

致尊敬的顾客

关于产品目录等资料中的旧公司名称

NEC电子公司与株式会社瑞萨科技于2010年4月1日进行业务整合（合并），整合后的新公司暨“瑞萨电子公司”继承两家公司的所有业务。因此，本资料中虽还保留有旧公司名称等标识，但是并不妨碍本资料的有效性，敬请谅解。

瑞萨电子公司网址：<http://www.renesas.com>

2010年4月1日
瑞萨电子公司

【发行】瑞萨电子公司（<http://www.renesas.com>）

【业务咨询】<http://www.renesas.com/inquiry>

Notice

1. All information included in this document is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas Electronics products listed herein, please confirm the latest product information with a Renesas Electronics sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas Electronics such as that disclosed through our website.
2. Renesas Electronics does not assume any liability for infringement of 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. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
3. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part.
4. 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 of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
5. When exporting the products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations. You should not use Renesas Electronics products or the technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. Renesas Electronics products and technology may 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.
6. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
7. Renesas Electronics products are classified according to the following three quality grades: “Standard”, “High Quality”, and “Specific”. The recommended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below. You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application categorized as “Specific” without the prior written consent of Renesas Electronics. Further, you may not use any Renesas Electronics product for any application for which it is not intended without the prior written consent of Renesas Electronics. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for an application categorized as “Specific” or for which the product is not intended where you have failed to obtain the prior written consent of Renesas Electronics. The quality grade of each Renesas Electronics product is “Standard” unless otherwise expressly specified in a Renesas Electronics data sheets or data books, etc.
 - “Standard”: Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots.
 - “High Quality”: Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; safety equipment; and medical equipment not specifically designed for life support.
 - “Specific”: Aircraft; aerospace equipment; submersible repeaters; nuclear reactor control systems; medical equipment or systems for life support (e.g. artificial life support devices or systems), surgical implantations, or healthcare intervention (e.g. excision, etc.), and any other applications or purposes that pose a direct threat to human life.
8. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, 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, please evaluate the safety of the final products or system manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

(Note 1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) “Renesas Electronics product(s)” means any product developed or manufactured by or for Renesas Electronics.

SH7203 群

瑞萨 32 位 RISC 单片机

SuperHTM RISC engine 族/SH7200 系列

SH7203

R5S72030W200FP

Notes regarding these materials

1. This document is provided for reference purposes only so that Renesas customers may select the appropriate Renesas products for their use. Renesas neither makes warranties or representations with respect to the accuracy or completeness of the information contained in this document nor grants any license to any intellectual property rights or any other rights of Renesas or any third party with respect to the information in this document.
2. Renesas shall have no liability for damages or infringement of any intellectual property or other rights arising out of the use of any information in this document, including, but not limited to, product data, diagrams, charts, programs, algorithms, and application circuit examples.
3. You should not use the products or the technology described in this document for the purpose of military applications such as the development of weapons of mass destruction or for the purpose of any other military use. When exporting the products or technology described herein, you should follow the applicable export control laws and regulations, and procedures required by such laws and regulations.
4. All information included in this document such as product data, diagrams, charts, programs, algorithms, and application circuit examples, is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas products listed in this document, please confirm the latest product information with a Renesas sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas such as that disclosed through our website. (<http://www.renesas.com>)
5. Renesas has used reasonable care in compiling the information included in this document, but Renesas assumes no liability whatsoever for any damages incurred as a result of errors or omissions in the information included in this document.
6. When using or otherwise relying on the information in this document, you should evaluate the information in light of the total system before deciding about the applicability of such information to the intended application. Renesas makes no representations, warranties or guarantees regarding the suitability of its products for any particular application and specifically disclaims any liability arising out of the application and use of the information in this document or Renesas products.
7. With the exception of products specified by Renesas as suitable for automobile applications, Renesas products are not designed, manufactured or tested for applications or otherwise in systems the failure or malfunction of which may cause a direct threat to human life or create a risk of human injury or which require especially high quality and reliability such as safety systems, or equipment or systems for transportation and traffic, healthcare, combustion control, aerospace and aeronautics, nuclear power, or undersea communication transmission. If you are considering the use of our products for such purposes, please contact a Renesas sales office beforehand. Renesas shall have no liability for damages arising out of the uses set forth above.
8. Notwithstanding the preceding paragraph, you should not use Renesas products for the purposes listed below:
 - (1) artificial life support devices or systems
 - (2) surgical implantations
 - (3) healthcare intervention (e.g., excision, administration of medication, etc.)
 - (4) any other purposes that pose a direct threat to human lifeRenesas shall have no liability for damages arising out of the uses set forth in the above and purchasers who elect to use Renesas products in any of the foregoing applications shall indemnify and hold harmless Renesas Technology Corp., its affiliated companies and their officers, directors, and employees against any and all damages arising out of such applications.
9. You should use the products described herein within the range specified by Renesas, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas shall have no liability for malfunctions or damages arising out of the use of Renesas products beyond such specified ranges.
10. Although Renesas endeavors to improve the quality and reliability of its products, IC products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Please be sure to implement safety measures to guard against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas product, 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 applicable measures. Among others, since the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or system manufactured by you.
11. In case Renesas products listed in this document are detached from the products to which the Renesas products are attached or affixed, the risk of accident such as swallowing by infants and small children is very high. You should implement safety measures so that Renesas products may not be easily detached from your products. Renesas shall have no liability for damages arising out of such detachment.
12. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written approval from Renesas.
13. Please contact a Renesas sales office if you have any questions regarding the information contained in this document, Renesas semiconductor products, or if you have any other inquiries.

注意

本文只是参考译文，前页所载英文版“Cautions”具有正式效力。

关于利用本资料时的注意事项

1. 本资料是为了让用户根据用途选择合适的本公司产品的参考资料，对于本资料中所记载的技术信息，并非意味着对本公司或者第三者的知识产权及其他权利做出保证或对实施权力进行的承诺。
2. 对于因使用本资料所记载的产品数据、图、表、程序、算法及其他应用电路例而引起的损害或者对第三者的知识产权及其他权利造成侵犯，本公司不承担任何责任。
3. 不能将本资料所记载的产品和技术用于大规模破坏性武器的开发等目的、军事目的或其他的军需用途方面。另外，在出口时必须遵守日本的《外汇及外国贸易法》及其他出口的相关法令并履行这些法令中规定的必要手续。
4. 本资料所记载的产品数据、图、表、程序、算法以及其他应用电路例等所有信息均为本资料发行时的内容，本公司有可能在未做事先通知的情况下，对本资料所记载的产品或者产品规格进行更改。所以在购买和使用本公司的半导体产品之前，请事先向本公司的营业窗口确认最新的信息并经常留意本公司通过公司主页 (<http://www.renesas.com>) 等公开的最新信息。
5. 对于本资料中所记载的信息，制作时我们尽力保证出版时的精确性，但不承担因本资料的叙述不当而致使顾客遭受损失等的任何相关责任。
6. 在使用本资料所记载的产品数据、图、表等所示的技术内容、程序、算法及其他应用电路例时，不仅要对所使用的技术信息进行单独评价，还要对整个系统进行充分的评价。请顾客自行负责，进行是否适用的判断。本公司对于是否适用不负任何责任。
7. 本资料中所记载的产品并非针对万一出现故障或是错误运行就会威胁到人的生命或给人体带来危害的机器、系统(如各种安全装置或者运输交通用的、医疗、燃烧控制、航天器械、核能、海底中继用的机器和系统等)而设计和制造的, 特别是对于品质和可靠性要求极高的机器和系统等(将本公司指定用于汽车方面的产品用于汽车时除外)。如果要用于上述的目的, 请务必事先向本公司的营业窗口咨询。另外, 对于用于上述目的而造成的损失等, 本公司概不负责。
8. 除上述第7项内容外, 不能将本资料中记载的产品用于以下用途。如果用于以下用途而造成的损失, 本公司概不负责。
 - 1) 生命维持装置。
 - 2) 植埋于人体使用的装置。
 - 3) 用于治疗(切除患部、给药等)的装置。
 - 4) 其他直接影响到人的生命的装置。
9. 在使用本资料所记载的产品时, 对于最大额定值、工作电源电压的范围、放热特性、安装条件及其他条件请在本公司规定的保证范围内使用。如果超出了本公司规定的保证范围使用时, 对于由此而造成的故障和出现的事故, 本公司将不承担任何责任。
10. 本公司一直致力于提高产品的质量和可靠性, 但一般来说, 半导体产品总会以一定的概率发生故障、或者由于使用条件不同而出现错误运行等。为了避免因本公司的产品发生故障或者错误运行而导致人身事故和火灾或造成社会性的损失, 希望客户能自行负责进行冗余设计、采取延烧对策及进行防止错误运行等的安全设计(包括硬件和软件两方面的设计)以及老化处理等, 这是作为机器和系统的出厂保证。特别是单片机的软件, 由于单独进行验证很困难, 所以要求在顾客制造的最终的机器及系统上进行安全检验工作。
11. 如果把本资料所记载的产品从其载体设备上卸下, 有可能造成婴儿误吞的危险。顾客在将本公司产品安装到顾客的设备上时, 请顾客自行负责将本公司产品设置为不容易剥落的安全设计。如果从顾客的设备上剥落而造成事故时, 本公司将不承担任何责任。
12. 在未得到本公司的事先书面认可时, 不可将本资料的一部分或者全部转载或者复制。
13. 如果需要了解关于本资料的详细内容, 或者有其他关心的问题, 请向本公司的营业窗口咨询。

产品使用时的注意事项

本文对适用于单片机所有产品的“使用时的注意事项”进行说明。有关个别的使用时的注意事项请参照正文。此外，如果在记载上有与本手册的正文有差异之处，请以正文为准。

1. 未使用的引脚的处理

【注意】将未使用的引脚按照正文的“未使用引脚的处理”进行处理。

CMOS产品的输入引脚的阻抗一般为高阻抗。如果在开路的状态下运行未使用的引脚，由于感应现象，外加LSI周围的噪声，在LSI内部产生穿透电流，有可能被误认为是输入信号而引起误动作。未使用的引脚，请按照正文的“未使用引脚的处理”中的指示进行处理。

2. 通电时的处理

【注意】通电时产品处于不定状态。

通电时，LSI内部电路处于不确定状态，寄存器的设定和各引脚的状态不定。通过外部复位引脚对产品进行复位时，从通电到复位有效之前的期间，不能保证引脚的状态。

同样，使用内部上电复位功能对产品进行复位时，从通电到达到复位产生的一定电压的期间，不能保证引脚的状态。

3. 禁止存取保留地址（保留区）

【注意】禁止存取保留地址（保留区）

在地址区域中，有被分配将来用作功能扩展的保留地址（保留区）。因为无法保证存取这些地址时的运行，所以不能对保留地址（保留区）进行存取。

4. 关于时钟

【注意】复位时，请在时钟稳定后解除复位。

在程序运行中转换时钟时，请在要转换成的时钟稳定之后进行。复位时，在通过使用外部振荡器（或者外部振荡电路）的时钟开始运行的系统中，必须在时钟充分稳定后解除复位。另外，在程序运行中，转换成使用外部振荡器（或者外部振荡电路）的时钟时，在要转换成的时钟充分稳定后再进行转换。

5. 关于产品间的差异

【注意】在变更不同型号的产品时，请对每一个产品型号进行系统评价测试。

即使是同一个群的单片机，如果产品型号不同，由于内部ROM、版本模式等不同，在电特性范围内有时特性值、动作容限、噪声耐量、噪声辐射量等也不同。因此，在变更不同型号的产品时，请对每一个型号的产品进行系统评价测试。

本手册的结构

1 本手册的结构

本手册的内容结构如下：

1. 产品使用时的注意事项
2. 本手册的结构
3. 前言
4. 目录
5. 概要
6. 各功能模块的说明
 - CPU 及系统控制
 - 内部外围模块

各模块功能说明的结构因模块而异，一般由：

①特点、②输入 / 输出引脚、③寄存器说明、④运行说明、⑤使用时的注意事项等章节构成。

采用本 LSI 设计应用系统时，首先需充分确认注意事项。
各章正文记载有关于说明的注意事项，各章的末尾记载有使用时的注意事项。
必须仔细阅读。（各章节根据需要记载使用时的注意事项）。

7. 寄存器一览表
8. 电特性
9. 附录
 - 产品型号、封装尺寸图等
10. 本版的修订部分（仅适用于修订版）

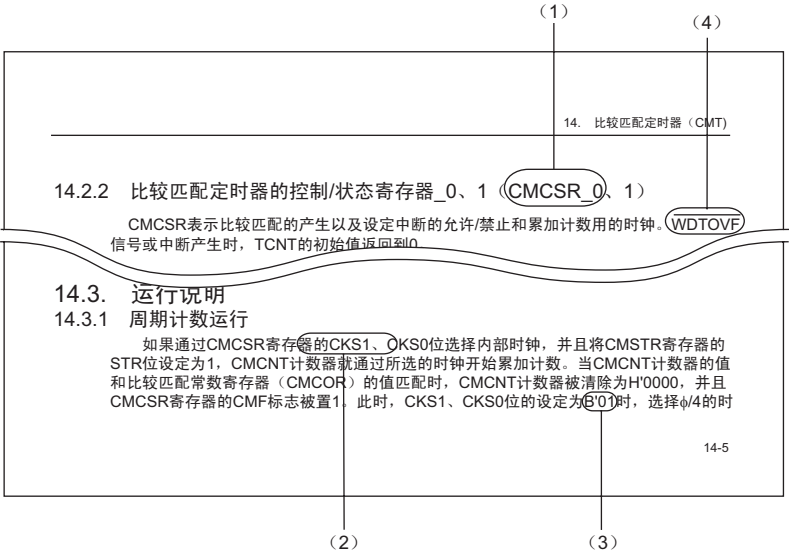
修订履历汇总了修正 / 追加前版内容的主要部分，并未记载所有修订内容，详情参阅本手册的正文。

11. 索引

2 凡例

本手册中使用的寄存器名、位名、数字和符号的表示范例说明如下。

- (1) 寄存器的表示
- 相同或类似的功能存在于多个通道时，使用“寄存器名称_通道号”的表示方法。
- (例) CMCSR_0
- (2) 位的表示
- 本文中所述位名以左侧为高位、右侧为低位的顺序表示。
- (例) CKS1、CKS0
- (3) 数字的表示
- 2进制数为B'xxxx或者xxxx，16进制数为H'xxxx，10进制数为xxxx
- (例) B'11或者11、H'EFA0、1234
- (4) 符号的表示
- 对低电平有效的信号加上划线。
- (例) WDTOVF



【注】 上述图中的位名和文章是举例，与本手册内容无关。

3 位图和位表的表示

各寄存器的说明中，有表示位排列的位图和说明位设定内容的位表。

(1) 位

表示位号码。
32位寄存器按31到0的顺序表示，16位寄存器按15到0的顺序表示。

(2) 位名

表示寄存器的位名或者字段名。
在字段需要明示位的位数时，追加如同ASID[3:0]一样的位数表示法。
保留位用“—”表示。
但也有不描述位名而表示为空白的情况，例如时间计数器等。

(3) 初始值

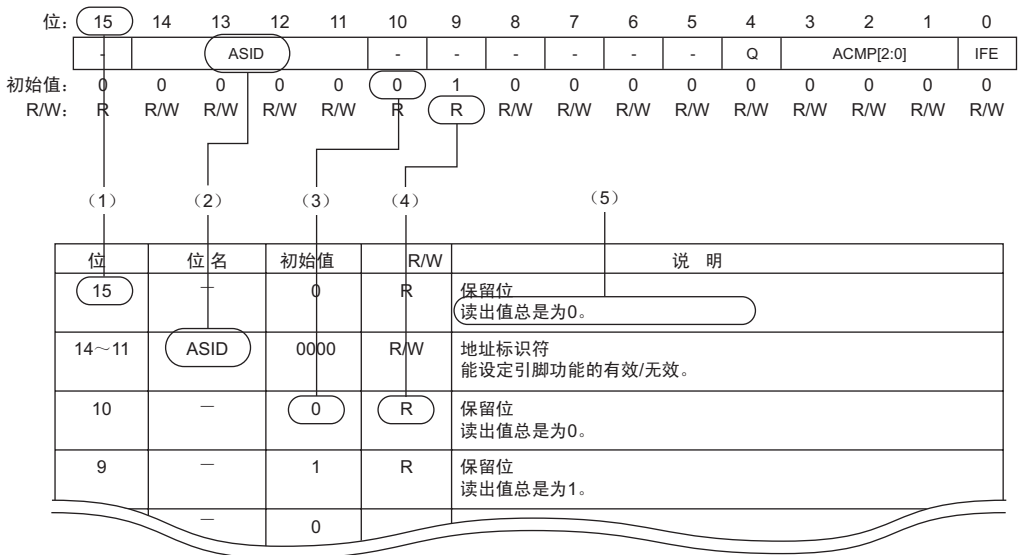
各位上电复位后的值作为初始值表示。
0 : 表示初始值为0。
1 : 表示初始值为1。
— : 表示初始值不定。

(4) R/W

表示各位可读出、写入或者不能读出、写入。
使用的表示法说明如下。
R/W : 可读出以及写入的位或者字段。
R/(W) : 可读出以及写入的位或者字段。
但仅是为了清除标志的写入才能进行。
R : 可读出位或者字段。
但保留位全部表示为“R”。需要写入时，
请写入位表指定的值或者初始值。
W : 可写入的位或者字段。
但读出的值是位表记载以外的情况时，不被保证。

(5) 说明

对通过设定位实现的功能进行说明。



【注】 上述图中的位名和文章是举例，与本手册内容无关。

所有商标及注册商标分别归属于其所有者。

前言

本 LSI 以瑞萨科技独创的 RISC 结构 CPU 为内核，是集成了系统构成所需外围功能的 RISC 单片机。

对象：本手册以使用本 LSI 设计应用系统的用户为对象。使用本手册的读者需具备电路、逻辑电路及单片机的相关基础知识。

目的：本手册以让用户了解本 LSI 的硬件功能与电特性为目的。

阅读方法

- 希望了解全部功能时
 - 请按照目录阅读。
 - 本手册大致以 CPU、系统控制功能、外围功能、电特性的顺序构成。
- 希望了解 CPU 功能的详细内容时
 - 请参阅“SH-2A、SH2A-FPU 软件手册”。
- 已知寄存器名称，希望了解详细功能时
 - 本手册后面附有“索引”。请从索引查找页码。
 - “第 30 章 寄存器一览表”汇总了地址、位、初始化等内容。

目 录

第 1 章	概要	1
1.1	SH7203 的特点	1
1.2	产品一览表	6
1.3	框图	6
1.4	引脚排列图	7
1.5	引脚功能	8
1.6	引脚一览表	13
第 2 章	CPU	28
2.1	寄存器结构	28
2.1.1	通用寄存器	28
2.1.2	控制寄存器	29
2.1.3	系统寄存器	30
2.1.4	寄存器存储体	31
2.1.5	寄存器的初始值	31
2.2	数据格式	32
2.2.1	寄存器的数据格式	32
2.2.2	存储器的数据格式	32
2.2.3	立即数的数据格式	32
2.3	指令特点	33
2.3.1	RISC 结构	33
2.3.2	寻址方式	36
2.3.3	指令格式	40
2.4	指令系统	43
2.4.1	指令系统分类	43
2.4.2	数据传送指令	47
2.4.3	算术运算指令	50
2.4.4	逻辑运算指令	52
2.4.5	移位指令	53
2.4.6	转移指令	54
2.4.7	系统控制指令	55
2.4.8	浮点运算指令	57
2.4.9	FPU 相关的 CPU 指令	59
2.4.10	位操作指令	59
2.5	处理状态	60
第 3 章	浮点单元 (FPU)	62
3.1	特点	62
3.2	数据格式	62
3.2.1	浮点格式	62
3.2.2	非数 (NaN)	64
3.2.3	非规格化数	64
3.3	寄存器说明	65
3.3.1	浮点寄存器	65
3.3.2	浮点状态 / 控制寄存器 (FPSCR)	66
3.3.3	浮点通信寄存器 (FPUL)	67
3.4	舍入	67
3.5	FPU 异常	68
3.5.1	FPU 异常源	68
3.5.2	FPU 异常处理	68

第 4 章	时钟脉冲振荡器 (CPG)	69
4.1	特点	69
4.2	输入 / 输出引脚	72
4.3	时钟运行模式	73
4.4	寄存器说明	76
4.4.1	频率控制寄存器 (FRQCR)	76
4.5	频率更改方法	78
4.5.1	倍频比的更改	78
4.5.2	分频比的更改	78
4.5.3	PLL 振荡电路使用时的注意事项	79
4.6	使用时的注意事项	79
第 5 章	异常处理	80
5.1	概要	80
5.1.1	异常处理的种类与优先顺序	80
5.1.2	异常处理的运行	81
5.1.3	异常处理向量表	82
5.2	复位	83
5.2.1	输入 / 输出引脚	83
5.2.2	复位种类	84
5.2.3	上电复位	84
5.2.4	手动复位	85
5.3	地址错误	86
5.3.1	地址错误的产生源	86
5.3.2	地址错误异常处理	86
5.4	寄存器存储体错误	87
5.4.1	寄存器存储体错误的产生源	87
5.4.2	寄存器存储体错误异常处理	87
5.5	中断	88
5.5.1	中断源	88
5.5.2	中断优先顺序	89
5.5.3	中断异常处理	89
5.6	指令引起的异常	90
5.6.1	指令引起的异常种类	90
5.6.2	陷阱指令	90
5.6.3	槽非法指令	91
5.6.4	一般非法指令	91
5.6.5	整数除法异常	91
5.6.6	FPU 异常	92
5.7	不接受异常处理时	92
5.8	异常处理后的堆栈状态	93
5.9	使用时的注意事项	94
5.9.1	堆栈指针 (SP) 的值	94
5.9.2	向量基址寄存器 (VBR) 的值	94
5.9.3	在地址错误异常处理的堆栈存取时产生的地址错误	94
第 6 章	中断控制器 (INTC)	95
6.1	特点	95
6.2	输入 / 输出引脚	96
6.3	寄存器说明	97
6.3.1	中断优先级设定寄存器 01、02、05 ~ 17 (IPR01、IPR02、IPR05 ~ IPR17)	98
6.3.2	中断控制寄存器 0 (ICR0)	99
6.3.3	中断控制寄存器 1 (ICR1)	100

6.3.4	中断控制寄存器 2 (ICR2)	101
6.3.5	IRQ 中断请求寄存器 (IRQRR)	102
6.3.6	PINT 中断允许寄存器 (PINTER)	103
6.3.7	PINT 中断请求寄存器 (PIRR)	104
6.3.8	存储体控制寄存器 (IBCR)	105
6.3.9	存储体编号寄存器 (IBNR)	106
6.4	中断源	107
6.4.1	NMI 中断	107
6.4.2	用户断点中断	107
6.4.3	H-UDI 中断	107
6.4.4	IRQ 中断	107
6.4.5	PINT 中断	108
6.4.6	内部外围模块中断	108
6.5	中断异常处理向量表与优先顺序	108
6.6	运行说明	113
6.6.1	中断运行的流程	113
6.6.2	中断异常处理结束后的堆栈状态	115
6.7	中断响应时间	115
6.8	寄存器存储器	120
6.8.1	存储体的对象寄存器与输入 / 输出方式	120
6.8.2	存储体保存、返回的运行	121
6.8.3	在保存至所有存储体的状态下执行的保存、返回	122
6.8.4	寄存器存储体异常	122
6.8.5	寄存器存储体错误异常处理	122
6.9	通过中断请求信号传送数据	123
6.9.1	中断请求信号作为 CPU 中断源而非 DMAC 启动源时	124
6.9.2	中断请求信号作为 DMAC 启动源而非 CPU 中断源时	124
6.10	使用时的注意事项	124
6.10.1	清除中断源的时序	124
6.10.2	IRQOUT 的无效时序	124
第 7 章	用户断点控制器 (UBC)	125
7.1	特点	125
7.2	输入 / 输出引脚	127
7.3	寄存器说明	127
7.3.1	断点地址寄存器 (BAR)	128
7.3.2	断点地址屏蔽寄存器 (BAMR)	128
7.3.3	断点数据寄存器 (BDR)	129
7.3.4	断点数据屏蔽寄存器 (BDMR)	129
7.3.5	断点总线周期寄存器 (BBR)	130
7.3.6	断点控制寄存器 (BRCR)	131
7.4	运行说明	133
7.4.1	用户断点运行的流程	133
7.4.2	取指令周期的断点	133
7.4.3	数据存取周期的断点	134
7.4.4	被保存的程序计数器值	135
7.4.5	使用例	135
7.5	使用时的注意事项	137
第 8 章	高速缓存	138
8.1	特点	138
8.1.1	高速缓存的结构	138
8.2	寄存器说明	139

8.2.1	高速缓存控制寄存器 1 (CCR1)	140
8.2.2	高速缓存控制寄存器 2 (CCR2)	141
8.3	运行说明	143
8.3.1	高速缓存的搜索	143
8.3.2	读取操作	144
8.3.3	预取操作 (仅限操作数高速缓存)	144
8.3.4	写入操作 (仅限操作数高速缓存)	144
8.3.5	回写缓冲器 (仅限操作数高速缓存)	145
8.3.6	高速缓存与外部存储器的比配	146
8.4	存储器分配高速缓存的结构	146
8.4.1	地址阵列	146
8.4.2	数据阵列	147
8.4.3	使用例	148
8.4.4	注意事项	148
第 9 章	总线状态控制器 (BSC)	149
9.1	特点	149
9.2	输入 / 输出引脚	152
9.3	区域概要	153
9.3.1	地址映射	153
9.3.2	各区域的数据总线宽度与引脚功能设定	154
9.4	寄存器说明	155
9.4.1	共用控制寄存器 (CMNCR)	156
9.4.2	CSn 空间总线控制寄存器 (CSnBCR) (n=0 ~ 7)	158
9.4.3	CSn 空间等待控制寄存器 (CSnWCR) (n=0 ~ 7)	161
9.4.4	SDRAM 控制寄存器 (SDCR)	184
9.4.5	刷新定时器控制 / 状态寄存器 (RTC SR)	186
9.4.6	刷新定时器计数器 (RTCNT)	187
9.4.7	刷新时间常数寄存器 (RTCOR)	187
9.5	运行说明	188
9.5.1	字节序 / 存取长度与数据对齐	188
9.5.2	普通空间接口	192
9.5.3	存取等待控制	197
9.5.4	CSn 有效期间的扩展	199
9.5.5	MPX-I/O 接口	200
9.5.6	SDRAM 接口	202
9.5.7	突发 ROM (时钟异步) 接口	228
9.5.8	带字节选择的 SRAM 接口	230
9.5.9	PCMCIA 接口	234
9.5.10	突发 MPX-I/O 接口	238
9.5.11	突发 ROM (时钟同步) 接口	241
9.5.12	存取周期之间的空闲周期	242
9.5.13	总线仲裁	248
9.5.14	其他	249
第 10 章	直接存储器存取控制器 (DMAC)	251
10.1	特点	251
10.2	输入 / 输出引脚	253
10.3	寄存器说明	253
10.3.1	DMA 源地址寄存器 (SAR)	256
10.3.2	DMA 目标地址寄存器 (DAR)	256
10.3.3	DMA 传送计数寄存器 (DMATCR)	256
10.3.4	DMA 通道控制寄存器 (CHCR)	257

10.3.5	DMA 重加载源地址寄存器 (RSAR)	261
10.3.6	DMA 重加载目标地址寄存器 (RDAR)	261
10.3.7	DMA 重加载传送计数寄存器 (RDMATCR)	262
10.3.8	DMA 操作寄存器 (DMAOR)	262
10.3.9	DMA 扩展资源选择器 0 ~ 3 (DMARS0 ~ DMARS3)	265
10.4	运行说明	267
10.4.1	传送流程	267
10.4.2	DMA 传送请求	269
10.4.3	通道的优先顺序	272
10.4.4	DMA 传送的种类	275
10.4.5	总线周期的状态数和 DREQ 引脚的采样时序	281
10.5	使用时的注意事项	284
10.5.1	半结束标志的置位及半结束中断	284
10.5.2	DACK 输出及 TEND 输出的时序	284
10.5.3	使用外部请求模式的注意事项	284
10.5.4	使用内部外围模块请求模式或自动请求模式时的注意事项	285
第 11 章	多功能定时器脉冲单元 2 (MTU2)	286
11.1	特点	286
11.2	输入 / 输出引脚	289
11.3	寄存器说明	290
11.3.1	定时器控制寄存器 (TCR)	293
11.3.2	定时器模式寄存器 (TMDR)	296
11.3.3	定时器 I/O 控制寄存器 (TIOR)	298
11.3.4	定时器中断允许寄存器 (TIER)	307
11.3.5	定时器状态寄存器 (TSR)	309
11.3.6	定时器缓冲运行传送模式寄存器 (TBTM)	312
11.3.7	定时器输入捕捉控制寄存器 (TICCR)	313
11.3.8	定时器 A/D 转换开始请求控制寄存器 (TADCR)	314
11.3.9	定时器 A/D 转换开始请求周期设定寄存器 (TADCORA/B_4)	315
11.3.10	定时器 A/D 转换开始请求周期设定缓冲寄存器 (TADCOBRA/B_4)	316
11.3.11	定时器计数器 (TCNT)	316
11.3.12	定时器通用寄存器 (TGR)	316
11.3.13	定时器开始寄存器 (TSTR)	317
11.3.14	定时器同步寄存器 (TSYR)	318
11.3.15	定时器读取 / 写入允许寄存器 (TRWER)	319
11.3.16	定时器输出主控允许寄存器 (TOER)	320
11.3.17	定时器输出控制寄存器 1 (TOCR1)	321
11.3.18	定时器输出控制寄存器 2 (TOCR2)	323
11.3.19	定时器输出电平缓冲寄存器 (TOLBR)	325
11.3.20	定时器门控寄存器 (TGCR)	326
11.3.21	定时器副计数器 (TCNTS)	327
11.3.22	定时器空载时间数据寄存器 (TDDR)	327
11.3.23	定时器周期数据寄存器 (TCDR)	327
11.3.24	定时器周期缓冲寄存器 (TCBR)	328
11.3.25	定时器中断跳过设定寄存器 (TITCR)	328
11.3.26	定时器中断跳过次数计数器 (TITCNT)	330
11.3.27	定时器缓冲传送设定寄存器 (TBTER)	331
11.3.28	定时器空载时间允许寄存器 (TDER)	332
11.3.29	定时器波形控制寄存器 (TWCR)	333
11.3.30	与总线主控器的接口	334
11.4	运行说明	334
11.4.1	基本运行	334

11.4.2	同步运行	339
11.4.3	缓冲运行	340
11.4.4	级联运行	344
11.4.5	PWM 模式	348
11.4.6	相位计数模式	352
11.4.7	复位同步 PWM 模式	358
11.4.8	互补 PWM 模式	360
11.4.9	A/D 转换开始请求延迟功能	388
11.4.10	互补 PWM 的“波峰 / 波谷”的 TCNT 捕捉运行	391
11.5	中断源	392
11.5.1	中断源和优先顺序	392
11.5.2	DMAC 的启动	393
11.5.3	A/D 转换器的启动	393
11.6	运行时序	395
11.6.1	输入 / 输出时序	395
11.6.2	中断信号时序	400
11.7	使用时的注意事项	403
11.7.1	模块待机模式的设定	403
11.7.2	输入时钟的限制事项	403
11.7.3	周期设定时的注意事项	403
11.7.4	TCNT 的写入和清除的竞争	404
11.7.5	TCNT 的写入和递增计数的竞争	404
11.7.6	TGR 的写入和比较匹配的竞争	405
11.7.7	缓冲寄存器的写入和比较匹配的竞争	405
11.7.8	缓冲寄存器的写入和 TCNT 清除的竞争	406
11.7.9	TGR 的读取和输入捕捉的竞争	406
11.7.10	TGR 的写入和输入捕捉的竞争	407
11.7.11	缓冲寄存器的写入和输入捕捉的竞争	407
11.7.12	级联中 TCNT_2 的写入和上溢 / 下溢的竞争	408
11.7.13	互补 PWM 模式停止时的计数器值	409
11.7.14	互补 PWM 模式时的缓冲运行的设定	409
11.7.15	复位同步 PWM 模式的缓冲运行和比较匹配标志	410
11.7.16	复位同步 PWM 模式的上溢标志	411
11.7.17	上溢 / 下溢和计数器清除的竞争	411
11.7.18	TCNT 的写入和上溢 / 下溢的竞争	412
11.7.19	从普通运行或 PWM 模式 1 转移至复位同步 PWM 模式时的注意事项	412
11.7.20	互补 PWM 模式、复位同步 PWM 模式的输出电平	412
11.7.21	模块待机时的中断	412
11.7.22	级联时 TCNT_1 和 TCNT_2 同时输入捕捉	413
11.8	MTU2 输出引脚的初始化方法	413
11.8.1	运行模式	413
11.8.2	复位开始时的运行	413
11.8.3	运行过程中因异常等而重新设定时的运行	414
11.8.4	因运行过程中的异常等引起引脚初始化的步骤、模式转移的概要	414
第 12 章	比较匹配定时器 (CMT)	433
12.1	特点	433
12.2	寄存器说明	434
12.2.1	比较匹配定时器开始寄存器 (CMSTR)	434
12.2.2	比较匹配定时器控制 / 状态寄存器 (CMCSR)	435
12.2.3	比较匹配计数器 (CMCNT)	436
12.2.4	比较匹配常数寄存器 (CMCOR)	436
12.3	运行说明	436

12.3.1	期间计数运行	436
12.3.2	CMCNT 计数时序	437
12.4	中断	437
12.4.1	中断源与 DMA 传送请求	437
12.4.2	比较匹配标志的置位时序	437
12.4.3	比较匹配标志的清除时序	438
12.5	使用时的注意事项	438
12.5.1	CMCNT 的写入与比较匹配的竞争	438
12.5.2	CMCNT 的字写入与递增计数的竞争	438
12.5.3	CMCNT 的字节写入与递增计数的竞争	439
第 13 章	看门狗定时器 (WDT)	440
13.1	特点	440
13.2	输入 / 输出引脚	441
13.3	寄存器说明	441
13.3.1	看门狗定时器计数器 (WTCNT)	441
13.3.2	看门狗定时器控制 / 状态寄存器 (WTC SR)	442
13.3.3	看门狗复位控制 / 状态寄存器 (WRCSR)	443
13.3.4	寄存器存取时的注意事项	444
13.4	WDT 的使用方法	445
13.4.1	软件待机模式的解除步骤	445
13.4.2	频率更改步骤	445
13.4.3	看门狗定时器模式的使用方法	445
13.4.4	间隔定时器模式的使用方法	446
13.5	使用时的注意事项	447
13.5.1	定时器误差	447
13.5.2	禁止设定 H'FF 为 WTCNT 的设定值	447
13.5.3	间隔定时器上溢标志	447
13.5.4	WDTOVF 信号引起的系统复位	447
13.5.5	看门狗定时器模式的手动复位	447
第 14 章	实时时钟 (RTC)	448
14.1	特点	448
14.2	输入 / 输出引脚	450
14.3	寄存器说明	450
14.3.1	64Hz 计数器 (R64CNT)	451
14.3.2	秒计数器 (RSECCNT)	451
14.3.3	分计数器 (RMINCNT)	452
14.3.4	小时计数器 (RHRCNT)	452
14.3.5	星期计数器 (RWKCNT)	453
14.3.6	日计数器 (RDAYCNT)	453
14.3.7	月计数器 (RMONCNT)	454
14.3.8	年计数器 (RYRCNT)	454
14.3.9	秒报警寄存器 (RSECAR)	455
14.3.10	分报警寄存器 (RMINAR)	455
14.3.11	小时报警寄存器 (RHRAR)	456
14.3.12	星期报警寄存器 (RWKAR)	456
14.3.13	日报警寄存器 (RDAYAR)	457
14.3.14	月报警寄存器 (RMONAR)	457
14.3.15	年报警寄存器 (RYRAR)	458
14.3.16	RTC 控制寄存器 1 (RCR1)	458
14.3.17	RTC 控制寄存器 2 (RCR2)	459
14.3.18	RTC 控制寄存器 3 (RCR3)	460

14.4	运行说明	460
14.4.1	上电后寄存器的初始设定	460
14.4.2	时间设定步骤	461
14.4.3	时间读取步骤	461
14.4.4	报警功能	462
14.5	使用时的注意事项	463
14.5.1	关于 RTC 计数运行时的寄存器写入	463
14.5.2	关于实时时钟 (RTC) 周期中断的使用	463
14.5.3	寄存器设定后的待机转移	463
14.5.4	读取 / 写入寄存器时的注意事项	463
第 15 章	带 FIFO 的串行通信接口 (SCIF)	464
15.1	特点	464
15.2	输入 / 输出引脚	465
15.3	寄存器说明	466
15.3.1	接收移位寄存器 (SCRSR)	467
15.3.2	接收 FIFO 数据寄存器 (SCFRDR)	467
15.3.3	发送移位寄存器 (SCTSR)	467
15.3.4	发送 FIFO 数据寄存器 (SCFTDR)	468
15.3.5	串行模式寄存器 (SCSMR)	468
15.3.6	串行控制寄存器 (SCSCR)	470
15.3.7	串行状态寄存器 (SCFSR)	472
15.3.8	位速率寄存器 (SCBRR)	476
15.3.9	FIFO 控制寄存器 (SCFCR)	483
15.3.10	FIFO 数据计数设定寄存器 (SCFDR)	485
15.3.11	串行端口寄存器 (SCSPTR)	486
15.3.12	线状态寄存器 (SCLSR)	487
15.3.13	串行扩展模式寄存器 (SCEMR)	488
15.4	运行说明	489
15.4.1	概要	489
15.4.2	异步模式时的运行	490
15.4.3	时钟同步模式时的运行	498
15.5	SCIF 的中断	505
15.6	使用时的注意事项	505
15.6.1	向 SCFTDR 的写入和 TDFE 标志	505
15.6.2	SCFRDR 的读取和 RDF 标志	505
15.6.3	DMAC 使用时的限制事项	506
15.6.4	断点的检测和处理	506
15.6.5	断点的发送	506
15.6.6	异步模式的接收数据采样时序与接收容限	506
15.6.7	选择异步基本时钟	507
第 16 章	同步串行通信单元 (SSU)	508
16.1	特点	508
16.2	输入 / 输出引脚	510
16.3	寄存器说明	510
16.3.1	SS 控制寄存器 H (SSCRH)	511
16.3.2	SS 控制寄存器 L (SSCRL)	512
16.3.3	SS 模式寄存器 (SSMR)	513
16.3.4	SS 允许寄存器 (SSER)	514
16.3.5	SS 状态寄存器 (SSSR)	515
16.3.6	SS 控制寄存器 2 (SSCR2)	517
16.3.7	SS 发送数据寄存器 0 ~ 3 (SSTDR0 ~ SSTDR3)	518

16.3.8	SS 接收数据寄存器 0 ~ 3 (SSRDR0 ~ SSRDR3)	519
16.3.9	SS 移位寄存器 (SSTRSR)	519
16.4	运行说明	520
16.4.1	传送时钟	520
16.4.2	时钟的相位、极性与数据的关系	520
16.4.3	数据输入 / 输出引脚与移位寄存器的关系	521
16.4.4	各通信模式与引脚功能	522
16.4.5	SSU 模式	523
16.4.6	SCS 引脚控制与竞争错误	530
16.4.7	时钟同步通信模式	531
16.5	SSU 的中断源与 DMAC	536
16.6	使用时的注意事项	536
16.6.1	模块待机模式的设定	536
16.6.2	在 SSU 从属模式连续发送 / 接收时的注意事项	536
第 17 章	I ² C 总线接口 3 (IIC3)	537
17.1	特点	537
17.2	输入 / 输出引脚	539
17.3	寄存器说明	540
17.3.1	I ² C 总线控制寄存器 1 (ICCR1)	541
17.3.2	I ² C 总线控制寄存器 2 (ICCR2)	543
17.3.3	I ² C 总线模式寄存器 (ICMR)	544
17.3.4	I ² C 总线中断允许寄存器 (ICIER)	545
17.3.5	I ² C 总线状态寄存器 (ICSR)	546
17.3.6	从属地址寄存器 (SAR)	548
17.3.7	I ² C 总线发送数据寄存器 (ICDRT)	548
17.3.8	I ² C 总线接收数据寄存器 (ICDRR)	548
17.3.9	I ² C 总线移位寄存器 (ICDRS)	549
17.3.10	NF2CYC 寄存器 (NF2CYC)	549
17.4	运行说明	550
17.4.1	I ² C 总线格式	550
17.4.2	主发送运行	551
17.4.3	主接收运行	552
17.4.4	从属发送运行	554
17.4.5	从属接收运行	555
17.4.6	时钟同步串行格式	557
17.4.7	噪声消除电路	559
17.4.8	使用例	560
17.5	中断请求	564
17.6	位同步电路	564
17.7	使用时的注意事项	566
17.7.1	设定 ICCR1.CKS[3:0] 时的注意事项	566
17.7.2	用作多主器件时的注意事项	566
17.7.3	主接收模式时的注意事项	566
17.7.4	主接收模式、ACKBT 设定时的注意事项	566
17.7.5	仲裁失败时 MST 与 TRN 位状态的注意事项	566
第 18 章	串行音频接口 (SSI)	567
18.1	特点	567
18.2	输入 / 输出引脚	569
18.3	寄存器说明	570
18.3.1	控制寄存器 (SSICR)	571
18.3.2	状态寄存器 (SSISR)	575

18.3.3	发送数据寄存器 (SSITDR)	577
18.3.4	接收数据寄存器 (SSIRDR)	578
18.4	运行说明	578
18.4.1	总线格式	578
18.4.2	非压缩模式	578
18.4.3	运行模式	586
18.4.4	发送运行	587
18.4.5	接收运行	589
18.4.6	发送时暂时停止、重新开始步骤	591
18.4.7	串行位时钟控制	591
18.5	使用时的注意事项	591
18.5.1	接收 DMA 运行过程中产生上溢时的限制事项	591
第 19 章	控制器局域网 (RCAN-TL1)	592
19.1	特点	592
19.1.1	RCAN-TL1 的特点	592
19.1.2	本单片机的特点	593
19.2	结构	593
19.2.1	框图	593
19.2.2	各块的功能	594
19.2.3	引脚结构	595
19.2.4	存储器映射	596
19.3	邮箱	597
19.3.1	邮箱结构	597
19.3.2	信息控制字段	601
19.3.3	局部接收过滤器屏蔽 (LAFM)	604
19.3.4	信息数据字段	605
19.3.5	时戳	606
19.3.6	发送触发时间 (TTT) 与时间触发控制	607
19.4	RCAN-TL1 的控制寄存器	610
19.4.1	主控寄存器 (MCR)	610
19.4.2	通用状态寄存器 (GSR)	614
19.4.3	位结构寄存器 0、1 (BCR0、BCR1)	615
19.4.4	中断请求寄存器 (IRR)	619
19.4.5	中断屏蔽寄存器 (IMR)	623
19.4.6	发送错误计数器 (TEC) / 接收错误计数器 (REC)	623
19.5	RCAN-TL1 邮箱寄存器	624
19.5.1	发送等待寄存器 1、0 (TXPR1、TXPR0)	625
19.5.2	发送取消寄存器 1、0 (TXCR1、TXCR0)	627
19.5.3	发送应答寄存器 1、0 (TXACK1、TXACK0)	628
19.5.4	中止应答寄存器 1、0 (ABACK1、ABACK0)	629
19.5.5	数据帧接收完成寄存器 1、0 (RXPR1、RXPR0)	630
19.5.6	远程帧接收完成寄存器 1、0 (RFPR1、RFPR0)	631
19.5.7	邮箱中断屏蔽寄存器 1、0 (MBIMR1、MBIMR0)	632
19.5.8	未读信息状态寄存器 1、0 (UMSR1、UMSR0)	633
19.6	定时器寄存器	634
19.6.1	时间触发控制寄存器 0 (TTCR0)	634
19.6.2	最大周期 /Tx_Enable_Window 寄存器 (CMAX_TEW)	636
19.6.3	基准触发偏移寄存器 (RFTROFF)	637
19.6.4	定时器状态寄存器 (TSR)	638
19.6.5	周期计数器寄存器 (CCR)	640
19.6.6	定时器计数器寄存器 (TCNTR)	641
19.6.7	周期时间寄存器 (CYCTR)	641

19.6.8	基准标记寄存器 (RFMK)	642
19.6.9	定时器比较匹配寄存器 0 ~ 2 (TCMR0 ~ 2)	642
19.6.10	发送触发时间选择寄存器 (TTTSEL)	644
19.7	运行说明	645
19.7.1	RCAN-TL1 的设定	645
19.7.2	测试模式的设定	648
19.7.3	信息发送顺序	649
19.7.4	信息接收顺序	659
19.7.5	邮箱的重新设定	660
19.8	中断源	662
19.9	DMAC 接口	663
19.10	CAN 总线接口	664
19.11	RCAN-TL1 引脚端口设定	664
19.12	使用时的注意事项	665
19.12.1	设定 1 个通道 64 个邮箱的端口时的注意事项	665
第 20 章	A/D 转换器 (ADC)	666
20.1	特点	666
20.2	输入 / 输出引脚	667
20.3	寄存器说明	667
20.3.1	A/D 数据寄存器 A ~ H (ADDRA ~ ADDRH)	668
20.3.2	A/D 控制 / 状态寄存器 (ADCSR)	669
20.4	运行说明	671
20.4.1	单通道模式	671
20.4.2	多通道模式	673
20.4.3	扫描模式	675
20.4.4	由外部触发、MTU2 启动 A/D 转换器	677
20.4.5	输入采样和 A/D 转换时间	677
20.4.6	外部触发输入时序	678
20.5	中断源和 DMAC 传送请求	679
20.6	A/D 转换精度的定义	679
20.7	使用时的注意事项	680
20.7.1	模块待机模式的设定	680
20.7.2	模拟电压的设定	680
20.7.3	电路板设计的注意事项	680
20.7.4	模拟输入引脚的处理	681
20.7.5	容许信号源阻抗	682
20.7.6	对绝对精度的影响	682
20.7.7	深度待机模式时的 A/D 转换	682
20.7.8	使用扫描模式及多通道模式时的注意事项	682
第 21 章	D/A 转换器 (DAC)	683
21.1	特点	683
21.2	输入 / 输出引脚	683
21.3	寄存器说明	684
21.3.1	D/A 数据寄存器 0、1 (DADR0、DADR1)	684
21.3.2	D/A 控制寄存器 (DACR)	684
21.4	运行说明	685
21.5	使用时的注意事项	686
21.5.1	模块待机模式的设定	686
21.5.2	软件待机模式时的 D/A 输出保持功能	686
21.5.3	模拟输入电压的设定	686
21.5.4	深度待机模式时的 D/A 转换	686

第 22 章	AND/NAND 闪存控制器 (FLCTL)	687
22.1	特点	687
22.2	输入 / 输出引脚	690
22.3	寄存器说明	691
22.3.1	共用控制寄存器 (FLCMNCR)	691
22.3.2	命令控制寄存器 (FLCMDRCR)	693
22.3.3	命令码寄存器 (FLCMCDR)	695
22.3.4	地址寄存器 (FLADR)	695
22.3.5	地址寄存器 2 (FLADR2)	696
22.3.6	数据计数器寄存器 (FLDTCNTR)	697
22.3.7	数据寄存器 (FLDATAR)	698
22.3.8	中断 DMA 控制寄存器 (FLINTDMACR)	699
22.3.9	就绪 / 忙超时设定寄存器 (FLBSYTMR)	701
22.3.10	就绪 / 忙超时计数器 (FLBSYCNT)	702
22.3.11	数据 FIFO 寄存器 (FLDTFIFO)	702
22.3.12	管理码 FIFO 寄存器 (FLECFIFO)	703
22.3.13	传送控制寄存器 (FLTRCR)	703
22.4	运行说明	704
22.4.1	存取步骤	704
22.4.2	运行模式	704
22.4.3	寄存器设定步骤	704
22.4.4	命令存取模式	706
22.4.5	扇区存取模式	710
22.4.6	ECC 的纠错	713
22.4.7	状态读取	713
22.5	中断处理	714
22.6	DMA 传送设定	714
第 23 章	USB2.0 主机 / 功能模块 (USB)	715
23.1	特点	715
23.2	输入 / 输出引脚	717
23.3	寄存器说明	718
23.3.1	系统结构控制寄存器 (SYSCFG)	720
23.3.2	系统结构状态寄存器 (SYSSTS)	721
23.3.3	器件状态控制寄存器 (DVSTCTR)	722
23.3.4	测试模式寄存器 (TESTMODE)	724
23.3.5	FIFO 端口结构寄存器 (CFBCFG、D0FBCFG、D1FBCFG)	725
23.3.6	FIFO 端口寄存器 (CFIFO、D0FIFO、D1FIFO)	727
23.3.7	FIFO 端口选择寄存器 (CFIFOSEL、D0FIFOSEL、D1FIFOSEL)	728
23.3.8	FIFO 端口控制寄存器 (CFIFOCTR、D0FIFOCTR、D1FIFOCTR)	731
23.3.9	CFIFO 端口 SIE 寄存器 (CFIFOSIE)	732
23.3.10	处理计数器寄存器 (D0FIFOTRN、D1FIFOTRN)	733
23.3.11	中断允许寄存器 0 (INTENB0)	733
23.3.12	中断允许寄存器 1 (INTENB1)	735
23.3.13	BRDY 中断允许寄存器 (BRDYENB)	736
23.3.14	NRDY 中断允许寄存器 (NRDYENB)	737
23.3.15	BEMP 中断允许寄存器 (BEMPENB)	738
23.3.16	中断状态寄存器 0 (INTSTS0)	739
23.3.17	中断状态寄存器 1 (INTSTS1)	740
23.3.18	BRDY 中断状态寄存器 (BRDYSTS)	742
23.3.19	NRDY 中断状态寄存器 (NRDYSTS)	743
23.3.20	BEMP 中断状态寄存器 (BEMPSTS)	744
23.3.21	帧编号寄存器 (FRMNUM)	745

23.3.22	μ 帧编号寄存器 (UFRMNUM)	746
23.3.23	USB 地址寄存器 (USBADDR)	747
23.3.24	USB 请求类型寄存器 (USBREQ)	747
23.3.25	USB 请求值寄存器 (USBVAL)	748
23.3.26	USB 请求变址寄存器 (USBINDX)	748
23.3.27	USB 请求长度寄存器 (USBLENG)	748
23.3.28	DCP 结构寄存器 (DCPCFG)	749
23.3.29	DCP Max 包大小寄存器 (DCPMAXP)	750
23.3.30	DCP 控制寄存器 (DCPCTR)	750
23.3.31	管道窗口选择寄存器 (PIPESEL)	752
23.3.32	管道结构寄存器 (PIPECFG)	753
23.3.33	管道缓冲器指定寄存器 (PIPEBUF)	755
23.3.34	管道 Max 包大小寄存器 (PEMAXP)	756
23.3.35	管道周期控制寄存器 (PIPEPERI)	757
23.3.36	管道 n 控制寄存器 (PIPE _n CTR) (n=1 ~ 7)	758
23.3.37	USB AC 特性转换寄存器 (USBACSWR)	759
23.4	运行说明	760
23.4.1	系统控制	760
23.4.2	中断功能	761
23.4.3	管道控制	772
23.4.4	缓冲存储器	776
23.4.5	控制传送 (DCP)	786
23.4.6	批量传送 (管道 1 ~ 5)	788
23.4.7	中断传送 (管道 6、7)	790
23.4.8	同步传送 (管道 1、2)	791
23.4.9	SOF 插值功能	796
23.4.10	管道安排	796
23.5	使用时的注意事项	797
23.5.1	使用同步 OUT 传送时的注意事项	797
23.5.2	USB 收发器的设定步骤	798
23.5.3	中断源的清除时序	798
第 24 章	LCD 控制器 (LCDC)	799
24.1	特点	799
24.2	输入 / 输出引脚	800
24.3	寄存器说明	801
24.3.1	LCDC 输入时钟寄存器 (LDICKR)	802
24.3.2	LCDC 模块类型寄存器 (LDMTR)	803
24.3.3	LCDC 数据格式寄存器 (LDDFR)	805
24.3.4	LCDC 扫描模式寄存器 (LDSPMR)	806
24.3.5	LCDC 上部显示面板用取数据起始地址寄存器 (LDSARU)	807
24.3.6	LCDC 下部显示用面板取数据起始地址寄存器 (LDSARL)	808
24.3.7	LCDC 显示面板取数据行地址偏移量寄存器 (LDLAOR)	808
24.3.8	LCDC 调色板控制寄存器 (LDPALCR)	809
24.3.9	调色板数据寄存器 00 ~ FF (LDPR00 ~ LDPRFF)	809
24.3.10	LCDC 水平字符数寄存器 (LDHCNR)	810
24.3.11	LCDC 水平同步信号寄存器 (LDHSYNR)	811
24.3.12	LCDC 垂直显示行数寄存器 (LDVDLNR)	812
24.3.13	LCDC 垂直总行数寄存器 (LDVTLNR)	812
24.3.14	LCDC 垂直同步信号寄存器 (LDVSYNR)	813
24.3.15	LCDC AC 调制信号交替行数寄存器 (LDACLNR)	814
24.3.16	LCDC 中断控制寄存器 (LDINTR)	814
24.3.17	LCDC 电源管理模式寄存器 (LDPMMR)	816

24.3.18	LCDC 上电顺序期间寄存器 (LDPSPR)	817
24.3.19	LCDC 控制寄存器 (LDCNTR)	818
24.3.20	LCDC 用户指定中断控制寄存器 (LDUINTR)	819
24.3.21	LCDC 用户指定中断行数寄存器 (LDUINTLNR)	820
24.3.22	LCDC 存储器存取间隔数寄存器 (LDLIRNR)	821
24.4	运行说明	822
24.4.1	关于 LCDC 可显示的液晶模块尺寸	822
24.4.2	旋转显示的分辨率 / 突发长度及连接存储器 (SDRAM) 的限制事项	823
24.4.3	彩色调色板规格	827
24.4.4	数据格式	828
24.4.5	显示分辨率的设定	831
24.4.6	电源控制顺序处理	831
24.4.7	硬件旋转运行说明	835
24.5	时钟和 LCD 数据信号例	836
24.6	使用时的注意事项	845
24.6.1	存取保存显示数据的 VRAM (区域 3 的同步 DRAM) 的停止步骤	845
第 25 章	引脚功能控制器 (PFC)	846
25.1	特点	851
25.2	寄存器的说明	851
25.2.1	端口 B/IO 寄存器 L (PBIORL)	852
25.2.2	端口 B 控制寄存器 L1 ~ L4 (PBCRL1 ~ PBCRL4)	852
25.2.3	端口 C/IO 寄存器 L (PCIORL)	856
25.2.4	端口 C 控制寄存器 L1 ~ L4 (PCCRL1 ~ PCCRL4)	856
25.2.5	端口 D/IO 寄存器 L (PDIORL)	860
25.2.6	端口 D 控制寄存器 L1 ~ L4 (PDCRL1 ~ PDCRL4)	860
25.2.7	端口 E/IO 寄存器 L (PEIORL)	867
25.2.8	端口 E 控制寄存器 L1 ~ L4 (PECRL1 ~ PECRL4)	868
25.2.9	端口 F/IO 寄存器 H、L (PFIORH、PFIORL)	872
25.2.10	端口 F 控制寄存器 H1 ~ H4、L1 ~ L4 (PFCRH1 ~ PFCRH4、PFCRL1 ~ PFCRL4)	872
25.2.11	IRQOUT 功能控制寄存器 (IFCR)	880
25.2.12	SSI 过采样时钟选择寄存器 (SCSR)	881
25.3	有关端口 A 的功能转换	882
25.4	使用时的注意事项	882
第 26 章	I/O 端口	883
26.1	特点	883
26.2	端口 A	883
26.2.1	寄存器说明	884
26.2.2	端口 A 数据寄存器 L (PADRL)	884
26.3	端口 B	885
26.3.1	寄存器说明	885
26.3.2	端口 B 数据寄存器 L (PBDRL)	885
26.3.3	端口 B 端口寄存器 L (PBPRRL)	887
26.4	端口 C	887
26.4.1	寄存器说明	888
26.4.2	端口 C 数据寄存器 L (PCDRL)	888
26.4.3	端口 C 端口寄存器 L (PCPRL)	889
26.5	端口 D	890
26.5.1	寄存器说明	890
26.5.2	端口 D 数据寄存器 L (PDDRL)	891
26.5.3	端口 D 端口寄存器 L (PDPRL)	892
26.6	端口 E	893

26.6.1	寄存器说明	893
26.6.2	端口 E 数据寄存器 L (PEDRL)	894
26.6.3	端口 E 端口寄存器 L (PEPRL)	895
26.7	端口 F.....	896
26.7.1	寄存器说明	896
26.7.2	端口 F 数据寄存器 H、L (PFDRH、PFDRL)	897
26.7.3	端口 F 端口寄存器 H、L (PFPRH、PFPRL)	899
26.8	使用时的注意事项	901
第 27 章	内部 RAM	902
27.1	特点	902
27.2	使用时的注意事项	903
27.2.1	页竞争	903
27.2.2	关于 RAME 位、RAMWE 位	903
27.2.3	指令配置禁止区域	903
第 28 章	低功耗模式	904
28.1	特点	904
28.1.1	低功耗模式的种类	904
28.2	寄存器说明	905
28.2.1	待机控制寄存器 (STBCR)	905
28.2.2	待机控制寄存器 2 (STBCR2)	906
28.2.3	待机控制寄存器 3 (STBCR3)	907
28.2.4	待机控制寄存器 4 (STBCR4)	908
28.2.5	待机控制寄存器 5 (STBCR5)	909
28.2.6	待机控制寄存器 6 (STBCR6)	910
28.2.7	系统控制寄存器 1 (SYSCR1)	911
28.2.8	系统控制寄存器 2 (SYSCR2)	912
28.2.9	系统控制寄存器 3 (SYSCR3)	913
28.2.10	深度待机控制寄存器 (DSCTR)	914
28.2.11	深度待机控制寄存器 2 (DSCTR2)	915
28.2.12	深度待机解除源选择寄存器 (DSSSR)	916
28.2.13	深度待机解除源标志寄存器 (DSFR)	917
28.2.14	保持用内部 RAM 调整寄存器 (DSRTR)	918
28.3	运行说明	919
28.3.1	睡眠模式	919
28.3.2	软件待机模式	919
28.3.3	软件待机模式的应用例	921
28.3.4	深度待机模式	922
28.3.5	模块待机功能	925
28.4	使用时的注意事项	926
28.4.1	写入寄存器时的注意事项	926
28.4.2	有关深度待机控制寄存器 2 (DSCTR2) 的注意事项	926
28.4.3	上电复位异常处理相关注意事项	926
第 29 章	用户调试接口 (H-UDI)	927
29.1	特点	927
29.2	输入 / 输出引脚	928
29.3	寄存器说明	928
29.3.1	旁路寄存器 (SDBPR)	929
29.3.2	指令寄存器 (SDIR)	929
29.4	运行说明	930
29.4.1	TAP 控制器	930

29.4.2	复位结构	931
29.4.3	TDO 输出时序	931
29.4.4	H-UDI 复位	932
29.4.5	H-UDI 中断	932
29.5	使用时的注意事项	932
第 30 章	寄存器一览表	933
30.1	寄存器地址一览表（按各功能模块、手册章节编号顺序）	934
30.2	寄存器位一览表	953
30.3	各运行模式的寄存器状态一览表	993
第 31 章	电特性	995
31.1	绝对最大额定值	995
31.2	上电、断电顺序	996
31.3	DC 特性	997
31.4	AC 特性	1002
31.4.1	时钟时序	1002
31.4.2	控制信号时序	1005
31.4.3	总线时序	1007
31.4.4	UBC 时序	1041
31.4.5	DMAC 时序	1041
31.4.6	MTU2 时序	1042
31.4.7	WDT 时序	1043
31.4.8	SCIF 时序	1043
31.4.9	SSU 时序	1044
31.4.10	IIC3 时序	1047
31.4.11	SSI 时序	1048
31.4.12	RCAN-TL1 时序	1050
31.4.13	ADC 时序	1050
31.4.14	FLCTL 时序	1051
31.4.15	USB 时序	1058
31.4.16	LCDC 时序	1060
31.4.17	I/O 端口时序	1061
31.4.18	H-UDI 时序	1061
31.4.19	AC 特性测量条件	1062
31.5	A/D 转换器特性	1063
31.6	D/A 转换器特性	1063
31.7	使用时的注意事项	1064
附录		1065
附录 A.	引脚状态	1065
附录 B.	封装尺寸图	1070
索引		1071

第 1 章 概要

1.1 SH7203 的特点

本 LSI 以瑞萨科技独创的 RISC（精简指令系统计算机）结构 CPU 为内核，是集成了系统构成所需外围功能的 RISC 单片机。

本 LSI 采用与 SH-1、SH-2、SH-2E 单片机在目标代码级高位兼容的 SH-2A CPU，由于具备 RISC 结构的指令系统，并采用超标量 / 哈佛体系结构，所以指令执行速度得以飞速提升。另外，采用与直接存储器存取控制器（DMAC）独立的内部 32 位总线结构，强化了数据处理能力。本 LSI 的 CPU 可构成以往单片机不可能实现的、需高速实时控制的应用程序等低成本、高性能 / 高功能的系统。

本 LSI 内置浮点单元（FPU）与高速缓存。此外，还内置 64KB 高速内部 RAM、用于保存 16KB 数据的 RAM、多功能定时器脉冲单元 2（MTU2）、比较匹配定时器（CMT）、实时时钟（RTC）、带 FIFO 的串行通信接口（SCIF）、同步串行通信单元（SSU）、I²C 总线接口 3（IIC3）、串行音频接口（SSI）、控制器局域网（RCAN-TL1）、A/D 转换器、D/A 转换器、AND/NAND 闪存控制器（FLCTL）、USB2.0 主机 / 功能模块（USB）、中断控制器（INTC）及 I/O 端口等，作为系统构成所需的外围功能。

本 LSI 支持外部存储器存取功能，可与存储器或外围 LSI 直接连接。所以，可大幅降低系统成本。另外，本 LSI 的 I/O 引脚带有抑制引脚电位变为中间电位状态的弱保持电路，无需固定输入电平的外接电路，也可大幅度减少元器件个数。

本 LSI 的特点如表 1.1 所示。

表 1.1 SH7203 的特点

项目	特 点
CPU	- 瑞萨独创的 SuperH 体系结构 - 与 SH-1、SH-2、SH-2E 在目标代码级高位兼容 32 位内部数据总线 - 通用寄存器体系结构 16 个 32 位通用寄存器 4 个 32 位控制寄存器 4 个 32 位系统寄存器 - 响应高速中断的寄存器存储体 - RISC 结构的指令系统（与 SH 系列高位兼容） - 指令长度：提高编码效率的 16 位基本指令与提高性能 / 适用性的 32 位指令 - 加载存储结构 - 延迟转移指令 - 基于 C 语言的指令系统 - 包含 FPU 的 2 条指令同时执行型超标量 - 指令执行时间：最多 2 条指令 / 周期 - 地址空间：4G 字节 - 内置乘法器 5 段流水线 - 哈佛体系结构
浮点单元（FPU）	- 内置浮点协处理器 - 支持单精度（32 位）及双精度（64 位） - 支持符合 IEEE754 标准的数据类型及异常 - 舍入模式：就近舍入及向 0 舍入 - 非规格化数的处理：舍入为 0 - 浮点寄存器 16 个 32 位浮点寄存器（单精度 ×16 字或双精度 ×8 字） 2 个 32 位浮点系统寄存器 - 支持 FMAC（乘法及累加）指令 - 支持 FDIV（除法）/FSQRT（平方根）指令 - 支持 FLDI0/FLDI1（加载常数 0/1）指令 - 指令执行时间 - 等待时间（FMAC/FADD/FSUB/FMUL）：3 个周期（单精度）、8 个周期（双精度） - 节距（FMAC/FADD/FSUB/FMUL）：1 个周期（单精度）、6 个周期（双精度） 【注】FMAC 仅支持单精度。 5 段流水线

项目	特 点
高速缓冲存储器	- 指令高速缓存: 8K 字节 - 操作数高速缓存: 8K 字节 - 各有 128 个入口 / 通路、4 路集联、16 字节的块长度 - 回写、直写、LRU 置换算法 - 具有通路锁定功能 (仅限操作数高速缓存): 可锁定通路 2、通路 3
中断控制器 (INTC)	17 个外部中断引脚 (NMI、IRQ7 ~ IRQ0、PINT7 ~ PINT0) - 内部外围中断: 对各模块设定优先顺序 - 可设定 16 级优先顺序 - 通过寄存器存储体, 可在中断处理的同时高速保存 / 返回寄存器
总线状态控制器 (BSC)	- 地址空间分别支持 8 个最大 64MB 的区域 (区域 0 ~ 7) - 各区域可独立设定以下功能: 总线宽度 (8、16、32 位)。各区域支持的宽度不同。 存取等待周期数 (有可对读取 / 写入设定独立等待周期的区域) 设定空闲等待周期 (相同区域 / 不同区域) 按区域指定连接的存储器, 可直接连接 SRAM、带字节选择的 SRAM、SDRAM 及突发 ROM (时钟同步 / 时钟异步)。另外, 支持地址 / 数据多路复用 I/O (MPX) 接口及突发 MPX-I/O 接口 支持 PCMCIA 接口 对相应区域输出片选信号 ($\overline{CS0} \sim \overline{CS7}$) (可由编程选择 CS 有效 / 无效时序) - SDRAM 刷新功能 支持自动刷新及自刷新模式 - SDRAM 突发存取功能
直接存储器存取控制器 (DMAC)	8 个通道。其中 4 个通道可用于外部请求 - 可从内部外围模块启动 - 突发模式及周期挪用模式 - 支持间歇模式 (支持 16/64 个周期) - 可自动重加载传送信息
时钟脉冲振荡器 (CPG)	- 时钟模式: 可从外部输入 (EXTAL、CKIO 或 USB_X1) 或晶体谐振器选择输入时钟 - 通过内部 PLL 电路, 最大可将输入时钟 16 倍频 - 生成 3 种时钟 CPU 时钟: 最大 200MHz 总线时钟: 最大 66MHz 外围时钟: 最大 33MHz
看门狗定时器 (WDT)	1 个通道的看门狗定时器 - 通过计数器上溢可复位本 LSI
低功耗模式	支持可降低本 LSI 功耗的 4 种低功耗模式 睡眠模式 软件待机模式 深度待机模式 模块待机模式

项目	特 点
多功能定时器脉冲单元 2 (MTU2)	- 以 5 个通道的 16 位定时器为基础，最多可输入 / 输出 16 种脉冲 18 个输出比较寄存器兼输入捕捉寄存器 - 输入捕捉功能 - 脉冲输出模式 交替 / PWM / 互补 PWM / 复位同步 PWM - 多个计数器的同步功能 - 互补 PWM 输出模式 输出控制 3 相变频器的非重叠波形 自动设定空载时间 可在 0 ~ 100% 内任意设定 PWM 占空比 A/D 转换请求延迟功能 波峰 / 波谷中断跳过功能 - 复位同步 PWM 模式 3 相输出任意占空比的正相 / 反相 PWM 波形 - 相位计数模式 可执行 2 相编码器计数处理
比较匹配定时器 (CMT)	2 个通道的 16 位计数器 - 可选择 4 种时钟 (Pφ/8、Pφ/32、Pφ/128、Pφ/512) - 比较匹配时，可选择产生 DMA 传送请求或中断请求
实时时钟 (RTC)	- 内部时钟、日历功能、报警功能 1/256 秒最大分辨率 (中断周期) 的内部 32.768kHz 晶体振荡器电路
带 FIFO 的串行通信接口 (SCIF)	4 个通道 - 可选择时钟同步 / 异步模式 - 可同时发送 / 接收 (全双工) - 内置专用的波特率发生器 - 内置分别用于发送 / 接收的 16 字节 FIFO - 调制解调器控制功能 (异步模式时)
同步串行通信单元 (SSU)	- 可选择主模式与从属模式 - 可选择标准模式 / 双向模式 - 可选择发送 / 接收数据长度为 8/16/32 位 - 可同时发送 / 接收 (全双工) - 可连续串行通信 2 个通道
I ² C 总线接口 3 (IIC3)	4 个通道 - 内置主模式 / 从属模式
串行音频接口 (SSI)	4 个通道的双向串行传送 - 支持多种串行音频格式 - 支持主 / 从属功能 - 生成可编程字时钟、位时钟的功能 - 多通道格式功能 - 支持 8/16/18/20/22/24/32 位数据格式

项目	特 点
控制器局域网 (RCAN-TL1)	• 2 个通道 • 所有通道均支持 TTCAN 级 1 • 支持 Bosch 2.0B active • 缓冲器大小：发送 / 接收 × 31、仅接收 × 1 • 为了增加 32 个通道输入的缓冲器，可给 1 个总线分配多个 RCAN 通道 • 设定 31 个用于发送或接收的邮箱
AND/NAND 闪存控制器 (FLCTL)	• 与 AND/NAND 型闪存直接连接的存储器接口 • 扇区单位的读取 / 写入 • 命令存取模式与扇区存取模式（512 字节数据 + 16 字节管理码：带 ECC）等 2 种传送模式 • 有中断请求、DMAC 传送请求 • 支持 5 字节地址（超过 2G 位）的闪存
USB 2.0 主机 / 功能模块 (USB)	• 以 USB 2.0 版本为依据 • 支持 480Mbps 及 12Mbps 的传送速率 • 也可用作功能模块 • 可由软件设定 • 内置用作通信缓冲器的 8K 字节 RAM
LCD 控制器 (LCDC)	• 支持 16 × 1 ~ 1,024 × 1,024 点 • 支持 4、8、15、16bpp 色彩模式 • 支持 1、2、4、6bpp 灰度模式 • 支持 TFT/DSTN/STN 显示器 • 可设定信号极性 • 24 位调色板存储器（24 位中，作为 R:5/G:6/B:5 的 16 位有效） • 统一图形存储器体系结构
I/O 端口	• 82 个（输入 / 输出）、16 个（输入）、1 个（输出） • 输入 / 输出端口可按位转换输入 / 输出 • 内置弱保持电路
A/D 转换器 (ADC)	• 分辨率：10 位 • 输入：8 个通道 • 可由外部触发器 / 定时器触发器启动 A/D 转换
D/A 转换器 (DAC)	• 分辨率：8 位 • 输出：2 个通道
用户断点控制器 (UBC)	• 断点通道 × 2 个通道 • 可设定地址、数据值、存取类型及数据长度为断点条件
用户调试接口 (H-UDI)	• 支持 E10A 仿真器 • JTAG 标准引脚配置
内部 RAM	• 内置用于高速存取的 64K 字节存储器（16KB × 4 面） • 内置用于保持数据的 16K 字节存储器（4KB × 4 面）
电源电压	• V_{CC}: 1.1 ~ 1.3V、PV_{CC}: 3.0 ~ 3.6V
封装	• QFP3232-240Cu（0.5 节距）

1.2 产品一览表

表 1.2 产品一览表

产品分类	产品型号	封装
SH7203	R5S72030W200FP	QFP3232-240Cu

1.3 框图

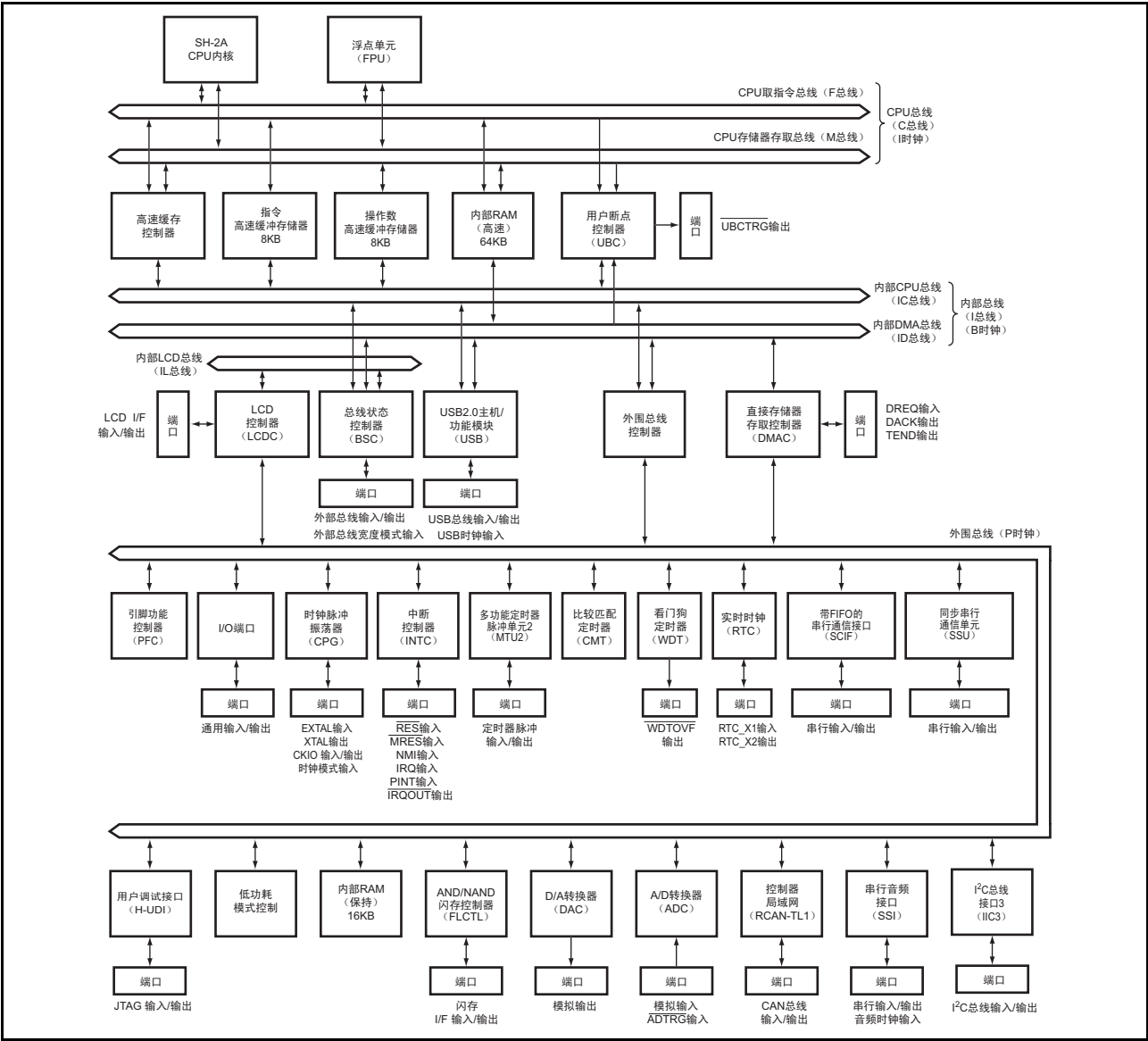


图 1.1 框图

1.4 引脚排列图

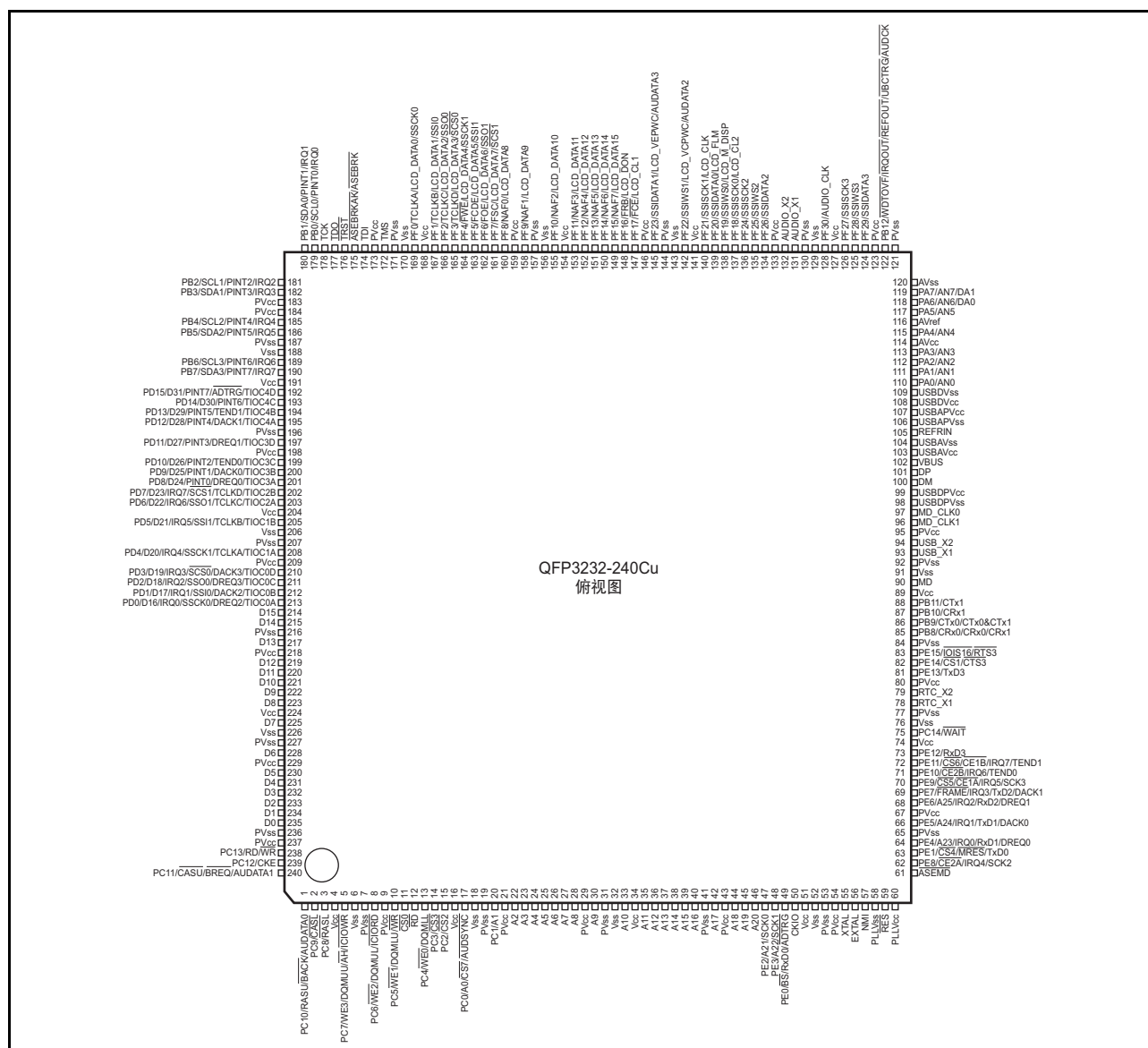


图 1.2 引脚排列图

1.5 引脚功能

表 1.3 引脚功能

分类	引脚名称	输入 / 输出	名称	功 能
电源	V _{CC}	输入	电源	电源引脚。必须将所有 V _{CC} 引脚连接至系统电源。如果有开路引脚，则不运行。
	V _{SS}	输入	接地	接地引脚。必须将所有 V _{SS} 引脚连接至系统电源（0V）。如果有开路引脚，则不运行。
	PV _{CC}	输入	输入 / 输出电路的电源	输入 / 输出引脚的电源引脚。必须将所有 PV _{CC} 引脚连接至系统电源。如果有开路引脚，则不运行。
	PV _{SS}	输入	输入 / 输出电路的接地	输入 / 输出引脚的接地引脚。必须将所有 PV _{SS} 引脚连接至系统电源（0V）。如果有开路引脚，则不运行。
	PLL _{VCC}	输入	PLL 的电源	内部 PLL 振荡器的电源引脚。
	PLL _{VSS}	输入	PLL 的接地	内部 PLL 振荡器的接地引脚。
时钟	EXTAL	输入	外部时钟	连接晶体谐振器。EXTAL 引脚可输入外部时钟。
	XTAL	输出	晶体	连接晶体谐振器。
	CKIO	输入 / 输出	输入 / 输出系统时钟	对外部时钟输入引脚或外部器件提供系统时钟。
运行模式控制	MD	输入	模式设定	设定运行模式。不得在运行过程中更改该引脚。
	MD_CLK1、MD_CLK0	输入	时钟模式设定	设定时钟运行模式。不得在运行过程中更改这些引脚。
	ASEMD	输入	ASE 模式	如果在 RES 引脚有效期间输入低电平，则变为 ASE 模式；如果输入高电平，则变为产品芯片模式。ASE 模式时，E10A-USB 仿真器功能有效。不使用仿真器功能时，必须固定为高电平。
系统控制	RES	输入	上电复位	该引脚为低电平时，为上电复位状态。
	MRES	输入	手动复位	该引脚为低电平时，为手动复位状态。
	WDTOVF	输出	看门狗定时器上溢	WDT 的上溢输出信号。
	BREQ	输入	总线权请求	外部器件请求释放总线权时，设置为低电平。
	BACK	输出	总线权请求应答	表示已将总线权释放给外部器件。可得知输出 BREQ 信号的器件已接受 BACK 信号，并获得总线权。
中断	NMI	输入	非屏蔽中断	非屏蔽的中断请求引脚。不使用时，必须固定为高电平。
	IRQ7 ~ RQ0	输入	中断请求 7 ~ 0	可屏蔽的中断请求引脚。 可选择电平输入或边沿输入。边沿输入时，可选择上升沿、下降沿或双边沿。
	PINT7 ~ PINT0	输入	中断请求 7 ~ 0	可屏蔽的中断请求引脚。 仅可选择电平输入。
	IRQOUT	输出	中断请求输出	表示已产生中断源。也可在总线权释放期间得知产生中断。
地址总线	A25 ~ A0	输出	地址总线	输出地址。
数据总线	D31 ~ D0	输入 / 输出	数据总线	双向数据总线。

分类	引脚名称	输入 / 输出	名称	功 能
总线控制	$\overline{CS7} \sim \overline{CS0}$	输出	片选 7 ~ 0	外部存储器或器件的片选信号。
	$\overline{RD}$	输出	读取	表示从外部器件读取。
	$\overline{RD}/\overline{WR}$	输出	读取 / 写入	读取信号或写入信号。
	$\overline{BS}$	输出	总线开始	总线周期开始信号。
	$\overline{AH}$	输出	地址保持	保持地址或数据多路复用 I/O 时地址的信号。
	$\overline{FRAME}$	输出	$\overline{FRAME}$ 信号	突发 MPX-I/O 接口时, 连接 $\overline{FRAME}$ 信号
	$\overline{WAIT}$	输入	等待	在存取外部空间的总线周期可插入等待周期的输入。
	$\overline{WE0}$	输出	字节指定	表示对外部存储器或器件数据的 bit7 ~ 0 写入。
	$\overline{WE1}$	输出	字节指定	表示对外部存储器或器件数据的 bit15 ~ 8 写入。
	$\overline{WE2}$	输出	字节指定	表示对外部存储器或器件数据的 bit23 ~ 16 写入。
	$\overline{WE3}$	输出	字节指定	表示对外部存储器或器件数据的 bit31 ~ 24 写入。
	$\overline{DQMLL}$	输出	字节指定	连接 SDRAM 时, 选择 D7 ~ D0。
	$\overline{DQMLU}$	输出	字节指定	连接 SDRAM 时, 选择 D15 ~ D8。
	$\overline{DQMUL}$	输出	字节指定	连接 SDRAM 时, 选择 D23 ~ D16。
	$\overline{DQMUU}$	输出	字节指定	连接 SDRAM 时, 选择 D31 ~ D24。
	$\overline{RASU}$ 、 $\overline{RASL}$	输出	RAS	连接 SDRAM 时, 连接至 $\overline{RAS}$ 引脚。
	$\overline{CASU}$ 、 $\overline{CASL}$	输出	CAS	连接 SDRAM 时, 连接至 $\overline{CAS}$ 引脚。
	$\overline{CKE}$	输出	CK 允许	连接 SDRAM 时, 连接至 $\overline{CKE}$ 引脚。
	$\overline{CE1A}$ 、 $\overline{CE1B}$	输出	PCMCIA 卡选低位	连接至 PCMCIA 卡选信号 D7 ~ D0。
	$\overline{CE2A}$ 、 $\overline{CE2B}$	输出	PCMCIA 卡选高位	连接至 PCMCIA 卡选信号 D15 ~ D8。
	$\overline{ICIORW}$	输出	PCMCIA 写入选通	连接 PCMCIA I/O 写入选通。
	$\overline{ICIOR}$	输出	PCMCIA 读取选通	连接 PCMCIA I/O 读取选通。
	$\overline{WE}$	输出	PCMCIA 存储器写入选通	连接 PCMCIA 存储器写入选通。
	$\overline{IOIS16}$	输入	PCMCIA 动态总线宽度更改 (dynamic bus sizing)	表示 PCMCIA 的 16 位 I/O。
	$\overline{REFOUT}$	输出	刷新请求	执行刷新请求的信号。
直接存储器存取控制器 (DMAC)	$\overline{DREQ0} \sim \overline{DREQ3}$	输入	DMA 传送请求	外部 DMA 传送请求的输入引脚。
	$\overline{DACK0} \sim \overline{DACK3}$	输出	接受 DMA 传送请求	接受外部 DMA 传送请求的输出引脚。
	$\overline{TEND0}$ 、 $\overline{TEND1}$	输出	DMA 传送结束输出	DMA 传送结束输出信号。

分类	引脚名称	输入 / 输出	名称	功 能
多功能定时器 脉冲单元 2 (MTU2)	TCLKA、 TCLKB、 TCLKC、 TCLKD	输入	MTU2 定时器时钟输入	定时器的外部时钟输入引脚。
	TIOC0A、 TIOC0B、 TIOC0C、 TIOC0D	输入 / 输出	MTU2 输入捕捉 / 输出 比较 (通道 0)	TGRA_0 ~ TGRD_0 的输入捕捉输入 / 输出比较 输出 / PWM 输出引脚。
	TIOC1A、 TIOC1B	输入 / 输出	MTU2 输入捕捉 / 输出 比较 (通道 1)	TGRA_1、TGRB_1 的输入捕捉输入 / 输出比较输 出 / PWM 输出引脚。
	TIOC2A、 TIOC2B	输入 / 输出	MTU2 输入捕捉 / 输出 比较 (通道 2)	TGRA_2、TGRB_2 的输入捕捉输入 / 输出比较输 出 / PWM 输出引脚。
	TIOC3A、 TIOC3B、 TIOC3C、 TIOC3D	输入 / 输出	MTU2 输入捕捉 / 输出 比较 (通道 3)	TGRA_3 ~ TGRD_3 的输入捕捉输入 / 输出比较 输出 / PWM 输出引脚。
	TIOC4A、 TIOC4B、 TIOC4C、 TIOC4D	输入 / 输出	MTU2 输入捕捉 / 输出 比较 (通道 4)	TGRA_4、TGRB_4 的输入捕捉输入 / 输出比较输 出 / PWM 输出引脚。
实时时钟 (RTC)	RTC_X1	输入	RTC 的晶体谐振器 / 外 部时钟	连接 32.768kHz 的晶体谐振器。另外，RTC_X1 引脚也可输入外部时钟。
	RTC_X2	输出		
带 FIFO 的串 行通信接口 (SCIF)	TxD3 ~ TxD0	输出	发送数据	数据输出引脚。
	RxD3 ~ RxD0	输入	接收数据	数据输入引脚。
	SCK3 ~ SCK0	输入 / 输出	串行时钟	时钟输入 / 输出引脚。
	RTS3	输出	发送请求	调制解调器控制引脚。
	CTS3	输入	可发送	调制解调器控制引脚。
同步串行通信 单元 (SSU)	SSO1、SSO0	输入 / 输出	数据	数据输入 / 输出引脚。
	SSI1、SSI0	输入 / 输出	数据	数据输入 / 输出引脚。
	SSCK1、 SSCK0	输入 / 输出	时钟	时钟输入 / 输出引脚。
	SCS1、SCS0	输入 / 输出	片选	片选输入 / 输出引脚。
I ² C 总线接口 3 (IIC3)	SCL3 ~ SCL0	输入 / 输出	串行时钟引脚	串行时钟输入 / 输出引脚。
	SDA3 ~ SDA0	输入 / 输出	串行数据引脚	串行数据输入 / 输出引脚。

分类	引脚名称	输入 / 输出	名称	功 能
串行音频接口 (SSI)	SSIDATA3 ~ SSIDATA0	输入 / 输出	SSI 数据输入 / 输出	串行数据输入 / 输出引脚。
	SSISCK3 ~ SSISCK0	输入 / 输出	SSI 时钟输入 / 输出	串行时钟输入 / 输出引脚。
	SSIWS3 ~ SSIWS0	输入 / 输出	SSI 时钟 LR 输入 / 输出	字选择输入 / 输出引脚。
	AUDIO_CLK	输入	SSI 音频的外部时钟	输入音频的外部时钟。 选择作为 AUDIO_X1、AUDIO_X2 的振荡时钟， 并输入分频器。
	AUDIO_X1	输入	SSI 音频的晶体谐振器 / 外部时钟	连接音频的晶体谐振器。另外，AUDIO_X1 引脚 也可输入外部时钟。选择作为 AUDIO_CLK 输入 时钟，并输入分频器
	AUDIO_X2	输出		
控制器局域网 (RCAN-TL1)	CTx0, CTx1	输出	CAN 总线发送数据	CAN 总线的发送引脚。
	CRx0, CRx1	输入	CAN 总线接收数据	CAN 总线的接收引脚。
AND/NAND 闪存控制器 (FLCTL)	FOE	输出	闪存输出允许	地址锁存允许：输出地址时有效；输入 / 输出数 据时无效。 输出允许：输入数据 / 读取状态时有效。
	FSC	输出	闪存串行时钟	读取允许：在下降沿读取数据。 串行时钟：与信号同步输入 / 输出数据。
	FCE	输出	闪存芯片允许	芯片允许：允许本 LSI 连接的闪存。
	FCDE	输出	闪存命令数据允许	命令锁存允许：输出命令时有效。 命令数据允许：输出命令时有效。
	FRB	输入	闪存就绪 / 忙	就绪 / 忙：高电平时表示就绪状态；低电平时表 示忙状态。
	FWE	输出	闪存写入允许	写入允许：在下降沿闪存锁存命令、地址及数 据。
	NAF7 ~ NAF0	输入 / 输出	闪存数据	数据输入 / 输出引脚。
USB2.0 主机 / 功能模块 (USB)	DP	输入 / 输出	USB D + 数据	USB 总线的 D + 数据。
	DM	输入 / 输出	USB D - 数据	USB 总线的 D - 数据。
	VBUS	输入	VBUS 输入	必须连接至 USB 总线的 Vbus。
	REFRIN	输入	基准输入	必须通过 5.6kΩ±1% 的电阻连接至 USBAPVss。
	USB_X1	输入	USB 的晶体谐振器 / 外部时钟	连接 USB 的晶体谐振器。另外，USB_X1 引脚 也可输入外部时钟。
	USB_X2	输出		
	USBAPVcc	输入	收发器部模拟引脚电源	引脚的电源。
	USBAPVss	输入	收发器部模拟引脚接地	引脚的接地。
	USBDPVcc	输入	收发器部数字引脚电源	引脚的电源。
	USBDPVss	输入	收发器部数字引脚接地	引脚的接地。
	USBAVcc	输入	收发器部模拟内核电源	内核的电源。
	USBAVss	输入	收发器部模拟内核接地	内核的接地。
	USBDVcc	输入	收发器部数字内核电源	内核的电源。
	USBDVss	输入	收发器部数字内核接地	内核的接地。

分类	引脚名称	输入 / 输出	名称	功 能
LCD 控制器 (LCDC)	LCD_DATA15 ~ LCD_DATA0	输出	LCD 数据	LCD 面板的数据输出引脚。
	LCD_CL1	输出	移位时钟	LCD 移位时钟 1/ 水平同步信号引脚。
	LCD_CL2	输出	移位时钟	LCD 移位时钟 2/ 点时钟引脚。
	LCD_CLK	输入	时钟源	LCD 时钟源输入引脚。
	LCD_FLM	输出	行标记	起始行标记 / 垂直同步信号引脚。
	LCD_DON	输出	LCD 显示开始	LCD 显示开始信号引脚。
	LCD_VCPWC	输出	电源控制	LCD 液晶模块电源控制 (VCC) 引脚。
	LCD_VEPWC	输出	电源控制 (VEE)	LCD 液晶模块电源控制 (VEE) 引脚。
	LCD_M_DISP	输出	液晶交流	液晶交流信号引脚。
A/D 转换器 (ADC)	AN7 ~ AN0	输入	模拟输入引脚	模拟输入引脚。
	ADTRG	输入	A/D 转换触发输入	开始 A/D 转换的外部触发输入引脚。
D/A 转换器 (DAC)	DA1、DA0	输出	模拟输出引脚	模拟输出引脚。
模拟通用	AV _{CC}	输入	模拟电源	A/D 转换器及 D/A 转换器的电源引脚。
	AV _{SS}	输入	模拟接地	A/D 转换器及 D/A 转换器的接地引脚。
	AV _{ref}	输入	模拟基准电压	A/D 转换器及 D/A 转换器的基准电压引脚。
I/O 端口	PB11 ~ PB8、 PC14 ~ PC0、 PD15 ~ PD0、 PE15 ~ PE0、 PF30 ~ PF0	输入 / 输出	通用端口	82 位通用输入 / 输出端口引脚。
	PA7 ~ PA0、 PB7 ~ PB0	输入	通用端口	16 位通用输入端口引脚。
	PB12	输出	通用端口	1 位通用输出端口引脚。
用户调试接口 (H-UDI)	TCK	输入	测试时钟	测试时钟输入引脚。
	TMS	输入	测试模式选择	测试模式选择信号输入引脚。
	TDI	输入	测试数据输入	指令与数据的串行输入引脚。
	TDO	输出	测试数据输出	指令与数据的串行输出引脚。
	TRST	输入	测试复位	初始化信号输入引脚。
仿真器接口	AUDATA3 ~ AUDATA0	输出	AUD 数据	转移目标 / 转移源地址输出引脚。
	AUDCK	输出	AUD 时钟	同步时钟输出引脚。
	AUDSYNC	输出	AUD 同步信号	数据起始位置识别信号输出引脚。
	ASEBRKAK	输出	断点模式应答	表示 E10A-USB 仿真器进入断点模式。
	ASEBRK	输入	断点请求	E10A-USB 仿真器断点输入。
用户断点控制 器 (UBC)	UBCTRIG	输出	用户断点触发输出	UBC 条件匹配的触发输出。

1.6 引脚一览表

表 1.4 引脚一览表

引脚编号	功能 1		功能 2		功能 3	
	引脚名称	I/O	引脚名称	I/O	引脚名称	I/O
1	PC10	I/O	RASU	O	BACK	O
2	PC9	I/O	CASL	O	—	—
3	PC8	I/O	RASL	O	—	—
4	V _{CC}					
5	PC7	I/O	WE3/DQM _{UU} /AH/ ICIOWR	O	—	—
6	V _{SS}					
7	PV _{SS}					
8	PC6	I/O	WE2/DQM _L /ICIORD	O	—	—
9	PV _{CC}					
10	PC5	I/O	WE1/DQM _{LU} /WR	O	—	—
11	CS0	O	—	—	—	—
12	RD	O	—	—	—	—
13	PC4	I/O	WE0/DQM _{LL}	O	—	—
14	PC3	I/O	CS3	O	—	—
15	PC2	I/O	CS2	O	—	—

引脚编号	功能 4		功能 5		功能 6		弱保持	上拉
	引脚名称	I/O	引脚名称	I/O	引脚名称	I/O		
1	AUDATA0	O	—	—	—	—	○	
2	—	—	—	—	—	—	○	
3	—	—	—	—	—	—	○	
4								
5	—	—	—	—	—	—	○	
6								
7								
8	—	—	—	—	—	—	○	
9								
10	—	—	—	—	—	—	○	
11	—	—	—	—	—	—	○	
12	—	—	—	—	—	—	○	
13	—	—	—	—	—	—	○	
14	—	—	—	—	—	—	○	
15	—	—	—	—	—	—	○	

引脚编号	功能 1		功能 2		功能 3	
	引脚名称	I/O	引脚名称	I/O	引脚名称	I/O
16	V _{CC}					
17	PC0	I/O	A0	O	$\overline{\text{CS7}}$	O
18	V _{SS}					
19	PV _{SS}					
20	PC1	I/O	A1	O	—	—
21	PV _{CC}					
22	A2	O	—	—	—	—
23	A3	O	—	—	—	—
24	A4	O	—	—	—	—
25	A5	O	—	—	—	—
26	A6	O	—	—	—	—
27	A7	O	—	—	—	—
28	A8	O	—	—	—	—
29	PV _{CC}					
30	A9	O	—	—	—	—
31	PV _{SS}					
32	V _{SS}					

引脚编号	功能 4		功能 5		功能 6		弱保持	上拉
	引脚名称	I/O	引脚名称	I/O	引脚名称	I/O		
16								
17	AUDSYNC	O	—	—	—	—	○	
18								
19								
20	—	—	—	—	—	—	○	
21								
22	—	—	—	—	—	—	○	
23	—	—	—	—	—	—	○	
24	—	—	—	—	—	—	○	
25	—	—	—	—	—	—	○	
26	—	—	—	—	—	—	○	
27	—	—	—	—	—	—	○	
28	—	—	—	—	—	—	○	
29								
30	—	—	—	—	—	—	○	
31								
32								

引脚编号	功能 1		功能 2		功能 3	
	引脚名称	I/O	引脚名称	I/O	引脚名称	I/O
33	A10	O	—	—	—	—
34	V _{CC}					
35	A11	O	—	—	—	—
36	A12	O	—	—	—	—
37	A13	O	—	—	—	—
38	A14	O	—	—	—	—
39	A15	O	—	—	—	—
40	A16	O	—	—	—	—
41	PV _{SS}					
42	A17	O	—	—	—	—
43	PV _{CC}					
44	A18	O	—	—	—	—
45	A19	O	—	—	—	—
46	A20	O	—	—	—	—
47	PE2	I(s)/O	A21	O	—	—
48	PE3	I(s)/O	A22	O	—	—
49	PE0	I(s)/O	$\overline{\text{BS}}$	O	—	—

引脚编号	功能 4		功能 5		功能 6		弱保持	上拉
	引脚名称	I/O	引脚名称	I/O	引脚名称	I/O		
33	—	—	—	—	—	—	○	
34								
35	—	—	—	—	—	—	○	
36	—	—	—	—	—	—	○	
37	—	—	—	—	—	—	○	
38	—	—	—	—	—	—	○	
39	—	—	—	—	—	—	○	
40	—	—	—	—	—	—	○	
41								
42	—	—	—	—	—	—	○	
43								
44	—	—	—	—	—	—	○	
45	—	—	—	—	—	—	○	
46	—	—	—	—	—	—	○	
47	SCK0	I(s)/O	—	—	—	—	○	
48	SCK1	I(s)/O	—	—	—	—	○	
49	RxD0	I(s)	$\overline{\text{ADTRG}}$	I(s)	—	—	○	

引脚编号	功能 1		功能 2		功能 3	
	引脚名称	I/O	引脚名称	I/O	引脚名称	I/O
50	CKIO	I/O	—	—	—	—
51	V _{CC}					
52	V _{SS}					
53	PV _{SS}					
54	PV _{CC}					
55	XTAL	O	—	—	—	—
56	EXTAL	I	—	—	—	—
57	NMI	I(s)	—	—	—	—
58	PLL _{VSS}					
59	RES	I(s)	—	—	—	—
60	PLL _{VCC}					
61	ASEMD	I(s)	—	—	—	—
62	PE8	I(s)/O	CE2A	O	IRQ4	I(s)
63	PE1	I(s)/O	CS4	O	MRES	I(s)
64	PE4	I(s)/O	A23	O	IRQ0	I(s)
65	PV _{SS}					
66	PE5	I(s)/O	A24	O	IRQ1	I(s)

引脚编号	功能 4		功能 5		功能 6		弱保持	上拉
	引脚名称	I/O	引脚名称	I/O	引脚名称	I/O		
50	—	—	—	—	—	—		
51								
52								
53								
54								
55	—	—	—	—	—	—		
56	—	—	—	—	—	—		
57	—	—	—	—	—	—		
58								
59	—	—	—	—	—	—		
60								
61	—	—	—	—	—	—		
62	SCK2	I(s)/O	—	—	—	—	○	
63	TxD0	O	—	—	—	—	○	
64	RxD1	I(s)	DREQ0	I(s)	—	—	○	
65								
66	TxD1	O	DACK0	O	—	—	○	

引脚编号	功能 1		功能 2		功能 3	
	引脚名称	I/O	引脚名称	I/O	引脚名称	I/O
67	PVcc					
68	PE6	I(s)/O	A25	O	IRQ2	I(s)
69	PE7	I(s)/O	FRAME	O	IRQ3	I(s)
70	PE9	I(s)/O	CS5/CE1A	O	IRQ5	I(s)
71	PE10	I(s)/O	CE2B	O	IRQ6	I(s)
72	PE11	I(s)/O	CS6/CE1B	O	IRQ7	I(s)
73	PE12	I(s)/O	—	—	—	—
74	Vcc					
75	PC14	I/O	WAIT	I	—	—
76	Vss					
77	PVss					
78	RTC_X1	I	—	—	—	—
79	RTC_X2	O	—	—	—	—
80	PVcc					
81	PE13	I(s)/O	—	—	—	—
82	PE14	I(s)/O	CS1	O	—	—
83	PE15	I(s)/O	IOIS16	I(s)	—	—

引脚编号	功能 4		功能 5		功能 6		弱保持	上拉
	引脚名称	I/O	引脚名称	I/O	引脚名称	I/O		
67								
68	RxD2	I(s)	DREQ1	I(s)	—	—	○	
69	TxD2	O	DACK1	O	—	—	○	
70	SCK3	I(s)/O	—	—	—	—	○	
71	—	—	TEND0	O	—	—	○	
72	—	—	TEND1	O	—	—	○	
73	RxD3	I(s)	—	—	—	—	○	
74								
75	—	—	—	—	—	—	○	
76								
77								
78	—	—	—	—	—	—		
79	—	—	—	—	—	—		
80								
81	TxD3	O	—	—	—	—	○	
82	CTS3	I(s)/O	—	—	—	—	○	
83	RTS3	I(s)/O	—	—	—	—	○	

引脚编号	功能 1		功能 2		功能 3	
	引脚名称	I/O	引脚名称	I/O	引脚名称	I/O
84	PVss					
85	PB8	I/O	CRx0	I	CRx0/CRx1	I
86	PB9	I/O	CTx0	O	CTx0&CTx1	O
87	PB10	I/O	CRx1	I	—	—
88	PB11	I/O	CTx1	O	—	—
89	Vcc					
90	MD	I(s)	—	—	—	—
91	Vss					
92	PVss					
93	USB_X1	I	—	—	—	—
94	USB_X2	O	—	—	—	—
95	PVcc					
96	MD_CLK1	I(s)	—	—	—	—
97	MD_CLK0	I(s)	—	—	—	—
98	USBDPVss					
99	USBDPVcc					
100	DM	I/O	—	—	—	—

引脚编号	功能 4		功能 5		功能 6		弱保持	上拉
	引脚名称	I/O	引脚名称	I/O	引脚名称	I/O		
84								
85	—	—	—	—	—	—	○	
86	—	—	—	—	—	—	○	
87	—	—	—	—	—	—	○	
88	—	—	—	—	—	—	○	
89								
90	—	—	—	—	—	—		
91								
92								
93	—	—	—	—	—	—		
94	—	—	—	—	—	—		
95								
96	—	—	—	—	—	—		
97	—	—	—	—	—	—		
98								
99								
100	—	—	—	—	—	—		

引脚编号	功能 1		功能 2		功能 3	
	引脚名称	I/O	引脚名称	I/O	引脚名称	I/O
101	DP	I/O	—	—	—	—
102	VBUS	I	—	—	—	—
103	USBAV _{CC}					
104	USBAV _{SS}					
105	REFRIN	I	—	—	—	—
106	USBAPV _{SS}					
107	USBAPV _{CC}					
108	USBDV _{CC}					
109	USBDV _{SS}					
110	PA0	I	AN0	I(a)	—	—
111	PA1	I	AN1	I(a)	—	—
112	PA2	I	AN2	I(a)	—	—
113	PA3	I	AN3	I(a)	—	—
114	AV _{CC}					
115	PA4	I	AN4	I(a)	—	—
116	AV _{ref}					
117	PA5	I	AN5	I(a)	—	—

引脚编号	功能 4		功能 5		功能 6		弱保持	上拉
	引脚名称	I/O	引脚名称	I/O	引脚名称	I/O		
101	—	—	—	—	—	—		
102	—	—	—	—	—	—		
103								
104								
105	—	—	—	—	—	—		
106								
107								
108								
109								
110	—	—	—	—	—	—		
111	—	—	—	—	—	—		
112	—	—	—	—	—	—		
113	—	—	—	—	—	—		
114								
115	—	—	—	—	—	—		
116								
117	—	—	—	—	—	—		

引脚编号	功能 1		功能 2		功能 3	
	引脚名称	I/O	引脚名称	I/O	引脚名称	I/O
118	PA6	I	AN6	I(a)	DA0	O(a)
119	PA7	I	AN7	I(a)	DA1	O(a)
120	AVss					
121	PVss					
122	PB12	O	WDTOVF	O	IRQOUT/REFOUT	O
123	PVcc					
124	PF29	I/O	SSIDATA3	I/O	—	—
125	PF28	I/O	SSIWS3	I/O	—	—
126	PF27	I/O	SSISCK3	I/O	—	—
127	Vcc					
128	PF30	I/O	AUDIO_CLK	I	—	—
129	Vss					
130	PVss					
131	AUDIO_X1	I	—	—	—	—
132	AUDIO_X2	O	—	—	—	—
133	PVcc					
134	PF26	I/O	SSIDATA2	I/O	—	—

引脚编号	功能 4		功能 5		功能 6		弱保持	上拉
	引脚名称	I/O	引脚名称	I/O	引脚名称	I/O		
118	—	—	—	—	—	—		
119	—	—	—	—	—	—		
120								
121								
122	UBCTR $\overline{G}$	O	AUDCK	O	—	—	○	
123								
124	—	—	—	—	—	—	○	
125	—	—	—	—	—	—	○	
126	—	—	—	—	—	—	○	
127								
128	—	—	—	—	—	—	○	
129								
130								
131	—	—	—	—	—	—		
132	—	—	—	—	—	—		
133								
134	—	—	—	—	—	—	○	

引脚编号	功能 1		功能 2		功能 3	
	引脚名称	I/O	引脚名称	I/O	引脚名称	I/O
135	PF25	I/O	SSIWS2	I/O	—	—
136	PF24	I/O	SSISCK2	I/O	—	—
137	PF18	I/O	SSISCK0	I/O	LCD_CL2	O
138	PF19	I/O	SSIWS0	I/O	LCD_M_DISP	O
139	PF20	I/O	SSIDATA0	I/O	LCD_FLM	O
140	PF21	I/O	SSISCK1	I/O	LCD_CLK	I
141	Vcc					
142	PF22	I/O	SSIWS1	I/O	LCD_VCPWC	O
143	Vss					
144	PVss					
145	PF23	I/O	SSIDATA1	I/O	LCD_VEPWC	O
146	PVcc					
147	PF17	I/O	$\overline{\text{FCE}}$	O	LCD_CL1	O
148	PF16	I/O	FRB	I	LCD_DON	O
149	PF15	I/O	NAF7	I/O	LCD_DATA15	O
150	PF14	I/O	NAF6	I/O	LCD_DATA14	O
151	PF13	I/O	NAF5	I/O	LCD_DATA13	O

引脚编号	功能 4		功能 5		功能 6		弱保持	上拉
	引脚名称	I/O	引脚名称	I/O	引脚名称	I/O		
135	—	—	—	—	—	—	○	
136	—	—	—	—	—	—	○	
137	—	—	—	—	—	—	○	
138	—	—	—	—	—	—	○	
139	—	—	—	—	—	—	○	
140	—	—	—	—	—	—	○	
141								
142	AUDATA2	O	—	—	—	—	○	
143								
144								
145	AUDATA3	O	—	—	—	—	○	
146								
147	—	—	—	—	—	—	○	
148	—	—	—	—	—	—	○	
149	—	—	—	—	—	—	○	
150	—	—	—	—	—	—	○	
151	—	—	—	—	—	—	○	

引脚编号	功能 1		功能 2		功能 3	
	引脚名称	I/O	引脚名称	I/O	引脚名称	I/O
152	PF12	I/O	NAF4	I/O	LCD_DATA12	O
153	PF11	I/O	NAF3	I/O	LCD_DATA11	O
154	V _{CC}					
155	PF10	I/O	NAF2	I/O	LCD_DATA10	O
156	V _{SS}					
157	PV _{SS}					
158	PF9	I/O	NAF1	I/O	LCD_DATA9	O
159	PV _{CC}					
160	PF8	I/O	NAF0	I/O	LCD_DATA8	O
161	PF7	I(s)/O	FSC	O	LCD_DATA7	O
162	PF6	I(s)/O	FOE	O	LCD_DATA6	O
163	PF5	I(s)/O	FCDE	O	LCD_DATA5	O
164	PF4	I(s)/O	$\overline{\text{FWE}}$	O	LCD_DATA4	O
165	PF3	I(s)/O	TCLKD	I(s)	LCD_DATA3	O
166	PF2	I(s)/O	TCLKC	I(s)	LCD_DATA2	O
167	PF1	I(s)/O	TCLKB	I(s)	LCD_DATA1	O
168	V _{CC}					

引脚编号	功能 4		功能 5		功能 6		弱保持	上拉
	引脚名称	I/O	引脚名称	I/O	引脚名称	I/O		
152	—	—	—	—	—	—	○	
153	—	—	—	—	—	—	○	
154								
155	—	—	—	—	—	—	○	
156								
157								
158	—	—	—	—	—	—	○	
159								
160	—	—	—	—	—	—	○	
161	$\overline{\text{SCS1}}$	I(s)/O	—	—	—	—	○	
162	SSO1	I(s)/O	—	—	—	—	○	
163	SSI1	I(s)/O	—	—	—	—	○	
164	SSCK1	I(s)/O	—	—	—	—	○	
165	$\overline{\text{SCS0}}$	I(s)/O	—	—	—	—	○	
166	SSO0	I(s)/O	—	—	—	—	○	
167	SSI0	I(s)/O	—	—	—	—	○	
168								

引脚编号	功能 1		功能 2		功能 3	
	引脚名称	I/O	引脚名称	I/O	引脚名称	I/O
169	PF0	I(s)/O	TCLKA	I(s)	LCD_DATA0	O
170	Vss					
171	PVss					
172	TMS	I	—	—	—	—
173	PVcc					
174	TDI	I	—	—	—	—
175	ASEBRKAK/ASEBRK	I(s)/O	—	—	—	—
176	TRST	I(s)	—	—	—	—
177	TDO	O	—	—	—	—
178	TCK	I	—	—	—	—
179	PB0	I(s)	SCL0	I(s)/O(o)	PINT0	I(s)
180	PB1	I(s)	SDA0	I(s)/O(o)	PINT1	I(s)
181	PB2	I(s)	SCL1	I(s)/O(o)	PINT2	I(s)
182	PB3	I(s)	SDA1	I(s)/O(o)		I(s)
183	PVcc					
184	PVcc					
185	PB4	I(s)	SCL2	I(s)/O(o)	PINT4	I(s)

引脚编号	功能 4		功能 5		功能 6		弱保持	上拉
	引脚名称	I/O	引脚名称	I/O	引脚名称	I/O		
169	SSCK0	I(s)/O	—	—	—	—	○	
170								
171								
172	—	—	—	—	—	—		○
173								
174	—	—	—	—	—	—		○
175	—	—	—	—	—	—	○	
176	—	—	—	—	—	—		○
177	—	—	—	—	—	—		
178	—	—	—	—	—	—		○
179	IRQ0	I(s)	—	—	—	—		
180	IRQ1	I(s)	—	—	—	—		
181	IRQ2	I(s)	—	—	—	—		
182	IRQ3	I(s)	—	—	—	—		
183								
184								
185	IRQ4	I(s)	—	—	—	—		

引脚编号	功能 1		功能 2		功能 3	
	引脚名称	I/O	引脚名称	I/O	引脚名称	I/O
186	PB5	I(s)	SDA2	I(s)/O(o)	PINT5	I(s)
187	PVss					
188	Vss					
189	PB6	I(s)	SCL3	I(s)/O(o)	PINT6	I(s)
190	PB7	I(s)	SDA3	I(s)/O(o)	PINT7	I(s)
191	Vcc					
192	PD15	I/O	D31	I/O	PINT7	I(s)
193	PD14	I/O	D30	I/O	PINT6	I(s)
194	PD13	I/O	D29	I/O	PINT5	I(s)
195	PD12	I/O	D28	I/O	PINT4	I(s)
196	PVss					
197	PD11	I/O	D27	I/O	PINT3	I(s)
198	PVcc					
199	PD10	I/O	D26	I/O	PINT2	I(s)
200	PD9	I/O	D25	I/O	PINT1	I(s)
201	PD8	I/O	D24	I/O	PINT0	I(s)
202	PD7	I/O	D23	I/O	IRQ7	I(s)

引脚编号	功能 4		功能 5		功能 6		弱保持	上拉
	引脚名称	I/O	引脚名称	I/O	引脚名称	I/O		
186	IRQ5	I(s)	—	—	—	—		
187								
188								
189	IRQ6	I(s)	—	—	—	—		
190	IRQ7	I(s)	—	—	—	—		
191								
192	—	—	ADTRG	I(s)	TIOC4D	I(s)/O	○	
193	—	—	—	—	TIOC4C	I(s)/O	○	
194	—	—	TEND1	O	TIOC4B	I(s)/O	○	
195	—	—	DACK1	O	TIOC4A	I(s)/O	○	
196								
197	—	—	DREQ1	I(s)	TIOC3D	I(s)/O	○	
198								
199	—	—	TEND0	O	TIOC3C	I(s)/O	○	
200	—	—	DACK0	O	TIOC3B	I(s)/O	○	
201	—	—	DREQ0	I(s)	TIOC3A	I(s)/O	○	
202	SCS1	I(s)/O	TCLKD	I(s)	TIOC2B	I(s)/O	○	

引脚编号	功能 1		功能 2		功能 3	
	引脚名称	I/O	引脚名称	I/O	引脚名称	I/O
203	PD6	I/O	D22	I/O	IRQ6	I(s)
204	V _{CC}					
205	PD5	I/O	D21	I/O	IRQ5	I(s)
206	V _{SS}					
207	PV _{SS}					
208	PD4	I/O	D20	I/O	IRQ4	I(s)
209	PV _{CC}					
210	PD3	I/O	D19	I/O	IRQ3	I(s)
211	PD2	I/O	D18	I/O	IRQ2	I(s)
212	PD1	I/O	D17	I/O	IRQ1	I(s)
213	PD0	I/O	D16	I/O	IRQ0	I(s)
214	D15	I/O	—	—	—	—
215	D14	I/O	—	—	—	—
216	PV _{SS}					
217	D13	I/O	—	—	—	—
218	PV _{CC}					
219	D12	I/O	—	—	—	—

引脚编号	功能 4		功能 5		功能 6		弱保持	上拉
	引脚名称	I/O	引脚名称	I/O	引脚名称	I/O		
203	SSO1	I(s)/O	TCLKC	I(s)	TIOC2A	I(s)/O	○	
204								
205	SSI1	I(s)/O	TCLKB	I(s)	TIOC1B	I(s)/O	○	
206								
207								
208	SSCK1	I(s)/O	TCLKA	I(s)	TIOC1A	I(s)/O	○	
209								
210	SCS0	I(s)/O	DACK3	O	TIOC0D	I(s)/O	○	
211	SSO0	I(s)/O	DREQ3	I(s)	TIOC0C	I(s)/O	○	
212	SSI0	I(s)/O	DACK2	O	TIOC0B	I(s)/O	○	
213	SSCK0	I(s)/O	DREQ2	I(s)	TIOC0A	I(s)/O	○	
214	—	—	—	—	—	—	○	
215	—	—	—	—	—	—	○	
216								
217	—	—	—	—	—	—	○	
218								
219	—	—	—	—	—	—	○	

引脚编号	功能 1		功能 2		功能 3	
	引脚名称	I/O	引脚名称	I/O	引脚名称	I/O
220	D11	I/O	—	—	—	—
221	D10	I/O	—	—	—	—
222	D9	I/O	—	—	—	—
223	D8	I/O	—	—	—	—
224	V _{CC}					
225	D7	I/O	—	—	—	—
226	V _{SS}					
227	PV _{SS}					
228	D6	I/O	—	—	—	—
229	PV _{CC}					
230	D5	I/O	—	—	—	—
231	D4	I/O	—	—	—	—
232	D3	I/O	—	—	—	—
233	D2	I/O	—	—	—	—
234	D1	I/O	—	—	—	—
235	D0	I/O	—	—	—	—
236	PV _{SS}					

引脚编号	功能 4		功能 5		功能 6		弱保持	上拉
	引脚名称	I/O	引脚名称	I/O	引脚名称	I/O		
220	—	—	—	—	—	—	○	
221	—	—	—	—	—	—	○	
222	—	—	—	—	—	—	○	
223	—	—	—	—	—	—	○	
224								
225	—	—	—	—	—	—	○	
226								
227								
228	—	—	—	—	—	—	○	
229								
230	—	—	—	—	—	—	○	
231	—	—	—	—	—	—	○	
232	—	—	—	—	—	—	○	
233	—	—	—	—	—	—	○	
234	—	—	—	—	—	—	○	
235	—	—	—	—	—	—	○	
236								

引脚编号	功能 1		功能 2		功能 3	
	引脚名称	I/O	引脚名称	I/O	引脚名称	I/O
237	PV _{CC}					
238	PC13	I/O	RD/WR	O	—	—
239	PC12	I/O	CKE	O	—	—
240	PC11	I/O	CASU	O	BREQ	I

引脚编号	功能 4		功能 5		功能 6		弱保持	上拉
	引脚名称	I/O	引脚名称	I/O	引脚名称	I/O		
237								
238	—	—	—	—	—	—	○	
239	—	—	—	—	—	—	○	
240	AUDATA1	O	—	—	—	—	○	

【符号说明】

(s): 施密特

(a): 模拟

(o): 漏极开路

第 2 章 CPU

2.1 寄存器结构

寄存器有通用寄存器（32 位 ×16 个）、控制寄存器（32 位 ×4 个）及系统寄存器（32 位 ×4 个）等 3 种。

2.1.1 通用寄存器

通用寄存器如图 2.1 所示，其长度为 32 位，从 R0 到 R15 共有 16 个，用于数据处理与地址运算。R0 也可用作变址寄存器，在多个指令可使用的寄存器固定为 R0。R15 可用作硬件堆栈指针（SP）。使用 R15 参照堆栈，执行异常处理时状态寄存器（SR）与程序计数器（PC）的保存与恢复。

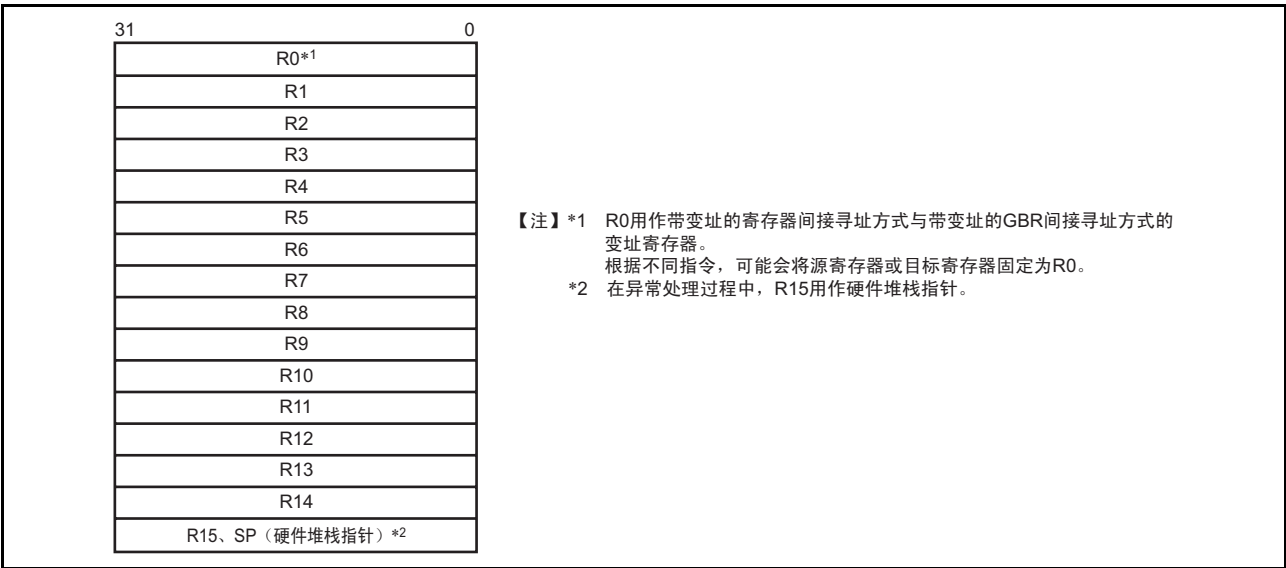


图 2.1 通用寄存器

2.1.2 控制寄存器

控制寄存器长度为 32 位，有状态寄存器（SR）、全局基址寄存器（GBR）、向量基址寄存器（VBR）及跳转表基址寄存器（TBR）等 4 个。

SR 表示各指令的处理状态。

GBR 作为 GBR 间接寻址方式的基址，用于传送内部外围模块寄存器的数据等。

VBR 用作包含中断的异常处理向量区域的基址。

TBR 用作函数表区域的基址。

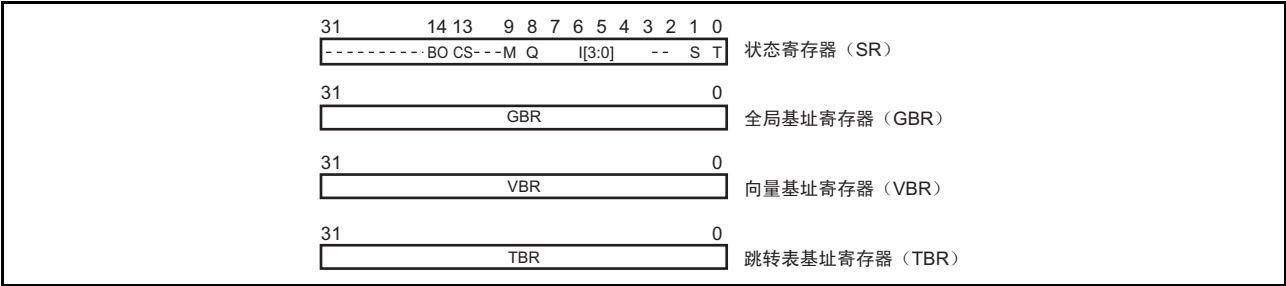


图 2.2 控制寄存器

(1) 状态寄存器（SR）

位：	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
初始值：	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W：	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位：	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	BO	CS	-	-	-	M	Q	I[3:0]				-	-	S	T
初始值：	0	0	0	0	0	0	-	-	1	1	1	1	0	0	-	-
R/W：	R	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W

位	位名称	初始值	R/W	说 明
31 ~ 15	—	均为 0	R	保留位 读取值、写入值总是为 0。
14	BO	0	R/W	BO 位 表示寄存器存储体上溢。
13	CS	0	R/W	CS 位 表示由于执行 CLIP 指令，超出饱和上限值或低于饱和下限值。
12 ~ 10	—	均为 0	R	保留位 读取值、写入值总是为 0。

位	位名称	初始值	R/W	说 明
9	M	—	R/W	M 位 Q 位 用于 DIV0S、DIV0U、DIV1 指令。
8	Q	—	R/W	
7 ~ 4	I[3:0]	1111	R/W	中断屏蔽级
3、2	—	均为 0	R	保留位 读取值、写入值总是为 0。
1	S	—	R/W	S 位 指定 MAC 指令的饱和运行。
0	T	—	R/W	T 位 真 / 假条件或进位 / 借位

(2) 全局基址寄存器（GBR）

GBR 被参照为 GBR 参照 MOV 指令的基址。

(3) 向量基址寄存器（VBR）

产生异常及中断时，VBR 被参照为转移目标的基址。

(4) 跳转表基址寄存器（TBR）

通过表参照子程序调用指令 JSR/N@@(disp8,TBR)，TBR 被参照为配置于存储器的函数表起始地址。

2.1.3 系统寄存器

系统寄存器长度为 32 位，有乘加寄存器（MACH、MACL）、过程寄存器（PR）及程序计数器（PC）等 4 个。MACH、MACL 保存乘法或乘法累加运算的结果。PR 保存从子程序过程返回的目标地址。PC 表示当前执行指令的 4 字节后的地址，并控制处理流程。

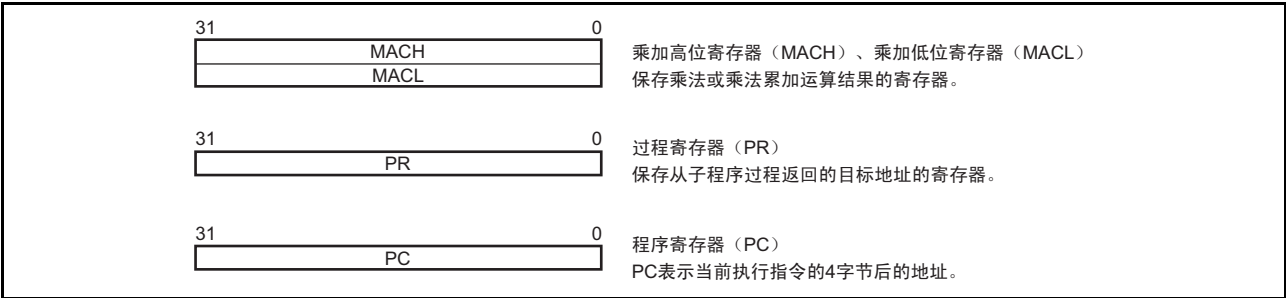


图 2.3 系统寄存器

(1) 乘加高位寄存器 (MACH)、乘加低位寄存器 (MACL)

MACH 及 MACL 用作 MAC 指令的加法值，同时也用于保存 MAC 指令、MUL 指令的运算结果。

(2) 过程寄存器 (PR)

PR 保存使用 BSR、BSRF 或 JSR 指令的子程序调用的返回地址，且通过程序的返回指令 (RTS) 被参照。

(3) 程序计数器 (PC)

PC 表示当前执行指令的 4 字节后的地址。

2.1.4 寄存器存储体

通用寄存器的 R0 ~ R14、控制寄存器 GBR、系统寄存器的 MACH、MACL 及 PR 等 19 个 32 位寄存器，使用寄存器存储体，可高速执行寄存器的保存与返回。CPU 接受使用寄存器存储体的中断后，自动保存至存储体。通过在中断处理程序发行 RESBANK 指令，可从存储体返回。

本 LSI 有 15 个存储体，详情参阅“SH-2A、SH2A-FPU 软件手册”及“6.8 寄存器存储器”。

2.1.5 寄存器的初始值

复位后的寄存器值如表 2.1 所示。

表 2.1 寄存器的初始值

分类	寄存器	初始值
通用寄存器	R0 ~ R14	不定
	R15 (SP)	向量地址表中 SP 的值
控制寄存器	SR	I[3:0] 为 1111 (H'F)，BO、CS 为 0，保留位为 0，其他位不定
	GBR、TBR	不定
	VBR	H'00000000
系统寄存器	MACH、MACL、PR	不定
	PC	向量地址表中 PC 的值

2.2 数据格式

2.2.1 寄存器的数据格式

寄存器操作数的数据长度总是为长字（32 位）。存储器的数据加载至寄存器时，如果存储器操作数的数据长度为字节（8 位）或字（16 位），则符号扩展为长字，并保存至寄存器。

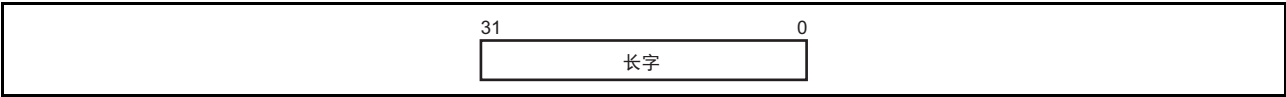


图 2.4 寄存器的数据格式

2.2.2 存储器的数据格式

存储器有字节、字及长字 3 种数据格式，能以 8 位字节、16 位字或 32 位长字存取。未满 32 位的存储器操作数符号扩展或零扩展后，可保存至寄存器。

字节操作数必须从字边界（每 2 个字节的偶数地址：地址 $2n$ ）存取，长字操作数必须从长字边界（每 4 个字节的偶数地址：地址 $4n$ ）存取。否则，会产生地址错误。可从任意地址存取字节操作数。

数据格式仅可选择大端法的字节顺序。

存储器的数据格式如图 2.5 所示。

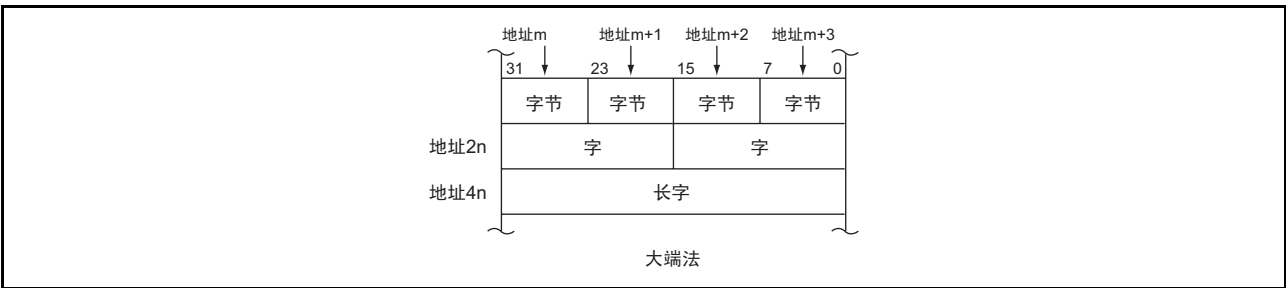


图 2.5 存储器的数据格式

2.2.3 立即数的数据格式

字节（8 位）长度的立即数配置于操作码。

MOV、ADD、CMP/EQ 指令时，将立即数符号扩展后，以长字运算；TST、AND、OR、XOR 指令时，将立即数零扩展后，以长字运算。如果 AND 指令使用立即数，则总是清除目标寄存器的高 24 位。

20 位立即数配置于 32 位传送指令 MOVI20 及 MOVI20S 的操作码。MOVI20 指令将立即数符号扩展，并保存至目标寄存器；MOVI20S 指令将立即数向高位移动 8 位，符号扩展并保存至目标寄存器。

字与长字的立即数不配置于操作码，而配置于存储器表。使用带位移量的 PC 相对寻址方式的立即数数据传送指令（MOV），参照存储器表。

详情参阅“2.3.1(10) 立即数”。

2.3 指令特点

2.3.1 RISC 结构

指令为 RISC 结构，特点如下：

(1) 16 位固定长度指令

基本指令为 16 位固定长度，可提高程序的编码效率。

(2) 32 位固定长度指令

在 SH-2A 追加了 32 位固定长度的指令，可提高性能与适用性。

(3) 1 条指令 / 1 个状态

采用流水线方式，能够以 1 条指令 / 1 个状态执行基本指令。

(4) 数据长度

运算的基本数据长度为长字。存储器的存取长度可选择字节 / 字 / 长字。存储器的字节数据与字数据符号扩展后，以长字运算。立即数在算术运算时，符号扩展后，以长字运算；在逻辑运算时，零扩展后，以长字运算。

表 2.2 字数据的符号扩展

SH-2A CPU	说 明	其他 CPU 的例子
MOV.W @ (disp,PC),R1 ADD R1,R0DATA.W H'1234	符号扩展为 32 位，R1 为 H'00001234。 接着由 ADD 指令执行运算。	ADD.W #H'1234,R0

【注】 通过 @ (disp,PC) 参照立即数。

(5) 加载存储结构

基本运算在寄存器间执行。将数据加载至寄存器后，执行与存储器的运算（加载存储结构）。但可直接对存储器执行 AND 等位的操作指令。

(6) 延迟转移

无条件转移指令为除去部分指令的延迟转移指令。延迟转移指令时，执行延迟转移指令的下一条指令后，再执行转移。由此，可减少转移时的流水线混乱。

延迟转移时，在执行槽指令后产生转移，但仍要按延迟转移指令 → 延迟槽指令的顺序执行指令（更新寄存器等）。例如，即使在延迟槽更改保存转移目标地址的寄存器，更改前的寄存器内容仍为转移目标地址。

表 2.3 延迟转移指令

SH-2A CPU	说 明	其他 CPU 的例子
BRA TRGET ADD R1,R0	转移至 TRGET 前，执行 ADD。	ADD.W R1,R0 BRA TRGET

(7) 无延迟槽的无条件转移指令

在 SH-2A 追加了不执行延迟槽指令的无条件转移指令，可削减不需要的 NOP 指令及代码长度。

(8) 乘法 / 乘法累加运算

以 1 ~ 2 个状态执行 $16 \times 16 \rightarrow 32$ 的乘法运算；以 2 ~ 3 个状态执行 $16 \times 16 + 64 \rightarrow 64$ 的乘法累加运算；以 2 ~ 4 个状态执行 $32 \times 32 \rightarrow 64$ 的乘法及 $32 \times 32 + 64 \rightarrow 64$ 的乘法累加运算。

(9) T 位

比较结果反映在状态寄存器（SR）的 T 位，根据其真假执行条件转移。仅用尽可能少的必要指令改变 T 位，使处理速度提高。

表 2.4 T 位

SH-2A CPU	说 明	其他 CPU 的例子
CMP/GE R1,R0 BT TRGET0 BF TRGET1	R0 ≥ R1 时，T 位置位。 R0 ≥ R1 时，转移至 TRGET0。 R0 < R1 时，转移至 TRGET1。	CMP.W R1,R0 BGE TRGET0 BLT TRGET1
ADD #-1,R0 CMP/EQ #0,R0 BT TRGET	ADD 时，T 位不变。 R0=0 时，T 位置位。 R0=0 时，转移。	SUB.W #1,R0 BEQ TRGET

(10) 立即数

字节的立即数配置于操作码。字与长字的立即数不配置于操作码，而配置于存储器表。通过使用带位移量的 PC 相对寻址方式的立即数数据传送指令（MOV），参照存储器表。

SH-2A 中，17 ~ 28 位立即数也可配置于操作码。但是，21 ~ 28 位立即数需在寄存器传送后，执行 OR 指令。

表 2.5 通过立即数参照

分类	SH-2A CPU	其他 CPU 的例子
8 位立即数	MOV #H'12,R0	MOV.B #H'12,R0
16 位立即数	MOVI20 #H'1234,R0	MOV.W #H'1234,R0
20 位立即数	MOVI20 #H'12345,R0	MOV.L #H'12345,R0
28 位立即数	MOVI20S #H'12345, R0 OR #H'67, R0	MOV.L #H'1234567,R0
32 位立即数	MOV.L @(disp,PC),R0DATA.L H'12345678	MOV.L #H'12345678,R0

【注】 通过 @(disp,PC) 参照立即数。

(11) 绝对地址

通过绝对地址参照数据时，先将绝对地址的值配置于存储器表。执行指令时，通过加载立即数的方法将该值传送至寄存器，并通过寄存器间接寻址方式参照数据。

在 SH-2A，由小于等于 28 位的绝对地址参照数据时，也可将配置于操作码的立即数传送至寄存器，并以寄存器间接寻址方式参照数据。但在使用 21 ~ 28 位绝对地址参照数据时，需在寄存器传送后，使用 OR 指令。

表 2.6 通过绝对地址参照

分类	SH-2A CPU	其他 CPU 的例子
小于等于 20 位	MOVI20 #H'12345,R1 MOV.B @R1,R0	MOV.B @H'12345,R0
21 ~ 28 位	MOVI20S #H'12345,R1 OR #H'67,R1 MOV.B @R1,R0	MOV.B @H'1234567,R0
大于等于 29 位	MOV.L @(disp, PC),R1 MOV.B @R1,R0DATA.L H'12345678	MOV.B @H'12345678,R0

(12) 16 位 /32 位位移量

通过 16 位或 32 位位移量参照数据时，先将位移量的值配置于存储器表。执行指令时，通过加载立即数的方法将该值传送至寄存器，并通过带变址的寄存器间接寻址方式参照数据。

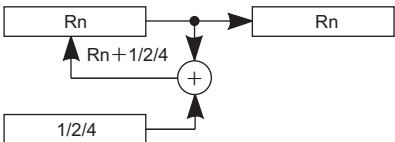
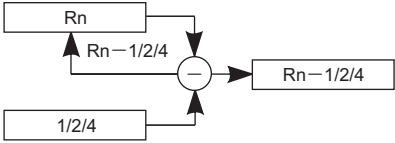
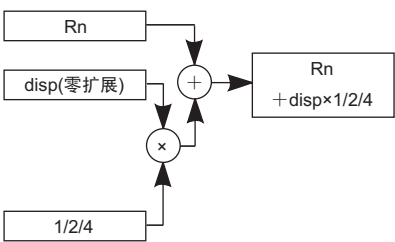
表 2.7 通过位移量参照

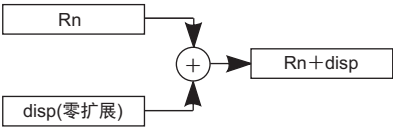
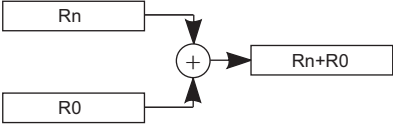
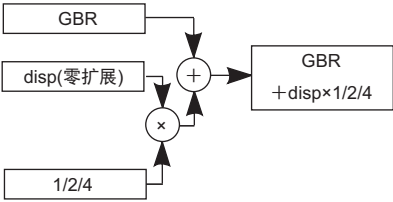
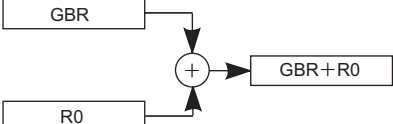
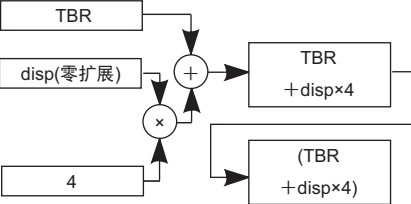
分类	SH-2A CPU	其他 CPU 的例子
16 位位移量	MOV.W @(disp,PC),R0 MOV.W @(R0,R1),R2DATA.W H'1234	MOV.W @(H'1234,R1),R2

2.3.2 寻址方式

寻址方式与有效地址的计算方法如下：

表 2.8 寻址方式与有效地址

寻址方式	指令格式	有效地址的计算方法	计算式
寄存器直接寻址	Rn	有效地址为寄存器 Rn。 (操作数为寄存器 Rn 的内容。)	—
寄存器间接寻址	@Rn	有效地址为寄存器 Rn 的内容。 	Rn
后增寄存器间接寻址	@Rn+	有效地址为寄存器 Rn 的内容。执行指令后给 Rn 加常数。操作数长度为字节时，常数是 1；为字时，常数是 2；为长字时，常数是 4。 	Rn 指令执行后 字节: $Rn+1 \rightarrow Rn$ 字: $Rn+2 \rightarrow Rn$ 长字: $Rn+4 \rightarrow Rn$
先减寄存器间接寻址	@-Rn	有效地址为先减去常数后寄存器 Rn 的内容。操作数长度为字节时，常数是 1；为字时，常数是 2；为长字时，常数是 4。 	字节: $Rn - 1 \rightarrow Rn$ 字: $Rn - 2 \rightarrow Rn$ 长字: $Rn - 4 \rightarrow Rn$ (用计算后的 Rn 执行指令)
带位移量的寄存器间接寻址	@(disp:4,Rn)	有效地址为寄存器 Rn 加 4 位位移量 disp 后的内容。disp 零扩展后，根据操作数长度倍增，操作数长度为字节时，是 1 倍；为字时，是 2 倍；为长字时，是 4 倍。 	字节: $Rn + disp$ 字: $Rn + disp \times 2$ 长字: $Rn + disp \times 4$

寻址方式	指令格式	有效地址的计算方法	计算式
带位移量的寄存器间接寻址	@(disp:12,Rn)	有效地址为寄存器 Rn 加 12 位位移量 disp 后的内容。disp 零扩展。 	字节: $Rn+disp$ 字: $Rn+disp$ 长字: $Rn+disp$
带变址的寄存器间接寻址	@(R0,Rn)	有效地址为寄存器 Rn 加 R0 后的内容。 	$Rn+R0$
带位移量的 GBR 间接寻址	@(disp:8,GBR)	有效地址为寄存器 GBR 加 8 位位移量 disp 后的内容。disp 零扩展后, 根据操作数长度倍增, 操作数长度为字节时, 是 1 倍; 为字时, 是 2 倍; 为长字时, 是 4 倍。 	字节: $GBR+disp$ 字: $GBR+disp \times 2$ 长字: $GBR+disp \times 4$
带变址的 GBR 间接寻址	@(R0, GBR)	有效地址为寄存器 GBR 加 R0 后的内容。 	$GBR+R0$
带位移量的 TBR 双重间接寻址	@@(disp:8,TBR)	有效地址为寄存器 TBR 加 8 位位移量 disp 后的内容。disp 零扩展后乘以 4。 	($TBR+disp \times 4$) 地址的内容

寻址方式	指令格式	有效地址的计算方法	计算式
带位移量的 PC 相对寻址	@(disp:8,PC)	有效地址为寄存器 PC 加 8 位位移量 disp 后的内容。disp 零扩展后, 根据操作数长度倍增, 操作数长度为字时, 是 2 倍; 为长字时, 是 4 倍。另外, 为长字时, 屏蔽 PC 的低 2 位。	字: $PC + \text{disp} \times 2$ 长字: $PC \& H'FFFFFFFC + \text{disp} \times 4$
PC 相对寻址	disp:8	有效地址为 8 位位移量 disp 符号扩展后乘以 2, 再加寄存器 PC 后的内容。	$PC + \text{disp} \times 2$
	disp:12	有效地址为 12 位位移量 disp 符号扩展后乘以 2, 再加寄存器 PC 后的内容。	$PC + \text{disp} \times 2$
	Rn	有效地址为寄存器 PC 加 Rn 后的内容。	$PC + Rn$

寻址方式	指令格式	有效地址的计算方法	计算式
立即寻址	#imm:20	符号扩展 MOVI20 指令的 20 位立即数 imm。 31190 符号扩展imm (20位)	—
		MOVI20S 指令的 20 位立即数 imm 左移 8 位，高位符号扩展、低位零填充。 312780 ↑ 符号扩展imm20位00000000	—
	#imm:8	零扩展 TST、AND、OR、XOR 指令的 8 位立即数 imm。	—
	#imm:8	符号扩展 MOV、ADD、CMP/EQ 指令的 8 位立即数 imm。	—
	#imm:8	零扩展 TRAPA 指令的 8 位立即数 imm 后乘以 4。	—
	#imm:3	BAND、BOR、BXOR、BST、BLD、BSET、BCLR 指令的 3 位立即数 imm 表示位的位置。	—

2.3.3 指令格式

本节描述了指令格式、源操作数及目标操作数。操作数的含义取决于操作码，符号说明如下：

xxxx：操作码
 mmmm：源寄存器
 nnnn：目标寄存器
 iiii：立即数
 dddd：位移量

表 2.9 指令格式

指令格式	源操作数	目标操作数	指令例
0 格式	—	—	NOP
n 格式	—	nnnn: 寄存器直接寻址	MOVT Rn
	控制寄存器或系统寄存器	nnnn: 寄存器直接寻址	STS MACH,Rn
	R0 (寄存器直接寻址)	nnnn: 寄存器直接寻址	DIVU R0,Rn
	控制寄存器或系统寄存器	nnnn: 先减寄存器间接寻址	STC.L SR,@-Rn
	mmmm: 寄存器直接寻址	R15 (先减寄存器间接寻址)	MOVMUL Rm,@-R15
	R15 (后增寄存器间接寻址)	nnnn: 寄存器直接寻址	MOVMUL @R15+, Rn
m 格式	R0 (寄存器直接寻址)	nnnn: 后增寄存器间接寻址	MOV.L R0,@Rn+
	mmmm: 寄存器直接寻址	控制寄存器或系统寄存器	LDC Rm,SR
	mmmm: 后增寄存器间接寻址	控制寄存器或系统寄存器	LDC.L @Rm+,SR
	mmmm: 寄存器间接寻址	—	JMP @Rm
	mmmm: 先减寄存器间接寻址	R0 (寄存器直接寻址)	MOV.L @-Rm, R0
nm 格式	mmmm: 使用 Rm 的 PC 相对寻址	—	BRAF Rm
	mmmm: 寄存器直接寻址	nnnn: 寄存器直接寻址	ADD Rm,Rn
	mmmm: 寄存器直接寻址	nnnn: 寄存器间接寻址	MOV.L Rm,@Rn
	mmmm: 后增寄存器间接寻址 (乘法累加运算)	MACH, MACL	MAC.W @Rm+,@Rn+
	nnnn*: 后增寄存器间接寻址 (乘法累加运算)		
	mmmm: 后增寄存器间接寻址	nnnn: 寄存器直接寻址	MOV.L @Rm+,Rn
md 格式	mmmm: 寄存器直接寻址	nnnn: 先减寄存器间接寻址	MOV.L Rm,@-Rn
	mmmm: 寄存器直接寻址	nnnn: 带变址的寄存器间接寻址	MOV.L Rm,@(R0,Rn)
	mmmmddd: 带位移量的寄存器间接寻址	R0 (寄存器直接寻址)	MOV.B @(disp,Rm),R0

指令格式	源操作数	目标操作数	指令例
nd4 格式	15 0 xxxx xxxx nnnn dddd	R0 (寄存器直接寻址)	nnnndddd: 带位移量的寄存器间接寻址
nmd 格式	15 0 xxxx nnnn mmmm dddd	mmmm: 寄存器直接寻址	nnnndddd: 带位移量的寄存器间接寻址
nmd12 格式	32 16 xxxx nnnn mmmm xxxx 15 0 xxxx dddd dddd dddd	mmmm: 寄存器直接寻址	nnnndddd: 带位移量的寄存器间接寻址
d 格式	15 0 xxxx xxxx dddd dddd	dddddddd: 带位移量的 GBR 间接寻址	R0 (寄存器直接寻址)
d12 格式	15 0 xxxx dddd dddd dddd	dddddddd: 带位移量的 GBR 间接寻址	—
nd8 格式	15 0 xxxx nnnn dddd dddd	dddddddd: 带位移量的 PC 相对寻址	—
i 格式	15 0 xxxx xxxx iiii iiii	iiiiiiii: 立即寻址	nnnn: 寄存器直接寻址
ni 格式	15 0 xxxx nnnn iiii iiii	iiiiiiii: 立即寻址	—
ni3 格式	15 0 xxxx xxxx nnnn xiii	nnnn: 寄存器直接寻址	—
ni20 格式	32 16 xxxx nnnn iiii xxxx 15 0 iiii iiii iiii iiii	iiiiiiiiiiiiiiiiiiii: 立即寻址	nnnn: 寄存器直接寻址

指令格式		源操作数	目标操作数	指令例
nid 格式	32 16 xxxx nnnn iiii xxxx	nnnnddddddddddd: 带位移量的寄存器间接寻址 iii: 立即寻址	—	BLD.B #imm3,@(disp12,Rn)
	15 0 xxxx dddd dddd dddd	—	nnnnddddddddddd: 带位移量的寄存器间接寻址 iii: 立即寻址	BST.B #imm3,@(disp12,Rn)

【注】 * 乘加指令时， nnnn 为源寄存器。

2.4 指令系统

2.4.1 指令系统分类

指令分类如表 2.10 所示。

表 2.10 指令分类

分类	指令的种类	操作码	功能	指令数
数据传送指令	13	MOV	传送数据 传送立即数 传送外围模块数据 传送结构体数据 传送反向堆栈	62
		MOVA	传送有效地址	
		MOVI20	传送 20 位立即数	
		MOVI20S	传送 20 位立即数 左移 8 位	
		MOVML	寄存器 R0 ~ Rn 的保存 / 返回	
		MOV MU	寄存器 Rn ~ R14、PR 的保存 / 返回	
		MOV RT	T 位取反并传送至 Rn	
		MOV T	传送 T 位	
		MOV U	传送无符号数据	
		NOT T	T 位取反	
		PREF	预取操作数高速缓存	
		SWAP	交换高位与低位	
		XTRCT	抽出连接寄存器的中间部分	
算术运算指令	26	ADD	2 进制加法	40
		ADDC	带进位的 2 进制加法	
		ADDV	带上溢的 2 进制加法	
		CMP/cond	比较	
		CLIPS	比较带符号的饱和值	
		CLIPU	比较无符号的饱和值	
		DIVS	带符号除法 (32÷32)	
		DIVU	无符号除法 (32÷32)	
		DIV1	单步除法	
		DIV0S	初始化带符号的单步除法	
		DIV0U	初始化无符号的单步除法	
		DMULS	带符号的双精度乘法	
		DMULU	无符号的双精度乘法	
		DT	递减与测试	
		EXTS	符号扩展	
		EXTU	零扩展	

分类	指令的种类	操作码	功能	指令数
算术运算指令	26	MAC	乘法累加运算、双精度乘法累加运算	40
		MUL	双精度乘法	
		MULR	保存 Rn 结果的带符号乘法	
		MULS	带符号乘法	
		MULU	无符号乘法	
		NEG	符号取反	
		NEGC	带借位的符号取反	
		SUB	2 进制减法	
		SUBC	带借位的 2 进制减法	
		SUBV	带下溢的 2 进制减法	
逻辑运算指令	6	AND	逻辑 “与” 运算	14
		NOT	位取反	
		OR	逻辑 “或” 运算	
		TAS	存储器测试与置位	
		TST	逻辑 “与” 运算的 T 位置位	
		XOR	逻辑 “异或” 运算	
移位指令	12	ROTL	左循环 1 位	16
		ROTR	右循环 1 位	
		ROTCL	带 T 位左循环 1 位	
		ROTCR	带 T 位右循环 1 位	
		SHAD	动态算术移位	
		SHAL	算术左移 1 位	
		SHAR	算术右移 1 位	
		SHLD	动态逻辑移位	
		SHLL	逻辑左移 1 位	
		SHLLn	逻辑左移 n 位	
		SHLR	逻辑右移 1 位	
		SHLRn	逻辑右移 n 位	
转移指令	10	BF	条件转移、带延迟的条件转移 (T=0 时转移)	15
		BT	条件转移、带延迟的条件转移 (T=1 时转移)	
		BRA	带延迟的无条件转移	
		BRAF	带延迟的无条件转移	
		BSR	转移至带延迟的子程序过程	
		BSRF	转移至带延迟的子程序过程	
		JMP	带延迟的无条件转移	
		JSR	转移至子程序过程 转移至带延迟的子程序过程	

分类	指令的种类	操作码	功能	指令数
转移指令	10	RTS	从子程序过程返回 从带延迟的子程序过程返回	15
		RTV/N	从带 Rm→R0 传送的子程序过程返回	
系统控制指令	14	CLRT	清除 T 位	36
		CLRMAC	清除 MAC 寄存器	
		LDBANK	从指定寄存器存储体入口的寄存器返回	
		LDC	加载至控制寄存器	
		LDS	加载至系统寄存器	
		NOP	空操作	
		RESBANK	从寄存器存储体的寄存器返回	
		RTE	从异常处理返回	
		SETT	T 位置位	
		SLEEP	转移至低功耗状态	
		STBANK	寄存器保存至指定寄存器存储体入口	
		STC	存储控制寄存器	
		STS	存储系统寄存器	
		TRAPA	陷阱异常处理	
浮点运算指令	19	FABS	浮点数绝对值	48
		FADD	浮点数加法	
		FCMP	比较浮点数	
		FCNVDS	从双精度转换为单精度	
		FCNVSD	从单精度转换为双精度	
		FDIV	浮点数除法	
		FLDI0	浮点数加载立即数 0	
		FLDI1	浮点数加载立即数 1	
		FLDS	向系统寄存器 FPUL 加载浮点数	
		FLOAT	从整数转换为浮点数	
		FMAC	浮点数乘法累加运算	
		FMOV	传送浮点数	
		FMUL	浮点数乘法	
		FNEG	浮点数符号取反	
		FSCHG	SZ 位取反	
		FSQRT	浮点平方根	
		FSTS	存储系统寄存器 FPUL 的浮点数	
		FSUB	浮点数减法	
		FTRC	舍入转换为浮点数的整数	

分类	指令的种类	操作码	功能	指令数
关于 FPU 的 CPU 指令	2	LDS	加载至浮点系统寄存器	8
		STS	存储浮点系统寄存器	
位操作指令	10	BAND	位 “与”	14
		BCLR	位清除	
		BLD	位加载	
		BOR	位逻辑 “或”	
		BSET	置位	
		BST	位存储	
		BXOR	位 “异或”	
		BANDNOT	位 “与非”	
		BORNOT	位 “或非”	
		BLDNOT	位 “非” 加载	
	计 112			253

指令的操作码、操作及执行状态，按下列格式分类说明：

指 令	操作码	操作概略	执行状态	T 位
用助记符表示。 【符号说明】 Rm : 源寄存器 Rn : 目标寄存器 imm : 立即数 disp : 位移量 *2	按照 MSB←→LSB 的顺序表示。 【符号说明】 mmmm : 源寄存器 nnnn : 目标寄存器 0000: R0 0001: R1 1111: R15 iiii : 立即数 dddd : 位移量	表示操作概略。 【符号说明】 →、← : 传送方向 (xx) : 存储器操作数 M/Q/T : SR 内的标志 & : 位 : 位 “与” ^ : 位 “或” ~ : 位 “异或” <<n : 位 “非” >>n : 左移 n 位 : 右移 n 位	无等待时的值。 *1	表示指令执行后 T 位的值。 【符号说明】 — : 不变化

【注】 *1 指令的执行状态：

表中所示的执行状态为最小值，实际根据以下条件，指令执行的状态数会增加。

(1) 取指令与数据存取产生竞争

(2) 加载指令（存储器→寄存器）的目标寄存器与紧随其后的指令所用的寄存器相同

*2 根据指令的操作数长度倍增（×1、×2、×4）。

详情参阅“SH-2A、SH2A-FPU 软件手册”。

2.4.2 数据传送指令

表 2.11 数据传送指令

指 令	操作码	操 作	执行 状态	T 位	适用指令		
					SH2、 SH2E	SH4	SH-2A
MOV #imm, Rn	1110nnnniiiiiii	imm→符号扩展→Rn	1	—	○	○	○
MOV.W @(disp, PC), Rn	1001nnnnddddd	(disp×2+PC)→符号扩展→Rn	1	—	○	○	○
MOV.L @(disp, PC), Rn	1101nnnnddddd	(disp×4+PC)→Rn	1	—	○	○	○
MOV Rm, Rn	0110nnnnmmmm0011	Rm→Rn	1	—	○	○	○
MOV.B Rm, @Rn	0010nnnnmmmm0000	Rm→(Rn)	1	—	○	○	○
MOV.W Rm, @Rn	0010nnnnmmmm0001	Rm→(Rn)	1	—	○	○	○
MOV.L Rm, @Rn	0010nnnnmmmm0010	Rm→(Rn)	1	—	○	○	○
MOV.B @Rm, Rn	0110nnnnmmmm0000	(Rm)→符号扩展→Rn	1	—	○	○	○
MOV.W @Rm, Rn	0110nnnnmmmm0001	(Rm)→符号扩展→Rn	1	—	○	○	○
MOV.L @Rm, Rn	0110nnnnmmmm0010	(Rm)→Rn	1	—	○	○	○
MOV.B Rm, @- Rn	0010nnnnmmmm0100	Rn-1→Rn、Rm→(Rn)	1	—	○	○	○
MOV.W Rm, @- Rn	0010nnnnmmmm0101	Rn-2→Rn、Rm→(Rn)	1	—	○	○	○
MOV.L Rm, @- Rn	0010nnnnmmmm0110	Rn-4→Rn、Rm→(Rn)	1	—	○	○	○
MOV.B @Rm+, Rn	0110nnnnmmmm0100	(Rm)→符号扩展→Rn、 Rm+1→Rm	1	—	○	○	○
MOV.W @Rm+, Rn	0110nnnnmmmm0101	(Rm)→符号扩展→Rn、 Rm+2→Rm	1	—	○	○	○
MOV.L @Rm+, Rn	0110nnnnmmmm0110	(Rm)→Rn、Rm+4→Rm	1	—	○	○	○
MOV.B R0, @(disp, Rn)	10000000nnnnddddd	R0→(disp+Rn)	1	—	○	○	○
MOV.W R0, @(disp, Rn)	10000001nnnnddddd	R0→(disp×2+Rn)	1	—	○	○	○
MOV.L Rm, @(disp, Rn)	0001nnnnmmmmddddd	Rm→(disp×4+Rn)	1	—	○	○	○
MOV.B @(disp, Rm), R0	10000100mmmmddddd	(disp+Rm)→符号扩展→R0	1	—	○	○	○
MOV.W @(disp, Rm), R0	10000101mmmmddddd	(disp×2+Rm)→符号扩展→R0	1	—	○	○	○
MOV.L @(disp, Rm), Rn	0101nnnnmmmmddddd	(disp×4+Rm)→Rn	1	—	○	○	○
MOV.B Rm, @(R0, Rn)	0000nnnnmmmm0100	Rm→(R0+Rn)	1	—	○	○	○
MOV.W Rm, @(R0, Rn)	0000nnnnmmmm0101	Rm→(R0+Rn)	1	—	○	○	○
MOV.L Rm, @(R0, Rn)	0000nnnnmmmm0110	Rm→(R0+Rn)	1	—	○	○	○
MOV.B @(R0, Rm), Rn	0000nnnnmmmm1100	(R0+Rm)→符号扩展→Rn	1	—	○	○	○
MOV.W @(R0, Rm), Rn	0000nnnnmmmm1101	(R0+Rm)→符号扩展→Rn	1	—	○	○	○
MOV.L @(R0, Rm), Rn	0000nnnnmmmm1110	(R0+Rm)→Rn	1	—	○	○	○
MOV.B R0, @(disp, GBR)	11000000ddddd	R0→(disp+GBR)	1	—	○	○	○
MOV.W R0, @(disp, GBR)	11000001ddddd	R0→(disp×2+GBR)	1	—	○	○	○
MOV.L R0, @(disp, GBR)	11000010ddddd	R0→(disp×4+GBR)	1	—	○	○	○

指 令	操作码	操 作	执行 状态	T 位	适用指令		
					SH2、 SH2E	SH4	SH-2A
MOV.B @(disp, GBR), R0	11000100dddddddd	(disp+GBR)→ 符号扩展 →R0	1	—	○	○	○
MOV.W @(disp, GBR), R0	11000101dddddddd	(disp×2+GBR)→ 符号扩展 →R0	1	—	○	○	○
MOV.L @(disp, GBR), R0	11000110dddddddd	(disp×4+GBR)→R0	1	—	○	○	○
MOV.B R0, @Rn+	0100nnnn10001011	R0→(Rn)、Rn+1→Rn	1	—			○
MOV.W R0, @Rn+	0100nnnn10011011	R0→(Rn)、Rn+2→Rn	1	—			○
MOV.L R0, @Rn+	0100nnnn10101011	R0→(Rn)、Rn+4→Rn	1	—			○
MOV.B @ – Rm, R0	0100mmmm11001011	Rm–1→Rm、 (Rm)→ 符号扩展 →R0	1	—			○
MOV.W @ – Rm, R0	0100mmmm11011011	Rm–2→Rm、 (Rm)→ 符号扩展 →R0	1	—			○
MOV.L @ – Rm, R0	0100mmmm11101011	Rm–4→Rm、(Rm)→R0	1	—			○
MOV.B Rm, @(disp12, Rn)	0011nnnnmmmm0001 0000dddddddddddd	Rm→(disp+Rn)	1	—			○
MOV.W Rm, @(disp12, Rn)	0011nnnnmmmm0001 0001dddddddddddd	Rm→(disp×2+Rn)	1	—			○
MOV.L Rm, @(disp12, Rn)	0011nnnnmmmm0001 0010dddddddddddd	Rm→(disp×4+Rn)	1	—			○
MOV.B @(disp12, Rm), Rn	0011nnnnmmmm0001 0100dddddddddddd	(disp+Rm)→ 符号扩展 →Rn	1	—			○
MOV.W @(disp12, Rm), Rn	0011nnnnmmmm0001 0101dddddddddddd	(disp×2+Rm)→ 符号扩展 →Rn	1	—			○
MOV.L @(disp12, Rm), Rn	0011nnnnmmmm0001 0110dddddddddddd	(disp×4+Rm)→Rn	1	—			○
MOVA @(disp, PC), R0	11000111dddddddd	disp×4+PC→R0	1	—	○	○	○
MOVI20 #imm20, Rn	0000nnnniiii0000 iiiiiiiiiiiiiiii	imm→ 符号扩展 →Rn	1	—			○
MOVI20S #imm20, Rn	0000nnnniiii0001 iiiiiiiiiiiiiiii	imm<<8→ 符号扩展 →Rn	1	—			○
MOVML.L Rm, @ – R15	0100mmmm11110001	R15 – 4→R15、Rm→(R15) R15 – 4→R15、Rm – 1→(R15) : R15 – 4→R15、R0→(R15) ※ Rm=R15 时, 将 Rm 改读 为 PR	1 ~ 16	—			○

指 令	操作码	操 作	执行 状态	T 位	适用指令		
					SH2、 SH2E	SH4	SH-2A
MOVML.L @R15+, Rn	0100nnnn11110101	(R15)→R0、R15+4→R15 (R15)→R1、R15+4→R15 : (R15)→Rn ※ Rn=R15 时, 将 Rn 改读 为 PR	1 ~ 16	—			○
MOVMUL Rm, @-R15	0100mmmm11110000	R15-4→R15、 PR→(R15) R15-4→R15、 R14→(R15) : R15-4→R15、 Rm→(R15) ※ Rm=R15 时, 将 Rm 改 读为 PR	1 ~ 16	—			○
MOVMUL @R15+, Rn	0100nnnn11110100	(R15)→Rn、R15+4→R15 (R15)→Rn+1、 R15+4→R15 : (R15)→R14、R15+4→R15 (R15)→PR ※ Rn=R15 时, 将 Rn 改读 为 PR	1 ~ 16	—			○
MOVRT Rn	0000nnnn00111001	~ T→Rn	1	—			○
MOVT Rn	0000nnnn00101001	T→Rn	1	—	○	○	○
MOVU.B @(disp12,Rm), Rn	0011nnnnmmmm0001 1000dddddddddd	(disp+Rm)→零扩展→Rn	1	—			○
MOVU.W @(disp12,Rm),Rn	0011nnnnmmmm0001 1001dddddddddd	(disp×2+Rm)→零扩展 →Rn	1	—			○
NOTT	0000000001101000	~ T→T	1	运算 结果			○
PREF @Rn	0000nnnn10000011	(Rn)→操作数高速缓存	1	—		○	○
SWAP.B Rm, Rn	0110nnnnmmmm1000	Rm→交换低位 2 字节的高 低字节→Rn	1	—	○	○	○
SWAP.W Rm, Rn	0110nnnnmmmm1001	Rm→交换高低字→Rn	1	—	○	○	○
XTRCT Rm, Rn	0010nnnnmmmm1101	Rm:Rn 的中间 32 位→Rn	1	—	○	○	○

2.4.3 算术运算指令

表 2.12 算术运算指令

指 令	操作码	操 作	执行 状态	T 位	适用指令		
					SH2、 SH2E	SH4	SH-2A
ADD Rm, Rn	0011nnnnmmmm1100	$Rn+Rm \rightarrow Rn$	1	—	○	○	○
ADD #imm, Rn	0111nnnniiiiiii	$Rn+imm \rightarrow Rn$	1	—	○	○	○
ADDC Rm, Rn	0011nnnnmmmm1110	$Rn+Rm+T \rightarrow Rn$ 、进位 $\rightarrow T$	1	进位	○	○	○
ADDV Rm, Rn	0011nnnnmmmm1111	$Rn+Rm \rightarrow Rn$ 、上溢 $\rightarrow T$	1	上溢	○	○	○
CMP/EQ #imm, R0	10001000iiiiiii	$R0=imm$ 时 $1 \rightarrow T$ 除此之外 $0 \rightarrow T$	1	比较 结果	○	○	○
CMP/EQ Rm, Rn	0011nnnnmmmm0000	$Rn=Rm$ 时 $1 \rightarrow T$ 除此之外 $0 \rightarrow T$	1	比较 结果	○	○	○
CMP/HS Rm, Rn	0011nnnnmmmm0010	无符号 $Rn \geq Rm$ 时 $1 \rightarrow T$ 除此之外 $0 \rightarrow T$	1	比较 结果	○	○	○
CMP/GE Rm, Rn	0011nnnnmmmm0011	带符号 $Rn \geq Rm$ 时 $1 \rightarrow T$ 除此之外 $0 \rightarrow T$	1	比较 结果	○	○	○
CMP/Hi Rm, Rn	0011nnnnmmmm0110	无符号 $Rn > Rm$ 时 $1 \rightarrow T$ 除此之外 $0 \rightarrow T$	1	比较 结果	○	○	○
CMP/GT Rm, Rn	0011nnnnmmmm0111	带符号且 $Rn > Rm$ 时 $1 \rightarrow T$ 除此之外 $0 \rightarrow T$	1	比较 结果	○	○	○
CMP/PL Rn	0100nnnn00010101	$Rn > 0$ 时 $1 \rightarrow T$ 除此之外 $0 \rightarrow T$	1	比较 结果	○	○	○
CMP/PZ Rn	0100nnnn00010001	$Rn \geq 0$ 时 $1 \rightarrow T$ 除此之外 $0 \rightarrow T$	1	比较 结果	○	○	○
CMP/STR Rm, Rn	0010nnnnmmmm1100	任意字节相等时 $1 \rightarrow T$ 除此之外 $0 \rightarrow T$	1	比较 结果	○	○	○
CLIPS.B Rn	0100nnnn10010001	$Rn > (H'0000007F)$ 时, $(H'0000007F) \rightarrow Rn$ 、 $1 \rightarrow CS$ $Rn < (H'FFFFFF80)$ 时, $(H'FFFFFF80) \rightarrow Rn$ 、 $1 \rightarrow CS$	1	—			○
CLIPS.W Rn	0100nnnn10010101	$Rn > (H'00007FFF)$ 时, $(H'00007FFF) \rightarrow Rn$ 、 $1 \rightarrow CS$ $Rn < (H'FFFF8000)$ 时, $(H'FFFF8000) \rightarrow Rn$ 、 $1 \rightarrow CS$	1	—			○
CLIPU.B Rn	0100nnnn10000001	$Rn > (H'000000FF)$ 时, $(H'000000FF) \rightarrow Rn$ 、 $1 \rightarrow CS$	1	—			○
CLIPU.W Rn	0100nnnn10000101	$Rn > (H'0000FFFF)$ 时, $(H'0000FFFF) \rightarrow Rn$ 、 $1 \rightarrow CS$	1	—			○
DIV1 Rm, Rn	0011nnnnmmmm0100	单步除法 ($Rn \div Rm$)	1	计算 结果	○	○	○

指 令	操作码	操 作	执行 状态	T 位	适用指令		
					SH2、 SH2E	SH4	SH-2A
DIV0S Rm, Rn	0010nnnnmmmm0111	Rn 的 MSB→Q、Rm 的 MSB→M, $M^{\wedge}Q \rightarrow T$	1	计算结果	○	○	○
DIV0U	0000000000011001	$0 \rightarrow M/Q/T$	1	0	○	○	○
DIVS R0, Rn	0100nnnn10010100	带符号 $Rn \div R0 \rightarrow Rn$ 32÷32→32 位	36	—			○
DIVU R0, Rn	0100nnnn10000100	无符号 $Rn \div R0 \rightarrow Rn$ 32÷32→32 位	34	—			○
DMULS.L Rm, Rn	0011nnnnmmmm1101	带符号 $Rn \times Rm \rightarrow MACH$ 、 $MACL$ 32×32→64 位	2	—	○	○	○
DMULU.L Rm, Rn	0011nnnnmmmm0101	无符号 $Rn \times Rm \rightarrow MACH$ 、 $MACL$ 32×32→64 位	2	—	○	○	○
DT Rn	0100nnnn00010000	$Rn - 1 \rightarrow Rn$ 、Rn 为 0 时 $1 \rightarrow T$ Rn 不为 0 时 $0 \rightarrow T$	1	比较结果	○	○	○
EXTS.B Rm, Rn	0110nnnnmmmm1110	Rm 从字节符号扩展 →Rn	1	—	○	○	○
EXTS.W Rm, Rn	0110nnnnmmmm1111	Rm 从字符号扩展 →Rn	1	—	○	○	○
EXTU.B Rm, Rn	0110nnnnmmmm1100	Rm 从字节零扩展 →Rn	1	—	○	○	○
EXTU.W Rm, Rn	0110nnnnmmmm1101	Rm 从字零扩展 →Rn	1	—	○	○	○
MAC.L @Rm+,@Rn+	0000nnnnmmmm1111	带符号 $(Rn) \times (Rm) + MAC \rightarrow MAC$ 32×32+64→64 位	4	—	○	○	○
MAC.W @Rm+,@Rn+	0100nnnnmmmm1111	带符号 $(Rn) \times (Rm) + MAC \rightarrow MAC$ 16×16+64→64 位	3	—	○	○	○
MUL.L Rm, Rn	0000nnnnmmmm0111	$Rn \times Rm \rightarrow MACL$ 32×32→32 位	2	—	○	○	○
MULR R0, Rn	0100nnnn10000000	$R0 \times Rn \rightarrow Rn$ 32×32→32 位	2				○
MULS.W Rm, Rn	0010nnnnmmmm1111	带符号 $Rn \times Rm \rightarrow MACL$ 16×16→32 位	1	—	○	○	○
MULU.W Rm, Rn	0010nnnnmmmm1110	无符号 $Rn \times Rm \rightarrow MACL$ 16×16→32 位	1	—	○	○	○
NEG Rm, Rn	0110nnnnmmmm1011	$0 - Rm \rightarrow Rn$	1	—	○	○	○
NEGC Rm, Rn	0110nnnnmmmm1010	$0 - Rm - T \rightarrow Rn$ 、借位 →T	1	借位	○	○	○
SUB Rm, Rn	0011nnnnmmmm1000	$Rn - Rm \rightarrow Rn$	1	—	○	○	○
SUBC Rm, Rn	0011nnnnmmmm1010	$Rn - Rm - T \rightarrow Rn$ 、借位 →T	1	借位	○	○	○
SUBV Rm, Rn	0011nnnnmmmm1011	$Rn - Rm \rightarrow Rn$ 、下溢 →T	1	上溢	○	○	○

2.4.4 逻辑运算指令

表 2.13 逻辑运算指令

指 令	操作码	操 作	执行 状态	T 位	适用指令		
					SH2、 SH2E	SH4	SH-2A
AND Rm, Rn	0010nnnnmmmm1001	$Rn \& Rm \rightarrow Rn$	1	—	○	○	○
AND #imm, R0	11001001iiiiiii	$R0 \& imm \rightarrow R0$	1	—	○	○	○
AND.B #imm, @(R0,GBR)	11001101iiiiiii	$(R0+GBR) \& imm \rightarrow (R0+GBR)$	3	—	○	○	○
NOT Rm, Rn	0110nnnnmmmm0111	$\sim Rm \rightarrow Rn$	1	—	○	○	○
OR Rm, Rn	0010nnnnmmmm1011	$Rn \mid Rm \rightarrow Rn$	1	—	○	○	○
OR #imm, R0	11001011iiiiiii	$R0 \mid imm \rightarrow R0$	1	—	○	○	○
OR.B #imm, @(R0,GBR)	11001111iiiiiii	$(R0+GBR) \mid imm \rightarrow (R0+GBR)$	3	—	○	○	○
TAS.B @Rn	0100nnnn00011011	(Rn) 为 0 时 1→T、除此之外 0→T、1→MSB of(Rn)	3	测试 结果	○	○	○
TST Rm, Rn	0010nnnnmmmm1000	$Rn \& Rm$ 、结果为 0 时 1→T、 其他 0→T	1	测试 结果	○	○	○
TST #imm, R0	11001000iiiiiii	$R0 \& imm$ 、结果为 0 时 1→T 其他 0→T	1	测试 结果	○	○	○
TST.B #imm, @(R0,GBR)	11001100iiiiiii	$(R0+GBR) \& imm$ 、结果为 0 时 1→T 其他 0→T	3	测试 结果	○	○	○
XOR Rm, Rn	0010nnnnmmmm1010	$Rn \wedge Rm \rightarrow Rn$	1	—	○	○	○
XOR #imm, R0	11001010iiiiiii	$R0 \wedge imm \rightarrow R0$	1	—	○	○	○
XOR.B #imm, @(R0,GBR)	11001110iiiiiii	$(R0+GBR) \wedge imm \rightarrow (R0+GBR)$	3	—	○	○	○

2.4.5 移位指令

表 2.14 移位指令

指 令	操作码	操 作	执行 状态	T 位	适用指令		
					SH2、 SH2E	SH4	SH-2A
ROTL Rn	0100nnnn00000100	$T \leftarrow Rn \leftarrow \text{MSB}$	1	MSB	○	○	○
ROTR Rn	0100nnnn00000101	$\text{LSB} \rightarrow Rn \rightarrow T$	1	LSB	○	○	○
ROTCL Rn	0100nnnn00100100	$T \leftarrow Rn \leftarrow T$	1	MSB	○	○	○
ROTCR Rn	0100nnnn00100101	$T \rightarrow Rn \rightarrow T$	1	LSB	○	○	○
SHAD Rm, Rn	0100nnnnmmmm1100	$Rm \geq 0$ 时 $Rn \ll Rm \rightarrow Rn$ $Rm < 0$ 时 $Rn \gg Rm \rightarrow [MSB \rightarrow Rn]$	1	—		○	○
SHAL Rn	0100nnnn00100000	$T \leftarrow Rn \leftarrow 0$	1	MSB	○	○	○
SHAR Rn	0100nnnn00100001	$MSB \rightarrow Rn \rightarrow T$	1	LSB	○	○	○
SHLD Rm, Rn	0100nnnnmmmm1101	$Rm \geq 0$ 时 $Rn \ll Rm \rightarrow Rn$ $Rm < 0$ 时 $Rn \gg Rm \rightarrow [0 \rightarrow Rn]$	1	—		○	○
SHLL Rn	0100nnnn00000000	$T \leftarrow Rn \leftarrow 0$	1	MSB	○	○	○
SHLR Rn	0100nnnn00000001	$0 \rightarrow Rn \rightarrow T$	1	LSB	○	○	○
SHLL2 Rn	0100nnnn00001000	$Rn \ll 2 \rightarrow Rn$	1	—	○	○	○
SHLR2 Rn	0100nnnn00001001	$Rn \gg 2 \rightarrow Rn$	1	—	○	○	○
SHLL8 Rn	0100nnnn00011000	$Rn \ll 8 \rightarrow Rn$	1	—	○	○	○
SHLR8 Rn	0100nnnn00011001	$Rn \gg 8 \rightarrow Rn$	1	—	○	○	○
SHLL16 Rn	0100nnnn00101000	$Rn \ll 16 \rightarrow Rn$	1	—	○	○	○
SHLR16 Rn	0100nnnn00101001	$Rn \gg 16 \rightarrow Rn$	1	—	○	○	○

2.4.6 转移指令

表 2.15 转移指令

指 令	操作码	操 作	执行 状态	T 位	适用指令		
					SH2、 SH2E	SH4	SH-2A
BF label	10001011dddddddd	T=0 时 $\text{disp} \times 2 + \text{PC} \rightarrow \text{PC}$ 、 T=1 时 nop	3/1*	—	○	○	○
BF/S label	10001111dddddddd	延迟转移、T=0 时 $\text{disp} \times 2 + \text{PC} \rightarrow \text{PC}$ 、 T=1 时 nop	2/1*	—	○	○	○
BT label	10001001dddddddd	T=1 时 $\text{disp} \times 2 + \text{PC} \rightarrow \text{PC}$ 、 T=0 时 nop	3/1*	—	○	○	○
BT/S label	10001101dddddddd	延迟转移、T=1 时 $\text{disp} \times 2 + \text{PC} \rightarrow \text{PC}$ 、 T=0 时 nop	2/1*	—	○	○	○
BRA label	1010dddddddddddd	延迟转移、 $\text{disp} \times 2 + \text{PC} \rightarrow \text{PC}$	2	—	○	○	○
BRAF Rm	0000mmmm00100011	延迟转移、 $\text{Rm} + \text{PC} \rightarrow \text{PC}$	2	—	○	○	○
BSR label	1011dddddddddddd	延迟转移、 $\text{PC} \rightarrow \text{PR}$ 、 $\text{disp} \times 2 + \text{PC} \rightarrow \text{PC}$	2	—	○	○	○
BSRF Rm	0000mmmm00000011	延迟转移、 $\text{PC} \rightarrow \text{PR}$ 、 $\text{Rm} + \text{PC} \rightarrow \text{PC}$	2	—	○	○	○
JMP @Rm	0100mmmm00101011	延迟转移、 $\text{Rm} \rightarrow \text{PC}$	2	—	○	○	○
JSR @Rm	0100mmmm00001011	延迟转移、 $\text{PC} \rightarrow \text{PR}$ 、 $\text{Rm} \rightarrow \text{PC}$	2	—	○	○	○
JSR/N @Rm	0100mmmm01001011	$\text{PC} - 2 \rightarrow \text{PR}$ 、 $\text{Rm} \rightarrow \text{PC}$	3	—			○
JSR/N @@(disp8, TBR)	10000011dddddddd	$\text{PC} - 2 \rightarrow \text{PR}$ 、 $(\text{disp} \times 4 + \text{TBR}) \rightarrow \text{PC}$	5	—			○
RTS	0000000000001011	延迟转移、 $\text{PR} \rightarrow \text{PC}$	2	—	○	○	○
RTS/N	0000000001101011	$\text{PR} \rightarrow \text{PC}$	3	—			○
RTV/N Rm	0000mmmm01111011	$\text{Rm} \rightarrow \text{R0}$ 、 $\text{PR} \rightarrow \text{PC}$	3	—			○

【注】 * 不转移时，为 1 个状态。

2.4.7 系统控制指令

表 2.16 系统控制指令

指 令	操作码	操 作	执行 状态	T 位	适用指令		
					SH2、 SH2E	SH4	SH-2A
CLRT	0000000000001000	0→T	1	0	○	○	○
CLRMAC	0000000000101000	0→MACH、MACL	1	—	○	○	○
LDBANK @Rm, R0	0100mmmm11100101	(指定寄存器存储体入口) →R0	6	—			○
LDC Rm, SR	0100mmmm00001110	Rm→SR	3	LSB	○	○	○
LDC Rm, TBR	0100mmmm01001010	Rm→TBR	1	—			○
LDC Rm, GBR	0100mmmm00011110	Rm→GBR	1	—	○	○	○
LDC Rm, VBR	0100mmmm00101110	Rm→VBR	1	—	○	○	○
LDC.L @Rm+, SR	0100mmmm00000111	(Rm)→SR、Rm+4→Rm	5	LSB	○	○	○
LDC.L @Rm+, GBR	0100mmmm00010111	(Rm)→GBR、Rm+4→Rm	1	—	○	○	○
LDC.L @Rm+, VBR	0100mmmm00100111	(Rm)→VBR、Rm+4→Rm	1	—	○	○	○
LDS Rm, MACH	0100mmmm00001010	Rm→MACH	1	—	○	○	○
LDS Rm, MACL	0100mmmm00011010	Rm→MACL	1	—	○	○	○
LDS Rm, PR	0100mmmm00101010	Rm→PR	1	—	○	○	○
LDS.L @Rm+, MACH	0100mmmm00000110	(Rm)→MACH、Rm+4→Rm	1	—	○	○	○
LDS.L @Rm+, MACL	0100mmmm00010110	(Rm)→MACL、Rm+4→Rm	1	—	○	○	○
LDS.L @Rm+, PR	0100mmmm00100110	(Rm)→PR、Rm+4→Rm	1	—	○	○	○
NOP	0000000000001001	空操作	1	—	○	○	○
RESBANK	0000000001011011	存储体 →R0 ~ R14、GBR、 MACH、MACL、PR	9*	—			○
RTE	0000000000101011	延迟转移、堆栈区 →PC/SR	6	—	○	○	○
SETT	0000000000011000	1→T	1	1	○	○	○
SLEEP	0000000000011011	睡眠	5	—	○	○	○
STBANK R0, @Rn	0100nnnn11100001	R0→ (指定寄存器存储体入口)	7	—			○
STC SR, Rn	0000nnnn00000010	SR→Rn	2	—	○	○	○
STC TBR, Rn	0000nnnn01001010	TBR→Rn	1	—			○
STC GBR, Rn	0000nnnn00010010	GBR→Rn	1	—	○	○	○
STC VBR, Rn	0000nnnn00100010	VBR→Rn	1	—	○	○	○
STC.L SR, @-Rn	0100nnnn00000011	Rn-4→Rn、SR→(Rn)	2	—	○	○	○
STC.L GBR, @-Rn	0100nnnn00010011	Rn-4→Rn、GBR→(Rn)	1	—	○	○	○
STC.L VBR, @-Rn	0100nnnn00100011	Rn-4→Rn、VBR→(Rn)	1	—	○	○	○

指 令	操作码	操 作	执行 状态	T 位	适用指令		
					SH2、 SH2E	SH4	SH-2A
STS MACH, Rn	0000nnnn00001010	MACH→Rn	1	—	○	○	○
STS MACL, Rn	0000nnnn00011010	MACL→Rn	1	—	○	○	○
STS PR, Rn	0000nnnn00101010	PR→Rn	1	—	○	○	○
STS.L MACH, @-Rn	0100nnnn00000010	Rn - 4→Rn、MACH→(Rn)	1	—	○	○	○
STS.L MACL, @-Rn	0100nnnn00010010	Rn - 4→Rn、MACL→(Rn)	1	—	○	○	○
STS.L PR, @-Rn	0100nnnn00100010	Rn - 4→Rn、PR→(Rn)	1	—	○	○	○
TRAPA #imm	11000011iiiiiii	PC/SR→堆栈区、 (imm×4+VBR)→PC	5	—	○	○	○

【注】 指令的执行状态：

表中所示的执行状态为最小值，实际根据以下条件，指令执行的状态数会增加。

- (1) 取指令与数据存取产生竞争
- (2) 加载指令（存储器→寄存器）的目标寄存器与紧随其后的指令所用的寄存器相同

* 存储体上溢时，状态数为 19。

2.4.8 浮点运算指令

表 2.17 浮点运算指令

指 令	操作码	操 作	执行 状态	T 位	适用指令		
					SH2E	SH4	SH-2A/ SH2A- FPU
FABS FRn	1111nnnn01011101	FRn →FRn	1	—	○	○	○
FABS DRn	1111nnn001011101	DRn →DRn	1	—		○	○
FADD FRm, FRn	1111nnnnmmmm0000	FRn+FRm→FRn	1	—	○	○	○
FADD DRm, DRn	1111nnn0mmmm0000	DRn+DRm→DRn	6	—		○	○
FCMP/EQ FRm, FRn	1111nnnnmmmm0100	(FRn=FRm)? 1:0→T	1	比较 结果	○	○	○
FCMP/EQ DRm, DRn	1111nnn0mmmm0100	(DRn=DRm)? 1:0→T	2	比较 结果		○	○
FCMP/GT FRm, FRn	1111nnnnmmmm0101	(FRn>FRm)? 1:0→T	1	比较 结果	○	○	○
FCMP/GT DRm, DRn	1111nnn0mmmm0101	(DRn>DRm)? 1:0→T	2	比较 结果		○	○
FCNVDS DRm, FPUL	1111mmm010111101	(float)DRm→FPUL	2	—		○	○
FCNVSD FPUL, DRn	1111nnn010101101	(double)FPUL→DRn	2	—		○	○
FDIV FRm, FRn	1111nnnnmmmm0011	FRn/FRm→FRn	10	—	○	○	○
FDIV DRm, DRn	1111nnn0mmmm0011	DRn/DRm→DRn	23	—		○	○
FLDI0 FRn	1111nnnn10001101	0×00000000→FRn	1	—	○	○	○
FLDI1 FRn	1111nnnn10011101	0×3F800000→FRn	1	—	○	○	○
FLDS FRm, FPUL	1111mmmm00011101	FRm→FPUL	1	—	○	○	○
FLOAT FPUL, FRn	1111nnnn00101101	(float)FPUL→FRn	1	—	○	○	○
FLOAT FPUL, DRn	1111nnn000101101	(double)FPUL→DRn	2	—		○	○
FMAC FR0, FRm, FRn	1111nnnnmmmm1110	FR0×FRm+FRn→FRn	1	—	○	○	○
FMOV FRm, FRn	1111nnnnmmmm1100	FRm→FRn	1	—	○	○	○
FMOV DRm, DRn	1111nnn0mmmm01100	DRm→DRn	2	—		○	○
FMOV.S @(R0, Rm), FRn	1111nnnnmmmm0110	(R0+Rm)→FRn	1	—	○	○	○
FMOV.D @(R0, Rm), DRn	1111nnn0mmmm0110	(R0+Rm)→DRn	2	—		○	○
FMOV.S @Rm+, FRn	1111nnnnmmmm1001	(Rm)→FRn, Rm+=4	1	—	○	○	○
FMOV.D @Rm+, DRn	1111nnn0mmmm1001	(Rm)→DRn, Rm+=8	2	—		○	○
FMOV.S @Rm, FRn	1111nnnnmmmm1000	(Rm)→FRn	1	—	○	○	○
FMOV.D @Rm, DRn	1111nnn0mmmm1000	(Rm)→DRn	2	—		○	○
FMOV.S @(disp12, Rm), FRn	0011nnnnmmmm0001 0111dddddddddd	(disp×4+Rm)→FRn	1	—			○

指 令	操作码	操 作	执行 状态	T 位	适用指令		
					SH2E	SH4	SH-2A/ SH2A- FPU
FMOV.D @(disp12,Rn),DRn	0011nnnn0mmm0001 0111ddddddddddd	(disp×8+Rn)→DRn	2	—			○
FMOV.S FRm, @(R0,Rn)	1111nnnnmmmm0111	FRm→(R0+Rn)	1	—	○	○	○
FMOV.D DRm, @(R0,Rn)	1111nnnnmmmm0011	DRm→(R0+Rn)	2	—		○	○
FMOV.S FRm, @-Rn	1111nnnnmmmm1011	Rn – =4、FRm→(Rn)	1	—	○	○	○
FMOV.D DRm, @-Rn	1111nnnnmmmm0101	Rn – =8、DRm→(Rn)	2	—		○	○
FMOV.S FRm, @Rn	1111nnnnmmmm1010	FRm→(Rn)	1	—	○	○	○
FMOV.D DRm, @Rn	1111nnnnmmmm0100	DRm→(Rn)	2	—		○	○
FMOV.S FRm,@(disp12,Rn)	0011nnnnmmmm0001 0011ddddddddddd	FRm→(disp×4+Rn)	1	—			○
FMOV.D DRm,@(disp12,Rn)	0011nnnnmmmm0000 0011ddddddddddd	DRm→(disp×8+Rn)	2	—			○
FMUL FRm, FRn	1111nnnnmmmm0010	FRn × FRm→FRn	1	—	○	○	○
FMUL DRm, DRn	1111nnnn0mmm00010	DRn × DRm→DRn	6	—		○	○
FNEG FRn	1111nnnn01001101	– FRn→FRn	1	—	○	○	○
FNEG DRn	1111nnnn001001101	– DRn→DRn	1	—		○	○
FSCHG	1111001111111101	FPSCR.SZ= ~ FPSCR.SZ	1	—		○	○
FSQRT FRn	1111nnnn01101101	$\sqrt{\text{FRn}}$ →FRn	9	—		○	○
FSQRT DRn	1111nnnn001101101	$\sqrt{\text{DRn}}$ →DRn	22	—		○	○
FSTS FPUL,FRn	1111nnnn00001101	FPUL→FRn	1	—	○	○	○
FSUB FRm, FRn	1111nnnnmmmm0001	FRn– FRm→FRn	1	—	○	○	○
FSUB DRm, DRn	1111nnnn0mmm00001	DRn– DRm→DRn	6	—		○	○
FTRC FRm, FPUL	1111mmmm00111101	(long)FRm→FPUL	1	—	○	○	○
FTRC DRm, FPUL	1111mmmm000111101	(long)DRm→FPUL	2	—		○	○

2.4.9 FPU 相关的 CPU 指令

表 2.18 FPU 相关的 CPU 指令

指 令	操作码	操 作	执行 状态	T 位	适用指令		
					SH2E	SH4	SH-2A/ SH2A- FPU
LDS Rm,FPSCR	0100mmmm01101010	Rm→FPSCR	1	—	○	○	○
LDS Rm,FPUL	0100mmmm01011010	Rm→FPUL	1	—	○	○	○
LDS.L @Rm+, FPSCR	0100mmmm01100110	(Rm)→FPSCR、Rm+=4	1	—	○	○	○
LDS.L @Rm+, FPUL	0100mmmm01010110	(Rm)→FPUL、Rm+=4	1	—	○	○	○
STS FPSCR, Rn	0000nnnn01101010	FPSCR→Rn	1	—	○	○	○
STS FPUL, Rn	0000nnnn01011010	FPUL→Rn	1	—	○	○	○
STS.L FPSCR, @-Rn	0100nnnn01100010	Rn -= 4、FPSCR→(Rn)	1	—	○	○	○
STS.L FPUL, @-Rn	0100nnnn01010010	Rn -= 4、FPUL→(Rn)	1	—	○	○	○

2.4.10 位操作指令

表 2.19 位操作指令

指 令	操作码	操 作	执行 状态	T 位	适用指令		
					SH2、 SH2E	SH4	SH-2A
BAND.B #imm3,@(disp12,Rn)	0011nnnn0iii1001 0100ddddddddddd	(imm of (disp+Rn))&T→T	3	运算 结果			○
BANDNOT.B #imm3,@(disp12,Rn)	0011nnnn0iii1001 1100ddddddddddd	~ (imm of (disp+ Rn))&T → T	3	运算 结果			○
BCLR.B #imm3,@(disp12,Rn)	0011nnnn0iii1001 0000ddddddddddd	0→(imm of (disp+Rn))	3	—			○
BCLR #imm3, Rn	10000110nnnn0iii	0→imm of Rn	1	—			○
BLD.B #imm3,@(disp12,Rn)	0011nnnn0iii1001 0011ddddddddddd	(imm of (disp+Rn))→T	3	运算 结果			○
BLD #imm3, Rn	10000111nnnn1iii	imm of Rn →T	1	运算 结果			○
BLDNOT.B #imm3,@(disp12,Rn)	0011nnnn0iii1001 1011ddddddddddd	~ (imm of (disp+Rn)) →T	3	运算 结果			○
BOR.B #imm3,@(disp12,Rn)	0011nnnn0iii1001 0101ddddddddddd	(imm of (disp+ Rn)) T →T	3	运算 结果			○
BORNOT.B #imm3,@(disp12,Rn)	0011nnnn0iii1001 1101ddddddddddd	~ (imm of (disp+ Rn)) T →T	3	运算 结果			○
BSET.B #imm3,@(disp12,Rn)	0011nnnn0iii1001 0001ddddddddddd	1→ (imm of (disp+Rn))	3	—			○
BSET #imm3, Rn	10000110nnnn1iii	1→imm of Rn	1	—			○
BST.B #imm3,@(disp12,Rn)	0011nnnn0iii1001 0010ddddddddddd	T→(imm of (disp+Rn))	3	—			○
BST #imm3, Rn	10000111nnnn0iii	T→imm of Rn	1	—			○
BXOR.B #imm3,@(disp12,Rn)	0011nnnn0iii1001 0110ddddddddddd	(imm of (disp+ Rn)) ^ T →T	3	运算 结果			○

2.5 处理状态

CPU 的处理状态有复位状态、异常处理状态、总线权释放状态、程序执行状态及低功耗状态等 5 种。各状态间的转移如图 2.6 所示。

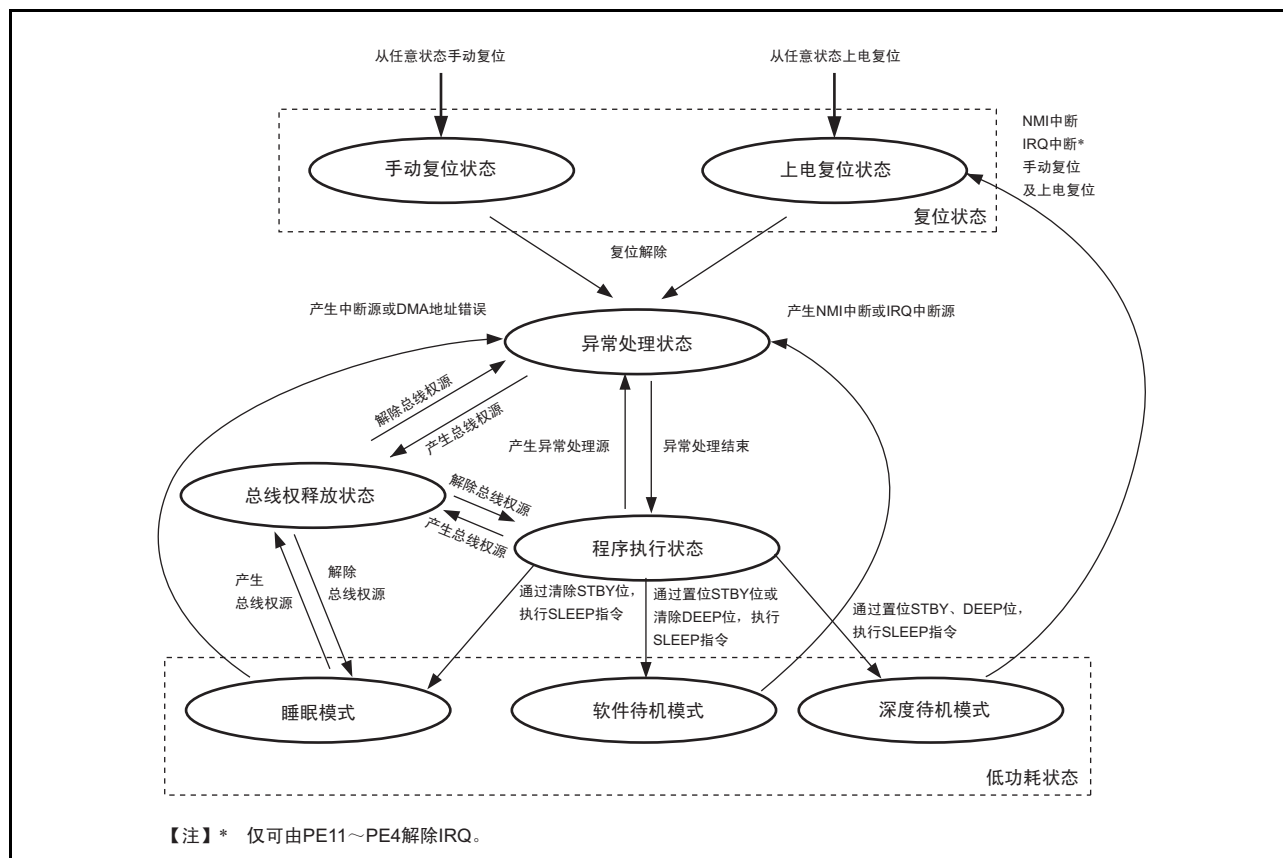


图 2.6 处理状态转移图

(1) 复位状态

CPU 复位的状态。复位有上电复位与手动复位 2 种。

(2) 异常处理状态

CPU 因复位或中断等异常处理源而改变处理状态流程时的过渡状态。

复位时，从异常处理向量表取出并分别保存执行起始地址（程序计数器 (PC) 的初始值）与堆栈指针 (SP) 的初始值，转移至起始地址后开始执行程序。

产生中断等时，参照 SP，将 PC 与状态寄存器 (SR) 压栈。从异常处理向量表取出异常服务程序的起始地址，转移至该地址后开始执行程序。

之后，处理状态变为程序执行状态。

(3) 程序执行状态

CPU 依次执行程序的状态。

(4) 低功耗状态

CPU 停止运行、处于低功耗的状态。通过睡眠指令进入睡眠模式、软件待机模式或深度待机模式。

(5) 总线权释放状态

CPU 向请求总线权的器件释放总线的状态。

第 3 章 浮点单元 (FPU)

3.1 特点

FPU 具有以下特点：

- 以IEEE754标准为依据
- 16个单精度浮点寄存器（也可被参照为8个双精度寄存器）
- 2种舍入模式：就近舍入及向0舍入
- 非规格化数处理模式：舍入为0
- 5个异常源：
无效运算、被0除、上溢、下溢、不正确
- 包含指令：
单精度、双精度、系统控制

3.2 数据格式

3.2.1 浮点格式

浮点由以下 3 个字段组成：

- 符号 (s)
- 指数 (e)
- 小数 (f)

本 LSI 可使用图 3.1 与图 3.2 所示的格式，处理单精度、双精度浮点。

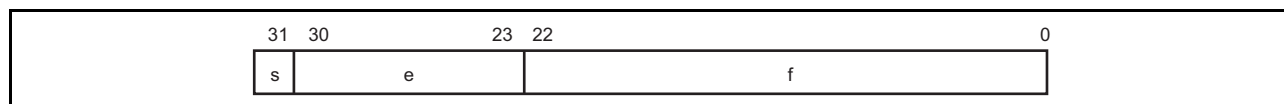


图 3.1 单精度浮点格式

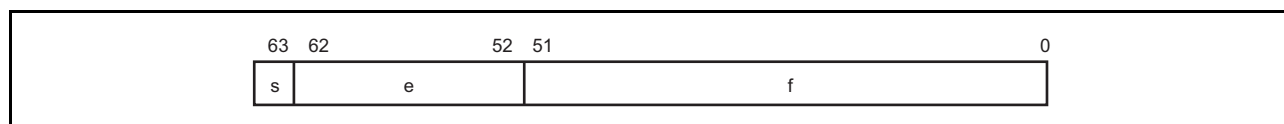


图 3.2 双精度浮点格式

用如下带偏差的算式表示指数：

$$e = E + \text{bias}$$

无偏差的指数 E 的范围为 $E_{\min}-1 \sim E_{\max}+1$ 。 $E_{\min}-1$ 的值与 $E_{\max}+1$ 的值区别如下： $E_{\min}-1$ 表示 0（正 / 负符号）与非规格化数； $E_{\max}+1$ 表示正 / 负无穷大或非数 (NaN)。 $E_{\min}$ 与 $E_{\max}$ 的值如表 3.1 所示。

表 3.1 浮点的格式与参数

参数	单精度	双精度
总的位宽度	32 位	64 位
符号位	1 位	1 位
指数字段	8 位	11 位
小数字段	23 位	52 位
精度	24 位	53 位
偏差	+ 127	+ 1023
$E_{\max}$	+ 127	+ 1023
$E_{\min}$	- 126	- 1022

浮点的数值 v 取决于以下情况：

$E=E_{\max}+1$ 且 $f \neq 0$ 时， v 与符号 s 无关，为非数 (NaN)。

$E=E_{\max}+1$ 且 $f=0$ 时， v 为 $(-1)^s$ (无穷大) “正 / 负无穷大”。

$E_{\min} \leq E \leq E_{\max}$ 时， v 为 $(-1)^s 2^E (1.f)$ “规格化数”。

$E=E_{\min}-1$ 且 $f \neq 0$ 时， v 为 $(-1)^s 2^{E_{\min}} (0.f)$ “非规格化数”。

$E=E_{\min}-1$ 且 $f=0$ 时， v 为 $(-1)^s 0$ “正 / 负 0”。

16 进制中各数的范围如表 3.2 所示。

表 3.2 浮点的范围

类型	单精度	双精度
信令非数	H'7FFF FFFF ~ H'7FC0 0000	H'7FFF FFFF FFFF FFFF ~ H'7FF8 0000 0000 0000
静态非数	H'7FBF FFFF ~ H'7F80 0001	H'7FF7 FFFF FFFF FFFF ~ H'7FF0 0000 0000 0001
正无穷大	H'7F80 0000	H'7FF0 0000 0000 0000
正的规格化数	H'7F7F FFFF ~ H'0080 0000	H'7FEF FFFF FFFF FFFF ~ H'0010 0000 0000 0000
正的非规格化数	H'007F FFFF ~ H'0000 0001	H'000F FFFF FFFF FFFF ~ H'0000 0000 0000 0001
正 0	H'0000 0000	H'0000 0000 0000 0000
负 0	H'8000 0000	H'8000 0000 0000 0000
负的非规格化数	H'8000 0001 ~ H'807F FFFF	H'8000 0000 0000 0001 ~ H'800F FFFF FFFF FFFF
负的规格化数	H'8080 0000 ~ H'FF7F FFFF	H'8010 0000 0000 0000 ~ H'FFEF FFFF FFFF FFFF
负无穷大	H'FF80 0000	H'FFF0 0000 0000 0000
静态非数	H'FF80 0001 ~ H'FFBF FFFF	H'FFF0 0000 0000 0001 ~ H'FFF7 FFFF FFFF FFFF
信令非数	H'FFC0 0000 ~ H'FFFF FFFF	H'FFF8 0000 0000 0000 ~ H'FFFF FFFF FFFF FFFF

