度
3:2	—	01b	设置已禁用（已设置为初始值）
5:4	SLC[1:0]	00b	由于该位专用于 UART 模式，因此在简化的 SPI 模式下它是固定的。
6	—	0	设置已禁用（已设置为初始值）
7	DIR	0/1	简化 SPI 和 UART 模式下数据传输顺序的选择 0： 输入或输出数据，最高有效位在前。 1： 输入或输出数据以最低有效位 (LSB) 为先。
9:8	PTC[1:0]	00b	由于该位专用于 UART 模式，因此在简化的 SPI 模式下它是固定的。
10	EOC	0	由于该位专用于 UART 接收模式，因此在简化的 SPI 模式下它是固定的。
11	—	0	设置已禁用（已设置为初始值）
13:12	DCP[1:0]	00b 至 11b	简化SPI模式下的数据和时钟相位选择。 有关设置的详细信息，请参阅第 19.3 节“寄存器描述”。
15:14	TRXE[1:0]	11b	在简化的 SPI 主控收发模式下，设置 TRXE[1:0] = 11b 是固定的

(c) 串行数据寄存器 mn (SDRmn)

表 19.62 简化 SPI 从设备发送和接收的串行数据寄存器mn(SDRmn) 内容示例

少量	象征	设置值	功能
7:0	DAT[7:0]	0xFF	发送数据或接收数据 (发送数据设置并接收数据读取)
8	DAT[8]	0	0 固定
15:9	STCLK[6:0]	0x00	波特率设置（请勿在任何从模式下使用）

(d) 串行输出寄存器 m (SOM)

仅设置目标通道的位。

表 19.63 简化 SPI 从设备发送和接收的串行输出寄存器m(SOM) 内容示例

少量	象征	设置值	功能
n	所以[n]	0/1	通道 n(n = 0 至 3 的串行数据输出) 0： 串口数据输出值为 0 1： 串口数据输出值为 1
n+8	CKO[n]	x	此模式下无法使用的位（在任何模式下未使用时，设置为初始值）

(e) 串行输出使能寄存器m (SOEm)

仅将目标通道的位设置为 1。

Table 19.64 Example of serial output enable register m (SOEm) contents for slave transmission and reception of simplified SPI

Bit	Symbol	Set value	Function
n	SOE[n]	1	Serial output enable or stop of channel n 1: Enable output by serial communication operation.

(f) Serial channel start register m (SSm)

Set only the bit of the target channel to 1.

Table 19.65 Example of serial channel start register m (SSm) contents for slave transmission and reception of simplified SPI

Bit	Symbol	Set value	Function
n	SS[n]	1	Operation start trigger of channel n 1: Set the SEm.SE[n] bit to 1 to place the channel in the communications waiting state.

Note: Be sure to set transmit data to the SDRmn.DAT[7:0] bits before the clock from the master is started.

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3), mn = 00 to 03

Note: ×: Bit that cannot be used in this mode (set to the initial value when not used in any mode)
0/1: Set to 0 or 1 depending on the usage of the user

(2) Operation procedure

Table 19.66 shows the procedure for initial setting of slave transmission and reception.

Table 19.66 Initial setting procedure for slave transmission and reception

Step		Process	Detail
Procedure for initial setting of slave transmission and reception	<1>	Starting initial setting	—
	<2>	Setting the SPSm register	Set the operation clock.
	<3>	Setting the SMRmn register	Set an operation mode.
	<4>	Setting the SCRmn register	Set a communication format.
	<5>	Setting the SDRmn register	Set the SDRmn.STCLK[6:0] bits to 0x00 for baud rate setting.
	<6>	Setting the SOm register	Set the initial output level of the serial data (SOm.SO[n]).
	<7>	Changing setting of the SOEm register	Set the SOEm.SOE[n] bit to 1 and enable data output of the target channel.
	<8>	Setting port	Enable data output of the target channel
	<9>	Writing to the SSm register	Set the SSm.SS[n] bit of the target channel to 1 and set SEm.SE[n] bit = 1 to enable operation.
	<10>	Completing initial setting	Initial setting is completed. Set transmit data to the SDRmn.DAT[7:0] bits and wait for a clock from the master

Note: Be sure to set transmit data to the SDRmn.DAT[7:0] bits before the clock from the master is started.

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3)

Table 19.67 shows the procedure for stopping slave transmission and reception.

表 19.64 简化 SPI 从设备发送和接收的串行输出使能寄存器m(SOEm) 内容示例

少量	象征	设置值	功能
n	SOE[n]	1	通道 n 的串行输出启用或停止 1: 通过串行通信操作启用输出。

(f) 串行通道起始寄存器m (SSm)

仅将目标通道的位设置为 1。

表 19.65 简化 SPI 从设备发送和接收的串行通道起始寄存器m (SSm) 内容示例

少量	象征	设置值	功能
n	SS[n]	1	通道 n 的操作启动触发 1: 将 SEm.SE[n] 位设置为 1，将信道置于通信等待状态状态。

注意：在主时钟启动之前，务必将发送数据设置为 SDRmn.DAT[7:0] 位。

注：m：单元编号(m = 0)，n：通道编号(n = 0至3)，mn = 00至03

注意：×: 此模式下不能使用的位（在任何模式下未使用时设置为初始值）
0/1: 根据用户使用情况设置为 0 或 1

(2) 操作流程

表 19.66 显示了从属传输和接收的初始设置程序。

表 19.66 从设备收发器的初始设置程序

步		过程	细节
从属传输和接收的初始设置程序	<1>	初始设置	—
	<2>	设置 SPSm 寄存器	设置运行时钟。
	<3>	设置 SMRmn 寄存器	设置操作模式。
	<4>	设置 SCRmn 寄存器	制定沟通模式。
	<5>	设置 SDRmn 寄存器	设置 SDRmn.STCLK[6:0] 位为0x00位，以设置波特率。
	<6>	设置 SOm 寄存器	设置串行数据的初始输出电平(SOm.SO[n])。
	<7>	更改 SOEm 寄存器的设置	将 SOEm.SOE[n] 位设置为 1，使能目标通道的数据输出。
	<8>	设置端口	启用目标通道的数据输出
	<9>	致函SSm登记处	将目标通道的 SSm.SS[n] 位设置为 1，并将 SEm.SE[n] 位= 1 设置为启用操作。
	<10>	完成初始设置	初始设置完成。 将发送数据设置为 SDRmn.DAT[7:0] 位，并等待主时钟信号。

注意：在主时钟启动之前，务必将发送数据设置为 SDRmn.DAT[7:0] 位。

注：m：单元编号(m = 0)，n：通道编号(n = 0 至3)

表 19.67 显示了停止从属传输和接收的程序。

Table 19.67 Procedure for stopping slave transmission and reception

Step		Process	Detail
Procedure for stopping slave transmission and reception	<1>	Starting setting to stop	—
	<2>	Wait until SSRmn.TSF is cleared (optional)	If there is any data being transferred, wait for their completion. If there is a requirement to stop, do not wait.
	<3>	Writing the STm register	Write 1 to the STm.ST[n] bit of the target channel and set SEm.SE[n] = 0 to stop operation.
	<4>	Changing setting of the SOEm register	Set the SOEm.SOE[n] bit to 0 and stop the output of the target channel.
	<5>	Changing setting of the SOm register (optional)	The levels of the serial data (SOm.SO[n]) on the target channel can be changed if required.
	<6>	Stop setting is completed	After the stop setting is completed, go to the next processing.

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3)

Table 19.68 shows the procedure for resuming slave transmission and reception.

Table 19.68 Procedure for resuming slave transmission and reception

Step		Process	Detail
Procedure for resuming slave transmission and reception	<1>	Starting setting for resumption	—
	<2>	Wait until completing master preparations	Wait until the communication target (master) stops or communication operation completed.
	<3>	Port manipulation	Disable data output of the target channel.
	<4>	Changing setting of the SPSm register (optional)	Reset the register to change the operation clock setting.
	<5>	Changing setting of the SMRmn register (optional)	Reset the register to change serial mode register mn (SMRmn) setting.
	<6>	Changing setting of the SCRmn register (optional)	Reset the register to change serial communication operation setting register mn (SCRmn) setting.
	<7>	Clearing error flag (optional)	If the SSRmn.OVF bit remains set, clear this using serial flag clear trigger register mn (SIRmn).
	<8>	Changing setting of the SOEm register (optional)	Set the SOEm.SOE[n] bit to 0 to stop output from the target channel.
	<9>	Changing setting of the SOm register (optional)	Set the initial output level of the serial data (SOm.SO[n]).
	<10>	Changing setting of the SOEm register (optional)	Set the SOEm.SOE[n] bit to 1 and enable output from the target channel.
	<11>	Port manipulation	Enable data output of the target channel.
	<12>	Writing to the SSm register	Set the SSm.SS[n] bit of the target channel to 1 and set SEm.SE[n] = 1 to enable operation.
	<13>	Starting communication	Set transmit data to the SDRmn.DAT[7:0] bits and wait for a clock from the master
	<14>	Completing resumption setting	—

Note: Be sure to set transmit data to the SDRmn.DAT[7:0] bits before the clock from the master is started.

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3)

(3) Processing flow (in single transmission and reception mode)

Figure 19.22 shows the timing of slave transmission and reception (in single transmission and reception mode) (type 1: SCRmn.DCP[1:0] = 00b).

表 19.67 停止从属传输和接收的程序

步		过程	细节
停止从属传输和接收的程序	<1>	开始设置停止	—
	<2>	等待 SSRmn.TSF 被清除（可选）	如果有数据传输，请等待传输完成。如果需要停车，不要等待。
	<3>	写入 STm 寄存器	将 1 写入目标信道的 STm.ST[n] 位，并将 SEm.SE[n] = 0 位设置为停止操作。
	<4>	更改 SOEm 寄存器的设置	将 SOEm.SOE[n] 位设置为 0，停止目标通道的输出。
	<5>	更改 SOm 寄存器的设置（可选）	如果需要，可以更改目标通道上的串行数据(SOm.SO[n]) 的电平。
	<6>	停止设置完成	停止设置完成后，进行下一步处理。

注: m: 单元编号(m = 0) n: 通道编号(n = 0 至3)

表 19.68 显示了恢复从属传输和接收的步骤。

表 19.68 恢复从属传输和接收的步骤

步		过程	细节
恢复从属传输和接收的程序	<1>	恢复的初始设置	—
	<2>	等完成硕士准备工作后再做决定。	等待通信目标（主设备）停止或通信操作完成。
	<3>	端口操作	禁用目标通道的数据输出。
	<4>	更改 SPSm 寄存器的设置（可选）	重置寄存器以更改操作时钟设置。
	<5>	更改 SMRmn 寄存器的设置（可选）	重置寄存器以更改串行模式寄存器 mn (SMRmn) 设置。
	<6>	更改 SCRmn 寄存器的设置（可选）	重置寄存器以更改串行通信操作设置寄存器 mn (SCRmn) 设置。
	<7>	清除错误标志（可选）	如果 SSRmn.OVF 位保持置位，则使用串行标志清除触发寄存器 m n (SIRmn) 清除此位。
	<8>	更改 SOEm 寄存器的设置（可选）	将 SOEm.SOE[n] 位设置为 0 以停止目标通道的输出。
	<9>	更改 SOm 寄存器的设置（可选）	设置串行数据的初始输出级别（SOm.SO[n]）。
	<10>	更改 SOEm 寄存器的设置（可选）	将 SOEm.SOE[n] 位设置为 1，并启用目标通道的输出。
	<11>	端口操作	启用目标通道的数据输出。
	<12>	致函SSm登记处	将目标通道的 SSm.SS[n] 位设置为 1，并将SEm.SE[n] = 1 设置为启用操作。
	<13>	开始沟通	将发送数据设置为 SDRmn.DAT[7:0] 位，并等待主时钟信号。
	<14>	完成恢复设置	—

注意: 在主时钟启动之前，务必将发送数据设置为 SDRmn.DAT[7:0] 位。

注: m: 单元编号(m = 0) n: 通道编号(n = 0 至3)

(3) 处理流程（单次发送和接收模式）

图 19.22 显示了从设备传输和接收的时序（在单次传输和接收模式下）（类型 1: SCRmn.DCP[1:0] = 00b）。

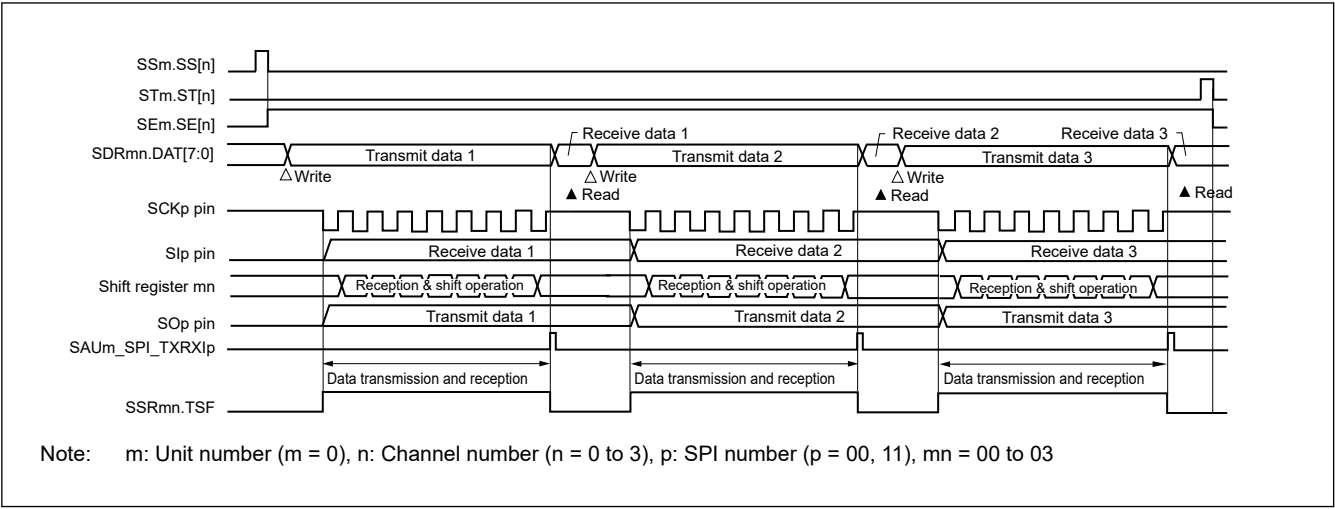


Figure 19.22 Timing of slave transmission and reception (in single transmission and reception mode)

Figure 19.23 shows the flowchart of slave transmission and reception (in single transmission and reception mode).

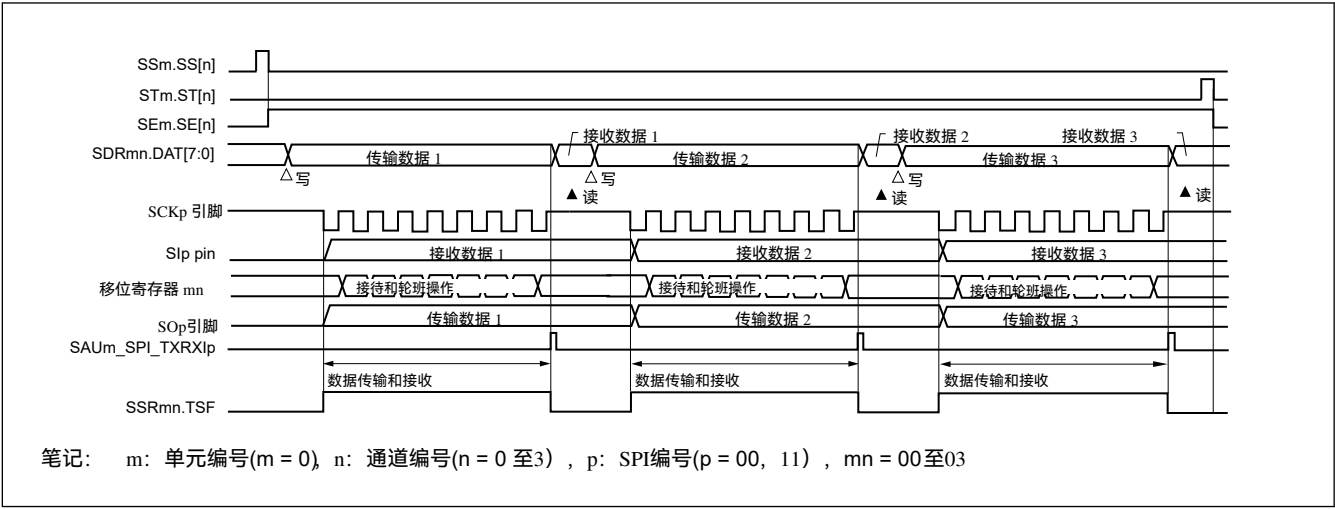


图 19.22 从设备发送和接收的时序（单次发送和接收模式）

图 19.23 显示了从设备传输和接收的流程图（在单次传输和接收模式下）。

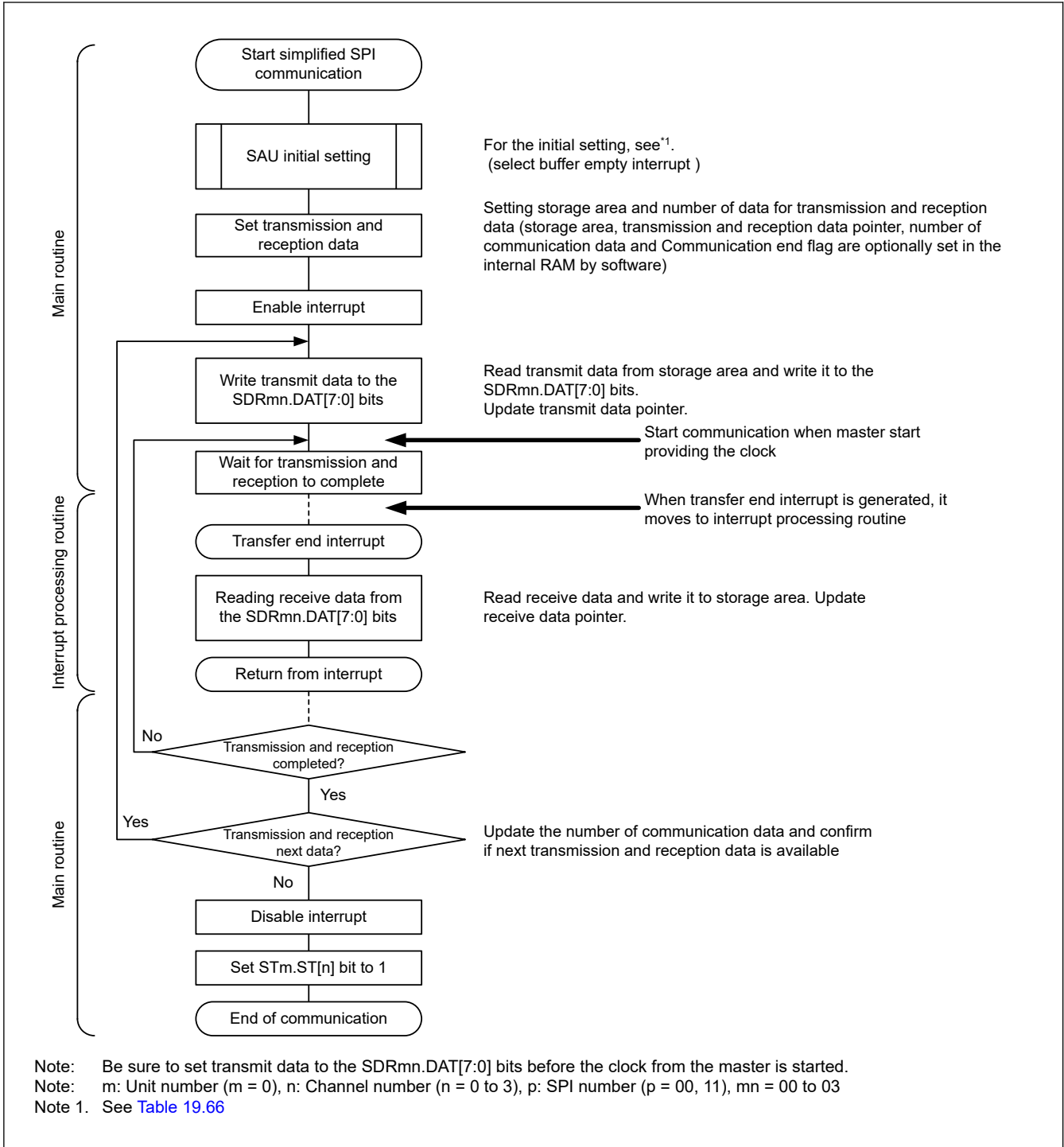


Figure 19.23 Flowchart of slave transmission and reception (in single transmission and reception mode)

(4) Processing flow (in continuous transmission and reception mode)

Figure 19.24 shows the timing of slave transmission and reception (in continuous transmission and reception mode) (type 1: SCRmn.DCP[1:0] = 00b).

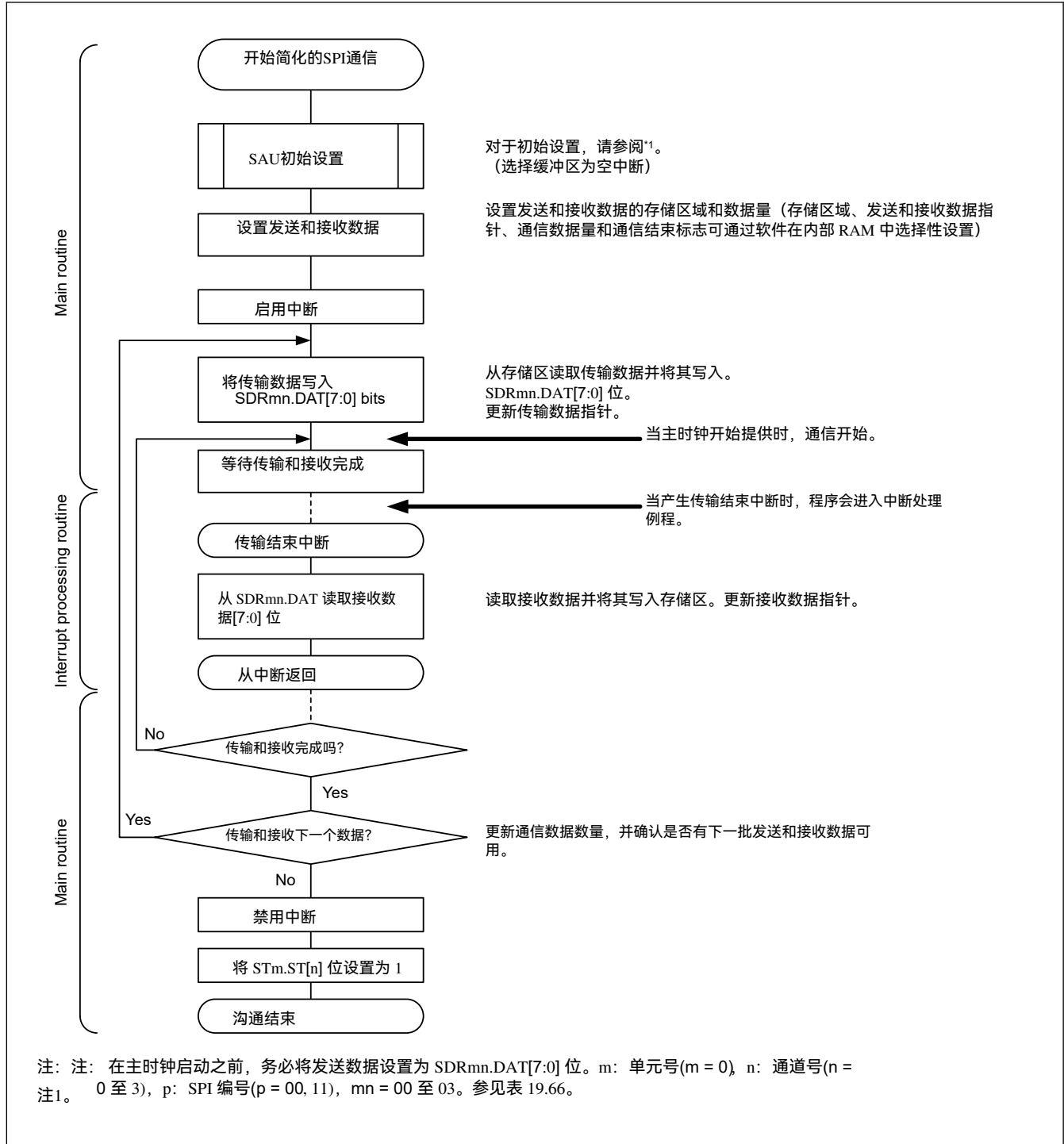


图 19.23 从设备发送和接收流程图 (单次发送和接收模式)

(4) 处理流程 (连续发送和接收模式)

图 19.24 显示了从设备传输和接收的时序 (在连续传输和接收模式下) (类型 1: SCRmn.DCP[1:0] = 00b).

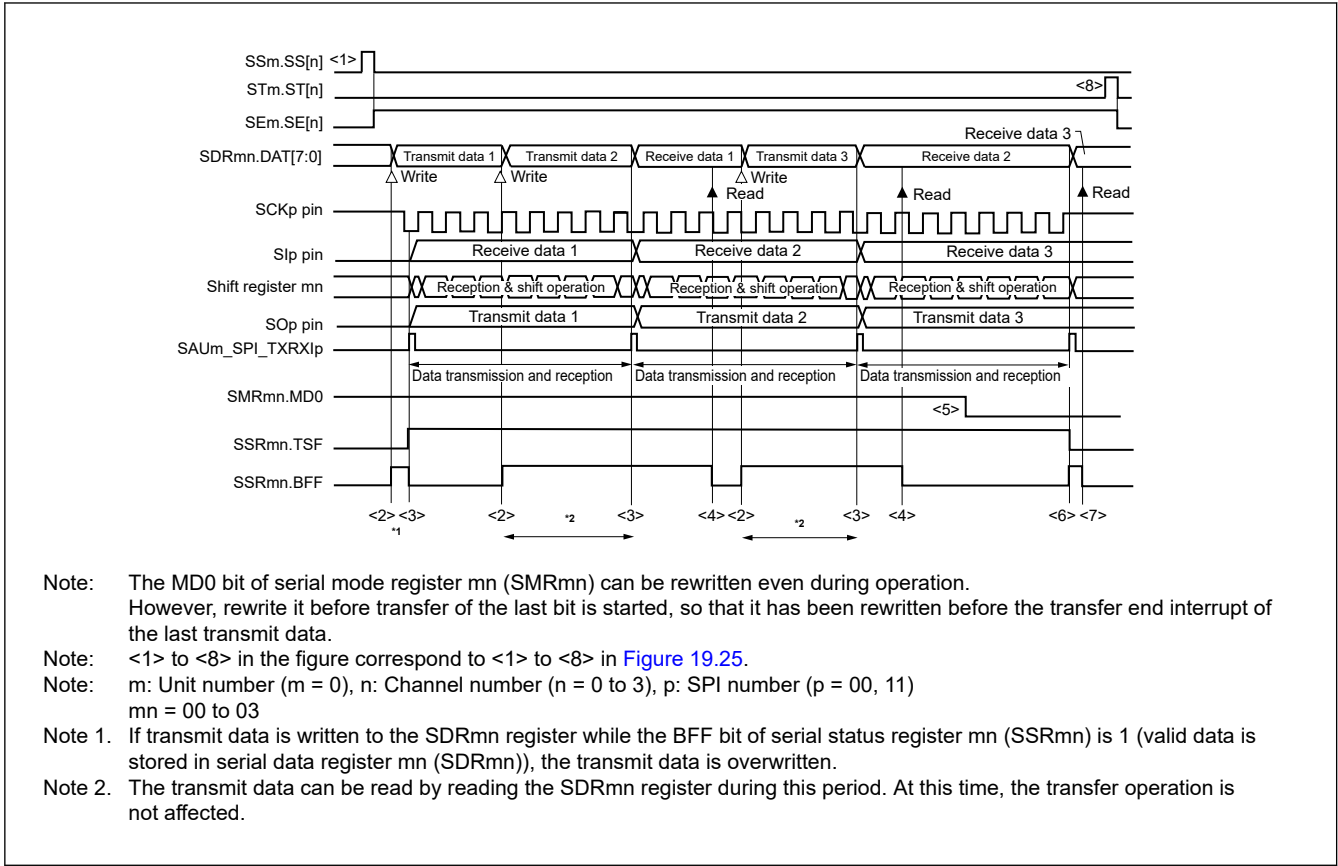


Figure 19.24 Timing of slave transmission and reception (in continuous transmission and reception mode) (type 1: SCRmn.DCP[1:0] = 00b)

Figure 19.25 shows the flowchart of slave transmission and reception (in continuous transmission and reception mode).

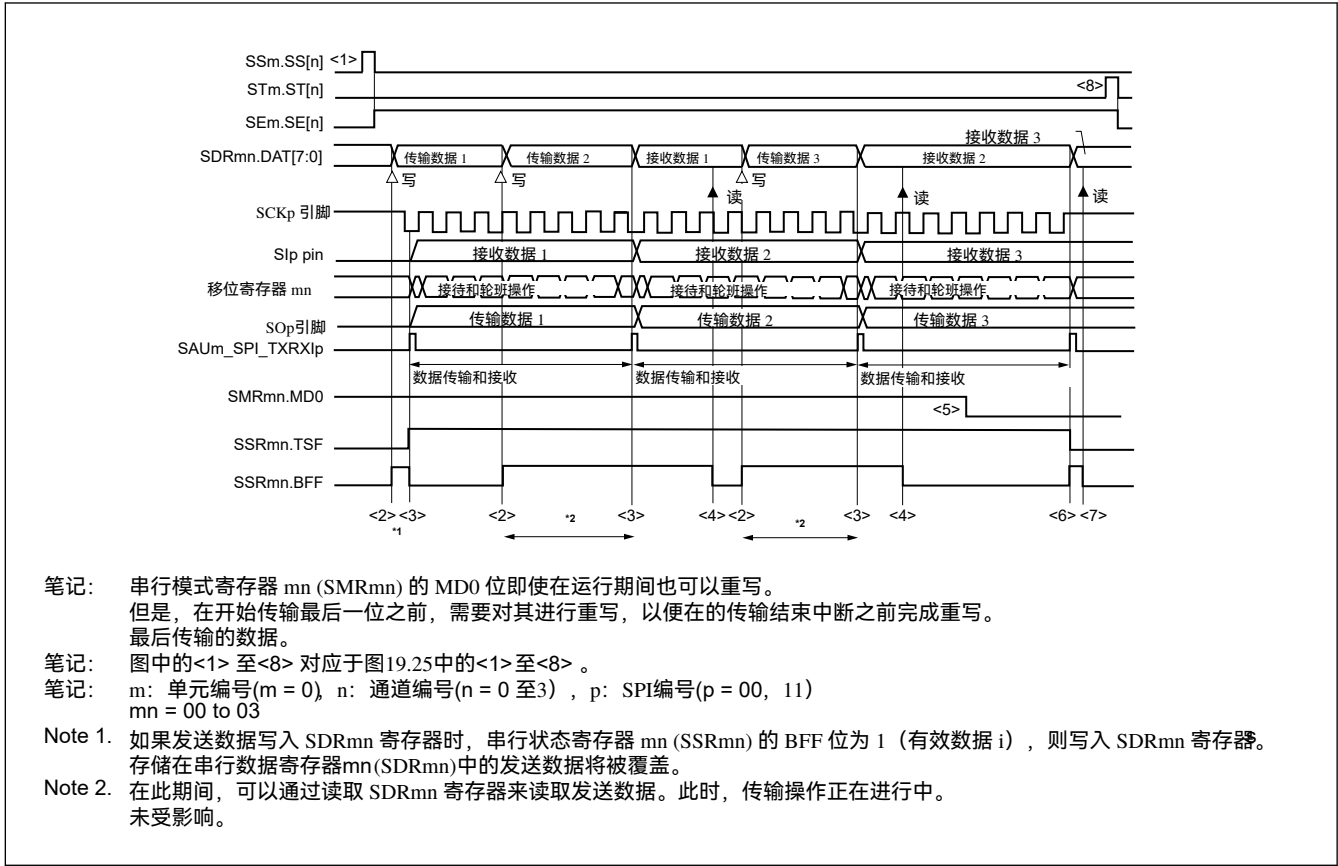


Figure 19.24 从设备传输和接收的时序 (连续传输和接收模式) (类型 1: SCRmn.DCP[1:0] = 00b)

图 19.25 显示了从设备传输和接收的流程图 (在连续传输和接收模式下)。

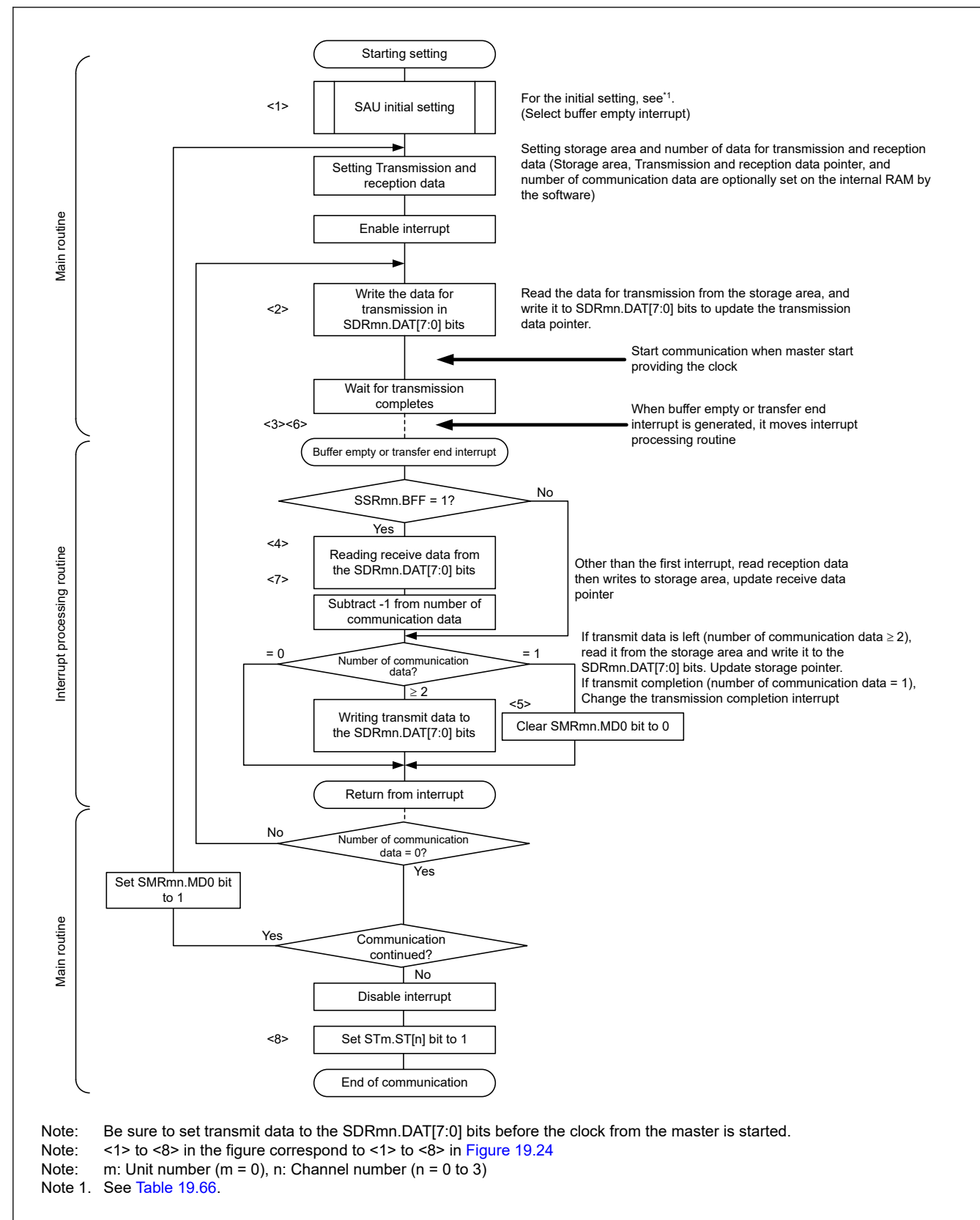


Figure 19.25 Flowchart of slave transmission and reception (in continuous transmission and reception mode)

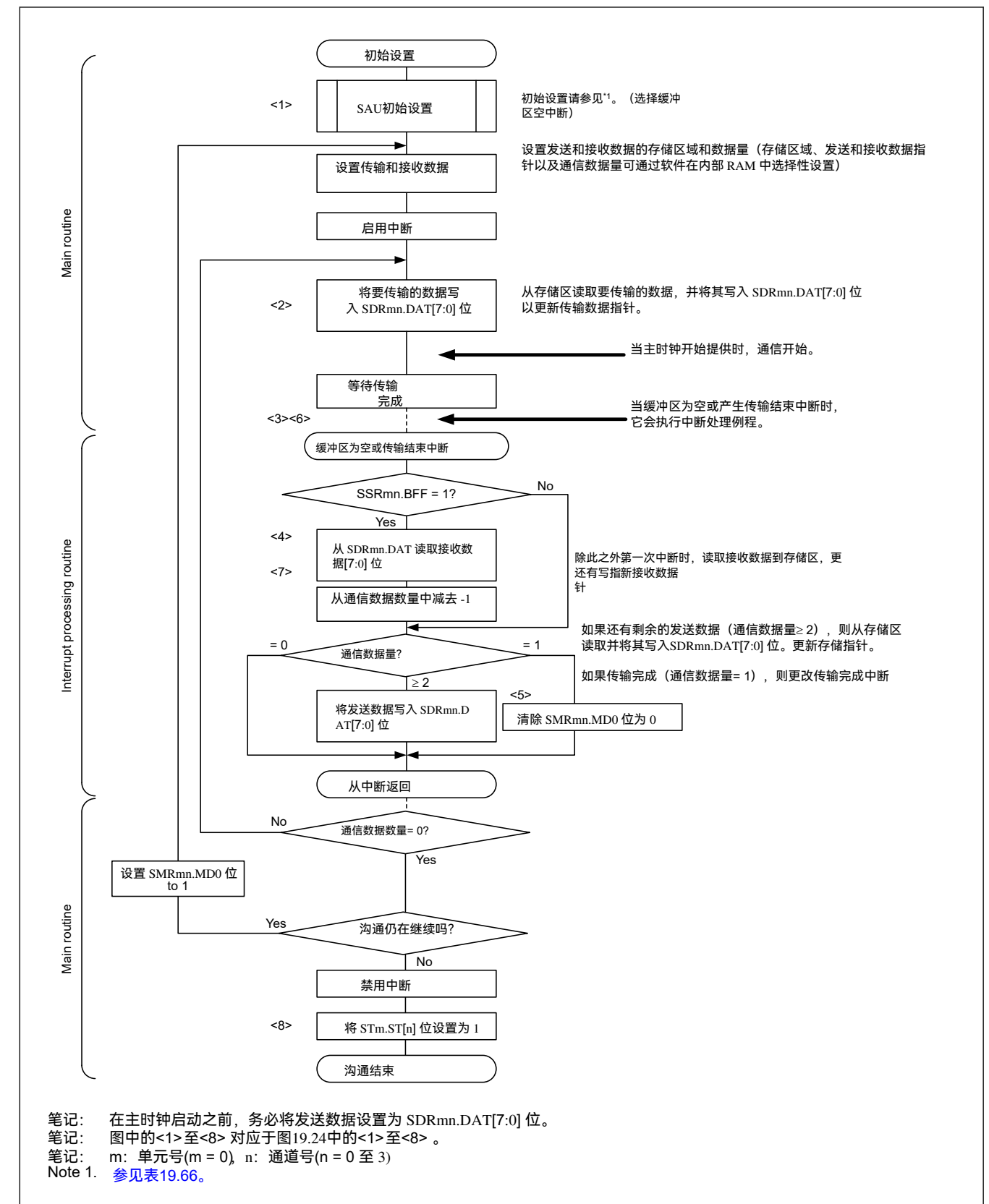


图 19.25 从设备发送和接收流程图（连续发送和接收模式）

19.5.7 Snooze Mode Function

The Snooze mode makes the simplified SPI perform reception operations on SCK00 pin input detection while in the Software Standby mode. Normally the simplified SPI stops communication in the Software Standby mode. However, using the Snooze mode enables the simplified SPI to perform reception operations without CPU operation on detection of the SCK00 pin input. Only SPI00 channel can be set to the Snooze mode.

When using the simplified SPI in Snooze mode, make the following setting before switching to the Software Standby mode (see Figure 19.27 and Figure 19.29.)

- When using the Snooze mode function, set the SWC bit of serial standby control register 0 (SSC0) to 1 just before switching to the Software Standby mode. After the initial setting has been completed, set the SS[0] bit of serial channel start register 0 (SS0) to 1.
- The CPU shifts to the Snooze mode on detecting the valid edge of the SCK00 signal following a transition to the Software Standby mode.
An SPI00 starts reception on detecting input of the serial clock on the SCK00 pin.

Note: The Snooze mode can only be specified when the high-speed on-chip oscillator clock or middle-speed on-chip oscillator clock is selected for PCLKB.

Note: The maximum transfer rate when using SPI00 in the Snooze mode is 1 Mbps.

(1) Snooze mode operation (on startup)

Figure 19.26 shows the timing of Snooze mode operation (on startup) (Type 1: SCR00.DCP[1:0] = 00b).

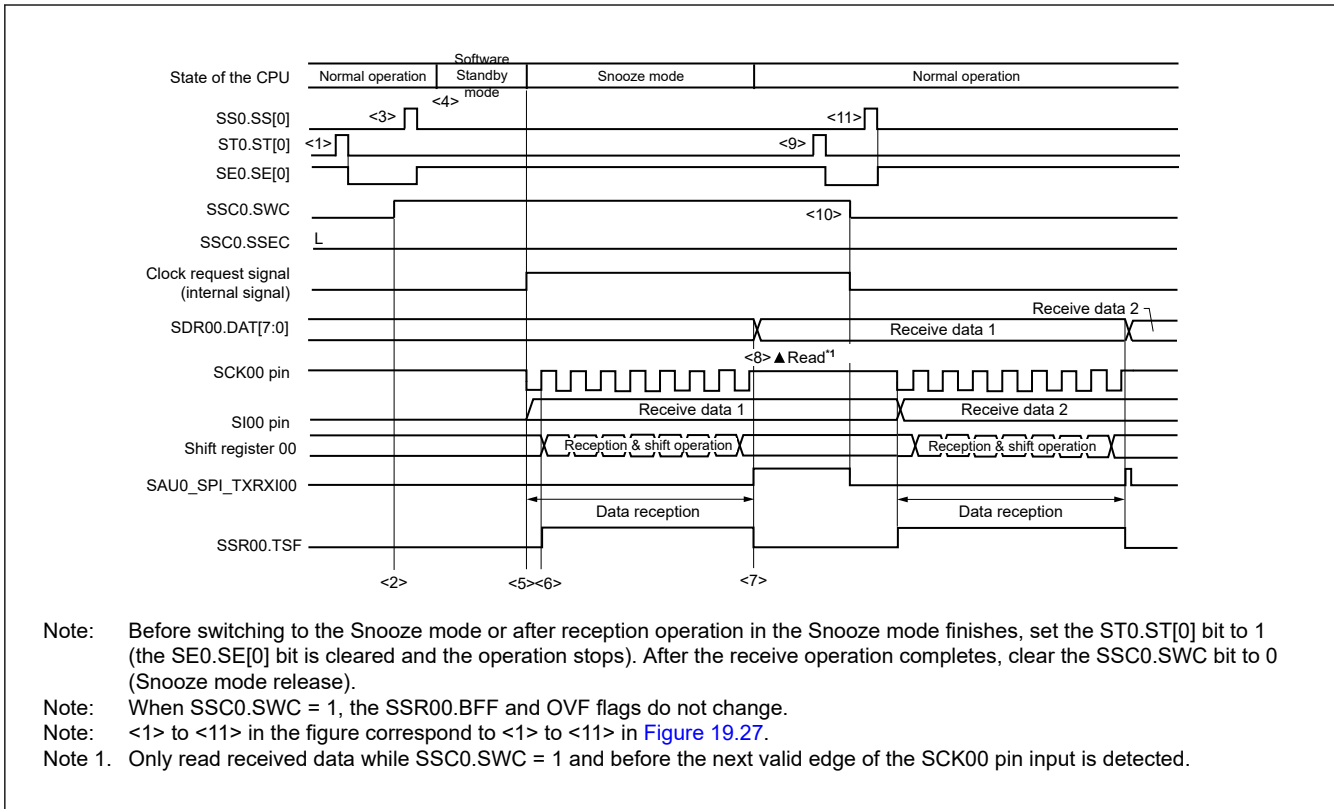


Figure 19.26 Timing of Snooze mode operation (on startup) (type 1: SCR00.DCP[1:0] = 00b)

Figure 19.27 shows the flowchart of Snooze mode operation (on startup).

19.5.7 贪睡模式功能

休眠模式使简化的 SPI 在软件待机模式下，检测到 SCK00 引脚输入信号时执行接收操作。通常情况下，简化的 SPI 在软件待机模式下会停止通信。但是，启用休眠模式后，简化的 SPI 可以在检测到 SCK00 引脚输入信号时执行接收操作，而无需 CPU 执行任何操作。只有 SPI00 通道可以设置为休眠模式。

在休眠模式下使用简化的 SPI 时，在切换到软件待机模式之前，请进行以下设置（参见图 19.27 和图 19.29）。

- 使用休眠模式功能时，在切换到软件待机模式之前，将串行待机控制寄存器0（SSC0）的SWC位设置为1。初始设置完成后，将串行通道起始寄存器0（SS0）的SS[0]位设置为1。
- CPU 在进入软件待机模式后，检测到 SCK00 信号的有效边沿时，会切换到休眠模式。

当检测到 SCK00 引脚上的串行时钟输入时，SPI00 开始接收。

注意：只有当为 PCLKB 选择高速片上振荡器时钟或中速片上振荡器时钟时，才能指定睡眠模式。

注意：在睡眠模式下使用 SPI00 时，最大传输速率为 1 Mbps。

(1) 启动时的贪睡模式操作

图 19.26 显示了 Snooze 模式操作的时序（启动时）（类型1: SCR00.DCP[1:0] = 00b）。

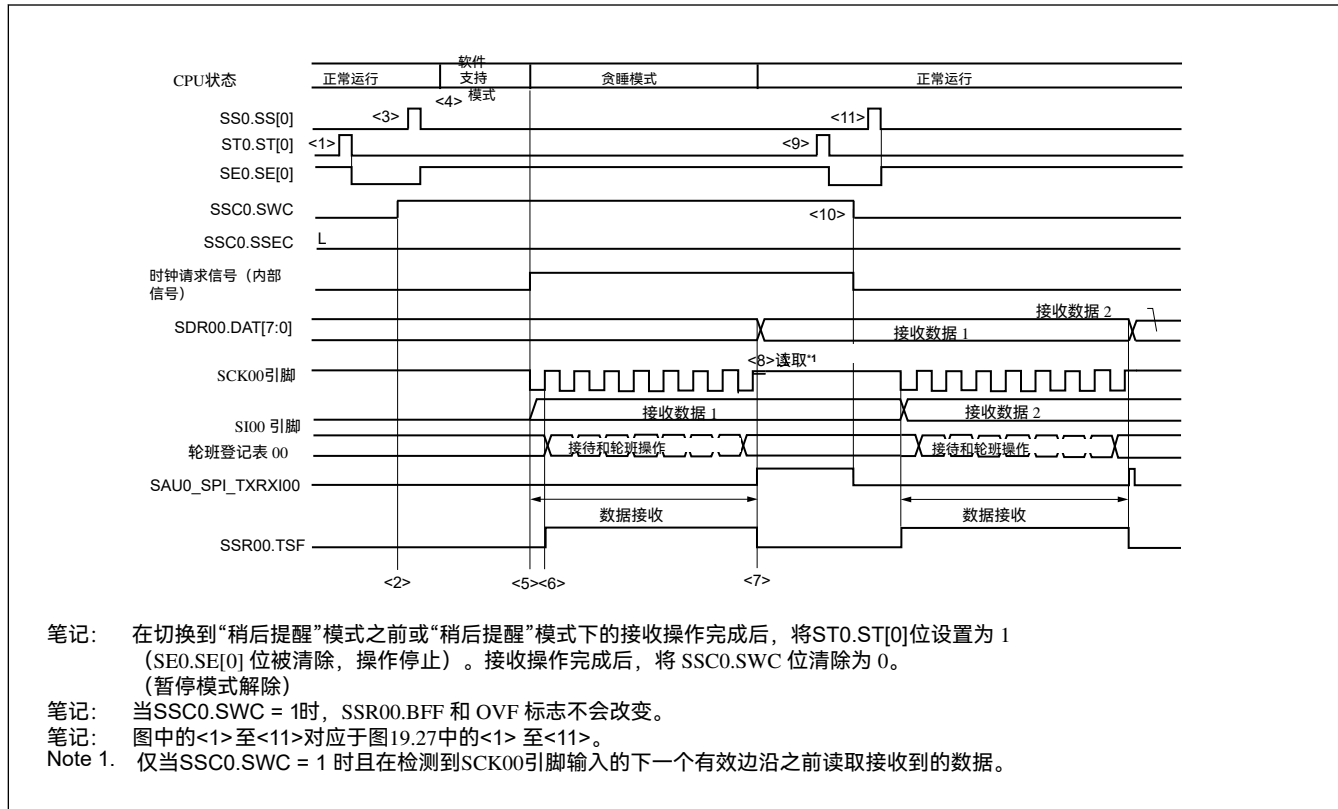


图 19.26 休眠模式操作的时序（启动时）（类型 1: SCR00.DCP[1:0] = 00b）

图 19.27 显示了睡眠模式操作（启动时）的流程图。

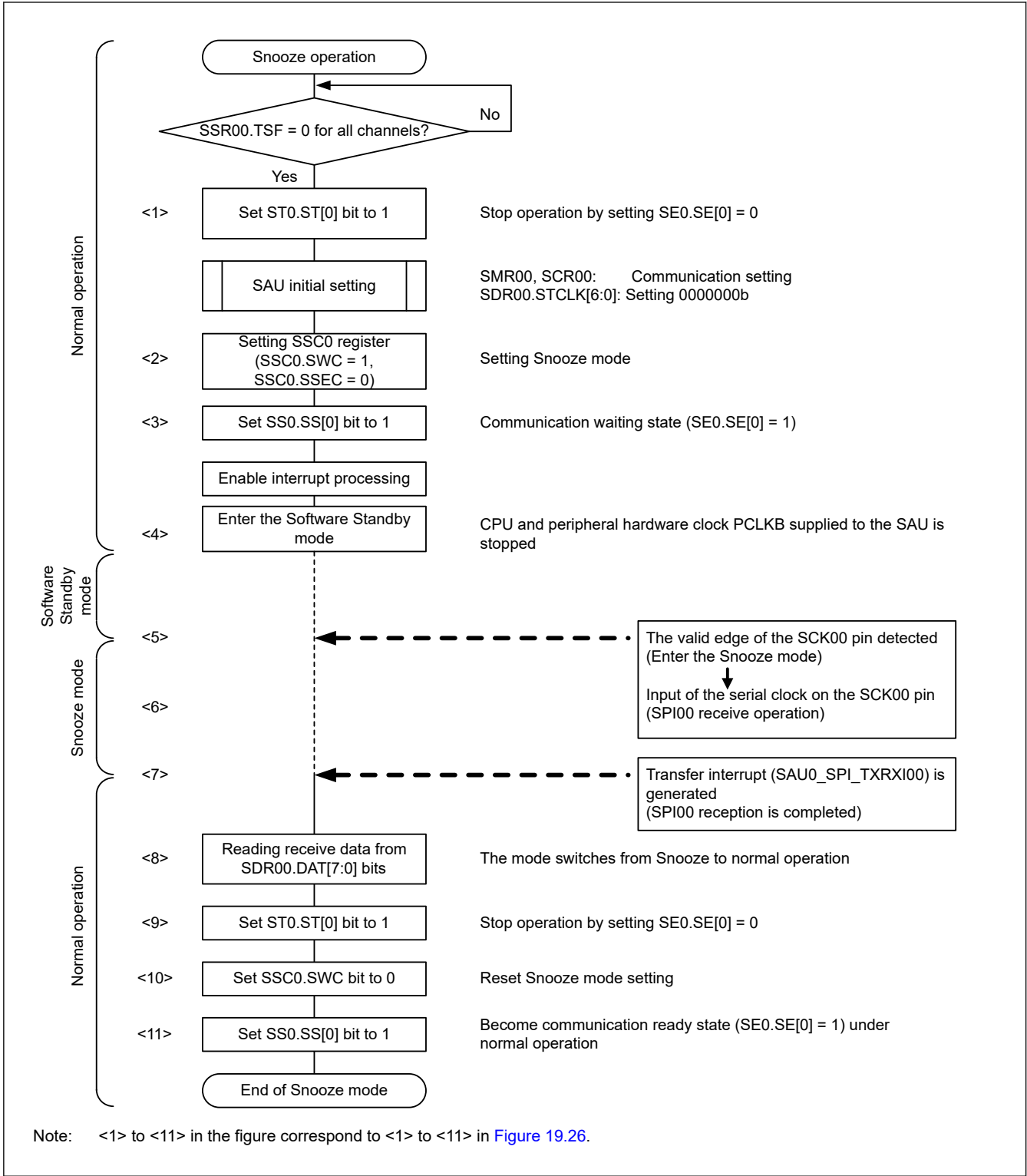


Figure 19.27 Flowchart of Snooze mode operation (on startup)

(2) Snooze mode operation (continuous startup)

Figure 19.28 shows the timing of Snooze mode operation (continuous startup) (type 1: SCR00.DCP[1:0] = 00b).

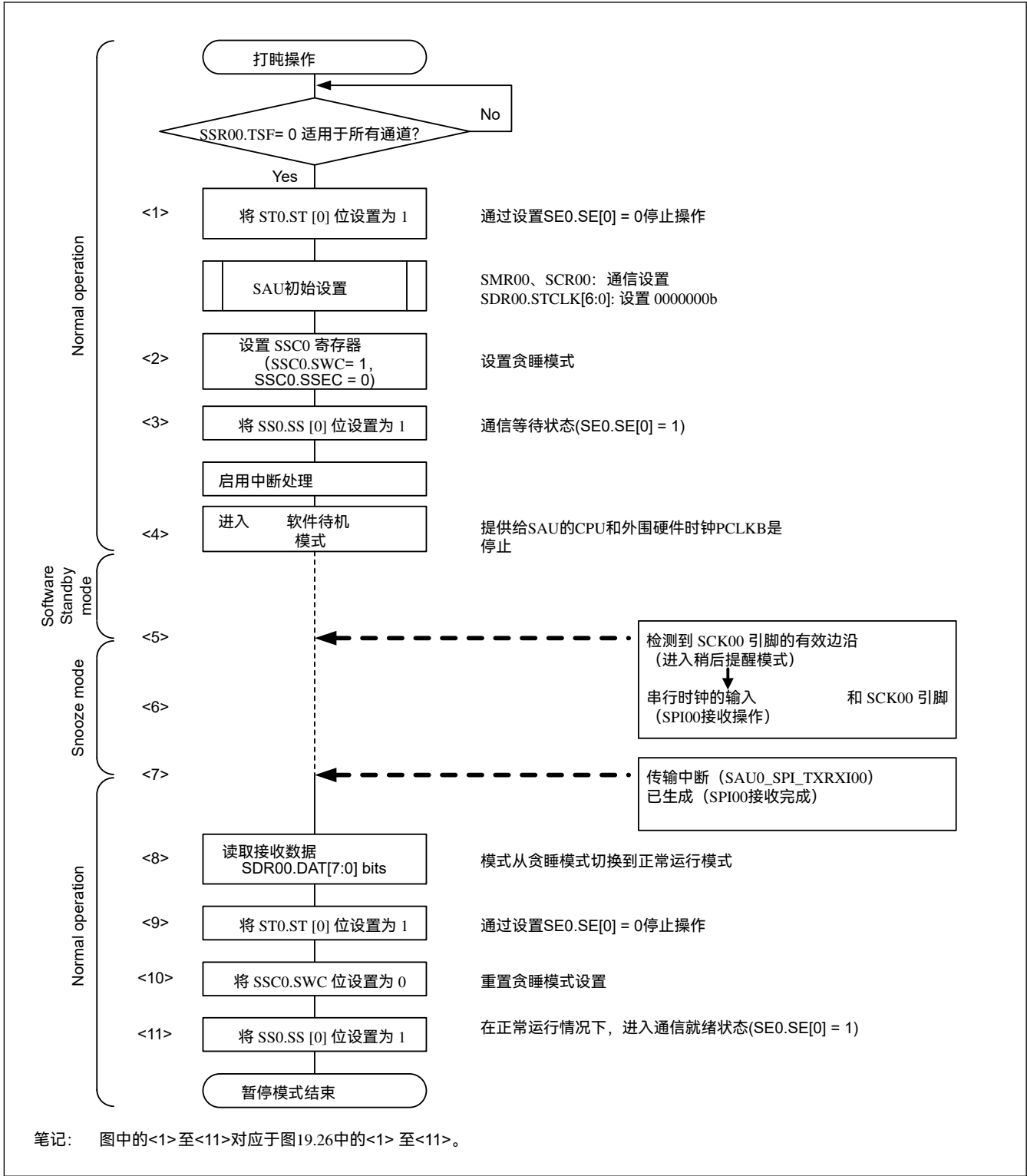


图 19.27 休眠模式操作流程（启动时）

(2) 休眠模式运行（持续启动）

图 19.28 显示了 Snooze 模式操作（连续启动）的时序（典型值e 1: SCR00.DCP[1:0] = 00b）。

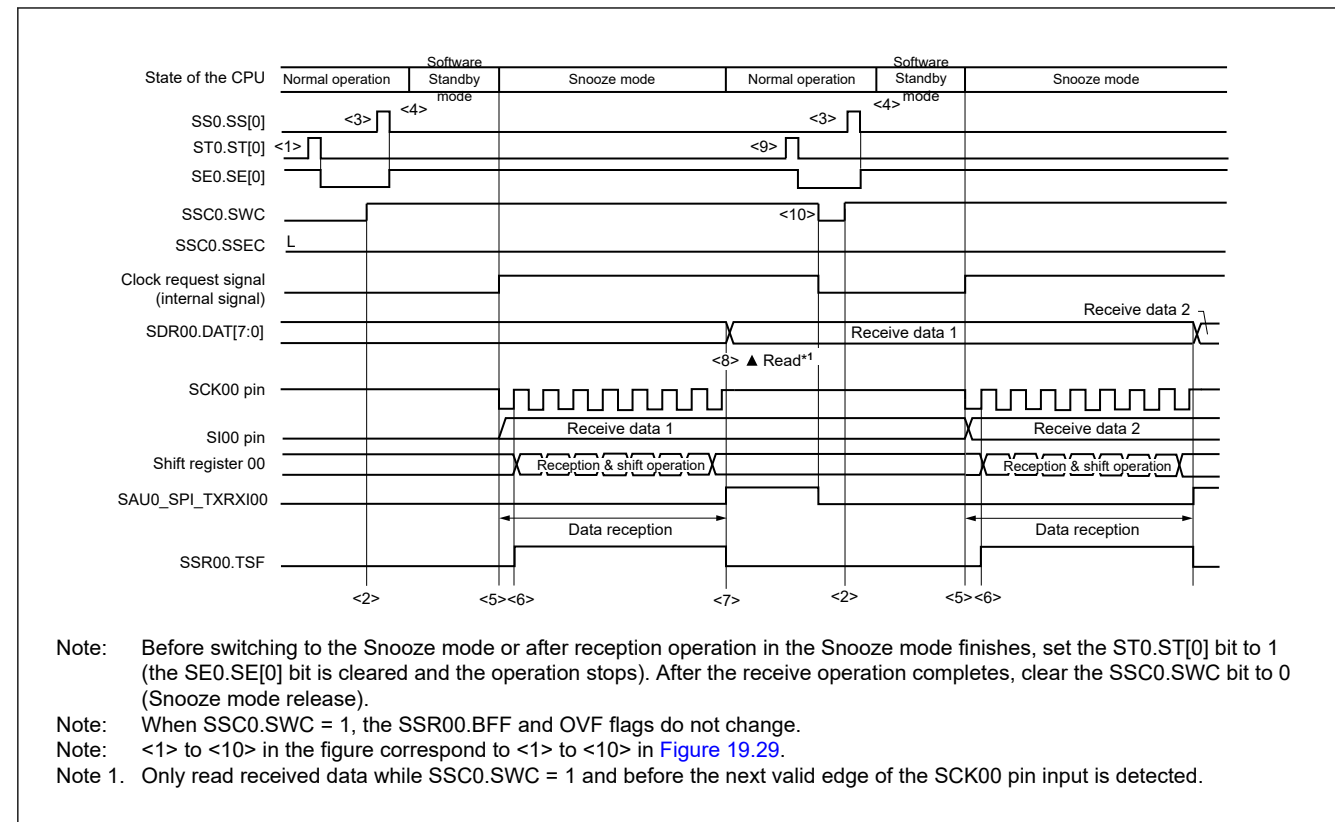


Figure 19.28 Timing of Snooze mode operation (continuous startup) (type 1: SCR00.DCP[1:0] = 00b)

Figure 19.29 shows the flowchart of Snooze mode operation (continuous startup).

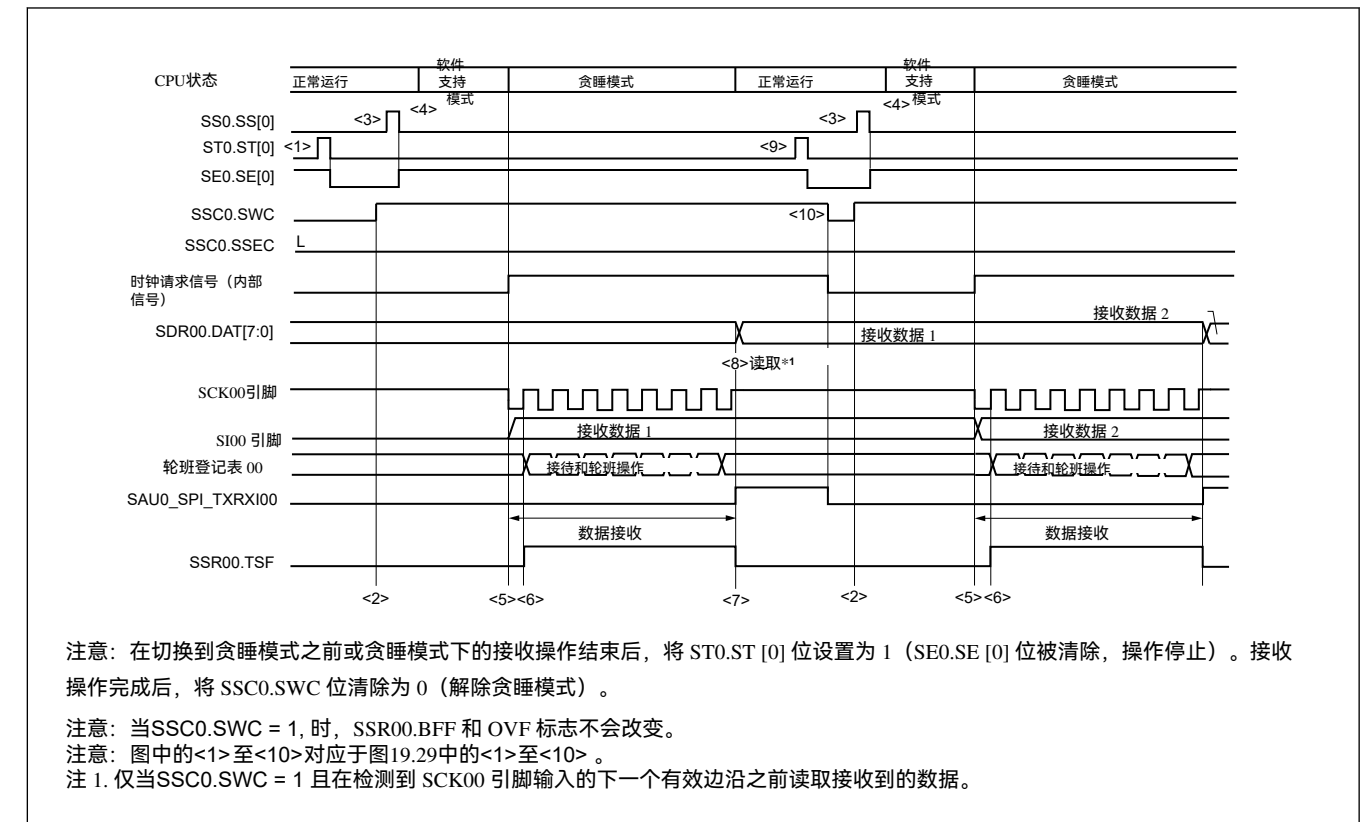


图 19.28 睡眠模式操作时序 (连续启动) (类型 1: SCR00.DCP[1:0] = 00b)

图 19.29 显示了睡眠模式操作（连续启动）的流程图。

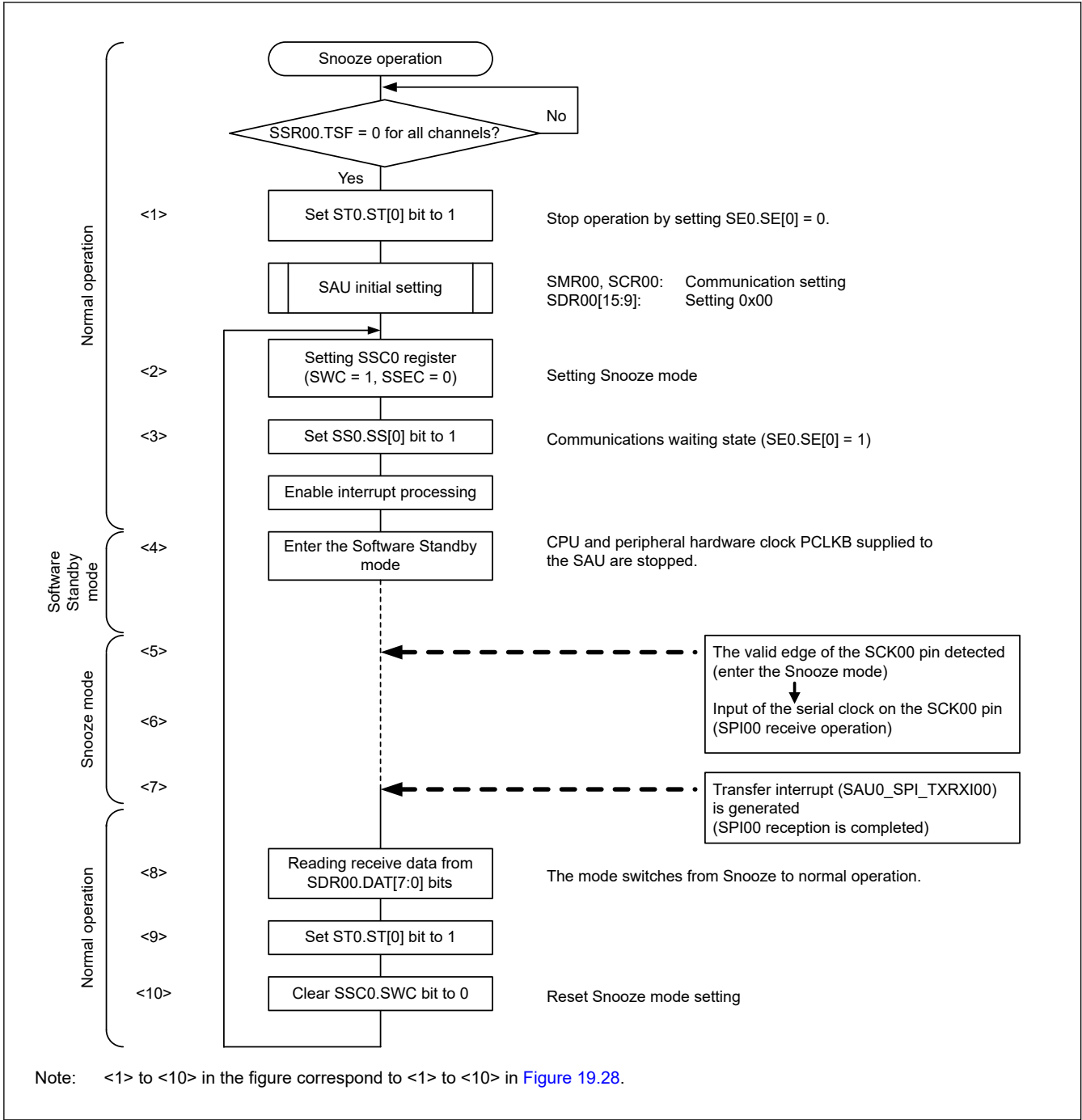


Figure 19.29 Flowchart of Snooze mode operation (continuous startup)

19.5.8 Calculating Transfer Clock Frequency

The transfer clock frequency for simplified SPI communication can be calculated by the following expressions.

1. Master
(Transfer clock frequency) = {Operation clock (f_{MCK}) frequency of target channel} $\div$ (SDRmn.STCLK[6:0] + 1) $\div$ 2 [Hz]
2. Slave
(Transfer clock frequency) = {Frequency of serial clock (SCK) supplied by master} $\times$ 1 [Hz]

Note 1. The permissible maximum transfer clock frequency is $f_{MCK}/6$.

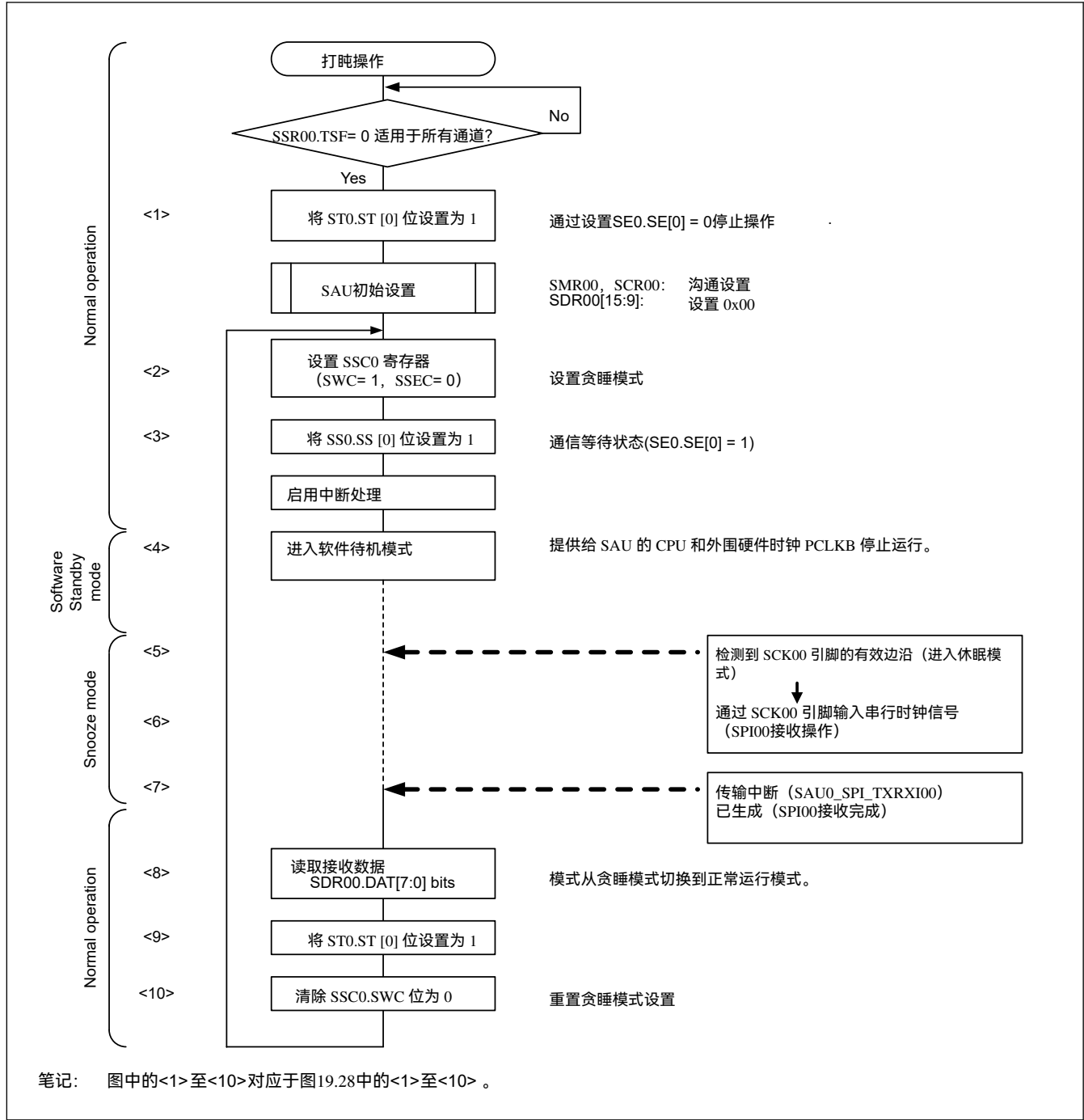


图 19.29 睡眠模式操作流程 (连续启动)

19.5.8 计算传输时钟频率

简化 SPI 通信的传输时钟频率可以通过以下表达式计算。

1. 大师
(传输时钟频率) = {操作时钟(f_{MCK}) 目标信道频率} $\div$ (SDRmn.STCLK[6:0] + 1) $\div$ 2 [Hz]
2. 奴隶
(传输时钟频率) = {主控端提供的串行时钟 (SCK) 频率} $\times$ 1 [Hz]

注1. 允许的最大传输时钟频率为 $f_{MCK}/6$ 。

The operation clock (f_{MCK}) is determined by serial clock select register m (SPSm) and CKS bit of serial mode register mn (SMRmn) as shown in Table 19.69.

Table 19.69 Selection of operation clock for simplified SPI, UART and simplified I²C

SMRmn register	SPSm register		Operation clock (f_{MCK}) *1	
CKS	PRS1[3:0] ²	PRS0[3:0] ²	PCLKB/2 ⁿ	PCLKB = 32 MHz
0	Don't care	0x0	PCLKB	32 MHz
		0x1	PCLKB/2	16 MHz
		0x2	PCLKB/2 ²	8 MHz
		0x3	PCLKB/2 ³	4 MHz
		0x4	PCLKB/2 ⁴	2 MHz
		0x5	PCLKB/2 ⁵	1 MHz
		0x6	PCLKB/2 ⁶	500 kHz
		0x7	PCLKB/2 ⁷	250 kHz
		0x8	PCLKB/2 ⁸	125 kHz
		0x9	PCLKB/2 ⁹	62.5 kHz
		0xA	PCLKB/2 ¹⁰	31.25 kHz
		0xB	PCLKB/2 ¹¹	15.63 kHz
		0xC	PCLKB/2 ¹²	7.81 kHz
		0xD	PCLKB/2 ¹³	3.91 kHz
		0xE	PCLKB/2 ¹⁴	1.95 kHz
		0xF	PCLKB/2 ¹⁵	977 Hz
1	0x0	Don't care	PCLKB	32 MHz
	0x1		PCLKB/2	16 MHz
	0x2		PCLKB/2 ²	8 MHz
	0x3		PCLKB/2 ³	4 MHz
	0x4		PCLKB/2 ⁴	2 MHz
	0x5		PCLKB/2 ⁵	1 MHz
	0x6		PCLKB/2 ⁶	500 kHz
	0x7		PCLKB/2 ⁷	250 kHz
	0x8		PCLKB/2 ⁸	125 kHz
	0x9		PCLKB/2 ⁹	62.5 kHz
	0xA		PCLKB/2 ¹⁰	31.25 kHz
	0xB		PCLKB/2 ¹¹	15.63 kHz
	0xC		PCLKB/2 ¹²	7.81 kHz
	0xD		PCLKB/2 ¹³	3.91 kHz
	0xE		PCLKB/2 ¹⁴	1.95 kHz
	0xF		PCLKB/2 ¹⁵	977 kHz

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3), mn = 00 to 03
Note 1. When changing the clock selected for PCLKB, do so after having stopped (serial channel stop register m (STm) = 0x000F) the operation of the serial array unit (SAU).
Note 2. In the Simplified I²C mode, setting the value greater than 0xB is prohibited.

操作时钟(f_{MCK})由串行时钟选择寄存器m (SPSm) 和串行模式寄存器mn (SMRmn) 的CKS位决定, 如表19.69所示。

表 19.69 简化 SPI、UART 和简化I²C的操作时钟选择

SMRmn 注册	SPSm 注册		操作时钟 (f_{MCK}) *1	
CKS	PRS1[3:0] ²	PRS0[3:0] ²	PCLKB/2 ⁿ	PCLKB = 32 MHz
0	不在乎	0x0	PCLKB	32 MHz
		0x1	PCLKB/2	16 MHz
		0x2	PCLKB/2 ²	8 MHz
		0x3	PCLKB/2 ³	4 MHz
		0x4	PCLKB/2 ⁴	2 MHz
		0x5	PCLKB/2 ⁵	1 MHz
		0x6	PCLKB/2 ⁶	500 kHz
		0x7	PCLKB/2 ⁷	250 kHz
		0x8	PCLKB/2 ⁸	125 kHz
		0x9	PCLKB/2 ⁹	62.5 kHz
		0xA	PCLKB/2 ¹⁰	31.25 kHz
		0xB	PCLKB/2 ¹¹	15.63 kHz
		0xC	PCLKB/2 ¹²	7.81 kHz
		0xD	PCLKB/2 ¹³	3.91 kHz
		0xE	PCLKB/2 ¹⁴	1.95 kHz
		0xF	PCLKB/2 ¹⁵	977 Hz
1	0x0	不在乎	PCLKB	32 MHz
	0x1		PCLKB/2	16 MHz
	0x2		PCLKB/2 ²	8 MHz
	0x3		PCLKB/2 ³	4 MHz
	0x4		PCLKB/2 ⁴	2 MHz
	0x5		PCLKB/2 ⁵	1 MHz
	0x6		PCLKB/2 ⁶	500 kHz
	0x7		PCLKB/2 ⁷	250 kHz
	0x8		PCLKB/2 ⁸	125 kHz
	0x9		PCLKB/2 ⁹	62.5 kHz
	0xA		PCLKB/2 ¹⁰	31.25 kHz
	0xB		PCLKB/2 ¹¹	15.63 kHz
	0xC		PCLKB/2 ¹²	7.81 kHz
	0xD		PCLKB/2 ¹³	3.91 kHz
	0xE		PCLKB/2 ¹⁴	1.95 kHz
	0xF		PCLKB/2 ¹⁵	977 kHz

注: m: 单元编号(m = 0) n: 通道编号(n = 0 至3) , mn = 00至03
注 1. 当更改为 PCLKB 选择的时钟时, 请在停止串行阵列单元 (SAU) 的操作 (串行通道停止寄存器m (STm) = 0x000F) 之后进行。
注 2. 在简化的I²C模式下, 禁止设置大于0xB 的值。

19.5.9 Procedure for Processing Errors that Occurred During Simplified SPI Communication

The procedure for processing errors that occurred during simplified SPI communication is described in [Table 19.70](#).

Table 19.70 Processing procedure in case of overrun error

Step	Software Manipulation		State of the Hardware	Remark
<1>	Reads serial data register mn (SDRmn).	→	The BFF bit of the SSRmn register is set to 0 and channel n is enabled to receive data.	This is to prevent an overrun error if the next reception is completed during error processing.
<2>	Reads serial status register mn (SSRmn).		—	The error type is identified and the read value is used to clear the error flag.
<3>	Writes 1 to serial flag clear trigger register mn (SIRmn).	→	The error flag is cleared.	The error only during reading can be cleared, by writing the value read from the SSRmn register to the SIRmn register without modification.

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3), mn = 00 to 03

19.6 Operation of UART Communication

This is a start-stop synchronization communication function using two lines: serial data transmission (TXD) and serial data reception (RXD) lines. By using these two communication lines, each data frame, which consist of a start bit, data, parity bit, and stop bit, is transferred asynchronously (using the internal baud rate) between the microcontroller and the other communication party. Full-duplex asynchronous communication UART communication can be performed by using a channel dedicated to transmission (even-numbered channel) and a channel dedicated to reception (odd-numbered channel). The LIN-bus can be implemented by using UART0, timer array unit 0 (channel 7), and an external interrupt (IRQ0).

[Data transmission and reception]

- Data length of 7, 8, or 9 bits*1
- MSB or LSB first selectable
- Level setting of transmit and receive data (selecting whether to reverse the level)
- Parity bit appending and parity check functions
- Stop bit appending and stop bit check function

[Interrupt function]

- Transfer end interrupt and buffer empty interrupt (SAU0_UART_TXI0/SAU0_UART_RXI0/SAU0_UART_TXI1/SAU0_UART_RXI1)
- Error interrupt in case of framing error, parity error, or overrun error (SAU0_UART_ERRI0/SAU0_UART_ERRI1)

[Error detection flag]

- Framing error, parity error, or overrun error

In addition, UART reception of following channels supports the Snooze mode. In the Snooze mode, data can be received without CPU processing upon detecting RXD input in the Software Standby mode. The Snooze mode is only available in UART0, which support the reception baud rate adjustment function.

The LIN-bus is accepted in UART0 (channels 0 and 1 of unit 0).

[LIN-bus functions]

LIN-bus functions are achieved using the external interrupt (IRQ0) and timer array unit 0 (channel 7).

- Wakeup signal detection
- Break field (BF) detection
- Sync field measurement, baud rate calculation

Note 1. Only UART0 supports the 9-bit data length.

19.5.9 简化SPI通信过程中发生的错误的处理程序

表 19.70 描述了简化 SPI 通信过程中发生的错误的处理程序。

表 19.70 超额误差处理程序

步	软件操控		硬件状态	评论
<1>	读取串行数据寄存器 mn (SDRmn n)。	→	SSRmn 寄存器的 BFF 位设置为 0，通道 n 被启用以接收数据。	这是为了防止在错误处理期间完成下一次接收时发生溢出错误。
<2>	读取串行状态寄存器 mn (SSRmn)。		—	识别错误类型，并使用读取的值清除错误标志。
<3>	向串行标志清除触发寄存器 mn (SIRmn) 写入 1。	→	错误标志已清除。	只有在读取过程中才会出现错误，可以通过将从 SSRmn 寄存器读取的值直接写入 SIRmn 寄存器来清除该错误。

注: m: 单元编号(m = 0) n: 通道编号(n = 0 至3) , mn = 00至03

19.6 UART通信的操作

这是一种使用两条线路的启动-停止同步通信功能：串行数据传输 (TXD) 线和串行数据接收 (RXD) 线。通过这两条通信线路，每个数据帧（包含起始位、数据、奇偶校验位和停止位）在微控制器和通信另一方之间异步传输（使用内部波特率）。全双工异步 UART 通信可以通过使用专用于发送的通道（偶数通道）和专用于接收的通道（奇数通道）来实现。LIN 总线可以通过使用 UART0、定时器阵列单元 0（通道 7）和外部中断（IRQ0）来实现。

[数据传输和接收]

- 数据长度为7、8或9位*1
- 可优先选择最高有效位 (MSB) 或最低有效位 (LSB)。
- 发送和接收数据的电平设置（选择是否反转电平）
- 奇偶校验位附加和奇偶校验功能
- 停止位追加和停止位校验功能

[中断功能]

- 传输结束中断和缓冲区空中断（SAU0_UART_TXI0/SAU0_UART_RXI0/SAU0_UART_TXI1/SAU0_UART_RXI1）
- 帧错误、奇偶校验错误或溢出错误时的错误中断（SAU0_UART_ERRI0/SAU0_UART_ERRI1）

[错误检测标志]

- 帧错误、奇偶校验错误或溢出错误

此外，以下通道的 UART 接收支持休眠模式。在休眠模式下，当软件待机模式下检测到 RXD 输入时，无需 CPU 处理即可接收数据。休眠模式仅适用于 UART0，UART0 支持接收波特率调节功能。

LIN 总线可通过 UART0（单元 0 的通道 0 和 1）接收。

[LIN总线功能]

LIN 总线功能是通过外部中断 (IRQ0) 和定时器阵列单元 0（通道 7）实现的。

- 唤醒信号检测
- 断场（BF）检测
- 同步场测量，波特率计算

注1: 只有UART0支持9位数据长度。

When the low-speed on-chip oscillator clock (LOCO) is selected for PCLKB, use the low-speed on-chip oscillator trimming register (LIOTRM) to correct oscillation frequency accuracy.

- UART0 uses channels 0 and 1 of SAU0
- UART1 uses channels 2 and 3 of SAU0

See [Table 19.1](#).

Select a single function for each channel. Only the selected function is possible. If UART0 is selected for channels 0 and 1 of unit 0, for example, the SPI00 functions cannot be used. At this time, however, channel 2 or 3 of the same unit can be used for a function other than UART0, such as SPI11, UART1, and IIC11.

Note: When using a serial array unit for UART, both the transmitter side (even-numbered channel) and the receiver side (odd-numbered channel) can only be used for UART.

UART performs the following four types of communication operations.

- UART transmission (See [section 19.6.1. UART Transmission.](#))
- UART reception (See [section 19.6.2. UART Reception.](#))
- LIN transmission (UART0 only) (See [section 19.7.1. LIN Transmission .](#))
- LIN reception (UART0 only) (See [section 19.7.2. LIN Reception .](#))

19.6.1 UART Transmission

UART transmission is an operation to transmit data from a microcontroller to another device asynchronously (start-stop synchronization).

Of the two channels used for UART, the even channel is used for UART transmission.

[Table 19.71](#) shows the specification of UART transmission.

Table 19.71 Specification of UART transmission

UART	UART0	UART1
Target channel	Channel 0 of SAU0	Channel 2 of SAU0
Pins used	TXD0	TXD1
Interrupt	SAU0_UART_TXI0	SAU0_UART_TXI1
	Transfer end interrupt (in single-transfer mode) or buffer empty interrupt (in continuous transfer mode) can be selected.	
Error detection flag	None	
Transfer data length	7, 8, or 9 bits*1	
Transfer rate*2	Max. f _{MCK} /6 [bps] (SDRmn.STCLK[6:0] = 2 or more), Min. PCLKB/ (2 × 2 ¹⁵ × 128) [bps]	
Data phase	Non-reverse output (default: high level) Reverse output (default: low level)	
Parity bit	The following selectable ● No parity bit ● Appending 0 parity ● Appending even parity ● Appending odd parity	
Stop bit	The following selectable: ● Appending 1 bit ● Appending 2 bit	
Data direction	MSB or LSB first	

Note: f_{MCK}: Operation clock frequency of target channel

Note: m: Unit number (m = 0), n: Channel number (n = 0, 2), mn = 00, 02

Note 1. Only UART0 supports the 9-bit data length.

Note 2. Use this operation within a range that satisfies the conditions above and the peripheral functions characteristics specified in the electrical characteristics. For details, see [section 27, Electrical Characteristics](#) .

当选择低速片上振荡器时钟 (LOCO) 作为 PCLKB 时，使用低速片上振荡器微调寄存器 (LIOTRM) 来校正振荡频率精度。

- UART0 使用 SAU0 的通道 0 和 1。
- UART1 使用 SAU0 的通道 2 和 3。

参见表 19.1。

为每个通道选择一个功能。只能使用所选功能。例如，如果为单元 0 的通道 0 和 1 选择 UART0，则无法使用 SP I00 功能。但此时，同一单元的通道 2 或 3 可以用于 UART0 以外的功能，例如 SPI11、UART1 和 IIC11。

注意：当使用串行阵列单元进行 UART 通信时，发送端（偶数通道）和接收端（奇数通道）都只能用于 UART 通信。

UART 执行以下四种类型的通信操作。

- UART传输（参见19.6.1节 UART传输。）
- UART接收（参见19.6.2节 UART接收。）
- LIN 传输（仅限 UART0）（参见 19.7.1 节 LIN 传输。）
- LIN 接收（仅限 UART0）（参见 19.7.2 节 LIN 接收。）

19.6.1 UART 传输

UART 传输是指将数据从微控制器异步传输到另一个设备（启动-停止同步）的操作。

用于 UART 的两个通道中，偶数通道用于 UART 传输。

表 19.71 显示了 UART 传输的规范。

表 19.71 UART 传输规范

UART	UART0	UART1
目标频道	SAU0 的 0 通道	SAU0 的 2 频道
使用的引脚	TXD0	TXD1
打断	SAU0_UART_TXI0	SAU0_UART_TXI1
	可以选择传输结束中断（单次传输模式）或缓冲区空中断（连续传输模式）。	
错误检测标志	None	
传输数据长度	7、8 或 9 位*1	
转账费率*2	最大限度。f _{MCK} /6[bps] （SDRmn.STCLK[6: 0] = 2或更多），最小\PCLKB/(2 × 2 ¹⁵ × 128) [bps]	
数据阶段	非反向输出（默认值：高电平） 反向输出（默认值：低电平）	
奇偶校验位	以下可选 ●无奇偶校验位 ●附加 0 奇偶校验 ●附加偶数奇偶性 ●附加奇偶性	
停止位	以下选项可供选择： ●追加 1 位 ●附加 2 位	
数据方向	先 MSB 还是 LSB	

注: f_{MCK}: 目标通道的工作时钟频率

注: m: 单元编号(m = 0), n: 通道编号(n = 0, 2), mn = 00, 02

注1: 只有UART0支持9位数据长度。

注2: 此操作应在满足上述条件和电气特性中规定的外围功能特性的范围内使用。详情请参见第27节“电气特性”。

(1) Register setting

Table 19.72 to Table 19.78 show examples of the register contents for UART transmission.

(a) Serial mode register mn (SMRmn)

Table 19.72 Example of serial mode register mn (SMRmn) contents for UART transmission

Bit	Symbol	Set value	Function
0	MD0	0/1	Interrupt source of channel n 0: Transfer end interrupt 1: Buffer empty interrupt
2:1	MD1[1:0]	01b	Setting of operation mode of channel n 0 1: UART mode
13:3	—	000_0000_0100 b	Setting disabled (set to the initial value)
14	CCS	0	Selection of transfer clock (f _{TCLK}) of channel n 0: Divided operation clock f _{MCK} specified by the CKS bit
15	CKS	0/1	Operation clock (f _{MCK}) of channel n 0: Prescaler output clock CKm0 set by the SPSm register 1: Prescaler output clock CKm1 set by the SPSm register

(b) Serial communication operation setting register mn (SCRmn)

Table 19.73 Example of serial communication operation setting register mn (SCRmn) contents for UART transmission

Bit	Symbol	Set value	Function
1:0	DLS[1:0]	01b to 11b	Setting of data length 0 1: 9-bit data length 1 0: 7-bit data length 1 1: 8-bit data length
3:2	—	01b	Setting disabled (set to the initial value)
5:4	SLC[1:0]	01b or 10b	Setting of stop bit 0 1: Appending 1 bit 1 0: Appending 2 bit
6	—	0	Setting disabled (set to the initial value)
7	DIR	0/1	Selection of data transfer sequence in simplified SPI and UART modes 0: Inputs or outputs data with MSB first. 1: Inputs or outputs data with LSB first.
9:8	PTC[1:0]	00b to 11b	Setting of parity bit 0 0: No parity 0 1: Appending 0 parity 1 0: Appending Even parity 1 1: Appending Odd parity
10	EOC	0	Since this bit is dedicated to UART receive modes, it is fixed in the UART transmission mode.
11	—	0	Setting disabled (set to the initial value)
13:12	DCP[1:0]	00b	Since this bit is dedicated to other modes, it is fixed in the UART mode
15:14	TRXE[1:0]	10b	Setting TRXE[1:0] = 10b is fixed in the UART transmission mode

(1) 寄存器设置

表 19.72 至表 19.78 显示了 UART 传输的寄存器内容示例。

(a) 串行模式寄存器 mn (SMRmn)

表 19.72 UART 传输的串行模式寄存器 mn (SMRmn) 内容示例

少量	象征	设置值	功能
0	MD0	0/1	通道 n 的中断源 0: 传输结束中断 1: 缓冲区为空中断
2:1	MD1[1:0]	01b	设置通道 n 的工作模式 0 1: UART 模式
13:3	—	000_0000_0100 b	设置已禁用 (已设置为初始值)
14	CCS	0	选择通道 n 的传输时钟 (f _{TCLK}) 0: 由 CKS 位指定的分频操作时钟 f _{MCK}
15	CKS	0/1	通道 n 的运行时钟 (f _{MCK}) 0: 预分频器输出时钟 CKm0 由 SPSm 寄存器设置 1: 预分频器输出时钟 CKm1 由 SPSm 寄存器设置

(b) 串行通信操作设置寄存器 mn (SCRmn)

表 19.73 串行通信操作示例, 设置寄存器 mn (SCRmn) 的内容以进行 UART 传输

少量	象征	设置值	功能
1:0	DLS[1:0]	01b 至 11b	数据长度设置: 0 1: 9 位数 据长度; 1 0: 7 位数数据长度 ; 1 1: 8 位数数据长度
3:2	—	01b	设置已禁用 (已设置为初始值)
5:4	SLC[1:0]	01b 或 10b	设置停止位 0 1: 追加 1 位 1 0: 追加 2 位
6	—	0	设置已禁用 (已设置为初始值)
7	DIR	0/1	简化 SPI 和 UART 模式下数据传输顺序的选择: 0: 以最高有效位 (MSB) 为优先进行输入或输出数据。1: 以最低有效位 (LSB) 为优先进行输入或输出数据。
9:8	PTC[1:0]	00b 至 11b	设置奇偶校验位 0 0: 无奇偶校验 0 1: 附加 0 奇偶校验 1 0: 附加偶奇偶性 1.1: 附加奇偶性
10	EOC	0	由于该位专用于 UART 接收模式, 因此在 UART 发送模式下它是固定的。
11	—	0	设置已禁用 (已设置为初始值)
13:12	DCP[1:0]	00b	由于该位专用于其他模式, 因此在 UART 模式下它是固定的。
15:14	TRXE[1:0]	10b	设置 TRXE[1:0] = 10b 在 UART 传输模式下是固定的

(c) Serial data register mn (SDRmn)

Table 19.74 Example of serial data register mn (SDRmn) contents for UART transmission

Bit	Symbol	Set value	Function
6:0	DAT[6:0]	0x00 to 0x7F	Setting transmit data [6:0]
7	DAT[7]	0/1	Setting transmit data [7] (8-bit and 9-bit data length)
		0	0 Fixed (7-bit data length)
8	DAT[8]*1	0/1	Setting transmit data [8] (9-bit data length)
		0	0 Fixed (7-bit and 8-bit data length)
15:9	STCLK[6:0]	0x02 to 0x7F	Baud rate setting (Operation clock (f _{MCK}) division setting)

Note 1. When UART0 performs 9-bit communication, SDRm0.DAT[8:0] are used as the transmission data specification area. Only UART0 supports the 9-bit data length.

(d) Serial output level register m (SOLm)

Set only the bit of the target channel.

Table 19.75 Example of serial output level register m (SOLm) contents for UART transmission

Bit	Symbol	Set value	Function
n	SOLn	0/1	Selects inversion of the level of the transmit data of channel 0 in UART mode 0: Non-reverse (normal) transmission 1: Reverse transmission

(e) Serial output register m (SOM)

Set only the bit of the target channel.

Table 19.76 Example of serial output register m (SOM) contents for UART transmission

Bit	Symbol	Set value	Function
n	SO[n]*1	0/1	Serial data output of channel n 0: Serial data output value is 0 1: Serial data output value is 1
n+8	CKO[n]	x	Bit that cannot be used in this mode (set to the initial value when not used in any mode)

Note 1. Before transmission is started, be sure to set to 1 when the SOLm.SOLn bit of the target channel is set to 0, and set to 0 when the SOLm.SOLn bit of the target channel is set to 1. The value varies depending on the communication data during communication operation.

(f) Serial output enable register m (SOEm)

Set only the bit of the target channel to 1.

Table 19.77 Example of serial output enable register m (SOEm) contents for UART transmission

Bit	Symbol	Set value	Function
n	SOE[n]	1	Serial output enable or stop of channel n 1: Enable output by serial communication operation.

(g) Serial channel start register m (SSm)

Set only the bit of the target channel to 1.

(c) 串行数据寄存器 mn (SDRmn)

表 19.74 UART 传输的串行数据寄存器 mn (SDRmn) 内容示例

少量	象征	设置值	功能
6:0	DAT[6:0]	0x00 至 0x7F	设置发送数据[6:0]
7	DAT[7]	0/1	设置发送数据[7] (8 位和 9 位数据长度)
		0	0 固定 (7 位数据长度)
8	DAT[8]*1	0/1	设置发送数据[8] (9 位数据长度)
		0	0 固定 (7 位和 8 位数据长度)
15:9	STCLK[6:0]	0x02 至 0x7F	波特率设置 (操作时钟(f _{MCK}) 分频设置)

注1: 当UART0执行9位通信时, SDRm0.DAT[8:0] 用作传输数据规范区域。只有UART0支持9位数据长度。

(d) 串行输出电平寄存器m (SOLm)

仅设置目标通道的位。

表 19.75 UART 传输的串行输出电平寄存器m (SOLm) 内容示例

少量	象征	设置值	功能
n	解决方案	0/1	选择反转 UART 模式下通道 0 的发送数据电平 0: 非反向 (正常) 传输 1: 倒档变速器

(e) 串行输出寄存器 m (SOM)

仅设置目标通道的位。

表 19.76 UART 传输的串行输出寄存器m(SOM) 内容示例

少量	象征	设置值	功能
n	SO[n]*1	0/1	通道 n 的串行数据输出 0: 串口数据输出值为 0 1: 串口数据输出值为 1
n+8	CKO[n]	x	此模式下无法使用的位 (在任何模式下未使用时, 设置为初始值)

注 1. 在开始传输之前, 当目标信道的 SOLm.SOLn 位设置为 0 时, 务必将其设置为 1; 当目标信道的 SOLm.SOLn 位设置为 1 时, 务必将其设置为 0。该值在通信操作期间会根据通信数据而变化。

(f) 串行输出使能寄存器 m (SOEm)

仅将目标通道的位设置为 1。

表 19.77 UART 传输的串行输出使能寄存器m(SOEm) 内容示例

少量	象征	设置值	功能
n	SOE[n]	1	通道 n 的串行输出启用或停止 1: 通过串口通信操作启用输出。

(g) 串行通道起始寄存器m (SSm)

仅将目标通道的位设置为 1。

Table 19.78 Table 15.84 Example of serial channel start register m (SSm) contents for UART transmission

Bit	Symbol	Set value	Function
n	SS[n]	1	Operation start trigger of channel n 1: Set the SEm.SE[n] bit to 1 to place the channel in communication waiting state

Note: m: Unit number (m = 0), n: Channel number (n = 0, 2), mn = 00, 02

Note: ×: Bit that cannot be used in this mode (set to the initial value when not used in any mode)
0/1: Set to 0 or 1 depending on the usage of the user

(2) Operation procedure

Table 19.79 shows the procedure for initial setting for UART transmission.

Table 19.79 Initial setting procedure for UART transmission

Step		Process	Detail
Procedure for initial setting of UART transmission	<1>	Starting initial setting	—
	<2>	Setting the SPSm register	Set the operation clock.
	<3>	Setting the SMRmn register	Set an operation mode.
	<4>	Setting the SCRmn register	Set a communication format.
	<5>	Setting the SDRmn register	Set a transfer baud rate (setting the transfer clock by dividing the operation clock (f _{MCK})).
	<6>	Changing setting of the SOLm register	Set an output data level.
	<7>	Setting the SOm register	Set the initial output level of the serial data (SOm.SO[n]).
	<8>	Changing setting of the SOEm register	Set the SOEm.SOE[n] bit to 1 and enable data output of the target channel.
	<9>	Setting port	Enable data output of the target channel.
	<10>	Writing to the SSm register	Set the SSm.SS[n] bit of the target channel to 1 and set SEm.SE[n] = 1 to enable operation.
	<11>	Completing initial setting	Initial setting is completed. Set transmit data to the SDRmn.DAT[7:0] bits (8 bits) or the SDRmn.DAT[8:0] bits (9 bits) and start communication.

Note: m: Unit number (m = 0), n: Channel number (n = 0, 2), mn = 00, 02

Table 19.80 shows the procedure for stopping master transmission

Table 19.80 Procedure for stopping UART transmission

Step		Process	Detail
Procedure for stopping UART transmission	<1>	Starting setting to stop	—
	<2>	Wait until SSRmn.TSF is cleared (optional)	If there is any data being transferred, wait for their completion. If there is a requirement to stop, do not wait.
	<3>	Writing the STm register	Write 1 to the STm.ST[n] bit of the target channel and set SEm.SE[n] = 0 to stop operation.
	<4>	Setting the SOEm register	Set the SOEm.SOE[n] bit to 0 and stop the output of the target channel.
	<5>	Changing setting of the SOm register (optional)	The levels of the serial data (SOm.SO[n]) on the target channel can be changed if required.
	<6>	Stop setting is completed	After the stop setting is completed, go to the next processing.

Note: m: Unit number (m = 0), n: Channel number (n = 0, 2), mn = 00, 02

Table 19.81 shows the procedure for resuming UART transmission.

表 19.78 表 15.84 UART 传输的串行通道起始寄存器m (SSm) 内容示例

少量	象征	设置值	功能
n	SS[n]	1	通道 n 的操作启动触发 1: 将 SEm.SE[n] 位设置为 1，使通道处于通信等待状态

注: m: 单元编号(m = 0) n: 通道编号(n = 0, 2), mn = 00, 02

注意: ×: 此模式下不能使用的位（在任何模式下未使用时设置为初始值）
0/1: 根据用户使用情况设置为 0 或 1

(2) 操作流程

表 19.79 显示了 UART 传输的初始设置步骤。

表 19.79 UART 传输的初始设置步骤

步		过程	细节
UART传输初始设置步骤	<1>	初始设置	—
	<2>	设置 SPSm 寄存器	设置运行时钟。
	<3>	设置 SMRmn 寄存器	设置操作模式。
	<4>	设置 SCRmn 寄存器	制定沟通模式。
	<5>	设置 SDRmn 寄存器	设置传输波特率（通过分频操作时钟来设置传输时钟(f _{MCK})）。
	<6>	更改 SOLm 寄存器的设置	设置输出数据级别。
	<7>	设置 SOm 寄存器	设置串行数据的初始输出电平(SOm.SO[n])。
	<8>	更改 SOEm 寄存器的设置	将 SOEm.SOE[n] 位设置为 1，使能目标通道的数据输出。
	<9>	设置端口	启用目标通道的数据输出。
	<10>	致函SSm登记处	将目标通道的 SSm.SS[n] 位设置为 1，并将SEm.SE[n] = 1 设置为启用操作。
	<11>	完成初始设置	初始设置完成。 将发送数据设置为 SDRmn.DAT[7:0] 位（8 位）或 SDRmn.DAT[8:0] 位（9 位），并开始通信。

注: m: 单元编号(m = 0) n: 通道编号(n = 0, 2) mn = 00, 02

表 19.80 显示了停止主传输的步骤

表 19.80 停止 UART 传输的步骤

步		过程	细节
停止UART传输的步骤	<1>	开始设置停止	—
	<2>	等待 SSRmn.TSF 被清除（可选）	如果有数据传输，请等待传输完成。 如果需要停车，不要等待。
	<3>	写入 STm 寄存器	将 1 写入目标通道的 STm.ST[n] 位，并设置SEm.SE[n] = 0 以停止操作。
	<4>	设置 SOEm 寄存器	将 SOEm.SOE[n] 位设置为 0，停止目标通道的输出。
	<5>	更改 SOm 寄存器的设置（可选）	如果需要，可以更改目标通道上的串行数据(SOm.SO[n]) 的电平。
	<6>	停止设置完成	停止设置完成后，进行下一步处理。

注: m: 单元编号(m = 0) n: 通道编号(n = 0, 2), mn = 00, 02

表 19.81 显示了恢复 UART 传输的步骤。

Table 19.81 Procedure for resuming UART transmission

Step		Process	Detail
Procedure for resuming UART transmission	<1>	Starting setting for resumption	—
	<2>	Wait until Communication target is ready	Wait until the communication target stops or communication operation completed
	<3>	Port manipulation	Disable data output of the target channel.
	<4>	Changing setting of the SPSm register (optional)	Reset the register to change the operation clock setting.
	<5>	Changing setting of the SDRmn register (optional)	Reset the register to change the transfer baud rate setting (setting the transfer clock by dividing the operation clock (f _{MCK})).
	<6>	Changing setting of the SMRmn register (optional)	Reset the register to change serial mode register mn (SMRmn) setting.
	<7>	Changing setting of the SCRmn register (optional)	Reset the register to change the serial communication operation setting register mn (SCRmn) setting.
	<8>	Changing setting of the SOLm register (Selective)	Reset the register to change serial output level register m (SOLm) setting.
	<9>	Changing setting of the SOEm register (optional)	Clear the SOEm.SOE[n] bit to 0 and stop output.
	<10>	Changing setting of the SOm register (optional)	Set the initial output level of the serial data (SOm.SO[n]).
	<11>	Changing setting of the SOEm register	Set the SOEm.SOE[n] bit to 1 and enable output.
	<12>	Port manipulation	Enable data output of the target channel.
	<13>	Writing to the SSm register	Set the SSm.SS[n] bit of the target channel to 1 and set the SEm.SE[n] bit to 1 (to enable operation).
	<14>	Completing resumption setting	Setting is completed. Set transmit data to the SDRmn.DAT[7:0] bits (8 bits) or the SDRmn.DAT[8:0] bits (9 bits) and start communication.

Note: m: Unit number (m = 0), n: Channel number (n = 0, 2), mn = 00, 02

(3) Processing flow (in single transmission mode)

Figure 19.30 shows the timing of UART transmission (in single transmission mode).

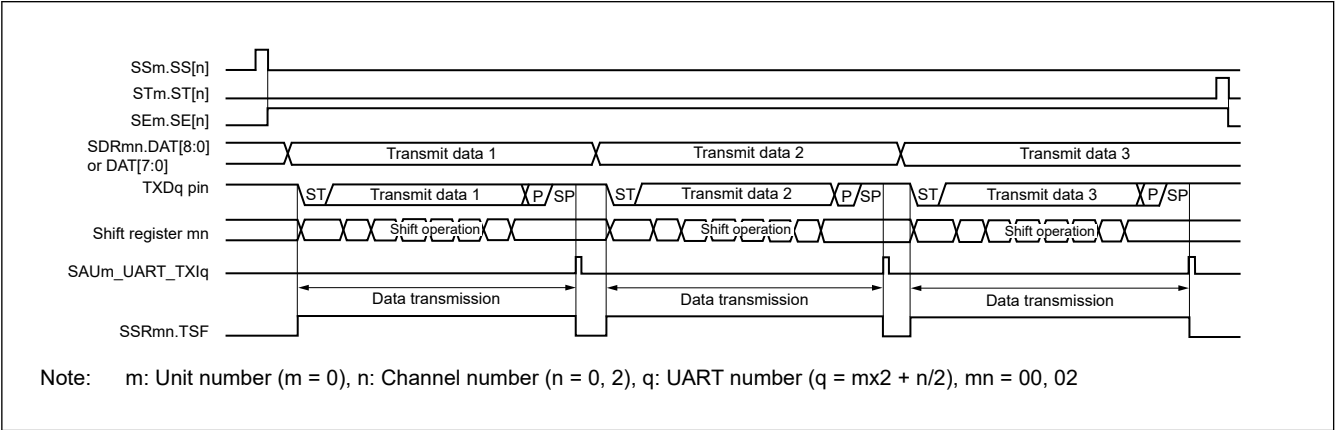


Figure 19.30 Timing of UART transmission (in single transmission mode)

Figure 19.31 shows the flowchart of UART transmission (in single transmission mode).

表 19.81 恢复 UART 传输的步骤

步		过程	细节
程序 恢复UART传输	<1>	恢复的初始设置	—
	<2>	等待通信目标准备就绪	等待通信目标停止或通信操作完成。
	<3>	端口操作	禁用目标通道的数据输出。
	<4>	更改 SPSm 寄存器的设置 (可选)	重置寄存器以更改操作时钟设置。
	<5>	更改 SDRmn 寄存器的设置 (可选)	重置寄存器以更改传输波特率设置 (通过分频操作时钟 (f _{MCK}) 来设置传输时钟)。
	<6>	更改 SMRmn 寄存器的设置 (可选)	重置寄存器以更改串行模式寄存器 mn (SMRmn) 设置。
	<7>	更改 SCRmn 寄存器的设置 (可选)	重置寄存器以更改串行通信操作设置寄存器 mn (SCRmn) 设置。
	<8>	更改 SOLm 寄存器的设置 (选择性)	重置寄存器以更改串行输出电平寄存器 m (SOLm) 设置。
	<9>	更改 SOEm 寄存器的设置 (可选)	将 SOEm.SOE[n] 位清除为 0 并停止输出。
	<10>	更改 SOm 寄存器的设置 (可选)	设置串行数据的初始输出电平(SOm.SO[n])。
	<11>	更改 SOEm 寄存器的设置	将 SOEm.SOE[n] 位设置为 1 并启用输出。
	<12>	端口操作	启用目标通道的数据输出。
	<13>	致函SSm登记处	将目标通道的 SSm.SS[n] 位设置为 1, 并将 SEm.SE[n] 位设置为 1 (以启用操作)。
	<14>	完成恢复设置	设置完成。 将发送数据设置为 SDRmn.DAT[7:0] 位 (8 位) 或 SDRmn.DAT[8:0] 位 (9 位), 并开始通信。

注: m: 单元编号(m = 0) n: 通道编号(n = 0, 2), mn = 00, 02

(3) 处理流程 (单次传输模式)

图 19.30 显示了 UART 传输的时序 (单次传输模式)。

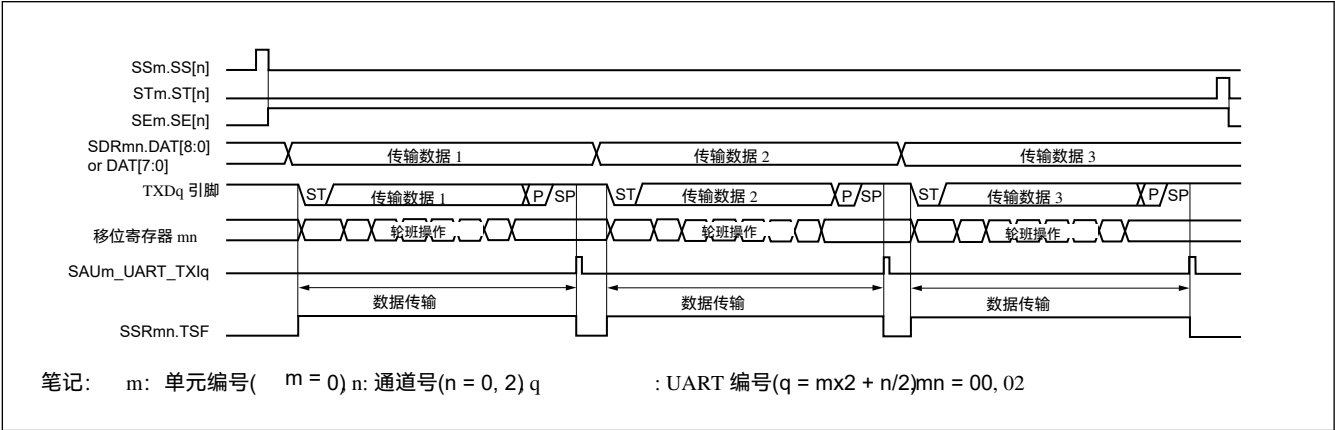


图 19.30 UART 传输时序 (单次传输模式)

图 19.31 显示了 UART 传输流程图 (单次传输模式)。

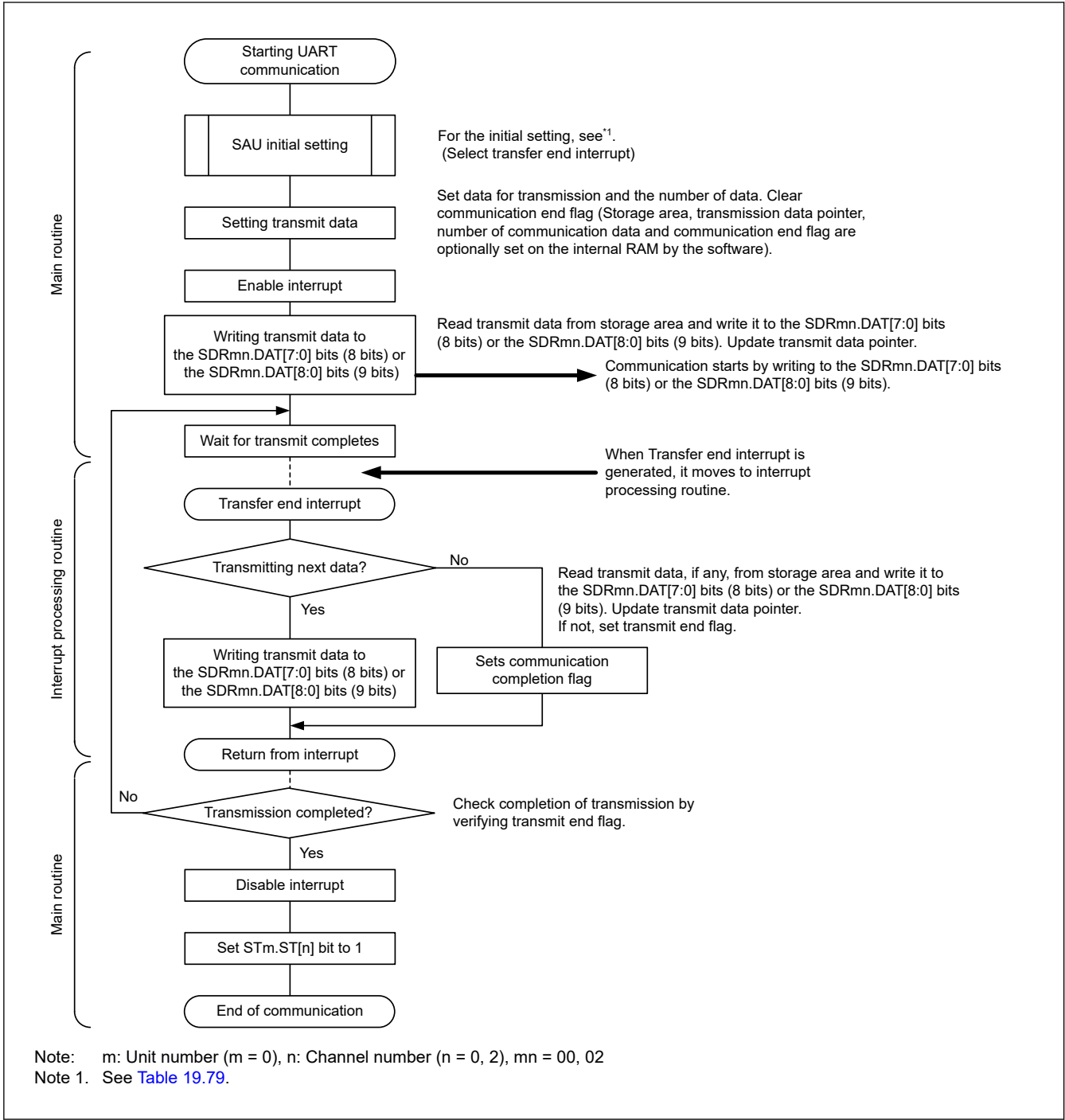


Figure 19.31 Flowchart of UART transmission (in single transmission mode)

(4) Processing flow (in continuous transmission mode)

Figure 19.32 shows the timing of UART transmission (in continuous transmission mode).

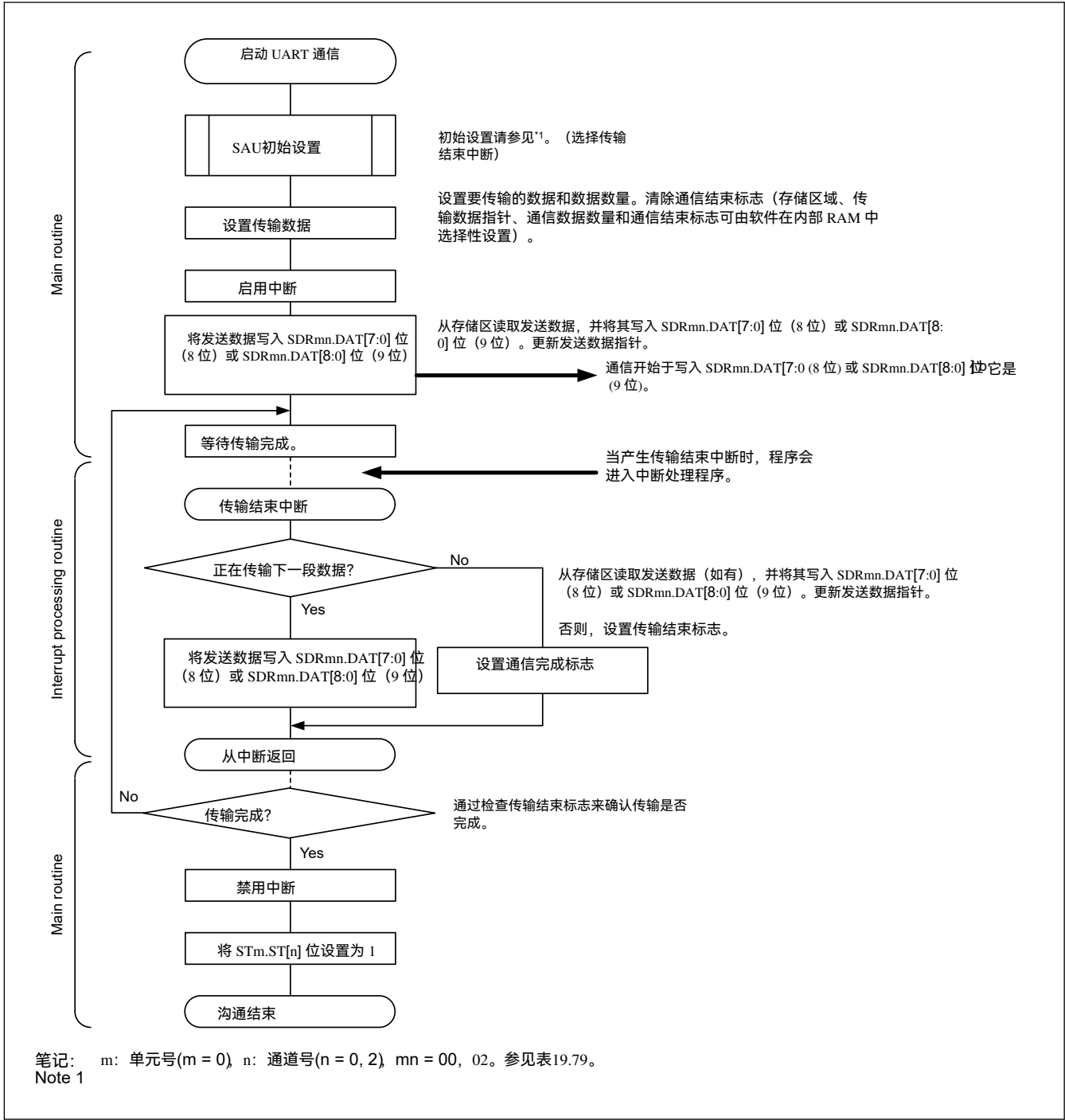


图 19.31 UART 传输流程图 (单次传输模式)

(4) 处理流程 (连续传输模式)

图 19.32 显示了 UART 传输的时序 (在连续传输模式下) 。

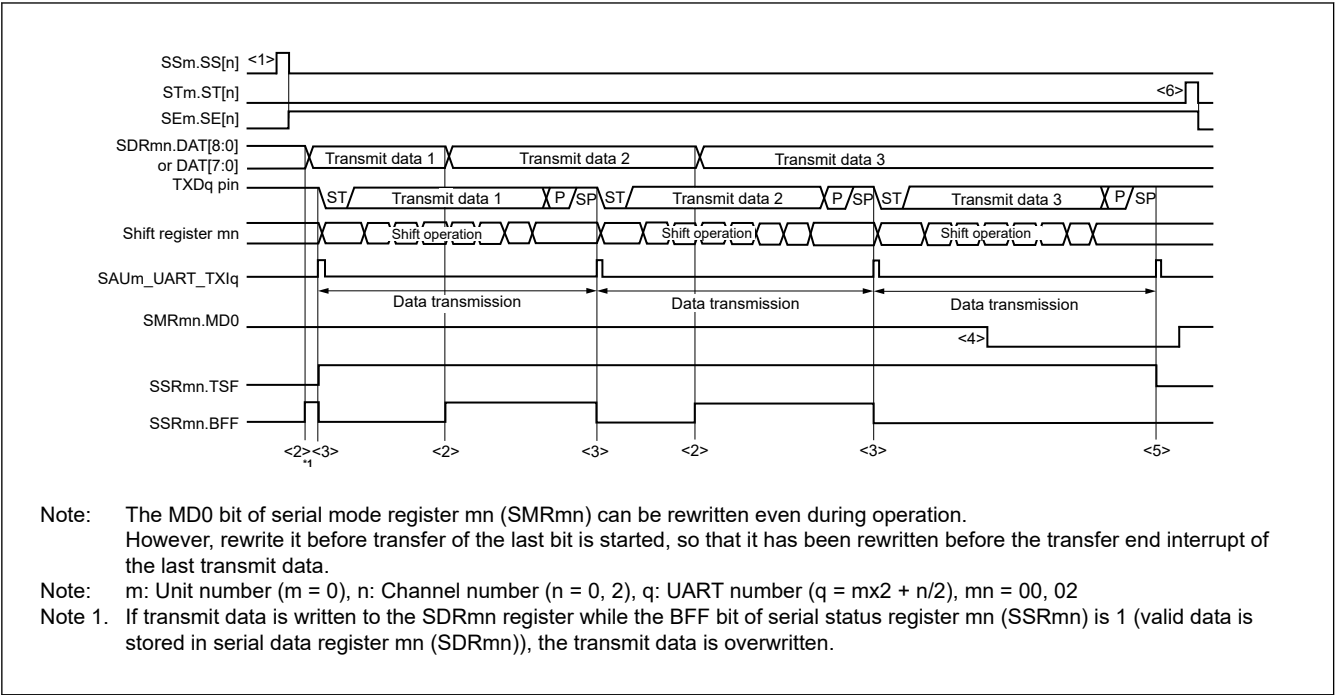


Figure 19.32 Timing of UART transmission (in continuous transmission mode)

Figure 19.33 shows the flowchart of UART transmission (in continuous transmission mode).

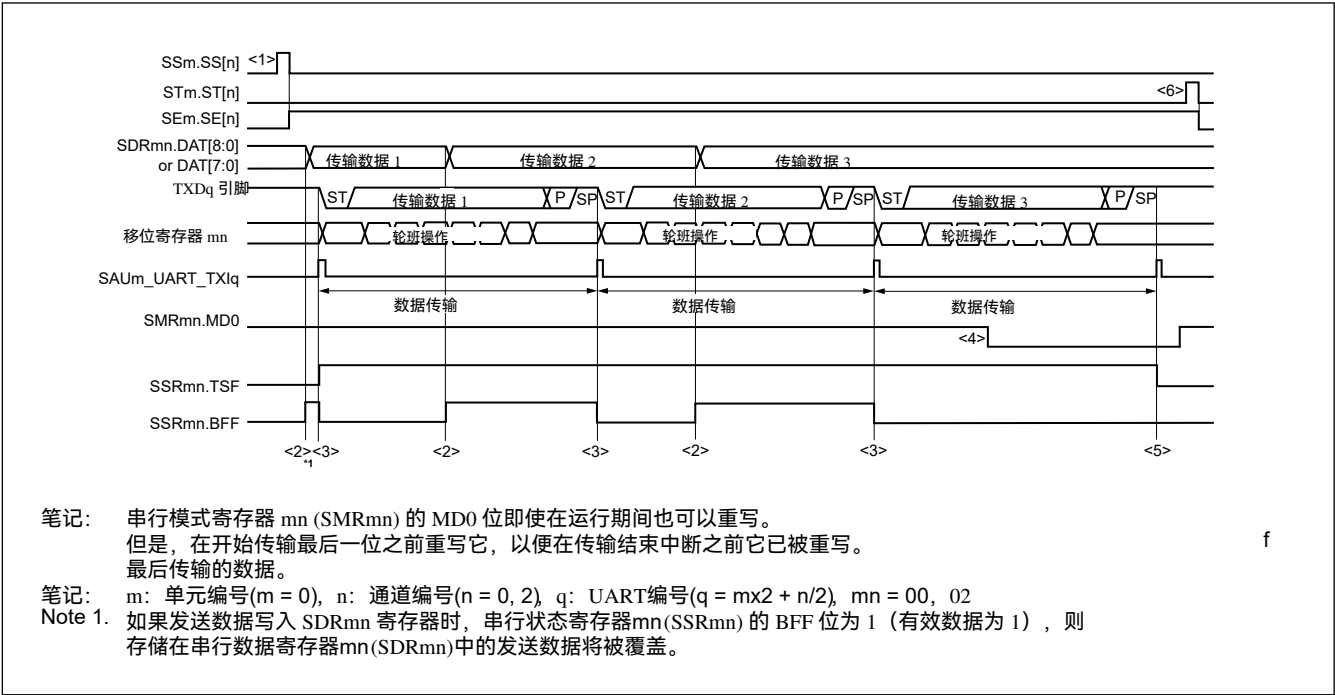


图 19.32 UART 传输时序 (连续传输模式)

图 19.33 显示了 UART 传输流程图 (连续传输模式)。

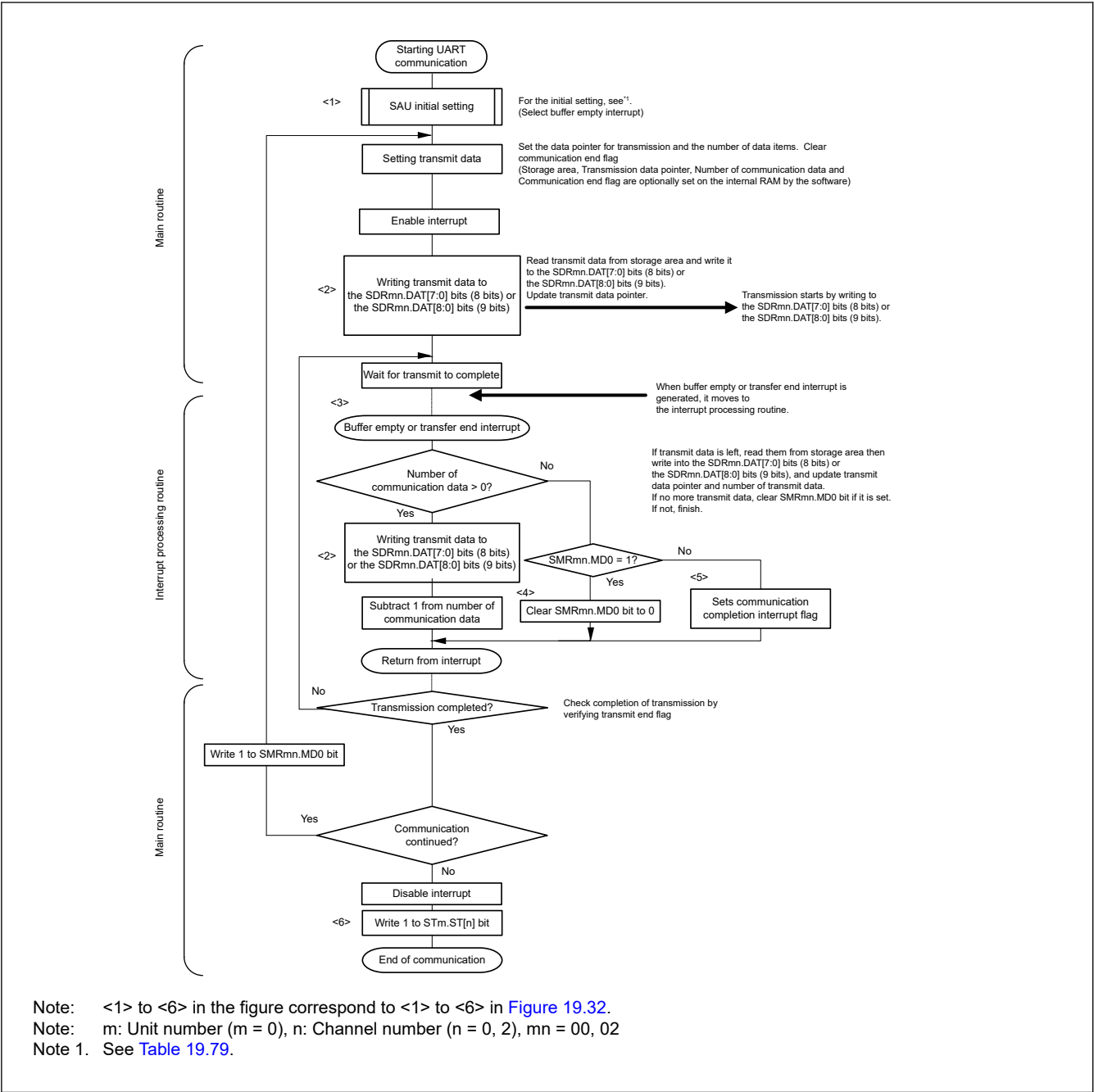


Figure 19.33 Flowchart of UART transmission (in continuous transmission mode)

19.6.2 UART Reception

UART reception is an operation wherein a microcontroller asynchronously receives data from another device (start-stop synchronization).
For UART reception, the odd-number channel of the two channels used for UART is used. The SMRmn register of both the odd- and even-numbered channels must be set.

Table 19.82 shows the specification of UART reception.

Table 19.82 Specification of UART reception (1 of 2)

UART	UART0	UART1
Target channel	Channel 1 of SAU0	Channel 3 of SAU0
Pins used	RXD0	RXD1

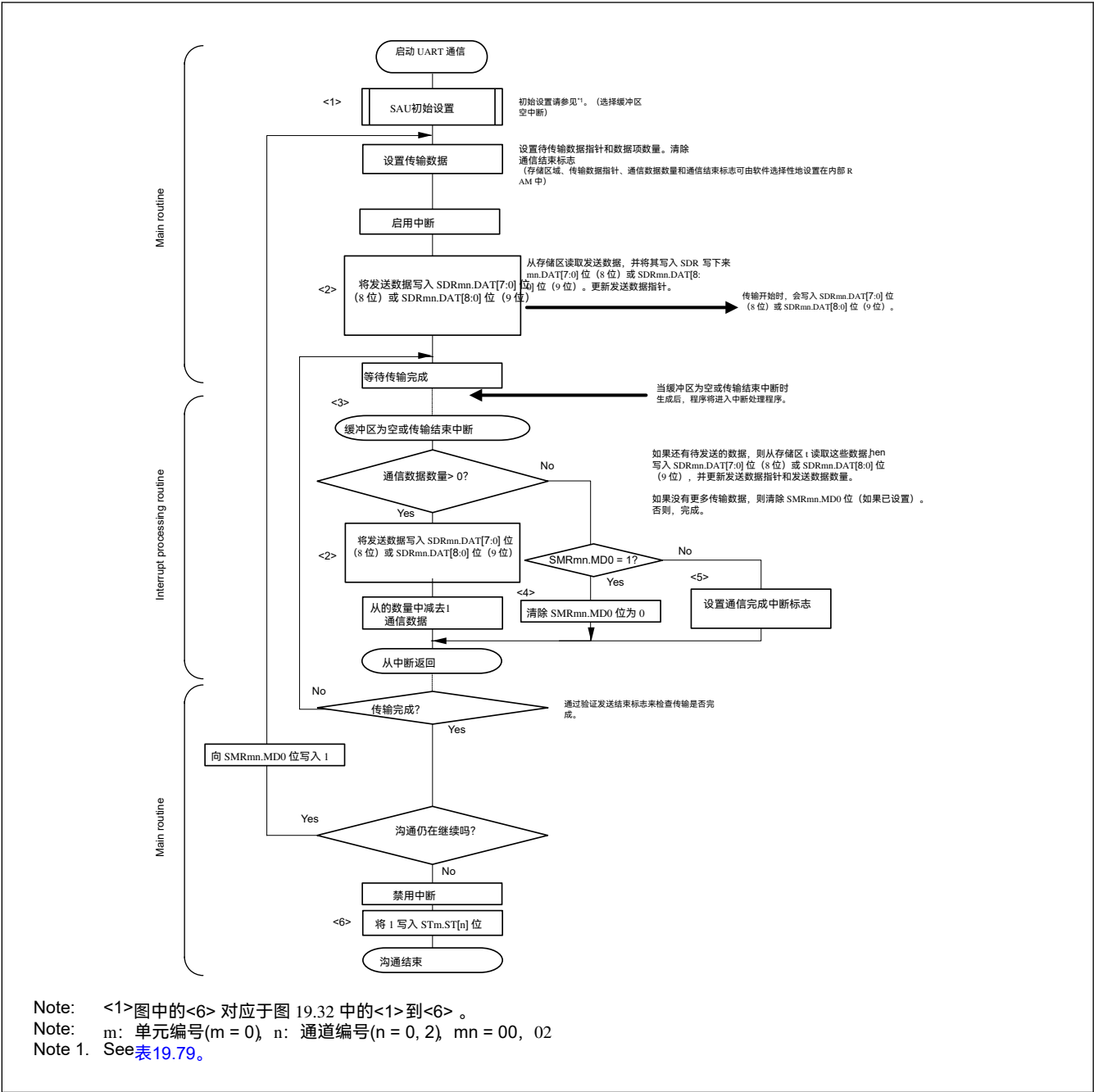


图 19.33 UART 传输流程图（连续传输模式）

19.6.2 UART接收

UART 接收是指微控制器异步接收来自另一个设备的数据（启动-停止同步）的操作。

对于 UART 接收，使用 UART 两个通道中的奇数通道。奇数通道和偶数通道的 SMRmn 寄存器都必须置位。

表 19.82 显示了 UART 接收的规格。

表 19.82 UART 接收规范 (1/2)

UART	UART0	UART1
目标频道	SAU0 频道 1	SAU0 的 3 频道
使用的引脚	RXD0	RXD1

Table 19.82 Specification of UART reception (2 of 2)

UART	UART0	UART1
Interrupt	SAU0_UART_RXI0	SAU0_UART_RXI1
	Transfer end interrupt only (Setting the buffer empty interrupt is prohibited.)	
Error interrupt	SAU0_UART_ERRI0	SAU0_UART_ERRI1
Error detection flag	- Framing error detection flag (SSRmn.FEF) - Parity error detection flag (SSRmn.PEF) - Overrun error detection flag (SSRmn.OVF)	
Transfer data length	7, 8, or 9 bits*1	
Transfer rate*2	Max. f _{MCK} /6 [bps] (SDRmn.STCLK[6:0] = 2 or more), Min. PCLKB/ (2 × 2 ¹⁵ × 128) [bps]	
Data phase	Non-reverse output (default: high level) Reverse output (default: low level)	
Parity bit	The following are selectable: - No parity bit (no parity check) - No parity judgment (0 parity) - Even parity check - Odd parity check	
Stop bit	Appending 1 bit	
Data direction	MSB or LSB first	

Note: f_{MCK}: Operation clock frequency of target channel
f_{SCK}: Serial clock frequency

Note: m: Unit number (m = 0), n: Channel number (n = 1, 3), mn = 01, 03

Note 1. Only UART0 supports the 9-bit data length.

Note 2. Use this operation within a range that satisfies the conditions above and the peripheral functions characteristics specified in the electrical characteristics. For details, see [section 27, Electrical Characteristics](#).

(1) Register setting

[Table 19.83](#) to [Table 19.89](#) show examples of the register contents for UART reception.

(a) Serial mode register mn (SMRmn)

Table 19.83 Example of serial mode register mn (SMRmn) contents for UART reception (1 of 2)

Bit	Symbol	Set value	Function
0	MD0	0	Interrupt source of channel n 0: Transfer end interrupt
2:1	MD1[1:0]	01b	Setting of operation mode of channel n 0 1: UART mode
5:3	—	100b	Setting disabled (set to the initial value)
6	SIS0	0/1	Controls inversion of level of receive data of channel n in UART mode 0: Normal reception 1: Reverse reception
7	—	0	Setting disabled (set to the initial value)
8	STS	1	Selection of start trigger source 1: Valid edge of the RXDq pin
13:9	—	0_0000b	Setting disabled (set to the initial value)
14	CCS	0	Selection of transfer clock (f _{TCLK}) of channel n 0: Divided operation clock f _{MCK} specified by the CKS bit

表 19.82 UART 接收规范 (2/2)

UART	UART0	UART1
打断	SAU0_UART_RXI0	SAU0_UART_RXI1
	仅传输结束中断（设置	缓冲区为空中断是被禁止的。）
错误中断	SAU0_UART_ERRI0	SAU0_UART_ERRI1
错误检测标志	- 帧错误检测标志（SSRmn.FEF） - 奇偶校验错误检测标志（SSRmn.PEF） - 溢出错误检测标志（SSRmn.OVF）	
传输数据长度	7、8 或 9 位*1	
转账费率*2	最大限度。f _{MCK} /6[bps]（SDRmn.STCLK[6:0] = 2 或更多），最小PCLKB/(2 × 2 ¹⁵ × 128)[bps]	
数据阶段	非反向输出（默认值：高电平） 反向输出（默认值：低电平）	
奇偶校验位	可选择以下内容： - 无奇偶校验位（无奇偶校验） - 无平局判定（0 平局） - 偶数奇偶校验 - 奇偶校验	
停止位	追加 1 位	
数据方向	先 MSB 还是 LSB	

注: f_{MCK}: 目标通道的工作时钟频率
f_{SCK}: 串行时钟频率

注: m: 单元编号(m = 0), n: 通道编号(n = 1, 3), mn = 01, 03

注1: 只有UART0支持9位数据长度。

注2: 此操作应在满足上述条件和电气特性中规定的外围功能特性的范围内使用。详情请参见第27节“电气特性”。

(1) 寄存器设置

表 19.83 至表 19.89 显示了 UART 接收的寄存器内容示例。

(a) 串行模式寄存器 mn (SMRmn)

表 19.83 UART 接收的串行模式寄存器 mn (SMRmn) 内容示例 (1/2)

少量	象征	设置值	功能
0	MD0	0	通道 n 的中断源 0: 传输结束中断
2:1	MD1[1:0]	01b	设置通道 n 的工作模式 0 1: UART模式
5:3	—	100b	设置已禁用（已设置为初始值）
6	SIS0	0/1	控制 UART 模式下通道 n 接收数据的电平反转 0: 正常接收 1: 反向接收
7	—	0	设置已禁用（已设置为初始值）
8	STS	1	选择启动触发源 1: RXDq引脚的有效边沿
13:9	—	0_0000b	设置已禁用（已设置为初始值）
14	CCS	0	选择通道 n 的传输时钟 (f _{TCLK}) 0: 由 CKS 位指定的分频操作时钟f _{MCK}

Table 19.83 Example of serial mode register mn (SMRmn) contents for UART reception (2 of 2)

Bit	Symbol	Set value	Function
15	CKS	0/1	Operation clock (f _{MCK}) of channel n 0: Prescaler output clock CKm0 set by the SPSm register 1: Prescaler output clock CKm1 set by the SPSm register

(b) Serial mode register mr (SMRmr)

Table 19.84 Example of serial mode register mr (SMRmr) contents for UART reception

Bit	Symbol	Set value	Function
0	MD0	0	Interrupt source of channel r 0: Transfer end interrupt
2:1	MD1[1:0]	01b	Setting of operation mode of channel r 0 1: UART mode
13:3	—	000_0000_0100 b	Setting disabled (set to the initial value)
14	CCS	0	Selection of transfer clock (f _{TCLK}) of channel r 0: Divided operation clock f _{MCK} specified by the CKS bit
15	CKS	0/1	Operation clock (f _{MCK}) of channel r (same setting value as SMRmn.CKS bit) 0: Prescaler output clock CKm0 set by the SPSm register 1: Prescaler output clock CKm1 set by the SPSm register

(c) Serial communication operation setting register mn (SCRmn)

Table 19.85 Example of serial communication operation setting register mn (SCRmn) contents for UART reception (1 of 2)

Bit	Symbol	Set value	Function
1:0	DLS[1:0]	01b to 11b	Setting of data length 0 1: 9-bit data length 1 0: 7-bit data length 1 1: 8-bit data length
3:2	—	01b	Setting disabled (set to the initial value)
5:4	SLC[1:0]	01b	Setting of stop bit 0 1: Appending 1 bit
6	—	0	Setting disabled (set to the initial value)
7	DIR	0/1	Selection of data transfer sequence in simplified SPI and UART modes 0: Inputs or outputs data with MSB first. 1: Inputs or outputs data with LSB first.
9:8	PTC[1:0]	00b to 11b	Setting of parity bit 0 0: No parity 0 1: Appending 0 parity 1 0: Appending Even parity 1 1: Appending Odd parity

表 19.83 串行模式寄存器mn(SMRmn) 内容示例（UART 接收）（2/2）

少量	象征	设置值	功能
15	CKS	0/1	通道 n 的运行时钟 (f _{MCK}) 0: 预分频器输出时钟CKm0由 SPSm 寄存器设置 1: 预分频器输出时钟 CKm1 由 SPSm 寄存器设置

(b) 串行模式寄存器 mr (SMRmr)

表 19.84 UART 接收的串行模式寄存器 mr (SMRmr) 内容示例

少量	象征	设置值	功能
0	MD0	0	通道 r 的中断源 0: 传输结束中断
2:1	MD1[1:0]	01b	设置通道 r 的工作模式 0 1: UART模式
13:3	—	000_0000_0100 b	设置已禁用（已设置为初始值）
14	CCS	0	通道 r 的传输时钟 (f _{TCLK}) 选择 0: 由 CKS 位指定的分频操作时钟f _{MCK}
15	CKS	0/1	通道 r 的运行时钟(f _{MCK})（与 SMRmn.CKS 位相同的设置值） 0: 预分频器输出时钟CKm0由 SPSm 寄存器设置 1: 预分频器输出时钟 CKm1 由 SPSm 寄存器设置

(c) 串行通信操作设置寄存器 mn (SCRmn)

表 19.85 串行通信操作示例：设置寄存器 mn (SCRmn) 的内容以进行 UART 接收（1/2）

少量	象征	设置值	功能
1:0	DLS[1:0]	01b 至 11b	数据长度设置: 0 1: 9 位数 据长度; 1 0: 7 位数数据长度 ; 1 1: 8 位数数据长度
3:2	—	01b	设置已禁用（已设置为初始值）
5:4	SLC[1:0]	01b	设置停止位 0 1: 追加 1 位
6	—	0	设置已禁用（已设置为初始值）
7	DIR	0/1	简化 SPI 和 UART 模式下数据传输顺序的选择: 0: 以最高有效位 (MS B) 为先进进行输入或输出数据。1: 以最低有效位 (LSB) 为先进进行输入或输出数据。
9:8	PTC[1:0]	00b 至 11b	设置奇偶校验位 0 0: 无奇偶校验 0 1: 附加 0 奇偶校验 1 0: 附加偶奇偶性 1.1: 附加奇偶性

Table 19.85 Example of serial communication operation setting register mn (SCRmn) contents for UART reception (2 of 2)

Bit	Symbol	Set value	Function
10	EOC	0/1	Mask control of error interrupt signal SAUm_UART_ERRIq 0: Disables generation of error interrupt SAUm_UART_ERRIq (SAUm_UART_RXIq is generated). 1: Enables generation of error interrupt SAUm_UART_ERRIq (SAUm_UART_RXIq is not generated if an error occurs).
11	—	0	Setting disabled (set to the initial value)
13:12	DCP[1:0]	00b	Since this bit is dedicated to other modes, it is fixed in the UART mode.
15:14	TRXE[1:0]	01b	Setting TRXE[1:0] = 01b is fixed in the UART reception mode

(d) Serial data register mn (SDRmn)

Table 19.86 Example of serial data register mn (SDRmn) contents for UART reception

Bit	Symbol	Set value	Function
6:0	DAT[6:0]	0x00 to 0x7F	Receive data[6:0]
7	DAT[7]	0/1	Receive data[7] (8-bit and 9-bit data length)
		0	0 Fixed (7-bit data length)
8	DAT[8]*1	0/1	Receive data[8] (9-bit data length)
		0	0 Fixed (7-bit and 8-bit data length)
15:9	STCLK[6:0]	0x02 to 0x7F	Baud rate setting (Operation clock (fMCK) division setting)

Note 1. When UART performs 9-bit communication, bits 0 to 8 of the SDRm1 register are used as the reception data specification area. Only UART0 supports the 9-bit data length.

(e) Serial output register m (SOM)

This register is not used in this mode.

Table 19.87 Example of serial output register m (SOM) contents for UART reception

Bit	Symbol	Set value	Function
n	SO[n]	x	Bit that cannot be used in this mode (set to the initial value when not used in any mode)
n+8	CKO[n]	x	Bit that cannot be used in this mode (set to the initial value when not used in any mode)

(f) Serial output enable register m (SOEm)

This register is not used in this mode.

Table 19.88 Example of serial output enable register m (SOEm) contents for UART reception

Bit	Symbol	Set value	Function
n	SOE[n]	x	Bit that cannot be used in this mode (set to the initial value when not used in any mode)

(g) Serial channel start register m (SSm)

Set only the bit of the target channel to 1.

Table 19.89 Example of serial channel start register m (SSm) contents for UART reception

Bit	Symbol	Set value	Function
n	SS[n]	1	Operation start trigger of channel n 1: Set the SEm.SE[n] bit to 1 to place the channel in the communications waiting state.

表 19.85 串行通信操作示例：设置寄存器 mn (SCRmn) 的内容以进行 UART 接收（2/2）

少量	象征	设置值	功能
10	EOC	0/1	掩码控制错误中断信号 SAUm_UART_ERRIq 0: 禁用错误中断 SAUm_UART_ERRIq 的生成（生成 SAUm_UART_RXIq）。 1: 启用错误中断 SAUm_UART_ERRIq 的生成（如果发生错误，则不会生成 SAUm_UART_RXIq）。
11	—	0	设置已禁用（已设置为初始值）
13:12	DCP[1:0]	00b	由于该位专用于其他模式，因此在 UART 模式下它是固定的。
15:14	TRXE[1:0]	01b	设置 TRXE[1:0] = 01b 在 UART 接收模式下是固定的

(d) 串行数据寄存器 mn (SDRmn)

表 19.86 UART 接收的串行数据寄存器 mn (SDRmn) 内容示例

少量	象征	设置值	功能
6:0	DAT[6:0]	0x00 至 0x7F	接收数据[6:0]
7	DAT[7]	0/1	接收数据[7]（8 位和 9 位数据长度）
		0	0 固定（7 位数据长度）
8	DAT[8]*1	0/1	接收数据[8]（9 位数据长度）
		0	0 固定（7 位和 8 位数据长度）
15:9	STCLK[6:0]	0x02 至 0x7F	波特率设置 (操作时钟(fMCK) 分频设置)

注1. 当UART执行9位通信时，SDRm1寄存器的第0位到第8位用作接收数据规范区域。只有 UART0 支持 9 位数据长度。

(e) 串行输出寄存器 m (SOM)

此寄存器在此模式下不使用。

表 19.87 UART 接收的串行输出寄存器m(SOM) 内容示例

少量	象征	设置值	功能
n	所以[n]	x	此模式下无法使用的位（在任何模式下未使用时，设置为初始值）
n+8	CKO[n]	x	此模式下无法使用的位（在任何模式下未使用时，设置为初始值）

(f) 串行输出使能寄存器 m (SOEm)

此寄存器在此模式下不使用。

表 19.88 UART 接收的串行输出使能寄存器m(SOEm) 内容示例

少量	象征	设置值	功能
n	SOE[n]	x	此模式下无法使用的位（在任何模式下未使用时，设置为初始值）

(g) 串行通道起始寄存器m (SSm)

仅将目标通道的位设置为 1。

表 19.89 UART 接收的串行通道起始寄存器m(SSm) 内容示例

少量	象征	设置值	功能
n	SS[n]	1	通道 n 的操作启动触发 1: 将 SEm.SE[n] 位设置为 1，将信道置于通信等待状态状态。

- Note: For the UART reception, be sure to set the SMRmr register of channel r to UART transmission mode that is to be paired with channel n.
- Note: m: Unit number (m = 0), n: Channel number (n = 1, 3), mn = 01, 03
r: Channel number (r = n – 1), q: UART number (q = m × 2 + n/2)
- Note: ×: Bit that cannot be used in this mode (set to the initial value when not used in any mode)
0/1: Set to 0 or 1 depending on the usage of the user

(2) Operation procedure

Table 19.90 shows the procedure for initial setting for UART Reception.

Table 19.90 Initial setting procedure for UART reception

Step		Process	Detail
Procedure for initial setting of UART reception	<1>	Starting initial setting	—
	<2>	Setting the SPSm register	Set the operation clock.
	<3>	Setting the SMRmn and SMRmr registers	Set an operation mode.
	<4>	Setting the SCRmn register	Set a communication format.
	<5>	Setting the SDRmn register	Set a transfer baud rate (setting the transfer clock by dividing the operation clock (f _{MCK})).
	<6>	Setting port	Enable data input of the target channel.
	<7>	Writing to the SSm register	Set the SSm.SS[n] bit of the target channel to 1 and set the SEm.SE[n] bit to 1 to enable operation. Wait for start bit detection.
	<8>	Completing initial setting	—

- Note: Set the TRXEmn[0] bit of SCRmn register to 1, and then be sure to set SSm.SS[n] to 1 after at least 4 f_{MCK} clock cycles have elapsed.
- Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3)

Table 19.91 shows the procedure for stopping UART reception.

Table 19.91 Procedure for stopping UART reception

Step		Process	Detail
Procedure for stopping UART transmission	<1>	Starting setting to stop	—
	<2>	Wait until SSRmn.TSF is cleared (optional)	If there is any data being transferred, wait for its completion. If there is a requirement to stop, do not wait.
	<3>	Writing the STm register	Write 1 to the STm.ST[n] bit of the target channel and set SEm.SE[n] = 0 to stop operation.
	<4>	Stop setting is completed	After the stop setting is completed, go to the next processing.

- Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3)

Table 19.92 shows the procedure for resuming UART Reception.

注意: 对于 UART 接收, 请务必将通道 r 的 SMRmr 寄存器设置为与通道 n 配对的 UART 发送模式。

- 注: m: 单元编号(m = 0) n: 通道编号(n = 1, 3), mn = 01, 03
r: 通道号(r = n – 1) q: UART 号(q = m × 2 + n/2)

注意: ×: 此模式下不能使用的位 (在任何模式下未使用时设置为初始值)
0/1: 根据用户使用情况设置为 0 或 1

(2) 操作流程

表 19.90 显示了 UART 接收的初始设置步骤。

表 19.90 UART 接收的初始设置步骤

步		过程	细节
UART接收初始设置步骤	<1>	初始设置	—
	<2>	设置 SPSm 寄存器	设置运行时钟。
	<3>	设置 SMRmn 和 SMRmr 寄存器	设置操作模式。
	<4>	设置 SCRmn 寄存器	制定沟通模式。
	<5>	设置 SDRmn 寄存器	设置传输波特率 (通过分频操作时钟来设置传输时钟(f _{MCK}))。
	<6>	设置端口	启用目标通道的数据输入。
	<7>	致函SSm登记处	将目标通道的 SSm.SS[n] 位设置为 1, 并将 SEm.SE[n] 位设置为 1 以启用操作。等待起始位检测。
	<8>	完成初始设置	—

注意: 将 SCRmn 寄存器的 TRXEmn [0] 位设置为 1, 然后确保在至少经过 4 个时钟周期后将 SSm.SS[n] 设置为 1。

- 注: m: 单元编号(m = 0) n: 通道编号(n = 0 至3)

表 19.91 显示了停止 UART 接收的步骤。

表 19.91 停止 UART 接收的步骤

步		过程	细节
程序停止UART传输	<1>	开始设置停止	—
	<2>	等待 SSRmn.TSF 被清除 (可选)	如果有数据传输正在进行, 请等待传输完成。如果需要停止传输, 请不要等待。
	<3>	写入 STm 寄存器	将 1 写入目标通道的 STm.ST[n] 位, 并设置SEm.SE[n] = 0 以停止操作。
	<4>	停止设置完成	停止设置完成后, 进行下一步处理。

- 注: m: 单元编号(m = 0), n: 通道编号(n = 0 至3)

表 19.92 显示了恢复 UART 接收的步骤。

Table 19.92 Procedure for resuming UART reception

Step		Process	Detail
Procedure for resuming UART reception	<1>	Starting setting for resumption	—
	<2>	Wait until the communication target is ready	Wait until the communication target stops or communication operation completed
	<3>	Changing setting of the SPSm register (optional)	Reset the register to change the operation clock setting.
	<4>	Changing setting of the SDRmn register (optional)	Reset the register to change the transfer baud rate setting (setting the transfer clock by dividing the operation clock (f_{MCK})).
	<5>	Changing setting of the SMRmn and SMRmr registers (optional)	Reset the registers to change serial mode registers mn, mr (SMRmn, SMRmr) setting.
	<6>	Changing setting of the SCRmn register (optional)	Reset the register to change the serial communication operation setting register mn (SCRmn) setting.
	<7>	Clearing error flag	If the SSRmn.FEF, PEF, and OVF flags remain set, clear them using serial flag clear trigger register mn (SIRmn).
	<8>	Setting port	Enable data input of the target channel.
	<9>	Writing to the SSm register	Set the SSm.SS[n] bit of the target channel to 1 and set the SEm.SE[n] bit to 1 to enable operation. Wait for start bit detection.
	<10>	Completing resumption setting	—

Note: Set the TRXE[0] bit of SCRmn register to 1, and then be sure to set SSm.SS[n] to 1 after at least 4 f_{MCK} clocks have elapsed.
Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3)

(3) Processing flow

Figure 19.34 shows the timing of UART reception.

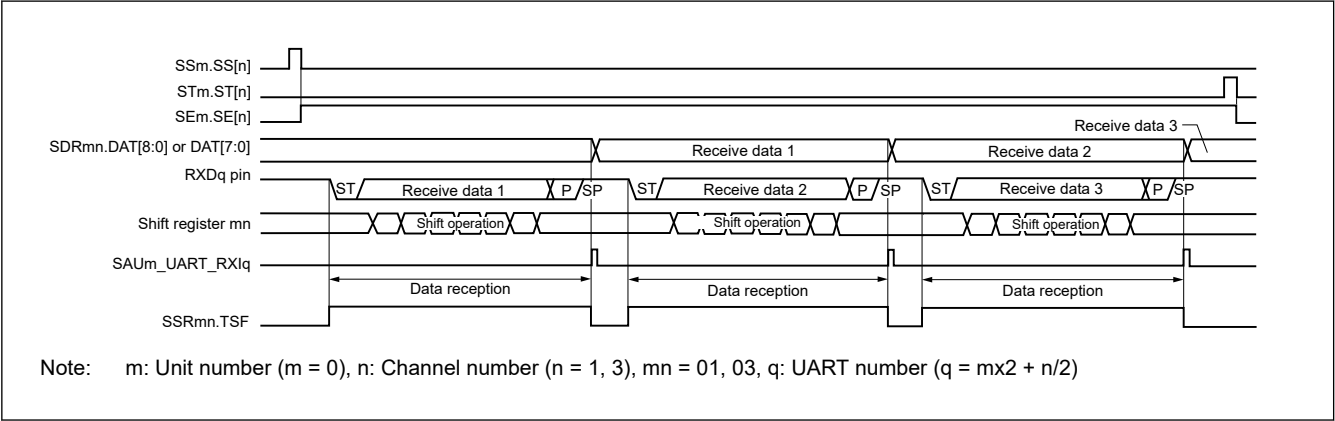


Figure 19.34 Timing of UART reception

Figure 19.35 shows the flowchart of UART reception.

表 19.92 恢复 UART 接收的步骤

步		过程	细节
恢复 UART 接收的步骤	<1>	恢复的初始设置	—
	<2>	等待通信目标准备就绪	等待通信目标停止或通信操作完成。
	<3>	更改 SPSm 寄存器的设置 (可选)	重置寄存器以更改操作时钟设置。
	<4>	更改 SDRmn 寄存器的设置 (可选)	重置寄存器以更改传输波特率设置 (通过分频操作时钟 (f_{MCK}) 来设置传输时钟)。
	<5>	改变 SMRmn 的设置和 SMRmr 寄存器 (可选)	重置寄存器以更改串行模式寄存器 mn、mr (SMRmn, SMRmr) 设置。
	<6>	更改 SCRmn 寄存器的设置 (可选)	重置寄存器以更改串行通信操作设置寄存器 mn (SCRmn) 设置。
	<7>	清除错误标志	如果 SSRmn.FEF、PEF 和 OVF 标志仍然设置, 则使用串行标志清除触发寄存器 mn (SIRmn) 清除它们。
	<8>	设置端口	启用目标通道的数据输入。
	<9>	致函SSm登记处	将目标通道的 SSm.SS[n] 位设置为 1, 并将 SEm.SE[n] 位设置为 1 以启用操作。等待起始位检测。
	<10>	完成恢复设置	—

注意: 将 SCRmn 寄存器的 TRXE [0] 位设置为 1, 然后确保在至少经过 4 个时钟周期后将 SSm.SS[n] 设置为 1。
注: m: 单元编号(m = 0), n: 通道编号(n = 0 至3)

(3) 加工流程

图 19.34 显示了 UART 接收的时序。

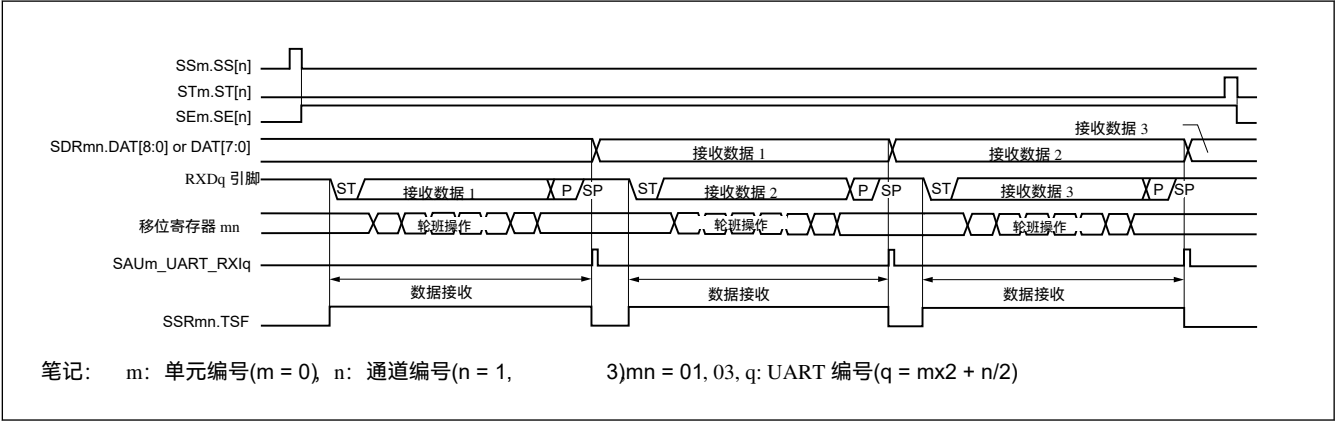


图 19.34 UART 接收时序

图 19.35 显示了 UART 接收的流程图。

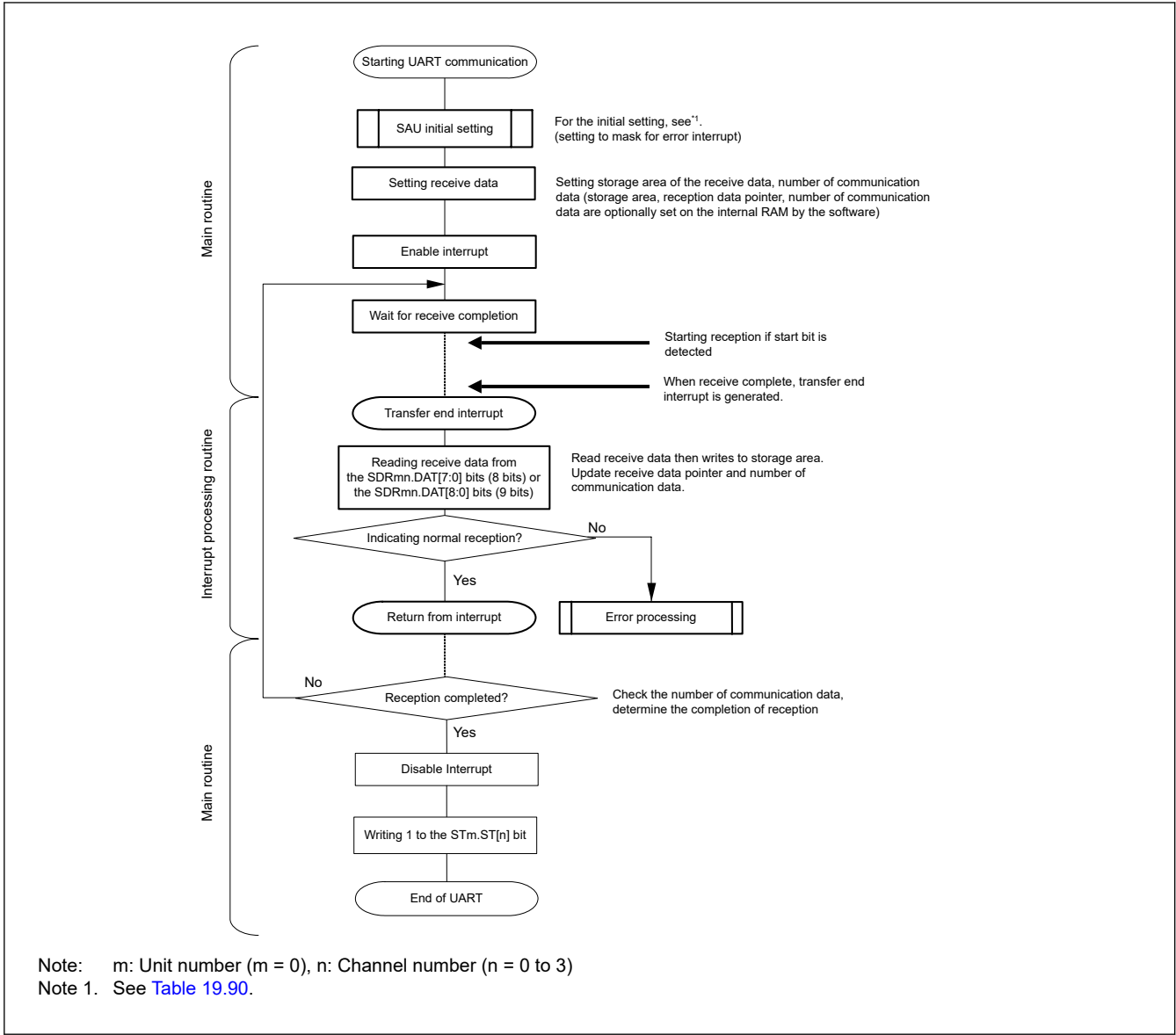


Figure 19.35 Flowchart of UART reception

19.6.3 Snooze Mode Function

The Snooze mode makes the UART perform reception operations on RXD0 pin input detection while in the Software Standby mode. Normally the UART stops communication in the Software Standby mode. However, using the Snooze mode enables the UART to perform reception operations without CPU operation.

Only UART0 channel can be set to Snooze mode.

When using UART0 in the Snooze mode, make the following settings before entering the Software Standby mode. (See Figure 19.38 and Figure 19.40.)

- In the Snooze mode, the baud rate setting for UART reception needs to be changed to a value different from that in normal operation. Set the SPS0 register and the SDR01.STCLK[6:0] bits with reference to Table 19.93.
- Set the SCR01.EOC and SSC0.SSEC bits. This is for enabling or stopping generation of an error interrupt (SAU0_UART_ERRIO) when a communication error occurs.
- When using the Snooze mode function, set the SWC bit of serial standby control register 0 (SSC0) to 1 just before switching to the Software Standby mode. After the initial setting has been completed, set the SS[1] bit of serial channel start register 0 (SS0) to 1.
- A UART0 starts reception in Snooze mode on detecting input of the start bit on the RXD0 pin following a transition of the CPU to the Software Standby mode.

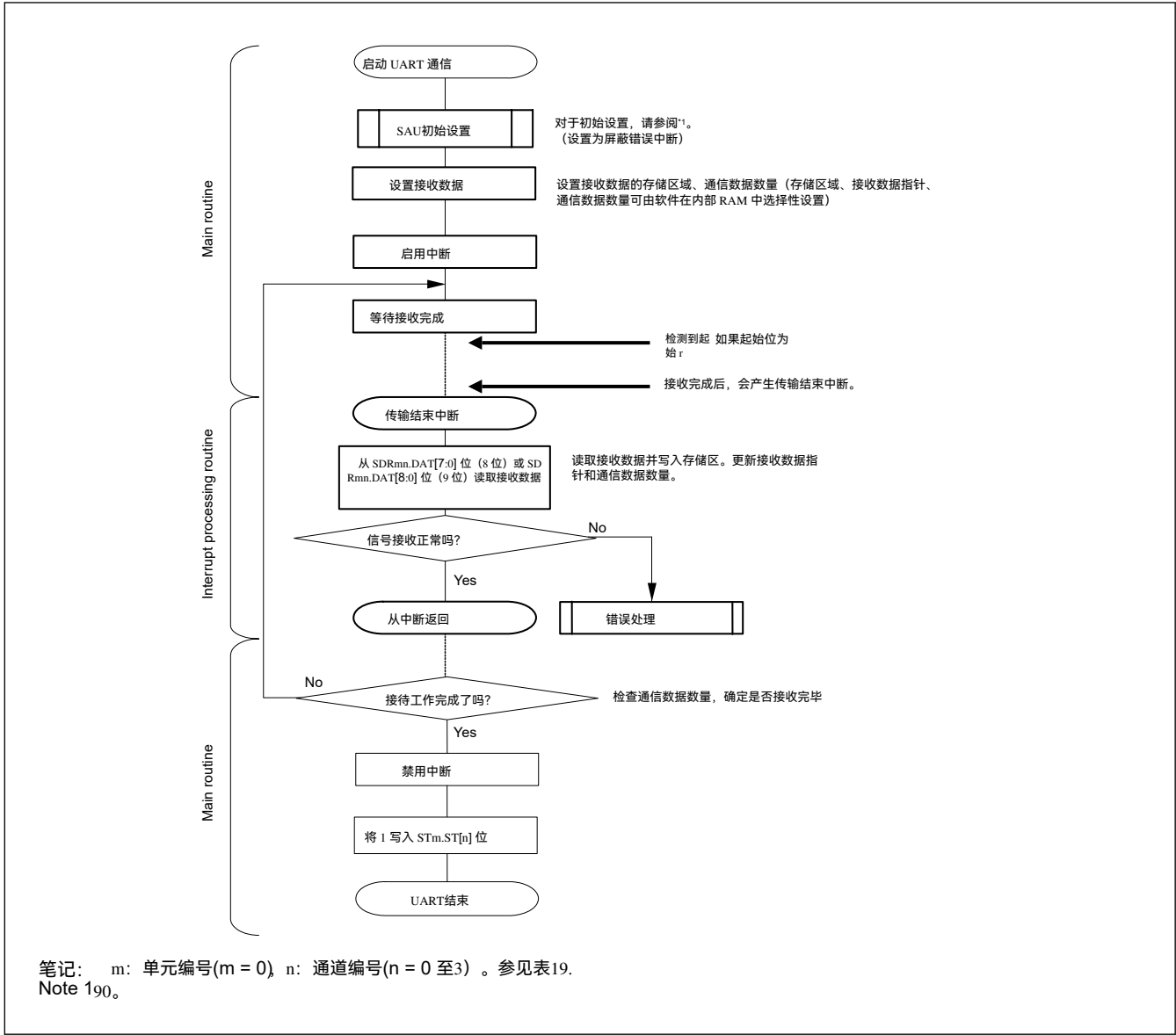


图 19.35 UART 接收流程图

19.6.3 贪睡模式功能

休眠模式使 UART 在软件待机模式下, 通过 RXD0 引脚输入检测来执行接收操作。通常情况下, UART 在软件待机模式下会停止通信。但是, 休眠模式允许 UART 在无需 CPU 操作的情况下执行接收操作。

只有 UART0 通道可以设置为休眠模式。

在休眠模式下使用 UART0 时, 请在进入软件待机模式之前进行以下设置。(参见图 19.38 和图 19.40。)

- 在睡眠模式下, UART接收的波特率设置需要更改为与正常操作时不同的值。参考表19.93设置SPS0寄存器和SD R01.STCLK[6:0] 位。
- 设置 SCR01.EOC 和 SSC0.SSEC 位。这用于在发生通信错误时启用或停止生成错误中断 (SAU0_UART_ERRIO)。
- 使用休眠模式功能时, 在切换到软件待机模式之前, 将串行待机控制寄存器0 (SSC0) 的SWC位设置为1。初始设置完成后, 将串行通道起始寄存器0 (SS0) 的SS[1]位设置为1。
- CPU 进入软件待机模式后, 当检测到 RXD0 引脚上的起始位输入时, UART0 开始以休眠模式接收数据。

Note: The Snooze mode can only be used when the high-speed on-chip oscillator clock is selected for PCLKB.

Note: The maximum transfer rate in the Snooze mode is 9600 bps (when setting the PCLKB = HOCO (32 MHz)).

Note: When SSC0.SWC = 1, UART0 can be used only when the reception operation is started in the Software Standby mode.

When used simultaneously with another Snooze mode function or interrupt, if the reception operation is started in a state other than the Software Standby mode, such as those given below, data may not be received correctly and a framing error or parity error may be generated.

- When after the SSC0.SWC bit has been set to 1, the reception operation is started before the Software Standby mode is entered
- When the reception operation is started while another function is in the Snooze mode
- When after returning from the Software Standby mode to normal operation due to an interrupt or other cause, the reception operation is started before the SSC0.SWC bit is returned to 0

Note: If a parity error, framing error, or overrun error occurs while the SSC0.SSEC bit is set to 1, the SSR01.PEF, FEF, or OVF flag is not set and an error interrupt (SAU0_UART_ERRI0) is not generated. Therefore, when the setting of SSC0.SSEC = 1 is made, clear the SSR01.PEF, FEF, and OVF before setting the SSC0.SWC bit to 1 and read the value in bits 7 to 0 of the SDR01 register.

Note: The CPU shifts from the Software Standby mode to the Snooze mode on detecting the valid edge of the RXD0 signal.

Note, however, that transfer through the UART channel may not start and the CPU may remain in the Snooze mode if an input pulse on the RXD0 pin is too short to be detected as a start bit. In such cases, data may not be received correctly, and this may lead to a framing error or parity error in the next UART transfer.

Table 19.93 shows the baud rate setting for UART reception in Snooze mode.

Table 19.93 Baud rate setting for UART reception in Snooze mode

Baud rate	High-speed on-chip oscillator (HOCO)	Operating clock (fMCK)	SDR01.STCLK [6:0]	Maximum permissible value	Minimum permissible value
4800 bps	32 MHz ± 1%*1	PCLKB/2 ⁵	106	1.45%	-1.67%
9600 bps	32 MHz ± 1%*1	PCLKB/2 ⁴	106	1.45%	-1.67%

Note 1. When the accuracy of the clock frequency of the high-speed on-chip oscillator is ±1.5% or ±2.0%, the permissible range becomes smaller as shown below.

- In the case of HOCO ±1.5%, perform (Maximum permissible value – 0.5%) and (Minimum permissible value + 0.5%) to the values in the above table.
- In the case of HOCO ±2.0%, perform (Maximum permissible value – 1.0%) and (Minimum permissible value + 1.0%) to the values in the above table.

(1) Snooze mode operation (SCR01.EOC = 0, SSC0.SSEC = 0/1)

Because of the setting of SCR01.EOC = 0, even though a communication error occurs, an error interrupt (SAU0_UART_ERRI0) is not generated, regardless of the setting of the SSC0.SSEC bit. However, a transfer end interrupt (SAU0_UART_RXI0) is generated.

Figure 19.36 shows the timing of Snooze mode operation (SCR01.EOC = 0, SSC0.SSEC = 0/1).

注意：只有当选择高速片上振荡器时钟作为 PCLKB 时，才能使用睡眠模式。

注意：在 Snooze 模式下，最大传输速率为 9600 bps（当设置 PCLKB = HOCO (32 MHz)时）。

注意：当 SSC0.SWC = 1时，只有在软件待机模式下启动接收操作时才能使用 UART0。

当与另一个贪睡模式功能或中断同时使用时，如果接收操作在软件待机模式以外的状态（例如以下状态）启动，则可能无法正确接收数据，并可能产生帧错误或奇偶校验错误。

- 当 SSC0.SWC 位被设置为 1 后，接收操作会在进入软件待机模式之前启动。
- 当另一个功能处于休眠模式时，如果开始接收操作
- 当因中断或其他原因从软件待机模式恢复正常运行后，在 SSC0.SWC 位返回 0 之前就开始接收操作时

注意：如果在 SSC0.SSEC 位设置为 1 时发生奇偶校验错误、帧错误或溢出错误，则 SSR01.PEF、FEF 或 OVF 标志不会被置位，也不会产生错误中断 (SAU0_UART_ERRI0)。因此，在设置 SSC0.SSEC = 1 时，应先清除 SSR01.PEF、FEF 和 OVF 标志，然后再将 SSC0.SWC 位设置为 1，并读取 SDR01 寄存器第 7 位到第 0 位的值。

注意：当检测到 RXD0 信号的有效边沿时，CPU 将从软件待机模式切换到休眠模式。

但请注意，如果 RXD0 引脚上的输入脉冲太短，无法被检测到作为起始位，则 UART 通道的传输可能无法启动，CPU 也可能保持在休眠模式。在这种情况下，数据可能无法正确接收，这可能会导致下一次 UART 传输出现帧错误或奇偶校验错误。

表 19.93 显示了 UART 接收在休眠模式下的波特率设置。

表 19.93 休眠模式下 UART 接收的波特率设置

波特率	高速片上振荡器 (HOCO)	运行时钟 (fMCK)	SDR01.STCLK [6:0]	最大允许值	最低允许值
4800 bps	32 MHz ± 1%*1	PCLKB/2 ⁵	106	1.45%	-1.67%
9600 bps	32 MHz ± 1%*1	PCLKB/2 ⁴	106	1.45%	-1.67%

注1. 当高速片上振荡器的时钟频率精度为±1.5%或±2.0%时，允许范围会变小，如下图所示。

- 对于HOCO ±1.5%，执行（最大允许值-0.5%）和（最小允许值+ 0.5%））到上表中的值。
- 对于 HOCO±2.0%，执行（最大允许值-1.0%）和（最小允许值+ 1.0%））到上表中的值。

(1) 贪睡模式操作(SCR01.EOC = 0,SSC0.SSEC = 0/1)

由于SCR01.EOC = 0的设置，即使发生通信错误，也不会生成错误中断 (SAU0_UART_ERRI0)，而与 SSC0.SSEC 位的设置无关。但是，会生成传输结束中断 (SAU0_UART_RXI0)。

图 19.36 显示了 Snooze 模式操作的时序(SCR01.EOC = 0, SSC0.SSEC = 0/1)。

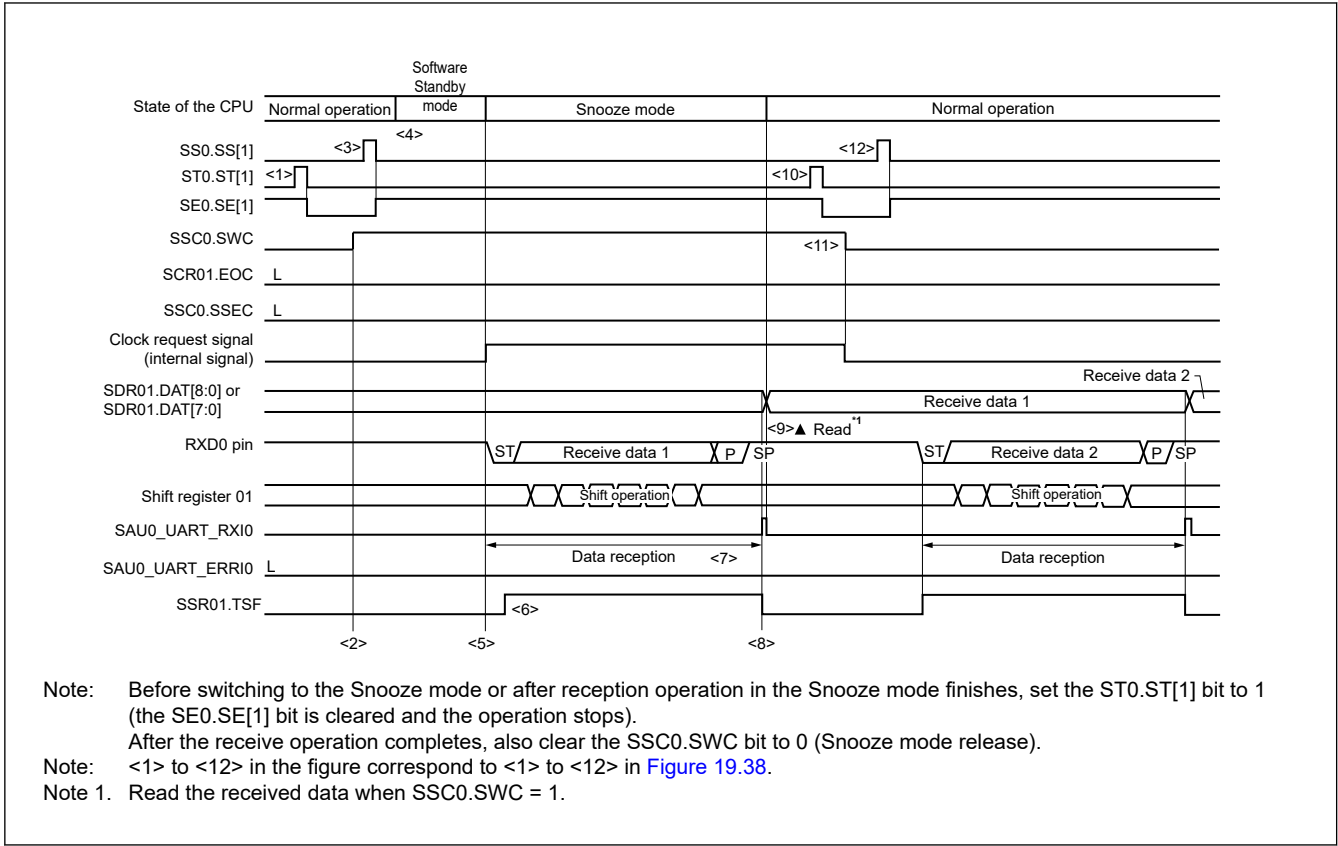


Figure 19.36 Timing of Snooze mode operation (SCR01.EOC = 0, SSC0.SSEC = 0/1)

(2) Snooze mode operation (SCR01.EOC = 1, SSC0.SSEC = 0: Error interrupt (SAU0_UART_ERRI0) generation is enabled)

Because SCR01.EOC = 1 and SSC0.SSEC = 0, an error interrupt (SAU0_UART_ERRI0) is generated when a communication error occurs.

Figure 19.37 shows the timing of Snooze mode operation (SCR01.EOC = 1, SSC0.SSEC = 0).

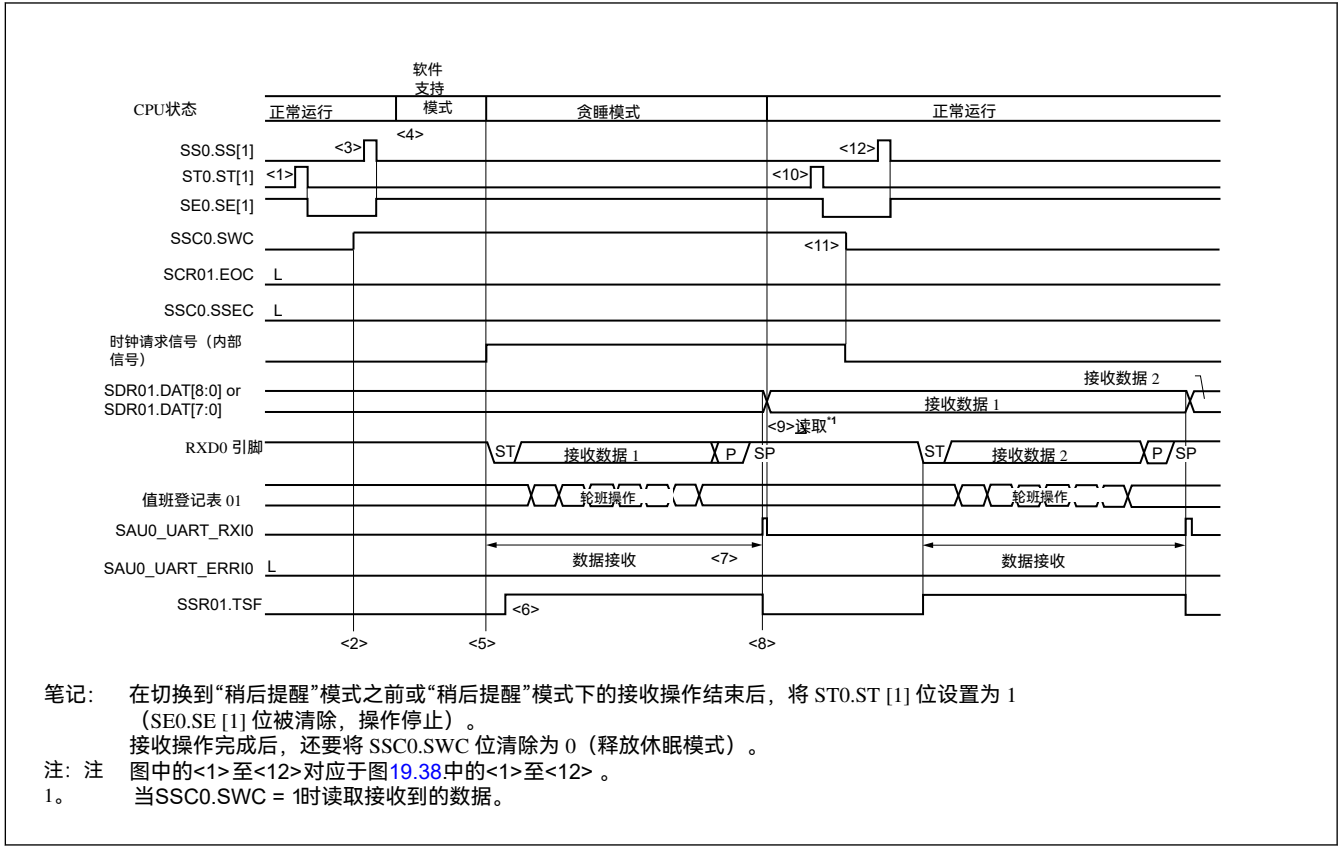


图 19.36 贪睡模式操作的时序 (SCR01.EOC= 0, SSC0.SSEC= 0/1)

(2) 休眠模式操作(SCR01.EOC = 1, SSC0.SSEC = 0: 错误中断 (SAU0_UART_ERRI0) 生成已启用)

因为SCR01.EOC = 1 和SSC0.SSEC = 0, 当发生通信错误时, 会产生错误中断 (SAU0_UART_ERRI0)。

图 19.37 显示了贪睡模式操作(SCR01.EOC = 1,SSC0.SSEC = 0)的时序。

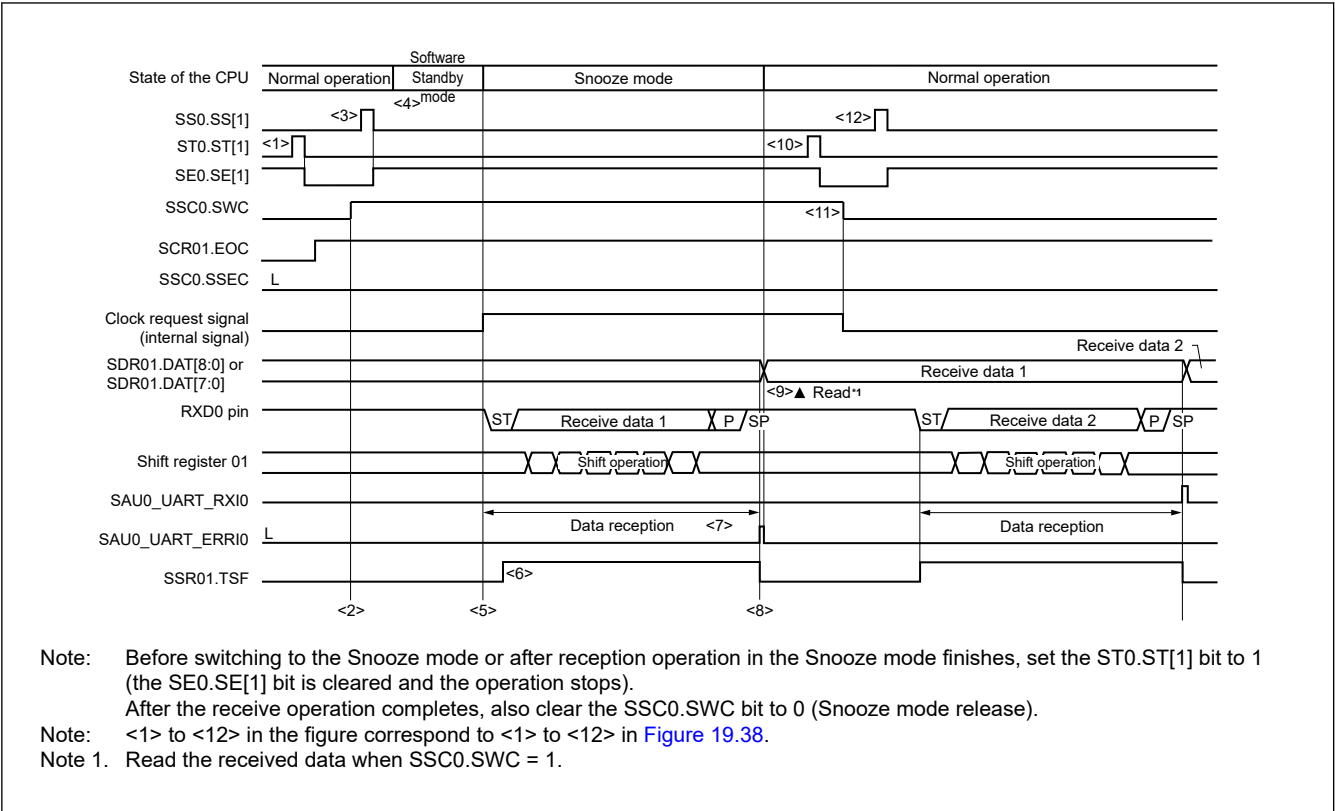


Figure 19.37 Timing of Snooze mode operation (SCR01.EOC = 1, SSC0.SSEC = 0)

Figure 19.38 shows the flowchart of Snooze mode operation (SCR01.EOC = 0, SSC0.SSEC = 0/1 or SCR01.EOC = 1, SSC0.SSEC = 0).

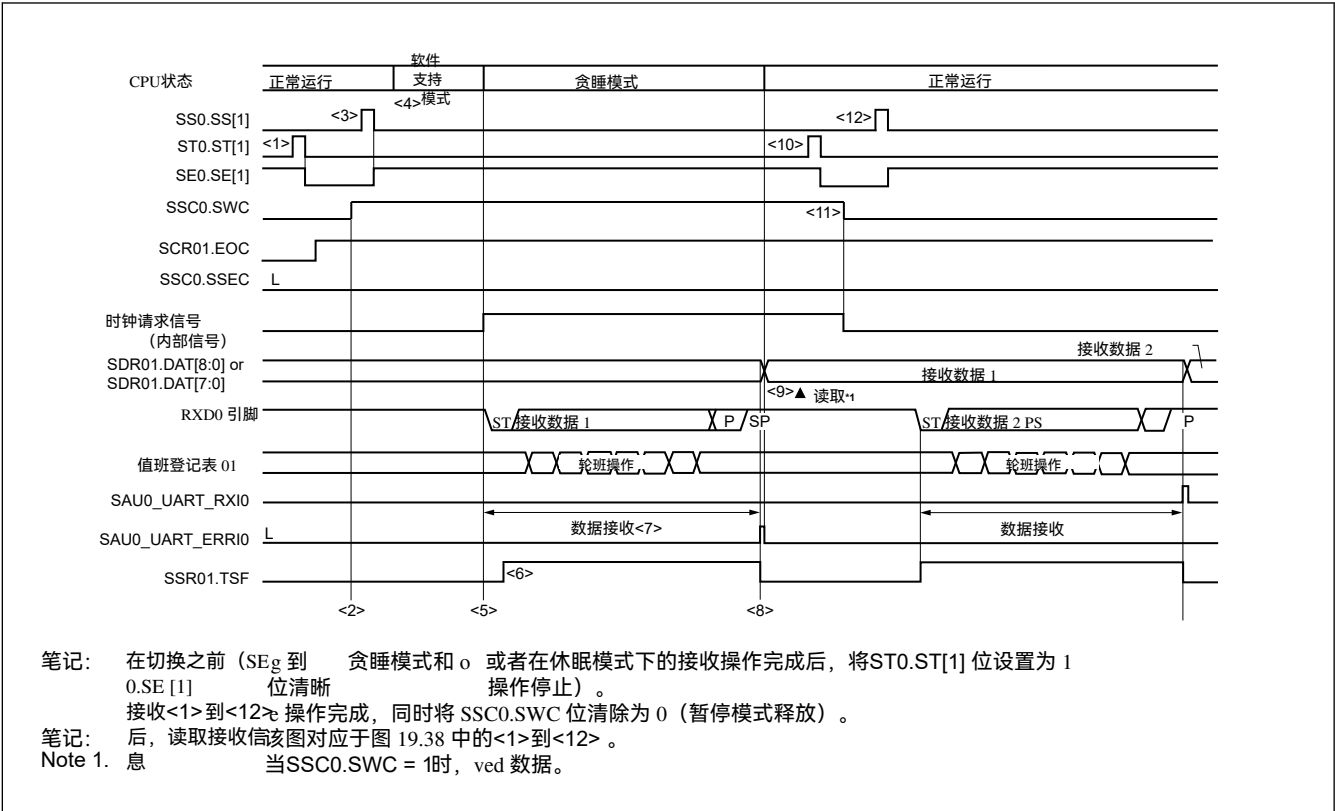


图 19.37 贪睡模式操作的时序(SCR01.EOC = 1, SSC0.SSEC = 0)

图 19.38 显示了贪睡模式操作(SCR01.EOC = 0, SSC0.SSEC = 0/1或SCR01.EOC = 1,SSC0.SSEC = 0)的流程图。

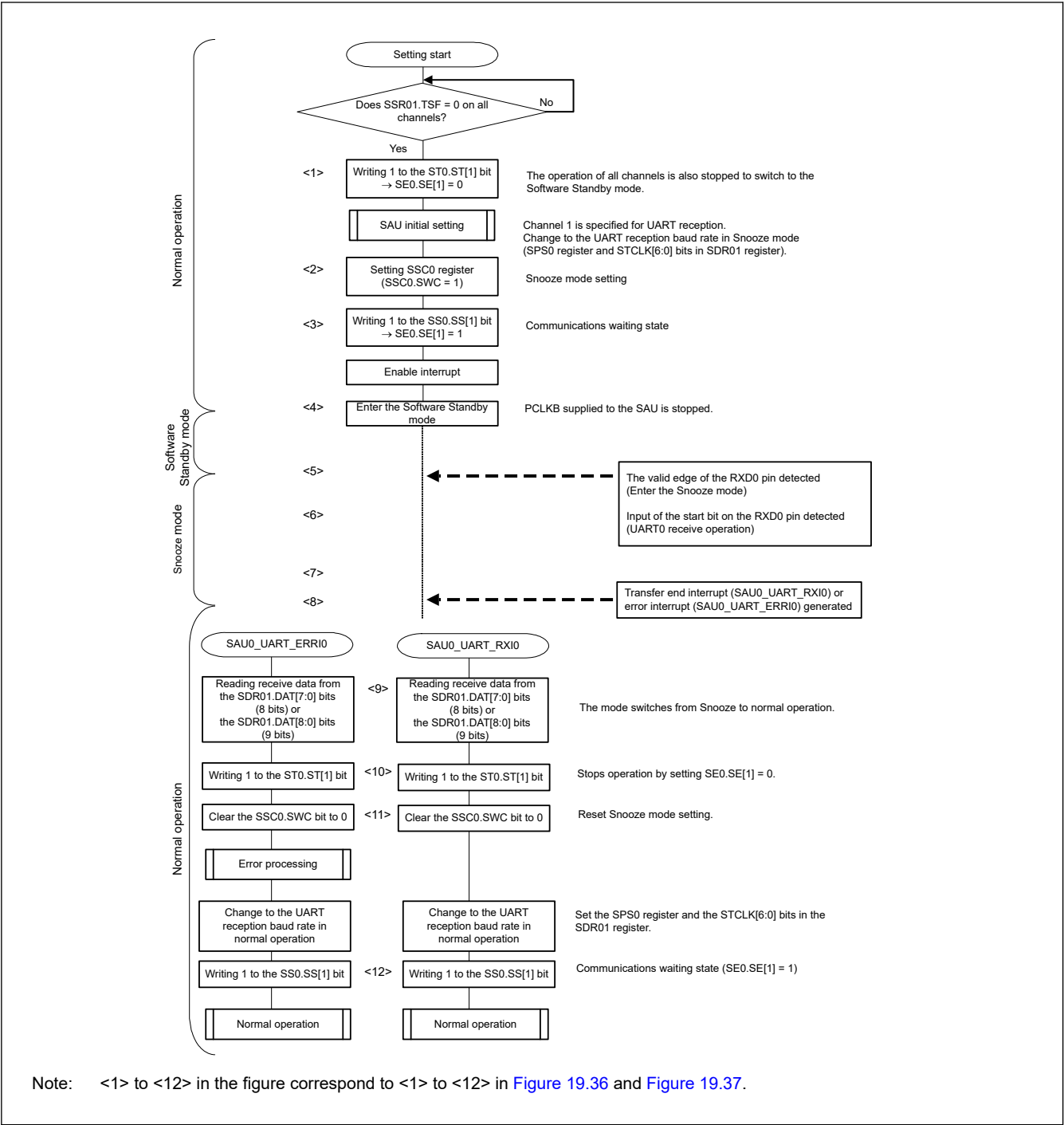


Figure 19.38 Flowchart of Snooze mode operation (SCR01.EOC = 0, SSC0.SSEC = 0/1 or SCR01.EOC = 1, SSC0.SSEC = 0)

(3) Snooze mode operation (SCR01.EOC = 1, SSC0.SSEC = 1: Error interrupt (SAU0_UART_ERRI0) generation is stopped)

Because SCR01.EOC = 1 and SSC0.SSEC = 1, an error interrupt (SAU0_UART_ERRI0) is not generated when a communication error occurs.

Figure 19.39 shows the timing of Snooze mode operation (SCR01.EOC = 1, SSC0.SSEC = 1).

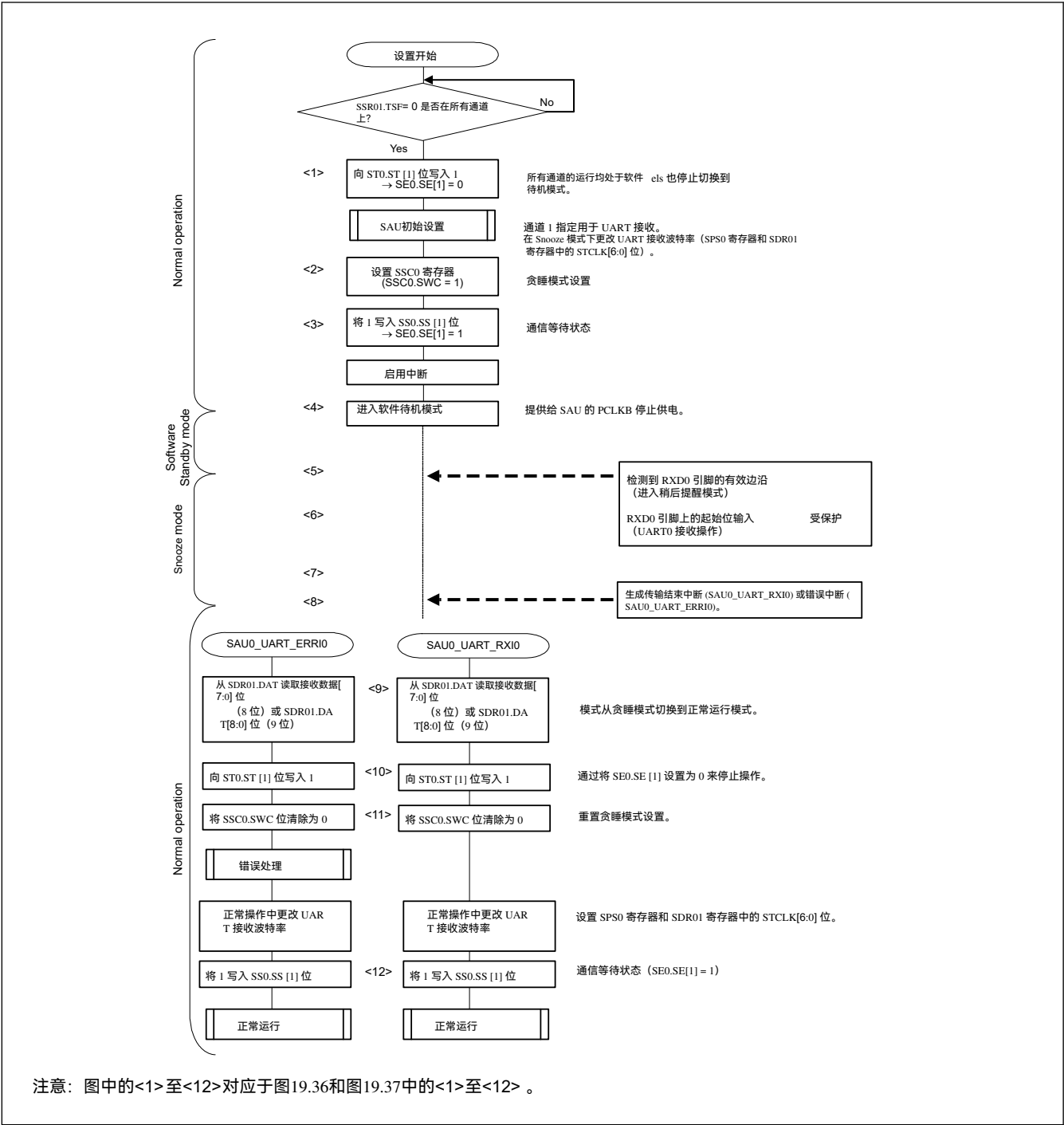


图 19.38 贪睡模式操作流程 (SCR01.EOC= 0, SSC0.SSEC= 0/1 或 SCR01.EOC = 1, SSC0.SSEC = 0)

(3) 睡眠模式操作(SCR01.EOC = 1, SSC0.SSEC= 1: 错误中断 (SAU0_UART_ERRI0) 生成已停止)

因为SCR01.EOC = 1 和SSC0.SSEC = 1, 所以当发生通信错误时, 不会生成错误中断 (SAU0_UART_ERRI0)。

图 19.39 显示了贪睡模式操作的时序(SCR01.EOC = 1, SSC0.SSEC = 1)。

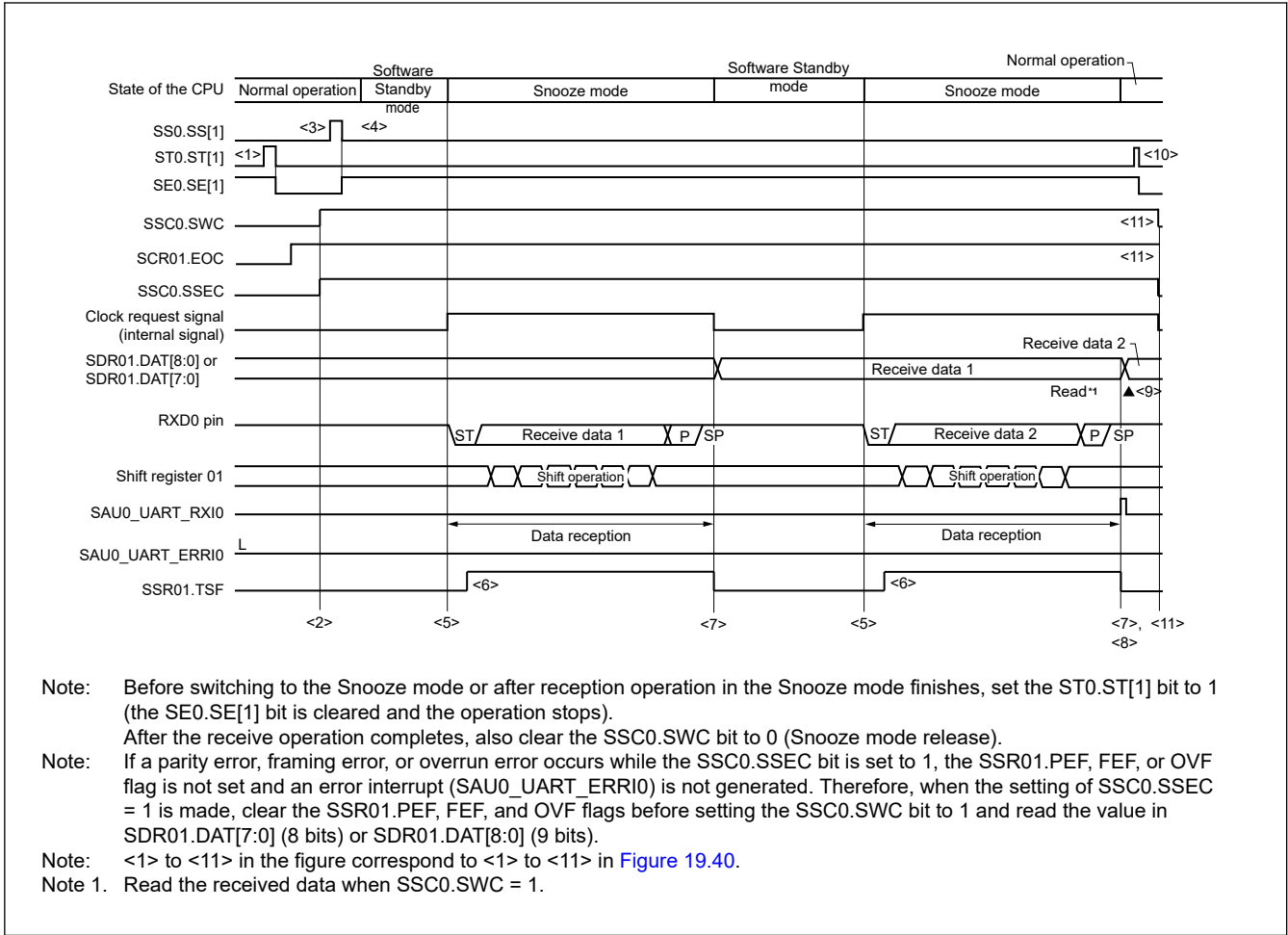


Figure 19.39 Timing of Snooze mode operation (SCR01.EOC = 1, SSC0.SSEC = 1)

Figure 19.40 shows the flowchart of Snooze mode operation (SCR01.EOC = 1, SSC0.SSEC = 1).

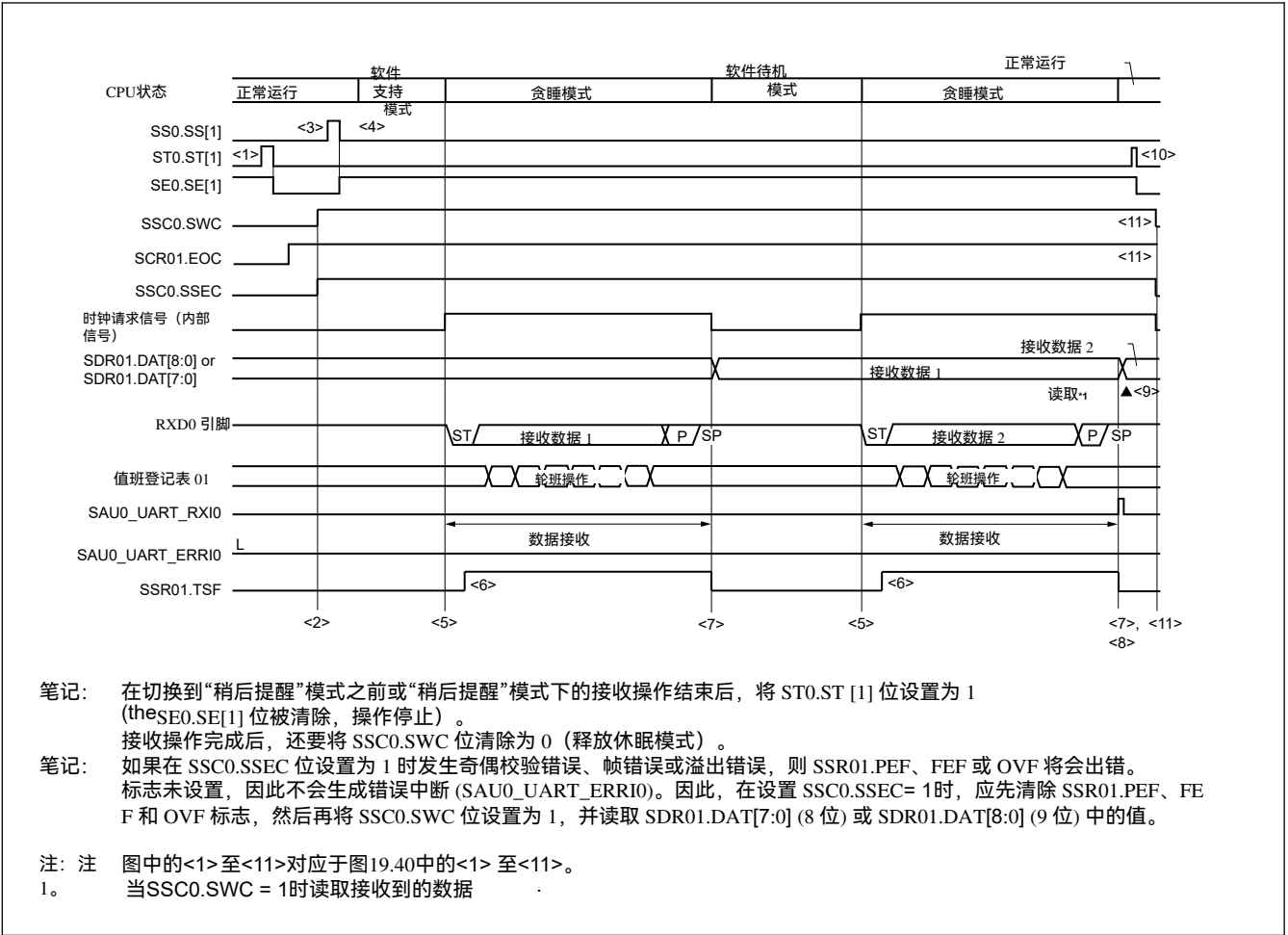


图 19.39 贪睡模式操作的时序 (SCR01.EOC= 1, SSC0.SSEC= 1)

图 19.40 显示了贪睡模式操作的流程图(SCR01.EOC = 1, SSC0.SSEC = 1)。

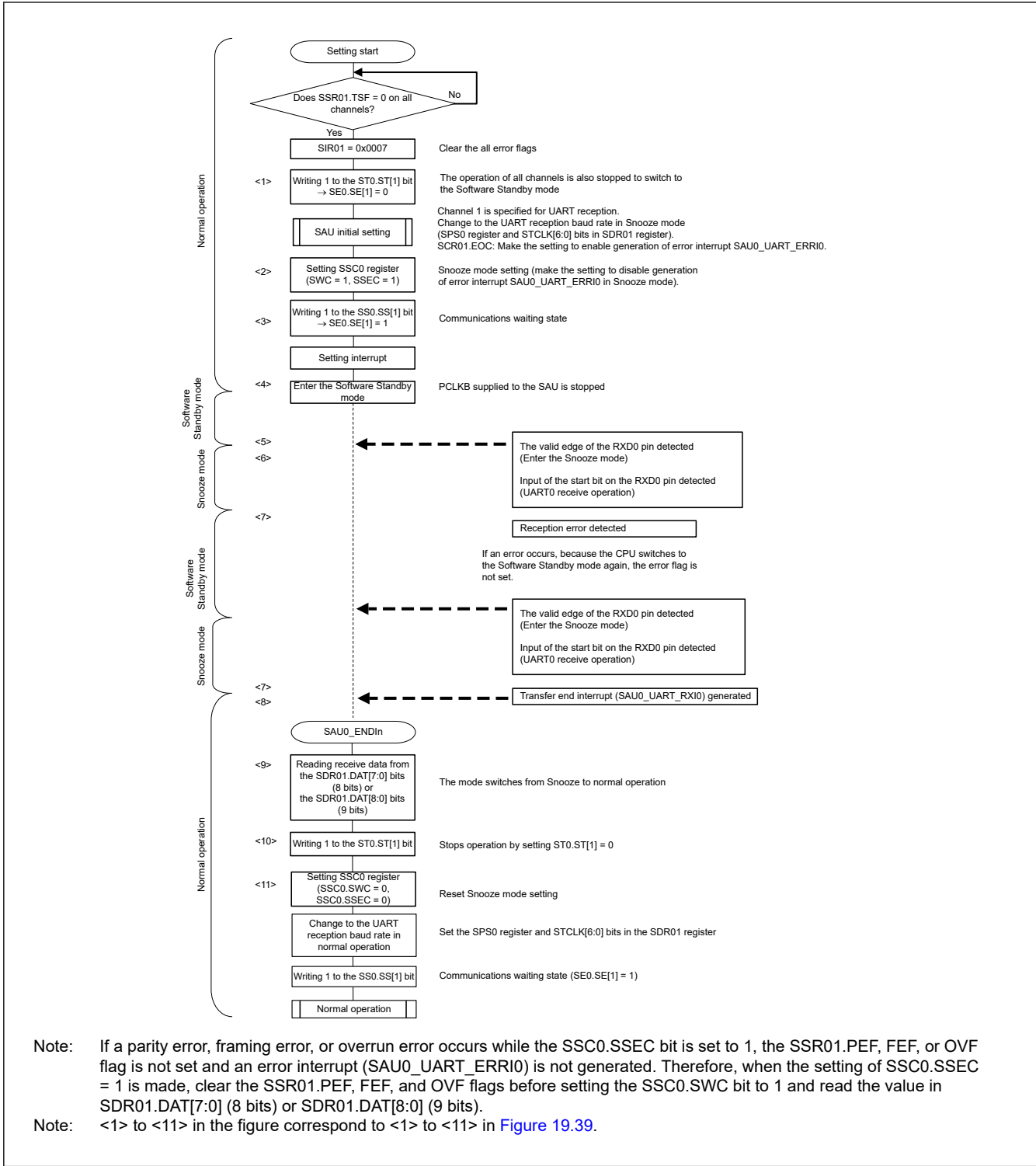


Figure 19.40 Flowchart of Snooze mode operation (SCR01.EOC = 1, SSC0.SSEC = 1)

19.6.4 Calculating Baud Rate

(1) Baud rate calculation expression

The baud rate for UART communication can be calculated by the following expressions.

$$(\text{Baud rate}) = \{\text{Operation clock (f}_{\text{MCK}}\text{) frequency of target channel}\} \div (\text{SDRmn.STCLK}[6:0] + 1) \div 2 \text{ [bps]}$$

Note: Setting SDRmn.STCLK[6:0] = (0x00, 0x01) is prohibited.

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3), mn = 00 to 03

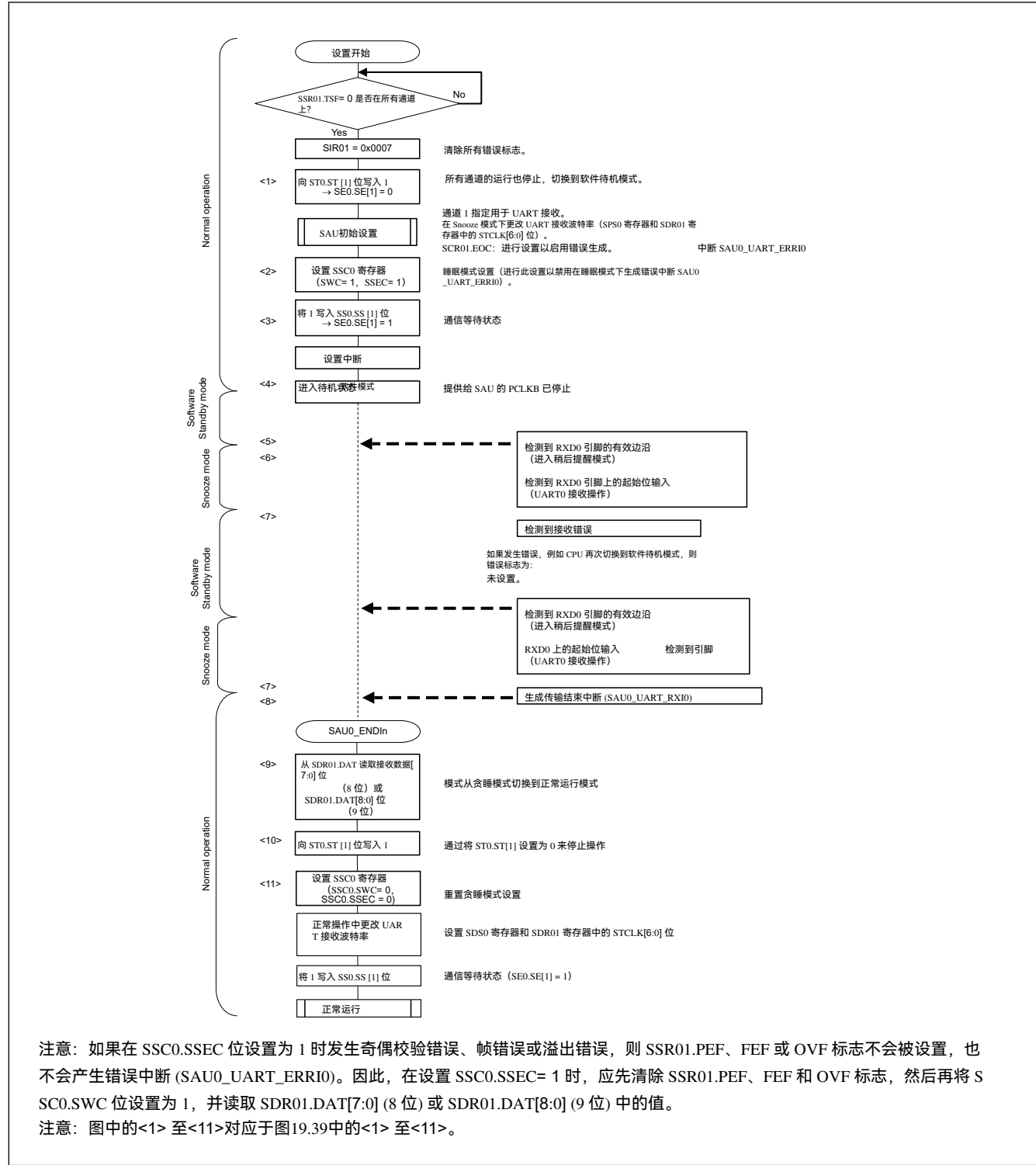


图 19.40 贪睡模式操作流程 (SCR01.EOC = 1, SSC0.SSEC = 1)

19.6.4 计算波特率

(1) 波特率计算表达式

UART 通信的波特率可以通过以下表达式计算。

$$(\text{波特率}) = \{\text{操作时钟 (f}_{\text{MCK}}\text{) 目标信道频率}\} \div (\text{SDRmn.STCLK}[6:0] + 1) \div 2 \text{ [bps]}$$

注意: 禁止设置 SDRmn.STCLK[6:0] = (0x00, 0x01)。

注: m: 单元编号 (m = 0), n: 通道编号 (n = 0 至 3), mn = 00 至 03

The operation clock (f_{MCK}) is determined by serial clock select register m (SPSm) and CKS bit of serial mode register mn (SMRmn). See [Table 19.69](#).

(2) Baud rate error during transmission

The baud rate error of UART communication during transmission can be calculated by the following expression. Make sure that the baud rate at the transmission side is within the permissible baud rate range at the reception side.

(Baud rate error) = (Calculated baud rate value) ÷ (Target baud rate) × 100 - 100 [%]

[Table 19.94](#) shows an example of setting a UART baud rate at PCLKB = 32 MHz.

Table 19.94 Example of setting UART baud rate at PCLKB = 32 MHz

UART baud rate (target baud rate)	PCLKB = 32 MHz			
	Operation clock (f _{MCK})	SDRmn.STCLK [6:0]	Calculated baud rate	Error from target baud rate
300 bps	PCLKB/2 ⁹	103	300.48 bps	+0.16%
600 bps	PCLKB/2 ⁸	103	600.96 bps	+0.16%
1200 bps	PCLKB/2 ⁷	103	1201.92 bps	+0.16%
2400 bps	PCLKB/2 ⁶	103	2403.85 bps	+0.16%
4800 bps	PCLKB/2 ⁵	103	4807.69 bps	+0.16%
9600 bps	PCLKB/2 ⁴	103	9615.38 bps	+0.16%
19200 bps	PCLKB/2 ³	103	19230.8 bps	+0.16%
31250 bps	PCLKB/2 ³	63	31250.0 bps	±0.0%
38400 bps	PCLKB/2 ²	103	38461.5 bps	+0.16%
76800 bps	PCLKB/2	103	76923.1 bps	+0.16%
153600 bps	PCLKB	103	153846 bps	+0.16%
312500 bps	PCLKB	50	313725.5 bps	+0.39%

Note: m: Unit number (m = 0), n: Channel number (n = 0, 2), mn = 00, 02

(3) Permissible baud rate range for reception

The permissible baud rate range for reception during UART communication can be calculated by the following expression. Make sure that the baud rate at the transmission side is within the permissible baud rate range at the reception side.

(Maximum receivable baud rate) = $\frac{2 \times k \times \text{Nfr}}{2 \times k \times \text{Nfr} - k + 2} \times \text{Brate}$

(Minimum receivable baud rate) = $\frac{2 \times k \times (\text{Nfr} - 1)}{2 \times k \times \text{Nfr} - k - 2} \times \text{Brate}$

- Brate: Calculated baud rate value at the reception side (See [\(1\) Baud rate calculation expression.](#))
- k: SDRmn.STCLK[6:0] + 1
- Nfr: 1 data frame length [bits] = (Start bit) + (Data length) + (Parity bit) + (Stop bit)

Note: m: Unit number (m = 0), n: Channel number (n = 1, 3), mn = 01, 03

[Figure 19.41](#) shows the permissible baud rate range for reception (1 Data Frame Length = 11 Bits).

操作时钟(f_{MCK}) 由串行时钟选择寄存器m (SPSm) 和串行模式寄存器mn (SMRmn) 的CKS位确定。参见表19.6 9。

(2) 传输过程中的波特率错误

UART通信传输过程中的波特率误差可以通过以下公式计算。请确保发送端的波特率在接收端允许的波特率范围内。

(波特率误差) = (计算出的波特率值) ÷ (目标波特率) × 100 - 100 [%]

表 19.94 显示了在PCLKB = 32 MHz设置 UART 波特率的示例

表 19.94 设置 UART 波特率的示例PCLKB = 32 MHz

UART波特率（目标波特率）	PCLKB = 32 MHz			
	运行时钟（f _{MCK} ）	SDRmn.STCLK [6:0]	计算出的波特率	目标波特率误差
300 bps	PCLKB/2 ⁹	103	300.48 bps	+0.16%
600 bps	PCLKB/2 ⁸	103	600.96 bps	+0.16%
1200 bps	PCLKB/2 ⁷	103	1201.92 bps	+0.16%
2400 bps	PCLKB/2 ⁶	103	2403.85 bps	+0.16%
4800 bps	PCLKB/2 ⁵	103	4807.69 bps	+0.16%
9600 bps	PCLKB/2 ⁴	103	9615.38 bps	+0.16%
19200 bps	PCLKB/2 ³	103	19230.8 bps	+0.16%
31250 bps	PCLKB/2 ³	63	31250.0 bps	±0.0%
38400 bps	PCLKB/2 ²	103	38461.5 bps	+0.16%
76800 bps	PCLKB/2	103	76923.1 bps	+0.16%
153600 bps	PCLKB	103	153846 bps	+0.16%
312500 bps	PCLKB	50	313725.5 bps	+0.39%

注: m: 单元编号(m = 0), n: 通道编号(n = 0, 2), mn = 00, 02

(3) 接收允许的波特率范围

UART 通信期间接收允许的波特率范围可以通过以下表达式计算。确保发送端的波特率在接收端允许的波特率范围内。

(最大可接收波特率rate) = $\frac{2 \times k \times \text{Nfr}}{2 \times k \times \text{Nfr} - k + 2} \times \text{Brate}$

(最低应收账款波特率) = $\frac{2 \times k \times \text{Nfr} - 1}{2 \times k \times \text{Nfr} - k - 2} \times \text{Brate}$

- 波特率: 接收端计算出的波特率值（参见（1）波特率计算表达式。）
- k: SDRmn.STCLK[6:0] + 1
- Nfr: 1 个数据帧长度 [bits] = (Start bit) + (Data length) + (Parity bit) + (Stop bit)

注: m: 单元号(m = 0), n: 通道号(n = 1, 3), mn = 01, 03

图 19.41 显示了接收的允许波特率范围（1 数据帧长度= 11 比特）。

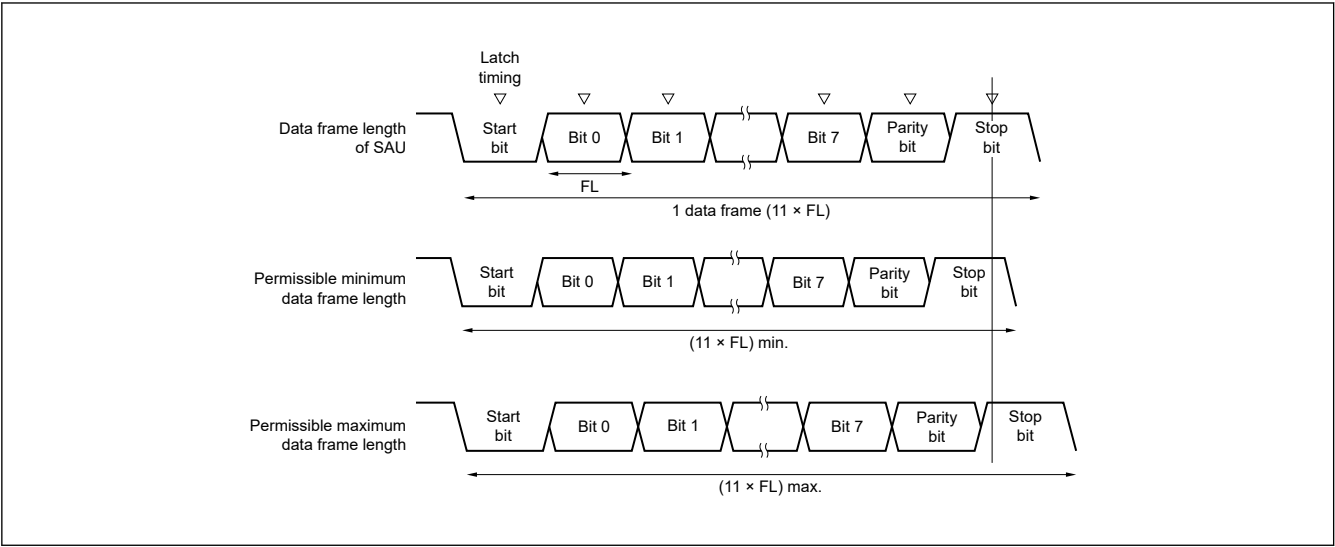


Figure 19.41 Permissible baud rate range for reception (1 data frame length = 11 bits)

As shown in Figure 19.41, the timing of latching receive data is determined by the division ratio set by STCLK[6:0] bits of serial data register mn (SDRmn) after the start bit is detected. If the last data (stop bit) is received before this latch timing, the data can be correctly received.

19.6.5 Procedure for Processing Errors that Occurred During UART Communication

The procedure for processing errors that occurred during UART communication is described in Table 19.95 and Table 19.96.

Table 19.95 Processing procedure for parity error or overrun error

Step	Software manipulation		State of the hardware	Note
<1>	Reads serial data register mn (SDRmn).	→	The BFF bit of the SSRmn register is set to 0 and channel n is enabled to receive data	This is to prevent an overrun error if the next reception is completed during error processing.
<2>	Reads serial status register mn (SSRmn).		—	The error type is identified and the read value is used to clear the error flag.
<3>	Writes 1 to serial flag clear trigger register mn (SIRmn).		The error flag is cleared.	Only the error generated during reading can be cleared, by writing the value read from the SSRmn register to the SIRmn register without modification.
		→		

Table 19.96 Processing procedure for framing error (1 of 2)

Step	Software manipulation		State of the hardware	Note
<1>	Reads serial data register mn (SDRmn).	→	The BFF bit of the SSRmn register is set to 0 and channel n is enabled to receive data	This is to prevent an overrun error if the next reception is completed during error processing.
<2>	Reads serial status register mn (SSRmn).		—	The error type is identified and the read value is used to clear the error flag.
<3>	Writes serial flag clear trigger register mn (SIRmn).		The error flag is cleared.	Only the error generated during reading can be cleared, by writing the value read from the SSRmn register to the SIRmn register without modification.
<4>	Sets the ST[n] bit of serial channel stop register m (ST[n]) to 1.	→	The SE[n] bit of serial channel enable status register m (SEm) is set to 0 and channel n stops operation.	—

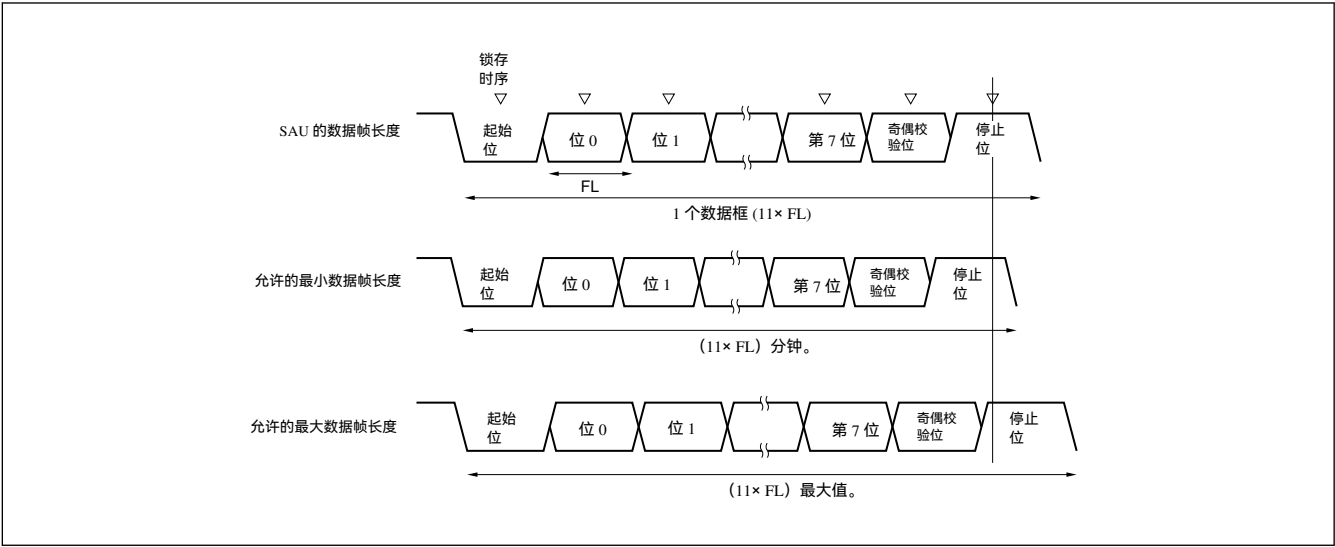


图 19.41 接收允许的波特率范围（1 个数据帧长度= 11 位）

如图 19.41 所示，接收数据锁存的时序由串行数据寄存器 mn (SDRmn) 在检测到起始位后，STCLK[6:0] 位所设定的分频比决定。如果在锁存时序之前接收到最后一个数据（停止位），则可以正确接收数据。

19.6.5 UART 通信过程中发生的错误的处理程序

表 19.95 和表 19.96 描述了 UART 通信过程中发生的错误的处理程序。

表 19.95 奇偶校验错误或超限错误的处理程序

步	软件操作		硬件状态	Note
<1>	读取串行数据寄存器 mn (SDRmn)。	→	SSRmn寄存器的BFF位设置为0，通道 n被启用以接收数据。	这是为了防止在错误处理期间完成下一次接收时发生溢出错误。
<2>	读取串行状态寄存器 mn (SSRmn)。		—	识别错误类型，并使用读取的值清除错误标志。
<3>	向串行标志清除触发寄存器 mn (SIRmn) 写入 1。		错误标志已清除。	只能通过将从 SSRmn 寄存器读取的值原封不动地写入 SIRmn 寄存器来清除读取过程中产生的错误。
		→		

表 19.96 帧错误处理程序（1/2）

步	软件操作		硬件状态	Note
<1>	读取串行数据寄存器 mn (SDRmn)。	→	SSRmn寄存器的BFF位设置为0，通道 n被启用以接收数据。	这是为了防止在错误处理期间完成下一次接收时发生溢出错误。
<2>	读取串行状态寄存器 mn (SSRmn)。		—	识别错误类型，并使用读取的值清除错误标志。
<3>	写入串行标志清除触发寄存器 mn (SIRmn)。		错误标志已清除。	只能通过将从 SSRmn 寄存器读取的值原封不动地写入 SIRmn 寄存器来清除读取过程中产生的错误。
<4>	将串行通道停止寄存器 m (ST[n]) 的 ST[n] 位设置为 1。	→	串行通道使能状态寄存器 m (SEm) 的 SE[n] 位设置为 0，通道 n 停止操作。	—

Table 19.96 Processing procedure for framing error (2 of 2)

Step	Software manipulation		State of the hardware	Note
<5>	Synchronization with other party of communication	→	—	Synchronization with the other party of communication is re-established and communication is resumed because it is considered that a framing error has occurred because the start bit has been shifted.
<6>	Sets the SSm.SS[n] bit of serial channel start register m (SSm) to 1.		The SE[n] bit of serial channel enable status register m (SEm) is set to 1 and channel n is enabled to operate.	—

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3), mn = 00 to 03

19.7 Operation of LIN Communication

19.7.1 LIN Transmission

UART0 supports LIN communication.

Channel 0 of unit 0 is used for LIN transmission.

Table 19.97 shows the specification of LIN transmission.

Table 19.97 Specification of LIN transmission

UART	UART0	UART1
Support of LIN communication	Supported	Not supported
Target channel	Channel 0 of SAU0	—
Pins used	TXD0	—
Interrupt	SAU0_UART_TXI0	—
	Transfer end interrupt (in single-transfer mode) or buffer empty interrupt (in continuous transfer mode) can be selected.	
Error detection flag	None	
Transfer data length	8 bits	
Transfer rate*1	Max. fMCK/6 [bps] (SDR00.STCLK[6:0] = 2 or more), Min. PCLKB/(2 × 2 ¹⁵ × 128) [bps]	
Data phase	Non-reverse output (default: high level) Reverse output (default: low level)	
Parity bit	No parity bit	
Stop bit	Appending 1 bit	
Data direction	LSB first	

Note: fMCK: Operation clock frequency of target channel

Note 1. Use this operation within a range that satisfies the conditions above and the peripheral functions characteristics specified in the electrical characteristics. For details, see [section 27, Electrical Characteristics](#). In general, 2.4, 9.6, or 19.2 kbps is often used in LIN communication.

LIN stands for Local Interconnect Network and is a low-speed (1 to 20 kbps) serial communication protocol designed to reduce the cost of an automobile network.

Communication of LIN is single-master communication and up to 15 slaves can be connected to one master. The slaves are used to control switches, actuators, and sensors, which are connected to the master via LIN. Usually, the master is connected to a network such as CAN (Controller Area Network).

A LIN bus is a single-wire bus to which nodes are connected via transceiver conforming to ISO9141.

According to the protocol of LIN, the master transmits a frame by attaching baud rate information to it. A slave receives this frame and corrects a baud rate error from the master. If the baud rate error of a slave is within ±15%, communication can be established.

表 19.96 帧误差处理程序 (2/2)

步	软件操作		硬件状态	Note
<5>	与通信另一方同步	→	—	由于起始位发生移位，导致帧错误，因此与通信另一方的同步得以重新建立，通信得以恢复。
<6>	将串行通道起始寄存器 m (SSm) 的 SS m.SS[n] 位设置为 1。		串行通道使能状态寄存器m(SEm) 的 SE[n] 位设置为 1，通道 n 被启用以进行操作。	—

注: m: 单元编号(m = 0), n: 通道编号(n = 0至3) , mn = 00至03

19.7 LIN通信的操作

19.7.1LIN 变速器

UART0 支持 LIN 通信。

0 号单元的 0 号通道用于 LIN 传输。

表 19.97 显示了 LIN 传输的规格。

表 19.97 LIN 传输规范

UART	UART0	UART1
支持LIN通信	支持	不支持
目标频道	SAU0 的 0 通道	—
使用的引脚	TXD0	—
打断	SAU0_UART_TXI0	—
	可以选择传输结束中断（单次传输模式）或缓冲区空中断（连续传输模式）。	
错误检测标志	None	
传输数据长度	8 位	
转账费率*1	最大限度。fMCK/6[bps]（SDR00.STCLK[6: 0] = 2 或更多），最小PCLKB/(2 × 2 ¹⁵ ×128)[比特/秒]	
数据阶段	非反向输出（默认值: 高电平） 反向输出（默认值: 低电平）	
奇偶校验位	无奇偶校验位	
停止位	追加 1 位	
数据方向	LSB 第一	

注: fMCK: 目标通道的工作时钟频率

注1: 此操作应在满足上述条件和电气特性中规定的外围功能特性的范围内使用。详情请参见第27节“电气特性”。通常，LIN通信中常用的速率为2.4、9.6或19.2 kbps。

LIN 代表本地互连网络，是一种低速（1 到20 kbps）串行通信协议，旨在降低汽车网络的成本。

LIN通信采用单主站通信模式，一个主站最多可连接15个从站。从站用于控制开关、执行器和传感器，这些设备通过LIN线路连接到主站。通常，主站会连接到CAN（控制器局域网）等网络。

LIN 总线是一种单线总线，节点通过符合 ISO9141 标准的收发器连接到该总线上。

根据LIN协议，主设备发送一个包含波特率信息的数据帧。从设备接收到该帧后，会纠正主设备发送的波特率错误。

如果从设备的波特率错误在±15%范围内，则可以建立通信。

Figure 19.42 outlines a transmission operation of LIN.