3.2.2 非数 (NaN)

非数 (NaN) 的位模式如图 3.3 所示。以下情况的值为 NaN:

- 符号位: Don't care
- 指数字段: 所有位均为 1
- 小数字段: 至少 1 位为 1

小数字段的 MSB 为 1 时, NaN 为信令非数 (sNaN); 为 0 时, NaN 为静态非数 (qNaN)。

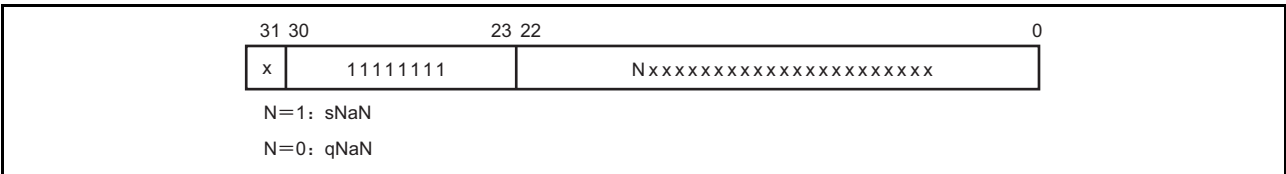


图 3.3 单精度的 NaN 位模式

在生成除复制、FABS 或 FNEG 之外的浮点值的运算中输入 sNaN。

- FPSCR 的 EN.V 位为 0 时, 运算结果 (输出) 为 qNaN。
- FPSCR 的 EN.V 位为 1 时, 由无效运算异常产生 FPU 异常处理, 此时不更改运算目标寄存器的内容。

在生成浮点值的运算中输入 qNaN。不输入 sNaN 时, 与 FPSCR 的 EN.V 位的设定无关, 总是输出 qNaN, 此时不产生异常。

作为运算结果生成的 qNaN 的值如下:

- 单精度 qNaN: H'7FBF FFFF
- 双精度 qNaN: H'7FF7 FFFF FFFF FFFF

有关输入非数 (NaN) 时的浮点运算, 详情参阅各指令的说明。

3.2.3 非规格化数

非规格化数的浮点值, 指数字段用 0、小数字段用除 0 之外的值表示。

SH2A-FPU 的状态寄存器 FPSCR 的 DN 位总是为 1, 因此在生成 (除复制、FNEG 及 FABS 之外的运算) 值的浮点运算中, 非规格化数 (源操作数或运算结果) 总是被清 0。

有关输入非规格化数时的浮点运算, 详情参阅各指令的说明。

3.3 寄存器说明

3.3.1 浮点寄存器

浮点寄存器的结构如图 3.4 所示，有 FPR0 ~ FPR15 等 16 个 32 位浮点寄存器。这些寄存器被参照为 FR0 ~ FR15、DR0/2/4/6/8/10/12/14。FPR_n 与参照名称的对应关系取决于 FPSCR 的 PR 位和 SZ 位，参照图 3.4。

- 1. 浮点寄存器：FPR_i (16 个寄存器)
FPR0、FPR1、FPR2、FPR3、FPR4、FPR5、FPR6、FPR7、
FPR8、FPR9、FPR10、FPR11、FPR12、FPR13、FPR14、FPR15
- 2. 单精度浮点寄存器：FR_i (16 个寄存器)
FR0 ~ FR15 分配给 FPR0 ~ FPR15。
- 3. 双精度浮点寄存器或单精度浮点寄存器对 DR_i (8 个寄存器)
DR 寄存器由 2 个 FR 寄存器构成。
DR0={FR0, FR1}、DR2={FR2, FR3}、DR4={FR4, FR5}、DR6={FR6, FR7}、
DR8={FR8, FR9}、DR10={FR10, FR11}、DR12={FR12, FR13}、DR14={FR14, FR15}

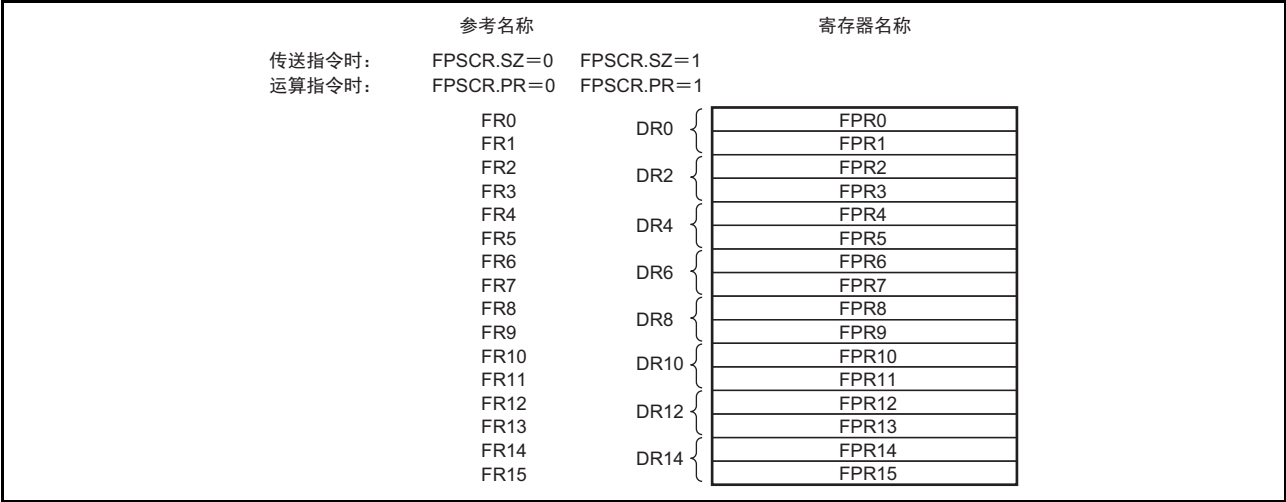


图 3.4 浮点寄存器

3.3.2 浮点状态 / 控制寄存器 (FPSCR)

FPSCR 为 32 位寄存器，可控制浮点指令、设定 FPU 异常及选择舍入的使用方法。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	QIS	-	SZ	PR	DN	Cause	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R	R/W	R/W

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Cause				Enable						Flag				RM1	RM0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
31 ~ 23	—	均为 0	R	保留位 读取值、写入值总是为 0。
22	QIS	0	R/W	非数处理模式 0: 直接处理 qNaN 或 $\pm\infty$ 。 1: qNaN 或 $\pm\infty$ 的处理与 sNaN 相同 (仅在 FPSCR 允许 V=1 时有效)。
21	—	0	R	保留位 读取值、写入值总是为 0。
20	SZ	0	R/W	传送长度模式 0: FMOV 指令的数据长度为 32 位。 1: FMOV 指令的数据长度为 32 位对 (64 位)。
19	PR	0	R/W	精度模式 0: 以单精度运算执行浮点指令。 1: 以双精度运算执行浮点指令。
18	DN	1	R	非规格化模式 (在 SH2A-FPU, 总是固定为 1) 1: 非规格化数作为 0 处理。
17 ~ 12	Cause	均为 0	R/W	FPU 异常源字段 FPU 异常允许字段 FPU 异常标志字段 执行浮点运算指令时, 首先将 FPU 异常源字段清 0。其次, 通过浮点运算产生 FPU 异常时, 对应 FPU 异常源字段及 FPU 异常标志字段的位置 1。由软件将 FPU 异常标志字段位清 0 之前, 保持值为 1。 FPU 异常允许字段的对应位置 1 时, 产生 FPU 异常处理。 各字段位的分配情况, 参照表 3.3。
11 ~ 7	Enable	均为 0	R/W	
6 ~ 2	Flag	均为 0	R/W	
1 0	RM1 RM0	0 1	R/W R/W	舍入模式 选择舍入方法。 00: 就近舍入 01: 向 0 舍入 10: 保留 11: 保留

表 3.3 FPU 异常处理相关位的分配

		FPU 错误 (E)	无效运算 (V)	被 0 除 (Z)	上溢 (O)	下溢 (U)	不正确 (I)
Cause	FPU 异常源字段	bit17	bit16	bit15	bit14	bit13	bit12
Enable	FPU 异常允许字段	无	bit11	bit10	bit9	bit8	bit7
Flag	FPU 异常标志字段	无	bit6	bit5	bit4	bit3	bit2

【注】 在 SH2A-FPU，不产生 FPU 错误。

3.3.3 浮点通信寄存器 (FPUL)

通过 FPUL 执行 FPU 与 CPU 之间的信息传递。FPUL 为 32 位系统寄存器，通过 LDS、STS 指令从 CPU 存取。例如，将保存于通用寄存器 R1 的整数转换为单精度浮点的处理流程如下：

R1 → (LDS 指令) → FPUL → (单精度 FLOAT 指令) → FR1

3.4 舍入

浮点指令中，从中间结果生成最终运算结果时，执行舍入。FMAC 执行 1 次舍入，FADD、FSUB、FMUL 执行 2 次舍入，舍入的次数不同。因此像 FMAC 这样的组合指令的结果，与仅使用 FADD、FSUB、FMUL 等基本指令的结果不同。

舍入有两种方法，使用哪种方法取决于 FPSCR 的 RM 字段。

FPSCR.RM[1:0] = 00：就近舍入

FPSCR.RM[1:0] = 01：向 0 舍入

(1) 就近舍入

运算结果舍入至可表示的最接近的值，最接近的值有两个时，选择 LSB 为 0 的那一个。

舍入前的值大于等于 $2^{E_{max}}(2 - 2^{-p})$ 时，结果为无穷，符号与舍入前相同。 E_{max} 、 p 为单精度时，结果为 127、24；为双精度时，结果为 1023、53。

(2) 向 0 舍入

将舍入前值的舍入位之后的位舍去。

但舍入前的值大于可表示的最大绝对值时，为可表示的最大绝对值。

3.5 FPU 异常

3.5.1 FPU 异常源

可通过浮点运算指令产生 FPU 异常，该异常源如下：

- FPU 错误 (E) : FPSCR 的 DN 位为 0，且输入非规格化数时 (在 SH2A-FPU，不产生该错误)
- 无效运算 (V) : 无效运算 (如 NaN 输入) 时
- 被 0 除 (Z) : 除数为 0 的除法
- 上溢 (O) : 运算结果上溢时
- 下溢 (U) : 运算结果下溢时
- 不正确异常 (I) : 产生上溢、下溢或舍入时

FPSCR 的 FPU 异常源字段包含上述 E、V、Z、O、U、I 对应的所有位，FPSCR 的标志及允许字段包含 V、Z、O、U、I 对应的位，但不包含 E 对应的位，因此无法禁止 FPU 错误。

产生 FPU 异常时，FPU 异常源字段对应的位置 1，并对 FPU 异常标志字段对应的位加 1；不产生 FPU 异常时，FPU 异常源字段对应的位清 0，FPU 异常标志字段对应的位不变。

3.5.2 FPU 异常处理

在以下情况产生 FPU 异常处理：

- FPU 错误 (E) : FPSCR 的 DN 位为 0，且输入非规格化数时 (在 SH2A-FPU，不产生该错误)
- 无效运算 (V) : FPSCR 的 Enable 的 V 位为 1，且无效运算时
- 被 0 除 (Z) : FPSCR 的 Enable 的 Z 位为 1，且被 0 除时
- 上溢 (O) : FPSCR 的 Enable 的 O 位为 1，且运算结果可能上溢的指令
- 下溢 (U) : FPSCR 的 Enable 的 U 位为 1，且运算结果可能下溢的指令
- 不正确异常 (I) : FPSCR 的 Enable 的 I 位为 1，且运算结果可能不正确的指令

有关浮点运算可能引起的各项异常处理，详情参阅各指令的说明。浮点运算引起的所有异常事件都作为相同的 FPU 异常事件被分配。通过读取 FPSCR 并解释保存的信息，由软件决定浮点运算引起的异常的含义。产生 FPU 异常处理时，目标寄存器不变。

除此之外，将 V、Z、O、U、I 对应的位置 1，生成作为运算结果的默认值。

- 无效运算 (V) : 结果为生成 qNaN。
- 被 0 除 (Z) : 生成带符号 (与舍入前相同) 的无穷大值。
- 上溢 (O) :
 - 向 0 舍入时，生成带符号 (与舍入前相同) 的最大规格化数。
 - 就近舍入时，生成带符号 (与舍入前相同) 的无穷大值。
- 下溢 (U) : 生成带符号 (与舍入前相同) 的 0。
- 不正确异常 (I) : 生成不正确的结果。

第 4 章 时钟脉冲振荡器 (CPG)

本 LSI 内置时钟脉冲振荡器，生成内部时钟 ($I\phi$)、外围时钟 ($P\phi$) 及总线时钟 ($B\phi$)。时钟脉冲振荡器由晶体振荡器、PLL 电路及分频电路构成。

4.1 特点

- 4种时钟运行模式
根据使用的频率范围、外接晶体谐振器、外部时钟输入或USB时钟输入的不同，有4种时钟运行模式可供选择。
- 3种时钟
可单独生成CPU、高速缓存使用的内部时钟 ($I\phi$)、外围模块使用的外围时钟 ($P\phi$) 及外部总线接口使用的总线时钟 ($B\phi=CKIO$)。
- 频率更改功能
通过CPG内部的PLL (Phase Locked Loop) 电路和分频电路，可单独更改内部时钟与外围时钟的频率。根据频率控制寄存器 (FRQCR) 的设定，由软件更改频率。
- 低功耗模式的控制
可通过睡眠模式、软件待机模式及深度待机模式停止时钟，并可通过模块待机功能停止特定模块。有关低功耗模式的控制，详情参阅“第28章 低功耗模式”。

时钟脉冲振荡器框图如图 4.1 所示。

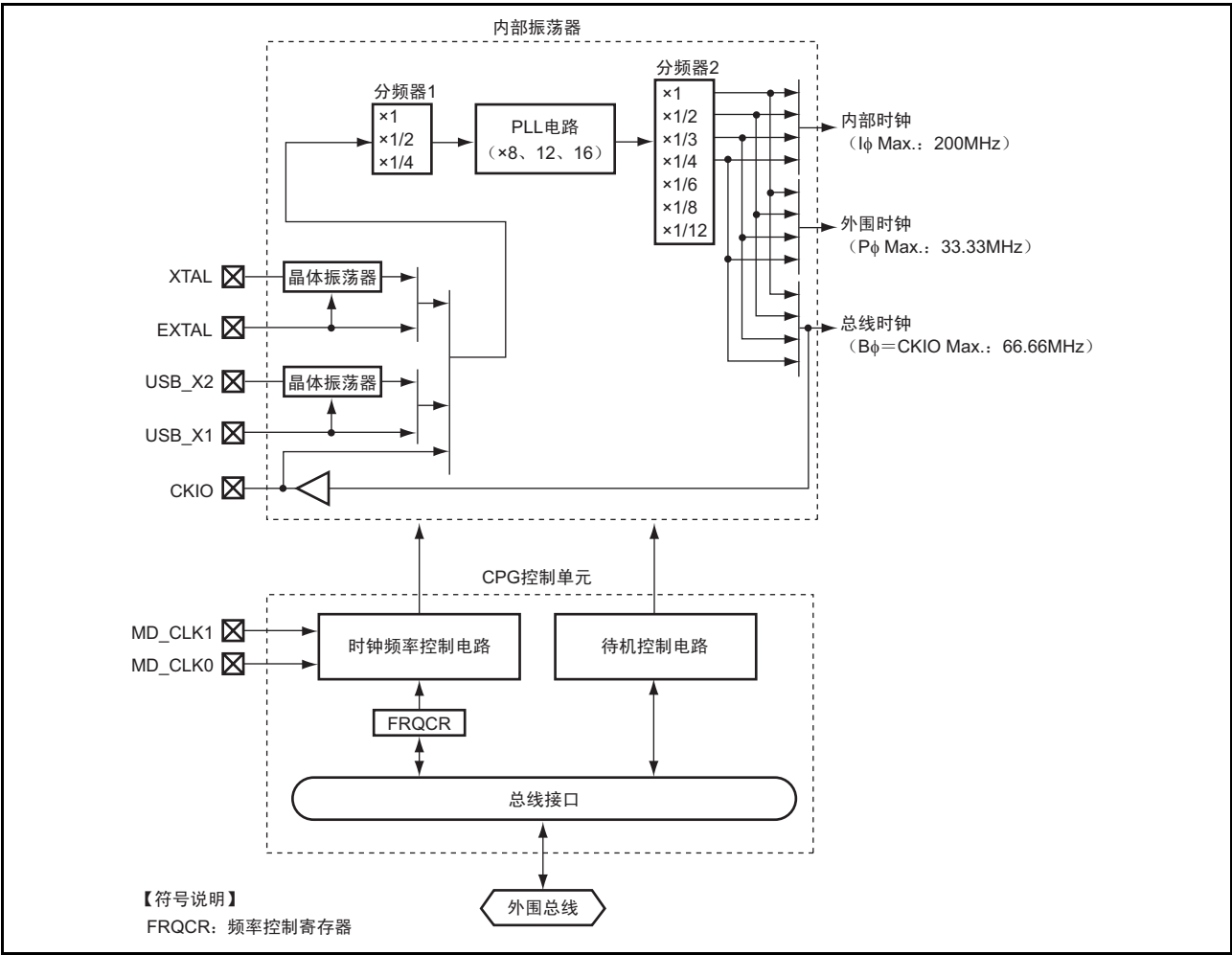


图 4.1 时钟脉冲振荡器框图

时钟脉冲振荡器各块的功能如下：

(1) 晶体振荡器

为将晶体谐振器连接至 XTAL、EXTAL 引脚及 USB_X1、USB_X2 引脚使用的振荡电路。根据时钟运行模式的设定，可使用晶体振荡器。

(2) 分频器 1

分频器 1 可将 EXTAL 引脚、CKIO 引脚的时钟及 USB_X1 输入时钟的频率分频。分频比取决于时钟运行模式的设定。

(3) PLL 电路

PLL 电路可将晶体振荡器或 EXTAL 引脚的输入时钟、CKIO 引脚的时钟及 USB_X1 输入时钟的频率 8 倍频、12 倍频或 16 倍频。

通过频率控制寄存器设定倍频比。此时，控制内部时钟上升沿的相位，使其与 CKIO 引脚的上升沿相位匹配。

输入时钟取决于时钟运行模式。通过 MD_CLK0、MD_CLK1 引脚设定时钟运行模式。有关时钟运行模式，参照表 4.2。

(4) 分频器 2

分频器 2 可生成内部时钟或外围时钟所使用工作频率的时钟。在不低于 CKIO 引脚的时钟频率范围内，工作频率可选择 PLL 电路的输出频率的 1 倍、1/2 倍、1/3 倍、1/4 倍、1/6 倍、1/8 倍或 1/12 倍。通过频率控制寄存器设定分频比。

(5) 时钟频率控制电路

时钟频率控制电路通过 MD_CLK0、MD_CLK1 引脚及频率控制寄存器 (FRQCR)，控制时钟频率。

(6) 待机控制电路

待机控制电路控制时钟转换或睡眠模式、软件待机模式及深度待机模式时内部振荡电路的状态及其他模块的状态。

另外，还有控制其他模块低功耗模式的待机控制寄存器。有关待机控制寄存器，详情参阅“第 28 章 低功耗模式”。

(7) 频率控制寄存器 (FRQCR)

在频率控制寄存器 (FRQCR) 分配软件待机模式时是否从 CKIO 引脚输出时钟的控制位、PLL 电路的倍频比控制位、内部时钟及外围时钟 (Pφ) 分频比的控制位。

4.2 输入 / 输出引脚

时钟脉冲振荡器的引脚结构与功能如表 4.1 所示。

表 4.1 振荡电路的引脚结构与功能

名称	引脚名称	输入 / 输出	功能 (时钟运行模式 0、1)	功能 (时钟运行模式 2)	功能 (时钟运行模式 3)
模式控制引脚	MD_CLK0	输入	设定时钟运行模式。		
	MD_CLK1	输入	设定时钟运行模式。		
晶体输入 / 输出 引脚 (时钟输入 引脚)	XTAL	输出	连接晶体谐振器。 (不使用晶体谐振器 时, 必须将引脚置为开 路)	必须将引脚置为开路。	必须将引脚置为开路。
	EXTAL	输入	连接晶体谐振器或用作 外部时钟输入引脚。	必须固定引脚 (上拉 / 下 拉 / 连接电源 / 接地)。	必须固定引脚 (上拉 / 下拉 / 连接电源 / 接 地)。
时钟输入 / 输出 引脚	CKIO	输入 / 输出	为时钟输出引脚。	为时钟输入引脚。	为时钟输出引脚。
USB 晶体输入 / 输出引脚 (时钟 输入引脚)	USB_X1	输入	作为 USB 专用时钟输 入连接晶体谐振器, 或 用作外部时钟输入引 脚。不使用 USB 时, 必须固定引脚 (上拉 / 下拉 / 连接电源 / 接 地)。	作为 USB 专用时钟输入连 接晶体谐振器, 或用作外 部时钟输入引脚。不使用 USB 时, 必须固定引脚 (上拉 / 下拉 / 连接电源 / 接地)。	作为 USB 及本 LSI 的 时钟输入连接晶体谐振 器, 或用作外部时钟输 入。
	USB_X2	输出	连接 USB 晶体谐振 器。 (不使用晶体谐振器 时, 必须将引脚置为开 路)	连接 USB 晶体谐振器。 (不使用晶体谐振器时, 必须将引脚置为开路)	连接 USB 及本 LSI 用 晶体谐振器。 (不使用晶体谐振器 时, 必须将引脚置为开 路)

4.3 时钟运行模式

模式控制引脚 (MD_CLK1、MD_CLK0) 组合与时钟运行模式的关系如表 4.2 所示。

时钟运行模式的可使用频率范围如表 4.3 所示。

表 4.2 时钟运行模式

模式	引脚组合		时钟输入 / 输出		分频器 1	PLL 电路	CKIO 的频率
	MD_CLK1	MD_CLK0	提供源	输出			
0	0	0	EXTAL/ 晶体谐振器	CKIO	1	ON (×8、12、16)	(EXTAL/ 晶体) ×4
1	0	1	EXTAL/ 晶体谐振器	CKIO	1/2	ON (×8、12、16)	(EXTAL/ 晶体) ×2
2	1	0	CKIO	—	1/4	ON (×8、12、16)	(CKIO)
3	1	1	USB_X1/ 晶体谐振器	CKIO	1/4	ON (×8、12、16)	(USB_X1/ 晶体)

- 模式0

在模式0, 从EXTAL引脚或晶体振荡器输入时钟。在PLL电路根据波形成形及频率控制寄存器的设定, 将频率倍频后, 提供给本LSI。EXTAL引脚输入及晶体谐振器均可使用10MHz~16.67MHz的振荡频率。CKIO的频率范围为40MHz~66.66MHz。另外, 为了减少消耗电流, 不使用USB时, 必须固定USB_X1引脚 (上拉/下拉/连接电源/接地), 并将USB_X2引脚置为开路。

- 模式1

在模式1, 从EXTAL引脚或晶体振荡器输入时钟。在PLL电路根据波形成形及频率控制寄存器的设定, 将频率倍频后, 提供给本LSI。EXTAL引脚输入及晶体谐振器均可使用20MHz~33.33MHz的振荡频率。CKIO的频率范围为40MHz~66.66MHz。另外, 为了减少消耗电流, 不使用USB时, 必须固定USB_X1引脚 (上拉/下拉/连接电源/接地), 并将USB_X2引脚置为开路。

- 模式2

在模式2, CKIO引脚为输入引脚, 向此引脚输入外部时钟, 在PLL电路根据波形成形及频率控制寄存器的设定, 将频率倍频后, 提供给本LSI。CKIO的频率范围为40MHz~66.66MHz。另外, 为了减少消耗电流, 在模式2使用时, 必须固定EXTAL引脚 (上拉/下拉/连接电源/接地), 并将XTAL引脚置为开路。不使用USB时, 必须固定USB_X1引脚 (上拉/下拉/连接电源/接地), 并将USB_X2引脚置为开路。

- 模式3

在模式3, 可从USB_X1引脚或晶体振荡器输入时钟。向此引脚输入外部时钟后, 在PLL电路根据波形成形及频率控制寄存器的设定, 将频率倍频后, 提供给本LSI。CKIO的频率为 (USB_X1/ 晶体) (48MHz)。另外, 为了减少消耗电流, 在模式3使用时, 必须固定EXTAL引脚 (上拉/下拉/连接电源/接地), 并将XTAL引脚置为开路。不使用USB晶体时, 必须将USB_X2引脚置为开路。

表 4.3 时钟运行模式与可设定的频率范围

时钟运行模式	FRQCR 寄存器设定值 *1	PLL 倍频比	内部时钟比 (I : B : P) *2	可设定的频率范围 (MHz)				
		PLL 电路		输入时钟 *3	输出时钟 (CKIO 引脚)	内部时钟 (I ϕ)	总线时钟 (B ϕ)	外围时钟 (P ϕ)
0	H'x003	ON ($\times 8$)	8 : 4 : 2	10 ~ 16.67	40 ~ 66.66	80 ~ 133.36	40 ~ 66.66	20 ~ 33.33
	H'x004	ON ($\times 8$)	8 : 4 : 4/3	10 ~ 16.67	40 ~ 66.66	80 ~ 133.36	40 ~ 66.66	13.33 ~ 22.22
	H'x005	ON ($\times 8$)	8 : 4 : 1	10 ~ 16.67	40 ~ 66.66	80 ~ 133.36	40 ~ 66.66	10 ~ 16.67
	H'x006	ON ($\times 8$)	8 : 4 : 2/3	10 ~ 16.67	40 ~ 66.66	80 ~ 133.36	40 ~ 66.66	6.67 ~ 11.11
	H'x104	ON ($\times 12$)	12 : 4 : 2	10 ~ 16.67	40 ~ 66.66	120 ~ 200	40 ~ 66.66	20 ~ 33.33
	H'x106	ON ($\times 12$)	12 : 4 : 1	10 ~ 16.67	40 ~ 66.66	120 ~ 200	40 ~ 66.66	10 ~ 16.67
	H'x205	ON ($\times 16$)	16 : 4 : 2	10 ~ 12.5	40 ~ 50	160 ~ 200	40 ~ 50	20 ~ 25
	H'x206	ON ($\times 16$)	16 : 4 : 4/3	10 ~ 12.5	40 ~ 50	160 ~ 200	40 ~ 50	13.33 ~ 16.67
	H'x215	ON ($\times 16$)	8 : 4 : 2	10 ~ 12.5	40 ~ 50	80 ~ 100	40 ~ 50	20 ~ 25
	H'x216	ON ($\times 16$)	8 : 4 : 4/3	10 ~ 12.5	40 ~ 50	80 ~ 100	40 ~ 50	13.33 ~ 16.67
1	H'x003	ON ($\times 8$)	4 : 2 : 1	20 ~ 33.33	40 ~ 66.66	80 ~ 133.36	40 ~ 66.66	20 ~ 33.33
	H'x004	ON ($\times 8$)	4 : 2 : 2/3	20 ~ 33.33	40 ~ 66.66	80 ~ 133.36	40 ~ 66.66	13.33 ~ 22.22
	H'x005	ON ($\times 8$)	4 : 2 : 1/2	20 ~ 33.33	40 ~ 66.66	80 ~ 133.36	40 ~ 66.66	10 ~ 16.67
	H'x006	ON ($\times 8$)	4 : 2 : 1/3	20 ~ 33.33	40 ~ 66.66	80 ~ 133.36	40 ~ 66.66	6.67 ~ 11.11
	H'x104	ON ($\times 12$)	6 : 2 : 1	20 ~ 33.33	40 ~ 66.66	120 ~ 200.0	40 ~ 66.66	20 ~ 33.33
	H'x106	ON ($\times 12$)	6 : 2 : 1/2	20 ~ 33.33	40 ~ 66.66	120 ~ 200.0	40 ~ 66.66	10 ~ 16.67
	H'x205	ON ($\times 16$)	8 : 2 : 1	20 ~ 25	40 ~ 50	160 ~ 200	40 ~ 50	20 ~ 25
	H'x206	ON ($\times 16$)	8 : 2 : 2/3	20 ~ 25	40 ~ 50	160 ~ 200	40 ~ 50	13.33 ~ 16.67
	H'x215	ON ($\times 16$)	4 : 2 : 1	20 ~ 25	40 ~ 50	80 ~ 100	40 ~ 50	20 ~ 25
	H'x216	ON ($\times 16$)	4 : 2 : 2/3	20 ~ 25	40 ~ 50	80 ~ 100	40 ~ 50	13.33 ~ 16.67
2	H'x003	ON ($\times 8$)	2 : 1 : 1/2	40 ~ 66.66	—	80 ~ 133.36	40 ~ 66.66	20 ~ 33.33
	H'x004	ON ($\times 8$)	2 : 1 : 1/3	40 ~ 66.66	—	80 ~ 133.36	40 ~ 66.66	13.33 ~ 22.22
	H'x005	ON ($\times 8$)	2 : 1 : 1/4	40 ~ 66.66	—	80 ~ 133.36	40 ~ 66.66	10 ~ 16.67
	H'x006	ON ($\times 8$)	2 : 1 : 1/6	40 ~ 66.66	—	80 ~ 133.36	40 ~ 66.66	6.67 ~ 11.11
	H'x104	ON ($\times 12$)	3 : 1 : 1/2	40 ~ 66.66	—	120 ~ 200.0	40 ~ 66.66	20 ~ 33.33
	H'x106	ON ($\times 12$)	3 : 1 : 1/4	40 ~ 66.66	—	120 ~ 200.0	40 ~ 66.66	10 ~ 16.67
	H'x205	ON ($\times 16$)	4 : 1 : 1/2	40 ~ 50	—	160 ~ 200	40 ~ 50	20 ~ 25
	H'x206	ON ($\times 16$)	4 : 1 : 1/3	40 ~ 50	—	160 ~ 200	40 ~ 50	13.33 ~ 16.67
	H'x215	ON ($\times 16$)	2 : 1 : 1/2	40 ~ 50	—	80 ~ 100	40 ~ 50	20 ~ 25
	H'x216	ON ($\times 16$)	2 : 1 : 1/3	40 ~ 50	—	80 ~ 100	40 ~ 50	13.33 ~ 16.67

时钟运行模式	FRQCR 寄存器设定值 *1	PLL 倍频比	内部时钟比 (I : B : P) *2	可设定的频率范围 (MHz)				
		PLL 电路		输入时钟 *3	输出时钟 (CKIO 引脚)	内部时钟 (I ϕ)	总线时钟 (B ϕ)	外围时钟 (P ϕ)
3	H'x003	ON ($\times 8$)	2 : 1 : 1/2	48	48	96	48	24
	H'x004	ON ($\times 8$)	2 : 1 : 1/3	48	48	96	48	16
	H'x005	ON ($\times 8$)	2 : 1 : 1/4	48	48	96	48	12
	H'x006	ON ($\times 8$)	2 : 1 : 1/6	48	48	96	48	8
	H'x104	ON ($\times 12$)	3 : 1 : 1/2	48	48	144	48	24
	H'x106	ON ($\times 12$)	3 : 1 : 1/4	48	48	144	48	12
	H'x205	ON ($\times 16$)	4 : 1 : 1/2	48	48	192	48	24
	H'x206	ON ($\times 16$)	4 : 1 : 1/3	48	48	192	48	16
	H'x215	ON ($\times 16$)	2 : 1 : 1/2	48	48	96	48	24
	H'x216	ON ($\times 16$)	2 : 1 : 1/3	48	48	96	48	16

【注】 *1 FRQCR 寄存器的设定值 x 取决于 bit12、13 的设定值。

*2 输入时钟频率为 1 时的时钟比。

*3 在模式 0、1，为 EXTAL 引脚的时钟输入或晶体谐振器的频率。

在模式 2，为 CKIO 引脚的时钟输入频率。

在模式 3，为 USB_X1 引脚的时钟输入或晶体谐振器的频率。

【注意事项】

- 内部时钟的频率，在模式 0，为 EXTAL 引脚的频率与 PLL 电路的倍频比及分频器 1 的分频比的乘积；在模式 1，为 EXTAL 引脚的频率 $\times 1/2$ 后，与 PLL 电路的倍频比及分频器 1 的分频比的乘积；在模式 2，为 CKIO 引脚的频率 $\times 1/4$ 后，与 PLL 电路的倍频比及分频器 1 的分频比的乘积；在模式 3，为 USB_X1 引脚的频率 $\times 1/4$ 后，与 PLL 电路的倍频比及分频器 1 的分频比的乘积。
内部时钟的频率不得低于 CKIO 引脚的频率。
- 外围时钟的频率，在模式 0，为 EXTAL 引脚的频率与 PLL 电路的倍频比及分频器 1 的分频比的乘积；在模式 1，为 EXTAL 引脚的频率 $\times 1/2$ 后，与 PLL 电路的倍频比及分频器 1 的分频比的乘积；在模式 2，为 CKIO 引脚的频率 $\times 1/4$ 后，与 PLL 电路的倍频比及分频器 1 的分频比的乘积；在模式 3，为 USB_X1 引脚的频率 $\times 1/4$ 后，与 PLL 电路的倍频比及分频器 1 的分频比的乘积。
外围时钟的频率必须小于等于 33.33MHz。同时不得高于 CKIO 引脚的频率。
- PLL 电路的倍频比可选择 $\times 8$ 、 $\times 12$ 或 $\times 16$ ；分频器 1 的分频比可选择 $\times 1$ 、 $\times 1/2$ 、 $\times 1/3$ 、 $\times 1/4$ 、 $\times 1/6$ 、 $\times 1/8$ 或 $\times 1/12$ 。以上分别通过频率控制寄存器 (FRQCR) 设定。
- PLL 电路的输出频率，在模式 0，为 EXTAL 引脚的频率与 PLL 电路的倍频比的乘积；在模式 1，为 EXTAL 引脚的频率 $\times 1/2$ 后，与 PLL 电路的倍频比的乘积；在模式 2、模式 3，为 CKIO 引脚或 USB_X1 引脚的频率 $\times 1/4$ 后，与 PLL 电路的倍频比的乘积。该频率必须小于等于 200MHz。

4.4 寄存器说明

时钟脉冲振荡器有以下寄存器。

表 4.4 寄存器结构

寄存器名称	简称	R/W	初始值	地址	存取长度
频率控制寄存器	FRQCR	R/W	H'0003	H'FFFE0010	16

4.4.1 频率控制寄存器 (FRQCR)

FRQCR 为 16 位可读取 / 写入的寄存器，可在普通运行、总线权释放、软件待机模式及待机解除时，指定是否从 CKIO 引脚输出时钟、并指定 PLL 电路的倍频比、内部时钟及外围时钟 (Pφ) 的分频比。FRQCR 只可进行字存取。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	CKOEN2	CKOEN[1:0]	-	-	STC[1:0]	-	-	-	-	IFC	-	PFC[2:0]	-	-	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W:	R	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R	R/W	R	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明															
15	—	0	R	保留位 读取值、写入值总是为 0。															
14	CKOEN2	0	R/W	时钟输出允许 2 CKOEN2 指定更改 PLL 电路的倍频比时，从 CKIO 引脚输出时钟或将 CKIO 引脚固定为低电平。 设定 1 时，在更改 PLL 电路的倍频比期间，CKIO 引脚固定为低电平。由此，可防止更改 PLL 电路的倍频比时，因 CKIO 时钟不稳定而导致外部电路误动作。在时钟运行模式 2，与此位的值无关，CKIO 引脚为输入。 0: 输出时钟 1: 输出低电平															
13、12	CKOEN[1:0]	00	R/W	时钟输出允许 CKOEN[1:0] 指定普通运行、总线释放、待机模式及待机解除时，从 CKIO 引脚输出时钟、固定 CKIO 引脚电平或设置为高阻抗状态（Hi-Z）。 设定 01 时，在待机模式及待机解除期间，CKIO 引脚固定为低电平。由此，可防止待机解除时，因 CKIO 时钟不稳定而导致外部电路误动作。在时钟运行模式 2，与此位的值无关，CKIO 引脚为输入。深度待机模式时，保持普通运行时的状态。 <table><tr><td>普通运行时</td><td>总线权释放时</td><td>待机模式时</td></tr><tr><td>00：输出</td><td>输出关闭（Hi-Z）</td><td>输出关闭（Hi-Z）</td></tr><tr><td>01：输出</td><td>输出</td><td>输出低电平</td></tr><tr><td>10：输出</td><td>输出</td><td>输出（输出不稳定的时钟）</td></tr><tr><td>11：输出关闭（Hi-Z）</td><td>输出关闭（Hi-Z）</td><td>输出关闭（Hi-Z）</td></tr></table>	普通运行时	总线权释放时	待机模式时	00：输出	输出关闭（Hi-Z）	输出关闭（Hi-Z）	01：输出	输出	输出低电平	10：输出	输出	输出（输出不稳定的时钟）	11：输出关闭（Hi-Z）	输出关闭（Hi-Z）	输出关闭（Hi-Z）
普通运行时	总线权释放时	待机模式时																	
00：输出	输出关闭（Hi-Z）	输出关闭（Hi-Z）																	
01：输出	输出	输出低电平																	
10：输出	输出	输出（输出不稳定的时钟）																	
11：输出关闭（Hi-Z）	输出关闭（Hi-Z）	输出关闭（Hi-Z）																	

位	位名称	初始值	R/W	说 明
11、10	—	均为 0	R	保留位 读取值、写入值总是为 0。
9、8	STC[1:0]	00	R/W	PLL 电路的倍频比 00: ×8 倍 01: ×12 倍 10: ×16 倍 11: 保留 (禁止设定)
7 ~ 5	—	均为 0	R	保留位 读取值、写入值总是为 0。
4	IFC	0	R/W	内部时钟频率的分频比 指定对应 PLL 电路输出频率的内部时钟分频比。 0: ×1 倍 1: ×1/2 倍
3	—	0	R	保留位 读取值、写入值总是为 0。
2 ~ 0	PFC[2:0]	011	R/W	外围时钟频率的分频比 指定对应 PLL 电路输出频率的外围时钟分频比。 000: 保留 (禁止设定) 001: 保留 (禁止设定) 010: 保留 (禁止设定) 011: ×1/4 倍 100: ×1/6 倍 101: ×1/8 倍 110: ×1/12 倍

4.5 频率更改方法

可通过改变 PLL 电路的倍频比与分频器的分频比, 更改内部时钟 ($I\phi$) 及外围时钟 ($P\phi$) 的频率。由软件通过频率控制寄存器 (FRQCR) 控制这些频率。方法如下:

4.5.1 倍频比的更改

更改 PLL 电路的倍频比时, 需要振荡稳定时间。通过内部 WDT 对稳定时间进行计数。振荡稳定时间与解除软件待机模式的时间相同。

1. 在初始状态, PLL 电路的倍频比为 8。
2. 给 WDT 设定可满足指定振荡稳定时间的值, 停止 WDT。需要如下设定:
WTCR.TME=0: WDT 停止
WTCR.CKS[2:0]: WDT 计数时钟的分频比
WTCNT: 计数器的初始值
(WDT 通过设定后的时钟递增计数。)
3. 给 STC[1:0] 设定目标值。同时也可给 IFC、PFC[2:0] 设定分频比。
4. 本 LSI 内部暂时停止, 并开始 WDT 的递增计数。停止内部时钟与外围时钟, 仅向 WDT 提供时钟, 且 CKIO 引脚持续输出时钟。
本状态与软件待机模式状态相同, 根据模块可初始化寄存器。请参阅“30.3 各运行模式的寄存器状态一览表”。
5. 开始提供 WDT 计数上溢时设定的时钟, 本 LSI 重新开始运行。WDT 上溢后停止。

4.5.2 分频比的更改

只更改分频比, 而不同时更改倍频比时, 不执行 WDT 的计数。

1. 在初始状态, IFC=B'0、PFC[2:0]=B'011。
2. 给 IFC、PFC[2:0] 位设定目标值。由于时钟模式及 PLL 电路的倍频比影响, 可设定的值有限。必须注意: 设定错误值会导致本 LSI 产生误动作。
3. 如果设定寄存器的各位 (IFC、PFC[2:0]), 则转换至已设定的时钟。

- 【注】
1. 频率更改后执行 SLEEP 指令时, 读取 3 次频率控制寄存器 (FRQCR) 后, 再执行 SLEEP 指令。
 2. 在 PLL 电路更改倍频比时及软件待机模式解除后的振荡稳定时间内, 时钟模式 0、1、3 时, 输出不稳定的 CKIO。为了防止由此导致的误动作, 必须更改寄存器 FRQCR 的 bit14、13、12。

4.5.3 PLL 振荡电路使用时的注意事项

在连接用于 PLL 的 $PLL_{V_{CC}}$ 与 $PLL_{V_{SS}}$ 的图案中，必须缩短从电路板电源提供引脚的布线长度，并加宽图案宽度，以降低感应系数成分。

PLL 的模拟电源对噪声等很敏感，因此很可能因其他电源的干扰而使整个系统产生误动作。因此，尽量避免在电路板上向本模拟电源与 V_{CC} 、 PV_{CC} 的数字电源提供相同电源。

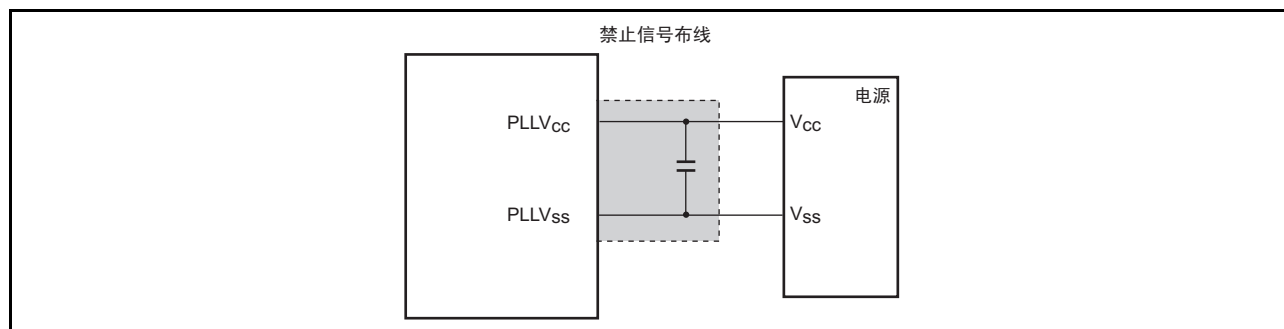


图 4.2 PLL 振荡电路使用时的注意事项

4.6 使用时的注意事项

在时钟模式 0、1、3 使用本 LSI 时，使 $\overline{RES}$ 信号无效后，CKIO 在 1 个周期内输出不稳定时钟。

第 5 章 异常处理

5.1 概要

5.1.1 异常处理的种类与优先顺序

如表 5.1 所示，由复位、地址错误、寄存器存储体错误、中断及指令等各异常源启动异常处理。异常源设置有优先顺序，同时产生多个异常源时，则按此优先顺序接受并处理。

表 5.1 异常源的种类与优先顺序

种类	异常处理	优先顺序
复位	上电复位	高
	手动复位	
地址错误	CPU 地址错误	
	DMAC 地址错误	
指令	FPU 异常	
	整数除法异常（被 0 除）	
	整数除法异常（上溢）	
寄存器存储体错误	存储体下溢	
	存储体上溢	
中断	NMI	
	用户断点	
	H-UDI	
	IRQ	
	PINT	
	内部外围模块	
	直接存储器存取控制器（DMAC）	
	USB2.0 主机 / 功能模块（USB）	
	LCD 控制器（LCDC）	
	比较匹配定时器（CMT）	
	总线状态控制器（BSC）	
	看门狗定时器（WDT）	
	多功能定时器脉冲单元 2（MTU2）	
	A/D 转换器（ADC）	
	I ² C 总线接口 3（IIC3）	
	带 FIFO 的串行通信接口（SCIF）	
	同步串行通信单元（SSU）	
	串行音频接口（SSI）	
	AND/NAND 闪存控制器（FLCTL）	
	实时时钟（RTC）	
	控制器局域网（RCAN-TL1）	
指令	陷阱指令（TRAPA 指令）	低
	一般非法指令（未定义代码）	
	槽非法指令（紧随延迟转移指令 *1 配置的未定义代码（包含 FPU 模块待机时的 FPU 指令及 FPU 相关的 CPU 指令）、改写 PC 的指令 *2、32 位指令 *3、RESBANK 指令、DIVS 指令或 DIVU 指令）	

【注】 *1 延迟转移指令: JMP、JSR、BRA、BSR、RTS、RTE、BF/S、BT/S、BSRF、BRA_F

*2 改写 PC 的指令: JMP、JSR、BRA、BSR、RTS、RTE、BT、BF、TRAPA、BF/S、BT/S、BSRF、BRA_F、JSR/N、RTV/N

*3 32 位指令: BAND.B、BANDNOT.B、BCLR.B、BLD.B、BLDNOT.B、BOR.B、BORNOT.B、BSET.B、BST.B、BXOR.B、MOV.B@disp12、MOV.W@disp12、MOV.L@disp12、MOVI20、MOVI20S、MOVU.B、MOVU.W

5.1.2 异常处理的运行

根据表 5.2 所示的时序检测各异常源，并开始异常处理。

表 5.2 异常源检测与异常处理开始的时序

异常处理		异常源检测及异常处理开始的时序
复位	上电复位	$\overline{\text{RES}}$ 引脚从低电平变为高电平、设置 H-UDI 复位复位有效命令后再设置 H-UDI 无效命令、WDT 上溢等情况时，开始异常处理。
	手动复位	$\overline{\text{MRES}}$ 引脚从低电平变为高电平或 WDT 上溢时，开始异常处理。
地址错误		在指令解码时被检测，并在完成执行的指令后，开始异常处理。
中断		
寄存器存储体错误	存储体下溢	未保存至寄存器存储体时，如果要执行 RESBANK 指令，则开始异常处理。
	存储体上溢	通过中断控制器设定为接受寄存器存储体上溢异常（INTC 的 IBNR 的 BOVE 位 =1），产生使用寄存器存储体的中断且被 CPU 接受时，如果保存至寄存器存储体的全部区域，则开始异常处理。
指令	陷阱指令	通过执行 TRAPA 指令，开始异常处理。
	一般非法指令	除紧随延迟转移指令之后（延迟槽）的未定义代码（包含 FPU 模块待机时的 FPU 指令及 FPU 相关的 CPU 指令）被解码时，开始异常处理。
	槽非法指令	配置于紧随延迟转移指令（延迟槽）之后的未定义代码（包含 FPU 模块待机时的 FPU 指令及 FPU 相关的 CPU 指令）、改写 PC 的指令、32 位指令、RESBANK 指令、DIVS 指令或 DIVU 指令被解码时，开始异常处理。
	整数除法异常	检测出被零除异常或由于用 -1 除负数的最大值（H'80000000）引起的上溢异常时，开始异常处理。
	FPU 异常	通过浮点运算指令的无效运算异常（IEEE754 标准）、被零除异常、上溢、下溢或不正确异常，开始异常处理。另外，FPSCR 的 QIS 位置位时，如果向浮点运算指令源输入 qNaN 或 $\pm \infty$ ，则开始异常处理。

启动异常处理时，CPU 如下运行：

(1) 复位引起的异常处理

从异常处理向量表（上电复位时，PC 与 SP 分别为地址 H'00000000 与地址 H'00000004；手动复位时，PC 与 SP 分别为地址 H'00000008 与地址 H'0000000C）取出程序计数器（PC）与堆栈指针（SP）的初始值。有关异常处理向量表，详情参阅“5.1.3 异常处理向量表”。接着将向量基址寄存器（VBR）初始化为 H'00000000、状态寄存器（SR）的中断屏蔽位（I3～I0）初始化为 H'F（B'1111）、BO 位与 CS 位初始化为 0。中断控制器（INTC）的 IBNR 的 BN 位也初始化为 0。上电复位时，将 FPSCR 初始化为 H'00040001。从异常处理向量表取出的 PC 地址开始执行程序。

(2) 地址错误、寄存器存储体错误、中断及指令引起的异常处理

将 SR 与 PC 保存至 R15 所示的堆栈。在 NMI 及用户断点以外的中断异常处理时，如果执行使用寄存器存储体的设定，则将通用寄存器 R0～R14、控制寄存器 GBR、系统寄存器 MACH、MACL、PR 及所执行中断异常处理的向量表地址偏移量保存至寄存器存储体。执行由地址错误、寄存器存储体错误、NMI 中断、用户断点中断及指令等引起的异常处理时，不保存至寄存器存储体。另外，保存至寄存器存储体的所有存储体时，代替寄存器存储体自动压栈。此时，需在中断控制器设定不接受寄存器存储体上溢异常（INTC 的 IBNR 的 BOVE 位 = 0），否则（INTC 的 IBNR 的 BOVE 位 = 1），会产生寄存器存储体上溢异常。执行中断异常处理时，对 SR 的 I3～I0 位写入中断优先级。地址错误或指令引起异常处理时，不影响 I3～I0 位。从异常处理向量表取出异常服务程序的起始地址，并从该地址开始执行程序。

5.1.3 异常处理向量表

执行异常处理前，需预先在存储器设定异常处理向量表。异常处理向量表保存着异常服务程序的起始地址（复位异常处理表保存着 PC 与 SP 的初始值）。

对各异常源分配不同的向量编号与向量表地址偏移量。由对应的向量编号与向量表地址偏移量计算向量表地址。异常处理时，从该向量表地址所示的异常处理向量表取出异常服务程序的起始地址。

向量编号与向量表地址偏移量如表 5.3 所示，向量表地址的计算方法如表 5.4 所示。

表 5.3 异常处理向量表

异常源		向量编号	向量表地址偏移量
上电复位	PC	0	H'00000000 ～ H'00000003
	SP	1	H'00000004 ～ H'00000007
手动复位	PC	2	H'00000008 ～ H'0000000B
	SP	3	H'0000000C ～ H'0000000F
一般非法指令		4	H'00000010 ～ H'00000013
（系统保留）		5	H'00000014 ～ H'00000017
槽非法指令		6	H'00000018 ～ H'0000001B
（系统保留）		7	H'0000001C ～ H'0000001F
		8	H'00000020 ～ H'00000023
CPU 地址错误		9	H'00000024 ～ H'00000027
DMAC 地址错误		10	H'00000028 ～ H'0000002B
中断	NMI	11	H'0000002C ～ H'0000002F
	用户断点	12	H'00000030 ～ H'00000033

异常源	向量编号	向量表地址偏移量
FPU 异常	13	H'00000034 ~ H'00000037
H-UDI	14	H'00000038 ~ H'0000003B
存储体上溢	15	H'0000003C ~ H'0000003F
存储体下溢	16	H'00000040 ~ H'00000043
整数除法异常（被 0 除）	17	H'00000044 ~ H'00000047
整数除法异常（上溢）	18	H'00000048 ~ H'0000004B
（系统保留）	19	H'0000004C ~ H'0000004F
	：	：
	31	H'0000007C ~ H'0000007F
陷阱指令（用户向量）	32	H'00000080 ~ H'00000083
	：	：
	63	H'000000FC ~ H'000000FF
外部中断（IRQ、PINT）、内部外围模块 *	64	H'00000100 ~ H'00000103
	：	：
	511	H'000007FC ~ H'000007FF

【注】 * 有关外部中断、各内部外围模块中断的向量编号及向量表地址偏移量，参照“第 6 章 中断控制器（INTC）”的表 6.4。

表 5.4 异常处理向量表地址的计算方法

异常源	向量表地址的计算方法
复位	向量表地址 = （向量表地址偏移量） = （向量编号）× 4
地址错误、寄存器存储体错误、中断、指令	向量表地址 = VBR + （向量表地址偏移量） = VBR + （向量号）× 4

- 【注】 1. 向量表地址偏移量：参照表 5.3
2. 向量编号：参照表 5.3

5.2 复位

5.2.1 输入 / 输出引脚

复位相关的引脚结构如表 5.5 所示。

表 5.5 引脚结构

名称	引脚名称	输入 / 输出	功 能
上电复位	$\overline{\text{RES}}$	输入	通过向引脚输入低电平，转移至上电复位处理。
手动复位	$\overline{\text{MRES}}$	输入	通过向引脚输入低电平，转移至手动复位处理。

5.2.2 复位种类

复位是优先顺序最高的异常处理源，有上电复位与手动复位两种。如表 5.6 所示，在上电复位或手动复位时，CPU 均被初始化。FPU 状态在上电复位时被初始化，但在手动复位时不被初始化。另外，上电复位时，初始化内部外围模块的寄存器（除部分寄存器之外），而手动复位时，则被不初始化。

表 5.6 复位状态

种类	转移至复位状态的条件				内部状态	
	RES	H-UDI 命令	MRES	WDT 上溢	CPU	除 CPU 之外的模块
上电复位	低电平	—	—	—	初始化	初始化
	高电平	设置 H-UDI 复位有效命令	—	—	初始化	初始化
	高电平	设置除 H-UDI 复位有效命令之外的命令	—	上电	初始化	*
手动复位	高电平	设置除 H-UDI 复位有效命令之外的命令	低电平	—	初始化	*
	高电平	设置除 H-UDI 复位有效命令之外的命令	高电平	手动	初始化	*

【注】 * 详情参阅“30.3 各运行模式的寄存器状态一览表”。

5.2.3 上电复位

(1) RES 引脚引起的上电复位

如果 RES 引脚设置为低电平，则本 LSI 为上电复位状态。为了确保本 LSI 复位，在上电或软件待机时（时钟停止运行时），RES 引脚必须在振荡稳定时间内保持低电平；时钟运行时，RES 引脚必须在大于等于 20tcyc 的期间保持低电平。在上电复位状态，CPU 的内部状态与内部外围模块的寄存器均被初始化。有关上电复位状态的各引脚状态，详情参阅“附录 A. 引脚状态”。

在上电复位状态，如果 RES 引脚保持一定时间的低电平后，再设置为高电平，则开始上电复位异常处理。此时，CPU 的运行如下：

1. 从异常处理向量表取出程序计数器（PC）的初始值（执行起始地址）。
2. 从异常处理向量表取出堆栈指针（SP）的初始值。
3. 将向量基址寄存器（VBR）清除为 H'00000000、状态寄存器（SR）的中断屏蔽级位（I3～I0）初始化为 H'F（B'1111）、BO 位与 CS 位初始化为 0。另外，将 INTC 的 IBNR 的 BN 位初始化为 0，并将 FPSCR 初始化为 H'00040001。
4. 将从异常处理向量表取出的值分别设定至 PC 与 SP 后，开始执行程序。

另外，必须设置为在接通系统电源时执行上电复位处理。

(2) H-UDI 复位有效命令引起的上电复位

设置 H-UDI 复位有效命令时，本 LSI 进入上电复位状态。H-UDI 复位有效命令与 RES 引脚引起的上电复位相同。通过设置 H-UDI 复位无效命令，可解除上电复位状态。H-UDI 复位有效命令与 H-UDI 复位无效命令之间所需的时间，与为了设置上电复位而保持 RES 引脚为低电平的时间相同。在 H-UDI 复位有效命令引起的上电复位状态下，如果设置 H-UDI 复位无效命令，则开始上电复位异常处理。此时，CPU 的运行与 RES 引脚引起的上电复位运行相同。

(3) WDT 引起的上电复位

设定为在 WDT 看门狗定时器模式产生上电复位，如果 WDT 的 WTCNT 上溢，则为上电复位状态。

此时，通过 WDT 引起的复位信号不能初始化 WDT 的 WRCSR 与 CPG 的 FRQCR。

另外，如果同时产生 $\overline{\text{RES}}$ 引脚、H-UDI 复位有效命令引起的复位与 WDT 上溢引起的复位时，优先进行 $\overline{\text{RES}}$ 引脚、H-UDI 复位有效命令引起的复位，且将 WRCSR 的 WOVF 位清 0。开始执行 WDT 引起的上电复位异常处理时，CPU 的运行与 $\overline{\text{RES}}$ 引脚引起的上电复位运行相同。

5.2.4 手动复位

(1) $\overline{\text{MRES}}$ 引脚引起的手动复位

如果 $\overline{\text{MRES}}$ 引脚设置为低电平，则本 LSI 为手动复位状态。为了确保本 LSI 复位， $\overline{\text{MRES}}$ 引脚必须在大于等于 20tcyc 的期间保持低电平。在手动复位状态，初始化 CPU 的内部状态，而不初始化内部外围模块的寄存器。在手动复位状态，如果 $\overline{\text{MRES}}$ 引脚保持一定时间的低电平后设置为高电平，则开始手动复位异常处理。此时，CPU 的运行如下：

1. 从异常处理向量表取出程序计数器（PC）的初始值（执行起始地址）。
2. 从异常处理向量表取出堆栈指针（SP）的初始值。
3. 将向量基址寄存器（VBR）清除为 H'00000000、状态寄存器（SR）的中断屏蔽级位（I3～I0）初始化为 H'F（B'1111）、BO 位与 CS 位初始化为 0。另外，将 INTC 的 IBNR 的 BN 位初始化为 0。
4. 将从异常处理向量表取出的值分别设定至 PC 与 SP，开始执行程序。

(2) WDT 引起的手动复位

设定为在 WDT 看门狗定时器模式产生手动复位，如果 WDT 的 WTCNT 上溢，则为手动复位状态。

开始执行 WDT 引起的手动复位异常处理时，CPU 的运行与 $\overline{\text{MRES}}$ 引脚引起的手动复位运行相同。

产生手动复位时，保持总线周期。在释放总线权或 DMAC 突发传送过程中产生手动复位时，保留手动复位异常处理直到 CPU 获得总线权为止。手动复位时，初始化 CPU 及 INTC 的 IBNR 的 BN 位，不初始化 FPU 及其他模块。

5.3 地址错误

5.3.1 地址错误的产生源

如表 5.7 所示，取指令、读取 / 写入数据时，产生地址错误。

表 5.7 总线周期与地址错误

总线周期		总线周期的内容	地址错误的产生
种类	总线主控器		
取指令	CPU	从偶数地址取指令	无（正常）
		从奇数地址取指令	产生地址错误
		从内部外围模块空间 *、内部 RAM 空间 * 的 H'F0000000 ~ H'F5FFFFFF 以外取命令	无（正常）
		从内部外围模块空间 *、内部 RAM 空间 * 的 H'F0000000 ~ H'F5FFFFFF 取命令	产生地址错误
读取 / 写入数据	CPU 或 DMAC	从偶数地址存取字数据	无（正常）
		从奇数地址存取字数据	产生地址错误
		从长字边界存取长字数据	无（正常）
		从长字边界以外存取长字数据	产生地址错误
		从双长字边界存取双长字数据	无（正常）
		从双长字边界以外存取双长字数据	产生地址错误
		在内部外围模块空间 * 存取字数据、字节数据	无（正常）
		在 16 位内部外围模块空间 * 存取长字数据	无（正常）
		在 8 位内部外围模块空间 * 存取长字数据	无（正常）

【注】 * 有关内部外围模块空间及内部 RAM 空间，详情参阅“第 9 章 总线状态控制器（BSC）”。

5.3.2 地址错误异常处理

产生地址错误时，结束引起地址错误的总线周期 *，并在执行完当前指令后，开始地址错误异常处理。此时，CPU 的运行如下：

1. 从异常处理向量表取出对应产生地址错误的异常服务程序的起始地址。
2. 将状态寄存器（SR）压栈。
3. 将程序计数器（PC）压栈。保存的 PC 值为最后执行指令的下一条指令的起始地址。
4. 跳转至从异常处理向量表取出的异常服务程序的起始地址后，开始执行程序。此时的跳转不是延迟转移。

【注】 * 由读取 / 写入数据引起的地址错误和由取指令引起的地址错误时，在上述运行结束之前，引起地址错误的总线周期未结束，则 CPU 重新执行地址错误异常处理，直到该总线周期结束。

5.4 寄存器存储体错误

5.4.1 寄存器存储体错误的产生源

(1) 存储体上溢

在中断控制器设定接受寄存器存储体上溢异常（INTC 的 IBNR 的 BOVE 位 = 1），产生使用寄存器存储体的中断，且被 CPU 接受时，保存至寄存器存储体的所有区域的情况下，存储体上溢。

(2) 存储体下溢

未保存至寄存器存储体时，欲执行 RESBANK 指令的情况下，存储体下溢。

5.4.2 寄存器存储体错误异常处理

产生寄存器存储体错误时，产生寄存器存储体错误异常处理。此时，CPU 的运行如下：

1. 从异常处理向量表取出对应的产生寄存器存储体错误的异常服务程序的起始地址。
2. 将状态寄存器（SR）压栈。
3. 将程序计数器（PC）压栈。存储体上溢时，保存的 PC 值为最后执行指令的下一条指令的起始地址；存储体下溢时，保存的 PC 值为已执行 RESBANK 指令的起始地址。
为了防止存储体上溢时产生多重中断，将引起存储体上溢的中断优先级写入状态寄存器（SR）的中断屏蔽级位（I3～I0）。
4. 跳转至从异常处理向量表取出的异常服务程序的起始地址后，开始执行程序。此时的跳转不是延迟转移。

5.5 中断

5.5.1 中断源

如表 5.8 所示，启动中断异常处理的中断源有 NMI、用户断点、H-UDI、IRQ、PINT 及内部外围模块。

表 5.8 中断源

种类	请求源	中断源个数
NMI	NMI 引脚（从外部输入）	1
用户断点	用户断点控制器（UBC）	1
H-UDI	用户调试接口（H-UDI）	1
IRQ	IRQ0 ~ IRQ7 引脚（从外部输入）	8
PINT	PINT0 ~ PINT7 引脚（从外部输入）	8
内部外围模块	直接存储器存取控制器（DMAC）	16
	USB2.0 主机 / 功能模块（USB）	1
	LCD 控制器（LCDC）	1
	比较匹配定时器（CMT）	2
	总线状态控制器（BSC）	1
	看门狗定时器（WDT）	1
	多功能定时器脉冲单元 2（MTU2）	25
	A/D 转换器（ADC）	1
	I ² C 总线接口 3（IIC3）	20
	带 FIFO 的串行通信接口（SCIF）	16
	同步串行通信单元（SSU）	6
	串行音频接口（SSI）	4
	AND/NAND 闪存控制器（FLCTL）	4
	实时时钟（RTC）	3
	控制器局域网（RCAN-TL1）	10

分别对各中断源分配不同的向量编号与向量表偏移量。有关向量编号与向量表地址偏移量，参照“第 6 章 中断控制器（INTC）”的表 6.4。

5.5.2 中断优先顺序

中断源设置有优先顺序，同时产生多个中断时（多重中断），由中断控制器（INTC）判断优先顺序，并按判断结果启动异常处理。

用优先级 0 ~ 16 表示中断源的优先顺序，0 为最低级，16 为最高级。NMI 中断为不可屏蔽的最优先中断（优先级 16），可随时接受。用户断点中断与 H-UDI 的优先级为 15。可通过 INTC 的中断优先级设定寄存器 01、02、05 ~ 17（IPR01、IPR02、IPR05 ~ IPR17）自由设定 IRQ 中断、PINT 中断及内部外围模块中断的优先级（表 5.9）。可设定优先级 0 ~ 15，但不得设定优先级 16。有关 IPR01、IPR02、IPR05 ~ IPR17，详情参阅“6.3.1 中断优先级设定寄存器 01、02、05 ~ 17（IPR01、IPR02、IPR05 ~ IPR17）”。

表 5.9 中断优先顺序

种类	优先级	备注
NMI	16	固定优先级，不可屏蔽
用户断点	15	固定优先级
H-UDI	15	固定优先级
IRQ	0 ~ 15	可通过中断优先级设定寄存器 01、02、05 ~ 17（IPR01、IPR02、IPR05 ~ IPR17）设定
PINT		
内部外围模块		

5.5.3 中断异常处理

产生中断时，由中断控制器（INTC）判断优先顺序。可随时接受 NMI 中断，但其他中断，仅在优先级高于状态寄存器（SR）的中断屏蔽级位（I3 ~ I0）设定的优先级时，才可接受。

接受中断时，开始中断异常处理。中断异常处理时，CPU 从异常处理向量表取出对应已接受中断的异常服务程序的起始地址，并将 SR 与程序计数器（PC）压栈。在 NMI 及用户断点以外的中断异常处理时，如果执行使用寄存器存储体的设定，则将通用寄存器 R0 ~ R14、控制寄存器 GBR、系统寄存器 MACH、MACL、PR 及所执行的异常处理向量表地址偏移量保存至寄存器存储体。执行由地址错误、NMI 中断、用户断点中断及指令引起的异常处理时，不保存至寄存器存储体。另外，保存至寄存器存储体的所有存储体（0 ~ 14）时，代替寄存器存储体自动压栈。此时，需在中断控制器设定不接受寄存器存储体上溢异常（INTC 的 IBNR 的 BOVE 位 = 0），否则（INTC 的 IBNR 的 BOVE 位 = 1），会产生寄存器存储体上溢异常。接着，对 SR 的 I3 ~ I0 位写入已接受中断的优先级值。但是，NMI 时的优先级为 16，而在 I3 ~ I0 位设定的值为 H'F（优先级 15）。之后，跳转至从异常处理向量表取出的中断异常服务程序的起始地址后，开始执行程序。此时的跳转不是延迟转移。有关中断异常处理，详情参阅“6.6 运行说明”。

5.6 指令引起的异常

5.6.1 指令引起的异常种类

如表 5.10 所示，启动异常处理的指令有陷阱指令、槽非法指令、一般非法指令、整数除法异常及 FPU 异常。

表 5.10 指令引起的异常种类

种类	异常源指令	备注
陷阱指令	TRAPA	
槽非法指令	配置于紧随延迟转移指令（延迟槽）之后的未定义代码（包含 FPU 模块待机时的 FPU 指令及 FPU 相关的 CPU 指令）、改写 PC 的指令、32 位指令、RESBANK 指令、DIVS 指令或 DIVU 指令	延迟转移指令：JMP、JSR、BRA、BSR、RTS、RTE、BF/S、BT/S、BSRF、BRA 改写 PC 的指令：JMP、JSR、BRA、BSR、RTS、RTE、BT、BF、TRAPA、BF/S、BT/S、BSRF、BRA、JSR/N、RTV/N 32 位指令：BAND.B、BANDNOT.B、BCLR.B、BLD.B、BLDNOT.B、BOR.B、BORNOT.B、BSET.B、BST.B、BXOR.B、MOV.B@disp12、MOV.W@disp12、MOV.L@disp12、MOVI20、MOVI20S、MOVU.B、MOVU.W
一般非法指令	延迟槽之外的未定义代码（包含 FPU 模块待机时的 FPU 指令及 FPU 相关的 CPU 指令）	
整数除法异常	被 0 除	DIVU、DIVS
	负的最大值 $\div (-1)$	DIVS
FPU 异常	引起 IEEE754 标准定义的无效运算异常或被零除异常的指令、上溢、下溢及可能引起不正确异常的指令	FADD、FSUB、FMUL、FDIV、FMAC、FCMP/EQ、FCMP/GT、FLOAT、FTRC、FCNVDS、FCNVSD、FSQRT

5.6.2 陷阱指令

执行 TRAPA 指令时，开始陷阱指令异常处理。此时，CPU 的运行如下：

1. 从异常处理向量表取出对应 TRAPA 指令指定的向量编号的异常服务程序的起始地址。
2. 将状态寄存器（SR）压栈。
3. 将程序计数器（PC）压栈，保存的 PC 值为 TRAPA 指令的下一条指令的起始地址。
4. 跳转至从异常处理向量表取出的异常服务程序的起始地址后，开始执行程序。此时的跳转不是延迟转移。

5.6.3 槽非法指令

将配置于紧随延迟转移指令之后的指令称为“配置于延迟槽的指令”。配置于延迟槽的指令为未定义代码（包含 FPU 模块待机时的 FPU 指令及 FPU 相关的 CPU 指令）、改写 PC 的指令、32 位指令、RESBANK 指令、DIVS 指令或 DIVU 指令时，如果解码这些指令，则开始槽非法指令异常处理。而且，在将 FPU 设置为模块待机状态时，浮点指令及 FPU 相关的 CPU 指令作为未定义代码处理，配置于延迟槽时，如果解码该指令，则开始槽非法指令异常处理。

执行槽非法指令异常处理时，CPU 的运行如下：

1. 从异常处理向量表取出异常服务程序的起始地址。
2. 将状态寄存器（SR）压栈。
3. 将程序计数器（PC）压栈。保存的 PC 值为未定义代码、改写 PC 的指令、32 位指令、RESBANK 指令、DIVS 指令或 DIVU 指令之前的延迟转移指令的跳转目标地址。
4. 跳转至从异常处理向量表取出的异常服务程序的起始地址后，开始执行程序。此时的跳转不是延迟转移。

5.6.4 一般非法指令

解码配置于紧随延迟转移指令（延迟槽）以外的未定义代码（包含 FPU 模块待机时的 FPU 指令及 FPU 相关的 CPU 指令）时，开始一般非法指令异常处理。另外，FPU 设置为模块待机状态时，浮点指令及 FPU 相关的 CPU 指令作为未定义代码处理，配置于紧随延迟转移指令（延迟槽）以外时，如果解码该指令，则开始一般非法指令异常处理。

一般非法指令异常处理时，CPU 按照与槽非法指令异常相同的步骤运行。但是，与槽非法指令异常处理不同，保存的 PC 值为该未定义代码的起始地址。

5.6.5 整数除法异常

整数除法指令执行被零除时，或整数除法运算的结果上溢时，产生整数除法异常。被零除异常源的指令为 DIVU 与 DIVS。上溢异常源的指令仅为 DIVS，仅在用 -1 除负的最大值时，产生该异常。产生整数除法异常时，CPU 的运行如下：

1. 从异常处理向量表取出对应已产生整数除法指令异常的异常服务程序的起始地址。
2. 将状态寄存器（SR）压栈。
3. 将程序计数器（PC）压栈，保存的 PC 值为产生异常的整数除法指令的起始地址。
4. 跳转至从异常处理向量表取出的异常服务程序的起始地址后，开始执行程序。此时的跳转不是延迟转移。

5.6.6 FPU 异常

浮点状态 / 控制寄存器（FPSCR）的 FPU 异常允许字段（Enable）的 V、Z、O、U 或 I 位置位时，产生 FPU 异常。这表示浮点运算指令引起 IEEE754 标准所定义的无效运算异常、被零除异常、上溢（有可能引起的指令）、下溢（有可能引起的指令）及不正确异常（有可能引起的指令）等。

作为 FPU 异常处理产生源的浮点运算指令如下：

FADD、FSUB、FMUL、FDIV、FMAC、FCMP/EQ、FCMP/GT、FLOAT、FTRC、FCNVDS、FCNVSD、FSQRT

仅在将对应的 FPU 异常允许位 (Enable) 置位时，产生 FPU 异常处理。FPU 通过浮点运算检测出异常源时，FPU 运行被中断，并通知 CPU 产生 FPU 异常处理。开始异常处理时，CPU 的运行如下：

1. 从异常处理向量表取出对应已产生 FPU 异常处理的异常服务程序的起始地址。
2. 将状态寄存器（SR）压栈。
3. 将程序计数器（PC）压栈，保存的 PC 值为最后执行指令的下一条指令的起始地址。
4. 跳转至从异常处理向量表取出的异常服务程序的起始地址后，开始执行程序。此时的跳转不是延迟转移。

FPSCR 的 FPU 异常标志字段（Flag）与是否接受 FPU 异常处理无关，总被更新，并一直保持置位状态，直到用户用明示由指令清除为止。每次执行浮点运算指令，FPSCR 的 FPU 异常源字段（Cause）都发生变化。

另外，FPSCR 的 FPU 异常允许字段（Enable）的 V 位及 FPSCR 的 QIS 位置位时，如果对浮点运算指令源输入 qNaN 或 $\pm \infty$ ，则产生 FPU 异常处理。

5.7 不接受异常处理时

如表 5.11 所示，紧随延迟转移指令之后产生地址错误、FPU 异常、寄存器存储体错误（上溢）及中断时，有可能会不立即接受而被保留。此时，如果解码接受异常的指令，则可接受。

表 5.11 产生紧随延迟转移指令之后的异常源

产生时间	异常源			
	地址错误	FPU 异常	寄存器存储体错误 (上溢)	中断
紧随延迟转移指令之后 *	×	×	×	×

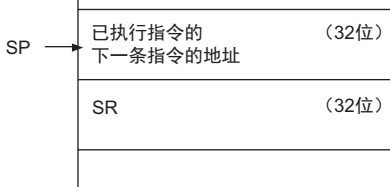
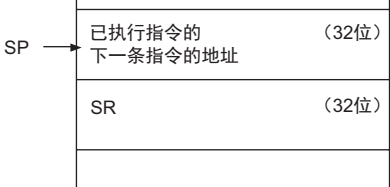
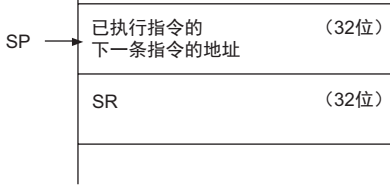
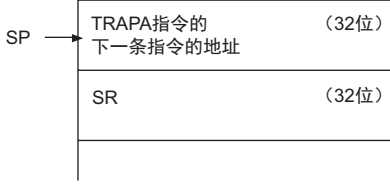
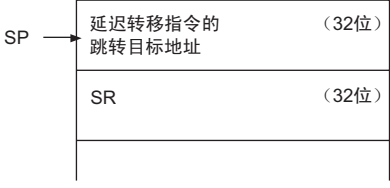
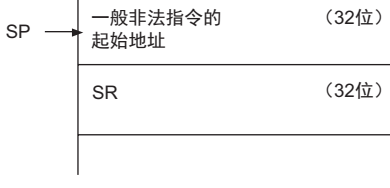
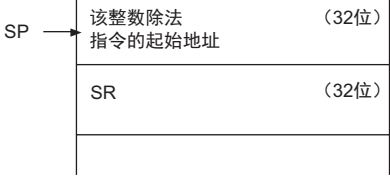
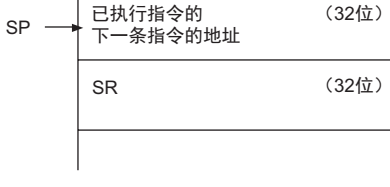
【符号说明】×：不接受

【注】* 延迟转移指令：JMP、JSR、BRA、BSR、RTS、RTE、BF/S、BT/S、BSRF、BRA

5.8 异常处理后的堆栈状态

异常处理结束后的堆栈状态，如表 5.12 所示。

表 5.12 异常处理结束后的堆栈状态

种类	堆栈状态	种类	堆栈状态
地址错误		中断	
寄存器存储体错误（上溢）		寄存器存储体错误（下溢）	
陷阱指令		槽非法指令	
一般非法指令		整数除法异常	
FPU 异常			

5.9 使用时的注意事项

5.9.1 堆栈指针（SP）的值

必须设定 SP 的值为 4 的倍数。否则，在异常处理的堆栈存取时，会产生地址错误。

5.9.2 向量基址寄存器（VBR）的值

必须设定 VBR 的值为 4 的倍数，否则，在异常处理的堆栈存取时，会产生地址错误。

5.9.3 在地址错误异常处理的堆栈存取时产生的地址错误

如果 SP 的值不为 4 的倍数，则在异常处理（中断等）的堆栈存取时产生地址错误，并在此异常处理结束后，转移至地址错误异常处理。地址错误异常处理的堆栈存取时，也产生地址错误，为了避免持续因无休止的地址错误异常处理导致的堆栈存取，不接受该地址错误。由此，可将程序的控制转移至地址错误异常服务程序，执行错误处理。

如果异常处理的堆栈存取时产生地址错误，则执行堆栈存取的总线周期（写入）。SR 与 PC 的堆栈存取过程中，各自的 SP 分别减 4，因此即使堆栈存取结束之后，SP 的值也不是 4 的倍数。另外，堆栈存取时输出的地址值为 SP 的值，输出产生错误的地址。此时，堆栈存取的写入数据不定。

第 6 章 中断控制器 (INTC)

中断控制器 (INTC) 判断中断源的优先顺序, 控制对 CPU 的中断请求。INTC 有设定各中断优先顺序的寄存器, 用户根据在该寄存器设定的优先顺序来处理中断请求。

6.1 特点

- 可设定 16 级中断优先顺序
通过 15 个中断优先级设定寄存器, 可按请求源将 IRQ 中断、PINT 中断及内部外围模块中断的优先顺序用最多 16 级进行设定。
- NMI 噪声消除器功能
具备表示 NMI 引脚状态的 NMI 输入电平位。在中断异常服务程序读取本位, 可确认引脚状态, 并可用作噪声消除器。
- 可对外部输出已产生的中断 ($\overline{\text{IRQOUT}}$ 引脚)
例如: 本 LSI 释放总线权时, 可向外部总线主控器通知已产生内部外围模块中断, 并可请求总线权。
- 寄存器存储体
本 LSI 内置寄存器存储体, 可在执行中断处理的同时, 高速执行寄存器的保存、返回。

INTC 框图如图 6.1 所示。

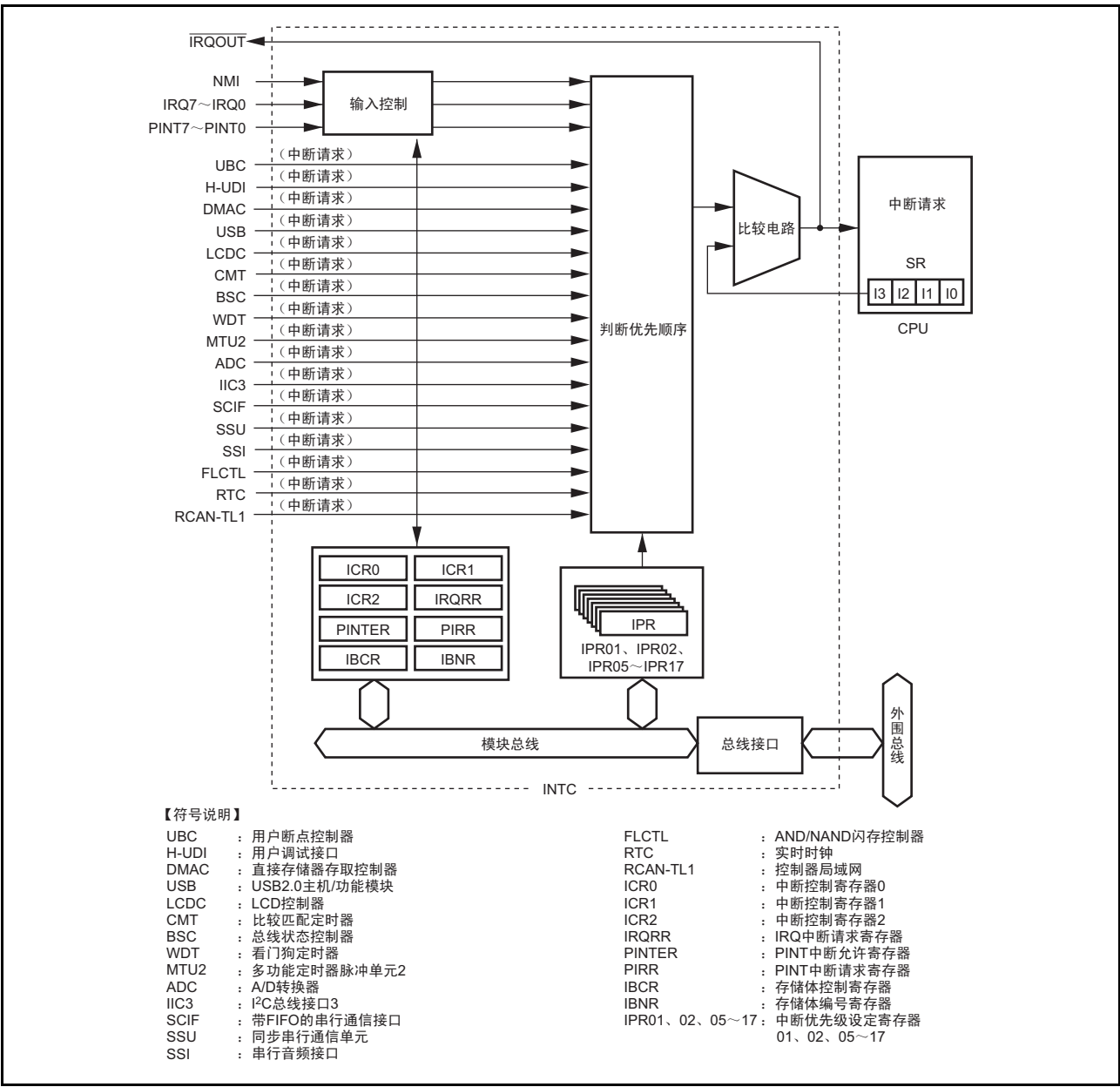


图 6.1 INTC 框图

6.2 输入 / 输出引脚

INTC 的引脚如表 6.1 所示。

表 6.1 引脚结构

名称	引脚名称	输入 / 输出	功 能
非屏蔽中断输入引脚	NMI	输入	输入不可屏蔽的中断请求信号
中断请求输入引脚	IRQ7 ~ IRQ0	输入	输入可屏蔽的中断请求信号
	PINT7 ~ PINT0	输入	
中断请求输出引脚	IRQOUT	输出	输出通知产生中断源的信号

6.3 寄存器说明

INTC 有以下寄存器。通过这些寄存器，可设定中断优先顺序及控制检测外部中断输入信号等。

表 6.2 寄存器结构

寄存器名称	简称	R/W	初始值	地址	存取长度
中断控制寄存器 0	ICR0	R/W	*1	H'FFFE0800	16、32
中断控制寄存器 1	ICR1	R/W	H'0000	H'FFFE0802	16、32
中断控制寄存器 2	ICR2	R/W	H'0000	H'FFFE0804	16、32
IRQ 中断请求寄存器	IRQRR	R/(W)*2	H'0000	H'FFFE0806	16、32
PINT 中断允许寄存器	PINTER	R/W	H'0000	H'FFFE0808	16、32
PINT 中断请求寄存器	PIRR	R	H'0000	H'FFFE080A	16、32
存储体控制寄存器	IBCR	R/W	H'0000	H'FFFE080C	16、32
存储体编号寄存器	IBNR	R/W	H'0000	H'FFFE080E	16、32
中断优先级设定寄存器 01	IPR01	R/W	H'0000	H'FFFE0818	16、32
中断优先级设定寄存器 02	IPR02	R/W	H'0000	H'FFFE081A	16、32
中断优先级设定寄存器 05	IPR05	R/W	H'0000	H'FFFE0820	16、32
中断优先级设定寄存器 06	IPR06	R/W	H'0000	H'FFFE0C00	16、32
中断优先级设定寄存器 07	IPR07	R/W	H'0000	H'FFFE0C02	16、32
中断优先级设定寄存器 08	IPR08	R/W	H'0000	H'FFFE0C04	16、32
中断优先级设定寄存器 09	IPR09	R/W	H'0000	H'FFFE0C06	16、32
中断优先级设定寄存器 10	IPR10	R/W	H'0000	H'FFFE0C08	16、32
中断优先级设定寄存器 11	IPR11	R/W	H'0000	H'FFFE0C0A	16、32
中断优先级设定寄存器 12	IPR12	R/W	H'0000	H'FFFE0C0C	16、32
中断优先级设定寄存器 13	IPR13	R/W	H'0000	H'FFFE0C0E	16、32
中断优先级设定寄存器 14	IPR14	R/W	H'0000	H'FFFE0C10	16、32
中断优先级设定寄存器 15	IPR15	R/W	H'0000	H'FFFE0C12	16、32
中断优先级设定寄存器 16	IPR16	R/W	H'0000	H'FFFE0C14	16、32
中断优先级设定寄存器 17	IPR17	R/W	H'0000	H'FFFE0C16	16、32

【注】 *1 NMI 引脚为高电平时：H'8000；低电平时：H'0000。

*2 为了清除标志，仅可在读取 1 后写入 0。

6.3.1 中断优先级设定寄存器 01、02、05 ~ 17 (IPR01、IPR02、IPR05 ~ IPR17)

IPR01、IPR02、IPR05 ~ IPR17 均为 16 位可读取 / 写入的寄存器，设定 IRQ 中断、PINT 中断及内部外围模块中断的优先顺序 (0 ~ 15 级)。中断请求源与 IPR01、IPR02、IPR05 ~ IPR17 各位的对应关系如表 6.3 所示。

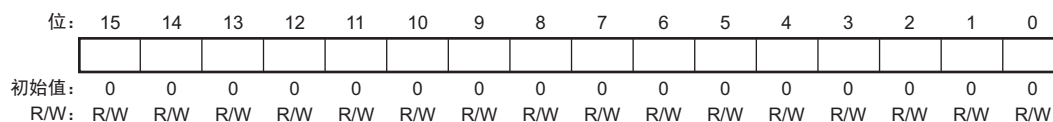


表 6.3 中断请求源与 IPR01、IPR02、IPR05 ~ IPR17

寄存器名称	位			
	15 ~ 12	11 ~ 8	7 ~ 4	3 ~ 0
中断优先级设定寄存器 01	IRQ0	IRQ1	IRQ2	IRQ3
中断优先级设定寄存器 02	IRQ4	IRQ5	IRQ6	IRQ7
中断优先级设定寄存器 05	PINT7 ~ PINT0	保留	保留	保留
中断优先级设定寄存器 06	DMAC0	DMAC1	DMAC2	DMAC3
中断优先级设定寄存器 07	DMAC4	DMAC5	DMAC6	DMAC7
中断优先级设定寄存器 08	USB	LCDC	CMT0	CMT1
中断优先级设定寄存器 09	BSC	WDT	MTU0 (TGI0A ~ TGI0D)	MTU0 (TCI0V、TGI0E、 TGI0F)
中断优先级设定寄存器 10	MTU1 (TGI1A、TGI1B)	MTU1 (TCI1V、TCI1U)	MTU2 (TGI2A、TGI2B)	MTU2 (TCI2V、TCI2U)
中断优先级设定寄存器 11	MTU3 (TGI3A ~ TGI3D)	MTU3 (TCI3V)	MTU4 (TGI4A ~ TGI4D)	MTU4 (TCI4V)
中断优先级设定寄存器 12	ADC	IIC3-0	IIC3-1	IIC3-2
中断优先级设定寄存器 13	IIC3-3	SCIF0	SCIF1	SCIF2
中断优先级设定寄存器 14	SCIF3	SSU0	SSU1	SSI0
中断优先级设定寄存器 15	SSI1	SSI2	SSI3	保留
中断优先级设定寄存器 16	FLCTL	保留	RTC	RCAN0
中断优先级设定寄存器 17	RCAN1	保留	保留	保留

如表 6.3 所示，通过对 bit15 ~ 12、bit11 ~ 8、bit7 ~ 4、bit3 ~ 0 的各 4 位设置 H'0 (0000) ~ H'F (1111) 范围内的值，可设定各自对应的中断优先顺序。设置 H'0 时，中断优先顺序为优先级 0 (最低级)；设置 H'F 时，中断优先顺序为优先级 15 (最高级)。

6.3.2 中断控制寄存器 0（ICR0）

ICR0 为 16 位寄存器，设定外部中断输入引脚 NMI 的输入信号检测模式，并表示 NMI 引脚的输入电平。

位：	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NMIL	-	-	-	-	-	-	NMIE	-	-	-	-	-	-	-	-
初始值：	*	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W：	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R

【注】* NMI引脚为高电平时，此位为1；为低电平时，此位值为0。

位	位名称	初始值	R/W	说 明
15	NMIL	*	R	NMI 输入电平 设定输入至 NMI 引脚的信号电平。读取本位，可得知 NMI 引脚的电平，写入无效。 0：向 NMI 引脚输入低电平。 1：向 NMI 引脚输入高电平。
14 ~ 9	—	均为 0	R	保留位 读取值、写入值总是为 0。
8	NMIE	0	R/W	选择 NMI 边沿 选择在 NMI 输入的下降沿或上升沿检测中断请求信号。 0：在 NMI 输入的下降沿检测中断请求。 1：在 NMI 输入的上升沿检测中断请求。
7 ~ 0	—	均为 0	R	保留位 读取值、写入值总是为 0。

6.3.3 中断控制寄存器 1 (ICR1)

ICR1 为 16 位寄存器，可对外部中断输入引脚 IRQ7 ~ IRQ0 分别指定低电平、下降沿、上升沿或双边沿的检测模式。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IRQ71S	IRQ70S	IRQ61S	IRQ60S	IRQ51S	IRQ50S	IRQ41S	IRQ40S	IRQ31S	IRQ30S	IRQ21S	IRQ20S	IRQ11S	IRQ10S	IRQ01S	IRQ00S
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15	IRQ71S	0	R/W	IRQ 检测选择 选择在低电平、下降沿、上升沿或双边沿检测 IRQ7 ~ IRQ0 引脚的中断信号。 00: 在 IRQn 输入的低电平检测中断请求。 01: 在 IRQn 输入的下降沿检测中断请求。 10: 在 IRQn 输入的上升沿检测中断请求。 11: 在 IRQn 输入的双边沿检测中断请求。
14	IRQ70S	0	R/W	
13	IRQ61S	0	R/W	
12	IRQ60S	0	R/W	
11	IRQ51S	0	R/W	
10	IRQ50S	0	R/W	
9	IRQ41S	0	R/W	
8	IRQ40S	0	R/W	
7	IRQ31S	0	R/W	
6	IRQ30S	0	R/W	
5	IRQ21S	0	R/W	
4	IRQ20S	0	R/W	
3	IRQ11S	0	R/W	
2	IRQ10S	0	R/W	
1	IRQ01S	0	R/W	
0	IRQ00S	0	R/W	

【符号说明】 n=7 ~ 0

6.3.4 中断控制寄存器 2（ICR2）

ICR2 为 16 位寄存器，可对外部中断输入引脚 PINT7 ~ PINT0 分别指定低电平或高电平的检测模式。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	PINT7S	PINT6S	PINT5S	PINT4S	PINT3S	PINT2S	PINT1S	PINT0S
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15 ~ 8	—	均为 0	R	保留位 读取值、写入值总是为 0。
7	PINT7S	0	R/W	PINT 检测选择 选择在低电平或高电平检测 PINT7 ~ PINT0 引脚的中断信号。 0: 在 PINTn 输入的低电平检测中断请求。 1: 在 PINTn 输入的高电平检测中断请求。
6	PINT6S	0	R/W	
5	PINT5S	0	R/W	
4	PINT4S	0	R/W	
3	PINT3S	0	R/W	
2	PINT2S	0	R/W	
1	PINT1S	0	R/W	
0	PINT0S	0	R/W	

【符号说明】 n=7 ~ 0

6.3.5 IRQ 中断请求寄存器 (IRQRR)

IRQRR 为 16 位寄存器，表示外部中断输入引脚 IRQ7 ~ IRQ0 的中断请求。IRQ7 ~ IRQ0 中断设定为边沿检测时，通过读取 IRQ7F ~ IRQ0F=1 后，对 IRQ7F ~ IRQ0F 写入 0，可取消已保持的中断请求。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	IRQ7F	IRQ6F	IRQ5F	IRQ4F	IRQ3F	IRQ2F	IRQ1F	IRQ0F
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*

【注】* 为了清除标志，仅可在读取1后写入0。

位	位名称	初始值	R/W	说 明
15 ~ 8	—	均为 0	R	保留位 读取值、写入值总是为 0。
7	IRQ7F	0	R/(W)*	IRQ 中断请求 表示 IRQ7 ~ IRQ0 中断请求的状态。 电平检测时 0: 不存在 IRQn 中断请求。 [清除条件] • IRQn 输入为高电平时 1: 存在 IRQn 中断请求。 [置位条件] • IRQn 输入为低电平时 边沿检测时 0: 未检测出 IRQn 中断请求。 [清除条件] • 读取 IRQnF=1 的状态后，写入 0 时 • 执行 IRQn 中断异常处理时 1: 检测出 IRQn 中断请求。 [置位条件] • 在 IRQn 引脚产生对应 ICR1 的 IRQn1S、IRQn0S 边沿时
6	IRQ6F	0	R/(W)*	
5	IRQ5F	0	R/(W)*	
4	IRQ4F	0	R/(W)*	
3	IRQ3F	0	R/(W)*	
2	IRQ2F	0	R/(W)*	
1	IRQ1F	0	R/(W)*	
0	IRQ0F	0	R/(W)*	

【符号说明】 n=7 ~ 0

6.3.6 PINT 中断允许寄存器（PINTER）

PINTER 为 16 位寄存器，允许对外部中断输入引脚 PINT7 ~ PINT0 输入中断请求。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	PINT7E	PINT6E	PINT5E	PINT4E	PINT3E	PINT2E	PINT1E	PINT0E
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15 ~ 8	—	均为 0	R	保留位 读取值、写入值总是为 0。
7	PINT7E	0	R/W	PINT 允许位 选择是否允许对 PINT7 ~ PINT0 引脚输入中断请求。 0: 禁止 PINTn 引脚输入中断请求。 1: 允许 PINTn 引脚输入中断请求。
6	PINT6E	0	R/W	
5	PINT5E	0	R/W	
4	PINT4E	0	R/W	
3	PINT3E	0	R/W	
2	PINT2E	0	R/W	
1	PINT1E	0	R/W	
0	PINT0E	0	R/W	

【符号说明】 n=7 ~ 0

6.3.7 PINT 中断请求寄存器 (PIRR)

PIRR 为 16 位寄存器，表示外部中断 PINT7 ~ PINT0 的中断请求。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	PINT7R	PINT6R	PINT5R	PINT4R	PINT3R	PINT2R	PINT1R	PINT0R
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名称	初始值	R/W	说 明
15 ~ 8	—	均为 0	R	保留位 读取值、写入值总是为 0。
7	PINT7R	0	R	PINT 中断请求 表示 PINT7 ~ PINT0 中断请求。 0: PINTn 引脚无中断。 1: PINTn 引脚有中断。
6	PINT6R	0	R	
5	PINT5R	0	R	
4	PINT4R	0	R	
3	PINT3R	0	R	
2	PINT2R	0	R	
1	PINT1R	0	R	
0	PINT0R	0	R	

【符号说明】 n=7 ~ 0

6.3.8 存储体控制寄存器 (IBCR)

对于中断优先级，IBCR 可设定允许 / 禁止使用寄存器存储体。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	E15	E14	E13	E12	E11	E10	E9	E8	E7	E6	E5	E4	E3	E2	E1	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

位	位名称	初始值	R/W	说 明
15	E15	0	R/W	允许 对于中断优先级 15 ~ 1，设定允许 / 禁止使用寄存器存储体。但是，用户断点中断总是禁止使用寄存器存储体。 0: 禁止使用寄存器存储体。 1: 允许使用寄存器存储体。
14	E14	0	R/W	
13	E13	0	R/W	
12	E12	0	R/W	
11	E11	0	R/W	
10	E10	0	R/W	
9	E9	0	R/W	
8	E8	0	R/W	
7	E7	0	R/W	
6	E6	0	R/W	
5	E5	0	R/W	
4	E4	0	R/W	
3	E3	0	R/W	
2	E2	0	R/W	
1	E1	0	R/W	
0	—	0	R	保留位 读取值、写入值总是为 0。

6.3.9 存储体编号寄存器 (IBNR)

IBNR 设定允许 / 禁止使用寄存器存储体及产生寄存器存储体上溢异常。另外，用 BN3 ~ BN0 表示下一个要保存的存储体编号。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BE[1:0]		BOVE	-	-	-	-	-	-	-	-	-	BN[3:0]			
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名称	初始值	R/W	说 明
15、14	BE[1:0]	00	R/W	寄存器存储体允许 设定允许 / 禁止使用寄存器存储体。 00: 禁止所有中断使用存储体。忽略 IBCR 的设定。 01: 允许除 NMI、用户断点之外的所有中断使用存储体。忽略 IBCR 的设定。 10: 保留 (禁止设定) 11: 根据 IBCR 的设定, 使用寄存器存储体。
13	BOVE	0	R/W	寄存器存储体上溢允许 设定允许 / 禁止产生寄存器存储体上溢异常。 0: 禁止产生寄存器存储体上溢异常。 1: 允许产生寄存器存储体上溢异常。
12 ~ 4	—	均为 0	R	保留位 读取值、写入值总是为 0。
3 ~ 0	BN[3:0]	0000	R	存储体编号 表示下一个要保存的存储体编号。接受使用了寄存器存储体的中断时, 保存至 BN3 ~ BN0 所示的寄存器存储体, 并将 BN+1。通过执行寄存器存储体返回指令, 将 BN-1 后, 从寄存器存储体返回。

6.4 中断源

中断源分 6 类，分别为 NMI、用户断点、H-UDI、IRQ、PINT 及内部外围模块。用中断优先级的值（0 ~ 16）表示各中断的优先顺序，0 为最低级，16 为最高级。设定为 0 时，屏蔽该中断。

6.4.1 NMI 中断

NMI 中断为优先级 16 的中断，总是被接受。在边沿检测 NMI 引脚的输入。通过设定中断控制寄存器 0（ICR0）的 NMI 边沿选择位（NMIE），可选择检测边沿为上升沿或下降沿。

NMI 中断的优先级为 16，但由 NMI 中断异常处理可将状态寄存器（SR）的中断屏蔽级位（I3 ~ I0）设定为 15。

6.4.2 用户断点中断

通过用户断点控制器（UBC）设定的断点条件成立时，产生用户断点中断，优先级为 15。由用户断点中断异常处理将 SR 的 I3 ~ I0 位设定为 15。有关用户断点，详情参阅“第 7 章 用户断点控制器（UBC）”。

6.4.3 H-UDI 中断

用户调试接口（H-UDI）中断的优先级为 15。串行输入时，产生 H-UDI 中断指令。在边沿检测出 H-UDI 中断请求，并保持到被接受为止。由 H-UDI 中断异常处理，将 SR 的 I3 ~ I0 位设定为 15。有关 H-UDI 中断，详情参阅“第 29 章 用户调试接口（H-UDI）”。

6.4.4 IRQ 中断

由 IRQ7 ~ IRQ0 引脚的输入产生 IRQ 中断。通过设定中断控制寄存器 1（ICR1）的 IRQ 检测选择位（IRQ71S ~ IRQ01S、IRQ70S ~ IRQ00S），IRQ 中断可对每个引脚选择低电平检测、下降沿检测、上升沿检测或双边沿检测。另外，通过中断优先级设定寄存器 01、02（IPR01、IPR02），可在 0 ~ 15 级范围内对每个引脚设定优先级。

IRQ 中断设定为低电平检测时，可在 IRQ7 ~ IRQ0 引脚为低电平期间，对 INTC 发送中断请求信号；IRQ7 ~ IRQ0 引脚为高电平时，不对 INTC 发送中断请求信号。通过读取 IRQ 中断请求寄存器（IRQRR）的 IRQ 中断请求位（IRQ7F ~ IRQ0F），可确认中断请求。

IRQ 中断设定为边沿检测时，根据 IRQ7 ~ IRQ0 引脚的变化检测中断请求，并对 INTC 发送中断请求信号。IRQ 中断请求的检测结果，保持到该中断请求被接受为止。另外，可通过读取 IRQRR 的 IRQ7F ~ IRQ0F 位，确认是否检测出 IRQ 中断请求，并可通过读取 1 后写入 0，取消 IRQ 中断请求的检测结果。

IRQ 中断异常处理时，将 SR 的 I3 ~ I0 位设定为已接受的 IRQ 中断优先级的值。

从 IRQ 中断异常服务程序返回时，为了避免再次接受，必须确认由 IRQ 中断请求寄存器（IRQRR）清除中断请求后，再执行 RTE 指令。

6.4.5 PINT 中断

由 PINT7 ~ PINT0 引脚的输入产生 PINT 中断。由 PINT 中断允许寄存器 (PINTER) 的 PINT 允许位 (PINT7E ~ PINT0E) 设定允许输入中断请求。通过设定中断控制寄存器 2 (ICR2) 的 PINT 检测选择位 (PINT7S ~ PINT0S), PINT7 ~ PINT0 可对各引脚选择低电平或高电平检测。通过设定中断优先级设定寄存器 05 (IPR05) 的 bit15 ~ 12, 可在 0 ~ 15 级范围内统一对 PINT7 ~ PINT0 中断设定优先级。

PINT7 ~ PINT0 中断设定为低电平检测时, 在 PINT 引脚为低电平期间, 可对 INTC 发送中断请求信号; PINT 引脚为高电平时, 不对 INTC 发送中断请求信号。通过读取 PINT 中断请求寄存器 (PIRR) 的 PINT 中断请求位 (PINT7R ~ PINT0R), 可确认中断请求的优先级。除极性相反的情况之外, 设定为高电平检测时也相同。PINT 中断异常处理时, 将 SR 的 I3 ~ I0 设定为 PINT 中断优先级的值。

从 PINT 中断异常服务程序返回时, 为了避免再次接受, 必须确认由 PINT 中断请求寄存器 (PIRR) 清除中断请求后, 再执行 RTE 指令。

6.4.6 内部外围模块中断

内部外围模块中断为以下内部外围模块产生的中断:

- 直接存储器存取控制器 (DMAC)
- USB2.0 主机/功能模块 (USB)
- LCD 控制器 (LCDC)
- 比较匹配定时器 (CMT)
- 总线状态控制器 (BSC)
- 看门狗定时器 (WDT)
- 多功能定时器脉冲单元 2 (MTU2)
- A/D 转换器 (ADC)
- I²C 总线接口 3 (IIC3)
- 带 FIFO 的串行通信接口 (SCIF)
- 同步串行通信单元 (SSU)
- 串行音频接口 (SSI)
- AND/NAND 闪存控制器 (FLCTL)
- 实时时钟 (RTC)
- 控制器局域网 (RCAN-TL1)

因为各中断源分配了不同的中断向量, 所以无需通过异常服务程序判断中断源。可通过中断优先级设定寄存器 05 ~ 17 (IPR05 ~ IPR17) 在 0 ~ 15 级范围内对各模块设定优先级。内部外围模块中断异常处理时, 将 SR 的 I3 ~ I0 位设定为已接受的内部外围模块中断优先级的值。

6.5 中断异常处理向量表与优先顺序

中断源、向量编号、向量表地址偏移量及中断优先顺序如表 6.4 所示。

对各中断源分配不同的向量编号和向量表地址偏移量。根据向量编号和向量表地址偏移量计算向量表地址。中断异常处理时, 从向量表地址所示的向量表取出中断异常服务程序的起始地址。有关向量表地址的计算方法, 参照“第 5 章 异常处理”的“表 5.4 异常处理向量表地址的计算方法”。

通过中断优先级设定寄存器 01、02、05 ~ 17 (IPR01、IPR02、IPR05 ~ IPR17), 可在优先级 0 ~ 15 的范围内对每个引脚或模块任意设定 IRQ 中断、PINT 中断及内部外围模块中断的优先顺序。但是, 在 IPR05 ~ IPR17 产生相同 IPR 内的多个中断的优先顺序按照表 6.4 规定的 IPR 设定单位内的优先顺序, 且不得更改。由上电复位将 IRQ 中断、PINT 中断及内部外围模块中断的优先顺序设定为优先级 0。将多个中断源的优先顺序设定为相同优先级, 且同时产生这些中断时, 按照表 6.4 所示的默认优先顺序处理。

表 6.4 中断异常向量表与优先顺序

中断源			中断向量		中断优先顺序 (初始值)	对应的 IPR (位)	IPR 设定单 位内的优先 顺序	默认优先顺序
			向量	向量表地址偏移量				
NMI			11	H'0000002C ~ H'0000002F	16	—	—	高 ↑
用户断点			12	H'00000030 ~ H'00000033	15	—	—	
H-UDI			14	H'00000038 ~ H'0000003B	15	—	—	
IRQ	IRQ0		64	H'00000100 ~ H'00000103	0 ~ 15(0)	IPR01(15 ~ 12)	—	
	IRQ1		65	H'00000104 ~ H'00000107	0 ~ 15(0)	IPR01(11 ~ 8)	—	
	IRQ2		66	H'00000108 ~ H'0000010B	0 ~ 15(0)	IPR01(7 ~ 4)	—	
	IRQ3		67	H'0000010C ~ H'0000010F	0 ~ 15(0)	IPR01(3 ~ 0)	—	
	IRQ4		68	H'00000110 ~ H'00000113	0 ~ 15(0)	IPR02(15 ~ 12)	—	
	IRQ5		69	H'00000114 ~ H'00000117	0 ~ 15(0)	IPR02(11 ~ 8)	—	
	IRQ6		70	H'00000118 ~ H'0000011B	0 ~ 15(0)	IPR02(7 ~ 4)	—	
	IRQ7		71	H'0000011C ~ H'0000011F	0 ~ 15(0)	IPR02(3 ~ 0)	—	
PINT	PINT0		80	H'00000140 ~ H'00000143	0 ~ 15(0)	IPR05(15 ~ 12)	1	
	PINT1		81	H'00000144 ~ H'00000147			2	
	PINT2		82	H'00000148 ~ H'0000014B			3	
	PINT3		83	H'0000014C ~ H'0000014F			4	
	PINT4		84	H'00000150 ~ H'00000153			5	
	PINT5		85	H'00000154 ~ H'00000157			6	
	PINT6		86	H'00000158 ~ H'0000015B			7	
	PINT7		87	H'0000015C ~ H'0000015F			8	
DMAC	DMAC0	DEI0	108	H'000001B0 ~ H'000001B3	0 ~ 15(0)	IPR06(15 ~ 12)	1	↓ 低
		HEI0	109	H'000001B4 ~ H'000001B7			2	
	DMAC1	DEI1	112	H'000001C0 ~ H'000001C3	0 ~ 15(0)	IPR06(11 ~ 8)	1	
		HEI1	113	H'000001C4 ~ H'000001C7			2	
	DMAC2	DEI2	116	H'000001D0 ~ H'000001D3	0 ~ 15(0)	IPR06(7 ~ 4)	1	
		HEI2	117	H'000001D4 ~ H'000001D7			2	
	DMAC3	DEI3	120	H'000001E0 ~ H'000001E3	0 ~ 15(0)	IPR06(3 ~ 0)	1	
		HEI3	121	H'000001E4 ~ H'000001E7			2	
	DMAC4	DEI4	124	H'000001F0 ~ H'000001F3	0 ~ 15(0)	IPR07(15 ~ 12)	1	
		HEI4	125	H'000001F4 ~ H'000001F7			2	
	DMAC5	DEI5	128	H'00000200 ~ H'00000203	0 ~ 15(0)	IPR07(11 ~ 8)	1	
		HEI5	129	H'00000204 ~ H'00000207			2	
	DMAC6	DEI6	132	H'00000210 ~ H'00000213	0 ~ 15(0)	IPR07(7 ~ 4)	1	
		HEI6	133	H'00000214 ~ H'00000217			2	
	DMAC7	DEI7	136	H'00000220 ~ H'00000223	0 ~ 15(0)	IPR07(3 ~ 0)	1	
		HEI7	137	H'00000224 ~ H'00000227			2	

中断源			中断向量		中断优先顺序 (初始值)	对应的 IPR (位)	IPR 设定单 位内的优先 顺序	默认优先顺序
			向量	向量表地址偏移量				
USB	USBI		140	H'00000230 ~ H'00000233	0 ~ 15(0)	IPR08(15 ~ 12)	—	高 ↑ ↓ 低
LCDC	LCDCI		141	H'00000234 ~ H'00000237	0 ~ 15(0)	IPR08(11 ~ 8)	—	
CMT	CMI0		142	H'00000238 ~ H'0000023B	0 ~ 15(0)	IPR08(7 ~ 4)		
	CMI1		143	H'0000023C ~ H'0000023F	0 ~ 15(0)	IPR08(3 ~ 0)	—	
BSC	CMI		144	H'00000240 ~ H'00000243	0 ~ 15(0)	IPR09(15 ~ 12)	—	
WDT	ITI		145	H'00000244 ~ H'00000247	0 ~ 15(0)	IPR09(11 ~ 8)	—	
MTU2	MTU0	TGI0A	146	H'00000248 ~ H'0000024B	0 ~ 15(0)	IPR09(7 ~ 4)	1	
		TGI0B	147	H'0000024C ~ H'0000024F			2	
		TGI0C	148	H'00000250 ~ H'00000253			3	
		TGI0D	149	H'00000254 ~ H'00000257			4	
		TCI0V	150	H'00000258 ~ H'0000025B	0 ~ 15(0)	IPR09(3 ~ 0)	1	
		TGI0E	151	H'0000025C ~ H'0000025F			2	
		TGI0F	152	H'00000260 ~ H'00000263			3	
	MTU1	TGI1A	153	H'00000264 ~ H'00000267	0 ~ 15(0)	IPR10(15 ~ 12)	1	
		TGI1B	154	H'00000268 ~ H'0000026B			2	
		TCI1V	155	H'0000026C ~ H'0000026F	0 ~ 15(0)	IPR10(11 ~ 8)	1	
		TCI1U	156	H'00000270 ~ H'00000273			2	
	MTU2	TGI2A	157	H'00000274 ~ H'00000277	0 ~ 15(0)	IPR10(7 ~ 4)	1	
		TGI2B	158	H'00000278 ~ H'0000027B			2	
		TCI2V	159	H'0000027C ~ H'0000027F	0 ~ 15(0)	IPR10(3 ~ 0)	1	
		TCI2U	160	H'00000280 ~ H'00000283			2	
	MTU3	TGI3A	161	H'00000284 ~ H'00000287	0 ~ 15(0)	IPR11(15 ~ 12)	1	
		TGI3B	162	H'00000288 ~ H'0000028B			2	
		TGI3C	163	H'0000028C ~ H'0000028F			3	
		TGI3D	164	H'00000290 ~ H'00000293			4	
		TCI3V	165	H'00000294 ~ H'00000297	0 ~ 15(0)	IPR11(11 ~ 8)	—	
	MTU4	TGI4A	166	H'00000298 ~ H'0000029B	0 ~ 15(0)	IPR11(7 ~ 4)	1	
		TGI4B	167	H'0000029C ~ H'0000029F			2	
		TGI4C	168	H'000002A0 ~ H'000002A3			3	
		TGI4D	169	H'000002A4 ~ H'000002A7			4	
		TCI4V	170	H'000002A8 ~ H'000002AB	0 ~ 15(0)	IPR11(3 ~ 0)	—	
ADC	ADI		171	H'000002AC ~ H'000002AF	0 ~ 15(0)	IPR12(15 ~ 12)	—	

中断源			中断向量		中断优先顺序 (初始值)	对应的 IPR (位)	IPR 设定单 位内的优先 顺序	默认优先顺序
			向量	向量表地址偏移量				
IIC3	IIC3-0	STPI0	172	H'000002B0 ~ H'000002B3	0 ~ 15(0)	IPR12(11 ~ 8)	1	高 ↑ ↓ 低
		NAKI0	173	H'000002B4 ~ H'000002B7			2	
		RXI0	174	H'000002B8 ~ H'000002BB			3	
		TXI0	175	H'000002BC ~ H'000002BF			4	
		TEI0	176	H'000002C0 ~ H'000002C3			5	
	IIC3-1	STPI1	177	H'000002C4 ~ H'000002C7	0 ~ 15(0)	IPR12(7 ~ 4)	1	
		NAKI1	178	H'000002C8 ~ H'000002CB			2	
		RXI1	179	H'000002CC ~ H'000002CF			3	
		TXI1	180	H'000002D0 ~ H'000002D3			4	
		TEI1	181	H'000002D4 ~ H'000002D7			5	
	IIC3-2	STPI2	182	H'000002D8 ~ H'000002DB	0 ~ 15(0)	IPR12(3 ~ 0)	1	
		NAKI2	183	H'000002DC ~ H'000002DF			2	
		RXI2	184	H'000002E0 ~ H'000002E3			3	
		TXI2	185	H'000002E4 ~ H'000002E7			4	
		TEI2	186	H'000002E8 ~ H'000002EB			5	
	IIC3-3	STPI3	187	H'000002EC ~ H'000002EF	0 ~ 15(0)	IPR13(15 ~ 12)	1	
		NAKI3	188	H'000002F0 ~ H'000002F3			2	
		RXI3	189	H'000002F4 ~ H'000002F7			3	
		TXI3	190	H'000002F8 ~ H'000002FB			4	
		TEI3	191	H'000002FC ~ H'000002FF			5	
SCIF	SCIF0	BRI0	192	H'00000300 ~ H'00000303	0 ~ 15(0)	IPR13(11 ~ 8)	1	
		ERI0	193	H'00000304 ~ H'00000307			2	
		RXI0	194	H'00000308 ~ H'0000030B			3	
		TXI0	195	H'0000030C ~ H'0000030F			4	
	SCIF1	BRI1	196	H'00000310 ~ H'00000313	0 ~ 15(0)	IPR13(7 ~ 4)	1	
		ERI1	197	H'00000314 ~ H'00000317			2	
		RXI1	198	H'00000318 ~ H'0000031B			3	
		TXI1	199	H'0000031C ~ H'0000031F			4	
	SCIF2	BRI2	200	H'00000320 ~ H'00000323	0 ~ 15(0)	IPR13(3 ~ 0)	1	
		ERI2	201	H'00000324 ~ H'00000327			2	
		RXI2	202	H'00000328 ~ H'0000032B			3	
		TXI2	203	H'0000032C ~ H'0000032F			4	

RCJ09B0067-0100 Rev.1.00 2009.03.27
Page 112 of 1077

6.6 运行说明

6.6.1 中断运行的流程

运行流程如图 6.2 所示，产生中断时的运行流程说明如下：

1. 从各中断请求源向中断控制器发送中断请求信号。
2. 中断控制器根据中断优先级设定寄存器 01、02、05～17 (IPR01、IPR02、IPR05～IPR17)，从发送的中断请求中选择优先顺序最高的中断，忽略*低于该优先顺序的中断。此时，如果产生多个设定为相同优先顺序的中断或相同 IPR 设定内的中断，则按表 6.4 所示的默认优先顺序与 IPR 设定单位内的优先顺序，选择优先顺序最高的中断。
3. 比较中断控制器选择的中断优先级与 CPU 状态寄存器 (SR) 的中断屏蔽级位 (I3～I0)。忽略优先级等于或低于在 I3～I0 位设定的优先级的中断。仅接受高于 I3～I0 位的优先级的中断，并对 CPU 发送中断请求信号。
4. 中断控制器接受中断时，从 $\overline{\text{IRQOUT}}$ 引脚输出低电平。
5. 解码 CPU 欲执行的指令时，检测中断控制器发送的中断请求，并由中断异常处理 (参照图 6.4) 代替执行该指令。
6. 从对应已接受中断的异常处理向量表取出中断异常服务程序的起始地址。
7. 将状态寄存器 (SR) 压栈，并对 SR 的 I3～I0 位写入已接受中断的优先级。
8. 将程序计数器 (PC) 压栈。
9. 跳转至已取出的中断异常服务程序的起始地址后，开始执行程序。此时的跳转不是延迟转移。
10. 从 $\overline{\text{IRQOUT}}$ 引脚输出高电平。但是，接受优先级高于中断控制器当前接收的中断时， $\overline{\text{IRQOUT}}$ 引脚仍为低电平。

【注】 必须在中断处理程序中清除中断源标志。从清除中断源标志到实际取消 CPU 的中断源，需设定的时间为表 6.5 记载的“从产生中断请求，到比较中断控制器判断优先顺序及与 SR 屏蔽位后，向 CPU 发送中断请求信号的时间”。为了避免再次接受应清除的中断源，需在清除后读取中断源标志，然后再执行 RTE 指令。

* 设定为边沿检测的中断请求，保留到被接受为止。但 IRQ 中断时，可通过存取 IRQ 中断请求寄存器 (IRQRR) 取消。详情参阅“6.4.4 IRQ 中断”。

由上电复位清除由边沿检测保留的中断。

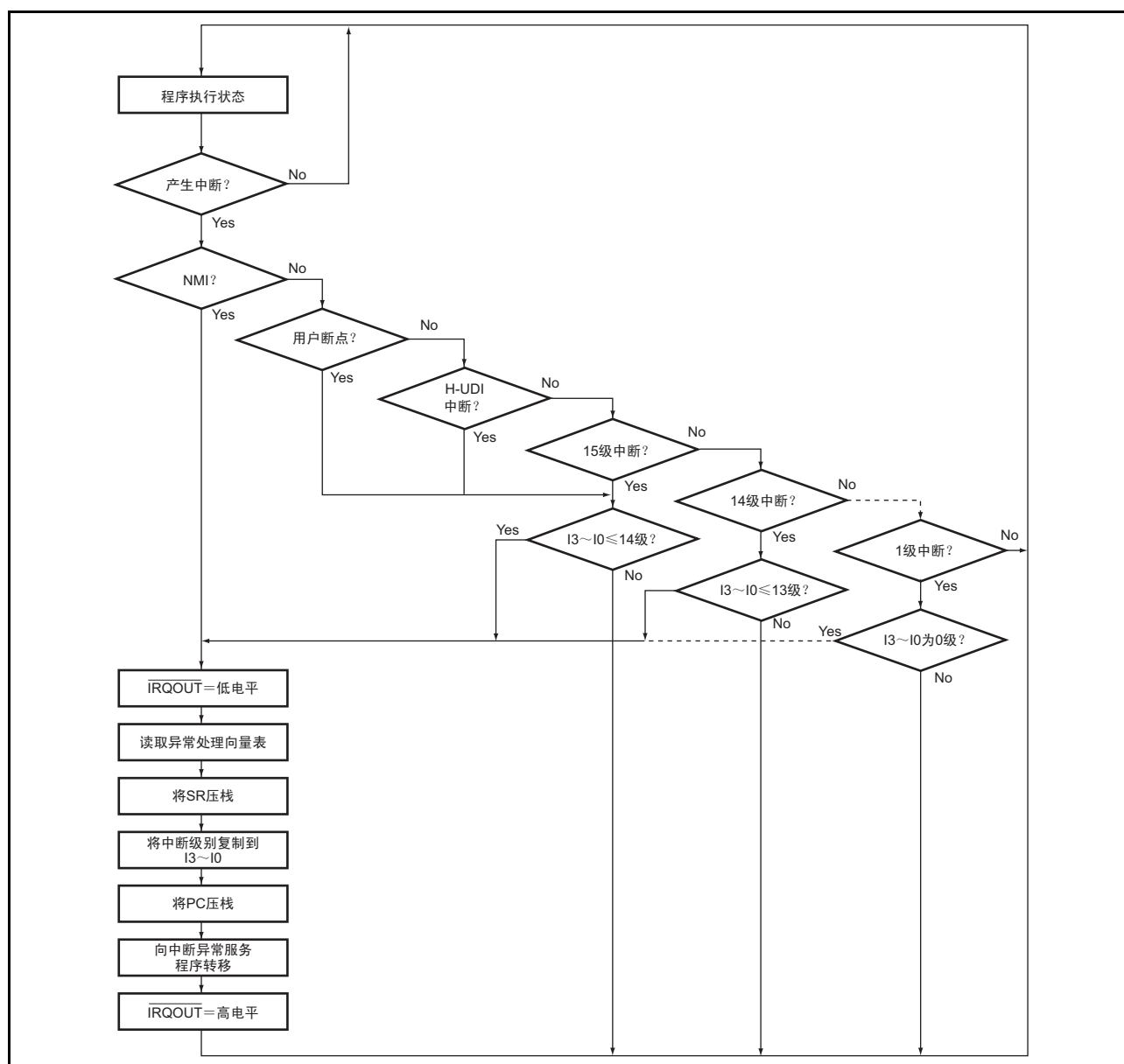


图 6.2 中断运行流程

6.6.2 中断异常处理结束后的堆栈状态

中断异常处理结束后的堆栈状态如图 6.3 所示。

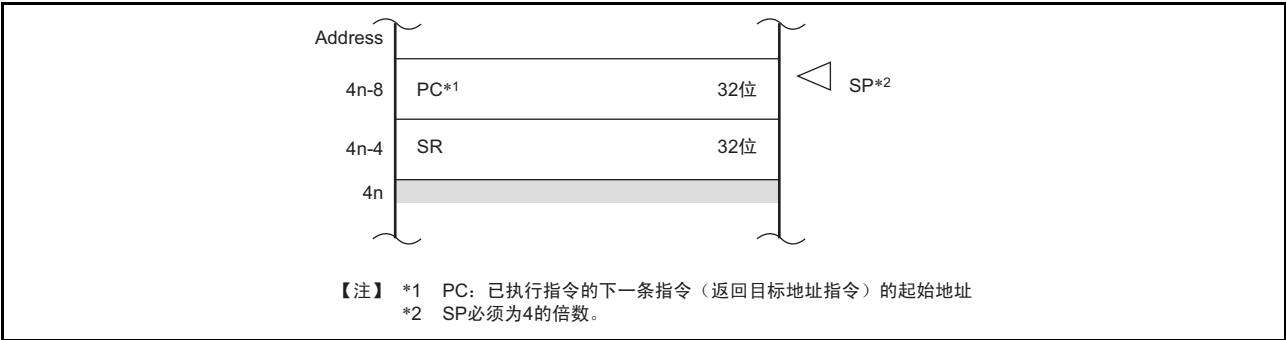


图 6.3 中断异常处理结束后的堆栈状态

6.7 中断响应时间

从产生中断请求到执行中断异常处理，并开始取异常服务程序起始指令的时间（中断响应时间）如表 6.5 所示。中断处理在无堆积、有堆积且无寄存器存储体上溢、有堆积且有寄存器存储体上溢时，运行各不同。无堆积时的流水线运行例如图 6.4、图 6.5 所示；有堆积且无寄存器存储体上溢时的流水线运行例如图 6.6、图 6.7 所示；有堆积且有寄存器存储体上溢时的流水线运行例如图 6.8、图 6.9 所示。

表 6.5 中断响应时间

项目			状态数					备注	
			NMI	用户断点	H-UDI	IRQ、PINT	USB		外围模块 (除 USB 之外)
从产生中断请求，到比较中断控制器判断优先顺序及与 SR 的屏蔽位后，向 CPU 发送中断请求信号的时间			2lcyc+2Bcyc+1Pcyc	3lcyc	2lcyc+1Pcyc	2lcyc+3Bcyc+1Pcyc	2lcyc+4Bcyc	2lcyc+2Bcyc	
从向 CPU 输入中断请求信号，到结束正在执行的顺序后，开始中断异常处理，并取中断异常服务程序的起始指令的时间。	无寄存器存储体。	最小值	3lcyc+m1+m2					最小值：中断等待时间为 0 时的值。 最大值：中断异常处理过程中产生更高位的中断请求时的值。	
		最大值	4lcyc+2(m1+m2)+m3						
	有寄存器存储体。 无寄存器存储体上溢。	最小值	—		3lcyc+m1+m2			最小值：中断等待时间为 0 时的值。 最大值：执行 RESBANK 指令过程中产生中断请求时的值。	
		最大值	—		12lcyc+m1+m2				
	有寄存器存储体。 有寄存器存储体上溢。	最小值	—		3lcyc+m1+m2			最小值：中断等待时间为 0 时的值。 最大值：执行 RESBANK 指令过程中产生中断请求时的值。	
		最大值	—		3lcyc+m1+m2+19(m4)				

项目			状态数						备注
			NMI	用户 断点	H-UDI	IRQ、PINT	USB	外围模块 (除 USB 之外)	
响应 时间	无寄存器 存储体	最小值	$5\text{lcy} + 2\text{Bcy} + 1\text{Pcy} + \text{m1} + \text{m2}$	$6\text{lcy} + \text{m1} + \text{m2}$	$5\text{lcy} + 1\text{Pcy} + \text{m1} + \text{m2}$	$5\text{lcy} + 3\text{Bcy} + 1\text{Pcy} + \text{m1} + \text{m2}$	$5\text{lcy} + 4\text{Bcy} + \text{m1} + \text{m2}$	$5\text{lcy} + 2\text{Bcy} + \text{m1} + \text{m2}$	200MHz 运行时 *1*2; 0.040 ~ 0.110μs
		最大值	$6\text{lcy} + 2\text{Bcy} + 1\text{Pcy} + 2(\text{m1} + \text{m2}) + \text{m3}$	$7\text{lcy} + 2(\text{m1} + \text{m2}) + \text{m3}$	$6\text{lcy} + 1\text{Pcy} + 2(\text{m1} + \text{m2}) + \text{m3}$	$6\text{lcy} + 3\text{Bcy} + 1\text{Pcy} + 2(\text{m1} + \text{m2}) + \text{m3}$	$6\text{lcy} + 4\text{Bcy} + 2(\text{m1} + \text{m2}) + \text{m3}$	$6\text{lcy} + 2\text{Bcy} + 2(\text{m1} + \text{m2}) + \text{m3}$	200MHz 运行时 *1*2; 0.060 ~ 0.130μs
	有寄存器 存储体。 无寄存器 存储体上 溢。	最小值	—	—	$5\text{lcy} + 1\text{Pcy} + \text{m1} + \text{m2}$	$5\text{lcy} + 3\text{Bcy} + 1\text{Pcy} + \text{m1} + \text{m2}$	$5\text{lcy} + 4\text{Bcy} + \text{m1} + \text{m2}$	$5\text{lcy} + 2\text{Bcy} + \text{m1} + \text{m2}$	200MHz 运行时 *1*2; 0.070 ~ 0.110μs
		最大值	—	—	$14\text{lcy} + 1\text{Pcy} + \text{m1} + \text{m2}$	$14\text{lcy} + 3\text{Bcy} + 1\text{Pcy} + \text{m1} + \text{m2}$	$14\text{lcy} + 4\text{Bcy} + \text{m1} + \text{m2}$	$14\text{lcy} + 2\text{Bcy} + \text{m1} + \text{m2}$	200MHz 运行时 *1*2; 0.120 ~ 0.155μs
	有寄存器 存储体。 有寄存器 存储体上 溢。	最小值	—	—	$5\text{lcy} + 1\text{Pcy} + \text{m1} + \text{m2}$	$5\text{lcy} + 3\text{Bcy} + 1\text{Pcy} + \text{m1} + \text{m2}$	$5\text{lcy} + 4\text{Bcy} + \text{m1} + \text{m2}$	$5\text{lcy} + 2\text{Bcy} + \text{m1} + \text{m2}$	200MHz 运行时 *1*2; 0.065 ~ 0.110μs
		最大值	—	—	$5\text{lcy} + 1\text{Pcy} + \text{m1} + \text{m2} + 19(\text{m4})$	$5\text{lcy} + 3\text{Bcy} + 1\text{Pcy} + \text{m1} + \text{m2} + 19(\text{m4})$	$5\text{lcy} + 4\text{Bcy} + \text{m1} + \text{m2} + 19(\text{m4})$	$5\text{lcy} + 2\text{Bcy} + \text{m1} + \text{m2} + 19(\text{m4})$	200MHz 运行时 *1*2; 0.160 ~ 0.205μs

【注】 m1 ~ m4 为存取以下存储器时所需的状态数。

m1 : 读取向量地址 (读取长字)

m2 : 保存SR (写入长字)

m3 : 保存PC (写入长字)

m4 : 从存储体对象寄存器 (R0 ~ R14、GBR、MACH、MACL、PR) 的堆栈返回

*1 : m1=m2=m3=m4=1lcy 时

*2 : (lφ、Bφ、Pφ) = (200MHz、66MHz、33MHz) 时

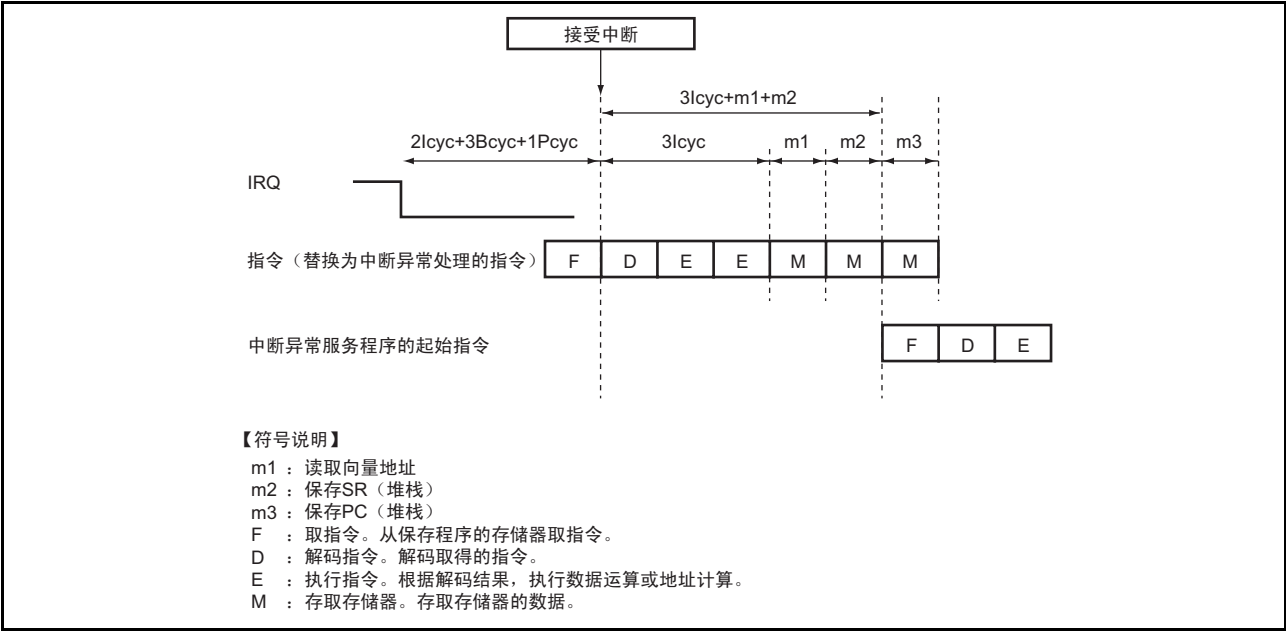


图 6.4 接受 IRQ 中断时的流水线运行例 (无寄存器存储体)

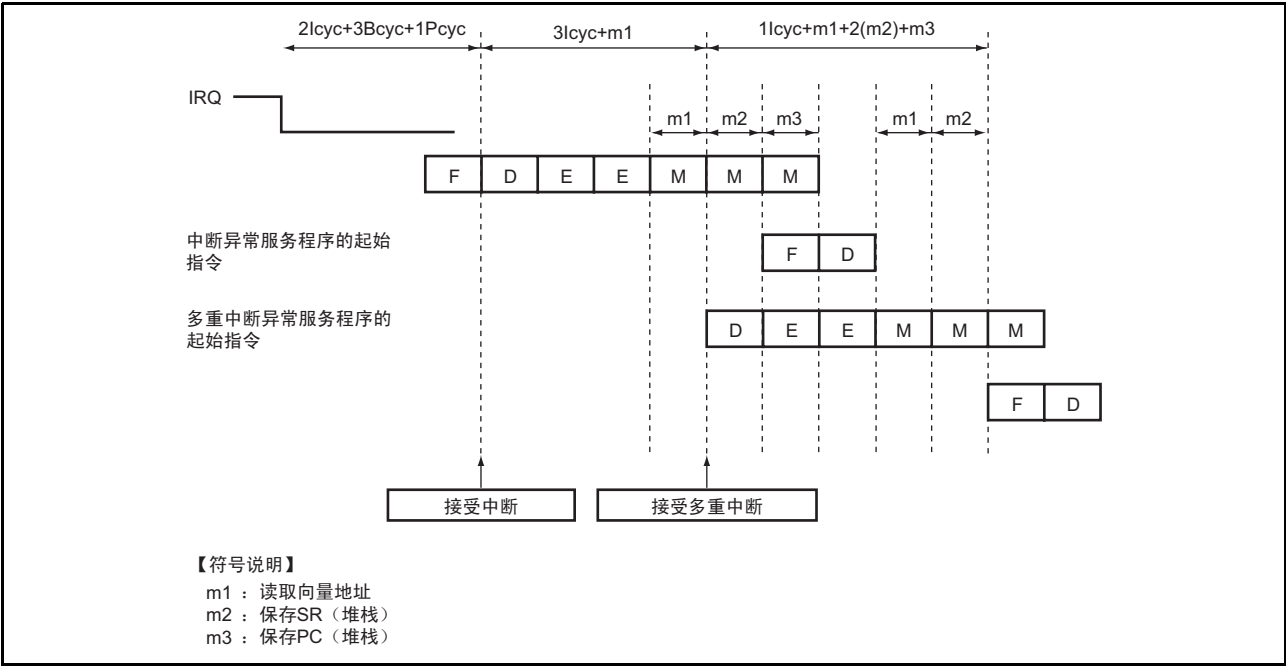


图 6.5 多重中断时的流水线运行例 (无寄存器存储体)

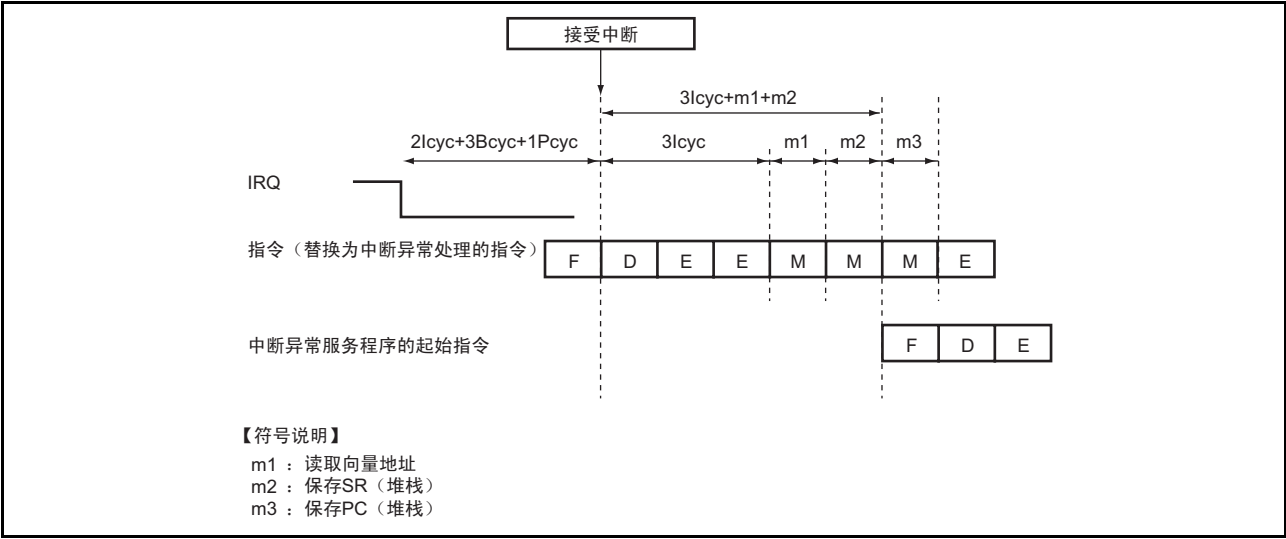


图 6.6 接受 IRQ 中断时的流水线运行例 (有寄存器存储体、无寄存器存储体上溢)

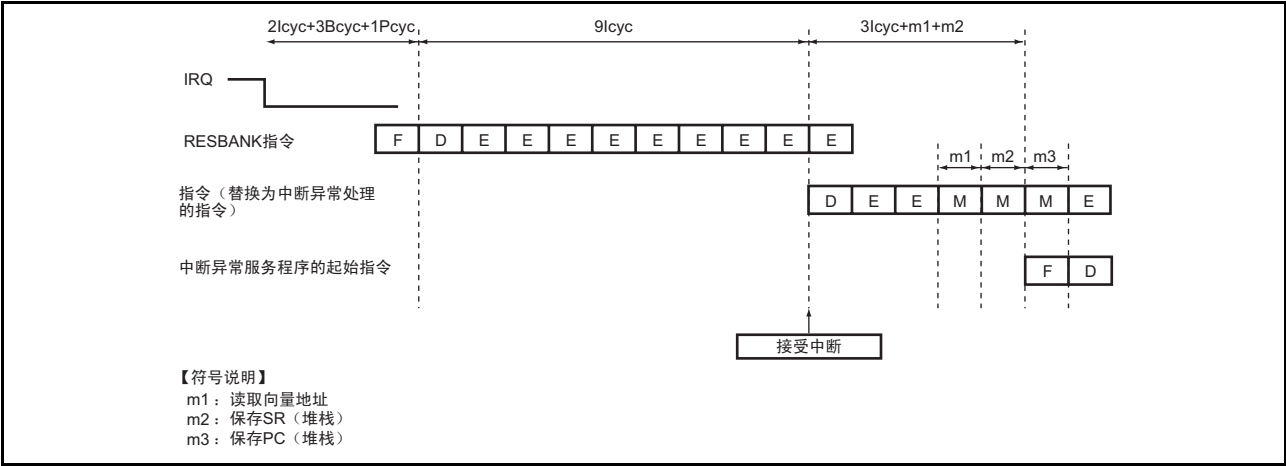


图 6.7 执行 RESBANK 指令过程中接受中断时的流水线运行例 (有寄存器存储体、无寄存器存储体上溢)

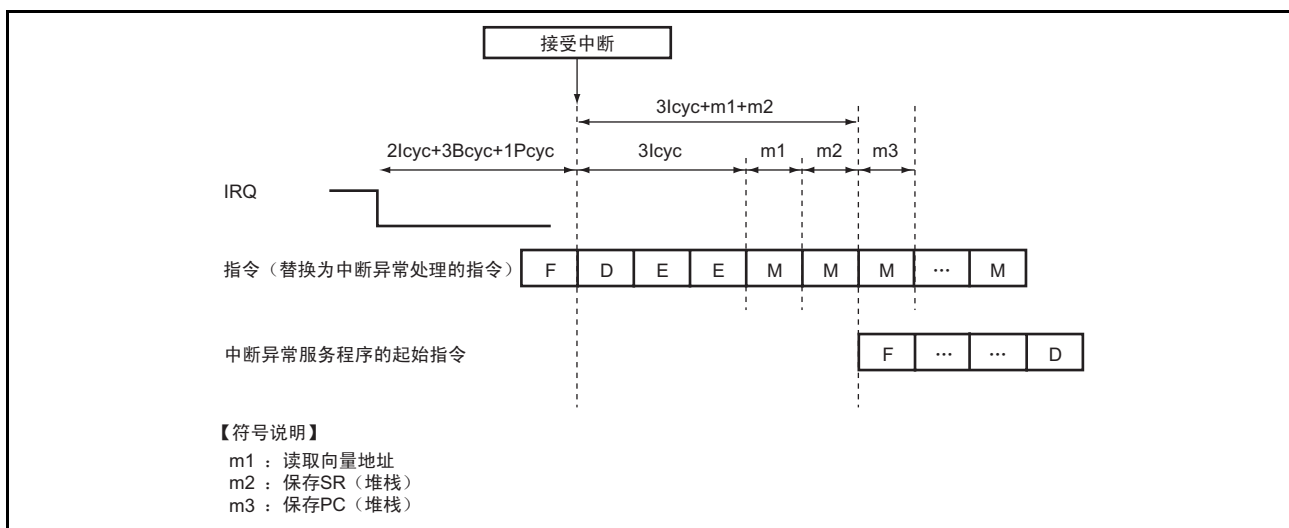


图 6.8 接受 IRQ 中断时的流水线运行例 (有寄存器存储体、有寄存器存储体上溢)

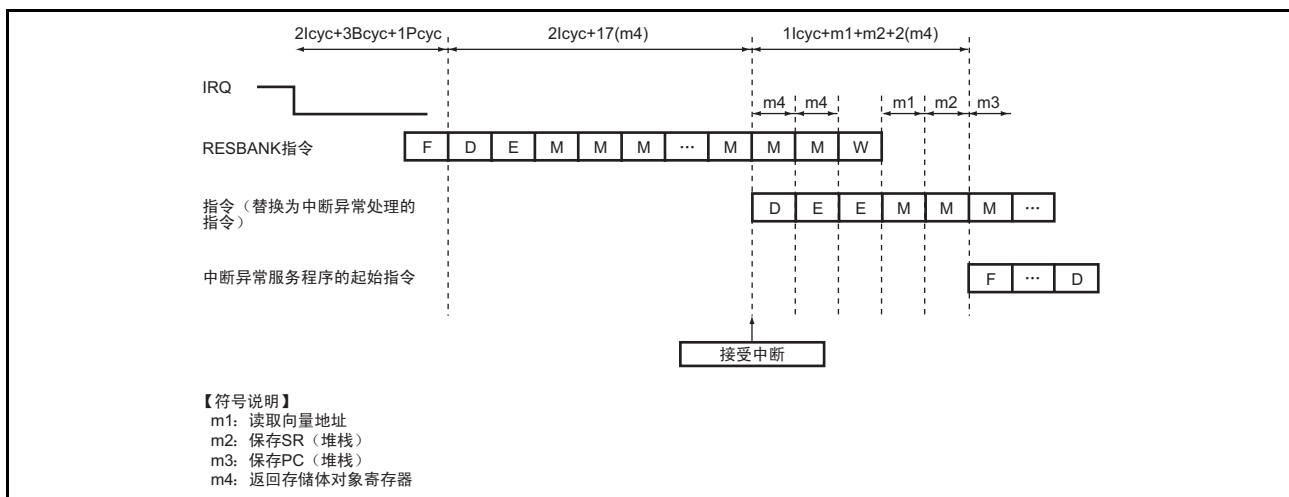


图 6.9 指令 RESBANK 执行过程中接受中断时的流水线运行例 (有寄存器存储体、有寄存器存储体上溢)

6.8 寄存器存储器

本 LSI 内置 15 个寄存器存储体，这些寄存器存储体可在中断处理的同时，高速执行寄存器的保存与返回。寄存器存储体的结构如图 6.10 所示。

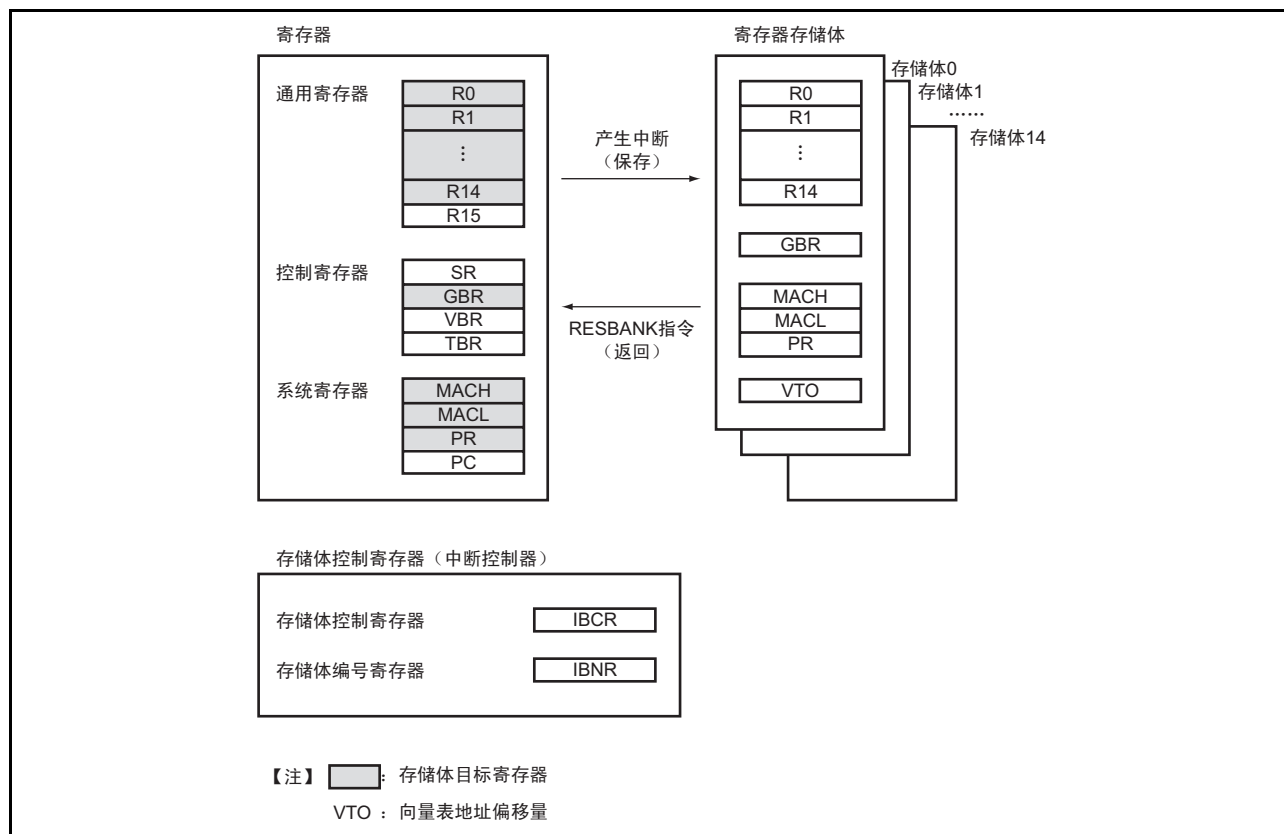


图 6.10 寄存器存储体结构的概要

6.8.1 存储体的对象寄存器与输入 / 输出方式

(1) 存储体的对象寄存器

将通用寄存器 (R0 ~ R14)、全局基址寄存器 (GBR)、乘加寄存器 (MACH, MACL)、过程寄存器 (PR) 与向量表地址偏移量作为存储体的对象。

(2) 存储体的输入 / 输出方式

寄存器存储体有 15 个存储体 (存储体 0 ~ 存储体 14)。寄存器存储体为先进后出 (FILO) 式的堆栈，按顺序从存储体 0 开始保存，从最后保存的存储体返回。

6.8.2 存储体保存、返回的运行

(1) 向存储体保存

向寄存器存储体保存的运行如图 6.11 所示。产生中断并允许 CPU 接受的中断使用寄存器存储体时，如下运行：

- (a) 将产生中断前的存储体编号寄存器 (IBNR) 的存储体编号位 (BN) 的值设置为 i。
- (b) 将寄存器 R0~R14、GBR、MACH、MACL、PR 和已接受中断的向量表地址偏移量 (VTO) 保存至 BN 所示的存储体 i。
- (c) 将 BN 的值+1。

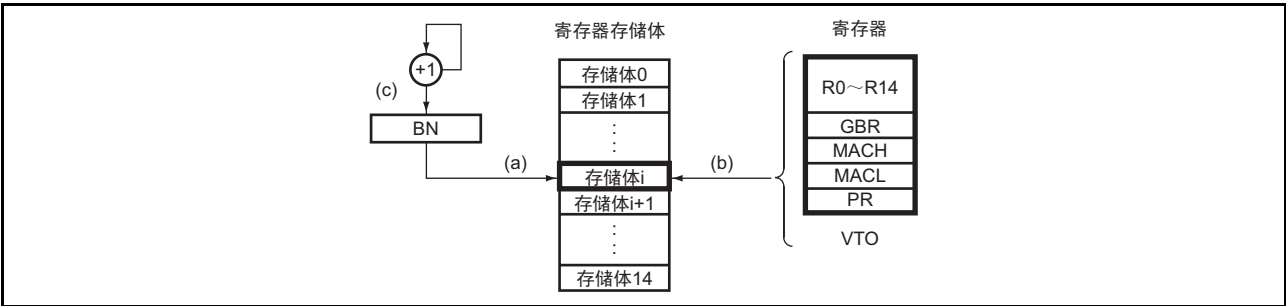


图 6.11 向存储体保存的运行

向寄存器存储体保存的时序如图 6.12 所示。在从开始中断异常处理到开始取中断异常服务程序的起始指令之前，执行向寄存器存储体的保存。

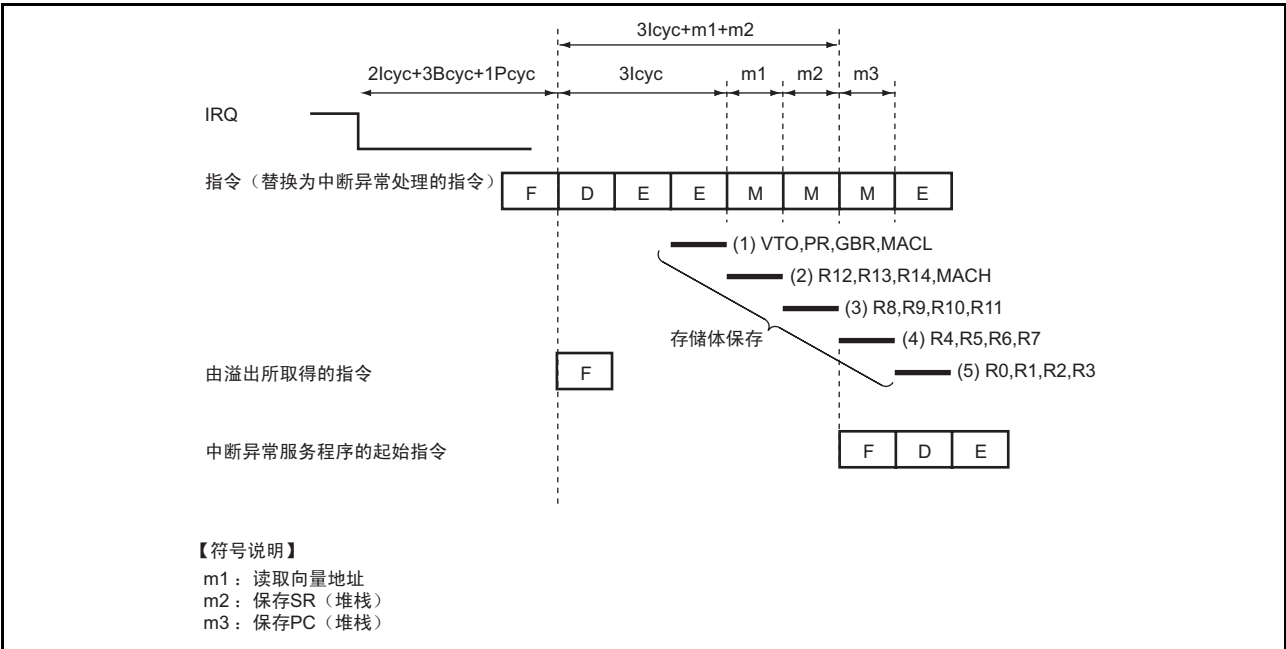


图 6.12 向存储体保存的时序

(2) 从存储体返回

返回已保存至存储体的数据时，使用存储体返回指令 RESBANK。在中断异常服务程序的最后，通过 RESBANK 指令从存储体返回后，必须由 RTE 指令从中断异常服务程序返回。

6.8.3 在保存至所有存储体的状态下执行的保存、返回

在保存至寄存器存储体所有存储体的状态下，产生中断，并允许 CPU 接受的中断使用寄存器存储体时，如果存储体编号寄存器 (IBNR) 的 BOVE 位为 0，则代替寄存器存储体自动压栈。另外，如果 IBNR 的 BOVE 位置 1，则产生寄存器存储体上溢异常，不执行压栈。

压栈、出栈的运行如下：

(1) 压栈

1. 中断异常处理时，将状态寄存器 (SR)、程序计数器 (PC) 压栈。
2. 将存储体对象寄存器 (R0～R14、GBR、MACH、MACL、PR) 压栈。压栈的寄存器顺序为 MACL、MACH、GBR、PR、R14、R13、……、R1、R0。
3. 将 SR 的寄存器存储体上溢位 (BO) 置 1。
4. 存储体编号寄存器 (IBNR) 的存储体编号位 (BN) 保持为最大值 15。

(2) 出栈

在 SR 的寄存器存储体上溢位 (BO) 置 1 的状态下，如果执行存储体返回指令 RESBANK，则如下运行：

1. 将存储体对象寄存器 (R0～R14、GBR、MACH、MACL、PR) 出栈。出栈的寄存器顺序为 R0、R1、……、R13、R14、PR、GBR、MACH、MACL。
2. 存储体编号寄存器 (IBNR) 的存储体编号位 (BN) 保持为最大值 15。

6.8.4 寄存器存储体异常

寄存器存储体异常 (寄存器存储体错误) 有寄存器存储体上溢和寄存器存储体下溢 2 种。

(1) 寄存器存储体上溢

在保存至寄存器存储体的所有存储体的状态下，产生中断并允许 CPU 接受的中断使用寄存器存储体时，如果将存储体编号寄存器 (IBNR) 的 BOVE 位置 1，则产生寄存器存储体上溢。此时，存储体编号寄存器 (IBNR) 的存储体编号位 (BN) 保持为存储体数 15，且不执行向寄存器存储体的保存。

(2) 寄存器存储体下溢

在完全未保存至寄存器存储体的状态下，执行寄存器存储体返回指令时，产生寄存器存储体下溢。此时，R0～R14、GBR、MACH、MACL、PR 的值不变。另外，存储体编号寄存器 (IBNR) 的存储体编号位 (BN) 保持为 0。

6.8.5 寄存器存储体错误异常处理

产生寄存器存储体错误时，开始寄存器存储体错误异常处理。此时 CPU 的运行如下：

1. 从已产生的寄存器存储体错误对应的异常处理向量表取出异常服务程序的起始地址。
2. 将状态寄存器 (SR) 压栈。
3. 将程序计数器 (PC) 压栈。寄存器存储体上溢时保存的 PC 值为最后执行指令的下一条指令的起始地址；寄存器存储体下溢时保存的 PC 值为该 RESBANK 指令的起始地址。另外，为了防止寄存器存储体上溢时产生多重中断，对状态寄存器 (SR) 的中断屏蔽级位 (I3～I0) 写入引起寄存器存储体上溢的中断级。
4. 从异常服务程序的起始地址开始执行程序。

6.9 通过中断请求信号传送数据

可通过中断请求信号启动 DMAC，并执行数据传送。

中断源中指定为 DMAC 启动源的中断源，不输入至 INTC 而是被屏蔽。屏蔽条件如下：

屏蔽条件 = $DME \cdot (DE0 \cdot \text{中断源选择 } 0 + DE1 \cdot \text{中断源选择 } 1 + DE2 \cdot \text{中断源选择 } 2 + DE3 \cdot \text{中断源选择 } 3 + DE4 \cdot \text{中断源选择 } 4 + DE5 \cdot \text{中断源选择 } 5 + DE6 \cdot \text{中断源选择 } 6 + DE7 \cdot \text{中断源选择 } 7)$

中断控制框图如图 6.13 所示。

在此，DME 为 DMAC 的 DMAOR 的 bit0， DE_n ($n=0 \sim 7$) 为 DMAC 的 CHCR0 $\sim$ CHCR7 的 bit0。详情参阅“第 10 章 直接存储器存取控制器 (DMAC)”。

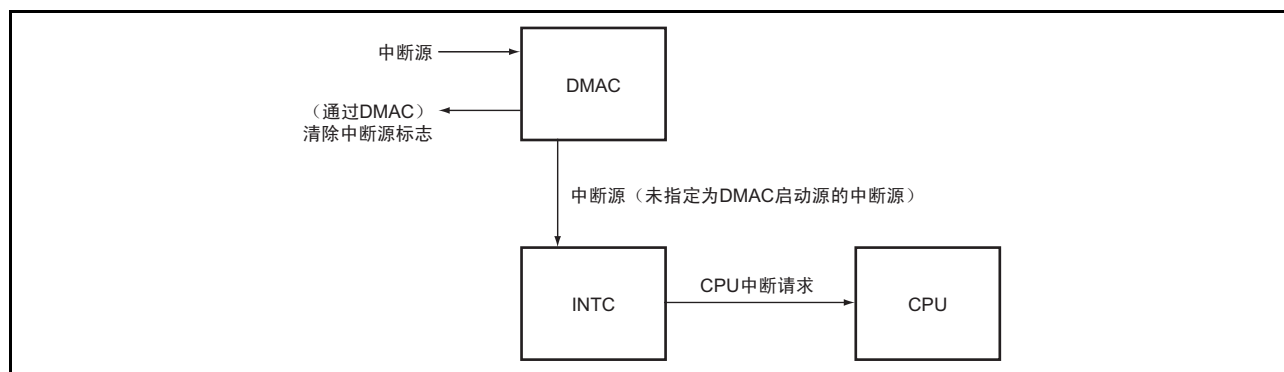


图 6.13 中断控制框图

6.9.1 中断请求信号作为 CPU 中断源而非 DMAC 启动源时

1. 不选择 DMAC 启动源、或将 DME 位清 0。另外，选择 DMAC 启动源时，将 DMAC 的该通道的 DE 位清 0。
2. 产生中断时，向 CPU 请求中断。
3. 在中断异常服务程序，CPU 清除中断源并执行必要的处理。

6.9.2 中断请求信号作为 DMAC 启动源而非 CPU 中断源时

1. 选择 DMAC 启动源，并设定 DE=1、DME=1。与中断优先级设定寄存器的设定无关，屏蔽 CPU 中断源。
2. 产生中断时，向 DMAC 发送启动源。
3. 传送时，DMAC 清除启动源。

6.10 使用时的注意事项

6.10.1 清除中断源的时序

必须在中断异常服务程序中清除中断源标志。从清除中断源标志后到实际取消 CPU 的中断源，需要的时间为表 6.5 记载的“从产生中断请求，到比较中断控制器判断优先顺序及与 SR 的屏蔽位后，向 CPU 发送中断请求信号的时间”。因此，为了避免再次接受应清除的中断源，需在清除后读取 * 中断源标志，然后再执行 RTE 指令。

【注】 * 清除 USB 的中断源标志时，必须在清除后读取 3 次中断源标志。

6.10.2 $\overline{\text{IRQOUT}}$ 的无效时序

中断控制器接受中断请求时，从 $\overline{\text{IRQOUT}}$ 引脚输出低电平，并跳转至中断异常服务程序的起始地址后，从 $\overline{\text{IRQOUT}}$ 引脚输出高电平。

但是，中断控制器接受中断请求，并从 $\overline{\text{IRQOUT}}$ 引脚输出低电平后，如果在跳转至中断异常服务程序的起始地址之前取消中断请求，则从 $\overline{\text{IRQOUT}}$ 引脚输出低电平直到由下一个中断请求跳转至中断异常服务程序的起始地址。

第 7 章 用户断点控制器 (UBC)

用户断点控制器 (UBC) 提供简化程序调试的功能, 使用此功能可轻松编写自监控调试程序, 不使用在线仿真器, 也可通过本 LSI 轻松调试程序。作为可能 UBC 设定的断点条件有, 支持取指令或读取 / 写入数据 (读取 / 写入数据时为总线周期 (CPU、DMAC))、数据长度、数据内容、地址值及取指令时的停止时序。本 LSI 采用哈佛体系结构, 因此 CPU 总线 (C 总线) 的取指令对取指令总线 (F 总线) 发行总线周期; C 总线的数据存取对存储器存取总线 (M 总线) 发行总线周期。内部总线 (I 总线) 有内部 CPU 总线与内部 DMA 总线, CPU 对内部 CPU 总线、DMA 对内部 DMA 总线发行总线周期。UBC 监控这些 C 总线与 I 总线。

7.1 特点

1. 可设定以下断点比较条件:
断点通道数: 2 个通道 (通道 0 和 1)
可单独设定通道 0、1 的用户断点。
 - 地址
可按位屏蔽 32 位地址的比较。
可从 4 种地址总线 (F 总线地址 (FAB)、M 总线地址 (MAB)、内部 CPU 总线地址 (ICAB)、内部 DMA 总线地址 (IDAB)) 选择。
 - 数据
可按位屏蔽 32 位数据的比较。
可从 3 种数据总线 (M 总线数据 (MDB)、内部 CPU 总线数据 (ICDB)、内部 DMA 总线数据 (IDDB)) 选择其一。
 - 选择 I 总线时的总线选择
内部 CPU 总线、内部 DMA 总线
 - 总线周期
取指令 (仅限选择 C 总线时) 或数据存取
 - 读取或写入
 - 操作数长度
支持字节、字及长字。
2. 在取指令周期, 可指定在指令执行前还是指令执行后开始执行用户断点中断处理。
3. 可将断点条件成立信号输出至 UBCTRG 引脚。

UBC 框图如图 7.1 所示:

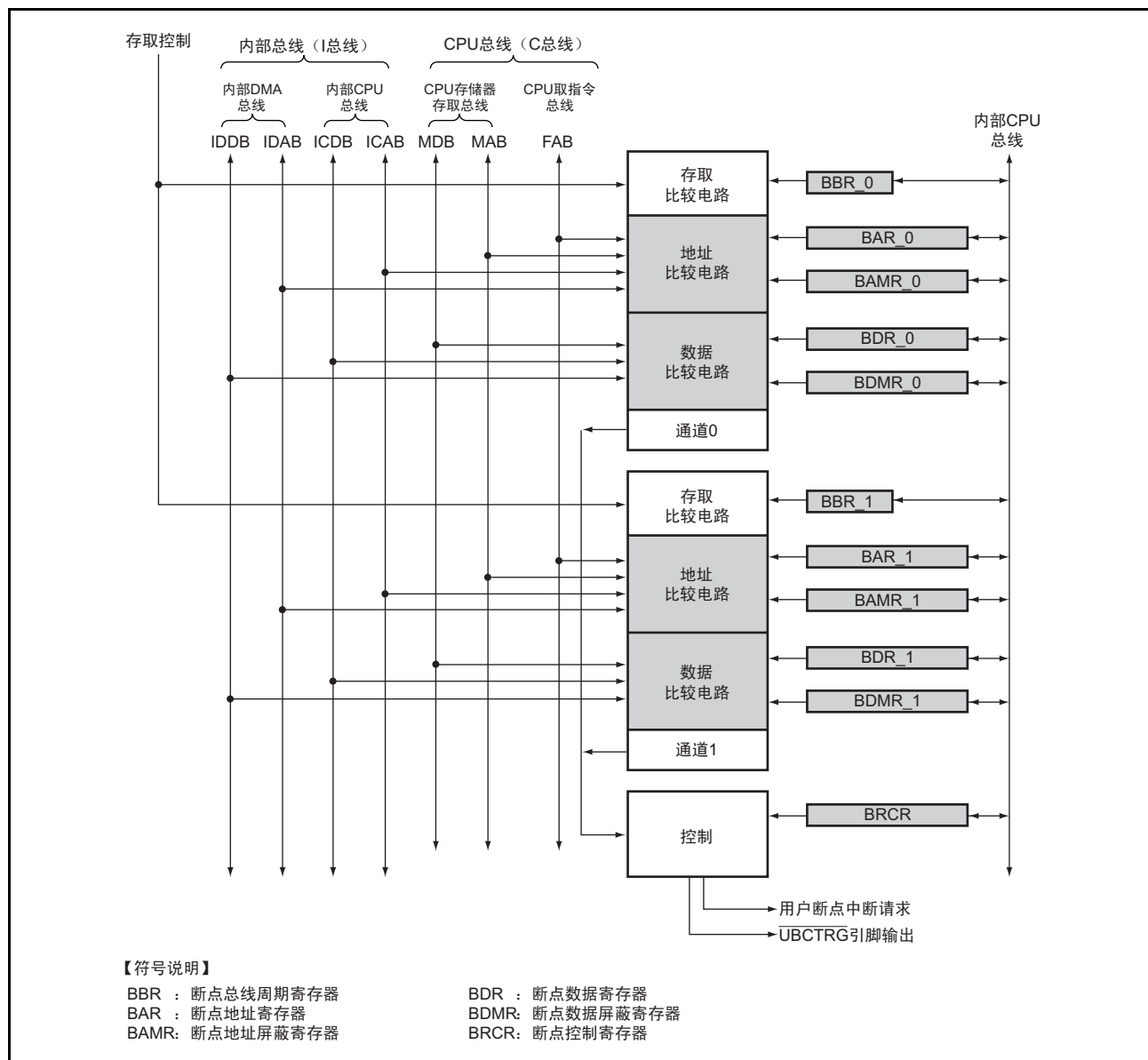


图 7.1 UBC 框图

7.2 输入 / 输出引脚

UBC 的引脚结构如表 7.1 所示：

表 7.1 引脚结构

名称	引脚名称	输入 / 输出	功 能
UBC 触发	UBCTRG	输出	表示在 UBC 的通道 0 或 1 设定条件成立。

7.3 寄存器说明

UBC 有以下寄存器。各通道有 5 个控制寄存器及 1 个通道 0、1 通用的控制寄存器。各通道的寄存器表示如下：如通道 0 的寄存器 BAR 表示为 BAR_0。

表 7.2 寄存器结构

通道	寄存器名称	简称	R/W	初始值	地址	存取长度
0	断点地址寄存器 _0	BAR_0	R/W	H'00000000	H'FFFC0400	32
	断点地址屏蔽寄存器 _0	BAMR_0	R/W	H'00000000	H'FFFC0404	32
	断点总线周期寄存器 _0	BBR_0	R/W	H'0000	H'FFFC04A0	16
	断点数据寄存器 _0	BDR_0	R/W	H'00000000	H'FFFC0408	32
	断点数据屏蔽寄存器 _0	BDMR_0	R/W	H'00000000	H'FFFC040C	32
1	断点地址寄存器 _1	BAR_1	R/W	H'00000000	H'FFFC0410	32
	断点地址屏蔽寄存器 _1	BAMR_1	R/W	H'00000000	H'FFFC0414	32
	断点总线周期寄存器 _1	BBR_1	R/W	H'0000	H'FFFC04B0	16
	断点数据寄存器 _1	BDR_1	R/W	H'00000000	H'FFFC0418	32
	断点数据屏蔽寄存器 _1	BDMR_1	R/W	H'00000000	H'FFFC041C	32
通用	断点控制寄存器	BRCCR	R/W	H'00000000	H'FFFC04C0	32

7.3.1 断点地址寄存器 (BAR)

BAR 为 32 位可读取 / 写入的寄存器，指定作为各通道断点条件的地址。作为断点条件对象的地址总线有 4 种，由断点总线周期寄存器 (BBR) 的控制位的 CD[1:0] 与 CP[1:0] 选择。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BA31	BA30	BA29	BA28	BA27	BA26	BA25	BA24	BA23	BA22	BA21	BA20	BA19	BA18	BA17	BA16
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BA15	BA14	BA13	BA12	BA11	BA10	BA9	BA8	BA7	BA6	BA5	BA4	BA3	BA2	BA1	BA0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
31 ~ 0	BA31 ~ BA0	均为 0	R/W	断点地址 保存指定断点条件的 CPU 地址总线 (FAB 或 MAB) 或内部地址总线 (ICAB 或 IDAB) 的地址。 由 BBR 选择 C 总线及取指令周期时，对 BA31 ~ BA0 指定 FAB 的地址。 由 BBR 选择 C 总线及数据存取周期时，对 BA31 ~ BA0 指定 MAB 的地址。 由 BBR 选择 I 总线及内部 CPU 总线时，对 BA31 ~ BA0 指定 ICAB 的地址。 由 BBR 选择 I 总线及内部 DMA 总线时，对 BA31 ~ BA0 指定 IDAB 的地址。

【注】 设定取指令周期为断点条件时，必须将 BAR 的 LSB 清 0。

7.3.2 断点地址屏蔽寄存器 (BAMR)

BAMR 为 32 位可读取 / 写入的寄存器，在 BAR 指定的断点地址位中，指定要屏蔽的位。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BAM31	BAM30	BAM29	BAM28	BAM27	BAM26	BAM25	BAM24	BAM23	BAM22	BAM21	BAM20	BAM19	BAM18	BAM17	BAM16
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BAM15	BAM14	BAM13	BAM12	BAM11	BAM10	BAM9	BAM8	BAM7	BAM6	BAM5	BAM4	BAM3	BAM2	BAM1	BAM0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
31 ~ 0	BAM31 ~ BAM0	均为 0	R/W	断点地址屏蔽 在 BAR (BA31 ~ BA0) 指定的断点地址位中，指定要屏蔽的位。 0: 断点地址位 BAn 包含在断点条件中 1: 断点地址位 BAn 被屏蔽，且不包含在断点条件中 【注】 n=31 ~ 0

7.3.3 断点数据寄存器 (BDR)

BDR 为 32 位可读取 / 写入的寄存器。作为断点条件对象的数据总线有 3 种，由断点总线周期寄存器 (BBR) 的控制位 CD[1:0] 及 CP[1:0] 选择。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BD31	BD30	BD29	BD28	BD27	BD26	BD25	BD24	BD23	BD22	BD21	BD20	BD19	BD18	BD17	BD16
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BD15	BD14	BD13	BD12	BD11	BD10	BD9	BD8	BD7	BD6	BD5	BD4	BD3	BD2	BD1	BD0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
31 ~ 0	BD31 ~ BD0	均为 0	R/W	断点数据位 保存指定断点条件的数据。 由 BBR 选择 C 总线时，对 BD31 ~ BD0 指定 MDB 的断点数据。 由 BBR 选择 I 总线及内部 CPU 总线时，对 BD31 ~ BD0 指定 ICDB 的地址。 由 BBR 选择 I 总线及内部 DMA 总线时，对 BD31 ~ BD0 指定 IDDB 的地址。

- 【注】 1. 断点条件包含数据总线的值时，必须指定操作数长度。
2. 指定字节长度为断点条件时，作为 BDR 的断点数据，必须对 bit31 ~ 24、23 ~ 16、15 ~ 8、7 ~ 0 设定相同的字节数据；指定字长度作为断点条件时，必须对 bit31 ~ 16、15 ~ 0 设定相同的字数据。

7.3.4 断点数据屏蔽寄存器 (BDMR)

BDMR 为 32 位可读取 / 写入的寄存器，在 BDR 指定的断点数据位中，指定要屏蔽的位。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BDM31	BDM30	BDM29	BDM28	BDM27	BDM26	BDM25	BDM24	BDM23	BDM22	BDM21	BDM20	BDM19	BDM18	BDM17	BDM16
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BDM15	BDM14	BDM13	BDM12	BDM11	BDM10	BDM9	BDM8	BDM7	BDM6	BDM5	BDM4	BDM3	BDM2	BDM1	BDM0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
31 ~ 0	BDM31 ~ BDM0	均为 0	R/W	断点数据屏蔽 在 BDR (BD31 ~ BD0) 指定的断点数据位中，指定要屏蔽的位。 0: 断点数据位 BDn 包含在断点条件中 1: 断点数据位 BDn 被屏蔽，且不包含在断点条件中 【注】 n=31 ~ 0

- 【注】 1. 断点条件包含数据总线的值时，必须指定操作数长度。
2. 指定字节长度为断点条件时，作为 BDMR 的断点屏蔽数据，必须对 bit31 ~ 24、23 ~ 16、15 ~ 8、7 ~ 0 设定相同的字节数据；指定字长度作为断点条件时，必须对 bit31 ~ 16、15 ~ 0 设定相同的字数据。

7.3.5 断点总线周期寄存器 (BBR)

BBR 为 16 位可读取 / 写入的寄存器, 指定 (1) 禁止 / 允许用户断点中断请求、(2) 包含 / 不包含数据总线的值、(3) 内部 CPU 总线或内部 DMA 总线、(4) C 总线周期或 I 总线周期、(5) 取指令或数据存取、(6) 读取或写入、(7) 操作数长度等为断点条件。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	UBID	DBE	-	-	CP[1:0]	CD[1:0]	ID[1:0]	RW[1:0]	SZ[1:0]					
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15、14	—	均为 0	R	保留位 读取值、写入值总是为 0。
13	UBID	0	R/W	用户断点中断禁止 指定禁止 / 允许条件匹配时产生用户断点中断请求。 0: 允许用户断点中断请求 1: 禁止用户断点中断请求
12	DBE	0	R/W	数据断点允许 选择断点条件是否包含数据总线条件。 0: 断点条件不包含数据总线条件 1: 断点条件包含数据总线条件
11、10	—	均为 0	R	保留位 读取值、写入值总是为 0。
9、8	CP[1:0]	00	R/W	I 总线的总线选择 选择断点条件的总线周期为 I 总线时的总线。但, 总线周期为 C 总线周期时, 本位无效 (仅限 CPU 周期)。 00: 不执行条件比较 01: 断点条件为内部 CPU 总线 10: 断点条件为内部 DMA 总线 11: 断点条件为内部 CPU 总线
7、6	CD[1:0]	00	R/W	选择 C 总线周期 / I 总线周期 选择 C 总线周期或 I 总线周期为断点条件的总线周期。 00: 不执行条件比较 01: 断点条件为 C 总线 (F 总线、M 总线) 周期 10: 断点条件为 I 总线周期 11: 断点条件为 C 总线 (F 总线、M 总线) 周期
5、4	ID[1:0]	00	R/W	选择取指令 / 数据存取 选择取指令周期或数据存取周期为断点条件的总线周期。选择取指令周期时, 必须选择 C 总线周期。 00: 不执行条件比较 01: 断点条件为取指令周期 10: 断点条件为数据存取周期 11: 断点条件为取指令周期或数据存取周期
3、2	RW[1:0]	00	R/W	选择读取 / 写入 选择读取周期或写入周期为断点条件的总线周期。 00: 不执行条件比较 01: 断点条件为读取周期 10: 断点条件为写入周期 11: 断点条件为读取周期或写入周期

位	位名称	初始值	R/W	说 明
1、0	SZ[1:0]	00	R/W	选择操作数长度 选择断点条件总线周期的操作数长度。 00: 断点条件不包含操作数长度 01: 断点条件为字节存取 10: 断点条件为字存取 11: 断点条件为长字存取

7.3.6 断点控制寄存器 (BRCCR)

BRCCR 设定以下条件:

1. 指定在指令执行前还是指令执行后开始执行由取指令周期引起的用户断点中断异常处理。
2. 设定断点条件匹配时 $\overline{\text{UBCTRG}}$ 输出的脉宽。
3. 设定断点条件匹配时是否执行 $\overline{\text{UBCTRG}}$ 输出。

BRCCR 为 32 位可读取 / 写入的寄存器, 具有设定断点条件匹配标志与其他断点条件的位。只有 bit15 ~ 12 的条件匹配标志为写入 1 无效 (保持之前的值) 仅可写入 0 的位, 因此, 清除时, 必须对清除的标志位写入 0; 对其他标志位写入 1。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	UTOD1	UTOD0	CKS[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SCMFC	SCMFC	SCMFD	SCMFD	-	-	-	-	-	PCB1	PCB0	-	-	-	-	-
	0	1	0	1												
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R	R	R	R	R

位	位名称	初始值	R/W	说 明
31 ~ 20	—	均为 0	R	保留位 读取值、写入值总是为 0。
19	UTOD1	0	R/W	$\overline{\text{UBCTRG}}$ 输出禁止 1 通道 1 的条件匹配时, 指定是否执行 $\overline{\text{UBCTRG}}$ 输出。 0: 通道 1 的条件匹配时, 向 $\overline{\text{UBCTRG}}$ 输出 1: 通道 1 的条件匹配时, 不向 $\overline{\text{UBCTRG}}$ 输出
18	UTOD0	0	R/W	$\overline{\text{UBCTRG}}$ 输出禁止 0 通道 0 的条件匹配时, 指定是否执行 $\overline{\text{UBCTRG}}$ 输出。 0: 通道 0 的断点条件匹配时, 向 $\overline{\text{UBCTRG}}$ 输出。 1: 通道 0 的断点条件匹配时, 不向 $\overline{\text{UBCTRG}}$ 输出。
17、16	CKS[1:0]	00	R/W	选择时钟 条件匹配时, 指定对 $\overline{\text{UBCTRG}}$ 输出的脉宽。 00: $\overline{\text{UBCTRG}}$ 的脉宽为 1 个总线时钟周期 01: $\overline{\text{UBCTRG}}$ 的脉宽为 2 个总线时钟周期 10: $\overline{\text{UBCTRG}}$ 的脉宽为 4 个总线时钟周期 11: $\overline{\text{UBCTRG}}$ 的脉宽为 8 个总线时钟周期

位	位名称	初始值	R/W	说 明
15	SCMFC0	0	R/W	C 总线周期条件匹配标志 0 如果满足对通道 0 设定的断点条件 C 总线周期条件, 则此标志置 1。为了清除此标志, 对此位写入 0。 0: 通道 0 的 C 总线周期条件不匹配 1: 通道 0 的 C 总线周期条件匹配
14	SCMFC1	0	R/W	C 总线周期条件匹配标志 1 如果满足对通道 1 设定的断点条件 C 总线周期条件, 则此标志置 1。为了清除此标志, 对此位写入 0。 0: 通道 1 的 C 总线周期条件不匹配 1: 通道 1 的 C 总线周期条件匹配
13	SCMFD0	0	R/W	I 总线周期条件匹配标志 0 如果满足对通道 0 设定的断点条件 I 总线周期条件, 则此标志置 1。为了清除此标志, 对此位写入 0。 0: 通道 0 的 I 总线周期条件不匹配 1: 通道 0 的 I 总线周期条件匹配
12	SCMFD1	0	R/W	I 总线周期条件匹配标志 1 如果满足通道 1 设定的断点条件 I 总线周期条件时, 则此标志置 1。为了清除此标志, 对此位写入 0。 0: 通道 1 的 I 总线周期条件不匹配 1: 通道 1 的 I 总线周期条件匹配
11 ~ 7	—	均为 0	R	保留位 读取值、写入值总是为 0。
6	PCB1	0	R/W	PC 断点选择 1 选择对通道 1 取指令周期的断点时序在指令执行前还是指令执行后。 0: 通道 1 的 PC 断点设定在指令执行前 1: 通道 1 的 PC 断点设定在指令执行后
5	PCB0	0	R/W	PC 断点选择 0 选择对通道 0 取指令周期的断点时序在指令执行前还是指令执行后。 0: 通道 0 的 PC 断点设定在指令执行前 1: 通道 0 的 PC 断点设定在指令执行后
4 ~ 0	—	均为 0	R	保留位 读取值、写入值总是为 0。

7.4 运行说明

7.4.1 用户断点运行的流程

从设定断点条件到用户断点中断异常处理的运行流程如下：

1. 给断点地址寄存器 (BAR) 指定断点地址；给断点地址屏蔽寄存器 (BAMR) 指定屏蔽地址；给断点数据寄存器 (BDR) 指定断点数据；给断点数据屏蔽寄存器 (BDMR) 指定屏蔽数据；给断点总线周期寄存器 (BBR) 指定总线断点条件。BBR 的 3 对控制位，即 C 总线周期/I 总线周期选择控制位、取指令/数据存取选择控制位及读取/写入选择控制位中，任意 1 对为 00 时，不产生用户断点。在 BRCCR 的位指定断点控制。设定完所有与断点相关的寄存器后，设定 BBR。必须读取最后写入的寄存器后，再执行转移。从转移目标的指令开始新写入寄存器的值有效。
2. 满足断点条件且允许用户断点中断请求时，UBC 在向 INTC 通知用户断点中断请求的同时，将各通道的 C 总线条件匹配标志 (SCMFC)、I 总线条件匹配标志 (SCMFD) 分别置位，并通过 CKS[1:0] 位设定的脉宽，向 UBCTRG 引脚输出。此外，通过将 BBR 的 UBID 位置 1，可不请求用户断点中断而在外部监控触发输出。
3. 接受用户断点中断请求信号时，INTC 判断优先顺序。用户断点中断的优先级为 15，因此，状态寄存器 (SR) 的中断屏蔽级位 (I3~I0) 小于等于 14 级时，接受该中断；I3~I0 位为 15 级时，不接受用户断点中断，但执行条件判断，如果匹配，则条件匹配标志置位。有关优先顺序判断，详情参阅“第 6 章 中断控制器 (INTC)”。
4. 为了检查设定条件是否匹配，可使用相应的条件匹配标志 (SCMFC、SCMFD)。必须在用户断点中断异常处理程序中清除条件匹配标志位。如果不清除，则会再次产生中断。
5. 通道 0 和通道 1 设定的断点有可能同时产生。此时，即使向 INTC 的用户断点中断请求只有 1 个，这 2 个断点通道匹配标志也可能同时置位。
6. 选择 I 总线为断点条件时，必须注意以下事项：
 - CPU 在 C 总线发行的存取是否在内部 CPU 总线发行，取决于高速缓存的设定。有关高速缓存条件下的 I 总线运行，详情参照“第 8 章 高速缓存”的表 8.8。
 - 对 I 总线设定断点条件时，仅监控数据存取周期，不监控取指令周期（含高速缓存更新周期）。
 - 内部 DMA 总线周期仅为数据存取周期。
 - 对 I 总线设定断点条件时，即使在 CPU 已执行指令引起的内部 CPU 总线周期产生条件匹配，也无法决定由哪条指令接受用户断点中断请求。

7.4.2 取指令周期的断点

1. 如果给断点总线周期寄存器 (BBR) 设定 C 总线/取指令/读取/字或长字，则断点条件为 FAB 总线的取指令。可通过相应通道的断点控制寄存器 (BRCCR) 的 PCB0 或 PCB1 位，选择在指令执行前还是指令执行后开始执行用户断点中断异常处理。如果设定取指令周期为断点条件，则断点地址寄存器 (BAR) 的 BA0 位必须设定为 0。如果此位设定为 1，则不产生断点。
2. 将取指令产生的用户断点设定在该指令执行前时，如果条件匹配，则取指令，并在确定执行指令时产生断点。因此，因溢出（转移或中断转移过程中被取出但未执行的指令）所取得的指令不产生断点。如果对延迟转移指令的延迟槽设定这种断点，则到转移目标的最初指令执行前，不接受用户断点中断请求。

【注】 延迟转移指令未转移时，则后续的指令不视为延迟槽。

3. 断点条件设定为在指令执行后产生断点时，执行完与断点条件匹配的指令后，在执行下一条指令前产生断点。与指令执行前的断点情况相同，被取的指令变为无效时不产生断点。如果对延迟转移指令及其延迟槽设定这些断点，则到转移目标的最初指令执行前，不接受用户断点中断请求。
4. 如果设定取指令周期，则忽视断点数据寄存器 (BDR)。因此，不可在取指令周期的断点设定断点数据。
5. 在取指令周期的断点设定 I 总线时无效。

7.4.3 数据存取周期的断点

1. 对于数据存取断点，如果指定 C 总线为断点条件，则根据已执行的指令对存取的地址（及数据）执行条件比较并产生断点。指定 I 总线为断点条件时，对由 I 总线选择指定的总线的存取周期的地址（及数据）执行条件比较并产生断点。有关在内部 CPU 总线发行的 CPU 总线周期，详情参阅“7.4.1 用户断点运行的流程”。
2. 数据存取周期地址与各操作数长度的比较条件关系，如表 7.3 所示。

表 7.3 数据存取周期地址与各操作数长度的比较条件

存取长度	比较地址
长字	比较断点地址寄存器的 bit31 ~ 2 与地址总线的 bit31 ~ 2
字	比较断点地址寄存器的 bit31 ~ 1 与地址总线的 bit31 ~ 1
字节	比较断点地址寄存器的 bit31 ~ 0 与地址总线的 bit31 ~ 0

例如，给断点地址寄存器（BAR）设定地址 H'00001003 时，满足断点条件的总线周期（假设满足其他所有条件）包括以下内容：

H'00001000 的长字存取
 H'00001002 的字存取
 H'00001003 的字节存取

3. 断点条件包含数据时
 断点条件包含数据时，给断点总线周期寄存器（BBR）指定长字、字或字节为操作数长度，并在地址条件与数据条件匹配时产生断点。此时，为了指定字节数据，必须对断点数据寄存器（BDR）与断点数据屏蔽寄存器（BDMR）的 bit31 ~ 24、bit23 ~ 16、bit15 ~ 8、bit7 ~ 0 等 4 个字节指定相同的数据。为了指定字数据，必须对 bit31 ~ 16、bit15 ~ 0 等 2 个字指定相同的数据。
4. PREF 指令用于读取无存取数据的长字。因此，PREF 指令的断点条件包含数据值时，不产生断点。
5. 选择数据存取周期时，无法确定产生断点的指令。

7.4.4 被保存的程序计数器值

接受用户断点中断请求时, 将应该重新开始执行的指令地址压栈, 并进入异常处理状态。指定 C 总线 (FAB) / 取指令为断点条件时, 可确定产生断点的指令; 指定 C 总线 / 数据存取周期或 I 总线 / 数据存取周期为断点条件时, 则不可确定产生断点的指令。

1. 指定 C 总线 (FAB) / 取指令 (指令执行前) 为断点条件时
将与断点条件匹配的指令地址压栈。不执行条件匹配的指令, 并在此前产生断点。但在延迟槽指令条件匹配时, 执行这些指令, 并将转移目标地址压栈。
2. 指定 C 总线 (FAB) / 取指令 (指令执行后) 为断点条件时
将与断点条件匹配指令的下一条指令地址压栈。执行条件匹配的指令, 并在执行下一条指令前产生断点。在延迟转移指令或延迟槽匹配时, 执行这些指令, 并将转移目标地址压栈。
3. 指定 C 总线 / 数据存取周期或 I 总线 / 数据存取周期为断点条件时
将数条与断点条件匹配的指令执行后的地址压栈。

7.4.5 使用例

(1) 指定为 C 总线取指令周期的断点条件

(例 1-1)

- 寄存器指定

BAR_0=H'00000404、BAMR_0=H'00000000、BBR_0=H'0054、BAR_1=H'00008010、
BAMR_1=H'00000006、BBR_1=H'0054、BDR_1=H'00000000、BDMR_1=H'00000000、
BRCCR=H'00000020

<通道 0 >

地址: H'00000404、地址屏蔽: H'00000000

总线周期: C 总线/取指令 (指令执行后) / 读取 (断点条件不包含操作数长度)

<通道 1 >

地址: H'00008010、地址屏蔽: H'00000006

数据: H'00000000、数据屏蔽: H'00000000

总线周期: C 总线/取指令 (指令执行前) / 读取 (断点条件不包含操作数长度)

地址 H'00000404 的指令执行后或地址 H'00008010 ~ H'00008016 的指令执行前, 产生用户断点。

(例 1-2)

- 寄存器指定

BAR_0=H'00027128、BAMR_0=H'00000000、BBR_0=H'005A、BAR_1=H'00031415、
BAMR_1=H'00000000、BBR_1=H'0054、BDR_1=H'00000000、BDMR_1=H'00000000、
BRCCR=H'00000000

<通道 0 >

地址: H'00027128、地址屏蔽: H'00000000

总线周期: C 总线/取指令 (指令执行前) / 写入/字

<通道 1 >

地址: H'00031415、地址屏蔽: H'00000000

数据: H'00000000、数据屏蔽: H'00000000

总线周期: C 总线/取指令 (指令执行前) / 读取 (断点条件不包含操作数长度)

在通道 0, 取指令不是写入周期, 因此不产生用户断点; 在通道 1, 对偶数地址取指令, 因此同样不产生用户断点。

(例 1-3)

- 寄存器指定

BAR_0=H'00008404、BAMR_0=H'00000FFF、BBR_0=H'0054、BAR_1=H'00008010、
BAMR_1=H'00000006、BBR_1=H'0054、BDR_1=H'00000000、BDMMR_1=H'00000000、
BRCCR=H'00000020

<通道 0>

地址: H'00008404、地址屏蔽: H'00000FFF

总线周期: C 总线/取指令 (指令执行后) /读取 (断点条件不包含操作数长度)

<通道 1>

地址: H'00008010、地址屏蔽: H'00000006

数据: H'00000000、数据屏蔽: H'00000000

总线周期: C 总线/取指令 (指令执行前) /读取 (断点条件不包含操作数长度)

地址 H'00008000 ~ H'00008FFE 的指令执行后或地址 H'00008010 ~ H'00008016 的指令执行前, 产生用户断点。

(2) 指定为 C 总线数据存取周期的断点条件

(例 2-1)

- 寄存器指定

BAR_0=H'00123456、BAMR_0=H'00000000、BBR_0=H'0064、BAR_1=H'000ABCDE、
BAMR_1=H'000000FF、BBR_1=H'106A、BDR_1=H'A512A512、
BDMMR_1=H'00000000、BRCCR=H'00000000

<通道 0>

地址: H'00123456、地址屏蔽: H'00000000

总线周期: C 总线/数据存取/读取 (断点条件不包含操作数长度)

<通道 1>

地址: H'000ABCDE、地址屏蔽: H'000000FF

数据: H'0000A512、数据屏蔽: H'00000000

总线周期: C 总线/数据存取/写入/字

在通道 0, 对地址 H'00123456 的长字读取、地址 H'00123456 的字读取、或地址 H'00123456 的字节读取时, 产生用户断点; 在通道 1, 对地址 H'000ABC00 ~ H'000ABCFF 写入字 H'A512 时, 产生用户断点。

(3) 指定为 I 总线数据存取周期的断点条件

(例 3-1)

- 寄存器指定:

BAR_0=H'00314156、BAMR_0=H'00000000、BBR_0=H'0194、BAR_1=H'00055555、
BAMR_1=H'00000000、BBR_1=H'12A9、BDR_1=H'78787878、BDMR_1=H'0F0F0F0F、
BRCCR=H'00000000

<通道 0 >

地址: H'00314156、地址屏蔽: H'00000000

总线周期: 内部CPU总线/取指令/读取 (断点条件不包含操作数长度)

<通道 1 >

地址: H'00055555、地址屏蔽: H'00000000

数据: H'00000078、数据屏蔽: H'0000000F

总线周期: 内部DMA总线/数据存取/写入/字节

在通道0, 内部CPU总线的取指令设定为无效。在通道1, 内部DMA总线的DMAC对地址H'00055555
写入字节H'7x时, 产生用户断点 (内部CPU总线的存取时, 不产生用户断点)。

7.5 使用时的注意事项

1. 通过内部CPU总线读取或写入UBC的寄存器。因此, 从执行改写UBC的寄存器的指令到实际反映该值期间, 可能不产生预期的断点。为获知更改UBC的寄存器的时序, 必须读取最后写入的寄存器, 之后的指令对新写入的寄存器值有效。
2. UBC不可在相同的通道监控C总线周期、内部CPU总线周期及内部DMA总线周期。
3. 由相同指令产生用户断点中断请求及其他异常源时, 按照“第5章 异常处理”的表5.1所规定的优先顺序判断。产生更高优先级的异常源时, 不接受用户断点中断请求。
4. 在延迟槽产生断点时, 有以下注意事项:
对延迟槽指令设定指令执行前断点时, 到该转移目标执行前, 不接受用户断点中断请求。
5. UBC模块待机时, 不可使用用户断点功能, 并不得读取/写入UBC的寄存器, 否则, 无法保证该值。
6. 中断优先级大于等于15 (含用户断点中断)的中断异常处理程序的地址, 不得设定为断点地址。
7. 不得对SLEEP指令与延迟槽为SLEEP指令的转移指令设定指令执行后断点。
8. 对32位指令设定断点地址时, 必须设定在高16位。在之后的低16位设定断点地址时, 即使断点条件设定为指令执行前, 也作为指令执行后的断点条件处理。
9. 不得设定为DIVU、DIVS指令的下一条指令执行前产生断点。对DIVU、DIVS指令的下一条指令设定指令执行前产生断点时, 在执行DIVU、DIVS指令过程中产生异常、中断, 且即使中断执行DIVU、DIVS指令, 也在下一指令执行前产生断点。

第 8 章 高速缓存

8.1 特点

- 容量
指令高速缓存：8K 字节
操作数高速缓存：8K 字节
- 结构：指令/数据分离、4路集联
- 通路锁定功能（仅限操作数高速缓存）：可锁定通路2、通路3
- 行长度：16 字节
- 入口数：128 个入口/通路
- 写入方式：可选择回写方式与直写方式
- 替换方式：LRU 替换算法

8.1.1 高速缓存的结构

高速缓存是指令 / 数据分离型的 4 路集联方式，由 4 个通路（存储体）组成，每个通路分为地址阵列与数据阵列。

每个通路的地址阵列与数据阵列分别由 128 个入口组成，入口的数据称为行，每行 16 字节（4 字节 × 4）。每个通路的数据容量为 2K 字节（16 字节 × 128 个入口），高速缓存整体（4 路）为 8K 字节。

操作数高速缓存的结构如图 8.1 所示。除没有 U 位之外，指令高速缓存的结构与操作数高速缓存的结构相同。

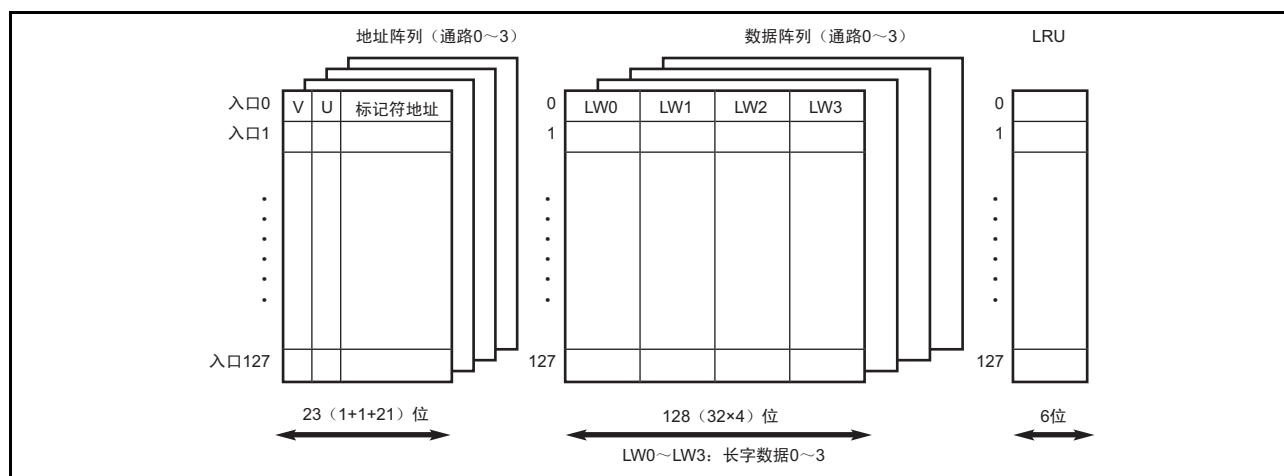


图 8.1 操作数高速缓存的结构

(1) 地址阵列

V 位表示入口数据是否有效，为 1 时表示有效；为 0 则表示无效。

U 位（仅限操作数高速缓存）表示在回写模式，该入口有写入。U 位为 1 时表示有写入；为 0 时表示无写入。

标记符地址保存用于外部存储器存取的地址，由高速缓存搜索时用于比较的 21 个位（地址 31 ~ 11）组成。在本 LSI，高速缓存有效空间的地址为 H'00000000 ~ H'1FFFFFFF（参阅“第 9 章 总线状态控制器（BSC）”），标记符地址的高 3 位为 0。

V、U 位在上电复位时初始化为 0，手动复位或软件待机模式时不被初始化。在上电复位、手动复位或软件待机模式，标记符地址均不被初始化。

(2) 数据阵列

数据阵列保存 16 字节的指令或数据。高速缓存入口的登录以行为单位（16 字节单位）进行。

在上电复位、手动复位或软件待机模式，数据阵列均不被初始化。

(3) LRU

通过 4 路集联方式，最多可登录 4 条入口地址相同的指令或数据到高速缓存。登录入口时，LRU 位表示登录到 4 路中的哪一个通路。LRU 位由 6 位组成，通过硬件控制。使用 LRU（Least Recently Used）算法作为选择通路的算法，选择最近被存取的通路。

通过 LRU 位（6 位）指定高速缓存未命中时被替换的通路。不使用高速缓存锁定功能（仅限操作数高速缓存）时的 LRU 位与被替换通路之间的关系如表 8.1 所示（使用高速缓存锁定功能时，参阅“8.2.2 高速缓存控制寄存器 2（CCR2）”）。由软件指定除表 8.1 所示之外的 LRU 位时，高速缓存无法正常运行；由软件更改 LRU 位时，必须设定表 8.1 所示的模式。

LRU 位在上电复位时初始化为 B'000000，在手动复位或软件待机模式下均不被初始化。

表 8.1 LRU 位与被替换的通路（不使用高速缓存锁定功能时）

LRU (bit5 ~ 0)	被替换的通路
000000、000100、010100、100000、110000、110100	3
000001、000011、001011、100001、101001、101011	2
000110、000111、001111、010110、011110、011111	1
111000、111001、111011、111100、111110、111111	0

8.2 寄存器说明

高速缓存有以下寄存器。

表 8.2 寄存器结构

寄存器名称	简称	R/W	初始值	地址	存取长度
高速缓存控制寄存器 1	CCR1	R/W	H'00000000	H'FFFC1000	32
高速缓存控制寄存器 2	CCR2	R/W	H'00000000	H'FFFC1004	32

8.2.1 高速缓存控制寄存器 1 (CCR1)

由 ICE 位指定允许或禁止指令高速缓存。ICF 位控制指令高速缓存所有入口的无效化。由 OCE 位指定允许或禁止操作数高速缓存。OCF 位控制操作数高速缓存所有入口的无效化。WT 位转换操作数高速缓存的直写模式与回写模式。

更改 CCR1 内容的程序配置于高速缓存无效空间，读取 CCR1 的内容后，必须存取高速缓存有效空间。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	ICF	-	-	ICE	-	-	-	-	OCF	-	WT	OCE
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R	R	R/W	R	R	R	R	R/W	R	R/W	R/W

位	位名称	初始值	R/W	说 明
31 ~ 12	—	均为 0	R	保留位 读取值、写入值总是为 0。
11	ICF	0	R/W	指令高速缓存清除 如果写入 1，则将指令高速缓存所有入口的 V、LRU 位清 0（清除）。读取值总是为 0。清除时，不执行向外部寄存器的回写。
10、9	—	均为 0	R	保留位 读取值、写入值总是为 0。
8	ICE	0	R/W	指令高速缓存有效 指定指令高速缓存功能的允许 / 禁止。 0: 禁止指令高速缓存 1: 允许指令高速缓存
7 ~ 4	—	均为 0	R	保留位 读取值、写入值总是为 0。
3	OCF	0	R/W	操作数高速缓存清除 如果写入 1，则将操作数高速缓存所有入口的 V、U、LRU 位清 0（清除）。读取值总是为 0。清除时，不执行向外部寄存器的回写。
2	—	0	R	保留位 读取值、写入值总是为 0。
1	WT	0	R/W	直写 回写 / 直写转换 0: 回写模式 1: 直写模式
0	OCE	0	R/W	操作数高速缓存有效 指定操作数高速缓存功能的允许 / 禁止。 0: 禁止操作数高速缓存 1: 允许操作数高速缓存

8.2.2 高速缓存控制寄存器 2（CCR2）

CCR2 为控制操作数高速缓存的高速缓存锁定功能的寄存器，高速缓存锁定功能仅在高速缓存锁定模式下有效。高速缓存锁定模式是指 CCR2 的锁定允许位（LE 位）=1 的状态。在非高速缓存锁定模式下，高速缓存锁定功能无效。

在高速缓存锁定模式执行预取指令（PREF@Rn）且高速缓存未命中时，根据 CCR2 的 bit9、8（W3LOAD、W3LOCK）及 bit1、0（W2LOAD、W2LOCK）的设定，将 Rn 指示的 1 行数据输入高速缓存。执行预取指令时，各位的设定与被替换通路之间的关系如表 8.3 所示。另一方面，执行预取指令且高速缓存命中时，不输入新数据，仍保持已经有效的入口。例如，Rn 指示的 1 行数据已存在于通路 0，设定高速缓存锁定模式、W3LOAD=1 且 W3LOCK=1，并执行预取指令时，为高速缓存命中，不向通路 3 输入数据。

在高速缓存锁定模式的预取指令以外存取高速缓存时，限制被 W3LOCK、W2LOCK 位替换的通路。CCR2 各位的设定与被替换通路之间的关系如表 8.4 所示。

更改 CCR2 内容的程序配置于高速缓存无效空间，读取 CCR2 内容后，必须存取高速缓存有效空间。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LE
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	W3 LOAD*	W3 LOCK	-	-	-	-	-	-	W2 LOAD*	W2 LOCK
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