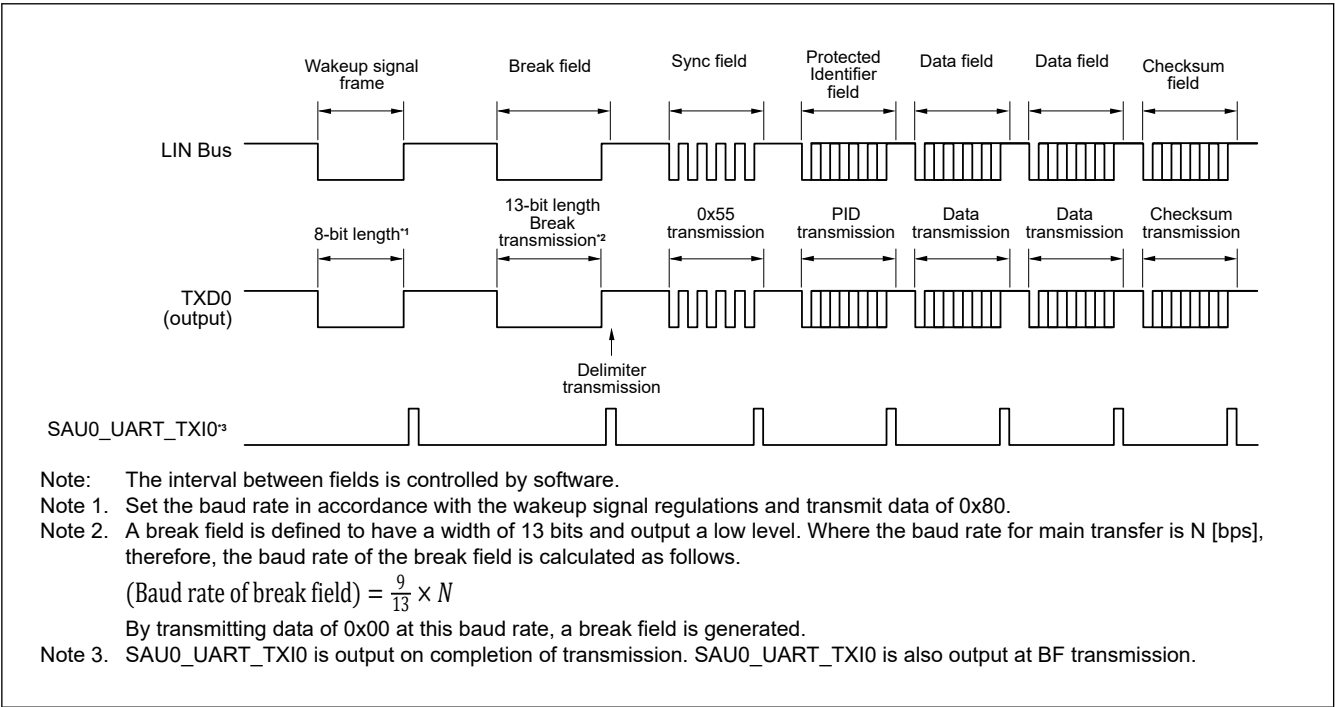


Figure 19.42 Transmission operation of LIN

Figure 19.43 shows the flowchart for LIN transmission.

图 19.42 概述了 LIN 的传输操作。

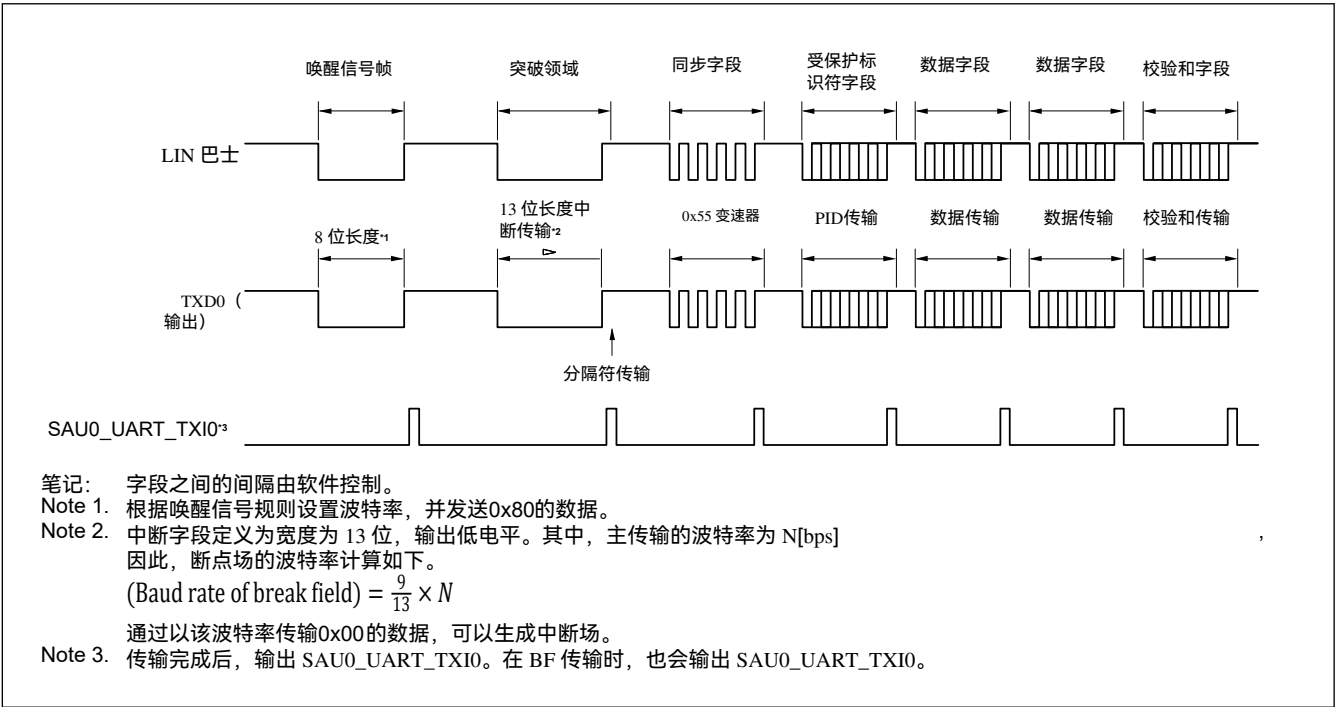


图 19.42 LIN 的传输操作

图 19.43 显示了 LIN 传输的流程图。

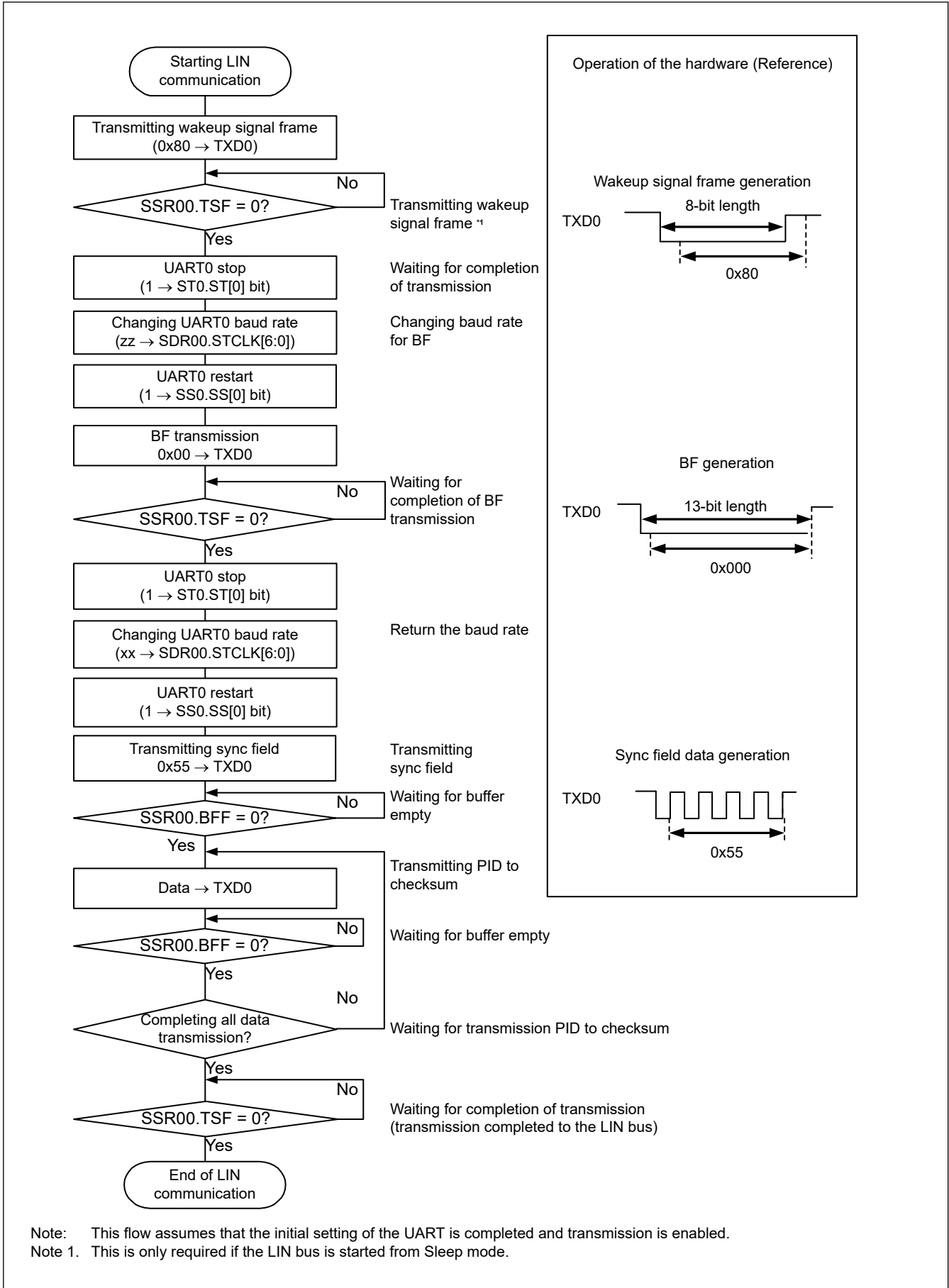


Figure 19.43 Flowchart for LIN transmission

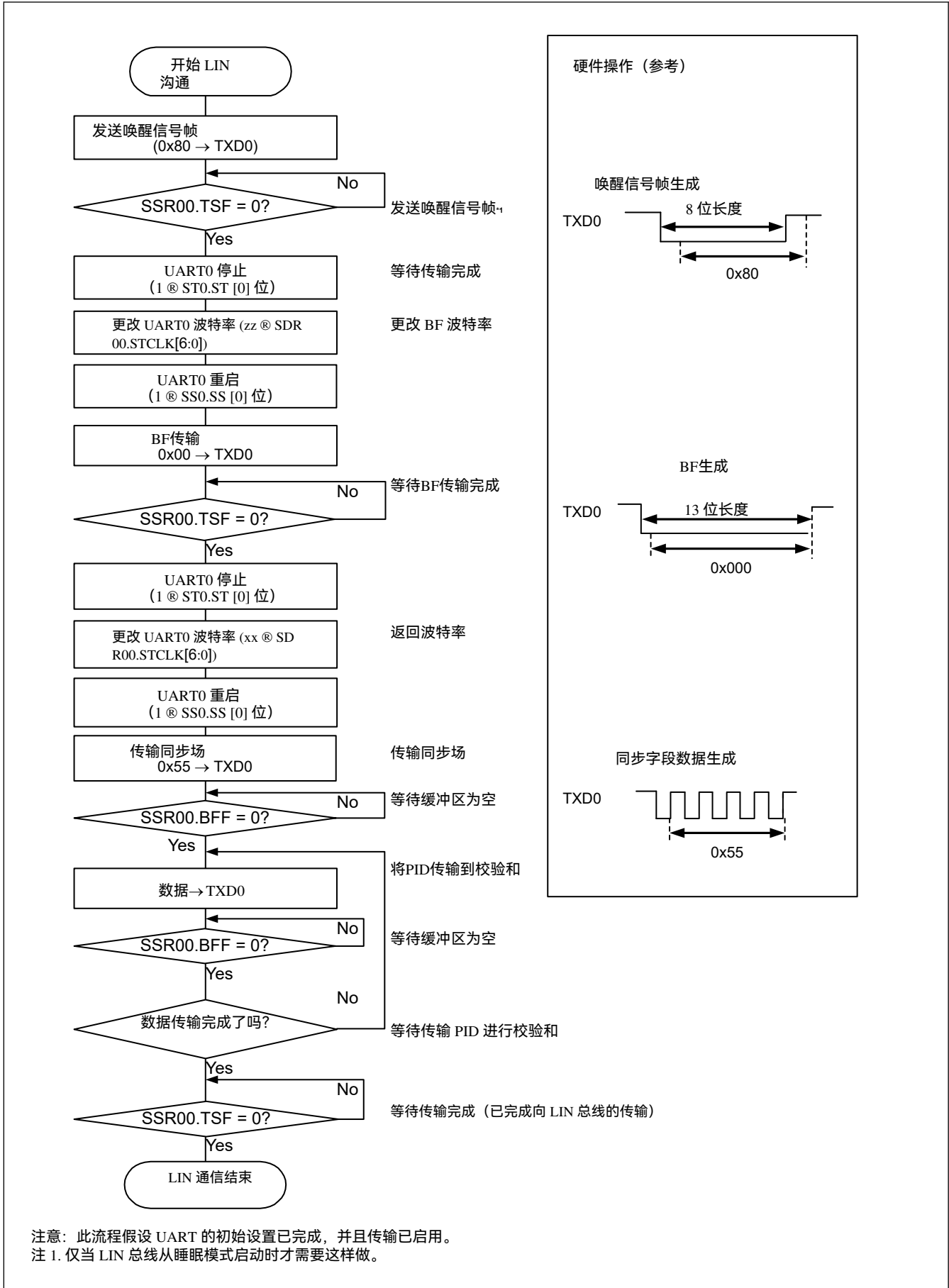


图 19.43 LIN 传输流程图

19.7.2 LIN Reception

Of UART reception, UART0 supports LIN communication.

For LIN reception, channel 1 of unit 0 is used.

Table 19.98 shows the specification of LIN reception.

Table 19.98 Specification of LIN reception

UART	UART0	UART1
Support of LIN communication	Supported	Not supported
Target channel	Channel 1 of SAU0	—
Pins used	RXD0	—
Interrupt	SAU0_UART_RXI0	—
	Transfer end interrupt only (setting the buffer empty interrupt is prohibited)	
Error interrupt	SAU0_UART_ERRI0	—
Error detection flag	- Framing Error detection flag (SSR01.FEF) - Overrun Error detection flag (SSR01.OVF)	
Transfer data length	8 bits	
Transfer rate*1	Max. f _{MCK} /6 [bps] (SDR01.STCLK[6:0] = 2 or more), Min. PCLKB/(2 × 2 ¹⁵ × 128) [bps]	
Data phase	Non-reverse output (default: high level) Reverse output (default: low level)	
Parity bit	No parity bit (the parity bit is not checked)	
Stop bit	Check the first bit	
Data direction	LSB first	

Note: f_{MCK}: Operation clock frequency of target channel

Note 1. Use this operation within a range that satisfies the conditions above and the peripheral functions characteristics specified in the electrical characteristics. For details, see [section 27, Electrical Characteristics](#) .

Figure 19.44 shows a reception operation of LIN.

19.7.2LIN 接待处

UART接收方面，UART0支持LIN通信。

对于 LIN 接收，使用单元 0 的通道 1。

表 19.98 显示了 LIN 接收的规格。

表 19.98 LIN 接收规格

UART	UART0	UART1
支持LIN通信	支持	不支持
目标频道	SAU0 频道 1	—
使用的引脚	RXD0	—
打断	SAU0_UART_RXI0	—
	仅传输结束中断（设置 b）允许空中断是被禁止的）	
错误中断	SAU0_UART_ERRI0	—
错误检测标志	- 帧错误检测标志（SSR01.FEF） - 溢出错误检测标志（SSR01.OVF）	
传输数据长度	8 位	
转账费率*1	最大限度。f _{MCK} /6[bps]（SDR01.STCLK[6： 0] = 2 或更多），最小\pCLKB/(2 × 2 ¹⁵ ×128)[比特/秒]	
数据阶段	非反向输出（默认值： 高电平） 反向输出（默认值： 低电平）	
奇偶校验位	无奇偶校验位（不检查奇偶校验位）	
停止位	检查第一部分	
数据方向	LSB 第一	

注：f_{MCK}：目标通道的工作时钟频率

注1： 此操作应在满足上述条件和电气特性中规定的外围功能特性的范围内使用。详情请参见第27节“电气特性”。

图 19.44 显示了 LIN 的接收操作。

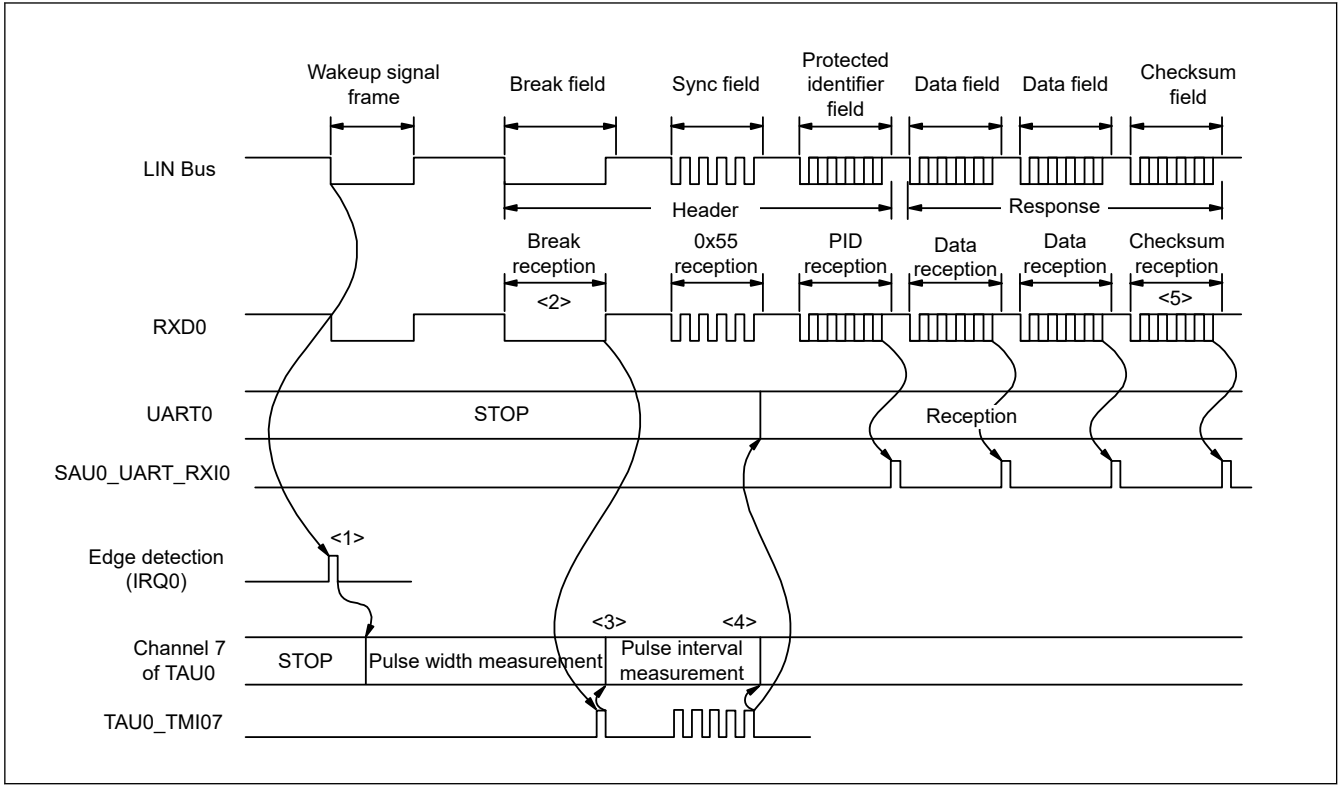


Figure 19.44 Reception operation of LIN

The flow of reception processing is as follows.

1. The wakeup signal is detected by using an edge on the external interrupt pin (IRQ0). When the wakeup signal is detected, set channel 7 of TAU0 to the pulse width measurement function to measure the low-level width of the BF signal. Then wait for BF signal reception.
2. Channel 7 of TAU0 starts measuring the low-level width on detection of the falling edge of the BF signal, and then captures the data on detection of the rising edge of the BF signal. The captured data is used to determine whether it is the BF signal.
3. When the BF signal has been received normally, change channel 7 of TAU0 to pulse interval measurement and measure the interval between the falling edges of the RXD0 signal in the Sync field four times. (See [section 16.7.4. Operation for Input Pulse Interval Measurement](#)).
4. Calculate a baud rate error from the bit interval of sync field (SF). Stop UART0 once and adjust (reset) the baud rate.
5. The checksum field should be distinguished by software. In addition, processing to initialize UART0 after the checksum field is received and to wait for reception of BF should also be performed by software.

Figure 19.45 shows the flowchart of LIN reception.

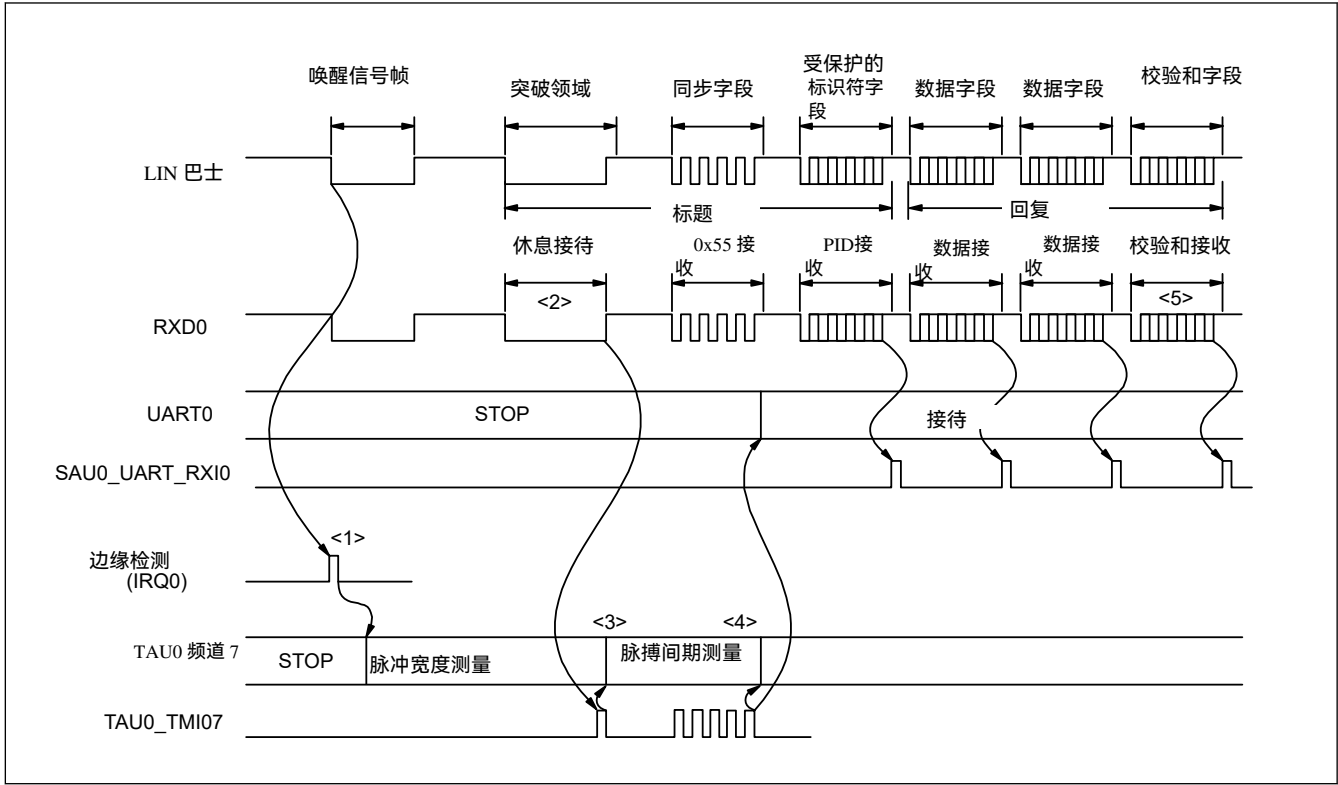


图 19.44 LIN 的接收操作

接收处理流程如下。

1. 通过外部中断引脚（IRQ0）上的边沿检测唤醒信号。检测到唤醒信号后，将TAU0的通道7设置为脉冲宽度测量功能，以测量BF信号的低电平宽度。然后等待接收到BF信号。
2. TAU0 的通道 7 在检测到 BF 信号下降沿时开始测量低电平宽度，然后在检测到 BF 信号上升沿时采集数据。采集到的数据用于判断是否为 BF 信号。
3. 当正常接收到 BF 信号后，将 TAU0 的通道 7 切换至脉冲间隔测量模式，并测量同步场中 RXD0 信号下降沿之间的间隔四次。（参见 16.7.4 节“输入脉冲间隔测量操作”）。
4. 根据同步字段 (SF) 的位间隔计算波特率误差。停止 UART0 一次，并调整（重置）波特率。
5. 校验和字段应由软件识别。此外，接收到校验和字段后初始化UART0以及等待接收BF的处理也应由软件完成。

图 19.45 显示了 LIN 接收的流程图。

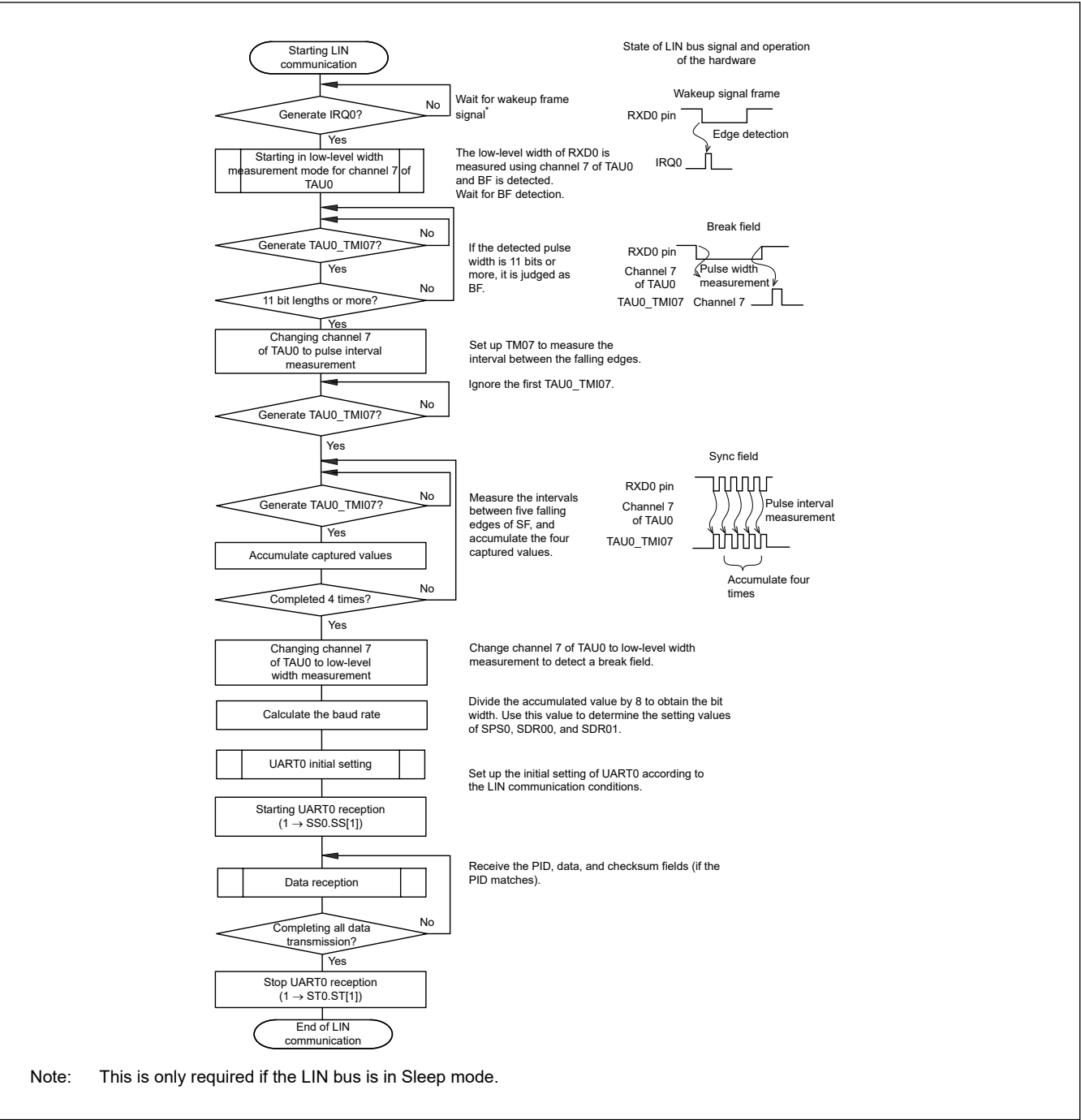


Figure 19.45 Flowchart of LIN reception

Figure 19.46 shows the configuration of ports used for LIN reception.

The wakeup signal transmitted from the master of LIN is received by detecting an edge of an external interrupt (IRQ0). The length of the sync field transmitted from the master can be measured by using the external event capture operation of the timer array unit 0 to calculate a baud-rate error.

By using the port input switching control (the ISC.ISC0 and ISC.ISC1 bits), the signal input to the reception port (RXD0) can be used as an external interrupt (IRQ0) or sent to the timer array unit without additional external connections.

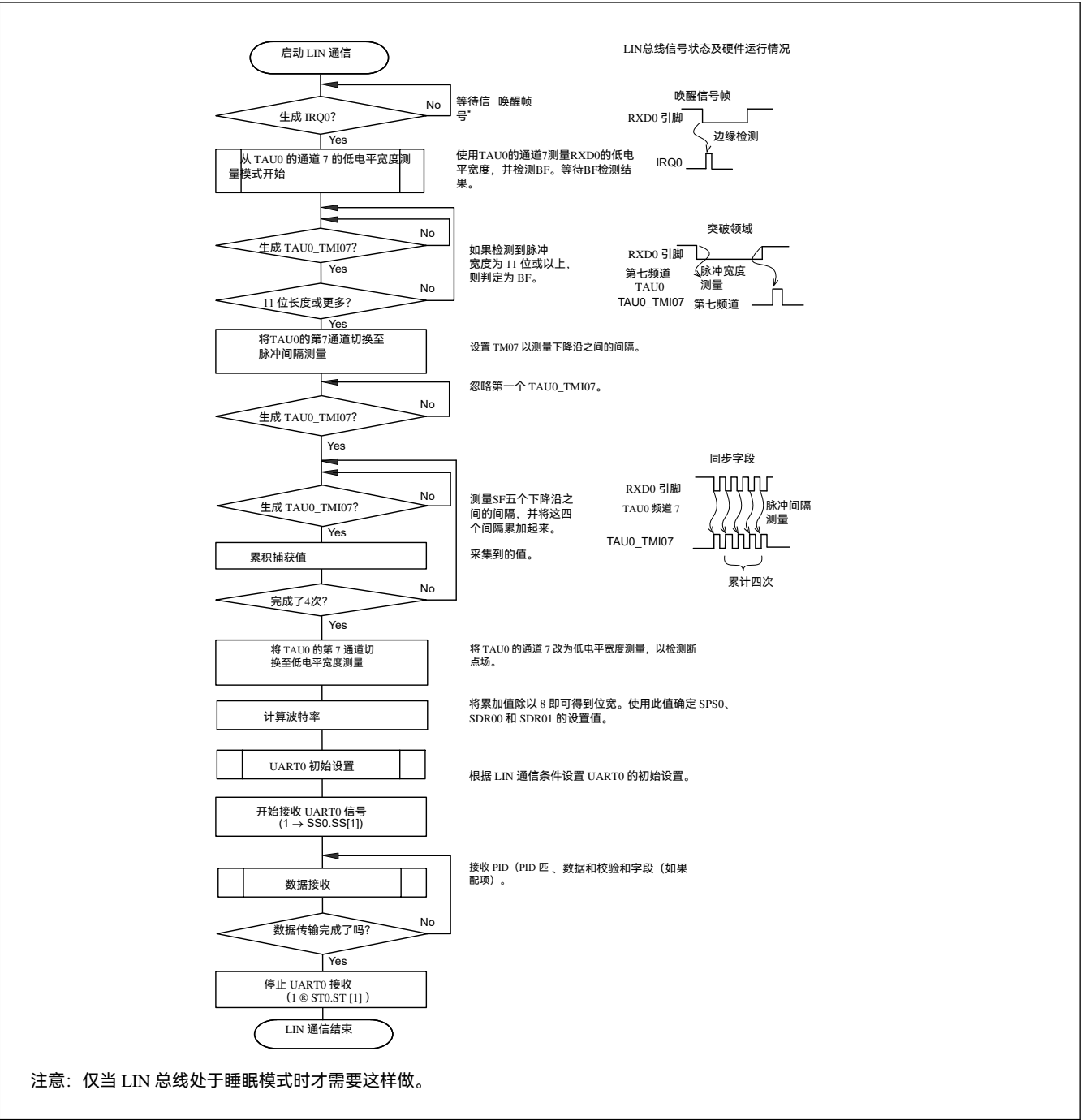


图 19.45 LIN 接收流程图

图 19.46 显示了用于 LIN 接收的端口配置。

通过检测外部中断 (IRQ0) 的边沿来接收来自 LIN 主控端的唤醒信号。利用定时器阵列单元 0 的外部事件捕获操作来计算波特率误差，从而测量来自主控端的同步字段的长度。

通过使用端口输入切换控制 (ISC.ISC0 和 ISC.ISC1 位)，输入到接收端口 (RXD0) 的信号可以用作外部中断 (IRQ0) 或发送到定时器阵列单元，而无需额外的外部连接。

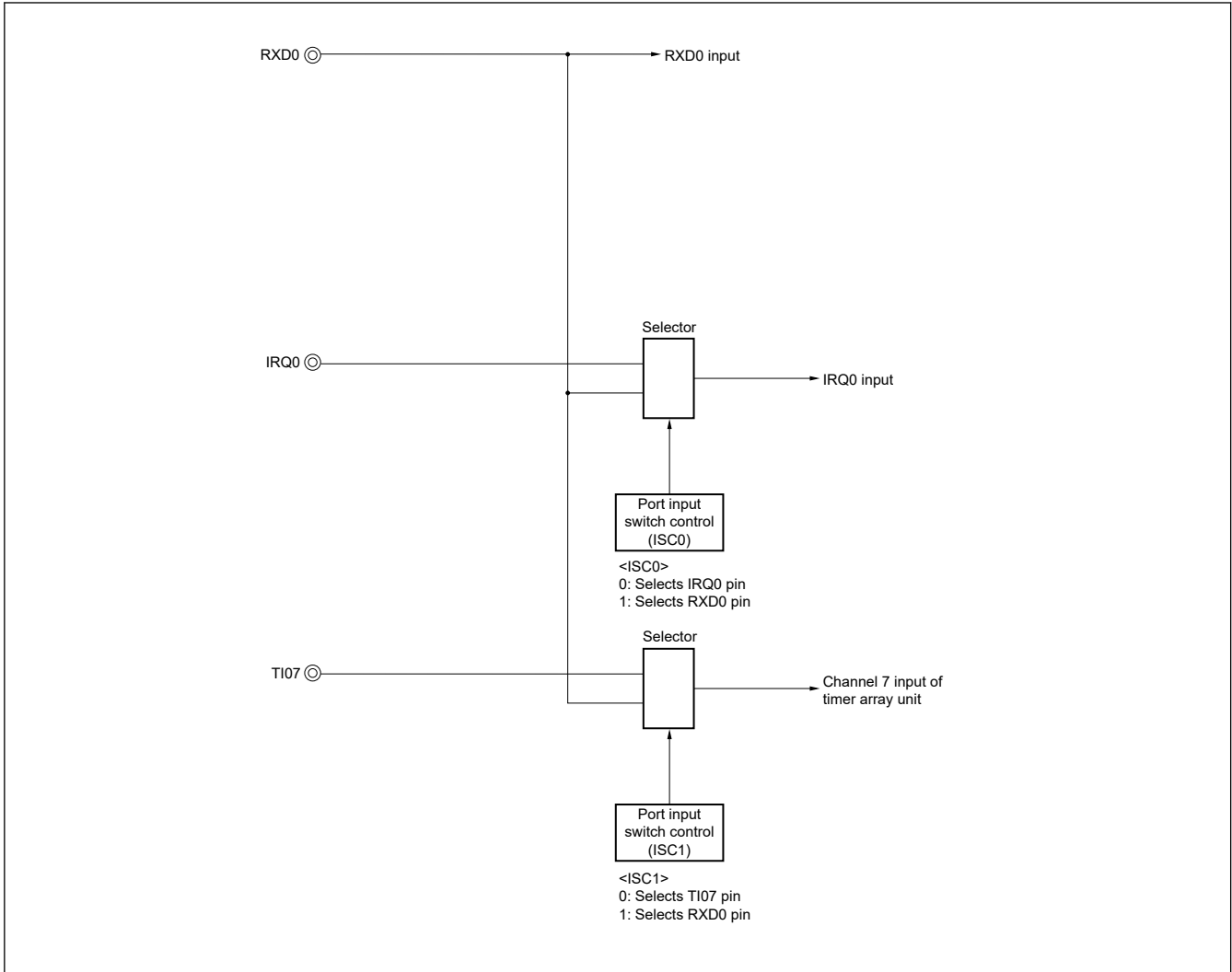


Figure 19.46 Port configuration for LIN reception

The peripheral functions used for the LIN communication operation are as follows.

<Peripheral functions used>

- External interrupt (IRQ0), wakeup signal detection.
Usage: To detect an edge of the wakeup signal and the start of communication.
- Channel 7 of timer array unit, baud rate error detection, break field detection.
Usage: To detect the length of the sync field (SF) and divide it by the number of bits in order to detect a baud rate error.
The interval of the edge input to RXD0 is measured in the capture mode.
To measure the low-level width to detect the break field (BF).
- Channels 0 and 1 (UART0) of serial array unit 0 (SAU0).

19.8 Operation of Simplified I²C Communication

This is a clocked communication function to communicate with two or more devices by using two lines: serial clock (SCL) and serial data (SDA). This simplified I²C is designed for single communication with a device such as EEPROM, flash memory, or A/D converter, and therefore, it functions only as a master.

Operate the control registers by software to set the start and stop conditions while observing the specifications of the I²C bus line.

[Data transmission and reception]

- Master transmission, master reception (only master function with a single master)
- ACK output function*1 and ACK detection function

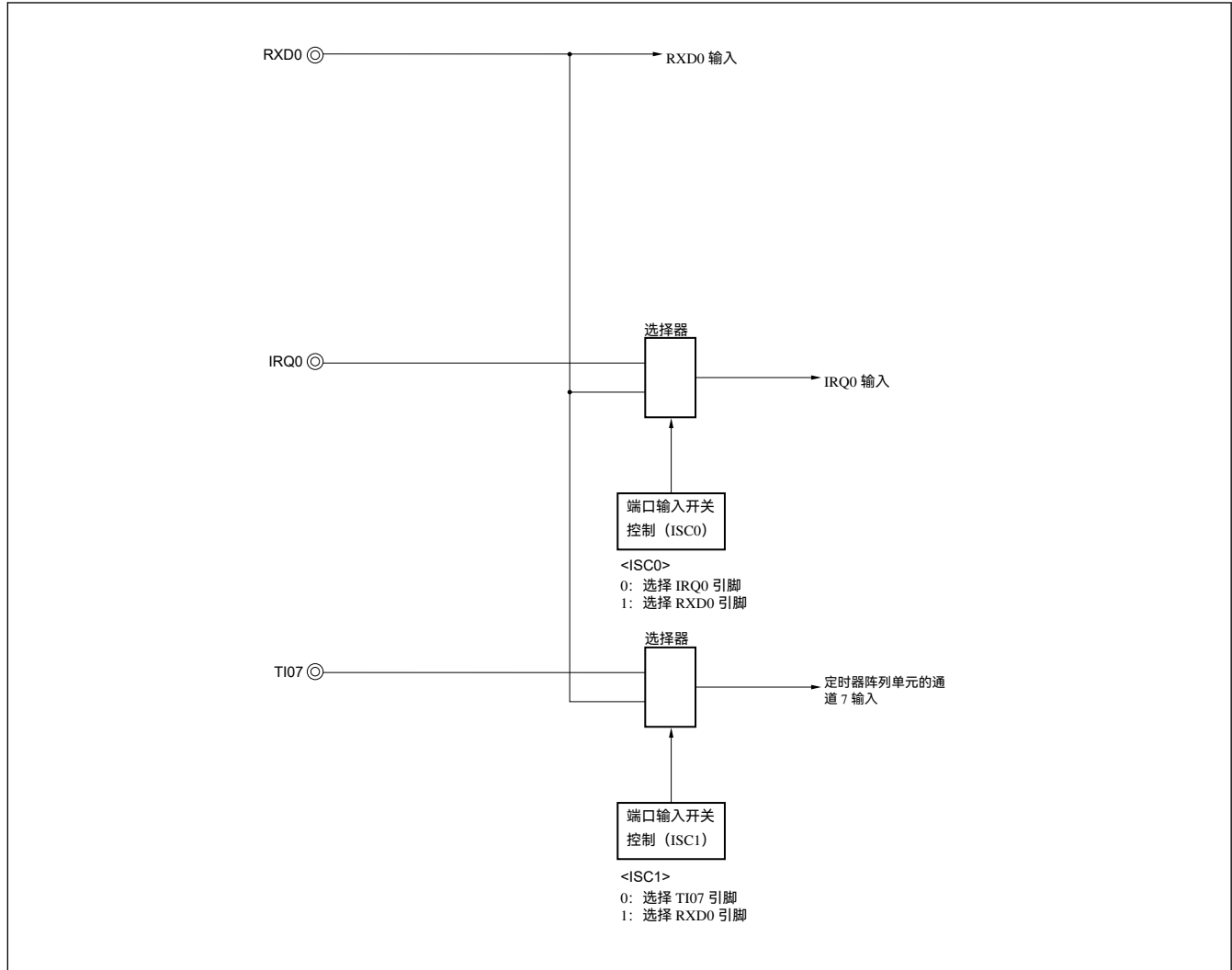


图 19.46 LIN 接收端口配置

LIN 通信操作所使用的外围功能如下。

<使用的外围功能>

- 外部中断（IRQ0），唤醒信号检测。
用途：检测唤醒信号的边沿和通信的开始。
- 定时器阵列单元的通道 7，波特率错误检测，断点检测。
用途：检测同步字段 (SF) 的长度，并将其除以位数，以便检测波特率错误。
在捕获模式下测量输入到 RXD0 的边缘的间隔。
测量低电平宽度以检测断裂场 (BF)。
- 串行阵列单元 0 (SAU0) 的通道 0 和 1 (UART0)。

19.8 简化通信的操作

这是一个时钟通信功能，它使用两条线路：串行时钟 (SCL) 和串行数据 (SDA)，与两个或多个设备进行通信。这个简化的 I²C 专为与单个设备（例如 EEPROM、闪存或 A/D 转换器）通信而设计，因此它仅作为主设备运行。

通过软件操作控制寄存器来设置启动和停止条件，同时遵守 I²C 总线线路的规范。

[数据传输和接收]

- 主控发送、主控接收（仅限单主控功能）
- ACK 输出功能*1 和 ACK 检测功能

- Data length of 8 bits
(when an address is transmitted, the address is specified by the upper 7 bits, and the least significant bit is used for R/W control.)
- Generation of start condition and stop condition for software

[Interrupt function]

- Transfer end interrupt (SAU0_IIC_TXRXI00/SAU0_IIC_TXRXI11)

[Error detection flag]

- Overrun error
- ACK error

[Functions not supported by simplified I²C]

- Slave transmission, slave reception
- Multi-master function (arbitration loss detection function)
- Clock stretch detection

Note 1. When receiving the last data, ACK is not output if 0 is written to the SOEm.SOE[n] bit and serial communication data output is stopped. See [\(2\) Processing flow](#) for details.

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3)

The channel supporting simplified I²C is channels 0 to 3 of SAU0. [section 19, Serial Array Unit \(SAU\)](#) show the channels supporting simplified I²C for each product.

Simplified I²C performs the following four types of communication operations:

- Address field transmission (see [section 19.8.1. Address Field Transmission](#))
- Data transmission (see [section 19.8.2. Data Transmission](#))
- Data reception (see [section 19.8.3. Data Reception](#))
- Stop condition generation (see [section 19.8.4. Stop Condition Generation.](#))

19.8.1 Address Field Transmission

Address field transmission is a transmission operation that first executes in I²C communication to identify the target for transfer (slave). After a start condition is generated, an address (7 bits) and a transfer direction (1 bit) are transmitted in one frame.

[Table 19.99](#) shows the specification of address field transmission of Simplified I²C.

Table 19.99 Specification of address field transmission of simplified I²C (1 of 2)

Simplified I ² C	IIC00	IIC11
Target channel	Channel 0 of SAU0	Channel 3 of SAU0
Pins used	SCL00, SDA00*1	SCL11, SDA11*1
Interrupt	SAU0_IIC_TXRXI00	SAU0_IIC_TXRXI11
	Transfer end interrupt only (setting the buffer empty interrupt is prohibited)	
Error detection flag	ACK error detection flag (SSRmn.PEF)	
Transfer data length	8 bits (transmitted with specifying the higher 7 bits as address and the least significant bit as R/W control)	
Transfer rate*2	Max.f _{MCK} /4 [Hz] (SDRmn.STCLK[6:0] = 1 or more) f _{MCK} : Operation clock frequency of target channel. However, the following conditions must be satisfied in each mode of I ² C: • Max. 1 MHz (fast mode plus) • Max. 400 kHz (fast mode) • Max. 100 kHz (standard mode)	

- Data length of 8 bits
(传输地址时，地址由高 7 位指定，最低有效位用于读/写控制。)

●生成软件的启动条件和停止条件

[中断功能]

- 传输结束中断（SAU0_IIC_TXRXI00/SAU0_IIC_TXRXI11）

[错误检测标志]

- 超限错误
- ACK错误

[简化版不支持的函数I²C]

- 从属传输，从属接收
- 多主控功能（仲裁损失检测功能）
- 时钟拉伸检测

注1: 当接收到最后一条数据时，如果SOEm.SOE[n] 位写入0，则不会输出ACK，并且串行通信数据输出已停止。详情请参见 [\(2\) 处理流程](#)。

注: m: 单元编号(m = 0) n: 通道编号(n = 0 至3)

支持简化的I²C的通道是 SAU0 的通道 0 到 3。第 19 节“串行阵列单元 (SAU)”显示了每个产品支持简化的I²C的通道。

简化的I²C执行以下四种类型的通信操作：

- [地址字段传输](#)（参见第 19.8.1 节地址字段传输）
- [数据传输](#)（参见第19.8.2节数据传输）
- [数据接收](#)（参见第19.8.3节数据接收）
- [停止条件生成](#)（参见第 19.8.4 节停止条件生成。）

19.8.1地址字段传输

地址字段传输是I²C 通信中首先执行的传输操作，用于识别传输目标（从设备）。生成起始条件后，在一个帧中传输地址（7 位）和传输方向（1 位）。

表 19.99 显示了简化的I²C的地址字段传输规范。

表 19.99 简化I²C地址字段传输规范（1/2）

简化的 I2C	IIC00	IIC11
目标频道	SAU0 的 0 通道	SAU0 的 3 频道
使用的引脚	SCL00, SDA00*1	SCL11, SDA11*1
打断	SAU0_IIC_TXRXI00	SAU0_IIC_TXRXI11
	仅传输结束中断（禁止设置缓冲区空中断）	
错误检测标志	ACK 错误检测标志 (SSRmn.PEF)	
传输数据长度	8 位（传输时指定高 7 位为地址，最低有效位为读/写控制）	
转账费率*2	Max.f _{MCK} /4[Hz] (SDRmn.STCLK[6:0] = 1 或更高) f _{MCK} : 目标通道的工作时钟频率。但是，在I ² C的每种模式下，必须满足以下条件： ●最高 1 MHz（快速模式增强） ●最大值400 kHz（快速模式） ●最高 100 kHz（标准模式）	

Table 19.99 Specification of address field transmission of simplified I²C (2 of 2)

Simplified I ² C	IIC00	IIC11
Data level	Non-reverse output (default: high level)	
Parity bit	No parity bit	
Stop bit	Appending 1 bit (for ACK transmission and reception timing)	
Data direction	MSB first	

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3)

- Note 1. To perform communication using simplified I²C, set the NMOS open drain output mode with the Port mn Pin Function Select Register (PmnPFS_A). For details, see [section 15, I/O Ports](#).
- Note 2. Use this operation within a range that satisfies the conditions above and the peripheral functions characteristics specified in the electrical characteristics. For details, see [section 27, Electrical Characteristics](#).

(1) Register setting

[Table 19.100](#) to [Table 19.105](#) show examples of the register contents for address field transmission of simplified I²C.

(a) Serial mode register mn (SMRmn)

Table 19.100 Example of serial mode register mn (SMRmn) contents for address field transmission of simplified I²C

Bit	Symbol	Set value	Function
0	MD0	0	Interrupt source of channel n 0: Transfer end interrupt
2:1	MD1[1:0]	10b	Setting of operation mode of channel n 1 0: Simplified I ² C mode
5:3	—	100b	Setting disabled (set to the initial value)
6	SIS0	0	Setting is fixed in the simplified I ² C mode
7	—	0	Setting disabled (set to the initial value)
8	STS	0	Selection of start trigger source 0: Only software trigger is valid (selected for simplified SPI, UART transmission, and simplified I ² C).
13:9	—	00000b	Setting disabled (set to the initial value)
14	CCS	0	Selection of transfer clock (f _{TCLK}) of channel n 0: Divided operation clock f _{MCK} specified by the CKS bit
15	CKS	0/1	Operation clock (f _{MCK}) of channel n 0: Prescaler output clock CKm0 set by the SPSm register 1: Prescaler output clock CKm1 set by the SPSm register

(b) Serial communication operation setting register mn (SCRmn)

Table 19.101 Example of serial communication operation setting register mn (SCRmn) contents for address field transmission of simplified I²C (1 of 2)

Bit	Symbol	Set value	Function
1:0	DLS[1:0]	11b	Setting of data length 1 1: 8-bit data length
3:2	—	01b	Setting disabled (set to the initial value)
5:4	SLC[1:0]	01b	Setting of stop bit 0 1: Appending 1 bit (ACK)
6	—	0	Setting disabled (set to the initial value)

表 19.99 简化I²C地址字段传输规范 (2/2)

简化的 I ² C	IIC00	IIC11
数据级别	非反向输出（默认值：高电平）	
奇偶校验位	无奇偶校验位	
停止位	附加 1 位（用于 ACK 发送和接收计时）	
数据方向	MSB 第一	

注: m: 单元编号(m = 0), n: 通道编号(n = 0 至3)

- 注1. 要使用简化的I²C进行通信，请使用端口mn引脚功能选择寄存器(PmnPFS_A)设置NMOS开漏输出模式。有关详细信息，请参阅第15节，I/O端口。
- 注2: 此操作应在满足上述条件和电气特性中规定的外围功能特性的范围内使用。详情请参见第27节“电气特性”。

(1) 寄存器设置

表 19.100 至表 19.105 显示了简化I²C地址字段传输的寄存器内容示例。

(a) 串行模式寄存器 mn (SMRmn)

表 19.100 简化的地址字段传输的串行模式寄存器 mn (SMRmn) 内容示例

少量	象征	设置值	功能
0	MD0	0	通道 n 的中断源 0: 传输结束中断
2:1	MD1[1:0]	10b	设置通道 n 的工作模式 1 0: 简化I ² C模式
5:3	—	100b	设置已禁用（已设置为初始值）
6	SIS0	0	设置在简化的I ² C模式下是固定的
7	—	0	设置已禁用（已设置为初始值）
8	STS	0	选择启动触发源 0: 仅软件触发有效（选择用于简化 SPI、UART 传输和简化I ² C）。
13:9	—	00000b	设置已禁用（已设置为初始值）
14	CCS	0	选择通道 n 的传输时钟 (f _{TCLK}) 0: 由 CKS 位指定的分频操作时钟 f _{MCK}
15	CKS	0/1	通道 n 的运行时钟(f _{MCK}) 0: 预分频器输出时钟CKm0由 SPSm 寄存器设置 1: 预分频器输出时钟 CKm1 由 SPSm 寄存器设置

(b) 串行通信操作设置寄存器 mn (SCRmn)

表 19.101 串行通信操作示例，设置寄存器 mn (SCRmn) 的内容以传输简化的地址字段I²C (1/2)

少量	象征	设置值	功能
1:0	DLS[1:0]	11b	数据长度设置 1 1: 8 位数 据长度
3:2	—	01b	设置已禁用（已设置为初始值）
5:4	SLC[1:0]	01b	设置停止位 0 1:附加 1 位（ACK）
6	—	0	设置已禁用（已设置为初始值）

Table 19.101 Example of serial communication operation setting register mn (SCRmn) contents for address field transmission of simplified I²C (2 of 2)

Bit	Symbol	Set value	Function
7	DIR	0	This bit is fixed in simplified I ² C mode because it is for simplified SPI and UART modes.
9:8	PTC[1:0]	00b	This bit is fixed in simplified I ² C mode because it is for UART mode.
10	EOC	0	This bit is fixed in simplified I ² C mode because it is for UART receive mode.
11	—	0	Setting disabled (set to the initial value)
13:12	DCP[1:0]	00b	This bit is fixed in simplified I ² C mode because it is for simplified SPI mode.
15:14	TRXE[1:0]	10b	Setting TRXE[1:0] = 10b is fixed in the simplified I ² C address field transmission

(c) Serial data register mn (SDRmn)

Table 19.102 Example of serial data register mn (SDRmn) contents for address field transmission of simplified I²C

Bit	Symbol	Set value	Function
7:0	DAT[7:0]	0x00 to 0xFF	Slave address + R/W (Transmit data setting)
8	DAT[8]	0	0 Fixed
15:9	STCLK[6:0]	0x00 to 0x7F	Baud rate setting (Operation clock (f _{MCK}) division setting)

(d) Serial output register m (SOM)

Start condition is generated by manipulating the SOM.SO[n] bit.

Table 19.103 Example of serial output register m (SOM) contents for address field transmission of simplified I²C

Bit	Symbol	Set value	Function
n	SO[n]	0/1	Serial data output of channel n 0: Serial data output value is 0 1: Serial data output value is 1
n+8	CKO[n]	0/1	Communication starts when a bit is 1 if the clock phase is non-reversed (SCRmn.DCP[0] = 0). If the clock phase is reversed (SCRmn.DCP[0] = 1), communication starts when a bit is 0

(e) Serial output enable register m (SOEm)

SOEm.SOE[n] = 0 until the start condition is generated, and SOEm.SOE[n] = 1 after generation.

Table 19.104 Example of serial output enable register m (SOEm) contents for address field transmission of simplified I²C

Bit	Symbol	Set value	Function
n	SOE[n]	0/1	Serial output enable or stop of channel n 0: Stop output by serial communication operation 1: Enable output by serial communication operation

(f) Serial channel start register m (SSm)

Set only the bit of the target channel to 1. SSm.SS[n] = 0 until the start condition is generated, and SSm.SS[n] = 1 after generation.

表 19.101 串行通信操作示例，设置寄存器 mn (SCRmn) 的内容以传输简化的地址字段I²C (2/2)

少量	象征	设置值	功能
7	DIR	0	该位在简化的I ² C模式下是固定的，因为它是为简化的 SPI 和 UART 模式设计的。
9:8	PTC[1:0]	00b	在简化的I ² C模式下，这一位是固定的，因为它是为 UART 模式设计的。
10	EOC	0	在简化的I ² C模式下，此位是固定的，因为它用于 UART 接收模式的。
11	—	0	设置已禁用（已设置为初始值）
13:12	DCP[1:0]	00b	该位在简化的I ² C模式下是固定的，因为它是为简化的 SPI 模式设计的。
15:14	TRXE[1:0]	10b	Setting TRXE[1:0] = 10b在简化的I ² C地址字段传输中是固定的

(c) 串行数据寄存器 mn (SDRmn)

表 19.102 简化地址字段传输的串行数据寄存器 mn (SDRmn) 内容示例 I²C

少量	象征	设置值	功能
7:0	DAT[7:0]	0x00 到 0xFF	从机地址+ 读/写 (传输数据设置)
8	DAT[8]	0	0 固定
15:9	STCLK[6:0]	0x00 至 0x7F	波特率设置 (运行时钟 (f _{MCK}) 分频设置)

(d) 串行输出寄存器 m (SOM)

起始条件是通过操作SOM.SO[n] 位生成的。

表 19.103 简化的地址字段传输的串行输出寄存器m (SOM) 内容示例 I²C

少量	象征	设置值	功能
n	所以[n]	0/1	通道 n 的串行数据输出 0: 串口数据输出值为 0 1: 串口数据输出值为 1
n+8	CKO[n]	0/1	当时钟相位未反转时，如果某个位为 1 (SCRmn.DCP[0] = 0)。如果时钟相位反转(SCRmn.DCP[0] = 1)，则当一个比特为 0

(e) 串行输出使能寄存器 m (SOEm)

SOEm.SOE[n] = 0直到生成起始条件， SOEm.SOE[n] = 1生成后。

表 19.104 串行输出使能寄存器m(SOEm) 内容示例，用于简化的I²C的地址字段传输

少量	象征	设置值	功能
n	SOE[n]	0/1	通道 n 的串行输出启用或停止 0: 停止通过串口通信操作输出 1: 通过串行通信操作启用输出

(f) 串行通道起始寄存器m (SSm)

仅将目标通道的位设置为1. SSm.SS[n] = 0，直到生成起始条件，并在生成后设置为SSm.SS[n] = 1。

Table 19.105 Example of serial channel start register m (SSm) contents for address field transmission of simplified I²C

Bit	Symbol	Set value	Function
n	SS[n]	0/1	Operation start trigger of channel n 0: No trigger operation 1: Set the SEm.SE[n] bit to 1 to place the channel in the communications waiting state

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3)

Note: 0/1: Set to 0 or 1 depending on the usage of the user

(2) Operation procedure

Table 19.106 shows the procedure for initial setting of simplified I²C address field transmission.

Table 19.106 Initial setting procedure for simplified I²C address field transmission

Step		Process	Detail
Procedure for initial setting of I ² C address field transmission	<1>	Starting initial setting	—
	<2>	Setting the SPSm register	Set the operation clock.
	<3>	Setting the SMRmn register	Set an operation mode.
	<4>	Setting the SCRmn register	Set a communication format.
	<5>	Setting the SDRmn register	Set a transfer baud rate (setting the transfer clock by dividing the operation clock (f _{MCK})).
	<6>	Setting the SOm register	Set the initial output level (1) of the serial data (SOm.SO[n]) and serial clock (SOm.CKO[n]).
	<7>	Setting port	Enable data output, clock output, and NMOS open-drain output of the target channel.
	<8>	Completing initial setting	—

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3)

(3) Processing flow

Figure 19.47 shows the timing of address field transmission.

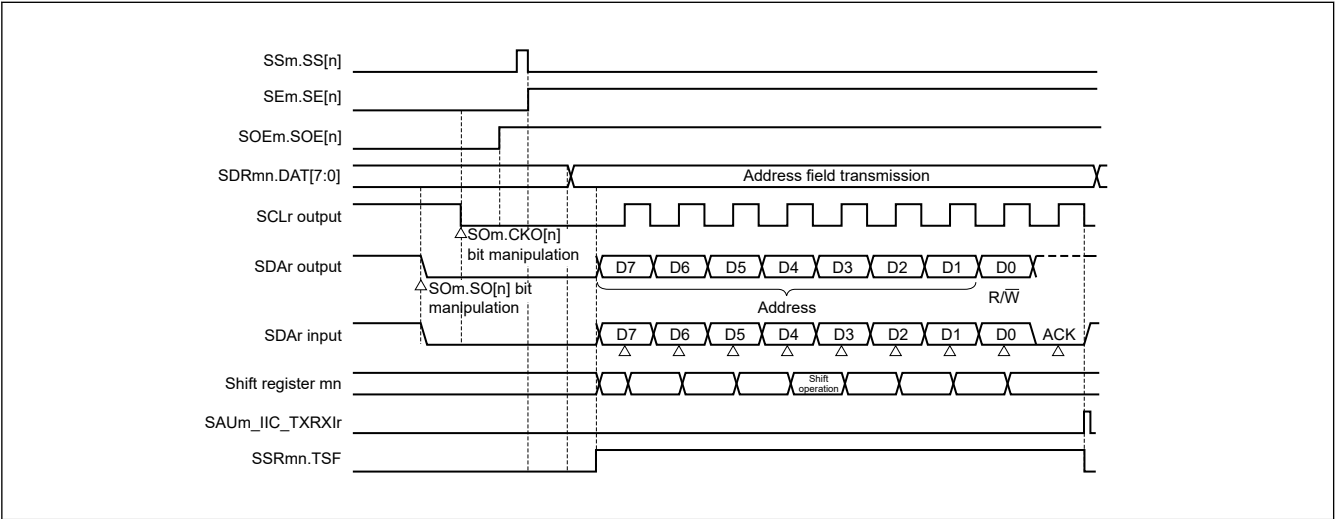


Figure 19.47 Timing of address field transmission

Table 19.107 shows the procedure for simplified I²C address field transmission.

表 19.105 串行通道起始寄存器m (SSm) 的内容示例，用于简化的I²C的地址字段传输

少量	象征	设置值	功能
n	SS[n]	0/1	通道 n 的操作启动触发 0: 无触发操作 1: 将 SEm.SE[n] 位设置为 1，将信道置于通信等待状态 状态

注: m: 单元编号(m = 0) n: 通道编号(n = 0 至3)

注意: 0/1: 根据用户使用情况设置为 0 或 1

(2) 操作流程

表 19.106 显示了简化的I²C地址字段传输的初始设置程序。

表 19.106 简化I²C地址字段传输的初始设置程序

步		过程	细节
I ² C 地址字段传输的初始设置程序	<1>	初始设置	—
	<2>	设置 SPSm 寄存器	设置运行时钟。
	<3>	设置 SMRmn 寄存器	设置操作模式。
	<4>	设置 SCRmn 寄存器	制定沟通模式。
	<5>	设置 SDRmn 寄存器	设置传输波特率（通过分频操作时钟来设置传输时钟(f _{MCK})）。
	<6>	设置 SOm 寄存器	设置串行数据（SOm.SO[n]）和串行时钟（SOm.CKO[n]）的初始输出电平（1）。
	<7>	设置端口	使能目标通道的数据输出、时钟输出和 NMOS 开漏输出。
	<8>	完成初始设置	—

注: m: 单元编号(m = 0) n: 通道编号(n = 0 至3)

(3) 加工流程

图 19.47 显示了地址字段传输的时序。

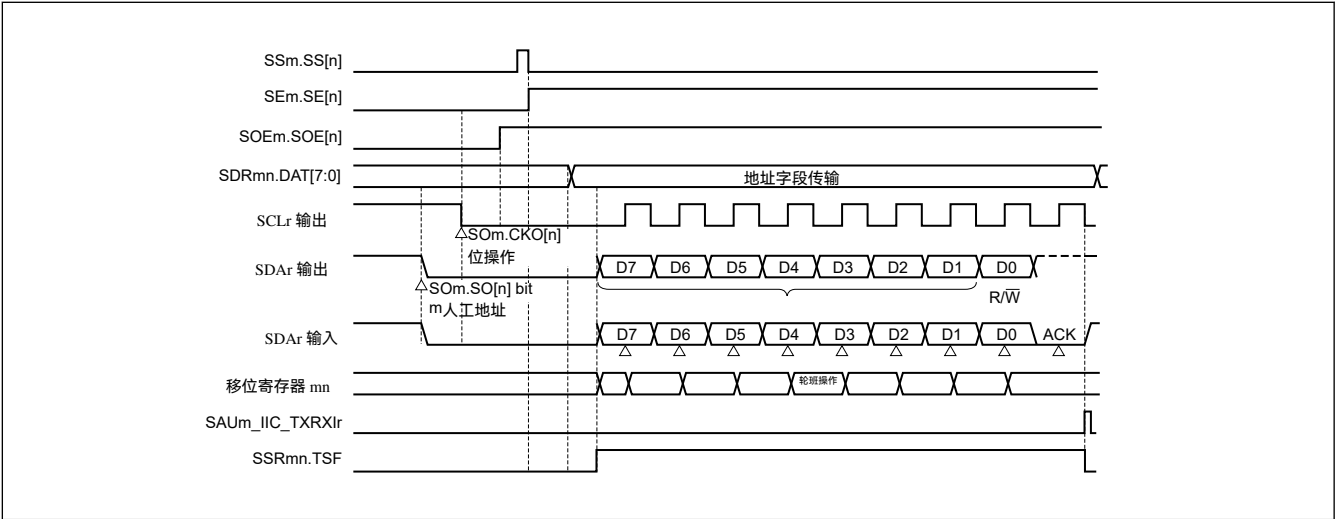


图 19.47 地址字段传输时序

表 19.107 显示了简化的I²C地址字段传输过程。

Table 19.107 Procedure for simplified I²C address field transmission

Step		Process	Detail
Procedure for simplified I ² C address field transmission	<1>	Transmitting address field	—
	<2>	Default setting	For the initial setting, see Table 19.106 .
	<3>	Writing 0 to the SOm.SO[n] bit	Set the SOm.SO[n] bit to 0
	<4>	Wait	Start condition generate Secure a hold time of SCL signal
	<5>	Writing 0 to the SOm.CKO[n] bit	Drive the SCL signal low and prepare for communications.
	<6>	Writing 1 to the SOEm.SOE[n] bit	Enable serial output
	<7>	Writing 1 to the SSm.SS[n] bit	Enable serial communications.
	<8>	Writing address and R /W data to SDRmn.DAT[7:0] bits	Transmitting address field
	<9>	Wait until transfer end interrupt generated.	Wait for address field transmission complete. Clear the interrupt request flag.
	<10>	Check if ACK responded. If yes, go to step <11>. If no, go to communication error processing	ACK response from the slave is confirmed in SSRmn.PEF bit. If ACK (SSRmn.PEF = 0), go to the next processing, if NACK (SSRmn.PEF = 1), go to error processing.
	<11>	Address field transmission completed	—
	<12>	Go to data transmission flow and data reception flow	—

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3)

19.8.2 Data Transmission

Data transmission is an operation to transmit data to the target for transfer (slave) after transmission of an address field. After all data are transmitted to the slave, a stop condition is generated and the bus is released.

[Table 19.108](#) shows the specification of data transmission of simplified I²C.

Table 19.108 Specification of data transmission of simplified I²C

Simplified I ² C	IIC00	IIC11
Target channel	Channel 0 of SAU0	Channel 3 of SAU0
Pins used	SCL00, SDA00*1	SCL11, SDA11*1
Interrupt	SAU0_IIC_TXRXI00	SAU0_IIC_TXRXI11
	Transfer end interrupt only (setting the buffer empty interrupt is prohibited.)	
Error detection flag	ACK error flag (SSRmn.PEF)	
Transfer data length	8 bits	
Transfer rate*2	Max.f _{MCK} /4 [Hz] (SDRmn[15:9] = 1 or more) However, the following condition must be satisfied in each mode of I ² C: - Max. 1 MHz (fast mode plus) - Max. 400 kHz (fast mode) - Max. 100 kHz (standard mode)	
Data level	Non-reverse output (default: high level)	
Parity bit	No parity bit	
Stop bit	Appending 1 bit (for ACK reception timing)	
Data direction	MSB first	

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3), mn = 00 to 03

Note: f_{MCK}: Operation clock frequency of target channel

Note 1. To perform communication using simplified I2C, set the NMOS open drain output mode with the Port mn Pin Function Select Register (PmnPFS_A). For details, see [section 15, I/O Ports](#).

Note 2. Use this operation within a range that satisfies the conditions above and the peripheral functions characteristics specified in the electrical characteristics. For details, see [section 27, Electrical Characteristics](#).

表 19.107 简化的I²C 地址字段传输程序

步		过程	细节
简化的I ² C地址字段传输程序	<1>	发送地址字段	—
	<2>	默认设置	有关初始设置，请参见表 19.106。
	<3>	将 0 写入 SOm.SO[n] 位	将 SOm.SO[n] 位设置为 0
	<4>	等待	起始条件生成 确保SCL信号的保持时间
	<5>	将 0 写入 SOm.CKO[n] 位	将 SCL 信号拉低，准备通信。
	<6>	向 SOEm.SOE[n] 位写入 1	启用串行输出
	<7>	将 1 写入 SSm.SS[n] 位	启用串行通信。
	<8>	写入地址和读/写数据 SDRmn.DAT[7:0] bits	发送地址字段
	<9>	等待直到生成传输结束中断。	等待地址字段传输完成。清除中断请求标志。
	<10>	检查 ACK 是否响应。如果响应，则转到步骤<11>。如果否，则进入通信错误处理流程。	从设备的 ACK 响应在 SSRmn.PEF 位中得到确认。如果 ACK(SSRmn.PEF = 0) 则进行下一步处理；如果 NACK(SSRmn.PEF = 1) 则进行错误处理。
	<11>	地址字段传输完成	—
	<12>	转到数据传输流程和数据接收流程	—

注: m: 单元编号(m = 0) n: 通道编号(n = 0 至3)

19.8.2 数据传输

数据传输是指在传输地址字段之后，将数据传输到目标（从设备）的操作。所有数据传输到从设备后，生成停止条件，总线被释放。

表 19.108 显示了简化的I²C的数据传输规范。

表 19.108 简化数据传输规范I²C

简化的 I2C	IIC00	IIC11
目标频道	SAU0 的 0 通道	SAU0 的 3 频道
使用的引脚	SCL00, SDA00*1	SCL11, SDA11*1
打断	SAU0_IIC_TXRXI00	SAU0_IIC_TXRXI11
	仅传输结束中断（禁止设置缓冲区空中断）。	
错误检测标志	ACK 错误标志 (SSRmn.PEF)	
传输数据长度	8 位	
转账费率*2	Max.f _{MCK} /4 [Hz] (SDRmn[15:9] = 1 or more) 但是，在I ² C的每个模式下，必须满足以下条件。 ●最高频率 1 MHz（快速模式增强版） ●最高频率 400 kHz（快速模式） ●最高100 kHz（标准模式）	
数据级别	非反向输出（默认值：高电平）	
奇偶校验位	无奇偶校验位	
停止位	附加 1 位（用于 ACK 接收计时）	
数据方向	MSB 第一	

注: m: 单元编号(m = 0) n: 通道编号(n = 0 至3) , mn = 00至03

注: f_{MCK}: 目标通道的工作时钟频率

注1: 要使用简化的I2C进行通信，请使用端口mn引脚功能选择寄存器（PmnPFS_A）将NMOS设置为开漏输出模式。有关详细信息，请参见第15节“I/O端口”。

注2: 此操作应在满足上述条件和电气特性中规定的外围功能特性的范围内使用。详情请参见第27节“电气特性”。

(1) Register setting

Table 19.109 to Table 19.114 show examples of the register contents for data transmission of simplified I²C.

(a) Serial mode register mn (SMRmn)

Do not manipulate this register during data transmission and reception.

Table 19.109 Example of serial mode register mn (SMRmn) contents for data transmission of simplified I²C

Bit	Symbol	Set value	Function
0	MD0	0	Interrupt source of channel n 0: Transfer end interrupt
2:1	MD1[1:0]	10b	Setting of operation mode of channel n 1 0: Simplified I ² C mode
5:3	—	100b	Setting disabled (set to the initial value)
6	SIS0	0	Setting is fixed in the simplified I ² C mode
7	—	0	Setting disabled (set to the initial value)
8	STS	0	Selection of start trigger source 0: Only software trigger is valid (selected for simplified SPI, UART transmission, and simplified I ² C)
13:9	—	00000b	Setting disabled (set to the initial value)
14	CCS	0	Selection of transfer clock (f _{TCLK}) of channel n 0: Divided operation clock f _{MCK} specified by the CKS bit
15	CKS	0/1	Operation clock (f _{MCK}) of channel n 0: Prescaler output clock CKm0 set by the SPSm register 1: Prescaler output clock CKm1 set by the SPSm register

(b) Serial communication operation setting register mn (SCRmn)

Do not manipulate the bits of this register, except the SCRmn.TRXE[1:0] bits, during data transmission and reception.

Table 19.110 Example of serial communication operation setting register mn (SCRmn) contents for data transmission of simplified I²C

Bit	Symbol	Set value	Function
1:0	DLS[1:0]	11b	Setting of data length 1 1: 8-bit data length
3:2	—	01b	Setting disabled (set to the initial value)
5:4	SLC[1:0]	01b	Setting of stop bit 0 1: Appending 1 bit (ACK)
6	—	0	Setting disabled (set to the initial value)
7	DIR	0	This bit is fixed in simplified I ² C mode because it is for simplified SPI and UART modes.
9:8	PTC[1:0]	00b	This bit is fixed in simplified I ² C mode because it is for UART mode.
10	EOC	0	This bit is fixed in simplified I ² C mode because it is for UART receive mode.
11	—	0	Setting disabled (set to the initial value)
13:12	DCP[1:0]	00b	This bit is fixed in simplified I ² C mode because it is for simplified SPI mode.
15:14	TRXE[1:0]	10b	Setting TRXE[1:0] = 10b is fixed in the simplified I ² C data transmission

(c) Serial data register mn (SDRmn)

During data transmission and reception, valid only lower 8-bits.

(1) 寄存器设置

表 19.109 至表 19.114 显示了简化的I²C数据传输的寄存器内容示例。

(a) 串行模式寄存器 mn (SMRmn)

在数据传输和接收过程中，请勿操作此寄存器。

表 19.109 简化数据传输的串行模式寄存器 mn (SMRmn) 内容示例
I²C

少量	象征	设置值	功能
0	MD0	0	通道 n 的中断源 0: 传输结束中断
2:1	MD1[1:0]	10b	设置通道 n 的工作模式 1 0: 简化I ² C模式
5:3	—	100b	设置已禁用（已设置为初始值）
6	SIS0	0	设置在简化的I ² C模式下是固定的
7	—	0	设置已禁用（已设置为初始值）
8	STS	0	选择启动触发源 0: 仅软件触发有效（选择用于简化 SPI、UART 传输和简化I ² C）
13:9	—	00000b	设置已禁用（已设置为初始值）
14	CCS	0	选择通道 n 的传输时钟 (f _{TCLK}) 0: 由 CKS 位指定的分频操作时钟 f _{MCK}
15	CKS	0/1	通道 n 的运行时钟(f _{MCK}) 0: 预分频器输出时钟 CKm0 由 SPSm 寄存器设置 1: 预分频器输出时钟 CKm1 由 SPSm 寄存器设置

(b) 串行通信操作设置寄存器 mn (SCRmn)

在数据传输和接收期间，不要操作此寄存器的位，但 SCRmn.TRXE[1:0] 位除外。

表 19.110 串行通信操作示例，设置寄存器 mn (SCRmn) 的内容以进行简化的数据传输I²C

少量	象征	设置值	功能
1:0	DLS[1:0]	11b	数据长度设置 1 1: 8 位数 据长度
3:2	—	01b	设置已禁用（已设置为初始值）
5:4	SLC[1:0]	01b	设置停止位 0 1: 附加 1 位 (ACK)
6	—	0	设置已禁用（已设置为初始值）
7	DIR	0	该位在简化的I ² C模式下是固定的，因为它是为简化的 SPI 和 UART 模式设计的。
9:8	PTC[1:0]	00b	在简化的I ² C模式下，这一位是固定的，因为它是为 UART 模式设计的。
10	EOC	0	在简化的I ² C模式下，此位是固定的，因为它用于 UART 接收模式的。
11	—	0	设置已禁用（已设置为初始值）
13:12	DCP[1:0]	00b	该位在简化的I ² C模式下是固定的，因为它是为简化的 SPI 模式设计的。
15:14	TRXE[1:0]	10b	在简化的数据传输中，设置TRXE[1:0] = 10b 是固定的。

(c) 串行数据寄存器 mn (SDRmn)

数据传输和接收过程中，只有低 8 位有效。

Table 19.111 Example of serial data register mn (SDRmn) contents for data transmission of simplified I²C

Bit	Symbol	Set value	Function
7:0	DAT[7:0]	0x00 to 0xFF	Transmit data (Transmit data setting)
8	DAT[8]	0	0 Fixed
15:9	STCLK[6:0]	0000000b to 1111111b	Baud rate setting Because the setting is completed by address field transmission, set the same value as before.

(d) Serial output register m (SOM)

Do not manipulate this register during data transmission and reception.

Table 19.112 Example of serial output register m (SOM) contents for data transmission of simplified I²C

Bit	Symbol	Set value	Function
n	SO[n]	0/1	The value varies depending on the communication data during communication operation.
n+8	CKO[n]	0/1	The value varies depending on the communication data during communication operation.

(e) Serial output enable register m (SOEm)

Do not manipulate this register during data transmission and reception.

Table 19.113 Example of serial output enable register m (SOEm) contents for data transmission of simplified I²C

Bit	Symbol	Set value	Function
n	SOE[n]	1	Serial output enable or stop of channel n 1: Enables output by serial communication operation.

(f) Serial channel start register m (SSm)

Do not manipulate this register during data transmission and reception.

Table 19.114 Example of serial channel start register m (SSm) contents for data transmission of simplified I²C

Bit	Symbol	Set value	Function
n	SS[n]	0/1	Operation start trigger of channel n 0: No trigger operation 1: Set the SEm.SE[n] bit to 1 to place the channel in the communications waiting state.

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3)

Note: 0/1: Set to 0 or 1 depending on the usage of the user.

(2) Processing flow

Figure 19.48 shows the timing of data transmission.

表 19.111 简化I²C数据传输的串行数据寄存器 mn (SDRmn) 内容示例

少量	象征	设置值	功能
7:0	DAT[7:0]	0x00 到 0xFF	传输数据 (传输数据设置)
8	DAT[8]	0	0 固定
15:9	STCLK[6:0]	0000000b to 1111111b	波特率设置 由于设置是通过地址字段传输完成的，因此设置值与之前相同。

(d) 串行输出寄存器 m (SOM)

在数据传输和接收过程中，请勿操作此寄存器。

表 19.112 串行输出寄存器m (SOM) 内容示例，用于简化I²C的数据传输

少量	象征	设置值	功能
n	所以[n]	0/1	该值会根据通信操作期间的通信数据而变化。
n+8	CKO[n]	0/1	该值会根据通信操作期间的通信数据而变化。

(e) 串行输出使能寄存器 m (SOEm)

在数据传输和接收过程中，请勿操作此寄存器。

表 19.113 简化数据传输的串行输出使能寄存器m (SOEm) 内容示例 I²C

少量	象征	设置值	功能
n	SOE[n]	1	通道 n 的串行输出启用或停止 1: 启用串行通信操作的输出。

(f) 串行通道起始寄存器m (SSm)

在数据传输和接收过程中，请勿操作此寄存器。

表 19.114 简化数据传输的串行通道起始寄存器m (SSm) 内容示例 I²C

少量	象征	设置值	功能
n	SS[n]	0/1	通道 n 的操作启动触发 0: 无触发操作 1: 将 SEm.SE[n] 位设置为 1，使信道处于通信等待状态。

注: m: 单元编号(m = 0), n: 通道编号(n = 0至3)

注意: 0/1: 根据用户的使用情况设置为 0 或 1。

(2) 加工流程

图 19.48 显示了数据传输的时间。

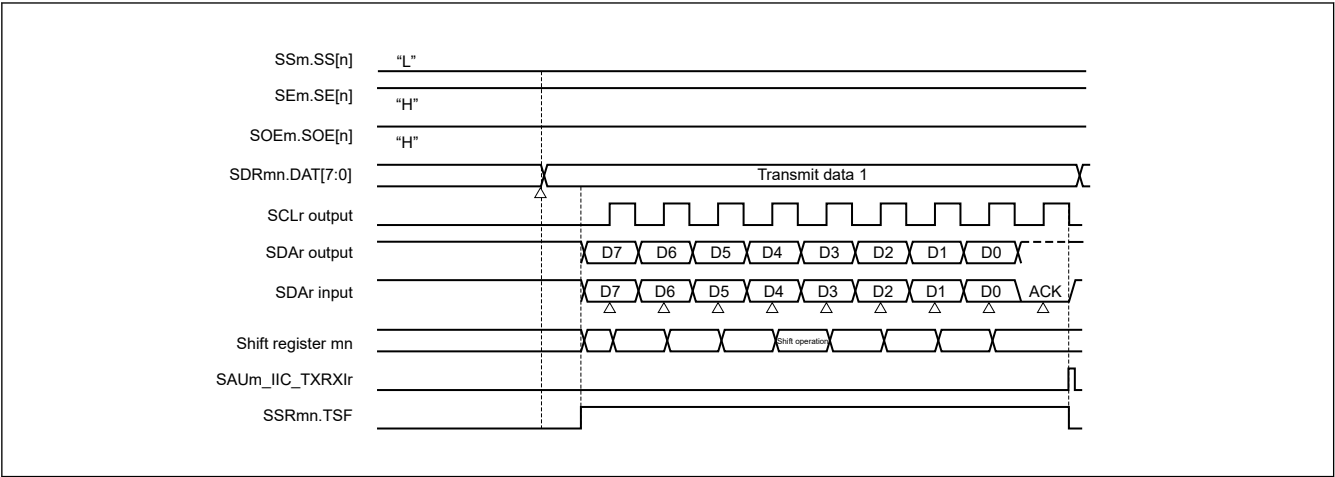


Figure 19.48 Timing of data transmission

Table 19.115 shows the procedure for simplified I²C data transmission.

Table 19.115 Procedure for simplified I²C data transmission

Step		Process	Detail
Procedure for simplified I ² C data transmission	<1>	Address field transmission completed	—
	<2>	Starting data transmission	—
	<3>	Writing data to SDRmn.DAT[7:0] bits	Transmission start by writing
	<4>	Wait until transfer end interrupt generated.	Wait for transmission complete. Clear the interrupt request flag.
	<5>	Check if ACK is responded. If yes, go to step <6>. If no, go to Communication error processing.	ACK acknowledgment from the slave. If ACK (SSRmn.PEF = 0), go to the next process. If NACK (SSRmn.PEF = 1), go to error handling.
	<6>	If data transfer completed, go to step <7>. If no, go to step <3>.	—
	<7>	Data transmission completed	—
	<8>	Stop condition generation	—

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3)

19.8.3 Data Reception

Data reception is an operation to receive data from the target for transfer (slave) after transmission of an address field. After all data are received from the slave, a stop condition is generated and the bus is released.

Table 19.116 shows the specification of data reception of simplified I²C.

Table 19.116 Specification of data reception of simplified I²C (1 of 2)

Simplified I ² C	IIC00	IIC11
Target channel	Channel 0 of SAU0	Channel 3 of SAU0
Pins used	SCL00, SDA00*1	SCL11, SDA11*1
Interrupt	SAU0_IIC_TXRXI00	SAU0_IIC_TXRXI11
	Transfer end interrupt only (Setting the buffer empty interrupt is prohibited.)	
Error detection flag	Overrun error detection flag (SSRmn.OVF) only	
Transfer data length	8 bits	

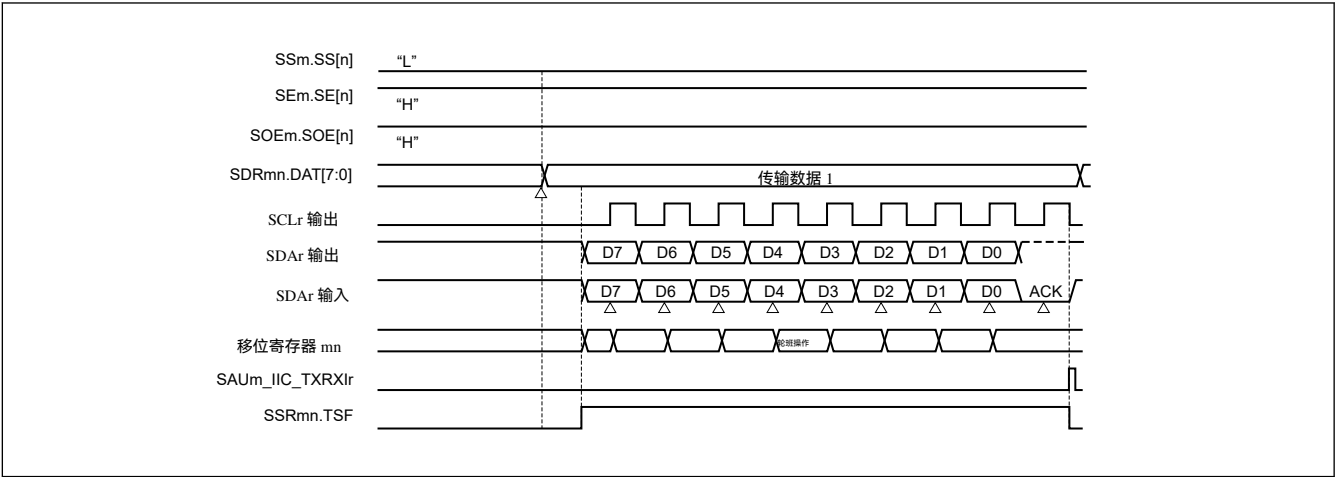


图 19.48 数据传输时序

表 19.115 显示了简化的I²C数据传输程序。

表 19.115 简化的I²C数据传输程序

步		过程	细节
简化的数据传输程序	<1>	地址字段传输完成	—
	<2>	开始数据传输	—
	<3>	将数据写入 SDRmn.DAT[7:0] 位	传输从写入开始
	<4>	等待直到生成传输结束中断。	等待传输完成。清除中断请求标志。
	<5>	检查是否收到 ACK 响应。如果是，则转到步骤<6>。如果否，则转到通信错误处理。	来自奴隶的 ACK 确认。如果收到 ACK (SSRmn.PEF= 0)，则转到下一个进程。如果收到 NACK (SSRmn.PEF= 1)，则转到错误处理。
	<6>	如果数据传输完成，请转至步骤<7>。如果没有，则转到步骤<3>。	—
	<7>	数据传输完成	—
	<8>	停止条件生成	—

注: m: 单元编号(m = 0), n: 通道编号(n = 0 至3)

19.8.3 数据接收

数据接收是指在发送地址字段后，从目标设备（从设备）接收待传输数据的操作。从从设备接收到所有数据后，生成停止条件并释放总线。

表 19.116 显示了简化的I²C的数据接收规范。

表 19.116 简化数据接收规范I²C (1/2)

简化的 I2C	IIC00	IIC11
目标频道	SAU0 的 0 通道	SAU0 的 3 频道
使用的引脚	SCL00, SDA00*1	SCL11, SDA11*1
打断	SAU0_IIC_TXRXI00	SAU0_IIC_TXRXI11
	仅传输结束中断（禁止设置缓冲区空中断）。	
错误检测标志	仅溢出错误检测标志 (SSRmn.OVF)	
传输数据长度	8 位	

Table 19.116 Specification of data reception of simplified I²C (2 of 2)

Simplified I ² C	IIC00	IIC11
Transfer rate*2	Max.f _{MCK} /4 [Hz] (SDRmn.STCLK[6:0] = 1 or more) However, the following conditions must be satisfied in each mode of I ² C: - Max. 1 MHz (fast mode plus) - Max. 400 kHz (fast mode) - Max. 100 kHz (standard mode)	
Data level	Non-reverse output (default: high level)	
Parity bit	No parity bit	
Stop bit	Appending 1 bit (ACK transmission)	
Data direction	MSB-first	

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3)

Note: f_{MCK}: Operation clock frequency of target channel

Note 1. To perform communication using simplified I²C, set the NMOS open drain output mode with the Port mn Pin Function Select Register (PmnPFS_A). For details, see [section 15, I/O Ports](#).

Note 2. Use this operation within a range that satisfies the conditions above and the peripheral functions characteristics specified in the electrical characteristics. For details, see [section 27, Electrical Characteristics](#).

(1) Register setting

[Table 19.117](#) to [Table 19.122](#) show examples of the register contents for data reception of simplified I²C.

(a) Serial mode register mn (SMRmn)

Do not manipulate this register during data transmission and reception.

Table 19.117 Example of serial mode register mn (SMRmn) contents for data reception of simplified I²C

Bit	Symbol	Set value	Function
0	MD0	0	Interrupt source of channel n 0: Transfer end interrupt
2:1	MD1[1:0]	10b	Setting of operation mode of channel n 1 0: Simplified I ² C mode
5:3	—	100b	Setting disabled (set to the initial value)
6	SIS0	0	Setting is fixed in the simplified I ² C mode
7	—	0	Setting disabled (set to the initial value)
8	STS	0	Selection of start trigger source 0: Only software trigger is valid (selected for simplified SPI, UART transmission, and simplified I ² C).
13:9	—	00000b	Setting disabled (set to the initial value)
14	CCS	0	Selection of transfer clock (f _{TCLK}) of channel n 0: Divided operation clock f _{MCK} specified by the CKSmn bit
15	CKS	0/1	Operation clock (f _{MCK}) of channel n 0: Prescaler output clock CKm0 set by the SPSm register 1: Prescaler output clock CKm1 set by the SPSm register

(b) Serial communication operation setting register mn (SCRmn)

Do not manipulate the bits of this register, except the TRXEmn [1:0] bits, during data transmission and reception.

表 19.116 简化版I²C数据接收规范 (2/2)

简化的 I2C	IIC00	IIC11
转账费率*2	Max.f _{MCK} /4 [Hz] (SDRmn.STCLK[6:0] = 1 or more) 然而，在I ² C的每个模式下，必须满足以下条件。 ●最高 1 MHz（快速模式增强） ●最高频率 400 kHz（快速模式） ●最高 100 kHz（标准模式）	
数据级别	非反向输出（默认值：高电平）	
奇偶校验位	无奇偶校验位	
停止位	附加 1 位（ACK 传输）	
数据方向	MSB优先	

注: m: 单元编号(m = 0) n: 通道编号(n = 0 至3)

注: fMCK: 目标通道的工作时钟频率

注1. 要使用简化的I²C进行通信，请使用端口mn引脚功能选择寄存器(PmnPFS_A)设置NMOS开漏输出模式。有关详细信息，请参阅第15节，I/O端口。

注2: 此操作应在满足上述条件和电气特性中规定的外围功能特性的范围内使用。详情请参见第27节“电气特性”。

(1) 寄存器设置

表 19.117 至表 19.122 显示了简化I²C数据接收的寄存器内容示例。

(a) 串行模式寄存器 mn (SMRmn)

在数据传输和接收过程中，请勿操作此寄存器。

表 19.117 简化I²C数据接收的串行模式寄存器 mn (SMRmn) 内容示例

少量	象征	设置值	功能
0	MD0	0	通道 n 的中断源 0: 传输结束中断
2:1	MD1[1:0]	10b	设置通道 n 的工作模式 1 0: 简化I ² C模式
5:3	—	100b	设置已禁用（已设置为初始值）
6	SIS0	0	设置在简化的I ² C模式下是固定的
7	—	0	设置已禁用（已设置为初始值）
8	STS	0	选择启动触发源 0: 仅软件触发有效（选择用于简化 SPI、UART 传输和简化I ² C）。
13:9	—	00000b	设置已禁用（已设置为初始值）
14	CCS	0	选择通道 n 的传输时钟 (fTCLK) 0: 由 CKSmn 位指定的分频操作时钟 fMCK
15	CKS	0/1	通道 n 的运行时钟(fMCK) 0: 预分频器输出时钟CKm0由 SPSm 寄存器设置 1: 预分频器输出时钟CKm1 由 SPSm 寄存器设置

(b) 串行通信操作设置寄存器 mn (SCRmn)

在数据传输和接收期间，不要操作此寄存器的位，但 TRXEmn[1:0] 位除外。

Table 19.118 Example of serial communication operation setting register mn (SCRmn) contents for data reception of simplified I²C

Bit	Symbol	Set value	Function
1:0	DLS[1:0]	11b	Setting of data length 1 1: 8-bit data length
3:2	—	01b	Setting disabled (set to the initial value)
5:4	SLC[1:0]	01b	Setting of stop bit 0 1: Appending 1 bit (ACK)
6	—	0	Setting disabled (set to the initial value)
7	DIR	0	This bit is fixed in simplified I ² C mode because it is for simplified SPI and UART modes.
9:8	PTC[1:0]	00b	This bit is fixed in simplified I ² C mode because it is for UART mode.
10	EOC	0	This bit is fixed in simplified I ² C mode because it is for UART receive mode.
11	—	0	Setting disabled (set to the initial value)
13:12	DCP[1:0]	00b	This bit is fixed in simplified I ² C mode because it is for simplified SPI mode.
15:14	TRXE[1:0]	01b	Setting TRXE[1:0] = 01b is fixed in the simplified I ² C data reception

(c) Serial data register mn (SDRmn)

Table 19.119 Example of serial data register mn (SDRmn) contents for data reception of simplified I²C

Bit	Symbol	Set value	Function
7:0	DAT[7:0]	0xFF	Receive data (Dummy transmit data setting 0xFF)
8	DAT[8]	0	0 Fixed
15:9	STCLK[6:0]	0x00 to 0x7F	Baud rate setting Because the setting is completed by address field transmission, set the same value as before.

(d) Serial output register m (SOM)

Do not manipulate this register during data transmission and reception.

Table 19.120 Example of serial output register m (SOM) contents for data reception of simplified I²C

Bit	Symbol	Set value	Function
n	SO[n]	0/1	The value varies depending on the communication data during communication operation.
n+8	CKO[n]	0/1	The value varies depending on the communication data during communication operation.

(e) Serial output enable register m (SOEm)

Do not manipulate this register during data transmission, and reception.

Table 19.121 Example of serial output enable register m (SOEm) contents for data reception of simplified I²C

Bit	Symbol	Set value	Function
n	SOE[n]	0/1	Serial output enable or stop of channel n 0: Stop output by serial communication operation 1: Enable output by serial communication operation

(f) Serial channel start register m (SSm)

Do not manipulate this register during data transmission and reception.

表 19.118 串行通信操作示例，设置寄存器 mn (SCRmn) 的内容以接收简化的I²C数据

少量	象征	设置值	功能
1:0	DLS[1:0]	11b	数据长度设置 1 1: 8 位数 据长度
3:2	—	01b	设置已禁用（已设置为初始值）
5:4	SLC[1:0]	01b	设置停止位 0 1: 附加 1 位（ACK）
6	—	0	设置已禁用（已设置为初始值）
7	DIR	0	该位在简化的I ² C模式下是固定的，因为它是为简化的 SPI 和 UART 模式设计的。
9:8	PTC[1:0]	00b	在简化的I ² C模式下，这一位是固定的，因为它是为 UART 模式设计的。
10	EOC	0	在简化的I ² C模式下，此位是固定的，因为它用于 UART 接收模式的。
11	—	0	设置已禁用（已设置为初始值）
13:12	DCP[1:0]	00b	该位在简化的I ² C模式下是固定的，因为它是为简化的 SPI 模式设计的。
15:14	TRXE[1:0]	01b	在简化的I ² C数据接收中，设置 TRXE[1:0] = 01b 是固定的

(c) 串行数据寄存器 mn (SDRmn)

表 19.119 简化I²C数据接收的串行数据寄存器 mn (SDRmn) 内容示例

少量	象征	设置值	功能
7:0	DAT[7:0]	0xFF	接收数据 (虚拟传输数据设置0xFF)
8	DAT[8]	0	0 固定
15:9	STCLK[6:0]	0x00 至 0x7F	波特率设置 由于设置是通过地址字段传输完成的，因此设置值与之前相同。

(d) 串行输出寄存器 m (SOM)

在数据传输和接收过程中，请勿操作此寄存器。

表 19.120 简化的串行输出寄存器m(SOM) 内容示例，用于接收数据I²C

少量	象征	设置值	功能
n	所以[n]	0/1	该值会根据通信操作期间的通信数据而变化。
n+8	CKO[n]	0/1	该值会根据通信操作期间的通信数据而变化。

(e) 串行输出使能寄存器 m (SOEm)

数据传输和接收期间，请勿操作此寄存器。

表 19.121 简化数据接收的串行输出使能寄存器m(SOEm) 内容示例
I²C

少量	象征	设置值	功能
n	SOE[n]	0/1	通道 n 的串行输出启用或停止 0: 停止通过串口通信操作输出 1: 通过串行通信操作启用输出

(f) 串行通道起始寄存器m (SSm)

在数据传输和接收过程中，请勿操作此寄存器。

Table 19.122 Example of serial channel start register m (SSm) contents for data reception of simplified I²C

Bit	Symbol	Set value	Function
n	SS[n]	0/1	Operation start trigger of channel n 0: No trigger operation 1: Set the SEm.SE[n] bit to 1 to place the channel in the communications waiting state.

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3)

Note: 0/1: Set to 0 or 1 depending on the usage of the user

(2) Processing flow

Figure 19.49 shows the timing of data reception.

表 19.122 简化数据接收的串行通道起始寄存器m (SSm) 内容示例
I²C

少量	象征	设置值	功能
n	SS[n]	0/1	通道 n 的操作启动触发 0: 无触发操作 1: 将 SEm.SE[n] 位设置为 1, 将信道置于通信等待状态 状态。

注: m: 单元编号(m = 0) n: 通道编号(n = 0至3)

注意: 0/1: 根据用户使用情况设置为 0 或 1

(2) 加工流程

图 19.49 显示了数据接收的时间。

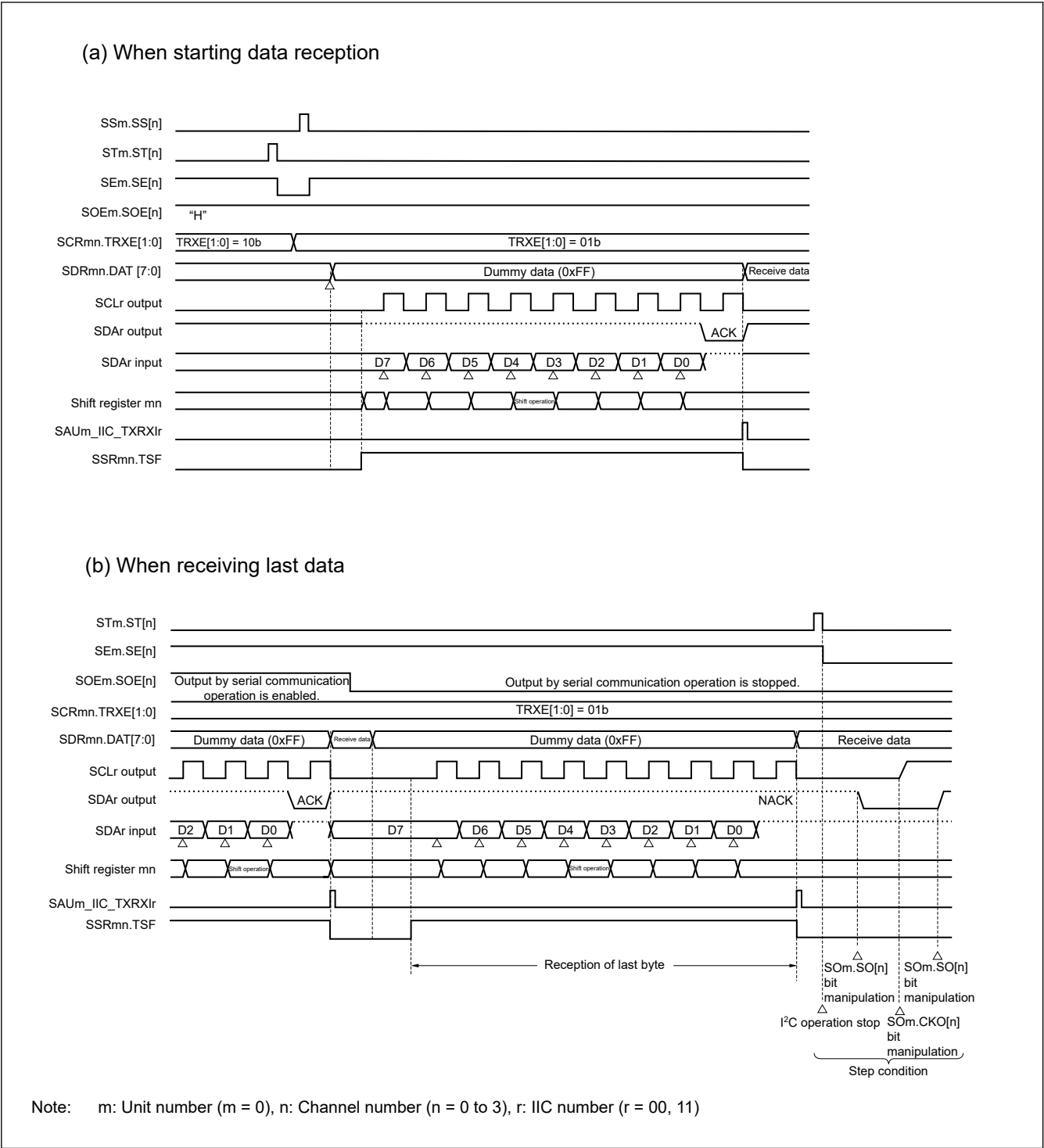


Figure 19.49 Timing of data reception

Table 19.123 shows the procedure for data reception.

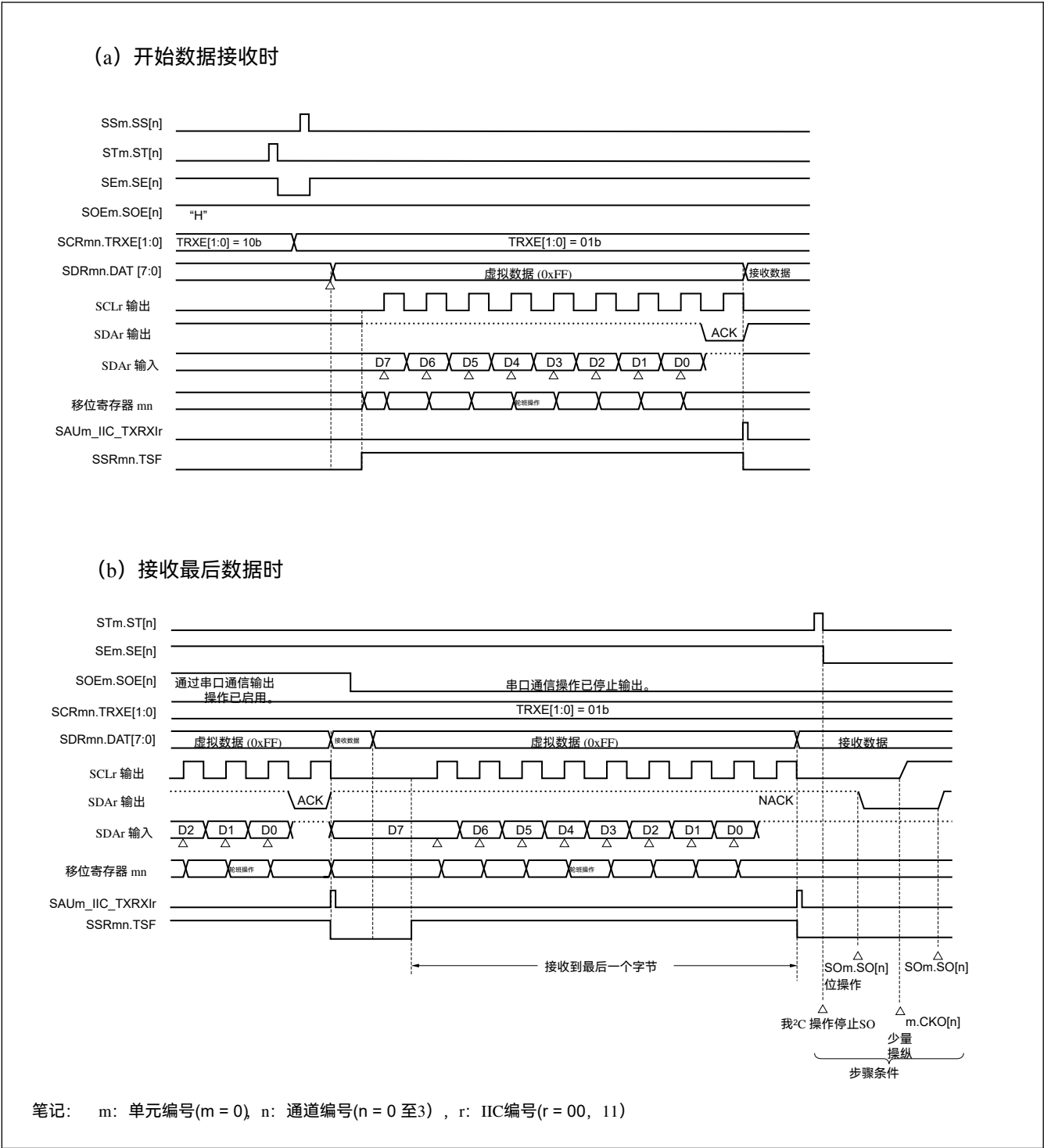


图 19.49 数据接收时序

表 19.123 显示了数据接收程序。

Table 19.123 Procedure for data reception

Step		Process	Detail
Procedure for data reception	<1>	Address field transmission completed	—
	<2>	Data reception	—
	<3>	Writing 1 to the STm.ST[n] bit	Stop operation for rewriting SCRmn register.
	<4>	Writing 01b to the SCRmn.TRXE[1:0] bits	Set the operation of the channel to receive-only mode.
	<5>	Writing 1 to the SSm.SS[n] bit	Operation restart
	<6>	Check if the last byte is received. If yes, go to step <7>. If No, go to step <8>.	Disable output so that it is not the ACK response to the last received data.
	<7>	Writing 0 to the SOEm.SOE[n] bit	
	<8>	Writing dummy data (0xFF) to SDRmn.DAT[7:0] bits	Starting reception operation
	<9>	Check if transfer end interrupt generated If yes, go to step <10>. If No, go to step <9>.	Wait for the completion of reception. Clear the interrupt request flag.
	<10>	Reading SDRmn.DAT[7:0] bits	Reading receive data, perform processing (stored in the RAM, for example).
	<11>	Check if data transfer completed. If yes, go to step <12>. If No, go to step <6>.	—
	<12>	Data reception completed	—
	<13>	Stop condition generation	—

Note: ACK is not output when the last data is received (NACK). Communication is then completed by setting 1 in the ST[n] bit of serial channel stop register m (STm) to stop operation and generating a stop condition.

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3)

19.8.4 Stop Condition Generation

After all data are transmitted to or received from the target slave, a stop condition is generated and the bus is released.

(1) Processing flow

Figure 19.50 shows the timing of stop condition generation.

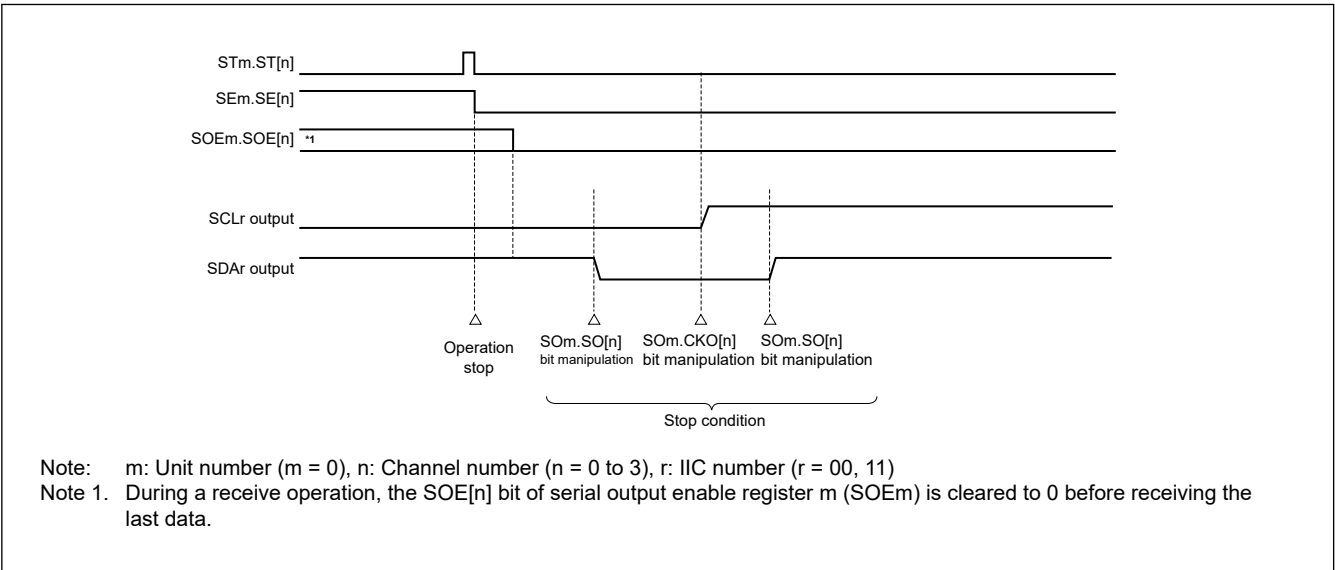


Figure 19.50 Timing of stop condition generation

Table 19.124 shows the procedure for stop condition generation.

表 19.123 数据接收程序

步		过程	细节
数据接收程序	<1>	地址字段传输完成	—
	<2>	数据接收	—
	<3>	将 1 写入 STm.ST[n] 位	停止对 SCRmn 寄存器的重写操作。
	<4>	将 01b 写入 SCRmn.TRXE[1:0] 位	将频道的操作模式设置为仅接收模式。
	<5>	将 1 写入 SSm.SS[n] 位	操作重启
	<6>	检查是否已接收到最后一个字节。 如果是，则转到步骤<7>。 如果否，则转到步骤<8>。	禁用输出，使其不是对最后接收到的数据的 ACK 响应。
	<7>	向 SOEm.SOE[n] 位写入 0	
	<8>	向目标写入虚拟数据 (0xFF) SDRmn.DAT[7:0] bits	开始接收操作
	<9>	检查是否生成了传输结束中断。如果是，则转到步骤<10>; 如果否，则转到步骤<9>。	等待接待工作完成。 清除中断请求标志。
	<10>	读取 SDRmn.DAT[7:0] 位	读取接收数据，执行处理（例如，存储在 RAM 中的数据）。
	<11>	检查数据传输是否完成。 如果是，则转到步骤<12>。 如果否，则转到步骤<6>。	—
	<12>	数据接收完成	—
	<13>	停止条件生成	—

注意：当接收到最后一个数据时，不会输出 ACK（NACK）。然后，通过将串行通道停止寄存器m(STm)的 ST[n] 位设置为 1 来停止操作并生成停止条件，从而完成通信。

注：m：单元编号(m = 0) n：通道编号(n = 0 至3)

19.8.4 停止条件生成

当所有数据都已传输到目标从设备或从目标从设备接收后，就会生成停止条件并释放总线。

(1) 加工流程

图 19.50 显示了停止条件生成的时序。

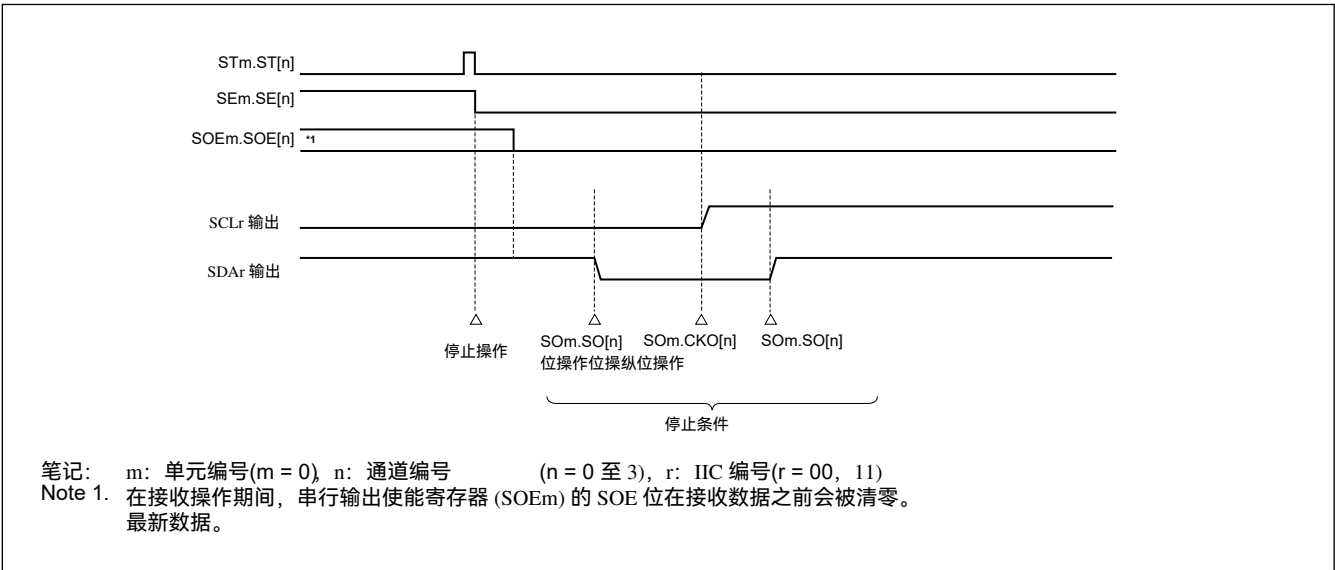


图 19.50 停止条件生成的时序

表 19.124 显示了停止条件生成的程序。

Table 19.124 Procedure for stop condition generation

Step		Process	Detail
Procedure for stop condition generation	<1>	Completion of data transmission and data reception	—
	<2>	Starting generation of stop condition	—
	<3>	Writing 1 to the STm.ST[n] bit (the SEm.SE[n] bit is cleared to 0)	Stop operation (SOm.CKO[n] can be manipulated).
	<4>	Writing 0 to the SOEm.SOE[n] bit	Disable output (SOm.SO[n] can be manipulated).
	<5>	Writing 0 to the SOm.SO[n] bit	—
	<6>	Writing 1 to the SOm.CKO[n] bit	Timing to satisfy the low width standard of SCL for the I ² C bus.
	<7>	Wait	Secure a wait time so that the specifications of I ² C on the slave side are satisfied.
	<8>	Writing 1 to the SOm.SO[n] bit	—
	<9>	End of I ² C communication	—

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3)

19.8.5 Calculating Transfer Rate

The transfer rate for simplified I²C communication can be calculated by the following expressions.

(Transfer rate) = {Operation clock (f_{MCK}) frequency of target channel} ÷ (SDRmn.STCLK[6:0] + 1) ÷ 2

Note: SDRmn.STCLK[6:0] must not be set to 0x00. Set SDRmn.STCLK[6:0] to 0x01 or greater.

The duty ratio of the SCL signal output by the simplified I²C is 50%. The I²C bus specifications define that the low-level width of the SCL signal is longer than the high-level width. If 400 kbps (fast mode) or 1 Mbps (fast mode plus) is specified, therefore, the low-level width of the SCL output signal becomes shorter than the value specified in the I²C bus specifications. Make sure that the SDRmn.STCLK[6:0] value satisfies the I²C bus specifications.

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3)

The operation clock (f_{MCK}) is determined by serial clock select register m (SPSm) and CKS bit of serial mode register mn (SMRmn). See Table 19.69.

Table 19.125 shows an example of setting an I²C transfer rate where f_{MCK} = PCLKB = 32 MHz.

Table 19.125 Example of setting I²C transfer rate where f_{MCK} = PCLKB = 32 MHz

I ² C transfer mode (desired transfer rate)	PCLKB = 32 MHz			
	Operation clock (f _{MCK})	SDRmn.STCLK[6:0]	Calculated transfer rate	Error from desired transfer rate
100 kHz	PCLKB/2	79	100 kHz	0.0%
400 kHz	PCLKB	41	380 kHz	5.0% ^{*1}
1 MHz	PCLKB	18	0.84 MHz	16.0% ^{*1}

Note 1. The error cannot be set to about 0% because the duty ratio of the SCL signal is 50%.

19.8.6 Procedure for Processing Errors that Occurred during Simplified I²C Communication

The procedure for processing errors that occurred during simplified I²C communication is described in Table 19.126 and Table 19.127.

表 19.124 停止条件生成程序

步		过程	细节
停止条件生成程序	<1>	数据传输和数据接收完成	—
	<2>	开始生成停止条件	—
	<3>	将 1 写入 STm.ST[n] 位 (SEm.SE[n] 位被清除为 0)	停止操作 (SOm.CKO[n] 可以被操纵)。
	<4>	向 SOEm.SOE[n] 位写入 0	禁用输出 (SOm.SO[n] 可以被操纵)。
	<5>	将 0 写入 SOm.SO[n] 位	—
	<6>	向 SOm.CKO[n] 位写入 1	时序满足I ² C总线的SCL低宽度标准。
	<7>	等待	确保等待时间，以便满足从端I ² C的规范。
	<8>	将 1 写入 SOm.SO[n] 位	—
	<9>	I ² C通信结束	—

注: m: 单元编号(m = 0) n: 通道编号(n = 0 至3)

19.8.5 计算转账费率

简化的I²C通信的传输速率可以通过以下表达式计算。

(传输速率) = {操作时钟(f_{MCK}) 目标信道频率} ÷ (SDRmn.STCLK[6:0] + 1) ÷ 2

注意: SDRmn.STCLK[6:0] 不得设置为0x00. 将 SDRmn.STCLK[6:0] 设置为0x01 或更高。

简化后的I²C输出的SCL信号的占空比为50%。I²C总线规范规定SCL信号的低电平宽度大于高电平宽度。因此，如果指定400 kbps（快速模式）或1 Mbps（快速模式增强版），则SCL输出信号的低电平宽度将小于I²C总线规范中规定的值。请确保SDRmn.STCLK[6:0] 值满足I²C总线规范。

注: m: 单元编号(m = 0) n: 通道编号(n = 0 至3)

操作时钟(f_{MCK}) 由串行时钟选择寄存器m (SPSm) 和串行模式寄存器mn (SMRmn) 的CKS位确定。参见表19.6 9。

表 19.125 显示了设置I²C转移率的示例，其中f_{MCK} = PCLKB = 32 MHz.

表 19.125 设置I²C转移率的示例，其中f_{MCK} = PCLKB = 32 MHz

I ² C 传输模式（期望传输速率）	PCLKB = 32 MHz			
	运行时钟 (f _{MCK})	SDRmn.STCLK[6:0]	计算出的转移率	与预期传输速率的误差
100 kHz	PCLKB/2	79	100 kHz	0.0%
400 kHz	PCLKB	41	380 kHz	5.0% ^{*1}
1 MHz	PCLKB	18	0.84 MHz	16.0% ^{*1}

注1. 误差不能设置为约0%，因为SCL信号的占空比为50%。

19.8.6 简化通信过程中发生的错误的处理程序

简化的I²C通信过程中发生的错误的处理程序在表 19.126 和表 19.127 中进行了描述。

Table 19.126 Processing procedure for overrun error

Step	Software manipulation		State of the hardware	Remark
<1>	Read serial data register mn (SDRmn).	→	The BFF bit of the SSRmn register is set to 0 and channel n is enabled to receive data	This is to prevent an overrun error if the next reception is completed during error processing.
<2>	Read serial status register mn (SSRmn).		—	The error type is identified and the read value is used to clear the error flag.
<3>	Write 1 to serial flag clear trigger register mn (SIRmn).	→	The error flag is cleared.	Only the error during reading can be cleared, by writing the value read from the SSRmn register to the SIRmn register without modification.

Table 19.127 Processing procedure for ACK error in simplified I²C mode

Step	Software manipulation		State of the hardware	Remark
<1>	Read serial status register mn (SSRmn).	→	—	The error type is identified and the read value is used to clear the error flag.
<2>	Write serial flag clear trigger register mn (SIRmn).		The error flag is cleared.	Only the error during reading can be cleared, by writing the value read from the SSRmn register to the SIRmn register without modification.
<3>	Set the ST[n] bit of serial channel stop register m (STm) to 1.	→	The SE[n] bit of serial channel enable status register m (SEm) is set to 0 and channel n stops operation.	The slave is not ready for reception because ACK is not returned. Therefore, a stop condition is created, the bus is released, and communication is started again from the start condition. Or, a restart condition is generated and transmission can be redone from address transmission.
<4>	Create a stop condition.	→	—	
<5>	Create a start condition.		—	
<6>	Set the SS[n] bit of serial channel start register m (SSm) to 1.	→	The SE[n] bit of serial channel enable status register m (SEm) is set to 1 and channel n is enabled to operate.	—

Note: m: Unit number (m = 0), n: Channel number (n = 0 to 3)

表 19.126 超限误差处理程序

步	软件操作		硬件状态	评论
<1>	读取串行数据寄存器 mn (SDRmn)。	→	SSRmn寄存器的BFF位设置为0，通道 n被启用以接收数据	这是为了防止在错误处理期间完成下一次接收时发生溢出错误。
<2>	读取串行状态寄存器 mn (SSRmn)。		—	识别错误类型，并使用读取的值清除错误标志。
<3>	向串行标志清除触发寄存器 mn (SIRmn) 写入 1。	→	错误标志已清除。	只能通过将从 SSRmn 寄存器读取的值原封不动地写入 SIRmn 寄存器来清除读取过程中出现的错误。

表 19.127 简化I²C模式下 ACK 错误的处理流程

步	软件操作		硬件状态	评论
<1>	读取串行状态寄存器 mn (SSRmn)。	→	—	识别错误类型，并使用读取的值清除错误标志。
<2>	写入串行标志清除触发寄存器 mn (SIRmn)。		错误标志已清除。	只能通过将从 SSRmn 寄存器读取的值原封不动地写入 SIRmn 寄存器来清除读取过程中出现的错误。
<3>	将串行通道停止寄存器 m (STm) 的 ST[n] 位设置为 1。	→	串行通道使能状态寄存器m(SEm) 的 SE[n] 位设置为 0，通道 n 停止操作。	从设备未准备好接收数据，因为未收到 ACK 确认。因此，创建停止条件，释放总线，并从从设备重新启动通信。
<4>	创建停止条件。	→	—	
<5>	创建起始条件。		—	起始条件。 或者，生成重启条件，可以从地址传输重新进行传输。
<6>	将串行通道起始寄存器 m (SSm) 的 SS[n] 位设置为 1。	→	串行通道使能状态寄存器 m (SEm) 的 SE[n] 位设置为 1，通道 n 被启用以进行操作。	—

注: m: 单元编号(m = 0), n: 通道编号(n = 0至3)

20. I²C Bus Interface (IICA)

20.1 Overview

The I²C bus interface has the following three modes:

- Operation stop mode
- I²C bus mode (multimaster supported)
- Wakeup mode

Table 20.1 shows specifications of the I²C bus interface.

Table 20.1 I²C specifications (1 of 2)

Parameter	Specifications
Communications format	● I²C-bus format ● Master or slave mode selectable ● Automatic securing of the setup times, hold times, and bus-free times for the transfer rate
Transfer rate	● Standard-mode, up to 100 kbps ● Fast-mode supported, up to 400 kbps ● Fast-mode Plus supported, up to 1 Mbps
SCL clock	For master operation, the duty cycle of the SCLA0 clock is selectable
Issuing and detecting conditions	● Start, restart, and stop conditions are automatically generated ● Start conditions (including restart conditions) and stop conditions are detectable
Slave address	7- and 10-bit address formats supported, including simultaneous use
Acknowledgment	● The reception side returns ACK each time it has received 8-bit data ● The transmission side usually receives ACK after transmitting 8-bit data ● How ACK is generated when data is received depends on the setting of the timing of clock stretching as follows: – 8th cycle clock stretching is selected: By setting the IICCTL00.ACKE bit to 1 before releasing from the clock stretch state, ACK is generated at the falling edge of the 8th clock cycle of the SCLA0 pin – 9th cycle clock stretching is selected: ACK is generated if the IICCTL00.ACKE bit is set to 1 in advance
Wait function (clock stretching)	During reception, the following wait periods are available by holding the SCLA0 clock low: ● Waiting between the 8th and 9th clock cycles ● Waiting between the 9th clock cycle and the 1st clock cycle of the next transfer
Arbitration	● If multiple master devices generate start conditions at the same time, the communication between each master device will be the master device with the longer number of clocks before the data changes. ● When arbitration lost occurs, both the SCLA0 and SDAA0 lines of that master device become high impedance and the bus is released. ● The arbitration loss is detected by checking IICS0.ALD = 1 by software at the timing of the next interrupt request
Noise cancellation	Digital noise filters for both the SCLA0 and SDAA0 signals
Interrupt sources (IICA0_TXRXI)	● the local address is received ● an address is received while the all address match function is enabled ● an extension code is received ● a stop condition is detected
Module-stop function	Module-stop state can be set
IIC operating modes	● Master transmit ● Master receive ● Slave transmit ● Slave receive
Event link function (output)	● the local address is received ● an address is received while the all address match function is enabled ● an extension code is received ● a stop condition is detected

20.I²C总线接口（IICA）

20.1 概述

I²C总线接口有以下三种模式：

- 操作停止模式
- I²C 总线模式（支持多主控）
- 唤醒模式

表 20.1 显示了I²C总线接口的规格。

表 20.1I²C 规格（1/2）

范围	规格
通讯格式	● I²C- 总线格式 ●可选择主模式或从模式 ●自动确保传输速率的建立时间、保持时间和总线空闲时间
转账费率	●标准模式，最高可达100 kbps ●支持快速模式，最高可达400 kbps ●支持快速模式Plus，最高可达1Mbps
SCL时钟	对于主操作，SCLA0 时钟的占空比是可以选择的。
发布和检测条件	●启动、重启和停止条件自动生成 ●启动条件（包括重启条件）和停止条件均可检测。
从地址	支持 7 位和 10 位地址格式，包括同时使用。
致谢	●接收端每次收到8位数据后都会返回ACK。 ●发送端通常在发送完8位数据后会收到ACK。 ●接收到数据时，ACK 的生成方式取决于时钟拉伸时序的设置，如下所示： –选择第 8 个时钟周期的时钟拉伸：通过在解除时钟拉伸状态之前将 IICCTL00.ACKE 位设置为 1，会在 SCLA0 引脚的第 8 个时钟周期的下降沿生成 ACK 信号。 –选择第 9 个时钟周期拉伸：如果预先将 IICCTL00.ACKE 位设置为 1，则生成 ACK。
等待功能（时钟拉伸）	在接收过程中，将 SCLA0 时钟保持低电平可以设置以下等待时间： ●在第8个和第9个时钟周期之间等待 ●等待第 9 个时钟周期和下一次传输的第 1 个时钟周期之间的时间
仲裁	●如果多个主设备同时产生启动条件，则各主设备之间的通信将由数据变化之前时钟数较长的主设备进行。 ●当仲裁丢失时，该主设备的 SCLA0 和 SDAA0 线都会变为高阻抗，总线将被释放。 ●仲裁丢失是通过软件在下一次中断请求发生时检查IICS0.ALD = 1来检测的。
降噪	SCLA0 和 SDAA0 信号的数字噪声滤波器
中断源（IICA0_TXRXI）	●收到本地地址 ●在启用所有地址匹配功能的情况下收到地址 ●收到分机号 ●检测到停止条件
模块停止功能	可以设置模块停止状态
IIC 操作模式	●主传输 ●主接收 ●从属传输 ●奴隶接收
事件链接功能（输出）	●收到本地地址 ●在启用所有地址匹配功能的情况下收到地址 ●收到分机号 ●检测到停止条件

Table 20.1 I²C specifications (2 of 2)

Parameter	Specifications
Wakeup function	CPU can return from Software Standby mode and Snooze mode using a wakeup event

Figure 20.1 shows a block diagram of the I²C bus interface.

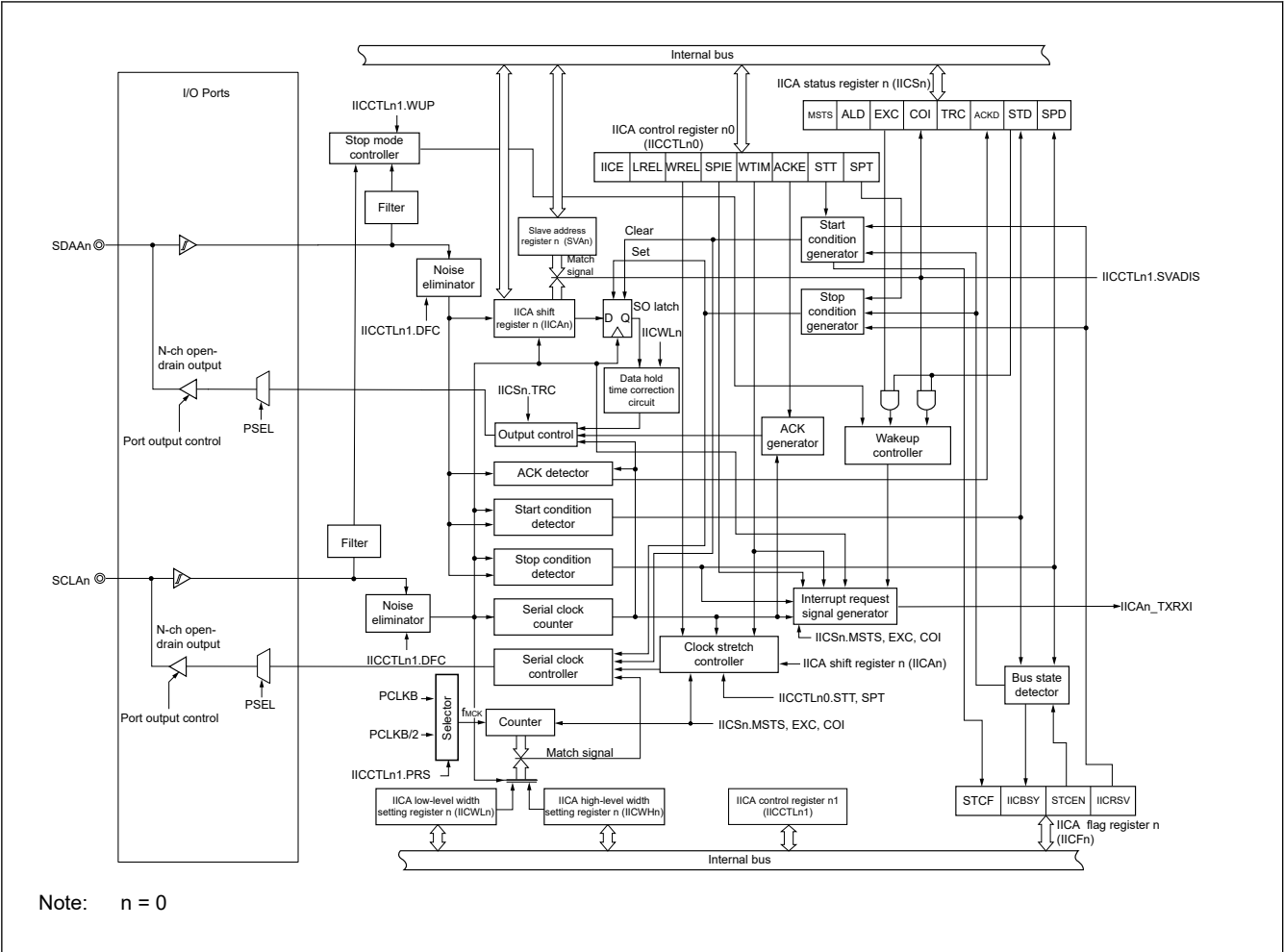


Figure 20.1 Block diagram of I²C bus interface

(1) Operation stop mode

This mode is used when serial transfers are not performed. The operating power can be reduced in this mode.

(2) I²C bus mode (multimaster supported)

This mode is used for 8-bit data transfers with several devices using two lines: a serial clock (SCLA0) line and a serial data bus (SDAA0) line.

This mode complies with the I²C bus format. In master mode, the master device can send start conditions, addresses, transfer directions, acknowledges (ACK), data, and stop conditions to the slave devices, through the serial data bus. The slave mode automatically detects these states and data by hardware. This function can simplify the part of application program that controls the I²C bus.

Since the SCLA0 and SDAA0 pins are used for open drain outputs, I²C bus interface (IICA) requires pull-up resistors for the serial clock line and the serial data bus line.

表 20.1I²C 规格 (2/2)

范围	规格
唤醒功能	CPU 可以通过唤醒事件从软件待机模式和休眠模式恢复。

图 20.1 显示了I²C总线接口的框图。

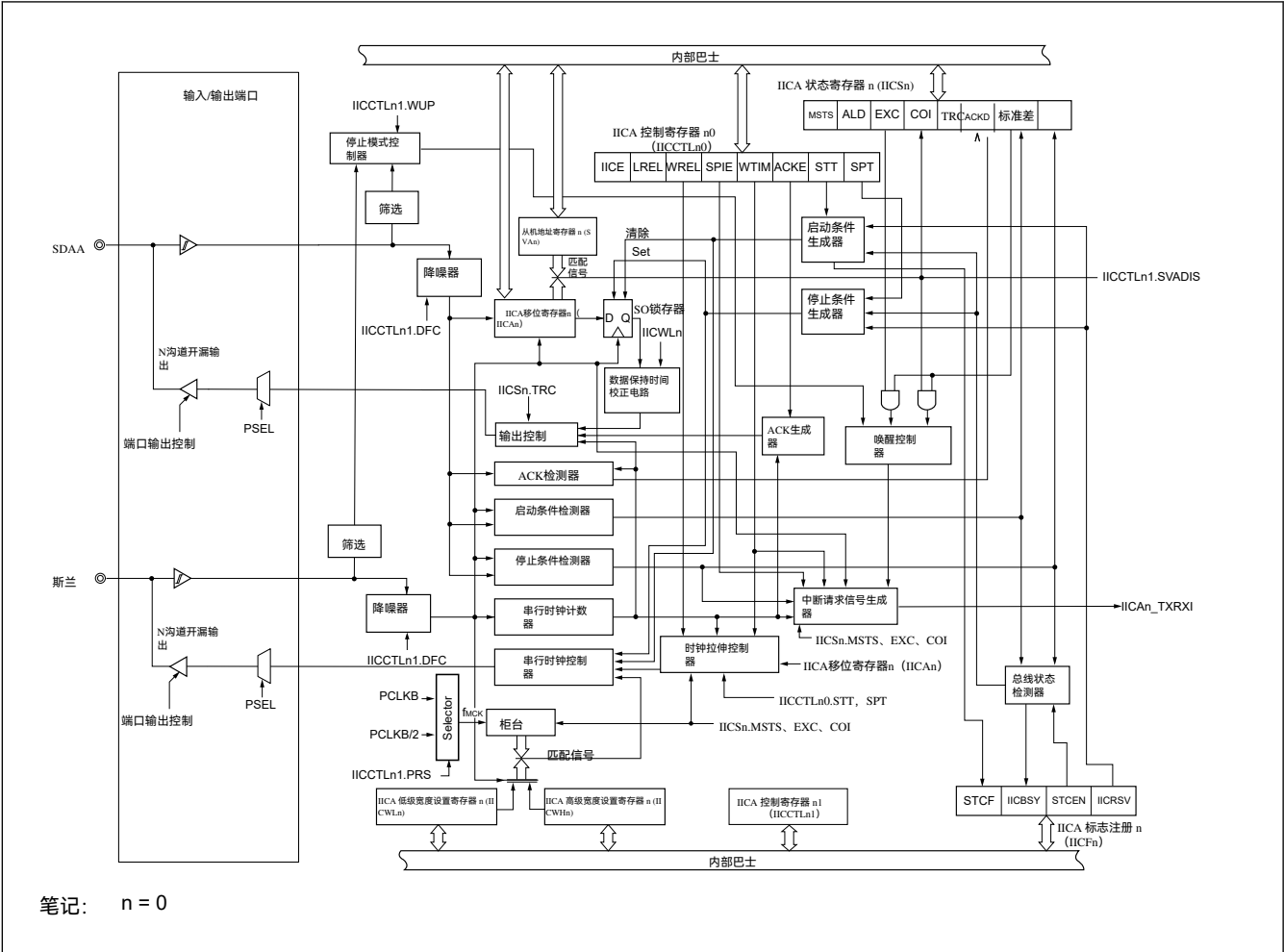


图 20.1I²C 总线接口框图

(1) 运行停止模式

当不进行串行传输时，使用这种模式。在此模式下，工作功耗会降低。

(2) I²C 总线模式（支持多主控）

此模式用于使用两条线路与多个设备进行 8 位数据传输：一条串行时钟 (SCLA0) 线路和一条串行数据总线 (SDAA0) 线路。

此模式符合I²C总线格式。在主模式下，主设备可以通过串行数据总线向从设备发送起始条件、地址、传输方向、确认 (ACK)、数据和停止条件。从设备通过硬件自动检测这些状态和数据。此功能可以简化应用程序中控制I²C总线的部分。

由于 SCLA0 和 SDAA0 引脚用于开漏输出，I²C 总线接口 (IICA) 需要为串行时钟线和串行数据总线线配备上拉电阻。

(3) Wakeup mode

The Software Standby mode can be released by generating an interrupt request signal (IICA0_TXRXI) when an extension code from the master device or the local address has been received while in Software Standby mode. This can be set by using the WUP bit of IICA control register 01 (IICCTL01).

The all address match function is enabled by setting the SVADIS bit of the IICCTL01 register to 1, allowing any received address is to be determined as a matched address.

(4) SO latch

The SO latch is used to retain the output level of SDAA0 pin.

(5) Wakeup controller

This circuit generates an interrupt request signal (IICA0_TXRXI) when the received address matches the address value set to the slave address register 0 (SVA0), when any address is received while the all address match function is enabled, or when an extension code is received.

(6) Serial clock counter

This counter counts the serial clock cycles that are output or input during transmit/receive operations and is used to verify that 8-bit data was transmitted or received.

(7) Interrupt request signal generator

This circuit controls the generation of interrupt request signals (IICA0_TXRXI). An I²C interrupt request is generated by the following two triggers:

- Falling edge of the 8th or 9th clock of the serial clock (set by the IICCTL00.WTIM bit)
- Interrupt request generated when a stop condition is detected (set by the IICCTL00.SPIE bit)

(8) Serial clock controller

In master mode, this circuit generates the serial clock, which is output using the SCLA0 pin.

(9) Clock stretch controller

This circuit controls the timing of clock stretching.

(10) ACK generator, stop condition detector, start condition detector, and ACK detector

These circuits generate or detect each state.

(11) Data hold time correction circuit

This circuit generates the hold time for data after the falling edge of the serial clock.

(12) Start condition generator

This circuit generates a start condition when the IICCTL00.STT bit is set to 1. However, while communication reservations are disabled (IICF0.IICRSV bit = 1) and the bus is busy (IICF0.IICBSY bit = 1), start condition requests are ignored and the IICF0.STCF bit is set to 1.

(13) Stop condition generator

This circuit generates a stop condition when the IICCTL00.SPT bit is set to 1.

(14) Bus state detector

This circuit detects whether the bus is released by detecting start conditions and stop conditions. However, as the bus state cannot be detected immediately after the IICA operation is enabled, the initial state is set by the IICF0.STCEN bit.

Figure 20.2 shows an example of the serial bus configuration using the I²C bus.

(3) 唤醒模式

软件待机模式可通过在接收到来自主设备或本地地址的扩展码时生成中断请求信号 (IICA0_TXRXI) 来解除。该信号可通过 IICA 控制寄存器 01 (IICCTL01) 的 WUP 位进行设置。

通过将 IICCTL01 寄存器的 SVADIS 位设置为 1 来启用所有地址匹配功能，允许将任何接收到的地址确定为匹配地址。

(4)SO锁存器

SO 锁存器用于保持 SDAA0 引脚的输出电平。

(5) 唤醒控制器

当接收到的地址与从地址寄存器 0 (SVA0) 中设置的地址值匹配时，当启用全地址匹配功能时接收到任何地址时，或者当接收到扩展代码时，该电路会生成中断请求信号 (IICA0_TXRXI)。

(6) 串行时钟计数器

该计数器对发送/接收操作期间输出或输入的串行时钟周期进行计数，并用于验证是否已发送或接收 8 位数据。

(7) 中断请求信号发生器

该电路控制中断请求信号 (IICA0_TXRXI) 的生成。I²C 中断请求由以下两个触发器生成：

- 串行时钟的第8或第9个时钟的下降沿（由IICCTL00.WTIM位设置）
- 检测到停止条件时生成的中断请求（由 IICCTL00.SPIE 位设置）

(8) 串行时钟控制器

在主模式下，该电路产生串行时钟，并通过 SCLA0 引脚输出。

(9) 时钟拉伸控制器

该电路控制时钟拉伸的时序。

(10) ACK生成器、停止条件检测器、启动条件检测器和ACK检测器

这些电路生成或检测每种状态。

(11) 数据保持时间校正电路

该电路产生串行时钟下降沿之后的数据保持时间。

(12) 启动条件生成器

当 IICCTL00.STT 位设置为 1 时，此电路产生启动条件。但是，当通信预留被禁用(IICF0.IICRSV bit = 1) 且总线繁忙时（IICF0.IICBSY 位= 1），启动条件请求将被忽略，并且 IICF0.STCF 位将被设置为 1。

(13) 停止条件生成器

当 IICCTL00.SPT 位设置为 1 时，该电路会产生停止条件。

(14) 总线状态检测器

该电路通过检测启动条件和停止条件来判断总线是否已释放。然而，由于在 IICA 操作使能后无法立即检测总线状态，因此初始状态由 IICF0.STCEN 位设置。

图 20.2 显示了使用I²C总线的串行总线配置示例。

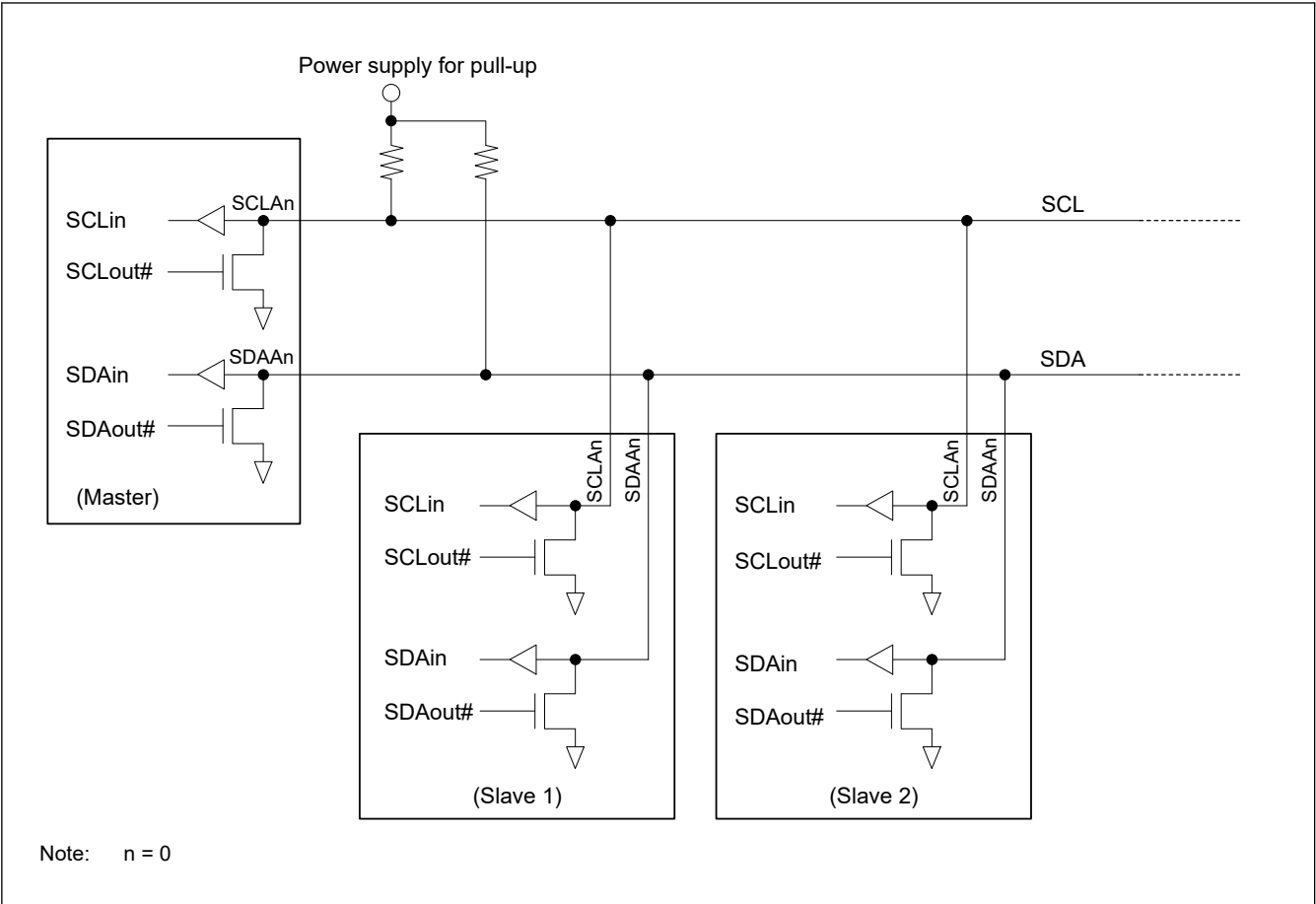


Figure 20.2 Example of the serial bus configuration using the I²C bus

20.2 Register Descriptions

20.2.1 IICA0 : IICA Shift Register 0

Base address:	IICA = 0x400A_3000							
Offset address:	0x0000							
Bit position:	7	6	5	4	3	2	1	0
Bit field:								
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
7:0	n/a	8-bit Transmit and Receive Data for IICA of Unit 0	R/W

The IICA0 register is used to convert 8-bit serial data to 8-bit parallel data and vice versa in synchronization with the serial clock. The IICA0 register can be used for both transmission and reception.

The actual transmit and receive operations can be controlled by writing to and reading from the IICA0 register. Release I²C bus interface (IICA) from the clock stretch state and start data transfer by writing data to the IICA0 register during the clock stretch period.

Do not write data to the IICA0 register during data transfer.

Write to or read from the IICA0 register only during the clock stretch period. Accessing the IICA0 register in a communication state other than during the clock stretch period is prohibited. When the device serves as the master mode, however, the IICA0 register can be written only once after the communication trigger bit (IICCTL00.STT) is set to 1.

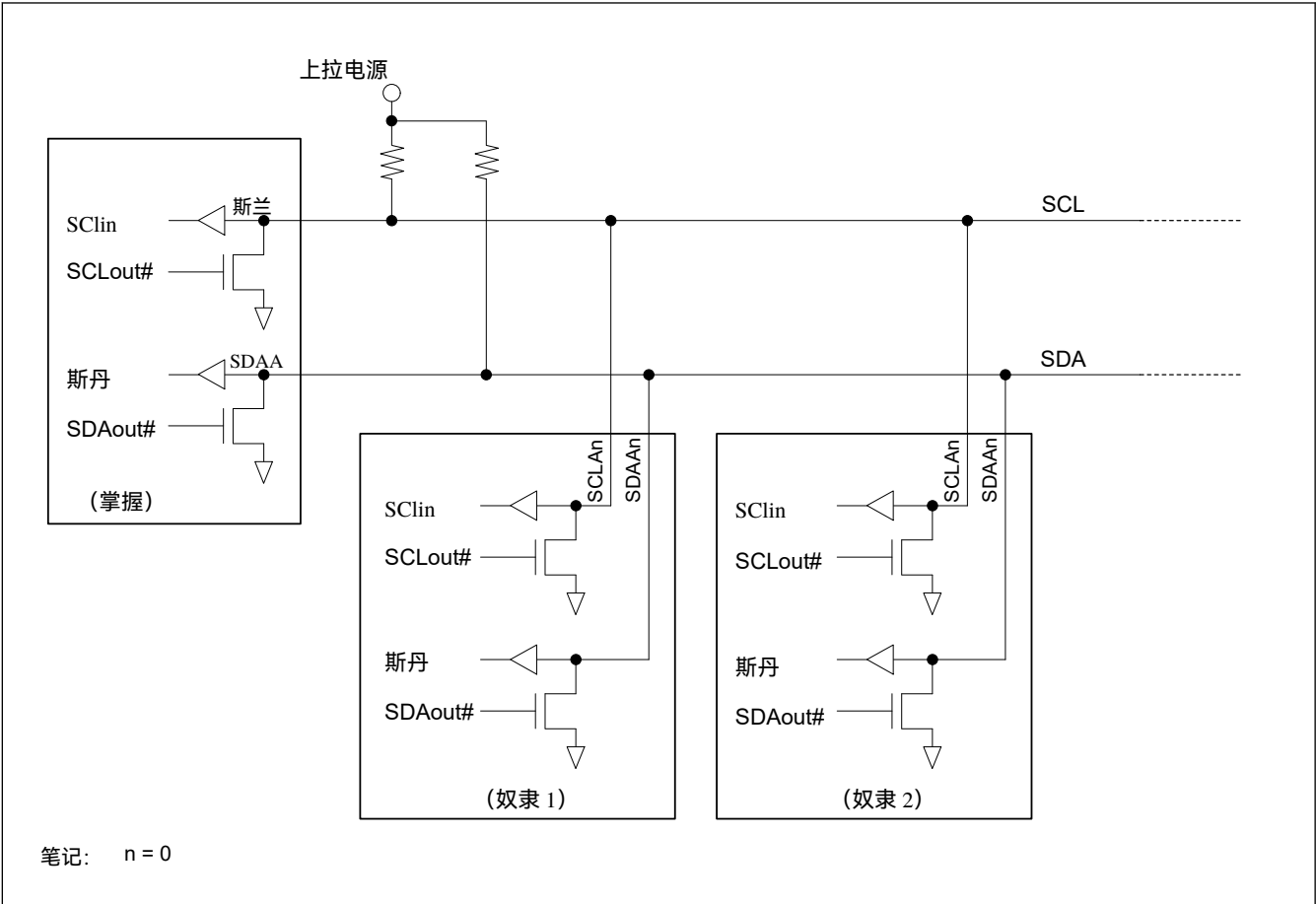


图 20.2 使用I²C总线的串行总线配置示例

20.2 寄存器描述

20.2.1IICA0: IICA 班次登记簿 0

基本地址:	IICA = 0x400A_3000							
偏移地址:	0x0000							
位位置:	7	6	5	4	3	2	1	0
位域:								
重置后的值:	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
7:0	n/a	单元 0 的 IICA 的 8 位发送和接收数据	R/W

IICA0寄存器用于在与串行时钟同步的情况下，将8位串行数据转换为8位并行数据，反之亦然。IICA0寄存器可用于发送和接收。

实际的发送和接收操作可以通过写入和读取 IICA0 寄存器来控制。在时钟拉伸期间，通过向 IICA0 寄存器写入数据，将I²C总线接口 (IICA) 从时钟拉伸状态释放出来，并开始数据传输。

数据传输期间，请勿向 IICA0 寄存器写入数据。

只能在时钟拉伸期间对 IICA0 寄存器进行读写操作。在时钟拉伸期间以外的通信状态下访问 IICA0 寄存器是被禁止的。但是，当设备处于主模式时，在通信触发位 (IICCTL00.STT) 设置为 1 后，只能对 IICA0 寄存器写入一次。

When communication is reserved, write data to the IICA0 register after the interrupt triggered by a stop condition is detected.

20.2.2 SVA0 : Slave Address Register 0

Base address: IICA = 0x400A_3000

Offset address: 0x0104

Bit position:

76543210

Bit field:

A[6:0]

—

Value after reset: 00000000

Bit	Symbol	Function	R/W
0	—	This bit is read as 0. The write value should be 0.	R/W
7:1	A[6:0]	7-bit Local Address when in Slave Mode of Unit 0	R/W

This register holds seven bits (A[6:0]) of the local address when in slave mode.

Rewriting to this register is prohibited while IICS0.STD = 1 (while the start condition is detected).

20.2.3 IICCTL00 : IICA Control Register 00

Base address: IICA = 0x400A_3000

Offset address: 0x0100

Bit position:

76543210

Bit field:

IICE

LREL

WREL

SPIE

WTIM

ACKE

STT

SPT

Value after reset: 00000000

Bit	Symbol	Function	R/W
0	SPT ^{*1}	Stop Condition Trigger 0: Stop condition is not generated 1: Stop condition is generated (End of transfer as master device)	R/W
1	STT ^{*2 *3}	Start Condition Trigger 0: Do not generate a start condition 1: When bus is released (in communication standby status, when IICF0.IICBSY = 0): If this bit is set to 1, a start condition is generated (startup as the master mode). When a third party is communicating: - When communication reservation function is enabled (IICF0.IICRSV = 0) Functions as the start condition reservation flag. When set to 1, automatically generates a start condition after the bus is released. - When communication reservation function is disabled (IICF0.IICRSV = 1) Even if this bit is set to 1, the STT bit is cleared and the STT clear flag (IICF0.STCF) is set to 1. No start condition is generated. In the clock stretch state (for a master device): Generates a restart condition after release from the clock stretch state.	R/W
2	ACKE ^{*4 *5}	Acknowledgment Control 0: Disable acknowledgment 1: Enable acknowledgment. During the 9th clock period, the SDAA0 line is set to low level.	R/W

当通信被保留时，在检测到停止条件触发的中断后，将数据写入 IICA0 寄存器。

20.2.2SVA0：从机地址寄存器 0

基地址：IICA = 0x400A_3000

偏移地址：0x0104

位位置：

76543210

位域：

A[6:0]

—

重置后的值： 00000000

少量	象征	功能	R/W
0	—	该位读取值为 0。写入值应为 0。	R/W
7:1	A[6:0]	单元 0 处于从属模式时的 7 位本地地址	R/W

该寄存器在从模式下保存本地地址的七位（A[6:0]）。

当IICS0.STD = 1（检测到起始条件时）时，禁止对该寄存器进行重写。

20.2.3 IICCTL00: IICA 控制寄存器 00

基地址：IICA= 0x400A_3000

偏移地址：0x0100

位位置：

76543210

位域：

IICE

LREL

WREL

SPIE

WTIM

ACKE

STT

SPT

重置后的值： 00000000

少量	象征	功能	R/W
0	SPT ^{*1}	停止条件触发器 0: 未生成停止条件 1: 生成停止条件（作为主设备传输结束）	R/W
1	STT ^{*2 *3}	启动条件触发器 0: 不生成启动条件 1: 当总线释放时（在通信待机状态下，当IICF0.IICBSY = 0）如果该位设置为 1，则会生成启动条件（以主模式启动）。 当第三方进行沟通时： ●启用通信预留功能后，(IICF0.IICRSV = 0)用作启动条件预留标志。设置为 1 时，总线释放后自动生成启动条件。 ●当通信预留功能被禁用时(IICF0.IICRSV = 1)即使此位设置为1，STT位也会被清除，STT清除标志（IICF0.STCF）也会被设置为1。不会生成启动条件。 在时钟拉伸状态下（对于主设备）： 从时钟拉伸状态释放后，生成重启条件。	R/W
2	ACKE ^{*4 *5}	确认控制 0: 禁用确认 1: 启用确认。在第 9 个时钟周期内，SDAA0 线被设置为低电平。	R/W

Bit	Symbol	Function	R/W
3	WTIM ^{*4}	Control of Clock Stretching and Interrupt Request Generation 0: An interrupt request is generated on the falling edge of the 8th clock cycle. Master mode: After the output of eight clock pulses, the clock output is set to the low level and clock stretching is set. Slave mode: After the input of eight clock pulses, the clock is set to the low level and clock stretching is set for the master device. 1: An interrupt request is generated on the falling edge of the 9th clock cycle. Master mode: After the output of nine clock pulses, the clock output is set to the low level and clock stretching is set. Slave mode: After the input of 9 clock pulses, the clock is set to the low level and clock stretching is set for the master device.	R/W
4	SPIE ^{*6}	Enable and Disable Generation of Interrupt Request when Stop Condition is Detected 0: Disable 1: Enable	R/W
5	WREL ^{*6 *7}	Release from the Clock Stretch State 0: The interface is not released from the clock stretch state. 1: The interface is released from the clock stretch state. After release from the clock stretch state, this bit is automatically cleared to 0.	R/W
6	LREL ^{*6 *7}	Exit from Communications 0: Normal operation 1: IICA exits from the current communications and sets communication standby status. This setting is automatically cleared to 0 after being executed. Its uses include cases in which a locally irrelevant extension code has been received. The SCLA0 and SDAA0 lines are set to high impedance. The following flags of IICA control register 00 (IICCTL00) and the IICA status register 0 (IICS0) are cleared to 0. - IICCTL00.STT - IICCTL00.SPT - IICS0.MSTS - IICS0.EXC - IICS0.COI - IICS0.TRC - IICS0.ACKD - IICS0.STD	R/W
7	IICE	I ² C Operation Enable 0: Stop operation. Reset the IICA status register 0 (IICS0) ^{*8} . Stop internal operation. 1: Enable operation.	R/W

- Note 1. The SPT bit is always read as 0.
- Note 2. The signal of this bit is invalid while IICE is 0.
- Note 3. The STT bit is always read as 0.
- Note 4. The signal of this bit is invalid while IICE is 0. Set this bit during that period.
- Note 5. The set value is invalid during address transfer and if the code is not an extension code, and the all address match function is disabled.
When the device serves as a slave mode and the addresses match, an acknowledgment is generated regardless of the set value.
- Note 6. The setting of this bit has no effect while the setting of IICE is 0.
- Note 7. Reading the LREL and WREL bits always returns 0.
- Note 8. The IICA status register 0 (IICS0), the STCF and IICBSY bits of the IICA flag register 0 (IICF0), and the CLD and DAD bits of IICA control register 01 (IICCTL01) are reset.

This register is used to enable or disable the I²C operations, set the timing of clock stretching, and set other I²C operations.

Note that bits SPIE, WTIM, and ACKE must be set while the setting of IICE is 0 or this module is in the clock stretch state. These bits can be set at the same time as setting the IICE bit 1.

When TRC bit of the IICA status register 0 (IICS0) is set to 1 (transmission state), WREL bit of IICA control register 00 (IICCTL00) is set to 1 during the 9th clock and the interface is released from the clock stretch state, after which the IICS0.TRC bit is cleared (reception state) and the SDAA0 line is set to the high impedance state. Release the interface from the clock stretch state while the IICS0.TRC bit is 1 (transmission state) by writing to the IICA shift register 0 (IICA0).

If the I²C operation is enabled (IICE = 1) when the SCLA0 line is high level, the SDAA0 line is low level, and the digital filter is turned on (DFC bit of IICCTL01 register = 1), a start condition is inadvertently detected immediately. In this case, set 1 to the LREL bit after enabling the I²C operation (IICE = 1).

少量	象征	功能	R/W
3	WTIM ^{*4}	时钟拉伸和中断请求生成的控制 0: 在第 8 个时钟周期的下降沿生成中断请求。 主模式: 输出八个时钟脉冲后, 时钟输出设置为低电平, 并设置时钟拉伸。 从模式: 输入八个时钟脉冲后, 时钟设置为低电平, 并对主设备设置时钟拉伸。 1: 在第 9 个时钟周期的下降沿生成中断请求。 主模式: 输出九个时钟脉冲后, 时钟输出设置为低电平, 并设置时钟拉伸。 从机模式: 输入 9 个时钟脉冲后, 时钟设置为低电平, 并对主设备进行时钟拉伸。	R/W
4	SPIE ^{*6}	启用和禁用检测到停止条件时生成中断请求 0: 禁用 1: 启用	R/W
5	WREL ^{*6 *7}	从时钟拉伸状态释放 0: 接口未从时钟拉伸状态释放。 1: 接口已从时钟拉伸状态释放。从时钟拉伸状态释放后, 此位会自动清零。	R/W
6	LREL ^{*6 *7}	退出通信 0: 正常操作 1: IICA 退出当前通信并将通信状态设置为待机状态。此设置执行后会自动清除为 0。 它的用途包括收到与本地无关的扩展代码的情况。 SCLA0 和 SDAA0 线路设置为高阻抗。 IICA 控制寄存器 00 (IICCTL00) 和 IICA 状态寄存器 0 (IICS0) 的以下标志被清除为 0。 - IICCTL00.STT - IICCTL00.SPT - IICS0.MSTS - IICS0.EXC - IICS0.COI - IICS0.TRC - IICS0.ACKD - IICS0.STD	R/W
7	IICE	I ² C 操作使能 0: 停止操作。重置 IICA 状态寄存器 0(IICS0) ^{*8} . 停止内部操作。 1: 启用操作。	R/W

- 注1. SPT 位始终读取为 0。
- 注2. 当IICE为0时, 此位的信号无效。
- 注3. STT 位始终读取为 0。
- 注 4. 当 IICE 为 0 时, 此位的信号无效。在此期间设置此位。
- 注 5. 在地址转移期间, 如果代码不是分机号, 则设置值无效, 并且所有地址匹配功能将被禁用。
- 当设备处于从属模式且地址匹配时, 无论设置值如何, 都会生成确认信息。
- 注 6. 当 IICE 设置为 0 时, 此位的设置无效。
- 注 7. 读取 LREL 和 WREL 位总是返回 0。
- 注 8. IICA 状态寄存器 0 (IICS0)、IICA 标志寄存器 0 (IICF0) 的 STCF 和 IICBSY 位以及 IICA 控制寄存器 01 (IICCTL01) 的 CLD 和 DAD 位被复位。

该寄存器用于启用或禁用I²C操作, 设置时钟拉伸的时序, 以及设置其他I²C操作。

请注意, 当 IICE 设置为 0 或该模块处于时钟拉伸状态时, 必须设置 SPIE、WTIM 和 ACKE 位。这些位可以与设置 IICE 位 1 同时设置。

当 IICA 状态寄存器 0 (IICS0) 的 TRC 位设置为 1 (发送状态) 时, IICA 控制寄存器 00 (IICCTL00) 的 WREL 位在第 9 个时钟周期被设置为 1, 接口从时钟拉伸状态释放, 之后 IICS0.TRC 位被清除 (接收状态), SDAA0 线被设置为高阻抗状态。当 IICS0.TRC 位为 1 (发送状态) 时, 通过写入 IICA 移位寄存器 0 (IICA0) 来释放接口的时钟拉伸状态。

如果当 SCLA0 线为高电平、SDAA0 线为低电平且数字滤波器开启 (IICCTL01 寄存器的 DFC 位为= 1) 时, 启用I²C操作, 则会立即意外检测到启动条件。在这种情况下, 在启用I²C操作后, 将 LREL 位设置为 1。

SPT bit (Stop Condition Trigger)

Cautions concerning set timing

- For master reception: Cannot be set to 1 during transfer.
After setting the ACKE bit to 0, telling the slave device that it is the last to receive, this bit can be set to 1 only during the clock stretch period.
- For master transmission: During the period of Acknowledge, it may not be possible to generate stop conditions correctly.
Therefore, set it during the clock stretch period that follows output of the ninth clock.
- Cannot be set to 1 at the same time as start condition trigger (STT).
- The SPT bit can be set to 1 only when in master mode.
- When the WTIM bit has been cleared to 0, if the SPT bit is set to 1 during the clock stretch period that follows output of eight clock pulses, note that a stop condition will be generated during the high-level period of the ninth clock after release from the clock stretch state. The WTIM bit should be changed from 0 to 1 during the clock stretch period following the output of eight clock pulses, and the SPT bit should be set to 1 during the clock stretch period that follows the output of the ninth clock.
- Once SPT is set to 1, setting it to 1 again before the clear condition is met is not allowed.

Condition for clearing (SPT = 0)

- Cleared by loss in arbitration
- Automatically cleared after stop condition is detected
- Cleared by LREL = 1 (exit from communications)
- When IICE = 0 (operation stop)
- Reset

Condition for setting (SPT = 1)

- Set by instruction

Note: The read value of the SPT bit is always 0.

STT bit (Start Condition Trigger)

Cautions concerning set timing

- For master reception: Cannot be set to 1 during transfer.
After setting the ACKE bit to 0, telling the slave device that it is the last to receive, this bit can be set to 1 only during the clock stretch period.
- For master transmission: During the period of Acknowledge, it may not be possible to generate start conditions correctly.
Set to 1 during the clock stretch period that follows output of the ninth clock.
- Cannot be set to 1 at the same time as stop condition trigger (SPT).
- Once STT is set to 1, setting it to 1 again before the clear condition is met is not allowed.

Condition for clearing (STT = 0)

- Cleared by setting the STT bit to 1 while communication reservation is prohibited.
- Cleared by loss in arbitration
- Cleared after start condition is generated by master device
- Cleared by LREL = 1 (exit from communications)
- When IICE = 0 (operation stop)
- Reset

Condition for setting (STT = 1)

SPT 位（停止条件触发）

关于设定时间的注意事项

- 对于主接收：传输过程中不能设置为 1。
将 ACKE 位设置为 0 后，告诉从设备它是最后一个接收的设备，该位只能在时钟拉伸期间设置为 1。
- 对于主传输：在确认期间，可能无法正确生成停止条件。
因此，在第九个时钟输出之后的时钟拉伸期间设置它。
- 不能与启动条件触发器（STT）同时设置为 1。
- 只有在主模式下，SPT 位才能设置为 1。
- 当 WTIM 位被清除为 0 时，如果在输出八个时钟脉冲后的时钟拉伸期间将 SPT 位置为 1，则在解除时钟拉伸状态后的第九个时钟的高电平期间会产生停止条件。WTIM 位应在输出八个时钟脉冲后的时钟拉伸期间从 0 变为 1，SPT 位应在输出第九个时钟脉冲后的时钟拉伸期间置为 1。
- 一旦 SPT 设置为 1，在满足清除条件之前，不允许再次将其设置为 1。

清除条件(SPT = 0)

- 仲裁败诉后获得赔偿
- 检测到停止条件后自动清除
- 已由LREL = 1清除（退出通信）
- 当IICE = 0（操作停止）
- 重置

设置(SPT = 1的条件)

- 按说明设置

注意：SPT 位的读取值始终为 0。

STT 位（起始条件触发）

关于设定时间的注意事项

- 对于主接收：传输过程中不能设置为 1。
将 ACKE 位设置为 0 后，告诉从设备它是最后一个接收的设备，该位只能在时钟拉伸期间设置为 1。
- 对于主传输：在确认期间，可能无法正确生成启动条件。
在第九个时钟输出后的时钟拉伸期间，设置为 1。
- 不能与停止条件触发器（SPT）同时设置为 1。
- 一旦 STT 设置为 1，在满足清除条件之前，不允许再次将其设置为 1。

清除条件(STT = 0)

- 通过将 STT 位设置为 1 来清除，同时禁止通信预留。
 - 仲裁败诉后免于处罚
- 主设备生成启动条件后清除
- 已通过 LREL= 1（退出通信）
- 当IICE = 0（操作停止）
- 重置

设置(STT = 1的条件)

- Set by instruction

ACKE bit (Acknowledgment Control)

Condition for clearing (ACKE = 0)

- Cleared by instruction
- Reset

Condition for setting (ACKE = 1)

- Set by instruction

WTIM bit (Control of Clock Stretching and Interrupt Request Generation)

An interrupt is generated on the falling edge of the ninth clock cycle during address transfer independently of the setting of this bit. The setting of this bit is valid when the address transfer is completed. In master mode, clock stretching is inserted at the falling edge of the ninth clock cycle during address transfer. In slave mode, when a local address is received, clock stretching is inserted at the falling edge of the 9th clock cycle after an acknowledge (ACK) is issued. However, in slave mode, when an extension code is received, clock stretching is inserted at the falling edge of the 8th clock cycle. When an address is received while the all address match function is enabled, clock stretching is inserted at the falling edge of the eighth clock cycle.

Condition for clearing (WTIM = 0)

- Cleared by instruction
- Reset

Condition for setting (WTIM = 1)

- Set by instruction

SPIE bit (Enable and Disable Generation of Interrupt Request when Stop Condition is Detected)

If the WUP bit of IICA control register 01 (IICCTL01) is 1, no stop condition interrupt will be generated even if SPIE = 1.

Condition for clearing (SPIE = 0)

- Cleared by instruction
- Reset

Condition for setting (SPIE = 1)

- Set by instruction

WREL bit (Release from the Clock Stretch State)

When the WREL bit is set (for release from the clock stretch state) during the clock stretch period at the ninth clock pulse in the transmission state (IICS0.TRC = 1), the SDAA0 line goes into the high impedance state (IICS0.TRC = 0).

Condition for clearing (WREL = 0)

- Automatically cleared after execution
- Reset

Condition for setting (WREL = 1)

- Set by instruction

LREL bit (Exit from Communications)

The communication standby status following exit from communications remains in effect until the following communications entry conditions are met.

- After a stop condition is detected, restart is in master mode.
- An address match, extension code reception, or address reception with the all address match function enabled occurs after the start condition.

- 按说明设置

ACKE 位（确认控制）

清除条件(ACKE = 0)

- 已按指示清除
- 重置

设置(ACKE = 1的条件)

- 按指令设置

WTIM 位（时钟拉伸和中断请求生成的控制）

在地址传输期间，无论该位的设置如何，都会在第九个时钟周期的下降沿产生中断。该位的设置仅在地址传输完成后有效。在主模式下，地址传输期间会在第九个时钟周期的下降沿插入时钟拉伸。在从模式下，当接收到本地地址时，在发出确认 (ACK) 后，会在第九个时钟周期的下降沿插入时钟拉伸。但是，在从模式下，当接收到扩展码时，会在第八个时钟周期的下降沿插入时钟拉伸。当启用全地址匹配功能时接收到地址，会在第八个时钟周期的下降沿插入时钟拉伸。

清除条件(WTIM = 0)

- 已按指示清除
- 重置

设置(WTIM = 1的条件)

- 按说明设置

SPIE 位（检测到停止条件时启用和禁用中断请求的生成）

如果 IICA 控制寄存器 01 (IICCTL01) 的 WUP 位为 1，即使SPIE = 1也不会产生停止条件中断。

清除条件(SPIE = 0)

- 已按指示清除
- 重置

设置(SPIE = 1的条件)

- 按说明设置

WREL 位（从时钟拉伸状态释放）

当在传输状态(IICS0.TRC = 1)的第九个时钟脉冲期间的时钟拉伸周期中，WREL 位被置位（用于从时钟拉伸状态释放）时，SDAA0 线进入高阻抗状态(IICS0.TRC = 0)。

清除条件(WREL = 0)

- 执行完毕后自动清除
- 重置

设置条件 (WREL= 1)

- 按指令设置

LREL 位（退出通信）

退出通信后，通信待机状态将持续有效，直到满足以下通信进入条件。

- 检测到停止条件后，重新启动以主模式进行。
- 在启动条件之后，会发生地址匹配、分机号接收或启用所有地址匹配功能的地址接收。

Condition for clearing (LREL = 0)

- Automatically cleared after execution
- Reset

Condition for setting (LREL = 1)

- Set by instruction

IICE bit (I²C Operation Enable)

Be sure to set this bit to 1 while the SCLA0 and SDAA0 lines are at high level.

Condition for clearing IICE = 0)

- Cleared by instruction
- Reset

Condition for setting (IICE = 1)

- Set by instruction

20.2.4 IICS0 : IICA Status Register 0

Base address: IICA = 0x400A_3000

Offset address: 0x0001

Bit position:	7	6	5	4	3	2	1	0
Bit field:	MSTS	ALD	EXC	COI	TRC	ACKD	STD	SPD
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
0	SPD	Detection of Stop Condition 0: Stop condition was not detected. 1: Stop condition was detected. Communication of the master device is terminated and the bus is released.	R
1	STD	Detection of Start Condition 0: Start condition was not detected. 1: Start condition was detected. This indicates that the address transfer period is in effect.	R
2	ACKD	Detection of Acknowledge (ACK) 0: Acknowledge was not detected. 1: Acknowledge was detected.	R
3	TRC	Detection of Transmit and Receive Status 0: Receive status (other than transmit status). The SDAA0 line is set for high impedance. 1: Transmit status. The value in the SOn latch is enabled for output to the SDAA0 line (valid starting at the falling edge of the first byte's ninth clock).	R
4	COI	Detection of Matching Addresses 0: Addresses do not match. 1: Addresses match. Or, the all address match function is enabled.	R
5	EXC	Detection of Extension Code Reception 0: Extension code was not received. 1: Extension code was received. Or, the all address match function is enabled.	R
6	ALD	Detection of Arbitration Loss 0: This status means either that there was no arbitration, or that the arbitration result was a win. 1: This status indicates the arbitration result was a loss. The MSTS bit is cleared.	R

清除条件(LREL = 0)

- 执行完毕后自动清除
- 重置

设置(LREL = 1)的条件

- 按说明设置

IICE 位(I²C操作使能)

当 SCLA0 和 SDAA0 线处于高电平时，务必将此位设置为 1。

清除条件IICE = 0)

- 已按指示清除
- 重置

设置(IICE = 1)的条件

- 按说明设置

20.2.4 IICS0: IICA 状态寄存器 0

基地址: IICA = 0x400A_3000

偏移地址: 0x0001

位位置:	7	6	5	4	3	2	1	0
位域:	MSTS	ALD	EXC	COI	TRC	ACKD	STD	SPD
重置后的值:	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
0	SPD	停止条件 0 检测：未检测到停止条件。 1：检测到停止状态。主设备通信终止，总线释放。	R
1	STD	检测启动条件 0：未检测到启动条件。 1：检测到启动条件。这表明地址转移期已生效。	R
2	ACKD	确认 (ACK) 检测结果为 0：未检测到确认。 1：检测到确认。	R
3	TRC	检测发送和接收状态 0：接收状态（非发送状态）。SDAA0 线设置为高阻抗。 1：发送状态。SOn 锁存器中的值被启用，可输出到 SDAA0 线（从第一个字节的第九个时钟的下降沿开始有效）。	R
4	COI	匹配地址检测结果 0：地址不匹配。 1：地址匹配。或者，已启用所有地址匹配功能。	R
5	EXC	分机号接收检测 0：未收到分机号。 1：已收到分机号。或者，已启用全地址匹配功能。	R
6	ALD	仲裁损失的检测 0：此状态表示要么没有进行仲裁，要么仲裁结果为胜诉。 1：此状态表示仲裁结果为败诉。MSTS 位已清除。	R

Bit	Symbol	Function	R/W
7	MSTS	Master Status Check Flag 0: Slave mode status or communication standby status 1: Master mode communication status	R

This register indicates the state of the I²C.

The IICS0 register can only be read while the setting of IICCTL00.STT is 1 or this module is in the clock stretch state.

Reading the IICS0 register while the address match wakeup function is enabled (IICCTL01.WUP = 1) in Software Standby mode is prohibited. When the IICCTL01.WUP bit is changed from 1 to 0 (wakeup operation is stopped), regardless of the IICA0_TXRXI interrupt request signal, the change in status is not reflected until the next start condition or stop condition is detected. To use the wakeup function, enable (SPIE = 1) the interrupt generated by detecting a stop condition and read the IICS0 register after the interrupt has been detected.

SPD bit (Detection of Stop Condition)

Condition for clearing (SPD = 0)

- At the rising edge of the address transfer byte's first clock following setting of this bit and detection of a start condition
- When the IICCTL01.WUP bit changes from 1 to 0
- When the IICCTL00.IICE bit changes from 1 to 0 (operation stop)
- Reset

Condition for setting (SPD = 1)

- When a stop condition is detected

STD bit (Detection of Start Condition)

Condition for clearing (STD = 0)

- When a stop condition is detected
- At the rising edge of the next byte's first clock following address transfer
- Cleared by IICCTL00.LREL = 1 (exit from communications)
- When the IICCTL00.IICE bit changes from 1 to 0 (operation stop)
- Reset

Condition for setting (STD = 1)

- When a start condition is detected

ACKD bit (Detection of Acknowledge (ACK))

Condition for clearing (ACKD = 0)

- When a stop condition is detected
- At the rising edge of the next byte's first clock
- Cleared by IICCTL00.LREL = 1 (exit from communications)
- When the IICCTL00.IICE bit changes from 1 to 0 (operation stop)
- Reset

Condition for setting (ACKD = 1)

- After the SDAA0 line is set to low level at the rising edge of SCLA0 line's ninth clock

TRC bit (Detection of Transmit and Receive Status)

Condition for clearing (TRC = 0)

<Both master mode and slave mode>

- When a stop condition is detected

少量	象征	功能	R/W
7	MSTS	主状态检查标志 0: 从模式状态或通信待机状态 1: 主模式通信状态	R

该寄存器指示I²C的状态。

只有当 IICCTL00.STT 设置为 1 或该模块处于时钟拉伸状态时，才能读取 IICS0 寄存器。

在软件待机模式下，当地址匹配唤醒功能启用时（(IICCTL01.WUP = 1)），禁止读取 IICS0 寄存器。当 IICCTL01.WUP 位从 1 变为 0（唤醒操作停止）时，无论 IICA0_TXRXI 中断请求信号如何，状态变化都不会立即生效，直到检测到下一个启动条件或停止条件。要使用唤醒功能，请启用（(SPIE = 1) 由检测到停止条件产生的中断，并在检测到中断后读取 IICS0 寄存器。

SPD 位（停止条件检测）

清除条件(SPD = 0)

- 在地址传输字节的第一个时钟的上升沿，在该位设置之后并检测到起始条件时
- 当 IICCTL01.WUP 位从 1 变为 0 时
- 当 IICCTL00.IICE 位从 1 变为 0 时（操作停止）
- 重置

设置(SPD = 1)的条件

- 检测到停止条件时

STD 位（起始条件检测）

清除条件(STD = 0)

- 检测到停止条件时
- 在地址传输之后，下一个字节的第一个时钟的上升沿。
- 已由IICCTL00.LREL = 1清除（退出通信）
- 当 IICCTL00.IICE 位从 1 变为 0 时（操作停止）
- 重置

设置(STD = 1)的条件

- 当检测到启动条件时

ACKD 位（确认检测 (ACK)

清除条件(ACKD = 0)

- 检测到停止条件时
- 在下一个字节第一个时钟的上升沿
- 已由 IICCTL00.LREL= 1清除（退出通信）
- 当 IICCTL00.IICE 位从 1 变为 0 时（操作停止）
- 重置

设置(ACKD = 1)的条件

- 在 SCLA0 线第九个时钟周期上升沿，SDAA0 线被设置为低电平后

TRC 位（发送和接收状态检测）

清除条件(TRC = 0)

<主模式和从模式>

- 检测到停止条件时

- Cleared by IICCTL00.LREL = 1 (exit from communications)
- When the IICCTL00.IICE bit changes from 1 to 0 (operation stop)
- Cleared by IICCTL00.WREL = 1 (release from the clock stretch state)*1
- When the ALD bit changes from 0 to 1 (arbitration loss)
- Reset
- Not participating in communication (MSTS, EXC, COI = 0)

<Master mode>

- When 1 is output to the LSB of the first byte (transfer direction specification bit)

<Slave mode>

- When a start condition is detected
- When 0 is input to the LSB of the first byte (transfer direction specification bit)

Condition for setting (TRC = 1)

<Master mode>

- When a start condition is generated
- When 0 (master transmission) is output to the LSB (transfer direction specification bit) of the first byte (during address transfer)

<Slave mode>

- When 1 (slave transmission) is input to the LSB (transfer direction specification bit) of the first byte from the master mode (during address transfer)

Note 1. When TRC bit of the IICA status register 0 (IICS0) is set to 1 (transmission state), WREL bit of IICA control register 00 (IICCTL00) is set to 1 during the 9th clock and the interface is released from the clock stretch state, after which the IICS0.TRC bit is cleared (reception state) and the SDAA0 line is set to the high impedance state. Release the interface from the clock stretch state while the IICS0.TRC bit is 1 (transmission state) by writing to the IICA shift register 0.

COI bit (Detection of Matching Addresses)

Condition for clearing (COI = 0)

- When a start condition is detected
- When a stop condition is detected
- Cleared by IICCTL00.LREL = 1 (exit from communications)
- When the IICCTL00.IICE bit changes from 1 to 0 (operation stop)
- Reset

Condition for setting (COI = 1)

- When the received address matches the local address (slave address register 0 (SVA0)) (set at the rising edge of the eighth clock).
- When an address is received while the all address match function is enabled (IICCTL01.SVADIS = 1) (set at the rising edge of the eighth clock).

EXC bit (Detection of Extension Code Reception)

Condition for clearing (EXC = 0)

- When a start condition is detected
- When a stop condition is detected
- Cleared by IICCTL00.LREL = 1 (exit from communications)
- When the IICCTL00.IICE bit changes from 1 to 0 (operation stop)

- 已由 IICCTL00.LREL= 1清除（退出通信）
- 当 IICCTL00.IICE 位从 1 变为 0 时（操作停止）
- 由IICCTL00.WREL = 1清除（从时钟拉伸状态释放）*1
- 当 ALD 位从 0 变为 1 时（仲裁丢失）
- 重置
- 不参与沟通（MSTS、EXC、COI = 0）

<大师模式>

- 当第一个字节的最低有效位（传输方向指定位）输出为 1 时

<从模式>

- 当检测到启动条件时
- 当第一个字节的最低有效位（传输方向指定位）输入为0时

设置(TRC = 1的条件)

<大师模式>

- 当生成起始条件时
- 当 0（主传输）输出到第一个字节的最低有效位（传输方向指定位）（在地址传输期间）

<从模式>

- 当从主机模式（地址传输期间）的第一个字节的最低有效位（传输方向指定位）输入为 1（从机传输）时

注1: 当IICA状态寄存器0（IICS0）的TRC位设置为1（发送状态）时，在第9个时钟周期，IICA控制寄存器00（IICCTL00）的WREL位设置为1，接口从时钟拉伸状态释放，之后IICS0.TRC位被清除（接收状态），SDAA0线被设置为高阻抗状态。当IICS0.TRC位为1（发送状态）时，通过写入IICA移位寄存器0来释放接口的时钟拉伸状态。

COI 位（匹配地址检测）

清除条件(COI = 0)

- 当检测到启动条件时
- 检测到停止条件时
- 已由 IICCTL00.LREL= 1清除（退出通信）
- 当 IICCTL00.IICE 位从 1 变为 0 时（操作停止）
- 重置

设置(COI = 1的条件)

- 当接收到的地址与本地地址（从地址寄存器 0 (SVA0)）匹配时（在第八个时钟的上升沿设置）。
- 当所有地址匹配功能启用时（IICCTL01.SVADIS= 1）（在第八个时钟的上升沿设置）收到地址。

EXC 位（检测分机号接收）

清除条件(EXC = 0)

- 当检测到启动条件时
- 当检测到停止条件时
- 已通过IICCTL00.LREL = 1清除（退出通信）
- 当 IICCTL00.IICE 位从 1 变为 0 时（操作停止）

- Reset

Condition for setting (EXC = 1)

- When the higher four bits of the received address data is either 0000b or 1111b (set at the rising edge of the eighth clock).
- When an address is received while the all address match function is enabled (IICCTL01.SVADIS = 1) (set at the rising edge of the eighth clock).

ALD bit (Detection of Arbitration Loss)

Condition for clearing (ALD = 0)

- Automatically cleared after the IICS0 register is read
- When the IICCTL00.IICE bit changes from 1 to 0 (operation stop)
- Reset

Condition for setting (ALD = 1)

- Loss in arbitration

MSTS flag (Master Status Check Flag)

Condition for clearing (MSTS = 0)

- When a stop condition is detected
- When ALD = 1 (arbitration loss)
- Cleared by IICCTL00.LREL = 1 (exit from communications)
- When the IICCTL00.IICE bit changes from 1 to 0 (operation stop)
- Reset

Condition for setting (MSTS = 1)

- When a start condition is generated

20.2.5 IICF0 : IICA Flag Register 0

Base address: IICA = 0x400A_3000							
Offset address: 0x0002							
Bit position:	7	6	5	4	3	2	1 0
Bit field:	STCF	IICBS Y	—	—	—	—	STCE N IICRS V
Value after reset:	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
0	IICRSV	Communication Reservation Function Disable Bit 0: Enable communication reservation 1: Disable communication reservation	R/W
1	STCEN	Initial Start Enable Trigger 0: After operation is enabled (IICCTL00.IICE = 1), enable generation of a start condition upon detection of a stop condition. 1: After operation is enabled (IICCTL00.IICE = 1), enable generation of a start condition without detecting a stop condition.	R/W
5:2	—	These bits are read as 0. The write value should be 0.	R/W
6	IICBSY	I ² C Bus Status Flag 0: Bus release status (communication initial status when STCEN = 1) 1: Bus communication status (communication initial status when STCEN = 0)	R

- 重置

设置(EXC = 1的条件)

- 当接收到的地址数据的高四位为0000b或1111b（在第八个时钟的上升沿设置）时。
- 当所有地址匹配功能启用时（IICCTL01.SVADIS= 1）（在第八个时钟的上升沿设置）收到地址。

ALD 位（仲裁损失检测）

清除条件(ALD = 0)

- 读取IICS0寄存器后自动清除
- 当 IICCTL00.IICE 位从 1 变为 0 时（操作停止）
- 重置

设置(ALD = 1的条件)

- 仲裁败诉

MSTS标志（主状态检查标志）

清除条件(MSTS = 0)

- 检测到停止条件时
- 当ALD = 1（仲裁损失）
- 已由 IICCTL00.LREL= 1清除（退出通信）
- 当 IICCTL00.IICE 位从 1 变为 0 时（操作停止）
- 重置

设置(MSTS = 1的条件)

- 当生成起始条件时

20.2.5IICF0： IICA 标志寄存器 0

基地址： IICA= 0x400A_3000							
偏移地址： 0x0002							
位位置：	7	6	5	4	3	2	1 0
位域：	STCF	IICBS Y	—	—	—	—	STCE N IICRS V
重置后的值：	0	0	0	0	0	0	0

少量	象征	功能	R/W
0	IICRSV	通信预留功能禁用位 0: 启用通信预留 1: 禁用通信预留	R/W
1	STCEN	初始启动启用触发器 0: 操作启用后(IICCTL00.IICE = 1)，在检测到停止条件时启用生成启动条件。 1: 操作启用后(IICCTL00.IICE = 1)，无需检测停止条件即可生成启动条件。	R/W
5:2	—	这些位读取为 0。写入值应为 0。	R/W
6	IICBSY	I ² C总线状态标志 0: 总线释放状态（通信初始状态，当STCEN = 1） 1: 总线通信状态（当 STCEN= 0时，通信初始状态）	R

Bit	Symbol	Function	R/W
7	STCF	IICCTL00.STT Clear Flag 0: Generate start condition 1: Start condition generation unsuccessful: clear the IICCTL00.STT flag	R

This register sets the operation mode of I²C and indicates the state of the I²C bus.

The IICCTL00.STT clear flag (STCF) and I²C bus status flag (IICBSY) bits are read-only.

The IICRSV bit can be used to enable or disable the communication reservation.

The STCEN bit can be used to set the initial value of the IICBSY bit.

The IICRSV and STCEN bits can be written only when the I²C operation is disabled (IICCTL00.IICE=0). The IICF0 register is read-only while the I²C operation is enabled.

Write to the STCEN bit only when the operation is stopped (IICCTL00.IICE = 0).

The bus release status (IICBSY = 0) is recognized regardless of the actual bus status when STCEN = 1. When generating the first start condition (IICCTL00.STT = 1), it is necessary to verify that no third-party communications are in progress in order to prevent such communications from being destroyed.

Write to the IICRSV bit only when the operation is stopped (IICCTL00.IICE = 0).

IICRSV bit (Communication Reservation Function Disable Bit)

Condition for clearing (IICRSV = 0)

- Cleared by instruction
- Reset

Condition for setting (IICRSV = 1)

- Set by instruction

STCEN bit (Initial Start Enable Trigger)

Condition for clearing (STCEN = 0)

- Cleared by instruction
- When a start condition is detected
- Reset

Condition for setting (STCEN = 1)

- Set by instruction

IICBSY flag (I²C Bus Status Flag)

Condition for clearing (IICBSY = 0)

- When a stop condition is detected
- When IICCTL00.IICE = 0 (operation stop)
- Reset

Condition for setting (IICBSY = 1)

- When a start condition is detected
- Setting of the IICCTL00.IICE bit when STCEN = 0

STCF flag (IICCTL00.STT Clear Flag)

Condition for clearing (STCF = 0)

- Cleared by IICCTL00.STT = 1
- When IICCTL00.IICE = 0 (operation stop)

少量	象征	功能	R/W
7	STCF	IICCTL00.STT 清除标志 0: 生成起始条件 1: 启动条件生成失败: 清除 IICCTL00.STT 标志	R

该寄存器设置I²C的操作模式，并指示I²C总线的状态。

IICCTL00.STT 清除标志 (STCF) 和I²C总线状态标志 (IICBSY) 位是只读的。

IICRSV 位可用于启用或禁用通信预留。

STCEN 位可用于设置 IICBSY 位的初始值。

只有在禁用I²C操作时，才能写入 IICRSV 和 STCEN 位（IICCTL00.IICE=0）。当启用I²C 操作时，IICF0 寄存器是只读的。

仅当操作停止时才写入 STCEN 位(IICCTL00.IICE = 0)。

当STCEN = 1时，无论实际总线状态如何，都会识别总线释放状态(IICBSY = 0)。生成第一个启动条件(IICCTL00.STT = 1)时，必须验证没有第三方通信正在进行，以防止此类通信被破坏。

仅当操作停止时才写入 IICRSV 位(IICCTL00.IICE = 0)。

IICRSV 位（通信预留功能禁用位）

清除条件(IICRSV = 0)

- 已按指示清除
- 重置

设置(IICRSV = 1的条件)

- 按说明设置

STCEN 位（初始启动使能触发）

清除条件(STCEN = 0)

- 已按指示清除
- 当检测到启动条件时
- 重置

设置(STCEN = 1的条件

- 按指令设置

IICBSY 标志(I²C总线状态标志)

清除条件(IICBSY = 0)

- 当检测到停止条件时
- 当 IICCTL00.IICE = 0 （操作停止）
- 重置

设置(IICBSY = 1的条件)

- 当检测到启动条件时
- 当STCEN = 0时设置 IICCTL00.IICE 位

STCF 标志（IICCTL00.STT 清除标志）

清除条件(STCF = 0)

- 已由IICCTL00.STT = 1审核通过
- 当 IICCTL00.IICE = 0 （操作停止）

- Reset

Condition for setting (STCF = 1)

- Generating start condition unsuccessful and the IICCTL00.STT bit cleared to 0 when communication reservation is disabled (IICRSV = 1).

20.2.6 IICCTL01 : IICA Control Register 01

Base address: IICA = 0x400A_3000								
Offset address: 0x0101								
Bit position:	7	6	5	4	3	2	1	0
Bit field:	WUP	SVADIS	CLD	DAD	SMC	DFC	—	PRS
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
0	PRS	IICA Operation Clock (f _{MCK}) 0: Selects PCLKB (1 MHz ≤ PCLKB ≤ 20 MHz) 1: Selects PCLKB/2 (20 MHz < PCLKB)	R/W
1	—	This bit is read as 0. The write value should be 0.	R/W
2	DFC	Digital Filter Operation Control 0: Digital filter off 1: Digital filter on	R/W
3	SMC	Operation Mode Switching 0: Operates in standard mode (fastest transfer rate: 100 kbps) 1: Operates in fast mode (fastest transfer rate: 400 kbps) or fast mode plus (fastest transfer rate: 1 Mbps)	R/W
4	DAD	Detection of SDAA0 Pin Level (Valid Only when IICCTL00.IICE = 1) 0: The SDAA0 pin was detected at low level 1: The SDAA0 pin was detected at high level	R
5	CLD	Detection of SCLA0 Pin Level (Valid Only when IICCTL00.IICE = 1) 0: The SCLA0 pin was detected at low level 1: The SCLA0 pin was detected at high level	R
6	SVADIS	Address Match Disabling Flag 0: Disables the all address match function 1: Enables the all address match function	R/W
7	WUP	Control of Address Match Wakeup 0: Stops operation of address match wakeup function in Software Standby mode 1: Enables operation of address match wakeup function in Software Standby mode	R/W

This register is used to set the operation mode of I²C and detect the states of the SCLA0 and SDAA0 pins.

The CLD and DAD bits are read-only.

Set the IICCTL01 register, except the WUP bit, while I²C operation is disabled (IICCTL00.IICE).

The fastest operation frequency of the IICA operation clock (f_{MCK}) is 20 MHz (max.).

Set PRS bit of the IICA control register 01 (IICCTL01) to 1 only when the PCLKB exceeds 20 MHz.

Note the minimum PCLKB operation frequency when setting the transfer clock.

The minimum PCLKB operation frequency for I²C bus interface (IICA) is determined according to the mode.

Fast mode: PCLKB = 3.5 MHz (min.)

Fast mode plus: PCLKB = 10 MHz (min.)

Normal mode: PCLKB = 1 MHz (min.)

- 重置

设置(STCF = 1的条件)

- 生成启动条件失败，当通信预留被禁用时，IICCTL00.STT 位被清除为 0(IICRSV = 1)

20.2.6IICCTL01： IICA 控制寄存器 01

基地址: IICA = 0x400A_3000

偏移地址: 0x0101

位位置:	7	6	5	4	3	2	1	0
位域:	WUP	SVADIS	CLD	DAD	SMC	DFC	—	PRS
重置后的值:	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
0	PRS	IICA 运行时钟 (fMCK) 0: 选择 PCLKB (1 MHz≤ PCLKB≤ 20 MHz) 1: 选择 PCLKB/2 (20 MHz< PCLKB)	R/W
1	—	该位读取值为 0。写入值应为 0。	R/W
2	DFC	数字滤波器运行控制 0: 数字滤波器关闭 1: 数字滤波器开启	R/W
3	SMC	操作模式切换 0: 以标准模式运行（最快传输速率: 100 kbps): 以快速模式运行（最快传输速率: 400 kbps) 或快速增强模式运行（最快传输速率: 1 Mbps)	R/W
4	DAD	检测 SDAA0 引脚电平（仅当 IICCTL00.IICE= 1时有效) 0: 检测到 SDAA0 引脚为低电平 1: 检测到 SDAA0 引脚为高电平	R
5	CLD	检测 SCLA0 引脚电平（仅当 IICCTL00.IICE= 1时有效) 0: 检测到 SCLA0 引脚为低电平 1: 检测到 SCLA0 引脚为高电平	R
6	SVADIS	地址匹配禁用标志 0: 禁用所有地址匹配功能 1: 启用所有地址匹配功能	R/W
7	WUP	地址匹配唤醒控制 0: 停止软件待机模式下的地址匹配唤醒功能 1: 启用软件待机模式下的地址匹配唤醒功能	R/W

该寄存器用于设置I²C的操作模式，并检测 SCLA0 和 SDAA0 引脚的状态。

CLD 和 DAD 位是只读的。

设置 IICCTL01 寄存器，但 WUP 位除外，同时禁用I²C操作 (IICCTL00.IICE)。

IICA 操作时钟(f_{MCK}) 的最快操作频率为20 MHz（最大值）。

仅当 PCLKB 超过20 MHz时，才将 IICA 控制寄存器 01 (IICCTL01) 的 PRS 位设置为 1

设置传输时钟时， 请注意 PCLKB 的最小工作频率。

根据模式确定I²C 总线接口（IICA）的最小PCLKB工作频率。

快速模式： PCLKB = 3.5 MHz (min.)

快速模式加强版： PCLKB = 10 MHz (min.)

正常模式： PCLKB= 1 MHz（最小值）

PRS bit (IICA Operation Clock (f_{MCK}))

The PRS bit is used to set IICA operation clock (f_{MCK}).

DFC bit (Digital Filter Operation Control)

Use the digital filter only in fast mode and fast mode plus.

The digital filter is used for noise elimination.

The transfer clock does not vary, regardless of the DFC bit being set to 1 or cleared to 0.

SMC bit (Operation Mode Switching)

The SMC bit is used for operation mode switching.

DAD bit (Detection of SDAA0 Pin Level (Valid Only when IICCTL00.IICE = 1))

Condition for clearing (DAD = 0)

- When the SDAA0 pin is at low level
- When IICCTL00.IICE = 0 (operation stop)
- Reset

Condition for setting (DAD = 1)

- When the SDAA0 pin is at high level

CLD bit (Detection of SCLA0 Pin Level (Valid Only when IICCTL00.IICE = 1))

Condition for clearing (CLD = 0)

- When the SCLA0 pin is at low level
- When IICCTL00.IICE = 0 (operation stop)
- Reset

Condition for setting (CLD = 1)

- When the SCLA0 pin is at high level

SVADIS bit (Address Match Disabling Flag)

When SVADIS = 1, IICA considers any address as address match, and performs the same operation as that when an extension code is received.

Therefore, IICS0.COI is set to 1, and IICS0.EXC is set to 1.

For details about extension code reception, see [section 20.3.13. Extension Code](#).

WUP bit (Control of Address Match Wakeup)

To shift to Software Standby mode when WUP = 1, execute WFI instruction while SBYCR.SSBY bit is 1 at least three cycles of f_{MCK} after setting the WUP bit to 1 (see [Table 20.5](#)).

Clear the WUP bit to 0 after the address has matched, an address has been received while the all address match function is enabled, or an extension code has been received. The subsequent communication can be entered by the clearing the WUP bit to 0. (The interface must be released from the clock stretch state and transmit data must be written after the WUP bit has been cleared to 0.)

The interrupt timing when the address has matched, when an address has been received while the all address match function is enabled, or when an extension code has been received, while WUP = 1, is identical to the interrupt timing when WUP = 0. (A delay of the difference of sampling by the clock will occur.) Furthermore, when WUP = 1, a stop condition interrupt is not generated even if the IICCTL00.SPIE bit is set to 1.

Condition for clearing (WUP = 0)

- Cleared by instruction (after address match, address reception with the all address match function enabled, or extension code reception)

Condition for setting (WUP = 1)

PRS 位 (IICA 操作时钟 (f_{MCK}))

PRS 位用于设置 IICA 操作时钟 (f_{MCK})。

DFC 位 (数字滤波器运行控制)

数字滤波器仅在快速模式和快速模式增强模式下使用。

数字滤波器用于消除噪声。

无论 DFC 位设置为 1 还是清除为 0，传输时钟都不会改变。

SMC 位 (操作模式切换)

SMC 位用于操作模式切换。

DAD 位 (检测 SDAA0 引脚电平 (仅当 IICCTL00.IICE= 1时有效))

清除条件(DAD = 0)

- 当 SDAA0 引脚为低电平时
- 当 IICCTL00.IICE = 0 (操作停止)
- 重置

设置(DAD = 1的条件

- 当 SDAA0 引脚为高电平时

CLD 位 (检测 SCLA0 引脚电平 (仅当 IICCTL00.IICE= 1时有效))

清除条件(CLD = 0)

- 当 SCLA0 引脚为低电平时
- 当 IICCTL00.IICE = 0 (操作停止)
- 重置

设置(CLD = 1的条件

- 当 SCLA0 引脚为高电平时

SVADIS 位 (地址匹配禁用标志)

当 SVADIS= 1时，IICA 将任何地址视为地址匹配，并执行与收到扩展码时相同的操作。

因此，IICS0.COI 设置为 1，IICS0.EXC 设置为 1。

有关分机号接收的详细信息，请参阅第 20.3.13 节。分机号。

WUP 位 (地址匹配唤醒控制)

当WUP = 1时，要切换到软件待机模式，在将 WUP 位设置为 1 后，至少经过三个f_{MCK} 周期，且 SBYCR.SSBY 位为 1 时执行 WFI 指令（参见表 20.5）。

在地址匹配成功、启用全地址匹配功能并接收到地址或接收到扩展码后，将 WUP 位清零。之后，可通过将 WUP 位清零来进入后续通信。（WUP 位清零后，必须先解除接口的时钟拉伸状态，并写入发送数据。）

当地址匹配、在启用全地址匹配功能时接收到地址或接收到扩展码时（WUP = 1），中断时序与WUP =为 0 时的中断时序相同。（会产生时钟采样差的延迟。）此外，当WUP = 1为 0 时，即使 IICCTL00.SPIE 位设置为 1，也不会产生停止条件中断。

清除条件(WUP = 0)

- 通过指令清除（地址匹配后、启用全地址匹配功能接收地址后、或接收分机号后)

设置(WUP = 1的条件

- Set by instruction (when the IICS0.MSTS, IICS0.EXC, and IICS0.COI bits are 0, and the IICS0.STD bit also 0 (communication not entered))

The status of the IICA status register 0 (IICS0) must be checked and the WUP bit must be set during the period shown in Figure 20.3.

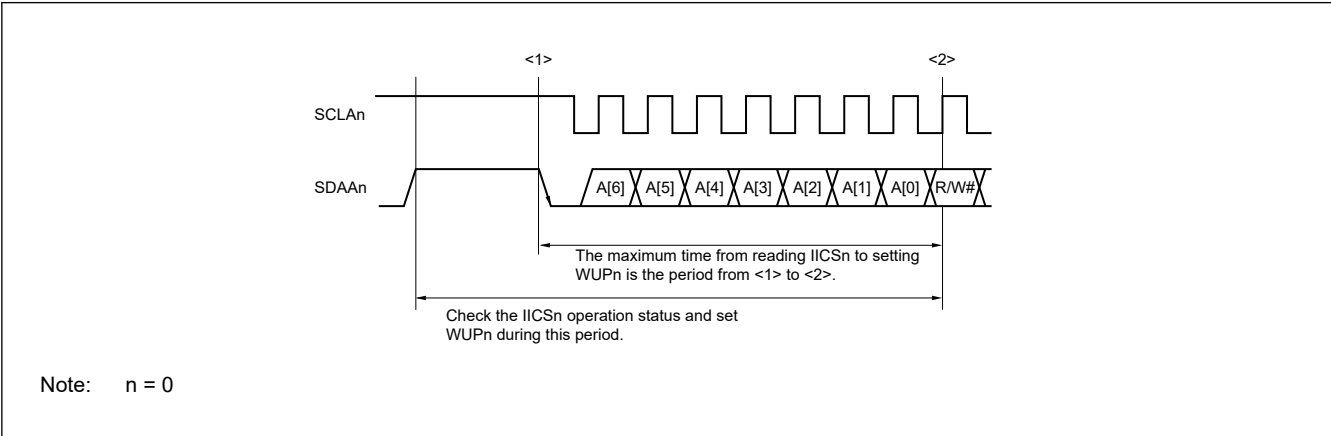


Figure 20.3 WUP bit setting period

20.2.7 IICWL0 : IICA Low-level Width Setting Register 0

Base address: IICA = 0x400A_3000
Offset address: 0x0102

Bit position: 7 6 5 4 3 2 1 0

Bit field: []

Value after reset: 1 1 1 1 1 1 1 1

Bit	Symbol	Function	R/W
7:0	n/a	SCLA0 Pin Low-width Configuration Data of Unit n	R/W

Note: The minimum serial clock cycle is (IICWL0 + 1) + (IICWH0 + 1)

This register is used to set the low-level width (t_{LOW}) of the SCLA0 pin signal that is output by I²C bus interface (IICA) and to control the SDAA0 pin signal.

Set the IICWL0 register while the I²C operation is disabled (IICCTL00.IICE).

For details about setting the IICWL0 register, see [section 20.3.2. Setting Transfer Clock Using IICWL0 and IICWH0 Registers](#). The data hold time is one-quarter of the time set by the IICWL0 register.

20.2.8 IICWH0 : IICA High-level Width Setting Register 0

Base address: IICA = 0x400A_3000
Offset address: 0x0103

Bit position: 7 6 5 4 3 2 1 0

Bit field: []

Value after reset: 1 1 1 1 1 1 1 1

Bit	Symbol	Function	R/W
7:0	n/a	SCLA0 Pin High-width Configuration Data of Unit n	R/W

Note: The minimum serial clock cycle is (IICWL0 + 1) + (IICWH0 + 1)

- 通过指令设置（当 IICS0.MSTS、IICS0.EXC 和 IICS0.COI 位均为 0，且 IICS0.STD 位也为 0 时（未进入通信状态））

必须检查 IICA 状态寄存器 0 (IICS0) 的状态，并且必须在图 20.3 所示的期间内设置 WUP 位。

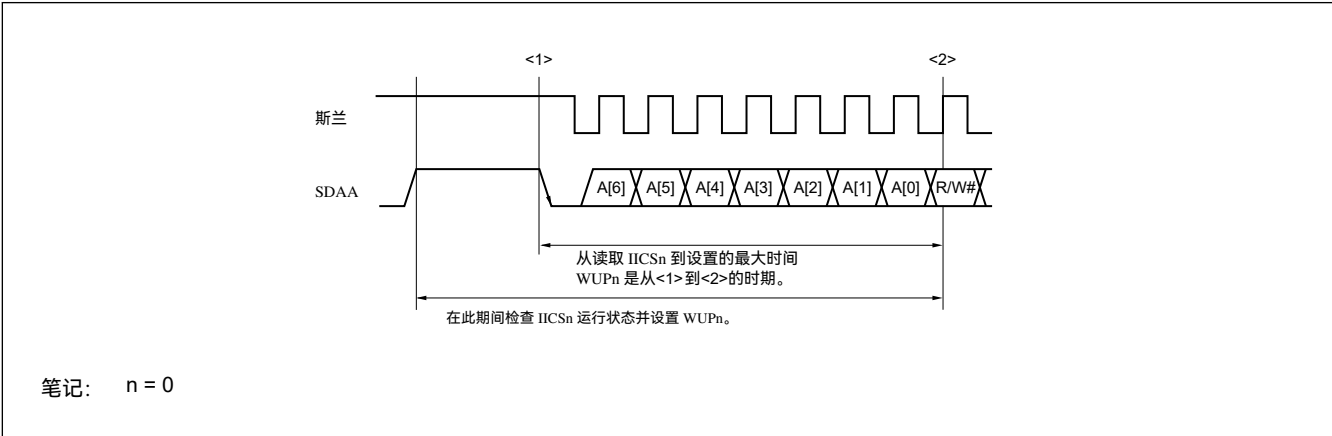


图 20.3WUP 位设置周期

20.2.7IICWL0: IICA 低级宽度设置寄存器 0

基本地址: IICA = 0x400A_3000
偏移地址: 0x0102

位位置: 7 6 5 4 3 2 1 0

位域: []

重置后的值: 1 1 1 1 1 1 1 1

少量	象征	功能	R/W
7:0	n/a	SCLA0 引脚低宽度配置数据（单元 n）	R/W

注意: 最小串行时钟周期为(IICWL0 + 1) + (IICWH0 + 1)

该寄存器用于设置总线接口 (IICA) 输出的 SCLA0 引脚信号的低电平宽度(t_{LOW})，并控制 SDAA0 引脚信号。

在禁用I²C操作时设置 IICWL0 寄存器 (IICCTL00.IICE)。

有关设置 IICWL0 寄存器的详细信息，请参见 20.3.2 节“使用 IICWL0 和 IICWH0 寄存器设置传输时钟”。数据保持时间为 IICWL0 寄存器设置时间的四分之一。

20.2.8IICWH0: IICA 高级宽度设置寄存器 0

基本地址: IICA = 0x400A_3000
偏移地址: 0x0103

位位置: 7 6 5 4 3 2 1 0

位域: []

重置后的值: 1 1 1 1 1 1 1 1

少量	象征	功能	R/W
7:0	n/a	SCLA0 引脚高宽度配置数据（单元 n）	R/W

注意: 最小串行时钟周期为(IICWL0 + 1) + (IICWH0 + 1)

This register is used to set the high-level width of the SCLAn pin signal that is output by I²C bus interface (IICA) and to control the SDAA0 pin signal.

Set the IICWH0 register while the I²C operation is disabled (IICCTL00.IICE).

For the procedures for setting the transfer clock in master mode and the IICWL0 and IICWH0 registers in slave mode, refer to (1) [Transfer clock setting in master mode](#) and (2) [Setting of IICWL0 and IICWH0 registers in slave mode](#), respectively.

20.2.9 Registers to Control the Port Function Multiplexed with the I²C I/O Pins

For information on how to set up the I/O ports, see [section 15, I/O Ports](#).

Set the IICCTL00.IICE bit to 1 before setting the output mode because the SCLAn and the SDAA0 pins output a low level (fixed) when the IICCTL00.IICE bit is 0.

20.3 I²C Bus Definitions and Control Methods

The following section describes the I²C bus's serial data communication format and the signals used by the I²C bus. [Figure 20.4](#) shows the transfer timing for the “start condition”, “address”, “data”, and “stop condition” output via the I²C bus's serial data bus.

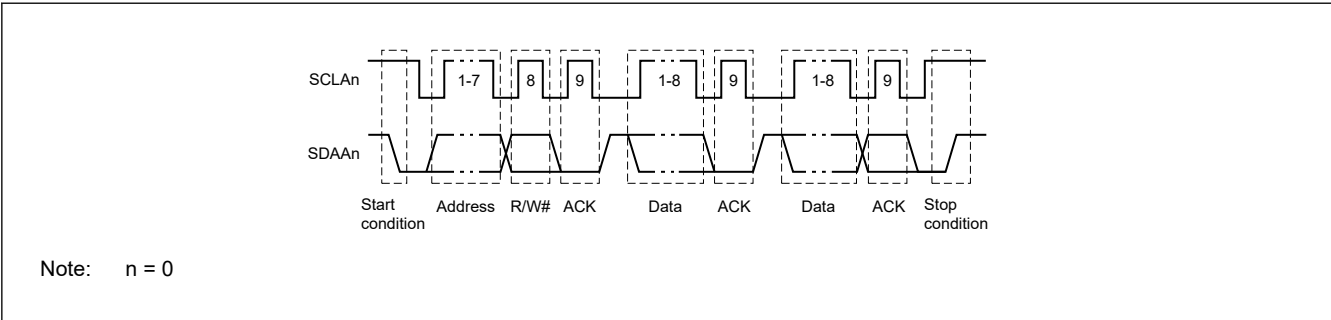


Figure 20.4 I²C bus serial data transfer timing

The master device generates the start condition, slave address, and stop condition.

The acknowledge (ACK) can be generated by either the master or slave device (normally, it is output by the device that receives 8-bit data).

The serial clock (SCLAn) is continuously output by the master device. However, for the slave device, the period over which the SCLAn pin is at the low level can be extended and clock stretching can be inserted.

20.3.1 Pin Configuration

The serial clock pin (SCLAn) and the serial data bus pin (SDAA0) are configured as follows.

1. SCLAn: This pin is used for serial clock input and output.
This pin is an N-ch open-drain output for both master and slave devices. Input is Schmitt input.
2. SDAA0: This pin is used for serial data input and output.
This pin is an N-ch open-drain output for both master and slave devices. Input is Schmitt input.

Since outputs from the serial clock line and the serial data bus line are N-ch open-drain outputs, an external pull-up resistor is required. [Figure 20.2](#) shows a serial bus configuration example using the I²C bus.

20.3.2 Setting Transfer Clock Using IICWL0 and IICWH0 Registers

(1) Transfer clock setting in master mode

$$\text{Transfer clock} = \frac{f_{MCK}}{IICWL + IICWH + f_{MCK}(t_R + t_F)}$$

At this time, the optimal setting values of the IICWL0 and IICWH0 registers are as follows. (The fractional parts of all setting values are rounded up.)

- When the fast mode

该寄存器用于设置由I²C总线接口 (IICA) 输出的 SCLAn 引脚信号的高电平宽度，并控制 SDAA0 引脚信号。

在禁用I²C操作时设置 IICWH0 寄存器 (IICCTL00.IICE)。

有关在主模式下设置传输时钟以及在从模式下设置 IICWL0 和 IICWH0 寄存器的步骤，请分别参阅 (1) 主模式下的传输时钟设置和 (2) 从模式下的 IICWL0 和 IICWH0 寄存器设置。

20.2.9 用于控制端口功能的寄存器与I²C I/O引脚复用

有关如何设置I/O端口的信息，请参阅第 15 节“I/O 端口”。

在设置输出模式之前，请将 IICCTL00.IICE 位设置为 1，因为当 IICCTL00.IICE 位为 0 时，SCLAn 和 SDAA0 引脚输出低电平（固定）。

20.3I²C 总线定义和控制方法

以下部分描述了I²C总线的串行数据通信格式以及I²C总线使用的信号。图20.4显示了通过I²C总线串行数据总线输出的“起始条件”、“地址”、“数据”和“停止条件”的传输时序。

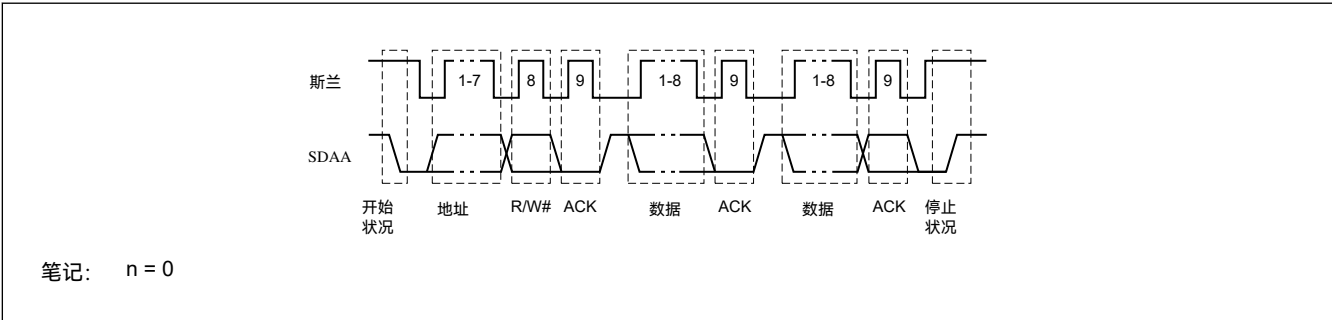


图 20.4 I2C 总线串行数据传输时序

主设备生成启动条件、从设备地址和停止条件。

确认（ACK）可以由主设备或从设备生成（通常由接收 8 位数据的设备输出）。

主设备持续输出串行时钟 (SCLAn)。然而，对于从设备，可以延长 SCLAn 引脚处于低电平的时间段，并插入时钟拉伸信号。

20.3.1引脚配置

串行时钟引脚（SCLAn）和串行数据总线引脚（SDAA0）配置如下。

1. SCLAn: 此引脚用于串行时钟输入和输出。
该引脚是N沟道开漏输出，适用于主设备和从设备。输入为施密特输入。
2. SDAA0: 此引脚用于串行数据输入和输出。
该引脚是N沟道开漏输出，适用于主设备和从设备。输入为施密特输入。

由于串行时钟线和串行数据总线的输出均为 N 沟道开漏输出，因此需要一个外部上拉电阻。图 20.2 显示了一个使用I²C总线的串行总线配置示例。

20.3.2 使用 IICWL0 和 IICWH0 寄存器设置传输时钟

(1) 主模式下的传输时钟设置

$$\text{Transfer clock} = \frac{f_{MCK}}{IICWL + IICWH + f_{MCK}(t_R + t_F)}$$

此时，IICWL0 和 IICWH0 寄存器的最佳设置值如下。（所有设置值的小数部分均向上取整。）

- 快速模式

$$\text{IICWL0} = \frac{0.52}{\text{Transfer clock}} \times f_{\text{MCK}}$$
$$\text{IICWH0} = \left(\frac{0.48}{\text{Transfer clock}} - t_R - t_F \right) \times f_{\text{MCK}}$$

- When the normal mode
$$\text{IICWL0} = \frac{0.47}{\text{Transfer clock}} \times f_{\text{MCK}}$$
$$\text{IICWH0} = \left(\frac{0.53}{\text{Transfer clock}} - t_R - t_F \right) \times f_{\text{MCK}}$$
- When the fast mode plus
$$\text{IICWL0} = \frac{0.50}{\text{Transfer clock}} \times f_{\text{MCK}}$$
$$\text{IICWH0} = \left(\frac{0.50}{\text{Transfer clock}} - t_R - t_F \right) \times f_{\text{MCK}}$$

(2) Setting of IICWL0 and IICWH0 registers in slave mode

The fractional parts of all setting values are rounded up.

- When the fast mode
$$\text{IICWL0} = 1.3 \mu\text{s} \times f_{\text{MCK}}$$
$$\text{IICWH0} = (1.2 \mu\text{s} - t_R - t_F) \times f_{\text{MCK}}$$
- When the normal mode
$$\text{IICWL0} = 4.7 \mu\text{s} \times f_{\text{MCK}}$$
$$\text{IICWH0} = (5.3 \mu\text{s} - t_R - t_F) \times f_{\text{MCK}}$$
- When the fast mode plus
$$\text{IICWL0} = 0.50 \mu\text{s} \times f_{\text{MCK}}$$
$$\text{IICWH0} = (0.50 \mu\text{s} - t_R - t_F) \times f_{\text{MCK}}$$

Note: Calculate the rise time (t_R) and fall time (t_F) of the SDAA0 and SCLA0 signals separately, because they differ depending on the pull-up resistance and wire load.

Note: IICWL0: IICA low-level width setting register 0
IICWH0: IICA high-level width setting register 0
t_F: SDAA0 and SCLA0 signal falling times
t_R: SDAA0 and SCLA0 signal rising times
f_{MCK}: IICA operation clock frequency

Note: The minimum serial clock cycle is (IICWL0 + 1) + (IICWH0 + 1). Determine the values to be set in the IICWL0 and IICWH0 registers by considering the rise time (t_R) and fall time (t_F) of the SDAA0 and SCLA0 signals.

20.3.3 Start Conditions

When the SCLA0 pin is at high level, changing the SDAA0 pin from the high level to the low level generates a start condition.

A start condition is a signal that the master device generates to the slave device when starting a serial transfer. When the device is used as a slave device, start conditions can be detected. [Figure 20.5](#) shows the start conditions.

$$\text{IICWL0} = \frac{0.52}{\text{Transfer clock}} \times f_{\text{MCK}}$$
$$\text{IICWH0} = \left(\frac{0.48}{\text{Transfer clock}} - t_R - t_F \right) \times f_{\text{MCK}}$$

- 正常模式
$$\text{IICWL0} = \frac{0.47}{\text{Transfer clock}} \times f_{\text{MCK}}$$
$$\text{IICWH0} = \left(\frac{0.53}{\text{Transfer clock}} - t_R - t_F \right) \times f_{\text{MCK}}$$
- 当快速模式+
$$\text{IICWL0} = \frac{0.50}{\text{Transfer clock}} \times f_{\text{MCK}}$$
$$\text{IICWH0} = \left(\frac{0.50}{\text{Transfer clock}} - t_R - t_F \right) \times f_{\text{MCK}}$$

(2) 从模式下的 IICWL0 和 IICWH0 寄存器设置

所有设置值的小数部分都向上取整。

- 快速模式
$$\text{IICWL0} = 1.3 \mu\text{s} \times f_{\text{MCK}}$$
$$\text{IICWH0} = (1.2 \mu\text{s} - t_R - t_F) \times f_{\text{MCK}}$$
- 正常模式
$$\text{IICWL0} = 4.7 \mu\text{s} \times f_{\text{MCK}}$$
$$\text{IICWH0} = (5.3 \mu\text{s} - t_R - t_F) \times f_{\text{MCK}}$$
- 当快速模式+
$$\text{IICWL0} = 0.50 \mu\text{s} \times f_{\text{MCK}}$$
$$\text{IICWH0} = (0.50 \mu\text{s} - t_R - t_F) \times f_{\text{MCK}}$$

注意: SDAA0 和 SCLA0 信号的上升时间(t_R) 和下降时间 (t_F) 分别计算, 因为它们会因上拉电阻和导线负载的不同而有所差异。

注: IICWL0: IICA 低级宽度设置寄存器 0
IICWH0: IICA 高级宽度设置寄存器 0
t_F: SDAA0 和 SCLA0 信号下降时间
t_R: SDAA0 和 SCLA0 信号上升时间
f_{MCK}: IICA 操作时钟频率

注意: 最小串行时钟周期为(IICWL0 + 1) + (IICWH0 + 1)。考虑SDAA0和SCLA0信号的上升时间(t_R)和下降时间(t_F) , 确定IICWL0和IICWH0寄存器中要设置的值。

20.3.3 起始条件

当 SCLA0 引脚为高电平时, 将 SDAA0 引脚从高电平变为低电平会产生启动条件。

启动条件是主设备在启动串行传输时向从设备发送的信号。当设备用作从设备时, 可以检测到启动条件。图 20.5 显示了启动条件。

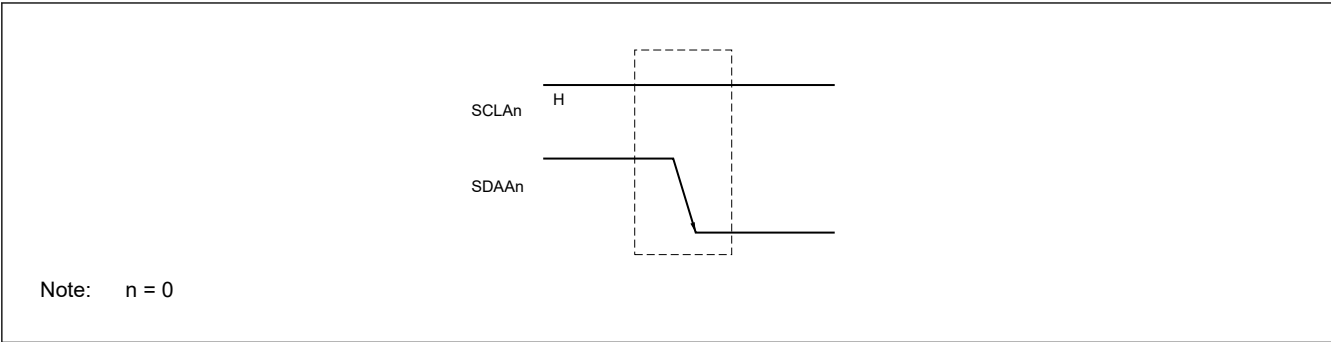


Figure 20.5 Start conditions

A start condition is output when STT bit of IICA control register 00 (IICCTL00) is set to 1 after a stop condition has been detected (SPD bit of the IICA status register 0 (IICS0) = 1)). When a start condition is detected, STD bit of the IICS0 register is set to 1.

20.3.4 Address

The address is defined by the 7 bits of data that follow the start condition.

An address is a 7-bit data segment that is output in order to select one of the slave devices that are connected to the master device via the bus lines. Therefore, each slave device connected via the bus lines must have a unique address. The slave devices include hardware that detects the start condition and checks whether the 7-bit address data matches the data values stored in the slave address register 0 (SVA0). If the address data matches the SVA0 register values, the slave device is selected and communicates with the master device until the master device generates a start condition or stop condition. Figure 20.6 shows the address.

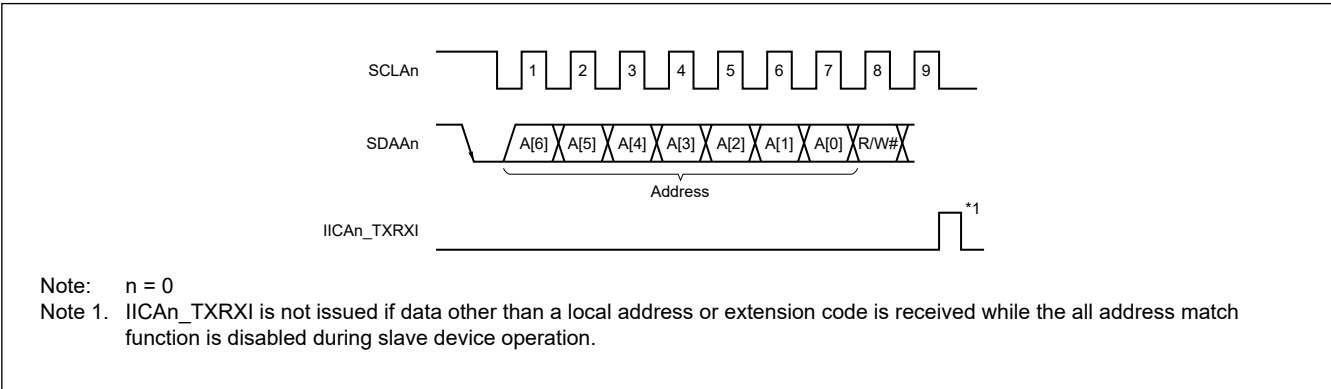


Figure 20.6 Addresses

Addresses are output when a total of 8 bits consisting of the slave address and the transfer direction described in [section 20.3.5. Transfer Direction Specification](#) are written to the IICA shift register 0 (IICA0). The received addresses are written to the IICA0 register.

The slave address is assigned to the higher 7 bits of the IICA0 register.

20.3.5 Transfer Direction Specification

In addition to the 7-bit address data, the master device sends 1 bit that specifies the transfer direction.

When this transfer direction specification bit has a value of 0, it indicates that the master device is transmitting data to a slave device. When the transfer direction specification bit has a value of 1, it indicates that the master device is receiving data from a slave device. Figure 20.7 shows the transfer direction specification.

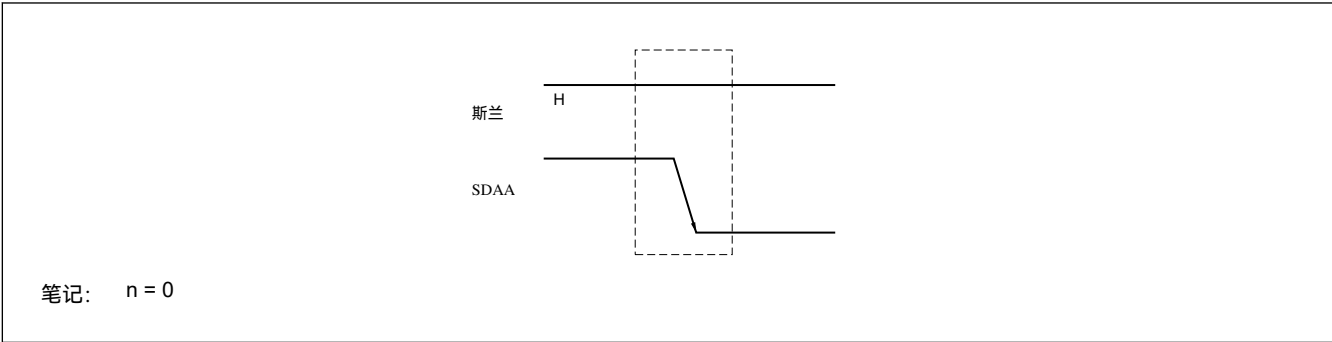


图 20.5 起始条件

当检测到停止条件（IICA 状态寄存器 0(IICS0) = 1)的 SPD 位）后，IICA 控制寄存器 00 (IICCTL00) 的 STT 位被置为 1 时，输出开始条件。当检测到开始条件时，IICS0 寄存器的 STD 位被置为 1。

20.3.4 地址

地址由起始条件之后的 7 位数据定义。

地址是一个7位数据段，用于选择通过总线连接到主设备的从设备之一。因此，每个通过总线连接的从设备都必须具有唯一的地址。从设备包含用于检测启动条件并检查7位地址数据是否与存储在从设备地址寄存器0 (SVA0) 中的数据值匹配的硬件。如果地址数据与SVA0寄存器值匹配，则选择该从设备并与主设备通信，直到主设备发出启动条件或停止条件。图 20.6 显示了地址。

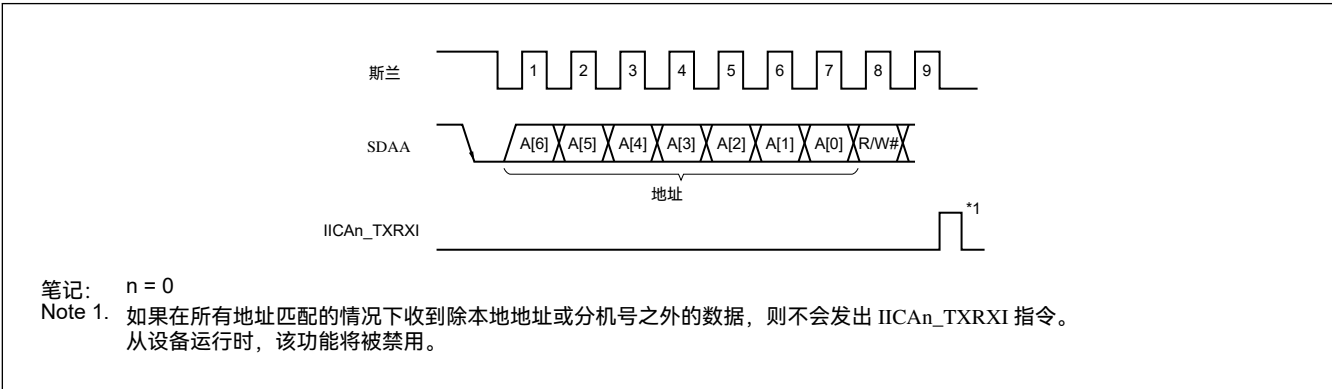


图 20.6 地址

当从机地址和传输方向（详见20.3.5节“传输方向规范”）共8位被写入IICA移位寄存器0（IICA0）时，地址即被输出。接收到的地址也被写入IICA0寄存器。

从机地址分配给 IICA0 寄存器的高 7 位。

20.3.5 传输方向规范

除了 7 位地址数据外，主设备还会发送 1 位来指定传输方向。

当传输方向指定位的值为 0 时，表示主设备正在向从设备发送数据。当传输方向指定位的值为 1 时，表示主设备正在从从设备接收数据。图 20.7 显示了传输方向指定。

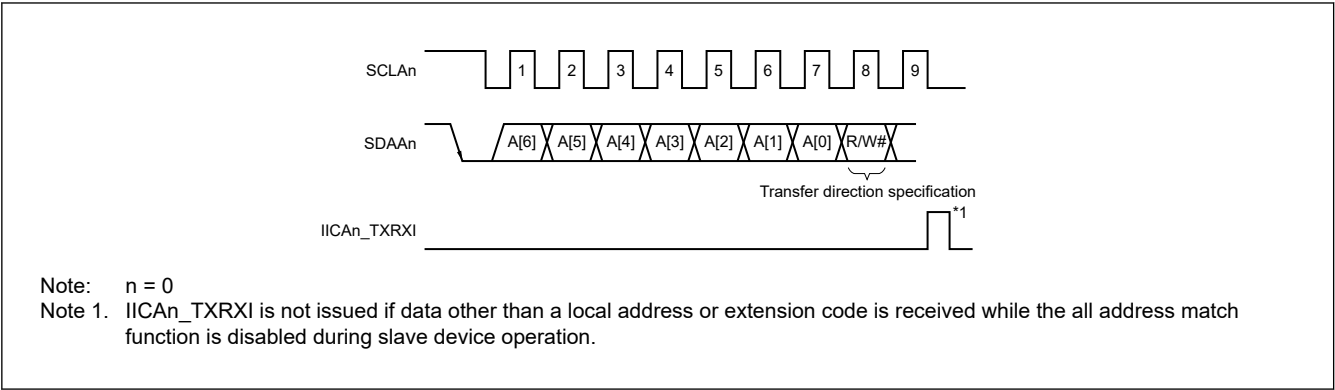


Figure 20.7 Transfer direction specification

20.3.6 Acknowledge (ACK)

ACK is used to check the status of serial data at the transmission and reception sides. The reception side returns ACK each time it has received 8-bit data.

The transmission side usually receives ACK after transmitting 8-bit data. When ACK is returned from the reception side, it is assumed that reception has been correctly performed and processing is continued. Whether ACK has been detected can be checked by using ACKD bit of the IICA status register 0 (IICS0).

When the master device receives the last data item, it does not return ACK and instead generates a stop condition. If a slave device does not return ACK after receiving data, the master device outputs a stop condition or restart condition and stops transmission. If ACK is not returned, the possible causes are as follows.

1. Reception was not performed normally.
2. The final data item was received.
3. The reception side specified by the address does not exist.

To generate ACK, the reception side makes the SDAA0 line low at the ninth clock (indicating normal reception). Automatic generation of ACK is enabled by setting ACKE bit of IICA control register 00 (IICCTL00) to 1. TRC bit of the IICS0 register is set to the value of the eighth bit that follows 7-bit address information. Usually, set the IICCTL00.ACKE bit to 1 for reception (IICCTL00.TRC = 0).

If a slave device can receive no more data during reception (IICCTL00.TRC = 0) or does not require the next data item, then the slave device must inform the master device, by clearing the IICCTL00.ACKE bit to 0, that it will not receive any more data.

When the master device does not require the next data item during reception (IICCTL00.TRC = 0), it must clear the IICCTL00.ACKE bit to 0 so that ACK is not generated. In this way, the master informs a slave device at the transmission side that it does not require any more data (transmission will be stopped). Figure 20.8 shows the ACK.

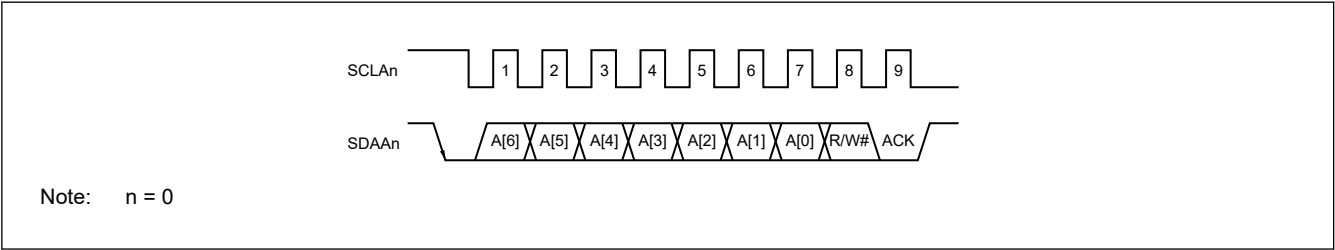


Figure 20.8 ACK

When the local address is received, ACK is automatically generated, regardless of the value of the IICCTL00.ACKE bit. When an address other than that of the local address is received, ACK is not generated (NACK).

When an extension code is received, or when an address is received while the all address match function is enabled, ACK is generated if the IICCTL00.ACKE bit is set to 1 in advance.

How ACK is generated when data is received depends on the setting of the timing of clock stretching as follows.

- When 8th cycle clock stretching is selected (IICCTL00.WTIM=0):

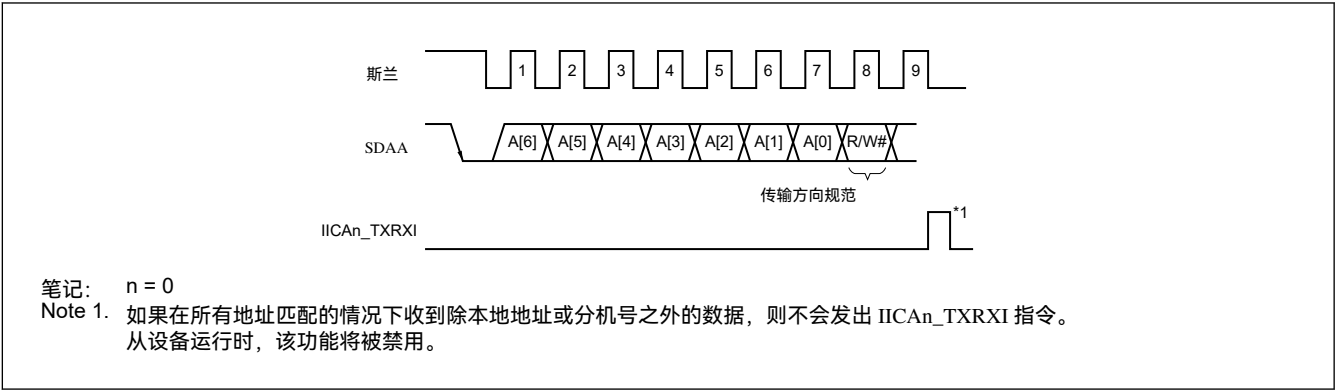


图 20.7 传输方向规范

20.3.6 确认 (ACK)

ACK用于检查串行数据在发送端和接收端的状态。接收端每次收到8位数据后都会返回ACK。

发送端通常在发送完 8 位数据后会收到确认 (ACK)。当接收端返回 ACK 时, 则假定接收已正确完成, 并继续进行处理。可以通过 IICA 状态寄存器 0 (IICS0) 的 ACKD 位来检查是否已检测到 ACK。

当主设备收到最后一个数据项时, 它不会返回 ACK, 而是生成停止条件。如果从设备在收到数据后没有返回 ACK, 主设备会输出停止条件或重启条件, 并停止传输。如果未返回 ACK, 可能的原因如下。

1. 接待工作未正常进行。
2. 已收到最后一个数据项。
3. 地址指定的接收方不存在。

为了生成 ACK, 接收端在第九个时钟周期将 SDAA0 线拉低 (表示正常接收)。通过将 IICA 控制寄存器 00 (IICCTL00) 的 ACKE 位设置为 1 来启用 ACK 的自动生成。IICS0 寄存器的 TRC 位设置为 7 位地址信息之后的第八位的值。通常, 对于接收(IICCTL00.TRC = 0), 将 IICCTL00.ACKE 位设置为 1。

如果从设备在接收期间无法接收更多数据, 或者不需要下一个数据项, 则从设备必须通过将 IICCTL00.ACKE 位清除为 0 来通知主设备, 表示它将不再接收任何数据。

当主设备在接收过程中不需要下一个数据项时, 它必须将 IICCTL00.ACKE 位清零, 以避免生成 ACK。这样, 主设备就通知发送端的从设备它不再需要任何数据 (传输将停止)。图 20.8 显示了 ACK。

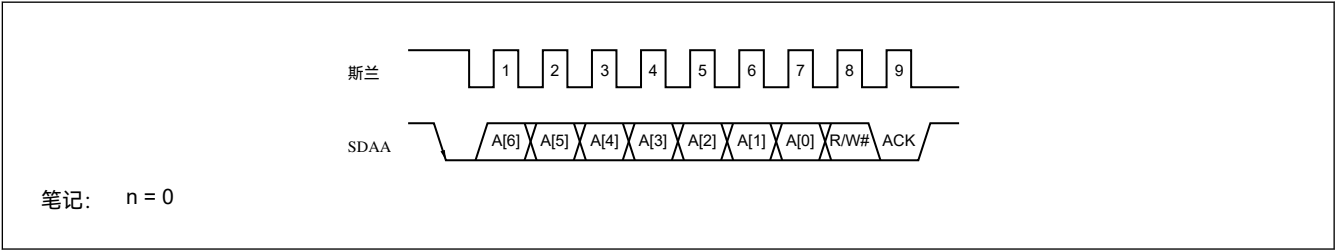


图 20.8ACK

当收到本地地址时, 无论 IICCTL00.ACKE 位的值如何, 都会自动生成 ACK。当收到与本地地址不同的地址时, 不会生成 ACK (NACK)。

当收到扩展码, 或者在启用所有地址匹配功能时收到地址时, 如果预先将 IICCTL00.ACKE 位设置为 1, 则会生成 ACK。

接收到数据时 ACK 的生成方式取决于时钟拉伸时序的设置, 如下所示。

- 当选择第 8 个周期时钟拉伸时 (IICCTL00.WTIM=0)

By setting the IICCTL00.ACKE bit to 1 before release from the clock stretch state, ACK is generated at the falling edge of the eighth clock cycle of the SCLAn pin.