【注】* W3LOAD与W2LOAD不可同时置1。

位	位名称	初始值	R/W	说 明
31 ~ 17	—	均为 0	R	保留位 读取值、写入值总是为 0。
16	LE	0	R/W	锁定允许 控制高速缓存锁定模式。 0: 非高速缓存锁定模式 1: 高速缓存锁定模式
15 ~ 10	—	均为 0	R	保留位 读取值、写入值总是为 0。
9 8	W3LOAD* W3LOCK	0 0	R/W R/W	通路 3 加载 通路 3 锁定 W3LOCK=1、W3LOAD=1 且高速缓存锁定模式时，通过预取指令高速缓存未命中的数据总被读取到通路 3，在其他条件下，高速缓存未命中的数据被读取到 LRU 所示的通路。
7 ~ 2	—	均为 0	R	保留位 读取值、写入值总是为 0。
1 0	W2LOAD* W2LOCK	0 0	R/W R/W	通路 2 加载 通路 2 锁定 W2LOCK=1、W2LOAD=1 且高速缓存锁定模式时，通过预取指令高速缓存未命中的数据总被读取到通路 2，在其他条件下，高速缓存未命中的数据被读取到 LRU 所示的通路。

【注】* W3LOAD 与 W2LOAD 不可同时置 1。

表 8.3 PREF 指令高速缓存未命中时被替换的通路

LE	W3LOAD*	W3LOCK	W2LOAD*	W2LOCK	被替换的通路
0	x	x	x	x	根据 LRU (表 8.1)
1	x	0	x	0	根据 LRU (表 8.1)
1	x	0	0	1	根据 LRU (表 8.5)
1	0	1	x	0	根据 LRU (表 8.6)
1	0	1	0	1	根据 LRU (表 8.7)
1	0	x	1	1	通路 2
1	1	1	0	x	通路 3

【符号说明】 x: Don't care

【注】 * W3LOAD 与 W2LOAD 不可同时置 1。

表 8.4 除 PREF 之外的指令高速缓存未命中时被替换的通路

LE	W3LOAD*	W3LOCK	W2LOAD*	W2LOCK	被替换的通路
0	x	x	x	x	根据 LRU (表 8.1)
1	x	0	x	0	根据 LRU (表 8.1)
1	x	0	x	1	根据 LRU (表 8.5)
1	x	1	x	0	根据 LRU (表 8.6)
1	x	1	x	1	根据 LRU (表 8.7)

【符号说明】 x: Don't care

【注】 * W3LOAD 与 W2LOAD 不可同时置 1。

表 8.5 LRU 位与被替换的通路 (W2LOCK=1 且 W3LOCK=0 时)

LRU (bit5 ~ 0)	被替换的通路
000000, 000001, 000100, 010100, 100000, 100001, 110000, 110100	3
000011, 000110, 000111, 001011, 001111, 010110, 011110, 011111	1
101001, 101011, 111000, 111001, 111011, 111100, 111110, 111111	0

表 8.6 LRU 位与被替换的通路 (W2LOCK=0 且 W3LOCK=1 时)

LRU (bit5 ~ 0)	被替换的通路
000000, 000001, 000011, 001011, 100000, 100001, 101001, 101011	2
000100, 000110, 000111, 001111, 010100, 010110, 011110, 011111	1
110000, 110100, 111000, 111001, 111011, 111100, 111110, 111111	0

表 8.7 LRU 位与被替换的通路（W2LOCK=1 且 W3LOCK=1 时）

LRU（bit5～0）	被替换的通路
000000, 000001, 000011, 000100, 000110, 000111, 001011, 001111, 010100, 010110, 011110, 011111	1
100000, 100001, 101001, 101011, 110000, 110100, 111000, 111001, 111011, 111100, 111110, 111111	0

8.3 运行说明

说明操作数高速缓存。除地址阵列中没有 U 位、预取操作、写入操作及回写缓冲器之外，指令高速缓存与操作数高速缓存相同。

8.3.1 高速缓存的搜索

如果操作数高速缓存为允许（CCR1 寄存器的 OCE 位 =1），并存取高速缓存有效空间的数据，则搜索高速缓存、并调查目标数据是否存在于高速缓存。高速缓存搜索方法的概念图如图 8.2 所示。

通过存储器存取地址的 bit10～4 选择入口，并读取该入口的标记符地址。此时，标记符地址的高 3 位总是为 0。比较存储器存取地址的 bit31～11 与读取的标记符地址。4 个通路均执行地址比较。比较结果匹配且被比较的入口有效（V=1）时，为高速缓存命中；除此之外的情况为高速缓存未命中。通路 1 命中时如图 8.2 所示。

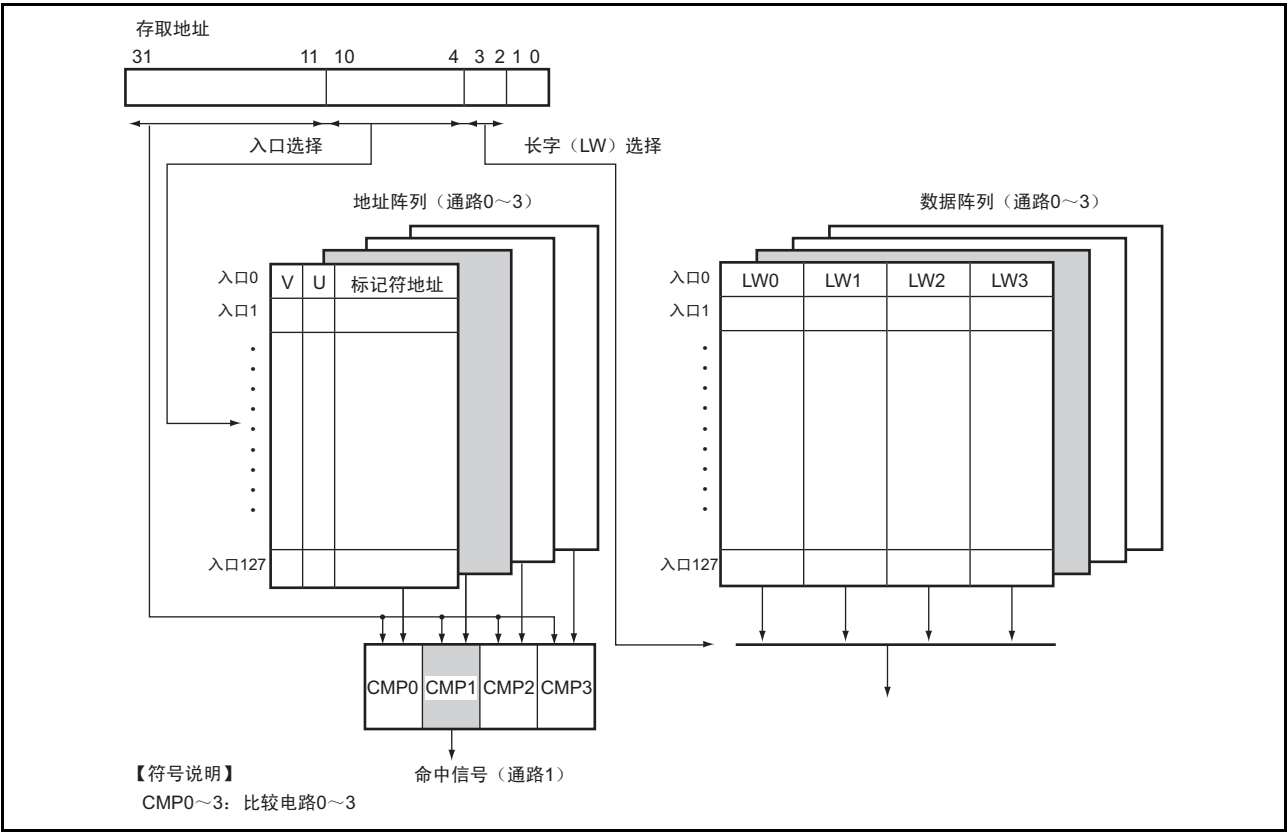


图 8.2 高速缓存的搜索方法

8.3.2 读取操作

(1) 读取命中

将数据从高速缓存传送至 CPU。更新 LRU 使命中的通路为最新。

(2) 读取未命中

启动外部总线周期，更新入口。替换的通路如表 8.4 所示。入口的更新单位为 16 字节。从外部存储器将目标数据登录至高速缓存的同时，该数据被传送至 CPU。登录至高速缓存时，V 位置 1，并更新 LRU 使被替换的通路为最新。在操作数高速缓存，U 位被置 0，并在回写模式，因入口更新而被替换的入口的 U 位为 1 时，将该入口传送至回写缓冲器后，开始高速缓存更新周期。高速缓存更新周期结束后，将传送至回写缓冲器的入口回写至存储器。回写的单位为 16 字节。以环绕方式更新高速缓存及回写存储器。例如，产生读取未命中的地址低 4 位为 H'4 时，地址的低 4 位按照 H'4→H'8→H'C→H'0 的顺序，更新高速缓存及回写存储器。

8.3.3 预取操作（仅限操作数高速缓存）

(1) 预取命中

更新 LRU 使命中的通路为最新，不更改其他高速缓存的内容，不向 CPU 传送数据。

(2) 预取未命中

不向 CPU 传送数据，替换的通路如表 8.3 所示。其他运行与读取未命中时相同。

8.3.4 写入操作（仅限操作数高速缓存）

(1) 写入命中

在回写模式，将数据写入高速缓存，不发行对外部存储器的写入周期。写入入口的 U 位置 1，并更新 LRU 使命中的通路为最新。

在直写模式，将数据写入高速缓存，并发行对外部存储器的写入周期。不更新写入入口的 U 位，但更新 LRU 使命中的通路为最新。

(2) 写入未命中

在回写模式，写入未命中时，启动外部总线周期，并更新入口。替换的通路如表 8.4 所示。由于入口更新而被替换的入口的 U 位为 1 时，将该入口传送至回写缓冲器后，开始高速缓存更新周期。将数据写入高速缓存后，U 位置 1，V 位也置 1。更新 LRU 使被替换的通路为最新。高速缓存更新周期结束后，将传送至回写缓冲器的入口回写至存储器。回写的单位为 16 字节。以环绕方式更新高速缓存及回写存储器。例如，产生写入未命中的地址低 4 位为 H'4 时，地址的低 4 位按照 H'4→H'8→H'C→H'0 的顺序，更新高速缓存及回写存储器。

在直写模式，写入未命中时，不对高速缓存写入，仅对外部存储器写入。

8.3.5 回写缓冲器（仅限操作数高速缓存）

在回写模式被替换入口的 U 位为 1 时，需对外部存储器回写。为了提高性能，首先将被替换的入口传送至回写缓冲器，并优先回写对高速缓存新入口的输入。对高速缓存新入口的输入结束后，回写缓冲器执行对外部存储器的回写，在回写过程中，高速缓存可存取。

回写缓冲器可保存高速缓存 1 行的数据（16 字节）及其地址。回写缓冲器的结构如图 8.3 所示。

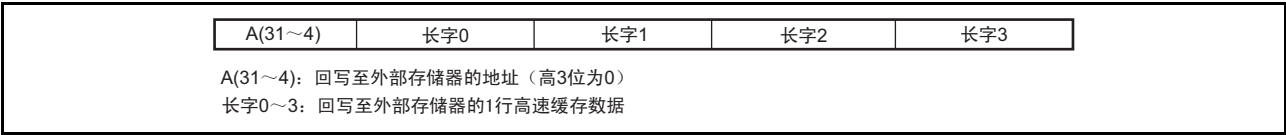


图 8.3 回写缓冲器的结构

上述 8.3.2 ~ 8.3.5 的运行汇总如表 8.8 所示。

表 8.8 高速缓存运行汇总

高速缓存的种类	CPU 周期	命中 / 未命中	回写模式 / 直写模式	U 位	对外部存储器的存取 (通过内部总线)	高速缓存的内容
指令高速缓存	取指令	命中	—	—	不产生。	不被更新。
		未命中	—	—	产生高速缓存更新周期。	用左侧更新周期的内容更新。
操作数高速缓存	预取 / 读取	命中	任意模式	x	不产生。	不被更新。
		未命中	直写模式	—	产生高速缓存更新周期。	用左侧更新周期的内容更新。
			回写模式	0	产生高速缓存更新周期。	用左侧更新周期的内容更新。
				1	产生高速缓存更新周期。 之后产生回写缓冲器的回写周期。	用左侧更新周期的内容更新。
	写入	命中	直写模式	—	产生 CPU 发行的写入周期。	用 CPU 发行的写入周期的内容更新。
			回写模式	x	不产生。	用 CPU 发行的写入周期的内容更新。
		未命中	直写模式	—	产生 CPU 发行的写入周期。	不被更新。*
			回写模式	0	产生高速缓存更新周期。	用左侧更新周期的内容的内容更新后，再用 CPU 发行的写入周期的内容更新。
				1	产生高速缓存更新周期。 之后产生回写缓冲器的回写周期。	用左侧更新周期的内容的内容更新后，再用 CPU 发行的写入周期的内容更新。

【符号说明】 x: Don't care

【注】 高速缓存更新周期：16 字节的读取存取
 回写缓冲器的回写周期：16 字节的写入存取

* LRU 也不被更新。在除此之外的所有情况下，LRU 均被更新。

8.3.6 高速缓存与外部存储器的比配

必须以软件保证高速缓存与外部存储器的比配。

将本 LSI 与其他器件的共用存储器配置于高速缓存有效空间时，必须根据需要执行存储器分配高速缓存，执行无效化与回写。本 LSI 的 CPU 与 DMAC 的共用存储器，也必须执行相同的操作。

8.4 存储器分配高速缓存的结构

用软件管理高速缓存，因此通过 MOV 指令，可读取、写入高速缓存的内容。指令高速缓存的地址阵列分配在 H'F0000000 ~ H'F07FFFFFFF，数据阵列分配在 H'F1000000 ~ H'F17FFFFFFF；操作数高速缓存的地址阵列分配在 H'F0800000 ~ H'F0FFFFFFF，数据阵列分配在 H'F1800000 ~ H'F1FFFFFFF。地址阵列与数据阵列的存取长度均固定为长字，且不可执行取指令。

8.4.1 地址阵列

存取地址阵列时，必须指定 32 位地址（读取 / 写入时）与 32 位数据（写入时）。

在地址指定选择入口的入口地址、选择通路的 W 位及指定有无联想操作的 A 位。W 位的 B'00 表示通路 0，B'01 表示通路 1，B'10 表示通路 2，B'11 表示通路 3。存取长度固定为长字，因此必须对地址的 bit1 ~ 0 指定 B'00。

在数据指定标记符地址、LRU 位、U 位（仅限操作数高速缓存）及 V 位。必须指定标记符地址的高 3 位（bit31 ~ 29）总是为 0。

有关地址及数据格式，详情参照图 8.4。

可对地址阵列执行以下 3 种运行：

(1) 读取地址阵列

从地址指定的入口地址及与通路对应的入口，读取标记符地址、LRU 位、U 位（仅限操作数高速缓存）及 V 位。读取时，无论地址指定的联想位（A 位）是 1 还是 0，均不执行联想运行。

(2) 写入地址阵列（无联想）

将地址的联想位（A 位）置 0 后写入时，对地址指定的入口地址及与通路相对应的入口，写入数据指定的标记符地址、LRU 位、U 位（仅限操作数高速缓存）及 V 位。对 U 位 =1、V 位 =1 的高速缓存行执行对操作数高速缓存地址阵列的写入时，在执行该高速缓存行回写后，写入数据指定的标记符地址、LRU 位、U 位及 V 位。但对 V 位写入 0 时，必须对该入口的 U 位也写入 0。向存储器回写时，地址低 4 位按照 H'0→H'4→H'8→H'C 的顺序执行。

(3) 写入地址阵列（有联想）

将地址的联想位（A 位）置 1 后写入时，执行地址指定的所有入口（4 个通路）与数据指定的标记符地址间的比配判断。对匹配判断的结果命中的通路，将数据指定的 U 位（仅限操作数高速缓存）与 V 位写入入口，但标记符地址与 LRU 位不变。任何通路均无命中时不执行写入，为空操作。此运行用于高速缓存特定入口的无效化。

操作数高速缓存时，如果命中入口的 U 位为 1，则产生回写。对 V 位写入 0 时，必须对该入口的 U 位也写入 0。向存储器回写时，地址低 4 位按照 H'0→H'4→H'8→H'C 的顺序执行。

8.4.2 数据阵列

存取数据阵列时，必须指定 32 位地址（读取 / 写入时）与 32 位数据（写入时）。在地址指定选择存取入口的信息；在数据指定写入数据阵列的长字数据。

在地址指定选择入口的入口地址、表示 1 行（16 字节）中长字位置的 L 位及指定通路的 W 位。L 位的 B'00 表示长字 0，B'01 表示长字 1，B'10 表示长字 2，B'11 表示长字 3；W 位的 B'00 表示通路 0，B'01 表示通路 1，B'10 表示通路 2，B'11 表示通路 3。存取长度固定为长字，因此必须对存取地址的 bit1 ~ 0 指定 B'00。

有关地址及数据格式，详情参照图 8.4。

可对数据阵列执行以下 2 种运行，此运行不会更改地址阵列的信息。

(1) 读取数据阵列

从地址指定的入口地址及与通路对应的入口，读取地址的 L 位指定的数据。

(2) 写入数据阵列

在地址指定的入口地址及与通路对应的入口中，向地址的 L 位指定的位置写入数据指定的长字数据。

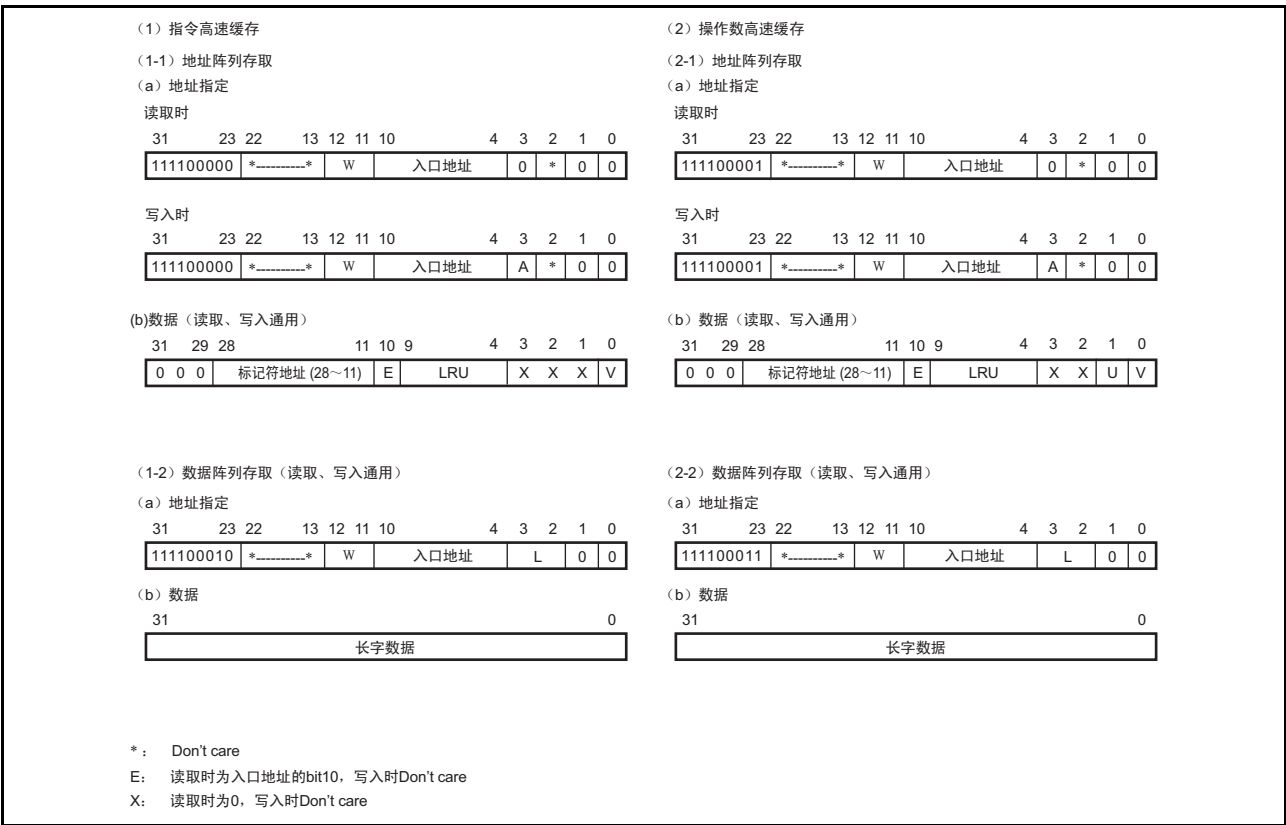


图 8.4 存储器分配高速缓存存取的地址及数据指定方法

8.4.3 使用例

(1) 特定入口的无效化

存储器分配高速缓存存取时，通过对该入口的 V 位写入 0，可实现特定高速缓存入口的无效化。A 位置 1，比较写入数据指定的标记符地址与入口地址选择的高速缓存的标记符地址，匹配时，写入写入数据指定的 V 位与 U 位；不匹配时空操作。如果将地址阵列的某入口的 V 位置 0，则该入口的 U 位为 1 时，回写该入口。

对 R0 指定写入数据、对 R1 指定地址时的例子如下所示：

```
； R0=H'0110 0010； 标记符地址（28 ～ 11）=B'0 0001 0001 0000 0000 0、U= 0、V= 0
； R1=H'F080 0088； 操作数高速缓存地址阵列存取、入口 = B'000 1000、A=1
；
MOV.L R0, @R1
```

(2) 读取特定入口的数据部

通过存储器分配高速缓存存取，可读取特定入口的数据部。图 8.4 的数据阵列的数据部所示的长字被读取到寄存器。

对 R0 指定地址、向 R1 读取的例子如下所示：

```
； R0=H'F100 004C； 指令高速缓存数据阵列存取、入口 =B'000 0100、通路 =0、
长字地址 =3
；
MOV.L @ R0, R1
```

8.4.4 注意事项

1. 存取存储器分配高速缓存的程序必须配置于高速缓存无效空间。
2. 禁止为了多个通路同时命中而改写地址阵列的内容；如果改写，则不保证其运行。
3. 寄存器及存储器分配高速缓存只可通过 CPU 存取，而不可通过 DMAC 存取。

第 9 章 总线状态控制器 (BSC)

外部总线控制器 (BSC) 对连接外部地址空间的各种存储器及外部器件输出控制信号, 可直接连接 SRAM、SDRAM 等各种存储器及外部器件。

9.1 特点

1. 外部地址空间

- CS0~CS7 各空间分别最大支持 64M 字节
- 可指定每个空间的普通空间接口、带字节选择的 SRAM 接口、突发 ROM (时钟同步或异步)、MPX-
- I/O、突发 MPX-I/O、SDRAM 存储器种类及 PCMCIA 接口
- 可选择每个空间的数据总线宽度 (8 位、16 位或 32 位)
- 可控制每个空间的插入等待状态
- 可控制每个读取存取、写入存取的插入等待状态
- 连续存取为读取—写入 (相同空间或不同空间)、读取—读取 (相同空间或不同空间) 及起始周期为写入等 5 种情况时, 可单独设定空闲周期

2. 普通空间接口

- 支持可直接连接 SRAM 的接口

3. 突发 ROM (时钟异步) 接口

- 可高速存取具有页模式功能的 ROM

4. MPX-I/O 接口

- 可直接连接需要地址/数据多路复用的外围 LSI

5. SDRAM 接口

- 最多可在 2 个区域设定 SDRAM
- 支持行地址/列地址的多路复用输出
- 可通过单次读取/单次写入实现高效存取
- 可通过存储体激活模式高速存取
- 支持自动刷新和自刷新
- 支持低频模式、掉电模式
- 支持发行 MRS 命令、EMRS 命令

6. PCMCIA 接口

- 支持 JEIDA 标准 Ver4.2 (PCMCIA2.1 Rev2.1) 规定的 IC 存储卡及 I/O 卡接口
- 可通过程序控制等待状态的插入

7. 带字节选择的 SRAM 接口

- 支持可直接连接带字节选择的 SRAM 接口

8. 突发 MPX-I/O 接口

- 可直接连接需要地址/数据多路复用的外围 LSI
- 支持突发传送

9. 突发 ROM (时钟同步) 接口

- 可直接连接时钟同步类型的突发 ROM

10. 总线仲裁

- 与其他的 CPU 共用所有资源, 可接受来自外部器件的总线权请求及输出总线使用允许

11. 刷新功能

- 支持自动刷新和自刷新
- 通过选择刷新计数器或时钟, 可设定刷新闻隔。
- 通过设定刷新次数 (1、2、4、6或8), 可集中刷新

12. 用作刷新计数器的间隔定时器

- 通过比较匹配定时器可产生中断请求

BSC 框图如图 9.1 所示。

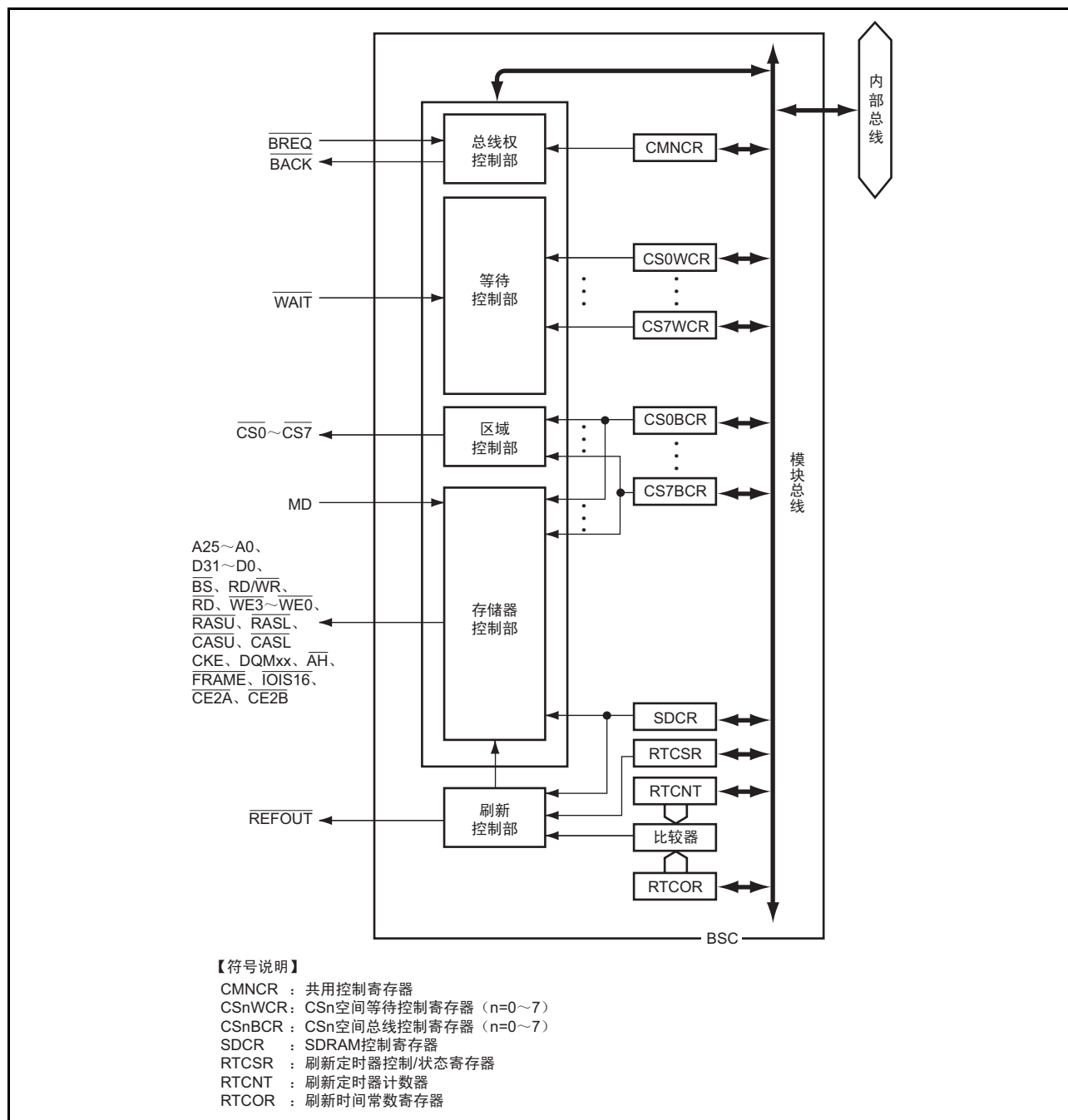


图 9.1 BSC 框图

9.2 输入 / 输出引脚

BSC 的引脚结构，如表 9.1 所示。

表 9.1 引脚结构

引脚名称	输入 / 输出	功 能
A25 ~ A0	输出	地址总线
D31 ~ D0	输入 / 输出	数据总线
$\overline{BS}$	输出	表示总线周期开始的信号
$\overline{CS0} \sim \overline{CS4}$ 、 $\overline{CS7}$	输出	片选
$\overline{CS5}/\overline{CE1A}$ 、 $\overline{CS6}/\overline{CE1B}$	输出	片选 使用 PCMCIA 时，对应 PCMCIA 卡选信号 D7 ~ D0
$\overline{CE2A}$ 、 $\overline{CE2B}$	输出	对应 PCMCIA 卡选信号 D15 ~ D8
$\overline{RD}/\overline{WR}$	输出	读取 / 写入信号 连接 SDRAM 或带字节选择的 SRAM 时，将此引脚连接至 $\overline{WE}$ 引脚
$\overline{RD}$	输出	读取脉冲信号（读取数据输出允许信号） 使用 PCMCIA 时，为表示存储器读取周期的选通信号
$\overline{WE3}/\overline{DQM0U}/\overline{ICI0WR}/\overline{AH}$	输出	指示对应 D31 ~ D24 的字节写入 连接带字节选择的 SRAM 时，将此引脚连接至字节选择引脚 连接 SDRAM 时，为对应 D31 ~ D24 的选择信号 使用 PCMCIA 时，为表示 I/O 写入的选通信号 使用 MPX-I/O 时，为用于保持地址的信号
$\overline{WE2}/\overline{DQM1U}/\overline{ICI1RD}$	输出	指示对应 D23 ~ D16 的字节写入 连接带字节选择的 SRAM 时，将此引脚连接至字节选择引脚 连接 SDRAM 时，为对应 D23 ~ D16 的选择信号 使用 PCMCIA 时，为表示 I/O 读取的选通信号
$\overline{WE1}/\overline{DQM2U}/\overline{WE}$	输出	指示对应 D15 ~ D8 的字节写入 连接带字节选择的 SRAM 时，将此引脚连接至字节选择引脚 连接 SDRAM 时，为对应 D15 ~ D8 的选择信号 使用 PCMCIA 时，为表示存储器写入周期的选通信号
$\overline{WE0}/\overline{DQM3U}$	输出	指示对应 D7 ~ D0 的字节写入 连接带字节选择的 SRAM 时，将此引脚连接至字节选择引脚 连接 SDRAM 时，为对应 D7 ~ D0 的选择信号
$\overline{RASU}$ 、 $\overline{RASL}$	输出	连接 SDRAM 时，将此引脚连接至 $\overline{RAS}$ 引脚
$\overline{CASU}$ 、 $\overline{CASL}$	输出	连接 SDRAM 时，将此引脚连接至 $\overline{CAS}$ 引脚
$\overline{CKE}$	输出	连接 SDRAM 时，将此引脚连接至 $\overline{CKE}$ 引脚
$\overline{FRAME}$	输出	使用突发 MPX-I/O 接口时，为 FRAME 信号
$\overline{WAIT}$	输入	输入外部等待
$\overline{BREQ}$	输入	输入总线权请求
$\overline{BACK}$	输出	输出总线使用允许
$\overline{REFOUT}$	输出	输出总线释放时刷新执行请求
$\overline{IOIS16}$	输入	表示 PCMCIA16 位 I/O 的信号 仅在小端法时有效，大端法时，必须设置为低电平
$\overline{MD}$	输入	选择区域 0 的总线宽度，选择区域 1 ~ 7 的总线宽度初始值

9.3 区域概要

9.3.1 地址映射

本 LSI 具有 32 位地址空间的体系结构，通过高位分为高速缓存有效空间、高速缓存无效空间、内部空间（内部 RAM、内部外围模块、保留）。

CS0 ~ CS7 的外部地址空间，在内部地址 A29=0 时，为高速缓存有效；A29=1 时，为高速缓存无效。

可指定各部分空间连接的存储器种类及数据总线宽度。外部地址空间的地址映射如表 9.2 所示。

表 9.2 地址映射

内部地址	空间	存储器种类	高速缓存
H'00000000 ~ H'03FFFFFF	CS0	普通空间、突发 ROM（异步、同步）	有效
H'04000000 ~ H'07FFFFFF	CS1	普通空间、带字节选择的 SRAM	
H'08000000 ~ H'0BFFFFFF	CS2	普通空间、带字节选择的 SRAM、SDRAM	
H'0C000000 ~ H'0FFFFFFF	CS3	普通空间、带字节选择的 SRAM、SDRAM	
H'10000000 ~ H'13FFFFFF	CS4	普通空间、带字节选择的 SRAM、突发 ROM（异步）	
H'14000000 ~ H'17FFFFFF	CS5	普通空间、带字节选择的 SRAM、MPX-I/O、PCMCIA	
H'18000000 ~ H'1BFFFFFF	CS6	普通空间、带字节选择的 SRAM、突发 MPX-I/O、PCMCIA	
H'1C000000 ~ H'1FFFFFFF	CS7	普通空间、带字节选择的 SRAM	
H'20000000 ~ H'23FFFFFF	CS0	普通空间、突发 ROM（异步、同步）	无效
H'24000000 ~ H'27FFFFFF	CS1	普通空间、带字节选择的 SRAM	
H'28000000 ~ H'2BFFFFFF	CS2	普通空间、带字节选择的 SRAM、SDRAM	
H'2C000000 ~ H'2FFFFFFF	CS3	普通空间、带字节选择的 SRAM、SDRAM	
H'30000000 ~ H'33FFFFFF	CS4	普通空间、带字节选择的 SRAM、突发 ROM（异步）	
H'34000000 ~ H'37FFFFFF	CS5	普通空间、带字节选择的 SRAM、MPX-I/O、PCMCIA	
H'38000000 ~ H'3BFFFFFF	CS6	普通空间、带字节选择的 SRAM、突发 MPX-I/O、PCMCIA	
H'3C000000 ~ H'3FFFFFFF	CS7	普通空间、带字节选择的 SRAM	
H'80000000 ~ H'FFFBFFFF	其他	内部 RAM、保留区域 *	—
H'FFFC0000 ~ H'FFFFFFF	其他	内部外围模块、保留区域 *	—

【注】 * 内部 RAM 空间的存取在“第 27 章 内部 RAM”所示的地址执行；内部外围模块空间的存取在“第 30 章 寄存器一览表”所示的地址执行。不得在未记载的地址存取，否则无法保证其运行。

9.3.2 各区域的数据总线宽度与引脚功能设定

本 LSI 在上电复位时，使用外部引脚，可将区域 0 的数据总线宽度与区域 1 ~ 7 初始状态的数据总线宽度设定为 16 位或 32 位。区域 0 的总线宽度在上电复位后不可更改；区域 1 ~ 7 初始状态的数据总线宽度与区域 0 相同，但可通过设定寄存器，在程序中更改为 8 位、16 位或 32 位。必须注意，根据设定的存储器类型，可能会限制数据总线宽度。

上电复位后，由保存在外部存储器（分配于区域 0）的程序启动。区域 0 的外部存储器假设为 ROM，因此用作地址总线、数据总线、 $\overline{\text{CS0}}$ 、 $\overline{\text{RD}}$ 等基本的引脚功能。本章记载的存取波形的例子中还包含 $\overline{\text{BS}}$ 、 $\overline{\text{RD/WR}}$ 、 $\overline{\text{WEn}}$ 等引脚，均为通过引脚功能控制器设定引脚功能时的例子。通过程序完成引脚的设定前，只能读取存取区域 0。由 32 位数据总线宽度启动后，将区域 0 以外的空间总线宽度更改为 16 位时，需设定 A1 引脚；更改为 8 位时，需设定 A1、A0 引脚。必须注意：使用区域 7 空间时，将 $\overline{\text{CS7}}$ 与 A0 的功能分配于相同引脚，因此，不可设定 8 位总线宽度。

有关引脚设定，详情参阅“第 25 章 引脚功能控制器 (PFC)”。

表 9.3 外部引脚 (MD) 设定与数据总线宽度的关系

MD	数据总线宽度
1	32 位
0	16 位

9.4 寄存器说明

BSC 有以下寄存器。

完成连接存储器接口的设定前，不得存取区域 0 以外的空间。

表 9.4 寄存器结构

寄存器名称	简称	R/W	初始值	地址	存取长度
共用控制寄存器	CMNCR	R/W	H'00001010	H'FFFC0000	32
CS0 空间总线控制寄存器	CS0BCR	R/W	H'36DB0600*	H'FFFC0004	32
CS1 空间总线控制寄存器	CS1BCR	R/W	H'36DB0600*	H'FFFC0008	32
CS2 空间总线控制寄存器	CS2BCR	R/W	H'36DB0600*	H'FFFC000C	32
CS3 空间总线控制寄存器	CS3BCR	R/W	H'36DB0600*	H'FFFC0010	32
CS4 空间总线控制寄存器	CS4BCR	R/W	H'36DB0600*	H'FFFC0014	32
CS5 空间总线控制寄存器	CS5BCR	R/W	H'36DB0600*	H'FFFC0018	32
CS6 空间总线控制寄存器	CS6BCR	R/W	H'36DB0600*	H'FFFC001C	32
CS7 空间总线控制寄存器	CS7BCR	R/W	H'36DB0600*	H'FFFC0020	32
CS0 空间等待控制寄存器	CS0WCR	R/W	H'00000500	H'FFFC0028	32
CS1 空间等待控制寄存器	CS1WCR	R/W	H'00000500	H'FFFC002C	32
CS2 空间等待控制寄存器	CS2WCR	R/W	H'00000500	H'FFFC0030	32
CS3 空间等待控制寄存器	CS3WCR	R/W	H'00000500	H'FFFC0034	32
CS4 空间等待控制寄存器	CS4WCR	R/W	H'00000500	H'FFFC0038	32
CS5 空间等待控制寄存器	CS5WCR	R/W	H'00000500	H'FFFC003C	32
CS6 空间等待控制寄存器	CS6WCR	R/W	H'00000500	H'FFFC0040	32
CS7 空间等待控制寄存器	CS7WCR	R/W	H'00000500	H'FFFC0044	32
SDRAM 控制寄存器	SDCR	R/W	H'00000000	H'FFFC004C	32
刷新定时器控制 / 状态寄存器	RTCSR	R/W	H'00000000	H'FFFC0050	32
刷新定时器计数器	RTCNT	R/W	H'00000000	H'FFFC0054	32
刷新时间常数寄存器	RTCOR	R/W	H'00000000	H'FFFC0058	32

【注】 * 通过外部引脚 (MD) 设定为 32 位总线宽度启动时的初始值。设定为 16 位总线宽度时，初始值为 H'36DB0400。

9.4.1 共用控制寄存器 (CMNCR)

CMNCR 为在各区域共同执行控制的 32 位寄存器。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	BLOCK	DPRTY[1:0]	DMAIW[2:0]			DMAIWA	-	-	-	HIZ MEM	HIZ CNT	
初始值:	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W

位	位名称	初始值	R/W	说 明
31 ~ 13	—	均为 0	R	保留位 读取值、写入值总是为 0。
12	—	1	R	保留位 读取值、写入值总是为 1。
11	BLOCK	0	R/W	总线锁定位 指定是否接受 $\overline{\text{BREQ}}$ 。 0: 接受 $\overline{\text{BREQ}}$ 。 1: 不接受 $\overline{\text{BREQ}}$ 。
10、9	DPRTY[1:0]	00	R/W	DMA 突发传送优先顺序 本位指定 DMA 突发传送过程中刷新请求 / 总线权使用请求的优先顺序。 00: 在 DMA 突发传送过程中接受刷新请求与总线权使用请求。 01: 在 DMA 突发传送过程中接受刷新请求, 但不接受总线权使用请求。 10: 在 DMA 突发传送过程中不接受刷新请求与总线权使用请求。 11: 保留 (禁止设定)

位	位名称	初始值	R/W	说 明
8 ~ 6	DMAIW[2:0]	000	R/W	指定 DMA 单地址传送时的存取周期之间的等待 本位指定 DMA 单地址传送时，从带 DACK 的外部器件输出数据后插入的空闲周期数。空闲周期的插入方法取决于后述 DMAIWA 位的指定。 000: 无空闲周期 001: 插入 1 个空闲周期 010: 插入 2 个空闲周期 011: 插入 4 个空闲周期 100: 插入 6 个空闲周期 101: 插入 8 个空闲周期 110: 插入 10 个空闲周期 111: 插入 12 个空闲周期
5	DMAIWA	0	R/W	指定 DMA 单地址传送时的存取周期之间的等待插入方法 本位指定空闲周期（由 DMAIW[2:0] 位指定）的插入方法。 本位为 0 时，带 DACK 的外部器件驱动数据总线后，在包括本 LSI 的其他器件驱动数据总线时插入空闲周期。带 DACK 的外部器件连续驱动数据总线时，不插入空闲周期。本位为 1 时，即使连续存取带 DACK 的外部器件，也必须在 1 次存取结束后插入空闲周期。 0: 带 DACK 的外部器件驱动数据总线后，在其他器件驱动数据总线时插入空闲周期 1: 存取带 DACK 的外部器件后，总是插入空闲周期
4	—	1	R	保留位 读取值、写入值总是为 1。
3、2	—	均为 0	R	保留位 读取值、写入值总是为 0。
1	HIZMEM	0	R/W	High-Z 存储器控制 本位指定 A25 ~ A0、 $\overline{BS}$ 、 $\overline{CSn}$ 、 $\overline{CE2x}$ 、 $\overline{RD/WR}$ 、 $\overline{WEn/DQMxx/AH}$ 、 $\overline{RD}$ 、 $\overline{FRAME}$ 等在软件待机模式及深度待机模式时的引脚状态。总线释放时，与本位无关，为高阻抗状态。 0: 软件待机模式及深度待机模式时为高阻抗 1: 软件待机模式及深度待机模式时为驱动
0	HIZCNT*	0	R/W	High-Z 控制 本位指定 CKE、 $\overline{RASU}$ 、 $\overline{RASL}$ 、 $\overline{CASU}$ 、 $\overline{CASL}$ 等在软件待机模式、深度待机模式及总线权释放时的引脚状态。 0: CKE、 $\overline{RASU}$ 、 $\overline{RASL}$ 、 $\overline{CASU}$ 、 $\overline{CASL}$ 在软件待机模式、深度待机模式及总线权释放时为高阻抗 1: CKE、 $\overline{RASU}$ 、 $\overline{RASL}$ 、 $\overline{CASU}$ 、 $\overline{CASL}$ 在软件待机模式、深度待机模式及总线权释放时为驱动

【注】 * 有关 CKIO 的 High-Z 控制，详情参阅“第 4 章 时钟脉冲振荡器 (CPG)”。

9.4.2 CSn 空间总线控制寄存器 (CSnBCR) (n=0 ~ 7)

CSnBCR 设定连接至各空间的存储器种类、空间数据总线宽度及存取周期期间的等待数。

完成寄存器的初始设定前，不得存取区域 0 以外的外部存储器。

即使指定无空闲周期，也有可能插入空闲周期，详情参阅“9.5.12 存取周期之间的空闲周期”。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16					
	-	IWW[2:0]				IWRWD[2:0]				IWRWS[2:0]				IWRRD[2:0]				IWRRS[2:0]			
初始值:	0	0	1	1	0	1	1	0	1	1	0	1	1	0	1	1					
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W					

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	TYPE[2:0]				ENDIAN	BSZ[1:0]		-	-	-	-	-	-	-	-
初始值:	0	0	0	0	0	1*	1*	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R

【注】* 上电复位时，采样指定数据总线宽度的外部引脚 (MD) 的值。

位	位名称	初始值	R/W	说 明
31	—	0	R	保留位 读取值、写入值总是为 0。
30 ~ 28	IWW[2:0]	011	R/W	指定写入 - 读取 / 写入 - 写入周期期间的空闲 本位指定存取连接至空间的存储器后插入的空闲周期数。插入的对 象周期为写入 - 读取与写入 - 写入周期。 000: 无空闲周期 001: 插入 1 个空闲周期 010: 插入 2 个空闲周期 011: 插入 4 个空闲周期 100: 插入 6 个空闲周期 101: 插入 8 个空闲周期 110: 插入 10 个空闲周期 111: 插入 12 个空闲周期
27 ~ 25	IWRWD[2:0]	011	R/W	指定不同空间读取 - 写入周期期间的空闲 本位指定存取连接至空间的存储器后插入的空闲周期数。插入的对 象周期是连续存取为不同空间、且为读取 - 写入周期。 000: 无空闲周期 001: 插入 1 个空闲周期 010: 插入 2 个空闲周期 011: 插入 4 个空闲周期 100: 插入 6 个空闲周期 101: 插入 8 个空闲周期 110: 插入 10 个空闲周期 111: 插入 12 个空闲周期

位	位名称	初始值	R/W	说 明
24 ~ 22	IWRWS[2:0]	011	R/W	指定相同空间读取 - 写入周期期间的空闲 本位指定存取连接至空间的存储器后插入的空闲周期数。插入的对象周期是连续存取为相同空间、且为读取 - 写入周期。 000: 无空闲周期 001: 插入 1 个空闲周期 010: 插入 2 个空闲周期 011: 插入 4 个空闲周期 100: 插入 6 个空闲周期 101: 插入 8 个空闲周期 110: 插入 10 个空闲周期 111: 插入 12 个空闲周期
21 ~ 19	IWRRD[2:0]	011	R/W	指定不同空间读取 - 读取周期期间的空闲 本位指定存取连接至空间的存储器后插入的空闲周期数。插入的对象周期是连续存取为不同空间、且为读取 - 读取周期。 000: 无空闲周期 001: 插入 1 个空闲周期 010: 插入 2 个空闲周期 011: 插入 4 个空闲周期 100: 插入 6 个空闲周期 101: 插入 8 个空闲周期 110: 插入 10 个空闲周期 111: 插入 12 个空闲周期
18 ~ 16	IWRRS[2:0]	011	R/W	指定相同空间读取 - 读取周期期间的空闲 本位指定存取连接至空间的存储器后插入的空闲周期数。插入的对象周期是连续存取为相同空间、且为读取 - 读取周期。 000: 无空闲周期 001: 插入 1 个空闲周期 010: 插入 2 个空闲周期 011: 插入 4 个空闲周期 100: 插入 6 个空闲周期 101: 插入 8 个空闲周期 110: 插入 10 个空闲周期 111: 插入 12 个空闲周期
15	—	0	R	保留位 读取值、写入值总是为 0。

位	位名称	初始值	R/W	说 明
14 ~ 12	TYPE[2:0]	000	R/W	本位设定连接至空间的存储器种类。 000: 普通空间 001: 突发 ROM (时钟异步) 010: MPX-I/O 011: 带字节选择的 SRAM 100: SDRAM 101: PCMCIA 110: 突发 MPX-I/O 111: 突发 ROM (时钟同步) 各区域的存储器类型参照表 9.2。 【注】将突发 ROM 连接至 CS0 空间时, 必须在使用 CS0WCR 寄存器的突发 ROM 更改必要设定后, 将 TYPE[2:0] 设定为突发 ROM。
11	ENDIAN	0	R/W	指定字节序 本位指定空间的数据排列。 0: 作为大端法运行 1: 作为小端法运行 【注】不可在区域 0 设定小端法。区域 0 时, 本位的读取值、写入值总是为 0。
10、9	BSZ[1:0]	11*	R/W	指定数据总线宽度 本位指定空间的数据总线宽度。 00: 保留 (禁止设定) 01: 8 位 10: 16 位 11: 32 位 MPX-I/O 时, 根据地址选择总线宽度 【注】 1. 将区域 5 设定为 MPX-I/O 时, 如果本位设定为 11, 则根据 CS5WCR 的 SZSEL 决定的地址, 选择总线宽度 (8 位或 16 位)。另外, 为固定总线宽度时, 可设定 8 或 16 位总线宽度。 2. 区域 0 到 7 的初始状态数据总线宽度由外部引脚设定。此时, 可忽略对 CS0BCR 的 BSZ[1:0] 位的写入, 但可更改 CS1BCR ~ CS7BCR 的总线宽度。 3. 将区域 6 设定为突发 MPX-I/O 时, 总线宽度只可设定为 32 位。 4. 将区域 5 或区域 6 设定为 PCMCIA 空间时, 总线宽度可选择 8 或 16 位。 5. 将区域 2 或区域 3 设定为 SDRAM 空间时, 总线宽度可选择 16 或 32 位。 6. 将区域 0 设定为时钟同步突发 ROM 空间时, 总线宽度可选择 16 或 32 位。 7. 使用 8 位数据总线宽度时, 不可使用区域 7。使用区域 7 时, 必须将所有使用的区域设定为 16 位或 32 位数据总线宽度。
8 ~ 0	—	均为 0	R	保留位 读取值、写入值总是为 0。

【注】 * 上电复位时, 采样指定数据总线宽度的外部引脚 (MD) 值。

9.4.3 CSn 空间等待控制寄存器 (CSnWCR) (n=0 ~ 7)

CSnWCR 设定存储器存取相关的各种等待周期。根据在 CSn 空间总线控制寄存器 (CSnBCR) 设定的存储器种类 (TYPE[2:0])，本寄存器的位结构发生以下变化。存取对象区域前，必须设定 CSnWCR。设定 CSnBCR 后，再设定 CSnWCR。

(1) 普通空间、带字节选择的 SRAM、MPX-I/O

- CS0WCR

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R	R/W	R/W

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	SW[1:0]						WM	-	-	-	-		HW[1:0]
初始值:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W

位	位名称	初始值	R/W	说 明
31 ~ 22	—	均为 0	R	保留位 读取值、写入值总是为 0。
21、20	— *	均为 0	R/W	保留位 普通空间接口时，必须置 0。
19、18	—	均为 0	R	保留位 读取值、写入值总是为 0。
17、16	— *	均为 0	R/W	保留位 普通空间接口时，必须置 0。
15 ~ 13	—	均为 0	R	保留位 读取值、写入值总是为 0。
12、11	SW[1:0]	00	R/W	地址、 $\overline{CS0}$ 有效 → $\overline{RD}$ 、 $\overline{WEn}$ 有效延迟周期数 本位指定地址、 $\overline{CS0}$ 有效到 $\overline{RD}$ 、 $\overline{WEn}$ 有效的延迟周期数。 00: 0.5 个周期 01: 1.5 个周期 10: 2.5 个周期 11: 3.5 个周期

位	位名称	初始值	R/W	说 明
10 ~ 7	WR[3:0]	1010	R/W	存取等待周期数 本位指定读取及写入存取所需的等待周期数。 0000: 无等待周期 0001: 1 个周期 0010: 2 个周期 0011: 3 个周期 0100: 4 个周期 0101: 5 个周期 0110: 6 个周期 0111: 8 个周期 1000: 10 个周期 1001: 12 个周期 1010: 14 个周期 1011: 18 个周期 1100: 24 个周期 1101: 保留 (禁止设定) 1110: 保留 (禁止设定) 1111: 保留 (禁止设定)
6	WM	0	R/W	指定外部等待屏蔽 本位指定外部等待输入有效或忽略。即使存取等待周期数为 0, 本位设定仍有效。 0: 外部等待输入有效 1: 外部等待输入忽视
5 ~ 2	—	均为 0	R	保留位 读取值、写入值总是为 0。
1、0	HW[1:0]	00	R/W	$\overline{\text{RD}}$、$\overline{\text{WEn}}$ 无效 → 地址、$\overline{\text{CS0}}$ 无效延迟周期数 本位指定 $\overline{\text{RD}}$、$\overline{\text{WEn}}$ 无效到地址、$\overline{\text{CS0}}$ 无效的延迟周期数。 00: 0.5 个周期 01: 1.5 个周期 10: 2.5 个周期 11: 3.5 个周期

【注】 * 将突发 ROM 连接至 CS0 空间, 并在启动后转换至突发 ROM 接口时, 由 bit21 及 20 指定突发数, 由 bit17 及 16 指定突发等待周期数后, 必须设定 CS0BCR 的 TYPE[2:0]。不得对上述之外的保留位写入 1。

- CS1WCR、CS7WCR

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	BAS	-	WW[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	SW[1:0]		WR[3:0]			WM	-	-	-	-	HW[1:0]		
初始值:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W

位	位名称	初始值	R/W	说 明
31 ~ 21	—	均为 0	R	保留位 读取值、写入值总是为 0。
20	BAS	0	R/W	带字节的 SRAM 字节存取选择 本位设定带字节选择的 SRAM 接口时的 $\overline{WEn}$ 及 $RD/\overline{WR}$ 信号的时序。 0: $\overline{WEn}$ 在读取 - 写入时序时有效, $RD/\overline{WR}$ 在写入存取周期中有效 1: $\overline{WEn}$ 在读取 - 写入存取周期中有效, $RD/\overline{WR}$ 在写入时序时有效
19	—	0	R	保留位 读取值、写入值总是为 0。
18 ~ 16	WW[2:0]	000	R/W	写入存取等待周期数 本位指定写入存取所需的周期数。 000: 与 WR[3:0] 设定 (读取存取等待数) 的周期相同 001: 无等待周期 010: 1 个周期 011: 2 个周期 100: 3 个周期 101: 4 个周期 110: 5 个周期 111: 6 个周期
15 ~ 13	—	均为 0	R	保留位 读取值、写入值总是为 0。

位	位名称	初始值	R/W	说 明
12、11	SW[1:0]	00	R/W	地址、 $\overline{\text{CSn}}$ 有效 $\rightarrow \overline{\text{RD}}$ 、 $\overline{\text{WEn}}$ 有效延迟周期数 本位指定地址、 $\overline{\text{CSn}}$ 有效到 $\overline{\text{RD}}$ 、 $\overline{\text{WEn}}$ 有效的延迟周期数。 00: 0.5 个周期 01: 1.5 个周期 10: 2.5 个周期 11: 3.5 个周期
10 ~ 7	WR[3:0]	1010	R/W	读取存取等待周期数 本位指定读取存取所需的等待周期数。 0000: 无等待周期 0001: 1 个周期 0010: 2 个周期 0011: 3 个周期 0100: 4 个周期 0101: 5 个周期 0110: 6 个周期 0111: 8 个周期 1000: 10 个周期 1001: 12 个周期 1010: 14 个周期 1011: 18 个周期 1100: 24 个周期 1101: 保留 (禁止设定) 1110: 保留 (禁止设定) 1111: 保留 (禁止设定)
6	WM	0	R/W	指定外部等待屏蔽 本位指定外部等待输入有效或忽略。即使存取等待周期数为 0, 本位设定仍有效。 0: 外部等待输入有效 1: 外部等待输入忽略
5 ~ 2	—	均为 0	R	保留位 读取值、写入值总是为 0。
1、0	HW[1:0]	00	R/W	$\overline{\text{RD}}$ 、 $\overline{\text{WEn}}$ 无效 $\rightarrow$ 地址、 $\overline{\text{CSn}}$ 无效延迟周期数 本位指定 $\overline{\text{RD}}$ 、 $\overline{\text{WEn}}$ 无效到地址、 $\overline{\text{CSn}}$ 无效的延迟周期数。 00: 0.5 个周期 01: 1.5 个周期 10: 2.5 个周期 11: 3.5 个周期

- CS2WCR、CS3WCR

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	BAS	-	-	-	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	WR[3:0]				WM	-	-	-	-	-	-
初始值:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R

位	位名称	初始值	R/W	说 明
31 ~ 21	—	均为 0	R	保留位 读取值、写入值总是为 0。
20	BAS	0	R/W	带字节选择的 SRAM 字节存取选择 本位设定带字节选择的 SRAM 接口时的 $\overline{WEn}$ 及 $RD/\overline{WR}$ 信号的时序。 0: $\overline{WEn}$ 在读取 - 写入时序时有效, $RD/\overline{WR}$ 在写入存取周期中有效 1: $\overline{WEn}$ 在读取 - 写入存取周期中有效, $RD/\overline{WR}$ 在写入时序时有效
19 ~ 11	—	均为 0	R	保留位 读取值、写入值总是为 0。
10 ~ 7	WR[3:0]	1010	R/W	存取等待周期数 本位指定读取及写入存取所需的等待周期数。 0000: 无等待周期 0001: 1 个周期 0010: 2 个周期 0011: 3 个周期 0100: 4 个周期 0101: 5 个周期 0110: 6 个周期 0111: 8 个周期 1000: 10 个周期 1001: 12 个周期 1010: 14 个周期 1011: 18 个周期 1100: 24 个周期 1101: 保留 (禁止设定) 1110: 保留 (禁止设定) 1111: 保留 (禁止设定)
6	WM	0	R/W	指定外部等待屏蔽 本位指定外部等待输入有效或忽略。即使存取等待周期数为 0, 本位设定仍有效。 0: 外部等待输入有效 1: 外部等待输入忽略
5 ~ 0	—	均为 0	R	保留位 读取值、写入值总是为 0。

• CS4WCR

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	BAS	-	WW[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	SW[1:0]		WR[3:0]			WM	-	-	-	-	HW[1:0]		
	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W

位	位名称	初始值	R/W	说 明
31 ~ 21	—	均为 0	R	保留位 读取值、写入值总是为 0。
20	BAS	0	R/W	带字节选择的 SRAM 字节存取选择 本位设定带字节选择的 SRAM 接口时的 $\overline{WEn}$ 及 $RD/\overline{WR}$ 信号的时序。 0: $\overline{WEn}$ 在读取 - 写入时序时有效, $RD/\overline{WR}$ 在写入存取周期中有效 1: $\overline{WEn}$ 在读取 - 写入存取周期中有效, $RD/\overline{WR}$ 在写入时序时有效
19	—	0	R	保留位 读取值、写入值总是为 0。
18 ~ 16	WW[2:0]	000	R/W	写入存取等待周期数 本位指定写入存取所需的周期数。 000: 与 WR[3:0] 设定 (读取存取等待数) 的周期相同 001: 无等待周期 010: 1 个周期 011: 2 个周期 100: 3 个周期 101: 4 个周期 110: 5 个周期 111: 6 个周期
15 ~ 13	—	均为 0	R	保留位 读取值、写入值总是为 0。

位	位名称	初始值	R/W	说 明
12、11	SW[1:0]	00	R/W	地址、 $\overline{CS4}$ 有效 $\rightarrow \overline{RD}$ 、 $\overline{WEn}$ 有效延迟周期数 本位指定地址、 $\overline{CS4}$ 有效到 $\overline{RD}$ 、 $\overline{WEn}$ 有效的延迟周期数。 00: 0.5 个周期 01: 1.5 个周期 10: 2.5 个周期 11: 3.5 个周期
10 ~ 7	WR[3:0]	1010	R/W	读取存取等待周期数 本位指定读取存取所需的等待周期数。 0000: 无等待周期 0001: 1 个周期 0010: 2 个周期 0011: 3 个周期 0100: 4 个周期 0101: 5 个周期 0110: 6 个周期 0111: 8 个周期 1000: 10 个周期 1001: 12 个周期 1010: 14 个周期 1011: 18 个周期 1100: 24 个周期 1101: 保留 (禁止设定) 1110: 保留 (禁止设定) 1111: 保留 (禁止设定)
6	WM	0	R/W	指定外部等待屏蔽 本位指定外部等待输入有效或忽略。即使存取等待周期数为 0, 本位设定仍有效。 0: 外部等待输入有效 1: 外部等待输入忽略
5 ~ 2	—	均为 0	R	保留位 读取值、写入值总是为 0。
1、0	HW[1:0]	00	R/W	$\overline{RD}$ 、 $\overline{WEn}$ 无效 $\rightarrow$ 地址、 $\overline{CS4}$ 无效延迟周期数 本位指定 $\overline{RD}$ 、 $\overline{WEn}$ 无效到地址、 $\overline{CS4}$ 无效的延迟周期数。 00: 0.5 个周期 01: 1.5 个周期 10: 2.5 个周期 11: 3.5 个周期

- CS5WCR

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	SZSEL	MPXW/ BAS	-	WW[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R/W	R/W	R/W

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	SW[1:0]		WR[3:0]			WM	-	-	-	-	-	HW[1:0]	
初始值:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W

位	位名称	初始值	R/W	说 明																		
31 ~ 22	—	均为 0	R	保留位 读取值、写入值总是为 0。																		
21	SZSEL	0	R/W	MPX-I/O 接口总线宽度指定位 本位指定选择 CS5BCR 的 BSZ[1:0] 设定为 11 时的总线宽度的地址。本设定仅在区域 5 设定为 MPX-I/O 时有效。 0: 通过地址 A14 选择总线宽度 1: 通过地址 A21 选择总线宽度 SZSEL 位与通过 A14、A21 选择总线宽度的关系如下所示： <table><tr><td>SZSEL</td><td>A14</td><td>A21</td><td>说明</td></tr><tr><td rowspan="2">0</td><td>0</td><td>无影响</td><td>8 位总线宽度</td></tr><tr><td>1</td><td>无影响</td><td>16 位总线宽度</td></tr><tr><td rowspan="2">1</td><td>无影响</td><td>0</td><td>8 位总线宽度</td></tr><tr><td>无影响</td><td>1</td><td>16 位总线宽度</td></tr></table>	SZSEL	A14	A21	说明	0	0	无影响	8 位总线宽度	1	无影响	16 位总线宽度	1	无影响	0	8 位总线宽度	无影响	1	16 位总线宽度
SZSEL	A14	A21	说明																			
0	0	无影响	8 位总线宽度																			
	1	无影响	16 位总线宽度																			
1	无影响	0	8 位总线宽度																			
	无影响	1	16 位总线宽度																			
20	MPXW	0	R/W	MPX-I/O 接口地址等待 本设定仅在区域 5 设定为 MPX-I/O 时有效。本位设定 MPX-I/O 接口的地址周期插入等待。 0: 无等待 1: 插入 1 个等待周期																		
	BAS	0	R/W	带字节选择的 SRAM 字节存取选择 本设定仅在区域 5 设定为带字节选择的 SRAM 时有效。 本位设定带字节选择的 SRAM 接口时的 $\overline{WEn}$ 及 $\overline{RD}/\overline{WR}$ 信号的时序。 0: $\overline{WEn}$ 在读取 - 写入时序时有效， $\overline{RD}/\overline{WR}$ 在写入存取周期中有效 1: $\overline{WEn}$ 在读取 - 写入存取周期中有效， $\overline{RD}/\overline{WR}$ 在写入时序时有效																		
19	—	0	R	保留位 读取值、写入值总是为 0。																		

位	位名称	初始值	R/W	说 明
18 ~ 16	WW[2:0]	000	R/W	写入存取等待周期数 本位指定写入存取所需的周期数。 000: 与 WR[3:0] 设定 (读取存取等待数) 的周期相同 001: 无等待周期 010: 1 个周期 011: 2 个周期 100: 3 个周期 101: 4 个周期 110: 5 个周期 111: 6 个周期
15 ~ 13	—	均为 0	R	保留位 读取值、写入值总是为 0。
12、11	SW[1:0]	00	R/W	地址、 $\overline{CS5}$ 有效 $\rightarrow \overline{RD}$ 、 $\overline{WEn}$ 有效延迟周期数 本位指定地址、 $\overline{CS5}$ 有效到 $\overline{RD}$ 、 $\overline{WEn}$ 有效的延迟周期数。 00: 0.5 个周期 01: 1.5 个周期 10: 2.5 个周期 11: 3.5 个周期
10 ~ 7	WR[3:0]	1010	R/W	读取存取等待周期数 本位指定读取存取所需的等待周期数。 0000: 无等待周期 0001: 1 个周期 0010: 2 个周期 0011: 3 个周期 0100: 4 个周期 0101: 5 个周期 0110: 6 个周期 0111: 8 个周期 1000: 10 个周期 1001: 12 个周期 1010: 14 个周期 1011: 18 个周期 1100: 24 个周期 1101: 保留 (禁止设定) 1110: 保留 (禁止设定) 1111: 保留 (禁止设定)
6	WM	0	R/W	指定外部等待屏蔽 本位指定外部等待输入有效或忽略。即使存取等待周期数为 0, 本位设定仍有效。 0: 外部等待输入有效 1: 外部等待输入忽略
5 ~ 2	—	均为 0	R	保留位 读取值、写入值总是为 0。
1、0	HW[1:0]	00	R/W	$\overline{RD}$ 、 $\overline{WEn}$ 无效 $\rightarrow$ 地址、 $\overline{CS5}$ 无效延迟周期数 本位指定 $\overline{RD}$ 、 $\overline{WEn}$ 无效到地址、 $\overline{CS5}$ 无效的延迟周期数。 00: 0.5 个周期 01: 1.5 个周期 10: 2.5 个周期 11: 3.5 个周期

• CS6WCR

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	BAS	-	-	-	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	SW[1:0]			WR[3:0]			WM	-	-	-	-	HW[1:0]	
初始值:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W

位	位名称	初始值	R/W	说 明
31 ~ 21	—	均为 0	R	保留位 读取值、写入值总是为 0。
20	BAS	0	R/W	带字节选择的 SRAM 字节存取选择 本位设定带字节选择的 SRAM 接口时的 $\overline{WEn}$ 及 $RD/\overline{WR}$ 信号的时序。 0: $\overline{WEn}$ 在读取 - 写入时序时有效, $RD/\overline{WR}$ 在写入存取周期中有效 1: $\overline{WEn}$ 在读取 - 写入存取周期中有效, $RD/\overline{WR}$ 在写入时序时有效
19 ~ 13	—	均为 0	R	保留位 读取值、写入值总是为 0。
12、11	SW[1:0]	00	R/W	地址、 $\overline{CS6}$ 有效 $\rightarrow \overline{RD}$ 、 $\overline{WEn}$ 有效延迟周期数 本位指定地址、 $\overline{CS6}$ 有效到 $\overline{RD}$ 、 $\overline{WEn}$ 有效的延迟周期数。 00: 0.5 个周期 01: 1.5 个周期 10: 2.5 个周期 11: 3.5 个周期

位	位名称	初始值	R/W	说 明
10 ~ 7	WR[3:0]	1010	R/W	存取等待周期数 本位指定读取及写入存取所需的等待周期数。 0000: 无等待周期 0001: 1 个周期 0010: 2 个周期 0011: 3 个周期 0100: 4 个周期 0101: 5 个周期 0110: 6 个周期 0111: 8 个周期 1000: 10 个周期 1001: 12 个周期 1010: 14 个周期 1011: 18 个周期 1100: 24 个周期 1101: 保留 (禁止设定) 1110: 保留 (禁止设定) 1111: 保留 (禁止设定)
6	WM	0	R/W	指定外部等待屏蔽 本位指定外部等待输入有效或忽略。即使存取等待周期数为 0, 本位设定仍有效。 0: 外部等待输入有效 1: 外部等待输入忽略
5 ~ 2	—	均为 0	R	保留位 读取值、写入值总是为 0。
1、0	HW[1:0]	00	R/W	$\overline{\text{RD}}$、$\overline{\text{WEn}}$ 无效 → 地址、$\overline{\text{CS6}}$ 无效延迟周期数 本位指定 $\overline{\text{RD}}$、$\overline{\text{WEn}}$ 无效到地址、$\overline{\text{CS6}}$ 无效的延迟周期数。 00: 0.5 个周期 01: 1.5 个周期 10: 2.5 个周期 11: 3.5 个周期

(2) 突发 ROM (时钟异步)

• CS0WCR

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	BST[1:0]	-	-	-	BW[1:0]	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R	R/W	R/W

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	W[3:0]	-	-	-	WM	-	-	-	-	-	-
初始值:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R

位	位名称	初始值	R/W	说 明																		
31 ~ 22	—	均为 0	R	保留位 读取值、写入值总是为 0。																		
21、20	BST[1:0]	00	R/W	指定突发数 本位指定产生 16 字节存取时的突发数。BST[1:0] 的 B'11 为保留，因此不得设定。 <table><tr><th>总线幅度</th><th>BST[1:0]</th><th>突发数</th></tr><tr><td rowspan="2">8 位</td><td>00</td><td>16 突发 ×1 次</td></tr><tr><td>01</td><td>4 突发 ×4 次</td></tr><tr><td rowspan="3">16 位</td><td>00</td><td>8 突发 ×1 次</td></tr><tr><td>01</td><td>2 突发 ×4 次</td></tr><tr><td>10</td><td>4-4 或 2-4-2 突发</td></tr><tr><td>32 位</td><td>xx</td><td>4 突发 ×1 次</td></tr></table>	总线幅度	BST[1:0]	突发数	8 位	00	16 突发 ×1 次	01	4 突发 ×4 次	16 位	00	8 突发 ×1 次	01	2 突发 ×4 次	10	4-4 或 2-4-2 突发	32 位	xx	4 突发 ×1 次
总线幅度	BST[1:0]	突发数																				
8 位	00	16 突发 ×1 次																				
	01	4 突发 ×4 次																				
16 位	00	8 突发 ×1 次																				
	01	2 突发 ×4 次																				
	10	4-4 或 2-4-2 突发																				
32 位	xx	4 突发 ×1 次																				
19、18	—	均为 0	R	保留位 读取值、写入值总是为 0。																		
17、16	BW[1:0]	00	R/W	突发等待周期数 本位指定在突发存取时的第 2 次之后（包括第 2 次）的存取周期插入的等待周期数。 00：无等待周期 01：1 个周期 10：2 个周期 11：3 个周期																		
15 ~ 11	—	均为 0	R	保留位 读取值、写入值总是为 0。																		

位	位名称	初始值	R/W	说 明
10 ~ 7	W[3:0]	1010	R/W	存取等待周期数 本位指定在第 1 次的存取周期插入的等待周期数。 0000: 无等待周期 0001: 1 个周期 0010: 2 个周期 0011: 3 个周期 0100: 4 个周期 0101: 5 个周期 0110: 6 个周期 0111: 8 个周期 1000: 10 个周期 1001: 12 个周期 1010: 14 个周期 1011: 18 个周期 1100: 24 个周期 1101: 保留 (禁止设定) 1110: 保留 (禁止设定) 1111: 保留 (禁止设定)
6	WM	0	R/W	指定外部等待屏蔽 本位指定外部等待输入有效或忽略。即使存取等待周期数为 0, 本位设定仍有效。 0: 外部等待输入有效 1: 外部等待输入忽略
5 ~ 0	—	均为 0	R	保留位 读取值、写入值总是为 0。

• CS4WCR

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	BST[1:0]	-	-	-	BW[1:0]	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R	R/W	R/W

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	SW[1:0]	-	-	-	-	-	WM	-	-	-	-	-	HW[1:0]
初始值:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W

位	位名称	初始值	R/W	说 明																		
31 ~ 22	—	均为 0	R	保留位 读取值、写入值总是为 0。																		
21、20	BST[1:0]	00	R/W	指定突发数 本位指定产生 16 字节存取时的突发数。BST[1:0] 的 B'11 为保留，因此不得设定。 <table><tr><th>总线幅度</th><th>BST[1:0]</th><th>突发数</th></tr><tr><td rowspan="2">8 位</td><td>00</td><td>16 突发 ×1 次</td></tr><tr><td>01</td><td>4 突发 ×4 次</td></tr><tr><td rowspan="3">16 位</td><td>00</td><td>8 突发 ×1 次</td></tr><tr><td>01</td><td>2 突发 ×4 次</td></tr><tr><td>10</td><td>4-4 或 2-4-2 突发</td></tr><tr><td>32 位</td><td>xx</td><td>4 突发 ×1 次</td></tr></table>	总线幅度	BST[1:0]	突发数	8 位	00	16 突发 ×1 次	01	4 突发 ×4 次	16 位	00	8 突发 ×1 次	01	2 突发 ×4 次	10	4-4 或 2-4-2 突发	32 位	xx	4 突发 ×1 次
总线幅度	BST[1:0]	突发数																				
8 位	00	16 突发 ×1 次																				
	01	4 突发 ×4 次																				
16 位	00	8 突发 ×1 次																				
	01	2 突发 ×4 次																				
	10	4-4 或 2-4-2 突发																				
32 位	xx	4 突发 ×1 次																				
19、18	—	均为 0	R	保留位 读取值、写入值总是为 0。																		
17、16	BW[1:0]	00	R/W	突发等待周期数 本位指定在突发存取时第 2 次之后（包括第 2 次）的存取周期插入的等待周期数。 00：无等待周期 01：1 个周期 10：2 个周期 11：3 个周期																		
15 ~ 13	—	均为 0	R	保留位 读取值、写入值总是为 0。																		

位	位名称	初始值	R/W	说 明
12、11	SW[1:0]	00	R/W	地址、 $\overline{CS4}$ 有效 $\rightarrow \overline{RD}$ 、 $\overline{WEn}$ 有效延迟周期数 本位指定地址、 $\overline{CS4}$ 有效到 $\overline{RD}$ 、 $\overline{WEn}$ 有效的延迟周期数。 00: 0.5 个周期 01: 1.5 个周期 10: 2.5 个周期 11: 3.5 个周期
10 ~ 7	W[3:0]	1010	R/W	存取等待周期数 本位指定在第 1 次的存取周期插入的等待周期数。 0000: 无等待周期 0001: 1 个周期 0010: 2 个周期 0011: 3 个周期 0100: 4 个周期 0101: 5 个周期 0110: 6 个周期 0111: 8 个周期 1000: 10 个周期 1001: 12 个周期 1010: 14 个周期 1011: 18 个周期 1100: 24 个周期 1101: 保留 (禁止设定) 1110: 保留 (禁止设定) 1111: 保留 (禁止设定)
6	WM	0	R/W	指定外部等待屏蔽 本位指定外部等待输入有效或忽略。即使存取等待周期数为 0, 本位设定仍有效。 0: 外部等待输入有效 1: 外部等待输入忽略
5 ~ 2	—	均为 0	R	保留位 读取值、写入值总是为 0。
1、0	HW[1:0]	00	R/W	$\overline{RD}$ 、 $\overline{WEn}$ 无效 $\rightarrow$ 地址、 $\overline{CS4}$ 无效延迟周期数 本位指定 $\overline{RD}$ 、 $\overline{WEn}$ 无效到地址、 $\overline{CS4}$ 无效的延迟周期数。 00: 0.5 个周期 01: 1.5 个周期 10: 2.5 个周期 11: 3.5 个周期

(3) SDRAM*

• CS2WCR

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	A2CL[1:0]	-	-	-	-	-	-	-	-
初始值:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R

位	位名称	初始值	R/W	说 明
31 ~ 11	—	均为 0	R	保留位 读取值、写入值总是为 0。
10	—	1	R	保留位 读取值、写入值总是为 1。
9	—	0	R	保留位 读取值、写入值总是为 0。
8、7	A2CL[1:0]	10	R/W	区域 2CAS 等待时间 本位指定区域 2 的 CAS 等待时间。 00: 1 个周期 01: 2 个周期 10: 3 个周期 11: 4 个周期
6 ~ 0	—	均为 0	R	保留位 读取值、写入值总是为 0。

【注】 * 将 SDRAM 仅连接至 1 个区域时，必须将区域 3 设定为 SDRAM，此时，区域 2 必须设定为普通空间或带字节选择的 SRAM。

• CS3WCR

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	WTRP[1:0]*	-	WTRCD[1:0]*	-	A3CL[1:0]	-	-	-	TRWL[1:0]*	-	WTRC[1:0]*	-	-	-	-
初始值:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R	R/W	R/W	R	R/W	R/W	R	R	R/W	R/W	R	R/W	R/W

【注】* 区域2与区域3均设定为SDRAM时，WTRP[1:0]、WTRCD[1:0]、TRWL[1:0]、WTRC[1:0] 位的设定通用。

位	位名称	初始值	R/W	说 明
31 ~ 15	—	均为 0	R	保留位 读取值、写入值总是为 0。
14、13	WTRP[1:0]*	00	R/W	预充电完成的等待周期数 指定以下预充电完成的等待最小周期数。 - 从启动自动预充电到对相同存储体发行 ACTV 命令 - 从发行 PRE/PALL 命令到对相同存储体发行 ACTV 命令 - 转移至掉电模式 / 深度掉电模式之前 - 从发行自动刷新时的 PALL 命令到发行 REF 命令 - 从发行自刷新时的 PALL 命令到发行 SELF 命令 区域 2 与区域 3 的设定通用。 00: 无等待周期 01: 1 个周期 10: 2 个周期 11: 3 个周期
12	—	0	R	保留位 读取值、写入值总是为 0。
11、10	WTRCD[1:0]*	01	R/W	ACTV 命令 → READ (A) /WRIT (A) 命令之间的等待周期数 本位指定从发行 ACTV 命令到发行 READ (A) /WRIT (A) 命令的最小等待周期数。区域 2 与区域 3 的设定通用。 00: 无等待周期 01: 1 个周期 10: 2 个周期 11: 3 个周期
9	—	0	R	保留位 读取值、写入值总是为 0。

位	位名称	初始值	R/W	说 明
8、7	A3CL[1:0]	10	R/W	区域 3CAS 等待时间 本位指定区域 3 的 CAS 等待时间。 00: 1 个周期 01: 2 个周期 10: 3 个周期 11: 4 个周期
6、5	—	均为 0	R	保留位 读取值、写入值总是为 0。
4、3	TRWL[1:0]*	00	R/W	预充电启动等待周期数 指定以下预充电启动的等待最小周期数。 - 从本 LSI 发行 WRITA 命令到在 SDRAM 启动自动预充电的周期数从发行 WRITEA 命令到对相同存储体发行 ACTV 命令的周期数。此外, 必须通过各 SDRAM 的数据表, 确认 SDRAM 接受 WRITA 命令几个周期后启动自动预充电。设定本位时, 必须使该周期数不超过本位指定的周期数。 - 从本 LSI 发行 WRIT 命令到发行 PRE 命令的周期数存储体激活模式时, 在相同存储体存取不同的行地址。 区域 2 与区域 3 的设定通用。 00: 无等待周期 01: 1 个周期 10: 2 个周期 11: 3 个周期
2	—	0	R	保留位 读取值、写入值总是为 0。
1、0	WTRC[1:0]*	00	R/W	REF 命令 / 自刷新解除 →ACTV/REF/MRS 命令之间的空闲周期数指定以下命令之间的最小空闲周期数。 - 从发行 REF 命令到发行 ACTV/REF/MRS 命令 - 从解除自刷新到发行 ACTV/REF/MRS 命令 区域 2 与区域 3 的设定通用。 00: 2 个周期 01: 3 个周期 10: 5 个周期 11: 8 个周期

【注】 * 区域 2 与区域 3 均设定为 SDRAM 时, WTRP[1:0]、WTRCD[1:0]、TRWL[1:0]、WTRC[1:0] 位的设定通用。
将 SDRAM 仅连接至 1 个区域时, 必须将区域 3 设定为 SDRAM, 此时, 区域 2 必须设定为普通空间或带字节选择的 SRAM。

(4) PCMCIA

- CS5WCR, CS6WCR

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	SA[1:0]	-	-	-	-	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R	R	R

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	TED[3:0]				PCW[3:0]				WM	-	-	TEH[3:0]			
初始值:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
31 ~ 22	—	均为 0	R	保留位 读取值、写入值总是为 0。
21、20	SA[1:0]	00	R/W	指定空间属性 设定 PCMCIA 接口时, 通过本位指定存储器卡接口或 I/O 卡接口。 - SA1 0: 指定 A25=1 的空间为存储器卡接口 1: 指定 A25=1 的空间为 I/O 卡接口 - SA0 0: 指定 A25=0 的空间为存储器卡接口 1: 指定 A25=0 的空间为 I/O 卡接口
19 ~ 15	—	均为 0	R	保留位 读取值、写入值总是为 0。
14 ~ 11	TED[3:0]	0000	R/W	地址 - $\overline{\text{RD}}/\overline{\text{WE}}$ 有效延迟 本位指定从 PCMCIA 接口的地址输出到存储器卡的 $\overline{\text{RD}}/\overline{\text{WE}}$ 有效或到 I/O 卡的 $\overline{\text{ICIORD}}/\overline{\text{ICIOWR}}$ 有效时的延迟时间。 0000: 0.5 个周期 0001: 1.5 个周期 0010: 2.5 个周期 0011: 3.5 个周期 0100: 4.5 个周期 0101: 5.5 个周期 0110: 6.5 个周期 0111: 7.5 个周期 1000: 8.5 个周期 1001: 9.5 个周期 1010: 10.5 个周期 1011: 11.5 个周期 1100: 12.5 个周期 1101: 13.5 个周期 1110: 14.5 个周期 1111: 15.5 个周期

位	位名称	初始值	R/W	说 明
10 ~ 7	PCW[3:0]	1010	R/W	存取等待周期数 本位设定插入等待状态数。 0000: 3 个周期 0001: 6 个周期 0010: 9 个周期 0011: 12 个周期 0100: 15 个周期 0101: 18 个周期 0110: 22 个周期 0111: 26 个周期 1000: 30 个周期 1001: 33 个周期 1010: 36 个周期 1011: 38 个周期 1100: 52 个周期 1101: 60 个周期 1110: 64 个周期 1111: 80 个周期
6	WM	0	R/W	指定外部等待屏蔽 本位指定外部等待输入有效或忽略。即使存取等待周期数为 0, 本位设定仍有效。 0: 外部等待输入有效 1: 外部等待输入忽略
5、4	—	均为 0	R	保留位 读取值、写入值总是为 0。
3 ~ 0	TEH[3:0]	0000	R/W	$\overline{\text{RD}}/\overline{\text{WE}}$ 无效 – 地址延迟 本位指定从 PCMCIA 接口的存储器卡的 $\overline{\text{RD}}/\overline{\text{WE}}$ 无效或 I/O 卡的 $\overline{\text{ICIOR}}/\overline{\text{ICIORW}}$ 无效时的地址保持时间。 0000: 0.5 个周期 0001: 1.5 个周期 0010: 2.5 个周期 0011: 3.5 个周期 0100: 4.5 个周期 0101: 5.5 个周期 0110: 6.5 个周期 0111: 7.5 个周期 1000: 8.5 个周期 1001: 9.5 个周期 1010: 10.5 个周期 1011: 11.5 个周期 1100: 12.5 个周期 1101: 13.5 个周期 1110: 14.5 个周期 1111: 15.5 个周期

(5) 突发 MPX-I/O

• CS6WCR

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	MPXAW[1:0]	MPXMD	-	-	-	BW[1:0]
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	W[3:0]	WM	-	-	-	-	-
初始值:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R

位	位名称	初始值	R/W	说 明																																											
31 ~ 22	—	均为 0	R	保留位 读取值、写入值总是为 0。																																											
21、20	MPXAW[1:0]	00	R/W	地址周期插入等待数 本位指定在地址周期插入的等待数。 00: 无等待周期 01: 1 个周期 10: 2 个周期 11: 3 个周期																																											
19	MPXMD	0	R/W	指定突发 MPX-I/O 接口模式 本位指定 16 字节存取时的存取模式。 0: 由 16 字节传送，产生 1 次 4 突发 1: 由四字（8 字节）传送，产生 2 次 2 突发 MPXMD=0 时的传送长度 <table border="1"><tr><th>D31</th><th>D30</th><th>D29</th><th>传送长度</th></tr><tr><td rowspan="4">0</td><td rowspan="2">0</td><td>0</td><td>字节（1 字节）</td></tr><tr><td>1</td><td>字（2 字节）</td></tr><tr><td rowspan="2">1</td><td>0</td><td>长字（4 字节）</td></tr><tr><td>1</td><td>保留 四字（8 字节）</td></tr><tr><td rowspan="3">1</td><td rowspan="2">0</td><td>0</td><td>16 字节</td></tr><tr><td>1</td><td>保留（32 字节）</td></tr><tr><td>1</td><td>0</td><td>保留（64 字节）</td></tr></table> MPXMD=1 时的传送长度 <table border="1"><tr><th>D31</th><th>D30</th><th>D29</th><th>传送长度</th></tr><tr><td rowspan="4">0</td><td rowspan="2">0</td><td>0</td><td>字节（1 字节）</td></tr><tr><td>1</td><td>字（2 字节）</td></tr><tr><td rowspan="2">1</td><td>0</td><td>长字（4 字节）</td></tr><tr><td>1</td><td>四字（8 字节）</td></tr><tr><td>1</td><td>0</td><td>0</td><td>保留（32 字节）</td></tr></table>	D31	D30	D29	传送长度	0	0	0	字节（1 字节）	1	字（2 字节）	1	0	长字（4 字节）	1	保留 四字（8 字节）	1	0	0	16 字节	1	保留（32 字节）	1	0	保留（64 字节）	D31	D30	D29	传送长度	0	0	0	字节（1 字节）	1	字（2 字节）	1	0	长字（4 字节）	1	四字（8 字节）	1	0	0	保留（32 字节）
D31	D30	D29	传送长度																																												
0	0	0	字节（1 字节）																																												
		1	字（2 字节）																																												
	1	0	长字（4 字节）																																												
		1	保留 四字（8 字节）																																												
1	0	0	16 字节																																												
		1	保留（32 字节）																																												
	1	0	保留（64 字节）																																												
D31	D30	D29	传送长度																																												
0	0	0	字节（1 字节）																																												
		1	字（2 字节）																																												
	1	0	长字（4 字节）																																												
		1	四字（8 字节）																																												
1	0	0	保留（32 字节）																																												

位	位名称	初始值	R/W	说 明
18	—	0	R	保留位 读取值、写入值总是为 0。
17、16	BW[1:0]	00	R/W	突发等待周期数 本位指定在突发存取时的第 2 次之后（包括第 2 次）的存取周期插入的等待周期数。 00: 无等待周期 01: 1 个周期 10: 2 个周期 11: 3 个周期
15 ~ 11	—	均为 0	R	保留位 读取值、写入值总是为 0。
10 ~ 7	W[3:0]	1010	R/W	存取等待周期数 本位指定在第 1 次突发存取周期或单次存取插入的等待周期数。 0000: 无等待周期 0001: 1 个周期 0010: 2 个周期 0011: 3 个周期 0100: 4 个周期 0101: 5 个周期 0110: 6 个周期 0111: 8 个周期 1000: 10 个周期 1001: 12 个周期 1010: 14 个周期 1011: 18 个周期 1100: 24 个周期 1101: 保留（禁止设定） 1110: 保留（禁止设定） 1111: 保留（禁止设定）
6	WM	0	R/W	指定外部等待屏蔽 本位指定外部等待输入有效或忽略。即使存取等待周期数为 0，本位设定仍有效。 0: 外部等待输入有效 1: 外部等待输入忽略
5 ~ 0	—	均为 0	R	保留位 读取值、写入值总是为 0。

(6) 突发 ROM (时钟同步)

• CS0WCR

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	BW[1:0]
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	W[3:0]				WM	-	-	-	-	-	-
初始值:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R

位	位名称	初始值	R/W	说 明
31 ~ 18	—	均为 0	R	保留位 读取值、写入值总是为 0。
17、16	BW[1:0]	00	R/W	突发等待周期数 本位指定在突发存取时的第 2 次之后 (包括第 2 次) 的存取周期插入的等待周期数。 00: 无等待周期 01: 1 个周期 10: 2 个周期 11: 3 个周期
15 ~ 11	—	均为 0	R	保留位 读取值、写入值总是为 0。
10 ~ 7	W[3:0]	1010	R/W	存取等待周期数 本位指定在第 1 次存取周期插入的等待周期数。 0000: 无等待周期 0001: 1 个周期 0010: 2 个周期 0011: 3 个周期 0100: 4 个周期 0101: 5 个周期 0110: 6 个周期 0111: 8 个周期 1000: 10 个周期 1001: 12 个周期 1010: 14 个周期 1011: 18 个周期 1100: 24 个周期 1101: 保留 (禁止设定) 1110: 保留 (禁止设定) 1111: 保留 (禁止设定)
6	WM	0	R/W	指定外部等待屏蔽 本位指定外部等待输入有效或忽略。即使存取等待周期数为 0, 本位设定仍有效。 0: 外部等待输入有效 1: 外部等待输入忽略
5 ~ 0	—	均为 0	R	保留位 读取值、写入值总是为 0。

9.4.4 SDRAM 控制寄存器 (SDCR)

SDCR 指定 SDRAM 的刷新方法、存取方法及所连接 SDRAM 的种类。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	A2ROW[1:0]	-	-	A2COL[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R/W	R/W

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	DEEP	SLOW	RFSH	RMODE	PDOWN	BACTV	-	-	-	A3ROW[1:0]	-	-	A3COL[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R	R/W	R/W

位	位名称	初始值	R/W	说 明
31 ~ 21	—	均为 0	R	保留位 读取值、写入值总是为 0。
20、19	A2ROW[1:0]	00	R/W	区域 2 的行地址位数 本位指定区域 2 的行地址位数。 00: 11 位 01: 12 位 10: 13 位 11: 保留 (禁止设定)
18	—	0	R	保留位 读取值、写入值总是为 0。
17、16	A2COL[1:0]	00	R/W	区域 2 的列地址位数 本位指定区域 2 的列地址位数。 00: 8 位 01: 9 位 10: 10 位 11: 保留 (禁止设定)
15、14	—	均为 0	R	保留位 读取值、写入值总是为 0。
13	DEEP	0	R/W	深度掉电模式 仅对低功耗 SDRAM 有效。如果在本位为 1 的状态将 RFSH 位及 RMODE 位设置 1, 则发行深度掉电入口命令后, 低功耗 SDRAM 转移至深度掉电模式。 0: 自刷新模式 1: 深度掉电模式
12	SLOW	0	R/W	低频模式 本位指定对 SDRAM 输出命令、地址、写入数据的时序, 及输入读取数据的时序。如果将本位设定为 1, 则与比通常状态晚半个周期的 CKIO 下降沿同步, 对 SDRAM 输出命令、地址及写入数据, 且与比通常状态早半个周期的 CKIO 下降沿同步, 输入来自 SDRAM 的读取数据。由此, 可延长命令、地址、写入数据及读取数据的保持时间。该模式为适用于以低频运行 SDRAM 的模式。 0: 与 CKIO 上升沿同步, 对 SDRAM 输出命令、地址及写入数据; 与 CKIO 上升沿同步, 输入来自 SDRAM 的读取数据。 1: 与 CKIO 下降沿同步, 对 SDRAM 输出命令、地址及写入数据; 与 CKIO 下降沿同步, 输入来自 SDRAM 的读取数据。

位	位名称	初始值	R/W	说 明
11	RFSH	0	R/W	控制刷新 本位指定是否对 SDRAM 执行刷新。 0: 不刷新 1: 刷新
10	RMODE	0	R/W	控制刷新 本位指定 RFSH 位为 1 时, 执行自动刷新或自刷新。如果 RFSH 位设定为 1 且本位也设定为 1, 则设定后转移至自刷新模式; 如果 RFSH 位设定为 1 且本位设定为 0, 则根据 RTCSR、RTCNT 及 RTCOR 寄存器设定的内容执行自动刷新。 0: 执行自动刷新 1: 执行自刷新
9	PDOWN	0	R/W	掉电模式 本位指定 SDRAM 存取结束后是否将 SDRAM 设定为掉电模式。如果本位设定为 1, 则存取结束后将 CKE 引脚设置为低电平, SDRAM 设置为掉电模式。 0: 存取结束后不将 SDRAM 设置为掉电模式 1: 存取结束后将 SDRAM 设置为掉电模式
8	BACTV	0	R/W	存储体激活模式 本位指定通过自动预充电模式 (使用 READA 及 WRITA 命令) 或存储体激活模式 (使用 READ 及 WRIT 命令) 存取。 0: 自动预充电模式 (使用 READA 及 WRITA 命令) 1: 存储体激活模式 (使用 READ 及 WRIT 命令) 【注】 存储体激活模式只可在区域 3 设定。此时, 总线宽度可设定为 16 或 32 位。区域 2 及区域 3 均设定为 SDRAM 时, 必须设定自动预充电模式。
7 ~ 5	—	均为 0	R	保留位 读取值、写入值总是为 0。
4、3	A3ROW[1:0]	00	R/W	区域 3 的行地址位数 本位指定区域 3 的行地址位数。 00: 11 位 01: 12 位 10: 13 位 11: 保留 (禁止设定)
2	—	0	R	保留位 读取值、写入值总是为 0。
1、0	A3COL[1:0]	00	R/W	区域 3 的列地址位数 本位指定区域 3 的列地址位数。 00: 8 位 01: 9 位 10: 10 位 11: 保留 (禁止设定)

9.4.5 刷新定时器控制 / 状态寄存器 (RTCSR)

RTCSR 执行 SDRAM 刷新相关的各种设定。

写入时，必须将写入数据的高 16 位设定为 H'A55A 来解除写入保护。

对刷新定时器计数器 (RTCNT) 执行递增计数的时钟仅在上电复位时与相位匹配，因此必须注意：从 CKS[2:0] 设定为除 B'000 之外的值到使定时器运行的第一个比较匹配标志置位的期间包含误差。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	CMF	CMIE	CKS[2:0]			RRC[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
31 ~ 8	—	均为 0	R	保留位 读取值总是为 0。
7	CMF	0	R/W	比较匹配标志 本位是表示刷新定时器计数器 (RTCNT) 与刷新时间常数寄存器 (RTCOR) 的值匹配的状态标志，在以下条件被置位 / 清除。 0: 清除条件: 在 CMF=1 的状态读取 RTCSR 后，对 CMF 写入 0 时 1: 置位条件: RTCNT=RTCOR 时
6	CMIE	0	R/W	比较匹配中断允许 本位设定 RTCSR 的 CMF 位置 1 时允许 / 禁止由 CMF 产生中断请求。 0: 禁止由 CMF 产生中断请求 1: 允许由 CMF 产生中断请求
5 ~ 3	CKS[2:0]	000	R/W	选择时钟 本位选择对刷新定时器计数器 (RTCNT) 执行递增计数的时钟。 000: 停止递增计数 001: B ϕ /4 010: B ϕ /16 011: B ϕ /64 100: B ϕ /256 101: B ϕ /1024 110: B ϕ /2048 111: B ϕ /4096
2 ~ 0	RRC[2:0]	000	R/W	刷新次数 本位指定刷新定时器计数器 (RTCNT) 与刷新时间常数寄存器 (RTCOR) 的值匹配，并产生刷新请求时的连续刷新次数。该功能可延长产生刷新的周期。 000: 1 次 001: 2 次 010: 4 次 011: 6 次 100: 8 次 101: 保留 (禁止设定) 110: 保留 (禁止设定) 111: 保留 (禁止设定)

9.4.6 刷新定时器计数器 (RTCNT)

RTCNT 为 8 位计数器。由 RTCSR 的 CKS[2:0] 位选择的时钟执行递增计数。如果 RTCNT 与 RTCOR 的值匹配, 则 RTCNT 清 0。递增计数到 255 时, RTCNT 的值返回 0。写入时, 必须将写入数据的高 16 位设为 H'A55A 来解除写入保护。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-								
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
31 ~ 8	—	均为 0	R	保留位 读取值总是为 0
7 ~ 0		均为 0	R/W	8 位计数器

9.4.7 刷新时间常数寄存器 (RTCOR)

RTCOR 为 8 位寄存器。如果 RTCOR 和 RTCNT 的值匹配, 则 RTCSR 的 CMF 位置 1, RTCNT 清 0。

SDCR 的 RFSH 位置 1 时, 由该匹配信号产生刷新请求, 到实际执行刷新前, 一直保持刷新请求。到下一次匹配前仍未处理刷新请求时, 则之前的请求无效。

在总线释放过程中产生刷新请求时, REFOUT 信号可设定为有效。详情参阅 “9.5.6(9) 刷新请求与总线周期的关系” 或 “9.5.13 总线仲裁”。

RTCSR 的 CMIE 位置 1 时, 由该匹配信号产生中断请求, 直到 RTCSR 的 CMF 位清除前, 一直持续输出中断请求。CMF 位的清除仅影响中断, 不会清除刷新请求。可在执行刷新的同时使用中断对刷新请求的次数计数, 还可同时设定刷新及间隔定时器中断。

写入时, 必须将写入数据的高 16 位设为 H'A55A 来解除写入保护。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-								
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
31 ~ 8	—	均为 0	R	保留位 读取值总是为 0。
7 ~ 0		均为 0	R/W	8 位寄存器

9.5 运行说明

9.5.1 字节序 / 存取长度与数据对齐

本 LSI 支持的字节数据排列方法为大端法与小端法，大端法的高位字节 (MSB) 靠近地址 0 侧，小端法的低位字节 (LSB) 靠近地址 0 侧。上电复位后，所有区域的初始状态均为大端法。区域 0 空间不可更改为小端法，但区域 1 ~ 7 空间在不存取目标空间时，可通过设定 CSnBCR 寄存器更改字节序。

另外，区域 1 ~ 7 空间的数据总线宽度，作为普通存储器与带字节选择的 SRAM 时，可选择 8 位、16 位或 32 位；作为 SDRAM 时，可选择 16 位或 32 位；作为 PCMCIA 接口时，可选择 8 位或 16 位；作为 MPX-I/O 时，为 8 位或 16 位固定宽度，或根据存取地址选择 8 位或 16 位可变宽度；作为突发 MPX-I/O 时，为 32 位固定宽度。根据器件的数据总线宽度执行数据对齐，因此为了从 8 位宽度的器件读取长字数据，需读取 4 次。在本 LSI 的各接口之间，自动对齐数据并转换数据长度。在区域 0 空间，通过设定上电复位时的 MD 引脚，固定为 16 或 32 位数据总线宽度。

器件的数据宽度与存取单位之间的关系，如表 9.5 ~ 表 9.10 所示。必须注意：32 位总线宽度及 16 位总线宽度时的选通信号的地址对应，因大端法与小端法的而不同。例如，在 32 位总线宽度的大端法时， $\overline{\text{WE3}}$ 靠近地址 0 侧；为小端法时， $\overline{\text{WE0}}$ 靠近地址 0 侧。区域 0 不可设置为小端法。取指令存在 32 位及 16 位存取，难以在小端法区域配置，因此，必须从大端法区域执行指令。

表 9.5 大端法 32 位外部器件的存取与数据对齐

操作	数据总线				选通信号			
	D31 ~ D24	D23 ~ D16	D15 ~ D8	D7 ~ D0	$\overline{\text{WE3}}$ 、 DQM _{UJ}	$\overline{\text{WE2}}$ 、 DQM _{UL}	$\overline{\text{WE1}}$ 、 DQM _{LU}	$\overline{\text{WE0}}$ 、 DQM _{LL}
地址 0 的字节存取	数据 7 ~ 0	—	—	—	有效	—	—	—
地址 1 的字节存取	—	数据 7 ~ 0	—	—	—	有效	—	—
地址 2 的字节存取	—	—	数据 7 ~ 0	—	—	—	有效	—
地址 3 的字节存取	—	—	—	数据 7 ~ 0	—	—	—	有效
地址 0 的字存取	数据 15 ~ 8	数据 7 ~ 0	—	—	有效	有效	—	—
地址 2 的字存取	—	—	数据 15 ~ 8	数据 7 ~ 0	—	—	有效	有效
地址 0 的长字存取	数据 31 ~ 24	数据 23 ~ 16	数据 15 ~ 8	数据 7 ~ 0	有效	有效	有效	有效

表 9.6 大端法 16 位外部器件的存取与数据对齐

操作		数据总线				选通信号			
		D31 ~ D24	D23 ~ D16	D15 ~ D8	D7 ~ D0	$\overline{\text{WE3}}$ 、 DQMUU	$\overline{\text{WE2}}$ 、 DQMUL	$\overline{\text{WE1}}$ 、 DQMLU	$\overline{\text{WE0}}$ 、 DQMLL
地址 0 的字节存取		—	—	数据 7 ~ 0	—	—	—	有效	—
地址 1 的字节存取		—	—	—	数据 7 ~ 0	—	—	—	有效
地址 2 的字节存取		—	—	数据 7 ~ 0	—	—	—	有效	—
地址 3 的字节存取		—	—	—	数据 7 ~ 0	—	—	—	有效
地址 0 的字存取		—	—	数据 15 ~ 8	数据 7 ~ 0	—	—	有效	有效
地址 2 的字存取		—	—	数据 15 ~ 8	数据 7 ~ 0	—	—	有效	有效
地址 0 的 长字存取	第 1 次 (地址 0)	—	—	数据 31 ~ 24	数据 23 ~ 16	—	—	有效	有效
	第 2 次 (地址 2)	—	—	数据 15 ~ 8	数据 7 ~ 0	—	—	有效	有效

表 9.7 大端法 8 位外部器件的存取与数据对齐

操作		数据总线				选通信号			
		D31 ~ D24	D23 ~ D16	D15 ~ D8	D7 ~ D0	$\overline{\text{WE3}}$ 、 DQMUU	$\overline{\text{WE2}}$ 、 DQMUL	$\overline{\text{WE1}}$ 、 DQMLU	$\overline{\text{WE0}}$ 、 DQMLL
地址 0 的字节存取		—	—	—	数据 7 ~ 0	—	—	—	有效
地址 1 的字节存取		—	—	—	数据 7 ~ 0	—	—	—	有效
地址 2 的字节存取		—	—	—	数据 7 ~ 0	—	—	—	有效
地址 3 的字节存取		—	—	—	数据 7 ~ 0	—	—	—	有效
地址 0 的 字存取	第 1 次 (地址 0)	—	—	—	数据 15 ~ 8	—	—	—	有效
	第 2 次 (地址 1)	—	—	—	数据 7 ~ 0	—	—	—	有效
地址 2 的 字存取	第 1 次 (地址 2)	—	—	—	数据 15 ~ 8	—	—	—	有效
	第 2 次 (地址 3)	—	—	—	数据 7 ~ 0	—	—	—	有效
地址 0 的 长字存取	第 1 次 (地址 0)	—	—	—	数据 31 ~ 24	—	—	—	有效
	第 2 次 (地址 1)	—	—	—	数据 23 ~ 16	—	—	—	有效
	第 3 次 (地址 2)	—	—	—	数据 15 ~ 8	—	—	—	有效
	第 4 次 (地址 3)	—	—	—	数据 7 ~ 0	—	—	—	有效

表 9.8 小端法 32 位外部器件的存取与数据对齐

操作	数据总线				选通信号			
	D31 ~ D24	D23 ~ D16	D15 ~ D8	D7 ~ D0	$\overline{\text{WE3}}$ 、 DQMUU	$\overline{\text{WE2}}$ 、 DQMUL	$\overline{\text{WE1}}$ 、 DQMLU	$\overline{\text{WE0}}$ 、 DQMLL
地址 0 的字节存取	—	—	—	数据 7 ~ 0	—	—	—	有效
地址 1 的字节存取	—	—	数据 7 ~ 0	—	—	—	有效	—
地址 2 的字节存取	—	数据 7 ~ 0	—	—	—	有效	—	—
地址 3 的字节存取	数据 7 ~ 0	—	—	—	有效	—	—	—
地址 0 的字存取	—	—	数据 15 ~ 8	数据 7 ~ 0	—	—	有效	有效
地址 2 的字存取	数据 15 ~ 8	数据 7 ~ 0	—	—	有效	有效	—	—
地址 0 的长字存取	数据 31 ~ 24	数据 23 ~ 16	数据 15 ~ 8	数据 7 ~ 0	有效	有效	有效	有效

表 9.9 小端法 16 位外部器件的存取和数据对齐

操作		数据总线				选通信号			
		D31 ~ D24	D23 ~ D16	D15 ~ D8	D7 ~ D0	$\overline{\text{WE3}}$ 、 DQMUU	$\overline{\text{WE2}}$ 、 DQMUL	$\overline{\text{WE1}}$ 、 DQMLU	$\overline{\text{WE0}}$ 、 DQMLL
地址 0 的字节存取		—	—	—	数据 7 ~ 0	—	—	—	有效
地址 1 的字节存取		—	—	数据 7 ~ 0	—	—	—	有效	—
地址 2 的字节存取		—	—	—	数据 7 ~ 0	—	—	—	有效
地址 3 的字节存取		—	—	数据 7 ~ 0	—	—	—	有效	—
地址 0 的字存取		—	—	数据 15 ~ 8	数据 7 ~ 0	—	—	有效	有效
地址 2 的字存取		—	—	数据 15 ~ 8	数据 7 ~ 0	—	—	有效	有效
地址 0 的 长字存取	第 1 次 (地址 0)	—	—	数据 15 ~ 8	数据 7 ~ 0	—	—	有效	有效
	第 2 次 (地址 2)	—	—	数据 31 ~ 24	数据 23 ~ 16	—	—	有效	有效

表 9.10 小端法 8 位外部器件的存取与数据对齐

操作		数据总线				选通信号			
		D31 ~ D24	D23 ~ D16	D15 ~ D8	D7 ~ D0	$\overline{\text{WE3}}$ 、 DQMUU	$\overline{\text{WE2}}$ 、 DQMUL	$\overline{\text{WE1}}$ 、 DQMLU	$\overline{\text{WE0}}$ 、 DQMLL
地址 0 的字节存取		—	—	—	数据 7 ~ 0	—	—	—	有效
地址 1 的字节存取		—	—	—	数据 7 ~ 0	—	—	—	有效
地址 2 的字节存取		—	—	—	数据 7 ~ 0	—	—	—	有效
地址 3 的字节存取		—	—	—	数据 7 ~ 0	—	—	—	有效
地址 0 的 字存取	第 1 次 (地址 0)	—	—	—	数据 7 ~ 0	—	—	—	有效
	第 2 次 (地址 1)	—	—	—	数据 15 ~ 8	—	—	—	有效
地址 2 的 字存取	第 1 次 (地址 2)	—	—	—	数据 7 ~ 0	—	—	—	有效
	第 2 次 (地址 3)	—	—	—	数据 15 ~ 8	—	—	—	有效
地址 0 的 长字存取	第 1 次 (地址 0)	—	—	—	数据 7 ~ 0	—	—	—	有效
	第 2 次 (地址 1)	—	—	—	数据 15 ~ 8	—	—	—	有效
	第 3 次 (地址 2)	—	—	—	数据 23 ~ 16	—	—	—	有效
	第 4 次 (地址 3)	—	—	—	数据 31 ~ 24	—	—	—	有效

9.5.2 普通空间接口

(1) 基本时序

普通空间存取时，主要考虑与无字节选择引脚的 SRAM 直接连接而输出选通信号。使用带字节选择引脚的 SRAM 时，参阅“9.5.8 带字节选择的 SRAM 接口”。普通空间存取的基本时序如图 9.2 所示。以 2 个周期结束无等待的普通存取。 $\overline{BS}$ 信号表示总线周期开始，且为 1 个周期有效。

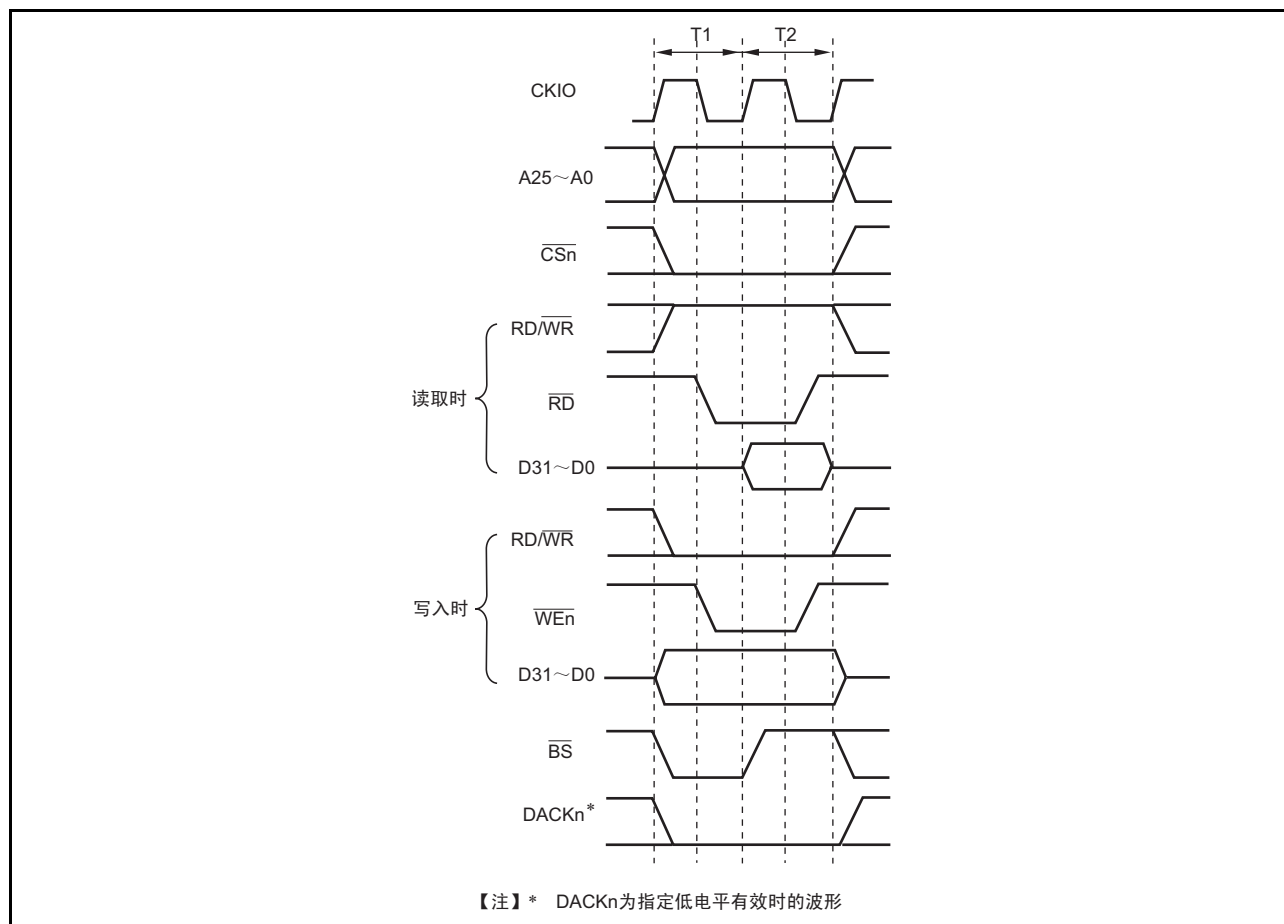


图 9.2 普通空间的基本存取（存取等待 0）

读取时，不对外部总线指定存取长度。向地址的最低位输出正确的存取起始地址，因为未指定存取长度，所以 32 位器件读取 32 位数据，16 位器件读取 16 位数据。写入时，仅限写入字节的 $\overline{WE}$ 信号有效。

在数据总线设置缓冲器时，需要使用 $\overline{RD}$ 执行读取数据的输出控制。必须注意： $\overline{RD}/\overline{WR}$ 信号在未执行存取时为读取状态（高电平输出），使用 $\overline{RD}/\overline{WR}$ 信号控制外接数据缓冲器时，有可能产生输出冲突。

普通空间连续存取的例子如图 9.3、图 9.4 所示。 $\overline{CSnWCR}$ 的 WM 位设定为 0 时，为了评价外部等待，需在存取设定的 $\overline{CSn}$ 空间后插入 1 个 T_{nop} 周期（图 9.3）。 $\overline{CSnWCR}$ 的位 WM 设定为 1 时，可忽略外部等待，并抑制插入 T_{nop} 周期（图 9.4）。

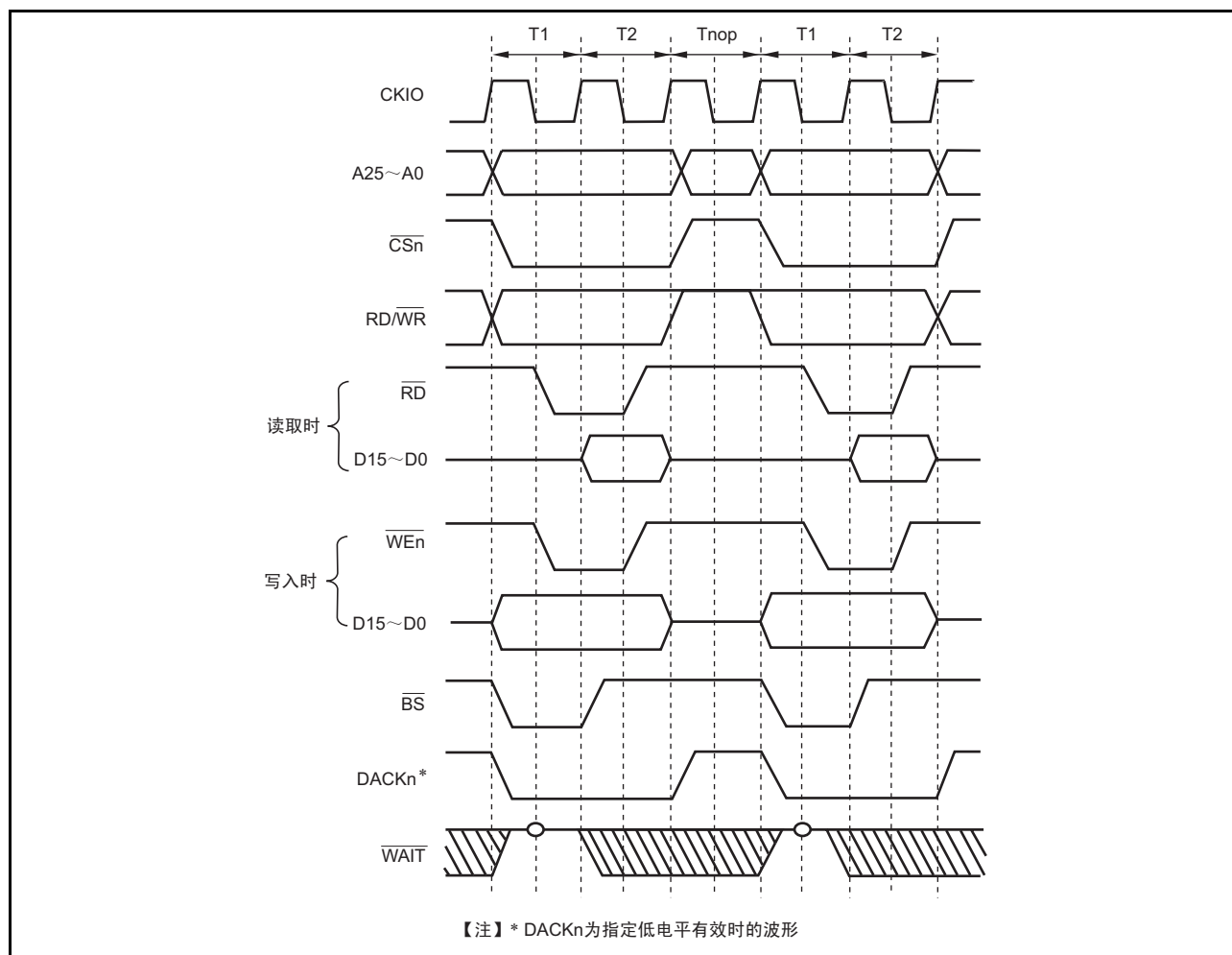


图 9.3 普通空间连续存取的例子 1
 16 位总线宽度、长字存取、CSnWCR.WM 位 =0
 (0 个存取等待、0 个周期间等待)

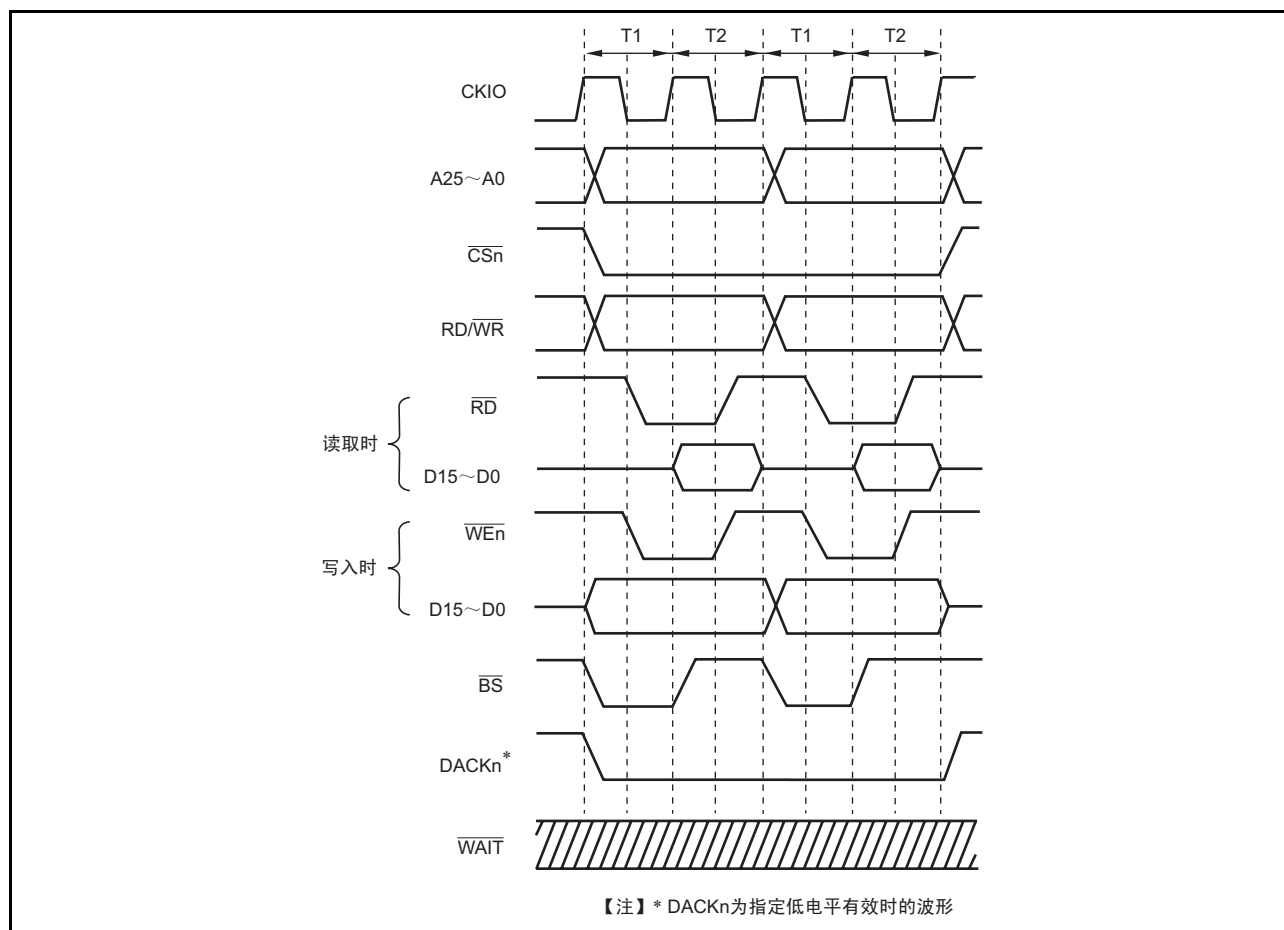


图 9.4 普通空间连续存取的例子 2
 16 位总线宽度、长字存取、CSnWCR.WM 位 =1
 (0 个存取等待、0 个周期间等待)

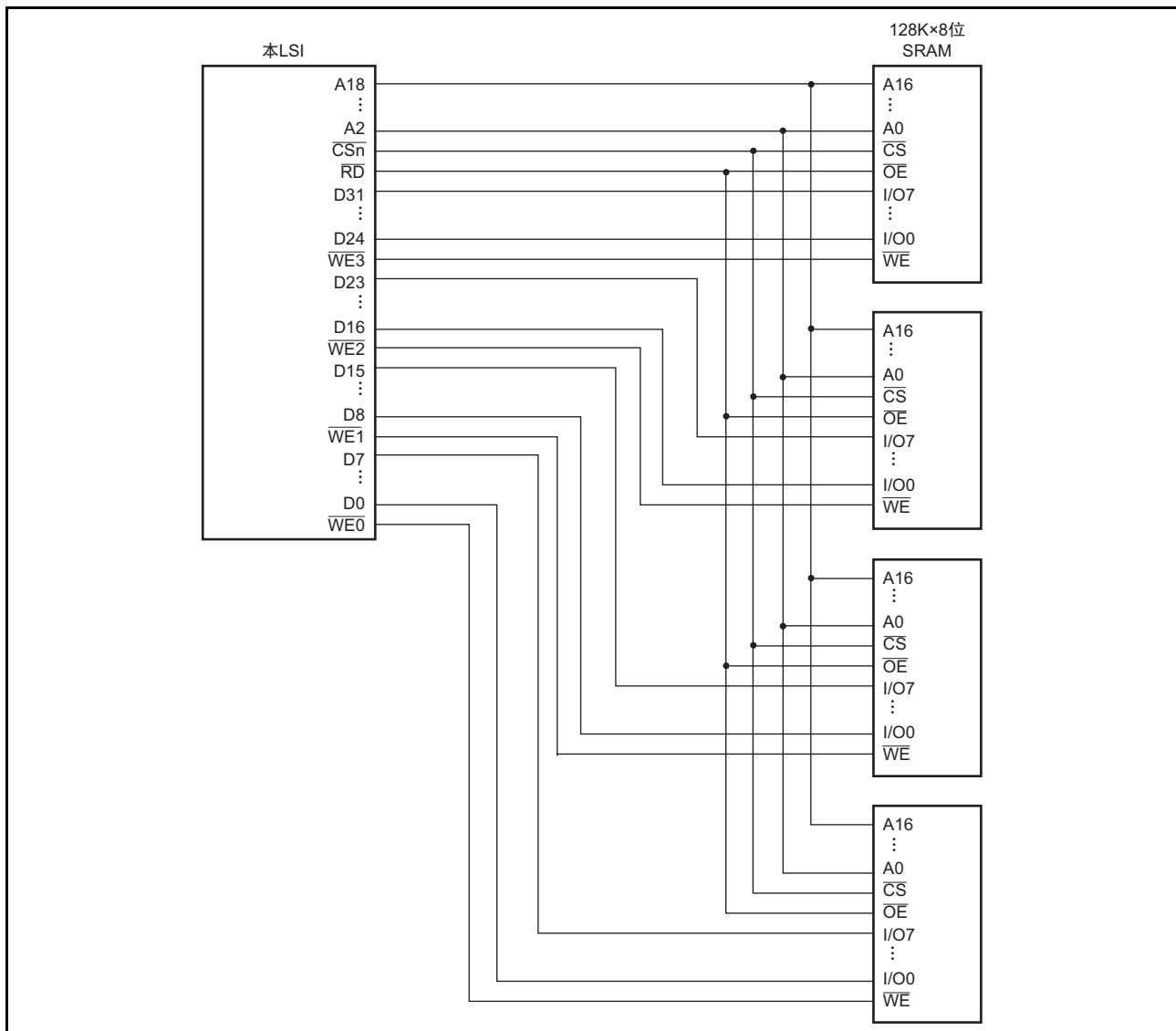


图 9.5 32 位数据宽度的 SRAM 连接例

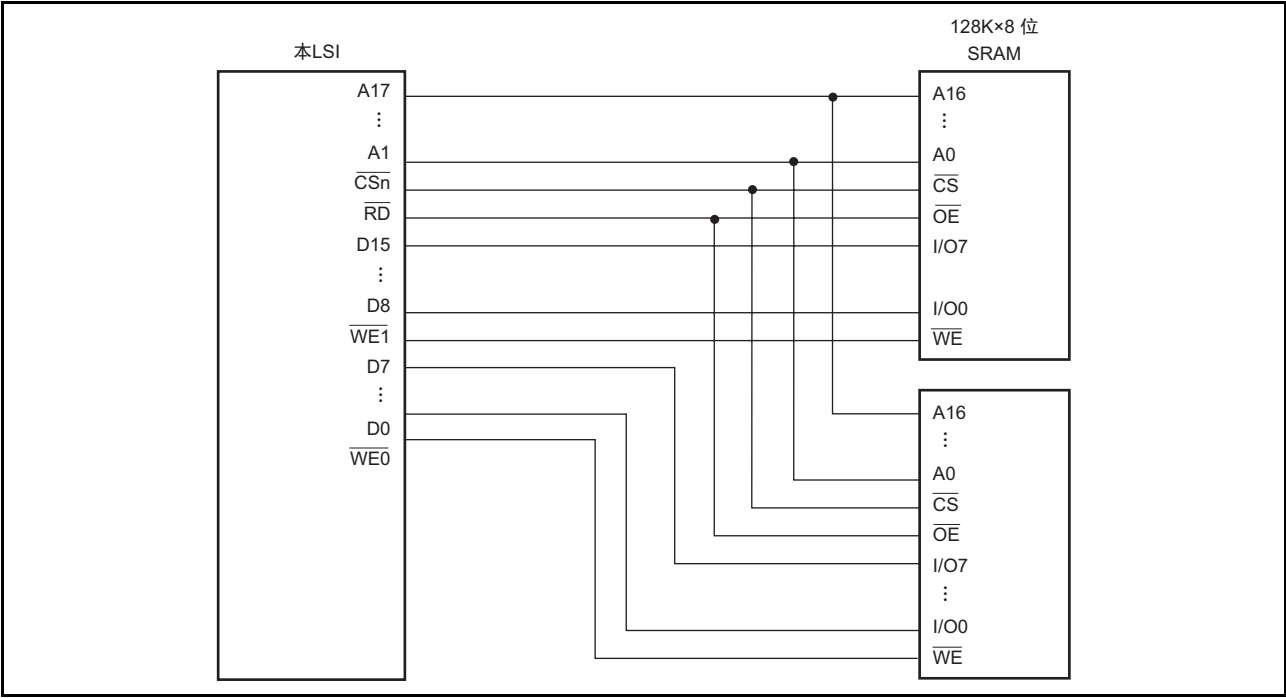


图 9.6 16 位数据宽度的 SRAM 连接例

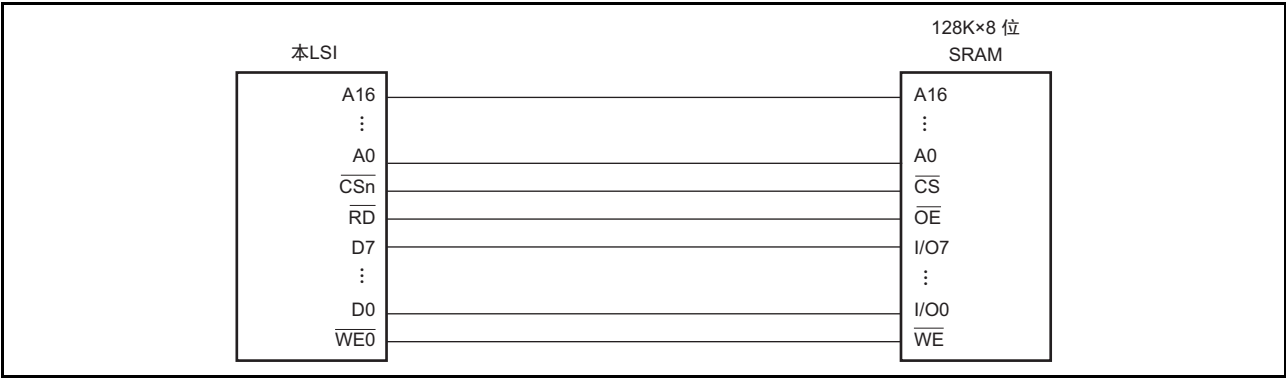


图 9.7 8 位数据宽度的 SRAM 连接例

9.5.3 存取等待控制

通过设定 CSnWCR 的 WR[3:0] 位, 可控制普通空间存取的插入等待周期。读取存取 / 写入存取时, 可在区域 1、4、5、7 单独插入等待周期。在读取 / 写入周期, 区域 0、2、3、6 的存取等待为通用。如图 9.8 所示, 普通空间存取时, 仅插入被指定为等待周期的 Tw 周期数。

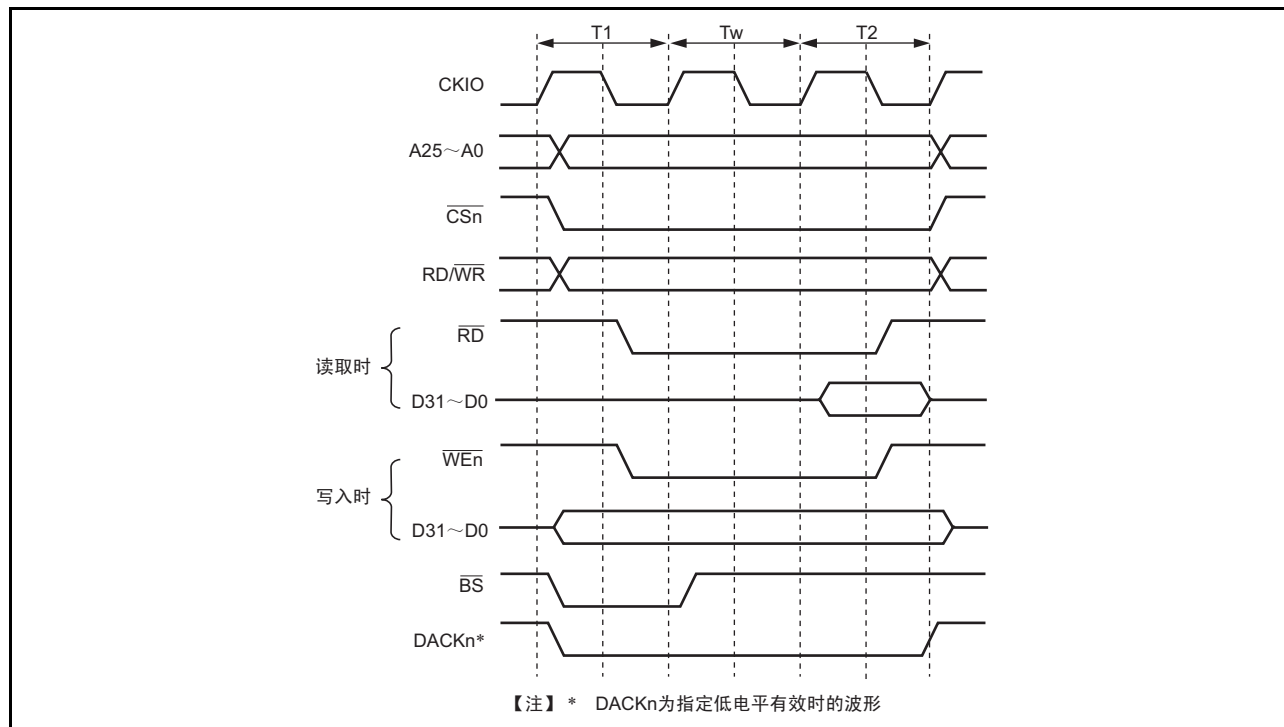


图 9.8 普通空间存取的等待时序 (仅限软件等待)

CSnWCR 的 WM 位设定为 0 时, 外部的等待输入 $\overline{\text{WAIT}}$ 信号也被采样。 $\overline{\text{WAIT}}$ 信号的采样如图 9.9 所示, 指定 2 个周期的等待为软件等待。从 T1 周期或 Tw 周期转移至 T2 周期时, 在 CKIO 的下降沿采样 $\overline{\text{WAIT}}$ 信号。

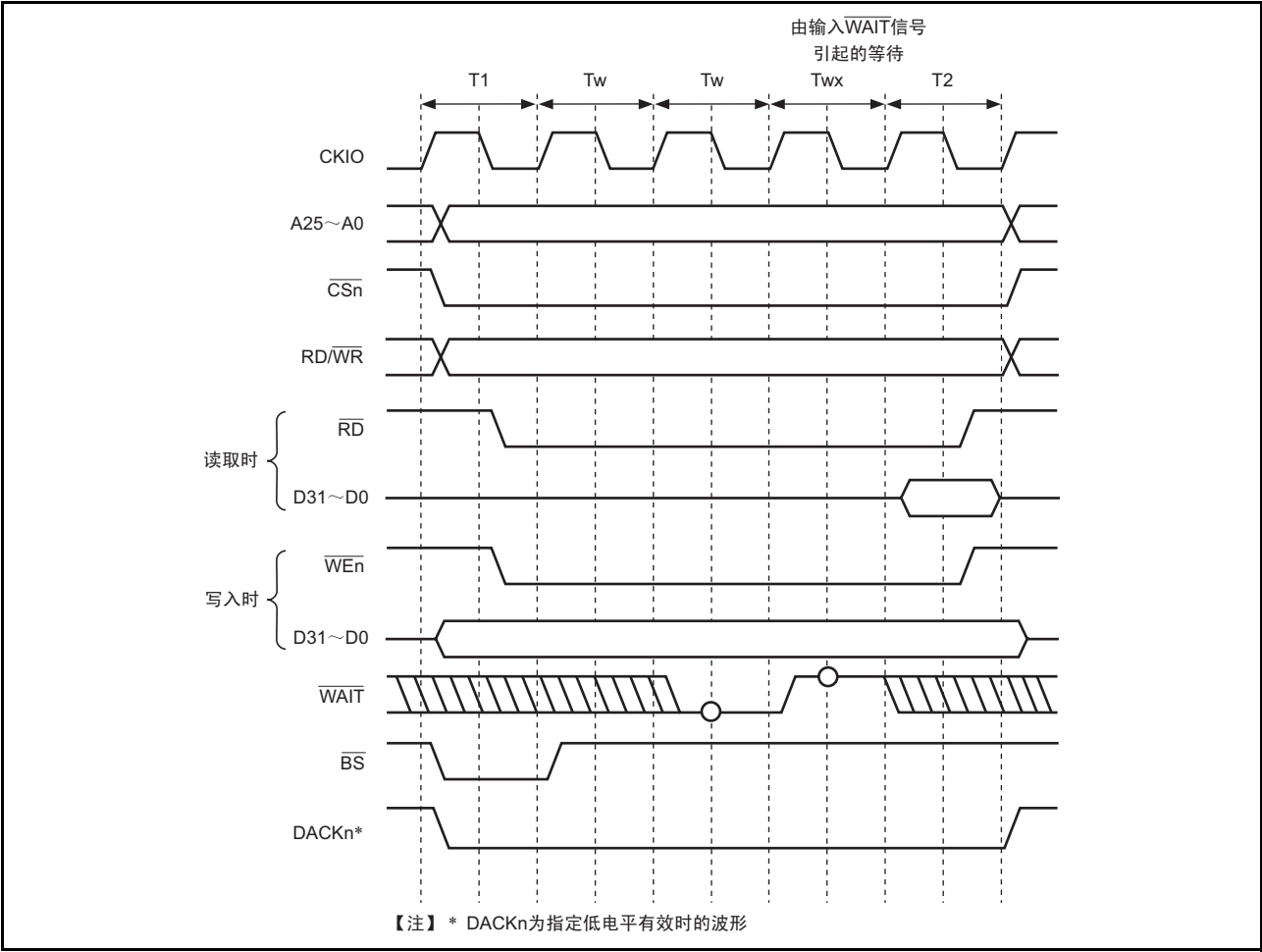


图 9.9 普通空间存取的等待时序（由 $\overline{WAIT}$ 信号插入等待）

9.5.4 $\overline{\text{CSn}}$ 有效期间的扩展

通过设定 $\overline{\text{CSnWCR}}$ 的 $\text{SW}[1:0]$ 位, 可指定从 $\overline{\text{CSn}}$ 有效到 $\overline{\text{RD}}$ 及 $\overline{\text{WEn}}$ 有效的周期数。通过设定 $\text{HW}[1:0]$ 位, 可指定从 $\overline{\text{RD}}$ 与 $\overline{\text{WEn}}$ 无效到 $\overline{\text{CSn}}$ 无效的周期数。由此, 可对应多种外部器件的接口。如图 9.10 所示, T_h 周期及 T_f 周期分别附加于普通周期的前、后。在这些周期, 除 $\overline{\text{RD}}$ 与 $\overline{\text{WEn}}$ 无效之外, 其他信号均有效。数据输出延长至 T_f 周期, 对写入操作较慢的器件等有效。

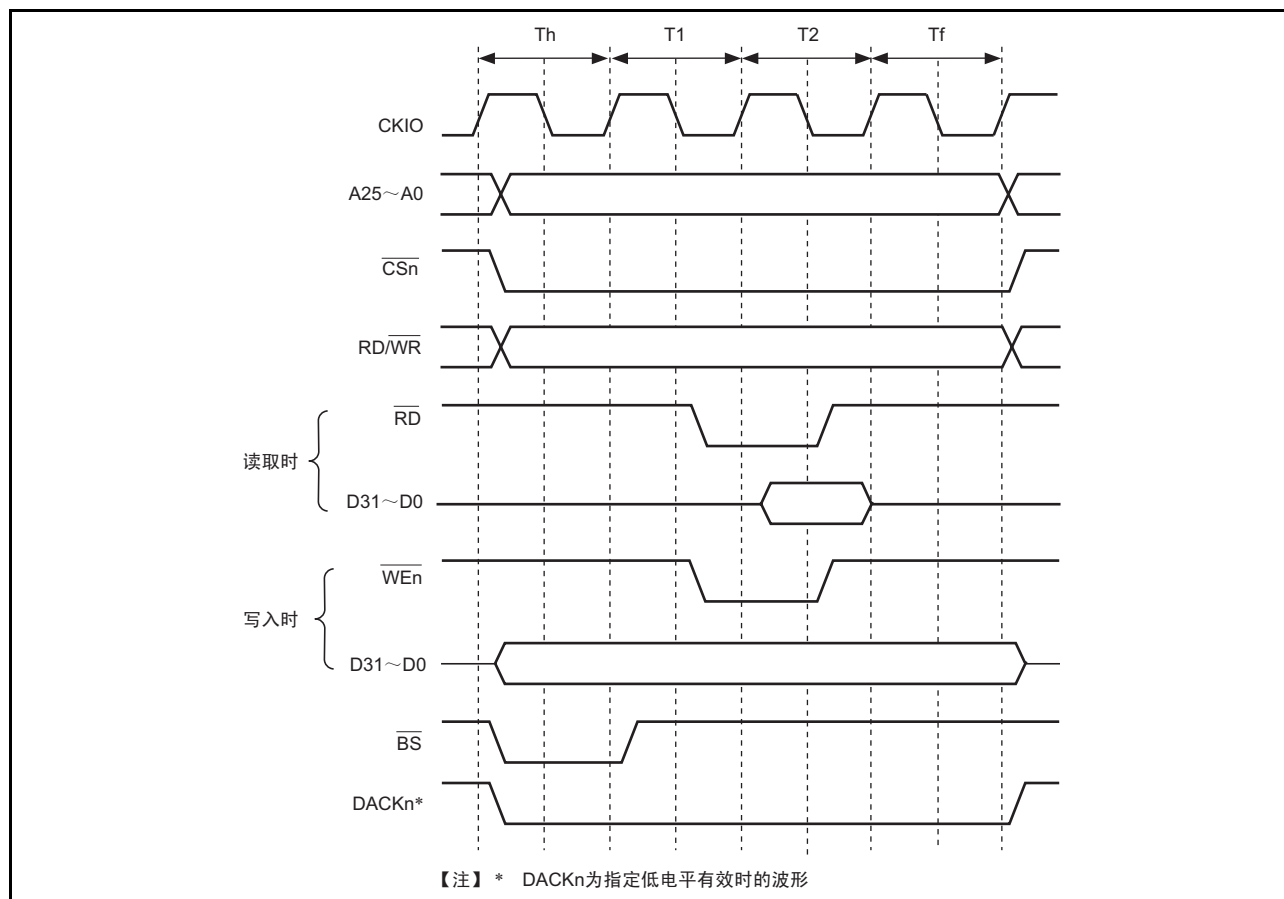


图 9.10 $\overline{\text{CSn}}$ 有效期间的扩展

9.5.5 MPX-I/O 接口

MPX 空间的存取时序如下。在 MPX 空间，通过 $\overline{\text{CS5}}$ 、 $\overline{\text{AH}}$ 、 $\overline{\text{RD}}$ 及 $\overline{\text{WEn}}$ 信号控制存取。MPX 空间的基本存取是先执行 2 个周期的地址输出，再继续存取普通空间。地址输出周期及数据输入 / 输出周期的总线宽度为 8 位或 16 位固定宽度，或根据存取地址为 8 位或 16 位可变宽度。

在 Ta2 周期 ~ Ta3 周期之间，从 D15 ~ D0 或 D7 ~ D0 输出地址。Ta1 周期为高阻抗状态，即使连续存取时不插入空闲周期也可避免地址与数据的冲突。通过将 CS5WCR 的 MPXW 位设定为 1，地址输出变为 3 个周期。

在与 $\overline{\text{CS5}}$ 信号相同的时序输出 $\overline{\text{RD}}/\overline{\text{WR}}$ 信号。为读取周期时，输出高电平；为写入周期时，输出低电平。数据周期与普通空间存取的周期相同。

时序图如图 9.11、图 9.12、图 9.13 所示。

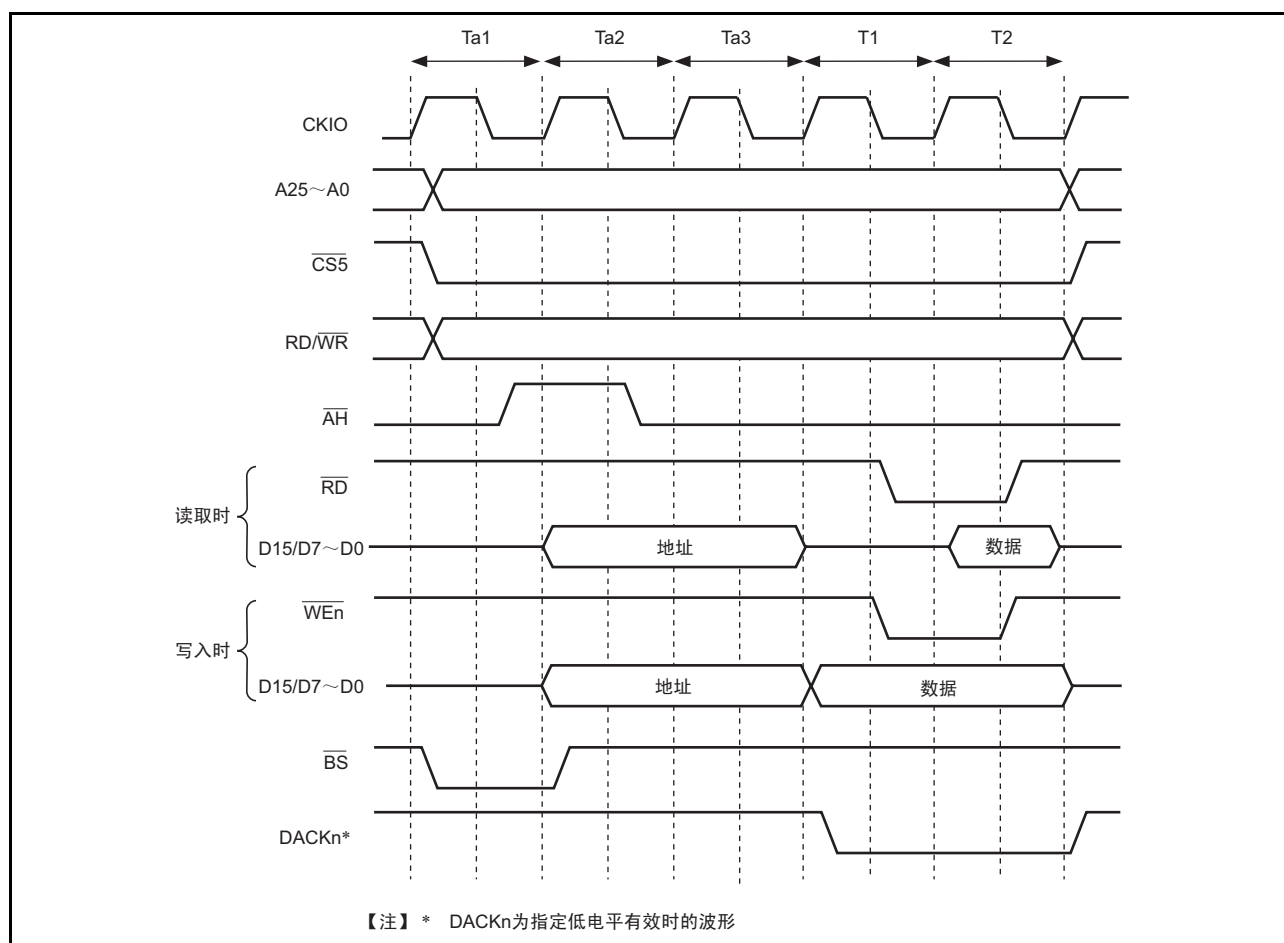


图 9.11 MPX 空间的存取时序
(地址周期无等待、数据周期无等待)

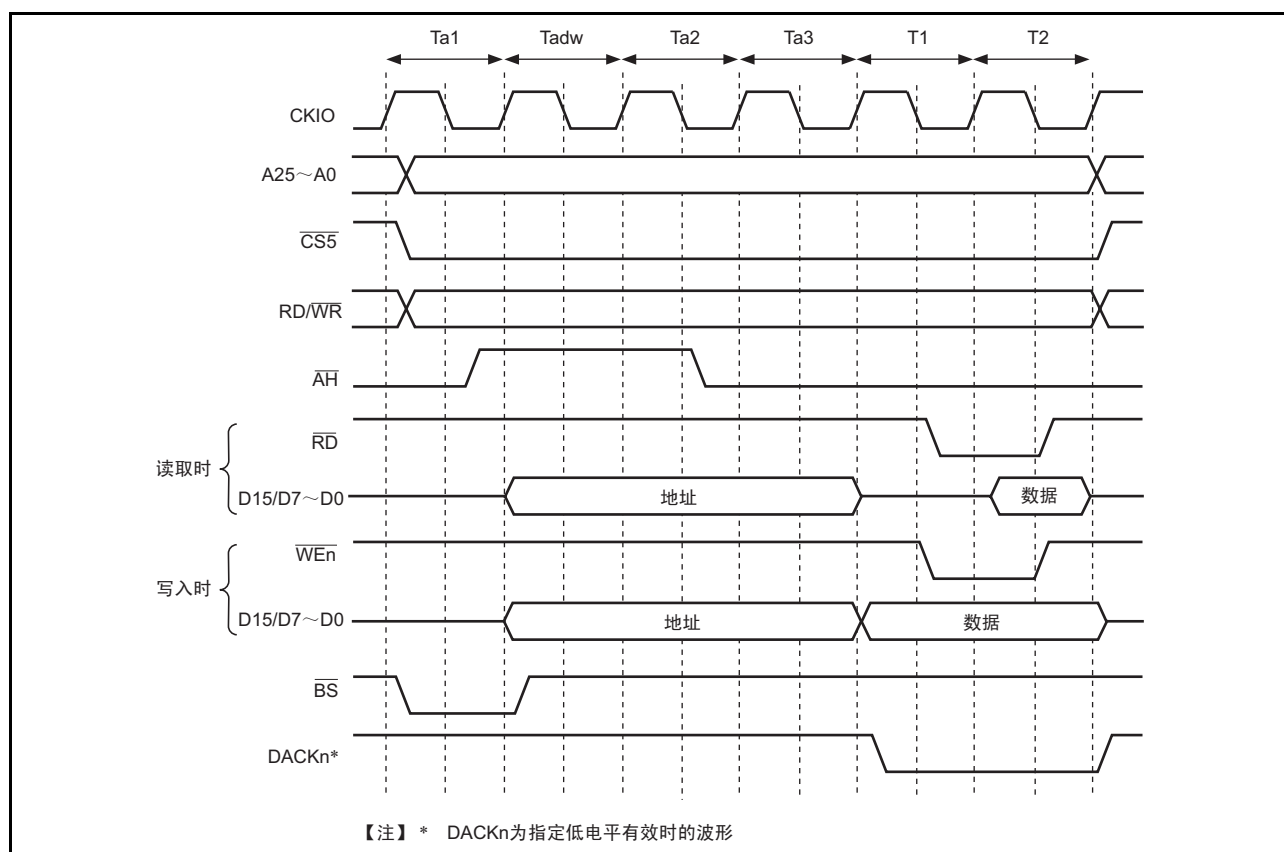


图 9.12 MPX 空间的存取时序
(1 个地址周期等待、无数据周期等待)

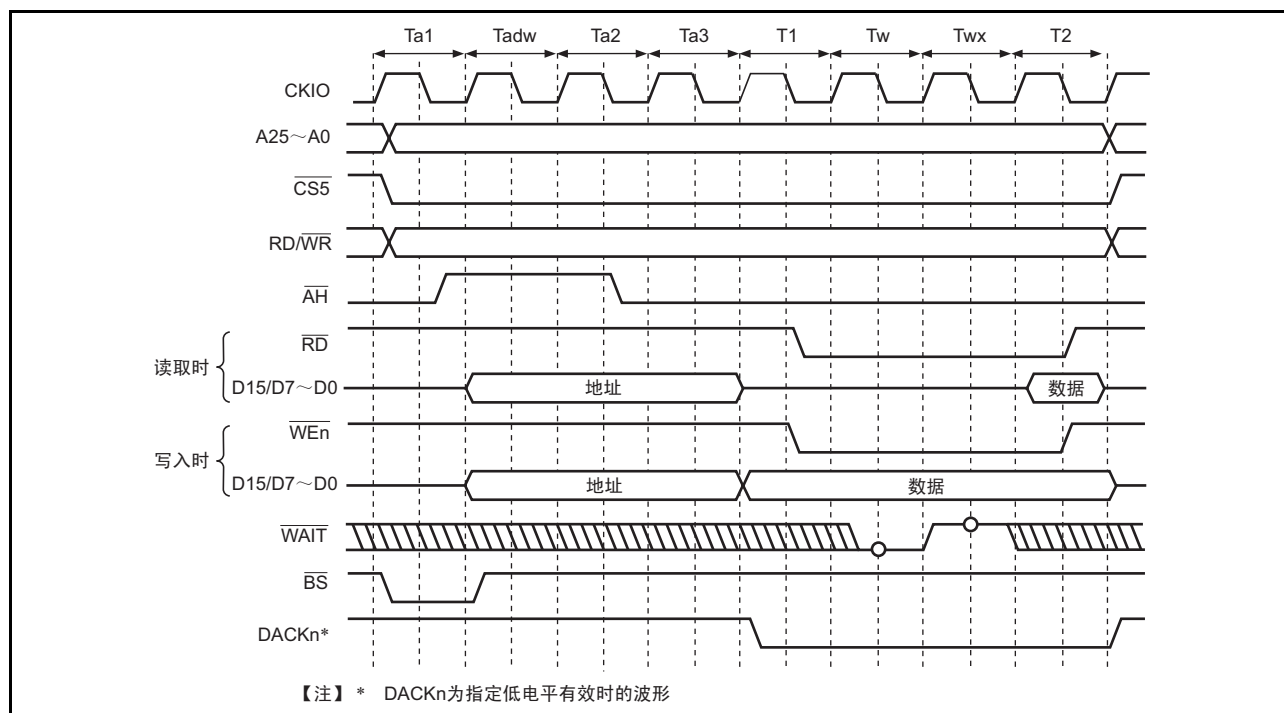


图 9.13 MPX 空间的存取时序
(1 个地址周期存取等待、1 个数据周期等待、1 个外部等待)

9.5.6 SDRAM 接口

(1) 直接连接 SDRAM 的接口

可连接至本 LSI 的 SDRAM 产品为：行地址为 11/12/13 位；列地址为 8/9/10 位；最多 4 个存储体；并在读取 - 写入命令周期使用 A10 引脚设定预充电模式。

直接连接 SDRAM 所需的控制信号为 $\overline{\text{RASU}}$ 、 $\overline{\text{RASL}}$ 、 $\overline{\text{CASU}}$ 、 $\overline{\text{CASL}}$ 、 $\text{RD}/\overline{\text{WR}}$ 、 DQM_{UU} 、 DQM_{UL} 、 DQM_{LU} 、 DQM_{LL} 、 CKE 、 $\overline{\text{CS2}}$ 及 $\overline{\text{CS3}}$ 。除 $\overline{\text{CS2}}$ 与 $\overline{\text{CS3}}$ 之外，所有信号在各空间通用， $\overline{\text{CS2}}$ 或 $\overline{\text{CS3}}$ 有效时，除 CKE 之外的信号均有效。最多可将 SDRAM 连接至 2 个空间。连接 SDRAM 的空间数据总线宽度，可设定为 32 位或 16 位。

SDRAM 运行模式支持突发读取 / 单次写入（突发长度为 1）及突发读取 / 突发写入（突发长度为 1）。

通过 $\overline{\text{RASU}}$ 、 $\overline{\text{RASL}}$ 、 $\overline{\text{CASU}}$ 、 $\overline{\text{CASL}}$ 、 $\text{RD}/\overline{\text{WR}}$ 及特定的地址信号，可指定 SDRAM 的命令。此命令支持 NOP、自动刷新（REF）、自刷新（SELF）、全部存储体预充电（PALL）、指定存储体预充电（PRE）、存储体激活（ACTV）、读取（READ）、带预充电的读取（READA）、写入（WRIT）、带预充电的写入（WRITA）及模式寄存器写入（MRS、EMRS）等。

由 DQM_{UU} 、 DQM_{UL} 、 DQM_{LU} 及 DQM_{LL} ，指定要存取的字节。对应的 DQM_{xx} 对低电平的字节进行读取 / 写入。 DQM_{xx} 与要存取字节的关系，参阅“9.5.1 字节序 / 存取长度与数据对齐”。

本 LSI 与 SDRAM 的连接例如图 9.14 ~ 图 9.16 所示。

如图 9.16 所示，通过在相同的 CS 空间使用 $\overline{\text{RASU}}$ 、 $\overline{\text{RASL}}$ 、 $\overline{\text{CASU}}$ 及 $\overline{\text{CASL}}$ ，本 LSI 可连接 2 个小于等于 32M 字节的 SDRAM 组。此时，相同的 CS 空间有 8 个存储体，即 $\overline{\text{RASL}}$ 与 $\overline{\text{CASL}}$ 指定的 4 个存储体、 $\overline{\text{RASU}}$ 与 $\overline{\text{CASU}}$ 指定的 4 个存储体。存取 A25=0 的地址时， $\overline{\text{RASL}}$ 与 $\overline{\text{CASL}}$ 有效；存取 A25=1 的地址时， $\overline{\text{RASU}}$ 与 $\overline{\text{CASU}}$ 有效。

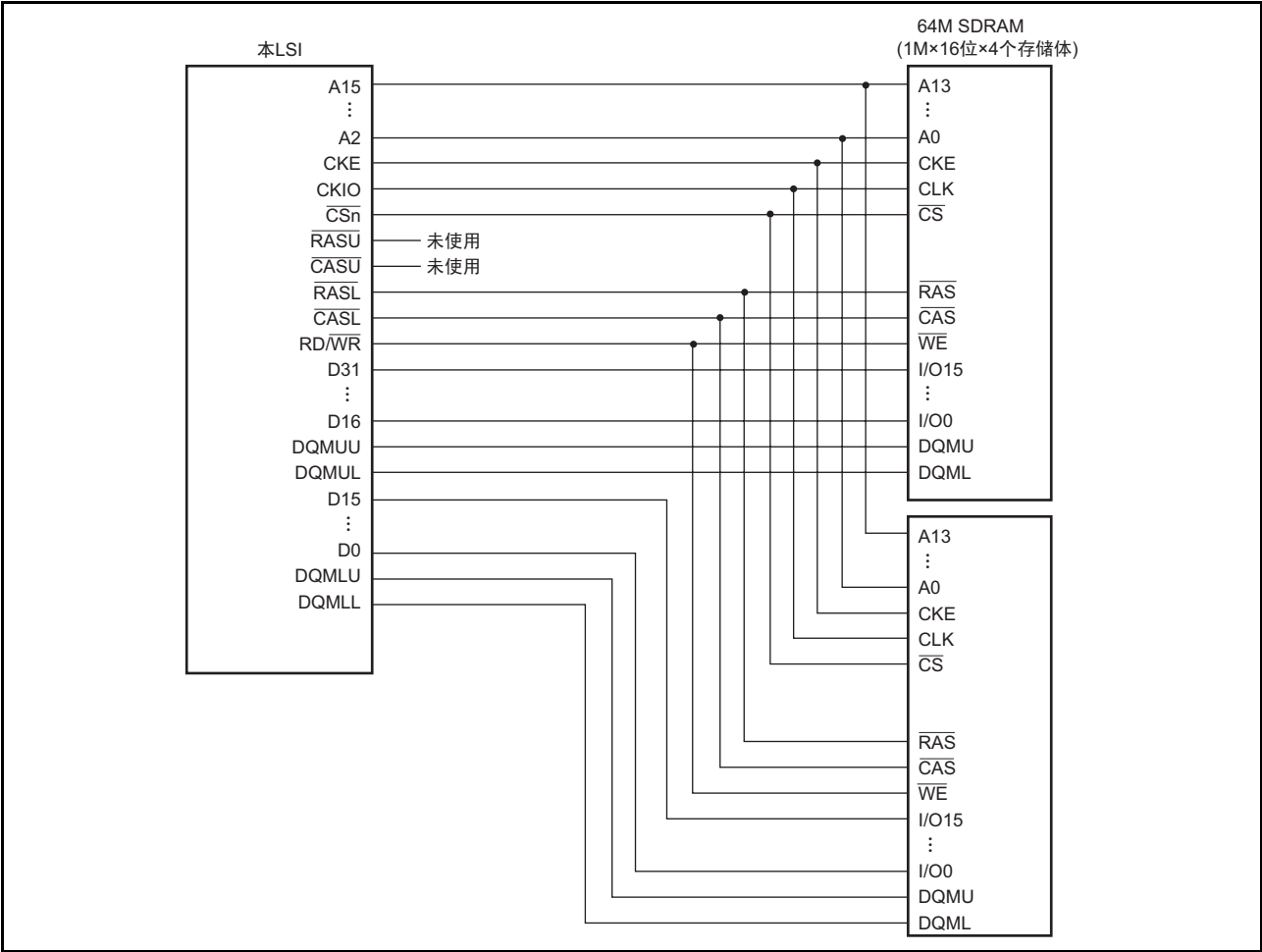


图 9.14 32 位数据宽度的 SDRAM 连接例（未使用 RASU、CASU）

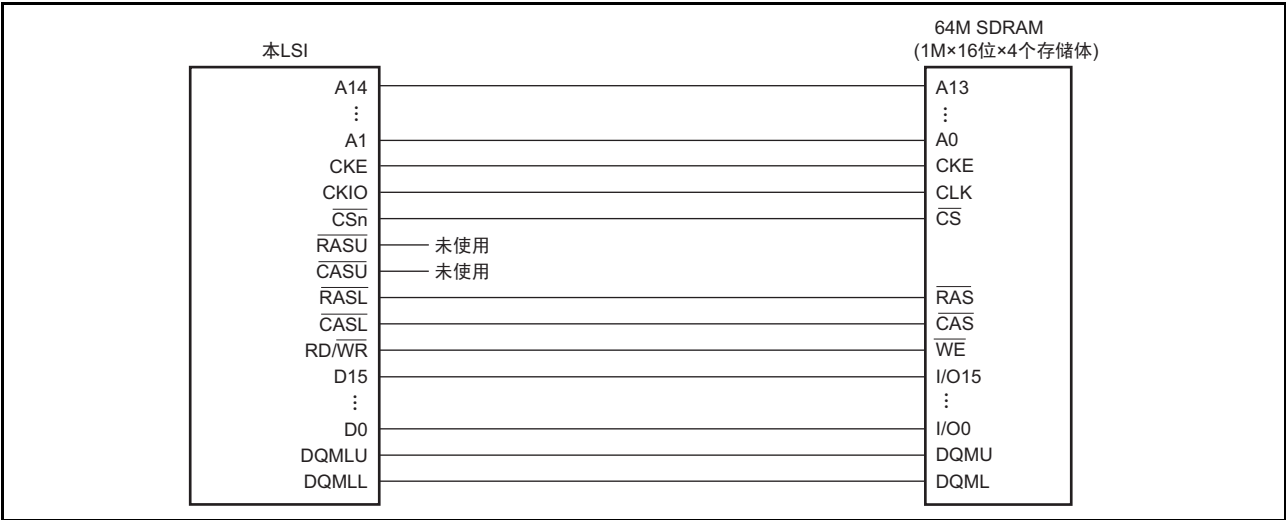
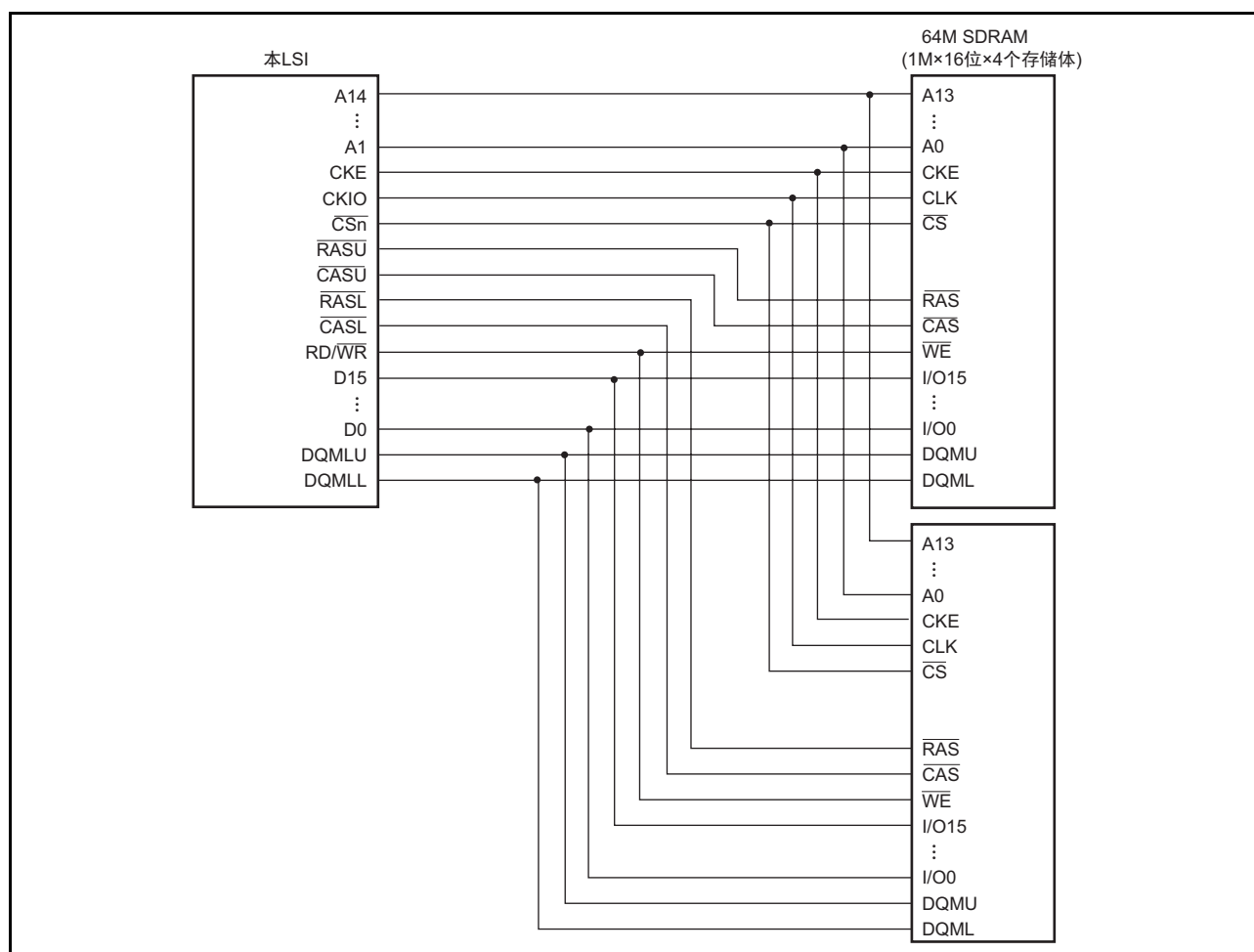


图 9.15 16 位数据宽度的 SDRAM 连接例（未使用 RASU、CASU）

图 9.16 16 位数据宽度的 SDRAM 连接例 (使用 $\overline{\text{RASU}}$ 、 $\overline{\text{CASU}}$)

(2) 地址多路复用

执行地址多路复用，以便按照 CSnBCR 的 $\text{BSZ}[1:0]$ 位、 SDCR 的 $\text{A2ROW}[1:0]$ 位、 $\text{A2COL}[1:0]$ 位、 $\text{A3ROW}[1:0]$ 位及 $\text{A3COL}[1:0]$ 位的设定，实现无外接地址多路复用电路也可连接 SDRAM 的功能。 $\text{BSZ}[1:0]$ 、 $\text{A2ROW}[1:0]$ 位、 $\text{A2COL}[1:0]$ 位、 $\text{A3ROW}[1:0]$ 位及 $\text{A3COL}[1:0]$ 位的设定与输出到地址引脚的位的关系如表 9.11 ~ 表 9.16 所示。不得执行此表之外的设定，否则无法保证运行。 $\text{A25} \sim \text{A18}$ 不执行多路复用，总是输出原地址。

数据总线宽度为 16 位时 ($\text{BSZ}[1:0]=\text{B}'10$)，SDRAM 的 A0 引脚指定字地址。因此，必须将 SDRAM 的 A0 引脚连接至本 LSI 的 A1 引脚，然后将 A1 引脚连接至 A2 引脚，之后的引脚连接以此类推。数据总线宽度为 32 位时 ($\text{BSZ}[1:0]=\text{B}'11$)，SDRAM 的 A0 引脚指定长字地址。因此必须将 SDRAM 的 A0 引脚连接至本 LSI 的 A2 引脚，然后将 A1 引脚连接至 A3 引脚，之后的引脚连接以此类推。

表 9.11 BSZ[1:0]、A2/3ROW[1:0]、A2/3COL[1:0] 与地址多路复用的关系 (1)

设 定				
BSZ[1:0]	A2/ 3ROW[1:0]	A2/ 3COL[1:0]		
11 (32 位)	00 (11 位)	00 (8 位)		
本 LSI 的 输出引脚	行地址 输出周期	列地址 输出周期	SDRAM 的引脚	功能
A17	A25	A17		未使用
A16	A24	A16		
A15	A23	A15		
A14	A22*2	A22*2	A12(BA1)	指定 存储体
A13	A21*2	A21*2	A11(BA0)	
A12	A20	L/H*1	A10/AP	指定地址 / 预充电
A11	A19	A11	A9	地址
A10	A18	A10	A8	
A9	A17	A9	A7	
A8	A16	A8	A6	
A7	A15	A7	A5	
A6	A14	A6	A4	
A5	A13	A5	A3	
A4	A12	A4	A2	
A3	A11	A3	A1	
A2	A10	A2	A0	
A1	A9	A1		未使用
A0	A8	A0		
连接存储器的例子				
1 个 64M 位产品 (512K 字 ×32 位 ×4 个存储体、列 8 位产品) 2 个 16M 位产品 (512K 字 ×16 位 ×2 个存储体、列 8 位产品)				

设 定				
BSZ[1:0]	A2/ 3ROW[1:0]	A2/ 3COL[1:0]		
11 (32 位)	01 (12 位)	00 (8 位)		
本 LSI 的 输出引脚	行地址 输出周期	列地址 输出周期	SDRAM 的引脚	功能
A17	A25	A17		未使用
A16	A24	A16		
A15	A23*2	A23*2	A13(BA1)	指定 存储体
A14	A22*2	A22*2	A12(BA0)	
A13	A21	A13	A11	地址
A12	A20	L/H*1	A10/AP	指定地址 / 预充电
A11	A19	A11	A9	地址
A10	A18	A10	A8	
A9	A17	A9	A7	
A8	A16	A8	A6	
A7	A15	A7	A5	
A6	A14	A6	A4	
A5	A13	A5	A3	
A4	A12	A4	A2	
A3	A11	A3	A1	
A2	A10	A2	A0	
A1	A9	A1		未使用
A0	A8	A0		
连接存储器的例子				
1 个 128M 位产品 (1M 字 ×32 位 ×4 个存储体、列 8 位产品) 2 个 64M 位产品 (1M 字 ×16 位 ×4 个存储体、列 8 位产品)				

【注】 *1 L/H 为用于指定命令的位，根据存取模式固定为低电平或高电平。

*2 指定存储体地址

表 9.12 BSZ[1:0]、A2/3ROW[1:0]、A2/3COL[1:0] 与地址多路复用的关系 (2)

设 定				
BSZ[1:0]	A2/ 3ROW[1:0]	A2/ 3COL[1:0]		
11 (32 位)	01 (12 位)	01 (9 位)		
本 LSI 的 输出引脚	行地址 输出周期	列地址 输出周期	SDRAM 的引脚	功能
A17	A26	A17		未使用
A16	A25	A16		
A15	A24*2	A24*2	A13(BA1)	指定 存储体
A14	A23*2	A23*2	A12(BA0)	
A13	A22	A13	A11	地址
A12	A21	L/H*1	A10/AP	指定地址 / 预充电
A11	A20	A11	A9	地址
A10	A19	A10	A8	
A9	A18	A9	A7	
A8	A17	A8	A6	
A7	A16	A7	A5	
A6	A15	A6	A4	
A5	A14	A5	A3	
A4	A13	A4	A2	
A3	A12	A3	A1	
A2	A11	A2	A0	
A1	A10	A1		未使用
A0	A9	A0		
连接存储器的例子				
1 个 256M 位产品 (2M 字 ×32 位 ×4 个存储体、列 9 位产品) 2 个 128M 位产品 (2M 字 ×16 位 ×4 个存储体、列 9 位产品)				

设 定				
BSZ[1:0]	A2/ 3ROW[1:0]	A2/ 3COL[1:0]		
11 (32 位)	01 (12 位)	10 (10 位)		
本 LSI 的 输出引脚	行地址 输出周期	列地址 输出周期	SDRAM 的引脚	功能
A17	A27	A17		未使用
A16	A26	A16		
A15	A25*2*3	A25*2*3	A13(BA1)	指定 存储体
A14	A24*2	A24*2	A12(BA0)	
A13	A23	A13	A11	地址
A12	A22	L/H*1	A10/AP	指定地址 / 预充电
A11	A21	A11	A9	地址
A10	A20	A10	A8	
A9	A19	A9	A7	
A8	A18	A8	A6	
A7	A17	A7	A5	
A6	A16	A6	A4	
A5	A15	A5	A3	
A4	A14	A4	A2	
A3	A13	A3	A1	
A2	A12	A2	A0	
A1	A11	A1		未使用
A0	A10	A0		
连接存储器的例子				
1 个 512M 位产品 (4M 字 ×32 位 ×4 个存储体、列 10 位产品) 2 个 256M 位产品 (4M 字 ×16 位 ×4 个存储体、列 10 位产品)				

【注】 *1 L/H 为用于指定命令的位，根据存取模式固定为低电平或高电平。

*2 指定存储体地址

*3 A25 指定存储体地址，因此仅 $\overline{\text{RASL}}$ 有效， $\overline{\text{RASU}}$ 无效。

表 9.13 BSZ[1:0]、A2/3ROW[1:0]、A2/3COL[1:0] 与地址多路复用的关系 (3)

设 定				
BSZ[1:0]	A2/3ROW[1:0]	A2/3COL[1:0]		
11 (32 位)	10 (13 位)	01 (9 位)		
本 LSI 的输出引脚	行地址 输出周期	列地址 输出周期	SDRAM 的引脚	功能
A17	A26	A17		未使用
A16	A25*2*3	A25*2*3	A14(BA1)	指定存储体
A15	A24*2	A24*2	A13(BA0)	
A14	A23	A14	A12	地址
A13	A22	A13	A11	
A12	A21	L/H*1	A10/AP	指定地址 / 预充电
A11	A20	A11	A9	地址
A10	A19	A10	A8	
A9	A18	A9	A7	
A8	A17	A8	A6	
A7	A16	A7	A5	
A6	A15	A6	A4	
A5	A14	A5	A3	
A4	A13	A4	A2	
A3	A12	A3	A1	
A2	A11	A2	A0	
A1	A10	A1		未使用
A0	A9	A0		
连接存储器的例子				
1 个 512M 位产品 (4M 字 ×32 位 ×4 个存储体、列 9 位产品) 2 个 256M 位产品 (4M 字 ×16 位 ×4 个存储体、列 9 位产品)				

【注】 *1 L/H 为用于指定命令的位，根据存取模式固定为低电平或者高电平。

*2 指定存储体地址

*3 A25 指定存储体地址，因此仅 $\overline{\text{RASL}}$ 有效， $\overline{\text{RASU}}$ 无效。

表 9.14 BSZ[1:0]、A2/3ROW[1:0]、A2/3COL[1:0] 与地址多路复用的关系 (4)

设 定				
BSZ[1:0]	A2/ 3ROW[1:0]	A2/ 3COL[1:0]		
10 (16 位)	00 (11 位)	00 (8 位)		
本 LSI 的 输出引脚	行地址 输出周期	列地址 输出周期	SDRAM 的引脚	功能
A17	A25	A17		未使用
A16	A24	A16		
A15	A23	A15		
A14	A22	A14		
A13	A21	A21		
A12	A20*2	A20*2	A11(BA0)	指定 存储体
A11	A19	L/H*1	A10/AP	指定地址 / 预充电
A10	A18	A10	A9	地址
A9	A17	A9	A8	
A8	A16	A8	A7	
A7	A15	A7	A6	
A6	A14	A6	A5	
A5	A13	A5	A4	
A4	A12	A4	A3	
A3	A11	A3	A2	
A2	A10	A2	A1	
A1	A9	A1	A0	
A0	A8	A0		未使用
连接存储器的例子				
1 个 16M 位产品 (512K 字 × 16 位 × 2 个存储体、列 8 位产品)				

设 定				
BSZ[1:0]	A2/ 3ROW[1:0]	A2/ 3COL[1:0]		
10 (16 位)	01 (12 位)	00 (8 位)		
本 LSI 的 输出引脚	行地址 输出周期	列地址 输出周期	SDRAM 的引脚	功能
A17	A25	A17		未使用
A16	A24	A16		
A15	A23	A15		
A14	A22*2	A22*2	A13(BA1)	指定 存储体
A13	A21*2	A21*2	A12(BA0)	
A12	A20	A12	A11	地址
A11	A19	L/H*1	A10/AP	指定地址 / 预充电
A10	A18	A10	A9	地址
A9	A17	A9	A8	
A8	A16	A8	A7	
A7	A15	A7	A6	
A6	A14	A6	A5	
A5	A13	A5	A4	
A4	A12	A4	A3	
A3	A11	A3	A2	
A2	A10	A2	A1	
A1	A9	A1	A0	
A0	A8	A0		未使用
连接存储器的例子				
1 个 64M 位产品 (1M 字 × 16 位 × 4 个存储体、列 8 位产品)				

【注】 *1 L/H 为用于指定命令的位，根据存取模式固定为低电平或高电平。

*2 指定存储体地址

表 9.15 BSZ[1:0]、A2/3ROW[1:0]、A2/3COL[1:0] 与地址多路复用的关系 (5)

设 定				
BSZ[1:0]	A2/ 3ROW[1:0]	A2/ 3COL[1:0]		
10 (16 位)	01 (12 位)	01 (9 位)		
本 LSI 的 输出引脚	行地址 输出周期	列地址 输出周期	SDRAM 的引脚	功能
A17	A26	A17		未使用
A16	A25	A16		
A15	A24	A15		
A14	A23*2	A23*2	A13(BA1)	指定 存储体
A13	A22*2	A22*2	A12(BA0)	
A12	A21	A12	A11	地址
A11	A20	L/H*1	A10/AP	指定地址 / 预充电
A10	A19	A10	A9	地址
A9	A18	A9	A8	
A8	A17	A8	A7	
A7	A16	A7	A6	
A6	A15	A6	A5	
A5	A14	A5	A4	
A4	A13	A4	A3	
A3	A12	A3	A2	
A2	A11	A2	A1	
A1	A10	A1	A0	
A0	A9	A0		未使用
连接存储器的例子				
1 个 128M 位产品 (2M 字 × 16 位 × 4 个存储体、列 9 位产品)				

设 定				
BSZ[1:0]	A2/ 3ROW[1:0]	A2/ 3COL[1:0]		
10 (16 位)	01 (12 位)	10 (10 位)		
本 LSI 的 输出引脚	行地址 输出周期	列地址 输出周期	SDRAM 的引脚	功能
A17	A27	A17		未使用
A16	A26	A16		
A15	A25	A15		
A14	A24*2	A24*2	A13(BA1)	指定 存储体
A13	A23*2	A23*2	A12(BA0)	
A12	A22	A12	A11	地址
A11	A21	L/H*1	A10/AP	指定地址 / 预充电
A10	A20	A10	A9	地址
A9	A19	A9	A8	
A8	A18	A8	A7	
A7	A17	A7	A6	
A6	A16	A6	A5	
A5	A15	A5	A4	
A4	A14	A4	A3	
A3	A13	A3	A2	
A2	A12	A2	A1	
A1	A11	A1	A0	
A0	A10	A0		未使用
连接存储器的例子				
1 个 256M 位产品 (4M 字 × 16 位 × 4 个存储体、列 10 位产品)				

【注】 *1 L/H 为用于指定命令的位，根据存取模式固定为低电平或者高电平。

*2 指定存储体地址

表 9.16 BSZ[1:0]、A2/3ROW[1:0]、A2/3COL[1:0] 与地址多路复用的关系 (6)

设 定				
BSZ[1:0]	A2/ 3ROW[1:0]	A2/ 3COL[1:0]		
10 (16 位)	10 (13 位)	01 (9 位)		
本 LSI 的 输出引脚	行地址 输出周期	列地址 输出周期	SDRAM 的引脚	功能
A17	A26	A17		未使用
A16	A25	A16		
A15	A24*2	A24*2	A14(BA1)	指定 存储体
A14	A23*2	A23*2	A13(BA0)	
A13	A22	A13	A12	地址
A12	A21	A12	A11	
A11	A20	L/H*1	A10/AP	指定地址 / 预充电
A10	A19	A10	A9	地址
A9	A18	A9	A8	
A8	A17	A8	A7	
A7	A16	A7	A6	
A6	A15	A6	A5	
A5	A14	A5	A4	
A4	A13	A4	A3	
A3	A12	A3	A2	
A2	A11	A2	A1	
A1	A10	A1	A0	
A0	A9	A0		未使用
连接存储器的例子				
1 个 256M 位产品 (4M 字 × 16 位 × 4 个存储体、列 9 位产品)				

设 定				
BSZ[1:0]	A2/ 3ROW[1:0]	A2/ 3COL[1:0]		
10 (16 位)	10 (13 位)	10 (10 位)		
本 LSI 的 输出引脚	行地址 输出周期	列地址 输出周期	SDRAM 的引脚	功能
A17	A27	A17		未使用
A16	A26	A16		
A15	A25*2*3	A25*2*3	A14(BA1)	指定 存储体
A14	A24*2	A24*2	A13(BA0)	
A13	A23	A13	A12	地址
A12	A22	A12	A11	
A11	A21	L/H*1	A10/AP	指定地址 / 预充电
A10	A20	A10	A9	地址
A9	A19	A9	A8	
A8	A18	A8	A7	
A7	A17	A7	A6	
A6	A16	A6	A5	
A5	A15	A5	A4	
A4	A14	A4	A3	
A3	A13	A3	A2	
A2	A12	A2	A1	
A1	A11	A1	A0	
A0	A10	A0		未使用
连接存储器的例子				
1 个 512M 位产品 (8M 字 × 16 位 × 4 个存储体、列 10 位产品)				

【注】 *1 L/H 为用于指定命令的位，根据存取模式固定为低电平或高电平。

*2 指定存储体地址

*3 A25 指定存储体地址，因此仅 $\overline{\text{RASL}}$ 有效， $\overline{\text{RASU}}$ 无效。

(3) 突发读取

在本 LSI 产生突发读取的条件如下：

1. 读取的存取长度大于数据总线宽度时
2. 高速缓存未命中时的 16 字节传送时
3. 由 DMAC 传送 16 字节时
4. LCDC 传送 16 字节～128 字节时*

本 LSI 总是以突发长度 1 存取 SDRAM。例如，从连接 32 位数据总线的 SDRAM 连续读取 16 字节的数据时，连续执行 4 次突发长度为 1 的读取。此时的存取称为突发数为 4 的突发读取。存取长度与突发数的关系如表 9.17 所示。

【注】 * 详情参阅“第 24 章 LCD 控制器 (LCDC)”。

表 9.17 存取长度与突发数的关系

总线宽度	存取长度	突发数
16 位	8 位	1
	16 位	1
	32 位	2
	16 字节	8
	32 字节 *	16
	64 字节 *	32
	128 字节 *	64
32 位	8 位	1
	16 位	1
	32 位	1
	16 字节	4
	32 字节 *	8
	64 字节 *	16
	128 字节 *	32

【注】 * 使用 LCDC 时，产生 32、64、128 字节存取。
详情参阅“第 24 章 LCD 控制器 (LCDC)”。

突发读取的时序图如图 9.17 和图 9.18 所示。突发读取时，在 T_r 周期输出 ACTV 命令后，接着在 T_{c1} 、 T_{c2} 、 T_{c3} 周期发行 READ 命令，在 T_{c4} 周期发行 READA 命令，并在 $T_{d1} \sim T_{d4}$ 周期，在外部时钟 (CKIO) 的上升沿接收读取数据。 T_{ap} 周期是在 SDRAM 内部等待 READA 命令引起的自动预充电结束的周期，在此期间不可对相同的存储体发行新命令。但是，可存取不同的 CS 空间或相同 SDRAM 的不同存储体。通过指定 CS3WCR 的 WTRP[1:0] 位，确定 T_{ap} 的周期数。

为了用各种频率连接 SDRAM，本 LSI 可通过设定 CS3WCR 的各位来插入等待周期。各等待周期的设定例如图 9.18 所示。通过 CS3WCR 的 WTRCD[1:0] 位，可指定从 ACTV 命令输出周期 T_r 到 READ 命令输出周期 T_{c1} 之间的周期数。如果 WTRCD[1:0] 设定为大于等于 1 个周期时，则在 T_r 周期与 T_{c1} 周期之间插入发行 NOP 命令的 T_{rw} 周期。通过 CS2WCR 的 A2CL[1:0] 位及 CS3WCR 的 A3CL[1:0] 位，可在 CS2 和与 CS3 空间独立指定从 READ 命令输出周期 T_{c1} 到读取数据取入周期 T_{d1} 之间的周期数。此周期数相当于 SDRAM 的 CAS 等待时间。SDRAM 的 CAS 等待时间最多为 3 个周期，但本 LSI 可设定为 1 ~ 4 个周期，其目的是为了在本 LSI 与 SDRAM 之间连接包含锁存器的电路。

T_{de} 周期是向本 LSI 内部传送读取数据时所需的空闲周期。突发读取或单次读取时总是产生 1 个 T_{de} 周期。

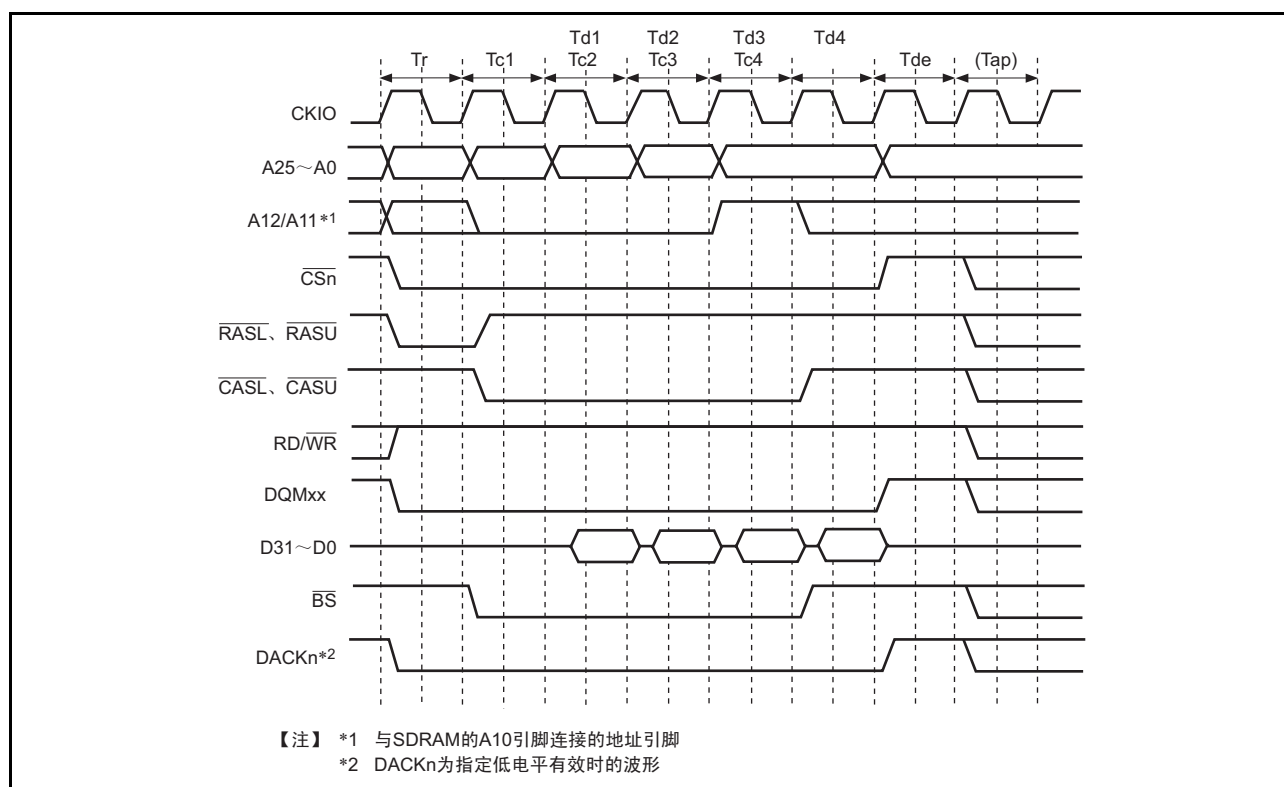


图 9.17 突发读取的基本时序 (1 个 CAS 等待时间、自动预充电)

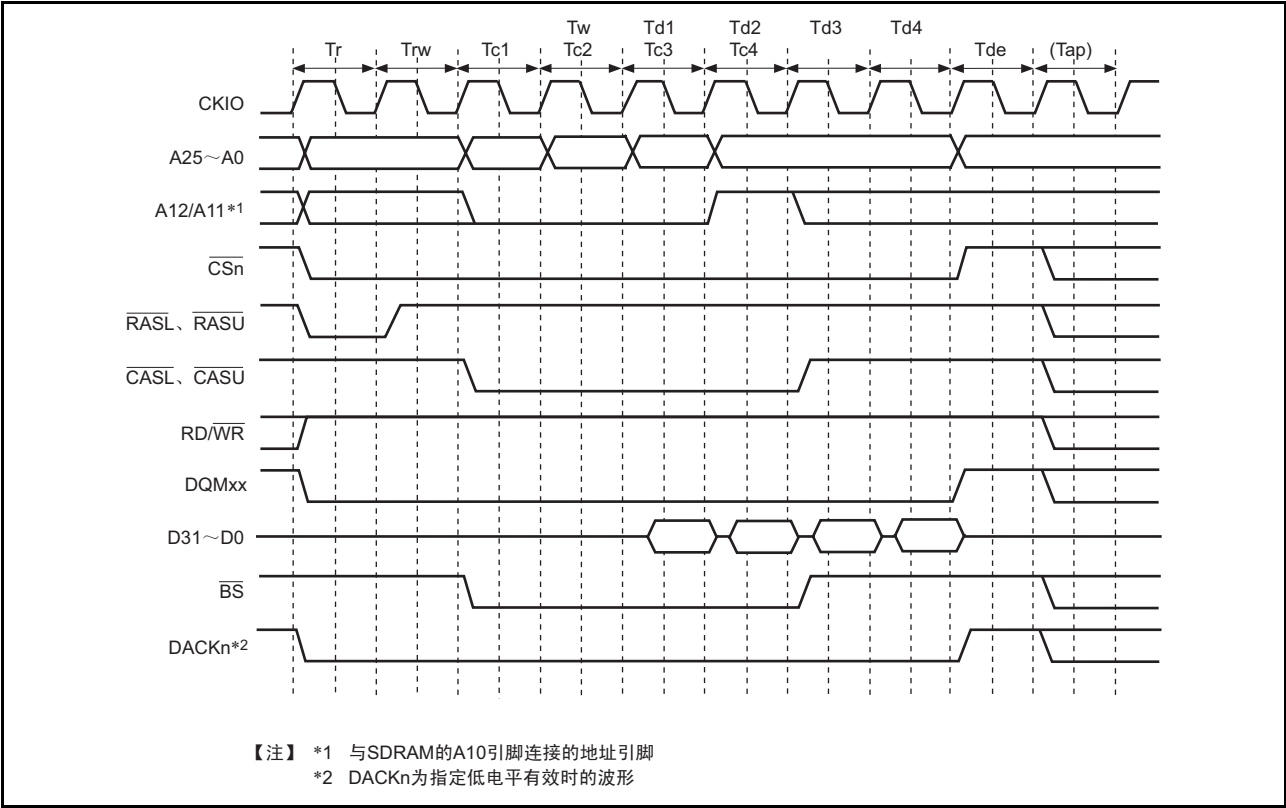


图 9.18 突发读取等待的指定时序
(2 个 CAS 等待时间、WTRCD[1:0]=1 周期、自动预充电)

(4) 单次读取

在高速缓存无效空间且数据总线宽度大于等于存取长度时，读取存取执行 1 次即结束。SDRAM 设定为突发长度为 1 的突发读取，因此仅输出所需的数据。执行 1 次即结束的读取存取称为单次读取。

单次读取的基本时序图如图 9.19 所示。

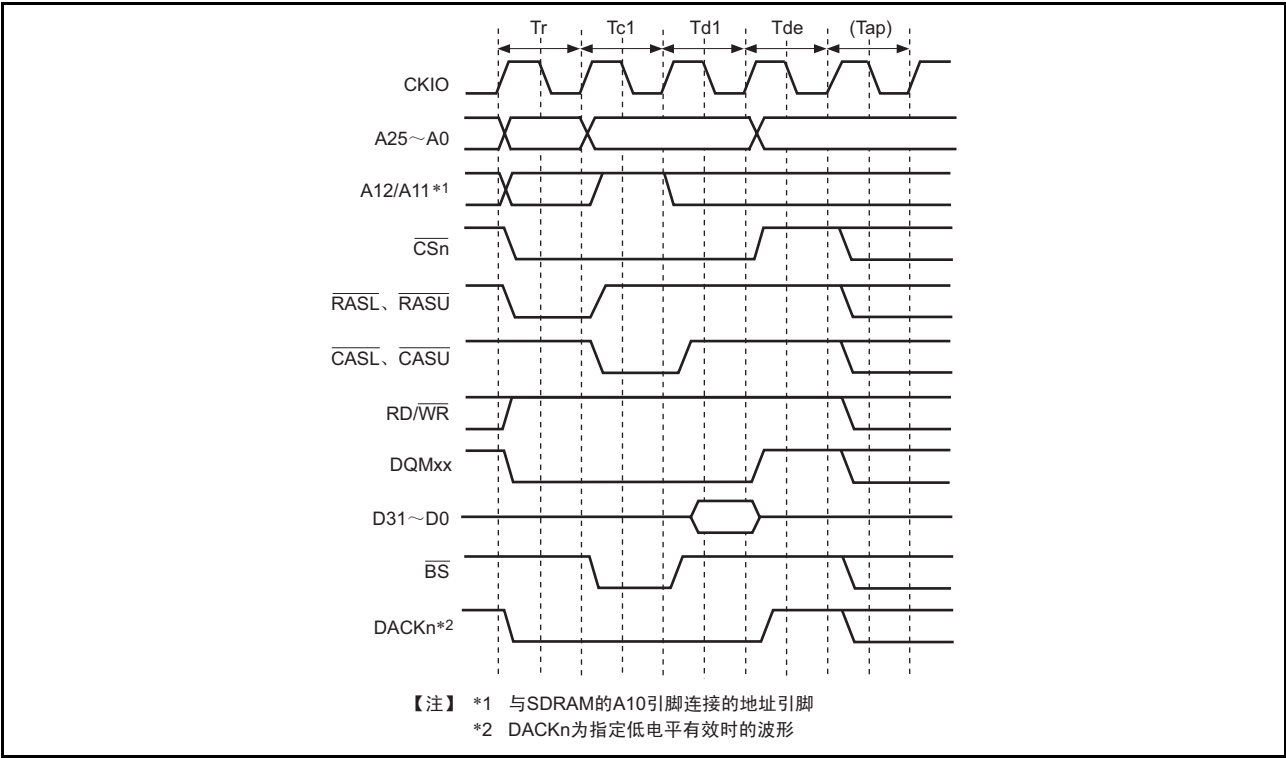


图 9.19 单次读取的基本时序（1 个 CAS 等待时间、自动预充电）

(5) 突发写入

- 本 LSI 产生突发写入的条件如下：
- 1. 写入的存取长度大于数据总线宽度时
 - 2. 产生高速缓存的回写时
 - 3. 由DMAC传送16字节时

本 LSI 总是以突发长度 1 存取 SDRAM。例如，从连接 32 位数据总线的 SDRAM 连续写入 16 字节的数据时，连续执行 4 次突发长度为 1 的写入。此时的存取称为突发数为 4 的突发写入。存取长度与突发数的关系如表 9.17 所示。突发写入的时序图如图 9.20 所示。突发写入时，在 T_r 周期输出 ACTV 命令后，接着在 T_{c1} 、 T_{c2} 、 T_{c3} 周期发行 WRIT 命令，在 T_{c4} 周期发行自动预充电的 WRITA 命令。写入周期时，写入数据与写入命令同时输出。带自动预充电的写入命令输出后，接着是等待启动自动预充电的 T_{rw1} 周期与等待自动预充电结束的 T_{ap} 周期。 T_{ap} 周期是在 SDRAM 内部等待 WRITA 命令引起的自动预充电结束的周期。在 T_{rw1} 周期及 T_{ap} 周期之间，不可对相同的存储体发行新命令。但是，可存取不同的 CS 空间或相同 SDRAM 的不同存储体。通过指定 CS3WCR 的 $TRWL[1:0]$ 位决定 T_{rw1} 周期；通过指定 CS3WCR 的 $WTRP[1:0]$ 位决定 T_{ap} 周期。

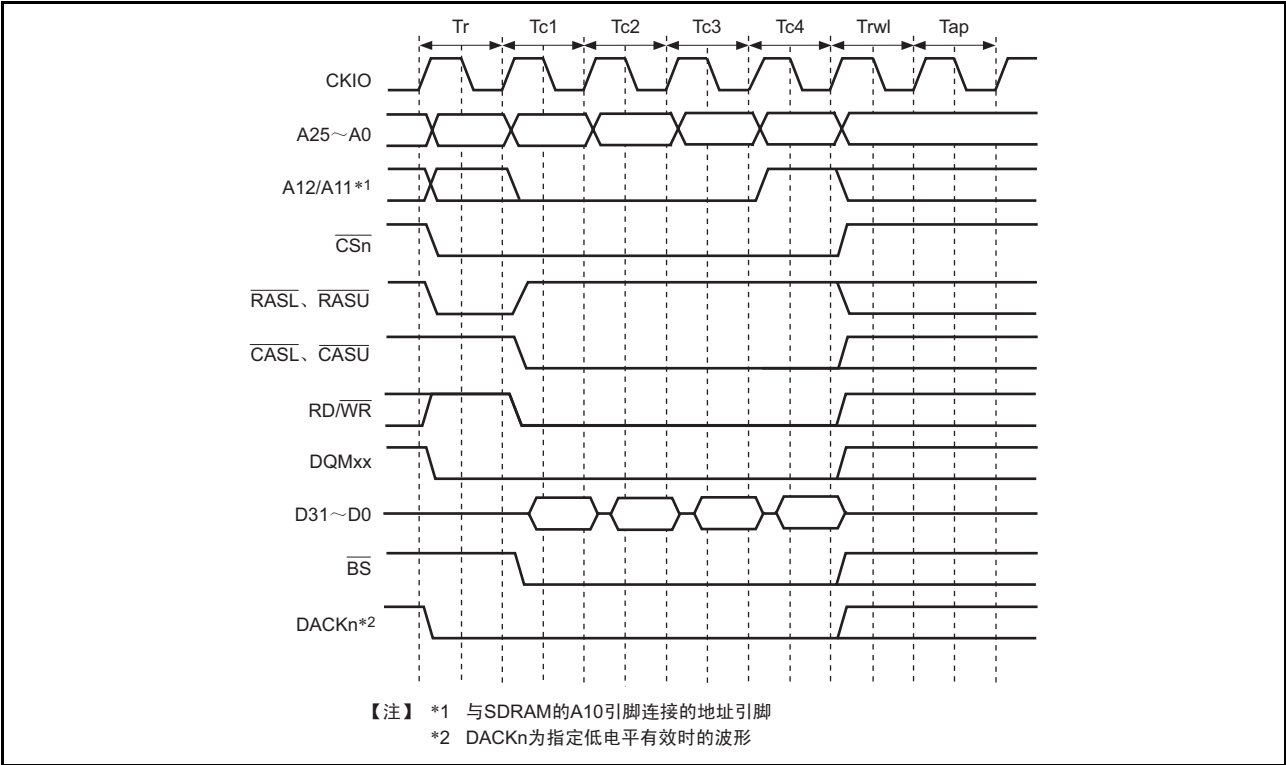


图 9.20 突发写入的基本时序（自动预充电）

(6) 单次写入

在高速缓存无效空间且数据总线宽度大于等于存取长度时，写入存取执行 1 次即结束。SDRAM 设定为单次写入或突发长度为 1 的突发写入，因此仅写入所需的数据。执行 1 次即结束的写入存取称为单次写入，单次写入的基本时序图如图 9.21 所示。

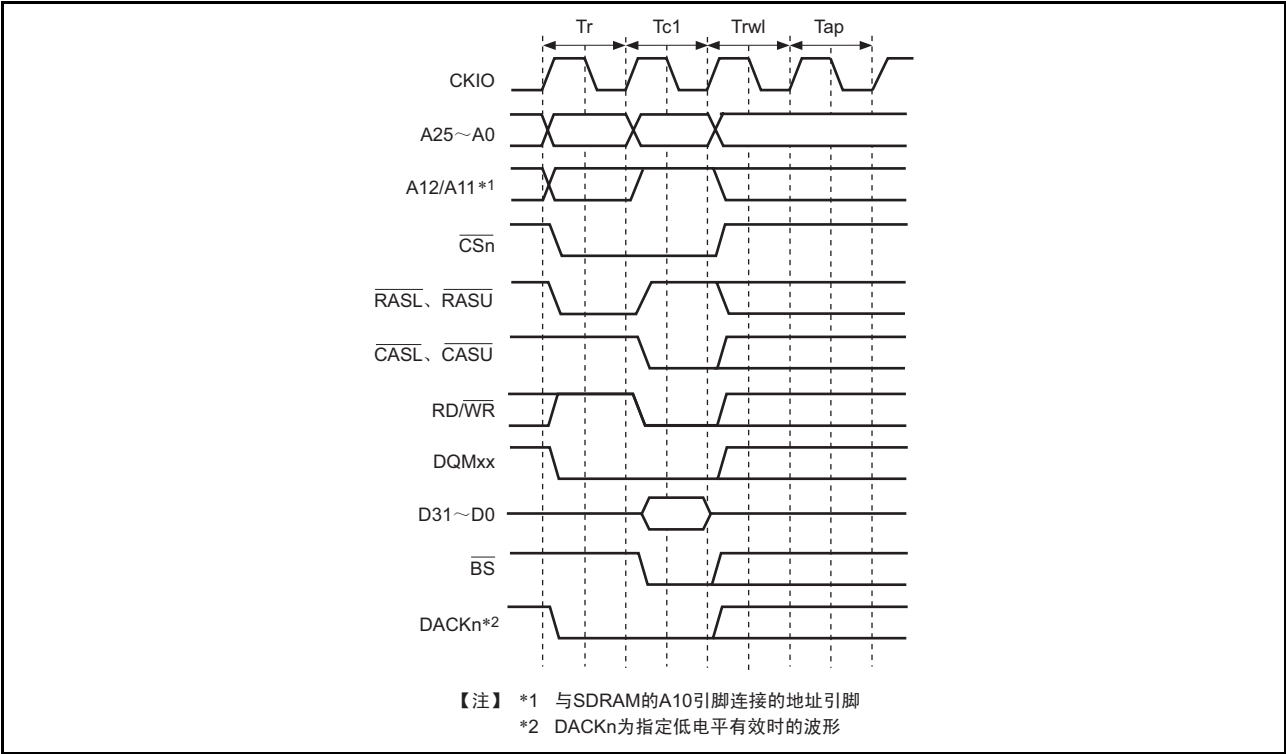


图 9.21 单次写入的基本时序（自动预充电）

(7) 存储体激活

为了高速存取相同的行地址，可使用 SDRAM 的存储体功能。SDCR 的 BACTV 位为 1 时，使用无自动预充电的命令（READ 或 WRIT）执行存取。这称作存储体激活功能。但存储体激活功能仅对区域 3 有效。将区域 3 设定为存储体激活模式时，区域 2 空间必须设定为普通空间或带字节选择的 SRAM。区域 2 及区域 3 都设为 SDRAM 时，必须为自动预充电模式。

使用存储体激活功能时，即使存取结束，也不执行预充电。存取相同存储体的相同行地址时，可不发行 ACTV 命令，而直接发行 READ 命令或 WRIT 命令。SDRAM 内部被分为多个存储体，因此各存储体的每个行地址均可设置为有效状态。下一个存取是对不同的行地址执行存取时，首先发行 PRE 命令然后对该存储体预充电，预充电完成后，再按顺序发行 ACTV 命令、READ 命令或 WRIT 命令。继续存取不同的行地址时，因产生存取请求后才执行预充电，反而可能延长存取时间。通过 CS3WCR 的 WTRP[1:0] 位，指定从发行 PRE 命令到发行 ACTV 命令的周期数。

写入时，如果自动预充电，则在发行 WRITA 命令后的 Trwl+Tap 周期，不可对相同存储体发行命令。如果使用存储体激活模式，则在相同的行地址可连续发行 READ 命令或 WRIT 命令。每写入 1 次仅可缩短 Trwl+Tap 周期数。

各存储体处于激活状态的时间（tRAS）有限制。根据程序的执行状况，如果在规定周期无法保证存取不同的行地址，则需设定刷新周期小于等于 tRAS。

无自动预充电的突发读取周期、相同行地址的突发读取周期及不同行地址的突发读取周期，分别如图 9.22、图 9.23 及图 9.24 所示。无自动预充电的单次写入周期、相同行地址的单次写入周期及不同行地址的单次写入周期，分别如图 9.25、图 9.26 及图 9.27 所示。

在图 9.23，为了从 SDRAM 读取数据时，使指定读取字节的 DQMxx 信号保持 2 个周期的等待时间，在发行 READ 命令的 Tc 周期前，插入不执行任何操作的 Tnop 周期。CAS 等待时间大于等于 2 个周期时，即使 Tc 周期后 DQMxx 信号有效，也可保证 2 个周期的等待时间，因此不插入 Tnop 周期。

仅从存取设定为存储体激活功能空间的各存储体来看，只要连续存取相同的行地址，就从图 9.22 或图 9.25 所示的周期开始，并重复图 9.23 或图 9.26 所示的周期。在此期间，即使存取不同空间或不同存储体，也不受影响。在存储体激活模式存取其他行地址时，执行图 9.24 或图 9.27 的总线周期，而不执行图 9.23 或图 9.26 的总线周期。即使在存储体激活模式下，刷新周期或通过总线仲裁释放总线后，所有存储体也变为非激活状态。

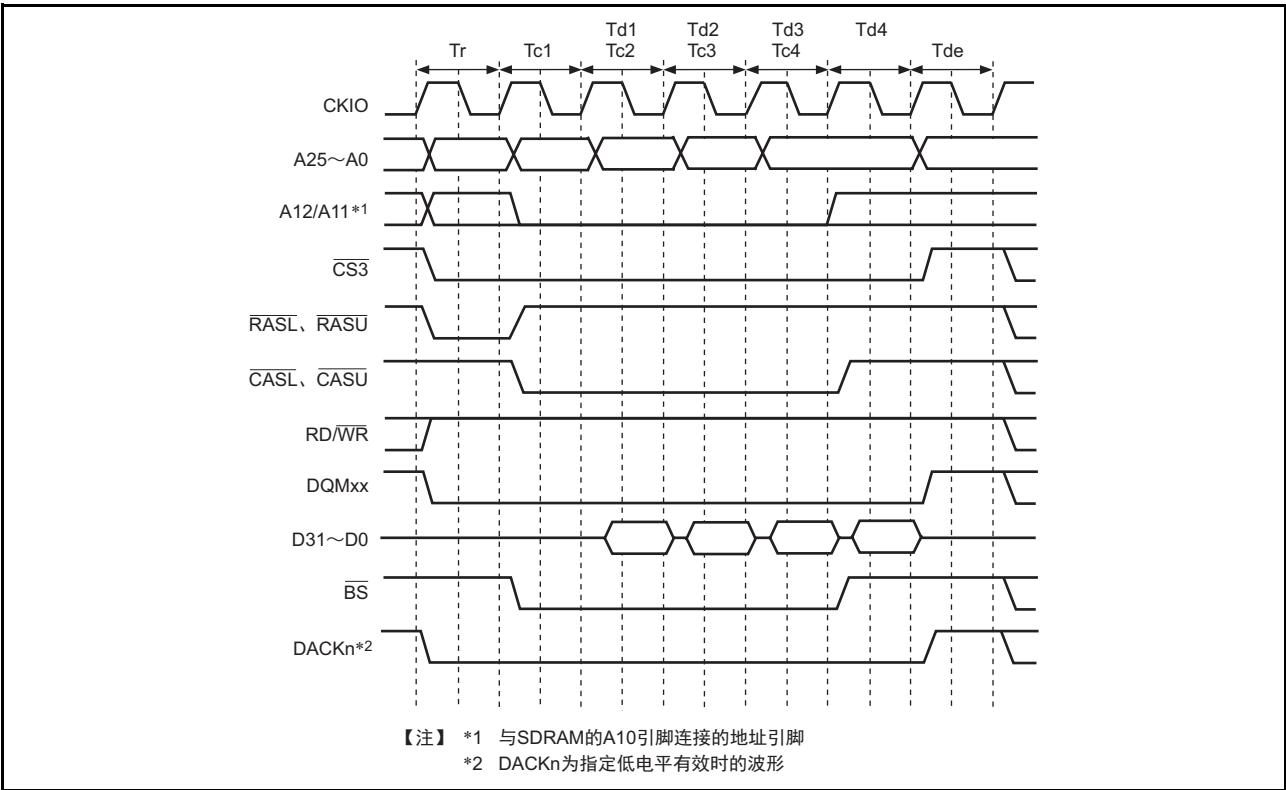


图 9.22 突发读取的时序（存储体激活、不同存储体、1 个 CAS 等待时间）

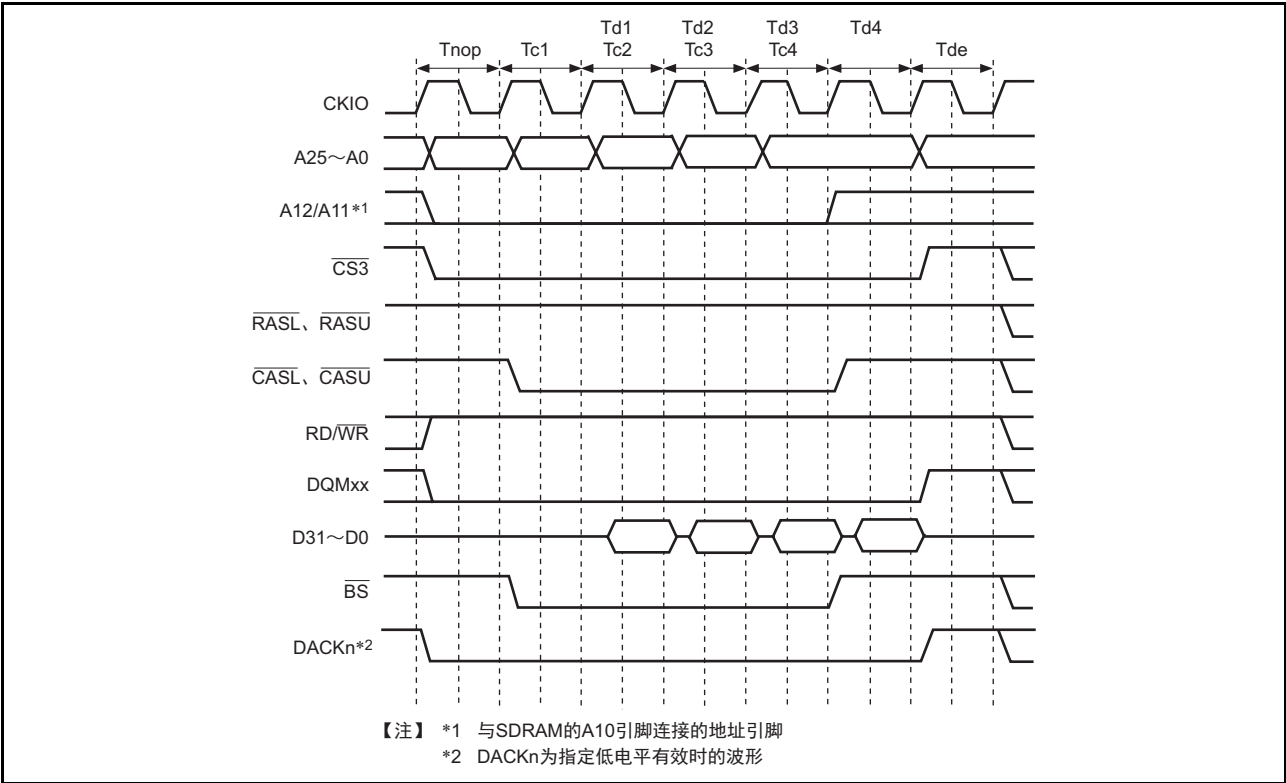


图 9.23 突发读取的时序
（存储体激活、相同存储体的相同行地址、1 个 CAS 等待时间）

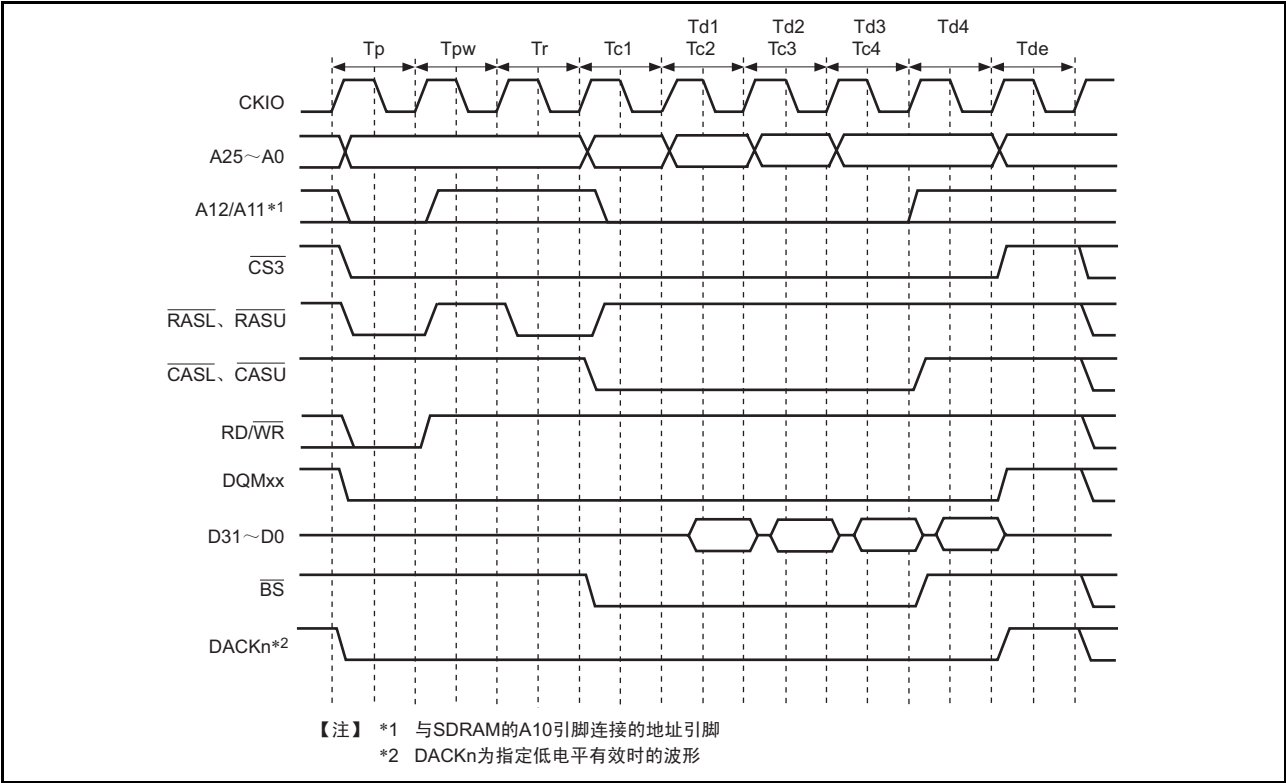


图 9.24 突发读取的时序
(存储体激活、相同存储体的不同行地址、1 个 CAS 等待时间)

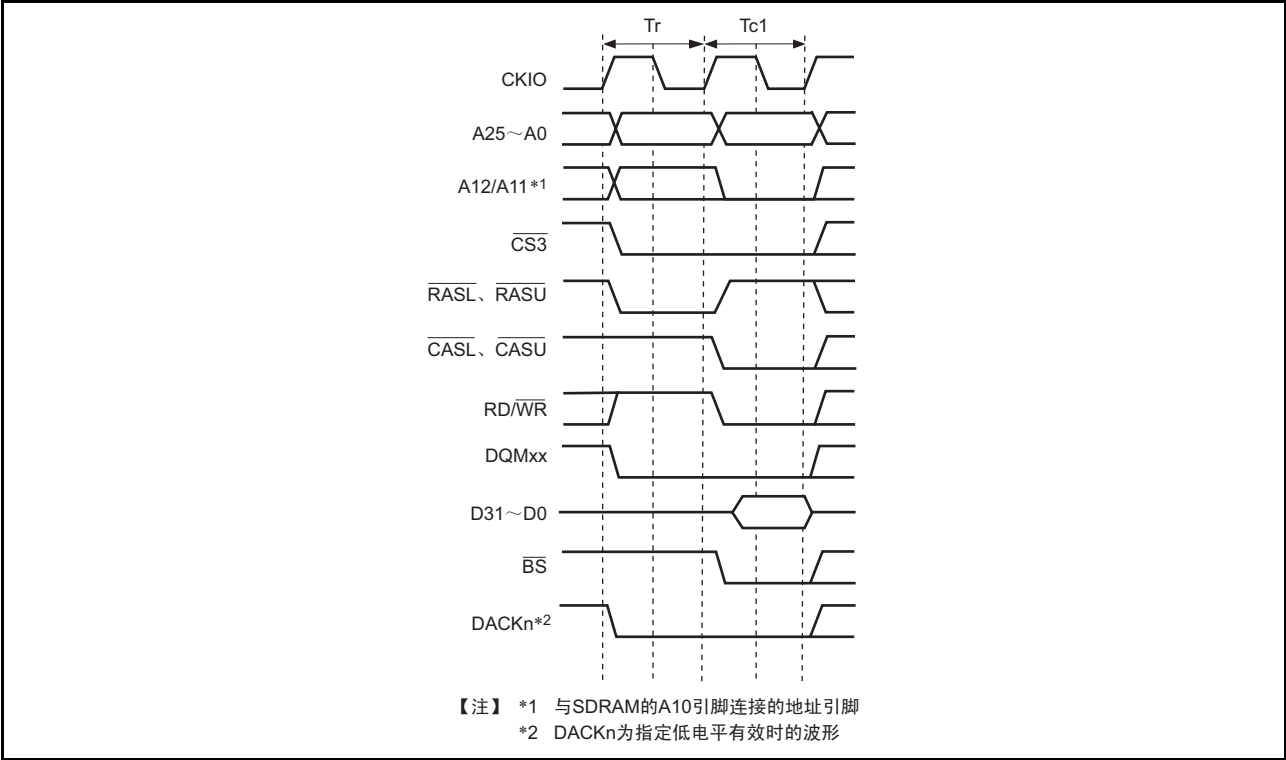


图 9.25 单次写入的时序 (存储体激活、不同存储体)

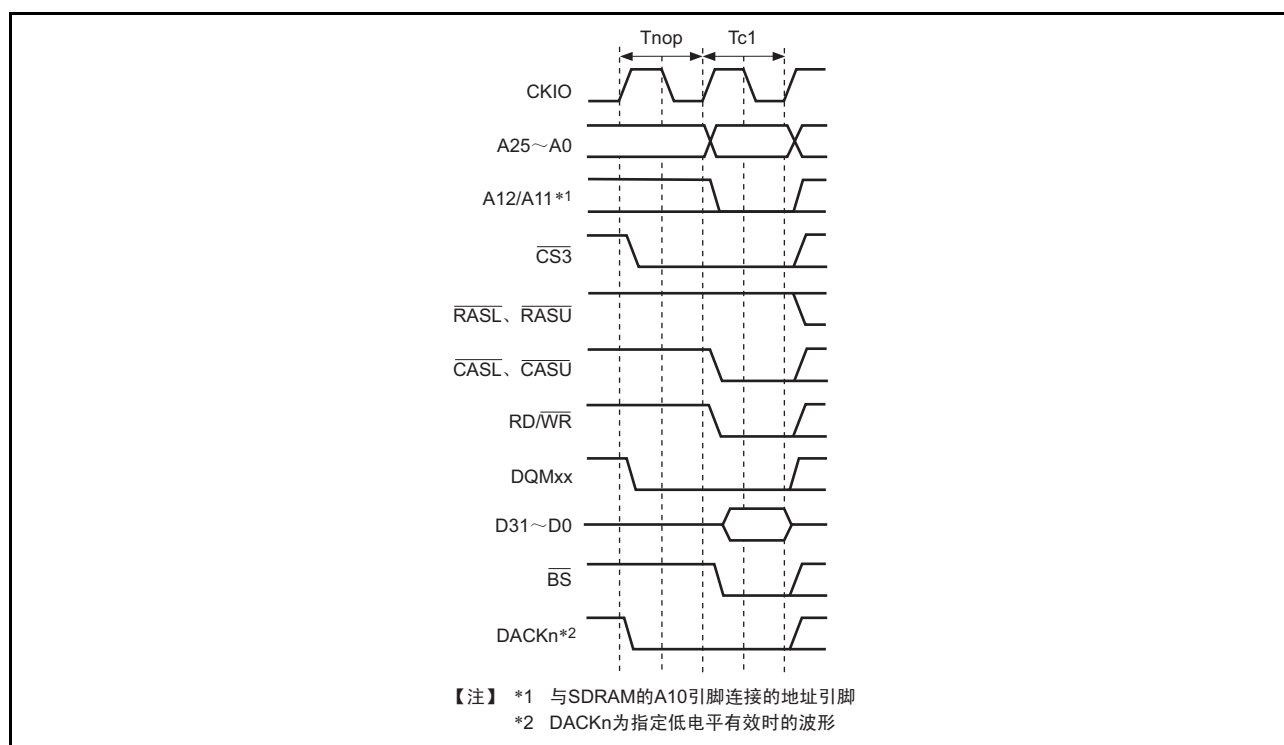


图 9.26 单次写入的时序（存储体激活、相同存储体的相同行地址）

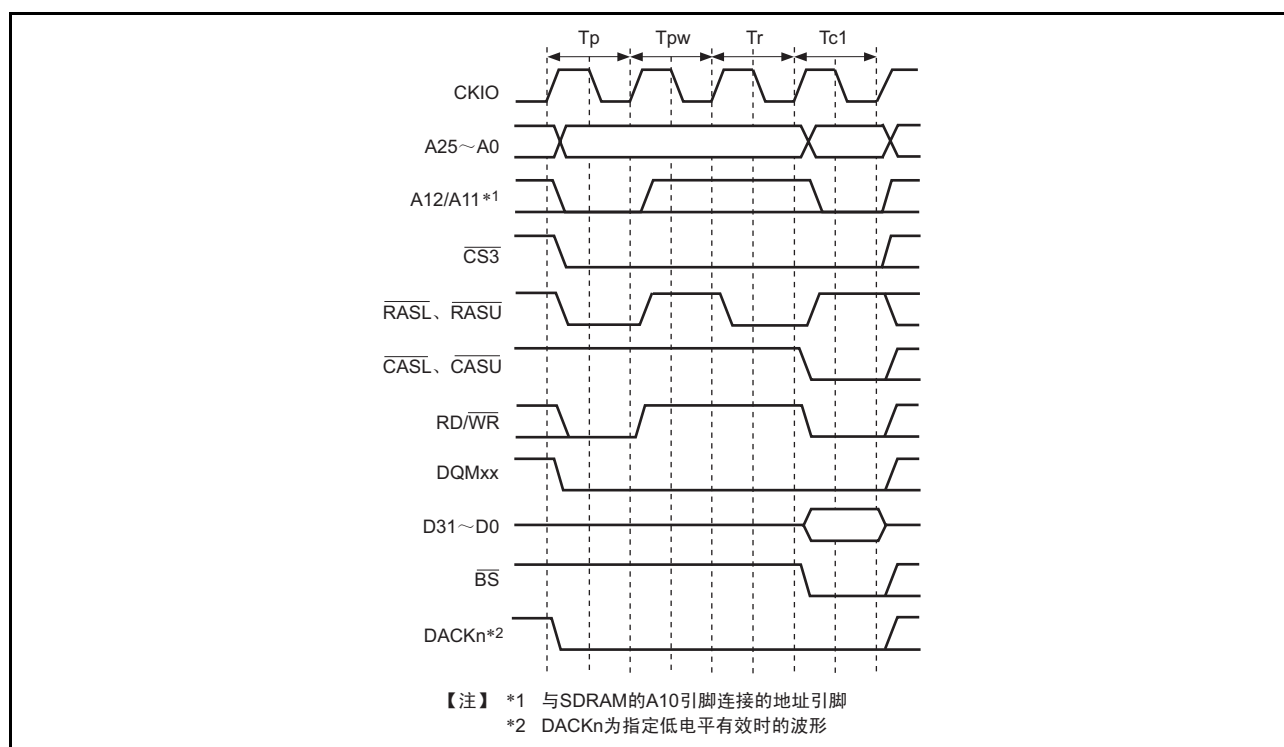


图 9.27 单次写入的时序（存储体激活、相同存储体的不同行地址）

(8) 刷新

BSC 具有控制 SDRAM 刷新的功能。通过将 SDCR 的 RFSH 位设定为 1、RMODE 位设定为 0，可执行自动刷新；通过设定 RTCSR 的 RRC[2:0] 位，可产生连续刷新。另外，长时间不存取 SDRAM 时，可同时将 RFSH 位与 RMODE 位设置为 1，启动低功耗的自刷新。

(a) 自动刷新

在 RTCSR 的 CKS[2:0] 位选择的输入时钟与 RTCOR 设定值确定的间隔，执行 RTCSR 的 RRC[2:0] 位设定次数的刷新。为了满足所使用 SDRAM 的刷新间隔规定，必须设定各寄存器。首先设定 RTCOR、RTCNT、SDCR 的 RFSH 位及 RMODE 位，接着设定 RTCSR 的 CKS[2:0] 位及 RRC[2:0] 位。由 CKS[2:0] 位选择输入时钟时，RTCNT 从当时的值开始递增计数。RTCNT 的值总是与 RTCOR 的值比较，如果两者的值匹配，则产生刷新请求并执行 RRC[2:0] 位设定次数的自动刷新。同时，将 RTCNT 清 0，重新开始递增计数。

自动刷新周期的时序如图 9.28 所示。启动自动刷新时，如果有正在预充电的存储体，则等待完成后，在 T_p 周期发行 PALL 命令，使全部的存储体从激活

状态进入预充电状态。接着，在插入 CS3WCR 的 WTRP[1:0] 位设定的空闲周期数后，在 T_{rr} 周期发行 REF 命令。在 T_{rr} 周期后与 CS3WCR 的 WTRC[1:0] 位指定的周期数之间，不发行新命令。为了满足 SDRAM 刷新周期时间的规定 (t_{RC})，需设定 WTRC[1:0] 位。CS3WCR 的 WTRP[1:0] 位的设定值大于等于 1 个周期时，在 T_p 周期与 T_{rr} 周期之间插入空闲周期。

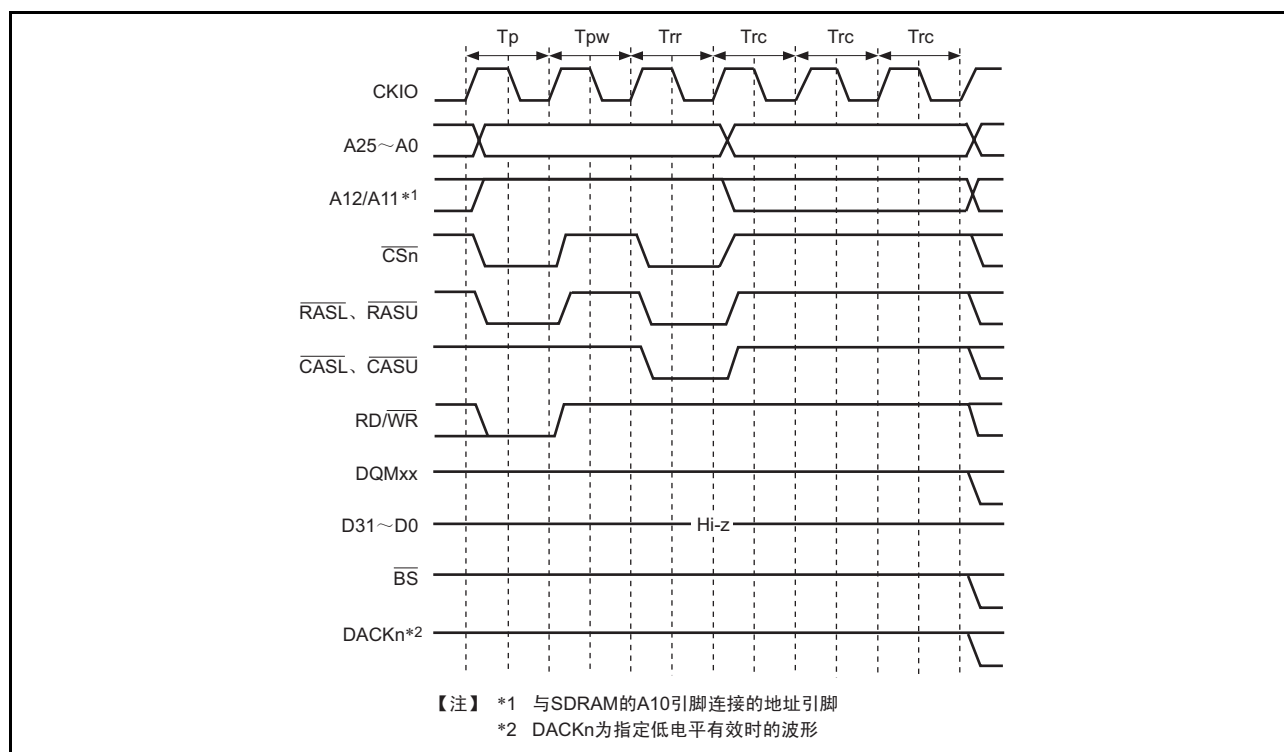


图 9.28 自动刷新时序

(b) 自刷新

自刷新是在 SDRAM 内部生成刷新时序与刷新地址的一种待机模式，通过将 SDCR 的 RFSH 位与 RMODE 位同时设置为 1 来启动。启动自刷新时，如果有正在预充电的存储体，则等待完成后，在 T_p 周期发行 PALL 命令。接着，在插入 CS3WCR 的 WTRP[1:0] 位设定的空闲周期后发行 SELF 命令。自刷新状态期间，不可存取 SDRAM。通过将 RMODE 位设定为 0，可解除自刷新。自刷新解除后，在 CS3WCR 的 WTRC[1:0] 位指定的周期数间，不发行命令。

自刷新的时序如图 9.29 所示。解除自刷新后，必须立即设定能用正确的间隔执行自动刷新。从设定为自动刷新的状态进入自刷新状态时，如果解除自刷新时设定 RFSH=1、RMODE=0，则重新开始自动刷新。如果从解除自刷新到开始自动刷新需一定时间，则可通过将（RTCOR 的值－1）设定在 RTCNT，立即开始自动刷新。

设定为自刷新后，本 LSI 设置为待机状态时仍继续保持自刷新状态；由中断从待机状态返回后也保持自刷新状态。但将 CMNCR 寄存器的 HIZCNT 位设定为 1 时，即使在待机状态，也需驱动 CKE 等引脚。

更改 PLL 电路的倍频比时，CKIO 输出不稳定，或为低电平。有关 CKIO 输出，详情参阅“第 4 章 时钟脉冲振荡器（CPG）”。更改倍频比前，通过设置为自刷新状态可保持 SDRAM 的内容。

即使手动复位，也不能解除自刷新状态。

上电复位时，BSC 的寄存器被初始化，因此自刷新状态被解除。

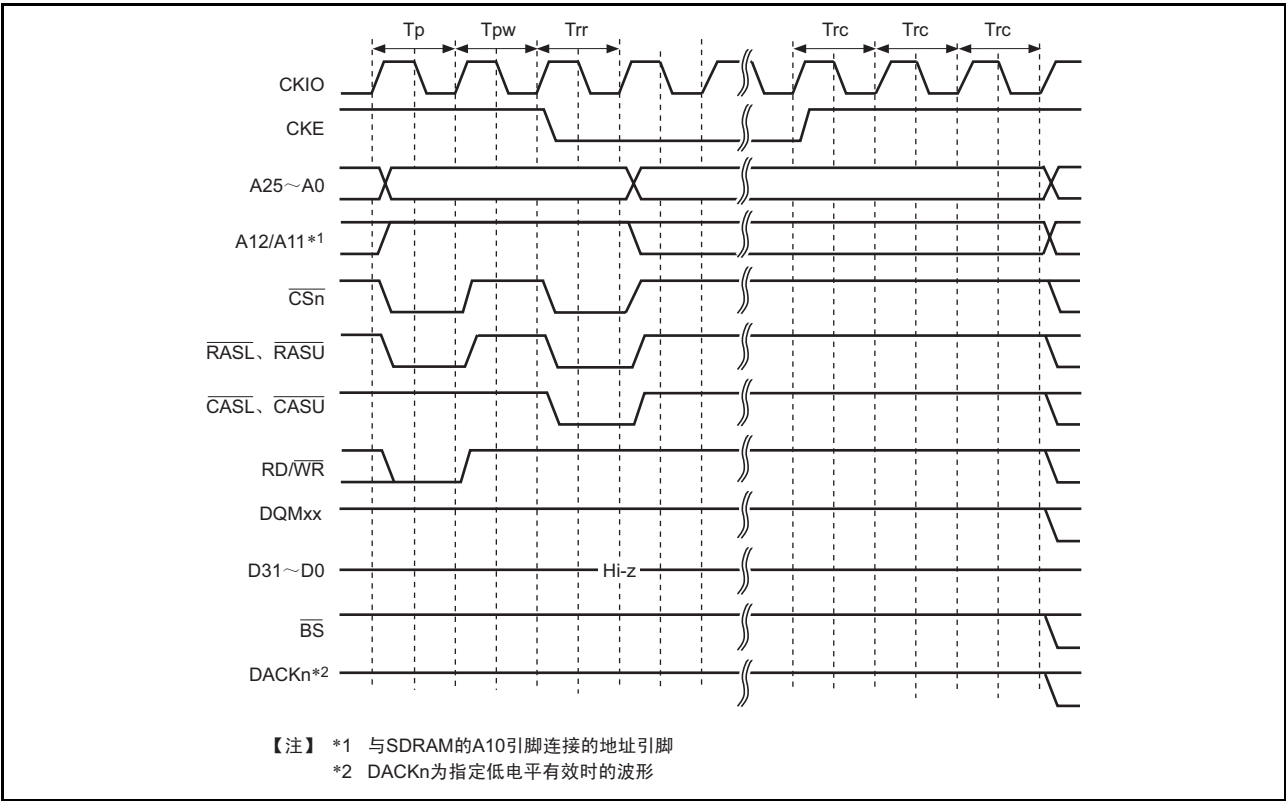


图 9.29 自刷新时序

(9) 刷新请求与总线周期的关系

在执行总线周期的过程中产生刷新请求时，等待总线周期结束后执行刷新。另外，如果在通过总线仲裁功能释放总线时产生刷新请求，则等待获取总线权后执行刷新。本 LSI 在等待执行刷新期间，作为总线权请求信号设置 **REFOUT** 引脚。有关 **REFOUT** 引脚的选择，详情参阅“第 25 章 引脚功能控制器 (PFC)”。在获得总线权之前，**REFOUT** 一直保持低电平有效。

REFOUT 有效时，外部器件必须使 **BREQ** 无效并返回总线权。注意：如果超过规定的刷新间隔时间，外部器件仍未返回总线权，则不可刷新并无法保证 SDRAM 的内容。

在等待执行刷新的状态产生新的刷新请求时，取消之前的刷新请求。为了正确执行刷新，要求总线周期大于刷新间隔时间或不占用总线权。

自刷新过程中，即使产生总线权请求，在解除自刷新前，也不会释放总线。

(10) 低频模式

如果 SDCR 的 **SLOW** 位设定为 1，则在适合低频运行 SDRAM 的时序输出命令、地址、写入数据及输入读取数据。

低频模式的存取时序如图 9.30 所示。在比通常晚半个周期的 **CKIO** 下降沿同步输出命令、地址、写入数据。另外，在比通常早半个周期的 **CKIO** 下降沿，输入读取数据。由此，可延长命令、地址、写入数据及读取数据的保持时间。

如果将 **SLOW** 位设定为 1，并以高频运行 SDRAM，则可能无法确保命令、地址、写入数据及读取数据的准备时间。必须充分考虑工作频率及电路板的时序设计后，确定 **SLOW** 位的设定。

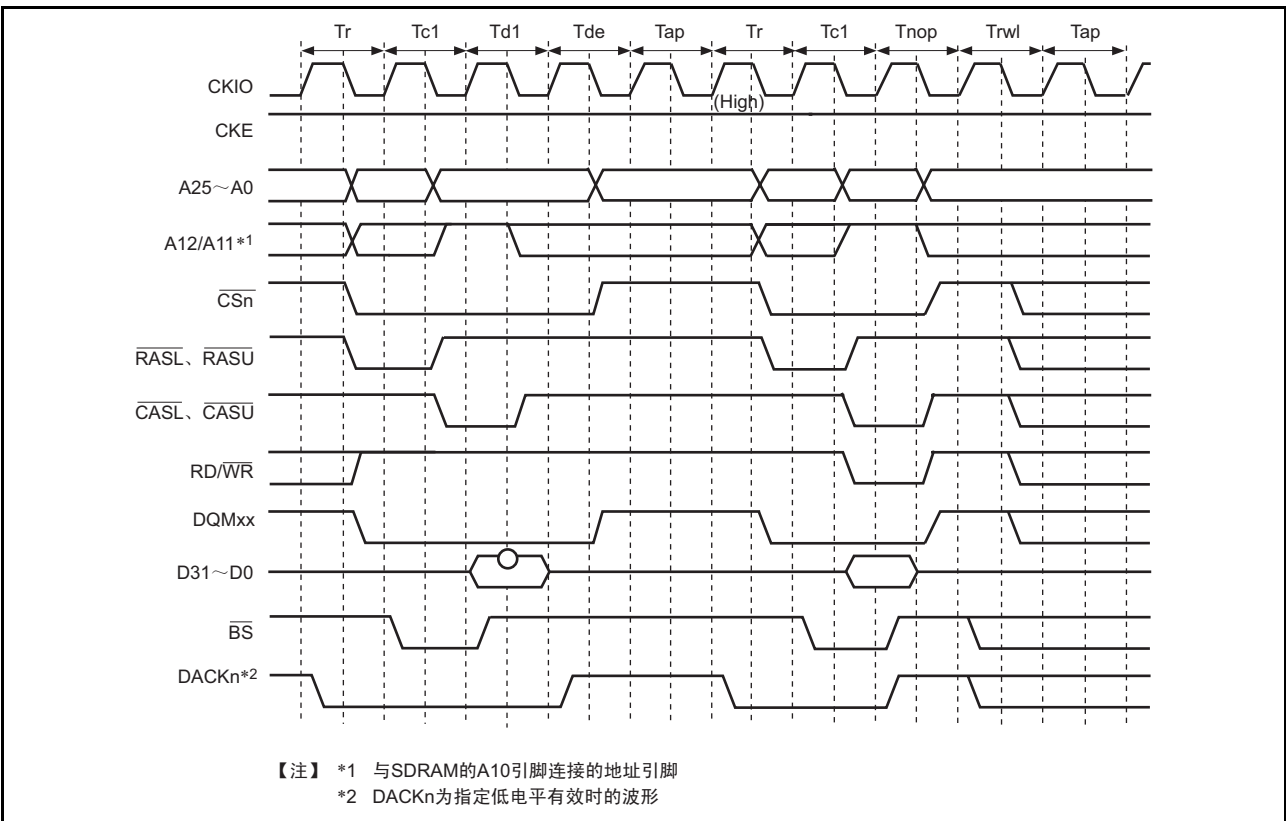


图 9.30 低频模式的存取时序

(11) 掉电模式

如果 SDCR 的 PDOWN 位设定为 1，则非存取时，可将 CKE 设置为低电平，并将 SDRAM 转移至掉电模式。由此，可大幅降低非存取时的功耗。但产生存取时，为了解除 SDRAM 的掉电模式，需插入使 CKE 有效的周期，因此会产生 1 个周期的开销。掉电模式的存取时序如图 9.31 所示。

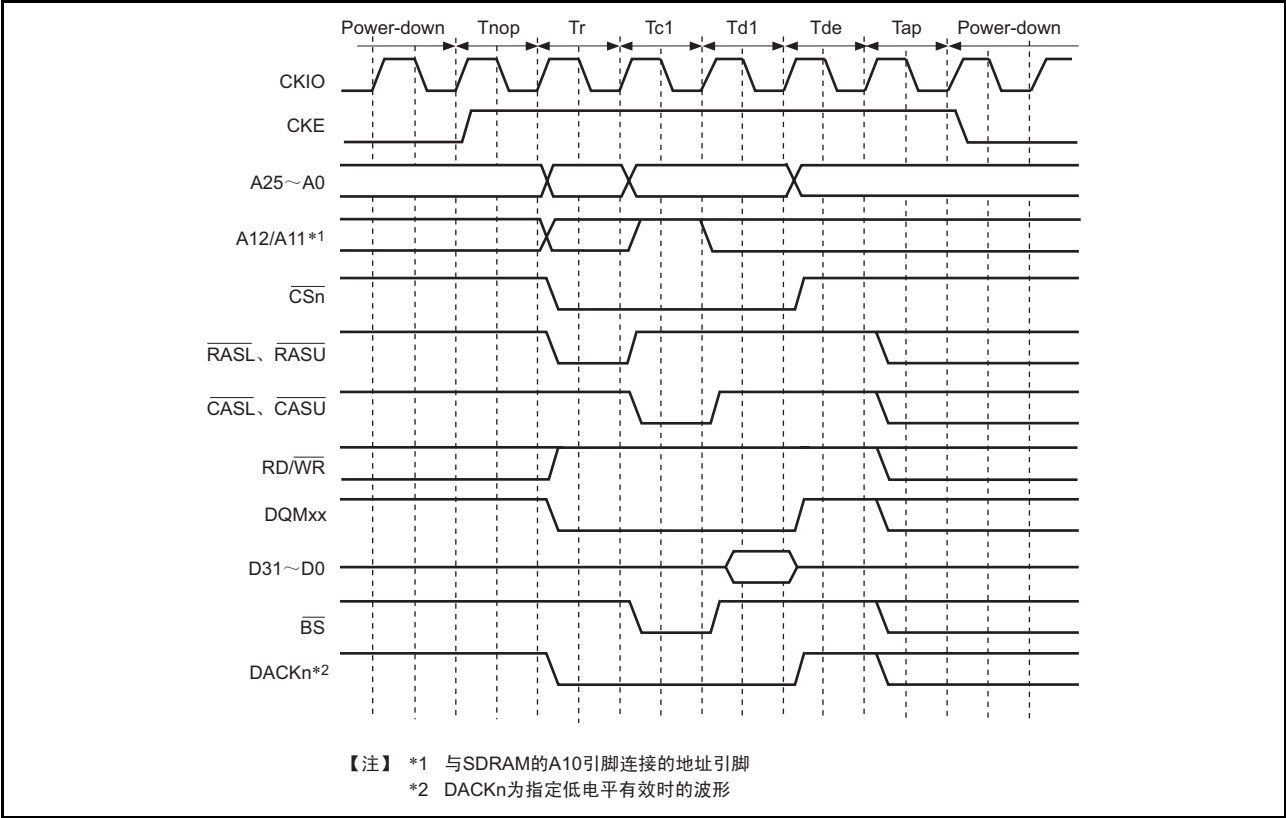


图 9.31 掉电模式的存取时序

(12) 上电顺序

为了使用 SDRAM，在上电后，经过所使用 SDRAM 规定的暂停期间后，需对 SDRAM 设定模式。必须由上电复位生成电路或软件实现暂停期间。

为了正确初始化 SDRAM，首先设定 BSC 的寄存器，然后写入 SDRAM 的模式寄存器。设定 SDRAM 的模式寄存器时，通过 $\overline{CSn}$ 、 $\overline{RASU}$ 、 $\overline{RASL}$ 、 $\overline{CASU}$ 、 $\overline{CASL}$ 及 $\overline{RD/WR}$ 信号组合，将当时的地址信号值输入 SDRAM。如果要设定的值为 X，可通过对 X+（区域 2：地址 H'FFFC4000、区域 3：地址 H'FFFC5000）执行字写入，将值 X 写入 SDRAM 内的模式寄存器。此时，忽略写入数据。设定本 LSI 支持的突发读取 / 单次写入（突发长度为 1）或突发读取 / 突发写入（突发长度为 1）、CAS 等待时间 2 ~ 3、Wrap 型 = 顺序及突发长度为 1 时，向表 9.18 所示的存取地址以字写入的方式写入任意数据。此时，在外部地址引脚的不低于 A12 的位输出 0。

表 9.18 写入 SDRAM 模式寄存器时的存取地址

- 区域2的设置

突发读取 / 单次写入 (突发长度为 1) 时

数据总线宽度	CAS 等待时间	存取地址	外部地址引脚
16 位	2	H'FFFC4440	H'0000440
	3	H'FFFC4460	H'0000460
32 位	2	H'FFFC4880	H'0000880
	3	H'FFFC48C0	H'00008C0

突发读取 / 突发写入 (突发长度为 1) 时

数据总线宽度	CAS 等待时间	存取地址	外部地址引脚
16 位	2	H'FFFC4040	H'0000040
	3	H'FFFC4060	H'0000060
32 位	2	H'FFFC4080	H'0000080
	3	H'FFFC40C0	H'00000C0

- 区域3的设置

突发读取 / 单次写入 (突发长度为 1) 时

数据总线宽度	CAS 等待时间	存取地址	外部地址引脚
16 位	2	H'FFFC5440	H'0000440
	3	H'FFFC5460	H'0000460
32 位	2	H'FFFC5880	H'0000880
	3	H'FFFC58C0	H'00008C0

突发读取 / 突发写入 (突发长度为 1) 时

数据总线宽度	CAS 等待时间	存取地址	外部地址引脚
16 位	2	H'FFFC5040	H'0000040
	3	H'FFFC5060	H'0000060
32 位	2	H'FFFC5080	H'0000080
	3	H'FFFC50C0	H'00000C0

模式寄存器的设定时序如图 9.32 所示。首先发行全部存储体预充电命令 (PALL)，接着发行 8 次自动刷新命令 (REF)，最后发行模式寄存器的写入命令 (MRS)。在 PALL 与第 1 次 REF 之间，插入 CS3WCR 的 WTRP[1:0] 位设定的空闲周期；在 REF 与 REF 之间及第 8 次 REF 与 MRS 之间，插入 CS3WCR 的 WTRC[1:0] 位设定的空闲周期。另外，在 MRS 与下一次发行的命令之间插入大于等于 1 个周期的空闲周期。

上电后且在全部存储体预充电 (PALL) 前，SDRAM 必须确保一定周期的空闲时间。所需的空闲时间，详情参阅 SDRAM 手册。复位信号的脉宽大于此空闲时间时，可立即设定模式寄存器。但是，复位信号的脉宽小于此空闲时间时，则须注意。

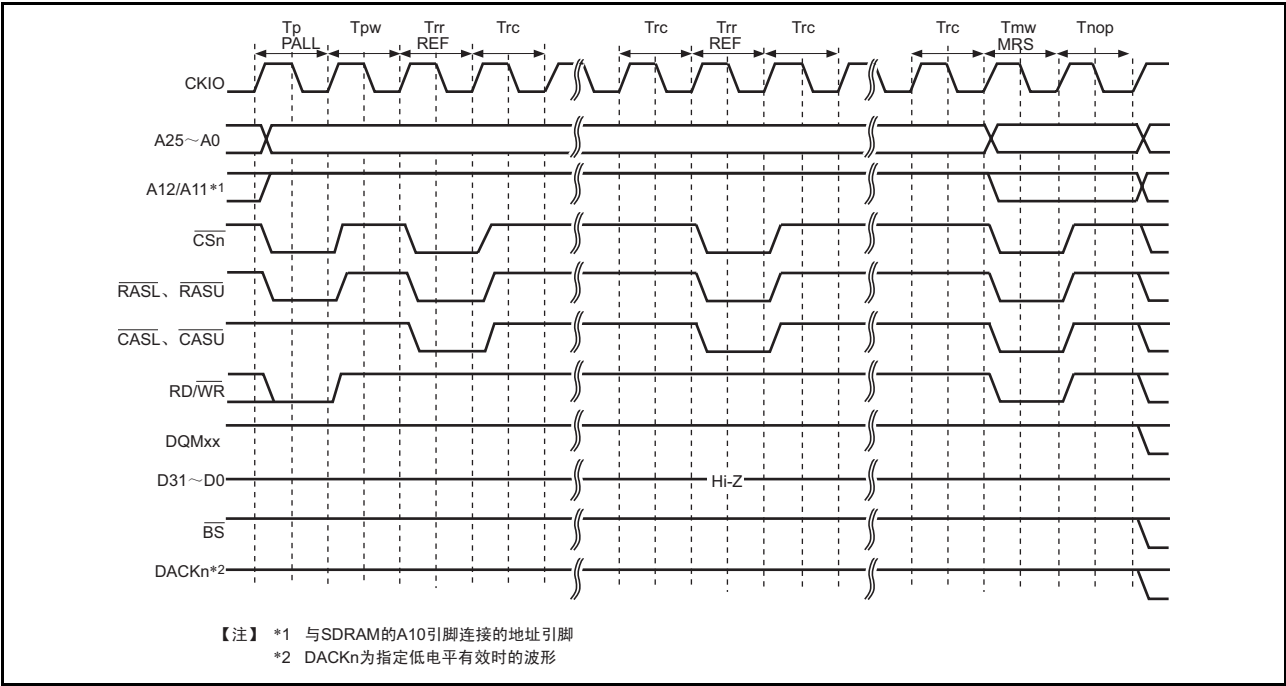


图 9.32 SDRAM 模式寄存器的写入时序 (以 JEDEC 为依据)

(13) 低功率 SDRAM

低功率 SDRAM 为可通过与普通 SDRAM 相同的协议存取的存储器。

低功率 SDRAM 与普通 SDRAM 规格的不同之处在于可更加详细地设定如下内容：自刷新时，使存储器的一部分为自刷新状态的部分刷新；或根据用户使用条件（温度）控制刷新时的低功耗等。部分刷新对在工作区（特定区域以外）内丢失数据也不影响运行的系统有效。详情参阅使用的低功率 SDRAM 的数据表。

低功率 SDRAM 除具备与普通 SDRAM 相同的模式寄存器之外，还具备扩展模式寄存器。扩展模式寄存器写入命令称为 EMRS，本 LSI 支持发行 EMRS 命令。

按照下表发行 EMRS。例如，将 H'0YYYYYYY 的数据以长字写入 H'FFFC5XX0 时，对 CS3 空间按 PALL→REF×8→MRS→EMRS 的顺序发行命令。发行 MRS 时的地址为 H'0000XX0，发行 EMRS 时的地址为 H'YYYYYYY。如果将 H'1YYYYYYY 的数据以长字写入 H'FFFC5XX0，则对 CS3 空间按 PALL→MRS→EMRS 的顺序发行命令。

表 9.19 发行 EMRS 命令时的输出地址

发行命令	存取地址	存取数据	写入存取长度	发行 MRS 命令时的地址	发行 EMRS 命令时的地址
CS2 MRS	H'FFFC4XX0	H'*****	16 位	H'0000XX0	-----
CS3 MRS	H'FFFC5XX0	H'*****	16 位	H'0000XX0	-----
CS2 MRS+EMRS (有刷新)	H'FFFC4XX0	H'0YYYYYYY	32 位	H'0000XX0	H'YYYYYYY
CS3 MRS+EMRS (有刷新)	H'FFFC5XX0	H'0YYYYYYY	32 位	H'0000XX0	H'YYYYYYY
CS2 MRS+EMRS (无刷新)	H'FFFC4XX0	H'1YYYYYYY	32 位	H'0000XX0	H'YYYYYYY
CS3 MRS+EMRS (无刷新)	H'FFFC5XX0	H'1YYYYYYY	32 位	H'0000XX0	H'YYYYYYY

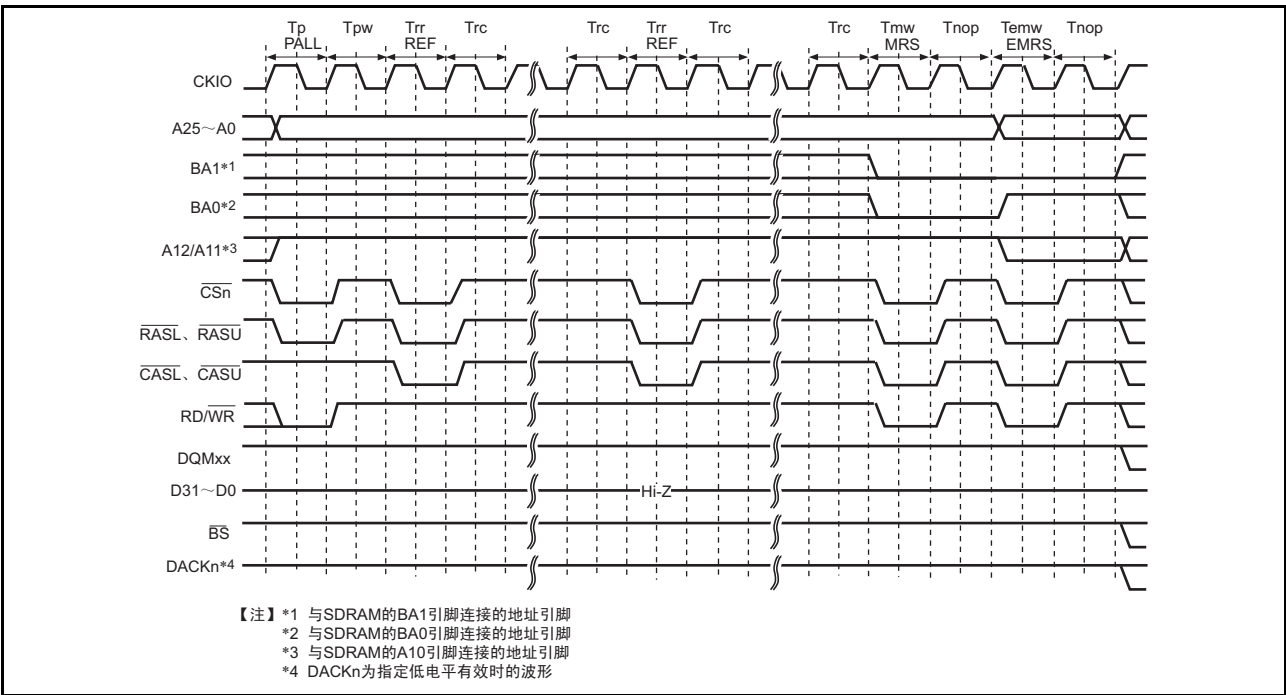


图 9.33 发行 EMRS 命令的时序

深度掉电模式

在低功率SDRAM还有称为深度掉电模式的低功耗模式。
部分自刷新仅对特定区域执行自刷新，而在深度掉电模式，整个存储器均不执行自刷新。
该模式对将整个存储器用作工作区域的系统有效。
在SDCR的DEEP位设定为1、RFSH位设定为1的状态，如果对RMODE位写入1，则低功率SDRAM转移至深度掉电模式；如果对RMODE位写入0，则CKE变为高电平，深度掉电模式被解除。解除深度掉电后，必须重新修改上电顺序，然后执行存取。

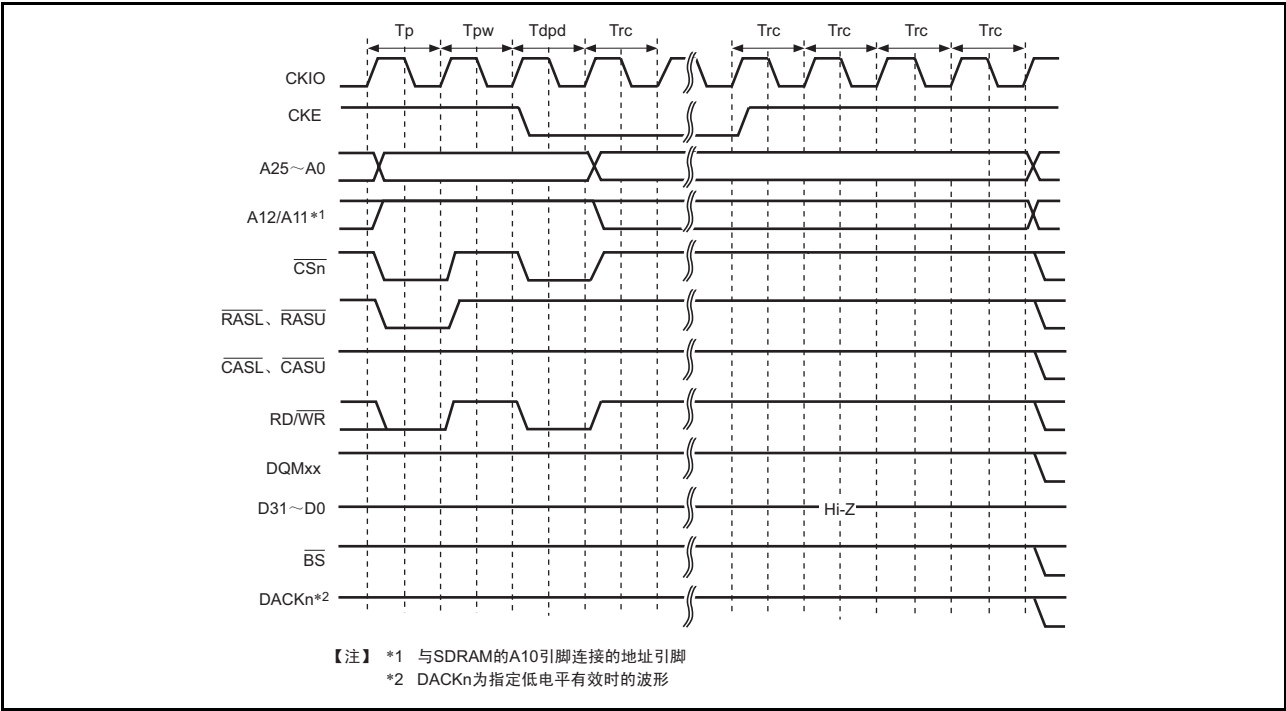


图 9.34 深度掉电模式转移时序

9.5.7 突发 ROM（时钟异步）接口

突发 ROM（时钟异步）接口可通过突发模式或页模式等地址转换，存取具有高速读取功能的存储器。基本执行与普通空间相同的存取，但是，第一个周期结束时，不将 RD 信号设定为无效，而仅转换地址，执行第 2 次及之后的存取。第 2 次及之后的存取时，在 CKIO 下降沿产生地址变化。
在第一个存取周期，插入 CSnWCR 的 W[3:0] 位设定的等待周期。在第 2 次及之后的存取周期，插入 CSnWCR 的 BW[1:0] 位指定的等待周期。
存取突发 ROM（时钟异步）时，BS 信号仅在第一个存取周期有效。另外，外部等待输入也仅在第一个存取周期有效。
不通过突发 ROM（时钟异步）接口执行突发运行的单次存取及写入存取时，存取时序与普通空间的时序相同。
总线宽度及存取长度与突发数的关系如表 9.20 所示。时序图如图 9.35 所示。

表 9.20 总线宽度及存取长度与突发数的关系

总线宽度	存取长度	CSnWCR.BST[1:0] 位	突发数	存取次数
8 位	8 位	无影响	1	1
	16 位	无影响	2	1
	32 位	无影响	4	1
	16 字节	00	16	1
		01	4	4
16 位	8 位	无影响	1	1
	16 位	无影响	1	1
	32 位	无影响	2	1
	16 字节	00	8	1
		01	2	4
		10*	4	2
			2、4、2	3
32 位	8 位	无影响	1	1
	16 位	无影响	1	1
	32 位	无影响	1	1
	16 字节	无影响	4	1

【注】* 总线宽度为 16 位、存取长度为 16 字节、CSnWCR.BST[1:0] 位为“10”时，突发数与存取次数因存取起始地址不同而变化：地址是 H'xxx0、H'xxx8 时，为 4-4 突发存取；地址是 H'xxx4、H'xxxC 时，为 2-4-2 突发存取。

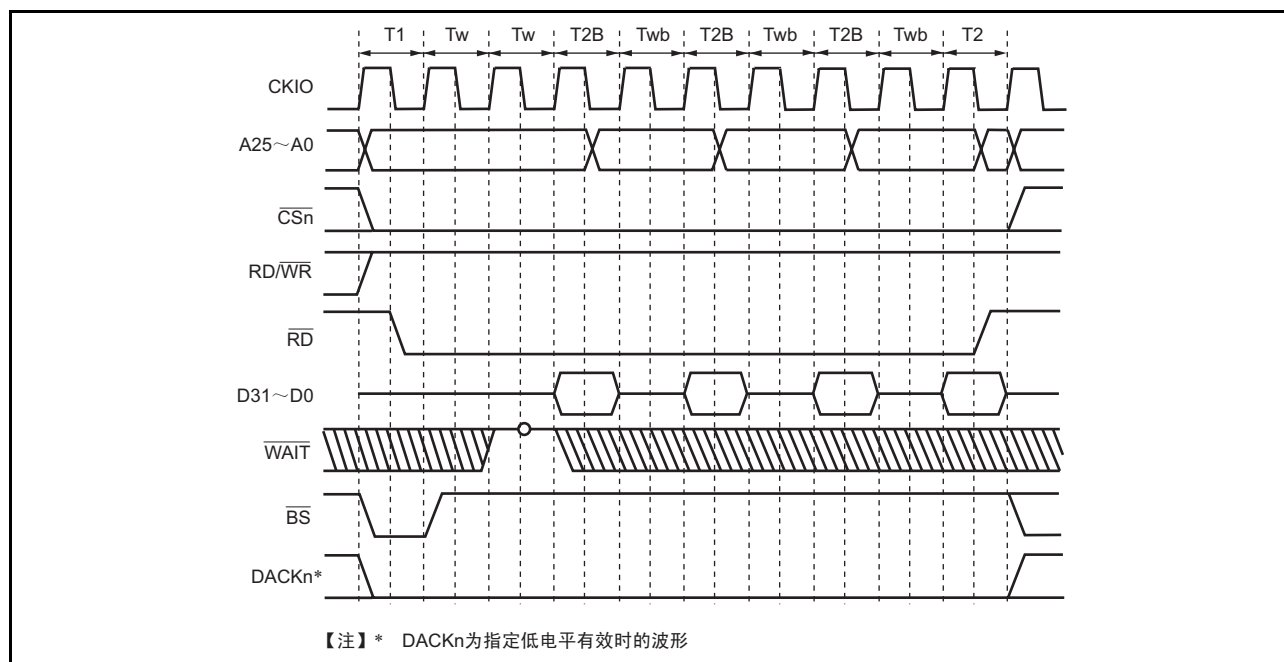


图 9.35 突发 ROM (时钟异步) 存取
(32 位总线宽度、16 字节传送 (突发数为 4)、2 个初次存取等待、
1 个第 2 次及之后的存取等待)

9.5.8 带字节选择的 SRAM 接口

带字节选择的 SRAM 接口是读取或写入的总线周期时均输出字节选择引脚 ($\overline{\text{WEn}}$) 的存储器接口。此接口具有 16 位数据引脚, 用于存取如 UB 或 LB 之类的带高位及低位字节选择引脚的 SRAM。

CSnWCR 的 BAS 位为 0 (初始值) 时, 带字节选择的 SRAM 接口的写入存取时序与普通空间接口的时序相同。但读取时, $\overline{\text{WEn}}$ 引脚的时序与普通空间接口时序不同, 从 $\overline{\text{WEn}}$ 引脚输出字节选择信号。基本存取时序如图 9.36 所示。写入时, 在字节选择引脚 ($\overline{\text{WEn}}$) 的时序写入存储器。详情参照所用存储器的数据表。

CSnWCR 的位 BAS 为 1 时, $\overline{\text{WEn}}$ 引脚与 RD/WR 引脚的时序有变化。基本存取时序如图 9.37 所示。特别是写入时, 在写入允许引脚 RD/WR 的时序写入存储器。必须设定 CSnWCR 的 HW[1:0] 位以确保从 RD/WR 无效到写入数据的保持时序。软件等待设定时的存取时序如图 9.38 所示。

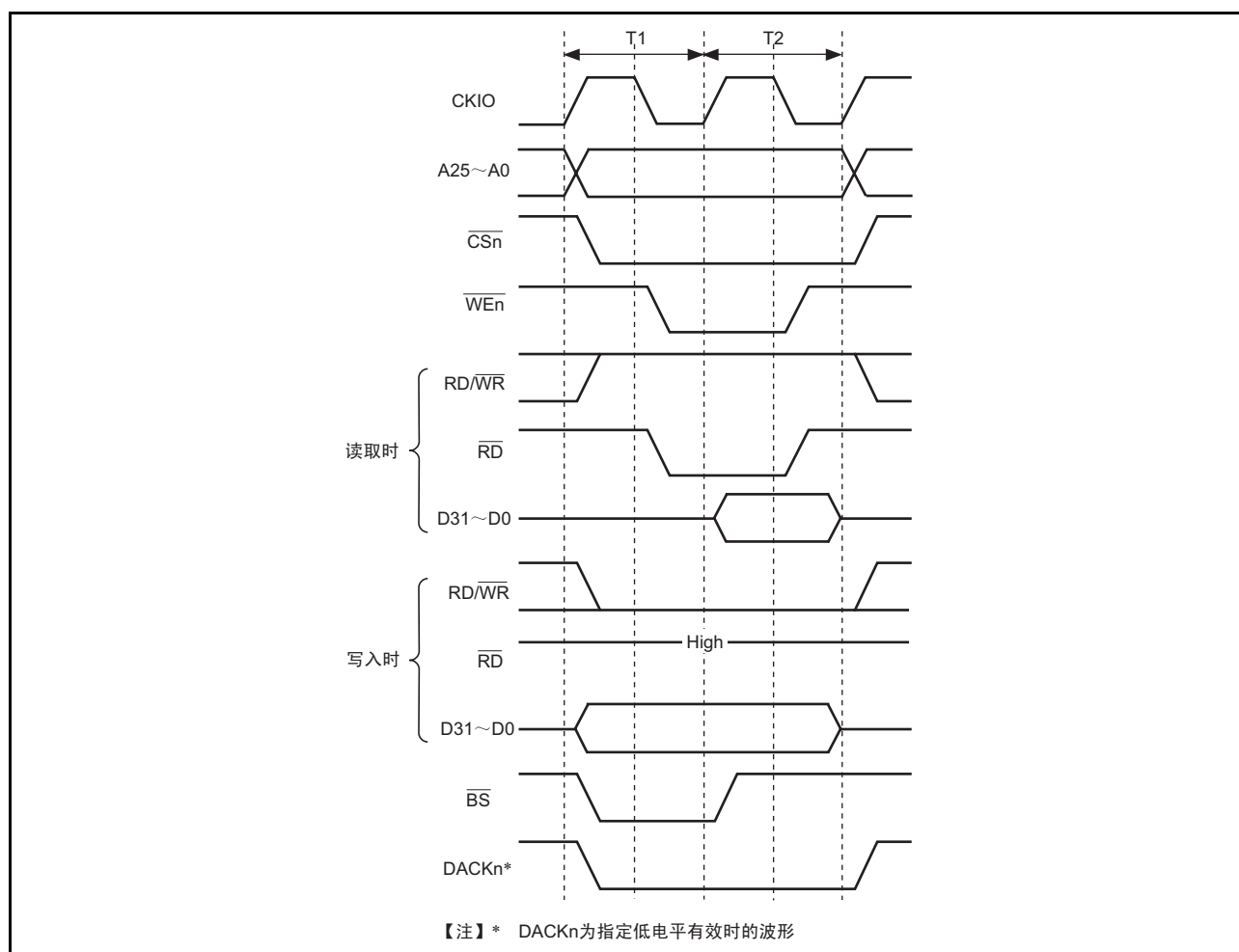


图 9.36 BAS=0、带字节选择的 SRAM 基本存取时序

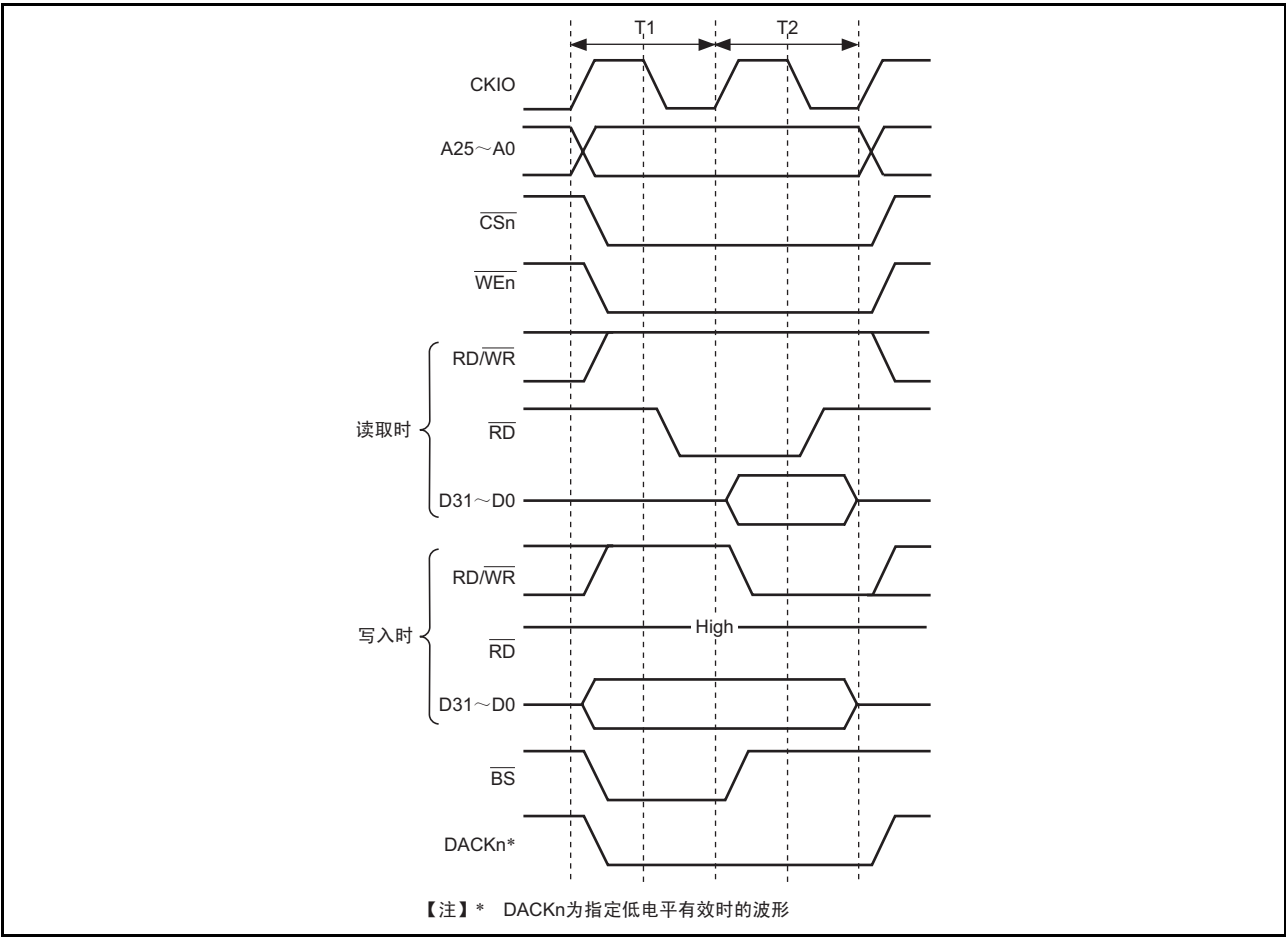


图 9.37 BAS=1、带字节选择的 SRAM 基本存取时序

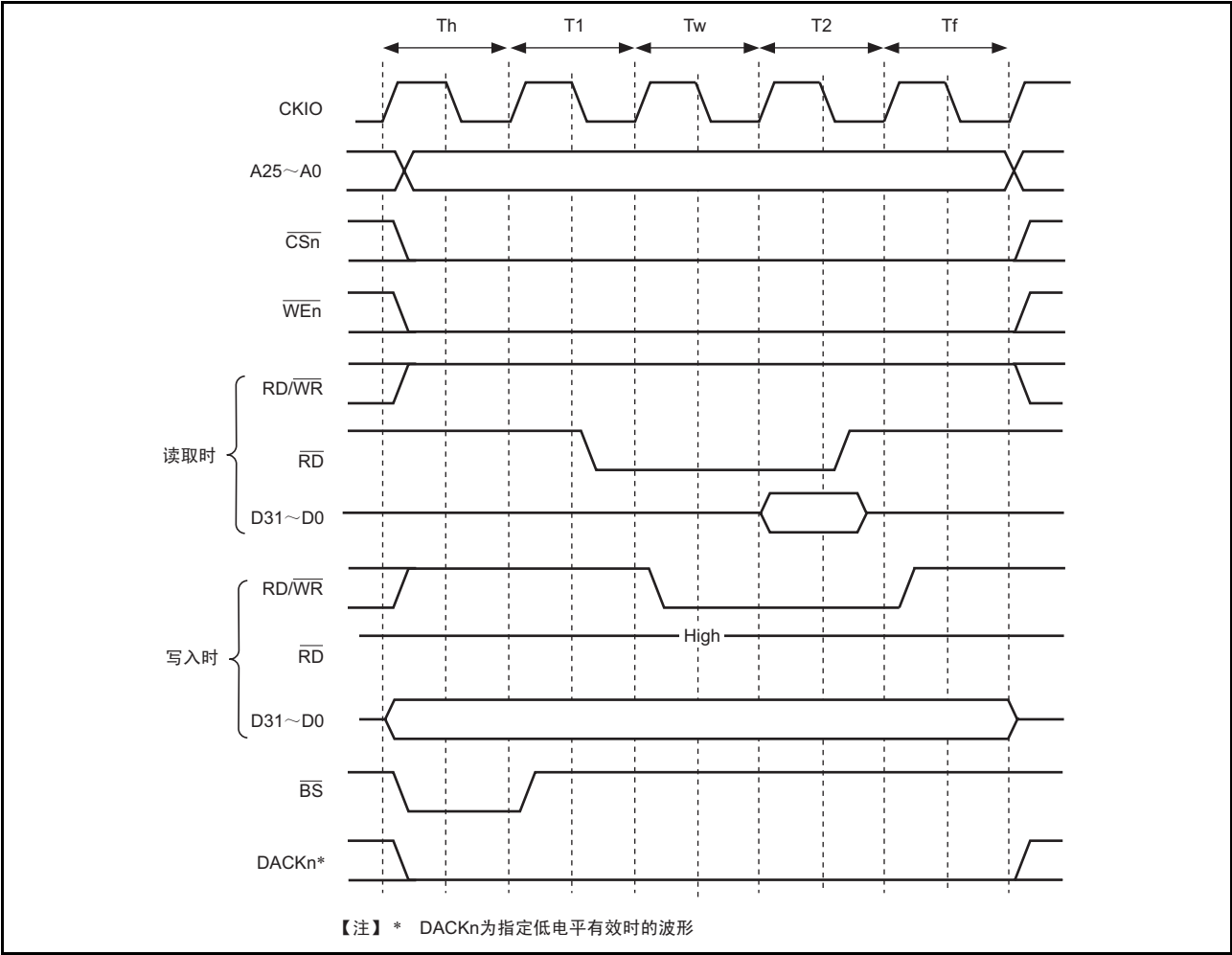


图 9.38 BAS=1、带字节选择的 SRAM 等待时序
(SW[1:0]=01、WR[3:0]=0001、HW[1:0]=01)

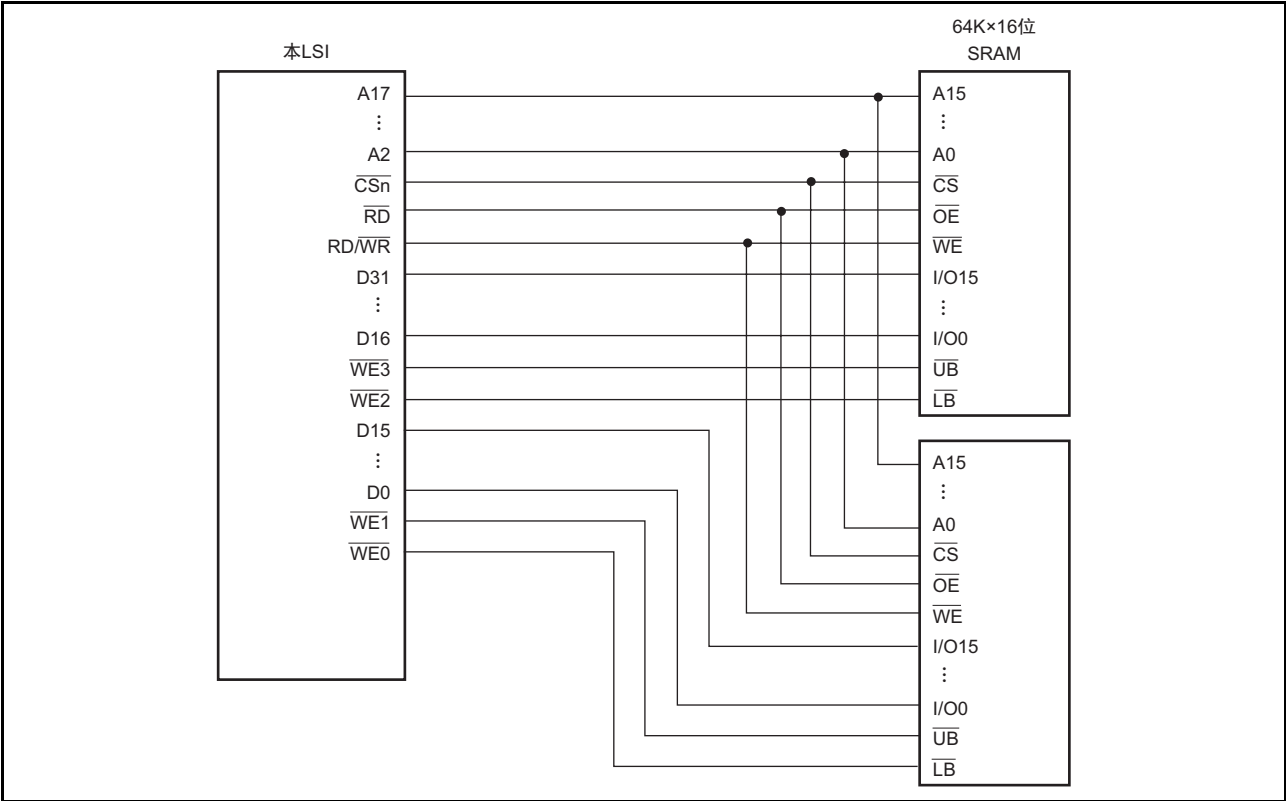


图 9.39 32 位数据宽度的带字节选择的 SRAM 连接例

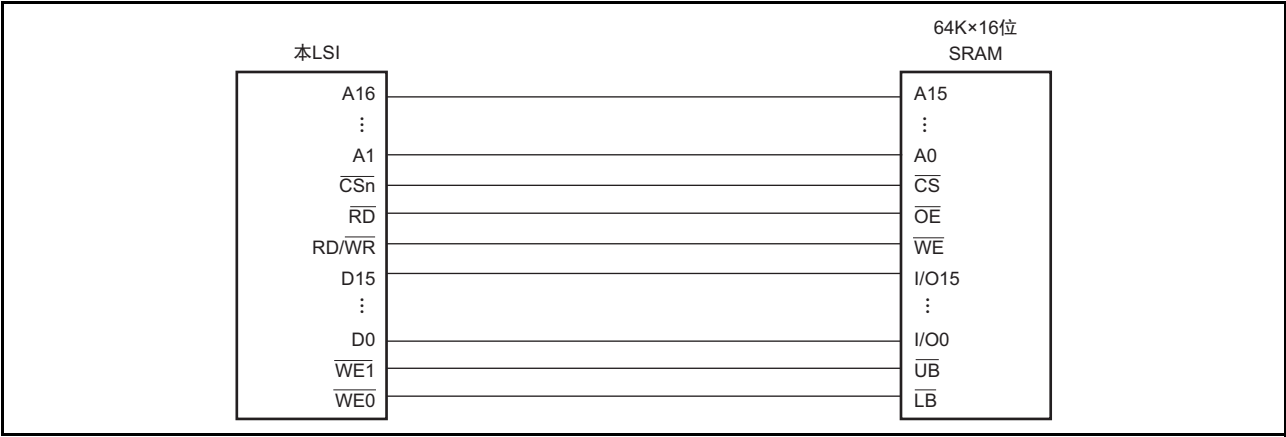


图 9.40 16 位数据宽度的带字节选择的 SRAM 连接例

9.5.9 PCMCIA 接口

在本 LSI，通过将 CSnBCR (n=5、6) 的 TYPE[2:0] 位设定为 B'101，将区域 5 及区域 6 设置为符合 JEIDA 标准 Ver4.2 (PCMCIA2.1Rev2.1) 的 IC 存储卡及 I/O 卡接口。通过 CSnWCR (n=5、6) 的 SA[1:0] 位，可将各区域的前 32MB 或后 32MB 设定为 IC 存储卡或 I/O 卡接口。例如，如果将 CS5WCR 的 SA1 位设定为 1，将 SA0 位设定为 0，则区域 5 的前 32MB 为 IC 存储卡接口、后 32MB 为 I/O 卡接口。

使用 PCMCIA 接口时，必须通过 CS5BCR 或 CS6BCR 的 BSZ[1:0] 位，将总线宽度设定为 8 位或 16 位。

本 LSI 的 PCMCIA 卡连接例如图 9.41 所示。为了实现 PCMCIA 卡的热插拔 (在系统通电的状态插拔卡)，本 LSI 的总线接口与 PCMCIA 卡之间，必须连接 3 态缓冲器。

由于 JEIDA 及 PCMCIA 标准未明确规定大端法模式的运行，所以单独规定了本 LSI 大端法模式的 PCMCIA 接口规格。

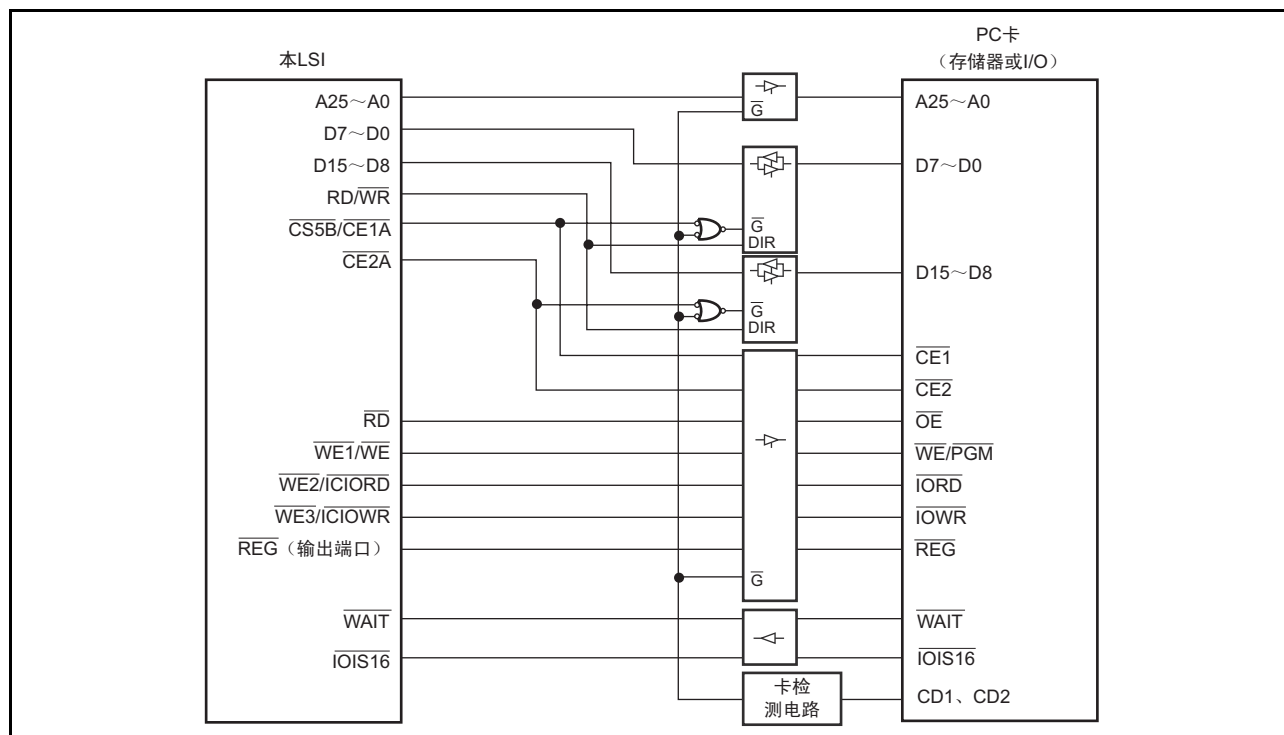


图 9.41 PCMCIA 接口连接例

(1) 存储卡接口的基本时序

PCMCIA 的 IC 存储卡接口的基本时序如图 9.42 所示。将区域 5 及 6 设定为 PCMCIA 接口时, 根据 CS5WCR 及 CS6WCR 的 SA[1:0] 位的设定, 作为 IC 存储卡接口执行总线存取。外部总线频率 (CKIO) 增大时, 对于 $\overline{\text{RD}}$ 及 $\overline{\text{WE}}$, 地址 (A25 ~ A0)、卡允许 ($\overline{\text{CE1A}}$ 、 $\overline{\text{CE2A}}$ 、 $\overline{\text{CE1B}}$ 、 $\overline{\text{CE2B}}$) 及写入周期时的写入数据 (D15 ~ D0) 的准备时间及保持时间会不足。对此, 本 LSI 可通过 CS5WCR 或 CS6WCR, 分别对区域 5 及区域 6 设定准备时间与保持时间。另外, 与普通空间接口一样, 可插入软件等待及 WAIT 引脚引起的硬件等待。PCMCIA 存储器的总线等待时序如图 9.43 所示。

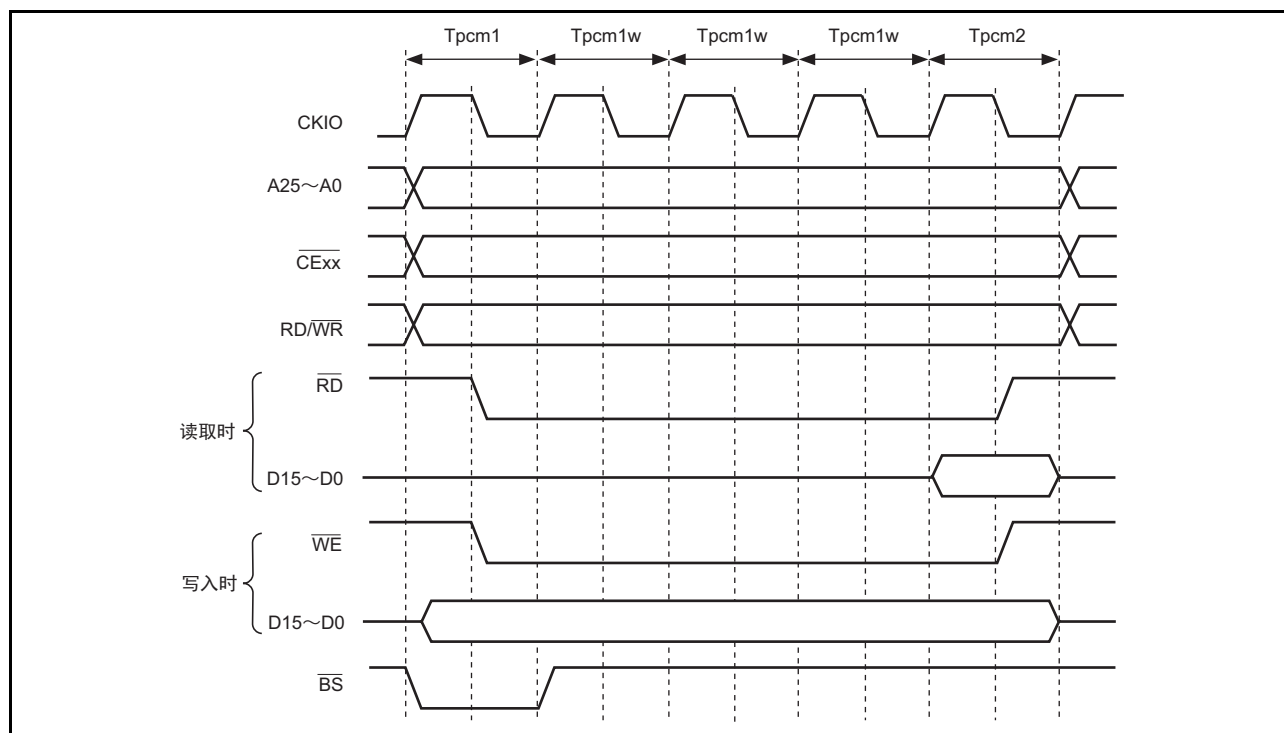


图 9.42 PCMCIA 存储卡接口的基本时序

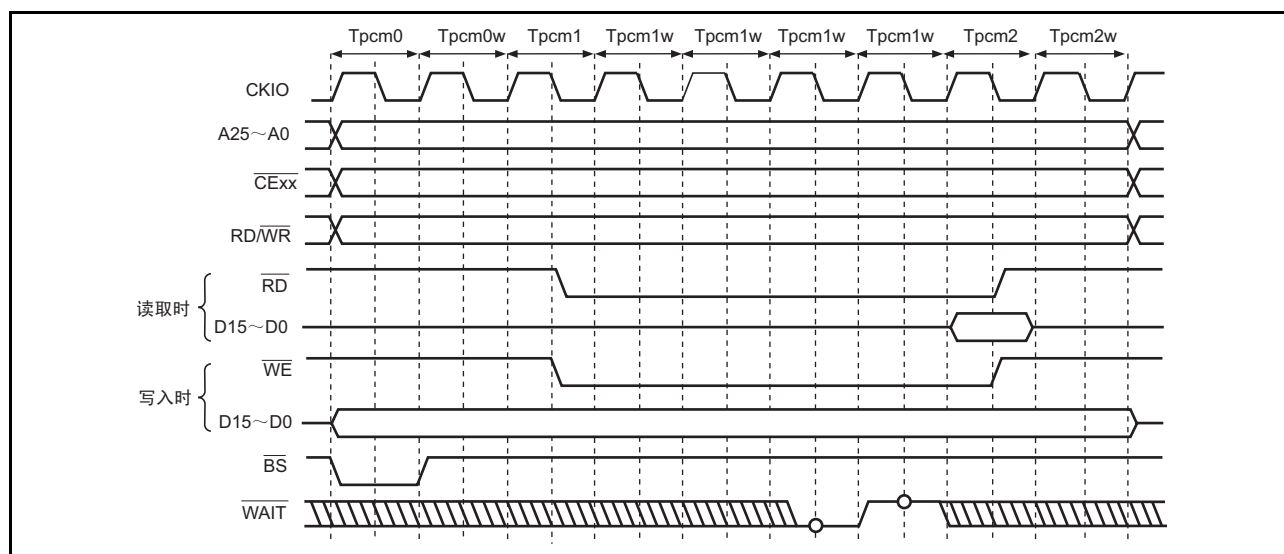


图 9.43 PCMCIA 存储卡接口的等待时序

(TED[3:0]=B'0010、PCW[3:0]=B'0000、TEH[3:0]=B'0001、1 个硬件等待)

使用端口等生成通用存储器与属性存储器的转换信号 $\overline{\text{REG}}$ 。如图 9.44 所示，通用存储器与属性存储器的总存储空间小于等于 32M 字节也可用时，通过将 A24 引脚用作 $\overline{\text{REG}}$ 信号，可作为 16M 字节的通用存储空间与 16M 字节的属性存储空间使用。



图 9.44 PCMCIA 空间分配的例子
(CS5WCR.SA[1:0]=B'10、CS6WCR.SA[1:0]=B'10)

(2) I/O 卡接口的基本时序

PCMCIA 的 I/O 卡接口时序如图 9.45 及图 9.46 所示。

存取 PCMCIA 的 I/O 卡时，必须存取高速缓存无效区域。

根据 CS5WCR 及 CS6WCR 的 SA[1:0] 位的设定及存取的地址，转换 I/O 卡接口与 IC 存储卡接口。

在小端法模式，将 PCMCIA 卡作为 I/O 卡接口存取时，使用 $\overline{\text{IOIS16}}$ 引脚，可更改 I/O 总线宽度的动态总线宽度更改 (dynamic bus sizing)。区域 5 或区域 6 的总线宽度设定为 16 位时，如果在字长度的 I/O 卡总线周期中， $\overline{\text{IOIS16}}$ 信号为高电平时，则总线宽度识别为 8 位，且在正在执行的 I/O 卡总线周期中，仅存取 8 位数据，并自动继续存取剩余的 8 位数据。

TED[3:0] 位设定为大于等于 1.5 个周期时，在 Tpci0 及所有 Tpci0w 周期的 CKIO 下降沿采样 $\overline{\text{IOIS16}}$ 信号，从 Tpci0 采样点开始经过 CKIO1.5 个周期后，反映在 CE2A 或 CE2B 信号。必须在所有采样点确定 $\overline{\text{IOIS16}}$ 信号，且中途不改变。

为了满足所用的 PC 卡从 $\overline{\text{ICIORD}}$ 与 $\overline{\text{ICIOWR}}$ 到 CE1 或 CE2 的准备时间，必须设定 TED[3:0] 位。

动态总线宽度更改 (dynamic bus sizing) 的基本波形如图 9.46 所示。

大端法模式不支持 $\overline{\text{IOIS16}}$ 信号，因此，在大端法模式，必须将 $\overline{\text{IOIS16}}$ 信号固定为低电平。

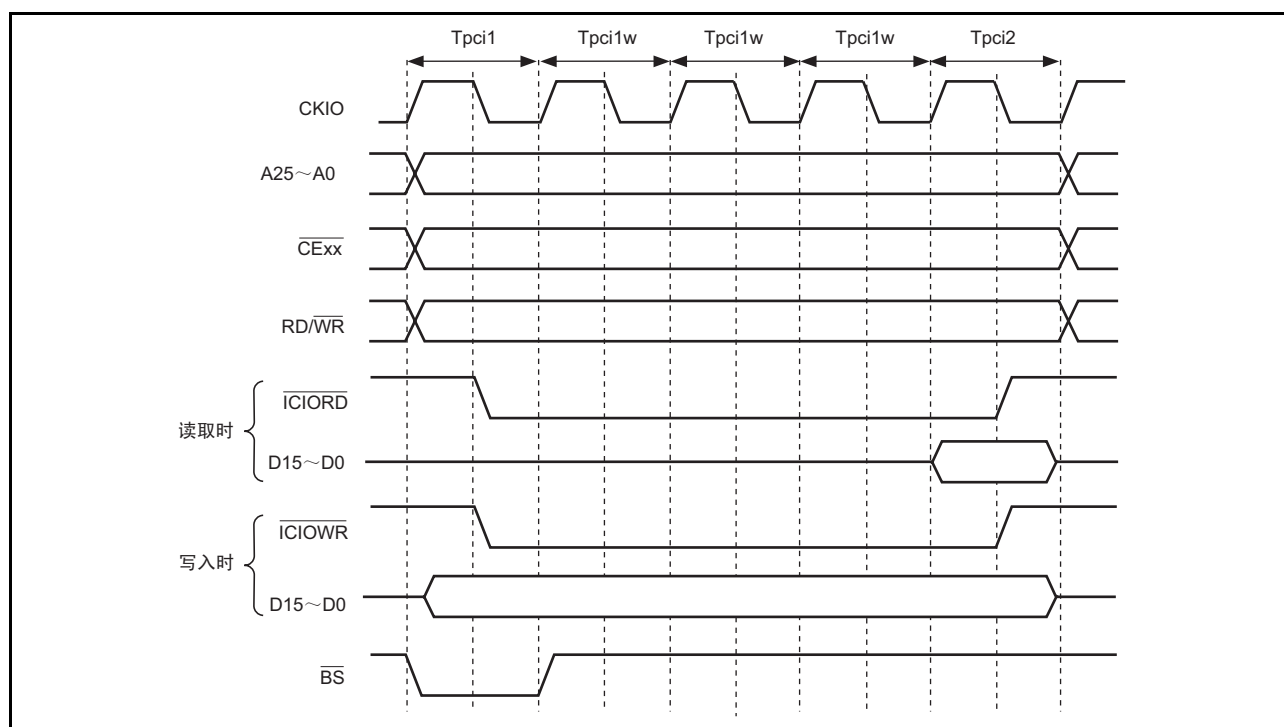


图 9.45 PCMCIA I/O 卡接口的基本时序

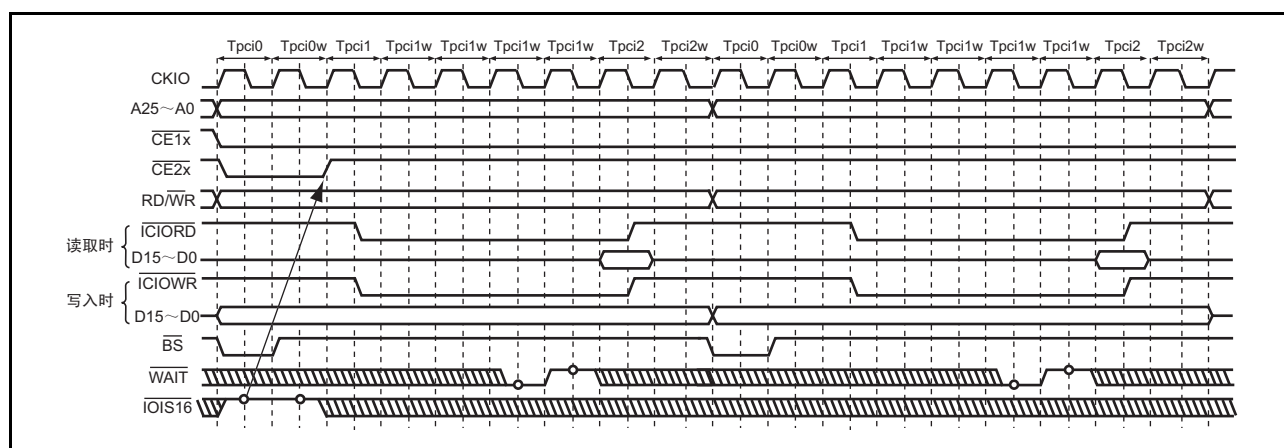


图 9.46 PCMCIA I/O 卡接口的动态总线宽度更改时序
(TED[3:0]=B'0010、PCW[3:0]=B'0000、TEH[3:0]=B'0001、1 个硬件等待)

9.5.10 突发 MPX-I/O 接口

突发 MPX 器件的连接例如图 9.47 所示。突发 MPX 空间的存取时序如图 9.48 ~ 图 9.51 所示。

通过设定 CS6BCR 的 TYPE[2:0] 位，可将区域 6 设定为地址 / 数据多路复用 I/O 接口。通过本接口可轻松连接使用地址 / 数据多路复用 32 位单总线的外部存储器控制器芯片。地址被输出至 D25 ~ D0；存取长度在地址存取时被输出至 D31 ~ D29。D31 ~ D29 输出与存取长度的对应关系参阅“9.4.3 CSn 空间等待控制寄存器 (CSnWCR) (n=0 ~ 7) (5) 突发 MPX-I/O”的 CS6WCR 说明。

对地址引脚 A25 ~ A0 输出原地址。

使用突发 MPX-I/O 接口时，总线宽度固定为 32 位。CS6BCR 的 BSZ[1:0] 位必须设定为 32 位。另外，可插入软件等待及 WAIT 引脚引起的硬件等待。

读取时，即使软件等待设定为 0，也可在地址输出后自动插入 1 个等待周期。

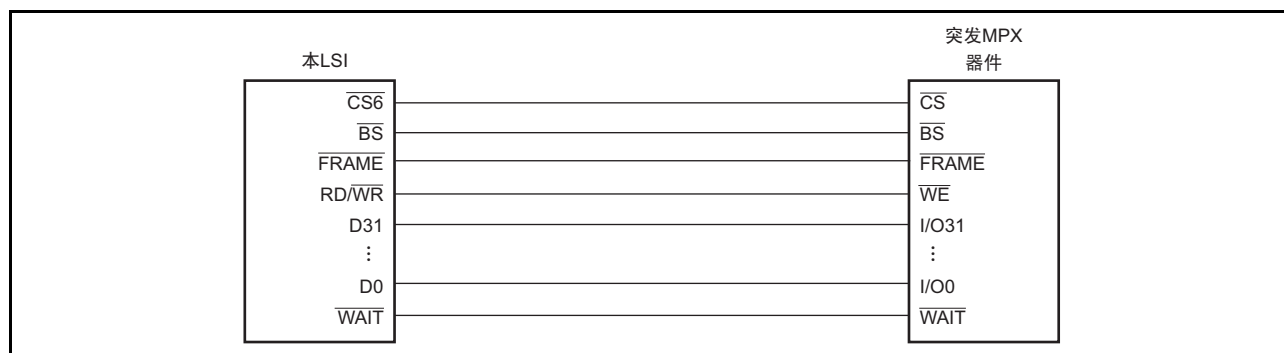


图 9.47 突发 MPX 器件的连接例

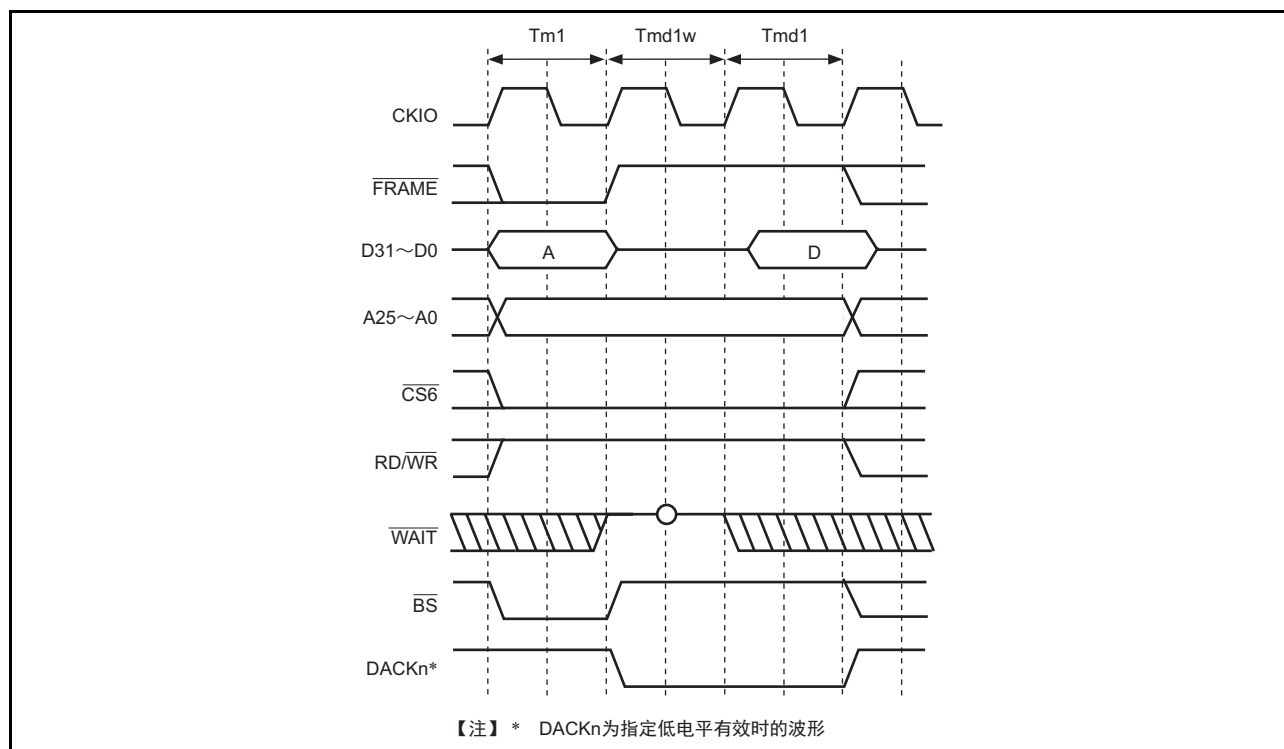


图 9.48 突发 MPX 空间的存取时序
(单次读取、无等待或 1 个软件等待)

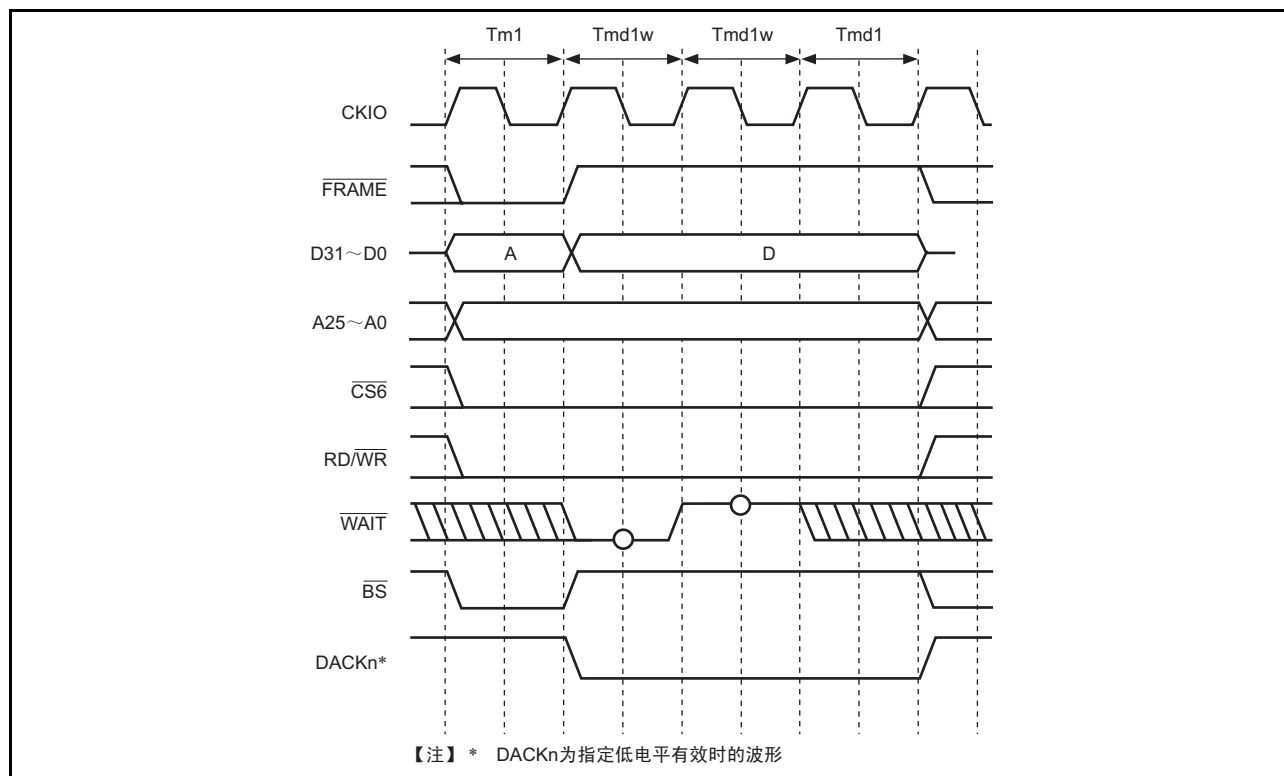


图 9.49 突发 MPX 空间的存取时序
(单次写入、1 个软件等待、1 个硬件等待)

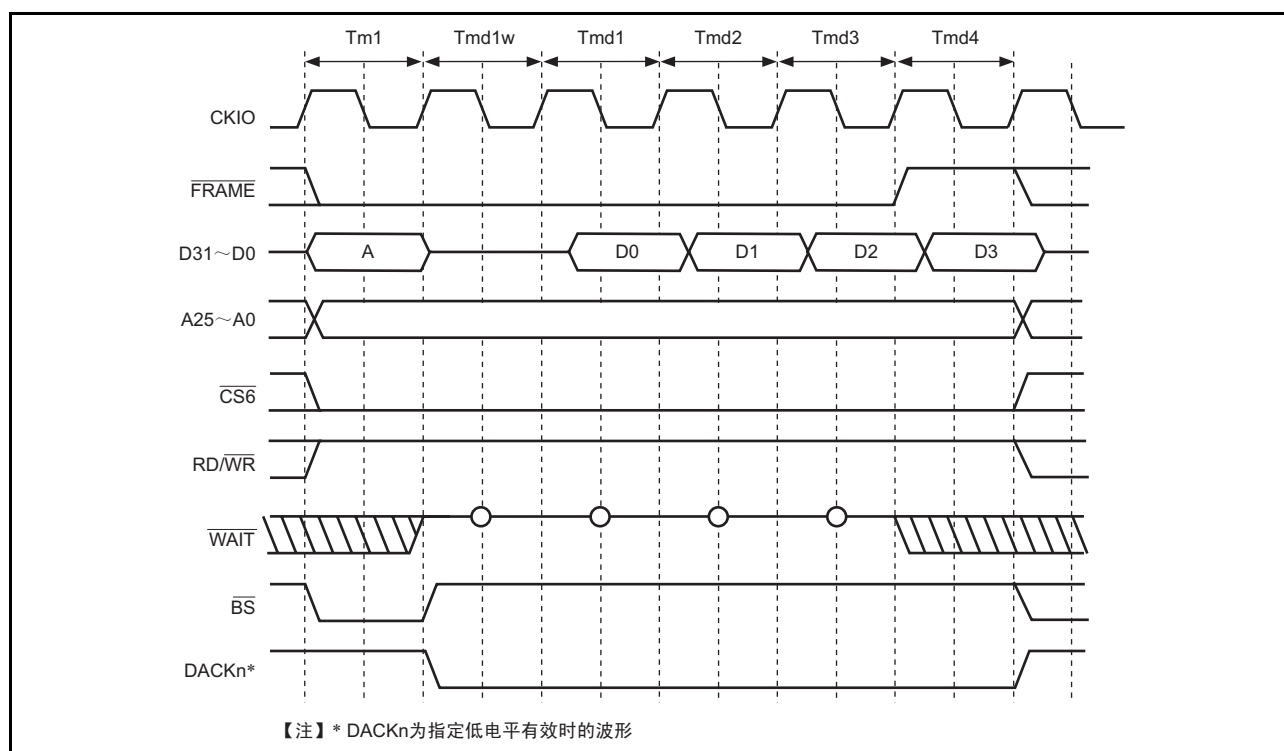


图 9.50 突发 MPX 空间的存取时序
(突发读取、无等待或 1 个软件等待、CS6WCR.MPXMD=0)

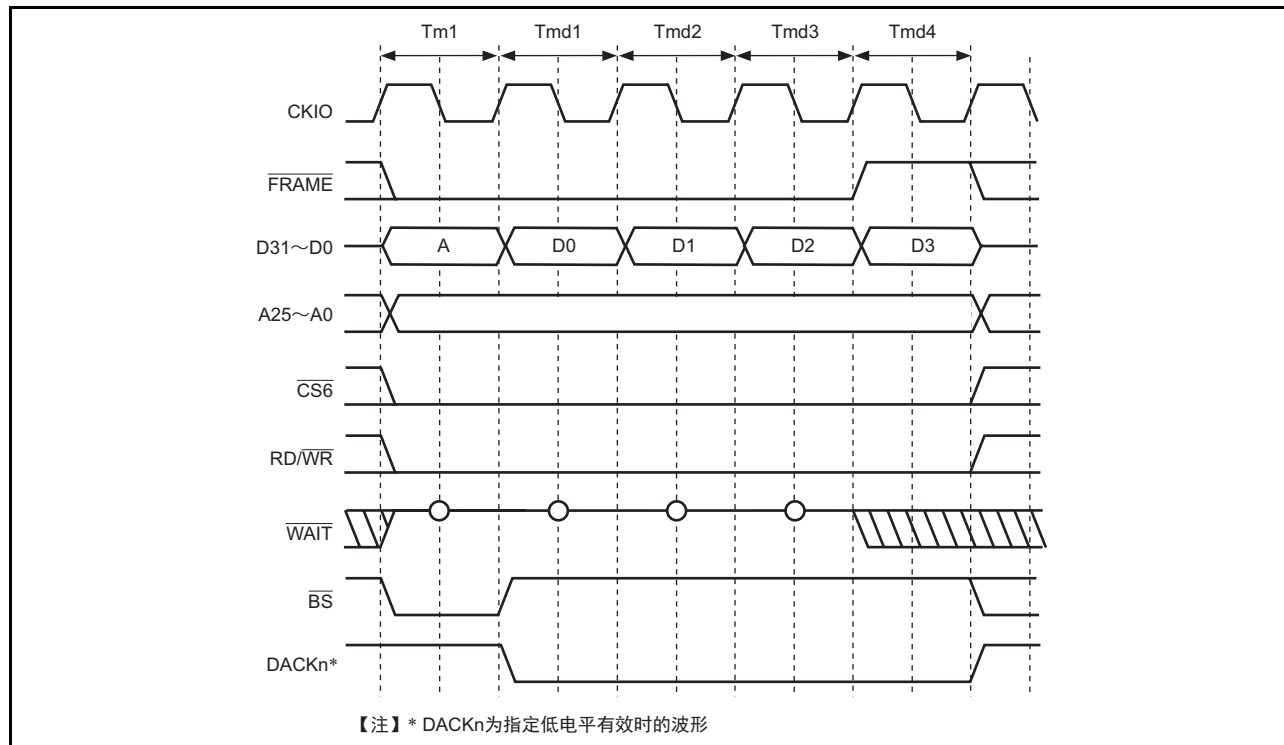


图 9.51 突发 MPX 空间的存取时序
(突发写入、无等待、CS6WCR.MPXMD=0)

9.5.11 突发 ROM (时钟同步) 接口

突发 ROM (时钟同步) 接口用于高速存取具有同步突发功能的 ROM。基本执行与普通空间相同的存取。本接口仅可在区域 0 设定。

在第一个存取周期，插入 CS0WCR 的 W[3:0] 位设定的等待周期。在第 2 次及之后的存取周期，插入 CS0WCR 的 BW[1:0] 位设定的等待周期。

存取突发 ROM (时钟同步) 时， $\overline{BS}$ 信号仅对第一个存取周期有效。外部等待输入也仅对第一个存取周期有效。

设定存储器时，如果总线宽度为 16 位，则突发长度必须设定为 8；如果总线宽度为 32 位，则突发长度必须设定为 4。不支持 8 位总线宽度。本接口对所有读取存取执行突发运行。例如，以 16 位总线宽度执行长字存取时，读取 2 个必要数据后，空读剩余的 6 个不要数据。

这样的空读周期会增加存储器存取时间、降低程序执行速度及 DMA 传送速度，因此，必须有效利用高速缓存有效空间的高速缓存填充的 16 字节读取或 DMA 的 16 字节读取。写入存取时，与普通空间存取时序相同。

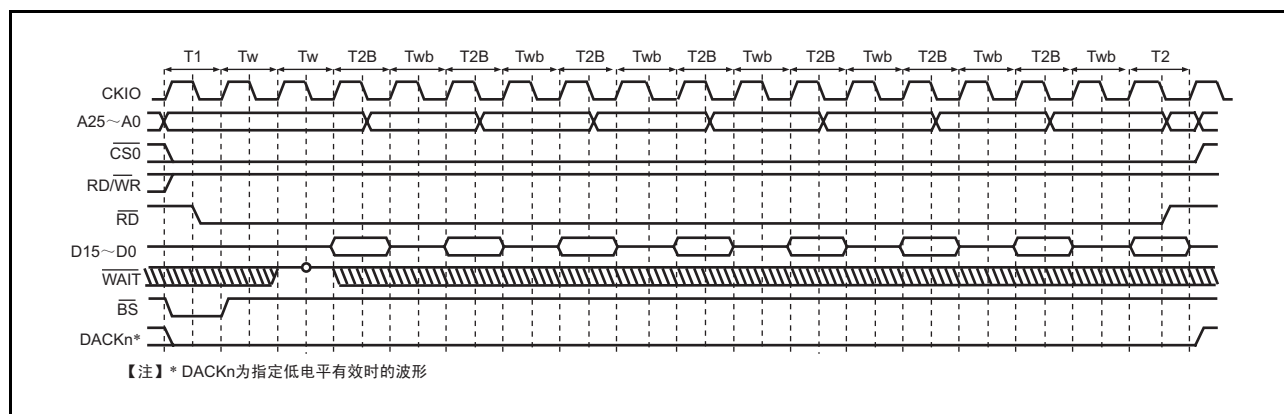


图 9.52 突发 ROM (时钟同步) 存取
(突发长度为 8、2 个初次存取等待、1 个第 2 次及之后的存取等待)

9.5.12 存取周期之间的空闲周期

由于 LSI 工作频率提高, 可能会在低速器件的数据输出完成时来不及关闭数据缓冲器, 导致与下一次器件的数据输出冲突, 从而降低器件的可靠性或引起误动作。为了防止出现此种情况, 需在连续存取周期之间插入空闲 (等待) 周期, 以避免数据冲突。

由 CSnWCR 的 WM 位及 CSnBCR 的 IWW[2:0] 位、IWRWD[2:0] 位、IWRWS[2:0] 位、IWRRD[2:0] 位、IWRRS[2:0] 位及 CMNCR 的 DMAIW[2:0] 位、DMAIWA 位指定存取周期之间的空闲周期数。可在存取周期之间插入空闲周期的条件如下:

1. 连续存取为写入—读取、写入—写入时
2. 连续存取为不同空间且为读取—写入时
3. 连续存取为相同空间且为读取—写入时
4. 连续存取为不同空间且为读取—读取时
5. 连续存取为相同空间且为读取—读取时
6. 在通过 DMA 单地址传送外部器件的数据输出周期后, 输出包括本 LSI 的其他器件的数据时 (DMAIWA=0)
7. 在通过 DMA 单地址传送外部器件的数据输出周期, 产生存取时 (DMAIWA=1)

有关上述存取周期之间空闲周期数的指定, 参阅各寄存器的说明。

除这些寄存器指定的存取周期之间等待的空闲周期之外, 为了确保连接内部总线的接口或多路复用引脚 ($\overline{WEn}$) 的最小脉宽, 有时会插入空闲周期。有关空闲周期的详情及空闲周期数估算方法, 说明如下:

说明从 $\overline{CSn}$ 无效到 $\overline{CSn}$ 或 $\overline{CSm}$ 有效的外部总线空闲周期数。此时的 $\overline{CSn}$ 及 $\overline{CSm}$ 也包括 PCMCIA 的 $\overline{CE2A}$ 、 $\overline{CE2B}$ 。

决定外部总线空闲周期数的项目有 8 项, 如表 9.21 所示。各项之间的关系如图 9.53 所示。

表 9.21 决定空闲周期数的项目

项目编号	内容	说 明	范围	注意事项
(1)	CMNCR.DMAIW[2:0] 的设定	指定通过 DMA 执行单地址传送时的空闲周期数。该项目仅在单地址传送时有效，为存取结束后产生的空闲周期。	0 ~ 12	注意：空闲周期数设定为 0 时，DACK 信号有可能会持续有效，并且带 DACK 的器件识别周期数与 DMAC 传送数不匹配，从而导致误动作。
(2)	CSnBCR.IW***[2:0] 的设定	指定非单地址传送时的空闲周期数。可对各个前后周期的组合分别指定空闲周期数。例如，读取 CS1 空间后读取其他 CS 空间时，如果将空闲周期数设定为大于等于 6，则将 CS1BCR.IWRRD[2:0] 设定为 B'100。该项目仅在非单地址传送时有效，为存取结束后产生的空闲周期。	0 ~ 12	注意：为无法连续存取的存储器类型时，不得设定为 0。
(3)	CSnWCR 的 SDRAM 相关设定	指定 SDRAM 存取时的预充电结束 / 启动等待、命令之间的空闲周期数。该项目仅在 SDRAM 存取时有效，为存取结束后产生的空闲周期。	0 ~ 3	必须根据使用的 SDRAM 的规格设定。
(4)	CSnWCR.WM 位设定	在 SDRAM 以外的存储器，可将外部 WAIT 引脚输入设定为有效或无效。设定为 0（外部 WAIT 有效）时，在存取结束后，插入用来评价外部 WAIT 引脚状态的 1 个空闲周期。设定为 1（无效）时，不产生本空闲周期。	0 ~ 1	
(5)	读取数据传送周期	为读取存取结束后产生的 1 个空闲周期。在被分割存取的最初及中途不产生空闲周期。另外，CSnWCR.HW[1:0] 不为 B'00 时，也不产生。	0 ~ 1	在 SDRAM、PCMCIA 的读取周期总是产生 1 个空闲周期。
(6)	内部总线空闲等	通过内部总线，传递来自 CPU、DMAC 等外部总线存取请求及结果。在内部总线的空闲周期及外部总线以外的存取中，外部总线为空闲状态。存取长度超过外部数据总线宽度时，由 BSC 进行分割存取，但在分割周期之间，不存在内部总线空闲周期等造成的影响。	0 ~	根据 $I\phi$: $B\phi$ 的时钟比，内部总线空闲周期数有时不为“0”。时钟比与内部总线的最小空闲周期数的关系如表 9.22、表 9.23 所示。
(7)	写入数据到达等待周期	写入存取时，等待写入数据到达后产生外部总线的写入周期。该写入数据等待为写入周期前产生的空闲周期。但前一个周期为写入且内部总线空闲周期比前一个存取的写入周期短时，由于与前一个存取并行处理，所以不产生本空闲周期（写入缓冲效果）。	0 ~ 1	写入 → 写入及写入 → 读取存取时，由于左边所述的写入缓冲效果，容易产生连续存取。不能连续存取时，必须通过 CSnBCR 等指定周期之间空闲的最小数。
(8)	不同存储器之间的空闲	为了确保多路引脚的最小脉宽，在产生存储器类型转换后的存取前，有时会插入空闲周期。根据存储器类型，即使不转换存储器类型也会产生空闲周期。	0 ~ 2.5	取决于每个存储器的类型。参照表 9.24。

(1)/(2) 项（任意一项为有效）、(3)/(4) 项（任意一项为有效）、(5)+(6)+(7) 项（按顺序产生，因此执行加法运算）及 (8) 项的这四项会同时产生，其中最大的一项为外部总线空闲周期数。确保最小空闲数周期时，必须设定 (1)/(2) 项的寄存器。

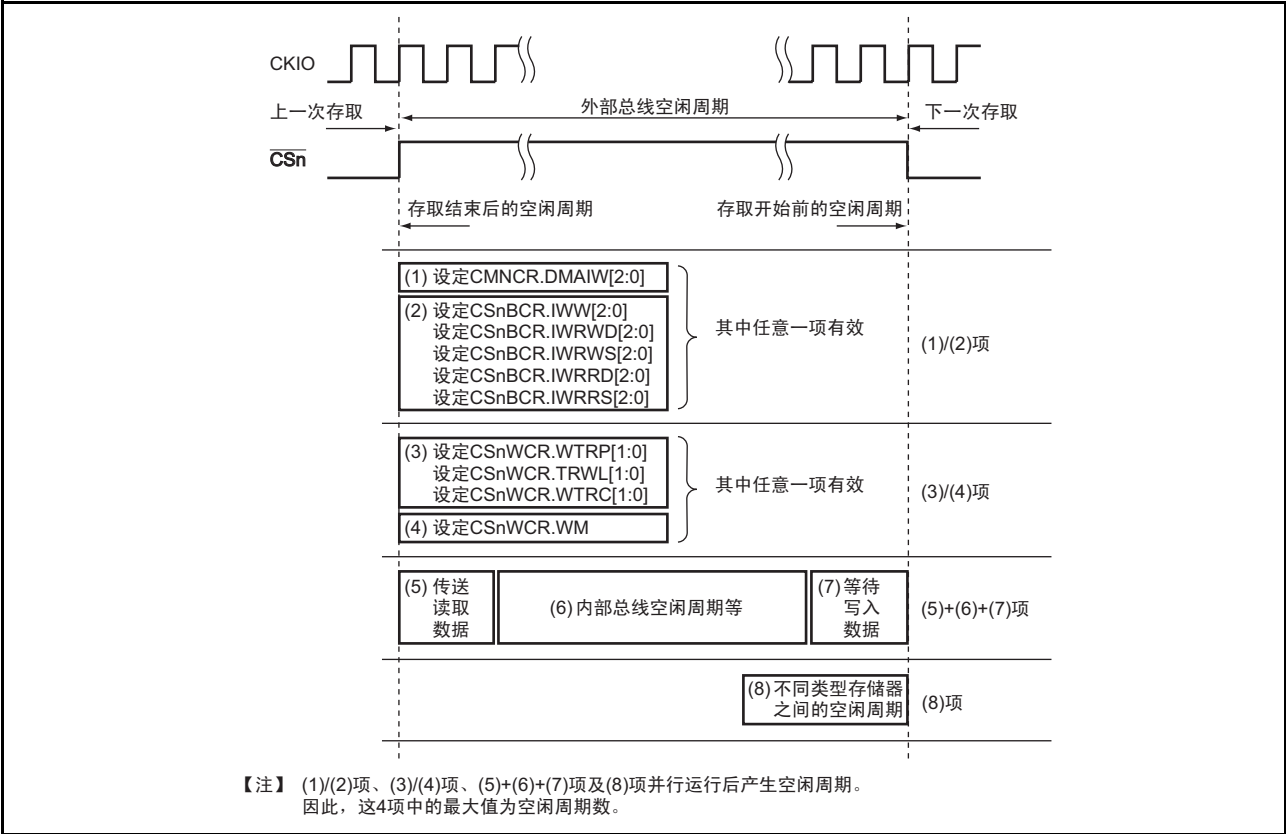


图 9.53 空闲周期的结构

表 9.22 内部总线的最小空闲周期数 (CPU 运行)

CPU 运行	时钟比 (I ϕ : B ϕ)					
	8 : 1	6 : 1	4 : 1	3 : 1	2 : 1	1 : 1
写入 → 写入	1	1	2	2	2	3
写入 → 读取	0	0	0	0	0	1
读取 → 写入	1	1	2	2	2	3
读取 → 读取	0	0	0	0	0	1

表 9.23 内部总线的最小空闲周期数 (DMAC 运行)

DMAC 运行	传送模式	
	双地址	单地址
写入 → 写入	0	2
写入 → 读取	0 或 2	0
读取 → 写入	0	0
读取 → 读取	0	2

- 【注】
1. 双地址传送的写入 → 写入、读取 → 读取运行在被分割的周期执行。
 2. 对于双地址传送的写入 → 读取运行，连续启动不同的通道时为 0；连续启动相同通道时为 2。
 3. 单地址的写入 → 读取、读取 → 写入连续启动不同的通道时。“写入”为从带 DACK 的器件向外部存储器的传送；“读取”为从外部存储器向带 DACK 的器件的传送。

表 9.24 在不同类型存储器之间存取前插入的空闲周期数

		后周期									
		SRAM	突发 ROM (异步)	MPX-I/O	字节 SRAM (BAS=0)	字节 SRAM (BAS=1)	SDRAM	SDRAM (低频模式)	PCMCIA	突发 MPX	突发 ROM (同期)
前 周 期	SRAM	0	0	1	0	1	1	1.5	0	0	0
	突发 ROM (异步)	0	0	1	0	1	1	1.5	0	0	0
	MPX-I/O	1	1	0	1	1	1	1.5	1	1	1
	字节 SRAM (BAS=0)	0	0	1	0	1	1	1.5	0	0	0
	字节 SRAM (BAS=1)	1	1	2	1	0	0	1.5	1	1	1
	SDRAM	1	1	2	1	0	0	—	1	1	1
	SDRAM (低频模 式)	1.5	1.5	2.5	1.5	0.5	—	1	1.5	1.5	1.5
	PCMCIA	0	0	1	0	1	1	1.5	0	0	0
	突发 MPX	0	0	1	0	1	1	1.5	0	0	0
	突发 ROM (同步)	0	0	1	0	1	1	1.5	0	0	0

估算周期之间最小空闲周期数的例子如图 9.54 所示。实际运行时，由于写入缓冲的效果，空闲周期可能比估算值小，或由于执行 CPU 指令或 CPU 寄存器竞争引起的拆分，产生内部总线空闲周期，从而产生比估算值大的空闲周期。因此，使用估算值时，必须同时考虑这些误差。

周期间空闲数的估算例

这是估算CPU存取时从CS1空间向CS2空间传送数据的例。按照CS1读取→CS1读取→CS2写入→CS2写入→CS1读取→…的顺序重复传送

- 条件
CS1BCR及CS2BCR的周期间空闲指定均指定为0。
CS1WCR及CS2WCR的WM位为1（外部WAIT引脚无效），HW[1:0]为00（不扩展CS无效）。
t_φ:B_φ设置为4:1，传送过程中不执行其他处理。
CS1与CS2均可连接普通SRAM，总线宽度为32位，存取长度也为32位。

在各周期之间估算决定空闲周期数的项目。下表中，R表示读取，W表示写入。

项目	R→R	R→W	W→W	W→R	备 注
(1)/(2)	0	0	0	0	CSnBCR的设定为0
(3)/(4)	0	0	0	0	WM位为1
(5)	1	1	0	0	读取周期后产生
(6)	0	2	2	0	参照表8.19的t _φ :B _φ =4:1部分
(7)	0	1	0	0	由于写入缓冲效果，因此在第2次不产生
(5)+(6)+(7)	1	4	2	0	
(8)	0	0	0	0	为SRAM→SRAM
估算的空闲周期数	1	4	2	0	(1)/(2)项、(3)/(4)项、(5)+(6)+(7)项、(8)项中的最大值
实际产生的空闲周期数	1	4	2	1	在W→R周期产生不匹配，是因为将(6)项的内部空闲周期数估算为0，而实际由于执行环路判断指令，产生内部空闲周期。

图 9.54 空闲周期数的估算例与实际的比较

9.5.13 总线仲裁

在通常状态, 本 LSI 的总线仲裁占用总线权。接受来自其它器件的总线权请求后, 释放总线。

在总线周期边界传递总线权。请求总线权时, 如果未执行总线周期, 则立即释放总线权。如果正在执行总线周期, 则等待总线周期完成后再释放总线权。即使从 LSI 外部看来未执行总线周期, 也会有在存取周期之间插入等待等内部已开始执行总线周期的情况。因此不能根据 $\overline{\text{CSn}}$ 信号及其他的总线控制信号立刻判断是否释放总线。不可执行总线释放的状态如下:

1. 高速缓存未命中引起的 16 字节传送过程中
2. 高速缓存回写过程中
3. TAS 指令的读取周期与写入周期之间
4. 数据总线宽度小于存取长度而产生的多个总线周期 (例如: 对 8 位数据总线宽度的存储器执行长字存取的总线周期之间)
5. 通过 DMAC 传送 16 字节时
6. CMNCR 的 BLOCK 位设定为 1 时
7. 传送 LCDC 的 16 字节~128 字节时

另外, 通过 CMNCR 的 DPRTY[1:0] 位, 可选择在 DMAC 突发传送过程中是否有接受总线权使用请求。

本 LSI 只要不接受总线请求, 就拥有总线权, 接受外部总线权请求 $\overline{\text{BREQ}}$ 有效 (低电平), 在正执行的总线周期结束后立即释放总线, 并使总线使用允许 $\overline{\text{BACK}}$ 有效 (低电平); 还可在接受表示外部器件已释放总线的 $\overline{\text{BREQ}}$ 无效 (高电平) 后, 使 $\overline{\text{BACK}}$ 无效, 并重新使用总线。

使用 SDRAM 接口时, 如果有激活的存储体, 则发行全部存储体的预充电命令 (PALL), 并在结束后释放总线。

总线释放的具体顺序如下。首先, 与 CKIO 的上升沿同步, 将地址总线及数据总线设置为高阻抗。0.5 个周期后, 使总线使用允许信号与 CKIO 的下降沿同步后有效。接着, 在 CKIO 上升沿, 将总线控制信号 ($\overline{\text{BS}}$ 、 $\overline{\text{CSn}}$ 、 $\overline{\text{RASU}}$ 、 $\overline{\text{RASL}}$ 、 $\overline{\text{CASU}}$ 、 $\overline{\text{CASL}}$ 、 $\overline{\text{CKE}}$ 、 $\overline{\text{DQMxx}}$ 、 $\overline{\text{WE}}_n$ 、 $\overline{\text{RD}}$ 和 $\overline{\text{RD/WR}}$) 设置为高阻抗。这些总线控制信号最迟在设置为高阻抗的 1 个周期前为高电平。在 CKIO 下降沿, 采样总线权请求信号。此外, 通过设定 CMNCR 的 HIZCNT 位, 在总线释放过程中, $\overline{\text{CKE}}$ 、 $\overline{\text{RASU}}$ 、 $\overline{\text{RASL}}$ 、 $\overline{\text{CASU}}$ 、 $\overline{\text{CASL}}$ 也可由之前的值继续驱动。

从外部器件再次获得总线权的顺序如下。如果在 CKIO 的下降沿, 检测出 $\overline{\text{BREQ}}$ 无效, 则 1.5 个周期后, 以高电平驱动总线控制信号。在随后的时钟下降沿, 总线使用允许信号无效。之后在 CKIO 的上升沿, 驱动地址及数据总线。最早从驱动地址及数据信号的同一时钟上升沿开始, 将总线控制信号设置为有效并实际开始执行总线周期。总线仲裁时序如图 9.55 所示。

总线权释放过程中需刷新 SDRAM 时, 要求返还总线权, 因此设置 $\overline{\text{REFOUT}}$ 作为请求信号。有关 $\overline{\text{REFOUT}}$ 引脚的选择, 详情参阅“第 25 章 引脚功能控制器 (PFC)”。 $\overline{\text{REFOUT}}$ 在获取总线权前一直保持低电平有效。外部器件必须通过 $\overline{\text{REFOUT}}$ 有效, 使 $\overline{\text{BREQ}}$ 无效并返还总线权。必须注意: 如果超过规定的刷新间隔后还未返还总线权, 则不能执行刷新, 因此无法保证 SDRAM 的内容。

本 LSI 获取总线权之前, 不执行释放总线权时的 SLEEP 指令 (转移至睡眠模式、深度待机模式或软件待机模式)。释放总线权时的手动复位也相同。

在软件待机模式或深度待机模式, 忽略 $\overline{\text{BREQ}}$ 输入, $\overline{\text{BACK}}$ 输出为高阻抗状态。在该状态需请求总线权时, 通过下拉 $\overline{\text{BACK}}$ 引脚, 转移至软件待机模式或深度待机模式的同时变为总线权释放状态。

必须在总线使用允许 ($\overline{\text{BACK}}$ 的低电平有效) 后, 执行总线权请求 ($\overline{\text{BREQ}}$ 的低电平有效) 后的总线权释放 ($\overline{\text{BREQ}}$ 的高电平无效)。 $\overline{\text{BACK}}$ 有效之前使 $\overline{\text{BREQ}}$ 无效时, 根据 $\overline{\text{BREQ}}$ 无效时序, $\overline{\text{BACK}}$ 仅在 1 个周期有效, 且外部器件与本 LSI 之间可能产生总线冲突。

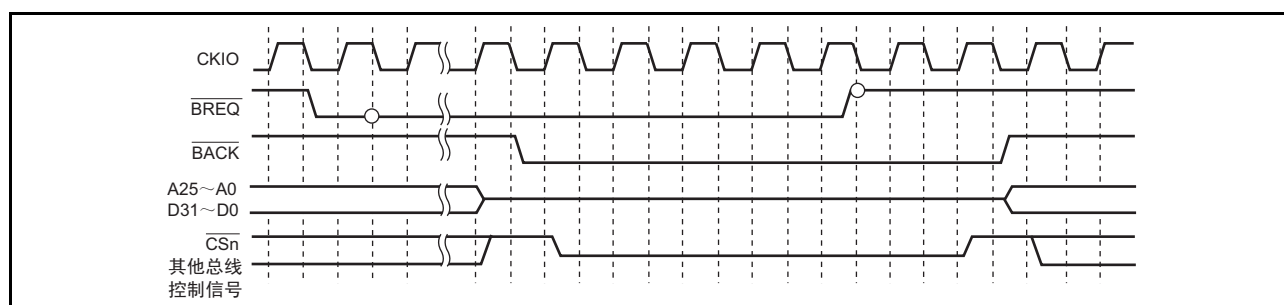


图 9.55 总线仲裁时序 (时钟模式 2)

9.5.14 其他

(1) 复位

总线状态控制器 (BSC) 仅在上电复位时完全被初始化。上电复位时, 使复位的内部时钟同步后, 与正在执行的总线周期无关, 使所有信号为无效, 并关闭数据输出缓冲器。另外, 初始化所有控制寄存器。在软件待机模式、睡眠及手动复位时, 总线状态控制器的控制寄存器不被初始化。手动复位时, 执行正在执行的总线周期直到完成。手动复位信号有效时, 因 RTCNT 递增计数, 所以产生刷新请求, 并启动刷新周期。

(2) 从 LSI 内部总线主控器看到的存取

本 LSI 内部分为 CPU 总线、内部总线及外围总线等 3 种总线。CPU 及高速缓冲存储器与 CPU 总线连接; 除 CPU 之外的内部总线主控器及总线状态控制器与内部总线连接; 低速的外围模块分别与外围总线连接。另外, 高速缓冲存储器以外的内部存储器, 与 CPU 总线及内部总线连接。可从 CPU 总线存取内部总线, 但不可从内部总线存取 CPU 总线。因此产生以下情况。

可从除 CPU 之外的内部总线主控器 (DMAC 等) 存取高速缓冲存储器以外的内部存储器, 但不可存取高速缓冲存储器。通过除 CPU 之外的内部总线主控器, 向外部存储器写入时, 其结果可能为外部存储器的内容与高速缓存的内容不符; 通过除 CPU 之外的内部总线主控器向外部存储器写入时, 传送至该地址的数据可能存在于高速缓存, 此时需用软件清除高速缓冲存储器。

在高速缓存有效空间, CPU 开始读取存取, 并搜索高速缓存。如果数据保留在高速缓存, 则输入该数据存取完成。高速缓存中无数据时, 通过内部总线, 填充高速缓存数据。此时, 启动 4 个连续的长字读取。关于字节、字操作数存取或转移至奇数字边界 ($4n+2$) 时的未命中, 在芯片外部接口必须通过 4 个长字存取进行填充。在高速缓存无效空间, 根据实际的存取地址存取。偶数字边界 ($4n$) 的取指令为长字存取; 奇数字边界 ($4n+2$) 的取指令为字存取。

在内部外围模块的读取周期, 通过内部总线与外围总线, 启动读取周期。读取的数据则经过外围总线、内部总线、CPU 总线, 传送至 CPU。

高速缓存有效空间的写入周期, 根据高速缓存写入方式的而不同。

在回写模式, 搜索高速缓存, 有对应地址的数据时, 写入高速缓存。在替换对应地址前, 不对存储器执行实际写入。没有对应地址的数据时, 更新高速缓存。首先, 将作为替换对象的数据保存至内部缓冲器, 然后, 读取包含对应地址数据的 16 字节数据, 并更新对应地址的数据。接着, 执行最初保存的 16 字节数据的回写周期。

在直写模式, 搜索高速缓存, 有对应地址的数据时, 在向高速缓存写入的同时, 通过内部总线执行实际写入。没有对应地址的数据时, 不更新高速缓存, 仅通过内部总线实际写入。

因为总线状态控制器包含 1 段写入缓冲器, 所以在写入周期, 即使芯片外部总线周期未完成, 也可将内部总线用于其他存取。对芯片外部的低速存储器写入后, 读取或写入内部外围模块时, 可不等待低速存储器的写入结束就存取内部外围模块。

读取时，CPU 处于等待状态直到完成读取运行。想要确认已完成对实际器件的数据写入并继续处理时，只需继续虚读相同地址并存取，即可确认写入结束。

从 DMAC 等其他总线主控器存取时，总线状态控制器的写入缓冲器同样有效。因此，双地址 DMA 传送时，不必等待写入周期结束即可启动下一个读取周期。但是，如果 DMA 的源地址与目标地址都在外部存储空间，则需等前一个写入周期结束后才能开始下一个写入周期。

如果写入缓冲器运行过程中更改 BSC 的寄存器，则不能正确写入存取。因此，写入存取后不得立即更改 BSC 的寄存器。需更改时，必须在虚读写入数据后进行。

(3) 内部外围模块的存取

存取内部模块的寄存器时，从内部总线至少需要 2 个外围模块时钟 ($P\phi$) 周期。另一方面，从 CPU 向内部外围寄存器写入时，CPU 不必等待寄存器的写入结束，即可开始执行后续的指令。

举例说明为了降低功耗而转移至软件待机模式的情况。为了实现转移，将 STBCR 寄存器的 STBY 位设定为 1 后，需执行 SLEEP 指令，但执行 SLEEP 指令前，必须虚读 STBCR 寄存器。如果不执行虚读，则 STBY 位置 1 前 CPU 执行 SLEEP 指令，所以不会转移至目标软件待机模式，而转移至睡眠模式。为了等待向 STBY 位的写入，必需虚读 STBCR 寄存器。

如本例子所示，执行后续指令时，如果要通过内部外围寄存器反映更改状况，则必须在寄存器写入指令后虚读该寄存器，然后再执行后续指令。

第 10 章 直接存储器存取控制器 (DMAC)

直接存储器存取控制器 (DMAC) 可替代 CPU 在带 DACK (传送请求的接受信号) 的外部器件、外部存储器、内部存储器、存储器映射的外部器件及内部外围模块之间高速传送数据。

10.1 特点

- 通道数: 8 个通道 CH0~CH7
 仅 CH0~CH3 这 4 个通道可接受外部请求。
- 地址空间: 在体系结构上为 4GB
- 传送数据单位: 字节、字 (2 字节)、长字 (4 字节)、16 字节 (长字 ¥4)
- 最大传送次数: 16,777,216 (24 位) 次
- 地址模式: 可选择单地址模式或双地址模式
- 传送请求:
 可选择外部请求、内部外围模块请求、自动请求 3 种
 以下模块可发行内部外围模块请求:
 SCIF: 8 个源, IIC3: 8 个源, A/D 转换器: 1 个源, MTU2: 5 个源, CMT: 2 个源, USB: 2 个源,
 FLCTL: 2 个源, RCAN-TL1: 2 个源, SSI: 4 个源, SSU: 4 个源
- 总线模式: 可选择周期挪用模式 (普通模式与间歇模式) 或突发模式
- 优先顺序: 可选择通道优先顺序固定模式或轮询模式
- 中断请求: 可在 1/2 数据传送结束或数据传送结束时向 CPU 产生中断请求
 通过 CHCR 的 HE 位、HIE 位, 在初始设定的 1/2 次结束时, 设定对 CPU 的中断。
- 外部请求检测: 可选择 DREQ 输入的低/高电平检测, 或上升沿/下降沿检测
- 传送请求接受信号/传送结束信号: DACK 及 TEND 可设定有效电平。
- DMA 设定的寄存器具备重加载功能, 因此不用重新设定与 DMA 传送相同设定的 DMA 传送就可反复执行。通过在 DMA 传送过程中预先设定重加载寄存器, 可通过不同的设定执行下一次 DMA 传送。
 此重加载功能可对每个通道及重加载寄存器设定 ON/OFF。

DMAC 框图如图 10.1 所示。

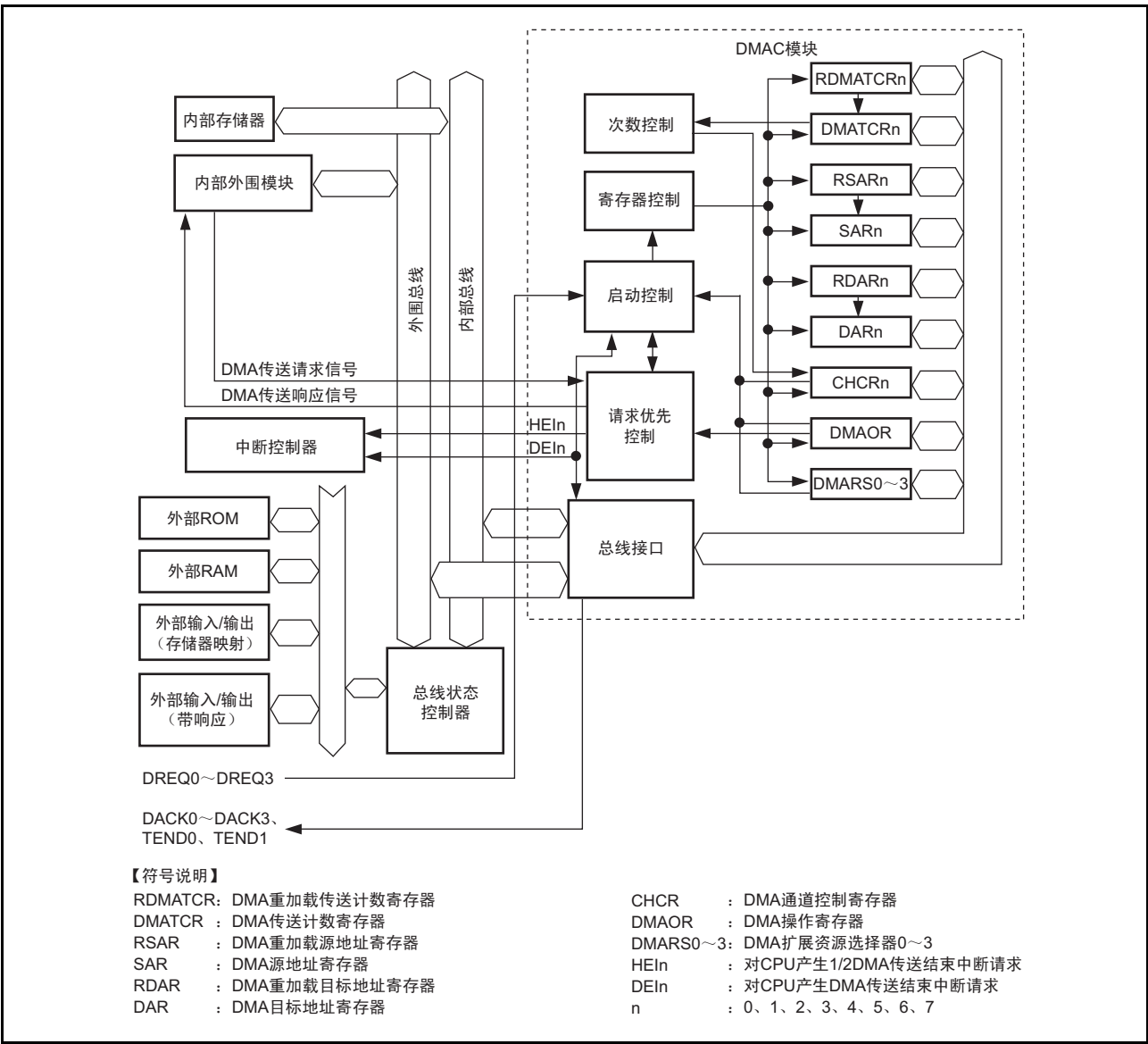


图 10.1 DMAC 框图

10.2 输入 / 输出引脚

与 DMAC 相关的外部引脚如下所示：

连接外部总线的引脚结构如表 10.1 所示。DMAC 有 4 个通道的引脚 (CH0 ~ CH3) 用于外部总线。

表 10.1 连接外部总线的引脚结构

通道	名称	引脚名称	输入 / 输出	功 能
0	DMA 传送请求	DREQ0	输入	输入从外部器件向通道 0 的 DMA 传送请求
	接受 DMA 传送请求	DACK0	输出	输出接受从 DMAC 通道 0 向外部器件的 DMA 传送请求
1	DMA 传送请求	DREQ1	输入	输入从外部器件向通道 1 的 DMA 传送请求
	接受 DMA 传送请求	DACK1	输出	输出接受从 DMAC 通道 1 向外部器件的 DMA 传送请求
2	DMA 传送请求	DREQ2	输入	输入从外部器件向通道 2 的 DMA 传送请求
	接受 DMA 传送请求	DACK2	输出	输出接受从 DMAC 通道 2 向外部器件的 DMA 传送请求
3	DMA 传送请求	DREQ3	输入	输入从外部器件向通道 3 的 DMA 传送请求
	接受 DMA 传送请求	DACK3	输出	输出接受从 DMAC 通道 3 向外部器件的 DMA 传送请求
0	DMA 传送结束	TEND0	输出	输出 DMAC 通道 0 的 DMA 传送结束
1	DMA 传送结束	TEND1	输出	输出 DMAC 通道 1 的 DMA 传送结束

10.3 寄存器说明

DMAC 有以下寄存器。各通道有 4 个控制寄存器和 3 个重加载寄存器，所有通道有 1 个通用的控制寄存器。而且，每 2 个通道有 1 个扩展资源选择器寄存器。各通道寄存器的表示方法：如通道 0 的 SAR 表示为 SAR_0。

表 10.2 寄存器结构

通道	寄存器名称	简称	R/W	初始值	地址	存取长度
0	DMA 源地址寄存器 _0	SAR0	R/W	H'00000000	H'FFFE1000	16、32
	DMA 目标地址寄存器 _0	DAR0	R/W	H'00000000	H'FFFE1004	16、32
	DMA 传送计数寄存器 _0	DMATCR0	R/W	H'00000000	H'FFFE1008	16、32
	DMA 通道控制寄存器 _0	CHCR0	R/W*1	H'00000000	H'FFFE100C	8、16、32
	DMA 重加载源地址寄存器 _0	RSAR0	R/W	H'00000000	H'FFFE1100	16、32
	DMA 重加载目标地址寄存器 _0	RDAR0	R/W	H'00000000	H'FFFE1104	16、32
	DMA 重加载传送计数寄存器 _0	RDMATCR0	R/W	H'00000000	H'FFFE1108	16、32
1	DMA 源地址寄存器 _1	SAR1	R/W	H'00000000	H'FFFE1010	16、32
	DMA 目标地址寄存器 _1	DAR1	R/W	H'00000000	H'FFFE1014	16、32
	DMA 传送计数寄存器 _1	DMATCR1	R/W	H'00000000	H'FFFE1018	16、32
	DMA 通道控制寄存器 _1	CHCR1	R/W*1	H'00000000	H'FFFE101C	8、16、32

通道	寄存器名称	简称	R/W	初始值	地址	存取长度
1	DMA 重加载源地址寄存器 _1	RSAR1	R/W	H'00000000	H'FFFE1110	16、32
	DMA 重加载目标地址寄存器 _1	RDAR1	R/W	H'00000000	H'FFFE1114	16、32
	DMA 重加载传送计数寄存器 _1	RDMATCR1	R/W	H'00000000	H'FFFE1118	16、32
2	DMA 源地址寄存器 _2	SAR2	R/W	H'00000000	H'FFFE1020	16、32
	DMA 目标地址寄存器 _2	DAR2	R/W	H'00000000	H'FFFE1024	16、32
	DMA 传送计数寄存器 _2	DMATCR2	R/W	H'00000000	H'FFFE1028	16、32
	DMA 通道控制寄存器 _2	CHCR2	R/W*1	H'00000000	H'FFFE102C	8、16、32
	DMA 重加载源地址寄存器 _2	RSAR2	R/W	H'00000000	H'FFFE1120	16、32
	DMA 重加载目标地址寄存器 _2	RDAR2	R/W	H'00000000	H'FFFE1124	16、32
	DMA 重加载传送计数寄存器 _2	RDMATCR2	R/W	H'00000000	H'FFFE1128	16、32
3	DMA 源地址寄存器 _3	SAR3	R/W	H'00000000	H'FFFE1030	16、32
	DMA 目标地址寄存器 _3	DAR3	R/W	H'00000000	H'FFFE1034	16、32
	DMA 传送计数寄存器 _3	DMATCR3	R/W	H'00000000	H'FFFE1038	16、32
	DMA 通道控制寄存器 _3	CHCR3	R/W*1	H'00000000	H'FFFE103C	8、16、32
	DMA 重加载源地址寄存器 _3	RSAR3	R/W	H'00000000	H'FFFE1130	16、32
	DMA 重加载目标地址寄存器 _3	RDAR3	R/W	H'00000000	H'FFFE1134	16、32
	DMA 重加载传送计数寄存器 _3	RDMATCR3	R/W	H'00000000	H'FFFE1138	16、32
4	DMA 源地址寄存器 _4	SAR4	R/W	H'00000000	H'FFFE1040	16、32
	DMA 目标地址寄存器 _4	DAR4	R/W	H'00000000	H'FFFE1044	16、32
	DMA 传送计数寄存器 _4	DMATCR4	R/W	H'00000000	H'FFFE1048	16、32
	DMA 通道控制寄存器 _4	CHCR4	R/W*1	H'00000000	H'FFFE104C	8、16、32
	DMA 重加载源地址寄存器 _4	RSAR4	R/W	H'00000000	H'FFFE1140	16、32
	DMA 重加载目标地址寄存器 _4	RDAR4	R/W	H'00000000	H'FFFE1144	16、32
	DMA 重加载传送计数寄存器 _4	RDMATCR4	R/W	H'00000000	H'FFFE1148	16、32
5	DMA 源地址寄存器 _5	SAR5	R/W	H'00000000	H'FFFE1050	16、32
	DMA 目标地址寄存器 _5	DAR5	R/W	H'00000000	H'FFFE1054	16、32
	DMA 传送计数寄存器 _5	DMATCR5	R/W	H'00000000	H'FFFE1058	16、32
	DMA 通道控制寄存器 _5	CHCR5	R/W*1	H'00000000	H'FFFE105C	8、16、32
	DMA 重加载源地址寄存器 _5	RSAR5	R/W	H'00000000	H'FFFE1150	16、32
	DMA 重加载目标地址寄存器 _5	RDAR5	R/W	H'00000000	H'FFFE1154	16、32
	DMA 重加载传送计数寄存器 _5	RDMATCR5	R/W	H'00000000	H'FFFE1158	16、32

通道	寄存器名称	简称	R/W	初始值	地址	存取长度
6	DMA 源地址寄存器 _6	SAR6	R/W	H'00000000	H'FFFE1060	16、32
	DMA 目标地址寄存器 _6	DAR6	R/W	H'00000000	H'FFFE1064	16、32
	DMA 传送计数寄存器 _6	DMATCR6	R/W	H'00000000	H'FFFE1068	16、32
	DMA 通道控制寄存器 _6	CHCR6	R/W*1	H'00000000	H'FFFE106C	8、16、32
	DMA 重加载源地址寄存器 _6	RSAR6	R/W	H'00000000	H'FFFE1160	16、32
	DMA 重加载目标地址寄存器 _6	RDAR6	R/W	H'00000000	H'FFFE1164	16、32
	DMA 重加载传送计数寄存器 _6	RDMATCR6	R/W	H'00000000	H'FFFE1168	16、32
7	DMA 源地址寄存器 _7	SAR7	R/W	H'00000000	H'FFFE1070	16、32
	DMA 目标地址寄存器 _7	DAR7	R/W	H'00000000	H'FFFE1074	16、32
	DMA 传送计数寄存器 _7	DMATCR7	R/W	H'00000000	H'FFFE1078	16、32
	DMA 通道控制寄存器 _7	CHCR7	R/W*1	H'00000000	H'FFFE107C	8、16、32
	DMA 重加载源地址寄存器 _7	RSAR7	R/W	H'00000000	H'FFFE1170	16、32
	DMA 重加载目标地址寄存器 _7	RDAR7	R/W	H'00000000	H'FFFE1174	16、32
	DMA 重加载传送计数寄存器 _7	RDMATCR7	R/W	H'00000000	H'FFFE1178	16、32
通用	DMA 操作寄存器	DMAOR	R/W*2	H'0000	H'FFFE1200	8、16
0/1	DMA 扩展资源选择器 0	DMARS0	R/W	H'0000	H'FFFE1300	16
2/3	DMA 扩展资源选择器 1	DMARS1	R/W	H'0000	H'FFFE1304	16
4/5	DMA 扩展资源选择器 2	DMARS2	R/W	H'0000	H'FFFE1308	16
6/7	DMA 扩展资源选择器 3	DMARS3	R/W	H'0000	H'FFFE130C	16

【注】 *1 为了清除标志，CHCRn 的 HE、TE 位仅可在读取 1 后写入 0。

*2 为了清除标志，DMAOR 的 AE、NMIF 位仅可在读取 1 后写入 0。

10.3.1 DMA 源地址寄存器 (SAR)

SAR 为 32 位可读取 / 写入的寄存器，指定 DMA 传送源的地址。在 DMA 运行过程中，表示下一个传送源地址。在单地址模式，传送源为带 DACK 的外部器件时，忽略 SAR。

传送字 (2 字节)、长字 (4 字节)、16 字节单位的数据时，必须分别指定 2 字节、4 字节、16 字节边界的地址。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

10.3.2 DMA 目标地址寄存器 (DAR)

DAR 为 32 位可读取 / 写入的寄存器，指定 DMA 传送目标地址。在 DMA 运行过程中，表示下一个传送目标地址。在单地址模式，传送目标为带 DACK 的外部器件数据时，忽略 DAR。

传送字 (2 字节)、长字 (4 字节)、16 字节单位的数据时，必须分别指定 2 字节、4 字节、16 字节边界的地址。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

10.3.3 DMA 传送计数寄存器 (DMATCR)

DMATCR 为 32 位可读取 / 写入的寄存器，指定 DMA 传送次数。设定值为 H'00000001 时，传送 1 次；设定值为 H'00FFFFFF 时，传送 16,777,215 次；设定值为 H'00000000 时，传送 16,777,216 次 (最大传送次数)。在 DMA 传送过程中，显示剩余的传送次数。

DMATCR 高 8 位的读取值、写入值总是为 0。

在执行 16 字节传送 1 时，传送 1 次 16 字节 (128 位) 就计数 1 次。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	0	R/W	R/W	R/W	R/W	R/W

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

10.3.4 DMA 通道控制寄存器（CHCR）

CHCR 为 32 位可读取 / 写入的寄存器，控制 DMA 传送模式。

决定外部引脚 DREQ、DACK 规格的位（DO、AM、AL、DL、DS），在通道 0 ~ 3 可读取 / 写入，但在通道 4 ~ 7 对应的位为保留位。

而且，决定外部引脚 TEND 规格的位（TL），在通道 0、1 可读取 / 写入，但在通道 2 ~ 7 对应的位为保留位。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TC	-	RLD SAR	RLD DAR	-	-	-	-	DO	TL	-	TE MASK	HE	HIE	AM	AL
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R/W	R/W	R	R	R	R	R/W	R/W	R	R/W	R/(W)*	R/W	R/W	R/W

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DM[1:0]		SM[1:0]		RS[3:0]			DL	DS	TB	TS[1:0]		IE	TE	DE	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/(W)*	R/W

【注】* 为了清除标志，仅可在读取1后写入0。

位	位名称	初始值	R/W	说 明
31	TC	0	R/W	传送计数模式 设定 1 次传送请求传送 1 次，或传送 DMATCR 的设定次数。本功能仅在内部外围模块请求时有效。设定 TC=0 时，TB 位不得设定为 1（突发模式）。传送请求源设定为 SCIF、IIC3、SSI、FLCTL、SSU 时，不得设定 TC=1。 0: 1 次传送请求传送 1 次 1: 1 次传送请求传送 DMATCR 的设定次数
30	—	0	R	保留位 读取值、写入值总是为 0。
29	RLDSAR	0	R/W	SAR 重加载功能 ON/OFF 设定对 SAR 及 DMATCR 的重加载功能有效（ON）或无效（OFF）。 0: 对 SAR 及 DMATCR 的重加载功能无效（OFF） 1: 对 SAR 及 DMATCR 的重加载功能有效（ON）
28	RLDDAR	0	R/W	DAR 重加载功能 ON/OFF 设定对 DAR 及 DMATCR 的重加载功能有效（ON）或无效（OFF）。 0: 对 DAR 及 DMATCR 的重加载功能无效（OFF） 1: 对 DAR 及 DMATCR 的重加载功能有效（ON）
27 ~ 24	—	均为 0	R	保留位 读取值、写入值总是为 0。
23	DO	0	R/W	DMA 溢出 选择通过溢出 0 检测 DREQ，或溢出 1 检测 DREQ。 本位仅在检测 CHCR_0 ~ 3 电平时有效，在 CHCR_4 ~ 7 为保留位，读取值、写入值总是为 0。 0: 通过溢出 0 检测 DREQ 1: 通过溢出 1 检测 DREQ

位	位名称	初始值	R/W	说 明
22	TL	0	R/W	传送结束电平 指定 TEND 信号为高电平有效或低电平有效。本位仅在 CHCR_0、1 有效，在 CHCR_2 ~ 7 为保留位，读取值、写入值总是为 0。 0: 低电平有效输出 TEND 1: 高电平有效输出 TEND
21	—	0	R	保留位 读取值、写入值总是为 0。
20	TEMASK	0	R/W	TE 设置屏蔽 指示 TE 位置 1 时，DMA 传送不停止。与 SAR 重加载功能或 DAR 重加载功能共同设定本位，可在停止传送请求前的这段时间执行 DMA 传送。检测出自动请求及外部请求的上升 / 下降沿时，忽略本位的设定，TE 位置位时，DMA 传送停止。RLDSAR 位或 RLDDAR 位的任意一个位置 1 时，本功能有效。 0: TE 位置位时，停止 DMA 传送 1: 即使 TE 位置位，仍继续 DMA 传送
19	HE	0	R(W)*	半结束标志 传送次数大于等于传送开始前设置的 DMATCR 值的一半时，HE 位置 1。传送次数未达到传送开始前设置的 DMATCR 的一半时，如果因 NMI 中断、DMA 地址错误结束传送，以及清除 DE 位、DMAOR 的 DME 位后结束传送，则 HE 位不置位。HE 位置位后，如果因 NMI 中断、DMA 地址错误结束传送，以及清除 DE 位、DMAOR 的 DME 位后结束传送，则 HE 位仍保持置位状态。清除 HE 位时，必须在读取 HE 位的 1 后写入 0。 0: 在 DMA 传送过程中或 DMA 传送中断， $DMATCR > (\text{传送前设置的 } DMATCR) / 2$ [清除条件] • 读取 HE 位的 1 后写入 0 1: $DMATCR \leq (\text{传送前设置的 } DMATCR) / 2$
18	HIE	0	R/W	半结束中断允许 指定传送次数为传送开始前设置的 DMATCR 值的一半时，是否向 CPU 请求中断。如果 HIE 位置 1，且 HE 位置位，则向 CPU 请求中断。 0: $DMATCR = (\text{传送前设置的 } DMATCR) / 2$ 时，禁止中断请求 1: $DMATCR = (\text{传送前设置的 } DMATCR) / 2$ 时，允许中断请求
17	AM	0	R/W	响应模式 选择在双地址模式的数据读取周期输出 DACK 及 TEND，或在写入周期输出 DACK 及 TEND。 单地址模式时，与本位的指定无关，总是输出 DACK 及 TEND。 本位仅在 CHCR_0 ~ 3 有效，在 CHCR_4 ~ 7 为保留位，读取值、写入值总是为 0。 0: 在读取周期输出 DACK 及 TEND (双地址模式) 1: 在写入周期输出 DACK 及 TEND (双地址模式)
16	AL	0	R/W	响应电平 指定 DACK 信号为高电平有效或低电平有效。本位仅在 CHCR_0 ~ 3 有效，在 CHCR_4 ~ 7 为保留位，读取值、写入值总是为 0。 0: 低电平有效输出 DACK 1: 高电平有效输出 DACK

位	位名称	初始值	R/W	说 明
15、14	DM[1:0]	00	R/W	目标地址模式 指定 DMA 传送目标地址的增减（在单地址模式，向带 DACK 的外部器件传送时，忽略 DM1 位和 DM0 位）。 00: 目标地址固定 01: 目标地址增加 （字节单位传送时 +1，字单位传送时 +2，长字单位传送时 +4，16 字节单位传送时 +16） 10: 目标地址减少 （字节单位传送时 -1，字单位传送时 -2，长字单位传送时 -4，16 字节单位传送时禁止设定） 11: 禁止设定
13、12	SM[1:0]	00	R/W	源地址模式 指定 DMA 传送源地址的增减（在单地址模式，从带 DACK 的外部器件传送时，忽略 SM1 位和 SM0 位）。 00: 源地址固定 01: 源地址增加 （字节单位传送时 +1，字单位传送时 +2，长字单位传送时 +4，16 字节单位传送时 +16） 10: 源地址减少 （字节单位传送时 -1，字单位传送时 -2，长字单位传送时 -4，16 字节单位传送时禁止设定） 11: 禁止设定
11 ~ 8	RS[3:0]	0000	R/W	资源选择 指定输入至 DMAC 的传送请求源。必须在 DMA 允许位（DE）为 0 的状态更改传送请求源。 0000: 外部请求、双地址模式 0001: 禁止设定 0010: 外部请求、单地址模式 外部地址空间→带 DACK 的外部器件 0011: 外部请求、单地址模式 带 DACK 的外部器件→外部地址空间 0100: 自动请求 0101: 禁止设定 0110: 禁止设定 0111: 禁止设定 1000: DMA 扩展资源选择器 1001: RCAN-TL10 1010: RCAN-TL11 1011: 禁止设定 1100: 禁止设定 1101: 禁止设定 1110: 禁止设定 1111: 禁止设定 【注】外部请求的指定仅在 CHCR_0 ~ 3 有效，在 CHCR_4 ~ 7 即使指定外部请求也不执行任何操作。

位	位名称	初始值	R/W	说 明
7 6	DL DS	0 0	R/W R/W	DREQ 电平 DREQ 边沿选择 选择 DREQ 输入的检测方法和检测电平。 本位仅在 CHCR_0 ~ 3 有效, 在 CHCR_4 ~ 7 为保留位, 读取值、写入值总是为 0。 将传送请求源指定为内部外围模块或自动请求时, 本位无效。 00: 低电平检测 01: 下降沿检测 10: 高电平检测 11: 上升沿检测
5	TB	0	R/W	传送总线模式 选择 DMA 传送的总线模式。但设定 TC=0 时, 不得设定为突发模式 0: 周期挪用模式 1: 突发模式
4、3	TS[1:0]	00	R/W	传送长度 选择 DMA 传送单位。传送源或传送目标是已指定传送长度的内部外围模块寄存器时, 必须选择该传送长度。 00: 字节单位 01: 字 (2 字节) 单位 10: 长字 (4 字节) 单位 11: 16 字节 (长字传送 ×4) 单位
2	IE	0	R/W	中断允许 指定在 DMA 传送结束时是否向 CPU 请求中断。IE 位置 1, 且 TE 位置位时, 向 CPU 请求中断 (DEI)。 0: 禁止中断请求 1: 允许中断请求
1	TE	0	R/(W)*	传送结束标志 DMATCR 的值为 0, 且 DMA 传送结束时, TE 位置 1。DMATCR 的值不为 0 时, 如果因 NMI 中断、DMA 地址错误结束传送、以及清除 DE 位、DMA 操作寄存器 (DMAOR) 的 DME 位后结束传送, 则 TE 位不置位。清除 TE 位时, 必须在读取 TE 位的 1 后写入 0。 TEMASK 位为 0, 且 TE 位置位时, 即使 DE 位置 1 也不允许传送。 0: DMA 传送过程中或 DMA 传送中断 [清除条件] • 读取 TE 位的 1 后写入 0 1: (通过 DMATCR=0) DMA 传送结束
0	DE	0	R/W	DMA 允许 允许或禁止 DMA 传送。在自动请求模式, 如果 DE 位和 DMAOR 的 DME 位置 1 时, 则开始传送。但 TE 位、DMAOR 的 NMIF 位、AE 位必须全部为 0。在外部请求、外围模块请求, DE 位和 DME 位置 1 后, 还存在相应器件或相应外围模块的 DMA 传送请求时, 开始传送。检测外部请求的低电平或高电平, 以及外围模块请求中, 如果 TEMASK 位为 1, NMIF 位及 AE 位必须为 0。如果 TEMASK 位为 0, TE 位必须为 0。检测外部请求的上升沿或下降沿时, 与自动请求模式相同, TE 位、NMIF 位、AE 位必须全部为 0。清除 DE 位时可中断传送。 0: 禁止 DMA 传送 1: 允许 DMA 传送

【注】 * 为了清除标志, 仅可在读取 1 后写入 0。

10.3.5 DMA 重加载源地址寄存器 (RSAR)

RSAR 为 32 位可读取 / 写入的寄存器。

设定 SAR 重加载功能为 ON、当前 DMA 传送结束时，RSAR 的内容写入源地址寄存器 (SAR)。此时，在 DMA 传送过程中预先设定，可预置下一次 DMA 传送的设定。设定 SAR 重加载功能为 OFF 时，对运行无任何影响。

传送字 (2 字节)、长字 (4 字节)、16 字节数据时，必须分别指定 2 字节、4 字节、16 字节边界的地
址。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

10.3.6 DMA 重加载目标地址寄存器 (RDAR)

RDAR 为 32 位可读取 / 写入的寄存器。

设定 DAR 重加载功能为 ON、当前 DMA 传送结束时，RDAR 的内容写入目标地址寄存器 (DAR)。此时，在 DMA 传送过程中预先设定，可预置下一次 DMA 传送的设定。设定 DAR 重加载功能为 OFF 时，对运行无任何影响。

传送字 (2 字节)、长字 (4 字节)、16 字节数据时，必须分别指定 2 字节、4 字节、16 字节边界的地
址。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

10.3.7 DMA 重加载传送计数寄存器 (RDMATCR)

RDMATCR 为 32 位可读取 / 写入的寄存器。

设定 SAR 或 DAR 重加载功能为 ON、在当前 DMA 传送结束时，RDMATCR 的内容写入传送计数寄存器 (DMATCR)。此时，在 DMA 传送过程中预先设定，可预置下一次 DMA 传送的设定。设定 SAR 或 DAR 重加载功能为 OFF 时，对运行无任何影响。

RDMATCR 的高 8 位的读取值、写入值总是为 0。

与 DMATCR 同样，设定值为 H'00000001 时，传送次数为 1 次；设定值为 H'00FFFFFF 时，传送次数为 16,777,215 次；设定值为 H'00000000 时，传送次数为 16,777,216 次（最大传送次数）。16 字节传送时，传送 1 次 16 字节（128 位）就计数 1 次。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

10.3.8 DMA 操作寄存器 (DMAOR)

DMAOR 为 16 位可读取 / 写入的寄存器，指定 DMA 传送时通道的优先顺序，也表示 DMA 的传送状态。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	CMS[1:0]	-	-	PR[1:0]	-	-	-	-	-	-	AE	NMIF	DME	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R	R	R	R/(W)*R/(W)*	R/W	R/W

【注】* 为了清除标志，仅可在读取1后写入0。

位	位名称	初始值	R/W	说 明
15、14	—	均为 0	R	保留位 读取值、写入值总是为 0。
13、12	CMS[1:0]	00	R/W	选择周期挪用模式 周期挪用模式时，选择普通模式或间歇模式。 为了使间歇模式有效，所有通道的总线模式必须为周期挪用模式。 00: 普通模式 01: 禁止设定 10: 间歇模式 16 B ϕ 时钟的每 16 个时钟执行 1 次 DMA 传送 11: 轮询模式 64 B ϕ 时钟的每 64 个时钟执行 1 次 DMA 传送
11、10	—	均为 0	R	保留位 读取值、写入值总是为 0。
9、8	PR[1:0]	00	R/W	优先级模式 多个通道同时产生传送请求时，本位决定要执行通道的优先顺序。 00: 固定模式 1: CH0 > CH1 > CH2 > CH3 > CH4 > CH5 > CH6 > CH7 01: 固定模式 2: CH0 > CH4 > CH1 > CH5 > CH2 > CH6 > CH3 > CH7 10: 禁止设定 11: 轮询模式 (仅 CH0 ~ CH3 轮询)
7 ~ 3	—	均为 0	R	保留位 读取值、写入值总是为 0。
2	AE	0	R/(W)*	地址错误标志 表示产生 DMAC 引起的地址错误。AE 位置位时，即使将 CHCR 的 DE 位和 DMAOR 的 DME 位置 1，也不允许 DMA 传送。清除 AE 位时，必须在读取 AE 位的 1 后写入 0。 0: 无 DMAC 引起的地址错误 1: 产生 DMAC 引起的地址错误 [清除条件] • 读取 AE 位的 1 后写入 0
1	NMIF	0	R/(W)*	NMI 标志 表示产生 NMI 中断。NMIF 位置位时，即使将 CHCR 的 DE 位和 DMAOR 的 DME 位置 1，也不允许 DMA 传送。清除 NMIF 位时，必须在读取 NMIF 位的 1 后写入 0。 可在 DMA 传送完成 1 个传送单位时输入 NMI。DMAC 不运行时，即使输入 NMI 中断，NMIF 位仍置 1。 0: 无 NMI 中断 1: 产生 NMI 中断 [清除条件] • 读取 NMIF 位的 1 后写入 0
0	DME	0	R/W	DMA 主控允许 允许 / 禁止所有通道的 DMA 传送。DME 位和 CHCR 的 DE 位置 1 时，允许 DMA 传送。但执行传送的通道 CHCR 的 TE 位和 DMAOR 的 NMIF 位、AE 位必须全部为 0。清除 DME 位时，中断所有通道的 DMA 传送。 0: 禁止所有通道的 DMA 传送 1: 允许所有通道的 DMA 传送

【注】 * 为了清除标志，仅可在读取 1 后写入 0。

结束一个传送后，更改优先权模式位的设定时，初始化优先顺序。

例如，在固定模式 2 重新设定时，优先顺序为 CH0 > CH4 > CH1 > CH5 > CH2 > CH6 > CH3 > CH7；在固定模式 1 重新设定时，优先顺序为 CH0 > CH1 > CH2 > CH3 > CH4 > CH5 > CH6 > CH7；重新设定为轮询模式时，传送结束通道被复位。

优先权模式位的各模式（模式 0 ~ 2）优先顺序的转移如表 10.3 所示。各模式根据传送结束的通道不同，下一个接受请求的通道的优先顺序也各异，最多有 3 种变化形式。

例如，传送结束的通道为 CH1 时，下一个接受传送请求的通道优先顺序为 CH2 > CH3 > CH0 > CH1 > CH4 > CH5 > CH6 > CH7。传送结束的通道为 CH4、CH5、CH6、CH7 时，不为轮询对象，因此即使通道 4、通道 5、通道 6 或通道 7 传送结束，优先顺序也不变。

产生地址错误时的 DMAC 的内部处理运行如下：

- 不产生地址错误时：Read（传送源→DMAC 内容）→Write（DMAC 内容→传送目标）
- 在源地址产生地址错误：Nop→Nop
- 在目标地址产生地址错误：Read→Nop

表 10.3 优先权模式位的组合

模式	传送结束	优先权模式位		传送结束后的优先顺序：高 ←→ 低							
	CHNo.	PR[1]	PR[0]	优先顺序 0	优先顺序 1	优先顺序 2	优先顺序 3	优先顺序 4	优先顺序 5	优先顺序 6	优先顺序 7
模式 0 (固定模式 1)	任意	0	0	CH0	CH1	CH2	CH3	CH4	CH5	CH6	CH7
模式 1 (固定模式 2)	任意	0	1	CH0	CH4	CH1	CH5	CH2	CH6	CH3	CH7
模式 2 (轮询模式)	CH0	1	1	CH1	CH2	CH3	CH0	CH4	CH5	CH6	CH7
	CH1	1	1	CH2	CH3	CH0	CH1	CH4	CH5	CH6	CH7
	CH2	1	1	CH3	CH0	CH1	CH2	CH4	CH5	CH6	CH7
	CH3	1	1	CH0	CH1	CH2	CH3	CH4	CH5	CH6	CH7
	CH4	1	1	CH0	CH1	CH2	CH3	CH4	CH5	CH6	CH7
	CH5	1	1	CH0	CH1	CH2	CH3	CH4	CH5	CH6	CH7
	CH6	1	1	CH0	CH1	CH2	CH3	CH4	CH5	CH6	CH7
	CH7	1	1	CH0	CH1	CH2	CH3	CH4	CH5	CH6	CH7

10.3.9 DMA 扩展资源选择器 0 ~ 3 (DMARS0 ~ DMARS3)

DMARS 为 16 位可读取 / 写入的寄存器，在每个通道指定外围模块的 DMA 传送请求源。DMARS0 设定通道 0、1，DMARS1 设定通道 2、3，DMARS2 设定通道 4、5，DMARS3 设定通道 6、7。可设定的组合如表 10.4 所示。

本寄存器可对以下启动源设定接受传送请求。

SCIF: 8 个源、IIC3: 8 个源、A/D 转换器: 1 个源、MTU2: 5 个源、CMT: 2 个源、USB: 2 个源、FLCTL: 2 个源、SSI: 4 个源、SSU: 4 个源

仅接受 RCAN-TL1 的 2 个源的传送请求，可在 DMA 通道控制寄存器的 RS[3:0] 设定，无需设定 DMA 扩展资源选择器。

- DMARS0

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CH1 MID[5:0]						CH1 RID[1:0]		CH0 MID[5:0]						CH0 RID[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- DMARS1

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CH3 MID[5:0]						CH3 RID[1:0]		CH2 MID[5:0]						CH2 RID[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- DMARS2

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CH5 MID[5:0]						CH5 RID[1:0]		CH4 MID[5:0]						CH4 RID[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- DMARS3

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CH7 MID[5:0]						CH7 RID[1:0]		CH6 MID[5:0]						CH6 RID[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

各模块的传送请求设定 MID、RID，详情如下表所示：

表 10.4 DMARS 的设定

外围模块	1 个通道的设定值 ({MID,RID})	MID	RID	功能
USB_0	H'03	B'000000	B'11	—
USB_1	H'07	B'000001	B'11	—
SSI_0	H'23	B'001000	B'11	—
SSI_1	H'27	B'001001	B'11	—
SSI_2	H'2B	B'001010	B'11	—
SSI_3	H'2F	B'001011	B'11	—
SSU_0	H'51	B'010100	B'01	发送
	H'52		B'10	接收
SSU_1	H'55	B'010101	B'01	发送
	H'56		B'10	接收
IIC3_0	H'61	B'011000	B'01	发送
	H'62		B'10	接收
IIC3_1	H'65	B'011001	B'01	发送
	H'66		B'10	接收
IIC3_2	H'69	B'011010	B'01	发送
	H'6A		B'10	接收
IIC3_3	H'6D	B'011011	B'01	发送
	H'6E		B'10	接收
SCIF_0	H'81	B'100000	B'01	发送
	H'82		B'10	接收
SCIF_1	H'85	B'100001	B'01	发送
	H'86		B'10	接收
SCIF_2	H'89	B'100010	B'01	发送
	H'8A		B'10	接收
SCIF_3	H'8D	B'100011	B'01	发送
	H'8E		B'10	接收
A/D 转换器_0	H'B3	B'101100	B'11	—
FLCTL_0	H'BB	B'101110	B'11	发送 / 接收数据
FLCTL_1	H'BF	B'101111	B'11	发送 / 接收管理码
MTU2_0	H'E3	B'111000	B'11	—
MTU2_1	H'E7	B'111001	B'11	—
MTU2_2	H'EB	B'111010	B'11	—
MTU2_3	H'EF	B'111011	B'11	—
MTU2_4	H'F3	B'111100	B'11	—
CMT_0	H'FB	B'111110	B'11	—
CMT_1	H'FF	B'111111	B'11	—

设定除表 10.4 之外的 MID 或 RID 时，无法保证运行。DMARS 寄存器的传送请求仅在 CHCR_0 ~ 7 寄存器的资源选择位 (RS[3:0]) = B'1000 时有效，B'1000 以外时，即使设定 DMARS 也不作为传送请求源接受。

10.4 运行说明

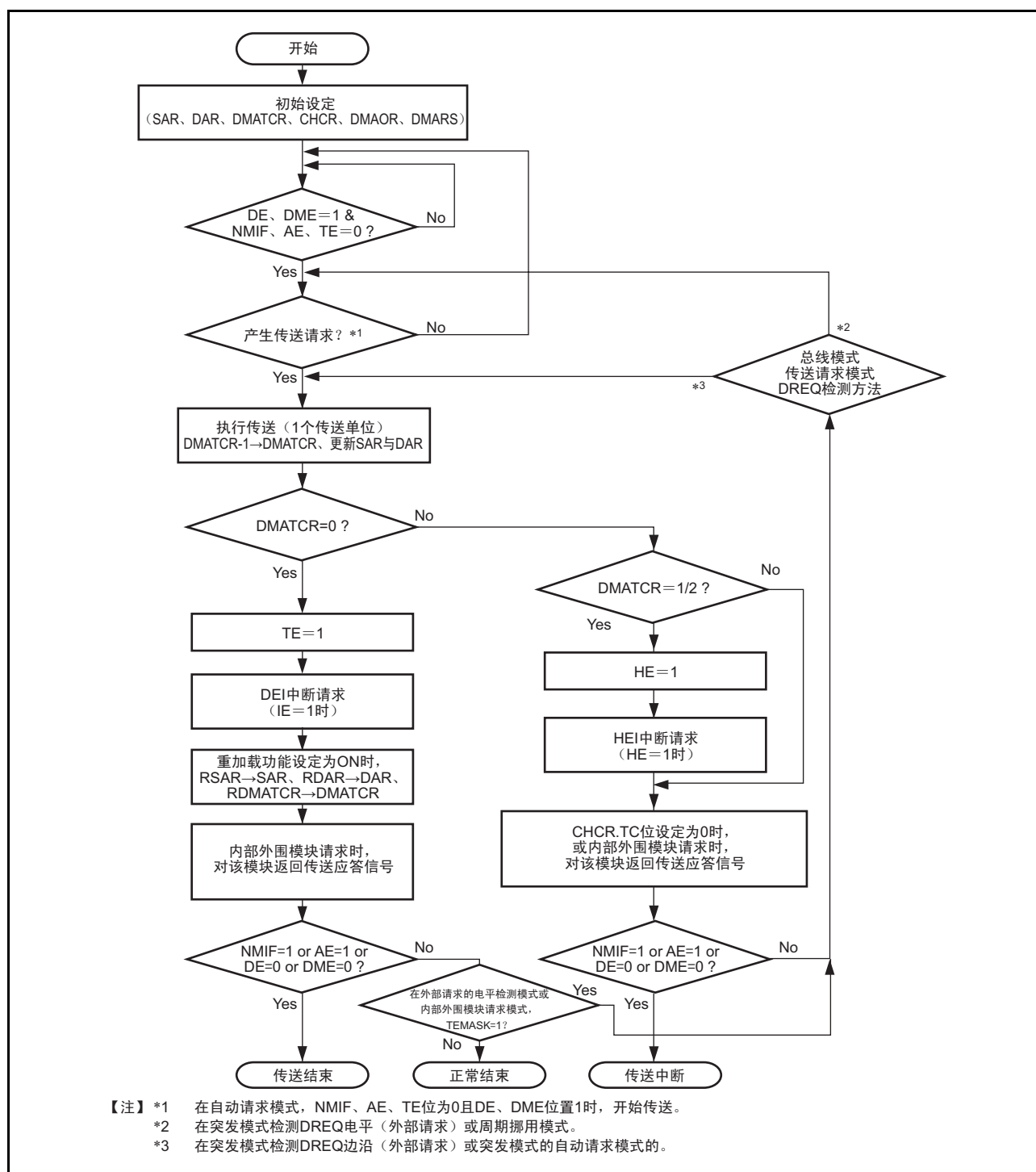
有 DMA 传送请求时，DMAC 按照规定的通道优先顺序开始传送，满足传送结束条件后，DMAC 结束传送。传送请求有自动请求、外部请求和内部外围模块请求 3 种模式。总线模式可选择突发模式或周期挪用模式。

10.4.1 传送流程

对 DMA 源地址寄存器 (SAR)、DMA 目标地址寄存器 (DAR)、DMA 传送计数寄存器 (DMATCR)、DMA 通道控制寄存器 (CHCR)、DMA 操作寄存器 (DMAOR)、3 个重加载寄存器 (RSAR、RDAR、RDMATCR) 和 DMA 扩展资源选择 (DMARS) 设定目标传送条件后，DMAC 按照以下顺序传送数据：

1. 检查是否为传送允许状态 (DE=1、DME=1、TEMASK=0 且 TE=0 或 TEMASK=1、AE=0、NMIF=0)。
2. 在传送允许状态产生传送请求时，传送 1 个传送单位的数据 (取决于 TS[1:0] 位的设定)。在自动请求模式，DE 位和 DME 位置 1 后，自动开始传送。每执行 1 次传送，DMATCR 的值递减 1。具体的传送流程因地址模式和总线模式的而不同。
3. 传送超过指定次数一半 (DMATCR 的值为初始值的 1/2) 时，如果 CHCR 的 HIE 位置 1，则在 CPU 产生 HEI 中断。
4. TEMASK=0 时，如果结束指定次数的传送 (DMATCR 值为 0)，则传送正常结束。此时如果 CHCR 的 IE 位置 1，则在 CPU 产生 DEI 中断。TEMASK=1 时，如果 DMATCR 的值为 0，则设置 TE=1 后，将指定的 RSAR、RDAR、RDMATCR 的值重加载到 SAR、DAR、DMATCR，继续传送直到没有传送请求。
5. 产生 DMAC 引起的地址错误或 NMI 中断时，中断传送。即使 CHCR 的 DE 位或 DMAOR 的 DME 位清 0，也中断传送。

上述步骤的流程如图 10.2 所示。



- 【注】*1 在自动请求模式，NMIF、AE、TE位为0且DE、DME位置1时，开始传送。
 *2 在突发模式检测DREQ电平（外部请求）或周期挪用模式。
 *3 在突发模式检测DREQ边沿（外部请求）或突发模式的自动请求模式的。

图 10.2 DMA 传送流程图

10.4.2 DMA 传送请求

DMA 传送请求基本的使用方法是使 DMA 传送请求在数据的传送源或传送目标产生，但在非传送源、非传送目标的外部器件、内部外围模块亦可产生。

传送请求有自动请求、外部请求和内部外围模块请求 3 种，由 CHCR_0 ~ CHCR_7 的 RS[3:0] 位及 DMARS0、DMARS1、DMARS2、DMARS3 寄存器选择传送请求。

(1) 自动请求模式

如存储器之间的传送，或不可产生传送请求的内部外围模块与存储器之间的传送一样，没有来自外部的传送请求信号时，自动请求模式在 DMAC 内部自动产生传送请求信号。CHCR_0 ~ CHCR_7 的 DE 位及 DMAOR 的 DME 位置 1 时，开始传送。但 CHCR_0 ~ CHCR_7 的 TE 位、DMAOR 的 AE 位、NMIF 位必须全部为 0。

(2) 外部请求模式

外部请求模式为通过 LSI 外部器件的传送请求信号 (DREQ0 ~ DREQ3) 开始传送的模式。根据系统，从表 10.5 所示的模式中选择 1 种使用。允许 DMA 传送 (电平检测时，DE=1、DME=1、TEMASK=0 且 TE=0 或 TEMASK=1、AE=0、NMIF=0；边沿检测时，DE=1、DME=1、TE=0、AE=0、NMIF=0) 时，如果输入 DREQ，则开始 DMA 传送。

表 10.5 由 RS 位选择外部请求模式

RS[3]	RS[2]	RS[1]	RS[0]	地址模式	传送源	传送目标
0	0	0	0	双地址模式	任意	任意
0	0	1	0	单地址模式	外部存储器或 存储器映射的外部器件	带 DACK 的外部器件
			1		带 DACK 的外部器件	外部存储器或 存储器映射的外部器件

如表 10.6 所示，通过 CHCR_0 ~ CHCR_3 的 DL 位和 DS 位，选择通过边沿或电平检测 DREQ。传送请求源无需是数据的传送源或传送目标。为上升沿或下降沿检测且突发模式时，1 次传送请求，可持续传送直至 DMATCR=0；为周期挪用模式时，1 次传送请求，执行 1 次传送。

表 10.6 由 DL 或 DS 位选择外部请求检测

CHCR		外部请求检测方法
DL	DS	
0	0	低电平检测
	1	下降沿检测
1	0	高电平检测
	1	上升沿检测

接受 DREQ 后，DREQ 引脚为不可接受请求状态（死区）。对已接受的 DREQ 输出响应 DACK 后，DREQ 引脚可再次接受请求。

通过电平检测使用 DREQ 时，根据输出 DACK 后到检测出下一个 DREQ 的时序不同，有以下两种情况：执行与请求次数相同的传送后中断（溢出 0）；执行比请求多 1 次的传送后中断（溢出 1）。由 CHCR 的 DO 位选择溢出 0 或溢出 1。

表 10.7 由 DO 位选择外部请求检测

CHCR 的 DO 位	外部请求
0	溢出 0
1	溢出 1

(3) 内部外围模块请求模式

在内部外围模块请求模式，根据内部外围模块的 DMA 传送请求信号执行传送。

内部外围模块对 DMAC 发送的 DMA 传送请求信号如表 10.8 所示。

选择内部外围模块请求模式时，如果为 DMA 传送允许状态（DE=1、DME=1、TEMASK=0 且 TE=0 或 TEMASK=1、AE=0、NMIF=0），则根据传送请求信号执行传送。

内部外围模块请求模式时，可能为固定的传送源、传送目标，详情参照表 10.8。

表 10.8 由 RS3 ~ RS0 位选择内部外围模块请求模式

CHCR	DMARS		DMA 传送	DMA 传送请求信号	传送源	传送目标	总线模式
RS[3:0]	MID	RID	请求源				
1001	任意	任意	RCAN-TL10 接收	DM0（接收完成）	RCAN0 MB0	任意	周期挪用
1010	任意	任意	RCAN-TL11 接收	DM0（接收完成）	RCAN1 MB0	任意	
1000	000000	11	USB	USB_DMA0（接收 FIFO 满）	D0FIFO	任意	周期挪用 / 突发
				USB_DMA0（传送 FIFO 空）	任意	D0FIFO	
	000001	11	USB	USB_DMA1（接收 FIFO 满）	D0FIFO	任意	
				USB_DMA1（传送 FIFO 空）	任意	D1FIFO	
	001000	11	SSI_0	DMA0（发送模式）	任意	SSIRD0R0	周期挪用
				DMA0（接收模式）	SSIRD0R0	任意	
	001001	11	SSI_1	DMA1（发送模式）	任意	SSITDR0	
				DMA1（接收模式）	SSIRD1R1	任意	
	001010	11	SSI_2	DMA2（发送模式）	任意	SSITDR2	
				DMA2（接收模式）	SSIRD2R2	任意	
	001011	11	SSI_3	DMA3（发送模式）	任意	SSITDR3	
				DMA3（接收模式）	SSIRD3R3	任意	

CHCR RS[3:0]	DMARS		DMA 传送 请求源	DMA 传送请求信号	传送源	传送目标	总线模式
	MID	RID					
1000	010100	01	SSU_0 发送	SSTXI0 (发送空或发送结束)	任意	SSTDRO ~ 3	周期挪用
		10	SSU_0 接收	SSRXI0 (接收满)	SSRDR0 ~ 3	任意	
	010101	01	SSU_1 发送	SSTXI1 (发送空或发送结束)	任意	SSTDRO ~ 3	
		10	SSU_1 接收	SSRXI1 (接收满)	SSRDR0 ~ 3	任意	
	011000	01	IIC3_0 发送	TXI0 (发送数据空)	任意	ICDRT0	
		10	IIC3_0 接收	RXI0 (接收数据满)	ICDRR0	任意	
	011001	01	IIC3_1 发送	TXI1 (发送数据空)	任意	ICDRT1	
		10	IIC3_1 接收	RXI1 (接收数据满)	ICDRR1	任意	
	011010	01	IIC3_2 发送	TXI2 (发送数据空)	任意	ICDRT2	
		10	IIC3_2 接收	RXI2 (接收数据满)	ICDRR2	任意	
	011011	01	IIC3_3 发送	TXI3 (发送数据空)	任意	ICDRT3	
		10	IIC3_3 接收	RXI3 (接收数据满)	ICDRR3	任意	
	100000	01	SCIF_0 发送	TXI0 (发送 FIFO 数据空)	任意	SCFTDR_0	
		10	SCIF_0 接收	RXI0 (接收 FIFO 数据满)	SCFRDR_0	任意	
	100001	01	SCIF_1 发送	TXI1 (发送 FIFO 数据空)	任意	SCFTDR_1	
		10	SCIF_1 接收	RXI1 (接收 FIFO 数据满)	SCFRDR_1	任意	
	100010	01	SCIF_2 发送	TXI2 (发送 FIFO 数据空)	任意	SCFTDR_2	
		10	SCIF_2 接收	RXI2 (接收 FIFO 数据满)	SCFRDR_2	任意	
	100011	01	SCIF_3 发送	TXI3 (发送 FIFO 数据空)	任意	SCFTDR_3	
		10	SCIF_3 接收	RXI3 (接收 FIFO 数据满)	SCFRDR_3	任意	
	101100	11	A/D 转换器	ADI (A/D 转换结束)	ADDR	任意	
	101110	11	FLCTL 数据部发送	发送 FIFO 数据空	任意	FLDTFIFO	
			FLCTL 数据部接收	接收 FIFO 数据满	FLDTFIFO	任意	
	101111	11	FLCTL 管理码部发送	发送 FIFO 数据空	任意	FLECFIFO	
			FLCTL 管理码部接收	接收 FIFO 数据满	FLECFIFO	任意	
	111000	11	MTU2_0	TGI0A (输入捕捉 / 比较匹配)	任意	任意	周期挪用 / 突发
	111001	11	MTU2_1	TGI1A (输入捕捉 / 比较匹配)	任意	任意	
	111010	11	MTU2_2	TGI2A (输入捕捉 / 比较匹配)	任意	任意	
	111011	11	MTU2_3	TGI3A (输入捕捉 / 比较匹配)	任意	任意	
	111100	11	MTU2_4	TGI4A (输入捕捉 / 比较匹配)	任意	任意	
	111110	11	CMT_0	CMIO (比较匹配)	任意	任意	
	111111	11	CMT_1	CMI1 (比较匹配)	任意	任意	

10.4.3 通道的优先顺序

多个通道同时产生传送请求时，DMAC 按照规定的优先顺序传送。可从固定模式 1、固定模式 2 和轮询模式这 3 种模式选择通道的优先顺序，由 DMAOR 的 PR1、PR0 位选择模式。

(1) 固定模式

在固定模式 1 和 2，通道优先顺序不变。

各模式的优先顺序如下：

- 固定模式 1: CH0 > CH1 > CH2 > CH3 > CH4 > CH5 > CH6 > CH7
- 固定模式 2: CH0 > CH4 > CH1 > CH5 > CH2 > CH6 > CH3 > CH7

由 DMAOR 的 PR1、PR0 位选择。

(2) 轮询模式

在轮询模式，每个通道的 1 个传送单位（字节、字、长字或 16 字节单位）的每次传送结束后，更改优先顺序，使该通道的优先顺序在轮询对象通道内变为最低。轮询调度的对象通道仅有 CH0 ~ CH3 的 4 个通道。此外的通道即使在轮询模式也不更改该优先顺序。此运行如图 10.3 所示。复位后轮询模式的优先顺序为 CH0 > CH1 > CH2 > CH3 > CH4 > CH5 > CH6 > CH7。

指定轮询模式时，不可使周期挪用模式和突发模式同时存在于多个通道的总线模式。

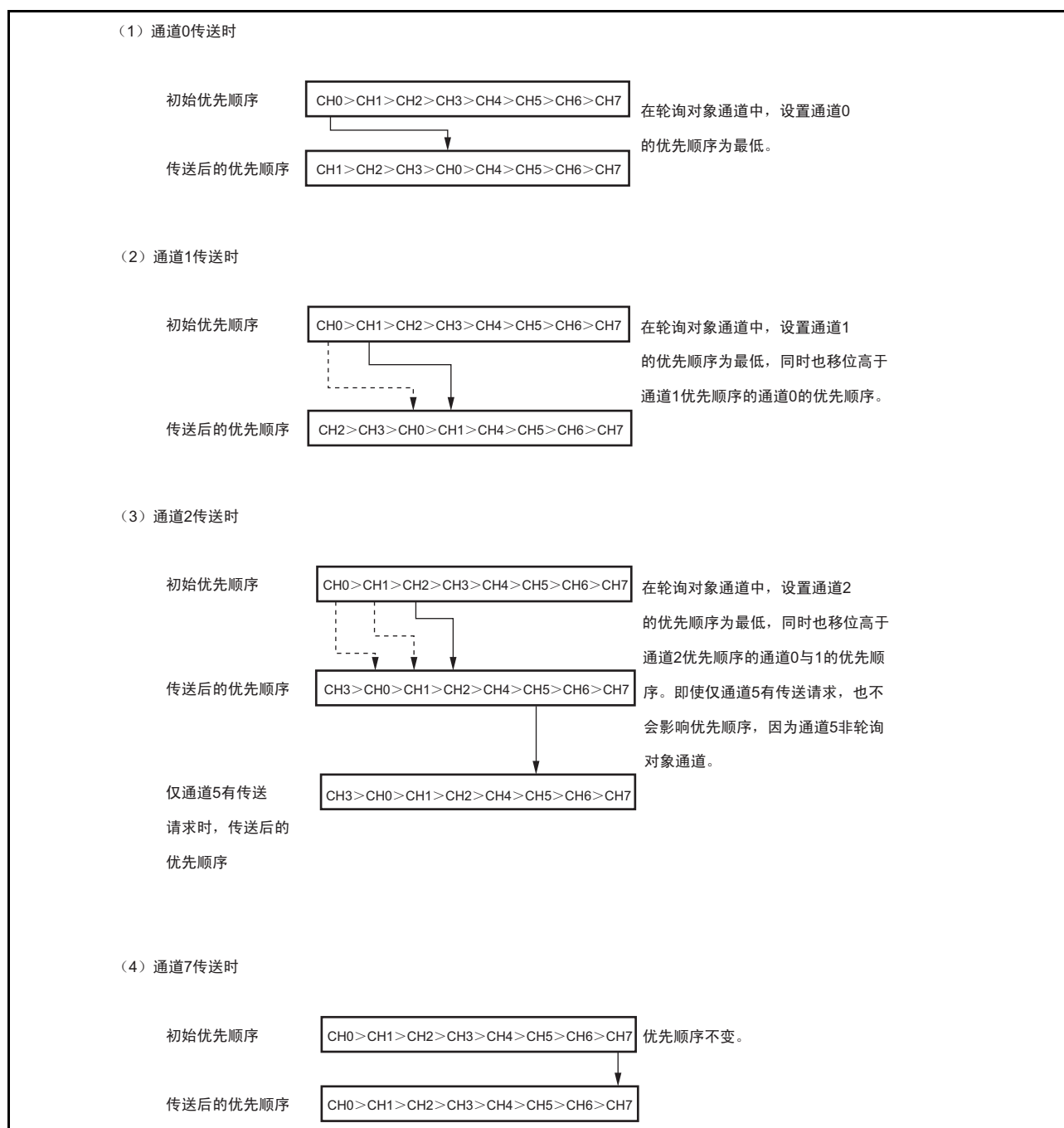


图 10.3 轮询模式

通道 0 和通道 3 同时产生传送请求，且在通道 0 传送过程中产生通道 1 的传送请求时，通道优先顺序的变化如图 10.4 所示。此时，DMAC 的运行如下：

1. 通道 0 和通道 3 同时产生传送请求。
2. 通道 0 的优先顺序高于通道 3，因此开始通道 0 的传送（通道 3 等待传送）。
3. 通道 0 传送过程中，通道 1 产生传送请求（通道 1 和通道 3 均等待传送）。
4. 通道 0 传送结束时，通道 0 的优先顺序在轮询对象通道内变为最低。
5. 此时通道 1 的优先顺序高于通道 3，因此开始通道 1 的传送（通道 3 等待传送）。
6. 通道 1 的传送结束时，通道 1 的优先顺序在轮询对象通道内变为最低。
7. 开始通道 3 的传送。
8. 通道 3 传送结束时，通道 3 的优先顺序在轮询对象通道内变为最低，同时也降低通道 2 的优先顺序。

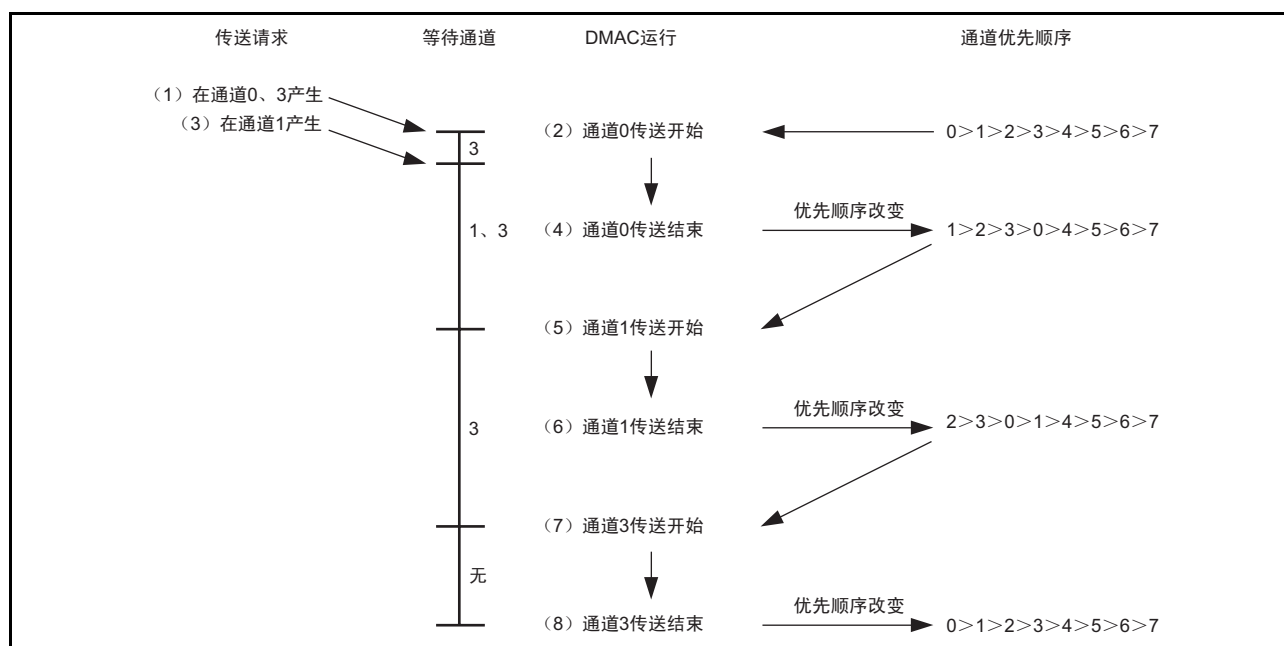


图 10.4 轮询模式的通道优先顺序

10.4.4 DMA 传送的种类

根据存取传送源和传送目标总线周期次数，DMA 传送分为单地址模式传送和双地址模式传送。具体的传送时序因总线模式而不同，总线模式有周期挪用模式和突发模式。DMAC 可支持的传送如表 10.9 所示。

表 10.9 可支持的 DMA 传送

传送源	传送目标				
	带 DACK 的外部器件	外部存储器	存储器映射的外部器件	内部外围模块	内部存储器
带 DACK 的外部器件	否	双、单	双、单	否	否
外部存储器	双、单	双	双	双	双
存储器映射的外部器件	双、单	双	双	双	双
内部外围模块	否	双	双	双	双
内部存储器	否	双	双	双	双