- When 9th cycle clock stretching is selected (IICCTL00.WTIM=1):
ACK is generated if the IICCTL00.ACKE bit is set to 1 in advance.

20.3.7 Stop Condition

When the SCLAn pin is at high level, changing the SDAA0 pin from low level to high level generates a stop condition. A stop condition is a signal that the master device generates to the slave device when serial transfer has been completed. When the device is used as a slave device, stop conditions can be detected.

Figure 20.9 shows the stop conditions.

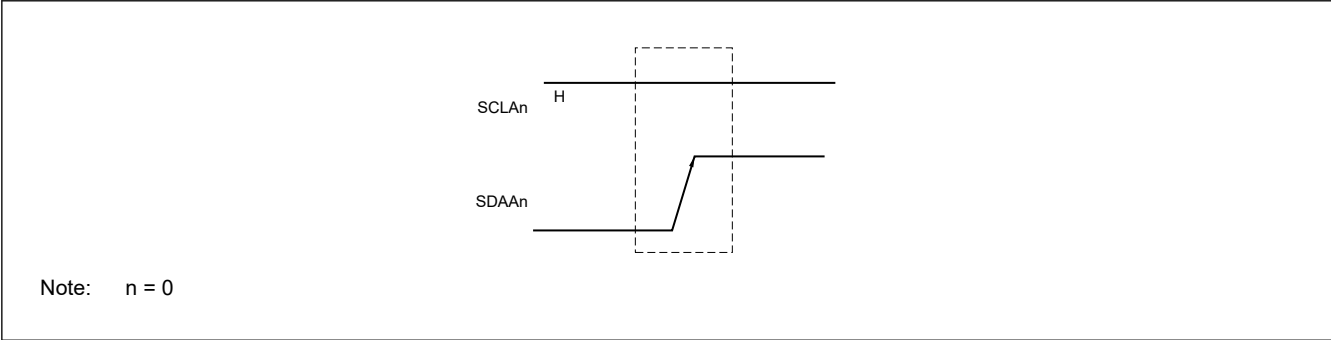


Figure 20.9 Stop condition

A stop condition is generated when SPT bit of IICA control register 00 (IICCTL00) is set to 1. When the stop condition is detected, SPD bit of the IICA status register 0 (IICS0) is set to 1 and IICA0_TXRXI is generated when SPIE bit of the IICCTL00 register is set to 1.

20.3.8 Clock Stretching

Clock stretching is used to notify the other party in communications that a device (master or slave) is preparing to transmit or receive data (i.e., the interface is in the clock stretch state).

Setting the SCLAn pin to the low level indicates the clock stretch state to the other party. When clock stretching is released for both the master and slave devices, the next data transfer can start. Figure 20.10 shows the clock stretching.

通过在从时钟拉伸状态释放之前将 IICCTL00.ACKE 位设置为 1，在 SCLAn 引脚的第八个时钟周期的下降沿生成 ACK。

- 选择第 9 个周期时钟拉伸时 (IICCTL00.WTIM=1)
如果预先将 IICCTL00.ACKE 位设置为 1，则会生成 ACK。

20.3.7 停止条件

当 SCLAn 引脚为高电平时，将 SDAA0 引脚从低电平变为高电平会产生停止信号。停止信号是主设备在串行传输完成后发送给从设备的信号。当设备用作从设备时，可以检测到停止信号。

图 20.9 显示了停止条件。

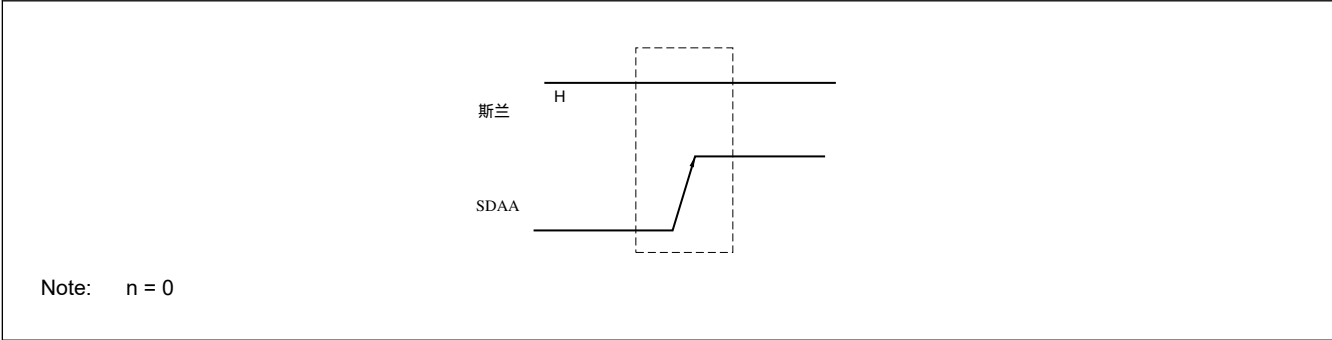


图 20.9 停止条件

当 IICA 控制寄存器 00 (IICCTL00) 的 SPT 位设置为 1 时，会产生停止条件。检测到停止条件时，IICA 状态寄存器 0 (IICS0) 的 SPD 位设置为 1，并且当 IICCTL00 寄存器的 SPIE 位设置为 1 时，会生成 IICA0_TXRXI。

20.3.8 点钟方向的拉伸

时钟拉伸用于在通信中通知另一方，设备（主设备或从设备）正在准备发送或接收数据（即，接口处于时钟拉伸状态）。

将 SCLAn 引脚置低表示时钟拉伸状态已解除。当主设备和从设备的时钟拉伸都解除后，即可开始下一次数据传输。图 20.10 显示了时钟拉伸过程。

(1) When clock stretching is set for the 9th and 8th clock cycles for the master and slave devices, respectively (master: transmission, slave: reception, and IICCTL00.ACKE = 1)

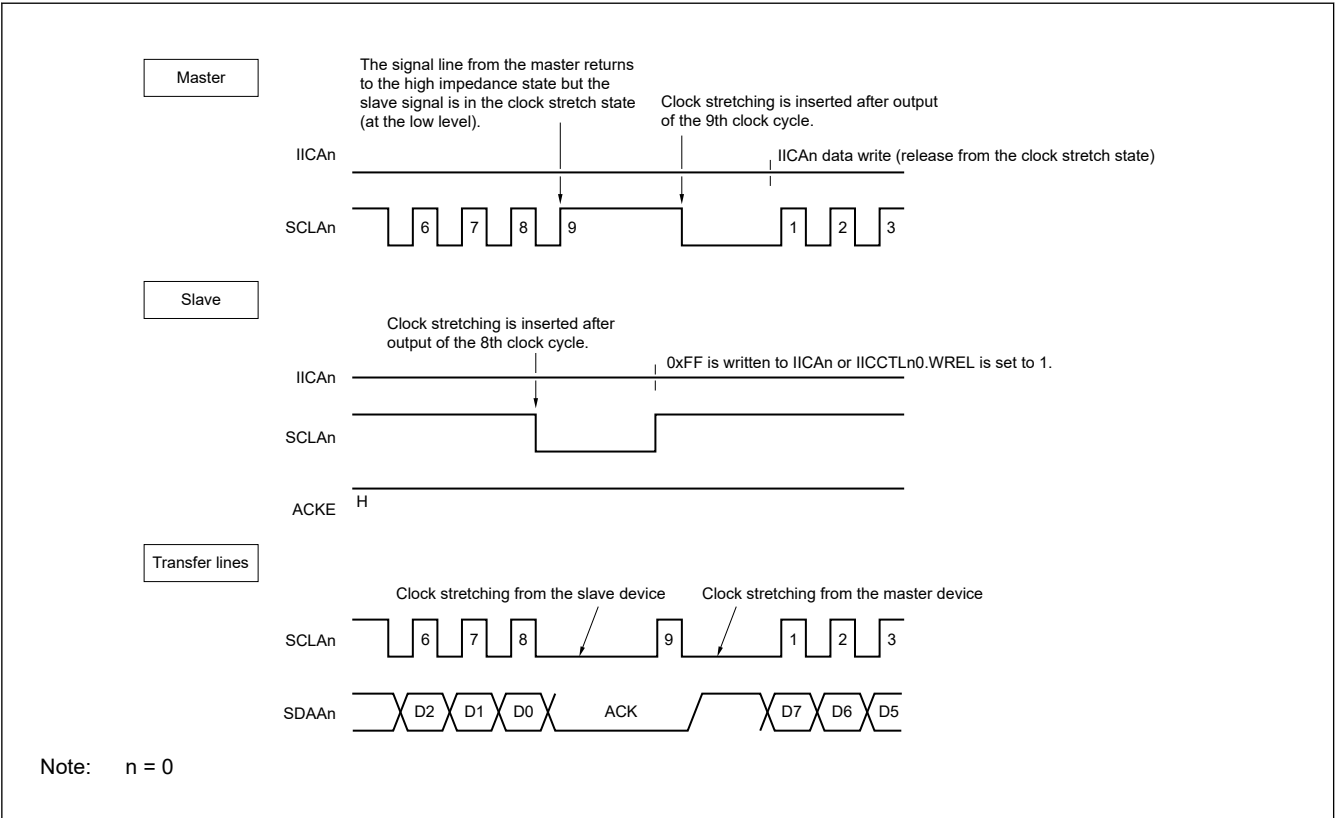


Figure 20.10 Clock stretching (1/2)

(1) 当主设备和从设备分别第 9 个和第 8 个时钟周期设置时钟拉伸时（主设备：发送，从设备：接收，并且 IICCTL00.ACKE= 1）

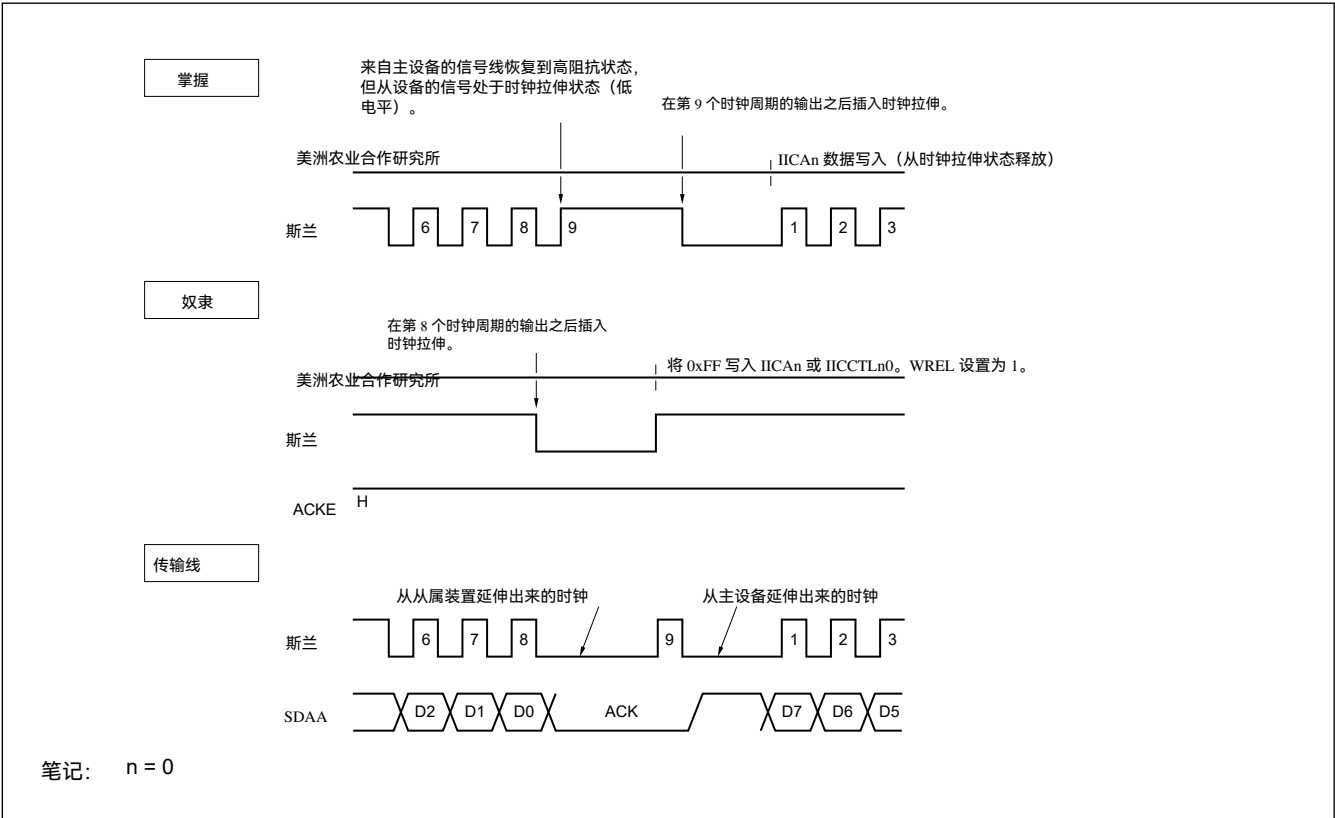


图 20.10 时钟拉伸 (1/2)

- (2) When clock stretching is set for the 9th clock cycle for both the master and slave devices (master: transmission, slave: reception, and IICCTL00.ACKE = 1)

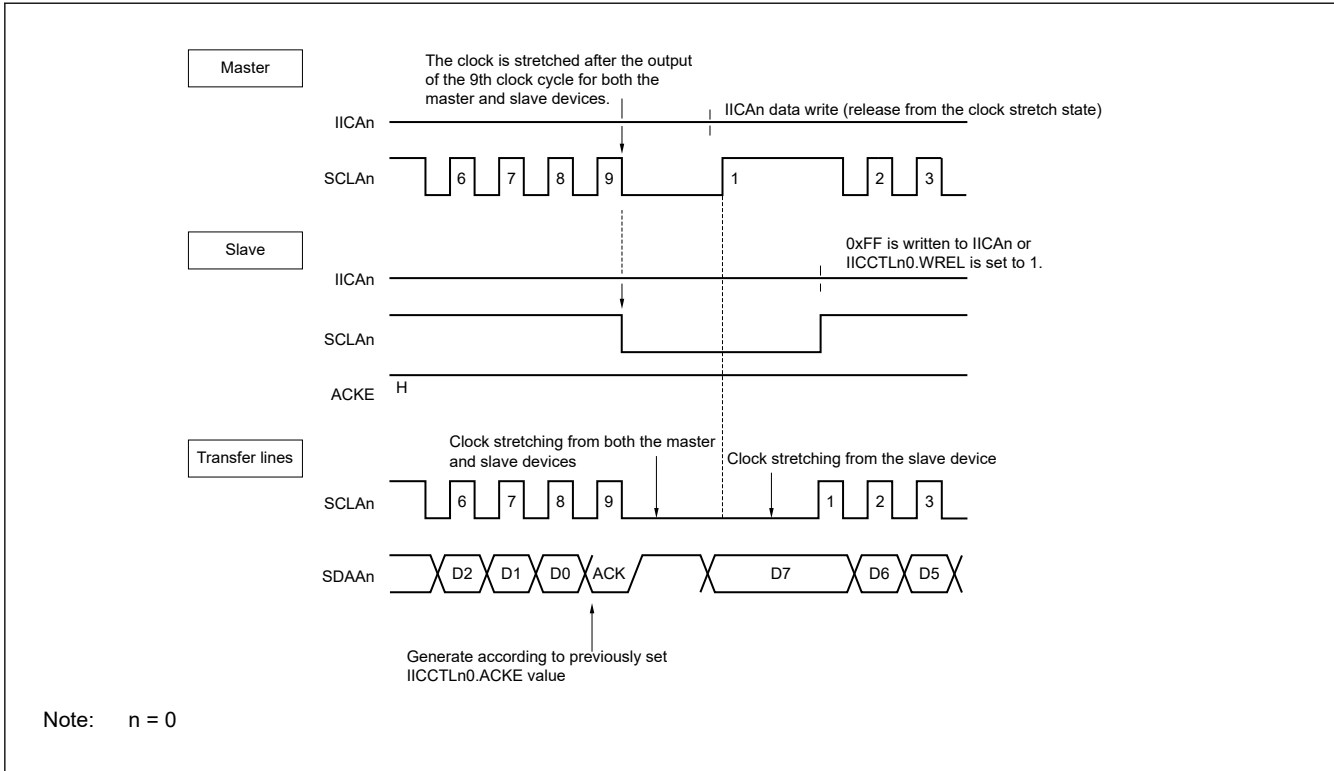


Figure 20.11 Clock stretching (2/2)

Clock stretching is automatically generated depending on the setting of WTIM bit of IICA control register 00 (IICCTL00). Normally, the receiving side releases the clock stretch state when WREL bit of the IICCTL00 register is set to 1 or when 0xFF is written to the IICA shift register 0 (IICA0), and the transmitting side releases the clock stretch state when data is written to the IICA0 register.

The master device can also releases the clock stretch state via either of the following methods.

- By setting STT bit of the IICCTL00 register to 1
- By setting SPT bit of the IICCTL00 register to 1

20.3.9 Release from Clock Stretching

The I²C interface usually releases the clock stretch state by the following processing.

- Writing data to the IICA shift register 0 (IICA0)
- Setting WREL bit of IICA control register 00 (IICCTL00) (release from the clock stretch state)
- Setting STT bit of the IICCTL00 register (generating start condition)*1
- Setting SPT bit of the IICCTL00 register (generating stop condition)*1

Note 1. Master mode only

Executing the above processing for release from clock stretching leads to IICA releasing the clock stretch state after which communications are resumed.

To release the clock stretch state and transmit data (including addresses), write the data to the IICA0 register.

To receive data after release from the clock stretch state, or to complete data transmission, set WREL bit of the IICCTL00 register to 1.

To generate a restart condition after release from the clock stretch state, set STT bit of the IICCTL00 register to 1.

To generate a stop condition after release from the clock stretch state, set SPT bit of the IICCTL00 register to 1.

- (2) 当主设备和从设备在第 9 个时钟周期设置时钟拉伸时（主设备：发送，从设备：接收，并且 IICCTL00.ACKE = 1）

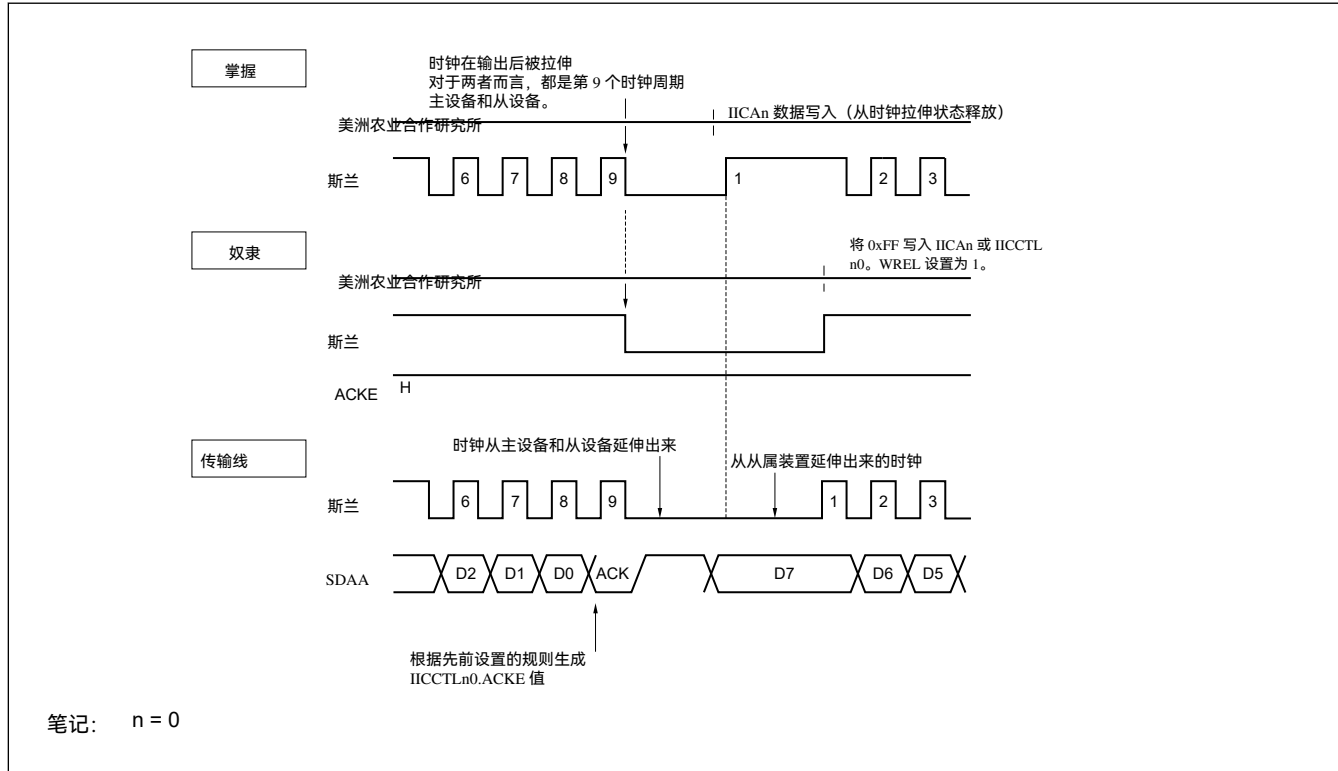


图 20.11 时钟拉伸 (2/2)

时钟拉伸是根据 IICA 控制寄存器 00 (IICCTL00) 的 WTIM 位设置自动生成的。通常情况下，当接收端将 IICCTL00 寄存器的 WREL 位设置为 1 或将 0xFF 写入 IICA 移位寄存器 0 (IICA0) 时，接收端会释放时钟拉伸状态；而当数据写入 IICA0 寄存器时，发送端会释放时钟拉伸状态。

主设备还可以通过以下两种方法之一释放时钟拉伸状态。

- 通过将 IICCTL00 寄存器的 STT 位设置为 1
- 通过将 IICCTL00 寄存器的 SPT 位设置为 1

20.3.9 时钟拉伸的释放

I²C 接口通常通过以下处理释放时钟拉伸状态。

- 向 IICA 移位寄存器 0 (IICA0) 写入数据
- 设置 IICA 控制寄存器 00 (IICCTL00) 的 WREL 位（从时钟拉伸状态释放）
- 设置 IICCTL00 寄存器的 STT 位（生成起始条件）*1
- 设置 IICCTL00 寄存器的 SPT 位（生成停止条件）*1

注1: 仅限主模式

执行上述时钟拉伸释放处理后，IICA 将释放时钟拉伸状态，之后通信恢复。

要释放时钟拉伸状态并传输数据（包括地址），请将数据写入 IICA0 寄存器。

要在从时钟拉伸状态释放后接收数据，或完成数据传输，请将 IICCTL00 寄存器的 WREL 位设置为 1。

要在从时钟拉伸状态释放后生成重启条件，请将 IICCTL00 寄存器的 STT 位设置为 1。要在从时钟拉伸状态释放后生成停止条件，请将 IICCTL00 寄存器的 SPT 位设置为 1。

Execute the processing for release only once for each period in the clock stretch state.

If, for example, data is written to the IICA0 register after release from the clock stretch state by setting the IICCTL00.WREL bit to 1, an incorrect value may be output to SDAA0 line because the timing for changing the SDAA0 line conflicts with the timing for writing the IICA0 register.

In addition to the above, communications are stopped if the IICCTL00.IICE bit is cleared to 0 when communications have been aborted, so that the clock stretch state can be released.

If the I²C bus has deadlocked due to noise, the device can exit from communications by setting LREL bit of the IICCTL00 register to 1, so that the clock stretch state can be released.

If the processing for release from clock stretching is executed when IICCTL01.WUP = 1, the clock stretch state is not released.

20.3.10 Timing of Generation of the Interrupt Request Signal (IICA0_TXRXI) and Control of Clock Stretching

The setting of WTIM bit of IICA control register 00 (IICCTL00) determines the timing by which IICA0_TXRXI is generated and controls clock stretching, as shown in Table 20.2.

The numbers in the table indicate the pulses of the serial clock signal. Interrupt requests and control of clock stretching are both synchronized with the falling edge of these clock pulses.

Table 20.2 IICA0_TXRXI generation timing and control of clock stretching

WTIM	During slave device operation			During master device operation		
	Address	Data reception	Data transmission	Address	Data reception	Data transmission
0	g*1 *2	g*2	g*2	9	8	8
1	g*1 *2 *3	g*2	g*2	9	9	9

1. During address transmission/reception
 - Slave device operation: The timing of the interrupt and clock stretching depends on the conditions described in *1, *2 and *3 below, regardless of the setting of the IICCTL00.WTIM bit.
 - Master device operation: The interrupt and clock stretching occur at the falling edge of the ninth clock cycle, regardless of the setting of the IICCTL00.WTIM bit.
2. During data reception
 - All operation: The timing of the interrupt and clock stretching depends on the setting of the IICCTL00.WTIM bit.
3. During data transmission
 - All operation: The timing of the interrupt and clock stretching depends on the setting of the IICCTL00.WTIM bit.
4. Release from clock stretching

The four types of processing for release from clock stretching are as follows.

 - Writing data to the IICA shift register 0 (IICA0)
 - Setting WREL bit of IICA control register 00 (IICCTL00) (release from the clock stretch state)
 - Setting STT bit of the IICCTL00 register (generating start condition)*4
 - Setting SPT bit of the IICCTL00 register (generating stop condition)*4

When 8th cycle clock stretching has been selected (IICCTL00.WTIM = 0), the presence or absence of ACK generation must be determined before release from the clock stretch state.

5. Detection of stop condition

IICA0_TXRXI is generated when a stop condition is detected (only when IICCTL00.SPIE = 1).

Note 1. The IICA0_TXRXI signal of the slave device and clock stretching occur at the falling edge of the 9th clock cycle only when there is a match with the address set to the slave address register 0 (SVA0).

在时钟拉伸状态下，每个周期只执行一次释放处理。

例如，如果通过将 IICCTL00.WREL 位设置为 1 来解除时钟拉伸状态后，向 IICA0 寄存器写入数据，则可能会向 SDAA0 线输出不正确的值，因为更改 SDAA0 线的时序与写入 IICA0 寄存器的时序冲突。

除了上述情况外，当通信中止时，如果将 IICCTL00.IICE 位清除为 0，则通信停止，以便释放时钟拉伸状态。

如果I²C 总线因噪声而死锁，则设备可以通过将 IICCTL00 寄存器的 LREL 位设置为 1 来退出通信，从而释放时钟拉伸状态。

如果在 IICCTL01.WUP= 1时执行时钟拉伸释放处理，则时钟拉伸状态不会被释放。

20.3.10 中断请求信号 (IICA0_TXRXI) 的生成时序和时钟拉伸控制

IICA 控制寄存器 00 (IICCTL00) 的 WTIM 位设置决定了 IICA0_TXRXI 的生成时序，并控制时钟拉伸，如表 20.2 所示。

表格中的数字表示串行时钟信号的脉冲数。中断请求和时钟展宽控制都与这些时钟脉冲的下降沿同步。

表 20.2IICA0_TXRXI 生成时序和时钟拉伸控制

WTIM	从设备运行期间			主设备运行期间		
	地址	数据接收	数据传输	地址	数据接收	数据传输
0	g*1 *2	g*2	g*2	9	8	8
1	g*1 *2 *3	g*2	g*2	9	9	9

1. 在地址传输/接收期间
 - 从设备操作：中断和时钟拉伸的时序取决于以下*1、*2和 *3 中描述的条件，而与 IICCTL00.WTIM 位的设置无关。
 - 主设备操作：中断和时钟拉伸发生在第九个时钟周期的下降沿，与 IICCTL00.WTIM 位的设置无关。
2. 数据接收期间
 - 所有操作：中断和时钟拉伸的时序取决于 IICCTL00.WTIM 位的设置。
3. 数据传输期间
 - 所有操作：中断和时钟拉伸的时序取决于 IICCTL00.WTIM 位的设置。
4. Release from clock stretching

时钟拉伸释放的四种处理类型如下。

 - 向 IICA 移位寄存器 0 (IICA0) 写入数据
 - 设置 IICA 控制寄存器 00 (IICCTL00) 的 WREL 位（从时钟拉伸状态释放）
 - 设置IICCTL00寄存器的STT位（生成起始条件）*4
 - 设置IICCTL00寄存器的SPT位（生成停止条件）*4

当选择了第 8 个周期时钟拉伸时（IICCTL00.WTIM= 0），必须确定是否存在 ACK 生成，然后才能从时钟拉伸状态释放。

5. 停止条件检测

当检测到停止条件时（仅当 IICCTL00.SPIE = 1时），会生成 IICA0_TXRXI。

注 1. 从设备的 IICA0_TXRXI 信号和时钟拉伸仅在第 9 个时钟周期的下降沿发生，且仅当与从地址寄存器 0 (SVA0) 中设置的地址匹配时才会发生。

At this point, ACK is generated regardless of the value set to the IICCTL00.ACKE bit. For a slave device that has received an extension code, or has received an address while the all address match function is enabled, IICA0_TXRXI occurs at the falling edge of the eighth clock.

However, if the address does not match after restart, IICA0_TXRXI is generated at the falling edge of the 9th clock cycle, but clock stretching does not occur.

Note 2. When the Slave Address Register n (SVAn) does not match the received address, the address match function is disabled, and an extended code has not been received, neither IICA0_TXRXI nor clock stretching will occur.

Note 3. When the WTIM bit is set to 1, for a slave device that has received an extension code or an address while the "all address match" function is enabled, IICAn_TXRXI and clock stretching occur at the falling edge of both the eighth and ninth clock cycles.

Note 4. Master mode only

20.3.11 Address Match Detection Method

In I²C bus mode, the master device can select a particular slave device by transmitting the corresponding slave address. Address match can be detected automatically by hardware. An interrupt request signal (IICA0_TXRXI) occurs only when the address set to the slave address register 0 (SVA0) matches the slave address sent by the master device, when an address is received while the all address match function is enabled (IICCTL01.SVADIS = 1), or when an extension code has been received.

20.3.12 Error Detection

In I²C bus mode, the status of the serial data bus (SDAA0) during data transmission is captured by the IICA shift register 0 (IICA0) of the transmitting device, so the IICA data prior to transmission can be compared with the transmitted IICA data to enable detection of transmission errors. A transmission error is judged as having occurred when the compared data values do not match.

20.3.13 Extension Code

- When the higher 4 bits of the receive address are either 0000b or 1111b, the extension code reception flag (IICS0.EXC) is set to 1 for extension code reception and an interrupt request signal (IICA0_TXRXI) is issued at the falling edge of the eighth clock.
When an address is received while the all address match function is enabled, it is also determined that an extension code has been received.
The local address stored in the slave address register 0 (SVA0) is not affected.
- The settings below are specified if 11110xx0b is transferred from the master device by using a 10-bit address transfer while the SVA0 register is set to 11110xx0b or if an address is received while the all address match function is enabled. Note that IICA0_TXRXI occurs at the falling edge of the eighth clock.
 - Higher four bits of data match or the all address match function is enabled: IICS0.EXC = 1
 - Seven bits of data match or the all address match function is enabled: IICS0.COI = 1
- Since the processing after the interrupt request occurs differs according to the data that follows the extension code, such processing is performed by software.
If the extension code is received or an address is received with the all address match function enabled during operation as a slave mode, then the slave device is participating in communication even if its address does not match.
For example, after the extension code is received, if you do not wish to operate the target device as a slave device, set LREL bit of IICA control register 00 (IICCTL00) to 1 to set the communication standby status for the next communication operation.

Table 20.3 shows the bit definitions for the major extension codes.

Table 20.3 Bit definitions of major extension codes

Slave address	R/W# bit	Description
0 0 0 0 0 0	0	General call address
1 1 1 1 0 x x	0	10-bit slave address specification (during address authentication)
1 1 1 1 0 x x	1	10-bit slave address specification (after address match, when read command is issued)

Note: See the I² bus specifications issued by NXP Semiconductors for details of extension codes other than those described above.

此时，无论 IICCTL00.ACKE 位的值如何，都会生成 ACK。对于已接收到扩展码或在启用全地址匹配功能时已接收到地址的从设备，IICA0_TXRXI 事件发生在第八个时钟周期的下降沿。

但是，如果重启后地址不匹配，则会在第 9 个时钟周期的下降沿生成 IICA0_TXRXI，但不会进行时钟拉伸。

注 2. 当从机地址寄存器 n (SVAn) 与接收到的地址不匹配时，地址匹配功能将被禁用，并且尚未接收到扩展代码，因此不会发生 IICA0_TXRXI 或时钟拉伸。
注 3. 当 WTIM 位设置为 1 时，对于已接收扩展码或地址且启用“所有地址匹配”功能的从设备，IICAn_TXRXI 和时钟拉伸发生在第八和第九个时钟周期的下降沿。

注4: 仅限主模式

20.3.11地址匹配检测方法

在I²C 总线模式下，主设备可以通过发送相应的从设备地址来选择特定的从设备。

地址匹配可以由硬件自动检测。只有当设置到从机地址寄存器 0 (SVA0) 的地址与主设备发送的从机地址匹配时，或者当启用全地址匹配功能 (IICCTL01.SVADIS= 1) 时收到地址，或者收到扩展代码时，才会产生中断请求信号 (IICA0_TXRXI)。

20.3.12错误检测

在I²C 总线模式下，数据传输期间串行数据总线（SDAA0）的状态由发送设备的IICA移位寄存器0（IICA0）捕获，因此可以将传输前的IICA数据与已传输的IICA数据进行比较，从而检测传输错误。当比较后的数据值不匹配时，则判定发生了传输错误。

20.3.13扩展代码

- 当接收地址的高 4 位是 0000b 或 1111b 时，扩展码接收标志 (IICS0.EXC) 设置为 1 以接收扩展码，并在第八个时钟的下降沿发出中断请求信号 (IICA0_TXRXI)。

当启用所有地址匹配功能时，如果收到地址，则还会确定已收到分机号。

存储在从地址寄存器 0 (SVA0) 中的本地地址不受影响。
- 如果当 SVA0 寄存器设置为11110xx0b时，使用 10 位地址传输从主设备传输11110xx0b，或者当启用所有地址匹配功能时接收到地址，则指定以下设置。
请注意，IICA0_TXRXI 发生在第八个时钟的下降沿。
 - 数据高四位匹配或全地址匹配功能已启用： IICS0.EXC = 1
 - 七位数据匹配或全地址匹配功能已启用： IICS0.COI = 1
- 由于中断请求发生后的处理方式会根据扩展代码后面的数据而有所不同，因此这种处理方式由软件执行。

如果在从机模式下运行期间，启用了全地址匹配功能并接收到了扩展码或地址，则即使从机的地址不匹配，该从机设备也会参与通信。
例如，收到扩展码后，如果您不希望将目标设备作为从设备操作，请将 IICA 控制寄存器 00 (IICCTL00) 的 LREL 位设置为 1，以设置通信待机状态，以便进行下一次通信操作。

表 20.3 显示了主要扩展码的位定义。

表 20.3 主要扩展码的位定义

从地址	R/W# 位	描述
0 0 0 0 0 0	0	通用呼叫地址
1 1 1 1 0 x x	0	10 位从机地址规范（地址认证期间）
1 1 1 1 0 x x	1	10 位从机地址规范（地址匹配后，发出读取命令时）

注意：有关除上述扩展代码之外的其他扩展代码的详细信息，请参阅 NXP Semiconductors 发布的I²总线规范。

20.3.14 Arbitration

When several master devices simultaneously generate a start condition (when the IICCTL00.STT bit is set to 1 before the IICS0.STD bit is set to 1), communication among the master devices is performed as the clocks are adjusted until the data differs. This kind of operation is called arbitration.

When one of the master devices loses in arbitration, an arbitration loss flag (IICS0.ALD) is set to 1 through the timing by which the arbitration loss occurred, and the SCLA0 and SDAA0 lines are both set to high impedance, which releases the bus.

The arbitration loss is detected by checking IICS0.ALD = 1 with software at the timing of the next interrupt request (the 8th or 9th clock cycle, when a stop condition is detected, for instance).

For details of interrupt request timing, see [section 20.3.10. Timing of Generation of the Interrupt Request Signal \(IICA0_TXRXI\) and Control of Clock Stretching](#).

Figure 20.12 shows the arbitration timing example.

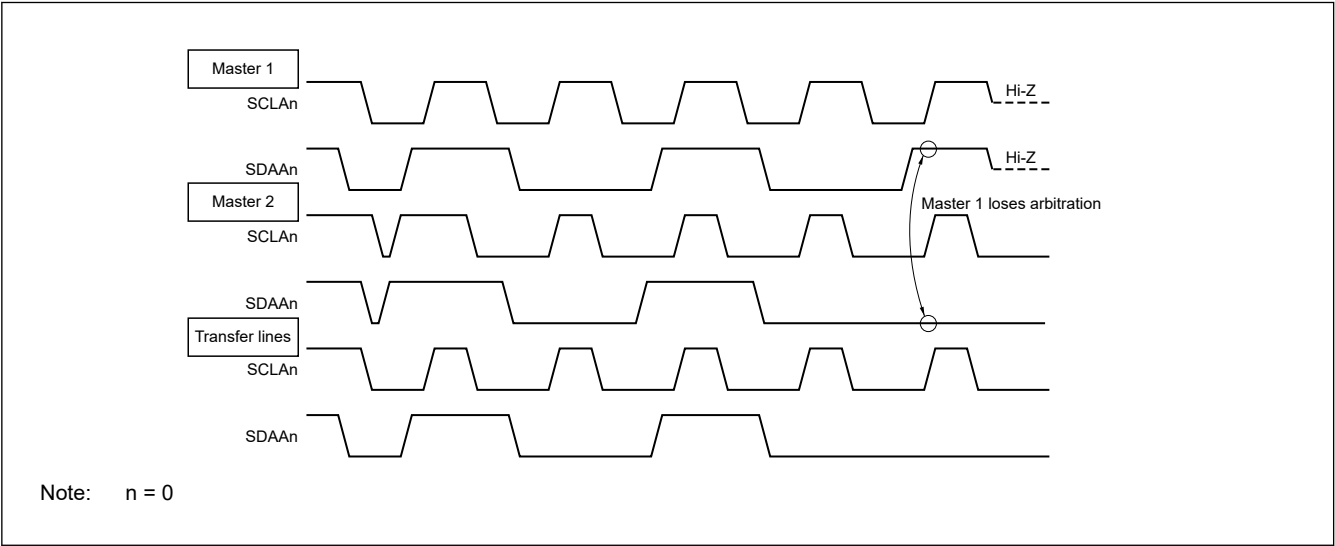


Figure 20.12 Arbitration timing example

Table 20.4 shows the status during arbitration and when interrupt requests are generated.

Table 20.4 Status during arbitration and interrupt request generation timing

Status during arbitration	Interrupt request generation timing
During address transmission	At falling edge of 8th or 9th clock following byte transfer*1
Read/write data after address transmission	
During extension code transmission	
Read/write data after extension code transmission	
During data transmission	
During ACK transfer period after data transmission	
When restart condition is detected during data transfer	
When stop condition is detected during data transfer	When stop condition is generated (when IICCTL00.SPIE = 1)*2
When data is at low level while attempting to generate a restart condition	At falling edge of eighth or ninth clock following byte transfer*1
When stop condition is detected while attempting to generate a restart condition	When stop condition is generated (when IICCTL00.SPIE = 1)*2
When data is at low level while attempting to generate a stop condition	At falling edge of 8th or 9th clock following byte transfer*1
When SCLA0 is at low level while attempting to generate a restart condition	

20.3.14 仲裁

当多个主设备同时产生起始条件（即在 IICS0.STD 位被置为 1 之前，IICCTL00.STT 位被置为 1）时，主设备之间会进行通信，通过调整时钟直到数据出现差异。这种操作称为仲裁。

当其中一个主设备在仲裁中失败时，仲裁失败标志（IICS0.ALD）会在仲裁失败发生的时间点被设置为 1，并且 SCLA0 和 SDAA0 线都会被设置为高阻抗，从而释放总线。

仲裁丢失是通过在下一次中断请求时（例如，检测到停止条件时的第 8 或 9 个时钟周期）使用软件检查 IICS0.ALD = 1 来检测的。

有关中断请求时序的详细信息，请参阅第 20.3.10 节。中断请求信号 (IICA0_TXRXI) 的生成时序和时钟拉伸的控制。

图 20.12 显示了仲裁时序示例。

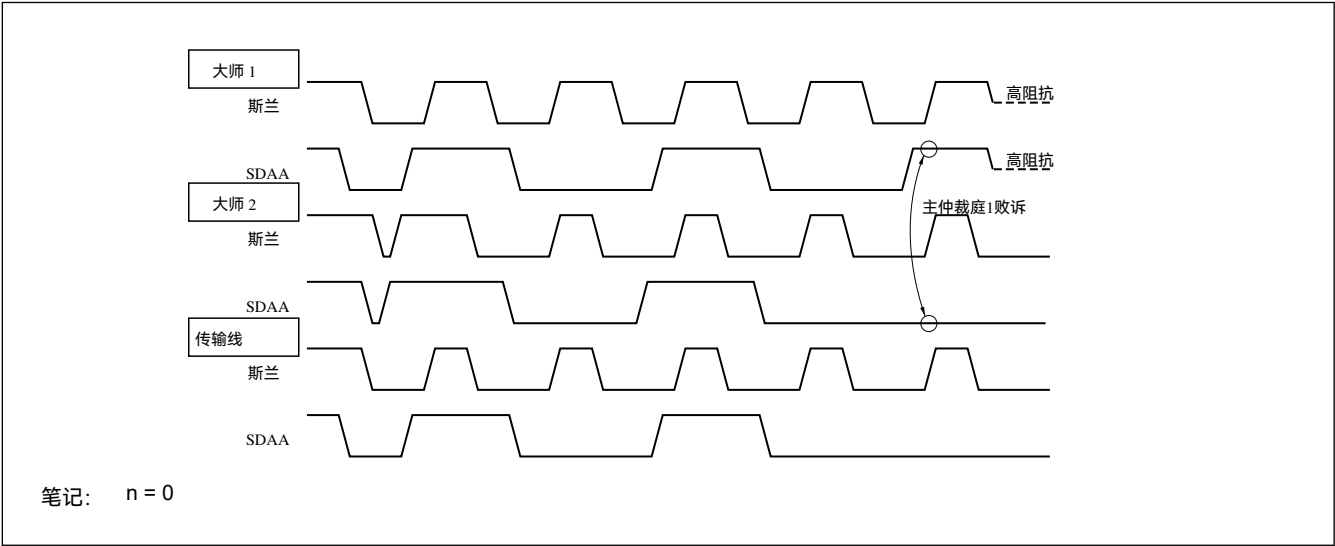


图 20.12 仲裁时序示例

表 20.4 显示了仲裁期间和生成中断请求时的状态。

表 20.4 仲裁期间的状态和中断请求生成时序

仲裁期间的状态	中断请求生成时序
地址传输期间	在字节传输后的第 8 或 9 个时钟周期的下降沿*1
地址传输后读取/写入数据	
在扩展码传输期间	
扩展码传输后读取/写入数据	
数据传输期间	
数据传输后的 ACK 传输期间	
数据传输过程中检测到重启条件时	
数据传输过程中检测到停止条件时	当生成停止条件时（当 IICCTL00.SPIE = 1）*2
当数据量较低时，尝试生成重启条件	在字节传输后的第八个或第九个时钟周期的下降沿*1
当在尝试生成重启条件时检测到停止条件	当生成停止条件时（当 IICCTL00.SPIE = 1）*2
当数据量较低时，尝试生成停止条件	在字节传输后的第 8 或 9 个时钟周期的下降沿*1
当 SCLA0 处于低电平时，尝试生成重启条件	

Note 1. When the IICCTL00.WTIM = 1, an interrupt request occurs at the falling edge of the ninth clock. When IICCTL00.WTIM = 0, the extension code's slave address is received, and an address is received while the all address match function is enabled, an interrupt request occurs at the falling edge of the 8th clock.

Note 2. When there is a chance that arbitration will occur, set IICCTL00.SPIE = 1 for master device operation.

20.3.15 Wakeup Function

The I²C bus slave function is a function that generates an interrupt request signal (IICA0_TXRXI) when the local address is received, an address is received while the all address match function is enabled, or an extension code is received.

This function makes processing more efficient by preventing unnecessary IICA0_TXRXI signal from occurring when addresses do not match while the all address match function is disabled.

When a start condition is detected, wakeup standby state is set. Even a master device that has generated a start condition enters the wakeup standby state while transmitting an address because the master device may become a slave device due to an arbitration loss.

To use the wakeup function in the Software Standby mode, set the IICCTL01.WUP bit to 1. Addresses can be received regardless of the operation clock. An interrupt request signal (IICA0_TXRXI) is also generated when the local address is received, an address is received while the all address match function is enabled, or an extension code is received. Operation returns to normal operation by using an instruction to clear (0) the IICCTL01.WUP bit after this interrupt has been generated.

Table 20.5 shows the step for setting IICCTL01.WUP = 1 and Table 20.6 shows the step for setting IICCTL01.WUP = 0 upon an address match (or when the all address match function is enabled).

Table 20.5 Step when setting IICCTL01.WUP = 1

Step		Process	Detail
Setting IICCTL01.WUP = 1	<1>	Start operation	—
	<2>	Status check	Wait until IICS0 (IICA status register 0) is in the following state: ● MSTS bit = 0 ● STD bit = 0 ● EXC bit = 0 ● COI bit = 0
	<3>	Enable operation of address match wakeup function	Set IICCTL01.WUP bit.
	<4>	Wait	Waits for three cycles of f _{MCK}
	<5>	Stop instruction execution	—

Table 20.6 Flow when setting IICCTL01.WUP = 0 on address match (or when the all address match function is enabled) (including extension code reception)

Step		Process	Detail
Setting IICCTL01.WUP = 0 on address match (or when the all address match function is enabled) (including extension code reception)	<1>	Start operation	Software Standby mode state
	<2>	Interrupt check	Wait until IICA0_TXRXI = 1
	<3>	Disable operation of address match wakeup function	Clear IICCTL01.WUP bit.
	<4>	Wait	Waits for 5 cycles of f _{MCK}
	<5>	Reading IICS0	—
	<6>	Executes next processing	Executes processing corresponding to the operation to be executed after checking the operation state of I ² C bus interface (IICA).

Use the following flows to perform the processing to release the Software Standby mode other than by an interrupt request signal (IICA0_TXRXI) generated from I²C bus interface (IICA).

- When operating next IIC communication as master device: Flow shown in Table 20.7.
- When operating next IIC communication as slave device:
When released by IICA0_TXRXI interrupt: Same as the flow in Table 20.6.

注1. 当IICCTL00.WTIM= 1时，中断请求发生在第九个时钟周期的下降沿。当IICCTL00.WTIM= 0时，接收到扩展代码的从机地址，并且在全地址匹配功能启用的情况下接收到地址，中断请求发生在第八个时钟周期的下降沿。

注 2. 当有可能发生仲裁时，设置 IICCTL00.SPIE= 1以进行主设备操作。

20.3.15唤醒功能

I²C总线从机功能是在接收到本地地址、在启用全地址匹配功能时接收到地址或接收到扩展代码时生成中断请求信号 (IICA0_TXRXI) 的功能。

该功能通过防止在地址不匹配且所有地址匹配功能被禁用时发生不必要的 IICA0_TXRXI 信号，使处理效率更高。

当检测到启动条件时，设备会进入唤醒待机状态。即使是已生成启动条件的主设备，在发送地址时也会进入唤醒待机状态，因为仲裁失败可能导致主设备变为从设备。

要在软件待机模式下使用唤醒功能，请将 IICCTL01.WUP 位设置为 1。此时，无论运行时钟如何，都可以接收地址。此外，当接收到本地地址、在启用全地址匹配功能的情况下接收到地址或接收到扩展码时，也会生成中断请求信号 (IICA0_TXRXI)。中断生成后，使用清除 (0) IICCTL01.WUP 位的指令恢复正常运行。

表 20.5 显示了设置IICCTL01.WUP = 1的步骤，表 20.6 显示了在地址匹配时（或启用所有地址匹配功能时）设置 IICCTL01.WUP= 0的步骤。

表 20.5 设置 IICCTL01.WUP= 1 时的步骤

步		过程	细节
设置 IICCTL01.WUP = 1	<1>	开始操作	—
	<2>	状态检查	等待 IICS0（IICA 状态寄存器 0）处于以下状态: ●MSTS 位= 0 ●标准位= 0 ●EXC 位= 0 ●COI 位= 0
	<3>	启用地址匹配唤醒功能	设置 IICCTL01.WUP 位。
	<4>	等待	等待三个周期的 fMCK
	<5>	停止指令执行	—

表 20.6 当设置 IICCTL01.WUP= 0进行地址匹配时（或启用所有地址匹配功能时）（包括分机号接收）的流程

步		过程	细节
设置 IICCTL01.WUP = 0地址匹配时（或当所有地址匹配功能已启用）（包括接收分机号）	<1>	开始操作	软件待机模式状态
	<2>	中断检查	等待 IICA0_TXRXI= 1
	<3>	禁用地址匹配唤醒功能	清除 IICCTL01.WUP 位。
	<4>	等待	等待 5 个 fMCK 周期
	<5>	读取 IICS0	—
	<6>	执行下一个处理	检查I ² C总线接口（IICA）的操作状态后，执行与要执行的操作对应的处理。

除了通过从I²C总线接口 (IICA) 生成的中断请求信号 (IICA0_TXRXI) 来释放软件待机模式之外，还应使用以下流程来执行处理。

- 当作为主设备操作下一个 IIC 通信时：流程如表 20.7 所示。
- 当下次以从设备身份进行 IIC 通信时：
当通过 IICA0_TXRXI 中断释放时：与表 20.6 中的流程相同。

When released by other than IICA0_TXRXI interrupt: Wait for IICA0_TXRXI interrupt with IICCTL01.WUP left set to 1.

Table 20.7 When operating as master device after releasing Software Standby mode other than by IICA0_TXRXI

Step		Process	Detail
When operating as master device after releasing Software Standby mode other than by IICA0_TXRXI	<1>	Start operation	—
	<2>	Enable generation of interrupt request	Set IICCTL00.SPIE bit.
	<3>	Enable operation of address match wakeup function	Set IICCTL01.WUP bit.
	<4>	Wait	Waits for 3 cycles of f _{MCK}
	<5>	Stop instruction execution	Software standby state
	<6>	Releasing Software Standby mode	Releases Software Standby mode by an interrupt other than IICA0_TXRXI
	<7>	Disable operation of address match wakeup function	Clear IICCTL01.WUP bit.
	<8>	Interrupt check	Wait until IICA0_TXRXI = 1
	<9>	Reading IICS0	—
	<10>	Executes next processing	Executes processing corresponding to the operation to be executed after checking the operation state of I ² C bus interface (IICA).

20.3.16 Communication Reservation

(1) When communication reservation function is enabled (IICRSV bit) of IICA flag register 0 (IICF0) = 0

To start master communications when not currently using a bus, a communication reservation can be made to enable transmission of a start condition when the bus is released. There are two modes under which the bus is not used.

- When arbitration results in neither master device nor slave device operation
- While the all address match function is disabled, when an extension code is received and slave device operation is disabled (ACK is not returned and the bus was released by setting LREL bit of IICA control register 00 (IICCTL00) to 1 and exiting from communication)

When setting bit 1 (STT) of the IICCTLn0 register while in a non-participatory state on the bus, after the bus is released (upon detecting a stop condition), it automatically generates a start condition and enters communication standby status.

Setting the SPIE bit of the IICCTLn0 register to 1, and detecting the release of the bus upon interrupt request (IICAn_TXRXI) (detecting a stop condition), after writing the address to the IICA shift register n (IICAn), automatically initiates communication as a master. Data written to the IICA0 register before the stop condition is detected is invalid.

When the IICCTL00.STT bit has been set to 1, the operation mode (as start condition or as communication reservation) is determined according to the bus status.

- If the bus has been released a start condition is generated
- If the bus has not been released (communication standby status) ... communication reservation

Check whether the communication reservation operates or not using the IICS0.MSTS bit after the IICCTL00.STT bit is set to 1 and the wait time elapses.

Use software to secure the wait time calculated by the following expression.

Wait time from setting IICCTL00.STT = 1 to checking the IICS0.MSTS flag:

(IICWL0 setting value + IICWH0 setting value + 4) / f_{MCK} + t_F × 2

Note: IICWL0: IICA low-level width setting register 0
IICWH0: IICA high-level width setting register 0

当由 IICA0_TXRXI 中断以外的其他中断释放时：等待 IICA0_TXRXI 中断，并将 IICCTL01.WUP 左设置为 1。

表 20.7 当在解除软件待机模式后作为主设备运行时（除以下方式外）：IICA0_TXRXI

步		过程	细节
当作为主设备发布软件后 除 IICA0_TXRXI 之外的待机模式	<1>	开始操作	—
	<2>	启用中断请求的生成	设置 IICCTL00.SPIE 位。
	<3>	启用地址匹配唤醒功能	设置 IICCTL01.WUP 位。
	<4>	等待	等待 3 个 f _{MCK} 周期
	<5>	停止指令执行	软件待机状态
	<6>	释放软件待机模式	通过除 IICA0_TXRXI 之外的中断释放软件待机模式
	<7>	禁用地址匹配唤醒功能	清除 IICCTL01.WUP 位。
	<8>	中断检查	等待 IICA0_TXRXI= 1
	<9>	读取 IICS0	—
	<10>	执行下一个处理	检查 I ² C 总线接口 (IICA) 的操作状态后，执行与要执行的操作对应的处理。

20.3.16 通信预留

(1) 当 IICA 标志寄存器 0 的通信预留功能（IICRSV 位）启用时(IICF0) = 0

当总线当前未被使用时，若要启动主通信，可以进行通信预留，以便在总线释放时发送启动条件。总线未被使用的情况有两种。

- 当仲裁结果既不导致主设备也不导致从设备操作时
- 当所有地址匹配功能被禁用时，收到扩展码后，从设备操作被禁用（未返回ACK，并且通过将IICA控制寄存器00（IICCTL00）的LREL位设置为1并退出通信来释放总线）。

当总线处于非参与状态时，如果设置 IICCTLn0 寄存器的第 1 位（STT），则在总线释放后（检测到停止条件时），它会自动生成启动条件并进入通信待机状态。

将 IICCTLn0 寄存器的 SPIE 位设置为 1，并在中断请求 (IICAn_TXRXI) 时检测到总线释放（即检测到停止条件），然后将地址写入 IICA 移位寄存器 n (IICAn)，即可自动启动主站通信。在检测到停止条件之前写入 IICA0 寄存器的数据无效。

当 IICCTL00.STT 位设置为 1 时，操作模式（作为启动条件或通信预留）根据总线状态确定。

- 如果公交车已释放……，则会生成启动条件。
- 如果巴士尚未放行（通讯待命状态）……通讯预订

在 IICCTL00.STT 位设置为 1 且等待时间过去后，使用 IICS0.MSTS 位检查通信预留是否运行。

使用软件确保以下表达式计算出的等待时间的安全。
从设置 IICCTL00.STT = 1到检查 IICS0.MSTS 标志的等待时间：
(IICWL0 设置值+ IICWH0设置值+ 4) / f_{MCK} + t_F × 2

注：IICWL0: IICA 低级宽度设置寄存器 0
IICWH0: IICA 高级宽度设置寄存器 0

t_F : SDAA0 and SCLAn signal falling times
 f_{MCK} : IICA operation clock frequency

Figure 20.13 shows the communication reservation timing.

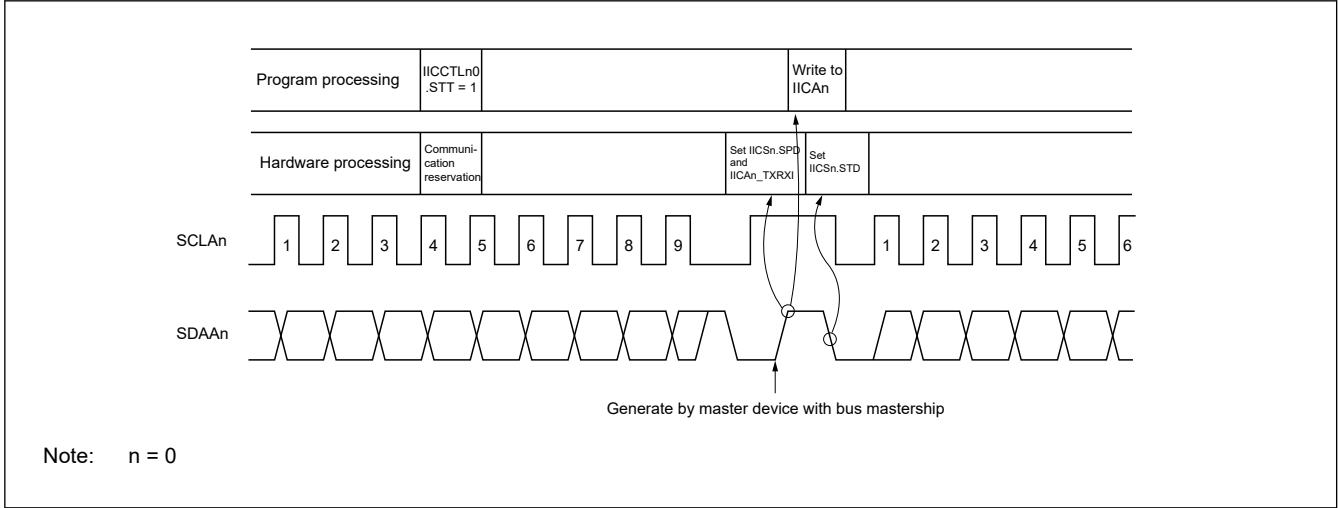


Figure 20.13 Communication reservation timing

Communication reservations are accepted with the timing shown in Figure 20.14. After STD bit of the IICA status register 0 (IICS0) is set to 1, a communication reservation can be made by setting STT bit of IICA control register 00 (IICCTL00) to 1 before a stop condition is detected.

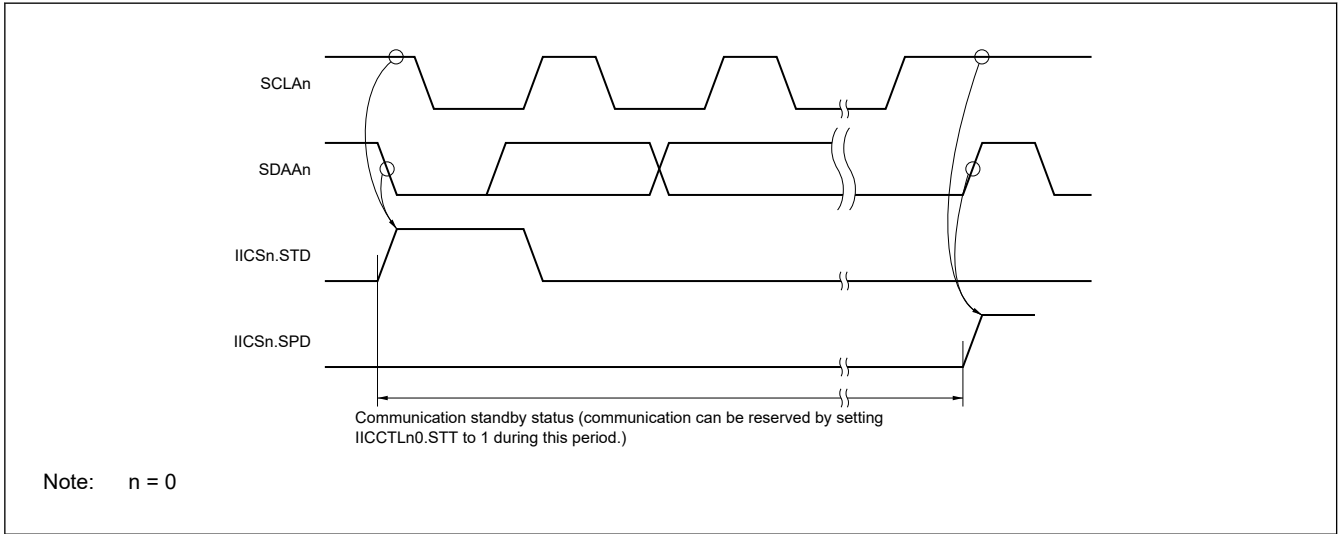


Figure 20.14 Timing for accepting communication reservations

Figure 20.15 shows the communication reservation protocol.

t_F : SDAA0 和 SCLAn 信号下降时间
 f_{MCK} : IICA 操作时钟频率

图 20.13 显示了通信预留时间。

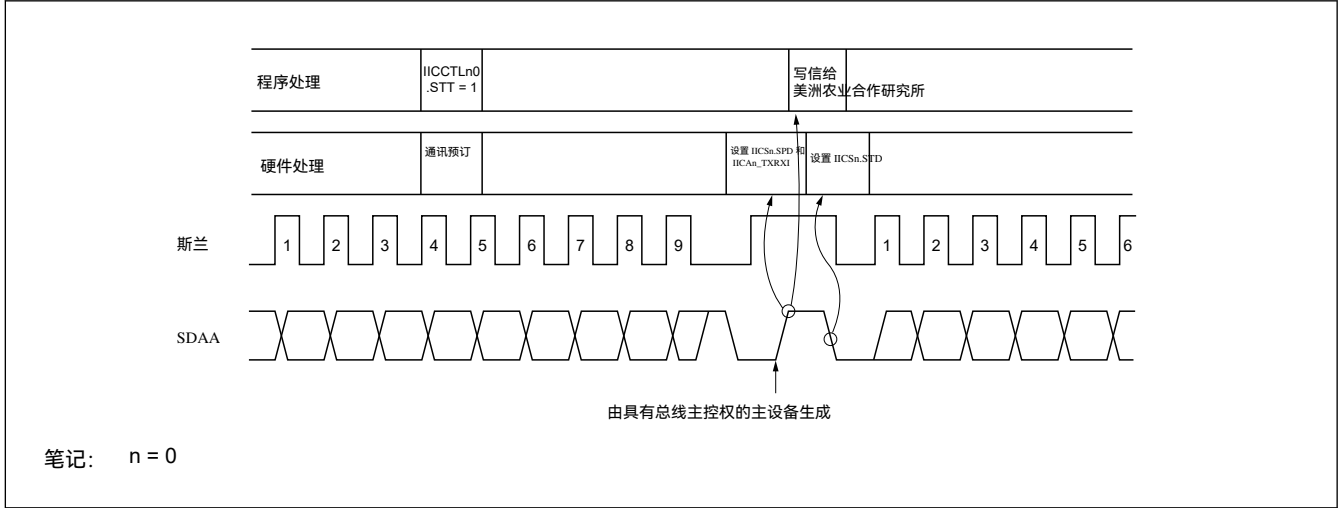


图 20.13 通信预留时间

通信预留的接受时间如图 20.14 所示。在 IICA 状态寄存器 0 (IICS0) 的 STD 位设置为 1 之后，在检测到停止条件之前，可以通过将 IICA 控制寄存器 00 (IICCTL00) 的 STT 位设置为 1 来进行通信预留。

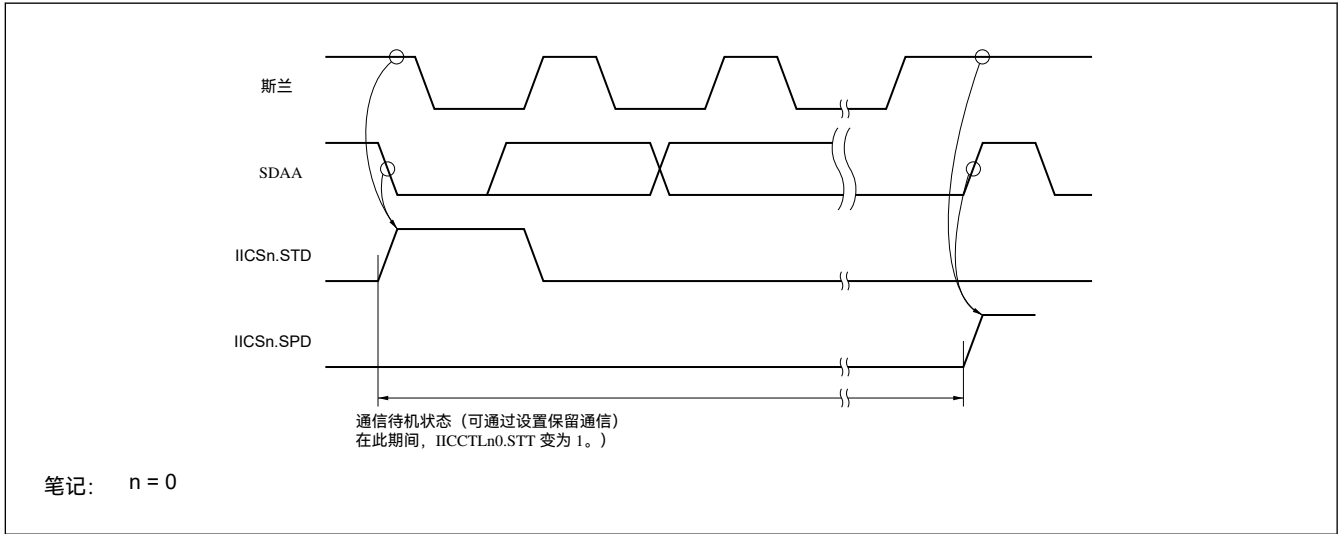


图 20.14 接受通信预订的时间

图 20.15 显示了通信预留协议。

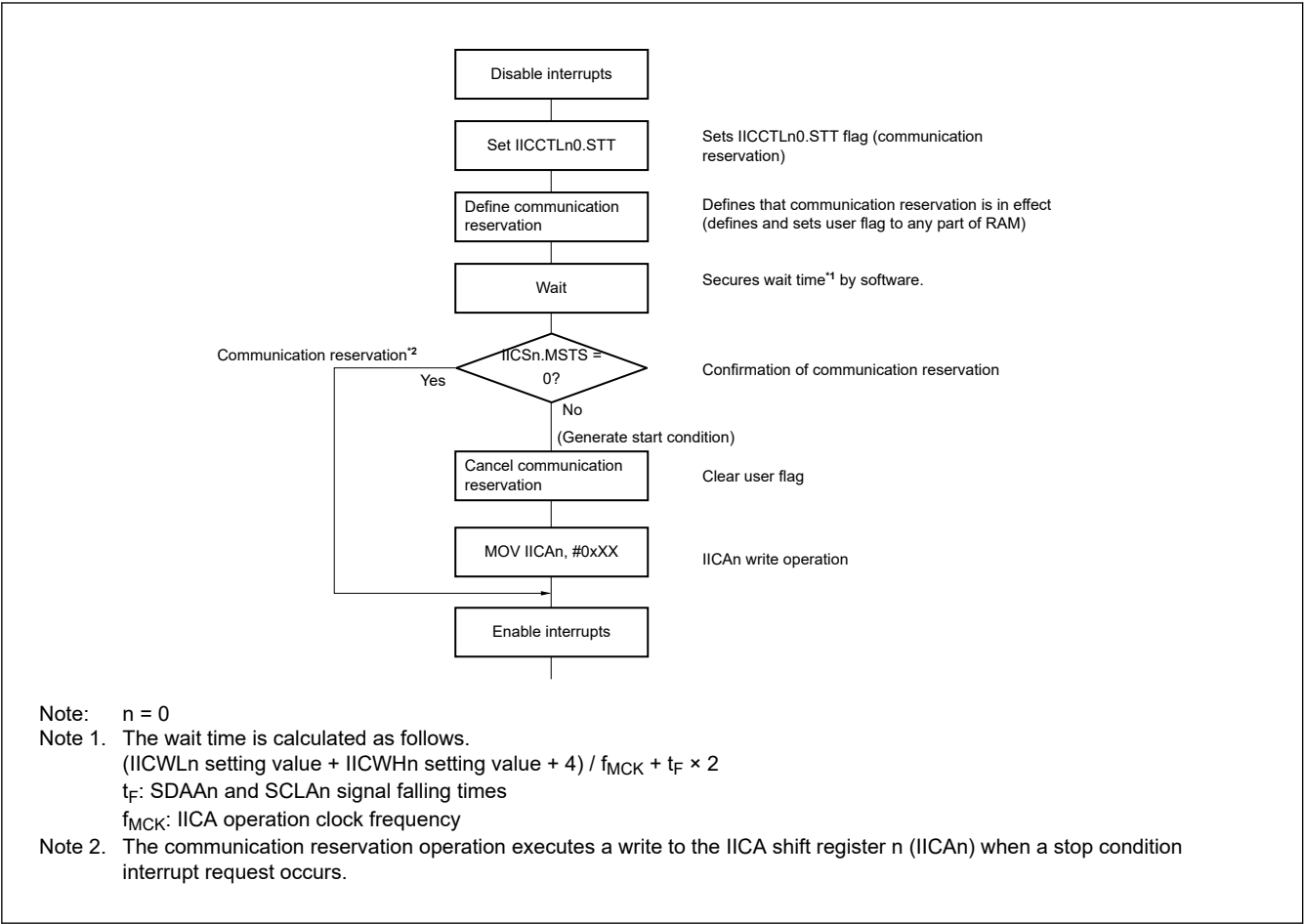


Figure 20.15 Communication reservation protocol

(2) When communication reservation function is disabled (IICF0.IICRSV = 1)

When STT bit of IICA control register 00 (IICCTL00) is set to 1 when the bus is in communication and is not participating in this communication, this request is rejected and a start condition is not generated. In this case, non-participation on the bus includes the following two states.

- When arbitration results in neither master device nor slave device operation
- While the all address match function is disabled, when an extension code is received and slave device operation is disabled (ACK is not returned and the bus was released by setting LREL bit of the IICCTL00 register to 1 and exiting from communication)

To confirm whether the start condition was generated or request was rejected, check IICF0.STCF bit. It takes up to 5 cycles of f_{MCK} until the IICF0.STCF bit is set to 1 after setting IICCTL00.STT = 1. Therefore, secure the time by software.

20.3.17 Usage Notes

1. When IICF0.STCEN = 0

The generation of a start condition is based upon the detection of a stop condition. Therefore, changing from a mode in which no stop condition has been detected, first generate a stop condition to release the bus.

Immediately after I²C operation is enabled (IICCTL00.IICE = 1), the bus communication status (IICF0.IICBSY = 1) is recognized regardless of the actual bus status. When changing from a mode in which no stop condition has been detected to a master communication mode, first generate a stop condition to release the bus, then perform master communication.

When using multiple master devices, it is not possible to perform master communication when the bus has not been released (when a stop condition has not been detected).

Use the following sequence for generating a stop condition.

<1> Set IICA control register 01 (IICCTL01).

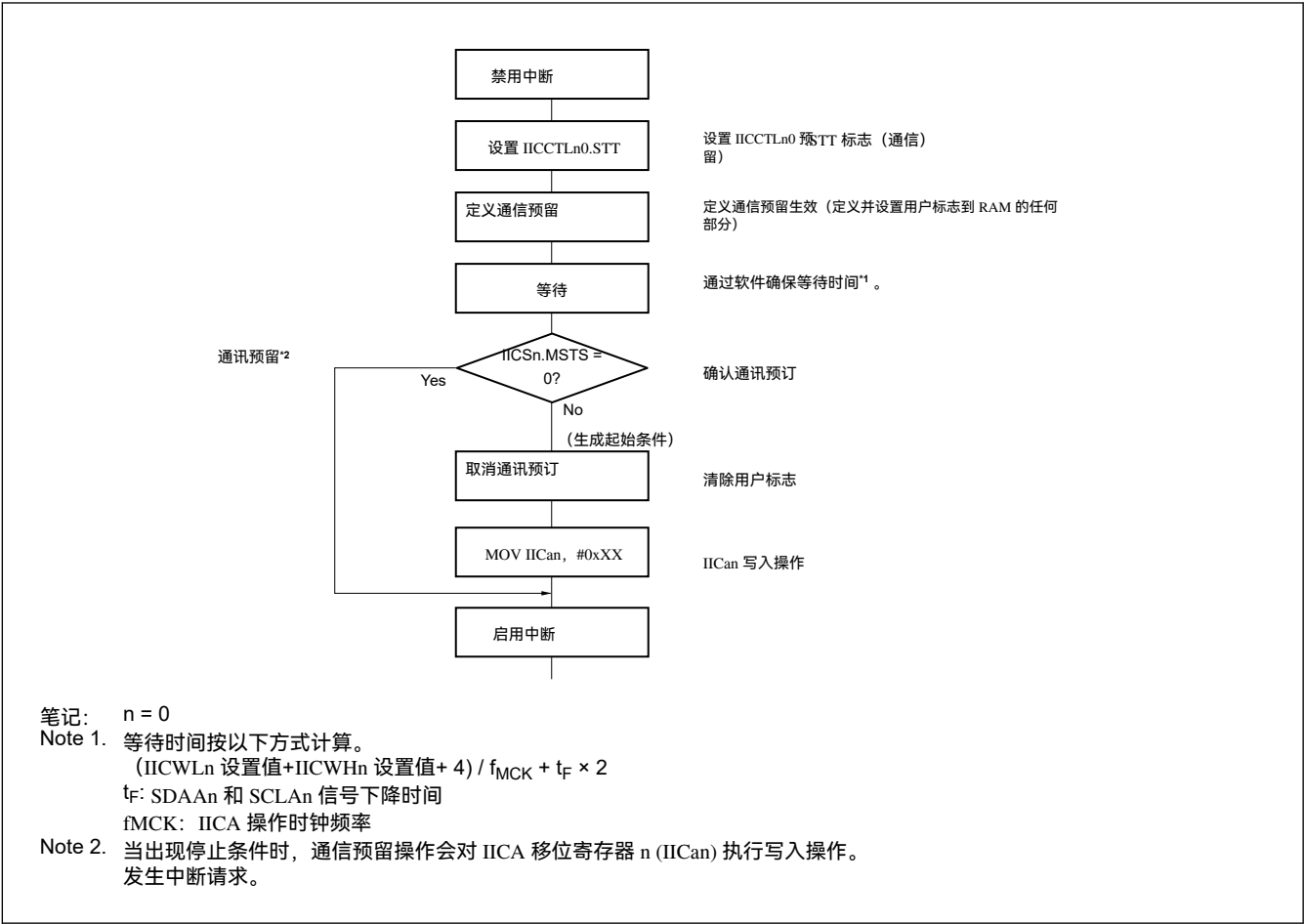


图 20.15 通信预留协议

(2) 当通信预留功能被禁用时(IICF0.IICRSV = 1)

当总线处于通信状态但未参与通信时, 如果 IICA 控制寄存器 00 (IICCTL00) 的 STT 位被设置为 1, 则此请求将被拒绝, 并且不会生成启动条件。在这种情况下, 总线上的非参与状态包括以下两种情况。

- 当仲裁结果既不导致主设备也不导致从设备操作时
- 当所有地址匹配功能被禁用时, 收到扩展码后, 从设备操作被禁用 (未返回ACK, 并且通过将IICCTL00寄存器的LREL位设置为1并退出通信来释放总线)。

为了确认启动条件是否已生成或请求是否被拒绝, 请检查 IICF0.STCF 位。在设置 IICCTL00.STT = 1后, 最多需要 5 个 f_{MCK} 周期, IICF0.STCF 位才会被置为 1。因此, 需要通过软件来保证时间安全。

20.3.17 使用说明

1. 当 IICF0.STCEN = 0

启动条件的生成基于停止条件的检测。因此, 从未检测到停止条件的模式切换到启动模式时, 首先需要生成一个停止条件以释放总线。

在启用 I²C 操作(IICCTL00.IICE = 1)后, 无论总线实际状态如何, 都会立即识别总线通信状态(IICF0.IICBSY = 1)。当从未检测到停止条件的模式切换到主通信模式时, 首先生成停止条件以释放总线, 然后再执行主通信。

当使用多个主设备时, 如果总线尚未释放 (尚未检测到停止条件), 则无法执行主设备通信。

使用以下序列生成停止条件。

<1> 设置 IICA 控制寄存器 01 (IICCTL01)。

- <2> Set IICE bit of IICA control register 00 (IICCTL00) to 1.
- <3> Set SPT bit of the IICCTL00 register to 1.
2. When IICF0.STCEN = 1
The generation of a start condition is independent with stop condition detection, so firstly it is necessary to confirm with the bus has been released.
Immediately after I²C operation is enabled (IICCTL00.IICE = 1), the bus released status (IICF0.IICBSY = 0) is recognized regardless of the actual bus status. To generate the first start condition (IICCTL00.STT = 1), it is necessary to confirm that the bus has been released, so as to not disturb other communications.
3. If other I²C communications are already in progress
If I²C operation is enabled and the device participates in communication already in progress when the SDAA0 pin is low and the SCLA0 pin is high, the IICA recognizes that the SDAA0 pin has gone low (detects a start condition). If the value on the bus at this time can be recognized as an extension code or the all address match function is enabled, ACK is returned, but this interferes with other I²C communications. To avoid this, start the IICA in the following sequence.
<1> Clear SPIE bit of the IICCTL00 register to 0 to disable generation of an interrupt request signal (IICA0_TXRXI) when the stop condition is detected.
<2> Set IICE bit of the IICCTL00 register to 1 to enable the operation of the IICA.
<3> Wait for detection of the start condition.
<4> Set LREL bit of the IICCTL00 register to 1 before ACK is returned (4 to 72 cycles of f_{MCK} after setting the IICCTL00.IICE bit to 1), to forcibly disable detection.
4. Setting the IICCTL00.STT and IICCTL00.SPT bits again after they are set and before they are cleared to 0 is prohibited.
5. When transmission is reserved, set the IICCTL00.SPIE bit to 1 so that an interrupt request is generated when the stop condition is detected. Transfer is started when communication data is written to the IICA shift register 0 (IICA0) after the interrupt request is generated. Unless the interrupt is generated when the stop condition is detected, the device stops in the wait state because the interrupt request is not generated when communication is started. However, it is not necessary to set the IICCTL00.SPIE bit to 1 when the IICS0.MSTS bit is detected by software.

20.3.18 Communication Operations

The following shows three operation procedures with the flowchart.

1. Master device operation in single-master system
The flowchart when using this product as the master device in a single master system is shown in [Figure 20.16](#). This flowchart is broadly divided into the initial settings and communication processing. Execute the initial settings at startup. If communication with the slave device is required, prepare the communication and then execute communication processing.
2. Master device operation in multi-master system
In the I²C bus multimaster system, whether the bus is released or used cannot be judged by the I²C bus specifications when a device takes part in a communication. Here, when data and clock are at a high level for a certain period (1 frame), this product takes part in a communication with bus released state.
This flowchart is broadly divided into the initial settings, communication waiting, and communication processing. The processing when this product loses in arbitration and is specified as the slave device is omitted in [Figure 20.17](#), and only the processing as the master device is shown. Execute the initial settings at startup to take part in a communication. Then, wait for the communication request as the master device or wait for the specification as the slave device. The actual communication is performed in the communication processing, and it supports the transmission and reception with the slave device and the arbitration with other masters devices.
3. Slave device operation
An example of when this product is used as the I²C bus slave device is shown in [Figure 20.21](#) and [Figure 20.22](#). When used as the slave, operation is started by an interrupt. Execute the initial settings at startup, then wait for the IICA0_TXRXI interrupt occurrence (communication waiting). When an IICA0_TXRXI interrupt occurs, the communication status is judged and its result is passed as a flag over to the main processing.
By checking the flags, necessary communication processing is performed.

(1) Master device operation in single-master system

Conform to the specifications of the product that is communicating, with respect to the transmission and reception formats.

- <2>将 IICA 控制寄存器 00 (IICCTL00) 的 IICE 位设置为 1。
- <3>将 IICCTL00 寄存器的 SPT 位设置为 1。
2. 当IICF0.STCEN = 1
启动条件的生成与停止条件的检测无关，因此首先需要确认总线已释放。
- 在启用I²C 操作(IICCTL00.IICE = 1)后，无论总线实际状态如何，都会立即识别总线已释放状态(IICF0.IICBSY = 0)。为了生成第一个启动条件(IICCTL00.STT = 1)，必须确认总线已释放，以免干扰其他通信。
3. 如果其他I²C通信已经在进行中
如果启用了I²C操作，并且设备在 SDAA0 引脚为低电平、SCLA0 引脚为高电平时参与正在进行的通信，则 IIC A 会识别到 SDAA0 引脚已变为低电平（检测到启动条件）。如果此时总线上的值可以被识别为扩展码或启用了全地址匹配功能，则会返回 ACK，但这会干扰其他I²C通信。为避免这种情况，请按以下顺序启动 IICA。<1> 将 IICCTL00 寄存器的 SPIE 位清除为 0，以在检测到停止条件时禁用中断请求信号 (IICA0_TXRXI) 的生成。
- <2>将 IICCTL00 寄存器的 IICE 位设置为 1，以启用 IICA 的操作。
- <3>等待检测到启动条件。
- <4>在返回 ACK 之前（在将 IICCTL00.IICE 位设置为 1 之后的 4 到 72 个周期内），将 IICCTL00 寄存器的 LREL 位设置为 1，以强制禁用检测。
4. 禁止在设置 IICCTL00.STT 和 IICCTL00.SPT 位之后、将其清除为 0 之前再次设置它们。
5. 当传输保留时，将 IICCTL00.SPIE 位设置为 1，以便在检测到停止条件时生成中断请求。中断请求生成后，当通信数据写入 IICA 移位寄存器 0 (IICA0) 时，传输开始。如果在检测到停止条件时未生成中断，则设备将停止在等待状态，因为在通信开始时未生成中断请求。但是，当软件检测到 IICS0.MSTS 位时，无需将 IICCTL00.SPIE 位设置为 1。

20.3.18 通信操作

下面用流程图展示了三个操作步骤。

1. 单主控系统中主控设备的操作
当使用该产品作为单个主系统中的主设备时的流程图如图 20.16 所示。
此流程图大致分为初始设置和通信处理两部分。启动时执行初始设置。如果需要与从设备通信，则准备通信，然后执行通信处理。
2. 多主控系统中主控设备的操作
在I²C总线多主系统中，当设备参与通信时，无法通过I²C总线规范来判断总线是已释放还是已使用。此处，当数据和时钟信号在一定时间内（1帧）保持高电平时，该产品以总线已释放状态参与通信。
- 此流程图大致分为初始设置、通信等待和通信处理三个部分。图 20.17 省略了该产品在仲裁中落败且被指定为从设备时的处理过程，仅显示了其作为主设备时的处理过程。启动时执行初始设置以参与通信。
- 然后，主设备等待通信请求，从设备等待通信规范。实际通信在通信处理程序中执行，支持与从设备进行收发通信，以及与其他主设备进行仲裁。
3. 从设备操作
图 20.21 和图 20.22 展示了该产品作为I²C总线从设备使用的示例。作为从设备时，操作由中断启动。启动时执行初始设置，然后等待 IICA0_TXRXI 中断发生（通信等待）。当发生 IICA0_TXRXI 中断时，判断通信状态，并将结果作为标志传递给主处理程序。
- 通过检查标志位，执行必要的通信处理。

(1) 单主控系统中的主控设备操作

在传输和接收格式方面，应符合通信产品的规格。

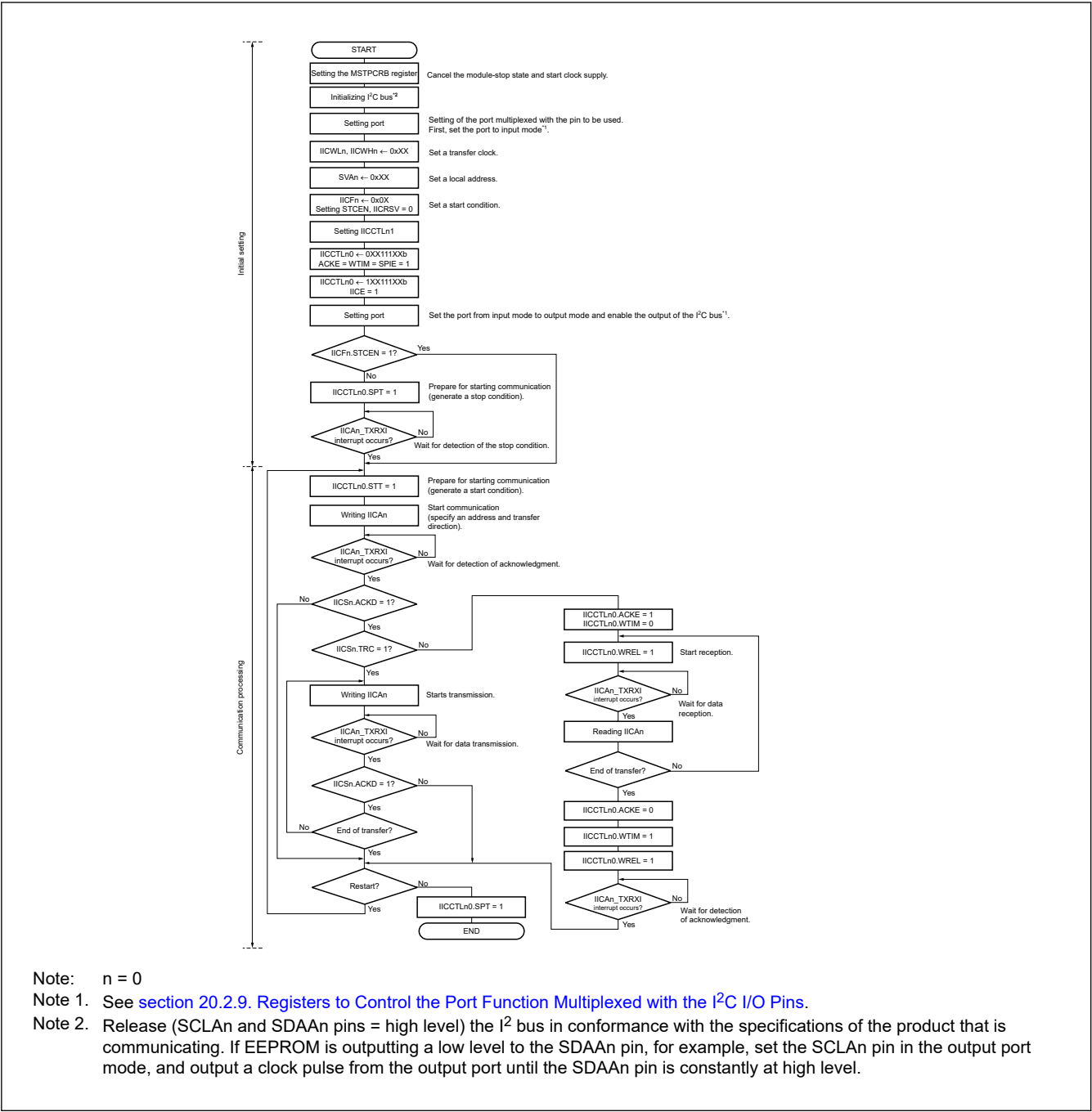


Figure 20.16 Master device operation in single-master system

(2) Master device operation in multi-master system

Conform to the specifications of the product that is communicating, with respect to the transmission and reception formats.

To use the device as a master device in a multi-master system, read the IICS0.MSTS bit each time interrupt IICA0_TXRRI occurred to check the arbitration result.

To use the device as a slave device in a multi-master system, check the status by using the IICA status register 0 (IICS0) and IICA flag register 0 (IICF0) each time interrupt IICA0_TXRRI occurred, and determine the processing to be performed next.

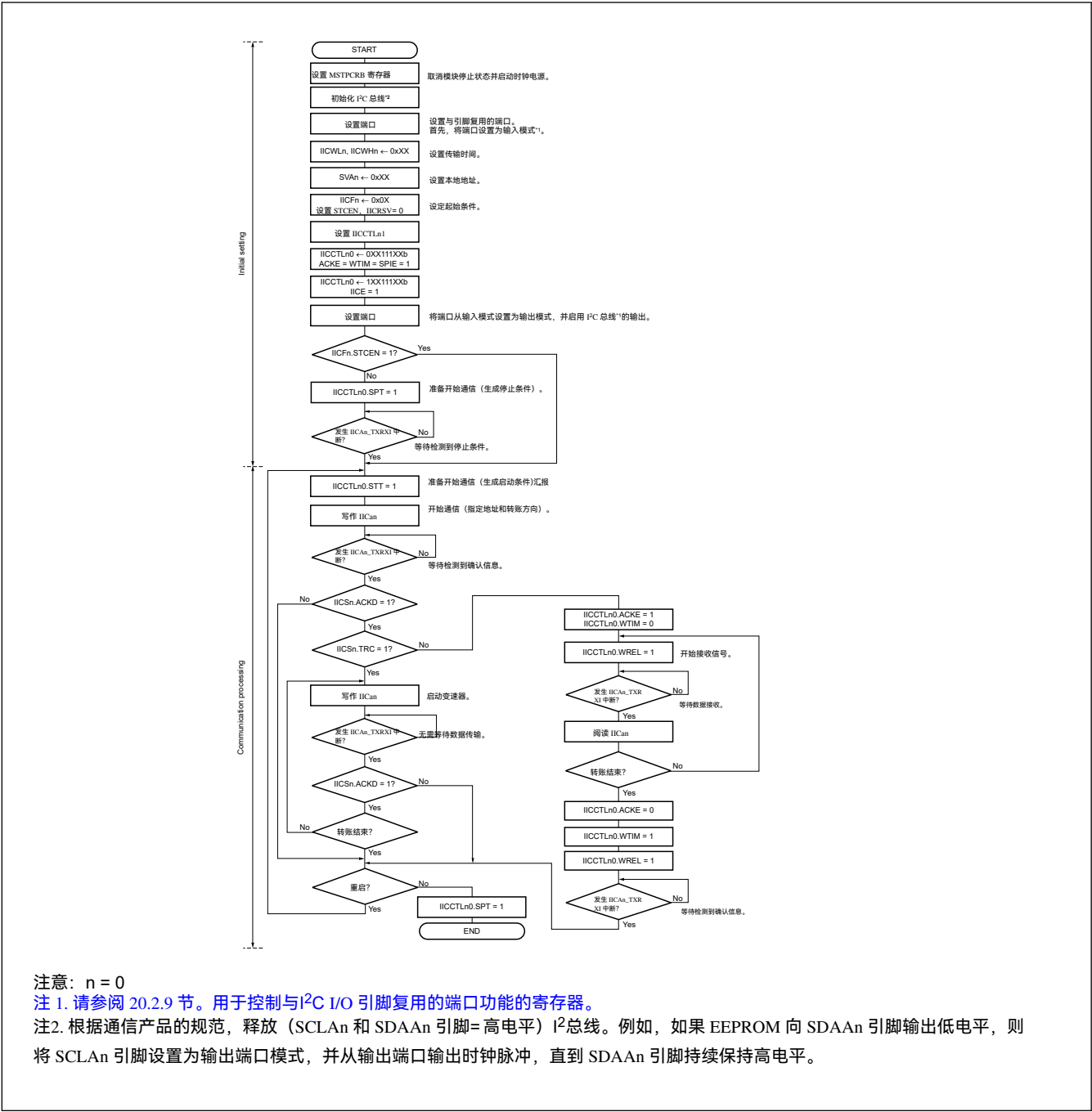


图 20.16 单主控系统中的主控设备操作

(2) 多主控系统中的主控设备操作

传输和接收格式应符合通信产品的规格。若要将该设备用作多主系统中的主设备, 则每次发生中断 IICA0_TXRRI 时, 都应读取 IICS0.MSTS 位以检查仲裁结果。

要将该设备用作多主系统中的从设备, 每次发生中断 IICA0_TXRRI 时, 使用 IICA 状态寄存器 0 (IICS0) 和 IICA 标志寄存器 0 (IICF0) 检查状态, 并确定接下来要执行的处理。

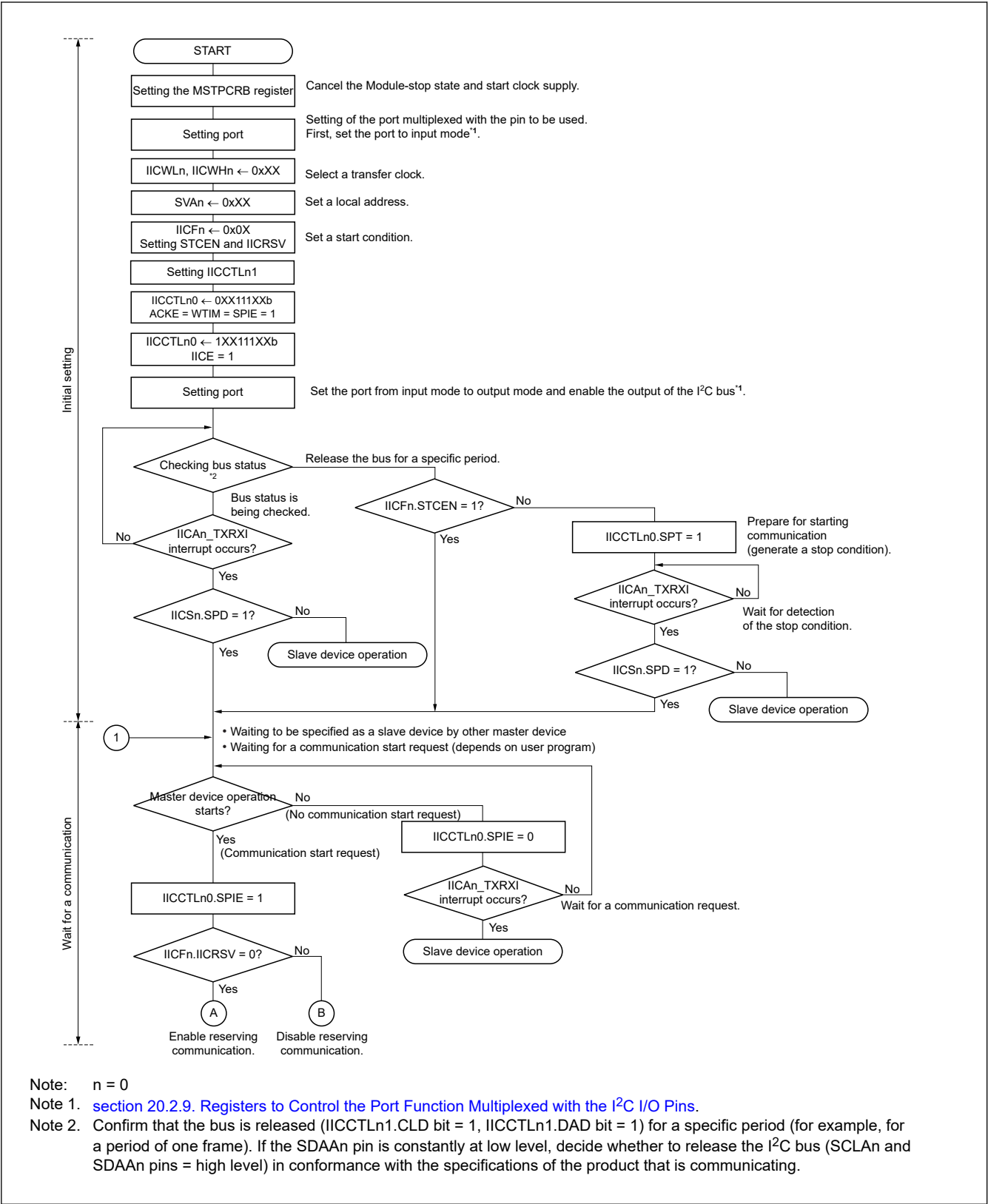


Figure 20.17 Master device operation in multi-master system (1/3)

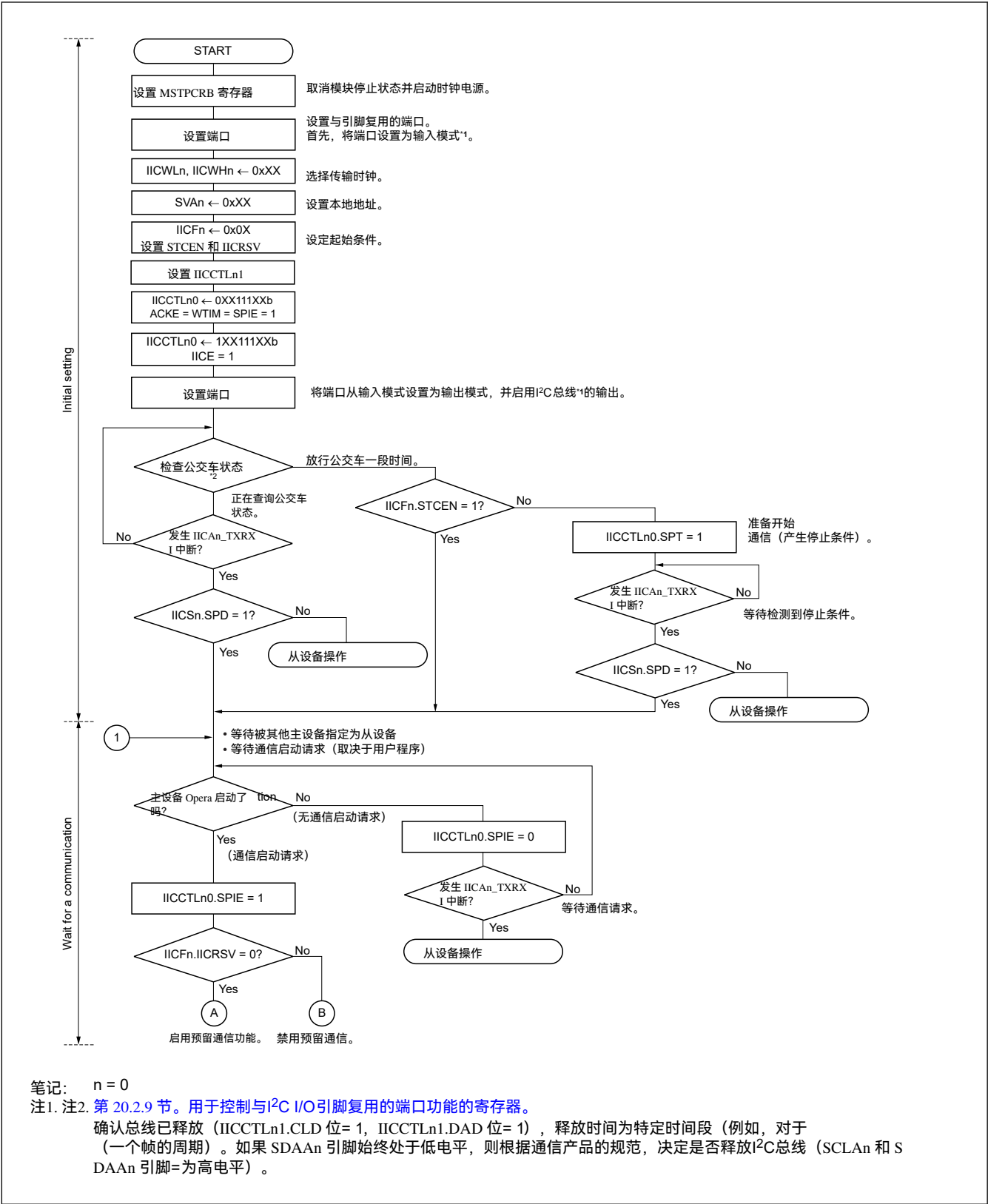


图 20.17 多主设备系统中的主设备操作 (1/3)

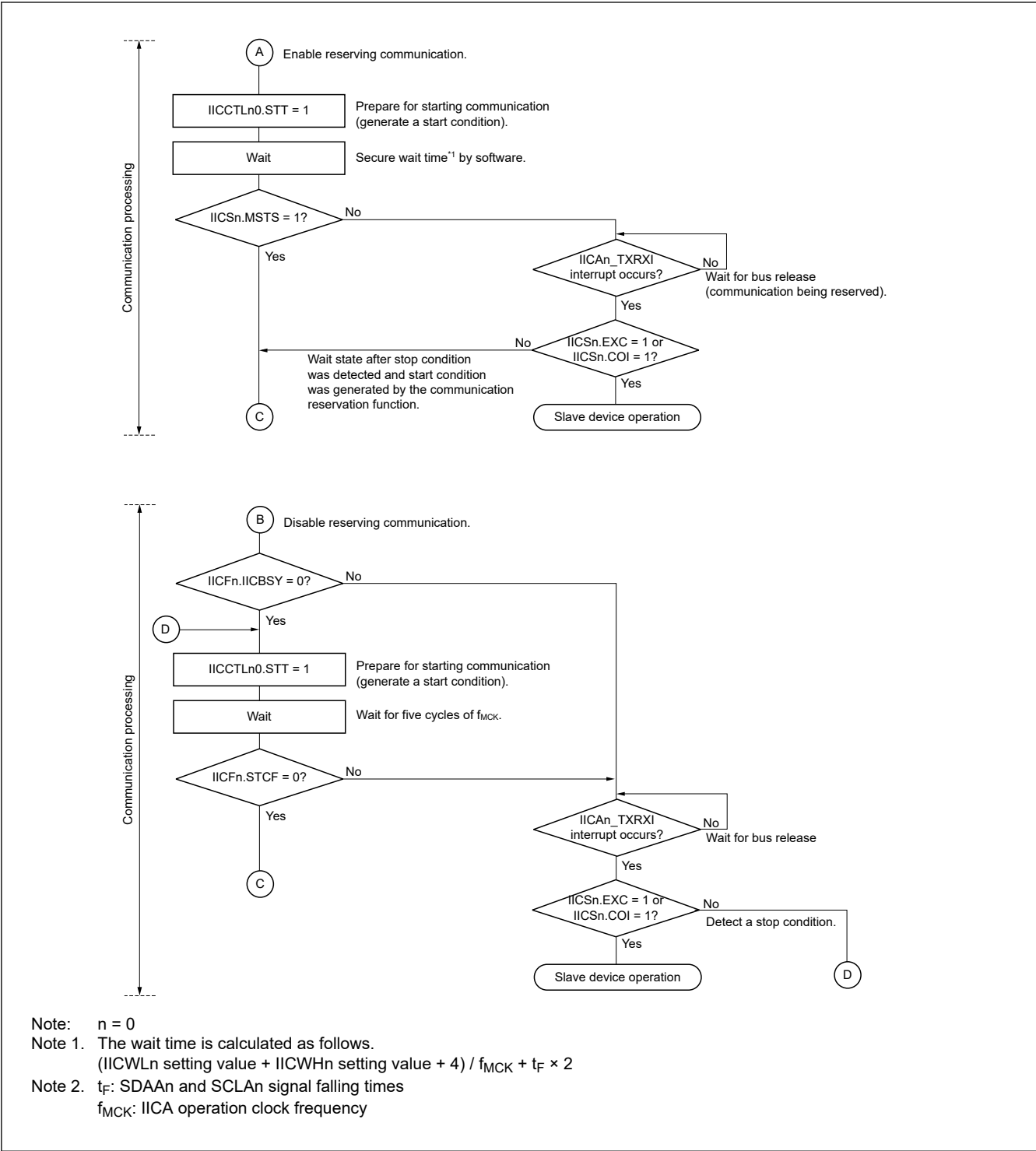


Figure 20.18 Master device operation in multi-master system (2/3)

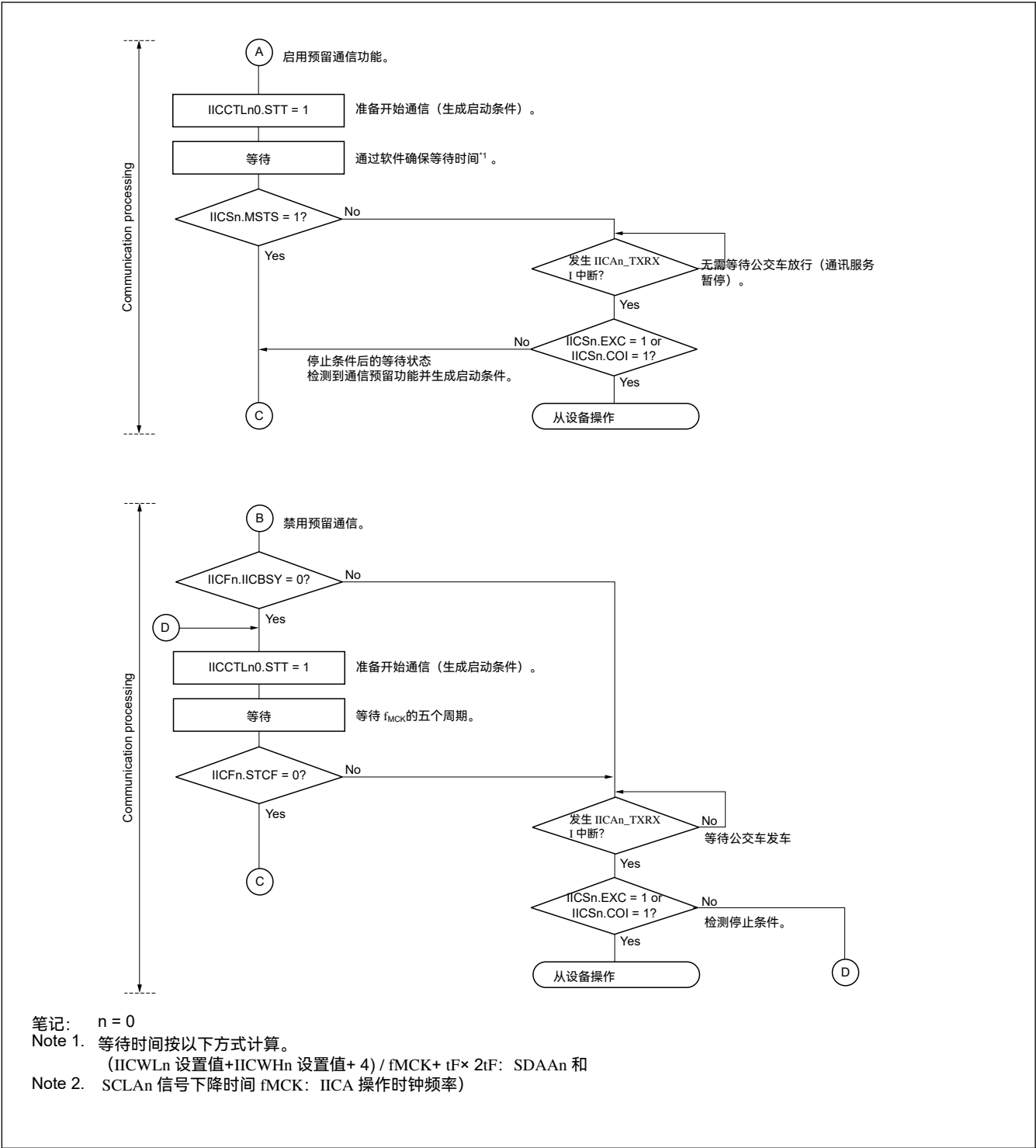


图 20.18 多主设备系统中的主设备操作 (2/3)

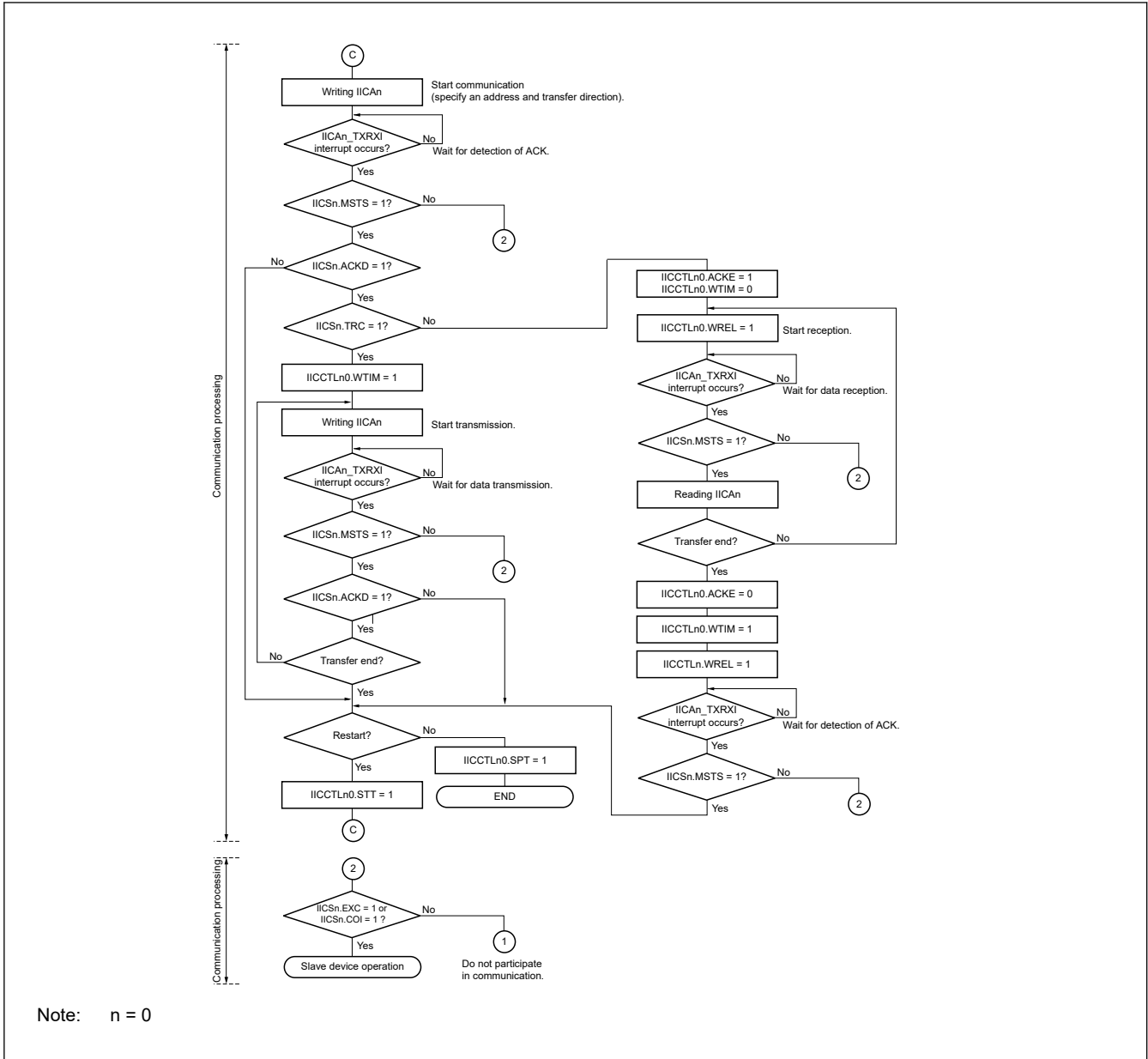


Figure 20.19 Master device operation in multi-master system (3/3)

(3) Slave device operation

The processing procedure of the slave device operation is as follows.

Basically, the slave device operation is event-driven. Therefore, processing by the IICA0_TXRXI interrupt (processing that must substantially change the operation status such as detection of a stop condition during communication) is necessary.

In the following explanation, it is assumed that the all address match function is disabled and the extension code is not supported for data communication. It is also assumed that the IICA0_TXRXI interrupt processing only performs status transition processing, and that actual data communication is performed by the main processing.

Figure 20.20 shows an interface configuration with the main processor in slave device operation.

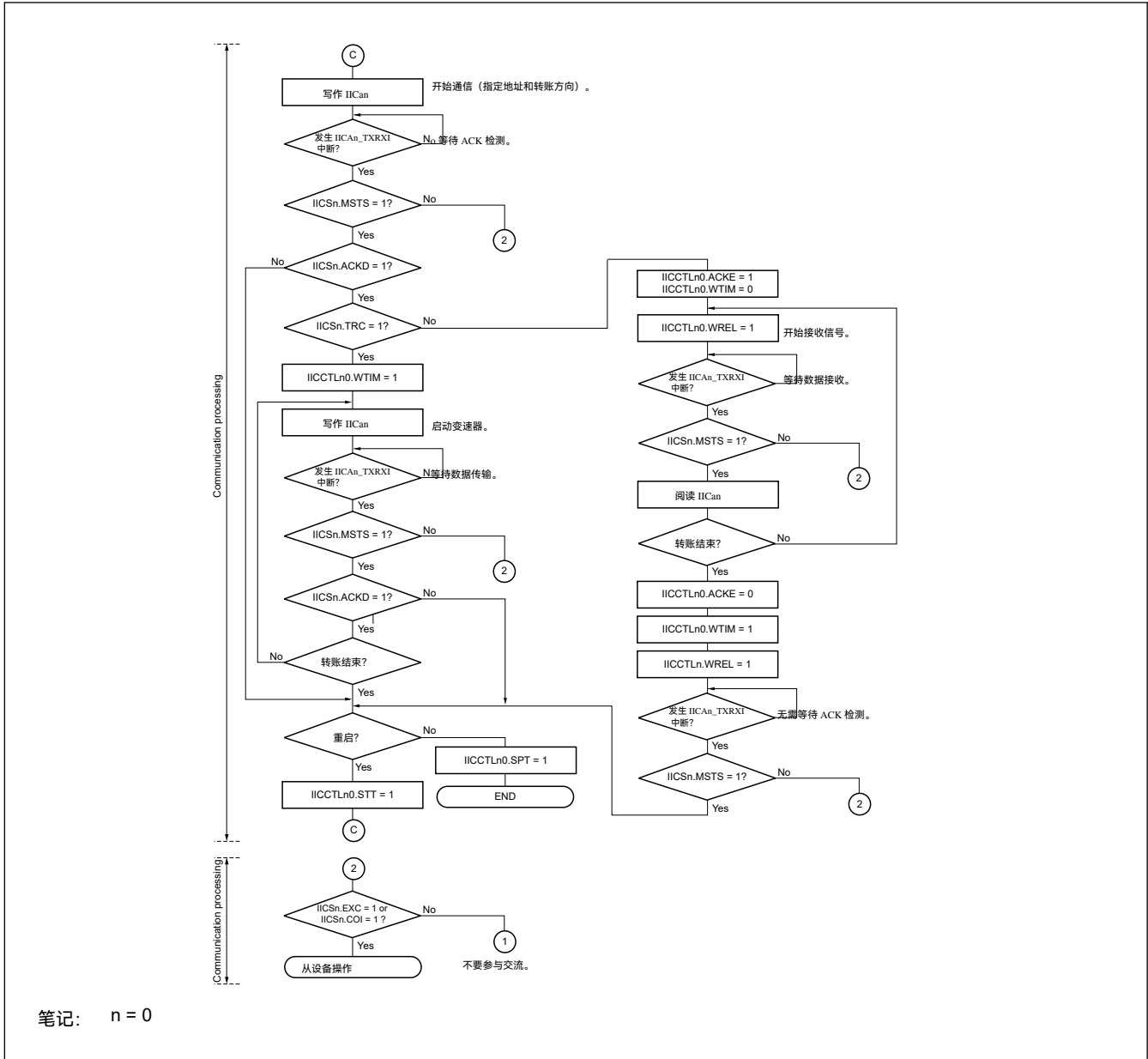


图 20.19 多主控系统中的主控设备操作 (3/3)

(3) 从设备操作

从设备操作的处理流程如下。

从设备的操作基本上是事件驱动的。因此，需要通过 IICA0_TXRXI 中断进行处理（例如，在通信过程中检测到停止条件等，这些处理必须显著改变设备的操作状态）。以下说明假设所有地址匹配功能已禁用，并且数据通信不支持扩展代码。此外，还假设 IICA0_TXRXI 中断处理仅执行状态转换处理，实际的数据通信由主处理程序执行。

图 20.20 显示了主处理器在从设备操作下的接口配置。

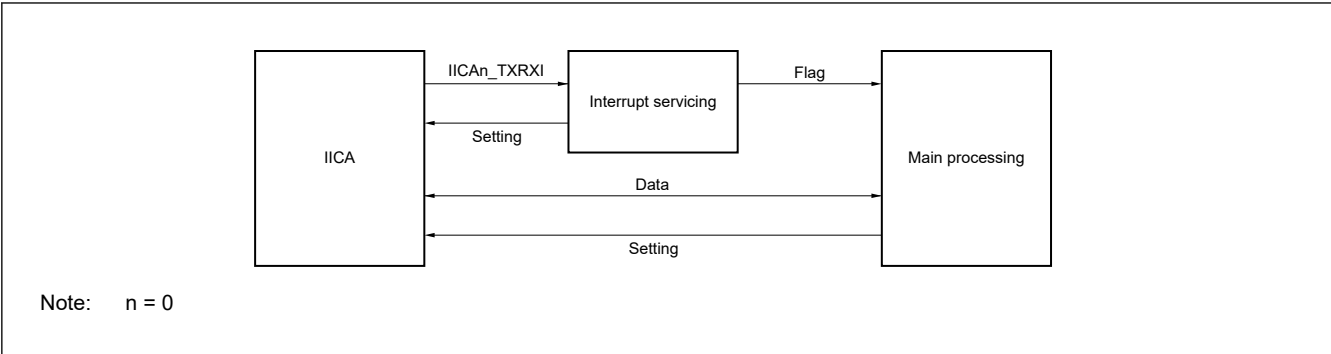


Figure 20.20 Interface configuration with the main processor in slave device operation

Therefore, data communication processing is performed by preparing the following three flags and passing them to the main processing instead of IICA0_TXRXI.

<1> Communication mode flag

This flag indicates the following two communication statuses.

- Clear mode: Status in which data communication is not performed
- Communication mode: Status in which data communication is performed (from valid address detection to stop condition detection, no detection of ACK from master device, address mismatch)

<2> Ready flag

This flag indicates that data communication is enabled. Its function is the same as the IICA0_TXRXI interrupt for ordinary data communication. This flag is set by interrupt servicing and cleared by the main processing. Clear this flag by interrupt servicing when communication is started. However, the ready flag is not set by interrupt servicing when the first data is transmitted. Therefore, the first data is transmitted without the flag being cleared (an address match is interpreted as a request for the next data).

<3> Communication direction flag

This flag indicates the direction of communication. Its value is the same as the IICS0.TRC bit.

The main processing of the slave device operation is explained next.

Start I²C bus interface (IICA) and wait until communication is enabled. When communication is enabled, execute communication by using the communication mode flag and ready flag (processing of the stop condition and start condition is performed by an interrupt. Here, check the status by using the flags).

The transmission operation is repeated until the master device no longer returns ACK. If ACK is not returned from the master device, communication is completed.

For reception, the necessary amount of data is received. When communication is completed, ACK is not returned as the next data. After that, the master device generates a stop condition or restart condition. Exit from the communication status occurs in this way.

Conform to the specifications of the product that is communicating, with respect to the transmission and reception formats.

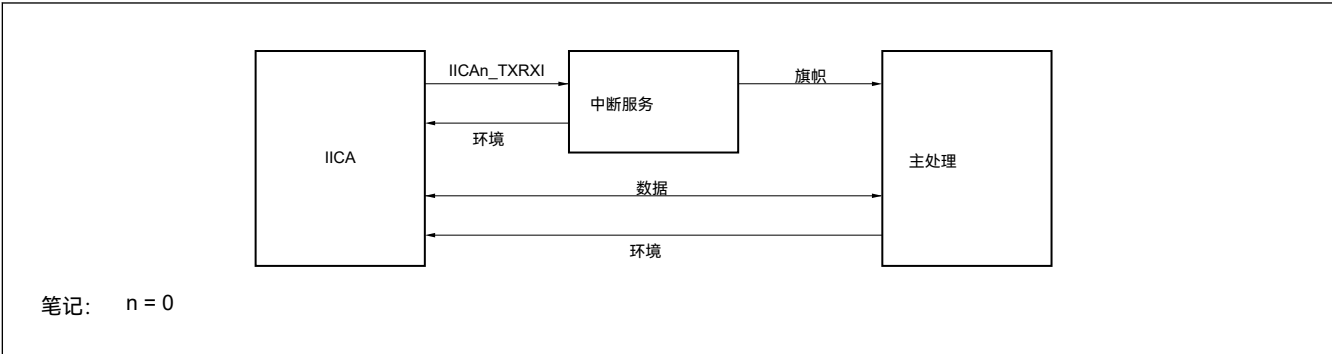


图 20.20 从设备运行状态下与主处理器的接口配置

因此，数据通信处理是通过准备以下三个标志并将它们传递给主处理而不是 IICA0_TXRXI 来进行的。

<1> 通信模式标志

该标志指示以下两种通信状态。

- 清除模式：不进行数据通信的状态
- 通信模式：数据通信执行的状态（从有效地址检测到停止条件检测、未检测到主设备ACK、地址不匹配）

<2>就绪标志

此标志位表示数据通信已启用。其功能与普通数据通信中的 IICA0_TXRXI 中断相同。此标志位由中断服务设置，并由主处理清除。通信开始时，中断服务会清除此标志位。但是，在发送第一个数据时，中断服务不会设置就绪标志位。因此，第一个数据会在标志位未被清除的情况下发送（地址匹配被解释为请求下一个数据）。

<3>通信方向标志

该标志位指示通信方向。其值与 IICS0.TRC 位相同。

接下来解释从设备操作的主要处理过程。

启动I²C总线接口 (IICA) 并等待通信启用。通信启用后，使用通信模式标志和就绪标志执行通信（停止条件和启动条件的处理由中断执行。此处，使用标志检查状态）。

传输操作会重复进行，直到主设备不再返回 ACK 为止。如果主设备没有返回 ACK，则通信完成。

接收端接收到所需数据量后，通信完成，不会返回 ACK 作为后续数据。之后，主设备会生成停止条件或重启条件，通信状态由此退出。

在传输和接收格式方面，应符合通信产品的规格。

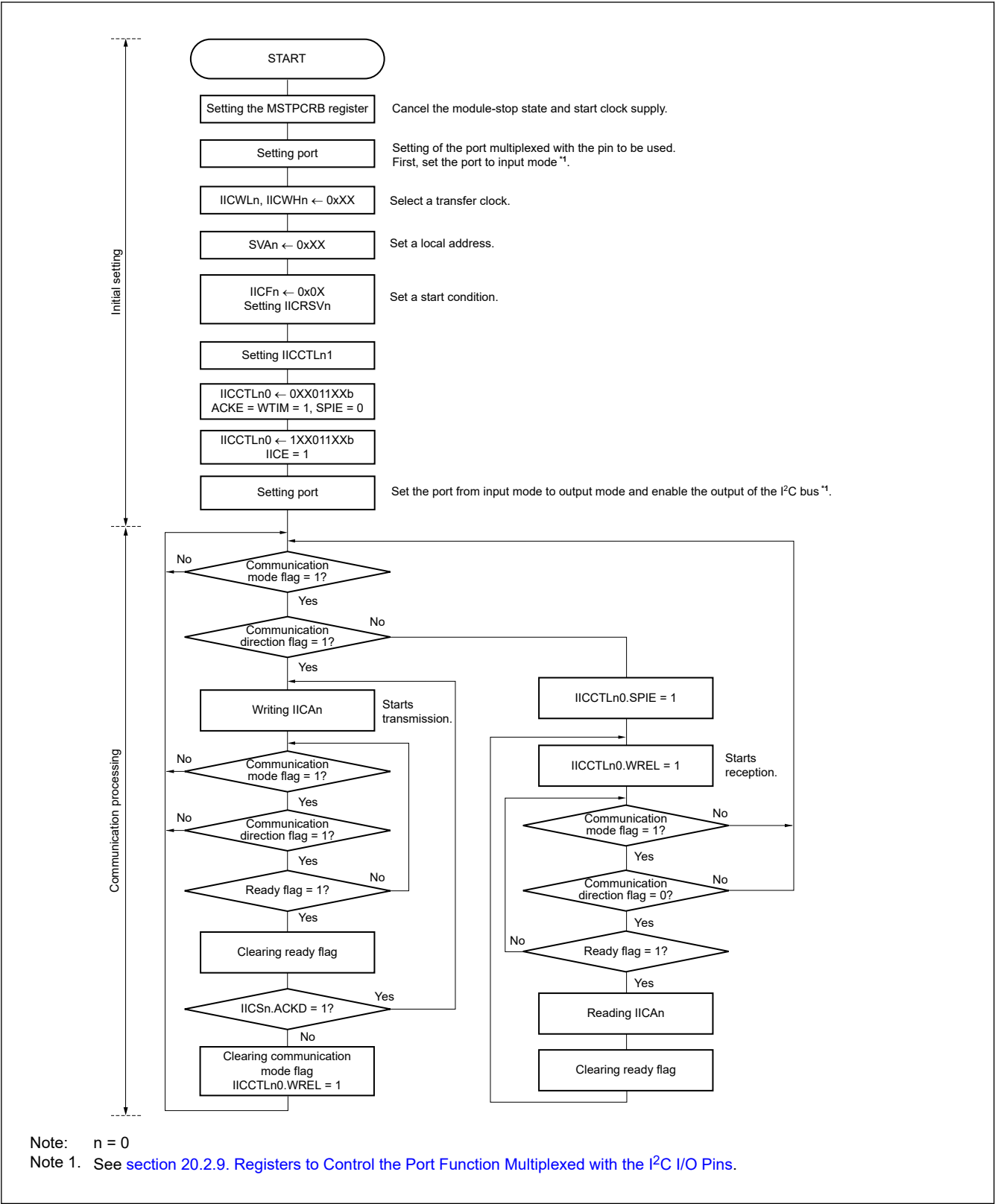


Figure 20.21 Slave device operation flowchart (1)

An example of the processing procedure of the slave device operation with the IICA0_TXRHI interrupt is explained below (processing is performed assuming that the all address match function is disabled and no extension code is used). The IICA0_TXRHI interrupt checks the status, and the following operations are performed.

- <1> Communication is stopped if the stop condition is issued.
- <2> If the start condition is issued, the address is checked and communication is completed if the address does not match.

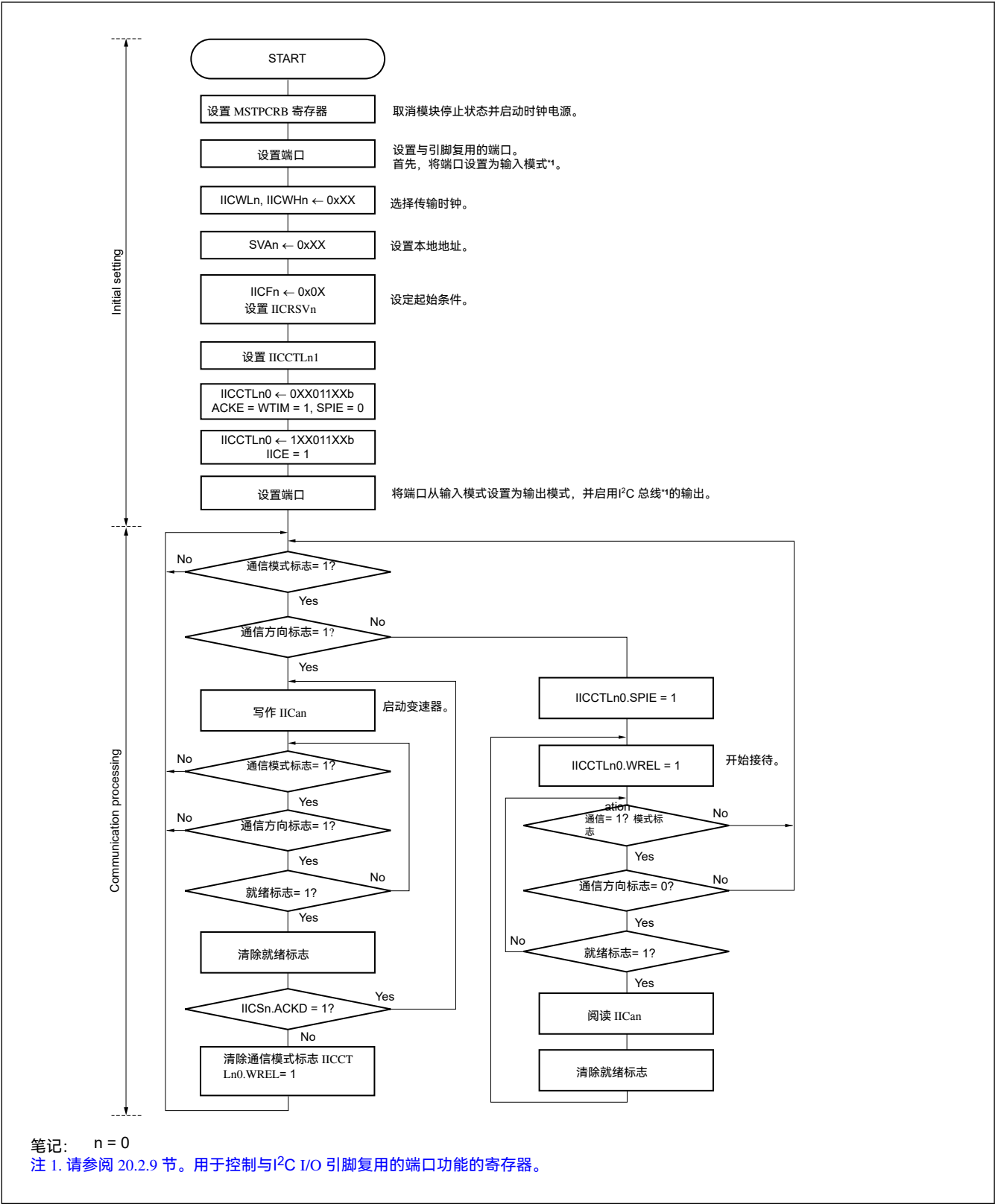


图 20.21 从设备操作流程 (1)

以下示例说明了使用 IICA0_TXRHI 中断处理从设备操作的过程 (假设所有地址匹配功能已禁用且未使用扩展代码)。IICA0_TXRHI 中断检查状态, 并执行以下操作。

- <1> 如果发出停止条件, 则通信停止。
- <2> 如果发出开始条件, 则检查地址, 如果地址不匹配, 则通信结束。

If the address matches, the communication mode is set, wait is canceled, and processing returns from the interrupt (the ready flag is cleared).

<3> For data transmit/receive, only the ready flag is set. Processing returns from the interrupt with the I²C bus remaining in the communication standby status.

Note: <1> to <3> above correspond to <1> to <3> in Figure 20.22.

Figure 20.22 shows the interrupt flowchart for slave device operation.

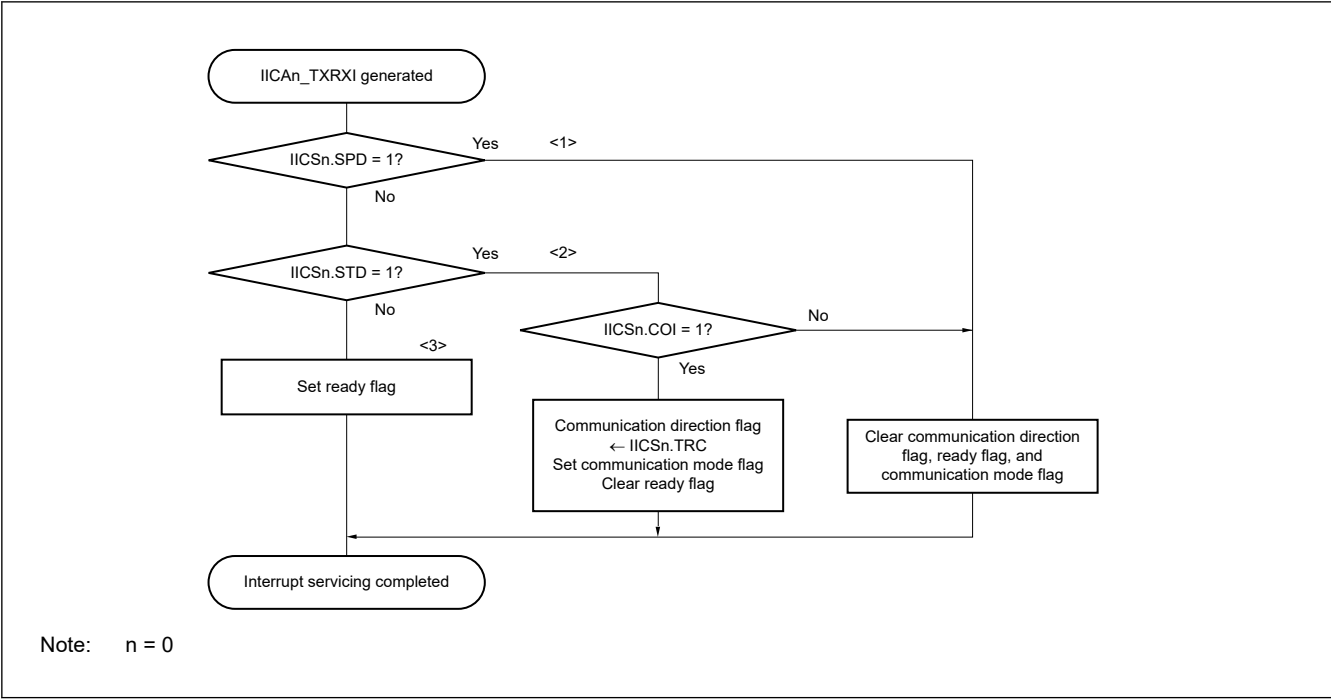


Figure 20.22 Slave device operation flowchart (2)

20.3.19 Timing of I²C Interrupt Request Signal (IICA0_TXRXI) Occurrence

The timing of transmitting or receiving data and generation of interrupt request signal IICA0_TXRXI, and the value of the IICA status register 0 (IICS0) when the IICA0_TXRXI signal is generated are shown in Figure 20.23 to Figure 20.62.

Note: ST: Start condition
AD6 to AD0: Address bits
R/W#: Transfer direction specification bit
ACK: Acknowledge bit
D7 to D0: Data bits
SP: Stop condition

如果地址匹配，则设置通信模式，取消等待，处理从中断返回（清除就绪标志）。

<3>数据传输/接收时，仅设置就绪标志。中断处理返回后，I²C总线仍处于通信待机状态。

注意：上面的<1>到<3>对应于图 20.22 中的<1>到<3>。

图 20.22 显示了从设备操作的中断流程图。

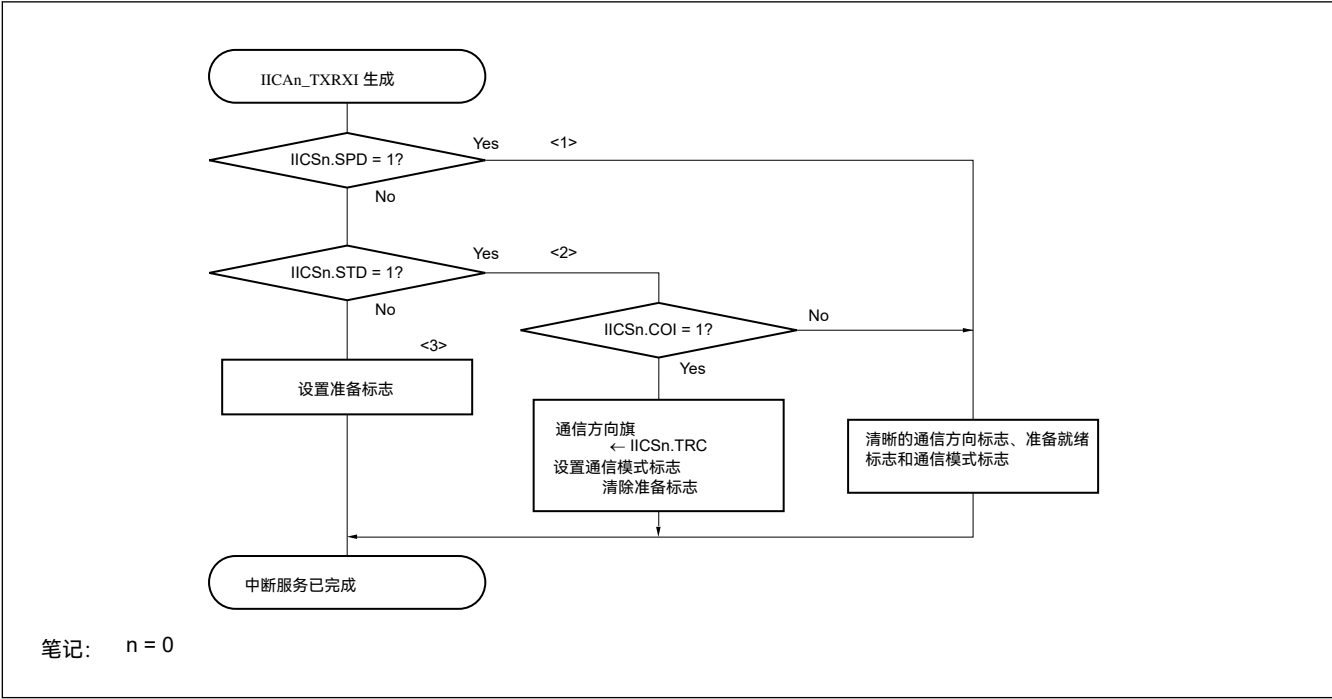


图 20.22 从设备操作流程图 (2)

20.3.19 I²C 中断请求信号 (IICA0_TXRXI) 发生的时间

图 20.23 至图 20.62 显示了发送或接收数据和生成中断请求信号 IICA0_TXRXI 的时序，以及生成 IICA0_TXRXI 信号时 IICA 状态寄存器 0 (IICS0) 的值。

注: ST: 起始条件
AD6 至 AD0: 地址位
R/W#: 传输方向指定位
ACK: 确认位
D7 至 D0: 数据位
SP: 停止条件

(1) Master device operation

(a) Start → Address → Data → Data → Stop (reception/transmission)

1. When IICCTL00.WTIM = 0

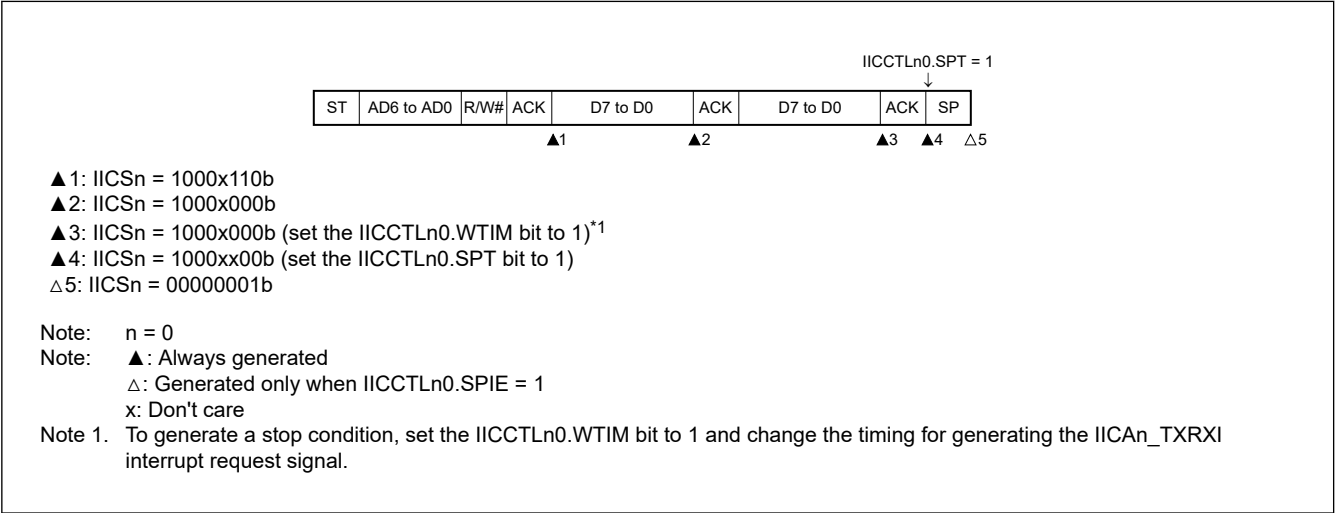


Figure 20.23 Master device operation reception/transmission (IICCTL00.WTIM = 0)

2. When IICCTL00.WTIM = 1

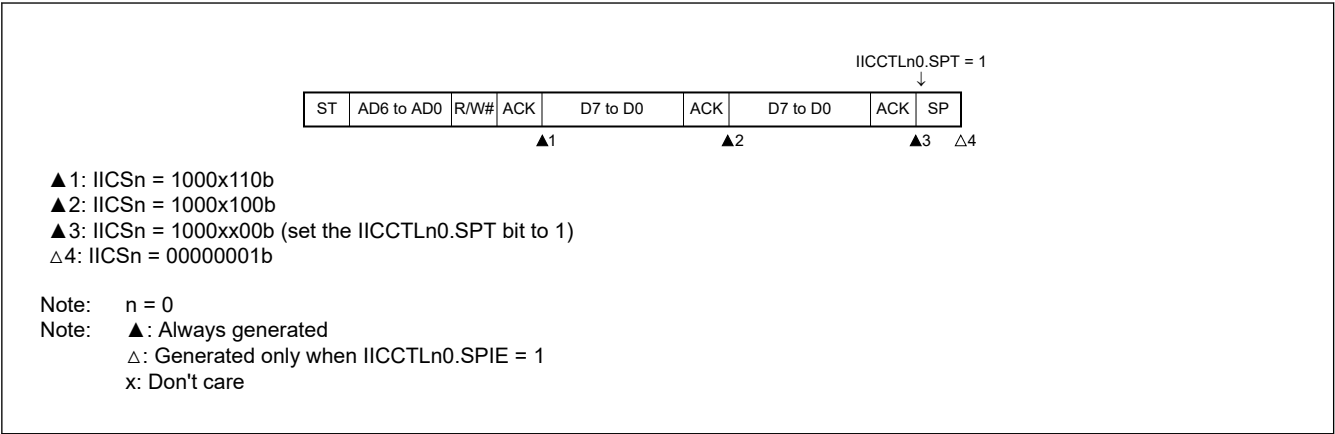


Figure 20.24 Master device operation reception/transmission (IICCTL00.WTIM = 1)

(1) 主设备操作

(a) 开始→地址→数据→数据→停止（接收/发送）

1. 当 IICCTL00.WTIM= 0

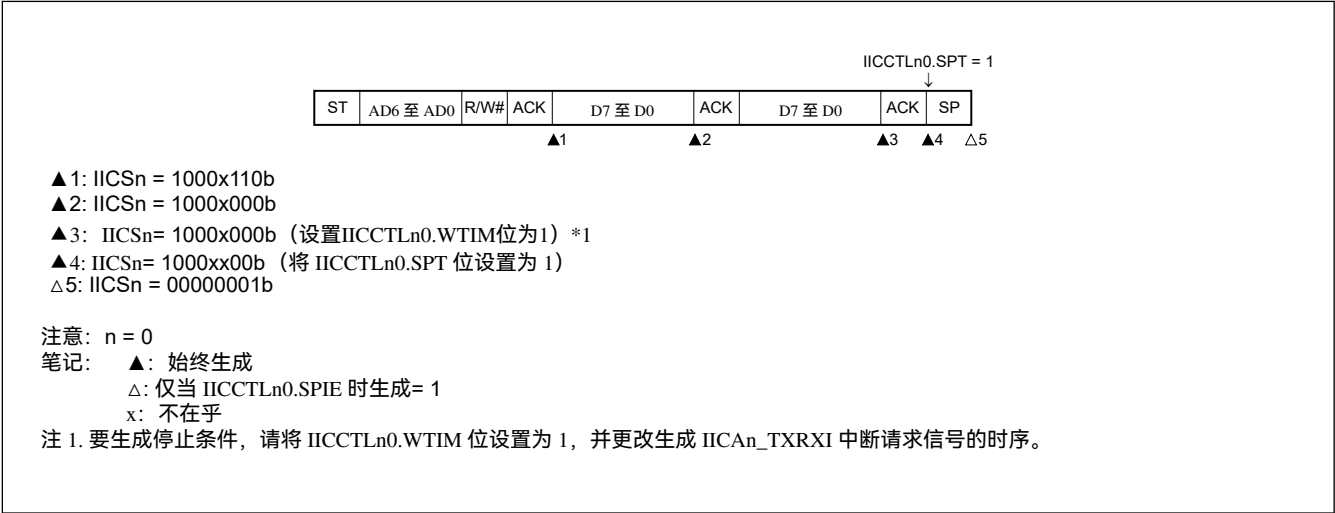


图 20.23 主设备操作接收/发送 (IICCTL00.WTIM= 0)

2. When IICCTL00.WTIM = 1

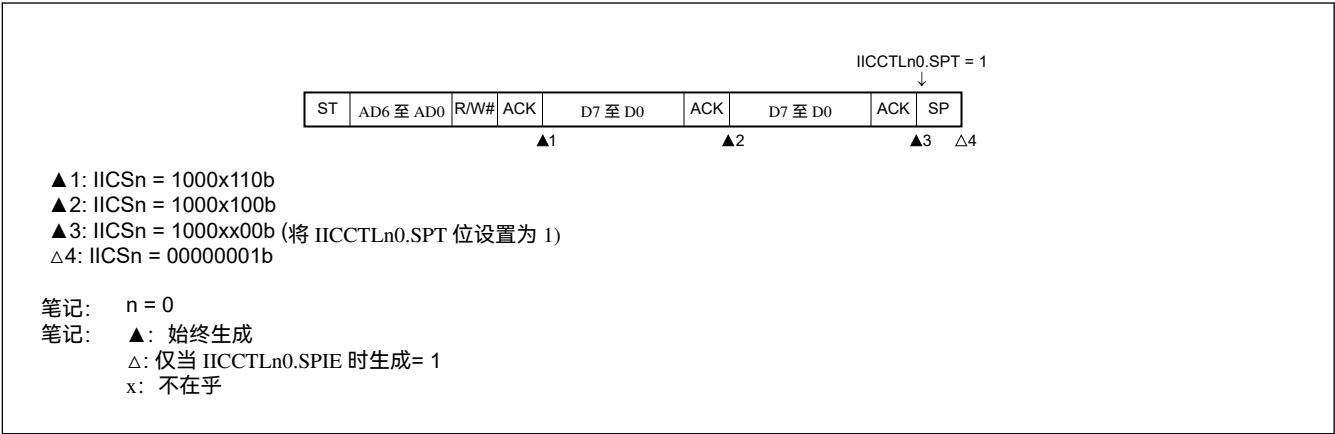


图 20.24 主设备操作接收/发送 (IICCTL00.WTIM= 1)

(b) Start → Address → Data → Start → Address → Data → Stop (restart)

1. When IICCTL00.WTIM = 0

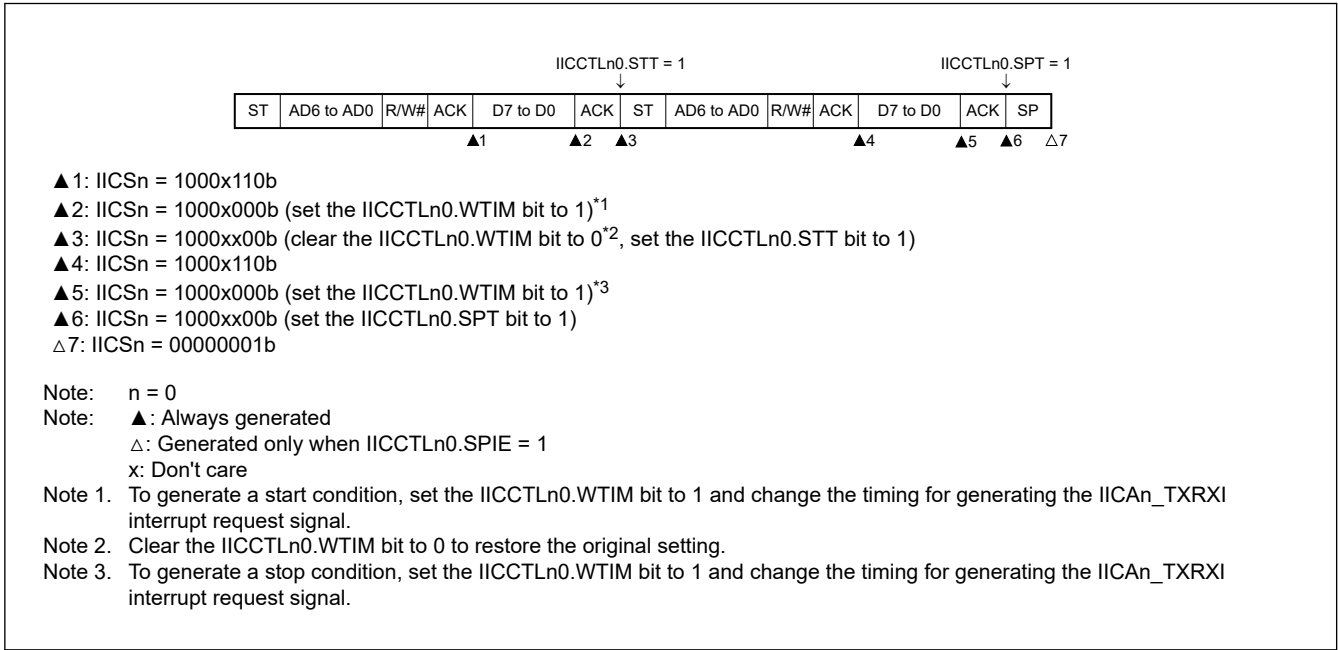


Figure 20.25 Master device operation restart (IICCTL00.WTIM = 0)

2. When IICCTL00.WTIM = 1

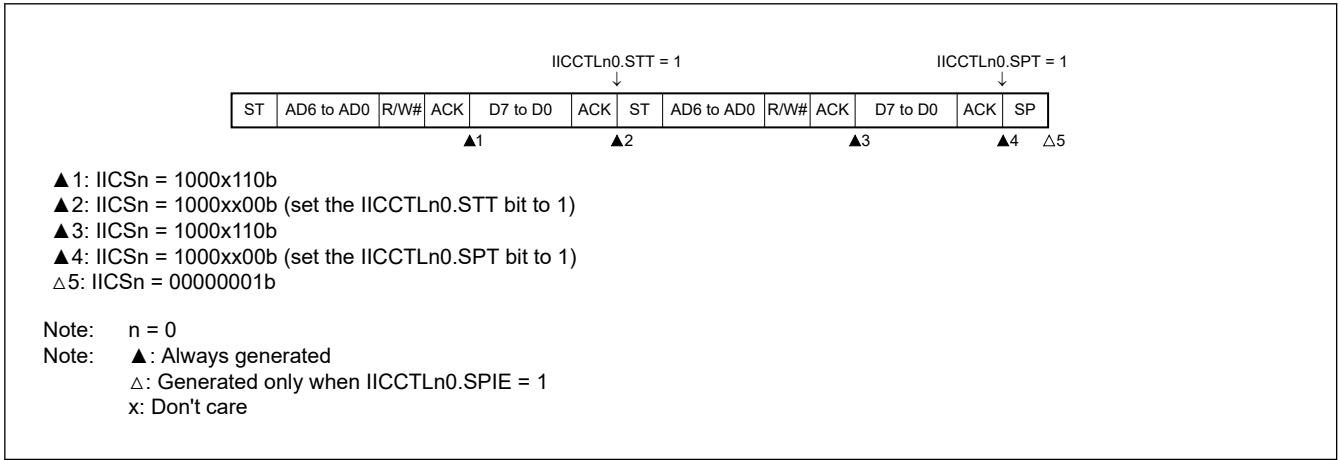


Figure 20.26 Master device operation restart (IICCTL00.WTIM = 1)

(b) 开始→地址→Data→Start→地址→数据→停止(重启)

1. When IICCTL00.WTIM = 0

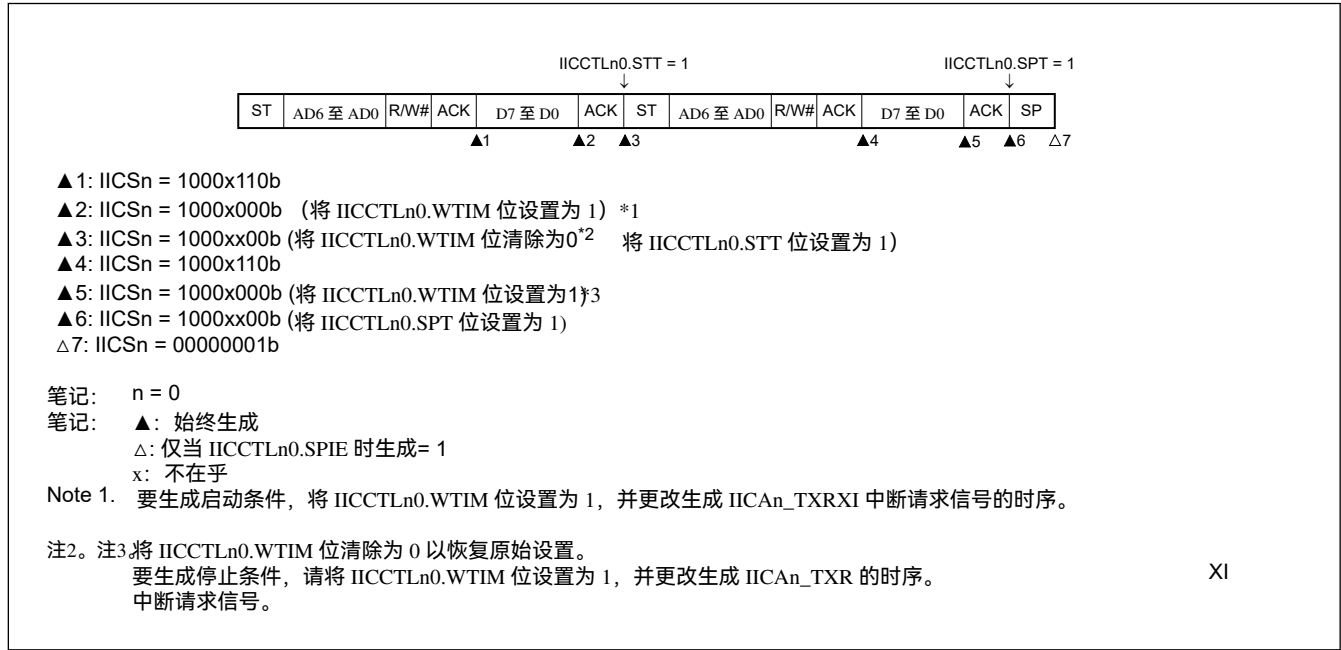


图 20.25 主设备操作重启 (IICCTL00.WTIM= 0)

2. 当 IICCTL00.WTIM = 1

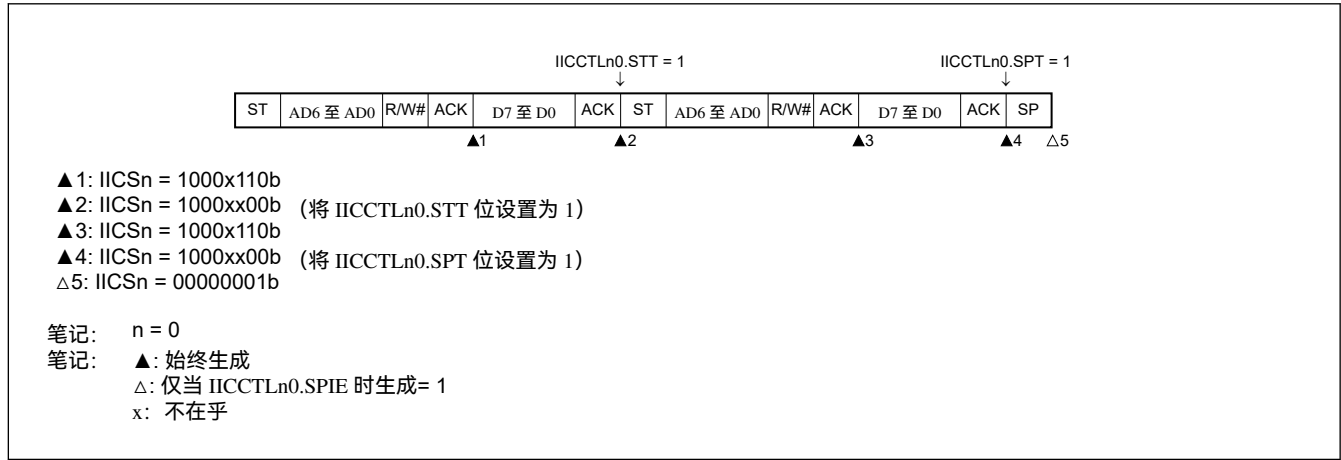


图 20.26 主设备操作重启 (IICCTL00.WTIM= 1)

(c) Start → Code → Data → Data → Stop (extension code transmission)

1. When IICCTL00.WTIM = 0

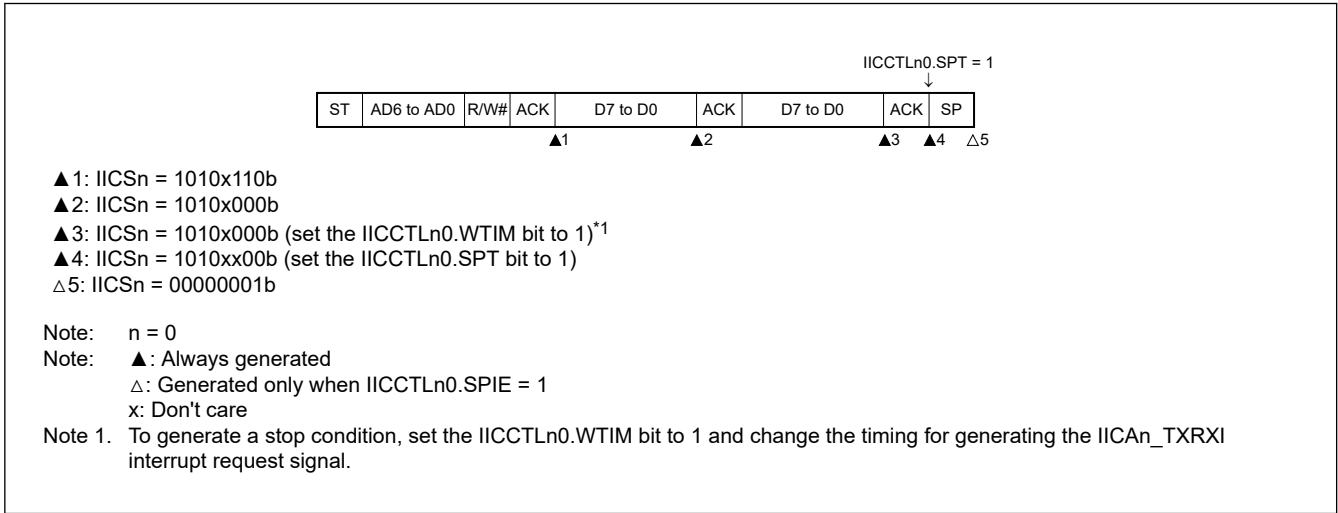


Figure 20.27 Master device operation extension code transmission (IICCTL00.WTIM = 0)

2. When IICCTL00.WTIM = 1

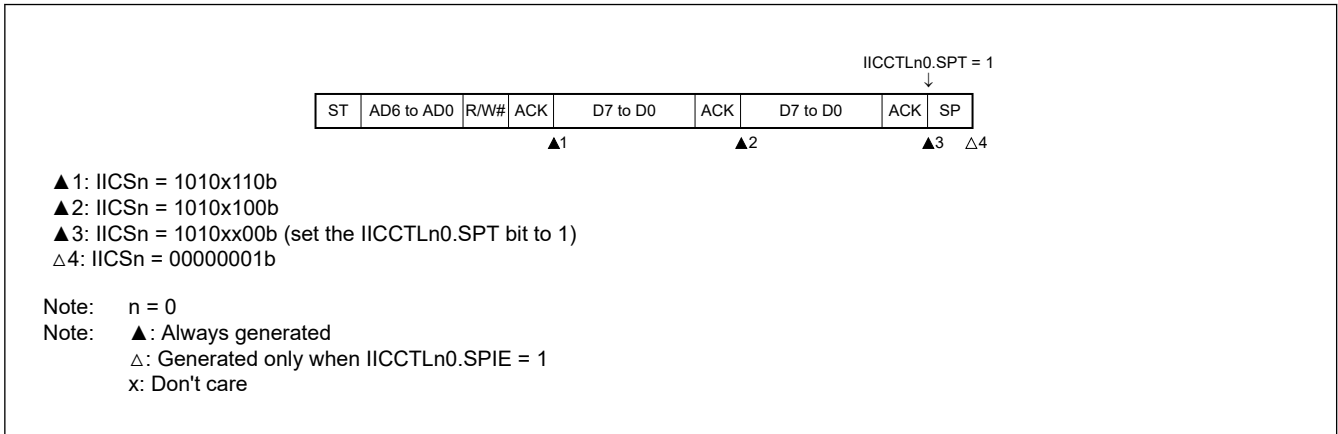


Figure 20.28 Master device operation extension code transmission (IICCTL00.WTIM = 1)

(2) Slave device operation (slave address data reception)

(a) Start → Address → Data → Data → Stop

1. When IICCTL00.WTIM = 0

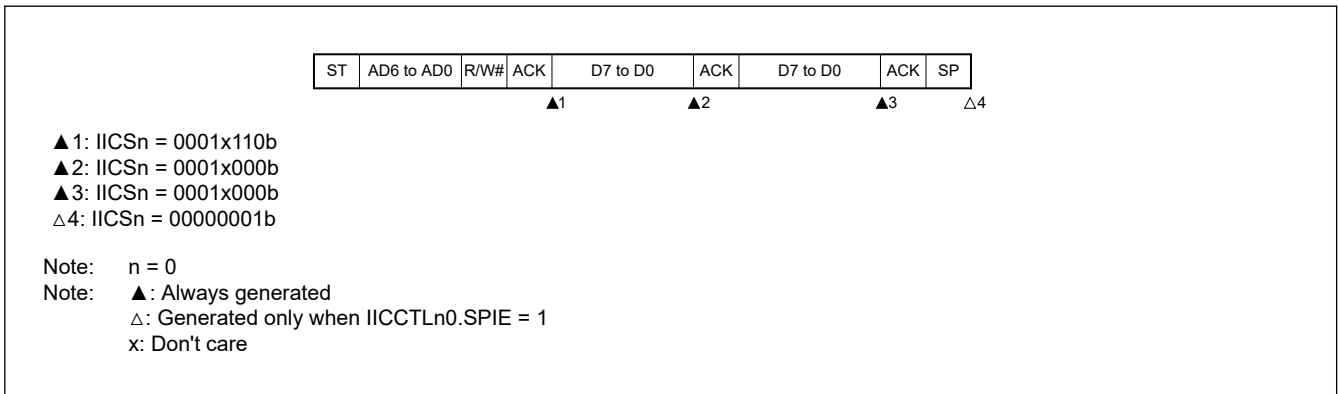


Figure 20.29 Slave device operation slave address data reception (IICCTL00.WTIM = 0)

(c) 开始→代码e → 数据→数据→停止 (扩展码传输)

1. 当 IICCTL00.WTIM= 0

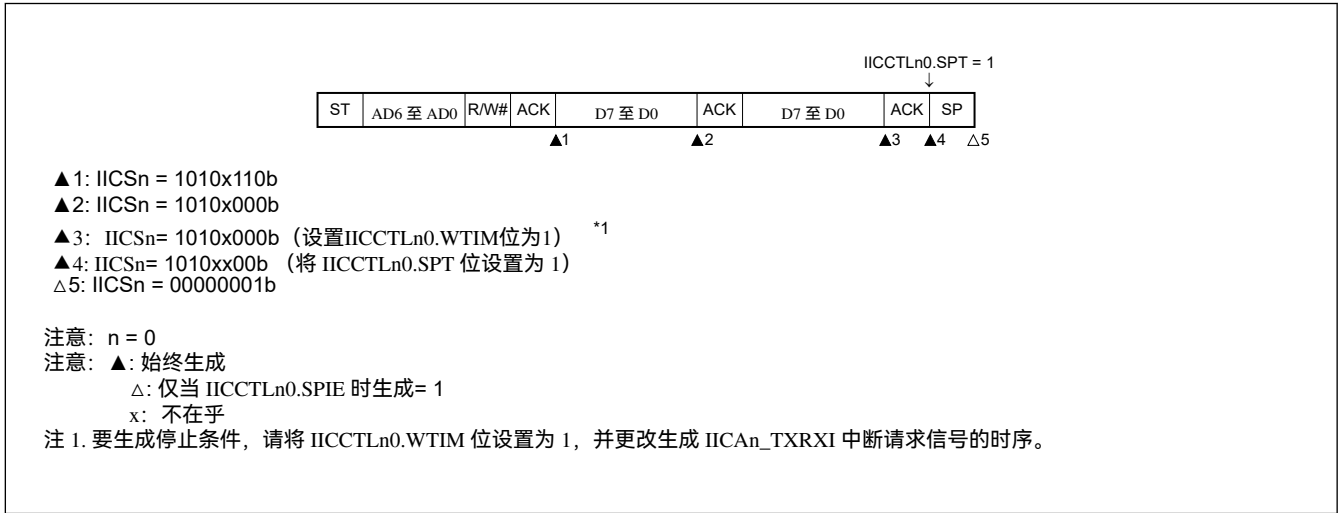


图 20.27 主设备操作扩展码传输 (IICCTL00.WTIM= 0)

2. 当 IICCTL00.WTIM= 1

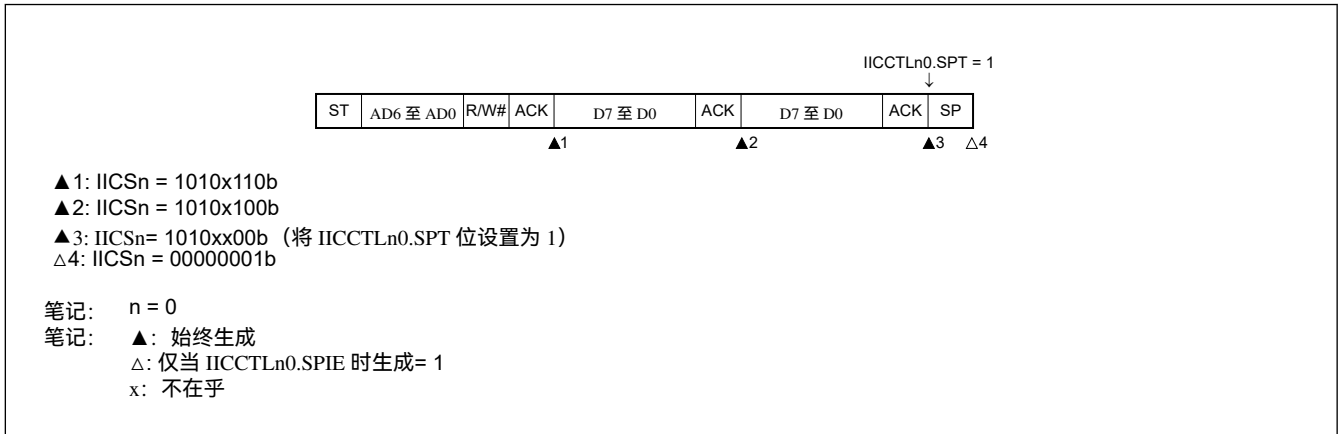


图 20.28 主设备操作扩展代码传输 (IICCTL00.WTIM= 1)

(2) 从设备操作 (从设备地址数据接收)

(a) 开始→地址→是ta → Data → 停止

1. 当 IICCTL00.WTIM= 0

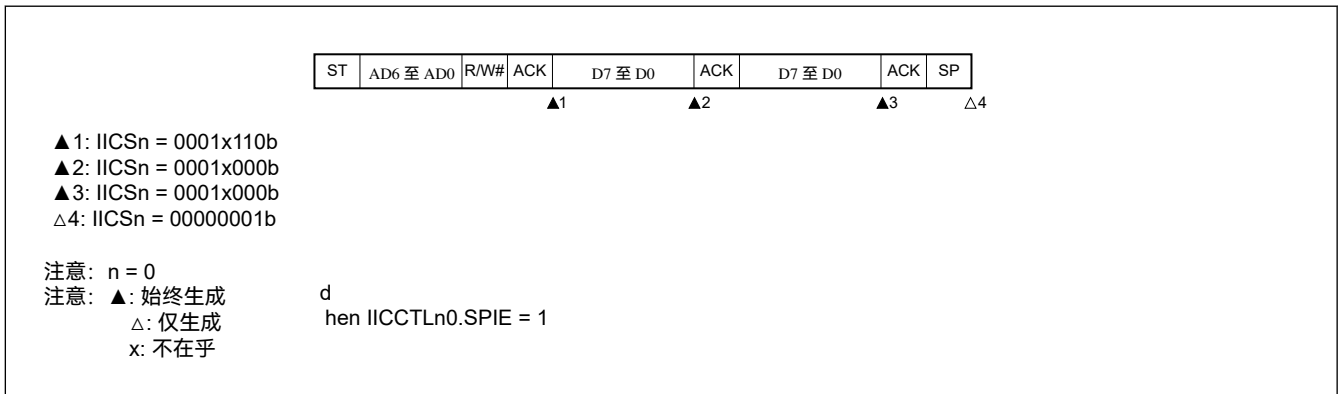


图 20.29 从设备操作从地址数据接收 (IICCTL00.WTIM= 0)

2. When IICCTL00.WTIM = 1

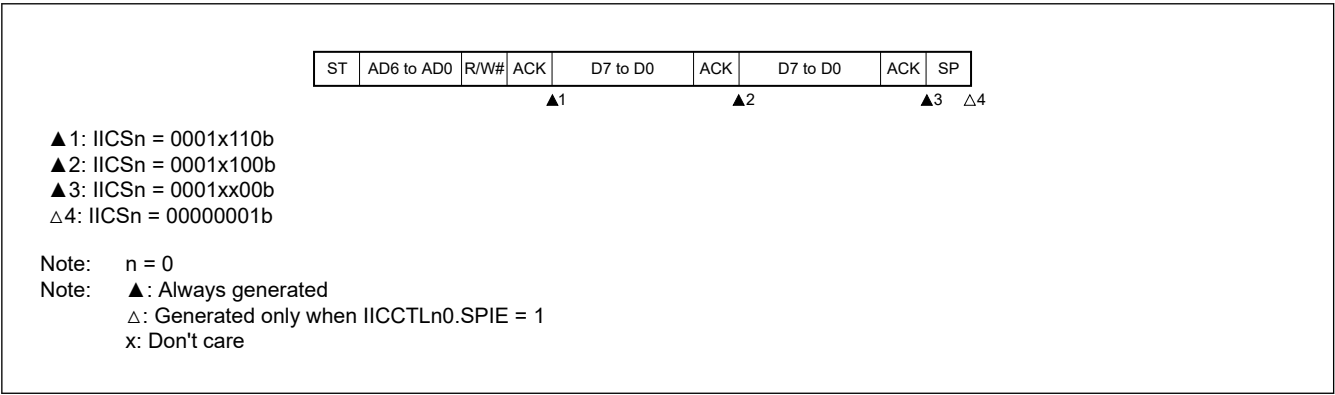


Figure 20.30 Slave device operation slave address data reception (IICCTL00.WTIM = 1)

(b) Start → Address → Data → Start → Address → Data → Stop

1. When IICCTL00.WTIM = 0 (after restart, matches with SVA0, the all address match function is disabled)

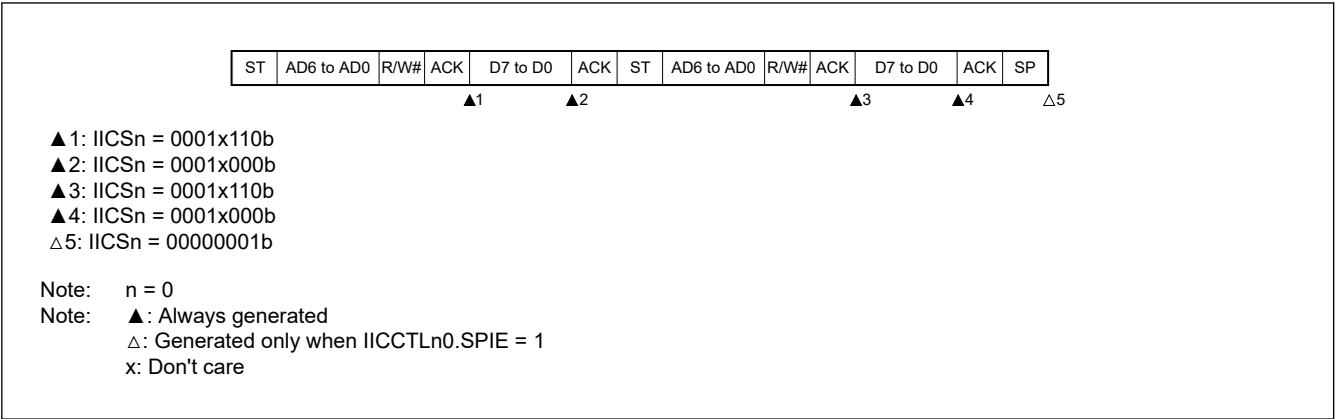


Figure 20.31 Slave device operation after normal access, matches with SVA0 (IICCTL00.WTIM = 0)

2. When IICCTL00.WTIM = 1 (after restart, matches with SVA0, the all address match function is disabled)

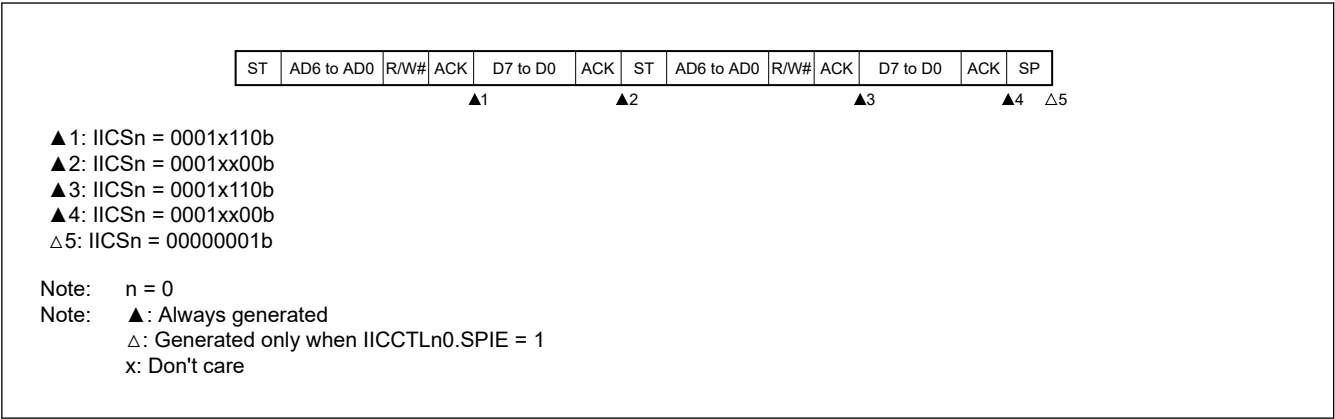


Figure 20.32 Slave device operation after normal access, matches with SVA0 (IICCTL00.WTIM = 1)

(c) Start → Address → Data → Start → Code → Data → Stop

1. When IICCTL00.WTIM = 0

(after restart, does not match address (extension code, the all address match function is disabled))

2. 当 IICCTL00.WTIM= 1

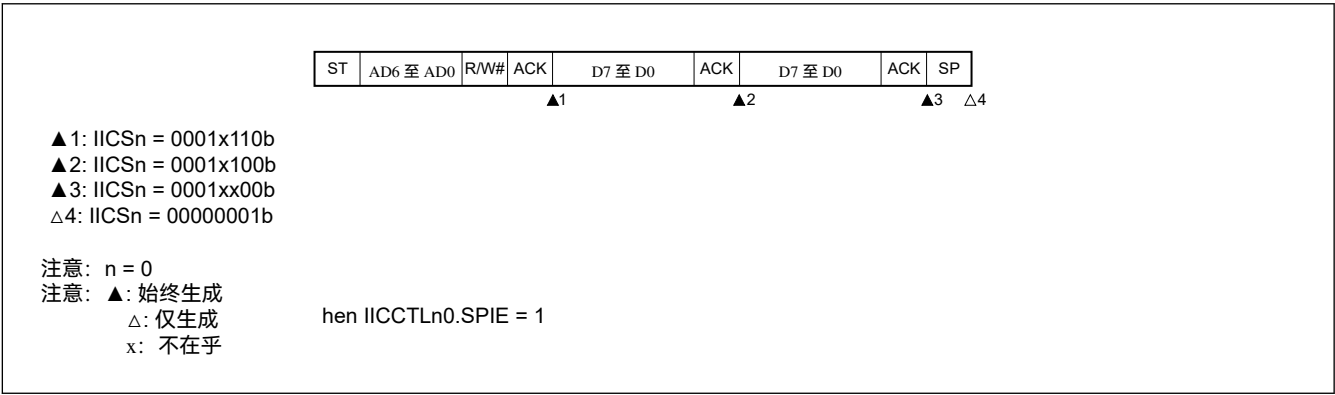


图 20.30 从设备操作从地址数据接收 (IICCTL00.WTIM= 1)

(b) 开始→地址→Data→Start→地址→数据→停止

1. 当 IICCTL00.WTIM= 0 （重启后，与 SVA0 匹配，所有地址匹配功能被禁用）

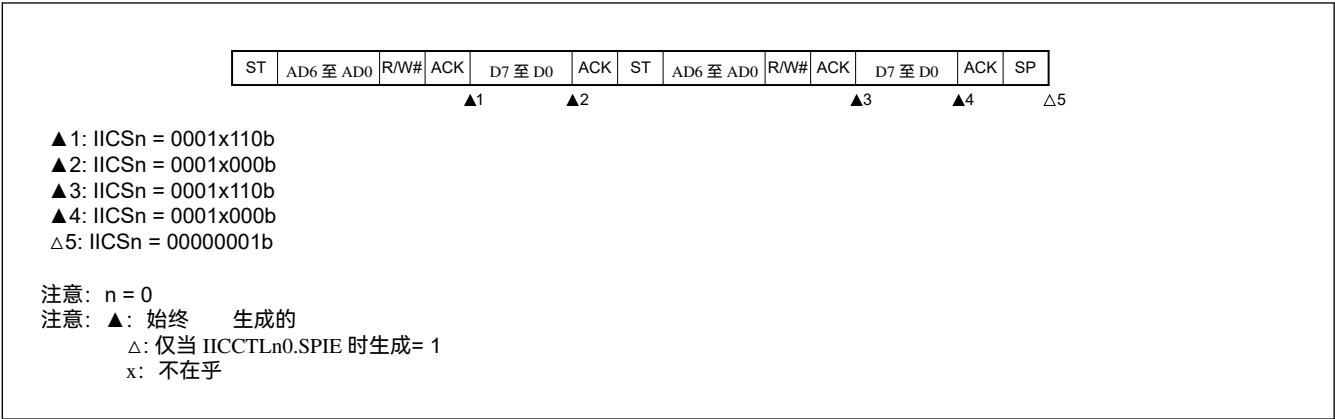


图 20.31 从设备在正常访问后的操作，与 SVA0 (IICCTL00.WTIM= 0) 匹配)

2. 当 IICCTL00.WTIM= 1 （重启后，与 SVA0 匹配，所有地址匹配功能被禁用）

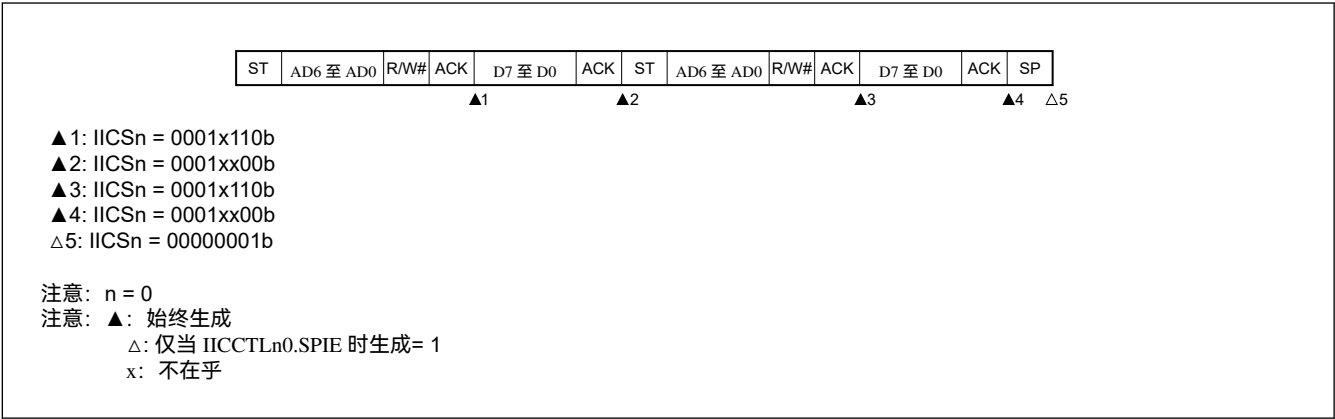


图 20.32 正常访问后从设备操作，与 SVA0 (IICCTL00.WTIM= 1) 匹配

(c) 起始→地址→数据→起始→代码→Data→停止

1. 当 IICCTL00.WTIM= 0

（重启后，地址不匹配（分机号，所有地址匹配功能已禁用））

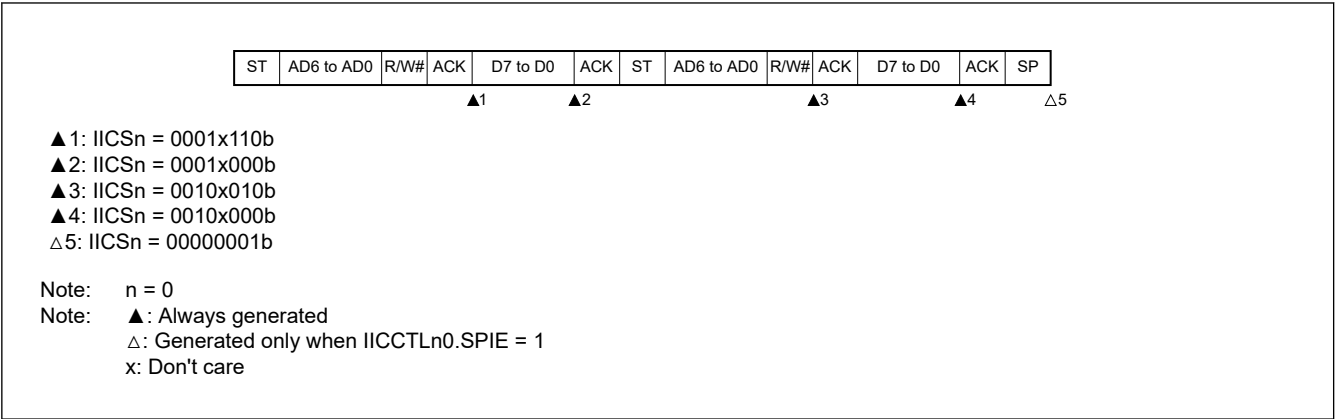


Figure 20.33 Slave device operation after normal access, matches the extension code (IICCTL00.WTIM = 0)

2. When IICCTL00.WTIM = 1

(after restart, does not match address (extension code, the all address match function is disabled))

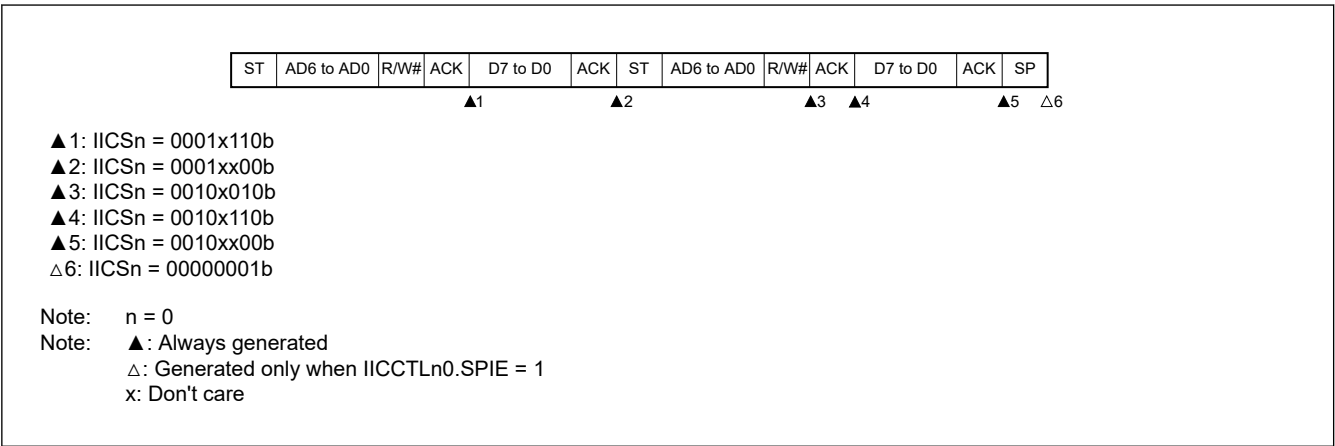


Figure 20.34 Slave device operation after normal access, matches the extension code (IICCTL00.WTIM = 1)

(d) Start → Address → Data → Start → Address → Data → Stop

1. When IICCTL00.WTIM = 0

(after restart, does not match address (not extension code, the all address match function is disabled))

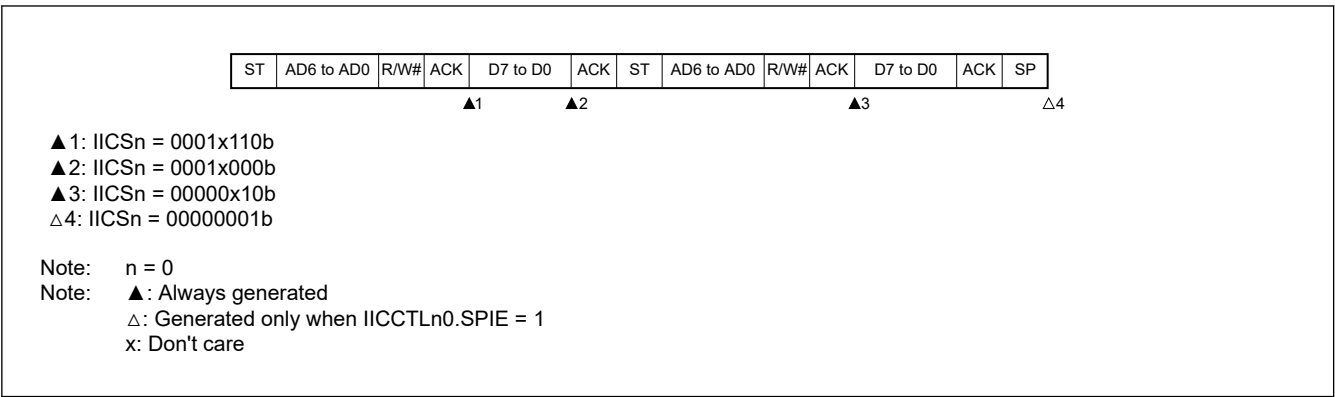


Figure 20.35 Slave device operation after normal access, does not match (IICCTL00.WTIM = 0)

2. When IICCTL00.WTIM = 1

(after restart, does not match address (not extension code, the all address match function is disabled))

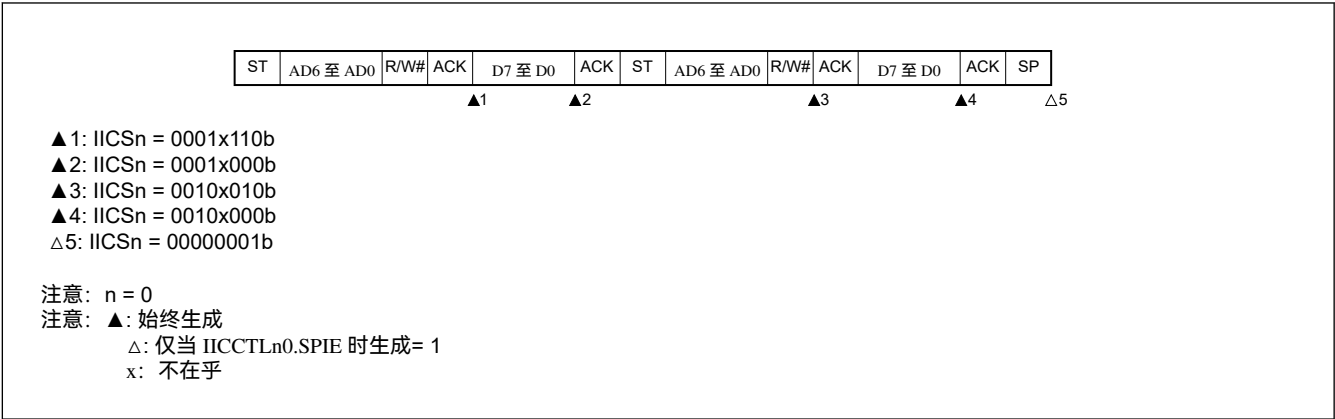


图 20.33 正常访问后从设备操作，与扩展代码 (IICCTL00.WTIM= 0) 匹配

2. 当 IICCTL00.WTIM= 1

(重启后，地址不匹配（分机号，所有地址匹配功能已禁用）)

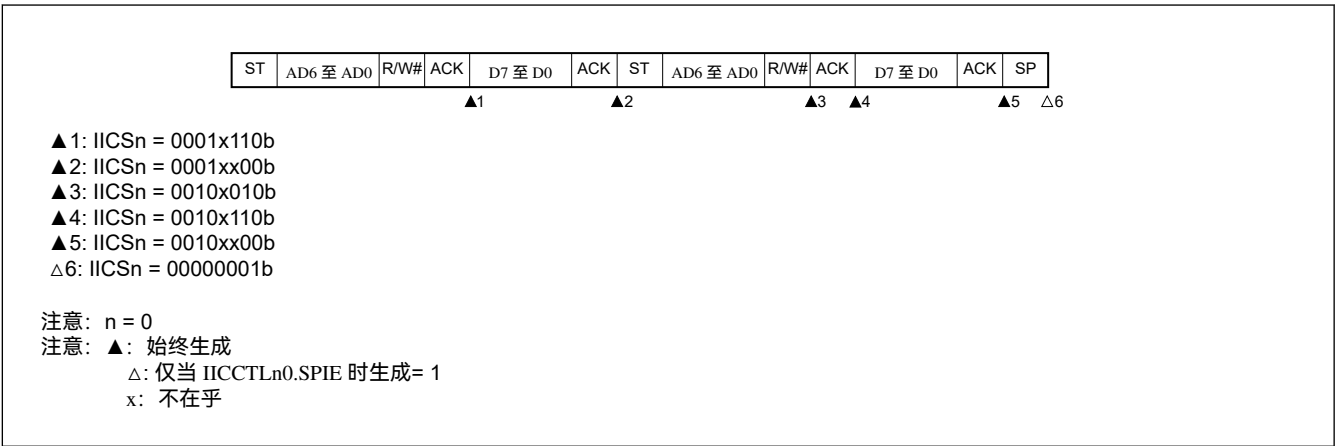


图 20.34 正常访问后从设备操作，与扩展代码 (IICCTL00.WTIM= 1) 匹配

(d) 开始 → 地址ss → Data → Start → A数据地址 → 数据 → 停止

1. 当 IICCTL00.WTIM= 0

(重启后，地址不匹配（非分机号，所有地址匹配功能已禁用）)

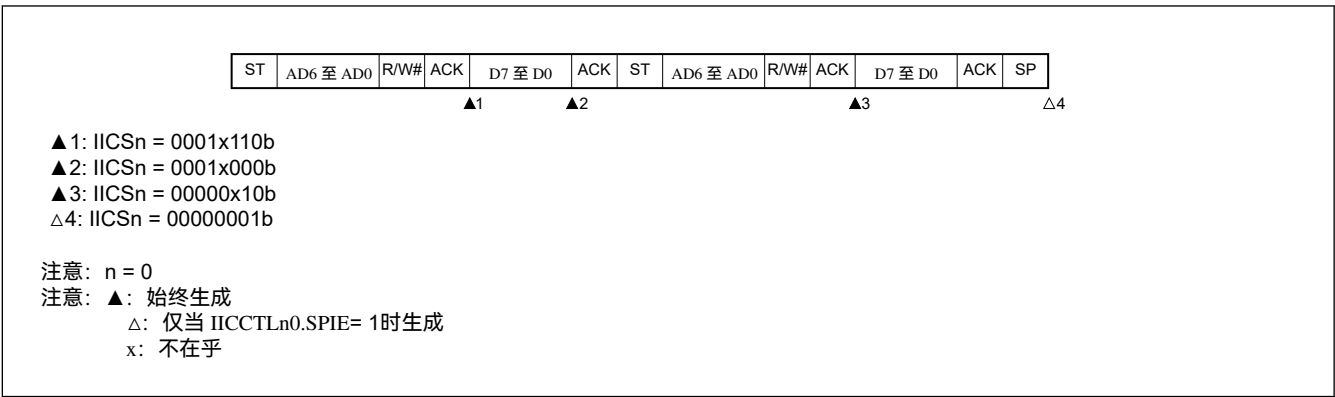


图 20.35 正常访问后从设备操作，与 (IICCTL00.WTIM= 0) 不匹配

2. 当 IICCTL00.WTIM= 1

(重启后，地址不匹配（非分机号，所有地址匹配功能已禁用）)

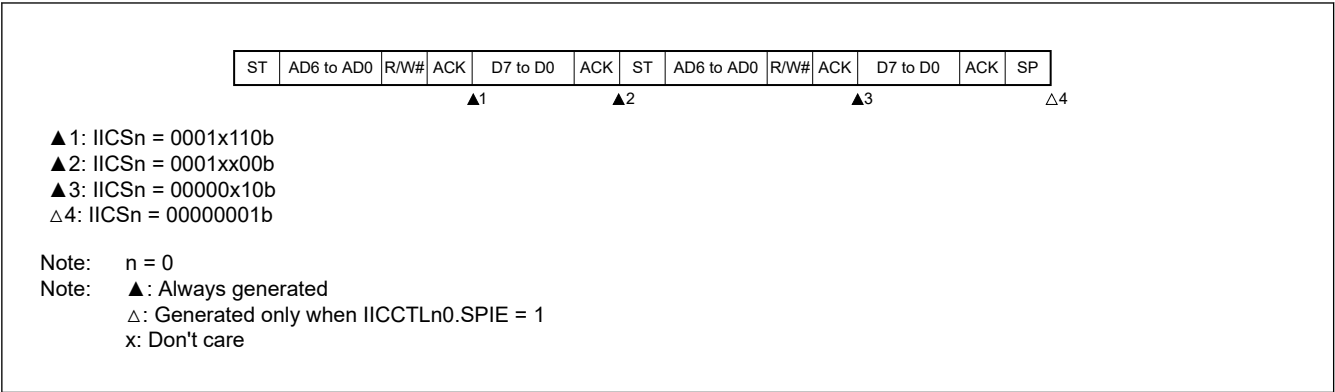


Figure 20.36 Slave device operation after normal access, does not match (IICCTL00.WTIM = 1)

(3) Slave device operation (when receiving extension code and the all address match function is disabled)

The device is always participating in communication when it receives an extension code.

(a) Start → Code → Data → Data → Stop

1. When IICCTL00.WTIM = 0

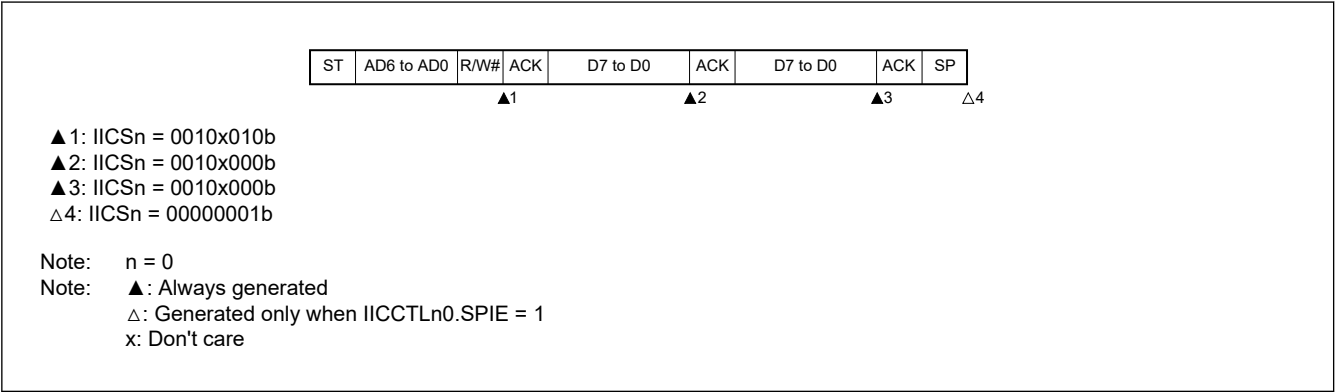


Figure 20.37 Slave device operation receiving extension code (IICCTL00.WTIM = 0)

2. When IICCTL00.WTIM = 1

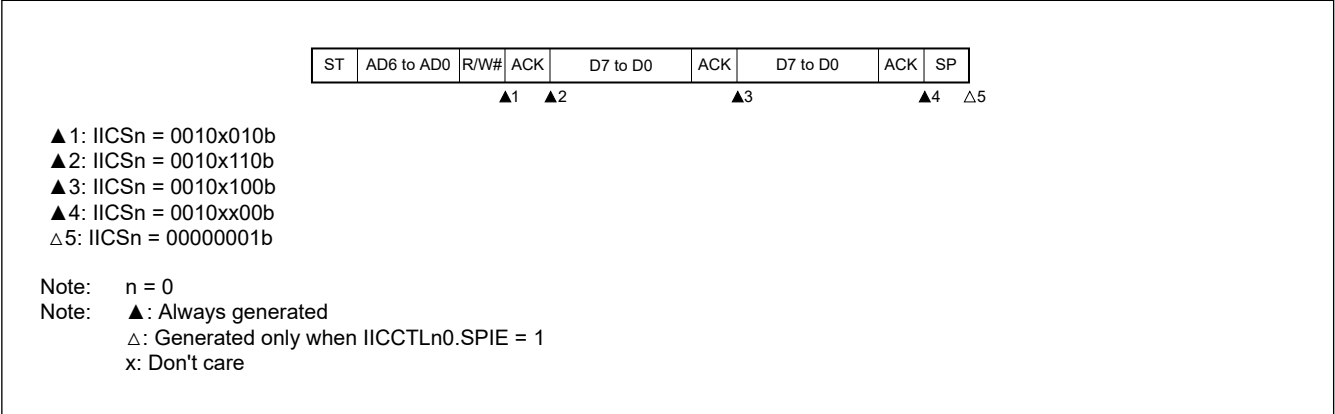


Figure 20.38 Slave device operation receiving extension code (IICCTL00.WTIM = 1)

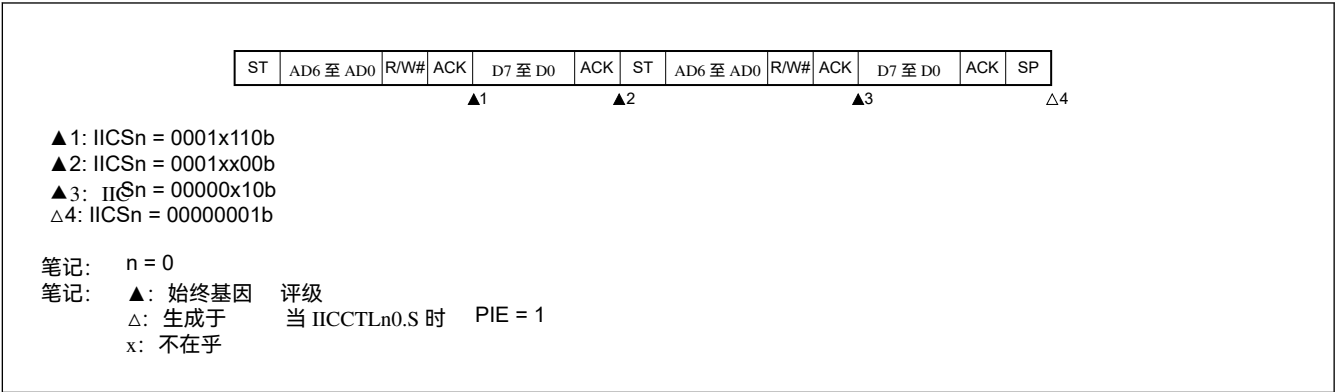


图 20.36 正常访问后从设备操作与 (IICCTL00.WTIM= 1) 不匹配

(3) 从设备操作（接收分机号且全地址匹配功能禁用时）

当设备收到分机号时，它始终处于通信状态。

(a) 开始→Code → Data → Data→停止

1. 当 IICCTL00.WTIM= 0

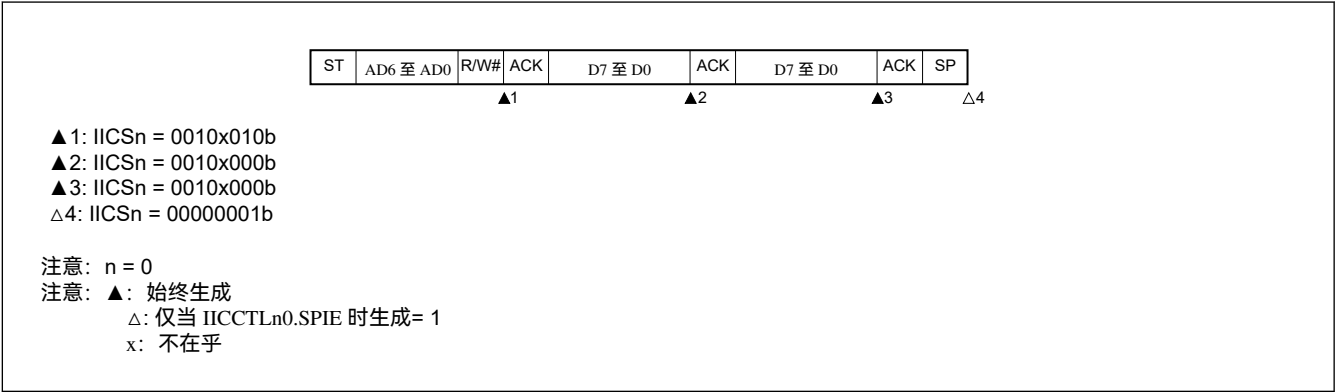


图 20.37 从设备操作接收扩展代码 (IICCTL00.WTIM= 0)

2. 当 IICCTL00.WTIM= 1

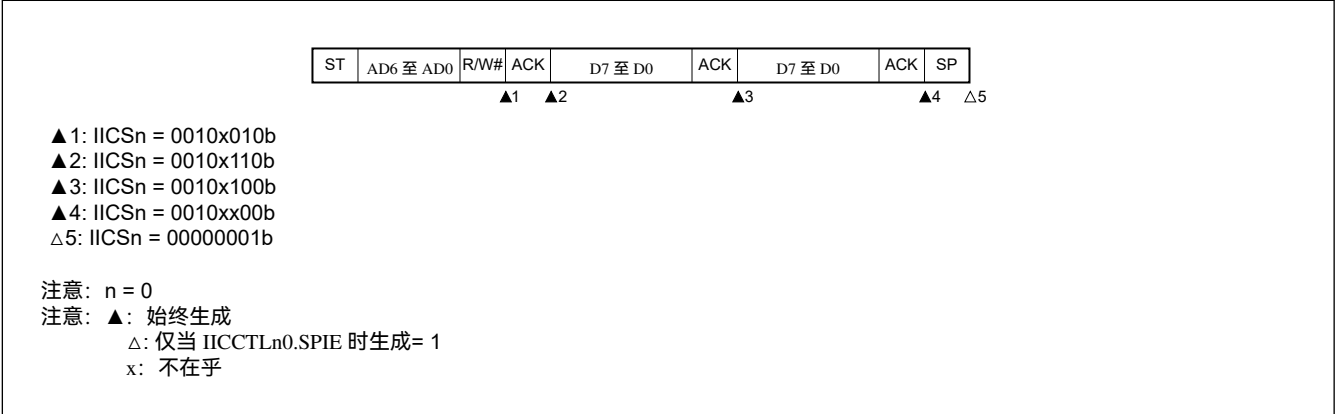


图 20.38 从设备操作接收扩展代码 (IICCTL00.WTIM= 1)

(b) Start → Code → Data → Start → Address → Data → Stop

1. When IICCTL00.WTIM = 0 (after restart, matches with SVA0, the all address match function is disabled)

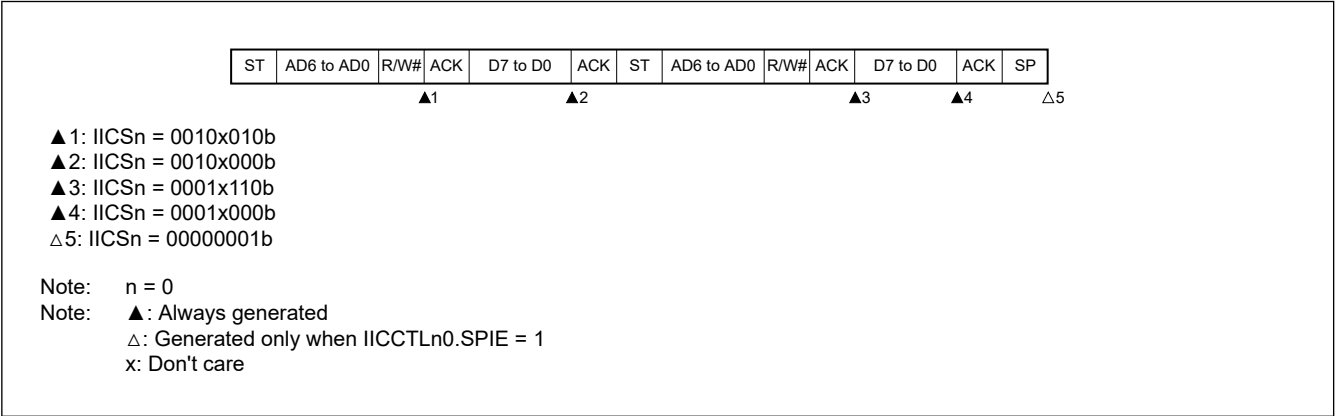


Figure 20.39 Slave device operation after code access, matches with SVA0 (IICCTL00.WTIM = 0)

2. When IICCTL00.WTIM = 1 (after restart, matches with SVA0, the all address match function is disabled)

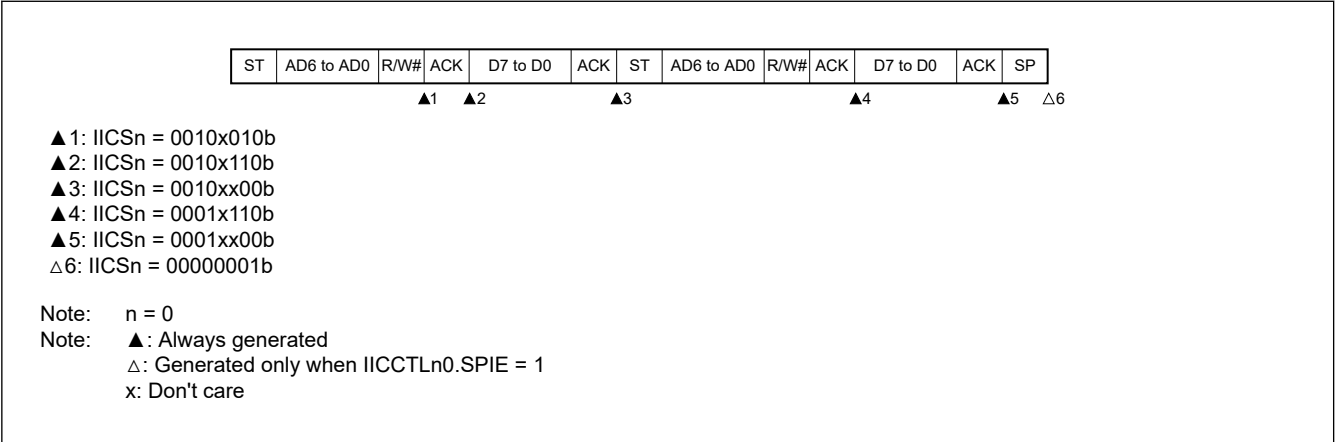


Figure 20.40 Slave device operation after code access, matches with SVA0 (IICCTL00.WTIM = 1)

(c) Start → Code → Data → Start → Code → Data → Stop

1. When IICCTL00.WTIM = 0 (after restart, extension code reception, the all address match function is disabled)

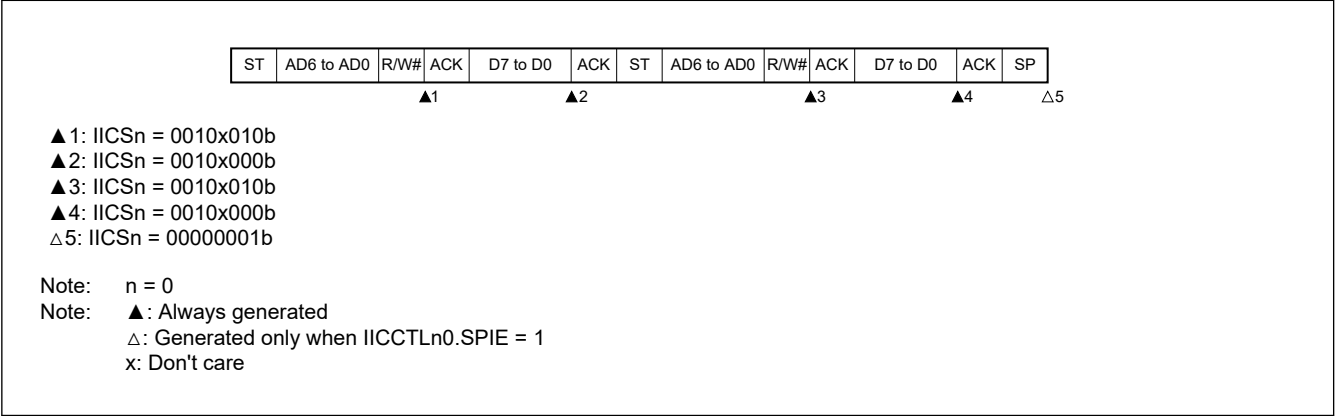


Figure 20.41 Slave device operation after code access, matches the extension code (IICCTL00.WTIM = 0)

(b) 开始→代码→数据→开始→地址→数据→停止

1. 当 IICCTL00.WTIM= 0 （重启后，与 SVA0 匹配，所有地址匹配功能被禁用）

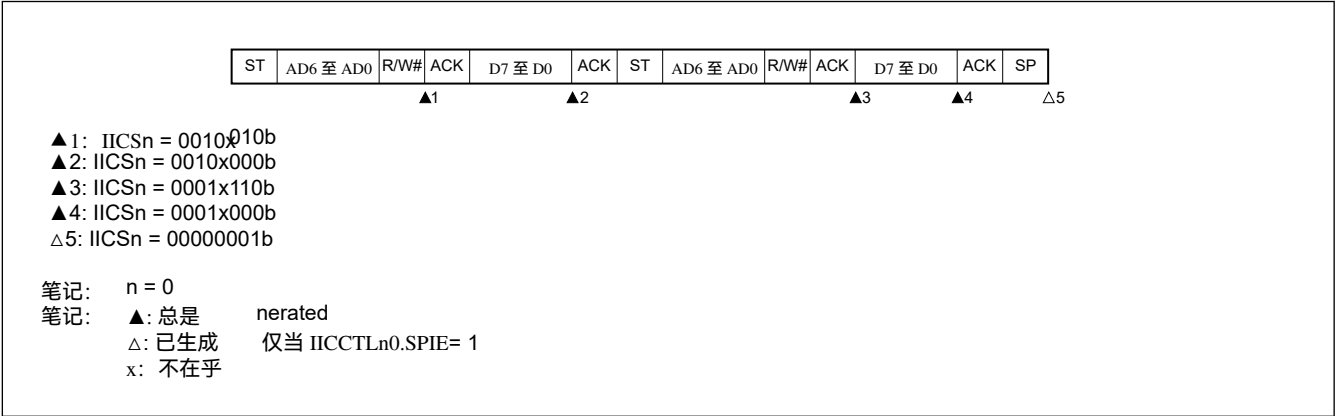


图 20.39 代码访问后从设备操作，与 SVA0 匹配 (IICCTL00.WTIM= 0)

2. 当 IICCTL00.WTIM= 1 （重启后，与 SVA0 匹配，所有地址匹配功能被禁用）

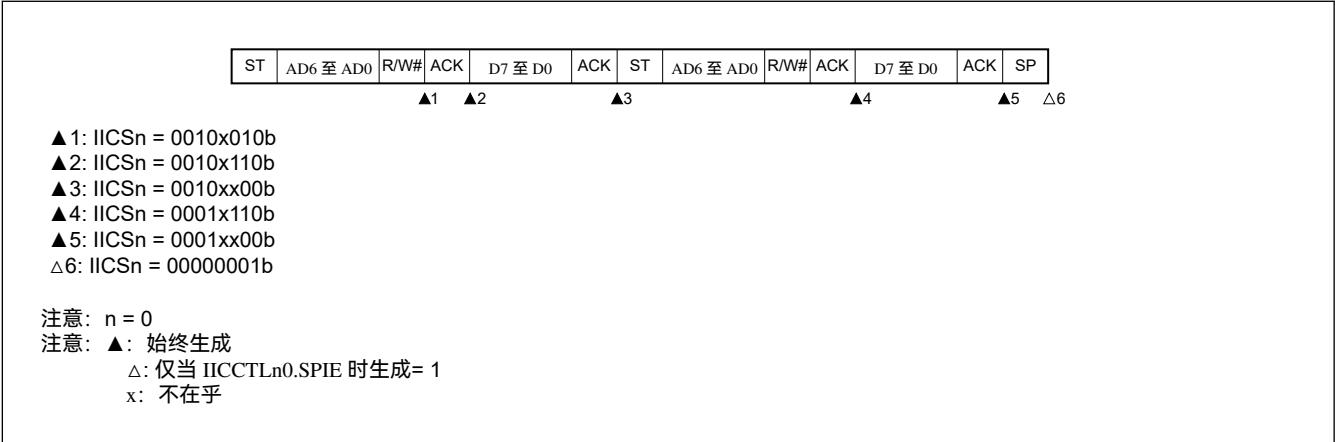


图 20.40 代码访问后从设备操作，与 SVA0 (IICCTL00.WTIM= 1) 匹配

(c) 开始→代码→数据→开始→公司de → Data → S顶部

1. 当 IICCTL00.WTIM= 0 （重启后，接收分机号，所有地址匹配功能被禁用）

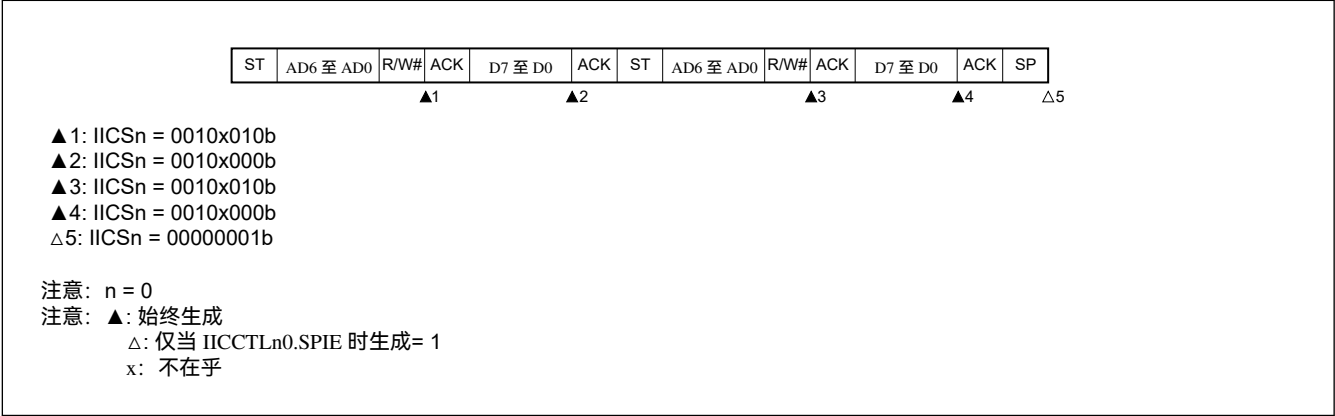


图 20.41 代码访问后从设备操作，与扩展代码 (IICCTL00.WTIM= 0) 匹配

2. When IICCTL00.WTIM = 1 (after restart, extension code reception, the all address match function is disabled)

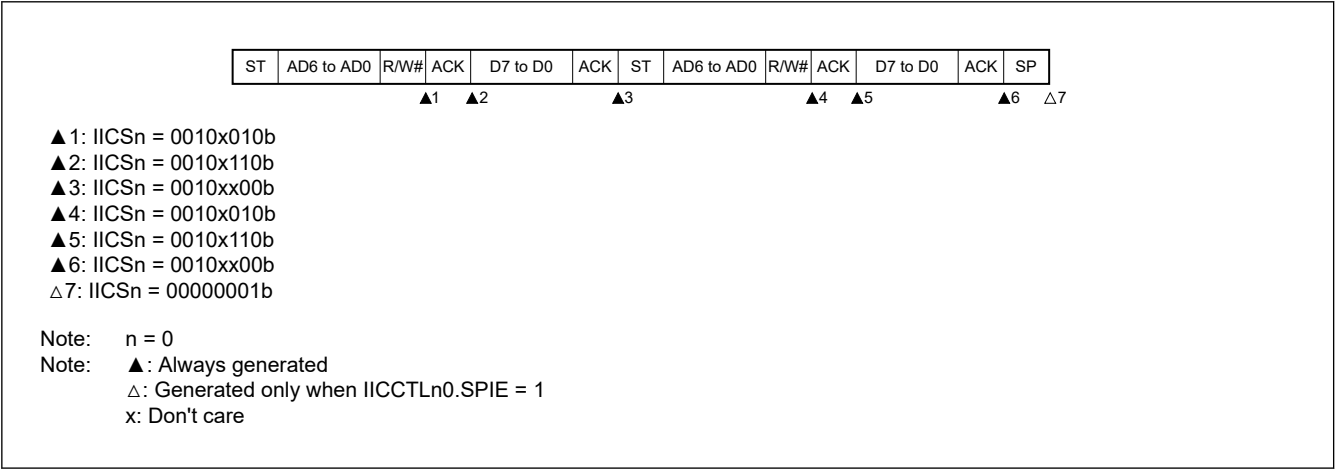


Figure 20.42 Slave device operation after code access, matches the extension code (IICCTL00.WTIM = 1)

(d) Start → Code → Data → Start → Address → Data → Stop

1. When IICCTL00.WTIM = 0

(after restart, does not match address (not extension code, the all address match function is disabled))

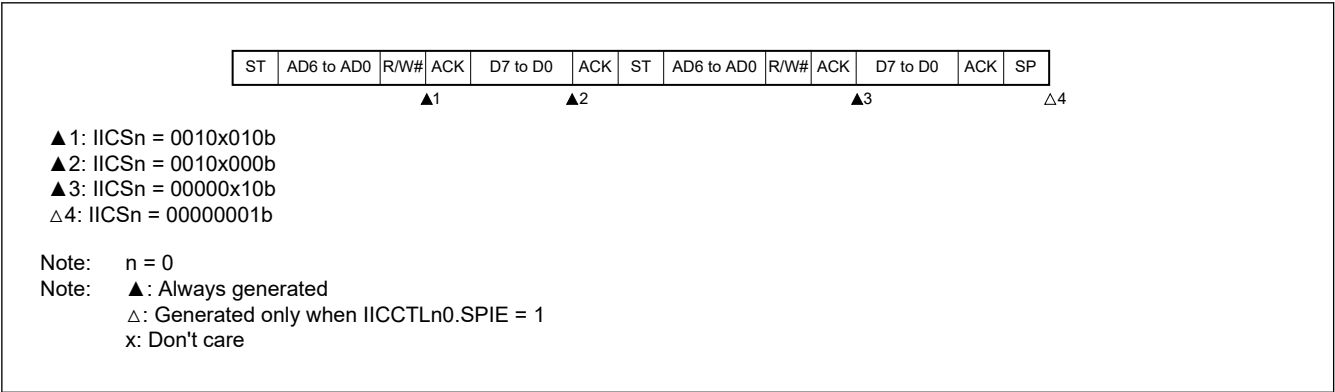


Figure 20.43 Slave device operation after code access, does not match (IICCTL00.WTIM = 0)

2. When IICCTL00.WTIM = 1

(after restart, does not match address (not extension code, the all address match function is disabled))

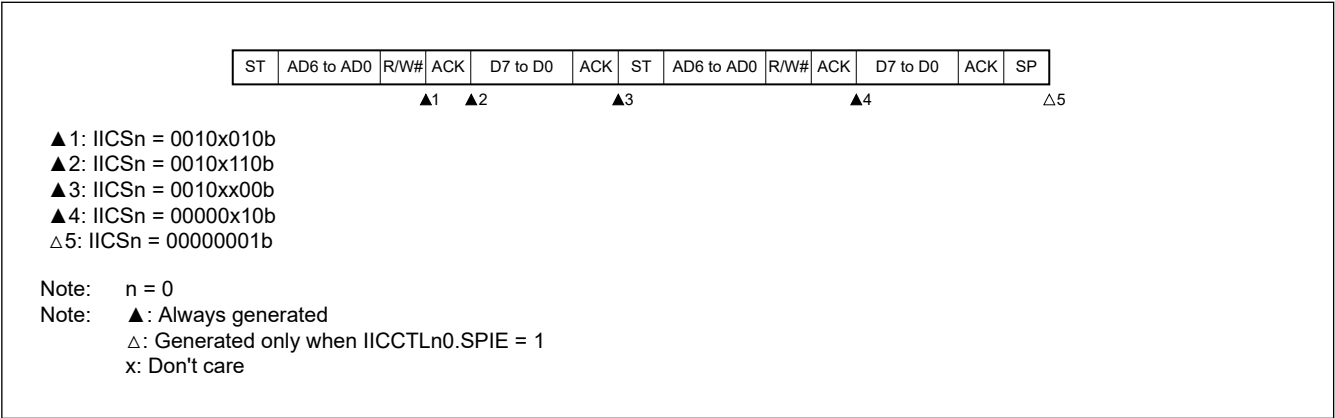


Figure 20.44 Slave device operation after code access, does not match (IICCTL00.WTIM = 1)

2. 当 IICCTL00.WTIM= 1 （重启后，接收分机号，所有地址匹配功能被禁用）

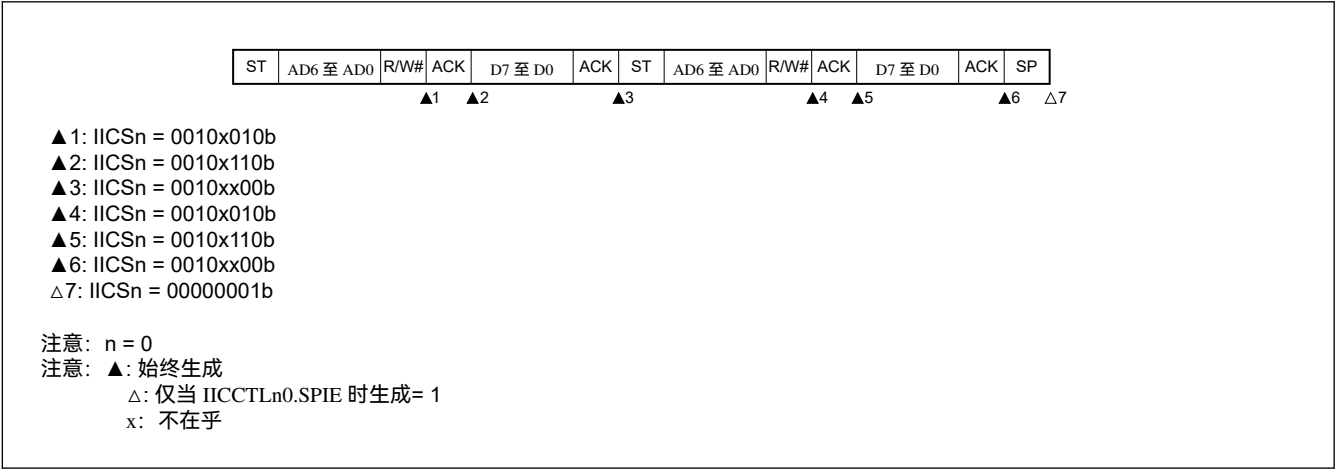


图 20.42 代码访问后从设备操作，与扩展代码 (IICCTL00.WTIM= 1) 匹配

(d) 开始→代码→数据→开始→地址→数据→停止

1. 当 IICCTL00.WTIM= 0

（重启后，地址不匹配（非分机号，所有地址匹配功能已禁用））

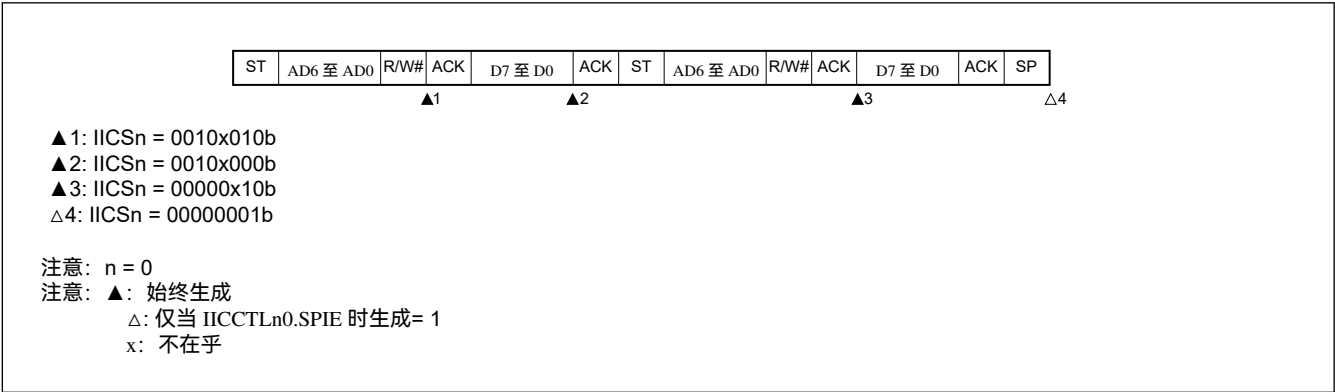


图 20.43 代码访问后从设备操作，与 (IICCTL00.WTIM= 0) 不匹配

2. 当 IICCTL00.WTIM= 1

（重启后，地址不匹配（非分机号，所有地址匹配功能已禁用））

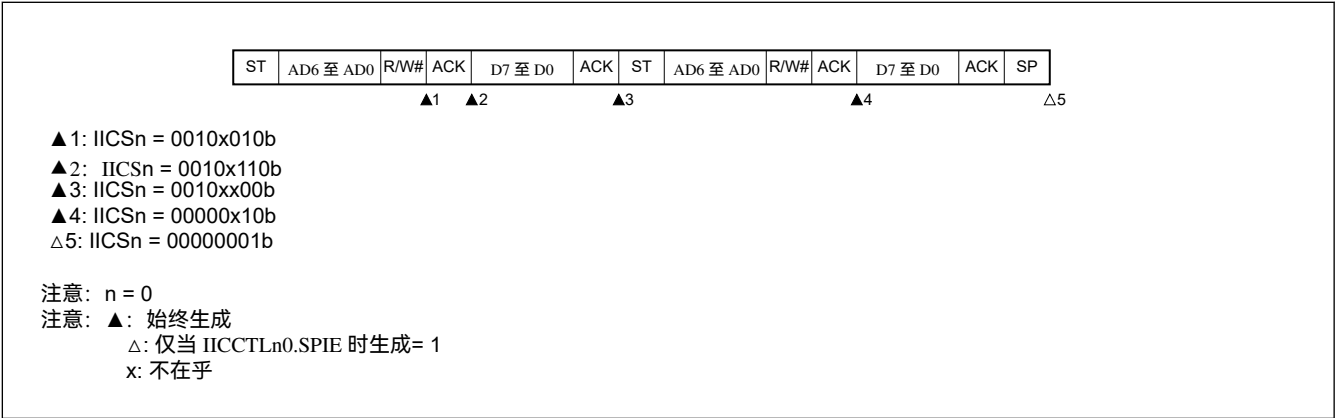


图 20.44 代码访问后从设备操作，与 (IICCTL00.WTIM= 1) 不匹配

(4) Operation without communication

(a) Start → Code → Data → Data → Stop

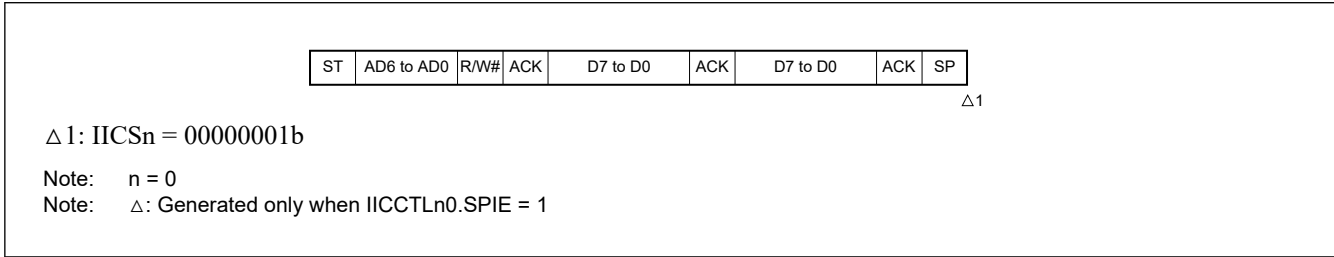


Figure 20.45 Operation without communication

(5) Arbitration loss operation (operation as slave mode after arbitration loss)

When the device is used as a master device in a multi-master system, read the IICS0.MSTS bit each time interrupt request signal IICA0_TXRXI has occurred to check the arbitration result.

(a) When arbitration loss occurs during transmission of slave address data

1. When IICCTL00.WTIM = 0

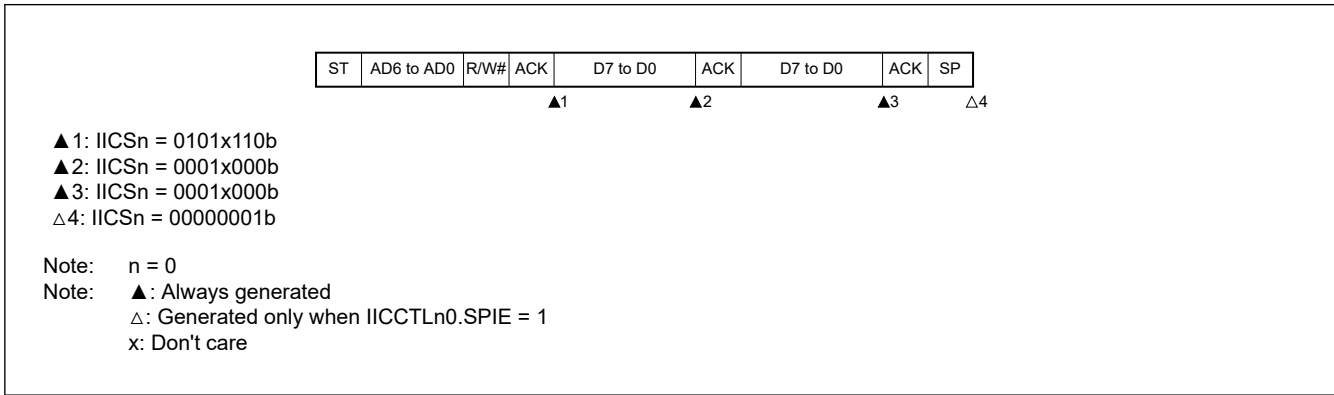


Figure 20.46 Arbitration loss when sending slave address data (IICCTL00.WTIM = 0)

2. When IICCTL00.WTIM = 1

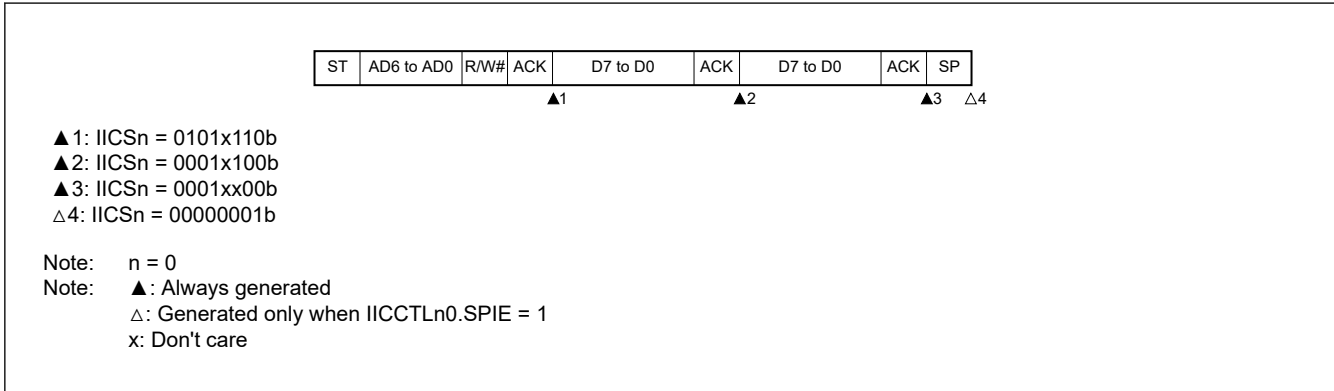


Figure 20.47 Arbitration loss when sending slave address data (IICCTL00.WTIM = 1)

(4) 无通信操作

(a) 开始 → 地址 → Data → Data → 结束

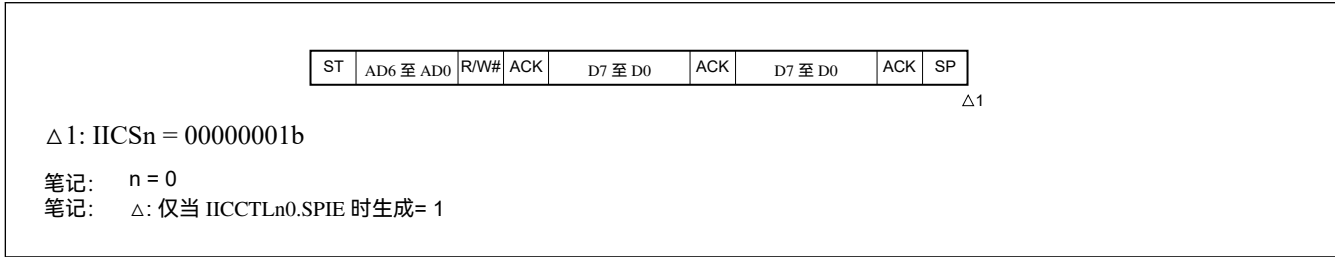


图 20.45 无通信操作

(5) 仲裁失败操作（仲裁失败后以从属模式运行）

当该设备用作多主系统中的主设备时，每次发生中断请求信号 IICA0_TXRXI 时，读取 IICS0.MSTS 位以检查仲裁结果。

(a) 当从地址数据传输过程中发生仲裁丢失时

1. 当 IICCTL00.WTIM = 0

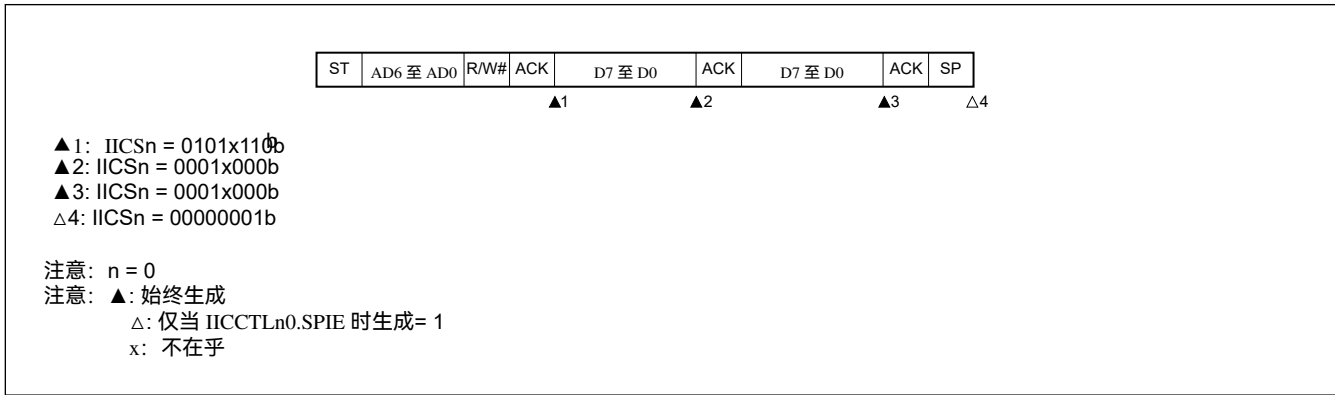


图 20.46 发送从地址数据时的仲裁丢失 (IICCTL00.WTIM = 0)

2. 当 IICCTL00.WTIM = 1

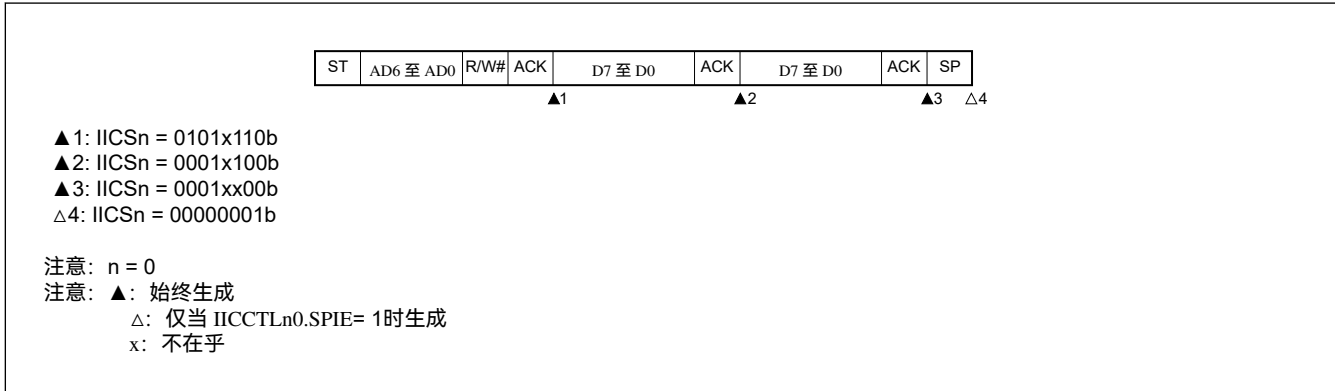


图 20.47 发送从地址数据时的仲裁丢失 (IICCTL00.WTIM = 1)

(b) When arbitration loss occurs during transmission of extension code (the all address match function is disabled)

1. When IICCTL00.WTIM = 0

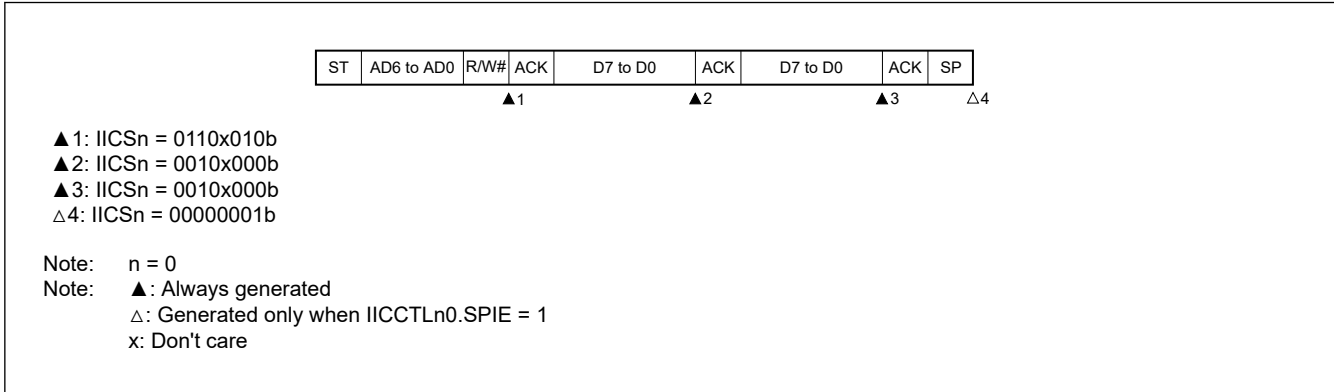


Figure 20.48 Arbitration loss when sending extension code (IICCTL00.WTIM = 0)

2. When IICCTL00.WTIM = 1

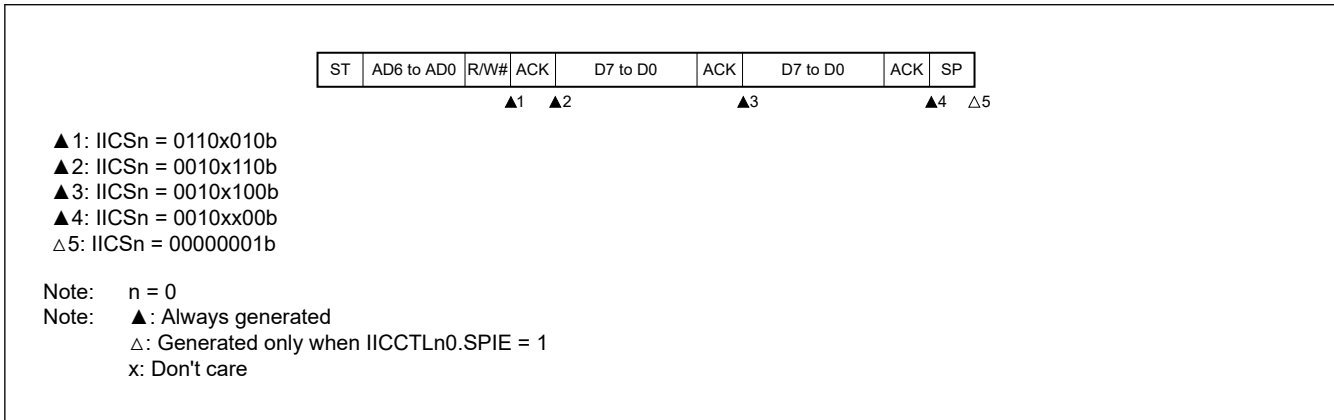


Figure 20.49 Arbitration loss when sending extension code (IICCTL00.WTIM = 1)

(6) Operation when arbitration loss occurs (no communication after arbitration loss)

When the device is used as a master device in a multi-master system, read the IICS0.MSTS bit each time interrupt request signal IICA0_TXRXI has occurred to check the arbitration result.

(a) When arbitration loss occurs during transmission of slave address data (when IICCTL00.WTIM = 1)

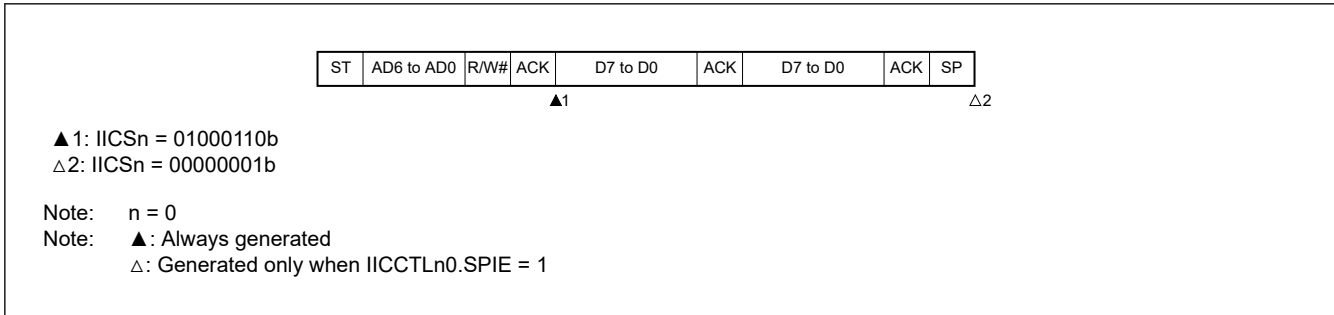


Figure 20.50 Operation when arbitration loss occurs during slave address data transmission (IICCTL00.WTIM = 1)

WTIM= 0) (b) 当扩展码传输过程中发生仲裁丢失时（所有地址匹配功能被禁用）

1. When IICCTL00.WTIM = 0

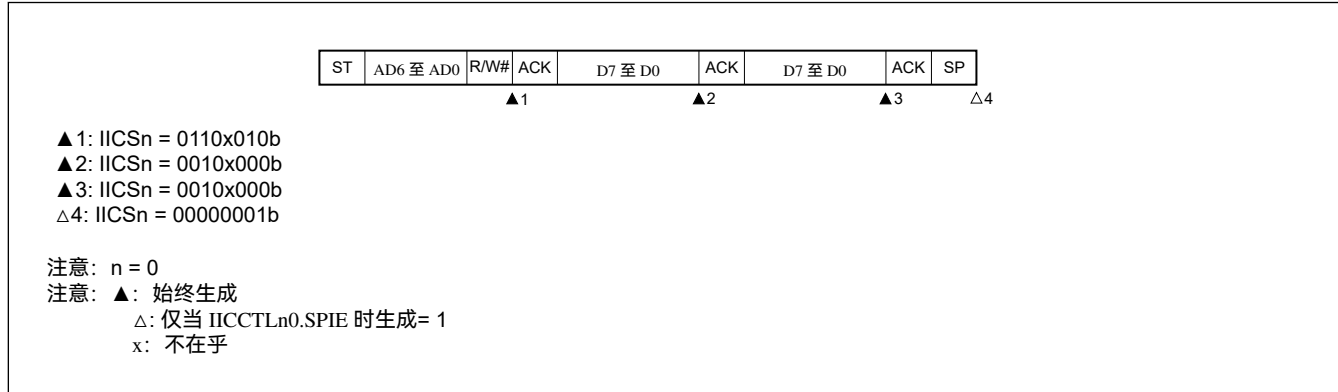


图 20.48 发送扩展码时的仲裁丢失 (IICCTL00.WTIM= 0)

2. 当 IICCTL00.WTIM= 1

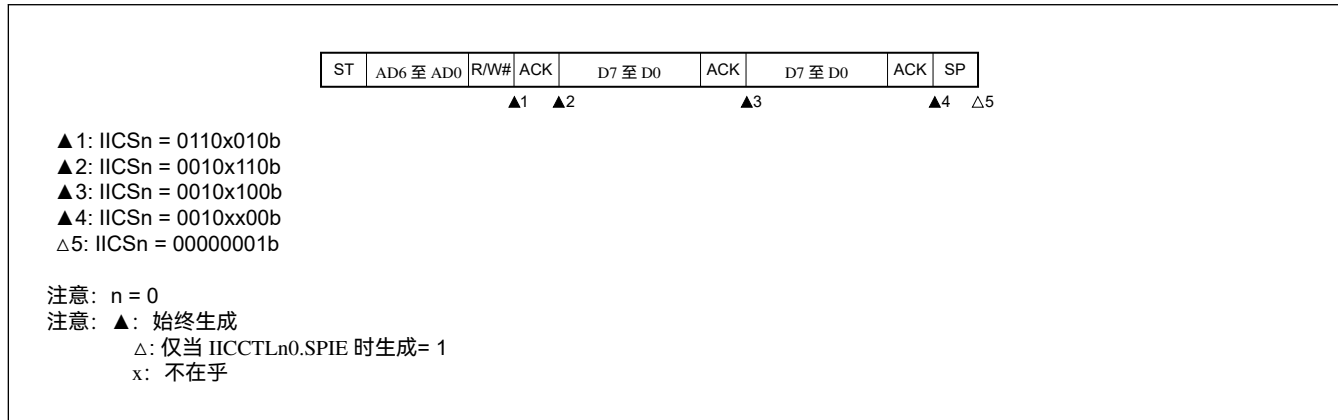


图 20.49 发送扩展码时的仲裁丢失 (IICCTL00.WTIM= 1)

(6) 仲裁失败时的操作（仲裁失败后无通讯）

当该设备用作多主系统中的主设备时，每次发生中断请求信号 IICA0_TXRXI 时，读取 IICS0.MSTS 位以检查仲裁结果。

(a) 当从地址数据传输期间发生仲裁丢失时（当 IICCTL00.WTIM= 1时）

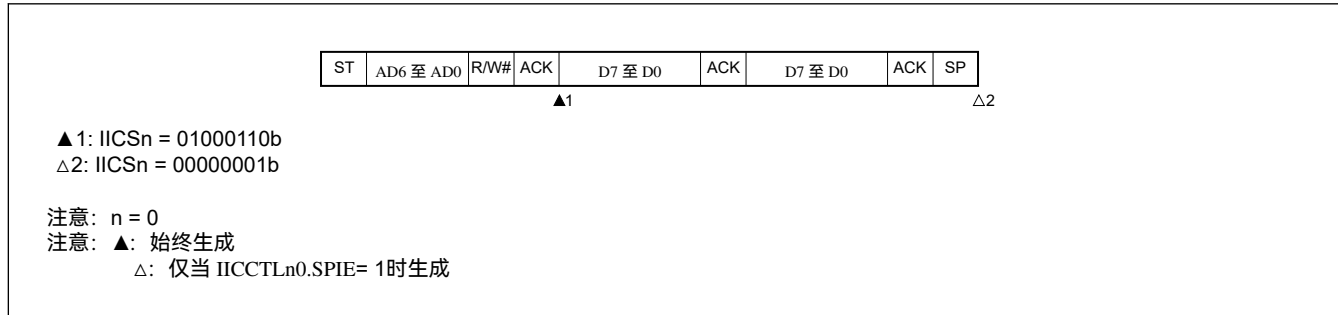


图 20.50 当从地址数据传输过程中发生仲裁丢失时的操作 (IICCTL00.WTIM= 1)

(b) When arbitration loss occurs during transmission of extension code (the all address match function is disabled)

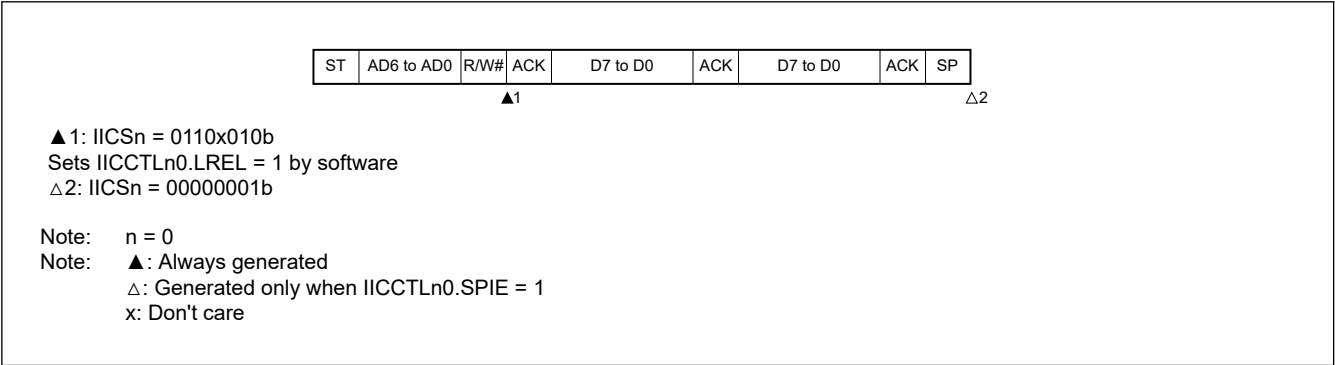


Figure 20.51 Operation when arbitration loss occurs during extension code transmission

(c) When arbitration loss occurs during transfer of data

1. When IICCTL00.WTIM = 0

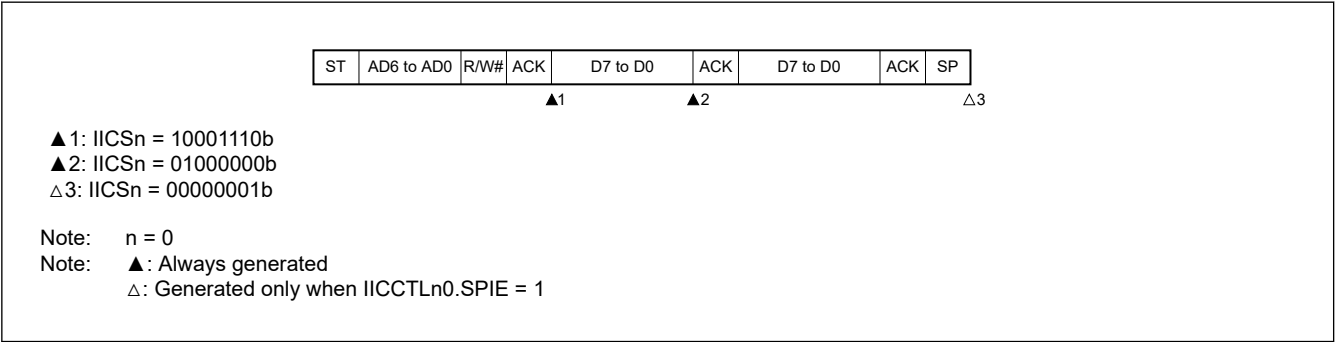


Figure 20.52 Operation when arbitration loss occurs during data transfer (IICCTL00.WTIM = 0)

2. When IICCTL00.WTIM = 1

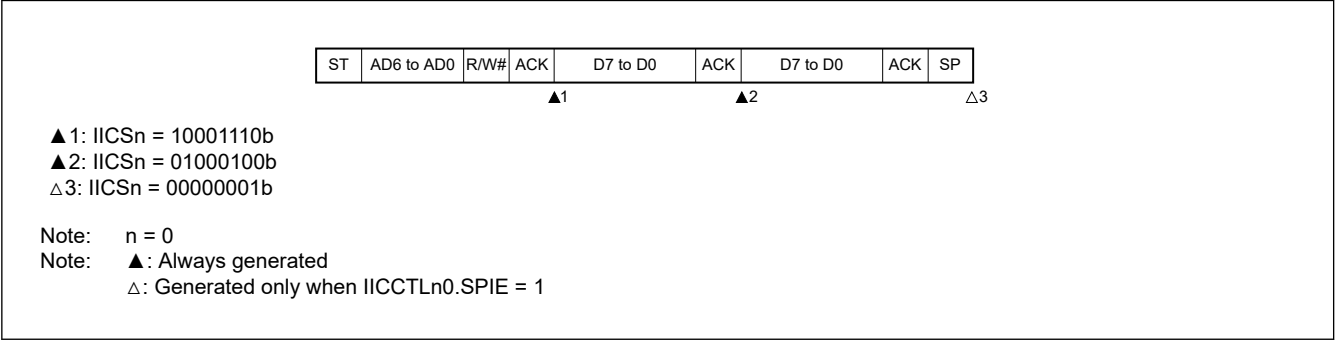


Figure 20.53 Operation when arbitration loss occurs during data transfer (IICCTL00.WTIM = 1)

WTIM= 1) (b) 当扩展码传输过程中发生仲裁丢失时（所有地址匹配功能被禁用）

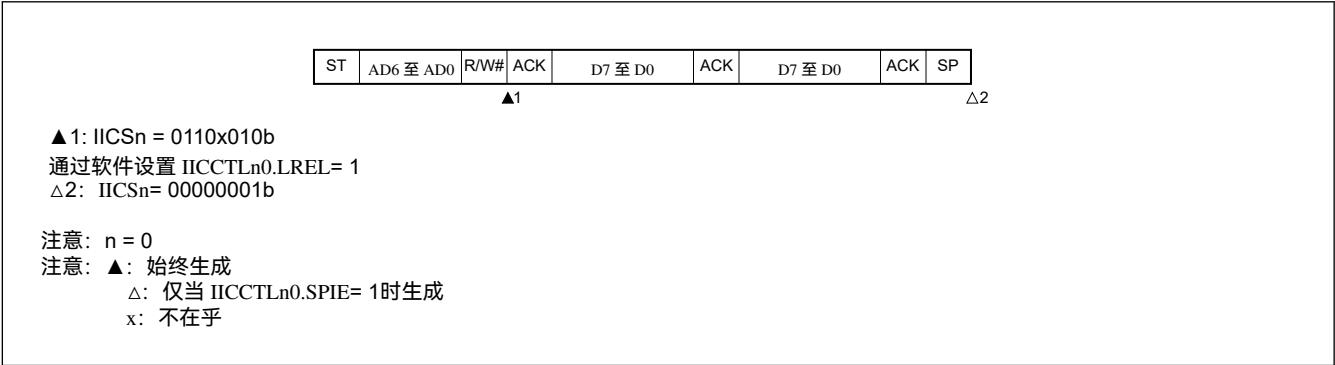


图 20.51 扩展码传输过程中发生仲裁丢失时的操作

(c) 数据传输过程中发生仲裁损失时

1. 当 IICCTL00.WTIM = 0

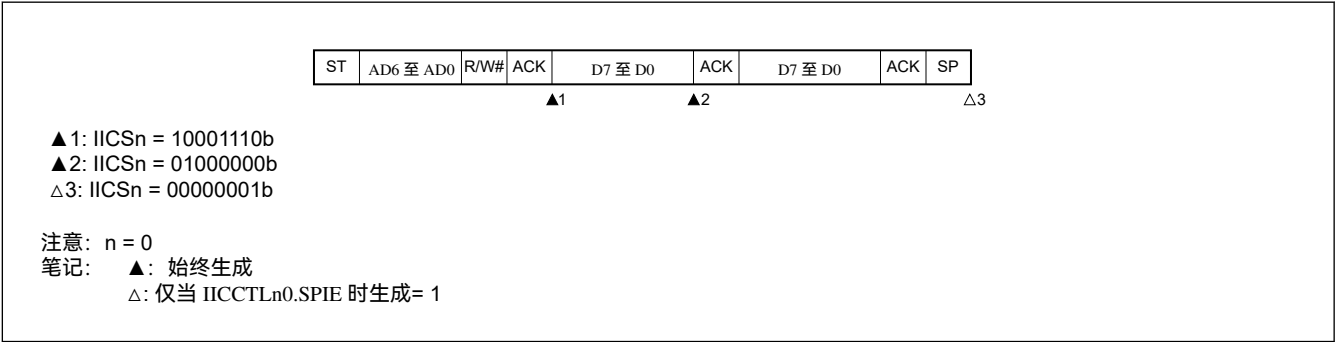


图 20.52 数据传输过程中发生仲裁丢失时的操作 (IICCTL00.WTIM= 0)

2. 当 IICCTL00 时。

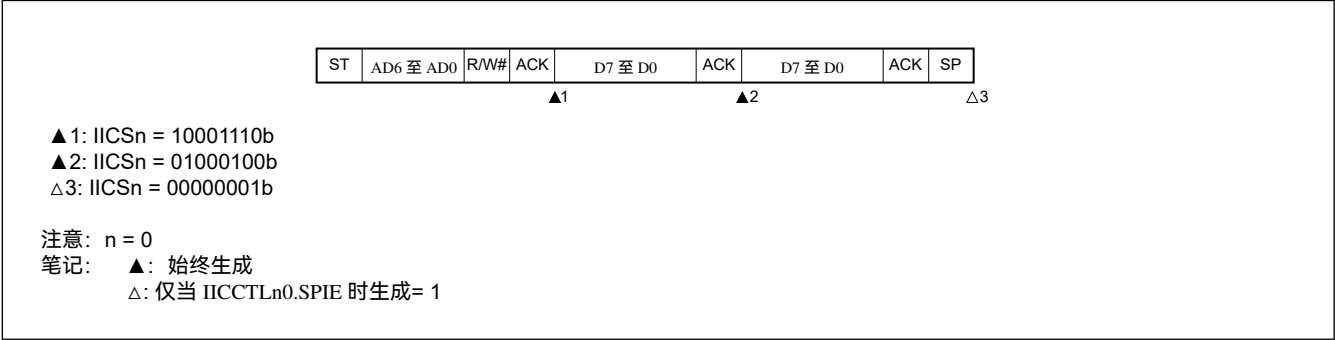


图 20.53 数据传输过程中发生仲裁丢失时的操作 (IICCTL00.WTIM= 1)

(d) When loss occurs due to restart condition during data transfer

1. Not extension code (Example: If it does not match SVA0, the all address match function is disabled)

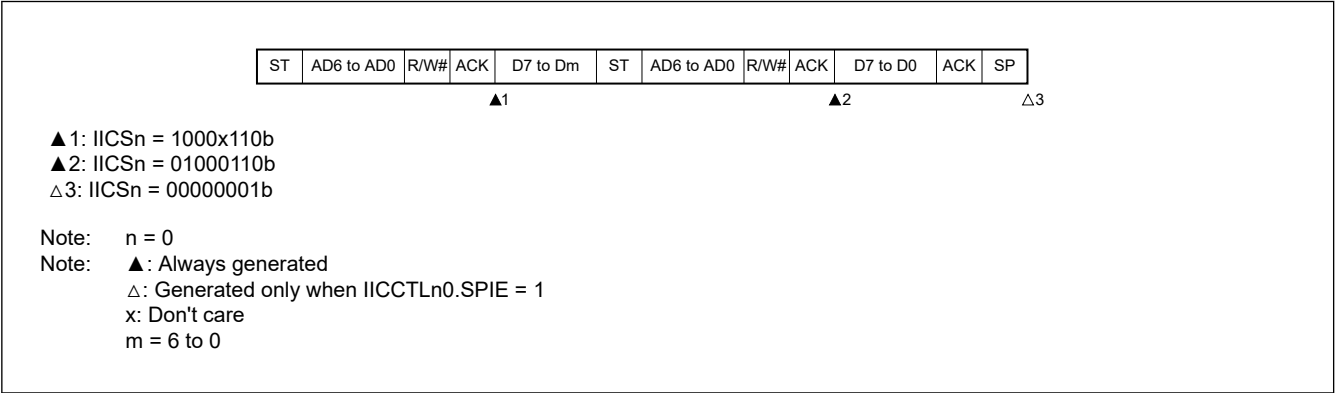


Figure 20.54 Operation when arbitration loss occurs due to restart during data transfer (not extension code)

2. Extension code (the all address match function is disabled)

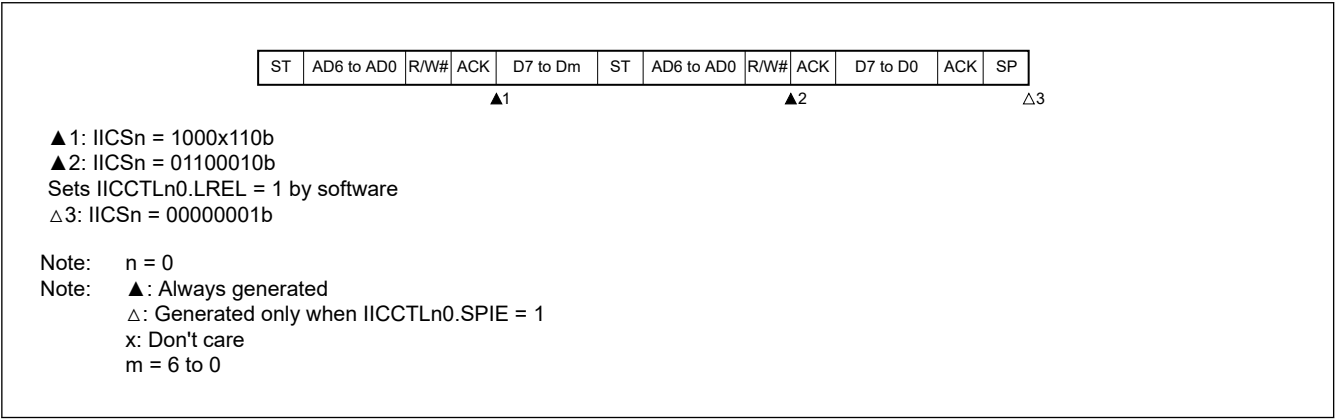


Figure 20.55 Operation when arbitration loss occurs due to restart during data transfer (extension code)

(e) When loss occurs due to stop condition during data transfer

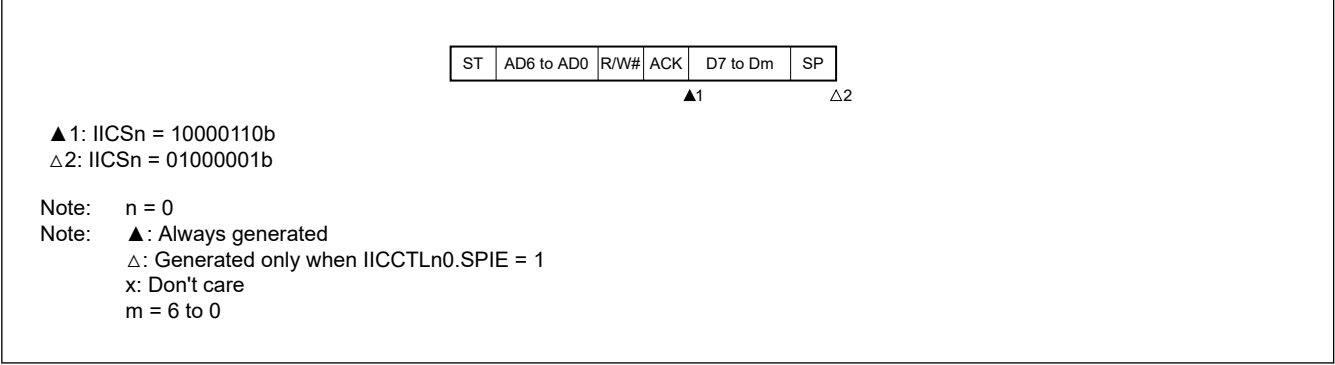


Figure 20.56 Operation when arbitration loss occurs due to stop condition during data transfer

WTIM (d) 数据传输过程中因重启而发生丢失时

1. 非扩展代码（例如：如果与 SVA0 不匹配，则禁用所有地址匹配功能）

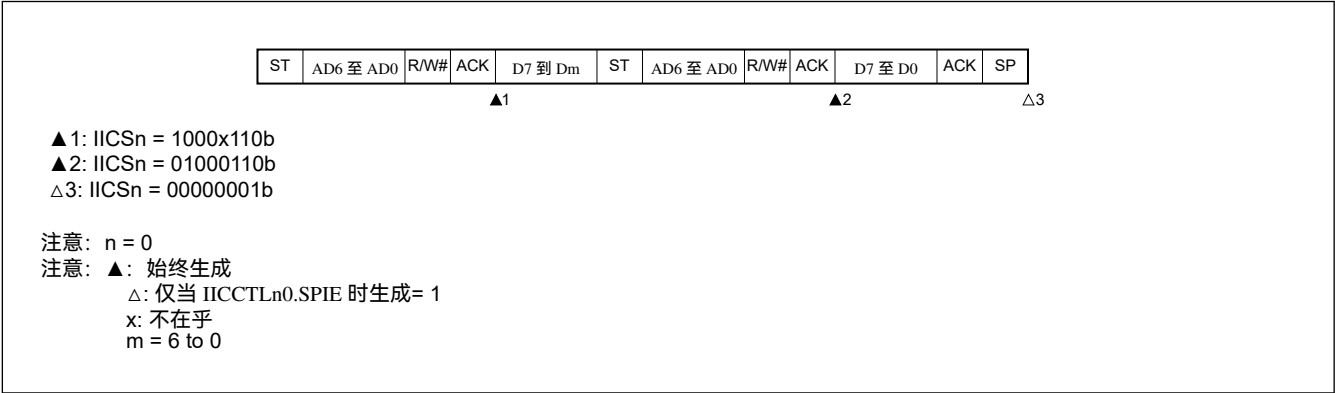


图 20.54 数据传输过程中因重启而导致仲裁丢失时的操作（非扩展码）

2. 分机号（所有地址匹配功能已禁用）

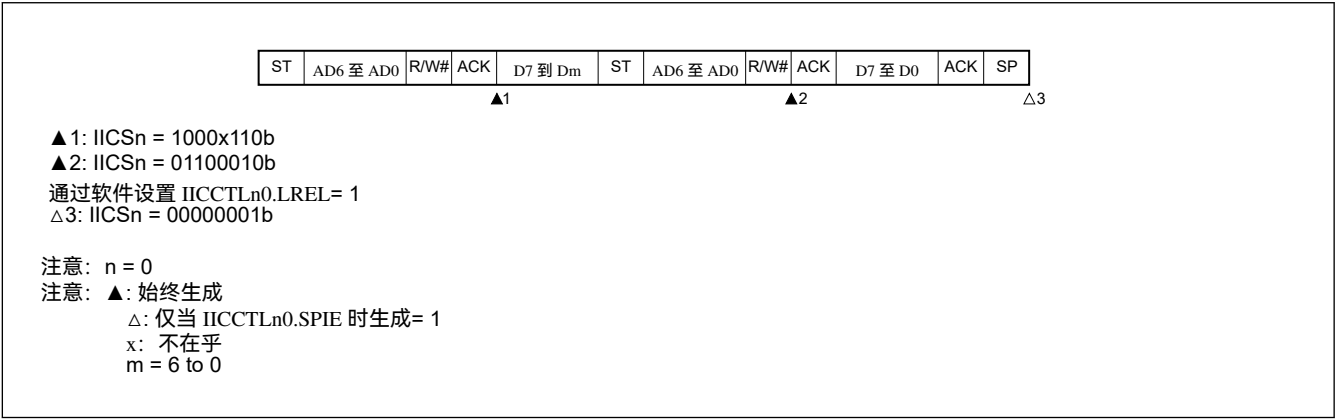


图 20.55 数据传输过程中因重启而导致仲裁丢失时的操作（扩展码）

(e) 数据传输过程中因停止条件导致数据丢失时

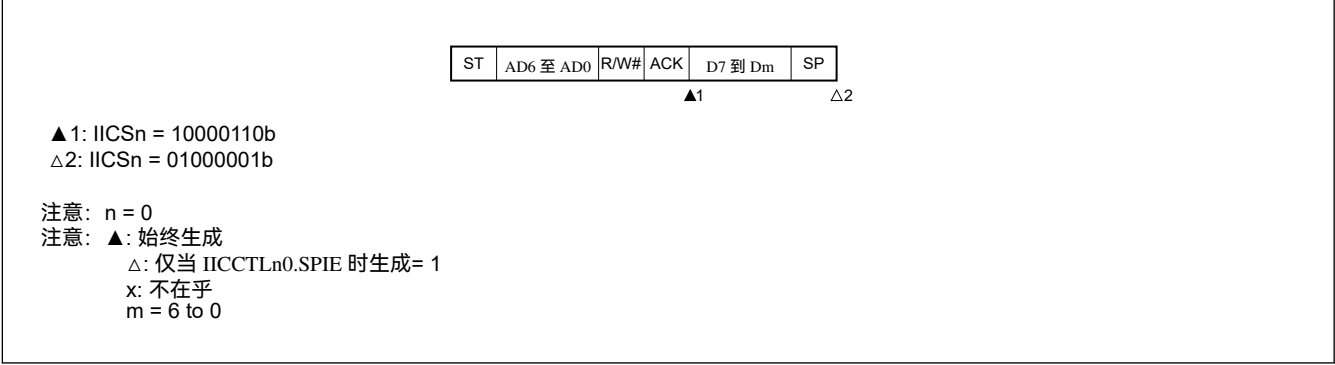


图 20.56 数据传输过程中因停止状态导致仲裁丢失时的操作

(f) When arbitration loss occurs due to low-level data when attempting to generate a restart condition

1. When IICCTL00.WTIM = 0

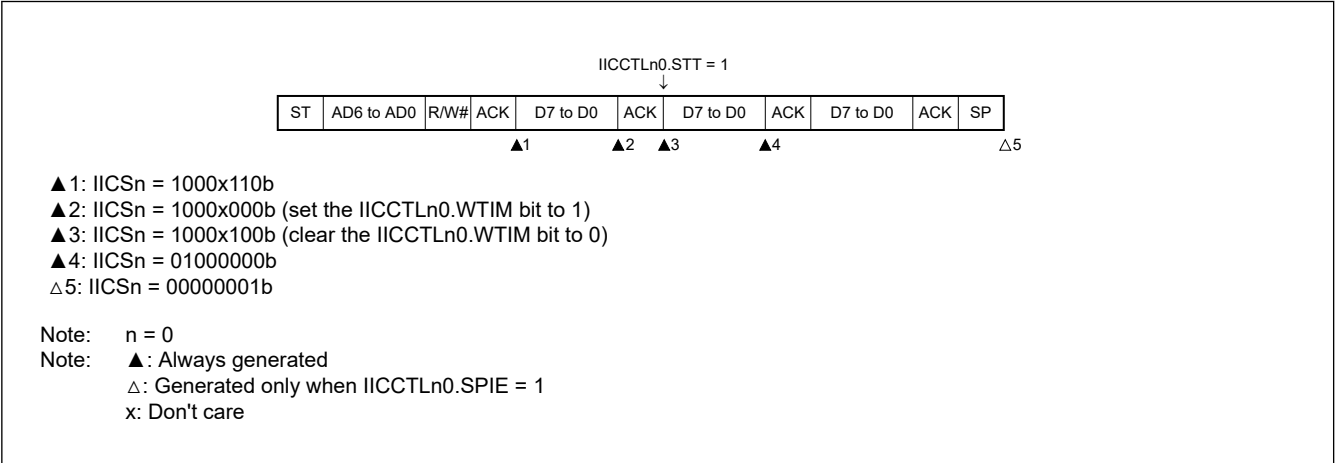


Figure 20.57 Operations when arbitration loss occurs due to low-level data when attempting to generate a restart condition (IICCTL00.WTIM = 0)

2. When IICCTL00.WTIM = 1

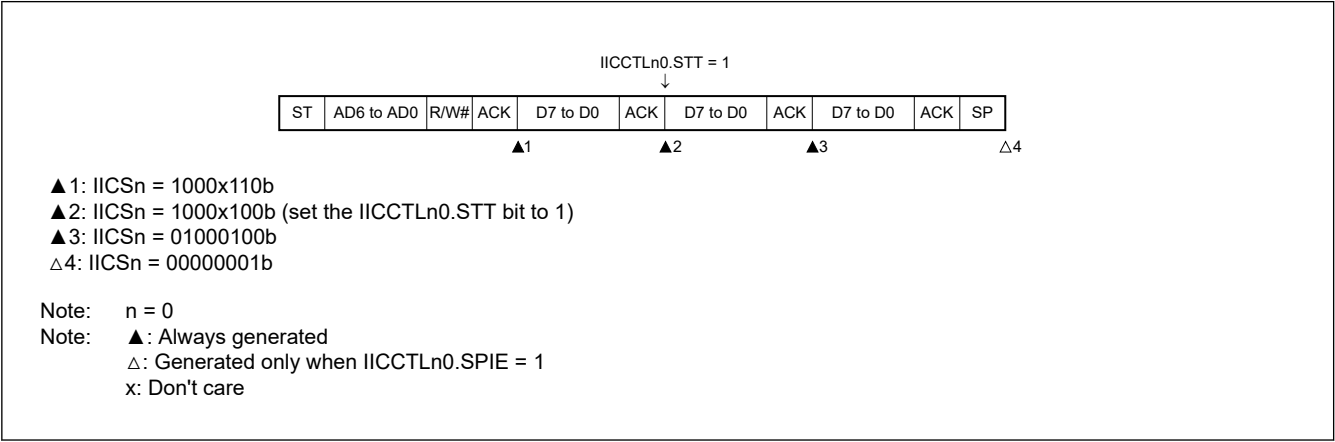


Figure 20.58 Operations when arbitration loss occurs due to low-level data when attempting to generate a restart condition (IICCTL00.WTIM = 1)

(f) 当尝试生成重启条件时，由于低级数据而导致仲裁丢失

1. 当 IICCTL00.WTIM= 0

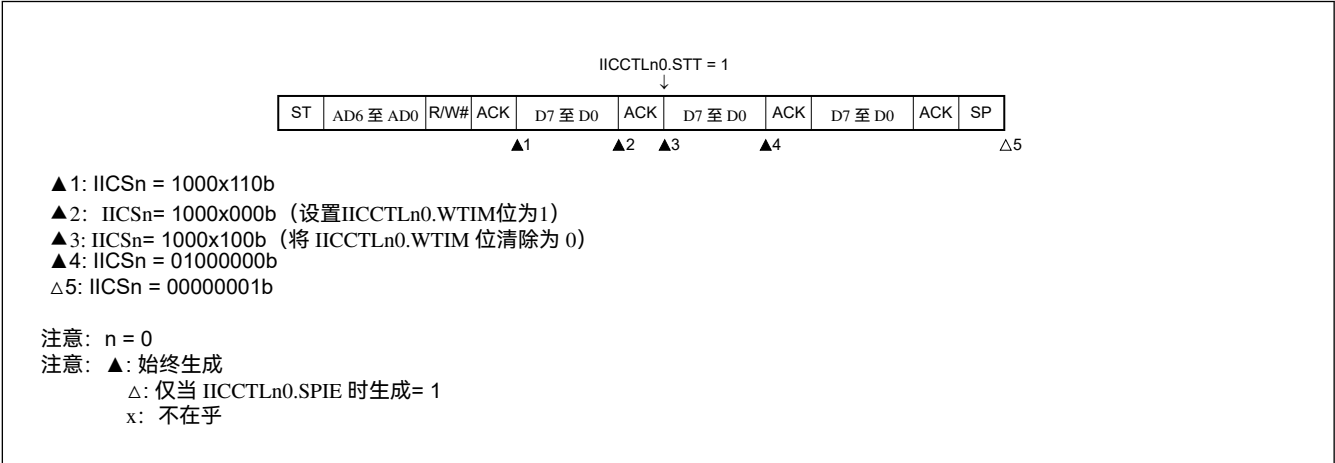


图 20.57 当尝试生成重启条件时，由于低级数据导致仲裁丢失而发生的操作 (IICCTL00.WTIM= 0)

2. 当 IICCTL00.WTIM= 1

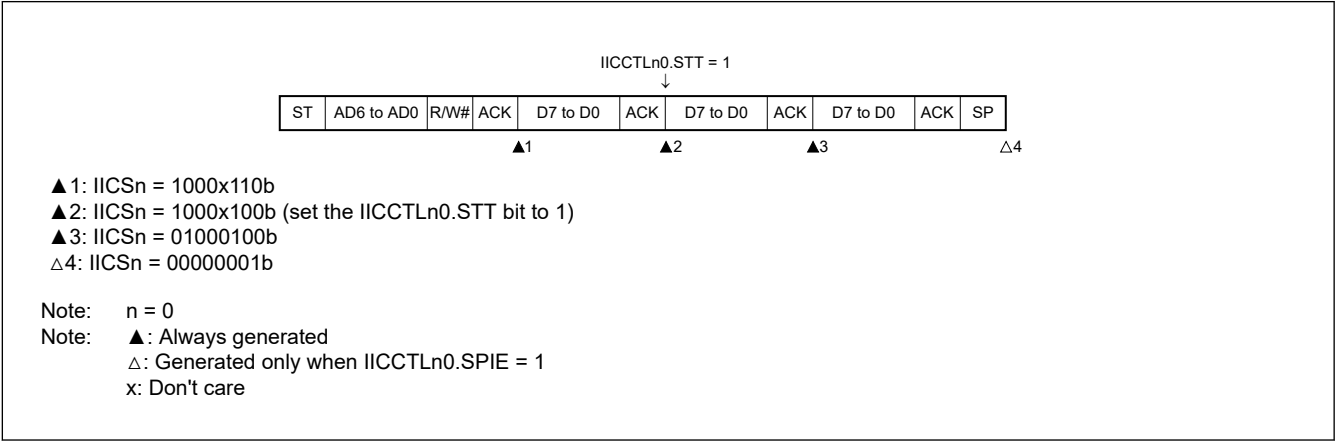


图 20.58 当尝试生成重启条件时，由于低级数据而发生仲裁丢失的操作 (IICCTL00)。

(g) When arbitration loss occurs due to a stop condition when attempting to generate a restart condition

1. When IICCTL00.WTIM = 0

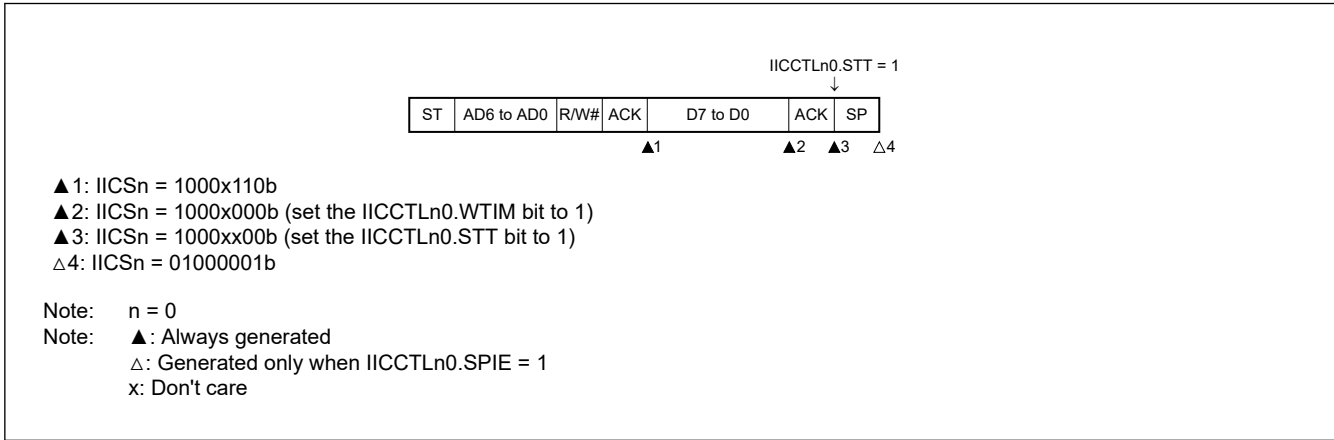


Figure 20.59 Operations when arbitration loss occurs due to stop condition when attempting to generate a restart condition (IICCTL00.WTIM = 0)

2. When IICCTL00.WTIM = 1

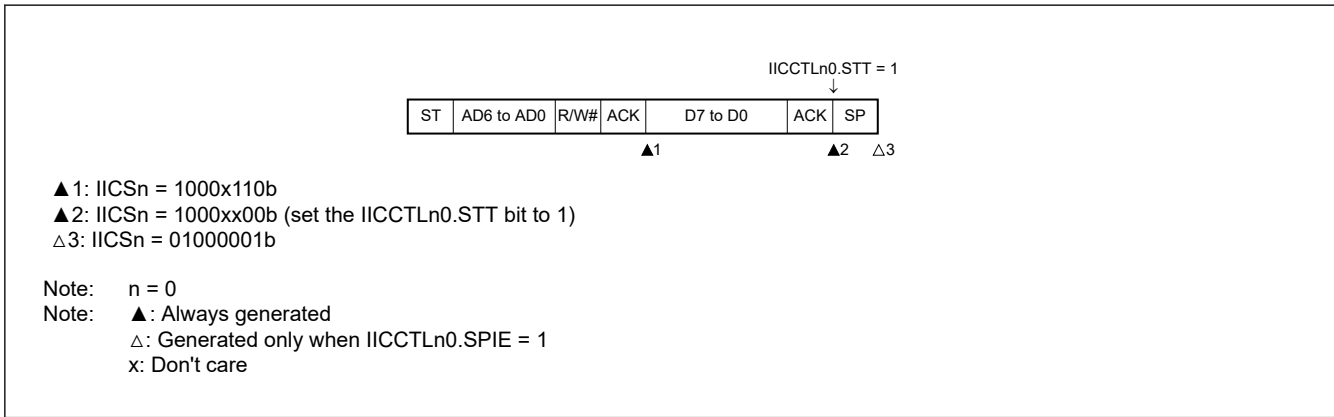


Figure 20.60 Operations when arbitration loss occurs due to stop condition when attempting to generate a restart condition (IICCTL00.WTIM = 1)

(h) When arbitration loss occurs due to low-level data when attempting to generate a stop condition

WTIM= 1)g) 当尝试生成重启条件时，由于停止条件而发生仲裁丢失时

1. 当 IICCTL00.WTIM= 0

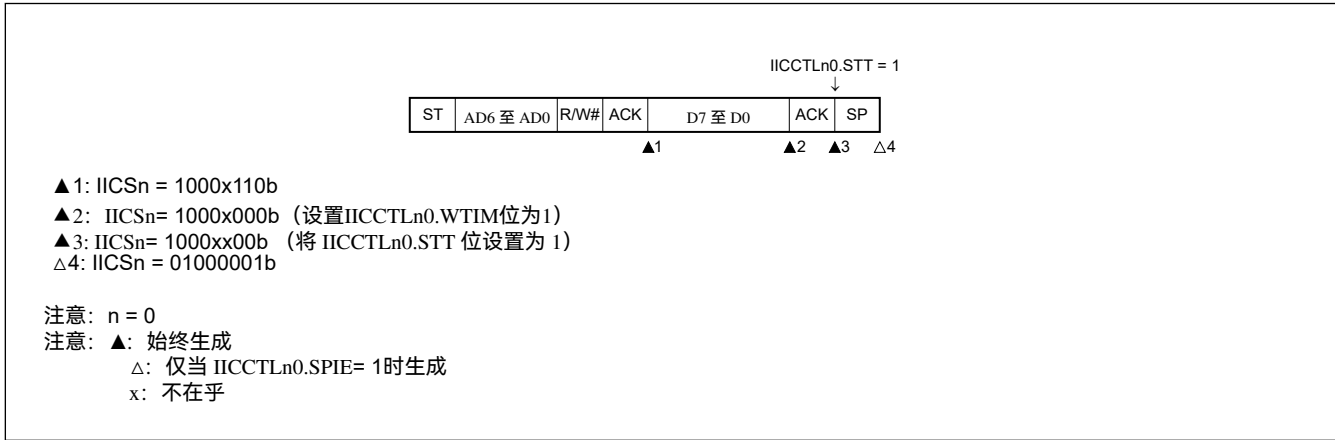


图 20.59 当尝试生成重启条件时，由于停止条件导致仲裁丢失的操作 (IICCTL00.WTIM= 0)

2. 当 IICCTL00.WTIM= 1

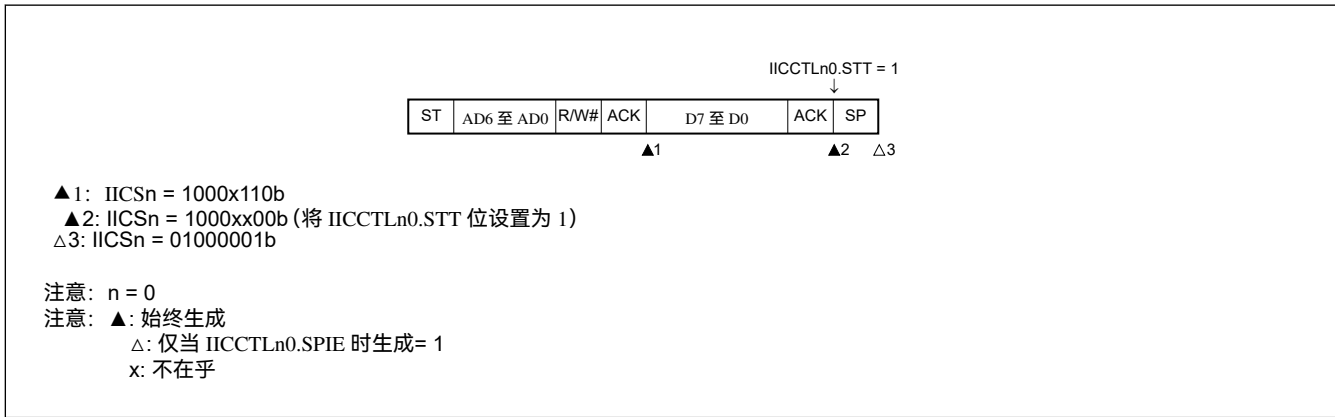


Figure 20.60 当尝试生成重启条件时，由于停止条件导致仲裁失败的操作(IICCTL00.WTIM = 1)

(h) 当尝试生成停止条件时，由于低级数据而导致仲裁失败

1. When IICCTL00.WTIM = 0

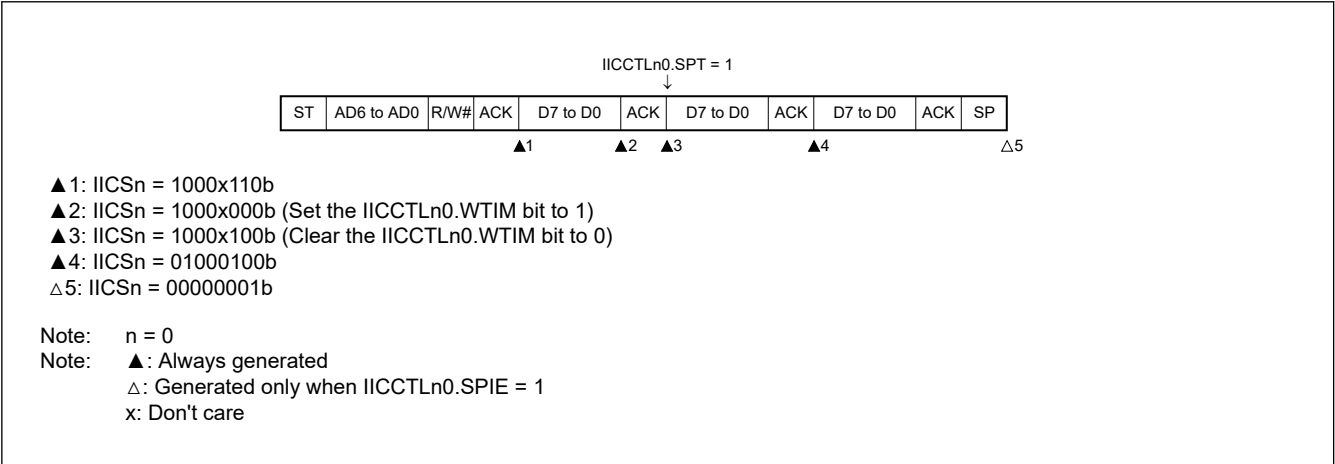


Figure 20.61 Operations when arbitration loss occurs due to S low-level data when attempting to generate a stop condition (IICCTL00.WTIM = 0)

2. When IICCTL00.WTIM = 1

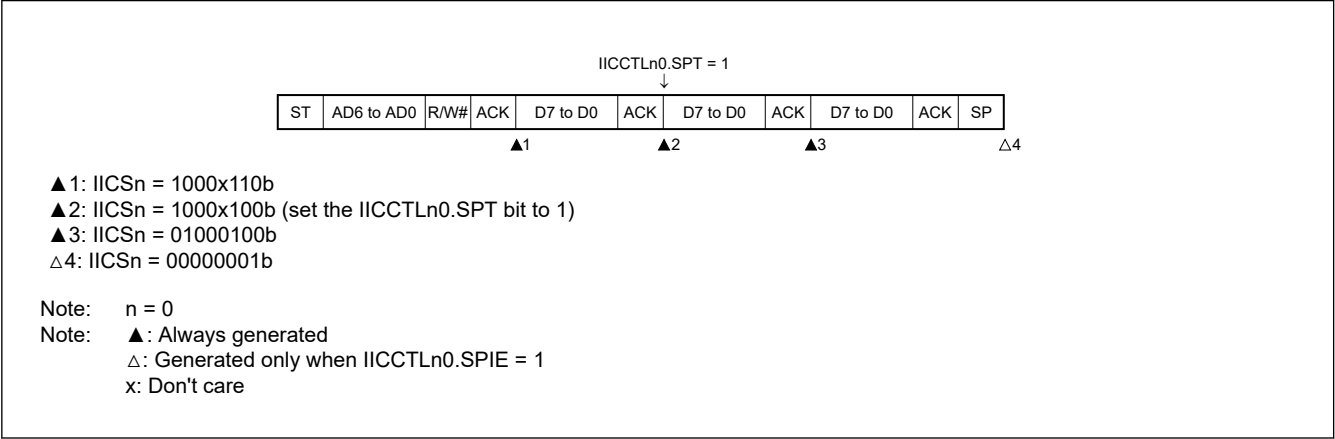


Figure 20.62 Operations when arbitration loss occurs due to S low-level data when attempting to generate a stop condition (IICCTL00.WTIM = 1)

20.4 Timing Charts

When using the I²C bus mode, the master device outputs an address through the serial bus to select one of several slave devices as its communication partner.

After outputting the slave address, the master device transmits the TRC bit (bit 3 of the IICA status register 0 (IICS0)), which specifies the data transfer direction, and then starts serial communication with the slave device.

(1) Example of master device to slave device communications (when the master device and the slave device insert clock stretching on the 9th cycle) and (2) Example of slave device to master device communications (8th cycle clock stretching is selected for the master device and 9th cycle clock stretching is selected for the slave device) show timing charts of the data communication.

The shift operation of the IICA shift register 0 (IICA0) is synchronized with the falling edge of the serial clock (SCLA0). The transmit data is transferred to the SO latch and is output (MSB first) using the SDAA0 pin.

Data input through the SDAA0 pin is captured into IICA0 at the rising edge of SCLA0.

In the timing diagrams described in this section, it is assumed that the all address match function is disabled.

1. 当 IICCTL00.WTIM= 0

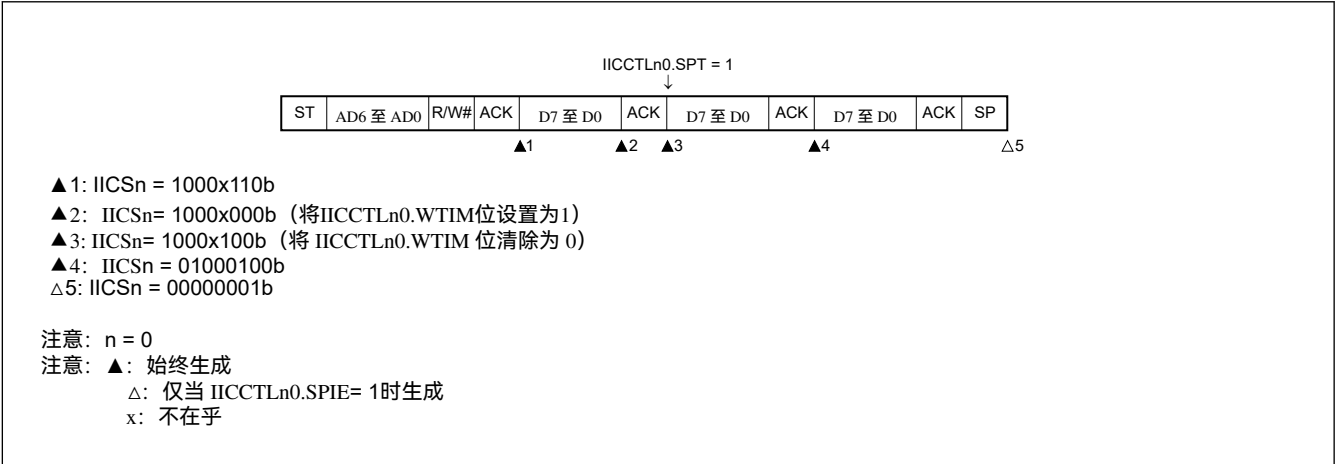


图 20.61 当尝试生成停止条件时，由于 S 低级数据导致仲裁丢失的操作 (IICCTL00.WTIM= 0)

2. 当 IICCTL00.WTIM= 1

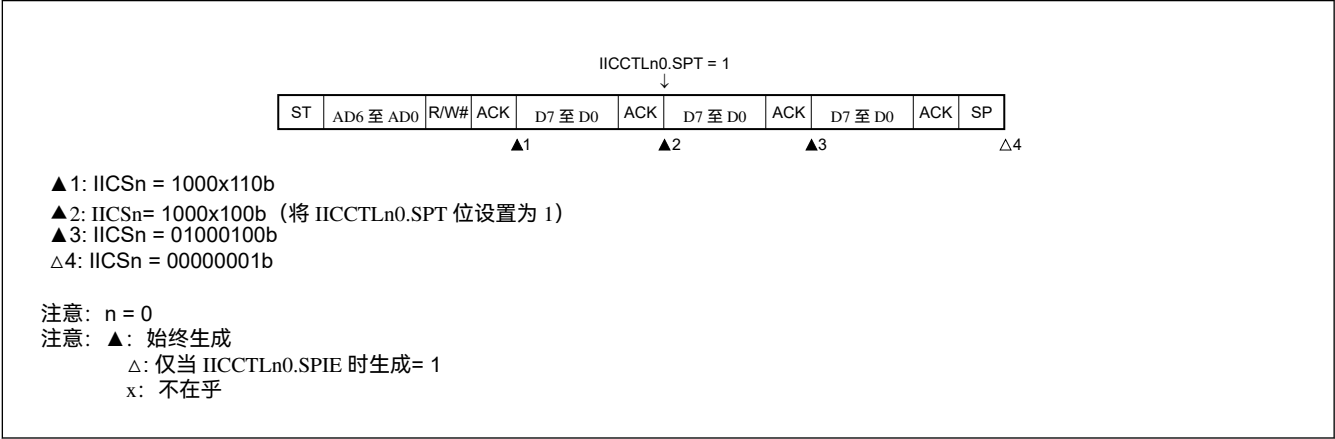


图 20.62 当尝试生成停止条件时，由于 S 低级数据而导致仲裁丢失的操作(IICCTL00.WTIM = 1)

20.4 时序图

使用I²C总线模式时，主设备通过串行总线输出地址，以选择多个从设备中的一个作为其通信伙伴。

主设备输出从设备地址后，发送 TRC 位（IICA 状态寄存器 0 (IICS0) 的第 3 位），该位指定数据传输方向，然后开始与从设备进行串行通信。

(1) 主设备到从设备通信示例（当主设备和从设备在第 9 个周期插入时钟拉伸时）和 (2) 从设备到主设备通信示例（主设备选择第 8 个周期时钟拉伸，从设备选择第 9 个周期时钟拉伸）显示了数据通信的时序图。

IICA 移位寄存器 0 (IICA0) 的移位操作与串行时钟 (SCLA0) 的下降沿同步。发送的数据被传输到 SO 锁存器，并使用 SDAA0 引脚输出（MSB 先输出）。

通过 SDAA0 引脚输入的数据在 SCLA0 的上升沿被捕获到 IICA0 中。

在本节描述的时序图中，假设所有地址匹配功能均已禁用。

(1) Example of master device to slave device communications (when the master device and the slave device insert clock stretching on the 9th cycle)

1. Start condition → address → data

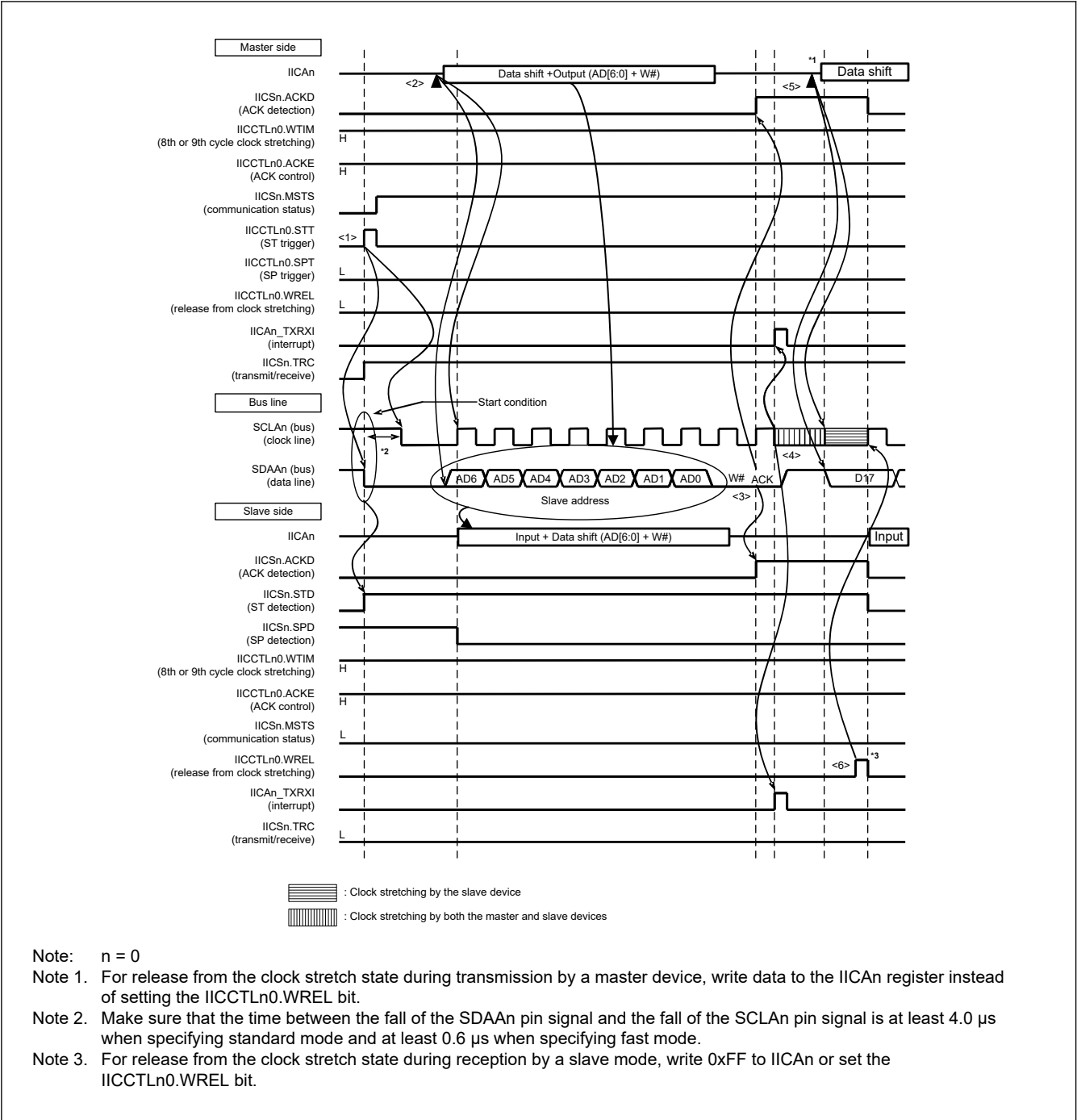


Figure 20.63 Example of master device to slave device communications (when the master device and the slave device insert clock stretching on the 9th cycle) (1/4)

The meanings of <1> to <6> in Figure 20.63 are explained below.

<1> The start condition trigger is set by the master device (IICCTL00.STT = 1) and a start condition (SCLAn = 1 and SDAAn changes from 1 to 0) is generated once the bus data line goes low (SDAA0).

When the start condition is subsequently detected, the master device enters the master mode communication status (IICSn.MSTS = 1). The master device is ready to communicate once the bus clock line goes low (SCLAn = 0) after the hold time has elapsed.

(1) 主设备与从设备通信示例（当主设备和从设备在第 9 个周期插入时钟拉伸时）

1. 起始条件→地址→数据

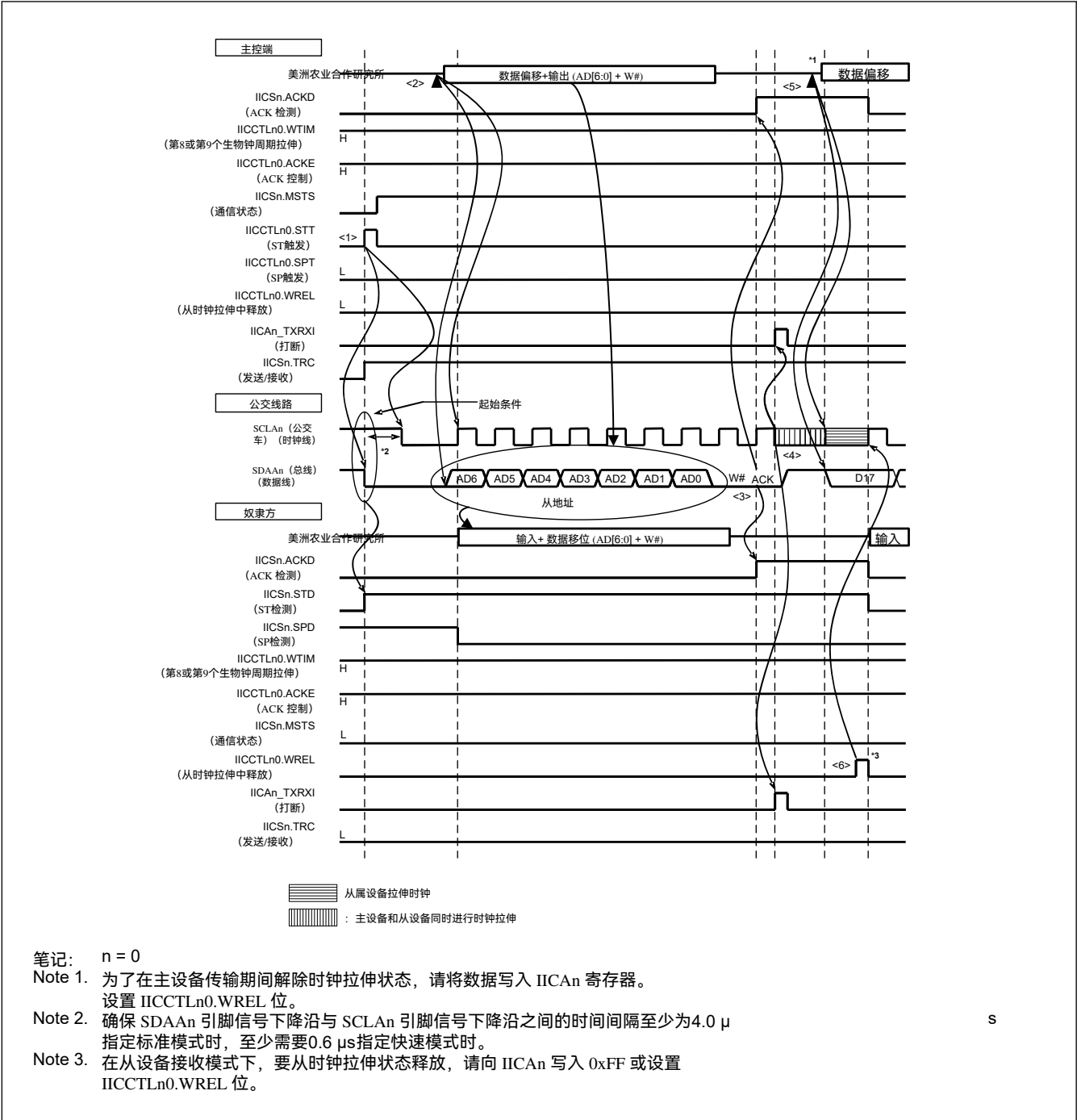


图 20.63 主设备与从设备通信示例（当主设备和从设备在第 9 个周期插入时钟拉伸时）(1/4)

图 20.63 中<1>至<6>的含义如下所述。

<1>主设备设置启动条件触发(IICCTL00.STT = 1) 并且一旦总线数据线变为低电平 (SDAA0), 就会生成启动条件(SCLAn = 1 并且 SDAAn 从 1 变为 0)。

当随后检测到启动条件时, 主设备进入主模式通信状态(IICSn.MSTS = 1), 保持时间结束后, 总线时钟线变为低电平(SCLAn = 0), 主设备即可开始通信。

<2> The master device writes the address + W (transmission) to the IICA shift register 0 (IICA0) and transmits the slave address.

<3> In the slave device if the address received matches the address (SVA0 value) of a slave device, that slave device sends an ACK by hardware to the master device. The ACK is detected by the master device (IICS0.ACKD = 1) at the rising edge of the 9th clock.

<4> The master device issues an interrupt (IICA0_TXRXI: end of address transmission) at the falling edge of the 9th clock. The slave device with the address matching the transmitted slave address sets the clock stretch state (SCLA0 = 0) and issues an interrupt (IICA0_TXRXI: address match).

<5> The master device writes the data to transmit to the IICA0 register and releases the clock stretch state set by the master device.

<6> If the slave device releases the clock stretch state (IICCTL00.WREL = 1), the master device starts transferring data to the slave device.

If the transmitted address does not match the address of the slave device, the slave device does not return an ACK to the master device (NACK: SDAA0 = 1). The slave device also does not issue the IICA0_TXRXI interrupt (address match) and does not set the clock stretch state.

The master device, however, issues the IICA0_TXRXI interrupt (end of address transmission) regardless of whether it receives an ACK or NACK.

Note: <1> to <15> in [\(1\) Example of master device to slave device communications \(when the master device and the slave device insert clock stretching on the 9th cycle\)](#) represent the entire procedure for communicating data using the I²C bus.

[Figure 20.63](#) shows the processing from <1> to <6>,
[Figure 20.64](#) shows the processing from <3> to <10>, and
[Figure 20.65](#) shows the processing from <7> to <15>.

<2>主设备将地址+ W （传输）写入 IICA 移位寄存器 0 (IICA0) 并发送从设备地址。

<3>在从设备中，如果接收到的地址与某个从设备的地址（SVA0 值）匹配，则该从设备通过硬件向主设备发送 A CK。主设备在第九个时钟周期的上升沿检测到该 ACK。（IICS0.ACKD = 1）

<4>主设备在第 9 个时钟的下降沿发出中断（IICA0_TXRXI：地址传输结束）。地址与发送的从设备地址匹配的从设备设置时钟拉伸状态(SCLA0 = 0) 并发出中断（IICA0_TXRXI：地址匹配）。

<5>主设备将要传输的数据写入 IICA0 寄存器，并释放主设备设置的时钟拉伸状态。

<6>如果从设备释放时钟拉伸状态（IICCTL00.WREL= 1），则主设备开始向从设备传输数据。

如果发送的地址与从设备的地址不匹配，则从设备不会向主设备返回 ACK（NACK：SDAA0 = 1）。从设备也不会发出 IICA0_TXRXI 中断（地址匹配），也不会设置时钟拉伸状态。

但是，无论收到 ACK 还是 NACK，主设备都会发出 IICA0_TXRXI 中断（地址传输结束）。

注意: (1) 中<1>到<15>的主设备到从设备通信示例（当主设备和从设备在第 9 个周期插入时钟拉伸时）代表了使用I²C 总线进行数据通信的整个过程。

图 20.63 显示了从<1>到<6>的处理过程，
图 20.64 显示了从<3>到<10>的处理过程，
图 20.65 显示了从<7>到<15>的处理过程。

2. Address → data → data

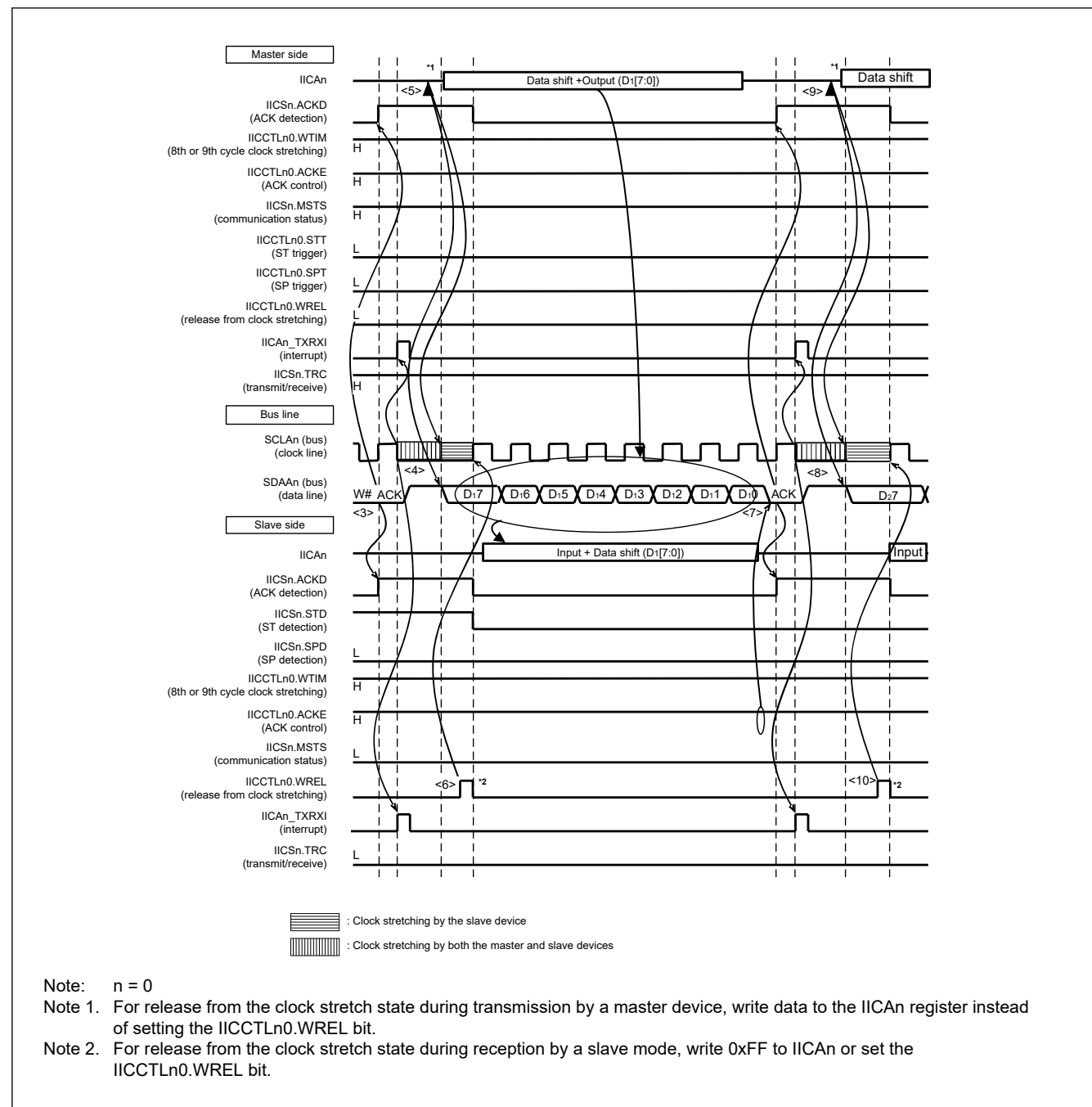


Figure 20.64 Example of master device to slave device communications (when the master device and the slave device insert clock stretching on the 9th cycle) (2/4)

The meanings of <3> to <10> in Figure 20.64 are explained below.

<3> In the slave device if the address received matches the address (SVA0 value) of a slave device, that slave device sends an ACK by hardware to the master device. The ACK is detected by the master device (IICSn.ACKD = 1) at the rising edge of the 9th clock.

<4> The master device issues an interrupt (IICAn_TXRXI: end of address transmission) at the falling edge of the 9th clock. The slave device with the address matching the transmitted slave address sets the clock stretch state (SCLAn = 0) and issues an interrupt (IICAn_TXRXI: address match).

<5> The master device writes the data to transmit to the IICA shift register 0 (IICAn) and releases the clock stretch state set by the master device.

2. 地址→数据→数据

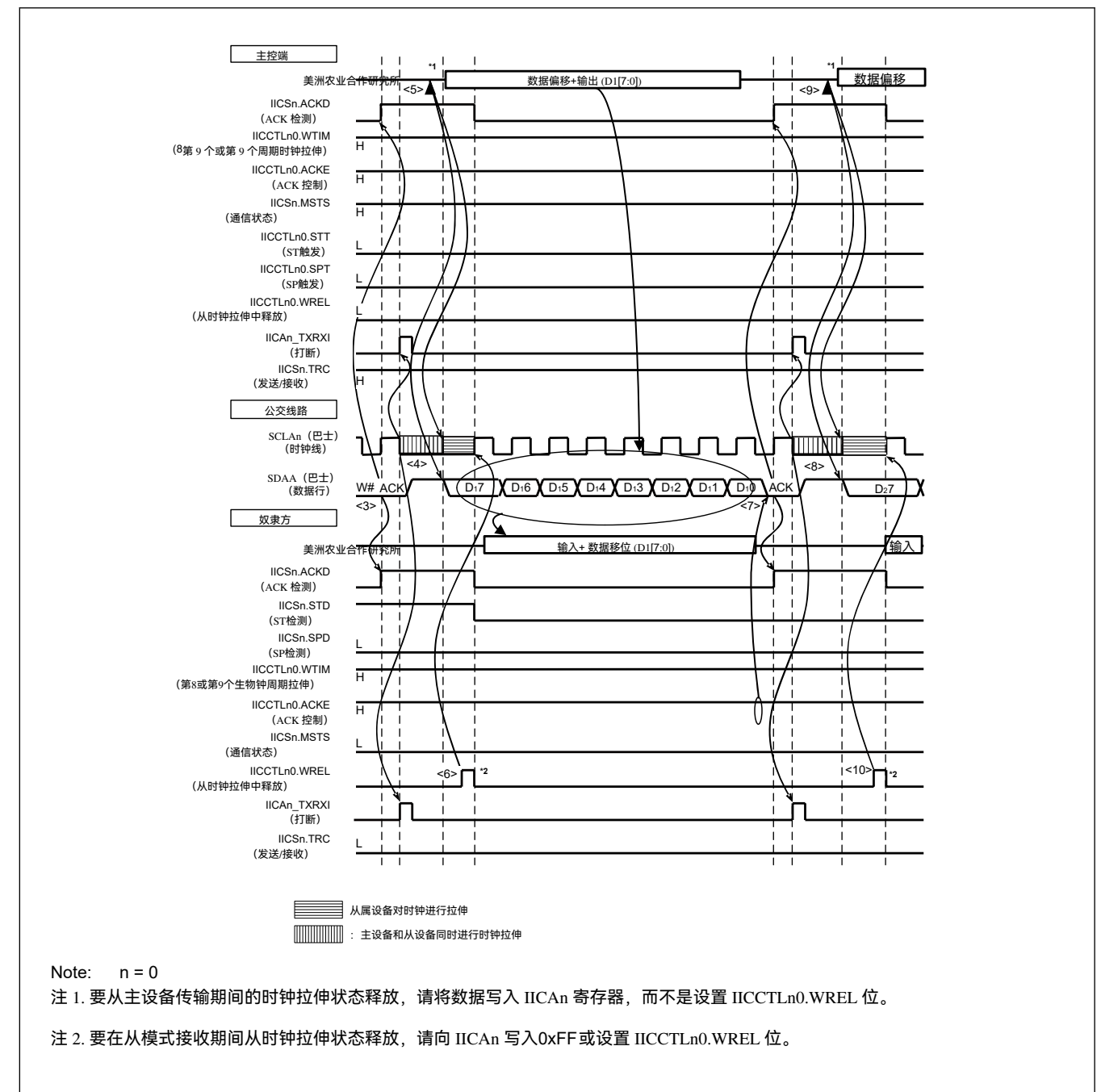


图 20.64 主设备与从设备通信示例 (当主设备和从设备在第 9 个周期插入时钟时) (2/4)

图 20.64 中 <3> 至 <10> 的含义如下所述。

<3> 在从设备中, 如果接收到的地址与某个从设备的地址 (SVA0 值) 匹配, 则该从设备通过硬件向主设备发送 ACK。主设备在第九个时钟周期的上升沿检测到该 ACK。(IICSn.ACKD = 1)

<4> 主设备在第 9 个时钟的下降沿发出中断 (IICAn_TXRXI: 地址传输结束)。地址与发送的从设备地址匹配的从设备设置时钟拉伸状态 (SCLAn = 0) 并发出中断 (IICAn_TXRXI: 地址匹配)。

<5> 主设备将要传输的数据写入 IICA 移位寄存器 0 (IICAn), 并释放主设备设置的时钟拉伸状态。

<6> If the slave device releases the clock stretch state (IICCTL00.WREL = 1), the master device starts transferring data to the slave device.

<7> After data transfer is completed, because of IICCTL00.ACKE = 1, the slave device sends an ACK by hardware to the master device. The ACK is detected by the master device (IICS0.ACKD = 1) at the rising edge of the 9th clock.

<8> The master device and slave device set the clock stretch state (SCLA0 = 0) at the falling edge of the 9th clock, and both the master device and slave device issue an interrupt (IICA0_TXRXI: end of transfer).

<9> The master device writes the data to transmit to the IICA0 register and releases the clock stretch state set by the master device.

<10>The slave device reads the received data and releases the clock stretch state (IICCTL00.WREL = 1). The master device then starts transferring data to the slave device.

If the transmitted address does not match the address of the slave device, the slave device does not return an ACK to the master device (NACK: SDAA0 = 1). The slave device also does not issue the IICA0_TXRXI interrupt (address match) and does not set the clock stretch state.

The master device, however, issues the IICA0_TXRXI interrupt (end of address transmission) regardless of whether it receives an ACK or NACK.

Note: <1> to <15> in [\(1\) Example of master device to slave device communications \(when the master device and the slave device insert clock stretching on the 9th cycle\)](#) represent the entire procedure for communicating data using the I²C bus.

[Figure 20.63](#) shows the processing from <1> to <6>,
[Figure 20.64](#) shows the processing from <3> to <10>, and
[Figure 20.65](#) shows the processing from <7> to <15>.

<6>如果从设备释放时钟拉伸状态（IICCTL00.WREL= 1），则主设备开始向从设备传输数据。

<7>数据传输完成后，由于 IICCTL00.ACKE= 1，从设备通过硬件向主设备发送 ACK。主设备在第九个时钟周期的上升沿检测到该 ACK。（IICS0.ACKD = 1）

<8>主设备和从设备在第 9 个时钟的下降沿设置时钟拉伸状态(SCLA0 = 0)，并且主设备和从设备都发出中断（IICA0_TXRXI：传输结束）。

<9>主设备将要传输的数据写入 IICA0 寄存器，并释放主设备设置的时钟拉伸状态。

从设备读取接收到的数据并释放时钟拉伸状态（IICCTL00.WREL= 1）。然后，主设备开始向从设备传输数据。

如果发送的地址与从设备的地址不匹配，则从设备不会向主设备返回 ACK（NACK: SDAA0 = 1）。从设备也不会发出 IICA0_TXRXI 中断（地址匹配），也不会设置时钟拉伸状态。

但是，无论收到 ACK 还是 NACK，主设备都会发出 IICA0_TXRXI 中断（地址传输结束）。

注意: (1) 中<1>到<15>的主设备到从设备通信示例（当主设备和从设备在第 9 个周期插入时钟拉伸时）代表了使用 I²C 总线进行数据通信的整个过程。

图 20.63 显示了从<1>到<6>的处理过程，
图 20.64 显示了从<3>到<10>的处理过程，
图 20.65 显示了从<7>到<15>的处理过程。

3. Data → data → stop condition

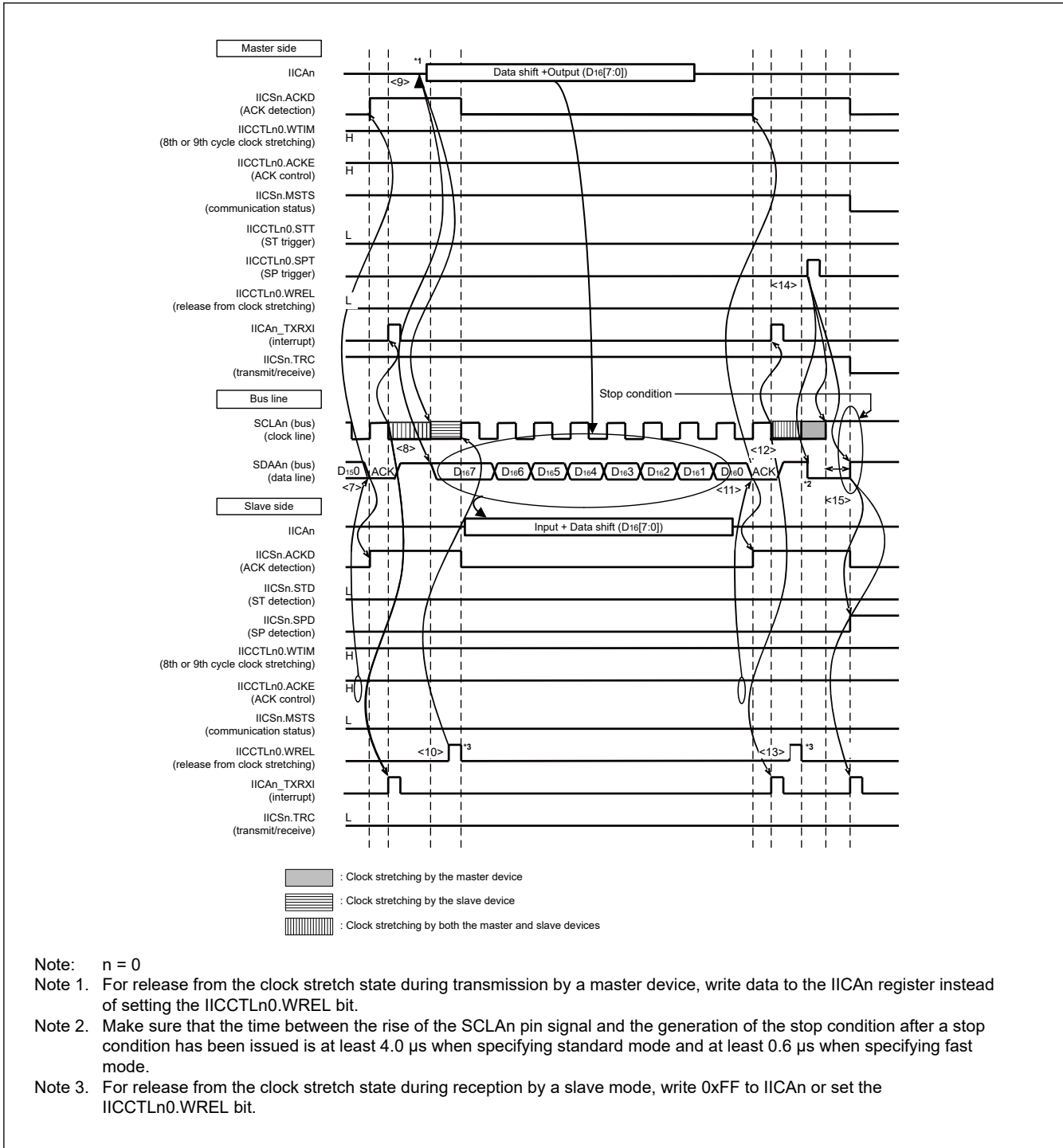


Figure 20.65 Example of master device to slave device communications (when the master device and the slave device insert clock stretching on the 9th cycle) (3/4)

The meanings of <7> to <15> in Figure 20.65 are explained below.

<7> After data transfer is completed, because of IICCTL00.ACKE = 1, the slave device sends an ACK by hardware to the master device. The ACK is detected by the master device (IICS0.ACKD = 1) at the rising edge of the 9th clock.

<8> The master device and slave device set the clock stretch state (SCLA0 = 0) at the falling edge of the 9th clock, and both the master device and slave device issue an interrupt (IICA0_TXRXI: end of transfer).

<9> The master device writes the data to transmit to the IICA shift register 0 (IICA0) and releases the clock stretch state set by the master device.

3. 数据→数据→停止条件

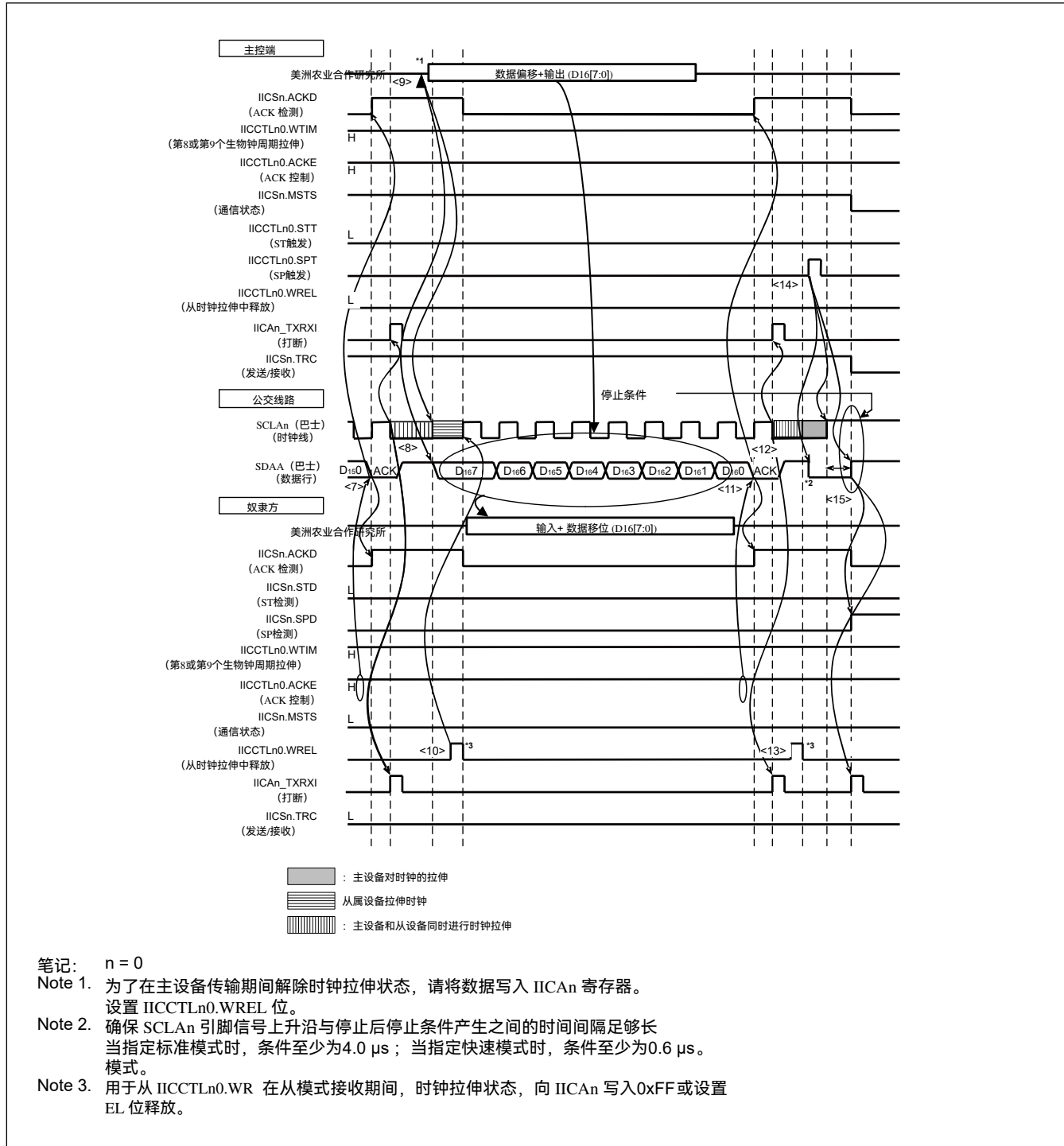


图 20.65 主设备与从设备通信示例 (当主设备和从设备在第 9 个周期插入时钟拉伸时) (3/4)

下面解释了图 20.65 中<7>到<15>的含义。

<7>数据传输完成后, 由于 IICCTL00.ACKE= 1, 从设备通过硬件向主设备发送 ACK。主设备在第九个时钟周期的上升沿检测到该 ACK。(IICS0.ACKD = 1)

<8>主设备和从设备在第 9 个时钟的下降沿设置时钟拉伸状态(SCLA0 = 0), 并且主设备和从设备都发出中断 (IICA0_TXRXI: 传输结束)。

<9>主设备将要传输的数据写入 IICA 移位寄存器 0 (IICA0), 并释放主设备设置的时钟拉伸状态。

<10>The slave device reads the received data and releases the clock stretch state (IICCTL00.WREL = 1). The master device then starts transferring data to the slave device.

<11>When data transfer is complete, the slave device (IICCTL00.ACKE = 1) sends an ACK by hardware to the master device.

The ACK is detected by the master device (IICS0.ACKD = 1) at the rising edge of the 9th clock.

<12>The master device and slave device set the clock stretch state (SCLA0 = 0) at the falling edge of the 9th clock, and both the master device and slave device issue an interrupt (IICA0_TXRXI: end of transfer).

<13>The slave device reads the received data and releases the clock stretch state (IICCTL00.WREL = 1).

<14>By the master device setting a stop condition trigger (IICCTL00.SPT = 1), the bus data line is cleared (SDAA0 = 0) and the bus clock line is set (SCLA0 = 1). After the stop condition setup time has elapsed, by setting the bus data line (SDAA0 = 1), the stop condition is then generated (SCLA0 = 1 and SDAA0 changes from 0 to 1).

<15>When a stop condition is generated, the slave device detects the stop condition and issues an interrupt (IICA0_TXRXI: stop condition).

Note: <1> to <15> in [\(1\) Example of master device to slave device communications \(when the master device and the slave device insert clock stretching on the 9th cycle\)](#) represent the entire procedure for communicating data using the I²C bus.

[Figure 20.63](#) shows the processing from <1> to <6>,
[Figure 20.64](#) shows the processing from <3> to <10>, and
[Figure 20.65](#) shows the processing from <7> to <15>.

从设备读取接收到的数据并释放时钟拉伸状态（IICCTL00.WREL= 1）。然后，主设备开始向从设备传输数据。

<11>当数据传输完成后，从设备(IICCTL00.ACKE = 1) 通过硬件向主设备发送 ACK。

主设备(IICS0.ACKD = 1) 在第 9 个时钟的上升沿检测到 ACK。

<12>主设备和从设备在第 9 个时钟的下降沿设置时钟拉伸状态(SCLA0 = 0)，并且主设备和从设备都发出中断（IICA0_TXRXI: 传输结束）。

<13>从设备读取接收到的数据并释放时钟拉伸状态(IICCTL00.WREL = 1)。

当主设备设置停止条件触发信号时，总线数据线被清零，总线时钟线被置位。停止条件建立时间过后，通过置位总线数据线，停止条件被生成，SDAA0 的值从 0 变为 1。

<15>当生成停止条件时，从设备检测到停止条件并发出中断（IICA0_TXRXI: 停止条件）。

注意: (1) 中<1>到<15>的主设备到从设备通信示例（当主设备和从设备在第 9 个周期插入时钟拉伸时）代表了使用I²C 总线进行数据通信的整个过程。

图 20.63 显示了从<1>到<6>的处理过程，
图 20.64 显示了从<3>到<10>的处理过程，
图 20.65 显示了从<7>到<15>的处理过程。

4. Data → restart condition → address

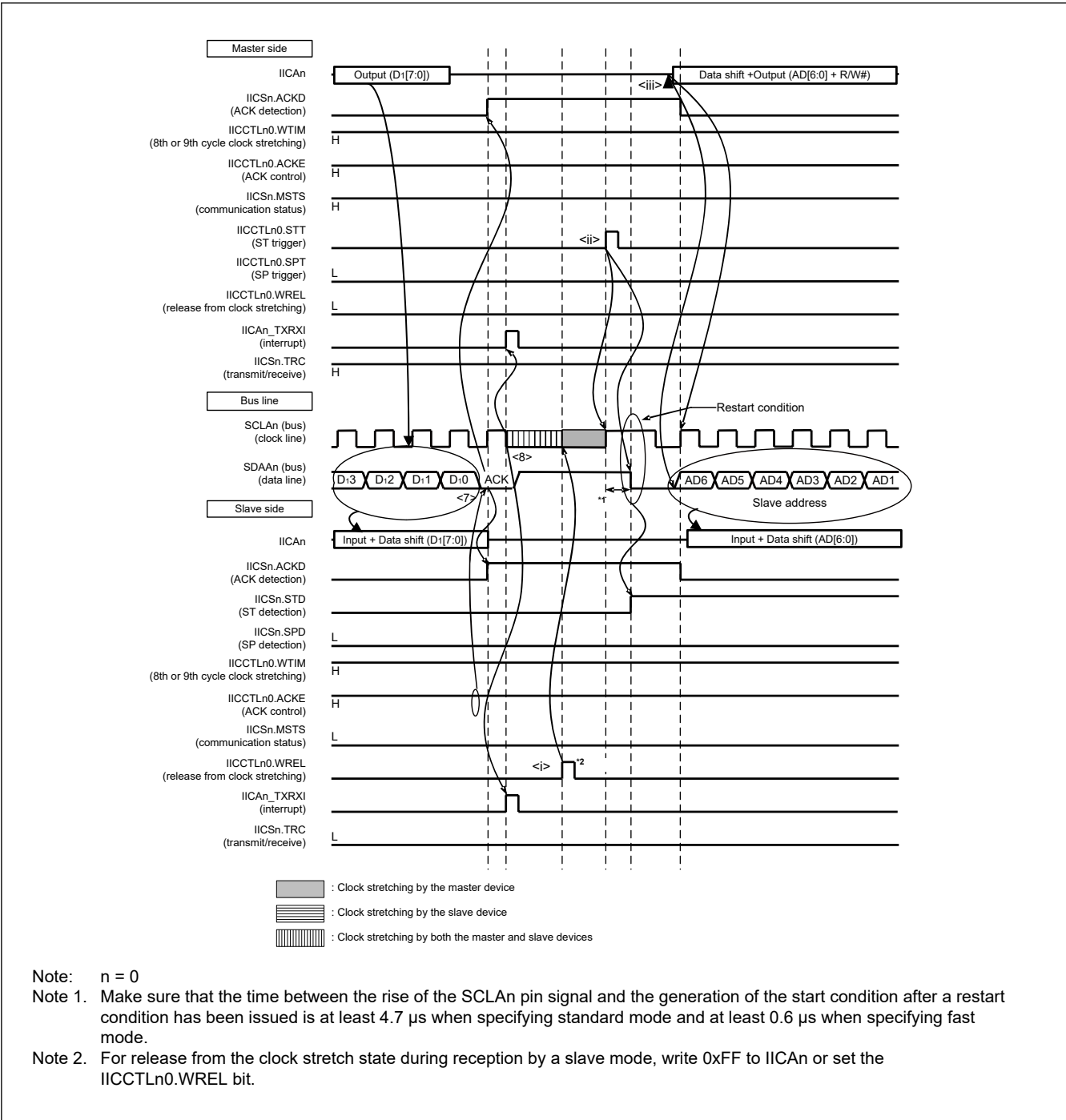


Figure 20.66 Example of master device to slave device communications (when the master device and the slave device insert clock stretching on the 9th cycle) (4/4)

The following describes the operations in Figure 20.66. After the operations in steps <7> and <8>, the operations in steps <i> to <iii> are performed. These steps return the processing to step <3>, the data transmission step.

<7> After data transfer is completed, because of IICCTL00.ACKE = 1, the slave device sends an ACK by hardware to the master device. The ACK is detected by the master device (IICS0.ACKD = 1) at the rising edge of the 9th clock.

<8> The master device and slave device set the clock stretch state (SCLA0 = 0) at the falling edge of the 9th clock, and both the master device and slave device issue an interrupt (IICA0_TXRXI: end of transfer).

<i> The slave device reads the received data and releases the clock stretch state (IICCTL00.WREL = 1).

4. 数据→重启条件→地址

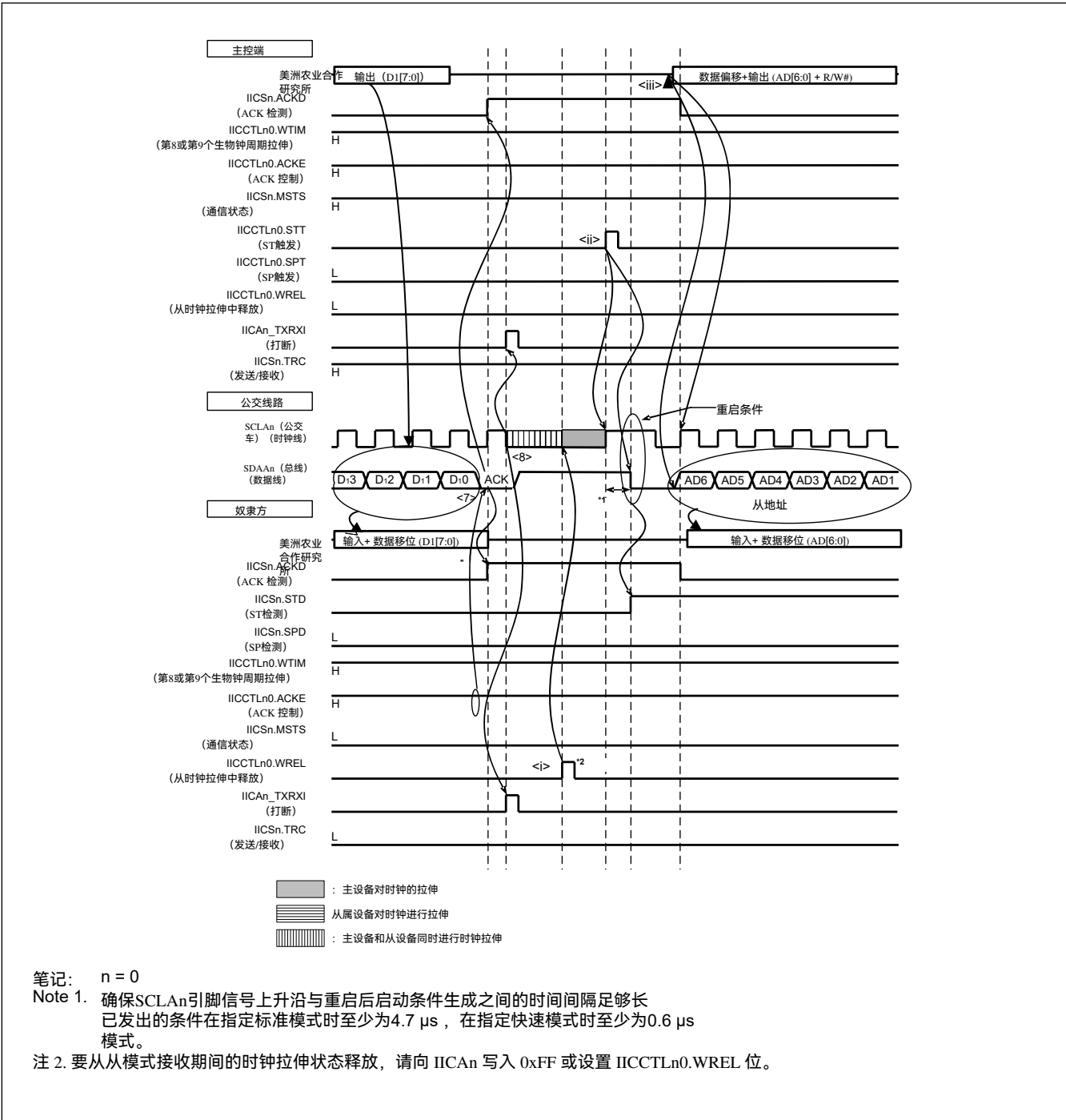


图 20.66 主设备与从设备通信示例（当主设备和从设备在第 9 个周期插入时钟拉伸时）(4/4)

以下描述了图 20.66 中的操作。在执行步骤<7>和<8>，中的操作之后，执行步骤<i>至<iii>中的操作。这些步骤将处理过程返回到步骤<3>，即数据传输步骤。

<7>数据传输完成后，由于 IICCTL00.ACKE = 1，从设备通过硬件向主设备发送 ACK。主设备(IICS0.ACKD = 1) 在第 9 个时钟周期的上升沿检测到该 ACK。

<8>主设备和从设备在第 9 个时钟的下降沿设置时钟拉伸状态(SCLA0 = 0)，并且主设备和从设备都发出中断（IICA0_TXRXI：传输结束）。

<i>从设备读取接收到的数据并释放时钟拉伸状态(IICCTL00.WREL = 1)。

<ii> The start condition trigger is set again by the master device (IICCTL00.STT = 1) and a start condition (SCLA0 = 1 and SDAA0 changes from 1 to 0) is generated once the bus clock line goes high (SCLA0 = 1) and the bus data line goes low (SDAA0 = 0) after the restart condition setup time has elapsed. When the start condition is subsequently detected, the master device is ready to communicate once the bus clock line goes low (SCLA0 = 0) after the hold time has elapsed.

<iii> The master device writing the address + R/W (transmission) to the IICA shift register 0 (IICA0) enables the slave address to be transmitted.

<ii>主设备再次设置启动条件触发信号(IICCTL00.STT = 1)，并在总线时钟线变为高电平(SCLA0 = 1且总线数据线变为低电平(SDAA0 = 0) 后，在重启条件建立时间结束后，生成启动条件（SCLA0= 1 和 SDAA0 由 1 变为 0）。当后续检测到启动条件时，在保持时间结束后总线时钟线变为低电平(SCLA0 = 0)，主设备即可开始通信。

<iii>主设备将地址+ R/W （传输）写入 IICA 移位寄存器 0 (IICA0) 以启用从设备地址的传输。

(2) Example of slave device to master device communications (8th cycle clock stretching is selected for the master device and 9th cycle clock stretching is selected for the slave device)

1. Start condition → address → data

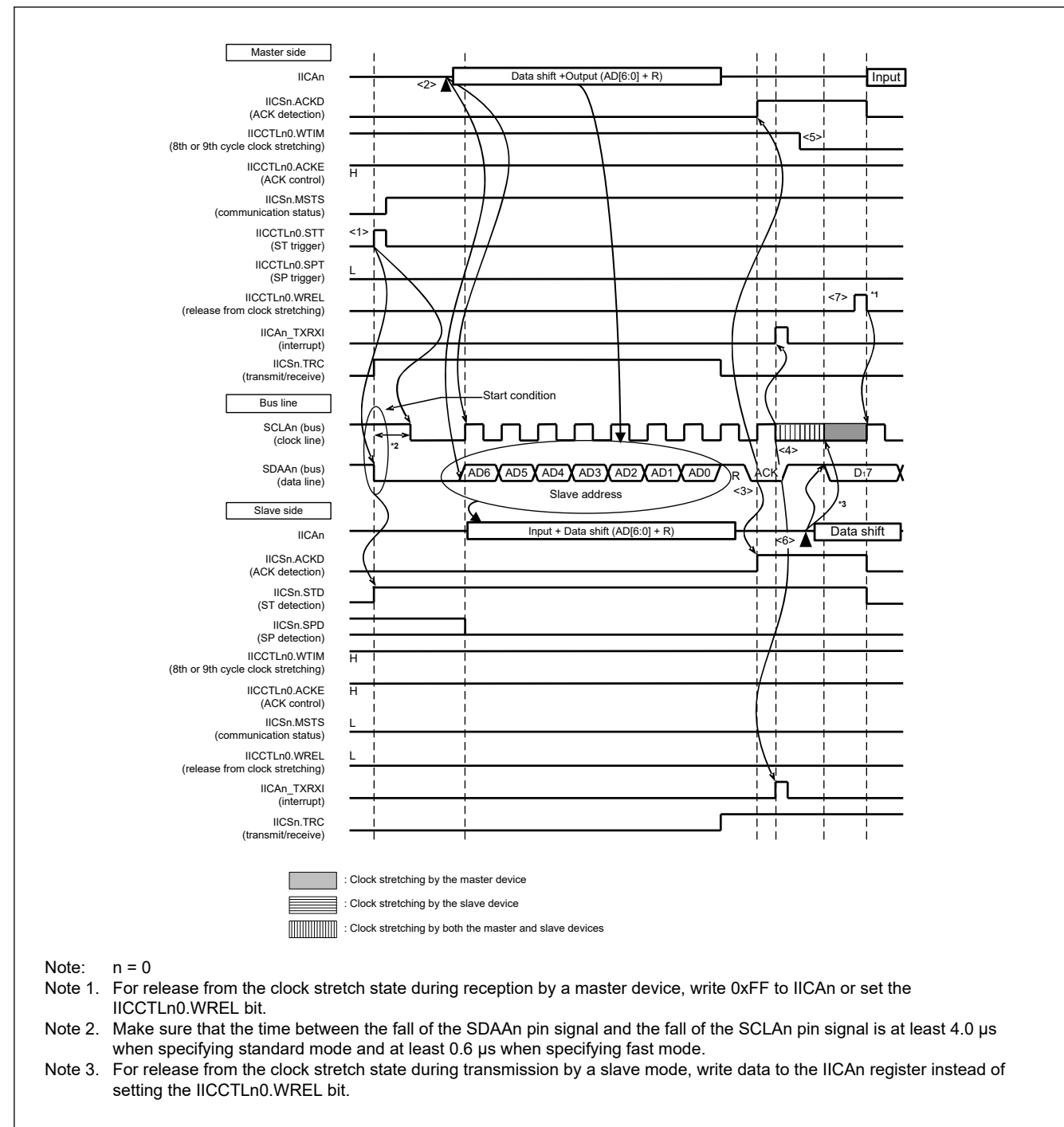


Figure 20.67 Example of slave device to master device communications (8th cycle clock stretching is selected for the master device and 9th cycle clock stretching is selected for the slave device) (1/3)

The meanings of <1> to <7> in Figure 20.67 are explained below.

<1> The start condition trigger is set by the master device (IICCTL00.STT = 1) and a start condition (SCLA0 = 1 and SDAA0 changes from 1 to 0) is generated once the bus data line goes low (SDAA0). When the start condition is subsequently detected, the master device enters the master mode communication status (IICS0.MSTS = 1). The master device is ready to communicate once the bus clock line goes low (SCLA0 = 0) after the hold time has elapsed.

(2) 从设备与主设备通信示例（主设备选择第 8 个时钟周期拉伸，从设备选择第 9 个时钟周期拉伸）

1. 起始条件→地址→数据

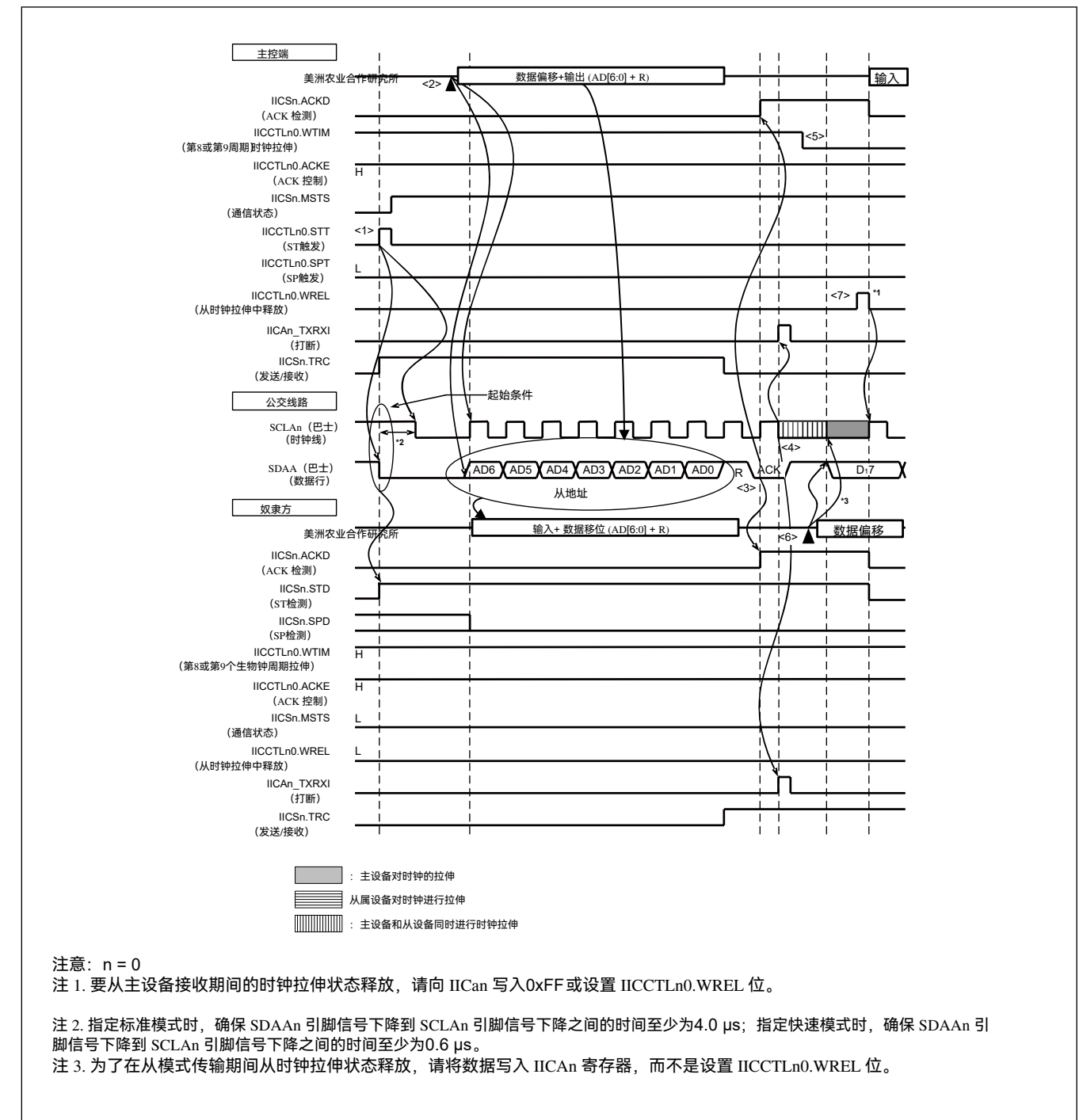


Figure 20.67 从设备与主设备通信示例（主设备选择第 8 个时钟周期拉伸，从设备选择第 9 个时钟周期拉伸）（1/3）

图 20.67 中<1>至<7>的含义如下所述。

<1>主设备设置启动条件触发信号(IICCTL00.STT = 1)，当总线数据线变为低电平(SDAA0)时，会生成启动条件(SCLA0 = 1 (SDAA0 从 1 变为 0))。当检测到启动条件后，主设备进入主模式通信状态(IICS0.MSTS = 1)。保持时间结束后，当总线时钟线变为低电平(SCLA0 = 0)时，主设备即可开始通信。

<2> The master device writes the address + R (reception) to the IICA shift register 0 (IICA0) and transmits the slave address.

<3> In the slave device if the address received matches the address (SVA0 value) of a slave device, that slave device sends an ACK by hardware to the master device. The ACK is detected by the master device (IICS0.ACKD = 1) at the rising edge of the 9th clock.

<4> The master device issues an interrupt (IICA0_TXRXI: end of address transmission) at the falling edge of the 9th clock. The slave device with the address matching the transmitted slave address sets the clock stretch state (SCLA0 = 0) and issues an interrupt (IICA0_TXRXI: address match).

<5> The timing at which the master device sets the clock stretch state changes to the 8th clock (WTIM = 0).

<6> The slave device writes the data to transmit to the IICA0 register and releases the clock stretch state set by the slave device.

<7> The master device releases the clock stretch state (IICCTL00.WREL = 1) and starts transferring data from the slave device to the master device.

If the transmitted address does not match the address of the slave device, the slave device does not return an ACK to the master device (NACK: SDAA0 = 1). The slave device also does not issue the IICA0_TXRXI interrupt (address match) and does not set the clock stretch state.

The master device, however, issues the IICA0_TXRXI interrupt (end of address transmission) regardless of whether it receives an ACK or NACK.

Note: <1> to <19> in [\(2\) Example of slave device to master device communications \(8th cycle clock stretching is selected for the master device and 9th cycle clock stretching is selected for the slave device\)](#) represent the entire procedure for communicating data using the I²C bus.

[Figure 20.67](#) shows the processing from <1> to <7>,

[Figure 20.68](#) shows the processing from <3> to <12>, and

[Figure 20.69](#) shows the processing from <8> to <19>.

<2>主设备将地址+ R （接收）写入 IICA 移位寄存器 0 (IICA0) 并发送从设备地址。

<3>在从设备中，如果接收到的地址与某个从设备的地址（SVA0 值）匹配，则该从设备通过硬件向主设备发送 ACK。主设备在第九个时钟周期的上升沿检测到该 ACK。（IICS0.ACKD = 1）

<4>主设备在第 9 个时钟的下降沿发出中断（IICA0_TXRXI：地址传输结束）。地址与发送的从设备地址匹配的从设备设置时钟拉伸状态(SCLA0 = 0) 并发出中断（IICA0_TXRXI：地址匹配）。

<5>主设备设置时钟拉伸状态的时机更改为第 8 个时钟(WTIM = 0)。

<6>从设备将要传输的数据写入 IICA0 寄存器，并释放从设备设置的时钟拉伸状态。

<7>主设备释放时钟拉伸状态 (IICCTL00.WREL= 1) 并开始从从设备向主设备传输数据。

如果发送的地址与从设备的地址不匹配，则从设备不会向主设备返回 ACK（NACK：SDAA0 = 1）。从设备也不会发出 IICA0_TXRXI 中断（地址匹配），也不会设置时钟拉伸状态。

但是，无论收到 ACK 还是 NACK，主设备都会发出 IICA0_TXRXI 中断（地址传输结束）。

注意: [\(2\) 从设备到主设备通信示例（主设备选择第 8 个周期时钟拉伸，从设备选择第 9 个周期时钟拉伸）](#) 中的<1>到<19>代表使用I²C总线进行数据通信的整个过程。

图 20.67 显示了从<1>到<7>的处理过程，

图 20.68 显示了从<3>到<12>的处理过程，

图 20.69 显示了从<8>到<19>的处理过程。

2. Address → data → data

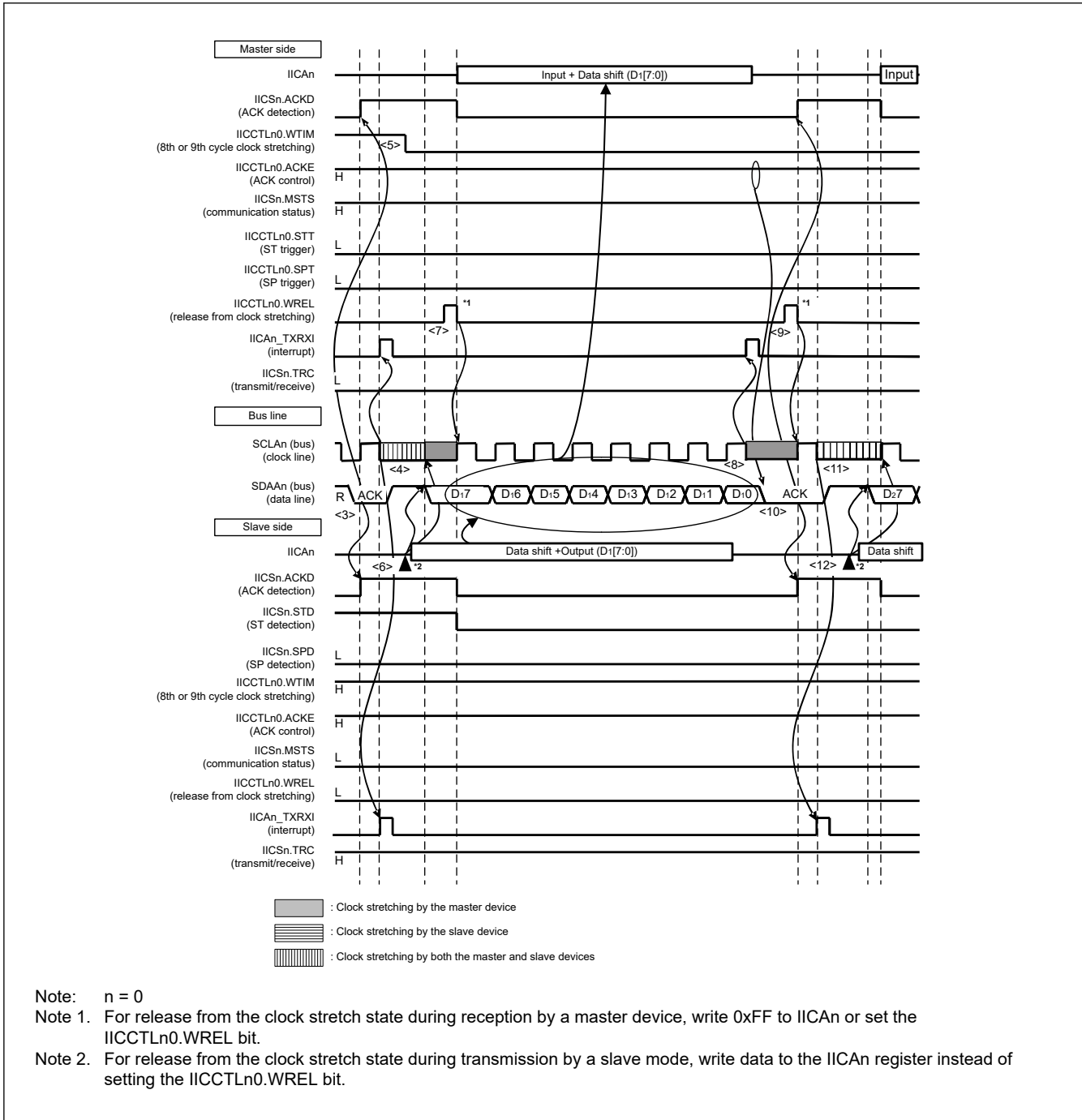


Figure 20.68 Example of slave device to master device communications (8th cycle clock stretching is selected for the master device and 9th cycle clock stretching is selected for the slave device) (2/3)

The meanings of <3> to <12> in [Figure 20.68](#) are explained below.

<3> In the slave device if the address received matches the address (SVA0 value) of a slave device, that slave device sends an ACK by hardware to the master device. The ACK is detected by the master device (IICS0.ACKD = 1) at the rising edge of the 9th clock.

<4> The master device issues an interrupt (IICA0_TXRXI: end of address transmission) at the falling edge of the 9th clock. The slave device with the address matching the transmitted slave address sets the clock stretch state (SCLA0 = 0) and issues an interrupt (IICA0_TXRXI: address match).

<5> The master device changes the timing of clock stretching to the 8th clock (IICCTL00.WTIM = 0).

2. 地址→数据→数据

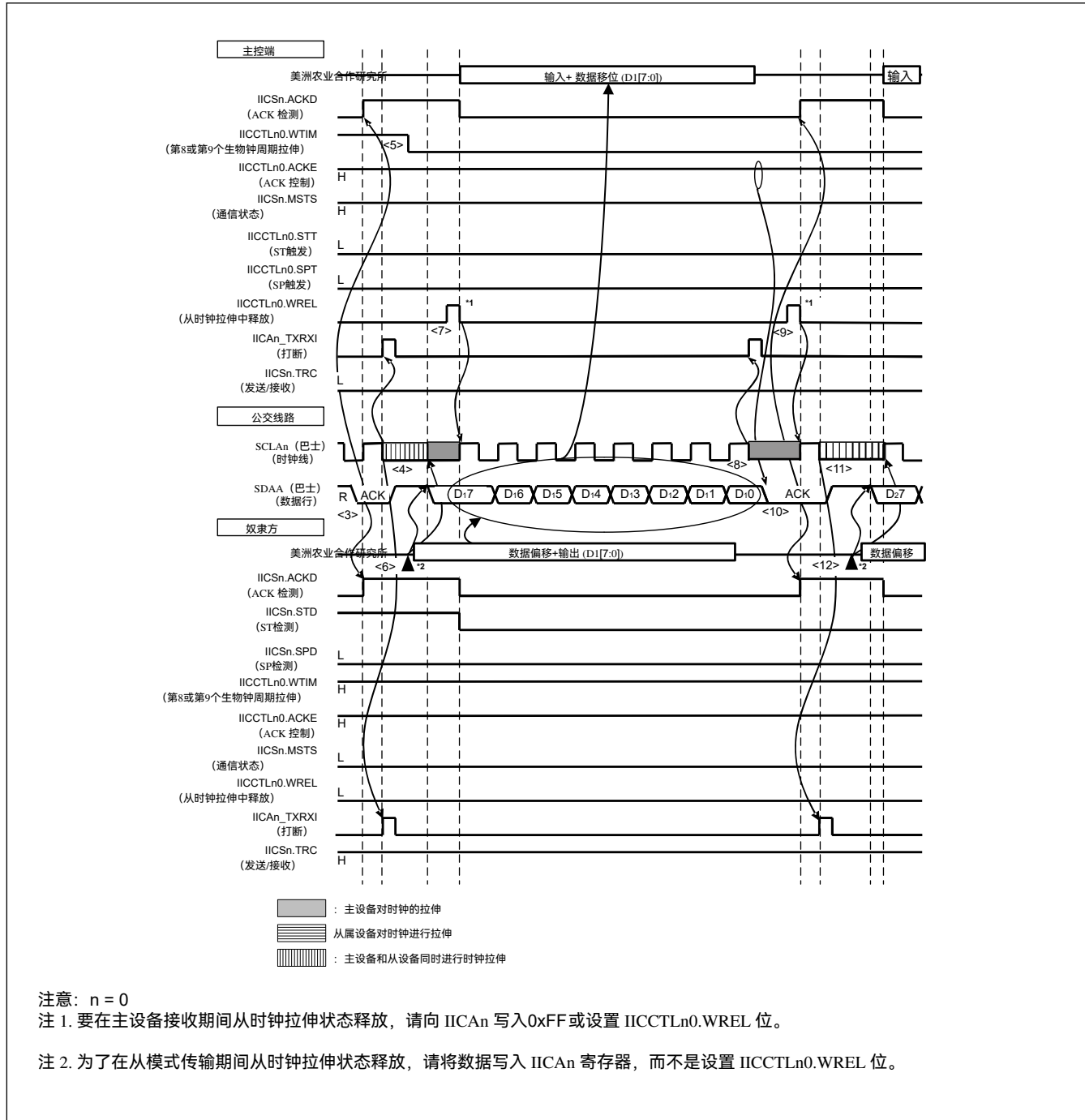


图 20.68 从设备到主设备的通信示例 (主设备选择第 8 个时钟周期拉伸, 从设备选择第 9 个时钟周期拉伸) (2/3)

图 20.68 中 <3> 至 <12> 的含义如下所述。

<3> 在从设备中, 如果接收到的地址与某个从设备的地址 (SVA0 值) 匹配, 则该从设备通过硬件向主设备发送 ACK。主设备在第九个时钟周期的上升沿检测到该 ACK。(IICS0.ACKD = 1)

<4> 主设备在第 9 个时钟的下降沿发出中断 (IICA0_TXRXI: 地址传输结束)。地址与发送的从设备地址匹配的从设备设置时钟拉伸状态 (SCLA0 = 0) 并发出中断 (IICA0_TXRXI: 地址匹配)。

<5> 主设备将时钟拉伸的时序更改为第 8 个时钟 (IICCTL00.WTIM = 0)。

<6> The slave device writes the data to transmit to the IICA shift register 0 (IICA0) and releases the clock stretch state set by the slave device.

<7> The master device releases the clock stretch state (IICCTL00.WREL = 1) and starts transferring data from the slave device to the master device.

<8> The master device sets the clock stretch state (SCLA0 = 0) at the falling edge of the 8th clock, and issues an interrupt (IICA0_TXRXI: end of transfer). Because of IICCTL00.ACKE = 1 in the master device, the master device then sends an ACK by hardware to the slave device.

<9> The master device reads the received data and releases the clock stretch state (IICCTL00.WREL = 1).

<10>The ACK is detected by the slave device (IICS0.ACKD = 1) at the rising edge of the 9th clock.

<11>The slave device sets the clock stretch state (SCLA0 = 0) at the falling edge of the 9th clock, and the slave device issue an interrupt (IICA0_TXRXI: end of transfer).

<12>By the slave device writing the data to transmit to the IICA0 register, the clock stretch state set by the slave device is released. The slave device then starts transferring data to the master device.

If the transmitted address does not match the address of the slave device, the slave device does not return an ACK to the master device (NACK: SDAA0 = 1). The slave device also does not issue the IICA0_TXRXI interrupt (address match) and does not set the clock stretch state.

The master device, however, issues the IICA0_TXRXI interrupt (end of address transmission) regardless of whether it receives an ACK or NACK.

Note: <1> to <19> in [\(2\) Example of slave device to master device communications \(8th cycle clock stretching is selected for the master device and 9th cycle clock stretching is selected for the slave device\)](#) represent the entire procedure for communicating data using the I²C bus.

Figure 20.67 shows the processing from <1> to <7>,
Figure 20.68 shows the processing from <3> to <12>, and
Figure 20.69 shows the processing from <8> to <19>.

<6>从设备将要传输的数据写入 IICA 移位寄存器 0 (IICA0)，并释放从设备设置的时钟拉伸状态。

<7>主设备释放时钟拉伸状态 (IICCTL00.WREL= 1) 并开始从从设备向主设备传输数据。

<8>主设备在第 8 个时钟周期的下降沿设置时钟拉伸状态(SCLA0 = 0)，并发出中断（IICA0_TXRXI：传输结束）。由于主设备中存在 IICCTL00.ACKE = 1，主设备随后通过硬件向从设备发送 ACK。

<9>主设备读取接收到的数据并释放时钟拉伸状态(IICCTL00.WREL = 1)。

<10>从设备(IICS0.ACKD = 1) 在第 9 个时钟的上升沿检测到 ACK。

<11>从设备在第 9 个时钟的下降沿设置时钟拉伸状态(SCLA0 = 0)，并且从设备发出中断（IICA0_TXRXI：传输结束）。

当从设备将要发送的数据写入 IICA0 寄存器时，从设备设置的时钟拉伸状态被释放。然后，从设备开始向主设备传输数据。

如果发送的地址与从设备的地址不匹配，则从设备不会向主设备返回 ACK（NACK：SDAA0 = 1）。从设备也不会发出 IICA0_TXRXI 中断（地址匹配），也不会设置时钟拉伸状态。

但是，无论收到 ACK 还是 NACK，主设备都会发出 IICA0_TXRXI 中断（地址传输结束）。

注意: (2) 从设备到主设备通信示例（主设备选择第 8 个周期时钟拉伸，从设备选择第 9 个周期时钟拉伸）中的<1>到<19>代表使用I²C总线进行数据通信的整个过程。

图 20.67 显示了从<1>到<7>的处理过程，
图 20.68 显示了从<3>到<12>的处理过程，
图 20.69 显示了从<8>到<19>的处理过程。

3. Data → data → stop condition

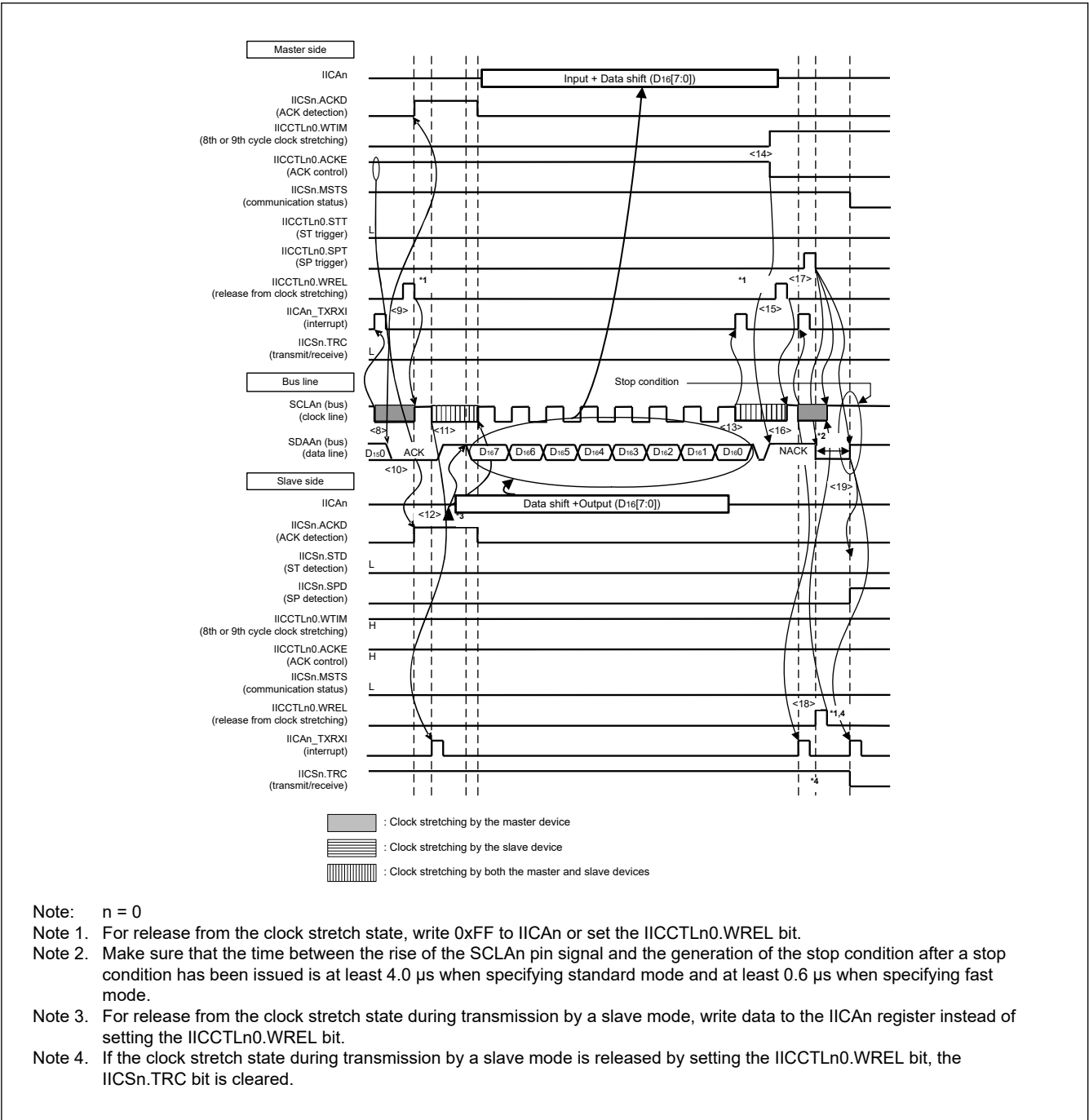


Figure 20.69 Example of slave device to master device communications (8th cycle clock stretching is selected for the master device and 9th cycle clock stretching is selected for the slave device) (3/3)

The meanings of <8> to <19> in Figure 20.69 are explained below.

<8> The master device sets the clock stretch state (SCLA0 = 0) at the falling edge of the 8th clock, and issues an interrupt (IICA0_TXRXI: end of transfer). Because of IICCTL00.ACKE = 0 in the master device, the master device then sends an ACK by hardware to the slave device.

<9> The master device reads the received data and releases the clock stretch state (IICCTL00.WREL = 1).

<10>The ACK is detected by the slave device (IICS0.ACKD = 1) at the rising edge of the 9th clock.

<11>The slave device sets the clock stretch state (SCLA0 = 0) at the falling edge of the 9th clock, and the slave device issue an interrupt (IICA0_TXRXI: end of transfer).

3. 数据→数据→停止条件

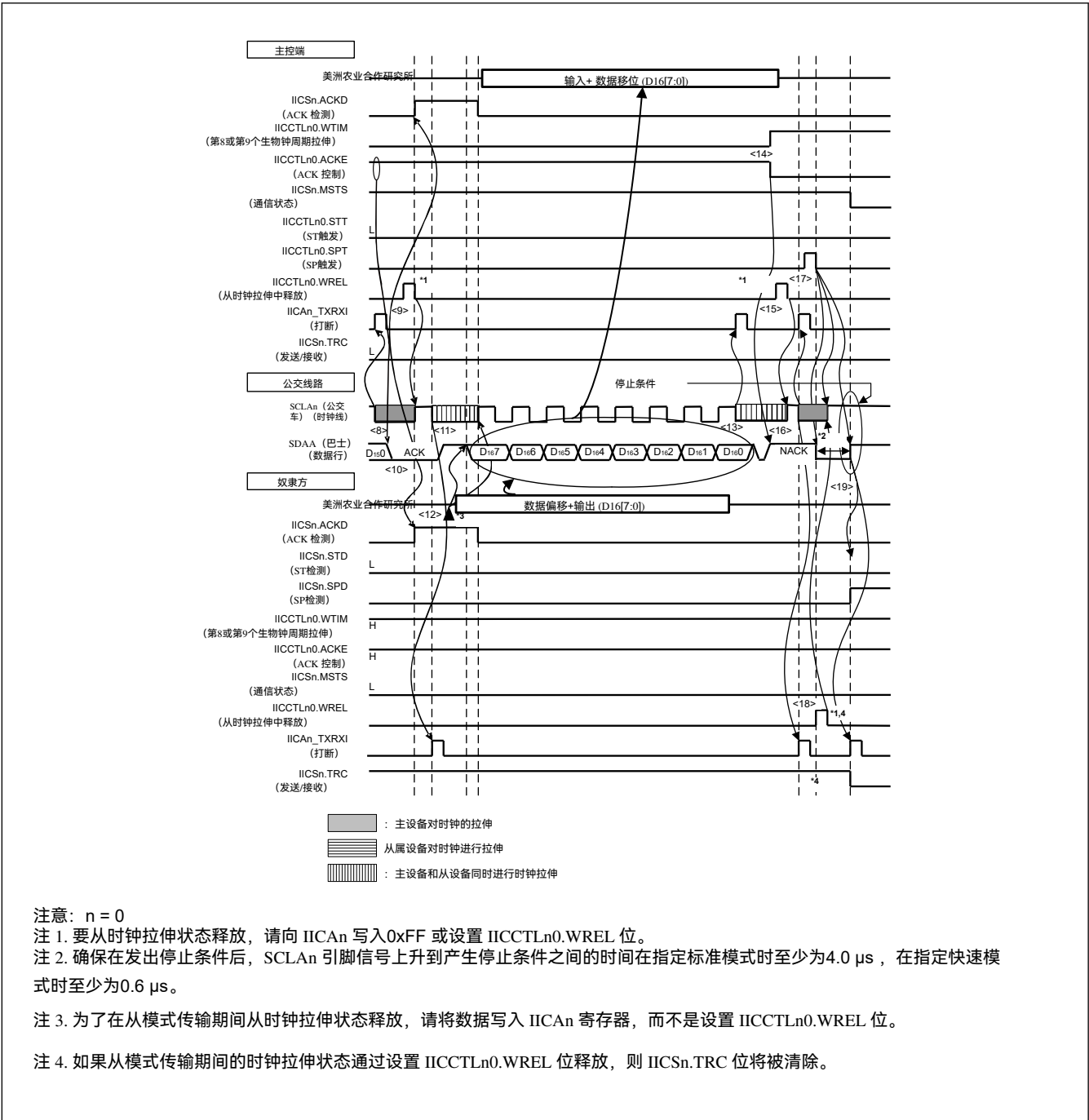


Figure 20.69 从设备与主设备通信示例 (主设备选择第 8 个时钟周期拉伸, 从设备选择第 9 个时钟周期拉伸) (3/3)

图 20.69 中<8>至<19>的含义解释如下。

<8>主设备在第 8 个时钟周期的下降沿设置时钟拉伸状态(SCLA0 = 0), 并发出中断 (IICA0_TXRXI: 传输结束)。由于主设备中存在 IICCTL00.ACKE = 0, 主设备随后通过硬件向从设备发送 ACK。

<9>主设备读取接收到的数据并释放时钟拉伸状态(IICCTL00.WREL = 1)。

<10>从设备(IICS0.ACKD = 1)在第 9 个时钟的上升沿检测到 ACK。

<11>从设备在第 9 个时钟的下降沿设置时钟拉伸状态(SCLA0 = 0), 并且从设备发出中断 (IICA0_TXRXI: 传输结束)。

<12>By the slave device writing the data to transmit to the IICA0 register, the clock stretch state set by the slave device is released. The slave device then starts transferring data to the master device.

<13>The master device issues an interrupt (IICA0_TXRXI: end of transfer) at the falling edge of the 8th clock, and sets the clock stretch state (SCLA0 = 0). Because ACK control (IICCTL00.ACKE = 1) is performed, the bus data line is at the low level (SDAA0 = 0) at this stage.

<14>The master device sets NACK as the response (IICCTL00.ACKE = 0) and changes the timing at which it sets the clock stretch state to the 9th clock (IICCTL00.WTIM = 1).

<15>If the master device releases the clock stretch state (IICCTL00.WREL = 1), the slave device detects the NACK (ACK = 0) at the rising edge of the 9th clock.

<16>The master device and slave device set the clock stretch state (SCLA0 = 0) at the falling edge of the 9th clock, and both the master device and slave device issue an interrupt (IICA0_TXRXI: end of transfer).

<17>When the master device issues a stop condition (IICCTL00.SPT = 1), the bus data line is cleared (SDAA0 = 0) and the master device releases the clock stretch state. The master device then waits until the bus clock line is set (SCLA0 = 1).

<18>The slave device acknowledges the NACK, halts transmission, and releases the clock stretch state (IICCTL00.WREL = 1) to end communication. Once the slave device releases the clock stretch state, the bus clock line is set (SCLA0 = 1).

<19>Once the master device recognizes that the bus clock line is set (SCLA0 = 1) and after the stop condition setup time has elapsed, the master device sets the bus data line (SDAA0 = 1) and issues a stop condition (SCLA0 = 1 and SDAA0 changes from 0 to 1). The slave device detects the generated stop condition and slave device issue an interrupt (IICA0_TXRXI: stop condition).

Note: <1> to <19> in [\(2\) Example of slave device to master device communications \(8th cycle clock stretching is selected for the master device and 9th cycle clock stretching is selected for the slave device\)](#) represent the entire procedure for communicating data using the I²C bus.

[Figure 20.67](#) shows the processing from <1> to <7>,
[Figure 20.68](#) shows the processing from <3> to <12>, and
[Figure 20.69](#) shows the processing from <8> to <19>.

当从设备将要发送的数据写入 IICA0 寄存器时，从设备设置的时钟拉伸状态被释放。然后，从设备开始向主设备传输数据。

<13>主设备在第 8 个时钟的下降沿发出中断（IICA0_TXRXI：传输结束），并设置时钟拉伸状态(SCLA0 = 0)，由于执行了 ACK 控制（IICCTL00.ACKE= 1），因此在此阶段总线数据线处于低电平(SDAA0 = 0)。

<14>主设备将 NACK 设置为响应(IICCTL00.ACKE = 0) 并将设置时钟拉伸状态的时序更改为第 9 个时钟(IICCTL00.WTIM = 1)。

<15>如果主设备释放时钟拉伸状态（IICCTL00.WREL= 1），则从设备在第 9 个时钟的上升沿检测到 NACK（ACK= 0）。

<16>主设备和从设备在第 9 个时钟的下降沿设置时钟拉伸状态(SCLA0 = 0)，并且主设备和从设备都发出中断（IICA0_TXRXI：传输结束）。

当主设备发出停止条件时，总线数据线被清除，主设备释放时钟拉伸状态。然后，主设备等待总线时钟线被置位。

<18>从设备确认 NACK，停止传输，并释放时钟拉伸状态（IICCTL00.WREL= 1）以结束通信。一旦从设备释放时钟拉伸状态，总线时钟线将被置位(SCLA0 = 1)。

<19>一旦主设备识别到总线时钟线已置位(SCLA0 = 1) 并且停止条件建立时间已过，主设备将总线数据线置位(SDAA0 = 1) 并发出停止条件(SCLA0 = 1，此时 SDAA0 从 0 变为 1)。从设备检测到生成的停止条件，并从设备发出中断（IICA0_TXRXI：停止条件）。

注意: [\(2\) 从设备到主设备通信示例（主设备选择第 8 个周期时钟拉伸，从设备选择第 9 个周期时钟拉伸）](#) 中的<1>到<19>代表使用I²C总线进行数据通信的整个过程。

图 20.67 显示了从<1>到<7>的处理过程，
图 20.68 显示了从<3>到<12>的处理过程，
图 20.69 显示了从<8>到<19>的处理过程。

21. Cyclic Redundancy Check (CRC)

21.1 Overview

The Cyclic Redundancy Check (CRC) generates CRC codes to detect errors in the data. The bit order of CRC calculation results can be selected LSB-first communication. Additionally, two CRC-generation polynomials (16-bit CRC-CCITT and 32-bit CRC-32) are available.

Table 21.1 lists the CRC calculator specifications and Figure 21.1 shows a block diagram.

Table 21.1 CRC calculator specifications

Item	Description	
Data size	8-bit	32-bit
Data for CRC calculation*1	CRC code generated for data in 8n-bit units (where n is a natural number)	CRC code generated for data in 32n-bit units (where n is a natural number)
CRC processor unit	Operation executed on 8 bits in parallel	Operation executed on 32 bits in parallel
CRC generating polynomial	[16-bit CRC] • $X^{16} + X^{12} + X^5 + 1$ (CRC-CCITT).	[32-bit CRC] • $X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$ (CRC-32)
Module-stop function	Module-stop state can be set to reduce power consumption	

Note 1. This function cannot divide data used in CRC calculations. Write data in 8-bit or 32-bit units.

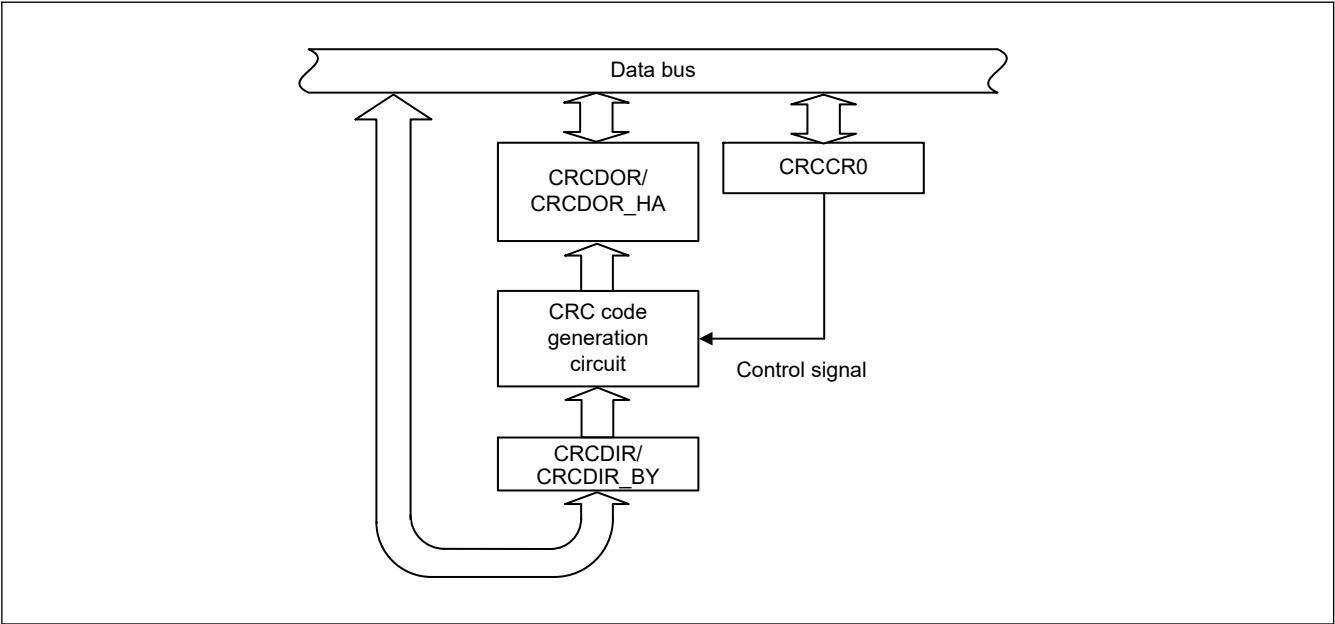


Figure 21.1 CRC calculator block diagram

21.2 Register Descriptions

21.2.1 CRCCR0 : CRC Control Register 0

Base address: CRC = 0x4007_4000							
Offset address: 0x0000							
Bit position:	7	6	5	4	3	2	1 0
Bit field:	DORC LR	—	—	—	—	GPS[2:0]	
Value after reset:	0	0	0	0	0	0	0

21.循环冗余校验 (CRC)

21.1 概述

循环冗余校验 (CRC) 生成 CRC 码以检测数据中的错误。CRC 计算结果的比特顺序可以选择 LSB 优先通信。此外，还提供两种 CRC 生成多项式（16 位 CRC-CCITT 和 32 位 CRC-32）。

表 21.1 列出了 CRC 计算器规格，图 21.1 显示了框图。

表 21.1 CRC 计算器规格

物品	描述	
数据大小	8 位	32 位
CRC 计算数据*1	为 8n 位数据单元（其中 n 为自然数）生成的 CRC 代码	为 32n 位数据单元（其中 n 为自然数）生成的 CRC 代码
CRC处理器单元	并行执行对 8 位的操作	操作在 32 位上并行执行。
CRC生成多项式	[16位 CRC] • $X^{16} + X^{12} + X^5 + 1$ (CRC-CCITT).	[32位 CRC] • $X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$ (CRC-32)
模块停止功能	可以设置模块停止状态以降低功耗	

注1：此函数无法对用于CRC计算的数据进行除法运算。数据应以8位或32位为单位写入。

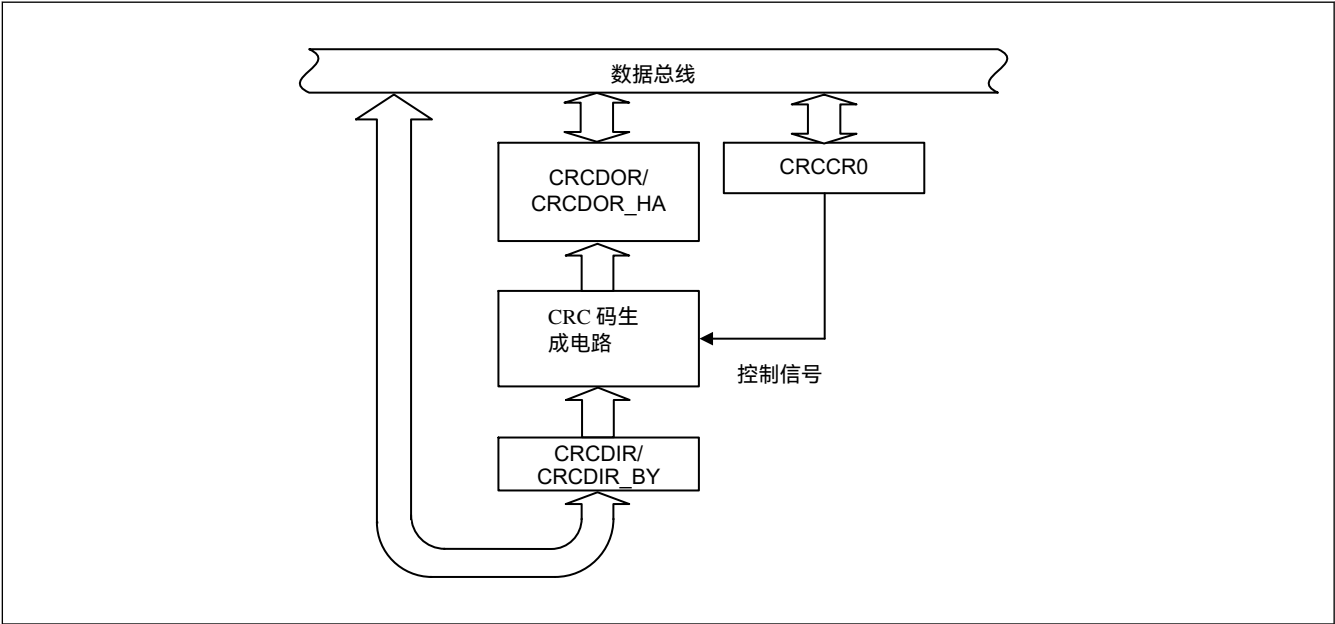


图 21.1 CRC 计算器框图

21.2 寄存器描述

21.2.1CRCCR0: CRC 控制寄存器 0

基地址: CRC = 0x4007_4000							
偏移地址: 0x0000							
位位置:	7	6	5	4	3	2	1 0
位域:	DORC LR	—	—	—	—	GPS[2:0]	
重置后的值:	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
2:0	GPS[2:0]	CRC Generating Polynomial Switching 0 1 1: 16-bit CRC-CCITT ($X^{16} + X^{12} + X^5 + 1$) 1 0 0: 32-bit CRC-32 ($X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$) Others: No calculation is executed	R/W
6:3	—	These bits are read as 0. The write value should be 0.	R/W
7	DORCLR	CRCDOR/CRCDOR_HA Register Clear 0: No effect 1: Clear the CRCDOR/CRCDOR_HA register	W

GPS[2:0] bits (CRC Generating Polynomial Switching)

The GPS[2:0] bits select the CRC generating polynomial.

DORCLR bit (CRCDOR/CRCDOR_HA Register Clear)

Write 1 to the DORCLR bit to set the CRCDOR/CRCDOR_HA register to 0x00000000. This bit is read as 0. Only 1 can be written to it.

21.2.2 CRCDIR/CRCDIR_BY : CRC Data Input Register

Base address:	CRC = 0x4007_4000
Offset address:	0x0004
Bit position:	310
Bit field:	
Value after reset:	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Bit	Symbol	Function	R/W
31:0	n/a	CRC input data The CRCDIR register is a 32-bit read/write register to write data for CRC-32 calculation. The CRCDIR_BY (CRCDIR[31:24], address: 0x4007_4004) is an 8-bit read/write register to write data for CRC-CCITT calculation.	R/W

21.2.3 CRCDOR/CRCDOR_HA : CRC Data Output Register

Base address:	CRC = 0x4007_4000
Offset address:	0x0008
Bit position:	310
Bit field:	
Value after reset:	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Bit	Symbol	Function	R/W
31:0	n/a	CRC output data The CRCDOR register is a 32-bit read/write register for CRC-32 calculation. The CRCDOR_HA (CRCDOR[31:16], address: 0x4007_4008) register is a 16-bit read/write register for CRC-CCITT calculation. Because its initial value is 0x00000000, rewrite the CRCDOR/CRCDOR_HA register to perform the calculations using a value other than the initial value. Data written to the CRCDIR/CRCDIR_BY register is CRC calculated and the result is stored in the CRCDOR/CRCDOR_HA register. If the CRC code is calculated following the transferred data and the result is 0x00000000, there is no CRC error.	R/W

少量	象征	功能	R/W
2:0	GPS[2:0]	CRC 生成多项式交换 0 1 1: 16 位 CRC-CCITT($X^{16} + X^{12} + X^5 + 1$) 1 0 0: 32 位 CRC-32($X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$) 其他: 不执行任何计算。	R/W
6:3	—	这些位读取为 0。写入值应为 0。	R/W
7	DORCLR	CRCDOR/CRCDOR_HA 注册清除 0: 无效果 1: 清除 CRCDOR/CRCDOR_HA 寄存器	W

GPS[2:0] 位（CRC 生成多项式交换）

GPS[2:0] 位选择 CRC 生成多项式。

DORCLR 位（CRCDOR/CRCDOR_HA 寄存器清除）

将 DORCLR 位写入 1，即可将 CRCDOR/CRCDOR_HA 寄存器设置为0x00000000。该位读取值为 0，只能写入 1。

21.2.2 CRCDIR/CRCDIR_BY: CRC 数据输入寄存器

基地址:	CRC = 0x4007_4000
偏移地址:	0x0004
位位置:	310
位域:	
重置后的值:	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

少量	象征	功能	R/W
31:0	n/a	CRC 输入数据 CRCDIR 寄存器是一个 32 位读/写寄存器，用于写入 CRC-32 计算的数据。 CRCDIR_BY (CRCDIR[31:24], 地址: 0x4007_4004) 是一个 8 位读/写寄存器，用于写入 CRC-CCITT 计算的数据。	R/W

21.2.3 CRCDOR/CRCDOR_HA: CRC 数据输出寄存器

基地址:	CRC = 0x4007_4000
偏移地址:	0x0008
位位置:	310
位域:	
重置后的值:	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

少量	象征	功能	R/W
31:0	n/a	CRC 输出数据 CRCDOR 寄存器是一个 32 位读/写寄存器，用于 CRC-32 计算。 CRCDOR_HA (CRCDOR[31:16], 地址: 0x4007_4008) 寄存器是一个 16 位读/写寄存器，用于 CRC-CITT 计算。 因为其初始值为0x00000000，所以重写 CRCDOR/CRCDOR_HA 寄存器，以使用初始值以外的值执行计算。 写入 CRCDIR/CRCDIR_BY 寄存器的数据经过 CRC 计算，结果存储在 CRCDOR/CRCDOR_HA 寄存器中。如果在传输数据之后计算 CRC 码，且结果为0x00000000，则不存在 CRC 错误。	R/W

21.3 Operation

21.3.1 Basic Operation

The CRC calculator generates CRC codes for use in LSB-first.

The following examples show CRC code generation for input data (0xF0) using the 16-bit CRC-CCITT generating polynomial ($X^{16} + X^{12} + X^5 + 1$). In these examples, the value of the CRC Data Output Register (CRCDOR_HA) is cleared before CRC calculation.

When a 32-bit CRC is in use, the valid bits of the CRC code are obtained in CRCDOR.

Figure 21.2 shows the LSB-first data transmission examples respectively. Figure 21.3 shows the LSB-first data reception examples.

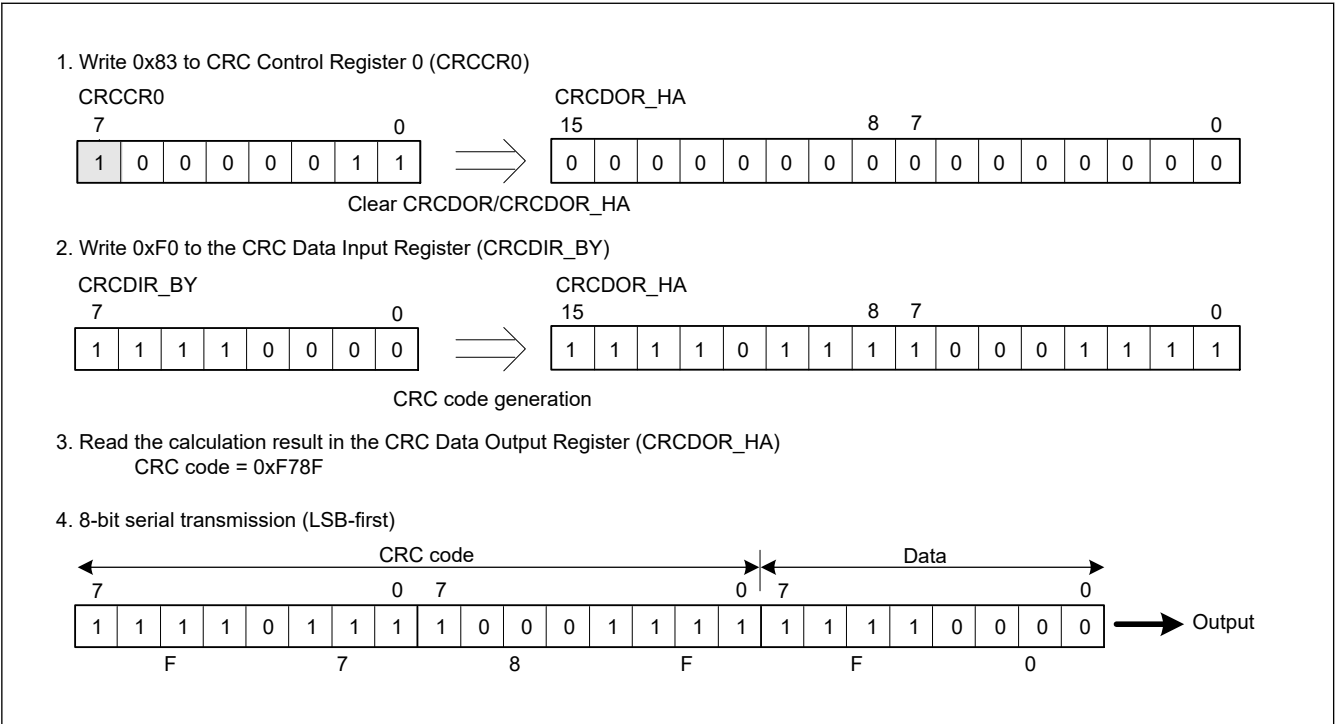


Figure 21.2 LSB-first data transmission

21.3 操作

21.3.1 基本操作

CRC 计算器生成用于 LSB 优先的 CRC 代码。

以下示例展示了如何使用 16 位 CRC-CCITT 生成多项式($X^{16} + X^{12} + X^5 + 1$)为输入数据(0xF0) 生成 CRC 代码。在这些示例中，CRC 数据输出寄存器 (CRCDOR_HA) 的值在 CRC 计算之前会被清除。

当使用 32 位 CRC 时，CRCDOR 中获取 CRC 代码的有效位。

图 21.2 分别展示了 LSB 优先的数据传输示例。图 21.3 展示了 LSB 优先的数据接收示例。

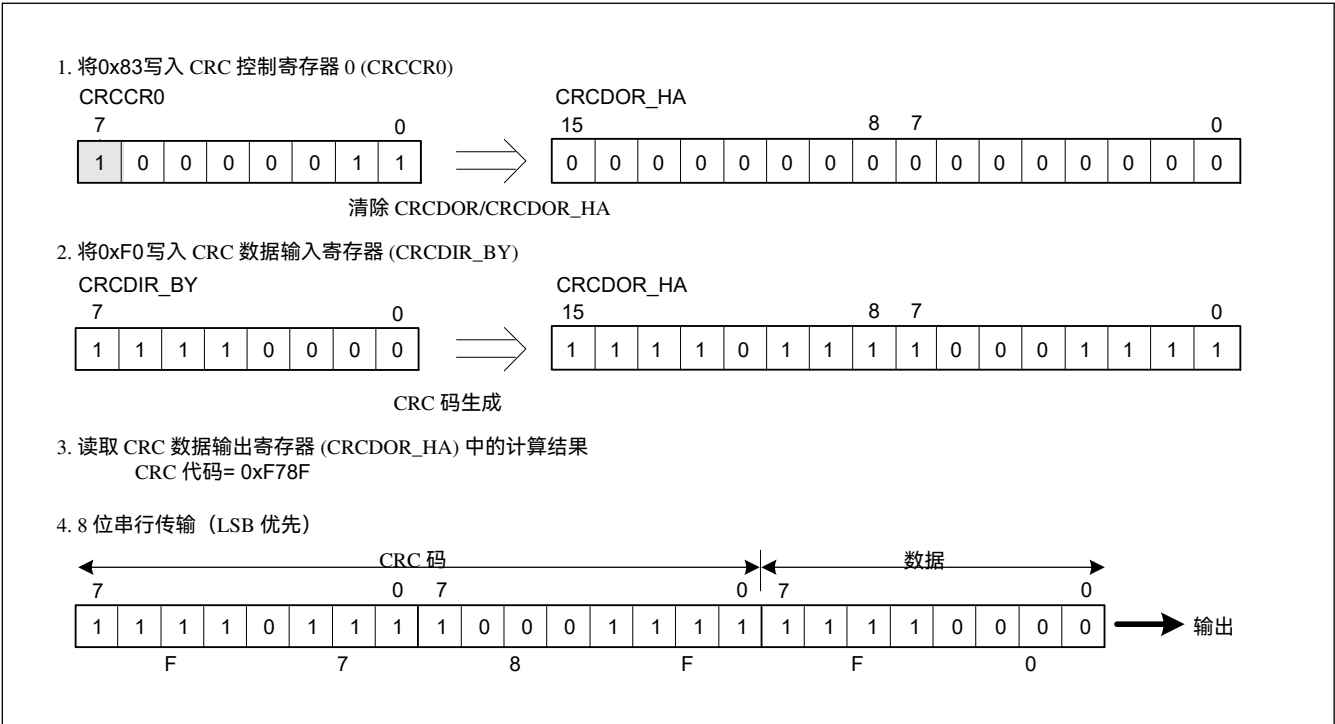


图 21.2 LSB 优先数据传输

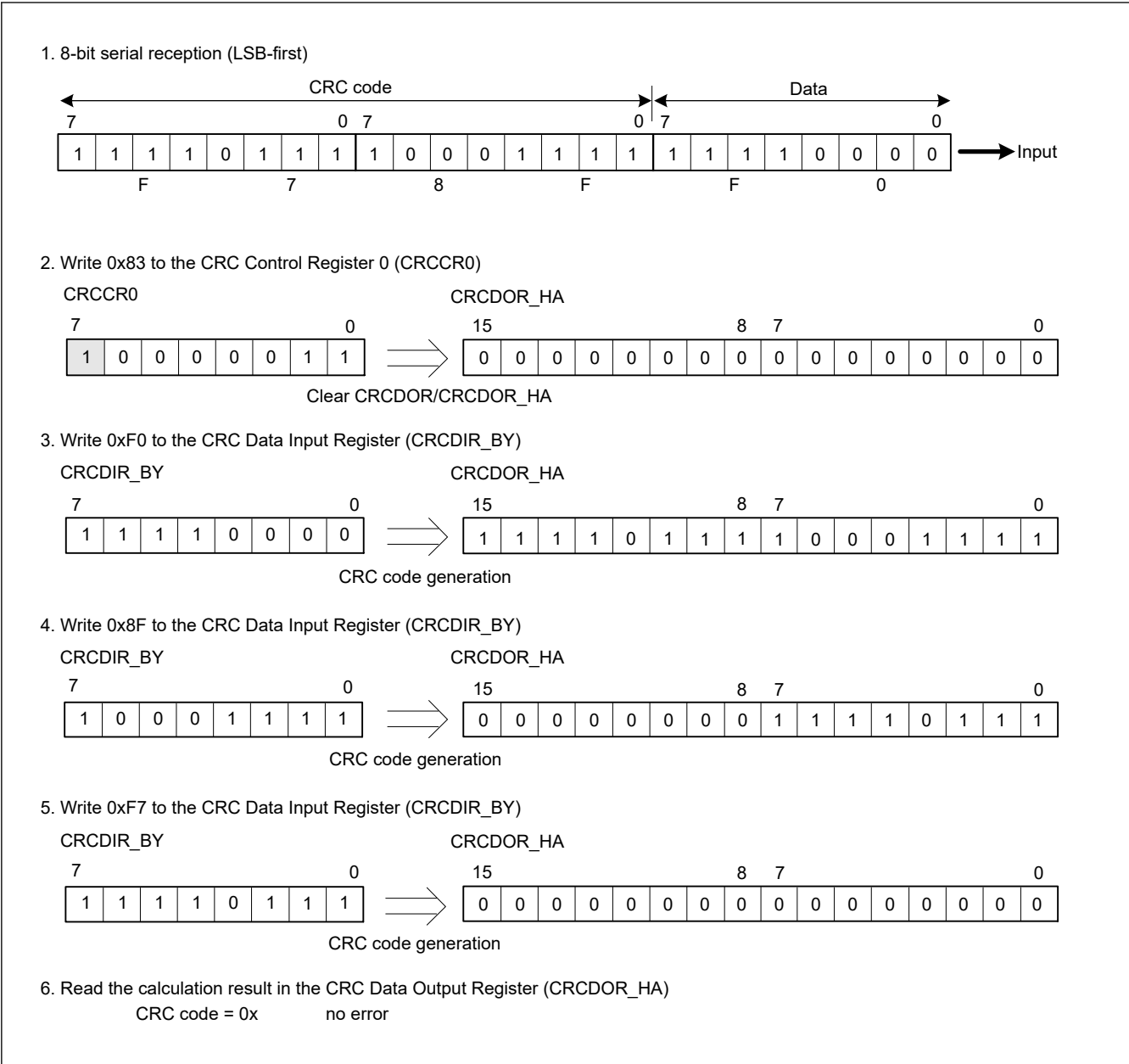


Figure 21.3 LSB-first data reception

21.4 Usage Notes

21.4.1 Settings for the Module-Stop State

The Module Stop Control Register C (MSTPCRC) can enable or disable CRC calculator operation. The CRC calculator is initially stopped after a reset. Releasing the module-stop state enables access to the registers. For details, see [section 9, Low Power Modes](#).

21.4.2 Note on Transmission

The transmission sequence for the CRC code differs based on whether the transmission is LSB-first. [Figure 21.4](#) shows an LSB-first data transmission.

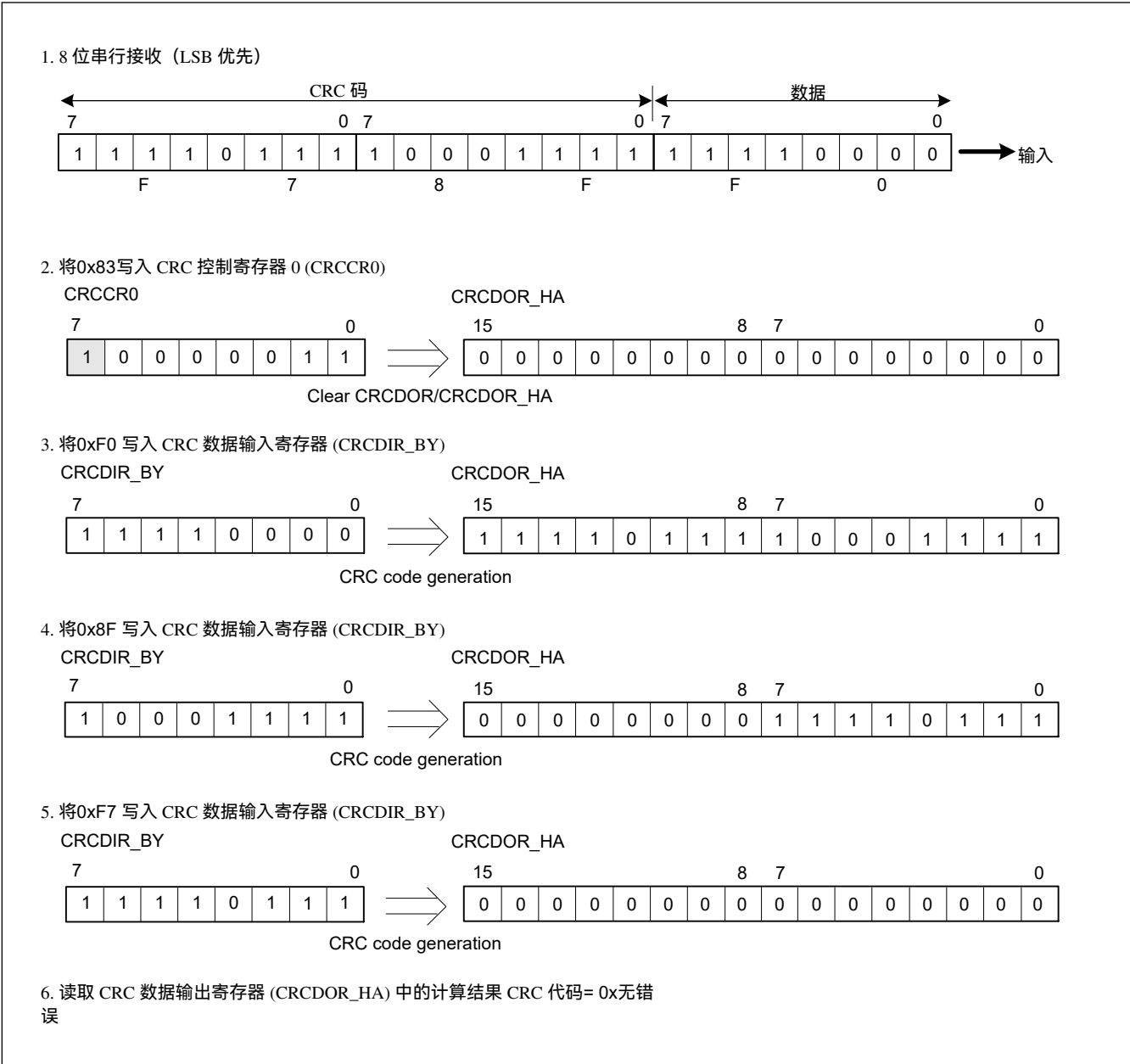


图 21.3 LSB 优先数据接收

21.4 使用说明

21.4.1 模块停止状态的设置

模块停止控制寄存器 C (MSTPCRC) 可以启用或禁用 CRC 计算器运行。复位后，CRC 计算器初始状态为停止。解除模块停止状态后，即可访问寄存器。详情请参见第 9 节“低功耗模式”。

21.4.2 关于变速器的说明

CRC码的传输顺序取决于传输方式是先传输最低有效位 (LSB) 还是先传输最高有效位 (MSB)。图21.4展示了先传输最低有效位 (LSB) 的数据传输方式。

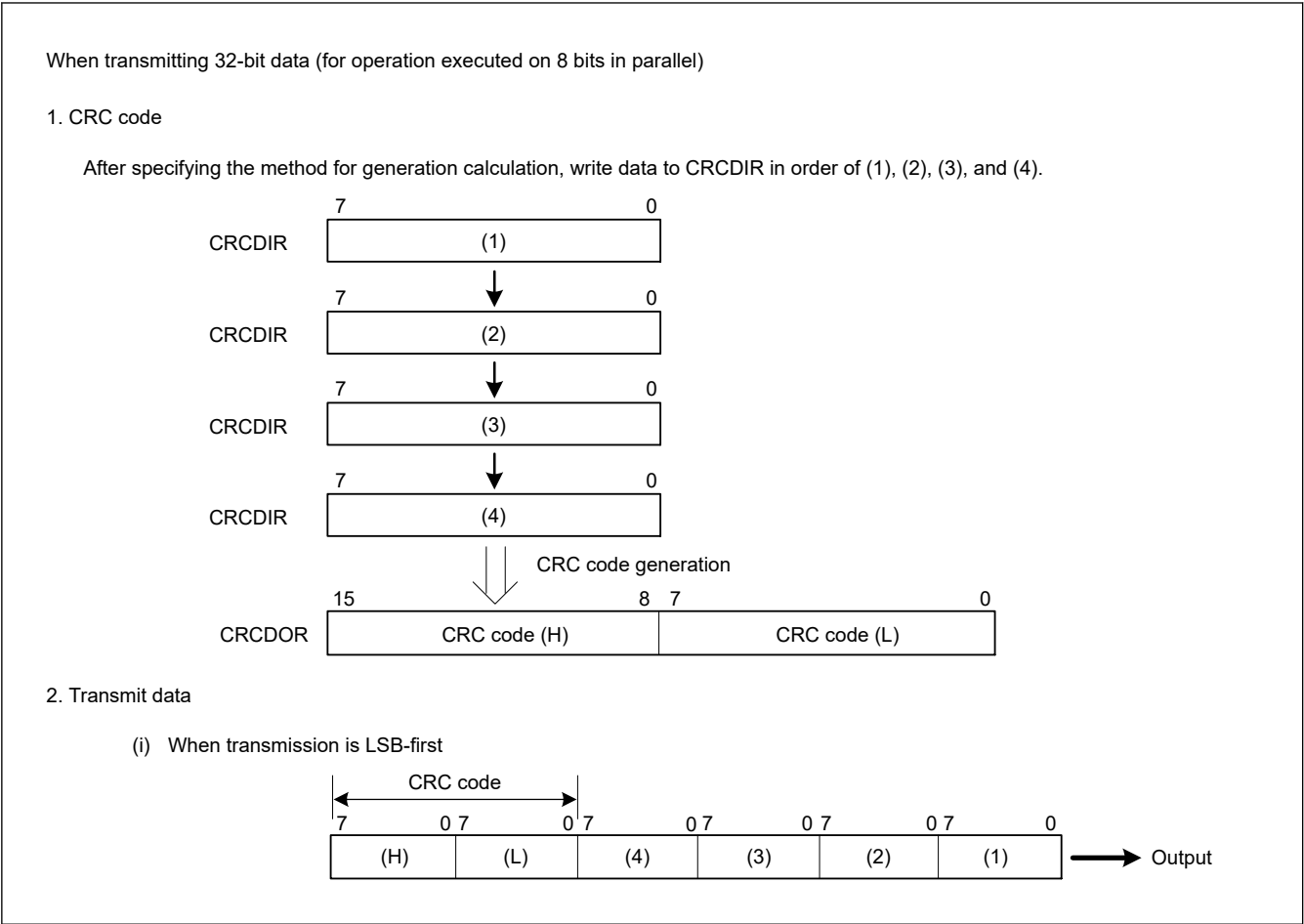


Figure 21.4 LSB-first data transmission

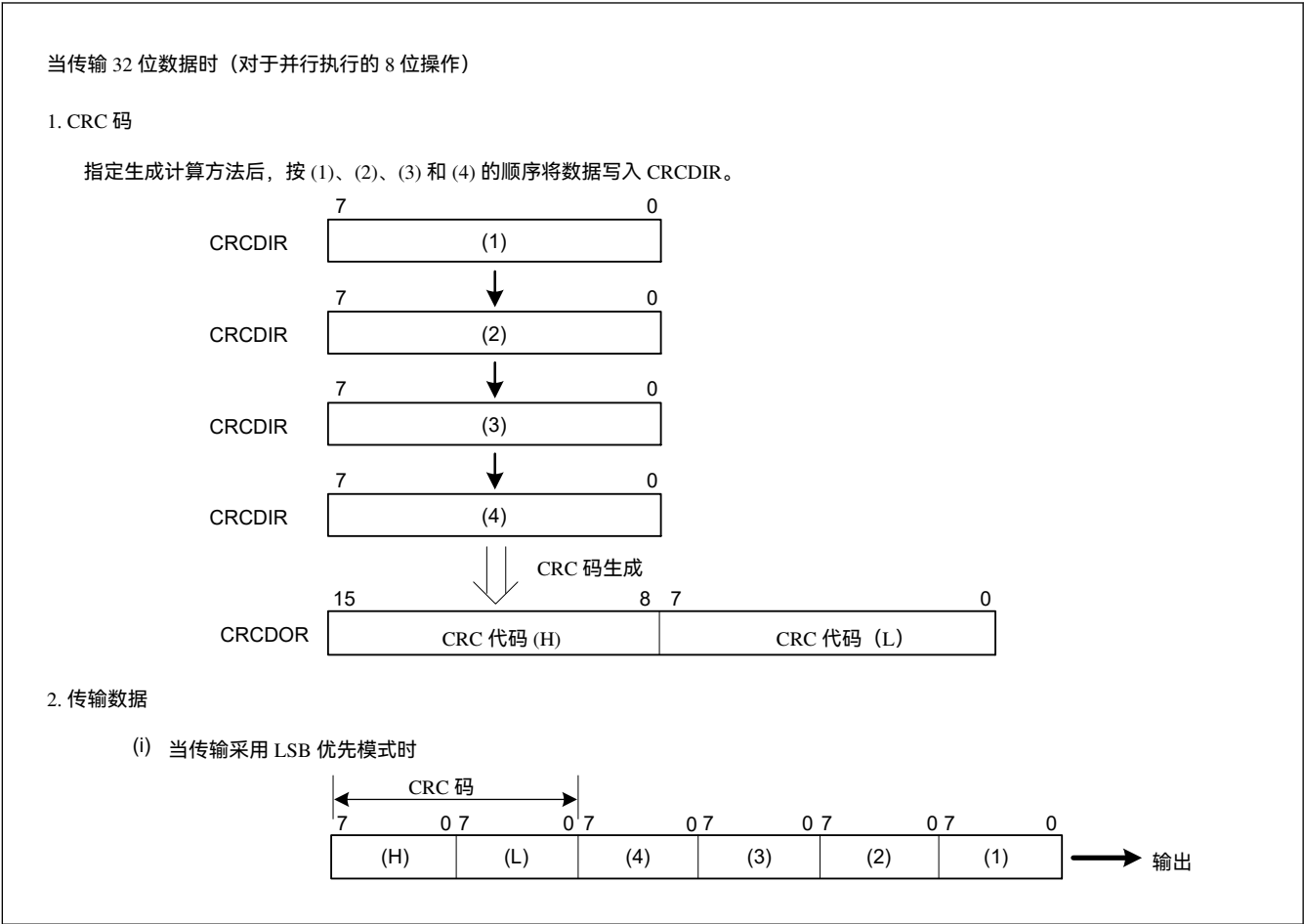


图 21.4 LSB 优先数据传输

22. 10-bit A/D Converter (ADC10)

This is the ADC_D version of the ADC10 peripheral module. ADC_D is referred to as ADC10 in this chapter.

22.1 Overview

The A/D converter is used to convert analog input signals into digital values, and is configured to control up to 6 channels of A/D converter analog inputs (AN000, AN001, AN004, AN005, AN021 and AN022). A 10-bit, or 8-bit resolution can be selected by the ADTYP bit of the A/D converter mode register 2 (ADM2).

Table 22.1 lists the ADC10 specifications and Figure 22.1 shows a block diagram of ADC10.

Table 22.1 ADC10 specifications

Parameter	Specifications
Number of units	One unit
Input channels	Up to 6 channels (AN000, AN001, AN004, AN005, AN021 and AN022)
Extended analog function	Temperature sensor output, internal reference voltage
A/D conversion method	Successive approximation method
Resolution	10-bit/8-bit
Conversion time	For details, see section 22.2.1. ADM0 : A/D Converter Mode Register 0 to section 22.2.1. ADM0 : A/D Converter Mode Register 0 .
A/D conversion clock	PCLKB and A/D conversion clock f _{AD} can be set with the following division ratios : f _{AD} to PCLKB frequency ratio = 1:2, 1:4, 1:5, 1:6, 1:8, 1:16, 1:32, 1:64
Data registers	10-bit A/D conversion result register 8-bit A/D conversion result register
Operating modes	Various A/D conversion modes can be specified by using the mode combinations shown in Table 22.2 and Table 22.3 .
Conditions for A/D conversion start	- Software trigger - Hardware trigger from Timer Array Unit channel 1 count or capture end interrupt - Hardware trigger from 32-bit interval timer interrupt
Functions	10-bit, or 8-bit resolution can be selected by ADTYP bit. - Trigger mode has 3 modes, Software trigger mode, Hardware trigger no-wait mode, and Hardware trigger wait mode. - Channel selection mode has 1 mode, Select mode. - Conversion operation mode has 2 modes, One-shot conversion mode and Sequential conversion mode. - Operation voltage mode has 4 modes, Normal 1, Normal 2, Low voltage 1, and Low voltage 2. - Snooze mode function
Interrupt sources	- In Select mode, an A/D conversion end interrupt request (ADC10_ADI) can be generated on completion of single scan. - The A/D conversion results and ADUL register value are compared, and interrupt signal (ADC10_ADI) generation is controlled in the range specified by the ADRCK bit of A/D converter mode register 2 (ADM2). - The A/D conversion results and ADLL register value are compared, and interrupt signal (ADC10_ADI) generation is controlled in the range specified by the ADRCK bit of A/D converter mode register 2 (ADM2). - ADC10_ADI can activate the Data Transfer Controller (DTC).
Reference voltage	- VREFH0, VCC, or internal reference voltage (BGR) (external reference voltage or output voltage from reference voltage generation circuit) can be selected as the analog reference voltage. - VREFL0 or VSS can be selected as the analog reference ground.
Module-stop function	Module-stop state can be set to reduce power consumption.*1

Note 1. For details, see [section 9, Low Power Modes](#).

22.10 位 A/D 转换器 (ADC10)

这是ADC10外设模块的ADC_D版本。本章中，ADC_D简称为ADC10。

22.1 概述

A/D转换器用于将模拟输入信号转换为数字值，并配置为控制最多 6 个通道的A/D转换器模拟输入（AN000、AN001、AN004、AN005、AN021 和 AN022）。可通过A/D转换器模式寄存器 2 (ADM2) 的 ADTYP 位选择 10 位或 8 位分辨率。

表 22.1 列出了 ADC10 的规格，图 22.1 显示了 ADC10 的框图。

表 22.1ADC10 规格

范围	规格
单位数量	一个单元
输入通道	最多 6 个通道（AN000、AN001、AN004、AN005、AN021 和 AN022）
扩展模拟功能	温度传感器输出，内部参考电压
A/D转换方法	逐次逼近法
解决	10 位/8 位
转换时间	有关详细信息，请参阅第 22.2.1 节。ADM0: A/D 转换器模式寄存器 0 到第 22.2.1 节。ADM0: A/D 转换器模式寄存器 0。
A/D转换时钟	PCLKB 和 A/D 转换时钟 fAD 可以设置以下分频比: :fAD 与 PCLKB 频率比= 1:2，分别为 1:4、1:5、1:6、1:8、1:16、1:32 和 1:64。
数据寄存器	10 位 A/D 转换结果寄存器 8 位 A/D 转换结果寄存器
操作模式	可以使用表 22.2 和表 22.3 中所示的模式组合来指定各种 A/D 转换模式。
A/D转换开始的条件	- 软件触发器 - 硬件触发来自定时器阵列单元通道 1 计数或捕获结束中断 - 硬件触发来自32位间隔定时器中断
函数	- 可通过 ADTYP 位选择 10 位或 8 位分辨率。 - 触发模式有3种模式，软件触发模式、硬件触发无等待模式和硬件触发等待模式。 - 频道选择模式有1种模式，选择模式。 - 转换操作模式有两种模式，单次转换模式和顺序转换模式。 - 工作电压模式有4种模式，正常1、正常2、低电压1和低电压2。 - 贪睡模式功能
中断源	- 在选择模式下，单次扫描完成后可以生成 A/D 转换结束中断请求 (ADC10_ADI)。 - 将 A/D 转换结果与 ADUL 寄存器值进行比较，并在 A/D 转换器模式寄存器 2 (ADM2) 的 ADRCK 位指定的范围内控制中断信号 (ADC10_ADI) 的生成。 - 将 A/D 转换结果与 ADLL 寄存器值进行比较，并在 A/D 转换器模式寄存器 2 (ADM2) 的 ADRCK 位指定的范围内控制中断信号 (ADC10_ADI) 的生成。 - ADC10_ADI 可以激活数据传输控制器 (DTC)。
参考电压	- 可以选择VREFH0、VCC或内部参考电压（BGR）（外部参考电压或参考电压生成电路的输出电压）作为模拟参考电压。 - 可以选择VREFL0或VSS作为模拟参考地。
模块停止功能	可设置模块停止状态以降低功耗。*1

注1：有关详细信息，请参阅第9节“低功耗模式”。

Table 22.2 A/D conversion mode

Function	Mode	Specification
Trigger mode	Software trigger	Conversion is started by software.
	Hardware trigger no-wait mode	Conversion is started by detecting a hardware trigger.
	Hardware trigger wait mode	The power to the A/D converter is turned on by detecting a hardware trigger while the A/D converter is off and in the conversion standby state, and conversion is then started automatically after the stabilization wait time passes. When using the Snooze mode function, specify the hardware trigger wait mode.
Channel selection mode	Select mode	A/D conversion is performed on the analog input of one selected channel.
Conversion operation mode	One-shot conversion mode	A/D conversion is performed on the selected channel once.
	Sequential conversion mode	A/D conversion is sequentially performed on the selected channels until it is stopped by software.

Table 22.3 shows the sampling clock cycle for each operation voltage mode.

Table 22.3 Sampling clock cycle for each operation voltage mode

Operation voltage mode*1	Sampling clock cycles	
Normal mode 1	7 f _{AD}	The sampling time in Normal 1 or Low voltage 1 mode is seven cycles of the conversion clock (f _{AD}). Select this mode when the output impedance of the analog input source is high and the sampling time should be long.
Low voltage mode 1	7 f _{AD}	
Normal mode 2	5 f _{AD}	The sampling time in Normal 2 or Low voltage 2 mode is five cycles of the conversion clock (f _{AD}). Select this mode when enough sampling time is ensured (for example, when the output impedance of the analog input source is low).
Low voltage mode 2	5 f _{AD}	

Note 1. The operation mode that can be selected differs depending on the analog input channel, VCC voltage, VREFH0 voltage, trigger mode, and PCLKB. See [section 22.2.1. ADM0 : A/D Converter Mode Register 0](#) for details.

表 22.2A/D 转换模式

功能	模式	规格
触发模式	软件触发器	转换是由软件启动的。
	硬件触发无等待模式	转换通过检测硬件触发器启动。
	硬件触发等待模式	当模数转换器处于关闭状态并处于转换待机状态时，通过检测硬件触发信号来开启模数转换器的电源，并在稳定等待时间过后自动开始转换。使用“休眠模式”功能时，请指定硬件触发等待模式。
频道选择模式	选择模式	对选定通道的模拟输入进行 A/D 转换。
转换操作模式	单次转换模式	对选定的通道执行一次 A/D 转换。
	顺序转换模式	对选定的通道依次执行 A/D 转换，直到软件停止为止。

表 22.3 显示了每种工作电压模式的采样时钟周期。

表 22.3 各工作电压模式下的采样时钟周期

工作电压模式*1	采样时钟周期	
普通模式 1	7 f _{AD}	正常模式 1 或低电压模式 1 的采样时间为七个周期。转换时钟（f _{AD} ）。当模拟输入源的输出阻抗较高且采样时间需要较长时，请选择此模式。
低电压模式 1	7 f _{AD}	
普通模式 2	5 f _{AD}	正常模式 2 或低电压模式 2 的采样时间为五个周期。转换时钟(f _{AD})。当保证足够的采样时间时（例如，当模拟输入源的输出阻抗较低时），选择此模式。
低电压模式 2	5 f _{AD}	

注1. 可选择的操作模式取决于模拟输入通道、VCC电压、VREFH0电压、触发模式和PCLKB。详情请参见22.2.1节。ADM0: A/D转换器模式寄存器0。

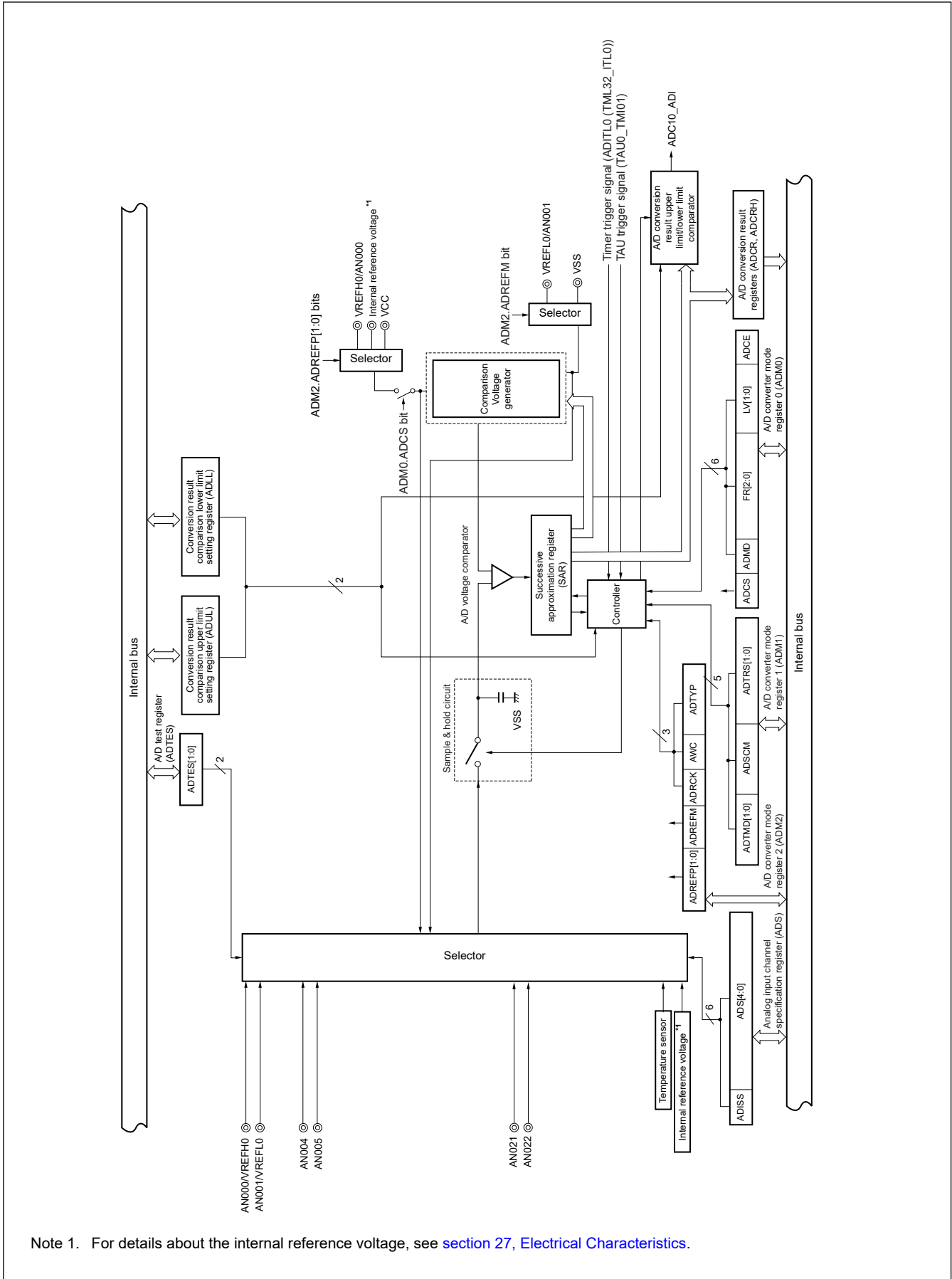


Figure 22.1 Block diagram of A/D converter

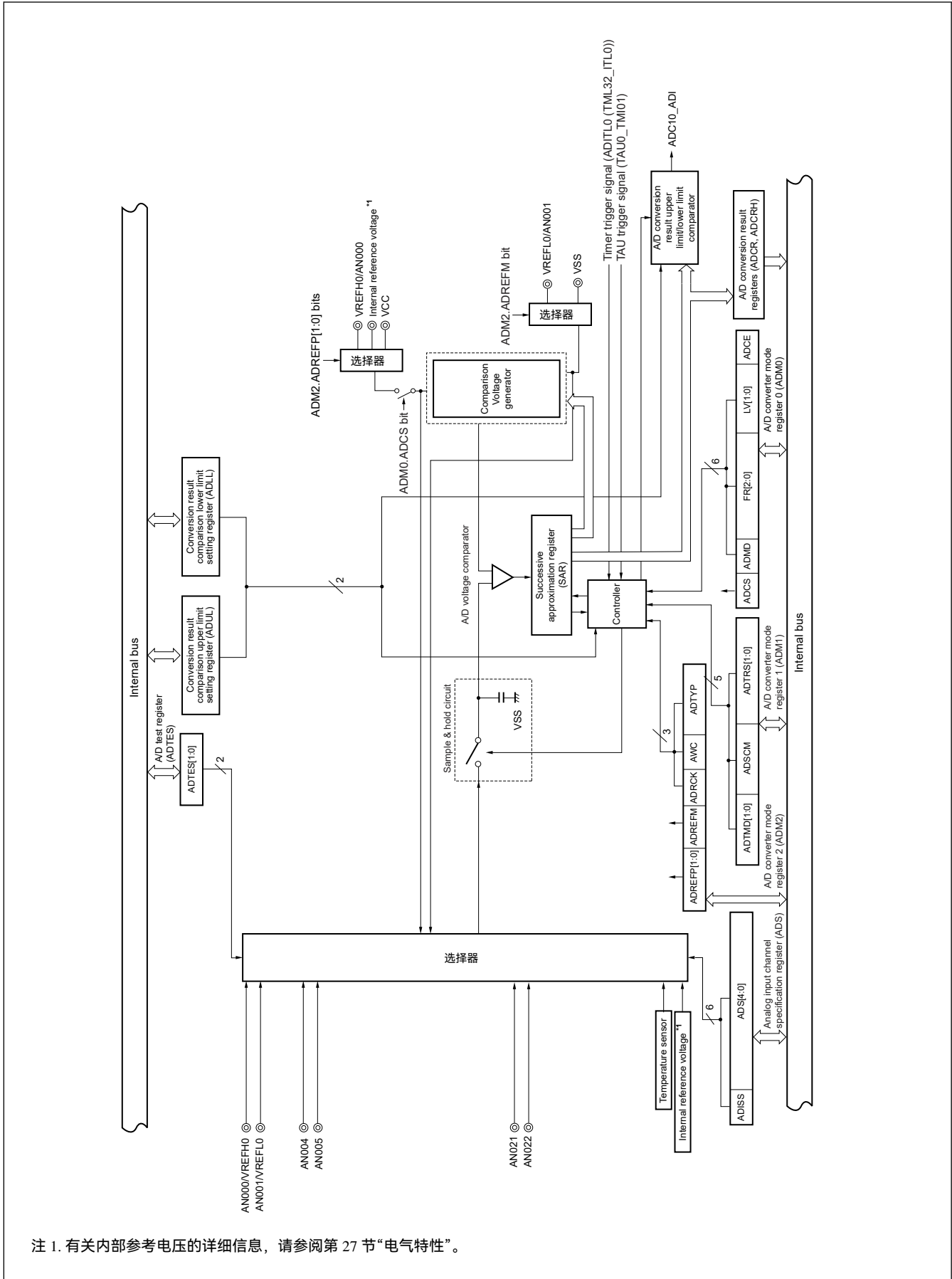


图 22.1 A/D 转换器框图

The A/D converter includes the following hardware.

1. AN000, AN001, AN004, AN005, AN021 and AN022 pins
These are the analog input pins of the 6 channels of the A/D converter. They input analog signals to be converted into digital signals. Pins other than the one selected as the analog input pin can be used as I/O port pins.
2. Sample & hold circuit
The sample & hold circuit samples each of the analog input voltages sequentially sent from the input circuit, and sends them to the A/D voltage comparator. This circuit also holds the sampled analog input voltage during A/D conversion.
3. A/D voltage comparator
This A/D voltage comparator compares the voltage generated from the voltage tap of the comparison voltage generator with the analog input voltage. If the analog input voltage is found to be greater than the reference voltage ($1/2 AV_{REF}$) as a result of the comparison, the most significant bit (MSB) of the successive approximation register (SAR) is set. If the analog input voltage is less than the reference voltage ($1/2 AV_{REF}$), the MSB of the SAR is reset. After that, bit 8 of the SAR register is automatically set, and the next comparison is made. The voltage tap of the comparison voltage generator is selected by the value of bit 9, to which the result has already been set.

- Bit 9 = 0: ($1/4 AV_{REF}$)
- Bit 9 = 1: ($3/4 AV_{REF}$)

The voltage tap of the comparison voltage generator and the analog input voltage are compared and bit 8 of the SAR register is manipulated according to the result of the comparison.

- Analog input voltage $\geq$ Voltage tap of comparison voltage generator: Bit 8 = 1
- Analog input voltage $\leq$ Voltage tap of comparison voltage generator: Bit 8 = 0

Comparison is continued like this to bit 0 of the SAR register.

When performing A/D conversion at a resolution of 8 bits, the comparison continues until bit 2 of the SAR register.

AV_{REF} : The ‘+’ side reference voltage of the A/D converter. This can be selected from VREFH0, the internal reference voltage*1, and VCC.

Note 1. For details about the internal reference voltage, see [section 27, Electrical Characteristics](#).

4. Comparison voltage generator
The comparison voltage generator generates the voltage to be compared with the input from an analog input pin.
5. Successive approximation register (SAR)
The SAR is used to set voltage tap data whose values from the comparison voltage generator match the voltage values of the analog input pins, one bit at a time starting from the most significant bit (MSB).
If data is set in the SAR register all the way to the least significant bit (LSB) (end of A/D conversion), the contents of the SAR register (conversion results) are held in the A/D conversion result register (ADCR). When all the specified A/D conversion operations have ended, an A/D conversion end interrupt request signal (ADC10_ADI) is generated.
6. 10-bit A/D conversion result register (ADCR)
The A/D conversion result is loaded from the successive approximation register to this register each time A/D conversion is completed, and the ADCR register holds the A/D conversion result in its higher 10 bits (the lower 6 bits are fixed to 0).
7. 8-bit A/D conversion result register (ADCRH)
The A/D conversion result is loaded from the successive approximation register to this register each time A/D conversion is completed, and the ADCRH register holds the higher 8 bits of the A/D conversion result.
8. Controller
This circuit controls the conversion time of an analog input signal that is to be converted into a digital signal, as well as starting and stopping of the conversion operation. When A/D conversion has been completed, this controller generates ADC10_ADI through the A/D conversion result upper limit/lower limit comparator.
9. VREFH0 pin
This pin inputs an external reference voltage (VREFH0).
If using VREFH0 as the ‘+’ side reference voltage of the A/D converter, set the ADREFP[1:0] bits of A/D converter mode register 2 (ADM2) to 01b, respectively.
The analog signals input to AN000, AN001, AN004, AN005, AN021 and AN022 are converted to digital signals based on the voltage applied between VREFH0 and the ‘-’ side reference voltage (VREFL0/V_{SS}).

A/D转换器包含以下硬件。

1. **AN000、AN001、AN004、AN005、AN021 和 AN022 引脚**
这些是模数转换器6个通道的模拟输入引脚。它们输入模拟信号，并将其转换为数字信号。除选定的模拟输入引脚外，其他引脚均可用作I/O端口引脚。
2. **采样保持电路**
采样保持电路依次对输入电路发送的每个模拟输入电压进行采样，并将采样结果发送到模数转换电压比较器。该电路还在A/D转换期间保持采样到的模拟输入电压。
3. **A/D电压比较器**
该A/D电压比较器将比较电压发生器的电压抽头产生的电压与模拟输入电压进行比较。如果比较结果显示模拟输入电压大于参考电压($1/2 AV_{REF}$)，则逐次逼近寄存器（SAR）的最高有效位（MSB）被置位。如果比较结果显示模拟输入电压小于参考电压($1/2 AV_{REF}$)，则SAR的最高有效位被复位。之后，SAR寄存器的第8位自动置位，并进行下一次比较。比较电压发生器的电压抽头由第9位的值决定，该位的值已被置位。

- 它将是t 9 = 0: ($1/4 AV_{REF}$)
- Bit 9 = 1: ($3/4 AV_{REF}$)

将比较电压发生器的电压抽头与模拟输入电压进行比较，并根据比较结果对 SAR 寄存器的第 8 位进行操作。

- 模拟输入电压 $\geq$ 比较电压发生器的电压抽头：位8 = 1
- 模拟输入电压 $\leq$ 比较电压发生器的电压抽头：位8 = 0

如此继续比较，直到 SAR 寄存器的第 0 位。

当以 8 位分辨率执行A/D转换时，比较将持续到 SAR 寄存器的第 2 位。 AV_{REF} : A/D 转换器的‘+’侧参考电压。该电压可从 VREFH0、内部参考电压*1 和 VCC 中选择。

注 1. 有关内部参考电压的详细信息，请参阅第 27 节“电气特性”。

4. **比较电压发生器**
比较电压发生器产生要与模拟输入引脚输入的电压进行比较的电压。
5. **逐次逼近寄存器（SAR）**
SAR 用于设置电压抽头数据，该数据的值来自比较电压发生器，与模拟输入引脚的电压值相匹配，从最高有效位 (MSB) 开始，一次匹配一位。
如果 SAR 寄存器中的数据一直置位到最低有效位 (LSB) (A/D 转换结束)，则 SAR 寄存器的内容 (转换结果) 保存在转换结果寄存器 (ADCR) 中。当所有指定的 A/D 转换操作都完成后，会生成转换结束中断请求信号 (ADC10_ADI)。
6. **10 位 A/D 转换结果寄存器 (ADCR)**
每次 A/D 转换完成后，都会将 A/D 转换结果从逐次逼近寄存器加载到该寄存器中，ADCR 寄存器在其高 10 位中保存 A/D 转换结果（低 6 位固定为 0）。
7. **8 位 A/D 转换结果寄存器 (ADCRH)**
每次 A/D 转换完成后，都会将 A/D 转换结果从逐次逼近寄存器加载到该寄存器中，ADCRH 寄存器保存 A/D 转换结果的高 8 位。
8. **控制器**
该电路控制模拟输入信号转换为数字信号的转换时间，以及转换操作的启动和停止。当模数转换完成后，该控制器通过模数转换结果的上限/下限比较器生成 ADC10_ADI。
9. **VREFH0 引脚**
该引脚输入外部参考电压（VREFH0）。
如果使用 VREFH0 作为 A/D 转换器的‘+’侧参考电压，则将 A/D 转换器模式寄存器 2 (ADM2) 的 ADREFP[1:0] 位分别设置为 01b。
输入到 AN000、AN001、AN004、AN005、AN021 和 AN022 的模拟信号根据 VREFH0 和‘-’侧参考电压(VREFL0/V_{SS})之间施加的电压转换为数字信号。

In addition to VREFH0, it is possible to select V_{CC} or the internal reference voltage*1 as the ‘+’ side reference voltage of the A/D converter.

Note 1. For details about the internal reference voltage, see [section 27, Electrical Characteristics](#).

10. VREFL0 pin

This pin inputs an external reference voltage (VREFL0). To use VREFL0 as the ‘-’ side reference voltage of the A/D converter, set the ADREFM bit of the ADM2 register to 1.
In addition to VREFL0, it is possible to select V_{SS} as the ‘-’ side reference voltage of the A/D converter.

11. Testing of the A/D converter

This test checks whether the A/D converter is operating normally by converting the positive and negative reference voltages, analog input channels (ANxxx), temperature sensor output voltage, and internal reference voltage of the A/D converter.

22.2 Registers to Control the A/D Converter

The following registers are used to control the A/D converter.

- [section 22.2.1. ADM0 : A/D Converter Mode Register 0](#)
- [section 22.2.2. ADM1 : A/D Converter Mode Register 1](#)
- [section 22.2.3. ADM2 : A/D Converter Mode Register 2](#)
- [section 22.2.4. ADCR: 10-bit A/D Conversion Result Register](#)
- [section 22.2.5. ADCRH : 8-bit A/D Conversion Result Register](#)
- [section 22.2.6. ADS : Analog Input Channel Specification Register](#)
- [section 22.2.7. ADUL : Conversion Result Comparison Upper Limit Setting Register](#)
- [section 22.2.8. ADLL : Conversion Result Comparison Lower Limit Setting Register](#)
- [section 22.2.9. ADTES : A/D Test Register](#)

22.2.1 ADM0 : A/D Converter Mode Register 0

Base address: ADC_D = 0x400A_1800							
Offset address: 0x0000							
Bit position:	7	6	5	4	3	2	1 0
Bit field:	ADCS	—	FR[2:0]		LV[1:0]		ADCE
Value after reset:	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
0	ADCE	A/D voltage comparator operation control*2 0: Stops A/D voltage comparator operation 1: Enables A/D voltage comparator operation	R/W
2:1	LV[1:0]*1	Select Operation voltage mode 0 0: Normal mode 1 0 1: Normal mode 2 1 0: Low voltage mode 1 1 1: Low voltage mode 2	R/W
5:3	FR[2:0]*1	Select Conversion Clock (f_{AD}) 0 0 0: PCLKB/64 0 0 1: PCLKB/32 0 1 0: PCLKB/16 0 1 1: PCLKB/8 1 0 0: PCLKB/6 1 0 1: PCLKB/5 1 1 0: PCLKB/4 1 1 1: PCLKB/2	R/W

除了 VREFH0 之外，还可以选择 V_{CC} 或内部参考电压*1 作为 A/D 转换器的‘+’ 侧参考电压。

注 1. 有关内部参考电压的详细信息，请参阅第 27 节“电气特性”。

10. VREFL0 引脚

该引脚输入外部参考电压 (VREFL0)。要将 VREFL0 用作 A/D 转换器的“-”侧参考电压，请将 ADM2 寄存器的 ADREFM 位设置为 1。
除了 VREFL0 之外，还可以选择 V_{SS} 作为 A/D 转换器的“-”侧参考电压。

11. A/D转换器的测试

该测试通过转换A/D转换器的正负参考电压、模拟输入通道（ANxxx）、温度传感器输出电压和内部参考电压来检查A/D转换器是否正常工作。

22.2 控制模数转换器的寄存器

以下寄存器用于控制 A/D 转换器。

- [第 22.2.1 节。ADM0: A/D 转换器模式寄存器 0](#)
- [第 22.2.2 节。ADM1: A/D 转换器模式寄存器 1](#)
- [第 22.2.3 节 ADM2: A/D 转换器模式寄存器 2](#)
- [第 22.2.4 节 ADCR: 10 位 A/D 转换结果寄存器](#)
- [第 22.2.5 节。ADCRH: 8位 A/D 转换结果寄存器](#)
- [第 22.2.6 节 ADS: 模拟输入通道规范寄存器](#)
- [第 22.2.7 节 ADUL: 转换结果比较上限设置寄存器](#)
- [第 22.2.8 节 ADLL: 转换结果比较下限设置寄存器](#)
- [第 22.2.9 节。ADTES: A/D 测试寄存器](#)

22.2.1ADM0: A/D转换器模式寄存器0

基地址: ADC_D = 0x400A_1800							
偏移地址: 0x0000							
位位置:	7	6	5	4	3	2	1 0
位域:	ADCS	—	FR[2:0]		LV[1:0]		ADCE
重置后的值:	0	0	0	0	0	0	0

少量	象征	功能	R/W
0	ADCE	A/D电压比较器运行控制*2 0: 停止模数转换电压比较器工作 1: 启用A/D电压比较器操作	R/W
2:1	LV[1:0]*1	选择工作电压模式 0 0: 正常模式 1 0 1: 普通模式 2 1 0: 低电压模式 1 1.1: 低电压模式 2	R/W
5:3	FR[2:0]*1	选择转换时钟（fAD） 0 0 0: PCLKB/64 0 0 1: PCLKB/32 0 1 0: PCLKB/16 0 1 1: PCLKB/8 1 0 0: PCLKB/6 1 0 1: PCLKB/5 1 1 0: PCLKB/4 1 1 1: PCLKB/2	R/W

Bit	Symbol	Function	R/W
6	—	This bit is read as 0. The write value should be 0.	R/W
7	ADCS	A/D conversion operation control 0: Stops conversion operation [When read] - Conversion is stopped or in standby 1: Enables conversion operation [When read] - While in the no wait mode (both software and hardware trigger mode): Conversion is enabled - While in the wait mode (both software and hardware trigger mode): A/D power supply stabilization wait time + conversion	R/W

Note 1. For details of the FR[2:0], LV[1:0] bits, and A/D conversion, see [Table 22.8](#) to [Table 22.9](#).
Note 2. While in the software trigger mode or hardware trigger no-wait mode, the operation of the A/D voltage comparator is controlled by the ADCS and ADCE bits, and it takes 1 μs from the start of operation for the operation to stabilize. Therefore, when the ADCS bit is set to 1 after 1 μs or more has elapsed from the time ADCE bit is set to 1, the conversion result at that time has priority over the first conversion result. Otherwise, ignore data of the first conversion.

This register sets the time for converting analog input to digital data, and starts and stops conversion.

Note: Do not set the ADCS bit to 1 and the ADCE bit to 0 at the same time.
Change the ADMD, FR2 to FR0, LV1, and LV0 bits while conversion is stopped (ADCS = 0, ADCE = 0).

Note: Do not set the ADCS bit to 1 and the ADCE bit to 0 at the same time.

Note: Do not change the ADCS and ADCE bits from 0 to 1 at the same time by using an 8-bit manipulation instruction. Be sure to follow the procedure described in [section 22.6. A/D Converter Setup Procedure](#).

Table 22.4 Relationship between ADCS and ADCE bits, including A/D operation status

ADCS	ADCE	A/D conversion mode	A/D operation state
0	0	All modes	Conversion stopped state
0	1	Hardware trigger wait mode	Trigger standby state
		Other than hardware trigger wait mode	Conversion standby state
1	0	Setting prohibited	
1	1	Hardware trigger no-wait mode	Trigger standby state or conversion operation state
		Hardware trigger wait mode or Software trigger	Conversion operation state

[Table 22.5](#) shows the conditions for setting and clearing the ADCS bit.

Table 22.5 Conditions for setting and clearing the ADCS bit (1 of 2)

A/D conversion mode			Set conditions	Clear conditions
Software trigger	Select mode	Sequential conversion mode	When 1 is written to ADCS	When 0 is written to ADCS
		One-shot conversion mode		- When 0 is written to ADCS - The bit is automatically cleared to 0 when A/D conversion ends.
Hardware trigger no-wait mode	Select mode	Sequential conversion mode		When 0 is written to ADCS
		One-shot conversion mode		When 0 is written to ADCS

少量	象征	功能	R/W
6	—	该位读取值为 0。写入值应为 0。	R/W
7	ADCS	A/D转换操作控制 0: 停止转换操作 [读取时] - 转换已停止或处于待机状态 1: 启用转换操作 [读取时] - 在无需等待模式（软件触发模式和硬件触发模式）下：转换功能已启用 - 在等待模式（软件触发模式和硬件触发模式）下：A/D电源稳定等待时间+转换	R/W

注 1. 有关 FR[2:0]、LV[1:0] 位和 A/D 转换的详细信息，请参阅表 22.8 至表 22.9。
注2: 在软件触发模式或硬件触发无等待模式下，A/D电压比较器的运行由ADCS和ADCE位控制，从运行开始到稳定运行需要1 μs时间。因此，当ADCS位在ADCE位置为1之后经过1 μs或更长时间后置为1时，此时的转换结果优先于第一次转换结果。否则，忽略第一次转换的数据。

该寄存器设置将模拟输入转换为数字数据的时间，并启动和停止转换。

注意: 不要同时将 ADCS 位设置为 1，将 ADCE 位设置为 0。
在转换停止时，将 ADMD、FR2 更改为 FR0、LV1 和 LV0 位(ADCS = 0、ADCE = 0)。

注意: 不要同时将 ADCS 位设置为 1，将 ADCE 位设置为 0。

注意: 请勿使用 8 位操作指令同时将 ADCS 和 ADCE 位从 0 更改为 1。务必遵循第 22.6 节“A/D 转换器设置步骤”中描述的步骤。

表 22.4 ADCS 和 ADCE 位之间的关系，包括 A/D 工作状态

ADCS	ADCE	A/D转换模式	A/D 工作状态
0	0	所有模式	转换停止状态
0	1	硬件触发等待模式	触发待机状态
		除了硬件触发等待模式之外	转换待机状态
1	0	设置禁止	
1	1	硬件触发无等待模式	触发待机状态或转换操作状态
		硬件触发等待模式或软件触发	转换操作状态

表 22.5 显示了设置和清除 ADCS 位的条件。

表 22.5 设置和清除 ADCS 位的条件 (1/2)

A/D转换模式			设置条件	晴朗的天气
软件触发器	选择模式	顺序转换模式	当 1 被写入 ADCS	当 0 被写入 ADCS 时
		单次转换模式		- 当 0 被写入 ADCS 时 - 当 A/D 转换结束时，该位会自动清除为 0。
硬件触发等待模式	选择模式	顺序转换模式		当 0 被写入 ADCS 时
		单次转换模式		当 0 被写入 ADCS 时

Table 22.5 Conditions for setting and clearing the ADCS bit (2 of 2)

A/D conversion mode			Set conditions	Clear conditions
Hardware trigger wait mode	Select mode	Sequential conversion mode	When a hardware trigger is input	When 0 is written to ADCS
		One-shot conversion mode		- When 0 is written to ADCS - The bit is automatically cleared to 0 when A/D conversion ends.

The timing when using the A/D voltage comparator is shown in Figure 22.2.

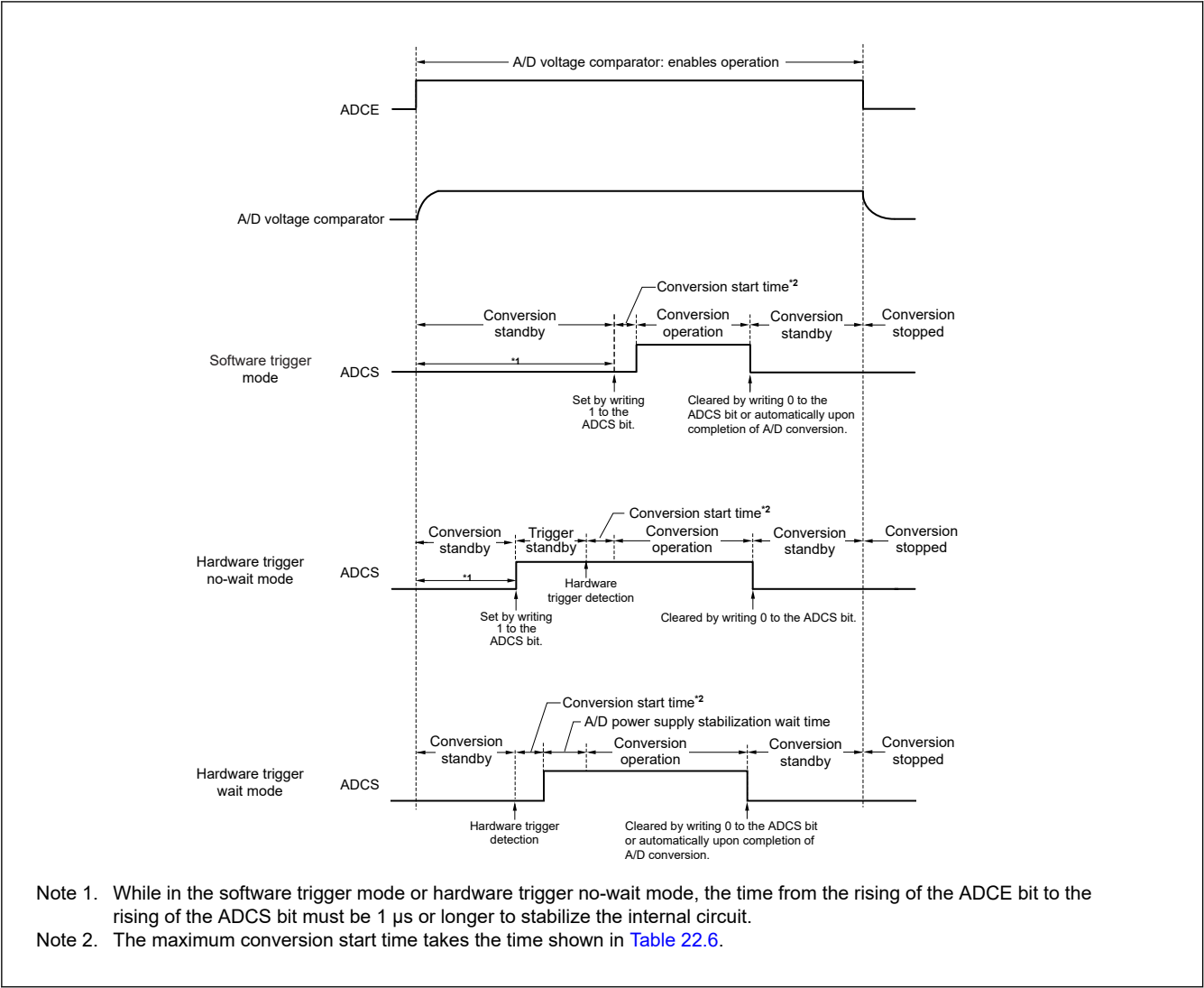


Figure 22.2 Timing when 10-bit A/D Converter is used

Table 22.6 shows the conversion start time with FR[2:0] bits setting.

Table 22.6 Settings of conversion start time (1 of 2)

ADM0	Conversion clock (f_{AD})	Conversion start time (number of PCLKB clock)	
		Software trigger mode/Hardware trigger no-wait mode	Hardware trigger wait mode
000b	PCLKB/64	63	1
001b	PCLKB/32	31	1
010b	PCLKB/16	15	1

表 22.5 设置和清除 ADCS 位的条件 (2/2)

A/D转换模式			设置条件	晴朗的天气
硬件触发等待模式	选择模式	顺序转换模式	当输入硬件触发器时	当 0 被写入 ADCS 时
		单次转换模式		- 当 0 被写入 ADCS 时 - 当 A/D 转换结束时，该位会自动清除为 0。

使用A/D 电压比较器的时序如图22.2所示。

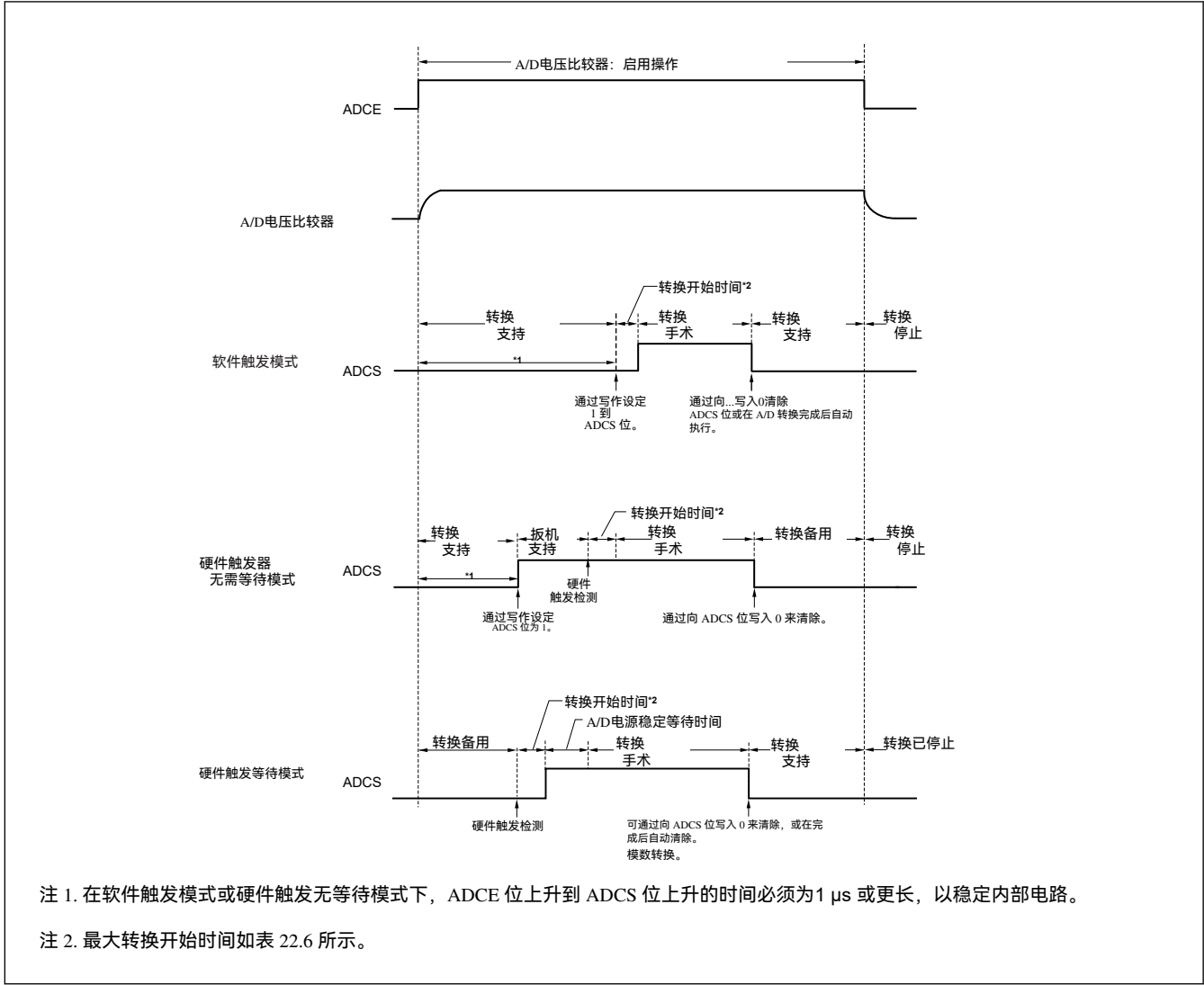


图 22.2 使用 10 位 A/D 转换器时的时序

表 22.6 显示了 FR[2:0] 位设置下的转换开始时间。

表 22.6 转换开始时间设置 (1/2)

ADM0	转换时钟 (f_{AD})	转换开始时间 (PCLKB 时钟周期数)	
		软件触发模式/硬件触发无等待模式	硬件触发等待模式
000b	PCLKB/64	63	1
001b	PCLKB/32	31	1
010b	PCLKB/16	15	1

Table 22.6 Settings of conversion start time (2 of 2)

ADM0	Conversion clock (f _{AD})	Conversion start time (number of PCLKB clock)	
FR[2:0]		Software trigger mode/Hardware trigger no-wait mode	Hardware trigger wait mode
011b	PCLKB/8	7	1
100b	PCLKB/6	5	1
101b	PCLKB/5	4	1
110b	PCLKB/4	3	1
111b	PCLKB/2	1	1

However, for the second and subsequent conversion in sequential conversion mode and for conversion of the channels specified for scan1, 2, and 3 in scan mode, the conversion start time and stabilization wait time for A/D power supply do not occur after a hardware trigger is detected.

- Note: If using the hardware trigger wait mode, setting the ADCS bit to 1 is prohibited (but the bit is automatically switched to 1 when the hardware trigger signal is detected). However, it is possible to clear the ADCS bit to 0 to specify the A/D conversion standby state.
- Note: While in the one-shot conversion mode of the hardware trigger no-wait mode, the ADCS bit is not automatically cleared to 0 when A/D conversion ends. Instead, 1 is retained.
- Note: Only rewrite the value of the ADCE bit when ADCS = 0 (while in the conversion stopped/conversion standby state).
- Note: To complete A/D conversion, specify at least the following time as the hardware trigger interval:
- Hardware trigger no wait mode: 2 PCLKB clock cycles + conversion time
 - Hardware trigger wait mode: 2 PCLKB clock cycles + A/D power supply stabilization wait time + conversion time

Table 22.7 shows the relationship between operation voltage mode and conversion time.

Table 22.7 Conversion time in each operation mode

Operation voltage mode	ADM0.LV[1:0]	Conversion time (number of f _{AD} clock) [cycles]
		Select mode
Normal mode 1	00b	19
Normal mode 2	01b	17
Low voltage mode 1	10b	19
Low voltage mode 2	11b	17

Note: These are the numbers of clock cycles when conversion is with 10-bit resolution. When eight-bit resolution is selected, the values are shorter by two cycles of the conversion clock (f_{AD}).

Table 22.8 shows the A/D conversion time with FR[2:0] and LV[1:0] bits setting.

表 22.6 转换开始时间设置 (2/2)

ADM0	转换时钟 (f _{AD})	转换开始时间 (PCLKB 时钟周期数)	
FR[2:0]		软件触发模式/硬件触发无等待模式	硬件触发等待模式
011b	PCLKB/8	7	1
100b	PCLKB/6	5	1
101b	PCLKB/5	4	1
110b	PCLKB/4	3	1
111b	PCLKB/2	1	1

然而，对于顺序转换模式下的第二次及后续转换，以及扫描模式下为 scan1、2 和 3 指定的通道的转换，在检测到硬件触发后，A/D 电源的转换开始时间和稳定等待时间不会发生。

- 注意: 如果使用硬件触发等待模式，则禁止将 ADCS 位设置为 1（但检测到硬件触发信号时，该位会自动切换为 1）。不过，可以将 ADCS 位清除为 0 以指定 A/D 转换待机状态。
- 注意: 在硬件触发无等待模式的单次转换模式下，A/D 转换结束后，ADCS 位不会自动清零，而是保持为 1。
- 注意: 仅当 ADCS = 0 时（转换停止/转换待机状态）才重写 ADCE 位的值。
- 注意: 要完成 A/D 转换，请至少指定以下时间作为硬件触发间隔:
- 硬件触发无等待模式: 2 个 PCLKB 时钟周期+转换时间
 - 硬件触发等待模式: 2 个 PCLKB 时钟周期+ A/D 电源稳定等待时间+ 转换时间

表 22.7 显示了工作电压模式与转换时间的关系。

表 22.7 各操作模式下的转换时间

工作电压模式	ADM0.LV[1:0]	转换时间 (f _{AD} 时钟周期数) [周期]
		选择模式
普通模式 1	00b	19
普通模式 2	01b	17
低电压模式 1	10b	19
低电压模式 2	11b	17

注意: 这些是 10 位分辨率转换时的时钟周期数。当选择 8 位分辨率时，这些值会缩短两个转换时钟周期(f_{AD})。

表 22.8 显示了 FR[2:0] 和 LV[1:0] 位设置下的 A/D 转换时间。

Table 22.8 A/D Conversion Time Selection 1 of 4

(1) When there is no A/D power supply stabilization wait time
Normal modes 1 and 2 (in software trigger mode and hardware trigger no-wait mode)

A/D Converter Mode Register 0 (ADM0)					Mode	Conversion clock (f _{AD})	Number of Conversion Clock*1	Conversion Time	Conversion Time at 10-Bit Resolution					
FR2	FR1	FR0	LV1	LV0					2.7 V ≤ VDD ≤ 5.5 V					
									PCLKB = 1 MHz	PCLKB = 4 MHz	PCLKB = 8 MHz	PCLKB = 16 MHz	PCLKB = 32 MHz	
0	0	0	0	0	Normal 1	PCLKB/64	19 f _{AD} (number of sampling clock: 7 f _{AD})	1216/PCLKB	Setting prohibited	Setting prohibited	Setting prohibited	76 μs	38 μs	
0	0	1		PCLKB/32		608/PCLKB		76 μs				38 μs	19 μs	
0	1	0		PCLKB/16		304/PCLKB		76 μs				38 μs	19 μs	9.5 μs
0	1	1		PCLKB/8		152/PCLKB		38 μs				19 μs	9.5 μs	4.75 μs
1	0	0		PCLKB/6		114/PCLKB		28.5 μs				14.25 μs	7.125 μs	3.5625 μs
1	0	1		PCLKB/5		95/PCLKB		95 μs	23.75 μs	11.875 μs	5.938 μs	2.9688 μs		
1	1	0		PCLKB/4		76/PCLKB		76 μs	19 μs	9.5 μs	4.75 μs	2.375 μs		
1	1	1		PCLKB/2		38/PCLKB		38 μs	9.5 μs	4.75 μs	2.375 μs	Setting prohibited		
0	0	0	0	1	Normal 2	PCLKB/64	17 f _{AD} (number of sampling clock: 5 f _{AD})	1088/PCLKB	Setting prohibited	Setting prohibited	Setting prohibited	68 μs	34 μs	
0	0	1		PCLKB/32		544/PCLKB		68 μs				34 μs	17 μs	
0	1	0		PCLKB/16		272/PCLKB		68 μs				34 μs	17 μs	8.5 μs
0	1	1		PCLKB/8		136/PCLKB		34 μs				17 μs	8.5 μs	4.25 μs
1	0	0		PCLKB/6		102/PCLKB		25.5 μs				12.75 μs	6.375 μs	3.1875 μs
1	0	1		PCLKB/5		85/PCLKB		85 μs	21.25 μs	10.625 μs	5.3125 μs	2.6563 μs		
1	1	0		PCLKB/4		68/PCLKB		68 μs	17 μs	8.5 μs	4.25 μs	2.125 μs		
1	1	1		PCLKB/2		34/PCLKB		34 μs	8.5 μs	4.25 μs	2.125 μs	Setting prohibited		

Note 1. These are the numbers of clock cycles when conversion is with 10-bit resolution. When eight-bit resolution is selected, the values are shorter by two cycles of the conversion clock (f_{AD}).

Note: The A/D conversion time must also be within the relevant range of conversion times (tCONV) described in .

Note: Rewrite the FR2 to FR0, LV1 and LV0 bits to other than the same data while conversion is stopped (ADCS = 0,ADCE = 0).

Note: The above conversion time does not include conversion start time. Conversion start time add in the first conversion. Select conversion time, taking clock frequency errors into consideration.