- 【注】
1. 双：双地址模式
 2. 单：单地址模式
 3. 内部外围模块，可仅对允许长字存取寄存器传送 16 字节。

(1) 地址模式

(a) 双地址模式

双地址模式用于通过地址存取（选择）传送源和传送目标。传送源和传送目标可在外部也可在内部。在此模式，DMAC 在读取周期存取传送源，在写入周期存取传送目标，在 2 个总线周期执行传送。此时，传送数据暂时保存在 DMAC。例如，图 10.5 所示的外部存储器之间的传送，在读取周期从一个外部存储器将数据读取至 DMAC，接着在写入周期，将此数据写入另一个外部存储器。

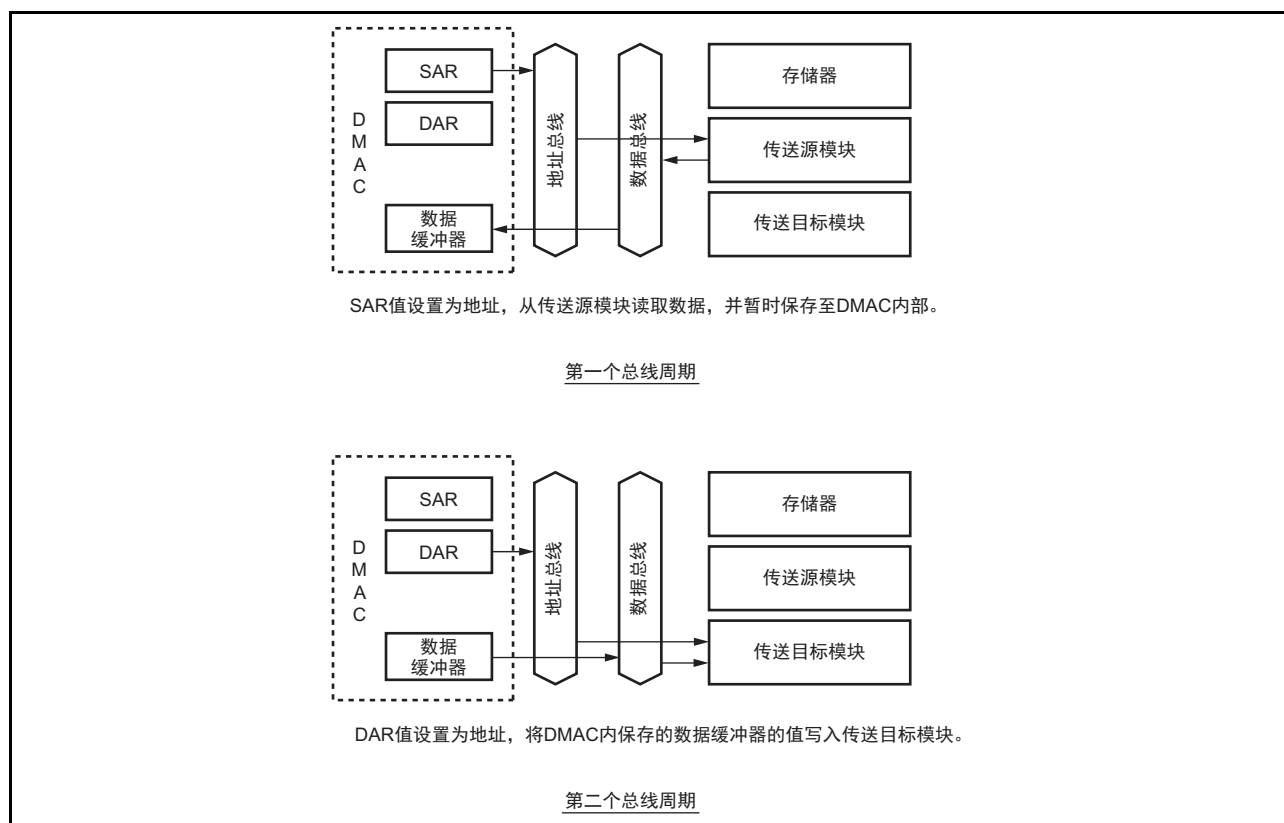


图 10.5 双地址模式的数据流程

传送请求可为自动请求、外部请求或内部外围模块请求。在双地址模式，可在读取周期或写入周期输出 DACK，并可通过 CHCR 的 AM 位设定在读取周期或写入周期输出。

双地址模式的 DMA 传送时序例如图 10.6 所示。

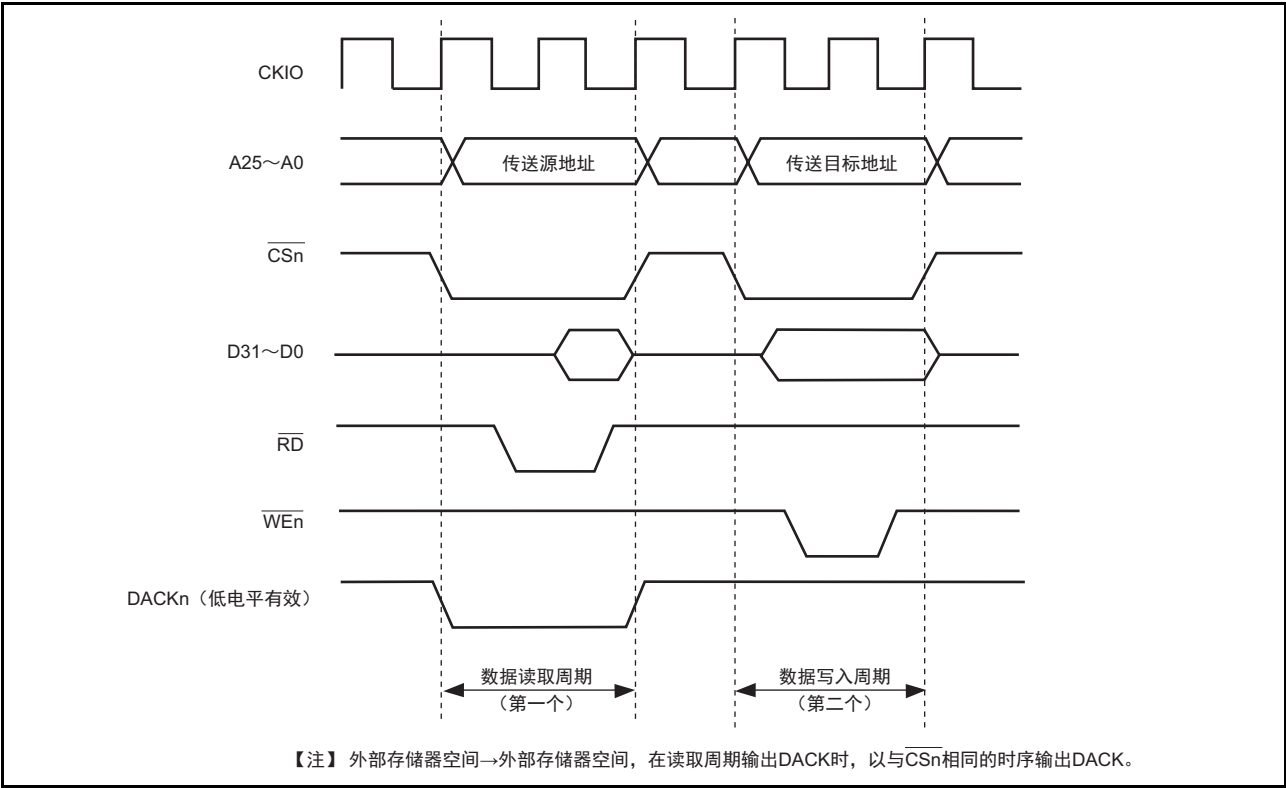


图 10.6 双地址模式的 DMA 传送时序
(传送源: 普通存储器, 传送目标: 普通存储器)

(b) 单地址模式

单地址模式的传送源和传送目标均在外部，通过 DACK 信号存取（选择）其中任意一个，通过地址存取另一个。在此模式，DMAC 在将传送请求的接受信号 DACK 输出至一个外部器件后，在存取的同时将地址输出至传送目标，在 1 个总线周期执行 DMA 传送。例如，图 10.7 所示的外部存储器和带 DACK 的外部器件之间的传送，在外部器件将数据输出到数据总线相同的总线周期，将此数据写入外部存储器。

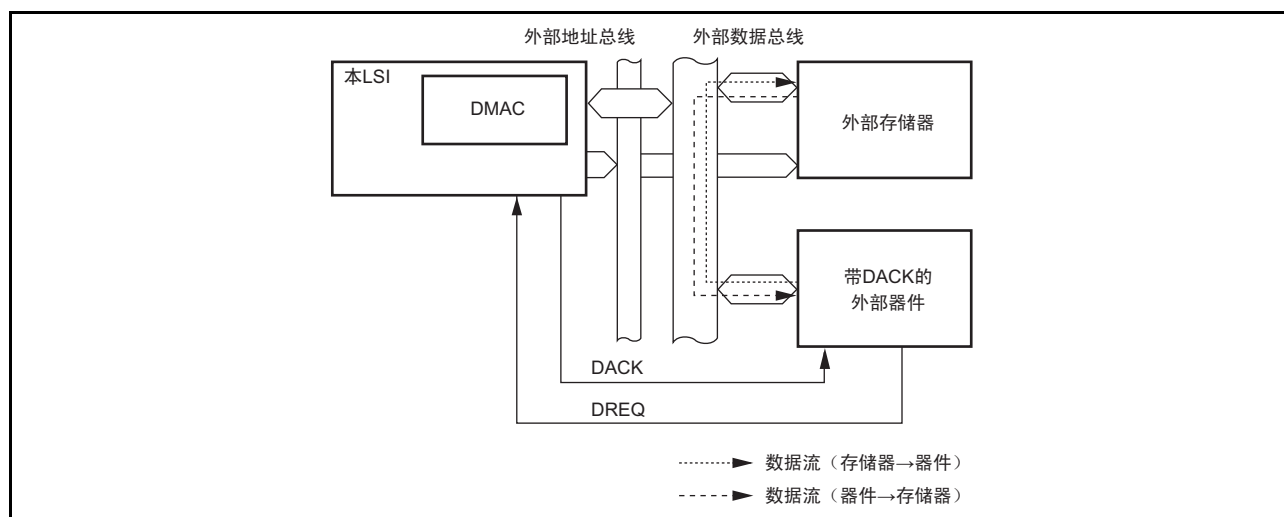


图 10.7 单地址模式的数据流程

单地址模式可执行的传送：（1）带 DACK 的外部器件和存储器映射的外部器件之间的传送；（2）带 DACK 的外部器件和外部存储器之间的传送。无论哪种情况，传送请求仅为外部请求（DREQ）。

单地址模式的 DMA 传送时序例如图 10.8 所示。

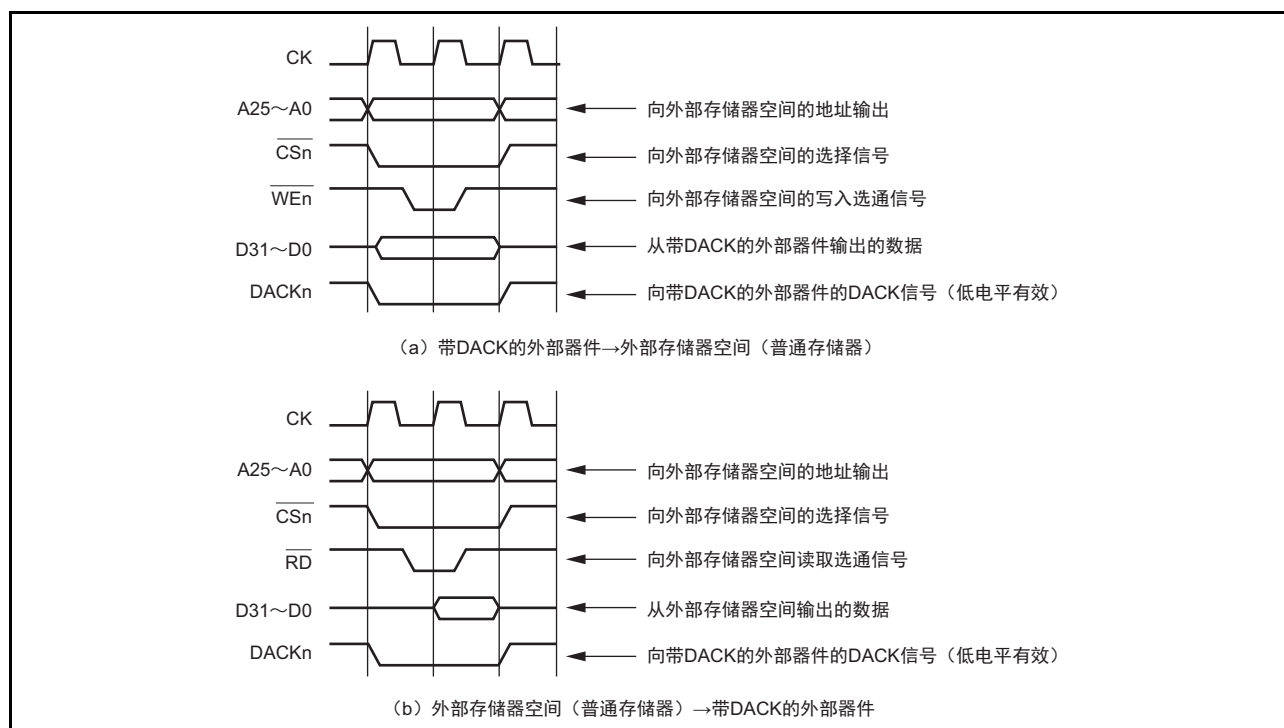


图 10.8 单地址模式的 DMA 传送时序例

(2) 总线模式

总线模式有周期挪用模式和突发模式，由通道控制寄存器 (CHCR) 的 TB 位选择模式。

(a) 周期挪用模式

• 普通模式

在周期挪用普通模式，每当1个传送单位（字节、字、长字或16字节单位）的传送结束时，DMAC 就将总线权交给其他总线主控器。此后如果产生传送请求，则可从其他总线主控器取回总线权，再次执行1个传送单位的传送，此传送结束时，又将总线权交给其他总线主控器。重复此操作直到满足传送结束条件为止。

周期挪用普通模式与传送请求源、传送源和传送目标无关，可在所有传送区间使用。

周期挪用普通模式的DMA传送时序例如图10.9所示，图中的传送条件如下：

- 双地址模式
- DREQ低电平检测

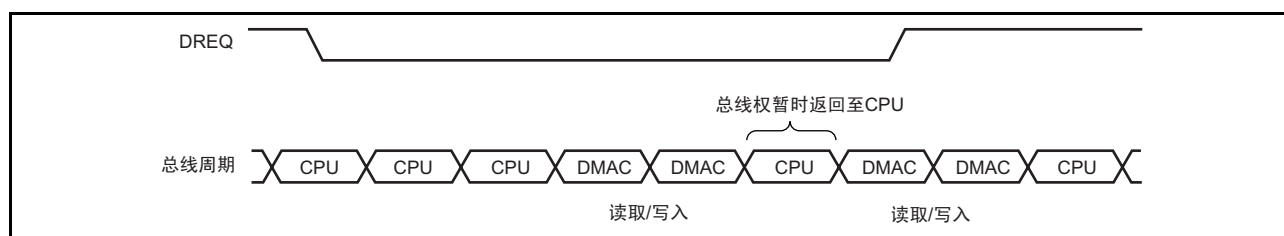


图 10.9 周期挪用普通模式的 DMA 传送例
(双地址、DREQ 低电平检测)

• 间歇模式16、间歇模式64

在周期挪用间歇模式，每当1个传送单位（字节、字、长字或16字节单位）的传送结束时，DMAC 就将总线权交给其他总线主控器。此后如果产生传送请求，则以 $B\phi$ 时钟计数等待16个时钟或64个时钟后，从其他总线主控器取回总线权，再次执行1个传送单位的传送，此传送结束时，又将总线权交给其他总线主控器。重复此操作直到满足结束条件为止。与周期挪用普通模式相比，可降低DMA传送的总线占用率。

DMAC再次取回总线权时，在执行因高速缓存未命中引起的入口更新时，DMA传送可能会等待更长时间。

间歇模式与传送请求源、传送源和传送目标无关，可在所有传送区间使用，但所有通道的总线模式必须为周期挪用模式。

周期挪用间歇模式的DMA传送时序例如图10.10所示，图中的传送条件如下：

- 双地址模式
- DREQ低电平检测

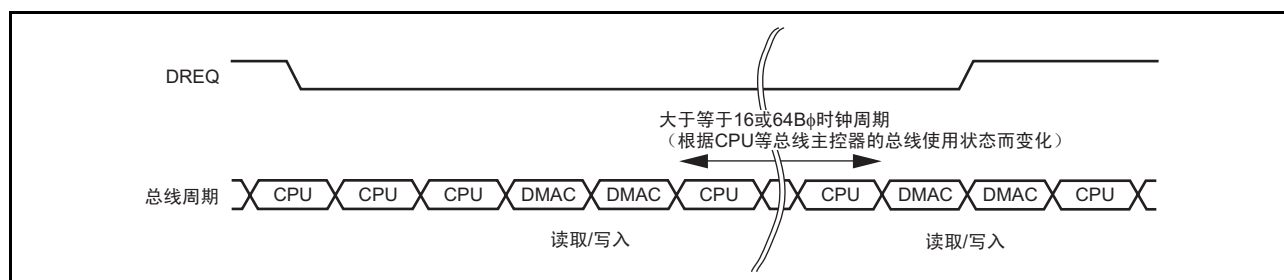


图 10.10 周期挪用间歇模式的 DMA 传送例 (双地址、DREQ 低电平检测)

(b) 突发模式

在突发模式，DMAC 一旦取得总线权，在满足传送结束条件前，不释放总线权而继续传送。但在外部请求模式，通过电平检测 DREQ 时，如果 DREQ 变为无效电平，即使未满足传送结束条件，也要在结束已接受请求的 DMAC 传送后，将总线权交给其他总线主控器。

突发模式的 DMA 传送时序如图 10.11 所示。

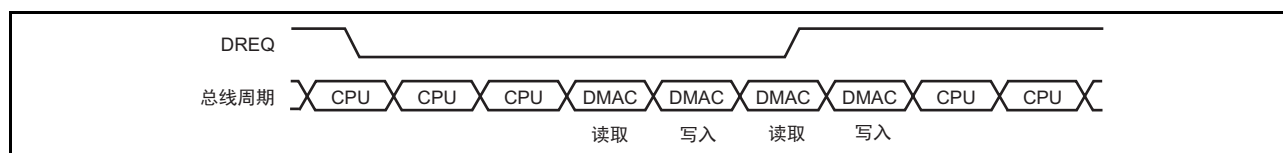


图 10.11 突发模式的 DMA 传送例（双地址、DREQ 低电平检测）

(3) DMA 传送区间与请求模式、总线模式的关系

DMA 传送区间与请求模式及总线模式等的相关事项如表 10.10 所示。

表 10.10 DMA 传送区间与请求模式、总线模式关系一览表

地址模式	传送区间	请求模式	总线模式	传送长度 (位)	可使用的通道
双地址模式	带 DACK 的外部器件和外部存储器	外部	B/C	8/16/32/128	0 ~ 3
	带 DACK 的外部器件和存储器映射的外部器件	外部	B/C	8/16/32/128	0 ~ 3
	外部存储器和外部存储器	均可 *4	B/C	8/16/32/128	0 ~ 7*3
	外部存储器和存储器映射的外部器件	均可 *4	B/C	8/16/32/128	0 ~ 7*3
	存储器映射的外部器件和存储器映射的外部器件	均可 *4	B/C	8/16/32/128	0 ~ 7*3
	外部存储器和内部外围模块	均可 *1	B/C*5	8/16/32/128*2	0 ~ 7*3
	存储器映射的外部器件和内部外围模块	均可 *1	B/C*5	8/16/32/128*2	0 ~ 7*3
	内部外围模块和内部外围模块	均可 *1	B/C*5	8/16/32/128*2	0 ~ 7*3
	内部存储器和内部存储器	均可 *4	B/C	8/16/32/128	0 ~ 7*3
	内部存储器和存储器映射的外部器件	均可 *4	B/C	8/16/32/128	0 ~ 7*3
	内部存储器和内部外围模块	均可 *1	B/C*5	8/16/32/128*2	0 ~ 7*3
	内部存储器和外部存储器	均可 *4	B/C	8/16/32/128	0 ~ 7*3
单地址模式	带 DACK 的外部器件和外部存储器	外部	B/C	8/16/32/128	0 ~ 3
	带 DACK 的外部器件和存储器映射的外部器件	外部	B/C	8/16/32/128	0 ~ 3

【符号说明】

B: 突发模式

C: 周期挪用模式

【注】 *1 外部请求、自动请求和内部外围模块请求均可。

但在内部外围模块请求时，除传送请求源为 MTU2、CMT 之外，传送源或传送目标需为各自的请求源寄存器。

*2 需为传送源或传送目标的内部外围模块的寄存器所允许的存取长度。

*3 传送请求为外部请求时仅适用于通道 0 ~ 3。

*4 外部请求、自动请求和内部外围模块请求均可。

但是为内部外围模块请求时，仅可使用 MTU2、CMT。

*5 在内部外围模块请求时，除传送请求源为 USB、SSI、MTU2 及 CMT 之外，仅为周期挪用模式。

(4) 总线模式和通道优先顺序

在优先顺序固定模式 ($CH0 > CH1$)，即使通道 1 正以突发模式传送，比此优先顺序更高的通道 0 产生传送请求时，也立即开始通道 0 的传送。

此时，通道 0 也为突发模式时，优先顺序高的通道 0 的传送全部结束后，继续通道 1 的传送。

通道 0 为周期挪用模式时，首先执行优先顺序高的通道 0 的 1 个传送单位的传送，之后不释放总线权而连续通道 1 的传送。然后按照通道 0→通道 1→通道 0→通道 1 的方式交替传送，即：总线状态变为周期挪用模式传送结束后，CPU 周期被替换为突发模式传送的状态（以下称为突发模式的优先执行）。此例如图 10.12 所示。产生竞争的突发模式有多个通道时，优先执行优先顺序最高的通道。

在多个通道执行 DMA 传送时，竞争的所有突发模式传送结束前，不向总线主控制器释放总线权。

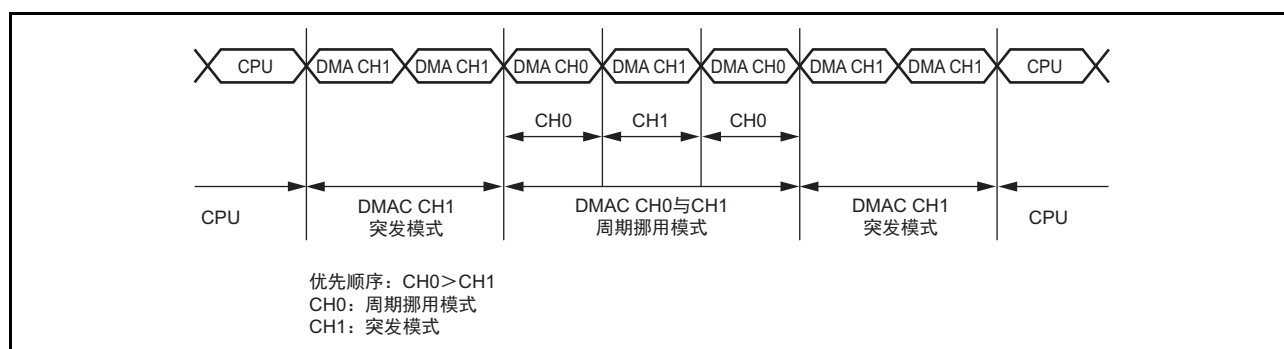


图 10.12 多个通道运行时的总线状态

在轮询模式，优先顺序按照图 10.13 所示的规格变化，但是，周期挪用模式的通道和突发模式的通道不可同时存在。

10.4.5 总线周期的状态数和 DREQ 引脚的采样时序

(1) 总线周期的状态数

DMAC 为总线主控制器时，与 CPU 为总线主控制器时相同，可通过总线状态控制器 (BSC) 控制总线周期状态数。详情参阅“第 9 章 总线状态控制器 (BSC)”。

(2) DREQ 引脚的采样时序

各总线模式的 DREQ 输入采样时序如图 10.13 ~ 图 10.16 所示。

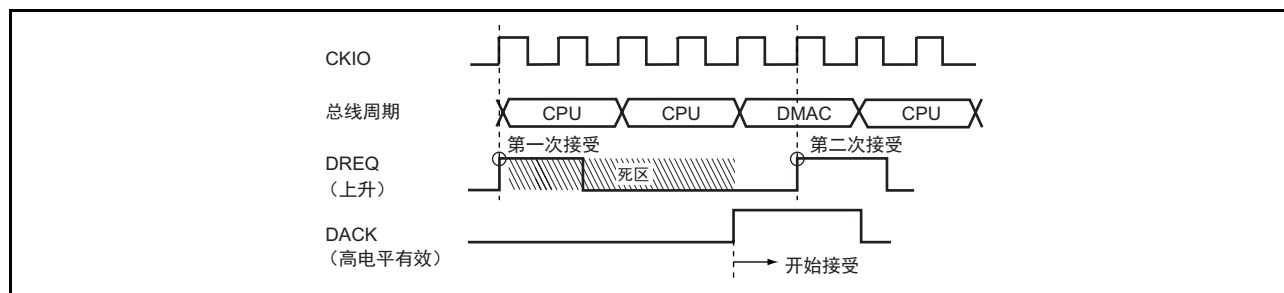


图 10.13 周期挪用模式 / 边沿检测时的 DREQ 输入检测时序

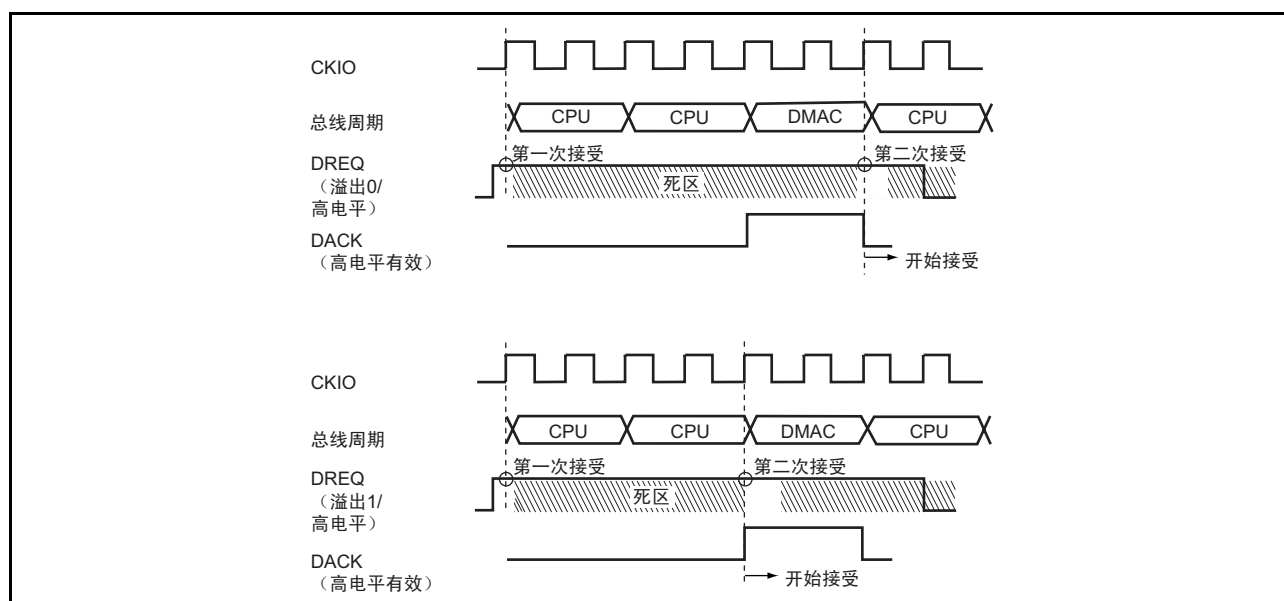


图 10.14 周期挪用模式 / 电平检测时的 DREQ 输入检测时序

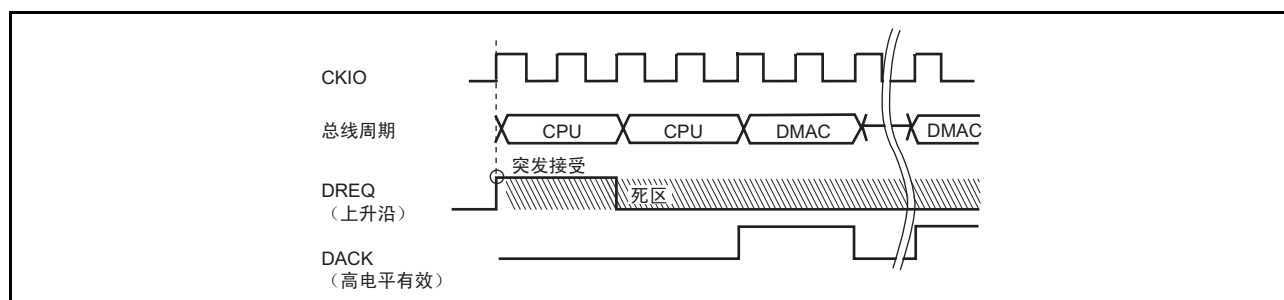


图 10.15 突发模式 / 边沿检测时的 DREQ 输入检测时序

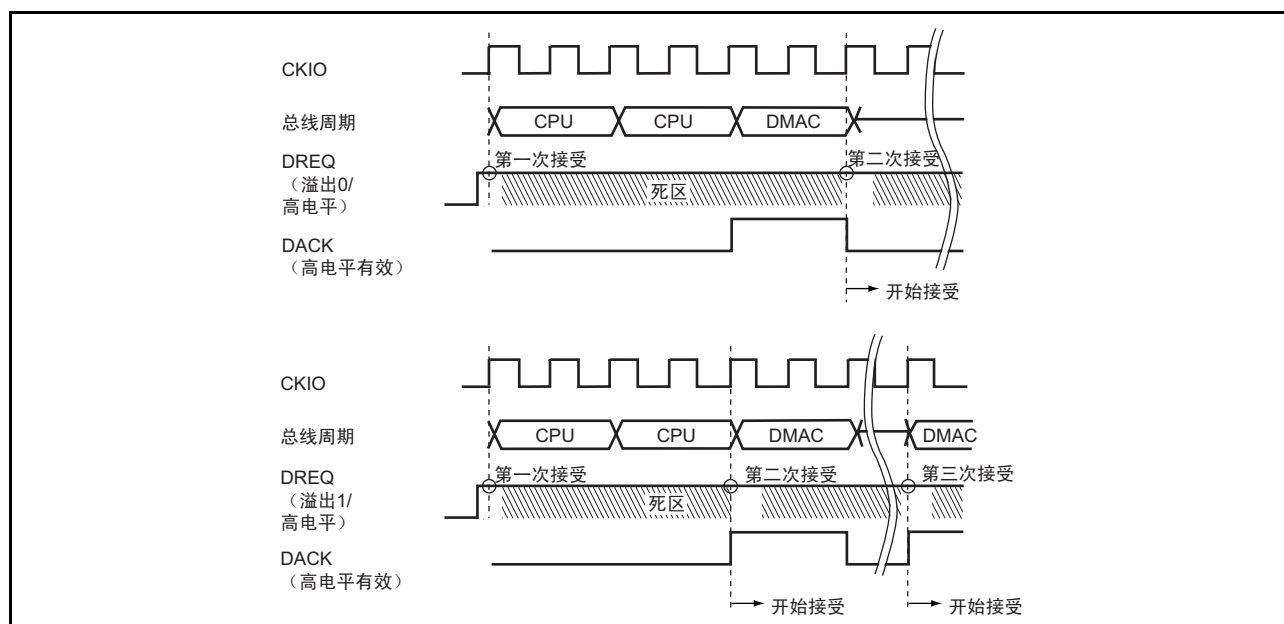


图 10.16 突发模式 / 电平检测时的 DREQ 输入检测时序

TEND 输出时序如图 10.17 所示。

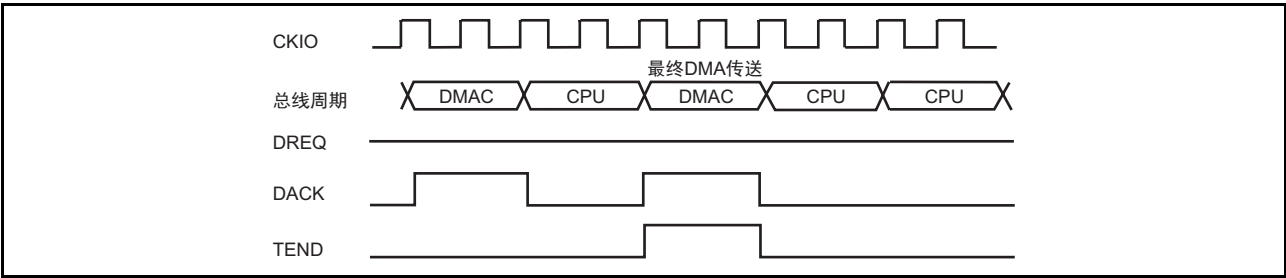


图 10.17 DMA 传送结束信号时序（周期挪用模式 / 电平检测）

对 8 位、16 位、32 位外部器件执行 16 字节传送，对 8 位、16 位外部器件执行长字存取，对 8 位外部器件执行字存取时，DMA 传送单位分割为多个总线周期。在 DMA 传送单位分割为多个总线周期，且在总线周期将 $\overline{CS}$ 设定为无效时，必须注意为了对齐数据，和 $\overline{CS}$ 信号一样， $\overline{DACK}$ 输出和 $\overline{TEND}$ 输出被分割。此例如图 10.18 所示。另外，DMA 传送时 $\overline{DACK}$ 、 $\overline{TEND}$ 不被分割的情况如图 10.13 ~ 图 10.17 所示。

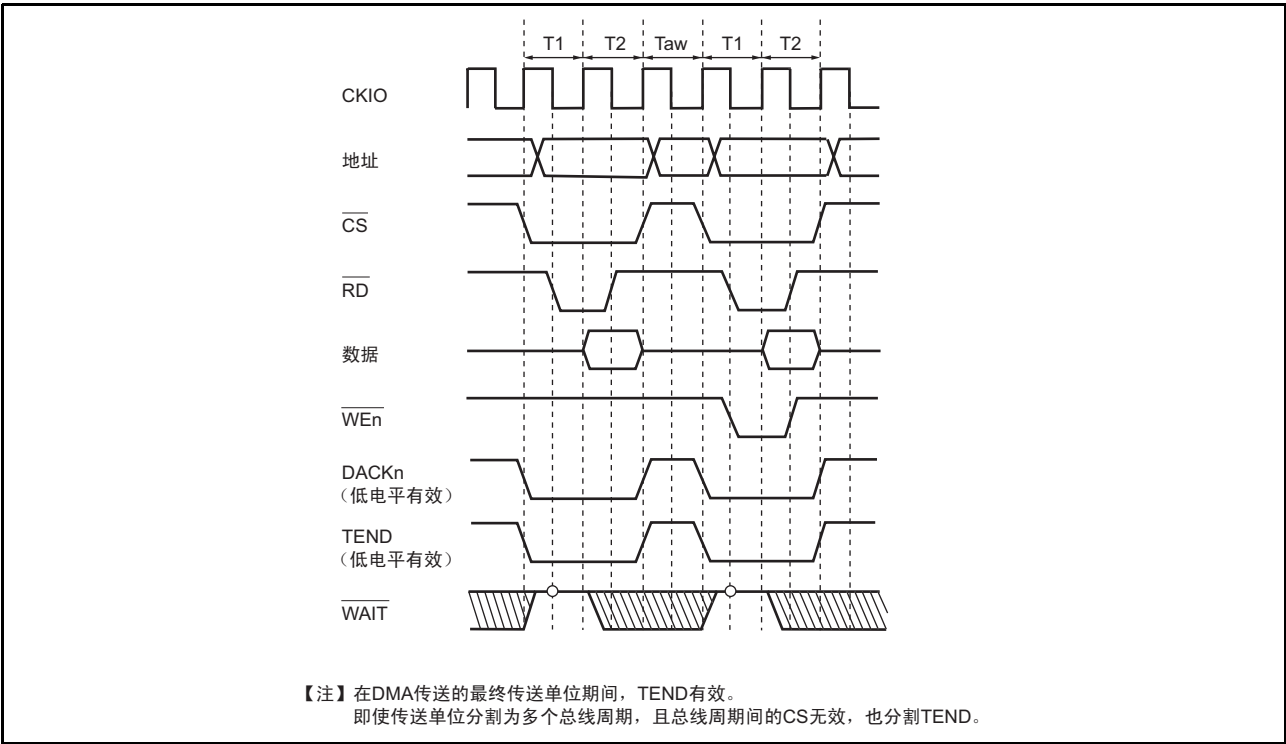


图 10.18 BSC 普通存储器存取
（无等待、1 个空闲周期、对 16 位器件的长字存取）

10.5 使用时的注意事项

10.5.1 半结束标志的置位及半结束中断

参照 CHCR 寄存器的半结束标志状态及使用半结束中断，并且与重加载功能一起使用时，必须注意以下事项：

必须将重加载的传送次数（对 RDMATCR 设定的值）设置为与最初设定的传送次数（对 DMATCR 设定的值）相同的次数。最初的 DMATCR 设定值与第 2 次之后传送所使用的 RDMATCR 设定值不同时，可能有半结束标志的置位时序快于传送次数的一半，或半结束标志不能置位的情况。半结束中断也相同。

10.5.2 DACK 输出及 TEND 输出的时序

外部存储器为 MPX-I/O 或突发 MPX-I/O 时，在数据周期的时序 DACK 输出有效。详情参阅“第 9 章 总线状态控制器 (BSC)”的“9.5.5 MPX-I/O 接口”或“9.5.10 突发 MPX-I/O 接口”的各图。

为 MPX-I/O 及突发 MPX-I/O 之外的存储器类型时，在与相应的 CS 有效的相同时序，DACK 输出也有效。

TEND 输出与存储器类型无关，总是在与相应的 CS 有效的相同时序有效。

10.5.3 使用外部请求模式的注意事项

存在由外部请求启动的通道时，必须使用如下 4 项中的一项：

1. 必须在周期挪用模式使用所有通道。
2. 在突发模式使用所有通道时，必须满足以下 3 个条件才可使用：
 - 2-1. 必须将通道的优先顺序设置为固定模式 1 或固定模式 2。
 - 2-2. 必将所有通道设置为双地址模式。
 - 2-3. 必须将所有通道的传送源地址及传送目标地址设置为以下任意一种：
 - A. 传送源地址：外部地址空间 传送目标地址：外部地址空间
 - B. 传送源地址：外部地址空间 传送目标地址：内部地址空间
 - C. 传送源地址：内部地址空间 传送目标地址：内部地址空间
3. 所有通道同时存在周期挪用模式和突发模式时，必须满足以下 3 个条件才可使用：
 - 3-1. 必须将通道的优先顺序设置为固定模式 1 或固定模式 2。
 - 3-2. 必将所有通道设置为双地址模式。
 - 3-3. 必须将所有通道的传送源地址及传送目标地址设置为以下任意一种：
 - A. 传送源地址：外部地址空间 传送目标地址：外部地址空间
 - B. 传送源地址：外部地址空间 传送目标地址：内部地址空间
 - C. 传送源地址：内部地址空间 传送目标地址：内部地址空间
4. 必须在单通道使用。

在上述 4 项之外使用时，DACK_n 引脚及 TEND_n 引脚表示错误传送通道，此后直到上电复位，为不可执行 DMA 传送的状态。而且，在突发模式产生此状态时，CPU 无法取指令，导致的结果就是系统运行停止。

10.5.4 使用内部外围模块请求模式或自动请求模式时的注意事项

由内部外围模块请求或自动请求启动，存在使用 DACKn 引脚及 TENDn 引脚的通道时，必须使用以下 4 项中的一项：

1. 必须在周期挪用模式使用所有通道。
2. 在突发模式使用所有通道时，必须满足以下 3 个条件才可使用：
 - 2-1. 必须将通道的优先顺序设置为固定模式 1 或固定模式 2。
 - 2-2. 必将将所有通道设置为双地址模式。
 - 2-3. 必须将所有通道的传送源地址及传送目标地址设置为以下任意一种：
 - A. 传送源地址：外部地址空间 传送目标地址：外部地址空间
 - B. 传送源地址：外部地址空间 传送目标地址：内部地址空间
 - C. 传送源地址：内部地址空间 传送目标地址：内部地址空间
3. 所有通道同时存在周期挪用模式和突发模式时，必须满足以下 3 个条件才可使用：
 - 3-1. 必须将通道的优先顺序设置为固定模式 1 或固定模式 2。
 - 3-2. 必将将所有通道设置为双地址模式。
 - 3-3. 必须将所有通道的传送源地址及传送目标地址设置为以下任意一种：
 - A. 传送源地址：外部地址空间 传送目标地址：外部地址空间
 - B. 传送源地址：外部地址空间 传送目标地址：内部地址空间
 - C. 传送源地址：内部地址空间 传送目标地址：内部地址空间
4. 必须在单通道使用。

在上述 4 项之外使用时，DACKn 引脚及 TENDn 引脚表示错误传送通道。

第 11 章 多功能定时器脉冲单元 2 (MTU2)

本 LSI 内置由 5 个通道的 16 位定时器构成的多功能定时器脉冲单元 2 (MTU2)。

11.1 特点

- 最多可输入/输出 16 个脉冲
- 各通道均可选择 8 种计数器输入时钟
- 可设定以下运行：由比较匹配输出波形、输入捕捉功能、计数器清除运行、同时写入多个定时器计数器 (TCNT)、由比较匹配/输入捕捉同时清除、因计数器的同步运行而产生各寄存器的同步输入/输出、与同步运行组合最多输出 12 相 PWM
- 通道 0、3、4 可设定缓冲运行
- 通道 1、2 可分别独立设定相位计数模式
- 级联运行
- 通过内部 16 位总线高速存取
- 28 种中断源
- 可自动传送寄存器数据
- 可生成 A/D 转换器的转换开始触发
- 可设定模块待机模式
- CH3、4 的联动，可设定互补 PWM、复位 PWM3 相的正、负共计 6 相波形输出
- CH0、3、4 联动，可设定使用互补 PWM、复位 PWM 的 AC 同步马达（无刷 DC 马达）驱动模式，并可选择 2 种（斩波、电平）波形输出
- 互补 PWM 模式时，可跳过计数器波峰/波谷处的中断及 A/D 转换器的转换开始触发

表 11.1 MTU2 功能一览表

项 目	通道 0	通道 1	通道 2	通道 3	通道 4
计数时钟	Pφ/1 Pφ/4 Pφ/16 Pφ/64 TCLKA TCLKB TCLKC TCLKD	Pφ/1 Pφ/4 Pφ/16 Pφ/64 Pφ/256 TCLKA TCLKB	Pφ/1 Pφ/4 Pφ/16 Pφ/64 Pφ/1024 TCLKA TCLKB TCLKC	Pφ/1 Pφ/4 Pφ/16 Pφ/64 Pφ/256 Pφ/1024 TCLKA TCLKB	Pφ/1 Pφ/4 Pφ/16 Pφ/64 Pφ/256 Pφ/1024 TCLKA TCLKB
通用寄存器 (TGR)	TGRA_0 TGRB_0 TGRE_0	TGRA_1 TGRB_1	TGRA_2 TGRB_2	TGRA_3 TGRB_3	TGRA_4 TGRB_4
通用寄存器 / 缓冲寄存器	TGRC_0 TGRD_0 TGRF_0	—	—	TGRC_3 TGRD_3	TGRC_4 TGRD_4
输入 / 输出引脚	TIOC0A TIOC0B TIOC0C TIOC0D	TIOC1A TIOC1B	TIOC2A TIOC2B	TIOC3A TIOC3B TIOC3C TIOC3D	TIOC4A TIOC4B TIOC4C TIOC4D
计数器清除功能	TGR 的比较匹配或输入捕捉	TGR 的比较匹配或输入捕捉	TGR 的比较匹配或输入捕捉	TGR 的比较匹配或输入捕捉	TGR 的比较匹配或输入捕捉

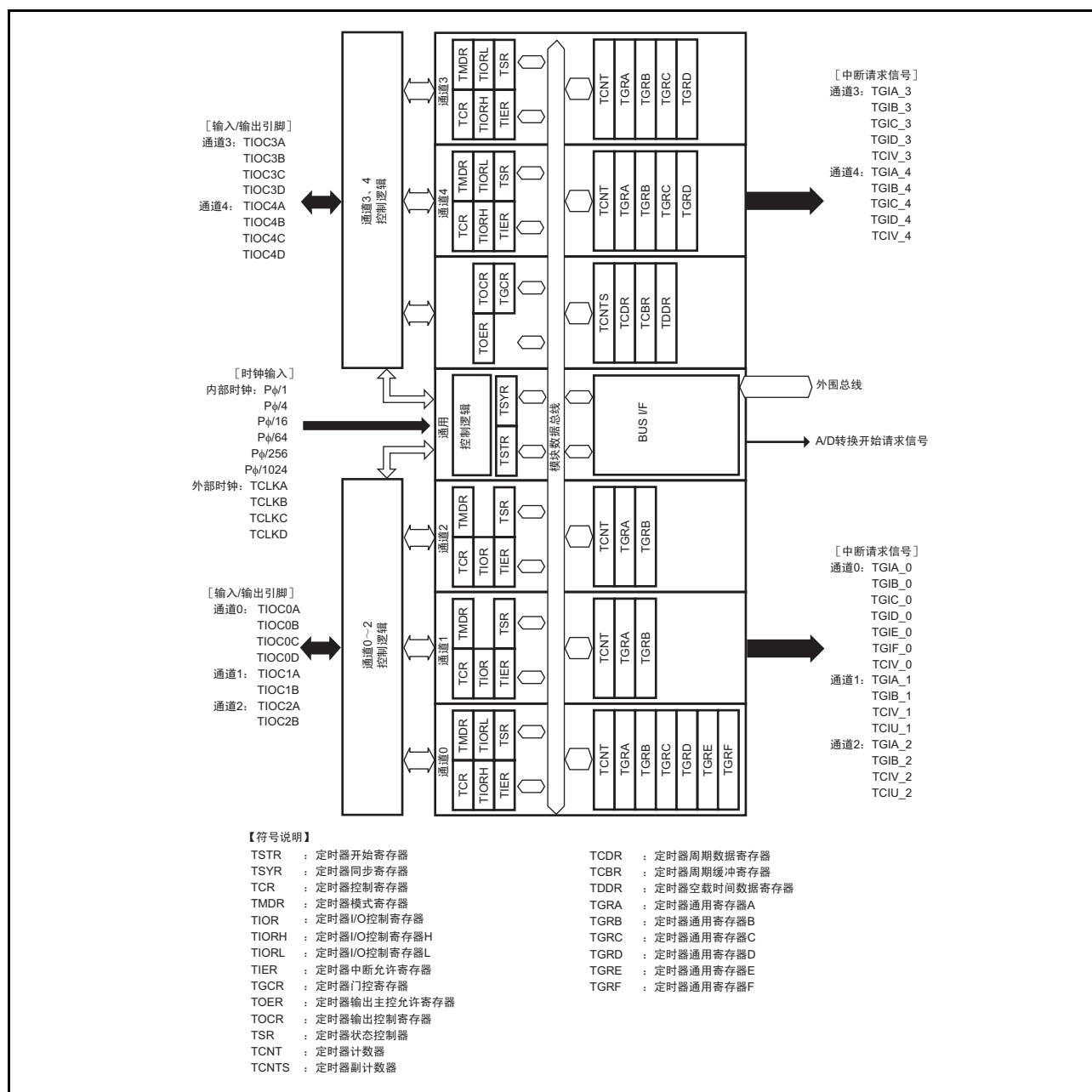
项 目	通道 0	通道 1	通道 2	通道 3	通道 4
比较匹配 输出	输出 0	○	○	○	○
	输出 1	○	○	○	○
	交替输出	○	○	○	○
输入捕捉功能	○	○	○	○	○
同步运行	○	○	○	○	○
PWM 模式 1	○	○	○	○	○
PWM 模式 2	○	○	○	—	—
互补 PWM 模式	—	—	—	○	○
复位 PWM 模式	—	—	—	○	○
AC 同步马达驱动模式	○	—	—	○	○
相位计数模式	—	○	○	—	—
缓冲运行	○	—	—	○	○
DMAC 的启动	TGR 的比较匹配 或输入捕捉	TGR 的比较匹配 或输入捕捉	TGR 的比较匹配 或输入捕捉	TGR 的比较匹配 或输入捕捉	TGR 的比较匹配 或输入捕捉及 TCNT 上溢 / 下溢
A/D 转换开始触发	TGRA_0 的比较 匹配或输入捕捉 TGRE_0 的比较 匹配	TGRA_1 的比较 匹配或输入捕捉	TGRA_2 的比较 匹配或输入捕捉	TGRA_3 的比较 匹配或输入捕捉	TGRA_4 的比较匹 配或输入捕捉 互补 PWM 模式时 TCNT_4 的下溢 (波谷)
中断源	7 个 • 比较匹配 / 输 入捕捉 0A • 比较匹配 / 输 入捕捉 0B • 比较匹配 / 输 入捕捉 0C • 比较匹配 / 输 入捕捉 0D • 比较匹配 0E • 比较匹配 0F • 上溢	4 个 • 比较匹配 / 输 入捕捉 1A • 比较匹配 / 输 入捕捉 1B • 上溢 • 下溢	4 个 • 比较匹配 / 输 入捕捉 2A • 比较匹配 / 输 入捕捉 2B • 上溢 • 下溢	5 个 • 比较匹配 / 输 入捕捉 3A • 比较匹配 / 输 入捕捉 3B • 比较匹配 / 输 入捕捉 3C • 比较匹配 / 输 入捕捉 3D • 上溢	5 个 • 比较匹配 / 输 入捕捉 4A • 比较匹配 / 输 入捕捉 4B • 比较匹配 / 输 入捕捉 4C • 比较匹配 / 输 入捕捉 4D • 上溢 / 下溢
A/D 转换开始 请求延迟功能	—	—	—	—	• TADCORA_4 与 TCNT_4 匹配 时, 请求开始 A/ D 转换 • TADCORB_4 与 TCNT_4 匹配 时, 请求开始 A/ D 转换
中断跳过功能	—	—	—	• 跳过 TGRA_3 的比较匹配中 断	• 跳过 TCIV_4 中 断

【符号说明】

○: 可

—: 不可

MTU2 框图如图 11.1 所示。



11.2 输入 / 输出引脚

表 11.2 引脚结构

通道	引脚名称	输入 / 输出	功 能
通用	TCLKA	输入	外部时钟 A 输入引脚 (通道 1 的相位计数模式 A 相输入)
	TCLKB	输入	外部时钟 B 输入引脚 (通道 1 的相位计数模式 B 相输入)
	TCLKC	输入	外部时钟 C 输入引脚 (通道 2 的相位计数模式 A 相输入)
	TCLKD	输入	外部时钟 D 输入引脚 (通道 2 的相位计数模式 B 相输入)
0	TIOC0A	输入 / 输出	TGRA_0 的输入捕捉输入 / 输出比较输出 / PWM 输出引脚
	TIOC0B	输入 / 输出	TGRB_0 的输入捕捉输入 / 输出比较输出 / PWM 输出引脚
	TIOC0C	输入 / 输出	TGRC_0 的输入捕捉输入 / 输出比较输出 / PWM 输出引脚
	TIOC0D	输入 / 输出	TGRD_0 的输入捕捉输入 / 输出比较输出 / PWM 输出引脚
1	TIOC1A	输入 / 输出	TGRA_1 的输入捕捉输入 / 输出比较输出 / PWM 输出引脚
	TIOC1B	输入 / 输出	TGRB_1 的输入捕捉输入 / 输出比较输出 / PWM 输出引脚
2	TIOC2A	输入 / 输出	TGRA_2 的输入捕捉输入 / 输出比较输出 / PWM 输出引脚
	TIOC2B	输入 / 输出	TGRB_2 的输入捕捉输入 / 输出比较输出 / PWM 输出引脚
3	TIOC3A	输入 / 输出	TGRA_3 的输入捕捉输入 / 输出比较输出 / PWM 输出引脚
	TIOC3B	输入 / 输出	TGRB_3 的输入捕捉输入 / 输出比较输出 / PWM 输出引脚
	TIOC3C	输入 / 输出	TGRC_3 的输入捕捉输入 / 输出比较输出 / PWM 输出引脚
	TIOC3D	输入 / 输出	TGRD_3 的输入捕捉输入 / 输出比较输出 / PWM 输出引脚
4	TIOC4A	输入 / 输出	TGRA_4 的输入捕捉输入 / 输出比较输出 / PWM 输出引脚
	TIOC4B	输入 / 输出	TGRB_4 的输入捕捉输入 / 输出比较输出 / PWM 输出引脚
	TIOC4C	输入 / 输出	TGRC_4 的输入捕捉输入 / 输出比较输出 / PWM 输出引脚
	TIOC4D	输入 / 输出	TGRD_4 的输入捕捉输入 / 输出比较输出 / PWM 输出引脚

【注】 有关互补 PWM 模式的引脚结构，详情参照“11.4.8 互补 PWM 模式”的表 11.54。

11.3 寄存器说明

MTU2 有以下寄存器。有关寄存器的地址及各处理状态的寄存器状态，详情参阅“第 30 章 寄存器一览表”。各通道寄存器名称的表示：通道 0 的 TCR 表示为 TCR_0。

表 11.3 寄存器结构

通道	寄存器名称	简称	R/W	初始值	地址	存取长度
0	定时器控制寄存器 _0	TCR_0	R/W	H'00	H'FFFE4300	8
	定时器模式寄存器 _0	TMDR_0	R/W	H'00	H'FFFE4301	8
	定时器 I/O 控制寄存器 H_0	TIORH_0	R/W	H'00	H'FFFE4302	8
	定时器 I/O 控制寄存器 L_0	TIORL_0	R/W	H'00	H'FFFE4303	8
	定时器中断允许寄存器 _0	TIER_0	R/W	H'00	H'FFFE4304	8
	定时器状态寄存器 _0	TSR_0	R/W	H'C0	H'FFFE4305	8
	定时器计数器 _0	TCNT_0	R/W	H'0000	H'FFFE4306	16
	定时器通用寄存器 A_0	TGRA_0	R/W	H'FFFF	H'FFFE4308	16
	定时器通用寄存器 B_0	TGRB_0	R/W	H'FFFF	H'FFFE430A	16
	定时器通用寄存器 C_0	TGRC_0	R/W	H'FFFF	H'FFFE430C	16
	定时器通用寄存器 D_0	TGRD_0	R/W	H'FFFF	H'FFFE430E	16
	定时器通用寄存器 E_0	TGRE_0	R/W	H'FFFF	H'FFFE4320	16
	定时器通用寄存器 F_0	TGRF_0	R/W	H'FFFF	H'FFFE4322	16
	定时器中断允许寄存器 2_0	TIER2_0	R/W	H'00	H'FFFE4324	8
	定时器状态寄存器 2_0	TSR2_0	R/W	H'00	H'FFFE4325	8
	定时器缓冲运行传送模式寄存器 _0	TBTM_0	R/W	H'C0	H'FFFE4326	8
1	定时器控制寄存器 _1	TCR_1	R/W	H'00	H'FFFE4380	8
	定时器模式寄存器 _1	TMDR_1	R/W	H'00	H'FFFE4381	8
	定时器 I/O 控制寄存器 _1	TIOR_1	R/W	H'00	H'FFFE4382	8
	定时器中断允许寄存器 _1	TIER_1	R/W	H'00	H'FFFE4384	8
	定时器状态寄存器 _1	TSR_1	R/W	H'C0	H'FFFE4385	8
	定时器计数器 _1	TCNT_1	R/W	H'0000	H'FFFE4386	16
	定时器通用寄存器 A_1	TGRA_1	R/W	H'FFFF	H'FFFE4388	16
	定时器通用寄存器 B_1	TGRB_1	R/W	H'FFFF	H'FFFE438A	16
	定时器输入捕捉控制寄存器	TICCR	R/W	H'00	H'FFFE4390	8

通道	寄存器名称	简称	R/W	初始值	地址	存取长度
2	定时器控制寄存器 _2	TCR_2	R/W	H'00	H'FFFE4000	8
	定时器模式寄存器 _2	TMDR_2	R/W	H'00	H'FFFE4001	8
	定时器 I/O 控制寄存器 _2	TIOR_2	R/W	H'00	H'FFFE4002	8
	定时器中断允许寄存器 _2	TIER_2	R/W	H'00	H'FFFE4004	8
	定时器状态寄存器 _2	TSR_2	R/W	H'C0	H'FFFE4005	8
	定时器计数器 _2	TCNT_2	R/W	H'0000	H'FFFE4006	16
	定时器通用寄存器 A_2	TGRA_2	R/W	H'FFFF	H'FFFE4008	16
	定时器通用寄存器 B_2	TGRB_2	R/W	H'FFFF	H'FFFE400A	16
3	定时器控制寄存器 _3	TCR_3	R/W	H'00	H'FFFE4200	8
	定时器模式寄存器 _3	TMDR_3	R/W	H'00	H'FFFE4202	8
	定时器 I/O 控制寄存器 H_3	TIORH_3	R/W	H'00	H'FFFE4204	8
	定时器 I/O 控制寄存器 L_3	TIORL_3	R/W	H'00	H'FFFE4205	8
	定时器中断允许寄存器 _3	TIER_3	R/W	H'00	H'FFFE4208	8
	定时器状态寄存器 _3	TSR_3	R/W	H'C0	H'FFFE422C	8
	定时器计数器 _3	TCNT_3	R/W	H'0000	H'FFFE4210	16
	定时器通用寄存器 A_3	TGRA_3	R/W	H'FFFF	H'FFFE4218	16
	定时器通用寄存器 B_3	TGRB_3	R/W	H'FFFF	H'FFFE421A	16
	定时器通用寄存器 C_3	TGRC_3	R/W	H'FFFF	H'FFFE4224	16
	定时器通用寄存器 D_3	TGRD_3	R/W	H'FFFF	H'FFFE4226	16
	定时器缓冲运行传送模式寄存器 _3	TBTM_3	R/W	H'00	H'FFFE4238	8
4	定时器控制寄存器 _4	TCR_4	R/W	H'00	H'FFFE4201	8
	定时器模式寄存器 _4	TMDR_4	R/W	H'00	H'FFFE4203	8
	定时器 I/O 控制寄存器 H_4	TIORH_4	R/W	H'00	H'FFFE4206	8
	定时器 I/O 控制寄存器 L_4	TIORL_4	R/W	H'00	H'FFFE4207	8
	定时器中断允许寄存器 _4	TIER_4	R/W	H'00	H'FFFE4209	8
	定时器状态寄存器 _4	TSR_4	R/W	H'C0	H'FFFE422D	8
	定时器计数器 _4	TCNT_4	R/W	H'0000	H'FFFE4212	16
	定时器通用寄存器 A_4	TGRA_4	R/W	H'FFFF	H'FFFE421C	16
	定时器通用寄存器 B_4	TGRB_4	R/W	H'FFFF	H'FFFE421E	16
	定时器通用寄存器 C_4	TGRC_4	R/W	H'FFFF	H'FFFE4228	16
	定时器通用寄存器 D_4	TGRD_4	R/W	H'FFFF	H'FFFE422A	16
	定时器缓冲运行传送模式寄存器 _4	TBTM_4	R/W	H'00	H'FFFE4239	8

通道	寄存器名称	简称	R/W	初始值	地址	存取长度
4	定时器 A/D 转换开始请求控制寄存器	TADCR	R/W	H'0000	H'FFFE4240	16
	定时器 A/D 转换开始请求周期设定寄存器 A_4	TADCORA_4	R/W	H'FFFF	H'FFFE4244	16
	定时器 A/D 转换开始请求周期设定寄存器 B_4	TADCORB_4	R/W	H'FFFF	H'FFFE4246	16
	定时器 A/D 转换开始请求周期设定缓冲寄存器 A_4	TADCOBRA_4	R/W	H'FFFF	H'FFFE4248	16
	定时器 A/D 转换开始请求周期设定缓冲寄存器 B_4	TADCOBRB_4	R/W	H'FFFF	H'FFFE424A	16
通用	定时器开始寄存器	TSTR	R/W	H'00	H'FFFE4280	8
	定时器同步寄存器	TSYR	R/W	H'00	H'FFFE4281	8
	定时器读取 / 写入允许寄存器	TRWER	R/W	H'01	H'FFFE4284	8
3/4 通用	定时器输出主许寄存器	TOER	R/W	H'C0	H'FFFE420A	8
	定时器输出控制寄存器 1	TOCR1	R/W	H'00	H'FFFE420E	8
	定时器输出控制寄存器 2	TOCR2	R/W	H'00	H'FFFE420F	8
	定时器门控寄存器	TGCR	R/W	H80	H'FFFE420D	8
	定时器周期数据寄存器	TCDR	R/W	H'FFFF	H'FFFE4214	16
	定时器空载时间数据寄存器	TDDR	R/W	H'FFFF	H'FFFE4216	16
	定时器副计数器	TCNTS	R	H'0000	H'FFFE4220	16
	定时器周期缓冲寄存器	TCBR	R/W	H'FFFF	H'FFFE4222	16
	定时器中断跳过设定寄存器	TITCR	R/W	H'00	H'FFFE4230	8
	定时器中断跳过次数计数器	TITCNT	R	H'00	H'FFFE4231	8
	定时器缓冲传送设定寄存器	TBTER	R/W	H'00	H'FFFE4232	8
	定时器空载时间允许寄存器	TDER	R/W	H'01	H'FFFE4234	8
	定时器波形控制寄存器	TWCR	R/W	H'00	H'FFFE4260	8
	定时器输出电平缓冲寄存器	TOLBR	R/W	H'00	H'FFFE4236	8

11.3.1 定时器控制寄存器 (TCR)

TCR 为控制各通道 TCNT 的 8 位可读取 / 写入的寄存器。MTU2 共有 5 个 TCR，通道 0 ~ 4 各有 1 个，必须在 TCNT 运行停止状态设定 TCR。

位:	7	6	5	4	3	2	1	0
	CCLR[2:0]			CKEG[1:0]		TPSC[2:0]		
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7 ~ 5	CCLR[2:0]	000	R/W	计数器清除 2、1、0 选择 TCNT 的计数器清除源。详情参照表 11.4、表 11.5。
4、3	CKEG[1:0]	00	R/W	时钟边沿 1、0 选择输入时钟的边沿。在双边沿计数内部时钟时，输入时钟的周期为 1/2（例如：P ϕ /4 的双边沿 = P ϕ /2 的上升沿）。在通道 1、2 使用相位计数模式时，忽略本设定，优先相位计数模式的设定。内部时钟的边沿选择在输入时钟为 P ϕ /4 或迟于 P ϕ /4 时有效。选择 P ϕ /1 或其他通道的上溢 / 下溢为输入时钟时，可写入值，但运行为初始值。 00: 在上升沿计数 01: 在下降沿计数 1x: 在双边沿计数
2 ~ 0	TPSC[2:0]	000	R/W	定时器预分频器 2、1、0 选择 TCNT 的计数器时钟。各通道可独立选择时钟源。详情参照表 11.6 ~ 表 11.9。

【符号说明】 x: Don't care

表 11.4 CCLR2 ~ CCLR0 (通道 0、3、4)

通道	bit7	bit6	bit5	说 明
	CCLR2	CCLR1	CCLR0	
0、3、4	0	0	0	禁止清除 TCNT
	0	0	1	通过 TGRA 的比较匹配 / 输入捕捉清除 TCNT
	0	1	0	通过 TGRB 的比较匹配 / 输入捕捉清除 TCNT
	0	1	1	通过清除正在同步清除 / 同步运行的其他通道的计数器，清除 TCNT*1
	1	0	0	禁止清除 TCNT
	1	0	1	通过 TGRC 的比较匹配 / 输入捕捉清除 TCNT*2
	1	1	0	通过 TGRD 的比较匹配 / 输入捕捉清除 TCNT*2
	1	1	1	通过清除正在同步清除 / 同步运行的其他通道的计数器，清除 TCNT*1

【注】 *1 通过将 TSYR 的 SYNC 位置 1，设定同步运行。

*2 TGRC 或 TGRD 用作缓冲寄存器时，优先缓冲寄存器的设定，并且不产生比较匹配 / 输入捕捉，因此不可清除 TCNT。

表 11.5 CCLR2 ~ CCLR0 (通道 1、2)

通道	bit7	bit6	bit5	说 明
	保留位 *2	CCLR1	CCLR0	
1、2	0	0	0	禁止清除 TCNT
	0	0	1	通过 TGRA 的比较匹配 / 输入捕捉清除 TCNT
	0	1	0	通过 TGRB 的比较匹配 / 输入捕捉清除 TCNT
	0	1	1	通过清除正在同步清除 / 同步运行的其他通道的计数器, 清除 TCNT*1

【注】 *1 通过将 TSYSR 的 SYNC 位置 1, 设定同步运行。

*2 通道 1、2 的 bit7 为保留位。读取值总是为 0, 写入无效。

表 11.6 TPSC2 ~ TPSC0 (通道 0)

通道	bit2	bit1	bit0	说 明
	TPSC2	TPSC1	TPSC0	
0	0	0	0	内部时钟: 以 P ϕ /1 计数
	0	0	1	内部时钟: 以 P ϕ /4 计数
	0	1	0	内部时钟: 以 P ϕ /16 计数
	0	1	1	内部时钟: 以 P ϕ /64 计数
	1	0	0	外部时钟: 由 TCLKA 引脚输入计数
	1	0	1	外部时钟: 由 TCLKB 引脚输入计数
	1	1	0	外部时钟: 由 TCLKC 引脚输入计数
	1	1	1	外部时钟: 由 TCLKD 引脚输入计数

表 11.7 TPSC2 ~ TPSC0 (通道 1)

通道	bit2	bit1	bit0	说 明
	TPSC2	TPSC1	TPSC0	
1	0	0	0	内部时钟: 以 P ϕ /1 计数
	0	0	1	内部时钟: 以 P ϕ /4 计数
	0	1	0	内部时钟: 以 P ϕ /16 计数
	0	1	1	内部时钟: 以 P ϕ /64 计数
	1	0	0	外部时钟: 由 TCLKA 引脚输入计数
	1	0	1	外部时钟: 由 TCLKB 引脚输入计数
	1	1	0	内部时钟: 以 P ϕ /256 计数
	1	1	1	由 TCNT_2 的上溢 / 下溢计数

【注】 通道 1 为相位计数模式时, 此设定无效。

表 11.8 TPSC2 ~ TPSC0 (通道 2)

通道	bit2	bit1	bit0	说 明
	TPSC2	TPSC1	TPSC0	
2	0	0	0	内部时钟: 以 $P\phi/1$ 计数
	0	0	1	内部时钟: 以 $P\phi/4$ 计数
	0	1	0	内部时钟: 以 $P\phi/16$ 计数
	0	1	1	内部时钟: 以 $P\phi/64$ 计数
	1	0	0	外部时钟: 由 TCLKA 引脚输入计数
	1	0	1	外部时钟: 由 TCLKB 引脚输入计数
	1	1	0	外部时钟: 由 TCLKC 引脚输入计数
	1	1	1	内部时钟: 以 $P\phi/1024$ 计数

【注】 通道 2 为相位计数模式时, 此设定无效。

表 11.9 TPSC2 ~ TPSC0 (通道 3、4)

通道	bit2	bit1	bit0	说 明
	TPSC2	TPSC1	TPSC0	
3、4	0	0	0	内部时钟: 以 $P\phi/1$ 计数
	0	0	1	内部时钟: 以 $P\phi/4$ 计数
	0	1	0	内部时钟: 以 $P\phi/16$ 计数
	0	1	1	内部时钟: 以 $P\phi/64$ 计数
	1	0	0	内部时钟: 以 $P\phi/256$ 计数
	1	0	1	内部时钟: 以 $P\phi/1024$ 计数
	1	1	0	外部时钟: 由 TCLKA 引脚输入计数
	1	1	1	外部时钟: 由 TCLKB 引脚输入计数

11.3.2 定时器模式寄存器 (TMDR)

TMDR 为 8 位可读取 / 写入的寄存器，可设定各通道的运行模式。MTU2 共有 5 个 TMDR，通道 0 ~ 4 各有 1 个。必须在 TCNT 运行停止状态设定 TMDR。

位:	7	6	5	4	3	2	1	0
	-	BFE	BFB	BFA	MD[3:0]			
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7	—	0	R	保留位 读取值、写入值总是为 0。
6	BFE	0	R/W	缓冲运行 E 选择 TGRE_0 和 TGRF_0 为通常运行或缓冲运行。 TGRF 用作缓冲寄存器时，产生 TGRF 的比较匹配。 在通道 1、2、3、4 为保留位。读取值、写入值总是为 0。 0: TGRE_0 和 TGRF_0 为通常运行 1: TGRE_0 和 TGRF_0 为缓冲运行
5	BFB	0	R/W	缓冲运行 B 设定 TGRB 为通常运行或组合 TGRB 和 TGRD 后缓冲运行。TGRD 用作缓冲寄存器时，除互补 PWM 模式之外，不产生 TGRD 的输入捕捉 / 输出比较，但在互补 PWM 模式产生 TGRD 的比较匹配。另外，在互补 PWM 模式的 Tb 区间产生比较匹配时，TGRD 置位，因此必须将定时器中断允许寄存器 3/4 (TIER_3/4) 的 TGIED 位设置为 0。 在无 TGRD 的通道 1、2，此位为保留位。读取值、写入值总是为 0。 0: TGRB 和 TGRD 为通常运行 1: TGRB 和 TGRD 为缓冲运行
4	BFA	0	R/W	缓冲运行 A 设定 TGRA 为通常运行或组合 TGRA 和 TGRC 后缓冲运行。TGRC 用作缓冲寄存器时，除互补 PWM 模式之外，不产生 TGRC 的输入捕捉 / 输出比较，但在互补 PWM 模式产生 TGRC 的比较匹配。另外，在互补 PWM 模式的 Tb 区间产生通道 4 的比较匹配时，TGFC 置位，因此必须将定时器中断允许寄存器_4 (TIER_4) 的 TGIEC 位设置为 0。 在无 TGRC 的通道 1、2，此位为保留位。读取值、写入值总是为 0。 0: TGRA 和 TGRC 为通常运行 1: TGRA 和 TGRC 为缓冲运行
3 ~ 0	MD[3:0]	0000	R/W	模式 3 ~ 0 MD3 ~ MD0 设定定时器的运行模式。 详情参照表 11.10。

表 11.10 通过 MD3 ~ MD0 位设定运行模式

bit3	bit2	bit1	bit0	说 明
MD3	MD2	MD1	MD0	
0	0	0	0	通常运行
0	0	0	1	禁止设定
0	0	1	0	PWM 模式 1
0	0	1	1	PWM 模式 2*1
0	1	0	0	相位计数模式 1*2
0	1	0	1	相位计数模式 2*2
0	1	1	0	相位计数模式 3*2
0	1	1	1	相位计数模式 4*2
1	0	0	0	复位同步 PWM 模式 *3
1	0	0	1	禁止设定
1	0	1	x	禁止设定
1	1	0	0	禁止设定
1	1	0	1	互补 PWM 模式 1 (在波峰传送) *3
1	1	1	0	互补 PWM 模式 2 (在波谷传送) *3
1	1	1	1	互补 PWM 模式 3 (在波峰 / 波谷传送) *3

【符号说明】 x: Don't care

【注】 *1 在通道 3、4，不可设定 PWM 模式 2。

*2 通道 0、3、4，不可设定相位计数模式。

*3 仅可在通道 3 设定复位同步 PWM 模式或互补 PWM 模式。

将通道 3 设定为复位同步 PWM 模式或互补 PWM 模式时，通道 4 的设定无效，自动遵从通道 3 的设定。但不得对通道 4 设定复位同步 PWM 模式或互补 PWM 模式。

在通道 0、1、2，不可设定复位同步 PWM 模式或互补 PWM 模式。

11.3.3 定时器 I/O 控制寄存器 (TIOR)

TIOR 为控制 TGR 的 8 位可读取 / 写入的寄存器。MTU2 共有 8 个 TIOR，通道 0、3、4 各有 2 个，通道 1、2 各有 1 个。

TMDR 设定为普通运行、PWM 模式、相位系数模式时，可设定 TIOR。

由 TIOR 指定的初始输出在计数器停止 (TSTR 的 CST 位清 0 后) 状态有效。在 PWM 模式 2，指定计数器清 0 时的输出。

将 TGRC 或 TGRD 设定为缓冲运行时，本设定无效，作为缓冲寄存器运行。

- TIORH_0、TIOR_1、TIOR_2、TIORH_3、TIORH_4

位:	7	6	5	4	3	2	1	0
	IOB[3:0]				IOA[3:0]			
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7 ~ 4	IOB[3:0]	0000	R/W	I/O 控制 B3 ~ B0 IOB3 ~ IOB0 位设定 TGRB 的功能。 请参照下列各表: TIORH_0: 表 11.11 TIOR_1: 表 11.13 TIOR_2: 表 11.14 TIORH_3: 表 11.15 TIORH_4: 表 11.17
3 ~ 0	IOA[3:0]	0000	R/W	I/O 控制 A3 ~ A0 IOA3 ~ IOA0 位设定 TGRA 的功能。 请参照下列各表: TIORH_0: 表 11.19 TIOR_1: 表 11.21 TIOR_2: 表 11.22 TIORH_3: 表 11.23 TIORH_4: 表 11.25

- TIORL_0、TIORL_3、TIORL_4

位:	7	6	5	4	3	2	1	0
	IOD[3:0]				IOC[3:0]			
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7 ~ 4	IOD[3:0]	0000	R/W	I/O 控制 D3 ~ D0 IOD3 ~ IOD0 位设定 TGRD 的功能。 请参照下列各表: TIORL_0: 表 11.12 TIORL_3: 表 11.16 TIORL_4: 表 11.18
3 ~ 0	IOC[3:0]	0000	R/W	I/O 控制 C3 ~ C0 IOC3 ~ IOC0 位设定 TGRC 的功能。 请参照下列各表: TIORL_0: 表 11.20 TIORL_3: 表 11.24 TIORL_4: 表 11.26

表 11.11 TIORH_0 (通道 0)

bit7	bit6	bit5	bit4	说 明	
IOB3	IOB2	IOB1	IOB0	TGRB_0 的功能	TIOC0B 引脚的功能
0	0	0	0	输出比较寄存器	保持输出 *
0	0	0	1		初始输出时输出 0 比较匹配时输出 0
0	0	1	0		初始输出时输出 0 比较匹配时输出 1
0	0	1	1		初始输出时输出 0 比较匹配时交替输出
0	1	0	0		保持输出
0	1	0	1		初始输出时输出 1 比较匹配时输出 0
0	1	1	0		初始输出时输出 1 比较匹配时输出 1
0	1	1	1		初始输出时输出 1 比较匹配时交替输出
1	0	0	0	输入捕捉寄存器	在上升沿输入捕捉
1	0	0	1		在下降沿输入捕捉
1	0	1	x		在双边沿输入捕捉
1	1	x	x		捕捉输入源为通道 1/ 计数时钟 由 TCNT_1 的递增 / 递减计数输入捕捉

【符号说明】x: Don't care

【注】* 上电复位后到设定 TIOR 前, 输出 0。

表 11.12 TIORL_0 (通道 0)

bit7	bit6	bit5	bit4	说 明	
IOD3	IOD2	IOD1	IOD0	TGRD_0 的功能	TIOC0D 引脚的功能
0	0	0	0	输出比较寄存器 *2	保持输出 *1
0	0	0	1		初始输出时输出 0 比较匹配时输出 0
0	0	1	0		初始输出时输出 0 比较匹配时输出 1
0	0	1	1		初始输出时输出 0 比较匹配时交替输出
0	1	0	0		保持输出
0	1	0	1		初始输出时输出 1 比较匹配时输出 0
0	1	1	0		初始输出时输出 1 比较匹配时输出 1
0	1	1	1		初始输出时输出 1 比较匹配时交替输出
1	0	0	0	输入捕捉寄存器 *2	在上升沿输入捕捉
1	0	0	1		在下降沿输入捕捉
1	0	1	x		在双边沿输入捕捉
1	1	x	x		捕捉输入源为通道 1/ 计数时钟 由 CNT_1 的递增 / 递减计数输入捕捉

【符号说明】x: Don't care

【注】*1 上电复位后到设定 TIOR 前, 输出 0。

*2 TMDR_0 的 BFB 位置 1, TGRD_0 用作缓冲寄存器时, 本设定无效, 不产生输入捕捉 / 输出比较。

表 11.13 TIOR_1 (通道 1)

bit7	bit6	bit5	bit4	说 明	
IOB3	IOB2	IOB1	IOB0	TGRB_1 的功能	TIOC1B 引脚的功能
0	0	0	0	TGRB_1 为输出比较寄存器	保持输出 *
0	0	0	1		初始输出时输出 0 比较匹配时输出 0
0	0	1	0		初始输出时输出 0 比较匹配时输出 1
0	0	1	1		初始输出时输出 0 比较匹配时交替输出
0	1	0	0		保持输出
0	1	0	1		初始输出时输出 1 比较匹配时输出 0
0	1	1	0		初始输出时输出 1 比较匹配时输出 1
0	1	1	1		初始输出时输出 1 比较匹配时交替输出
1	0	0	0	输入捕捉寄存器	在上升沿输入捕捉
1	0	0	1		在下降沿输入捕捉
1	0	1	x		在双边沿输入捕捉
1	1	x	x		产生 TGRC_0 的比较匹配 / 输入捕捉时输入捕捉

【符号说明】x: Don't care

【注】* 上电复位后到设定 TIOR 前, 输出 0。

表 11.14 TIOR_2 (通道 2)

bit7	bit6	bit5	bit4	说 明	
IOB3	IOB2	IOB1	IOB0	TGRB_2 的功能	TIOC2B 引脚的功能
0	0	0	0	TGRB_2 为输出比较寄存器	保持输出 *
0	0	0	1		初始输出时输出 0 比较匹配时输出 0
0	0	1	0		初始输出时输出 0 比较匹配时输出 1
0	0	1	1		初始输出时输出 0 比较匹配时交替输出
0	1	0	0		保持输出
0	1	0	1		初始输出时输出 1 比较匹配时输出 0
0	1	1	0		初始输出时输出 1 比较匹配时输出 1
0	1	1	1		初始输出时输出 1 比较匹配时交替输出
1	x	0	0	输入捕捉寄存器	在上升沿输入捕捉
1	x	0	1		在下降沿输入捕捉
1	x	1	x		在双边沿输入捕捉

【符号说明】x: Don't care

【注】* 上电复位后到设定 TIOR 前, 输出 0。

表 11.15 TIORH_3 (通道 3)

bit7	bit6	bit5	bit4	说 明	
IOB3	IOB2	IOB1	IOB0	TGRB_3 的功能	TIOC3B 引脚的功能
0	0	0	0	输出比较寄存器	保持输出 *
0	0	0	1		初始输出时输出 0 比较匹配时输出 0
0	0	1	0		初始输出时输出 0 比较匹配时输出 1
0	0	1	1		初始输出时输出 0 比较匹配时交替输出
0	1	0	0		保持输出
0	1	0	1		初始输出时输出 1 比较匹配时输出 0
0	1	1	0		初始输出时输出 1 比较匹配时输出 1
0	1	1	1		初始输出时输出 1 比较匹配时交替输出
1	x	0	0	输入捕捉寄存器	在上升沿输入捕捉
1	x	0	1		在下降沿输入捕捉
1	x	1	x		在双边沿输入捕捉

【符号说明】x: Don't care

【注】 * 上电复位后到设定 TIOR 前, 输出 0。

表 11.16 TIORL_3 (通道 3)

bit7	bit6	bit5	bit4	说 明	
IOD3	IOD2	IOD1	IOD0	TGRD_3 的功能	TIOC3D 引脚的功能
0	0	0	0	输出比较寄存器 *2	保持输出 *1
0	0	0	1		初始输出时输出 0 比较匹配时输出 0
0	0	1	0		初始输出时输出 0 比较匹配时输出 1
0	0	1	1		初始输出时输出 0 比较匹配时交替输出
0	1	0	0		保持输出
0	1	0	1		初始输出时输出 1 比较匹配时输出 0
0	1	1	0		初始输出时输出 1 比较匹配时输出 1
0	1	1	1		初始输出时输出 1 比较匹配时交替输出
1	x	0	0	输入捕捉寄存器 *2	在上升沿输入捕捉
1	x	0	1		在下降沿输入捕捉
1	x	1	x		在双边沿输入捕捉

【符号说明】x: Don't care

【注】 *1 上电复位后到设定 TIOR 前, 输出 0。

*2 TMDR_3 的 BFB 位置 1, TGRD_3 用作缓冲寄存器时, 本设定无效, 不产生输入捕捉 / 输出比较。

表 11.17 TIORH_4 (通道 4)

bit7	bit6	bit5	bit4	说 明	
IOB3	IOB2	IOB1	IOB0	TGRB_4 的功能	TIOC4B 引脚的功能
0	0	0	0	TGRB_4 为输出比较寄存器	保持输出 *
0	0	0	1		初始输出时输出 0 比较匹配时输出 0
0	0	1	0		初始输出时输出 0 比较匹配时输出 1
0	0	1	1		初始输出时输出 0 比较匹配时交替输出
0	1	0	0		保持输出
0	1	0	1		初始输出时输出 1 比较匹配时输出 0
0	1	1	0		初始输出时输出 1 比较匹配时输出 1
0	1	1	1		初始输出时输出 1 比较匹配时交替输出
1	x	0	0	输入捕捉寄存器	在上升沿输入捕捉
1	x	0	1		在下降沿输入捕捉
1	x	1	x		在双边沿输入捕捉

【符号说明】x: Don't care

【注】 * 上电复位后到设定 TIOR 前, 输出 0。

表 11.18 TIORL_4 (通道 4)

bit7	bit6	bit5	bit4	说 明	
IOD3	IOD2	IOD1	IOD0	TGRD_4 的功能	TIOC4D 引脚的功能
0	0	0	0	输出比较寄存器 *2	保持输出 *1
0	0	0	1		初始输出时输出 0 比较匹配时输出 0
0	0	1	0		初始输出时输出 0 比较匹配时输出 1
0	0	1	1		初始输出时输出 0 比较匹配时交替输出
0	1	0	0		保持输出
0	1	0	1		初始输出时输出 1 比较匹配时输出 0
0	1	1	0		初始输出时输出 1 比较匹配时输出 1
0	1	1	1		初始输出时输出 1 比较匹配时交替输出
1	x	0	0	输入捕捉寄存器 *2	在上升沿输入捕捉
1	x	0	1		在下降沿输入捕捉
1	x	1	x		在双边沿输入捕捉

【符号说明】x: Don't care

【注】 *1 上电复位后到设定 TIOR 前, 输出 0。

*2 TMDR_4 的 BFB 位置 1, TGRD_4 用作缓冲寄存器时, 本设定无效, 不产生输入捕捉 / 输出比较。

表 11.19 TIORH_0 (通道 0)

bit3	bit2	bit1	bit0	说 明	
IOA3	IOA2	IOA1	IOA0	TGRA_0 的功能	TIOC0A 引脚的功能
0	0	0	0	输出比较寄存器	保持输出 *
0	0	0	1		初始输出时输出 0 比较匹配时输出 0
0	0	1	0		初始输出时输出 0 比较匹配时输出 1
0	0	1	1		初始输出时输出 0 比较匹配时交替输出
0	1	0	0		保持输出
0	1	0	1		初始输出时输出 1 比较匹配时输出 0
0	1	1	0		初始输出时输出 1 比较匹配时输出 1
0	1	1	1		初始输出时输出 1 比较匹配时交替输出
1	0	0	0	输入捕捉寄存器	在上升沿输入捕捉
1	0	0	1		在下降沿输入捕捉
1	0	1	x		在双边沿输入捕捉
1	1	x	x		捕捉输入源为通道 1/ 计数时钟 由 TCNT_1 的递增 / 递减计数输入捕捉

【符号说明】 x: Don't care

【注】 * 上电复位后到设定 TIOR 前, 输出 0。

表 11.20 TIORL_0 (通道 0)

bit3	bit2	bit1	bit0	说 明	
IOC3	IOC2	IOC1	IOC0	TGRC_0 的功能	TIOC0C 引脚的功能
0	0	0	0	输出比较寄存器 *2	保持输出 *1
0	0	0	1		初始输出时输出 0 比较匹配时输出 0
0	0	1	0		初始输出时输出 0 比较匹配时输出 1
0	0	1	1		初始输出时输出 0 比较匹配时交替输出
0	1	0	0		保持输出
0	1	0	1		初始输出时输出 1 比较匹配时输出 0
0	1	1	0		初始输出时输出 1 比较匹配时输出 1
0	1	1	1		初始输出时输出 1 比较匹配时交替输出
1	0	0	0	输入捕捉寄存器 *2	在上升沿输入捕捉
1	0	0	1		在下降沿输入捕捉
1	0	1	x		在双边沿输入捕捉
1	1	x	x		捕捉输入源为通道 1/ 计数时钟 由 TCNT_1 的递增 / 递减计数输入捕捉

【符号说明】 x: Don't care

【注】 *1 上电复位后到设定 TIOR 前, 输出 0。

*2 TMDR_0 的 BFA 位置 1, TGRC_0 用作缓冲寄存器时, 本设定无效, 不产生输入捕捉 / 输出比较。

表 11.21 TIOR_1 (通道 1)

bit3	bit2	bit1	bit0	说 明	
IOA3	IOA2	IOA1	IOA0	TGRA_1 的功能	TIOC1A 引脚的功能
0	0	0	0	输出比较寄存器	保持输出 *
0	0	0	1		初始输出时输出 0 比较匹配时输出 0
0	0	1	0		初始输出时输出 0 比较匹配时输出 1
0	0	1	1		初始输出时输出 0 比较匹配时交替输出
0	1	0	0		保持输出
0	1	0	1		初始输出时输出 1 比较匹配时输出 0
0	1	1	0		初始输出时输出 1 比较匹配时输出 1
0	1	1	1		初始输出为 1 输出 比较匹配时交替输出
1	0	0	0	输入捕捉寄存器	在上升沿输入捕捉
1	0	0	1		在下降沿输入捕捉
1	0	1	x		在双边沿输入捕捉
1	1	x	x		产生 TGRA_0 的比较匹配 / 输入捕捉时输入捕捉

【符号说明】x: Don't care

【注】* 上电复位后到设定 TIOR 前, 输出 0。

表 11.22 TIOR_2 (通道 2)

bit3	bit2	bit1	bit0	说 明	
IOA3	IOA2	IOA1	IOA0	TGRA_2 的功能	TIOC2A 引脚的功能
0	0	0	0	输出比较寄存器	保持输出 *
0	0	0	1		初始输出时输出 0 比较匹配时输出 0
0	0	1	0		初始输出时输出 0 比较匹配时输出 1
0	0	1	1		初始输出时输出 0 比较匹配时交替输出
0	1	0	0		保持输出
0	1	0	1		初始输出时输出 1 比较匹配时输出 0
0	1	1	0		初始输出时输出 1 比较匹配时输出 1
0	1	1	1		初始输出时输出 1 比较匹配时交替输出
1	x	0	0	输入捕捉寄存器	在上升沿输入捕捉
1	x	0	1		在下降沿输入捕捉
1	x	1	x		在双边沿输入捕捉

【符号说明】x: Don't care

【注】* 上电复位后到设定 TIOR 前, 输出 0。

表 11.23 TIORH_3 (通道 3)

bit3	bit2	bit1	bit0	说 明	
IOA3	IOA2	IOA1	IOA0	TGRA_3 的功能	TIOC3A 引脚的功能
0	0	0	0	输出比较寄存器	保持输出 *
0	0	0	1		初始输出时输出 0 比较匹配时输出 0
0	0	1	0		初始输出时输出 0 比较匹配时输出 1
0	0	1	1		初始输出时输出 0 比较匹配时交替输出
0	1	0	0		保持输出
0	1	0	1		初始输出时输出 1 比较匹配时输出 0
0	1	1	0		初始输出时输出 1 比较匹配时输出 1
0	1	1	1		初始输出时输出 1 比较匹配时交替输出
1	x	0	0	输入捕捉寄存器	在上升沿输入捕捉
1	x	0	1		在下降沿输入捕捉
1	x	1	x		在双边沿输入捕捉

【符号说明】x: Don't care

【注】 * 上电复位后到设定 TIOR 前, 输出 0。

表 11.24 TIORL_3 (通道 3)

bit3	bit2	bit1	bit0	说 明	
IOC3	IOC2	IOC1	IOC0	TGRC_3 的功能	TIOC3C 引脚的功能
0	0	0	0	输出比较寄存器 *2	保持输出 *1
0	0	0	1		初始输出时输出 0 比较匹配时输出 0
0	0	1	0		初始输出时输出 0 比较匹配时输出 1
0	0	1	1		初始输出时输出 0 比较匹配时交替输出
0	1	0	0		保持输出
0	1	0	1		初始输出时输出 1 比较匹配时输出 0
0	1	1	0		初始输出时输出 1 比较匹配时输出 1
0	1	1	1		初始输出时输出 1 比较匹配时交替输出
1	x	0	0	输入捕捉寄存器 *2	在上升沿输入捕捉
1	x	0	1		在下降沿输入捕捉
1	x	1	x		在双边沿输入捕捉

【符号说明】x: Don't care

【注】 *1 上电复位后到设定 TIOR 前, 输出 0。

*2 TMDR_3 的 BFA 位置 1, TGRC_3 用作缓冲寄存器时, 本设定无效, 不产生输入捕捉 / 输出比较。

表 11.25 TIORH_4 (通道 4)

bit3	bit2	bit1	bit0	说 明	
IOA3	IOA2	IOA1	IOA0	TGRA_4 的功能	TIOC4A 引脚的功能
0	0	0	0	输出比较寄存器	保持输出 *
0	0	0	1		初始输出时输出 0 比较匹配时输出 0
0	0	1	0		初始输出时输出 0 比较匹配时输出 1
0	0	1	1		初始输出时输出 0 比较匹配时交替输出
0	1	0	0		保持输出
0	1	0	1		初始输出时输出 1 比较匹配时输出 0
0	1	1	0		初始输出时输出 1 比较匹配时输出 1
0	1	1	1		初始输出时输出 1 比较匹配时交替输出
1	x	0	0	输入捕捉寄存器	在上升沿输入捕捉
1	x	0	1		在下降沿输入捕捉
1	x	1	x		在双边沿输入捕捉

【符号说明】x: Don't care

【注】* 上电复位后到设定 TIOR 前, 输出 0。

表 11.26 TIORL_4 (通道 4)

bit3	bit2	bit1	bit0	说 明	
IOC3	IOC2	IOC1	IOC0	TGRC_4 的功能	TIOC4C 引脚的功能
0	0	0	0	输出比较寄存器 *2	保持输出 *1
0	0	0	1		初始输出时输出 0 比较匹配时输出 0
0	0	1	0		初始输出时输出 0 比较匹配时输出 1
0	0	1	1		初始输出时输出 0 比较匹配时交替输出
0	1	0	0		保持输出
0	1	0	1		初始输出时输出 1 比较匹配时输出 0
0	1	1	0		初始输出时输出 1 比较匹配时输出 1
0	1	1	1		初始输出时输出 1 比较匹配时交替输出
1	x	0	0	输入捕捉寄存器 *2	在上升沿输入捕捉
1	x	0	1		在下降沿输入捕捉
1	x	1	x		在双边沿输入捕捉

【符号说明】x: Don't care

【注】*1 上电复位后到设定 TIOR 前, 输出 0。

*2 TMDR_4 的 BFA 位置 1, TGRC_4 用作缓冲寄存器时, 本设定无效, 不产生输入捕捉 / 输出比较。

11.3.4 定时器中断允许寄存器 (TIER)

TIER 为 8 位可读取 / 写入的寄存器，控制允许 / 禁止各通道的中断请求。MTU2 共有 6 个 TIER，通道 0 有 2 个，通道 1 ~ 4 各有 1 个。

- TIER_0、TIER_1、TIER_2、TIER_3、TIER_4

位:	7	6	5	4	3	2	1	0
	TTGE	TTGE2	TCIEU	TCIEV	TGIED	TGIEC	TGIEB	TGIEA
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7	TTGE	0	R/W	A/D 转换开始请求允许 允许或禁止由 TGRA 的输入捕捉 / 比较匹配产生 A/D 转换开始请求。 0: 禁止产生 A/D 转换开始请求 1: 允许产生 A/D 转换开始请求
6	TTGE2	0	R/W	A/D 转换开始请求允许 2 在互补 PWM 模式，允许或禁止由 TCNT_4 下溢（波谷）产生 A/D 转换请求。 在通道 0 ~ 3 为保留位。读取值、写入值总是为 0。 0: 禁止由 TCNT_4 下溢（波谷）产生 A/D 转换请求 1: 允许由 TCNT_4 下溢（波谷）产生 A/D 转换请求
5	TCIEU	0	R/W	下溢中断允许 在通道 1、2，TSR 的 TCFU 标志置 1 时，允许或禁止由 TCFU 标志产生中断请求（TCIU）。 在通道 0、3、4 为保留位。读取值、写入值总是为 0。 0: 禁止由 TCFU 产生中断请求（TCIU） 1: 允许由 TCFU 产生中断请求（TCIU）
4	TCIEV	0	R/W	上溢中断允许 TSR 的 TCFV 标志置 1 时，允许或禁止由 TCFV 标志产生中断请求（TCIV）。 0: 禁止由 TCFV 产生中断请求（TCIV） 1: 允许由 TCFV 产生中断请求（TCIV）
3	TGIED	0	R/W	TGR 中断允许 D 在通道 0、3、4，TSR 的 TGFD 位置 1 时，允许或禁止由 TGFD 位产生中断请求（TGID）。 在通道 1、2 为保留位。读取值、写入值总是为 0。 0: 禁止由 TGFD 位产生中断请求（TGID） 1: 允许由 TGFD 位产生中断请求（TGID）
2	TGIEC	0	R/W	TGR 中断允许 C 在通道 0、3、4，TSR 的 TGFC 位置 1 时，允许或禁止由 TGFC 位产生中断请求（TGIC）。 在通道 1、2 为保留位。读取值、写入值总是为 0。 0: 禁止由 TGFC 位产生中断请求（TGIC） 1: 允许由 TGFC 位产生中断请求（TGIC）

位	位名称	初始值	R/W	说 明
1	TGIEB	0	R/W	TGR 中断允许 B TSR 的 TGFB 位置 1 时, 允许或禁止由 TGFB 位产生中断请求 (TGIB)。 0: 禁止由 TGFB 位产生中断请求 (TGIB) 1: 允许由 TGFB 位产生中断请求 (TGIB)
0	TGIEA	0	R/W	TGR 中断允许 A TSR 的 TGFA 位置 1 时, 允许或禁止由 TGFA 位产生中断请求 (TGIA)。 0: 禁止由 TGFA 位产生中断请求 (TGIA) 1: 允许由 TGFA 位产生中断请求 (TGIA)

• TIER2_0

位:	7	6	5	4	3	2	1	0
	TTGE2	-	-	-	-	-	TGIEF	TGIEE
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R/W	R/W

位	位名称	初始值	R/W	说 明
7	TTGE2	0	R/W	A/D 转换开始请求允许 2 允许或禁止由 TCNT_0 和 TGRE_0 的比较匹配产生 A/D 转换开始请求。 0: 禁止由 TCNT_0 和 TGRE_0 的比较匹配产生 A/D 转换开始请求 1: 允许由 TCNT_0 和 TGRE_0 的比较匹配产生 A/D 转换开始请求
6 ~ 2	—	均为 0	R	保留位 读取值、写入值总是为 0。
1	TGIEF	0	R/W	TGR 中断允许 F 允许或禁止由 TCNT_0 和 TGRF_0 的比较匹配产生中断请求。 0: 禁止由 TGFE 位产生中断请求 (TGIF) 1: 允许由 TGFE 位产生中断请求 (TGIF)
0	TGIEE	0	R/W	TGR 中断允许 E 允许或禁止由 TCNT_0 和 TGRE_0 的比较匹配产生中断请求。 0: 禁止由 TGEE 位产生中断请求 (TGIE) 1: 允许由 TGEE 位产生中断请求 (TGIE)

11.3.5 定时器状态寄存器 (TSR)

TSR 为 8 位可读取 / 写入的寄存器，表示各通道的状态。MTU2 共有 6 个 TSR，通道 0 有 2 个，通道 1 ~ 4 各有 1 个。

- TSR_0、TSR_1、TSR_2、TSR_3、TSR_4

位:	7	6	5	4	3	2	1	0
	TCFD	-	TCFU	TCFV	TGFD	TGFC	TGFB	TGFA
初始值:	1	1	0	0	0	0	0	0
R/W:	R	R	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1

【注】 *1 为了清除标志，仅可在读取1后写入0。

位	位名称	初始值	R/W	说 明
7	TCFD	1	R	计数方向标志 表示通道 1 ~ 4 的 TCNT 计数方向的状态标志。 在通道 0 为保留位。读取值、写入值总是为 1。 0: TCNT 递减计数 1: TCNT 递增计数
6	—	1	R	保留位 读取值、写入值总是为 1。
5	TCFU	0	R/(W)*1	下溢标志 表示通道 1、2 为相位计数模式时，产生 TCNT 下溢的状态标志。为了清除标志，仅可写入 0。 在通道 0、3、4 为保留位。读取值、写入值总是为 0。 [清除条件] • 在 TCFU=1 的状态读取 TCFU 后，对 TCFU 写入 0 时 *2 [置位条件] • TCNT 的值下溢 (H'0000→H'FFFF) 时
4	TCFV	0	R/(W)*1	上溢标志 表示产生 TCNT 上溢的状态标志。为了清除标志，仅可写入 0。 [清除条件] • 在 TCFV=1 状态读取 TCFV 后，对 TCFV 写入 0 时 *2 [置位条件] • TCNT 的值上溢 (H'FFFF→H'0000) 时 通道 4 在互补 PWM 模式时，如果 TCNT_4 的值下溢 (H'0001→H'0000)，则本标志也置位。
3	TGFD	0	R/(W)*1	输入捕捉 / 输出比较标志 D 表示通道 0、3、4 产生 TGRD 输入捕捉或比较匹配的状态标志。为了清除标志，仅可写入 0。在通道 1、2 为保留位。读取值、写入值总是为 0。 [清除条件] • 在 TGFD=1 的状态读取 TGFD 后，对 TGFD 写入 0 时 *2 [置位条件] • TGRD 用作输出比较寄存器，且 TCNT=TGRD 时 • TGRD 用作输入捕捉寄存器，且通过输入捕捉信号向 TGRD 传送 TCNT 的值时

位	位名称	初始值	R/W	说 明
2	TGFC	0	R/(W)*1	输入捕捉 / 输出比较标志 C 表示通道 0、3、4 产生 TGRC 输入捕捉或比较匹配的状态标志。为了清除标志，仅可写入 0。在通道 1、2 为保留位。读取值、写入值总是为 0。 [清除条件] - 在 TGFC=1 的状态读取 TGFC 后，对 TGFC 写入 0 时 *2 [置位条件] - TGRC 用作输出比较寄存器，且 TCNT=TGRC 时 - TGRC 用作输入捕捉寄存器，且通过输入捕捉信号向 TGRC 传送 TCNT 的值时
1	TGFB	0	R/(W)*1	输入捕捉 / 输出比较标志 B 表示产生 TGRB 输入捕捉或比较匹配的状态标志。为了清除标志，仅可写入 0。 [清除条件] - 在 TGFB=1 的状态读取 TGFB 后，对 TGFB 写入 0 时 *2 [置位条件] - TGRB 用作输出比较寄存器，且 TCNT=TGRB 时 - TGRB 用作输入捕捉寄存器，且通过输入捕捉信号向 TGRB 传送 TCNT 的值时
0	TGFA	0	R/(W)*1	输入捕捉 / 输出比较标志 A 表示产生 TGRA 输入捕捉或比较匹配的状态标志。为了清除标志，仅可写入 0。 [清除条件] - 由 TGIA 中断启动 DMAC 时 - 在 TGFA=1 的状态读取 TGFA 后，对 TGFA 写入 0 时 *2 [置位条件] - TGRA 用作输出比较寄存器，且 TCNT=TGRA 时 - TGRA 用作输入捕捉寄存器，且通过输入捕捉信号向 TGRA 传送 TCNT 的值时

【注】 *1 为了清除标志，仅可在读取 1 后写入 0。

*2 写入定时器状态寄存器 (TSR) 时，必须在读取 1 后仅对要清除的位写入 0，对其他位写入 1。但实际并不写入 1，而是保留之前的值。

• TSR2_0

位:	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	TGFF	TGFE
初始值:	1	1	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/(W)*1	R/(W)*1

【注】 *1 为了清除标志，仅可在读取1后写入0。

位	位名称	初始值	R/W	说 明
7、6	—	均为 1	R	保留位 读取值、写入值总是为 1。
5 ~ 2	—	均为 0	R	保留位 读取值、写入值总是为 0。
1	TGFF	0	R/(W)*1	比较匹配标志 F 表示产生 TCNT_0 和 TGRF_0 比较匹配的状态标志。 [清除条件] • 在 TGFF=1 的状态读取 TGFF 后，对 TGFF 写入 0*2 [置位条件] • TGRF_0 用作比较寄存器，且 TCNT_0=TGRF_0 时
0	TGFE	0	R/(W)*1	比较匹配标志 E 表示产生 TCNT_0 和 TGRE_0 比较匹配的状态标志。 [清除条件] • 在 TGFE=1 的状态读取 TGFE 后，对 TGFE 写入 0*2 [置位条件] • TGRE_0 用作比较寄存器，且 TCNT_0=TGFE_0 时

【注】 *1 为了清除标志，仅可在读取 1 后写入 0。

*2 写入定时器状态寄存器 (TSR) 时，必须在读取 1 后仅对要清除的位写入 0，其他位写入 1。但实际并不写入 1，而是保留之前的值。

11.3.6 定时器缓冲运行传送模式寄存器 (TBTM)

TBTM 为 8 位可读取 / 写入的寄存器，设定从 PWM 模式的缓冲寄存器向定时器通用寄存器传送的时序。
MTU2 共有 3 个 TBTM，通道 0、3、4 各有 1 个。

位:	7	6	5	4	3	2	1	0
	-	-	-	-	-	TTSE	TTSB	TTSA
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7 ~ 3	—	均为 0	R	保留位 读取值、写入值总是为 0。
2	TTSE	0	R/W	时序选择 E 设定缓冲运行时从 TGRF_0 向 TGRE_0 传送的时序。 在通道 3、4 为保留位。读取值、写入值总是为 0。 在 PWM 模式以外使用通道 0 时，本位不得设定为 1。 0: 产生通道 0 的比较匹配 E 时 1: 清除 TCNT_0 时
1	TTSB	0	R/W	时序选择 B 设定各通道缓冲运行时从 TGRD 向 TGRB 传送的时序。 在 PWM 模式以外使用的通道，本位不得设定为 1。 0: 产生各通道的比较匹配 B 时 1: 清除各通道的 TCNT 时
0	TTSA	0	R/W	时序选择 A 设定各通道缓冲运行时从 TGRC 向 TGRA 传送的时序。 在 PWM 模式以外使用通道时，本位不得设定为 1。 0: 产生各通道的比较匹配 A 时 1: 清除各通道的 TCNT 时

11.3.7 定时器输入捕捉控制寄存器 (TICCR)

TICCR 为 8 位可读取 / 写入的寄存器, 控制 TCNT_1 和 TCNT_2 级联时的输入捕捉条件。MTU2 的通道 1 有 1 个 TICCR。

位:	7	6	5	4	3	2	1	0
	-	-	-	-	I2BE	I2AE	I1BE	I1AE
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7 ~ 4	—	均为 0	R	保留位 读取值、写入值总是为 0。
3	I2BE	0	R/W	输入捕捉允许 选择是否追加 TIOC2B 引脚作为 TGRB_1 的输入捕捉条件。 0: 不追加 TIOC2B 引脚作为 TGRB_1 的输入捕捉条件 1: 追加 TIOC2B 引脚作为 TGRB_1 的输入捕捉条件
2	I2AE	0	R/W	输入捕捉允许 选择是否追加 TIOC2A 引脚作为 TGRA_1 的输入捕捉条件。 0: 不追加 TIOC2A 引脚作为 TGRA_1 的输入捕捉条件 1: 追加 TIOC2A 引脚作为 TGRA_1 的输入捕捉条件
1	I1BE	0	R/W	输入捕捉允许 选择是否追加 TIOC1B 引脚作为 TGRB_2 的输入捕捉条件。 0: 不追加 TIOC1B 引脚作为 TGRB_2 的输入捕捉条件 1: 追加 TIOC1B 引脚作为 TGRB_2 的输入捕捉条件
0	I1AE	0	R/W	输入捕捉允许 选择是否追加 TIOC1A 引脚作为 TGRA_2 的输入捕捉条件。 0: 不追加 TIOC1A 引脚作为 TGRA_2 的输入捕捉条件 1: 追加 TIOC1A 引脚作为 TGRA_2 的输入捕捉条件

11.3.8 定时器 A/D 转换开始请求控制寄存器 (TADCR)

TADCR 为 16 位可读取 / 写入的寄存器，设定允许或禁止 A/D 转换开始请求及是否联动中断跳过与 A/D 转换开始请求。MTU2 的通道 4 有 1 个 TADCR。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BF[1:0]		-	-	-	-	-	-	UT4AE	DT4AE	UT4BE	DT4BE	ITA3AE	ITA4VE	ITB3AE	ITB4VE
初始值:	0	0	0	0	0	0	0	0	0	0*	0	0*	0*	0*	0*	0*
R/W:	R/W	R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

【注】* 除互补PWM模式之外，不得设定为1。

位	位名称	初始值	R/W	说 明
15、14	BF[1:0]	00	R/W	选择 TADCOBRA/B_4 传送时序 选择从 TADCOBRA/B_4 向 TADCORA/B_4 传送的时序。 详情参照表 11.27。
13 ~ 8	—	均为 0	R	保留位 读取值、写入值总是为 0。
7	UT4AE	0	R/W	递增计数 TRG4AN 允许 TCNT_4 递增计数时，设定允许或禁止 A/D 转换开始请求 (TRG4AN)。 0: TCNT_4 递增计数时，禁止 A/D 转换开始请求 (TRG4AN) 1: TCNT_4 递增计数时，允许 A/D 转换开始请求 (TRG4AN)
6	DT4AE	0*	R/W	递减计数 TRG4AN 允许 TCNT_4 递减计数时，设定允许或禁止 A/D 转换开始请求 (TRG4AN)。 0: TCNT_4 递减计数时，禁止 A/D 转换开始请求 (TRG4AN) 1: TCNT_4 递减计数时，允许 A/D 转换开始请求 (TRG4AN)
5	UT4BE	0	R/W	递增计数 TRG4BN 允许 TCNT_4 递增计数时，设定允许或禁止 A/D 转换开始请求 (TRG4BN)。 0: TCNT_4 递增计数时，禁止 A/D 转换开始请求 (TRG4BN) 1: TCNT_4 递增计数时，允许 A/D 转换开始请求 (TRG4BN)
4	DT4BE	0*	R/W	递减计数 TRG4BN 允许 TCNT_4 递减计数时，设定允许或禁止 A/D 转换开始请求 (TRG4BN)。 0: TCNT_4 递减计数时，禁止 A/D 转换开始请求 (TRG4BN) 1: TCNT_4 递减计数时，允许 A/D 转换开始请求 (TRG4BN)
3	ITA3AE	0*	R/W	TGIA_3 中断跳过联动允许 选择 A/D 转换开始请求 (TRG4AN) 是否与 TGIA_3 中断跳过功能联动。 0: 不与 TGIA_3 中断跳过功能联动 1: 与 TGIA_3 中断跳过功能联动
2	ITA4VE	0*	R/W	TCIV_4 中断跳过联动允许 选择 A/D 转换开始请求 (TRG4AN) 是否与 TCIV_4 中断跳过功能联动。 0: 不与 TCIV_4 中断跳过功能联动 1: 与 TCIV_4 中断跳过功能联动
1	ITB3AE	0*	R/W	TGIA_3 中断跳过联动允许 选择 A/D 转换开始请求 (TRG4BN) 是否与 TGIA_3 中断跳过功能联动。 0: 不与 TGIA_3 中断跳过功能联动 1: 与 TGIA_3 中断跳过功能联动

位	位名称	初始值	R/W	说 明
0	ITB4VE	0*	R/W	TCIV_4 中断跳过联动允许 选择 A/D 转换开始请求 (TRG4BN) 是否与 TCIV_4 中断跳过功能联动。 0: 不与 TCIV_4 中断跳过功能联动 1: 与 TCIV_4 中断跳过功能联动

- 【注】 1. 禁止以 8 位为单位存取 TADCR，必须以 16 位为单位存取。
2. 禁止中断跳过时（将定时器中断跳过设定寄存器 (TITCR) 的 T3AEN、T4VEN 位设定为 0，或将 TITCR 的跳过次数设定 (3ACOR 和 4VCOR) 设定为 0 时），不可设定为与中断跳过功能联动（将定时器 A/D 转换开始请求控制寄存器 (TADCR) 的 ITA3AE、ITA4VE、ITB3AE、ITB4VE 位设定为 0）。
3. 禁止中断跳过时，如果已设定为与中断跳过联动，则不执行 A/D 转换开始请求。
- * 除互补 PWM 模式之外，不可设定为 1。

表 11.27 通过 BF1、BF0 位设定传送时序

bit7	bit6	说 明
BF1	BF0	
0	0	不从周期设定缓冲寄存器向周期设定寄存器传送
0	1	在 TCNT_4 的波峰，从周期设定缓冲寄存器向周期设定寄存器传送 *1
1	0	在 TCNT_4 的波谷，从周期设定缓冲寄存器向周期设定寄存器传送 *2
1	1	在 TCNT_4 的波峰和波谷，从周期设定缓冲寄存器向周期设定寄存器传送 *2

- 【注】 *1 在互补 PWM 模式的 TCNT_4 的波峰及复位同步 PWM 模式的 TCNT_3 和 TGRA_3 比较匹配时，或在 PWM 模式 1/ 普通运行模式的 TCNT_4 和 TGRA_4 比较匹配时，从周期设定缓冲寄存器向周期设定寄存器传送。
- *2 除互补 PWM 模式之外，禁止设定。

11.3.9 定时器 A/D 转换开始请求周期设定寄存器 (TADCORA/B_4)

TADCORA/B_4 为 16 位可读取 / 写入的位寄存器。与 TCNT_4 匹配时，产生相应的 A/D 转换开始请求。
TADCORA/B_4 的初始值为 H'FFFF。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
初始值:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- 【注】 禁止以 8 位为单位存取 TADCORA/B_4，必须以 16 位为单位存取。

11.3.10 定时器 A/D 转换开始请求周期设定缓冲寄存器 (TADCOBRA/B_4)

TADCOBRA/B_4 为 16 位可读取 / 写入的位寄存器。在波峰或波谷从 TADCORA/B_4 的缓冲寄存器传送到 TADCORA/B_4。

TADCOBRA/B_4 的初始值为 H'FFFF。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
初始值:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

【注】 禁止以 8 位为单位存取 TADCOBRA/B_4，必须以 16 位为单位存取。

11.3.11 定时器计数器 (TCNT)

TCNT 为 16 位可读取 / 写入的位计数器，共有 5 个 TCNT，通道 0 ~ 4 各有 1 个。

禁止以 8 位为单位存取 TCNT，必须以 16 位为单位存取。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

【注】 禁止以 8 位为单位存取 TCNT，必须固定以 16 位为单位存取。

11.3.12 定时器通用寄存器 (TGR)

TGR 为 16 位可读取 / 写入的寄存器，共 18 个通用寄存器，通道 0 有 6 个，通道 1、2 各有 2 个，通道 3、4 各有 4 个。

TGRA、TGRB、TGRC、TGRD 为输出比较 / 输入捕捉复用的寄存器。通道 0、3、4 的 TGRC 和 TGRD 可设定为缓冲寄存器。TGR 和缓冲寄存器的组合为 TGRA-TGRC、TGRB-TGRD。

TGRE_0、TGRF_0 可用作比较寄存器。TCNT_0 与 TGRE_0 匹配时，可产生 A/D 转换开始请求。TGRF 可用作缓冲寄存器。TGR 和缓冲寄存器的组合为 TGRE-TGRF。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
初始值:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

【注】 禁止以 8 位单位存取 TGR，必须以 16 位单位存取。TGR 的初始值为 H'FFFF。

11.3.13 定时器开始寄存器 (TSTR)

TSTR 为 8 位可读取 / 写入的寄存器，选择通道 0 ~ 4 的 TCNT 运行 / 停止。
 对 TMDR 设定运行模式或对 TCR 设定 TCNT 的计数时钟时，必须先停止 TCNT 的计数再进行设定。

位:	7	6	5	4	3	2	1	0
	CST4	CST3	-	-	-	CST2	CST1	CST0
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7	CST4	0	R/W	计数器开始 4、3 选择 TCNT 运行或停止。 在 TIOC 引脚为输出状态的运行过程中，对 CST 位写入 0 时，计数器停止运行，但保持 TIOC 引脚的输出比较输出电平。如果在 CST 位为 0 的状态写入 TIOR，引脚的输出电平则更新为已设定的初始输出值。 0: TCNT_4、TCNT_3 计数停止 1: TCNT_4、TCNT_3 计数运行
6	CST3	0	R/W	
5 ~ 3	—	均为 0	R	保留位 读取值、写入值总是为 0。
2	CST2	0	R/W	计数器开始 2 ~ 0 选择 TCNT 运行或停止。 在 TIOC 引脚为输出状态的运行过程中，对 CST 位写入 0 时，计数器停止运行，但保持 TIOC 引脚的输出比较输出电平。如果在 CST 位为 0 的状态写入 TIOR，引脚的输出电平则更新为已设定的初始输出值。 0: TCNT_2 ~ TCNT_0 计数停止 1: TCNT_2 ~ TCNT_0 计数运行
1	CST1	0	R/W	
0	CST0	0	R/W	

11.3.14 定时器同步寄存器 (TSYR)

TSYR 为 8 位可读取 / 写入的寄存器，选择通道 0 ~ 4 的 TCNT 独立运行或同步运行。相应位置 1 后的通道执行同步运行。

位:	7	6	5	4	3	2	1	0
	SYNC4	SYNC3	-	-	-	SYNC2	SYNC1	SYNC0
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7	SYNC4	0	R/W	定时器同步 4、3
6	SYNC3	0	R/W	选择与其他通道的独立运行或同步运行。 选择同步运行时，可同步预置多个 TCNT，并可通过清除其他通道的计数器执行同步清除。 设定同步运行时，至少需将 2 个通道的 SYNC 位置 1。设定同步清除时，除 SYNC 位之外，还需通过 TCR 的 CCLR2 ~ CCLR0 位设定 TCNT 的清除源。 0: TCNT_4、TCNT_3 独立运行 (TCNT 的预置 / 清除与其他通道无关) 1: TCNT_4、TCNT_3 同步运行 可同步预置 / 同步清除 TCNT
5 ~ 3	—	均为 0	R	保留位 读取值、写入值总是为 0。
2	SYNC2	0	R/W	定时器同步 2 ~ 0
1	SYNC1	0	R/W	选择与其他通道的独立运行或同步运行。 选择同步运行时，可同步预置多个 TCNT，并可通过清除其他通道的计数器执行同步清除。 设定同步运行时，至少需将 2 个通道的 SYNC 位置 1。设定同步清除时，除 SYNC 位之外，还需通过 TCR 的 CCLR2 ~ CCLR0 位设定 TCNT 的清除源。 0: TCNT_2 ~ TCNT_0 独立运行 (TCNT 的预置 / 清除与其他通道无关) 1: TCNT_2 ~ TCNT_0 同步运行 可同步预置 / 同步清除 TCNT
0	SYNC0	0	R/W	

11.3.15 定时器读取 / 写入允许寄存器（TRWER）

TRWER 为 8 位可读取 / 写入的寄存器，设定允许或禁止存取通道 3、4 误写防止对象寄存器 / 计数器。

位:	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	RWE
初始值:	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R/W

位	位名称	初始值	R/W	说 明
7 ~ 1	—	均为 0	R	保留位 读取值、写入值总是为 0。
0	RWE	1	R/W	读取 / 写入允许 设定允许 / 禁止读取 / 写入误写防止寄存器。 0: 禁止读取 / 写入寄存器 1: 允许读取 / 写入寄存器 [清除条件] • 在 RWE=1 的状态读取 RWE 后，对 RWE 写入 0 时

- 误写防止的对象寄存器/计数器

TCR_3、4、TMDR_3、4、TIO RH_3、4、TIO RL_3、4、TIER_3、4、TGRA_3、4、TGRB_3、4、TOER、TOCR1、TOCR2、TGCR、TCDR、TDDR 及 TCNT_3、4，共计 22 个寄存器。

11.3.16 定时器输出主控允许寄存器（TOER）

TOER 为 8 位可读取 / 写入的寄存器，允许或禁止执行输出引脚 TIOC4D、TIOC4C、TIOC3D、TIOC4B、TIOC4A、TIOC3B 的输出设定。如果未设定 TOER 的各位，则这些引脚无法正确输出，必须在设定通道 3、4 的 TIOR 前，设定 TOER 的值。

位:	7	6	5	4	3	2	1	0
	-	-	OE4D	OE4C	OE3D	OE4B	OE4A	OE3B
初始值:	1	1	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7、6	—	均为 1	R	保留位 读取值、写入值总是为 1。
5	OE4D	0	R/W	主控允许 TIOC4D 允许 / 禁止 TIOC4D 引脚的 MTU2 输出。 0: 禁止 MTU2 输出（无效电平）* 1: 允许 MTU2 输出
4	OE4C	0	R/W	主控允许 TIOC4C 允许 / 禁止 TIOC4C 引脚的 MTU2 输出。 0: 禁止 MTU2 输出（无效电平）* 1: 允许 MTU2 输出
3	OE3D	0	R/W	主控允许 TIOC3D 允许 / 禁止 TIOC3D 引脚的 MTU2 输出。 0: 禁止 MTU2 输出（无效电平）* 1: 允许 MTU2 输出
2	OE4B	0	R/W	主控允许 TIOC4B 允许 / 禁止 TIOC4B 引脚的 MTU2 输出。 0: 禁止 MTU2 输出（无效电平）* 1: 允许 MTU2 输出
1	OE4A	0	R/W	主控允许 TIOC4A 允许 / 禁止 TIOC4A 引脚的 MTU2 输出。 0: 禁止 MTU2 输出（无效电平）* 1: 允许 MTU2 输出
0	OE3B	0	R/W	主控允许 TIOC3B 允许 / 禁止 TIOC3B 引脚的 MTU2 输出。 0: 禁止 MTU2 输出（无效电平）* 1: 允许 MTU2 输出

【注】* 无效电平取决于定时器输出控制寄存器 1/2（TOCR1/2）的设定。详情参阅“11.3.17 定时器输出控制寄存器 1（TOCR1）”和“11.3.18 定时器输出控制寄存器 2（TOCR2）”。如果在除互补 PWM 模式 / 复位同步 PWM 模式之外执行 MTU2 输出，则本位必须设定为 1，设定为 0 时，输出低电平。

11.3.17 定时器输出控制寄存器 1 (TOCR1)

TOCR1 为 8 位可读取 / 写入的寄存器，允许或禁止与互补 PWM 模式 / 复位同步 PWM 模式的 PWM 周期同步交替输出，并控制 PWM 输出电平的取反。

位:	7	6	5	4	3	2	1	0
	-	PSYE	-	-	TOCL	TOCS	OLSN	OLSP
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R	R/(W)*	R/W	R/W	R/W

【注】* 上电复位后，仅可写入1次1。不可在写入1后写入0。

位	位名称	初始值	R/W	说 明
7	—	0	R	保留位 读取值、写入值总是为 0。
6	PSYE	0	R/W	PWM 同步输出允许 设定允许 / 禁止与 PWM 同步的交替输出。 0: 禁止交替输出 1: 允许交替输出
5、4	—	均为 0	R	保留位 读取值、写入值总是为 0。
3	TOCL	0	R/(W)*	TOC 寄存器写入禁止位 *1 设定禁止 / 允许写入 TOCR1 寄存器的 TOCS 位、OLSN 位及 OLSP 位。 0: 允许写入 TOCS 位、OLSN 位、OLSP 位 1: 禁止写入 TOCS 位、OLSN 位、OLSP 位
2	TOCS	0	R/W	TOC 选择位 选择 TOCR1 有效或 TOCR2 有效，取决于互补 PWM 模式 / 复位同步 PWM 模式的输出电平的设定。 0: TOCR1 的设定有效 1: TOCR2 的设定有效
1	OLSN	0	R/W	输出电平选择 N*2 复位同步 PWM 模式 / 互补 PWM 模式时，选择反相输出电平。详情参照表 11.28。
0	OLSP	0	R/W	输出电平选择 P*2 复位同步 PWM 模式 / 互补 PWM 模式时，选择正相输出电平。详情参照表 11.29。

【注】 *1 将 TOCL 位设定为 1，可防止 CPU 失控时的错误写入。

*2 将 TOCS 位设定为 0，使本设定有效。

表 11.28 输出电平选择功能

bit1	功 能			
OLSN	初始输出	有效电平	比较匹配输出	
			递增计数	递减计数
0	高电平	低电平	高电平	低电平
1	低电平	高电平	低电平	高电平

【注】 反相波形的初始输出值，在计数开始并经过空载时间后，变为有效电平。

表 11.29 输出电平选择功能

bit0	功 能			
OLSP	初始输出	有效电平	比较匹配输出	
			递增计数	递减计数
0	高电平	低电平	低电平	高电平
1	低电平	高电平	高电平	低电平

OLSN=1、OLSP=1 时，互补 PWM 模式的输出例（1 个相位）如图 11.2 所示。

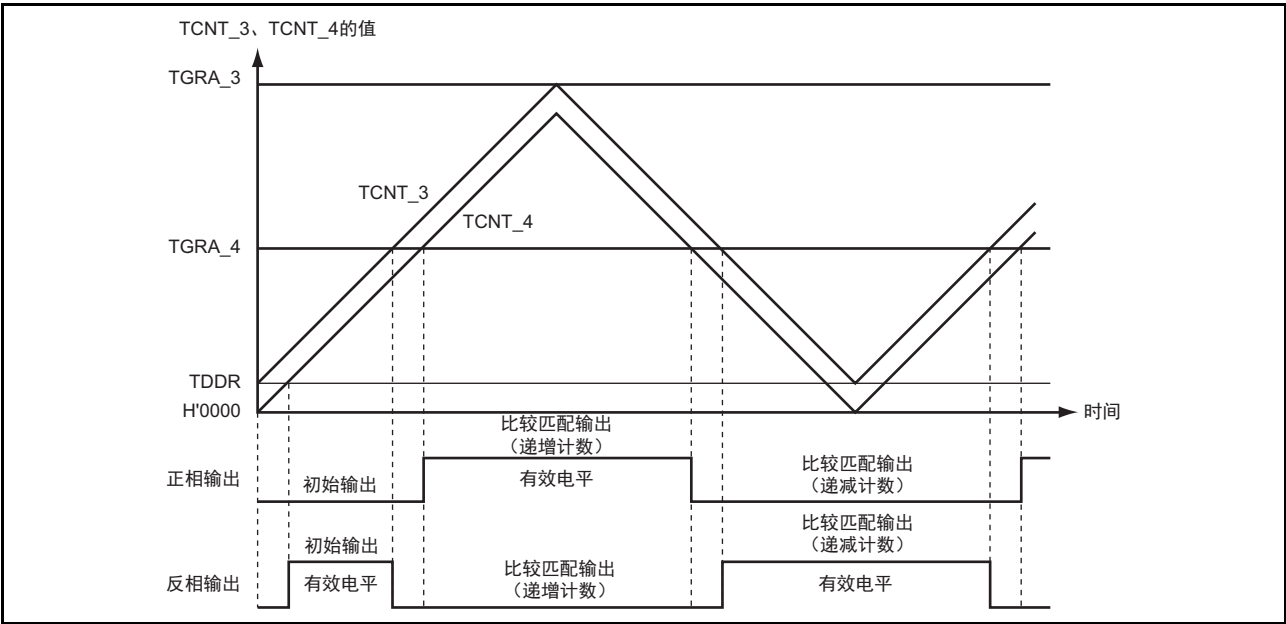


图 11.2 互补 PWM 模式的输出电平的例子

11.3.18 定时器输出控制寄存器 2 (TOCR2)

TOCR2 为 8 位可读取 / 写入的寄存器，在互补 PWM 模式 / 复位同步 PWM 模式，控制 PWM 输出电平的取反。

位:	7	6	5	4	3	2	1	0
	BF[1:0]		OLS3N	OLS3P	OLS2N	OLS2P	OLS1N	OLS1P
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7、6	BF[1:0]	00	R/W	选择 TOLBR 缓冲传送时序 选择从 TOLBR 向 TOCR2 的缓冲传送时序。 详情参照表 11.30。
5	OLS3N	0	R/W	输出电平选择 3N* 复位同步 PWM 模式 / 互补 PWM 模式时，选择 TIOC4D 的输出电平。详情参照表 11.31。
4	OLS3P	0	R/W	输出电平选择 3P* 复位同步 PWM 模式 / 互补 PWM 模式时，选择 TIOC4B 的输出电平。详情参照表 11.32。
3	OLS2N	0	R/W	输出电平选择 2N* 复位同步 PWM 模式 / 互补 PWM 模式时，选择 TIOC4C 的输出电平。详情参照表 11.33。
2	OLS2P	0	R/W	输出电平选择 2P* 复位同步 PWM 模式 / 互补 PWM 模式时，选择 TIOC4A 的输出电平。详情参照表 11.34。
1	OLS1N	0	R/W	输出电平选择 1N* 复位同步 PWM 模式 / 互补 PWM 模式时，选择 TIOC3D 的输出电平。详情参照表 11.35。
0	OLS1P	0	R/W	输出电平选择 1P* 复位同步 PWM 模式 / 互补 PWM 模式，选择 TIOC3B 的输出电平。详情参照表 11.36。

【注】 * 通过将 TOCR1 的 TOCS 位设定为 1，使本设定有效。

表 11.30 BF1、BF0 位的设定

bit7	bit6	说 明	
BF1	BF0	互补 PWM 模式	复位 PWM 模式
0	0	不从缓冲寄存器 (TOLBR) 向 TOCR2 传送	不从缓冲寄存器 (TOLBR) 向 TOCR2 传送
0	1	在 TCNT_4 的波峰，从缓冲寄存器 (TOLBR) 向 TOCR2 传送	清除 TCNT_3/4 计数器时，从缓冲寄存器 (TOLBR) 向 TOCR2 传送
1	0	在 TCNT_4 的波谷，从缓冲寄存器 (TOLBR) 向 TOCR2 传送	禁止设定
1	1	在 TCNT_4 的波峰和波谷，从缓冲寄存器 (TOLBR) 向 TOCR2 传送	禁止设定

表 11.31 TIOC4D 输出电平选择功能

bit5	功 能			
OLS3N	初始输出	有效电平	比较匹配输出	
			递增计数	递减计数
0	高电平	低电平	高电平	低电平
1	低电平	高电平	低电平	高电平

【注】 反相波形的初始输出值，在计数开始并经过空载时间后，变为有效电平。

表 11.32 TIOC4B 输出电平选择功能

bit4	功 能			
OLS3P	初始输出	有效电平	比较匹配输出	
			递增计数	递减计数
0	高电平	低电平	低电平	高电平
1	低电平	高电平	高电平	低电平

表 11.33 TIOC4C 输出电平选择功能

bit3	功 能			
OLS2N	初始输出	有效电平	比较匹配输出	
			递增计数	递减计数
0	高电平	低电平	高电平	低电平
1	低电平	高电平	低电平	高电平

【注】 反相波形的初始输出值，在计数开始并经过空载时间后，变为有效电平。

表 11.34 TIOC4A 输出电平选择功能

bit2	功 能			
OLS2P	初始输出	有效电平	比较匹配输出	
			递增计数	递减计数
0	高电平	低电平	低电平	高电平
1	低电平	高电平	高电平	低电平

表 11.35 TIOC3D 输出电平选择功能

bit1	功 能			
OLS1N	初始输出	有效电平	比较匹配输出	
			递增计数	递减计数
0	高电平	低电平	高电平	低电平
1	低电平	高电平	低电平	高电平

【注】 反相波形的初始输出值，在计数开始并经过空载时间后，变为有效电平。

表 11.36 TIOC4B 输出电平选择功能

bit0	功 能			
OLS1P	初始输出	有效电平	比较匹配输出	
			递增计数	递减计数
0	高电平	低电平	低电平	高电平
1	低电平	高电平	高电平	低电平

11.3.19 定时器输出电平缓冲寄存器（TOLBR）

TOLBR 为 TOCR2 的缓冲寄存器，设定互补 PWM 模式 / 复位同步 PWM 模式的 PWM 输出电平。TOLBR 寄存器为 8 位可读取 / 写入的寄存器。

位:	7	6	5	4	3	2	1	0
	-	-	OLS3N	OLS3P	OLS2N	OLS2P	OLS1N	OLS1P
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7、6	—	均为 0	R	保留位 读取值、写入值总是为 0。
5	OLS3N	0	R/W	必须对 TOCR2 的 OLS3N 位设定缓冲传送的值。
4	OLS3P	0	R/W	必须对 TOCR2 的 OLS3P 位设定缓冲传送的值。
3	OLS2N	0	R/W	必须对 TOCR2 的 OLS2N 位设定缓冲传送的值。
2	OLS2P	0	R/W	必须对 TOCR2 的 OLS2P 位设定缓冲传送的值。
1	OLS1N	0	R/W	必须对 TOCR2 的 OLS1N 位设定缓冲传送的值。
0	OLS1P	0	R/W	必须对 TOCR2 的 OLS1P 位设定缓冲传送的值。

缓冲运行时设定 PWM 输出电平的步骤如图 11.3 所示。

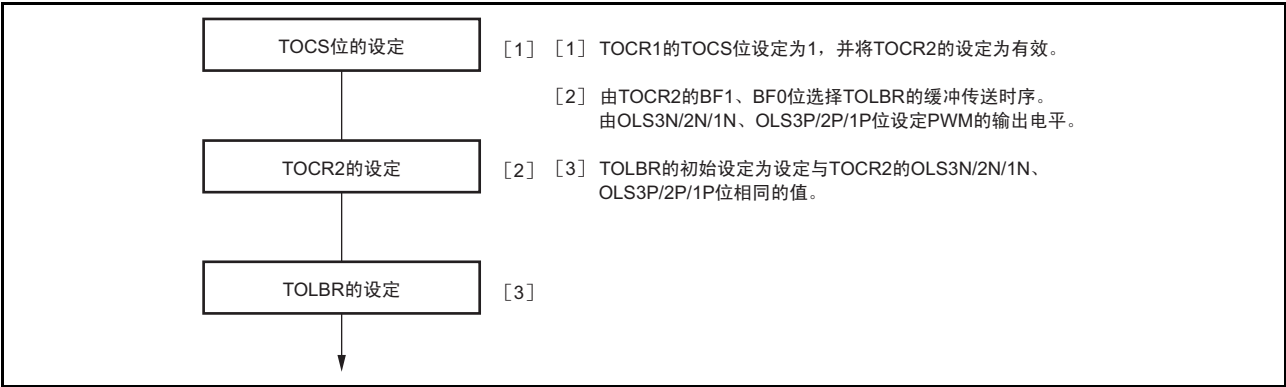


图 11.3 缓冲运行时设定 PWM 输出电平的步骤

11.3.20 定时器门控寄存器 (TGCR)

TGCR 在复位同步 PWM 模式 / 互补 PWM 模式, 可对无刷 DC 马达控制执行必要的波形输出控制。TGCR 寄存器为 8 位可读取 / 写入的寄存器。

除互补 PWM 模式 / 复位同步 PWM 模式之外, 本寄存器的设定均无效。

位:	7	6	5	4	3	2	1	0
	-	BDC	N	P	FB	WF	VF	UF
初始值:	1	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7	—	1	R	保留位 读取值、写入值总是为 1。
6	BDC	0	R/W	无刷 DC 马达 选择将本寄存器的功能设置为有效或无效。 0: 普通输出 1: 本寄存器的功能有效
5	N	0	R/W	反相输出控制 (N) 输出反相引脚 (TIOC3D 引脚、TIOC4C 引脚、TIOC4D 引脚) 时, 选择电平输出或复位同步 PWM/ 互补 PWM 输出。 0: 电平输出 1: 复位同步 PWM/ 互补 PWM 输出
4	P	0	R/W	正相输出控制 (P) 输出正相引脚 (TIOC3B 引脚、TIOC4A 引脚、TIOC4B 引脚) 时, 选择电平输出或复位同步 PWM/ 互补 PWM 输出。 0: 电平输出 1: 复位同步 PWM/ 互补 PWM 输出
3	FB	0	R/W	外部反馈信号允许 选择通过 MTU2/ 通道 0 的 TGRA、TGRB、TGRC 的输入捕捉信号自动转换正相 / 反相输出, 或通过对 TGCR 的 bit2 ~ 0 写入 0 或 1 自动转换正相 / 反相输出。 0: 通过外部输入 (输入源为通道 0 的 TGRA、TGRB、TGRC 的输入捕捉信号) 转换输出 1: 通过软件 (TGCR 的 UF、VF、WF 的设定值) 转换输出
2	WF	0	R/W	输出相位转换 2 ~ 0
1	VF	0	R/W	设定正相 / 反相输出相位的 ON、OFF。这些位的设定仅在本寄存器 FB 位为 1 时有效。此时, bit2 ~ 0 的设定取代外部输入。详情参照表 11.37。
0	UF	0	R/W	

表 11.37 输出电平选择功能

bit2 WF	bit1 VF	bit0 UF	功 能					
			TIOC3B	TIOC4A	TIOC4B	TIOC3D	TIOC4C	TIOC4D
			U 相	V 相	W 相	U 相	V 相	W 相
0	0	0	OFF	OFF	OFF	OFF	OFF	OFF
0	0	1	ON	OFF	OFF	OFF	OFF	ON
0	1	0	OFF	ON	OFF	ON	OFF	OFF
0	1	1	OFF	ON	OFF	OFF	OFF	ON
1	0	0	OFF	OFF	ON	OFF	ON	OFF
1	0	1	ON	OFF	OFF	OFF	ON	OFF
1	1	0	OFF	OFF	ON	ON	OFF	OFF
1	1	1	OFF	OFF	OFF	OFF	OFF	OFF

11.3.21 定时器副计数器 (TCNTS)

TCNTS 为 16 位只读计数器，仅可在互补 PWM 模式使用。TCNTS 的初始值为 H'0000。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

【注】 禁止以 8 位为单位存取 TCNTS，必须以 16 位单位存取。

11.3.22 定时器空载时间数据寄存器 (TDDR)

TDDR 为 16 位寄存器，仅可在互补 PWM 模式使用。在互补 PWM 模式，设定 TCNT_3、TCNT_4 计数器的偏移值。在互补 PWM 模式，清除 TCNT_3、TCNT_4 计数器后重新开始计数时，将 TDDR 寄存器的值加载到 TCNT_3 计数器并开始计数。TDDR 的初始值为 H'FFFF。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
初始值:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

【注】 禁止以 8 位为单位存取 TDDR，必须以 16 位单位存取。

11.3.23 定时器周期数据寄存器 (TCDR)

TCDR 为 16 位寄存器，仅可在互补 PWM 模式使用。TCDR 寄存器的值必须设定为 1/2PWM 载波周期。本寄存器在互补 PWM 模式，可随时和 TCNTS 计数器比较，如果匹配，TCNTS 计数器就转换计数方向（递减计数→递增计数）。TCDR 的初始值为 H'FFFF。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
初始值:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

【注】 禁止以 8 位为单位存取 TCDR，必须以 16 位单位存取。

11.3.24 定时器周期缓冲寄存器（TCBR）

TCBR 为 16 位寄存器，仅可在互补 PWM 模式使用，并可用作 TCDR 寄存器的缓冲寄存器。在 TMDR 寄存器设定的传送时序，TCBR 寄存器的值传送至 TCDR 寄存器。TCBR 的初始值为 H'FFFF。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
初始值:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

【注】 禁止以8位为单位存取TCBR，必须以16位单位存取。

11.3.25 定时器中断跳过设定寄存器（TITCR）

TITCR 为 8 位可读取 / 写入的寄存器，禁止 / 允许中断跳过，控制中断跳过次数的设定。MTU2 有 1 个 TITCR。

位:	7	6	5	4	3	2	1	0
	T3AEN	3ACOR[2:0]			T4VEN	4VCOR[2:0]		
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7	T3AEN	0	R/W	T3AEN 设定禁止 / 允许 TGIA_3 中断跳过。 0: 禁止 TGIA_3 中断跳过 1: 允许 TGIA_3 中断跳过
6 ~ 4	3ACOR[2:0]	000	R/W	设定 TGIA_3 中断跳过次数为 0 ~ 7 次。* 详情参照表 11.38。
3	T4VEN	0	R/W	T4VEN 设定禁止 / 允许 TCIV_4 中断跳过。 0: 禁止 TCIV_4 中断跳过 1: 允许 TCIV_4 中断跳过
2 ~ 0	4VCOR[2:0]	000	R/W	设定 TCIV_4 中断跳过次数为 0 ~ 7 次。* 详情参照表 11.39。

【注】 * 中断跳过次数设定为 0 时，不执行中断跳过。
更改中断跳过次数前，必须将 T3AEN、T4VEN 位设定为 0，并清除跳过次数计数器（TITCNT）。

表 11.38 通过 3ACOR2 ~ 3ACOR0 位设定中断跳过次数

bit6	bit5	bit4	说 明
3ACOR2	3ACOR1	3ACOR0	
0	0	0	不执行 TGIA_3 的中断跳过
0	0	1	将 TGIA_3 的中断跳过次数设定为 1 次
0	1	0	将 TGIA_3 的中断跳过次数设定为 2 次
0	1	1	将 TGIA_3 的中断跳过次数设定为 3 次
1	0	0	将 TGIA_3 的中断跳过次数设定为 4 次
1	0	1	将 TGIA_3 的中断跳过次数设定为 5 次
1	1	0	将 TGIA_3 的中断跳过次数设定为 6 次
1	1	1	将 TGIA_3 的中断跳过次数设定为 7 次

表 11.39 通过 4VCOR2 ~ 4VCOR0 位设定中断跳过次数

bit2	bit1	bit0	说 明
4VCOR2	4VCOR1	4VCOR0	
0	0	0	不执行 TCIV_4 的中断跳过
0	0	1	将 TCIV_4 的中断跳过次数设定为 1 次
0	1	0	将 TCIV_4 的中断跳过次数设定为 2 次
0	1	1	将 TCIV_4 的中断跳过次数设定为 3 次
1	0	0	将 TCIV_4 的中断跳过次数设定为 4 次
1	0	1	将 TCIV_4 的中断跳过次数设定为 5 次
1	1	0	将 TCIV_4 的中断跳过次数设定为 6 次
1	1	1	将 TCIV_4 的中断跳过次数设定为 7 次

11.3.26 定时器中断跳过次数计数器 (TITCNT)

TITCNT 为 8 位可读取 / 写入的计数器，MTU2 有 1 个 TITCNT。TCNT_3 和 TCNT_4 计数停止后，TITCNT 仍保持其值。

位:	7	6	5	4	3	2	1	0
	-	3ACNT[2:0]			-	4VCNT[2:0]		
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名称	初始值	R/W	说 明
7	—	0	R	保留位 读取值总是为 0。
6 ~ 4	3ACNT[2:0]	000	R	TGIA_3 中断计数器 如果 TITCR 的 T3AEN 位设定为 1，则产生 TGIA_3 中断源时本计数器递增 1。 [清除条件] • TITCR 的 3ACOR2 ~ 3ACOR0 和 TITCNT 的 3ACNT2 ~ 3ACNT0 匹配时 • TITCR 的 T3AEN 位为 0 时 • TITCR 的 3ACOR2 ~ 3ACOR0 为 0 时
3	—	0	R	保留位 读取值总是为 0。
2 ~ 0	4VCNT[2:0]	000	R	TCIV_4 中断计数器 如果 TITCR 的 T4VEN 位设定为 1，则产生 TCIV_4 中断源时本计数器递增 1。 [清除条件] • TITCR 的 4VCOR2 ~ 4VCOR0 和 TITCNT 的 4VCNT2 ~ 4VCNT0 匹配时 • TITCR 的 T4VEN 位为 0 时 • TITCR 的 4VCOR2 ~ 4VCOR0 为 0 时

【注】 清除 TITCNT 的值时，必须将 TITCR 的 T3AEN 位和 T4VEN 位清 0。

11.3.27 定时器缓冲传送设定寄存器 (TBTER)

TBTER 为 8 位可读取 / 写入的寄存器，可设定是否抑制从互补 PWM 模式时使用的缓冲寄存器 * 向暂寄存器传送，或设定是否与中断跳过功能联动。MTU2 有 1 个 TBTER。

位:	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	BTE[1:0]	
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W

位	位名称	初始值	R/W	说 明
7 ~ 2	—	均为 0	R	保留位 读取值、写入值总是为 0。
1、0	BTE[1:0]	00	R/W	设定是否抑制从互补 PWM 模式使用的缓冲寄存器 * 向暂寄存器传送，或设定是否与中断跳过功能联动。详情参照表 11.40。

【注】 * 对象缓冲寄存器：

TGRC_3、TGRD_3、TGRC_4、TGRD_4、TCBR

表 11.40 BTE1、BTE0 位的设定

bit1	bit0	说 明
BIE1	BIE0	
0	0	允许从缓冲寄存器向暂寄存器传送 *1 不与中断跳过功能联动
0	1	抑制从缓冲寄存器向暂寄存器传送
1	0	从缓冲寄存器向暂寄存器的传送与中断跳过功能联动 *2
1	1	禁止设定

【注】 *1 按照 TMDR 的 MD3 ~ MD0 位的设定传送。详情参阅“11.4.8 互补 PWM 模式”。

*2 禁止中断跳过时（定时器中断跳过设定寄存器（TITCR）的 T3AEN、T4VEN 位设定为 0，或 TITCR 的跳过次数设定位（3ACOR、4VCOR）设定为 0 时），必须设定为缓冲传送不与中断跳过联动（定时器缓冲传送寄存器（TBTER）的 BTE1 设定为 0）。

禁止中断跳过时，如果设定为缓冲传送与中断跳过联动，则不执行缓冲传送。

11.3.28 定时器空载时间允许寄存器（TDER）

TDER 为 8 位可读取 / 写入的寄存器，通道 3 有 1 个 TDER，可控制生成互补 PWM 模式的空载时间。
MTU2 有 1 个 TDER，必须在 TCNT 停止运行状态设定 TDER。

位:	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	TDER
初始值:	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R/(W)

位	位名称	初始值	R/W	说 明
7 ~ 1	—	均为 0	R	保留位 读取值、写入值总是为 0。
0	TDER	1	R/(W)	空载时间允许寄存器 设定是否生成空载时间。 0: 不生成空载时间 1: 生成空载时间 * [清除条件] • 在 TDER=1 的状态读取 TDER 后，对 TDER 写入 0 时

【注】 * 必须设定为 $TDDR \geq 1$ 。

11.3.29 定时器波形控制寄存器 (TWCR)

TWCR 为 8 位可读取 / 写入的寄存器，控制在互补 PWM 模式产生 TCNT_3、TCNT_4 的同步计数器清除时的输出波形，并设定是否执行 TGRA_3 比较匹配引起的计数器清除。必须在 TCNT 停止运行状态设定 TWCR 的 CCE 位和 WRE 位。

位:	7	6	5	4	3	2	1	0
	CCE	-	-	-	-	-	-	WRE
初始值:	0*	0	0	0	0	0	0	0
R/W:	R/(W)	R	R	R	R	R	R	R/(W)

【注】* 除互补PWM1模式之外，不得设定为1。

位	位名称	初始值	R/W	说 明
7	CCE	0*	R/(W)	比较匹配清除允许 在互补 PWM 模式，设定是否执行 TGRA_3 比较匹配引起的计数器清除。 0: 不执行 TGRA_3 比较匹配引起的计数器清除 1: 执行 TGRA_3 比较匹配引起的计数器清除 [置位条件] • 在 CCE=0 的状态读取 CCE 后，对 CCE 写入 1 时
6 ~ 1	—	均为 0	R	保留位 读取值、写入值总是为 0。
0	WRE	0	R/(W)	波形保持允许 选择在互补 PWM 模式产生同步计数器清除时的输出波形。 仅在互补 PWM 模式的波谷 Tb 区间产生同步清除时，可由此功能保持波形。在此之外产生同步清除时，与 WRE 位的设定无关，输出由 TOCR 寄存器设定的初始值。在 TCNT_3、TCNT_4 开始后的波谷 Tb 区间产生同步清除时，也输出由 TOCR 寄存器设定的初始值。 关于互补 PWM 模式的波谷 Tb 区间，参照图 11.40。 0: 输出由 TOCR 寄存器设定的初始值 1: 保持同步清除前的波形 [置位条件] • 在 WRE=0 的状态读取 WRE 后，对 WRE 写入 1 时

【注】* 除互补 PWM 模式 1 之外，均不可设定为 1。

11.3.30 与总线主控器的接口

定时器计数器 (TCNT)、通用寄存器 (TGR)、定时器副计数器 (TCNTS)、定时器周期缓冲寄存器 (TCBR)、定时器空载时间数据寄存器 (TDDR)、定时器周期数据寄存器 (TCDR)、定时器 A/D 转换开始请求控制寄存器 (TADCR)、定时器 A/D 转换开始请求周期设定寄存器 (TADCOR) 及定时器 A/D 转换开始请求周期设定缓冲寄存器 (TADCOBR) 均为 16 位寄存器。与总线主控器之间的数据总线宽度为 16 位, 因此能以 16 位为单位读取 / 写入, 但不能以 8 位为单位读取 / 写入, 并总是以 16 位为单位存取。

上述情况之外的寄存器均为 8 位寄存器, 与 CPU 之间的数据总线宽度为 16 位, 能以 16 位为单位读取 / 写入, 也能以 8 位为单位读取 / 写入。

11.4 运行说明

11.4.1 基本运行

各通道有 TCNT 和 TGR。TCNT 可执行递增计数运行、自由运行、周期计数器运行或外部事件计数运行。TGR 可分别用作输入捕捉寄存器或输出比较寄存器。

必须由引脚功能控制器 (PFC) 设定 MTU2 的外部引脚功能。

(1) 计数器的运行

TSTR 的 CST0 ~ CST4 位置 1 时, 对应通道的 TCNT 开始计数运行。可执行自由运行计数器运行、周期计数器运行等。

(a) 计数运行的设定步骤例

计数运行的设定步骤例如图 11.4 所示。

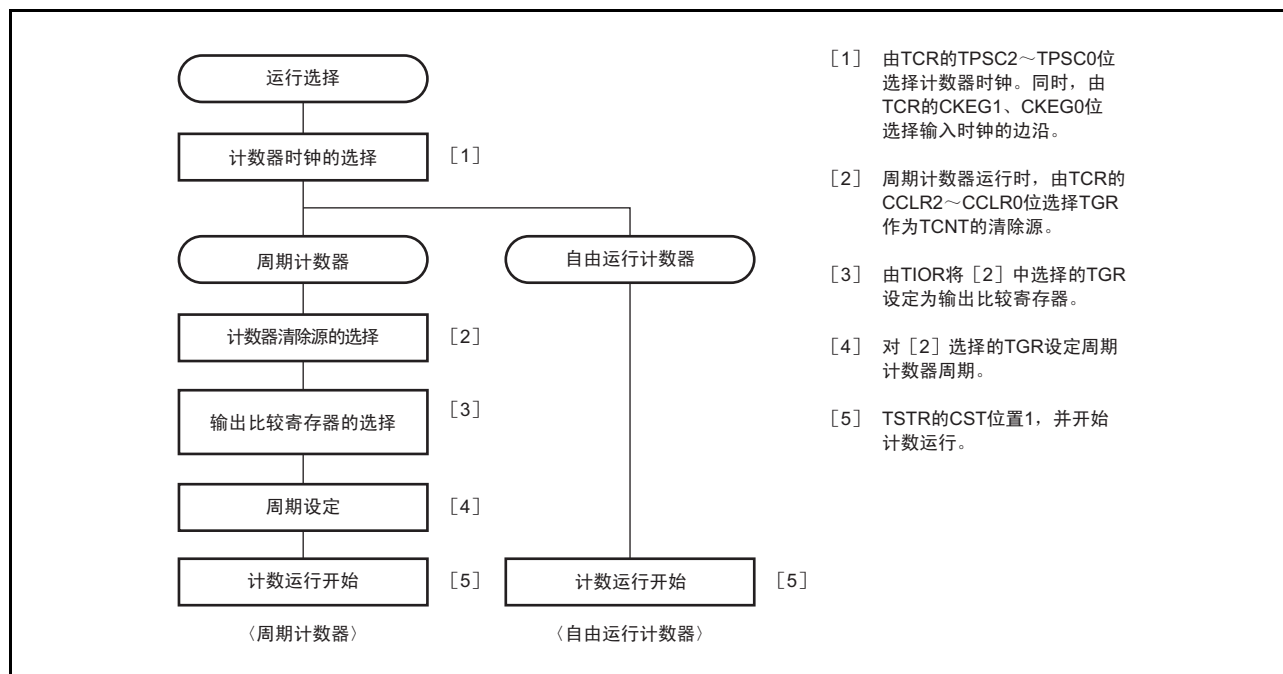


图 11.4 计数器运行的设定步骤例

(b) 自由运行计数和周期计数运行

MTU2 的 TCNT 在刚复位后均为自由运行计数器的设定，如果 TSTR 的对应位置 1，则作为自由运行计数器开始递增计数。TCNT 上溢 (H'FFFF → H'0000) 时，TSR 的 TCFV 位置 1。此时，如果对应的 TIER 的 TCIEV 位为 1，则 MTU2 请求中断。TCNT 上溢后，从 H'0000 开始继续递增计数。

自由运行计数器的运行如图 11.5 所示。

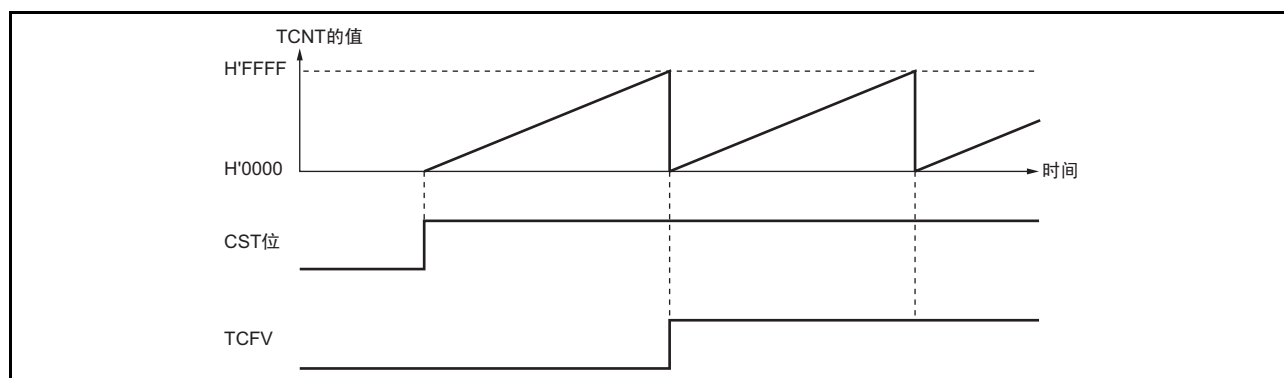


图 11.5 自由运行计数器的运行

选择比较匹配为 TCNT 的清除源时，对应通道的 TCNT 执行周期计数器运行。用于周期设定的 TGR 设定为输出比较寄存器，由 TCR 的 CCLR2 ~ CCLR0 位选择比较匹配产生的计数器清除。设定后，如果 TSTR 的对应位置 1，则作为周期计数器开始递增计数。计数值和 TGR 的值匹配时，TSR 的 TGF 位置 1，TCNT 清除为 H'0000。

此时，如果对应 TIER 的 TGIE 位为 1，则 MTU2 请求中断。TCNT 在比较匹配后，从 H'0000 开始继续递增计数。

周期计数器的运行如图 11.6 所示。

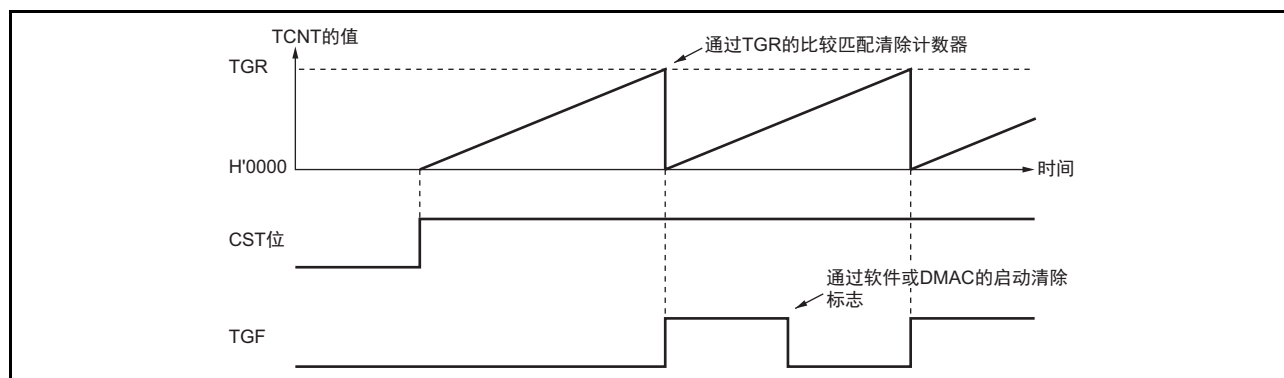


图 11.6 周期计数器的运行

(2) 比较匹配产生的波形输出功能

MTU2 可由比较匹配从对应的输出引脚执行输出 0/ 输出 1/ 交替输出。

(a) 比较匹配产生的波形输出运行的设定步骤例

比较匹配产生的波形输出运行的设定步骤例如图 11.7 所示。

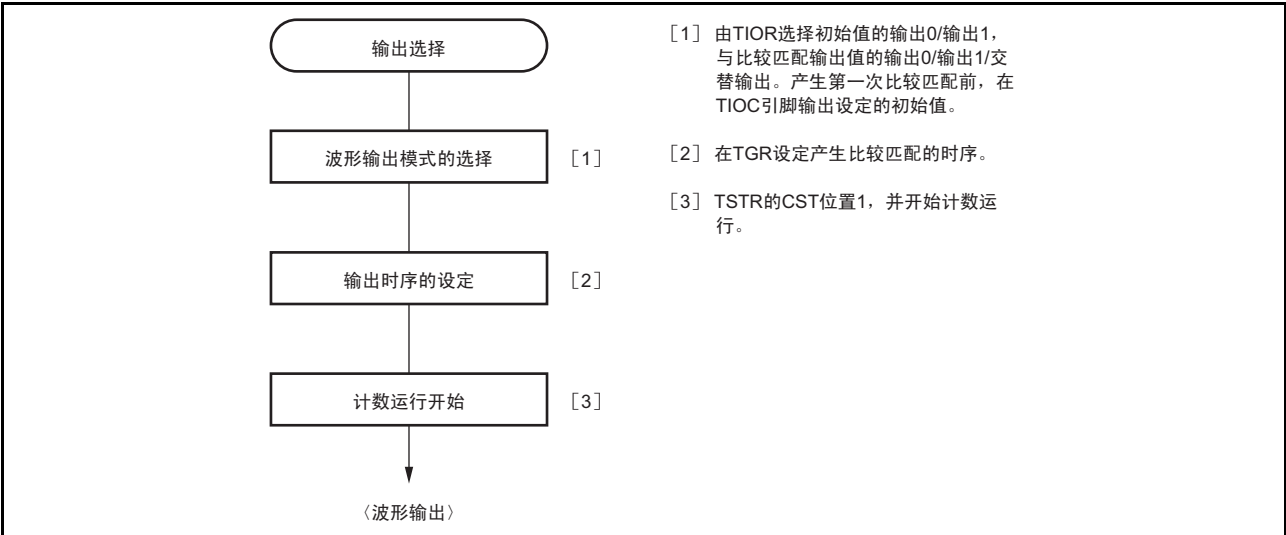


图 11.7 比较匹配产生的波形输出的设定步骤例

(b) 波形输出运行例

输出 0/ 输出 1 的例子如图 11.8 所示。

在此例，设定 TCNT 为自由运行计数运行，由比较匹配 A 输出 1，比较匹配 B 输出 0。设定的电平和引脚电平匹配时，引脚电平不变。

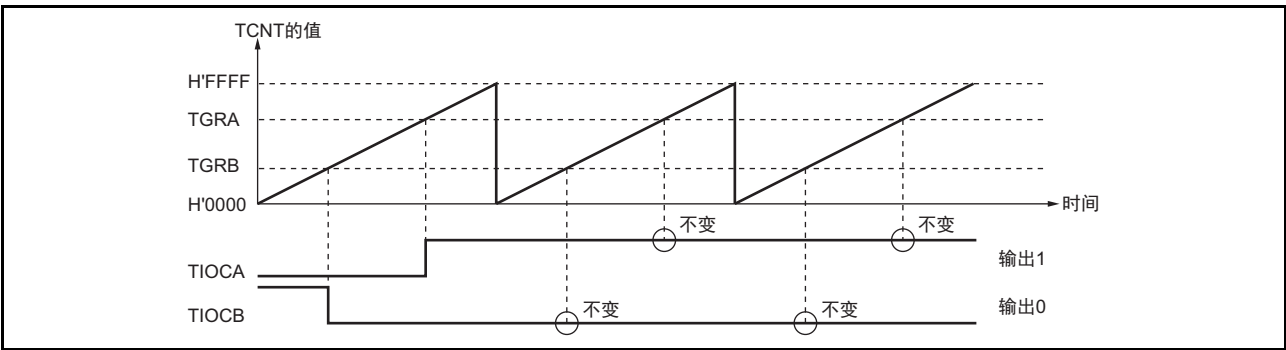


图 11.8 输出 0/ 输出 1 的运行例

交替输出的例子如图 11.9 所示。
在此例，设定 TCNT 为周期计数运行（由比较匹配 B 清除计数器），比较匹配 A、B 均为交替输出。

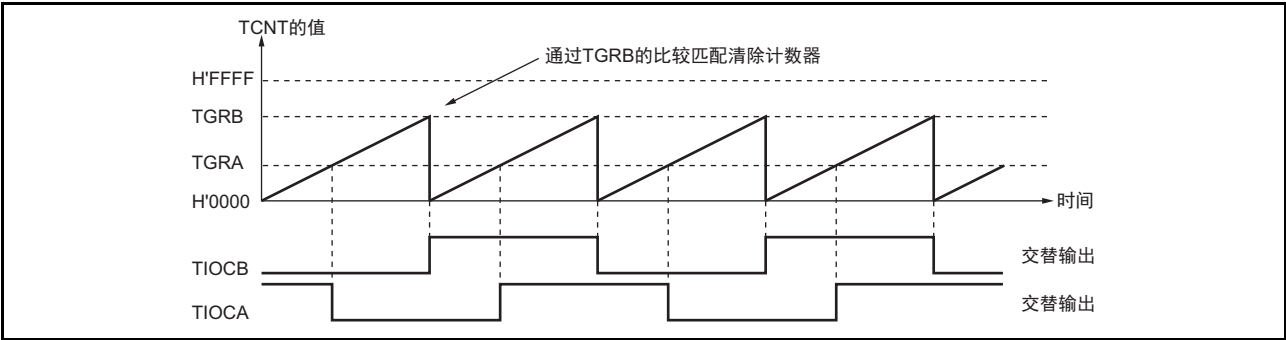


图 11.9 交替输出的运行例

(3) 输入捕捉功能

本功能可检测 TIOC 引脚的输入沿，并将 TCNT 的值传送至 TGR。
检测沿可选择上升沿 / 下降沿 / 双边沿。通道 0、1 可将其他通道的计数器输入时钟，或比较匹配信号作为输入捕捉源。
【注】 在通道 0、1 将其他通道的计数器输入时钟作为输入捕捉输入时，作为输入捕捉输入的计数器输入时钟不得选择 P ϕ /1。如果选择 P ϕ /1，则不产生输入捕捉。

(a) 输入捕捉运行的设定步骤例

输入捕捉运行的设定步骤例如图 11.10 所示。

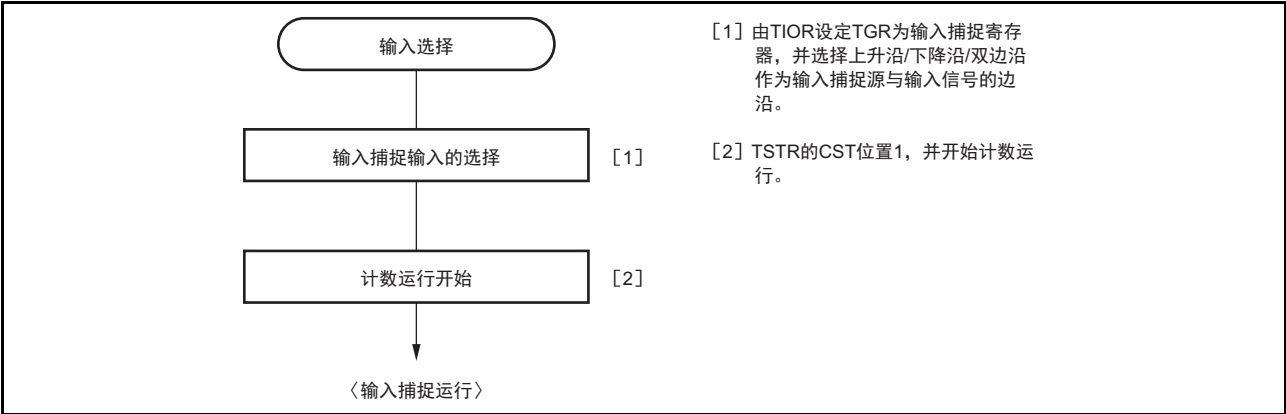


图 11.10 输入捕捉运行的设定例

(b) 输入捕捉运行例

输入捕捉运行例如图 11.11 所示。

本例是选择上升沿 / 下降沿的双边沿为 TIOCA 引脚输入捕捉的输入沿，选择下降沿为 TIOCB 引脚输入捕捉的输入沿，并将 TCNT 设定为由 TGRB 的输入捕捉清除计数器的例子。

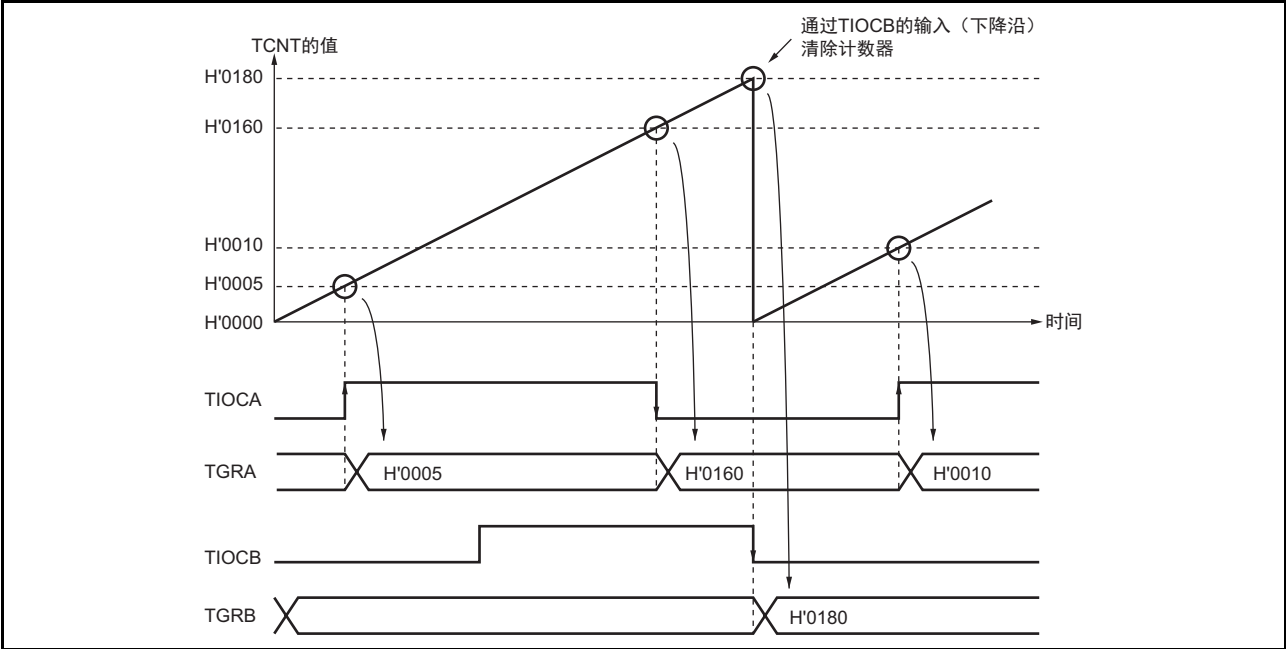


图 11.11 输入捕捉运行例

11.4.2 同步运行

同步运行可同时改写多个 TCNT 的值（同步预置），也可通过设定 TCR 同时清除多个 TCNT（同步清除）。

同步运行可增加 1 个时基运行的 TGR 个数。

通道 0 ~ 4 均可设定为同步运行。

(1) 同步运行的设定步骤例

同步运行的设定步骤例如图 11.12 所示。

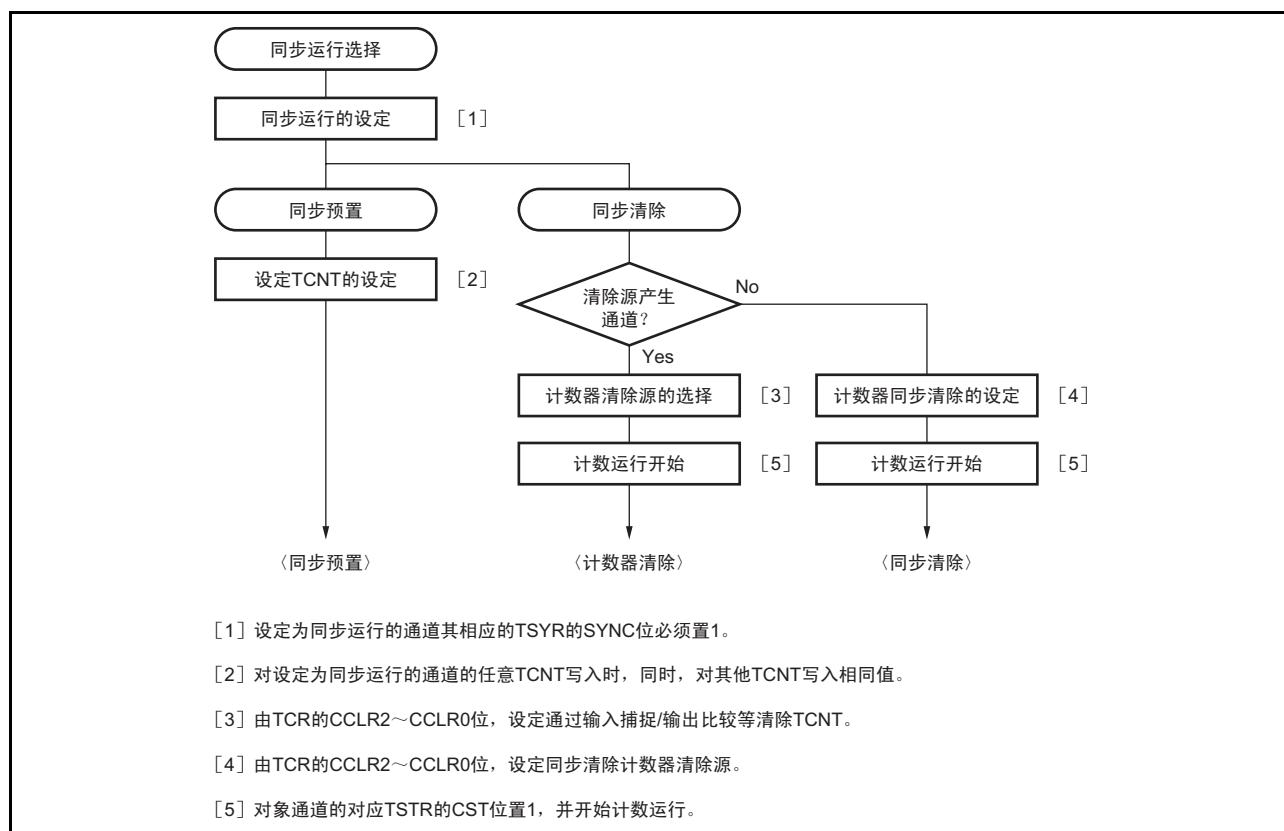


图 11.12 同步运行的设定步骤例

(2) 同步运行的例子

同步运行的例子如图 11.13 所示。

本例是设定通道 0 ~ 2 为同步运行且为 PWM 模式 1，TGRB_0 的比较匹配为通道 0 的计数器清除源，同步清除通道 1、2 的计数器清除源的例子。

从 TIOC0A、TIOC1A 和 TIOC2A 引脚输出 3 相 PWM 波形。此时，通道 0 ~ 2 的 TCNT 执行同步预置，TGRB_0 比较匹配产生同步清除，对 TGRB_0 设定的数据为 PWM 周期。

有关 PWM 模式，详情参阅“11.4.5 PWM 模式”。

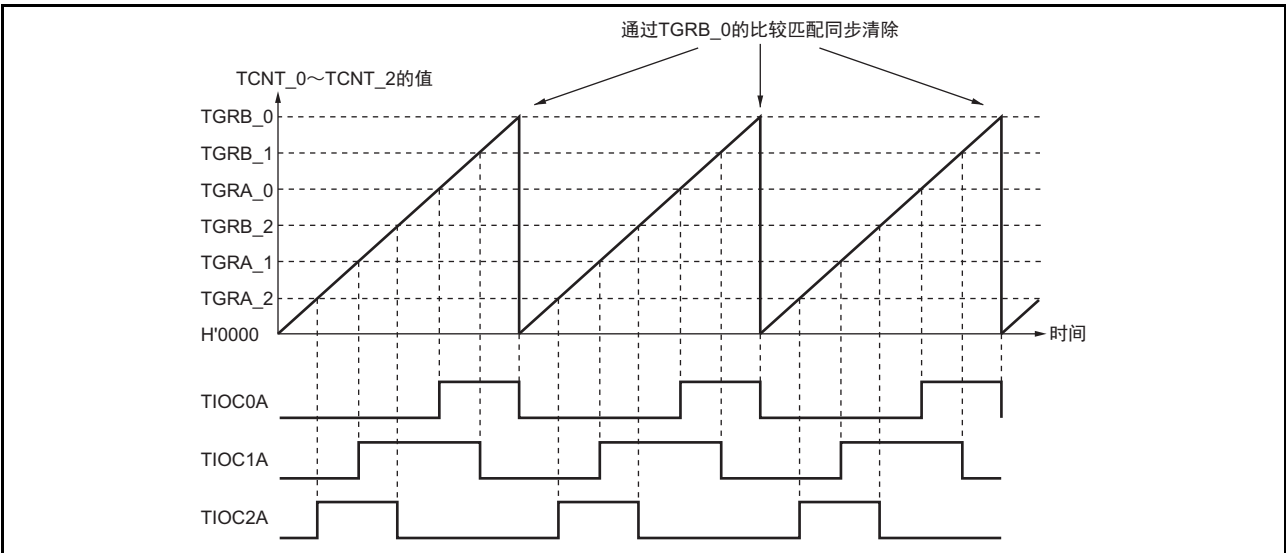


图 11.13 同步运行的例子

11.4.3 缓冲运行

通道 0、3、4 有缓冲运行功能。TGRC 和 TGRD 可用作缓冲寄存器，通道 0 的 TGRF 也可用作缓冲寄存器。

TGR 设定为输入捕捉寄存器和设定为比较匹配寄存器时的缓冲运行内容不同。

【注】 TGRE_0 不可设定为输入捕捉寄存器，仅作为比较匹配寄存器运行。

缓冲运行时的寄存器组合如表 11.41 所示。

表 11.41 寄存器组合

通道	定时器通用寄存器	缓冲寄存器
0	TGRA_0	TGRC_0
	TGRB_0	TGRD_0
	TGRE_0	TGRF_0
3	TGRA_3	TGRC_3
	TGRB_3	TGRD_3
4	TGRA_4	TGRC_4
	TGRB_4	TGRD_4

- TGR为输出比较寄存器时
产生比较匹配时，对应通道的缓冲寄存器值传送至定时器通用寄存器。
此运行如图 11.14 所示。

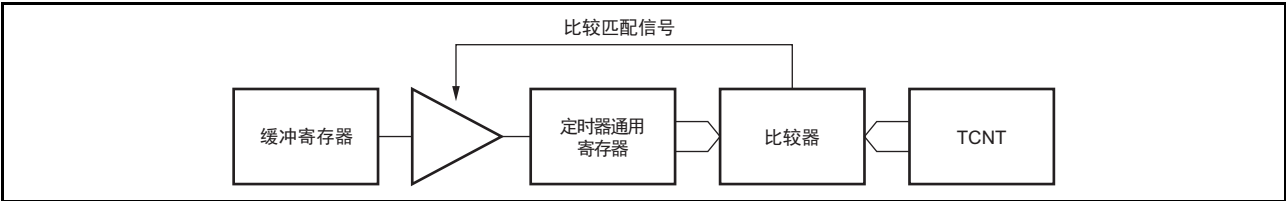


图 11.14 比较匹配缓冲运行

- TGR为输入捕捉寄存器时
产生输入捕捉时，TCNT 的值传送至 TGR 的同时，之前保存的 TGR 的值也传送至缓冲寄存器。
此运行如图 11.15 所示。

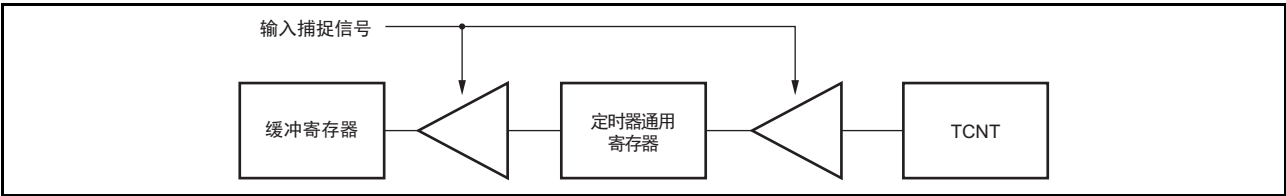


图 11.15 输入捕捉缓冲运行

(1) 缓冲运行的设定步骤例

缓冲运行的设定步骤例如图 11.16 所示。

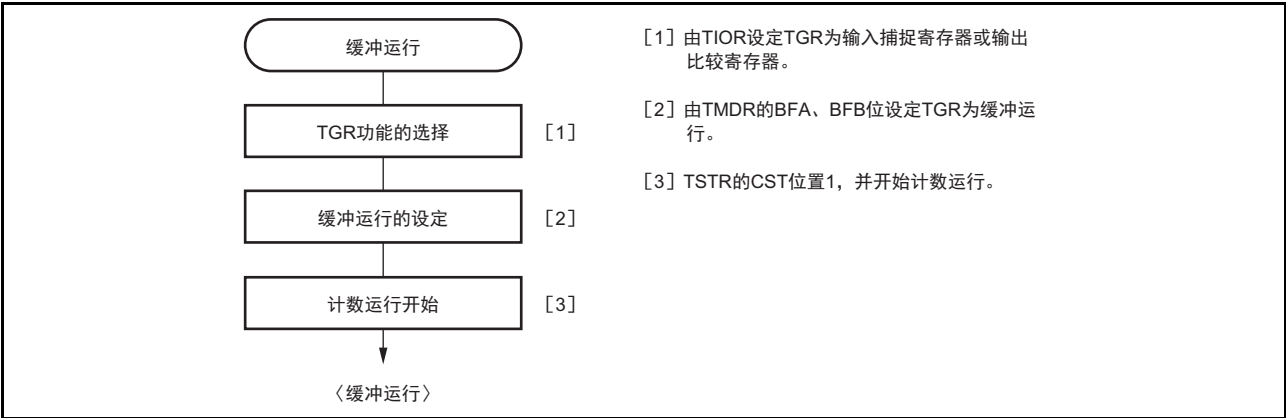


图 11.16 缓冲运行的设定步骤例

(2) 缓冲运行例

(a) TGR 为输出比较寄存器时

通道 0 设定为 PWM 模式 1、TGRA 和 TGRC 设定为缓冲运行时的运行例如图 11.17 所示。设定由比较匹配 B 清除 TCNT，由比较匹配 A 输出 1、由比较匹配 B 输出 0。此例中，将 TBTM 的 TTSA 位设定为 0。

由于设定了缓冲运行，因此产生比较匹配 A 后输出发生变化的同时，缓冲寄存器 TGRC 的值传送至定时器通用寄存器 TGRA。每产生一次比较匹配 A 时，重复此运行。

有关 PWM 模式，详情参阅“11.4.5 PWM 模式”。

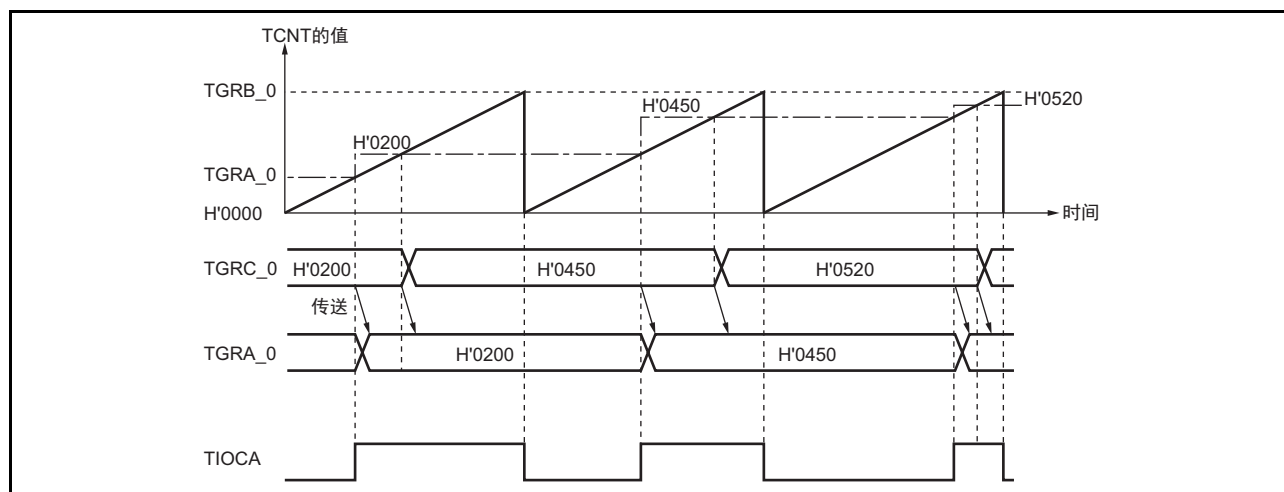


图 11.17 缓冲运行例 (1)

(b) TGR 为输入捕捉寄存器时

TGRA 设定为输入捕捉寄存器，TGRA 和 TGRC 设定为缓冲运行时的运行例如图 11.18 所示。

TCNT 由 TGRA 的输入捕捉清除计数器，TIOCA 引脚的输入捕捉输入沿可选择上升 / 下降双边沿。

由于设定了缓冲运行，所以由输入捕捉 A 将 TCNT 的值保存至 TGRA，与此同时将之前保存在 TGRA 的值传送至 TGRC。

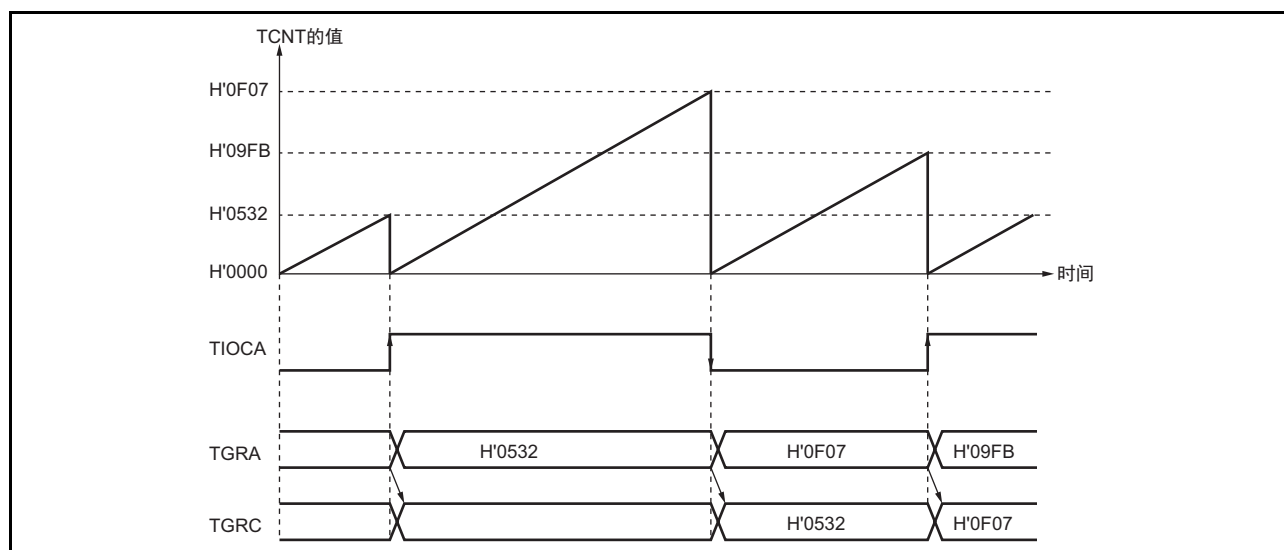


图 11.18 缓冲运行例 (2)

(3) 选择从缓冲运行时的缓冲寄存器向定时器通用寄存器传送的时序

通过设定缓冲运行传送模式寄存器 (TBTM_0、TBTM_3、TBTM_4)，可选择通道 0 为 PWM 模式 1、2，通道 3、4 为 PWM 模式 1 时，从缓冲寄存器向定时器通用寄存器传送的时序。可选的缓冲传送时序为产生比较匹配（初始值）或清除 TCNT 时。以下任意条件成立时，可清除 TCNT。

- TCNT 上溢时 (H'FFFF → H'0000)
- 计数器运行过程中，对 TCNT 写入 H'0000 时
- 由 TCR 的 CCLR2 ~ CCLR0 位设定的清除源，将 TCNT 清除为 H'0000 时

【注】 必须在 TCNT 停止状态设定 TBTM。

通道 0 设定为 PWM 模式 1，TGRC_0 和 TGRA_0 设定为缓冲运行时的运行例如图 11.19 所示。设定由比较匹配 B 清除 TCNT_0，由比较匹配 A 输出 1、比较匹配 B 输出 0，并且 TBTM_0 的 TTSA 位设定为 1。

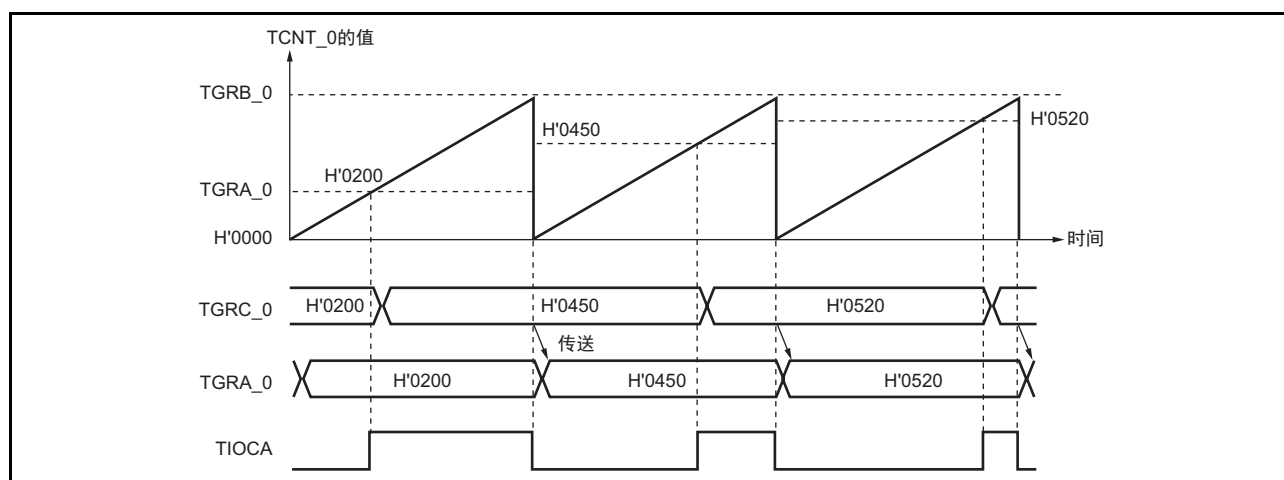


图 11.19 选择 TGRC_0 到 TGRA_0 的缓冲传送时序为清除 TCNT_0 时的运行例

11.4.4 级联运行

级联运行可连接 2 个通道的 16 位计数器作为 32 位计数器运行。

由 TCR 的 TPSC2 ~ TPSC0 位将通道 1 的计数器时钟设定为通过 TCNT_2 的上溢 / 下溢进行计数时，此功能运行。

仅在低 16 位的 TCNT 为相位计数模式时产生下溢。

级联组合如表 11.42 所示。

【注】 通道 1 设定为相位计数模式时，计数器时钟的设定无效，以相位计数模式独立运行。

表 11.42 级联组合

组合	高 16 位	低 16 位
通道 1 和通道 2	TCNT_1	TCNT_2

级联运行时，如果同时输入捕捉 TCNT_1 和 TCNT_2，则通过设定输入捕捉控制寄存器 (TICCR)，可将输入引脚追加到输入捕捉条件。有关级联时的输入捕捉，详情参阅 “11.7.22 级联时 TCNT_1 和 TCNT_2 同时输入捕捉”。

TICCR 设定值和输入捕捉输入引脚的对应关系如表 11.43 所示。

表 11.43 TICCR 设定值和输入捕捉输入引脚的对应关系

输入捕捉对象	TICCR 设定值	输入捕捉输入引脚
从 TCNT_1 向 TGRA_1 的输入捕捉	I2AE 位 =0 (初始值)	TIOC1A
	I2AE 位 =1	TIOC1A、TIOC2A
从 TCNT_1 向 TGRB_1 的输入捕捉	I2BE 位 =0 (初始值)	TIOC1B
	I2BE 位 =1	TIOC1B、TIOC2B
从 TCNT_2 向 TGRA_2 的输入捕捉	I1AE 位 =0 (初始值)	TIOC2A
	I1AE 位 =1	TIOC2A、TIOC1A
从 TCNT_2 向 TGRB_2 的输入捕捉	I1BE 位 =0 (初始值)	TIOC2B
	I1BE 位 =1	TIOC2B、TIOC1B

(1) 级联运行的设定步骤例

级联运行的设定步骤例如图 11.20 所示。

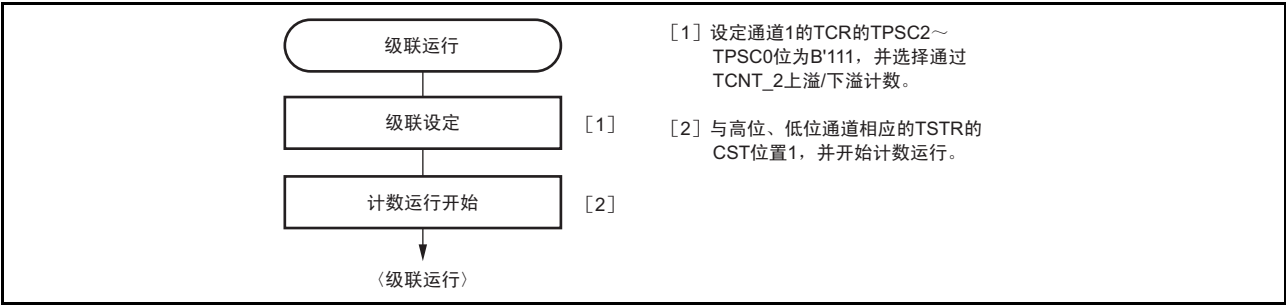


图 11.20 级联运行的设定步骤

(2) 级联运行例 (a)

TCNT_1 通过 TCNT_2 的上溢 / 下溢计数，通道 2 设定为相位计数模式时的运行如图 11.21 所示。
TCNT_1 通过 TCNT_2 的上溢执行递增计数，通过 TCNT_2 的下溢执行递减计数。

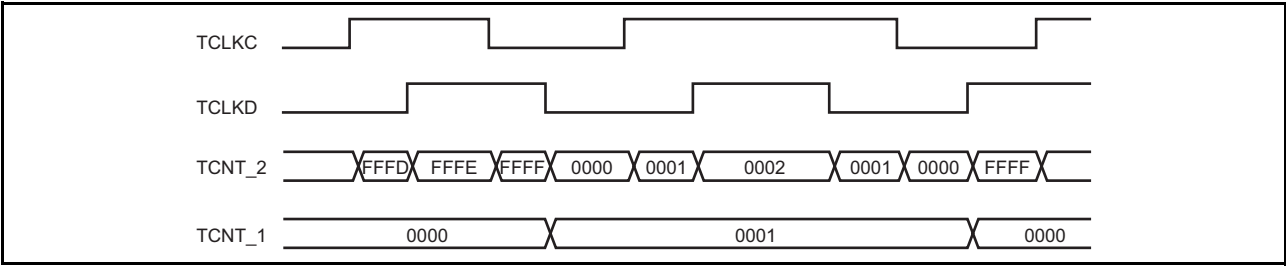


图 11.21 级联运行例 (a)

(3) 级联运行例 (b)

级联 TCNT_1 和 TCNT_2, TICCRR 的 I2AE 位置 1 后, 将 TIOC2A 引脚追加到 TGRA_1 的输入捕捉条件时的运行如图 11.22 所示。在此例, TIOR_1 的 IOA0 ~ IOA3 设定为在 (TIOC1A 的) 上升沿执行输入捕捉, TIOR_2 的 IOA0 ~ IOA3 设定为在 (TIOC2A 的) 上升沿执行输入捕捉。

此时, TIOC1A 和 TIOC2A 的上升沿设定为 TGRA_1 的输入捕捉条件, TIOC2A 的上升沿为 TGRA_2 的输入捕捉条件。

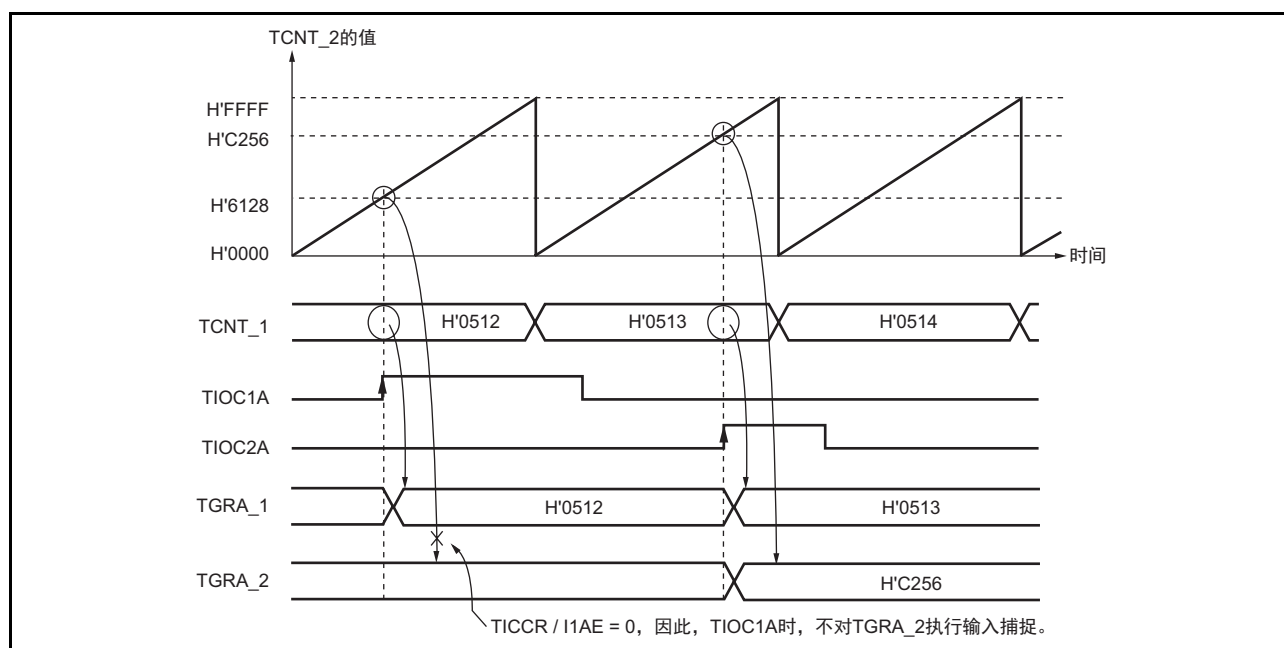


图 11.22 级联运行例 (b)

(4) 级联运行例 (c)

级联 TCNT_1 和 TCNT_2, TICCR 的 I2AE 和 I1AE 位置 1 后, 将 TIOC2A 引脚追加到 TGRA_1 的输入捕捉条件、TIOC1A 引脚追加到 TGRA_2 的输入捕捉条件时的运行如图 11.23 所示。此例中, TIOR_1 和 TIOR_2 的 IOA0 ~ IOA3 位设定为可在双边沿执行输入捕捉。此时, TIOC1A 和 TIOC2A 输入的 OR 为 TGRA_1 和 TGRA_2 的输入捕捉条件。

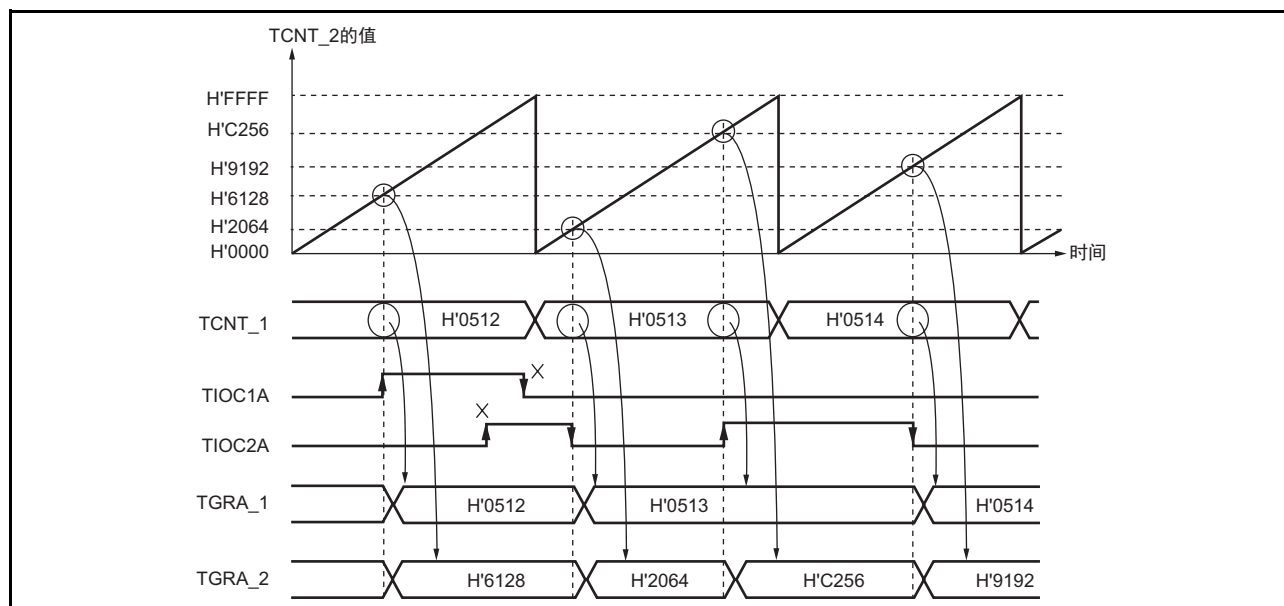


图 11.23 级联运行例 (c)

(5) 级联运行例 (d)

级联 TCNT_1 和 TCNT_2, TICCIR 的 I2AE 位置 1, 将 TIOC2A 引脚追加到 TGRA_1 的输入捕捉条件时的运行如图 11.24 所示。此例中, TIOR_1 的 IOA0 ~ IOA3 设定为通过产生 TGRA_0 的比较匹配 / 输入捕捉执行输入捕捉, TIOR_2 的 IOA0 ~ IOA3 设定为在 (TIOC2A 的) 上升沿执行输入捕捉。

TIOR_1 设定为通过产生 TGRA_0 的比较匹配 / 输入捕捉输入捕捉, 因此即使将 TICCIR 的 I2AE 位置 1, TIOC2A 的边沿也不可作为 TGRA_1 的输入捕捉条件。

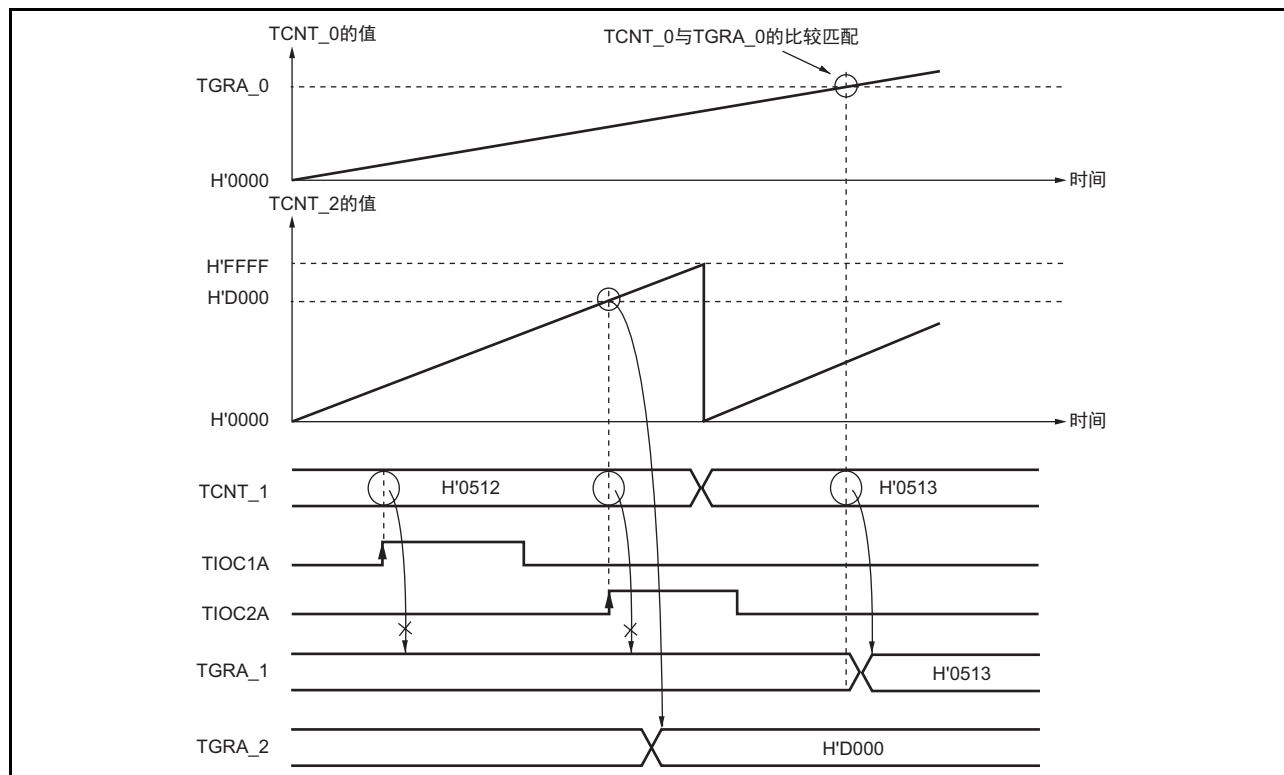


图 11.24 级联运行例 (d)