表 22.8 A/D 转换时间选择 1/4

(1) 当没有A/D电源稳定等待时间时
普通模式 1 和 2（软件触发模式和硬件触发无等待模式）

A/D转换器模式 寄存器 0 (ADM0)					模式	转换时钟 (f AD)	的数量 转换 n 时钟*1	转换 时间	10 位分辨率下的转换时间							
FR2	FR1	FR0	LV1	LV0					2.7 V ≤ VDD ≤ 5.5 V							
									PCLKB = 1 MHz	PCLKB = 4 MHz	PCLKB = 8 MHz	PCLKB = 16 MHz	PCLKB = 32 MHz			
0	0	0	0	0	普通的 1	PCLKB/64	19 fAD (采样次数) 时钟: 7 fAD)	1216/PCLKB	设置禁止 d	设置禁止 d	设置禁止 d	76 μs	38 μs			
0	0	1				PCLKB/32		608/PCLKB					76 μs	38 μs	19 μs	
0	1	0				PCLKB/16		304/PCLKB					76 μs	38 μs	19 μs	9.5 μs
0	1	1				PCLKB/8		152/PCLKB					38 μs	19 μs	9.5 μs	4.75 μs
1	0	0				PCLKB/6		114/PCLKB					28.5 μs	14.25 μs	7.125 μs	3.5625 μs
1	0	1				PCLKB/5		95/PCLKB	95 μs	23.75 μs	11.875 μs	5.938 μs	2.9688 μs			
1	1	0				PCLKB/4		76/PCLKB	76 μs	19 μs	9.5 μs	4.75 μs	2.375 μs			
1	1	1				PCLKB/2		38/PCLKB	38 μs	9.5 μs	4.75 μs	2.375 μs	设置禁止 d			
0	0	0	0	1	普通的 2	PCLKB/64	17 fAD (采样次数) 时钟: 5 fAD)	1088/PCLKB	设置禁止 d	设置禁止 d	设置禁止 d	68 μs	34 μs			
0	0	1				PCLKB/32		544/PCLKB					68 μs	34 μs	17 μs	
0	1	0				PCLKB/16		272/PCLKB					68 μs	34 μs	17 μs	8.5 μs
0	1	1				PCLKB/8		136/PCLKB					34 μs	17 μs	8.5 μs	4.25 μs
1	0	0				PCLKB/6		102/PCLKB					25.5 μs	12.75 μs	6.375 μs	3.1875 μs
1	0	1				PCLKB/5		85/PCLKB	85 μs	21.25 μs	10.625 μs	5.3125 μs	2.6563 μs			
1	1	0				PCLKB/4		68/PCLKB	68 μs	17 μs	8.5 μs	4.25 μs	2.125 μs			
1	1	1				PCLKB/2		34/PCLKB	34 μs	8.5 μs	4.25 μs	2.125 μs	设置禁止 d			

注 1. 这些是 10 位分辨率转换时的时钟周期数。当选择 8 位分辨率时，这些值会缩短两个转换时钟周期(f_{AD})。

注意: A/D 转换时间也必须在相关转换时间 (tCONV) 范围内。

注意: 在转换停止时，将 FR2 到 FR0、LV1 和 LV0 位重写为与相同数据不同的值(ADCS =0,ADCE = 0)。

注: 以上转换时间不包含转换开始时间。转换开始时间计入首次转换时间。
选择转换时间时，需考虑时钟频率误差。

Table 22.9 A/D Conversion Time Selection 2 of 4

(2) When there is no A/D power supply stabilization wait time
Low-voltage modes 1 and 2 (in software trigger mode and hardware trigger no-wait mode)

A/D Converter Mode Register 0 (ADM0)									Conversion Time at 10-Bit Resolution								
FR2	FR1	FR0	LV1	LV0					Mode	Conversion clock (f _{AD})	Number of Conversion Clock*1	Conver sion Time	1.6 V ≤ VDD ≤ 5.5 V		1.8 V ≤ VDD ≤ 5.5 V	2.4 V ≤ VDD ≤ 5.5 V	2.7 V ≤ VDD ≤ 5.5 V
													PCLKB = 1 MHz	PCLKB = 4 MHz	PCLKB = 8 MHz	PCLKB = 16 MHz	PCLKB = 32 MHz
0	0	0	1	0	Low voltage 1	PCLKB/64	19 f _{AD} (number of sampling clock: 7 f _{AD})	1216/ PCLKB	Setting prohibit ed	Setting prohibit ed	Setting prohibit ed	76 μs	38 μs				
0	0	1		PCLKB/32		608/ PCLKB		76 μs				38 μs	19 μs				
0	1	0		PCLKB/16		304/ PCLKB		76 μs				38 μs	19 μs	9.5 μs			
0	1	1		PCLKB/8		152/ PCLKB		38 μs				19 μs	9.5 μs	4.75 μs			
1	0	0		PCLKB/6		114/ PCLKB		28.5 μs				14.25 μs	7.125 μs	3.5625 μs			
1	0	1		PCLKB/5		95/ PCLKB		95 μs				23.75 μs	11.875 μs	5.938 μs	2.9688 μs		
1	1	0		PCLKB/4		76/ PCLKB		76 μs				19 μs	9.5 μs	4.75 μs	2.375 μs		
1	1	1		PCLKB/2		38/ PCLKB		38 μs				9.5 μs	4.75 μs	2.375 μs	Setting prohibit ed		
0	0	0	1	1	Low voltage 2	PCLKB/64	17 f _{AD} (number of sampling clock: 5 f _{AD})	1088/ PCLKB	Setting prohibit ed	Setting prohibit ed	Setting prohibit ed	68 μs	34 μs				
0	0	1		PCLKB/32		544/ PCLKB		68 μs				34 μs	17 μs				
0	1	0		PCLKB/16		272/ PCLKB		68 μs				34 μs	17 μs	8.5 μs			
0	1	1		PCLKB/8		136/ PCLKB		34 μs				17 μs	8.5 μs	4.25 μs			
1	0	0		PCLKB/6		102/ PCLKB		25.5 μs				12.75 μs	6.375 μs	3.1875 μs			
1	0	1		PCLKB/5		85/ PCLKB		85 μs				21.25 μs	10.625 μs	5.3125 μs	2.6563 μs		
1	1	0		PCLKB/4		68/ PCLKB		68 μs				17 μs	8.5 μs	4.25 μs	2.125 μs		
1	1	1		PCLKB/2		34/ PCLKB		34 μs				8.5 μs	4.25 μs	2.125 μs	Setting prohibit ed		

Note 1. These are the numbers of clock cycles when conversion is with 10-bit resolution. When eight-bit resolution is selected, the values are shorter by two cycles of the conversion clock (f_{AD}).

Note: The A/D conversion time must also be within the relevant range of conversion times (tCONV) described in .

Note: Rewrite the FR2 to FR0, LV1 and LV0 bits to other than the same data while conversion is stopped (ADCS = 0,ADCE = 0).

Note: The above conversion time does not include conversion start time. Conversion start time add in the first conversion. Select conversion time, taking clock frequency errors into consideration.

表 22.9 A/D 转换时间选择 2/4

(2) 当没有A/D电源稳定等待时间时
低电压模式 1 和 2（软件触发模式和硬件触发无等待模式）

A/D转换器模式寄存器0 (ADM0)					模式	转换时钟 (fAD)	的数量 转换 时钟*1	转换时 间	10 位分辨率下的转换时间							
FR2	FR1	FR0	LV1	LV0					1.6 V ≤ VDD ≤ 5.5 V		1.8 V ≤ VDD ≤ 5.5 V	2.4 V ≤ VDD ≤ 5.5 V	2.7 V ≤ VDD ≤ 5.5 V			
									PCLKB = 1 MHz	PCLKB = 4 MHz	PCLKB = 8 MHz	PCLKB = 16 MHz	PCLKB = 32 MHz			
0	0	0	1	0	低电压 1	PCLKB/64	19 fAD (采样时钟数量: 7 fAD)	1216/ PCLKB	设置禁止编辑	设置禁止编辑	设置禁止编辑	76 μs	38 μs			
0	0	1				PCLKB/32		608/ PCLKB				76 μs	38 μs	19 μs		
0	1	0				PCLKB/16		304/ PCLKB				76 μs	38 μs	19 μs	9.5 μs	
0	1	1				PCLKB/8		152/ PCLKB				38 μs	19 μs	9.5 μs	4.75 μs	
1	0	0				PCLKB/6		114/ PCLKB				28.5 μs	14.25 μs	7.125 μs	3.5625 μs	
1	0	1				PCLKB/5		95/ PCLKB				95 μs	23.75 μs	11.875 μs	5.938 μs	2.9688 μs
1	1	0				PCLKB/4		76/ PCLKB				76 μs	19 μs	9.5 μs	4.75 μs	2.375 μs
1	1	1				PCLKB/2		38/ PCLKB				38 μs	9.5 μs	4.75 μs	2.375 μs	设置禁止编辑
0	0	0	1	1	低电压 2	PCLKB/64	17 fAD (采样次数) 时钟: 5 fAD)	1088/ PCLKB	设置禁止编辑	设置禁止编辑	设置禁止编辑	68 μs	34 μs			
0	0	1				PCLKB/32		544/ PCLKB				68 μs	34 μs	17 μs		
0	1	0				PCLKB/16		272/ PCLKB				68 μs	34 μs	17 μs	8.5 μs	
0	1	1				PCLKB/8		136/ PCLKB				34 μs	17 μs	8.5 μs	4.25 μs	
1	0	0				PCLKB/6		102/ PCLKB				25.5 μs	12.75 μs	6.375 μs	3.1875 μs	
1	0	1				PCLKB/5		85/ PCLKB				85 μs	21.25 μs	10.625 μs	5.3125 μs	2.6563 μs
1	1	0				PCLKB/4		68/ PCLKB				68 μs	17 μs	8.5 μs	4.25 μs	2.125 μs
1	1	1				PCLKB/2		34/ PCLKB				34 μs	8.5 μs	4.25 μs	2.125 μs	设置禁止编辑

注 1. 这些是 10 位分辨率转换时的时钟周期数。当选择 8 位分辨率时，这些值会缩短两个转换时钟周期(f_{AD})。

注意: A/D 转换时间也必须在相关转换时间 (tCONV) 范围内。

注意: 在转换停止时，将 FR2 到 FR0、LV1 和 LV0 位重写为与相同数据不同的值(ADCS =0,ADCE = 0)。

注: 以上转换时间不包含转换开始时间。转换开始时间计入首次转换时间。选择转换时间时，需考虑时钟频率误差。

Table 22.10 A/D Conversion Time Selection 3 of 4

(3) When there is no A/D power supply stabilization wait time
Normal modes 1 and 2 (in hardware trigger wait mode)

A/D Converter Mode Register 0 (ADM0)					Mode	Conversion clock (f _{AD})	Number of Stabilization Wait Clock	Number of Conversion Clock *1	Conversion Time	Stabilization Wait Time + Conversion Time at 10-Bit Resolution							
FR2	FR1	FR0	LV1	LV0						2.7 V ≤ VDD ≤ 5.5 V							
										PCLK B = 1 MHz	PCLK B = 4 MHz	PCLK B = 8 MHz	PCLK B = 16 MHz	PCLK B = 32 MHz			
0	0	0	0	0	Normal 1	PCLKB/64	8 f _{AD}	19 f _{AD} (number of sampling clock: 7 f _{AD})	1728/ PCLK B	Setting prohibited	Setting prohibited	Setting prohibited	108 μs	54 μs			
0	0	1				PCLKB/32			864/ PCLK B				108 μs	54 μs	27 μs		
0	1	0				PCLKB/16			432/ PCLK B				108 μs	54 μs	27 μs	13.5 μs	
0	1	1				PCLKB/8			216/ PCLK B				54 μs	27 μs	13.5 μs	6.75 μs	
1	0	0				PCLKB/6			162/ PCLK B				40.5 μs	20.25 μs	10.125 μs	5.0625 μs	
1	0	1				PCLKB/5			135/ PCLK B				135 μs	33.75 μs	16.875 μs	8.4375 μs	4.21875 μs
1	1	0				PCLKB/4			108/ PCLK B				108 μs	27 μs	13.5 μs	6.75 μs	3.375 μs
1	1	1				PCLKB/2			54/ PCLK B				54 μs	13.5 μs	6.75 μs	3.375 μs	Setting prohibited
0	0	0	0	1	Normal 2	PCLKB/64	8 f _{AD}	17 f _{AD} (number of sampling clock: 5 f _{AD})	1600/ PCLK B	Setting prohibited	Setting prohibited	Setting prohibited	100 μs	50 μs			
0	0	1				PCLKB/32			800/ PCLK B				100 μs	50 μs	25 μs		
0	1	0				PCLKB/16			400/ PCLK B				100 μs	50 μs	25 μs	12.5 μs	
0	1	1				PCLKB/8			200/ PCLK B				50 μs	25 μs	12.5 μs	6.25 μs	
1	0	0				PCLKB/6			150/ PCLK B				37.5 μs	18.75 μs	9.375 μs	4.6875 μs	
1	0	1				PCLKB/5			125/ PCLK B				125 μs	31.25 μs	15.625 μs	7.8125 μs	3.90625 μs
1	1	0				PCLKB/4			100/ PCLK B				100 μs	25 μs	12.5 μs	6.25 μs	3.125 μs
1	1	1				PCLKB/2			50/ PCLK B				50 μs	12.5 μs	6.25 μs	3.125 μs	Setting prohibited

表 22.10 A/D 转换时间选择 3/4

(3) 当没有A/D电源稳定等待时间时
普通模式 1 和 2（硬件触发等待模式）

A/D转换器模式寄存器0 (ADM0)					模式	转换 n 时钟 (fAD)	等待时钟的稳定次数	转换时钟编号 *1	转换时间	稳定等待时间+ 转换 10 位分辨率下的时间							
FR2	FR1	FR0	LV1	LV0						2.7 V ≤ VDD ≤ 5.5 V							
										PCLK B = 1 MHz	PCLK B = 4 MHz	PCLK B = 8 MHz	PCLK B = 16 MHz	PCLK B = 32 MHz			
0	0	0	0	0	普通的 1	PCLKB/64	8 fAD	19 fAD (的数量) 时钟 采样: 7 fAD)	1728/ PCLK B	设置禁止	设置禁止	设置禁止	108 μs	54 μs			
0	0	1				PCLKB/32			864/ PCLK B				108 μs	54 μs	27 μs		
0	1	0				PCLKB/16			432/ PCLK B				108 μs	54 μs	27 μs	13.5 μs	
0	1	1				PCLKB/8			216/ PCLK B				54 μs	27 μs	13.5 μs	6.75 μs	
1	0	0				PCLKB/6			162/ PCLK B				40.5 μs	20.25 μs	10.125 μs	5.0625 μs	
1	0	1				PCLKB/5			135/ PCLK B				135 μs	33.75 μs	16.875 μs	8.4375 μs	4.21875 μs
1	1	0				PCLKB/4			108/ PCLK B				108 μs	27 μs	13.5 μs	6.75 μs	3.375 μs
1	1	1				PCLKB/2			54/ PCLK B				54 μs	13.5 μs	6.75 μs	3.375 μs	设置禁止
0	0	0	0	1	普通的 2	PCLKB/64	8 fAD	17 fAD (的数量) 时钟 示例: 5 fAD)	1600/ PCLK B	设置禁止	设置禁止	设置禁止	100 μs	50 μs			
0	0	1				PCLKB/32			800/ PCLK B				100 μs	50 μs	25 μs		
0	1	0				PCLKB/16			400/ PCLK B				100 μs	50 μs	25 μs	12.5 μs	
0	1	1				PCLKB/8			200/ PCLK B				50 μs	25 μs	12.5 μs	6.25 μs	
1	0	0				PCLKB/6			150/ PCLK B				37.5 μs	18.75 μs	9.375 μs	4.6875 μs	
1	0	1				PCLKB/5			125/ PCLK B				125 μs	31.25 μs	15.625 μs	7.8125 μs	3.90625 μs
1	1	0				PCLKB/4			100/ PCLK B				100 μs	25 μs	12.5 μs	6.25 μs	3.125 μs
1	1	1				PCLKB/2			50/ PCLK B				50 μs	12.5 μs	6.25 μs	3.125 μs	设置禁止

Note 1. These are the numbers of clock cycles when conversion is with 10-bit resolution. When eight-bit resolution is selected, the values are shorter by two cycles of the conversion clock (f_{AD}).

Note: The A/D conversion time must also be within the relevant range of conversion times (t_{CONV}) described in .

Note: Rewrite the FR2 to FR0, LV1 and LV0 bits to other than the same data while conversion is stopped ($ADCS = 0, ADCE = 0$).

Note: The above conversion time does not include conversion start time. Conversion start time add in the first conversion. Select conversion time, taking clock frequency errors into consideration.

Note: When hardware trigger wait mode, specify the conversion time, including the stabilization wait time from the hardware trigger detection.

Table 22.11 A/D Conversion Time Selection 4 of 4 (1 of 2)

(4) When there is no A/D power supply stabilization wait time
Low-voltage modes 1 and 2 (in hardware trigger wait mode)

A/D Converter Mode Register 0 (ADM0)					Mode	Conversion clock (f _{AD})	Number of Stabilization Wait Clock	Number of Conversion Clock *1	Conversion Time	Stabilization Wait Time + Conversion Time at 10-Bit Resolution							
FR2	FR1	FR0	LV1	LV0						1.6 V ≤ VDD ≤ 5.5 V		1.8 V ≤ VDD ≤ 5.5 V	2.4 V ≤ VDD ≤ 5.5 V	2.7 V ≤ VDD ≤ 5.5 V			
										PCLK B = 1 MHz	PCLK B = 4 MHz	PCLK B = 8 MHz	PCLK B = 16 MHz	PCLK B = 32 MHz			
0	0	0	1	0	Low voltage 1	PCLKB/64	2 f _{AD}	19 f _{AD} (number of sampling clock: 7 f _{AD})	1344/ PCLK B	Setting prohibited	Setting prohibited	Setting prohibited	84 μs	42 μs			
0	0	1		PCLKB/32		672/ PCLK B							84 μs	42 μs	21 μs		
0	1	0		PCLKB/16		336/ PCLK B							84 μs	42 μs	21 μs	10.5 μs	
0	1	1		PCLKB/8		168/ PCLK B							42 μs	21 μs	10.5 μs	5.25 μs	
1	0	0		PCLKB/6		126/ PCLK B							31.5 μs	15.75 μs	7.875 μs	3.9375 μs	
1	0	1		PCLKB/5		105/ PCLK B							105 μs	26.25 μs	13.125 μs	6.5625 μs	3.238125 μs
1	1	0		PCLKB/4		84/ PCLK B							84 μs	21 μs	10.5 μs	5.25 μs	2.625 μs
1	1	1		PCLKB/2		42/ PCLK B							42 μs	10.5 μs	5.25 μs	2.625 μs	Setting prohibited

注1: 这些是采用10位分辨率进行转换时的时钟周期数。当选择8位分辨率时, 这些值会缩短两个转换时钟周期 (f_{AD})。

注意: A/D 转换时间也必须在相关转换时间 (t_{CONV}) 范围内。

注意: 在转换停止时, 将 FR2 到 FR0、LV1 和 LV0 位重写为与相同数据不同的值($ADCS = 0, ADCE = 0$)。

注: 以上转换时间不包含转换开始时间。转换开始时间计入首次转换时间。选择转换时间时, 需考虑时钟频率误差。

注意: 在硬件触发等待模式下, 指定转换时间, 包括从硬件触发检测到稳定等待的时间。

表 22.11 A/D 转换时间选择 4/4 (1/2)

(4) 当没有A/D电源稳定等待时间时
Low-voltage modes 1 and 2 (in hardware trigger wait mode)

A/D转换器模式寄存器0 (ADM0)					模式	转换 n 时钟 (fAD)	等待时钟的稳定次数	转换时钟编号 *1	转换时间	稳定等待时间+ 转换 10 位分辨率下的时间						
FR2	FR1	FR0	LV1	LV0						1.6 V ≤ VDD ≤ 5.5 V		1.8 V ≤ VDD ≤ 5.5 V	2.4 V ≤ VDD ≤ 5.5 V	2.7 V ≤ VDD ≤ 5.5 V		
										PCLK B = 1 MHz	PCLK B = 4 MHz	PCLK B = 8 MHz	PCLK B = 16 MHz	PCLK B = 32 MHz		
0	0	0	1	0	低电压 1	PCLKB/64	2 fAD	19 fAD (的数量) 时钟样本: 7 f)	1344/ PCLK B	设置禁止	设置禁止	设置禁止	84 μs	42 μs		
0	0	1		PCLKB/32		84 μs							42 μs	21 μs		
0	1	0		PCLKB/16		84 μs							42 μs	21 μs	10.5 μs	
0	1	1		PCLKB/8		42 μs							21 μs	10.5 μs	5.25 μs	
1	0	0		PCLKB/6		31.5 μs							15.75 μs	7.875 μs	3.9375 μs	
1	0	1		PCLKB/5		105 μs							26.25 μs	13.125 μs	6.5625 μs	3.2381 25 μs
1	1	0		PCLKB/4		84 μs							21 μs	10.5 μs	5.25 μs	2.625 μs
1	1	1		PCLKB/2		42 μs							10.5 μs	5.25 μs	2.625 μs	设置禁止

Table 22.11 A/D Conversion Time Selection 4 of 4 (2 of 2)

(4) When there is no A/D power supply stabilization wait time
Low-voltage modes 1 and 2 (in hardware trigger wait mode)

A/D Converter Mode Register 0 (ADM0)					Mode	Conversion clock (f _{AD})	Number of Stabilization Wait Clock	Number of Conversion Clock *1	Conversion Time	Stabilization Wait Time + Conversion Time at 10-Bit Resolution							
FR2	FR1	FR0	LV1	LV0						1.6 V ≤ VDD ≤ 5.5 V		1.8 V ≤ VDD ≤ 5.5 V	2.4 V ≤ VDD ≤ 5.5 V	2.7 V ≤ VDD ≤ 5.5 V			
										PCLK B = 1 MHz	PCLK B = 4 MHz	PCLK B = 8 MHz	PCLK B = 16 MHz	PCLK B = 32 MHz			
0	0	0	1	1	Low voltage 2	PCLKB/64	2 f _{AD}	17 f _{AD} (number of sampling clock: 5 f _{AD})	1216/ PCLK B	Setting prohibited	Setting prohibited	Setting prohibited	76 μs	38 μs			
0	0	1				PCLKB/32			608/ PCLK B				76 μs	38 μs	19 μs		
0	1	0				PCLKB/16			304/ PCLK B				76 μs	38 μs	19 μs	9.5 μs	
0	1	1				PCLKB/8			152/ PCLK B				38 μs	19 μs	9.5 μs	4.75 μs	
1	0	0				PCLKB/6			114/ PCLK B				28.5 μs	14.25 μs	7.125 μs	3.5625 μs	
1	0	1				PCLKB/5			95/ PCLK B				96 μs	23.75 μs	11.88 μs	5.938 μs	2.9688 μs
1	1	0				PCLKB/4			76/ PCLK B				76 μs	19 μs	9.5 μs	4.75 μs	2.375 μs
1	1	1				PCLKB/2			38/ PCLK B				38 μs	9.5 μs	4.75 μs	2.375 μs	Setting prohibited

Note 1. These are the numbers of clock cycles when conversion is with 10-bit resolution. When eight-bit resolution is selected, the values are shorter by two cycles of the conversion clock (f_{AD}).

- Note: The A/D conversion time must also be within the relevant range of conversion times (tCONV) described in .
- Note: Rewrite the FR2 to FR0, LV1 and LV0 bits to other than the same data while conversion is stopped (ADCS = 0,ADCE = 0).
- Note: The above conversion time does not include conversion start time. Conversion start time add in the first conversion. Select conversion time, taking clock frequency errors into consideration.
- Note: When hardware trigger wait mode, specify the conversion time, including the stabilization wait time from the hardware trigger detection.

Figure 22.3 shows A/D Converter Sampling and A/D Conversion Timing (Example for Software Trigger Mode).

表 22.11 A/D 转换时间选择 4/4 (2/2)

(4) 当没有A/D电源稳定等待时间时
低电压模式 1 和 2（硬件触发等待模式）

A/D转换器模式寄存器0 (ADM0)					模式	转换 n 时钟 (f _{AD})	等待时钟的稳定次数	转换时钟编号 *1	转换时间	稳定等待时间+ 转换 10 位分辨率下的时间						
FR2	FR1	FR0	LV1	LV0						1.6 V ≤ VDD ≤ 5.5 V		1.8 V ≤ VDD ≤ 5.5 V	2.4 V ≤ VDD ≤ 5.5 V	2.7 V ≤ VDD ≤ 5.5 V		
										PCLK B = 1 MHz	PCLK B = 4 MHz	PCLK B = 8 MHz	PCLK B = 16 MHz	PCLK B = 32 MHz		
0	0	0	1	1	低电压 2	PCLKB/64	2 f _{AD}	17 f _{AD} (的数量) 时钟示例: 5 f _{AD})	1216/ PCLK B	设置禁止	设置禁止	设置禁止	76 μs	38 μs		
0	0	1		PCLKB/32		608/ PCLK B							76 μs	38 μs	19 μs	
0	1	0		PCLKB/16		304/ PCLK B							76 μs	38 μs	19 μs	9.5 μs
0	1	1		PCLKB/8		152/ PCLK B							38 μs	19 μs	9.5 μs	4.75 μs
1	0	0		PCLKB/6		114/ PCLK B							28.5 μs	14.25 μs	7.125 μs	3.5625 μs
1	0	1		PCLKB/5		95/ PCLK B			96 μs				23.75 μs	11.88 μs	5.938 μs	2.9688 μs
1	1	0		PCLKB/4		76/ PCLK B			76 μs				19 μs	9.5 μs	4.75 μs	2.375 μs
1	1	1		PCLKB/2		38/ PCLK B			38 μs				9.5 μs	4.75 μs	2.375 μs	设置禁止

注 1. 这些是 10 位分辨率转换时的时钟周期数。当选择 8 位分辨率时，这些值会缩短两个转换时钟周期(f_{AD})。

- 注意: A/D 转换时间也必须在相关转换时间 (tCONV) 范围内。
- 注意: 在转换停止时，将 FR2 到 FR0、LV1 和 LV0 位重写为与相同数据不同的值(ADCS =0,ADCE = 0)。
- 注: 以上转换时间不包含转换开始时间。转换开始时间计入首次转换时间。选择转换时间时，需考虑时钟频率误差。
- 注意: 在硬件触发等待模式下，指定转换时间，包括从硬件触发检测到稳定等待的时间。

图 22.3 显示了A/D转换器采样和 A/D 转换时序（软件触发模式示例）。

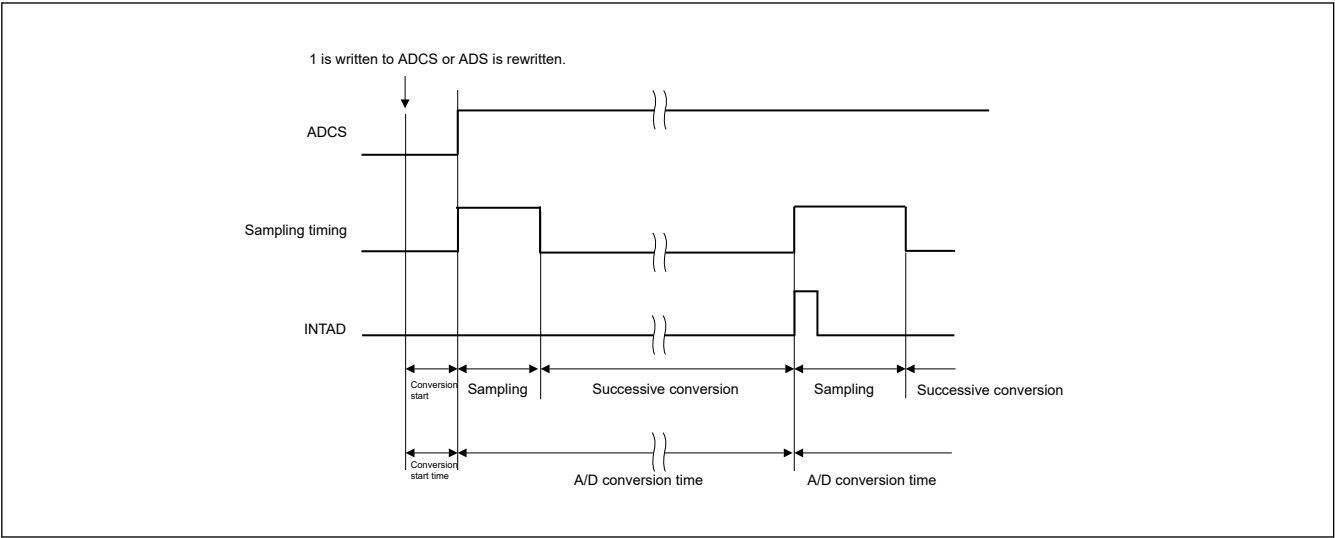


Figure 22.3 A/D Converter Sampling and A/D Conversion Timing (Example for Software Trigger Mode)

22.2.2 ADM1 : A/D Converter Mode Register 1

Base address: ADC_D = 0x400A_1800
Offset address: 0x0002

Bit position:	7	6	5	4	3	2	1	0
Bit field:	ADTMD[1:0]		ADSC M	—	—	—	ADTRS[1:0]	
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
1:0	ADTRS[1:0]	Selection of the Hardware Trigger Signal 0 0: Timer Array Unit channel 1 count or capture end interrupt signal (TAU0_TMI01) 1 1: 32-bit interval timer interrupt signal (ADITL0 (= TML32_ITL0)) Others: Setting prohibited.	R/W
4:2	—	These bits are read as 0. The write value should be 0.	R/W
5	ADSCM	Specification of the A/D Conversion Mode 0: Sequential conversion mode 1: One-shot conversion mode	R/W
7:6	ADTMD[1:0]	Selection of the A/D Conversion Trigger Mode 1 0: Hardware trigger no-wait mode 1 1: Hardware trigger wait mode 0 x: Software trigger mode	R/W

This register is used to specify the A/D conversion trigger, conversion mode, and hardware trigger signal.

Note: Only rewrite the value of the ADM1 register while conversion operation is stopped (ADCS = 0, ADCE = 0).

Note: To complete A/D conversion, specify at least the following time as the hardware trigger interval:
Hardware trigger no wait mode: 2 PCLKB clock cycles + conversion start time + A/D conversion time
Hardware trigger wait mode: 2 PCLKB clock cycles + conversion start time + A/D power supply stabilization wait time + A/D conversion time

Note: In modes other than Snooze mode, input of the next ADITL0 (= TML32_ITL0) is not recognized as a valid hardware trigger for up to 4 PCLKB cycles after the first ADITL0 (= TML32_ITL0) is input.

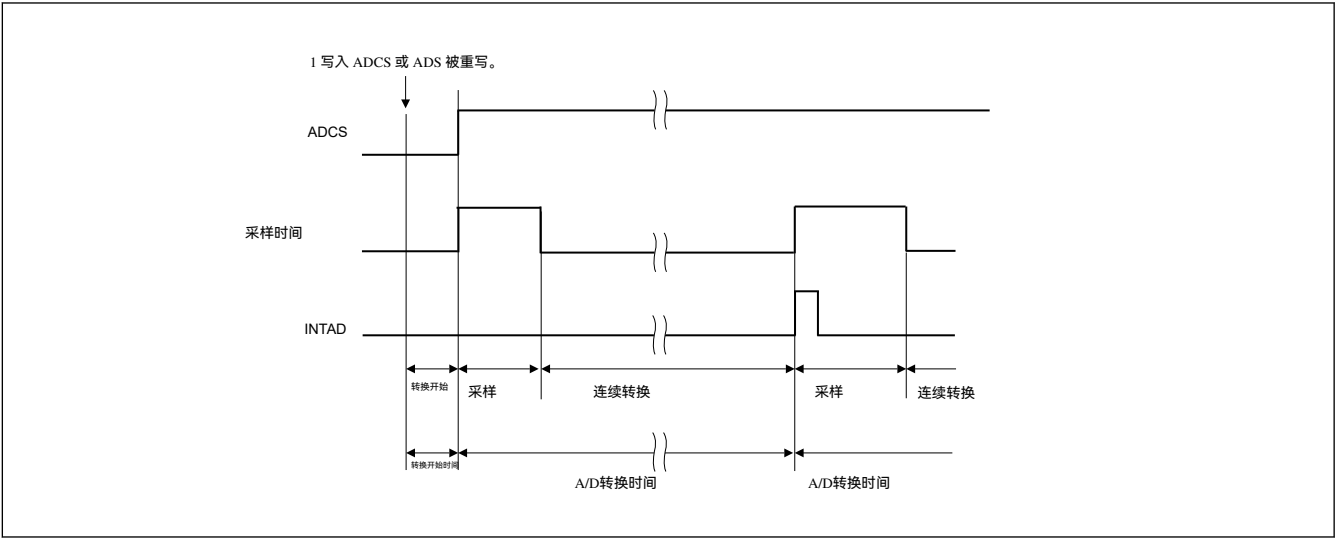


图 22.3 A/D 转换器采样和 A/D 转换时序（软件触发模式示例）

22.2.2ADM1: A/D转换器模式寄存器1

基地址: ADC_D = 0x400A_1800
偏移地址: 0x0002

位位置:	7	6	5	4	3	2	1	0
位域:	ADTMD[1:0]		ADSC M	—	—	—	ADTRS[1:0]	
重置后的值:	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
1:0	ADTRS[1:0]	硬件触发信号的选择0 0:定时器阵列单元通道 1 计数或捕获结束中断信号 (TAU0_TMI01) 1 1: 32 位间隔定时器中断信号 (ADITL0(=TML32_ITL0)) 其他: 禁止设置。	R/W
4:2	—	这些位读取为 0。写入值应为 0。	R/W
5	ADSCM	A/D转换模式的规范 0: 顺序转换模式 1: 单次转换模式	R/W
7:6	ADTMD[1:0]	选择A/D转换触发模式 1 0:硬件触发无等待模式 1 1:硬件触发等待模式 0 x: 软件触发模式	R/W

该寄存器用于指定A/D转换触发、转换模式和硬件触发信号。

注意: 仅在转换操作停止时重写 ADM1 寄存器的值(ADCS = 0, ADCE = 0)。

注意: 要完成 A/D 转换, 请至少指定以下时间作为硬件触发间隔:
硬件触发无等待模式: 2 个 PCLKB 时钟周期+转换开始时间+ A/D 转换时间
硬件触发等待模式: 2 个 PCLKB 时钟周期+转换开始时间+ A/D 电源稳定等待时间+ A/D 转换时间

注意: 在除贪睡模式之外的其他模式下, 在输入第一个 ADITL0(= TML32_ITL0) 后的最多 4 个 PCLKB 周期内, 下一个 ADITL0(= TML32_ITL0) 的输入不会被识别为有效的硬件触发信号。

22.2.3 ADM2 : A/D Converter Mode Register 2

Base address:	ADC_D = 0x400A_1800						
Offset address:	0x0110						
Bit position:	7	6	5	4	3	2	1 0
Bit field:	ADREFP[1:0]	ADRE FM	—	ADRC K	AWC	—	ADTY P
Value after reset:	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
0	ADTYP	Selection of the resolution 0: 10-bit resolution 1: 8-bit resolution Others: Setting prohibited.	R/W
1	—	This bit is read as 0. The write value should be 0.	R/W
2	AWC	Specification of the Snooze Mode 0: Do not use the Snooze mode function. 1: Use the Snooze mode function.	R/W
3	ADRCK	Checking the Upper Limit and Lower Limit Conversion Result Values 0: The interrupt signal (ADC10_ADI) is output when the ADLL register ≤ the ADCR register ≤ the ADUL register (AREA 1). 1: The interrupt signal (ADC10_ADI) is output when the ADCR register < the ADLL register (AREA 2) or the ADUL register < the ADCR register (AREA 3).	R/W
4	—	This bit is read as 0. The write value should be 0.	R/W
5	ADREFM	Selection of the '-' Side Reference Voltage of the A/D Converter 0: Supplied from V _{SS} 1: Supplied from VREFL0/AN001	R/W
7:6	ADREFP[1:0]	Selection of the '+' Side Reference Voltage Source of the A/D Converter 0 0: Supplied from V _{CC} 0 1: Supplied from VREFH0/AN000 1 0: Supplied from the internal reference voltage 1 1: Setting prohibited	R/W

- Note: Only rewrite the value of the ADM2 register while conversion operation is stopped (ADCS = 0, ADCE = 0).
- Note: Do not set the ADREFP[1:0] bits to 10 when shifting to STOP mode while the CPU is operating on the subsystem clock. When the internal reference voltage is selected (ADREFP[1:0] = 10b), the A/D converter reference voltage current (I_{ADREF}) indicated in [section 27.3.2. Operating and Standby Current](#) will be added.
- Note: When using VREFH0 and VREFL0, specify AN000 and AN001 as the analog inputs and set the Pin Mode Control bit (PMC) to 1, the N-Channel Open-Drain Control bit (NCODR) to 0, and the Port Direction bit (PDR) to 0 in the Port mn Pin Function Select Register PmnPFS_A.

ADTYP bit (Selection of the resolution)

These bits are used for selection of the resolution.

AWC bit (Specification of the Snooze Mode)

This bit is used for specification of the Snooze mode.

When there is a hardware trigger signal in the Software Standby mode, the Software Standby mode is exited, and A/D conversion is performed without operating the CPU (the Snooze mode).

- The SNOOZE mode function can only be specified when the high-speed on-chip oscillator clock is selected for the CPU/peripheral hardware clock (PCLKB). If any other clock is selected, specifying this mode is prohibited.
- Using the Snooze mode function in the software trigger mode, software trigger wait mode, or hardware trigger no-wait mode is prohibited.
- Using the SNOOZE mode function in the sequential conversion mode is prohibited.

22.2.3 ADM2: A/D转换器模式寄存器2

基地址:	ADC_D = 0x400A_1800						
偏移地址:	0x0110						
位位置:	7	6	5	4	3	2	1 0
位域:	ADREFP[1:0]	ADRE FM	—	ADRC K	AWC	—	ADTY P
重置后的值:	0	0	0	0	0	0	0

少量	象征	功能	R/W
0	ADTYP	分辨率选择 0: 10 位分辨率 1: 8 位分辨率其他: 禁止设置。	R/W
1	—	该位读取值为 0。写入值应为 0。	R/W
2	AWC	贪睡模式 0 的规格: 请勿使用贪睡模式功能。 1: 使用贪睡模式功能。	R/W
3	ADRCK	检查上限和下限转换结果值 0: 当 ADLL 寄存器≤ADCR 寄存器≤ADUL 寄存器 (区域 1) 时, 输出中断信号 (ADC10_ADI)。 1: 当 ADCR 寄存器<等于 ADLL 寄存器 (区域 2) 或 ADUL 寄存器< 等于 ADCR 寄存器 (区域3) 时, 输出中断信号 (ADC10_ADI)。	R/W
4	—	该位读取值为 0。写入值应为 0。	R/W
5	ADREFM	选择模数转换器的 '-' 侧参考电压 0: 由 V _{SS} 提供 1: 由 VREFL0/AN001 提供	R/W
7:6	ADREFP[1:0]	选择模数转换器的 '+' 侧参考电压源 0 0: 由 V _{CC} 提供 0 1: 由 VREFH0/AN000 提供 1 0: 由内部参考电压供电 1.1: 禁止设置	R/W

- 注意: 仅在转换操作停止时重写 ADM2 寄存器的值(ADCS = 0, ADCE = 0)。
- 注意: 当 CPU 使用子系统时钟运行时, 切换到停止模式时, 请勿将 ADREFP[1:0] 位设置为 10。选择内部参考电压(ADREFP[1:0] = 10b)时, 将把 27.3.2 节中所示的 A/D 转换器参考电压电流 (I_{ADREF}) 相加。工作电流和待机电流将相加。
- 注意: 当使用 VREFH0 和 VREFL0 时, 请将 AN000 和 AN001 指定为模拟输入, 并将端口 mn 引脚功能选择寄存器 PmnPFS_A 中的引脚模式控制位 (PMC) 设置为 1, N 沟道开漏控制位 (NCODR) 设置为 0, 端口方向位 (PDR) 设置为 0。

ADTYP 位 (分辨率选择)

这些位用于选择分辨率。

AWC 位 (睡眠模式规范)

此位用于指定贪睡模式。

当软件待机模式下有硬件触发信号时, 软件待机模式退出, A/D 转换在不运行 CPU 的情况下执行 (休眠模式)。

- 只有当CPU/外设硬件时钟 (PCLKB) 选择高速片上振荡器时钟时, 才能指定睡眠模式功能。如果选择其他时钟, 则禁止指定此模式。
- 禁止在软件触发模式、软件触发等待模式或硬件触发无等待模式下使用贪睡模式功能。
- 禁止在顺序转换模式下使用贪睡模式功能。

- When using the Snooze mode function, specify a hardware trigger interval of at least "shift time to Snooze mode^{*1} + conversion start time + A/D power supply stabilization wait time + A/D conversion time + 2 PCLKB clock cycles".
- Even when using Snooze mode, be sure to set the AWC bit to 0 in normal operation and change it to 1 just before shifting to Software Standby mode.
Also, be sure to change the AWC bit to 0 after returning from Software Standby mode to normal operation.
If the AWC bit is left set to 1, A/D conversion will not start normally in spite of the subsequent Snooze mode or normal operation.

Note 1. Refer to Table 27.20 in section 27.4.2. Wakeup Time.

ADRCCK bit (Checking the Upper Limit and Lower Limit Conversion Result Values)

This bit is used for checking the upper limit and lower limit conversion result values.

Figure 22.4 shows the generation range of the interrupt signal (ADC10_ADI) for AREA 1 to AREA 3.

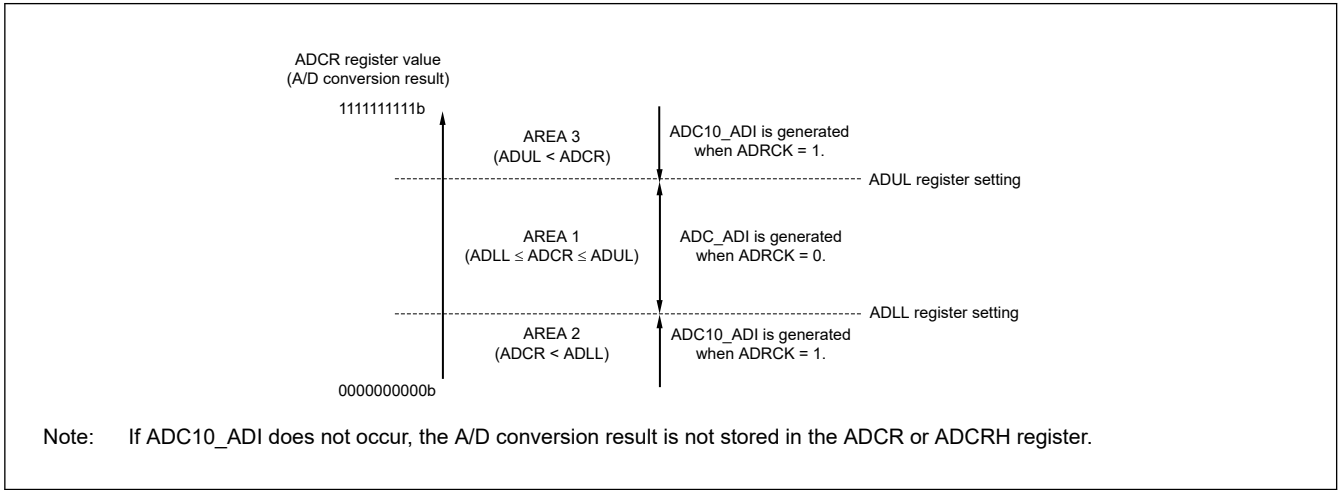


Figure 22.4 ADRCCK bit interrupt signal generation range (in 10-bit resolution mode)

ADREFM bit (Selection of the ‘-’ Side Reference Voltage of the A/D Converter)

This bit is used for selection of the ‘-’ side reference voltage of the 10-bit A/D converter.

ADREFP[1:0] bits (Selection of the ‘+’ Side Reference Voltage Source of the A/D Converter)

These bits are used for selection of the ‘+’ side reference voltage source of the 10-bit A/D converter.

- When ADREFP1 or ADREFP0 bit is rewritten, this must be configured in accordance with the following procedures.
 - (1) Set ADCE = 0
 - (2) Change the values of ADREFP1 and ADREFP0
 - (3) Reference voltage stabilization wait time (A)
 - (4) Set ADCE = 1
 - (5) Reference voltage stabilization wait time (B)
 When ADREFP1 and ADREFP0 are set to 1 and 0, the setting is changed to A = 5 μs, B = 1 μs.
 When ADREFP1 and ADREFP0 are set to 0 and 0 or 0 and 1, A needs no wait and B = 1 μs.
 After (5) stabilization time, start the A/D conversion.
- When ADREFP1 and ADREFP0 are set to 1 and 0, respectively, A/D conversion cannot be performed on the temperature sensor output voltage and internal reference voltage (1.48 V (typ.)).
Be sure to perform A/D conversion while ADISS = 0.

- 使用贪睡模式功能时，请指定至少为 1 小时的硬件触发间隔。
“切换到休眠模式的时间*1+转换开始时间+ A/D 电源稳定等待时间+ A/D 转换时间+ 2PCLKB 时钟周期”。
- 即使在使用睡眠模式时，也务必在正常操作中将 AWC 位设置为 0，并在切换到软件待机模式之前将其更改为 1。

另外，从软件待机模式恢复到正常运行模式后，务必将 AWC 位更改为 0。
如果 AWC 位设置为 1，则即使随后进入休眠模式或正常运行，A/D 转换也不会正常启动。

注1.请参阅第27.4.2节中的表27.20。起床时间。

ADRCCK 位（检查上限和下限转换结果值）

此位用于检查转换结果值的上限和下限。

图 22.4 显示了区域 1 到区域 3 的中断信号 (ADC10_ADI) 的生成范围。

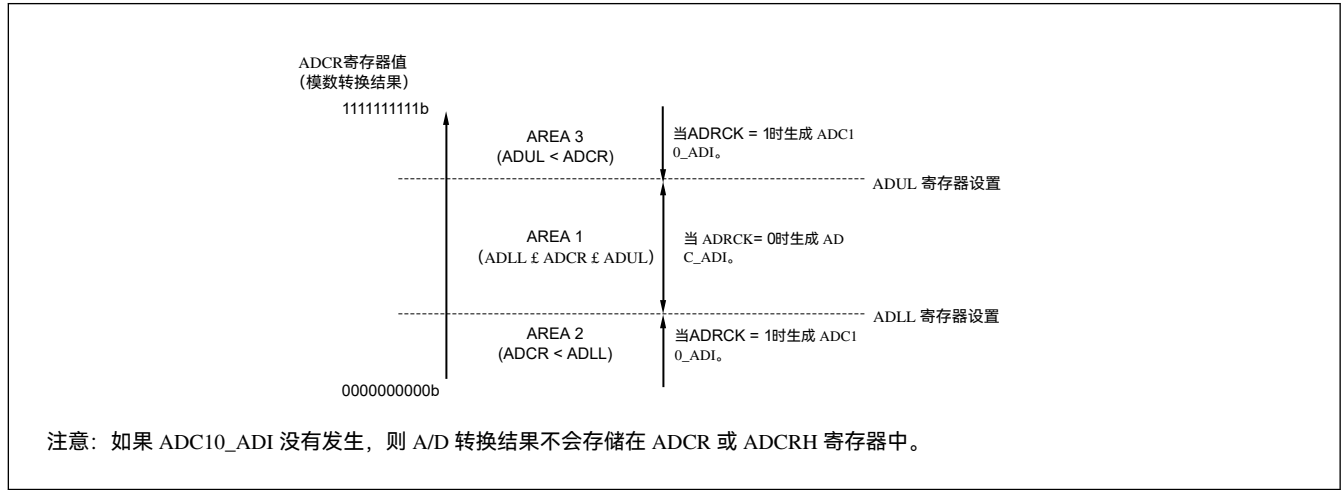


图 22.4 ADRCCK 位中断信号生成范围（10 位分辨率模式）

ADREFM 位（选择 A/D 转换器的‘-’侧参考电压）

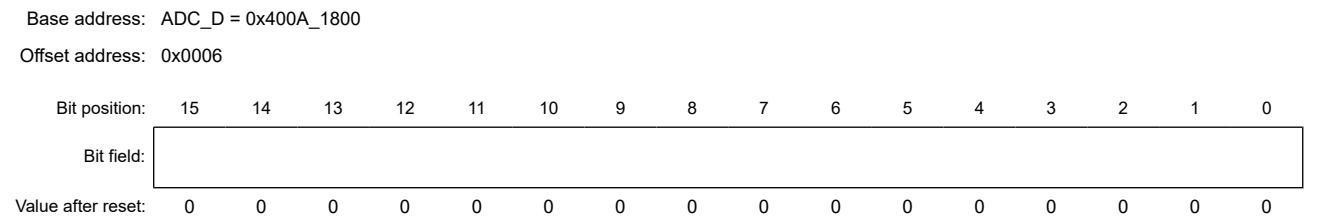
该位用于选择 10 位 A/D 转换器的“-”侧参考电压。

ADREFP[1:0] 位（选择 A/D 转换器的‘+’侧参考电压源）

这些位用于选择 10 位 A/D 转换器的“+”侧参考电压源。

- 当 ADREFP1 或 ADREFP0 位被重写时，必须按照以下步骤进行配置。
 - (1) Set ADCE = 0
 - (2) 更改 ADREFP1 和 ADREFP0 的值
 - (3) 参考电压稳定等待时间 (A)
 - (4) Set ADCE = 1
 - (5) 参考电压稳定等待时间 (B)
 当 ADREFP1 和 ADREFP0 分别设置为 1 和 0 时，设置将更改为 A = 5 μs, B = 1 μs。
 当 ADREFP1 和 ADREFP0 设置为 0 和 0 或 0 和 1 时，A 不需要等待，并且 B = 1 μs。
 经过 (5) 稳定时间后，开始 A/D 转换。
- 当 ADREFP1 和 ADREFP0 分别设置为 1 和 0 时，无法对温度传感器输出电压和内部参考电压(1.48 V (典型值))执行 A/D 转换。
务必执行 A/D 转换，同时 ADISS = 0。

22.2.4 ADCR: 10-bit A/D Conversion Result Register



Bit	Symbol	Function	R/W
15:0	n/a	10-bit Resolution A/D Converter Result	R

Note: When selecting 10-bit mode, the lower 6 bits are fixed at 0. The conversion result is stored in the higher 10 bits of ADCR register.

ADCR is a 16-bit register that holds the A/D conversion result.

Each time A/D conversion ends, the conversion result is loaded from the successive approximation register (SAR).

Note: If the A/D conversion result is outside the range specified by using the A/D conversion comparison function (set up by the ADM2.ADRCK bit, ADUL register, and ADLL register; see [Figure 22.4](#)), the result is not stored.

Note: When 8-bit resolution A/D conversion is selected (when the ADTYP[1:0] bits of A/D converter mode register 2 (ADM2) are respectively set to 01b) and the ADCR register is read, 0 is read from the bits other than the higher 8 bits.

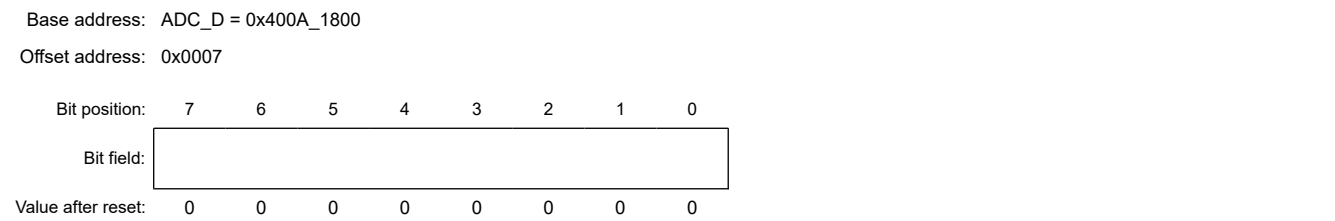
Note: When the ADCR register is accessed in 16-bit units, and A/D conversion with 10-bit resolution is selected, the higher 10 bits of the conversion result are read in order starting at bit 15 of the ADCR register.

Note: The contents of the ADCR register may become undefined when writing to any of the following registers.

- A/D converter mode register 0 (ADM0)
- Analog input channel specification register (ADS)

Read the conversion result following conversion completion before writing to any of these registers. Otherwise, the correct conversion result may not be obtained.

22.2.5 ADCRH : 8-bit A/D Conversion Result Register



Bit	Symbol	Function	R/W
7:0	n/a	8-bit Resolution A/D Converter Result for Channel n	R

ADCRH is an 8-bit register that holds the A/D conversion result. The higher 8 bits of 10-bit resolution are stored^{*1}.

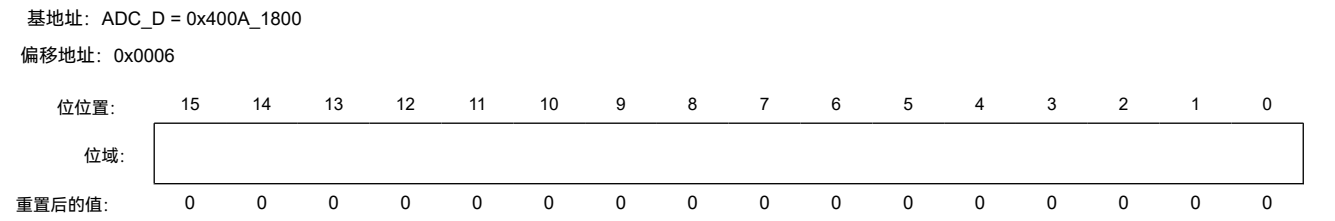
Note 1. If the A/D conversion result is outside the range specified by using the A/D conversion comparison function (setup by the ADRCK bit of the ADM2 register, ADUL register, and ADLL register; see [Figure 22.4](#)), the result is not stored.

Note: The contents of the ADCRH register may become undefined when writing to any of the following registers.

- A/D converter mode register 0 (ADM0)
- Analog input channel specification register (ADS)

Read the conversion result following conversion completion before writing to any of these registers. Otherwise, the correct conversion result may not be obtained.

22.2.4ADCR： 10 位 A/D 转换结果寄存器



少量	象征	功能	R/W
15:0	n/a	10 位分辨率 A/D 转换器结果	R

注意: 选择 10 位模式时，低 6 位固定为 0。转换结果存储在 ADCR 寄存器的高 10 位中。

ADCR 是一个 16 位寄存器，用于保存A/D 转换结果。

每次A/D转换结束时，转换结果都会从逐次逼近寄存器 (SAR) 中加载。

注意: 如果A/D转换结果超出使用 A/D 转换比较功能指定的范围（由 ADM2.ADRCK 位、ADUL 寄存器和 ADLL 寄存器设置；参见图 22.4），则结果不会被存储。

注意: 当选择 8 位分辨率 A/D 转换时（当 A/D 转换器模式寄存器 2 (ADM2) 的 ADTYP[1:0] 位分别设置为 01b），并且读取 ADCR 寄存器时，除高 8 位之外的其他位均读取 0。

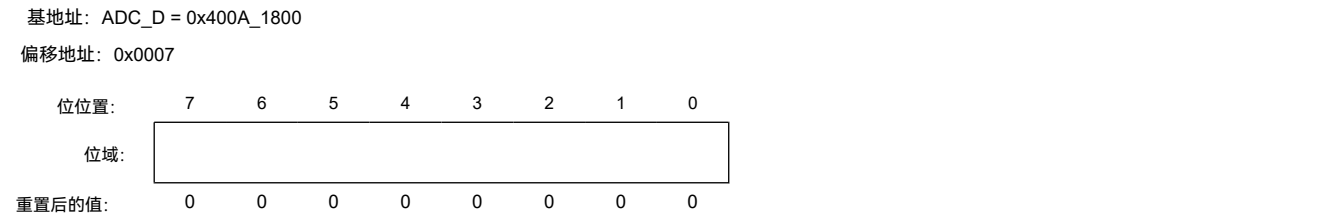
注意: 当以 16 位为单位访问 ADCR 寄存器，并选择 10 位分辨率的 A/D 转换时，转换结果的高 10 位将从 ADCR 寄存器的第 15 位开始按顺序读取。

注意: 向以下任何寄存器写入数据时，ADCR 寄存器的内容可能会变为未定义。

- A/D转换器模式寄存器0（ADM0）
- 模拟输入通道规范寄存器（ADS）

转换完成后，请先读取转换结果，然后再写入任何寄存器。否则，可能无法获得正确的转换结果。

22.2.5ADCRH： 8 位 A/D 转换结果寄存器



少量	象征	功能	R/W
7:0	n/a	通道 n 的 8 位分辨率 A/D 转换器结果	R

ADCRH 是一个 8 位寄存器，用于保存A/D转换结果。10 位分辨率中的高 8 位被存储^{*1}。

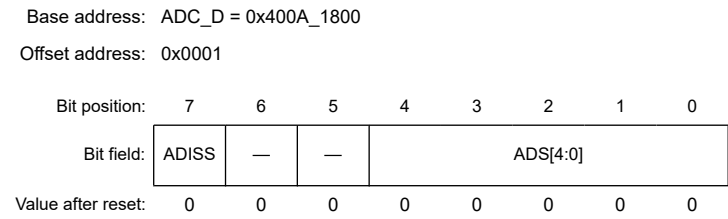
注 1. 如果A/D转换结果超出使用 A/D 转换比较功能指定的范围（由 ADM2 寄存器、ADUL 寄存器和 ADLL 寄存器的 ADRCK 位设置；参见图 22.4），则结果不会被存储。

注意: 向以下任何寄存器写入数据时，ADCRH 寄存器的内容可能会变为未定义。

- A/D转换器模式寄存器0（ADM0）
- 模拟输入通道规范寄存器（ADS）

转换完成后，请先读取转换结果，然后再写入任何寄存器。否则，可能无法获得正确的转换结果。

22.2.6 ADS : Analog Input Channel Specification Register



Bit	Symbol	Function	R/W
4:0	ADS[4:0]	Selection of the Analog Input Channel (See Table 22.12.)	R/W
6:5	—	These bits are read as 0. The write value should be 0.	R/W
7	ADISS	Select Internal or External of Analog Input (See Table 22.12.) 0: External input 1: Internal circuit input	R/W

This register specifies the input channel of the analog voltage to be A/D converted.

[Table 22.12](#) shows the input sources that can be selected for ADS[4:0] bits and ADISS bit in each operating mode.

Table 22.12 Input source selection by ADS[4:0] bits and ADISS bit in select mode

ADISS	ADS[4:0]	Analog input channel	Input source
0	00000b	AN000	P010
0	00001b	AN001	P011
0	00100b	AN004	P012
0	00101b	AN005	P013
0	10101b	AN021	P101
0	10110b	AN022	P100
1	00000b	—	Temperature sensor output voltage
1	00001b	—	Internal reference voltage*1
Other than the above		Setting prohibited	

Note 1. For details about the internal reference voltage, see [section 27, Electrical Characteristics](#).

Note: Rewrite the value of the ADISS bit while conversion is stopped (ADCS = 0, ADCE = 0).

Note: If using VREFH0 as the '+' side reference voltage of the A/D converter, do not select AN000 as an A/D conversion channel.

Note: If using VREFL0 as the '-' side reference voltage of the A/D converter, do not select AN001 as an A/D conversion channel.

Note: When the setting of the ADISS bit is 1, the internal reference voltage cannot be used for the '+' side reference voltage. After the ADISS bit is set to 1, the initial conversion result cannot be used. For the setting flow, see [section 22.6.4. Example of Using the ADC10 when Selecting the Temperature Sensor Output Voltage or Internal Reference Voltage, and Software Trigger Mode and One-shot Conversion Mode](#).

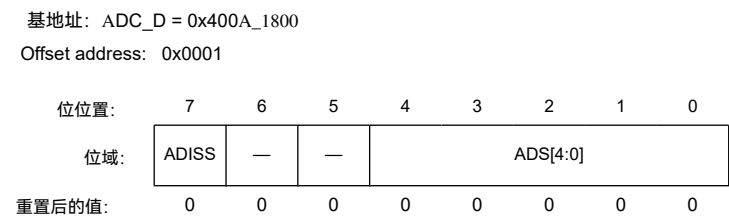
For details about the internal reference voltage, see [section 27, Electrical Characteristics](#).

Note: Do not set the ADISS bit to 1 when shifting to Software Standby mode, or to Sleep mode while the CPU is operating on the subsystem clock. When the ADISS bit is set to 1, the A/D converter reference voltage current (IADREF) indicated in [section 27.3.2. Operating and Standby Current](#) will be added.

Note: When the setting of the ADISS bit is 1, the hardware trigger wait mode and one-shot conversion mode cannot be used at the same time.

Note: When the setting of the ADISS bit is 1, the software trigger wait mode and one-shot conversion mode cannot be used at the same time.

22.2.6 ADS: 模拟输入通道规范寄存器



少量	象征	功能	R/W
4:0	ADS[4:0]	模拟输入通道的选择（参见表 22.12。）	R/W
6:5	—	这些位读取为 0。写入值应为 0。	R/W
7	ADISS	选择模拟输入的内部或外部（参见表 22.12）。 0: 外部输入 1: 内部电路输入	R/W

该寄存器指定要转换的模拟电压的输入通道A/D。

表 22.12 显示了在每种操作模式下，ADS[4:0] 位和 ADISS 位可以选择的输入源。

表 22.12 输入源选择通过 ADS[4:0] 位和 ADISS 位在选择模式下进行

ADISS	ADS[4:0]	模拟输入通道	输入源
0	00000b	AN000	P010
0	00001b	AN001	P011
0	00100b	AN004	P012
0	00101b	AN005	P013
0	10101b	AN021	P101
0	10110b	AN022	P100
1	00000b	—	温度传感器输出电压
1	00001b	—	内部参考电压*1
除上述内容外		设置禁止	

注 1. 有关内部参考电压的详细信息，请参阅第 27 节“电气特性”。

注意：在转换停止时重写 ADISS 位的值(ADCS = 0, ADCE = 0)。

注意：如果使用 VREFH0 作为A/D转换器的‘+’侧参考电压，则不要选择 AN000 作为A/D 转换通道。

注意：如果使用 VREFL0 作为 A/D 转换器的‘-’ 侧参考电压，则不要选择 AN001 作为 A/D 转换通道。

注意：当 ADISS 位设置为 1 时，内部参考电压不能用作‘+’侧的参考电压。ADISS 位设置为 1 后，初始转换结果将无法使用。有关设置流程，请参见 22.6.4 节“选择温度传感器输出电压或内部参考电压、软件触发模式和单次转换模式下使用 ADC10 的示例”。

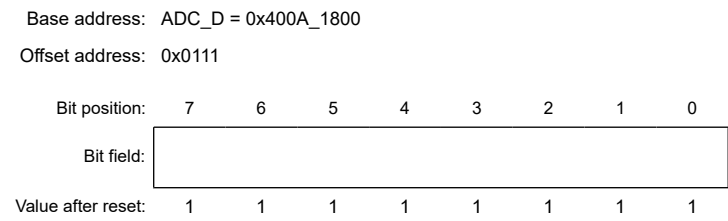
有关内部参考电压的详细信息，请参阅第 27 节“电气特性”。

注意：切换到软件待机模式或 CPU 使用子系统时钟运行时切换到睡眠模式时，请勿将 ADISS 位设置为 1。如果将 ADISS 位设置为 1，则 27.3.2 节中所示的 A/D 转换器参考电压电流 (IADREF) 将与工作电流和待机电流相加。

注意：当 ADISS 位设置为 1 时，硬件触发等待模式和单次转换模式不能同时使用。

注意：当 ADISS 位设置为 1 时，软件触发等待模式和单次转换模式不能同时使用。

22.2.7 ADUL : Conversion Result Comparison Upper Limit Setting Register

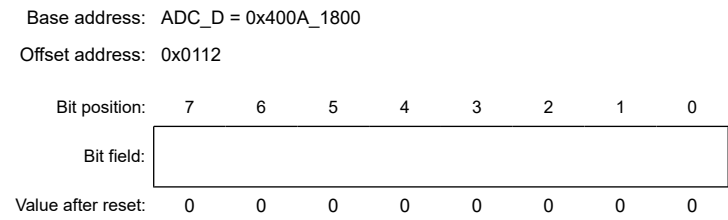


Bit	Symbol	Function	R/W
7:0	n/a	Setting the Upper Limit for A/D Conversion Results	R/W

This register is used to specify the setting for checking the upper limit of the A/D conversion results.

The A/D conversion results and ADUL register value are compared, and interrupt signal (ADC10_ADI) generation is controlled in the range specified by the ADRCK bit of A/D converter mode register 2 (ADM2) (shown in [Figure 22.4](#)).

22.2.8 ADLL : Conversion Result Comparison Lower Limit Setting Register



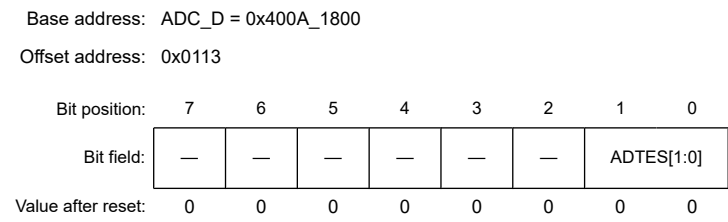
Bit	Symbol	Function	R/W
7:0	n/a	Setting the Lower Limit for A/D Conversion Results	R/W

This register is used to specify the setting for checking the lower limit of the A/D conversion results.

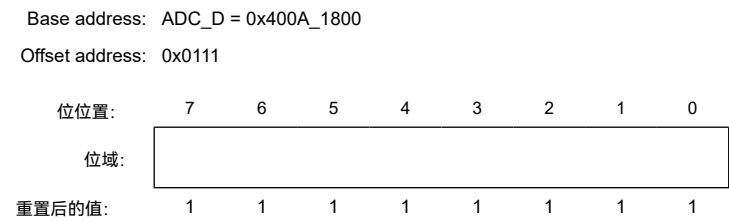
The A/D conversion results and ADLL register value are compared, and interrupt signal (ADC10_ADI) generation is controlled in the range specified by the ADRCK bit of A/D converter mode register 2 (ADM2) (shown in [Figure 22.4](#)).

- Note: When A/D conversion with 10-bit resolution is selected, the A/D conversion result register ADCR[15:8] value is compared with the values in the ADUL and ADLL registers.
- Note: Only write new values to the ADUL and ADLL registers while conversion is stopped (ADCS = 0, ADCE = 0).
- Note: The setting of the ADUL register must be greater than that of the ADLL register.

22.2.9 ADTES : A/D Test Register



22.2.7ADUL：转换结果比较上限设置寄存器



少量	象征	功能	R/W
7:0	n/a	设置模数转换结果的上限	R/W

该寄存器用于指定检查A/D转换结果上限的设置。

将A/D转换结果与 ADUL 寄存器值进行比较，并在A/D转换器模式寄存器 2 (ADM2) 的 ADRCK 位指定的范围内控制中断信号 (ADC10_ADI) 的生成（如图 22.4 所示）。

22.2.8ADLL：转换结果比较下限设置寄存器



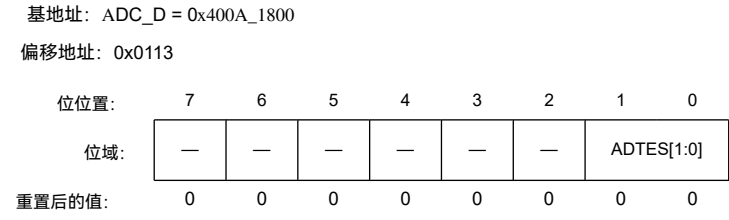
少量	象征	功能	R/W
7:0	n/a	设置模数转换结果的下限	R/W

该寄存器用于指定检查A/D转换结果下限的设置。

将A/D转换结果与 ADLL 寄存器值进行比较，并在A/D转换器模式寄存器 2 (ADM2) 的 ADRCK 位指定的范围内控制中断信号 (ADC10_ADI) 的生成（如图 22.4 所示）。

- 注意: 当选择 10 位分辨率的 A/D 转换时，A/D 转换结果寄存器 ADCR[15:8] 的值与 ADUL 和 ADLL 寄存器中的值进行比较。
- 注意: 仅在转换停止时才向 ADUL 和 ADLL 寄存器写入新值(ADCS = 0,ADCE = 0)。
- 注意: ADUL 寄存器的值必须大于 ADLL 寄存器的值。

22.2.9ADTES：A/D 测试寄存器



Bit	Symbol	Function	R/W
1:0	ADTES[1:0]	Selection of A/D Conversion Target for Testing 0 0: ANxxx, temperature sensor output voltage or internal reference voltage*1 (Set by analog input channel specification register (ADS)) 1 0: The '-' side reference voltage (selected by the ADREFM bit of the ADM2 register) 1 1: The '+' side reference voltage (selected by the ADREFP[1:0] bits of the ADM2 register) Others: Setting prohibited.	R/W
7:2	—	These bits are read as 0. The write value should be 0.	R/W

Note 1. For details about the internal reference voltage, see [section 27, Electrical Characteristics](#).

This register is used to select the ‘+’ side reference voltage or ‘-’ side reference voltage for the converter, an analog input channel (ANxxx), the temperature sensor output voltage, or the internal reference voltage*1 as the target for A/D conversion.

When using this register to test the converter, set as follows.

- For zero-scale measurement, select the ‘-’ side reference voltage as the target for conversion.
- For full-scale measurement, select the ‘+’ side reference voltage as the target for conversion.

22.3 A/D Converter Operations

The A/D converter conversion operations are described below.

<1> The voltage input to the selected analog input channel is sampled by the sample & hold circuit.

<2> When sampling has been done for a certain time, the sample & hold circuit is placed in the hold state and the sampled voltage is held until the A/D conversion operation has ended.

<3> Bit 9 of the successive approximation register (SAR) is set to 1. The series resistor string voltage tap is set to 1/2 AVREF by the tap selector.

<4> The voltage difference between the series resistor string voltage tap and sampled voltage is compared by the voltage comparator. If the analog input is greater than 1/2 AVREF, the MSB of the SAR register remains set to 1. If the analog input is smaller than 1/2 AVREF, the MSB is reset to 0.

<5> Next, bit 8 of the SAR register is automatically set to 1, and the operation proceeds to the next comparison. The series resistor string voltage tap is selected according to the preset value of bit 9, as described below.

- Bit 9 = 1: (3/4) AVREF
- Bit 9 = 0: (1/4) AVREF

The voltage tap and sampled voltage are compared and bit 8 of the SAR register is manipulated as follows.

- Sampled voltage ≥ Voltage tap: Bit 8 = 1
- Sampled voltage < Voltage tap: Bit 8 = 0

<6> Comparison is continued in this way up to bit 0 of the SAR register.

<7> Upon completion of the comparison of 10 bits, an effective digital result value remains in the SAR register, and the result value is transferred to the A/D conversion result register (ADCR, ADCRH) and then latched*1.

At the same time, the A/D conversion end interrupt request signal (ADC10_ADI) can also be generated*1.

<8> Repeat steps <1> to <7>, until the ADCS bit is cleared to 0*2.

To stop the A/D converter, clear the ADCS bit to 0.

Note 1. If the A/D conversion result is outside the A/D conversion result range specified by the ADRCK bit and the ADUL and ADLL registers (see [Figure 22.4](#)), the A/D conversion end interrupt request signal is not generated and no A/D conversion results are stored in the ADCR and ADCRH registers.

Note 2. While in the sequential conversion mode, the ADCS flag is not automatically cleared to 0. This flag is not automatically cleared to 0 while in the one-shot conversion mode of the hardware trigger no-wait mode, either. Instead, 1 is retained.

Note: Two types of the A/D conversion result registers are available.

- ADCR register (16 bits): Store 10-bit A/D conversion value

少量	象征	功能	R/W
1:0	ADTES[1:0]	测试用A/D转换目标的选择0 0:ANxxx、温度传感器输出电压或内部参考电压*1（由模拟输入通道规范寄存器（ADS）设置）1 0:‘-’侧参考电压（由ADM2寄存器的ADREFM位选择）‘+’侧参考电压（由ADM2寄存器的ADREFP位选择）其他：禁止设置。	R/W
7:2	—	这些位读取为 0。写入值应为 0。	R/W

注 1. 有关内部参考电压的详细信息，请参阅第 27 节“电气特性”。

该寄存器用于选择转换器的‘+’ 侧参考电压或‘-’侧参考电压、模拟输入通道(ANxxx)、温度传感器输出电压或内部参考电压*1作为A/D 转换的目标。

使用此寄存器测试转换器时，请按如下方式设置。

- 对于零标度测量，选择‘-’侧参考电压作为转换目标。
- 对于全量程测量，选择‘+’侧参考电压作为转换目标。

22.3A/D转换器操作

下面描述了A/D转换器转换操作。

<1>采样保持电路对选定的模拟输入通道的电压输入进行采样。

<2>当采样持续一定时间后，采样保持电路&进入保持状态，采样电压保持到转换操作结束为止。

<3>逐次逼近寄存器 (SAR) 的第 9 位设置为 1。串联电阻串电压抽头通过抽头选择器设置为 1/2AVREF 。

<4>串联电阻串电压抽头与采样电压之间的电压差由电压比较器进行比较。如果模拟输入大于1/2 AVREF，则 SAR 寄存器的最高有效位 (MSB) 保持为 1。如果模拟输入小于1/2 AVREF，则 MSB 复位为 0。

当<5> Next时，SAR 寄存器的第 8 位自动设置为 1，并进行下一次比较。串联电阻串电压抽头的选择根据第 9 位的预设值进行，如下所述。

- Bit 9 = 1: (3/4) AVREF
- Bit 9 = 0: (1/4) AVREF

将电压抽头和采样电压进行比较，并对 SAR 寄存器的第 8 位进行如下操作。

- 采样电压≥电压抽头：位8 = 1
- 采样电压< 电压抽头：位8 = 0

<6>以这种方式继续比较，直到 SAR 寄存器的第 0 位。

<7>10 位比较完成后，有效数字结果值保留在 SAR 寄存器中，并将结果值传输到A/D转换结果寄存器 (ADCR, ADCRH) 并锁存*1。

同时，还可以生成A/D转换结束中断请求信号（ADC10_ADI）*1。

<8>重复步骤<1>至<7>，直到 ADCS 位被清除为0*2。

要停止A/D转换器，请将 ADCS 位清除为 0。

注 1. 如果 A/D 转换结果超出 ADRCK 位和 ADUL 和 ADLL 寄存器指定的 A/D 转换结果范围（参见图 22.4），则不会生成 A/D 转换结束中断请求信号，并且不会在 ADCR 和 ADCRH 寄存器中存储任何 A/D 转换结果。

注 2. 在顺序转换模式下，ADCS 标志不会自动清除为 0。在硬件触发无等待模式的单次转换模式下，该标志也不会自动清除为 0。

相反，保留 1。

注意：有两种类型的 A/D 转换结果寄存器可用。

- ADCR寄存器（16位）：存储10位A/D转换值

- ADCRH register (8 bits): Store 8-bit A/D conversion value

Note: AV_{REF} : The '+' side reference voltage of the A/D converter. This can be selected from VREFH0, the internal reference voltage, and V_{CC} .

For details about the internal reference voltage, see [section 27, Electrical Characteristics](#).

Note: $n = 0$ to 3

Figure 22.5 shows the conversion operation of the A/D converter during software trigger mode.

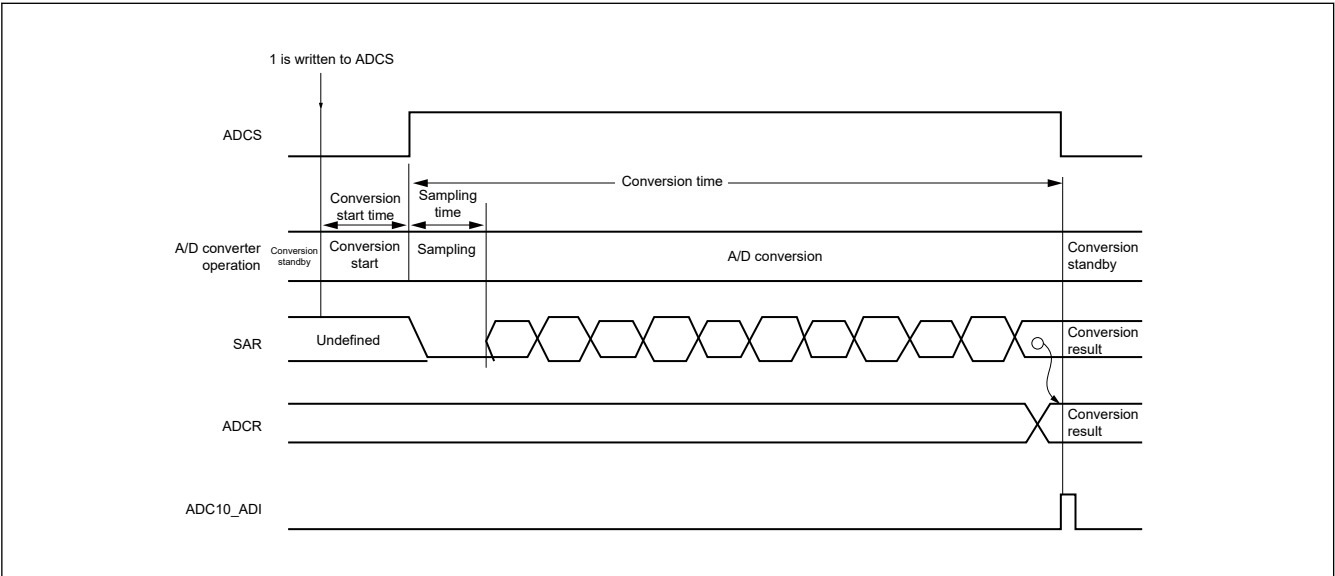


Figure 22.5 Conversion operation of A/D converter (software trigger mode)

In one-shot conversion mode, the ADSC bit is automatically cleared to 0 after completion of A/D conversion.

In sequential conversion mode, A/D conversion operations proceed continuously until the software clears bit 7 (ADSC) of the A/D converter mode register 0 (ADM0) to 0.

Writing to the analog input channel specification register (ADS) during A/D conversion interrupts the current conversion after which A/D conversion of the analog input specified by the ADS register proceeds. Data from the A/D conversion that was in progress are discarded.

The value of the A/D conversion result register (ADCR, ADCRH) is 0x00 or 0x0000 following a reset.

22.4 Input Voltage and Conversion Results

The relationship between the analog voltage input to the analog input pins (AN000, AN001, AN004, AN005, AN021 and AN022) and the theoretical A/D conversion result (stored in the 10-bit A/D conversion result register (ADCR)) is shown by the following expression.

$$SAR = \text{INT}\left(\frac{V_{AIN}}{AV_{REF}} \times 1024 + 0.5\right)$$

$$ADCR = SAR \times 64$$

or

$$(ADCR/64 - 0.5) \times \frac{AV_{REF}}{1024} \leq V_{AIN} < (ADCR/64 + 0.5) \times \frac{AV_{REF}}{1024}$$

where,

INT(): Function which returns integer part of value in parentheses

V_{AIN} : Analog input voltage

AV_{REF} : AV_{REF} pin voltage

ADCR: 10-bit A/D conversion result register (ADCR) value

- ADCRH寄存器（8位）：存储8位A/D转换值

注: AV_{REF} : 是 A/D 转换器的 '+' 侧参考电压。该电压可从内部参考电压 VREFH0 和 V_{CC} 中选择。

有关内部参考电压的详细信息，请参阅第 27 节“电气特性”。

注意: $n = 0$ 至 3

图 22.5 显示了 A/D 转换器在软件触发模式下的转换操作。

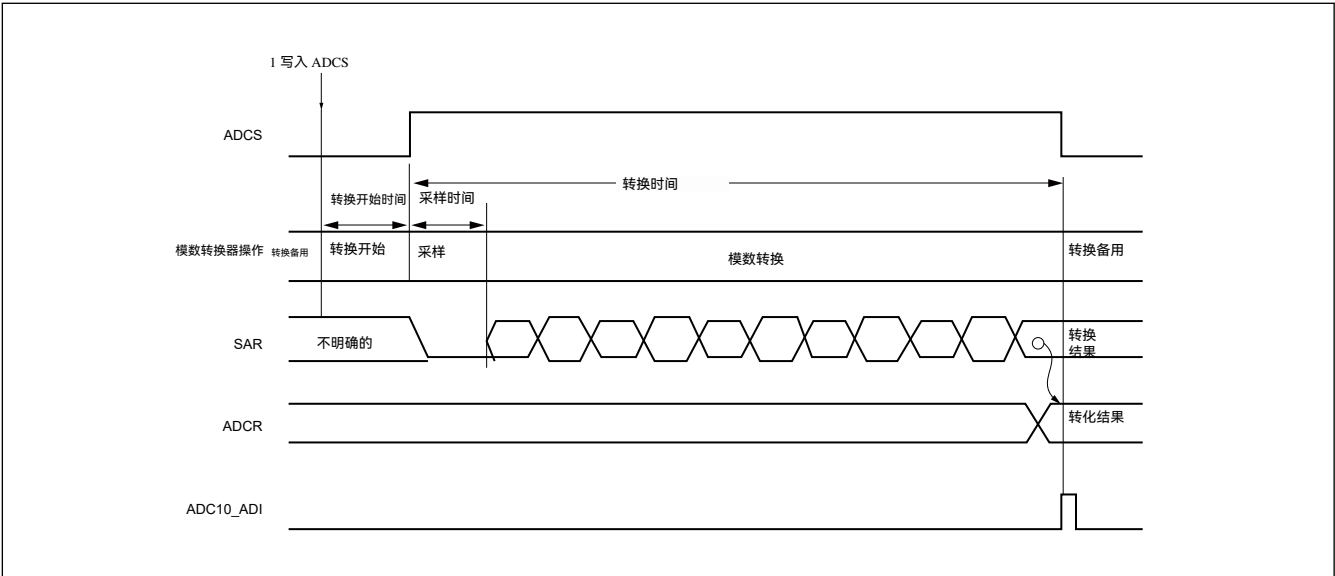


图 22.5 A/D 转换器的转换操作（软件触发模式）

在单次转换模式下，完成 A/D 转换后，ADSC 位会自动清除为 0。

在顺序转换模式下，A/D 转换操作会持续进行，直到软件将转换器模式寄存器 0 (ADM0) 的第 7 位 (ADSC) 清除为 0。

在 A/D 转换期间写入模拟输入通道规范寄存器 (ADS) 会中断当前转换，之后会继续进行由 ADS 寄存器指定的模拟输入的 A/D 转换。正在进行的 A/D 转换的数据将被丢弃。

复位后，转换结果寄存器 (ADCR、ADCRH) 的值将变为 0x00 或 0x0000。

22.4 输入电压和转换结果

模拟输入引脚 (AN000、AN001、AN004、AN005、AN021 和 AN022) 的模拟电压输入与理论转换结果（存储在 10 位 A/D 转换结果寄存器 (ADCR) 中）之间的关系由以下表达式表示。

$$SAR = \text{INT}\left(\frac{V_{AIN}}{AV_{REF}} \times 1024 + 0.5\right)$$

$$ADCR = SAR \times 64$$

or

$$(ADCR/64 - 0.5) \times \frac{AV_{REF}}{1024} \leq V_{AIN} < (ADCR/64 + 0.5) \times \frac{AV_{REF}}{1024}$$

在哪里，

INT(): 返回括号中值的整数部分的函数

V_{AIN} : 模拟输入电压

AV_{REF} : AV_{REF} 引脚电压

ADCR: 10 位 A/D 转换结果寄存器 (ADCR) 值

SAR: Successive approximation register

Figure 22.6 shows relationship between analog input voltage and A/D conversion result.

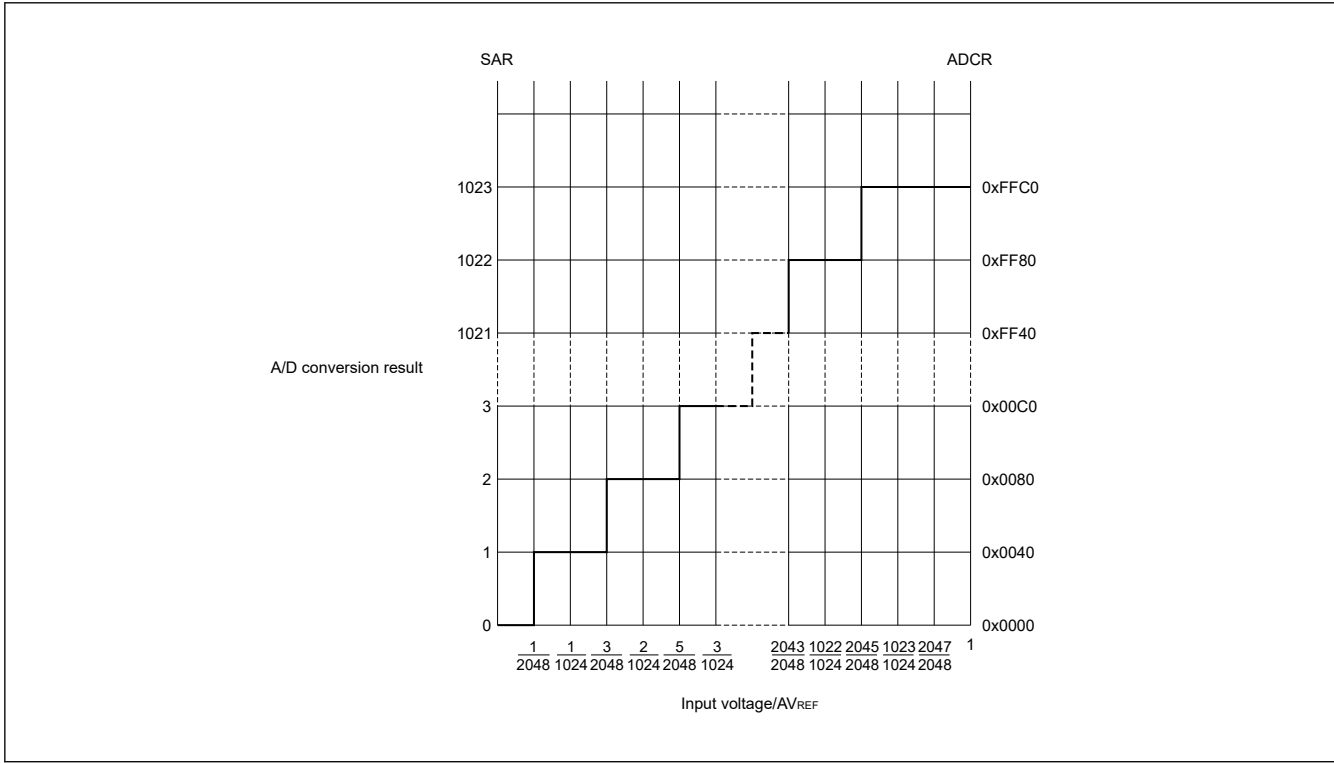


Figure 22.6 Relationship between analog input voltage and A/D conversion result

V_{REF} : The ‘+’ side reference voltage of the A/D converter. This can be selected from VREFH0 the internal reference voltage^{*1}, and V_{CC} .

Note 1. For details about the internal reference voltage, see section 27, Electrical Characteristics.

22.5 A/D Converter Operation Modes

The operation of each A/D converter mode is described below. In addition, the procedure for specifying each mode is described in section 22.6. A/D Converter Setup Procedure.

22.5.1 Software Trigger Mode (Select Mode, Sequential Conversion Mode)

- <1> In the stop state, the ADCE bit of A/D converter mode register 0 (ADM0) is set to 1, and the A/D converter enters the standby state.
- <2> After the software counts up to the stabilization wait time (1 μ s), the ADCS bit of the ADM0 register is set to 1 to perform the A/D conversion of the analog input specified by the analog input channel specification register (ADS).
- <3> When A/D conversion ends, the conversion result is stored in the A/D conversion result registers (ADCR, ADCRH), and the A/D conversion end interrupt request signal (ADC10_ADI) is generated. After A/D conversion ends, the next A/D conversion immediately starts.
- <4> When ADCS is overwritten with 1 during conversion operation, the current A/D conversion is interrupted, and conversion restarts. The partially converted data is discarded.
- <5> When the value of the ADS register is rewritten or overwritten during conversion operation, the current A/D conversion is interrupted, and A/D conversion is performed on the analog input respecified by the ADS register. The partially converted data is discarded.
- <6> Even if a hardware trigger is input during conversion operation, A/D conversion does not start.
- <7> When ADCS is cleared to 0 during conversion operation, the current A/D conversion is interrupted, and the A/D converter enters the standby state.

SAR：逐次逼近寄存器

图 22.6 显示了模拟输入电压与 A/D 转换结果之间的关系。

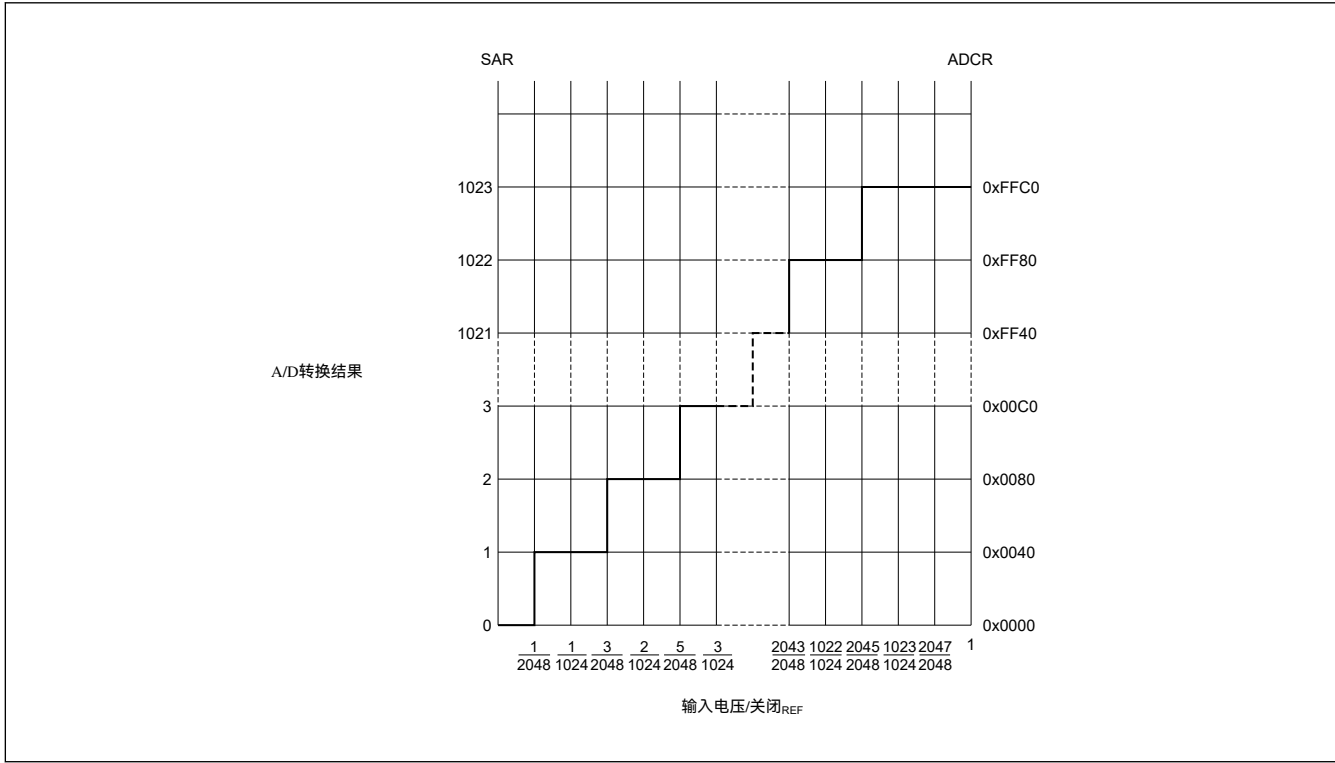


图 22.6 模拟输入电压与 A/D 转换结果的关系

V_{REF} : ‘+’ 转换器的参考电压。它可以从内部参考电压 $V_{REFH0} \times 1$ 和 V_{CC} 中选择。

注 1. 有关内部参考电压的详细信息，请参阅第 27 节“电气特性”。

22.5 A/D 转换器工作模式

下面描述了每个 A/D 转换器模式的操作。此外，第 22.6 节描述了指定每种模式的步骤。A/D 转换器设置步骤。

22.5.1 软件触发模式（选择模式、顺序转换模式）

- <1>在停止状态下，A/D 转换器模式寄存器 0 (ADM0) 的 ADCE 位设置为 1，A/D 转换器进入待机状态。
- <2>软件计数到稳定等待时间后(1 μ s)ADM0 寄存器的 ADCS 位设置为 1，以执行模拟输入通道规范寄存器 (ADS) 指定的模拟输入的 A/D 转换。
当模数转换结束时，转换结果存储在转换结果寄存器 (ADCR、ADCRH) 中，并生成转换结束中断请求信号 (ADC10_ADI)。模数转换结束后，立即开始下一次模数转换。
- <4>当转换操作期间 ADCS 被覆盖为 1 时，当前的 A/D 转换将被中断，转换将重新开始。部分转换的数据将被丢弃。
当转换操作期间 ADS 寄存器的值被重写或覆盖时，当前的转换将被中断，并对 ADS 寄存器重新指定的模拟输入执行转换。部分转换的数据将被丢弃。
- <6>即使在转换操作期间输入硬件触发信号，A/D 转换也不会开始。
- <7>当转换操作期间 ADCS 被清除为 0 时，当前的 A/D 转换中断，A/D 转换器进入待机状态。

<8> When ADCE is cleared to 0 while in the A/D conversion standby state, the A/D converter enters the stop state. When ADCE = 0, specifying 1 for ADCS is ignored and A/D conversion does not start.

Figure 22.7 shows the example of software trigger mode (select mode, sequential conversion mode) operation timing.

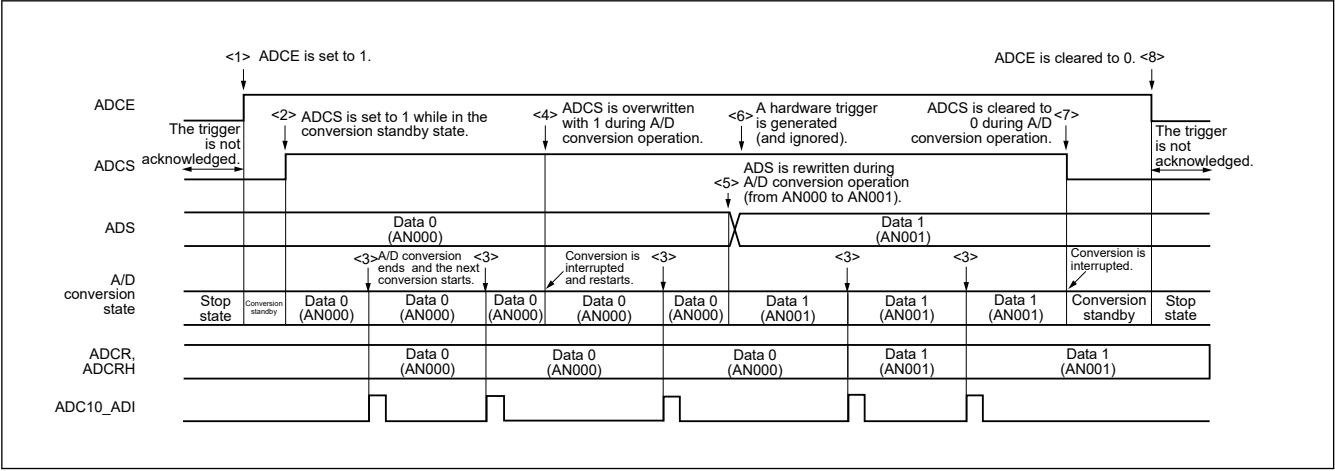


Figure 22.7 Example of software trigger no-wait mode (select mode, sequential conversion mode) operation timing

22.5.2 Software Trigger Mode (Select Mode, One-shot Conversion Mode)

- <1> In the stop state, the ADCE bit of A/D converter mode register 0 (ADM0) is set to 1, and the A/D converter enters the standby state.
- <2> After the software counts up to the stabilization wait time (1 μ s), the ADCS bit of the ADM0 register is set to 1 to perform the A/D conversion of the analog input specified by the analog input channel specification register (ADS).
- <3> When A/D conversion ends, the conversion result is stored in the A/D conversion result registers (ADCR, ADCRH), and the A/D conversion end interrupt request signal (ADC10_ADI) is generated.
- <4> After A/D conversion ends, the ADCS bit is automatically cleared to 0, and the A/D converter enters the standby state.
- <5> When ADCS is overwritten with 1 during conversion operation, the current A/D conversion is interrupted, and conversion restarts. The partially converted data is discarded.
- <6> When the value of the ADS register is rewritten or overwritten during conversion operation, the current A/D conversion is interrupted, and A/D conversion is performed on the analog input respecified by the ADS register. The partially converted data is discarded.
- <7> When ADCS is cleared to 0 during conversion operation, the current A/D conversion is interrupted, and the A/D converter enters the standby state.
- <8> When ADCE is cleared to 0 while in the A/D conversion standby state, the A/D converter enters the stop state. In addition, A/D conversion does not start even if a hardware trigger is input while in the A/D conversion standby state. When ADCE = 0, specifying 1 for ADCS is ignored and A/D conversion does not start.

Figure 22.8 shows the example of software select no-wait mode (select mode, one-shot conversion mode) operation timing.

<8>当 ADCE 在 A/D 转换待机状态下被清除为 0 时，A/D 转换器进入停止状态。ADCE = 0 指定 ADCS 为 1 时，转换不会启动。

图 22.7 显示了软件触发模式（选择模式、顺序转换模式）操作时序的示例。

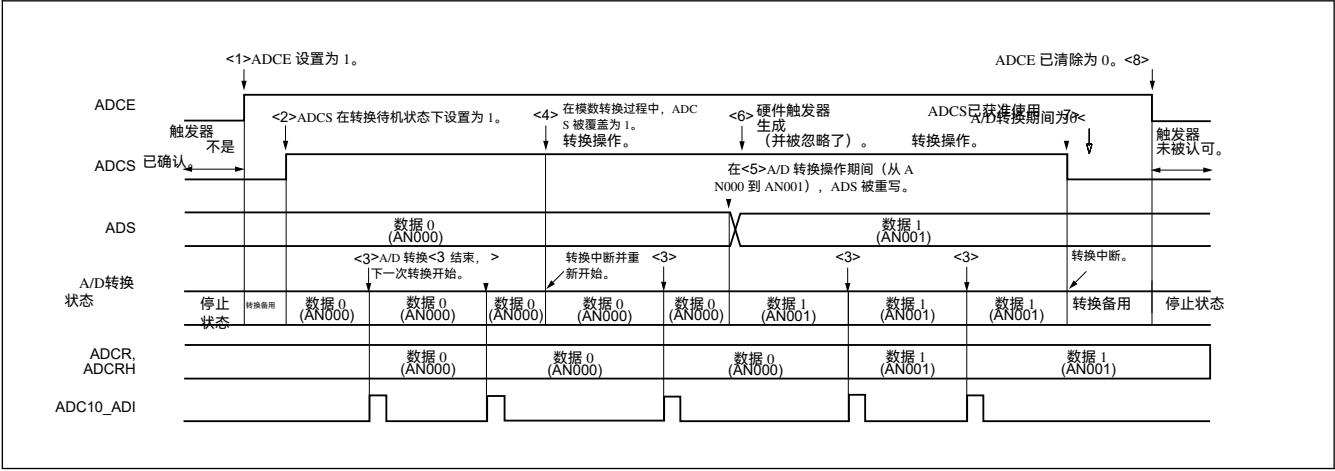


图 22.7 软件触发无等待模式（选择模式、顺序转换模式）操作时序示例

22.5.2 软件触发模式（选择模式、单次转换模式）

- <1>在停止状态下，转换器模式寄存器 0 (ADM0) 的 ADCE 位设置为 1，转换器进入待机状态。
- <2>软件计数到稳定等待时间(1 μ s)后，ADM0 寄存器的 ADCS 位设置为 1，以执行模拟输入通道规范寄存器 (ADS) 指定的模拟输入的 A/D 转换。
- <3>当 A/D 转换结束时，转换结果存储在 A/D 转换结果寄存器 (ADCR、ADCRH) 中，并生成 A/D 转换结束中断请求信号 (ADC10_ADI)。
- <4>转换结束后，A/DADCS 位自动清零，A/D 转换器进入待机状态。<5>转换过程中，若 ADCS 位被写入 1，则当前转换中断，A/D 转换重新开始。部分转换的数据将被丢弃。
- <6>当转换操作期间 ADS 寄存器的值被重写或覆盖时，当前的 A/D 转换将被中断，A/D 转换将对 ADS 寄存器重新指定的模拟输入执行。部分转换的数据将被丢弃。
- <7>当转换操作期间 ADCS 被清除为 0 时，当前的 A/D 转换中断，A/D 转换器进入待机状态。
- 当 ADCE 在转换待机状态下被清除为 0 时，转换器进入停止状态。此外，即使在 A/D 转换待机状态下输入硬件触发信号，转换也不会启动。当 ADCS 被指定为 1 时，转换不会启动。

图 22.8 显示了软件选择无等待模式（选择模式、单次转换模式）操作时序的示例。

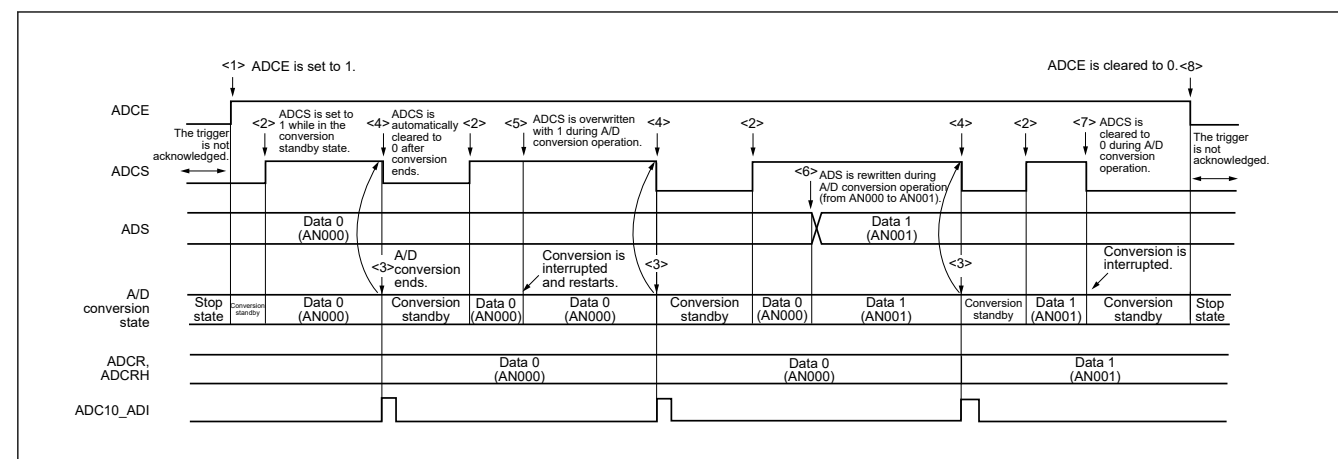


Figure 22.8 Example of software select mode (select mode, one-shot conversion mode) operation timing

22.5.3 Hardware Trigger No-wait Mode (Select Mode, Sequential Conversion Mode)

<1> In the stop state, the ADCE bit of A/D converter mode register 0 (ADM0) is set to 1, and the A/D converter enters the standby state.

<2> After the software counts up to the stabilization wait time (1 μ s), the ADCS bit of the ADM0 register is set to 1 to place the A/D converter in the hardware trigger standby state (and conversion does not start at this stage). Note that, while in this state, A/D conversion does not start even if ADCS is set to 1.

<3> If a hardware trigger is input while ADCS = 1, A/D conversion is performed on the analog input specified by the analog input channel specification register (ADS).

<4> When A/D conversion ends, the conversion result is stored in the A/D conversion result registers (ADCR, ADCRH), and the A/D conversion end interrupt request signal (ADC10_ADI) is generated. After A/D conversion ends, the next A/D conversion immediately starts.

<5> If a hardware trigger is input during conversion operation, the current A/D conversion is interrupted, and conversion restarts. The partially converted data is discarded.

<6> When the value of the ADS register is rewritten or overwritten during conversion operation, the current A/D conversion is interrupted, and A/D conversion is performed on the analog input respecified by the ADS register. The partially converted data is discarded.

<7> When ADCS is overwritten with 1 during conversion operation, the current A/D conversion is interrupted, and conversion restarts. The partially converted data is discarded.

<8> When ADCS is cleared to 0 during conversion operation, the current A/D conversion is interrupted, and the A/D converter enters the standby state. However, the A/D converter does not stop in this state.

<9> When ADCE is cleared to 0 while in the A/D conversion standby state, the A/D converter enters the stop state.

When ADCS = 0, inputting a hardware trigger is ignored and A/D conversion does not start.

Figure 22.9 shows the example of hardware trigger no-wait mode (select mode, sequential conversion mode) operation timing.

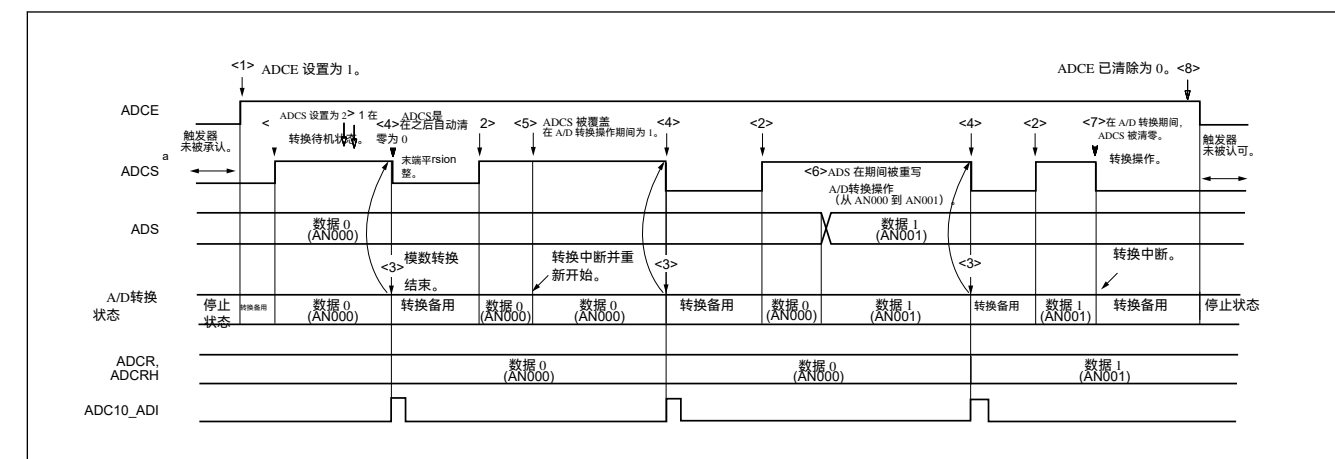


图 22.8 软件选择模式（选择模式、单次转换模式）操作时序示例

22.5.3 硬件触发无等待模式（选择模式、顺序转换模式）

<1>在停止状态下，转换器模式寄存器 0 (ADM0) 的 ADCE 位设置为 1，转换器进入待机状态。

当软件计数到稳定等待时间后，ADM0寄存器的ADCS位被设置为1，使转换器进入硬件触发待机状态（此时转换不会开始）。请注意，在此状态下，即使ADCS位设置为1，A/D转换也不会开始。

<3>如果输入硬件触发信号，则在模拟输入通道规范寄存器 (ADS) 指定的模拟输入上执行转换。

当转换结束时，转换结果存储在转换结果寄存器（ADCR、ADCRH）中，并生成转换结束中断请求信号（ADC1_0_ADI）。A/D 转换结束后，下一个 A/D 转换立即开始。

<5>如果在转换过程中输入硬件触发信号，则当前的A/D转换将被中断，转换将重新开始。已部分转换的数据将被丢弃。

<6>当转换操作期间 ADS 寄存器的值被重写或覆盖时，当前的 A/D 转换将被中断，A/D 转换将对 ADS 寄存器重新指定的模拟输入执行。部分转换的数据将被丢弃。

<7>当转换操作期间ADCS被覆盖为1时，当前的A/D转换将被中断，转换将重新开始。部分转换的数据将被丢弃。

<8>当转换操作期间ADCS被清除为0时，当前的A/D转换中断，A/D转换器进入待机状态。但是，A/D转换器在此状态下不会停止工作。

<9>当 ADCE 在 A/D 转换待机状态下被清除为 0 时，A/D 转换器进入停止状态。

当ADCS = 0, 输入硬件触发器时, 硬件触发器输入将被忽略, 并且A/D转换不会开始。

图 22.9 显示了硬件触发无等待模式（选择模式、顺序转换模式）操作时序的示例。

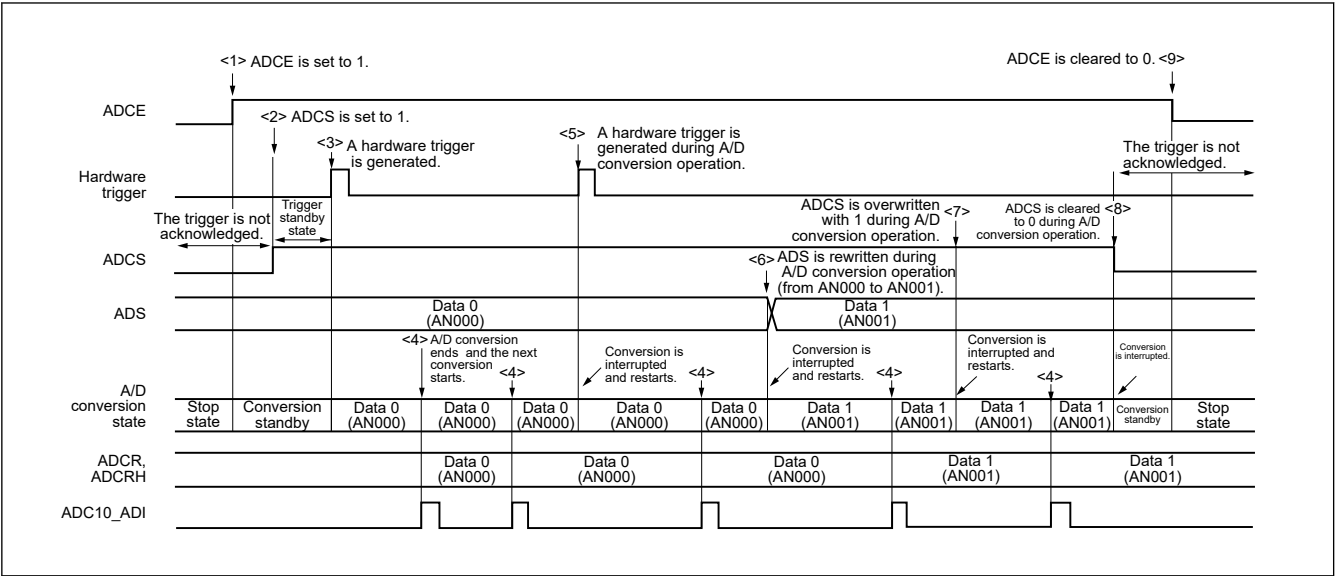


Figure 22.9 Example of hardware trigger no-wait mode (select mode, sequential conversion mode) operation timing

22.5.4 Hardware Trigger No-wait Mode (Select Mode, One-shot Conversion Mode)

<1> In the stop state, the ADCE bit of A/D converter mode register 0 (ADM0) is set to 1, and the A/D converter enters the standby state.

<2> After the software counts up to the stabilization wait time (1 μ s), the ADCS bit of the ADM0 register is set to 1 to place the A/D converter in the hardware trigger standby state (and conversion does not start at this stage). Note that, while in this state, A/D conversion does not start even if ADCS is set to 1.

<3> If a hardware trigger is input while ADCS = 1, A/D conversion is performed on the analog input specified by the analog input channel specification register (ADS).

<4> When A/D conversion ends, the conversion result is stored in the A/D conversion result registers (ADCR, ADCRH), and the A/D conversion end interrupt request signal (ADC10_ADI) is generated.

<5> After A/D conversion ends, the ADCS bit remains set to 1, and the A/D converter enters the standby state.

<6> If a hardware trigger is input during conversion operation, the current A/D conversion is interrupted, and conversion restarts. The partially converted data is discarded.

<7> When the value of the ADS register is rewritten or overwritten during conversion operation, the current A/D conversion is interrupted, and A/D conversion is performed on the analog input respecified by the ADS register. The partially converted data is discarded.

<8> When ADCS is overwritten with 1 during conversion operation, the current A/D conversion is interrupted, and conversion restarts. The partially converted data is discarded.

<9> When ADCS is cleared to 0 during conversion operation, the current A/D conversion is interrupted, and the A/D converter enters the standby state. However, the A/D converter does not stop in this state.

<10> When ADCE is cleared to 0 while in the A/D conversion standby state, the A/D converter enters the stop state.

When ADCS = 0, inputting a hardware trigger is ignored and A/D conversion does not start.

Figure 22.10 shows the example of hardware trigger no-wait mode (select mode, one-shot conversion mode) operation timing.

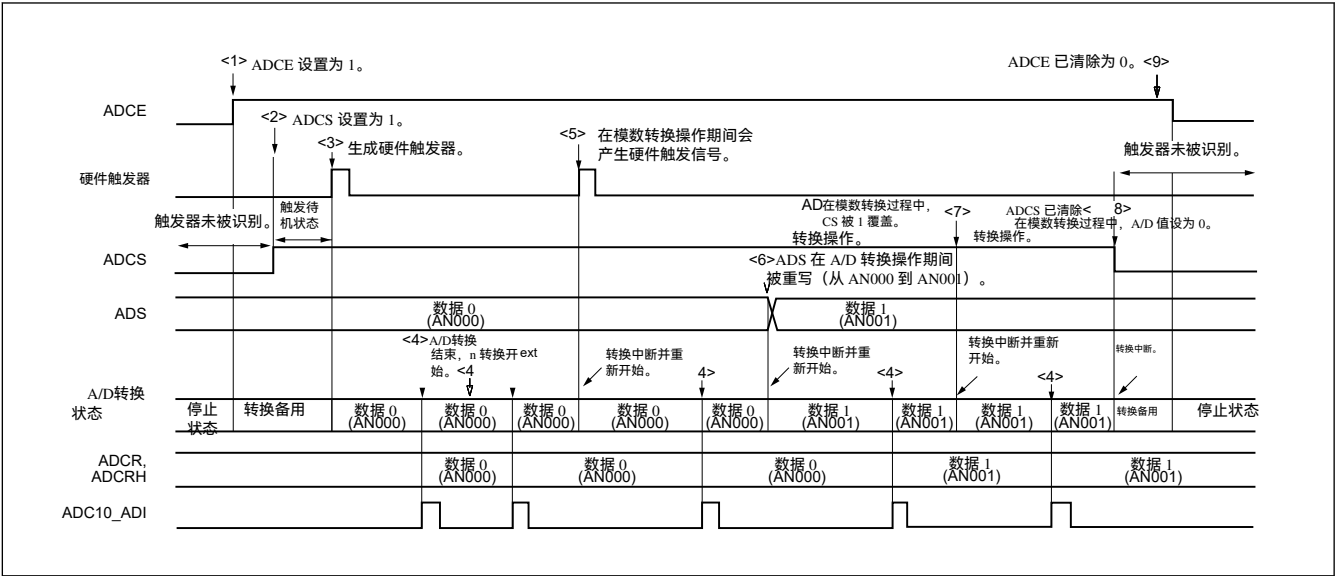


图 22.9 硬件触发无等待模式（选择模式、顺序转换模式）操作时序示例

22.5.4 硬件触发无等待模式（选择模式、单次转换模式）

<1>在停止状态下，转换器模式寄存器 0 (ADM0) 的 ADCE 位设置为 1，转换器进入待机状态。

当软件计数到稳定等待时间后，ADM0寄存器的ADCS位被设置为1，使转换器进入硬件触发待机状态（此时转换不会开始）。请注意，在此状态下，即使ADCS位设置为1，A/D转换也不会开始。

<3>如果输入硬件触发信号，则在模拟输入通道规范寄存器 (ADS) 指定的模拟输入上执行转换。

<4>当 A/D 转换结束时，转换结果存储在 A/D 转换结果寄存器 (ADCR, ADCRH) 中，并生成A/D 转换结束中断请求信号 (ADC10_ADI)。

<5>转换结束后，ADCS 位保持设置为 1，A/D 转换器进入待机状态。

<6>如果在转换操作期间输入硬件触发信号，则当前转换将被中断，并重新开始转换。已部分转换的数据将被丢弃。

当转换操作期间 ADS 寄存器的值被重写或覆盖时，当前的转换将被中断，并对 ADS 寄存器重新指定的模拟输入执行转换。部分转换的数据将被丢弃。

<8>当转换操作期间ADCS被覆盖为1时，当前的A/D转换将被中断，转换将重新开始。部分转换的数据将被丢弃。

<9>当转换操作期间ADCS被清除为0时，当前的A/D转换中断，A/D转换器进入待机状态。但是，在此状态下，转换器不会停止运行。

当 ADCE 在转换待机状态下被清除为 0 时，A/D 转换器进入停止状态。

当ADCS = 0，输入硬件触发器时，硬件触发器输入将被忽略，并且A/D转换不会开始。

图 22.10 显示了硬件触发无等待模式（选择模式、单次转换模式）操作时序的示例。

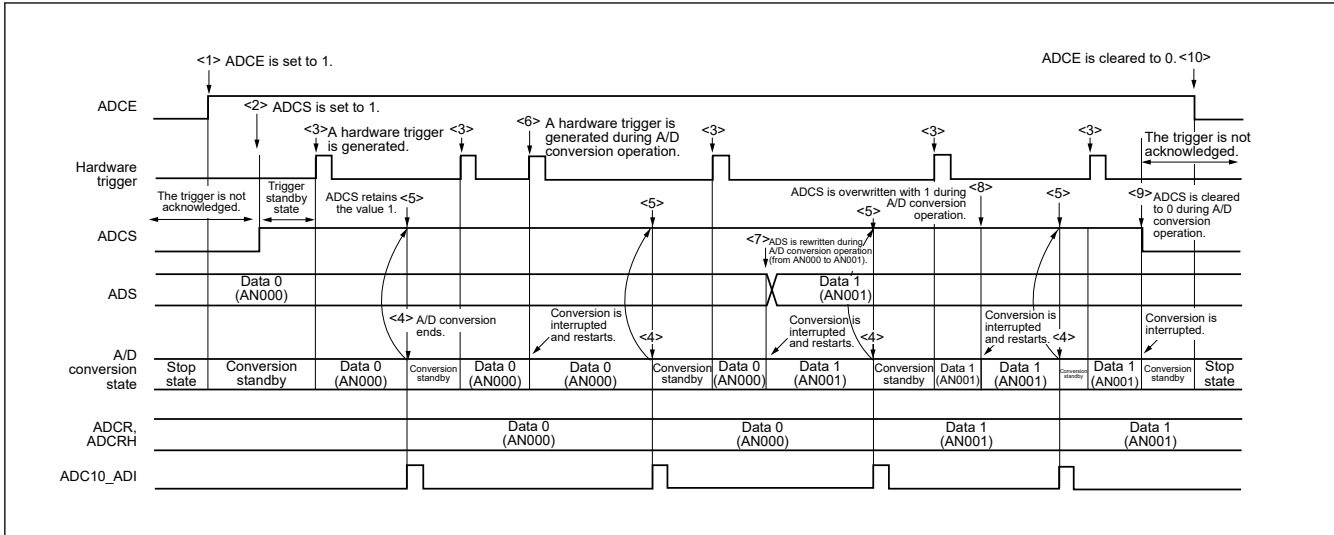


Figure 22.10 Example of hardware trigger no-wait mode (select mode, one-shot conversion mode) operation timing

22.5.5 Hardware Trigger Wait Mode (Select Mode, Sequential Conversion Mode)

<1> In the stop state, the ADCE bit of A/D converter mode register 0 (ADM0) is set to 1, and the A/D converter enters the hardware trigger standby state.

<2> If a hardware trigger is input while in the hardware trigger standby state, A/D conversion is performed on the analog input specified by the analog input channel specification register (ADS). The ADCS bit of the ADM0 register is automatically set to 1 according to the hardware trigger input.

<3> When A/D conversion ends, the conversion result is stored in the A/D conversion result registers (ADCR, ADCRH), and the A/D conversion end interrupt request signal (ADC10_ADI) is generated. After A/D conversion ends, the next A/D conversion immediately starts. (At this time, no hardware trigger is necessary.)

<4> If a hardware trigger is input during conversion operation, the current A/D conversion is interrupted, and conversion restarts. The partially converted data is discarded.

<5> When the value of the ADS register is rewritten or overwritten during conversion operation, the current A/D conversion is interrupted, and A/D conversion is performed on the analog input respecified by the ADS register. The partially converted data is discarded.

<6> When ADCS is overwritten with 1 during conversion operation, the current A/D conversion is interrupted, and conversion restarts. The partially converted data is discarded.

<7> When ADCS is cleared to 0 during conversion operation, the current A/D conversion is interrupted, the A/D converter enters the hardware trigger standby state, and the A/D converter enters the stop state.

<8> If ADCE = 0 is set during the hardware trigger wait state, the A/D converter is stopped. When ADCE = 0, the hardware trigger input is ignored and A/D conversion does not start.

Figure 22.11 shows the example of hardware trigger wait mode (select mode, sequential conversion mode) operation timing.

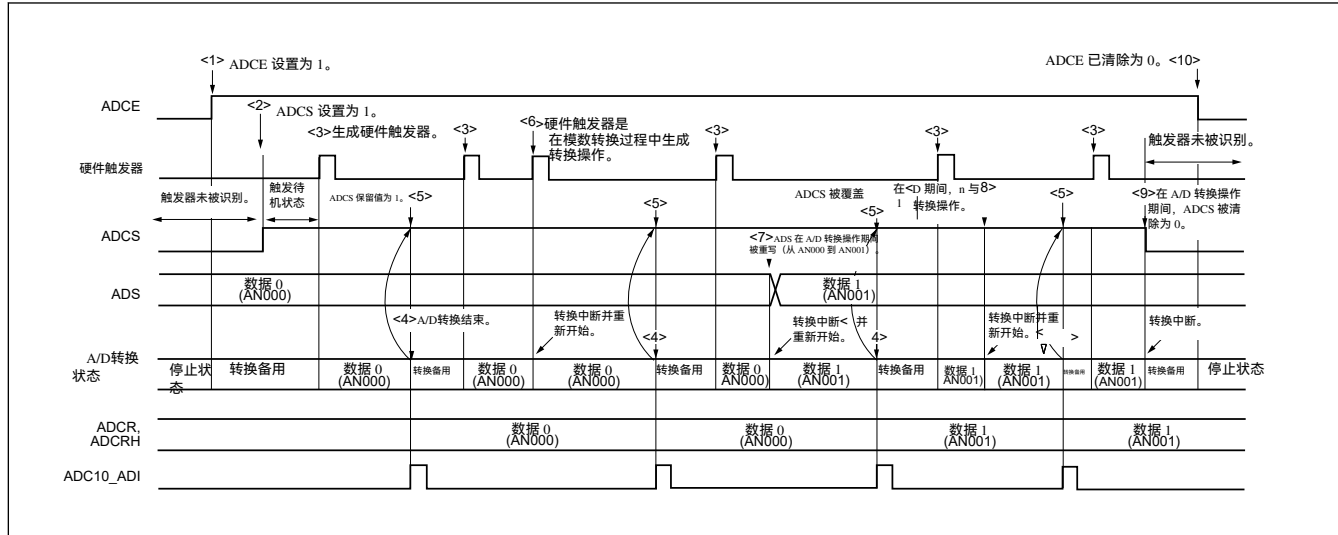


图 22.10 硬件触发无等待模式（选择模式、单次转换模式）操作时序示例

22.5.5 硬件触发等待模式（选择模式、顺序转换模式）

<1>在停止状态下，转换器模式寄存器 0 (ADM0) 的 ADCE 位设置为 1，转换器进入硬件触发待机状态。

<2>如果在硬件触发待机状态下输入硬件触发信号，则会对模拟输入通道规范寄存器 (ADS) 指定的模拟输入信号执行模数转换。根据硬件触发输入信号，ADM0 寄存器的 ADCS 位会自动设置为 1。

<3>当模数转换结束时，转换结果存储在模数转换结果寄存器 (ADCR、ADCRH) 中，并生成转换结束中断请求信号 (ADC10_ADI)。模数转换结束后，下一个模数转换立即开始。（此时无需硬件触发。）

<4>如果在转换过程中输入硬件触发信号，则当前的A/D转换将被中断，转换将重新开始。已部分转换的数据将被丢弃。

当转换操作期间 ADS 寄存器的值被重写或覆盖时，当前的转换将被中断，并对 ADS 寄存器重新指定的模拟输入执行转换。部分转换的数据将被丢弃。

<6>当转换操作期间 ADCS 被 1 覆盖时，当前的 A/D 转换将被中断，转换将重新开始。部分转换的数据将被丢弃。

<7>当转换操作期间 ADCS 被清除为 0 时，当前的 A/D 转换中断，A/D 转换器进入硬件触发待机状态，A/D转换器进入停止状态。

<8>如果在硬件触发等待状态下设置了ADCE = 0，则 A/D 转换器停止工作。当ADCE = 0, 时，硬件触发输入将被忽略，并且A/D转换不会开始。

图 22.11 显示了硬件触发等待模式（选择模式、顺序转换模式）操作时序的示例。

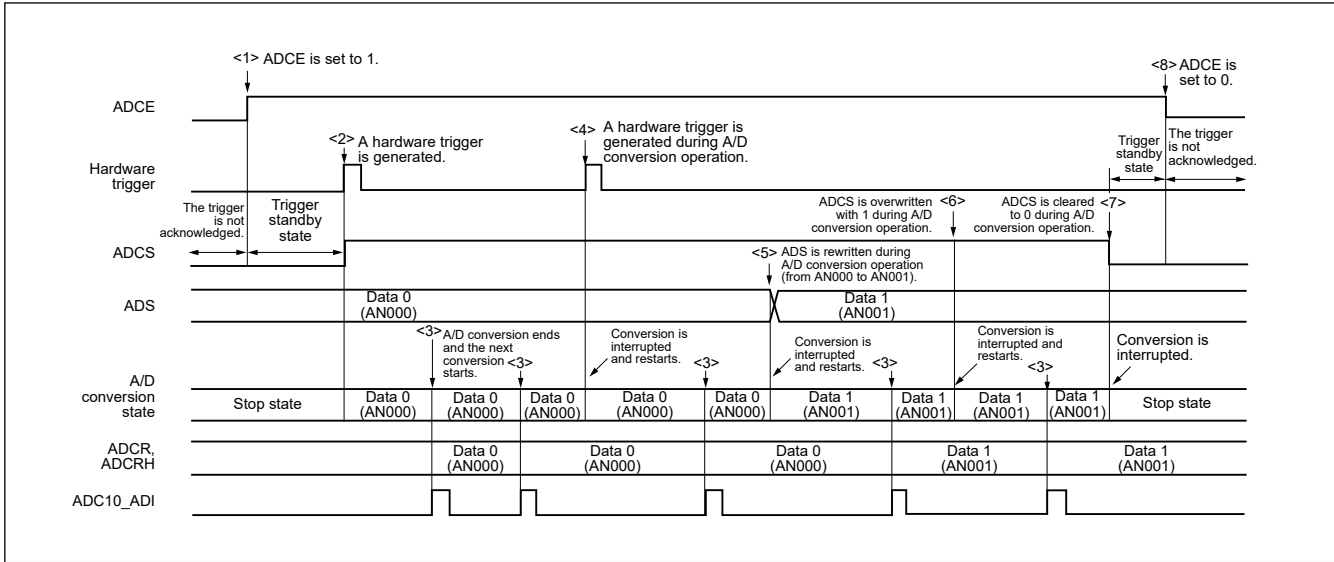


Figure 22.11 Example of hardware trigger wait mode (select mode, sequential conversion mode) operation timing

22.5.6 Hardware Trigger Wait Mode (Select Mode, One-shot Conversion Mode)

- <1> In the stop state, the ADCE bit of A/D converter mode register 0 (ADM0) is set to 1, and the A/D converter enters the hardware trigger standby state.
- <2> If a hardware trigger is input while in the hardware trigger standby state, A/D conversion is performed on the analog input specified by the analog input channel specification register (ADS). The ADCS bit of the ADM0 register is automatically set to 1 according to the hardware trigger input.
- <3> When A/D conversion ends, the conversion result is stored in the A/D conversion result registers (ADCR, ADCRH), and the A/D conversion end interrupt request signal (ADC10_ADI) is generated.
- <4> After A/D conversion ends, the ADCS bit is automatically cleared to 0, and the A/D converter enters the stop state.
- <5> If a hardware trigger is input during conversion operation, the current A/D conversion is interrupted, and conversion restarts. The partially converted data is discarded.
- <6> When the value of the ADS register is rewritten or overwritten during conversion operation, the current A/D conversion is interrupted, and A/D conversion is performed on the analog input respecified by the ADS register. The partially converted data is discarded.
- <7> When ADCS is overwritten with 1 during conversion operation, the current A/D conversion is interrupted, and conversion restarts. The partially converted data is initialized.
- <8> When ADCS is cleared to 0 during conversion operation, the current A/D conversion is interrupted, the A/D converter enters the hardware trigger standby state, and the A/D converter enters the stop state.
- <9> If ADCE = 0 is set during the hardware trigger wait state, the A/D converter is stopped. When ADCE = 0, the hardware trigger input is ignored and A/D conversion does not start.

Figure 22.12 shows the example of hardware trigger wait mode (select mode, one-shot conversion mode) operation timing.

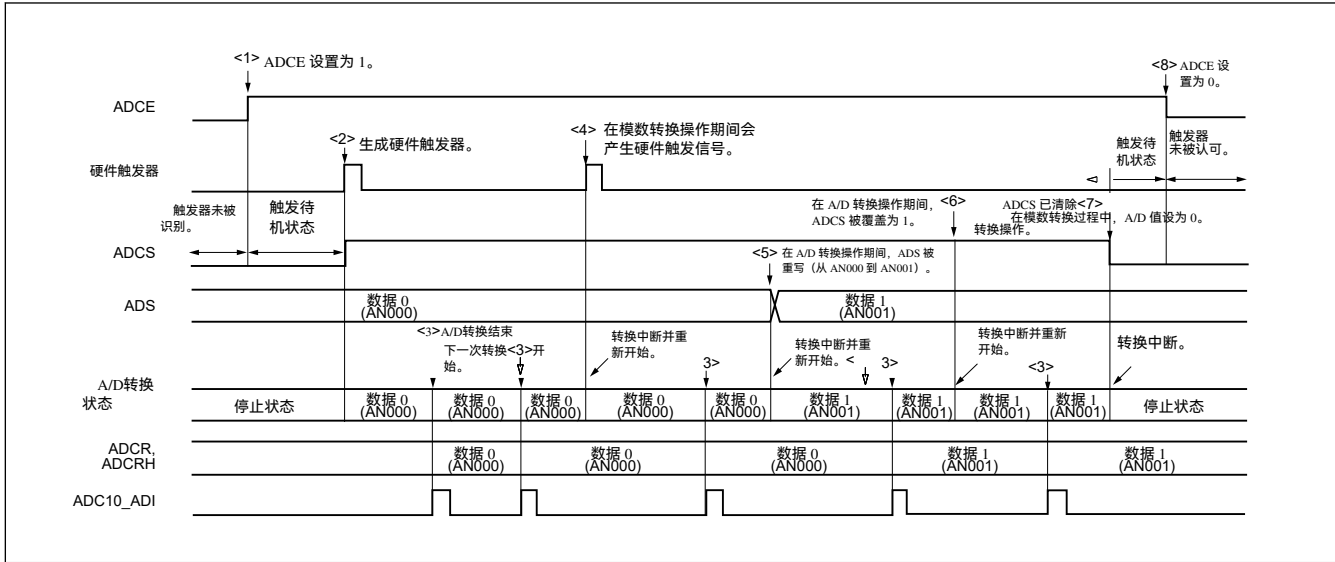


图 22.11 硬件触发等待模式（选择模式、顺序转换模式）操作时序示例

22.5.6 硬件触发等待模式（选择模式、单次转换模式）

- <1>在停止状态下，转换器模式寄存器 0 (ADM0) 的 ADCE 位设置为 1，转换器进入硬件触发待机状态。
- <2>如果在硬件触发待机状态下输入硬件触发信号，则会对模拟输入通道规范寄存器 (ADS) 指定的模拟输入信号执行模数转换。根据硬件触发输入信号，ADM0 寄存器的 ADCS 位会自动设置为 1。
- 当转换结束时，转换结果存储在 A/D 转换结果寄存器 (ADCR、ADCRH) 中，并生成转换结束中断请求信号 (ADC10_ADI)。
- <4>转换结束后，ADCS 位自动清零，A/D 转换器进入停止状态。
- <5>如果在转换过程中输入硬件触发信号，则当前的 A/D 转换将被中断，转换将重新开始。已部分转换的数据将被丢弃。
- 当转换操作期间 ADS 寄存器的值被重写或覆盖时，当前的转换将被中断，并对 ADS 寄存器重新指定的模拟输入执行转换。部分转换的数据将被丢弃。
- <7>当转换操作期间 ADCS 被覆盖为 1 时，当前的 A/D 转换将被中断，转换将重新开始。部分转换的数据将被初始化。
- <8>当转换操作期间 ADCS 被清除为 0 时，当前的 A/D 转换中断，A/D 转换器进入硬件触发待机状态，A/D 转换器进入停止状态。
- <9>如果在硬件触发等待状态下设置了 ADCE = 0，则 A/D 转换器停止工作。当 ADCE = 0 时，硬件触发输入将被忽略，并且 A/D 转换不会开始。

图 22.12 显示了硬件触发等待模式（选择模式、单次转换模式）操作时序的示例。

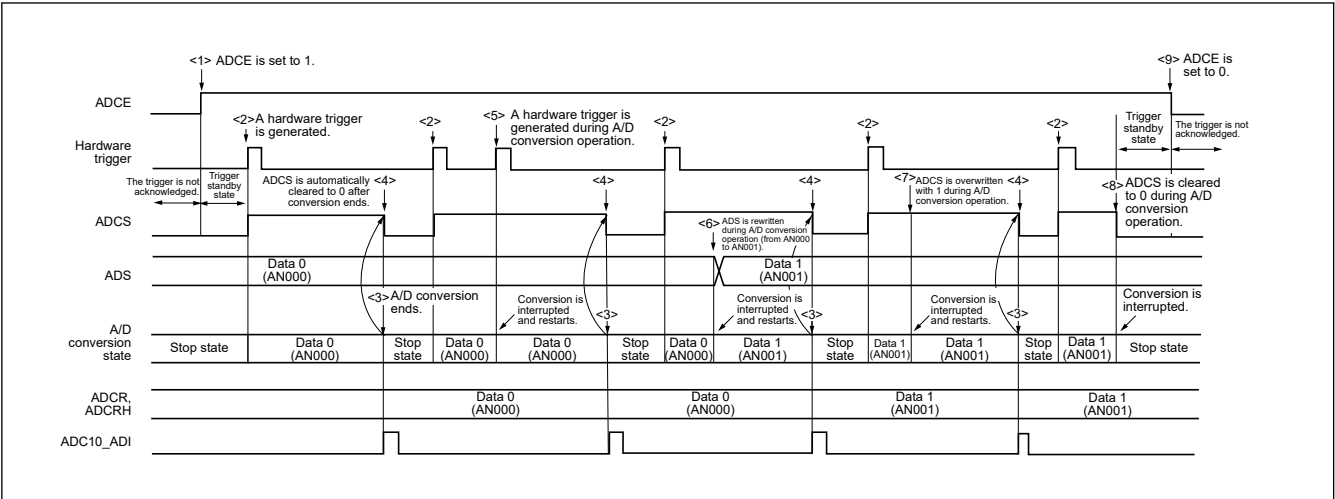


Figure 22.12 Example of hardware trigger wait mode (select mode, one-shot conversion mode) operation timing

22.6 A/D Converter Setup Procedure

The A/D converter setup procedure in each operation mode is described in the following section.

22.6.1 Setting up Software Trigger Mode

Table 22.13 shows the setup steps in software trigger mode.

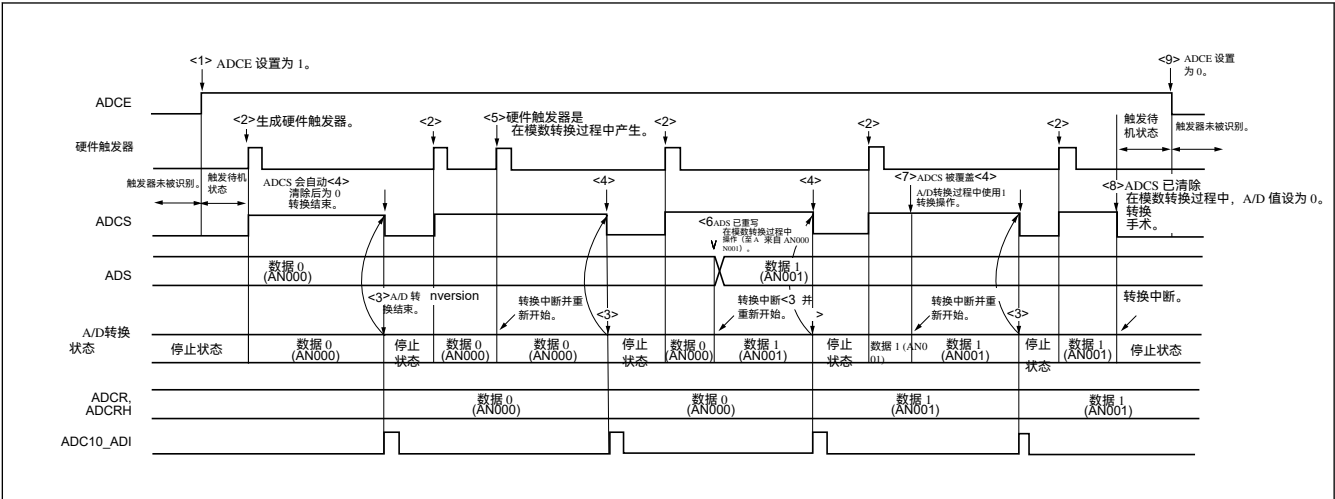


图 22.12 硬件触发等待模式（选择模式、单次转换模式）操作时序示例

22.6A/D转换器设置步骤

以下部分描述了每种操作模式下的A/D转换器设置过程。

22.6.1 设置软件触发模式

表 22.13 显示了软件触发模式下的设置步骤。

Table 22.13 Setting up software trigger mode

Step		Process	Detail
Setting up Software Trigger No-Wait Mode	<1>	MSTPCRD register setting	The MSTPD16 bit of the MSTPCRD register is set to 0, and supplying the clock starts.
	<2>	PmnPFS_A register settings	The ports are set as the analog input. (See section 15.5.4. Notes on Using Analog Functions.)
	<3>	- ADM0 register setting - ADM1 register setting - ADM2 register setting - ADUL and ADLL register setting - ADS register setting (The order of the settings is irrelevant.)	- ADM0 register FR[2:0], LV[1:0]: bits: These are used to specify the A/D conversion time . ADMD bit: Select mode - ADM1 register ADTMD1 and ADTMD0 bits: These are used to specify the software trigger mode. ADSCM bit: Sequential conversion mode or one-shot conversion mode - ADM2 register ADREFP[1:0], and ADREFM bits: These used to select the reference voltage. ADRCK bit: This is used to select the range for the A/D conversion result comparison value for generating the interrupt signal from AREA1, AREA3, and AREA 2. ADTYP bit: 10-bit, or 8-bit resolution - ADUL and ADLL register These are used to specify the upper limit and lower limit A/D conversion result comparison values. - ADS register ADS[4:0] bits: These are used to select the analog input channels.
	<6>	Reference voltage stabilization wait time count A	The reference voltage stabilization wait time count indicated by A below may be required if the values of the ADREFP[1:0] bits are changed. If the values of ADREFP[1:0] are changed to 10b, respectively: A = 5 μs A wait is not required if the values of ADREFP[1:0] are changed to 00b or 01b, respectively.
	<7>	ADCE bit setting	The ADCE bit of the ADM0 register is set to 1, and the A/D converter enters the standby state.
	<8>	Reference voltage stabilization wait time count B	Use software to control waiting until reference voltage stabilization wait time count B (1 μs) elapses.
	<9>	ADCS bit setting	After reference voltage stabilization wait time count B elapses, the ADCS bit of the ADM0 register is set to 1, and A/D conversion starts.
	<10>	Start of A/D conversion	—
	—	:	(The A/D conversion operations are performed)
	<11>	End of A/D conversion	The A/D conversion end interrupt (ADC10_ADI) is generated.*1
	<12>	Storage of conversion results in the ADCR and ADCRH register	The conversion results are stored in the ADCR and ADCRH register.

Note 1. Depending on the settings of the ADRCK bit, ADUL and ADLL registers, there is a possibility of no interrupt signal being generated.
In this case, the results are not stored in the ADCR and ADCRH register.

22.6.2 Setting up Hardware Trigger No-wait Mode

Table 22.14 shows the setup steps in hardware trigger no-wait mode.

表 22.13 设置软件触发模式

步		过程	细节
设置软件触发器 无需等待模式	<1>	MSTPCRD寄存器设置	MSTPCRD 寄存器的 MSTPD16 位设置为 0，开始提供时钟信号。
	<2>	PmnPFS_A 寄存器设置	端口设置为模拟输入。 (参见第 15.5.4 节“模拟函数使用注意事项”。)
	<3>	●ADM0寄存器设置 ●ADM1寄存器设置 ●ADM2寄存器设置 ●ADUL 和 ADLL 寄存器设置 ●ADS寄存器设置（设置顺序无关紧要。）	●ADM0寄存器 FR[2:0], LV[1:0]: bits: 这些用于指定 A/D 转换时间。 ADMD 位：选择模式 ●ADM1 注册 ADTMD1 和 ADTMD0 位：用于指定软件触发模式。 ADSCM 位：顺序转换模式或单次转换模式 ●ADM2寄存器 ADREFP[1:0]和 ADREFM 位： 这些用于选择参考电压。 ADRCK 位：用于选择 A/D 转换结果比较值的范围，以便从 AR EA1、AREA3 和 AREA 2 生成中断信号。 ADTYP 位： 10 位或 8 位分辨率 ●ADUL 和 ADLL 寄存器 这些用于指定 A/D 转换结果比较值的上限和下限。 ●广告交易登记册 ADS[4:0] 位： 这些位用于选择模拟输入通道。
	<6>	参考电压稳定等待时间计数 A	如果 ADREFP[1:0] 位的值发生变化，则可能需要下面 A 指示的参考电压稳定等待时间计数。 如果 ADREFP[1:0] 的值分别更改为 10b: A =5 μs 如果 ADREFP[1:0] 的值分别更改为 00b 或 01b, 则不需要等待。
	<7>	ADCE 位设置	ADM0 寄存器的 ADCE 位设置为 1，A/D 转换器进入待机状态。
	<8>	参考电压稳定等待时间计数 B	使用软件控制等待，直到参考电压稳定等待时间计数 B(1 μs) 结束。
	<9>	ADCS 位设置	参考电压稳定等待时间计数 B 结束后，ADM0 寄存器的 ADCS 位被设置为 1，A/D 转换开始。
	<10>	模数转换开始	—
	—	:	(执行模数转换操作)
	<11>	A/D转换结束	生成模数转换结束中断（ADC10_ADI）。*1
	<12>	转换结果的存储方式 ADCR 和 ADCRH 注册	转换结果存储在 ADCR 和 ADCRH 寄存器中。

注1. 根据 ADRCK 位、ADUL 和 ADLL 寄存器的设置，有可能不会产生中断信号。
在这种情况下，结果不会存储在 ADCR 和 ADCRH 寄存器中。

22.6.2 设置硬件触发无等待模式

表 22.14 显示了硬件触发无等待模式下的设置步骤。

Table 22.14 Setting up hardware trigger no-wait mode

Step		Process	Detail
Setting up Hardware Trigger No-Wait Mode	<1>	MSTPCRD register setting	The MSTPD16 bit of the MSTPCRD register is set to 0, and supplying the clock starts.
	<2>	PmnPFS_A register settings	The ports are set as the analog input. (See section 15.5.4. Notes on Using Analog Functions.)
	<3>	- ADM0 register setting - ADM1 register setting - ADM2 register setting - ADUL and ADLL register setting - ADS register setting (The order of the settings is irrelevant.)	- ADM0 register FR[2:0], LV[1:0]: bits: These are used to specify the A/D conversion time . ADMD bit: Select mode - ADM1 register ADTMD1 and ADTMD0 bits: These are used to specify the hardware trigger no-wait mode. ADSCM bit: Sequential conversion mode or one-shot conversion mode - ADM2 register ADREFP[1:0], and ADREFM bits: These used to select the reference voltage. ADRCK bit: This is used to select the range for the A/D conversion result comparison value for generating the interrupt signal from AREA1, AREA3, and AREA 2. ADTYP bit: 10-bit, or 8-bit resolution - ADUL and ADLL register These are used to specify the upper limit and lower limit A/D conversion result comparison values. - ADS register ADS[4:0] bits: These are used to select the analog input channels.
	<6>	Reference voltage stabilization wait time count A	The reference voltage stabilization wait time count indicated by A below may be required if the values of the ADREFP[1:0] bits are changed. If the values of ADREFP[1:0] are changed to 10b, respectively: A = 5 μs A wait is not required if the values of ADREFP[1:0] are changed to 00b or 01b, respectively.
	<7>	ADCE bit setting	The ADCE bit of the ADM0 register is set to 1, and the A/D converter enters the standby state.
	<8>	Reference voltage stabilization wait time count B	Use software to control waiting until reference voltage stabilization wait time count B (1 μs) elapses.
	<9>	ADCS bit setting	After reference voltage stabilization wait time count B elapses, the ADCS bit of the ADM0 register is set to 1, and A/D converter enters the hardware trigger standby state.
	<10>	Start of A/D conversion	—
	—	:	(The A/D conversion operations are performed)
	<11>	End of A/D conversion	The A/D conversion end interrupt (ADC10_ADI) is generated.*1
	<12>	Storage of conversion results in the ADCR and ADCRH register	The conversion results are stored in the ADCR and ADCRH register.

Note 1. Depending on the settings of the ADRCK bit, ADUL and ADLL registers, there is a possibility of no interrupt signal being generated.
In this case, the results are not stored in the ADCR and ADCRH register.

22.6.3 Setting up Hardware Trigger Wait Mode

Table 22.15 shows the setup steps in hardware trigger wait mode.

表 22.14 设置硬件触发无等待模式

步		过程	细节
设置硬件触发器 无需等待模式	<1>	MSTPCRD寄存器设置	MSTPCRD 寄存器的 MSTPD16 位设置为 0，开始提供时钟信号。
	<2>	PmnPFS_A 寄存器设置	端口设置为模拟输入。 (参见第 15.5.4 节“模拟函数使用注意事项”。)
	<3>	●ADM0寄存器设置 ●ADM1寄存器设置 ●ADM2寄存器设置 ●ADUL 和 ADLL 寄存器设置 ●ADS寄存器设置（设置顺序无关紧要。）	●ADM0寄存器 FR[2:0], LV[1:0]: bits: 这些用于指定 A/D 转换时间。 ADMD 位：选择模式 ●ADM1 注册 ADTMD1 和 ADTMD0 位： 这些位用于指定硬件触发无等待模式。 ADSCM 位： 顺序转换模式或单次转换模式 ●ADM2寄存器 ADREFP[1:0]和 ADREFM 位： 这些用于选择参考电压。 ADRCK 位： 用于选择 A/D 转换结果比较值的范围，以便从 AREA1、AREA3 和 AREA 2 生成中断信号。 ADTYP 位： 10 位或 8 位分辨率 ●ADUL 和 ADLL 寄存器 这些用于指定 A/D 转换结果比较值的上限和下限。 ●广告交易登记册 ADS[4:0] 位： 这些位用于选择模拟输入通道。
	<6>	参考电压稳定等待时间计数 A	如果 ADREFP[1:0] 位的值发生变化，则可能需要下面 A 指示的参考电压稳定等待时间计数。 如果 ADREFP[1:0] 的值分别更改为 10b: A =5 μs 如果 ADREFP[1:0] 的值分别更改为 00b 或 01b, 则不需要等待。
	<7>	ADCE 位设置	ADM0 寄存器的 ADCE 位设置为 1，A/D 转换器进入待机状态。
	<8>	参考电压稳定等待时间计数 B	使用软件控制等待，直到参考电压稳定等待时间计数 B(1 μs) 结束。
	<9>	ADCS 位设置	参考电压稳定等待时间计数 B 结束后，ADM0 寄存器的 ADCS 位被设置为 1，A/D 转换器进入硬件触发待机状态。
	<10>	模数转换开始	—
	—	:	(执行模数转换操作)
	<11>	A/D转换结束	生成模数转换结束中断（ADC10_ADI）。*1
	<12>	转换结果的存储方式 ADCR 和 ADCRH 注册	转换结果存储在 ADCR 和 ADCRH 寄存器中。

注1. 根据 ADRCK 位、ADUL 和 ADLL 寄存器的设置，有可能不会产生中断信号。
在这种情况下，结果不会存储在 ADCR 和 ADCRH 寄存器中。

22.6.3 设置硬件触发等待模式

表 22.15 显示了硬件触发等待模式下的设置步骤。

Table 22.15 Setting up hardware trigger wait mode

Step		Process	Detail
Setting up Hardware Trigger Wait Mode	<1>	MSTPCRD register setting	The MSTPD16 bit of the MSTPCRD register is set to 0, and supplying the clock starts.
	<2>	PmnPFS_A register settings	The ports are set as the analog input. (See section 15.5.4. Notes on Using Analog Functions.)
	<3>	- ADM0 register setting - ADM1 register setting - ADM2 register setting - ADUL and ADLL register setting - ADS register setting (The order of the settings is irrelevant.)	- ADM0 register FR[2:0], LV[1:0]: bits: These are used to specify the A/D conversion time . ADMD bit: Select mode - ADM1 register ADTMD1 and ADTMD0 bits: These are used to specify the hardware trigger wait mode. ADSCM bit: Sequential conversion mode or one-shot conversion mode - ADM2 register ADREFP[1:0], and ADREFM bits: These used to select the reference voltage. ADRCK bit: This is used to select the range for the A/D conversion result comparison value for generating the interrupt signal from AREA1, AREA3, and AREA 2. ADTYP bit: 10-bit, or 8-bit resolution - ADUL and ADLL register These are used to specify the upper limit and lower limit A/D conversion result comparison values. - ADS register ADS[4:0] bits: These are used to select the analog input channels.
	<6>	Reference voltage stabilization wait time count A	The reference voltage stabilization wait time count indicated by A below may be required if the values of the ADREFP[1:0] bits are changed. If the values of ADREFP[1:0] are changed to 10b, respectively: A = 5 μs A wait is not required if the values of ADREFP[1:0] are changed to 00b or 01b, respectively.
	<7>	ADCE bit setting	The ADCE bit of the ADM0 register is set to 1, and the A/D converter enters the standby state.
	<8>	Hardware trigger generation	Set the trigger signal output of other modules.
	<9>	Stabilization wait time for A/D power supply	The A/D converter automatically counts up to the stabilization wait time for A/D power supply.
	<10>	Start of A/D conversion	After counting up to the stabilization wait time for A/D power supply ends, A/D conversion starts.
	—	:	(The A/D conversion operations are performed)
	<11>	End of A/D conversion	The A/D conversion end interrupt (ADC10_ADI) is generated.*1
	<12>	Storage of conversion results in the ADCR or ADCRH register	The conversion results are stored in the ADCR or ADCRH register.

Note 1. Depending on the settings of the ADRCK bit, ADUL and ADLL registers, there is a possibility of no interrupt signal being generated. In this case, the results are not stored in the ADCR or ADCRH register.

22.6.4 Example of Using the ADC10 when Selecting the Temperature Sensor Output Voltage or Internal Reference Voltage, and Software Trigger Mode and One-shot Conversion Mode

Table 22.16 shows the setup steps when temperature sensor output voltage and internal reference voltage is selected.

表 22.15 设置硬件触发等待模式

步		过程	细节
设置硬件触发器等待模式	<1>	MSTPCRD寄存器设置	MSTPCRD 寄存器的 MSTPD16 位设置为 0，开始提供时钟信号。
	<2>	PmnPFS_A 寄存器设置	端口设置为模拟输入。 (参见第 15.5.4 节“模拟函数使用注意事项”。)
	<3>	●ADM0寄存器设置 ●ADM1寄存器设置 ●ADM2寄存器设置 ●ADUL 和 ADLL 寄存器设置 ●ADS寄存器设置（设置顺序无关紧要。）	●ADM0寄存器 FR[2:0], LV[1:0]: bits: 这些用于指定 A/D 转换时间。 ADMD 位: 选择模式 ●ADM1 注册 ADTMD1 和 ADTMD0 位: 用于指定硬件触发等待模式。 ADSCM 位: 顺序转换模式或单次转换模式 ●ADM2寄存器 ADREFP[1:0]和 ADREFM 位: 这些用于选择参考电压。 ADRCK 位: 用于选择 A/D 转换结果比较值的范围，以便从 AREA1、AREA3 和 AREA 2 生成中断信号。 ADTYP 位: 10 位或 8 位分辨率 ●ADUL 和 ADLL 寄存器 这些用于指定 A/D 转换结果比较值的上限和下限。 ●广告交易登记册 ADS[4:0] 位: 这些位用于选择模拟输入通道。
	<6>	参考电压稳定等待时间计数 A	如果 ADREFP[1:0] 位的值发生变化，则可能需要下面 A 指示的参考电压稳定等待时间计数。 如果 ADREFP[1:0] 的值分别更改为 10b: A = 5 μs 如果 ADREFP[1:0] 的值分别更改为 00b 或 01b，则不需要等待。
	<7>	ADCE 位设置	ADM0 寄存器的 ADCE 位设置为 1，A/D 转换器进入待机状态。
	<8>	硬件触发器生成	设置其他模块的触发信号输出。
	<9>	A/D电源稳定等待时间	A/D转换器会自动计数，直到达到A/D电源的稳定等待时间。
	<10>	模数转换开始	经过 A/D 电源稳定等待时间后，A/D 转换开始。
	—	:	(执行模数转换操作)
	<11>	A/D转换结束	生成模数转换结束中断（ADC10_ADI）。*1
	<12>	转换结果的存储方式 ADCR 或 ADCRH 登记册	转换结果存储在 ADCR 或 ADCRH 寄存器中。

注1. 根据 ADRCK 位、ADUL 和 ADLL 寄存器的设置，有可能不会产生中断信号。在这种情况下，结果不会存储在 ADCR 或 ADCRH 寄存器中。

22.6.4 选择温度传感器输出电压或内部参考电压、软件触发模式和单次转换模式时使用 ADC 10 的示例

表 22.16 显示了选择温度传感器输出电压和内部参考电压时的设置步骤。

Table 22.16 Setup when temperature sensor output voltage and internal reference voltage is selected

Step		Process	Detail
Setup when temperature sensor output voltage and internal reference voltage is selected	<1>	MSTPCRD register setting	The MSTPD16 bit of the MSTPCRD register is set to 0, and supplying the clock starts.
	<2>	● ADM0 register setting ● ADM1 register setting ● ADM2 register setting ● ADUL and ADLL register setting ● ADS register setting	● ADM0 register FR[2:0], LV[1:0]: bits: These are used to specify the A/D conversion time . ADMD bit: This is used to specify the select mode. ● ADM1 register ADTMD1 and ADTMD0 bits: These are used to specify the software trigger mode. ADSCM bit: One-shot conversion mode ● ADM2 register ADREFP[1:0] and ADREFM bits: These are used to select the reference voltage. ADRCK bit: This is used to select the range for the A/D conversion result comparison value for generating the interrupt signal from AREA1, AREA3, and AREA 2. ADTYP bit: 10-bit, or 8-bit resolution ● ADUL and ADLL register These are used to specify the upper limit and lower limit A/D conversion result comparison values. ● ADS register ADISS and ADS[4:0] bits: These are used to select the temperature sensor output voltage or internal reference voltage.
	<3>	Reference voltage stabilization wait time count A	The reference voltage stabilization wait time count A may be required if the values of the ADREFP[1:0] bits are changed. A wait is not required if the values of ADREFP[1:0] are changed to 00b or 01b, respectively. Setting the values of ADREFP[1:0] to 10b, respectively is prohibited.
	<4>	ADCE bit setting	The ADCE bit of the ADM0 register is set to 1, and the A/D converter enters the standby state.
	<5>	Reference voltage stabilization wait time count B	Use software to control waiting until reference voltage stabilization wait time count B (1 μs) elapses.
	<6>	ADCS bit setting	After reference voltage stabilization wait time count B elapses, the ADCS bit of the ADM0 register is set to 1, and A/D conversion starts.
	<7>	Start of A/D conversion	—
	<8>	End of A/D conversion	The A/D conversion end interrupt (ADC10_ADI) will be generated. After ADISS is set to 1, the initial conversion result cannot be used.
	<9>	ADCS bit setting	The ADCS bit of the ADM0 register is set to 1, and A/D conversion starts.
	<10>	Start of A/D conversion	—
	<11>	End of A/D conversion	The A/D conversion end interrupt (ADC10_ADI) is generated.*1
	<12>	Storage of conversion results in the ADCR or ADCRH register	The conversion results are stored in the ADCR or ADCRH register.

Note 1. Depending on the settings of the ADRCK bit, ADUL and ADLL registers, there is a possibility of no interrupt signal being generated. In this case, the results are not stored in the ADCR or ADCRH register.

22.6.5 Setting Up Test Mode

Table 22.17 shows the setup steps in test mode.

表 22.16 选择温度传感器输出电压和内部参考电压时的设置

步		过程	细节
设置温度传感器输出电压和内部参考电压的选择情况	<1>	MSTPCRD寄存器设置	MSTPCRD 寄存器的 MSTPD16 位设置为 0，开始提供时钟信号。
	<2>	●ADM0寄存器设置 ●ADM1寄存器设置 ●ADM2寄存器设置 ●ADUL 和 ADLL 寄存器设置 ●ADS寄存器设置	●ADM0寄存器 FR[2:0], LV[1:0]: bits: 这些用于指定 A/D 转换时间。 ADMD 位：用于指定选择模式。 ●ADM1 注册 ADTMD1 和 ADTMD0 位：用于指定软件触发模式。 ADSCM 位：单次转换模式 ●ADM2寄存器 ADREFP[1:0] 和 ADREFM 位：这些用于选择参考电压。 ADRCK 位：用于选择 A/D 转换结果比较值的范围，以便从 AR EA1、AREA3 和 AREA 2 生成中断信号。 ADTYP 位：10 位或 8 位分辨率 ●ADUL 和 ADLL 寄存器 这些用于指定 A/D 转换结果比较值的上限和下限。 ●广告交易登记册 ADISS 和 ADS[4:0] 位：这些用于选择温度传感器输出电压或内部参考电压。
	<3>	参考电压稳定等待时间计数 A	如果 ADREFP[1:0] 位的值发生变化，则可能需要参考电压稳定等待时间计数 A。 如果 ADREFP[1:0] 的值分别更改为 00b 或 01b，则不需要等待。 禁止将 ADREFP[1:0] 的值分别设置为 10b。
	<4>	ADCE 位设置	ADM0 寄存器的 ADCE 位设置为 1，A/D 转换器进入待机状态。
	<5>	参考电压稳定等待时间计数 B	使用软件控制等待时间，直到参考电压稳定等待时间计数 B（1 μs）结束。
	<6>	ADCS 位设置	参考电压稳定等待时间计数 B 结束后，ADM0 寄存器的 ADCS 位被设置为 1，A/D 转换开始。
	<7>	模数转换开始	—
	<8>	A/D转换结束	将生成A/D转换结束中断（ADC10_ADI）。ADISS设置为1后，初始转换结果将无法使用。
	<9>	ADCS 位设置	ADM0 寄存器的 ADCS 位设置为 1，A/D 转换开始。
	<10>	模数转换开始	—
	<11>	A/D转换结束	生成模数转换结束中断（ADC10_ADI）。*1
	<12>	转换结果的存储方式 ADCR 或 ADCRH 登记册	转换结果存储在 ADCR 或 ADCRH 寄存器中。

注1. 根据 ADRCK 位、ADUL 和 ADLL 寄存器的设置，有可能不会产生中断信号。在这种情况下，结果不会存储在 ADCR 或 ADCRH 寄存器中。

22.6.5 设置测试模式

表 22.17 显示了测试模式下的设置步骤。

Table 22.17 Setting up test mode

Step		Process	Detail
Setting up Test Mode	<1>	MSTPCRD register setting	The MSTPD16 bit of the MSTPCRD register is set to 0, and supplying the clock starts.
	<2>	● ADM0 register setting ● ADM1 register setting ● ADM2 register setting ● ADUL and ADLL register setting ● ADS register setting ● ADTES register setting (The order of the settings is irrelevant.)	● ADM0 register FR[2:0], LV[1:0]: bits: These are used to specify the A/D conversion time . ADMD bit: This is used to specify the select mode. ● ADM1 register ADTMD1 and ADTMD0 bits: These are used to specify the software trigger mode. ADSCM bit: This is used to specify the one-shot conversion mode. ● ADM2 register ADREFP[1:0] and ADREFM bits: These are used to select the reference voltage. ADRCK bit: This is used to select the range for the A/D conversion result comparison value for generating the interrupt signal to AREA 2. ADTYP bit: 10-bit, or 8-bit resolution ● ADUL and ADLL register These set ADUL to 0xFF and ADLL to 0x00 (initial values). ● ADS register ADS[4:0] bits: These are used to set to AN000. ● ADTES register ADTES[1:0] bits: VREFL0 or VREFH0.
	<5>	Reference voltage stabilization wait time count A	The reference voltage stabilization wait time count indicated by A below may be required if the values of the ADREFP[1:0] bits are changed. If the values of ADREFP[1:0] are changed to 10b, respectively: A = 5 μs A wait is not required if the values of ADREFP[1:0] are changed to 00b or 01b, respectively.
	<6>	ADCE bit setting	The ADCE bit of the ADM0 register is set to 1, and the A/D converter enters the standby state.
	<7>	Reference voltage stabilization wait time count B	Use software to control waiting until reference voltage stabilization wait time count B (1 μs) elapses.
	<8>	ADCS bit setting	After reference voltage stabilization wait time count B elapses, the ADCS bit of the ADM0 register is set to 1, and A/D conversion starts.
	<9>	Start of A/D conversion	—
	<10>	End of A/D conversion	The A/D conversion end interrupt (ADC10_ADI) is generated.*1
	<11>	Storage of conversion results in the ADCR or ADCRH register	The conversion results are stored in the ADCR or ADCRH register.

Note 1. Depending on the settings of the ADRCK bit, ADUL and ADLL registers, there is a possibility of no interrupt signal being generated. In this case, the results are not stored in the ADCR or ADCRH register.

Note: For the procedure for testing the A/D converter, see [section 22.8. Testing of the A/D Converter](#).

22.7 Snooze Mode Function

In Snooze mode, A/D conversion is triggered by inputting a hardware trigger in the Software Standby mode. Normally, A/D conversion is stopped while in the Software Standby mode, but, by using the Snooze mode function, A/D conversion can be performed without operating the CPU. This is effective for reducing the operating current.

22.7.1 A/D Conversion by Inputting a Hardware Trigger

In Snooze mode, A/D conversion is triggered by inputting a hardware trigger.

When performing A/D conversion by inputting a hardware trigger in Snooze mode, only the following conversion mode can be used:

- Hardware trigger wait mode (select mode, one-shot conversion mode)

表 22.17 设置测试模式

步		过程	细节
设置测试模式	<1>	MSTPCRD寄存器设置	MSTPCRD 寄存器的 MSTPD16 位设置为 0，开始提供时钟信号。
	<2>	●ADM0寄存器设置 ●ADM1寄存器设置 ●ADM2寄存器设置 ●ADUL 和 ADLL 寄存器设置 ●ADS寄存器设置 ●ADTES寄存器设置（设置顺序无关紧要。）	●ADM0寄存器 FR[2:0], LV[1:0]: bits: 这些用于指定 A/D 转换时间。 ADMD 位：用于指定选择模式。 ●ADM1 注册 ADTMD1 和 ADTMD0 位：用于指定软件触发模式。 ADSCM 位：用于指定一次性转换模式。 ●ADM2寄存器 ADREFP[1:0] 和 ADREFM 位：这些用于选择参考电压。 ADRCK 位：用于选择 A/D 转换结果比较值的范围，以便生成到区域 2 的中断信号。 ADTYP 位：10 位或 8 位分辨率 ●ADUL 和 ADLL 寄存器 这些设置将 ADUL 设置为 0xFF，将 ADLL 设置为 0x00（初始值）。 ●广告交易登记册 ADS[4:0] 位：这些位用于设置为 AN000。 ●ADTES 寄存器 ADTES[1:0] 位: VREFL0 或 VREFH0。
	<5>	参考电压稳定等待时间计数 A	如果 ADREFP[1:0] 位的值发生变化，则可能需要下面 A 指示的参考电压稳定等待时间计数。 如果 ADREFP[1:0] 的值分别更改为 10b: A =5 μs 如果 ADREFP[1:0] 的值分别更改为 00b 或 01b，则不需要等待。
	<6>	ADCE 位设置	ADM0 寄存器的 ADCE 位设置为 1，A/D 转换器进入待机状态。
	<7>	参考电压稳定等待时间计数 B	使用软件控制等待时间，直到参考电压稳定等待时间计数 B（1 μs）结束。
	<8>	ADCS 位设置	参考电压稳定等待时间计数 B 结束后，ADM0 寄存器的 ADCS 位被设置为 1，A/D 转换开始。
	<9>	模数转换开始	—
	<10>	A/D转换结束	生成模数转换结束中断（ADC10_ADI）。*1
	<11>	转换结果的存储方式 ADCR 或 ADCRH 登记册	转换结果存储在 ADCR 或 ADCRH 寄存器中。

注1. 根据 ADRCK 位、ADUL 和 ADLL 寄存器的设置，有可能不会产生中断信号。在这种情况下，结果不会存储在 ADCR 或 ADCRH 寄存器中。

注：有关 A/D 转换器的测试程序，请参阅第 22.8 节 A/D 转换器的测试。

22.7 贪睡模式功能

在休眠模式下，软件待机模式下通过输入硬件触发信号来触发A/D转换。通常情况下，软件待机模式下A/D转换会停止，但通过使用休眠模式功能，可以在不运行CPU的情况下执行A/D转换。这可以有效降低运行电流。

22.7.1 通过输入硬件触发信号进行A/D转换

在睡眠模式下，A/D 转换是通过输入硬件触发信号来触发的。

在休眠模式下通过输入硬件触发信号执行 A/D 转换时，只能使用以下转换模式：

- 硬件触发等待模式（选择模式、单次转换模式）

If the A/D conversion result range is specified using the ADUL and ADLL registers, A/D conversion results can be determined at a certain interval of time. Using this function enables power supply voltage monitoring and input key determination based on A/D inputs.

Note: The Snooze mode can only be specified when the high-speed on-chip oscillator clock is selected for PCLKB.

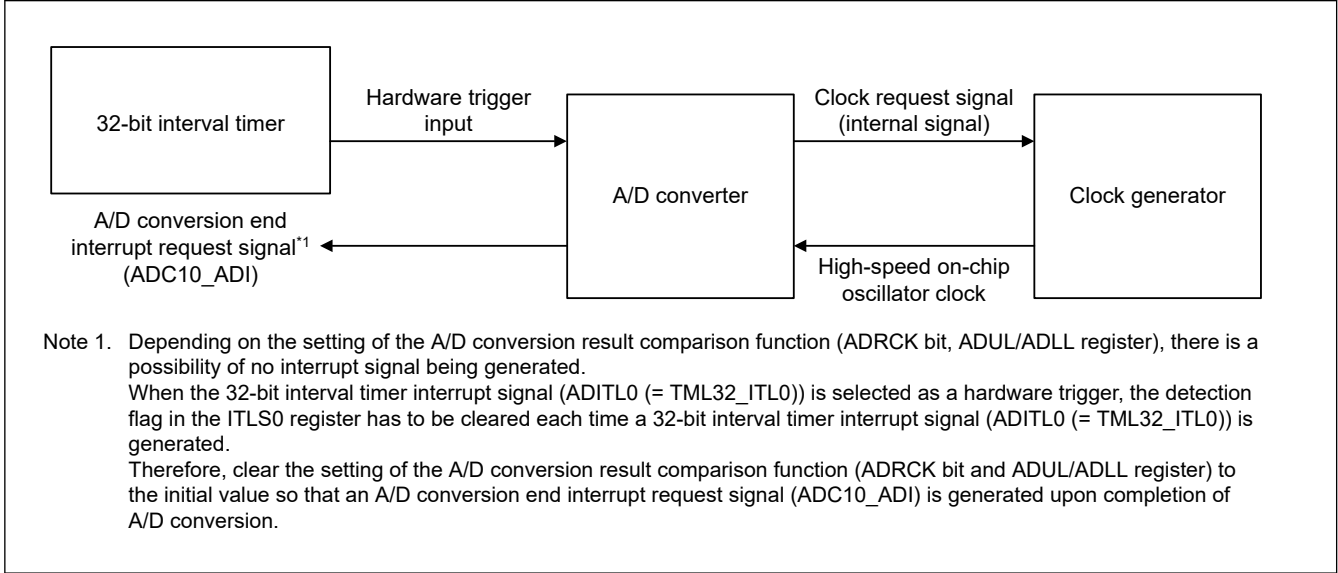


Figure 22.13 Block diagram when using Snooze mode in hardware trigger wait mode

When using the Snooze mode function, the initial setting of each register is specified before switching to the Software Standby mode (for details about these settings, see Table 22.18). Just before moving to Software Standby mode, set bit 2 (AWC) of A/D converter mode register 2 (ADM2) to 1. After the initial settings are specified, set bit 0 (ADCE) of A/D converter mode register 0 (ADM0) to 1.

If a hardware trigger is input after switching to the Software Standby mode, the high-speed on-chip oscillator clock is supplied to the A/D converter. After supplying this clock, the A/D converter automatically counts up to the A/D power supply stabilization wait time, and then A/D conversion starts.

The Snooze mode operation after A/D conversion ends differs depending on whether an interrupt signal is generated.*1

Note: Select the hardware trigger signal from 32-bit interval timer event signal (ADITL0 (= TML32_ITL0)).

(1) If an interrupt is generated after A/D conversion ends

If the A/D conversion result value is inside the range of values specified by the A/D conversion result comparison function (which is set up by using the ADRCK bit, ADUL and ADLL registers), the A/D conversion end interrupt request signal (ADC10_ADI) is generated.

- While in the select mode
When A/D conversion ends and an A/D conversion end interrupt request signal (ADC10_ADI) is generated, if SBYEDCR0.ADC10ED = 1, the A/D converter returns to normal operation mode from Snooze mode. At this time, be sure to clear bit 2 (AWC = 0: Snooze mode release) of the A/D converter mode register 2 (ADM2). If the AWC bit is left set to 1, A/D conversion will not start normally in the subsequent Snooze or normal operation mode.

Figure 22.14 shows an operation example when interrupt is generated after A/D conversion ends (while in select mode).

如果使用 ADUL 和 ADLL 寄存器指定 A/D 转换结果范围，则可以按一定时间间隔确定 A/D 转换结果。使用此功能可以基于 A/D 输入进行电源电压监控和输入键判断。

注意：只有当为 PCLKB 选择高速片上振荡器时钟时，才能指定睡眠模式。

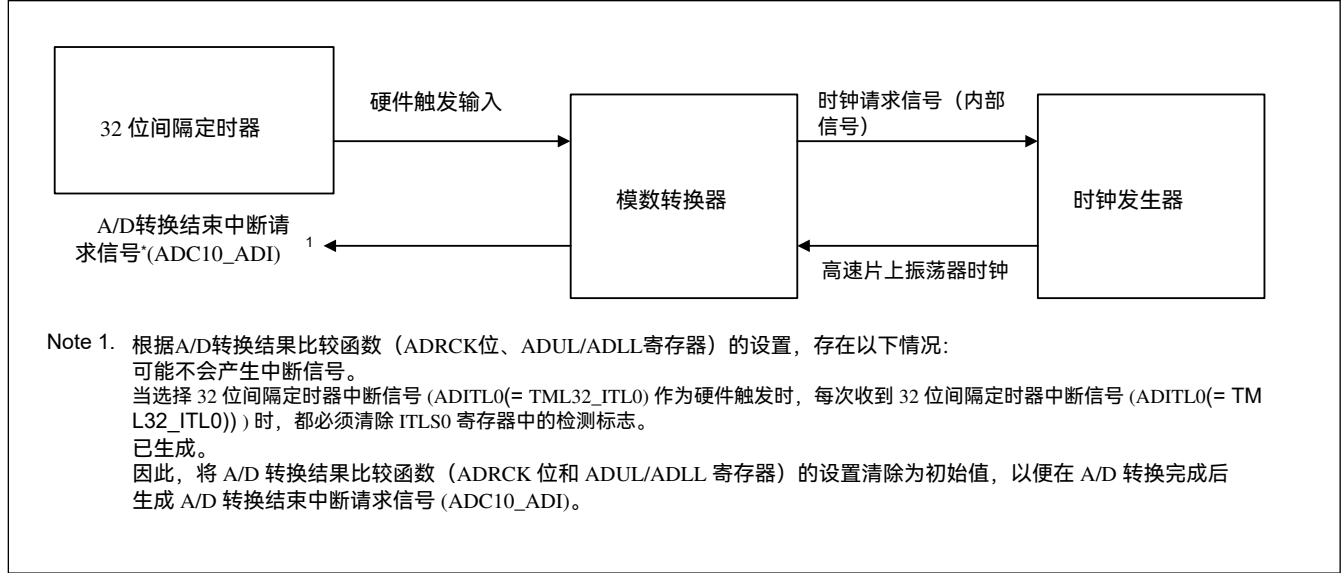


图 22.13 硬件触发等待模式下使用“稍后提醒”模式时的框图

使用休眠模式功能时，在切换到软件待机模式之前，需要指定每个寄存器的初始设置（有关这些设置的详细信息，请参见表 22.18）。在切换到软件待机模式之前，将转换器模式寄存器 2 (ADM2) 的第 2 位 (AWC) 设置为 1。指定初始设置后，将转换器模式寄存器 0 (ADM0) 的第 0 位 (ADCE) 设置为 1。

如果在切换到软件待机模式后输入硬件触发信号，则会向 A/D 转换器提供高速片上振荡器时钟。提供该时钟后，A/D 转换器会自动计数到 A/D 电源稳定等待时间，然后开始 A/D 转换。

A/D 转换结束后，睡眠模式的操作会根据是否产生中断信号而有所不同。*1

注意：从 32 位间隔定时器事件信号 (ADITL0 (= TML32_ITL0)) 中选择硬件触发信号。

(1) 如果在模数转换结束后产生中断

如果 A/D 转换结果值在 A/D 转换结果比较函数指定的值范围内（该函数由 ADRCK 位、ADUL 和 ADLL 寄存器设置），则会生成 A/D 转换结束中断请求信号 (ADC10_ADI)。

- 在选择模式下
当 A/D 转换结束并生成转换结束中断请求信号 (ADC10_ADI) 时，如果转换器从休眠模式返回到正常工作模式，则需要清除 A/D 转换器模式寄存器 2 (ADM2) 的第 2 位 (睡眠模式释放位)。如果 AWC 位保持为 1，则在后续的休眠模式或正常工作模式下，A/D 转换将无法启动。

图 22.14 显示了 A/D 转换结束后（在选择模式下）产生中断的操作示例。

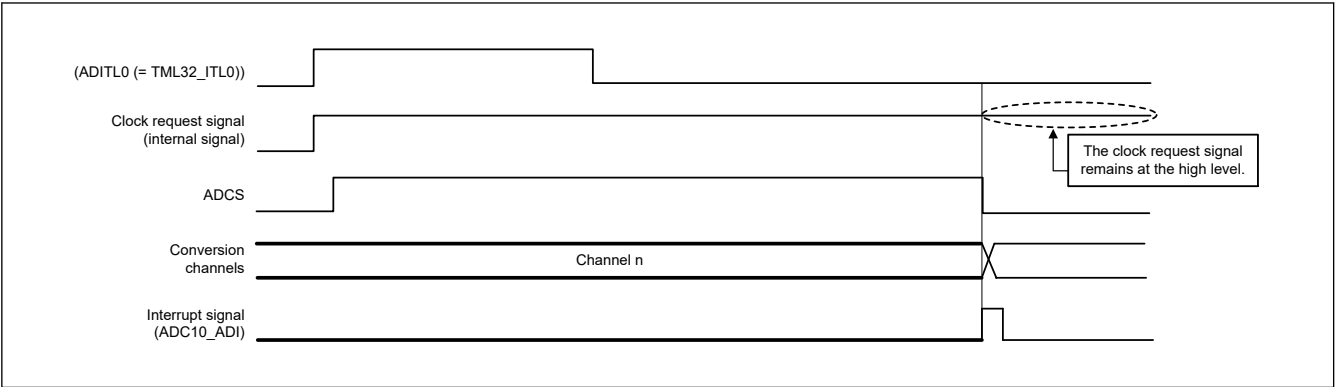


Figure 22.14 Operation example when interrupt is generated after A/D conversion ends (while in select mode)

(2) If no interrupt is generated after A/D conversion ends

If the A/D conversion result value is outside the range of values specified by the A/D conversion result comparison function (which is set up by using the ADRCK bit, ADUL and ADLL registers), the A/D conversion end interrupt request signal (ADC10_ADI) is not generated.

- While in the select mode
If the A/D conversion end interrupt request signal (ADC10_ADI) is not generated after A/D conversion ends, the clock request signal (an internal signal) is automatically set to the low level, and the clock is stopped. If a hardware trigger is input later, A/D conversion is performed again in the Snooze mode.

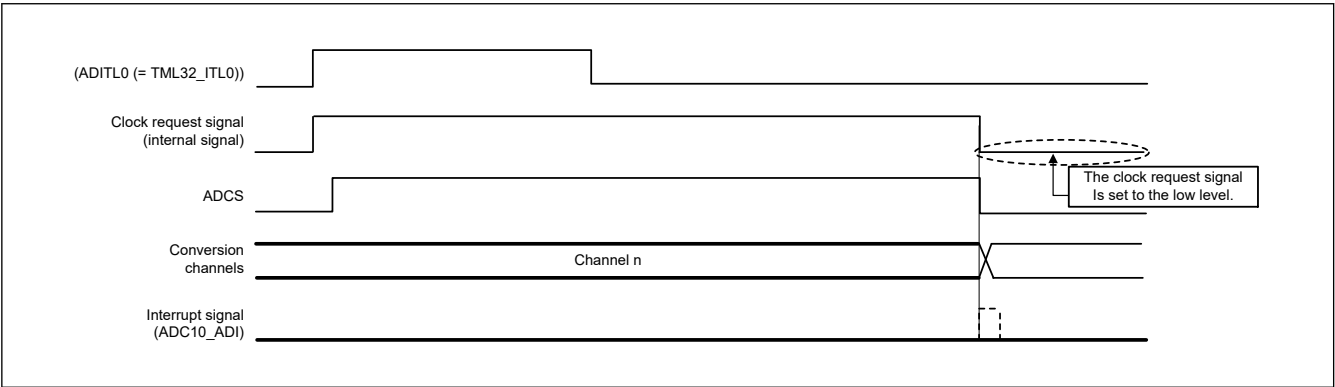


Figure 22.15 Operation example when no interrupt is generated after A/D conversion ends (while in select mode)

Table 22.18 shows procedure for setting up Snooze mode (hardware trigger)

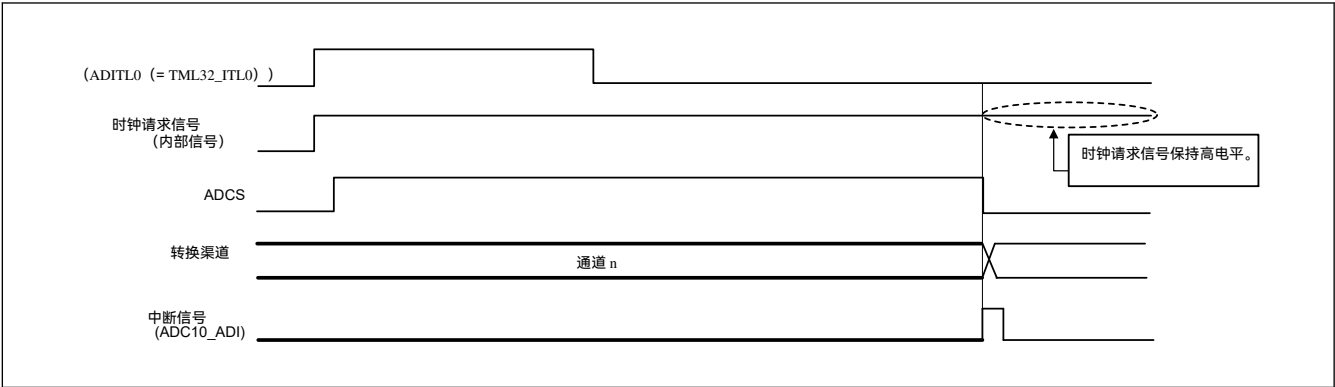


图 22.14 A/D 转换结束后产生中断的操作示例（在选择模式下）

(2) 如果模数转换结束后没有产生中断

如果A/D转换结果值超出A/D转换结果比较函数（通过使用 ADRCK 位、ADUL 和 ADLL 寄存器设置）指定的值范围，则不会生成A/D转换结束中断请求信号 (ADC10_ADI)。

- 在选择模式下
如果在A/D转换结束后未生成A/D转换结束中断请求信号 (ADC10_ADI)，则时钟请求信号（内部信号）会自动置低，时钟停止。如果之后输入硬件触发信号，则会在休眠模式下再次执行A/D转换。

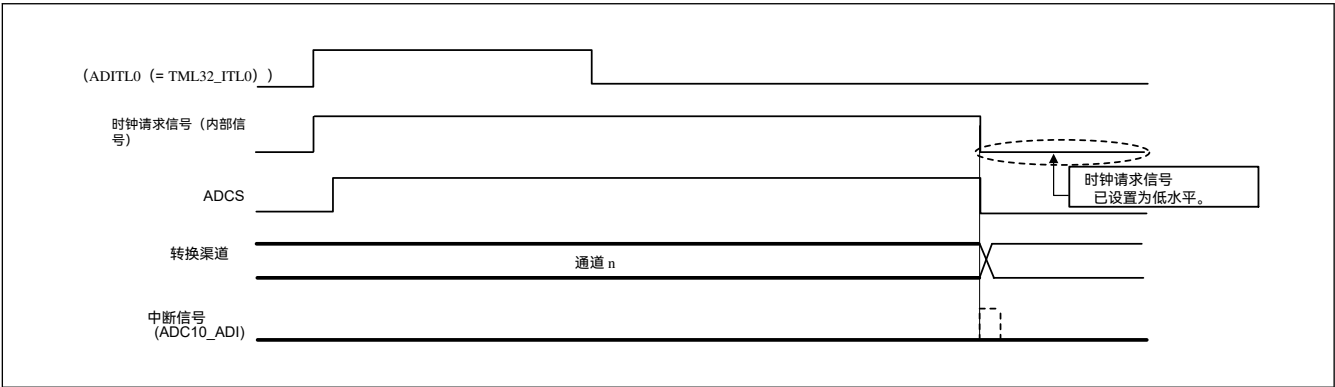


图 22.15 A/D 转换结束后未产生中断的操作示例（在选择模式下）

表 22.18 显示了设置贪睡模式（硬件触发）的步骤

Table 22.18 Procedure for setting up Snooze mode (hardware trigger)

Step		Process	Detail
Normal operation	<1>	MSTPCRD register setting	The MSTPD16 bit of the MSTPCRD register is set to 0, and supplying the clock starts.
	<2>	PmnPFS_A register settings	The ports are set as the analog input. (See section 15.5.4. Notes on Using Analog Functions.)
	<3>	- ADM0 register setting - ADM1 register setting - ADM2 register setting - ADUL and ADLL register setting - ADS register setting (The order of the settings is irrelevant.)	- ADM0 register FR[2:0], LV[1:0]: bits: These are used to specify the A/D conversion time. - ADMD bit: Select mode - ADM1 register ADTMD1 and ADTMD0 bits: These are used to specify the hardware trigger wait mode. - ADSCM bit: One-shot conversion mode - ADM2 register ADREFP[1:0] and ADREFM bits: These are used to select the reference voltage. - ADRCK bit: This is used to select the range for the A/D conversion result comparison value for generating the interrupt signal from AREA1, AREA3, and AREA 2. - ADTYP bit: 10-bit, or 8-bit resolution - ADUL and ADLL register These are used to specify the upper limit and lower limit A/D conversion result comparison values. - ADS register ADS[4:0] bits: These are used to select the analog input channels.
	<4>	Reference voltage stabilization wait time count A	The reference voltage stabilization wait time count indicated by A below may be required if the values of the ADREFP[1:0] bits are changed. If the values of ADREFP[1:0] are changed to 10b, respectively: A = 5 μs A wait is not required if the values of ADREFP[1:0] are changed to 00b or 01b, respectively.
	<5>	AWC = 1	Immediately before entering the Software Standby mode, enable the Snooze mode by setting the AWC bit of the ADM2 register to 1.
	<6>	Normal operation	The ADCE bit of the ADM0 register is set to 1, and the A/D converter enters the standby state.
Software Standby mode	<7>	Enter the Software Standby mode	—
Snooze mode	<8>	Hardware trigger generation	After hardware trigger is generated, the A/D converter automatically counts up to the stabilization wait time for A/D power supply and A/D conversion is started in the Snooze mode.
	—	:	(The A/D conversion operations are performed)
	<9>	End of A/D conversion	The A/D conversion end interrupt (ADC10_ADI) is generated.*1
	<10>	ADC10_ADI generation	- ADC10_ADI is generated : Go to step <11> - ADC10_ADI is not generated : The clock request signal (an internal signal) is automatically set to the low level in the Snooze mode. And go to step <7>.
Normal operation	<11>	Storage of conversion results in the ADCR or ADCRH register	The conversion results are stored in the ADCR or ADCRH register.
	<12>	AWC = 0	Release the Snooze mode by clearing the AWC bit of the ADM2 register to 0.*2
	<13>	Normal operation :	—

Note 1. If the A/D conversion end interrupt request signal (ADC10_ADI) is not generated depending on the settings of the ADRCK bit, ADUL and ADLL registers, the result is not stored in the ADCR or ADCRH register. The A/D converter enters the Software Standby mode again. If a hardware trigger is input later, A/D conversion operation is again performed in the Snooze mode.

Note 2. If the AWC bit is left set to 1, A/D conversion will not start normally in the subsequent Snooze or normal operation mode. Be sure to clear the AWC bit to 0.

表 22.18 设置贪睡模式（硬件触发）的步骤

步		过程	细节
正常运行	<1>	MSTPCRD寄存器设置	MSTPCRD 寄存器的 MSTPD16 位设置为 0，开始提供时钟信号。
	<2>	PmnPFS_A 寄存器设置	端口设置为模拟输入。 (参见第 15.5.4 节“模拟函数使用注意事项”。)
	<3>	●ADM0寄存器设置 ●ADM1寄存器设置 ●ADM2寄存器设置 ●ADUL 和 ADLL 寄存器设置 ●ADS寄存器设置（设置顺序无关紧要。）	●ADM0寄存器 FR[2:0], LV[1:0]: 位：这些用于指定 A/D 转换时间。 - ADMD 位：选择模式 ●ADM1 注册 ADTMD1 和 ADTMD0 位：用于指定硬件触发等待模式。 - ADSCM 位：单次转换模式 ●ADM2寄存器 ADREFP[1:0] 和 ADREFM 位：这些用于选择参考电压。 - ADRCK 位：用于选择 A/D 转换结果比较值的范围，以便从 AR EA1、AREA3 和 AREA 2 生成中断信号。 - ADTYP 位：10 位或 8 位分辨率 ●ADUL 和 ADLL 寄存器 这些用于指定 A/D 转换结果比较值的上限和下限。 ●广告交易登记册 ADS[4:0] 位：这些位用于选择模拟输入通道。
	<4>	参考电压稳定等待时间计数 A	如果 ADREFP[1:0] 位的值发生变化，则可能需要下面 A 指示的参考电压稳定等待时间计数。 如果 ADREFP[1:0] 的值分别更改为 10b: A = 5 μs 如果 ADREFP[1:0] 的值分别更改为 00b 或 01b，则不需要等待。
	<5>	AWC = 1	在进入软件待机模式之前，将 ADM2 寄存器的 AWC 位设置为 1，启用休眠模式。
	<6>	正常运行	ADM0 寄存器的 ADCE 位设置为 1，A/D 转换器进入待机状态。
软件待机模式	<7>	进入软件待机模式	—
贪睡模式	<8>	硬件触发器生成	硬件触发后，A/D转换器会自动计数到A/D电源的稳定等待时间，然后A/D转换以休眠模式开始。
	—	:	(执行模数转换操作)
	<9>	A/D转换结束	生成模数转换结束中断（ADC10_ADI）。*1
	<10>	ADC10_ADI 生成	●ADC10_ADI 已生成：转至步骤<11> ●未生成ADC10_ADI: 在贪睡模式下，时钟请求信号（内部信号）自动设置为低电平。然后转到步骤<7>。
正常运行	<11>	转换结果的存储方式 ADCR 或 ADCRH 登记册	转换结果存储在 ADCR 或 ADCRH 寄存器中。
	<12>	AWC = 0	通过将 ADM2 寄存器的 AWC 位清除为0.*2来解除休眠模式
	<13>	正常运行 :	—

注1：如果根据ADRCK位、ADUL和ADLL寄存器的设置，未生成A/D转换结束中断请求信号（ADC10_ADI），则结果不会存储在ADCR或ADCRH寄存器中。A/D转换器将再次进入软件待机模式。如果之后输入硬件触发信号，则A/D转换操作将再次以休眠模式执行。

注 2. 如果 AWC 位设置为 1，则在后续的睡眠或正常操作模式下，A/D 转换将不会正常启动。务必将 AWC 位清除为 0。

22.8 Testing of the A/D Converter

The IEC60730 standard mandates testing of the A/D converter. This test checks whether the A/D converter is operating normally by converting the positive and negative reference voltages, analog input channels (ANxxx), temperature sensor output voltage, and internal reference voltage of the A/D converter.

Use the following procedure to check the analog multiplexer.

<1> Select the ANxxx pin for A/D conversion using the ADTES register (ADTES1 = 0, ADTES0 = 0).

<2> Perform A/D conversion for the ANxxx pin (conversion result 1-1).

<3> Select the negative reference voltage of the A/D converter for A/D conversion using the ADTES register (ADTES1 = 1, ADTES0 = 0).

<4> Perform A/D conversion of the negative reference voltage of the A/D converter (conversion result 2-1).

<5> Select the ANxxx pin for A/D conversion using the ADTES register (ADTES1 = 0, ADTES0 = 0).

<6> Perform A/D conversion for the ANxxx pin (conversion result 1-2).

<7> Select the positive reference voltage of the A/D converter for A/D conversion using the ADTES register (ADTES1 = 1, ADTES0 = 1).

<8> Perform A/D conversion of the positive reference voltage of the A/D converter (conversion result 2-2).

<9> Select the ANxxx pin for A/D conversion using the ADTES register (ADTES1 = 0, ADTES0 = 0).

<10>Perform A/D conversion for the ANxxx pin (conversion result 1-3).

<11>Check that the conversion results 1-1, 1-2, and 1-3 are equal.

<12>Check that the A/D conversion result 2-1 is all 0 and the A/D conversion result 2-2 is all 1.

Using the procedure above can confirm that the analog multiplexer is selected and all wiring is connected.

Note: If the analog input voltage is variable during A/D conversion in steps <1> to <10> above, use another method to check the analog multiplexer.

Note: The results of conversion might include an error. Consider an appropriate level of error in comparison of the results of conversion.

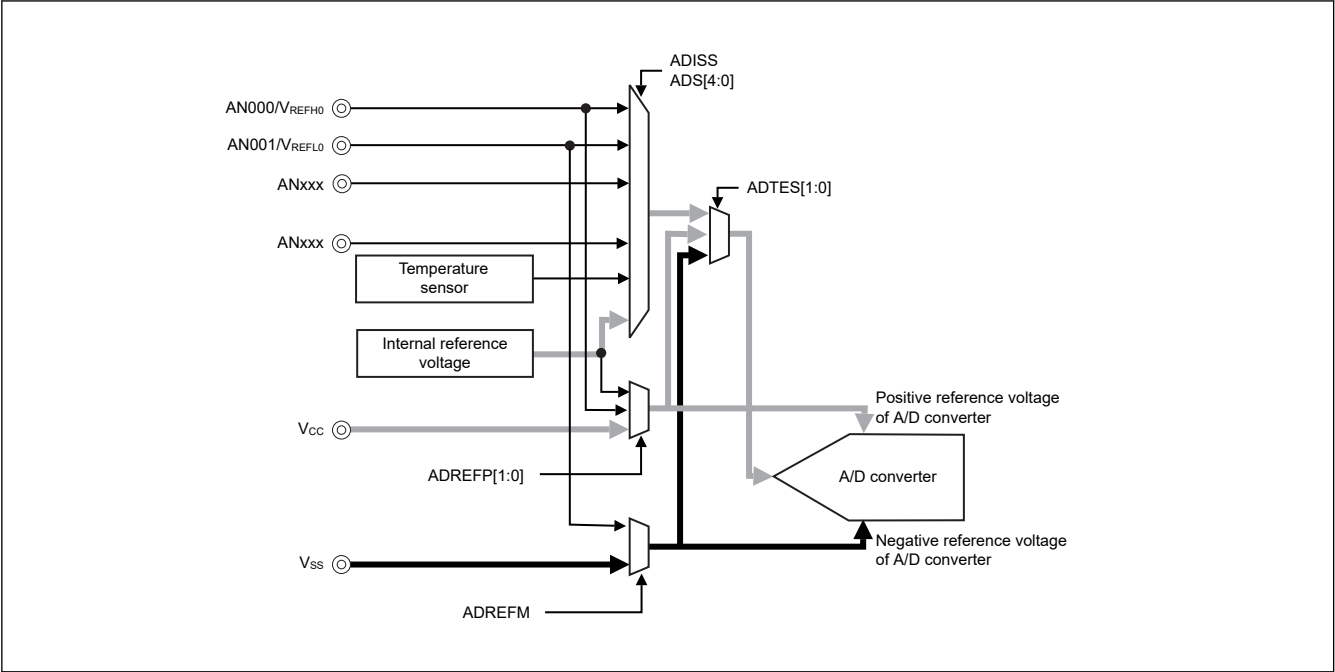


Figure 22.16 Configuration of testing of the A/D converter

22.9 How to Read A/D Converter Characteristics Table

This section describes special terms that are unique to the A/D converter.

22.8 A/D转换器的测试

IEC60730 标准规定了 A/D 转换器的测试。该测试通过转换正负参考电压、模拟输入通道 (ANxxx)、温度传感器输出电压和A/D 转换器的内部参考电压来检查A/D 转换器是否正常工作。

使用以下步骤检查模拟多路复用器。

<1>选择 ANxxx 引脚进行A/D使用 ADTES 寄存器进行转换(ADTES1 = 0,ADTES0 = 0),

对 ANxxx 引脚执行<2>转换（转换结果 1-1）。

<3>使用 ADTES 寄存器选择 A/D 转换器的负参考电压进行 A/D 转换(ADTES1 =1,ADTES0 = 0)。

<4>对A/D转换器的负参考电压执行 A/D 转换（转换结果 2-1）。

<5>使用 ADTES 寄存器选择 ANxxx 引脚进行 A/D 转换(ADTES1 = 0,ADTES0 = 0)

<6>对 ANxxx 引脚执行 A/D 转换（转换结果 1-2）。

<7>使用 ADTES 寄存器选择 A/D 转换器的正参考电压进行 A/D 转换(ADTES1 = 1,ADTES0 = 1),

<8>执行A/DA/D 转换器的正参考电压转换（转换结果 2-2）。

<9>使用 ADTES 寄存器选择 ANxxx 引脚进行 A/D 转换(ADTES1 = 0,ADTES0 = 0)

<10>对 ANxxx 引脚执行 A/D 转换（转换结果 1-3）。

<11>检查转换结果 1-1、1-2 和 1-3 是否相等。

<12>糟糕的是A/D转换结果 2-1 全部为 0，而A/D转换结果 2-2 全部为 1。

使用上述步骤可以确认已选择模拟多路复用器并且所有线路均已连接。

注意：如果在 A/D 转换过程中，模拟输入电压在上述步骤<1>到<10>中发生变化，则应使用另一种方法来检查模拟多路复用器。

注意：转换结果可能存在误差。在比较转换结果时，请考虑适当的误差范围。

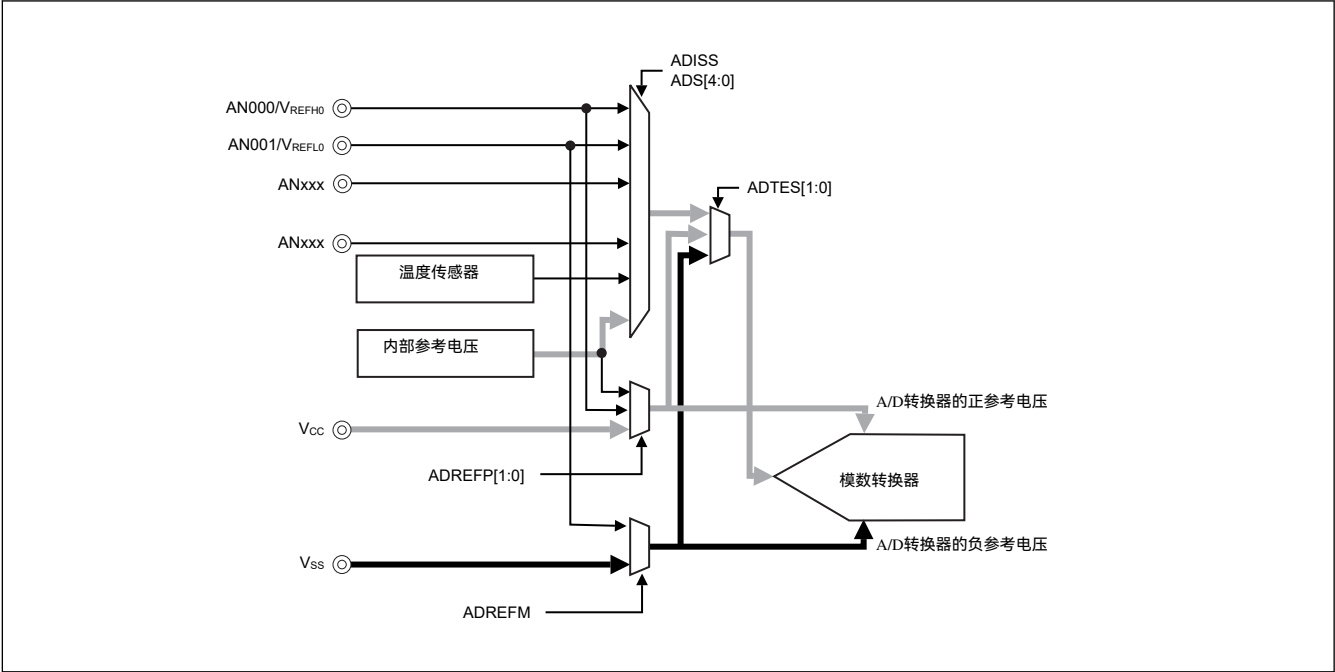


图 22.16 A/D 转换器测试配置

22.9 如何解读模数转换器特性表

本节介绍A/D 转换器特有的特殊术语。

(1) Resolution

This is the minimum analog input voltage that can be identified. That is, the percentage of the analog input voltage per bit of digital output is called 1 LSB (Least Significant Bit). The percentage of 1 LSB with respect to the full scale is expressed by %FSR (Full Scale Range).

1 LSB is as follows when the resolution is 10 bits.

$$1 \text{ LSB} = 1/2^{10} = 1/1024$$
$$\approx 0.098 \% \text{FSR}$$

Accuracy has no relation to resolution, but is determined by overall error.

(2) Overall error

This shows the maximum error value between the actual measured value and the theoretical value.

Zero-scale error, full-scale error, integral linearity error, differential linearity errors, and combinations of these express the overall error.

Note that the quantization error is not included in the overall error in the characteristics table.

(3) Quantization error

When analog values are converted to digital values, a $\pm 1/2\text{LSB}$ error naturally occurs. In an A/D converter, an analog input voltage in a range of $\pm 1/2\text{LSB}$ is converted to the same digital code, so a quantization error cannot be avoided.

Note that the quantization error is not included in the overall error, zero-scale error, full-scale error, integral linearity error, and differential linearity error in the characteristics table.

Figure 22.17 shows the overall error, and Figure 22.18 shows the quantization error.

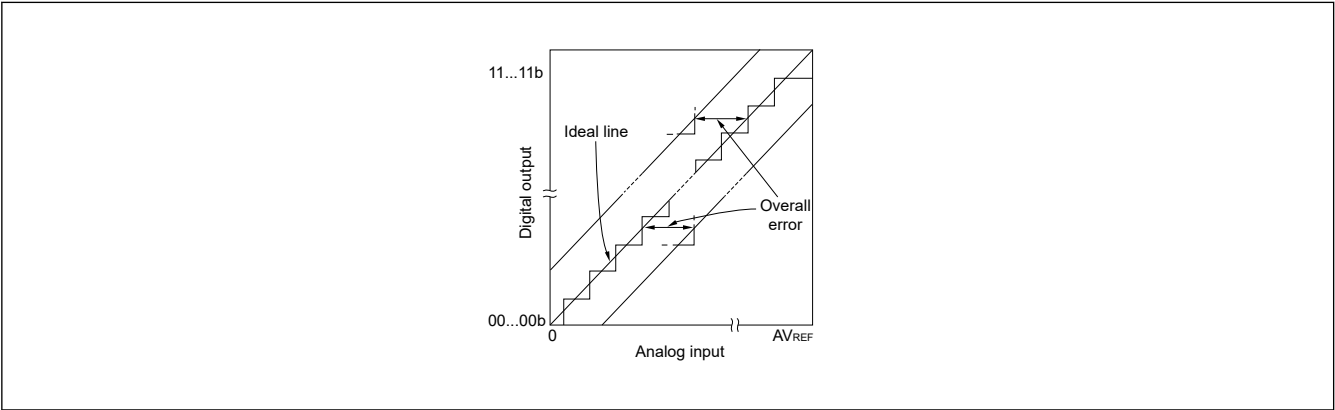


Figure 22.17 Overall error

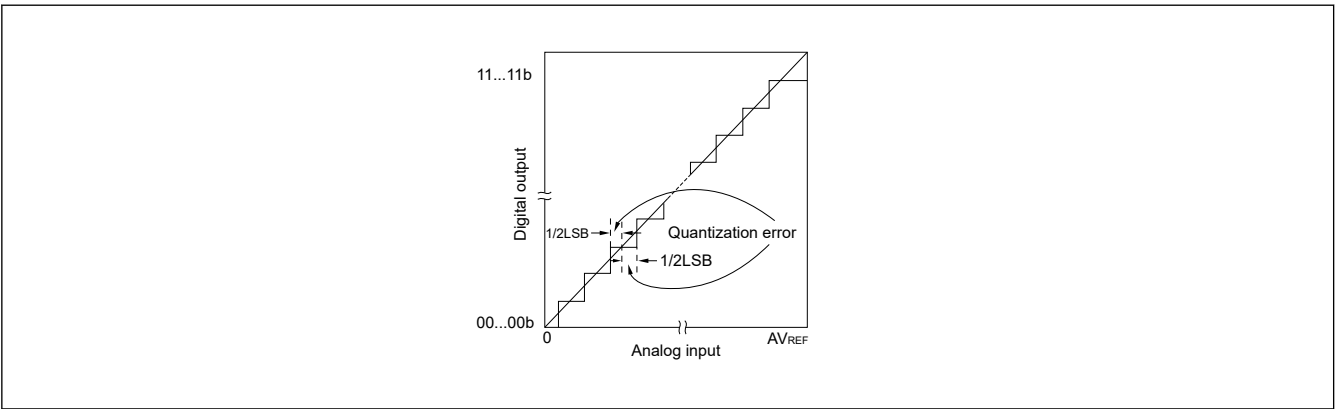


Figure 22.18 Quantization error

(1) 决议

这是可以识别的最小模拟输入电压。也就是说，每一位数字输出所对应的模拟输入电压百分比称为 1 LSB（最低有效位）。1 LSB 相对于满量程的百分比用 %FSR（满量程范围）表示。

当分辨率为 10 位时，1 LSB 如下所示。

$$1 \text{ LSB} = 1/2^{10} = 1/1024$$
$$\approx 0.098 \% \text{FSR}$$

准确度与分辨率无关，而是由总体误差决定。

(2) 总体误差

这显示了实际测量值与理论值之间的最大误差值。

零刻度误差、满刻度误差、积分线性误差、微分线性误差以及它们的组合表示总体误差。

请注意，量化误差不包含在特性表中的总误差中。

(3) 量化误差

当模拟值转换为数字值时，自然会产生 $\pm 1/2\text{LSB}$ 误差。在 A/D 转换器中， $\pm 1/2\text{LSB}$ 范围内的模拟输入电压会被转换为相同的数字代码，因此量化误差无法避免。

请注意，特性表中的总误差、零刻度误差、满刻度误差、积分线性误差和微分线性误差中不包含量化误差。

图 22.17 显示了总体误差，图 22.18 显示了量化误差。

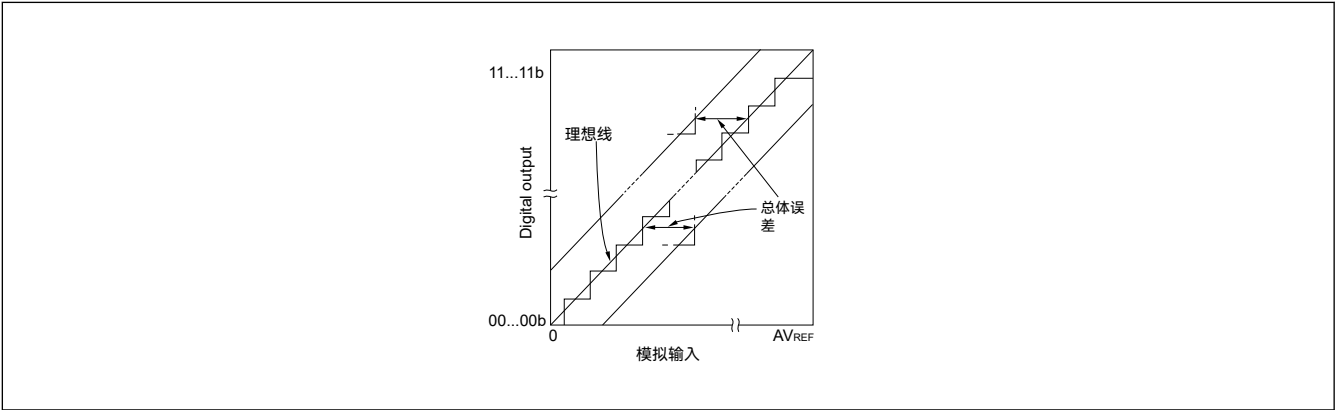


图 22.17 总体误差

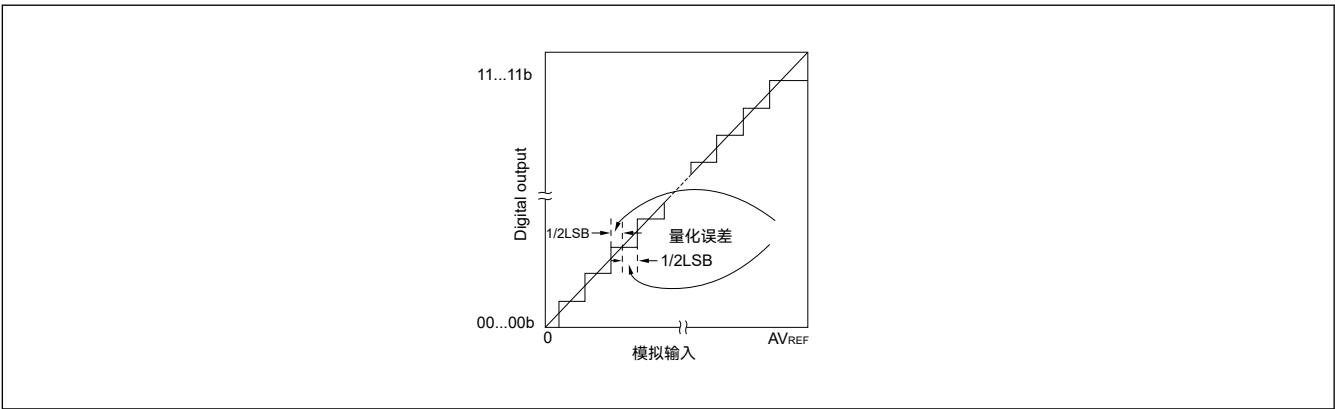


图 22.18 量化误差

(4) Zero-scale error

This shows the difference between the actual measurement value of the analog input voltage and the theoretical value (1/2LSB) when the digital output changes from 0.....000b to 0.....001b.

If the actual measurement value is greater than the theoretical value, it shows the difference between the actual measurement value of the analog input voltage and the theoretical value (3/2LSB) when the digital output changes from 0.....001b to 0.....010b.

(5) Full-scale error

This shows the difference between the actual measurement value of the analog input voltage and the theoretical value (Full-scale – 3/2LSB) when the digital output changes from 1.....110b to 1.....111b.

(6) Integral linearity error

This shows the degree to which the conversion characteristics deviate from the ideal linear relationship. It expresses the maximum value of the difference between the actual measurement value and the ideal straight line when the zero-scale error and full-scale error are 0.

(7) Differential linearity error

While the ideal width of code output is 1 LSB, this indicates the difference between the actual measurement value and the ideal value of the width of output code.

For differential inputs, the zero-scale error is shown in [Figure 22.19](#), the full-scale error is shown in [Figure 22.20](#), the integral linearity error is shown in [Figure 22.21](#), and the differential linearity error is shown in [Figure 22.22](#).

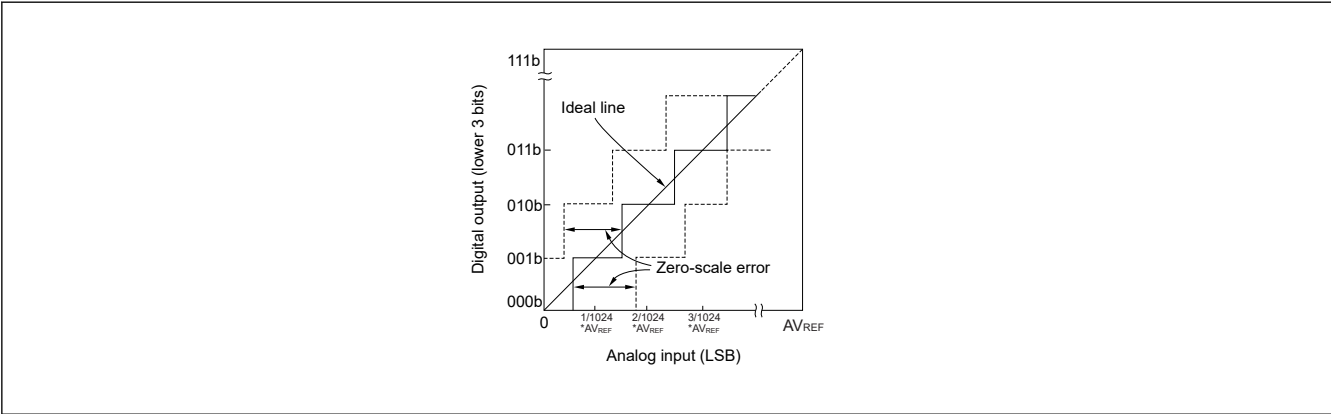


Figure 22.19 Zero-scale error

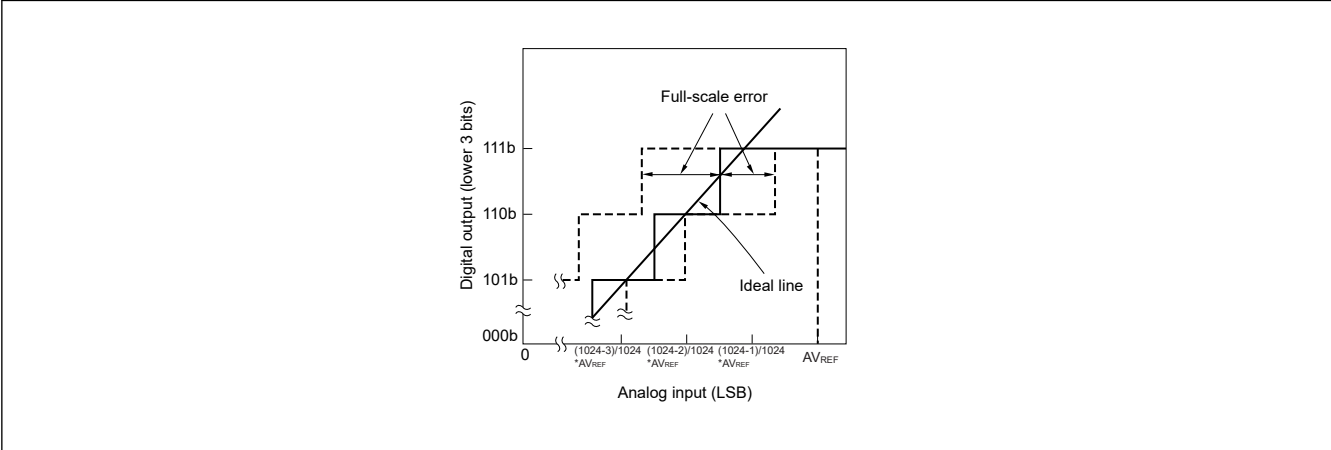


Figure 22.20 Full-scale error

(4) 零尺度误差

这显示了当数字输出从 0.....000b 变为 0.....001b 时，模拟输入电压的实际测量值与理论值（1/2LSB）之间的差异。

如果实际测量值大于理论值，则显示当数字输出从 0.....001b 变为 0.....010b 时，模拟输入电压的实际测量值与理论值 (3/2LSB) 之间的差异。

(5) 满量程误差

这显示了当数字输出从 1.....110b 变为 1.....111b 时，模拟输入电压的实际测量值与理论值（满量程 - 3/2LSB）之间的差异。

(6) 积分线性误差

这显示了转换特性偏离理想线性关系的程度。它表示当零点误差和满量程误差均为0时，实际测量值与理想直线之间差异的最大值。

(7) 微分线性误差

虽然理想的代码输出宽度为 1 LSB，但这表示实际测量值与理想输出代码宽度值之间的差异。

对于微分输入，零尺度误差如图 22.19 所示，满尺度误差如图 22.20 所示，积分线性误差如图 22.21 所示，微分线性误差如图 22.22 所示。

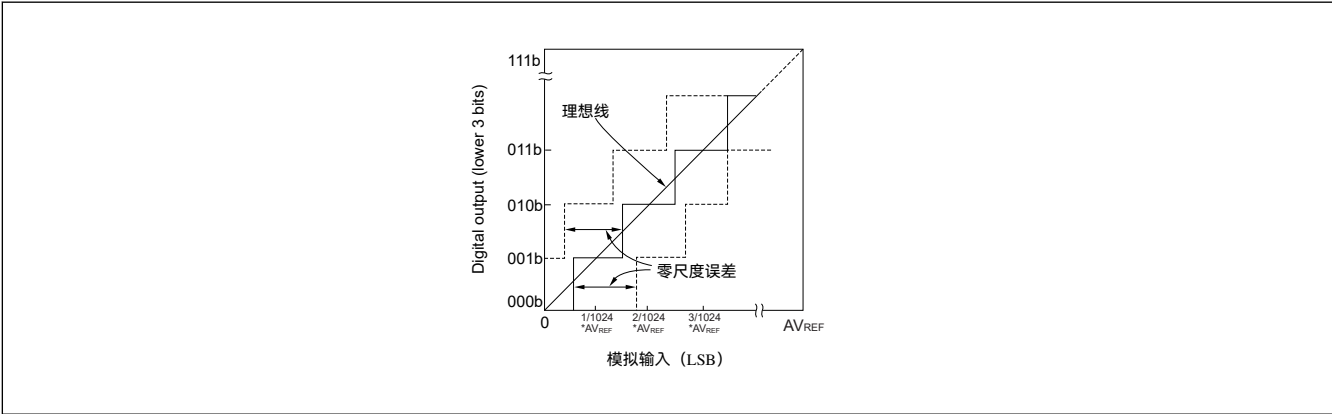


图 22.19 零尺度误差

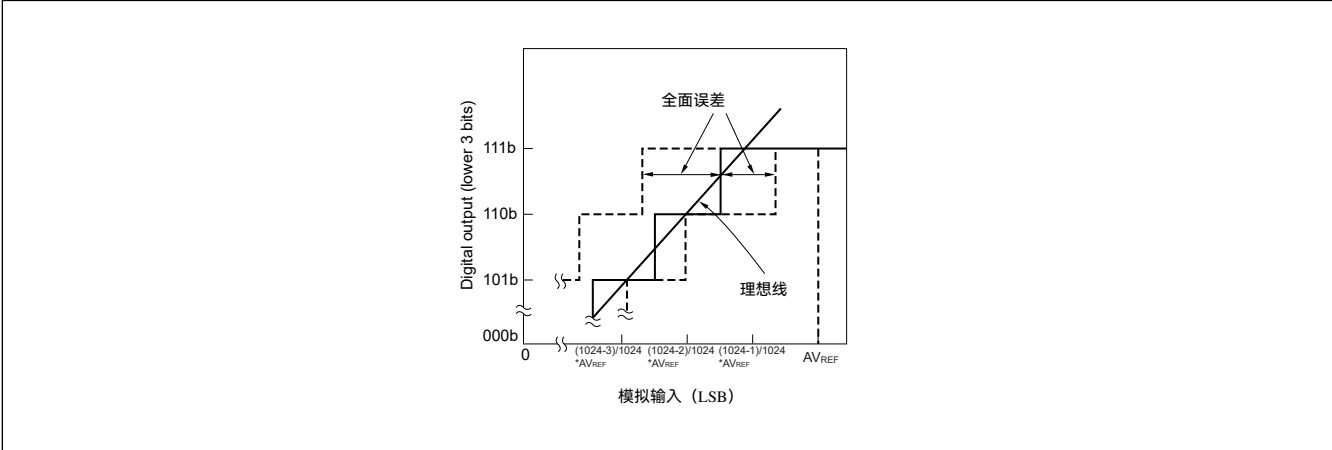


图 22.20 满量程误差

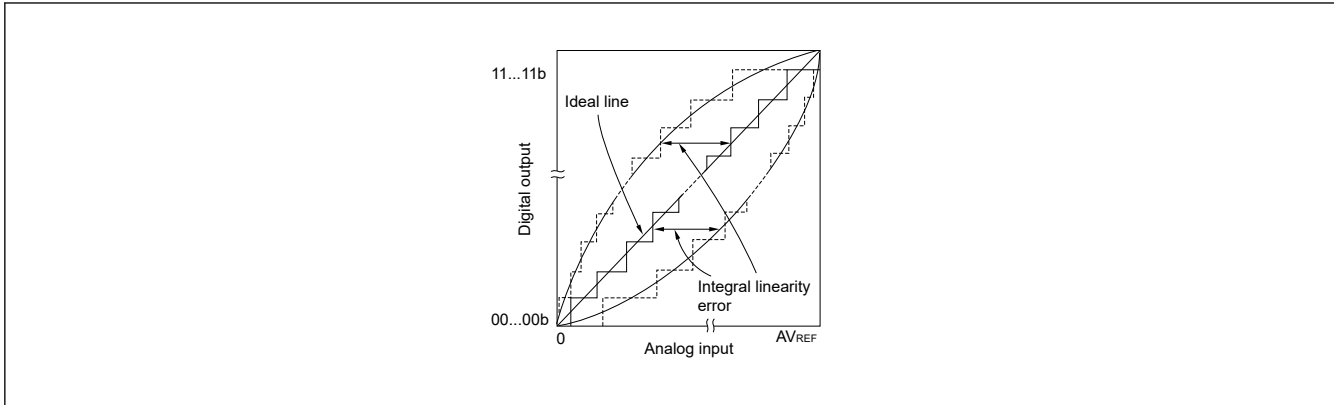


Figure 22.21 Integral linearity error

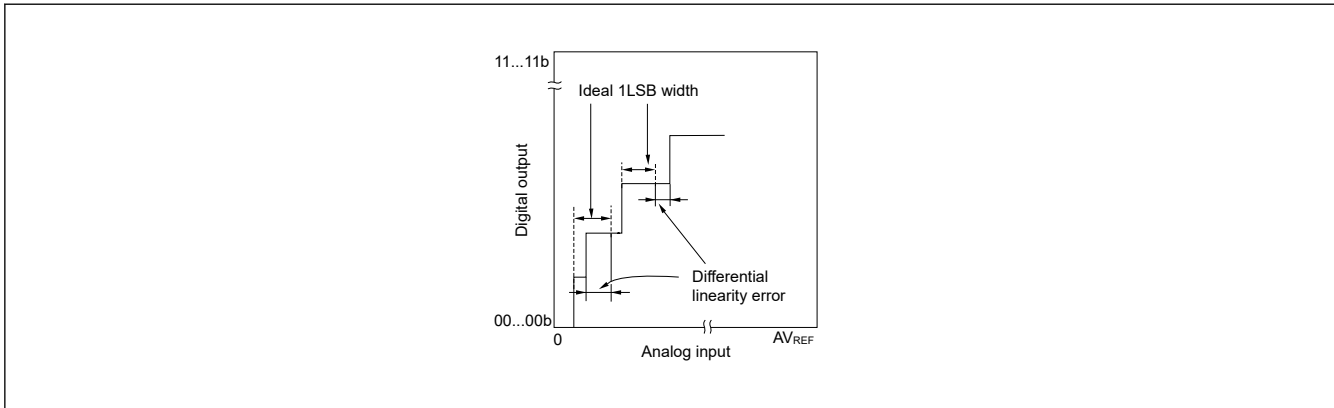


Figure 22.22 Differential linearity error

(8) Conversion time

This expresses the time from the start of sampling to when the digital output is obtained. The sampling time is included in the conversion time in the characteristics table.

(9) Sampling time

This is the time the analog switch is turned on for the analog voltage to be sampled by the sample & hold circuit.

Figure 22.23 shows the sampling time at the A/D conversion time.

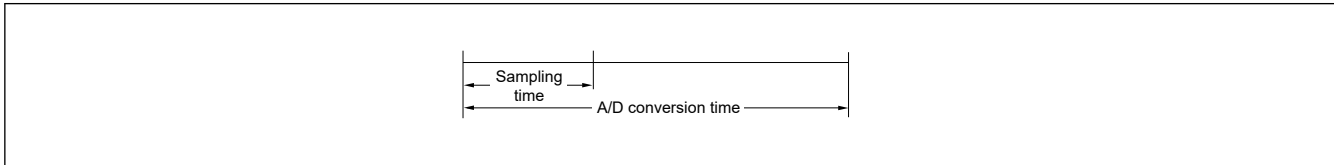


Figure 22.23 Sampling time at the A/D conversion time

22.10 Usage Notes

(1) Operating current in Software Standby mode

Shift to Software Standby mode after stopping the A/D converter (by setting bit 7 (ADCS) of A/D converter mode register 0 (ADM0) to 0). The operating current can be reduced by setting bit 0 (ADCE) of the ADM0 register to 0 at the same time.

To restart from the standby state, clear the ADC10_ADI bit in the corresponding NVIC_ICPR0 register and start operation.

(2) Input range of AN000, AN001, AN004, AN005, AN021 and AN022 pins

Observe the rated range of the AN000, AN001, AN004, AN005, AN021 and AN022 pins input voltage. If a voltage exceeding VCC and VREFH0 or a voltage lower than VSS and VREFL0 (even in the range of absolute maximum ratings)

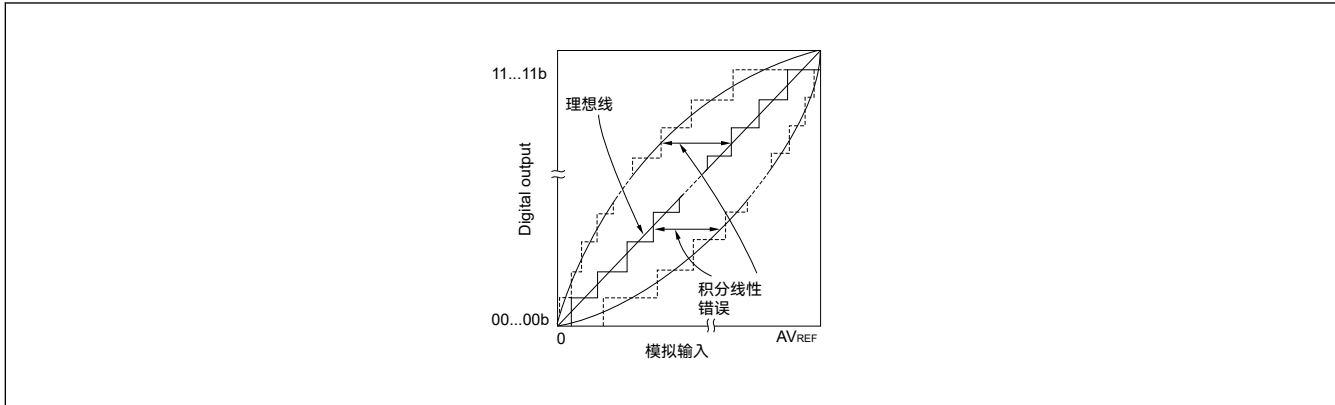


图 22.21 积分线性误差

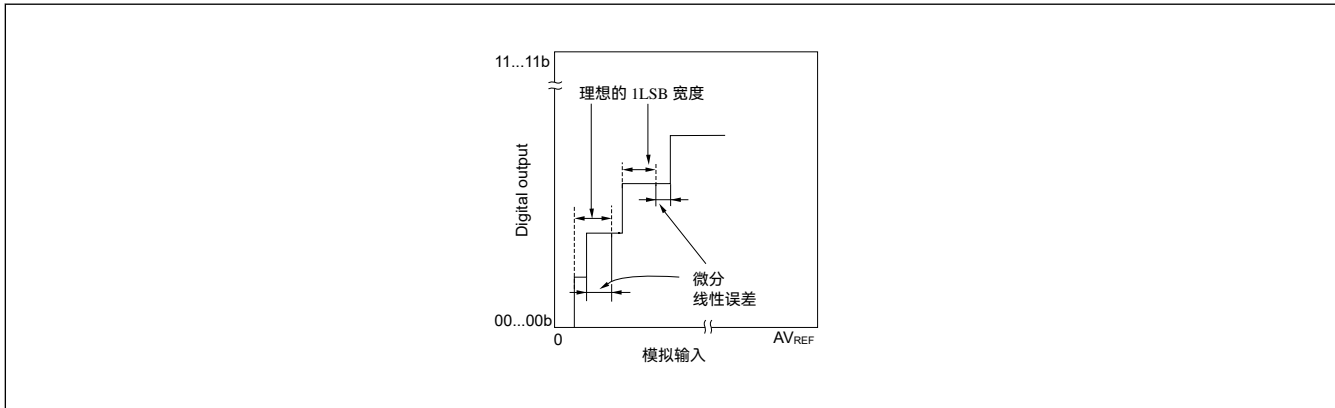


图 22.22 微分线性误差

(8) 转换时间

这表示从开始采样到获得数字输出所需的时间。采样时间已包含在特性表中的转换时间中。

(9) 采样时间

此时模拟开关打开，模拟电压将被采样保持电路采样。

图 22.23 显示了 A/D 转换时间的采样时间。

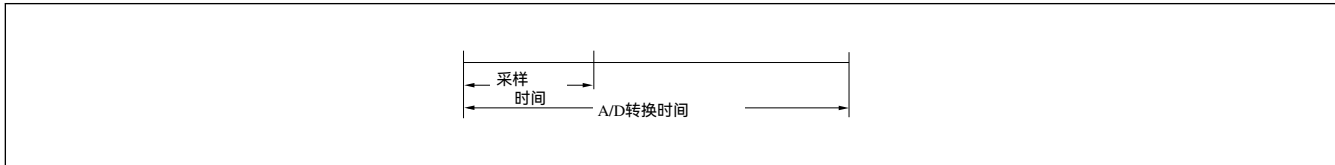


图 22.23 A/D 转换时的采样时间

22.10 使用说明

(1) 软件待机模式下的工作电流

停止模数转换器后（将转换器模式寄存器 0 (ADM0) 的第 7 位 (ADCS) 设置为 0），系统将进入软件待机模式。同时，将 ADM0 寄存器的第 0 位 (ADCE) 设置为 0 可以降低工作电流。要从此待机状态重新启动，请清除相应 NVIC_ICPR0 寄存器中的 ADC10_ADI 位并开始运行。

(2) AN000、AN001、AN004、AN005、AN021 和 AN022 引脚的输入范围

请注意 AN000、AN001、AN004、AN005、AN021 和 AN022 引脚输入电压的额定范围。如果输入到模拟输入通道的电压超过 VCC 和 VREFH0，或低于 VSS 和 VREFL0（即使在绝对最大额定值范围内），则该通道的转换值将变为未定义。

is input to an analog input channel, the converted value of that channel becomes undefined. In addition, the converted values of the other channels may also be affected.

When internal reference voltage is selected as the reference voltage for the ‘+’ side of the A/D converter, do not input a voltage equal to or higher than the internal reference voltage to a pin selected by the ADS register. However, it is no problem that a voltage equal to or higher than the internal reference voltage is input to a pin not selected by the ADS register.

Note: For details about the internal reference voltage, see [section 27, Electrical Characteristics](#).

(3) Conflicting operations

<1> Conflict between the conversion result being stored in the A/D conversion result register (ADCR or ADCRH) at the end of conversion and the read access to the ADCR or ADCRH register by instruction.

The ADCR or ADCRH register read has priority. After the read operation, the new conversion result is written to the ADCR or ADCRH registers.

<2> Conflict between the conversion result being stored in the A/D conversion result register (ADCR or ADCRH) at the end of conversion and the write access to the A/D converter mode register 0 (ADM0) or analog input channel specification register (ADS) by instruction.

The ADM0 and ADS registers write have priority. The ADCR and ADCRH registers write is not performed, nor is the conversion end interrupt signal (ADC10_ADI) generated.

(4) Noise countermeasures

To maintain the 10-bit resolution, attention must be paid to noise input to the VREFH0, VCC, AN000, AN001, AN004, AN005, AN021 and AN022 pins.

<1> Connect a capacitor with a low equivalent resistance and a good frequency response (capacitance of about 0.1 μF) via the shortest possible run of relatively thick wiring to the VCC and VREFH0 pins.

<2> The higher the output impedance of the analog input source, the greater the influence. To reduce the noise, connecting an external capacitor as shown in [Figure 22.24](#) is recommended.

<3> Do not switch these pins with other pins during conversion.

<4> The accuracy is improved if the Sleep mode is set immediately after the start of conversion.

[Figure 22.24](#) shows connections of VCC, VREFH0, and analog input pins.

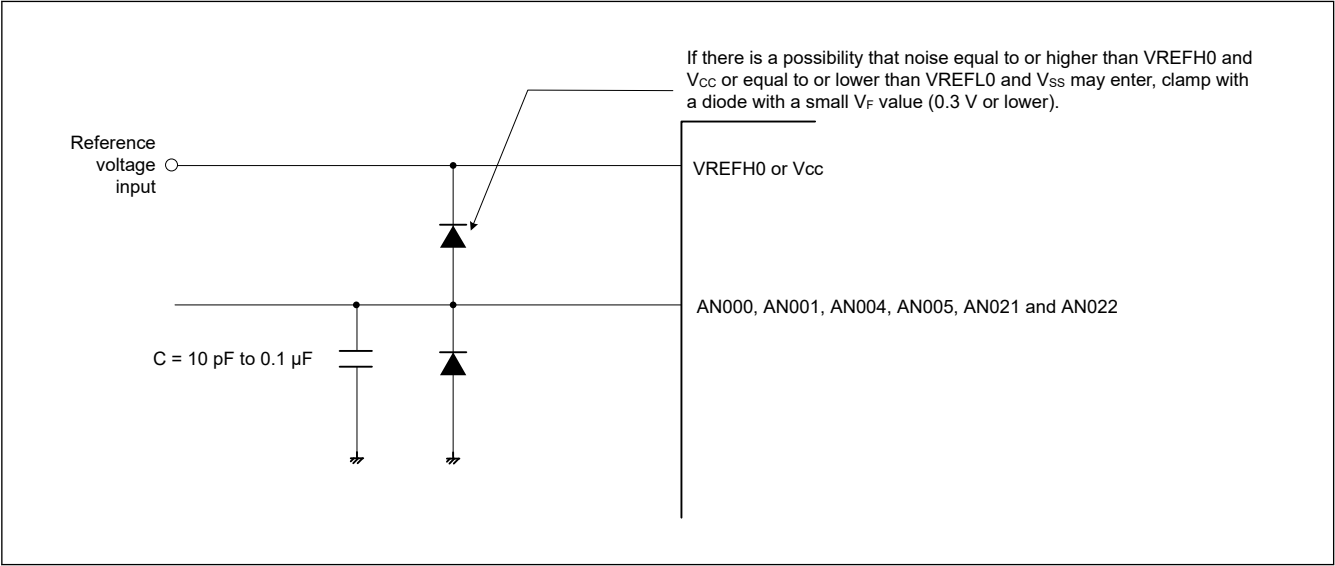


Figure 22.24 Connections of VCC, VREFH0, and analog input pins

此外，其他通道的转换值也可能受到影响。

当选择内部参考电压作为A/D转换器侧‘+’的参考电压时，请勿向ADS寄存器选定的引脚输入等于或高于内部参考电压的电压。但是，向ADS寄存器未选定的引脚输入等于或高于内部参考电压的电压则不会有问题。

注：有关内部参考电压的详细信息，请参阅第 27 节“电气特性”。

(3) 冲突操作

<1>转换结束时存储在 A/D 转换结果寄存器 (ADCR 或 ADCRH) 中的转换结果与通过指令读取 ADCR 或 ADCRH 寄存器之间存在冲突。

ADCR 或 ADCRH 寄存器读取操作具有优先权。读取操作完成后，新的转换结果将被写入 ADCR 或 ADCRH 寄存器。

<2>转换结束时存储在 A/D 转换结果寄存器 (ADCR 或 ADCRH) 中的转换结果与通过指令对 A/D 转换器模式寄存器 0 (ADM0) 或模拟输入通道规范寄存器 (ADS) 的写入访问之间存在冲突。

ADM0 和 ADS 寄存器写入操作优先级最高。ADCR 和 ADCRH 寄存器写入操作不会执行，也不会生成转换结束中断信号 (ADC10_ADI)。

(4) 噪声对抗措施

为了保持 10 位分辨率，必须注意输入到 VREFH0、VCC、AN000、AN001、AN004、AN005、AN021 和 AN022 引脚的噪声。

<1> 将一个等效电阻低、频率响应好的电容器（电容约为0.1 μF ）通过尽可能短的相对较粗的导线连接到VCC和VREFH0 引脚。

<2>模拟输入源的输出阻抗越高，影响越大。为了降低噪声，建议连接一个外部电容器，如图 22.24 所示。

<3>转换过程中请勿将这些引脚与其他引脚互换。

<4>如果在转换开始后立即设置睡眠模式，则可以提高准确性。

图 22.24 显示了VCC、VREFH0 和模拟输入引脚的连接。

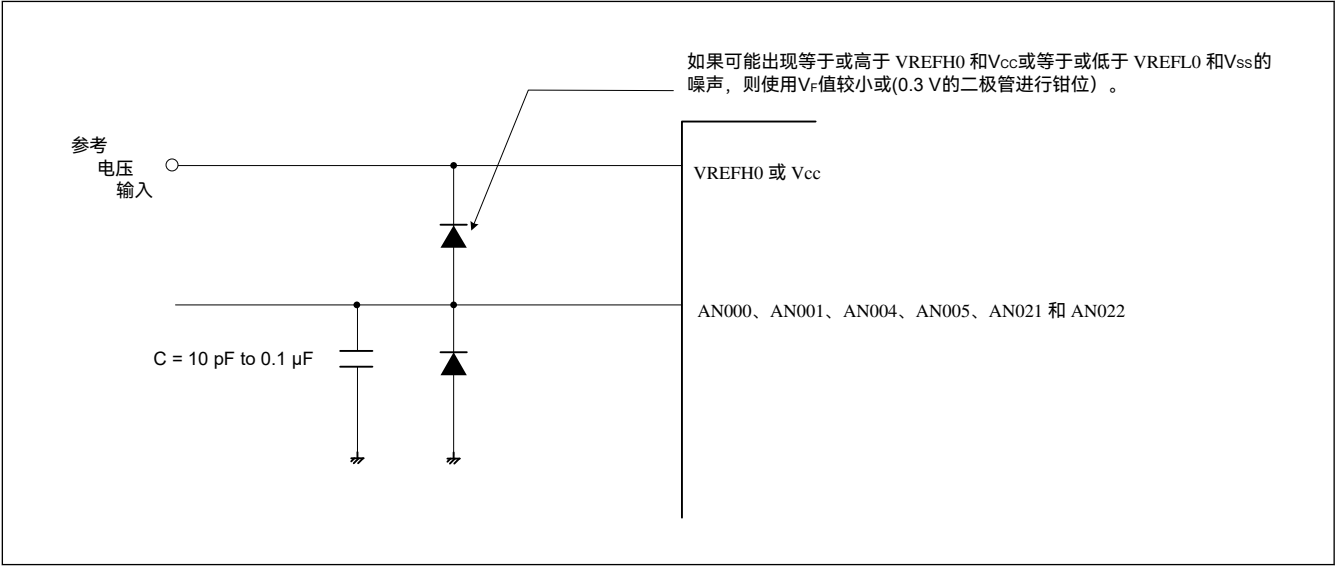


图 22.24VCC、VREFH0 和模拟输入引脚的连接

(5) Analog input (ANxxx) pins

<1> The analog input pins (AN000, AN001, AN004, AN005, AN021 and AN022) are also used as input port pins (P010 to P013, P100, and P101). When A/D conversion is performed with any of the AN000, AN001, AN004, AN005, AN021, and AN022 pins selected, do not change to output value P010 to P013, P100, and P101 while conversion is in progress; otherwise the conversion resolution may be degraded.

<2> If a pin adjacent to a pin that is being A/D converted is used as a digital I/O port pin, the A/D conversion result might differ from the expected value due to a coupling noise. Be sure to avoid the input or output of digital signals and signals with similarly sharp transitions during conversion.

(6) Input impedance of analog input (ANxxx) pins

This A/D converter charges a sampling capacitor for sampling during sampling time.

Therefore, only a leakage current flows when sampling is not in progress, and a current that charges the capacitor flows during sampling. Consequently, the input impedance fluctuates depending on whether sampling is in progress.

To make sure that sampling is effective, we recommend using the converter with analog input sources that have output impedance no greater than 1 kΩ. If a source has a higher output impedance, lengthen the sampling time or connect a larger capacitor (with a value of about 0.1 μF) to the pin from among AN000, AN001, AN004, AN005, AN021, and AN022 to which the source is connected (see Figure 22.24). The sampling capacitor may be being charged while the setting of the ADCS bit is 0 and immediately after sampling is restarted and so is not defined at these times. Accordingly, the state of conversion is undefined after charging starts in the next round of conversion after the value of the ADCS bit has been 1 or when conversion is repeated. Thus, to secure full charging regardless of the size of fluctuations in the analog signal, ensure that the output impedances of the sources of analog inputs are low or secure sufficient time for the completion of sampling.

(7) Interrupt Clear-pending Register (NVIC_ICPR0)

The Interrupt Clear-pending Register (NVIC_ICPR0) is not cleared even if the analog input channel specification register (ADS) is changed.

Therefore, if an analog input pin is changed during A/D conversion, the A/D conversion result and NVIC_ICPR0 for the prechanged analog input may have been set before the ADS register is rewritten. When reading the NVIC_ICPR0 immediately after rewriting to the ADS register, note that the NVIC_ICPR0 is set although A/D conversion for the post-change analog input has not ended.

When A/D conversion is stopped and then resumed, clear NVIC_ICPR0 before the A/D conversion operation is resumed.

Figure 22.25 shows timing of A/D conversion end interrupt request generation.

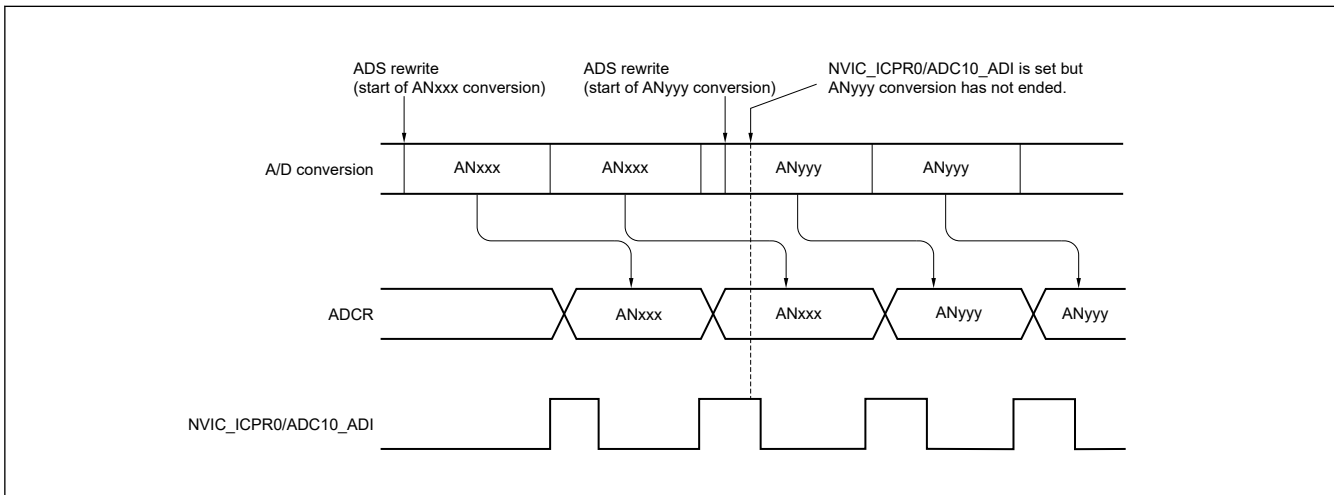


Figure 22.25 Timing of A/D conversion end interrupt request generation

(8) Conversion results just after A/D conversion start

While in the software trigger no-wait mode or hardware trigger no-wait mode, the first A/D conversion value immediately after A/D conversion starts may not fall within the rating range if the ADCS bit is set to 1 within 1 μs after the ADCE bit

(5) 模拟输入 (ANxxx) 引脚

<1> 模拟输入引脚 (AN000、AN001、AN004、AN005、AN021 和 AN022) 也可用作输入端口引脚 (P010 至 P013、P100 和 P101)。当选择 AN000、AN001、AN004、AN005、AN021 和 AN022 引脚中的任何一个进行 A/D 转换时, 请勿在转换过程中将 P010 至 P013、P100 和 P101 的输出值更改为 P010 至 P013、P100 和 P101; 否则, 转换分辨率可能会降低。

<2> 如果将与正在转换的引脚相邻的引脚用作数字 I/O 端口引脚, 则由于耦合噪声, A/D 转换结果可能与预期值有所不同。请务必避免在转换期间输入或输出数字信号以及具有类似陡峭变化的信号。

(6) 模拟输入 (ANxxx) 引脚的输入阻抗

该 A/D 转换器在采样期间为采样电容器充电, 以进行采样。

因此, 采样未进行时仅存在漏电流, 采样进行时则存在为电容器充电的电流。因此, 输入阻抗会根据采样是否进行而波动。

为确保采样有效, 我们建议将转换器与输出阻抗不大于 1 kΩ 的模拟输入源一起使用。如果输入源的输出阻抗较高, 请延长采样时间或将一个容量约为 0.1 μF 的电容器连接到与输入源连接的引脚 AN000、AN001、AN004、AN005、AN021 和 AN022 中的任意一个引脚 (参见图 22.24)。采样电容可能在 ADCS 位设置为 0 时以及采样重新启动后立即充电, 因此在这些时刻其状态未定义。相应地, 在 ADCS 位值为 1 之后的下一轮转换中, 或者在转换重复进行时, 转换状态在充电开始后未定义。因此, 为了确保无论模拟信号波动大小如何都能完全充电, 请确保模拟输入源的输出阻抗较低, 或确保有足够的时间完成采样。

(7) 中断清除挂起寄存器 (NVIC_ICPR0)

即使模拟输入通道规范寄存器 (ADS) 发生改变, 中断清除挂起寄存器 (NVIC_ICPR0) 也不会被清除。

因此, 如果在模数转换过程中模拟输入引脚发生变化, 则在 ADS 寄存器被重写之前, 变化前模拟输入的模数转换结果和 NVIC_ICPR0 可能已被置位。在重写 ADS 寄存器后立即读取 NVIC_ICPR0 时, 请注意, 即使变化后模拟输入的模数转换尚未结束, NVIC_ICPR0 也已被置位。

当 A/D 转换停止后又恢复时, 在恢复 A/D 转换操作之前, 清除 NVIC_ICPR0。

图 22.25 显示了 A/D 转换结束中断请求生成的时序。

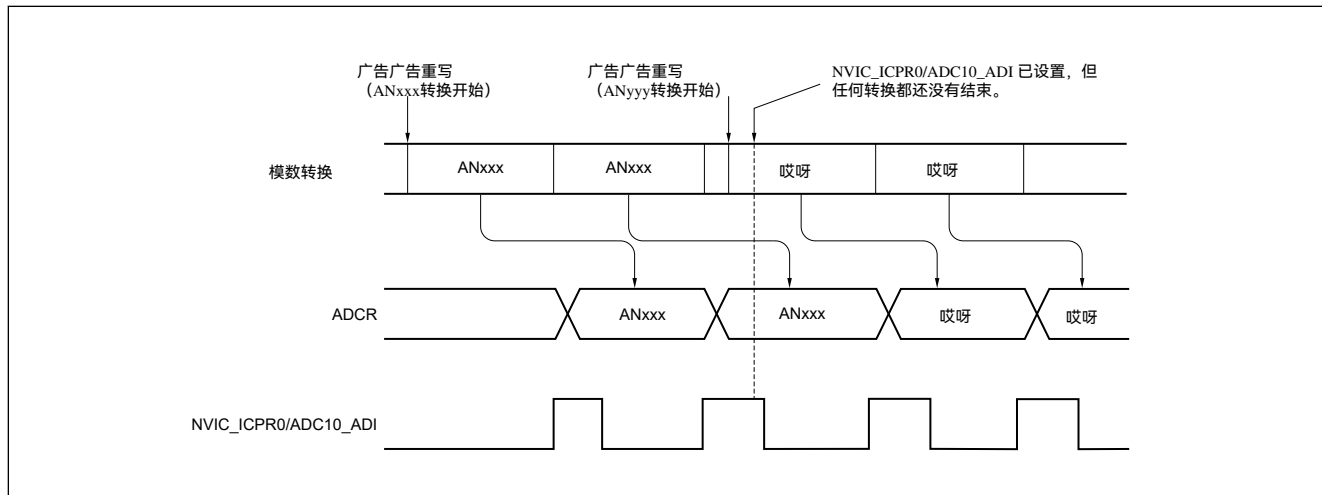


图 22.25 A/D 转换结束中断请求生成的时序

(8) 转换开始后的转换结果

在软件触发无等待模式或硬件触发无等待模式下, 如果在 ADCE 位设置为 1 之后的 1 μs 内将 ADCS 位设置为 1, 则 A/D 转换开始后的第一个 A/D 转换值可能不会落在额定范围内。

was set to 1. Take measures such as polling the A/D conversion end interrupt request signal (ADC10_ADI) and removing the first conversion result.

(9) A/D conversion result register (ADCR, ADCRH) read operation

When a write operation is performed to A/D converter mode register 0 (ADM0), analog input channel specification register (ADS), I/O port registers (PODRm, PDRm, PORRm, POSRm, and PmnPFS_A), the contents of the ADCR and ADCRH registers may become undefined. After the completion of conversion, read the conversion result before writing to the ADM0, ADS, I/O port registers (PODRm, PDRm, PORRm, POSRm, and PmnPFS_A), otherwise, an incorrect conversion result may be read.

(10) Starting the A/D converter

Start the A/D converter after the VREFH0 and V_{CC} voltages stabilize.

采取诸如轮询A/D 转换结束中断请求信号 (ADC10_ADI) 并删除第一个转换结果等措施。

(9) 模数转换结果寄存器 (ADCR、ADCRH) 读取操作

当对转换器模式寄存器 0 (ADM0)、模拟输入通道规范寄存器 (ADS) 以及 I/O 端口寄存器 (PODRm、PDRm、PORRm、POSRm 和 PmnPFS_A) 执行写入操作时，ADCR 和 ADCRH 寄存器的内容可能变为未定义。转换完成后，在写入 ADM0、ADS 和 I/O 端口寄存器 (PODRm、PDRm、PORRm、POSRm 和 PmnPFS_A) 之前，应先读取转换结果，否则可能会读取到错误的转换结果。

(10) 启动模数转换器

在 VREFH0 和V_{CC}电压稳定后启动 A/D 转换器。

23. Temperature Sensor (TSN)

23.1 Overview

The on-chip Temperature Sensor (TSN) determines and monitors the die temperature for reliable operation of the device. The sensor outputs a voltage directly proportional to the die temperature, and the relationship between the die temperature and the output voltage is fairly linear. The output voltage is provided to the ADC10 for conversion and can be further used by the end application.

Table 23.1 lists the TSN specifications, and Figure 23.1 shows a block diagram.

Table 23.1 TSN specifications

Item	Description
Temperature sensor voltage output	Temperature sensor outputs a voltage to the 10-bit A/D converter

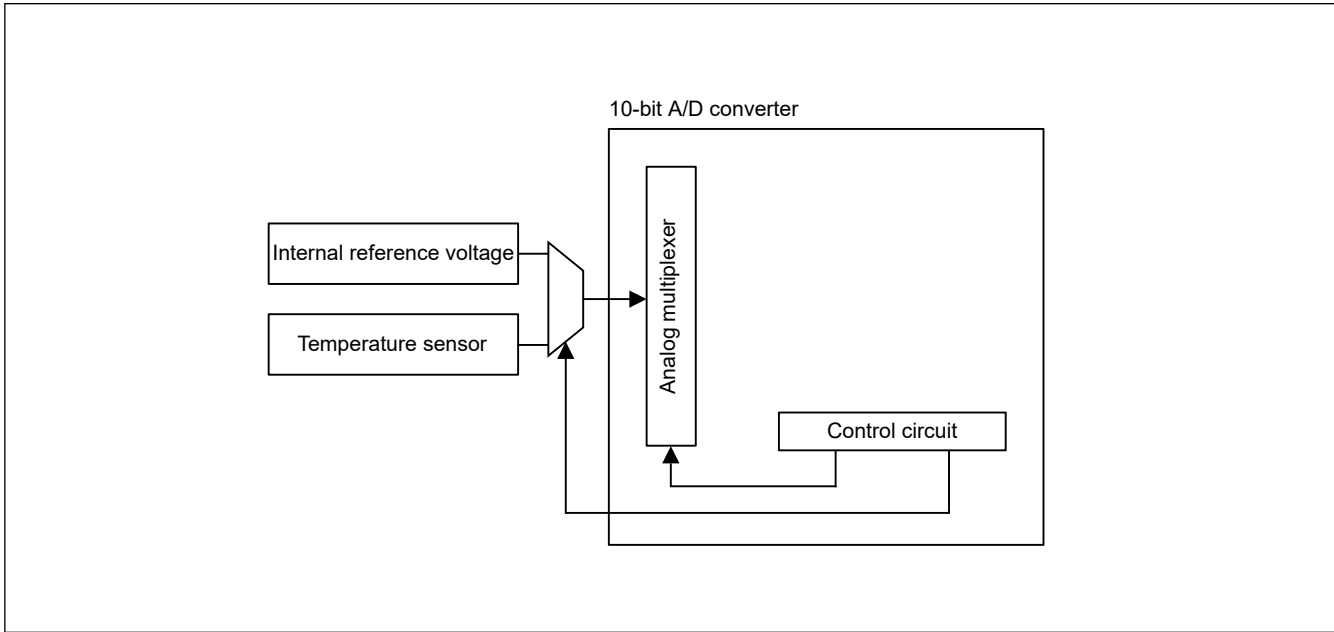


Figure 23.1 TSN block diagram

23.2 Using the Temperature Sensor

The temperature sensor outputs a voltage that varies with the temperature. This voltage is converted to a digital value by the 10-bit A/D converter. To obtain the die temperature, convert this value into the temperature.

23.2.1 Preparation for Using the Temperature Sensor

The ambient temperature (T) is proportional to the temperature sensor voltage output (Vs), so ambient temperature is calculated with the following formula:

$$T = (V_s - V_1) / \text{slope} + T_1$$

- T: Ambient temperature of MCU as calculation result (°C)
- Vs: Voltage output by the temperature sensor on temperature measurement (V)
- T1: Temperature experimentally measured at one point (°C)
- V1: Voltage output by the temperature sensor on measurement of T1 (V)
- T2: Temperature experimentally measured at a second point (°C)
- V2: Voltage output by the temperature sensor on measurement of T2 (V)
- Slope: Temperature gradient of the temperature sensor (V / °C), slope = (V2 - V1) / (T2 - T1)

Characteristics vary between sensors, so Renesas recommends measuring two different sample temperatures as follows:

23.温度传感器 (TSN)

23.1 概述

片上温度传感器 (TSN) 用于检测和监控芯片温度，以确保器件可靠运行。该传感器输出的电压与芯片温度成正比，且芯片温度与输出电压之间的关系基本呈线性。输出电压被送至 ADC10 进行转换，转换后的信号可供最终应用使用。

表 23.1 列出了 TSN 规格，图 23.1 显示了框图。

表 23.1 TSN 规格

物品	描述
温度传感器电压输出	温度传感器向10位模数转换器输出电压信号。

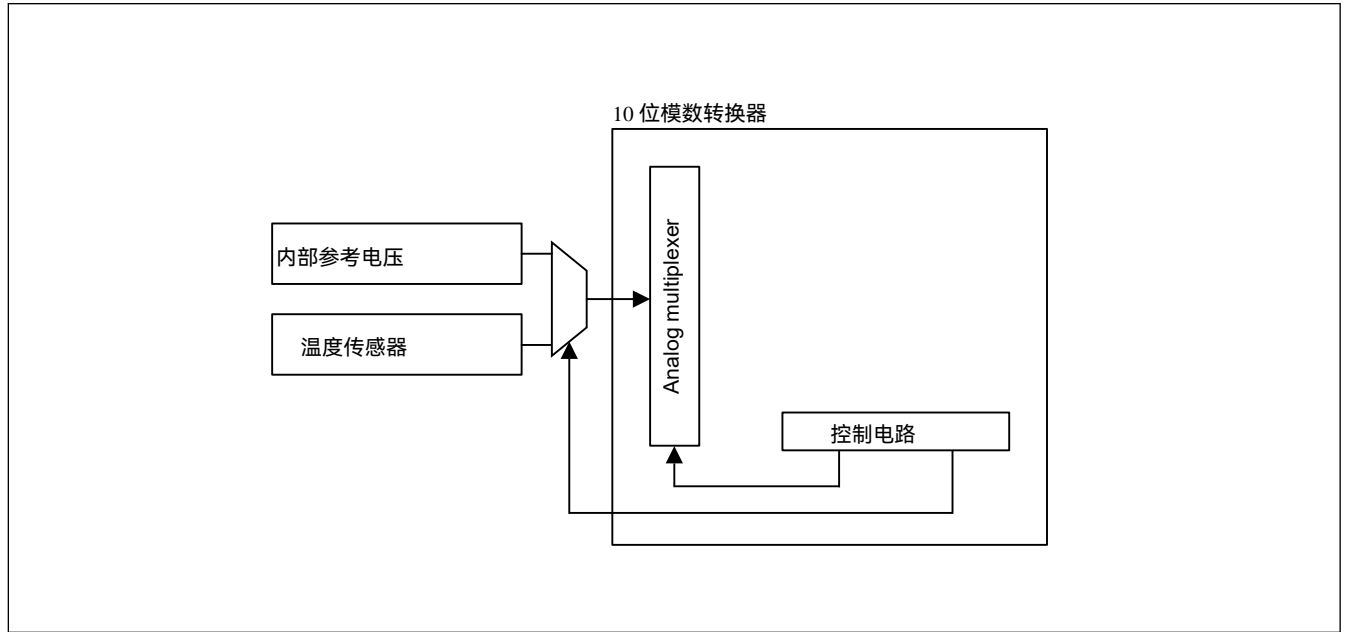


图 23.1 TSN 框图

23.2 使用温度传感器

温度传感器输出一个随温度变化的电压。该电压由 10 位 A/D 转换器转换为数字值。要获得芯片温度，需要将该数字值转换为温度值。

23.2.1 温度传感器使用前的准备工作

环境温度 (T) 与温度传感器电压输出 (Vs) 成正比，因此环境温度可以用以下公式计算：

$$T = (V_s - V_1) / \text{slope} + T_1$$

- T: MCU 的环境温度 (计算结果) (°C)
- Vs: 温度传感器在温度测量时输出的电压 (V)
- T1: 在这一点(°C)处实验测量的温度
- V1: 温度传感器测量T1时输出的电压(V)
- T2: 在第二个点(°C)处实验测量的温度
- V2: 温度传感器测量T2时输出的电压(V)
- 斜率: 温度传感器的温度梯度(V / °C), slope = (V2 - V1) / (T2 - T1)

不同传感器的特性各不相同，因此瑞萨电子建议按如下方式测量两个不同的样品温度：

1. Use the 10-bit A/D converter to measure the voltage V1 output by the temperature sensor at temperature T1.
2. Again use the 10-bit A/D converter to measure the voltage V2 output by the temperature sensor at a different temperature T2.
3. Obtain the temperature gradient (slope = (V2 - V1) / (T2 - T1)) from these results.
4. Subsequently, obtain temperatures by substituting the slope into the formula for the temperature characteristic (T = (Vs - V1) / slope + T1).

If you are using the temperature gradient given in [section 27, Electrical Characteristics](#), use the A/D converter to measure the voltage V1 output by the temperature sensor at temperature T1, then calculate the temperature characteristic using the following formula:

$$T = (V_s - V1) / \text{slope} + T1$$

Note: This method produces less accurate temperatures than measurement at two points.

23.2.2 Procedures for Using the Temperature Sensor

For details, see [section 22, 10-bit A/D Converter \(ADC10\)](#).

1. 使用 10 位A/D转换器测量温度传感器在温度 T1 下输出的电压V1 。
2. 再次使用 10 位A/D转换器测量温度传感器在不同温度 T2 下输出的电压。
3. 根据这些结果求得温度梯度(slope = (V2 - V1) / (T2 - T1)) 。
4. 随后， 将斜率代入温度特性公式(T = (Vs - V1) / slope + T1)。

如果您使用第 27 节“电气特性”中给出的温度梯度，请使用A/D转换器测量温度传感器在温度 T1 下输出的电压，然后使用以下公式计算温度特性：

$$T = (V_s - V1) / \text{slope} + T1$$

注意： 此方法测得的温度精度低于两点测量法。

23.2.2 温度传感器的使用步骤

有关详细信息，请参阅第 22 节，10 位 A/D 转换器 (ADC10)。

24. SRAM

24.1 Overview

The MCU provides an on-chip, high-density SRAM module.

Table 24.1 lists the SRAM specifications.

Table 24.1 SRAM specifications

Parameter	Description
SRAM capacity	SRAM0: 2 KB
SRAM address	SRAM0: 0x2000_4000 to 0x2000_47FF
Access*1	0 wait for both reading and writing

Note: SRAM0 and Trace RAM are shared. For the Trace RAM specifications, see *ARM® CoreSight™ MTB-M23 Technical Reference Manual (ARM DDI 0564C)*.
Note 1. For details, see [section 24.2.1. Access Cycle](#).

24.2 Operation

24.2.1 Access Cycle

Table 24.2 SRAM0 (0x2000_4000 to 0x2000_47FF)

Read (cycles)		Write (cycles)	
Word access	Halfword/Byte access	Word access	Halfword/Byte access
2		2	

24.3 Usage Notes

24.3.1 Instruction Fetch from the SRAM Area

When using SRAM0 to operate a program, initialize the SRAM area so that the CPU can correctly prefetch data. Initialize the additional 2-byte area from the end address of a program with a 4-byte boundary. Renesas recommends using the NOP instruction for data initialization.

24.3.2 SRAM Store Buffer

For fast access between SRAM and CPU, a store buffer is used. When a load instruction is executed from the same address after a store instruction to SRAM, the load instruction might read data from the buffer instead of data on the SRAM. To read data on the SRAM correctly, use either of the following procedures:

- After writing to the SRAM (address = A), use the NOP instruction, then read the SRAM (address = A)
- After writing to the SRAM (address = A), read data from area other than SRAM (address = A), then read the SRAM (address = A).

24. SRAM

24.1 概述

MCU 提供了一个片上高密度 SRAM 模块。

表 24.1 列出了 SRAM 规格。

表 24.1 SRAM 规格

范围	描述
SRAM 容量	SRAM0: 2 KB
SRAM 地址	SRAM0: 0x2000_4000 至 0x2000_47FF
访问*1	0 等待读写

注：SRAM0 和 Trace RAM 是共享的。有关 Trace RAM 的规格，请参阅 ARM® CoreSight™ MTB-M23 技术参考手册（ARM DDI 0564C）。
注1. 详情请参见第24.2.1节。访问周期。

24.2 操作

24.2.1访问周期

表 24.2SRAM0（0x2000_4000 至 0x2000_47FF）

读取（周期）		写入（循环）	
Word访问	半字/字节访问	Word访问	半字/字节访问
2		2	

24.3 使用说明

24.3.1 从 SRAM 区域获取指令

使用 SRAM0 运行程序时，需要初始化 SRAM 区域，以便 CPU 能够正确预取数据。对于边界为 4 字节的程序，应从程序结束地址开始初始化额外的 2 字节区域。瑞萨电子建议使用 NOP 指令进行数据初始化。

24.3.2 SRAM 存储缓冲区

为了实现 SRAM 和 CPU 之间的快速访问，使用了存储缓冲区。当从同一地址执行加载指令，且该加载指令与 SRAM 中的数据读取位置相同时，加载指令可能会从缓冲区读取数据，而不是从 SRAM 中读取数据。要正确读取 SRAM 中的数据，请使用以下任一方法：

- 写入 SRAM（地址= A）后，使用 NOP 指令，然后读取 SRAM（地址= A）。
- 写入 SRAM（地址= A）后，从 SRAM 以外的区域（地址= A）读取数据，然后读取 SRAM（地址= A）。

25. Flash Memory

25.1 Overview

The MCU provides 16-KB code flash memory. The Flash Control Block (FCB) controls the programming commands. This product uses SuperFlash® technology licensed from Silicon Storage Technology, Inc.

Table 25.1 lists the specifications of the code flash memory and Figure 25.1 shows a block diagram of the related modules. Figure 25.2 shows the configuration of the code flash memory.

Table 25.1 Code flash memory specifications

Parameter	Code flash memory
Memory capacity	16-KB of user area - Configuration setting area (See section 6, Option-Setting Memory)
Read cycle	A read operation takes 2 cycles
Value after erasure	0xFF
Programming/erasing method	- Programming and erasure of code flash memory through the FCB commands specified in the registers - Programming of flash memory by user program (self-programming)*1.
Security function	Protection against illicit tampering with or reading of data in flash memory
Protection	Protection against erroneous overwriting of flash memory
Units of programming and erasure	32-bit units for programming in user area 2-KB units for erasure in user area.
Other functions	Interrupts accepted during self-programming
	Option-setting memory can be set in the initial MCU settings
On-board programming	Programming in on-chip debug mode: - SWD interface used - Dedicated hardware not required. Programming by a routine for code flash memory programming within the user program: - Allows code flash memory programming without resetting the system.

Note 1. HOCO should be stably oscillated. See section 25.9. Self-programming.

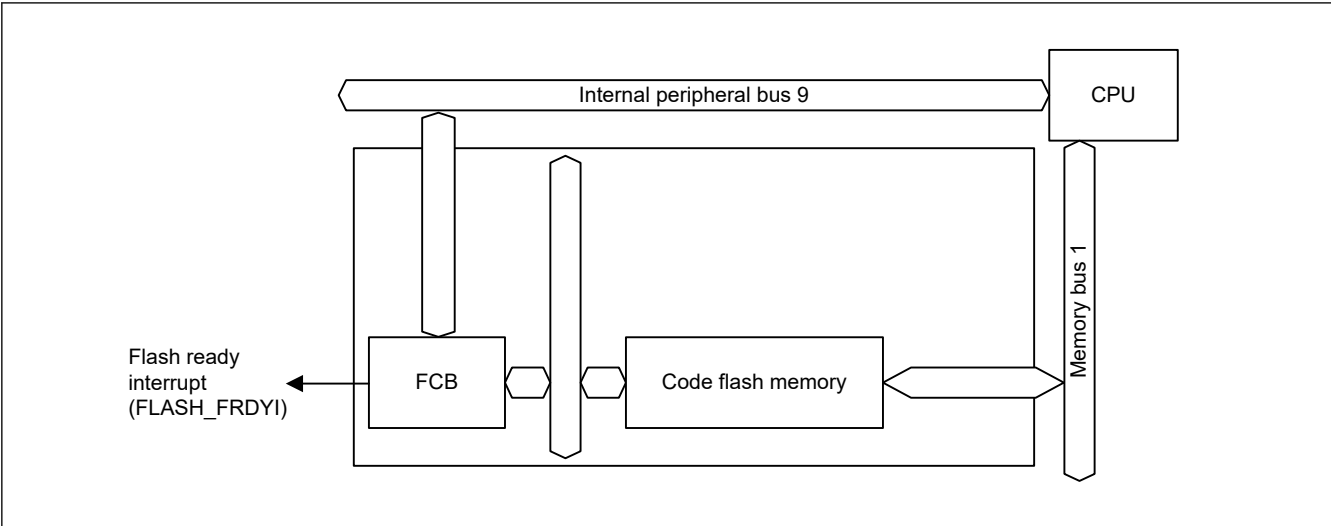


Figure 25.1 Flash memory-related modules block diagram

25.闪存

25.1 概述

MCU 提供 16 KB 的代码闪存。闪存控制块 (FCB) 控制编程命令。本产品采用 Silicon Storage Technology, Inc. 授权的 SuperFlash® 技术。

表 25.1 列出了代码闪存的规格，图 25.1 显示了相关模块的框图。图 25.2 显示了代码闪存的配置。

表 25.1 代码闪存规格

范围	代码闪存
内存容量	●16KB 用户区域 ●配置设置区域（参见第6节，选项设置记忆）
读取周期	●读取操作需要2个时钟周期
擦除后的值	0xFF
编程/擦除方法	●通过寄存器中指定的FCB命令对代码闪存进行编程和擦除 ●通过用户程序对闪存进行编程（自编程）*1.
安全功能	防止非法篡改或读取闪存中的数据
保护	防止闪存被错误覆盖
编程和擦除单元	●用于用户区编程的32位单元 ●用户区域擦除单位为2KB。
其他功能	自编程过程中接受的中断
	可以在MCU初始设置中设置选项设置存储器。
车载编程	片上调试模式编程： ●使用SWD接口 ●无需专用硬件。 用户程序中通过例程对闪存进行代码编程： ●允许在不重置系统的情况下对闪存进行代码编程。

注1. HOCO应稳定振荡。参见第25.9节。自编程。

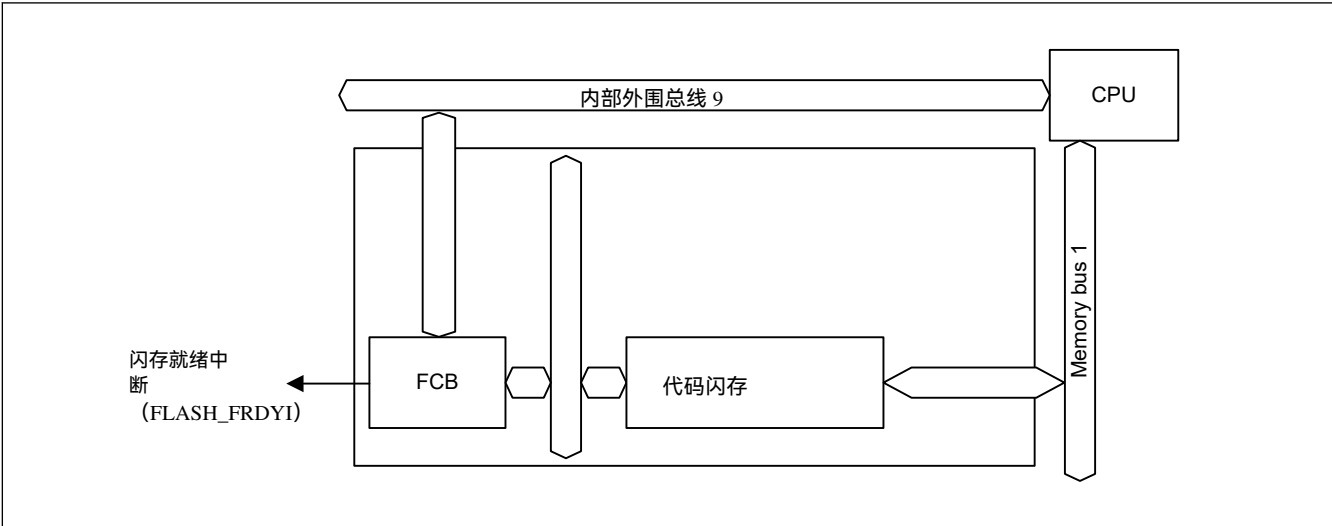


图 25.1 闪存相关模块框图

25.2 Memory Structure

Figure 25.2 shows the mapping of the code flash memory, and Table 25.2 shows the read and programming and erasure (P/E) addresses of the code flash memory. The user area of the code flash memory is divided into 2-KB blocks that serve as the units of erasure. The user area is available for storing the user program.

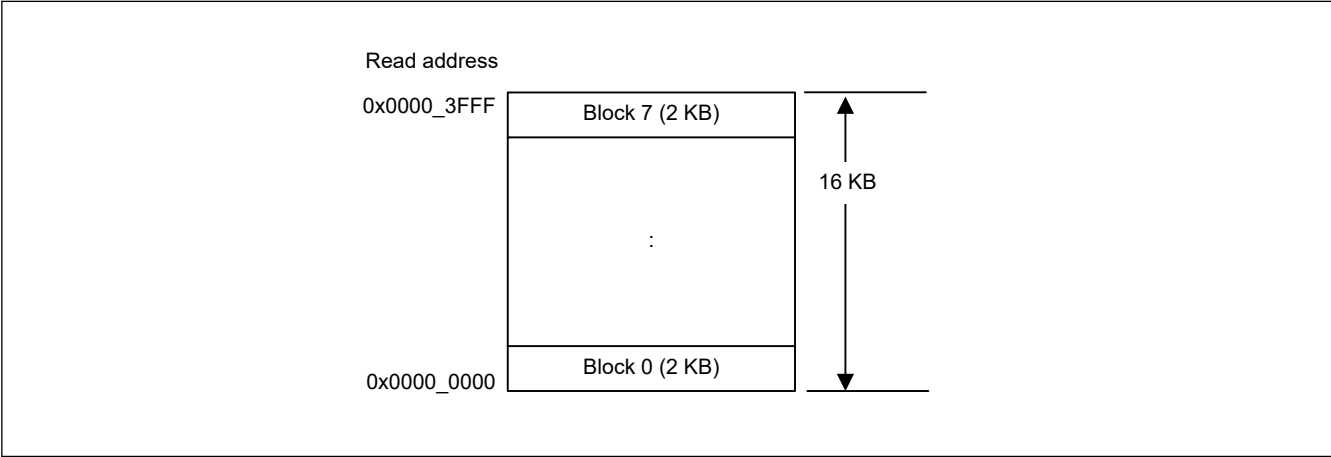


Figure 25.2 Mapping of the code flash memory

Table 25.2 Read and P/E addresses of the code flash memory

Size of code flash memory	Read address	P/E address	Number of blocks
16 KB	0x0000_0000 to 0x0000_3FFF	0x0000_0000 to 0x0000_3FFF	0 to 7

25.3 Register Descriptions

25.3.1 FENTRYR : Flash P/E Mode Entry Register

Base address: FLCN = 0x407E_C000																
Offset address: 0x021A																
Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	FEKEY[7:0]								—	—	—	—	—	—	—	FENT RY0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
0	FENTRY0	Code Flash P/E Mode Entry 0 0: The code flash is the read mode 1: The code flash is the P/E mode.	R/W
7:1	—	These bits are read as 0. The write value should be 0.	R/W
15:8	FEKEY[7:0]	Key Code	W

To program the code flash, the FENTRY0 bit must be set to 1 to enter the P/E mode. Clearing the FENTRY0 bit allows the code flash to be in read mode, but it is necessary to confirm the value of this bit before changing it. See [section 25.10.1. Sequencer Modes](#).

FENTRY0 bit (Code Flash P/E Mode Entry 0)

[Setting condition]

- Set 0xAA01 to the FENTRYR register when it is 0x0000.

[Clearing conditions]

- Data is written by byte access

25.2 内存结构

图 25.2 显示了代码闪存的映射关系，表 25.2 显示了代码闪存的读取地址和编程/擦除 (P/E) 地址。代码闪存的用户区被划分为 2 KB 的数据块，这些数据块用作擦除单元。用户区可用于存储用户程序。

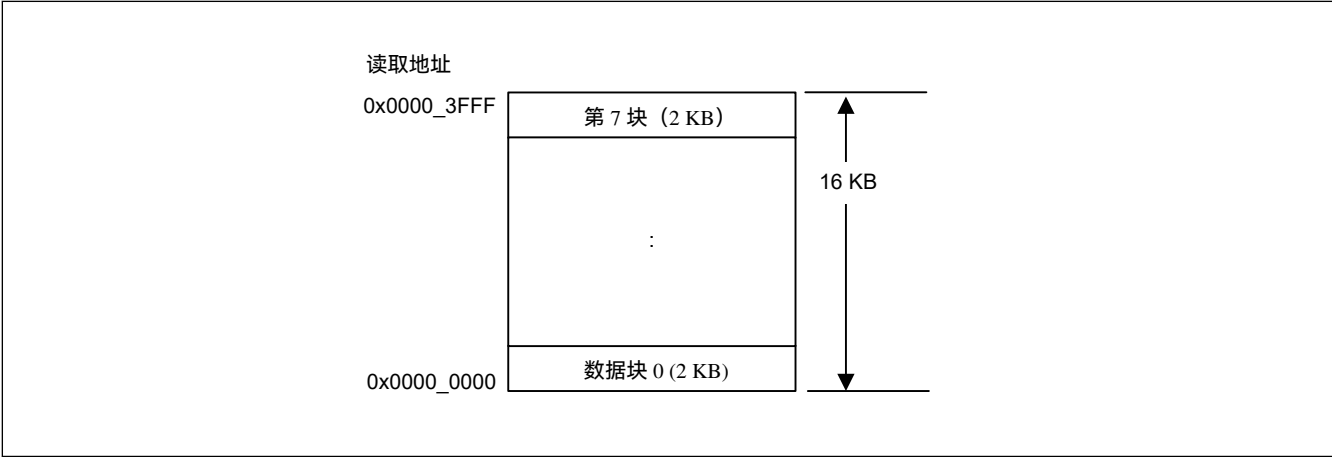


图 25.2 代码闪存映射

表 25.2 代码闪存的读取地址和 P/E 地址

代码闪存的大小	读取地址	私募股权地址	方块数量
16 KB	0x0000_0000 to 0x0000_3FFF	0x0000_0000 to 0x0000_3FFF	0 to 7

25.3 寄存器描述

25.3.1FENTRYR： 闪存 P/E 模式输入寄存器

基地址：FLCN= 0x407E_C000																
偏移地址：0x021A																
位位置：	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域：	FEKEY[7:0]								—	—	—	—	—	—	—	FENT RY0
重置后的值：	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
0	FENTRY0	代码闪存 P/E 模式入口 0 0: 代码闪存为读取模式 1: 代码闪存为P/E模式。	R/W
7:1	—	这些位读取为 0。写入值应为 0。	R/W
15:8	FEKEY[7:0]	密钥	W

要对代码闪存进行编程，必须将 FENTRY0 位设置为 1 以进入P/E模式。清除 FENTRY0 位可使代码闪存进入读取模式，但在更改此位之前必须确认其值。请参阅 25.10.1 节。
[音序器模式。](#)

FENTRY0 位（代码闪存 P/E 模式入口 0）

[设置条件]

- 当 FENTRYR 寄存器为0x0000时，将其设置为 0xAA01

[清算条件]

- 数据通过字节访问写入

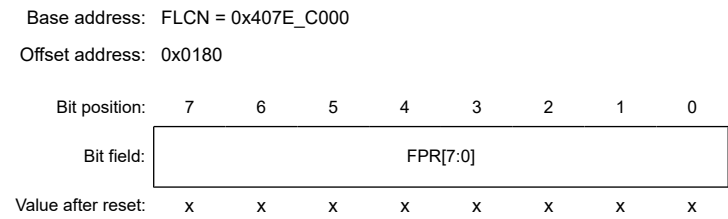
- A value other than 0xAA is set to the FEKEY[7:0] bits and written to the FENTRYR register
- Set 0xAA00 to the FENTRYR register
- Data is written to the FENTRYR register while the register has a value other than 0x0000.

FEKEY[7:0] bits (Key Code)

The FEKEY[7:0] bits protect from unauthorized setting of FENTRY0 bit.

Setting 0xAA to FEKEY[7:0] allows setting the FENTRY0 bit. The FEKEY[7:0] bits are read as 0x00.

25.3.2 FPR : Protection Unlock Register



Bit	Symbol	Function	R/W
7:0	FPR[7:0]	Protection Unlock This register is used to protect the FPMCR register from being rewritten inadvertently when the CPU runs out of control.	W

FPR[7:0] bits (Protection Unlock)

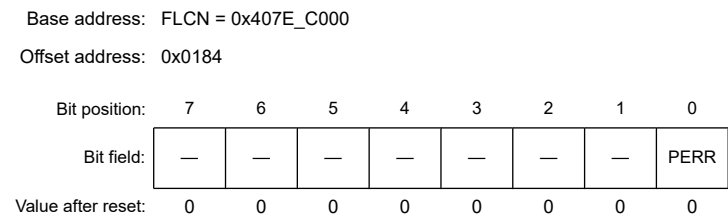
Writing to the FPMCR register is allowed only when the following procedure is used to access the register.

Procedure to unlock protection:

1. Write 0xA5 to the FPR register.
2. Write a set value to the FPMCR register
3. Write the inverted set value to the FPMCR register.
4. Write a set value to the FPMCR register again.

When a procedure other than the specified procedure is used to write data, the FPSR.PERR flag is set to 1.

25.3.3 FPSR : Protection Unlock Status Register



Bit	Symbol	Function	R/W
0	PERR	Protect Error Flag 0: No error 1: An error occurs	R
7:1	—	These bits are read as 0.	R

PERR bit (Protect Error Flag)

When the FPMCR register is not accessed as described in the procedure to unlock protection, data is not written to the register and this flag is set to 1.

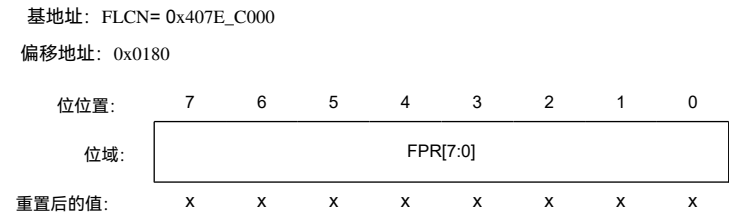
- FEKEY[7:0] 位的值被设置为除 0xAA 以外的值，并写入 FENTRYR 寄存器。
- 将 FENTRYR 寄存器的值设置为 0xAA00
- 当寄存器的值不是0x0000时，数据将被写入 FENTRYR 寄存器。

FEKEY[7:0] 位（密钥代码）

FEKEY[7:0] 位防止未经授权设置 FENTRY0 位。

将 0xAA 设置为 FEKEY[7:0] 可以设置 FENTRY0 位。FEKEY[7:0] 位读取为0x00.

25.3.2FPR：保护解锁寄存器



少量	象征	功能	R/W
7:0	FPR[7:0]	保护解锁 该寄存器用于防止 CPU 失控时 FPMCR 寄存器被意外重写。	W

FPR[7:0] 位（保护解锁）

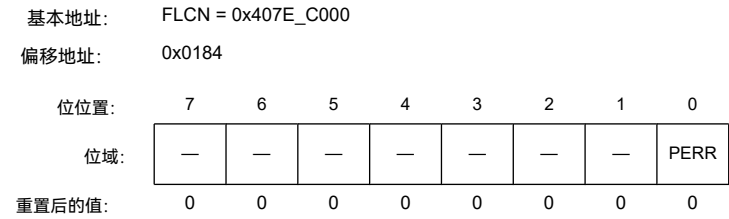
仅当使用以下程序访问寄存器时，才允许写入 FPMCR 寄存器。

解锁保护的步骤:

1. 向 FPR 寄存器写入 0xA5。
2. 向 FPMCR 寄存器写入一个设定值
3. 将反转后的值写入 FPMCR 寄存器。
4. 再次向 FPMCR 寄存器写入一个设定值。

当使用指定过程以外的过程写入数据时，FPSR.PERR 标志设置为 1。

25.3.3FPSR：保护解锁状态寄存器



少量	象征	功能	R/W
0	PERR	保护错误标志 0: 无错误 1: 发生错误	R
7:1	—	这些位读取为 0。	R

PERR 位（保护错误标志）

当未按照解锁保护的步骤访问 FPMCR 寄存器时，数据不会写入寄存器，并且该标志位设置为 1。

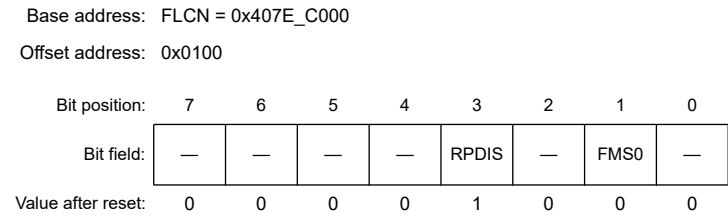
[Setting condition]

- The FPMCR register is not accessed as described in the procedure to unlock protection described in [section 25.3.2. FPR : Protection Unlock Register](#).

[Clearing conditions]

- The FPMCR register is accessed according to the procedure to unlock protection described in [section 25.3.2. FPR : Protection Unlock Register](#).

25.3.4 FPMCR : Flash P/E Mode Control Register



Bit	Symbol	Function	R/W
0	—	This bit is read as 0. The write value should be 0.	R/W
1	FMS0	Flash Operating Mode Select 0 0: Read mode 1: Code flash P/E mode	R/W
2	—	This bit is read as 0. The write value should be 0.	R/W
3	RPDIS	Code Flash P/E Disable 0: Programming of the code flash is enabled 1: Programming of the code flash is disabled.	R/W
7:4	—	These bits are read as 0. The write value should be 0.	R/W

The FPMCR register sets the operating mode of the flash memory and is protected from unauthorized setting.
See [Figure 25.11](#) and [Figure 25.12](#) for this register write control method.
See [section 25.3.2. FPR : Protection Unlock Register](#) for the procedure to unlock the protection.

FMS0 bit (Flash Operating Mode Select 0)

These bits set the operating mode of the flash memory.

[How to enter the code flash from the read mode to the code flash P/E mode]

Set FMS0 = 1, and RPDIS = 0. Wait for the mode setup time t_{MS} (see [section 27, Electrical Characteristics](#)).

[How to enter the code flash from the code flash P/E mode to the read mode]

Set FMS0 = 0, and RPDIS = 1.

Wait for the read mode transition time (see [section 27, Electrical Characteristics](#)).

RPDIS bit (Code Flash P/E Disable)

RPDIS bit protects the code flash from unauthorized programming. Setting RPDIS bit to 0 allows the code flash to program.

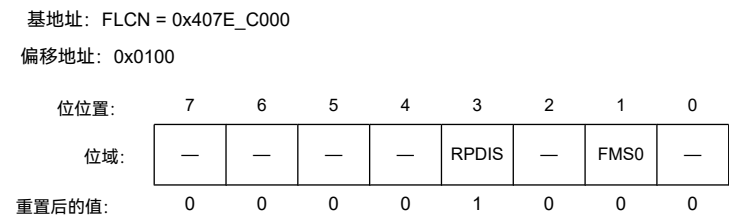
[设置条件]

- FPMCR 寄存器未按照第 25.3.2 节所述的解锁保护程序进行访问。
FPR: [保护解锁寄存器](#)。

[清算条件]

- 根据第25.3.2节所述的解锁保护程序访问FPMCR寄存器。FPR：
[保护解锁注册](#)。

25.3.4FPMCR： 闪存 P/E 模式控制寄存器



少量	象征	功能	R/W
0	—	该位读取值为 0。写入值应为 0。	R/W
1	FMS0	闪存操作模式选择 0 0: 读取模式 1: 代码闪光 P/E 模式	R/W
2	—	该位读取值为 0。写入值应为 0。	R/W
3	RPDIS	代码闪存编程禁用 0: 代码闪存编程已启用 1: 代码 闪存编程已禁用。	R/W
7:4	—	这些位读取为 0。写入值应为 0。	R/W

FPMCR 寄存器用于设置闪存的工作模式，并受到保护，防止未经授权的设置。
有关此寄存器写入控制方法，请参见图 25.11 和图 25.12。
有关解锁保护的程序，请参阅第 25.3.2 节。FPR:[保护解锁寄存器](#)。

FMS0 位（闪存操作模式选择 0）

这些位设置闪存的运行模式。

[如何从读取模式进入代码闪存模式P/E]

设置FMS0 = 1和RPDIS = 0。等待模式建立时间 t_{MS}（参见第 27 节，电气特性）。

如何从代码闪存模式进入读取模式？

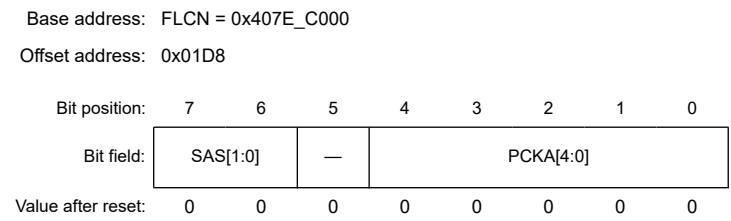
设置FMS0 = 0和RPDIS = 1。

等待读取模式转换时间（参见第 27 节，电气特性）。

RPDIS 位（代码闪光 P/E 禁用）

RPDIS 位用于保护代码闪存免受未经授权的编程。将 RPDIS 位设置为 0 允许对代码闪存进行编程。

25.3.5 FISR : Flash Initial Setting Register



Bit	Symbol	Function	R/W
4:0	PCKA[4:0]	Flash-IF Clock Notification	R/W
5	—	This bit is read as 0. The write value should be 0.	R/W
7:6	SAS[1:0]	Startup Area Select 1 0: The startup area is switched to the default area temporarily 1 1: The startup area is switched to the alternate area temporarily. Others: The startup area is selected according to the settings of the extra area.	R/W

Note: Set or clear this register only in P/E mode. Additionally, the SAS[1:0] bits are allowed to set or clear when the FSPR is 1. The FSPR bit is the protection flag of the access window and is stored in the extra area.

PCKA[4:0] bits (Flash-IF Clock Notification)

The hardware sequencer for the flash programming executes the commands according to the PCKA[4:0] bits. For this reason, it is necessary to set the PCKA[4:0] bits according to Flash-IF clock (ICLK) before execution of the programming and not during the programming.

Note: A wrong frequency setting may cause the flash memory to be damaged.

The following information describes how to set the PCKA[4:0] bits when the frequency is not an integral number, for example 31.5 MHz.

[When the frequency is higher than 4 MHz]

Set a rounded-up value for a non-integer frequency.

For example, set 32 MHz (PCKA = 11111b) when the frequency is 31.5 MHz.

[When the frequency is 4 MHz or lower]

Do not use a non-integer frequency. Use the frequency of 1, 2, 3, or 4 MHz.

Table 25.3 Frequency settings

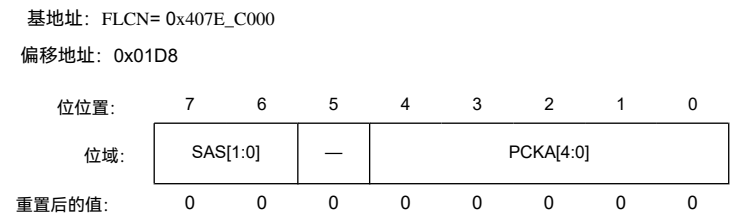
Flash-IF clock frequency [MHz]	PCKA[4:0]	Flash-IF clock frequency [MHz]	PCKA[4:0]	Flash-IF clock frequency [MHz]	PCKA[4:0]
32	11111b	24	10111b	20	10011b
19	10010b	18	10001b	17	10000b
16	01111b	15	01110b	14	01101b
13	01100b	12	01011b	11	01010b
10	01001b	9	01000b	8	00111b
7	00110b	6	00101b	5	00100b
4	00011b	3	00010b	2	00001b
1	00000b	—	—	—	—

SAS[1:0] bits (Startup Area Select)

The SAS[1:0] bits select the startup area. To change the startup area, the following methods can be used:

- When selecting the startup area according to the startup area settings of the extra area with the SAS[1:0] bits set to 00b or 01b, the startup area is selected accordingly. The settings are enabled after a reset is released.

25.3.5 FISR: Flash 初始设置寄存器



少量	象征	功能	R/W
4:0	PCKA[4:0]	Flash-IF 时钟通知	R/W
5	—	该位读取值为 0。写入值应为 0。	R/W
7:6	SAS[1:0]	启动区域选择1 0:启动区域暂时切换到默认区域 1 1:启动区域暂时切换到备用区域。 其他：启动区域根据额外区域的设置进行选择。	R/W

注意：仅在P/E 模式下设置或清除此寄存器。此外，当 FSPR 为 1 时，允许设置或清除 SAS[1:0] 位。FSPR 位是访问窗口的保护标志，存储在额外区域中。

PCKA[4:0] 位（Flash-IF 时钟通知）

用于闪存编程的硬件序列器根据 PCKA[4:0] 位执行命令。因此，必须在执行编程之前，而不是在编程过程中，根据闪存中频时钟 (ICLK) 设置 PCKA[4:0] 位。

注意：错误的频率设置可能会导致闪存损坏。

以下信息描述了当频率不是整数时，如何设置 PCKA[4:0] 位，例如31.5 MHz。

[当频率高于4 MHz]

对于非整数频率，设置向上取整的值。

例如，当频率为31.5 MHz时，设置32 MHz (PCKA = 11111b)

[当频率为4 MHz或更低]

请勿使用非整数频率。请使用频率 1、2、3 或4 MHz。

表 25.3 频率设置

Flash-IF 时钟频率[MHz]	PCKA[4:0]	Flash-IF 时钟频率[MHz]	PCKA[4:0]	Flash-IF 时钟频率[MHz]	PCKA[4:0]
32	11111b	24	10111b	20	10011b
19	10010b	18	10001b	17	10000b
16	01111b	15	01110b	14	01101b
13	01100b	12	01011b	11	01010b
10	01001b	9	01000b	8	00111b
7	00110b	6	00101b	5	00100b
4	00011b	3	00010b	2	00001b
1	00000b	—	—	—	—

SAS[1:0] 位（启动区选择）

SAS[1:0] 位用于选择启动区域。要更改启动区域，可以使用以下方法：

- 当根据额外区域的启动区域设置选择启动区域时，如果 SAS[1:0] 位设置为00b或 01b，则会相应地选择启动区域。复位释放后，这些设置将生效。

- When switching the startup area to the default area temporarily with 10b written to the SAS[1:0] bits, the startup area is switched to the default area immediately after data is written to the register, regardless of the startup area settings of the extra area. When a reset is generated after this, the area is selected according to the startup area settings of the extra area.
- When switching the startup area to the alternative area temporarily with 11b written to the SAS[1:0] bits, the startup area is switched to the alternative area, regardless of the startup area settings of the extra area. When a reset is generated after this, the area is selected according to the startup area settings of the extra area.

25.3.6 FRESETR : Flash Reset Register

Base address:	FLCN = 0x407E_C000						
Offset address:	0x0124						
Bit position:	7	6	5	4	3	2	1 0
Bit field:	—	—	—	—	—	—	FRES ET
Value after reset:	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
0	FRESET	Software Reset of the Registers 0: The registers related to the flash programming are not reset 1: The registers related to the flash programming are reset.	R/W
7:1	—	These bits are read as 0. The write value should be 0.	R/W

FRESET bit (Software Reset of the Registers)

When this bit is set to 1, the FASR, FSARH, FSARL, FEARH, FEARL, FWBH0, FWBL0, FCR, and FEXCR registers are reset. Setting this bit to 0 allows the corresponding registers to be released from the reset state. Software commands are not allowed to execute while the FRESET bit is 1.

25.3.7 FASR : Flash Area Select Register

Base address:	FLCN = 0x407E_C000						
Offset address:	0x0104						
Bit position:	7	6	5	4	3	2	1 0
Bit field:	—	—	—	—	—	—	EXS
Value after reset:	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
0	EXS	Extra Area Select 0: User area 1: Extra area	R/W
7:1	—	These bits are read as 0. The write value should be 0.	R/W

Note: Set or clear this register only in P/E mode.

EXS bit (Extra Area Select)

Set the EXS bit to 1 when programming the extra area using the FEXCR register. Set this bit to 0 when not programming the extra area.

- 当向 SAS[1:0] 位写入 10 位数据，并将启动区临时切换到默认区时，无论额外区域的启动区设置如何，数据写入寄存器后，启动区都会立即切换到默认区。此后，如果发生复位，则会根据额外区域的启动区设置选择相应的区域。
- 当通过向 SAS[1:0] 位写入 11b 将启动区临时切换到备用区时，无论额外区域的启动区设置如何，启动区都会切换到备用区。此后，如果发生复位，则会根据额外区域的启动区设置选择启动区。

25.3.6FRESETR：闪存复位寄存器

基地址:	FLCN= 0x407E_C000						
偏移地址:	0x0124						
位位置:	7	6	5	4	3	2	1 0
位域:	—	—	—	—	—	—	FRES ET
重置后的值:	0	0	0	0	0	0	0

少量	象征	功能	R/W
0	FRESET	软件复位寄存器 0: 与闪存编程相关的寄存器未复位 1: 与闪存编程相关的寄存器已复位。	R/W
7:1	—	这些位读取为 0。写入值应为 0。	R/W

FRESET 位（寄存器的软件复位）

当此位设置为 1 时，FASR、FSARH、FSARL、FEARH、FEARL、FWBH0、FWBL0、FCR 和 FEXCR 寄存器将被复位。将此位设置为 0 则允许将相应的寄存器从复位状态释放。当 FRESET 位为 1 时，不允许执行软件命令。

25.3.7FASR：闪存区域选择寄存器

基地址:	FLCN = 0x407E_C000						
偏移地址:	0x0104						
位位置:	7	6	5	4	3	2	1 0
位域:	—	—	—	—	—	—	EXS
重置后的值:	0	0	0	0	0	0	0

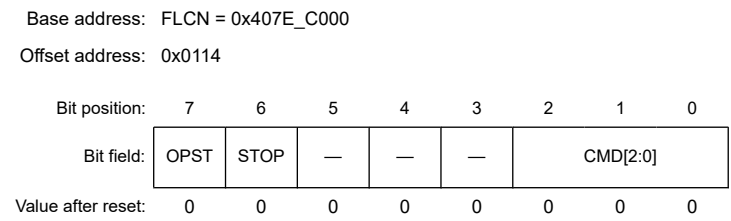
少量	象征	功能	R/W
0	EXS	额外区域选择 0: 用户区域 1: 额外区域	R/W
7:1	—	这些位读取为 0。写入值应为 0。	R/W

注意: 只能在 P/E 模式下设置或清除此寄存器。

EXS 位（额外区域选择）

使用 FEXCR 寄存器对额外区域进行编程时，将 EXS 位设置为 1；不使用 FEXCR 寄存器对额外区域进行编程时，将该位设置为 0。

25.3.8 FCR : Flash Control Register



Bit	Symbol	Function	R/W
2:0	CMD[2:0]	Software Command Setting 0x1: Program 0x3: Blank check (code flash) 0x4: Block erase Others: Setting prohibited*1.	R/W
5:3	—	These bits are read as 0. The write value should be 0.	R/W
6	STOP	Forced Processing Stop When this bit is set to 1, the processing being executed can be forcibly stopped.	R/W
7	OPST	Processing Start 0: Processing stops 1: Processing starts.	R/W

Note: Set or clear this register only in P/E mode. Additionally, it is not allowed to be reset by the FRESETR register while the software command is being executed.

Note 1. This does not include writing 0x00 to the FCR register when the FSTATR1.FRDY bit is 1.

CMD[2:0] bits (Software Command Setting)

The following information describes the function of each software command.

[Program]

Writes data of the FWBH0 and FWBL0 registers to the flash memory to the address pointed by the FSARH and FSARL registers.

[Blank check]

Verifies whether the flash memory is the blank state (not to be programmed) from the start address pointed by the FSARH and FSARL registers to the end address pointed by the FEARH and FEARL registers. The blank check command is allowed to execute within the region of flash memory.

Note: The blank check result cannot guarantee that the flash memory is erased.

[Block erase]

Erases block of the flash memory.

Set the start address of the target erasure block in the FSARH and FSARL registers, and set the end address of the target erasure block in the FEARH and FEARL registers. If a setting other than the specified is made, erasure may not be executed correctly. The block erase command is allowed to execute within the region of flash memory.

STOP bit (Forced Processing Stop)

The STOP bit stops the execution of the erase command or the blank check command.

After setting 1 to the STOP bit, it is necessary to wait until the FSTATR1.FRDY bit becomes 1 (processing completed) before setting the OPST bit to 0.

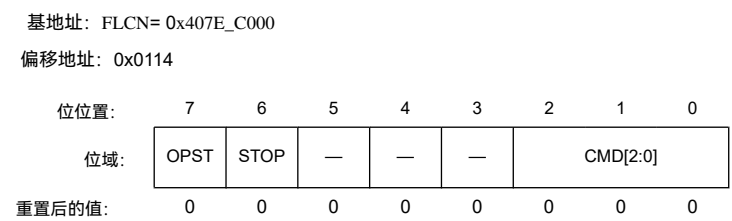
OPST bit (Processing Start)

The OPST bit starts the command set for the CMD[2:0] bits. Setting the OPST bit to 0 terminates the execution of the command after the FRDY bit of the FSTATR1 register becomes 1, and is required to confirm that the FRDY bit is 0.

Note:

- Commands cannot be executed when the ID authorization for the flash programmer has failed.

25.3.8 FCR: 闪存控制寄存器



少量	象征	功能	R/W
2:0	CMD[2:0]	软件命令设置 0x1: 程序0x3: 空白检查（代码闪烁） 0x4: 块擦除 其他: 设置禁止*1。	R/W
5:3	—	这些位读取为 0。写入值应为 0。	R/W
6	STOP	强制停止处理 当此位设置为 1 时，可以强制停止正在执行的处理。	R/W
7	OPST	处理开始 0: 处理停止 1: 处理开始。	R/W

注意: 此寄存器只能在 P/E 模式下设置或清除。此外，在执行软件命令期间，不允许使用 FRESETR 寄存器复位此寄存器。

注 1. 这包括当 FSTATR1.FRDY 位为 1 时向 FCR 寄存器写入0x00。

CMD[2:0] 位（软件命令设置）

以下信息描述了每个软件命令的功能。

[节目]

将 FWBH0 和 FWBL0 寄存器的数据写入闪存中 FSARH 和 FSARL 寄存器指向的地址。

[空白支票]

验证从 FSARH 和 FSARL 寄存器指向的起始地址到 FEARH 和 FEARL 寄存器指向的结束地址之间的闪存区域是否处于空白状态（不可编程）。空白检查命令允许在闪存区域内执行。

注意: 空白检查结果不能保证闪存已被擦除。

[块擦除]

擦除闪存中的一块数据。

在 FSARH 和 FSARL 寄存器中设置目标擦除块的起始地址，并在 FEARH 和 FEARL 寄存器中设置目标擦除块的结束地址。如果设置值与指定值不同，则擦除操作可能无法正确执行。块擦除命令允许在闪存区域内执行。

停止位（强制处理停止）

STOP 位停止擦除命令或空白校验命令的执行。

将 STOP 位设置为 1 后，必须等到 FSTATR1.FRDY 位变为 1（处理完成）后，才能将 OPST 位设置为 0。

OPST 位（处理开始）

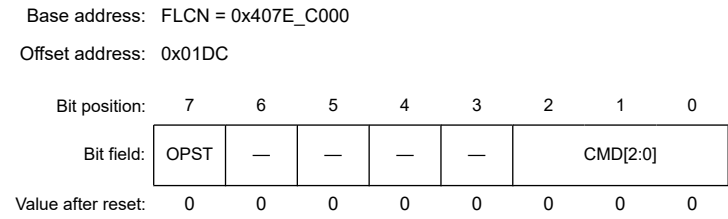
OPST 位用于启动 CMD[2:0] 位的命令集。将 OPST 位设置为 0 会在 FSTATR1 寄存器的 FRDY 位变为 1 后终止命令的执行，并且需要确认 FRDY 位为 0。

注意:

- 当闪存编程器的 ID 授权失败时，无法执行命令。

- The program, the block erase, and the read commands cannot be executed when the address of each command points to an area that is protected by the access window.

25.3.9 FEXCR : Flash Extra Area Control Register



Bit	Symbol	Function	R/W
2:0	CMD[2:0]	Software Command Setting 0 1 0: Access window information program Startup area selection and security setting 0 1 1: OCDID1 program 1 0 0: OCDID2 program 1 0 1: OCDID3 program 1 1 0: OCDID4 program Others: Setting prohibited*1.	R/W
6:3	—	These bits are read as 0. The write value should be 0.	R/W
7	OPST	Processing Start 0: Processing stops 1: Processing starts.	R/W

Note: Set or clear this register only in P/E mode. Additionally, it is not allowed to be reset by the FRESETR register while the software command is being executed.

Note 1. This does not include writing 0x00 to the FEXCR register when the FSTATR1.EXRDY bit is 1.

The FEXCR register programs the extra area. Before execution of each command, it is necessary to set the FWBL0 and FWBH0 registers.

When programming using the FEXCR register, the programming area is erased automatically before execution, therefore it is not necessary to erase beforehand.

CMD[2:0] bits (Software Command Setting)

The CMD[2:0] bits select the software command from the:

- Startup area selection and security setting
- Access window information program
- OCDID program.

The following information describes the function of each software command.

[Startup area selection and security setting]

Setting data to the FWBL0/FWBH0 registers, this command is allowed to select the startup area from the default area (4 KB) to the alternative area (next 4 KB) and set the security. For details, see [section 25.8.1. Startup Program Protection](#).

Bit [15] of the FWBH0 register is 0 and the alternative area (next 4 KB) is selected as the startup area.

Bit [15] of the FWBH0 register is 1 and the default area (4 KB) is selected as the startup area.

Bit [15] of the FWBL0 register is 0.

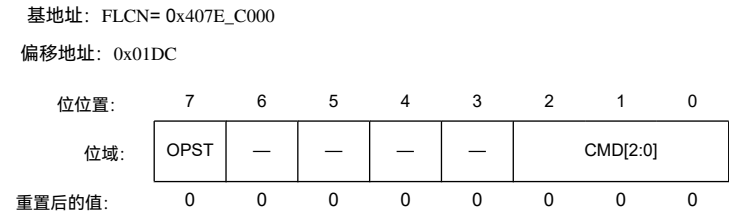
- The access window cannot be updated because the access window information program command cannot be executed.
- The startup area cannot be changed.
- Data of the SAS bits of the FISR register cannot be changed.

Note: The security setting command cannot be set to 1 for the corresponding bit of the extra area after 0 is set.

The following information describes mapping for the extra bit of the startup area selection and security setting.

- 当每个命令的地址指向受访问窗口保护的区域时，程序、块擦除和读取命令都无法执行。

25.3.9FEXCR：闪存额外区域控制寄存器



少量	象征	功能	R/W
2:0	CMD[2:0]	软件命令设置 0 1 0: 访问窗口信息程序启动区域选择和 安全设置 0 1 1: OCDID1 程序 1 0 0: OCDID2 程序 1 0 1: OCDID3 程序 1 1 0: OCDID4 程序其他: 设置禁止*1。	R/W
6:3	—	这些位读取为 0。写入值应为 0。	R/W
7	OPST	处理开始 0: 处理停止 1: 处理开始。	R/W

注意: 此寄存器只能在P/E 模式下设置或清除。此外，在执行软件命令期间，不允许使用 FRESETR 寄存器复位此寄存器。

注 1. 这不包括当 FSTATR1.EXRDY 位为 1 时向 FEXCR 寄存器写入0x00。

FEXCR寄存器用于对额外区域进行编程。在执行每个命令之前，必须设置FWBL0和FWBH0寄存器。

使用 FEXCR 寄存器进行编程时，编程区域会在执行前自动擦除，因此无需事先擦除。

CMD[2:0] 位（软件命令设置）

CMD[2:0] 位用于从以下位置选择软件命令：

- 启动区域选择和安全设置
- 访问窗口信息程序
- 强迫症治疗方案。

以下信息描述了每个软件命令的功能。

[启动区域选择和安全设置]

此命令用于设置 FWBL0/FWBH0 寄存器的数据，允许从默认区域（4 KB）到备用区域（下一个 4 KB）选择启动区域并设置安全设置。有关详细信息，请参阅第 25.8.1 节“启动程序保护”。

FWBH0 寄存器的第 [15] 位为 0，并且选择备用区域（下一个4 KB）作为启动区域。

FWBH0 寄存器的第 [15] 位为 1，默认区域（4 KB）被选为启动区域。

FWBL0 寄存器的第 [15] 位为 0。

- 由于无法执行访问窗口信息程序命令，因此无法更新访问窗口。
- 启动区域无法更改。
- FISR寄存器的SAS位数据不能更改。

注意: 安全设置命令不能将额外区域的相应位设置为 1，即使该位设置为 0。

以下信息描述了启动区域选择和安全设置的额外部分的映射。

Table 25.4 Mapping for the extra bit of the startup area selection and security setting (address (P/E) : 0x0000_0010)

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
BTFL G	—	—	—	—	FAWE[10:0]										
b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
FSPR *1	—	—	—	—	FAWS[10:0]										

Note 1. Once 0 is set for the bit, it cannot be changed to 1.

[Access window information program]

This command sets the access window used for area protection. The program command and block erase command of the protected area cannot be executed. It is necessary to set the start block address of the access window to the FWBL0 register bits [10:0] and the next block address of the end block address of the access window to the FWBH0 register bits [10:0] before the execution of the access window information program command. When the start address and the end address are set to the same value, all areas of the code flash can be accessed. When the start address is larger than the end block address, all areas of the code flash cannot be accessed.

The FWBL0[10] bit for the start block address must be set to 0 when the access window is set (the end block address of the access window is larger than the start block address).

The following information describes mapping for the extra bit of the access window information program.

Table 25.5 Mapping for the extra bit of the access window information program (address (P/E) : 0x0000_0010)

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
BTFL G	—	—	—	—	FAWE[10:0]										
b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
FSPR *1	—	—	—	—	FAWS[10:0]										

Note 1. Once 0 is set for the bit, it cannot be changed to 1.

[OCDID1-4 program]

These commands set the OCDID[127:0] bits.

Table 25.6 OCDID settings

Command	OCDID	FWBH0	FWBL0
OCDID1 program	OCDID [31:0]	OCDID [31:16]	OCDID [15:0]
OCDID2 program	OCDID [63:32]	OCDID [63:48]	OCDID [47:32]
OCDID3 program	OCDID [95:64]	OCDID [95:80]	OCDID [79:64]
OCDID4 program	OCDID [127:96]	OCDID [127:112]	OCDID [111:96]

The following information describes mapping for the extra bit of OCDID1-4 program.

Table 25.7 Mapping for the extra bit of OCDID1-4 program (address (P/E) : 0x0000_0018)

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
OCDID[31:16]															
b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
OCDID[15:0]															

Table 25.8 Mapping for the extra bit of OCDID1-4 program (address (P/E) : 0x0000_0020) (1 of 2)

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
OCDID[63:48]															

表 25.4 启动区域选择和安全设置额外位的映射（地址 (P/E) ）：0x0000_0010)

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
BTFL G	—	—	—	—	FAWE[10:0]										
b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
FSPR *1	—	—	—	—	FAWS[10:0]										

注1：一旦某个位被设置为0，就不能再将其更改为1。

[访问窗口信息程序]

此命令设置用于区域保护的访问窗口。受保护区域的编程命令和块擦除命令将无法执行。在执行访问窗口信息编程命令之前，必须将访问窗口的起始块地址设置为 FWBL0 寄存器的[10:0]，并将访问窗口的结束块地址的下一个块地址设置为 FWBH0 寄存器的[10:0]。当起始地址和结束地址设置为相同值时，可以访问代码闪存的所有区域。当起始地址大于结束块地址时，无法访问代码闪存的所有区域。

设置访问窗口时，起始块地址的 FWBL0 [10] 位必须设置为 0（访问窗口的结束块地址大于起始块地址）。

以下信息描述了访问窗口信息程序的额外位映射。

表 25.5 访问窗口信息程序的额外位映射（地址 (P/E) ）：0x0000_0010)

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
BTFL G	—	—	—	—	FAWE[10:0]										
b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
FSPR *1	—	—	—	—	FAWS[10:0]										

注1：一旦某个位被设置为0，就不能再将其更改为1。

[OCDID1-4 程序]

这些命令设置 OCDID[127:0] 位。

表 25.6 OCDID 设置

命令	OCDID	FWBH0	FWBL0
OCDID1 计划	OCDID [31:0]	OCDID [31:16]	OCDID [15:0]
OCDID2 计划	OCDID [63:32]	OCDID [63:48]	OCDID [47:32]
OCDID3 程序	OCDID [95:64]	OCDID [95:80]	OCDID [79:64]
OCDID4 计划	OCDID [127:96]	OCDID [127:112]	OCDID [111:96]

以下信息描述了 OCDID1-4 程序额外位的映射。

表 25.7 OCDID1-4 程序额外位的映射（地址 (P/E) : 0x0000_0018)

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
OCDID[31:16]															
b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
OCDID[15:0]															

表 25.8 OCDID1-4 程序额外位的映射（地址 (P/E) : 0x0000_0020) （1/2）

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
OCDID[63:48]															

Table 25.8 Mapping for the extra bit of OCDID1-4 program (address (P/E) : 0x0000_0020) (2 of 2)

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
OCDID[47:32]															

Table 25.9 Mapping for the extra bit of OCDID1-4 program (address (P/E) : 0x0000_0028)

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
OCDID[95:80]															
b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
OCDID[79:64]															

Table 25.10 Mapping for the extra bit of OCDID1-4 program (address (P/E) : 0x0000_0030)

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
OCDID[127:112]															
b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
OCDID[111:96]															

OPST bit (Processing Start)

The OPST bit starts the command set for the CMD[2:0] bits. Setting the OPST bit to 0 terminates the execution of the command after the EXRDY bit of the FSTATR1 register becomes 1, and is necessary to confirm that the EXRDY bit is 0.

25.3.10 FSARH : Flash Processing Start Address Register H

Base address:	FLCN = 0x407E_C000
Offset address:	0x0110
Bit position:	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Bit field:	FSARH[15:0]
Value after reset:	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Bit	Symbol	Function	R/W
15:0	FSARH[15:0]	Flash Processing Start Address H Flash Processing Start Address upper 16 bits See FSARL for details.	R/W

Note: Set or clear this register only in P/E mode. The write value should be 0 for b8 to b4, and those bits are read as 0.

25.3.11 FSARL : Flash Processing Start Address Register L

Base address:	FLCN = 0x407E_C000
Offset address:	0x0108
Bit position:	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Bit field:	FSARL[15:0]
Value after reset:	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Bit	Symbol	Function	R/W
15:0	FSARL[15:0]	Flash Processing Start Address L Flash processing start address lower 16 bits	R/W

Note: Set or clear this register only in P/E mode.

The FSARH and FSARL registers set the start address of the software command. When the FSARH and FSARL registers are read while executing a software command set by the FEXCR register, an undefined value is read.

表 25.8 OCDID1-4 程序额外位的映射（地址 (P/E) : 0x0000_0020）（2/2）

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
OCDID[47:32]															

表 25.9 OCDID1-4 程序额外位的映射（地址 (P/E) : 0x0000_0028）

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
OCDID[95:80]															
b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
OCDID[79:64]															

表 25.10 OCDID1-4 程序额外位的映射（地址 (P/E) : 0x0000_0030）

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
OCDID[127:112]															
b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
OCDID[111:96]															

OPST 位（处理开始）

OPST 位开始对CMD[2:0] 位进行命令集设置。将 OPST 位设置为 0 会在 FSTATR1 寄存器的 EXRDY 位变为 1 后终止命令的执行，并且需要确认 EXRDY 位为0。

25.3.10FSARH： 闪存处理起始地址寄存器 H

基地址:	FLCN = 0x407 E_C000
Offset address:	0x0110
位位置:	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
位域:	FSARH[15:0]
重置后的值:	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

少量	象征	功能	R/W
15:0	FSARH[15:0]	闪存处理起始地址 H 闪存处理起始地址高 16 位，详情请参阅 FSA RL。	R/W

注意： 仅在 P/E 模式下设置或清除此寄存器。写入值应为 b8 至 b4 位的 0，且这些位读取时也为 0。

25.3.11FSARL： 闪存处理起始地址寄存器 L

基地址:	FLCN = 0x407E_C000
偏移地址:	0x0108
位位置:	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
位域:	FSARL[15:0]
重置后的值:	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

少量	象征	功能	R/W
15:0	FSARL[15:0]	闪存处理起始地址 L 闪存处理起始地址低 16 位	R/W

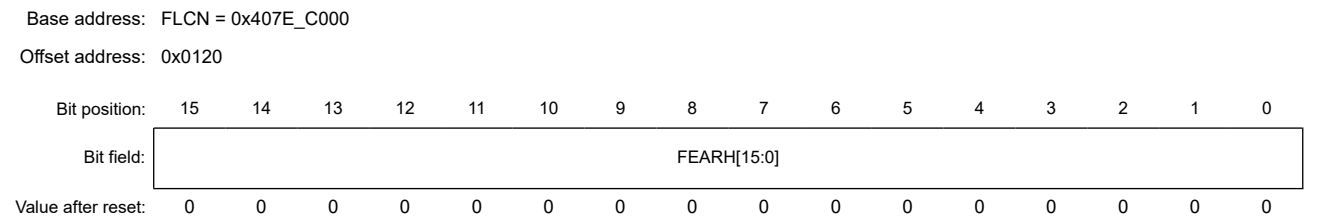
注意： 只能在 P/E 模式下设置或清除此寄存器。

FSARH 和 FSARL 寄存器用于设置软件命令的起始地址。当执行由 FEXCR 寄存器设置的软件命令时，如果读取 FSA RH 和 FSARL 寄存器，则会读取到未定义的值。

Note: This product does not have the auto increment function of the program command. It is necessary to set the next address to the FSARH and FSARL registers every time programming flash.

See [Figure 25.2](#) and [section 25.2. Memory Structure](#) for details on the addresses of the flash memory.

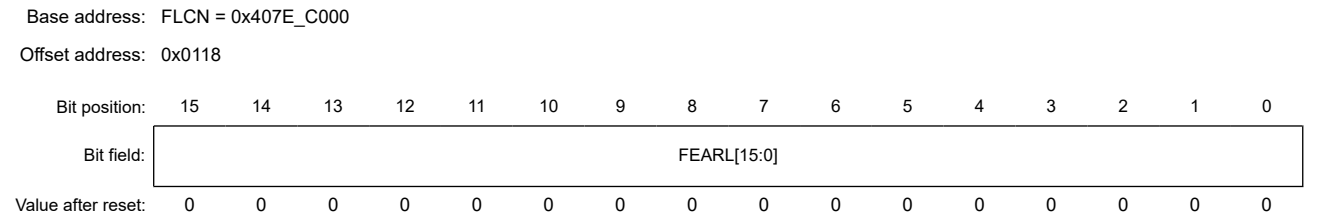
25.3.12 FEARH : Flash Processing End Address Register H



Bit	Symbol	Function	R/W
15:0	FEARH[15:0]	Flash Processing End Address H Flash processing end address upper 16 bits See FEARL for details.	R/W

Note: Set or clear this register only in P/E mode. The write value should be 0 for b8 to b4, and those bits are read as 0.

25.3.13 FEARL : Flash Processing End Address Register L



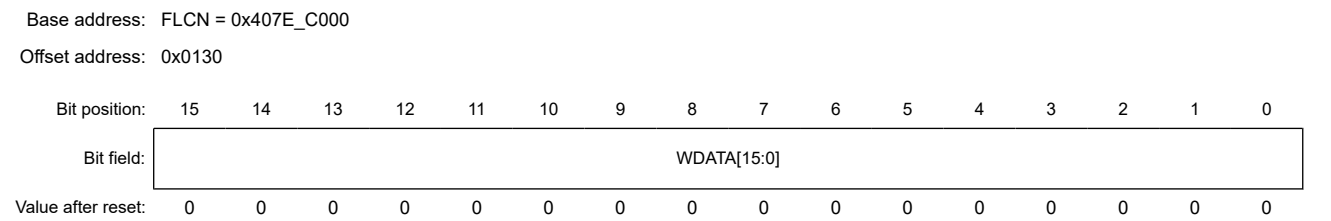
Bit	Symbol	Function	R/W
15:0	FEARL[15:0]	Flash Processing End Address L Flash processing end address lower 16 bits	R/W

Note: Set or clear this register only in P/E mode.

The FEARH and FEARL registers set the end address of the blank check and the block erase command. When the FEARH and FEARL registers are read while executing a software command set by the FEXCR register, an undefined value is read.

See [Figure 25.2](#) and [section 25.2. Memory Structure](#) for details on the addresses of the flash memory.

25.3.14 FWBL0 : Flash Write Buffer Register L0



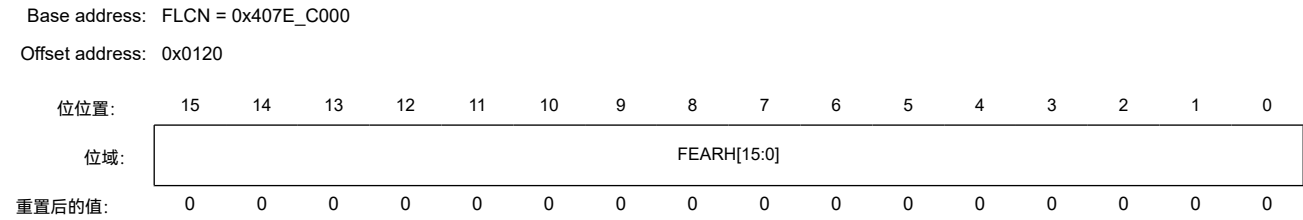
Bit	Symbol	Function	R/W
15:0	WDATA[15:0]	Flash Write Buffer L0 Flash write buffer data lower 16 bits See FWBH0 for details.	R/W

Note: Set or clear this register only in P/E mode.

注意: 本产品不具备编程指令自动递增功能。每次对闪存进行编程时, 都需要手动设置FSARH和FSARL寄存器的下一个地址。

有关闪存地址的详细信息, 请参见图 25.2 和第 25.2 节“存储器结构”。

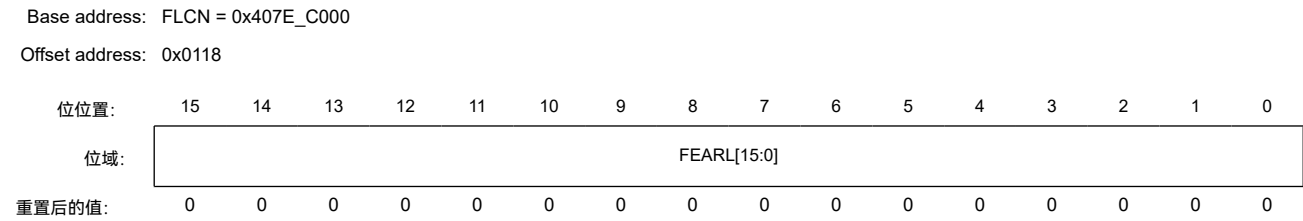
25.3.12FEARH: 闪存处理结束地址寄存器 H



少量	象征	功能	R/W
15:0	FEARH[15:0]	Flash 处理结束地址 H Flash 处理结束地址高 16 位有关详细信息, 请参阅 FEARL。	R/W

注意: 仅在 P/E 模式下设置或清除此寄存器。写入值应为 b8 至 b4 位的 0, 且这些位读取时也为 0。

25.3.13FEARL: 闪存处理结束地址寄存器 L



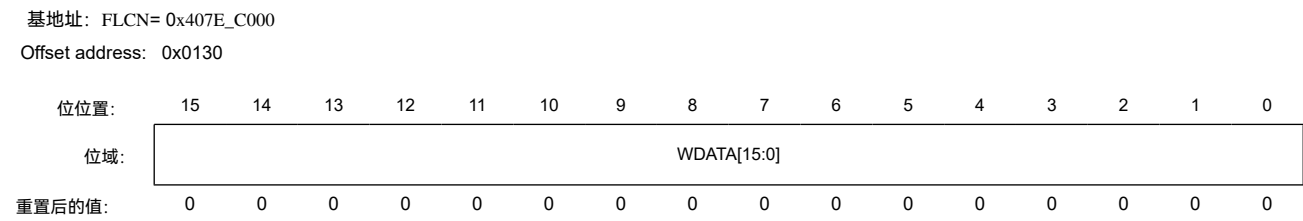
少量	象征	功能	R/W
15:0	FEARL[15:0]	闪存处理结束地址 L 闪存处理结束地址低 16 位	R/W

注意: 只能在 P/E 模式下设置或清除此寄存器。

FEARH 和 FEARL 寄存器分别设置空白校验和块擦除命令的结束地址。当执行由 FEXCR 寄存器设置的软件命令时, 如果读取 FEARH 和 FEARL 寄存器, 则会读取到未定义的值。

有关闪存地址的详细信息, 请参见图 25.2 和第 25.2 节“存储器结构”。

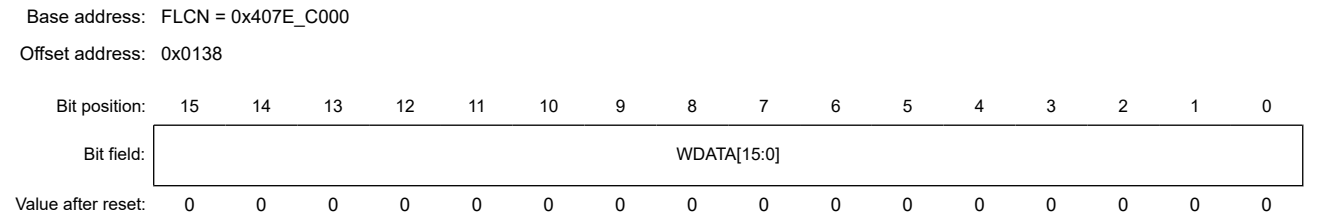
25.3.14FWBL0: 闪存写入缓冲寄存器 L0



少量	象征	功能	R/W
15:0	WDATA[15:0]	闪存写入缓冲区 L0 闪存写入缓冲区数据低 16 位 详情请参见FWBH0。	R/W

注意: 只能在 P/E 模式下设置或清除此寄存器。

25.3.15 FWBH0 : Flash Write Buffer Register H0



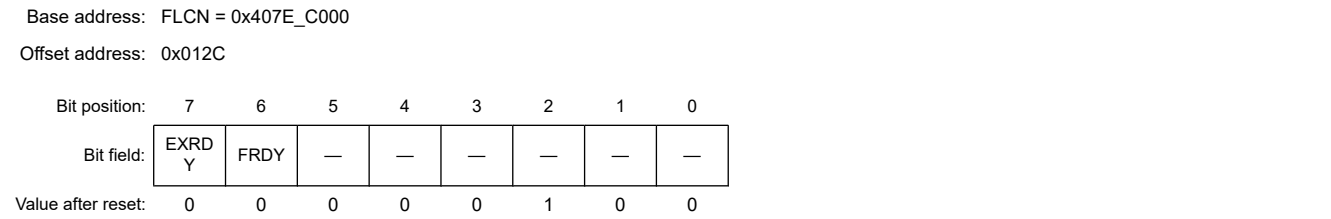
Bit	Symbol	Function	R/W
15:0	WDATA[15:0]	Flash Write Buffer H0 Flash write buffer data upper 16 bits	R/W

Note: Set or clear this register only in P/E mode.

The FWBH0 and FWBL0 registers set program data of the program command, the startup selection and security setting command, the access window information program command, and the OCDID program command. The following table describes how to set data according to each command.

Register	What is set to the register
FWBH0 FWBL0	- Bits [31:0] of the programming data of the program command for the code flash - Bits [31:0] of the programming data of the startup selection and security setting command, the access window information program command, and the OCDID program command.

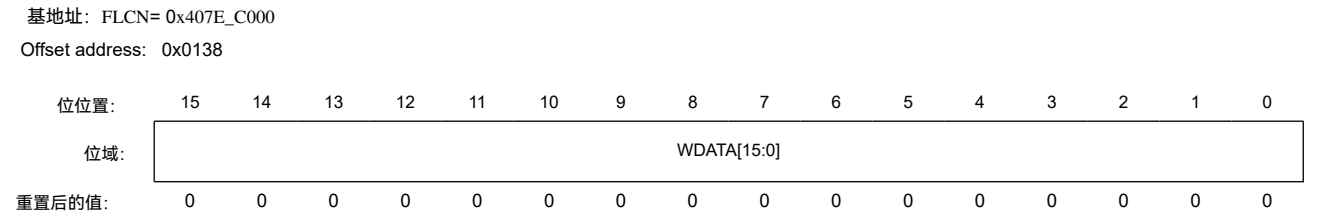
25.3.16 FSTATR1 : Flash Status Register 1



Bit	Symbol	Function	R/W
1:0	—	These bits are read as 0.	R
2	—	This bit is read as 1.	R
5:3	—	These bits are read as 0.	R
6	FRDY	Flash Ready Flag 0: The software command of the FCR register is not terminated. 1: The software command of the FCR register is terminated.	R
7	EXRDY	Extra Area Ready Flag 0: The software command of the FEXCR register is not terminated. 1: The software command of the FEXCR register is terminated.	R

FSTATR1 is a status register used to confirm the execution result of a software command. Each flag is set to 0 when the next software command is executed.

25.3.15 FWBH0: Flash 写入缓冲寄存器 H0



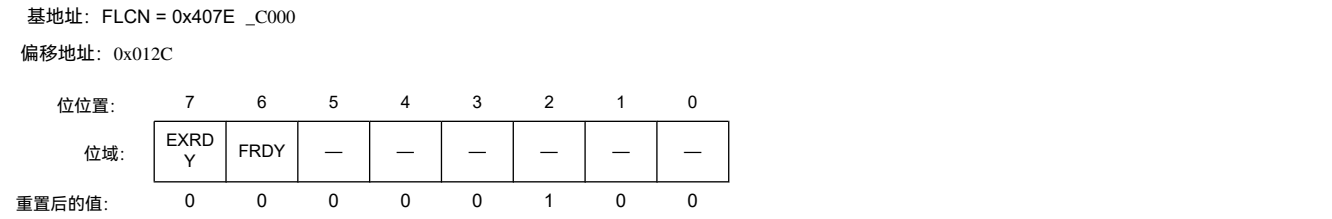
少量	象征	功能	R/W
15:0	WDATA[15:0]	闪存写入缓冲区 H0 闪存写入缓冲区数据高 16 位	R/W

注意: 只能在 P/E 模式下设置或清除此寄存器。

FWBH0 和 FWBL0 寄存器用于设置程序命令、启动选择和安全设置命令、访问窗口信息程序命令以及 OCDID 程序命令的程序数据。下表描述了如何根据每条命令设置数据。

登记	寄存器中设置了什么
FWBH0 FWBL0	- 代码闪存的程序命令编程数据的位[31:0] - 启动选择和安全设置命令的编程数据位[31:0]，访问窗口信息程序命令和 OCDID 程序命令。

25.3.16FSTATR1： 闪存状态寄存器 1



少量	象征	功能	R/W
1:0	—	这些位读取为 0。	R
2	—	这一位读作 1。	R
5:3	—	这些位读取为 0。	R
6	FRDY	闪光灯就绪标志 0: FCR寄存器的软件命令未终止。 1: FCR寄存器的软件命令已终止。	R
7	EXRDY	额外区域准备就绪旗帜 0: FEXCR 寄存器的软件命令未终止。 1: FEXCR 寄存器的软件命令已终止。	R

FSTATR1 是一个状态寄存器，用于确认软件命令的执行结果。当执行下一个软件命令时，每个标志位都会被置为 0。

25.3.17 FSTATR2 : Flash Status Register 2

Base address: FLCN = 0x407E_C000																
Offset address: 0x01F0																
Bit position:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Bit field:	—	—	—	—	—	—	—	—	—	—	EILGL ERR	ILGLE RR	BCER R	—	PRGE RR	ERER R
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Function	R/W
0	ERERR	Erase Error Flag 0: Erasure terminates normally 1: An error occurs during erasure	R
1	PRGERR	Program Error Flag 0: Programming terminates normally 1: An error occurs during programming.	R
2	—	This bit is read as 0.	R
3	BCERR	Blank Check Error Flag 0: Blank checking terminates normally 1: An error occurs during blank checking.	R
4	ILGLERR	Illegal Command Error Flag 0: No illegal software command or illegal access is detected 1: An illegal command or illegal access is detected.	R
5	EILGLERR	Extra Area Illegal Command Error Flag 0: No illegal command or illegal access to the extra area is detected 1: An illegal command or illegal access to the extra area is detected.	R
15:6	—	These bits are read as 0.	R

FSTATR2 is a status register used to confirm the execution result of a software command. Each error flag is set to 0 when the next software command is executed.

ERERR flag (Erase Error Flag)

The value of the ERERR bit is undefined when the FCR.STOP bit is set to 1 (processing is forcibly stopped) during erasure.

PRGERR flag (Program Error Flag)

The PRGERR bit is set when the program command of the FCR register or each command of the FEXCR register is abnormally terminated.

ILGLERR flag (Illegal Command Error Flag)

The ILGLERR flag indicates the execution of the software command of the FCR register with unexpected condition.

[Setting condition]

- Programming/erasure/read commands are executed to an area protected by the access window range
- The blank check and the block erase commands are executed when the start address set to the FSARH and FSARL registers is larger than the end address set to the FEARH and FEARL registers
- The program, the block erase and the blank check commands are executed when the FASR.EXS bit is 1
- The code flash is set to P/E mode simultaneously and a software command is executed.

[Clearing conditions]

- The next software command is executed.

EILGLERR flag (Extra Area Illegal Command Error Flag)

The EILGLERR flag indicates the execution of the software command of the FEXCR register with unexpected condition.

[Setting condition]

25.3.17 FSTATR2: Flash 状态寄存器 2

基地址: FLCN= 0x407E_C000																
偏移地址: 0x01F0																
位位置:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域:	—	—	—	—	—	—	—	—	—	—	EILGL ERR	ILGLE RR	BCER R	—	PRGE RR	ERER R
重置后的值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

少量	象征	功能	R/W
0	ERERR	擦除错误标志 0: 擦除操作正常终止。 1: 擦除过程中发生错误	R
1	PRGERR	程序错误标志 0: 程序正常终止 1: 程序运行过程中发生错误。	R
2	—	该位读取值为 0。	R
3	BCERR	空白支票错误标志 0: 空白支票正常终止 1: 空白支票检查期间发生错误。	R
4	ILGLERR	非法命令错误标志 0: 未检测到非法软件命令或非法访问。1: 检测到非法命令或非法访问。	R
5	EILGLERR	额外区域非法命令错误标志 0: 未检测到非法命令或非法访问额外区域。1: 检测到非法命令或非 法访问额外区域。	R
15:6	—	这些位读取为 0。	R

FSTATR2 是一个状态寄存器，用于确认软件命令的执行结果。每次执行下一个软件命令时，错误标志都会被置为 0。

ERERR 标志（擦除错误标志）

当擦除过程中 FCR.STOP 位设置为 1（强制停止处理）时，ERERR 位的值未定义。

PRGERR 标志（程序错误标志）

当 FCR 寄存器的程序命令或 FEXCR 寄存器的每个命令异常终止时，PRGERR 位将被设置。

ILGLERR 标志（非法命令错误标志）

ILGLERR 标志表示 FCR 寄存器的软件命令执行出现意外情况。

[设置条件]

- 编程/擦除/读取命令在受访问窗口范围保护的区域内执行。
- 当FSARH和FSARL寄存器设置的起始地址大于FEARH和FEARL寄存器设置的结束地址时，将执行空白检查和块擦除命令。
- 当 FASR.EXS 位为 1 时，执行程序、块擦除和空白检查命令。
- 代码闪存同时设置为P/E 模式，并执行软件命令。

[清算条件]

- 执行下一个软件命令。

EILGLERR 标志（额外区域非法命令错误标志）

EILGLERR 标志表示 FEXCR 寄存器的软件命令执行时出现意外情况。

[设置条件]

- The software commands of the FEXCR register is executed when the EXS bit of the FASR register is 0
- The access window information program command is executed when the FSPR bit is 0

[Clearing conditions]

- The next software command is executed.

25.3.18 FEAMH : Flash Error Address Monitor Register H

Base address:	FLCN = 0x407E_C000
Offset address:	0x01E8
Bit position:	1514131211109876543210
Bit field:	FEAMH[15:0]
Value after reset:	000000000000000000

Bit	Symbol	Function	R/W
15:0	FEAMH[15:0]	Flash Error Address Monitor Register H Flash error address monitor upper 16 bits See FEAML for details.	R

25.3.19 FEAML : Flash Error Address Monitor Register L

Base address:	FLCN = 0x407E_C000
Offset address:	0x01E0
Bit position:	1514131211109876543210
Bit field:	FEAML[15:0]
Value after reset:	000000000000000000

Bit	Symbol	Function	R/W
15:0	FEAML[15:0]	Flash Error Address Monitor Register L Flash error address monitor lower 16 bits	R

The error address is withdrawn from the FEAMH and FEAML registers after a software command execution. See [Figure 25.2](#) and [section 25.2. Memory Structure](#) for details on the addresses of the flash memory.

25.3.20 FSCMR : Flash Startup Setting Monitor Register

Base address:	FLCN = 0x407E_C000
Offset address:	0x01C0
Bit position:	1514131211109876543210
Bit field:	—FSPRSASMFS—
Value after reset:	0x ¹ 0000000x ¹ 0000000

Note 1. The reset value depends on the state of the extra area.

Bit	Symbol	Function	R/W
7:0	—	These bits are read as 0.	R
8	SASMF	Startup Area Setting Monitor Flag 0: Setting to start up using the alternative area 1: Setting to start up using the default area	R

- 当FASR寄存器的EXS位为0时，执行FEXCR寄存器的软件指令。
- 当 FSPR 位为 0 时，执行访问窗口信息程序命令。

[清算条件]

- 执行下一个软件命令。

25.3.18FEAMH： 闪存错误地址监视器寄存器 H

Base address:	FLCN = 0x407E_C000
Offset address:	0x01E8
位位置:	1514131211109876543210
位域:	FEAMH[15:0]
重置后的值:	000000000000000000

少量	象征	功能	R/W
15:0	FEAMH[15:0]	Flash 错误地址监视器寄存器 H Flash 错误地址监视器高 16 位有关详细信息，请参阅 FEAML。	R

25.3.19FEAML： 闪存错误地址监视器寄存器 L

Base address:	FLCN = 0x407E_C000
Offset address:	0x01E0
位位置:	1514131211109876543210
位域:	FEAML[15:0]
重置后的值:	000000000000000000

少量	象征	功能	R/W
15:0	FEAML[15:0]	闪存错误地址监视器寄存器 L 闪存错误地址监视器低 16 位	R

软件命令执行完毕后，错误地址将从 FEAMH 和 FEAML 寄存器中移除。参见图有关闪存地址的详细信息，请参阅第 25.2 节和第 25.2 节“存储器结构”。

25.3.20FSCMR：Flash 启动设置监视器寄存器

基地址:	FLCN = 0x407E_C000
偏移地址:	0x01C0
位位置:	1514131211109876543210
位域:	—FSPRSASMFS—
重置后的值:	0x ¹ 0000000x ¹ 0000000

注1. 复位值取决于额外区域的状态。

少量	象征	功能	R/W
7:0	—	这些位读取为 0。	R
8	SASMF	启动区域设置监视器标志 0: 设置为使用备用区域启动 1: 设置为使用默认区域启动	R

Bit	Symbol	Function	R/W
13:9	—	These bits are read as 0.	R
14	FSPR	Access Window Protection Flag 0: Access window setting disabled. 1: Access window setting enabled.	R
15	—	This bit is read as 0.	R

The FSCMR register monitors the extra area setting. Data of this register is updated at the reset sequence or execution of the software command of the FEXCR register.

25.3.21 FAWSMR : Flash Access Window Start Address Monitor Register

Base address: FLCN = 0x407E_C000															
Offset address: 0x01C8															
Bit position: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0															
Bit field:		FSPR	—	—	—	—	FAWS[10:0]								
Value after reset:		x	0	0	0	0	The value set by the user*1								

Note 1. The value of the blank product is 1. It is set to the same value set in bits [10:0] in the FWBH0 register after the access window information program command is executed.

Bit	Symbol	Function	R/W
10:0	FAWS[10:0]	Access Window Start Address This register is used to confirm the set value of the access window start address used for area protection	R
14:11	—	These bits are read as 0.	R
15	FSPR	Access Window Protection Flag This bit has the same value as the FSPR bit of the FSCMR register.	R

25.3.22 FAWEMR : Flash Access Window End Address Monitor Register

Base address: FLCN = 0x407E_C000															
Offset address: 0x01D0															
Bit position: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0															
Bit field:		SASMF	—	—	—	—	FAWE[10:0]								
Value after reset:		x	0	0	0	0	The value set by the user*1								

Note 1. The value of the blank product is 1. It is set to the same value set in bits [10:0] in the FWBL0 register after the access window information program command is executed.

Bit	Symbol	Function	R/W
10:0	FAWE[10:0]	Access Window End Address This register is used to confirm the set value of the access window end address used for area protection	R
14:11	—	These bits are read as 0.	R
15	SASMF	Startup Area Setting Monitor Flag This bit has the same value as the SASMF bit of the FSCMR register.	R

少量	象征	功能	R/W
13:9	—	这些位读取为 0。	R
14	FSPR	访问窗口保护标志：0：访问窗口设置已禁用；1：访问窗口设置已启用。	R
15	—	该位读取值为 0。	R

FSCMR 寄存器用于监控额外区域的设置。该寄存器的数据会在复位序列执行或执行 FEXCR 寄存器的软件命令时更新。

25.3.21FAWSMR： 闪存访问窗口起始地址监视器寄存器

基地址: FLCN = 0x407E_C000															
偏移地址: 0x01C8															
位位置: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0															
位域:		FSPR	—	—	—	—	FAWS[10:0]								
重置后的值:		x	0	0	0	0	用户设置的值*1								

注 1. 空白乘积的值为 1。执行访问窗口信息程序命令后， 它被设置为与 FWBH0 寄存器中的位[10:0] 中设置的值相同的值。

少量	象征	功能	R/W
10:0	FAWS[10:0]	访问窗口起始地址 该寄存器用于确认用于区域保护的访问窗口起始地址的设定值。	R
14:11	—	这些位读取为 0。	R
15	FSPR	访问窗口保护标志 该位的值与 FSCMR 寄存器的 FSPR 位相同。	R

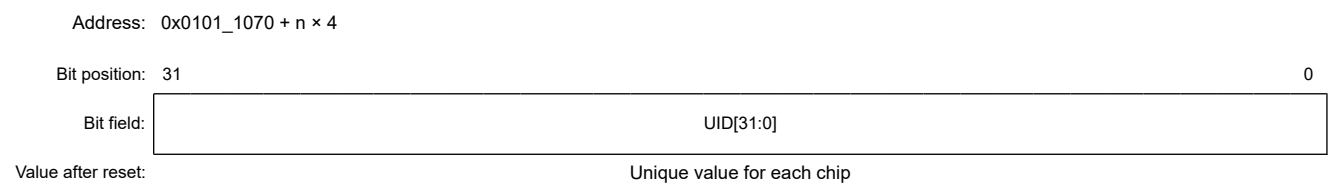
25.3.22FAWEMR： 闪存访问窗口结束地址监视器寄存器

基地址: FLCN = 0x407E_C000															
偏移地址: 0x01D0															
位位置: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0															
位域:		SASMF	—	—	—	—	FAWE[10:0]								
重置后的值:		x	0	0	0	0	用户设置的值*1								

注 1. 空白乘积的值为 1。执行访问窗口信息程序命令后， 它被设置为与 FWBL0 寄存器中的位[10:0] 中设置的值相同的值。

少量	象征	功能	R/W
10:0	FAWE[10:0]	访问窗口结束地址 该寄存器用于确认用于区域保护的访问窗口结束地址的设定值。	R
14:11	—	这些位读取为 0。	R
15	SASMF	启动区域设置监视器标志 该位的值与 FSCMR 寄存器的 SASMF 位相同。	R

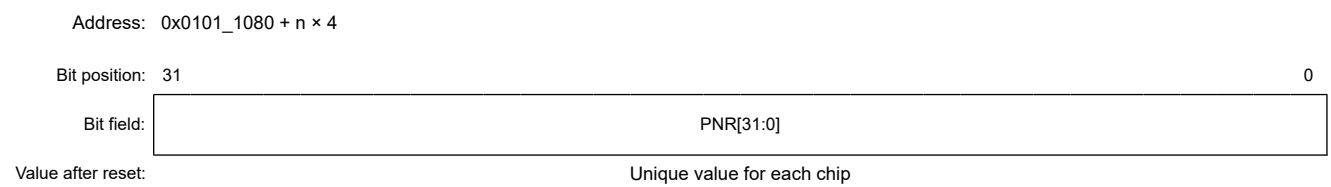
25.3.23 UIDRn : Unique ID Registers n (n = 0 to 3)



Bit	Symbol	Function	R/W
31:0	UID[31:0]	Unique ID	R

The UIDRn is a read-only register that stores a 16-byte ID code (unique ID) for identifying the individual MCU. The UIDRn register should be read in 32-bit units.

25.3.24 PNRn : Part Numbering Register n (n = 0 to 3)



Bit	Symbol	Function	R/W
31:0	PNR[31:0]	Part Number	R

The PNRn is a read-only register that stores a 16-byte part numbering. The PNRn register should be read in 32-bit units. Each byte corresponds to the ASCII code representation of the product part number as detailed in product list.

In case of the part number is 'R7FA0E3034ZSD', 16-byte part numbering is stored as follows.

Address 0x0101_1080: 'D', 0x44 in ASCII code

Address 0x0101_1081: 'S', 0x53 in ASCII code

Address 0x0101_1082: 'Z', 0x5A in ASCII code

Address 0x0101_1083: '4', 0x34 in ASCII code

Address 0x0101_1084: '3', 0x33 in ASCII code

Address 0x0101_1085: '0', 0x30 in ASCII code

Address 0x0101_1086: '3', 0x33 in ASCII code

Address 0x0101_1087: 'E', 0x45 in ASCII code

Address 0x0101_1088: '0', 0x30 in ASCII code

Address 0x0101_1089: 'A', 0x41 in ASCII code

Address 0x0101_108A: 'F', 0x46 in ASCII code

Address 0x0101_108B: '7', 0x37 in ASCII code

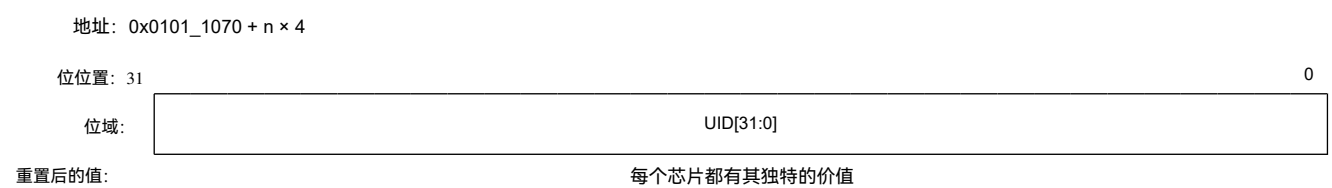
Address 0x0101_108C: 'R', 0x52 in ASCII code

Address 0x0101_108D: "(space)", 0x20 in ASCII code

Address 0x0101_108E: "(space)", 0x20 in ASCII code

Address 0x0101_108F: "(space)", 0x20 in ASCII code

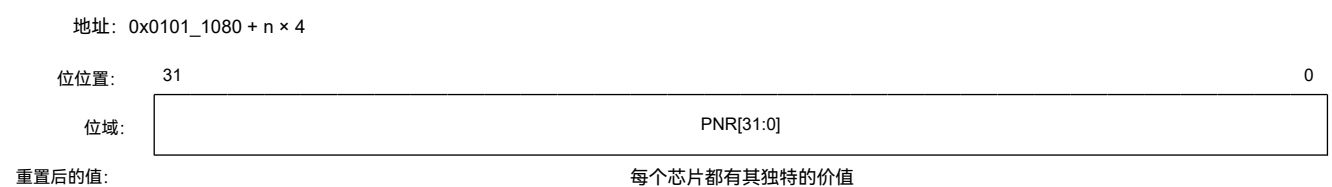
25.3.23 UIDRn: 唯一 ID 寄存器 n(n = 0 至 3)



少量	象征	功能	R/W
31:0	UID[31:0]	唯一标识符	R

UIDRn 是一个只读寄存器，用于存储一个 16 字节的 ID 代码（唯一 ID），以识别各个 MCU。UIDRn 寄存器应以 32 位为单位读取。

25.3.24 PNRn: 零件编号寄存器 n(n = 0 至 3)



少量	象征	功能	R/W
31:0	PNR[31:0]	零件编号	R

PNRn 是一个只读寄存器，用于存储 16 字节的零件编号。PNRn 寄存器应以 32 位为单位读取。每个字节对应于产品列表中详细列出的产品零件编号的 ASCII 代码表示。

如果零件编号为“R7FA0E3034ZSD”，则 16 字节的零件编号存储如下。

地址0x0101_1080: 'D',0x44的 ASCII 码

地址 0x0101_1081: 'S', ASCII 码为 0x53

地址 0x0101_1082: 'Z', ASCII 码为 0x5A

地址 0x0101_1083: ASCII 码中的“4”，0x34

地址 0x0101_1084: '3', ASCII 码为 0x33

地址 0x0101_1085: ASCII 码为 '0', 0x30

地址 0x0101_1086: ASCII 码为 '3', 0x33

地址 0x0101_1087: 'E', 0x45的 ASCII 码

地址 0x0101_1088: ASCII 码为 '0', 0x30

地址 0x0101_1089: 'A', ASCII 码为 0x41

地址 0x0101_108A: 'F', ASCII 码为 0x46

地址 0x0101_108B: '7', ASCII 码为 0x37

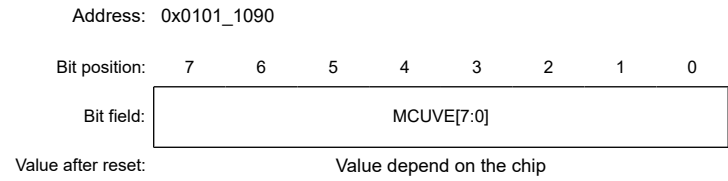
地址 0x0101_108C: 'R', ASCII 码为 0x52

地址 0x0101_108D: "(空格),0x20 的 ASCII 码

地址 0x0101_108E: "(空格),0x20 的 ASCII 码

地址 0x0101_108F: "(空格),0x20的 ASCII 码

25.3.25 MCUVER : MCU Version Register



Bit	Symbol	Function	R/W
7:0	MCUVE[7:0]	MCU Version	R

The MCUVER is a read-only register that stores an MCU version. The MCUVER register should be read in 8-bit units. The higher the value, the newer MCU version.

25.4 Operating Modes Associated with the Flash Memory

Figure 25.3 shows a diagram of the mode transitions associated with the flash memory. For information on setting up the modes, see section 3, Operating Modes.

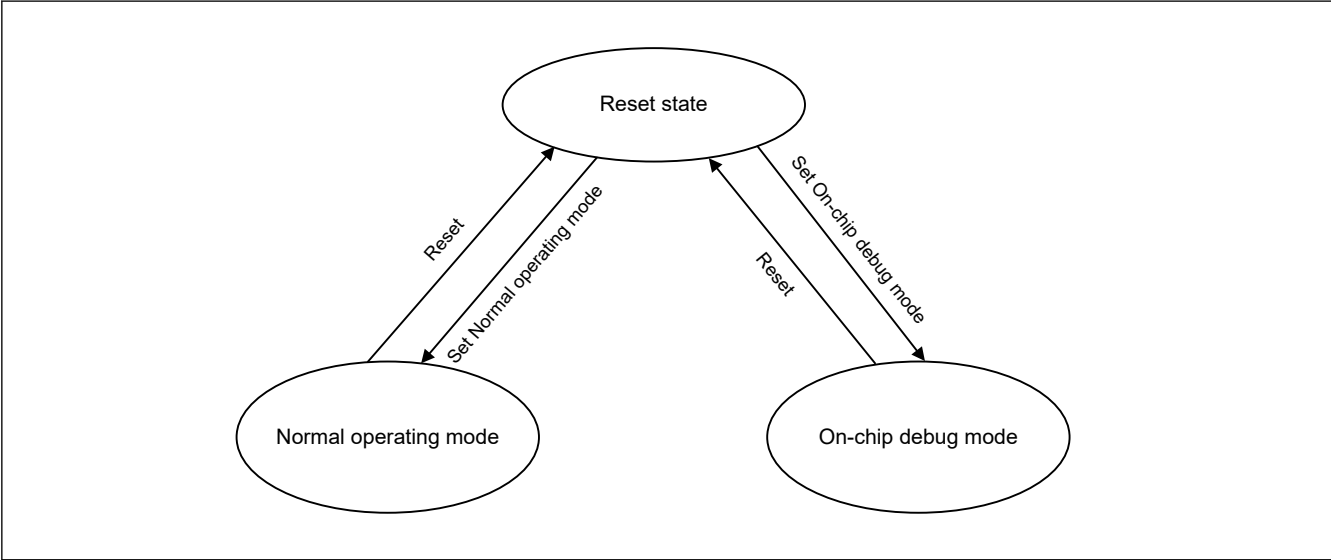


Figure 25.3 Mode transitions associated with flash memory

The flash memory areas where programming and erasure are permitted and where the boot program executes at a reset, differ with the mode. Table 25.11 shows the differences between the modes.

Table 25.11 Difference between modes

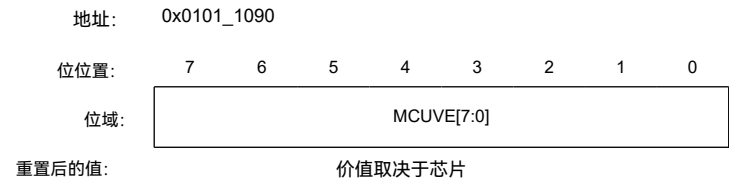
Parameter	Normal operating mode	On-chip debug mode
Programmable and erasable areas	• Code flash memory	• Code flash memory
Erasure in block units	Possible	Possible
Boot program at a reset	User area program	Depends on debug command

25.4.1 ID Code Protection

The ID code protection function prohibits programming and on-chip debugging. When ID code protection is enabled, the device validates or invalidates the ID code sent from the host by comparing it with the ID code stored in the flash memory. Programming and on-chip debugging are enabled only when the two match.

The ID code in flash memory consists of four 32-bit words. ID code bits [127] and [126] determine whether ID code protection is enabled and the authentication method to use with the host. Table 25.12 shows how the ID code determines the authentication method.

25.3.25 MCUVER: MCU 版本寄存器



少量	象征	功能	R/W
7:0	MCUVE[7:0]	MCU版本	R

MCUVER 是一个只读寄存器，用于存储 MCU 版本。MCUVER 寄存器应以 8 位为单位读取。值越高，表示 MCU 版本越新。

25.4 与闪存相关的操作模式

图 25.3 显示了与闪存相关的模式转换图。有关模式设置的信息，请参见第 3 节“操作模式”。

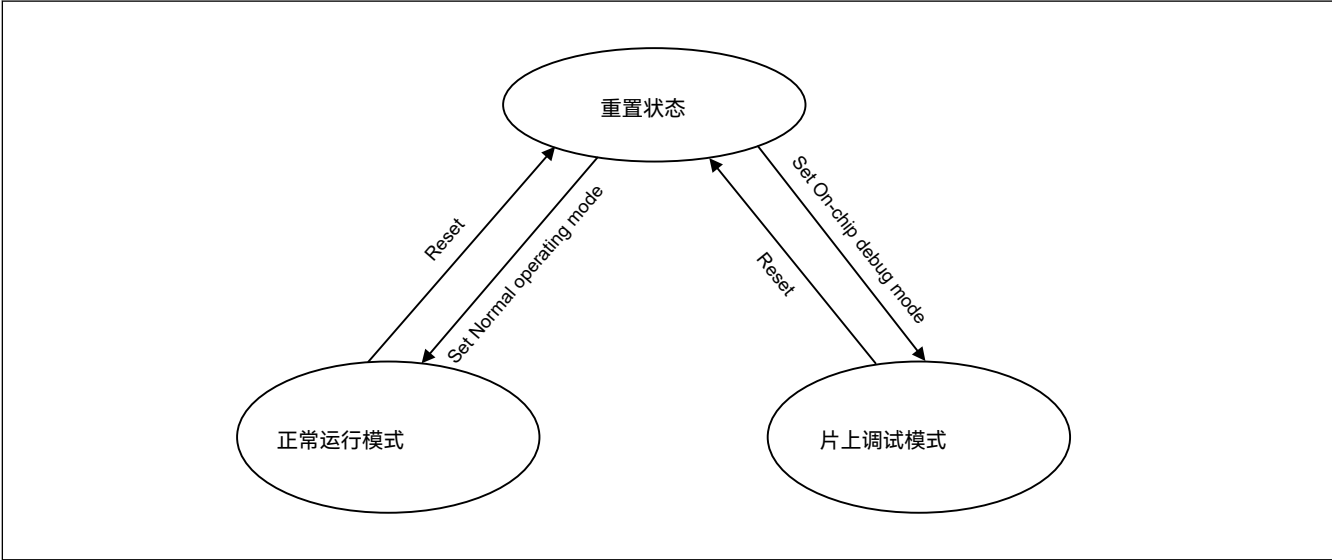


图 25.3 与闪存相关的模式转换

不同模式下，闪存中允许编程和擦除的区域以及复位时启动程序执行的区域有所不同。表 25.11 显示了不同模式之间的差异。

表 25.11 各模式之间的差异

范围	正常运行模式	片上调试模式
可编程和可擦除区域	●代码闪存	●代码闪存
以块为单位的擦除	可能的	可能的
复位时的启动程序	用户区程序	取决于调试命令

25.4.1 ID 代码保护

ID码保护功能会禁止对设备进行编程和片上调试。启用ID码保护后，设备会将主机发送的ID码与闪存中存储的ID码进行比较，以验证该ID码的有效性。只有当编程和片上调试条件匹配时，才能启用这两个条件。

闪存中的 ID 码由四个 32 位字组成。ID 码位 [127] 和 [126] 决定是否启用 ID 码保护以及与主机交互时使用的身份验证方法。表 25.12 显示了 ID 码如何确定身份验证方法。

Table 25.12 Specifications for ID code protection

Operating mode on boot up	ID code	State of protection	Operations on connection with the programmer or on-chip debugger
On-chip debug mode	0xFF, ..., 0xFF (all bytes 0xFF)	Protection disabled	The ID code is not checked, the ID code always matches, and the connection to the on-chip debugger or serial programmer*1 is permitted.
	Bit [127] = 1, bit [126] = 1, and at least one of all 16 bytes is not 0xFF	Protection enabled	Matching ID code indicates that authentication is complete and connection to the on-chip debugger or serial programmer is permitted. Mismatching ID code indicates transition to the ID code protection wait state. When the ID code sent from the on-chip debugger or serial programmer is ALeRASE in ASCII code (0x414C_6552_4153_45FF_FFFF_FFFF_FFFF_FFFF), the content of the user flash area is erased and all bits in the OSIS register are 1. However, when the AWS.FSPR bit is 0, the content of the user flash area is not erased.
	Bit [127] = 1 and bit [126] = 0	Protection enabled	Matching ID code indicates that authentication is complete and connection to the on-chip debugger or serial programmer is permitted. Mismatching ID code indicates transition to the ID code protection wait state.
	Bit [127] = 0	Protection enabled	The ID code is not checked, the ID code is always mismatching, the connection to the on-chip debugger or serial programmer is prohibited.

Note 1. Never send the ID code from on-chip debugger. Or send ID code 0xFF (all bytes 0xFF) from on-chip debugger.

25.5 Overview of Functions

By using a dedicated flash-memory programmer to program the on-chip flash memory through SWD interface (on-chip debug mode), the device can be programmed before or after it is mounted on the target system. Additionally, security functions to prohibit overwriting of the user program prevent tampering by third parties.

Programming by the user program (self-programming) is available for applications that might require updating after system manufacturing or shipment. Protection features for safely overwriting the flash memory area are also provided. Additionally, interrupt processing during self-programming is supported so that programming can proceed while processing external communications and other functions. Table 25.13 lists the programming methods and the associated operating modes.

Table 25.13 Programming methods

Programming method	Functional overview	Operating mode
Self-programming	A user program written to memory can also program the flash memory.	Normal operating mode
SWD programming	A dedicated flash-memory programmer or an on-chip debugger connected through SWD can program the on-board flash memory after the device is mounted on the target system.	On-chip debug mode
	A dedicated flash-memory programmer or an on-chip debugger connected through SWD and a dedicated programming adapter board allow off-board programming of the flash memory, before it is mounted on the target system.	

Table 25.14 lists the functions of the on-chip flash memory. For self-programming, use the programming commands to read the on-chip flash memory or run the user program.

表 25.12 ID 代码保护规范

启动时的操作模式	ID 代码	保护状态	与编程器或片上调试器连接的操作
片上调试模式	0xFF, ..., 0xFF (所有字节均为 0xFF)	保护已禁用	不检查 ID 代码, ID 代码始终匹配, 允许连接到片上调试器或串行编程器*1。
	位[127] = 1, 位[126] = 1, 且所有 16 个字节中至少有一个不为 0xFF。	已启用保护	匹配的 ID 代码表示身份验证完成, 并且允许连接到片上调试器或串行编程器。 ID 代码不匹配表示进入 ID 代码保护等待状态。 当片上调试器或串行编程器发送的 ID 代码为 ASCII 码 ALeRASE 时 (0x414C_6552_4153_45FF_FFFF_FFFF_FFFF_FFFF) 用户闪存区域的内容被擦除, OSIS 寄存器中的所有位均为 1。 但是, 当 AWS.FSPR 位为 0 时, 用户闪存区域的内容不会被擦除。
	位[127] = 1 且位[126] = 0	已启用保护	匹配的 ID 代码表示身份验证完成, 并且允许连接到片上调试器或串行编程器。 ID 代码不匹配表示进入 ID 代码保护等待状态。
	位[127] = 0	已启用保护	ID 代码未进行检查, ID 代码始终不匹配, 禁止连接到片上调试器或串行编程器。

注1: 切勿从片上调试器发送ID代码。也不要从片上调试器发送ID代码0xFF(所有字节均为0xFF)。

25.5 函数概述

通过使用专用闪存编程器, 经由SWD接口(片上调试模式)对片上闪存进行编程, 可以在器件安装到目标系统之前或之后进行编程。此外, 禁止覆盖用户程序的安全功能可防止第三方篡改。

用户程序编程(自编程)适用于系统制造或发货后可能需要更新的应用。此外, 还提供了用于安全覆盖闪存区域的保护功能。同时, 支持自编程期间的中断处理, 以便在处理外部通信和其他功能的同时进行编程。表 25.13 列出了编程方法及其相关的操作模式。

表 25.13 编程方法

编程方法	功能概述	操作模式
自编程	写入内存的用户程序也可以对闪存进行编程。	正常运行模式
SWD编程	将设备安装到目标系统后, 可以使用专用闪存编程器或通过 SWD 连接的片上调试器对板载闪存进行编程。	片上调试模式
	通过 SWD 连接的专用闪存编程器或片上调试器以及专用编程适配器板, 可以在将闪存安装到目标系统之前对其进行板外编程。	

表25.14列出了片上闪存的功能。对于自编程, 可以使用编程命令读取片上闪存或运行用户程序。

Table 25.14 Basic functions

Function	Functional overview	Availability	
		Self-programming	SWD programming
Blank check	Checks a specified block to ensure that writing to it has not already proceeded.	Supported	Supported
Block erasure	Erases the memory contents in the specified block	Supported	Supported
Programming	Writes to the specified address	Supported	Supported
Read	Reads data programmed in the flash memory	Not supported (read by user program is possible)	Not supported
ID code check	Compares the ID code sent by the host with the code stored in the code flash memory. If the two match, the FCB enters the wait state for programming and erasure commands from the host.	Not supported (ID authentication is not performed)	Supported
Security configuration	Configures the protection of security function (Access window and Start-up area selection)	Supported with conditions See section 25.8. Protection	Supported with conditions See section 25.8. Protection
Protection configuration	Configures the access window for flash area protection in the code flash memory	Supported	Supported

The on-chip flash memory supports the ID code check function. Authentication of ID code check is a security function for use with SWD programming. [Table 25.15](#) lists the available operations and security settings.

Table 25.15 Available operations and security settings

Function	All security settings and erasure, programming, and read operations		Constraints on the security setting configuration
	On-chip debug mode	Self-programming mode	
ID authentication	When the ID codes do not match: - Block erasure commands: not supported - Programming commands: not supported - Read commands: not supported - Security configuration commands: not supported - Protection configuration commands: not supported. When the ID codes match: - Block erasure commands: supported - Programming commands: supported - Read commands: supported - Security configuration commands: supported - Protection configuration commands: supported.	- Blank check: supported - Block erasure: supported - Programming: supported - Security configuration: supported - Protection configuration: supported.	ID authentication is not performed in the Self-programming mode.

25.5.1 Configuration Area Bit Map

The bits used for ID authentication, startup area select, access window protection, and security configuration functions are mapped in [Figure 25.4](#). The boot program must use these bits as hexadecimal data.

表 25.14 基本功能

功能	功能概述	可用性	
		自编程	SWD编程
空白支票	检查指定的数据块，确保尚未对其进行写入操作。	支持	支持
块擦除	擦除指定内存块中的内容	支持	支持
编程	写入指定地址	支持	支持
读	读取闪存中编程的数据	不支持（但用户程序可以读取）	不支持
ID码检查	将主机发送的ID代码与存储在代码闪存中的代码进行比较。如果两者匹配，FCB进入等待状态，接收来自主机的编程和擦除命令。	不支持（不执行身份验证）	支持
安全配置	配置安全功能保护（访问窗口和启动区域选择）	有条件支持，参见第 25.8 节。 保护	有条件支持，参见第 25.8 节。 保护
保护配置	配置代码闪存中闪存区域保护的访问窗口	支持	支持

片上闪存支持ID码校验功能。ID码校验是一项用于SWD编程的安全功能。表25.15列出了可用的操作和安全设置。

表 25.15 可用操作和安全设置

功能	所有安全设置以及擦除、编程和读取操作		安全性的限制 设置配置
	片上调试模式	自编程模式	
身份认证	当 ID 代码不匹配时： ●块擦除命令：不支持 ●编程命令：不支持 ●读取命令：不支持 ●安全配置命令：不支持 ●保护配置命令：不支持。 当 ID 代码匹配时： ●块擦除命令：支持 ●编程命令：支持 ●读取命令：支持 ●安全配置命令：支持 ●保护配置命令：支持。	●空白支票：支持 ●块擦除：支持 ●编程：支持 ●安全配置：有 ●保护配置：已支持。	自编程模式下不执行身份认证。

25.5.1 配置区域位图

图 25.4 映射了用于ID身份验证、启动区域选择、访问窗口保护和安全配置功能的位。引导程序必须将这些位用作十六进制数据。

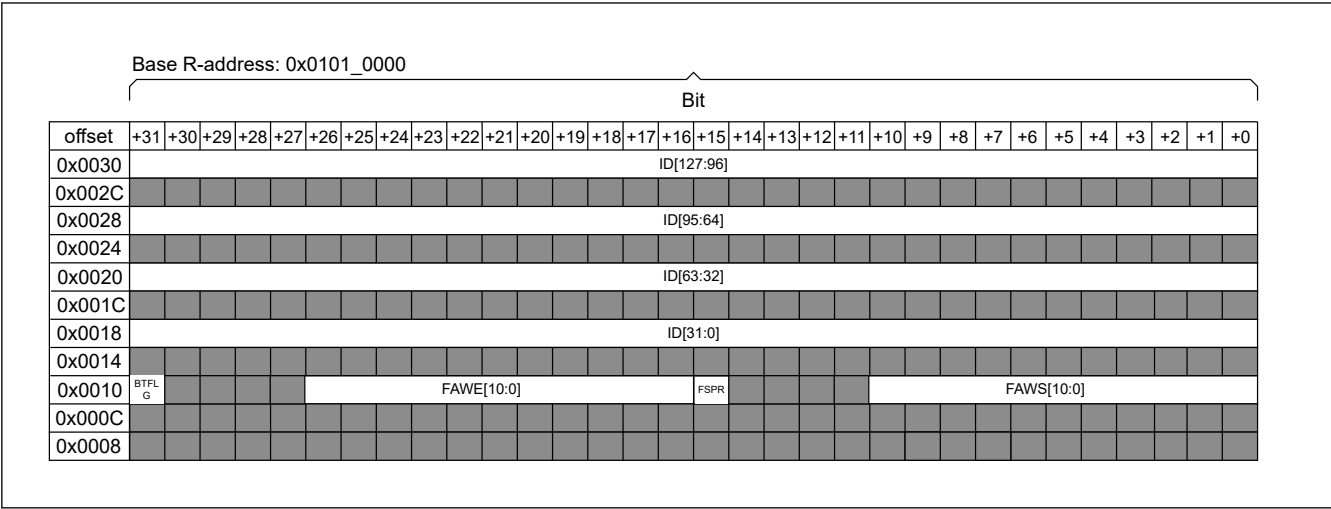


Figure 25.4 Configuration area bit map

25.5.2 Startup Area Select

The startup area select function allows the boot program to be safely updated. The startup area is 4 KB of space located in the user area. The FCB controls the address of the startup area based on the Startup Area Select Flag (BTFLG) that is located in the configuration area which names as AWS register. The startup area can be locked by the FSPR bit.

Figure 25.5 shows an overview of the startup program protection.

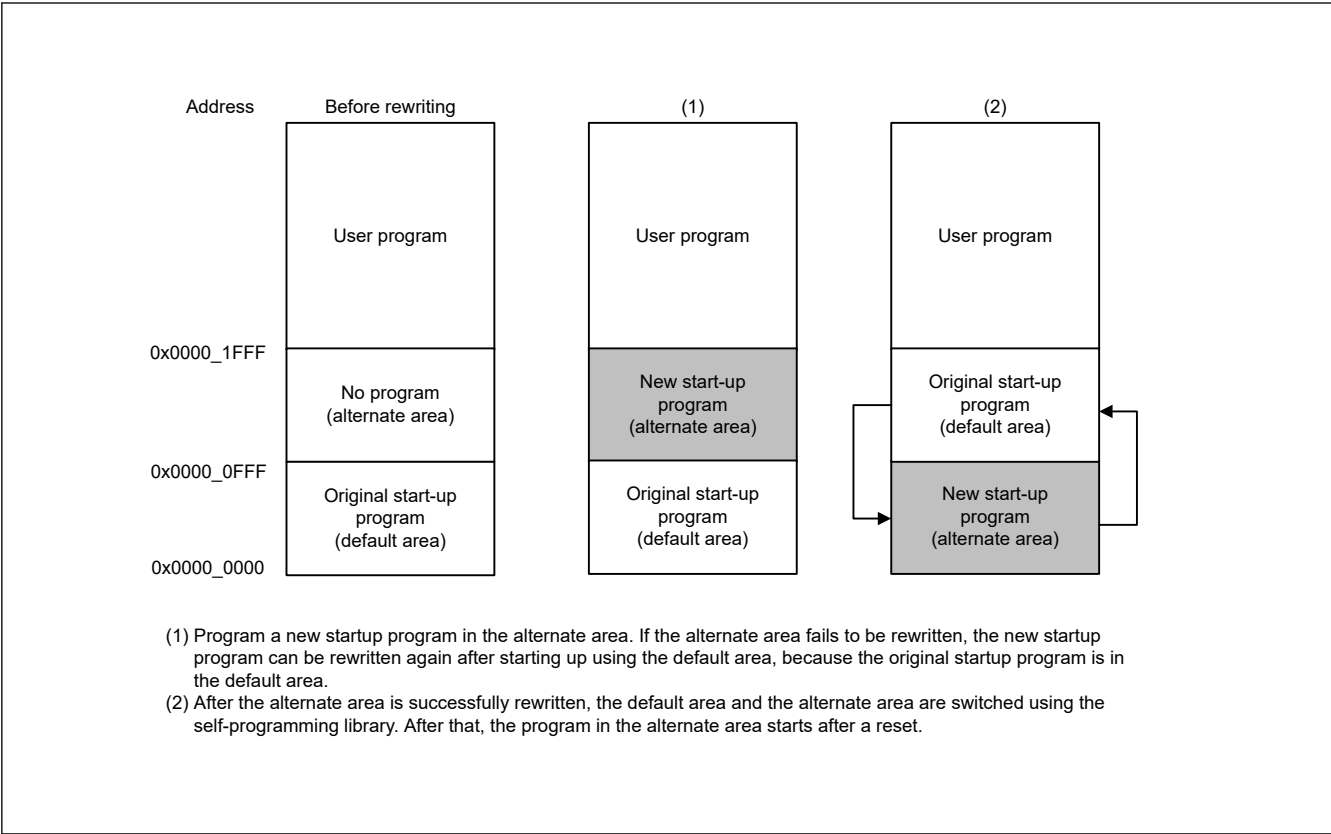


Figure 25.5 Overview of startup program protection

25.5.3 Protection by Access Window

Issuing the program or block erase command to a flash memory area outside the access window results in the command-locked state. The access window is only valid in the user area of the code flash memory. The access window provides protection in self-programming, and on-chip debug modes. Figure 25.6 shows an overview of flash area protection.

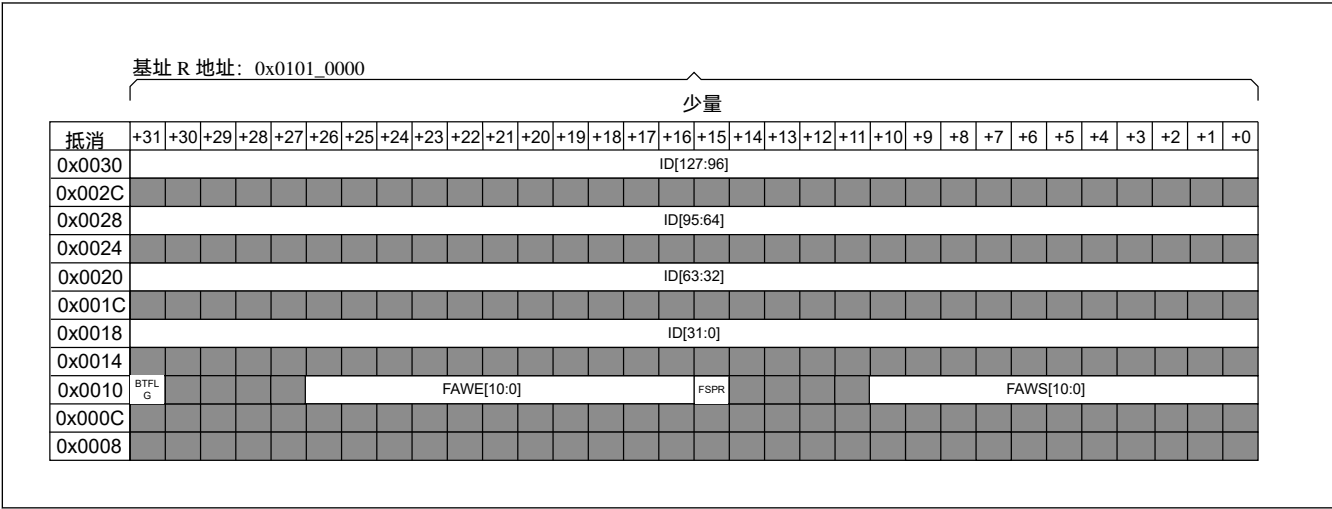


图 25.4 配置区域位图

25.5.2 启动区域选择

启动区选择功能允许安全地更新引导程序。启动区位于用户区，占用空间为4 KB。功能控制块 (FCB) 根据位于配置区（名为 AWS 寄存器）中的启动区选择标志 (BTFLG) 控制启动区的地址。启动区可以通过 FSPR 位锁定。

图 25.5 显示了启动程序保护的概览。

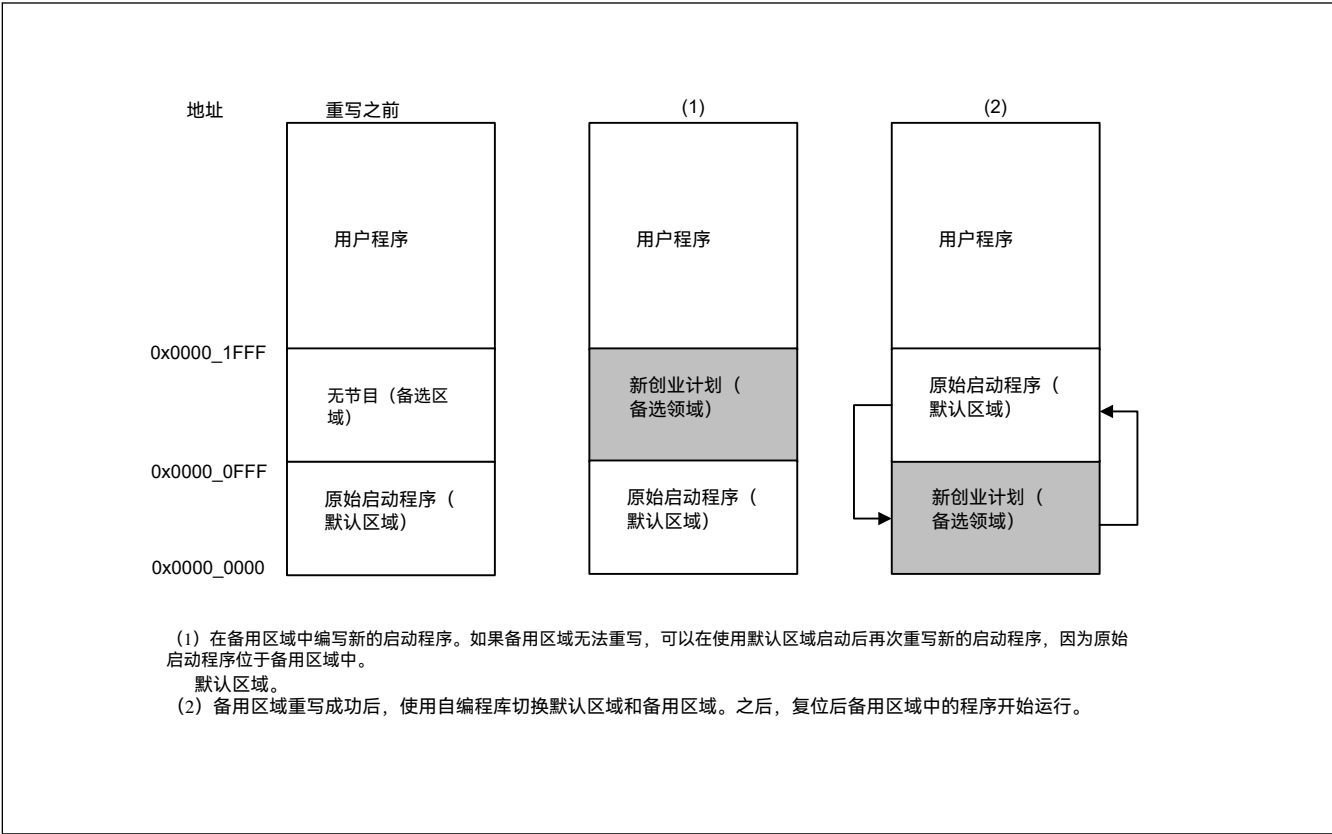


图 25.5 启动程序保护概述

25.5.3 通过访问窗口进行保护

在访问窗口之外向闪存区域发出程序擦除或块擦除命令会导致命令锁定状态。访问窗口仅在代码闪存的用户区域有效。访问窗口在自编程和片上调试模式下提供保护。图 25.6 显示了闪存区域保护的概览。

The access window is specified in both the FAWS [10:0] and FAWE [10:0] bits. See [section 6.2.3. AWS : Access Window Setting Register](#). Setting of the FAWE[10:0] and FAWS[10:0] bits in various conditions is described as follows:

- FAWE [10:0] = FAWS [10:0]: The P/E command can execute anywhere in the user area of the code flash memory
- FAWE [10:0] > FAWS [10:0]: The P/E command can only execute in the window from the block pointed to by the FAWS bits to one block lower than the block pointed to by the FAWE[10:0] bits
- FAWE [10:0] < FAWS [10:0]: The P/E command cannot execute anywhere in the user area of the code flash memory.

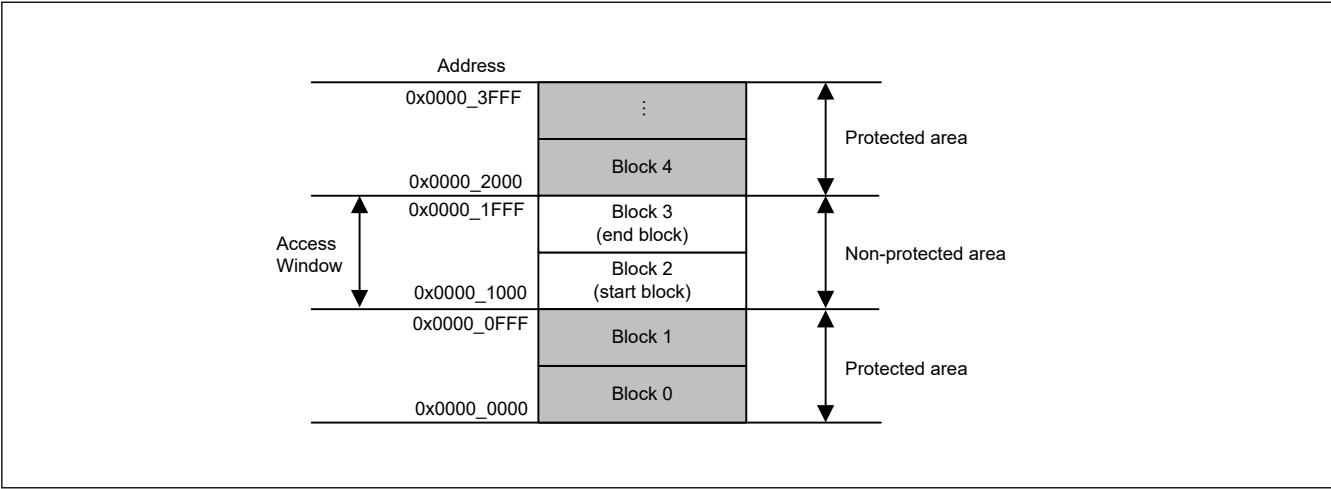


Figure 25.6 Flash area protection overview

25.6 Programming Commands

The FCB controls the programming commands.

25.7 Suspend Operation

The forced stop command forces the blank check command or the block erase command to stop. When a forced stop is executed, the stopped address values are stored in the registers. The command can restart from the stopped address after a reset to the registers for command execution by copying the saved addresses.

25.8 Protection

The types of protection provided include:

- Startup Program Protection
- Area Protection

25.8.1 Startup Program Protection

When programming of the startup area is interrupted by temporary blackout, the startup program may not be successfully programmed and the user program may not start properly.

This problem can be avoided by programming the startup program without erasing the existing startup program using the startup program protection.

Figure 25.7 shows an overview of the Startup Program Protection. In this figure, the default area indicates the 4-KB region from the start address and the alternate area indicates the next 4-KB region.

访问窗口在 FAWS[10:0] 和 FAWE[10:0] 位中均有指定。参见 6.2.3 节。AWS: 访问窗口设置寄存器。FAWE[10:0] 和 FAWS[10:0] 位在各种情况下的设置如下所述:

- FAWE[10:0] = FAWS [10:0]:P/E命令可以在代码闪存的用户区域内的任何位置执行
- FAWE[10:0] >FAWS[10:0]:P/E命令只能在由指向的块的窗口中执行
FAWS 位指向比 FAWE[10:0] 位指向的块低一个块的块
- FAWE[10:0] <FAWS[10:0]:P/E命令不能在代码闪存的用户区域内的任何位置执行。

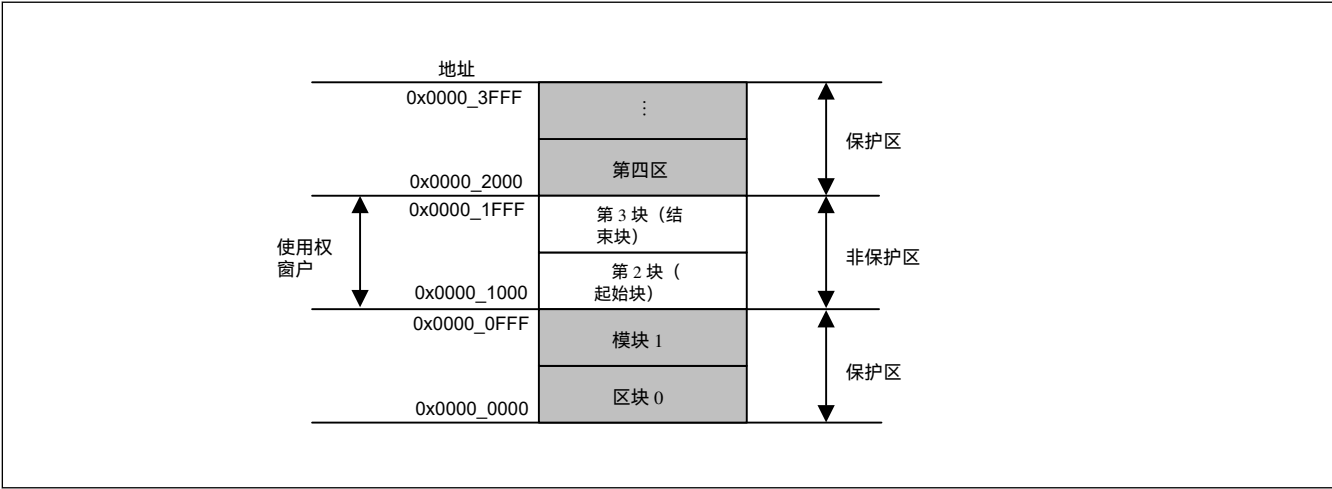


图 25.6 闪光区域保护概览

25.6 编程命令

FCB 控制编程指令。

25.7 暂停操作

强制停止命令会强制停止空白检查命令或块擦除命令。执行强制停止时，停止的地址值会存储在寄存器中。复位寄存器后，可以通过复制已保存的地址，使命令从停止的地址重新启动执行。

25.8 保护

提供的保护类型包括:

- 初创企业保护计划
- 区域保护

25.8.1 初创企业保护

当启动区域的编程因临时断电而中断时，启动程序可能无法成功编程，用户程序也可能无法正常启动。

通过使用启动程序保护功能，在不擦除现有启动程序的情况下对启动程序进行编程，可以避免这个问题。

图 25.7 显示了启动程序保护的概览。图中，默认区域表示从起始地址开始的 4 KB 区域，备用区域表示接下来的 4 KB 区域。

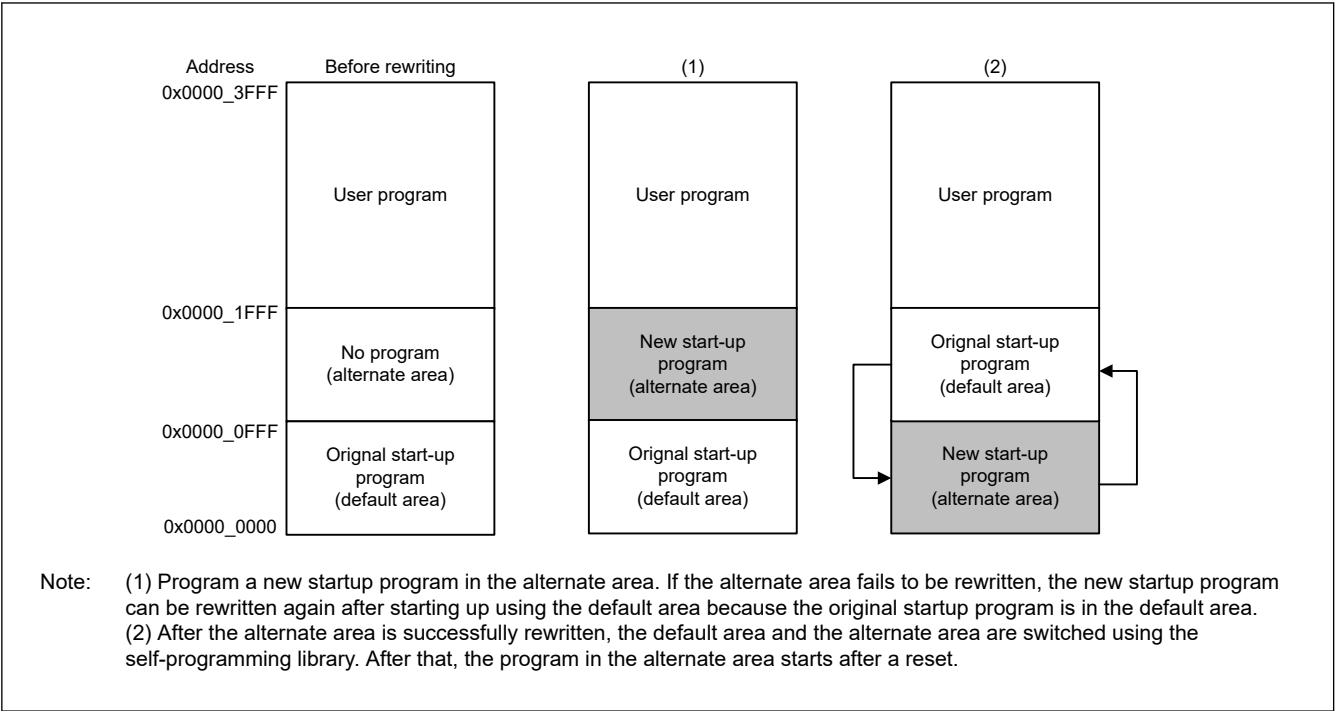


Figure 25.7 Overview of the startup program protection

25.8.2 Area Protection

Area protection enables rewriting for only selected blocks (access window) in the user area and disables programming for the other blocks.

Select the start block and end block to set the access window. The access window is changeable and valid in programming mode (self-programming mode and OCD mode).

Figure 25.8 shows an overview of area protection.

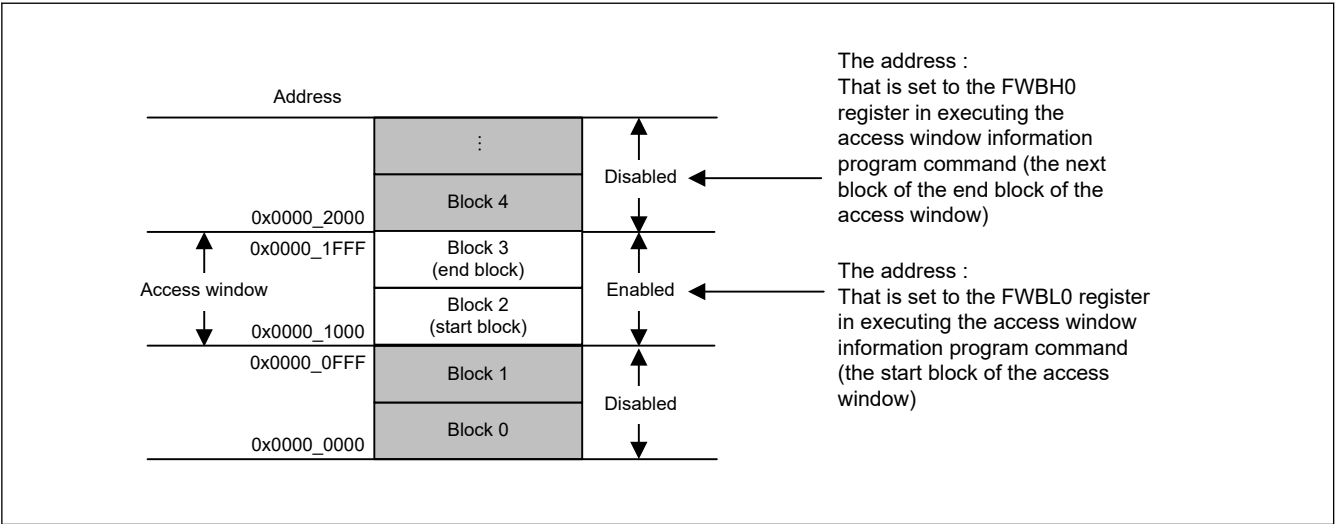


Figure 25.8 Area protection overview

25.9 Self-programming

25.9.1 Overview

The MCU supports programming of the flash memory by the user program. The programming commands can be used with user programs for writing to the code flash memory. This enables updates to the user programs and overwriting of constant data fields.

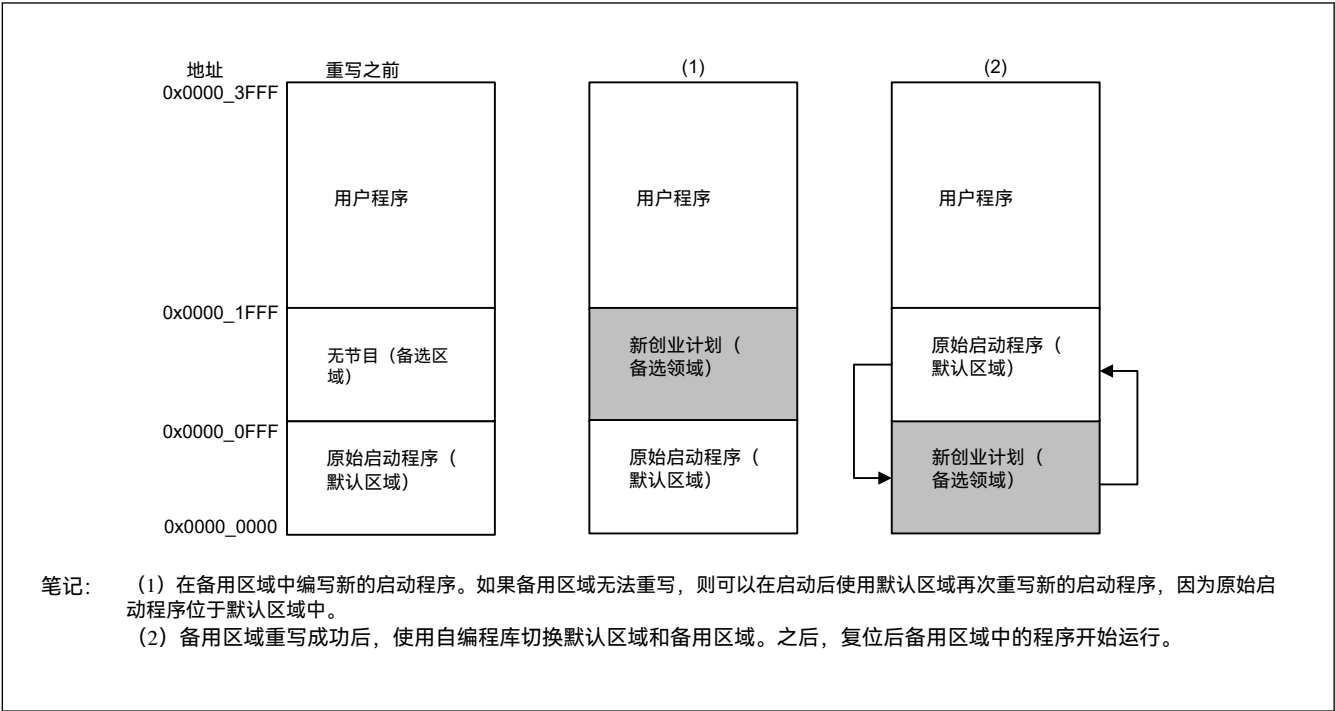


图 25.7 启动程序保护概述

25.8.2 区域保护

区域保护功能仅允许对用户区域中选定的块 (访问窗口) 进行重写, 并禁用对其他块的编程。

选择起始块和结束块以设置访问窗口。访问窗口可更改, 并且在编程模式 (自编程模式和OCD模式) 下有效。

图 25.8 显示了区域保护概览。

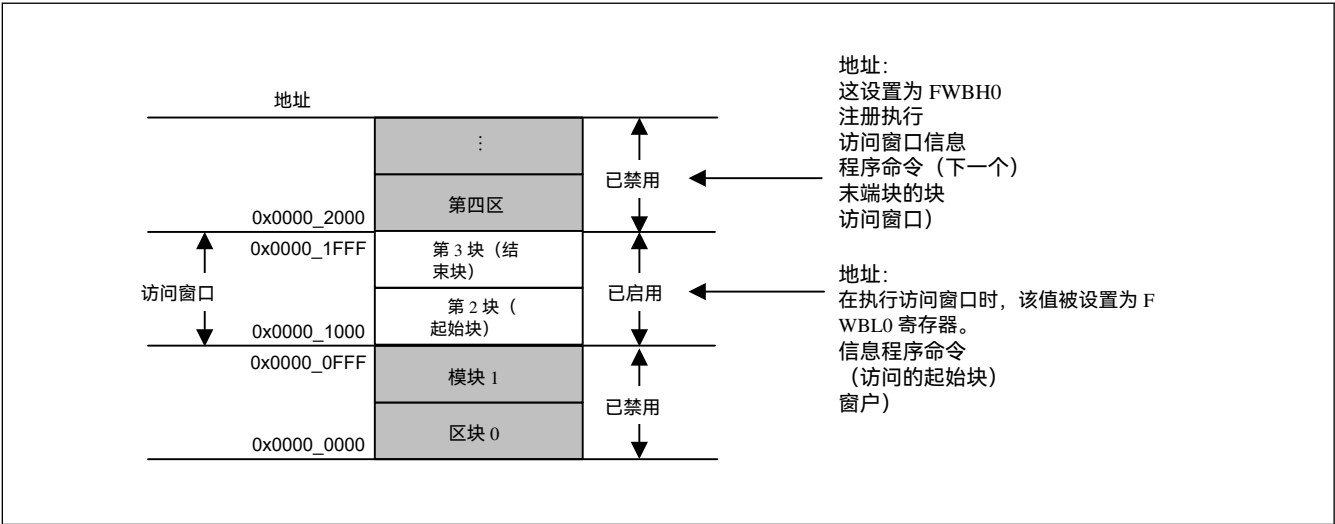


图 25.8 区域保护概览

25.9 自编程

25.9.1 概述

MCU支持用户程序对闪存进行编程。用户程序可以使用编程命令将代码写入闪存。这使得用户程序可以进行更新, 并可以覆盖常量数据字段。

In self-programming, it is necessary to supply a stable HOCO clock to the flash memory in order to generate the program voltage and erase voltage. Therefore, in case that the HOCO is stopped where another clock source is selected as the system clock, it is necessary to start the HOCO operation and ensure that the oscillation is in a stable state before executing the self-programming. For details of HOCO clock oscillation stabilization check, see [section 8.2.5. OSCSF : Oscillation Stabilization Flag Register](#).

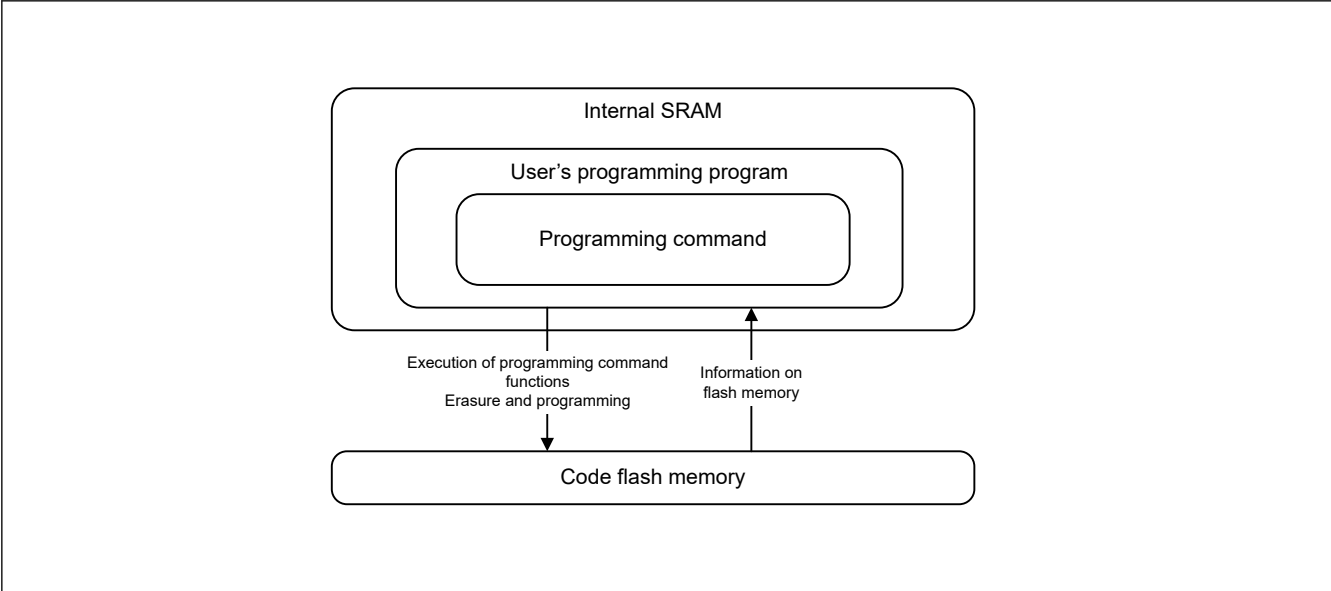


Figure 25.9 Schematic view of self-programming

25.10 Programming and Erasure

The code flash can be programmed and erased by changing the mode of the dedicated sequencer for programming and erasure, and by issuing commands for programming and erasure. The mode transitions and commands required to program or erase the code flash is described in the sections that follow. The descriptions apply in common to single-chip mode.

25.10.1 Sequencer Modes

The sequencer has two modes and transitions between modes occur by issuing commands to set the FPMCR register. [Figure 25.10](#) shows mode transitions of the flash memory.

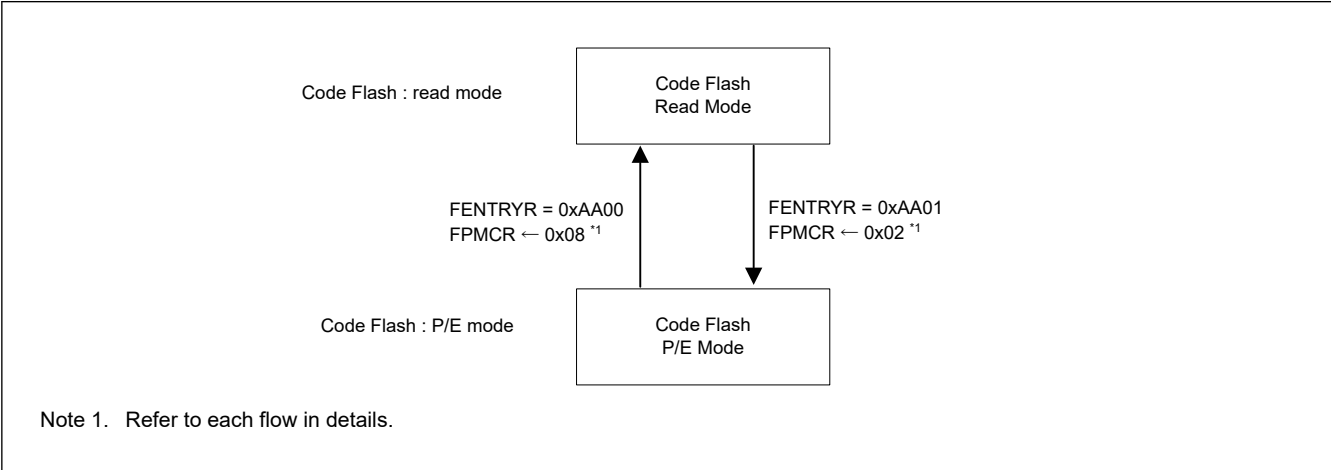


Figure 25.10 Mode transitions of the flash memory

25.10.1.1 Read Mode

Read mode is used for high-speed reading of the code flash.

在自编程过程中，需要向闪存提供稳定的 HOCO 时钟，以产生编程电压和擦除电压。因此，如果 HOCO 停止运行，并且选择了另一个时钟源作为系统时钟，则必须重新启动 HOCO 操作，并确保振荡处于稳定状态后再执行自编程。有关 HOCO 时钟振荡稳定性检查的详细信息，请参见 8.2.5 节。OSCSF：振荡稳定标志寄存器。

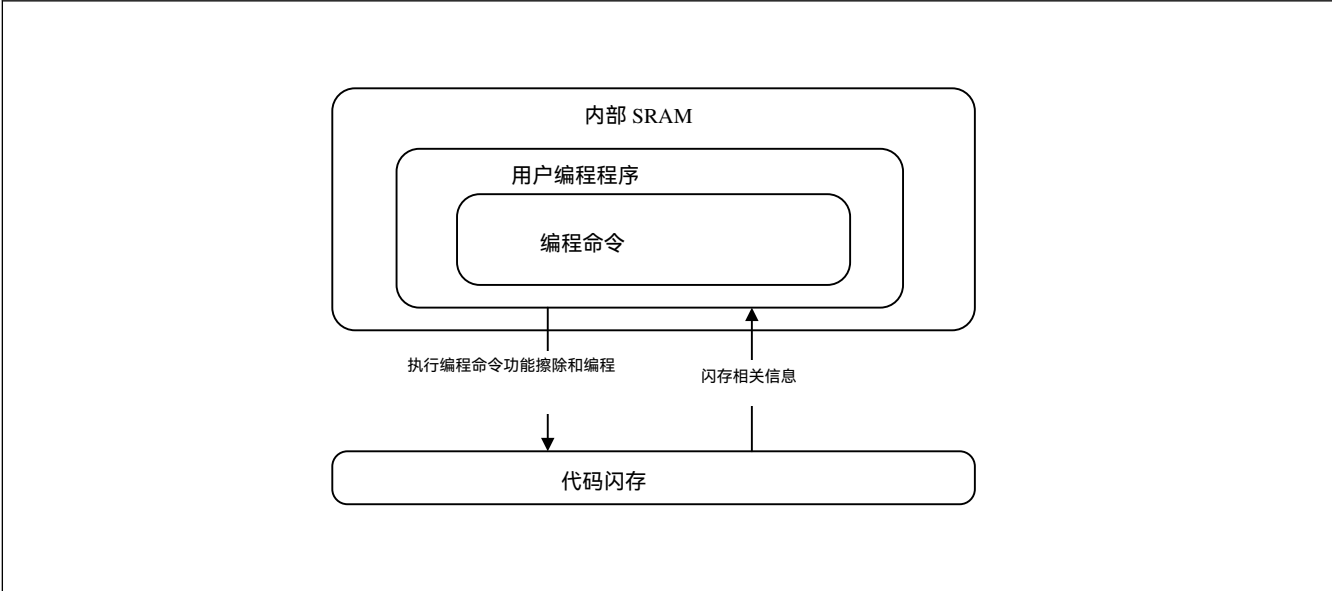


图 25.9 自编程示意图

25.10编程和擦除

可以通过改变专用编程和擦除序列器的模式，以及发出编程和擦除命令来对代码闪存进行编程和擦除。

以下各节将介绍模式转换以及对代码闪存进行编程或擦除所需的命令。这些描述同样适用于单芯片模式。

25.10.1 音序器模式

序列器有两种模式，模式之间的转换是通过发出设置 FPMCR 寄存器的命令来实现的。[25.10](#) 显示了闪存的模式转换。

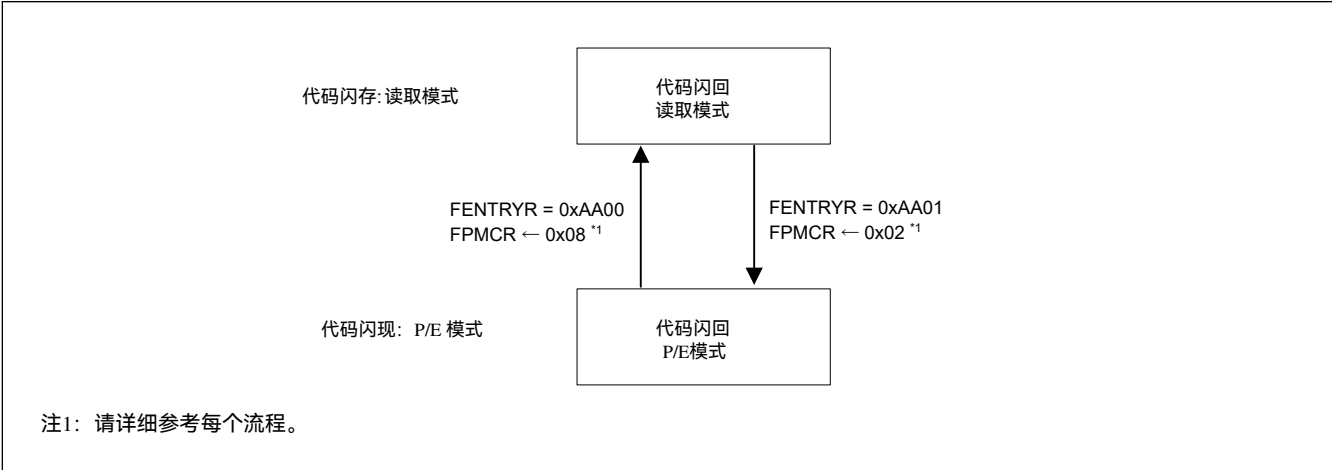


图 25.10 闪存的模式转换

25.10.1.1读取模式

读取模式用于高速读取代码闪存。

(1) Code Flash Read Mode

This mode is used for reading the code flash. The sequencer enters this mode when the FENTRYR.FENTRY0 bit is set to 0.

25.10.1.2 P/E Modes

(1) Code Flash P/E Mode

The code flash P/E mode is used for programming and erasure of the code flash. The sequencer enters this mode when the FENTRYR.FENTRY0 bit is set to 1.

25.10.2 Software Commands

Software commands consist of commands for programming and erasure, and commands for programming startup program area information and access window information. [Table 25.16](#) lists the software commands for use with the flash memory.

Table 25.16 Software commands

Command	Function
Program	Code flash programming (4 bytes)
Block erase	Code flash erasure
Blank check	Check whether the specified area is blank. Confirm that data is not programmed in the area. This command does not guarantee whether the area remains erased.
Startup area information and security program	Set the FSPR or the SASMF to the extra area
Access window information program	Set the access window used for area protection to the extra area
OCDID program	Set the OCDID to the extra area

25.10.3 Software Command Usage

The following sections describe the usage of each software command.

(1) Switching to Code Flash P/E Mode

It is necessary to enter the code flash P/E mode by setting the FENTRY0 bit of the FENTRYR register before executing the software command for the code flash. [Figure 25.11](#) shows the procedure for entering code flash P/E Mode.

(1) 代码闪读模式

此模式用于读取代码闪存。当 FENTRYR.FENTRY0 位设置为 0 时，序列器进入此模式。

25.10.1.2P/E 模式

(1) 代码闪光 P/E 模式

代码闪存模式P/E 用于代码闪存的编程和擦除。当 FENTRYR.FENTRY0 位设置为 1 时，序列器进入此模式。

25.10.2 软件命令

软件命令包括编程和擦除命令，以及用于编程启动程序区信息和访问窗口信息的命令。表 25.16 列出了闪存使用的软件命令。

表 25.16 软件命令

命令	功能
程序	代码闪存编程（4字节）
块擦除	代码闪存擦除
空白支票	检查指定区域是否为空白。 确认该区域内未写入任何数据。此命令不能保证该区域始终保持擦除状态。
初创领域信息和安全计划	将 FSPR 或 SASMF 设置为额外区域
访问窗口信息程序	将用于区域保护的访问窗口设置为额外区域
强迫症项目	将 OCDID 设置为额外区域

25.10.3 软件命令用法

以下各节将介绍每个软件命令的用法。

(1) 切换到代码闪断 P/E 模式

在执行代码闪存的软件命令之前，必须通过设置 FENTRYR 寄存器的 FENTRY0 位来进入代码闪存P/E模式。图 25.11 显示了进入代码闪存P/E模式的步骤。

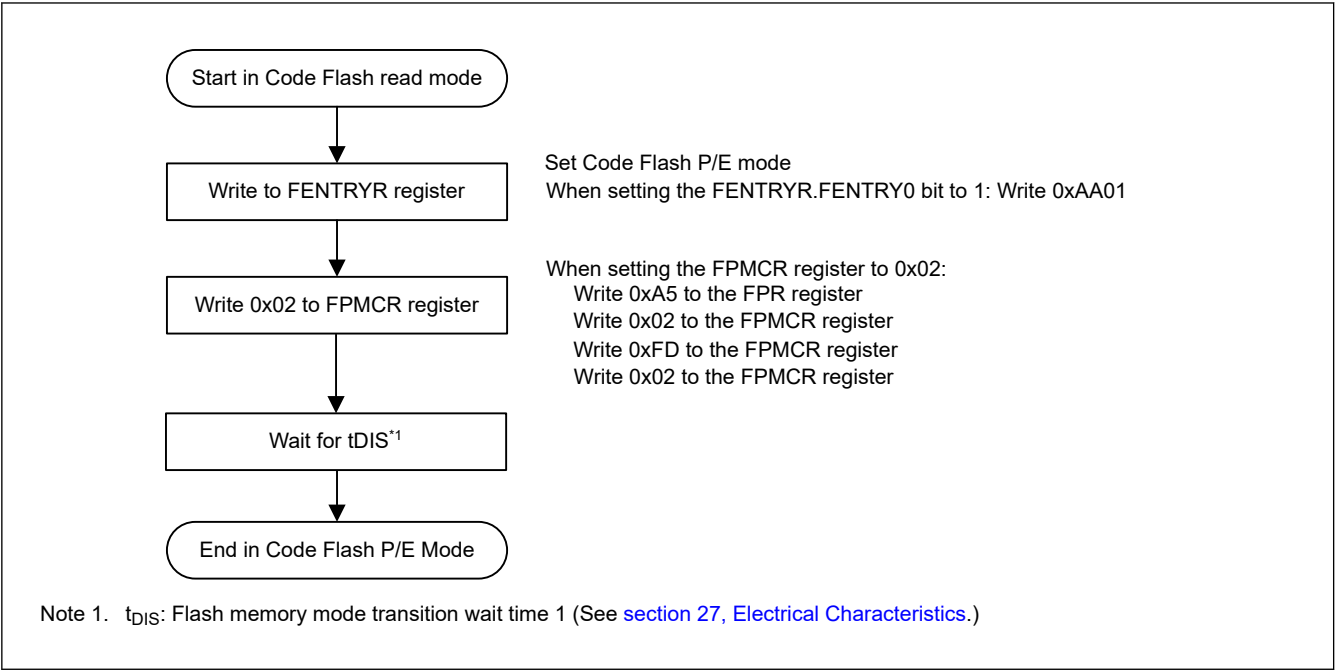


Figure 25.11 Procedure for changing from read mode to code flash P/E mode

(2) Switching the Code Flash P/E Mode to Read Mode

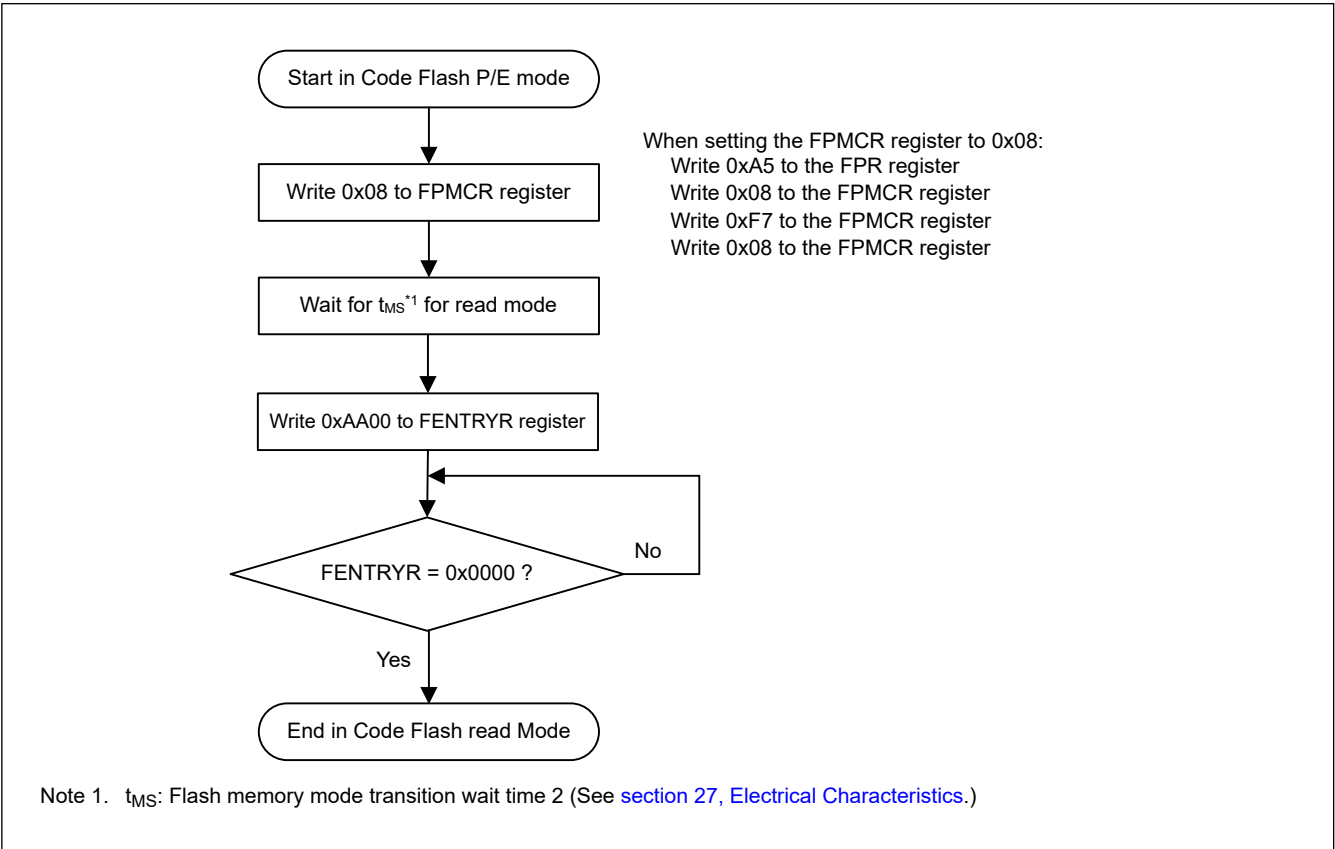


Figure 25.12 Procedure for changing from code flash P/E mode to read mode

(3) Flowchart for programming the code flash

The following figures describe the flow for programming the code flash.

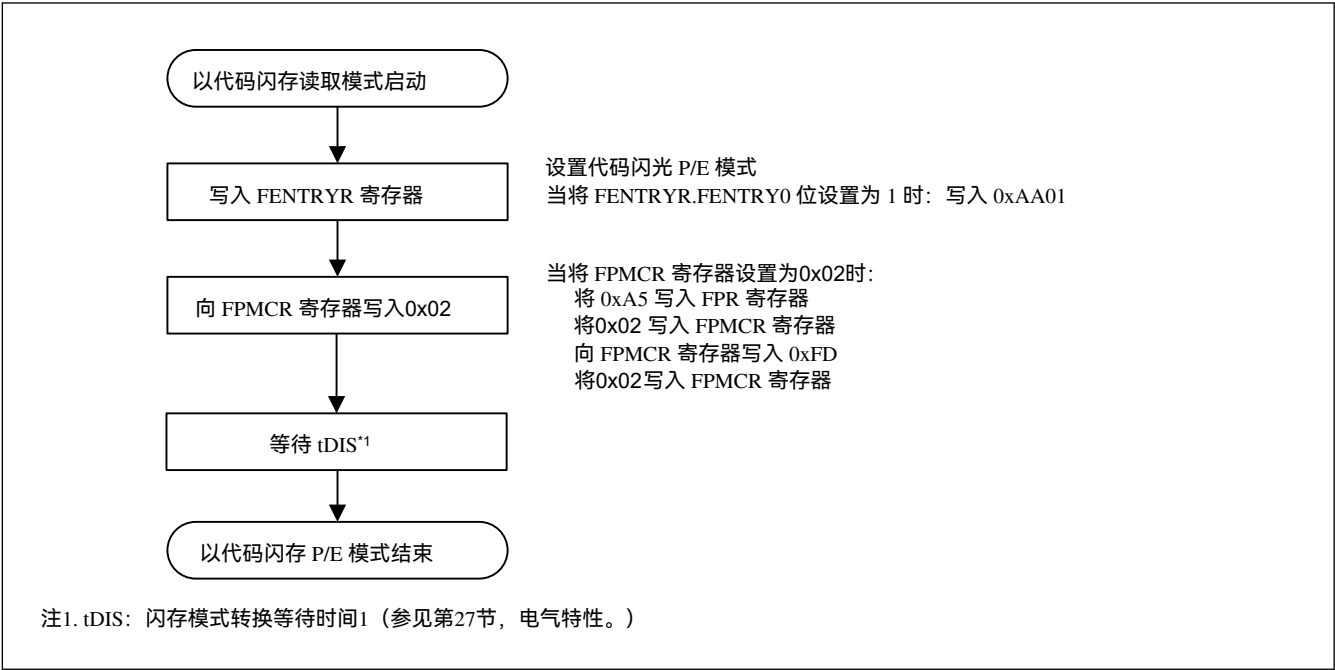


图 25.11 从读取模式切换到代码闪存 P/E 模式的步骤

(2) 将代码闪光灯P/E模式切换到读取模式

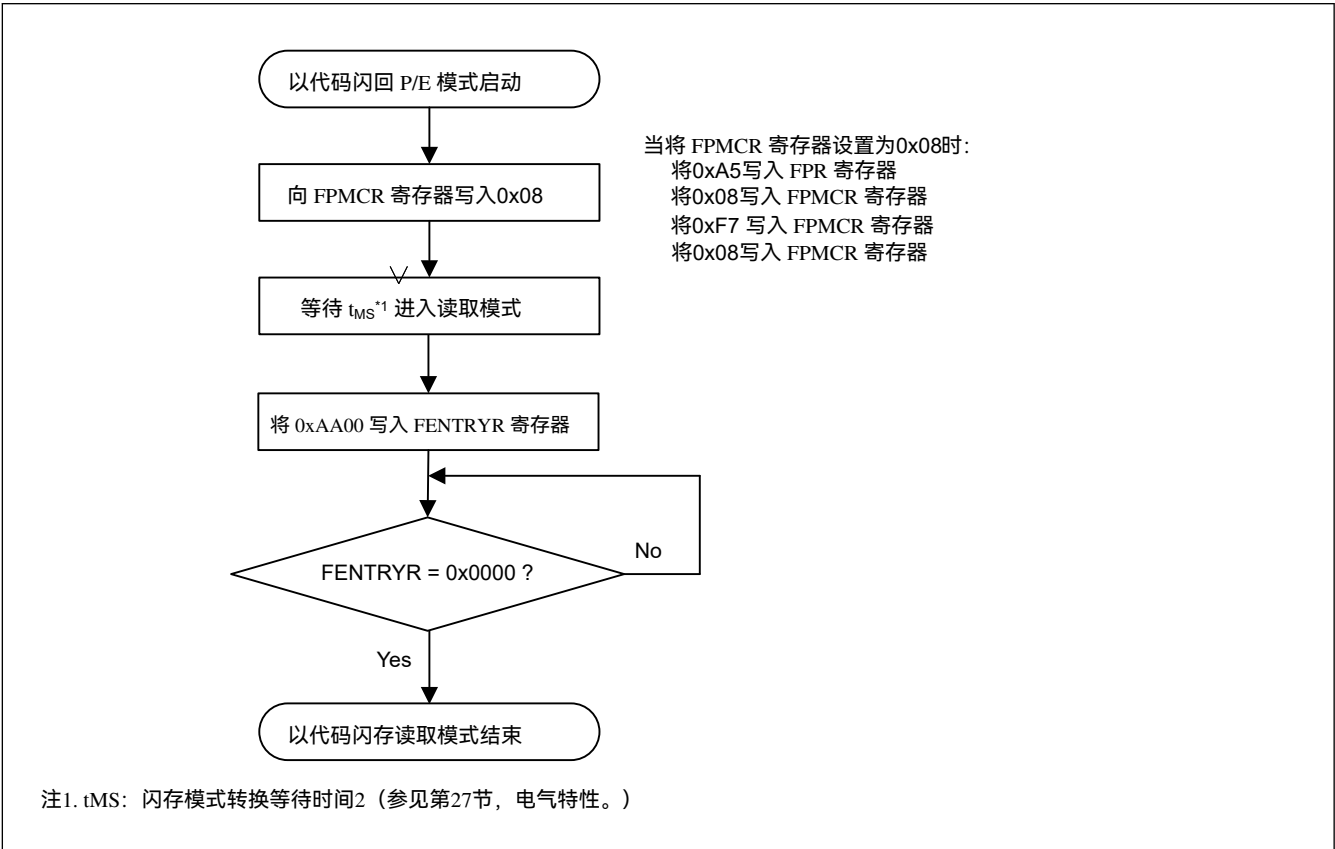


图 25.12 从代码闪存 P/E 模式切换到读取模式的步骤

(3) 代码闪存编程流程图

下图描述了代码闪存的编程流程。

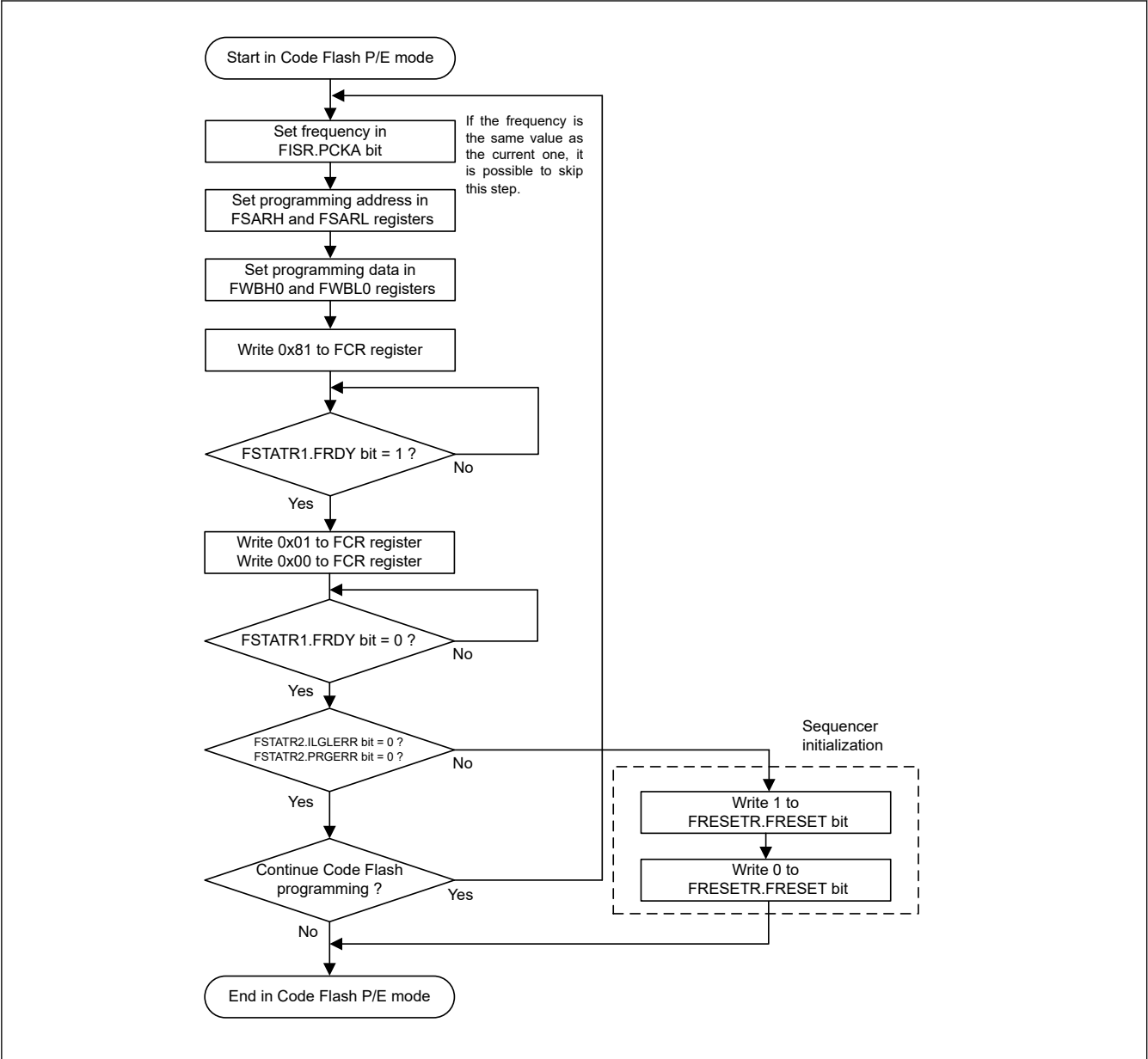


Figure 25.13 Flowchart for programming of the code flash

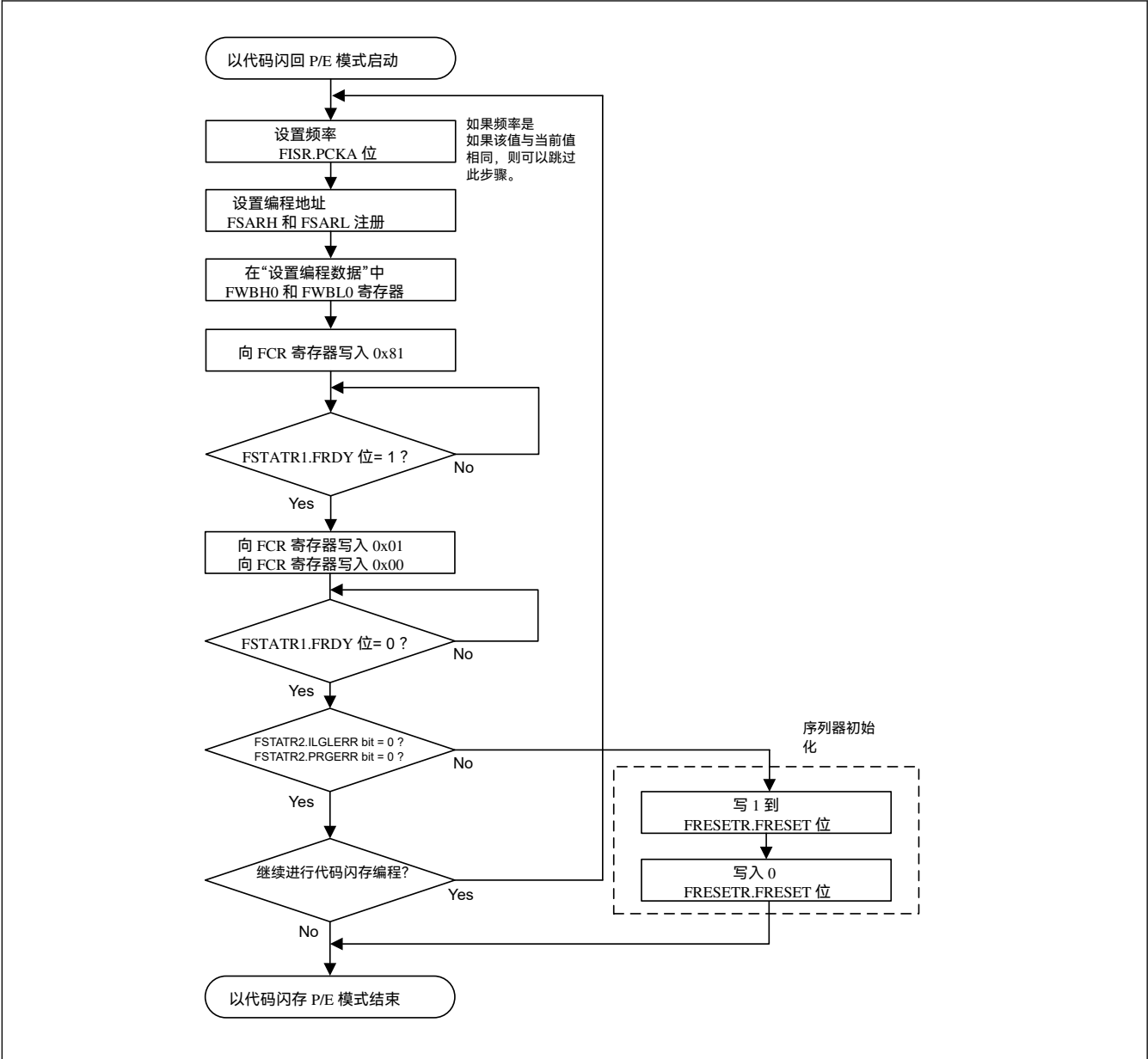


图 25.13 代码闪存编程流程图

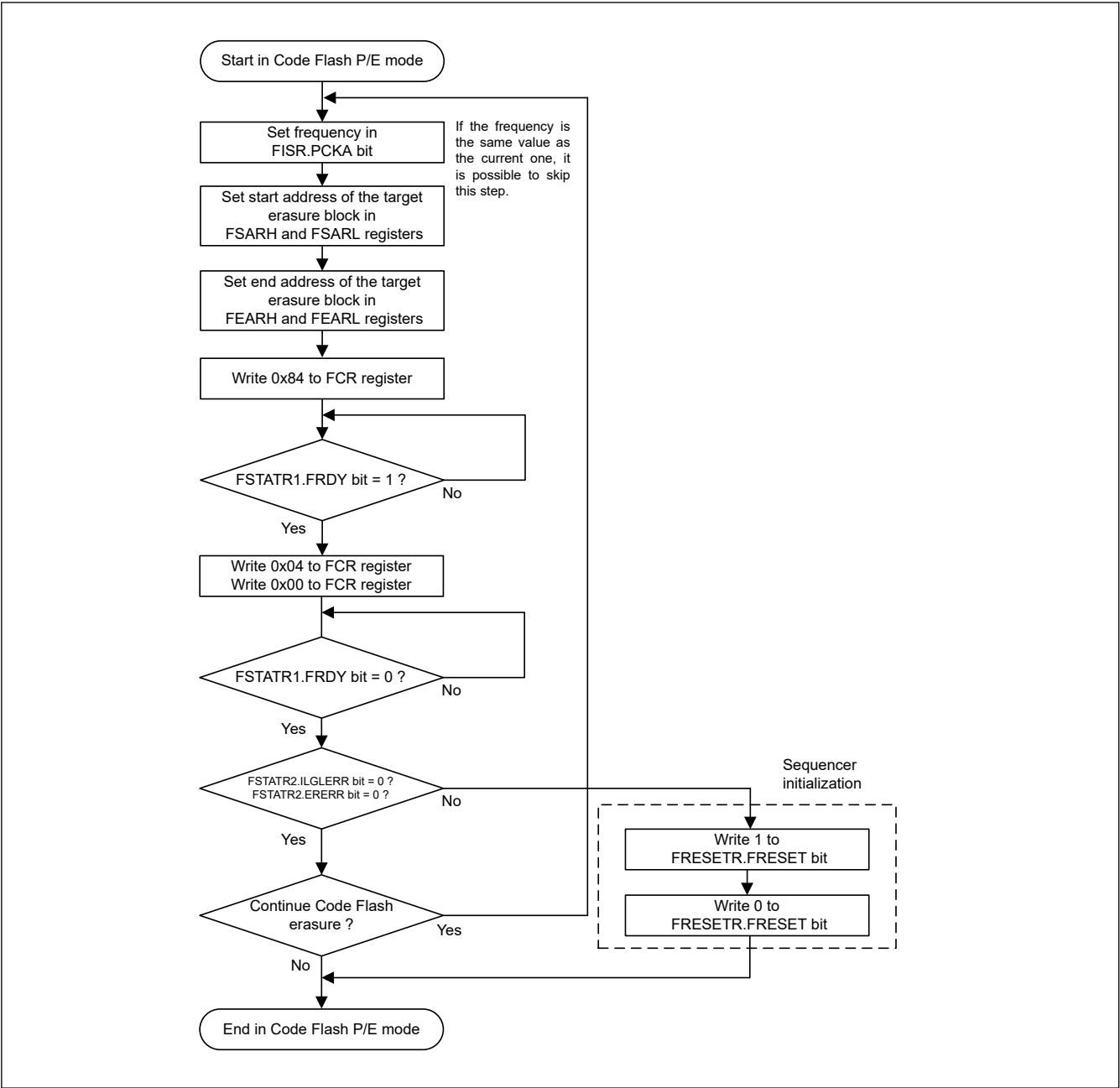


Figure 25.14 Flowchart for the code flash block erase procedure

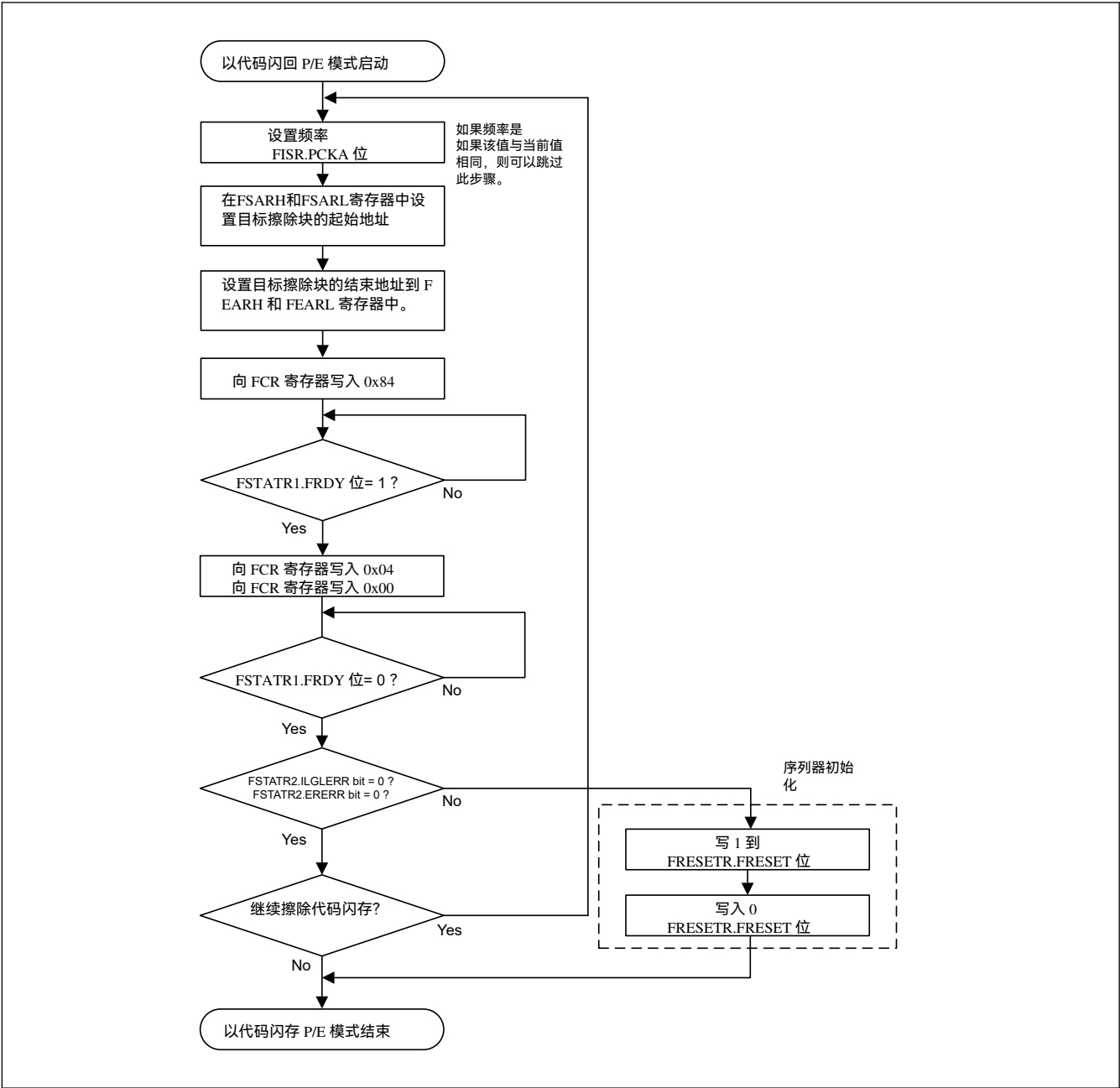


图 25.14 代码闪存块擦除流程图

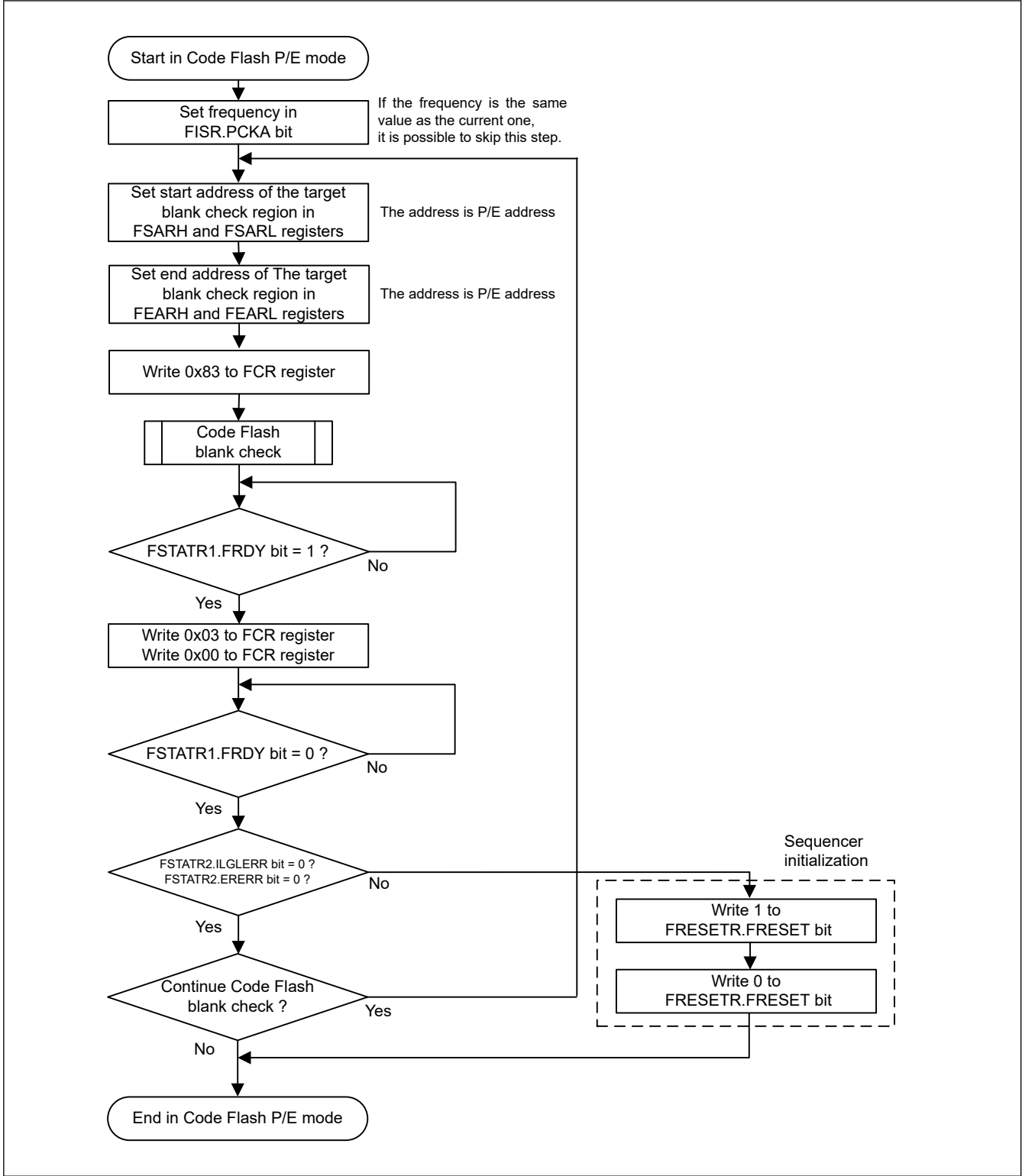


Figure 25.15 Flowchart for the code flash blank check procedure

(4) Startup Area Information and FSPR Program/Access Window Information Program/OCDID information Program

Figure 25.16 is a simple flowchart of the procedure for the startup area information and FSPR program/access window information program/OCDID information program.

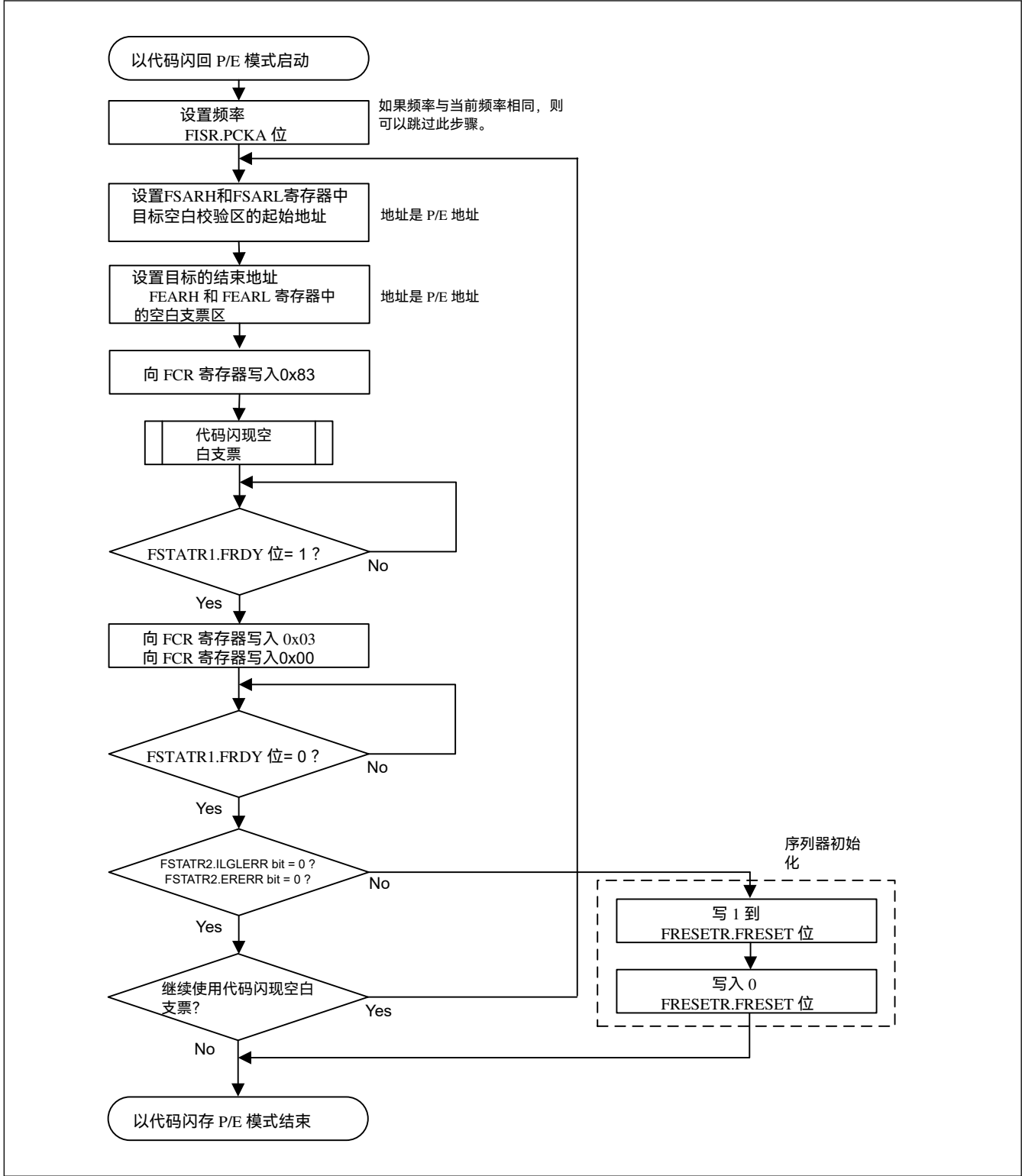


图 25.15 代码闪存空白检查流程图

(4) 启动区信息和FSPR计划/访问窗口信息计划/OCDID信息计划

图 25.16 是启动区域信息和 FSPR 程序/访问窗口信息程序/OCDID 信息程序的简单流程图。

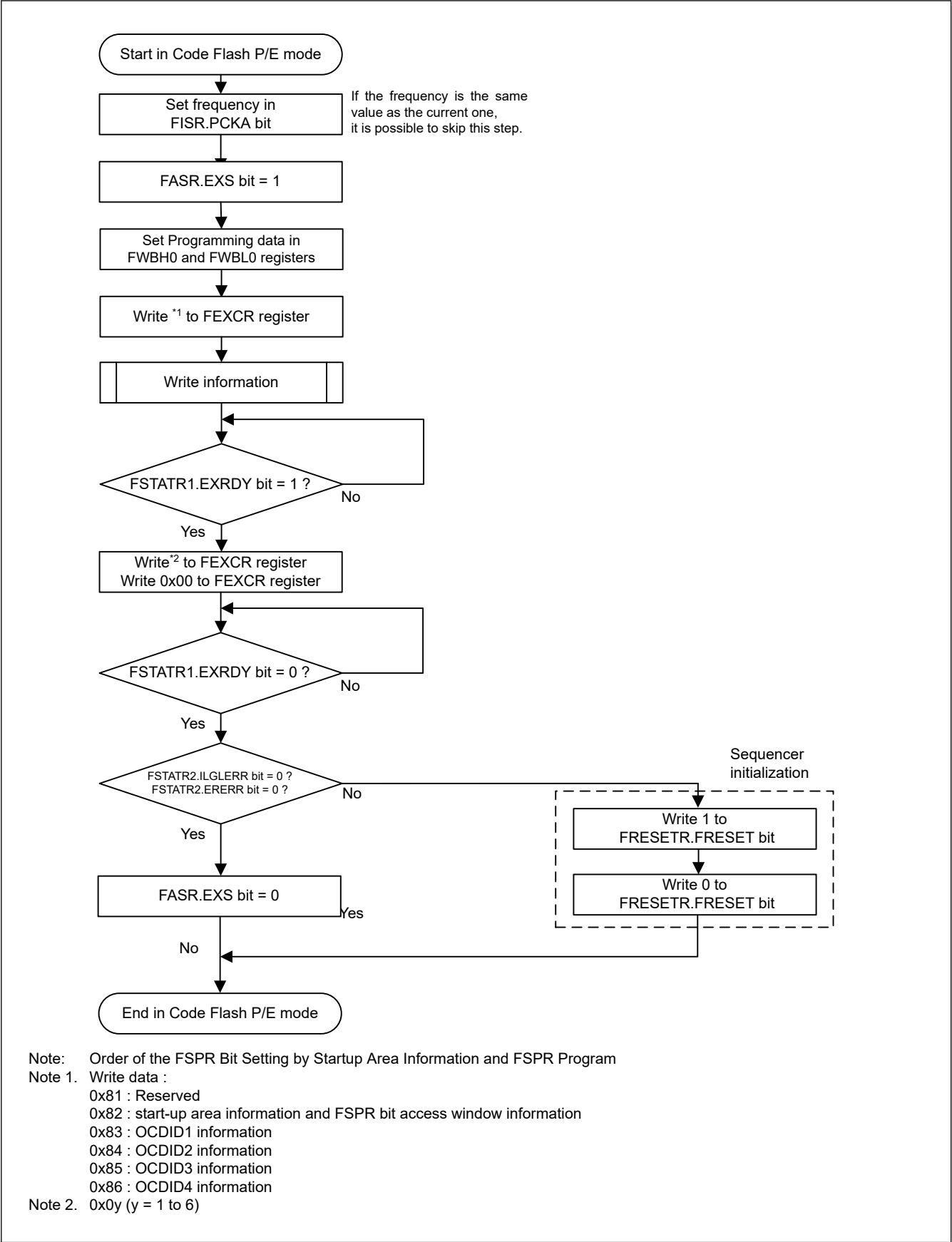


Figure 25.16 Simple flowchart for the procedure for Startup Area Information and FSPR Program/Access Window Information Program/OCDID information Program

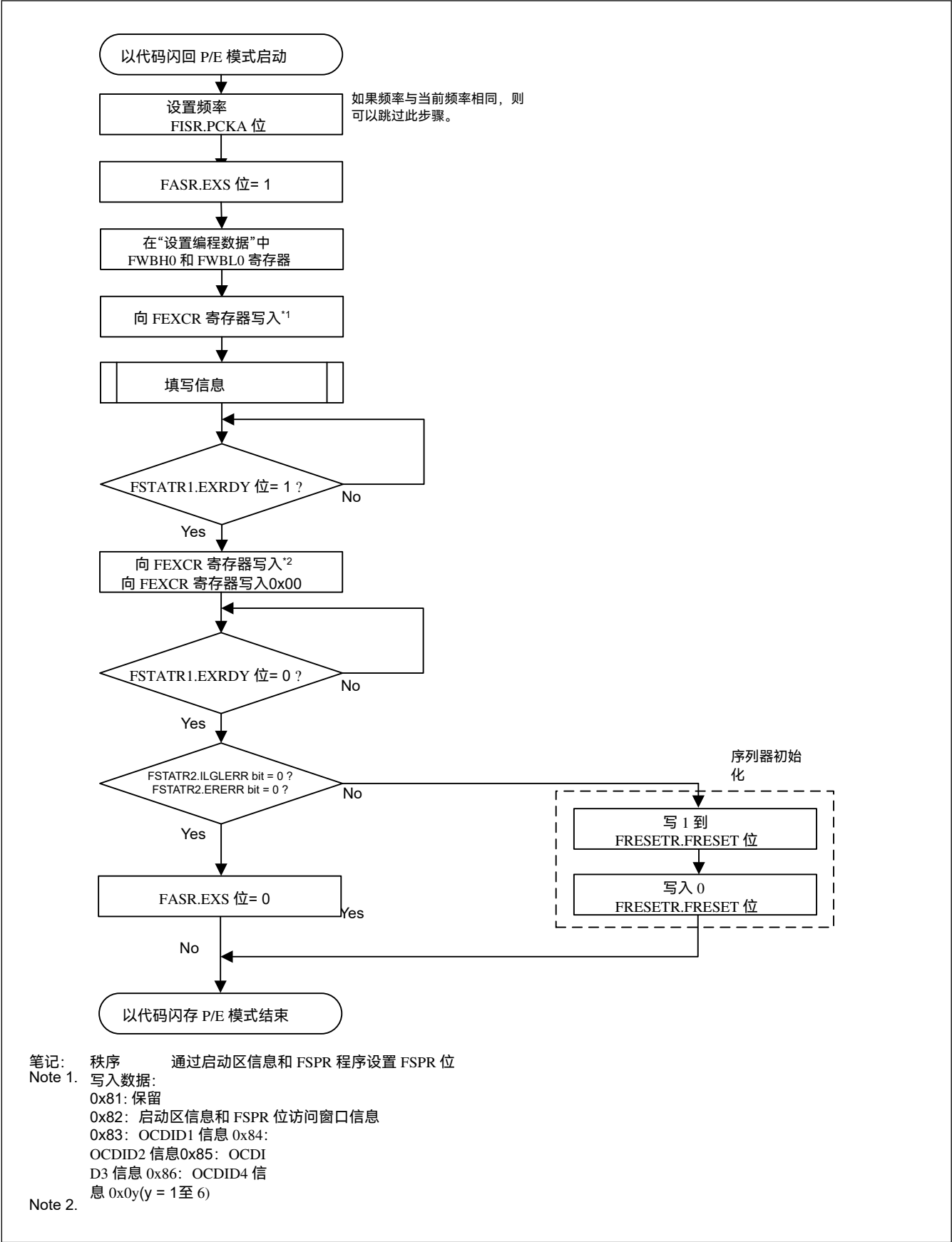


图 25.16 初创企业区域信息和 FSPR 计划/准入流程简图 窗口信息程序/强迫症信息程序

Set the FSPR bit after programing of the startup area information and the access window information. If the FSPR bit is set before programing of the startup area information and the access window information, the programming cannot be performed because of the security function in the FSPR. When programming using the hex file, programming in the ascending order of the address. In this case, the FSPR bit is written before the access window information. Therefore, divide the hex file for FSPR into another file, and use it after setting the access window information.

(5) Forced Stop by Software Command

Figure 25.17 shows a simple flowchart for the forced stop procedure to stop the blank check command or the block erase forcibly. When the forced stop command is executed, FEAMH/FEAML registers store the stopped address value. For the blank check command, the blank check can restart from the stopped address by copying the value of FEAMH/FEAML registers to FSARH/FSARL registers, respectively.

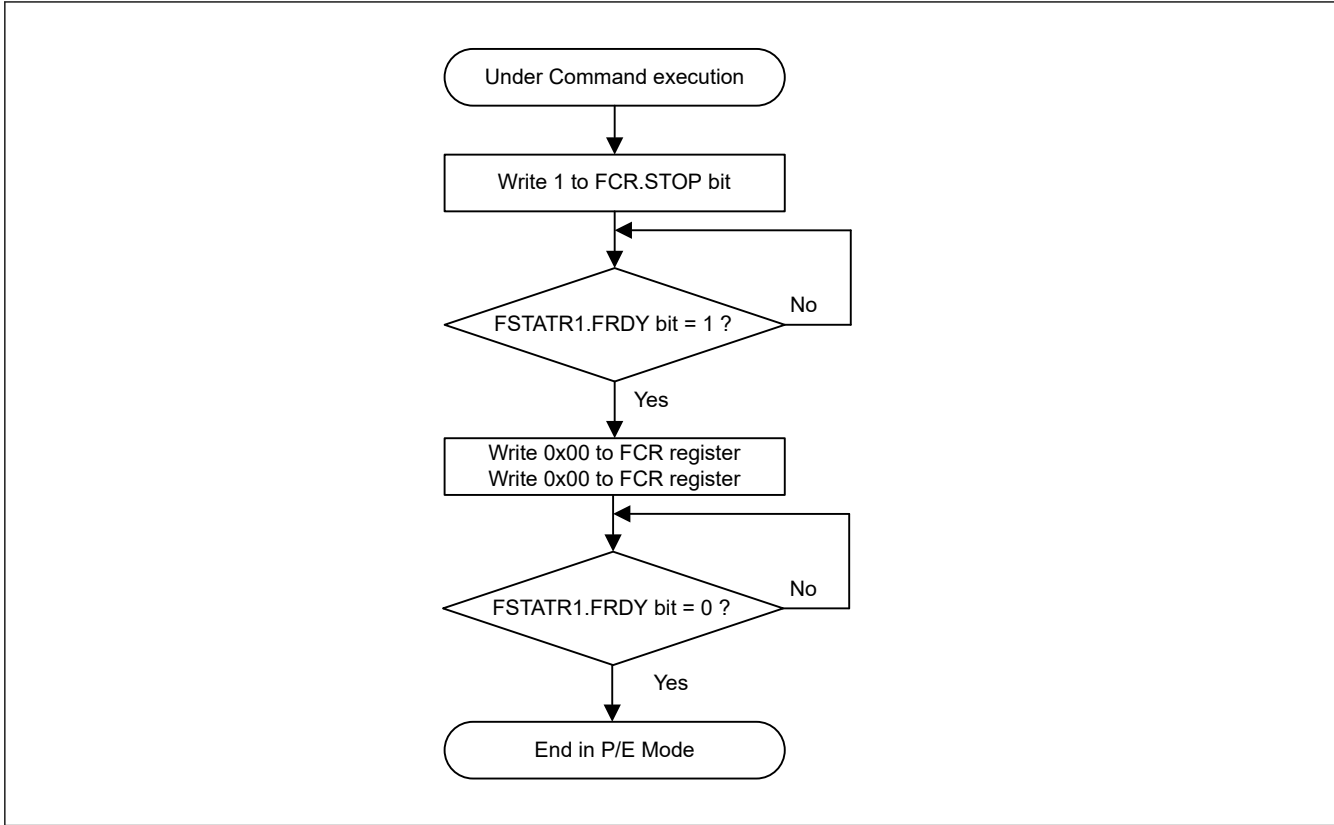


Figure 25.17 Simple flowchart for the forced stop procedure

25.11 Reading the Flash Memory

25.11.1 Reading the Code Flash Memory

No special settings are required to read the code flash memory in Normal mode. Data can be read by accessing the addresses in the code flash memory. When reading code flash memory that is erased but not yet reprogrammed, such as code flash memory in the non-programmed state, all bits are read as 1s.

25.12 Usage Notes

25.12.1 Erase Suspended Area

Data in areas where an erase operation is suspended is undefined. To avoid malfunctions caused by reading undefined data, do not execute commands and read data in the area where erase operation is suspended.

在写入启动区信息和访问窗口信息之后，设置 FSPR 位。如果在写入启动区信息和访问窗口信息之前设置 FSPR 位，由于 FSPR 的安全功能，将无法执行编程。使用十六进制文件进行编程时，应按地址升序进行编程。在这种情况下，FSPR 位会被写入访问窗口信息之前。因此，应将 FSPR 的十六进制文件拆分成另一个文件，并在设置访问窗口信息之后使用该文件。

(5) 通过软件命令强制停止

图 25.17 显示了强制停止过程的简单流程图，该过程用于强制停止空白校验命令或块擦除。执行强制停止命令时，FEAMH/FEAML 寄存器会存储停止地址的值。对于空白校验命令，可以通过将 FEAMH/FEAML 寄存器的值分别复制到 FSARH/FSARL 寄存器，从停止地址重新启动空白校验。

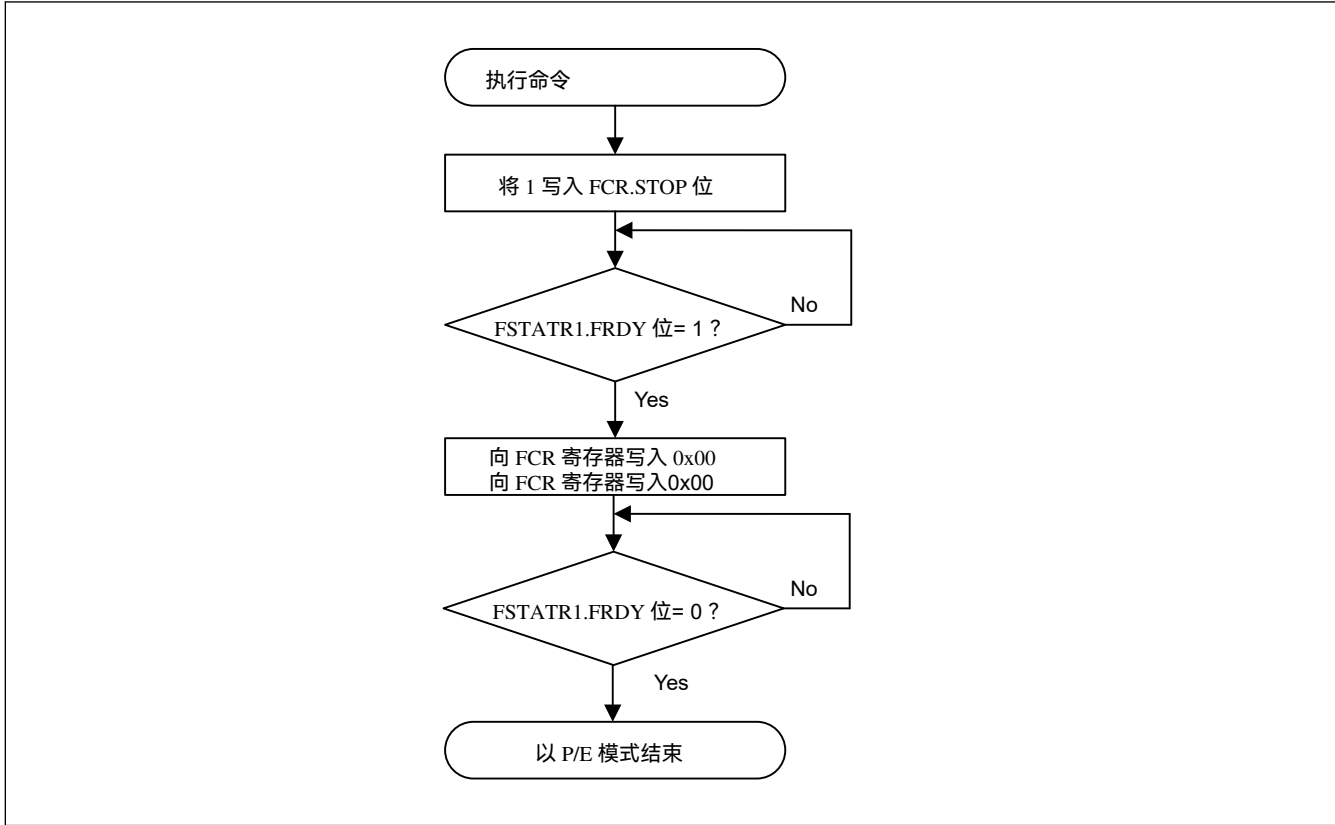


图 25.17 强制停止程序的简单流程图

25.11 读取闪存

25.11.1 读取代码闪存

在正常模式下读取代码闪存无需任何特殊设置。可通过访问代码闪存中的地址来读取数据。读取已擦除但尚未重新编程的代码闪存（例如处于未编程状态的代码闪存）时，所有位均读取为 1。

25.12 使用说明

25.12.1 擦除悬浮区域

擦除操作暂停期间，数据处于未定义状态。为避免因读取未定义数据而导致故障，请勿在擦除操作暂停期间执行命令或读取数据。

25.12.2 Constraints on Additional Writes

Other than the configuration area, no other area can be written to twice. After a write to a flash memory area is complete, erase the area before attempting to overwrite data in that area. The configuration area can be overwritten.

25.12.3 Reset during Programming and Erasure

If inputting a reset from the RES pin, release the reset after a reset input time of at least t_{RESW}. See [section 27.4.1. Reset Timing](#) within the range of the operating voltage defined in the electrical characteristics.

The IWDT reset and software reset do not require a t_{RESW} input time.

25.12.4 Non-Maskable Interrupt Disabled during Programming and Erasure

When a non-maskable interrupt*¹ occurs during a programming or erasure operation, the vectors are fetched from the code flash memory, and undefined data is read. Therefore, do not generate a non-maskable interrupt during programming and erasure operations in the code flash memory. This constraint applies only to the code flash memory.

Note 1. A non-maskable interrupt is an NMI pin interrupt, IWDT underflow or refresh error, and voltage monitor interrupt.

25.12.5 Location of Interrupt Vectors during Programming and Erasure

When an interrupt occurs during a programming and erasure operation, the vector can be fetched from the code flash memory as default setting. To avoid fetching the vector from the code flash memory, set the destination for fetching interrupt vectors to an area other than the code flash memory with the interrupt table.

25.12.6 Programming and Erasure in Subosc-speed Operating Mode

Do not program or erase the flash memory when subosc-speed operating mode is selected in the ICLKSCR register for low-power consumption functions.

25.12.7 Abnormal Termination during Programming and Erasure

When the voltage exceeds the range of the operating voltage during a programming and erasure operation, or when a programming or erasure operation did not complete successfully because of a reset or prohibited actions as described in [section 25.12.8. Actions Prohibited during Programming and Erasure](#), erase the area again.

25.12.8 Actions Prohibited during Programming and Erasure

To prevent damage to the flash memory, comply with the following instructions during programming and erasure:

- Do not use an MCU power supply that is outside the operating voltage range
- Do not update the FLMODE.MODE[1:0] bits value
- Do not update the ICLKSCR.CKSEL bit value
- Do not change the division ratio of the system clock (ICLK)
- Do not place the MCU in Software Standby mode

25.12.9 Flash-IF clock (ICLK) during Program/Erase

For programming/erasure by self-programming, it is necessary to specify an integer frequency by setting the Flash Initial Setting Register (FISR).

25.12.2 额外写入的限制

除了配置区域外，其他任何区域都不能被写入两次。写入闪存区域完成后，必须先擦除该区域，然后再尝试覆盖该区域的数据。配置区域可以被覆盖写入。

25.12.3 编程和擦除过程中的重置

如果从 RES 引脚输入复位信号，则在复位输入时间至少为 t_{RESW} 后释放复位信号。参见 27.4.1 节。复位时序应在电气特性中定义的工作电压范围内。

IWDT 复位和软件复位不需要 t_{RESW} 输入时间。

25.12.4 编程和擦除期间禁用不可屏蔽中断

当在编程或擦除操作期间发生不可屏蔽中断*1时，将从代码闪存中读取向量，并读取未定义的数据。因此，在代码闪存的编程和擦除操作期间，请勿生成不可屏蔽中断。此限制仅适用于代码闪存。

注 1. 不可屏蔽中断包括 NMI 引脚中断、IWDT 下溢或刷新错误以及电压监视器中断。

25.12.5 编程和擦除期间中断向量的位置

在编程和擦除操作期间发生中断时，默认情况下会从代码闪存中获取中断向量。为了避免从代码闪存中获取中断向量，请在中断表中将中断向量的获取目标设置为代码闪存以外的区域。

25.12.6 亚振荡器速度操作模式下的编程和擦除

当 ICLKSCR 寄存器中选择亚振荡器速度运行模式以实现低功耗功能时，请勿对闪存进行编程或擦除。

25.12.7 编程和擦除过程中的异常终止

当编程和擦除操作期间电压超过工作电压范围时，或者当由于复位或第 25.12.8 节所述的禁止操作而导致编程或擦除操作未成功完成时，请再次擦除该区域。

25.12.8 编程和擦除期间禁止的操作

为防止闪存损坏，请在编程和擦除过程中遵守以下说明：

- 请勿使用超出MCU工作电压范围的电源
- 不要更新 FLMODE.MODE[1:0] 位值
- 不要更新 ICLKSCR.CKSEL 位值
- 不要更改系统时钟 (ICLK) 的分频比。
- 请勿将MCU置于软件待机模式

25.12.9 编程/擦除期间的闪存-IF时钟 (ICLK)

对于通过自编程进行编程/擦除，需要通过设置闪存初始设置寄存器 (FISR) 来指定整数频率。

26. Internal Voltage Regulator

26.1 Overview

The MCU includes one internal voltage regulator:

- Linear regulator (LDO)

This regulator supplies voltage to all internal circuits and memory except for I/O and analog domains.

26.2 Operation

Table 26.1 lists the LDO pin settings, and Figure 26.1 shows the LDO settings.

Table 26.1 LDO pin

Pins	Setting descriptions
VCC	• Connect VCC to the system power supply. • Connect VCC to VSS through a 0.1 μF multilayer ceramic capacitor. Place the capacitor close to the pin.
VCL	Connect the pin to VSS through a 0.47 μF to 1 μF multilayer ceramic capacitor. Place the capacitor close to the pin.