11.4.5 PWM 模式

PWM 模式是从输出引脚输出各种 PWM 波形的模式。各 TGR 比较匹配的输出电平可选择输出 0/ 输出 1/ 交替输出。

通过设定各 TGR，可输出占空比为 0 ~ 100% 的 PWM 波形。

将 TGR 的比较匹配设定为计时器清除源，可对该寄存器设定周期。所有通道均可独立设定为 PWM 模式，也可同步运行。

PWM 模式有以下 2 种：

- PWM 模式 1

TGRA 和 TGRB、TGRC 和 TGRD 配对使用，从 TIOCA、TIOCC 引脚生成 PWM 输出。通过比较匹配 A、C，从 TIOCA、TIOCC 引脚执行 TIOR 的 IOA3 ~ IOA0、IOC3 ~ IOC0 位指定的输出，同时通过比较匹配 B、D 执行 TIOR 的 IOB3 ~ IOB0 和 IOD3 ~ IOD0 位指定的输出。初始输出值为在 TGRA、TGRC 设定的值。配对使用的 TGR 的设定值相同时，即使产生比较匹配，输出值也不变。

PWM 模式 1 最多可输出 8 相 PWM。

- PWM 模式 2

将 1 个 TGR 用作周期寄存器，其他 TGR 用作占空比寄存器，生成 PWM 输出。通过比较匹配，执行 TIOR 指定的输出。并通过同步寄存器的比较匹配清除计数器后，各引脚输出 TIOR 设定的初始值。周期寄存器和占空比寄存器的设定值相同时，即使产生比较匹配，输出值也不变。

在 PWM 模式 2，通过与同步运行并用，最多可输出 8 相 PWM。

PWM 输出引脚和寄存器的对应关系如表 11.44 所示。

表 11.44 各 PWM 输出的寄存器和输出引脚

通道	寄存器	输出引脚	
		PWM 模式 1	PWM 模式 2
0	TGRA_0	TIOC0A	TIOC0A
	TGRB_0		TIOC0B
	TGRC_0	TIOC0C	TIOC0C
	TGRD_0		TIOC0D
1	TGRA_1	TIOC1A	TIOC1A
	TGRB_1		TIOC1B
2	TGRA_2	TIOC2A	TIOC2A
	TGRB_2		TIOC2B
3	TGRA_3	TIOC3A	不可设定
	TGRB_3		
	TGRC_3	TIOC3C	
	TGRD_3		
4	TGRA_4	TIOC4A	
	TGRB_4		
	TGRC_4	TIOC4C	
	TGRD_4		

【注】 在 PWM 模式 2，已设定周期的 TGR 不可执行 PWM 输出。

(1) PWM 模式的设定步骤例

PWM 模式的设定步骤例如图 11.25 所示。

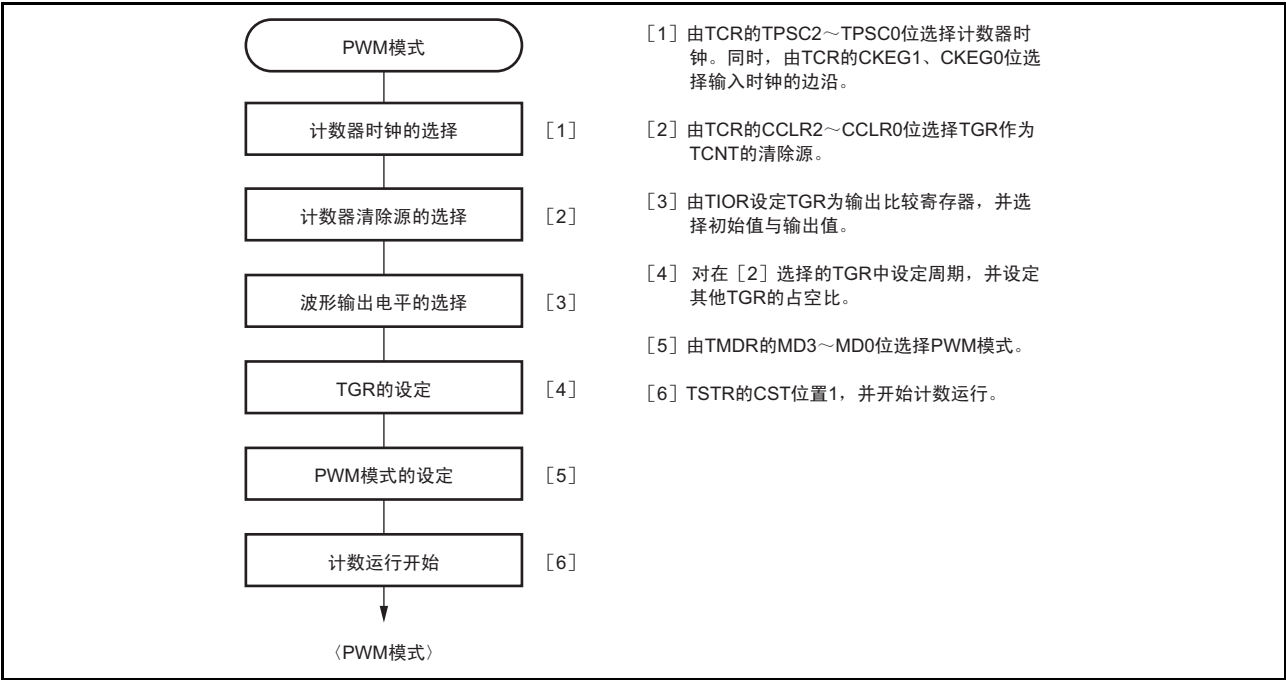


图 11.25 PWM 模式的设定步骤例

(2) PWM 模式运行例

PWM 模式 1 的运行例如图 11.26 所示。

此图将 TCNT 的清除源作为 TGRA 的比较匹配，并将 TGRA 的初始输出值和输出值设定为 0，TGRB 的输出值设定为 1。

此时，对 TGRA 设定的值为周期，对 TGRB 设定的值为占空比。

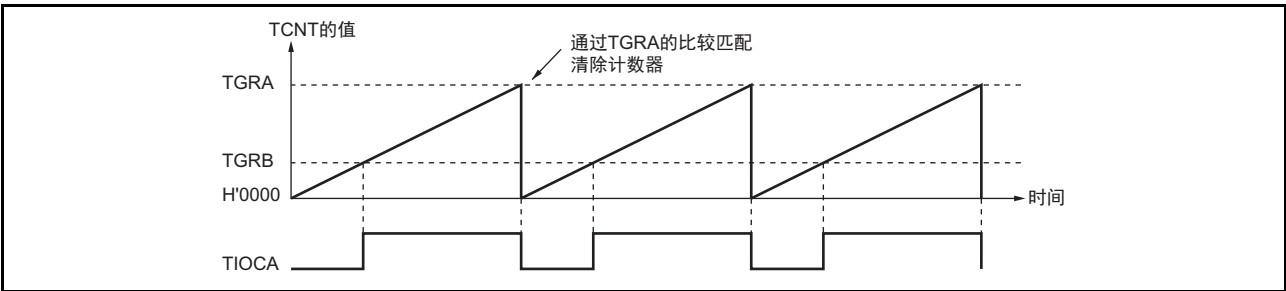


图 11.26 PWM 模式的运行例

PWM 模式 2 的运行例如图 11.27 所示。

此图是使通道 0 和 1 同步运行，将 TCNT 的清除源作为 TGRB_1 的比较匹配，并将其他 TGR (TGRA_0 ~ TGRD_0、TGRA_1) 的初始输出值设定为 0、输出值设定为 1，输出 5 相 PWM 波形的例子。
对 TGR1B 设定的值为周期，对其他 TGR 设定的值为占空比。

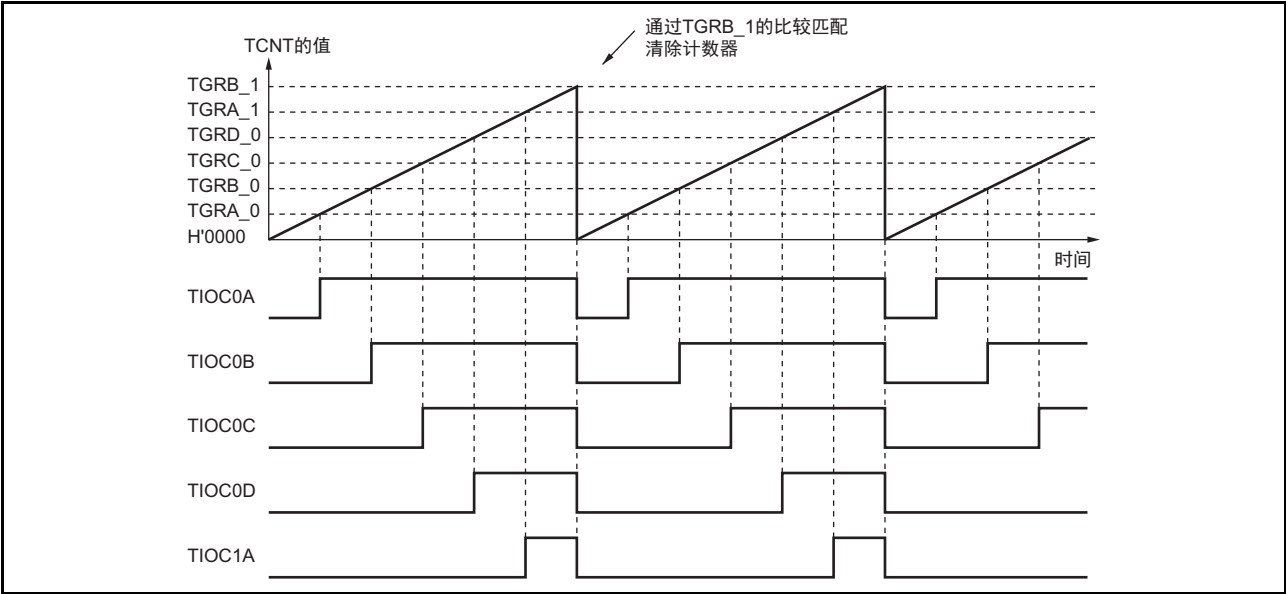


图 11.27 PWM 模式的运行例

在 PWM 模式，分别输出占空比 0% 和 100% 的 PWM 波形的例子如图 11.28 所示。

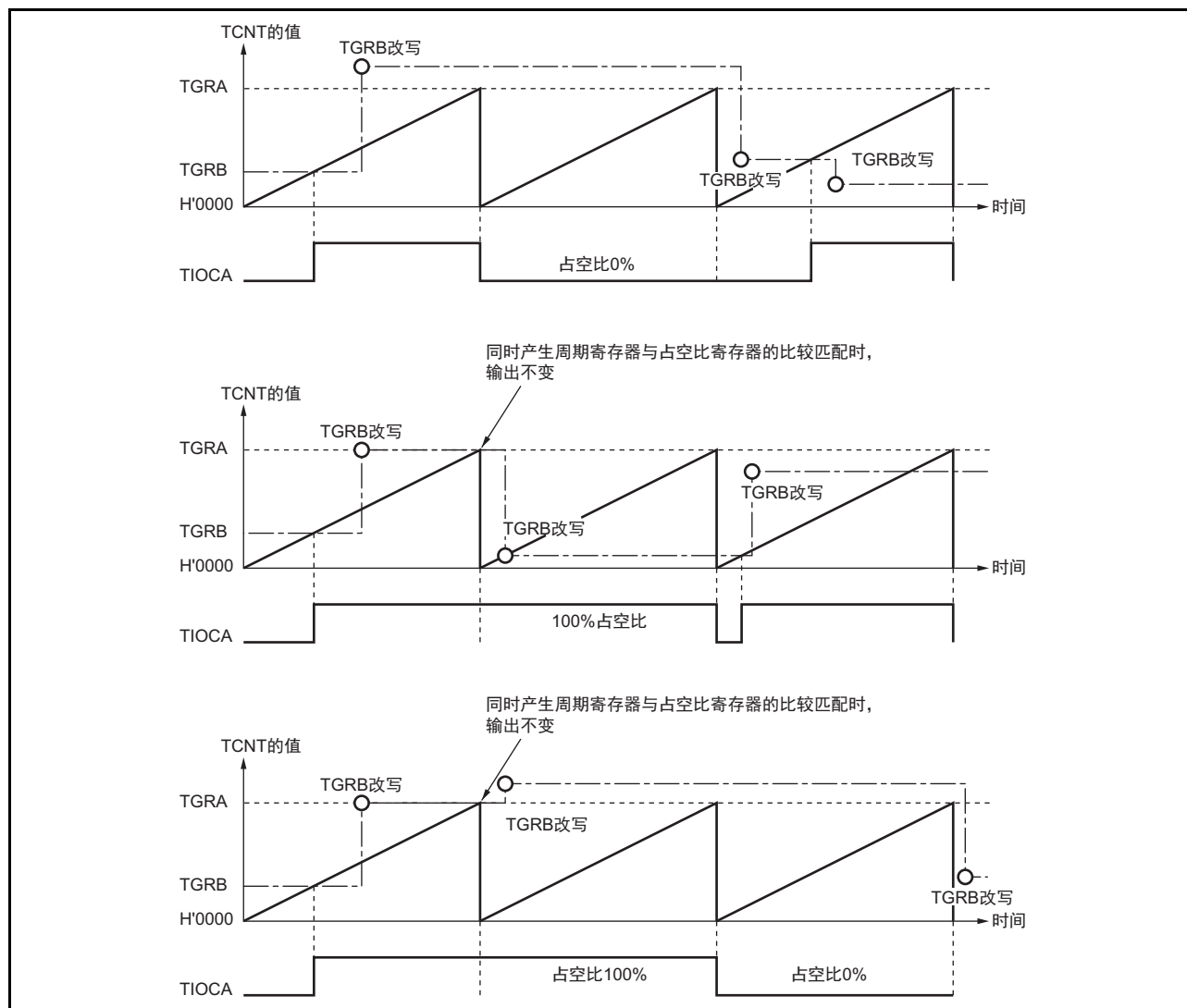


图 11.28 PWM 模式的运行例

11.4.6 相位计数模式

相位计数模式通过设定通道 1、2，可检测 2 个外部时钟输入的相位差，并递增 / 递减计数 TCNT。

设定为相位计数模式时，与 TCR 的 TPSC2 ~ TPSC0 位、CKEG1、CKEG0 位的设定无关，选择外部时钟为计数器输入时钟，TCNT 作为递增 / 递减计数器运行。但 TCR 的 CCLR1、CCLR0 位、TIOR、TIER、TGR 的功能有效，因此可使用输入捕捉 / 比较匹配功能和中断功能。

可用于 2 相编码器脉冲的输入。

TCNT 递增计数时，如果产生上溢，则 TSR 的 TCFV 标志置位；TCNT 递减计数时，如果产生下溢，则 TCFU 标志置位。

TSR 的 TCFD 位为计数方向标志。读取 TCFD 标志，可确认 TCNT 为递增计数或递减计数。

外部时钟引脚和通道的对应关系如表 11.45 所示。

表 11.45 相位计数模式时钟输入引脚

通道	外部时钟引脚	
	A 相	B 相
通道 1 设置为相位计数模式时	TCLKA	TCLKB
通道 2 设置为相位计数模式时	TCLKC	TCLKD

(1) 相位计数模式的设定步骤例

相位计数模式的设定步骤例如图 11.29 所示。

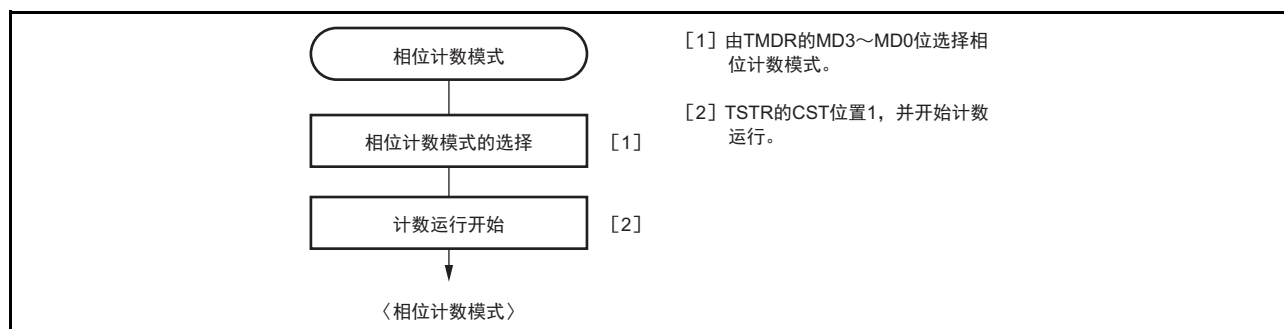


图 11.29 相位计数模式的设定步骤例

(2) 相位计数模式的运行例

在相位计数模式，根据 2 个外部时钟的相位差，TCNT 递增 / 递减计数。根据计数条件，有 4 种不同的模式。

(a) 相位计数模式 1

相位计数模式 1 的运行例如图 11.30 所示，TCNT 的递增 / 递减计数条件如表 11.46 所示。

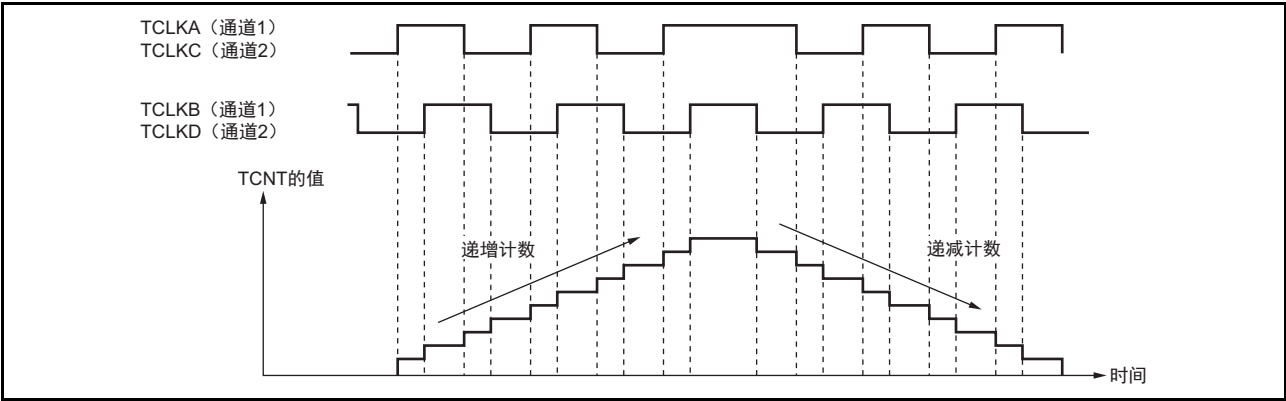


图 11.30 相位计数模式 1 的运行例

表 11.46 相位计数模式 1 的递增 / 递减计数条件

TCLKA (通道 1) TCLKC (通道 2)	TCLKB (通道 1) TCLKD (通道 2)	运行内容
高电平	上升沿	递增计数
低电平	下降沿	
上升沿	低电平	
下降沿	高电平	
高电平	下降沿	递减计数
低电平	上升沿	
上升沿	高电平	
下降沿	低电平	

【符号说明】

上升沿

下降沿

(b) 相位计数模式 2

相位计数模式 2 的运行例如图 11.31 所示，TCNT 的递增 / 递减计数条件如表 11.47 所示。

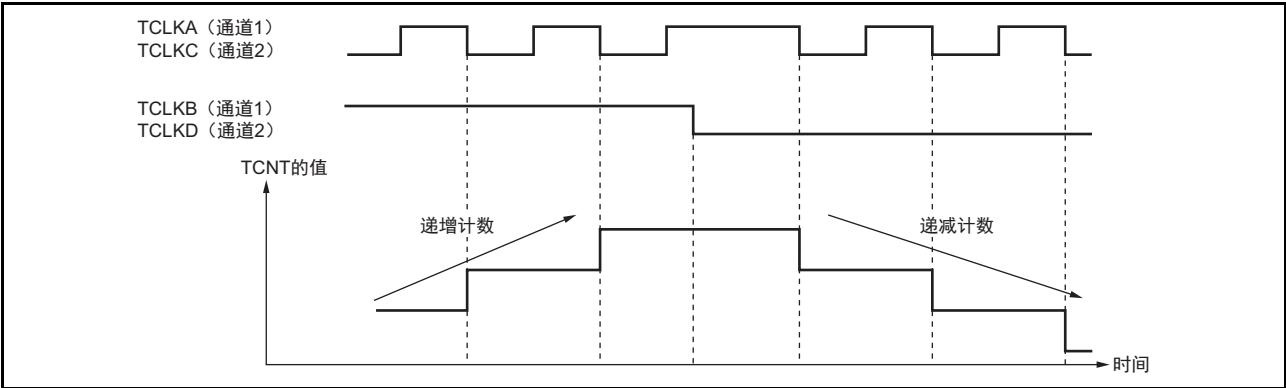


图 11.31 相位计数模式 2 的运行例

表 11.47 相位计数模式 2 的递增 / 递减计数条件

TCLKA（通道 1） TCLKC（通道 2）	TCLKB（通道 1） TCLKD（通道 2）	运行内容
高电平	上升沿	不计数（Don't care）
低电平	下降沿	不计数（Don't care）
上升沿	低电平	不计数（Don't care）
下降沿	高电平	递增计数
高电平	下降沿	不计数（Don't care）
低电平	上升沿	不计数（Don't care）
上升沿	高电平	不计数（Don't care）
下降沿	低电平	递减计数

【符号说明】

上升沿

下降沿

(c) 相位计数模式 3

相位计数模式 3 的运行例如图 11.32 所示，TCNT 的递增 / 递减计数条件如表 11.48 所示。

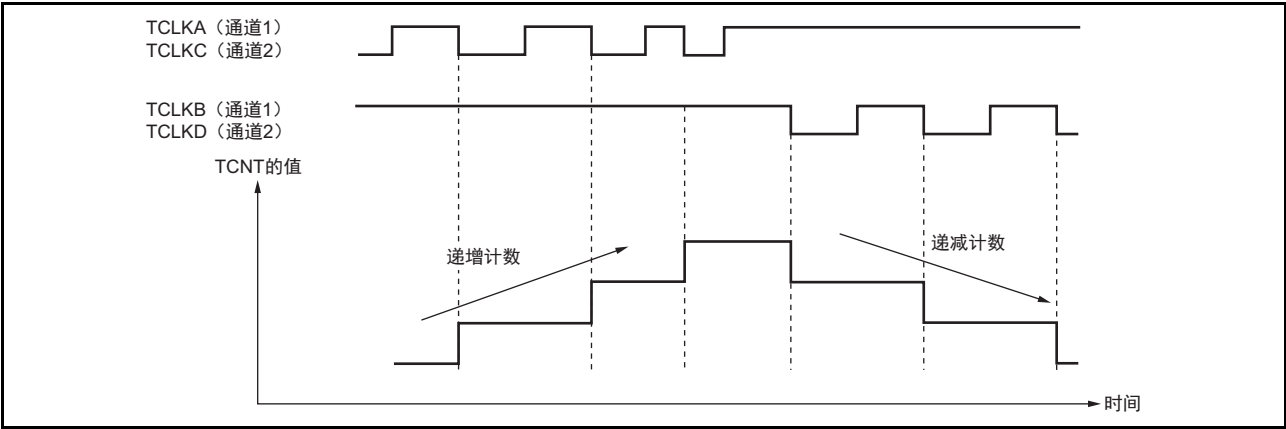


图 11.32 相位计数模式 3 的运行例

表 11.48 相位计数模式 3 的递增 / 递减计数条件

TCLKA（通道 1） TCLKC（通道 2）	TCLKB（通道 1） TCLKD（通道 2）	运行内容
高电平	上升沿	不计数（Don't care）
低电平	下降沿	不计数（Don't care）
上升沿	低电平	不计数（Don't care）
下降沿	高电平	递增计数
高电平	下降沿	递减计数
低电平	上升沿	不计数（Don't care）
上升沿	高电平	不计数（Don't care）
下降沿	低电平	不计数（Don't care）

【符号说明】

上升沿

下降沿

(d) 相位计数模式 4

相位计数模式 4 的运行例如图 11.33 所示，TCNT 的递增 / 递减计数条件如表 11.49 所示。

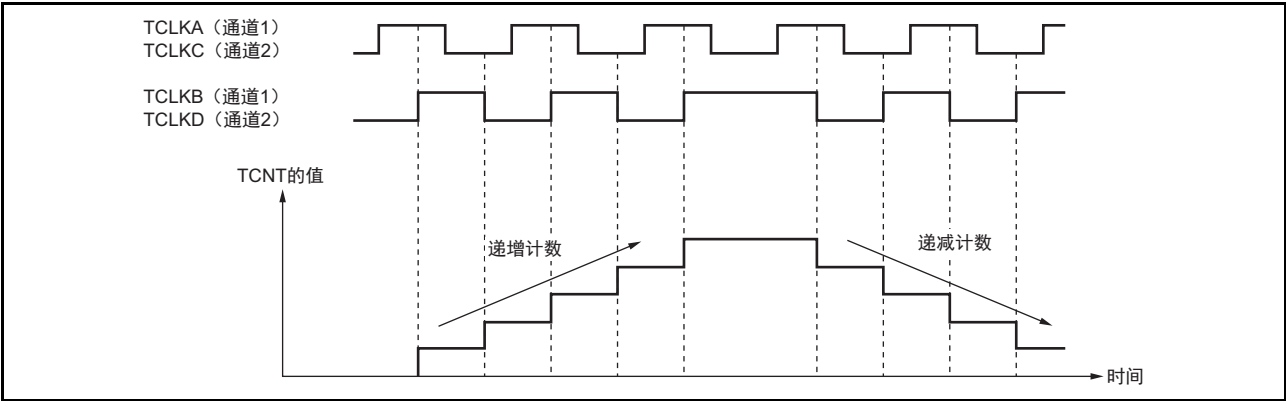


图 11.33 相位计数模式 4 的运行例

表 11.49 相位计数模式 4 的递增 / 递减计数条件

TCLKA（通道 1） TCLKC（通道 2）	TCLKB（通道 1） TCLKD（通道 2）	运行内容
高电平	上升沿	递增计数
低电平	下降沿	
上升沿	低电平	不计数（Don't care）
下降沿	高电平	
高电平	下降沿	递减计数
低电平	上升沿	
上升沿	高电平	不计数（Don't care）
下降沿	低电平	

【符号说明】

上升沿

下降沿

(3) 相位计数模式的应用例

通道 1 设定为相位计数模式，与通道 0 一起输入伺服马达的 2 相编码器脉冲后，检测位置或速度的例子如图 11.34 所示。

通道 1 设定为相位计数模式 1，将编码器脉冲的 A 相和 B 相输入至 TCLKA 和 TCLKB。

通道 0 可通过 TGRC_0 的比较匹配清除 TCNT 计数器，TGRC_0 和 TGRA_0 用作比较匹配功能，设定速度控制周期和位置控制周期。TGRB_0 用作输入捕捉功能，使 TGRB_0 和 TGRD_0 缓冲运行。TGRB_0 的输入捕捉源设置为通道 1 的计时器输入时钟，检测 2 相编码器的 4 倍频脉冲的脉宽。

通道 1 的 TGRA_1 和 TGRB_1 设定为输入捕捉功能，输入捕捉源选择通道 0 的 TGRA_0 和 TGRC_0 的比较匹配，并保存各控制周期的递增 / 递减计数器的值。

由此可检测正确的位置 / 速度。

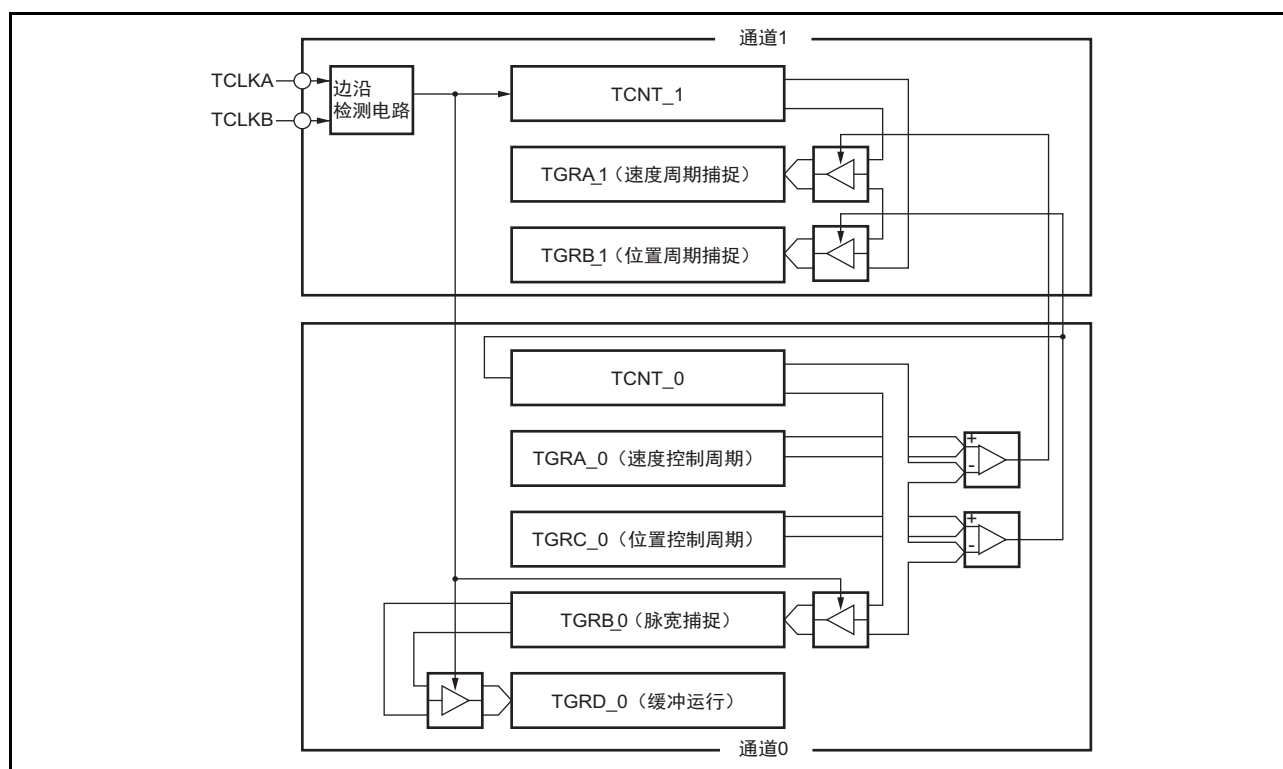


图 11.34 相位计数模式的应用例

11.4.7 复位同步 PWM 模式

在复位同步 PWM 模式，通过组合通道 3、4，可 3 相输出通用一个波形变化点的 PWM 波形（正相 / 反相）。

设定为复位同步 PWM 模式时，TIOC3B、TIOC3D、TIOC4A、TIOC4C、TIOC4B 和 TIOC4D 引脚为 PWM 输出引脚，定时器计数器 3 (TCNT_3) 作为递增计数器运行。

使用的 PWM 输出引脚和寄存器的设定分别如表 11.50 和表 11.51 所示。

表 11.50 复位同步 PWM 模式时的输出引脚

通道	输出引脚	说 明
3	TIOC3B	PWM 输出引脚 1
	TIOC3D	PWM 输出引脚 1' (PWM 输出 1 的反相波形)
4	TIOC4A	PWM 输出引脚 2
	TIOC4C	PWM 输出引脚 2' (PWM 输出 2 的反相波形)
	TIOC4B	PWM 输出引脚 3
	TIOC4D	PWM 输出引脚 3' (PWM 输出 3 的反相波形)

表 11.51 复位同步 PWM 模式时的寄存器设定

寄存器	设定内容
TCNT_3	初始设定 H'0000
TCNT_4	初始设定 H'0000
TGRA_3	设定 TCNT_3 的计数周期
TGRB_3	设定从 TIOC3B、TIOC3D 引脚输出的 PWM 波形的变化点
TGRA_4	设定从 TIOC4A、TIOC4C 引脚输出的 PWM 波形的变化点
TGRB_4	设定从 TIOC4B、TIOC4D 引脚输出的 PWM 波形的变化点

(1) 复位同步 PWM 模式的设定步骤例

复位同步 PWM 模式的设定步骤例如图 11.35 所示。

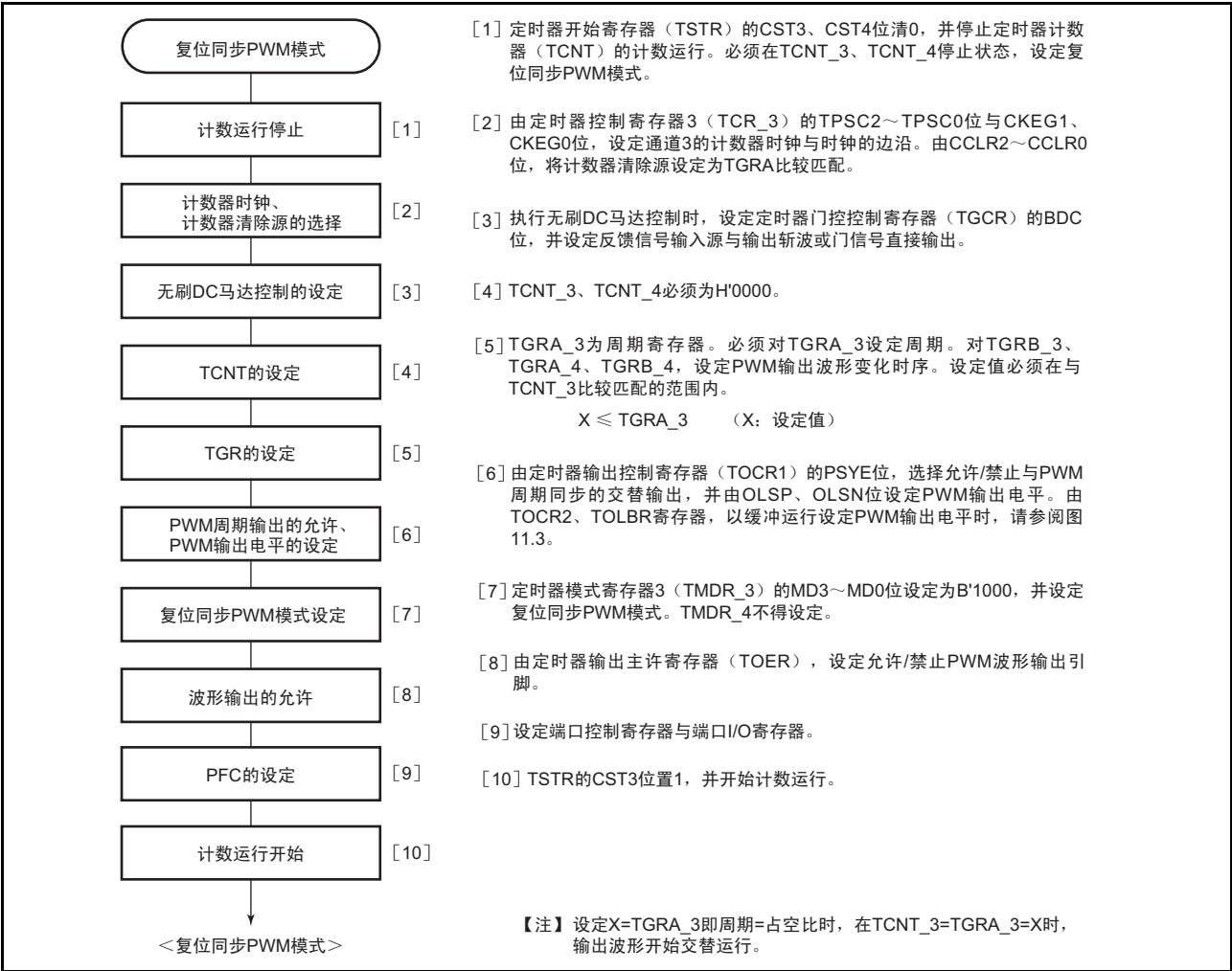


图 11.35 复位同步 PWM 模式的设定步骤例

(2) 复位同步 PWM 模式的运行例

复位同步 PWM 模式的运行例如图 11.36 所示。

在复位同步 PWM 模式，TCNT_3 和 TCNT_4 作为递增计数器运行。TCNT_3 和 TGRA_3 比较匹配时，清除计数器，并从 H'0000 重新开始递增计数。每当 TGRB_3、TGRA_4、TGRB_4 分别产生比较匹配和计数器清除时，PWM 输出引脚交替输出。

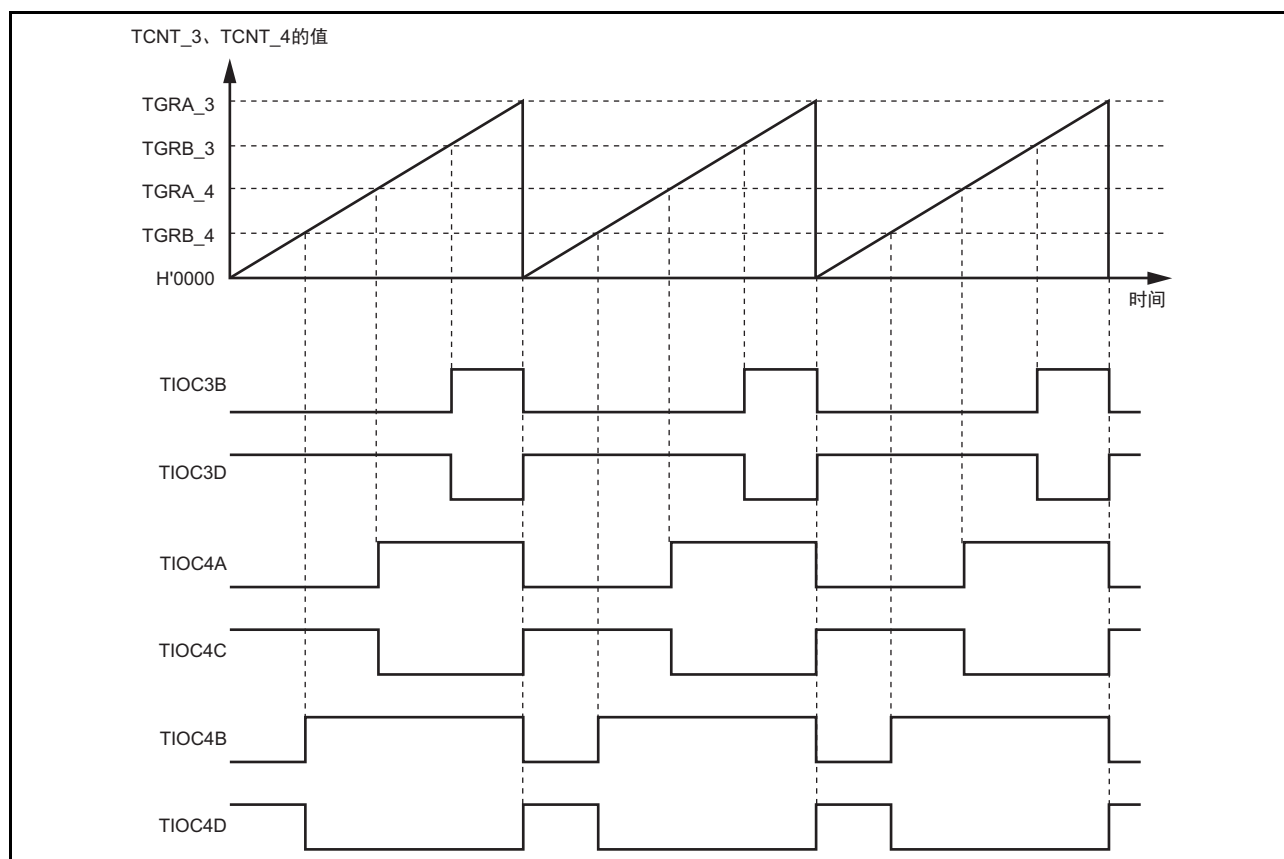


图 11.36 复位同步 PWM 模式的运行例（设定 TOCR 的 OLSN=1、OLSP=1 时）

11.4.8 互补 PWM 模式

在互补 PWM 模式，通过组合通道 3、4，可 3 相输出正、反相为非重叠关系的 PWM 波形，也可设定为非重叠时间。

设定为互补 PWM 模式时，TIOC3B、TIOC3D、TIOC4A、TIOC4B、TIOC4C、TIOC4D 引脚为 PWM 输出引脚，TIOC3A 引脚可设定为与 PWM 周期同步的交替输出。

TCNT_3 和 TCNT_4 作为递增 / 递减计数器运行。

使用的 PWM 输出引脚和寄存器的设定分别如表 11.52 和表 11.53 所示。

作为端口功能，支持由外部信号直接关闭 PWM 输出。

表 11.52 互补 PWM 模式时的输出引脚

通道	输出引脚	说 明
3	TIOC3A	与 PWM 周期同步的交替输出 (或输入 / 输出端口)
	TIOC3B	PWM 输出引脚 1
	TIOC3C	输入 / 输出端口 *
	TIOC3D	PWM 输出引脚 1' (和 PWM 输出 1 有非重叠关系的反相波形, 也可设定无非重叠时间)
4	TIOC4A	PWM 输出引脚 2
	TIOC4C	PWM 输出引脚 2' (和 PWM 输出 2 有非重叠关系的反相波形, 也可设定无非重叠时间)
	TIOC4B	PWM 输出引脚 3
	TIOC4D	PWM 输出引脚 3' (和 PWM 输出 3 有非重叠关系的反相波形, 也可设定无非重叠时间)

【注】 * TIOC3C 引脚在互补 PWM 模式, 不得设定为定时器的输入 / 输出引脚。

表 11.53 互补 PWM 模式时的寄存器设定

通道	计数器 / 寄存器	说 明	由 CPU 读取 / 写入
3	TCNT_3	从空载时间寄存器所设定的值开始递增计数	通过设定 TRWER* 可执行屏蔽
	TGRA_3	设定 TCNT_3 的上限值 (1/2 的载波周期 + 空载时间)	通过设定 TRWER* 可执行屏蔽
	TGRB_3	PWM 输出 1 的比较寄存器	通过设定 TRWER* 可执行屏蔽
	TGRC_3	TGRA_3 的缓冲寄存器	总是可读取 / 写入
	TGRD_3	PWM 输出 1/TGRB_3 的缓冲寄存器	总是可读取 / 写入
4	TCNT_4	初始设定 H'0000, 并开始递增计数	通过设定 TRWER* 可执行屏蔽
	TGRA_4	PWM 输出 2 的比较寄存器	通过设定 TRWER* 可执行屏蔽
	TGRB_4	PWM 输出 3 的比较寄存器	通过设定 TRWER* 可执行屏蔽
	TGRC_4	PWM 输出 2/TGRA_4 的缓冲寄存器	总是可读取 / 写入
	TGRD_4	PWM 输出 3/TGRB_4 的缓冲寄存器	总是可读取 / 写入
定时器空载时间数据寄存器 (TDDR)		设定 TCNT_4 和 TCNT_3 的偏移量值 (空载时间的值)	通过设定 TRWER* 可执行屏蔽
定时器周期数据寄存器 (TCDR)		设定 TCNT_4 的上限值 (1/2 载波周期)	通过设定 TRWER* 可执行屏蔽
定时器周期缓冲寄存器 (TCBR)		TCDR 的缓冲寄存器	总是可读取 / 写入
副计数器 (TCNTS)		生成空载时间的副计数器	只读
暂存器 1 (TEMP1)		PWM 输出 1/TGRB_3 的暂存器	不可读取 / 写入
暂存器 2 (TEMP2)		PWM 输出 2/TGRA_4 的暂存器	不可读取 / 写入
暂存器 3 (TEMP3)		PWM 输出 3/TGRB_4 的暂存器	不可读取 / 写入

【注】 * 通过设定 TRWER (定时器读取 / 写入允许寄存器), 可允许 / 禁止存取。

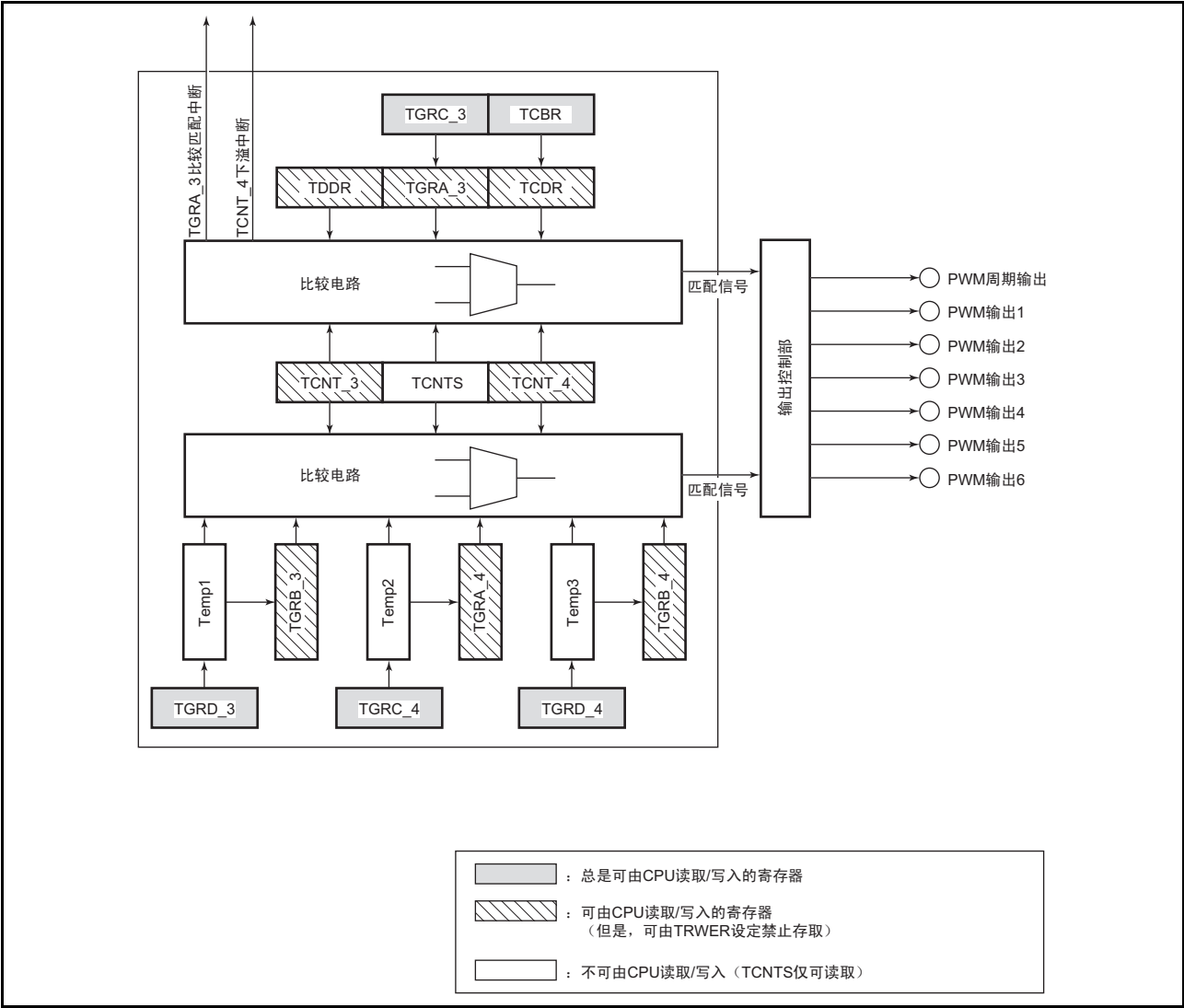


图 11.37 互补 PWM 模式时通道 3、4 的框图

(1) 互补 PWM 模式的设定步骤例

互补 PWM 模式的设定步骤例如图 11.38 所示。

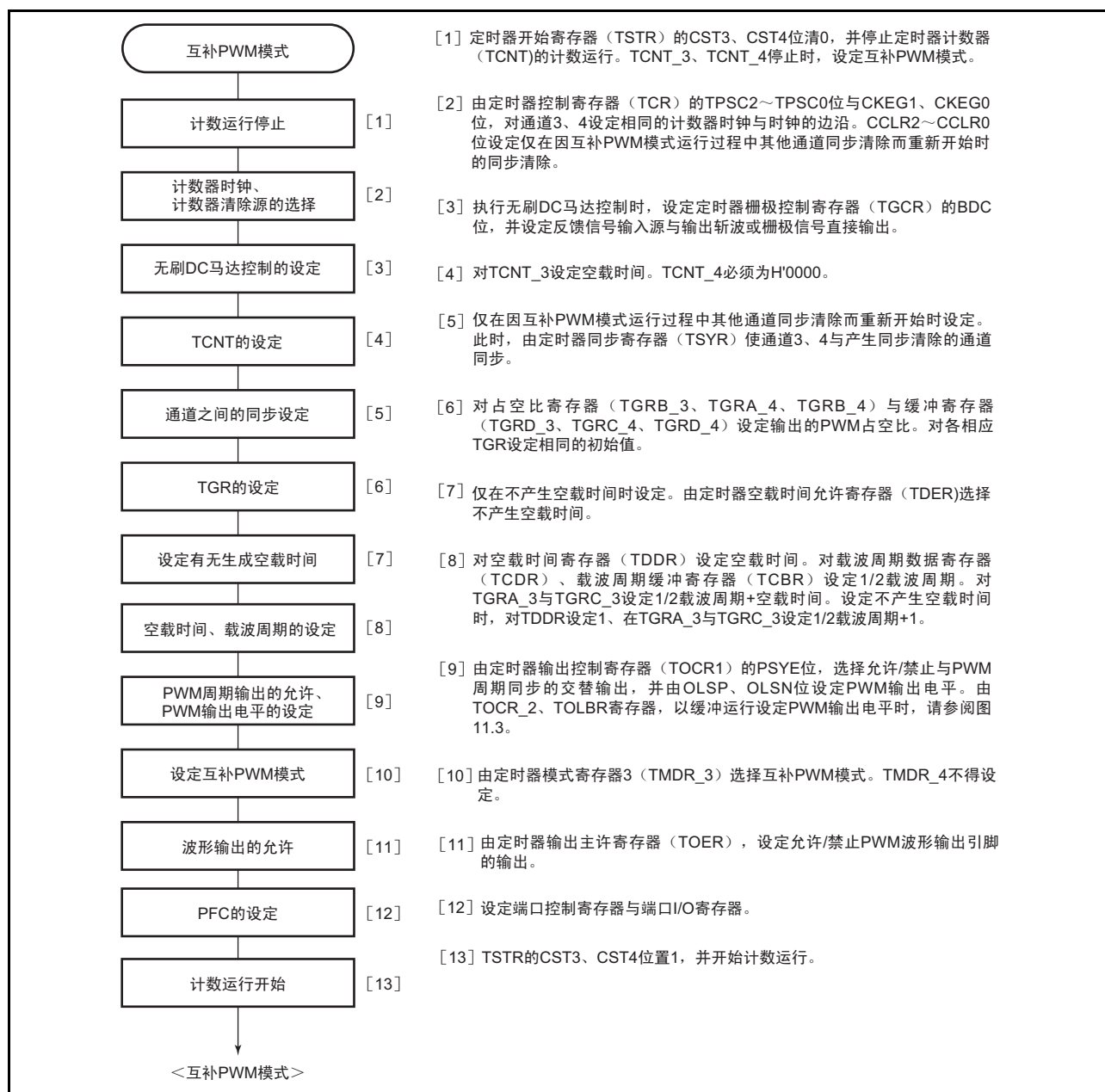


图 11.38 互补 PWM 模式的设定步骤例

(2) 互补 PWM 模式的运行概要

在互补 PWM 模式, 可实现 6 相 PWM 输出。互补 PWM 模式的计数器运行和互补 PWM 模式的运行例分别如图 11.39 和图 11.40 所示。

(a) 计数器的运行

在互补 PWM 模式, TCNT_3、TCNT_4 和 TCNTS 等 3 个计数器递增 / 递减计数。

TCNT_3 设定为互补 PWM 模式, 且 TSTR 的 CST 位为 0 时, 对 TDDR 设定的值自动为初始值。

CST 位设定为 1 时, 递增计数至 TGRA_3 的设定值, 如果与 TGRA_3 匹配, 则转换为递减计数。之后, 如果与 TDDR 匹配, 则转换为递增计数, 重复此运行。

TCNT_4 的初始值设定为 H'0000。

CST 位设定为 1 时, 与 TCNT_3 同步递增计数, 如果与 TCDR 匹配, 则转换为递减计数。之后, 如果与 H'0000 匹配, 则转换为递增计数, 重复此运行。

TCNTS 为只读计数器, 无需设定初始值。

TCNT_3、4 递增 / 递减计数时, 如果 TCNT_3 与 TCDR 匹配, 则开始递减计数; 如果 TCNTS 与 TCDR 匹配, 则转换为递增计数。如果与 TGRA_3 匹配, 则清除为 H'0000。

TCNT_3、TCNT_4 递减计数时, 如果 TCNT_4 与 TDDR 匹配, 则开始递增计数; 如果 TCNTS 与 TDDR 匹配, 则转换为递减计数; 如果与 H'0000 匹配, 则 TCNTS 设定为 TGRA_3 的值。

TCNTS 仅在计数运行期间与设定 PWM 占空比的比较寄存器和暂存器进行比较。

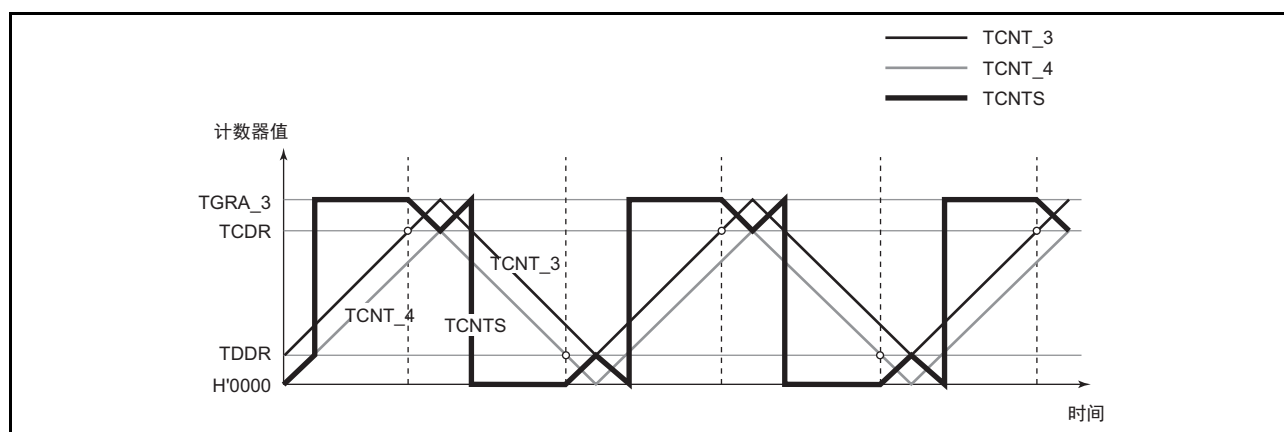


图 11.39 互补 PWM 模式的计数器运行

(b) 寄存器的运行

在互补 PWM 模式，使用比较寄存器、缓冲寄存器和暂存器等 9 个寄存器。互补 PWM 模式的运行例如图 11.40 所示。

为了输出 PWM，总是与计数器比较的寄存器有：TGRB_3、TGRA_4 及 TGRB_4。这些寄存器与计数器匹配时，输出定时器输出控制寄存器 (TOCR) 的 OLSN、OLSP 位设定的值。

这些比较寄存器的缓冲寄存器为 TGRD_3、TGRC_4 及 TGRD_4。

缓冲寄存器和比较寄存器之间有暂存器，但不可由 CPU 存取暂存器。

为了更改比较寄存器的数据，必须对相应的缓冲寄存器写入更改数据。缓冲寄存器可随时读取 / 写入。

在 Ta 区间，随时将写入缓冲寄存器的数据传送至暂存器，但在 Tb 区间不可传送至暂存器。Tb 区间结束后，将在此区间写入缓冲寄存器的数据传送至暂存器。

如果 Tb 区间结束的 TCNTS 在递增计数时与 TGRA_3 匹配，或在递减计数时与 H'0000 匹配，则将传送至暂存器的值传送到比较寄存器。由定时器模式寄存器 (TMDR) 的 MD3 ~ MD0 位，可选择从此暂存器向比较寄存器传送的时序。如图 11.40 是选择在波谷更改的模式为例。

在不向暂存器传送数据的 Tb (图 11.40 中的 Tb1) 区间，暂存器具有与比较寄存器相同的功能，可与计数器比较。在此区间，对于 1 相输出有 2 个比较匹配寄存器，向比较寄存器输入更改前的数据，向暂存器输入新更改的数据。此区间内，TCNT_3、TCNT_4 及 TCNTS 这 3 个计数器与比较寄存器、暂存器这 2 个寄存器比较，可控制 PWM 输出。

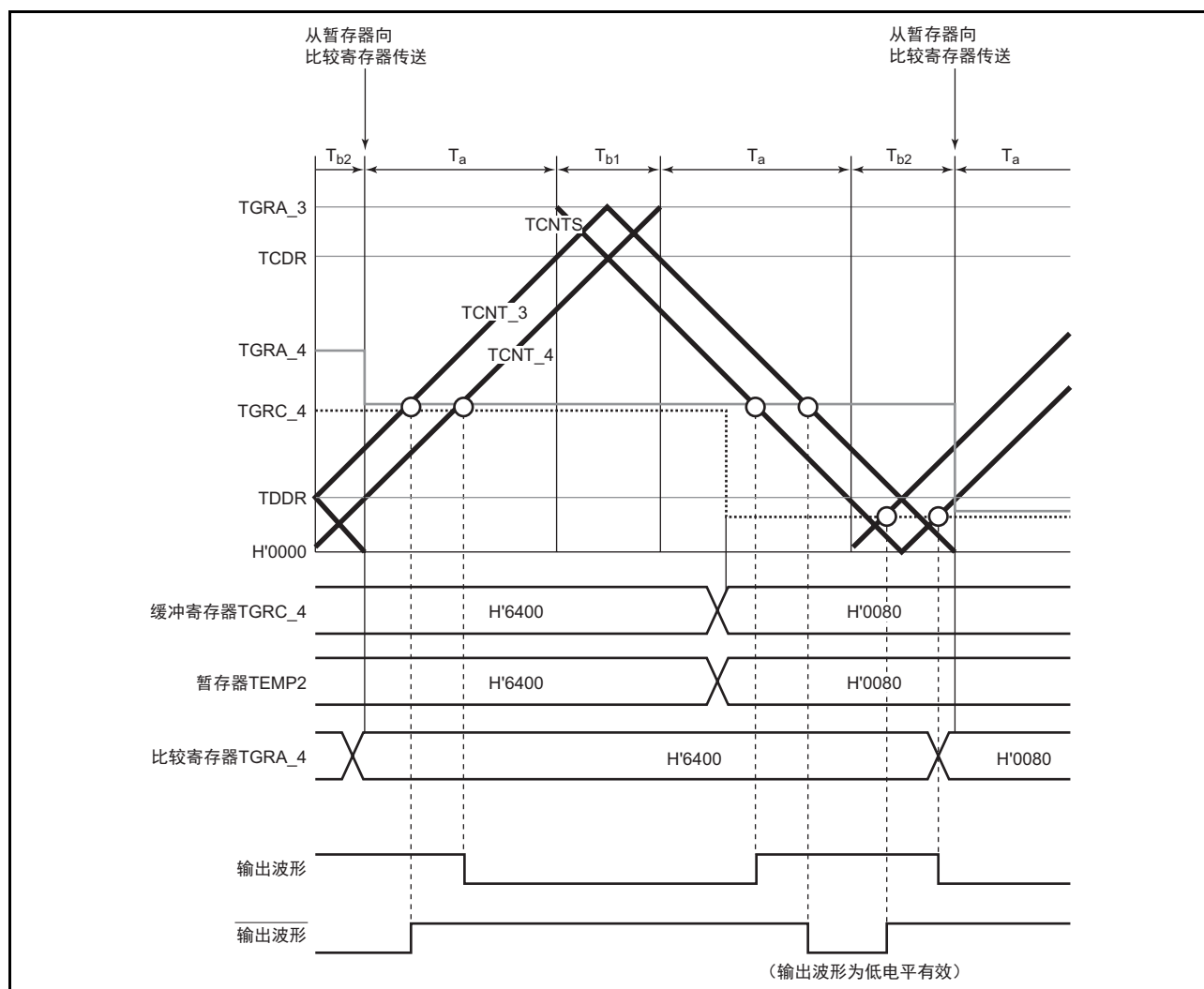


图 11.40 互补 PWM 模式运行例

(c) 初始设定

互补 PWM 模式有 6 个需要初始设定的寄存器，还有 1 个设定是否生成空载时间的寄存器（仅在不生成空载时间时设定）。

由定时器模式寄存器（TMDR）的 MD3 ~ MD0 位设定为互补 PWM 模式前，必须设定以下寄存器的初始值。

TGRC_3 作为 TGRA_3 的缓冲寄存器运行，设定 1/2 的 PWM 载波周期 + 空载时间 T_d 。定时器周期缓冲寄存器（TCBR）作为定时器周期数据寄存器（TCDR）的缓冲寄存器运行，设定 1/2 的 PWM 载波周期。对定时器空载时间数据寄存器（TDDR）设定空载时间 T_d 。

不生成空载时间时，将定时器空载时间允许寄存器（TDER）的 TDER 位设定为 0，对 TGRC_3 和 TGRA_3 设定 1/2 的 PWM 载波周期 +1，对 TDDR 设定 1。

对 3 个缓冲寄存器 TGRD_3、TGRC_4 及 TGRD_4，可分别设定 PWM 占空比的初始值。

设定为互补 PWM 模式的同时，将除 TDDR 之外的 5 个缓冲寄存器所设定的值分别传送至对应的比较寄存器。

必须在设定为互补 PWM 模式前，将 TCNT_4 设定为 H'0000。

表 11.54 需初始设定的寄存器和计数器

寄存器 / 计数器	设定值
TGRC_3	1/2 的 PWM 载波周期 + 空载时间 T_d (由 TDER 设定为不生成空载时间时，为 1/2 的 PWM 载波周期 +1)
TDDR	空载时间 T_d (由 TDER 设定为不生成空载时间时，为 1) 空载时间
TCBR	1/2 的 PWM 载波周期
TGRD_3、TGRC_4、 TGRD_4	各相 PWM 占空比的初始值
TCNT_4	H'0000

【注】 TGRC_3 的设定值必须为 1/2 的 TCBR 设定的 PWM 载波周期与 TDDR 设定的空载时间 T_d 值之和。但由 TDER 设定为不生成空载时间时，必须设定为 1/2 的 PWM 载波周期 +1。

(d) PWM 输出电平の設定

在互补 PWM 模式，通过定时器输出控制寄存器 1（TOCR1）的 OLSN、OLSP 位，或定时器输出控制寄存器 2（TOCR2）的 OLS1P ~ OLS3P、OLS1N ~ OLS3N 位，设定 PWM 脉冲的输出电平。

6 相输出电平可设定为正反相各 3 相。

必须在解除互补 PWM 模式的状态，设定 / 更改输出电平。

(e) 空载时间的设定

在互补 PWM 模式，输出正、反相为非重叠关系的 PWM 脉冲，此非重叠时间称为空载时间。

对定时器空载时间数据寄存器（TDDR）设定非重叠时间。TDDR 设定的值为 TCNT_3 的计数器开始值，生成 TCNT_3 和 TCNT_4 的非重叠时间，必须在解除互补 PWM 模式的状态，更改 TDDR 的内容。

(f) 不生成空载时间的设定

设定为不生成空载时间，就是将定时器空载时间允许寄存器 (TDER) 的 TDER 位设定为 0。仅可在 TDER=1 的状态，读取 TDER 后对 TDER 写入 0 时，将 TDER 设定为 0。

对 TGRA_3、TGRC_3 设定 1/2 的 PWM 载波周期 +1，对定时器空载时间数据寄存器 (TDDR) 设定 1。

如果设定为不生成空载时间，则可输出无空载时间的 PWM 波形。不生成空载时间的运行例如图 11.41 所示。

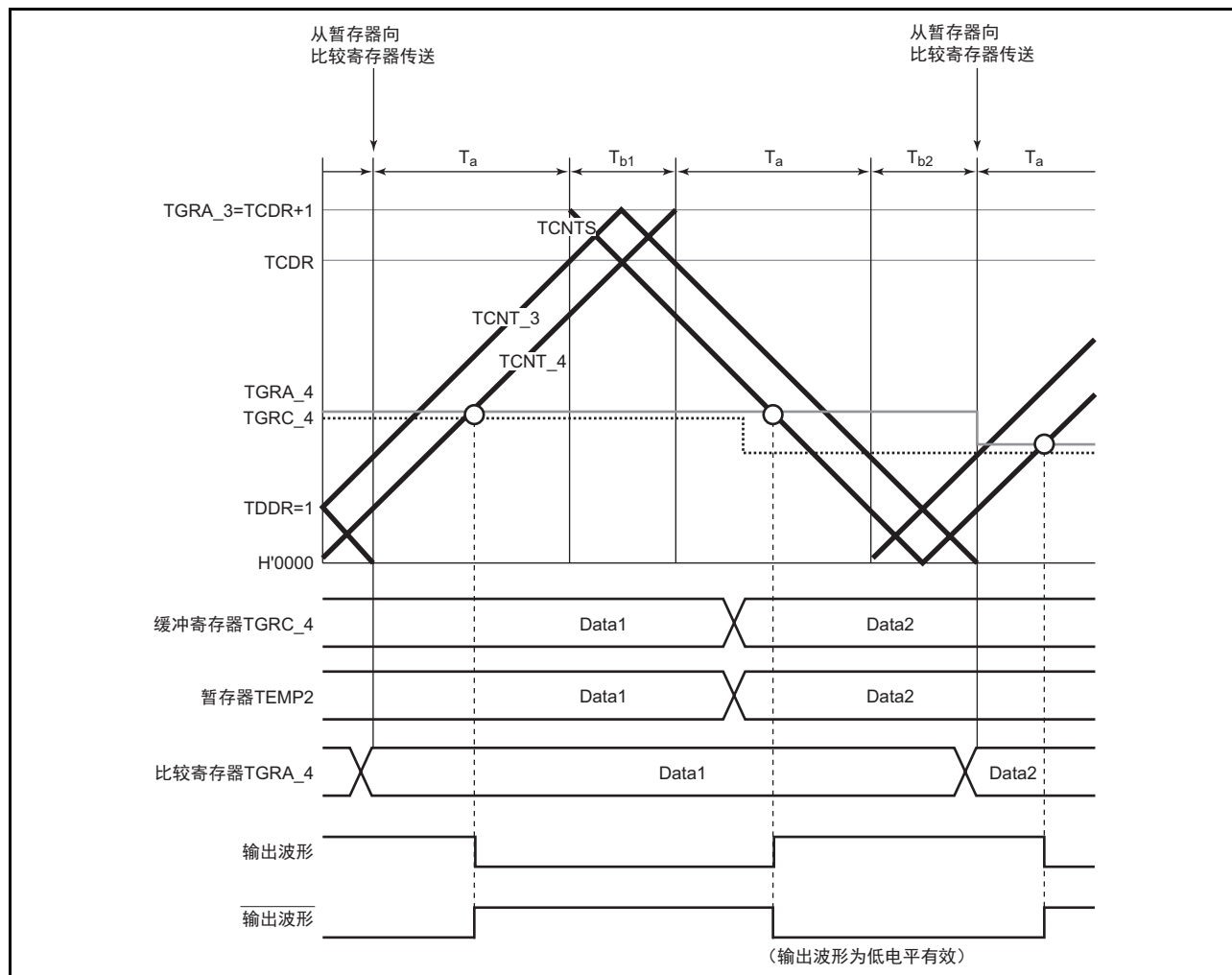


图 11.41 不生成空载时间的运行例

(g) PWM 周期的设定

互补 PWM 模式下, 可对设定 TCNT3 上限值的 TGRA_3 和设定 TCNT_4 上限值的 TCDR 这 2 个寄存器设定 PWM 脉冲的周期。这 2 个寄存器的关系必须如下设定:

生成空载时间: TGRA_3 的设定值 = TCDR 的设定值 + TDDR 的设定值

不生成空载时间: TGRA_3 的设定值 = TCDR 的设定值 + 1。

另外, 必须通过对缓冲寄存器的 TGRC_3、TCBR 设定值来设定 TGRA_3、TCDR。在定时器模式寄存器 (TMDR) 的 MD3 ~ MD0 选择的传送时序, 将 TGRC_3 和 TCBR 的设定值同时传送至 TGRA_3 和 TCDR。

在波峰更新数据时, 从下一周期反映更改的 PWM 周期; 在波谷更新数据时, 从该周期反映更改的 PWM 周期。在波峰更改 PWM 周期的运行例如图 11.42 所示。

有关各缓冲寄存器数据的更新方法, 详情参阅以下“(h) 寄存器数据的更新”。

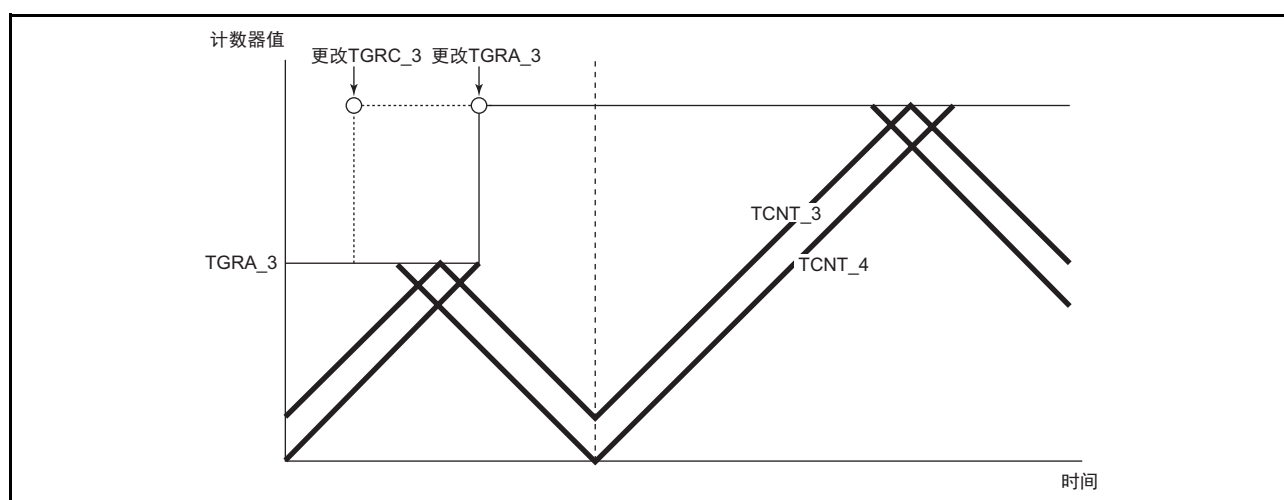


图 11.42 PWM 周期的更改例

(h) 寄存器数据的更新

在互补 PWM 模式, 更新比较寄存器的数据时, 使用缓冲寄存器。更新数据可随时写入缓冲寄存器。具有缓冲寄存器并在运行过程中可更改的寄存器有用于 PWM 占空比和载波周期的 5 个。

这些寄存器和缓冲寄存器之间有各自的暂存器。副计数器 TCNTS 不计数期间, 如果更新缓冲寄存器的数据, 暂存器的值也随之改写; TCNTS 计数时, 不从缓冲寄存器向暂存器传送, TCNTS 停止后, 可传送写入缓冲寄存器的值。

暂存器的值在定时器模式寄存器 (TMDR) 的 MD3 ~ MD0 位设定的数据更新时序被传送至比较寄存器。互补 PWM 模式时的数据更新例如图 11.43 所示, 此图是在计数器的波峰和波谷更新数据时的模式。

改写缓冲寄存器的数据时, 最后必须写入 TGRD_4。写入 TGRD_4 后, 5 个寄存器同时从缓冲寄存器向暂存器传送数据。

5 个寄存器均未更新或未更新 TGRD_4 的数据时, 在写入要更新的寄存器数据后, 必须写入 TGRD_4。此时, 写入 TGRD_4 的数据必须和写入前相同。

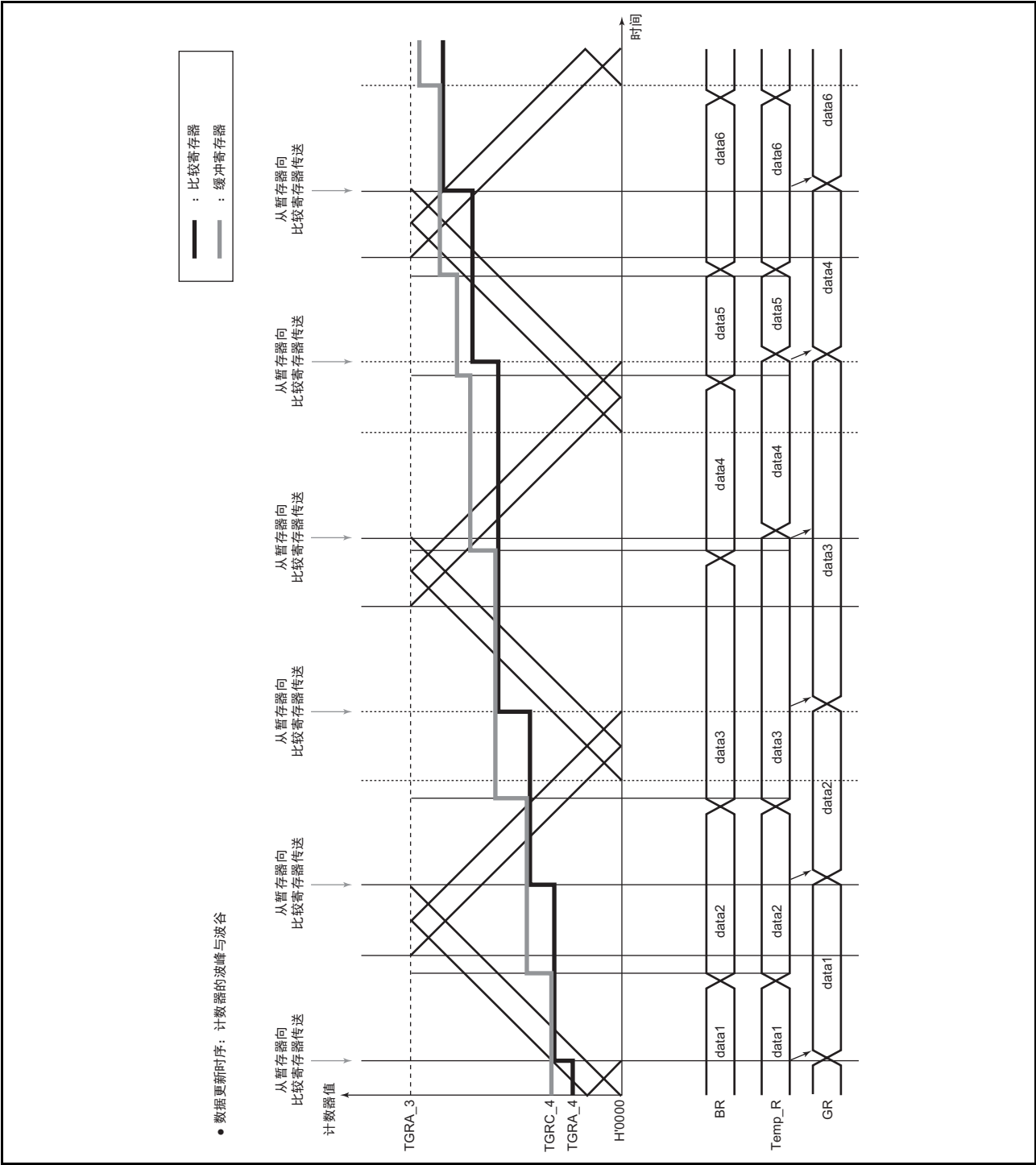


图 11.43 互补 PWM 模式的数据更新例

(i) 互补 PWM 模式的初始输出

在互补 PWM 模式，通过设定定时器输出控制寄存器 1 (TOCR1) 的 OLSN、OLSP 位，或定时器输出控制寄存器 2 (TOCR2) 的 OLS1N ~ OLS3N、OLS1P ~ OLS3P 位，决定初始输出。

此初始输出为 PWM 脉冲的无效电平，在定时器模式寄存器 (TMDR) 设定互补 PWM 模式后，直到 TCNT_4 超过在空载时间寄存器 (TDDR) 设定的值为止，输出此初始输出。互补 PWM 模式的初始输出例如图 11.44 所示。

PWM 占空比的初始值小于 TDDR 值时的波形例如图 11.45 所示。

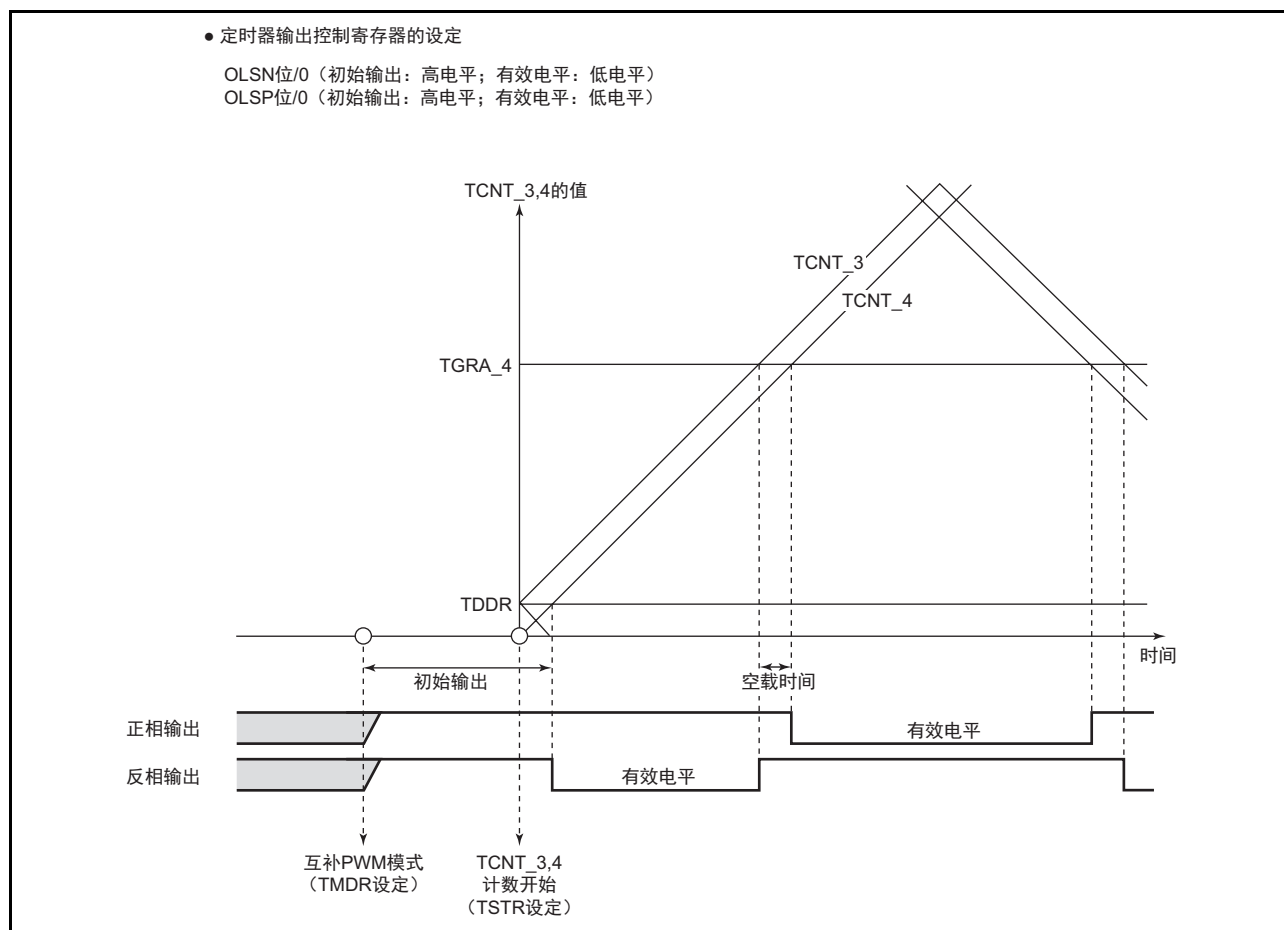


图 11.44 互补 PWM 模式的初始输出例 (1)

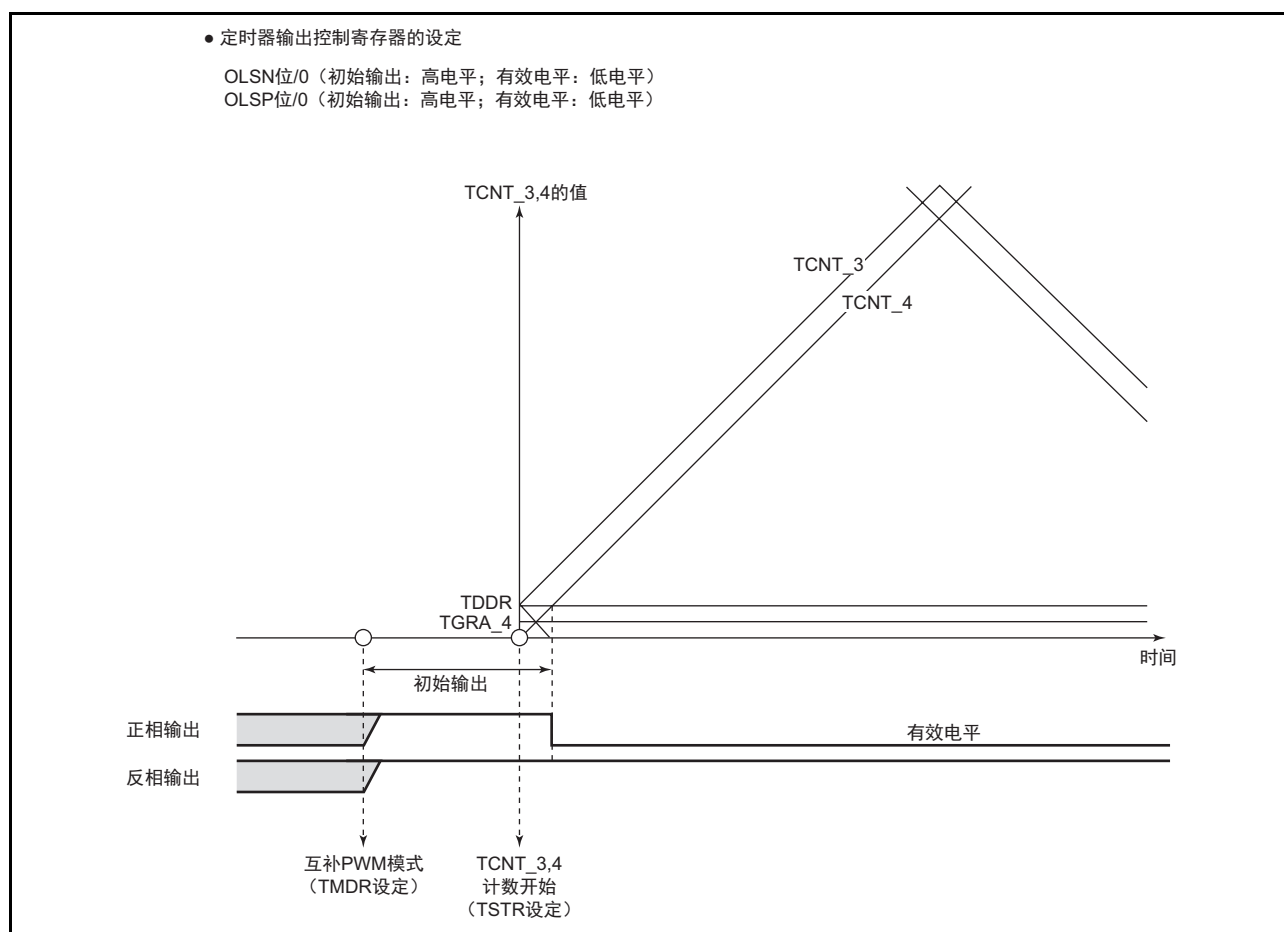


图 11.45 互补 PWM 模式的初始输出例 (2)

(j) 互补 PWM 模式的 PWM 输出生成方法

在互补 PWM 模式, 3 相输出正、反相有非重叠时间的 PWM 波形, 此非重叠时间称为空载时间。

计数器和数据寄存器产生比较匹配时, 通过输出定时器输出控制寄存器选择的输出电平, 生成 PWM 波形。TCNTS 计数期间, 为了生成 0 ~ 100% 的连续 PWM 脉冲, 可同时比较数据寄存器和暂存器的值。此时, 产生 ON、OFF 比较匹配的时序可能会颠倒, 为了确保空载时间, 并且避免正 / 反相的 ON 时间重叠, 必须优先各相 OFF 的比较匹配。互补 PWM 模式的波形生成例如图 11.46 ~ 图 11.48 所示。

通过与实线计数器的比较匹配生成正 / 反相的 OFF 时序, 通过与延迟实线计数器空载时间的虚线计数器的比较匹配, 生成 ON 时序。在 T1 期间, 优先反相 OFF 的 a 的比较匹配, 忽略比 a 先产生的比较匹配; 在 T2 期间, 优先正相 OFF 的 c 的比较匹配, 忽略比 c 先产生的比较匹配。

如图 11.46 所示, 通常按照 a→b→c→d (或 c→d→a'→b') 的顺序产生比较匹配。

比较匹配不符合 a→b→c→d 的顺序时, 反相 OFF 的时间小于空载时间的 2 倍, 因此正相不为 ON。比较匹配不符合 c→d→a'→b' 的顺序时, 正相 OFF 的时间小于空载时间的 2 倍, 因此反相不为 ON。

如图 11.47 所示, 在 a 的比较匹配之后先产生 c 的比较匹配时, 忽略 b 的比较匹配, 并由 d 的比较匹配使反相 OFF。这是由于正相 OFF 时序的 c 先于正相 ON 时序的 b 产生比较匹配, 因此可优先使正相 OFF (正相从 OFF 到 OFF, 所以波形不变)。

同样, 如图 11.48 所示, 在 c 的比较匹配前与暂存器的新数据产生比较匹配 a', 但在产生使正相 OFF 的 c 前, 可忽略其他比较匹配。因此, 反相不为 ON。

在互补 PWM 模式, 可优先执行 OFF 时序的比较匹配, 即使 ON 时序的比较匹配比 OFF 先产生, 也被忽略。

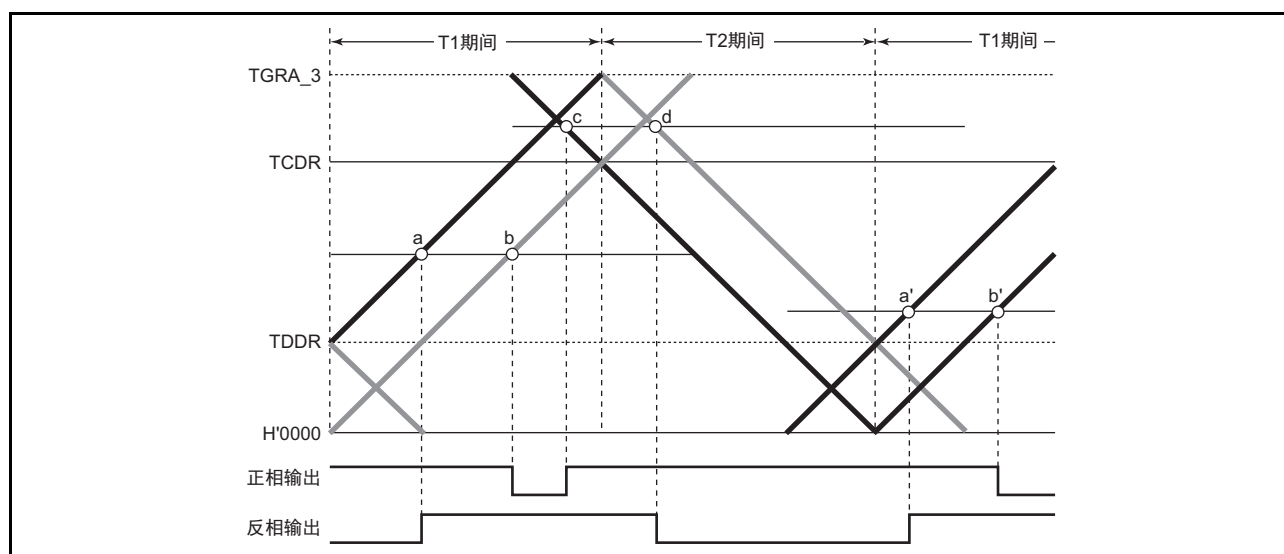


图 11.46 互补 PWM 模式波形输出例 (1)

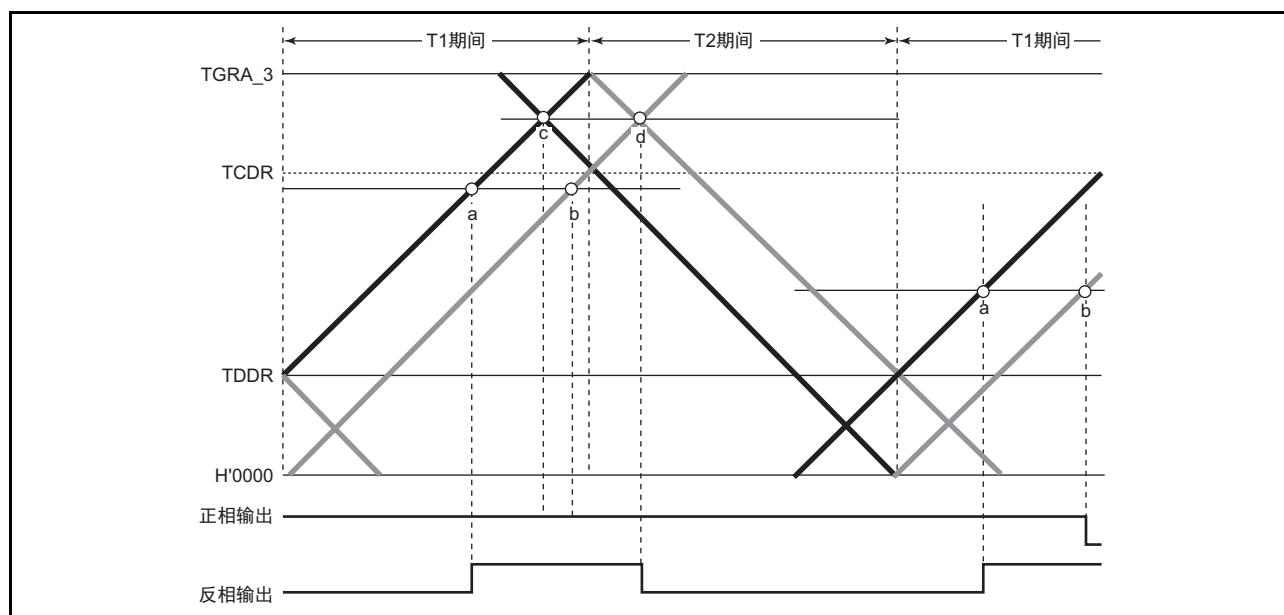


图 11.47 互补 PWM 模式波形输出例 (2)

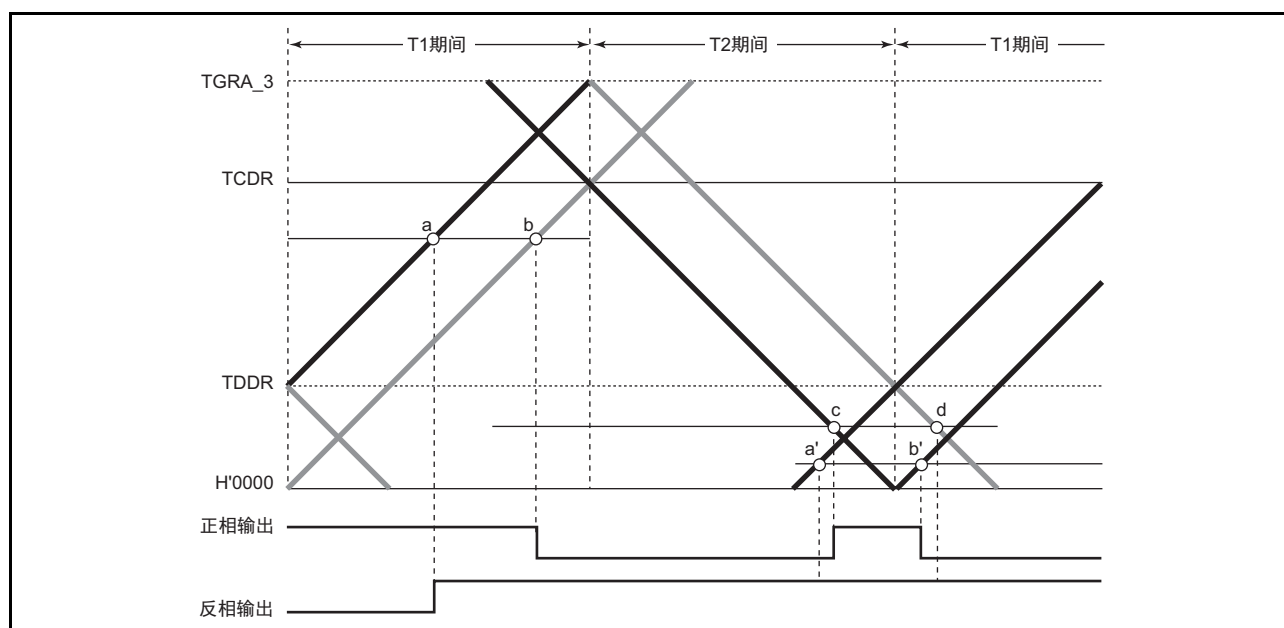


图 11.48 互补 PWM 模式波形输出例 (3)

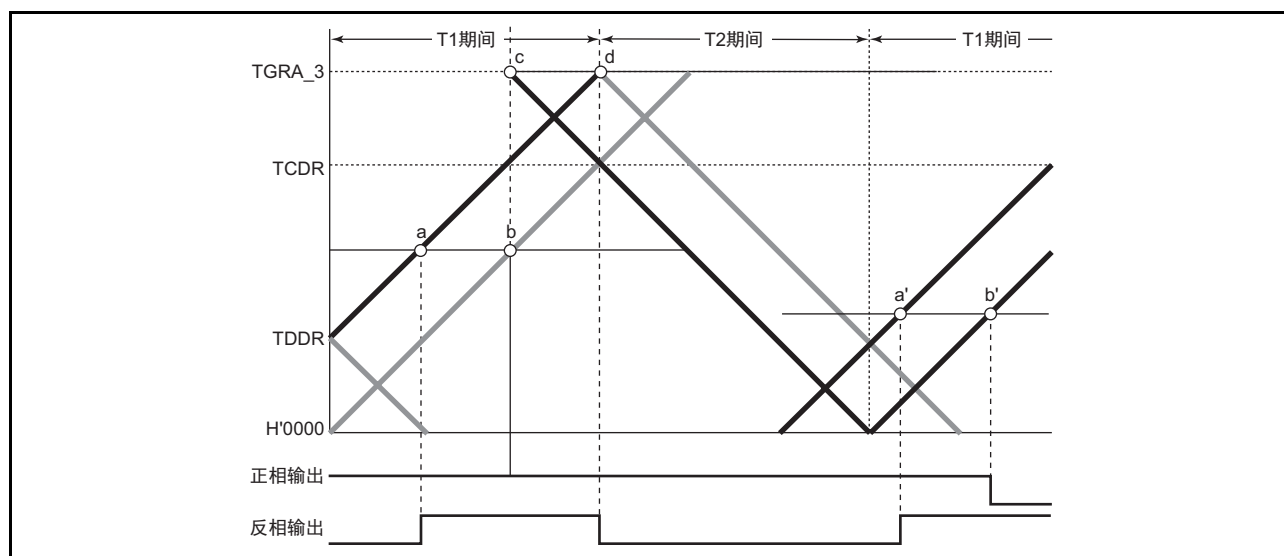


图 11.49 互补 PWM 模式 0%、100% 波形输出例 (1)

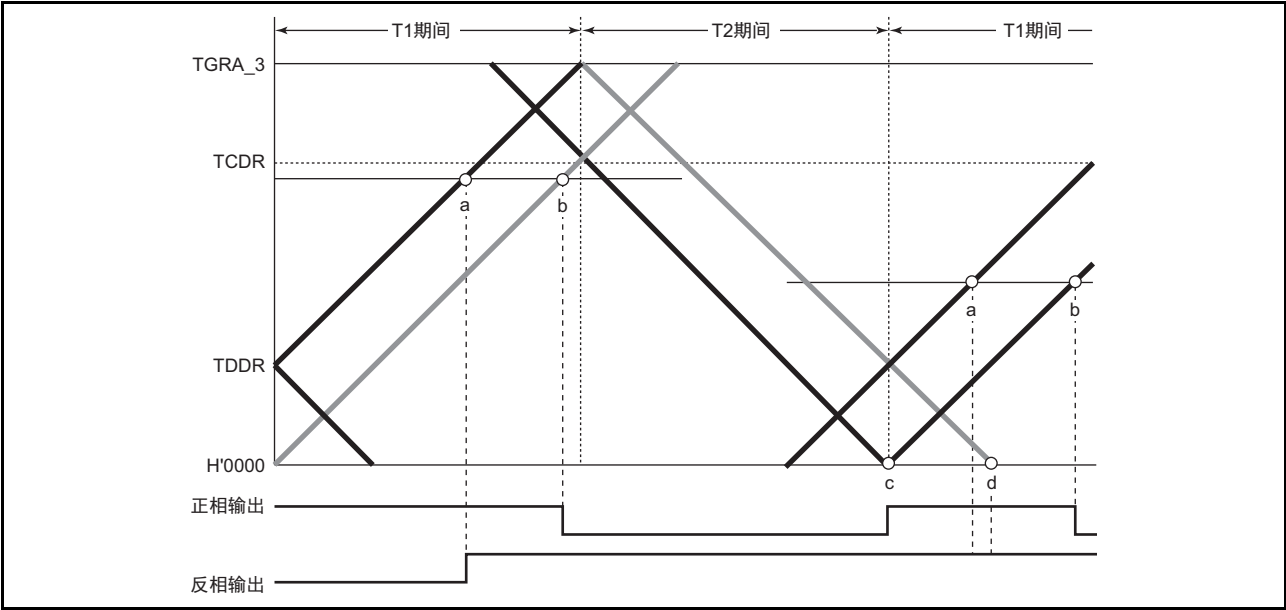


图 11.50 互补 PWM 模式 0%、100% 波形输出例 (2)

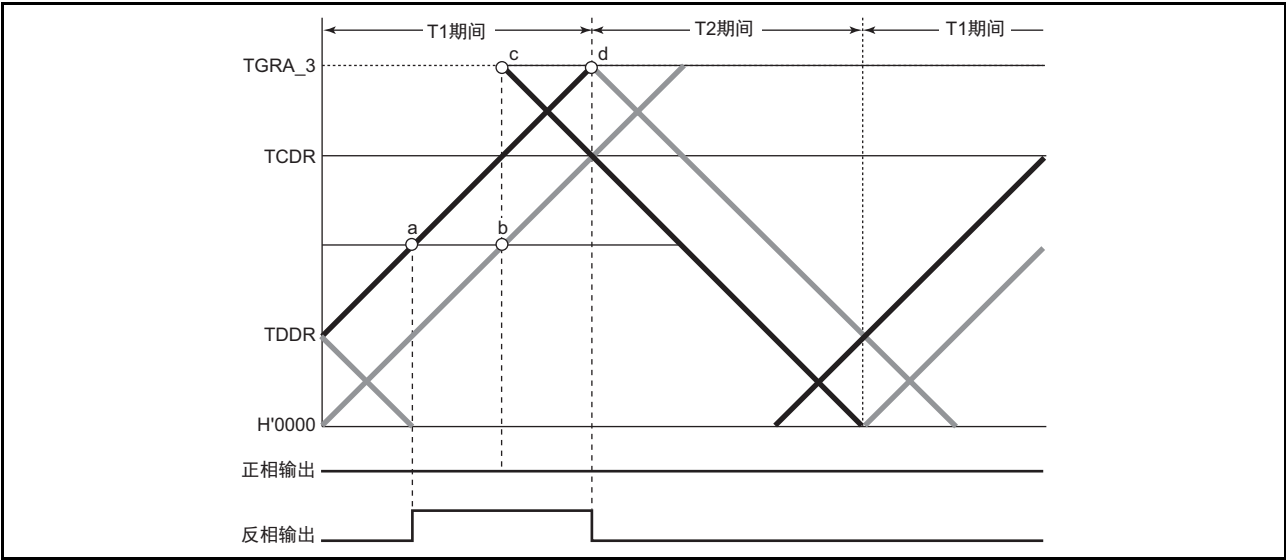


图 11.51 互补 PWM 模式 0%、100% 波形输出例 (3)

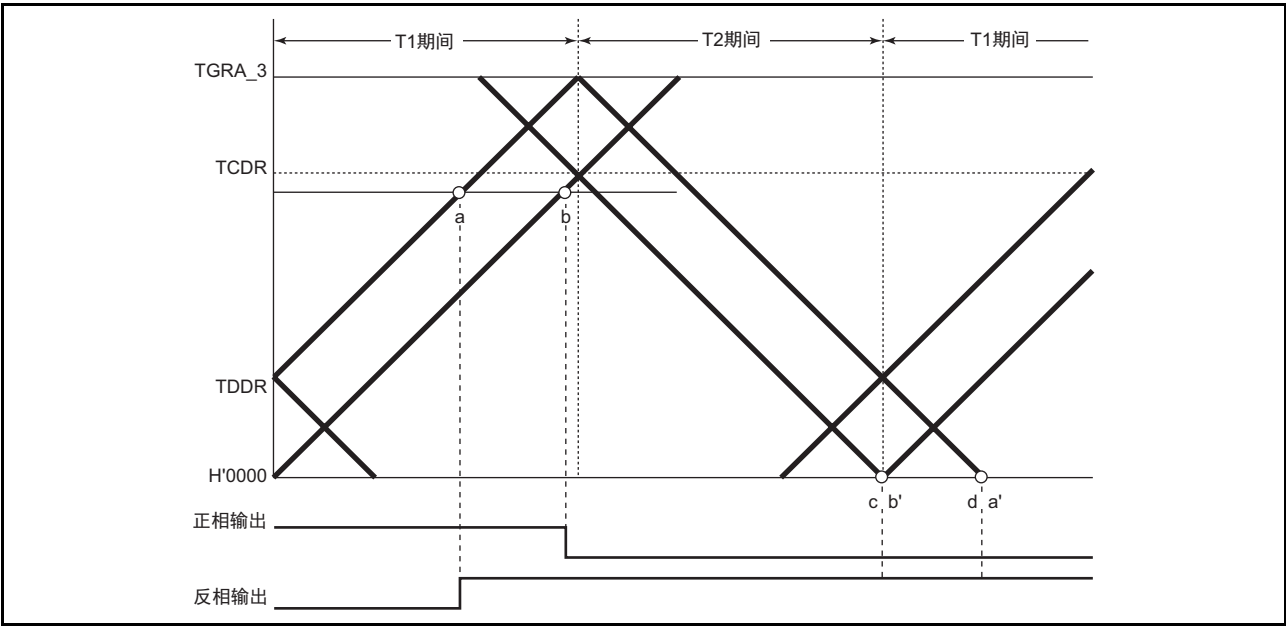


图 11.52 互补 PWM 模式 0%、100% 波形输出例 (4)

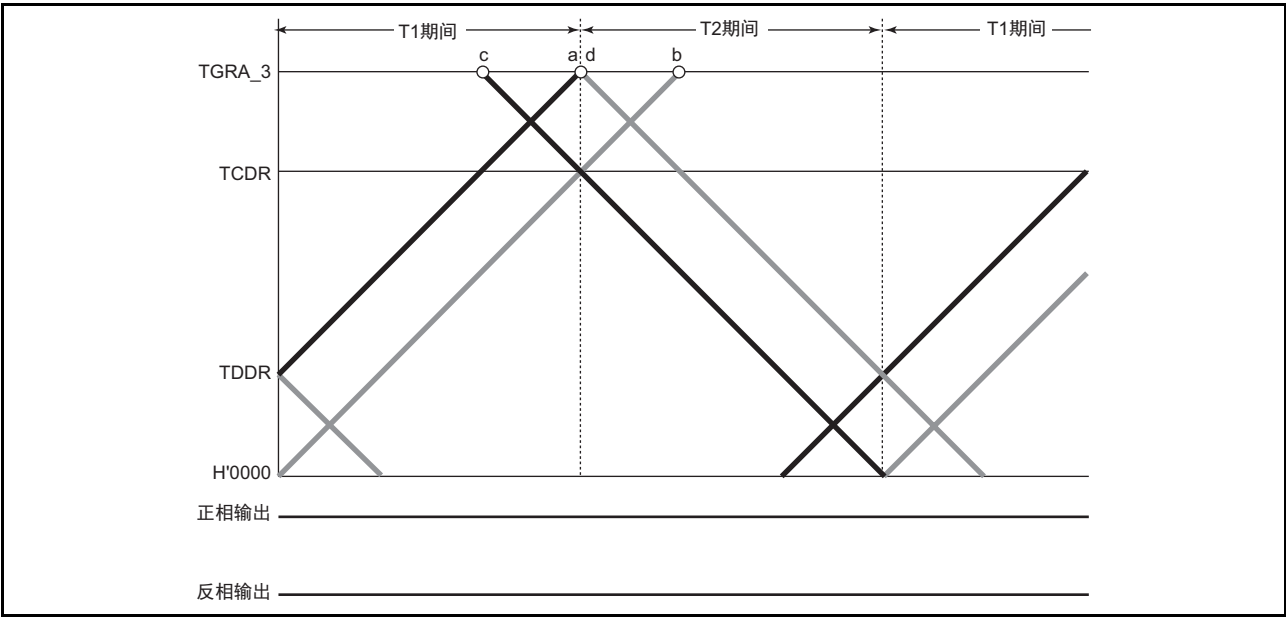


图 11.53 互补 PWM 模式 0%、100% 波形输出例 (5)

(k) 互补 PWM 模式的占空比 0%、100% 的输出

在互补 PWM 模式，可任意输出占空比 0%、100%。输出例如图 11.49 ~ 图 11.53 所示。

数据寄存器的值设定为 H'0000 时，输出占空比 100%，此时的波形为正相 100%ON 状态的波形；数据寄存器的值设定为与 TGRA_3 相同的值时，输出占空比 0%，此时的波形为正相 100%OFF 状态的波形。

此时，比较匹配同时产生 ON 和 OFF，但是如果同时产生相同相的 ON 和 OFF 的比较匹配，则忽略双方的比较匹配，且波形不变。

(l) 与 PWM 周期同步的交替输出

在互补 PWM 模式，将定时器输出控制寄存器 (TOCR) 的 PSYE 位置 1，可与 PWM 载波周期同步交替输出，交替输出的波形例如图 11.54。

通过 TCNT_3 和 TGRA_3 的比较匹配及 TCNT4 和 H'0000 的比较匹配执行交替输出。

此交替输出的输出引脚为 TIOC3A 引脚，初始输出为 1。

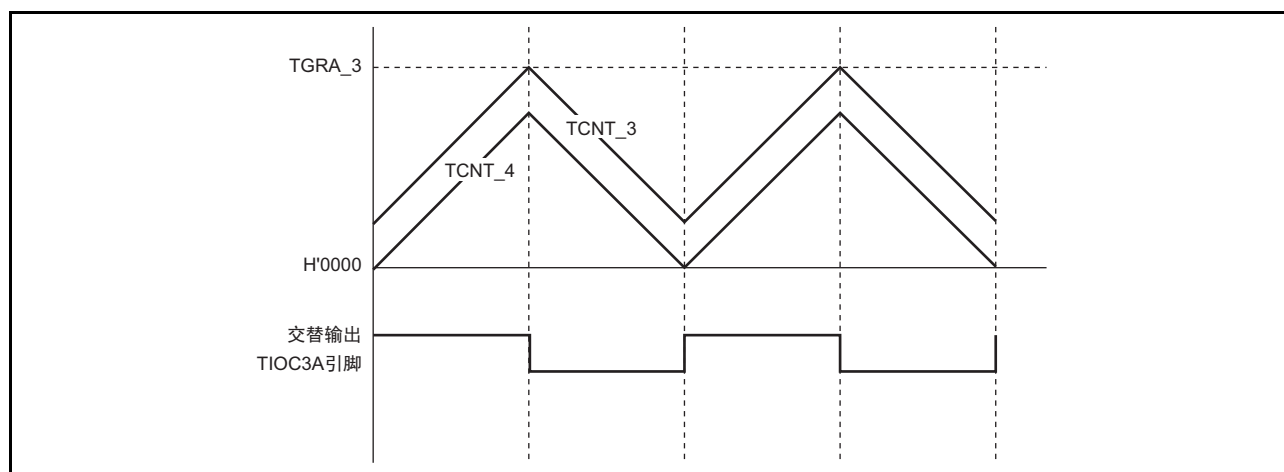


图 11.54 与 PWM 输出同步的交替输出波形例

(m) 通过其他通道清除计数器

在互补 PWM 模式，由定时器同步寄存器 (TSYR) 设定为与其他通道的同步模式，并由定时器控制寄存器 (TCR) 的 CCLR2 ~ CCLR0 位选择同步清除，可通过其他通道清除 TCNT_3、TCNT_4 和 TCNTS。

运行例如图 11.55 所示。

使用此功能，可由外部信号清除计数器并重新开始计数。

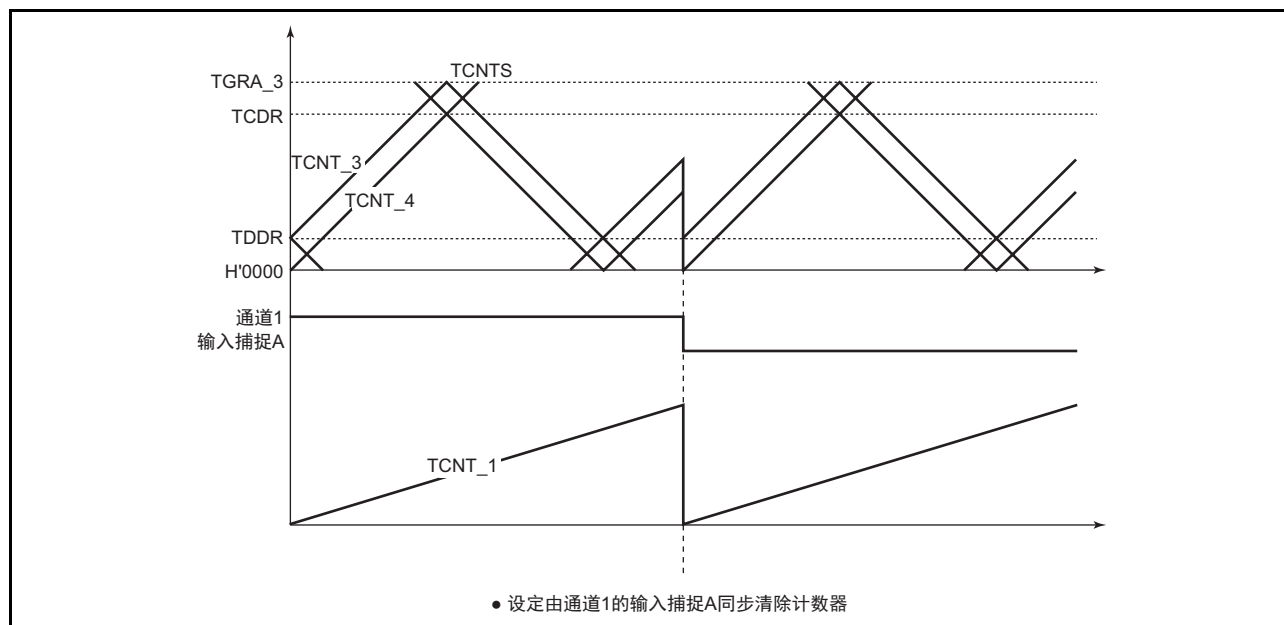


图 11.55 清除与其他通道同步的计数器

(n) 在互补 PWM 模式同步清除计数器时控制输出波形

将 TWCR 寄存器的 WRE 位设定为 1，可抑制在互补 PWM 模式的波谷 Tb 区间同步清除计数器时的初始输出。由此可抑制同步清除计数器时的占空比的急剧变化。

如图 11.56 的 (10)、(11) 所示，仅在波谷 Tb 区间执行同步清除时，才可通过将 WRE 位设定为 1，抑制初始输出。在其他时序产生同步清除时，可输出 TOCR 寄存器的 OLS 位设定的初始值。即使在波谷 Tb 区间，如果在图 11.56 的 (1) 所示的计数器开始后的初始输出期间产生同步清除，也不抑制初始值输出。

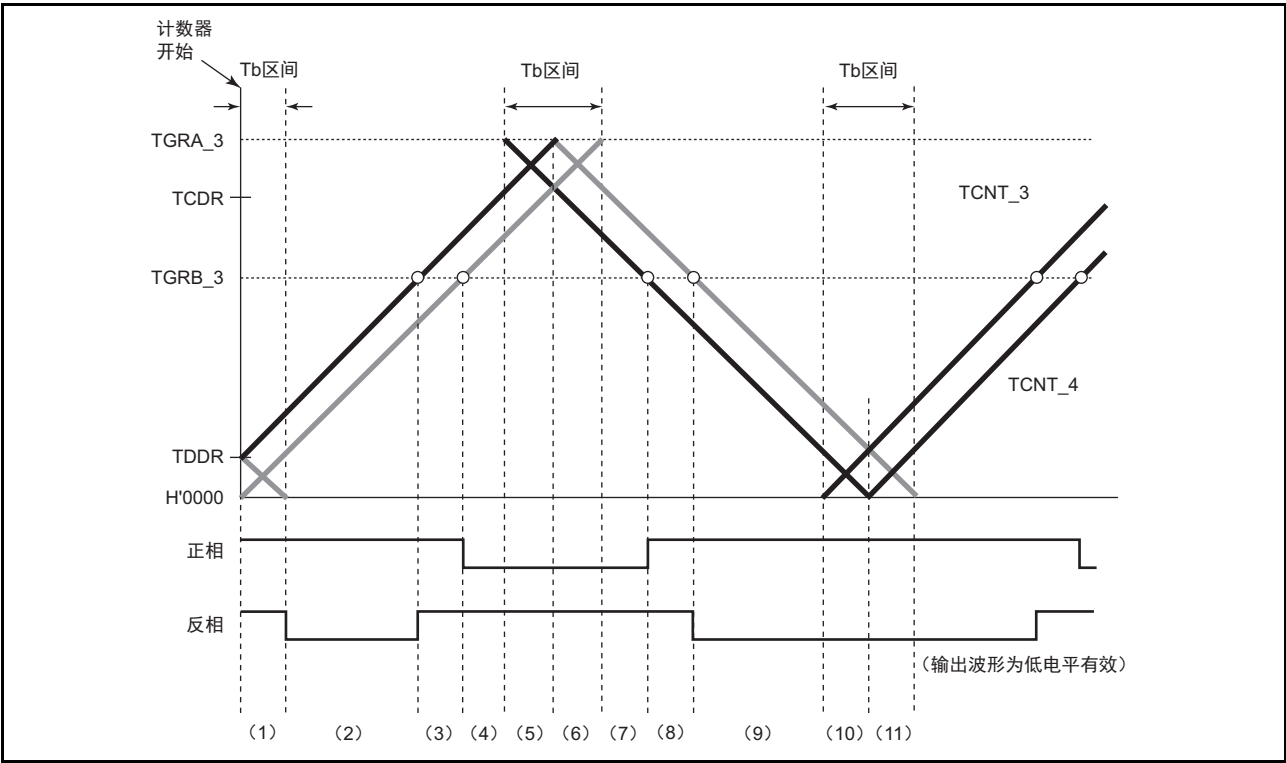


图 11.56 同步清除计数器的时序

- 在互补PWM模式同步清除计数器时控制输出波形的设定步骤例
在互补 PWM 模式，同步清除计数器时，控制输出波形的设定步骤例如图 11.57 所示。

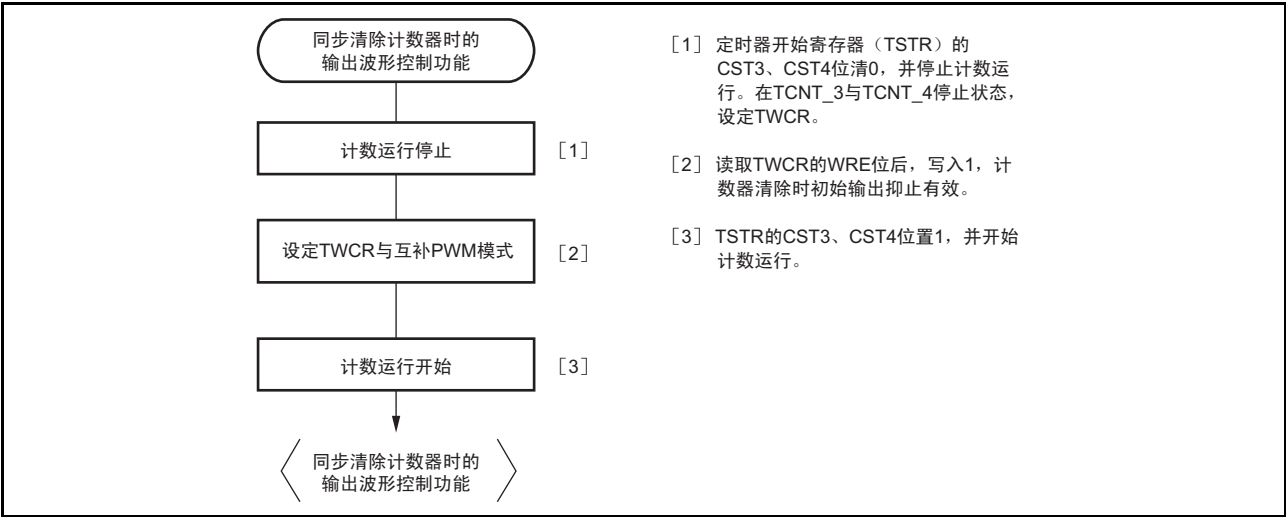


图 11.57 在互补 PWM 模式同步清除计数器时控制输出波形的设定步骤例

- 在互补 PWM 模式同步清除计数器时控制输出波形的运行例

在 TWCR 的 WRE 位设定为 1 的状态, 使 MTU2 以互补 PWM 模式运行, 同步清除计数器时的运行例如图 11.58 ~ 图 11.61 所示。图 11.58 ~ 图 11.61 所示的同步清除计数器的时序分别为图 11.56 的 (3)、(6)、(8)、(11) 所示的时序。

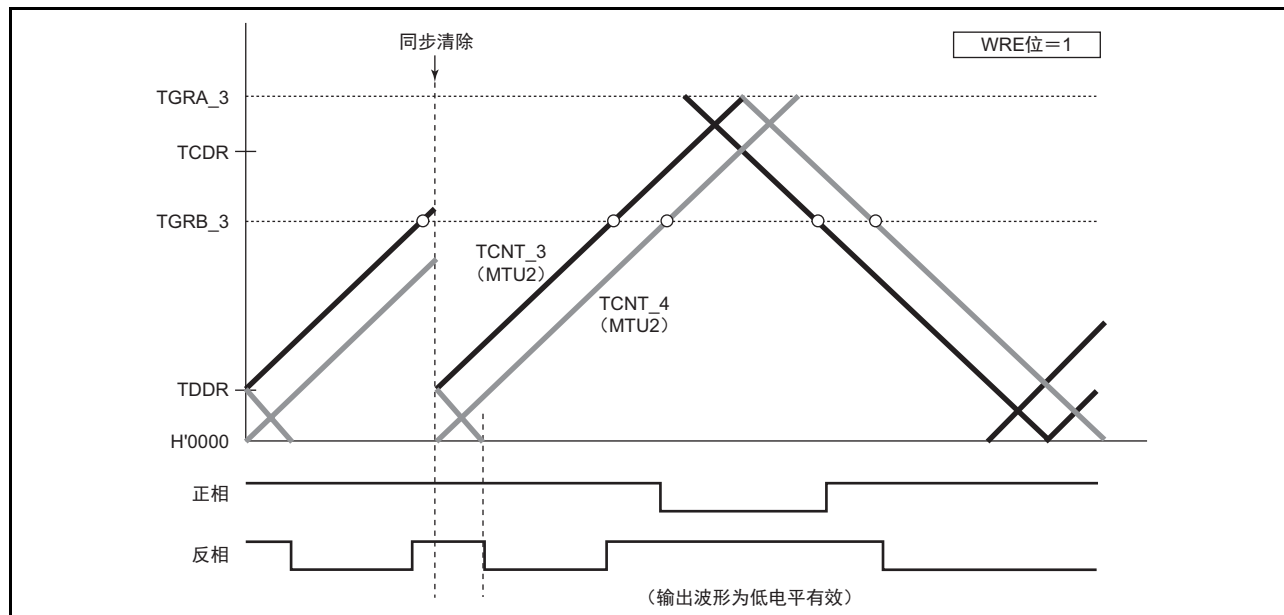


图 11.58 在递增计数的空载时间产生同步清除时
(图 11.56 的时序 (3)、MTU2 的 TWCR 寄存器的 WRE 位 =1)

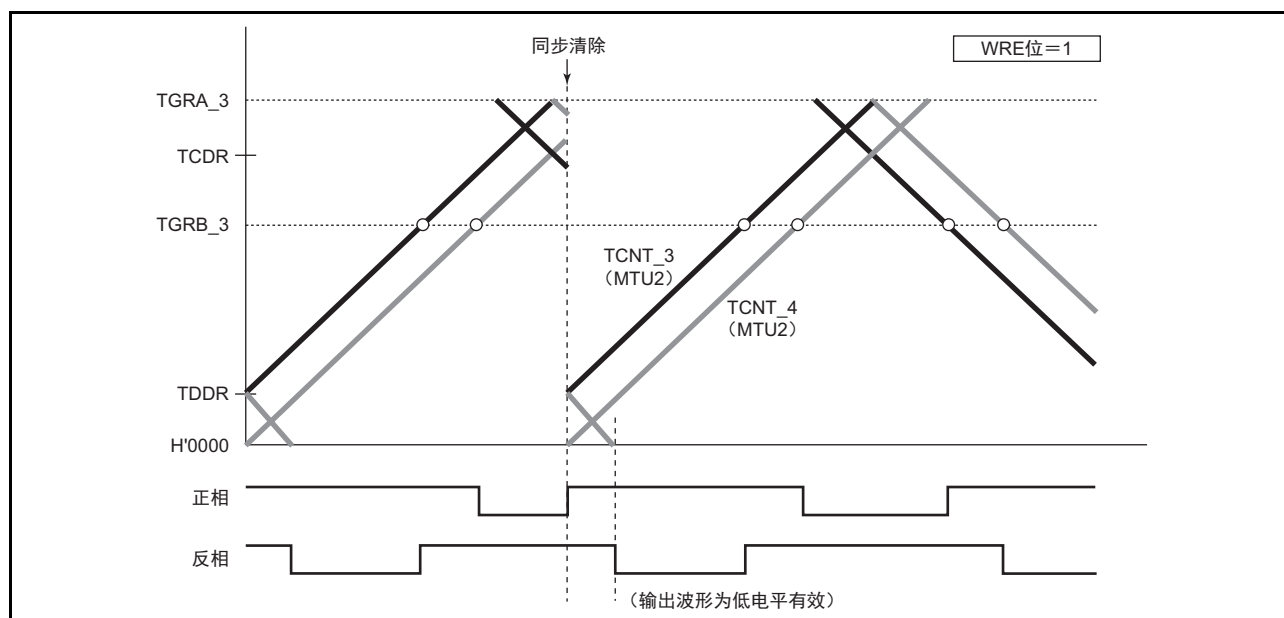


图 11.59 在波峰 Tb 区间产生同步清除时
(图 11.56 的时序 (6)、MTU2 的 TWCR 寄存器的 WRE 位 =1)

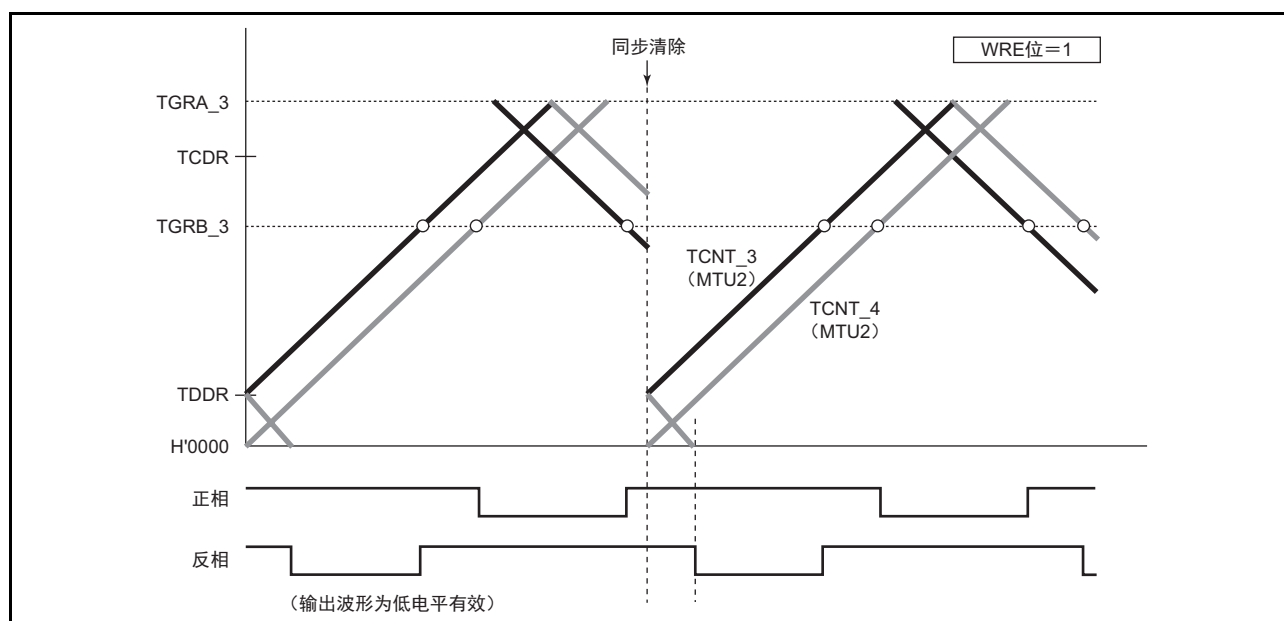


图 11.60 在递减计数的空载时间产生同步清除时
(图 11.56 的时序 (8)、TWCR 寄存器的 WRE 位 =1)

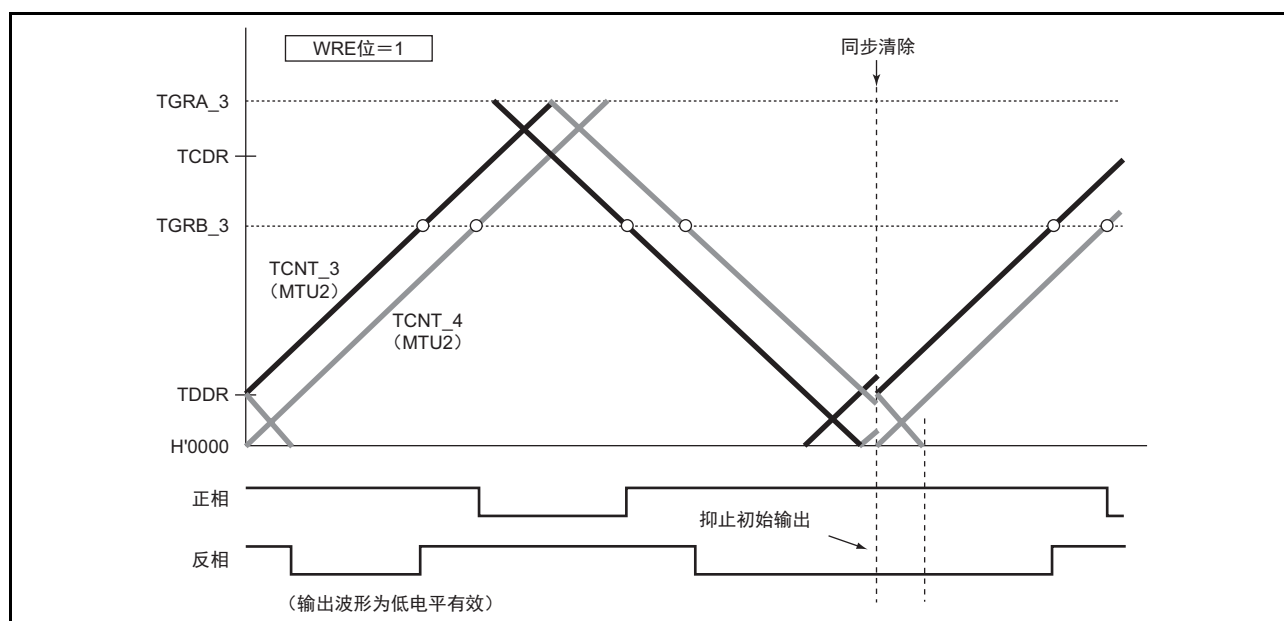


图 11.61 在波谷 Tb 区间产生同步清除时
(图 11.56 的时序 (11)、TWCR 寄存器的 WRE 位 =1)

(o) 由 TGRA_3 比较匹配清除计数器

在互补 PWM 模式，通过设定定时器波形控制寄存器 (TWCR) 的 CCE 位，可由 TGRA_3 的比较匹配，清除 TCNT_3、TCNT_4 和 TCNTS。

运行例如图 11.62 所示。

- 【注】
1. 仅在互补 PWM 模式 1 (在波峰传送) 使用。
 2. 不得设定为与其他通道同步清除的功能 (不得将定时器同步寄存器 (TSYR) 的 SYNC0 ~ SYNC4 位和定时器同步清除寄存器 (TSYCR) 的 CE0A/B/C/D、CE1A/B/C/D 位设定为 1)。
 3. PWM 占空比不得设定为 H'0000。
 4. 不得将定时器输出控制寄存器 1 (TOCR1) 的 PSYE 位设定为 1。

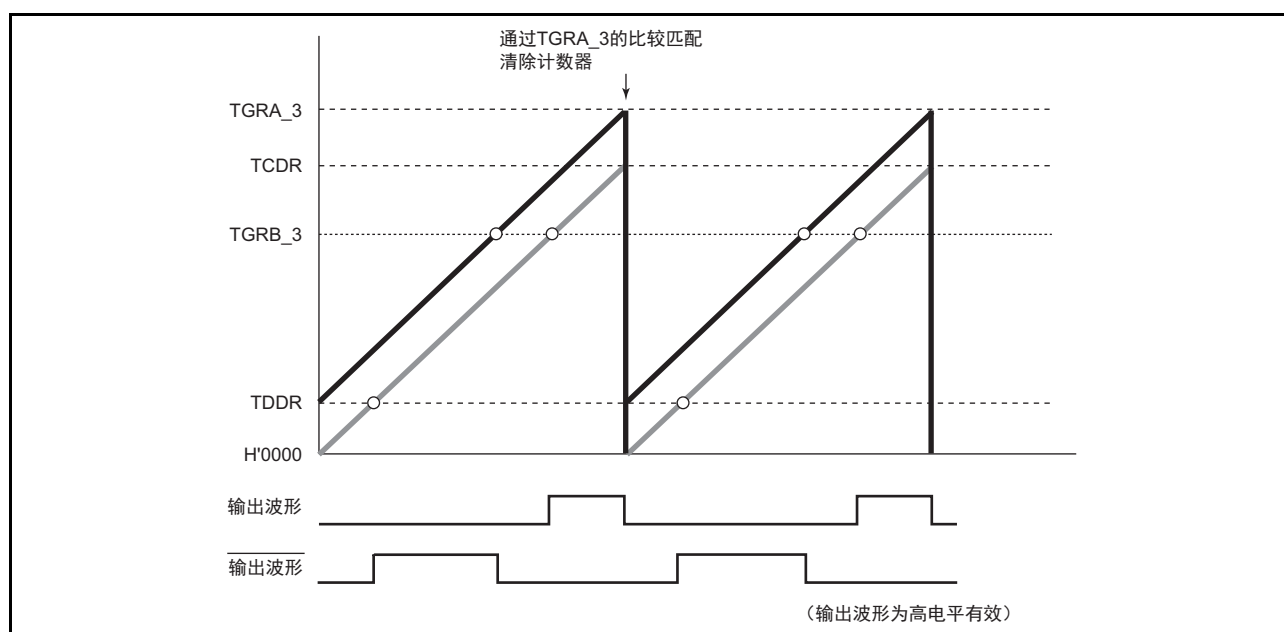


图 11.62 由 TGRA_3 的比较匹配清除计数器的运行例

(p) AC 同步马达 (无刷 DC 马达) 的驱动波形输出例

在互补 PWM 模式，使用定时器门控寄存器 (TGCR)，可轻松控制无刷 DC 马达。使用 TGCR 的无刷 DC 马达的驱动波形例如图 11.63 ~ 图 11.66 所示。

转换 3 相无刷 DC 马达的输出相时，使用霍尔元件等检测外部信号后，将 TGCR 的 FB 位设定为 0。此时，将表示磁极位置的外部信号输入通道 0 的定时器输入引脚 TI0C0A、TI0C0B、TI0C0C (必须由 PFC 设定)。如果 TI0C0A、TI0C0B、TI0C0C 等 3 个引脚产生边沿，则输出的 ON/OFF 自动转换。

FB 位为 1 时，如果将 TGCR 的 UF、VF、WF 各位设定为 0 或 1，则转换输出的 ON/OFF。

从互补 PWM 模式的 6 相输出引脚输出驱动波形。

通过将 N 位或 P 位设定为 1，在 ON 输出时，此 6 相输出使用互补 PWM 模式的输出，并可进行斩波输出。N 位或 P 位为 0 时，为电平输出。

6 相输出的有效电平 (ON 输出时的电平) 与 N 位和 P 位的设定无关，可由定时器输出控制寄存器 (TOCR) 的 OLSN 位和 OLSP 位设定。

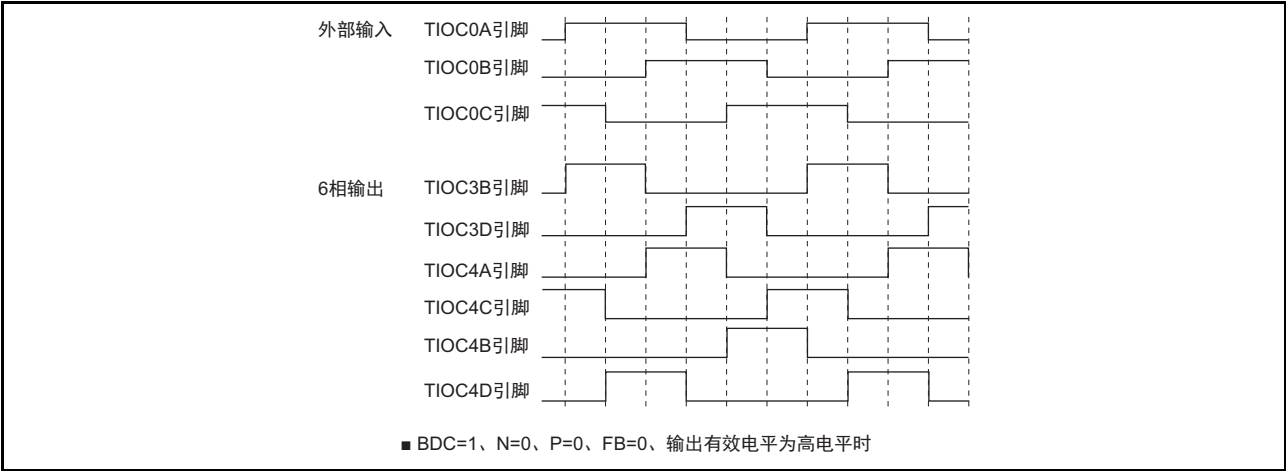


图 11.63 由外部输入转换输出相的运行例 (1)

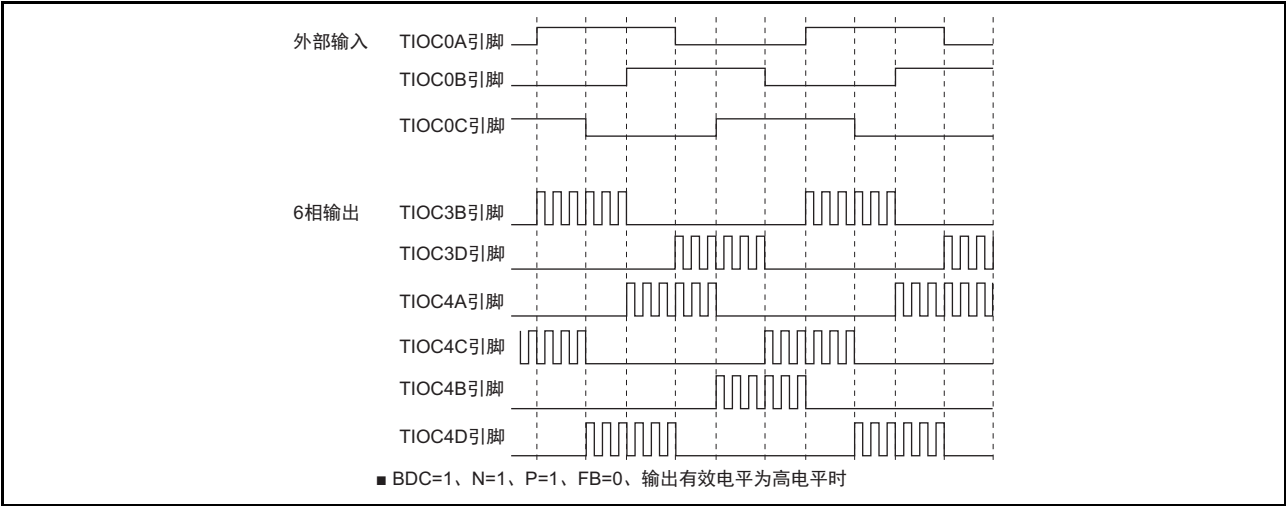


图 11.64 由外部输入转换输出相的运行例 (2)

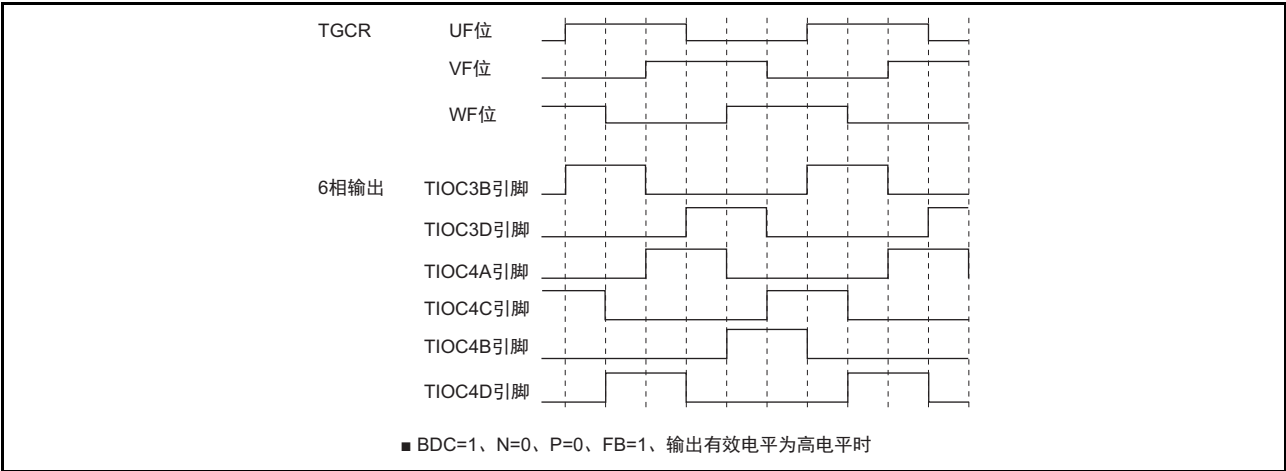


图 11.65 通过设定 UF、VF、WF 位转换输出相的运行例 (1)

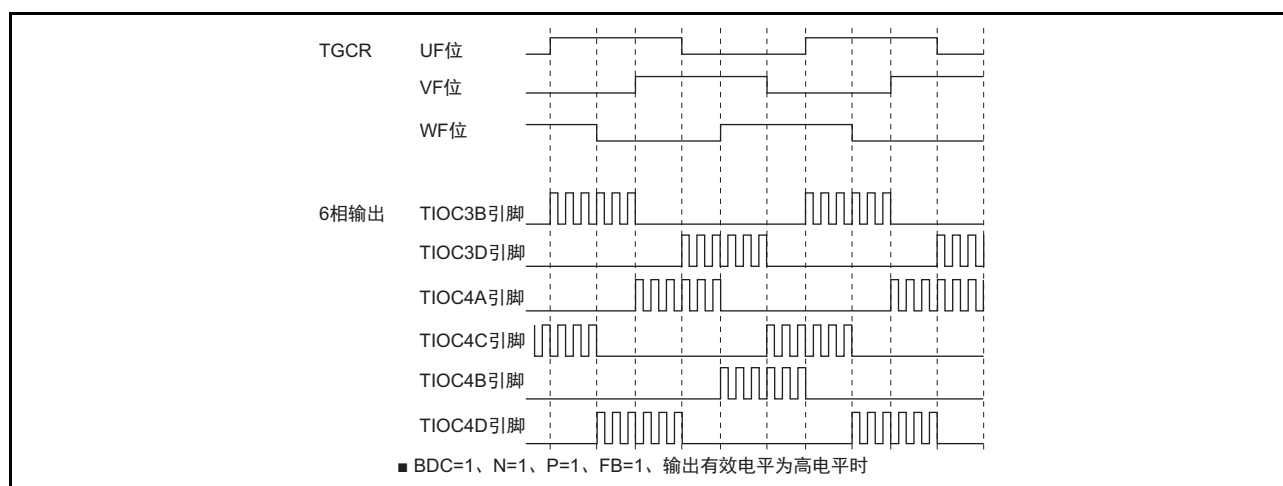


图 11.66 通过设定 UF、VF、WF 位转换输出相的运行例 (2)

(q) A/D 转换开始请求的设定

在互补 PWM 模式，可使用 TGRA_3 的比较匹配、TCNT_4 的下溢（波谷）或除通道 3、4 之外的通道的比较匹配，执行 A/D 转换开始请求。

使用 TGRA_3 的比较匹配设定开始请求时，可在 TCNT_3 的波峰开始 A/D 转换。

通过将定时器中断允许寄存器（TIER）的 TTGE 位置 1，可设定 A/D 转换的开始请求；通过将 TIER_4 的 TTGE2 位置 1，可设定 TCNT_4 下溢（波谷）的 A/D 转换开始请求。

(3) 互补 PWM 模式的中断跳过功能

通过设定定时器中断跳过设定寄存器（TITCR），通道 3 和通道 4 的 TGIA_3（波峰的中断）及 TCIV_4（波谷的中断）最多可跳过 7 次中断。

通过设定定时器缓冲传送寄存器（TBTER），可与缓冲寄存器向暂存器 / 比较寄存器的传送联动，跳过中断。有关与缓冲寄存器的联动，详情参阅“(c) 与中断跳过功能联动的缓冲传送控制”。

通过设定定时器 A/D 转换请求控制寄存器（TADCR），可与 A/D 转换开始请求延迟功能的 A/D 转换开始请求联动，跳过中断。有关和 A/D 转换开始请求延迟功能的联动，详情参阅“11.4.9 A/D 转换开始请求延迟功能”。

必须通过设定 TIER_3、TIER_4 寄存器禁止 TGIA_3 和 TCIV_4 中断请求的状态，以及在不产生比较匹配引起的 TGFA_3、TCFV_4 标志置位的状态，设定定时器中断跳过设定寄存器（TITCR）。更改跳过次数前，必须将 T3AEN 和 T4VEN 位置 0，清除跳过计数器。

(a) 中断跳过功能的设定步骤例

中断跳过功能的设定步骤例如图 11.67 所示，中断跳过次数的可更改期间如图 11.68 所示。

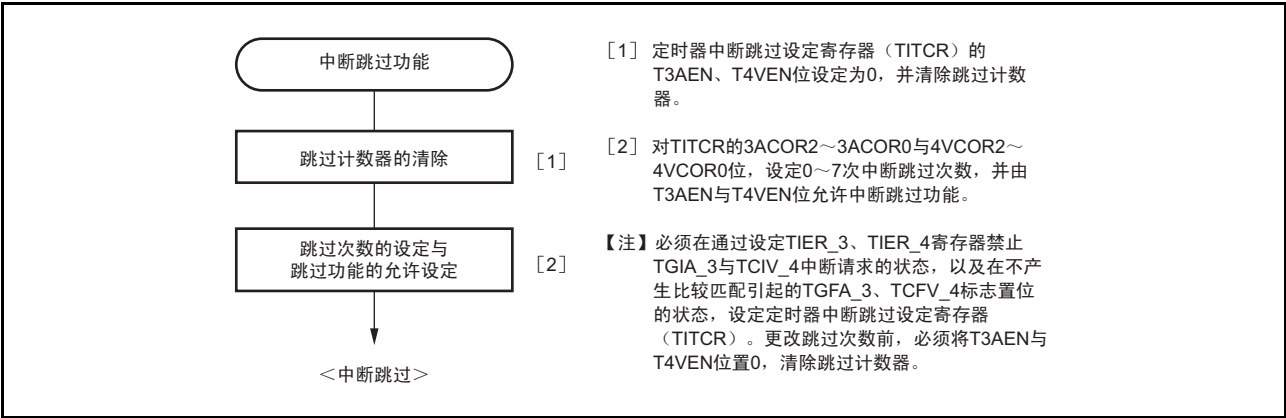


图 11.67 中断跳过功能的设定步骤例

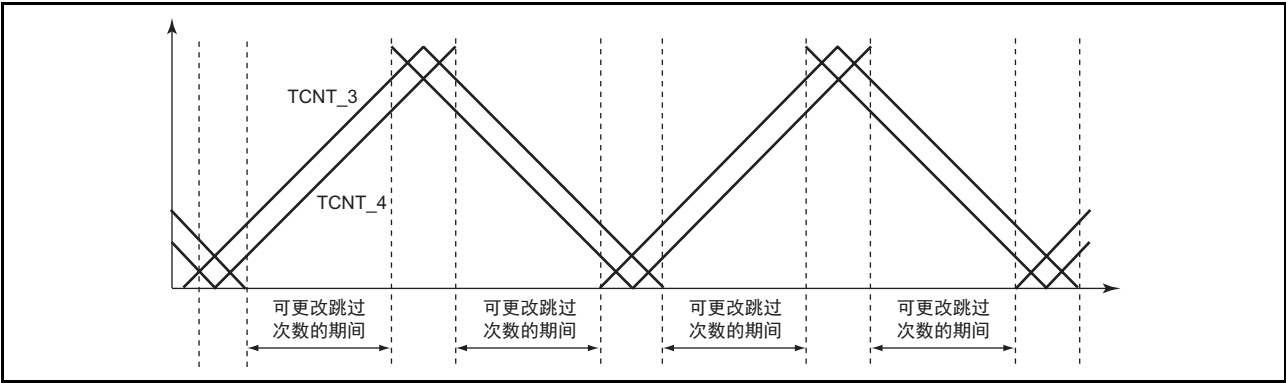


图 11.68 中断跳过次数的可更改期间

(b) 中断跳过功能的运行例

通过定时器中断跳过设定寄存器 (TITCR) 的 3ACOR 位，将中断跳过次数设定为 3 次，并将 T3AEN 位设定为 1 时，TGIA_3 中断跳过功能的运行例如图 11.69 所示。

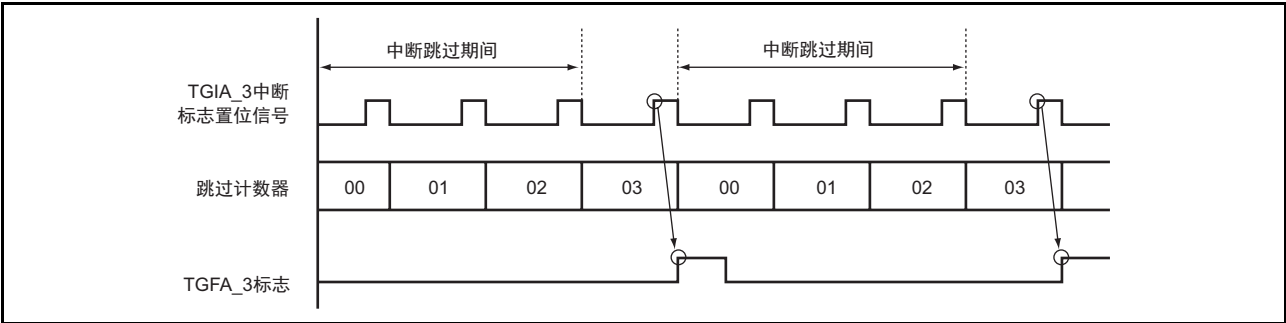


图 11.69 中断跳过功能的运行例

(c) 与中断跳过功能联动的缓冲传送控制

通过设定定时器缓冲传送设定寄存器 (TBTER) 的 BTE1、BTE0 位, 可选择在互补 PWM 模式是否从缓冲寄存器向暂寄存器缓冲传送, 或是否与中断跳过联动。

设定为抑制缓冲传送 (BTE1=0、BTE0=1) 时的运行例如图 11.70 所示。在此设定期间, 不从缓冲寄存器向暂寄存器传送。

设定缓冲传送与中断跳过功能 (BTE1=1、BTE0=0) 联动的运行例如图 11.71 所示。此设定时, 在缓冲传送允许期间之外, 不从缓冲寄存器向暂寄存器传送。

根据产生中断的缓冲寄存器改写时序不同, 有从缓冲寄存器向暂寄存器的传送时序, 及从暂时寄存器向通用寄存器的传送时序这 2 种。

将定时器中断跳过设定寄存器 (TITCR) 的 T3AEN 位、T4VEN 位、T3AEN/T4VEN 位设定为 1 时, 各缓冲传送允许期间各不相同。TITCR 的 T3AEN、T4VEN 位的设定与缓冲传送允许期间的关系如图 11.72 所示。

【注】 本功能必须与中断跳过功能组合使用。

禁止中断跳过时 (定时器中断跳过设定寄存器 (TITCR) 的 T3AEN、T4VEN 位或 TITCR 的跳过次数设定位 (3ACOR、4VCOR) 设定为 0 时), 必须设定为缓冲传送不与中断跳过联动 (定时器缓冲传送设定寄存器 (TBTER) 的 BTE1 位设定为 0)。禁止中断跳过时, 如果设定缓冲传送与中断跳过联动, 则不执行缓冲传送。

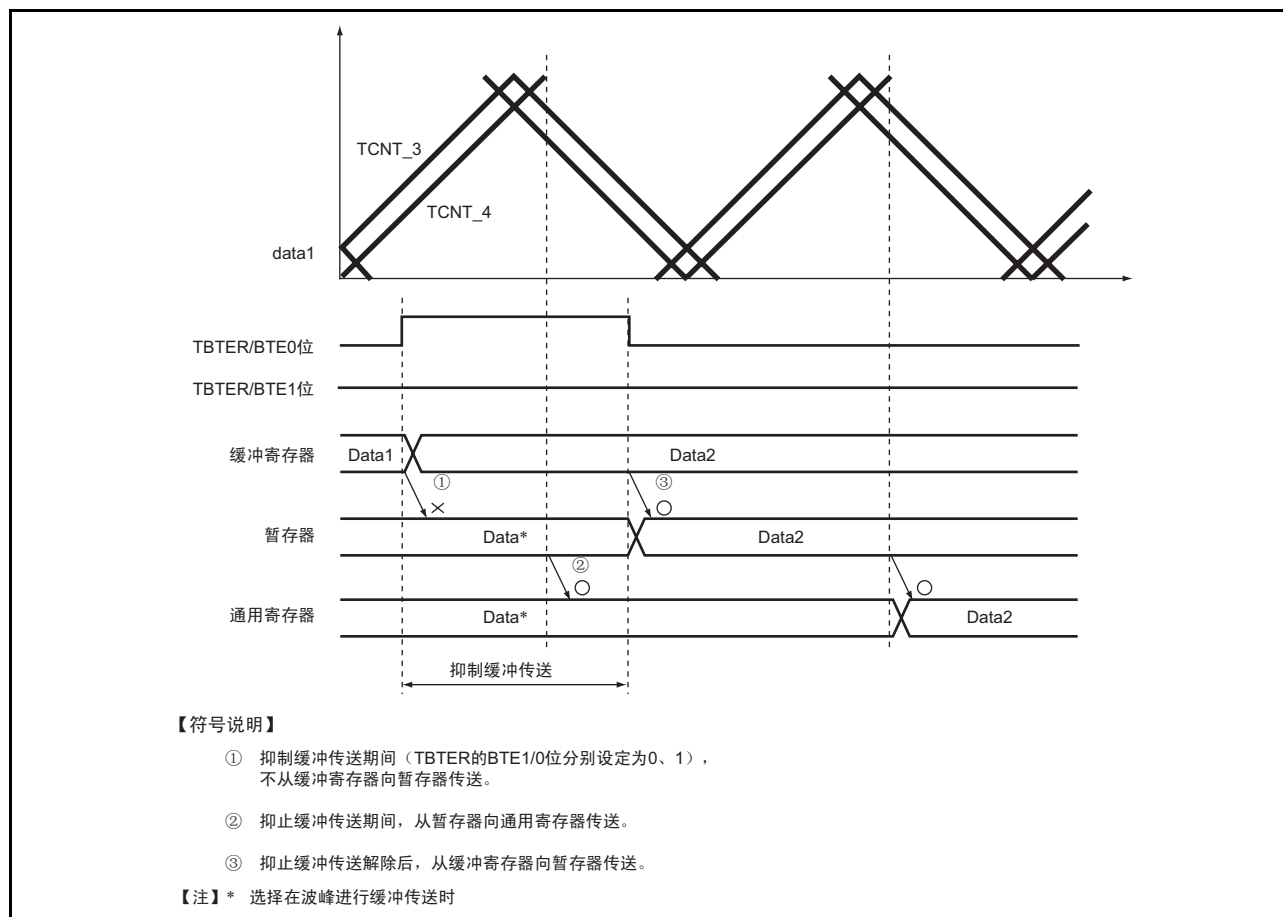
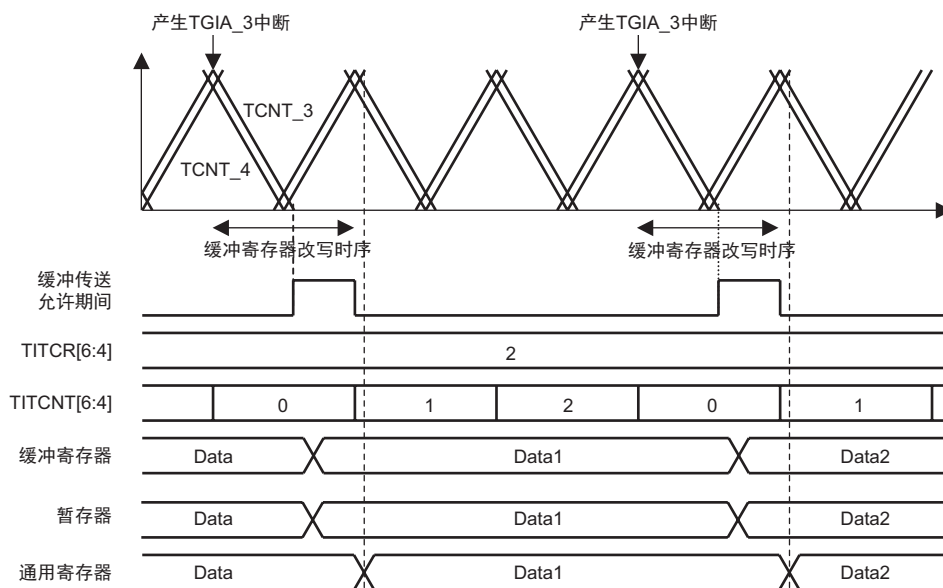
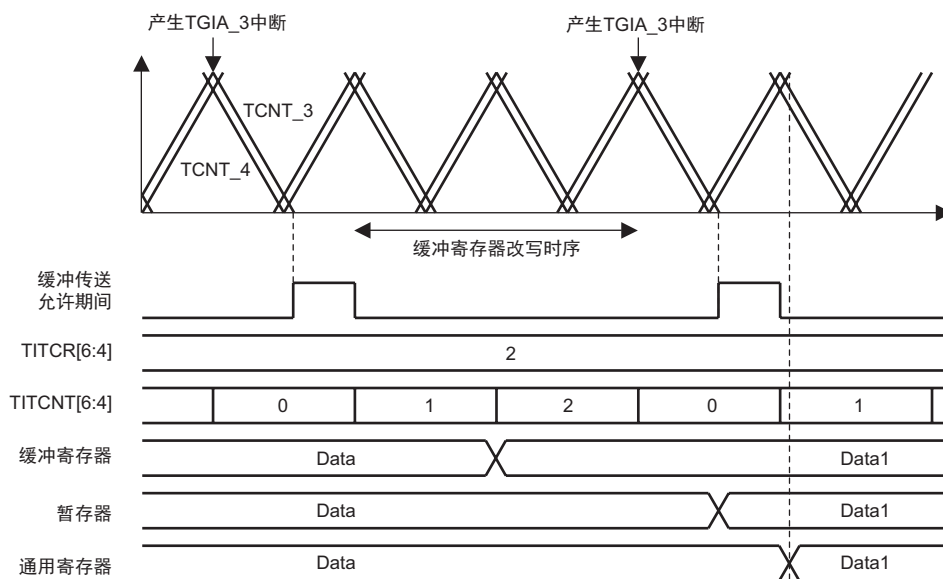


图 11.70 设定为抑制缓冲传送 (BTE1=0、BTE0=1) 时的运行例

(1) 在从TGIA_3中断开始1个载波周期内, 改写缓冲寄存器时



(2) 在从TGIA_3中断开始经过1个载波周期后, 改写缓冲寄存器时



【注】 TMDR_3的MD[3:0]=1101, 选择在波峰进行缓冲传送。
 跳过次数设定为2次。
 T3AEN设定为1, T4VEN设定为0。

图 11.71 设定为缓冲传送与中断跳过联动 (BTE1=1、BTE0=0) 时的运行例

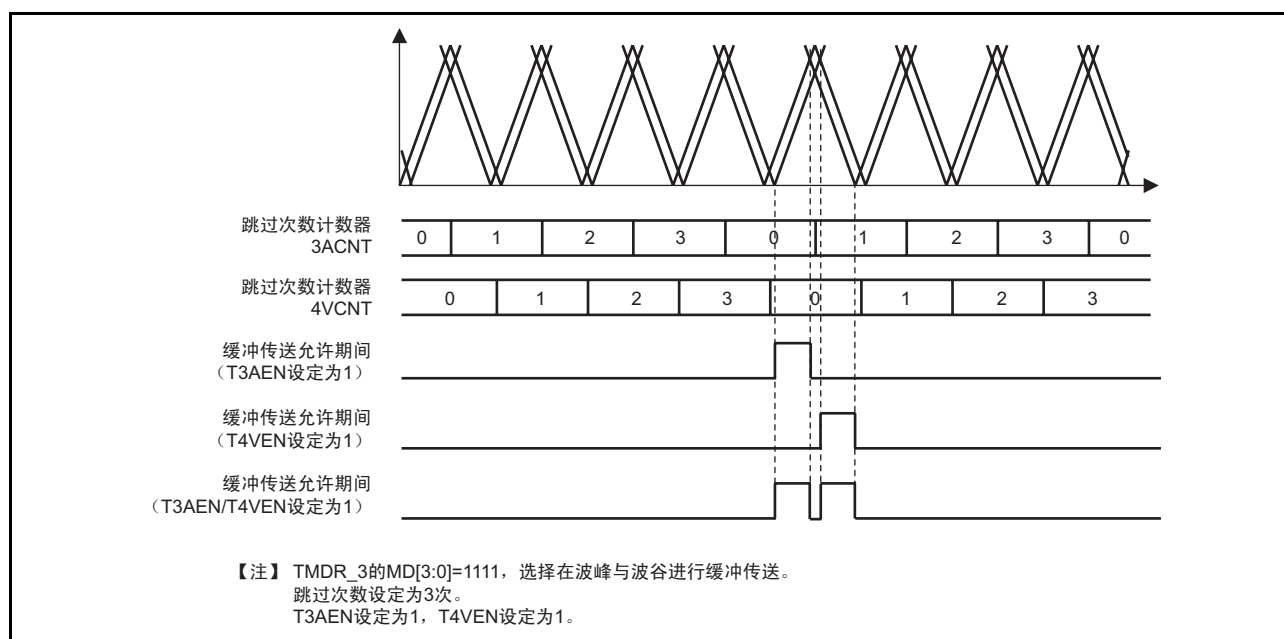


图 11.72 定时器中断跳过设定寄存器（TITCR）的 T3AEN、T4VEN 位的设定和缓冲传送允许周期的关系

(4) 互补 PWM 模式的输出保护功能

互补 PWM 模式的输出有以下保护功能:

(a) 寄存器、计数器误写防止功能

在互补 PWM 模式使用的寄存器、计数器中，除总是执行改写的缓冲寄存器之外，模式寄存器、控制寄存器、比较寄存器及计数器通过设定定时器读取/写入允许寄存器（TRWER）的 RWE 位，可选择允许/禁止由 CPU 存取。对象寄存器为通道 3 和通道 4 的部分寄存器，适用的寄存器如下：

TCR_3和TCR_4、TMDR_3和TMDR_4、TIO RH_3和TIO RH_4、TIO RL_3和TIO RL_4、TIER_3和TIER_4、TCNT_3和TCNT_4、TGRA_3和TGRA_4、TGRB_3和TGRB_4、TOER、TOCR、TGCR、TCDR、TDDR 共 21 个寄存器

由此功能通过将模式寄存器、控制寄存器和计数器等设定为禁止由 CPU 存取，可防止因 CPU 失控而导致的误写。在禁止存取状态，读取对象寄存器时，读取不定值，写入无效。

11.4.9 A/D 转换开始请求延迟功能

通过设定通道 4 的定时器 A/D 转换开始请求控制寄存器 (TADCR)、定时器 A/D 启动请求周期寄存器 (TADCORA_4、TADCORB_4) 和定时器 A/D 启动请求周期缓冲寄存器 (TADCOBRA_4、TADCOBRB_4)，可请求开始 A/D 转换。

A/D 转换开始请求延迟功能是比较 TCNT_4 和 TADCORA_4、TADCORB_4，匹配时，执行各自的 A/D 转换开始请求 (TRG4AN、TRG4BN)。

通过设定 TADCR 的 ITA3AE、ITA4VE、ITB3AE 和 ITB4VE 位，可与中断跳过功能联动，跳过 A/D 转换开始请求 (TRG4AN、TRG4BN)。

• A/D 转换开始请求延迟功能的设定步骤例

A/D 转换开始请求延迟功能的设定步骤例如图 11.73 所示。

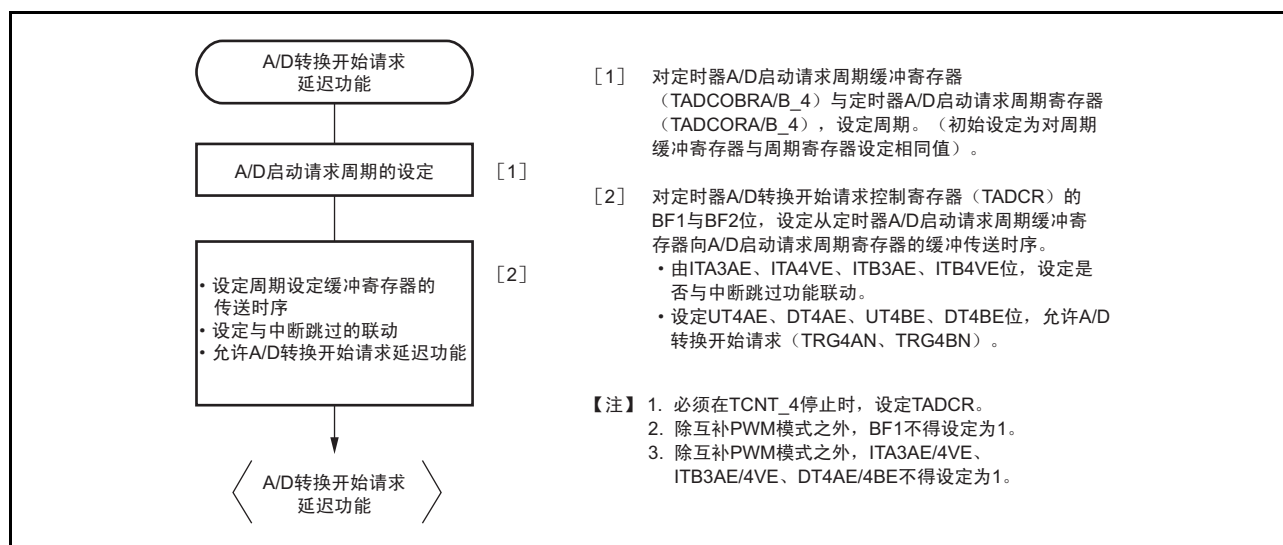


图 11.73 A/D 转换开始请求延迟功能的设定步骤例

- A/D 转换开始请求延迟功能的基本运行例

将缓冲传送时序设定为 TCNT_4 的波谷，并设定在 TCNT_4 递减计数时输出 A/D 转换开始请求信号 (TRG4AN) 时，A/D 转换开始请求信号 (TRG4AN) 的基本运行例如图 11.74 所示。

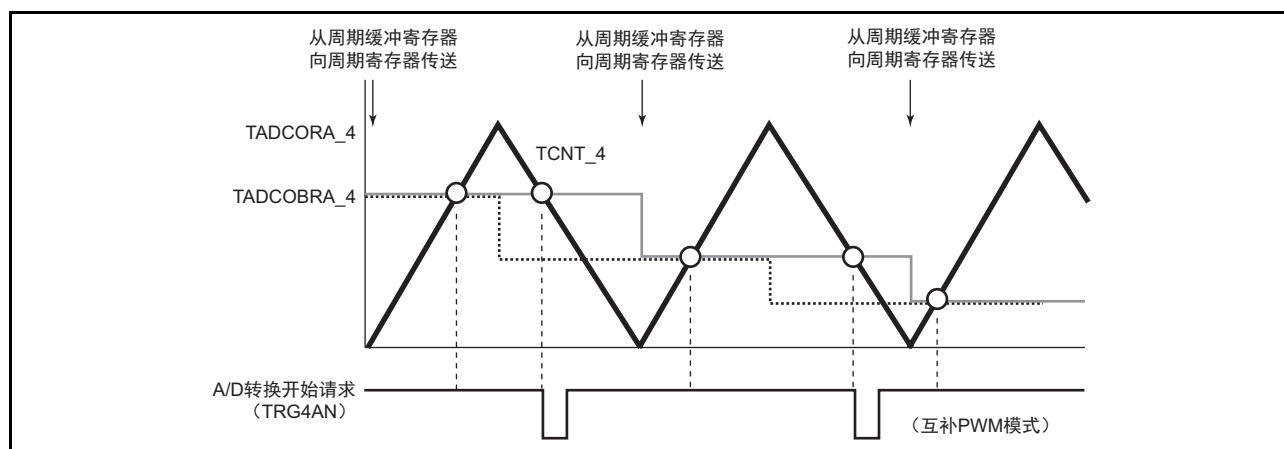


图 11.74 A/D 转换开始请求信号 (TRG4AN) 的基本运行例

- 缓冲传送

通过对定时器 A/D 启动请求周期设定缓冲寄存器 (TADCOBRA/B_4) 写入数据，更新定时器 A/D 启动请求周期设定寄存器 (TADCORA/B_4) 的数据。通过设定定时器 A/D 转换开始请求控制寄存器 (TADCR_4) 的 BF1、BF0 位，可选择从定时器 A/D 启动请求周期设定缓冲寄存器向定时器 A/D 启动请求周期设定寄存器的传送时序。

- 与中断跳过功能联动的 A/D 转换开始请求延迟功能

通过设定定时器 A/D 转换开始请求控制寄存器 (TADCR) 的 UT4AE、DT4AE、UT4BE 和 DT4BE 位，可与中断跳过功能联动，执行 A/D 转换的开始请求 (TRG4AN、TRG4BN)。设定在 TCNT_4 递增计数及递减计数时允许 TRG4AN 输出，并与中断跳过功能联动时的 A/D 转换开始请求信号 (TRG4AN) 的运行例如图 11.75 所示。

设定在 TCNT_4 递增计数时允许 TRG4AN 输出，并与中断跳过功能联动时的 A/D 转换开始请求信号 (TRG4AN) 的运行例如图 11.76 所示。

【注】 本功能必须与中断跳过功能组合使用。

禁止中断跳过时 (定时器中断跳过设定寄存器 (TITCR) 的 T3AEN、T4VEN 位或 TITCR 的跳过次数设定 (3ACOR、4VCOR) 设定为 0 时)，必须设定为不与中断跳过功能联动 (定时器 A/D 转换开始请求控制寄存器 (TADCR) 的 ITA3AE、ITA4VE、ITB3AE 和 ITB4VE 位设定为 0)。

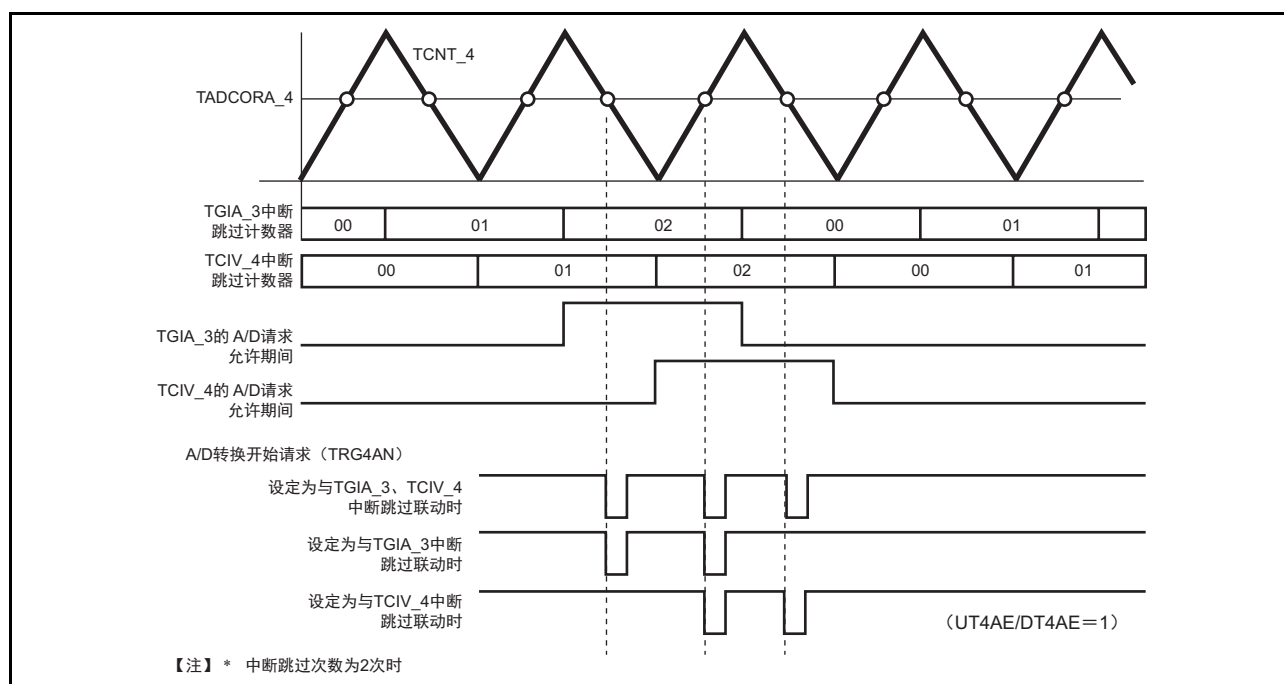


图 11.75 与中断跳过功能联动时的 A/D 转换开始请求信号 (TRG4AN) 的运行例

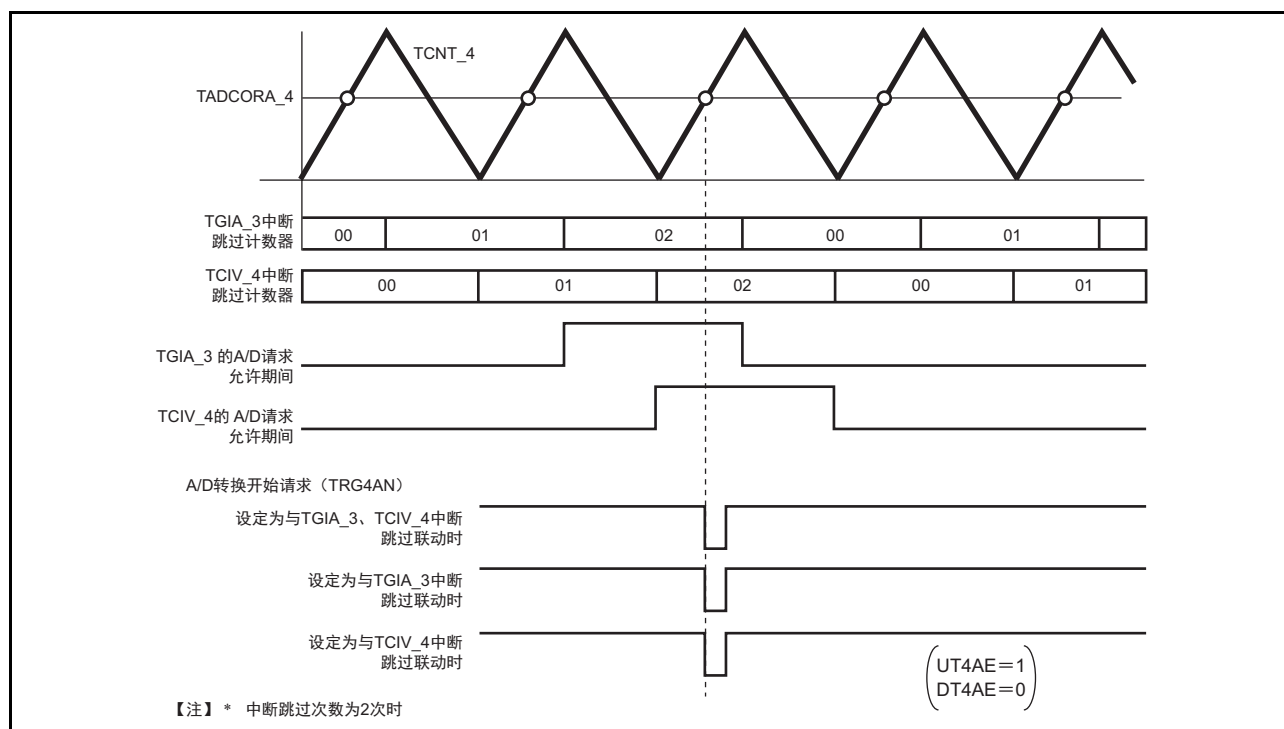


图 11.76 与中断跳过功能联动时的 A/D 转换开始请求信号 (TRG4AN) 的运行例

11.4.10 互补 PWM 的“波峰 / 波谷”的 TCNT 捕捉运行

互补 PWM 运行时，在“波峰、波谷、波峰 / 波谷”将 TCNT 的值保存至 TGR。由 TIOR 选择捕捉到 TGR 的时序转换。

图 11.77 所示为 TCNT 自由运行，且不被清除直接使用，并在设定的“波峰、波谷”捕捉到 TGR 的运行例。

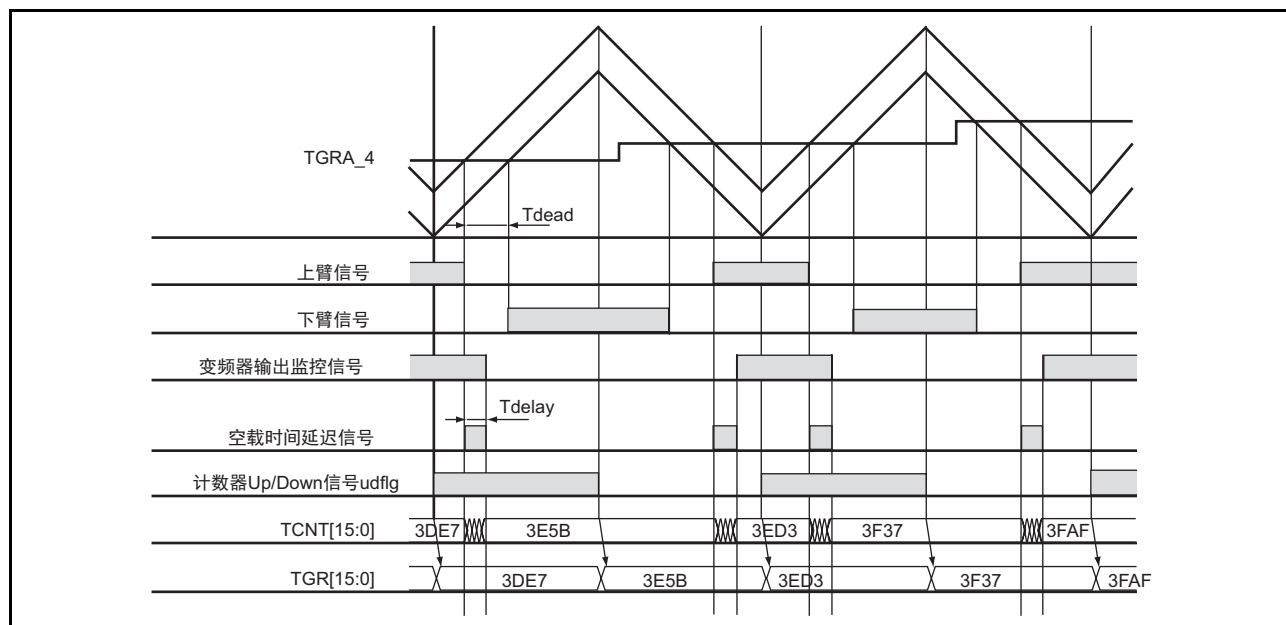


图 11.77 互补 PWM 的“波峰 / 波谷”的 TCNT 捕捉运行

11.5 中断源

11.5.1 中断源和优先顺序

MTU2 的中断源有 TGR 的输入捕捉 / 比较匹配、TCNT 的上溢和下溢 3 种。各中断源有各自专用的状态标志和允许 / 禁止位，因此可独立允许或禁止产生中断请求信号。

产生中断源时，TSR 对应的状态标志置 1。此时，如果 TIER 对应的允许 / 禁止位置 1，则请求中断。将状态标志清 0，可解除中断请求。

通道之间的优先顺序可由中断控制器更改，通道内的优先顺序是固定的。详情参阅“第 6 章 中断控制器 (INTC)”。

MTU2 的中断源一览表如表 11.55 所示。

表 11.55 MTU2 中断源

通道	名称	中断源	中断标志	DMAC 的启动	优先顺序
0	TGIA_0	TGRA_0 的输入捕捉 / 比较匹配	TGFA_0	可	高
	TGIB_0	TGRB_0 的输入捕捉 / 比较匹配	TGFB_0	不可	
	TGIC_0	TGRC_0 的输入捕捉 / 比较匹配	TGFC_0	不可	
	TGID_0	TGRD_0 的输入捕捉 / 比较匹配	TGFD_0	不可	
	TCIV_0	TCNT_0 的上溢	TCFV_0	不可	
	TGIE_0	TGRE_0 的比较匹配	TGFE_0	不可	
	TGIF_0	TGRF_0 的比较匹配	TGFF_0	不可	
1	TGIA_1	TGRA_1 的输入捕捉 / 比较匹配	TGFA_1	可	↑ ↓ 低
	TGIB_1	TGRB_1 的输入捕捉 / 比较匹配	TGFB_1	不可	
	TCIV_1	TCNT_1 的上溢	TCFV_1	不可	
	TCIU_1	TCNT_1 的下溢	TCFU_1	不可	
2	TGIA_2	TGRA_2 的输入捕捉 / 比较匹配	TGFA_2	可	
	TGIB_2	TGRB_2 的输入捕捉 / 比较匹配	TGFB_2	不可	
	TCIV_2	TCNT_2 的上溢	TCFV_2	不可	
	TCIU_2	TCNT_2 的下溢	TCFU_2	不可	
3	TGIA_3	TGRA_3 的输入捕捉 / 比较匹配	TGFA_3	可	
	TGIB_3	TGRB_3 的输入捕捉 / 比较匹配	TGFB_3	不可	
	TGIC_3	TGRC_3 的输入捕捉 / 比较匹配	TGFC_3	不可	
	TGID_3	TGRD_3 的输入捕捉 / 比较匹配	TGFD_3	不可	
	TCIV_3	TCNT_3 的上溢	TCFV_3	不可	
4	TGIA_4	TGRA_4 的输入捕捉 / 比较匹配	TGFA_4	可	
	TGIB_4	TGRB_4 的输入捕捉 / 比较匹配	TGFB_4	不可	
	TGIC_4	TGRC_4 的输入捕捉 / 比较匹配	TGFC_4	不可	
	TGID_4	TGRD_4 的输入捕捉 / 比较匹配	TGFD_4	不可	
	TCIV_4	TCNT_4 的上溢 / 下溢	TCFV_4	不可	

【注】 表示复位后的初始状态，通道之间的优先顺序可由中断控制器更改。

(1) 输入捕捉 / 比较匹配中断

通过产生各通道的 TGR 的输入捕捉 / 比较匹配, 在 TSR 的 TGF 标志置 1 时, 如果 TIER 的 TGIE 位置 1, 则请求中断。将 TGF 标志清 0, 可解除中断请求。MTU2 共有 18 个输入捕捉 / 比较匹配中断, 通道 0 有 6 个, 通道 3、4 各有 4 个, 通道 1、2 各有 2 个。输入捕捉时, 通道 0 的 TGFE_0、TGFF_0 标志不置位。

(2) 上溢中断

通过产生各通道的 TCNT 上溢, 在 TSR 的 TCFV 标志置 1 时, 如果 TIER 的 TCIEV 位置 1, 则请求中断。将 TCFV 标志清 0, 可解除中断请求。MTU2 共有 5 个上溢中断, 各通道有 1 个。

(3) 下溢中断

通过产生各通道的 TCNT 下溢, 在 TSR 的 TCFU 标志置 1 时, 如果 TIER 的 TCIEU 位置 1, 则请求中断。将 TCFU 标志清 0, 可解除中断请求。MTU2 共有 2 个下溢中断, 通道 1、2 各有 1 个。

11.5.2 DMAC 的启动

由各通道的 TGRA 的输入捕捉 / 比较匹配中断, 可启动 DMAC。详情参阅“第 10 章 直接存储器存取控制器 (DMAC)”。

MTU2 共有 5 个 TGRA 寄存器的输入捕捉 / 比较匹配中断可作为 DMAC 的启动源, 通道 0 ~ 4 各有 1 个。

11.5.3 A/D 转换器的启动

在 MTU2 可通过以下 3 种方法启动 A/D 转换器。

各中断源和 A/D 转换开始请求的对应关系如表 11.56 所示。

(1) TGRA 的输入捕捉 / 比较匹配及互补 PWM 模式时在 TCNT_4 波谷启动 A/D 转换器

可由各通道 TGRA 的输入捕捉 / 比较匹配启动 A/D 转换器。在 TIER_4 的 TTGE2 位置 1 的状态, 互补 PWM 运行且 TCNT_4 为波谷 (TCNT_4=H'0000) 时也可启动 A/D 转换器。

在以下条件, 对 A/D 转换器产生 A/D 转换开始请求 TRGAN。

- 通过产生各通道的 TGRA 的输入捕捉 / 比较匹配, TSR 的 TGFA 标志置 1, 且 TIER 的 TTGE 位置 1 时
 - 在 TIER_4 的 TTGE2 位置 1 的状态, 互补 PWM 运行且 TCNT_4 为波谷 (TCNT_4=H'0000) 时
- 此时, 如果在 A/D 转换器选择 MTU2 的转换开始触发 TRGAN, 则开始 A/D 转换。

(2) 由 TCNT_0 和 TGRE_0 的比较匹配启动 A/D 转换器

由通道 0 的 TCNT_0 和 TGRE_0 的比较匹配, 可产生 A/D 转换开始请求 TRG0N, 启动 A/D 转换器。

通过产生通道 0 的 TCNT_0 和 TGRE_0 的比较匹配, 在 TSR2_0 的 TGFE 标志置 1 时, 如果 TIER2_0 的 TTGE2 位置 1, 则对 A/D 转换器产生 A/D 转换开始请求 TRG0N。此时, 如果在 A/D 转换器选择 MTU2 的转换开始触发 TRG0N, 开始 A/D 转换。

(3) 由 A/D 转换开始请求延迟功能启动 A/D

A/D 转换开始请求控制寄存器 (TADCR) 的 TAD4AE、TAD4BE 位置 1 时, 通过 TADCORA、TADCORB 与 TCNT_4 匹配, 可产生 TRG4AN、TRG4BN, 启动 A/D 转换器。详情参阅 “11.4.9 A/D 转换开始请求延迟功能”。

产生 TRG4AN 时, 如果在 A/D 转换器选择 MTU2 的转换开始触发 TRG4AN, 则开始 A/D 转换。产生 TRG4BN 时, 如果在 A/D 转换器选择 MTU2 的转换开始触发 TRG4BN, 则开始 A/D 转换。

表 11.56 各中断源和 A/D 转换开始请求的对应关系

对 象	中断源	A/D 转换开始请求
TGRA_0 和 TCNT_0	输入捕捉 / 比较匹配	TRGAN
TGRA_1 和 TCNT_1		
TGRA_2 和 TCNT_2		
TGRA_3 和 TCNT_3		
TGRA_4 和 TCNT_4		
TCNT_4	互补 PWM 模式时 TCNT_4 的波谷	
TGRE_0 和 TCNT_0	比较匹配	TRG0N
TADCORA 和 TCNT_4		TRG4AN
TADCORB 和 TCNT_4		TRG4BN

11.6 运行时序

11.6.1 输入 / 输出时序

(1) TCNT 的计数时序

内部时钟运行时的 TCNT 计数时序如图 11.78 所示。外部时钟运行（普通模式）时的 TCNT 计数时序如图 11.79 所示，外部时钟运行（相位计数模式）时的 TCNT 计数时序如图 11.80 所示。

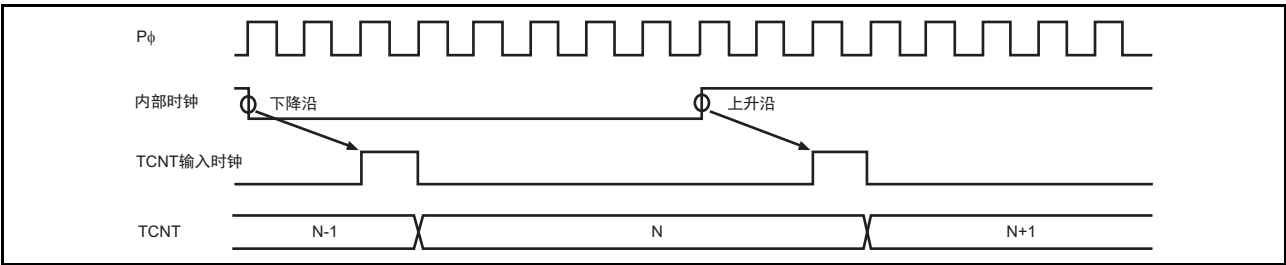


图 11.78 内部时钟运行时的计数时序

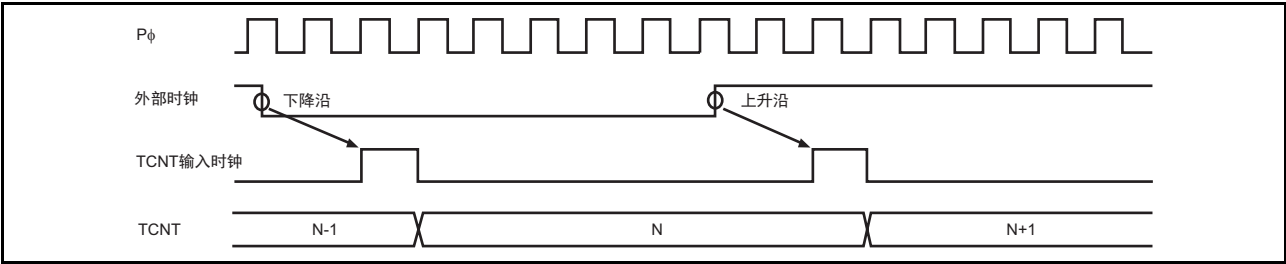


图 11.79 外部时钟运行时的计数时序

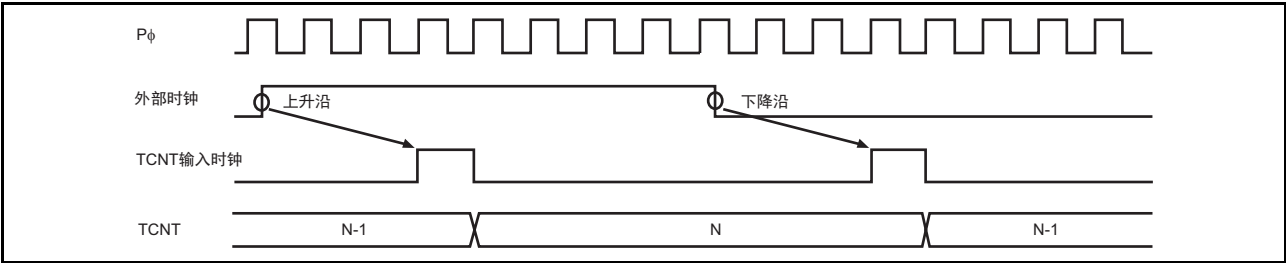


图 11.80 外部时钟运行时的计数时序（相位计数模式）

(2) 输出比较输出时序

在 TCNT 和 TGR 匹配的最后状态（更新 TCNT 匹配后的计数值时）产生比较匹配信号。产生比较匹配信号时，向输出比较输出引脚（TIOC 引脚）输出由 TIOR 设定的输出值。TCNT 与 TGR 匹配后，直到产生 TCNT 输入时钟为止，不产生比较匹配信号。

输出比较输出时序（普通模式、PWM 模式）如图 11.81 所示，输出比较输出时序（互补 PWM 模式、复位同步 PWM 模式）如图 11.82 所示。

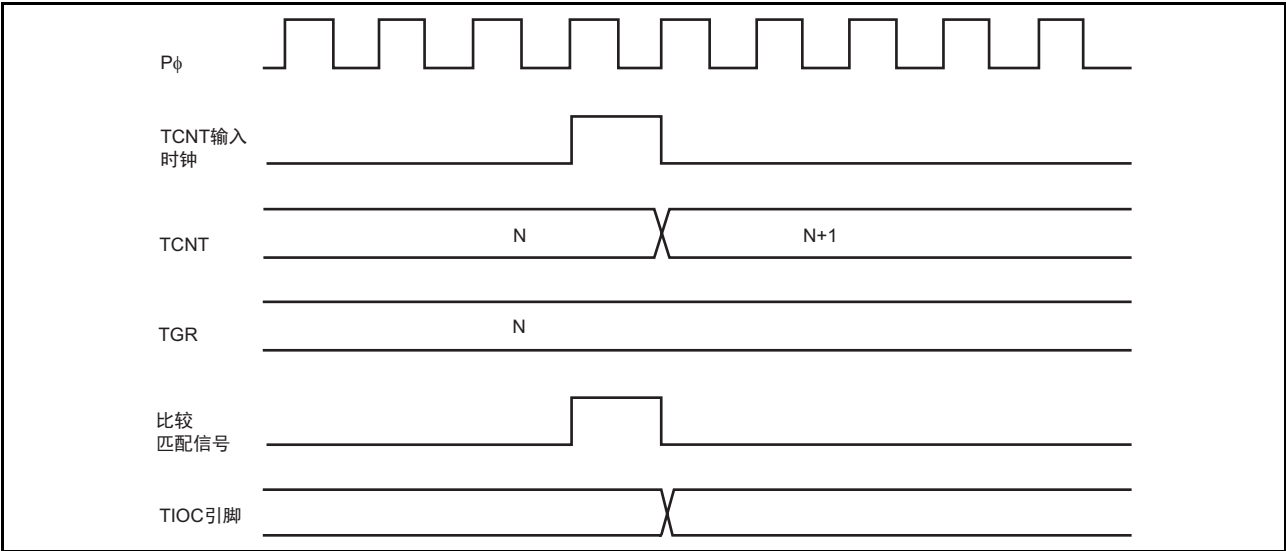


图 11.81 输出比较输出时序（普通模式、PWM 模式）

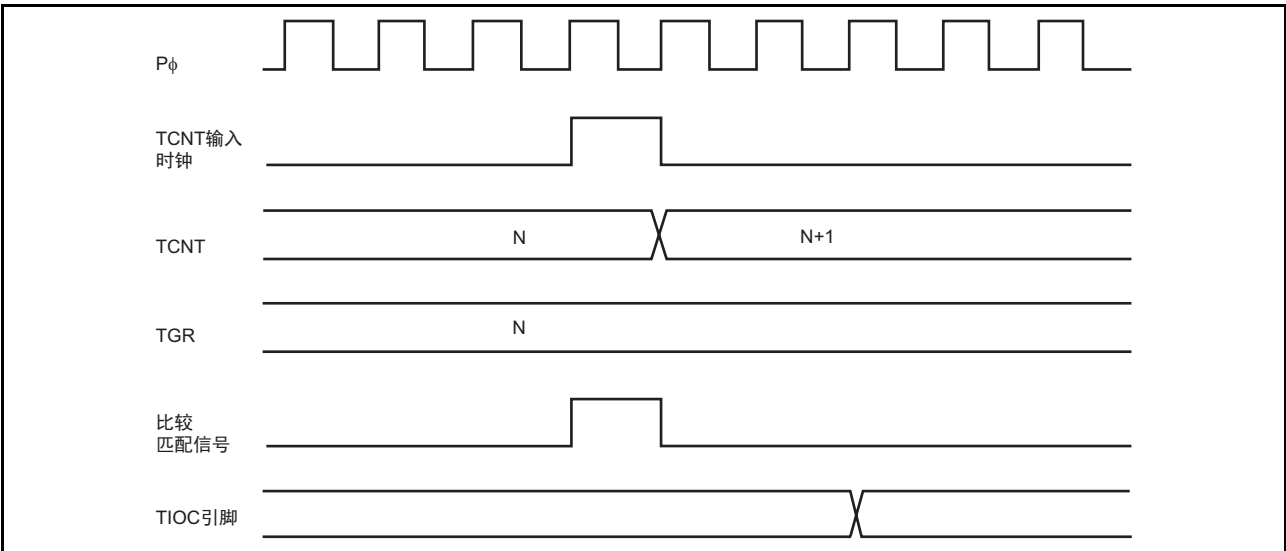


图 11.82 输出比较输出时序（互补 PWM 模式、复位同步 PWM 模式）

(3) 输入捕捉信号时序

输入捕捉的时序如图 11.83 所示。

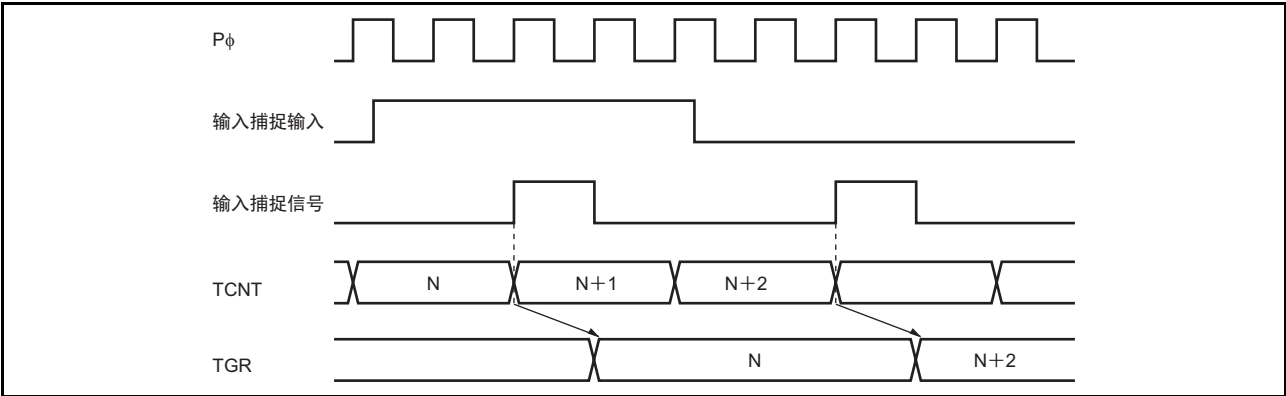


图 11.83 输入捕捉输入信号时序

(4) 由比较匹配 / 输入捕捉清除计数器的时序

指定由产生比较匹配清除计数器时的时序如图 11.84 所示。

指定由产生输入捕捉清除计数器时的时序如图 11.85 所示。

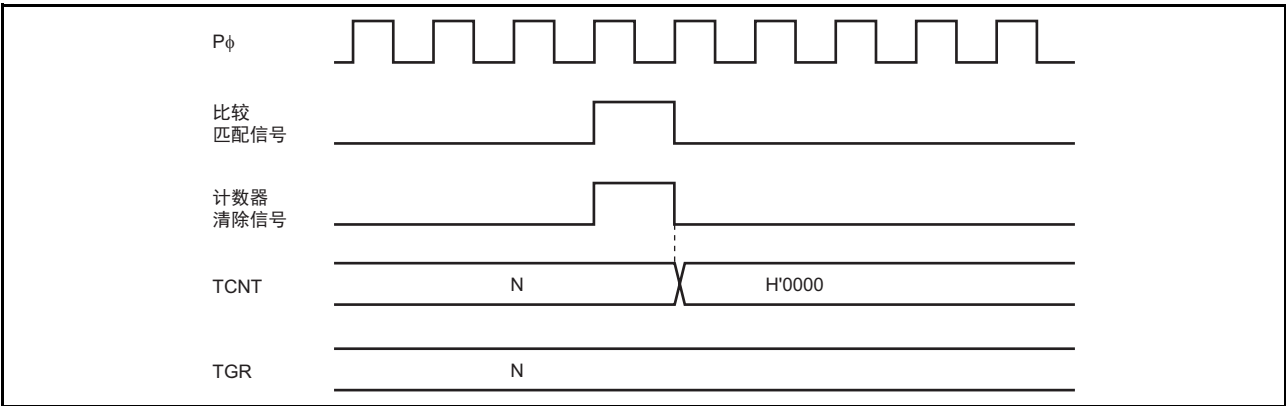


图 11.84 计数器清除时序 (比较匹配)

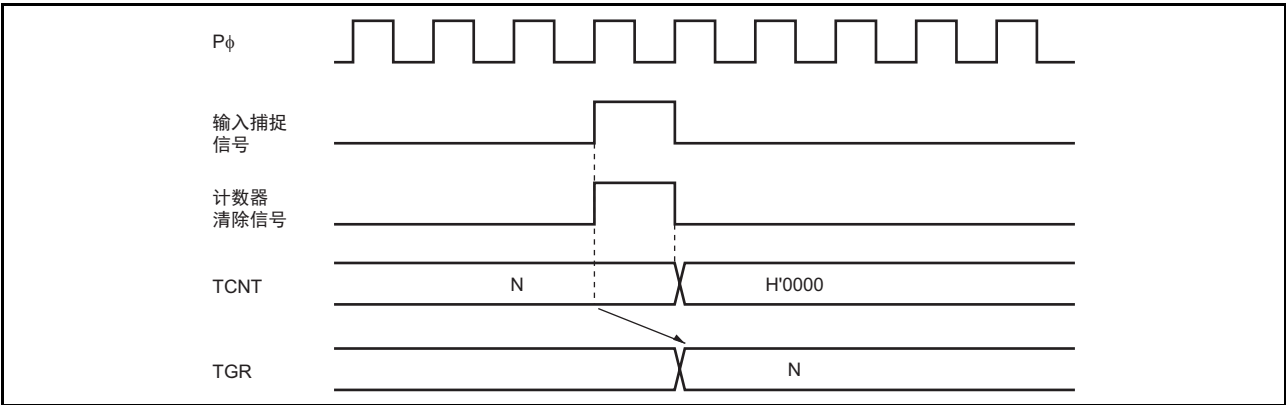


图 11.85 计数器清除时序 (输入捕捉)