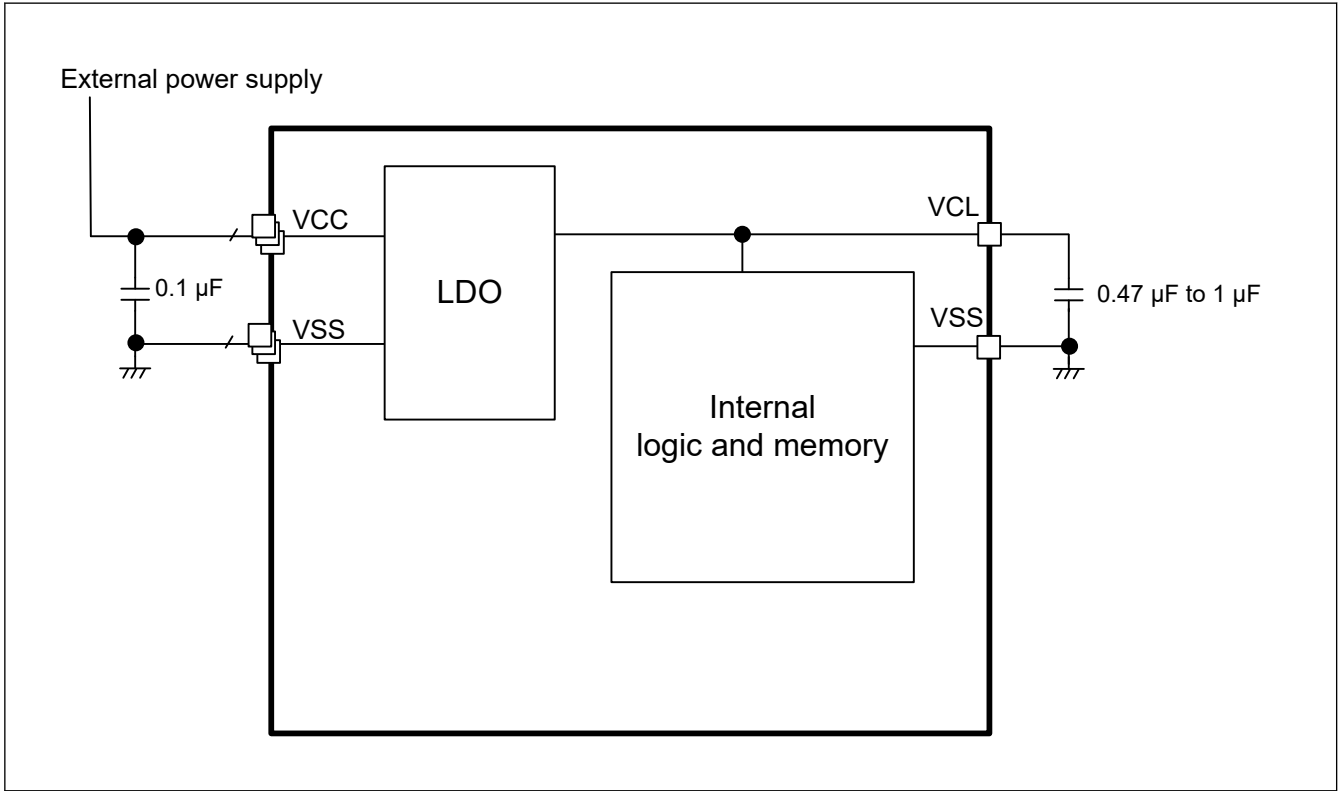


Figure 26.1 LDO settings

26.内部电压调节器

26.1 概述

该微控制器包含一个内部电压调节器：

- 线性稳压器（LDO）

该稳压器为除I/O和模拟域之外的所有内部电路和存储器提供电压。

26.2 操作

表 26.1 列出了 LDO 引脚设置，图 26.1 显示了 LDO 设置。

表 26.1 LDO 引脚

别针	设置描述
VCC	● 将VCC连接到系统电源。 ● 通过多层陶瓷电容器将VCC连接到VSS。将电容器靠近引脚放置。
VCL	通过一个从0.47 μF 到1 μF 的多层陶瓷电容器将引脚连接到 VSS。将电容器靠近引脚放置。

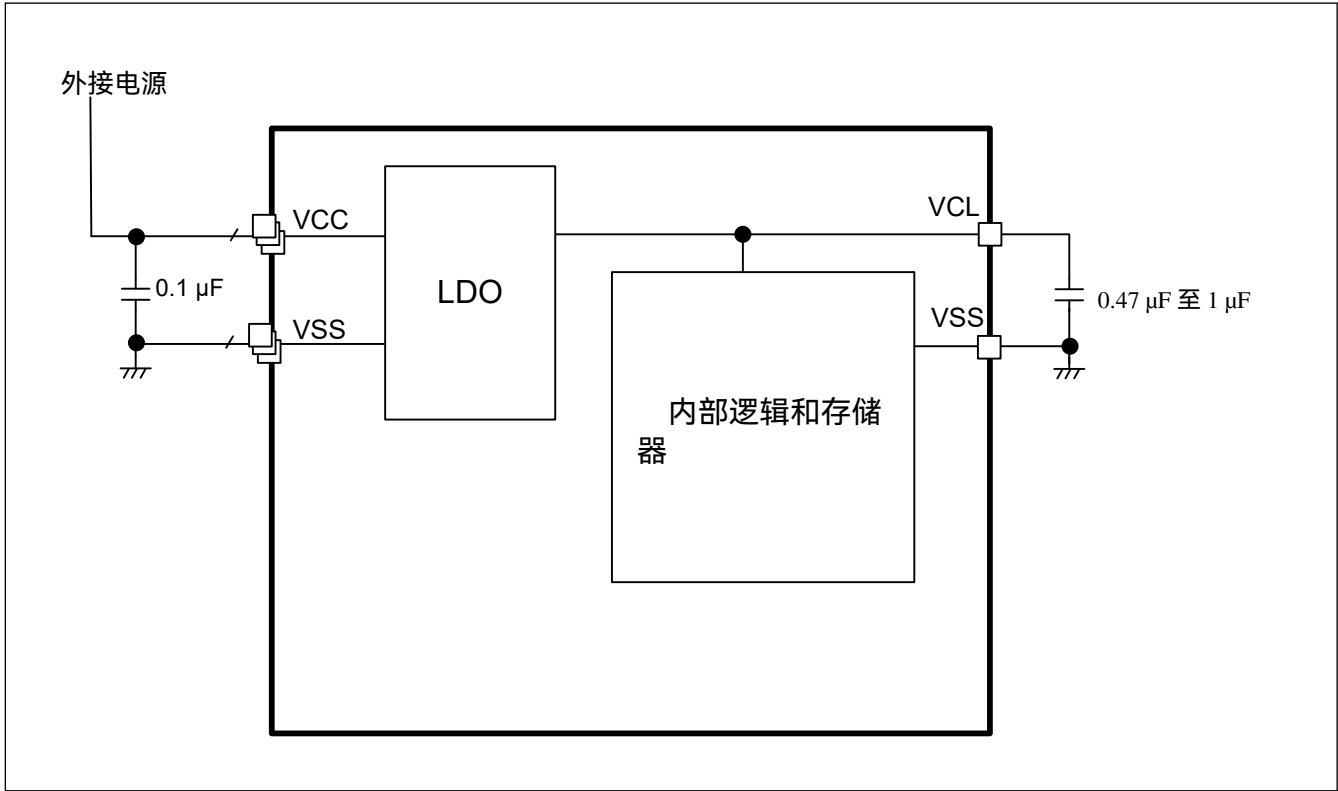


图 26.1 LDO 设置

27. Electrical Characteristics

Unless otherwise specified, the electrical characteristics of the MCU are defined under the following conditions:

$VCC^{*1} = VREFH0 = 1.6 \text{ to } 5.5 \text{ V}$

$VSS = VREFL0 = 0 \text{ V}, T_a = T_{opr}$

Note 1. The typical condition is set to $VCC = 3.3 \text{ V}$.

Figure 27.1 shows the timing conditions.

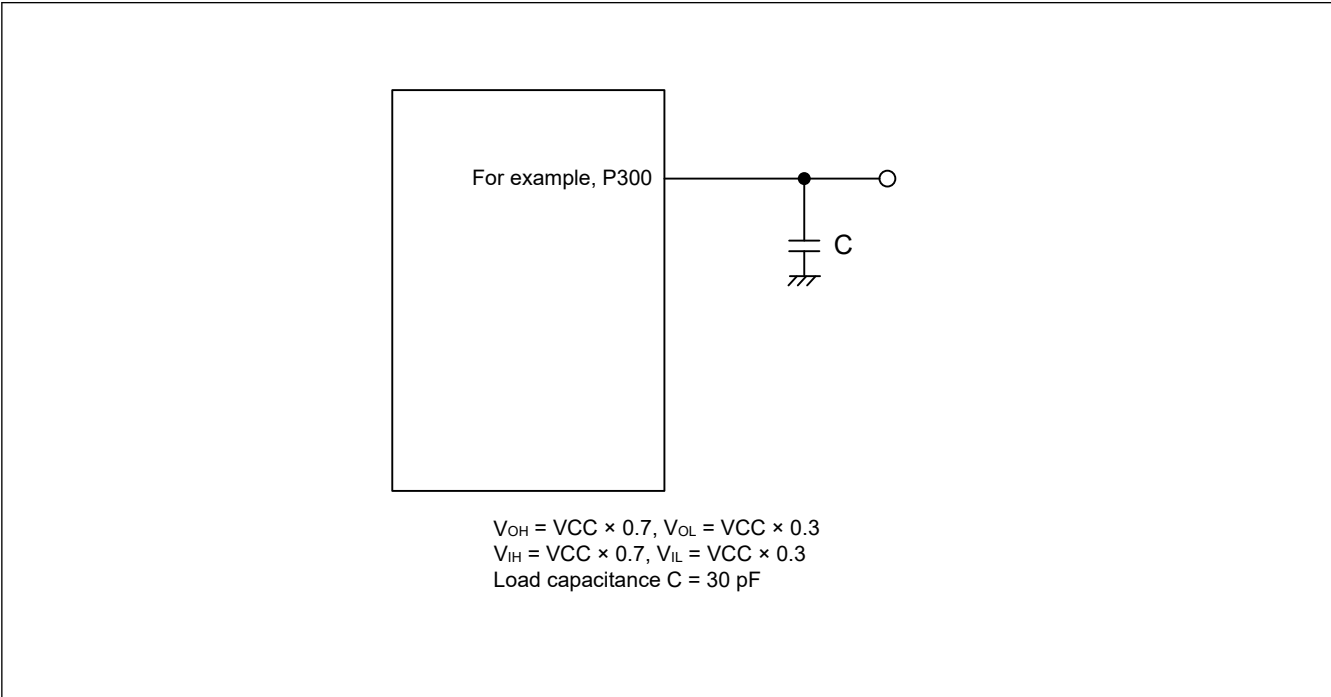


Figure 27.1 Input or output timing measurement conditions

27.1 Absolute Maximum Ratings

Table 27.1 Absolute maximum ratings (1 of 2)

Parameter		Symbol	Value	Unit
Power supply voltage		VCC	-0.5 to +6.5	V
VCL pin input voltage		V _{IVCL}	-0.3 to +2.1 and -0.3 to VCC + 0.3 ^{*1}	V
Input voltage	P100 to P102, P108 to P110, P112, P200, P201, P206, P300	V _{I1}	-0.3 to VCC + 0.3 ^{*2}	V
	P010 to P013, P212, P213	V _{I2}	-0.3 to VCC + 0.3 ^{*2}	V
Output voltage	P100 to P102, P108 to P110, P112, P201, P206, P300	V _{O1}	-0.3 to VCC + 0.3 ^{*2}	V
	P010 to P013, P212, P213	V _{O2}	-0.3 to VCC + 0.3 ^{*2}	V
Analog input voltage	AN000 to AN001, AN004 to AN005	V _{AI1}	-0.3 to VCC + 0.3 and -0.3 to VREFH0 + 0.3 ^{*2 *3}	V
	AN021 to AN022	V _{AI2}	-0.3 to VCC + 0.3 and -0.3 to VREFH0 + 0.3 ^{*2 *3}	V

27.电气特性

除非另有规定，MCU的电气特性在以下条件下定义：

$VCC^{*1} = VREFH0 = 1.6 \text{ to } 5.5 \text{ V}$

$VSS = VREFL0 = 0 \text{ V}, T_a = T_{opr}$

注1. 典型条件设置为 $VCC = 3.3 \text{ V}$ 。

图 27.1 显示了时序条件。

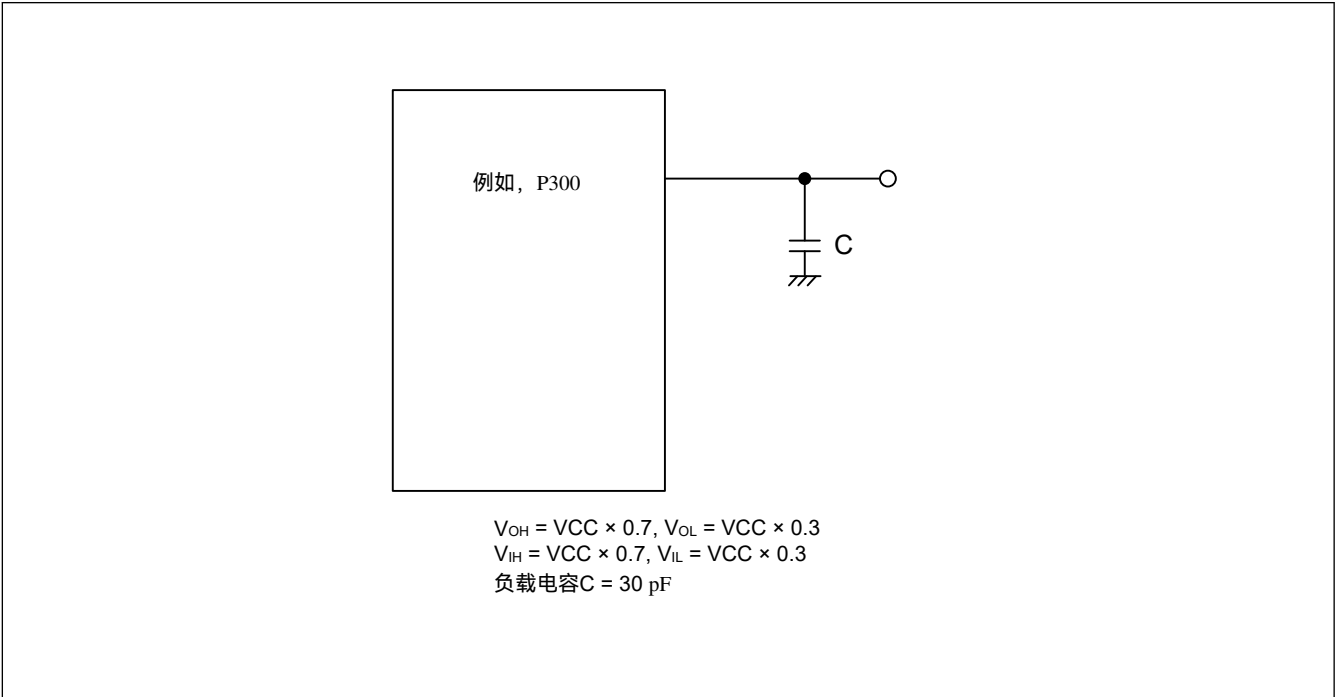


图 27.1 输入或输出时序测量条件

27.1 绝对最大额定值

表 27.1 绝对最大评级 (1/2)

范围		象征	价值	单元
电源电压		VCC	-0.5 to +6.5	V
VCL引脚输入电压		V _{IVCL}	-0.3 至+2.1和 -0.3 至VCC + 0.3 ^{*1}	V
输入电压	P100 至 P102、P108 至 P110、P112、P200、P201、P206、P300	V _{I1}	-0.3 to VCC + 0.3 ^{*2}	V
	P010 至 P013、P212、P213	V _{I2}	-0.3 to VCC + 0.3 ^{*2}	V
输出电压	P100 至 P102、P108 至 P110、P112、P201、P206、P300	V _{O1}	-0.3 to VCC + 0.3 ^{*2}	V
	P010 至 P013、P212、P213	V _{O2}	-0.3 to VCC + 0.3 ^{*2}	V
模拟输入电压	AN000 至 AN001, AN004 至 AN005	V _{AI1}	-0.3 至VCC + 0.3和 -0.3 至 VR EFH0 + 0.3 ^{*2 *3}	V
	AN021 to AN022	V _{AI2}	-0.3 至VCC + 0.3和 -0.3 至 VR EFH0 + 0.3 ^{*2 *3}	V

Table 27.1 Absolute maximum ratings (2 of 2)

Parameter			Symbol	Value	Unit
High-level output current	P100 to P102, P108 to P110, P112, P201, P206, P300	Per pin	I _{OH1}	-40	mA
		Total of all pins		-100	mA
	P010 to P013, P212, P213	Per pin	I _{OH2}	-5	mA
		Total of all pins		-20	mA
Low-level output current	P100 to P102, P108 to P110, P112, P201, P206, P300	Per pin	I _{OL1}	40	mA
		Total of all pins		100	mA
	P010 to P013, P212, P213	Per pin	I _{OL2}	10	mA
		Total of all pins		20	mA
Ambient operating temperature	In normal operation mode		T _a	-40 to +125	°C
	In flash memory programming mode			-40 to +125	°C
Storage temperature			T _{stg}	-65 to +150	°C

Note 1. Connect the VCL pin to VSS via a capacitor (0.47 to 1 μF). The listed value is the absolute maximum rating of the VCL pins. Only use the capacitor connection. Do not apply a specific voltage to this pin.

Note 2. This voltage must be no higher than 6.5 V.

Note 3. The voltage on a pin in use for A/D conversion must not exceed VREFH0 + 0.3.

Note: The characteristics of functions multiplexed on a given pin are the same as those for the port pin unless otherwise specified.

Note: VREFH0 refers to the positive reference voltage of the A/D converter.

Note: The reference voltage is VSS.

Caution: Product quality may suffer if the absolute maximum rating is exceeded even momentarily for any parameter. That is, the absolute maximum ratings are rated values at which the product is on the verge of suffering physical damage, and therefore the product must be used under conditions that ensure that the absolute maximum ratings are not exceeded.

Table 27.2 Recommended operating conditions

Parameter	Symbol		Min.	Typ.	Max.	Unit
Power supply voltages	VCC		1.6	—	5.5	V
	VSS		—	0	—	V
Analog power supply voltages	VREFH0	When used as ADC10 Reference	1.6	—	VCC	V
	VREFL0		—	0	—	V

27.1.1 Tj/Ta Definition

Table 27.3 Tj/Ta definition

Conditions: Products with operating temperature Ta = -40 to +125°C

Parameter	Symbol	Typ.	Max.	Unit	Test conditions
Permissible junction temperature	Tj	—	140*1	°C	High-speed mode Middle-speed mode Low-speed mode Subosc-speed mode

Note 1. The upper limit of operating temperature is 125°C.

Note: Make sure that $T_j = T_a + \theta_{ja} \times \text{total power consumption (W)}$, where total power consumption = $(V_{CC} - V_{OH}) \times \Sigma I_{OH} + V_{OL} \times \Sigma I_{OL} + I_{CCmax} \times V_{CC}$.

表 27.1 绝对最大评级 (2/2)

范围			象征	价值	单元
高输出电流	P100 至 P102, P108 至 P110、P112、P201、P206、P300	别针	I _{OH1}	-40	mA
		所有引脚总数		-100	mA
	P010 至 P013、P212、P213	别针	I _{OH2}	-5	mA
		所有引脚总数		-20	mA
低电平输出电流	P100 至 P102, P108 至 P110、P112、P201、P206、P300	别针	I _{OL1}	40	mA
		所有引脚总数		100	mA
	P010 至 P013、P212、P213	别针	I _{OL2}	10	mA
		所有引脚总数		20	mA
环境工作温度	正常运行模式		Ta	-40 to +125	°C
	闪存编程模式			-40 to +125	°C
储存温度			测试	-65 to +150	°C

注1: 使用电容（0.47至1 μF）将VCL引脚连接至VSS。所列数值为VCL引脚的绝对最大额定值。仅使用电容连接，切勿向该引脚施加任何电压。

注2. 此电压不得高于6.5 V.

注 3. 用于 A/D 转换的引脚上的电压不得超过 VREFH0+ 0.3。

注意：除非另有规定，否则在给定引脚上复用的功能的特性与端口引脚的特性相同。

注：VREFH0 指的是 A/D 转换器的正参考电压。

注：参考电压为VSS。

注意：即使任何参数的绝对最大额定值被瞬间超过，产品质量也可能受到影响。也就是说，绝对最大额定值是指产品即将遭受物理损坏时的额定值，因此必须在确保不超过绝对最大额定值的条件下使用产品。

表 27.2 推荐运行条件

范围	象征		最小。	类型。	最大限	单元
电源电压	VCC		1.6	—	5.5	V
	VSS		—	0	—	V
模拟电源电压	VREFH0	当用作 ADC10 时参考	1.6	—	VCC	V
	VREFL0		—	0	—	V

27.1.1Tj/Ta 定义

表 27.3Tj/Ta 定义

条件: 工作温度在Ta = -40至+125°C范围内的产品

范围		象征	类型。	最大限度。	单元	测试条件
允许的结温		Tj	—	140*1	°C	高速模式 中速模式 低速模式 子振荡器速度模式

注1. 工作温度上限为125°C。

注意：确保 $T_j = T_a + \theta_{ja} \times \text{总功耗(W)}$ ，其中总功耗= $(V_{CC} - V_{OH}) \times \Sigma I_{OH} + V_{OL} \times \Sigma I_{OL} + I_{CCmax} \times V_{CC}$ 。

27.2 Oscillators Characteristics

27.2.1 On-chip Oscillators Characteristics

Table 27.4 On-chip oscillators characteristics

Conditions: VCC = 1.6 to 5.5 V, VSS = 0 V, Ta = -40 to +125°C

Parameter		Symbol	Min.	Typ.	Max.	Unit	Test conditions
High-speed on-chip oscillator clock frequency		f _{HOCO}	1	—	32	MHz	—
High-speed on-chip oscillator clock frequency accuracy	OSCSF.HOCOSF = 1	—	-1.0	—	+1.0	%	Ta = -40 to +125°C, 1.6 V ≤ VCC ≤ 5.5 V
High-speed on-chip oscillator clock frequency trimming resolution		—	—	0.05	—	%	—
High-speed on-chip oscillator clock oscillation stabilization time*3		t _{HOCO}	—	—	4.4	μs	—
Low-speed on-chip oscillator clock frequency*1		f _{LOCO}	—	32.768	—	kHz	—
Low-speed on-chip oscillator clock frequency accuracy		—	-15	—	15	%	—
Low-speed on-chip oscillator clock frequency trimming resolution		—	—	0.3	—	%	—
Low-speed on-chip oscillator clock oscillation stabilization time		t _{LOCO}	—	—	100	μs	—
Low-speed on-chip oscillator frequency temperature coefficient		—	—	—	±0.21*2	%/°C	—

Note 1. The listed values only indicate the characteristics of the oscillators. Refer to AC Characteristics for instruction execution time.

Note 2. These values are the results of characteristic evaluation and are not checked for shipment.

Note 3. Check OSCSF.HOCOSF to confirm whether stabilization time has elapsed.

27.3 DC Characteristics

27.3.1 Pin Characteristics

Table 27.5 I/O I_{OH}

Conditions: VCC = 1.6 to 5.5 V, VSS = 0 V, Ta = -40 to +125°C

Parameter		Symbol	Min.	Typ.	Max.	Unit	Test conditions
Allowable high-level output current*1	Per pin for P100 to P102, P108 to P110, P112, P201, P206, P300	I _{OH1}	—	—	-10*2	mA	1.6 V ≤ VCC ≤ 5.5 V
	Total of all pins (when duty ≤ 70%*3)		—	—	-80*4	mA	4.0 V ≤ VCC ≤ 5.5 V
			—	—	-19	mA	2.7 V ≤ VCC < 4.0 V
			—	—	-10	mA	1.8 V ≤ VCC < 2.7 V
			—	—	-5	mA	1.6 V ≤ VCC < 1.8 V
	Per pin for P010 to P013, P212, P213	I _{OH2}	—	—	-3*2	mA	4.0 V ≤ VCC ≤ 5.5 V
			—	—	-1*2	mA	2.7 V ≤ VCC < 4.0 V
			—	—	-1*2	mA	1.8 V ≤ VCC < 2.7 V
			—	—	-0.5*2	mA	1.6 V ≤ VCC < 1.8 V
	Total of all pins (when duty ≤ 70%*3)		—	—	-20	mA	4.0 V ≤ VCC ≤ 5.5 V
			—	—	-10	mA	2.7 V ≤ VCC < 4.0 V
			—	—	-5	mA	1.8 V ≤ VCC < 2.7 V
			—	—	-5	mA	1.6 V ≤ VCC < 1.8 V

Note 1. Device operation is guaranteed at the listed currents even if current is flowing from the VCC pin to an output pin.

27.2 振荡器特性

27.2.1 片上振荡器特性

表 27.4 片上振荡器特性

条件: VCC = 1.6至5.5 V, VSS = 0 V, Ta = -40 至+125°C

范围		象征	最小。	类型。	最大限度。	单元	测试条件
高速片上振荡器时钟频率		f _{HOCO}	1	—	32	MHz	—
高速片上振荡器时钟频率精度	OSCSF.HOCOSF = 1	—	-1.0	—	+1.0	%	Ta = -40至+125°C, 1.6 V ≤ VCC ≤ 5.5 V
高速片上振荡器时钟频率微调分辨率		—	—	0.05	—	%	—
高速片上振荡器时钟振荡稳定时间*3		t _{HOCO}	—	—	4.4	μs	—
低速片上振荡器时钟频率*1		薄片	—	32.768	—	kHz	—
低速片上振荡器时钟频率精度		—	-15	—	15	%	—
低速片上振荡器时钟频率微调分辨率		—	—	0.3	—	%	—
低速片上振荡器时钟振荡稳定时间		疯狂	—	—	100	μs	—
低速片上振荡器频率温度系数		—	—	—	±0.21*2	%/°C	—

注1: 所列数值仅表示振荡器的特性。有关指令执行时间, 请参阅交流特性。

注2. 这些数值是特性评估的结果, 并非出货时的检查值。

注3. 检查 OSCSF.HOCOSF 以确认稳定时间是否已过。

27.3直流特性

27.3.1引脚特性

表 27.5 I/O I_{OH}

条件: VCC = 1.6至5.5 V, VSS = 0 V, Ta= -40 至+125°C

范围		象征	最小。	类型。	最大限度。	单元	测试条件
允许的高电平输出 电流*1	P100 至 P102 的每个引脚, P108 至 P110、P112、 P201, P206, P300	I _{OH1}	—	—	-10* ²	mA	1.6 V ≤ VCC ≤ 5.5 V
	所有引脚总数 (当 占空比为≤ 70%* ³)		—	—	-80* ⁴	mA	4.0 V ≤ VCC ≤ 5.5 V
			—	—	-19	mA	2.7 V ≤ VCC < 4.0 V
			—	—	-10	mA	1.8 V ≤ VCC < 2.7 V
			—	—	-5	mA	1.6 V ≤ VCC < 1.8 V
	P010 至 P013 的每个引脚, P212, P213	I _{OH2}	—	—	-3* ²	mA	4.0 V ≤ VCC ≤ 5.5 V
			—	—	-1* ²	mA	2.7 V ≤ VCC < 4.0 V
			—	—	-1* ²	mA	1.8 V ≤ VCC < 2.7 V
			—	—	-0.5* ²	mA	1.6 V ≤ VCC < 1.8 V
	所有引脚总数 (当职责≤ 70%* ³)		—	—	-20	mA	4.0 V ≤ VCC ≤ 5.5 V
			—	—	-10	mA	2.7 V ≤ VCC < 4.0 V
			—	—	-5	mA	1.8 V ≤ VCC < 2.7 V
			—	—	-5	mA	1.6 V ≤ VCC < 1.8 V

注1. 即使电流从 VCC 引脚流向输出引脚, 也能保证器件在所列电流下正常工作。

Note 2. The combination of these and other pins must also not exceed the value for maximum total current.

Note 3. The listed currents apply when the duty cycle is no greater than 70%. Use the following formula to calculate the output current when the duty cycle is greater than 70%, where n is the duty cycle.

- Total output current from the listed pins = $(I_{OH} \times 0.7)/(n \times 0.01)$
Example when n = 80% and $I_{OH} = -10.0\text{ mA}$
Total output current from the listed pins = $(-10.0 \times 0.7)/(80 \times 0.01) = -8.75\text{ mA}$
Note that the duty cycle has no effect on the current that is allowed to flow into a single pin. A current higher than the absolute maximum rating must not flow into a single pin.

Note 4. The maximum value is -50 mA with an ambient operating temperature range of 85°C to 125°C.

Note: The following pins are not capable of the output of high-level signals in the N-ch open-drain mode.

P100 to P102, P108 to P110, P112, P201, P212, P213, P300

Note: The characteristics of functions multiplexed on a given pin are the same as those for the port pin unless otherwise specified.

Table 27.6 I/O I_{OL}

Conditions: VCC = 1.6 to 5.5 V, VSS = 0 V, Ta = -40 to +125°C

Parameter		Symbol	Min.	Typ.	Max.	Unit	Test conditions
Allowable low-level output current*1	Per pin for P100 to P102, P108 to P110, P112, P201, P206, P300	I _{OL1}	—	—	20*2	mA	—
	Total of all pins (when duty ≤ 70%*3)		—	—	80*4	mA	4.0 V ≤ VCC ≤ 5.5 V
			—	—	35	mA	2.7 V ≤ VCC < 4.0 V
			—	—	20	mA	1.8 V ≤ VCC < 2.7 V
			—	—	10	mA	1.6 V ≤ VCC < 1.8 V
	Per pin for P010 to P013, P212, P213	I _{OL2}	—	—	8.5*2	mA	4.0 V ≤ VCC ≤ 5.5 V
			—	—	1.5*2	mA	2.7 V ≤ VCC < 4.0 V
			—	—	0.6*2	mA	1.8 V ≤ VCC < 2.7 V
			—	—	0.4*2	mA	1.6 V ≤ VCC < 1.8 V
	Total of all pins (when duty ≤ 70%*3)		—	—	20	mA	4.0 V ≤ VCC ≤ 5.5 V
			—	—	20	mA	2.7 V ≤ VCC < 4.0 V
			—	—	15	mA	1.8 V ≤ VCC < 2.7 V
			—	—	10	mA	1.6 V ≤ VCC < 1.8 V

Note 1. Device operation is guaranteed at the listed currents even if current is flowing from an output pin to VSS pin.

Note 2. The combination of these and other pins must also not exceed the value for maximum total current.

Note 3. The listed currents apply when the duty cycle is no greater than 70%. Use the following formula to calculate the output current when the duty cycle is greater than 70%, where n is the duty cycle.

- Total output current from the listed pins = $(I_{OL} \times 0.7)/(n \times 0.01)$
Example when n = 80% and $I_{OL} = 10.0\text{ mA}$
Total output current from the listed pins = $(10.0 \times 0.7)/(80 \times 0.01) = 8.75\text{ mA}$
Note that the duty cycle has no effect on the current that is allowed to flow into a single pin. A current higher than the absolute maximum rating must not flow into a single pin.

Note 4. The maximum value is 40 mA with an ambient operating temperature range of 85°C to 125°C.

Note: The characteristics of functions multiplexed on a given pin are the same as those for the port pin unless otherwise specified.

注2: 这些引脚和其他引脚的组合电流也不得超过最大总电流值。

注3. 所列电流适用于占空比不大于70%的情况。当占空比大于70% 时, 使用以下公式计算输出电流, 其中n为占空比。

- 所列引脚的总输出电流= $(I_{OH} \times 0.7)/(n \times 0.01)$
例如, 当n = 80%和 $I_{OH} = -10.0\text{ mA}$ 时
所列引脚的总输出电流= $(-10.0 \times 0.7)/(80 \times 0.01) = -8.75\text{ mA}$
请注意, 占空比对允许流入单个引脚的电流没有影响。流入单个引脚的电流不得超过绝对最大额定值。

注4. 最大值为-50 mA, 环境温度工作范围为85°C 至125°C。

注意: 在 N 沟道开漏模式下, 以下引脚不能输出高电平信号。

P100至P102、P108 至 P110、P112、P201、P212、P213、P300

注意: 除非另有规定, 否则在给定引脚上复用的功能的特性与端口引脚的特性相同。

表 27.6 I/O 人工晶状体

条件: VCC = 1.6至5.5 V, VSS = 0 V, Ta = -40 至+125°C

范围			象征	最小。	类型。	最大限度。	单元	测试条件
允许的低电平输出电流*1	P100 至 P102 的每个引脚, P108 至 P110、P112、 P201, P206, P300	I _{OL1}	—	—	20* ²	mA	—	
	所有引脚总数 (当 占空比为≤ 70%* ³)		—	—	80* ⁴	mA	4.0 V ≤ VCC ≤ 5.5 V	
			—	—	35	mA	2.7 V ≤ VCC < 4.0 V	
			—	—	20	mA	1.8 V ≤ VCC < 2.7 V	
			—	—	10	mA	1.6 V ≤ VCC < 1.8 V	
	P010 至 P013 的每个引脚, P212, P213	I _{OL2}	—	—	8.5* ²	mA	4.0 V ≤ VCC ≤ 5.5 V	
			—	—	1.5* ²	mA	2.7 V ≤ VCC < 4.0 V	
			—	—	0.6* ²	mA	1.8 V ≤ VCC < 2.7 V	
			—	—	0.4* ²	mA	1.6 V ≤ VCC < 1.8 V	
	所有引脚总数 (当职责≤ 70%* ³)		—	—	20	mA	4.0 V ≤ VCC ≤ 5.5 V	
			—	—	20	mA	2.7 V ≤ VCC < 4.0 V	
			—	—	15	mA	1.8 V ≤ VCC < 2.7 V	
			—	—	10	mA	1.6 V ≤ VCC < 1.8 V	

注 1. 即使电流从输出引脚流向 VSS 引脚, 也能保证器件在所列电流下正常工作。

注2: 这些引脚和其他引脚的组合电流也不得超过最大总电流值。

注3: 所列电流适用于占空比不大于70%的情况。当占空比不大于时, 请使用以下公式计算输出电流。

占空比大于70%, 其中 n 为占空比。

- 所列引脚的总输出电流= $(I_{OL} \times 0.7)/(n \times 0.01)$
例如, 当n = 80% 和 $I_{OL} = 10.0\text{ mA}$
所列引脚的总输出电流= $(10.0 \times 0.7)/(80 \times 0.01) = 8.75\text{ mA}$
请注意, 占空比对允许流入单个引脚的电流没有影响。
单个引脚的电流不得超过其绝对最大额定值。

注4: 最大值为40 mA, 环境温度工作范围为85°C至125°C。

注意: 除非另有规定, 否则在给定引脚上复用的功能的特性与端口引脚的特性相同。

Table 27.7 I/O V_{IH}, V_{IL}

Conditions: VCC = 1.6 to 5.5 V, VSS = 0 V, Ta = -40 to +125°C

Parameter			Symbol	Min.	Typ.	Max.	Unit	Test conditions
Input voltage, high	P100 to P102, P108 to P110, P112, P200, P201, P206, P300	Normal input buffer	V _{IH1}	VCC × 0.8	—	VCC	V	—
	P100 to P102, P108 to P110, P112, P201, P300	TTL input buffer	V _{IH2}	2.2	—	VCC	V	4.0 V ≤ VCC ≤ 5.5 V
				2.0	—	VCC	V	3.3 V ≤ VCC < 4.0 V
				1.5	—	VCC	V	1.6 V ≤ VCC < 3.3 V
	P010 to P013		V _{IH3}	VCC × 0.7	—	VCC	V	—
	P212 to P213		V _{IH4}	VCC × 0.8	—	VCC	V	—
Input voltage, low	P100 to P102, P108 to P110, P112, P200, P201, P206, P300	Normal input buffer	V _{IL1}	0	—	VCC × 0.2	V	—
	P100 to P102, P108 to P110, P112, P201, P300	TTL input buffer	V _{IL2}	0	—	0.8	V	4.0 V ≤ VCC ≤ 5.5 V
				0	—	0.5	V	3.3 V ≤ VCC < 4.0 V
				0	—	0.32	V	1.6 V ≤ VCC < 3.3 V
	P010 to P013		V _{IL3}	0	—	VCC × 0.3	V	—
	P212 to P213		V _{IL4}	0	—	VCC × 0.2	V	—

Note: The maximum value of V_{IH} of pins P100 to P102, P108 to P110, P112, P201, P212, P213, P300 is VCC, even in the N-ch open-drain mode.

Note: The characteristics of functions multiplexed on a given pin are the same as those for the port pin unless otherwise specified.

Table 27.8 I/O V_{OH}, V_{OL} (1 of 2)

Conditions: VCC = 1.6 to 5.5 V, VSS = 0 V, Ta = -40 to +125°C

Parameter		Symbol	Min.	Typ.	Max.	Unit	Test conditions
Output voltage, high	P100 to P102, P108 to P110, P112, P201, P206, P300	V _{OH1}	VCC - 1.5	—	—	V	4.0 V ≤ VCC ≤ 5.5 V I _{OH1} = -10 mA
			VCC - 0.7	—	—	V	4.0 V ≤ VCC ≤ 5.5 V I _{OH1} = -3 mA
			VCC - 0.6	—	—	V	2.7 V ≤ VCC ≤ 5.5 V I _{OH1} = -2 mA
			VCC - 0.5	—	—	V	1.8 V ≤ VCC ≤ 5.5 V I _{OH1} = -1.5 mA
			VCC - 0.5	—	—	V	1.6 V ≤ VCC ≤ 5.5 V I _{OH1} = -1 mA
	P010 to P013, P212, P213	V _{OH2}	VCC - 0.7	—	—	V	4.0 V ≤ VCC ≤ 5.5 V I _{OH2} = -3 mA
			VCC - 0.5	—	—	V	2.7 V ≤ VCC < 4.0 V I _{OH2} = -1 mA
			VCC - 0.5	—	—	V	1.8 V ≤ VCC < 2.7 V I _{OH2} = -1 mA
			VCC - 0.5	—	—	V	1.6 V ≤ VCC < 1.8 V I _{OH2} = -0.5 mA

表 27.7I/O V_{IH}, V_{IL}

条件: VCC = 1.6至5.5 V, VSS= 0 V, Ta = -40 至+125°C

范围			象征	最小。	类型。	最大限度。	单元	测试条件
输入电压高	P100 至 P102, P108 至 P110, P112, P200, P201, P206, P300	普通输入缓冲区	V _{IH1}	VCC × 0.8	—	VCC	V	—
	P100 至 P102, P108 至 P110, P112, P201, P300	TTL输入缓冲器	V _{IH2}	2.2	—	VCC	V	4.0 V ≤ VCC ≤ 5.5 V
				2.0	—	VCC	V	3.3 V ≤ VCC < 4.0 V
				1.5	—	VCC	V	1.6 V ≤ VCC < 3.3 V
	P010 至 P013		V _{IH3}	VCC × 0.7	—	VCC	V	—
	P212 至 P213		V _{IH4}	VCC × 0.8	—	VCC	V	—
输入电压低	P100 至 P102, P108 至 P110, P112, P200, P201, P206, P300	普通输入缓冲区	V _{IL1}	0	—	VCC × 0.2	V	—
	P100 至 P102, P108 至 P110, P112, P201, P300	TTL输入缓冲器	V _{IL2}	0	—	0.8	V	4.0 V ≤ VCC ≤ 5.5 V
				0	—	0.5	V	3.3 V ≤ VCC < 4.0 V
				0	—	0.32	V	1.6 V ≤ VCC < 3.3 V
	P010 至 P013		V _{IL3}	0	—	VCC × 0.3	V	—
	P212 至 P213		V _{IL4}	0	—	VCC × 0.2	V	—

注意: 引脚P100至P102, P108 至 P110、P112、P201、P212、P213、P300 的V_{IH}的最大值为 VCC, 即使在 N 沟道开漏模式下也是如此。

注意: 除非另有规定, 否则在给定引脚上复用的功能的特性与端口引脚的特性相同。

表 27.8I/O V_{OH}, V_{OL} (1/2)

条件: VCC = 1.6至5.5 V, VSS = 0 V Ta = -40 至+125°C

范围		象征	最小。	类型。	最大限	单元	测试条件
输出电压高	P100 至 P102, P108 至 P110, P112, P201, P206 P300	V _{OH1}	VCC - 1.5	—	度。	V	4.0 V ≤ VCC ≤ 5.5 V I _{OH1} = -10 mA
			VCC - 0.7	—	—	V	4.0 V ≤ VCC ≤ 5.5 V I _{OH1} = -3 mA
			VCC - 0.6	—	—	V	2.7 V ≤ VCC ≤ 5.5 V I _{OH1} = -2 mA
			VCC - 0.5	—	—	V	1.8 V ≤ VCC ≤ 5.5 V I _{OH1} = -1.5 mA
			VCC - 0.5	—	—	V	1.6 V ≤ VCC ≤ 5.5 V I _{OH1} = -1 mA
	P010 至 P013、P212、P213	V _{OH2}	VCC - 0.7	—	—	V	4.0 V ≤ VCC ≤ 5.5 V I _{OH2} = -3 mA
			VCC - 0.5	—	—	V	2.7 V ≤ VCC < 4.0 V I _{OH2} = -1 mA
			VCC - 0.5	—	—	V	1.8 V ≤ VCC < 2.7 V I _{OH2} = -1 mA
			VCC - 0.5	—	—	V	1.6 V ≤ VCC < 1.8 V I _{OH2} = -0.5 mA

Table 27.8 I/O V_{OH}, V_{OL} (2 of 2)

Conditions: VCC = 1.6 to 5.5 V, VSS = 0 V, Ta = -40 to +125°C

Parameter		Symbol	Min.	Typ.	Max.	Unit	Test conditions
Output voltage, low	P100 to P102, P108 to P110, P112, P201, P206, P300	V _{OL1}	—	—	1.3	V	4.0 V ≤ VCC ≤ 5.5 V I _{OL1} = 20 mA
			—	—	0.7	V	4.0 V ≤ VCC ≤ 5.5 V I _{OL1} = 8.5 mA
			—	—	0.6	V	2.7 V ≤ VCC ≤ 5.5 V I _{OL1} = 3 mA
			—	—	0.4	V	2.7 V ≤ VCC ≤ 5.5 V I _{OL1} = 1.5 mA
			—	—	0.4	V	1.8 V ≤ VCC ≤ 5.5 V I _{OL1} = 0.6 mA
			—	—	0.4	V	1.6 V ≤ VCC ≤ 5.5 V I _{OL1} = 0.3 mA
	P010 to P013, P212, P213	V _{OL2}	—	—	0.7	V	4.0 V ≤ VCC ≤ 5.5 V I _{OL2} = 8.5 mA
			—	—	0.5	V	2.7 V ≤ VCC < 4.0 V I _{OL2} = 1.5 mA
			—	—	0.4	V	1.8 V ≤ VCC < 2.7 V I _{OL2} = 0.6 mA
			—	—	0.4	V	1.6 V ≤ VCC < 1.8 V I _{OL2} = 0.4 mA

Note: P100 to P102, P108 to P110, P112, P201, P212, P213, P300 do not output high-level signals in the N-ch open-drain mode.

Note: The characteristics of functions multiplexed on a given pin are the same as those for the port pin unless otherwise specified.

Table 27.9 I/O other characteristics

Conditions: VCC = 1.6 to 5.5 V, VSS = 0 V, Ta = -40 to +125°C

Parameter		Symbol	Min.	Typ.	Max.	Unit	Test conditions
Input leakage current, high	P100 to P102, P108 to P110, P112, P200, P201, P206, P300	I _{LIH1}	—	—	1	μA	V _I = VCC
	P010 to P013	I _{LIH2}	—	—	1	μA	V _I = VCC
	P212 to P213	I _{LIH3}	—	—	1	μA	V _I = VCC
Input leakage current, low	P100 to P102, P108 to P110, P112, P200, P201, P206, P300	I _{LIL1}	—	—	-1	μA	V _I = VSS
	P010 to P013	I _{LIL2}	—	—	-1	μA	V _I = VSS
	P212 to P213	I _{LIL3}	—	—	-1	μA	V _I = VSS
On-chip pull-up resistance	P100 to P102, P108 to P110, P112, P201, P206, P300	R _U	10	20	100	kΩ	V _I = VSS In input port
Input capacitance	P200	C _{in}	—	—	30	pF	V _{in} = 0 V, f = 1 MHz, Ta = 25°C
	Other input pins		—	—	15		

Note: The characteristics of functions multiplexed on a given pin are the same as those for the port pin unless otherwise specified.

表 27.8I/O V_{OH}, V_{OL} (2/2)

条件: VCC = 1.6至5.5 V, VSS= 0 V, Ta = -40 至+125°C

范围		象征	最小。	类型。	最大限	单元	测试条件
输出电压低	P100 至 P102, P108 至 P110、P112、P201、P206 P300	V _{OL1}	—	—	1.3	V	4.0 V ≤ VCC ≤ 5.5 V I _{OL1} = 20 mA
			—	—	0.7	V	4.0 V ≤ VCC ≤ 5.5 V I _{OL1} = 8.5 mA
			—	—	0.6	V	2.7 V ≤ VCC ≤ 5.5 V I _{OL1} = 3 mA
			—	—	0.4	V	2.7 V ≤ VCC ≤ 5.5 V I _{OL1} = 1.5 mA
			—	—	0.4	V	1.8 V ≤ VCC ≤ 5.5 V I _{OL1} = 0.6 mA
			—	—	0.4	V	1.6 V ≤ VCC ≤ 5.5 V I _{OL1} = 0.3 mA
	P010 至 P013、P212、P213	V _{OL2}	—	—	0.7	V	4.0 V ≤ VCC ≤ 5.5 V I _{OL2} = 8.5 mA
			—	—	0.5	V	2.7 V ≤ VCC < 4.0 V I _{OL2} = 1.5 mA
			—	—	0.4	V	1.8 V ≤ VCC < 2.7 V I _{OL2} = 0.6 mA
			—	—	0.4	V	1.6 V ≤ VCC < 1.8 V I _{OL2} = 0.4 mA

注意: P100 至P102, P108 至 P110、P112、P201、P212、P213、P300 在 N 沟道开漏模式下不输出高电平信号。

注意: 除非另有规定, 否则在给定引脚上复用的功能的特性与端口引脚的特性相同。

表 27.9 I/O 其他特性

条件: VCC = 1.6至5.5 V, VSS= 0V, Ta= -40 至+125°C

范围		象征	最小。	类型。	最大限度。	单元	测试条件
输入漏电流高	P100 至 P102, P108 至 P110、P112、P200、P201、P206, P300	I _{LIH1}	—	—	1	μA	V _I = VCC
		I _{LIH2}	—	—	1	μA	V _I = VCC
		I _{LIH3}	—	—	1	μA	V _I = VCC
	P212 至 P213	I _{LIH3}	—	—	1	μA	V _I = VCC
输入漏电流低	P100 至 P102, P108 至 P110、P112、P200、P201、P206, P300	I _{LIL1}	—	—	-1	μA	V _I = VSS
		I _{LIL2}	—	—	-1	μA	V _I = VSS
		I _{LIL3}	—	—	-1	μA	V _I = VSS
片上上拉电阻	P100 至 P102, P108 至 P110、P112、P201、P206 P300	R _U	10	20	100	kΩ	V _I = VSS 输入端口
输入电容	P200	吃	—	—	30	pF	V _{in} = 0 V, f = 1 MHz, Ta = 25°C
	其他输入引脚		—	—	15		

注意: 除非另有规定, 否则在给定引脚上复用的功能的特性与端口引脚的特性相同。

27.3.2 Operating and Standby Current

Table 27.10 Operating and standby current (1) (1 of 2)

Conditions: VCC = 1.6 to 5.5 V

Parameter					Symbol	Typ.*4	Max.	Unit	Test Conditions
Supply current*1	High-speed mode*2	Normal mode	All peripheral clocks disabled, CoreMark code executing from flash	ICLK = 32 MHz	Icc	2.6	—	mA	—
			All peripheral clocks enabled, CoreMark code executing from flash*5	ICLK = 32 MHz		—	3.5		—
		Sleep mode	All peripheral clocks disabled	ICLK = 32 MHz		0.82	—		—
			All peripheral clocks enabled*5	ICLK = 32 MHz		—	1.4		—
	Middle-speed mode*2	Normal mode	All peripheral clocks disabled, CoreMark code executing from flash	ICLK = 16 MHz		1.5	—		—
				ICLK = 8 MHz		0.9	—		—
				ICLK = 4 MHz		0.70	—		—
			All peripheral clocks enabled, CoreMark code executing from flash*5	ICLK = 16 MHz		—	2.0		—
				ICLK = 8 MHz		—	1.3		—
				ICLK = 4 MHz		—	0.9		—
		Sleep mode	All peripheral clocks disabled	ICLK = 16 MHz		0.61	—		—
				ICLK = 8 MHz		0.50	—		—
				ICLK = 4 MHz		0.44	—		—
			All peripheral clocks enabled*5	ICLK = 16 MHz		—	0.9		—
				ICLK = 8 MHz		—	0.7		—
				ICLK = 4 MHz		—	0.6		—
	Low-speed mode*2	Normal mode	All peripheral clocks disabled, CoreMark code executing from flash	ICLK = 2 MHz		450	—	μA	—
			All peripheral clocks enabled, CoreMark code executing from flash*5	ICLK = 2 MHz		—	600		—
		Sleep mode	All peripheral clocks disabled	ICLK = 2 MHz		324	—		—
			All peripheral clocks enabled*5	ICLK = 2 MHz		—	448		—

27.3.2 工作电流和待机电流

表 27.10 工作电流和待机电流 (1) (1/2)

条件: VCC = 1.6至5.5 V

范围					象征	4型	最大	单元	测试状况
供电电流*1	高速模式*2	正常模式	所有外设时钟均已禁用, CoreMark 代码从闪存执行	ICLK = 32 MHz	国际商会	2.6	限度。	mA	—
			所有外设时钟均已启用, CoreMark 代码从闪存*5 执行	ICLK = 32 MHz		—	3.5		—
		睡眠模式	所有外设时钟均已禁用	ICLK = 32 MHz		0.82	—		—
			所有外设时钟均已启用*5	ICLK = 32 MHz		—	1.4		—
	中速模式*2	正常模式	所有外设时钟均已禁用, CoreMark 代码从闪存执行	ICLK = 16 MHz		1.5	—		—
				ICLK = 8 MHz		0.9	—		—
				ICLK = 4 MHz		0.70	—		—
			所有外设时钟已启用, CoreMark 代码正在执行 来自闪光灯*5	ICLK = 16 MHz		—	2.0		—
				ICLK = 8 MHz		—	1.3		—
				ICLK = 4 MHz		—	0.9		—
		睡眠模式	所有外设时钟均已禁用	ICLK = 16 MHz		0.61	—		—
				ICLK = 8 MHz		0.50	—		—
				ICLK = 4 MHz		0.44	—		—
			所有外设时钟均已启用*5	ICLK = 16 MHz		—	0.9		—
				ICLK = 8 MHz		—	0.7		—
				ICLK = 4 MHz		—	0.6		—
	低速模式*2	正常模式	所有外设时钟均已禁用, CoreMark 代码从闪存执行	ICLK = 2 MHz		450	—	μA	—
			所有外设时钟均已启用, CoreMark 代码从闪存*5 执行	ICLK = 2 MHz		—	600		—
		睡眠模式	所有外设时钟均已禁用	ICLK = 2 MHz		324	—		—
			所有外设时钟均已启用*5	ICLK = 2 MHz		—	448		—

Table 27.10 Operating and standby current (1) (2 of 2)

Conditions: VCC = 1.6 to 5.5 V

Parameter						Symbol	Typ.*4	Max.	Unit	Test Conditions
Supply current*1	Subosc-speed mode*3	Normal mode	Peripheral clocks disabled	ICLK = 32.768 kHz	Ta = -40°C	Icc	2.8	—	μA	—
					Ta = 25°C		3.0	—		
					Ta = 50°C		3.2	—		
					Ta = 70°C		3.3	—		
					Ta = 85°C		3.6	—		
					Ta = 105°C		4.2	—		
					Ta = 125°C		5.5	—		
			Peripheral clocks enabled*6	ICLK = 32.768 kHz	Ta = -40°C		—	6.0		
					Ta = 25°C		—	6.2		
					Ta = 50°C		—	9.2		
					Ta = 70°C		—	12.6		
					Ta = 85°C		—	15.9		
					Ta = 105°C		—	21.7		
					Ta = 125°C		—	42.0		
		Sleep mode	Peripheral clocks disabled	ICLK = 32.768 kHz	Ta = -40°C		0.8	—		—
					Ta = 25°C		0.9	—		
					Ta = 50°C		1.0	—		
					Ta = 70°C		1.1	—		
					Ta = 85°C		1.3	—		
					Ta = 105°C		1.7	—		
					Ta = 125°C		2.9	—		
			Peripheral clocks enabled*6	ICLK = 32.768 kHz	Ta = -40°C		—	3.6		
					Ta = 25°C		—	3.8		
					Ta = 50°C		—	6.2		
					Ta = 70°C		—	9.3		
					Ta = 85°C		—	12.6		
					Ta = 105°C		—	18.7		
					Ta = 125°C		—	38.5		

Note 1. Supply current is the total current flowing into VCC. Supply current values apply when internal pull-up MOSs are in the off state and these values do not include output charge/discharge current from any of the pins.

Note 2. The clock source is high-speed on-chip oscillator (HOCO).

Note 3. The clock source is LOCO.

Note 4. VCC = 3.3 V.

Note 5. Includes operating current for PCLBUZ, TAU, SAU, and IICA functions only. For other peripheral operating currents, please add the current in Peripheral Functions Supply current in [Table 27.12](#).

Note 6. Includes operating current for PCLBUZ, TAU and SAU functions only. For other peripheral operating currents, please add the current in Peripheral Functions Supply current in [Table 27.12](#).

表 27.10 工作电流和待机电流 (1) (2/2)

条件: VCC = 1.6至5.5 V

范围						象征	4型	最大	单元	测试状况
供电电流*1	次振荡模式*3	正常模式	外围时钟已禁用	ICLK = 32.768 kHz	Ta = -40°C	国际商会	2.8	限度。	μA	—
					Ta = 25°C		3.0	—		
					Ta = 50°C		3.2	—		
					Ta = 70°C		3.3	—		
					Ta = 85°C		3.6	—		
					Ta = 105°C		4.2	—		
					Ta = 125°C		5.5	—		
			外围时钟已启用*6	ICLK = 32.768 kHz	Ta = -40°C		—	6.0		
					Ta = 25°C		—	6.2		
					Ta = 50°C		—	9.2		
					Ta = 70°C		—	12.6		
					Ta = 85°C		—	15.9		
					Ta = 105°C		—	21.7		
					Ta = 125°C		—	42.0		
		睡眠模式	外围时钟已禁用	ICLK = 32.768 kHz	Ta = -40°C		0.8	—		—
					Ta = 25°C		0.9	—		
					Ta = 50°C		1.0	—		
					Ta = 70°C		1.1	—		
					Ta = 85°C		1.3	—		
					Ta = 105°C		1.7	—		
					Ta = 125°C		2.9	—		
			外围时钟已启用*6	ICLK = 32.768 kHz	Ta = -40°C		—	3.6		
					Ta = 25°C		—	3.8		
					Ta = 50°C		—	6.2		
					Ta = 70°C		—	9.3		
					Ta = 85°C		—	12.6		
					Ta = 105°C		—	18.7		
					Ta = 125°C		—	38.5		

注1: 电源电流是指流入VCC的总电流。电源电流值适用于内部上拉MOS管处于截止状态的情况, 并且这些值不包括任何引脚的输出充放电电流。

注2: 时钟源为高速片上振荡器 (HOCO)。

注3.时钟源为LOCO。

Note 4. VCC = 3.3 V.

注5: 仅包含PCLBUZ、TAU、SAU和IICA功能的运行电流。其他外围设备的运行电流, 请参考表27.12中的“外围设备供电电流”部分。

注6: 仅包含PCLBUZ、TAU和SAU功能的运行电流。其他外设的运行电流, 请参考表27.12中的“外设功能供电电流”部分。

Table 27.11 Operating and standby current (2)

Conditions: VCC = 1.6 to 5.5 V

Parameter				Symbol	Typ.*3	Max.	Unit	Test conditions
Supply current*1	Software Standby mode*2	Peripheral modules stop	Ta = -40°C	Icc	0.2	0.9	μA	—
			Ta = 25°C		0.2	0.9		
			Ta = 50°C		0.3	2.2		
			Ta = 70°C		0.4	4.7		
			Ta = 85°C		0.5	8.1		
			Ta = 105°C		1.1	17		
			Ta = 125°C		2.3	35		

Note 1. Supply current is the total current flowing into VCC. Supply current values apply when internal pull-up MOSs are in the off state and these values do not include output charge/discharge current from any of the pins.

Note 2. The IWDT and LVD are not operating.

Note 3. VCC = 3.3 V.

Table 27.12 Peripheral Functions Supply current

Conditions: VCC = 1.6 to 5.5 V

Parameter			Symbol	Typ.*10	Max.	Unit	Test conditions	
Peripheral Functions Supply current*1	High-speed on chip oscillator operating current*1		OFS1.HOCOFRQ1[2:0] are 010b	I _{HOCO}	320	—	μA	—
	Low-speed on chip oscillator operating current*1			I _{LOCO}	0.24	—	μA	—
	32-bit interval timer operating current*1*2*3			I _{IT}	0.06	—	μA	—
	Independent watchdog timer operating current*1*2*4		f _{LOCO} = 32.768 kHz (typ.)	I _{IWDT}	0.03	—	μA	—
	A/D converter operating current*1*5	When conversion at maximum speed	Normal mode, VREFH0 = VCC = 5.0 V	I _{ADC}	1.3	1.7	mA	—
			Low voltage mode, VREFH0 = VCC = 3.0 V		0.5	0.7	mA	—
	A/D converter internal reference voltage current*1			I _{ADREF}	100	—	μA	—
	Temperature sensor operating current*1			I _{TMPS}	100	—	μA	—
	LVD operating current*1	LVD0 is enabled*7	I _{LVD0}	0.03	—	μA	—	
		LVD1 is enabled*8	I _{LVD1}	0.03	—	μA	—	
	Self-programming operating current*1*9			I _{FSP}	—	12.2	mA	—
DTC		Data transfer to RAM	I _{DTC}	1.82	—	mA	—	

Note 1. This current flows into V_{CC}.

Note 2. The listed currents apply when the high-speed on-chip oscillator (HOCO) is stopped.

Note 3. This current only flows to the 32-bit interval timer. It does not include the operating current of the low-speed on-chip oscillator (LOCO).

The supply current of the RA0 microcontrollers is the sum of either I_{cc} and I_{IT}.
When the low-speed on-chip oscillator (LOCO) is selected, I_{LOCO} should be included in the supply current.

Note 4. This current only flows to the independent watchdog timer. It does not include the operating current of the low-speed on-chip oscillator (LOCO) .

The supply current of the RA0 microcontrollers is the sum of either I_{cc}, I_{IWDT} and I_{LOCO}.

Note 5. This current only flows to the A/D converter. The supply current of the RA0 microcontrollers is the sum of I_{cc} and I_{ADC} when the A/D converter is operating or in the SLEEP mode.

Note 6. This current flows into VREFH0.

Note 7. This current only flows to the LVD0 circuit. The supply current of the RA0 microcontrollers is the sum of I_{cc} and I_{LVD0} when the LVD0 circuit is in operation.

Note 8. This current only flows to the LVD1 circuit. The supply current of the RA0 microcontrollers is the sum of I_{cc} and I_{LVD1} when the LVD1 circuit is in operation.

Note 9. This current only flows during self programming.

Note 10. VCC = 3.3 V.

表 27.11 工作电流和待机电流 (2)

条件: VCC = 1.6到 5.5 V

范围				象征	类型.*3	最大限度。	单元	测试条件
供电电流*1	软件待机模式*2	外围模块停止	Ta = -40°C	国际商会	0.2	0.9	μA	—
			Ta = 25°C		0.2	0.9		
			Ta = 50°C		0.3	2.2		
			Ta = 70°C		0.4	4.7		
			Ta = 85°C		0.5	8.1		
			Ta = 105°C		1.1	17		
			Ta = 125°C		2.3	35		

注1：电源电流是指流入VCC的总电流。电源电流值适用于内部上拉MOS管处于截止状态的情况，并且这些值不包括任何引脚的输出充放电电流。

注2. IWDT和LVD未运行。

Note 3. VCC = 3.3 V.

表 27.12 外围功能供电电流

条件: VCC = 1.6到 5.5 V

范围			象征	类型*10	最大限度。	测试条件		
外围功能 供电电流* 1	高速片上振荡器工作电流*1		OFS1.HOCOFRQ1[2:0] are 010b	I _{HOCO}	320	—	μA	—
	低速片上振荡器工作电流*1			I _{LOCO}	0.24	—	μA	—
	32 位间隔定时器工作电流*1*2*3			I _{IT}	0.06	—	μA	—
	独立看门狗定时器工作电流*1*2*4		f _{LOCO} = 32.768 kHz（典型值）	I _{IWDT}	0.03	—	μA	—
	A/D转换器工作电流*1*5	转换时	正常模式，VREFH0= VCC = 5.0 V	I _{ADC}	1.3	1.7	mA	—
		最高速度	低电压模式，VREFH0= VCC = 3.0 V		0.5	0.7	mA	—
	A/D转换器内部参考电压电流*1			I _{ADREF}	100	—	μA	—
	温度传感器工作电流*1			I _{TMPS}	100	—	μA	—
	低压差线性稳压器工作电流*1	LVD0 已启用*7		I _{LVD0}	0.03	—	μA	—
		LVD1 已启用*8		I _{LVD1}	0.03	—	μA	—
自编程工作电流*1*9			I _{FSP}	—	12.2	mA	—	
DTC		数据传输到内存	I _{DTC}	1.82	—	mA	—	

注1. 此电流流入V_{CC}.

注 2. 所列电流适用于高速片上振荡器 (HOCO) 停止工作的情况。

注3. 此电流仅流向32位间隔定时器。它不包括低速片上振荡器（LOCO）的工作电流。

RA0 微控制器的供电电流是 I_{cc} 和I_{IT}之和。

选择低速片上振荡器 (LOCO) 时，应将 I_{LOCO} 包含在电源电流中。

注 4. 此电流仅流向独立的看门狗定时器。它不包括低速片上振荡器 (LOCO) 的工作电流。

RA0 微控制器的供电电流是 I_{cc}、IIWDT 和 I_{LOCO} 之和。

注5. 此电流仅流向A/D转换器。当A/D转换器工作或处于睡眠模式时，RA0微控制器的供电电流为I_{cc}和I_{ADC}之和。

注 6. 此电流流入 VREFH0。

注7：该电流仅流向LVD0电路。当LVD0电路工作时，RA0微控制器的供电电流为I_{cc}和I_{LVD0}之和。

注8. 此电流仅流向LVD1电路。当LVD1电路工作时，RA0微控制器的供电电流是I_{cc}和I_{LVD1}之和。

注9：此电流仅在自编程期间流动。

Note 10. VCC = 3.3 V.

27.3.3 Thermal Characteristics

The maximum value of junction temperature (Tj) must not exceed the value specified in the [section 27.1.1. Tj/Ta Definition](#).

Tj is calculated by either of the following equations.

- $T_j = T_a + \theta_{ja} \times \text{Total power consumption}$
- $T_j = T_t + \Psi_{jt} \times \text{Total power consumption}$
Tj : Junction Temperature (°C)
Ta : Ambient Temperature (°C)
Tt : Top Center Case Temperature (°C)
θja : Thermal Resistance of “Junction”-to-“Ambient” (°C/W)
Ψjt : Thermal Resistance of “Junction”-to-“Top Center Case” (°C/W)
- Total power consumption = Voltage × (Leakage current + Dynamic current)
- Leakage current of IO = $\Sigma (I_{OL} \times V_{OL}) / \text{Voltage} + \Sigma (|I_{OH}| \times |V_{CC} - V_{OH}|) / \text{Voltage}$
- Dynamic current of IO = $\Sigma IO (C_{in} + C_{load}) \times IO \text{ switching frequency} \times \text{Voltage}$
Cin: Input capacitance
Cload: Output capacitance

Regarding θja and Ψjt, see [Table 27.13](#).

Table 27.13 Thermal resistance

Parameter	Package	Symbol	Value*1	Unit	Test condition
Thermal resistance	20-pin TSSOP	θja	76.0	°C/W	JESD 51-2 and 51-7 compliant
	20-pin TSSOP	Ψjt	2.95	°C/W	JESD 51-2 and 51-7 compliant

Note 1. The values are reference values when the 4-layer board is used. Thermal resistance depends on the number of layers or size of the board. For details, refer to the JEDEC standards.

27.4 AC Characteristics

Table 27.14 AC characteristics (1 of 2)

Conditions: VCC = 1.6 to 5.5 V, VSS = 0 V, Ta = -40 to +125°C

Parameter			Symbol	Min.	Typ.	Max.	Unit	Test conditions
Instruction cycle (minimum instruction execution time)	Main system clock (FMAIN) operation	High-speed mode	T _{CY}	0.03125	—	1	μs	1.8 V ≤ VCC ≤ 5.5 V
				0.25	—	1	μs	1.6 V ≤ VCC < 1.8 V
		Middle-speed mode		0.0625	—	1	μs	1.8 V ≤ VCC ≤ 5.5 V
				0.25	—	1	μs	1.6 V ≤ VCC < 1.8 V
		Low-speed mode		0.5	—	1	μs	1.6 V ≤ VCC ≤ 5.5 V
	Subsystem clock (FSUB) operation			26.041	30.5	31.3	μs	1.6 V ≤ VCC ≤ 5.5 V
	In the self-programming mode	High-speed mode		0.03125	—	1	μs	1.8 V ≤ VCC ≤ 5.5 V
		Middle-speed mode		0.0625	—	1	μs	1.8 V ≤ VCC ≤ 5.5 V
TI00 to TI07 input high-level width, low-level width			t _{TIH} t _{TIL}	1/f _{MCK} + 10*1	—	—	ns	

27.3.3 热特性

结温 (Tj) 的最大值不得超过第 27.1.1 节中规定的值。Tj/Ta 定义。

Tj 可通过以下任一公式计算。

- $T_j = T_a + \theta_{ja} \times \text{总功耗}$
- $T_j = T_t + \Psi_{jt} \times \text{总功耗}$
Tj: 结温(°C)
Ta: 环境温度(°C)
Tt: 顶部中心外壳温度(°C)
θja: 结到环境的热阻(°C/W)
Ψjt: 结到“顶部中心外壳”的热阻(°C/W)
- 总功耗= 电压× (漏电流+ 动态电流)
- IO = Σ 的漏电流 $(I_{OL} \times V_{OL}) / \text{Voltage} + \Sigma (|I_{OH}| \times |V_{CC} - V_{OH}|) / \text{Voltage}$
- 动态电流IO = $\Sigma IO (C_{in} + C_{load}) \times IO \text{开关频率} \times \text{电压}$
Cin: 输入电容
Cload: 输出电容

关于θja 和Ψjt, 请参见表 27.13。

表 27.13 热阻

范围	包裹	象征	值*1	单元	测试条件
热阻	20引脚TSSOP封装	θja	76.0	°C/W	符合 JESD 51-2 和 51-7 标准
	20引脚TSSOP封装	Ψjt	2.95	°C/W	符合 JESD 51-2 和 51-7 标准

注1： 这些数值是使用四层板时的参考值。热阻取决于层数或电路板尺寸。详情请参阅JEDEC标准。

27.4AC特性

表 27.14 交流特性 (1/2)

条件: VCC = 1.6至5.5 V, VSS= 0 V, Ta = -40 至+125°C

范围			象征	最小。	类型。	最大限	单元	测试条件
指令周期（最短指令执行时间）	主系统时钟（FMAIN）操作	高速模式	T _{CY}	0.03125	—	—	μs	1.8 V ≤ VCC ≤ 5.5 V
				0.25	—	1	μs	1.6 V ≤ VCC < 1.8 V
		中速模式		0.0625	—	1	μs	1.8 V ≤ VCC ≤ 5.5 V
				0.25	—	1	μs	1.6 V ≤ VCC < 1.8 V
		低速模式		0.5	—	1	μs	1.6 V ≤ VCC ≤ 5.5 V
	子系统时钟（FSUB） 手术			26.041	30.5	31.3	μs	1.6 V ≤ VCC ≤ 5.5 V
	在自编程模式下	高速模式		0.03125	—	1	μs	1.8 V ≤ VCC ≤ 5.5 V
		中速模式		0.0625	—	1	μs	1.8 V ≤ VCC ≤ 5.5 V
TI00 至 TI07 输入高电平宽度、低电平宽度			t _{TIH} t _{TIL}	1/f _{MCK} + 10*1	—	—	ns	

Table 27.14 AC characteristics (2 of 2)

Conditions: VCC = 1.6 to 5.5 V, VSS = 0 V, Ta = -40 to +125°C

Parameter		Symbol	Min.	Typ.	Max.	Unit	Test conditions
TO00 to TO07 output frequency	High-speed mode Middle-speed mode	f _{TO}	—	—	16*2	MHz	4.0 V ≤ VCC ≤ 5.5 V
			—	—	8	MHz	2.7 V ≤ VCC < 4.0 V
			—	—	4	MHz	1.8 V ≤ VCC < 2.7 V
			—	—	2	MHz	1.6 V ≤ VCC < 1.8 V
	Low-speed mode		—	—	2	MHz	1.6 V ≤ VCC ≤ 5.5 V
PCLBUZ0 output frequency	High-speed mode Middle-speed mode	f _{PCL}	—	—	16*2	MHz	4.0 V ≤ VCC ≤ 5.5 V
			—	—	8	MHz	2.7 V ≤ VCC < 4.0 V
			—	—	4	MHz	1.8 V ≤ VCC < 2.7 V
			—	—	2	MHz	1.6 V ≤ VCC < 1.8 V
	Low-speed mode		—	—	2	MHz	1.6 V ≤ VCC ≤ 5.5 V
Interrupt input high-level width, low-level width	NMI/IRQ0, IRQ1 to IRQ5	f _{IRQH} f _{IRQL}	1	—	—	μs	1.6 V ≤ VCC ≤ 5.5 V

Note 1. f_{MCK}: Timer array unit operating clock frequency
To set this operating clock, use the CKS[1:0] bits of the timer mode register 0n (TMR0n).
m: Unit number (m = 0), n: Channel number (n = 0 to 7)

Note 2. The maximum value is 12MHz with an ambient operating temperature range of 105°C to 125°C.

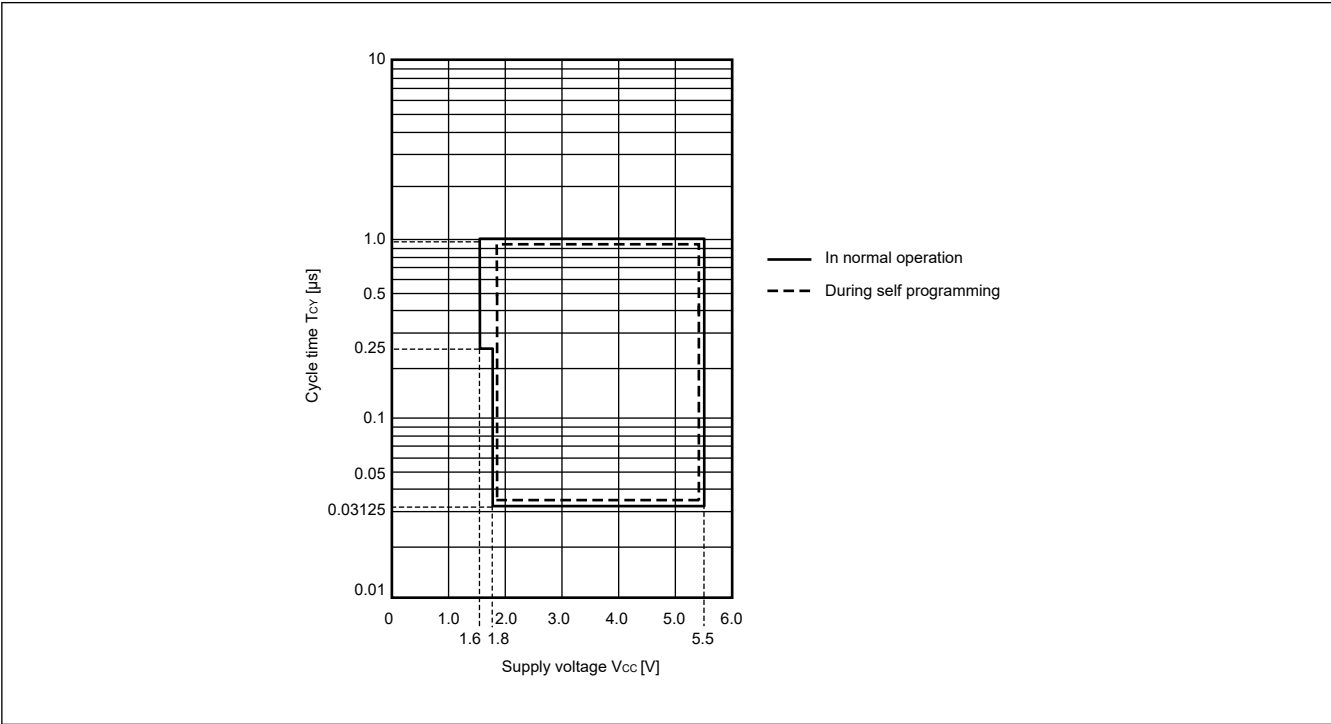


Figure 27.2 Tcy vs Vcc in High-speed mode

表 27.14 交流特性 (2/2)

条件: VCC = 1.6至5.5 V, VSS= 0V, Ta = -40 至+125°C

范围		象征	最小。	类型。	最大限	单元	测试条件
TO00至TO07输出频率	高速模 式中速 模式	f _{TO}	—	—	16*2	MHz	4.0 V ≤ VCC ≤ 5.5 V
			—	—	8	MHz	2.7 V ≤ VCC < 4.0 V
			—	—	4	MHz	1.8 V ≤ VCC < 2.7 V
			—	—	2	MHz	1.6 V ≤ VCC < 1.8 V
	低速模式		—	—	2	MHz	1.6 V ≤ VCC ≤ 5.5 V
PCLBUZ0 输出频率	高速模 式中速 模式	f _{PCL}	—	—	16*2	MHz	4.0 V ≤ VCC ≤ 5.5 V
			—	—	8	MHz	2.7 V ≤ VCC < 4.0 V
			—	—	4	MHz	1.8 V ≤ VCC < 2.7 V
			—	—	2	MHz	1.6 V ≤ VCC < 1.8 V
	低速模式		—	—	2	MHz	1.6 V ≤ VCC ≤ 5.5 V
中断输入高电平宽度, 低电平宽度	NMI/IRQ0, IRQ1 到 IRQ5	f _{IRQH} f _{IRQL}	1	—	—	μs	1.6 V ≤ VCC ≤ 5.5 V

注1. f_{MCK}: 定时器阵列单元工作时钟频率
要设置此工作时钟, 请使用定时器模式寄存器 0n (TMR0n) 的 CKS[1:0] 位。
m: 单元编号(m = 0), n: 通道编号(n = 0 至7)

注2. 最大值为12MHz, 环境工作温度范围为105°C 至125°C。

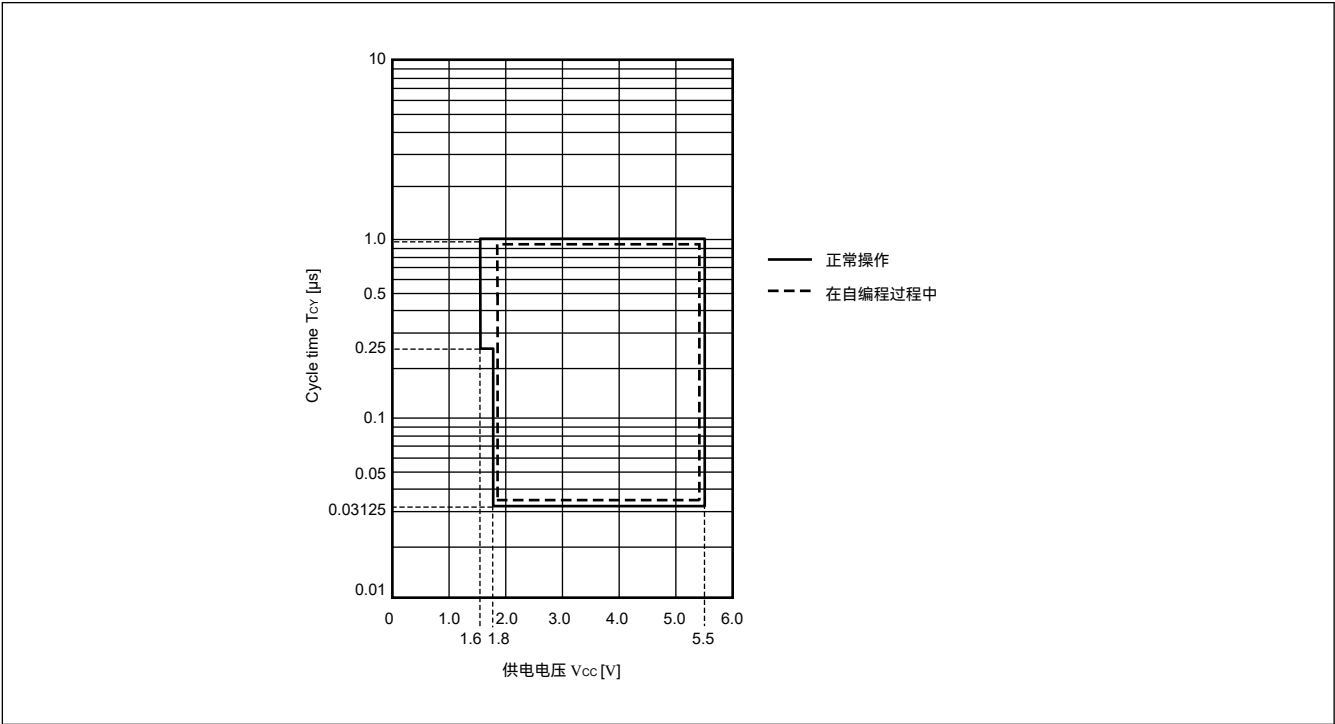


图 27.2TCY 与高速模式下的Vcc的关系

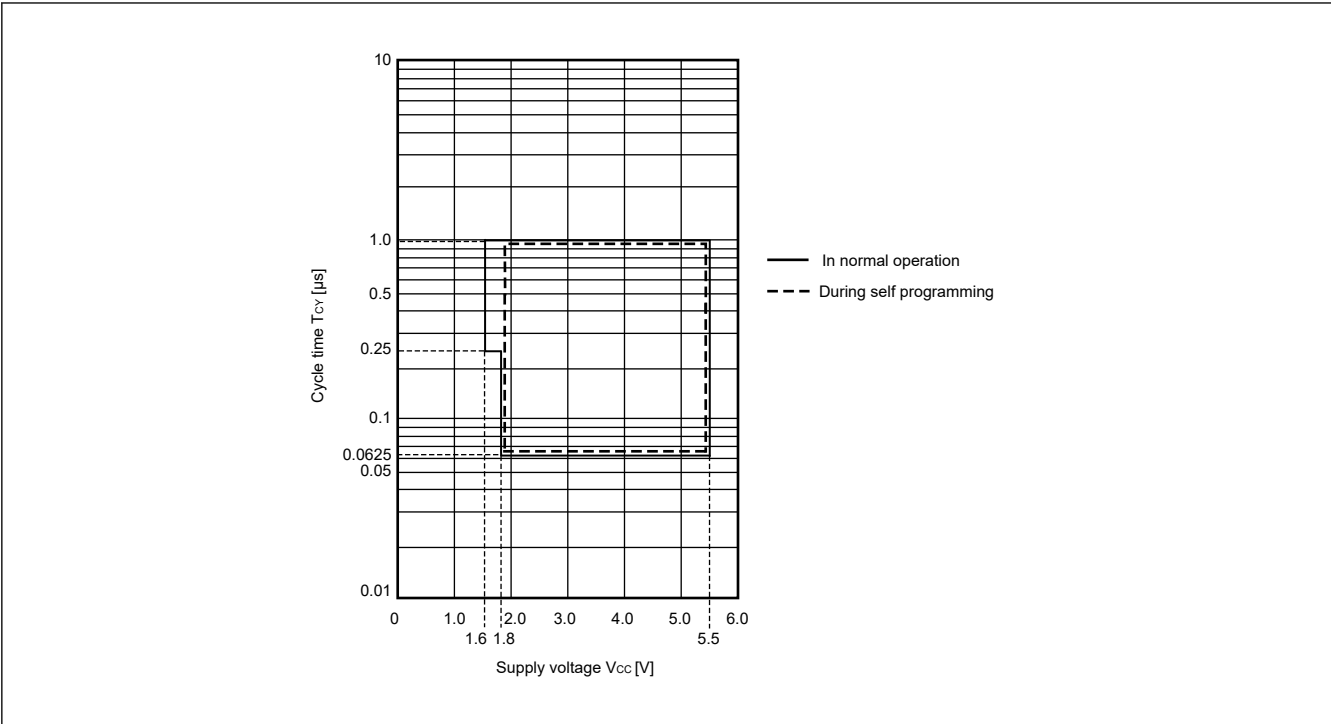


Figure 27.3 T_{CY} vs V_{CC} in Middle-speed mode

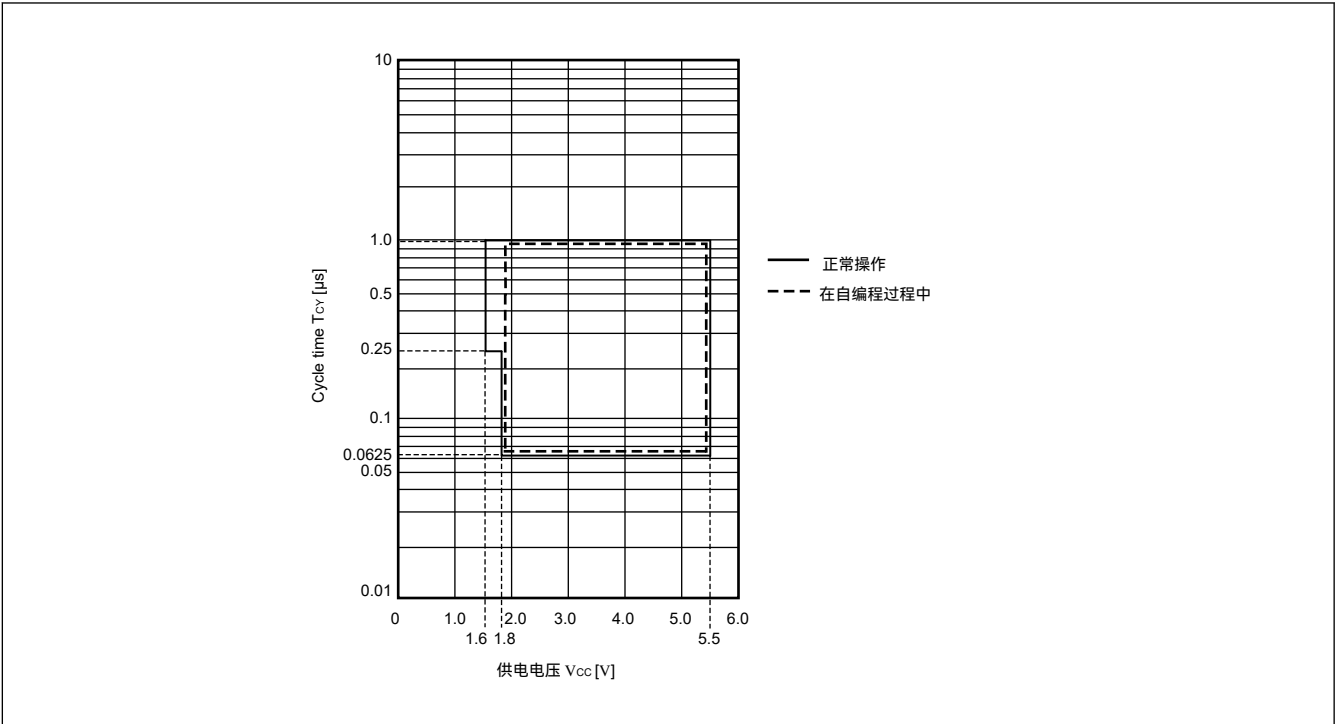


图 27.3 中速模式下的 TCY 与Vcc的关系

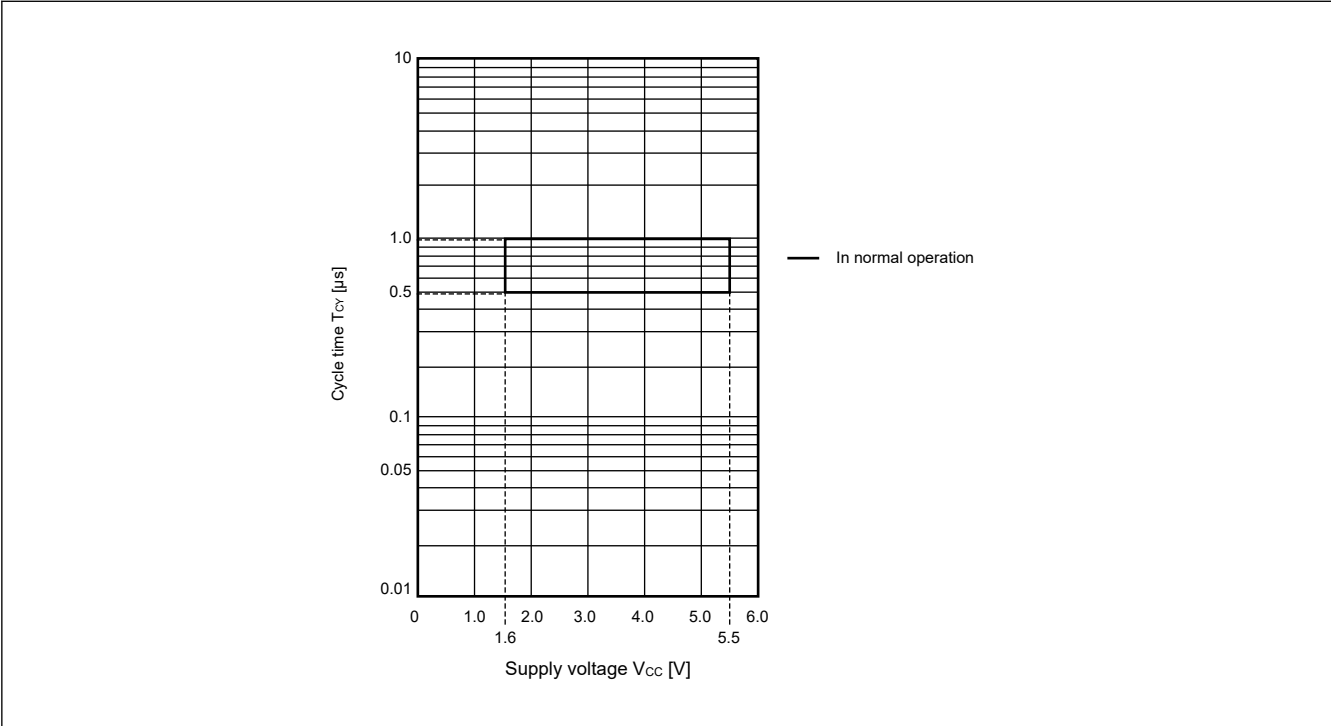


Figure 27.4 T_{CY} vs V_{CC} in Low-speed mode

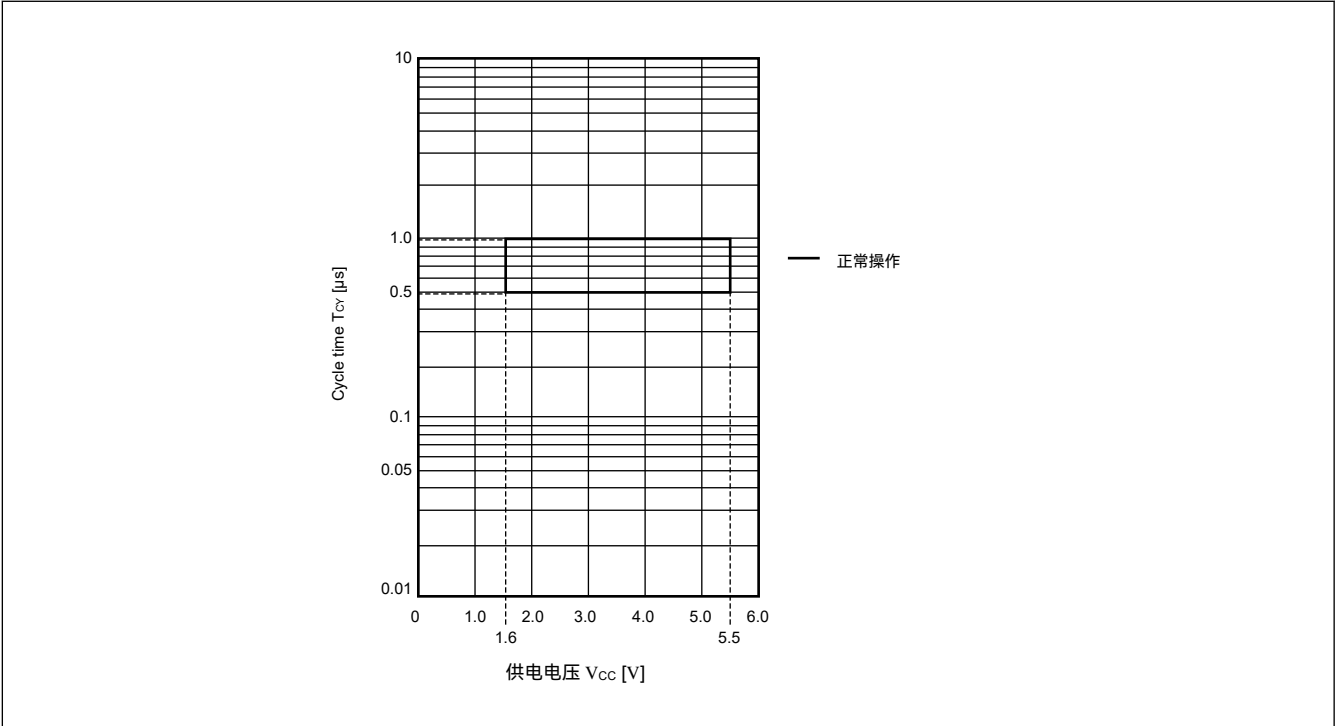


图 27.4 低速模式下 TCY 与Vcc的关系

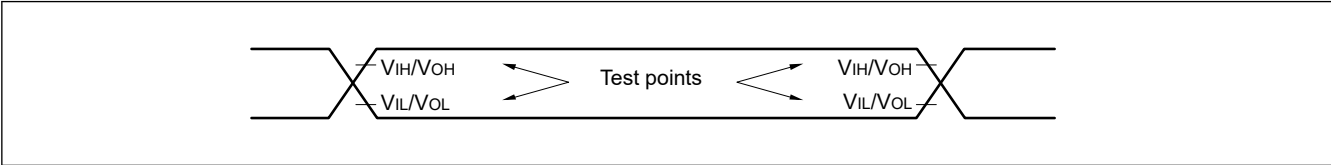


Figure 27.5 AC timing test points

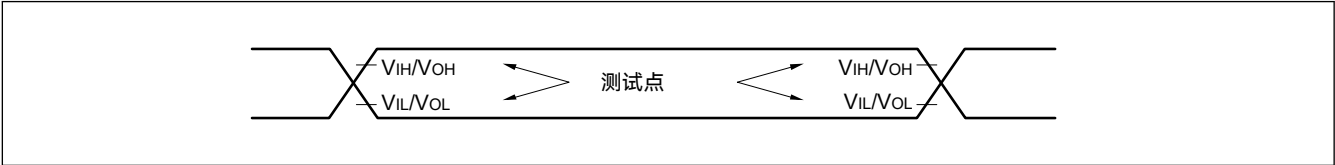


图 27.5AC 时序测试点

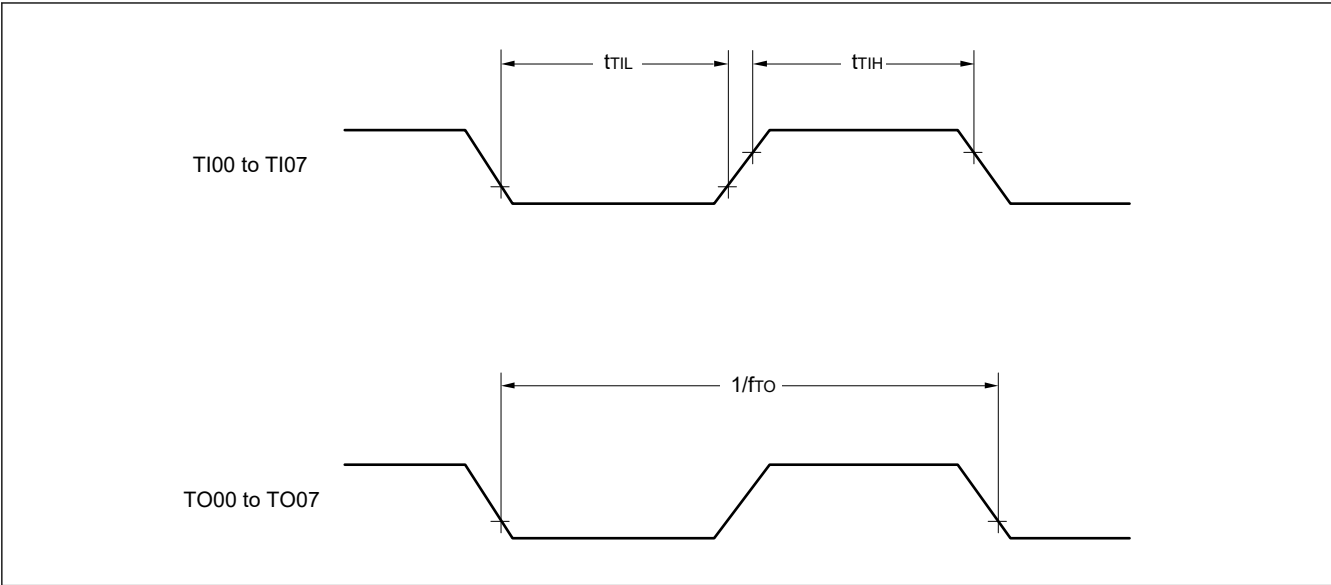


Figure 27.6 TI/TO timing

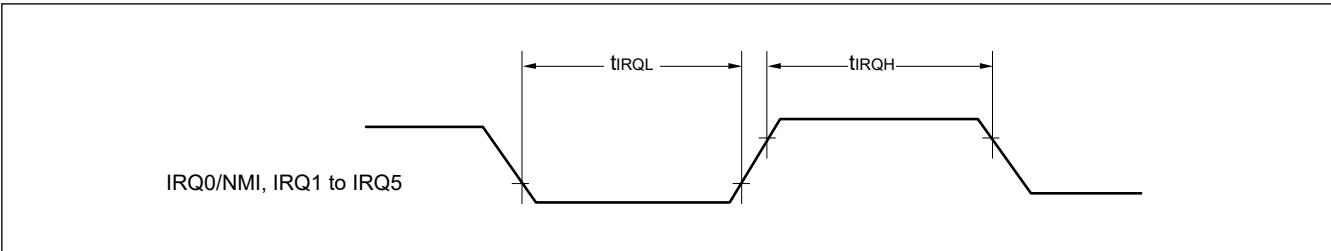


Figure 27.7 IRQ interrupt input timing

27.4.1 Reset Timing

Table 27.15 Reset timing

Parameter		Symbol	Min.	Typ.	Max.	Unit	Test conditions
RES pulse width	At power-on*3	tRESWP	9.9	—	—	ms	—
	Not at power-on	tRESW	10	—	—	μs	—
Wait time after RES cancellation (at power-on)	LVD0 enabled*1	tRESWT	—	0.506	0.694	ms	—
	LVD0 disabled*2		—	0.201	0.335	ms	—
Wait time after RES cancellation (during powered-on state)	LVD0 enabled*1	tRESWT2	—	0.476	0.616	ms	—
	LVD0 disabled*2		—	0.170	0.257	ms	—
Internal reset by Independent watch dog timer reset, software reset		tRESW2	—	0.04	0.041	ms	—

Note 1. When OFS1.LVDAS = 0.
Note 2. When OFS1.LVDAS = 1.
Note 3. When RES pin is not used as the external reset input, this specification can be ignored.

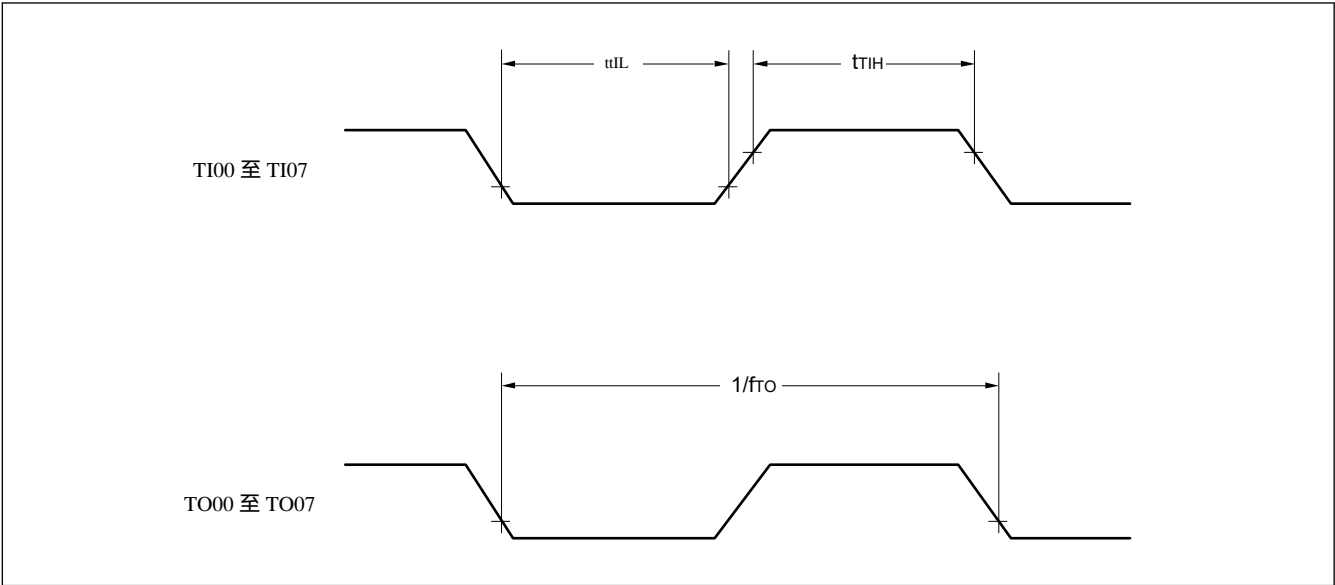


图 27.6TI/TO 时序

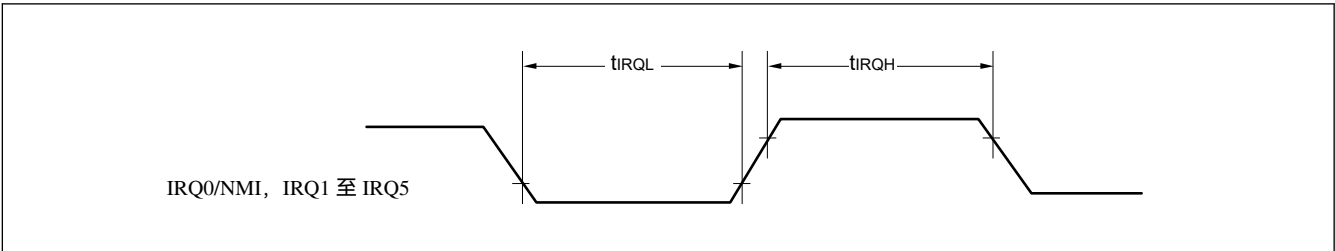


图 27.7 IRQ 中断输入时序

27.4.1 复位时序

表 27.15 复位时序

范围		象征	最小。	类型。	最大限度。	单元	测试条件
RES脉冲宽度	开机时*3	tRESWP	9.9	—	—	ms	—
	未通电	tRESW	10	—	—	μs	—
取消预订后的等待时间 (开机时)	LVD0 已启用*1	tRESWT	—	0.506	0.694	ms	—
	LVD0 已禁用*2		—	0.201	0.335	ms	—
取消预订后的等待时间 (通电状态下)	LVD0 已启用*1	tRESWT2	—	0.476	0.616	ms	—
	LVD0 已禁用*2		—	0.170	0.257	ms	—
内部复位方式包括独立看门狗定时器复位和软件复位		tRESW2	—	0.04	0.041	ms	—

注1. 当 OFS1.LVDAS= 0时。
注2. 当 OFS1.LVDAS= 1时。
注 3. 当 RES 引脚未用作外部复位输入时，可以忽略此规范。

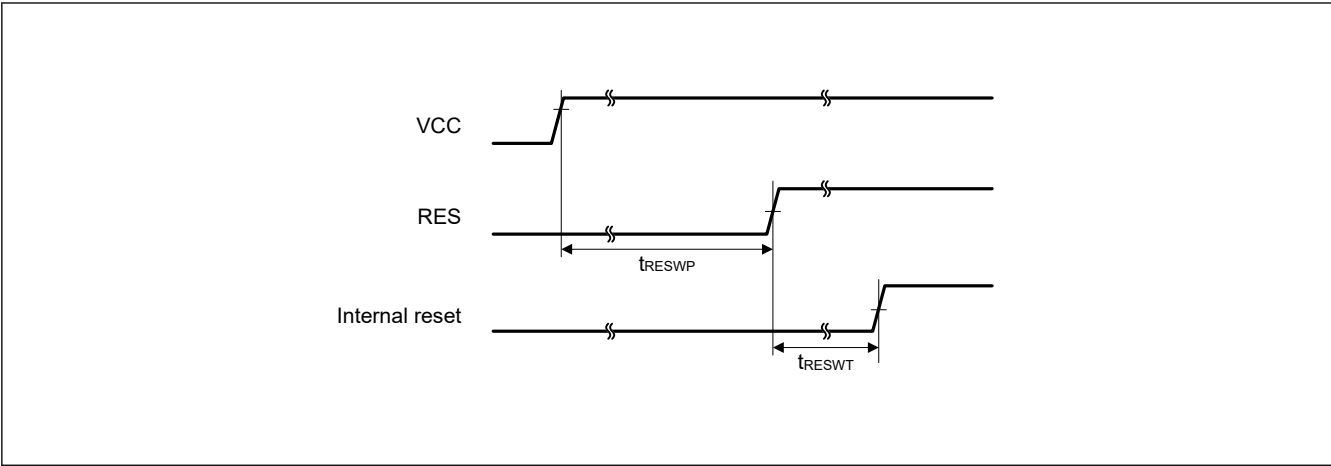


Figure 27.8 Reset input timing at power-on

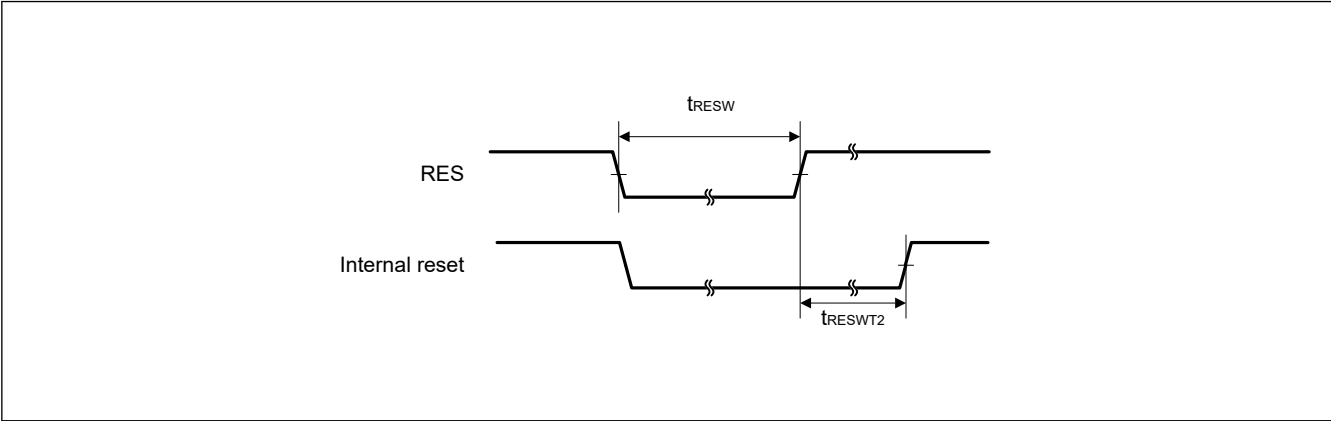


Figure 27.9 Reset input timing (1)

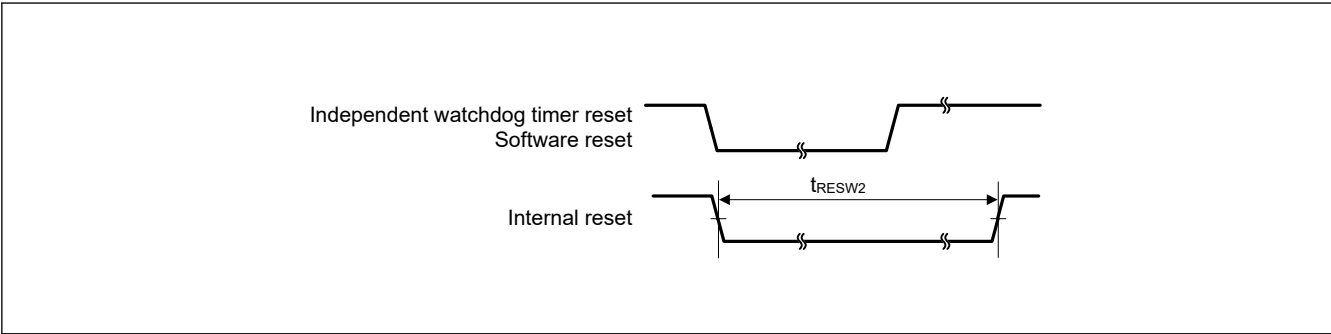


Figure 27.10 Reset input timing (2)

27.4.2 Wakeup Time

Table 27.16 Timing of recovery from low power modes (1)

Parameter				Symbol	Min.	Typ.	Max.	Unit	Test conditions
Recovery time from Software Standby mode*1	High-speed mode	System clock source is HOCO	System clock source is HOCO (32 MHz) VCC = 1.8 V to 5.5 V	t_{SBYHO}	—	5.3	5.7	μs	Figure 27.11
			System clock source is HOCO (4 MHz) VCC = 1.6 V to 1.8 V		—	7.3	7.7	μs	

Note 1. The division ratio of ICLK is the minimum division ratio within the allowable frequency range. The recovery time is determined by the system clock source.

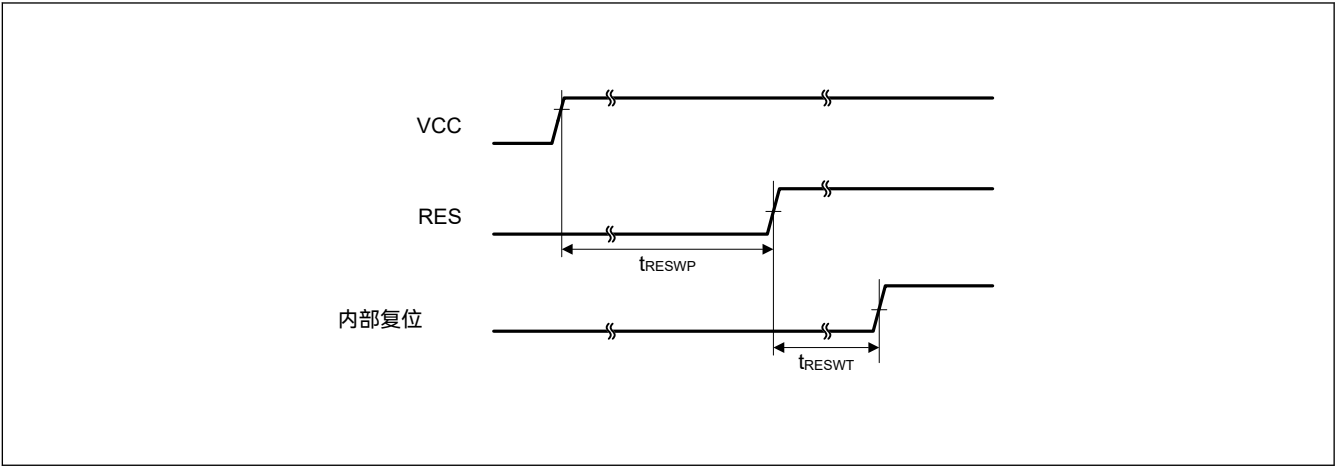


图 27.8 上电时的复位输入时序

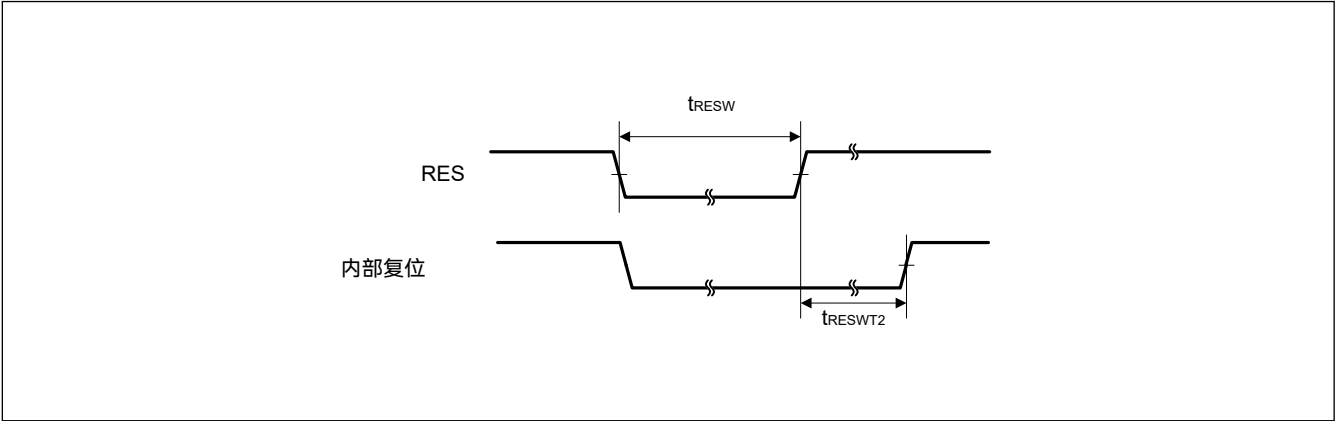


图 27.9 复位输入时序 (1)

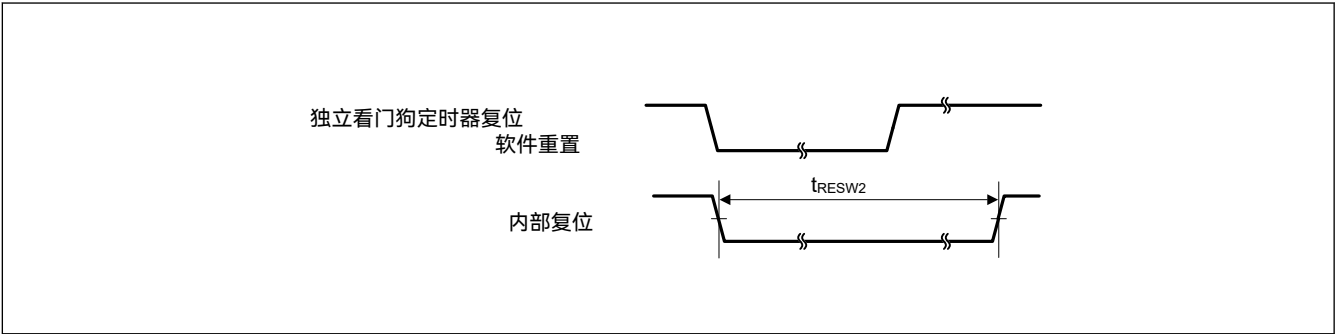


图 27.10 复位输入时序 (2)

27.4.2 起床时间

表 27.16 低功耗模式恢复时间 (1)

范围				象征	最小。	类型。	最大	单元	测试条件
软件待机模式恢复时间*1	高速模式	系统时钟源是 HOCO	系统时钟源为 HOCO (32 MHz) VCC= 10.8 V 至 5.5 V	t_{SBYHO}	—	5.3	限度。	μs	图 27.11
			系统时钟源为 HOCO (4 MHz) VCC = 1.6 V 至 1.8 V		—	7.3	7.7	μs	

注1. ICLK的分频比是允许频率范围内的最小分频比。恢复时间由系统时钟源决定。

Table 27.17 Timing of recovery from low power modes (2)

Parameter				Symbol	Min.	Typ.	Max.	Unit	Test conditions
Recovery time from Software Standby mode*1	Middle-speed mode	System clock source is HOCO	System clock source is HOCO (16 MHz) VCC = 1.8 V to 5.5 V	t _{SBYHO}	—	5.5	5.8	μs	Figure 27.11
			System clock source is HOCO (4 MHz) VCC = 1.6 V to 1.8 V		—	7.3	7.7	μs	

Note 1. The division ratio of ICLK is the minimum division ratio within the allowable frequency range.
The recovery time is determined by the system clock source.

Table 27.18 Timing of recovery from low power modes (3)

Parameter			Symbol	Min.	Typ.	Max.	Unit	Test conditions
Recovery time from Software Standby mode*1	Low-speed mode	System clock source is HOCO (2 MHz)	t _{SBYHO}	—	10.0	10.5	μs	Figure 27.11

Note 1. The division ratio of ICLK is the minimum division ratio within the allowable frequency range.
The recovery time is determined by the system clock source.

Table 27.19 Timing of recovery from low power modes (4)

Parameter			Symbol	Min.	Typ.	Max.	Unit	Test conditions
Recovery time from Software Standby mode*1	Subosc-speed mode	System clock source is LOCO (32.768 kHz)	t _{SBYLO}	—	0.29	0.36	ms	Figure 27.11

Note 1. The LOCO itself continues oscillating in Software Standby mode during Subosc-speed mode.

表 27.17 低功耗模式恢复时间 (2)

范围				象征	最小。	类型。	最大限	单元	测试条件
软件待机模式恢复时间*1	中速模式	系统时钟源为 HOCO	系统时钟源为 HOCO (16 MHz)VCC = 1.8 V至 5.5 V	t _{SBYHO}	—	5.5	5.8	μs	图 27.11
			系统时钟源为 HOCO (4 MHz)VCC = 1.6 V至 1.8 V		—	7.3	7.7	μs	

注1. ICLK的分频比是允许频率范围内的最小分频比。
恢复时间由系统时钟源决定。

表 27.18 低功耗模式恢复时间 (3)

范围				象征	最小。	类型。	最大限	单元	测试条件
软件待机模式恢复时间*1	低速模式	系统时钟源为 HOCO (2 MHz)		t _{SBYHO}	—	10.0	10.5	μs	图 27.11

注1. ICLK的分频比是允许频率范围内的最小分频比。
恢复时间由系统时钟源决定。

表 27.19 低功耗模式恢复时间 (4)

范围				象征	最小。	类型。	最大限	单元	测试条件
软件待机模式恢复时间*1	子振荡器速度模式	系统时钟源为 LOCO (32.768 kHz)		t _{SBYLO}	—	0.29	0.36	ms	图 27.11

注 1. 在 Subosc 速度模式下，LOCO 本身在软件待机模式下继续振荡。

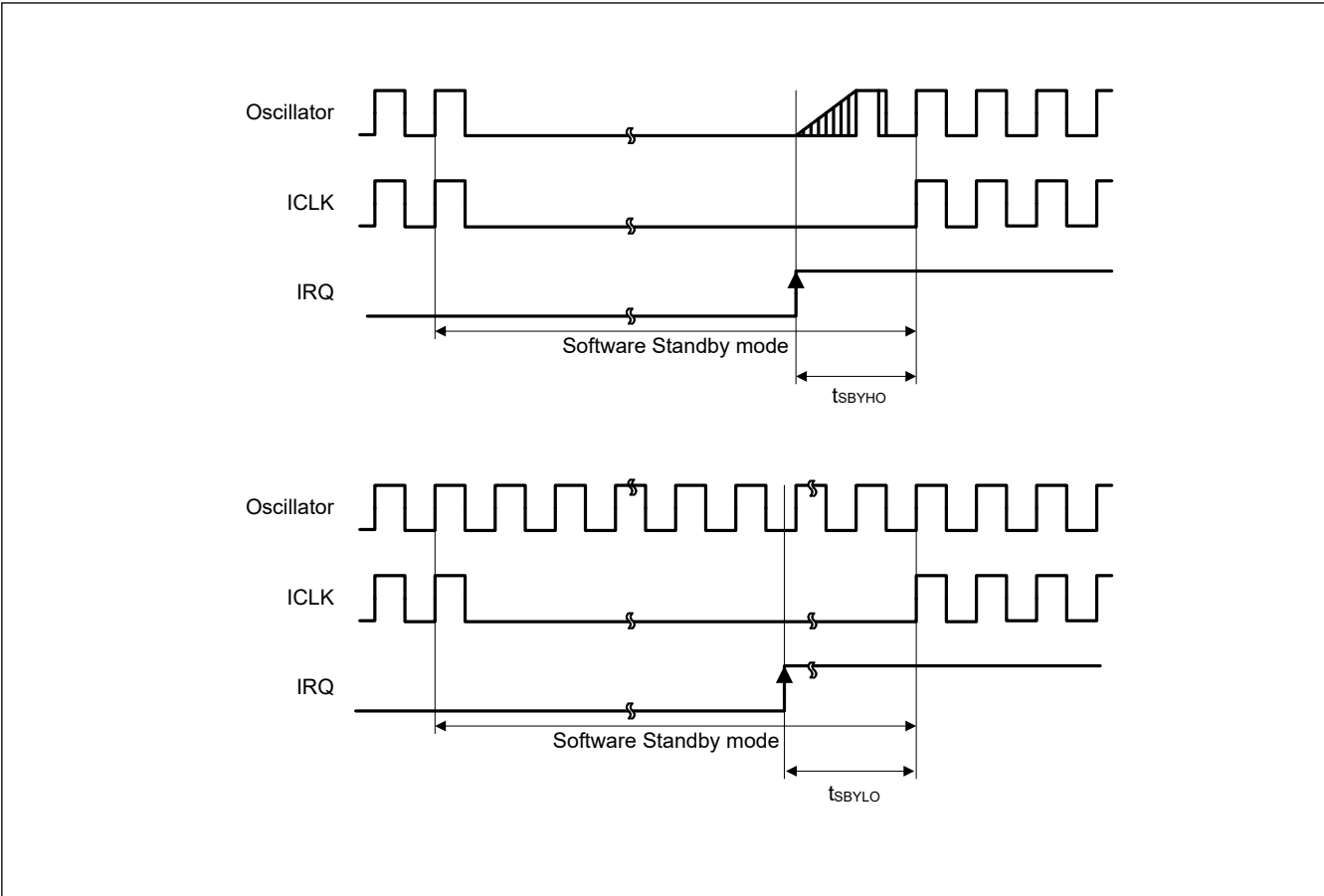


Figure 27.11 Software Standby mode cancellation timing

Table 27.20 Timing of recovery from low power modes (5)

Parameter		Symbol	Min.	Typ.	Max.	Unit	Test conditions
Recovery time from Software Standby mode to Snooze mode	High-speed mode System clock source is HOCO	t _{SNZ}	—	5.3	5.7	μs	Figure 27.12
	Middle-speed mode System clock source is HOCO (16 MHz) VCC = 1.8 V to 5.5 V	t _{SNZ}	—	5.5	5.8	μs	
	Middle-speed mode System clock source is HOCO (4 MHz) VCC = 1.6 V to 1.8 V	t _{SNZ}	—	6.3	6.7	μs	
	Low-speed mode System clock source is HOCO (2 MHz)	t _{SNZ}	—	8.0	8.4	μs	

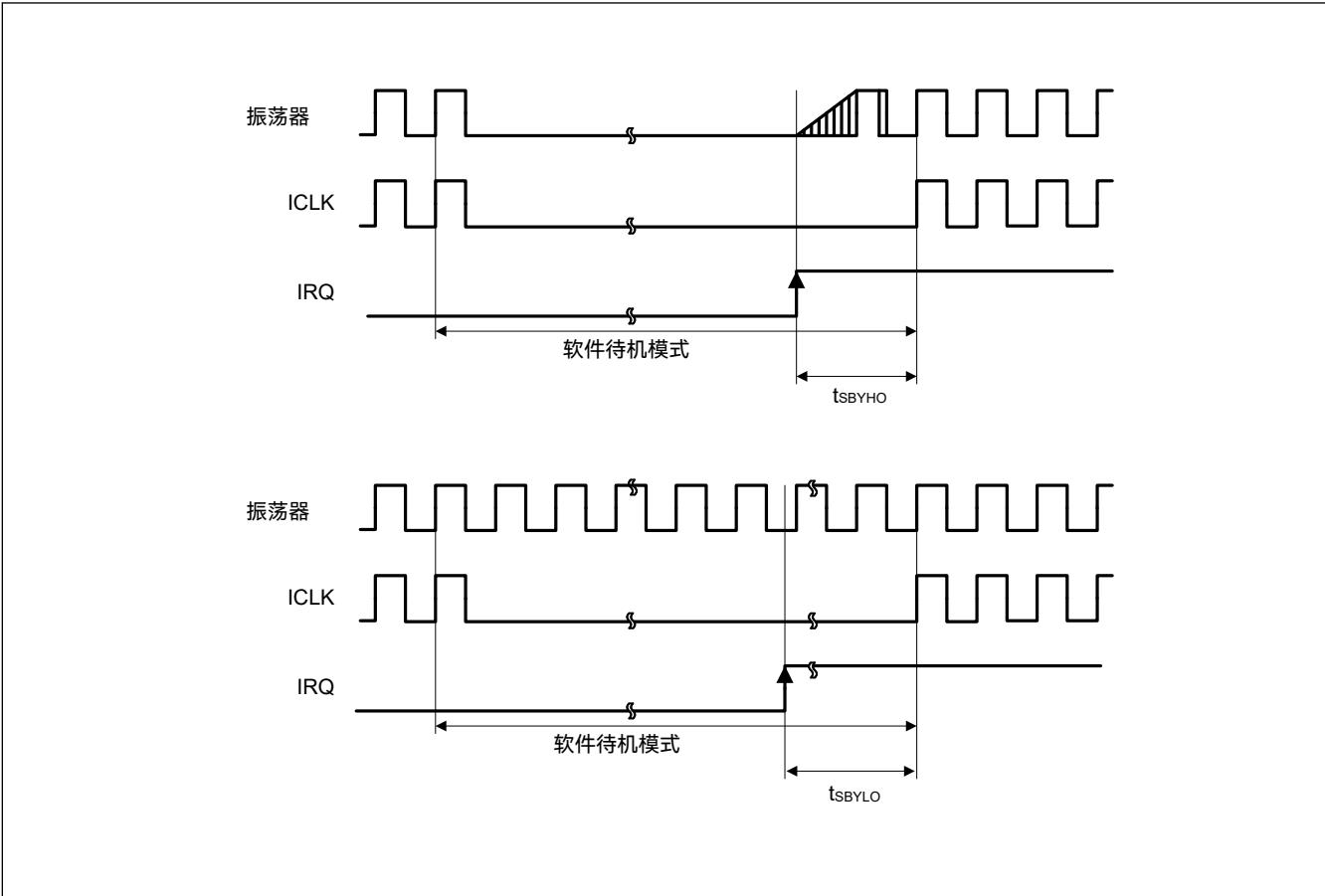


图 27.11 软件待机模式取消时序

表 27.20 低功耗模式恢复时间 (5)

范围		象征	最小。	类型。	最大限度。	单元	测试条件
恢复时间 软件待机模式 贪睡模式	高速模式系统时钟源为 HO CO	t _{SNZ}	—	5.3	5.7	μs	图 27.12
	中速模式 系统时钟源是 HOCO (16 MHz) VCC = 1.8 V to 5.5 V	t _{SNZ}	—	5.5	5.8	μs	
	中速模式 系统时钟源是 HOCO (4 MHz) VCC = 1.6 V to 1.8 V	t _{SNZ}	—	6.3	6.7	μs	
	低速模式 系统时钟源是 HOCO (2 MHz)	t _{SNZ}	—	8.0	8.4	μs	

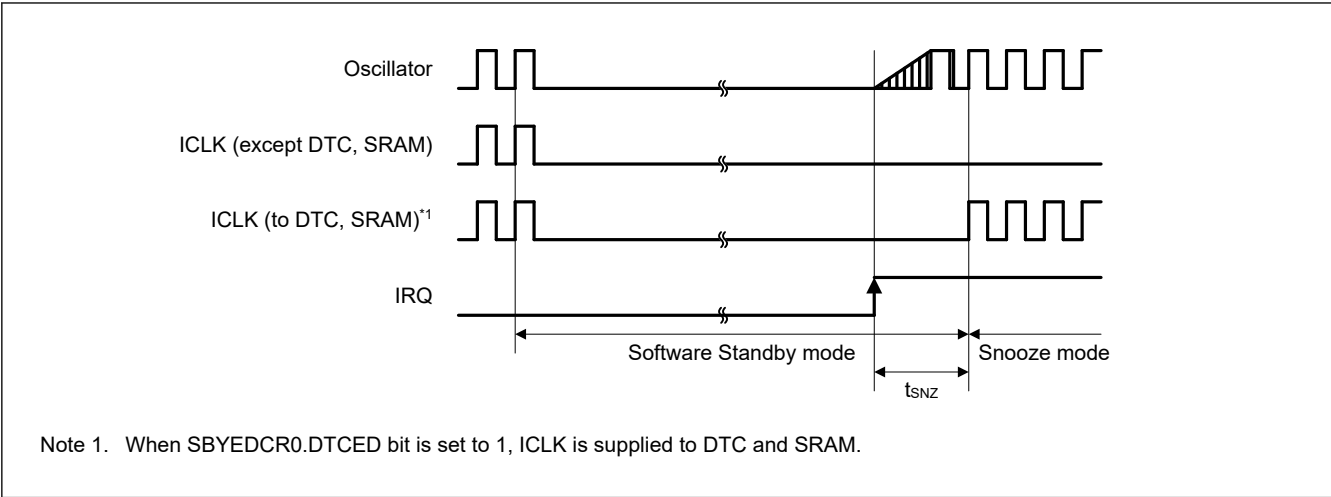


Figure 27.12 Recovery timing from Software Standby mode to Snooze mode

27.5 Peripheral Function Characteristics

27.5.1 Serial Array Unit (SAU)

Table 27.21 In UART communications with devices operating at the same voltage levels

Conditions: VCC = 1.6 to 5.5 V, VSS = 0 V, Ta = -40 to +125°C

Parameter		Symbol	High-speed mode		Middle-speed mode		Low-speed mode		Unit	Test Conditions
			Min.	Max.	Min.	Max.	Min.	Max.		
Transfer rate*1	1.6 V ≤ VCC ≤ 5.5 V	—	—	f _{MCK} /6	—	f _{MCK} /6	—	f _{MCK} /6	bps	Figure 27.14
			—	5.3	—	4	—	0.33	Mbps	

Note 1. The transfer rate in SNOOZE mode is within the range from 4800 to 9600 bps when SBYCR.FWKUP = 0, and within the range from 4800 to 115200 bps when SBYCR.FWKUP = 1.

Note 2. The maximum operating frequencies of the peripheral module clock (PCLKB) are as follows.

High-speed mode: 32 MHz (1.8 V ≤ VCC ≤ 5.5 V), 4 MHz (1.6 V ≤ VCC ≤ 5.5 V)
Middle-speed mode: 16 MHz (1.8 V ≤ VCC ≤ 5.5 V), 4 MHz (1.6 V ≤ VCC ≤ 5.5 V)
Low-speed mode: 2 MHz (1.6 V ≤ VCC ≤ 5.5 V)

Note: Select the normal input buffer for the RXDq pin and the normal output mode for the TXDq pin by using the Port gh Pin Function Select Register (PghPFS_A.PIM and PghPFS_A.NCODR).

gh: Port number (gh = 100, 101, 109, 110, 200, 212, 213, 300)

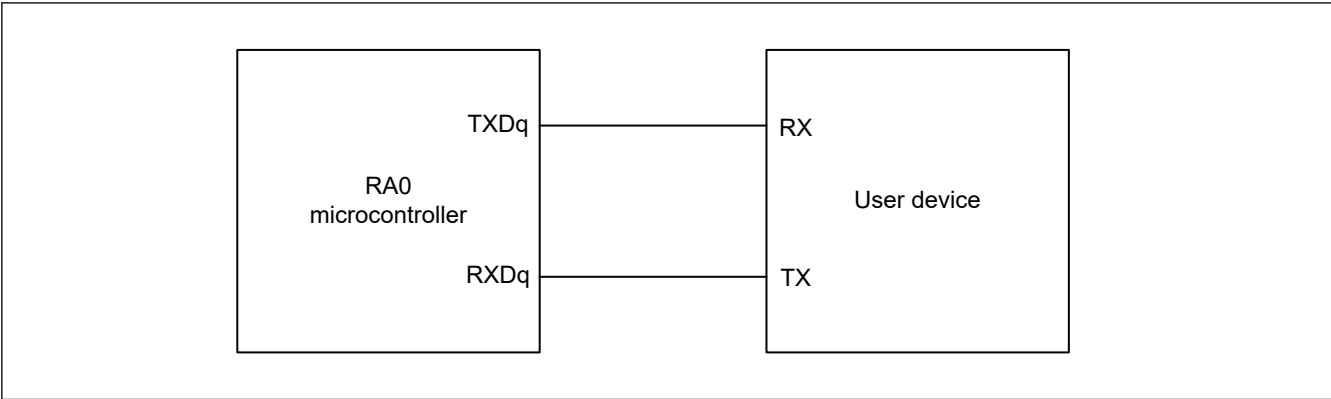


Figure 27.13 Connection in the UART communications with devices operating at the same voltage levels

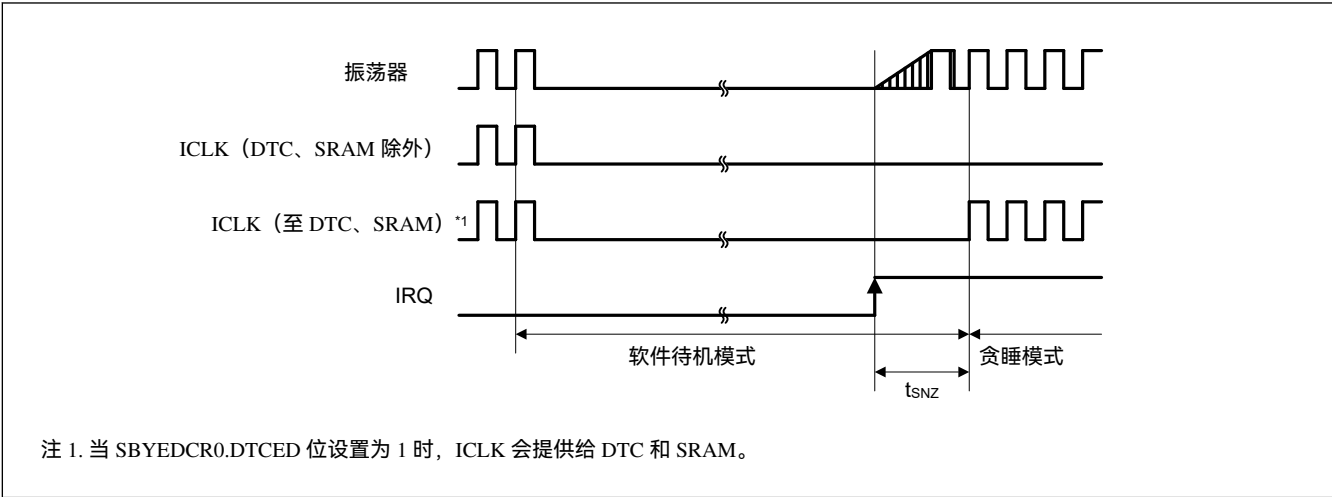


图 27.12 从软件待机模式恢复到休眠模式的时序

27.5 外周功能特征

27.5.1 串行阵列单元 (SAU)

表 27.21 在与工作电压相同的设备进行 UART 通信时

条件: VCC = 1.6至5.5 V, VSS= 0 V, Ta= -40 至+125°C

范围		象征	高速模式		中速模式		低速模式		单元	测试状况
			最小。	最大限	最小。	最大限	最小。	最大限		
转账费率*1	1.6 V ≤ VCC ≤ 5.5 V	—	—	f _{MCK} /6	—	f _{MCK} /6	—	f _{MCK} /6	bps	图 27.14
			—	5.3	—	4	—	0.33	兆比特	

注1. 当 SBYCR.FWKUP= 0, 时, SNOOZE模式下的传输速率在4800到9600 bps范围内; 当 SBYCR.FWKUP= 1时, SNOOZE模式下的传输速率在4800到115200 bps范围内。

注2. 外设模块时钟 (PCLKB) 的最大工作频率如下。

高速模式: 32 MHz (1.8 V ≤ VCC ≤ 5.5 V), 4 MHz (1.6 V ≤ VCC ≤ 5.5 V) 中速模式: 16MHz (1.8 V ≤ VCC ≤ 5.5 V), 4 MHz (1.6 V ≤ VCC ≤ 5.5 V) 低速模式: 2 MHz (1.6 V ≤ VCC ≤ 5.5 V)

注意: 使用端口 gh 引脚功能选择寄存器 (PghPFS_A.PIM 和 PghPFS_A.NCODR) 为 RXDq 引脚选择正常输入缓冲器, 为 TXDq 引脚选择正常输出模式。

gh: 端口号(gh = 100, 101, 109, 110, 200, 212, 213, 300)

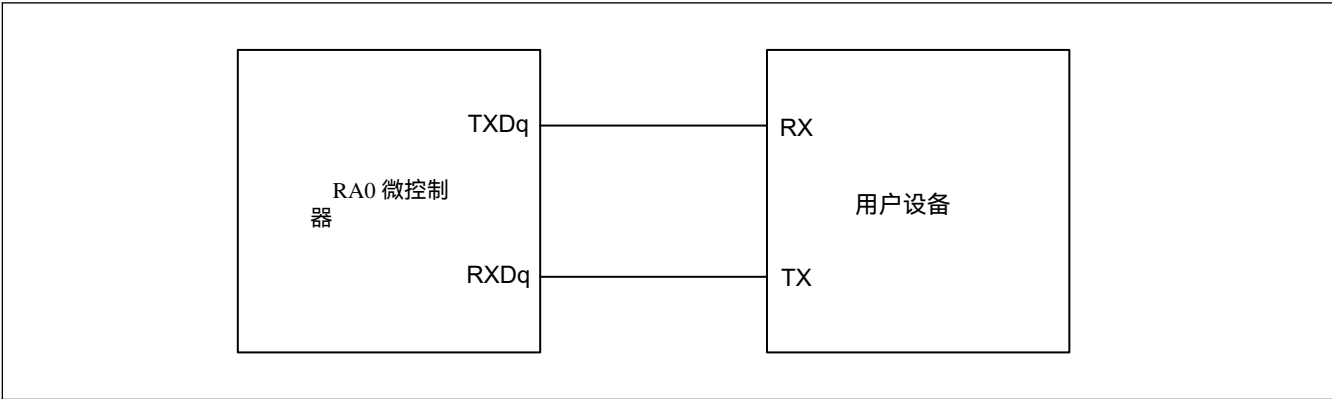


图 27.13 UART 通信中与工作电压相同的设备的连接

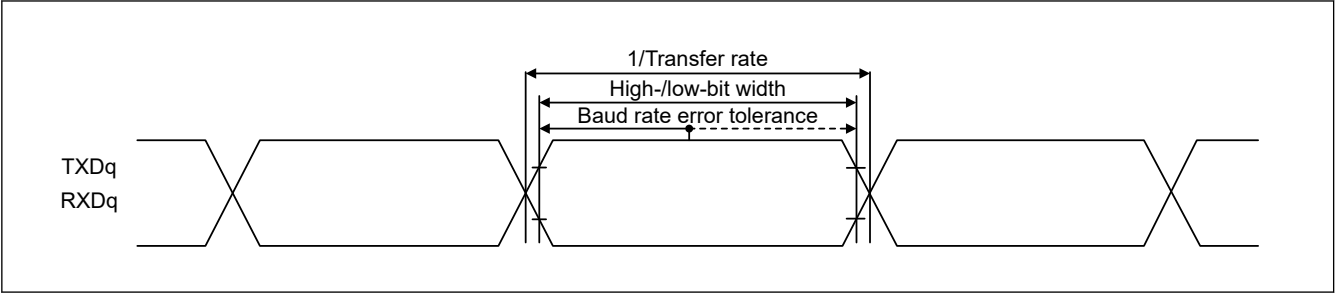


Figure 27.14 Bit width in the UART communications when interfacing devices operate at the same voltage level (reference)

Note:

- q: UART number (q = 0, 1), gh: Port number (gh = 100, 101, 109, 110, 200, 212, 213, 300)
- f_{MCK}: Serial array unit operation clock frequency
To set this operating clock, set the CKS bit in the serial mode register mn (SMRmn).
- m: Unit number, n: Channel number (mn = 00, 01, 02, 03)

Table 27.22 In simplified SPI communications in the master mode with devices operating at the same voltage levels with the internal SCKp clock (the ratings below are only applicable to SPI00)

Conditions: VCC = 2.7 to 5.5 V, VSS = 0 V, Ta = -40 to +85°C

Parameter			Symbol	High-speed mode		Middle-speed mode		Low-speed mode		Unit	Test Conditions
				Min.	Max.	Min.	Max.	Min.	Max.		
SCKp cycle time	t _{KCY1} ≥ 2/ PCLKB	4.0 V ≤ VCC ≤ 5.5 V	t _{KCY1}	62.5	—	125	—	1000	—	ns	Figure 27.16 Figure 27.17
		2.7 V ≤ VCC ≤ 5.5 V		125	—	125	—	1000	—	ns	
SCKp high-/low-level width	4.0 V ≤ VCC ≤ 5.5 V		t _{KH1} , t _{KL1}	t _{KCY1} /2 - 7	—	t _{KCY1} /2 - 10	—	t _{KCY1} /2 - 50	—	ns	
	2.7 V ≤ VCC ≤ 5.5 V			t _{KCY1} /2 - 10	—	t _{KCY1} /2 - 15	—	t _{KCY1} /2 - 50	—	ns	
Slp setup time (to SCKp↑) ^{*1}	4.0 V ≤ VCC ≤ 5.5 V		t _{SIK1}	23	—	33	—	110	—	ns	
	2.7 V ≤ VCC ≤ 5.5 V			33	—	50	—	110	—	ns	
Slp hold time (from SCKp↑) ^{*1}	2.7 V ≤ VCC ≤ 5.5 V		t _{KSI1}	10	—	10	—	10	—	ns	
Delay time from SCKp↓ to SOp output ^{*2}	C = 20 pF ^{*3}		t _{KSO1}	—	10	—	10	—	10	ns	

Note 1. The setting applies when SCRmn.DCP0[1:0] = 00b or 11b. The setting for the Slp setup time becomes to SCKp↓ and that for the Slp hold time becomes from SCKp↓ when SCRmn.DCP0[1:0] = 01b or 10b.

Note 2. This setting applies when SCRmn.DCP0[1:0] = 00b or 11b. The setting for the delay time to SOp output becomes from SCKp↑ when SCRmn.DCP0[1:0] = 01b or 10b.

Note 3. C is the load capacitance of the SCKp and SOp output lines.

Note: Select the normal input buffer for the Slp pin and the normal output mode for the SOp pin and SCKp pin by using the Port gh Pin Function Select Register (PghPFS_A.PIM and PghPFS_A.NCODR).

Note:

- The listed times are only valid when the peripheral I/O redirect function of SPI00 is not in use.
- p: Simplified SPI number (p = 00), m: Unit number (m = 0), n: Channel number (n = 0), gh: Port number (gh = 100 to 102, 108 to 110, 112, 200)
- f_{MCK}: Serial array unit operation clock frequency
To set this operating clock, use the CKS bit in the serial mode register mn (SMRmn).
- m: Unit number, n: Channel number (mn = 00)

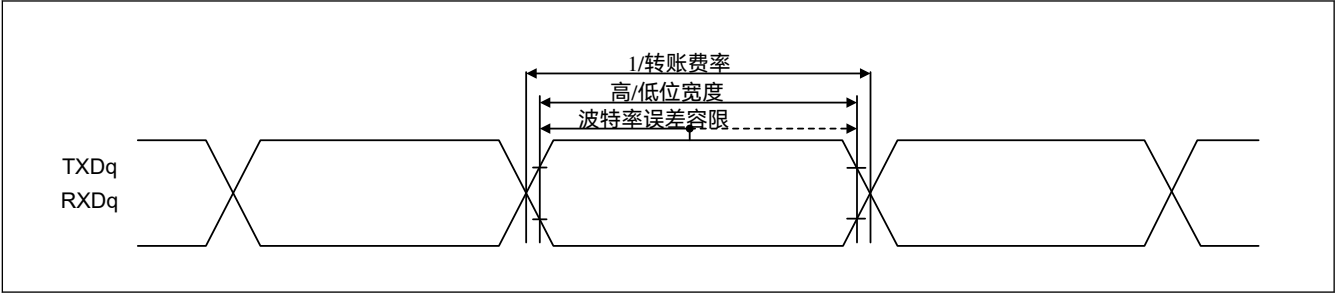


图 27.14 当接口设备工作在相同电压电平（参考）时，UART 通信中的位宽

注: ●q: UART 号(q = 0, 1) gh: 端口号(gh = 100, 101, 109、110、200、212、213、300)

- f_{MCK}: 串行阵列单元操作时钟频率
要设置此工作时钟，请设置串行模式寄存器 mn (SMRmn) 中的 CKS 位。
- m: 单元编号，n: 通道编号(mn = 00, 01, 02, 03)

表 27.22 在主模式下，简化的 SPI 通信中，设备在与内部 SCKp 时钟相同的电压电平下工作（以下额定值仅适用于 SPI00）

条件: VCC = 2.7 至 5.5 V, VSS = 0 V, Ta = -40 至 +85°C

范围			象征	高速模式		中速模式		低速模式		单元	测试状况
				最小。	最大限	最小。	最大限	最小。	最大限		
SCKp 循环时间	t _{KCY1} ≥ 2/ PCLKB	4.0 V ≤ VCC ≤ 5.5 V	t _{KCY1}	62.5	—	125	—	1000	—	ns	图 27.16 图 27.17
		2.7 V ≤ VCC ≤ 5.5 V	125	—	125	—	1000	—	ns		
SCKp 高/低电平宽度	4.0 V ≤ VCC ≤ 5.5 V		t _{KH1} , t _{KL1}	t _{KCY1} /2 - 7	—	t _{KCY1} /2 - 10	—	t _{KCY1} /2 - 50	—	ns	
	2.7 V ≤ VCC ≤ 5.5 V			t _{KCY1} /2 - 10	—	t _{KCY1} /2 - 15	—	t _{KCY1} /2 - 50	—	ns	
Slp 建立时间 (到 SCKp↑) *1	4.0 V ≤ VCC ≤ 5.5 V		t _{SIK1}	23	—	33	—	110	—	ns	
	2.7 V ≤ VCC ≤ 5.5 V			33	—	50	—	110	—	ns	
Slp 保持时间 (来自SCKp↑)*1	2.7 V ≤ VCC ≤ 5.5 V		t _{KSH1}	10	—	10	—	10	—	ns	
从 SCKp↓ 到SO p输出的延迟时间 *2	C = 20 pF ¹³		t _{KSO1}	—	10	—	10	—	10	ns	

注 1. 当SCRmn.DCP0[1:0] = 00b 或 11b 时，此设置适用。Slp 建立时间的设置变为SCKp↓，Slp 保持时间的设置变为当 SCRmn.DCP0[1:0] = 01b 或 10b 时，SCKp↓。

注 2. 此设置适用于 SCRmn.DCP0[1:0] = 00b 或 11b。当SCRmn.DCP0[1:0] = 01b 或 10b 时，延迟时间到SOp输出的设置将变为SCKp↑。

注 3. C 是SCKp和SOp输出线的负载电容。

注意: 使用端口 gh 引脚功能选择寄存器 (PghPFS_A.PIM 和 PghPFS_A.NCODR) 为Slp引脚选择正常输入缓冲模式，为SOp引脚和 SCKp 引脚选择正常输出模式。

注意: ●所列时间仅在未使用外设SPI00重定向功能时有效。

- p: 简化的 SPI 编号(p = 00), m: 单元编号(m = 0) n: 通道编号(n = 0), gh: 端口编号(gh = (100 至 102、108 至 110、112、200)
- f_{MCK}: 串行阵列单元操作时钟频率
要设置此工作时钟，请使用串行模式寄存器 mn (SMRmn) 中的 CKS 位。
- m: 单元编号，n: 通道编号(mn = 00)

Table 27.23 In simplified SPI communications in the master mode with devices operating at the same voltage levels with the internal SCKp clock

Conditions: VCC = 1.6 to 5.5 V, VSS = 0 V, Ta = -40 to +125°C

Parameter			Symbol	High-speed mode		Middle-speed mode		Low-speed mode		Unit	Test Conditions
				Min.	Max.	Min.	Max.	Min.	Max.		
SCKp cycle time	t _{KCY1} ≥ 4/ PCLKB	2.7 V ≤ VCC ≤ 5.5 V	t _{KCY1}	125	—	166	—	2000	—	ns	Figure 27.16 Figure 27.17
		2.4 V ≤ VCC ≤ 5.5 V		250	—	250	—	2000	—	ns	
		1.8 V ≤ VCC ≤ 5.5 V		500	—	500	—	2000	—	ns	
		1.6 V ≤ VCC ≤ 5.5 V		1000	—	1000	—	2000	—	ns	
SCKp high-/low-level width	4.0 V ≤ VCC ≤ 5.5 V		t _{KH1} , t _{KL1}	t _{KCY1} /2 - 12	—	t _{KCY1} /2 - 21	—	t _{KCY1} /2 - 50	—	ns	
	2.7 V ≤ VCC ≤ 5.5 V			t _{KCY1} /2 - 18	—	t _{KCY1} /2 - 25	—	t _{KCY1} /2 - 50	—	ns	
	2.4 V ≤ VCC ≤ 5.5 V			t _{KCY1} /2 - 38	—	t _{KCY1} /2 - 38	—	t _{KCY1} /2 - 50	—	ns	
	1.8 V ≤ VCC ≤ 5.5 V			t _{KCY1} /2 - 50	—	t _{KCY1} /2 - 50	—	t _{KCY1} /2 - 50	—	ns	
	1.6 V ≤ VCC ≤ 5.5 V			t _{KCY1} /2 - 100	—	t _{KCY1} /2 - 100	—	t _{KCY1} /2 - 100	—	ns	
Slp setup time (to SCKp↑) ¹	4.0 V ≤ VCC ≤ 5.5 V		t _{SIK1}	44	—	54	—	110	—	ns	
	2.7 V ≤ VCC ≤ 5.5 V			44	—	54	—	110	—	ns	
	2.4 V ≤ VCC ≤ 5.5 V			75	—	75	—	110	—	ns	
	1.8 V ≤ VCC ≤ 5.5 V			110	—	110	—	110	—	ns	
	1.6 V ≤ VCC ≤ 5.5 V			220	—	220	—	220	—	ns	
Slp hold time (from SCKp↑) ¹	1.6 V ≤ VCC ≤ 5.5 V		t _{KSH1}	19	—	19	—	19	—	ns	
Delay time from SCKp↓ to SOp output ²	1.6 V ≤ VCC ≤ 5.5 V C = 30 pF ³		t _{KSO1}	—	25	—	25	—	25	ns	

Note 1. This setting applies when SCRmn.DCP[1:0] = 00b or 11b. The setting for the Slp setup time becomes to SCKp↓ and that for the Slp hold time becomes from SCKp↓ when SCRmn.DCP[1:0] = 01b or 10b.

Note 2. This setting applies when SCRmn.DCP[1:0] = 00b or 11b. The setting for the delay time to SOp output becomes from SCKp↑ when SCRmn.DCP[1:0] = 01b or 10b.

Note 3. C is the load capacitance of the SCKp and SOp output lines.

Note: Select the normal input buffer for the Slp pin and the normal output mode for the SOp pin and SCKp pin by using the Port gh Pin Function Select Register (PghPFS_A.PIM and PghPFS_A.NCODR).

- Note:
- p: Simplified SPI number (p = 00, 11), m: Unit number (m = 0), n: Channel number (n = 0, 3), gh: Port number (gh = 100 to 102, 108 to 110, 112, 200, 201, 212, 213)
 - fMCK: Serial array unit operation clock frequency
To set this operating clock, use the CKS bit in the serial mode register mn (SMRmn).
m: Unit number, n: Channel number (mn = 00, 03)

表 27.23 简化的主模式下 SPI 通信，设备在与内部 SCKp 时钟相同的电压电平下工作。

条件: VCC= 1.6至 5.5 V, VSS= 0 V,Ta= -40至+125°C

范围			象征	高速模式		中速模式		低速模式		单元	测试 状况
				最小。	最大 限度。	最小。	最大 限度。	最小。	最大 限度。		
SCK _p 循环时间	t _{KCY1} ≥ 4/ PCLKB	2.7 V ≤ VCC ≤ 5.5 V	t _{KCY1}	125	—	166	—	2000	—	ns	图 27.16 图 27.17
		2.4 V ≤ VCC ≤ 5.5 V		250	—	250	—	2000	—	ns	
		1.8 V ≤ VCC ≤ 5.5 V		500	—	500	—	2000	—	ns	
		1.6 V ≤ VCC ≤ 5.5 V		1000	—	1000	—	2000	—	ns	
SCK _p 高/低水平 宽度	4.0 V ≤ VCC ≤ 5.5 V	t _{KH1} , t _{KL1}	t _{KCY1} /2 - 12	—	t _{KCY1} /2 - 21	—	t _{KCY1} /2 - 50	—	ns		
	2.7 V ≤ VCC ≤ 5.5 V		t _{KCY1} /2 - 18	—	t _{KCY1} /2 - 25	—	t _{KCY1} /2 - 50	—	ns		
	2.4 V ≤ VCC ≤ 5.5 V		t _{KCY1} /2 - 38	—	t _{KCY1} /2 - 38	—	t _{KCY1} /2 - 50	—	ns		
	1.8 V ≤ VCC ≤ 5.5 V		t _{KCY1} /2 - 50	—	t _{KCY1} /2 - 50	—	t _{KCY1} /2 - 50	—	ns		
	1.6 V ≤ VCC ≤ 5.5 V		t _{KCY1} /2 - 100	—	t _{KCY1} /2 - 100	—	t _{KCY1} /2 - 100	—	ns		
Slp 设置时间 (至 SCK _p ↑) *1	4.0 V ≤ VCC ≤ 5.5 V	t _{SIK1}	44	—	54	—	110	—	ns		
	2.7 V ≤ VCC ≤ 5.5 V		44	—	54	—	110	—	ns		
	2.4 V ≤ VCC ≤ 5.5 V		75	—	75	—	110	—	ns		
	1.8 V ≤ VCC ≤ 5.5 V		110	—	110	—	110	—	ns		
	1.6 V ≤ VCC ≤ 5.5 V		220	—	220	—	220	—	ns		
Slp 保持时间 (来自 SCK _p ↑) *1	1.6 V ≤ VCC ≤ 5.5 V	t _{KSH1}	19	—	19	—	19	—	ns		
从 SCK _p ↓ 到 SOp 输出的 延迟时间*2	1.6 V ≤ VCC ≤ 5.5 V C = 30 pF ³	t _{KSO1}	—	25	—	25	—	25	ns		

注 1. 此设置适用于 SCRmn.DCP[1:0] = 00b 或 11b。当 SCRmn.DCP[1:0] = 01b 或 10b 时，Slp 设置时间变为 SCKp↓，Slp 保持时间变为 SCKp↓。

注 2. 此设置适用于 SCRmn.DCP[1:0] = 00b 或 11b。当 SCRmn.DCP[1:0] = 01b 或 10b 时，延迟到 SOp 输出的设置将变为 SCKp↑。

注 3. C 是 SCKp 和 SOp 输出线的负载电容。

注意: 使用端口 gh 引脚功能选择寄存器 (PghPFS_A.PIM 和 PghPFS_A.NCODR) 为 Slp 引脚选择正常输入缓冲模式，为 SOp 引脚和 SCKp 引脚选择正常输出模式。

注: ●p: 简化的 SPI 编号(p = 00, 11) , m: 单元编号(m = 0), n: 通道编号(n = 0, 3) gh: 端口编号(gh = 100至 102、108 至 110、112、200、201、212、213)

- fMCK: 串行阵列单元操作时钟频率
要设置此工作时钟，请使用串行模式寄存器 mn (SMRmn) 中的 CKS 位。
m: 单元编号，n: 通道编号(mn = 00, 03)

Table 27.24 In simplified SPI communications in the slave mode with devices operating at the same voltage levels with the SCKp external clock

Conditions: VCC = 1.6 to 5.5 V, VSS = 0 V, Ta = -40 to +125°C

Item	Conditions		Symbol	High-speed mode		Middle-speed mode		Low-speed mode		Unit	Test Conditions
				Min.	Max.	Min.	Max.	Min.	Max.		
SCKp cycle time ^{*4}	4.0 V ≤ VCC ≤ 5.5 V	20 MHz < f _{MCK}	t _{KCY2}	8/f _{MCK}	—	8/f _{MCK}	—	—	—	ns	Figure 27.16 Figure 27.17
		f _{MCK} ≤ 20 MHz		6/f _{MCK}	—	6/f _{MCK}	—	6/f _{MCK}	—	ns	
	2.7 V ≤ VCC ≤ 5.5 V	16 MHz < f _{MCK}		8/f _{MCK}	—	8/f _{MCK}	—	—	—	ns	
		f _{MCK} ≤ 16 MHz		6/f _{MCK}	—	6/f _{MCK}	—	6/f _{MCK}	—	ns	
	2.4 V ≤ VCC ≤ 5.5 V			Greater of: 6/f _{MCK} or 500	—	Greater of: 6/f _{MCK} or 500	—	Greater of: 6/f _{MCK} or 500	—	ns	
	1.8 V ≤ VCC ≤ 5.5 V			Greater of: 6/f _{MCK} or 750	—	Greater of: 6/f _{MCK} or 750	—	Greater of: 6/f _{MCK} or 750	—	ns	
	1.6 V ≤ VCC ≤ 5.5 V			Greater of: 6/f _{MCK} or 1500	—	Greater of: 6/f _{MCK} or 1500	—	Greater of: 6/f _{MCK} or 1500	—	ns	
SCKp high-/low-level width	4.0 V ≤ VCC ≤ 5.5 V		t _{KH2} , t _{KL2}	t _{KCY2} /2 - 7	—	t _{KCY2} /2 - 7	—	t _{KCY2} /2 - 7	—	ns	
	2.7 V ≤ VCC ≤ 5.5 V			t _{KCY2} /2 - 8	—	t _{KCY2} /2 - 8	—	t _{KCY2} /2 - 8	—	ns	
	1.8 V ≤ VCC ≤ 5.5 V			t _{KCY2} /2 - 18	—	t _{KCY2} /2 - 18	—	t _{KCY2} /2 - 18	—	ns	
	1.6 V ≤ VCC ≤ 5.5 V			t _{KCY2} /2 - 66	—	t _{KCY2} /2 - 66	—	t _{KCY2} /2 - 66	—	ns	
Slp setup time (to SCKp↑) ^{*1}	2.7 V ≤ VCC ≤ 5.5 V		t _{SIK2}	1/f _{MCK} + 20	—	1/f _{MCK} + 30	—	1/f _{MCK} + 30	—	ns	
	1.8 V ≤ VCC ≤ 5.5 V			1/f _{MCK} + 30	—	1/f _{MCK} + 30	—	1/f _{MCK} + 30	—	ns	
	1.6 V ≤ VCC ≤ 5.5 V			1/f _{MCK} + 40	—	1/f _{MCK} + 40	—	1/f _{MCK} + 40	—	ns	
Slp hold time (from SCKp↑) ^{*1}	1.8 V ≤ VCC ≤ 5.5 V		t _{KSI2}	1/f _{MCK} + 31	—	1/f _{MCK} + 31	—	1/f _{MCK} + 31	—	ns	
	1.6 V ≤ VCC ≤ 5.5 V			1/f _{MCK} + 250	—	1/f _{MCK} + 250	—	1/f _{MCK} + 250	—	ns	
Delay time from SCKp↓ to SOp output ^{*2}	C = 30 pF ^{*3}	2.7 V ≤ VCC ≤ 5.5 V	t _{KSO2}	—	2/f _{MCK} + 44	—	2/f _{MCK} + 110	—	2/f _{MCK} + 110	ns	
		2.4 V ≤ VCC ≤ 5.5 V		—	2/f _{MCK} + 75	—	2/f _{MCK} + 110	—	2/f _{MCK} + 110	ns	
		1.8 V ≤ VCC ≤ 5.5 V		—	2/f _{MCK} + 110	—	2/f _{MCK} + 110	—	2/f _{MCK} + 110	ns	
		1.6 V ≤ VCC ≤ 5.5 V		—	2/f _{MCK} + 220	—	2/f _{MCK} + 220	—	2/f _{MCK} + 220	ns	

- Note 1. This setting applies when SCRmn.DCP[1:0] = 00b or 11b. The setting for the Slp setup time becomes to SCKp↓ and that for the Slp hold time becomes from SCKp↓ when SCRmn.DCP[1:0] = 01b or 10b.
- Note 2. This setting applies when SCRmn.DCP[1:0] = 00b or 11b. The setting for the delay time to SOp output becomes from SCKp↑ when SCRmn.DCP[1:0] = 01b or 10b.
- Note 3. C is the load capacitance of the SOp output line.
- Note 4. Transfer rate in the Snooze mode is 1 Mbps at the maximum.

Note: Select the normal input buffer for the Slp pin and SCKp pin and the normal output mode for the SOp pin by using the Port gh Pin Function Select Register (PghPFS_A.PIM and PghPFS_A.NCODR).

- Note:
- p: Simplified SPI number (p = 00, 11), m: Unit number (m = 0, 1), n: Channel number (n = 0, 3), gh: Port number (gh = 100 to 102, 108 to 110, 112, 200, 201, 212, 213)
 - fMCK: Serial array unit operation clock frequency
To set this operating clock, use the CKS bit in the serial mode register mn (SMRmn).
m: Unit number, n: Channel number (mn = 00, 03)

表 27.24 在简化的 SPI 从模式通信中，当设备工作电压与外部时钟SCKp相同时，通信方式如下：

条件: VCC= 1.6至 5.5 V, VSS= 0 V,Ta= -40至+125°C

物品	状况		象征	高速模式		中速模式		低速模式		单元	测试状况
				最小。	最大限度。	最小。	最大限度。	最小。	最大限度。		
SCKp 循环时间*4	4.0 V ≤ VCC ≤ 5.5 V	20 MHz < fMCK	tKCY2	8/fMCK	—	8/fMCK	—	—	—	ns	数字 27.16 数字 27.17
		fMCK ≤ 20 MHz		6/fMCK	—	6/fMCK	—	6/fMCK	—	ns	
	2.7 V ≤ VCC ≤ 5.5 V	16 MHz < fMCK		8/fMCK	—	8/fMCK	—	—	—	ns	
		fMCK ≤ 16 MHz		6/fMCK	—	6/fMCK	—	6/fMCK	—	ns	
	2.4 V ≤ VCC ≤ 5.5 V			以下两者中的较大值: 6/fMCK 或 500		取较大值: 6/fMCK 或 500	—	以下两者中的较大值: 6/fMCK 或 500		ns	
	1.8 V ≤ VCC ≤ 5.5 V			以下两者中的较大值: 6/fMCK 或 750		以下两者中的较大值: 6/fMCK 或 750	—	以下两者中的较大值: 6/fMCK 或 750		ns	
	1.6 V ≤ VCC ≤ 5.5 V			以下两者中的较大值: 6/fMCK 或 1500		以下两者中的较大值: 6/fMCK 或 1500	—	以下两者中的较大值: 6/fMCK 或 1500		ns	
SCKp 高/低电平宽度	4.0 V ≤ VCC ≤ 5.5 V		tKH2, tKL2	tKCY2/2 - 7	—	tKCY2/2 - 7	—	tKCY2/2 - 7	—	ns	
	2.7 V ≤ VCC ≤ 5.5 V			tKCY2/2 - 8	—	tKCY2/2 - 8	—	tKCY2/2 - 8	—	ns	
	1.8 V ≤ VCC ≤ 5.5 V			tKCY2/2 - 18	—	tKCY2/2 - 18	—	tKCY2/2 - 18	—	ns	
	1.6 V ≤ VCC ≤ 5.5 V			tKCY2/2 - 66	—	tKCY2/2 - 66	—	tKCY2/2 - 66	—	ns	
Slp 设置时间 (至 SCKp↑) *1	2.7 V ≤ VCC ≤ 5.5 V		tSIK2	1/fMCK + 20	—	1/fMCK + 30	—	1/fMCK + 30	—	ns	
	1.8 V ≤ VCC ≤ 5.5 V			1/fMCK + 30	—	1/fMCK + 30	—	1/fMCK + 30	—	ns	
	1.6 V ≤ VCC ≤ 5.5 V			1/fMCK + 40	—	1/fMCK + 40	—	1/fMCK + 40	—	ns	
啜饮保持时间 (来自SCKp↑)*1	1.8 V ≤ VCC ≤ 5.5 V		tKSI2	1/fMCK + 31	—	1/fMCK + 31	—	1/fMCK + 31	—	ns	
	1.6 V ≤ VCC ≤ 5.5 V			1/fMCK + 250	—	1/fMCK + 250	—	1/fMCK + 250	—	ns	
从 SCKp↓ 到 SOp 输出的延迟时间*2	C = 30 pF*3	2.7 V ≤ VCC ≤ 5.5 V	tKSO2	—	2/fMCK + 44	—	2/fMCK + 110	—	2/fMCK + 110	ns	
		2.4 V ≤ VCC ≤ 5.5 V		—	2/fMCK + 75	—	2/fMCK + 110	—	2/fMCK + 110	ns	
		1.8 V ≤ VCC ≤ 5.5 V		—	2/fMCK + 110	—	2/fMCK + 110	—	2/fMCK + 110	ns	
		1.6 V ≤ VCC ≤ 5.5 V		—	2/fMCK + 220	—	2/fMCK + 220	—	2/fMCK + 220	ns	

- 注 1. 此设置适用于 SCRmn.DCP[1:0] = 00b 或 11b。当 SCRmn.DCP[1:0] = 01b 或 10b 时，Slp 设置时间变为 SCKp↓，Slp 保持时间变为 SCKp↓。
- 注 2. 此设置适用于 SCRmn.DCP[1:0] = 00b 或 11b。当 SCRmn.DCP[1:0] = 01b 或 10b 时，延迟到 SOp 输出的设置将来自 SCKp↑。
- 注3. C是SOp 输出线的负载电容。
- 注 4. 休眠模式下的最大传输速率为 1 Mbps。

注意: 使用端口 gh 引脚功能选择寄存器 (PghPFS_A.PIM 和 PghPFS_A.NCODR) 为 Slp 引脚和 SCKp 引脚选择正常输入缓冲模式，为 SOp 引脚选择正常输出模式。

- 注: ●p: 简化的SPI编号(p = 00, 11) , m: 单元编号(m = 0, 1) , n: 通道编号(n = 0, 3) gh: 端口编号(gh = 100至102、108至110、112、200、201、212、213)
- fMCK: 串行阵列单元操作时钟频率
要设置此工作时钟，请使用串行模式寄存器 mn (SMRmn) 中的 CKS 位。
m: 单元编号, n: 通道编号(mn = 00, 03)

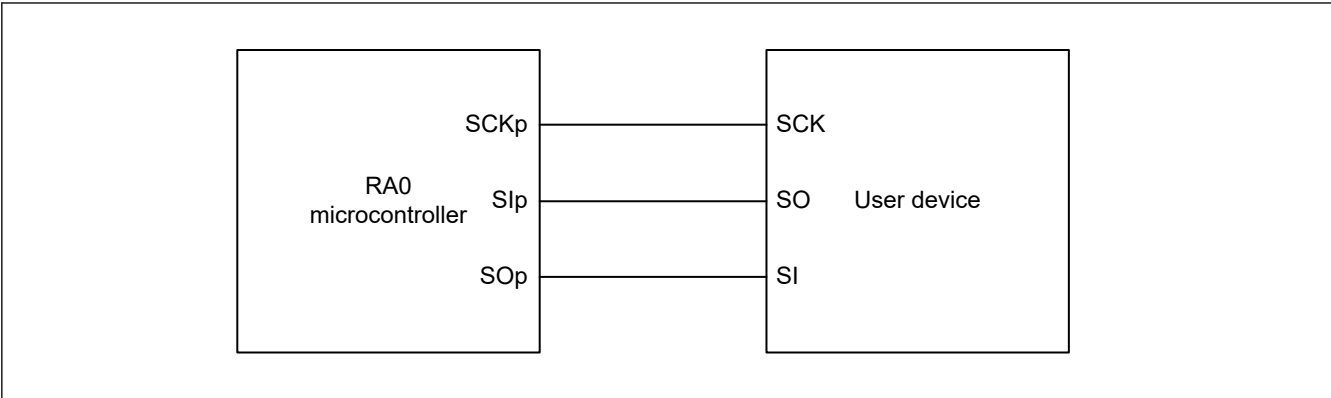


Figure 27.15 Connection in the simplified SPI communications with devices operating at the same voltage levels

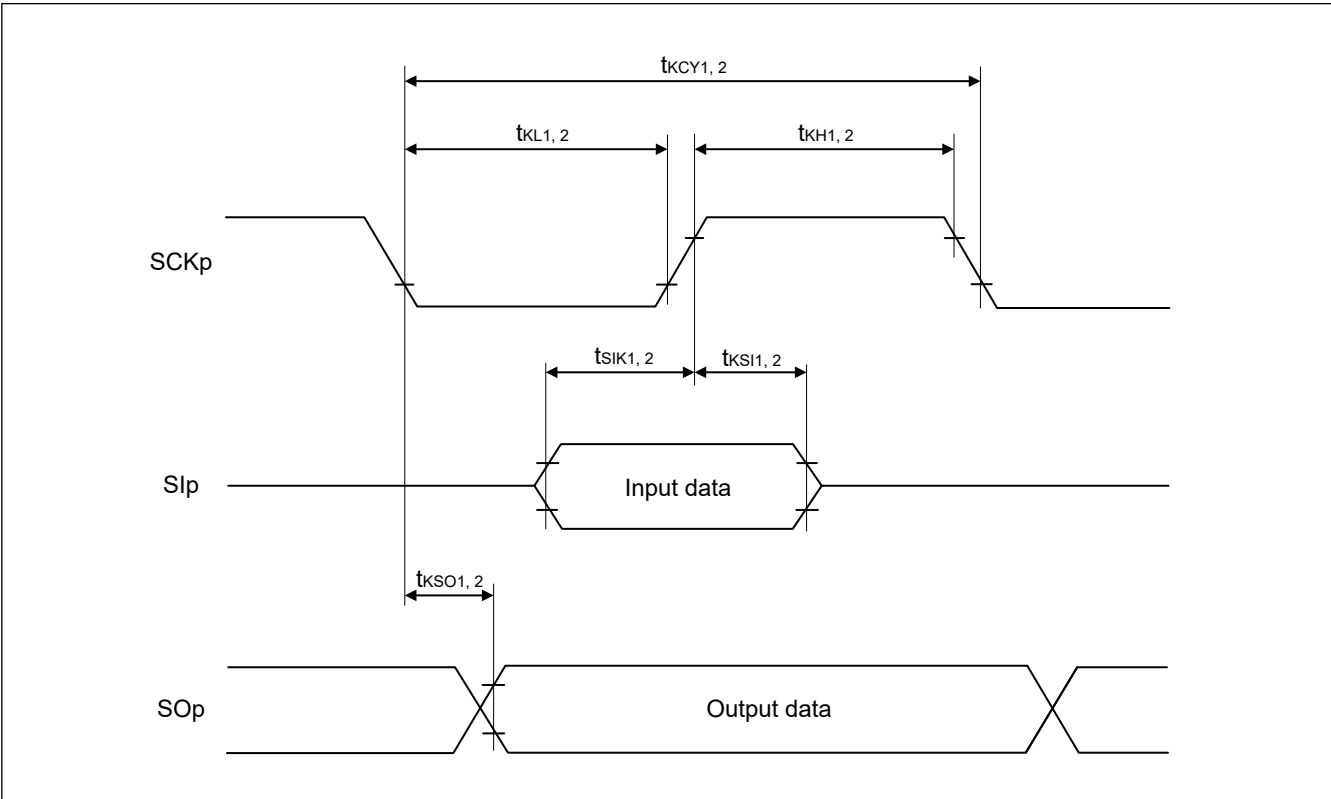


Figure 27.16 Timing of serial transfer in the simplified SPI communications with devices operating at the same voltage levels when SCRmn.DCP[1:0] = 00b or 11b

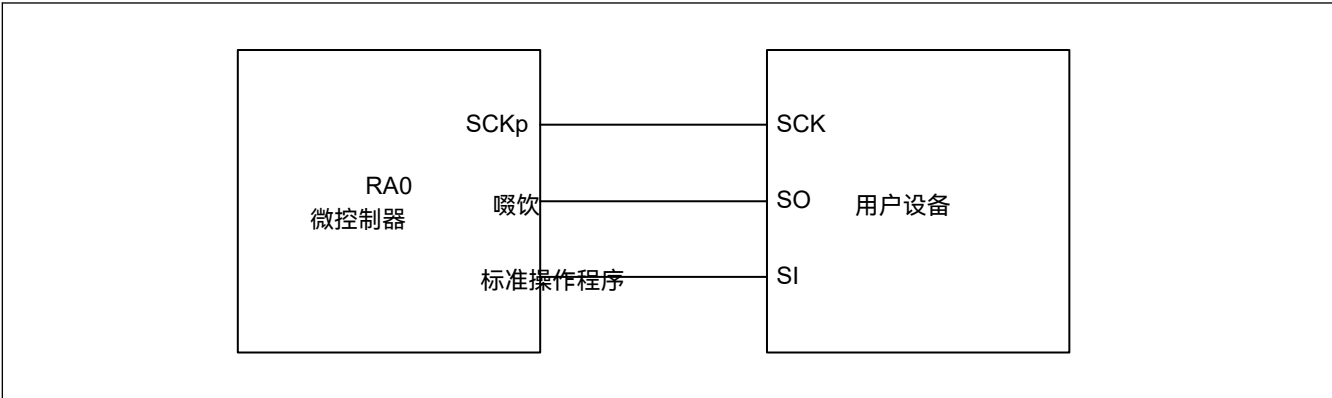


Figure 27.15 简化的SPI通信连接，适用于工作电压相同的设备。

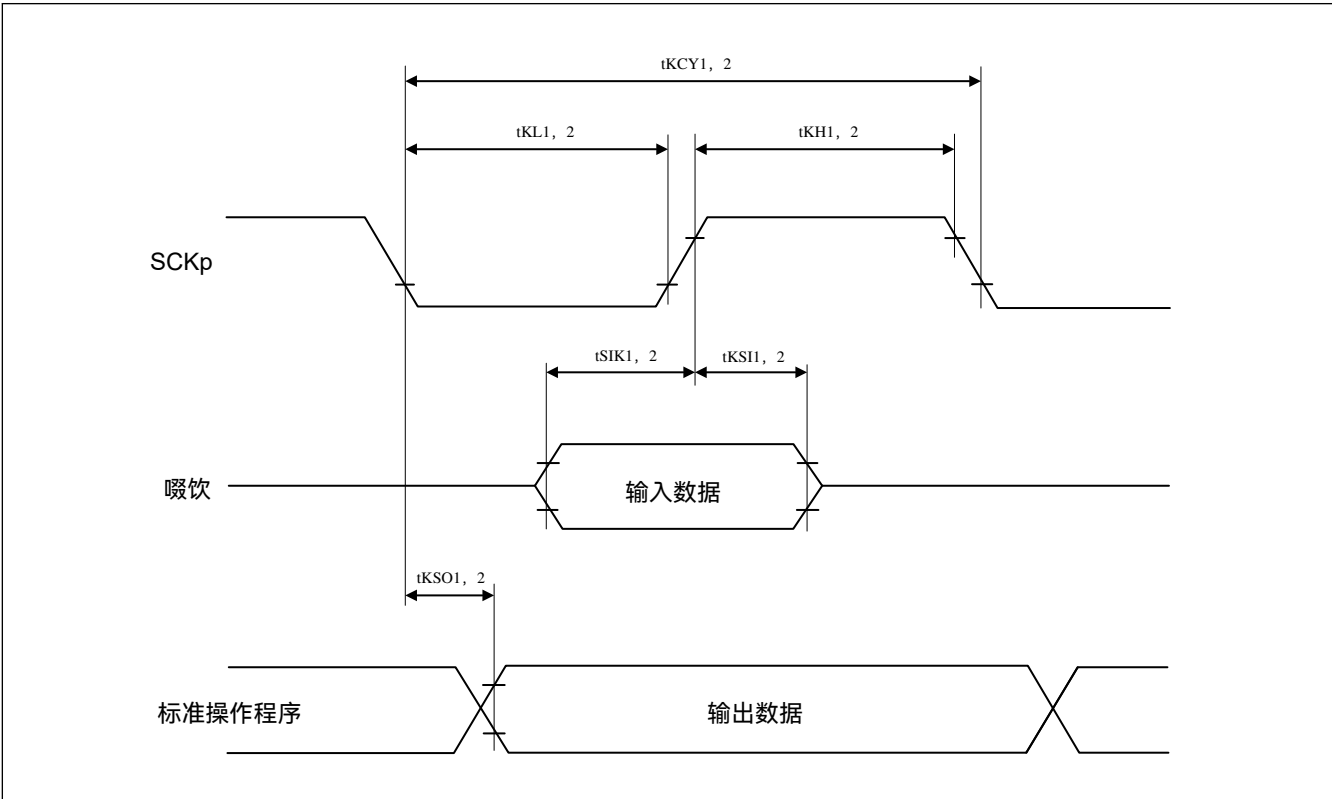


图 27.16 简化 SPI 通信中，当 SCRmn.DCP[1:0] = 00b 或 11b 时，工作电压相同的器件的串行传输时序

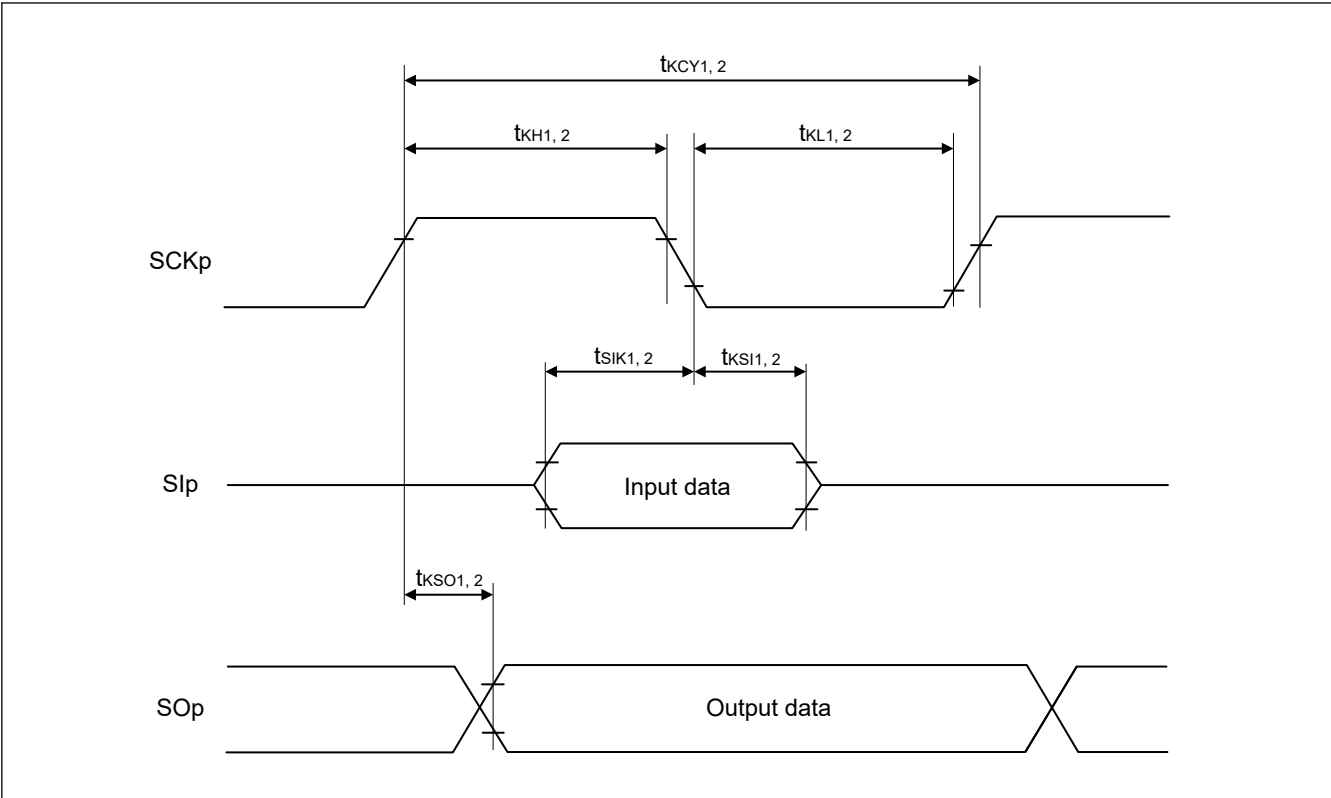


Figure 27.17 Timing of serial transfer in the simplified SPI communications with devices operating at the same voltage levels when SCRmn.DCP[1:0] = 01b or 10b

- Note:
- p: Simplified SPI number (p = 00, 11)
 - m: Unit number, n: Channel number (mn = 00, 03)

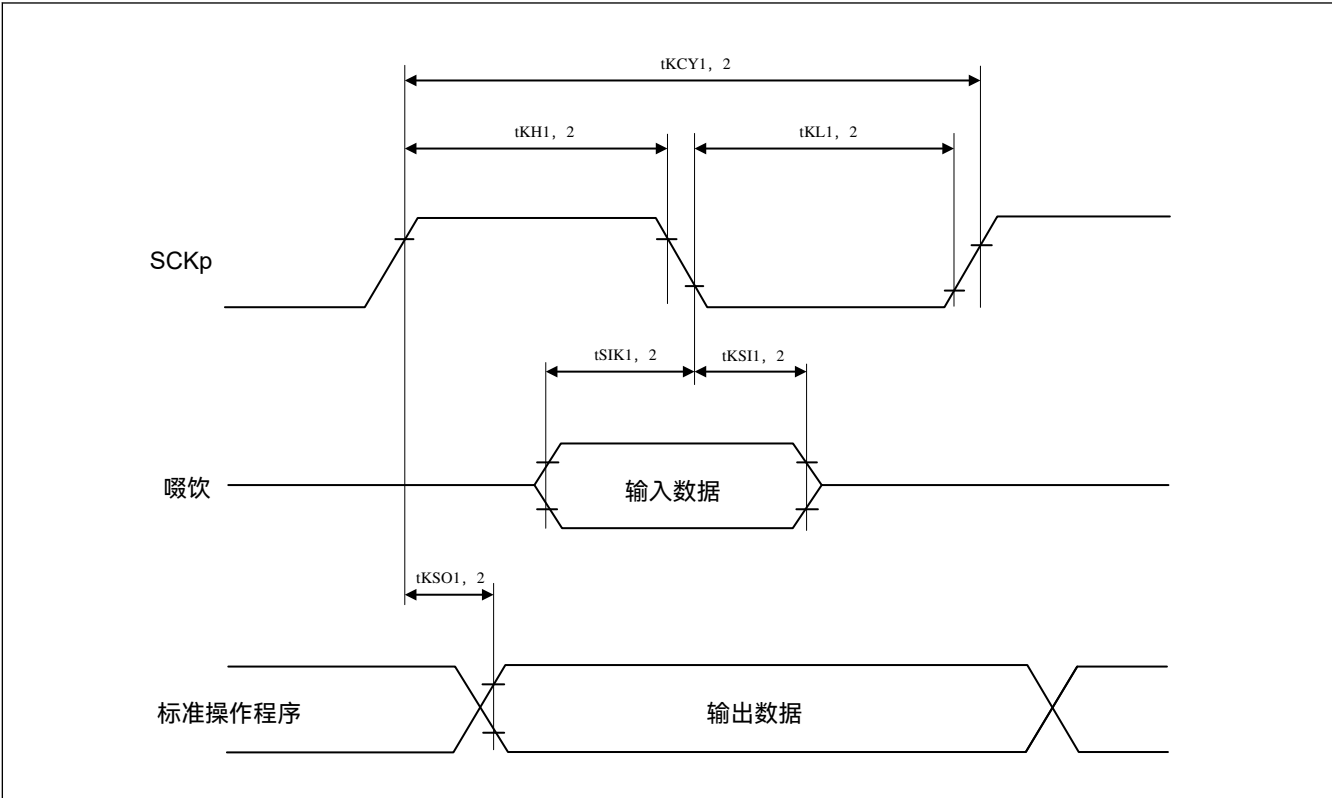


图 27.17 简化 SPI 通信中，当 SCRmn.DCP[1:0] = 01b 或 10b 时，工作电压相同的器件的串行传输时序

- 注:
- p: 简化的 SPI 编号(p = 00, 11)
 - m: 单元编号, n: 通道编号(mn = 00, 03)

Table 27.25 In simplified IIC communications with devices operating at the same voltage levels (1 of 2)

Conditions: VCC = 1.6 to 5.5 V, VSS = 0 V, Ta = -40 to +125°C

Parameter		Symbol	High-speed mode		Middle-speed mode		Low-speed mode		Unit	Test Conditions
			Min.	Max.	Min.	Max.	Min.	Max.		
SCLr clock frequency	2.7 V ≤ VCC ≤ 5.5 V, C _b = 50 pF, R _b = 2.7 kΩ	f _{SCL}	—	1000 ^{*1}	—	1000 ^{*1}	—	400 ^{*1}	kHz	Figure 27.19
	1.8 V ≤ VCC ≤ 5.5 V, C _b = 100 pF, R _b = 3 kΩ		—	400 ^{*1}	—	400 ^{*1}	—	400 ^{*1}	kHz	
	1.8 V ≤ VCC < 2.7 V, C _b = 100 pF, R _b = 5 kΩ		—	300 ^{*1}	—	300 ^{*1}	—	300 ^{*1}	kHz	
	1.6 V ≤ VCC < 1.8 V, C _b = 100 pF, R _b = 5 kΩ		—	250 ^{*1}	—	250 ^{*1}	—	250 ^{*1}	kHz	
Hold time when SCLr is low	2.7 V ≤ VCC ≤ 5.5 V, C _b = 50 pF, R _b = 2.7 kΩ	t _{LOW}	475	—	475	—	1150	—	ns	
	1.8 V ≤ VCC ≤ 5.5 V, C _b = 100 pF, R _b = 3 kΩ		1150	—	1150	—	1150	—	ns	
	1.8 V ≤ VCC < 2.7 V, C _b = 100 pF, R _b = 5 kΩ		1550	—	1550	—	1550	—	ns	
	1.6 V ≤ VCC < 1.8 V, C _b = 100 pF, R _b = 5 kΩ		1850	—	1850	—	1850	—	ns	
Hold time when SCLr is high	2.7 V ≤ VCC ≤ 5.5 V, C _b = 50 pF, R _b = 2.7 kΩ	t _{HIGH}	475	—	475	—	1150	—	ns	
	1.8 V ≤ VCC ≤ 5.5 V, C _b = 100 pF, R _b = 3 kΩ		1150	—	1150	—	1150	—	ns	
	1.8 V ≤ VCC < 2.7 V, C _b = 100 pF, R _b = 5 kΩ		1550	—	1550	—	1550	—	ns	
	1.6 V ≤ VCC < 1.8 V, C _b = 100 pF, R _b = 5 kΩ		1850	—	1850	—	1850	—	ns	

表 27.25 简化的 IIC 通信，适用于工作电压相同的设备（1/2）

条件：VCC = 1.6至5.5 V, VSS = 0 V, Ta = -40 至+125°C

范围		象征	高速模式		中速模式		低速模式		单元	测试状况
			最小。	最大限	最小。	最大限	最小。	最大限		
SCLr 时钟频率	2.7 V ≤ VCC ≤ 5.5 V, C _b = 50 pF, R _b = 2.7 kΩ	f _{SCL}	—	1000 ^{*1}	—	1000 ^{*1}	—	400 ^{*1}	kHz	图 27.19
	1.8 V ≤ VCC ≤ 5.5 V, C _b = 100 pF, R _b = 3 kΩ		—	400 ^{*1}	—	400 ^{*1}	—	400 ^{*1}	kHz	
	1.8 V ≤ VCC < 2.7 V, C _b = 100 pF, R _b = 5 kΩ		—	300 ^{*1}	—	300 ^{*1}	—	300 ^{*1}	kHz	
	1.6 V ≤ VCC < 1.8 V, C _b = 100 pF, R _b = 5 kΩ		—	250 ^{*1}	—	250 ^{*1}	—	250 ^{*1}	kHz	
保持时间 SCLr 低	2.7 V ≤ VCC ≤ 5.5 V, C _b = 50 pF, R _b = 2.7 kΩ	低	475	—	475	—	1150	—	ns	
	1.8 V ≤ VCC ≤ 5.5 V, C _b = 100 pF, R _b = 3 kΩ		1150	—	1150	—	1150	—	ns	
	1.8 V ≤ VCC < 2.7 V, C _b = 100 pF, R _b = 5 kΩ		1550	—	1550	—	1550	—	ns	
	1.6 V ≤ VCC < 1.8 V, C _b = 100 pF, R _b = 5 kΩ		1850	—	1850	—	1850	—	ns	
保持时间 SCLr 值高	2.7 V ≤ VCC ≤ 5.5 V, C _b = 50 pF, R _b = 2.7 kΩ	大腿	475	—	475	—	1150	—	ns	
	1.8 V ≤ VCC ≤ 5.5 V, C _b = 100 pF, R _b = 3 kΩ		1150	—	1150	—	1150	—	ns	
	1.8 V ≤ VCC < 2.7 V, C _b = 100 pF, R _b = 5 kΩ		1550	—	1550	—	1550	—	ns	
	1.6 V ≤ VCC < 1.8 V, C _b = 100 pF, R _b = 5 kΩ		1850	—	1850	—	1850	—	ns	

Table 27.25 In simplified IIC communications with devices operating at the same voltage levels (2 of 2)

Conditions: VCC = 1.6 to 5.5 V, VSS = 0 V, Ta = -40 to +125°C

Parameter		Symbol	High-speed mode		Middle-speed mode		Low-speed mode		Unit	Test Conditions
			Min.	Max.	Min.	Max.	Min.	Max.		
Data setup time (reception)	2.7 V ≤ VCC ≤ 5.5 V, C _b = 50 pF, R _b = 2.7 kΩ	t _{SU:DAT}	1/f _{MCK} + 85 ^{°2}	—	1/f _{MCK} + 85 ^{°2}	—	1/f _{MCK} + 145 ^{°2}	—	ns	Figure 27.19
	1.8 V ≤ VCC ≤ 5.5 V, C _b = 100 pF, R _b = 3 kΩ		1/f _{MCK} + 145 ^{°2}	—	1/f _{MCK} + 145 ^{°2}	—	1/f _{MCK} + 145 ^{°2}	—	ns	
	1.8 V ≤ VCC < 2.7 V, C _b = 100 pF, R _b = 5 kΩ		1/f _{MCK} + 230 ^{°2}	—	1/f _{MCK} + 230 ^{°2}	—	1/f _{MCK} + 230 ^{°2}	—	ns	
	1.6 V ≤ VCC < 1.8 V, C _b = 100 pF, R _b = 5 kΩ		1/f _{MCK} + 290 ^{°2}	—	1/f _{MCK} + 290 ^{°2}	—	1/f _{MCK} + 290 ^{°2}	—	ns	
Data hold time (transmission)	2.7 V ≤ VCC ≤ 5.5 V, C _b = 50 pF, R _b = 2.7 kΩ	t _{HD:DAT}	0	305	0	305	0	305	ns	
	1.8 V ≤ VCC ≤ 5.5 V, C _b = 100 pF, R _b = 3 kΩ		0	355	0	355	0	355	ns	
	1.8 V ≤ VCC < 2.7 V, C _b = 100 pF, R _b = 5 kΩ		0	405	0	405	0	405	ns	
	1.6 V ≤ VCC < 1.8 V, C _b = 100 pF, R _b = 5 kΩ		0	405	0	405	0	405	ns	

Note 1. The listed times must be no greater than f_{MCK}/4.

Note 2. Set f_{MCK} so that it does not exceed the hold time when SCLr is low or high.

Note: Select the normal input buffer and the N-ch open drain output [withstand voltage of VCC] mode for the SDAr pin and the normal output mode for the SCLr pin by using the Port gh Pin Function Select Register (PghPFS_A.PIM and PghPFS_A.NCODR).

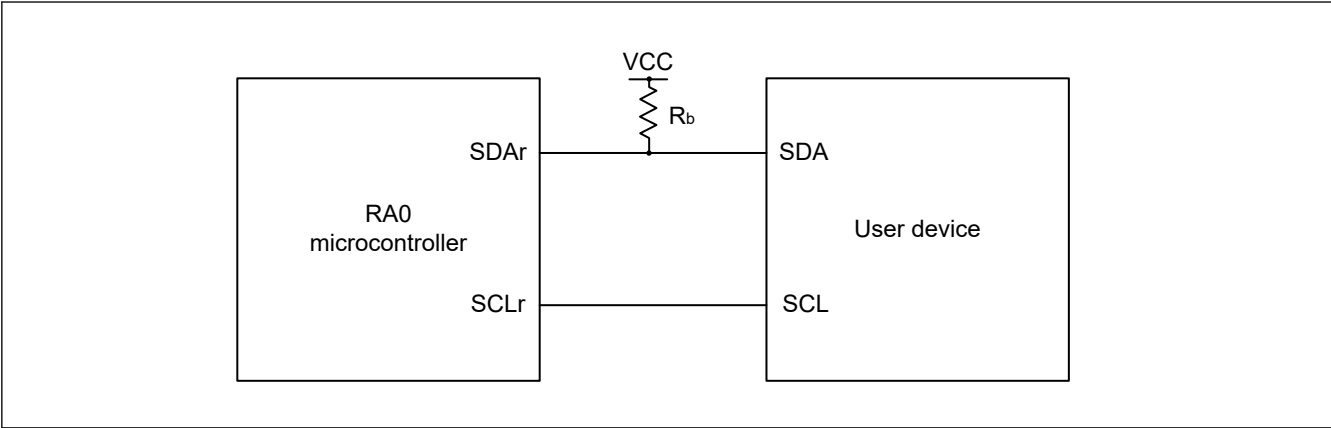


Figure 27.18 Connection in the simplified IIC communications with devices operating at the same voltage levels

表 27.25 简化的 IIC 通信，适用于工作电压相同的设备（2/2）

条件: VCC = 1.6至5.5 V, VSS= 0 V, Ta = -40 至+125°C

范围	象征	高速模式		中速模式		低速模式		单元	测试状况
		最小。	最大限	最小。	最大限	最小。	最大限		
数据建立时间 (接收)	2.7 V ≤ VCC ≤ 5.5 V, C _b = 50 pF, R _b = 2.7 kΩ	1/f _{MCK} + 85 ^{°2}	度。	1/f _{MCK} + 85 ^{°2}	度。	1/f _{MCK} + 145 ^{°2}	度。	ns	图 27.19
	1.8 V ≤ VCC ≤ 5.5 V, C _b = 100 pF, R _b = 3 kΩ	1/f _{MCK} + 145 ^{°2}	—	1/f _{MCK} + 145 ^{°2}	—	1/f _{MCK} + 145 ^{°2}	—	ns	
	1.8 V ≤ VCC < 2.7 V, C _b = 100 pF, R _b = 5 kΩ	1/f _{MCK} + 230 ^{°2}	—	1/f _{MCK} + 230 ^{°2}	—	1/f _{MCK} + 230 ^{°2}	—	ns	
	1.6 V ≤ VCC < 1.8 V, C _b = 100 pF, R _b = 5 kΩ	1/f _{MCK} + 290 ^{°2}	—	1/f _{MCK} + 290 ^{°2}	—	1/f _{MCK} + 290 ^{°2}	—	ns	
数据保持时间 (传输)	2.7 V ≤ VCC ≤ 5.5 V, C _b = 50 pF, R _b = 2.7 kΩ	0	305	0	305	0	305	ns	
	1.8 V ≤ VCC ≤ 5.5 V, C _b = 100 pF, R _b = 3 kΩ	0	355	0	355	0	355	ns	
	1.8 V ≤ VCC < 2.7 V, C _b = 100 pF, R _b = 5 kΩ	0	405	0	405	0	405	ns	
	1.6 V ≤ VCC < 1.8 V, C _b = 100 pF, R _b = 5 kΩ	0	405	0	405	0	405	ns	

注1. 所列时间不得大于f_{MCK}/4.

注 2. 设置f_{MCK}，使其在 SCLr 为低或高时不超过保持时间。

注意：使用端口 gh 引脚功能选择寄存器（PghPFS_A.PIM 和 PghPFS_A.NCODR）为 SDAr 引脚选择正常输入缓冲器和 N 通道开漏输出[耐压 VCC] 模式，为 SCLr 引脚选择正常输出模式。

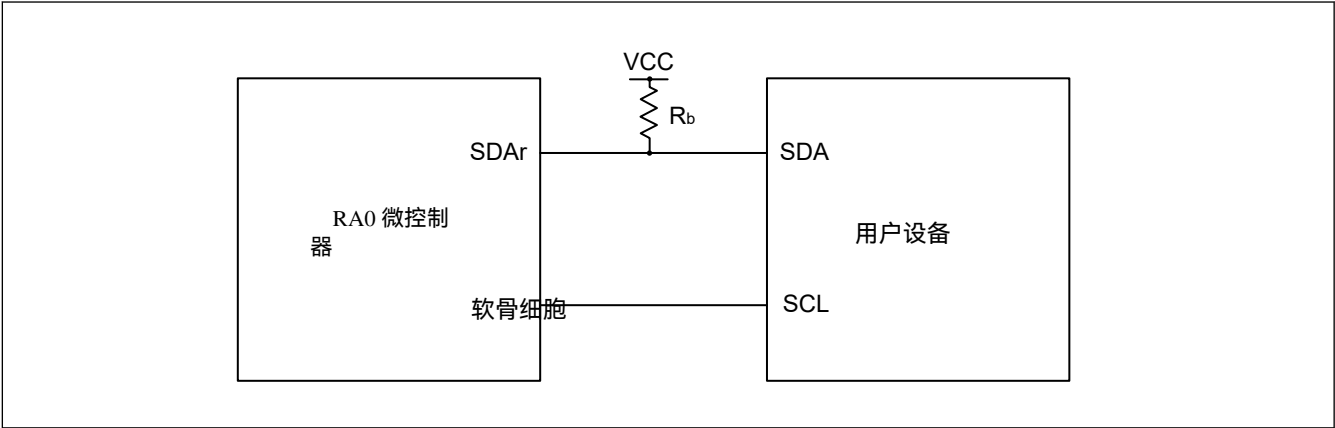


图 27.18 简化的 IIC 通信中与工作电压相同的设备的连接

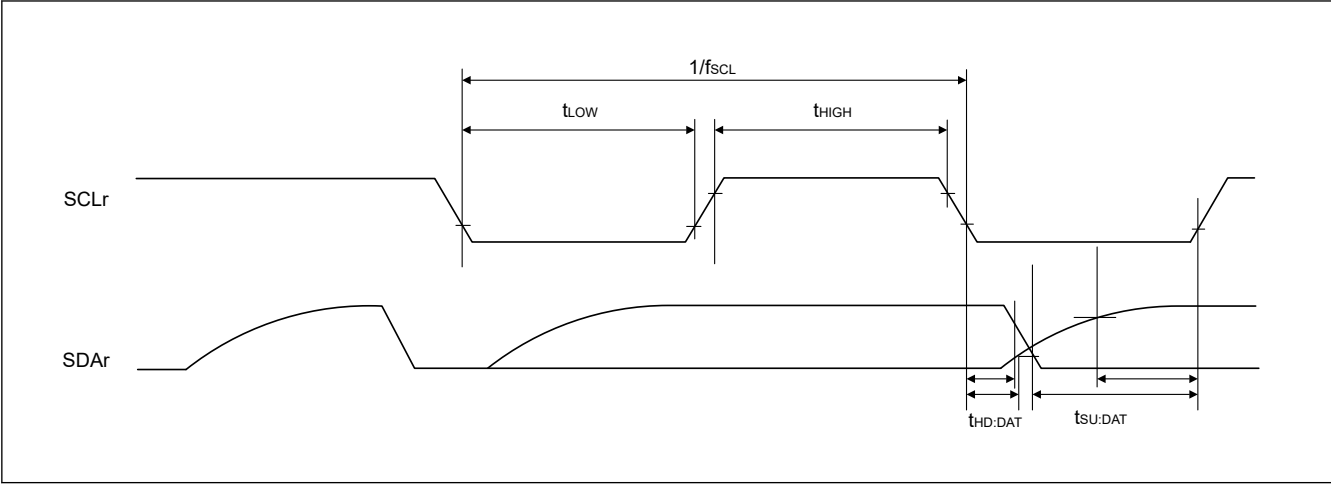


Figure 27.19 Timing of serial transfer in the simplified IIC communications with devices operating at the same voltage levels

Note:

- $R_b[\Omega]$: Communication line (SDAr) pull-up resistance, $C_b[F]$: Communication line (SDAr, SCLr) load capacitance
- r: IIC number (r = 00, 11), gh: Port number (gh = 100, 102, 110, 112, 201, 212)
- f_{MCK} : Serial array unit operation clock frequency
To set this operating clock, use the CKSmn bit in the serial mode register mn (SMRmn).
m: Unit number, n: Channel number (mn = 00, 03)

Table 27.26 In UART communications with devices operating at different voltage levels (1.8 V, 2.5 V, 3 V) (1)

Conditions: VCC = 1.8 to 5.5 V, VSS = 0 V, Ta = -40 to +125°C

Parameter		Symbol	High-speed mode		Middle-speed mode		Low-speed mode		Unit	Test Conditions
			Min.	Max.	Min.	Max.	Min.	Max.		
Transfer rate	Reception	4.0 V ≤ VCC ≤ 5.5 V, 2.7 V ≤ Vb ≤ 4.0 V	—	$f_{MCK}/6^{*1}$	—	$f_{MCK}/6^{*1}$	—	$f_{MCK}/6^{*1}$	bps	Figure 27.21
			—	5.3	—	4	—	0.33	Mbps	
		Theoretical value of the maximum transfer rate $f_{MCK} = PCLKB^{*3}$	—	5.3	—	4	—	0.33	Mbps	
			—	$f_{MCK}/6^{*1}$	—	$f_{MCK}/6^{*1}$	—	$f_{MCK}/6^{*1}$	bps	
		Theoretical value of the maximum transfer rate $f_{MCK}^{*3} = PCLKB^{*3}$	—	5.3	—	4	—	0.33	Mbps	
			—	$f_{MCK}/6^{*1}$	—	$f_{MCK}/6^{*1}$	—	$f_{MCK}/6^{*1}$	bps	
		Theoretical value of the maximum transfer rate $f_{MCK} = PCLKB^{*3}$	—	5.3	—	4	—	0.33	Mbps	
			—	$f_{MCK}/6^{*1}$	—	$f_{MCK}/6^{*1}$	—	$f_{MCK}/6^{*1}$	bps	

Note 1. Transfer rate in the Snooze mode is within the range from 4800 to 9600 bps.

Note 2. Use this rate with VCC ≥ Vb.

Note 3. The maximum operating frequencies of the system clock (PCLKB) are:
High-speed mode: 32 MHz (1.8 V ≤ VCC ≤ 5.5 V), 4 MHz (1.6 V ≤ VCC ≤ 5.5 V)
Middle-speed mode: 16 MHz (1.8 V ≤ VCC ≤ 5.5 V), 4 MHz (1.6 V ≤ VCC ≤ 5.5 V)
Low-speed mode: 2 MHz (1.6 V ≤ VCC ≤ 5.5 V)

Note: Select the TTL input buffer for the RXDq pin and the N-ch open drain output [withstand voltage of VCC] mode for the TXDq pin by using the Port gh Pin Function Select Register (PghPFS_A.PIM and PghPFS_A.NCODR). For VIH and VIL, see the DC characteristics with the TTL input buffer selected.

Note:

- Vb[V]: Communication line voltage
- q: UART number (q = 0, 1), gh: Port number (gh = 100, 101, 109, 110, 212, 213, 300)
- f_{MCK} : Serial array unit operation clock frequency

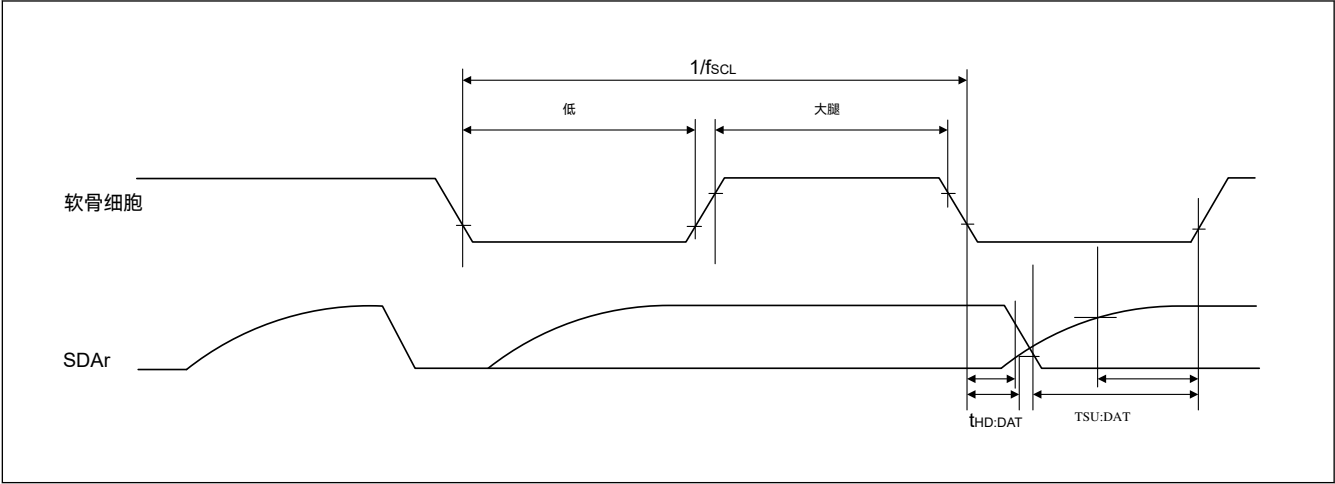


图 27.19 简化的 IIC 通信中，工作电压相同的设备之间的串行传输时序

注: ● $R_b[\Omega]$: 通信线 (SDAr) 上拉电阻, $C_b[F]$: 通信线 (SDAr, SCLr) 负载电容

- r: IIC 号码(r = 00, 11) , gh: 端口号(gh = 100, 102、110、112、201、212)
- f_{MCK} : 串行阵列单元操作时钟频率
要设置此工作时钟, 请使用串行模式寄存器 mn (SMRmn) 中的 CKSmn 位。
m: 单元编号, n: 通道编号(mn = 00, 03)

表 27.26 与工作电压等级不同的设备 (1.8 V、2.5 V、3 V) 进行 UART 通信 (1)

条件: VCC = 1.8至5.5 V, VSS = 0 V,Ta= -40 至+125°C

范围		象征	高速模式		中速模式		低速模式		单元	测试状况
			最小。	最大限度。	最小。	最大限度。	最小。	最大限度。		
Transfer rate	Reception	4.0 V ≤ VCC ≤ 5.5 V, 2.7V ≤ Vb ≤ 4.0V	—	$f_{MCK}/6^{*1}$	—	$f_{MCK}/6^{*1}$	—	$f_{MCK}/6^{*1}$	bps	数字 27.21
			—	5.3	—	4	—	0.33	兆比特	
		最大转移率的理论值 $f_{MCK} = PCLKB^{*3}$	—	5.3	—	4	—	0.33	兆比特	
			—	$f_{MCK}/6^{*1}$	—	$f_{MCK}/6^{*1}$	—	$f_{MCK}/6^{*1}$	bps	
		最大转移率的理论值 $f_{MCK}^{*3} = PCLKB^{*3}$	—	5.3	—	4	—	0.33	兆比特	
			—	$f_{MCK}/6^{*1}$	—	$f_{MCK}/6^{*1}$	—	$f_{MCK}/6^{*1}$	bps	
		最大转移率的理论值 $f_{MCK} = PCLKB^{*3}$	—	5.3	—	4	—	0.33	兆比特	
			—	$f_{MCK}/6^{*1}$	—	$f_{MCK}/6^{*1}$	—	$f_{MCK}/6^{*1}$	bps	

注1: 睡眠模式下的传输速率在4800至9600 bps范围内。

注2. 将此利率与VCC ≥ Vb一起使用

注3. 系统时钟 (PCLKB) 的最大工作频率为:
高速模式: 32 MHz (1.8 V ≤ VCC ≤ 5.5 V), 4 MHz(1.6 V ≤ VCC ≤ 5.5 V)
中速模式: 16 MHz(1.8 V ≤ VCC ≤ 5.5 V), 4MHz (1.6 V ≤ VCC ≤ 5.5 V)
低速模式: 2 MHz(1.6 V ≤ VCC ≤ 5.5 V)

注意: 使用端口gh引脚功能选择寄存器 (PghPFS_A.PIM和PghPFS_A.NCODR) 为RXDq引脚选择TTL输入缓冲器, 并为TXDq引脚选择N沟道开漏输出模式 ([VCC耐压模式,])。对于 VIH和VIL, 请参阅选择TTL输入缓冲器后的直流特性。

注: ● Vb[V]: 通信线路电压

- q: UART 号(q = 0, 1) gh: 端口号(gh = 100, 101, 109、110、212、213、300)
- f_{MCK} : 串行阵列单元操作时钟频率

To set this operating clock, use the CKS bit in the serial mode register mn (SMRmn).
m: Unit number, n: Channel number (mn = 00, 01, 02, 03)

- Communications by using P212 and P213 with devices operating at different voltage levels are not possible since P212PFS_A and P213PFS_A registers do not have PIM bit.

Table 27.27 In UART communications with devices operating at different voltage levels (1.8 V, 2.5 V, 3 V) (2)

Conditions: VCC = 1.8 to 5.5 V, VSS = 0 V, Ta = -40 to +125°C

Parameter		Symbol	High-speed mode		Middle-speed mode		Low-speed mode		Unit	Test Conditions
			Min.	Max.	Min.	Max.	Min.	Max.		
Transfer rate	Transmission	4.0 V ≤ VCC ≤ 5.5 V, 2.7 V ≤ V _b ≤ 4.0 V	—	*1	—	*1	—	*1	bps	Figure 27.21
			—	2.8*2	—	2.8*2	—	2.8*2	Mbps	
		Theoretical value of the maximum transfer rate C _b = 50 pF, R _b = 1.4 kΩ, V _b = 2.7 V	—	2.8*2	—	2.8*2	—	2.8*2	Mbps	
			—	2.8*2	—	2.8*2	—	2.8*2	Mbps	
		2.7 V ≤ VCC < 4.0 V, 2.3 V ≤ V _b ≤ 2.7 V	—	*3	—	*3	—	*3	bps	
			—	1.2*4	—	1.2*4	—	1.2*4	Mbps	
		Theoretical value of the maximum transfer rate C _b = 50 pF, R _b = 2.7 kΩ, V _b = 2.3 V	—	1.2*4	—	1.2*4	—	1.2*4	Mbps	
			—	1.2*4	—	1.2*4	—	1.2*4	Mbps	
		1.8 V ≤ VCC < 3.3 V, 1.6 V ≤ V _b ≤ 2.0 V	—	*5 *6	—	*5 *6	—	*5 *6	bps	
			—	0.43*7	—	0.43*7	—	0.43*7	Mbps	
		Theoretical value of the maximum transfer rate C _b = 50 pF, R _b = 5.5 kΩ, V _b = 1.6 V	—	0.43*7	—	0.43*7	—	0.43*7	Mbps	
			—	0.43*7	—	0.43*7	—	0.43*7	Mbps	

Note 1. The smaller maximum transfer rate derived by using f_{MCK}/6 or the following expression is the valid maximum transfer rate.
Expression for calculating the transfer rate when 4.0 V ≤ VCC ≤ 5.5 V, 2.7 V ≤ V_b ≤ 4.0 V

Maximum transfer rate = $\frac{1}{\left\{ -C_b \times R_b \times \ln \left(1 - \frac{2.2}{V_b} \right) \right\} \times 3}$ [bps]

Baud rate error (theoretical value) = $\frac{\frac{1}{\text{Transfer rate} \times 2} - \left\{ -C_b \times R_b \times \ln \left(1 - \frac{2.2}{V_b} \right) \right\}}{\left(\frac{1}{\text{Transfer rate}} \right) \times \text{Number of transferred bits}} \times 100[\%]$

This value is the theoretical value of the relative difference between the transmission and reception sides.

Note 2. This rate is calculated as an example when the conditions described in the Conditions column are met. See *1 above to calculate the maximum transfer rate under the conditions of the customer.

Note 3. The smaller maximum transfer rate derived by using f_{MCK}/6 or the following expression is the valid maximum transfer rate.
Expression for calculating the transfer rate when 2.7 V ≤ VCC < 4.0 V, 2.3 V ≤ V_b ≤ 2.7 V

Maximum transfer rate = $\frac{1}{\left\{ -C_b \times R_b \times \ln \left(1 - \frac{2.0}{V_b} \right) \right\} \times 3}$ [bps]

Baud rate error (theoretical value) = $\frac{\frac{1}{\text{Transfer rate} \times 2} - \left\{ -C_b \times R_b \times \ln \left(1 - \frac{2.0}{V_b} \right) \right\}}{\left(\frac{1}{\text{Transfer rate}} \right) \times \text{Number of transferred bits}} \times 100[\%]$

This value is the theoretical value of the relative difference between the transmission and reception sides.

Note 4. This rate is calculated as an example when the conditions described in the Conditions column are met. See *3 above to calculate the maximum transfer rate under the conditions of the customer.

Note 5. Use this rate with VCC ≥ V_b.

Note 6. The smaller maximum transfer rate derived by using f_{MCK}/6 or the following expression is the valid maximum transfer rate.
Expression for calculating the transfer rate when 1.8 V ≤ VCC < 3.3 V, 1.6 V ≤ V_b ≤ 2.0 V

Maximum transfer rate = $\frac{1}{\left\{ -C_b \times R_b \times \ln \left(1 - \frac{1.5}{V_b} \right) \right\} \times 3}$ [bps]

Baud rate error (theoretical value) = $\frac{\frac{1}{\text{Transfer rate} \times 2} - \left\{ -C_b \times R_b \times \ln \left(1 - \frac{1.5}{V_b} \right) \right\}}{\left(\frac{1}{\text{Transfer rate}} \right) \times \text{Number of transferred bits}} \times 100[\%]$

要设置此工作时钟，请使用串行模式寄存器 mn (SMRmn) 中的 CKS 位。
m：单元编号，n：通道编号(mn = 00, 01, 02, 03)

- 由于 P212PFS_A 和 P213PFS_A 寄存器没有 PIM 位，因此无法使用 P212 和 P213 与工作电压不同的设备进行通信。

表 27.27 与工作电压等级不同的设备（1.8 V、2.5 V、3 V）进行 UART 通信 (2)

条件：VCC = 1.8到5.5 V, VSS = 0 V,然后= -40 到+125°C

范围		象征	高速模式		中速模式		低速模式		单元	测试状况
			最小。	最大限度。	最小。	最大限度。	最小。	最大限度。		
Transfer rate	Transmission	4.0 V ≤ VCC ≤ 5.5 V, 2.7V ≤ V _b ≤ 4.0V	—	*1	—	*1	—	*1	bps	数字 27.21
			—	2.8*2	—	2.8*2	—	2.8*2	兆比特	
		Theoretical value of the maximum transfer rate: C _b = 50pF, R _b = 10.4 kΩ, V _b = 20.7 V	—	2.8*2	—	2.8*2	—	2.8*2	兆比特	
			—	2.8*2	—	2.8*2	—	2.8*2	兆比特	
		2.7 V ≤ VCC < 4.0 V, 2.3V ≤ V _b ≤ 2.7V	—	*3	—	*3	—	*3	bps	
			—	1.2*4	—	1.2*4	—	1.2*4	兆比特	
		Theoretical value of the maximum transfer rate C _b = 50pF, R _b = 2.7 kΩ, V _b = 2.3 V	—	1.2*4	—	1.2*4	—	1.2*4	兆比特	
			—	1.2*4	—	1.2*4	—	1.2*4	兆比特	
		1.8 V ≤ VCC < 3.3 V, 1.6V ≤ V _b ≤ 2.0V	—	*5 *6	—	*5 *6	—	*5 *6	bps	
			—	0.43*7	—	0.43*7	—	0.43*7	兆比特	
		Theoretical value of the maximum transfer rate C _b = 50pF, R _b = 5.5 kΩ, V _b = 1.6 V	—	0.43*7	—	0.43*7	—	0.43*7	兆比特	
			—	0.43*7	—	0.43*7	—	0.43*7	兆比特	

注 1. 使用f_{MCK}/6 或以下表达式得出的较小最大转移率是有效的最大转移率。
当4.0 V ≤ VCC ≤ 5.5 V, 2.7 V ≤ V_b ≤ 4.0 V时，计算转移率的表达式

最大转账速率 = $\frac{1}{\left\{ -C_b \times R_b \times \ln \left(1 - \frac{2.2}{V_b} \right) \right\} \times 3}$ [bps]

波特率误差 (oretical value) = $\frac{\frac{1}{\text{Transfer rate} \times 2} - \left\{ -C_b \times R_b \times \ln \left(1 - \frac{2.2}{V_b} \right) \right\}}{\left(\frac{1}{\text{Transfer rate}} \right) \times \text{Number of transferred bits}} \times 100[\%]$

该值是发射端和接收端之间相对差异的理论值。

注2：此费率是在满足“条件”列中所述条件时计算的示例费率。See *1 用于计算客户条件下的最大转账费率。

注 3. 使用f_{MCK}/6 或以下表达式得出的较小最大转移率是有效的最大转移率。

Expression for calculating the transfer rate when 2.7 V ≤ VCC < 4.0 V, 2.3 V ≤ V_b ≤ 2.7 V

格言 m transfer rate = $\frac{1}{\left\{ -C_b \times R_b \times \ln \left(1 - \frac{2.0}{V_b} \right) \right\} \times 3}$ [bps]

波特率 error (theoretical value) = $\frac{\frac{1}{\text{Transfer rate} \times 2} - \left\{ -C_b \times R_b \times \ln \left(1 - \frac{2.0}{V_b} \right) \right\}}{\left(\frac{1}{\text{Transfer rate}} \right) \times \text{Number of transferred bits}} \times 100[\%]$

该值是发射端和接收端之间相对差异的理论值。

注4：此费率是在满足“条件”列中所述条件时计算的示例。See *3用于计算客户条件下的最大转账费率。

注5. 将此利率与VCC ≥ V_b.一起使用

注 6. 使用f_{MCK}/6 或以下表达式得出的较小最大转移率是有效的最大转移率。
当1.8 V ≤ VCC < 3.3 V, 1.6 V ≤ V_b ≤ 2.0 V时，计算转移率的表达式

Maximum transfer rate = $\frac{1}{\left\{ -C_b \times R_b \times \ln \left(1 - \frac{1.5}{V_b} \right) \right\} \times 3}$ [bps]

Baud rate error (theoretical value) = $\frac{\frac{1}{\text{Transfer rate} \times 2} - \left\{ -C_b \times R_b \times \ln \left(1 - \frac{1.5}{V_b} \right) \right\}}{\left(\frac{1}{\text{Transfer rate}} \right) \times \text{Number of transferred bits}} \times 100[\%]$

This value is the theoretical value of the relative difference between the transmission and reception sides.

Note 7. This rate is calculated as an example when the conditions described in the Conditions column are met. See *6 above to calculate the maximum transfer rate under the conditions of the customer.

Note: Select the TTL input buffer for the RXDq pin and the N-ch open drain output [withstand voltage of VCC] mode for the TXDq pin by using the Port gh Pin Function Select Register (PghPFS_A.PIM and PghPFS_A.NCODR). For V_{IH} and V_{IL} , see the DC characteristics with the TTL input buffer selected.

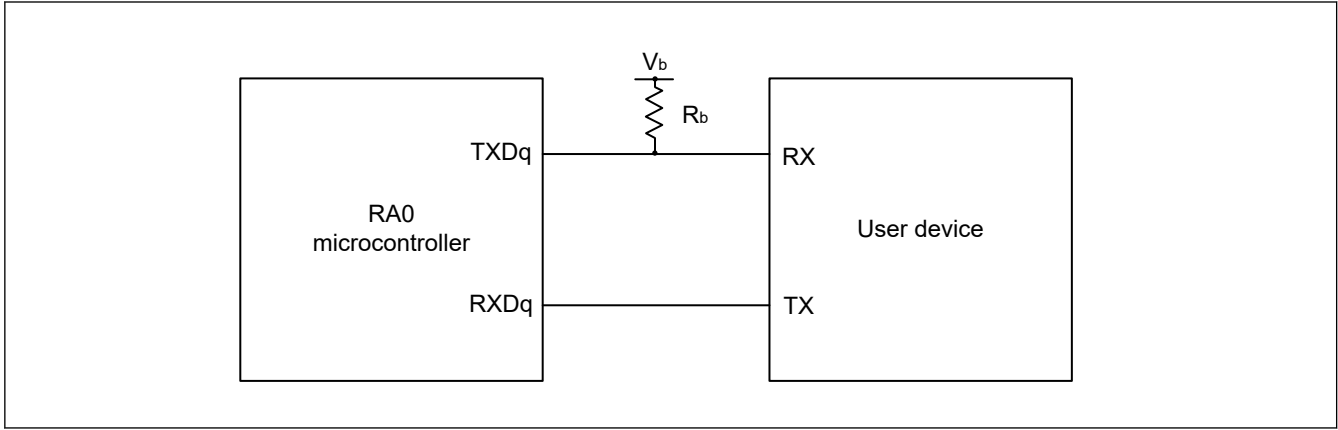


Figure 27.20 In UART communications with devices operating at different voltage levels

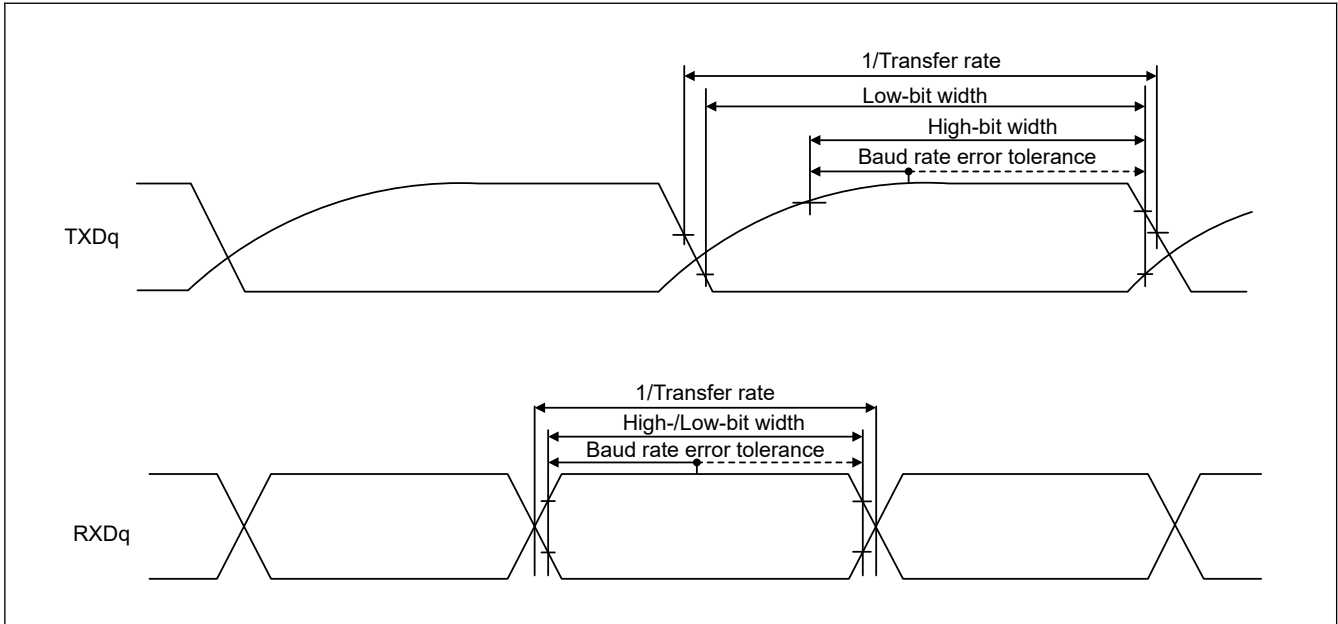


Figure 27.21 Bit width in the UART communications with devices operating at different voltage levels (reference)

- Note:
- $R_b[\Omega]$: Communication line (TXDq) pull-up resistance, $C_b[F]$: Communication line (TXDq) load capacitance, $V_b[V]$: Communication line voltage
 - q: UART number (q = 0, 1), gh: Port number (gh = 100, 101, 109, 110, 212, 213, 300)
 - f_{MCK} : Serial array unit operation clock frequency
To set this operating clock, use the CKS bit in the serial mode register mn (SMRmn).
m: Unit number, n: Channel number (mn = 00, 01, 02, 03)
 - Communications by using P212 and P213 with devices operating at different voltage levels are not possible since P212PFS_A and P213PFS_A registers do not have PIM bit.

该值是发射端和接收端之间相对差异的理论值。

注7: 此费率仅为示例, 计算时需满足“条件”栏中所述的条件。有关客户具体条件下的最高转账费率, 请参见上文*6。

注意: 使用端口 gh 引脚功能选择寄存器 (PghPFS_A.PIM 和 PghPFS_A.NCODR) 为 RXDq 引脚选择 TTL 输入缓冲器, 并为 TXDq 引脚选择 N 沟道开漏输出模式 ([耐压 VCC])。对于 V_{IH} 和 V_{IL} , 请参阅选择 TTL 输入缓冲器后的直流特性。

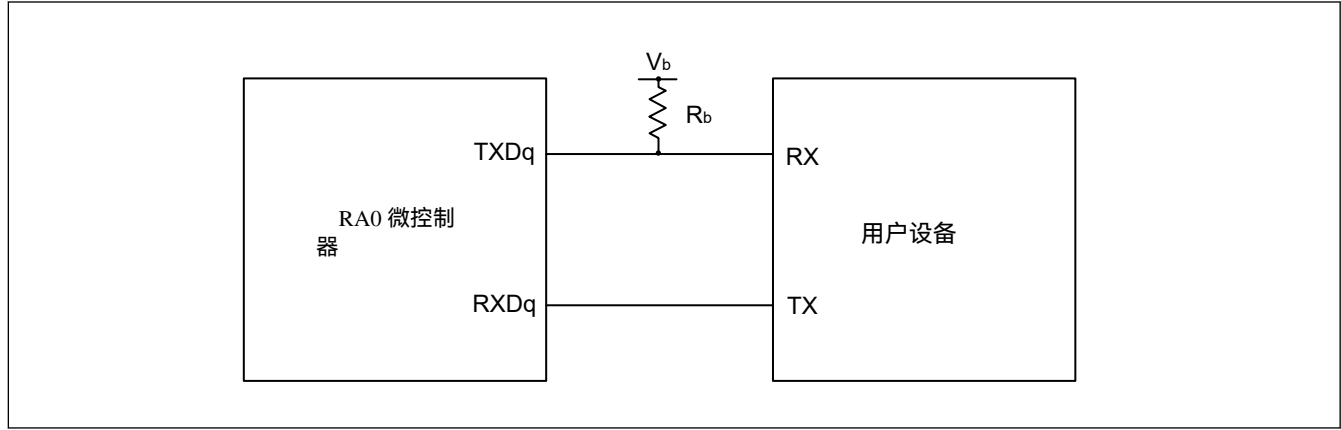


图 27.20 UART 与工作电压不同的设备进行通信

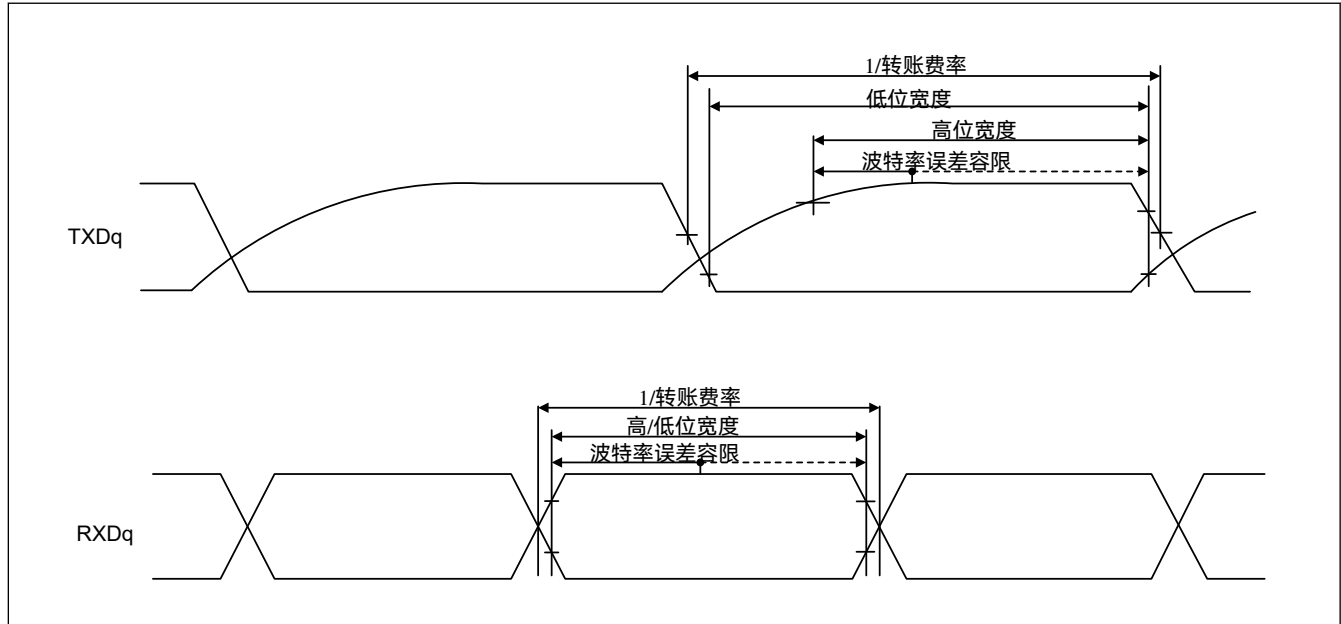


图 27.21 不同电压电平下设备的 UART 通信中的位宽 (参考)

- 注: ● $R_b[\Omega]$: 通信线 (TXDq) 上拉电阻, $C_b[F]$: 通信线 (TXDq) 负载电容, $V_b[V]$: 通信线路电压
- q: UART 号 (q = 0, 1) gh: 端口号 (gh = 100, 101, 109, 110, 212, 213, 300)
 - f_{MCK} : 串行阵列单元操作时钟频率
要设置此工作时钟, 请使用串行模式寄存器 mn (SMRmn) 中的 CKS 位。
m: 单元编号, n: 通道编号 (mn = 00, 01, 02, 03)
 - 由于 P212PFS_A 和 P213PFS_A 寄存器没有 PIM 位, 因此无法使用 P212 和 P213 与工作电压不同的设备进行通信。

Table 27.28 In simplified SPI communications in the master mode with devices operating at different voltage levels (2.5 V or 3 V) with the internal SCKp clock (the ratings below are only applicable to SPI00)

Conditions: VCC = 2.7 to 5.5 V, VSS = 0 V, Ta = -40 to +105°C

Parameter			Symbol	High-speed mode		Middle-speed mode		Low-speed mode		Unit	Test Conditions
				Min.	Max.	Min.	Max.	Min.	Max.		
SCKp cycle time	$t_{KCY1} \geq 2/PCLKB$	$4.0\text{ V} \leq VCC \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$, $C_b = 20\text{ pF}$, $R_b = 1.4\text{ k}\Omega$	t_{KCY1}	200	—	200	—	2300	—	ns	Figure 27.23 Figure 27.24
		$2.7\text{ V} \leq VCC < 4.0\text{ V}$, $2.3\text{ V} \leq V_b \leq 2.7\text{ V}$, $C_b = 20\text{ pF}$, $R_b = 2.7\text{ k}\Omega$		300	—	300	—	2300	—	ns	
SCKp high-level width		$4.0\text{ V} \leq VCC \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$, $C_b = 20\text{ pF}$, $R_b = 1.4\text{ k}\Omega$	t_{KH1}	$t_{KCY1}/2 - 50$	—	$t_{KCY1}/2 - 50$	—	$t_{KCY1}/2 - 50$	—	ns	
		$2.7\text{ V} \leq VCC < 4.0\text{ V}$, $2.3\text{ V} \leq V_b \leq 2.7\text{ V}$, $C_b = 20\text{ pF}$, $R_b = 2.7\text{ k}\Omega$		$t_{KCY1}/2 - 120$	—	$t_{KCY1}/2 - 120$	—	$t_{KCY1}/2 - 120$	—	ns	
SCKp low-level width		$4.0\text{ V} \leq VCC \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$, $C_b = 20\text{ pF}$, $R_b = 1.4\text{ k}\Omega$	t_{KL1}	$t_{KCY1}/2 - 7$	—	$t_{KCY1}/2 - 7$	—	$t_{KCY1}/2 - 50$	—	ns	
		$2.7\text{ V} \leq VCC < 4.0\text{ V}$, $2.3\text{ V} \leq V_b \leq 2.7\text{ V}$, $C_b = 20\text{ pF}$, $R_b = 2.7\text{ k}\Omega$		$t_{KCY1}/2 - 10$	—	$t_{KCY1}/2 - 10$	—	$t_{KCY1}/2 - 50$	—	ns	
Slp setup time (to SCKp↑)*1		$4.0\text{ V} \leq VCC \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$, $C_b = 20\text{ pF}$, $R_b = 1.4\text{ k}\Omega$	t_{SIK1}	58	—	58	—	479	—	ns	
		$2.7\text{ V} \leq VCC < 4.0\text{ V}$, $2.3\text{ V} \leq V_b \leq 2.7\text{ V}$, $C_b = 20\text{ pF}$, $R_b = 2.7\text{ k}\Omega$		121	—	121	—	479	—	ns	
Slp hold time (from SCKp↑)*1		$4.0\text{ V} \leq VCC \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$, $C_b = 20\text{ pF}$, $R_b = 1.4\text{ k}\Omega$	t_{KSI1}	10	—	10	—	10	—	ns	
		$2.7\text{ V} \leq VCC < 4.0\text{ V}$, $2.3\text{ V} \leq V_b \leq 2.7\text{ V}$, $C_b = 20\text{ pF}$, $R_b = 2.7\text{ k}\Omega$		10	—	10	—	10	—	ns	
Delay time from SCKp↓ to SOp output*1		$4.0\text{ V} \leq VCC \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$, $C_b = 20\text{ pF}$, $R_b = 1.4\text{ k}\Omega$	t_{KSO1}	—	60	—	60	—	60	ns	
		$2.7\text{ V} \leq VCC < 4.0\text{ V}$, $2.3\text{ V} \leq V_b \leq 2.7\text{ V}$, $C_b = 20\text{ pF}$, $R_b = 2.7\text{ k}\Omega$		—	130	—	130	—	130	ns	
Slp setup time (to SCKp↓)*2		$4.0\text{ V} \leq VCC \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$, $C_b = 20\text{ pF}$, $R_b = 1.4\text{ k}\Omega$	t_{SIK1}	23	—	23	—	110	—	ns	
		$2.7\text{ V} \leq VCC < 4.0\text{ V}$, $2.3\text{ V} \leq V_b \leq 2.7\text{ V}$, $C_b = 20\text{ pF}$, $R_b = 2.7\text{ k}\Omega$		33	—	33	—	110	—	ns	
Slp hold time (from SCKp↓)*2		$4.0\text{ V} \leq VCC \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$, $C_b = 20\text{ pF}$, $R_b = 1.4\text{ k}\Omega$	t_{KSI1}	10	—	10	—	10	—	ns	
		$2.7\text{ V} \leq VCC < 4.0\text{ V}$, $2.3\text{ V} \leq V_b \leq 2.7\text{ V}$, $C_b = 20\text{ pF}$, $R_b = 2.7\text{ k}\Omega$		10	—	10	—	10	—	ns	
Delay time from SCKp↑ to SOp output*2		$4.0\text{ V} \leq VCC \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$, $C_b = 20\text{ pF}$, $R_b = 1.4\text{ k}\Omega$	t_{KSO1}	—	10	—	10	—	10	ns	
		$2.7\text{ V} \leq VCC < 4.0\text{ V}$, $2.3\text{ V} \leq V_b \leq 2.7\text{ V}$, $C_b = 20\text{ pF}$, $R_b = 2.7\text{ k}\Omega$		—	10	—	10	—	10	ns	

Note 1. This setting applies when SCRmn.DCP[1:0] = 00b or 11b.

Note 2. This setting applies when SCRmn.DCP[1:0] = 01b or 10b.

Note: Select the TTL input buffer for the Slp pin and the N-ch open drain output [withstand voltage of VCC] mode for the SOp pin and SCKp pin by using the Port gh Pin Function Select Register (PghPFS_A.PIM and PghPFS_A.NCODR). For V_{IH} and V_{IL} , see the DC characteristics with the TTL input buffer selected.