(5) 缓冲运行时序

缓冲运行的时序如图 11.86 ~ 图 11.88 所示。

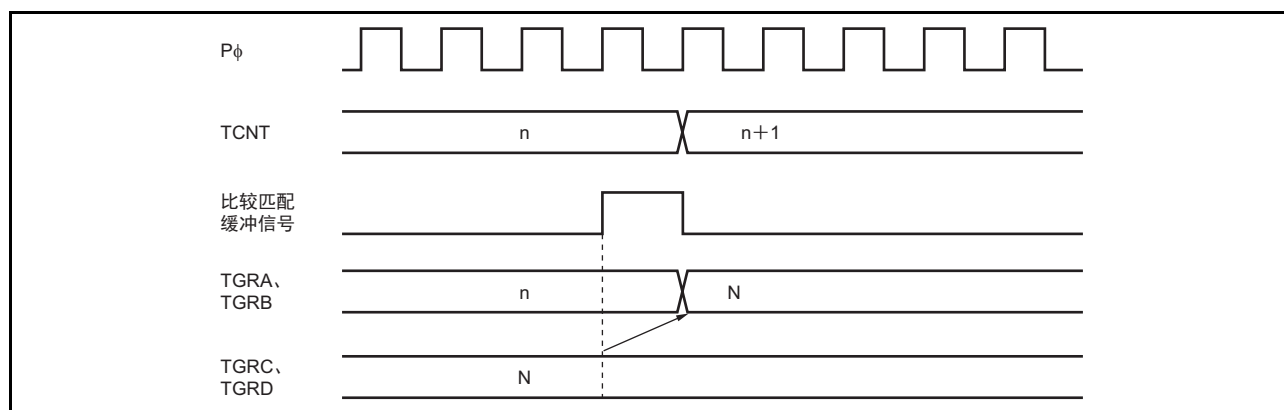


图 11.86 缓冲运行时序（比较匹配）

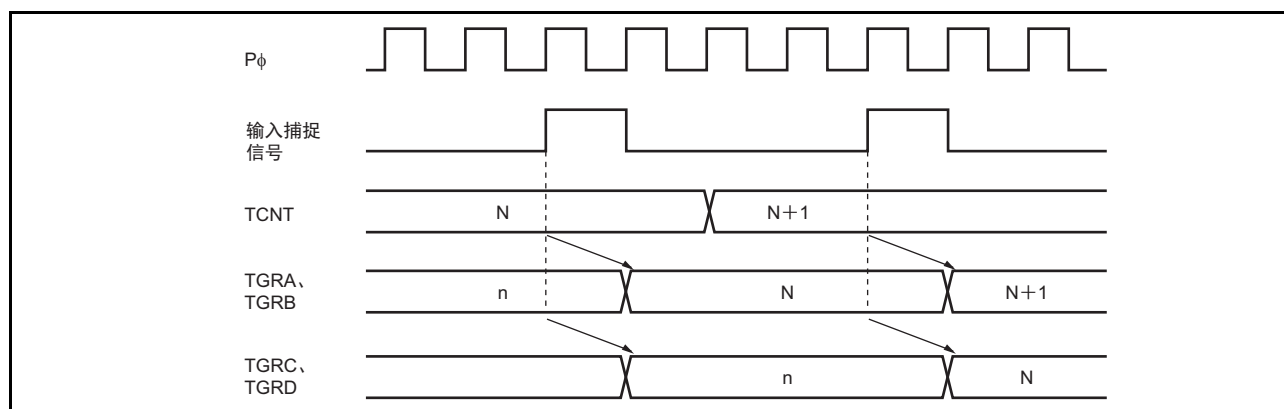


图 11.87 缓冲运行时序（输入捕捉）

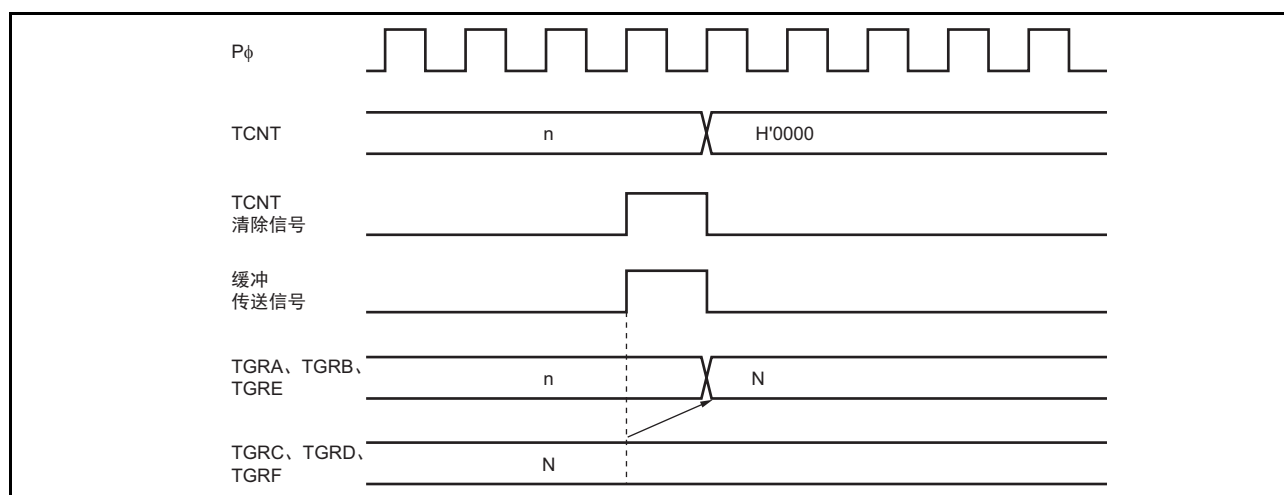


图 11.88 缓冲运行时序（清除 TCNT 时）

(6) 缓冲传送时序 (互补 PWM 模式)

互补 PWM 模式时的缓冲传送时序如图 11.89 ~ 图 11.91 所示。

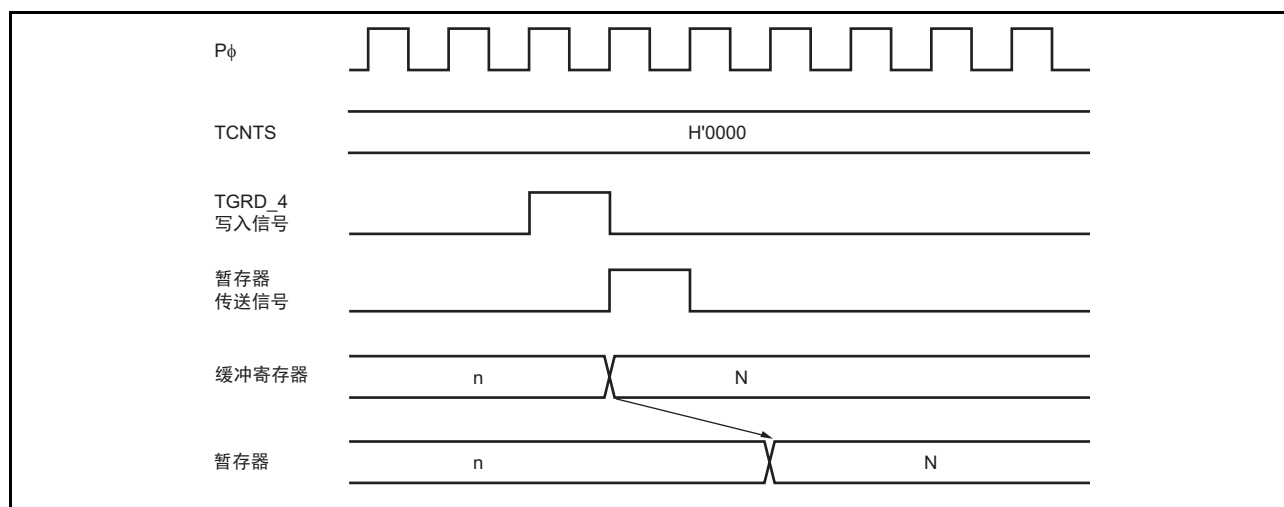


图 11.89 从缓冲寄存器向暂存器传送的时序 (TCNTS 停止时)

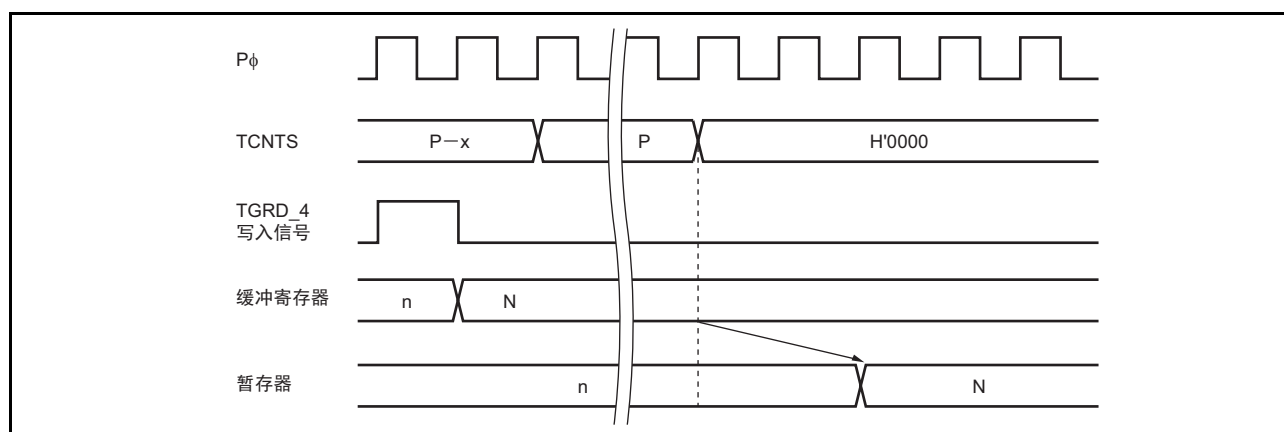


图 11.90 从缓冲寄存器向暂存器传送的时序 (TCNTS 运行时)

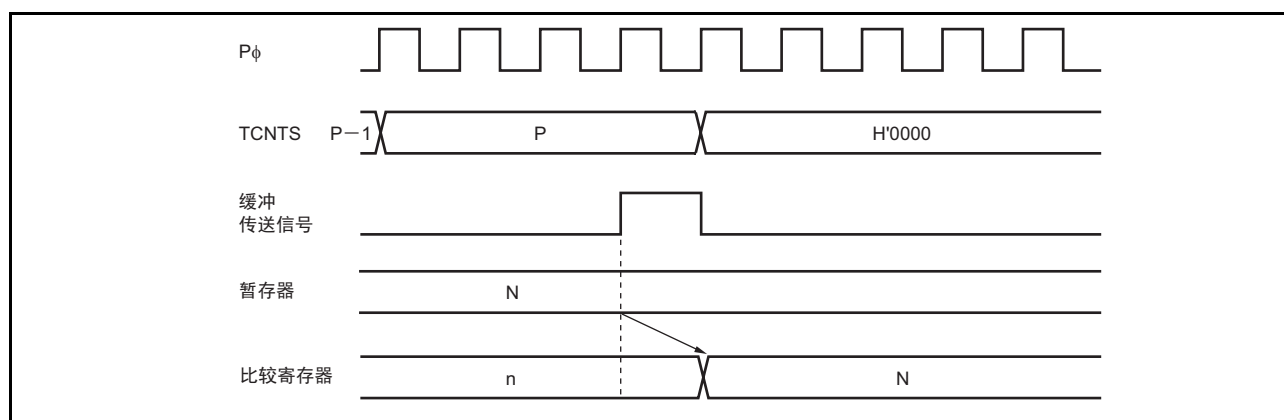


图 11.91 从暂存器向比较寄存器传送的时序

11.6.2 中断信号时序

(1) 比较匹配时的 TGF 标志的置位时序

因产生比较匹配引起的 TSR 的 TGF 标志置位时序和 TGI 中断请求信号的时序如图 11.92 所示。

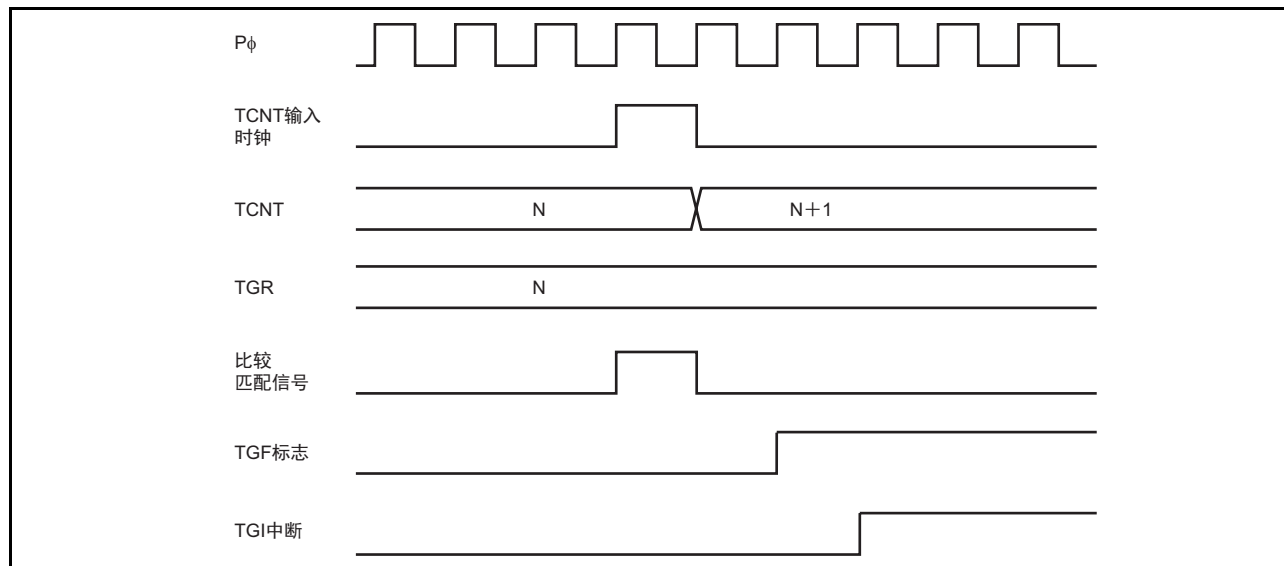


图 11.92 TGI 中断时序（比较匹配）

(2) 输入捕捉时的 TGF 标志的置位时序

因产生输入捕捉引起的 TSR 的 TGF 标志置位时序和 TGI 中断请求信号的时序如图 11.93 所示。

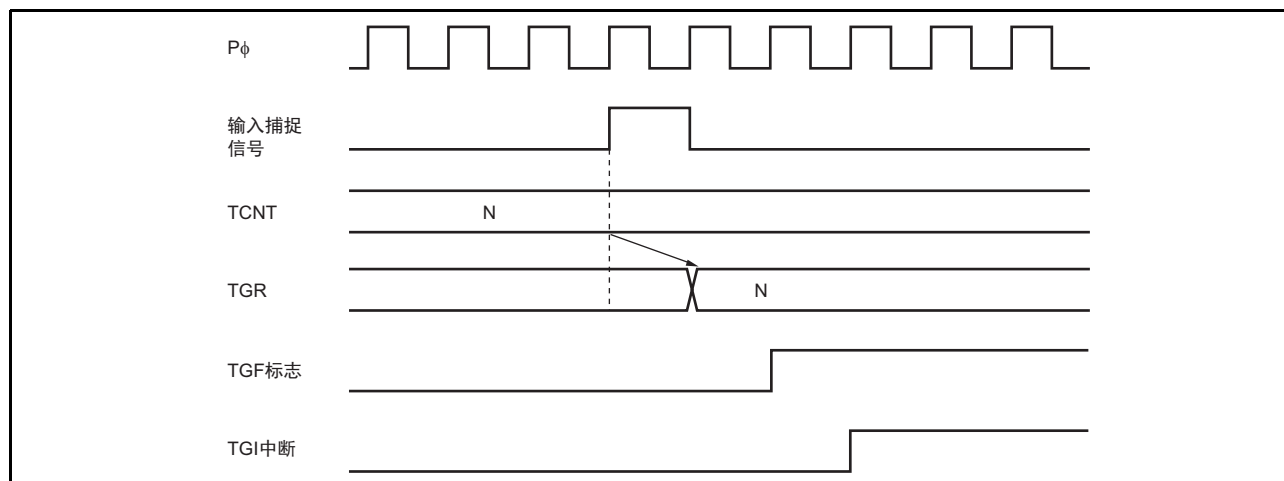


图 11.93 TGI 中断时序（输入捕捉）

(3) TCFV 标志 /TCFU 标志的置位时序

因产生上溢引起的 TSR 的 TCFV 标志置位时序和 TCIV 中断请求信号的时序如图 11.94 所示。
因产生下溢引起的 TSR 的 TCFU 标志置位时序和 TCIU 中断请求信号的时序如图 11.95 所示。

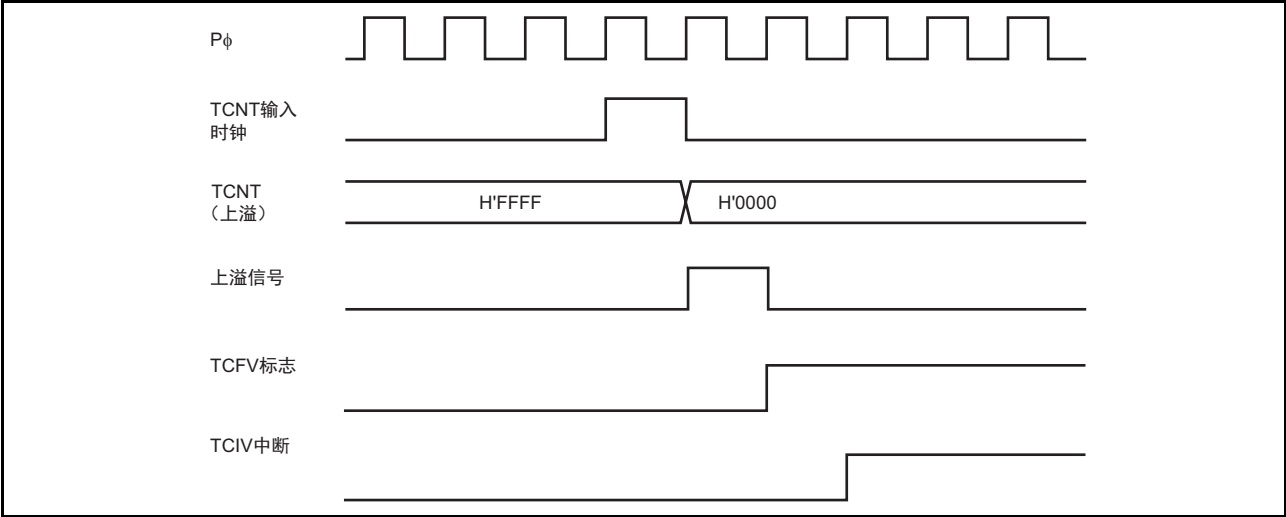


图 11.94 TCIV 中断的置位时序

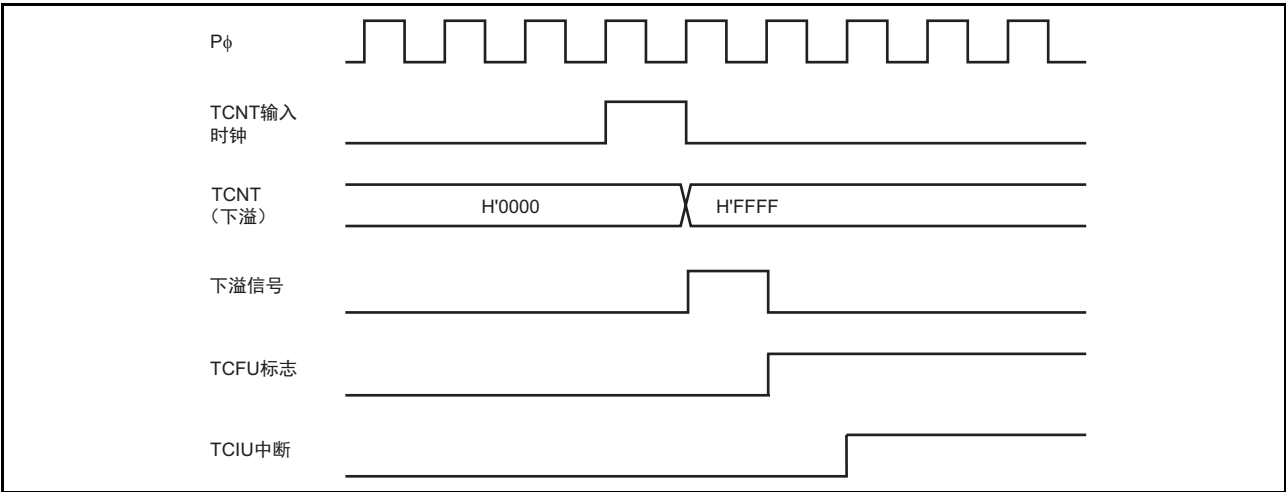


图 11.95 TCIU 中断的置位时序

(4) 状态标志的清除时序

CPU 读取 1 后写入 0，可清除状态标志。启动 DMAC 时，也可自动清除。通过 CPU 清除状态标志的时序如图 11.96 所示，通过 DMAC 清除状态标志的时序如图 11.97 所示。

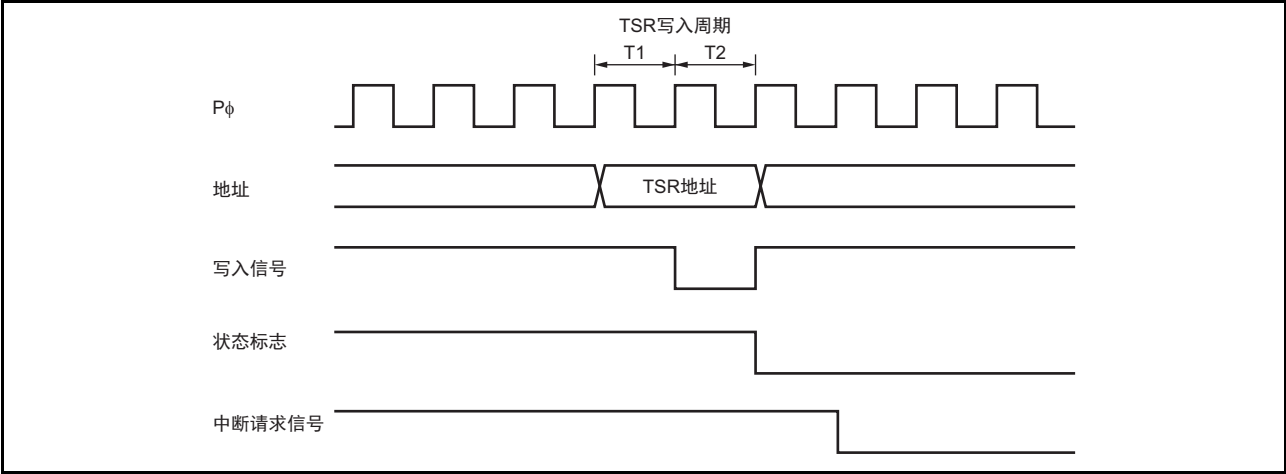


图 11.96 通过 CPU 清除状态标志的时序

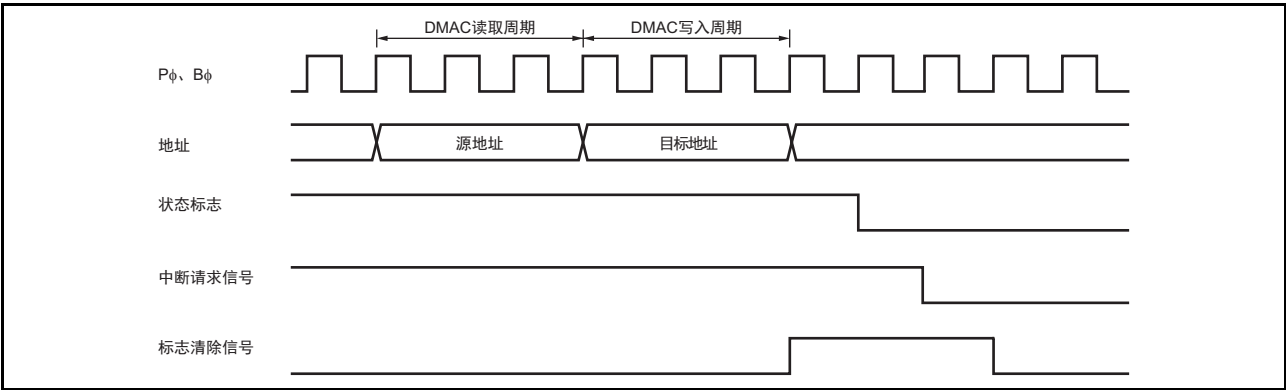


图 11.97 通过启动 DMAC 清除状态标志的时序

11.7 使用时的注意事项

11.7.1 模块待机模式的设定

MTU2 可由待机控制寄存器设定禁止 / 允许本模块运行。待机控制寄存器为初始值时，MTU2 停止运行。通过解除模块待机模式，即可存取寄存器。详情参阅“第 28 章 低功耗模式”。

11.7.2 输入时钟的限制事项

在单边沿时输入时钟的脉宽至少需要 1.5 个状态时钟，在双边沿时至少需要 2.5 个状态时钟。必须注意，如果没有达到脉宽则无法正常运行。

相位计数模式时，2 个输入时钟的相位差及重叠分别至少需要 1.5 个状态时钟，脉宽至少为 2.5 个状态时钟。相位计数模式的输入时钟条件如图 11.98 所示。

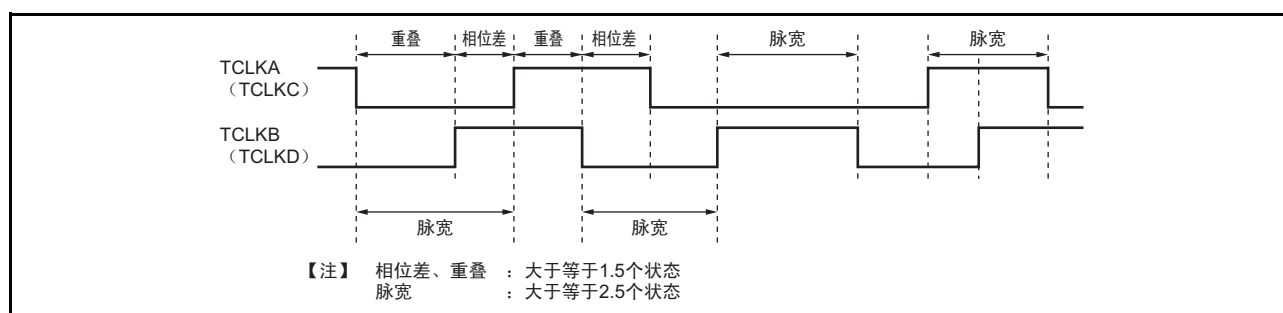


图 11.98 相位计数模式时的相位差、重叠及脉宽

11.7.3 周期设定时的注意事项

设定为由比较匹配清除计数器时，则在 TCNT 和 TGR 的值匹配的最后状态（更新 TCNT 匹配后的计数值时序）清除 TCNT。因此，实际计算计数器频率的公式如下：

$$f = \frac{P\phi}{(n+1)}$$

f : 计数器频率

$P\phi$: 外围时钟运行频率

N : TGR 的设定值

11.7.4 TCNT 的写入和清除的竞争

如果在 TCNT 写入周期的 T2 状态产生计数器清除信号，则不写入 TCNT，而优先清除 TCNT。
此时序如图 11.99 所示。

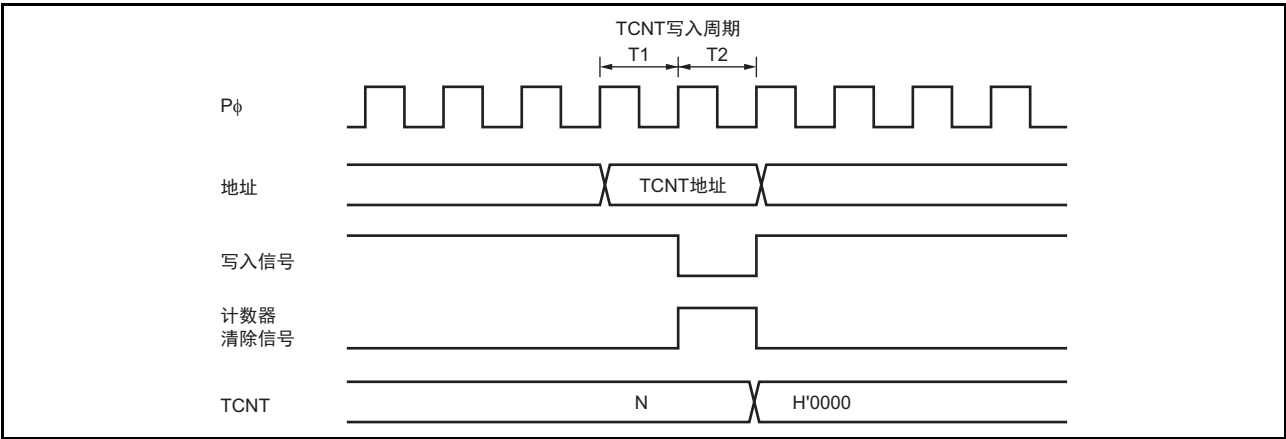


图 11.99 TCNT 的写入和清除的竞争

11.7.5 TCNT 的写入和递增计数的竞争

即使在 TCNT 写入周期的 T2 状态产生递增计数，也不执行递增计数，而优先写入 TCNT。
此时序如图 11.100 所示。

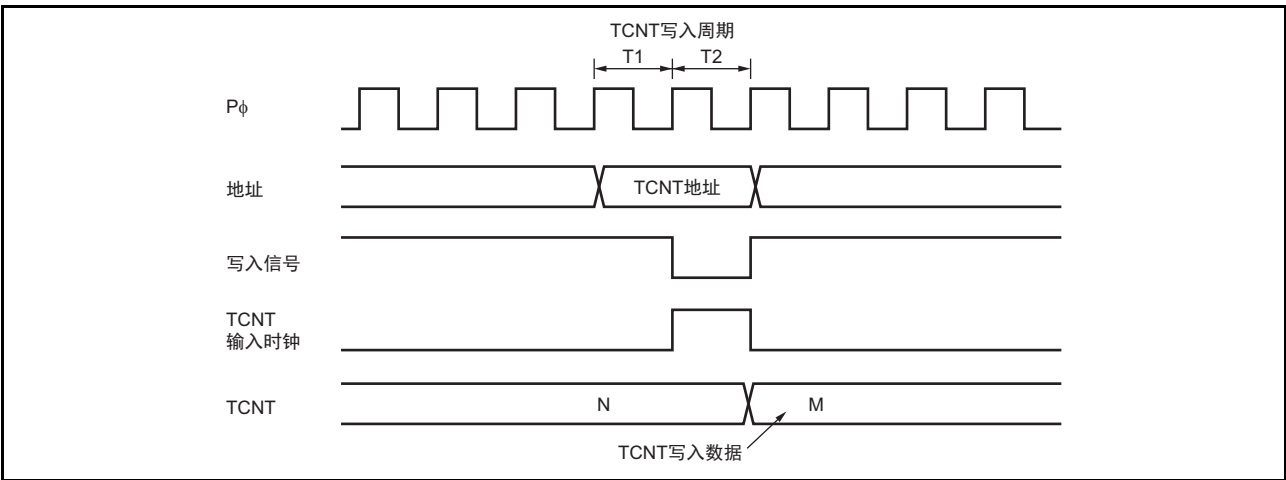


图 11.100 TCNT 的写入和递增计数的竞争

11.7.6 TGR 的写入和比较匹配的竞争

如果在 TGR 写入周期的 T2 状态产生比较匹配，则写入 TGR，并且也产生比较匹配信号。此时序如图 11.101 所示。

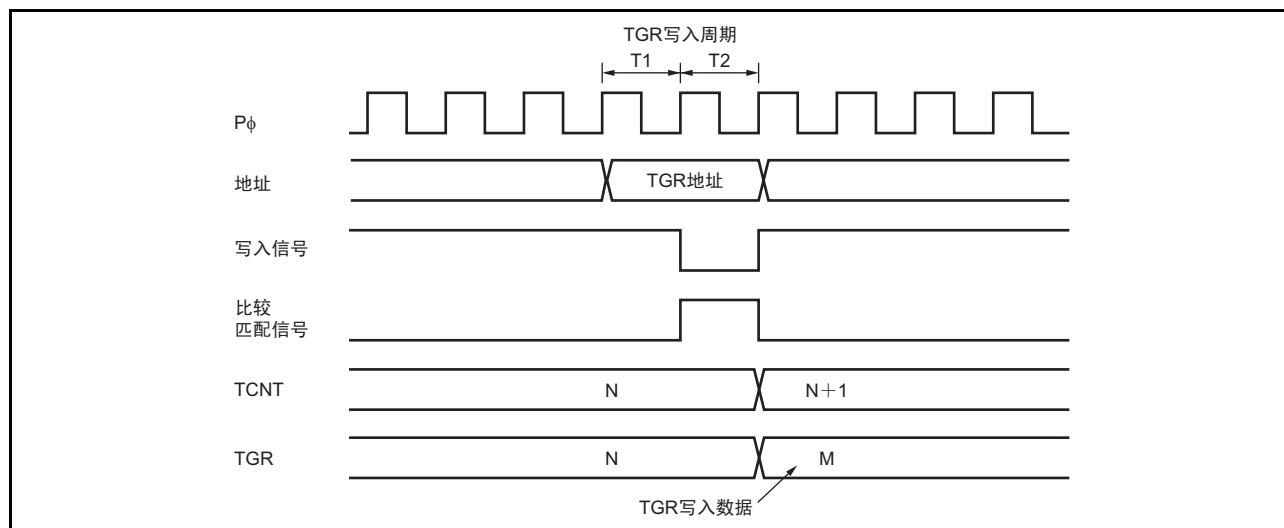


图 11.101 TGR 的写入和比较匹配的竞争

11.7.7 缓冲寄存器的写入和比较匹配的竞争

如果在 TGR 写入周期的 T2 状态产生比较匹配，则由缓冲运行传送至 TGR 的数据为写入前的数据。此时序如图 11.102 所示。

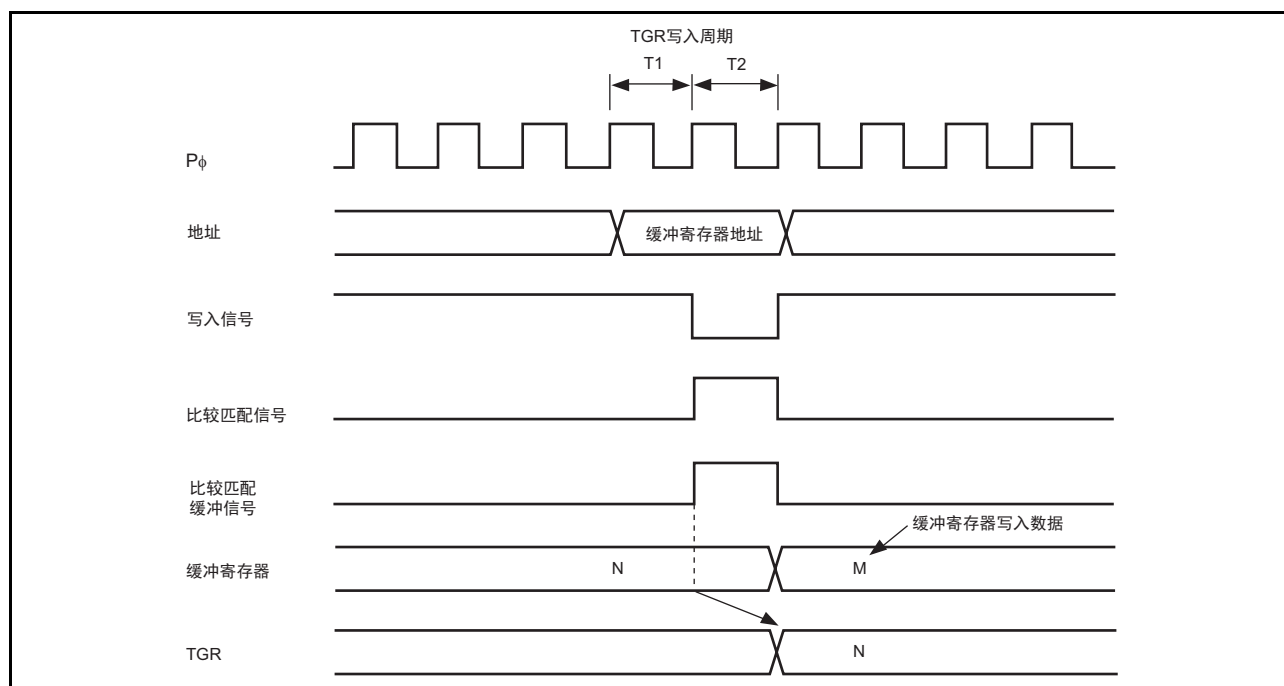


图 11.102 缓冲寄存器的写入和比较匹配的竞争

11.7.8 缓冲寄存器的写入和 TCNT 清除的竞争

通过缓冲传送模式寄存器 (TBTM) 将缓冲传送时序设定为清除 TCNT 时, 如果在 TGR 写入周期的 T2 状态产生 TCNT 清除, 则通过缓冲运行传送至 TGR 的数据为写入前的数据。

此时序如图 11.103 所示。

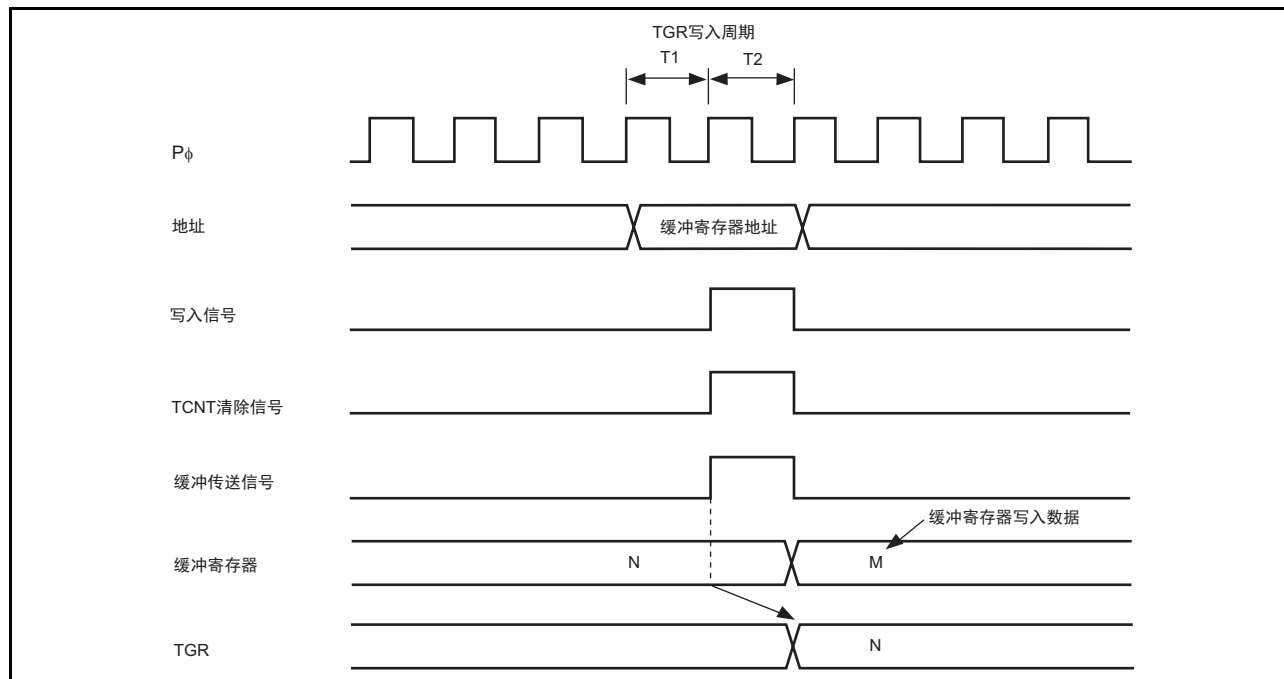


图 11.103 缓冲寄存器的写入和 TCNT 清除的竞争

11.7.9 TGR 的读取和输入捕捉的竞争

如果在 TGR 读取周期的 T1 状态产生输入捕捉信号, 则读取的数据为输入捕捉传送前的数据。

此时序如图 11.104 所示。

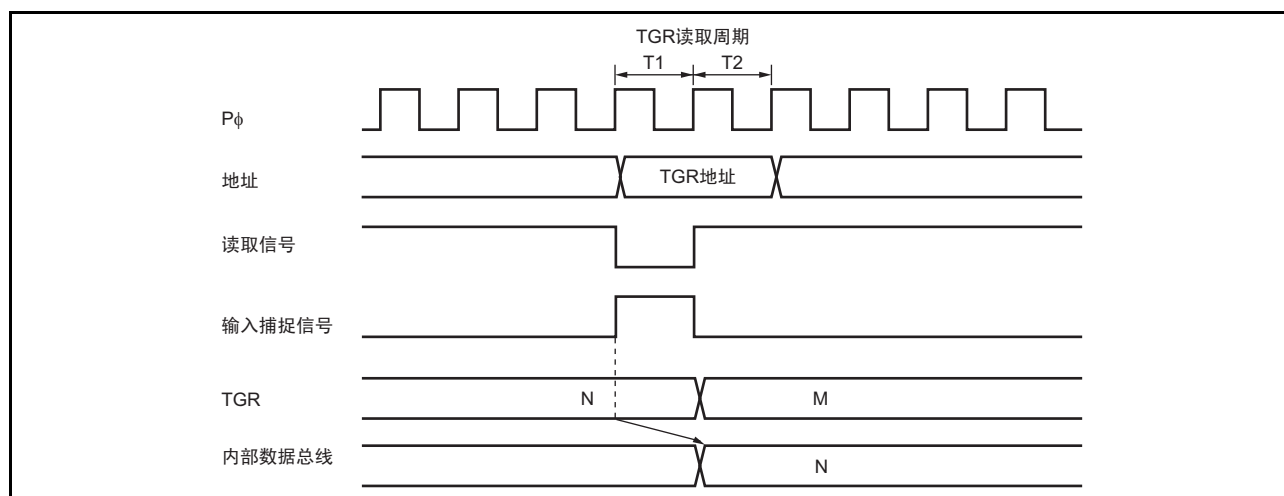


图 11.104 TGR 的读取和输入捕捉的竞争

11.7.10 TGR 的写入和输入捕捉的竞争

如果在 TGR 写入周期的 T2 状态产生输入捕捉信号，则不写入 TGR，而优先输入捕捉。
此时序如图 11.105 所示。

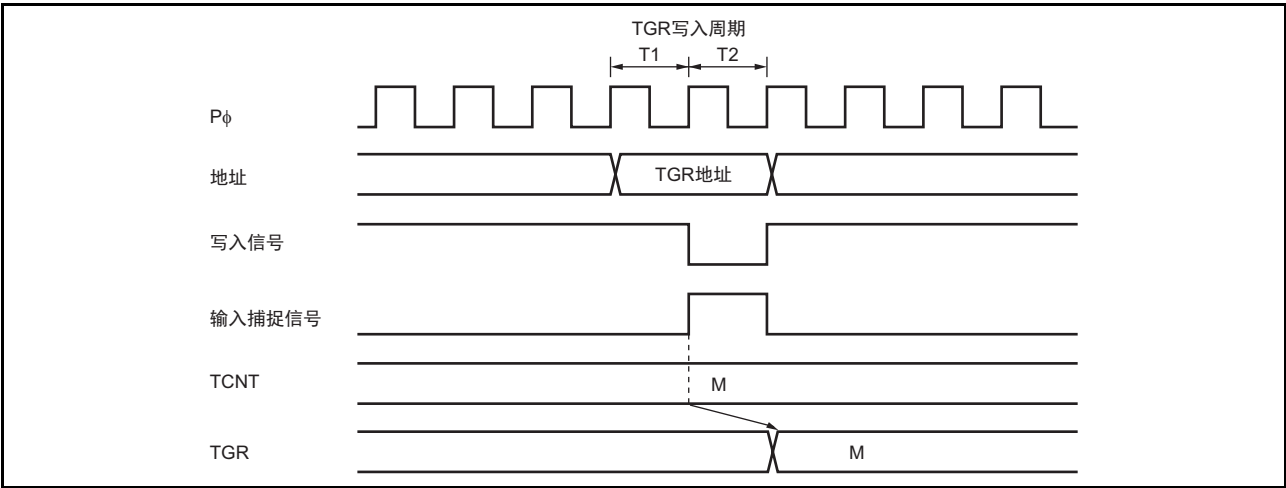


图 11.105 TGR 的写入和输入捕捉的竞争

11.7.11 缓冲寄存器的写入和输入捕捉的竞争

如果在缓冲器写入周期的 T2 状态产生输入捕捉信号，则不写入缓冲寄存器，而优先缓冲运行。
此时序如图 11.106 所示。

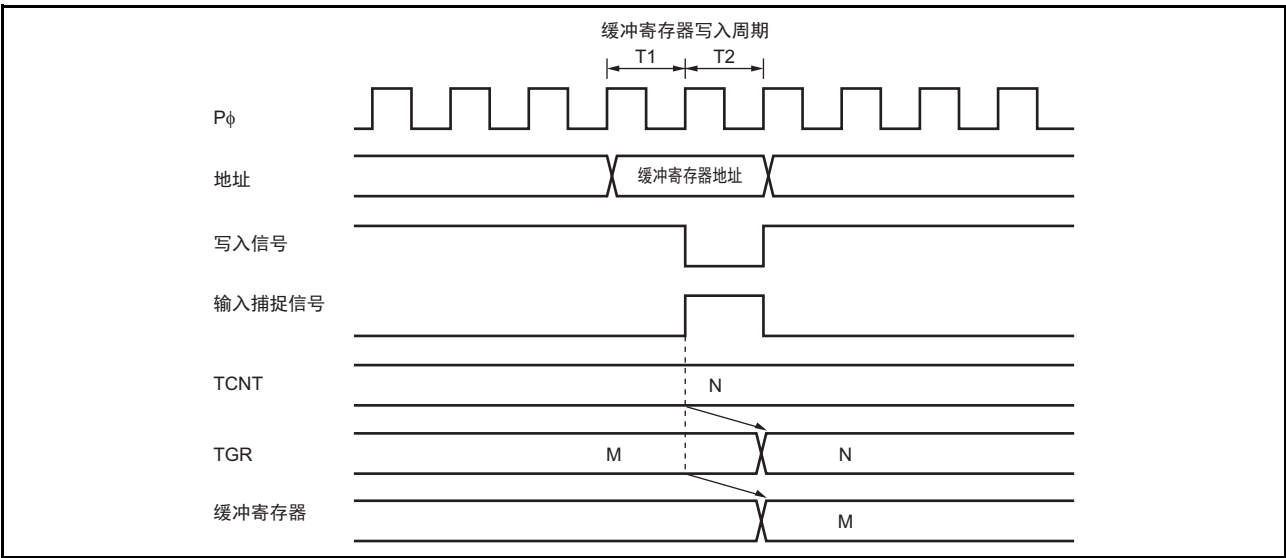


图 11.106 缓冲寄存器的写入和输入捕捉的竞争

11.7.12 级联中 TCNT_2 的写入和上溢 / 下溢的竞争

将定时器计数器 (TCNT_1 和 TCNT_2) 级联, 如果 TCNT_1 在计数的瞬间 (TCNT_2 产生上溢 / 下溢的瞬间) 和 TCNT_2 写入周期的 T2 状态产生竞争, 则写入 TCNT_2, 禁止 TCNT_1 的计数信号。此时, TGRA_1 作为比较匹配寄存器运行, 与 TCNT_1 的值匹配时, 产生比较匹配信号。

选择 TCNT_1 计数时钟作为通道 0 的输入捕捉源时, TGRA_0 ~ D_0 执行输入捕捉, 并且选择 TGRC_0 的比较匹配 / 输入捕捉作为 TGRB_1 的输入捕捉源时, TGRB_1 执行输入捕捉。

此时序如图 11.107 所示。

通过级联运行设定清除 TCNT 时, 必须同步设定通道 1 和通道 2。

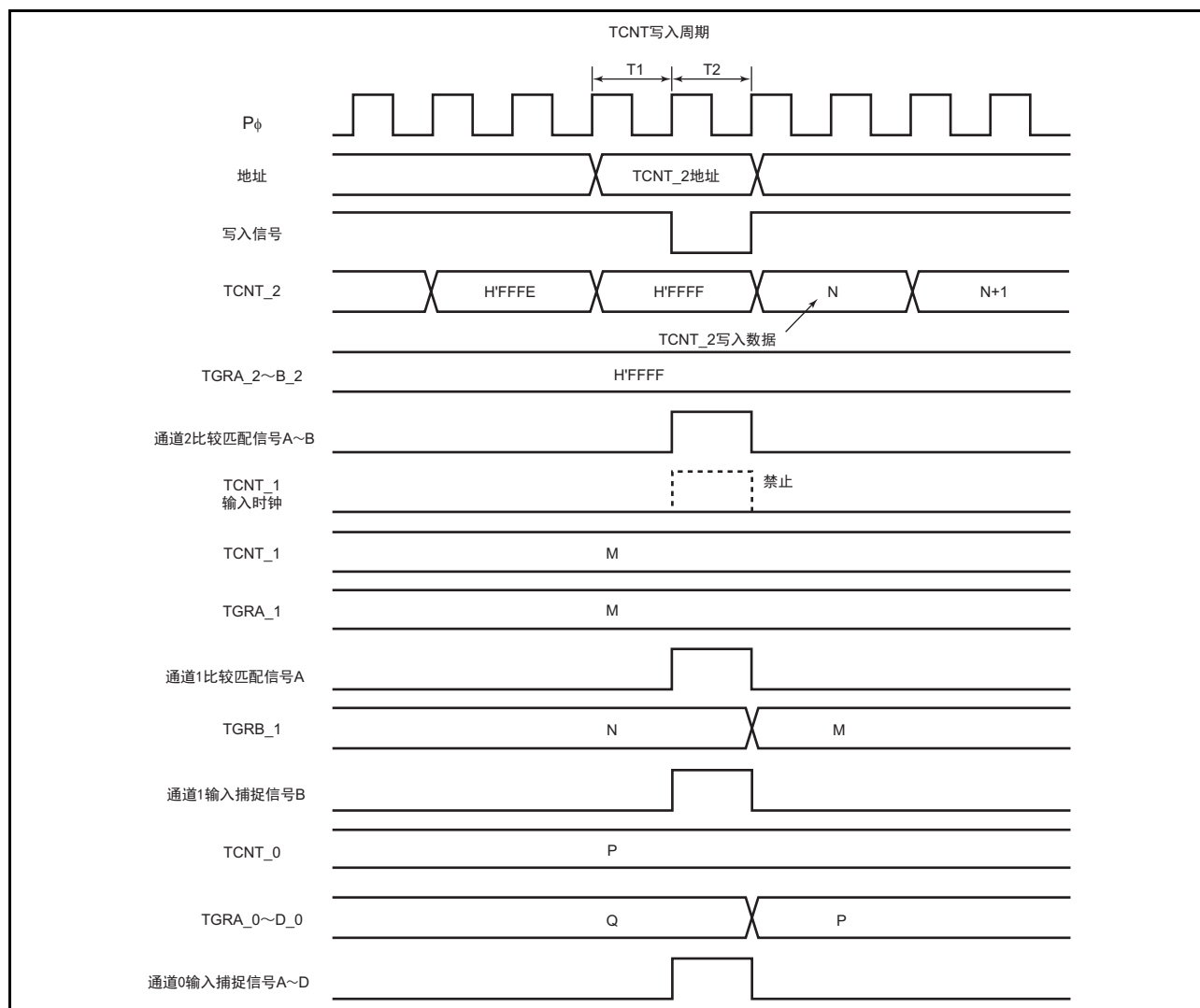


图 11.107 级联中 TCNT_2 的写入和上溢 / 下溢的竞争

11.7.13 互补 PWM 模式停止时的计数器值

如果 TCNT_3 和 TCNT_4 在互补 PWM 模式运行时停止计数，则 TCNT_3 为定时器空载时间寄存器 (TDDR) 的值，TCNT_4 为 H'0000。

重新开始互补 PWM 时，自动从初始状态计数。

此说明如图 11.108 所示。

在其他运行模式开始计数时，必须对 TCNT_3 和 TCNT_4 设定计数初始值。

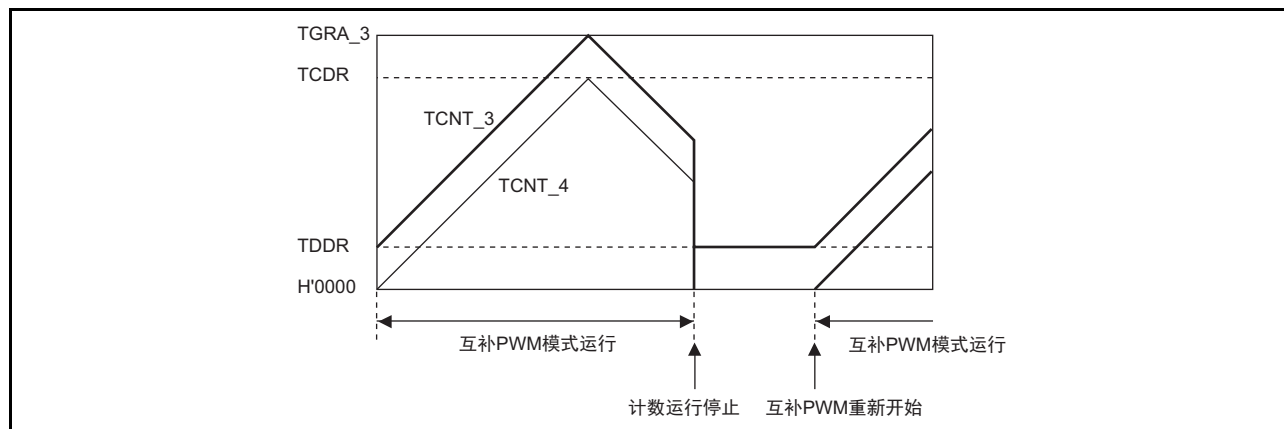


图 11.108 互补 PWM 模式停止时的计数器值

11.7.14 互补 PWM 模式时的缓冲运行的设定

在互补 PWM 模式，必须在缓冲运行中改写 PWM 周期设定寄存器 (TGRA_3)、定时器周期数据寄存器 (TCDR) 和占空比设定寄存器 (TGRB_3、TGRA_4、TGRB_4)。

互补 PWM 模式时通道 3 和通道 4 的缓冲运行按照 TMDR_3 的 BFA 和 BFB 位的设定运行，TMDR_3 的 BFA 位置 1，TGRC_3 用作 TGRA_3 的缓冲寄存器，同时，TGRC_4 用作 TGRA_4 的缓冲寄存器，TCBR 用作 TCDR 的缓冲寄存器。

11.7.15 复位同步 PWM 模式的缓冲运行和比较匹配标志

在复位同步 PWM 模式设定缓冲运行时，必须将 TMDR_4 的 BFA、BFB 位设定为 0。如果 TMDR_4 的 BFA 位设定为 1，则不可输出 TIOC4C 引脚的波形。

复位同步 PWM 模式时通道 3 和通道 4 的缓冲运行按照 TMDR_3 的 BFA、BFB 位的设定运行。例如，TMDR_3 的 BFA 位置 1 时，TGRC_3 用作 TGRA_3 的缓冲寄存器，同时，TGRC_4 用作 TGRA_4 的缓冲寄存器。

TGRC_3、TGRD_3 作为缓冲寄存器运行时，TSR_3 及 TSR_4 的 TGFC 位和 TGFD 位不置位。

TMDR_3 的 BFA、BFB 位置 1，TMDR_4 的 BFA、BFB 位置 0 时，TGR_3、TGR_4、TIOC3、TIOC4 的运行例如图 11.109 所示。

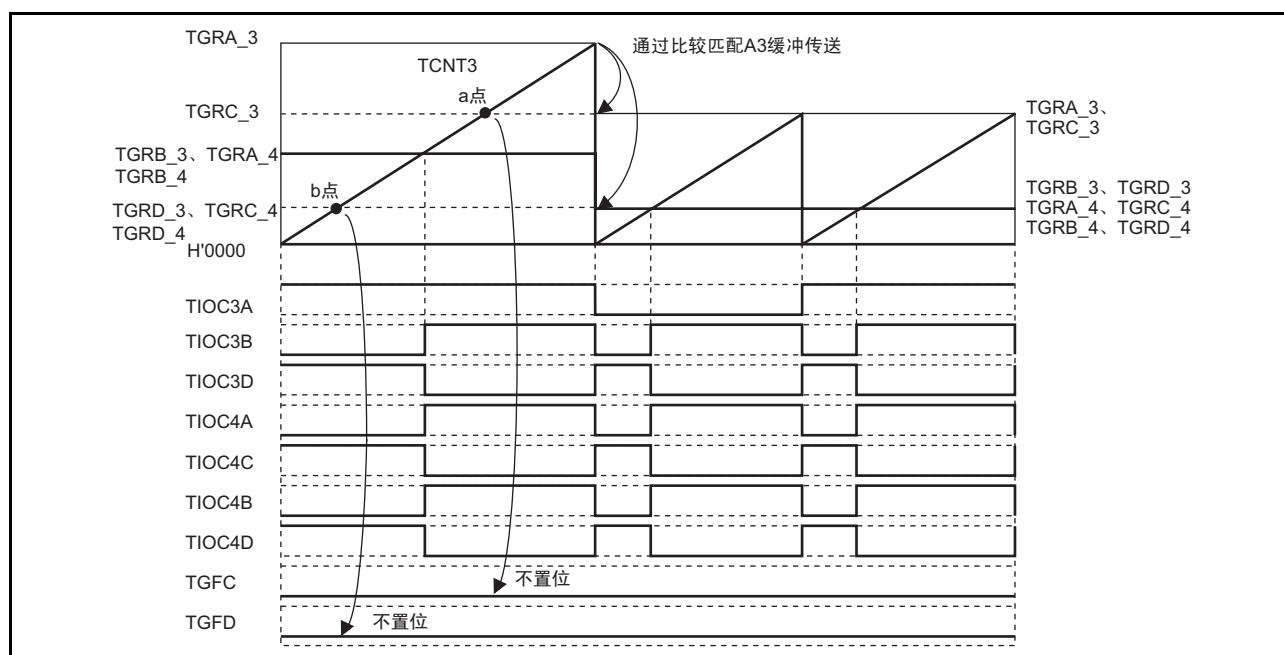


图 11.109 复位同步 PWM 模式的缓冲运行和比较匹配标志

11.7.16 复位同步 PWM 模式的上溢标志

设定复位同步 PWM 模式，TSTR 的 CST3 位设定为 1 时，则 TCNT_3 和 TCNT_4 开始计数运行。此时，TCNT_4 的计数时钟源和计数边沿取决于与 TCR_3 的设定。

在复位同步 PWM 模式，周期寄存器 TGRA_3 的设定值为 H'FFFF，指定 TGRA_3 的比较匹配为计数器清除源时，如果 TCNT_3 和 TCNT_4 递增计数到 H'FFFF，则产生与 TGRA_3 的比较匹配，并同时清除 TCNT_3 和 TCNT_4 计数。TSR 的上溢标志 TCFV 位不置位。

在复位同步 PWM 模式，周期寄存器 TGRA_3 的设定值为 H'FFFF，并指定 TGRA_3 的比较匹配为计数器清除源，未设定同步设定时的 TCFV 位的运行例如图 11.110 所示。

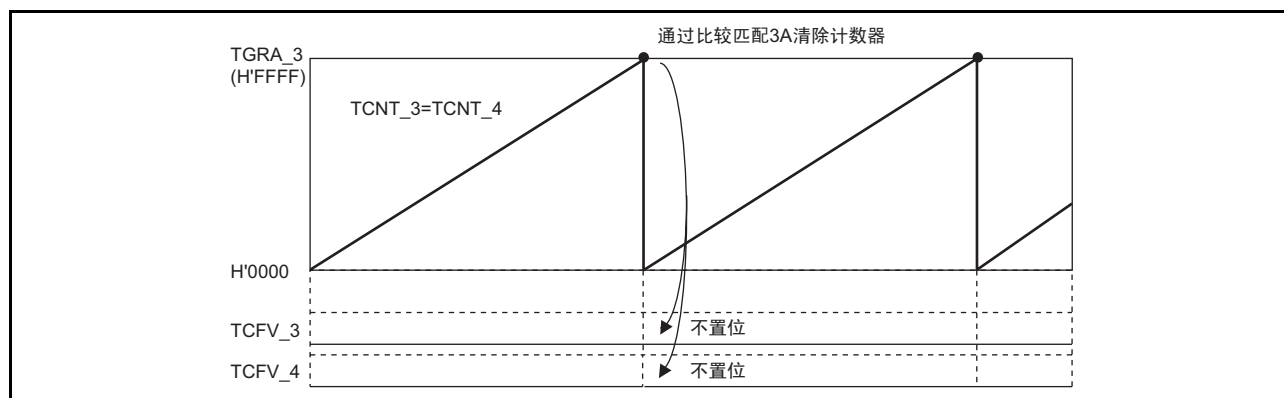


图 11.110 复位同步 PWM 模式的上溢标志

11.7.17 上溢 / 下溢和计数器清除的竞争

如果同时产生上溢 / 下溢和计数器清除，则 TSR 的 TCFV/TCFU 标志不置位，而优先清除 TCNT。将 TGR 的比较匹配作为清除源，并对 TGR 设定 H'FFFF 时的运行时序如图 11.111 所示。

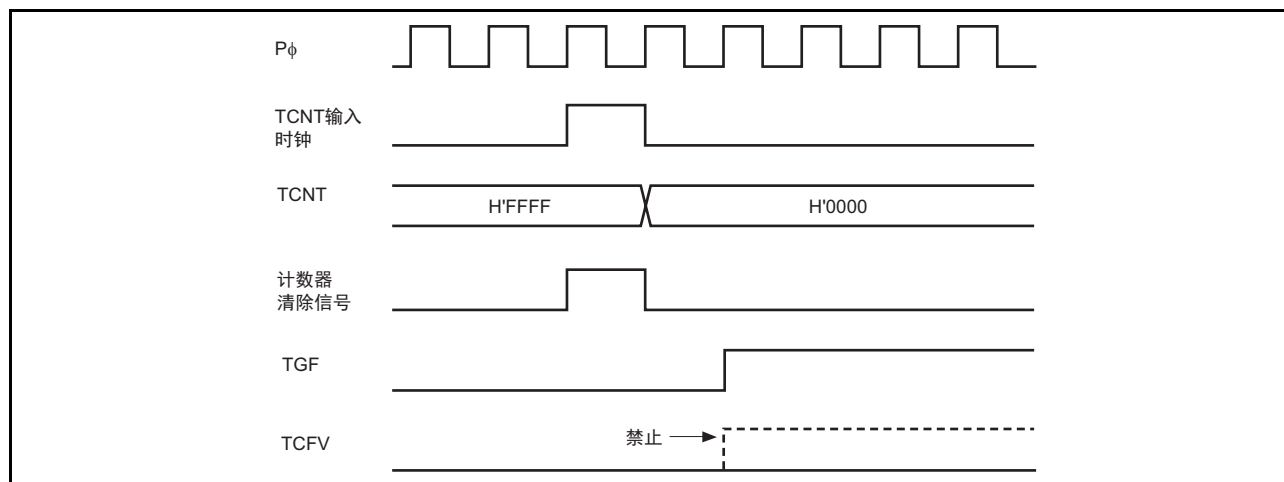


图 11.111 上溢和计数器清除的竞争

11.7.18 TCNT 的写入和上溢 / 下溢的竞争

在 TCNT 写入周期的 T2 状态，即使产生递增计数 / 递减计数和上溢 / 下溢，也要优先写入 TCNT，TSR 的 TCFV/TCFU 标志不置位。

TCNT 的写入和上溢竞争时的运行时序如图 11.112 所示。

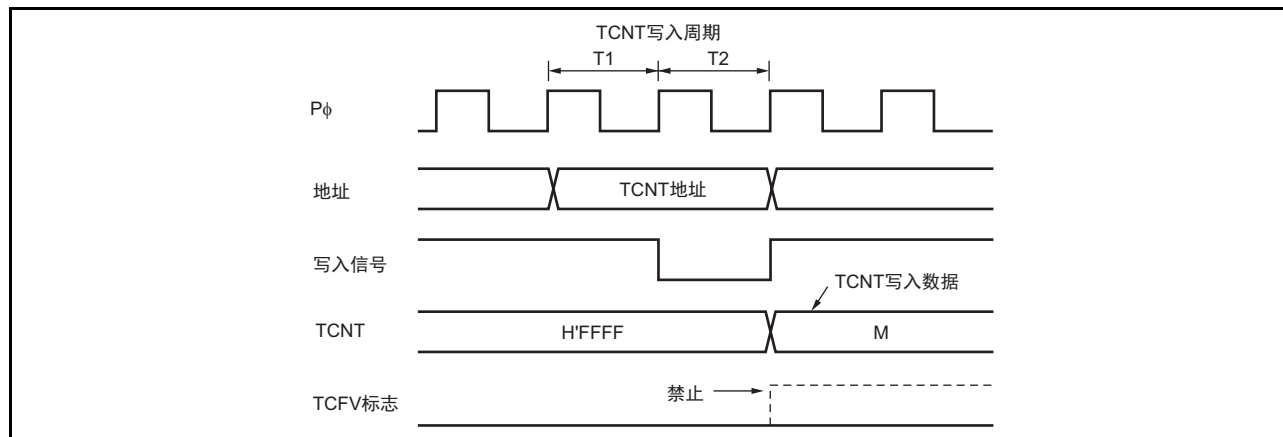


图 11.112 TCNT 的写入和上溢的竞争

11.7.19 从普通运行或 PWM 模式 1 转移至复位同步 PWM 模式时的注意事项

从通道 3、4 的普通运行或 PWM 模式 1 转移至复位同步 PWM 模式时，必须注意：如果在输出引脚（TIOC3B、TIOC3D、TIOC4A、TIOC4C、TIOC4B、TIOC4D）保持高电平状态停止计数器，并转移至复位同步 PWM 模式后使其运行，则无法正确输出引脚的初始输出。

从通常运行转移至复位同步 PWM 模式时，必须对 TIORH_3、TIORL_3、TIORH_4、TIORL_4 寄存器写入 H'11，并将输出引脚初始化为低电平，设定寄存器的初始值 H'00 之后再转移。

从 PWM 模式 1 向复位同步 PWM 模式转移时，先转移至通常运行，将输出引脚初始化为低电平后，设定寄存器的初始值 H'00，再转移至复位同步 PWM 模式。

11.7.20 互补 PWM 模式、复位同步 PWM 模式的输出电平

通道 3、4 为互补 PWM 模式或复位同步 PWM 模式时，由定时器输出控制寄存器（TOCR）的 OLSP、OLSN 位设定 PWM 波形的输出电平。在互补 PWM 模式或复位同步 PWM 模式，TIOB 必须设置为 H'00。

11.7.21 模块待机时的中断

在请求中断的状态进入模块待机时，不可清除 CPU 的中断源或 DMAC 的启动源。

必须预先禁止中断后，再设置模块待机模式。

11.7.22 级联时 TCNT_1 和 TCNT_2 同时输入捕捉

将定时器计数器 1、2 (TCNT_1 和 TCNT_2) 级联后作为 32 位计数器运行时, 即使 TIOC1A 和 TIOC2A, 或 TIOC1B 和 TIOC2B 同时执行输入捕捉输入, 与内部时钟同步将输入到 TCNT_1、TCNT_2 的外部输入捕捉信号输入内部时, 产生 TIOC1A、TIOC2A、或 TIOC1B 和 TIOC2B 输入时序偏移, 可能无法正常捕捉级联计数器值。

例如, TCNT_1 (高 16 位计数器) 本应由 TCNT_2 (低 16 位计数器) 的上溢捕捉递增计数值, 却捕捉递增计数前的计数值。此时, 本应将 TCNT_1=H'FFF1、TCNT_2=H'0000 的值传送至 TGRA_1 和 TGRA_2 或 TGRB_1 和 TGRB_2, 却误传 TCNT_1=H'FFF0、TCNT_2=H'0000 的值。

11.8 MTU2 输出引脚的初始化方法

11.8.1 运行模式

MTU2 有以下 6 种运行模式, 可在任意一种模式输出波形。

- 普通模式 (通道0~4)
- PWM 模式1 (通道0~4)
- PWM 模式2 (通道0~2)
- 相位计数模式1~4 (通道1、2)
- 互补PWM模式 (通道3、4)
- 复位同步PWM模式 (通道3、4)

在此说明各模式中 MTU2 输出引脚的初始化方法。

11.8.2 复位开始时的运行

MTU2 的输出引脚 (TIOC*) 在复位或待机模式时初始化为低电平。由引脚功能控制器 (PFC) 选择 MTU2 的引脚功能, 因此设定 PFC 时, 向端口输出当时 MTU2 的引脚状态。复位后立即由 PFC 选择 MTU2 输出时, 向端口输出 MTU2 输出的初始状态低电平。有效电平为低电平时, 系统已经运行, 因此必须在 MTU2 输出引脚的初始设定结束后, 设定 PFC。

【注】 * 通道编号 + 端口符号。

11.8.3 运行过程中因异常等而重新设定时的运行

MTU2 运行过程中产生异常时，必须通过系统截止 MTU2 的输出。由 PFC 将引脚的输出转换为端口输出，并通过输出有效电平的取反电平执行截止。因运行过程中的异常等重新设定时的引脚的初始化步骤及重新设定后在其他运行模式重新开始的步骤如下所示。

如前文所述，MTU2 有 6 种运行模式。模式转移的组合有 36 种，但也有通道和模式的组合中不存在的转移，一览表如表 11.57 所示。

使用以下符号表示：

Normal: 普通模式 PWM1: PWM 模式 1 PWM2: PWM 模式 2
PCM: 相位计数模式 1 ~ 4 CPWM: 互补 PWM 模式 RPWM: 复位同步 PWM 模式

表 11.57 模式转移的组合

	Normal	PWM1	PWM2	PCM	CPWM	RPWM
Normal	(1)	(2)	(3)	(4)	(5)	(6)
PWM1	(7)	(8)	(9)	(10)	(11)	(12)
PWM2	(13)	(14)	(15)	(16)	none	none
PCM	(17)	(18)	(19)	(20)	none	none
CPWM	(21)	(22)	none	none	(23) (24)	(25)
RPWM	(26)	(27)	none	none	(28)	(29)

11.8.4 因运行过程中的异常等引起引脚初始化的步骤、模式转移的概要

- 根据定时器 I/O 控制寄存器（TIOR）的设定转移至选择引脚输出电平的模式（Normal、PWM1、PWM2、PCM）时，必须通过设定 TIOR 初始化引脚。
- 在 PWM 模式 1，不对 TIOC*B（TIOC*D）引脚输出波形，因此即使设定 TIOR 也不初始化引脚。需初始化时，必须在普通模式初始化后，转移至 PWM 模式 1。
- 在 PWM 模式 2，不对周期寄存器的引脚输出波形，因此即使设定 TIOR 也不初始化引脚。需初始化时，必须在普通模式初始化后，转移至 PWM 模式 2。
- 在普通模式或 PWM 模式 2，TGRC、TGRD 作为缓冲寄存器运行时，即使设定 TIOR 也不初始化缓冲寄存器的引脚。需初始化时，必须解除缓冲模式并初始化后，重新设定缓冲模式。
- 在 PWM 模式 1，TGRC 或 TGRD 作为缓冲寄存器运行时，即使设定 TIOR 也不初始化缓冲寄存器 TGRC 的引脚。需初始化 TGRC 的引脚时，必须在解除缓冲模式并且初始化后，重新设定缓冲模式。
- 根据定时器输出控制寄存器（TOCR）的设定，转移至选择引脚输出电平的模式（CPWM、RPWM）时，必须转移至普通模式并由 TIOR 初始化，TIOR 返回初始值后，由定时器输出主控允许寄存器（TOER）暂时禁止通道 3、4 的输出。之后按照模式设定步骤（TOCR 设定、TMDR 设定、TOER 设定）运行。

【注】 本项记述中的 * 为通道编号。

按照表 11.57 的组合 No. 初始化引脚的步骤如下所示，有效电平为低电平。

(1) 在普通模式运行过程中产生异常，并在普通模式重新开始时的运行

在普通模式产生异常，重新设定后，在普通模式重新开始时的说明如图 11.113 所示。

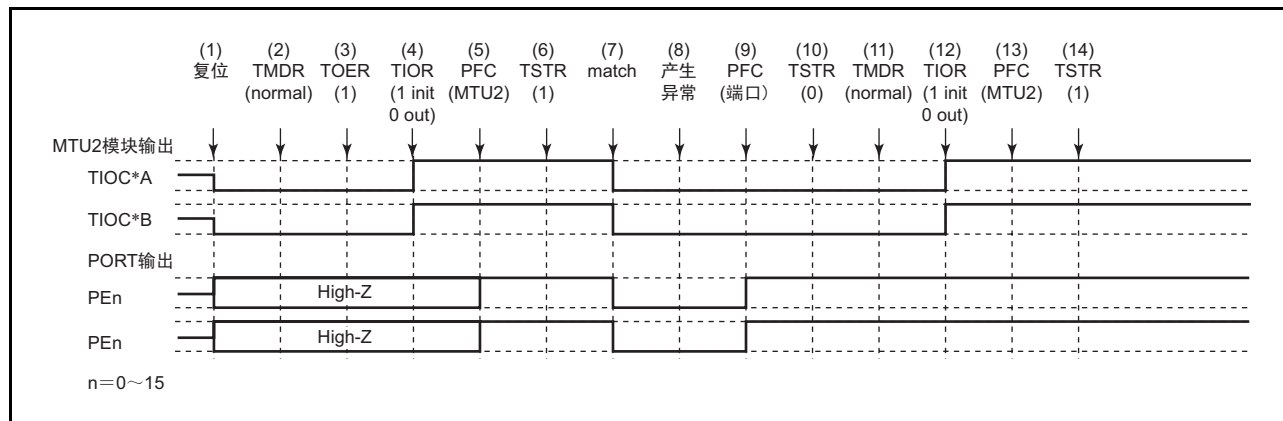


图 11.113 在普通模式产生异常，并在普通模式恢复时

- (1) 由于复位，MTU2 输出为低电平，端口为高阻抗。
- (2) 由于复位，TMDR 为普通模式设定。
- (3) 在通道 3、4 由 TIOR 初始化引脚前，必须由 TOER 允许输出。
- (4) 由 TIOR 初始化引脚（例如，初始输出为高电平，比较匹配时为低电平输出）。
- (5) 由 PFC 设定 MTU2 输出。
- (6) 由 TSTR 开始计数运行。
- (7) 通过产生比较匹配输出低电平。
- (8) 产生异常。
- (9) 由 PFC 设置为端口输出，输出取反的有效电平。
- (10) 由 TSTR 停止计数运行。
- (11) 在普通模式重新开始时无需此步骤。
- (12) 由 TIOR 初始化引脚。
- (13) 由 PFC 设定为 MTU2 输出。
- (14) 由 TSTR 重新开始。

(2) 在普通模式运行过程中产生异常，并在 PWM 模式 1 重新开始时的运行

在普通模式产生异常，重新设定后，在 PWM 模式 1 重新开始时的说明如图 11.114 所示。

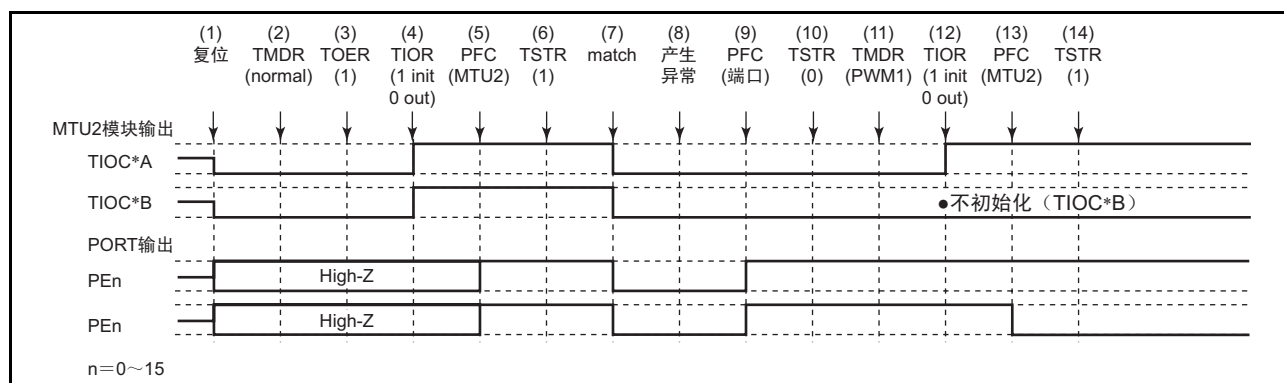


图 11.114 在普通模式产生异常，并在 PWM 模式 1 恢复时

(1) ~ (10) 和图 11.113 通用。

(11) 设定 PWM 模式 1。

(12) 由 TIOR 初始化引脚（在 PWM 模式 1 不初始化 TIOC*B。如果要初始化时，在普通模式初始化后，

转移至 PWM 模式 1）。

(13) 由 PFC 设定为 MTU2 输出。

(14) 由 TSTR 重新开始。

(3) 在普通模式运行过程中产生异常，并在 PWM 模式 2 重新开始时的运行

在普通模式产生异常，重新设定后，在 PWM 模式 2 重新开始时的说明如图 11.115 所示。

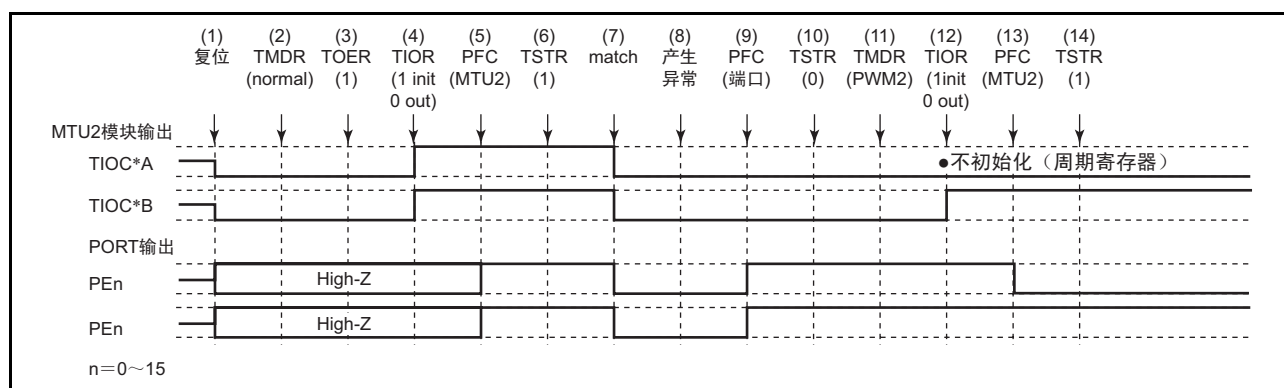


图 11.115 在普通模式产生异常，并在 PWM 模式 2 恢复时

(1) ~ (10) 和图 11.113 通用。

(11) 设定 PWM 模式 2。

(12) 由 TIOR 初始化引脚（在 PWM 模式 2 不初始化周期寄存器的引脚。如果要初始化时，在普通模式初始化后，转移至 PWM 模式 2）。

(13) 由 PFC 设置为 MTU2 输出。

(14) 由 TSTR 重新开始。

【注】 仅可在通道 0 ~ 2 设定 PWM 模式 2，因此，无需设定 TOER。

(4) 在普通模式运行过程中产生异常，并在相位计数模式重新开始时的运行

在普通模式产生异常，重新设定后，在相位计数模式重新开始时的说明如图 11.116 所示。

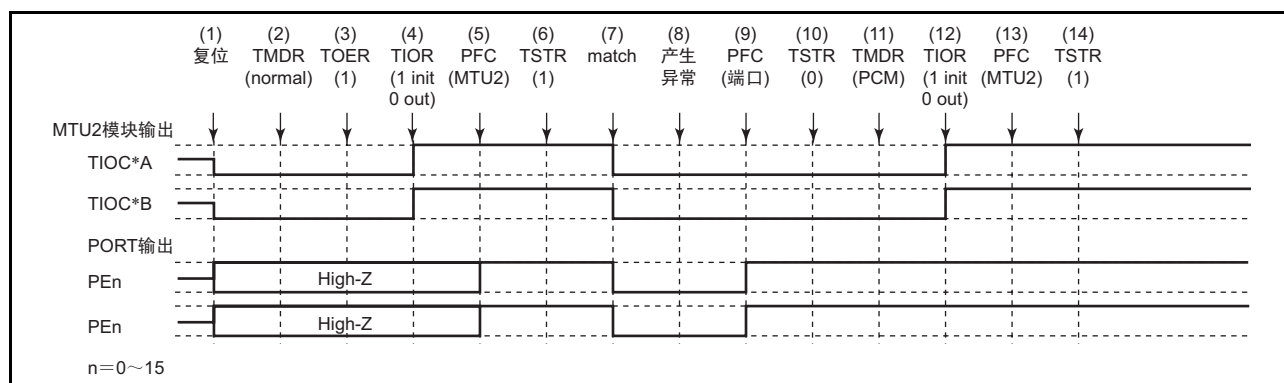


图 11.116 在普通模式产生异常，并在相位计数模式恢复时

(1) ~ (10) 和图 11.113 通用。

(11) 设定相位计数模式。

(12) 由 TIOR 初始化引脚。

(13) 由 PFC 设置为 MTU2 输出。

(14) 由 TSTR 重新开始。

【注】 仅可在通道 1、2 设定相位计数模式，因此，无需设定 TOER。

(5) 在普通模式运行过程中产生异常，并在互补 PWM 模式重新开始时的运行

在普通模式产生异常，重新设定后，在互补 PWM 模式重新开始时的说明如图 11.117 所示。

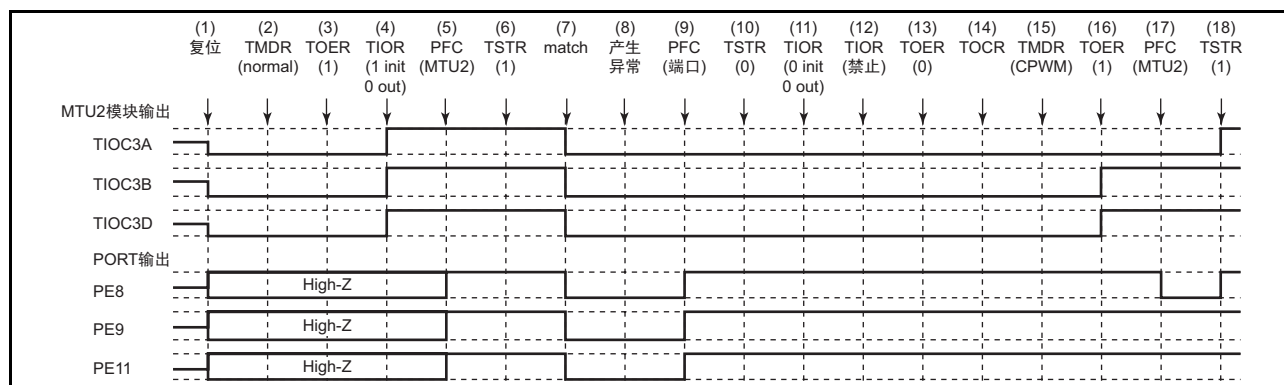


图 11.117 在普通模式产生异常，并在互补 PWM 模式恢复时

(1) ~ (10) 和图 11.113 通用。

(11) 由 TIOR 初始化普通模式的波形生成部分。

(12) 由 TIOR 禁止运行普通模式波形生成部分的运行。

(13) 由 TOER 禁止通道 3、4 的输出。

(14) 由 TOCR 选择互补 PWM 的输出电平和周期输出的允许 / 禁止。

(15) 设定互补 PWM。

(16) 由 TOER 允许通道 3、4 的输出。

(17) 由 PFC 设置为 MTU2 输出。

(18) 由 TSTR 重新开始。

(6) 在普通模式运行过程中产生异常，并在复位同步 PWM 模式重新开始时的运行

在普通模式产生异常，重新设定后，在复位同步 PWM 模式重新开始时的说明如图 11.118 所示。

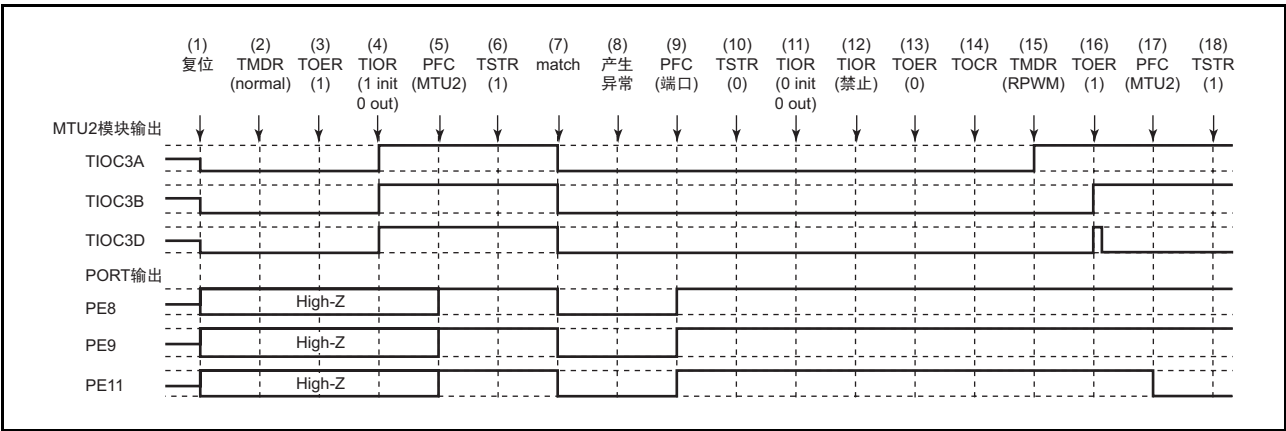


图 11.118 在普通模式产生异常，并在复位同步 PWM 模式恢复时

- (1) ~ (13) 和图 11.113 通用。
- (14) 由 TOCR 选择复位同步 PWM 输出电平和周期输出的允许 / 禁止。
- (15) 设定复位同步 PWM。
- (16) 由 TOER 允许通道 3、4 的输出。
- (17) 由 PFC 设置为 MTU2 输出。
- (18) 由 TSTR 重新开始。

(7) 在 PWM 模式 1 运行过程中产生异常，并在普通模式重新开始时的运行

在 PWM 模式 1 产生异常，重新设定后，在普通模式重新开始时的说明如图 11.119 所示。

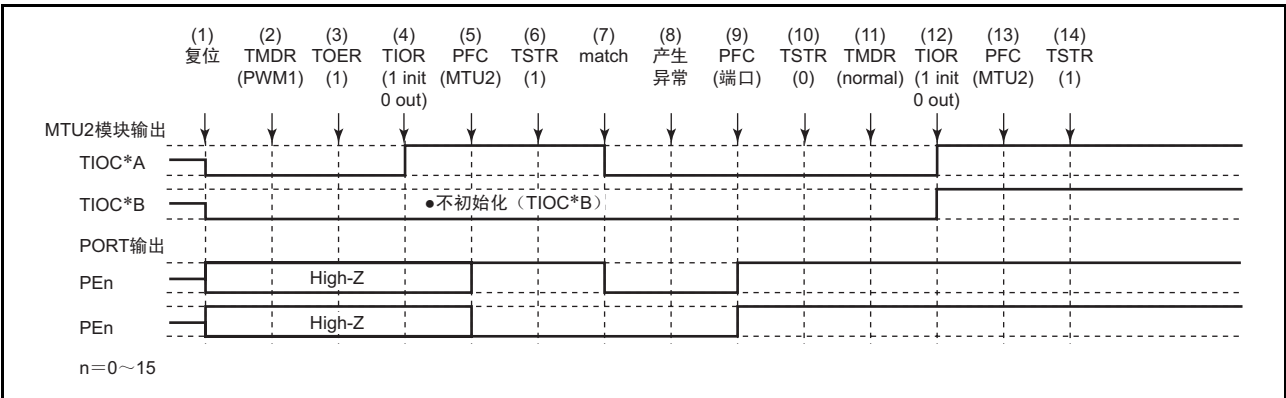


图 11.119 在 PWM 模式 1 产生异常，并在普通模式恢复时

- (1) 由于复位，MTU2 输出为低电平，端口为高阻抗。
- (2) 设定 PWM 模式 1。
- (3) 在通道 3、4 由 TIOR 初始化引脚前，由 TOER 允许输出。
- (4) 必须由 TIOR 初始化引脚（例如，初始输出为高电平，比较匹配时为低电平输出。在 PWM 模式 1 不初始化 TIOC*B）。
- (5) 由 PFC 设置为 MTU2 输出。
- (6) 由 TSTR 开始计数运行。
- (7) 通过产生比较匹配输出低电平。
- (8) 产生异常。
- (9) 由 PFC 设置为端口输出，输出取反的有效电平。
- (10) 由 TSTR 停止计数运行。
- (11) 设定普通模式。
- (12) 由 TIOR 初始化引脚。
- (13) 由 PFC 设置为 MTU2 输出。
- (14) 由 TSTR 重新开始。

(8) 在 PWM 模式 1 运行过程中产生异常，并在 PWM 模式 1 重新开始时的运行

在 PWM 模式 1 产生异常，重新设定后，在 PWM 模式重新开始时的说明如图 11.120 所示。

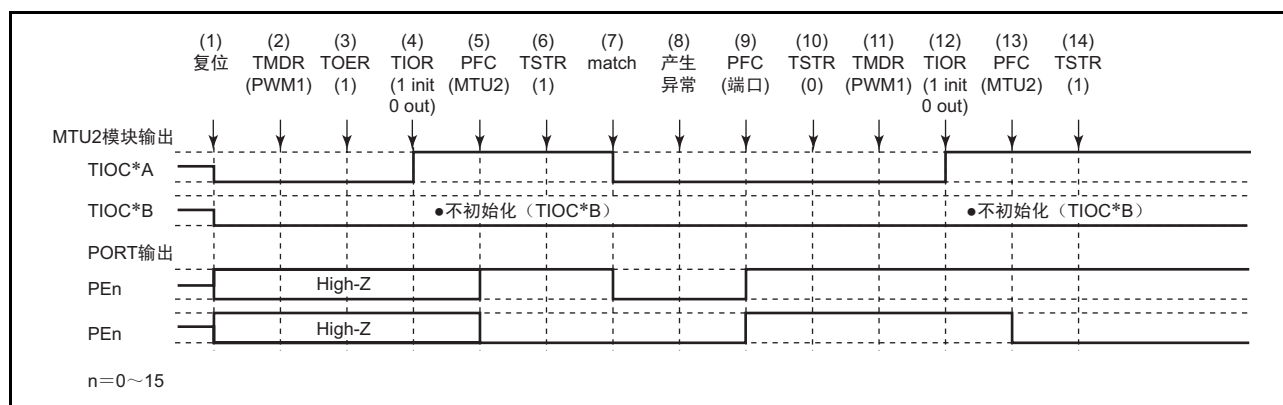


图 11.120 在 PWM 模式 1 产生异常，并在 PWM 模式 1 恢复时

- (1) ~ (10) 和图 11.119 通用。
- (11) 在 PWM 模式 1 重新开始时无需此步骤。
- (12) 由 TIOR 初始化引脚（在 PWM 模式 1，不初始化 TIOC*B）。
- (13) 由 PFC 设置为 MTU2 输出。
- (14) 由 TSTR 重新开始。

(9) 在 PWM 模式 1 运行过程中产生异常，并在 PWM 模式 2 重新开始时的运行

在 PWM 模式 1 产生异常，重新设定后，在 PWM 模式 2 重新开始时的说明如图 11.121 所示。

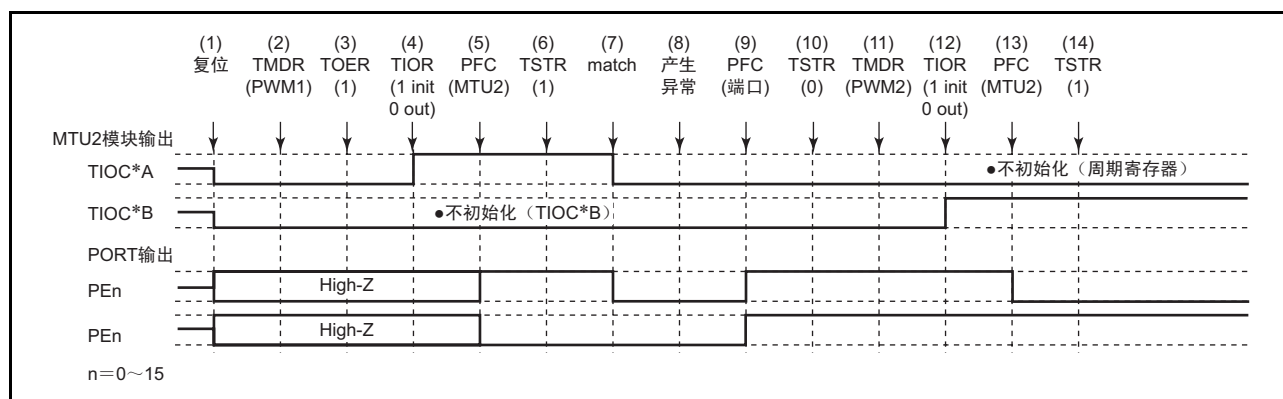


图 11.121 在 PWM 模式 1 产生异常，并在 PWM 模式 2 恢复时

(1) ~ (10) 和图 11.119 通用。

(11) 设定 PWM 模式 2。

(12) 由 TIOR 初始化引脚 (在 PWM 模式 2，不初始化周期寄存器的引脚)。

(13) 由 PFC 设置为 MTU2 输出。

(14) 由 TSTR 重新开始。

【注】 仅可在通道 0 ~ 2 设定 PWM 模式 2，因此，无需设定 TOER。

(10) 在 PWM 模式 1 运行过程中产生异常，并在相位计数模式重新开始时的运行

在 PWM 模式 1 产生异常，重新设定后，在相位计数模式重新开始时的说明如图 11.122 所示。

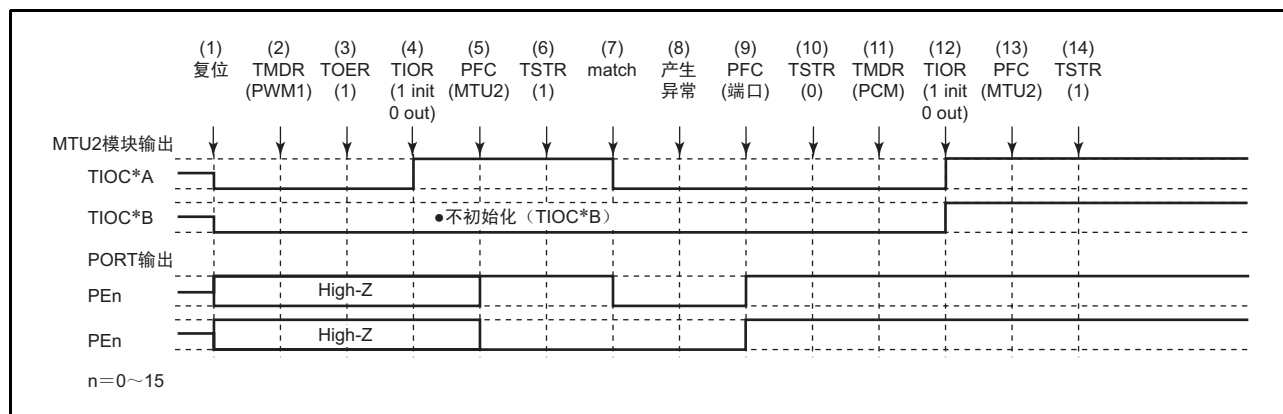


图 11.122 在 PWM 模式 1 产生异常，并在相位计数模式恢复时

(1) ~ (10) 和图 11.119 通用。

(11) 设定相位计数模式。

(12) 由 TIOR 初始化引脚。

(13) 由 PFC 设置为 MTU2 输出。

(14) 由 TSTR 重新开始。

【注】 仅可在通道 1、2 设定相位计数模式，因此，无需设定 TOER。

(11) 在 PWM 模式 1 运行过程中产生异常，并在互补 PWM 模式重新开始时的运行

在 PWM 模式 1 产生异常，重新设定后，在互补 PWM 模式重新开始时的说明如图 11.123 所示。

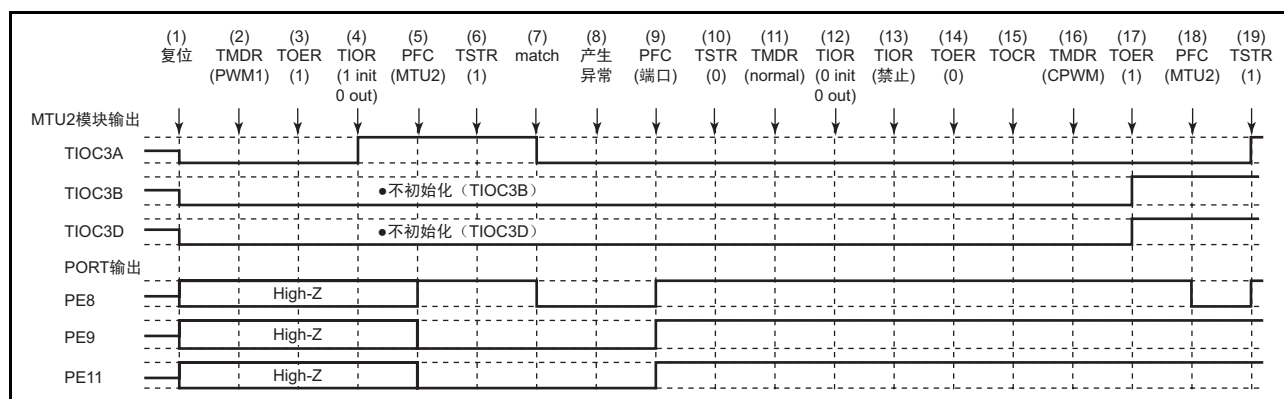


图 11.123 在 PWM 模式 1 产生异常，并在互补 PWM 模式恢复时

(1) ~ (10) 和图 11.119 通用。

(11) 为了初始化波形生成部分，必须设定普通模式。

(12) 由 TIOR 初始化 PWM 模式 1 的波形生成部分。

(13) 由 TIOR 禁止运行 PWM 模式 1 的波形生成部分的运行。

(14) 由 TOER 禁止通道 3、4 的输出。

(15) 由 TOCR 选择互补 PWM 的输出电平和周期输出的允许 / 禁止。

(16) 设定互补 PWM。

(17) 由 TOER 允许通道 3、4 的输出。

(18) 由 PFC 设置为 MTU2 输出。

(19) 由 TSTR 重新开始。

(12) 在 PWM 模式 1 运行过程中产生异常，并在复位同步 PWM 模式重新开始时的运行

在 PWM 模式 1 产生异常，重新设定后，在复位同步 PWM 模式重新开始时的说明如图 11.124 所示。

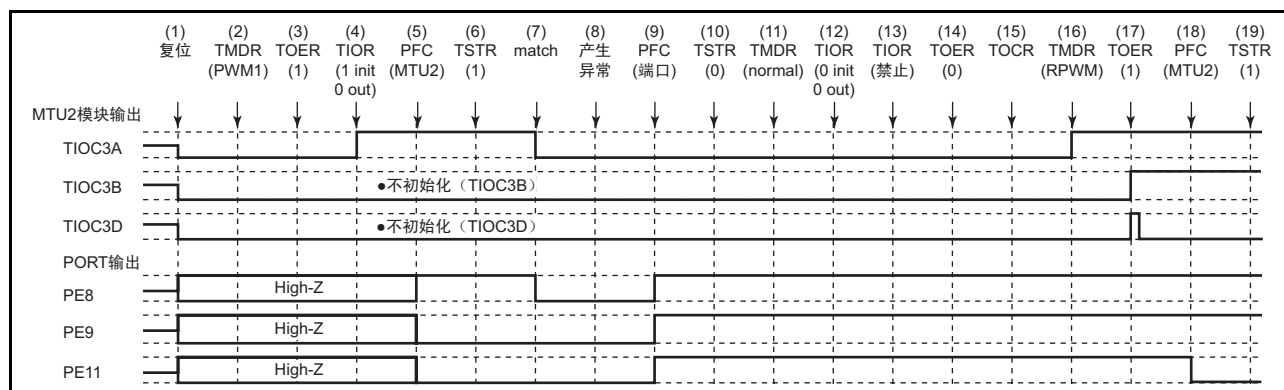


图 11.124 在 PWM 模式 1 产生异常，并在复位同步 PWM 模式恢复时

(1) ~ (14) 和图 11.123 通用。

(15) 由 TOCR 选择复位同步 PWM 的输出电平和周期输出的允许 / 禁止。

(16) 设定复位同步 PWM。

(17) 由 TOER 允许通道 3、4 的输出。

(18) 由 PFC 设置为 MTU2 输出。

(19) 由 TSTR 重新开始。

(13) 在 PWM 模式 2 运行过程中产生异常，并在普通模式重新开始时的运行

在 PWM 模式 2 产生异常，重新设定后，在普通模式重新开始时的说明如图 11.125 所示。

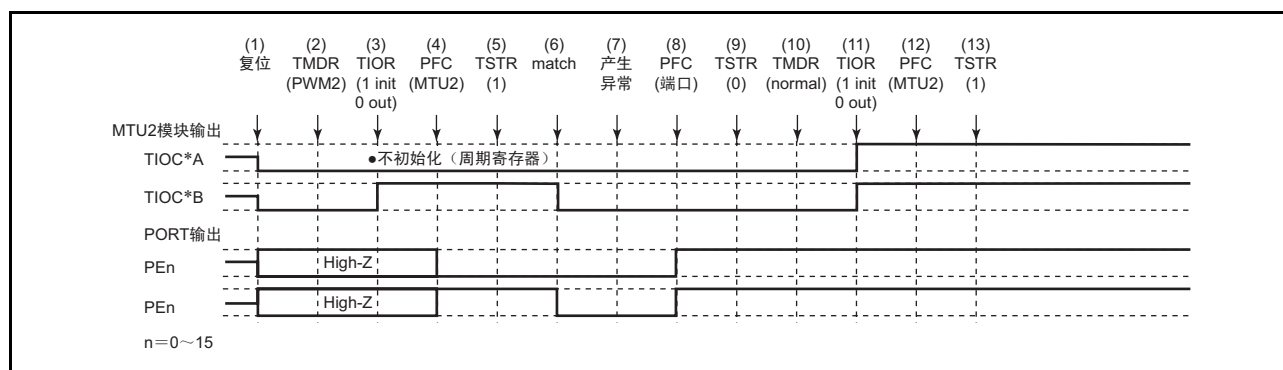


图 11.125 在 PWM 模式 2 产生异常，并在普通模式恢复时

- (1) 由于复位，MTU2 输出为低电平，端口为高阻抗。
- (2) 设定 PWM 模式 2。
- (3) 由 TIOR 初始化引脚（例如，初始输出为高电平，比较匹配时为低电平输出。在 PWM 模式 2，不初始化周期寄存器的引脚。此例子中，TIOC*A 为周期寄存器）。
- (4) 由 PFC 设置为 MTU2 输出。
- (5) 由 TSTR 开始计数运行。
- (6) 通过产生比较匹配输出低电平。
- (7) 产生异常。
- (8) 由 PFC 设置为端口输出，输出取反的有效电平。
- (9) 由 TSTR 停止运行计数。
- (10) 设定普通模式。
- (11) 由 TIOR 初始化引脚。
- (12) 由 PFC 设置为 MTU2 输出。
- (13) 由 TSTR 重新开始。

(14) 在 PWM 模式 2 运行过程中产生异常，并在 PWM 模式 1 重新开始时的运行

在 PWM 模式 2 产生异常，重新设定后，在 PWM 模式 1 重新开始时的说明如图 11.126 所示。

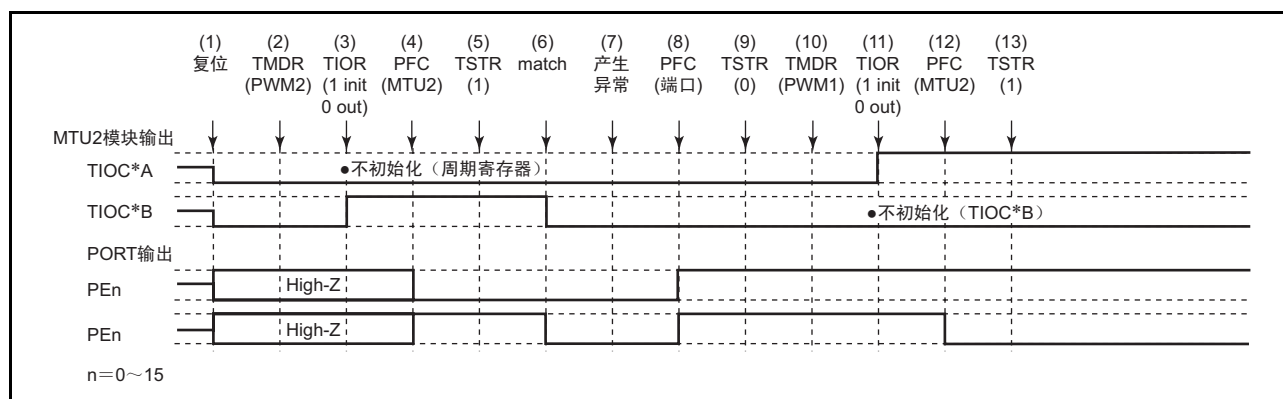


图 11.126 在 PWM 模式 2 产生异常，并在 PWM 模式 1 恢复时

(1) ~ (9) 和图 11.125 通用。

(10) 设定 PWM 模式 1。

(11) 由 TIOR 初始化引脚 (在 PWM 模式 1, 不初始化 TIOC*B)。

(12) 由 PFC 设置为 MTU2 输出。

(13) 由 TSTR 重新开始。

(15) 在 PWM 模式 2 运行过程中产生异常，并在 PWM 模式 2 重新开始时的运行

在 PWM 模式 2 产生异常，重新设定后，在 PWM 模式 2 重新开始时的说明如图 11.127 所示。

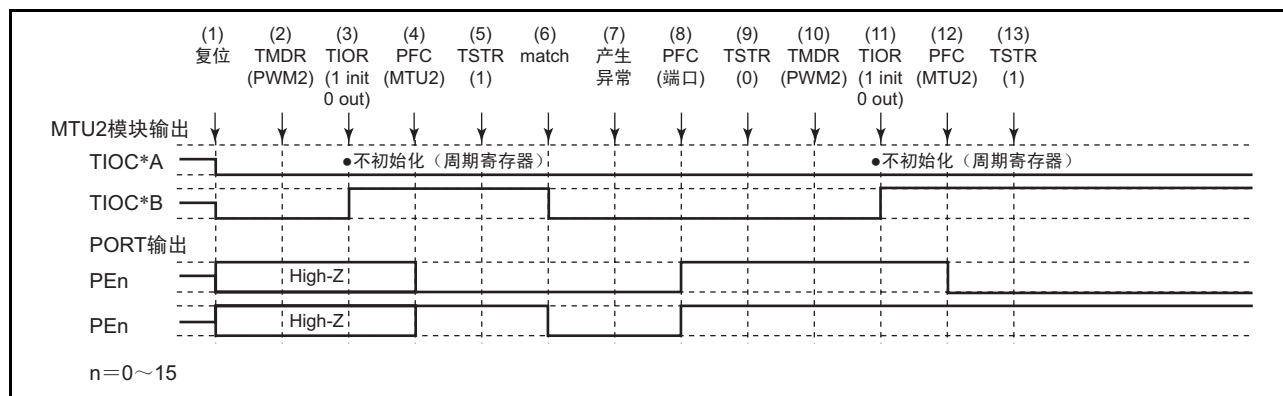


图 11.127 在 PWM 模式 2 产生异常，并在 PWM 模式 2 恢复时

(1) ~ (9) 和图 11.125 通用。

(10) 在 PWM 模式 2 重新开始时无需此步骤。

(11) 由 TIOR 初始化引脚 (在 PWM 模式 2, 不初始化周期寄存器的引脚)。

(12) 由 PFC 设置为 MTU2 输出。

(13) 由 TSTR 重新开始。

(16) 在 PWM 模式 2 运行过程中产生异常，并在相位计数模式重新开始时的运行

在 PWM 模式 2 产生异常，重新设定后，在相位计数模式重新开始时的说明如图 11.128 所示。

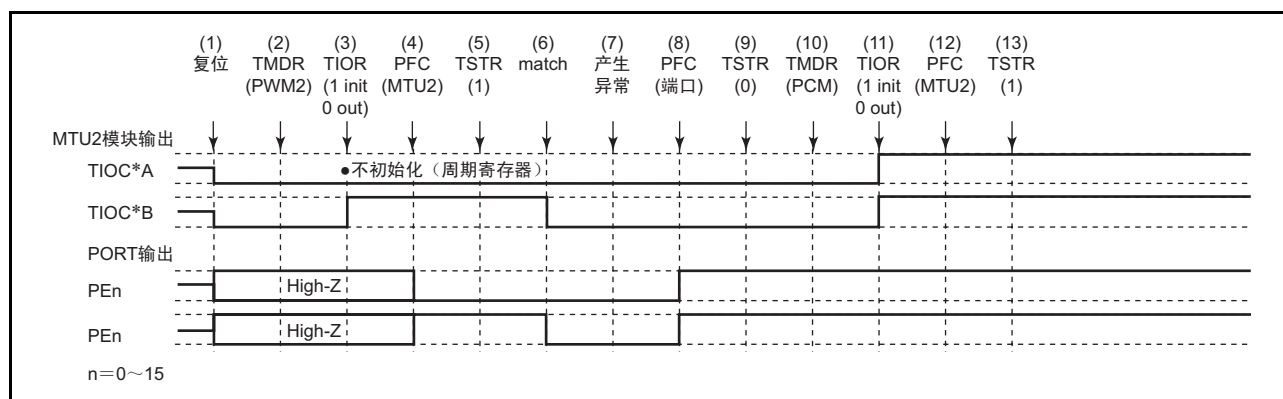


图 11.128 在 PWM 模式 2 产生异常，并在相位计数模式恢复时

- (1) ~ (9) 和图 11.125 通用。
- (10) 设定相位计数模式。
- (11) 由 TIOR 初始化引脚。
- (12) 由 PFC 设置为 MTU2 输出。
- (13) 由 TSTR 重新开始。

(17) 在相位计数模式运行过程中产生异常，并在普通模式重新开始时的运行

在相位计数模式产生异常，重新设定后，在普通模式重新开始时的说明如图 11.129 所示。

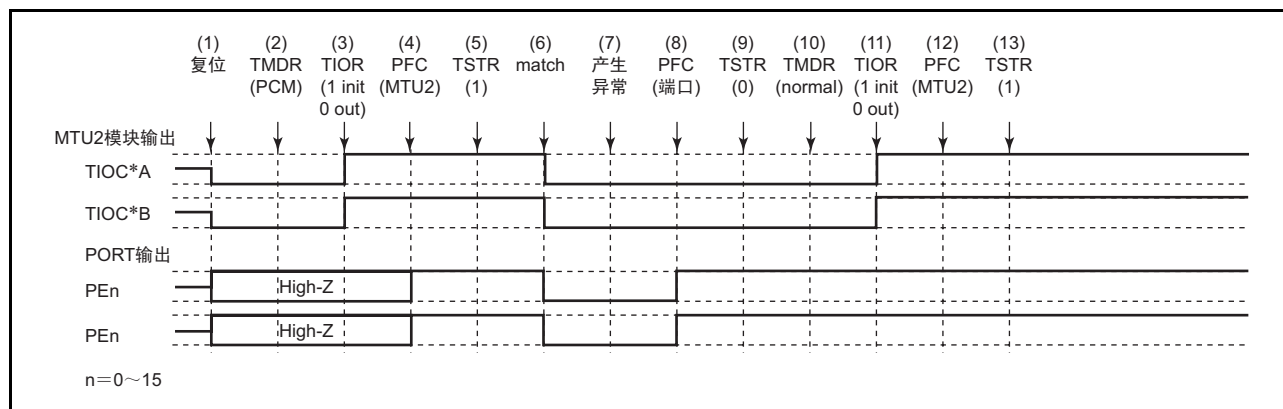


图 11.129 在相位计数模式产生异常，并在普通模式恢复时

- (1) 由于复位，MTU2 输出为低电平，端口为高阻抗。
- (2) 设定相位计数模式。
- (3) 由 TIOR 初始化引脚（例如，初始输出为高电平，比较匹配时为低电平输出）。
- (4) 由 PFC 设置为 MTU2 输出。
- (5) 由 TSTR 开始计数运行。
- (6) 通过产生比较匹配输出低电平。
- (7) 产生异常。
- (8) 由 PFC 设置为端口输出，输出取反的有效电平。
- (9) 由 TSTR 停止计数运行。
- (10) 设定普通模式。
- (11) 由 TIOR 初始化引脚。
- (12) 由 PFC 设置为 MTU2 输出。
- (13) 由 TSTR 重新开始。

(18) 在相位计数模式运行过程中产生异常，并在 PWM 模式 1 重新开始时的运行

在相位计数模式产生异常，重新设定后，在 PWM 模式 1 重新开始时的说明如图 11.130 所示。

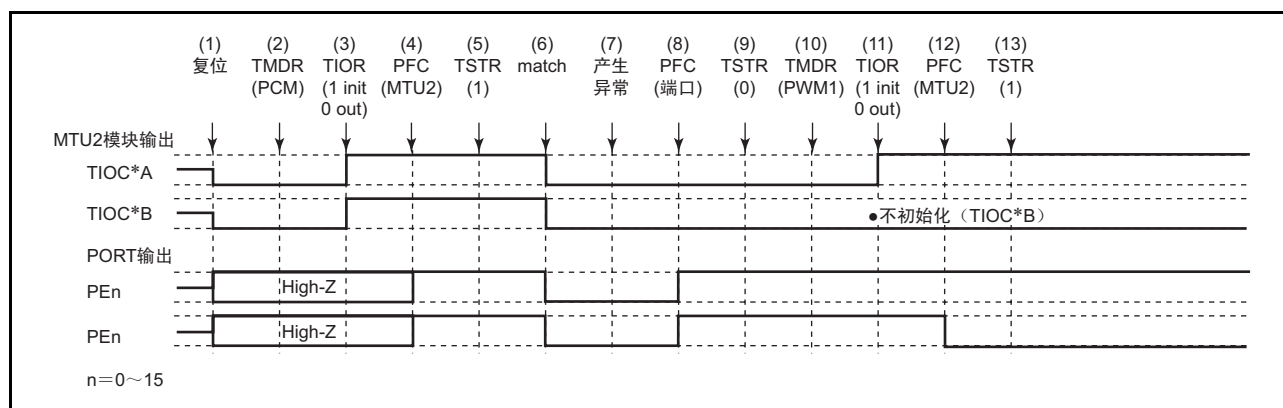


图 11.130 在相位计数模式产生异常，并在 PWM 模式 1 恢复时

- (1) ~ (9) 和图 11.129 通用。
- (10) 设定 PWM 模式 1。
- (11) 由 TIOR 初始化引脚（在 PWM 模式 1，不初始化 TIOC*B）。
- (12) 由 PFC 设置为 MTU2 输出。
- (13) 由 TSTR 重新开始。

(19) 在相位计数模式运行过程中产生异常，并在 PWM 模式 2 重新开始时的运行

在相位计数模式产生异常，重新设定后，在 PWM 模式 2 重新开始时的说明如图 11.131 所示。

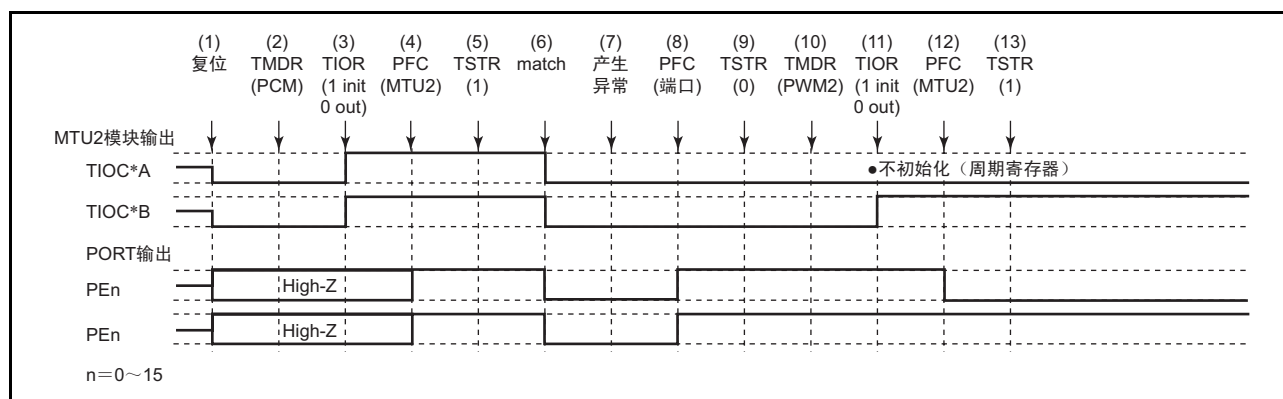


图 11.131 在相位计数模式产生异常，并在 PWM 模式 2 恢复时

(1) ~ (9) 和图 11.129 通用。

(10) 设定 PWM 模式 2。

(11) 由 TIOR 初始化引脚 (在 PWM 模式 2，不初始化周期寄存器的引脚)。

(12) 由 PFC 设置为 MTU2 输出。

(13) 由 TSTR 重新开始。

(20) 在相位计数模式运行过程中产生异常，并在相位计数模式重新开始时的运行

在相位计数模式产生异常，重新设定后，在相位计数模式重新开始时的说明如图 11.132 所示。

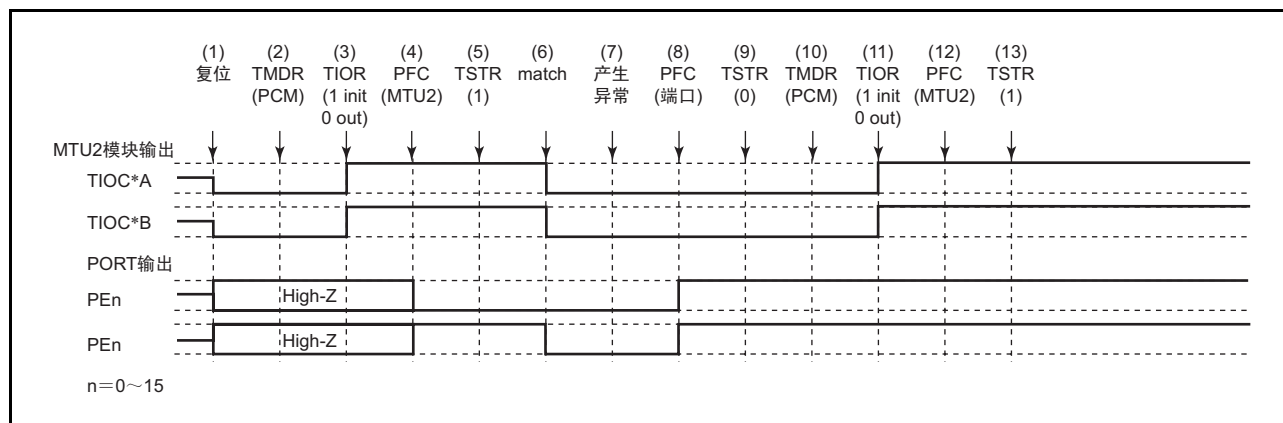


图 11.132 在相位计数模式产生异常，并在相位计数模式恢复时

(1) ~ (9) 和图 11.129 通用。

(10) 在相位计数模式重新开始时，无需此步骤。

(11) 由 TIOR 初始化引脚。

(12) 由 PFC 设置为 MTU2 输出。

(13) 由 TSTR 重新开始。

(21) 在互补 PWM 模式运行过程中产生异常，并在普通模式重新开始时的运行

在互补 PWM 模式产生异常，重新设定后，在普通模式重新开始时的说明如图 11.133 所示。

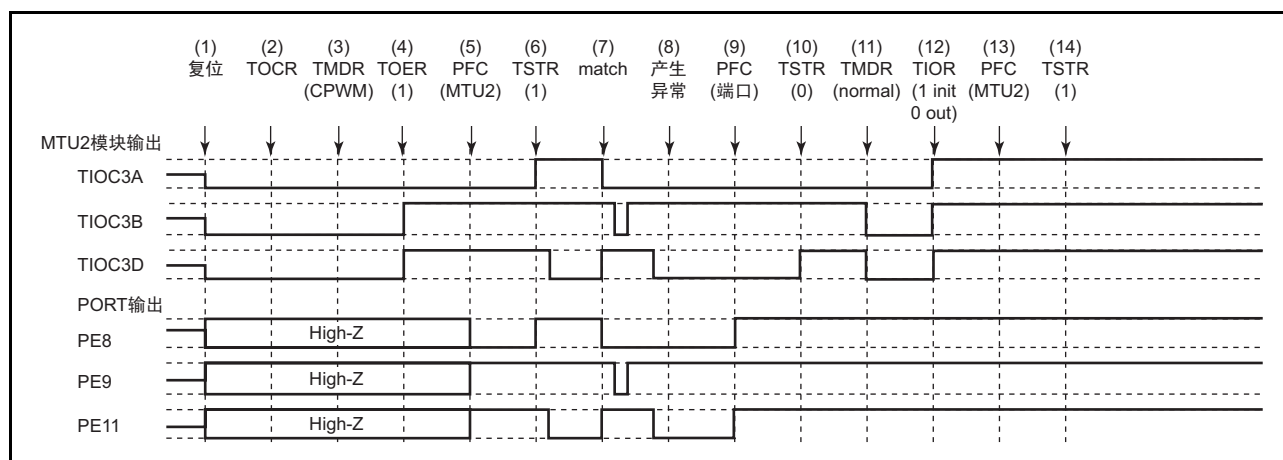


图 11.133 在互补 PWM 模式产生异常，并在普通模式恢复时

- (1) 由于复位，MTU2 输出为低电平，端口为高阻抗。
- (2) 由 TOCR 选择互补 PWM 的输出电平和周期输出的允许 / 禁止。
- (3) 设定互补 PWM。
- (4) 由 TOER 允许通道 3、4 的输出。
- (5) 由 PFC 设置为 MTU2 输出。
- (6) 由 TSTR 开始计数运行。
- (7) 通过产生比较匹配输出互补 PWM 波形。
- (8) 产生异常。
- (9) 由 PFC 设置为端口输出，输出反转的有效电平。
- (10) 由 TSTR 停止计数运行 (MTU2 输出为互补 PWM 输出的初始值)。
- (11) 设定普通模式 (MTU2 输出为低电平)。
- (12) 由 TIOR 初始化引脚。
- (13) 由 PFC 设置为 MTU2 输出。
- (14) 由 TSTR 重新开始。

(22) 在互补 PWM 模式运行过程中产生异常，并在 PWM 模式 1 重新开始时的运行

在互补 PWM 模式产生异常，重新设定后，在 PWM 模式 1 重新开始时的说明如图 11.134 所示。

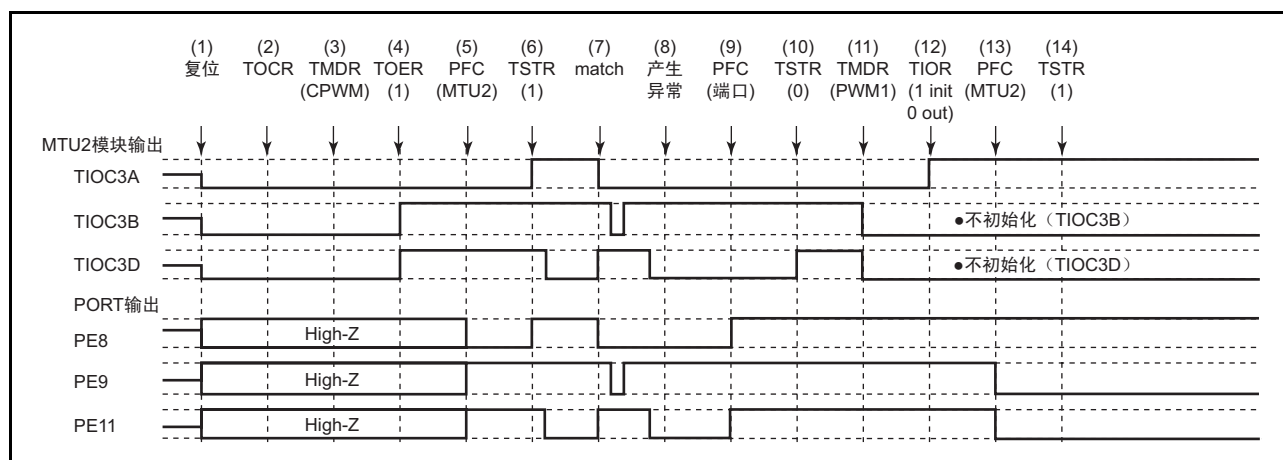


图 11.134 在互补 PWM 模式产生异常，并在 PWM 模式 1 恢复时

(1) ~ (10) 和图 11.133 通用。

(11) 设定 PWM 模式 1 (MTU2 输出为低电平)。

(12) 由 TIOC3A 初始化引脚 (在 PWM 模式 1, 不初始化 TIOC3B)。

(13) 由 PFC 设置为 MTU2 输出。

(14) 由 TSTR 重新开始。

(23) 在互补 PWM 模式运行过程中产生异常，并在互补 PWM 模式重新开始时的运行

在互补 PWM 模式产生异常，重新设定后，在互补 PWM 模式重新开始时的说明如图 11.135 所示 (从停止计数器时的周期、占空比设定值重新时)。

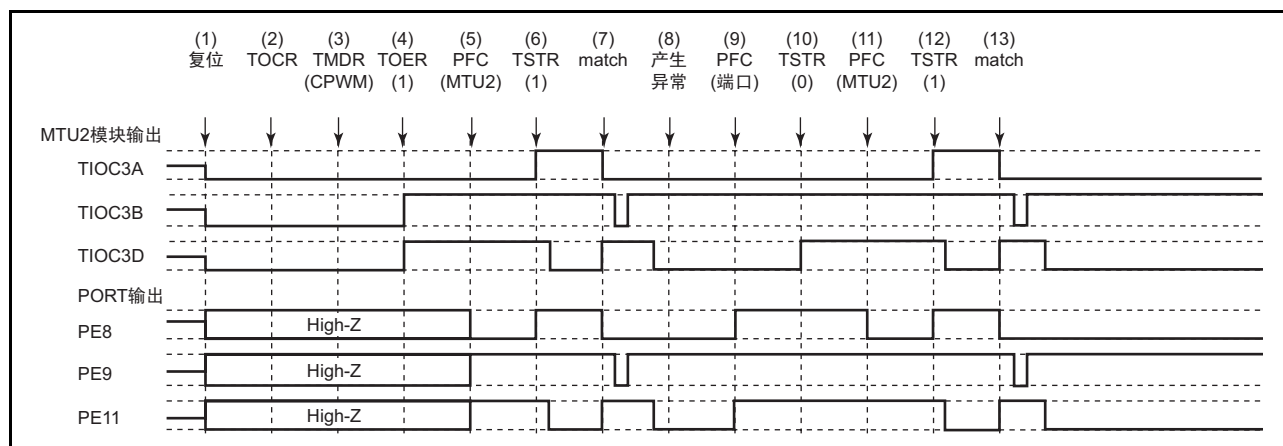


图 11.135 在互补 PWM 模式产生异常，并在互补 PWM 模式恢复时

(1) ~ (10) 和图 11.133 通用。

(11) 由 PFC 设置为 MTU2 输出。

(12) 由 TSTR 重新开始。

(13) 通过产生比较匹配输出互补 PWM 波形。

(24) 在互补 PWM 模式运行过程中产生异常，并在互补 PWM 模式重新开始时的运行

在互补 PWM 模式产生异常，重新设定后，并在互补 PWM 模式重新开始时的说明如图 11.136 所示（通过新的周期、占空比设定值重新时）。

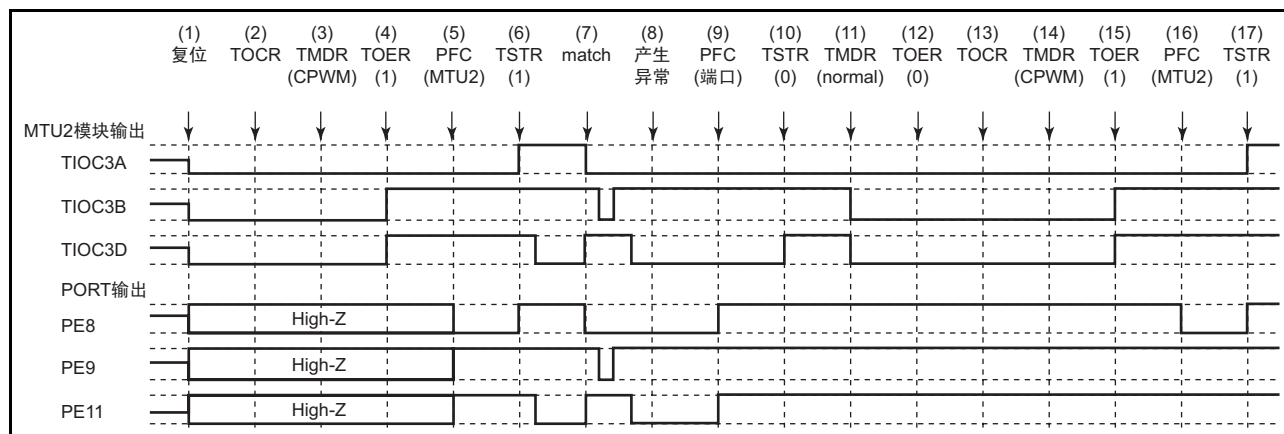


图 11.136 在互补 PWM 模式产生异常，并在互补 PWM 模式恢复时

(1) ~ (10) 和图 11.133 通用。

(11) 设定普通模式和新的设定值 (MTU2 输出为低电平)。

(12) 由 TOER 禁止通道 3、4 的输出。

(13) 由 TOCR 选择互补 PWM 模式的输出电平和周期输出的允许 / 禁止。

(14) 设定互补 PWM。

(15) 由 TOER 允许通道 3、4 的输出。

(16) 由 PFC 设置为 MTU2 输出。

(17) 由 TSTR 重新开始。

(25) 在互补 PWM 模式运行过程中产生异常，并在复位同步 PWM 模式重新开始时的运行

在互补 PWM 模式产生异常，重新设定后，在复位同步 PWM 模式重新开始时的说明如图 11.137 所示。

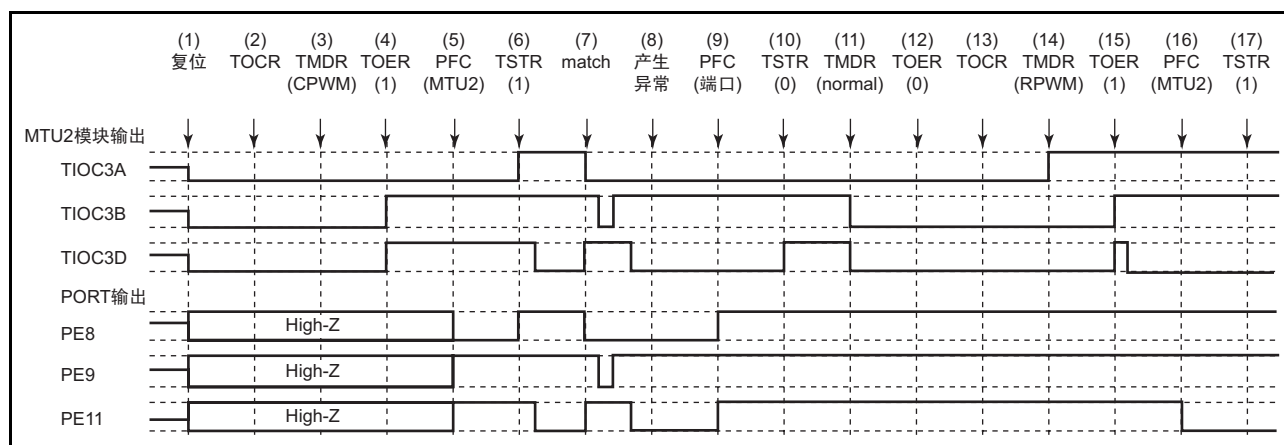


图 11.137 在互补 PWM 模式产生异常，并在复位同步 PWM 模式恢复时

- (1) ~ (10) 和图 11.133 通用。
- (11) 设定普通模式 (MTU2 输出为低电平)。
- (12) 由 TOER 禁止通道 3、4 的输出。
- (13) 由 TOCR 选择复位同步 PWM 模式的输出电平和周期输出的允许 / 禁止。
- (14) 设定复位同步 PWM。
- (15) 由 TOER 允许通道 3、4 的输出。
- (16) 由 PFC 设置为 MTU2 输出。
- (17) 由 TSTR 重新开始。

(26) 在复位同步 PWM 模式运行过程中产生异常，并在普通模式重新开始时的运行

在复位同步 PWM 模式产生异常，重新设定后，在普通模式重新开始时的说明如图 11.138 所示。

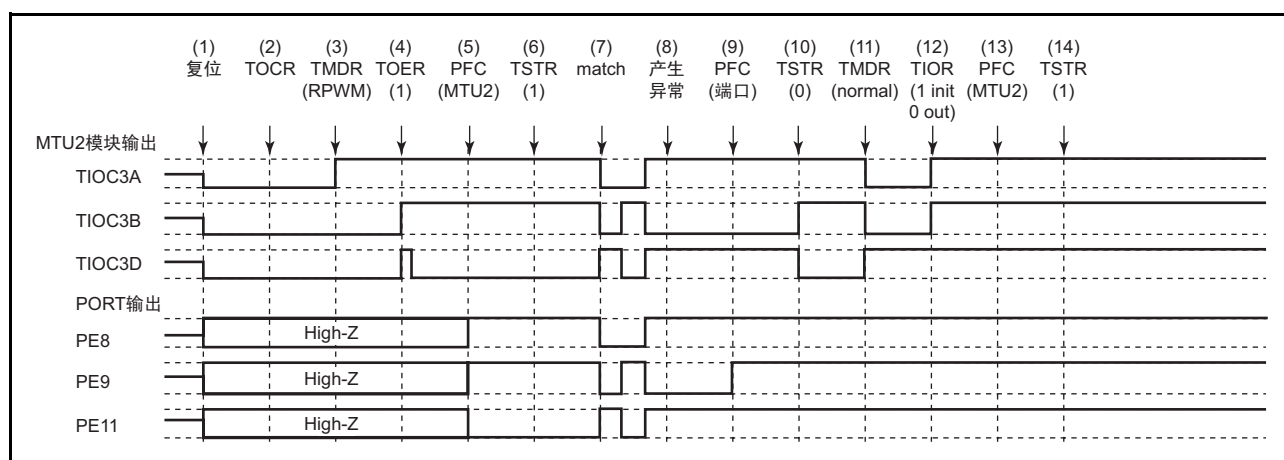


图 11.138 在复位同步 PWM 模式产生异常，并在普通模式恢复时

- (1) 由于复位，MTU2 输出为低电平，端口为高阻抗状。
- (2) 由 TOCR 选择复位同步 PWM 的输出电平和周期输出的允许 / 禁止。
- (3) 设定复位同步 PWM。
- (4) 由 TOER 允许通道 3、4 的输出。
- (5) 由 PFC 作设置为 MTU2 输出。
- (6) 由 TSTR 开始计数运行。
- (7) 通过产生比较匹配输出复位同步 PWM 波形。
- (8) 产生异常。
- (9) 由 PFC 设置为端口输出，输出取反的有效电平。
- (10) 由 TSTR 停止计数运行 (MTU2 输出为复位同步 PWM 输出的初始值)。
- (11) 设定普通模式 (MTU2 输出的正相为低电平，反相为高电平)。
- (12) 由 TIOR 初始化引脚。
- (13) 由 PFC 作设置为 MTU2 输出。
- (14) 由 TSTR 重新开始。

(27) 在复位同步 PWM 模式运行过程中产生异常，并在 PWM 模式 1 重新开始时的运行

在复位同步 PWM 模式产生异常，重新设定后，在 PWM 模式 1 重新开始时的说明如图 11.139 所示。

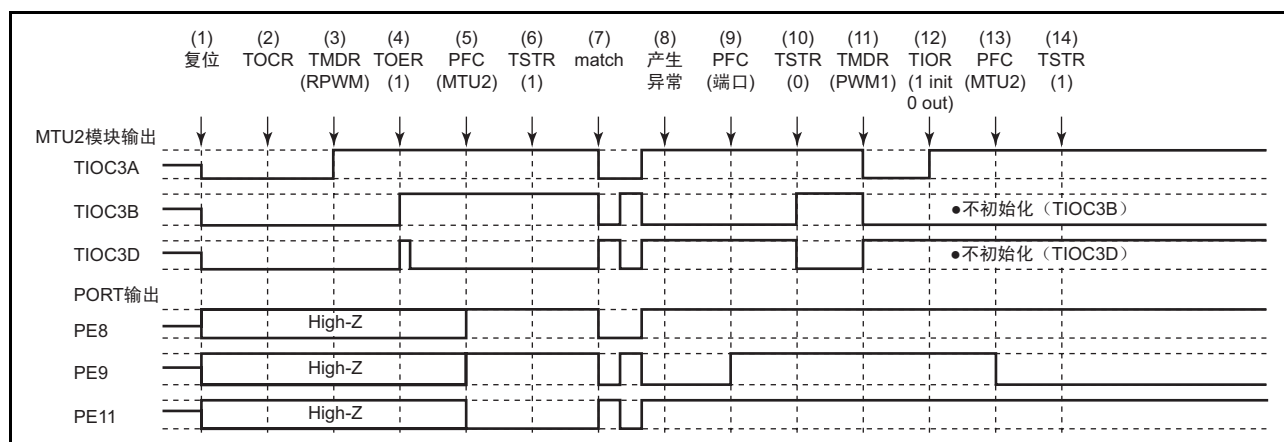


图 11.139 在复位同步 PWM 模式产生异常，并在 PWM 模式 1 恢复时

(1) ~ (10) 和图 11.138 通用。

(11) 设定 PWM 模式 1 (MTU2 输出的正相为低电平，反相为高电平)。

(12) 由 TIOR 初始化引脚 (在 PWM 模式 1, 不初始化 TIOC*B)。

(13) 由 PFC 设置为 MTU2 输出。

(14) 由 TSTR 重新开始。

(28) 在复位同步 PWM 模式运行过程中产生异常，并在互补 PWM 模式重新开始时的运行

在复位同步 PWM 模式产生异常，重新设定后，在互补 PWM 模式重新开始时的说明如图 11.140 所示。

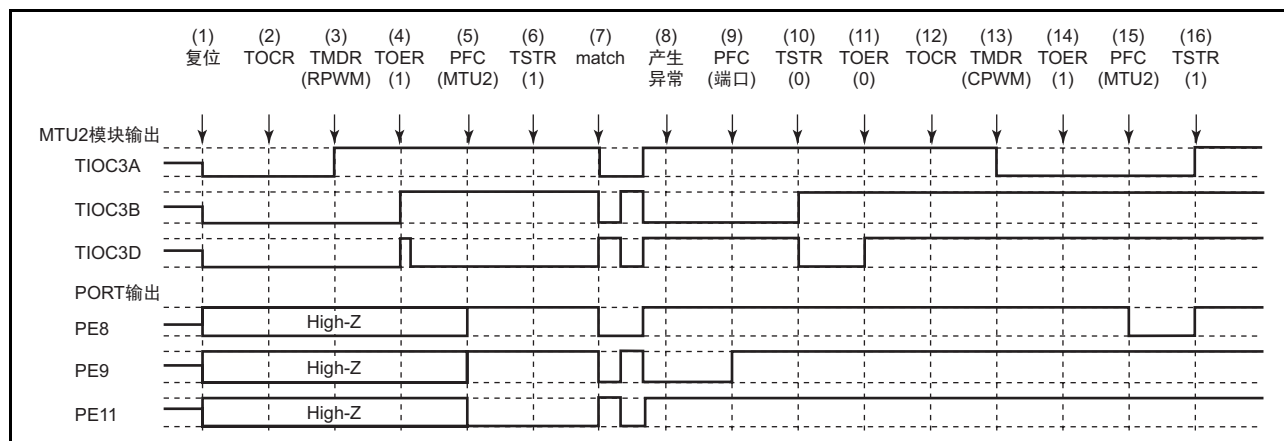


图 11.140 在复位同步 PWM 模式产生异常，并在互补 PWM 模式恢复时

(1) ~ (10) 和图 11.138 通用。

(11) 由 TOER 禁止通道 3、4 的输出。

(12) 由 TOCR 选择互补 PWM 的输出电平和周期输出的允许 / 禁止。

(13) 设定互补 PWM (MTU2 的周期输出引脚为低电平)。

(14) 由 TOER 允许通道 3、4 的输出。

(15) 由 PFC 设置为 MTU2 输出。

(16) 由 TSTR 重新开始。

(29) 在复位同步 PWM 模式运行过程中产生异常，并在复位同步 PWM 模式重新开始时的运行

在复位同步 PWM 模式产生异常，重新设定后，在复位同步 PWM 模式重新开始时的说明如图 11.141 所示。

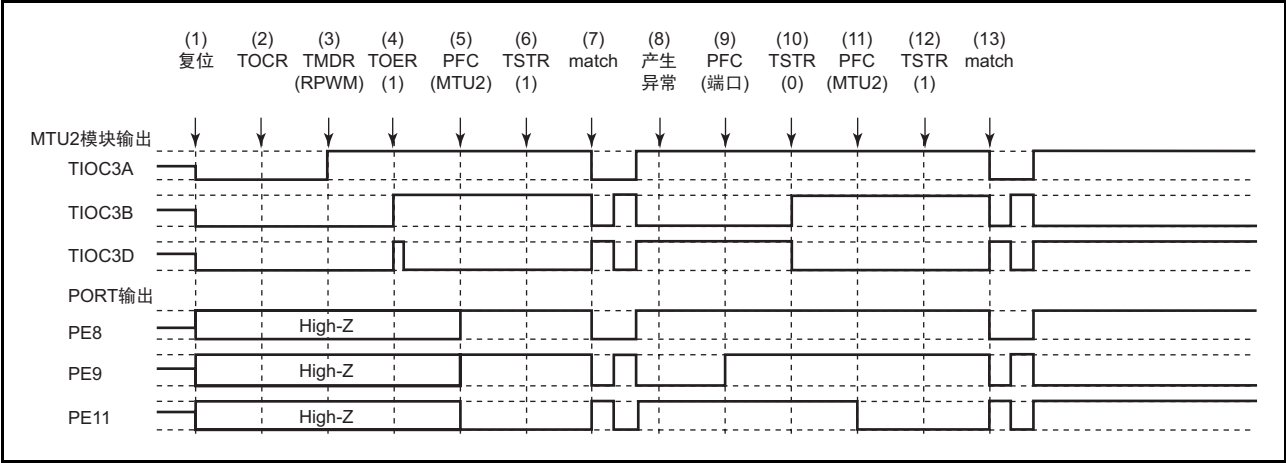


图 11.141 在复位同步 PWM 模式产生异常，并在复位同步 PWM 模式恢复时

- (1) ~ (10) 和图 11.138 通用。
- (11) 由 PFC 设置为 MTU2 输出。
- (12) 由 TSTR 重新开始。
- (13) 通过产生比较匹配输出复位同步 PWM 波形。

第 12 章 比较匹配定时器 (CMT)

本 LSI 内置 2 个通道的 16 位定时器构成的比较匹配定时器 (CMT)。CMT 有 16 位计数器, 可在每个已设定的周期产生中断。

12.1 特点

- 2 个通道可独立选择 4 种计数器输入时钟
可选择 4 种内部时钟 ($P\phi/8$ 、 $P\phi/32$ 、 $P\phi/128$ 、 $P\phi/512$)
- 比较匹配时, 通过 DMAC 的设定, 可选择产生 DMA 传送请求或中断请求
- 不使用 CMT 时, 为了降低功耗, 可停止向 CMT 提供时钟, 停止运行。

CMT 框图如图 12.1 所示。

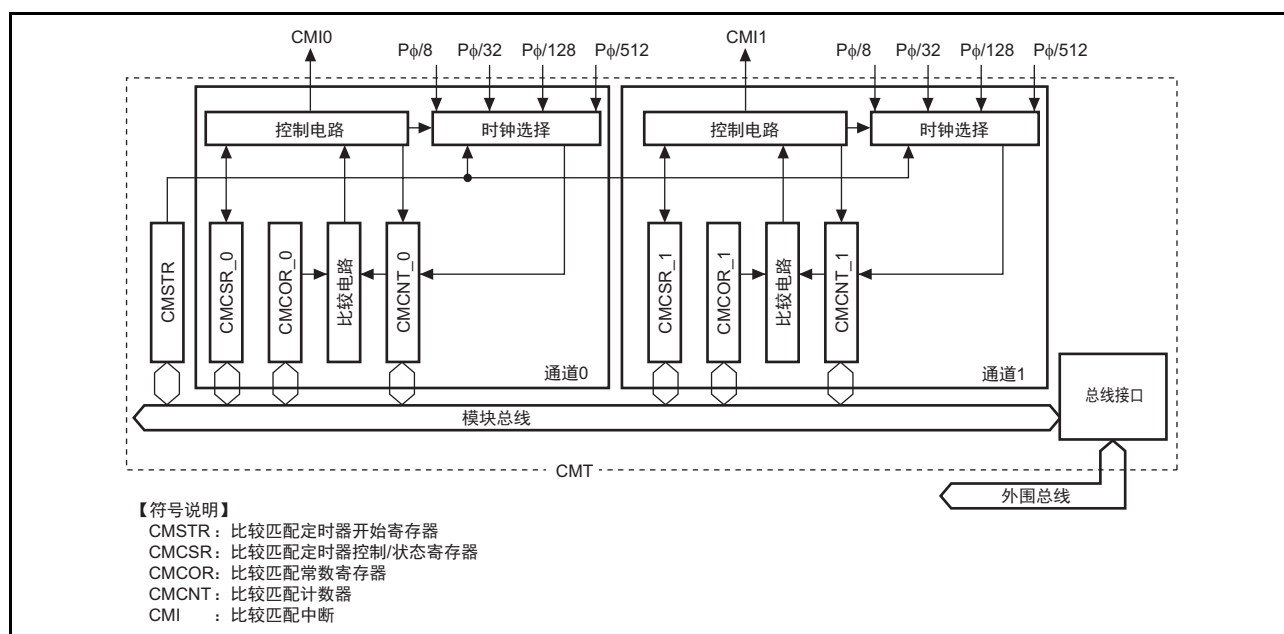


图 12.1 CMT 框图

12.2 寄存器说明

CMT 有以下寄存器。

表 12.1 寄存器结构

通道	寄存器名称	简称	R/W	初始值	地址	存取长度
通用	比较匹配定时器开始寄存器	CMSTR	R/W	H'0000	H'FFFE0000	16
0	比较匹配定时器控制 / 状态寄存器 _0	CMCSR_0	R/W	H'0000	H'FFFE0002	16
	比较匹配计数器 _0	CMCNT_0	R/W	H'0000	H'FFFE0004	8、16
	比较匹配常数寄存器 _0	CMCOR_0	R/W	H'FFFF	H'FFFE0006	8、16
1	比较匹配定时器控制 / 状态寄存器 _1	CMCSR_1	R/W	H'0000	H'FFFE0008	16
	比较匹配计数器 _1	CMCNT_1	R/W	H'0000	H'FFFE000A	8、16
	比较匹配常数寄存器 _1	CMCOR_1	R/W	H'FFFF	H'FFFE000C	8、16

12.2.1 比较匹配定时器开始寄存器（CMSTR）

CMSTR 为 16 位寄存器，选择比较匹配计数器（CMCNT）的运行 / 停止。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	STR1	STR0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

位	位名称	初始值	R/W	说 明
15 ~ 2	—	均为 0	R	保留位 读取值、写入值总是为 0。
1	STR1	0	R/W	计数开始 1 选择比较匹配计数器 _1 的运行 / 停止。 0: CMCNT_1 停止计数 1: CMCNT_1 开始计数
0	STR0	0	R/W	计数开始 0 选择比较匹配计数器 _0 的运行 / 停止。 0: CMCNT_0 停止计数 1: CMCNT_0 开始计数

12.2.2 比较匹配定时器控制 / 状态寄存器（CMCSR）

CMCSR 为 16 位寄存器，表示产生比较匹配，设定允许 / 禁止中断及计数器输入时钟。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	CMF	CMIE	-	-	-	-	CKS[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/(W)*	R/W	R	R	R	R	R/W	R/W

【注】* 为了清除标志，仅可在读取1后写入0。

位	位名称	初始值	R/W	说 明
15 ~ 8	—	均为 0	R	保留位 读取值、写入值总是为 0。
7	CMF	0	R(W)*	比较匹配标志 本标志表示 CMCNT 和 CMCOR 的值是否匹配。 0: CMCNT 和 CMCOR 的值不匹配 [清除条件] • 读取 CMF=1 后，对 CMF 写入 0 时 1: CMCNT 和 CMCOR 的值匹配
6	CMIE	0	R/W	比较匹配中断允许 CMCNT 和 CMCOR 的值匹配时（CMF=1），选择允许或禁止产生比较匹配中断（CMI）。 0: 禁止比较匹配中断（CMI） 1: 允许比较匹配中断（CMI）
5 ~ 2	—	均为 0	R	保留位 读取值、写入值总是为 0。
1、0	CKS[1:0]	00	R/W	时钟选择 从分频外围时钟（Pφ）的 4 种内部时钟，选择输入 CMCNT 的时钟。 CMSTR 的 STR 位置 1 时，CMCNT 通过 CKS[1:0] 位选择的时钟开始计数。 00: Pφ/8 01: Pφ/32 10: Pφ/128 11: Pφ/512

12.2.3 比较匹配计数器 (CMCNT)

CMCNT 为 16 位寄存器，用作递增计数器。由 CMCSR 的 CKS[1:0] 位选择计数器输入时钟，CMSTR 的 STR 位置 1 时，CMCNT 通过所选的时钟开始计数。CMCNT 的值和比较匹配常数寄存器 (CMCOR) 的值匹配时，CMCNT 清除为 H'0000，CMCSR 的 CMF 标志置 1。

将比较匹配定时器开始寄存器 (CMSTR) 相应通道的计数开始位从 1 清 0 时，初始化为 H'0000。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

12.2.4 比较匹配常数寄存器 (CMCOR)

CMCOR 为 16 位寄存器，设定与 CMCNT 比较匹配前的时间。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
初始值:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

12.3 运行说明

12.3.1 期间计数运行

内部时钟由 CMCSR 的 CKS[1:0] 位选择，将 CMSTR 的 STR 位置 1 时，CMCNT 通过所选的时钟开始递增计数。CMCNT 和 CMCOR 的值匹配时，CMCNT 清除为 H'0000，CMCSR 的 CMF 标志置 1。此时，如果 CMCSR 寄存器的 CMIE 位设定为 1，则请求比较匹配中断 (CMI)。CMCNT 从 H'0000 重新开始递增计数。

比较匹配计数器运行如图 12.2 所示。

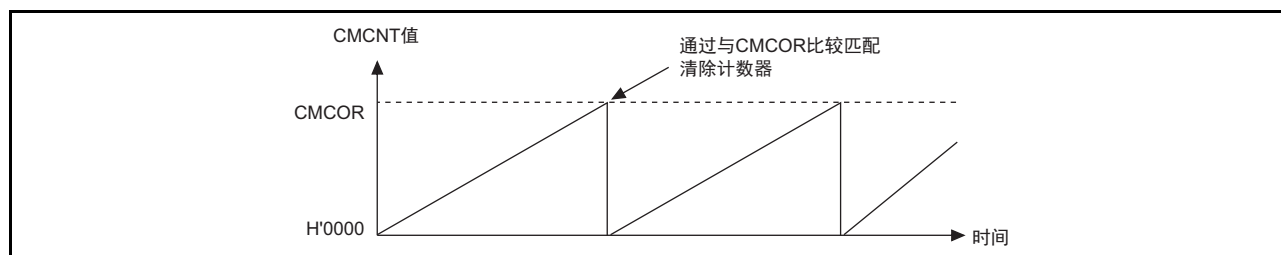


图 12.2 计数器运行

12.3.2 CMCNT 计数时序

可由 CMCSR 的 CKS[1:0] 位, 从分频外围时钟 ($P\phi$) 所得到的 4 种时钟 ($P\phi/8$ 、 $P\phi/32$ 、 $P\phi/128$ 、 $P\phi/512$) 中任选一种。其时序如图 12.3 所示。

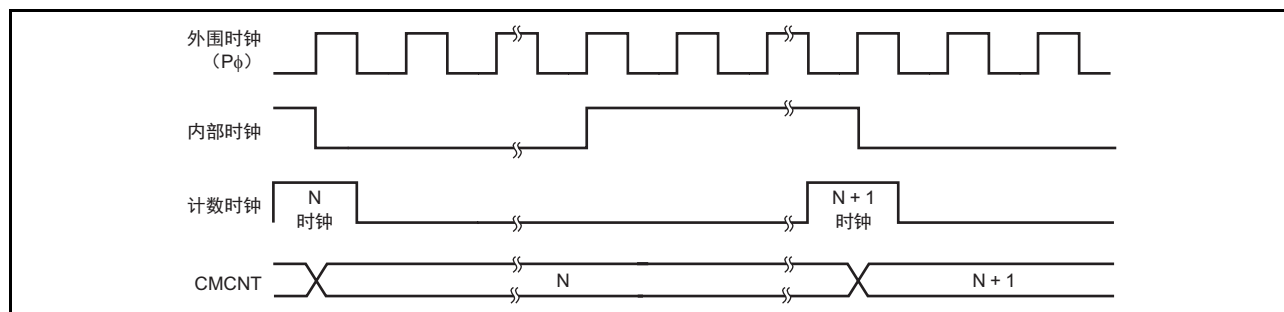


图 12.3 计数时序

12.4 中断

12.4.1 中断源与 DMA 传送请求

CMT 的各通道都有比较匹配中断, 分别分配独立的向量地址。比较匹配标志 (CMF) 和中断允许位 (CMIE) 均置 1 时, 输出相应的中断请求。由中断请求启动 CPU 中断时, 设定中断控制器可更改通道间的优先顺序。详情参阅“第 6 章 中断控制器 (INTC)”。

必须在用户异常处理程序清除 CMF 位。否则, 会再次产生中断。请求比较匹配中断时, 通过设定直接存储器存取控制器 (DMAC) 可启动 DMAC。此时, 不对 CPU 产生中断。未执行 DMAC 启动设定时, 对 CPU 产生中断请求。通过 DMAC 传送数据时, CMF 位自动清除。

12.4.2 比较匹配标志的置位时序

CMCOR 与 CMCNT 匹配时, 在匹配的最终状态 (CMCNT 的值更新为 H'0000 的时序) 产生比较匹配信号, CMCSR 的 CMF 位置 1。即 CMCOR 与 CMCNT 匹配后, 如果不输入 CMCNT 的计数器时钟, 则不产生比较匹配信号。CMF 位的置位时序如图 12.4 所示。

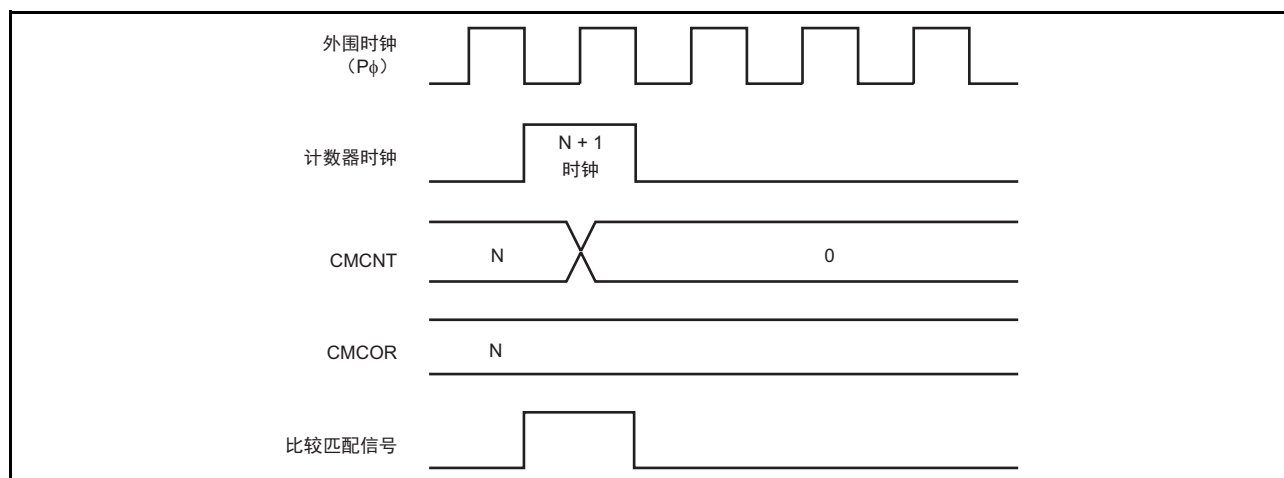


图 12.4 CMF 置位时序

12.4.3 比较匹配标志的清除时序

读取 CMF=1 后写入 0，可清除 CMCSR 的 CMF 位。但 DMAC 启动后，并在传送完数据时，CMF 位自动清 0。

12.5 使用时的注意事项

12.5.1 CMCNT 的写入与比较匹配的竞争

如果在 CMCNT 计数器的写入周期的 T2 状态产生比较匹配信号，则不向 CMCNT 计数器写入，而优先清除 CMCNT 计数器。此时序如图 12.5 所示。

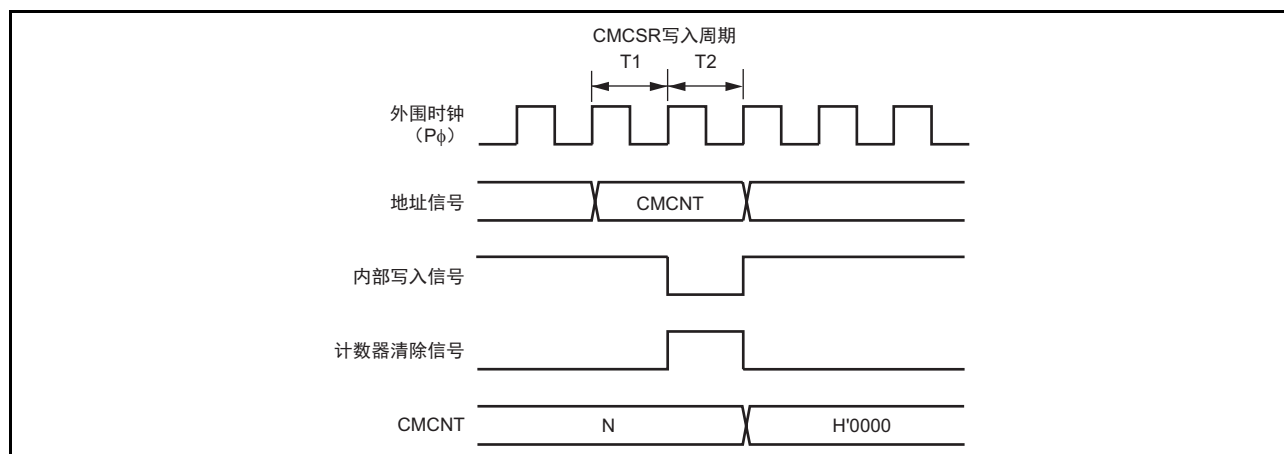


图 12.5 CMCNT 的写入与比较匹配的竞争

12.5.2 CMCNT 的字写入与递增计数的竞争

在 CMCNT 计数器的字写入周期的 T2 状态，即使产生递增计数，也不递增计数，而是优先写入计数器。此时序如图 12.6 所示。

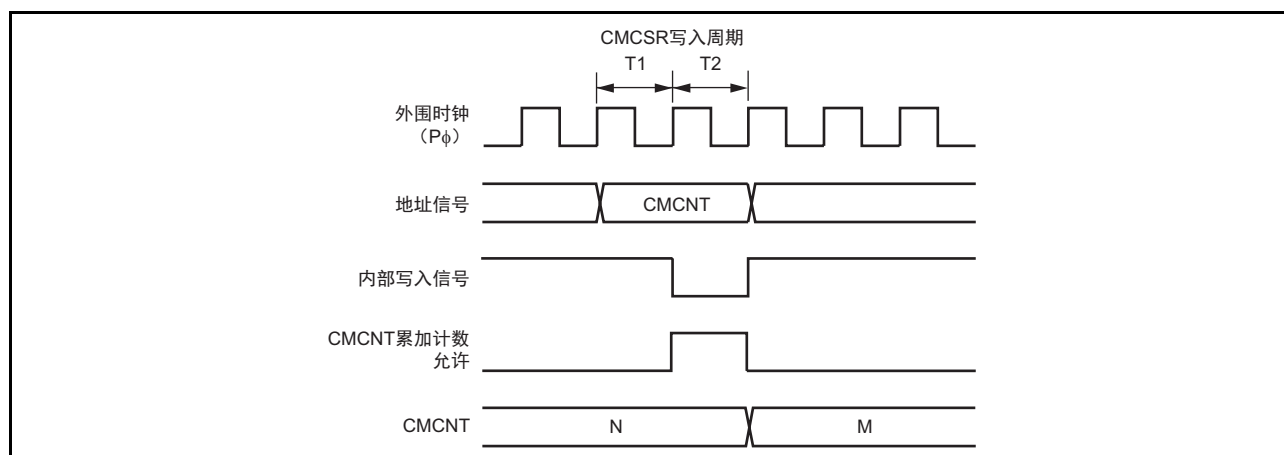


图 12.6 CMCNT 的字写入与递增计数的竞争

12.5.3 CMCNT 的字节写入与递增计数的竞争

在 CMCNT 的字节写入周期的 T2 状态，即使产生递增计数，已写入的写入数据也不递增递增计数，而是优先写入计数器。未写入的字节数据也不递增计数，仍为写入前的内容。

在 CMCNTH 写入周期的 T2 状态产生递增计数时的时序，如图 12.7 所示。

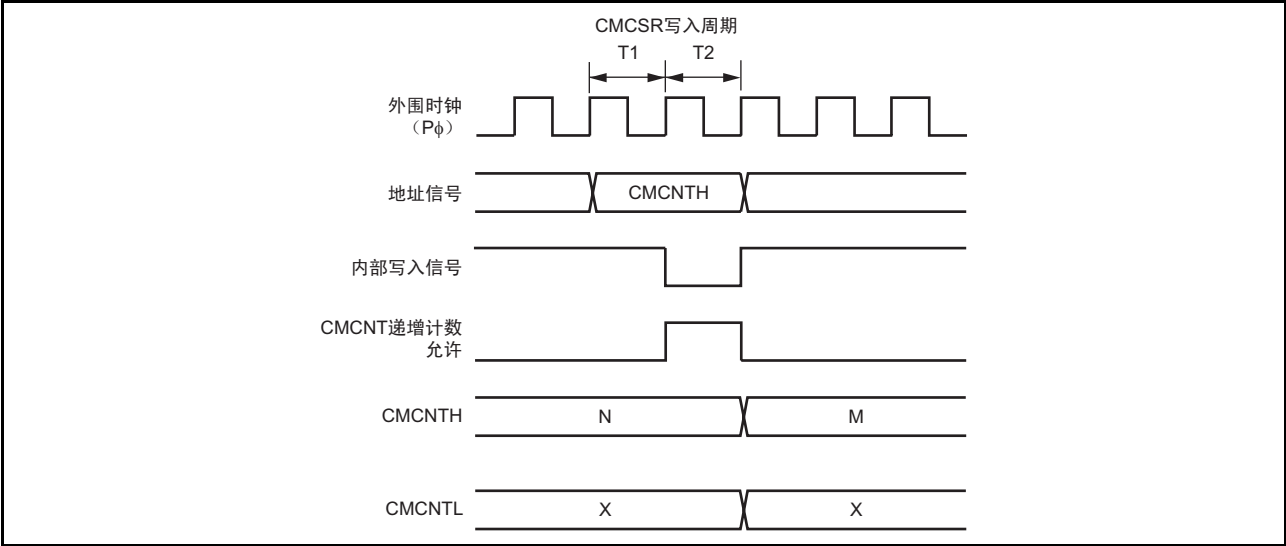


图 12.7 CMCNT 的字节写入与递增计数的竞争

第 13 章 看门狗定时器 (WDT)

本 LSI 内置看门狗定时器 (WDT)，因系统失控等导致计数器的值不可改写而上溢时，向外部输出上溢信号 ($\overline{\text{WDTOVF}}$)。同时，可产生本 LSI 的内部复位信号。

WDT 为 1 个通道定时器，用于计数在解除软件待机模式及频率更改时暂时的待机状态的时钟振荡稳定时间，也可用作普通的看门狗定时器或间隔定时器。

13.1 特点

- 可用于确保时钟振荡稳定时间
在用于解除软件待机模式及时钟频率更改时的暂时待机状态。
- 可在看门狗定时器模式与间隔定时器模式之间转换
- 看门狗定时器模式时，输出信号 $\overline{\text{WDTOVF}}$
计数器上溢时，向外部输出 $\overline{\text{WDTOVF}}$ 信号。此时，可选择是否同时复位本 LSI 内部。此内部复位可选择上电复位或手动复位。
- 间隔定时器模式时，产生中断
由计数器上溢，产生间隔定时器中断。
- 可选择 8 种计数器输入时钟
可从分频外围时钟得到的 8 种时钟 ($P\phi \times 1 \sim \times 1/16384$) 选择

WDT 框图如图 13.1 所示。

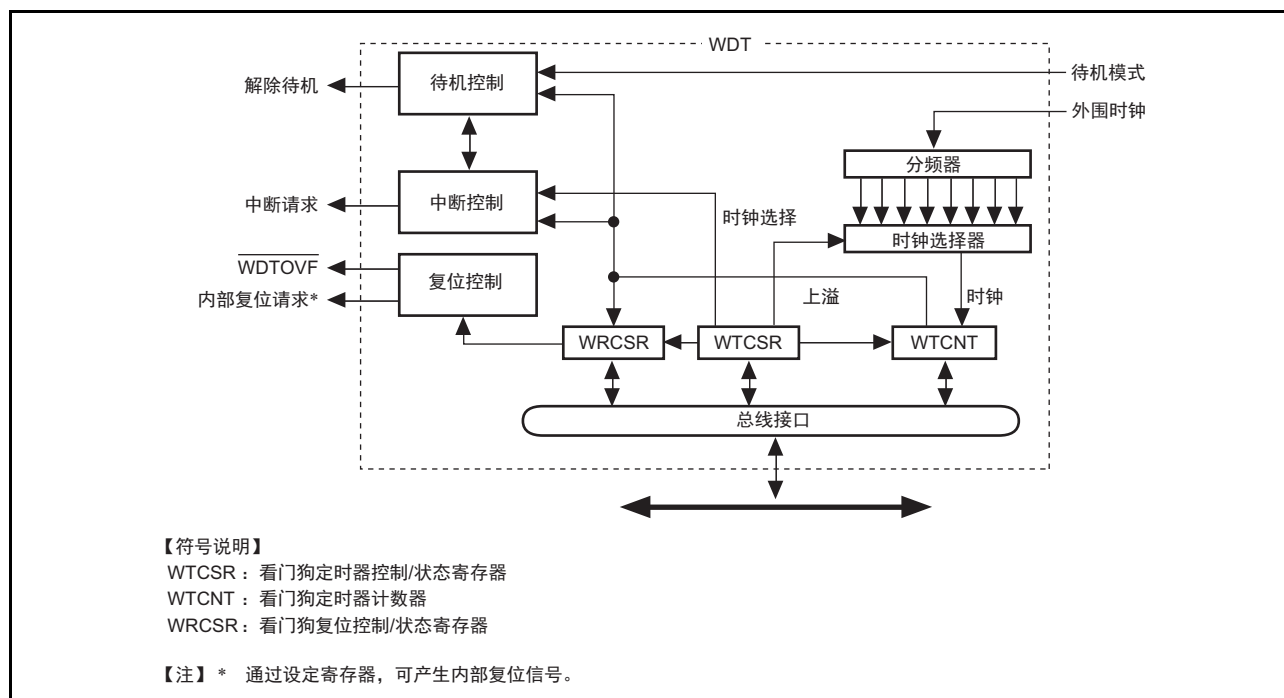


图 13.1 WDT 框图

13.2 输入 / 输出引脚

WDT 的引脚如表 13.1 所示。

表 13.1 引脚结构

名称	引脚名称	输入 / 输出	功 能
看门狗定时器上溢	WDTOVF	输出	输出看门狗定时器模式时的计数器上溢信号

13.3 寄存器说明

WDT 有以下寄存器。

表 13.2 寄存器结构

寄存器名称	简称	R/W	初始值	地址	存取长度
看门狗定时器计数器	WTCNT	R/W	H'00	H'FFFE0002	16*
看门狗定时器控制 / 状态寄存器	WTCSR	R/W	H'18	H'FFFE0000	16*
看门狗复位控制 / 状态寄存器	WRCSR	R/W	H'1F	H'FFFE0004	16*

【注】 * 有关存取长度，详情参阅“13.3.4 寄存器存取时的注意事项”。

13.3.1 看门狗定时器计数器（WTCNT）

WTCNT 为 8 位可读取 / 写入的寄存器，是通过所选的时钟递增计数的计数器。上溢时，在看门狗定时器模式产生看门狗定时器上溢信号（WDTOVF），且在间隔定时器模式产生中断。

必须将高位字节设置为 H'5A，以字长度写入 WTCNT。以字节长度读取 WTCNT。

【注】 为了防止错误改写，本寄存器的写入方法与一般的寄存器不同，详情参阅“13.3.4 寄存器存取时的注意事项”。

位:	7	6	5	4	3	2	1	0
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

13.3.2 看门狗定时器控制 / 状态寄存器 (WTCSR)

WTCSR 为 8 位可读取 / 写入的寄存器，由选择计数时钟的位、上溢标志位及允许位构成。

用于计数解除软件待机模式的时钟振荡稳定时间时，计数器上溢后仍保持该值。

必须将高位字节设置为 H'A5，以字长度写入 WTCSR。以字节长度读取 WTCSR。

【注】 为了防止错误改写，本寄存器的写入方法与一般的寄存器不同。详情参阅“13.3.4 寄存器存取时的注意事项”。

位:	7	6	5	4	3	2	1	0
	IOVF	WT/IT	TME	-	-	CKS[2:0]		
初始值:	0	0	0	1	1	0	0	0
R/W:	R/(W)	R/W	R/W	R	R	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明																		
7	IOVF	0	R/(W)	间隔定时器上溢 表示在间隔定时器模式 WTCNT 上溢。在看门狗定时器模式，此位不置位。 0: 无上溢 1: 在间隔定时器模式 WTCNT 上溢 [清除条件] • 读取 IOVF 后写入 0																		
6	WT/ $\overline{\text{IT}}$	0	R/W	定时器模式选择 指定用作看门狗定时器或间隔定时器。 0: 间隔定时器模式 1: 看门狗定时器模式 【注】在看门狗定时器模式，WTCNT 上溢时，向外部输出 $\overline{\text{WDTOVF}}$ 信号。 WDT 运行过程中改写 WT/ $\overline{\text{IT}}$ 时，可能无法正确执行递增计数																		
5	TME	0	R/W	定时器允许 设定开始或停止定时器运行。在软件待机模式或时钟频率更改时使用 WDT，此位必须置 0。 0: 禁止定时器运行 停止递增计数，保持 WTCNT 的值。 1: 允许定时器运行																		
4、3	—	均为 1	R	保留位 读取值、写入值总是为 1。																		
2 ~ 0	CKS[2:0]	000	R/W	时钟选择 从分频外围时钟（P ϕ ）后得到的 8 种时钟中，选择用于 WTCNT 计数的时钟。括号内的上溢周期为外围时钟（P ϕ ）=33MHz 时的值。 <table><tr><td>时钟分频比</td><td>上溢周期</td></tr><tr><td>000: 1 × Pϕ</td><td>（7.7μs）</td></tr><tr><td>001: 1/64×Pϕ</td><td>（500μs）</td></tr><tr><td>010: 1/128×Pϕ</td><td>（1.0ms）</td></tr><tr><td>011: 1/256×Pϕ</td><td>（2.0ms）</td></tr><tr><td>100: 1/512×Pϕ</td><td>（4.0ms）</td></tr><tr><td>101: 1/1024×Pϕ</td><td>（8.0ms）</td></tr><tr><td>110: 1/4096×Pϕ</td><td>（32ms）</td></tr><tr><td>111: 1/16384×Pϕ</td><td>（128ms）</td></tr></table> 【注】 WDT 运行过程中改写 CKS[2:0] 位时，可能无法正确执行递增计数。 改写 CKS[2:0] 位时，必须停止 WDT。	时钟分频比	上溢周期	000: 1 × P ϕ	（7.7 μ s）	001: 1/64×P ϕ	（500 μ s）	010: 1/128×P ϕ	（1.0ms）	011: 1/256×P ϕ	（2.0ms）	100: 1/512×P ϕ	（4.0ms）	101: 1/1024×P ϕ	（8.0ms）	110: 1/4096×P ϕ	（32ms）	111: 1/16384×P ϕ	（128ms）
时钟分频比	上溢周期																					
000: 1 × P ϕ	（7.7 μ s）																					
001: 1/64×P ϕ	（500 μ s）																					
010: 1/128×P ϕ	（1.0ms）																					
011: 1/256×P ϕ	（2.0ms）																					
100: 1/512×P ϕ	（4.0ms）																					
101: 1/1024×P ϕ	（8.0ms）																					
110: 1/4096×P ϕ	（32ms）																					
111: 1/16384×P ϕ	（128ms）																					

13.3.3 看门狗复位控制 / 状态寄存器 (WRCSR)

WRCSR 为 8 位可读取 / 写入的寄存器，控制由看门狗定时器计数器 (WTCNT) 上溢所引起的内部复位信号的产生。

【注】 为了防止错误改写，本寄存器的写入方法与一般的寄存器不同。详情参阅 “13.3.4 寄存器存取时的注意事项”。

位:	7	6	5	4	3	2	1	0
	WOVF	RSTE	RSTS	-	-	-	-	-
初始值:	0	0	0	1	1	1	1	1
R/W:	R/(W)	R/W	R/W	R	R	R	R	R

位	位名称	初始值	R/W	说 明
7	WOVF	0	R/(W)	看门狗定时器上溢 表示在看门狗定时器模式 WTCNT 上溢。在间隔定时器模式，此位不置位。 0: 无上溢 1: 在看门狗定时器模式 WTCNT 上溢 [清除条件] • 读取 WOVF 后写入 0
6	RSTE	0	R/W	复位允许 在看门狗定时器模式，WTCNT 上溢时，可选择是否产生本 LSI 内部复位信号。在间隔定时器模式，忽略此设定值。 0: WTCNT 上溢时内部不复位 * 1: WTCNT 上溢时内部复位 【注】* 本 LSI 内部不复位，但 WDT 内的 WTCNT、WTCSR 复位。
5	RSTS	0	R/W	复位选择 选择在看门狗定时器模式，WTCNT 上溢时的复位种类。在间隔定时器模式，忽略此设定值。 0: 上电复位 1: 手动复位
4 ~ 0	—	均为 1	R	保留位 读取值、写入值总是为 1。

13.3.4 寄存器存取时的注意事项

为了防止轻易改写看门狗定时器计数器 (WTCNT)、看门狗定时器控制 / 状态寄存器 (WTCSR)、看门狗复位控制 / 状态寄存器 (WRCSR)，写入方法与一般寄存器不同，必须按照以下方法读取或写入。

(1) 写入 WTCNT 及 WTCSR

写入 WTCNT 及 WTCSR 时，必须使用字传送指令，用字节传送或长字传送指令不能写入。

如图 13.2 所示，写入 WTCNT 时，必须将高位字节设置为 H'5A、低位字节设置为写入数据后再传送。写入 WTCSR 时，必须将高位字节设置为 H'A5、低位字节设置为写入数据后再传送。如此传送，低位字节的数据即可写入 WTCNT 或 WTCSR。

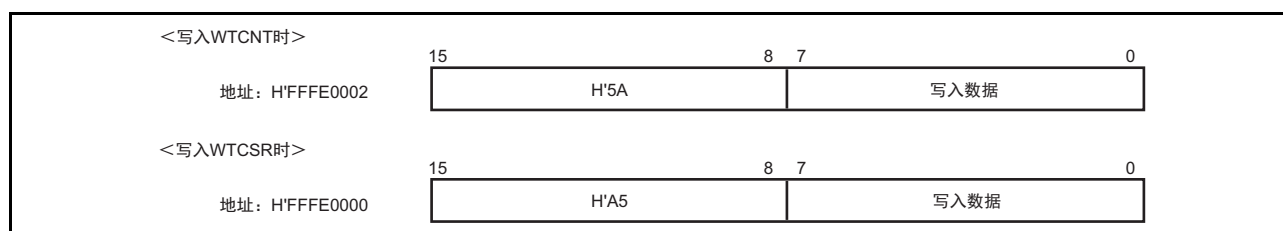


图 13.2 写入 WTCNT 及 WTCSR

(2) 写入 WRCSR

写入 WRCSR 时，必须对地址 H'FFFE0004 执行字传送，使用字节传送指令或长字节传送指令则不能写入。

对 WOVF 位 (bit7) 写入 0 及写入 RSTE 位 (bit6) 和 RSTS 位 (bit5) 时，如图 13.3 所示，写入方法不同。

对 WOVF 位写入 0 时，必须将高位字节设置为 H'5A、低位字节设置为 H'00 后再传送，这样可使 WOVF 位清 0。此时，RSTE、RSTS 位不受影响。写入 RSTE、RSTS 位时，必须将高位字节设置为 H'5A、低位字节设置为写入数据，这样可分别将低位字节的 bit6 和 bit5 的值写入 RSTE 位和 RSTS 位。此时，WOVF 位不受影响。

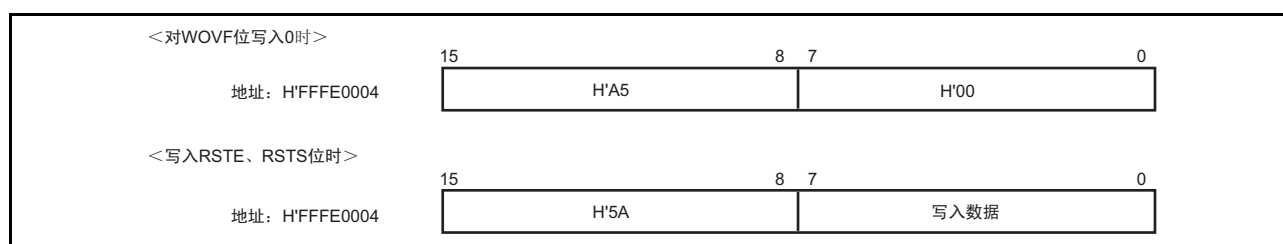


图 13.3 写入 WRCSR

(3) 从 WTCNT、WTCSR、WRCSR 读取

可用与一般的寄存器相同的方法读取。WTCSR 分配至地址 H'FFFE0000，WTCNT 分配至地址 H'FFFE0002，WRCSR 分配至地址 H'FFFE0004。读取时，必须使用字节传送指令。

13.4 WDT 的使用方法

13.4.1 软件待机模式的解除步骤

由 NMI 等中断解除中断软件待机模式时，需使用 WDT。步骤如下（由复位解除时，WDT 不运行，因此在时钟振荡稳定前，必须保持 $\overline{\text{RES}}$ 引脚和 $\overline{\text{MRES}}$ 引脚为低电平状态）。

1. 转移至软件待机模式前，必须将 WTCSR 的 TME 位设定为 0。如果将 TME 位设定为 1，可能在计数上溢时错误复位或产生间隔定时器中断。
2. 预先对 WTCSR 的 CKS[2:0] 位设定使用的计数时钟类型和对 WTCNT 设定初始值。设定这些值时，计数上溢前的时间必须大于等于时钟振荡稳定时间。
3. 将待机控制寄存器（STBCR：参阅“第 28 章 低功耗模式”）的 STBY 位设定为 1 后，通过执行 SLEEP 指令转移至软件待机模式，时钟停止。
4. 通过 NMI 信号变化的边沿等中断检测，WDT 开始计数。
5. WDT 计数上溢时，CPG 开始提供时钟，本 LSI 重新运行。此时，WRCSR 的 WOVF 不置位。

13.4.2 频率更改步骤

使用 WDT 更改使用 PLL 的频率。仅通过转换分频器更改频率时，不使用 WDT。

1. 频率更改前，WTCSR 的 TME 位必须清 0。如果 TME 位设定为 1，则可能在计数上溢时错误复位或产生间隔定时器中断。
2. 预先对 WTCSR 的 CKS[2:0] 位设定使用的计数时钟种类和对 WTCNT 设定初始值。设定这些值时，计数上溢前的时间必须大于等于时钟振荡稳定时间。但是 WDT 通过设定后的时钟递增计数。
3. 改写频率控制寄存器（FRQCR）时，处理器内部暂时停止，WDT 开始计数。
4. WDT 计数上溢时，CPG 重新提供时钟，本 LSI 重新开始运行。此时，WRCSR 的 WOVF 不置位。
5. 计数器值为 H'00 时停止。
6. 在频率更改指令后改写 WTCNT 时，必须读取 WTCNT，确认为 H'00 之后再改写。

13.4.3 看门狗定时器模式的使用方法

1. 设定 WTCSR 的 WT/IT 位为 1 后，对 CKS[2:0] 设定时钟种类、对 WRCSR 的 RSTE 位设定是否复位本 LSI 的内部、对 RSTS 位设定此时的复位类型、并对 WTCNT 设定初始值。
2. WTCSR 的 TME 位置 1 时，以看门狗定时器模式开始计数。
3. 以看门狗定时器模式运行时，为了防止计数器上溢，必须定期将计数器改写为 H'00。
4. 计数器上溢时，WDT 将 WRCSR 的 WOVF 置 1，并向外部输出 WDTOVF 信号，如图 13.4 所示。使用 WDTOVF 信号可复位系统。在 $64 \times P\phi$ 时钟期间输出 WDTOVF 信号。
5. WRCSR 的 RSTE 位预先置 1 时，可与 WDTOVF 信号同时产生复位本 LSI 内部信号。通过设定 WRCSR 的 RSTS 位，可选择上电复位或手动复位。在 $128 \times P\phi$ 时钟期间输出内部复位信号。
6. 同时产生 $\overline{\text{RES}}$ 引脚的输入信号引起的复位及 WDT 上溢引起的复位时，优先 $\overline{\text{RES}}$ 引脚引起的复位，WRCSR 的 WOVF 位清 0。

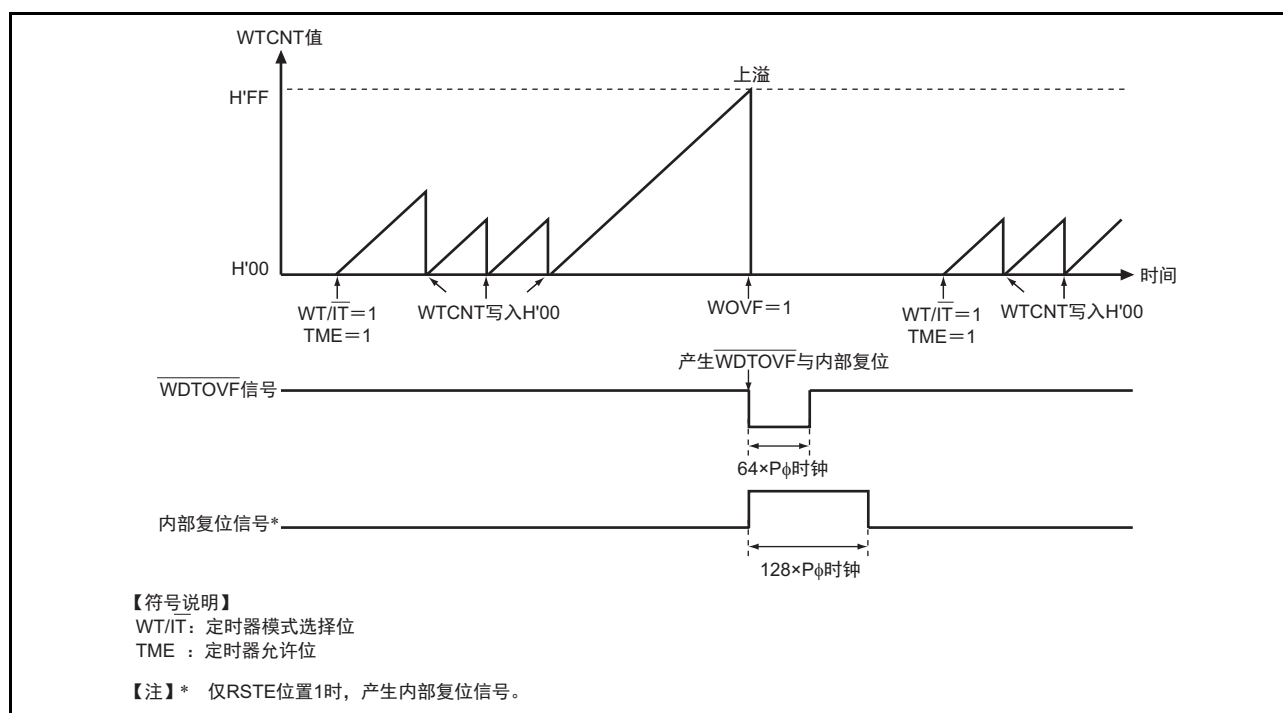


图 13.4 看门狗定时器模式时的运行

13.4.4 间隔定时器模式的使用方法

在以间隔定时器模式运行的过程中, 计数器每次上溢时, 产生间隔定时器中断。因此, 可定期产生中断。

1. WTCSR的WT/IT位置0, 对CKS[2:0]位设定WTCSR的计数时钟类型, 对WTCNT设定初始值。
2. WTCSR的TME位置1时, 以间隔定时器模式开始计数。
3. 计数器上溢时, WDT将WTCSR的IOVF位置1, 并将间隔定时器的中断请求传送至INTC, 计数器继续计数。

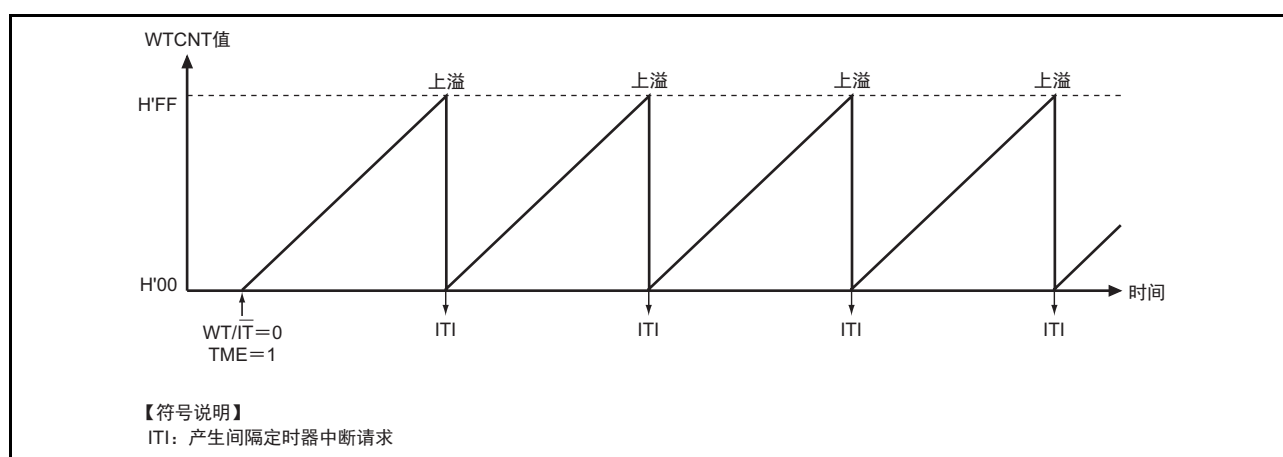


图 13.5 间隔定时器模式时的运行

13.5 使用时的注意事项

间隔定时器模式或看门狗定时器模式，有以下注意事项。

13.5.1 定时器误差

定时器开始运行后 WTCNT 寄存器的第一个递增计数时序为：根据以上电复位为基点的 WTCSR 寄存器的 TME 位的置位时序，从 $P\phi$ 的 1 个周期后（最短）到 CKS[2:0] 选择的分频时序（最长）的期间。第 2 次及其后的递增计数时序为选择的分频时序。因此，上述第一个递增计数前的时间差为定时器误差。定时器运行时，改写 WTCNT 寄存器后，第一个递增计数时序也相同。

13.5.2 禁止设定 H'FF 为 WTCNT 的设定值

WTCNT 的值为 H'FF 时，WDT 判断为上溢。对 WTCNT 设定 H'FF 时，与 CKS[2:0] 的选择时钟无关，立刻产生间隔定时器中断或 WDT 复位。

13.5.3 间隔定时器上溢标志

WTCNT 的值为 H'FF 时，WTCSR 的 IOVF 标志不可清除。

WTCNT 的值为 H'00 后清除 IOVF 标志，或将 WTCNT 的值改写为 H'FF 以外的值后清除 IOVF 标志。

13.5.4 $\overline{\text{WDTOVF}}$ 信号引起的系统复位

将 $\overline{\text{WDTOVF}}$ 信号输入本 LSI 的 $\overline{\text{RES}}$ 引脚时，无法正确初始化本 LSI。

不得对本 LSI 的 $\overline{\text{RES}}$ 引脚逻辑输入 $\overline{\text{WDTOVF}}$ 信号。通过 $\overline{\text{WDTOVF}}$ 信号复位整个系统时，必须按图 13.6 所示的电路进行复位。

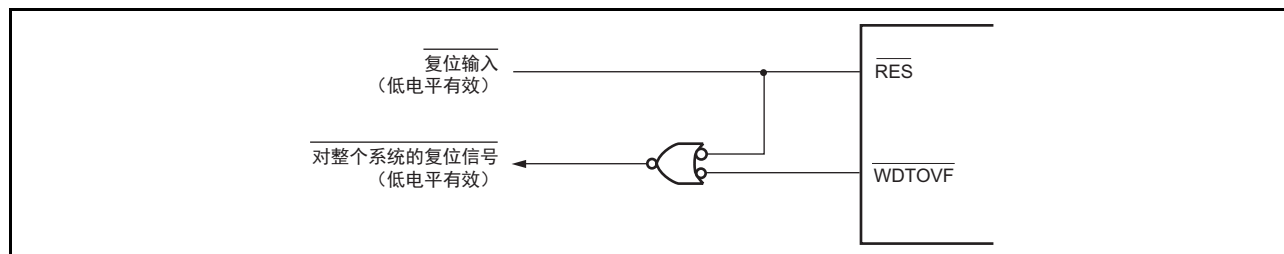


图 13.6 $\overline{\text{WDTOVF}}$ 信号引起的系统复位电路例

13.5.5 看门狗定时器模式的手动复位

产生看门狗定时器模式的手动复位时，可保持总线周期。总线权释放及 DMAC 突发传送过程中，产生手动复位时，在 CPU 获得总线权前一直保留手动复位异常处理。

第 14 章 实时时钟（RTC）

本 LSI 内置实时时钟（RTC: Real Time Clock）及用于 RTC 的 32.768kHz 晶体振荡电路。

14.1 特点

- 装载了时钟和日历功能（BCD表示）
计数秒、分、小时、星期、日、月、年
- 装载了 1～64Hz 定时器（2 进制表示）
64Hz 计数器表示 RTC 分频电路中 64Hz～1Hz 的状态。
- 开始/停止功能
- 30 秒调整功能
- 报警中断
可选择是否与秒、分、小时、星期、日、月、年任一个进行比较作为报警中断条件
- 周期中断
可选择 1/256 秒、1/64 秒、1/16 秒、1/4 秒、1/2 秒、1 秒、2 秒周期作为中断周期
- 进位中断
表示秒计数器进位或读取 64Hz 计数器时产生 64Hz 计数器进位的进位中断功能
- 闰年自动修正功能

RTC 框图如图 14.1 所示。

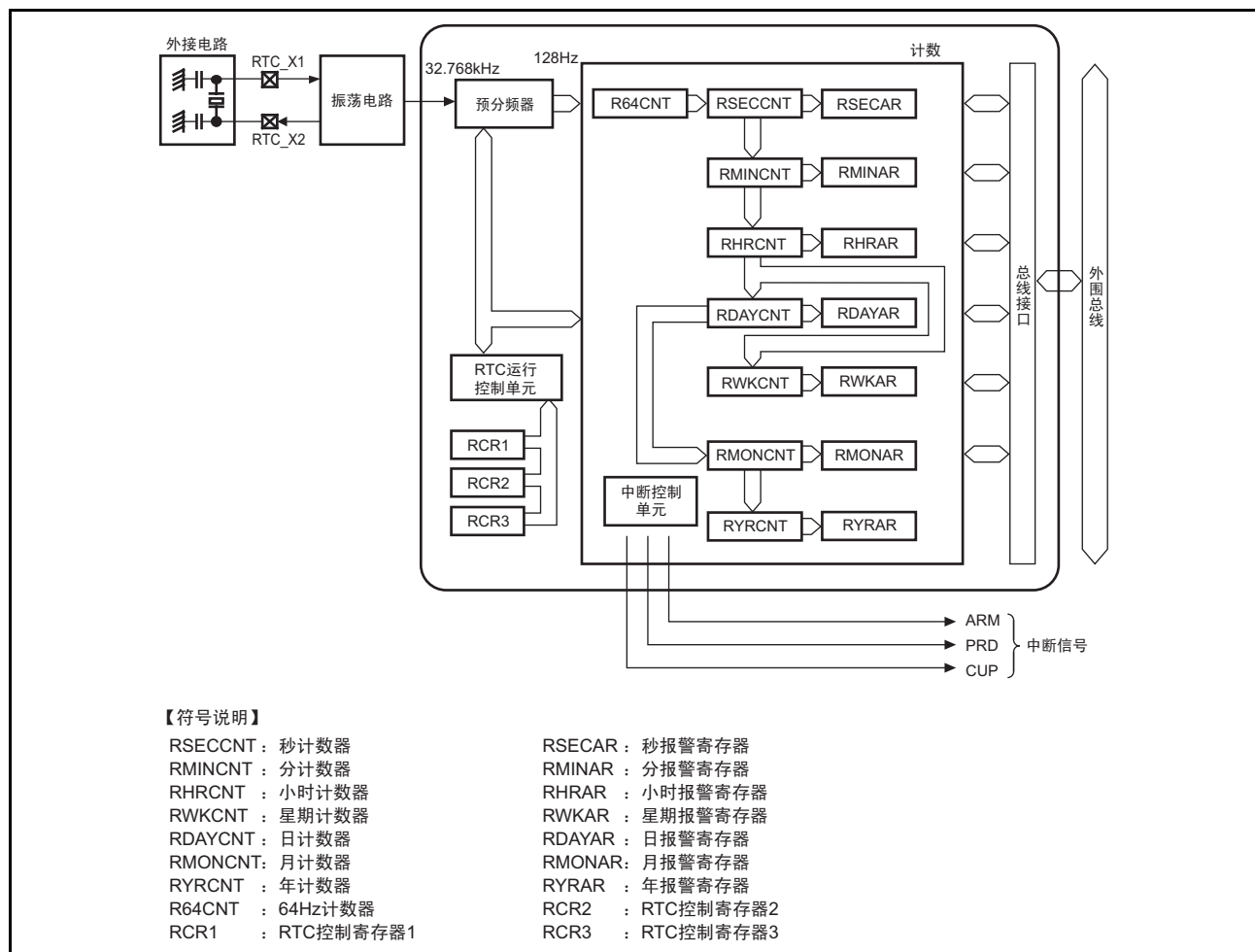


图 14.1 RTC 框图

14.2 输入 / 输出引脚

RTC 引脚结构如表 14.1 所示。

表 14.1 引脚结构

名称	引脚名称	输入 / 输出	功 能
用于 RTC 的晶体谐振器 / 外部时钟	RTC_X1	输入	连接 RTC 的 32.768KHz 晶体谐振器，RTC_X1 引脚也可输入外部时钟。
	RTC_X2	输出	

14.3 寄存器说明

RTC 有以下寄存器。

表 14.2 寄存器结构

寄存器名称	简称	R/W	初始值	地址	存取长度
64Hz 计数器	R64CNT	R	H'xx	H'FFFF2000	8
秒计数器	RSECCNT	R/W	H'xx	H'FFFF2002	8
分计数器	RMINCNT	R/W	H'xx	H'FFFF2004	8
小时计数器	RHRCNT	R/W	H'xx	H'FFFF2006	8
星期计数器	RWKCNT	R/W	H'0x	H'FFFF2008	8
日计数器	RDAYCNT	R/W	H'xx	H'FFFF200A	8
月计数器	RMONCNT	R/W	H'xx	H'FFFF200C	8
年计数器	RYRCNT	R/W	H'xxxx	H'FFFF200E	16
秒报警寄存器	RSECAR	R/W	H'xx	H'FFFF2010	8
分报警寄存器	RMINAR	R/W	H'xx	H'FFFF2012	8
小时报警寄存器	RHRAR	R/W	H'xx	H'FFFF2014	8
星期报警寄存器	RWKAR	R/W	H'0x	H'FFFF2016	8
日报警寄存器	RDAYAR	R/W	H'xx	H'FFFF2018	8
月报警寄存器	RMONAR	R/W	H'xx	H'FFFF201A	8
年报警寄存器	RYRAR	R/W	H'xxxx	H'FFFF2020	16
RTC 控制寄存器 1	RCCR1	R/W	H'00	H'FFFF201C	8
RTC 控制寄存器 2	RCCR2	R/W	H'09	H'FFFF201E	8
RTC 控制寄存器 3	RCCR3	R/W	H'00	H'FFFF2024	8

14.3.1 64Hz 计数器（R64CNT）

R64CNT 表示 RTC 分频电路中 64Hz ~ 1Hz 的状态。

产生来自 128Hz 分频段的进位时，如果读取此寄存器，则表示 RTC 控制寄存器 1（RCR1）的 CF 位置 1，64Hz 计数器的读取与进位重合。此时，读取值无效，对 RCR1 的 CF 位写入 0 并清除后，需再次读取 R64CNT。

如果 RTC 控制寄存器 2（RCR2）的 RESET 位置 1，或 RCR2 的 ADJ 位置 1，则初始化 RTC 的分频电路与 R64CNT。

位:	7	6	5	4	3	2	1	0
	-	1Hz	2Hz	4Hz	8Hz	16Hz	32Hz	64Hz
初始值:	0	不定	不定	不定	不定	不定	不定	不定
R/W:	R	R	R	R	R	R	R	R

位	位名称	初始值	R/W	说 明
7	—	0	R	保留位 读取值、写入值总是为 0。
6	1Hz	不定	R	表示 RTC 分频电路中 1Hz ~ 64Hz 的状态。
5	2Hz	不定	R	
4	4Hz	不定	R	
3	8Hz	不定	R	
2	16Hz	不定	R	
1	32Hz	不定	R	
0	64Hz	不定	R	

14.3.2 秒计数器（RSECCNT）

RSECCNT 计数器设定 / 计数 RTC 被 BCD 码化的秒部分，根据 64Hz 计数器每秒的进位执行计数。

可设定的范围为 10 进制（BCD）、00 ~ 59。设定此范围以外的值时，无法正常运行。写入处理必须在 RCR2 的 START 位停止计数后执行。

位:	7	6	5	4	3	2	1	0
	-	10秒				1秒		
初始值:	0	不定	不定	不定	不定	不定	不定	不定
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7	—	0	R	保留位 读取值、写入值总是为 0。
6 ~ 4	10 秒	不定	R/W	秒十位计数 秒十位从 0 到 5 计数，执行 60 秒计数。
3 ~ 0	1 秒	不定	R/W	秒个位计数 秒个位每秒从 0 到 9 计数。产生进位时，秒十位加 1。