- Note:
- $R_b[\Omega]$: Communication line (SCKp, SOp) pull-up resistance, $C_b[F]$: Communication line (SCKp, SOp) load capacitance, $V_b[V]$: Communication line voltage
 - p: Simplified SPI number (p = 00), m: Unit number (m = 0), n: Channel number (n = 0), gh: Port number (gh = 100 to 102, 108 to 110, 112)
 - f_{MCK} : Serial array unit operation clock frequency

Table 27.28 在主模式下，简化的 SPI 通信中，设备以不同的电压电平(2.5 V或3 V) 运行，并使用内部 SCKp 时钟（以下额定值仅适用于 SPI00）。

条件: VCC = 2.7至5.5 V, VSS= 0 V, Ta= -40 至+105°C

范围			象征	高速模式		中速模式		低速模式		单元	测试状况
				最小。	最大限 度。	最小。	最大限 度。	最小。	最大限 度。		
SCKp 周期	$t_{KCY1} \geq 2/PCLKB$	$4.0\text{ V} \leq VCC \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$, $C_b = 20\text{ pF}$, $R_b = 1.4\text{ k}\Omega$	t_{KCY1}	200	—	200	—	2300	—	ns	图 27.23 图 27.24
		$2.7\text{ V} \leq VCC < 4.0\text{ V}$, $2.3\text{ V} \leq V_b \leq 2.7\text{ V}$, $C_b = 20\text{ pF}$, $R_b = 2.7\text{ k}\Omega$		300	—	300	—	2300	—	ns	
SCKp 高电平宽度		$4.0\text{ V} \leq VCC \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$, $C_b = 20\text{ pF}$, $R_b = 1.4\text{ k}\Omega$	t_{KH1}	$t_{KCY1}/2 - 50$	—	$t_{KCY1}/2 - 50$	—	$t_{KCY1}/2 - 50$	—	ns	
		$2.7\text{ V} \leq VCC < 4.0\text{ V}$, $2.3\text{ V} \leq V_b \leq 2.7\text{ V}$, $C_b = 20\text{ pF}$, $R_b = 2.7\text{ k}\Omega$		$t_{KCY1}/2 - 120$	—	$t_{KCY1}/2 - 120$	—	$t_{KCY1}/2 - 120$	—	ns	
SCKp 低层宽度		$4.0\text{ V} \leq VCC \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$, $C_b = 20\text{ pF}$, $R_b = 1.4\text{ k}\Omega$	t_{KL1}	$t_{KCY1}/2 - 7$	—	$t_{KCY1}/2 - 7$	—	$t_{KCY1}/2 - 50$	—	ns	
		$2.7\text{ V} \leq VCC < 4.0\text{ V}$, $2.3\text{ V} \leq V_b \leq 2.7\text{ V}$, $C_b = 20\text{ pF}$, $R_b = 2.7\text{ k}\Omega$		$t_{KCY1}/2 - 10$	—	$t_{KCY1}/2 - 10$	—	$t_{KCY1}/2 - 50$	—	ns	
Slp 设置时间 (至 SCKp↑)*1		$4.0\text{ V} \leq VCC \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$, $C_b = 20\text{ pF}$, $R_b = 1.4\text{ k}\Omega$	t_{SIK1}	58	—	58	—	479	—	ns	
		$2.7\text{ V} \leq VCC < 4.0\text{ V}$, $2.3\text{ V} \leq V_b \leq 2.7\text{ V}$, $C_b = 20\text{ pF}$, $R_b = 2.7\text{ k}\Omega$		121	—	121	—	479	—	ns	
啣饮保持时间 (来自 SCKp↑)*1		$4.0\text{ V} \leq VCC \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$, $C_b = 20\text{ pF}$, $R_b = 1.4\text{ k}\Omega$	t_{KSI1}	10	—	10	—	10	—	ns	
		$2.7\text{ V} \leq VCC < 4.0\text{ V}$, $2.3\text{ V} \leq V_b \leq 2.7\text{ V}$, $C_b = 20\text{ pF}$, $R_b = 2.7\text{ k}\Omega$		10	—	10	—	10	—	ns	
从SCKp↓到SOp输出的延迟时间*1		$4.0\text{ V} \leq VCC \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$, $C_b = 20\text{ pF}$, $R_b = 1.4\text{ k}\Omega$	t_{KSO1}	—	60	—	60	—	60	ns	
		$2.7\text{ V} \leq VCC < 4.0\text{ V}$, $2.3\text{ V} \leq V_b \leq 2.7\text{ V}$, $C_b = 20\text{ pF}$, $R_b = 2.7\text{ k}\Omega$		—	130	—	130	—	130	ns	
Slp 设置时间 (至SCKp↓)*2		$4.0\text{ V} \leq VCC \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$, $C_b = 20\text{ pF}$, $R_b = 1.4\text{ k}\Omega$	t_{SIK1}	23	—	23	—	110	—	ns	
		$2.7\text{ V} \leq VCC < 4.0\text{ V}$, $2.3\text{ V} \leq V_b \leq 2.7\text{ V}$, $C_b = 20\text{ pF}$, $R_b = 2.7\text{ k}\Omega$		33	—	33	—	110	—	ns	
啣饮保持时间 (从 SCKp↓)*2		$4.0\text{ V} \leq VCC \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$, $C_b = 20\text{ pF}$, $R_b = 1.4\text{ k}\Omega$	t_{KSI1}	10	—	10	—	10	—	ns	
		$2.7\text{ V} \leq VCC < 4.0\text{ V}$, $2.3\text{ V} \leq V_b \leq 2.7\text{ V}$, $C_b = 20\text{ pF}$, $R_b = 2.7\text{ k}\Omega$		10	—	10	—	10	—	ns	
从SCKp↑到SOp输出的延迟时间*2		$4.0\text{ V} \leq VCC \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$, $C_b = 20\text{ pF}$, $R_b = 1.4\text{ k}\Omega$	t_{KSO1}	—	10	—	10	—	10	ns	
		$2.7\text{ V} \leq VCC < 4.0\text{ V}$, $2.3\text{ V} \leq V_b \leq 2.7\text{ V}$, $C_b = 20\text{ pF}$, $R_b = 2.7\text{ k}\Omega$		—	10	—	10	—	10	ns	

注 1. 此设置适用于 SCRmn.DCP[1:0] = 00b 或 11b。

注 2. 此设置适用于 SCRmn.DCP[1:0] = 01b 或 10b。

注意: 使用端口_{gh}引脚功能选择寄存器 (PghPFS_A.PIM和PghPFS_A.NCODR) 为Slp引脚选择TTL输入缓冲器，并为[引脚选择N沟道开漏输出耐压模式 (VCC为)]，为SOp引脚和SCKp引脚选择N沟道开漏输出模式。对于 V_{IH} 和 V_{IL} ，引脚，请参阅选择TTL输入缓冲器后的直流特性。

- 注: ● $R_b[\Omega]$: 通信线路(SCKp, SOp) 上拉电阻, $C_b[F]$: 通信线路 (SCKp、SOp) 负载电容, $V_b[V]$: 通信线路电压
- p: 简化的 SPI 编号(p = 00) , m: 单元编号(m = 0) n: 通道编号(n = 0), gh: 端口编号(gh = (100 至 102, 108 至 110, 112)
- fMCK: 串行阵列单元操作时钟频率

To set this operating clock, use the CKSmn bit in the serial mode register mn (SMRmn).
m: Unit number, n: Channel number (mn = 00)

Table 27.29 In simplified SPI communications in the master mode with devices operating at different voltage levels (1.8 V, 2.5 V, or 3 V) with the internal SCKp clock (1)

Conditions: VCC = 1.8 to 5.5 V, VSS = 0 V, Ta = -40 to +125°C

Parameter			Symbol	High-speed mode		Middle-speed mode		Low-speed mode		Unit	Test Conditions
				Min.	Max.	Min.	Max.	Min.	Max.		
SCKp cycle time	$t_{KCY1} \geq 4/PCLKB$	$4.0\text{ V} \leq VCC \leq 5.5\text{ V}$, $2.7\text{ V} \leq Vb \leq 4.0\text{ V}$, $C_b = 30\text{ pF}$, $R_b = 1.4\text{ k}\Omega$	t_{KCY1}	300	—	300	—	2300	—	ns	Figure 27.23 Figure 27.24
		$2.7\text{ V} \leq VCC < 4.0\text{ V}$, $2.3\text{ V} \leq Vb \leq 2.7\text{ V}$, $C_b = 30\text{ pF}$, $R_b = 2.7\text{ k}\Omega$		500	—	500	—	2300	—	ns	
		$1.8\text{ V} \leq VCC < 3.3\text{ V}$, $1.6\text{ V} \leq Vb \leq 2.0\text{ V}^{*1}$, $C_b = 30\text{ pF}$, $R_b = 5.5\text{ k}\Omega$		1150	—	1150	—	2300	—	ns	
SCKp high-level width		$4.0\text{ V} \leq VCC \leq 5.5\text{ V}$, $2.7\text{ V} \leq Vb \leq 4.0\text{ V}$, $C_b = 30\text{ pF}$, $R_b = 1.4\text{ k}\Omega$	t_{KH1}	$t_{KCY1}/2 - 75$	—	$t_{KCY1}/2 - 75$	—	$t_{KCY1}/2 - 75$	—	ns	
		$2.7\text{ V} \leq VCC < 4.0\text{ V}$, $2.3\text{ V} \leq Vb \leq 2.7\text{ V}$, $C_b = 30\text{ pF}$, $R_b = 2.7\text{ k}\Omega$		$t_{KCY1}/2 - 170$	—	$t_{KCY1}/2 - 170$	—	$t_{KCY1}/2 - 170$	—	ns	
		$1.8\text{ V} \leq VCC < 3.3\text{ V}$, $1.6\text{ V} \leq Vb \leq 2.0\text{ V}^{*1}$, $C_b = 30\text{ pF}$, $R_b = 5.5\text{ k}\Omega$		$t_{KCY1}/2 - 458$	—	$t_{KCY1}/2 - 458$	—	$t_{KCY1}/2 - 458$	—	ns	
SCKp low-level width		$4.0\text{ V} \leq VCC \leq 5.5\text{ V}$, $2.7\text{ V} \leq Vb \leq 4.0\text{ V}$, $C_b = 30\text{ pF}$, $R_b = 1.4\text{ k}\Omega$	t_{KL1}	$t_{KCY1}/2 - 12$	—	$t_{KCY1}/2 - 12$	—	$t_{KCY1}/2 - 50$	—	ns	
		$2.7\text{ V} \leq VCC < 4.0\text{ V}$, $2.3\text{ V} \leq Vb \leq 2.7\text{ V}$, $C_b = 30\text{ pF}$, $R_b = 2.7\text{ k}\Omega$		$t_{KCY1}/2 - 18$	—	$t_{KCY1}/2 - 18$	—	$t_{KCY1}/2 - 50$	—	ns	
		$1.8\text{ V} \leq VCC < 3.3\text{ V}$, $1.6\text{ V} \leq Vb \leq 2.0\text{ V}^{*1}$, $C_b = 30\text{ pF}$, $R_b = 5.5\text{ k}\Omega$		$t_{KCY1}/2 - 50$	—	$t_{KCY1}/2 - 50$	—	$t_{KCY1}/2 - 50$	—	ns	

Note 1. Use this setting with $VCC \geq Vb$.

Note: Select the TTL input buffer for the Slp pin and the N-ch open drain output [withstand voltage of VCC] mode for the Sop pin and SCKp pin by using the Port gh Pin Function Select Register (PghPFS_A.PIM and PghPFS_A.NCODR). For V_{IH} and V_{IL} , see the DC characteristics with the TTL input buffer selected.

要设置此工作时钟，请使用串行模式寄存器 mn (SMRmn) 中的 CKSmn 位。
m：单元编号，n：通道编号(mn = 00)

表 27.29 在主模式下，简化的 SPI 通信中，设备以不同的电压电平（1.8 V、2.5 V、或3 V），内部 SCKp 时钟 (1) 运行。

条件：VCC = 1.8至5.5 V, VSS = 0 V, Ta = -40 to +125°C

范围			象征	高速模式		中速模式		低速模式		单元	测试状况
				最小。	最大限	最小。	最大限	最小。	最大限		
SCKp 循环时间	$t_{KCY1} \geq 4/PCLKB$	$4.0\text{ V} \leq VCC \leq 5.5\text{ V}$, $2.7\text{ V} \leq Vb \leq 4.0\text{ V}$, $C_b = 30\text{ pF}$, $R_b = 1.4\text{ k}\Omega$	t_{KCY1}	300	度。	300	度。	2300	度。	ns	数字 27.23 数字 27.24
		$2.7\text{ V} \leq VCC < 4.0\text{ V}$, $2.3\text{ V} \leq Vb \leq 2.7\text{ V}$, $C_b = 30\text{ pF}$, $R_b = 2.7\text{ k}\Omega$		500	—	500	—	2300	—	ns	
		$1.8\text{ V} \leq VCC < 3.3\text{ V}$, $1.6\text{ V} \leq Vb \leq 2.0\text{ V}^{*1}$, $C_b = 30\text{ pF}$, $R_b = 5.5\text{ k}\Omega$		1150	—	1150	—	2300	—	ns	
SCKp 高级别宽度		$4.0\text{ V} \leq VCC \leq 5.5\text{ V}$, $2.7\text{ V} \leq Vb \leq 4.0\text{ V}$, $C_b = 30\text{ pF}$, $R_b = 1.4\text{ k}\Omega$	t_{KH1}	$t_{KCY1}/2 - 75$	—	$t_{KCY1}/2 - 75$	—	$t_{KCY1}/2 - 75$	—	ns	
		$2.7\text{ V} \leq VCC < 4.0\text{ V}$, $2.3\text{ V} \leq Vb \leq 2.7\text{ V}$, $C_b = 30\text{ pF}$, $R_b = 2.7\text{ k}\Omega$		$t_{KCY1}/2 - 170$	—	$t_{KCY1}/2 - 170$	—	$t_{KCY1}/2 - 170$	—	ns	
		$1.8\text{ V} \leq VCC < 3.3\text{ V}$, $1.6\text{ V} \leq Vb \leq 2.0\text{ V}^{*1}$, $C_b = 30\text{ pF}$, $R_b = 5.5\text{ k}\Omega$		$t_{KCY1}/2 - 458$	—	$t_{KCY1}/2 - 458$	—	$t_{KCY1}/2 - 458$	—	ns	
SCKp 低水平宽度		$4.0\text{ V} \leq VCC \leq 5.5\text{ V}$, $2.7\text{ V} \leq Vb \leq 4.0\text{ V}$, $C_b = 30\text{ pF}$, $R_b = 1.4\text{ k}\Omega$	t_{KL1}	$t_{KCY1}/2 - 12$	—	$t_{KCY1}/2 - 12$	—	$t_{KCY1}/2 - 50$	—	ns	
		$2.7\text{ V} \leq VCC < 4.0\text{ V}$, $2.3\text{ V} \leq Vb \leq 2.7\text{ V}$, $C_b = 30\text{ pF}$, $R_b = 2.7\text{ k}\Omega$		$t_{KCY1}/2 - 18$	—	$t_{KCY1}/2 - 18$	—	$t_{KCY1}/2 - 50$	—	ns	
		$1.8\text{ V} \leq VCC < 3.3\text{ V}$, $1.6\text{ V} \leq Vb \leq 2.0\text{ V}^{*1}$, $C_b = 30\text{ pF}$, $R_b = 5.5\text{ k}\Omega$		$t_{KCY1}/2 - 50$	—	$t_{KCY1}/2 - 50$	—	$t_{KCY1}/2 - 50$	—	ns	

注1. 将此设置与 $VCC \geq Vb$ 一起使用。

注意：使用端口gh引脚功能选择寄存器（PghPFS_A.PIM和PghPFS_A.NCODR）为Slp 引脚选择TTL输入缓冲器，为N引脚选择VCC耐压模式，为[引脚和] 引脚选择VCC耐压模式。对于 V_{IH} 和 V_{IL} 引脚，请参阅选择TTL输入缓冲器后的直流特性。

Table 27.30 In simplified SPI communications in the master mode with devices operating at different voltage levels (1.8 V, 2.5 V, or 3 V) with the internal SCKp clock (2)

Conditions: VCC = 1.8 to 5.5 V, VSS = 0 V, Ta = -40 to +125°C

Parameter	Symbol	High-speed mode		Middle-speed mode		Low-speed mode		Unit	Test Conditions
		Min.	Max.	Min.	Max.	Min.	Max.		
Slp setup time (to SCKp↑)*1	4.0 V ≤ VCC ≤ 5.5 V, 2.7 V ≤ Vb ≤ 4.0 V, Cb = 30 pF, Rb = 1.4 kΩ	81	—	81	—	479	—	ns	Figure 27.23 Figure 27.24
	2.7 V ≤ VCC < 4.0 V, 2.3 V ≤ Vb ≤ 2.7 V, Cb = 30 pF, Rb = 2.7 kΩ	177	—	177	—	479	—	ns	
	1.8 V ≤ VCC < 3.3 V, 1.6 V ≤ Vb ≤ 2.0 V*2, Cb = 30 pF, Rb = 5.5 kΩ	479	—	479	—	479	—	ns	
Slp hold time (from SCKp↑)*1	4.0 V ≤ VCC ≤ 5.5 V, 2.7 V ≤ Vb ≤ 4.0 V, Cb = 30 pF, Rb = 1.4 kΩ	19	—	19	—	19	—	ns	
	2.7 V ≤ VCC < 4.0 V, 2.3 V ≤ Vb ≤ 2.7 V, Cb = 30 pF, Rb = 2.7 kΩ	19	—	19	—	19	—	ns	
	1.8 V ≤ VCC < 3.3 V, 1.6 V ≤ Vb ≤ 2.0 V*2, Cb = 30 pF, Rb = 5.5 kΩ	19	—	19	—	19	—	ns	
Delay time from SCKp↓ to SOp output*1	4.0 V ≤ VCC ≤ 5.5 V, 2.7 V ≤ Vb ≤ 4.0 V, Cb = 30 pF, Rb = 1.4 kΩ	—	100	—	100	—	100	ns	
	2.7 V ≤ VCC < 4.0 V, 2.3 V ≤ Vb ≤ 2.7 V, Cb = 30 pF, Rb = 2.7 kΩ	—	195	—	195	—	195	ns	
	1.8 V ≤ VCC < 3.3 V, 1.6 V ≤ Vb ≤ 2.0 V*2, Cb = 30 pF, Rb = 5.5 kΩ	—	483	—	483	—	483	ns	

Note 1. This setting applies when SCRmn.DCP[1:0] = 00b or 11b.

Note 2. Use this setting with VCC ≥ Vb.

Note: Select the TTL input buffer for the Slp pin and the N-ch open drain output [withstand voltage of VCC] mode for the SOp pin and SCKp pin by using the Port gh Pin Function Select Register (PghPFS_A.PIM and PghPFS_A.NCODR). For VIH and VIL, see the DC characteristics with the TTL input buffer selected.

表 27.30 在主模式下，简化的 SPI 通信中，设备以不同的电压电平（1.8 V、2.5 V, 或3 V），内部 SCKp 时钟 (2) 运行

条件: VCC = 1.8至5.5 V, VSS= 0 V, Ta = -40 至+125°C

范围		象征	高速模式		中速模式		低速模式		单元	测试状况
			最小。	最大限度。	最小。	最大限度。	最小。	最大限度。		
Slp 建立时间 (到 SCKp↑) *1	4.0 V ≤ VCC ≤ 5.5 V, 2.7V≤ Vb≤ 4.0V, Cb= 30 pF, Rb= 1.4 kΩ	tSIK1	81	—	81	—	479	—	ns	图 27.23 图 27.24
	2.7 V ≤ VCC < 4.0 V, 2.3V≤ Vb≤ 2.7V, Cb= 30 pF, Rb= 2.7 kΩ		177	—	177	—	479	—	ns	
	1.8 V ≤ VCC < 3.3 V, 1.6V≤ Vb≤ 2.0V*2, Cb= 30 pF, Rb= 5.5 kΩ		479	—	479	—	479	—	ns	
Slp 保持时间 (来自SCKp↑)*1	4.0 V ≤ VCC ≤ 5.5 V, 2.7 V ≤ Vb ≤ 4.0 V, Cb = 30pF, Rb = 1.4 kΩ	tKS11	19	—	19	—	19	—	ns	
	2.7 V ≤ VCC < 4.0 V, 2.3V≤ Vb≤ 2.7V, Cb= 30 pF, Rb = 2.7 kΩ		19	—	19	—	19	—	ns	
	1.8 V ≤ VCC < 3.3 V, 1.6V≤ Vb≤ 2.0V*2, Cb = 30pF, Rb = 5.5 kΩ		19	—	19	—	19	—	ns	
从SCKp↓ 到 SOp 输出的延迟时间*1	4.0 V ≤ VCC ≤ 5.5 V, 2.7V≤ Vb≤ 4.0V, Cb = 30pF, Rb = 1.4 kΩ	tKSO1	—	100	—	100	—	100	ns	
	2.7 V ≤ VCC < 4.0 V, 2.3V≤ Vb≤ 2.7V, Cb= 30 pF, Rb = 2.7 kΩ		—	195	—	195	—	195	ns	
	1.8 V ≤ VCC < 3.3 V, 1.6V≤ Vb≤ 2.0V*2, Cb = 30pF, Rb = 5.5 kΩ		—	483	—	483	—	483	ns	

注 1. 此设置适用于 SCRmn.DCP[1:0] = 00b 或 11b。

注2. 将此设置与VCC ≥ Vb一起使用。

注意: 使用端口gh引脚功能选择寄存器（PghPFS_A.PIM和PghPFS_A.NCODR）为Slp 引脚选择TTL输入缓冲器，为N引脚选择VCC耐压模式，为[引脚和] 引脚选择VCC耐压模式。对于VIH和VIL引脚，请参阅选择TTL输入缓冲器后的直流特性。

Table 27.31 In simplified SPI communications in the master mode with devices operating at different voltage levels (1.8 V, 2.5 V, or 3 V) with the internal SCKp clock (3)

Conditions: VCC = 1.8 to 5.5 V, VSS = 0 V, Ta = -40 to +125°C

Parameter	Symbol	High-speed mode		Middle-speed mode		Low-speed mode		Unit	Test Conditions
		Min.	Max.	Min.	Max.	Min.	Max.		
Slp setup time (to SCKp↓) *1	4.0 V ≤ VCC ≤ 5.5 V, 2.7 V ≤ Vb ≤ 4.0 V, Cb = 30 pF, Rb = 1.4 kΩ	44	—	44	—	110	—	ns	Figure 27.23 Figure 27.24
	2.7 V ≤ VCC < 4.0 V, 2.3 V ≤ Vb ≤ 2.7 V, Cb = 30 pF, Rb = 2.7 kΩ	44	—	44	—	110	—	ns	
	1.8 V ≤ VCC < 3.3 V, 1.6 V ≤ Vb ≤ 2.0 V*2, Cb = 30 pF, Rb = 5.5 kΩ	110	—	110	—	110	—	ns	
Slp hold time (from SCKp↓) *1	4.0 V ≤ VCC ≤ 5.5 V, 2.7 V ≤ Vb ≤ 4.0 V, Cb = 30 pF, Rb = 1.4 kΩ	19	—	19	—	19	—	ns	
	2.7 V ≤ VCC < 4.0 V, 2.3 V ≤ Vb ≤ 2.7 V, Cb = 30 pF, Rb = 2.7 kΩ	19	—	19	—	19	—	ns	
	1.8 V ≤ VCC < 3.3 V, 1.6 V ≤ Vb ≤ 2.0 V*2, Cb = 30 pF, Rb = 5.5 kΩ	19	—	19	—	19	—	ns	
Delay time from SCKp↑ to SOp output*1	4.0 V ≤ VCC ≤ 5.5 V, 2.7 V ≤ Vb ≤ 4.0 V, Cb = 30 pF, Rb = 1.4 kΩ	—	25	—	25	—	25	ns	
	2.7 V ≤ VCC < 4.0 V, 2.3 V ≤ Vb ≤ 2.7 V, Cb = 30 pF, Rb = 2.7 kΩ	—	25	—	25	—	25	ns	
	1.8 V ≤ VCC < 3.3 V, 1.6 V ≤ Vb ≤ 2.0 V*2, Cb = 30 pF, Rb = 5.5 kΩ	—	25	—	25	—	25	ns	

Note 1. This setting applies when SCRmn.DCP[1:0] = 01b or 10b.
Note 2. Use this setting with VCC ≥ Vb.

Note: Select the TTL input buffer for the Slp pin and the N-ch open drain output [withstand voltage of VCC] mode for the SOp pin and SCKp pin by using the Port gh Pin Function Select Register (PghPFS_A.PIM and PghPFS_A.NCODR). For VIH and VIL, see the DC characteristics with the TTL input buffer selected.

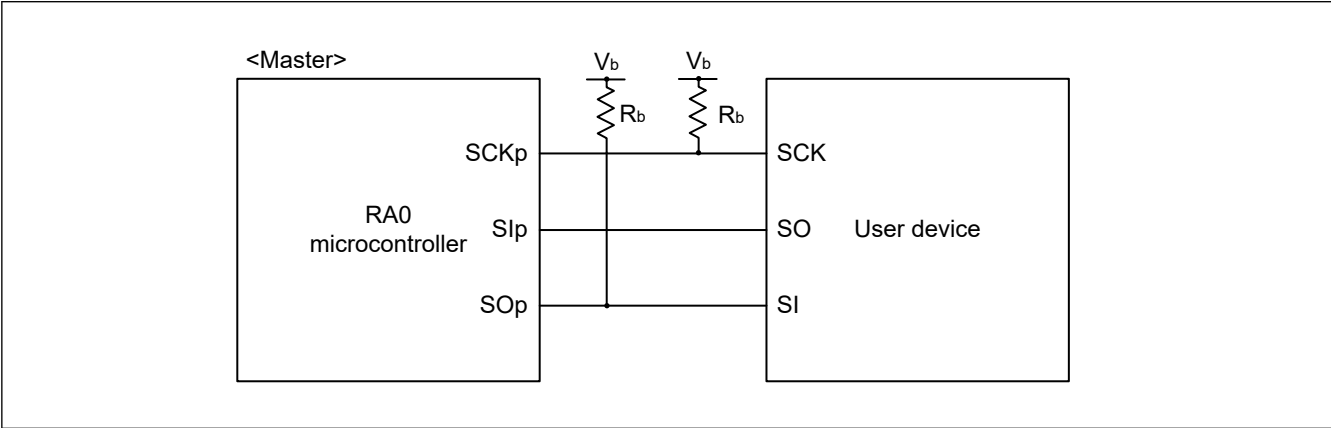


Figure 27.22 Connection in the simplified SPI communications with devices operating at different voltage levels

Note: ● Rb[Ω]: Communication line (SCKp, SOp) pull-up resistance, Cb[F]: Communication line (SCKp, SOp) load capacitance, Vb[V]: Communication line voltage

表 27.31 在主模式下，简化的 SPI 通信中，设备以不同的电压电平（1.8 V、2.5 V, 或 3 V），内部 SCKp 时钟（3）运行。

条件: VCC = 1.8至5.5 V, VSS= 0 V, Ta = -40 至+125°C

范围	象征	高速模式		中速模式		低速模式		单元	测试状况
		最小。	最大限度。	最小。	最大限度。	最小。	最大限度。		
Slp 设置时间 (至 SCKp↓) *1	4.0 V ≤ VCC ≤ 5.5 V, 2.7V≤ Vb≤ 4.0V, Cb= 30 pF, Rb= 1.4 kΩ	44	—	44	—	110	—	ns	图 27.23 图 27.24
	2.7 V ≤ VCC < 4.0 V, 2.3V≤ Vb≤ 2.7V, Cb= 30 pF, Rb= 2.7 kΩ	44	—	44	—	110	—	ns	
	1.8 V ≤ VCC < 3.3 V, 1.6 V ≤ Vb ≤ 2.0 V*2, Cb= 30 pF, Rb= 5.5 kΩ	110	—	110	—	110	—	ns	
Slp 保持时间 (来自 SCKp↓) *1	4.0 V ≤ VCC ≤ 5.5 V, 2.7 V ≤ Vb ≤ 4.0 V, Cb = 30pF, Rb = 1.4 kΩ	19	—	19	—	19	—	ns	
	2.7 V ≤ VCC < 4.0 V, 2.3V≤ Vb≤ 2.7V, Cb= 30 pF, Rb = 2.7 kΩ	19	—	19	—	19	—	ns	
	1.8 V ≤ VCC < 3.3 V, 1.6V≤ Vb≤ 2.0V*2, Cb= 30 pF, Rb = 5.5 kΩ	19	—	19	—	19	—	ns	
从SCKp↑到SOp输出的延迟时间*1	4.0 V ≤ VCC ≤ 5.5 V, 2.7V≤ Vb≤ 4.0V, Cb = 30pF, Rb = 1.4 kΩ	—	25	—	25	—	25	ns	
	2.7 V ≤ VCC < 4.0 V, 2.3V≤ Vb≤ 2.7V, Cb= 30 pF, Rb = 2.7 kΩ	—	25	—	25	—	25	ns	
	1.8 V ≤ VCC < 3.3 V, 1.6V≤ Vb≤ 2.0V*2, Cb= 30 pF, Rb = 5.5 kΩ	—	25	—	25	—	25	ns	

注 1. 此设置适用于 SCRmn.DCP[1:0] = 01b 或 10b。
注2. 将此设置与VCC ≥ Vb一起使用。

注意: 使用端口gh引脚功能选择寄存器（PghPFS_A.PIM和PghPFS_A.NCODR）为Slp 引脚选择TTL输入缓冲器，为N引脚选择VCC耐压模式，为[引脚和] 引脚选择VCC耐压模式。对于VIH和VIL引脚，请参阅选择TTL输入缓冲器后的直流特性。

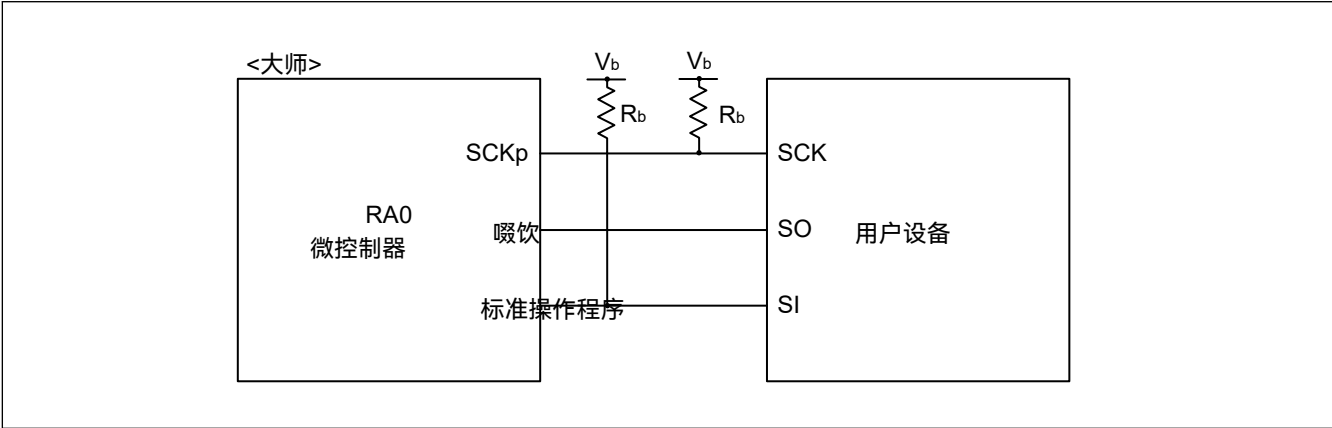


图 27.22 简化的 SPI 通信中与不同电压等级设备的连接

注: ●Rb[Ω]: 通信线（SCKp、SOP）上拉电阻，Cb[F]: 通信线（SCKp、SOP）负载电容，Vb[V]: 通信线电压●p:

- p: Simplified SPI number (p = 00, 11), m: Unit number, n: Channel number (mn = 00, 03), gh: Port number (gh = 100 to 102, 108 to 110, 112, 201, 212, 213)
- f_{MCK}: Serial array unit operation clock frequency
To set this operating clock, use the CKS bit in the serial mode register mn (SMRmn).
m: Unit number, n: Channel number (mn = 00, 03)
- Communications by using P212 and P213 with devices operating at different voltage levels are not possible since P212PFS_A and P213PFS_A registers do not have PIM bit.

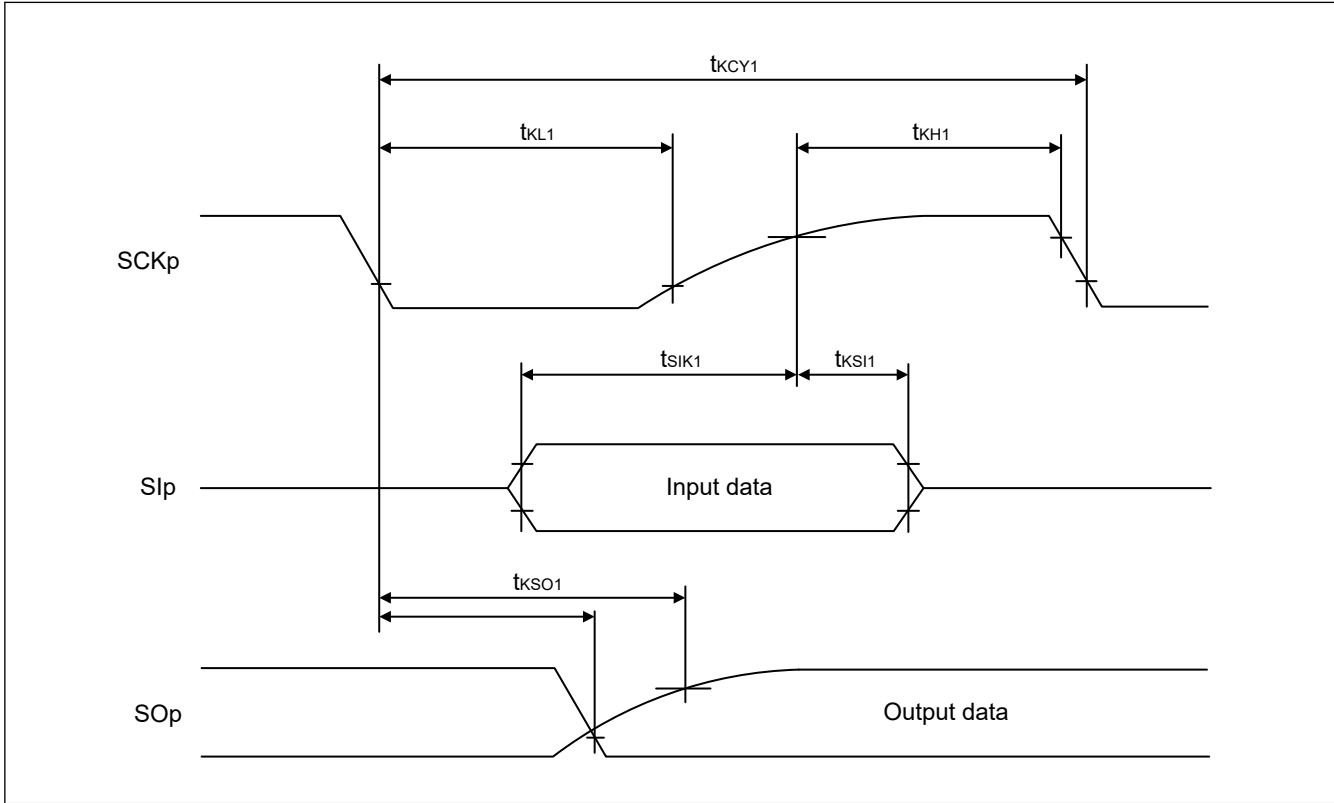


Figure 27.23 Timing of serial transfer in the simplified SPI communications in the master mode with devices operating at different voltage levels when SCRmn.DCP[1:0] = 00b or 11b

- 简化的 SPI 编号(p = 00, 11), m: 单元号, n: 通道号(mn = 00, 03), gh: 端口号(gh = 100 至 102, 108 至 110, 112, 201, 212, 213)
- f_{MCK}: 串行阵列单元操作时钟频率
要设置此工作时钟, 请使用串行模式寄存器 mn (SMRmn) 中的 CKS 位。
m: 单元编号, n: 通道编号(mn = 00, 03)
 - 由于 P212PFS_A 和 P213PFS_A 寄存器没有 PIM 位, 因此无法使用 P212 和 P213 与工作电压不同的设备进行通信。

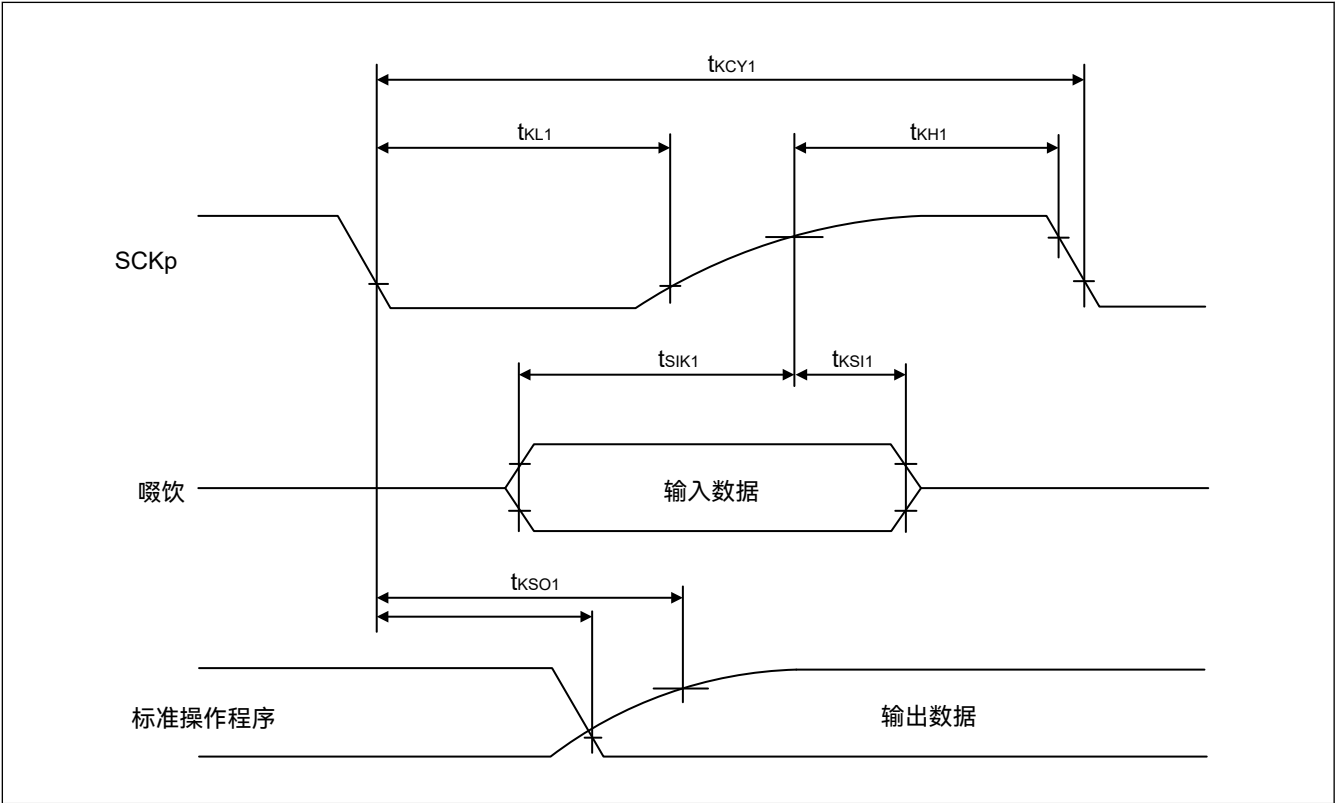


图 27.23 简化 SPI 通信中, 主模式下, 当 SCRmn.DCP[1:0] = 00b 或 11b 时, 器件工作在不同电压电平下的串行传输时序

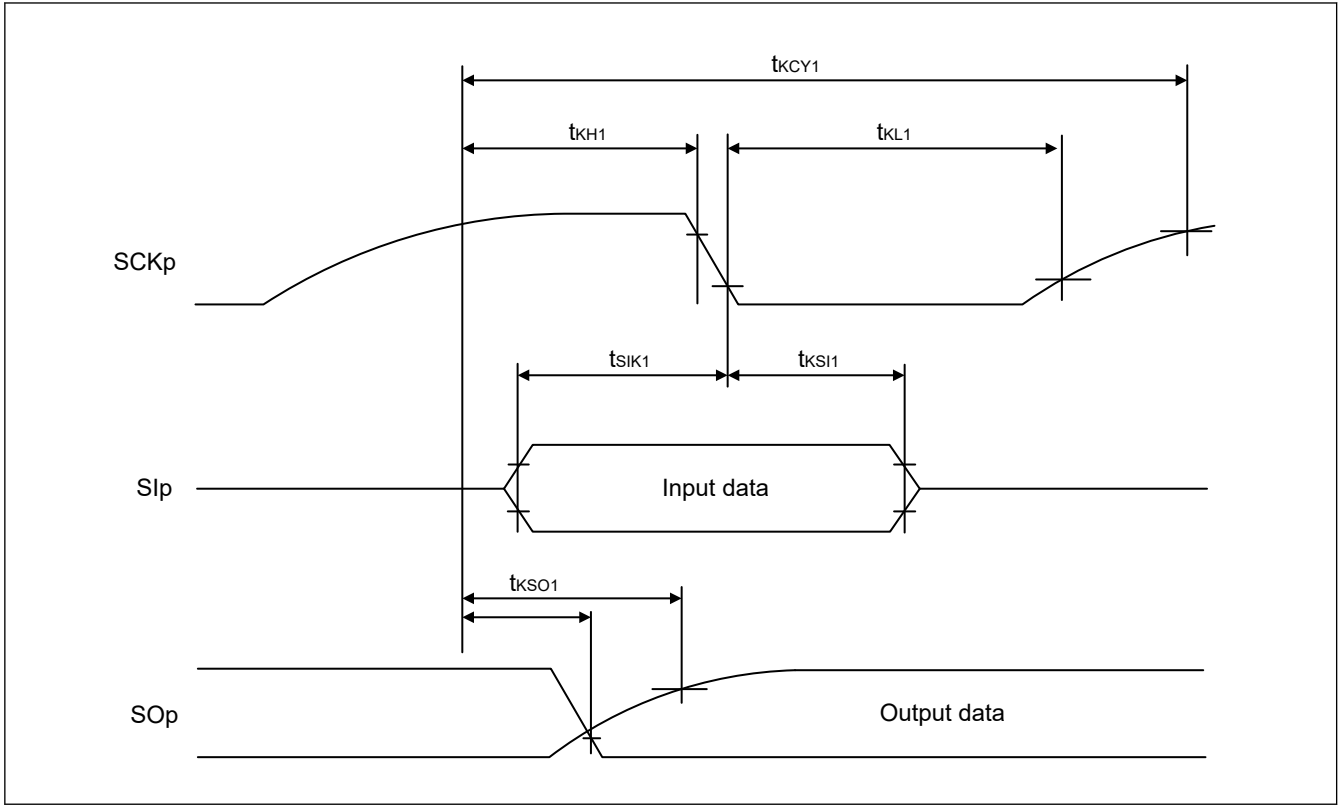


Figure 27.24 Timing of serial transfer in the simplified SPI communications in the master mode with devices operating at different voltage levels when SCRmn.DCP[1:0] = 01b or 10b

Note:

- p: Simplified SPI number (p = 00, 11), m: Unit number, n: Channel number (mn = 00, 03), gh: Port number (gh = 100 to 102, 108 to 110, 112, 201, 212, 213)
- Communications by using P212 and P213 with devices operating at different voltage levels are not possible since P212PFS_A and P213PFS_A registers do not have PIM bit.

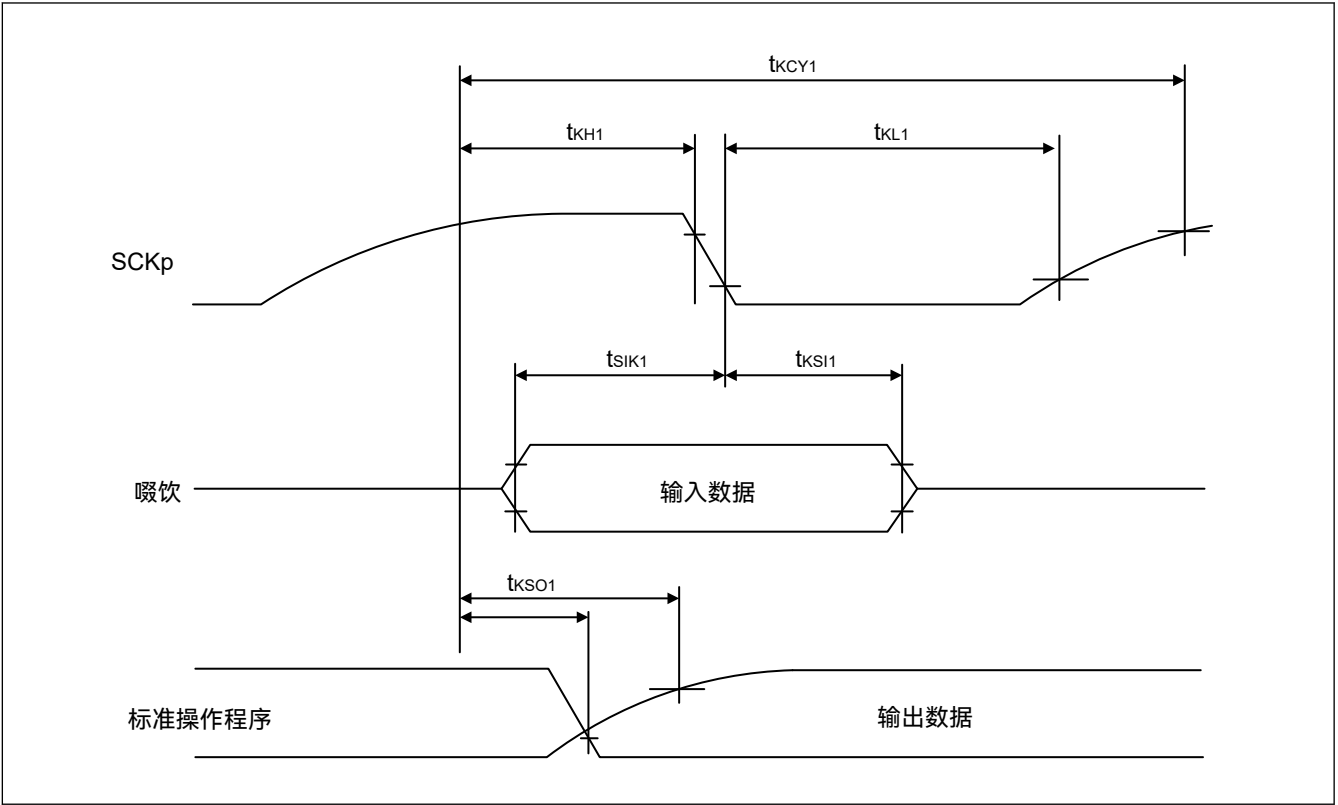


图 27.24 简化 SPI 通信中，主模式下，当 SCRmn.DCP[1:0] = 01b 或 10b 时，器件工作在不同电压电平下的串行传输时序

注: ●p: 简化的SPI编号(p = 00, 11) , m: 单元编号, n: 通道编号(mn = 00, 03) , gh: 端口编号(gh = (100至102, 108至110, 112, 201, 212, 213)

●由于 P212PFS_A 和 P213PFS_A 寄存器没有 PIM 位，因此无法使用 P212 和 P213 与工作电压不同的设备进行通信。

Table 27.32 In simplified SPI communications in the slave mode with devices operating at different voltage levels (1.8 V, 2.5 V, or 3 V) with the external SCKp clock

Conditions: VCC = 1.8 to 5.5 V, VSS = 0 V, Ta = -40 to +125°C

Parameter		Symbol	High-speed mode		Middle-speed mode		Low-speed mode		Unit	Test Conditions
			Min.	Max.	Min.	Max.	Min.	Max.		
SCKp cycle time*1	4.0 V ≤ VCC ≤ 5.5 V, 2.7 V ≤ Vb ≤ 4.0 V	fMCK = 32 MHz	14/fMCK	—	—	—	—	—	ns	Figure 27.26 Figure 27.27
		8 MHz < fMCK ≤ 16 MHz	10/fMCK	—	10/fMCK	—	—	—	ns	
		4 MHz < fMCK ≤ 8 MHz	8/fMCK	—	8/fMCK	—	—	—	ns	
		fMCK ≤ 4 MHz	6/fMCK	—	6/fMCK	—	10/fMCK	—	ns	
	2.7 V ≤ VCC < 4.0 V, 2.3 V ≤ Vb ≤ 2.7 V	fMCK = 32 MHz	20/fMCK	—	—	—	—	—	ns	
		8 MHz < fMCK ≤ 16 MHz	12/fMCK	—	12/fMCK	—	—	—	ns	
		4 MHz < fMCK ≤ 8 MHz	8/fMCK	—	8/fMCK	—	—	—	ns	
		fMCK ≤ 4 MHz	6/fMCK	—	6/fMCK	—	10/fMCK	—	ns	
	1.8 V ≤ VCC < 3.3 V, 1.6 V ≤ Vb ≤ 2.0 V*2	fMCK = 32 MHz	48/fMCK	—	—	—	—	—	ns	
		8 MHz < fMCK ≤ 16 MHz	26/fMCK	—	26/fMCK	—	—	—	ns	
		4 MHz < fMCK ≤ 8 MHz	16/fMCK	—	16/fMCK	—	—	—	ns	
		fMCK ≤ 4 MHz	10/fMCK	—	10/fMCK	—	10/fMCK	—	ns	
SCKp high-/low-level width	4.0 V ≤ VCC ≤ 5.5 V, 2.7 V ≤ Vb ≤ 4.0 V	tKH2, tKL2	tKCY2/2 - 12	—	tKCY2/2 - 12	—	tKCY2/2 - 50	—	ns	
	2.7 V ≤ VCC < 4.0 V, 2.3 V ≤ Vb ≤ 2.7 V		tKCY2/2 - 18	—	tKCY2/2 - 18	—	tKCY2/2 - 50	—	ns	
	1.8 V ≤ VCC < 3.3 V, 1.6 V ≤ Vb ≤ 2.0 V*2		tKCY2/2 - 50	—	tKCY2/2 - 50	—	tKCY2/2 - 50	—	ns	
Slp setup time (to SCKp↑)*3	4.0 V ≤ VCC ≤ 5.5 V, 2.7 V ≤ Vb ≤ 4.0 V	tSIK2	1/fMCK + 20	—	1/fMCK + 20	—	1/fMCK + 30	—	ns	
	2.7 V ≤ VCC < 4.0 V, 2.3 V ≤ Vb ≤ 2.7 V		1/fMCK + 20	—	1/fMCK + 20	—	1/fMCK + 30	—	ns	
	1.8 V ≤ VCC < 3.3 V, 1.6 V ≤ Vb ≤ 2.0 V*2		1/fMCK + 30	—	1/fMCK + 30	—	1/fMCK + 30	—	ns	
Slp hold time (from SCKp↑)*3		tKSI2	1/fMCK + 31	—	1/fMCK + 31	—	1/fMCK + 31	—	ns	
Delay time from SCKp↓ to SOp output*4	4.0 V ≤ VCC ≤ 5.5 V, 2.7 V ≤ Vb ≤ 4.0 V, Cb = 30 pF, Rb = 1.4 kΩ	tKSO2	—	2/fMCK + 120	—	2/fMCK + 120	—	2/fMCK + 573	ns	
	2.7 V ≤ VCC < 4.0 V, 2.3 V ≤ Vb ≤ 2.7 V, Cb = 30 pF, Rb = 2.7 kΩ		—	2/fMCK + 214	—	2/fMCK + 214	—	2/fMCK + 573	ns	
	1.8 V ≤ VCC < 3.3 V, 1.6 V ≤ Vb ≤ 2.0 V*2, Cb = 30 pF, Rb = 5.5 kΩ		—	2/fMCK + 573	—	2/fMCK + 573	—	2/fMCK + 573	ns	

Note 1. Transfer rate in the Snooze mode: 1 Mbps (max.)

Note 2. Use this setting with VCC ≥ Vb.

Note 3. This setting applies when SCRmn.DCP[1:0] = 00b or 11b. The Slp setup time becomes to SCKp↓ and Slp hold time becomes from SCKp↓ when SCRmn.DCP[1:0] = 01b or 10b.

Note 4. This setting applies when SCRmn.DCP[1:0] = 00b or 11b. The delay time to SOp output becomes from SCKp↑ when SCRmn.DCP[1:0] = 01b or 10b.

Note: Select the TTL input buffer for the Slp pin and the N-ch open drain output [withstand voltage of VCC] mode for the SOp pin and SCKp pin by using the Port gh Pin Function Select Register (PghPFS_A.PIM and PghPFS_A.NCODR). For VIH and VIL, see the DC characteristics with the TTL input buffer selected.

表 27.32 简化的 SPI 从模式通信，设备工作电压各不相同
使用外部 SCKp 时钟时，电压为 1.8 V、2.5 V 或 3 V

条件: VCC= 10.8 至 5.5 V, VSS= 0 V,Ta= -40 至+125°C

范围			象征	高速模式		中速模式		低速模式		单元	测试状况	
				最小。	最大限度。	最小。	最大限度。	最小。	最大限度。			
SCKp 循环时间*1	4.0 V ≤ VCC ≤ 5.5 V, 2.7V≤ Vb≤ 4.0V	fMCK = 32 MHz	tKCY2	14/fMCK	—	—	—	—	—	ns	图 27.26 图 27.27	
		8 MHz < fMCK ≤ 16 MHz		10/fMCK	—	10/fMCK	—	—	—	ns		
		4 MHz < fMCK ≤ 8 MHz		8/fMCK	—	8/fMCK	—	—	—	ns		
		fMCK ≤ 4 MHz		6/fMCK	—	6/fMCK	—	10/fMCK	—	ns		
	2.7 V ≤ VCC < 4.0 V, 2.3 V ≤ Vb ≤ 2.7 V	fMCK = 32 MHz		20/fMCK	—	—	—	—	—	—		ns
		8 MHz < fMCK ≤ 16 MHz		12/fMCK	—	12/fMCK	—	—	—	—		ns
		4 MHz < fMCK ≤ 8 MHz		8/fMCK	—	8/fMCK	—	—	—	—		ns
		fMCK ≤ 4 MHz		6/fMCK	—	6/fMCK	—	10/fMCK	—	ns		
	1.8 V ≤ VCC < 3.3 V, 1.6 V ≤ Vb ≤ 2.0 V*2	fMCK = 32 MHz		48/fMCK	—	—	—	—	—	—		ns
		8 MHz < fMCK ≤ 16 MHz		26/fMCK	—	26/fMCK	—	—	—	—		ns
		4 MHz < fMCK ≤ 8 MHz		16/fMCK	—	16/fMCK	—	—	—	—		ns
		fMCK ≤ 4 MHz		10/fMCK	—	10/fMCK	—	10/fMCK	—	ns		
SCKp 高/低电平宽度	4.0 V ≤ VCC ≤ 5.5 V, 2.7 V ≤ Vb ≤ 4.0 V	tKCY2, tKL2	tKCY2/2 - 12	—	tKCY2/2 - 12	—	tKCY2/2 - 50	—	ns			
	2.7 V ≤ VCC < 4.0 V, 2.3V≤ Vb≤ 2.7V		tKCY2/2 - 18	—	tKCY2/2 - 18	—	tKCY2/2 - 50	—	ns			
	1.8 V ≤ VCC < 3.3 V, 1.6V≤ Vb≤ 2.0V*2		tKCY2/2 - 50	—	tKCY2/2 - 50	—	tKCY2/2 - 50	—	ns			
Slp 设置时间 (至SCKp↑)*3	4.0 V ≤ VCC ≤ 5.5 V, 2.7 V ≤ Vb ≤ 4.0 V	tSIK2	1/fMCK + 20	—	1/fMCK + 20	—	1/fMCK + 30	—	ns			
	2.7 V ≤ VCC < 4.0 V, 2.3V≤ Vb≤ 2.7V		1/fMCK + 20	—	1/fMCK + 20	—	1/fMCK + 30	—	ns			
	1.8 V ≤ VCC < 3.3 V, 1.6V≤ Vb≤ 2.0V*2		1/fMCK + 30	—	1/fMCK + 30	—	1/fMCK + 30	—	ns			
啜饮保持时间 (来自 SCKp↑)*3		tKSI2	1/fMCK + 31	—	1/fMCK + 31	—	1/fMCK + 31	—	ns			
从SCKp↓到SOp输出的延迟时间*4	4.0 V ≤ VCC ≤ 5.5 V, 2.7V≤ Vb≤ 4.0V, Cb= 30 pF, Rb = 1.4 kΩ	tKSO2	—	2/fMCK + 120	—	2/fMCK + 120	—	2/fMCK + 573	ns			
	2.7 V ≤ VCC < 4.0 V, 2.3V≤ Vb≤ 2.7V, Cb= 30 pF, Rb = 2.7 kΩ		—	2/fMCK + 214	—	2/fMCK + 214	—	2/fMCK + 573	ns			
	1.8 V ≤ VCC < 3.3 V, 1.6V≤ Vb≤ 2.0V*2, Cb = 30pF, Rb = 5.5 kΩ		—	2/fMCK + 573	—	2/fMCK + 573	—	2/fMCK + 573	ns			

注1：休眠模式下的传输速率：1 Mbps（最大值）

注2. 将此设置与VCC ≥ Vb一起使用。

注 3. 此设置适用于 SCRmn.DCP[1:0] = 00b 或 11b。当 SCRmn.DCP[1:0] = 01b 或 10b 时，Slp 的建立时间变为 SCKp↓，Slp 的保持时间变为 SCKp↓。

注 4. 此设置适用于 SCRmn.DCP[1:0] = 00b 或 11b。当 SCRmn.DCP[1:0] = 01b 或 10b 时，SOp 输出的延迟时间将变为 SCKp↑。

注意：使用端口gh引脚功能选择寄存器（PghPFS_A.PIM和PghPFS_A.NCODR）为Slp 引脚选择TTL输入缓冲器，并为[引脚选择N沟道开漏输出耐压模式（VCC），为SOp引脚和SCKp引脚选择N沟道开漏输出模式。对于VIH和VIL引脚，请参阅选择TTL输入缓冲器后的直流特性。

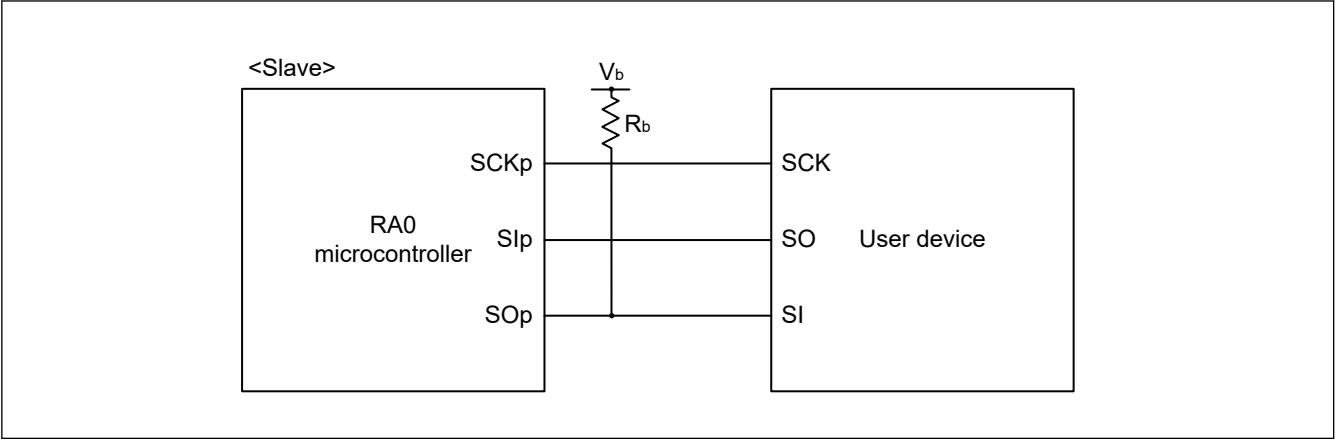


Figure 27.25 Connection in the simplified SPI communications with devices operating at different voltage levels

- Note:
- $R_b[\Omega]$: Communication line (SO_p) pull-up resistance, $C_b[F]$: Communication line (SO_p) load capacitance, $V_b[V]$: Communication line voltage
 - p: Simplified SPI number (p = 00, 11), m: Unit number, n: Channel number (mn = 00, 03), gh: Port number (gh = 100 to 102, 108 to 110, 112, 201, 212, 213)
 - f_{MCK} : Serial array unit operation clock frequency
To set this operating clock, use the CKS bit in the serial mode register mn (SMRmn).
m: Unit number, n: Channel number (mn = 00, 03, 10, 11)
 - Communications by using P212 and P213 with devices operating at different voltage levels are not possible since P212PFS_A and P213PFS_A registers do not have PIM bit.

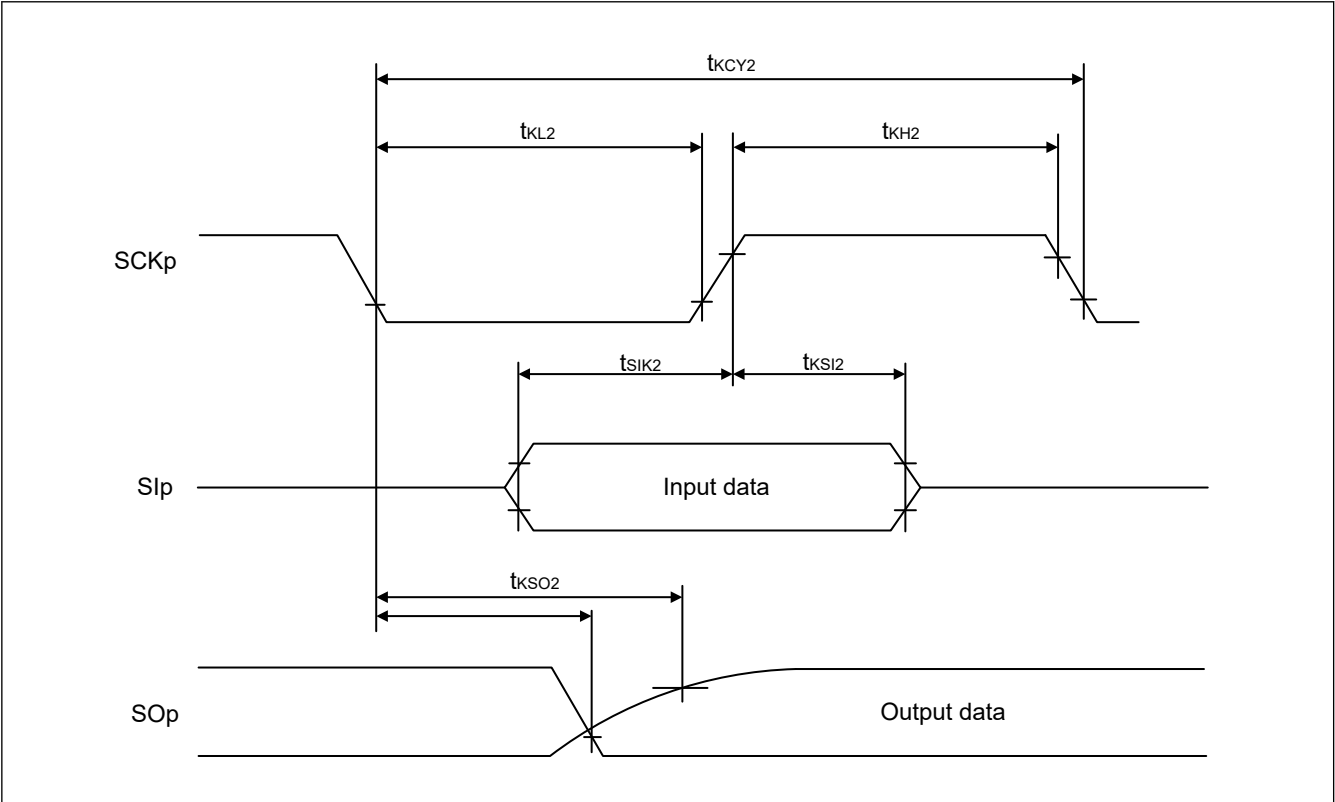


Figure 27.26 Timing of serial transfer in the simplified SPI communications in the slave mode with devices operating at different voltage levels when SCRmn.DCP[1:0] = 00b or 11b

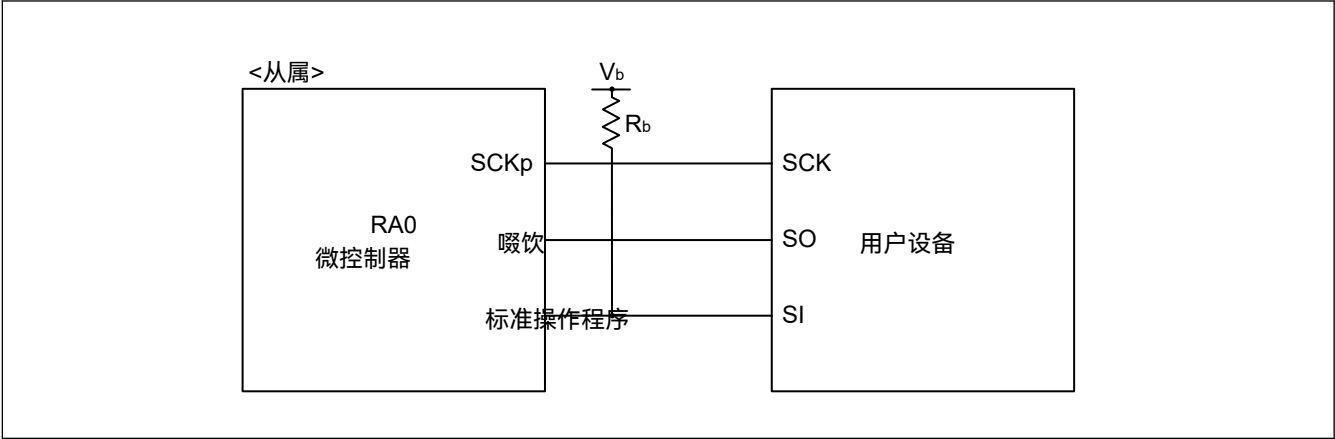


图 27.25 简化的 SPI 通信中与不同电压等级设备的连接

- 注:
- $R_b[\Omega]$: 通信线 (SO_p) 上拉电阻, $C_b[F]$: 通信线, (SO_p) 负载电容, $V_b[V]$: 通信线路电压
 - p: 简化的 SPI 编号 ((p = 00, 11), m: 单元编号, n: 通道编号 ((mn = 00, 03), gh: 端口编号 ((gh = 100 至 102、108 至 110、112、201、212、213)
 - f_{MCK} : 串行阵列单元操作时钟频率
要设置此工作时钟, 请使用串行模式寄存器 mn (SMRmn) 中的 CKS 位。
m: 单元编号, n: 通道编号(mn = 00, 03, 10, 11)
 - 由于 P212PFS_A 和 P213PFS_A 寄存器没有 PIM 位, 因此无法使用 P212 和 P213 与工作电压不同的设备进行通信。

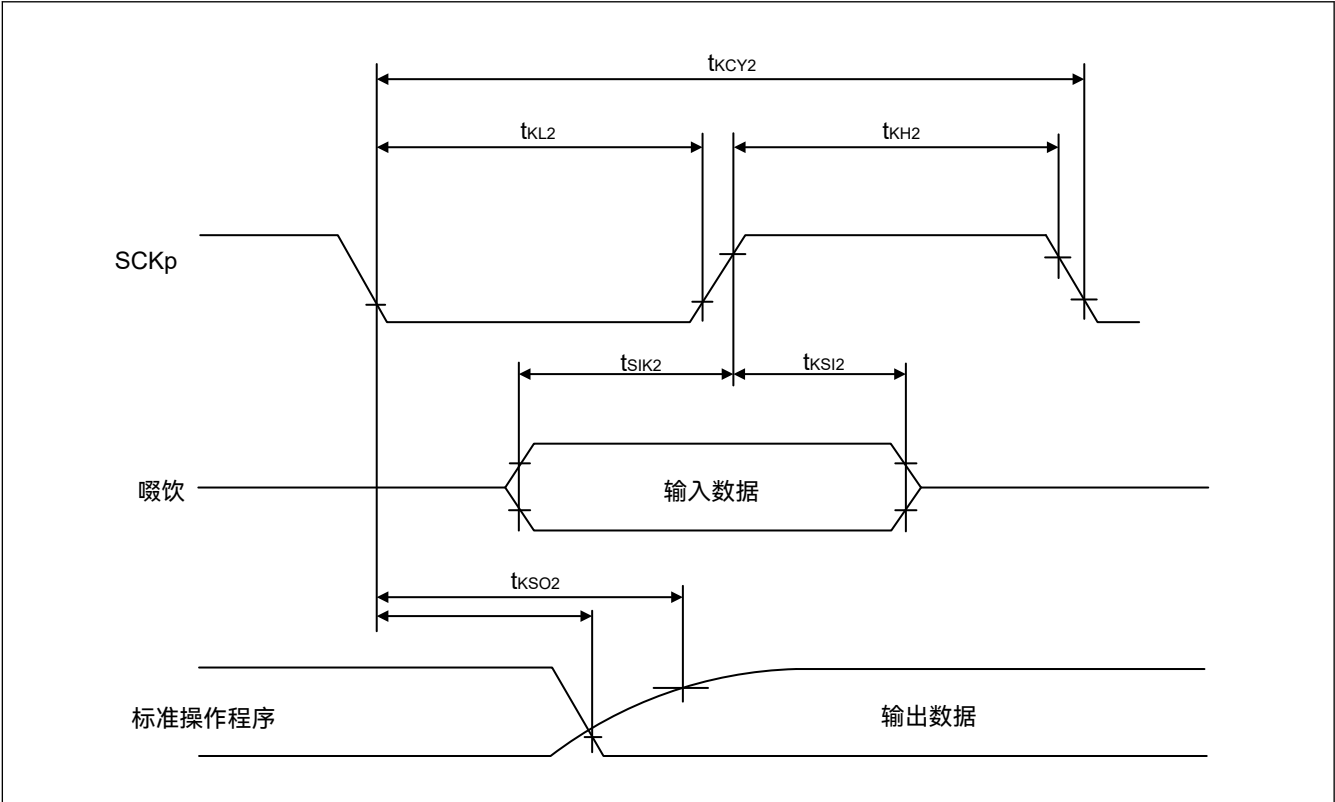


图 27.26 简化 SPI 通信中从模式下串行传输的时序, 其中器件工作在不同电压电平, 当 SCRmn.DCP[1:0] = 00b 或 11b

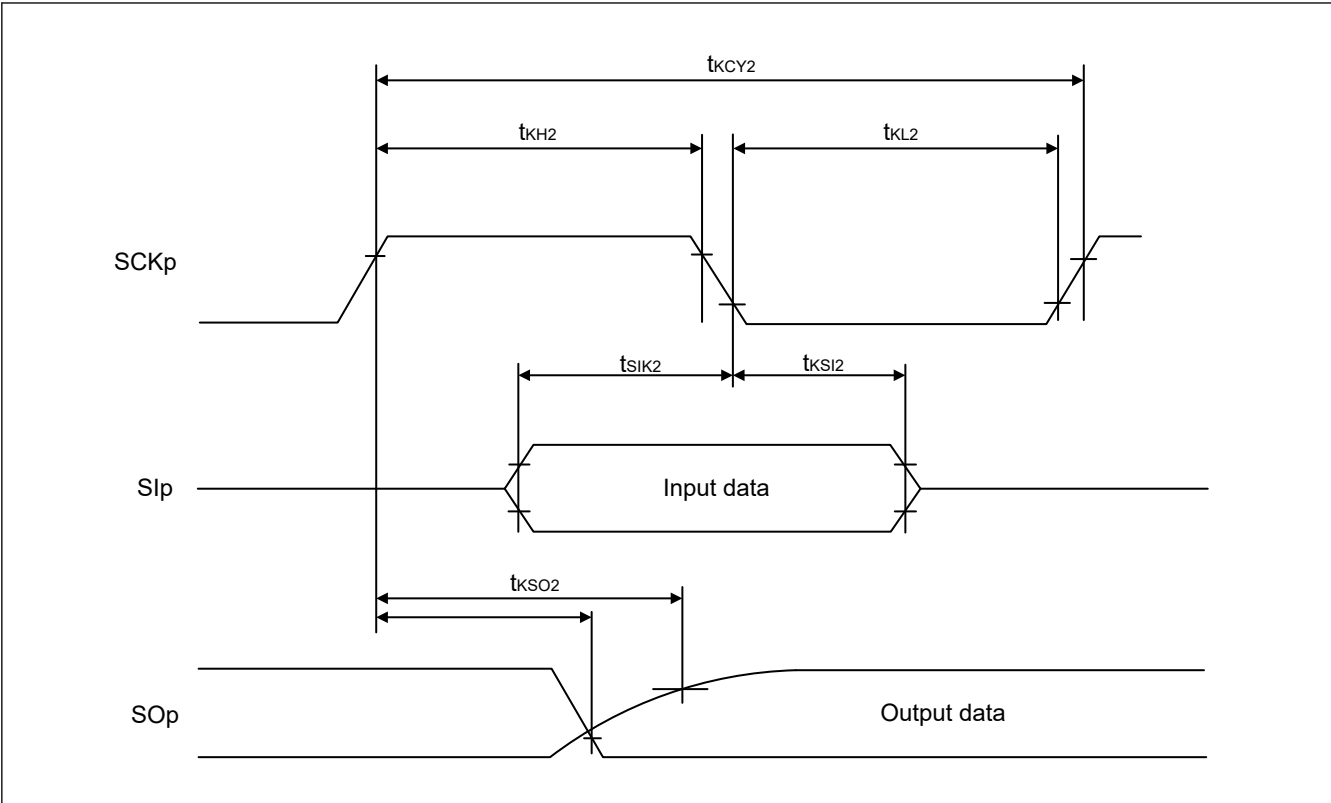


Figure 27.27 Timing of serial transfer in the simplified SPI communications in the slave mode with devices operating at different voltage levels when SCRmn.DCP[1:0] = 01b or 10b

Note:

- p: Simplified SPI number (p = 00, 11), m: Unit number, n: Channel number (mn = 00, 03), gh: Port number (gh = 100 to 102, 108 to 110, 112, 201, 212, 213)
- Communications by using P212 and P213 with devices operating at different voltage levels are not possible since P212PFS_A and P213PFS_A registers do not have PIM bit.

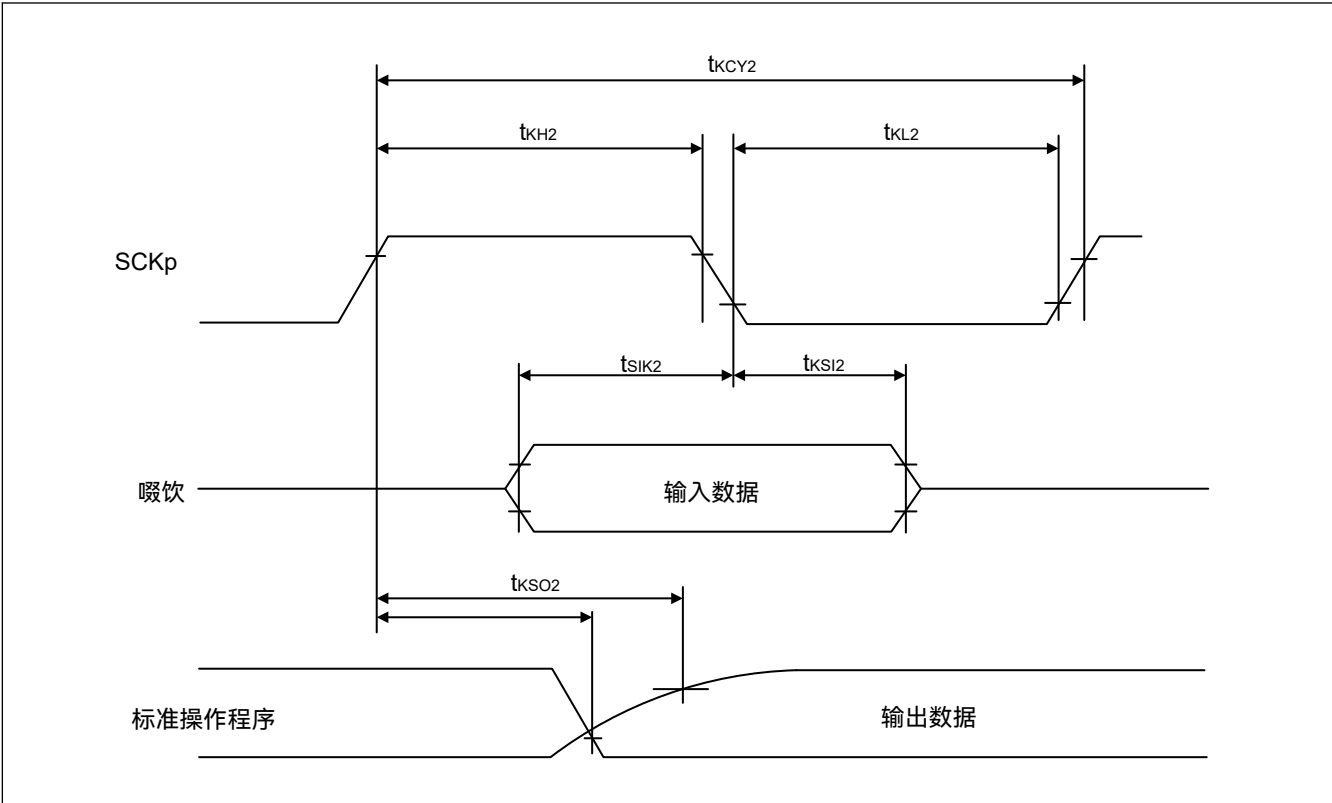


图 27.27 简化 SPI 通信中从模式下串行传输的时序，其中器件工作在不同电压电平，当 SCRmn.DCP[1:0] = 01b 或 10b 时

注: ●p: 简化的SPI编号(p = 00, 11) , m: 单元编号, n: 通道编号(mn = 00, 03) , gh: 端口编号(gh = (100至102, 108至110, 112, 201, 212, 213)

- 由于 P212PFS_A 和 P213PFS_A 寄存器没有 PIM 位，因此无法使用 P212 和 P213 与工作电压不同的设备进行通信。

Table 27.33 Simplified IIC communications with devices operating at different voltage levels (1.8 V, 2.5 V, or 3 V) (1 of 2)

Conditions: VCC = 1.8 to 5.5 V, VSS = 0 V, Ta = -40 to +125°C

Parameter	Symbol	High-speed mode		Middle-speed mode		Low-speed mode		Unit	Test Conditions
		Min.	Max.	Min.	Max.	Min.	Max.		
SCLr clock frequency	f_{SCL}	—	1000 ^{*1}	—	1000 ^{*1}	—	300 ^{*1}	kHz	Figure 27.29
		—	1000 ^{*1}	—	1000 ^{*1}	—	300 ^{*1}	kHz	
		—	400 ^{*1}	—	400 ^{*1}	—	300 ^{*1}	kHz	
		—	400 ^{*1}	—	400 ^{*1}	—	300 ^{*1}	kHz	
		—	300 ^{*1}	—	300 ^{*1}	—	300 ^{*1}	kHz	
Hold time when SCLr is low	t_{LOW}	475	—	475	—	1550	—	ns	
		475	—	475	—	1550	—	ns	
		1150	—	1550	—	1550	—	ns	
		1150	—	1550	—	1550	—	ns	
		1550	—	1550	—	1550	—	ns	
Hold time when SCLr is high	t_{HIGH}	245	—	245	—	610	—	ns	
		200	—	200	—	610	—	ns	
		675	—	675	—	610	—	ns	
		600	—	600	—	610	—	ns	
		610	—	610	—	610	—	ns	

表 27.33 简化的 IIC 通信，适用于工作电压不同的设备(1.8 V, 2.5 V, 或 3 V) (1/2)

条件: VCC= 1.8 至 5.5 V, VSS= 0 V, Ta= -40 至+125°C

范围	象征	高速模式		中速模式		低速模式		单元	测试条件
		最小。	最大限	最小。	最大限	最小。	最大限		
SCLr 时钟频率	f_{SCL}	—	1000 ^{*1}	—	1000 ^{*1}	—	300 ^{*1}	kHz	图 27.29
		—	1000 ^{*1}	—	1000 ^{*1}	—	300 ^{*1}	kHz	
		—	400 ^{*1}	—	400 ^{*1}	—	300 ^{*1}	kHz	
		—	400 ^{*1}	—	400 ^{*1}	—	300 ^{*1}	kHz	
		—	300 ^{*1}	—	300 ^{*1}	—	300 ^{*1}	kHz	
保持时间 SCLr 低	低	475	—	475	—	1550	—	ns	
		475	—	475	—	1550	—	ns	
		1150	—	1550	—	1550	—	ns	
		1150	—	1550	—	1550	—	ns	
		1550	—	1550	—	1550	—	ns	
保持时间 SCLr 值高	大脚	245	—	245	—	610	—	ns	
		200	—	200	—	610	—	ns	
		675	—	675	—	610	—	ns	
		600	—	600	—	610	—	ns	
		610	—	610	—	610	—	ns	

Table 27.33 Simplified IIC communications with devices operating at different voltage levels (1.8 V, 2.5 V, or 3 V) (2 of 2)

Conditions: VCC = 1.8 to 5.5 V, VSS = 0 V, Ta = -40 to +125°C

Parameter	Symbol	High-speed mode		Middle-speed mode		Low-speed mode		Unit	Test Conditions
		Min.	Max.	Min.	Max.	Min.	Max.		
Data setup time (reception)	4.0 V ≤ VCC ≤ 5.5 V, 2.7 V ≤ Vb ≤ 4.0 V, Cb = 50 pF, Rb = 2.7 kΩ	1/fMCK + 135 ⁻³	—	1/fMCK + 135 ⁻³	—	1/fMCK + 190 ⁻³	—	ns	Figure 27.29
	2.7 V ≤ VCC < 4.0 V, 2.3 V ≤ Vb ≤ 2.7 V, Cb = 50 pF, Rb = 2.7 kΩ	1/fMCK + 135 ⁻³	—	1/fMCK + 135 ⁻³	—	1/fMCK + 190 ⁻³	—	ns	
	4.0 V ≤ VCC ≤ 5.5 V, 2.7 V ≤ Vb ≤ 4.0 V, Cb = 100 pF, Rb = 2.8 kΩ	1/fMCK + 190 ⁻³	—	1/fMCK + 190 ⁻³	—	1/fMCK + 190 ⁻³	—	ns	
	2.7 V ≤ VCC < 4.0 V, 2.3 V ≤ Vb ≤ 2.7 V, Cb = 100 pF, Rb = 2.7 kΩ	1/fMCK + 190 ⁻³	—	1/fMCK + 190 ⁻³	—	1/fMCK + 190 ⁻³	—	ns	
	1.8 V ≤ VCC < 3.3 V, 1.6 V ≤ Vb ≤ 2.0 V ² , Cb = 100 pF, Rb = 5.5 kΩ	1/fMCK + 190 ⁻³	—	1/fMCK + 190 ⁻³	—	1/fMCK + 190 ⁻³	—	ns	
Data hold time (transmission)	4.0 V ≤ VCC ≤ 5.5 V, 2.7 V ≤ Vb ≤ 4.0 V, Cb = 50 pF, Rb = 2.7 kΩ	0	305	0	305	0	305	ns	
	2.7 V ≤ VCC < 4.0 V, 2.3 V ≤ Vb ≤ 2.7 V, Cb = 50 pF, Rb = 2.7 kΩ	0	305	0	305	0	305	ns	
	4.0 V ≤ VCC ≤ 5.5 V, 2.7 V ≤ Vb ≤ 4.0 V, Cb = 100 pF, Rb = 2.8 kΩ	0	355	0	355	0	355	ns	
	2.7 V ≤ VCC < 4.0 V, 2.3 V ≤ Vb ≤ 2.7 V, Cb = 100 pF, Rb = 2.7 kΩ	0	355	0	355	0	355	ns	
	1.8 V ≤ VCC < 3.3 V, 1.6 V ≤ Vb ≤ 2.0 V ² , Cb = 100 pF, Rb = 5.5 kΩ	0	405	0	405	0	405	ns	

Note 1. The listed times must be no greater than fMCK/4.
Note 2. Use this setting with VCC ≥ Vb.
Note 3. Set fMCK so that it does not exceed the hold time when SCLr is low or high.

Note: Select the TTL input buffer and the N-ch open drain output [withstand voltage of VCC] mode for the SDAr pin and the N-ch open drain output [withstand voltage of VCC] mode for the SCLr pin by using the Port gh Pin Function Select Register (PghPFS_A.PIM and PghPFS_A.NCODR). For VIH and VIL, see the DC characteristics with the TTL input buffer selected.

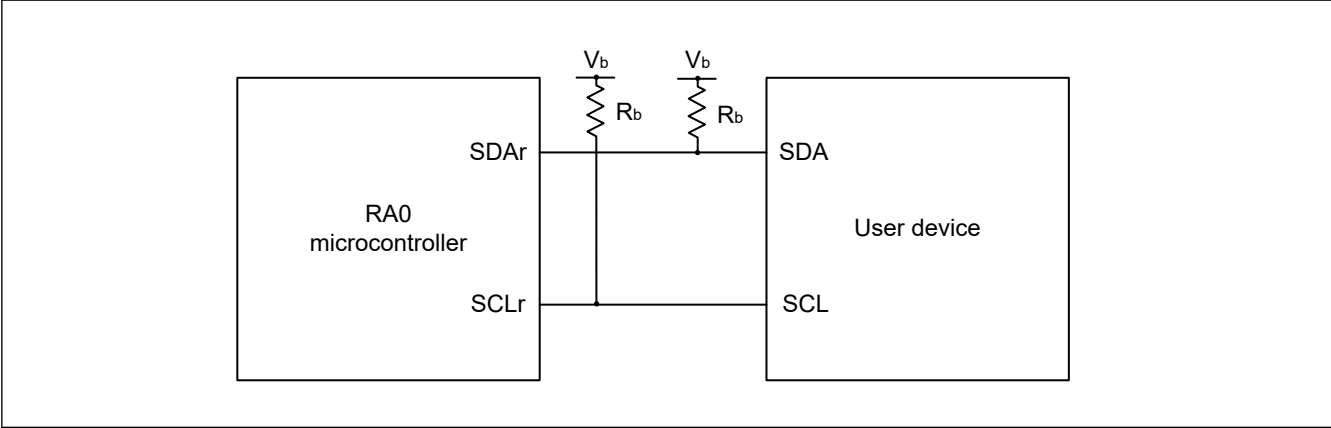


Figure 27.28 Connection in the IIC communications with devices operating at different voltage levels

表 27.33 简化的 IIC 通信，适用于工作电压不同的设备(1.8 V, 2.5 V, 或 3 V) (2/2)

条件: VCC = 1.8至 5.5 V, VSS= 0 V, Ta= -40 至+125°C

范围	象征	高速模式		中速模式		低速模式		单元	测试条件
		最小。	最大限	最小。	最大限	最小。	最大限		
数据建立时间 (接收)	4.0 V ≤ VCC ≤ 5.5 V, 2.7V ≤ Vb ≤ 4.0V, Cb= 50 pF, Rb = 2.7 kΩ	1/fMCK + 135 ⁻³	度。	1/fMCK + 135 ⁻³	度。	1/fMCK + 190 ⁻³	度。	ns	图 27.29
	2.7 V ≤ VCC < 4.0 V, 2.3V ≤ Vb ≤ 2.7V, Cb= 50 pF, Rb = 2.7 kΩ	1/fMCK + 135 ⁻³	—	1/fMCK + 135 ⁻³	—	1/fMCK + 190 ⁻³	—	ns	
	4.0 V ≤ VCC ≤ 5.5 V, 2.7V ≤ Vb ≤ 4.0V, Cb= 100 pF, Rb = 2.8 kΩ	1/fMCK + 190 ⁻³	—	1/fMCK + 190 ⁻³	—	1/fMCK + 190 ⁻³	—	ns	
	2.7 V ≤ VCC < 4.0 V, 2.3V ≤ Vb ≤ 2.7V, Cb= 100 pF, Rb = 2.7 kΩ	1/fMCK + 190 ⁻³	—	1/fMCK + 190 ⁻³	—	1/fMCK + 190 ⁻³	—	ns	
	1.8 V ≤ VCC < 3.3 V, 1.6V ≤ Vb ≤ 2.0V*2, Cb= 100 pF, Rb = 5.5 kΩ	1/fMCK + 190 ⁻³	—	1/fMCK + 190 ⁻³	—	1/fMCK + 190 ⁻³	—	ns	
数据保持时间 (传输)	4.0 V ≤ VCC ≤ 5.5 V, 2.7 V ≤ Vb ≤ 4.0 V, Cb = 50pF, Rb = 2.7 kΩ	0	305	0	305	0	305	ns	
	2.7 V ≤ VCC < 4.0 V, 2.3 V ≤ Vb ≤ 2.7 V, Cb = 50pF, Rb = 2.7 kΩ	0	305	0	305	0	305	ns	
	4.0 V ≤ VCC ≤ 5.5 V, 2.7 V ≤ Vb ≤ 4.0 V, Cb = 100 pF, Rb = 2.8 kΩ	0	355	0	355	0	355	ns	
	2.7 V ≤ VCC < 4.0 V, 2.3V ≤ Vb ≤ 2.7V, Cb= 100 pF, Rb = 2.7 kΩ	0	355	0	355	0	355	ns	
	1.8 V ≤ VCC < 3.3 V, 1.6V ≤ Vb ≤ 2.0V*2, Cb = 100 pF, Rb = 5.5 kΩ	0	405	0	405	0	405	ns	

注1.所列时间不得大于fMCK/4。
注2. 将此设置与VCC ≥ Vb一起使用。
注 3. 设置 fMCK，使其在 SCLr 低或高时不超过保持时间。

注意：使用端口gh引脚功能选择寄存器（PghPFS_A.PIM和PghPFS_A.NCODR）为SDAr引脚选择TTL输入缓冲器和N沟道开漏输出[耐压]模式，为SCLr引脚选择N沟道开漏输出[耐压]模式。对于VIH和VIL，请参阅选择TTL输入缓冲器时的直流特性。

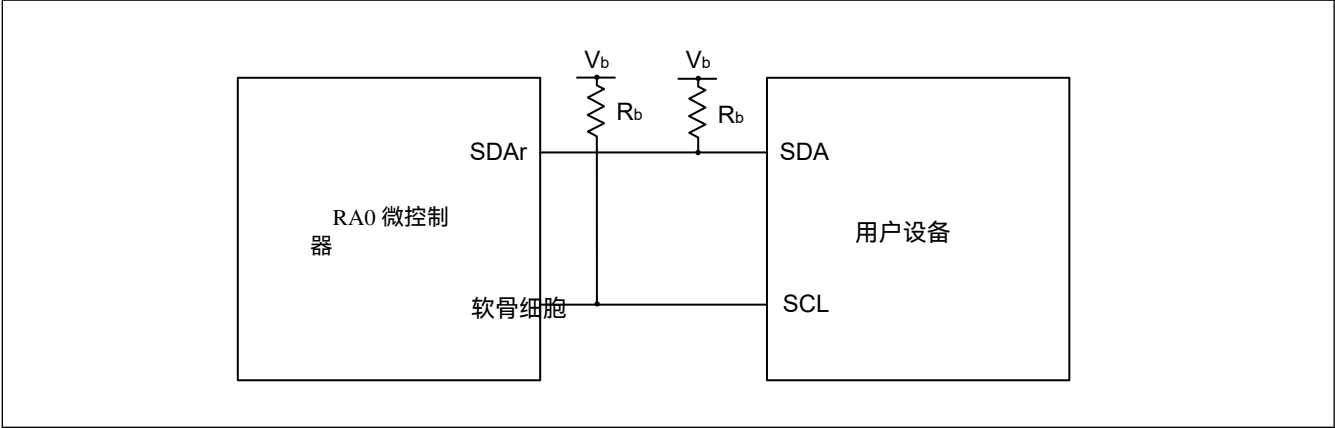


图 27.28 IIC 通信中与不同电压等级设备的连接

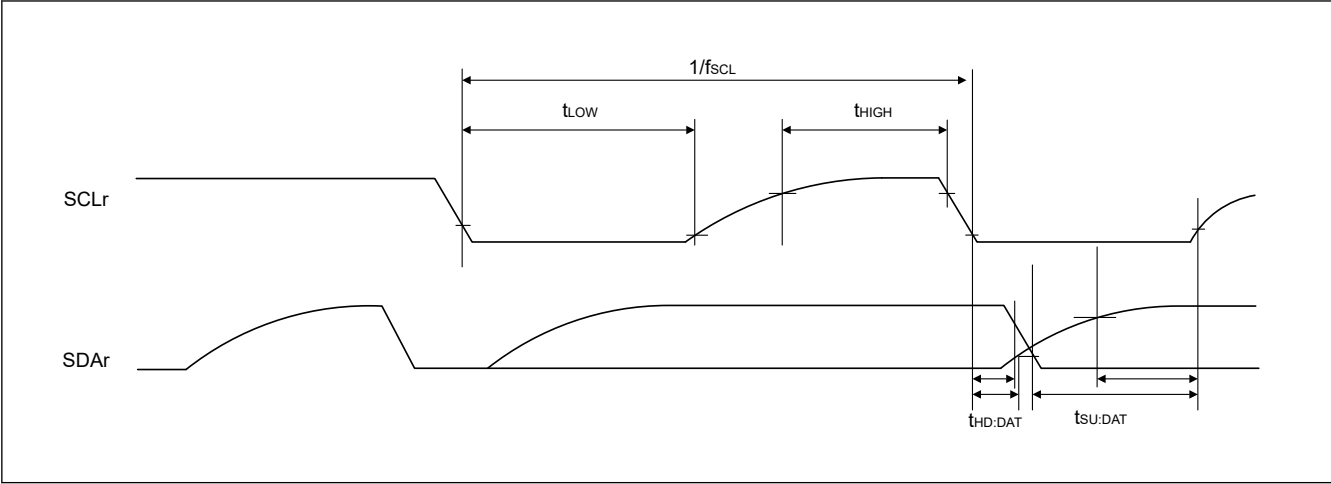


Figure 27.29 Timing of serial transfer in simplified IIC communications with devices operating at different voltage levels

Note:

- $R_b[\Omega]$: Communication line (SDAr, SCLr) pull-up resistance, $C_b[F]$: Communication line (SDAr, SCLr) load capacitance, $V_b[V]$: Communication line voltage
- r: Simplified IIC number (r = 00, 11), gh: Port number (gh = 100, 102, 110, 112, 201, 212)
- f_{MCK} : Serial array unit operation clock frequency
To set this operating clock, use the CKS bit in the serial mode register mn (SMRmn).
m: Unit number, n: Channel number (mn = 00, 03)

27.5.2 I²C Bus Interface (IICA)

Table 27.34 I²C standard mode

Conditions: VCC = 1.6 to 5.5 V, VSS = 0 V, Ta = -40 to +125°C

Parameter		Symbol	Min.	Typ.	Max.	Unit	Test conditions
SCLAn clock frequency	Standard mode: PCLKB ≥ 1 MHz	f _{SCL}	0	—	100	kHz	Figure 27.30
Setup time of restart condition	—	t _{SU:STA}	4.7	—	—	μs	
Hold time ^{*1}	—	t _{HD:STA}	4	—	—	μs	
Hold time when SCLAn is low	—	t _{LOW}	4.7	—	—	μs	
Hold time when SCLAn is high	—	t _{HIGH}	4	—	—	μs	
Data setup time (reception)	—	t _{SU:DAT}	250	—	—	ns	
Data hold time (transmission) ^{*2}	—	t _{HD:DAT}	0	—	3.45	μs	
Setup time of stop condition	—	t _{SU:STO}	4	—	—	μs	
Bus-free time	—	t _{BUF}	4.7	—	—	μs	

Note 1. The first clock pulse is generated after this period when the start or restart condition is detected.
Note 2. The maximum value of t_{HD:DAT} applies to normal transfer. The clock stretching will be inserted on reception of an acknowledgment (ACK) signal.

Note: n: Unit number (0)

Note: Communications by using P212 and P213 with devices operating at different voltage levels are not possible since P212PFS_A and P213PFS_A registers do not have PIM bit.

Note: The maximum value of communication line capacitance (C_b) and communication line pull-up resistor (R_b) are as follows.
 $C_b = 400 \text{ pF}$, $R_b = 2.7 \text{ k}\Omega$

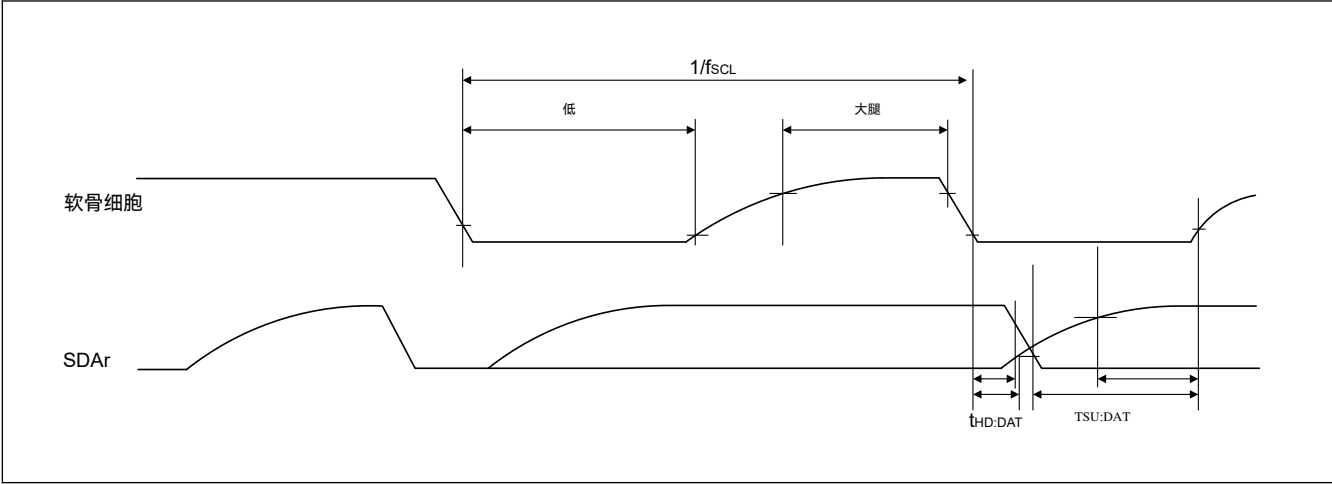


图 27.29 简化 IIC 通信中，不同电压电平器件的串行传输时序

注: ● $R_b[\Omega]$: 通信线路 (SDAr、SCLr) 上拉电阻, $C_b[F]$: 通信线路 (SDAr、SCLr) 负载电容, $V_b[V]$: 通信线路电压

●r: 简化的 IIC 号码(r = 00, 11) , gh: 端口号(gh = 100, 102, 110、112、201、212)

● f_{MCK} : 串行阵列单元操作时钟频率
要设置此工作时钟, 请使用串行模式寄存器 mn (SMRmn) 中的 CKS 位。
m: 单元编号, n: 通道编号(mn = 00, 03)

27.5.2I²C 总线接口 (IICA)

表 27.34I²C 标准模式

条件: VCC = 1.6至5.5 V, VSS= 0 V, Ta = -40 至+125°C

范围		象征	最小。	类型。	最大限	单元	测试条件
SCLAn 时钟频率	标准模式: PCLKB≥ 1 MHz	f _{SCL}	0	—	100	kHz	图 27.30
重启条件的设置时间	—	津苏:STA	4.7	—	—	μs	
保持时间*1	—	t _{HD:STA}	4	—	—	μs	
SCLAn 低时的保持时间	—	低	4.7	—	—	μs	
SCLAn 高的保持时间	—	大腿	4	—	—	μs	
数据建立时间 (接收)	—	TSU:DAT	250	—	—	ns	
数据保持时间 (传输) *2	—	t _{HD:DAT}	0	—	3.45	μs	
停止条件的设置时间	—	津: STO	4	—	—	μs	
无公交时段	—	t _{BUF}	4.7	—	—	μs	

注1. 第一个时钟脉冲是在此周期之后, 当检测到启动或重启条件时产生的。
注2: t_{HD:DAT}的最大值适用于正常传输。时钟拉伸将在收到确认 (ACK) 信号时插入。

注: n: 单元编号 (0)

注意: 由于 P212PFS_A 和 P213PFS_A 寄存器没有 PIM 位, 因此无法使用 P212 和 P213 与工作电压不同的设备进行通信。

注意: 通信线路电容(C_b) 和通信线路上拉电阻(R_b)的最大值如下。
 $C_b = 400 \text{ pF}$, $R_b = 2.7 \text{ k}\Omega$

Table 27.35 I²C fast mode

Conditions: VCC = 1.8 to 5.5 V, VSS = 0 V, Ta = -40 to +125°C

Parameter		Symbol	Min.	Typ.	Max.	Unit	Test conditions
SCLAn clock frequency	Fast mode: PCLKB ≥ 3.5 MHz 1.8 V ≤ VCC ≤ 5.5 V	f _{SCL}	0	—	400	kHz	Figure 27.30
Setup time of restart condition	1.8 V ≤ VCC ≤ 5.5 V	t _{SU:STA}	0.6	—	—	μs	
Hold time*1	1.8 V ≤ VCC ≤ 5.5 V	t _{HD:STA}	0.6	—	—	μs	
Hold time when SCLAn is low	1.8 V ≤ VCC ≤ 5.5 V	t _{LOW}	1.3	—	—	μs	
Hold time when SCLAn is high	1.8 V ≤ VCC ≤ 5.5 V	t _{HIGH}	0.6	—	—	μs	
Data setup time (reception)	1.8 V ≤ VCC ≤ 5.5 V	t _{SU:DAT}	100	—	—	ns	
Data hold time (transmission)*2	1.8 V ≤ VCC ≤ 5.5 V	t _{HD:DAT}	0	—	0.9	μs	
Setup time of stop condition	1.8 V ≤ VCC ≤ 5.5 V	t _{SU:STO}	0.6	—	—	μs	
Bus-free time	1.8 V ≤ VCC ≤ 5.5 V	t _{BUF}	1.3	—	—	μs	

Note 1. The first clock pulse is generated after this period when the start or restart condition is detected.

Note 2. The maximum value of t_{HD:DAT} applies to normal transfer. The clock stretching will be inserted on reception of an acknowledgment (ACK) signal.

Note: Communications by using P212 and P213 with devices operating at different voltage levels are not possible since P212PFS_A and P213PFS_A registers do not have PIM bit.

Note: The maximum value of communication line capacitance (C_b) and communication line pull-up resistor (R_b) are as follows.

$C_b = 320\text{ pF}$, $R_b = 1.1\text{ k}\Omega$

Table 27.36 I²C fast mode plus

Conditions: VCC = 2.7 to 5.5 V, VSS = 0 V, Ta = -40 to +125°C

Parameter		Symbol	Min.	Typ.	Max.	Unit	Test conditions
SCLAn clock frequency	Fast mode plus: PCLKB ≥ 10 MHz 2.7 V ≤ VCC ≤ 5.5 V	f _{SCL}	0	—	1000	kHz	Figure 27.30
Setup time of restart condition	2.7 V ≤ VCC ≤ 5.5 V	t _{SU:STA}	0.26	—	—	μs	
Hold time*1	2.7 V ≤ VCC ≤ 5.5 V	t _{HD:STA}	0.26	—	—	μs	
Hold time when SCLAn is low	2.7 V ≤ VCC ≤ 5.5 V	t _{LOW}	0.5	—	—	μs	
Hold time when SCLAn is high	2.7 V ≤ VCC ≤ 5.5 V	t _{HIGH}	0.26	—	—	μs	
Data setup time (reception)	2.7 V ≤ VCC ≤ 5.5 V	t _{SU:DAT}	50	—	—	ns	
Data hold time (transmission)*2	2.7 V ≤ VCC ≤ 5.5 V	t _{HD:DAT}	0	—	0.45	μs	
Setup time of stop condition	2.7 V ≤ VCC ≤ 5.5 V	t _{SU:STO}	0.26	—	—	μs	
Bus-free time	2.7 V ≤ VCC ≤ 5.5 V	t _{BUF}	0.5	—	—	μs	

Note 1. The first clock pulse is generated after this period when the start or restart condition is detected.

Note 2. The maximum value of t_{HD:DAT} applies to normal transfer. The clock stretching will be inserted on reception of an acknowledgment (ACK) signal.

Note: Communications by using P212 and P213 with devices operating at different voltage levels are not possible since P212PFS_A and P213PFS_A registers do not have PIM bit.

Note: The maximum value of communication line capacitance (C_b) and communication line pull-up resistor (R_b) are as follows.

$C_b = 120\text{ pF}$, $R_b = 1.1\text{ k}\Omega$

表 27.35I²C 快速模式

条件: VCC = 1.8至5.5 V, VSS= 0V, Ta= -40 至+125°C

范围		象征	最小。	类型。	最大限	单元	测试条件
SCLAn 时钟频率	快速模式: PCLKB≥ 3.5 MHz 1.8 V ≤ VCC ≤ 5.5 V	f _{SCL}	0	—	400	kHz	图 27.30
重启条件的设置时间	1.8 V ≤ VCC ≤ 5.5 V	津苏:STA	0.6	—	—	μs	
保持时间*1	1.8 V ≤ VCC ≤ 5.5 V	t _{HD:STA}	0.6	—	—	μs	
SCLAn 低时的保持时间	1.8 V ≤ VCC ≤ 5.5 V	低	1.3	—	—	μs	
SCLAn 高的保持时间	1.8 V ≤ VCC ≤ 5.5 V	t _{HIGH}	0.6	—	—	μs	
数据建立时间（接收）	1.8 V ≤ VCC ≤ 5.5 V	TSU:DAT	100	—	—	ns	
数据保持时间（传输）*2	1.8 V ≤ VCC ≤ 5.5 V	t _{HD:DAT}	0	—	0.9	μs	
停止条件的设置时间	1.8 V ≤ VCC ≤ 5.5 V	津: STO	0.6	—	—	μs	
无公交时段	1.8 V ≤ VCC ≤ 5.5 V	t _{BUF}	1.3	—	—	μs	

注1. 第一个时钟脉冲是在此周期之后，当检测到启动或重启条件时产生的。

注2: t_{HD:DAT}的最大值适用于正常传输。时钟拉伸将在收到确认（ACK）信号时插入。

注意：由于 P212PFS_A 和 P213PFS_A 寄存器没有 PIM 位，因此无法使用 P212 和 P213 与工作电压不同的设备进行通信。

注意：通信线路电容(C_b)和通信线路上拉电阻(R_b)的最大值如下。

$C_b = 320\text{ pF}$, $R_b = 1.1\text{ k}\Omega$

表 27.36I²C 快速模式加

条件: VCC = 2.7至5.5 V, VSS= 0V, Ta = -40 至+125°C

范围		象征	最小。	类型。	最大限	单元	测试条件
SCLAn 时钟频率	快速模式增强: PCLKB≥ 10 MHz 2.7 V ≤ VCC ≤ 5.5 V	f _{SCL}	0	—	1000	kHz	图 27.30
重启条件的设置时间	2.7 V ≤ VCC ≤ 5.5 V	津苏:STA	0.26	—	—	μs	
保持时间*1	2.7 V ≤ VCC ≤ 5.5 V	t _{HD:STA}	0.26	—	—	μs	
SCLAn 低时的保持时间	2.7 V ≤ VCC ≤ 5.5 V	低	0.5	—	—	μs	
SCLAn 高的保持时间	2.7 V ≤ VCC ≤ 5.5 V	大腿	0.26	—	—	μs	
数据建立时间（接收）	2.7 V ≤ VCC ≤ 5.5 V	TSU:DAT	50	—	—	ns	
数据保持时间（传输）*2	2.7 V ≤ VCC ≤ 5.5 V	t _{HD:DAT}	0	—	0.45	μs	
停止条件的设置时间	2.7 V ≤ VCC ≤ 5.5 V	津: STO	0.26	—	—	μs	
无公交时间	2.7 V ≤ VCC ≤ 5.5 V	t _{BUF}	0.5	—	—	μs	

注1. 第一个时钟脉冲是在此周期之后，当检测到启动或重启条件时产生的。

注2: t_{HD:DAT}的最大值适用于正常传输。时钟拉伸将在收到确认（ACK）信号时插入。

注意：由于 P212PFS_A 和 P213PFS_A 寄存器没有 PIM 位，因此无法使用 P212 和 P213 与工作电压不同的设备进行通信。

注意：通信线路电容(C_b)和通信线路上拉电阻(R_b)的最大值如下。

$C_b = 120\text{ pF}$, $R_b = 1.1\text{ k}\Omega$

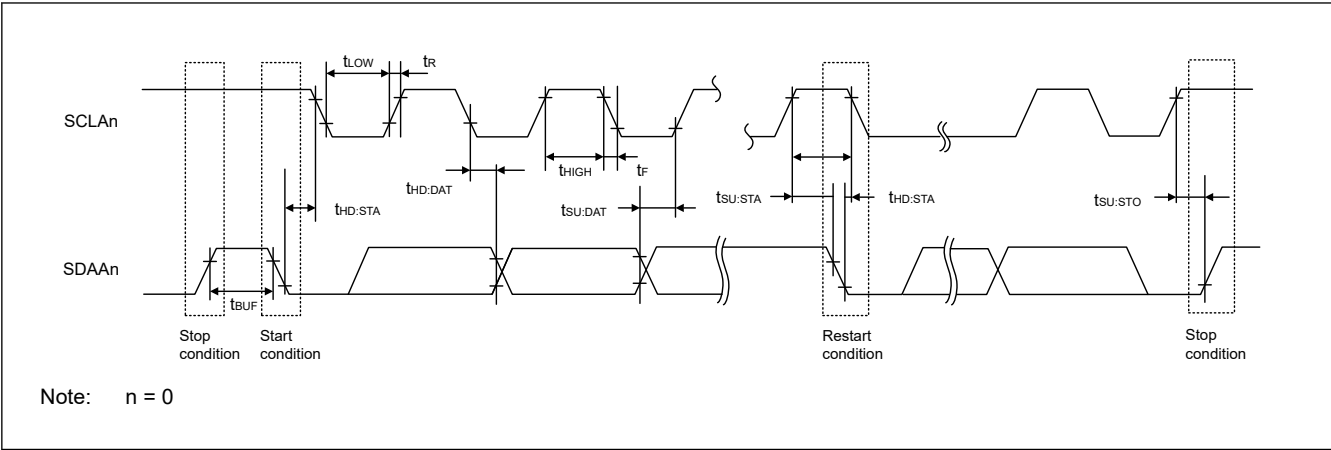


Figure 27.30 I2C serial transfer timing

27.6 Analog Characteristics

27.6.1 A/D Converter Characteristics for $T_A = -40$ to $+85^\circ\text{C}$

Table 27.37 Reference for the characteristics of the A/D converter

Input channel	Reference Voltage		
	Reference voltage (+) = VREFH0 Reference voltage (-) = VREFL0	Reference voltage (+) = VCC Reference voltage (-) = VSS	Reference voltage (+) = V_{BGR} Reference voltage (-) = VREFL0
AN000, AN001, AN004, AN005	Refer to Table 27.38.	Refer to Table 27.40.	Refer to Table 27.41.
AN021, AN022	Refer to Table 27.39.		
Internal reference voltage Temperature sensor output voltage	Refer to Table 27.38.		—

Table 27.38 A/D conversion characteristics (AN004, AN005, internal reference voltage, and temperature sensor output voltage) (1 of 2)

Conditions: $1.6\text{ V} \leq V_{REFH0} \leq V_{CC} \leq 5.5\text{ V}$, $V_{SS} = 0\text{ V}$, $T_a = -40$ to $+85^\circ\text{C}$
Reference voltage range applied to the VREFH0 (ADREFP[1:0] = 01b) and VREFL0 (ADREFM = 1b).
Target pins: AN004, AN005, internal reference voltage, and temperature sensor output voltage
(Reference voltage (+) = VREFH0, Reference voltage (-) = VREFL0 = 0 V)

Parameter	Symbol	Min.	Typ.	Max.	Unit	Conditions	
Resolution	RES	8	—	10	bit	—	
Overall error*1*2*3	AINL	—	1.2	± 3.5	LSB	10-bit resolution VREFH0 = V_{CC}^{*5}	$1.8\text{ V} \leq V_{REFH0} \leq 5.5\text{ V}$
		—	1.2	± 7.0	LSB		$1.6\text{ V} \leq V_{REFH0} \leq 5.5\text{ V}^{*6}$

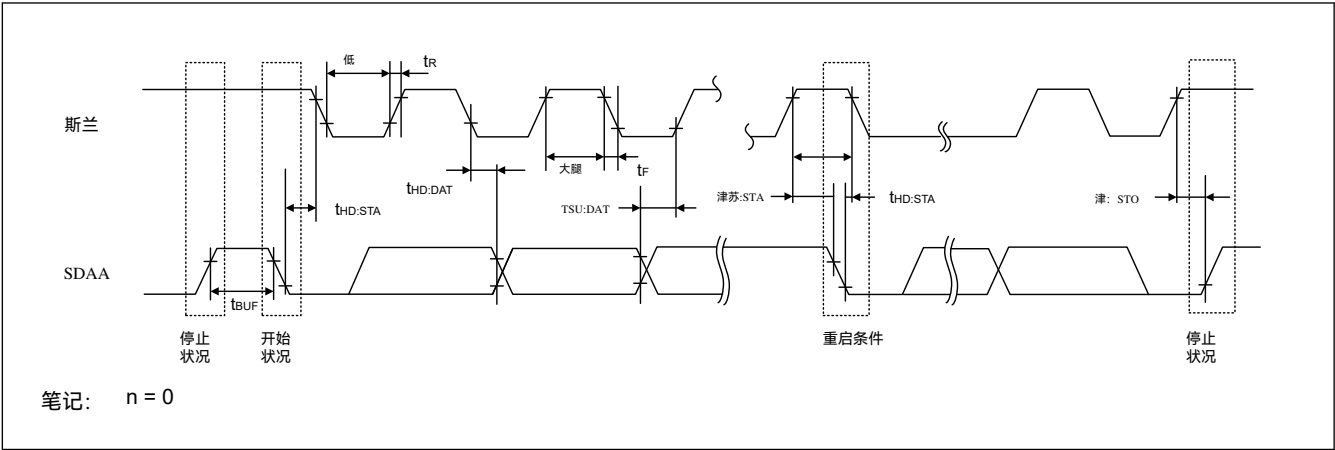


图 27.30 I2C 串行传输时序

27.6 模拟特性

27.6.1 $T_A = -40$ 至 $+85^\circ\text{C}$ 的 A/D 转换器特性

表 27.37 A/D 转换器特性参考

输入通道	参考电压		
	参考电压(+) = VREFH0 参考电压(-) = VREFL0	参考电压(+) = VCC 参考电压(-) = VSS	参考电压(+) = V_{BGR} 参考电压(-) = VREFL0
AN000、AN001、AN004、AN005	请参阅表 27.38。	请参阅表 27.40。	请参阅表 27.41。
AN021, AN022	请参阅表 27.39。		
内部参考电压温度传感器 输出电压	请参阅表 27.38。		—

表 27.38 A/D 转换特性 (AN004、AN005、内部参考电压和温度传感器输出电压) (1/2)

条件: $1.6\text{ V} \leq V_{REFH0} \leq V_{CC} \leq 5.5\text{ V}$, $V_{SS} = 0\text{ V}$, $T_a = -40$ to $+85^\circ\text{C}$
参考电压范围应用于 VREFH0 (ADREFP[1:0] = 01b) 和 VREFL0 (ADREFM = 1b)。
目标引脚: AN004、AN005、内部参考电压和温度传感器输出电压
(参考电压(+) = VREFH0, 参考电压(-) = VREFL0 = 0 V)

范围	象征	最小。	类型。	最大限度。	单元	状况	
解决	RES	8	—	10	少量	—	
全面的 错误*1*2*3	AINL	—	1.2	± 3.5	LSB	10 位分辨率 VREFH0 = V_{CC}^{*5}	$1.8\text{ V} \leq V_{REFH0} \leq 5.5\text{ V}$
		—	1.2	± 7.0	LSB		$1.6\text{ V} \leq V_{REFH0} \leq 5.5\text{ V}^{*6}$

Table 27.38 A/D conversion characteristics (AN004, AN005, internal reference voltage, and temperature sensor output voltage) (2 of 2)

Conditions: 1.6 V ≤ VREFH0 ≤ VCC ≤ 5.5 V, VSS = 0 V, Ta = -40 to +85°C
Reference voltage range applied to the VREFH0 (ADREFP[1:0] = 01b) and VREFL0 (ADREFM = 1b).
Target pins: AN004, AN005, internal reference voltage, and temperature sensor output voltage
(Reference voltage (+) = VREFH0, Reference voltage (–) = VREFL0 = 0 V)

Parameter	Symbol	Min.	Typ.	Max.	Unit	Conditions	
Conversion time	tCONV	2.125	—	39	μs	10-bit resolution Target pin: AN004, AN005	3.6 V ≤ VCC ≤ 5.5 V
		3.1875	—	39	μs		2.7 V ≤ VCC ≤ 5.5 V
		17	—	39	μs		1.8 V ≤ VCC ≤ 5.5 V
		57	—	95	μs		1.6 V ≤ VCC ≤ 5.5 V
		2.375	—	39	μs	10-bit resolution Target pin: Internal reference voltage, and temperature sensor output voltage	3.6 V ≤ VCC ≤ 5.5 V
		3.5625	—	39	μs		2.7 V ≤ VCC ≤ 5.5 V
		17	—	39	μs		1.8 V ≤ VCC ≤ 5.5 V
Zero-scale error ^{*1*2*3*4}	E _{ZS}	—	—	±0.25	%FSR	10-bit resolution VREFH0 = VCC ^{*5}	1.8 V ≤ VREFH0 ≤ 5.5 V
		—	—	±0.50	%FSR		1.6 V ≤ VREFH0 ≤ 5.5 V ^{*6}
Full-scale error ^{*1*2*3*4}	E _{FS}	—	—	±0.25	%FSR	10-bit resolution VREFH0 = VCC ^{*5}	1.8 V ≤ VREFH0 ≤ 5.5 V
		—	—	±0.50	%FSR		1.6 V ≤ VREFH0 ≤ 5.5 V ^{*6}
Integral linearity error ^{*1*2*3}	ILE	—	—	±2.5	LSB	10-bit resolution VREFH0 = VCC ^{*5}	1.8 V ≤ VREFH0 ≤ 5.5 V
		—	—	±5.0	LSB		1.6 V ≤ VREFH0 ≤ 5.5 V ^{*6}
Differential linearity error ^{*1*2*3}	DLE	—	—	±1.5	LSB	10-bit resolution VREFH0 = VCC ^{*5}	1.8 V ≤ VREFH0 ≤ 5.5 V
		—	—	±2.0	LSB		1.6 V ≤ VREFH0 ≤ 5.5 V ^{*6}
Analog input voltage	V _{AIN}	0	—	VREFH0	V	AN004, AN005	
		V _{BGR} ^{*7}			V	Internal reference voltage (1.8 V ≤ VCC ≤ 5.5 V)	
		V _{TMPS25} ^{*7}			V	Temperature sensor output voltage (1.8 V ≤ VCC ≤ 5.5 V)	

Note 1. The Typ. value is an average value at TA = 25°C. The Max. value is an average value ±3σ at normal distribution.
Note 2. These values are the results of characteristic evaluation and are not checked for shipment.
Note 3. This value does not include the quantization error (±1/2 LSB).
Note 4. This value is indicated as a ratio (%FSR) to the full-scale value.
Note 5. When VREFH0 < VCC, the maximum values are as follows.

表 27.38 A/D 转换特性（AN004、AN005、内部参考电压和温度传感器输出电压）（2/2）

条件: 1.6 V ≤ VREFH0 ≤ VCC ≤ 5.5 V, VSS = 0 V, Ta = -40 to +85°C
Reference voltage range applied to the VREFH0 (ADREFP[1:0] = 01b) and VREFL0 (ADREFM = 1b).
Target pins: AN004, AN005, internal reference voltage, and temperature sensor output voltage
(Reference voltage (+) = VREFH0, Reference voltage (–) = VREFL0 = 0 V)

范围	象征	最小。	类型。	最大限度。	单元	状况	
转换时间	tCONV	2.125	—	39	μs	10 位分辨率 目标引脚: AN004, AN005	3.6 V ≤ VCC ≤ 5.5 V
		3.1875	—	39	μs		2.7 V ≤ VCC ≤ 5.5 V
		17	—	39	μs		1.8 V ≤ VCC ≤ 5.5 V
		57	—	95	μs		1.6 V ≤ VCC ≤ 5.5 V
		2.375	—	39	μs	10 位分辨率 目标引脚: 内部参考	3.6 V ≤ VCC ≤ 5.5 V
		3.5625	—	39	μs		2.7 V ≤ VCC ≤ 5.5 V
		17	—	39	μs	电压和温度传感器输出电压	1.8 V ≤ VCC ≤ 5.5 V
零尺度 错误 ^{*1*2*3*4}	E _{ZS}	—	—	±0.25	%FSR	10 位分辨率 VREFH0 = VCC ^{*5}	1.8 V ≤ VREFH0 ≤ 5.5 V
		—	—	±0.50	%FSR		1.6 V ≤ VREFH0 ≤ 5.5 V ^{*6}
全面 错误 ^{*1*2*3*4}	E _{FS}	—	—	±0.25	%FSR	10 位分辨率 VREFH0 = VCC ^{*5}	1.8 V ≤ VREFH0 ≤ 5.5 V
		—	—	±0.50	%FSR		1.6 V ≤ VREFH0 ≤ 5.5 V ^{*6}
积分线 性 错误 ^{*1*2*3}	ILE	—	—	±2.5	LSB	10 位分辨率 VREFH0 = VCC ^{*5}	1.8 V ≤ VREFH0 ≤ 5.5 V
		—	—	±5.0	LSB		1.6 V ≤ VREFH0 ≤ 5.5 V ^{*6}
微分线性 错误 ^{*1*2*3}	DLE	—	—	±1.5	LSB	10 位分辨率 VREFH0 = VCC ^{*5}	1.8 V ≤ VREFH0 ≤ 5.5 V
		—	—	±2.0	LSB		1.6 V ≤ VREFH0 ≤ 5.5 V ^{*6}
模拟输入电 压	V _{AIN}	0	—	VREFH0	V	AN004, AN005	
		V _{BGR} ^{*7}			V	内部参考电压 (1.8 V ≤ VCC ≤ 5.5 V)	
		V _{TMPS25} ^{*7}			V	温度传感器输出电压 (1.8 V ≤ VCC ≤ 5.5 V)	

注 1. 典型值是TA = 25°C. 处的平均值，最大值是±3σ处的平均值。
注2. 这些数值是特性评估的结果，并非出货时的检查值。
注3.此值不包括量化误差(±1/2 LSB)。
注 4. 此值表示为(%FSR) 与满量程值的比率。
注5. 当VREFH0 < VCC, 时，最大值如下。

- Overall error: Add ± 1.0 LSB to the maximum value when $VCC = VREFH0$.
 - Zero-scale/full-scale error: Add $\pm 0.05\%$ FSR to the maximum value when $VCC = VREFH0$.
 - Integral linearity error and differential linearity error: Add ± 0.5 LSB to the maximum value when $VCC = VREFH0$.
- Note 6. The listed value applies when the settings of the maximum and minimum conversion time values are respectively 57 μ s and 95 μ s.
- Note 7. See [section 27.6.3. Temperature Sensor/Internal Reference Voltage Characteristics](#).

Table 27.39 A/D conversion characteristics (AN021 to AN022)

Conditions: $1.6\text{ V} \leq VREFH0 \leq VCC \leq 5.5\text{ V}$, $VSS = 0\text{ V}$, $Ta = -40$ to $+85^{\circ}\text{C}$
Reference voltage range applied to the $VREFH0$ (ADREFP[1:0] = 01b) and $VREFL0$ (ADREFM = 1b).
Target pins: AN021 to AN022
(Reference voltage (+) = $VREFH0$, Reference voltage (–) = $VREFL0 = 0\text{ V}$)

Parameter	Symbol	Min.	Typ.	Max.	Unit	Conditions	
Resolution	RES	8	—	10	bit	—	
Overall error*1*2*3	AINL	—	1.2	± 5.0	LSB	10-bit resolution $VREFH0 = VCC^{*5}$	$1.8\text{ V} \leq VREFH0 \leq 5.5\text{ V}$
		—	1.2	± 8.5	LSB		$1.6\text{ V} \leq VREFH0 \leq 5.5\text{ V}^{*6}$
Conversion time	tCONV	2.125	—	39	μ s	10-bit resolution Target pin: AN021, AN022	$3.6\text{ V} \leq VCC \leq 5.5\text{ V}$
		3.1875	—	39	μ s		$2.7\text{ V} \leq VCC \leq 5.5\text{ V}$
		17	—	39	μ s		$1.8\text{ V} \leq VCC \leq 5.5\text{ V}$
		57	—	95	μ s		$1.6\text{ V} \leq VCC \leq 5.5\text{ V}$
Zero-scale error*1*2*3*4	E _{ZS}	—	—	± 0.35	%FSR	10-bit resolution $VREFH0 = VCC^{*5}$	$1.8\text{ V} \leq VREFH0 \leq 5.5\text{ V}$
		—	—	± 0.60	%FSR		$1.6\text{ V} \leq VREFH0 \leq 5.5\text{ V}^{*6}$
Full-scale error*1*2*3*4	E _{FS}	—	—	± 0.35	%FSR	10-bit resolution $VREFH0 = VCC^{*5}$	$1.8\text{ V} \leq VREFH0 \leq 5.5\text{ V}$
		—	—	± 0.60	%FSR		$1.6\text{ V} \leq VREFH0 \leq 5.5\text{ V}^{*6}$
Integral linearity error*1*2*3	ILE	—	—	± 3.5	LSB	10-bit resolution $VREFH0 = VCC^{*5}$	$1.8\text{ V} \leq VREFH0 \leq 5.5\text{ V}$
		—	—	± 6.0	LSB		$1.6\text{ V} \leq VREFH0 \leq 5.5\text{ V}^{*6}$
Differential linearity error*1*2*3	DLE	—	—	± 2.0	LSB	10-bit resolution $VREFH0 = VCC^{*5}$	$1.8\text{ V} \leq VREFH0 \leq 5.5\text{ V}$
		—	—	± 2.5	LSB		$1.6\text{ V} \leq VREFH0 \leq 5.5\text{ V}^{*6}$
Analog input voltage	V _{AIN}	0	—	VREFH0	V	AN021, AN022	

- Note 1. The Typ. value is an average value at $Ta = 25^{\circ}\text{C}$. The Max. value is an average value $\pm 3\sigma$ at normal distribution.
- Note 2. These values are the results of characteristic evaluation and are not checked for shipment.
- Note 3. This value does not include the quantization error ($\pm 1/2$ LSB).
- Note 4. This value is indicated as a ratio (%FSR) to the full-scale value.
- Note 5. When $VREFH0 < VCC$, the maximum values are as follows.

- Overall error: Add ± 1.0 LSB to the maximum value when $VCC = VREFH0$.

- 总体误差：当 $VCC = VREFH0$ 时，将 ± 1.0 LSB添加到最大值。
 - 零刻度/满刻度误差：当 $VCC = VREFH0$ 时，将 $\pm 0.05\%$ FSR加到最大值。
 - 积分线性误差和微分线性误差：当 $VCC = VREFH0$ 时，将 ± 0.5 LSB 加到最大值。
- 注6：所列值适用于最大和最小转换时间值分别设置为57 μ s和95 μ s的情况。
- 注7.参见第27.6.3节。温度传感器/内部参考电压特性。

表 27.39 A/D 转换特性（AN021 至 AN022）

条件: $1.6\text{ V} \leq VREFH0 \leq VCC \leq 5.5\text{ V}$, $VSS = 0\text{ V}$, $Ta = -40$ 至 $+85^{\circ}\text{C}$ 参考电压范围应用于 $VREFH0$ (ADREFP[1:0] = 01b) 和 $VREFL0$ (ADREFM= 1b)。
目标引脚: AN021 至 AN022
(参考电压(+) = $VREFH0$, 参考电压(–) = $VREFL0 = 0\text{ V}$)

范围	象征		最小。	类型。	最大限度。	单元	状况	
解决	RES		8	—	10	少量	—	
全面的 错误*1*2*3	AINL		—	1.2	± 5.0	LSB	10 位分辨率 $VREFH0 = VCC^{*5}$	$1.8\text{ V} \leq VREFH0 \leq 5.5\text{ V}$
			—	1.2	± 8.5	LSB		$1.6\text{ V} \leq VREFH0 \leq 5.5\text{ V}^{*6}$
转换时间	tCONV		2.125	—	39	μ s	10 位分辨率 目标引脚: AN021, AN022	$3.6\text{ V} \leq VCC \leq 5.5\text{ V}$
			3.1875	—	39	μ s		$2.7\text{ V} \leq VCC \leq 5.5\text{ V}$
			17	—	39	μ s		$1.8\text{ V} \leq VCC \leq 5.5\text{ V}$
			57	—	95	μ s		$1.6\text{ V} \leq VCC \leq 5.5\text{ V}$
零尺度 错误*1*2*3*4	E _{ZS}		—	—	± 0.35	%FSR	10 位分辨率 $VREFH0 = VCC^{*5}$	$1.8\text{ V} \leq VREFH0 \leq 5.5\text{ V}$
			—	—	± 0.60	%FSR		$1.6\text{ V} \leq VREFH0 \leq 5.5\text{ V}^{*6}$
全面 错误*1*2*3*4	E _{FS}		—	—	± 0.35	%FSR	10 位分辨率 $VREFH0 = VCC^{*5}$	$1.8\text{ V} \leq VREFH0 \leq 5.5\text{ V}$
			—	—	± 0.60	%FSR		$1.6\text{ V} \leq VREFH0 \leq 5.5\text{ V}^{*6}$
积分线 性 错误*1*2*3	ILE		—	—	± 3.5	LSB	10 位分辨率 $VREFH0 = VCC^{*5}$	$1.8\text{ V} \leq VREFH0 \leq 5.5\text{ V}$
			—	—	± 6.0	LSB		$1.6\text{ V} \leq VREFH0 \leq 5.5\text{ V}^{*6}$
微分线性 错误*1*2*3	DLE		—	—	± 2.0	LSB	10 位分辨率 $VREFH0 = VCC^{*5}$	$1.8\text{ V} \leq VREFH0 \leq 5.5\text{ V}$
			—	—	± 2.5	LSB		$1.6\text{ V} \leq VREFH0 \leq 5.5\text{ V}^{*6}$
模拟输入电 压	V _{AIN}		0	—	VREFH0	V	AN021, AN022	

- 注 1. 典型值是 $Ta = 25^{\circ}\text{C}$. 处的平均值，最大值是 $\pm 3\sigma$ 处的平均值。
- 注2. 这些数值是特性评估的结果，并非出货时的检查值。
- 注3.此值不包括量化误差($\pm 1/2$ LSB)。
- 注 4. 此值表示为(%FSR)与满量程值的比率。
- 注5. 当 $VREFH0 < VCC$ 时，最大值如下。

- 总体误差：当 $VCC = VREFH0$ 时，将 ± 1.0 LSB加到最大值。

- Zero-scale/full-scale error: Add $\pm 0.05\%$ FSR to the maximum value when $VCC = VREFH0$.
 - Integral linearity error/differential linearity error: Add ± 0.5 LSB to the maximum value when $VCC = VREFH0$.
- Note 6. The listed value applies when the settings of the maximum and minimum conversion time values are respectively 57 μs and 95 μs .

Table 27.40 A/D conversion characteristics (AN000, AN001, AN004, AN005, AN021, AN022, internal reference voltage, and temperature sensor output voltage) (1 of 2)

Conditions: $1.6\text{ V} \leq VCC \leq 5.5\text{ V}$, $VSS = 0\text{ V}$, $Ta = -40$ to $+85^\circ\text{C}$
Reference voltage range applied to the VCC (ADREFP[1:0] = 00b) and VSS (ADREFM = 0b).
Target pins: AN000, AN001, AN004, AN005, AN021, AN022, internal reference voltage*7, and temperature sensor output voltage*7
(Reference voltage (+) = VCC, Reference voltage (–) = VSS)

Parameter	Symbol	Min.	Typ.	Max.	Unit	Conditions	
Resolution	RES	8	—	10	bit	—	
Overall error*1*2*3	AINL	—	1.2	± 7.0	LSB	10-bit resolution	$1.8\text{ V} \leq VREFH0 \leq 5.5\text{ V}$
		—	1.2	± 10.5	LSB		$1.6\text{ V} \leq VREFH0 \leq 5.5\text{ V}^5$
Conversion time	tCONV	2.125	—	39	μs	10-bit resolution Target pin: AN000, AN001, AN004, AN005, AN021, AN022	$3.6\text{ V} \leq VCC \leq 5.5\text{ V}$
		3.1875	—	39	μs		$2.7\text{ V} \leq VCC \leq 5.5\text{ V}$
		17	—	39	μs		$1.8\text{ V} \leq VCC \leq 5.5\text{ V}$
		57	—	95	μs		$1.6\text{ V} \leq VCC \leq 5.5\text{ V}$
		2.375	—	39	μs	10-bit resolution Target pin: Internal reference voltage, and temperature sensor output voltage	$3.6\text{ V} \leq VCC \leq 5.5\text{ V}$
		3.5625	—	39	μs		$2.7\text{ V} \leq VCC \leq 5.5\text{ V}$
		17	—	39	μs		$1.8\text{ V} \leq VCC \leq 5.5\text{ V}$
Zero-scale error*1*2*3*4	Ezs	—	—	± 0.60	%FSR	10-bit resolution	$1.8\text{ V} \leq VCC \leq 5.5\text{ V}$
		—	—	± 0.85	%FSR		$1.6\text{ V} \leq VCC \leq 5.5\text{ V}^5$
Full-scale error*1*2*3*4	Efs	—	—	± 0.60	%FSR	10-bit resolution	$1.8\text{ V} \leq VCC \leq 5.5\text{ V}$
		—	—	± 0.85	%FSR		$1.6\text{ V} \leq VCC \leq 5.5\text{ V}^5$
Integral linearity error*1*2*3	ILE	—	—	± 4.0	LSB	10-bit resolution	$1.8\text{ V} \leq VCC \leq 5.5\text{ V}$
		—	—	± 6.0	LSB		$1.6\text{ V} \leq VCC \leq 5.5\text{ V}^5$
Differential linearity error*1*2*3	DLE	—	—	± 2.0	LSB	10-bit resolution	$1.8\text{ V} \leq VCC \leq 5.5\text{ V}$
		—	—	± 2.5	LSB		$1.6\text{ V} \leq VCC \leq 5.5\text{ V}^5$

- 零刻度/满刻度误差：当 $VCC = VREFH0$ 时，将 $\pm 0.05\%$ FSR 加到最大值。
 - 积分线性误差/微分线性误差：当 $VCC = VREFH0$ 时，将 ± 0.5 LSB 加到最大值。
- 注6：所列值适用于最大和最小转换时间值分别设置为57 μs 和95 μs 的情况。

表 27.40A/D 转换特性（AN000、AN001、AN004、AN005、AN021、AN022、内部参考电压和温度传感器输出电压）（1/2）

条件: $1.6\text{ V} \leq VCC \leq 5.5\text{ V}$ $VSS = 0\text{ V}$ $Ta = -40$ 至 $+85^\circ\text{C}$
应用于 VCC 的参考电压范围（ADREFP[1:0] = 00b）和 VSS(ADREFM = 0b)。
目标引脚: AN000、AN001、AN004、AN005、AN021、AN022、内部参考电压*7 和温度传感器输出电压*7
(参考电压(+) = VCC, 参考电压(–) = VSS))

范围	象征	最小。	类型。	最大限度。	单元	状况	
解决	RES	8	—	10	少量	—	
全面的 错误*1*2*3	AINL	—	1.2	± 7.0	LSB	10 位分辨率	$1.8\text{ V} \leq VREFH0 \leq 5.5\text{ V}$
		—	1.2	± 10.5	LSB		$1.6\text{ V} \leq VREFH0 \leq 5.5\text{ V}^5$
转换时间	tCONV	2.125	—	39	μs	10 位分辨率 目标引脚: AN000, AN001, AN004, AN005, AN021, AN022	$3.6\text{ V} \leq VCC \leq 5.5\text{ V}$
		3.1875	—	39	μs		$2.7\text{ V} \leq VCC \leq 5.5\text{ V}$
		17	—	39	μs		$1.8\text{ V} \leq VCC \leq 5.5\text{ V}$
		57	—	95	μs		$1.6\text{ V} \leq VCC \leq 5.5\text{ V}$
		2.375	—	39	μs	10 位分辨率 目标引脚: 内部参考 电压和温度传 感器输出电压	$3.6\text{ V} \leq VCC \leq 5.5\text{ V}$
		3.5625	—	39	μs		$2.7\text{ V} \leq VCC \leq 5.5\text{ V}$
		17	—	39	μs		$1.8\text{ V} \leq VCC \leq 5.5\text{ V}$
零尺度 错误*1*2*3*4	Ezs	—	—	± 0.60	%FSR	10 位分辨率	$1.8\text{ V} \leq VCC \leq 5.5\text{ V}$
		—	—	± 0.85	%FSR		$1.6\text{ V} \leq VCC \leq 5.5\text{ V}^5$
全面 错误*1*2*3*4	Efs	—	—	± 0.60	%FSR	10 位分辨率	$1.8\text{ V} \leq VCC \leq 5.5\text{ V}$
		—	—	± 0.85	%FSR		$1.6\text{ V} \leq VCC \leq 5.5\text{ V}^5$
积分线 性 错误*1*2*3	ILE	—	—	± 4.0	LSB	10 位分辨率	$1.8\text{ V} \leq VCC \leq 5.5\text{ V}$
		—	—	± 6.0	LSB		$1.6\text{ V} \leq VCC \leq 5.5\text{ V}^5$
微分线性 错误*1*2*3	DLE	—	—	± 2.0	LSB	10 位分辨率	$1.8\text{ V} \leq VCC \leq 5.5\text{ V}$
		—	—	± 2.5	LSB		$1.6\text{ V} \leq VCC \leq 5.5\text{ V}^5$

Table 27.40 A/D conversion characteristics (AN000, AN001, AN004, AN005, AN021, AN022, internal reference voltage, and temperature sensor output voltage) (2 of 2)

Conditions: 1.6 V ≤ VCC ≤ 5.5 V, VSS = 0 V, Ta = -40 to +85°C
Reference voltage range applied to the VCC (ADREFP[1:0] = 00b) and VSS (ADREFM = 0b).
Target pins: AN000, AN001, AN004, AN005, AN021, AN022, internal reference voltage^{*7}, and temperature sensor output voltage^{*7}
(Reference voltage (+) = VCC, Reference voltage (–) = VSS)

Parameter	Symbol	Min.	Typ.	Max.	Unit	Conditions
Analog input voltage	V _{AIN}	0	—	VCC	V	AN000, AN001, AN004, AN005, AN021, AN022
		V _{BGR} ^{*6}			V	Internal reference voltage (1.8 V ≤ VCC ≤ 5.5 V)
		V _{TMPS25} ^{*6}			V	Temperature sensor output voltage (1.8 V ≤ VCC ≤ 5.5 V)

Note 1. The Typ. value is an average value at TA = 25°C. The Max. value is an average value ±3σ at normal distribution.
Note 2. These values are the results of characteristic evaluation and are not checked for shipment.
Note 3. This value does not include the quantization error (±1/2 LSB).
Note 4. This value is indicated as a ratio (%FSR) to the full-scale value.
Note 5. The listed value applies when the settings of the maximum and minimum conversion time values are respectively 57 μs and 95 μs.
Note 6. See [section 27.6.3. Temperature Sensor/Internal Reference Voltage Characteristics](#).
Note 7. If the internal reference voltage or temperature sensor output voltage is to be A/D converted, VCC must be at least 1.8 V.

Table 27.41 A/D conversion characteristics (AN000, AN004, AN005, AN021, AN022)

Conditions: 1.8 V ≤ VREFH0 ≤ VCC ≤ 5.5 V, VSS = 0 V, Ta = -40 to +85°C
Reference voltage range applied to the VBGR (ADREFP[1:0] = 10b) and VREFL0 (ADREFM = 1b).
Target pins: AN000, AN004, AN005, AN021, AN022
(Reference voltage (+) = V_{BGR}^{*5}, Reference voltage (–) = VREFL0^{*6} = 0 V)

Parameter	Symbol	Min.	Typ.	Max.	Unit	Conditions
Resolution	RES	8			bit	—
Conversion time	tCONV	17	—	39	μs	—
Zero-scale error ^{*1*2*3*4}	E _{ZS}	—	—	±0.60	%FSR	—
Integral linearity error ^{*1*2*3}	ILE	—	—	±2.0	LSB	—
Differential linearity error ^{*1*2*3}	DLE	—	—	±1.0	LSB	—
Analog input voltage	V _{AIN}	0	—	V _{BGR} ^{*5}	V	—

Note 1. The Typ. value is an average value at TA = 25°C. The MAX. value is an average value ±3σ at normal distribution.
Note 2. These values are the results of characteristic evaluation and are not checked for shipment.
Note 3. This value does not include the quantization error (±1/2 LSB).
Note 4. This value is indicated as a ratio (%FSR) to the full-scale value.
Note 5. See [section 27.6.3. Temperature Sensor/Internal Reference Voltage Characteristics](#).
Note 6.

- Zero-scale error: Add ±0.35%FSR to the maximum value when reference voltage (–) = VREFL0.
- Integral linearity error: Add ±0.5 LSB to the maximum value when reference voltage (–) = VREFL0.
- Differential linearity error: Add ±0.2 LSB to the maximum value when reference voltage (–) = VREFL0.

27.6.2 A/D Converter Characteristics for T_A = -40 to +125°C

Table 27.42 Reference for the characteristics of the A/D converter

Input channel	Reference Voltage		
	Reference voltage (+) = VREFH0 Reference voltage (–) = VREFL0	Reference voltage (+) = VCC Reference voltage (–) = VSS	Reference voltage (+) = V _{BGR} Reference voltage (–) = VREFL0
AN000, AN001, AN004, AN005	Refer to Table 27.43 .	Refer to Table 27.45 .	Refer to Table 27.46 .
AN021, AN022	Refer to Table 27.44 .		
Internal reference voltage Temperature sensor output voltage	Refer to Table 27.43 .		—

表 27.40A/D 转换特性（AN000、AN001、AN004、AN005、AN021、AN022、内部参考电压和温度传感器输出电压）（2/2）

Conditions: 1.6 V ≤ VCC ≤ 5.5 V, VSS = 0 V, Ta = -40 to +85°C
Reference voltage range applied to the VCC (ADREFP[1:0] = 00b) and VSS (ADREFM = 0b).
Target pins: AN000, AN001, AN004, AN005, AN021, AN022, internal reference voltage^{*7}, and temperature sensor output voltage^{*7}
(参考电压(+) = VCC, 参考电压(–) = VSS)

范围	象征	最小。	类型。	最大限度。	单元	状况
模拟输入电压	V _{AIN}	0	—	VCC	V	AN000、AN001、AN004、AN005、AN021、AN022
		V _{BGR} ^{*6}			V	内部参考电压 (1.8 V ≤ VCC ≤ 5.5 V)
		V _{TMPS25} ^{*6}			V	温度传感器输出电压 (1.8 V ≤ VCC ≤ 5.5 V)

注 1. 典型值是TA = 25°C. 处的平均值，最大值是±3σ处的平均值。
注2. 这些数值是特性评估的结果，并非出货时的检查值。
注3.此值不包括量化误差(±1/2 LSB)。
注 4. 此值表示为(%FSR) 与满量程值的比率。
注5： 所列值适用于最大和最小转换时间值分别设置为57 μs和95 μs的情况。
注 6. 请参阅第 27.6.3 节。温度传感器/内部参考电压特性。
注7. 如果内部参考电压或温度传感器输出电压需要进行A/D转换，则VCC必须至少为1.8 V.

表 27.41A/D 转换特性（AN000、AN004、AN005、AN021、AN022）

条件: 1.8 V ≤ VREFH0 ≤ VCC ≤ 5.5 V、VSS = 0 V、Ta = -40至+85°C
应用于 VBGR (ADREFP[1:0] = 10b) 和 VREFL0 (ADREFM= 1b) 的参考电压范围。
目标引脚: AN000、AN004、AN005、AN021、AN022
(参考电压(+) = V_{BGR}^{*5}, 参考电压(–) = VREFL0^{*6} = 0 V)

范围	象征	最小。	类型。	最大限度。	单元	状况
解决	RES	8			少量	—
转换时间	tCONV	17	—	39	μs	—
零刻度误差*1*2*3*4	E _{ZS}	—	—	±0.60	%FSR	—
积分线性误差*1*2*3	ILE	—	—	±2.0	LSB	—
微分线性误差*1*2*3	DLE	—	—	±1.0	LSB	—
模拟输入电压	V _{AIN}	0	—	V _{BGR} ^{*5}	V	—

注 1. 典型值是TA = 25°C. 处的平均值，最大值是±3σ处的平均值。
注2. 这些数值是特性评估的结果，并非出货时的检查值。
注3.此值不包括量化误差(±1/2 LSB)。
注 4. 此值表示为(%FSR)与满量程值的比率。
注5.参见第27.6.3节。温度传感器/内部参考电压特性。
注6. ●零刻度误差：当参考电压(–) = VREFL0时，将±0.35%FSR加到最大值。
●积分线性误差：当参考电压(–) = VREFL0时，将±0.5 LSB加到最大值。
●差分线性误差：当参考电压(–) = VREFL0 时，将±0.2LSB 加到最大值。

27.6.2T_A = -40 to +125°C的 A/D 转换器特性

表 27.42 A/D 转换器特性参考

输入通道	参考电压		
	参考电压(+) = VREFH0 参考电压(–) = VREFL0	参考电压(+) = VCC 参考电压(–) = VSS	参考电压(+) = V _{BGR} 参考电压(–) = VREFL0
AN000、AN001、AN004、AN005	请参阅表 27.43。	请参阅表 27.45。	请参阅表 27.46。
AN021, AN022	请参阅表 27.44。		
内部参考电压温度传感器 输出电压	请参阅表 27.43。		—

Table 27.43 A/D conversion characteristics

Conditions: $2.4\text{ V} \leq \text{VREFH0} \leq \text{VCC} \leq 5.5\text{ V}$, $\text{VSS} = 0\text{ V}$, $\text{Ta} = -40\text{ to }+125^\circ\text{C}$
Reference voltage range applied to the VREFH0 (ADREFP[1:0] = 01b) and VREFL0 (ADREFM = 1b).
Target pins: AN004, AN005, internal reference voltage, and temperature sensor output voltage
(Reference voltage (+) = VREFH0, Reference voltage (-) = VREFL0 = 0 V)

Parameter	Symbol	Min.	Typ.	Max.	Unit	Conditions	
Resolution	RES	8	—	10	bit	—	
Overall error*1*2*3	AINL	—	1.2	±3.5	LSB	10-bit resolution VREFH0 = VCC*5	$2.4\text{ V} \leq \text{VREFH0} \leq 5.5\text{ V}$
Conversion time	tCONV	2.125	—	39	μs	10-bit resolution Target pin: AN004, AN005	$3.6\text{ V} \leq \text{VCC} \leq 5.5\text{ V}$
		3.1875	—	39	μs		$2.7\text{ V} \leq \text{VCC} \leq 5.5\text{ V}$
		17	—	39	μs		$2.4\text{ V} \leq \text{VCC} \leq 5.5\text{ V}$
		2.375	—	39	μs	10-bit resolution Target pin: Internal reference voltage, and temperature sensor output voltage	$3.6\text{ V} \leq \text{VCC} \leq 5.5\text{ V}$
		3.5625	—	39	μs		$2.7\text{ V} \leq \text{VCC} \leq 5.5\text{ V}$
		17	—	39	μs		$2.4\text{ V} \leq \text{VCC} \leq 5.5\text{ V}$
Zero-scale error*1*2*3*4	Ezs	—	—	±0.25	%FSR	10-bit resolution VREFH0 = VCC*5	$2.4\text{ V} \leq \text{VREFH0} \leq 5.5\text{ V}$
Full-scale error*1*2*3*4	EFS	—	—	±0.25	%FSR	10-bit resolution VREFH0 = VCC*5	$2.4\text{ V} \leq \text{VREFH0} \leq 5.5\text{ V}$
Integral linearity error*1*2*3	ILE	—	—	±2.5	LSB	10-bit resolution VREFH0 = VCC*5	$2.4\text{ V} \leq \text{VREFH0} \leq 5.5\text{ V}$
Differential linearity error*1*2*3	DLE	—	—	±1.5	LSB	10-bit resolution VREFH0 = VCC*5	$2.4\text{ V} \leq \text{VREFH0} \leq 5.5\text{ V}$
Analog input voltage	VAIN	0	—	VREFH0	V	AN004, AN005	
		VBGR*6			V	Internal reference voltage	
		VTMPS25*6			V	Temperature sensor output voltage	

Note 1. The Typ. value is an average value at TA = 25°C. The Max. value is an average value ±3σ at normal distribution.
Note 2. These values are the results of characteristic evaluation and are not checked for shipment.
Note 3. This value does not include the quantization error (±1/2 LSB).
Note 4. This value is indicated as a ratio (%FSR) to the full-scale value.
Note 5. When VREFH0 < VCC, the maximum values are as follows.
 • Overall error: Add ±1.0 LSB to the maximum value when VCC = VREFH0.
 • Zero-scale/full-scale error: Add ±0.05%FSR to the maximum value when VCC = VREFH0.
 • Integral linearity error and differential linearity error: Add ±0.5 LSB to the maximum value when VCC = VREFH0.
Note 6. See [section 27.6.3. Temperature Sensor/Internal Reference Voltage Characteristics](#).

表 27.43 A/D 转换特性

条件: $2.4\text{ V} \leq \text{VREFH0} \leq \text{VCC} \leq 5.5\text{ V}$ 、 $\text{VSS} = 0\text{ V}$ $\text{Ta} = -40\text{至}+125^\circ\text{C}$ 参考电压范围应用于 VREFH0 (ADREFP[1:0] = 01b) 和 VREFL0 (ADREFM= 1b)。
目标引脚: AN004、AN005、内部参考电压和温度传感器输出电压 (参考电压(+)= VREFH0, 参考电压(-)= VREFL0 = 0 V)

范围	象征	最小。	类型。	最大限度。	单元	状况	
解决	RES	8	—	10	少量	—	
全面的 错误*1*2*3	AINL	—	1.2	±3.5	LSB	10 位分辨率 VREFH0=VCC*5	$2.4\text{ V} \leq \text{VREFH0} \leq 5.5\text{ V}$
转换时间	tCONV	2.125	—	39	μs	10 位分辨率 目标引脚: AN004, AN005	$3.6\text{ V} \leq \text{VCC} \leq 5.5\text{ V}$
		3.1875	—	39	μs		$2.7\text{ V} \leq \text{VCC} \leq 5.5\text{ V}$
		17	—	39	μs		$2.4\text{ V} \leq \text{VCC} \leq 5.5\text{ V}$
		2.375	—	39	μs	10 位分辨率 目标引脚: 内部参考 电压和温度传感器输出电压	$3.6\text{ V} \leq \text{VCC} \leq 5.5\text{ V}$
		3.5625	—	39	μs		$2.7\text{ V} \leq \text{VCC} \leq 5.5\text{ V}$
		17	—	39	μs		$2.4\text{ V} \leq \text{VCC} \leq 5.5\text{ V}$
零尺度 错误*1*2*3*4	Ezs	—	—	±0.25	%FSR	10 位分辨率 VREFH0=VCC*5	$2.4\text{ V} \leq \text{VREFH0} \leq 5.5\text{ V}$
全面 错误*1*2*3*4	EFS	—	—	±0.25	%FSR	10 位分辨率 VREFH0 = VCC*5	$2.4\text{ V} \leq \text{VREFH0} \leq 5.5\text{ V}$
积分线性 错误*1*2*3	ILE	—	—	±2.5	LSB	10 位分辨率 VREFH0 = VCC*5	$2.4\text{ V} \leq \text{VREFH0} \leq 5.5\text{ V}$
微分线性 错误*1*2*3	DLE	—	—	±1.5	LSB	10 位分辨率 VREFH0=VCC*5	$2.4\text{ V} \leq \text{VREFH0} \leq 5.5\text{ V}$
模拟输入电压	VAIN	0	—	VREFH0	V	AN004, AN005	
		VBGR*6			V	内部参考电压	
		VTMPS25*6			V	温度传感器输出电压	

注 1. 典型值是TA = 25°C. 处的平均值，最大值是±3σ处的平均值。
注2. 这些数值是特性评估的结果，并非出货时的检查值。
注3.此值不包括量化误差(±1/2 LSB)。
注 4. 此值表示为(%FSR)与满量程值的比率。
注5. 当VREFH0 < VCC时，最大值如下。
 ● 总体误差: 当 VCC=VREFH0 时，将±1.0 LSB添加到最大值。
 ● 零刻度/满刻度误差: 当VCC = VREFH0时，将±0.05%FSR加到最大值。
 ● 积分线性误差和微分线性误差: 当 VCC=VREFH0 时，将±0.5 LSB加到最大值。
注 6. 请参阅第 27.6.3 节。温度传感器/内部参考电压特性。

Table 27.44 A/D conversion characteristics (AN021 to AN022)

Conditions: $2.4\text{ V} \leq \text{VREFH0} \leq \text{VCC} \leq 5.5\text{ V}$, $\text{VSS} = 0\text{ V}$, $\text{Ta} = -40\text{ to }+125^\circ\text{C}$
Reference voltage range applied to the VREFH0 (ADREFP[1:0] = 01b) and VREFL0 (ADREFM = 1b).
Target pins: AN021 to AN022
(Reference voltage (+) = VREFH0, Reference voltage (–) = VREFL0 = 0 V)

Parameter	Symbol	Min.	Typ.	Max.	Unit	Conditions	
Resolution	RES	8	—	10	bit	—	
Overall error*1*2*3	AINL	—	1.2	±5.0	LSB	10-bit resolution VREFH0 = VCC*5	$2.4\text{ V} \leq \text{VREFH0} \leq 5.5\text{ V}$
Conversion time	tCONV	2.125	—	39	μs	10-bit resolution Target pin: AN021, AN022	$3.6\text{ V} \leq \text{VCC} \leq 5.5\text{ V}$
		3.1875	—	39	μs		$2.7\text{ V} \leq \text{VCC} \leq 5.5\text{ V}$
		17	—	39	μs		$2.4\text{ V} \leq \text{VCC} \leq 5.5\text{ V}$
Zero-scale error*1*2*3*4	Ezs	—	—	±0.35	%FSR	10-bit resolution VREFH0 = VCC*5	$2.4\text{ V} \leq \text{VREFH0} \leq 5.5\text{ V}$
Full-scale error*1*2*3*4	Efs	—	—	±0.35	%FSR	10-bit resolution VREFH0 = VCC*5	$2.4\text{ V} \leq \text{VREFH0} \leq 5.5\text{ V}$
Integral linearity error*1*2*3	ILE	—	—	±3.5	LSB	10-bit resolution VREFH0 = VCC*5	$2.4\text{ V} \leq \text{VREFH0} \leq 5.5\text{ V}$
Differential linearity error*1*2*3	DLE	—	—	±2.0	LSB	10-bit resolution VREFH0 = VCC*5	$2.4\text{ V} \leq \text{VREFH0} \leq 5.5\text{ V}$
Analog input voltage	VAIN	0	—	VREFH0	V	AN021, AN022	

Note 1. The Typ. value is an average value at TA = 25°C. The Max. value is an average value ±3σ at normal distribution.
Note 2. These values are the results of characteristic evaluation and are not checked for shipment.
Note 3. This value does not include the quantization error (±1/2 LSB).
Note 4. This value is indicated as a ratio (%FSR) to the full-scale value.
Note 5. When VREFH0 < VCC, the maximum values are as follows.

- Overall error: Add ±1.0 LSB to the maximum value when VCC = VREFH0.
- Zero-scale/full-scale error: Add ±0.05%FSR to the maximum value when VCC = VREFH0.
- Integral linearity error and differential linearity error: Add ±0.5 LSB to the maximum value when VCC = VREFH0.

Table 27.45 A/D conversion characteristics (1 of 2)

Conditions: $2.4\text{ V} \leq \text{VCC} \leq 5.5\text{ V}$, $\text{VSS} = 0\text{ V}$, $\text{Ta} = -40\text{ to }+125^\circ\text{C}$
Reference voltage range applied to the VCC (ADREFP[1:0] = 00b) and VSS (ADREFM = 0b).
Target pins: AN000, AN001, AN004, AN005, AN021, AN022, internal reference voltage, and temperature sensor output voltage
(Reference voltage (+) = VCC, Reference voltage (–) = VSS)

Parameter	Symbol	Min.	Typ.	Max.	Unit	Conditions	
Resolution	RES	8	—	10	bit	—	
Overall error*1*2*3	AINL	—	1.2	±7.0	LSB	10-bit resolution	$2.4\text{ V} \leq \text{VREFH0} \leq 5.5\text{ V}$

表 27.44A/D 转换特性 (AN021 至 AN022)

条件: $2.4\text{ V} \leq \text{VREFH0} \leq \text{VCC} \leq 5.5\text{ V}$, $\text{VSS} = 0\text{ V}$, $\text{Ta} = -40\text{ 至 }+125^\circ\text{C}$ 参考电压范围应用于 VREFH0 (ADREFP[1:0] = 01b) 和 VREFL0 (ADREFM = 1b)。
目标引脚: AN021 至 AN022
(参考电压(+) = VREFH0, 参考电压(–) = VREFL0 = 0 V)

范围	象征	最小。	类型。	最大限度。	单元	状况	
解决	RES	8	—	10	少量	—	
全面的 错误*1*2*3	AINL	—	1.2	±5.0	LSB	10 位分辨率 VREFH0=VCC*5	$2.4\text{ V} \leq \text{VREFH0} \leq 5.5\text{ V}$
转换时间	tCONV	2.125	—	39	μs	10 位分辨率 目标引脚: AN021, AN022	$3.6\text{ V} \leq \text{VCC} \leq 5.5\text{ V}$
		3.1875	—	39	μs		$2.7\text{ V} \leq \text{VCC} \leq 5.5\text{ V}$
		17	—	39	μs		$2.4\text{ V} \leq \text{VCC} \leq 5.5\text{ V}$
零尺度 错误*1*2*3*4	Ezs	—	—	±0.35	%FSR	10 位分辨率 VREFH0=VCC*5	$2.4\text{ V} \leq \text{VREFH0} \leq 5.5\text{ V}$
全面 错误*1*2*3*4	Efs	—	—	±0.35	%FSR	10 位分辨率 VREFH0=VCC*5	$2.4\text{ V} \leq \text{VREFH0} \leq 5.5\text{ V}$
积分线性 误差*1*2*3	ILE	—	—	±3.5	LSB	10 位分辨率 VREFH0=VCC*5	$2.4\text{ V} \leq \text{VREFH0} \leq 5.5\text{ V}$
微分线性 错误*1*2*3	DLE	—	—	±2.0	LSB	10 位分辨率 VREFH0=VCC*5	$2.4\text{ V} \leq \text{VREFH0} \leq 5.5\text{ V}$
模拟输入电压	VAIN	0	—	VREFH0	V	AN021, AN022	

注 1. 典型值是TA = 25°C. 处的平均值，最大值是±3σ处的平均值。
注2. 这些数值是特性评估的结果，并非出货时的检查值。
注3.此值不包括量化误差(±1/2 LSB)。
注 4. 此值表示为(%FSR)与满量程值的比率。
注5. 当VREFH0 < VCC, 时，最大值如下。

- 总体误差: 当VCC = VREFH0时，将±1.0LSB 加到最大值。
- 零刻度/满刻度误差: 当VCC = VREFH0时，将±0.05%FSR加到最大值。
- 积分线性误差和微分线性误差: 当 VCC= VREFH0 时，将±0.5LSB 加到最大值。

表 27.45 A/D 转换特性 (1/2)

条件: $2.4\text{ V} \leq \text{VCC} \leq 5.5\text{ V}$, $\text{VSS} = 0\text{ V}$, $\text{Ta} = -40\text{ 至 }+125^\circ\text{C}$
应用于 VCC 的参考电压范围 (ADREFP[1:0] = 00b) 和 VSS(ADREFM = 0b))。
目标引脚: AN000、AN001、AN004、AN005、AN021、AN022、内部参考电压和温度传感器输出电压
(参考电压(+) = VCC, 参考电压(–) = VSS))

范围	象征	最小。	类型。	最大限度。	单元	状况	
解决	RES	8	—	10	少量	—	
全面的 错误*1*2*3	AINL	—	1.2	±7.0	LSB	10 位分辨率	$2.4\text{ V} \leq \text{VREFH0} \leq 5.5\text{ V}$

Table 27.45 A/D conversion characteristics (2 of 2)

Conditions: $2.4\text{ V} \leq V_{CC} \leq 5.5\text{ V}$, $V_{SS} = 0\text{ V}$, $T_a = -40$ to $+125^\circ\text{C}$
Reference voltage range applied to the VCC (ADREFP[1:0] = 00b) and VSS (ADREFM = 0b).
Target pins: AN000, AN001, AN004, AN005, AN021, AN022, internal reference voltage, and temperature sensor output voltage
(Reference voltage (+) = VCC, Reference voltage (–) = VSS)

Parameter	Symbol	Min.	Typ.	Max.	Unit	Conditions	
Conversion time	tCONV	2.125	—	39	μs	10-bit resolution	$3.6\text{ V} \leq V_{CC} \leq 5.5\text{ V}$
		3.1875	—	39	μs	Target pin: AN000, AN001, AN004, AN005, AN021, AN022	$2.7\text{ V} \leq V_{CC} \leq 5.5\text{ V}$
		17	—	39	μs		$2.4\text{ V} \leq V_{CC} \leq 5.5\text{ V}$
		2.375	—	39	μs	10-bit resolution	$3.6\text{ V} \leq V_{CC} \leq 5.5\text{ V}$
		3.5625	—	39	μs	Target pin: Internal reference voltage, and temperature sensor output voltage	$2.7\text{ V} \leq V_{CC} \leq 5.5\text{ V}$
		17	—	39	μs		$2.4\text{ V} \leq V_{CC} \leq 5.5\text{ V}$
Zero-scale error*1*2*3*4	Ezs	—	—	±0.60	%FSR	10-bit resolution	$2.4\text{ V} \leq V_{CC} \leq 5.5\text{ V}$
Full-scale error*1*2*3*4	Efs	—	—	±0.60	%FSR	10-bit resolution	$2.4\text{ V} \leq V_{CC} \leq 5.5\text{ V}$
Integral linearity error*1*2*3	ILE	—	—	±4.0	LSB	10-bit resolution	$2.4\text{ V} \leq V_{REFH0} \leq 5.5\text{ V}$
Differential linearity error*1*2*3	DLE	—	—	±2.0	LSB	10-bit resolution	$2.4\text{ V} \leq V_{REFH0} \leq 5.5\text{ V}$
Analog input voltage	VAIN	0	—	VCC	V	AN000, AN001, AN004, AN005, AN021, AN022	
		VBGR*5			V	Internal reference voltage	
		VTMP25*5			V	Temperature sensor output voltage	

Note 1. The Typ. value is an average value at TA = 25°C. The Max. value is an average value ±3σ at normal distribution.
Note 2. These values are the results of characteristic evaluation and are not checked for shipment.
Note 3. This value does not include the quantization error (±1/2 LSB).
Note 4. This value is indicated as a ratio (%FSR) to the full-scale value.
Note 5. See [section 27.6.3. Temperature Sensor/Internal Reference Voltage Characteristics](#).

Table 27.46 A/D conversion characteristics (AN000, AN004, AN005, AN021, AN022)

Conditions: $2.4\text{ V} \leq V_{REFH0} \leq V_{CC} \leq 5.5\text{ V}$, $V_{SS} = 0\text{ V}$, $T_a = -40$ to $+125^\circ\text{C}$
Reference voltage range applied to the VBGR (ADREFP[1:0] = 10b) and VREFL0 (ADREFM = 1b).
Target pins: AN000, AN004, AN005, AN021, AN022
(Reference voltage (+) = VBGR*5, Reference voltage (–) = VREFL0*6 = 0 V)

Parameter	Symbol	Min.	Typ.	Max.	Unit	Conditions
Resolution	RES	8			bit	—
Conversion time	tCONV	17	—	39	μs	—
Zero-scale error*1*2*3*4	Ezs	—	—	±0.60	%FSR	—
Integral linearity error*1*2*3	ILE	—	—	±2.0	LSB	—
Differential linearity error*1*2*3	DLE	—	—	±1.0	LSB	—
Analog input voltage	VAIN	0	—	VBGR*5	V	—

Note 1. The Typ. value is an average value at TA = 25°C. The Max. value is an average value ±3σ at normal distribution.
Note 2. These values are the results of characteristic evaluation and are not checked for shipment.
Note 3. This value does not include the quantization error (±1/2 LSB).

表 27.45 A/D 转换特性 (2/2)

条件: $2.4\text{ V} \leq V_{CC} \leq 5.5\text{ V}$, $V_{SS} = 0\text{ V}$, $T_a = -40$ 至 $+125^\circ\text{C}$
应用于 VCC 的参考电压范围 (ADREFP[1:0] = 00b) 和 VSS(ADREFM = 0b)。
目标引脚: AN000、AN001、AN004、AN005、AN021、AN022、内部参考电压和温度传感器输出电压 (参考电压(+)= VCC, 参考电压(–)= VSS)。

范围		象征	最小。	类型。	最大限度。	单元	状况	
转换时间		tCONV	2.125	—	39	μs	10 位分辨率	$3.6\text{ V} \leq V_{CC} \leq 5.5\text{ V}$
			3.1875	—	39	μs	目标引脚: AN000, AN001, AN004, AN005, AN021, AN022	$2.7\text{ V} \leq V_{CC} \leq 5.5\text{ V}$
			17	—	39	μs		$2.4\text{ V} \leq V_{CC} \leq 5.5\text{ V}$
			2.375	—	39	μs	10 位分辨率	$3.6\text{ V} \leq V_{CC} \leq 5.5\text{ V}$
			3.5625	—	39	μs	目标引脚: 内部参考	$2.7\text{ V} \leq V_{CC} \leq 5.5\text{ V}$
			17	—	39	μs	电压和温度传感器输出电压	$2.4\text{ V} \leq V_{CC} \leq 5.5\text{ V}$
零尺度错误*1*2*3*4		Ezs	—	—	±0.60	%FSR	10 位分辨率	$2.4\text{ V} \leq V_{CC} \leq 5.5\text{ V}$
满量程误差*1*2*3*4		Efs	—	—	±0.60	%FSR	10 位分辨率	$2.4\text{ V} \leq V_{CC} \leq 5.5\text{ V}$
积分线性错误*1*2*3		ILE	—	—	±4.0	LSB	10 位分辨率	$2.4\text{ V} \leq V_{REFH0} \leq 5.5\text{ V}$
微分线性错误*1*2*3		DLE	—	—	±2.0	LSB	10 位分辨率	$2.4\text{ V} \leq V_{REFH0} \leq 5.5\text{ V}$
模拟输入电压	VAIN	0	—	VCC	V	AN000、AN001、AN004、AN005、AN021、AN022		
		VBGR*5				V	内部参考电压	
		VTMP25*5				V	温度传感器输出电压	

注 1. 典型值是TA = 25°C. 处的平均值，最大值是±3σ处的平均值。
注2. 这些数值是特性评估的结果，并非出货时的检查值。
注3.此值不包括量化误差(±1/2 LSB)。
注 4. 此值以与满量程值的比率 (%FSR) 表示。
注5.参见第27.6.3节。温度传感器/内部参考电压特性。

表 27.46A/D 转换特性 (AN000、AN004、AN005、AN021、AN022)

条件: $2.4\text{ V} \leq V_{REFH0} \leq V_{CC} \leq 5.5\text{ V}$, $V_{SS} = 0\text{ V}$, $T_a = -40$ 至 $+125^\circ\text{C}$
应用于 VBGR(ADREFP[1:0] = 10b)和 VREFL0(ADREFM = 1b)的参考电压范围
目标引脚: AN000、AN004、AN005、AN021、AN022
(参考电压(+)= VBGR*5, 参考电压(–)= VREFL0*6 = 0 V))

范围		象征	最小。	类型。	最大限度。	单元	状况
解决		RES	8			少量	—
转换时间		tCONV	17	—	39	μs	—
零尺度误差*1*2*3*4		Ezs	—	—	±0.60	%FSR	—
积分线性误差*1*2*3		ILE	—	—	±2.0	LSB	—
微分线性误差*1*2*3		DLE	—	—	±1.0	LSB	—
模拟输入电压		VAIN	0	—	VBGR*5	V	—

注 1. 典型值是TA = 25°C. 处的平均值，最大值是±3σ处的平均值。
Note 2. These values are the results of characteristic evaluation and are not checked for shipment.
注3.此值不包括量化误差(±1/2 LSB)。

Note 4. This value is indicated as a ratio (%FSR) to the full-scale value.

Note 5. See [section 27.6.3. Temperature Sensor/Internal Reference Voltage Characteristics](#).

Note 6.

- Zero-scale error: Add $\pm 0.35\%$ FSR to the maximum value when reference voltage (-) = VREFL0.
- Integral linearity error: Add ± 0.5 LSB to the maximum value when reference voltage (-) = VREFL0.
- Differential linearity error: Add ± 0.2 LSB to the maximum value when reference voltage (-) = VREFL0.

27.6.3 Temperature Sensor/Internal Reference Voltage Characteristics

Table 27.47 Temperature sensor/internal reference voltage characteristics

Conditions: $1.8\text{ V} \leq \text{VCC} \leq 5.5\text{ V}$, $\text{VSS} = 0\text{ V}$, $\text{Ta} = -40\text{ to }+125^\circ\text{C}$

Parameter	Symbol	Min.	Typ.	Max.	Unit	Test conditions
Temperature sensor output voltage	V_{TMPS25}	—	1.05	—	V	at = 25°C
Internal reference voltage	V_{BGR}	1.40	1.48	1.56	V	—
Temperature coefficient	F_{VTMPS}	—	-3.3	—	mV/ $^\circ\text{C}$	—
Operation stabilization wait time	t_{AMP}	5	—	—	μs	—

27.6.4 POR Characteristics

Table 27.48 POR characteristics

Conditions: $\text{VSS} = 0\text{ V}$, $\text{Ta} = -40\text{ to }+125^\circ\text{C}$

Parameter	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Detection voltage	VPOR VPDR	1.43	1.50	1.57	V	—
Minimum pulse width*1	TPW	300	—	—	μs	—

Note 1. This width is the minimum time required for a POR reset when VCC falls below VPDR. This width is also the minimum time required for a POR reset from when VCC falls below 0.7 V to when VCC exceeds VPOR in the Software standby mode or while the main system clock is stopped through setting HOCOCR.HCSTOP bit.

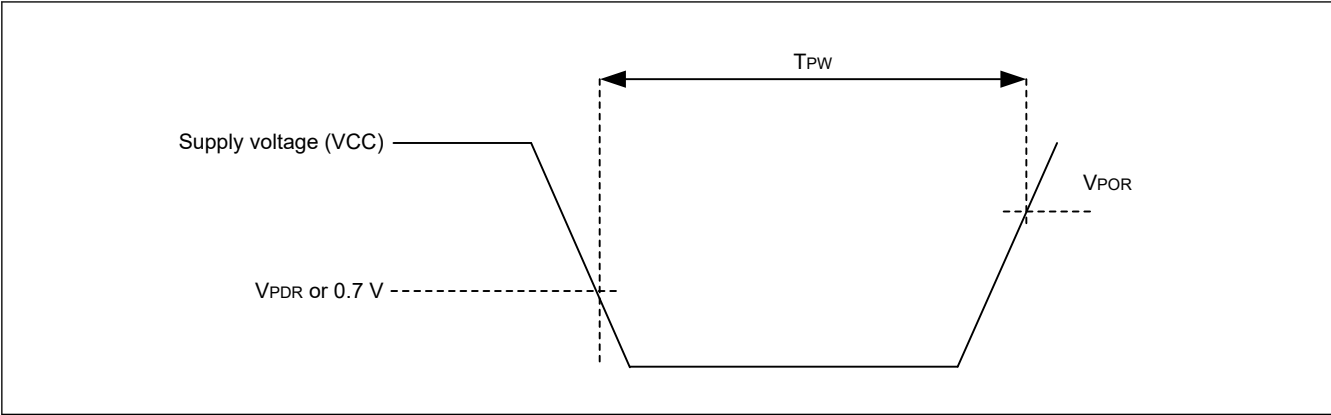


Figure 27.31 Minimum VCC pulse width

注 4. 此值以与满量程值的比率 (%FSR) 表示。

注5.参见第27.6.3节。温度传感器/内部参考电压特性。

注6.●零刻度误差：当参考电压(-) = VREFL0时，将 $\pm 0.35\%$ FSR 加到最大值。

- 积分线性误差：当参考电压(-) = VREFL0 时，将 ± 0.5 LSB 加到最大值。
- 差分线性误差：当参考电压(-) = VREFL0 时，将 ± 0.2 LSB 加到最大值。

27.6.3 温度传感器/内部参考电压特性

表 27.47 温度传感器/内部参考电压特性

条件： $1.8\text{ V} \leq \text{VCC} \leq 5.5\text{ V}$, $\text{VSS} = 0\text{ V}$, $\text{Ta} = -40\text{ 至}+125^\circ\text{C}$

范围	象征	最小。	类型。	最大限度。	单元	测试条件
温度传感器输出电压	V_{TMPS25}	—	1.05	—	V	at = 25°C
内部参考电压	V_{BGR}	1.40	1.48	1.56	V	—
温度系数	F_{VTMPS}	—	-3.3	—	mV/ $^\circ\text{C}$	—
运行稳定等待时间	秀	5	—	—	μs	—

27.6.4 POR特性

表 27.48 POR 特征

条件： $\text{VSS} = 0\text{ V}$, $\text{Ta} = -40\text{至}+125^\circ\text{C}$

范围	象征	最小。	类型。	最大限度。	单元	测试条件
检测电压	VPOR VPDR	1.43	1.50	1.57	V	—
最小脉冲宽度*1	TPW	300	—	—	μs	—

注1： 此宽度为VCC低于VPDR时POR复位所需的最小时间。此宽度也是在软件待机模式下， 或通过设置HOCOCR.HCSTOP位停止主系统时钟时， 从VCC低于0.7 V到VCC超过VPOR之间POR复位所需的最小时间。

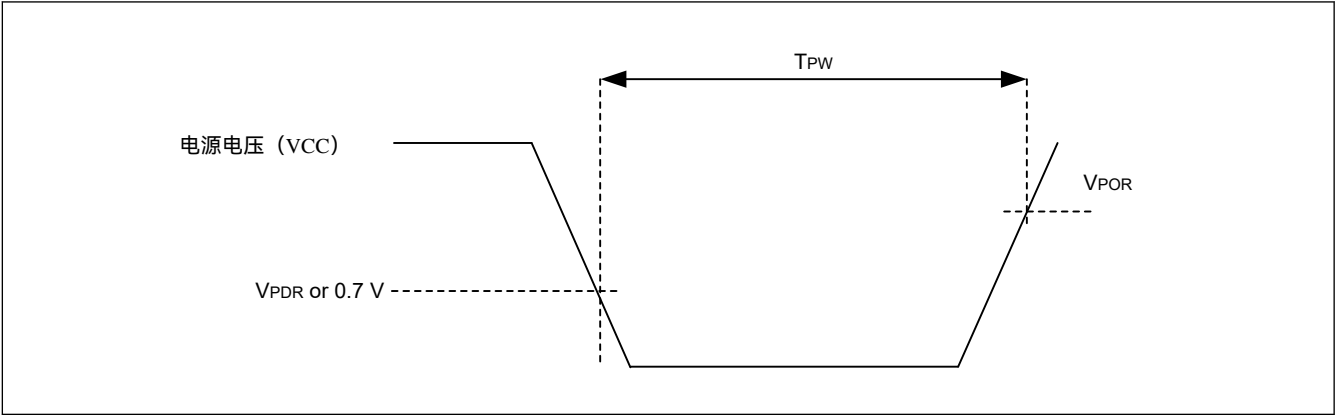


图 27.31 最小 VCC 脉冲宽度

27.6.5 LVD Characteristics

Table 27.49 LVD0 characteristics

Conditions: VPDR ≤ VCC ≤ 5.5 V, VSS = 0 V, Ta = -40 to +125°C

Parameter		Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Detection voltage	Supply voltage level	V _{det0_0}	3.84	3.96	4.08	V	The power supply voltage is rising.
			3.76	3.88	4.00	V	The power supply voltage is falling.
		V _{det0_1}	2.88	2.97	3.06	V	The power supply voltage is rising.
			2.82	2.91	3.00	V	The power supply voltage is falling.
		V _{det0_2}	2.59	2.67	2.75	V	The power supply voltage is rising.
			2.54	2.62	2.70	V	The power supply voltage is falling.
		V _{det0_3}	2.31	2.38	2.45	V	The power supply voltage is rising.
			2.26	2.33	2.40	V	The power supply voltage is falling.
		V _{det0_4}	1.84	1.90	1.95	V	The power supply voltage is rising.
			1.80	1.86	1.91	V	The power supply voltage is falling.
		V _{det0_5}	1.64	1.69	1.74	V	The power supply voltage is rising.
			1.60	1.65	1.70	V	The power supply voltage is falling.
Minimum pulse width		t _{LW0}	500	—	—	μs	—
Detection delay time		t _{det0}	—	—	500	μs	—

Table 27.50 LVD1 characteristics (1 of 2)

Conditions: VPDR ≤ VCC ≤ 5.5 V, VSS = 0 V, Ta = -40 to +125°C

Parameter		Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Detection voltage	Supply voltage level	V _{det1_0}	4.08	4.16	4.24	V	The power supply voltage is rising.
			4.00	4.08	4.16	V	The power supply voltage is falling.
		V _{det1_1}	3.88	3.96	4.04	V	The power supply voltage is rising.
			3.80	3.88	3.96	V	The power supply voltage is falling.
		V _{det1_2}	3.68	3.75	3.82	V	The power supply voltage is rising.
			3.60	3.67	3.74	V	The power supply voltage is falling.
		V _{det1_3}	3.48	3.55	3.62	V	The power supply voltage is rising.
			3.40	3.47	3.54	V	The power supply voltage is falling.
		V _{det1_4}	3.28	3.35	3.42	V	The power supply voltage is rising.
			3.20	3.27	3.34	V	The power supply voltage is falling.
		V _{det1_5}	3.07	3.13	3.19	V	The power supply voltage is rising.
			3.00	3.06	3.12	V	The power supply voltage is falling.
		V _{det1_6}	2.91	2.97	3.03	V	The power supply voltage is rising.
			2.85	2.91	2.97	V	The power supply voltage is falling.
		V _{det1_7}	2.76	2.82	2.87	V	The power supply voltage is rising.
			2.70	2.76	2.81	V	The power supply voltage is falling.
		V _{det1_8}	2.61	2.66	2.71	V	The power supply voltage is rising.
			2.55	2.60	2.65	V	The power supply voltage is falling.
		V _{det1_9}	2.45	2.50	2.55	V	The power supply voltage is rising.
			2.40	2.45	2.50	V	The power supply voltage is falling.
		V _{det1_A}	2.35	2.40	2.45	V	The power supply voltage is rising.
			2.30	2.35	2.40	V	The power supply voltage is falling.

27.6.5LVD特性

表 27.49LVD0 特征

条件：VPDR≤VCC≤ 5.5 V、VSS = 0 V Ta= -40至+125°C

范围		象征	最小。	类型。	最大限	单元	测试条件
检测电压	供电电压等级	V _{det0_0}	3.84	3.96	4.08	V	电源电压正在上升。
			3.76	3.88	4.00	V	电源电压正在下降。
		V _{det0_1}	2.88	2.97	3.06	V	电源电压正在上升。
			2.82	2.91	3.00	V	电源电压正在下降。
		V _{det0_2}	2.59	2.67	2.75	V	电源电压正在上升。
			2.54	2.62	2.70	V	电源电压正在下降。
		V _{det0_3}	2.31	2.38	2.45	V	电源电压正在上升。
			2.26	2.33	2.40	V	电源电压正在下降。
		V _{det0_4}	1.84	1.90	1.95	V	电源电压正在上升。
			1.80	1.86	1.91	V	电源电压正在下降。
		V _{det0_5}	1.64	1.69	1.74	V	电源电压正在上升。
			1.60	1.65	1.70	V	电源电压正在下降。
最小脉冲宽度		t _{LW0}	500	—	—	μs	—
检测延迟时间		t _{det0}	—	—	500	μs	—

表 27.50LVD1 特征（1/2）

条件：VPDR≤VCC≤ 5.5 V、VSS= 0 V, Ta= -40至+125°C

范围		象征	最小。	类型。	最大限	单元	测试条件
检测电压	供电电压等级	V _{det1_0}	4.08	4.16	4.24	V	电源电压正在上升。
			4.00	4.08	4.16	V	电源电压正在下降。
		V _{det1_1}	3.88	3.96	4.04	V	电源电压正在上升。
			3.80	3.88	3.96	V	电源电压正在下降。
		V _{det1_2}	3.68	3.75	3.82	V	电源电压正在上升。
			3.60	3.67	3.74	V	电源电压正在下降。
		V _{det1_3}	3.48	3.55	3.62	V	电源电压正在上升。
			3.40	3.47	3.54	V	电源电压正在下降。
		V _{det1_4}	3.28	3.35	3.42	V	电源电压正在上升。
			3.20	3.27	3.34	V	电源电压正在下降。
		V _{det1_5}	3.07	3.13	3.19	V	电源电压正在上升。
			3.00	3.06	3.12	V	电源电压正在下降。
		V _{det1_6}	2.91	2.97	3.03	V	电源电压正在上升。
			2.85	2.91	2.97	V	电源电压正在下降。
		V _{det1_7}	2.76	2.82	2.87	V	电源电压正在上升。
			2.70	2.76	2.81	V	电源电压正在下降。
		V _{det1_8}	2.61	2.66	2.71	V	电源电压正在上升。
			2.55	2.60	2.65	V	电源电压正在下降。
		V _{det1_9}	2.45	2.50	2.55	V	电源电压正在上升。
			2.40	2.45	2.50	V	电源电压正在下降。
		V _{det1_A}	2.35	2.40	2.45	V	电源电压正在上升。
			2.30	2.35	2.40	V	电源电压正在下降。

Table 27.50 LVD1 characteristics (2 of 2)

Conditions: VPDR ≤ VCC ≤ 5.5 V, VSS = 0 V, Ta = -40 to +125°C

Parameter		Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Detection voltage	Supply voltage level	V _{det1_B}	2.25	2.30	2.34	V	The power supply voltage is rising.
			2.20	2.25	2.29	V	The power supply voltage is falling.
		V _{det1_C}	2.15	2.20	2.24	V	The power supply voltage is rising.
			2.10	2.15	2.19	V	The power supply voltage is falling.
		V _{det1_D}	2.05	2.09	2.13	V	The power supply voltage is rising.
			2.00	2.04	2.08	V	The power supply voltage is falling.
		V _{det1_E}	1.94	1.98	2.02	V	The power supply voltage is rising.
			1.90	1.94	1.98	V	The power supply voltage is falling.
		V _{det1_F}	1.84	1.88	1.91	V	The power supply voltage is rising.
			1.80	1.84	1.87	V	The power supply voltage is falling.
		V _{det1_10}	1.74	1.78	1.81	V	The power supply voltage is rising.
			1.70	1.74	1.77	V	The power supply voltage is falling.
		V _{det1_11}	1.64	1.67	1.70	V	The power supply voltage is rising.
			1.60	1.63	1.66	V	The power supply voltage is falling.
Minimum pulse width		t _{LW1}	500	—	—	μs	—
Detection delay time		t _{det1}	—	—	500	μs	—
LVD1 detection voltage stabilization time (after changing the LVD1 detection voltage)		t _{d(E-A)}	—	—	1500	μs	—

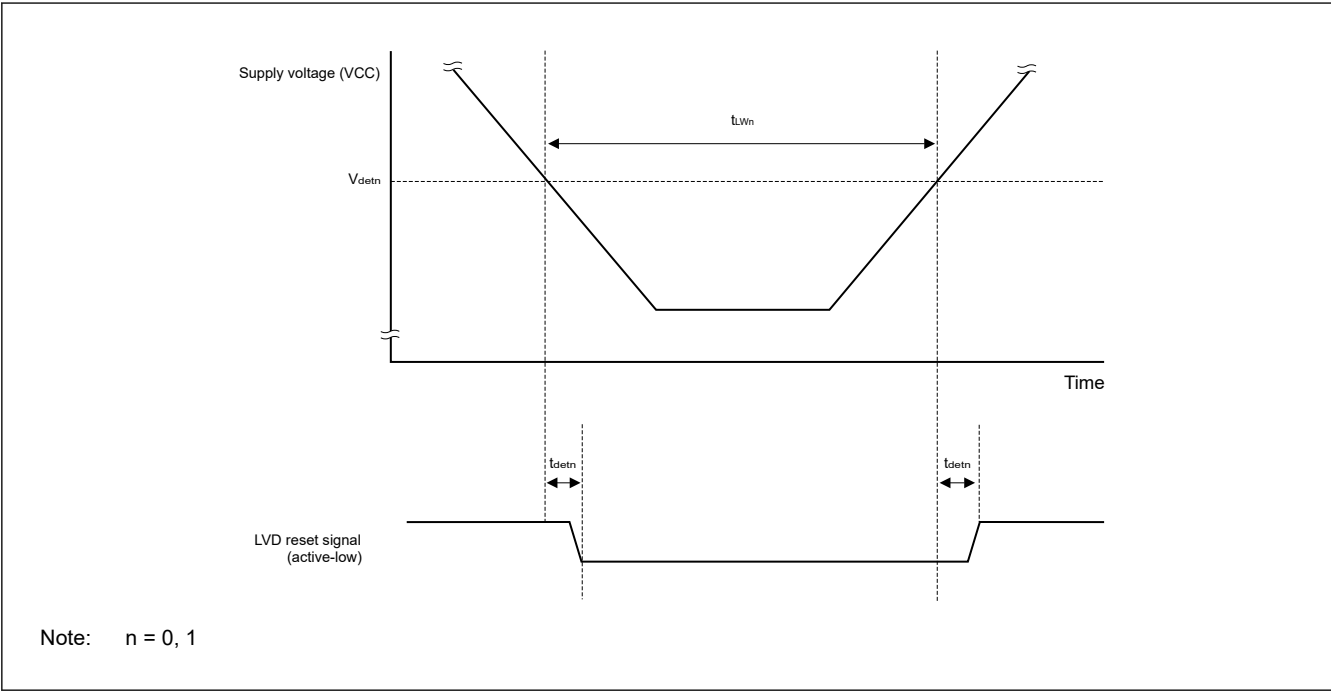


Figure 27.32 Voltage detection circuit timing

表 27.50 LVD1 特征 (2/2)

条件: VPDR ≤ VCC ≤ 5.5 V, VSS = 0 V, Ta = -40至+125°C

范围			象征	最小。	类型。	最大限	单元	测试条件
检测电压	供电电压等级	V _{det1_B}	2.25	2.30	2.34	V	电源电压正在上升。	
			2.20	2.25	2.29	V	电源电压正在下降。	
		V _{det1_C}	2.15	2.20	2.24	V	电源电压正在上升。	
			2.10	2.15	2.19	V	电源电压正在下降。	
		V _{det1_D}	2.05	2.09	2.13	V	电源电压正在上升。	
			2.00	2.04	2.08	V	电源电压正在下降。	
		V _{det1_E}	1.94	1.98	2.02	V	电源电压正在上升。	
			1.90	1.94	1.98	V	电源电压正在下降。	
		V _{det1_F}	1.84	1.88	1.91	V	电源电压正在上升。	
			1.80	1.84	1.87	V	电源电压正在下降。	
		V _{det1_10}	1.74	1.78	1.81	V	电源电压正在上升。	
			1.70	1.74	1.77	V	电源电压正在下降。	
		V _{det1_11}	1.64	1.67	1.70	V	电源电压正在上升。	
			1.60	1.63	1.66	V	电源电压正在下降。	
最小脉冲宽度		t _{LW1}	500	—	—	μs	—	
检测延迟时间		t _{det1}	—	—	500	μs	—	
LVD1检测电压稳定时间（改变LVD1检测电压后）		t _{d(E-A)}	—	—	1500	μs	—	

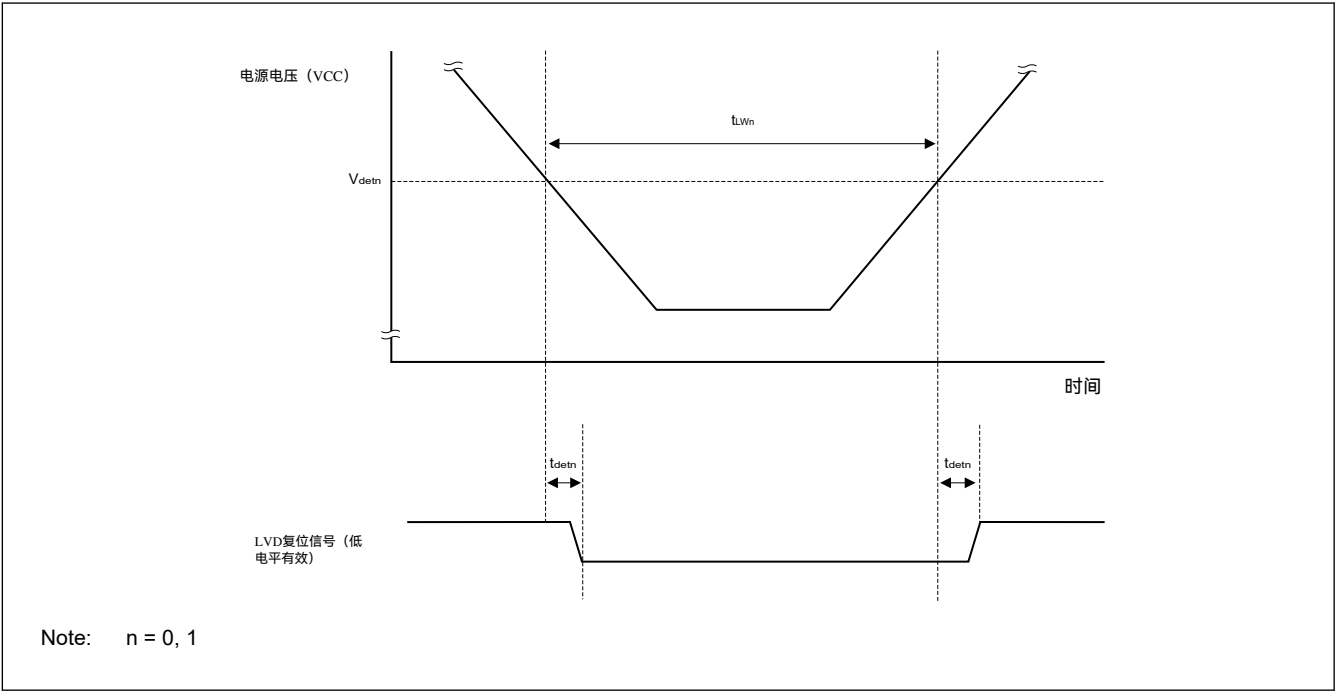


图 27.32 电压检测电路时序

27.6.6 Power Supply Voltage Rising Slope Characteristics

Table 27.51 Power supply voltage rising slope characteristics

Conditions: VSS = 0 V, Ta = -40 to +125°C

Parameter	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Power supply voltage rising slope	S _{VCC}	—	—	54	V/ms	—

Note: Make sure to keep the internal reset state by the LVD0 circuit or an external reset until VCC reaches the operating voltage range shown in AC characteristics.

27.7 RAM Data Retention Characteristics

Table 27.52 RAM data retention characteristics

Conditions: VSS = 0 V, Ta = -40 to +125°C

Parameter	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Data retention supply voltage	V _{CCDR}	1.43*1	—	5.5	V	—

Note 1. This voltage depends on the POR detection voltage. When the voltage drops, the data in RAM are retained until a POR is applied, but are not retained following a POR.

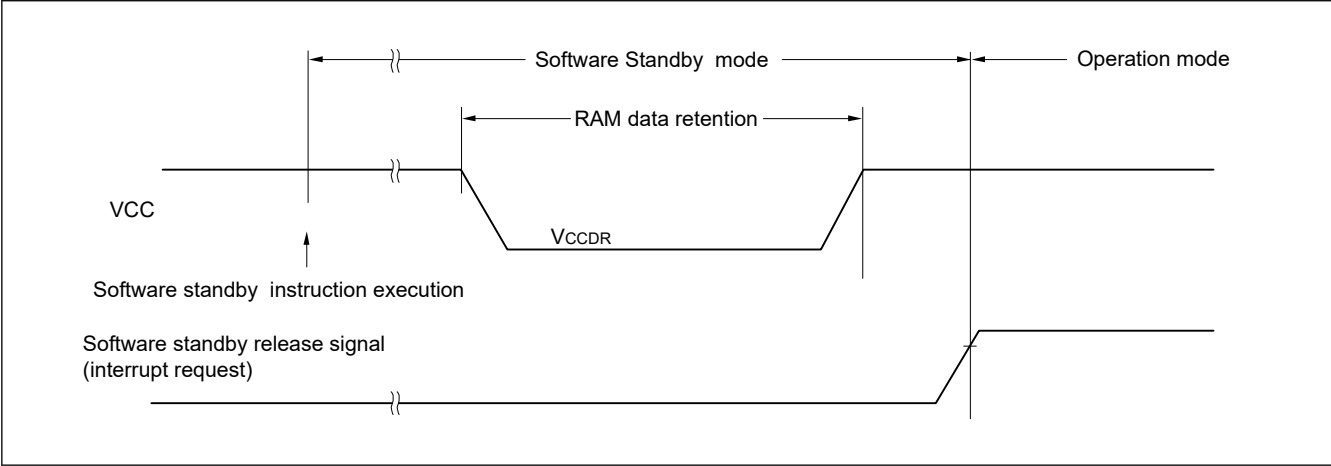


Figure 27.33 RAM data retention

27.8 Flash Memory Programming Characteristics

Table 27.53 Flash memory programming characteristics

Conditions: 1.8 V ≤ VCC ≤ 5.5 V, VSS = 0 V, Ta = -40 to +125°C

Parameter	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
CPU/peripheral hardware clock frequency	I _{CLK}	1	—	32	MHz	—
Number of code flash rewrites*1 *2 *3	Cerwr	10000	—	—	Times	Retained for 10 years Ta = 85°C
		1000	—	—		Retained for 20 years Ta = 85°C

Note 1. 1 erase + 1 write after the erase is regarded as 1 rewrite. The retaining years are until next rewrite after the rewrite.
Note 2. The listed numbers of times apply when using the flash memory programmer and self-programming.
Note 3. These are the characteristics of the flash memory and the results obtained from reliability testing by Renesas Electronics Corporation.

Table 27.54 Code flash memory characteristics (1 of 2)

Conditions: 1.8 V ≤ VCC ≤ 5.5 V, VSS = 0 V, Ta = -40 to +125°C

Parameter	Symbol		ICLK = 1 MHz			ICLK = 2 MHz, 3 MHz			4 MHz ≤ ICLK < 8 MHz			8 MHz ≤ ICLK < 32 MHz			ICLK = 32 MHz			Unit
			Min.	Typ.	Max.	Min.	Typ.	Max.	Min.	Typ.	Max.	Min.	Typ.	Max.	Min.	Typ.	Max.	
Programming time	4 bytes	t _{p4}	—	74.7	656.5	—	51.0	464.6	—	41.7	384.8	—	37.1	346.2	—	34.2	321.9	μs

27.6.6 电源电压上升斜率特性

表 27.51 电源电压上升斜率特性

条件: VSS = 0 V, Ta = -40 至 +125°C

范围	象征	最小。	类型。	最大限度。	单元	测试条件
电源电压上升斜率	S _{VCC}	—	—	54	伏/毫秒	—

注意: 务必通过 LVD0 电路或外部复位保持内部复位状态, 直到 VCC 达到交流特性中所示的工作电压范围。

27.7 RAM 数据保持特性

表 27.52 RAM 数据保持特性

条件: VSS = 0 V, Ta = -40 至 +125°C

范围	象征	最小。	类型。	最大限度。	单元	测试条件
数据保持供电电压	V _{CCDR}	1.43*1	—	5.5	V	—

注1: 此电压取决于POR检测电压。当电压下降时, RAM中的数据会保留, 直到施加POR操作, 但POR操作后数据不会被保留。

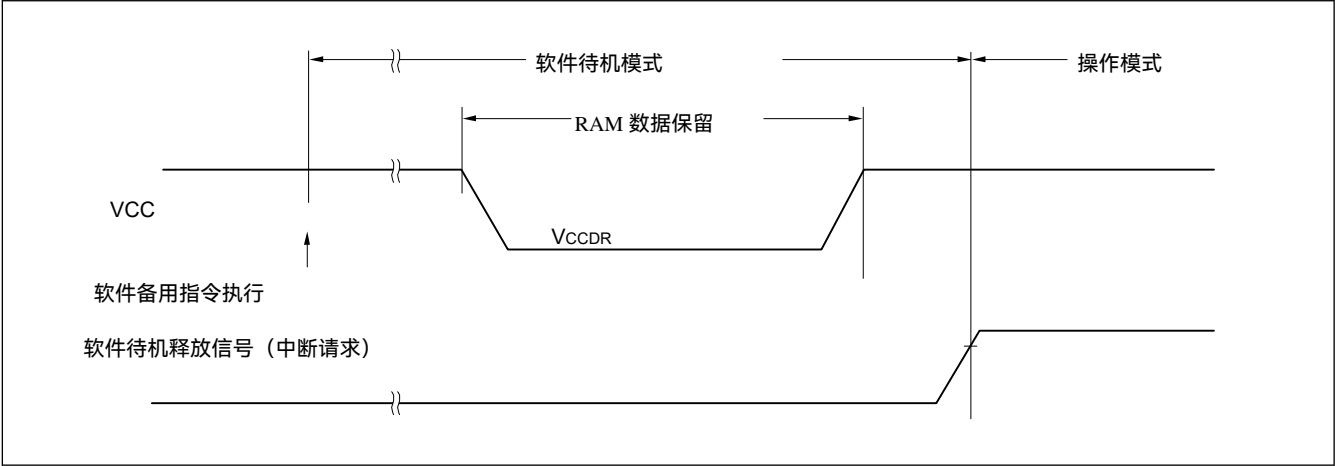


图 27.33 RAM 数据保持

27.8 闪存编程特性

表 27.53 闪存编程特性

条件: 1.8 V ≤ VCC ≤ 5.5 V, VSS = 0 V, Ta = -40 至 +125°C

范围	象征	最小。	类型。	最大限度。	单元	测试条件
CPU/外围硬件时钟频率	I _{CLK}	1	—	32	MHz	—
代码闪存重写次数*1 *2 *3	卡弗	10000	—	—	《时代周刊》	保留10年 Ta = 85°C
		1000	—	—		保留了20年 Ta = 85°C

注1. 擦除后写入操作视为1次重写。保留期限至下次重写为止。
注2. 所列次数适用于使用闪存编程器和自行编程的情况。
注3. 这些是闪存的特性以及瑞萨电子株式会社通过可靠性测试获得的结果。

表 27.54 代码闪存特性 (1/2)

条件: 1.8V ≤ VCC ≤ 5.5V, VSS = 0V, Ta = -40 至 +125°C

范围	象征		ICLK = 1 MHz			ICLK = 2 MHz, 3 MHz			4 MHz ≤ ICLK < 8 MHz			8 MHz ≤ ICLK < 32 MHz			ICLK = 32 MHz			单元
			最小。	类型。	最大限	最小。	类型。	最大限	最小。	类型。	最大限	最小。	类型。	最大限	最小。	类型。	最大限	
编程时间	4 字节	t _{p4}	—	74.7	656.5	—	51.0	464.6	—	41.7	384.8	—	37.1	346.2	—	34.2	321.9	μs

Table 27.54 Code flash memory characteristics (2 of 2)

Conditions: 1.8 V ≤ VCC ≤ 5.5 V, VSS = 0 V, Ta = -40 to +125°C

Parameter		Symbol	ICLK = 1 MHz			ICLK = 2 MHz, 3 MHz			4 MHz ≤ ICLK < 8 MHz			8 MHz ≤ ICLK < 32 MHz			ICLK = 32 MHz			Unit
			Min.	Typ.	Max.	Min.	Typ.	Max.	Min.	Typ.	Max.	Min.	Typ.	Max.	Min.	Typ.	Max.	
Erasure time	2 Kbytes	tE2K	—	10.4	312.2	—	7.7	258.5	—	6.4	231.8	—	5.8	218.4	—	5.6	214.4	ms
Blank checking time	4 bytes	tBC4	—	—	38.4	—	—	19.2	—	—	13.1	—	—	10.2	—	—	8.3	μs
	2 Kbytes	tBC2K	—	—	2618.9	—	—	1309.5	—	—	658.3	—	—	332.8	—	—	234.1	μs
Time taken to forcibly stop the erasure		tSED	—	—	18.0	—	—	14.0	—	—	12.0	—	—	11.0	—	—	10.3	μs
Security setting time		tAWSSAS	—	18.0	525.5	—	14.3	468.7	—	12.5	440.7	—	11.6	426.7	—	11.3	422.3	ms
Time until programming starts following cancellation of the Software standby instruction		—	20	—	—	20	—	—	20	—	—	20	—	—	20	—	—	μs
Flash memory mode transition wait time 1		tDIS	2	—	—	2	—	—	2	—	—	2	—	—	2	—	—	μs
Flash memory mode transition wait time 2		tMS	15	—	—	15	—	—	15	—	—	15	—	—	15	—	—	μs

Note: The listed values do not include the time until the operations of the flash memory start following execution of an instruction by software.

27.9 Serial Wire Debug (SWD)

Table 27.55 SWD characteristics (1)

Conditions: VCC = 2.4 to 5.5 V

Parameter	Symbol	Min.	Typ.	Max.	Unit	Test conditions
SWCLK clock cycle time	tSWCKcyc	80	—	—	ns	Figure 27.34
SWCLK clock high pulse width	tSWCKH	35	—	—	ns	
SWCLK clock low pulse width	tSECKL	35	—	—	ns	
SWCLK clock rise time	tSWCKr	—	—	5	ns	
SWCLK clock fall time	tSWCKf	—	—	5	ns	
SWDIO setup time	tSWDS	16	—	—	ns	Figure 27.35
SWDIO hold time	tSWDH	16	—	—	ns	
SWDIO data delay time	tSWDD	2	—	70	ns	

Table 27.56 SWD characteristics (2)

Conditions: VCC = 1.6 to 2.4 V

Parameter	Symbol	Min.	Typ.	Max.	Unit	Test conditions
SWCLK clock cycle time	tSWCKcyc	250	—	—	ns	Figure 27.34
SWCLK clock high pulse width	tSWCKH	120	—	—	ns	
SWCLK clock low pulse width	tSECKL	120	—	—	ns	
SWCLK clock rise time	tSWCKr	—	—	5	ns	
SWCLK clock fall time	tSWCKf	—	—	5	ns	
SWDIO setup time	tSWDS	50	—	—	ns	Figure 27.35
SWDIO hold time	tSWDH	50	—	—	ns	
SWDIO data delay time	tSWDD	2	—	170	ns	

表 27.54 代码闪存特性 (2/2)

条件: 1.8V≤ VCC≤ 5.5 V、VSS= 0 V、Ta= -40 至+125°C

范围		象征	ICLK = 1 MHz			ICLK = 2 MHz, 3 MHz			4 MHz ≤ ICLK < 8 MHz			8 MHz ≤ ICLK < 32 MHz			ICLK = 32 MHz			单元
			最小。	类型。	最大限	最小。	类型。	最大限	最小。	类型。	最大限	最小。	类型。	最大限	最小。	类型。	最大限度。	
抹除时间	2 千字节	tE2K	—	10.4	312.2	—	7.7	258.5	—	6.4	231.8	—	5.8	218.4	—	5.6	214.4	ms
空白支票时间	4 字节	tBC4	—	—	38.4	—	—	19.2	—	—	13.1	—	—	10.2	—	—	8.3	μs
	2 千字节	tBC2K	—	—	2618.9	—	—	1309.5	—	—	658.3	—	—	332.8	—	—	234.1	μs
强制阻止抹除所花费的时间		tSED	—	—	18.0	—	—	14.0	—	—	12.0	—	—	11.0	—	—	10.3	μs
安全设置时间		tAWSSAS	—	18.0	525.5	—	14.3	468.7	—	12.5	440.7	—	11.6	426.7	—	11.3	422.3	ms
取消软件待机指令后，编程开始所需的时间		—	20	—	—	20	—	—	20	—	—	20	—	—	20	—	—	μs
闪存模式转换等待时间 1		tDIS	2	—	—	2	—	—	2	—	—	2	—	—	2	—	—	μs
闪存模式转换等待时间 2		tMS	15	—	—	15	—	—	15	—	—	15	—	—	15	—	—	μs

注意：所列数值不包括软件执行指令后闪存开始运行所需的时间。

27.9串行线调试 (SWD)

表 27.55SWD 特征 (1)

条件: VCC = 2.4至5.5 V

范围	象征	最小。	类型。	最大限度。	单元	测试条件
SWCLK 时钟周期时间	tSWCKcyc	80	—	—	ns	图 27.34
SWCLK时钟高脉冲宽度	tSWCKH	35	—	—	ns	
SWCLK时钟低脉冲宽度	tSECKL	35	—	—	ns	
SWCLK 时钟上升时间	tSWCKr	—	—	5	ns	
SWCLK 时钟下降时间	tSWCKf	—	—	5	ns	
SWDIO 设置时间	tSWDS	16	—	—	ns	图 27.35
SWDIO保持时间	tSWDH	16	—	—	ns	
SWDIO 数据延迟时间	tSWDD	2	—	70	ns	

表 27.56SWD 特征 (2)

条件: VCC = 1.6至2.4 V

范围	象征	最小。	类型。	最大限度。	单元	测试条件
SWCLK 时钟周期时间	tSWCKcyc	250	—	—	ns	图 27.34
SWCLK时钟高脉冲宽度	tSWCKH	120	—	—	ns	
SWCLK时钟低脉冲宽度	tSECKL	120	—	—	ns	
SWCLK 时钟上升时间	tSWCKr	—	—	5	ns	
SWCLK 时钟下降时间	tSWCKf	—	—	5	ns	
SWDIO 设置时间	tSWDS	50	—	—	ns	图 27.35
SWDIO保持时间	tSWDH	50	—	—	ns	
SWDIO 数据延迟时间	tSWDD	2	—	170	ns	

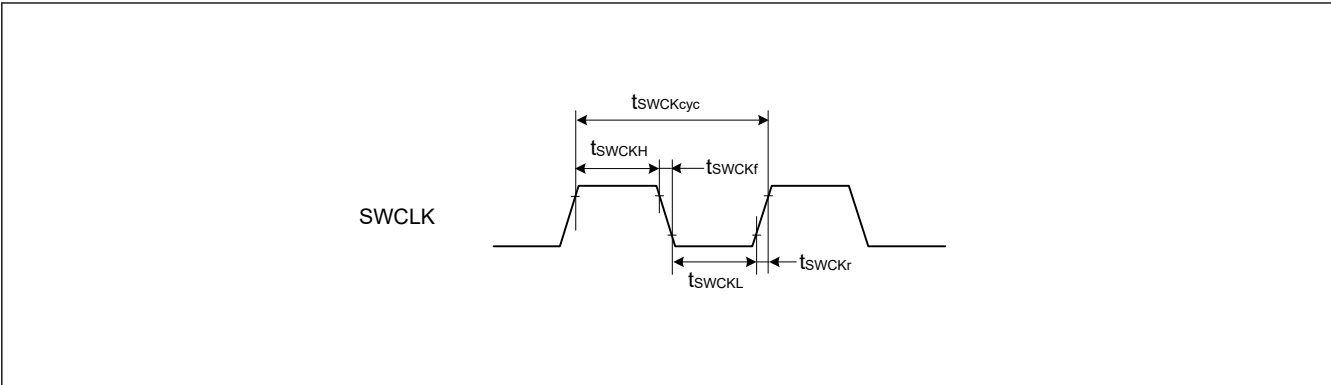


Figure 27.34 SWD SWCLK timing

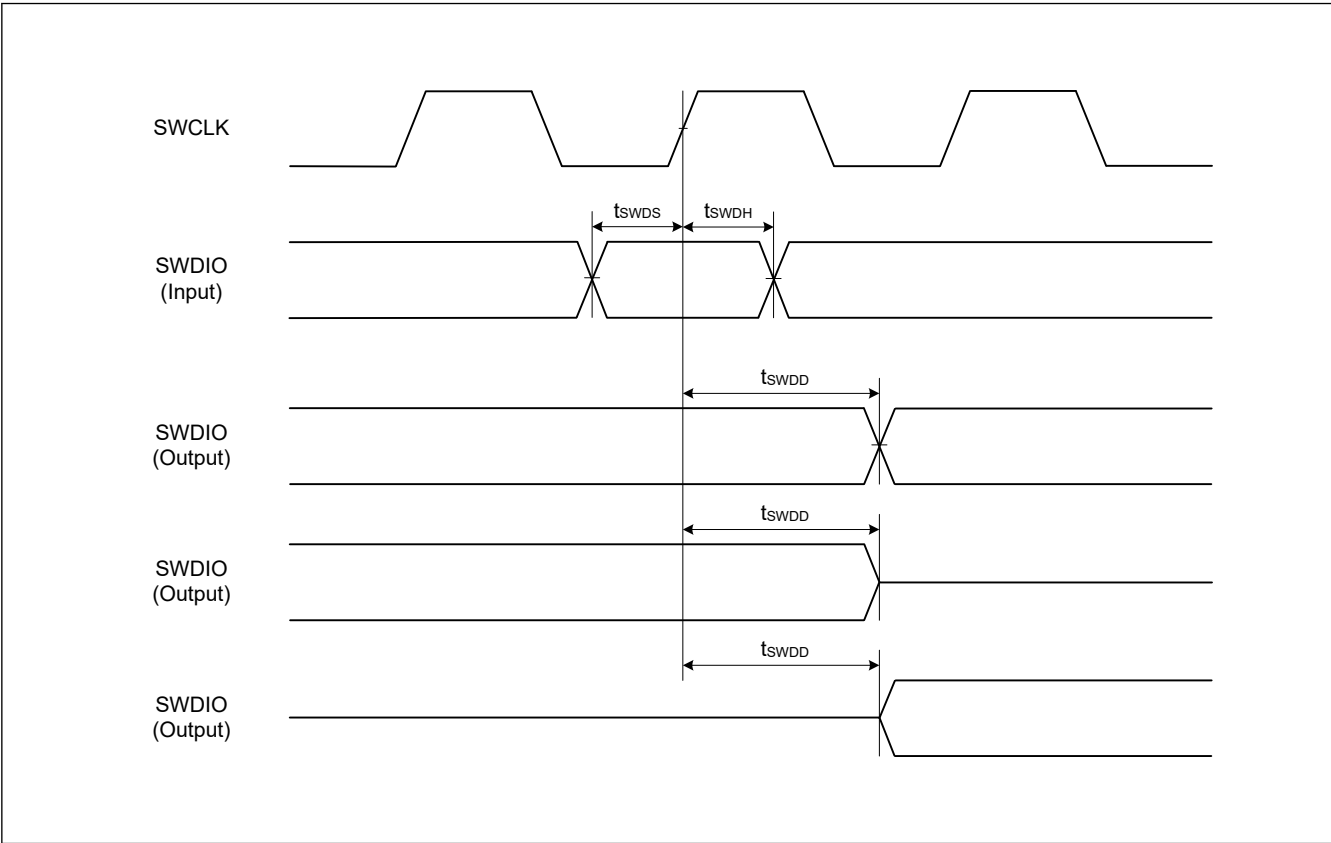


Figure 27.35 SWD input/output timing

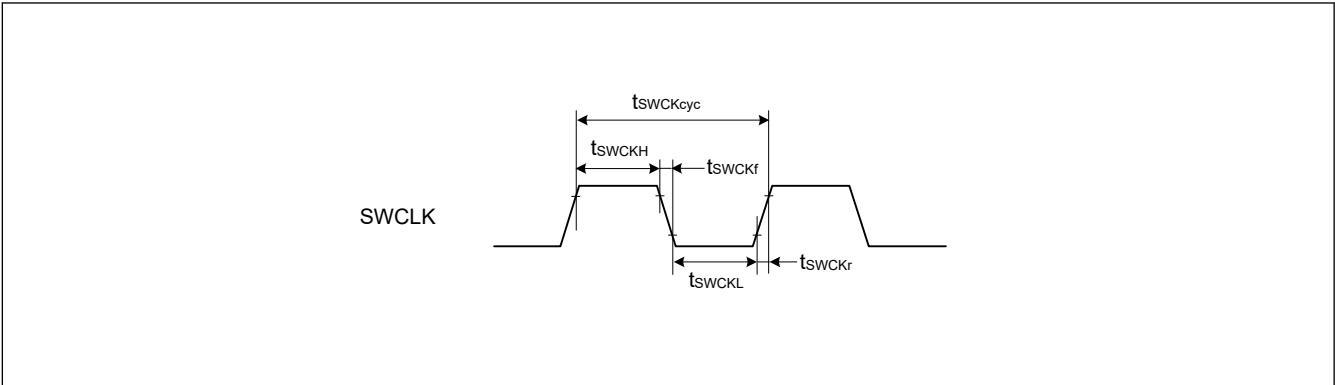


图 27.34SWD SWCLK 时序

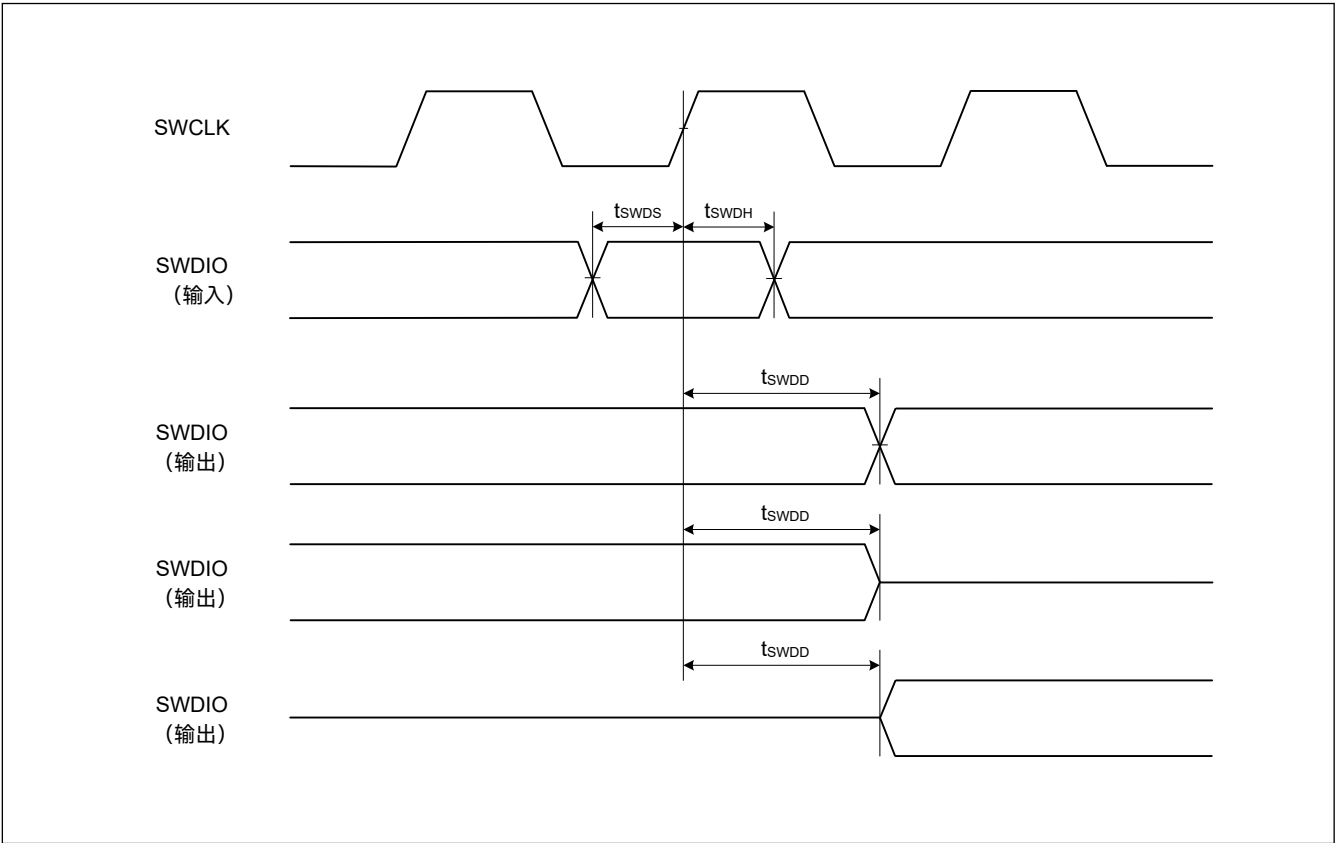


图 27.35SWD 输入/输出时序

Appendix 1. Port States in each Processing Mode

Table A1.1 Port states in each processing mode (1 of 2)

Port name	Reset	Software Standby Mode
P010/VREFH0/AN000	Hi-Z	Keep-O
P011/VREFL0/AN001	Hi-Z	Keep-O
P012/AN004	Hi-Z	Keep-O
P013/AN005	Hi-Z	Keep-O
P100/AN022/IRQ2_A/TI04_A/TO04_A/TI01_B/TO01_B/RXD0_A/SI00_A/SDA00_A/SCLA0_D	Hi-Z	[IRQ2_A selected] IRQ2_A input [SCLA0_D selected] SCLA0_D inout [Other than the above] Keep-O
P101/AN021/IRQ3_A/TI07_A/TO07_A/TI00_C/TXD0_A/SO00_A/SDAA0_D	Hi-Z	[IRQ3_A selected] IRQ3_A input [SDAA0_D selected] SDAA0_D inout [Other than the above] Keep-O
P102/IRQ4_A/TI06_A/TO06_A/TO00_C/PCLBUZ0_B/SCK00_A/SCL00_A	Hi-Z	[IRQ4_A selected] IRQ4_A input [PCLBUZ0_B selected] PCLBUZ0_B output [Other than the above] Keep-O
P108/SWDIO/TI03_B/TO03_B/PCLBUZ0_D/SCLA0_E/SCK00_C	Pull-up	[SCLA0_E selected] SCLA0_E inout [PCLBUZ0_D selected] PCLBUZ0_D output Keep-O
P109/IRQ4_B/TI02_A/TO02_A/TXD0_B/SO00_B/SDAA0_C	Hi-Z	[IRQ4_B selected] IRQ4_B input [SDAA0_C selected] SDAA0_C inout [Other than the above] Keep-O
P110/IRQ3_B/TI01_A/TO01_A/RXD0_B/SI00_B/SDA00_B/SCLA0_C	Hi-Z	[IRQ3_B selected] IRQ3_B input [SCLA0_C selected] SCLA0_C inout [Other than the above] Keep-O
P112/IRQ2_B/TI03_A/TO03_A/SCK00_B/SCL00_B/SSI00_C	Hi-Z	[IRQ2_B selected] IRQ2_B input [Other than the above] Keep-O
P200/NMI/IRQ0_A/TI06_B/SSI00_D/RXD1_B	Hi-Z	[NMI/IRQ0_A selected] NMI/IRQ0_A input [Other than the above] Hi-Z
P201/IRQ5_B/TI05_B/TO05_B/PCLBUZ0_A/SSI00_B/SCK11_B/SCL11_B	Hi-Z	[IRQ5_B selected] IRQ5_B input [PCLBUZ0_A selected] PCLBUZ0_A output [Other than the above] Keep-O

附录 1. 各处理模式下的端口状态

表 A1.1 各处理模式下的端口状态 (1/2)

端口名称	重置	软件待机模式
P010/VREFH0/AN000	高阻抗	Keep-O
P011/VREFL0/AN001	高阻抗	Keep-O
P012/AN004	高阻抗	Keep-O
P013/AN005	高阻抗	Keep-O
P100/AN022/IRQ2_A/TI04_A/TO04_A/TI01_B/TO01_B/RXD0_A/SI00_A/SDA00_A/SCLA0_D	高阻抗	[IRQ2_A 已选择] IRQ2_A 输入 [SCLA0_D 已选择] SCLA0_D 输入/输出 [除上述内容外] Keep-O
P101/AN021/IRQ3_A/TI07_A/TO07_A/TI00_C/TXD0_A/SO00_A/SDAA0_D	高阻抗	[IRQ3_A 已选择] IRQ3_A 输入 [SDAA0_D 已选择] SDAA0_D 输入/输出 [除上述内容外] Keep-O
P102/IRQ4_A/TI06_A/TO06_A/TO00_C/PCLBUZ0_B/SCK00_A/SCL00_A	高阻抗	[IRQ4_A 已选择] IRQ4_A 输入 [PCLBUZ0_B 已选择] PCLBUZ0_B 输出 [除上述内容外] Keep-O
P108/SWDIO/TI03_B/TO03_B/PCLBUZ0_D/SCLA0_E/SCK00_C	引体向上	[SCLA0_E 已选择] SCLA0_E 输入/输出 [PCLBUZ0_D 已选择] PCLBUZ0_D 输出 Keep-O
P109/IRQ4_B/TI02_A/TO02_A/TXD0_B/SO00_B/SDAA0_C	高阻抗	[IRQ4_B 已选择] IRQ4_B 输入 [SDAA0_C 已选择] SDAA0_C 输入/输出 [除上述内容外] Keep-O
P110/IRQ3_B/TI01_A/TO01_A/RXD0_B/SI00_B/SDA00_B/SCLA0_C	高阻抗	[IRQ3_B 已选择] IRQ3_B 输入 [SCLA0_C 已选择] SCLA0_C 输入输出 [除上述内容外] Keep-O
P112/IRQ2_B/TI03_A/TO03_A/SCK00_B/SCL00_B/SSI00_C	高阻抗	[IRQ2_B 已选择] IRQ2_B 输入 [除上述内容外] Keep-O
P200/NMI/IRQ0_A/TI06_B/SSI00_D/RXD1_B	高阻抗	[已选择 NMI/IRQ0_A] NMI/IRQ0_A 输入 [除上述内容外] 高阻抗
P201/IRQ5_B/TI05_B/TO05_B/PCLBUZ0_A/SSI00_B/SCK11_B/SCL11_B	高阻抗	[IRQ5_B 已选择] IRQ5_B 输入 [PCLBUZ0_A 已选择] PCLBUZ0_A 输出 [除上述内容外] Keep-O

Table A1.1 Port states in each processing mode (2 of 2)

Port name	Reset	Software Standby Mode
P206/RES/IRQ1_C	Pull-up	[IRQ1_C selected] IRQ1_C input [RES (OFS1.PORTSELB = 1) selected] RES input [P206 (OFS1.PORTSELB = 0) selected] Keep-O
P212/IRQ1_B/TO00_A/TI03_C/TO03_C/RXD1_A/SI11_A/SDA11_A/SCLA0_B	Hi-Z	[IRQ1_B selected] IRQ1_B input [SCLA0_B selected] SCLA0_B inout [Other than the above] Keep-O
P213/IRQ0_B/TI00_A/TI02_B/TO02_B/TXD1_A/SO11_A/SDAA0_B	Hi-Z	[IRQ0_B selected] IRQ0_B input [SDAA0_B selected] SDAA0_B inout [Other than the above] Keep-O
P300/SWCLK/TI04_B/TO04_B/SDAA0_E/TXD1_B	Pull-up	[SDAA0_E selected] SDAA0_E inout [Other than the above] Keep-O

Note: Hi-Z: High-impedance
Keep-O: Output pins retain their previous values. Input pins become high-impedance.

表 A1.1 各处理模式下的端口状态 (2/2)

端口名称	重置	软件待机模式
P206/RES/IRQ1_C	引体向上	[IRQ1_C 已选择]IRQ1_C 输入[RES (OFS1.PORTSELB= 1) 已选择]RES 输入[P206 (OFS1.PORTSELB= 0) 已选择]保持 O
P212/IRQ1_B/TO00_A/TI03_C/TO03_C/RXD1_A/SI11_A/SDA11_A/SCLA0_B	高阻抗	[IRQ1_B 已选择] IRQ1_B 输入 [SCLA0_B 已选择] SCLA0_B 输入/输出 [除上述内容外] Keep-O
P213/IRQ0_B/TI00_A/TI02_B/TO02_B/TXD1_A/SO11_A/SDAA0_B	高阻抗	[IRQ0_B 已选择] IRQ0_B 输入 [SDAA0_B 已选择] SDAA0_B 输入/输出 [除上述内容外] Keep-O
P300/SWCLK/TI04_B/TO04_B/SDAA0_E/TXD1_B	引体向上	[SDAA0_E 已选择] SDAA0_E 输入/输出 [除上述内容外] Keep-O

注: Hi-Z: 高阻抗
Keep-O: 输出引脚保持其先前的值。输入引脚变为高阻抗。

Appendix 2. Package Dimensions

Information on the latest version of the package dimensions or mountings is displayed in “Packages” on the Renesas Electronics Corporation website.

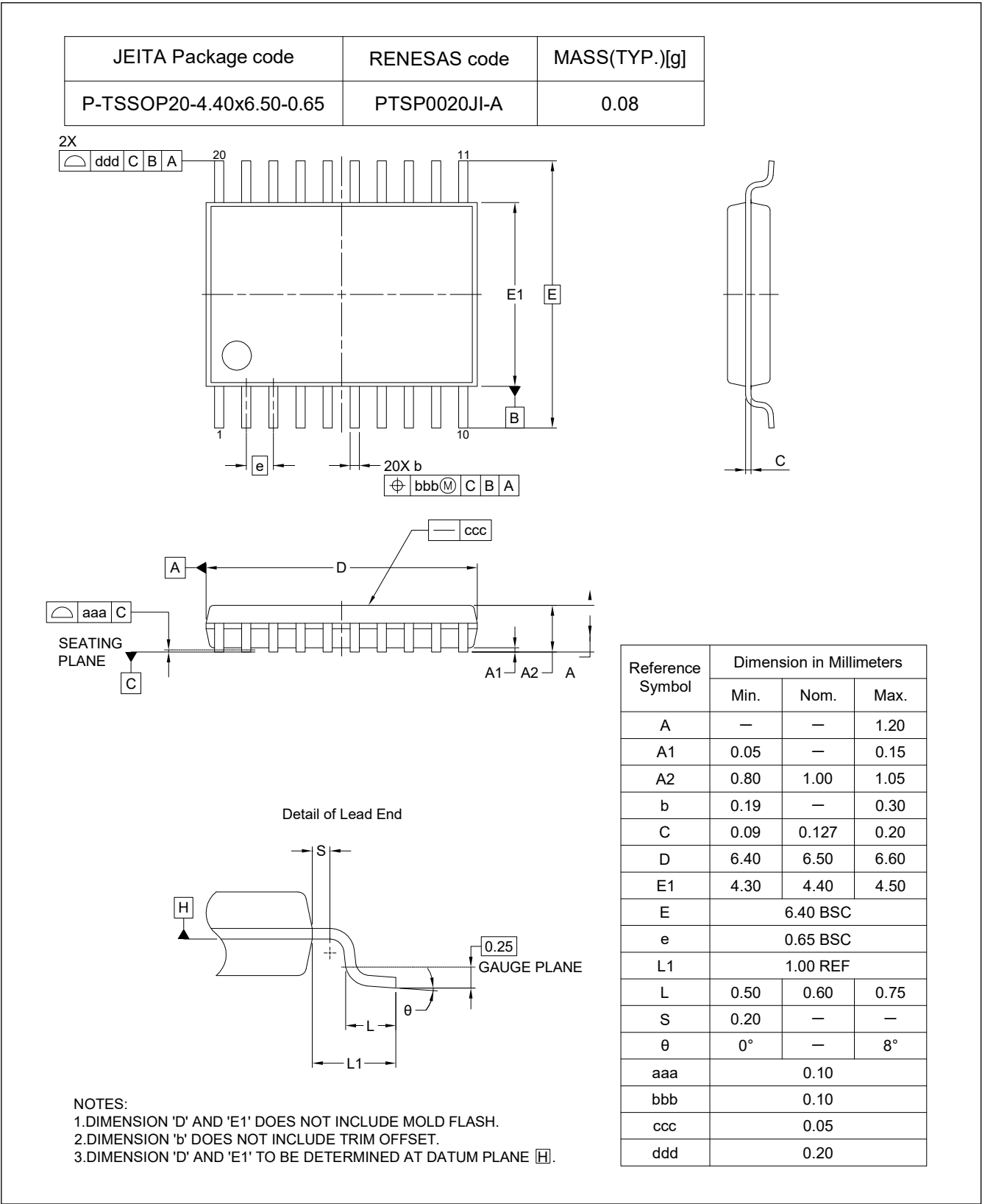


Figure A2.1 TSSOP 20-pin

附录 2. 包装尺寸

有关最新版本封装尺寸或安装方式的信息，请参阅瑞萨电子株式会社网站上的“封装”部分。

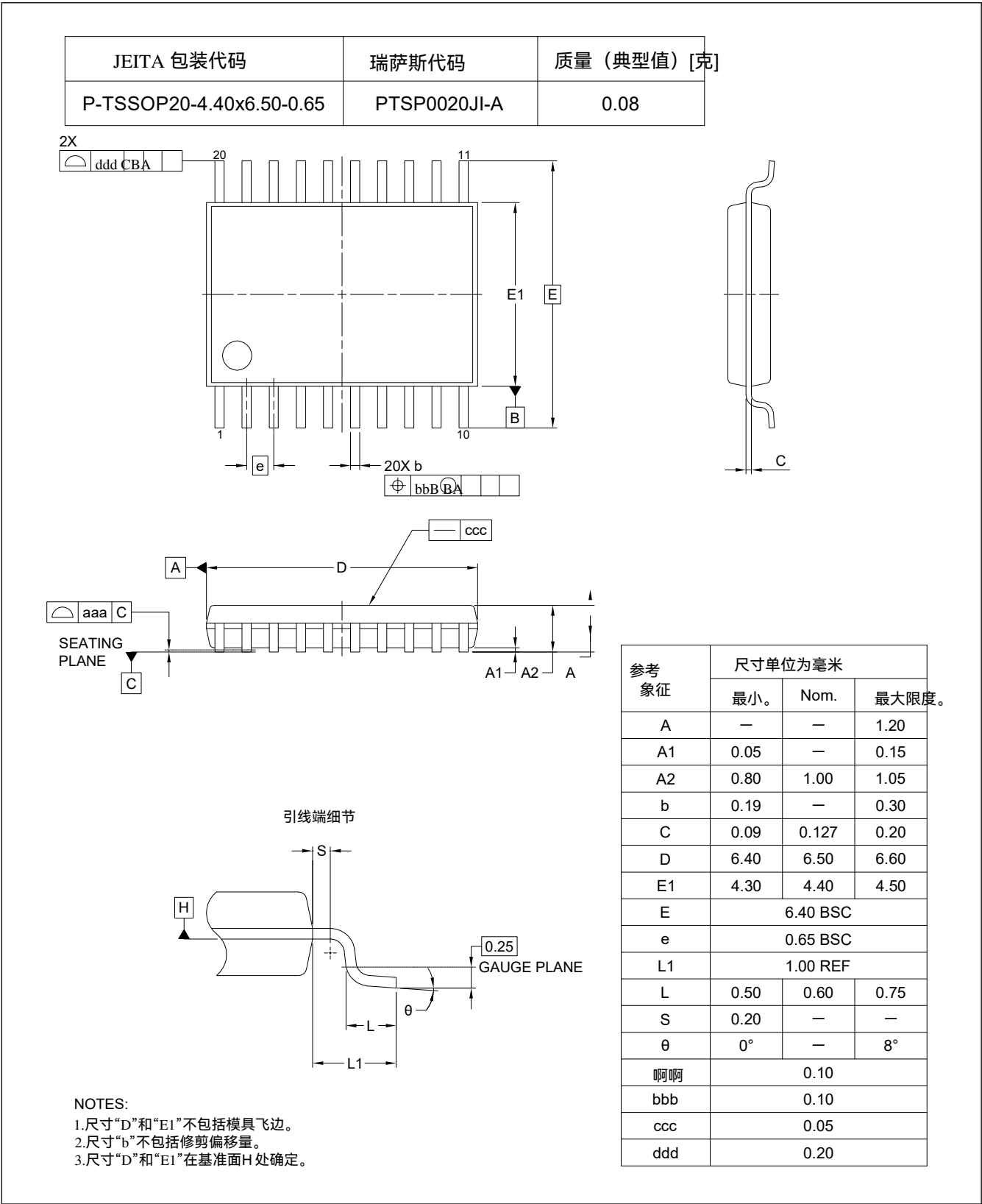


图 A2.1TSSOP 20引脚封装

Appendix 3. I/O Registers

This appendix describes I/O register addresses, access cycles, and reset values by function.

3.1 Peripheral Base Addresses

This section provides the base addresses for peripherals described in this manual.

Table A3.1 shows the name, description, and the base address of each peripheral.

Table A3.1 Peripheral base address

Name	Description	Base address
BUS	BUS Control	0x4000_3000
DTC	Data Transfer Controller	0x4000_5400
ICU	Interrupt Controller	0x4000_6000
DBG	Debug Function	0x4001_B000
SYSC	System Control	0x4001_E000
IWDT	Independent Watchdog Timer	0x4004_4400
MSTP	Module Stop Control	0x4004_7000
CRC	CRC Calculator	0x4007_4000
PORT0	Port 0 Control	0x400A_0000
PORT1	Port 1 Control	0x400A_0020
PORT2	Port 2 Control	0x400A_0040
PORT3	Port 3 Control	0x400A_0060
PFS_A	Pmn Pin Function Select	0x400A_0200
PORGA	Product Organize	0x400A_1000
ADC_D	10-bit A/D Converter	0x400A_1800
SAU0	Serial Array Unit 0	0x400A_2000
TAU	Timer Array Unit	0x400A_2600
IICA	I ² C Bus Interface	0x400A_3000
TML32	32-bit Interval Timer	0x400A_3800
PCLBUZ	Clock Output/Buzzer Output Controller	0x400A_3B00
FLCN	Flash I/O Registers	0x407E_C000

Note: Name = Peripheral name
Description = Peripheral functionality
Base address = Lowest reserved address or address used by the peripheral

3.2 Access Cycles

This section provides access cycle information for the I/O registers described in this manual.

The following information applies to Table A3.2:

- Registers are grouped by associated module.
- The number of access cycles indicates the number of cycles based on the specified reference clock.
- In the internal I/O area, reserved addresses that are not allocated to registers must not be accessed, otherwise operations cannot be guaranteed.
- The number of I/O access cycles depends on bus cycles of the internal peripheral bus, divided clock synchronization cycles, and wait cycles of each module.

Note: This applies to the number of cycles when access from the CPU does not conflict with the instruction fetching to the external memory or bus access from other bus master such as DTC.

附录 3. I/O 寄存器

本附录按功能描述 I/O 寄存器地址、访问周期和复位值。

3.1 外围设备基地址

本节提供本手册中所述外围设备的基地址。

表 A3.1 显示了每个外围设备的名称、描述和基地址。

表 A3.1 外围设备基址

Name	描述	基地址
BUS	总线控制	0x4000_3000
DTC	数据传输控制器	0x4000_5400
ICU	中断控制器	0x4000_6000
DBG	调试功能	0x4001_B000
SYSC	系统控制	0x4001_E000
IWDT	独立看门狗定时器	0x4004_4400
MSTP	模块停止控制	0x4004_7000
CRC	CRC计算器	0x4007_4000
PORT0	端口 0 控制	0x400A_0000
PORT1	端口 1 控制	0x400A_0020
PORT2	端口 2 控制	0x400A_0040
PORT3	端口 3 控制	0x400A_0060
PFS_A	PMN引脚功能选择	0x400A_0200
PORGA	产品组织	0x400A_1000
ADC_D	10位模数转换器	0x400A_1800
SAU0	串行阵列单元 0	0x400A_2000
TAU	定时器阵列单元	0x400A_2600
IICA	I2C总线接口	0x400A_3000
TML32	32 位间隔定时器	0x400A_3800
PCLBUZ	时钟输出/蜂鸣器输出控制器	0x400A_3B00
FLCN	闪存 I/O 寄存器	0x407E_C000

注意: 名称= 外围设备名称
描述=外围功能
基地址= 最低保留地址或外设使用的地址

3.2 访问周期

本节提供本手册中描述的I/O寄存器的访问周期信息。

以下信息适用于表 A3.2:

- 寄存器按关联模块分组。
- 访问周期数表示基于指定参考时钟的周期数。
- 在内部I/O 区域中，未分配给寄存器的保留地址不得访问，否则无法保证操作。
- I/O访问周期数取决于内部外设总线的总线周期、分频时钟同步周期以及每个模块的等待周期。

注意: 这适用于 CPU 访问与从外部存储器获取指令或从其他总线主控器（如 DTC）访问总线不冲突时的周期数。

Table A3.2 shows the register access cycles.

Table A3.2 Access cycles

Peripherals	Address		Number of access cycles			
	From	To	Read	Write	Cycle unit	Related function
BUS, DTC, ICU, DBG	0x4000_3000	0x4001_BFFF	3		ICLK	Buses, Data Transfer Controller, Interrupt Controller, CPU, Flash Memory
SYSC	0x4001_E000	0x4001_E6FF	2		ICLK	Low Power Modes, Resets, Low Voltage Detection, Clock Generation Circuit, Register Write Protection
IWDT, MSTP	0x4004_0000	0x4004_7FFF	3		PCLKB	Watchdog Timer, Module Stop Control
CRC	0x4007_4000	0x4007_4FFF	3		PCLKB	CRC Calculator
PORT, PFS_A, PORGA, ADC10, SAU0, TAU, IICA, TML32, PCLBUZ	0x400A_0000	0x400A_3FFF	2		PCLKB	I/O Ports, 10-bit A/D Converter, Serial Array Unit 0, Timer Array Unit, I ² C Bus Interface, 32-bit Interval Timer, Clock/Buzzer Output Controller
FLCN	0x407E_C000	0x407E_FFFF	7		ICLK	Flash Control

表 A3.2 显示了寄存器访问周期。

表 A3.2 访问周期

外围设备	地址		访问周期数			
	从	To	读	写	循环单元	相关功能
BUS、DTC、ICU、DBG	0x4000_3000	0x4001_BFFF	3		ICLK	总线、数据传输控制器 中断控制器、CPU、闪存 记忆
SYSC	0x4001_E000	0x4001_E6FF	2		ICLK	低功耗模式、重置、低 电压检测，时钟生成 电路寄存器写保护
IWDT，MSTP	0x4004_0000	0x4004_7FFF	3		PCLKB	看门狗定时器，模块停止控制
CRC	0x4007_4000	0x4007_4FFF	3		PCLKB	CRC计算器
端口、PFS_A、PORGA、 ADC10、SAU0、TAU、 IICA、TML32、PCLBUZ	0x400A_0000	0x400A_3FFF	2		PCLKB	输入/输出端口、10 位模数转换器、串行 数组单元 0，定时器数组单元，I ² C总线 接口，32 位间隔定时器，时钟/ 蜂鸣器输出控制器
FLCN	0x407E_C000	0x407E_FFFF	7		ICLK	闪光控制

Appendix 4. Peripheral Variant

Table A4.1 shows the correspondence between the module name used in this manual and the peripheral variant.

Table A4.1 Module name vs peripheral variant

Module name	Peripheral variant
ADC10	ADC_D

附录 4.外周变异

表 A4.1 显示了本手册中使用的模块名称与外围设备型号之间的对应关系。

表 A4.1 模块名称与外围设备变体

模块名称	外周变异
ADC10	ADC_D

Revision History

Revision 1.00 — Oct 29, 2025

Initial release

修订历史

版本 1.00 — 2025 年 10 月 29 日

初始版本

RA0E3 Group User's Manual: Hardware

Publication Date: Rev.1.00 Oct 29, 2025
Rev.0.50 Jun 16, 2025

Published by: Renesas Electronics Corporation

RA0E3 组用户手册：硬件

发布日期: Rev.1.00 2025年10月29日
Rev.0.50 2025年6月16日

出版方: 瑞萨电子公司

32-Bit MCU
RA0E3 Group

32位微控制器
RA0E3组