14.3.3 分计数器（RMINCNT）

RMINCNT 计数器设定 / 计数 RTC 被 BCD 码化的分钟部分，根据秒计数器每分的进位执行计数。

可设定的范围为 10 进制（BCD）时，00 ~ 59。设定此范围以外的值时，无法正常运行，必须在 RCR2 的 START 位停止计数后，执行写入处理。

位:	7	6	5	4	3	2	1	0
	-	10分			1分			
初始值:	0	不定	不定	不定	不定	不定	不定	不定
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7	—	0	R	保留位 读取值、写入值总是为 0。
6 ~ 4	10 分	不定	R/W	分十位计数 分十位从 0 到 5 计数，执行 60 分计数。
3 ~ 0	1 分	不定	R/W	分个位计数 分个位每分从 0 到 9 计数。产生进位时，分十位加 1。

14.3.4 小时计数器（RHRCNT）

RHRCNT 计数器设定 / 计数 RTC 被 BCD 码化的小时部分，根据分计数器每小时的进位执行计数。

可设定的范围为 10 进制（BCD）时，00 ~ 23。设定此范围以外的值时，无法正常运行，必须在 RCR2 的 START 位停止计数后，执行写入处理。

位:	7	6	5	4	3	2	1	0
	-	-	10小时		1小时			
初始值:	0	0	不定	不定	不定	不定	不定	不定
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7、6	—	均为 0	R	保留位 读取值、写入值总是为 0。
5、4	10 小时	不定	R/W	小时十位计数 小时十位从 0 到 2 计数。
3 ~ 0	1 小时	不定	R/W	小时个位计数 小时个位每小时从 0 到 9 计数。产生进位时，小时十位加 1。

14.3.5 星期计数器（RWKCNT）

RWKCNT 计数器设定 / 计数 RTC 被 BCD 码化的星期部分，根据小时计数器每天的进位执行计数。

可设定的范围为 10 进制（BCD）时，0～6。设定此范围以外的值时，无法正常运行。写入处理必须在 RCR2 的 START 位停止计数后执行。

位:	7	6	5	4	3	2	1	0
	-	-	-	-	-	星期		
初始值:	0	0	0	0	0	不定	不定	不定
R/W:	R	R	R	R	R	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7～3	—	均为 0	R	保留位 读取值、写入值总是为 0。
2～0	星期	不定	R/W	星期计数 以 2 进制代码表示星期。 000: 星期日 001: 星期一 010: 星期二 011: 星期三 100: 星期四 101: 星期五 110: 星期六 111: 保留（禁止设定）

14.3.6 日计数器（RDAYCNT）

RDAYCNT 计数器设定 / 计数 RTC 被 BCD 码化的日部分，根据小时计数器每天的进位执行计数。

可设定的范围为 10 进制（BCD）时，01～31。设定此范围以外的值时，无法正常运行。写入处理必须在 RCR2 的 START 位停止计数后执行。

RDAYCNT 的设定范围根据每月及闰年变化，因此必须确认后再设定。将年计数器（RYRCNT）作为西历，通过可否被 400、100、4 整除计算闰年。

位:	7	6	5	4	3	2	1	0
	-	-	10日		1日			
初始值:	0	0	不定	不定	不定	不定	不定	不定
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7～6	—	均为 0	R	保留位 读取值、写入值总是为 0。
5、4	10 日	不定	R/W	日十位计数
3～0	1 日	不定	R/W	日个位计数 日个位每日从 0 到 9 计数。产生进位时，日十位加 1。

14.3.7 月计数器（RMONCNT）

RMONCNT 计数器设定 / 计数 RTC 被 BCD 码化的月部分，根据日计数器每月的进位执行计数。

可设定的范围为 10 进制（BCD）时，01 ~ 12。设定此范围以外的值时，无法正常运行。写入处理必须在 RCR2 的 START 位停止计数后执行。

位:	7	6	5	4	3	2	1	0
	-	-	-	10个月			1个月	
初始值:	0	0	0	不定	不定	不定	不定	不定
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7 ~ 5	—	均为 0	R	保留位 读取值、写入值总是为 0。
4	10 个月	不定	R/W	月十位计数
3 ~ 0	1 个月	不定	R/W	月个位计数 月个位每月从 0 到 9 计数。产生进位时，月十位加 1。

14.3.8 年计数器（RYRCNT）

RYRCNT 计数器设定 / 计数 RTC 被 BCD 码化的年部分，根据月计数器每年的进位执行计数。

可设定的范围为 10 进制（BCD）时，0000 ~ 9999。设定此范围以外的值时，无法正常运行。写入处理必须在 RCR2 的 START 位停止计数后执行。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
初始值:	不定	不定	不定	不定	不定	不定	不定	不定	不定	不定	不定	不定	不定	不定	不定	不定
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15 ~ 12	1000 年	不定	R/W	年千位计数
11 ~ 8	100 年	不定	R/W	年百位计数
7 ~ 4	10 年	不定	R/W	年十位计数
3 ~ 0	1 年	不定	R/W	年个位计数

14.3.9 秒报警寄存器（RSECAR）

RSECAR 为 RTC 被 BCD 码化的秒部分的计数器 RSECCNT 对应的报警寄存器。ENB 位置 1 时，与 RSECCNT 的值进行比较。报警寄存器（RSECAR、RMINAR、RHRAR、RWKAR、RDAYAR、RMONAR、RCR3）中，只有 ENB 位置 1 的报警寄存器进行计数器与报警寄存器的比较，各自全部匹配时，将 RCR1 的报警标志置 1。

可设定的范围为 10 进制（BCD）时，00 ~ 59+ENB 位。设定此范围以外的值时，无法正常运行。

位:	7	6	5	4	3	2	1	0
	ENB	10秒			1秒			
初始值:	0	不定	不定	不定	不定	不定	不定	不定
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7	ENB	0	R/W	置 1 时，与 RSECCNT 的值比较。
6 ~ 4	10 秒	不定	R/W	秒十位的设定值
3 ~ 0	1 秒	不定	R/W	秒个位的设定值

14.3.10 分报警寄存器（RMINAR）

RMINAR 为 RTC 被 BCD 码化的分部分的计数器 RMINCNT 对应的报警寄存器。ENB 位置 1 时，与 RMINCNT 的值进行比较。报警寄存器（RSECAR、RMINAR、RHRAR、RWKAR、RDAYAR、RMONAR、RCR3）中，只有 ENB 位置 1 的报警寄存器进行计数器与报警寄存器的比较，各自全部匹配时，将 RCR1 的报警标志置 1。

可设定的范围为 10 进制（BCD）时，00 ~ 59+ENB 位。设定此范围以外的值时，无法正常运行。

位:	7	6	5	4	3	2	1	0
	ENB	10分			1分			
初始值:	0	不定	不定	不定	不定	不定	不定	不定
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7	ENB	0	R/W	置 1 时，与 RMINCNT 的值比较。
6 ~ 4	10 分	不定	R/W	分十位的设定值
3 ~ 0	1 分	不定	R/W	分个位的设定值

14.3.11 小时报警寄存器（RHRAR）

RHRAR 为 RTC 被 BCD 码化的小时部分的计数器 RHRCNT 对应的报警寄存器。ENB 位置 1 时，与 RHRCNT 的值进行比较。报警寄存器（RSECAR、RMINAR、RHRAR、RWKAR、RDAYAR、RMONAR、RCR3）中，只有 ENB 位置 1 的报警寄存器进行计数器与报警寄存器的比较，各自全部匹配时，将 RCR1 的报警标志置 1。

可设定的范围为 10 进制（BCD）时，00 ~ 23+ENB 位。设定此范围以外的值时，无法正常运行。

位:	7	6	5	4	3	2	1	0
	ENB	-	10小时		1小时			
初始值:	0	0	不定	不定	不定	不定	不定	不定
R/W:	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7	ENB	0	R/W	置 1 时，与 RHRCNT 的值比较。
6	—	0	R	保留位 读取值、写入值总是为 0。
5、4	10 小时	不定	R/W	小时十位的设定值
3 ~ 0	1 小时	不定	R/W	小时个位的设定值

14.3.12 星期报警寄存器（RWKAR）

RWKAR 为 RTC 被 BCD 码化的星期部分的计数器 RWKCNT 对应的报警寄存器。ENB 位置 1 时，与 RWKCNT 的值进行比较。报警寄存器（RSECAR、RMINAR、RHRAR、RWKAR、RDAYAR、RMONAR、RCR3）中，只有 ENB 位置 1 的报警寄存器进行计数器与报警寄存器的比较，各自全部匹配时，将 RCR1 的报警标志置 1。

可设定的范围为 10 进制（BCD）时，0 ~ 6+ENB 位。设定此范围以外的值时，无法正常运行。

位:	7	6	5	4	3	2	1	0
	ENB	-	-	-	-	星期		
初始值:	0	0	0	0	0	不定	不定	不定
R/W:	R/W	R	R	R	R	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7	ENB	0	R/W	置 1 时，与 RWKCNT 的值比较。
6 ~ 3	—	均为 0	R	保留位 读取值、写入值总是为 0。
2 ~ 0	星期	不定	R/W	星期的设定值 000: 星期日 001: 星期一 010: 星期二 011: 星期三 100: 星期四 101: 星期五 110: 星期六 111: 保留（禁止设定）

14.3.13 日报警寄存器（RDAYAR）

RDAYAR 为对应 RTC 被 BCD 码化的日部分的计数器 RDAYCNT 的报警寄存器。ENB 位置 1 时，与 RDAYCNT 的值进行比较。报警寄存器（RSECAR、RMINAR、RHRAR、RWKAR、RDAYAR、RMONAR、RCR3）中，只有 ENB 位置 1 的报警寄存器进行计数器与报警寄存器的比较，各自全部匹配时，将 RCR1 的报警标志置 1。

可设定的范围为 10 进制（BCD）时，01 ~ 31+ENB 位。设定此范围以外的值时，则无法正常运行。

位:	7	6	5	4	3	2	1	0
	ENB	-	10日		1日			
初始值:	0	0	不定	不定	不定	不定	不定	不定
R/W:	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7	ENB	0	R/W	置 1 时，与 RDAYCNT 的值比较。
6	—	0	R	保留位 读取值、写入值总是为 0。
5、4	10 日	不定	R/W	日十位的设定值
3 ~ 0	1 日	不定	R/W	日个位的设定值

14.3.14 月报警寄存器（RMONAR）

RMONAR 为 RTC 被 BCD 码化的月部分的计数器 RMONCNT 对应的报警寄存器。ENB 位置 1 时，与 RMONCNT 的值进行比较。报警寄存器（RSECAR、RMINAR、RHRAR、RWKAR、RDAYAR、RMONAR、RCR3）中，只有 ENB 位置 1 的报警寄存器进行计数器与报警寄存器的比较，各自全部匹配时，将 RCR1 的报警标志置 1。

可设定的范围为 10 进制（BCD）时，01 ~ 12+ENB 位。设定此范围以外的值时，无法正常运行。

位:	7	6	5	4	3	2	1	0
	ENB	-	-	10个月	1个月			
初始值:	0	0	0	不定	不定	不定	不定	不定
R/W:	R/W	R	R	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7	ENB	0	R/W	置 1 时，与 RMONCNT 的值比较。
6、5	—	均为 0	R	保留位 读取值、写入值总是为 0。
4	10 个月	不定	R/W	月十位的设定值
3 ~ 0	1 个月	不定	R/W	月个位的设定值

14.3.15 年报警寄存器（RYRAR）

RYRAR 为 RTC 被 BCD 码化的年部分的计数器 RYRCNT 对应的报警寄存器。
可设定的范围为 10 进制（BCD）时，0000 ～ 9999。设定此范围以外的值时，无法正常运行。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	1000年				100年				10年				1年			
初始值:	不定	不定	不定	不定	不定	不定	不定	不定	不定	不定	不定	不定	不定	不定	不定	不定
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15 ～ 12	1000 年	不定	R/W	年千位的设定值
11 ～ 8	100 年	不定	R/W	年百位的设定值
7 ～ 4	10 年	不定	R/W	年十位的设定值
3 ～ 0	1 年	不定	R/W	年个位的设定值

14.3.16 RTC 控制寄存器 1（RCR1）

RCR1 为与进位标志及报警标志相关的寄存器。可对各标志选择是否产生中断。

分频电路复位（RCR2 的 RESET 位和 ADJ 位置 1）前，CF 标志不定。使用 CF 标志时，必须在使用前复位分频电路。

位:	7	6	5	4	3	2	1	0
	CF	-	-	CIE	AIE	-	-	AF
初始值:	不定	0	0	0	0	0	0	0
R/W:	R/W	R	R	R/W	R/W	R	R	R/W

位	位名称	初始值	R/W	说 明
7	CF	不定	R/W	进位标志 表示该标志置 1 时，秒计数器进位或读取 64Hz 计数器时产生 64Hz 计数器进位，此时无法保证读取的计数寄存器的值，需再次读取。 0: 无秒计数器进位或读取 64Hz 计数器时的 64Hz 计数器进位 [清除条件] 对 CF 写入 0 时 1: 有秒计数器进位或读取 64Hz 计数器时的 64Hz 计数器进位 [置位条件] 有秒计数器进位或读取 64Hz 计数器时的 64Hz 计数器进位，或对 CF 写入 1 时
6、5	—	均为 0	R	保留位 读取值、写入值总是为 0。
4	CIE	0	R/W	进位中断允许标志 进位标志（CF）置 1 时，为允许产生中断的位。 0: CF 标志置 1 时，不产生进位中断 1: CF 标志置 1 时，产生进位中断
3	AIE	0	R/W	报警中断允许标志 报警标志（AF）置 1 时，为允许产生中断的位。 0: AF 标志置 1 时，不产生报警中断 1: AF 标志置 1 时，产生报警中断
2、1	—	均为 0	R	保留位 读取值、写入值总是为 0。

位	位名称	初始值	R/W	说 明
0	AF	0	R/W	报警标志 本标志是在报警寄存器（RSECAR、RMINAR、RHRAR、RWKAR、RDAYAR、RMONAR、RYRAR）设定的报警时间（仅限 ENB 位设定为 1 的寄存器）与计数器匹配时，被置 1 的标志。 0：报警寄存器与计数器不匹配 [清除条件] 对 AF 写入 0 时 1：报警寄存器与计数器匹配 * [置位条件] 报警寄存器（仅限 ENB 位设定为 1 的寄存器）与计数器匹配时 【注】 * 写入 1 时，保持原值。

14.3.17 RTC 控制寄存器 2（RCR2）

RCR2 是与周期中断控制、30 秒调整、分频电路复位及 RTC 计数控制相关的寄存器。

在上电复位或深度待机模式时初始化 RCR2。手动复位时，除 RTCEN 位及 START 位之外，其他均被初始化。

位:	7	6	5	4	3	2	1	0
	PEF	PES[2:0]			RTCEN	ADJ	RESET	START
初始值:	0	0	0	0	1	0	0	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7	PEF	0	R/W	周期中断标志 表示在 PES2 ~ PES0 位设定的周期产生中断。本标志置 1 时，产生周期中断。 0：在 PES2 ~ PES0 位设定的周期不产生中断 [清除条件] 对 PEF 写入 0 时 1：在 PES2 ~ PES0 位设定的周期产生中断 [置位条件] 在 PES2 ~ PES0 位设定的周期产生中断时，或对 PEF 标志写入 1 时
6 ~ 4	PES[2:0]	000	R/W	中断允许标志 设定周期中断的周期。 000：不产生周期中断 001：设置为每 1/256 秒产生周期中断 010：设置为每 1/64 秒产生周期中断 011：设置为每 1/16 秒产生周期中断 100：设置为每 1/4 秒产生周期中断 101：设置为每 1/2 秒产生周期中断 110：设置为每 1 秒产生周期中断 111：设置为每 2 秒产生周期中断
3	RTCEN	1	R/W	振荡器有效 控制 RTC 晶体振荡器的运行。 0：停止 RTC 晶体振荡器。 1：运行 RTC 晶体振荡器。

位	位名称	初始值	R/W	说 明
2	ADJ	0	R/W	30 秒调整 用于 30 秒调整，通过写入 1，在 29 秒以前舍为 00 秒、30 秒以后进位为 1 分。此时，分频电路（RTC 预分频器及 R64CNT）也同时复位。此 ADJ 位自动变为 0，因此无需特意写入 0，读取时为 0。 0：通常的时钟运行。 1：执行 30 秒调整。
1	RESET	0	R/W	复位 通过写入 1，初始化分频电路。写入 1 时，分频电路（RTC 预分频器及 R64CNT）复位后，此 RESET 位自动变为 0，因此无需特意写入 0，读取时为 0。 0：通常的时钟运行 1：复位分频电路
0	START	1	R/W	START 位 停止及重新启动计数器（时钟）。 0：秒、分、小时、日、星期、月、年计数器停止 * 1：秒、分、小时、日、星期、月、年计数器正常运行 * 【注】 * 只要不通过 RTCEN 位使 64Hz 计数器停止，该计数器持续运行。

14.3.18 RTC 控制寄存器 3（RCR3）

ENB 位置 1 时，RCR3 与 RYRCNT 的值进行比较。报警寄存器（RSECAR、RMINAR、RHRAR、RWKAR、RDAYAR、RMONAR、RCR3）中，只有 ENB 位置 1 的报警寄存器与计数器比较，各自全部匹配时，RCR1 的报警标志置 1。

位:	7	6	5	4	3	2	1	0
	ENB	-	-	-	-	-	-	-
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R

位	位名称	初始值	R/W	说 明
7	ENB	0	R/W	置 1 时，与 RYRCNT 的值比较
6 ~ 0	—	均为 0	R	保留位 读取值、写入值总是为 0。

14.4 运行说明

RTC 的使用例。

14.4.1 上电后寄存器的初始设定

上电后必须初始设定所有寄存器。

14.4.2 时间设定步骤

时间设定步骤例如图 14.2 所示。

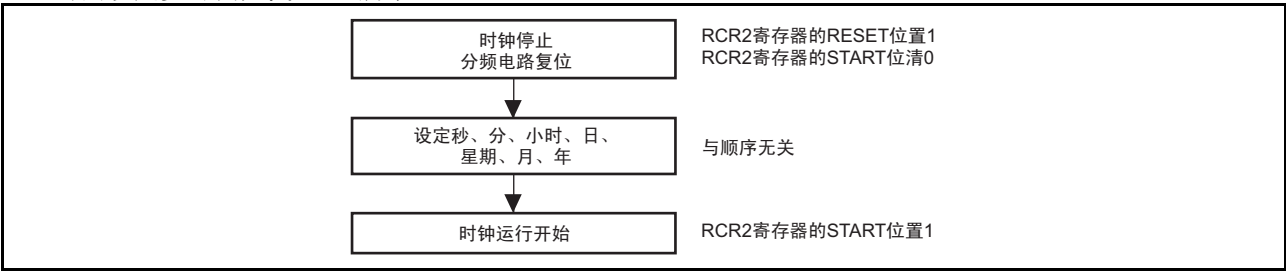


图 14.2 时间设定步骤

14.4.3 时间读取步骤

时间读取步骤如图 14.3 所示。

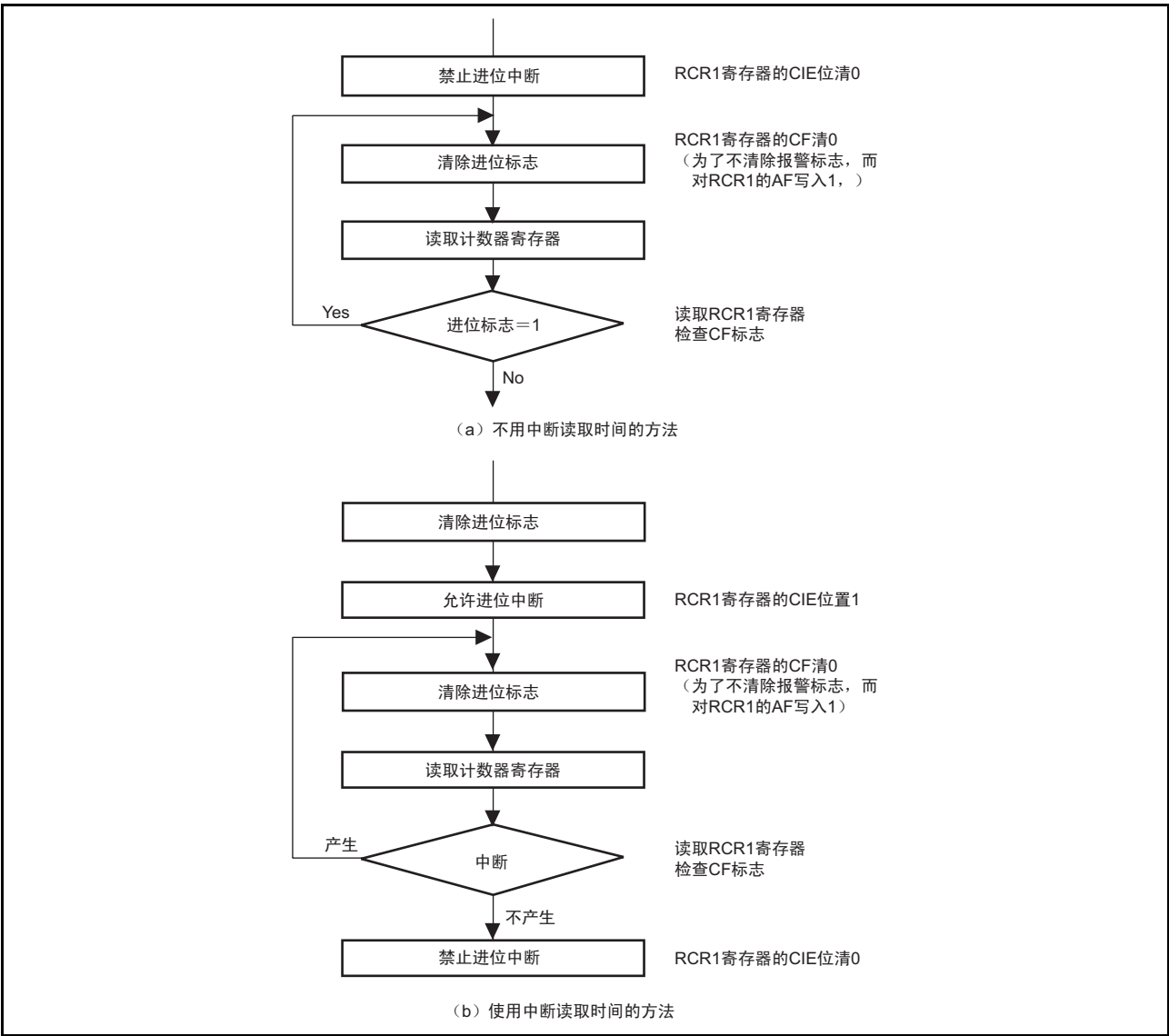


图 14.3 时间读取步骤

如果读取时间过程中产生进位，则可能无法得到正确时间，所以必须重新读取。不使用中断的方法如图 14.3（a）所示，使用进位中断的方法如图 14.3（b）所示。为了使程序简便，通常使用不使用中断的方法。

14.4.4 报警功能

报警功能的使用例如图 14.4 所示。

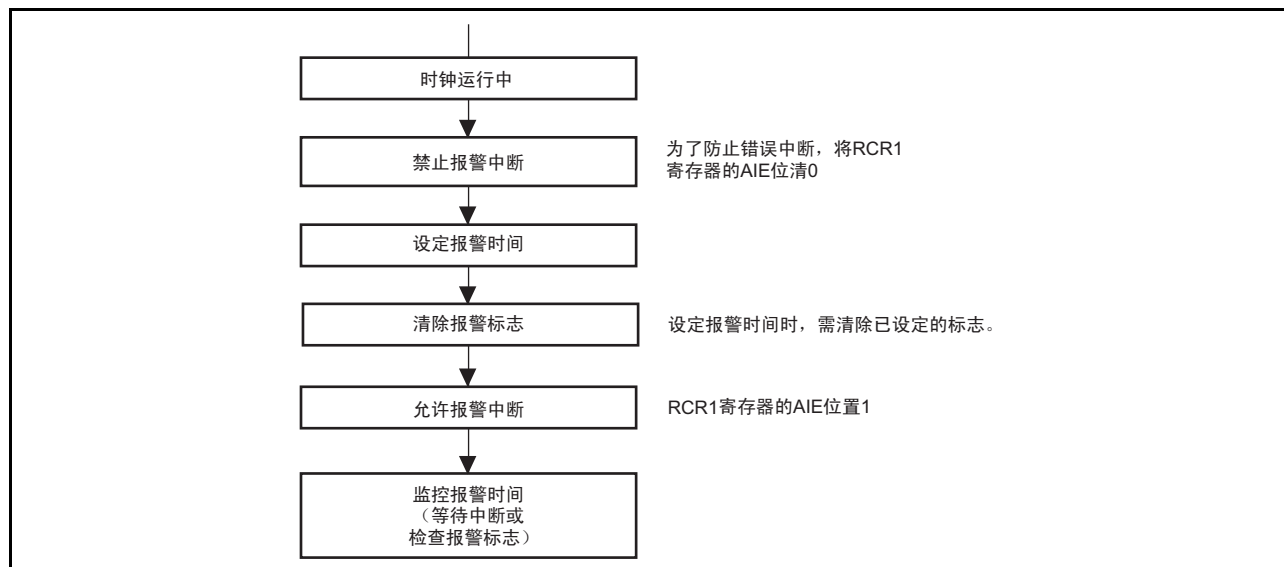


图 14.4 报警功能的使用方法

可通过秒、分、小时、星期、日、月、年中的任意一个或组合产生报警。对报警对象的报警寄存器的 ENB 位写入 1，并对低位设定报警时间。为非报警对象的寄存器时，对 ENB 位写入 0。

计数器与报警时间匹配时，RCR1 寄存器的 AF 位置 1。可通过读取此位确认报警检测，但通常通过中断执行。对 RCR1 寄存器的 AIE 位写入 1 时，可产生报警中断，并检测报警。

计数器与报警时间匹配时，报警标志置位，但是，对报警标志写入 0，即可清除报警标志。

14.5 使用时的注意事项

14.5.1 关于 RTC 计数运行时的寄存器写入

RTC 计数时（RCR2 寄存器的 START=1 时），不可对以下寄存器写入：

RSECCNT、RMINCNT、RHRCNT、RDAYCNT、RWKCNT、RMONCNT、RYRCONT

对上述寄存器写入时，必须暂停 RTC 计数运行后再写入。

14.5.2 关于实时时钟（RTC）周期中断的使用

周期中断的使用方法如图 14.5 所示。

周期中断是在 RCR2 寄存器的 PES[2:0] 位设定的周期可定期产生的中断。经过 PES[2:0] 位设定的时间后，PEF 置 1。

在设定 PES[2:0] 位及产生周期中断时，PEF 清 0。可通过读取此位确认周期中断的产生，但通常使用中断功能。

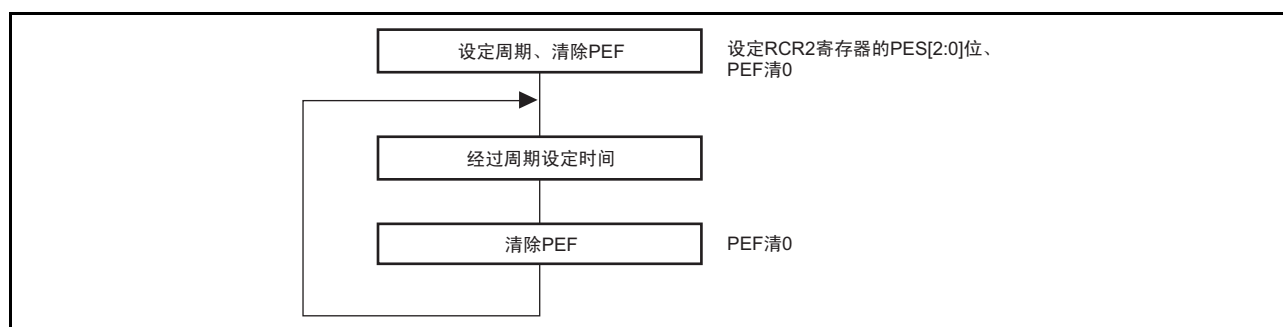


图 14.5 周期中断功能的使用方法

14.5.3 寄存器设定后的待机转移

设定 RTC 内的寄存器后，转移至待机状态时，可能导致不正确计数，必须在设定寄存器后，等待至少 2RTC 时钟，再转移至待机模式。

14.5.4 读取 / 写入寄存器时的注意事项

- 写入秒计数器等计数寄存器后再读取时，必须遵循“时间读取步骤”。
- 写入 RCR2 寄存器后再读取时，必须虚读 2 次后再读取。虚读 2 次可读取写入前的值。第 3 次读取即可反映写入的值。
- 写入除上述情况之外的寄存器后，再立即读取时，可反映写入值。

第 15 章 带 FIFO 的串行通信接口 (SCIF)

本 LSI 内置支持异步通信与时钟同步通信 2 种方式的 4 通道带 FIFO 的串行通信接口 (SCIF)。各通道均内置用于独立发送 / 接收的 16 段 FIFO 寄存器, 可实现高效、且高速地连续通信。

15.1 特点

- 异步串行通信
通过以字符单位取得同步的异步方式执行串行数据通信。可与 Universal Asynchronous Receiver/Transmitter (UART) 或 Asynchronous Communication Interface Adapter (ACIA) 等标准异步通信的 LSI 进行串行数据通信。可从 8 种格式选择串行数据的通信格式。
数据长度 : 7 位或 8 位
停止位长度 : 1 位或 2 位
奇偶校验 : 偶校验、奇校验或无奇偶校验
接收错误检测 : 检测奇偶校验错误、帧错误及溢出错误
断点检测 : 产生帧错误且随后有不少于 1 帧长度的空间 0 (低电平) 时, 检测断点;
产生帧错误时, 也可通过从串行端口寄存器直接读取 RxD 引脚的电平, 检测断点。
- 时钟同步串行通信
与时钟同步执行串行数据通信。能够与具有时钟同步通信功能的其他 LSI 进行串行数据通信。串行数据的通信格式只有 1 种。
数据长度 : 8 位
接收错误检测 : 检测溢出错误
- 可执行全双工通信
具有独立的发送部与接收部, 因此可同时发送与接收。另外, 发送部与接收部均为 16 段 FIFO 缓冲结构, 可高速连续发送及连续接收串行数据。
- 可通过内部波特率发生器选择任意位速率
- 内部或外部发送/接收时钟源
可从波特率发生器 (内部时钟) 或 SCK 引脚 (外部时钟) 选择
- 4 种中断源
即发送 FIFO 数据空中断、断点中断、接收 FIFO 数据满中断及接收错误中断等, 可分别独立请求中断。
- 不使用 SCIF 时, 为了降低功耗, 可停止对 SCIF 提供时钟, 并使运行停止。
- 在异步模式, 内置调制解调器控制功能 (RTS 及 CTS) (仅限通道 3)。
- 可检测发送及接收 FIFO 数据寄存器的数据个数, 及接收 FIFO 数据寄存器接收数据的接收错误个数。
- 在异步模式接收时, 可检测超时错误 (DR)。
- 在异步模式, 可选择以 16/8 倍位速率的基本时钟运行。
- 在异步模式, 选择内部时钟为时钟源、SCK 引脚为输入引脚时, 可选择波特率发生器的普通/倍速模式。

SCIF 框图如图 15.1 所示。

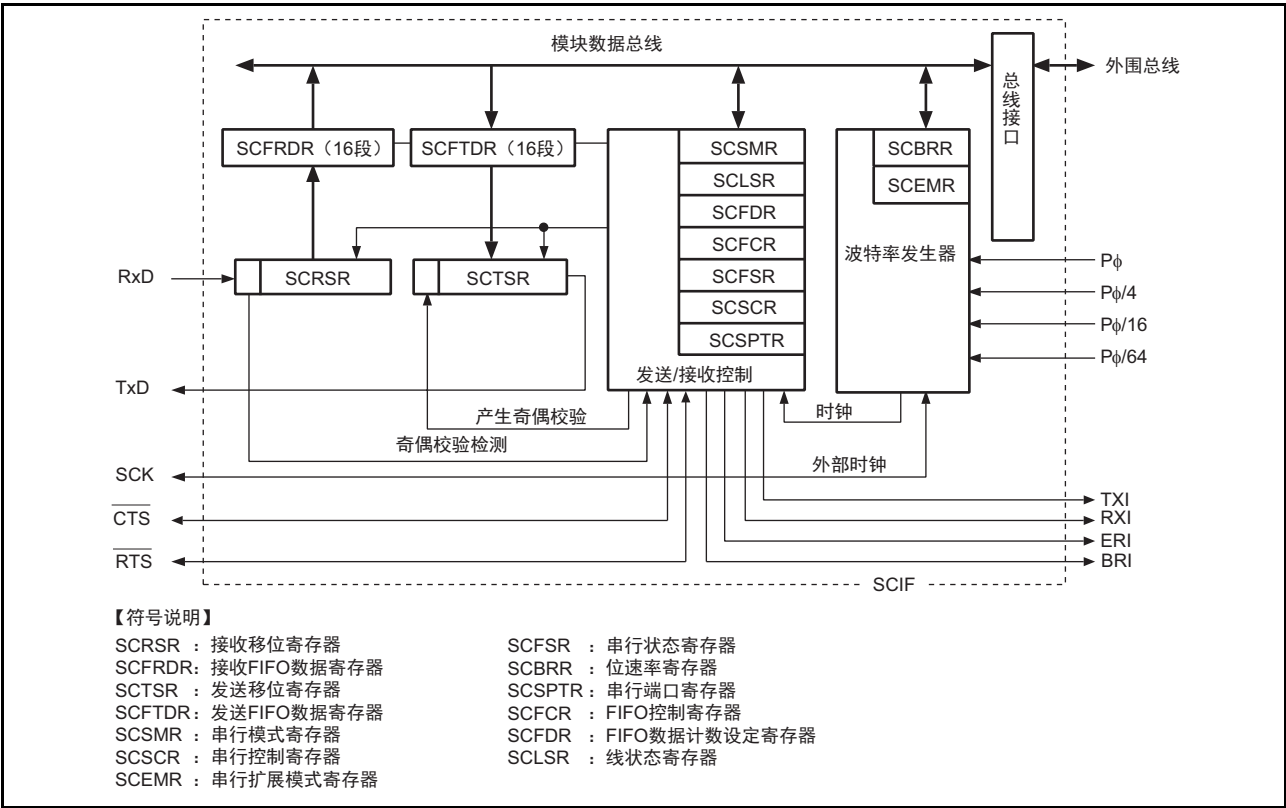


图 15.1 SCIF 框图

15.2 输入 / 输出引脚

SCIF 的引脚结构如表 15.1 所示。

表 15.1 引脚结构

通道	名称	引脚名称	输入 / 输出	功 能
0 ~ 3	串行时钟引脚	SCK0 ~ SCK3	输入 / 输出	输入 / 输出时钟
	接收数据引脚	RxD0 ~ RxD3	输入	输入接收数据
	发送数据引脚	TxD0 ~ TxD3	输出	输出发送数据
3	请求发送引脚	$\overline{\text{RTS3}}$	输入 / 输出	请求发送
	清除发送引脚	$\overline{\text{CTS3}}$	输入 / 输出	清除发送

15.3 寄存器说明

SCIF 有以下寄存器。

表 15.2 寄存器结构

通道	寄存器名称	简称	R/W	初始值	地址	存取长度
0	串行模式寄存器_0	SCSMR_0	R/W	H'0000	H'FFFE8000	16
	位速率寄存器_0	SCBRR_0	R/W	H'FF	H'FFFE8004	8
	串行控制寄存器_0	SCSCR_0	R/W	H'0000	H'FFFE8008	16
	发送 FIFO 数据寄存器_0	SCFTDR_0	W	不定	H'FFFE800C	8
	串行状态寄存器_0	SCFSR_0	R/(W)*1	H'0060	H'FFFE8010	16
	接收 FIFO 数据寄存器_0	SCFRDR_0	R	不定	H'FFFE8014	8
	FIFO 控制寄存器_0	SCFCR_0	R/W	H'0000	H'FFFE8018	16
	FIFO 数据计数设置寄存器_0	SCFDR_0	R	H'0000	H'FFFE801C	16
	串行端口寄存器_0	SCSPTR_0	R/W	H'0050	H'FFFE8020	16
	线状态寄存器_0	SCLSR_0	R/(W)*2	H'0000	H'FFFE8024	16
	串行扩展模式寄存器_0	SCEMR_0	R/W	H'0000	H'FFFE8028	16
1	串行模式寄存器_1	SCSMR_1	R/W	H'0000	H'FFFE8800	16
	位速率寄存器_1	SCBRR_1	R/W	H'FF	H'FFFE8804	8
	串行控制寄存器_1	SCSCR_1	R/W	H'0000	H'FFFE8808	16
	发送 FIFO 数据寄存器_1	SCFTDR_1	W	不定	H'FFFE880C	8
	串行状态寄存器_1	SCFSR_1	R/(W)*1	H'0060	H'FFFE8810	16
	接收 FIFO 数据寄存器_1	SCFRDR_1	R	不定	H'FFFE8814	8
	FIFO 控制寄存器_1	SCFCR_1	R/W	H'0000	H'FFFE8818	16
	FIFO 数据计数设置寄存器_1	SCFDR_1	R	H'0000	H'FFFE881C	16
	串行端口寄存器_1	SCSPTR_1	R/W	H'0050	H'FFFE8820	16
	线状态寄存器_1	SCLSR_1	R/(W)*2	H'0000	H'FFFE8824	16
	串行扩展模式寄存器_1	SCEMR_1	R/W	H'0000	H'FFFE8828	16
2	串行模式寄存器_2	SCSMR_2	R/W	H'0000	H'FFFE9000	16
	位速率寄存器_2	SCBRR_2	R/W	H'FF	H'FFFE9004	8
	串行控制寄存器_2	SCSCR_2	R/W	H'0000	H'FFFE9008	16
	发送 FIFO 数据寄存器_2	SCFTDR_2	W	不定	H'FFFE900C	8
	串行状态寄存器_2	SCFSR_2	R/(W)*1	H'0060	H'FFFE9010	16
	接收 FIFO 数据寄存器_2	SCFRDR_2	R	不定	H'FFFE9014	8
	FIFO 控制寄存器_2	SCFCR_2	R/W	H'0000	H'FFFE9018	16
	FIFO 数据计数设置寄存器_2	SCFDR_2	R	H'0000	H'FFFE901C	16
	串行端口寄存器_2	SCSPTR_2	R/W	H'0050	H'FFFE9020	16
	线状态寄存器_2	SCLSR_2	R/(W)*2	H'0000	H'FFFE9024	16
	串行扩展模式寄存器_2	SCEMR_2	R/W	H'0000	H'FFFE9028	16
3	串行模式寄存器_3	SCSMR_3	R/W	H'0000	H'FFFE9800	16
	位速率寄存器_3	SCBRR_3	R/W	H'FF	H'FFFE9804	8
	串行控制寄存器_3	SCSCR_3	R/W	H'0000	H'FFFE9808	16
	发送 FIFO 数据寄存器_3	SCFTDR_3	W	不定	H'FFFE980C	8
	串行状态寄存器_3	SCFSR_3	R/(W)*1	H'0060	H'FFFE9810	16
	接收 FIFO 数据寄存器_3	SCFRDR_3	R	不定	H'FFFE9814	8
	FIFO 控制寄存器_3	SCFCR_3	R/W	H'0000	H'FFFE9818	16
	FIFO 数据计数设置寄存器_3	SCFDR_3	R	H'0000	H'FFFE981C	16
	串行端口寄存器_3	SCSPTR_3	R/W	H'0050	H'FFFE9820	16
	线状态寄存器_3	SCLSR_3	R/(W)*2	H'0000	H'FFFE9824	16
	串行扩展模式寄存器_3	SCEMR_3	R/W	H'0000	H'FFFE9828	16

【注】 *1 为了清除标志，仅可写入 0。bit15 ~ 8、3、2 为只读位，不可写入。

*2 为了清除标志，仅可写入 0。bit15 ~ 1 为只读位，不可写入。

15.3.1 接收移位寄存器 (SCRSR)

SCRSR 为接收串行数据的寄存器。SCIF 将 RxD 引脚输入的串行数据, 按照从 LSB (bit0) 开始的接收顺序保存至 SCRSR, 并转换为并行数据。接收完 1 字节的数据后, 自动将数据传送至接收 FIFO 数据寄存器 (SCFRDR)。

CPU 不可直接读取 / 写入 SCRSR。

位:	7	6	5	4	3	2	1	0
初始值:	-	-	-	-	-	-	-	-
R/W:	-	-	-	-	-	-	-	-

15.3.2 接收 FIFO 数据寄存器 (SCFRDR)

SCFRDR 为保存已接收串行数据的 16 段 FIFO 寄存器。接收完 1 字节的串行数据时, SCIF 将已接收的串行数据从接收移位寄存器 (SCRSR) 传送至 SCFRDR 并保存, 然后完成接受运行。16 字节保存结束前, 可连续接收。CPU 从 SCFRDR 读取但不可写入。如果在接收 FIFO 数据寄存器中没有接收数据的状态下读取数据, 则读取值不定。

如果 SCFRDR 为接收数据满, 则之后接收的串行数据可能会丢失。

位:	7	6	5	4	3	2	1	0
初始值:	-	-	-	-	-	-	-	-
R/W:	R	R	R	R	R	R	R	R

15.3.3 发送移位寄存器 (SCTSR)

SCTSR 为发送串行数据的寄存器。SCIF 先将发送数据从发送 FIFO 数据寄存器 (SCFTDR) 传送至 SCTSR, 再按照从 LSB 开始的顺序发送到 TxD 引脚, 以发送串行数据。发送完 1 字节的数据后, 自动从 SCFTDR 向 SCTSR 传送下一个发送数据, 并开始发送。

CPU 不可直接读取 / 写入 SCTSR。

位:	7	6	5	4	3	2	1	0
初始值:	-	-	-	-	-	-	-	-
R/W:	-	-	-	-	-	-	-	-

15.3.4 发送 FIFO 数据寄存器（SCFTDR）

SCFTDR 为保存串行发送数据的 16 段 FIFO 寄存器。SCIF 如果检测出发送移位寄存器（SCTSR）为空，则 SCIF 将写入 SCFTDR 的发送数据传送至 SCTSR，并开始串行发送。连续执行串行发送直到 SCFTDR 发送数据为空。CPU 总是可写入 SCFTDR。

如果 SCFTDR 为发送数据满（16 字节），则不可写入下一个数据。即使写入，也被忽略。

位:	7	6	5	4	3	2	1	0
初始值:	-	-	-	-	-	-	-	-
R/W:	W	W	W	W	W	W	W	W

15.3.5 串行模式寄存器（SCSMR）

SCSMR 为设定 SCIF 的串行通信格式及选择波特率发生器时钟源的寄存器。

CPU 总是可读取 / 写入 SCSMR。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	C/Ā	CHR	PE	O/Ē	STOP	-	-	CKS[1:0]
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W

位	位名称	初始值	R/W	说 明
15 ~ 8	—	均为 0	R	保留位 读取值、写入值总是为 0。
7	C/Ā	0	R/W	通信模式 SCIF 的运行模式可选择异步模式或时钟同步模式。 0: 异步模式 1: 时钟同步模式
6	CHR	0	R/W	字符长度 异步模式的数据长度可选择 7 位或 8 位。在时钟同步模式，与 CHR 的设定无关，数据长度固定为 8 位。 0: 8 位数据 1: 7 位数据 * 【注】* 如果选择 7 位数据，发送 FIFO 数据寄存器的 MSB（bit7）就不发送。
5	PE	0	R/W	奇偶校验允许 在异步模式，选择发送时是否附加奇偶校验位及接收时是否检查奇偶校验位。在时钟同步模式，与 PE 位的设定无关，不附加及检查奇偶校验位。 0: 禁止附加及检查奇偶校验位 1: 允许附加及检查奇偶校验位 * 【注】* 如果 PE 位置 1，则发送时在发送数据后附加 O/Ē 位所指定的偶校验位或奇校验位后，再执行发送。接收时则检查接收的奇偶校验位是否为 O/Ē 位指定的偶校验位或奇校验位。

位	位名称	初始值	R/W	说 明
4	$O/\bar{E}$	0	R/W	奇偶校验模式 选择以偶校验或奇校验附加及检查奇偶校验位。仅在异步模式的 PE 位置 1 并允许附加或检查奇偶校验位时，$O/\bar{E}$ 位的设定才有效；在时钟同步模式或异步模式禁止附加或检查奇偶校验位时，$O/\bar{E}$ 位的指定无效。 0: 偶校验 *1 1: 奇校验 *2 【注】 *1 如果设定为偶校验，则发送时附加奇偶校验位，使奇偶校验位与发送字符中 1 个数为偶数，然后发送。接收时检查奇偶校验位与接收字符中 1 的个数是否为偶数。 *2 如果设定为奇校验，则发送时附加奇偶校验位，使奇偶校验位与发送字符中 1 的个数为奇数，然后发送。接收时检查奇偶校验位与接收字符中 1 的个数是否为奇数。
3	STOP	0	R/W	停止位长度 异步模式时停止位长度可选择 1 位 /2 位。STOP 位的设定仅在异步模式有效。设定为时钟同步模式时不附加停止位，因此本位的设定无效。此外，接收时，与 STOP 位的设定无关，仅检查已接收的第 1 个停止位。如果第 2 个停止位为 1，则作为停止位处理；为 0 则作为下一发送字符的开始位处理。 0: 1 个停止位 发送时，在发送字符的末尾附加 1 位的 1（停止位）后发送。 1: 2 个停止位 发送时，在发送字符的末尾附加 2 位的 1（停止位）后发送。
2	—	0	R	保留位 读取值、写入值总是为 0。
1、0	CKS[1:0]	00	R/W	选择时钟 选择内置波特率发生器的内部时钟源。有关时钟源、位速率寄存器的设定值及波特率详情参阅“15.3.8 位速率寄存器 (SCBRR)”。 00: $P\phi$ 01: $P\phi/4$ 10: $P\phi/16$ 11: $P\phi/64$ 【注】 $P\phi$: 外围时钟

15.3.6 串行控制寄存器（SCSCR）

SCSCR 为执行 SCIF 的发送 / 接收运行、允许 / 禁止中断请求及选择发送 / 接收时钟源的寄存器。CPU 总是可读取 / 写入 SCSCR。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	TIE	RIE	TE	RE	REIE	-	CKE[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W

位	位名称	初始值	R/W	说 明
15 ~ 8	—	均为 0	R	保留位 读取值、写入值总是为 0。
7	TIE	0	R/W	发送中断允许 从发送 FIFO 数据寄存器（SCFTDR）向发送移位寄存器（SCTSR）传送串行发送数据，并且发送 FIFO 寄存器的数据个数少于指定的发送触发个数，串行状态寄存器（SCFSR）的 TDFE 标志置 1 时，允许 / 禁止产生 FIFO 数据空中断（TXI）请求。 0: 禁止发送 FIFO 数据空中断（TXI）请求 1: 允许发送 FIFO 数据空中断（TXI）请求 * 【注】 * 通过写入大于 SCFTDR 指定的发送触发个数的发送数据，读取 TDFE 标志的 1 后清 0，或通过将 TIE 清 0，解除 TXI。
6	RIE	0	R/W	接收中断允许 允许 / 禁止产生以下请求：串行状态寄存器（SCFSR）的 RDF 标志或 DR 标志置 1 时的接收 FIFO 数据满中断（RXI）请求；SCFSR 的 ER 标志置 1 时的接收错误中断（ERI）请求；及 SCFSR 的 BRK 标志或线状态寄存器（SCLSR）的 ORER 标志置 1 时的断点中断（BRI）请求。 0: 禁止接收 FIFO 数据满中断（RXI）请求、接收错误中断（ERI）请求及断点中断（BRI）请求 1: 允许接收 FIFO 数据满中断（RXI）请求、接收错误中断（ERI）请求及断点中断（BRI）请求 * 【注】 * 通过读取 DR 标志或 RDF 标志的 1 后清 0，或将 RIE 位清 0，可解除 RXI 中断请求。通过读取 ER、BRK 或 ORER 标志的 1 后清 0，或将 RIE、REIE 位清 0，可解除 ERI、BRI 中断请求。
5	TE	0	R/W	发送允许 允许 / 禁止开始串行发送。 0: 禁止发送 1: 允许发送 * 【注】 * 在此状态，如果对 SCFTDR 写入发送数据，则开始串行发送。 TE 位置 1 前，必须设定 SCSMR 及 SCFCR 确定发送格式，并复位发送 FIFO。

位	位名称	初始值	R/W	说 明
4	RE	0	R/W	接收允许 允许 / 禁止开始串行接收。 0: 禁止接收 *1 1: 允许接收 *2 【注】 *1 必须注意: 即使 RE 位清 0, DR、ER、BRK、RDF、FER、PER 及 ORER 等位也不受影响而保持原状态。 *2 在此状态, 如果在异步模式时检测出开始位或时钟同步模式检测出同步时钟, 则开始串行接收。RE 位置 1 前, 必须设定串行模式寄存器 (SCSMR) 及 FIFO 控制寄存器 (SCFCR), 确定接收格式, 并复位接收 FIFO。
3	REIE	0	R/W	接收错误中断允许 允许 / 禁止产生接收错误中断 (ERI) 请求及断点中断 (BRI) 请求。但 REIE 位的设定仅在 RIE 位为 0 时有效。 0: 禁止接收错误中断 (ERI) 请求及断点中断 (BRI) 请求 1: 允许接收错误中断 (ERI) 请求及断点中断 (BRI) 请求 * 【注】 * 通过读取 ER、BRK 或 ORER 标志的 1 后清 0, 或将 RIE、REIE 位清 0, 可解除 ERI、BRI 中断请求。如果 REIE 位设定为 1, 则即使 RIE 位设定为 0, 也产生 ERI 中断或 BRI 中断。需在 DMAC 传送时向中断控制器通知时设定。
2	—	0	R	保留位 读取值、写入值总是为 0。
1、0	CKE[1:0]	00	R/W	时钟允许 选择 SCIF 时钟源及禁止 / 允许 SCK 引脚的输出时钟。由 CKE[1:0] 决定 SCK 引脚作为串行时钟的输出引脚或是串行时钟的输入引脚。在时钟同步模式设定为同步时钟输出时, 必须在 SCSMR 的 C/A 位设定为 1 后, 再设定 CKE[1:0]。 - 异步模式 00: 内部时钟 /SCK 引脚为输入引脚 (忽略输入信号) 01: 内部时钟 /SCK 引脚为时钟输出引脚 (输出频率为 16/8 倍位倍率的时钟) 10: 外部时钟 /SCK 引脚为时钟输入引脚 (输入频率为 16/8 倍位倍率的时钟) 11: 禁止设定 - 时钟同步模式 00: 内部时钟 /SCK 引脚为同步时钟输出引脚 01: 内部时钟 /SCK 引脚为同步时钟输出引脚 10: 外部时钟 /SCK 引脚为同步时钟输入引脚 11: 禁止设定

15.3.7 串行状态寄存器（SCFSR）

SCFSR 为 16 位寄存器，高 8 位表示接收 FIFO 数据寄存器数据的接收错误个数，低 8 位表示 SCIF 运行状态的状态标志。

CPU 总是可读取 / 写入 SCFSR。但，不可对 ER、TEND、TDFE、BRK、RDF 及 DR 等状态标志写入 1。另外，要将这些标志清 0，必须先读取 1。PER 标志（bit15 ~ 12、2）及 FER 标志（bit11 ~ 8、3）为只读位，不可写入。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PER[3:0]				FER[3:0]				ER	TEND	TDFE	BRK	FER	PER	RDF	DR
初始值:	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R	R	R/(W)*	R/(W)*

【注】* 为了清除标志，仅可在读取1后写入0。

位	位名称	初始值	R/W	说 明
15 ~ 12	PER[3:0]	0000	R	奇偶校验错误个数 表示保存到接收 FIFO 数据寄存器（SCFRDR）的接收数据中产生奇偶校验错误的的数据个数。 SCFSR 的 ER 置位后，bit15 ~ 12 所示的值表示产生奇偶校验错误的的数据个数。SCFRDR 的 16 字节接收数据均产生奇偶校验错误时，则 PER[3:0] 表示 0000。
11 ~ 8	FER[3:0]	0000	R	帧错误个数 表示保存到接收 FIFO 数据寄存器（SCFRDR）的接收数据中产生帧错误的的数据个数。SCFSR 的 ER 位置位后，bit11 ~ 8 所示的值表示产生帧错误的的数据个数。SCFRDR 的 16 字节接收数据均产生帧错误时，则 FER[3:0] 表示 0000。
7	ER	0	R/(W)*	接收错误 表示产生帧错误或接收含奇偶性的数据时产生奇偶校验错误。*1 0: 表示正在接收或正常完成接收 [清除条件] - 上电复位 - 读取 ER=1 的状态后写入 0 时 1: 表示接收时产生帧错误或者奇偶校验错误 [置位条件] 1 次数据接收结束时检查接收数据的最后停止位是否为 1，如果停止位为 0 时 *2 - 接收时接收数据和奇偶校验位的 1 的总数与串行模式寄存器（SCSMR）的 O/E 位指定的偶 / 奇校验的设定不匹配时 【注】 *1 SCSCR 的 RE 位清 0 时，ER 位不受影响仍保持原状态。即使产生接收错误，仍将接收数据传送到 SCFRDR 并继续接收。可由 SCFSR 的 FER 位或 PER 位判断从 SCFRDR 读取的数据是否包含接收错误。 *2 在 2 个停止位的模式，仅检查第 1 个停止位而不检查第 2 个停止位。

位	位名称	初始值	R/W	说 明
6	TEND	1	R/(W)*	发送结束 表示发送字符的最末尾位被发送时 SCFTDR 中无有效数据而结束发送。 0: 表示正在发送 [清除条件] - 向 SCFTDR 写入发送数据、读取 TEND=1 的状态后, 给 TEND 标志写入 0 时 *1 1: 表示发送结束 [置位条件] - 上电复位 - SCSCR 的 TE 位为 0 时 - 发送 1 字节串行发送字符的最末尾位时, SCFTDR 中无发送数据时 【注】 *1 由 DMAC 通过 TXI 中断请求将数据写入 SCFTDR 时, 不得用作发送结束标志。
5	TDFE	1	R/(W)*	发送 FIFO 数据空 表示从发送 FIFO 数据寄存器 (SCFTDR) 将数据传送到发送移位寄存器 (SCTSR), SCFTDR 的数据个数小于 FIFO 控制寄存器 (SCFCR) 的 TTRG[1:0] 所指定的发送触发数据个数, 允许将发送数据写入 SCFTDR。 0: 表示写入 SCFTDR 的发送数据个数大于指定的发送触发个数 [清除条件] - 读取 TDFE=1 的状态后, 将大于指定发送触发个数的数据写入 SCFTDR, 并对 TDFE 写入 0 时 - 由发送 FIFO 数据空中断 (TXI) 启动 DMAC, 将大于指定发送触发个数的数据写入 SCFTDR 时 1: 表示写入 SCFTDR 的发送数据个数小于等于指定发送触发个数 *1 [置位条件] - 上电复位 - 写入发送结果 SCFTDR 的发送数据个数小于等于指定发送触发个数时 【注】 *1 SCFTDR 为 16 字节 FIFO 寄存器, 因此在 TDFE=1 的状态可写入数据的最大数据个数为 “16 减去指定发送触发个数之后的数”。即使想写入更多的数据, 仍被忽略。SCFTDR 的数据个数显示于 SCFDR 的高 8 位。
4	BRK	0	R/(W)*	断点检测 表示接收数据中检测出断点信号。 0: 无断点信号 [清除条件] - 上电复位 - 读取 BRK=1 的状态后对 BRK 标志写入 0 时 1: 接收断点信号 *1 [置位条件] - 接收包含帧错误的数据, 随后有大于等于 1 帧长度的空间 0 (低电平) 时 【注】 *1 检测断点时, 在检测出后停止向 SCFRDR 传送接收数据 (H'00)。断点结束且接收信号为标记 1 时, 重新开始传送接收数据。

位	位名称	初始值	R/W	说 明
3	FER	0	R	表示帧错误 表示在异步模式从接收 FIFO 数据寄存器 (SCFRDR) 读取的数据中是否有帧错误。 0: 表示在下一个要从 SCFRDR 读取的接收数据中未产生帧错误 [清除条件] • 上电复位 • 下一个要从 SCFRDR 读取的数据中无帧错误 1: 表示在下一个要从 SCFRDR 读取的接收数据中产生帧错误 [置位条件] • 下一个要从 SCFRDR 读取的数据中有帧错误
2	PER	0	R	表示奇偶校验错误 表示在异步模式从接收 FIFO 数据寄存器 (SCFRDR) 读取的数据中是否有奇偶校验错误。 0: 表示在下一个要从 SCFRDR 读取的接收数据中未产生奇偶校验错误 [清除条件] • 上电复位 • 下一个要从 SCFRDR 读取的数据中无奇偶校验错误 1: 表示在下一个要从 SCFRDR 读取的接收数据中产生奇偶校验错误 [置位条件] • 下一个要从 SCFRDR 读取的数据中有奇偶校验错误
1	RDF	0	R/(W)*	接收 FIFO 数据满 表示接收数据被传送到接收 FIFO 数据寄存器 (SCFRDR)，且 SCFRDR 的数据个数大于 FIFO 控制寄存器 (SCFCR) 的 RTRG[1:0] 位所指定的接收触发个数。 0: 表示写入 SCFRDR 的接收数据个数小于指定的接收触发个数 [清除条件] • 上电复位 • 读取 RDF=1 后，读取 SCFRDR 直到 SCFRDR 的接收数据个数小于指定接收触发个数，并对 RDF 写入 0 时 • 由接收 FIFO 数据满中断 (RXI) 启动 DMAC，读取 SCFRDR 直到 SCFRDR 的接收数据个数小于指定接收触发个数时 1: 表示 SCFRDR 的接收数据个数大于等于指定的接收触发个数 [置位条件] • 将大于等于指定接收触发个数的接收数据个数保存在 SCFRDR 时 *1 【注】 *1 SCFRDR 为 16 字节 FIFO 寄存器，因此 RDF 为 1 时，可读取的最大数据个数为指定接收触发个数。读取 SCFRDR 的全部数据后，如果继续读取数据，则数据为不定。由 SCFDR 的低 8 位显示 SCFRDR 的接收数据个数。

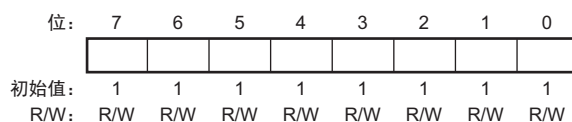
位	位名称	初始值	R/W	说 明
0	DR	0	R/(W)*	接收数据就绪 表示在异步模式将小于指定接收触发个数的数据保存到接收 FIFO 数据寄存器 (SCFRDR)，且从最后的停止位经过 15ETU 的时间后还未接收到下一数据。在时钟同步模式，此位不置位。 0：表示正在接收或正常完成接收后 SCFRDR 中未剩余接收数据 [清除条件] • 上电复位 • 读取 DR=1 的状态后，读取 SCFRDR 的全部接收数据，并写入 0 时 • 由接收 FIFO 数据满中断 (RXI) 启动 DMAC，读取 SCFRDR 的全部接收数据时 1：表示未接收到下一接收数据 [置位条件] • 将小于指定接收触发个数的数据保存到 SCFRDR，且从最后的停止位经过 15ETU 的时间 *1 后仍未接收到下一数据时 【注】 *1 相当于 8 位、1 个停止位格式的 1.5 帧。(ETU: Element Time Unit : 时间素单元)

【注】 * 为了清除标志，仅可在读取 1 后写入 0。

15.3.8 位速率寄存器 (SCBRR)

SCBRR 为 8 位寄存器，与串行模式寄存器 (SCSMR) 的 CKS[1:0]、串行扩展模式寄存器 (SCEMR) 的 BGDM 位及 ABCS 位结合，设定串行发送 / 接收的位速率。

CPU 总是可读取 / 写入 SCBRR。上电复位时，SCBRR 初始化为 H'FF。各通道具有独立的波特率发生器控制，因此，可对 3 个通道设定不同的值。



通过以下公式求 SCBRR 的设定值：

【异步模式】

- 波特率发生器普通模式时 (SCEMR 的 BGDM=0)

$$N = (P\phi / (64 \times 2^{2n} - 1 \times B)) \times 10^6 - 1$$

(以频率为 16 倍位速率的基本时钟运行时)

$$N = (P\phi / (32 \times 2^{2n} - 1 \times B)) \times 10^6 - 1$$

(以频率为 8 倍位速率的基本时钟运行时)

- 波特率发生器倍速模式时 (SCEMR 的 BGDM=1)

$$N = (P\phi / (32 \times 2^{2n} - 1 \times B)) \times 10^6 - 1$$

(以频率为 16 倍位速率的基本时钟运行时)

$$N = (P\phi / (16 \times 2^{2n} - 1 \times B)) \times 10^6 - 1$$

(以频率为 8 倍位速率的基本时钟运行时)

【时钟同步模式】

$$N = \frac{P\phi}{8 \times 2^{2n-1} \times B} \times 10^6 - 1$$

B : 位速率 (bit/s)

N : 波特率发生器 SCBRR 的设定值 (0 ≤ N ≤ 255)

(必须设定为满足电特性的值。)

Pφ: 外围模块的工作频率 (MHz)

n : 波特率发生器输入时钟 (n=0、1、2、3)

(n 与时钟的关系参照表 15.3)

表 15.3 SCSMR 的设置

n	时钟	SCSMR 的设置值	
		CKS[1]	CKS[0]
0	$P\phi$	0	0
1	$P\phi/4$	0	1
2	$P\phi/16$	1	0
3	$P\phi/64$	1	1

通过以下公式，可求得异步模式的位速率误差：

- 波特率发生器普通模式 (SCSMR 的 BGDM=0)
 - 误差 (%) = $\{(P\phi \times 10^6)/(N+1) \times B \times 64 \times 2^{2n-1}) - 1\} \times 100$
(以频率为 16 倍位速率的基本时钟运行时)
 - 误差 (%) = $\{(P\phi \times 10^6)/(N+1) \times B \times 32 \times 2^{2n-1}) - 1\} \times 100$
(以频率为 8 倍位速率的基本时钟运行时)
- 波特率发生器倍速模式时 (SCSMR 的 BGDM=1)
 - 误差 (%) = $\{(P\phi \times 10^6)/(N+1) \times B \times 32 \times 2^{2n-1}) - 1\} \times 100$
(以频率为 16 倍位速率的基本时钟运行时)
 - 误差 (%) = $\{(P\phi \times 10^6)/(N+1) \times B \times 16 \times 2^{2n-1}) - 1\} \times 100$
(以频率为 8 倍位速率的基本时钟运行时)

波特率发生器普通模式 (SCSMR 的 BGDM=0)、16 倍的基本时钟运行 (SCSMR 的 ABCS=0) 时，异步模式的 SCBRR 的设置例如表 15.4 所示；时钟同步模式的 SCBRR 的设置例如表 15.5 所示。

表 15.4 位速率的 SCBRR 的设置例 (异步模式、BGDM=0、ABCS=0) (1)

位速率 (bit/s)	$P\phi$ (MHz)											
	8			9.8304			10			12		
	n	N	误差 (%)	n	N	误差 (%)	n	N	误差 (%)	n	N	误差 (%)
110	2	141	0.03	2	174	- 0.26	2	177	- 0.25	2	212	0.03
150	2	103	0.16	2	127	0.00	2	129	0.16	2	155	0.16
300	1	207	0.16	1	255	0.00	2	64	0.16	2	77	0.16
600	1	103	0.16	1	127	0.00	1	129	0.16	1	155	0.16
1200	0	207	0.16	0	255	0.00	1	64	0.16	1	77	0.16
2400	0	103	0.16	0	127	0.00	0	129	0.16	0	155	0.16
4800	0	51	0.16	0	63	0.00	0	64	0.16	0	77	0.16
9600	0	25	0.16	0	31	0.00	0	32	- 1.36	0	38	0.16
19200	0	12	0.16	0	15	0.00	0	15	1.73	0	19	0.16
31250	0	7	0.00	0	9	-1.70	0	9	0.00	0	11	0.00
38400	0	6	- 6.99	0	7	0.00	0	7	1.73	0	9	- 2.34

表 15.4 位速率的 SCBRR 的设定例 (异步模式、BGDM=0、ABCS=0) (2)

位速率 (bit/s)	P ϕ (MHz)											
	12.288			14.7456			16			19.6608		
	n	N	误差 (%)	n	N	误差 (%)	n	N	误差 (%)	n	N	误差 (%)
110	2	217	0.08	3	64	0.70	3	70	0.03	3	86	0.31
150	2	159	0.00	2	191	0.00	2	207	0.16	2	255	0.00
300	2	79	0.00	2	95	0.00	2	103	0.16	2	127	0.00
600	1	159	0.00	1	191	0.00	1	207	0.16	1	255	0.00
1200	1	79	0.00	1	95	0.00	1	103	0.16	1	127	0.00
2400	0	159	0.00	0	191	0.00	0	207	0.16	0	255	0.00
4800	0	79	0.00	0	95	0.00	0	103	0.16	0	127	0.00
9600	0	39	0.00	0	47	0.00	0	51	0.16	0	63	0.00
19200	0	19	0.00	0	23	0.00	0	25	0.16	0	31	0.00
31250	0	11	2.40	0	14	-1.70	0	15	0.00	0	19	-1.70
38400	0	9	0.00	0	11	0.00	0	12	0.16	0	15	0.00

表 15.4 位速率的 SCBRR 的设定例 (异步模式、BGDM=0、ABCS=0) (3)

位速率 (bit/s)	P ϕ (MHz)											
	20			24			24.576			28.7		
	n	N	误差 (%)	n	N	误差 (%)	n	N	误差 (%)	n	N	误差 (%)
110	3	88	-0.25	3	106	-0.44	3	108	0.08	3	126	0.31
150	3	64	0.16	3	77	0.16	3	79	0.00	3	92	0.46
300	2	129	0.16	2	155	0.16	2	159	0.00	2	186	-0.08
600	2	64	0.16	2	77	0.16	2	79	0.00	2	92	0.46
1200	1	129	0.16	1	155	0.16	1	159	0.00	1	186	-0.08
2400	1	64	0.16	1	77	0.16	1	79	0.00	1	92	0.46
4800	0	129	0.16	0	155	0.16	0	159	0.00	0	186	-0.08
9600	0	64	0.16	0	77	0.16	0	79	0.00	0	92	0.46
19200	0	32	-1.36	0	38	0.16	0	39	0.00	0	46	-0.61
31250	0	19	0.00	0	23	0.00	0	24	-1.70	0	28	-1.03
38400	0	15	1.73	0	19	-2.34	0	19	0.00	0	22	1.55

表 15.4 位速率的 SCBRR 的设定例 (异步模式、BGDM=0、ABCS=0) (4)

位速率 (bit/s)	P ϕ (MHz)					
	30			33		
	n	N	误差 (%)	n	N	误差 (%)
110	3	132	0.13	3	145	0.33
150	3	97	-0.35	3	106	0.39
300	2	194	0.16	2	214	-0.07
600	2	97	-0.35	2	106	0.39
1200	1	194	0.16	1	214	-0.07
2400	1	97	-0.35	1	106	0.39
4800	0	194	-1.36	0	214	-0.07
9600	0	97	-0.35	0	106	0.39
19200	0	48	-0.35	0	53	-0.54
31250	0	29	0.00	0	32	0.00
38400	0	23	1.73	0	26	-0.54

【注】 设定时尽量保证误差在 1% 以内。

表 15.5 位速率对应的 SCBRR 设定例 (时钟同步模式)

位速率 (bit/s)	P ϕ (MHz)									
	8		16		28.7		30		33	
	n	N	n	N	n	N	n	N	n	N
110	—	—	—	—	—	—	—	—	—	—
250	3	124	3	249	—	—	—	—	—	—
500	2	249	3	124	3	223	3	233	3	255
1k	2	124	2	249	3	111	3	116	3	125
2.5k	1	199	2	99	2	178	2	187	2	200
5k	1	99	1	199	2	89	2	93	2	100
10k	0	199	1	99	1	178	1	187	1	200
25k	0	79	0	159	1	71	1	74	1	80
50k	0	39	0	79	0	143	0	149	0	160
100k	0	19	0	39	0	71	0	74	0	80
250k	0	7	0	15	—	—	0	29	0	31
500k	0	3	0	7	—	—	0	14	0	15
1M	0	1	0	3	—	—	—	—	0	7
2M	0	0*	0	1	—	—	—	—	—	—

【符号说明】

空白栏：不可设定。

—：可设定，但会出现误差。

*：不可连续发送 / 接收。

使用波特率发生器时，异步模式各频率的最大位速率如表 15.6 所示。另外，输入外部时钟时，异步模式的最大位速率如表 15.7 所示。输入外部时钟时 ($t_{\text{Scyc}}=12t_{\text{pcyc}}$ 时 *) 时，时钟同步模式的最大位速率如表 15.8 所示。

【注】 * 必须确认满足本 LSI 与通信对象的电特性。

表 15.6 使用波特率发生器时各频率的最大位速率 (异步模式)

P ϕ (MHz)	设定值				最大位速率 (bit/s)
	BGDM	ABCS	n	N	
8	0	0	0	0	250000
		1	0	0	500000
	1	0	0	0	500000
		1	0	0	1000000
9.8304	0	0	0	0	307200
		1	0	0	614400
	1	0	0	0	614400
		1	0	0	1228800
12	0	0	0	0	375000
		1	0	0	750000
	1	0	0	0	750000
		1	0	0	1500000
14.7456	0	0	0	0	460800
		1	0	0	921600
	1	0	0	0	921600
		1	0	0	1843200
16	0	0	0	0	500000
		1	0	0	1000000
	1	0	0	0	1000000
		1	0	0	2000000
19.6608	0	0	0	0	614400
		1	0	0	1228800
	1	0	0	0	1228800
		1	0	0	2457600
20	0	0	0	0	625000
		1	0	0	1250000
	1	0	0	0	1250000
		1	0	0	2500000

P _φ (MHz)	设定值				最大位速率 (bit/s)
	BGDM	ABCS	n	N	
24	0	0	0	0	750000
		1	0	0	1500000
	1	0	0	0	1500000
		1	0	0	3000000
24.576	0	0	0	0	768000
		1	0	0	1536000
	1	0	0	0	1536000
		1	0	0	3072000
28.7	0	0	0	0	896875
		1	0	0	1793750
	1	0	0	0	1793750
		1	0	0	3587500
30	0	0	0	0	937500
		1	0	0	1875000
	1	0	0	0	1875000
		1	0	0	3750000
33	0	0	0	0	1031250
		1	0	0	2062500
	1	0	0	0	2062500
		1	0	0	4125000

表 15.7 输入外部时钟时的最大位速率 (异步模式)

P ϕ (MHz)	外部输入时钟 (MHz)	设定值	最大位速率 (bit/s)
		ABCS	
8	2.0000	0	125000
		1	250000
9.8304	2.4576	0	153600
		1	307200
12	3.0000	0	187500
		1	375000
14.7456	3.6864	0	230400
		1	460800
16	4.0000	0	250000
		1	500000
19.6608	4.9152	0	307200
		1	614400
20	5.0000	0	312500
		1	625000
24	6.0000	0	375000
		1	750000
24.576	6.1440	0	384000
		1	768000
28.7	4.9152	0	448436
		1	896872
30	7.5000	0	468750
		1	937500
33	8.2500	0	515625
		1	1031250

表 15.8 输入外部时钟时的最大位速率 (时钟同步模式、 $t_{\text{Scyc}}=12t_{\text{pcyc}}$ 时)

P ϕ (MHz)	外部输入时钟 (MHz)	最大位速率 (bit/s)
8	0.6666	666666.6
16	1.3333	1333333.3
24	2.0000	2000000.0
28.7	2.3916	2391666.6
30	2.5000	2500000.0
33	2.7500	2750000.0

15.3.9 FIFO 控制寄存器 (SCFCR)

SCFCR 为复位发送及接收 FIFO 数据寄存器的数据个数，并设定触发数据个数的寄存器。SCFCR 还具有环回测试的允许位。

CPU 总是可读取 / 写入 SCFCR。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	RSTRG[2:0]		RTRG[1:0]		TTRG[1:0]		MCE	TFRST	RFRST	LOOP	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15 ~ 11	—	均为 0	R	保留位 读取值、写入值总是为 0。
10 ~ 8	RSTRG[2:0]	000	R/W	RTS 输出有效触发 接收 FIFO 数据寄存器 (SCFRDR) 保存的接收数据个数大于等于下表所示的触发设定个数时， $\overline{\text{RTS}}$ 信号为高电平。 000: 15 001: 1 010: 4 011: 6 100: 8 101: 10 110: 12 111: 14
7、6	RTRG[1:0]	00	R/W	接收 FIFO 数据个数的触发 设定接收数据个数 (指定接收触发个数)，作为串行状态寄存器 (SCFSR) 的 RDF 标志置位的基准。如果接收 FIFO 数据寄存器 (SCFRDR) 保存的接收数据个数大于等于 以下所示的设定触发个数，则 RDF 标志置 1。 • 异步模式 00: 1 01: 4 10: 8 11: 14 • 时钟同步模式 00: 1 01: 2 10: 8 11: 14 【注】 在时钟同步模式时，如果通过 DMAC 传送接收数据，则必须将接收触发个数设定为 1，否则，则需要通过 CPU 读取 SCFRDR 中剩余的接收数据。

位	位名称	初始值	R/W	说 明
5、4	TTRG[1:0]	00	R/W	发送 FIFO 数据个数的触发 设定发送数据个数（指定发送触发个数），作为串行状态寄存器（SCFSR）的 TDFE 标志置位的基准。如果发送 FIFO 数据寄存器（SCFTDR）保存的发送数据个数小于等于以下所示的设定触发个数，则 TDFE 标志置 1。 00: 8（8）* 01: 4（12）* 10: 2（14）* 11: 0（16）* 【注】* （）内的数值表示 TDFE 标志置 1 时 SCFTDR 寄存器的空字节数。
3	MCE	0	R/W	调制解调器控制允许 允许 / 禁止调制解调器的控制信号 $\overline{\text{CTS}}$ 与 $\overline{\text{RTS}}$。 在时钟同步模式，通道 0 ~ 2 的 MCE 总是为 0。 0: 禁止调制解调器信号 * 1: 允许调制解调器信号 【注】* 与输入值无关，$\overline{\text{CTS}}$ 电平不影响发送运行；$\overline{\text{RTS}}$ 电平不影响接收运行。
2	TFRST	0	R/W	发送 FIFO 数据寄存器复位 将发送 FIFO 数据寄存器的发送数据设置为无效，并将数据复位为空状态。 0: 禁止复位 * 1: 允许复位 【注】* 上电复位时执行复位。
1	RFRST	0	R/W	接收 FIFO 数据寄存器复位 将接收 FIFO 数据寄存器的接收数据设置为无效，并将数据复位为空状态。 0: 禁止复位 * 1: 允许复位 【注】* 上电复位时执行复位。
0	LOOP	0	R/W	环回测试 允许在内部将发送输出引脚（TxD）与接收输入引脚（RxD）、$\overline{\text{RTS}}$ 引脚与 $\overline{\text{CTS}}$ 引脚连接，并执行环回测试。 0: 禁止环回测试 1: 允许环回测试

15.3.10 FIFO 数据计数设定寄存器（SCFDR）

SCFDR 表示保存在发送 FIFO 数据寄存器（SCFTDR）与接收 FIFO 数据寄存器（SCFRDR）的数据个数。

高 8 位表示 SCFTDR 的发送数据个数，低 8 位表示 SCFRDR 的接收数据个数。CPU 总是可读取 SCFDR。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	T[4:0]				-	-	-	R[4:0]					
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名称	初始值	R/W	说 明
15 ~ 13	—	均为 0	R	保留位 读取值、写入值总是为 0。
12 ~ 8	T[4:0]	0000	R	表示保存在 SCFTDR 的未发送数据的个数。 H'00 表示无发送数据，H'10 表示所有发送数据均保存在 SCFTDR。
7 ~ 5	—	均为 0	R	保留位 读取值、写入值总是为 0。
4 ~ 0	R[4:0]	0000	R	表示保存在 SCFRDR 的接收数据的个数。 H'00 表示无接收数据，H'10 表示所有接收数据均保存在 SCFRDR。

15.3.11 串行端口寄存器 (SCSPTR)

SCSPTR 控制被多路复用到 SCIF 的引脚的数据及端口的输入 / 输出。由 bit7、6 可对 $\overline{\text{RTS}}$ 引脚读取数据或写入输出数据。由 bit5、4 可对 $\overline{\text{CTS}}$ 引脚读取数据或写入输出数据。由 bit3、2 可对 SCK 引脚读取数据或写入输出数据。由 bit1、0 可从 RxD 引脚读取输入数据并向 TxD 引脚写入输出数据，从而控制串行发送 / 接收的断点。

CPU 总是可读取 / 写入 SCSPTR。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	RTSIO	RTSDT	CTSIO	CTSDT	SCKIO	SCKDT	SPB2IO	SPB2DT
初始值:	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15 ~ 8	—	均为 0	R	保留位 读取值、写入值总是为 0。
7	RTSIO	0	R/W	$\overline{\text{RTS}}$ 端口输入 / 输出 指定串行端口 $\overline{\text{RTS}}$ 引脚的输入 / 输出。实际将 $\overline{\text{RTS}}$ 引脚作为端口输出引脚，输出 RTSDT 位设定的值时，必须将 SCFCR 的 MCE 位设定为 0。 0: 不将 RTSDT 位的值输出至 $\overline{\text{RTS}}$ 引脚 1: 将 RTSDT 位的值输出至 $\overline{\text{RTS}}$ 引脚
6	RTSDT	1	R/W	$\overline{\text{RTS}}$ 端口数据 指定串行端口 $\overline{\text{RTS}}$ 引脚的输入 / 输出数据。由 RTSIO 位指定是输入或是输出。输出时，将 RTSDT 位的值输出至 $\overline{\text{RTS}}$ 引脚。与 RTSIO 位的值无关，从 RTSDT 位读取 $\overline{\text{RTS}}$ 引脚的值。但必须预先由 PFC 设定 $\overline{\text{RTS}}$ 输入 / 输出。 0: 输入 / 输出数据为低电平 1: 输入 / 输出数据为高电平
5	CTSIO	0	R/W	$\overline{\text{CTS}}$ 端口输入 / 输出 指定串行端口 $\overline{\text{CTS}}$ 引脚的输入 / 输出。实际将 $\overline{\text{CTS}}$ 引脚作为端口输出引脚输出 CTSDT 位设定的值时，必须将 SCFCR 的 MCE 位设定为 0。 0: 不将 CTSDT 位的值输出至 $\overline{\text{CTS}}$ 引脚 1: 将 CTSDT 位的值输出至 $\overline{\text{CTS}}$ 引脚
4	CTSDT	1	R/W	$\overline{\text{CTS}}$ 端口数据 指定串行端口 $\overline{\text{CTS}}$ 引脚的输入 / 输出数据。由 CTSIO 位指定是输入或是输出。输出时，CTSDT 位的值输出至 $\overline{\text{CTS}}$ 引脚。与 CTSIO 位的值无关，从 CTSDT 位读取 $\overline{\text{CTS}}$ 引脚的值。但必须预先由 PFC 预先设定 $\overline{\text{CTS}}$ 的输入 / 输出。 0: 输入 / 输出数据为低电平 1: 输入 / 输出数据为高电平
3	SCKIO	0	R/W	SCK 端口输入 / 输出 指定串行端口 SCK 引脚的输入 / 输出。实际将 SCK 引脚作为端口输出引脚输出 SCKDT 位设定的值时，必须将 SCSCR 的 CK1、CK0 位设定为 0。 0: 不将 SCKDT 位的值输出至 SCK 引脚 1: 将 SCKDT 位的值输出至 SCK 引脚
2	SCKDT	0	R/W	SCK 端口数据 指定串行端口 SCK 引脚的输入 / 输出数据。由 SCKIO 位指定是输入或是输出。输出时，SCKDT 位的值输出至 SCK 引脚。与 SCKIO 位的值无关，从 SCKDT 位读取 SCK 引脚的值。但必须预先由 PFC 设定 SCK 输入 / 输出。 0: 输入 / 输出数据为低电平 1: 输入 / 输出数据为高电平

位	位名称	初始值	R/W	说 明
1	SPB2IO	0	R/W	串行端口断点输入 / 输出 指定串行端口 TxD 引脚的输出条件。实际将 TxD 引脚作为端口输出引脚输出 SPB2DT 位设定的值时，必须将 SCSCR 的 TE 位设定为 0。 0：不将 SPB2DT 位的值输出至 TxD 引脚 1：将 SPB2DT 位的值输出至 TxD 引脚
0	SPB2DT	0	R/W	串行端口断点数据 指定串行端口 RxD 引脚的输入数据及 TxD 引脚的输出数据。由 SPB2IO 位指定是输入或是输出。TxD 引脚设定为输出时，将 SPB2DT 位的值输出至 TxD 引脚。与 SPB2IO 位的值无关，从 SPB2DT 位读取 RxD 引脚的值。但必须预先由 PFC 设定 RxD 输入、TxD 输出。 0：输入 / 输出数据为低电平 1：输入 / 输出数据为高电平

15.3.12 线状态寄存器（SCLSR）

CPU 总是可读取 / 写入 SCLSR。但不可对 ORER 的状态标志写入 1。为了清 0 必须先读取 1。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	ORER
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/(W)*

【注】* 为了清除标志，仅可在读取 1 后写入 0。

位	位名称	初始值	R/W	说 明
15 ~ 1	—	均为 0	R	保留位 读取值、写入值总是为 0。
0	ORER	0	R/(W)*	溢出错误 表示接收时产生溢出错误并异常结束。 0：表示正在接收或接收正常结束 *1 [清除条件] • 上电复位 • 读取 ORER=1 的状态后写入 0 时 1：表示接收时产生溢出错误 *2 [置位条件] • 在接收 FIFO 均为 16 字节数据的状态，完成下一个串行接收时 【注】 *1 串行控制寄存器（SCSCR）的 RE 位清 0 时，ORER 标志不受影响仍保持之前的状态。 *2 接收 FIFO 数据寄存器（SCFRDR）保持产生溢出错误前的接收数据，丢失产生溢出错误后的接收数据。在 ORER=1 的状态，不可继续执行之后的串行接收。

15.3.13 串行扩展模式寄存器（SCEMR）

CPU 总是可读取 / 写入 SCEMR。通过将 BGDM 位置 1，异步模式（SCSMR 的 $\overline{C/A}=0$ ）且将内部时钟作为时钟源、SCK 引脚用作输入引脚（SCSCR 的 $CKE[1:0]=00$ ）时，SCIF 内部的波特率发生器以倍速模式运行。

可通过更改 ABCS 位的设定，选择在异步模式的 1 位期间的基本时钟。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	BGDM	-	-	-	-	-	-	ABCS
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R/W

位	位名称	初始值	R/W	说 明
15 ~ 8	—	均为 0	R	保留位 读取值、写入值总是为 0。
7	BGDM	0	RW	波特率发生器倍速模式 BGDM 位置 1 时，SCIF 内部的波特率发生器以倍速模式运行。仅在异步模式（SCSMR 的 $\overline{C/A}=0$ ）且将内部时钟作为时钟源、SCK 引脚用作输入引脚（SCSCR 的 $CKE[1:0]=00$ ）时，BGDM 的设定才有效。除上述之外时的设定，必须使用普通模式。 0: 普通模式 1: 倍速模式
6 ~ 1	—	均为 0	R	保留位 读取值、写入值总是为 0。
0	ABCS	0	RW	选择异步基本时钟 选择异步模式时 1 位期间的基本时钟。仅在异步模式（SCSMR 的 $\overline{C/A}=0$ ）时，ABCS 位的设定才有效。 0: 以频率为 16 倍位速率的基本时钟运行 1: 以频率为 8 倍位速率的基本时钟运行

15.4 运行说明

15.4.1 概要

SCIF 可使用以字符为单位一边同步一边通信的异步模式和通过时钟脉冲一边同步一边通信的时钟同步模式这两种方式执行串行通信。

在接收部和发送部分别内置 16 段 FIFO 缓冲器，因此可减少 CPU 的开销并可高速连续通信。而且，通道 3 还具有用作调制解调器控制信号的 $\overline{\text{RTS}}$ 和 $\overline{\text{CTS}}$ 信号。由串行模式寄存器（SCSMR）选择发送 / 接收格式，如表 15.9 所示。由串行控制寄存器（SCSCR）的 CKE[1:0] 的组合决定 SCIF 时钟源，如表 15.10 所示。

(1) 异步模式

- 数据长度：可选择 7 位或 8 位
- 可选择附加奇偶校验位及 1 位 / 2 位的停止位（由这些组合决定发送 / 接收格式及字符长度）
- 接收时，可检测帧错误、奇偶校验错误、接收 FIFO 数据满、溢出错误、接收数据就绪及断点
- 表示各发送 / 接收 FIFO 寄存器保存的数据个数
- SCIF 的时钟源：可选择内部时钟或外部时钟
选择内部时钟时：SCIF 以波特率发生器的时钟运行
选择外部时钟时：需输入频率为 16/8 倍位速率的时钟
(不使用内部波特率发生器)

(2) 时钟同步模式

- 发送 / 接收格式：固定为 8 位数据
- 接收时，可检测溢出错误
- SCIF 的时钟源：可选择内部时钟或外部时钟
选择内部时钟时：SCIF 以波特率发生器的时钟运行，且将此时钟作为同步时钟向外部输出
选择外部时钟时：不使用内部波特率发生器，而以输入的外部同步时钟运行

表 15.9 SCSMR 的设定值和 SCIF 发送 / 接收格式

SCSMR				模式	SCIF 发送 / 接收格式		
bit7	bit6	bit5	bit4		数据长度	奇偶校验位	停止位长
$\text{C}/\overline{\text{A}}$	CHR	PE	STOP				
0	0	0	0	异步模式	8 位	无	1 位
			1				2 位
		1	0			有	1 位
			1				2 位
	1	0	0		7 位	无	1 位
			1				2 位
		1	0			有	1 位
			1				2 位
1	x	x	x	时钟同步模式	8 位	无	无

【符号说明】 x: Don't care

表 15.10 SCSMR、SCSCR 的设定值和 SCIF 时钟源的选择

SCSMR	SCSCR	模式	时钟源	SCK 引脚的功能
bit7	bit1、0			
$\overline{C/A}$	CKE[1:0]			
0	00	异步模式	内部	SCIF 不使用 SCK 引脚
	01			输出频率为 16/8 倍位速率的时钟
	10		外部	输入频率为 16/8 倍位速率的时钟
	11			禁止设定
1	0x	时钟同步模式	内部	输出同步时钟
	10		外部	输入同步时钟
	11		禁止设定	

【符号说明】x: Don't care

【注】使用波特率发生器倍速模式 (BGDM=1) 时, 必须设定异步模式 ($\overline{C/A}=0$) 并且内部时钟作为时钟源 / 不使用 SCK 引脚 (CKE[1:0]=00)。

15.4.2 异步模式时的运行

异步模式是以字符为单位一边同步一边执行串行通信的模式, 用来发送 / 接收将表示通信开始的开始位和表示通信结束的停止位附加到数据的字符。

SCIF 内部的发送部和接收部是独立的, 因此, 可执行全双工通信。发送部和接收部均为 16 段 FIFO 缓冲结构, 在发送 / 接收过程中可读取 / 写入数据, 因此可执行连续发送 / 接收。

异步模式串行通信的一般格式如图 15.2 所示。

异步串行通信时, 通信线路通常保持为标记状态 (高电平)。SCIF 监控通信线路, 将变为空闲 (低电平) 的位视为开始位, 开始串行通信。

按照从开始位 (低电平) 开始, 数据 (LSB first: 从最低位开始)、奇偶校验位 (高 / 低电平), 最后为停止位 (高电平) 的顺序, 构成串行通信的 1 个字符。

在异步模式, SCIF 在接收时在开始位的下降沿执行同步。另外, SCIF 在 1 位期间频率为 16/8 的时钟的第 8/4 个时钟采样数据。因此, 在各位的中央输入通信数据。

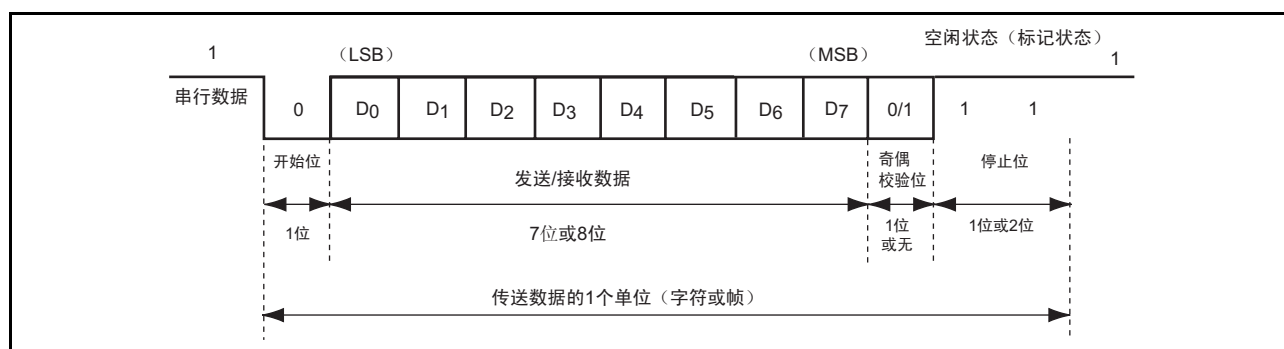


图 15.2 异步通信的数据格式
(8 位数据 / 有奇偶校验位 / 2 个停止位的例子)

(1) 发送 / 接收格式

可设定的发送 / 接收格式如表 15.11 所示。
发送 / 接收格式有 8 种，可通过设定串行模式寄存器（SCSMR）来选择。

表 15.11 串行发送 / 接收格式（异步模式）

SCSMR 的设定			串行发送/接收格式和帧长度											
CHR	PE	STOP	1	2	3	4	5	6	7	8	9	10	11	12
0	0	0	START	8位数据							STOP			
		1	START	8位数据							STOP	STOP		
	1	0	START	8位数据							P	STOP		
		1	START	8位数据							P	STOP	STOP	
1	0	0	START	7位数据						STOP				
		1	START	7位数据						STOP	STOP			
	1	0	START	7位数据						P	STOP			
		1	START	7位数据						P	STOP	STOP		

[符号说明]
START : 开始位
STOP : 停止位
P : 奇偶校验位

(2) 时钟

通过设定 SCSCR 的 $\overline{C/A}$ 位及串行控制寄存器 (SCSCR) 的 CKE[1:0] 位, 可从内部波特率发生器生成的内部时钟或从 SCK 引脚输入的外部时钟中, 选择 SCIF 的发送 / 接收时钟。有关 SCIF 时钟源的选择, 详情参照表 15.10。

从 SCK 引脚输入外部时钟时, 必须输入频率为 16/8 倍位速率的时钟。

通过内部时钟运行时, 可从 SCK 引脚输出时钟。输出时钟频率为位速率的 16/8 倍。

(3) 数据的发送 / 接收运行

• SCIF 初始化 (异步模式)

发送/接收数据前, 在串行控制寄存器 (SCSCR) 的 TE 位及 RE 位清 0 后, 必须按以下顺序初始化 SCIF。

更改运行模式或通信格式时, 必须将 TE 位和 RE 位清 0 后, 按照以下步骤执行。如果 TE 位清 0, 则初始化发送移位寄存器 (SCTSR)。必须注意: 即使 TE 位和 RE 位清 0, 串行状态寄存器 (SCFSR)、发送 FIFO 数据寄存器 (SCFTDR)、接收 FIFO 数据寄存器 (SCFRDR) 也不被初始化, 仍保持以前的内容。必须在发送所有的发送数据, 且 SCFSR 的 TEND 标志置位后, 再将 TE 位清 0。TE 位即使在发送过程中也可清 0, 但清 0 后, 发送数据为标记状态。在重新将 TE 位置 1 后开始发送前, 必须将 SCFCR 的位 TFRST 置 1 再复位 SCFTDR。

使用外部时钟时, 在包含初始化的运行过程中, 不可停止时钟。否则, 运行将不稳定。SCIF 的初始化流程图的例子如图 15.3 所示。

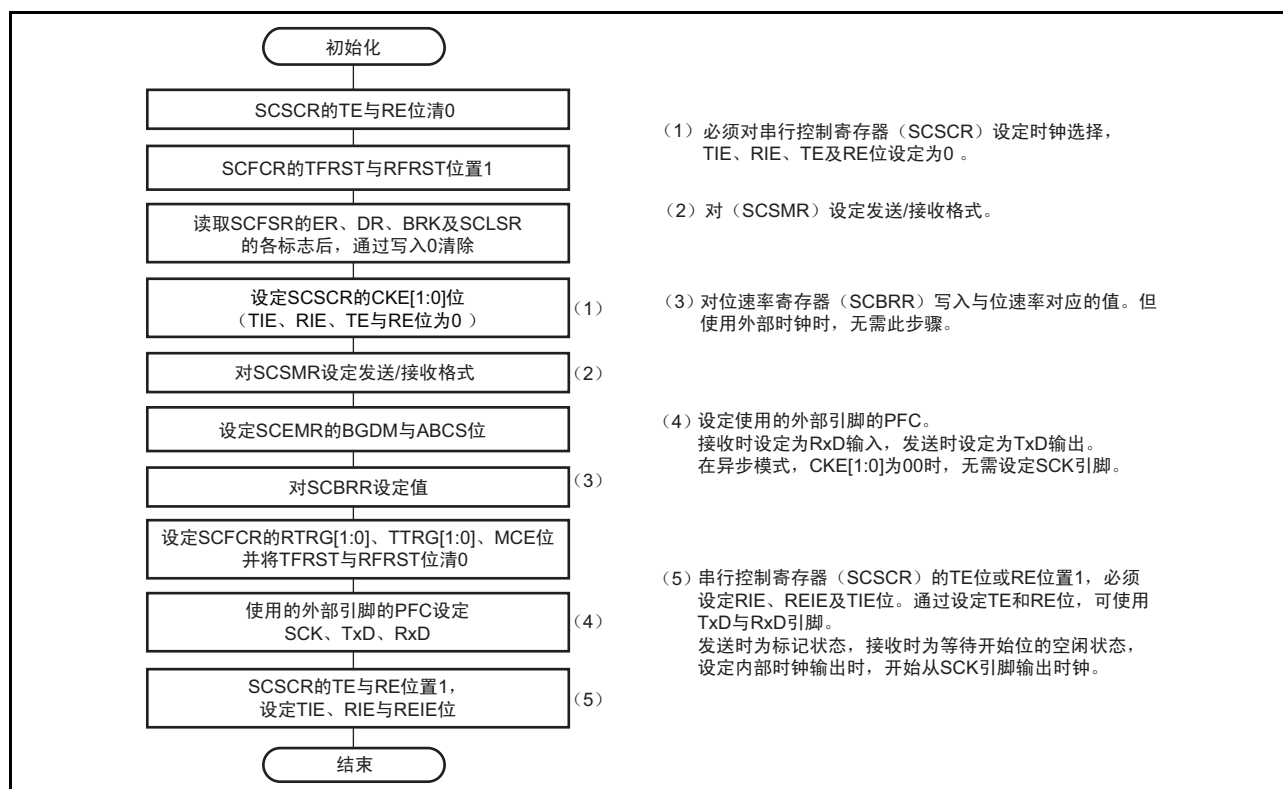


图 15.3 SCIF 初始化流程图的例子

- 发送串行数据 (异步模式)

串行发送的流程图的例子如图 15.4 所示。

SCIF 设定为可发送状态后, 必须按照以下步骤发送串行数据。

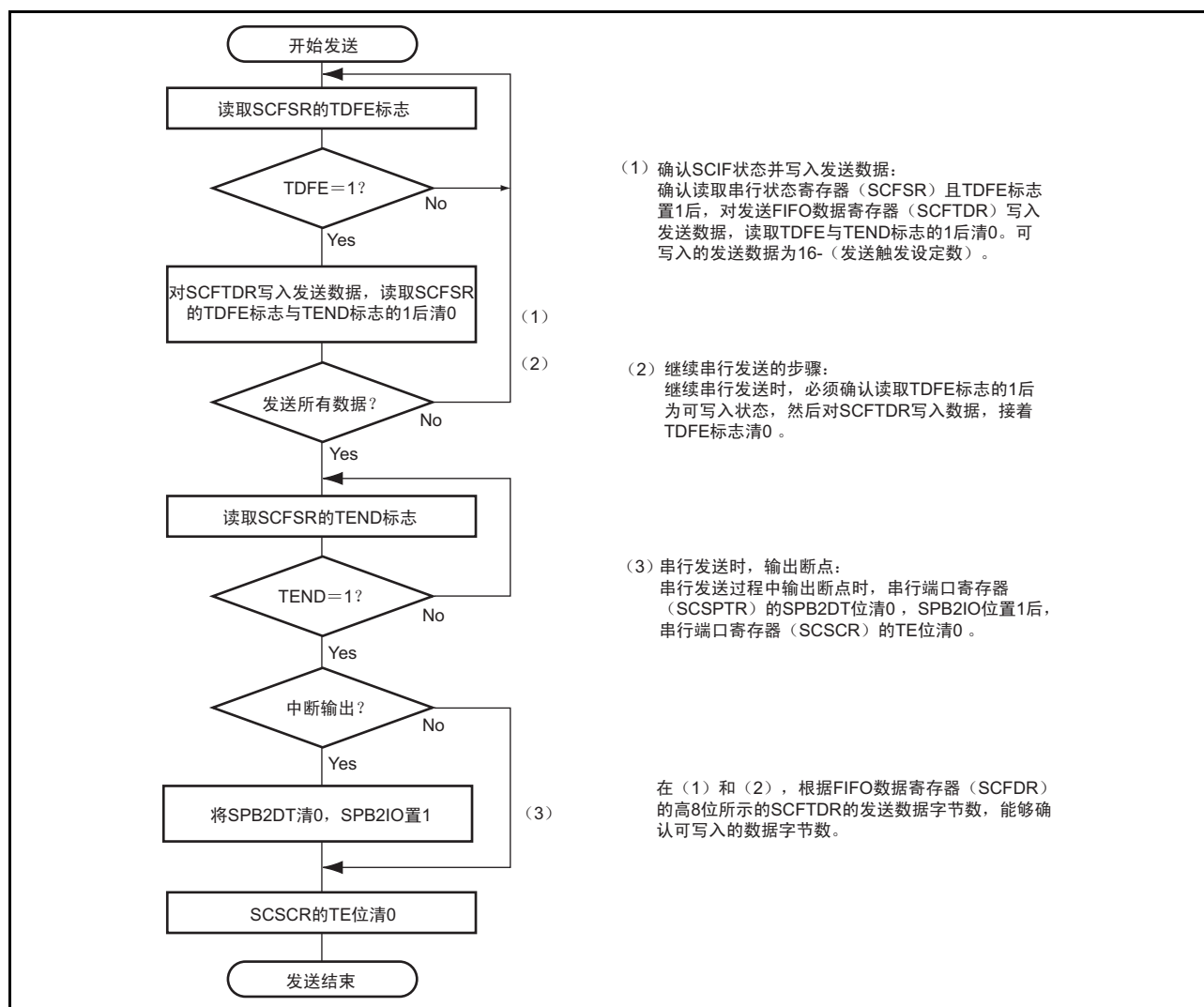


图 15.4 串行发送的流程图的例子

串行发送时，SCIF 如下运行：

1. 数据写入发送FIFO数据寄存器（SCFTDR）时，SCIF将数据从SCFTDR传送到发送移位寄存器（SCTSR）。将发送数据写入SCFTDR前，必须确认串行状态寄存器（SCFSR）的TDFE标志置1。可写入的数据字节数是（16 – 发送触发设定数）。
2. 如果数据从SCFTDR传送到SCTSR并开始发送，则执行连续发送，直到SCFTDR无发送数据。如果SCFTDR的发送数据字节数小于等于FIFO控制寄存器（SCFCR）设定的发送触发个数，则TDFE标志置位。此时，如果串行控制寄存器（SCSCR）的TIE位置1，则产生发送FIFO数据空中断（TXI）请求。

按照以下顺序，从TxD引脚发送串行发送数据。

- (a) 开始位：输出1位0。
 - (b) 发送数据：从LSB开始，按顺序输出8位或7位数据。
 - (c) 奇偶校验位：输出1位奇偶校验位（偶校验或者奇校验）
（另外，也可选择不输出奇偶校验位的格式）。
 - (d) 停止位：输出1位或2位1（停止位）。
 - (e) 标记状态：在发送开始下一发送的开始位前，连续输出1
3. SCIF在发送停止位的时序，检查SCFTDR的发送数据。如果有数据，则将数据从SCFTDR传送到SCTSR，并在发送停止位后，开始下一帧的串行发送。

异步模式的发送时的运行例如图 15.5 所示。

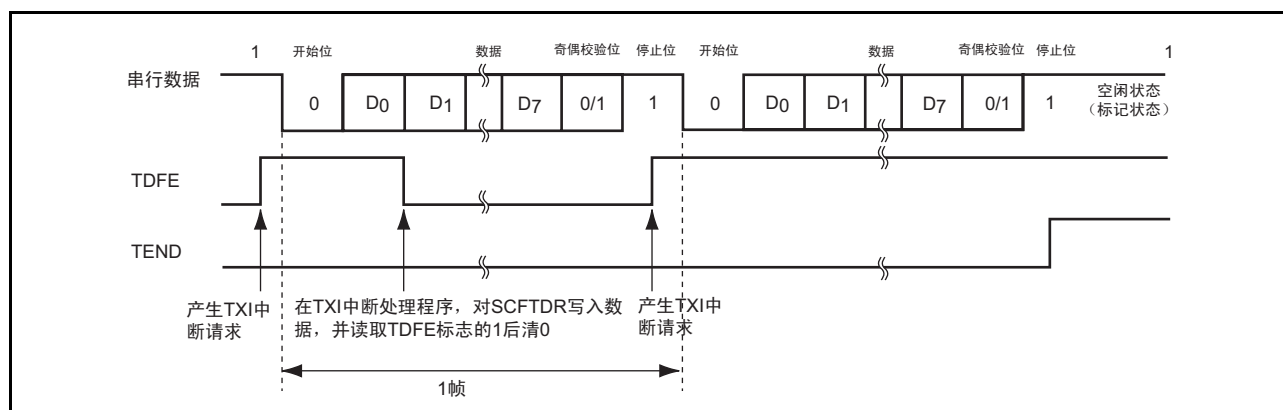


图 15.5 发送时的运行例（8 位数据 / 有奇偶校验位 / 1 个停止位）

4. 在通道3允许调制解调器控制时,可根据 $\overline{\text{CTS}}$ 输入值停止/重新发送。 $\overline{\text{CTS}}$ 置1时,如果正在发送,则1帧发送结束后进入标记状态。如果 $\overline{\text{CTS}}$ 置0,则从开始位开始输出下一个发送数据。

使用调制解调器控制时的运行例如图 15.6 所示。

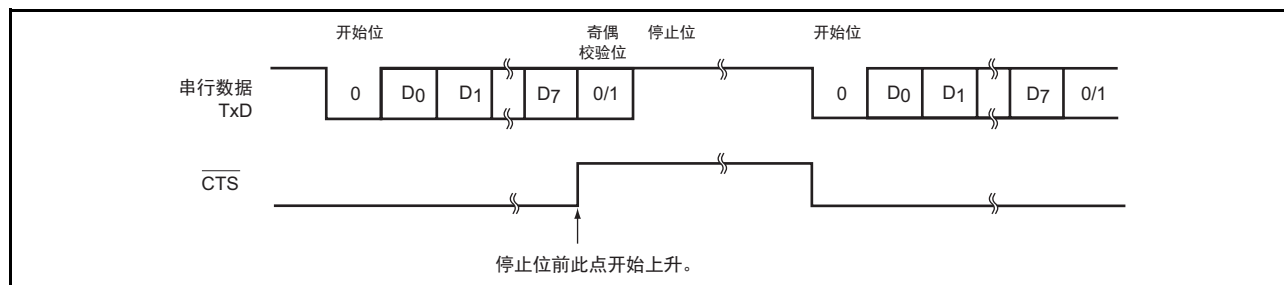


图 15.6 使用调制解调器控制时的运行例 ($\overline{\text{CTS}}$)

- 接收串行数据 (异步模式)

串行接收的流程图的例子如图 15.7 和图 15.8 所示。

SCIF 设定为可接收状态后,必须按照以下步骤接收串行数据。

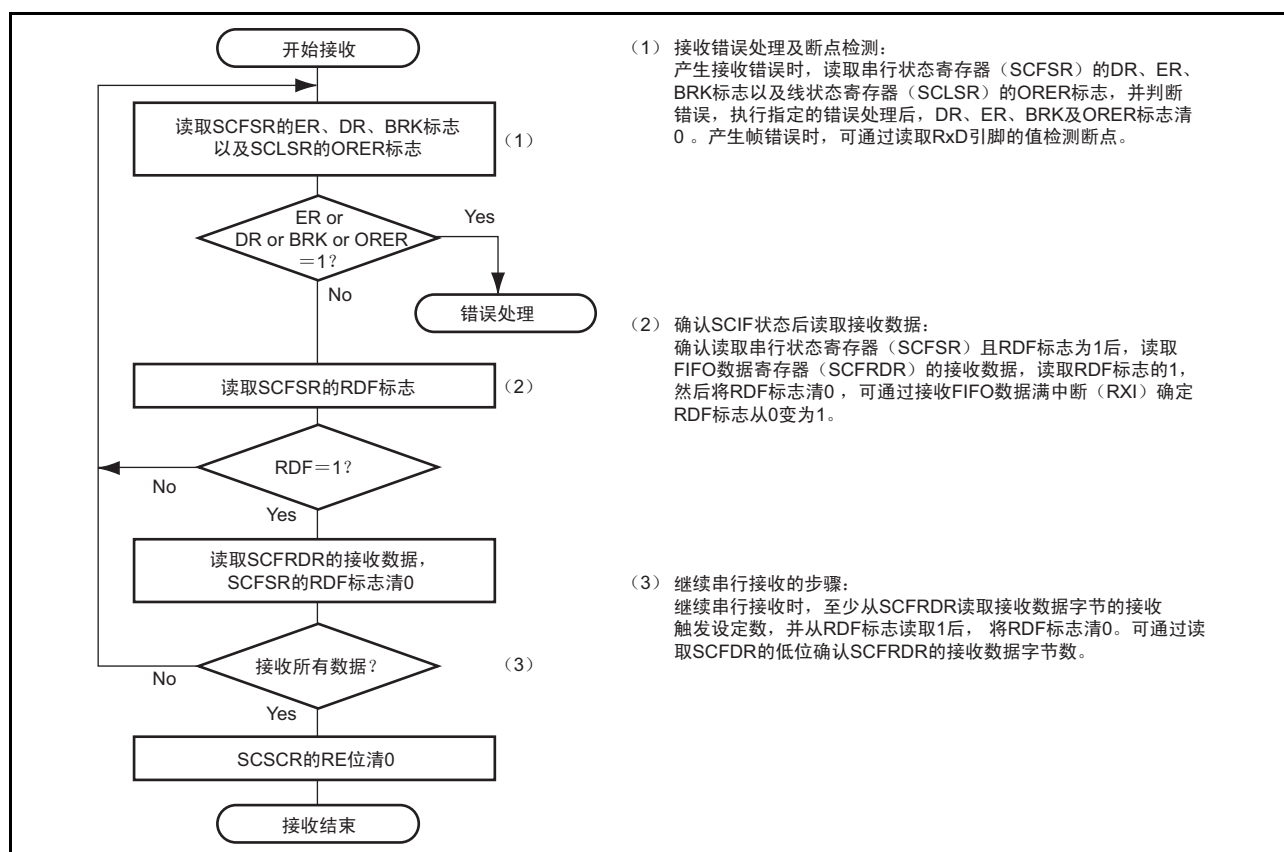


图 15.7 串行接收的流程图的例子 (1)

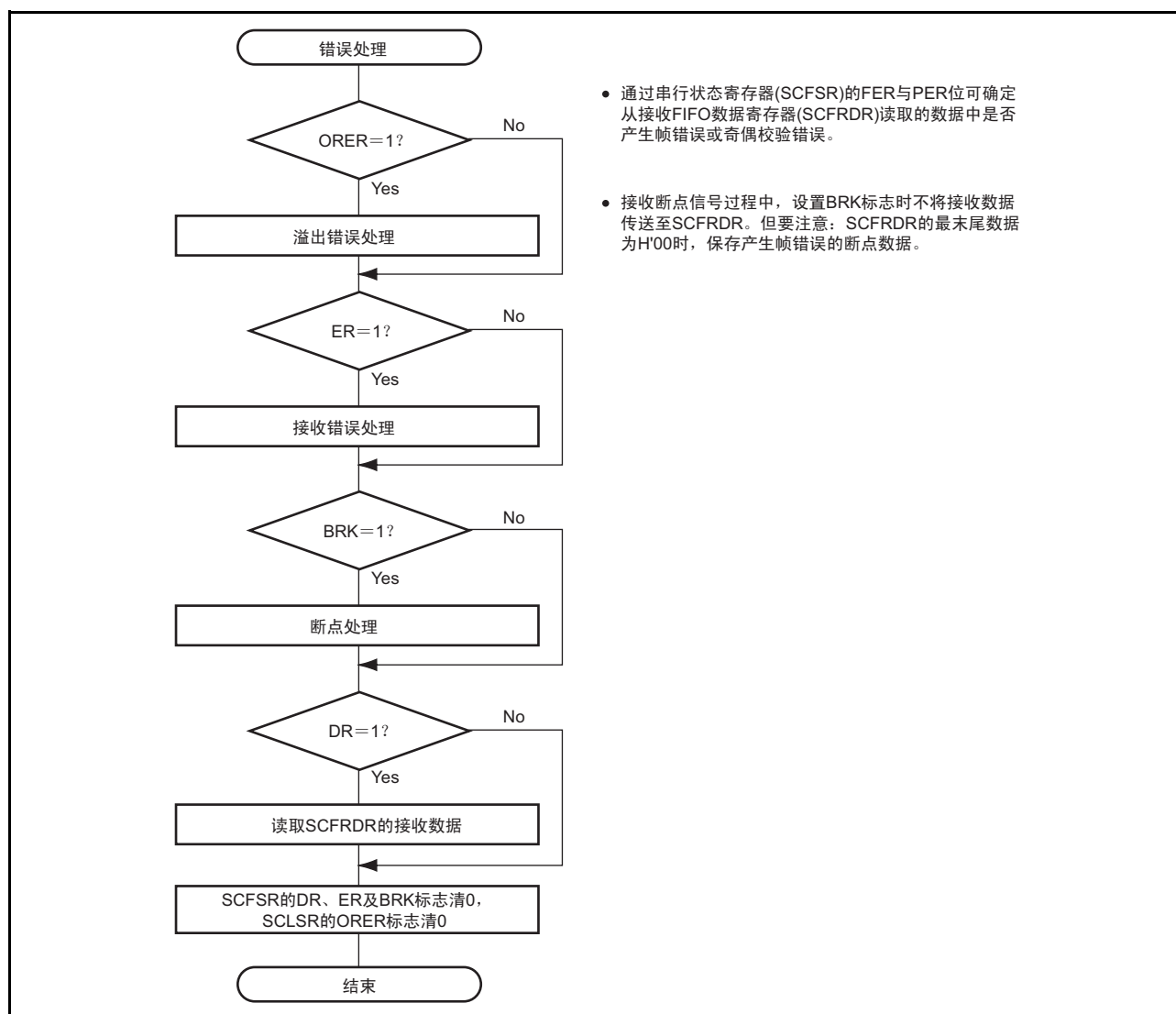


图 15.8 串行接收的流程图的例子 (2)

接收时, SCIF 如下运行:

1. SCIF 监控通信线路, 如果检测出开始位 0, 则内部同步且开始接收。
2. 按照从 SCRSR 的 LSB 到 MSB 的顺序, 保存已接收数据。
3. 接收奇偶校验位和停止位。

接收后, SCIF 执行以下检查:

- (a) 检查停止位: 检查停止位是否为 1。有 2 个停止位时, 仅检查第 1 个停止位。
- (b) 检查是否为可将接收数据从接收移位寄存器 (SCRSR) 传送到 SCFRDR 的状态。
- (c) 检查溢出: 检查表示未产生溢出错误的 ORER 标志是否为 0。
- (d) 检查断点: 检查表示未设置断点状态的 BRK 标志是否为 0。

以上检查全部通过后, 将接收数据保存到 SCFRDR。

【注】 即使产生奇偶校验错误或帧错误, 仍继续接收。

4. RDF或DR标志变为1时，如果SCSCR的RIE位置1，则产生接收FIFO数据满中断（RXI）请求。ER标志变为1时，如果SCSCR的RIE位或REIE位置1，则产生接收错误中断（ERI）请求；BRK标志或ORER标志变为1时，如果SCSCR的RIE位或REIE位置1，则产生断点接收中断（BRI）请求。

异步模式接收时的运行例如图 15.9 所示。

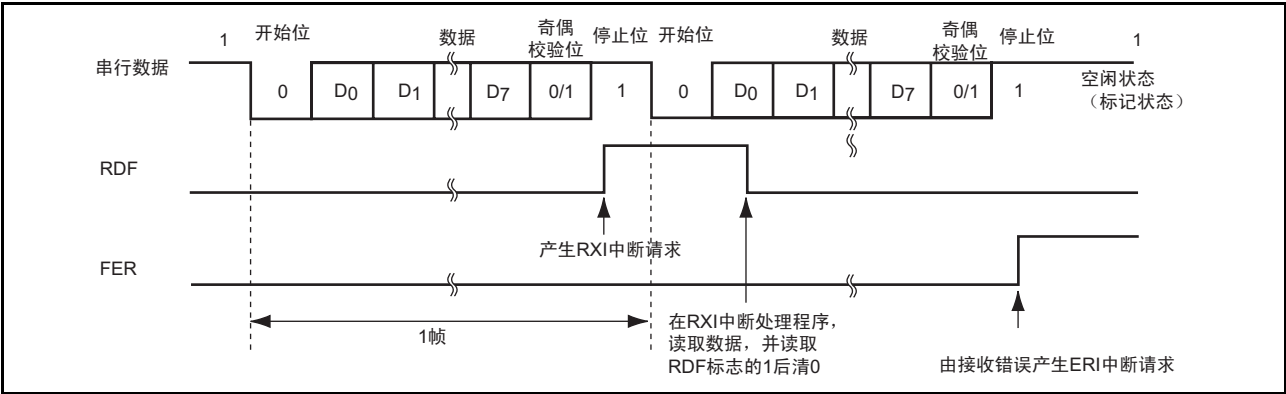


图 15.9 SCIF 接收时的运行例（8 位数据 / 有奇偶校验位 / 1 个停止位）

5. 如果通道3的调制解调器控制有效，则SCFRDR为空时输出 $\overline{\text{RTS}}$ 信号。 $\overline{\text{RTS}}$ 为0时，可接收。 $\overline{\text{RTS}}$ 为1时，表示SCFRDR的数据个数大于等于RTS输出有效触发的设定个数。

使用调制解调器控制时的运行例如图 15.10 所示。

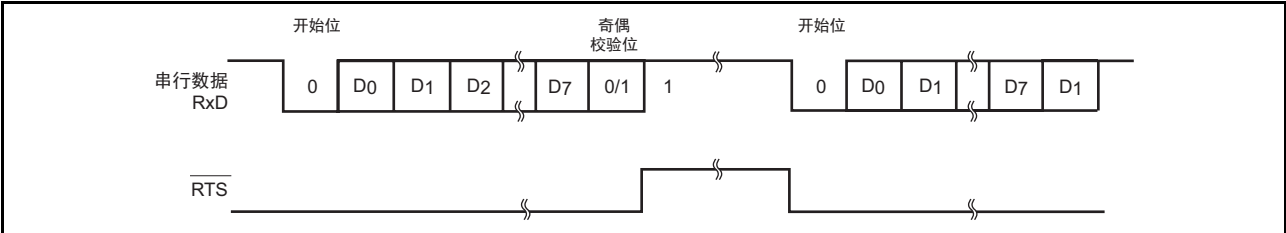


图 15.10 调制解调器控制时的运行例（ $\overline{\text{RTS}}$ ）

15.4.3 时钟同步模式时的运行

时钟同步模式是与时钟脉冲同步执行数据发送 / 接收的模式，适用于高速串行通信。

SCIF 内部的发送部和接收部独立，可通过共用时钟执行全双工通信。发送部和接收部均为 16 段 FIFO 缓冲结构，发送 / 接收过程中可读取 / 写入数据，也可执行连续发送 / 接收。

时钟同步串行通信的一般格式如图 15.11 所示。

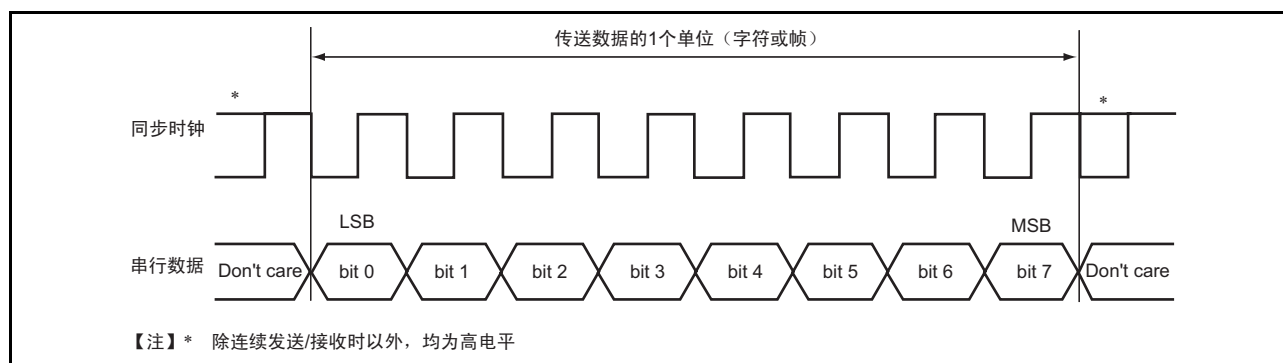


图 15.11 时钟同步通信的数据格式

时钟同步串行通信时，从同步时钟的下降沿到下一个下降沿输出通信线路的数据。在同步时钟的上升沿，保证数据有效。

按照从数据 LSB 到 MSB 的顺序输出串行通信的 1 个字符。输出 MSB 后的通信线路保持 MSB 的状态。在时钟同步模式，SCIF 与串行时钟的上升沿同步接收数据。

(1) 发送 / 接收格式

固定为 8 位数据。
不可附加奇偶校验位。

(2) 时钟

通过设定 SCSMR 的 $\overline{C/A}$ 位和 SCSCR 的 CKE[1:0] 位，选择内部波特率发生器生成的内部时钟或从 SCK 引脚输入的外部时钟。

以内部时钟运行时，从 SCK 引脚输出同步时钟。发送和接收 1 个字符时同步时钟输出 8 个脉冲。无发送 / 接收时，固定为高电平。仅接收运行时，如果选择内部时钟，则在 SCSCR 的 RE 位为 1 期间，输出时钟脉冲，直至接收 FIFO 内的数据个数达到接收触发设定的个数。

(3) 数据发送 / 接收运行

- SCIF 的初始化（时钟同步模式）
发送 / 接收数据之前，必须将串行控制寄存器（SCSCR）的 TE 和 RE 位清 0，再按以下顺序初始化 SCIF。
更改模式或通信格式时，必须将 TE 和 RE 位清 0，再按以下顺序更改。如果 TE 位清 0，则初始化发送移位寄存器（SCTSR）。必须注意：即使 RE 位清 0，也将保持 RDF、PER、FER、ORER 各标志及接收数据寄存器（SCRDR）的内容。

SCIF 的初始化流程图的例子如图 15.12 所示。

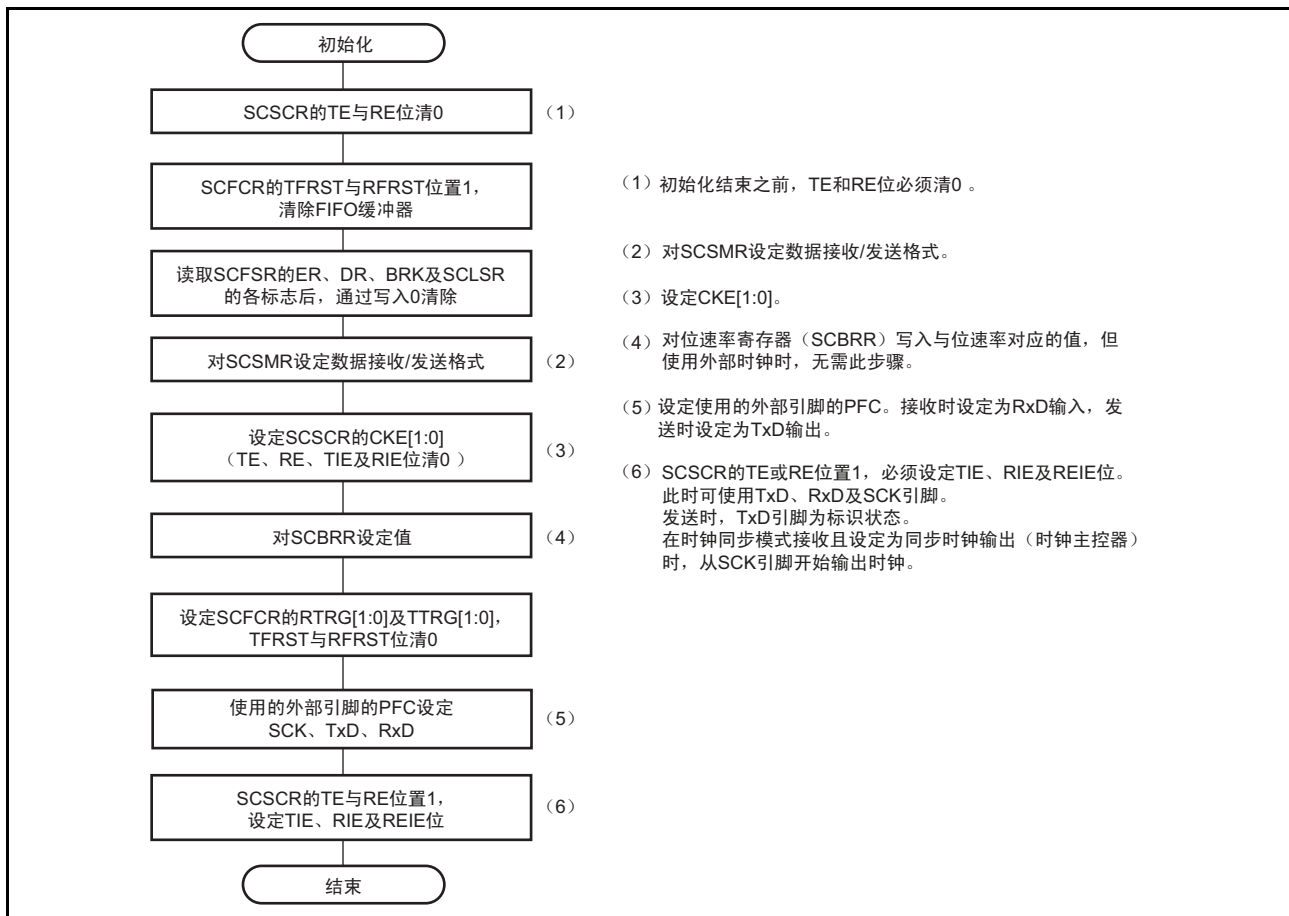


图 15.12 SCIF 的初始化流程图的例子

- 发送串行数据 (时钟同步模式)

串行发送的流程图的例子如图 15.13 所示。

SCIF 设定为可发送状态后, 必须按照以下步骤发送串行数据。

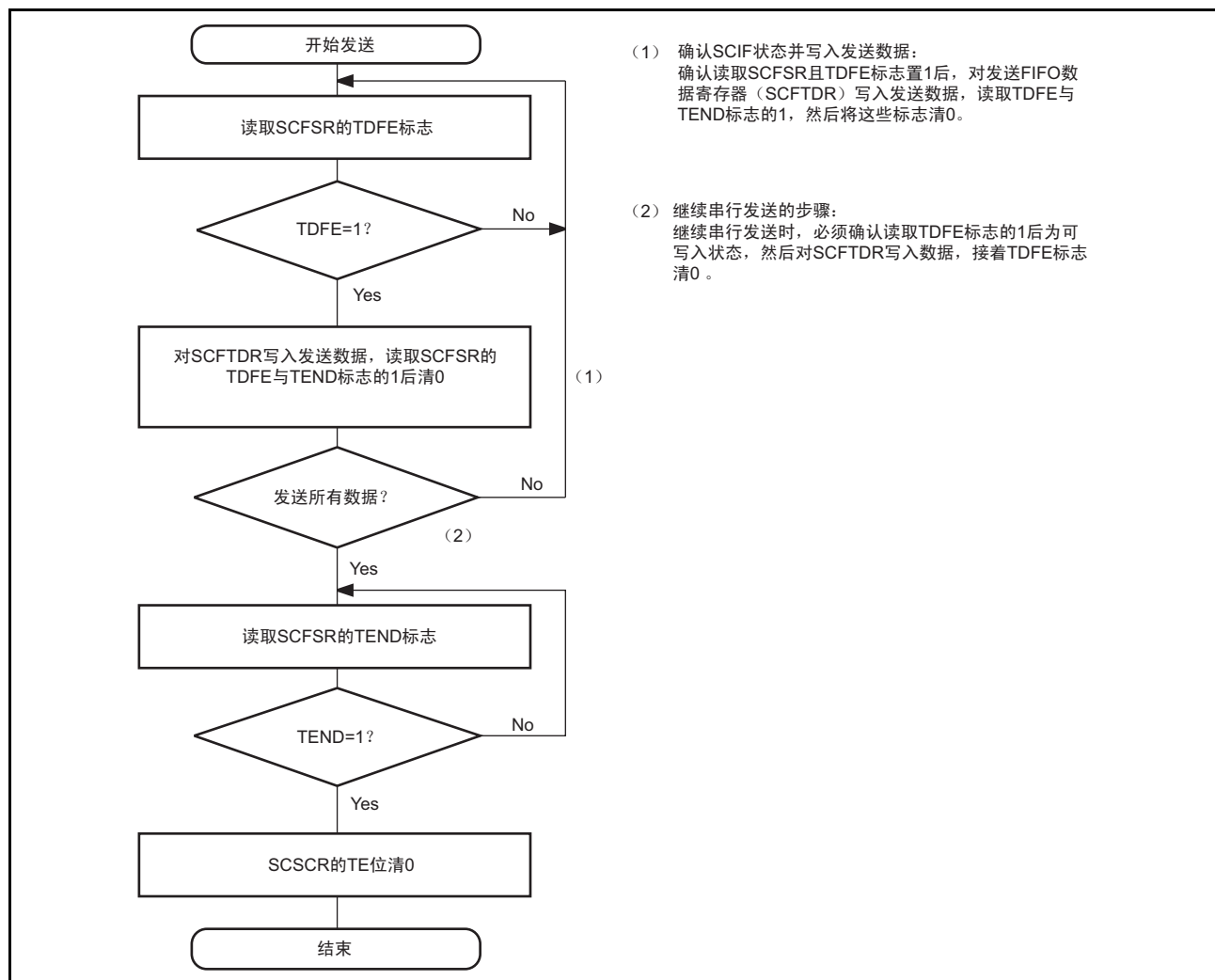


图 15.13 串行发送的流程图的例子

串行发送时，SCIF 如下运行：

1. 数据写入发送FIFO数据寄存器（SCFTDR）时，SCIF将数据从SCFTDR传送到发送移位寄存器（SCTSR）。发送数据写入SCFTDR前，必须确认串行状态寄存器（SCFSR）的TDFE标志已置1。可写入的数据字节数为（16-发送触发设定数）。
2. 如果数据从SCFTDR传送到SCTSR并开始发送，则执行连续发送，直到SCFTDR中无发送数据。如果SCFTDR的发送数据字节数小于等于FIFO控制寄存器（SCFCR）设定的发送触发个数时，则置位TDFE标志。此时，如果串行控制寄存器（SCSCR）的TIE位置1，则产生发送FIFO数据空中断（TXI）请求。

设定为时钟输出模式时，SCIF输出8个脉冲的同步时钟。设定为外部时钟时，与输入时钟同步输出数据。按照LSB（bit0）～MSB（bit7）的顺序，从TxD引脚输出串行发送数据。

3. 在发送最末位的时序，SCIF检查SCFTDR的发送数据。如果有发送数据，则将数据从SCFTDR传送到SCTSR，并开始下一帧的串行发送；如果无数据，则将SCFSR的TEND标志置1并输出最末位后，TxD引脚保持状态。
4. 串行发送结束后，SCK引脚固定为高电平。

SCIF发送时的运行例如图 15.14 所示。

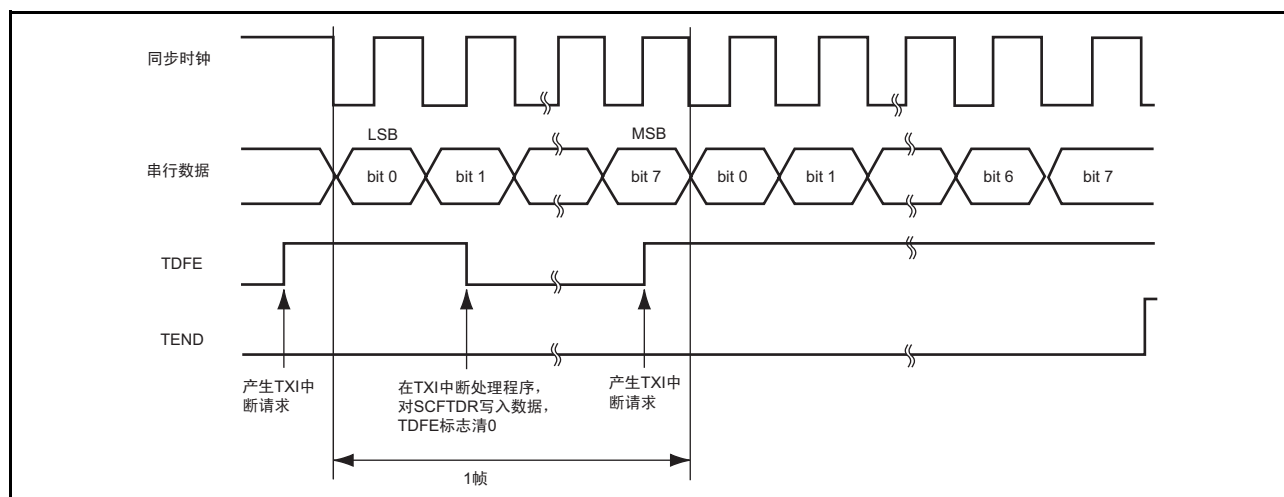


图 15.14 SCIF 发送时的运行例

- 接收串行数据（时钟同步模式）

串行接收的流程图的例子，如图 15.15 和图 15.16 所示。

SCIF 设定为可接收状态后，必须按照以下步骤接收串行数据。

不执行 SCIF 初始化，并将运行模式从异步模式转换到时钟同步模式时，必须确认已将 ORER、PER、FER 各标志清 0。

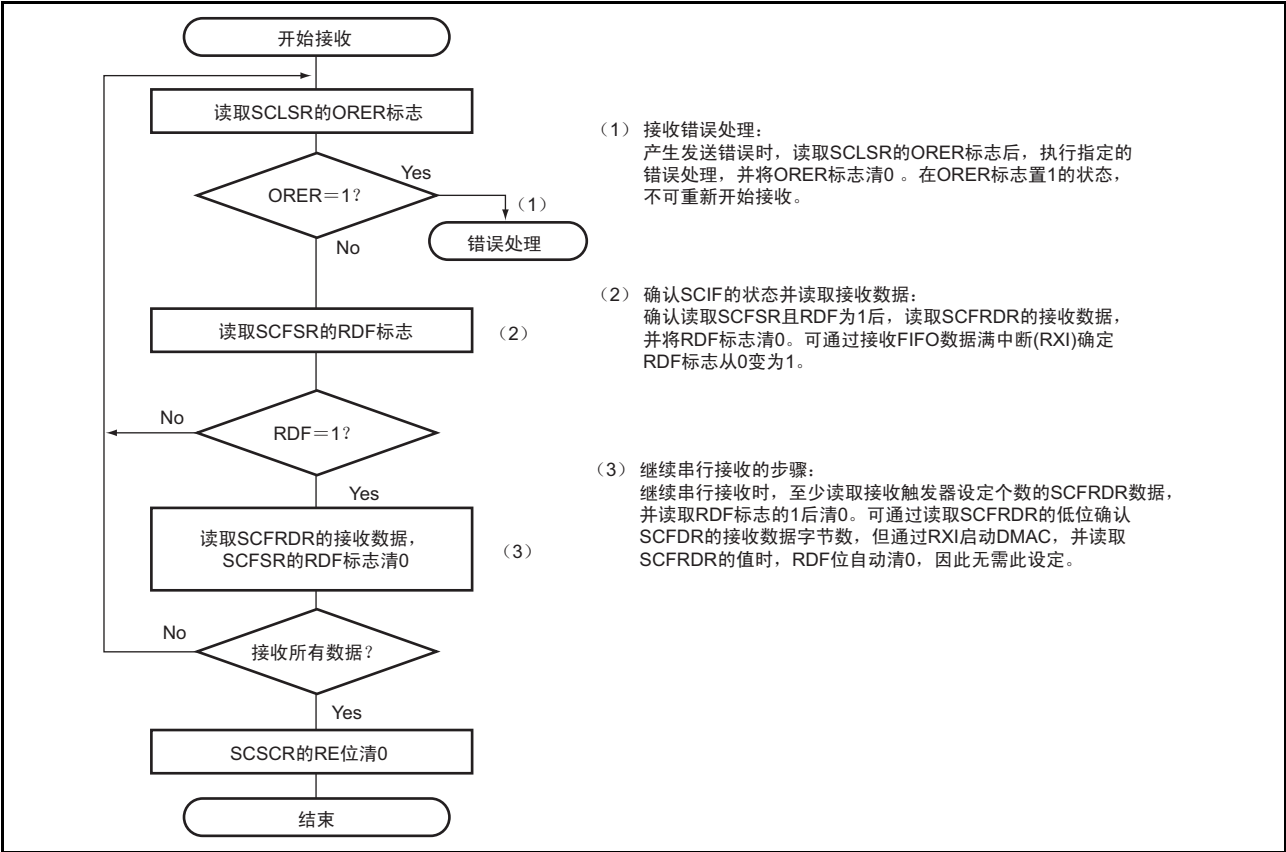


图 15.15 串行接收的流程图的例子（1）

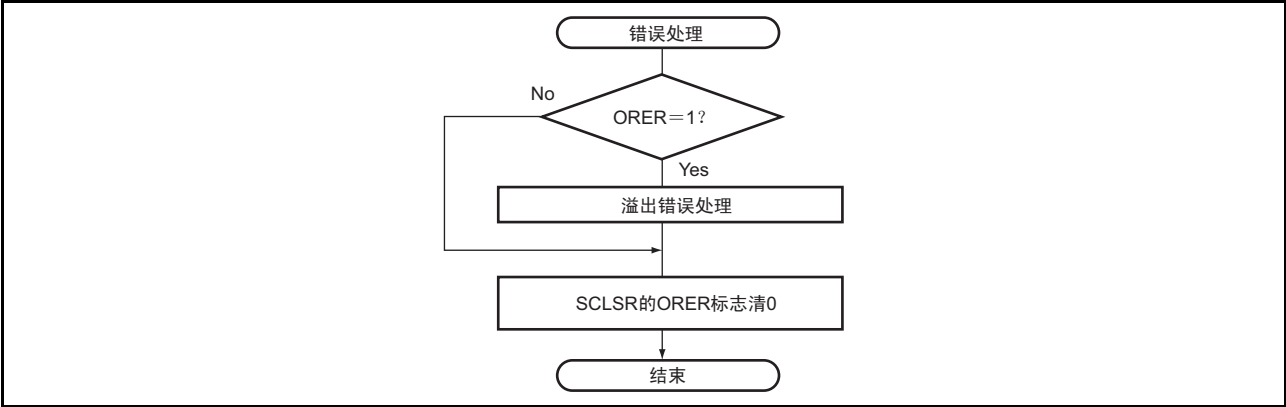


图 15.16 串行接收的流程图的例子（2）

串行接收时，SCIF 如下运行：

1. SCIF 与同步时钟的输入或输出同步后开始接收。
2. 按照从 LSB 到 MSB 的顺序，将接收数据保存到接收移位寄存器 (SCRSR)。接收后，SCIF 检查是否为可将接收数据从 SCRSR 传送到 SCFRDR 的状态。如果检查通过，则 RDF 标志置 1，并将接收数据保存到 SCFRDR。如果在错误检查时检测出溢出错误，则不能执行之后的接收。
3. RDF 标志为 1 时，如果串行控制寄存器 (SCSCR) 的 RIE 位置 1，则产生接收 FIFO 数据满中断 (RXI) 请求；ORER 标志为 1 时，如果 SCSCR 的 RIE 位或 REIE 置 1，则产生断点中断 (BRI) 请求。

SCIF 接收时的运行例如图 15.17 所示。

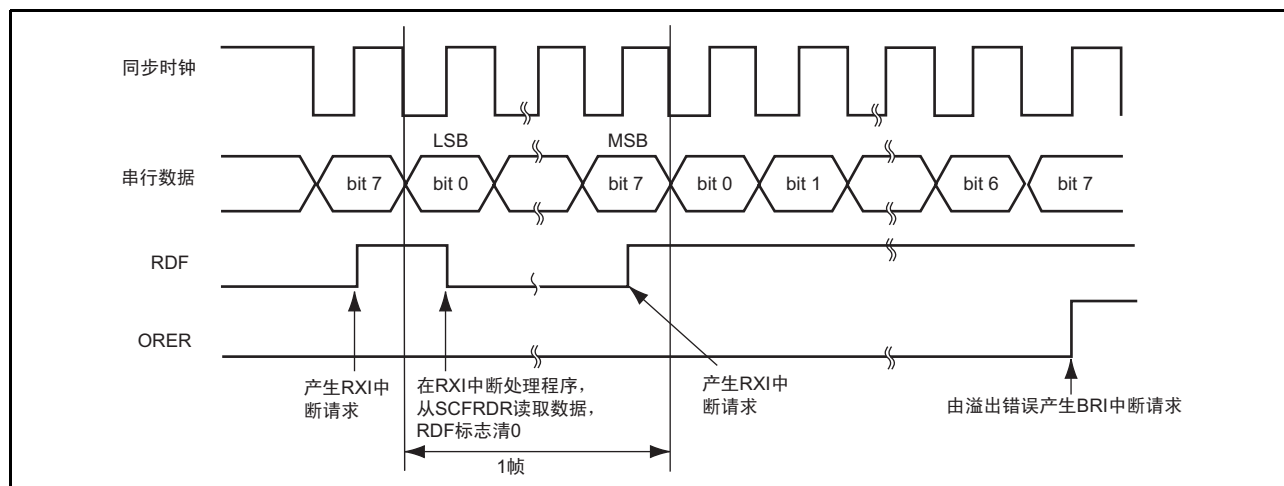


图 15.17 SCIF 接收时的运行例

- 串行数据发送/接收同时运行 (时钟同步模式)

串行发送/接收同时运行的流程图的例子如图 15.18 所示。

SCIF 设定为可发送/接收状态后, 必须按照以下步骤同时发送/接收串行数据。

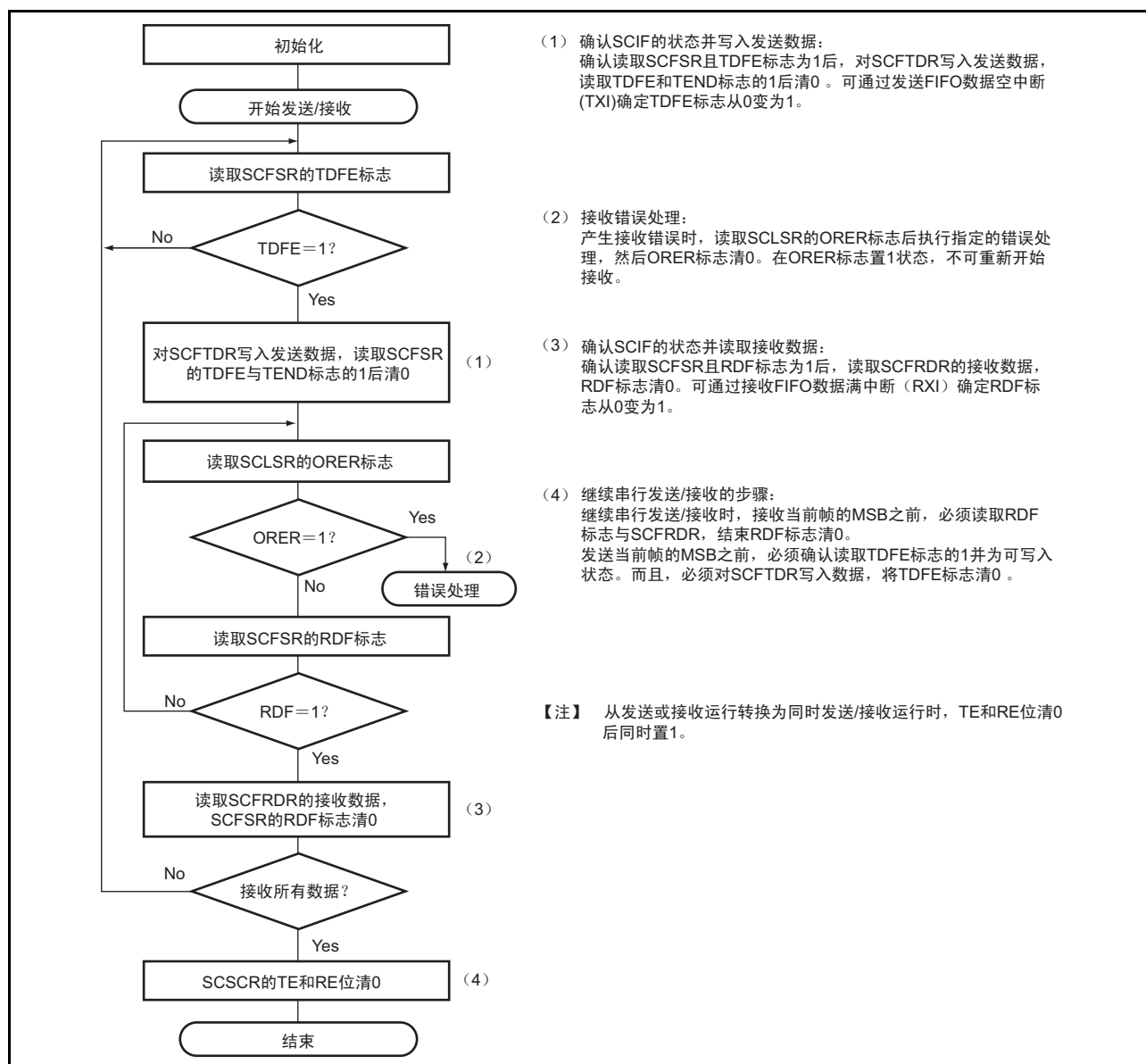


图 15.18 串行数据发送 / 接收的流程图例

15.5 SCIF 的中断

SCIF 有共 4 种中断源，分别为发送 FIFO 数据空中断（TXI）请求、接收错误中断（ERI）请求、接收 FIFO 数据满中断（RXI）请求和断点中断（BRI）请求。

各中断源和优先顺序如表 15.12 所示。由 SCSCR 的 TIE 位、RIE 位、REIE 位，允许或者禁止各中断源。另外，可分别将各中断请求单独传送到中断控制器。

由 TIE 位允许 TXI 时，如果串行状态寄存器（SCFSR）的 TDFE 标志置 1，则产生 TXI 中断请求。由此 TXI 中断请求可启动 DMAC，执行数据传送。此时，不产生对 CPU 的中断请求。

由 RIE 位允许 RXI 时，如果 SCFSR 的 RDF 标志或 DR 标志置 1，则产生 RXI 中断请求。由此 RXI 中断请求可启动 DMAC，执行数据传送。此时，不产生对 CPU 的中断请求。仅在异步模式时，通过 DR 标志置 1 产生 RXI 中断请求。

如果 SCSCR 的 RIE 位设定为 0 并且 REIE 位设定为 1，则可不产生 RXI 中断请求而仅产生 ERI 中断请求。

TXI 表示可写入发送数据，RXI 表示 SCFRDR 中有接收数据。

表 15.12 SCIF 中断源

中断源	内 容	启动 DMAC	解除复位时 优先顺序
BRI	断点（BRK）或溢出（ORER）引起的中断	不可	高 ↑ ↓ 低
ERI	接收错误（ER）引起的中断	不可	
RXI	接收 FIFO 数据满（RDF）或者数据就绪（DR）引起的中断	可	
TXI	发送 FIFO 数据空（TDFE）引起的中断	可	

15.6 使用时的注意事项

使用 SCIF 时，必须注意以下事项。

15.6.1 向 SCFTDR 的写入和 TDFE 标志

如果写入发送 FIFO 数据寄存器（SCFTDR）的发送数据字节数小于 FIFO 控制寄存器（SCFCR）的 TTRG[1:0] 设定的发送触发个数时，则置位串行状态寄存器（SCFSR）的 TDFE 标志。TDFE 标志置位后，可在达到 SCFTDR 的空字节数之前写入发送数据，实现高效率的连续发送。

但写入 SCFTDR 的数据字节数小于等于发送触发个数时，即使读取 1 后清 0，TDFE 标志也将再次置 1。清除 TDFE 标志时，必须在写入 SCFTDR 的数据字节数超过发送触发个数时读取 1 再执行。

可通过 FIFO 数据计数寄存器（SCFDR）的高 8 位，获取 SCFTDR 的发送数据字节数。

15.6.2 SCFRDR 的读取和 RDF 标志

如果接收 FIFO 数据寄存器（SCFRDR）的接收数据字节数大于等于 FIFO 控制寄存器（SCFCR）的 RTRG[1:0] 设定的接收触发个数，则置位串行状态寄存器（SCFSR）的 RDF 标志。RDF 标志置位后，可从 SCFRDR 读取触发个数的接收数据，实现高效率的连续接收。

但 SCFRDR 的数据字节数超过接收触发个数时，即使读取 1 后清 0，RDF 标志也将再次置 1。清除 RDF 标志时，必须读取接收数据，使接收 FIFO 数据寄存器（SCFRDR）内的数据个数少于触发个数，再读取 RDF 标志的 1 并清 0。可通过 FIFO 数据计数寄存器（SCFDR）的低 8 位，获取 SCFRDR 的接收数据字节数。

15.6.3 DMAC 使用时的限制事项

通过 TXI 中断请求由 DMAC 向 SCFTDR 写入数据时, TEND 标志的状态不定, 不得将 TEND 标志作为传送结束标志使用。

15.6.4 断点的检测和处理

检测帧错误 (FER) 时, 通过直接读取 RxD 引脚的值, 可检测断点。断点时, RxD 引脚的输入均为 0, 因此置位 FER 标志, 有时也可能置位奇偶校验错误标志 (PER)。

如果检测出断点, SCIF 则停止向 SCFRDR 传送接收数据, 但仍继续接收。

15.6.5 断点的发送

TxD 引脚的输入 / 输出条件和电平, 由串行端口寄存器 (SCSPTR) 的 SPB2IO 位和 SPB2DR 位决定。可利用此功能发送断点。

从串行发送的初始化到 TE 位置 1 (允许发送) 的期间, TxD 引脚无效。在此期间, 由 SPB2DT 位的值代替标记状态。因此, 首先要将 SPB2IO 和 SPB2DT 位设定为 1 (输出高电平)。

为了在串行发送时发送断点信号, 必须将 SPB2DT 位清 0 (指定低电平) 后, 将 TE 位清 0 (停止发送)。如果将 TE 位清 0, 则与当前发送状态无关, 初始化发送部并从 TxD 引脚输出 0。

15.6.6 异步模式的接收数据采样时序与接收容限

SCIF 以频率为 16/8 倍位速率的基本时钟运行。接收时, SCIF 通过基本时钟采样开始位的下降沿后同步内部。并在基本时钟的第 8/4 个时钟的上升沿, 将接收数据输入内部。以频率为 16 倍位速率的基本时钟运行时的时序如图 15.19 所示。

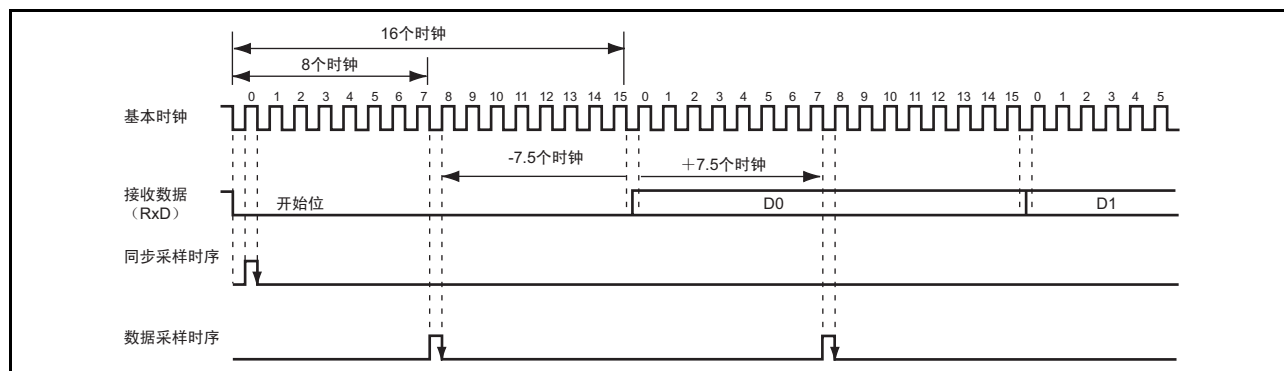


图 15.19 异步模式的接收数据采样时序 (以频率为 16 倍位速率的基本时钟运行)

用公式 (1) 表示异步模式时的接收容限。

$$M = \left\{ \left(0.5 - \frac{1}{2N} \right) - (L - 0.5) F - \frac{|D - 0.5|}{N} (1 + F) \right\} \times 100\% \quad \dots \text{公式 (1)}$$

M: 接收容限 (%)

N: 时钟频率与位速率的比 (N=16/8)

D: 时钟占空比 (D:0 ~ 1.0)

L: 帧长度 (L=9 ~ 12)

F: 时钟频率偏差的绝对值

如果公式 (1) 中 F=0、D=0.5、N=16, 则根据公式 (2) 得出接收容限为 46.875%。

D=0.5、F=0 时

$$M = (0.5 - 1 / (2 \times 16)) \times 100\% = 46.875\% \quad \dots \text{公式 (2)}$$

但是, 此值为计算值。系统设计时必须留有 20 ~ 30% 的余地。

15.6.7 选择异步基本时钟

本 LSI 通过更改串行扩展模式寄存器 (SCEMR) 的 ABCS 位, 可将异步模式时 1 位期间的基本时钟频率设定为位速率的 16/8 倍。

但必须注意: 根据 “15.6.6 异步模式的接收数据采样时序与接收容限” 的公式 (1), 如果基本时钟的频率低于 8 倍位速率的值时, 接收容限将减少。

如果仅通过设定串行模式寄存器 (SCSMR) 的 CKS[1:0]、位速率寄存器 (SCBRR) 才能得到期望的位速率, 则建议将 1 位期间的基本时钟频率设定为位速率的 16 倍 (SCEMR 的 ABCS=0)。如果选择内部时钟为时钟源 / 不使用 SCK 引脚, 则可通过设定波特率发生器倍速模式 (SCEMR 的 BGDM=1), 在不减少接收容限的情况下提高位速率。

第 16 章 同步串行通信单元 (SSU)

本 LSI 具有 2 个通道的同步串行通信单元 (SSU: Synchronous Serial communication Unit)。SSU 有以本 LSI 为主器件, 向外部输出时钟并执行同步串行通信的主模式及输入来自外部器件的时钟并执行同步串行通信的从属模式。可在时钟极性与时钟相位不同的器件之间执行同步串行通信。

16.1 特点

- 可选择 SSU 模式与时钟同步通信模式
- 可选择主模式与从属模式
- 可选择标准模式与双向模式
- 可在时钟相位与时钟极性不同的其他器件之间执行同步串行通信
- 可选择发送/接收数据长度为 8 位、16 位或 32 位
- 可执行全双工通信
配备可同时执行发送与接收的移位寄存器
- 可连续串行通信
- 可选择 LSB first 方式/MSB first 方式
- 可选择内部时钟 (Pφ/4、Pφ/8、Pφ/16、Pφ/32、Pφ/64、Pφ/128、Pφ/256 等 7 种) 与外部时钟作为时钟源
- 中断源: 5 种
即: 发送结束、发送数据空、接收数据满、溢出错误及竞争错误等
可由发送数据空请求或接收数据满请求启动直接存储器存取控制器 (DMAC), 并执行数据传送。
- 可设定模块待机模式
为了降低功耗, 可停止向 SSU 提供时钟, 中止 SSU 运行。

SSU 框图如图 16.1 所示。

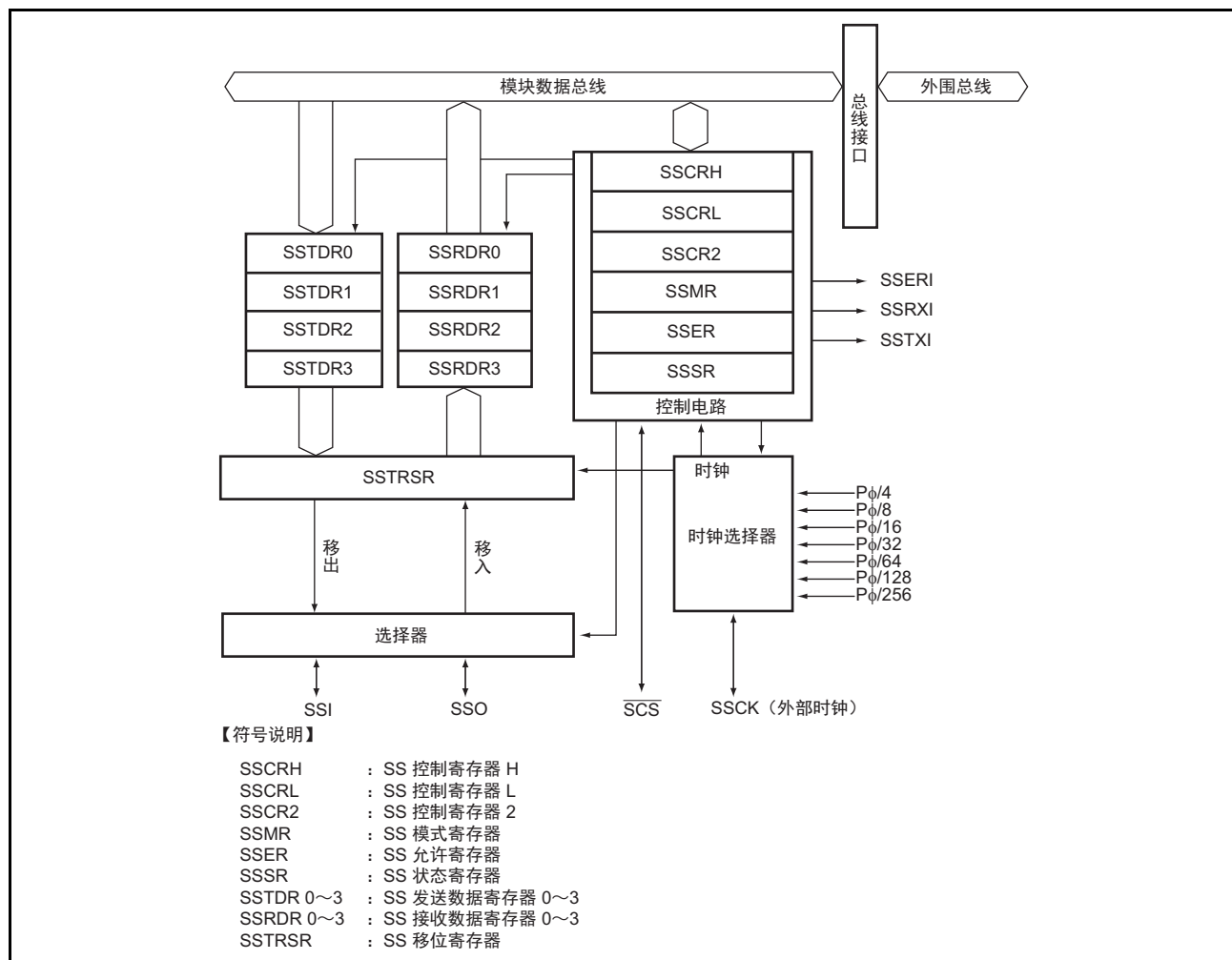


图 16.1 SSU 框图

16.2 输入 / 输出引脚

SSU 有表 16.1 所示的输入 / 输出引脚。

表 16.1 引脚结构

通道	引脚名称	输入 / 输出	功 能
0、1	SSCK0、SSCK1	输入 / 输出	SSU 时钟输入 / 输出引脚
	SSI0、SSI1	输入 / 输出	SSU 数据输入 / 输出引脚
	SSO0、SSO1	输入 / 输出	SSU 数据输入 / 输出引脚
	SCS0、SCS1	输入 / 输出	SSU 片选输入 / 输出引脚

16.3 寄存器说明

SSU 有以下寄存器。有关寄存器的地址及各处理状态的寄存器状态，详情参阅“第 30 章 寄存器一览表”。

表 16.2 寄存器结构

通道	寄存器名称	简称	R/W	初始值	地址	存取长度
0	SS 控制寄存器 H_0	SSCRH_0	R/W	H'0D	H'FFFE7000	8、16
	SS 控制寄存器 L_0	SSCRL_0	R/W	H'00	H'FFFE7001	8
	SS 模式寄存器 _0	SSMR_0	R/W	H'00	H'FFFE7002	8、16
	SS 允许寄存器 _0	SSER_0	R/W	H'00	H'FFFE7003	8
	SS 状态寄存器 _0	SSSR_0	R/W	H'04	H'FFFE7004	8、16
	SS 控制寄存器 2_0	SSCR2_0	R/W	H'00	H'FFFE7005	8
	SS 发送数据寄存器 0_0	SSTDR0_0	R/W	H'00	H'FFFE7006	8、16
	SS 发送数据寄存器 1_0	SSTDR1_0	R/W	H'00	H'FFFE7007	8
	SS 发送数据寄存器 2_0	SSTDR2_0	R/W	H'00	H'FFFE7008	8、16
	SS 发送数据寄存器 3_0	SSTDR3_0	R/W	H'00	H'FFFE7009	8
	SS 接收数据寄存器 0_0	SSRDR0_0	R	H'00	H'FFFE700A	8、16
	SS 接收数据寄存器 1_0	SSRDR1_0	R	H'00	H'FFFE700B	8
	SS 接收数据寄存器 2_0	SSRDR2_0	R	H'00	H'FFFE700C	8、16
	SS 接收数据寄存器 3_0	SSRDR3_0	R	H'00	H'FFFE700D	8
1	SS 控制寄存器 H_1	SSCRH_1	R/W	H'0D	H'FFFE7800	8、16
	SS 控制寄存器 L_1	SSCRL_1	R/W	H'00	H'FFFE7801	8
	SS 模式寄存器 _1	SSMR_1	R/W	H'00	H'FFFE7802	8、16
	SS 允许寄存器 _1	SSER_1	R/W	H'00	H'FFFE7803	8
	SS 状态寄存器 _1	SSSR_1	R/W	H'04	H'FFFE7804	8、16
	SS 控制寄存器 2_1	SSCR2_1	R/W	H'00	H'FFFE7805	8

通道	寄存器名称	简称	R/W	初始值	地址	存取长度
1	SS 发送数据寄存器 0_1	SSTDR0_1	R/W	H'00	H'FFFE7806	8、16
	SS 发送数据寄存器 1_1	SSTDR1_1	R/W	H'00	H'FFFE7807	8
	SS 发送数据寄存器 2_1	SSTDR2_1	R/W	H'00	H'FFFE7808	8、16
	SS 发送数据寄存器 3_1	SSTDR3_1	R/W	H'00	H'FFFE7809	8
	SS 接收数据寄存器 0_1	SSRDR0_1	R	H'00	H'FFFE780A	8、16
	SS 接收数据寄存器 1_1	SSRDR1_1	R	H'00	H'FFFE780B	8
	SS 接收数据寄存器 2_1	SSRDR2_1	R	H'00	H'FFFE780C	8、16
	SS 接收数据寄存器 3_1	SSRDR3_1	R	H'00	H'FFFE780D	8

16.3.1 SS 控制寄存器 H (SSCRH)

SSCRH 可设定主 / 从属器件选择、双向模式允许、SSO 引脚输出值选择、SSCK 引脚及 SCS 引脚选择。

位:	7	6	5	4	3	2	1	0
	MSS	BIDE	-	SOL	SOLP	-	CSS[1:0]	
初始值:	0	0	0	0	1	1	0	1
R/W:	R/W	R/W	R	R/W	R/W	R	R/W	R/W

位	位名称	初始值	R/W	说 明
7	MSS	0	R/W	选择主 / 从属器件 选择 SSU 是用作主模式或是从属模式。用作主模式时，从 SSCK 引脚输出传送时钟。SSSR 的 CE 位置位时，自动清除本位。 0: 选择从属模式 1: 选择主模式
6	BIDE	0	R/W	双向模式允许 选择使用串行数据输入、输出这 2 个引脚或仅使用其中 1 个引脚。但选择双向模式时，不得同时执行发送 / 接收。详情参阅“16.4.3 数据输入 / 输出引脚与移位寄存器的关系”。 0: 标准模式（使用数据输入与输出 2 个引脚执行通信） 1: 双向模式（仅使用数据输入或输出的 1 个引脚执行通信）
5	—	0	R	保留位 读取值、写入值总是为 0。
4	SOL	0	R/W	选择串行数据输出值 发送结束后的串行数据输出保存发送数据最末位的值，但发送前或发送后可更改串行数据的输出电平。更改输出电平时，必须由 MOV 指令将 SOLP 位清 0。数据传送过程中对本位写入会产生误动作，因此发送过程中不得对本位写入。 0: 串行数据的输出电平更改为低电平 1: 串行数据的输出电平更改为高电平
3	SOLP	1	R/W	SOL 位写入保护 更改串行数据的输出电平时，必须由 MOV 指令设定 SOL=1 且 SOLP=0，或 SOL=0 且 SOLP=0。 0: 可根据 SOL 的值更改输出电平 1: 不可根据 SOL 的值更改输出电平 对本位写入 0 时，必须在确认读取 1 后写入 0。

位	位名称	初始值	R/W	说 明
2	—	1	R	保留位 读取值、写入值总是为 1。
1、0	CSS[1:0]	01	R/W	选择 $\overline{\text{SCS}}$ 引脚 选择将 $\overline{\text{SCS}}$ 引脚用作 $\overline{\text{SCS}}$ 输入或 $\overline{\text{SCS}}$ 输出。 00: 禁止设定 01: 禁止设定 10: $\overline{\text{SCS}}$ 自动输入 / 输出功能 (传送前、传送后为 $\overline{\text{SCS}}$ 输入, 传送过程中为低电平输出) 11: $\overline{\text{SCS}}$ 自动输出功能 (传送前、传送后为高电平输出, 传送过程中为低电平输出)

16.3.2 SS 控制寄存器 L (SSCRL)

SSCRL 可选择运行模式、软件复位及发送 / 接收数据的长度。

位:	7	6	5	4	3	2	1	0
	-	SSUMS	SRES	-	-	-	DATS[1:0]	
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R	R	R	R/W	R/W

位	位名称	初始值	R/W	说 明
7	—	0	R	保留位 读取值、写入值总是为 0。
6	SSUMS	0	R/W	选择 SSU 模式与时钟同步通信模式。 0: SSU 模式 1: 时钟同步通信模式
5	SRES	0	R/W	软件复位 如果本位置 1, 则强行复位 SSU 内部定时器。此后, 自动清除本位, 初始化 SSSR 的 ORER、TEND、TDRE、RDRF、CE 各位及 SSER 的 TE、RE 位, 且保持其他 SSU 内部寄存器的值。 欲在中途中断传送, 必须对本位写入 1, 并复位内部定时器。
4 ~ 2	—	均为 0	R	保留位 读取值、写入值总是为 0。
1、0	DATS[1:0]	00	R/W	选择发送 / 接收数据长度 选择串行数据长度。 00: 8 位数据长度 01: 16 位数据长度 10: 32 位数据长度 11: 设定无效

16.3.3 SS 模式寄存器 (SSMR)

SSMR 可选择 MSB first/LSB first、时钟极性、时钟相位及传送时钟速率。

位:	7	6	5	4	3	2	1	0
	MLS	CPOS	CPHS	-	-	CKS[2:0]		
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R	R	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7	MLS	0	R/W	选择 MSB first/LSB first 选择以 MSB first 或 LSB first 传送串行数据。 0: LSB first 1: MSB first
6	CPOS	0	R/W	选择时钟极性 选择 SSCK 时钟的极性。 0: 空闲时为高电平输出, 有效时为低电平输出 1: 空闲时为低电平输出, 有效时为高电平输出
5	CPHS	0	R/W	选择时钟相位 (仅在 SSU 模式时有效) 选择 SSCK 时钟的相位。 0: 在第一个边沿数据变化 1: 在第一个边沿锁存数据
4、3	—	均为 0	R	保留位 读取值、写入值总是为 0。
2 ~ 0	CKS[2:0]	000	R/W	选择传送时钟速率 选择内部时钟时的传送时钟速率 (预分频器分频比)。 000: 保留 001: P ϕ /4 010: P ϕ /8 011: P ϕ /16 100: P ϕ /32 101: P ϕ /64 110: P ϕ /128 111: P ϕ /256

16.3.4 SS 允许寄存器 (SSER)

SSER 可设定发送允许、接收允许及中断请求允许。

位:	7	6	5	4	3	2	1	0
	TE	RE	-	-	TEIE	TIE	RIE	CEIE
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7	TE	0	R/W	发送允许 如果本位置 1，则可发送。
6	RE	0	R/W	接收允许 如果本位置 1，则可接收。
5、4	—	均为 0	R	保留位 读取值、写入值总是为 0。
3	TEIE	0	R/W	发送结束中断允许 如果本位置 1，则允许发送结束时的 SSTXI 中断请求。
2	TIE	0	R/W	发送中断允许 如果本位置 1，则允许发送数据空时的 SSTXI 中断请求。
1	RIE	0	R/W	接收中断允许 如果本位置 1，则允许 SSRXI 中断及溢出错误时的 SSERI 中断请求。
0	CEIE	0	R/W	竞争错误中断允许 如果本位置 1，则允许竞争错误时的 SSERI 中断请求。

16.3.5 SS 状态寄存器 (SSSR)

SSSR 为各种中断的状态标志寄存器。

位:	7	6	5	4	3	2	1	0
	-	ORER	-	-	TEND	TDRE	RDRF	CE
初始值:	0	0	0	0	0	1	0	0
R/W:	R	R/W	R	R	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7	—	0	R	保留位 读取值、写入值总是为 0。
6	ORER	0	R/W	溢出错误 表示在 RDRF=1 的状态，如果接收下一个数据，则产生溢出错误，并异常结束。SSRDR 保持产生溢出错误前的 1 帧接收数据，之后接收到的数据丢失。且在 ORER=1 的状态，不得继续此后的串行接收 / 串行发送。另外，SSU 模式 (SSCRL 的 SSUMS=0) 的从属数据接收运行 (SSCRH 的 MSS=0 且 SSER 的 TE=0、RE=1) 时，本位无效。 [置位条件] • 在 RDRF=1 的状态，下一个串行接收的 1 字节结束时 (除 SSU 模式的从属数据接收运行之外) [清除条件] • 读取 1 后写入 0 时
5、4	—	均为 0	R	保留位 读取值、写入值总是为 0。
3	TEND	0	R/W	发送结束 [置位条件] • SSCR2 的 TENDSTS 为 0 时，在 TDRE=1 的状态，发送数据的最末位时 • SSCR2 的 TENDSTS 为 1 时，在 TDRE=1 的状态，发送数据的最末位后 [清除条件] • 读取 TEND=1 的状态后，对 TEND 标志写入 0 时 • 对 SSTDR 写入数据时

位	位名称	初始值	R/W	说 明
2	TDRE	1	R/W	发送数据空 表示 SSTDR 内有无数据。 [置位条件] - SSER 的 TE 为 0 时 - 从 SSTDR 向 SSTRSR 传送数据, 并可对 SSTDR 写入数据时 [清除条件] - 读取 TDRE=1 的状态后, 对 TDRE 标志写入 0 时 - 在 TE=1 的状态, 对 SSTDR 写入数据时 - 由 SSTXI 中断启动 DMAC, 通过 DMAC 传送对 SSTDR 写入传送数据时
1	RDRF	0	R/W	接收数据满 表示 SSRDR 内有无数据。 [置位条件] - 串行接收正常结束, 并从 SSTRSR 向 SSRDR 传送接收数据时 [清除条件] - 读取 RDRF=1 的状态后, 对 RDRF 标志写入 0 时 - 从 SSRDR 读取接收数据时 - 由 SSRXI 中断启动 DMAC, 通过 DMAC 传送从 SSRDR 读取接收数据时
0	CE	0	R/W	竞争错误 / 未完成错误 表示在 SSUMS=0 (SSU 模式)、MSS=1 (主器件) 的状态, 由外部向 $\overline{SCS}$ 输入 0 时, 产生竞争错误。另外, 在 SSUMS=0 (SSU 模式)、MSS=0 (从属器件) 的状态, $\overline{SCS}$ 引脚为 1 时, 判断主器件传送停止, 并产生未完成错误。如果在下一帧开始时, 尚未在 SSU 模式的从属器件接收运行状态读取接收数据 (读取 SSRDR) 及清除 SSSR 的 RDRF, 或在相同从属器件接收运行状态写入发送数据 (写入 SSTDR) 及清除 SSSR 的 TDRE, 则会在该帧结束时产生未完成错误。 在 CE=1 的状态, 不得继续此后的串行接收或发送。开始重新传送前, 必须将 SSCRL 的 SRES 置 1, 并复位内部定序器。 [置位条件] - 为主器件 (SSCRH 的 MSS=1) 并向 $\overline{SCS}$ 引脚输入低电平时 - 为从属器件 (SSCRH 的 MSS=0) 并在传送过程中 $\overline{SCS}$ 引脚变为 1 时 - 在从属器件接收运行期间, SSRDR 读取及 RDRF 清除在下一帧开始前未结束, 而该帧已结束 - 在从属器件发送运行期间, SSTDR 写入及 TDRE 清除在下一帧开始前未结束, 而该帧已结束 [清除条件] - 读取 1 后写入 0 时

16.3.6 SS 控制寄存器 2 (SSCR2)

SSCR2 寄存器可设定 $\overline{\text{SCS}}$ 引脚的有效时序、SSO 引脚的数据输出时序及 TEND 位的置位时序。

位:	7	6	5	4	3	2	1	0
	-	-	-	TENDSTS	SCSATS	SSODTS	-	-
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R	R

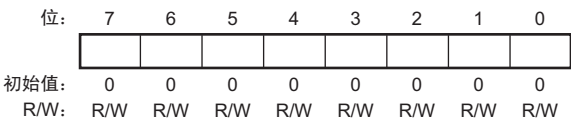
位	位名称	初始值	R/W	说 明
7 ~ 5	—	均为 0	R	保留位 读取值、写入值总是为 0。
4	TENDSTS	0	R/W	选择 TEND 位的置位时序（仅在设定 SSU 模式及主模式时有效） 0: 发送最末位的过程中置位 TEND 位 1: 发送最末位后置位 TEND 位
3	SCSATS	0	R/W	选择 $\overline{\text{SCS}}$ 引脚的有效时序（仅在设定 SSU 模式及主模式时有效） 0: t_{LEAD} 、 t_{LAG} 输出期间的 Min. 为 $1/2 \times t_{\text{SUcyc}}$ 1: t_{LEAD} 、 t_{LAG} 输出期间的 Min. 为 $3/2 \times t_{\text{SUcyc}}$
2	SSODTS	0	R/W	选择 SSO 引脚的数据输出时序（仅在设定 SSU 模式及主模式时有效） 0: BIDE=0、MSS=1、TE=1，或 BIDE=1、TE=1、RE=0 时，SSO 引脚输出数据 1: BIDE=0、MSS=1、TE=1，或 BIDE=1、TE=1、RE=0 且 $\overline{\text{SCS}}$ 引脚为低电平期间，SSO 引脚输出数据
1、0	—	均为 0	R	保留位 读取值、写入值总是为 0。

16.3.7 SS 发送数据寄存器 0 ～ 3 （SSTDR0 ～ SSTDR3）

SSTDR 为 8 位保存发送数据的寄存器。根据 SSCRL 的 DATS1、DATS0 位的设定，选择 8 位数据长度时 SSTDR0 有效；选择 16 位数据长度时 SSTDR0、SSTDR1 有效；选择 32 位数据长度时 SSTDR0、SSTDR1、SSTDR2、SSTDR3 有效。不得存取无效的 SSTDR。

如果检测出 SSTRSR 为空，则 SSU 将写入 SSTDR 的发送数据传送至 SSTRSR，并开始串行发送。
SSTRSR 串行数据发送过程中，如果预先将下一个数据写入 SSTDR，则可连续串行发送。

CPU 和 DMAC 总是可读取 / 写入 SSTDR，但为了确实执行串行通信，必须在确认 SSSR 的 TDRE 置 1 后写入 SSTDR。



位	位名称	初始值	R/W	说 明
7 ～ 0		均为 0	R/W	串行发送数据

表 16.3 DATS 位的设定与 SSTDR 的对应表

SSTDR	DATS[1:0]（SSCRL[1:0]）			
	00	01	10	11（设定无效）
0	有效	有效	有效	无效
1	无效	有效	有效	无效
2	无效	无效	有效	无效
3	无效	无效	有效	无效

16.3.8 SS 接收数据寄存器 0 ~ 3 (SSRDR0 ~ SSRDR3)

SSRDR 为 8 位保存接收数据的寄存器。根据 SSCRL 的 DATS1、DATS0 位的设定，选择 8 位数据长度时 SSRDR0 有效；选择 16 位数据长度时 SSRDR0、SSRDR1 有效；选择 32 位数据长度时 SSRDR0、SSRDR1、SSRDR2、SSRDR3 有效。不得存取无效的 SSRDR。

接收完 1 字节的数据时，SSU 从 SSTRSR 向 SSRDR 传送并保存接收的串行数据。此后，SSTRSR 为可接收状态。SSTRSR 与 SSRDR 为双缓冲结构，所以可连续接收。

必须在确认 SSSR 寄存器的 RDRF 位置 1 后读取 SSRDR。

SSRDR 为只读寄存器，不可由 CPU 写入。

位:	7	6	5	4	3	2	1	0
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名称	初始值	R/W	说 明
7 ~ 0		均为 0	R	串行接收数据

表 16.4 DATS 位的设定与 SSRDR 的对应表

SSRDR	DATS[1:0] (SSCRL[1:0])			
	00	01	10	11 (设定无效)
0	有效	有效	有效	无效
1	无效	有效	有效	无效
2	无效	无效	有效	无效
3	无效	无效	有效	无效

16.3.9 SS 移位寄存器 (SSTRSR)

SSTRSR 为发送 / 接收串行数据的移位寄存器。

从 SSTDR 向 SSTRSR 传送发送数据时，如果 SSMR 的 MLS=0，则传送 SSTDR 的 bit0 (LSB first 通信)；如果 MLS=1 时，则传送 SSTDR 的 bit7 (MSB first 通信)。之后，通过从 SSTRSR 的 LSB (bit0) 开始按顺序向 SSO 引脚传送数据，来发送串行数据。

接收时，按从 LSB (bit0) 开始的接收顺序对 SSTRSR 设置从 SSI 引脚输入的串行数据。如果接收完 1 字节的数据，则自动将 SSTRSR 的数据传送至 SSRDR。不可由 CPU 直接存取 SSTRSR。

位:	7	6	5	4	3	2	1	0
初始值:	-	-	-	-	-	-	-	-
R/W:	-	-	-	-	-	-	-	-

16.4 运行说明

16.4.1 传送时钟

可从内部时钟（7 种）与外部时钟选择传送时钟。首先，使用本模块时，需预先通过 PFC 将 SSCK 引脚设置为有效。SSCRH 的 MSS=1 时，选择内部时钟，SSCK 引脚为输出引脚。开始传送时，从 SSCK 引脚输出在 SSMR 的 CKS2 ~ CKS0 设定的传送速率的时钟。MSS=0 时，选择外部时钟，SSCK 引脚为输入引脚。

16.4.2 时钟的相位、极性与数据的关系

SSCRL 的 SSUMS=0 时，通过组合 SSMR 的 CPOS 与 CPHS，可改变时钟的相位、极性与传送数据的关系，这些关系如图 16.2 所示。SSUMS=1 时，CPOS 的设定有效，但 CPHS 的设定无效。SSUMS=1 时的发送数据的变化时序及接收数据的输入时序，与图 16.2 的“(1) CPHS=0 时”的时序相同。

另外，通过设定 SSMR 的 MLS，可选择 MSB first 传送或 LSB first 传送。MLS=0 时，按照从 LSB 到 MSB 的顺序传送；MLS=1 时，则按照从 MSB 到 LSB 的顺序传送。

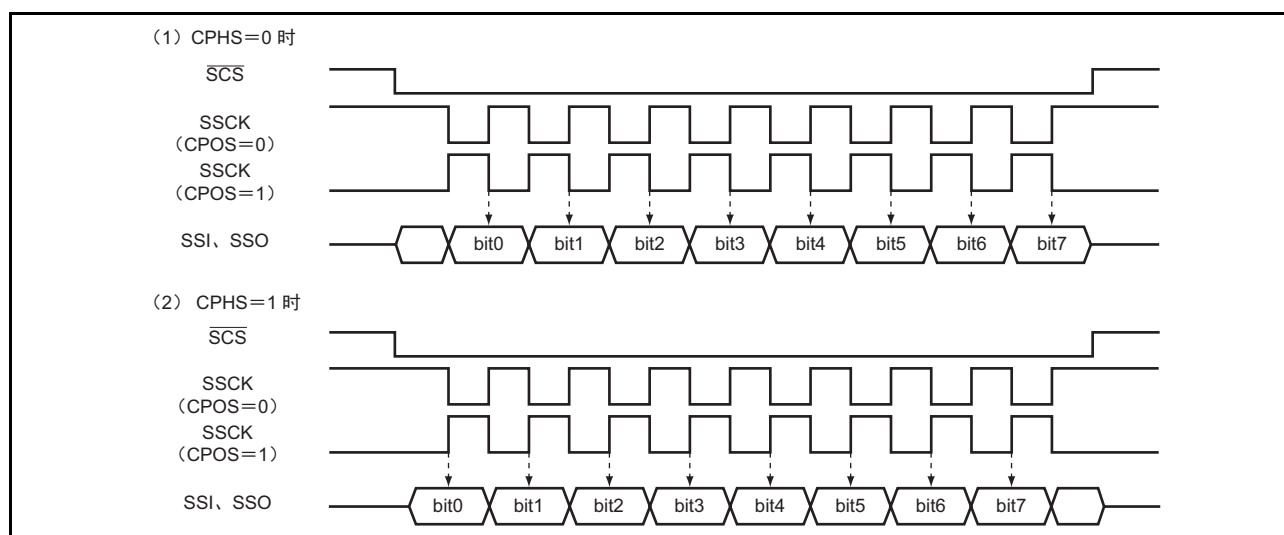


图 16.2 时钟的相位、极性与数据的关系

16.4.3 数据输入 / 输出引脚与移位寄存器的关系

通过组合 SSCRH 的 MSS、BIDE 和 SSCRL 的 SSUMS，会改变数据输入 / 输出引脚与 SS 移位寄存器 (SSTRSR) 的连接关系，这些连接关系如图 16.3 所示。

SSU 在 BIDE=0、MSS=1 (标准、主模式) 的状态运行时，从 SSO 引脚发送串行数据，从 SSI 引脚接收串行数据 (图 16.3 (1))；在 BIDE=0、MSS=0 (标准、从属模式) 的状态运行时，从 SSI 引脚发送串行数据，从 SSO 引脚接收串行数据 (图 16.3 (2))。

在 BIDE=1 (双向模式) 的状态，与主模式及从属模式无关，从 SSO 引脚发送或接收串行数据 (图 16.3 (3)、图 16.3 (4))。

不可同时执行 TE 与 RE 同时置 1 时的发送 / 接收，只可选择 TE、RE 中的一个。

在 SSUMS=1 的状态运行时，从 SSO 引脚发送串行数据，从 SSI 引脚接收串行数据。MSS=1 时，从 SSCK 引脚输出内部时钟；MSS=0 时，SSCK 引脚为输入引脚 (图 16.3 (5)、图 16.3 (6))

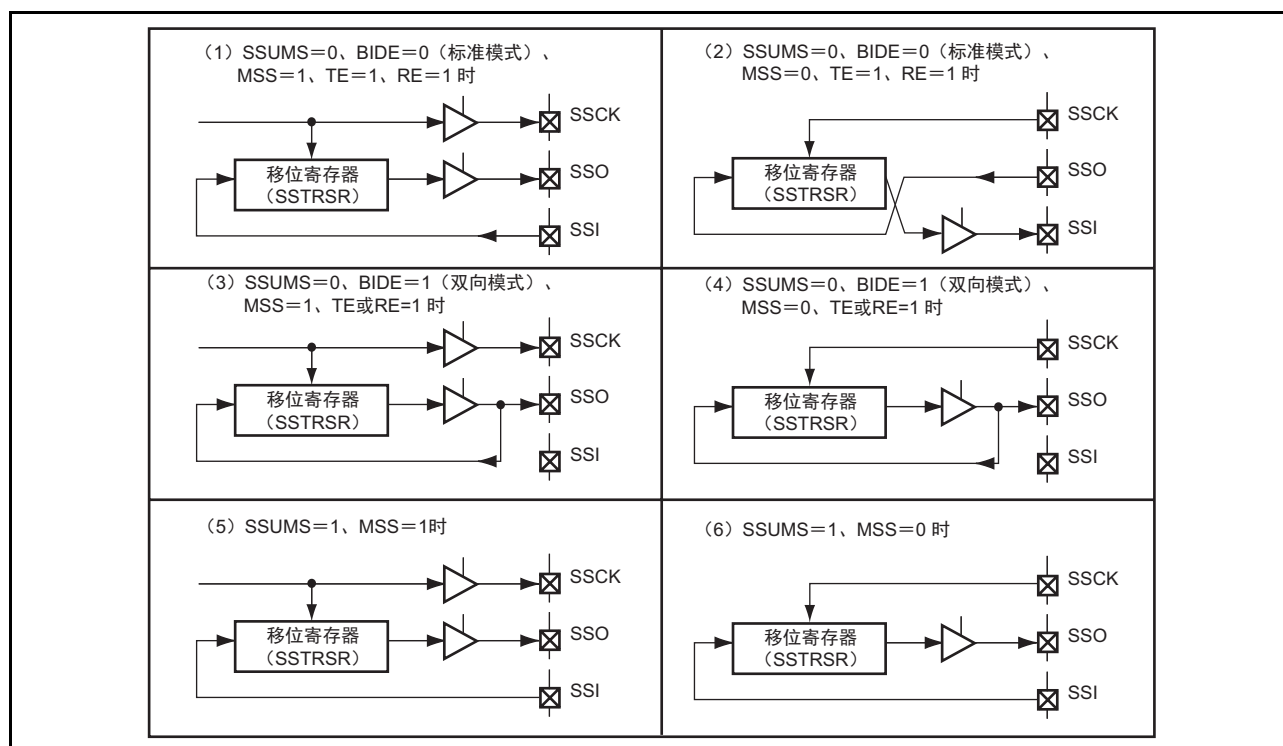


图 16.3 数据输入 / 输出引脚与移位寄存器的关系

16.4.4 各通信模式与引脚功能

SSU 根据各通信模式与寄存器的设定, 转换输入 / 输出引脚 (SSI、SSO、SSCK、 $\overline{\text{SCS}}$) 的功能。各通信模式与输入 / 输出引脚的关系如表 16.5 ~ 表 16.7 所示。

表 16.5 各通信模式与 SSI、SSO 引脚的状态

通信模式	寄存器状态					引脚状态	
	SSUMS	BIDE	MSS	TE	RE	SSI	SSO
SSU 通信模式	0	0	0	0	1	—	输入
				1	0	输出	—
					1	输出	输入
			1	0	1	输入	—
				1	0	—	输出
					1	输入	输出
SSU (双向) 通信模式	0	1	0	0	1	—	输入
				1	0	—	输出
			1	0	1	—	输入
				1	0	—	输出
时钟同步通信模式	1	0	0	0	1	输入	—
				1	0	—	输出
					1	输入	输出
			1	0	1	输入	—
				1	0	—	输出
					1	输入	输出

【符号说明】 — : 不用作 SSU 引脚 (可用作 I/O 端口)

表 16.6 各通信模式与 SSCK 引脚的状态

通信模式	寄存器状态		引脚状态
	SSUMS	MSS	SSCK
SSU 通信模式	0	0	输入
		1	输出
时钟同步通信模式	1	0	输入
		1	输出

表 16.7 各通信模式与 $\overline{\text{SCS}}$ 引脚的状态

通信模式	寄存器状态				引脚状态
	SSUMS	MSS	CSS1	CSS0	$\overline{\text{SCS}}$
SSU 通信模式	0	0	x	x	输入
		1	0	0	(禁止设定)
			0	1	(禁止设定)
			1	0	自动输入 / 输出
			1	1	输出
时钟同步通信模式	1	x	x	x	—

【符号说明】x : Don't care
— : 不用作 SSU 引脚 (可用作 I/O 端口)

16.4.5 SSU 模式

SSU 模式使用时钟线 (SSCK)、数据输入线 (SSI 或 SSO)、数据输出线 (SSI 或 SSO) 及片选 ($\overline{\text{SCS}}$) 等 4 条总线执行数据通信。

另外, 也支持 1 个引脚用作数据输入线及数据输出线的双向模式。

(1) SSU 模式的初始设定

SSU 模式的初始设定例如图 16.4 所示。发送 / 接收数据前, 必须将 SSER 的 TE 及 RE 清 0, 并执行初始设定。

【注】更改运行模式、通信格式时, 必须先将 TE 及 RE 清 0。注意: 如果 TE 清 0, 则 TDRE 位置 1; 但 RE 清 0 时, 仍保持 RDRF、ORER 的各标志及 SSRDR 的内容。

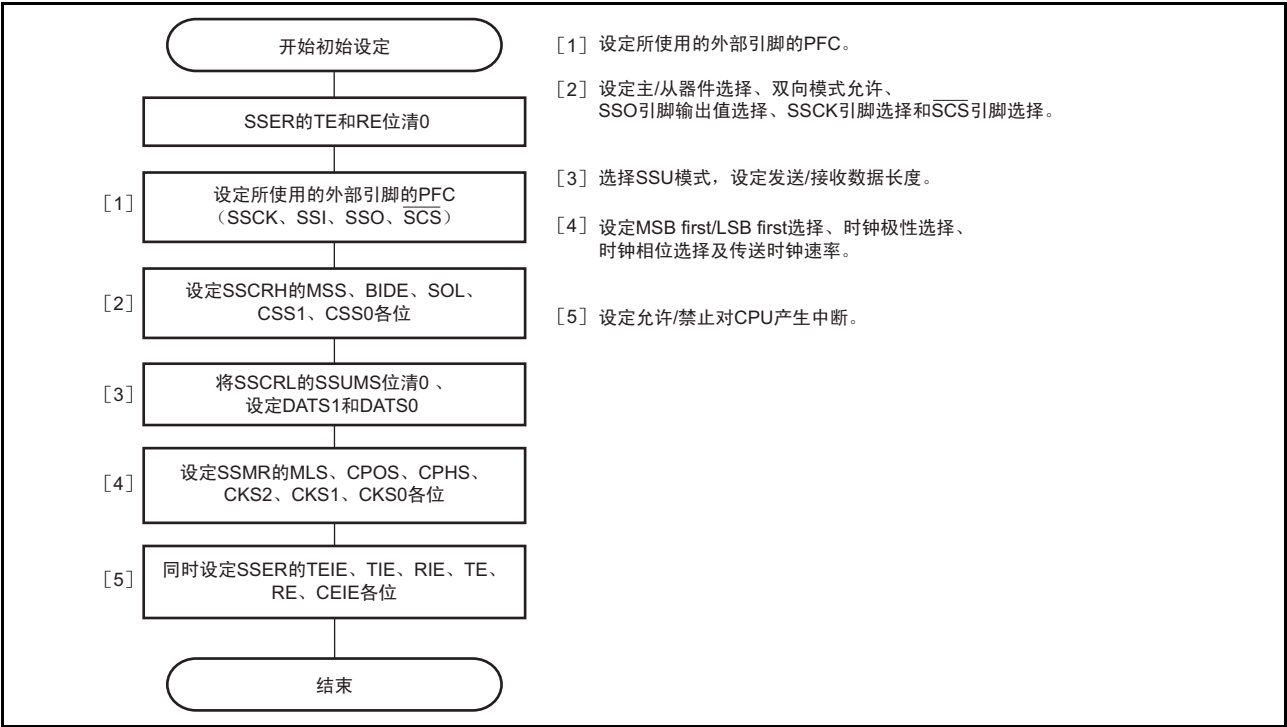


图 16.4 SSU 模式的初始设定例

(2) 发送数据

发送时的运行例如图 16.5 所示，发送数据的流程图的例子如图 16.6 所示。

发送数据时，SSU 如下运行：

将 SSU 设定为主器件时，输出传送时钟与数据；设定为从属器件时，如果向 $\overline{\text{SCS}}$ 引脚输入低电平，并从 SSCK 引脚输入传送时钟，则与该传送时钟同步输出数据。

SSU 在 SSER 的 TE 置 1 后，如果对 SSTDR 写入发送数据，则自动将 SSSR 的 TDRE 清 0，并从 SSTDR 向 SSTRSR 传送数据。之后，将 TDRE 置 1，并开始发送。此时，如果将 SSER 的 TIE 置 1，则产生发送数据空时的 SSTXI 中断请求。

在 TDRE=0 的状态，如果传送完 1 帧的数据，则从 SSTDR 向 SSTRSR 传送数据，并开始发送下一帧。在 TDRE=1 的状态，如果发送第 8 位，则 SSSR 的 TEND 置 1，并保持状态。此时，如果将 SSER 的 TEIE 置 1，则产生发送结束时的 SSTXI 中断。发送结束后，SSCK 引脚在 SSMR 的 CPOS=0 时被固定为高电平；CPOS=1 时，则被固定为低电平。

在 SSSR 的 ORER 置 1 的状态，不可执行发送。发送前必须先确认 ORER 已清 0。

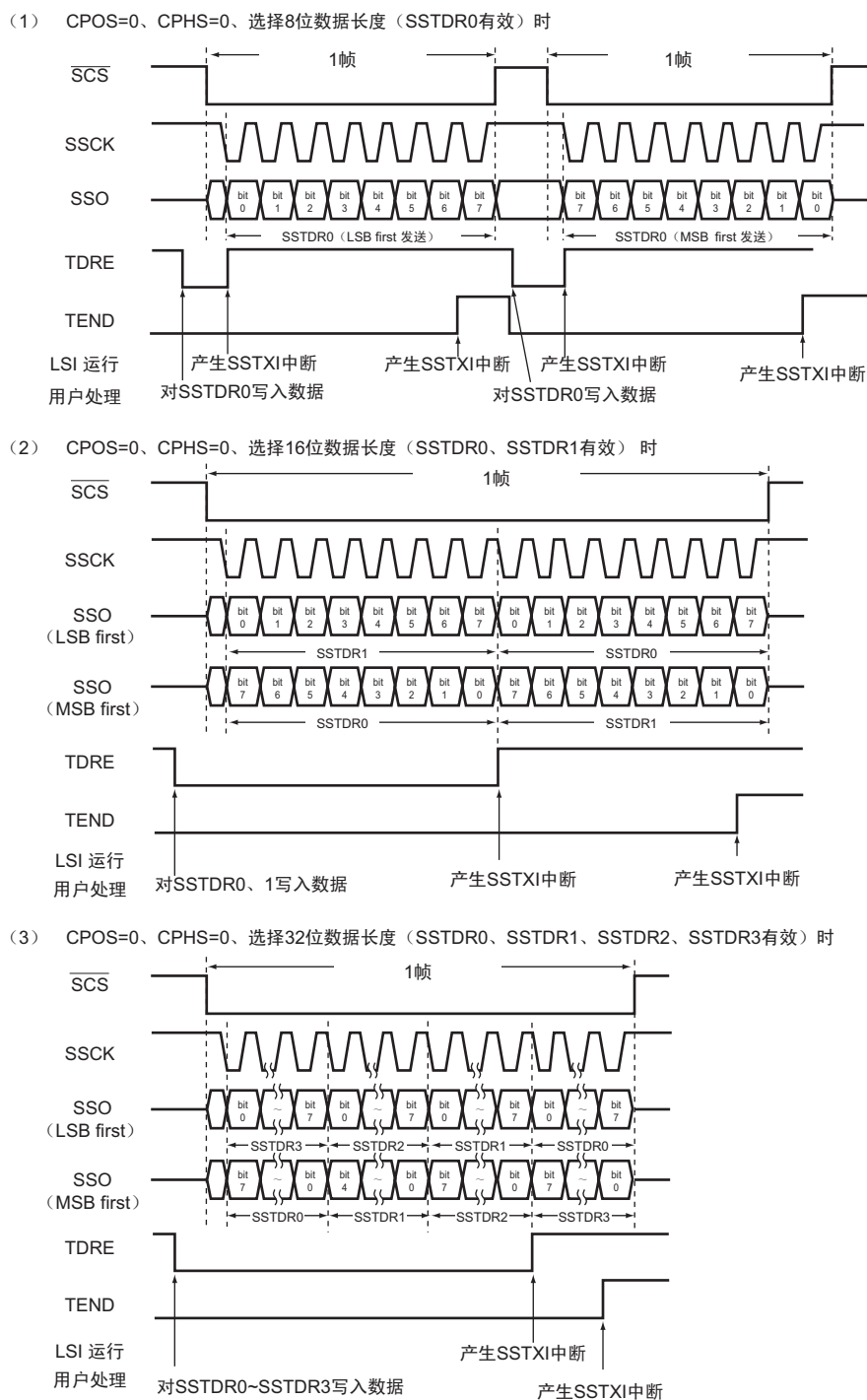


图 16.5 发送时的运行例 (SSU 模式)

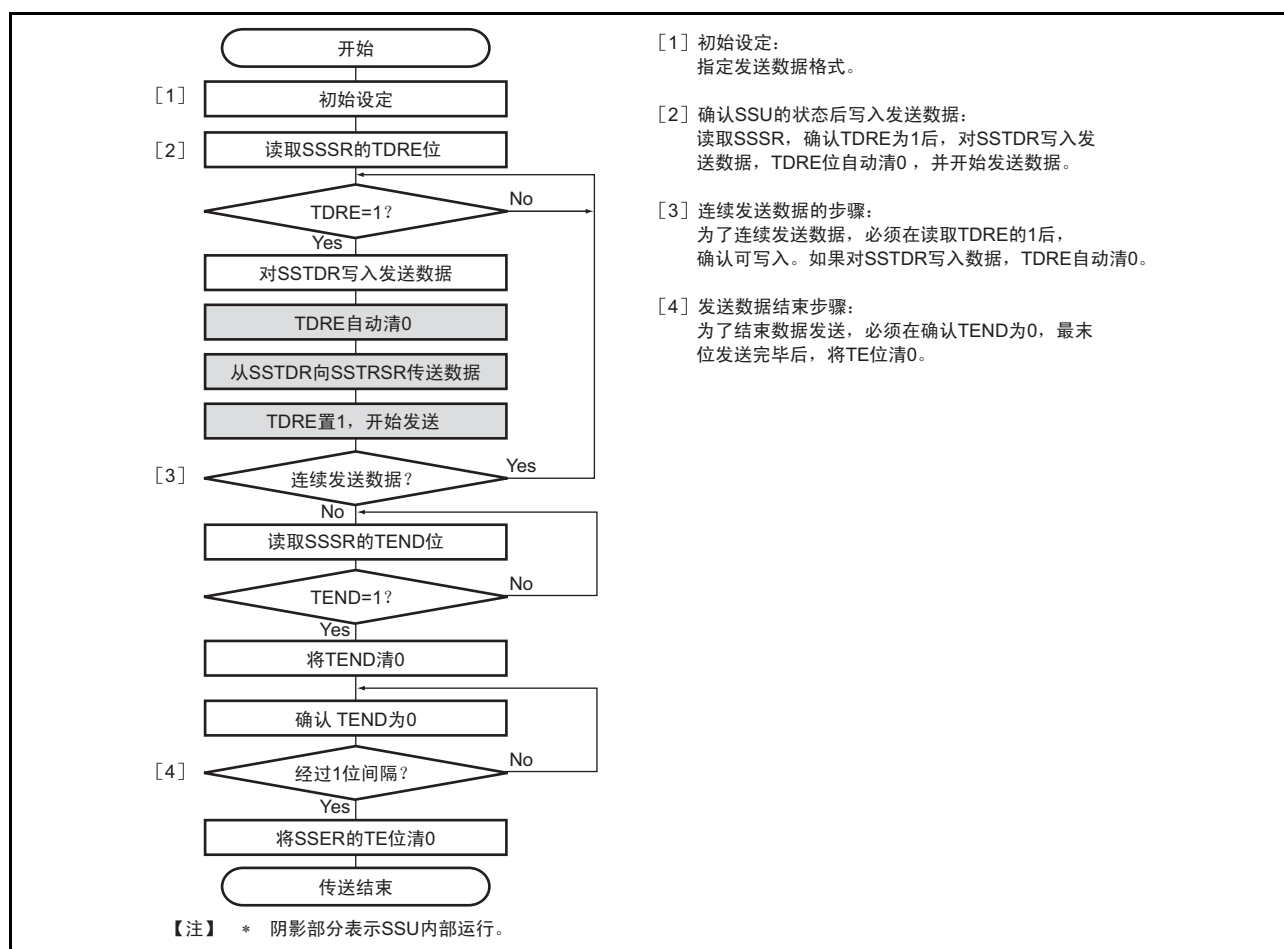


图 16.6 发送数据的流程图的例子 (SSU 模式)

(3) 接收数据

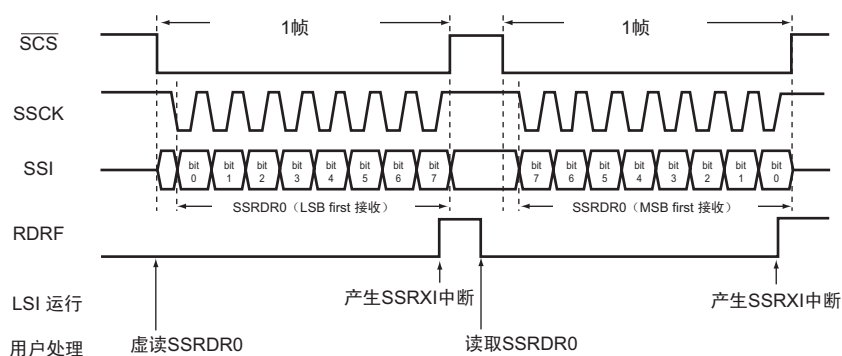
接收时的运行例如图 16.7 所示，接收数据的流程图的例子如图 16.8 所示。接收数据时，SSU 如下运行：
通过将 SSER 的 RE 置 1，并虚读 SSRDR，SSU 开始接收运行。

将 SSU 设定为主器件时，输出传送时钟，并输入接收数据；设定为从属器件时，如果向 $\overline{\text{SCS}}$ 引脚输入低电平，并从 SSCK 引脚输入传送时钟，则与该时钟同步输入接收数据。

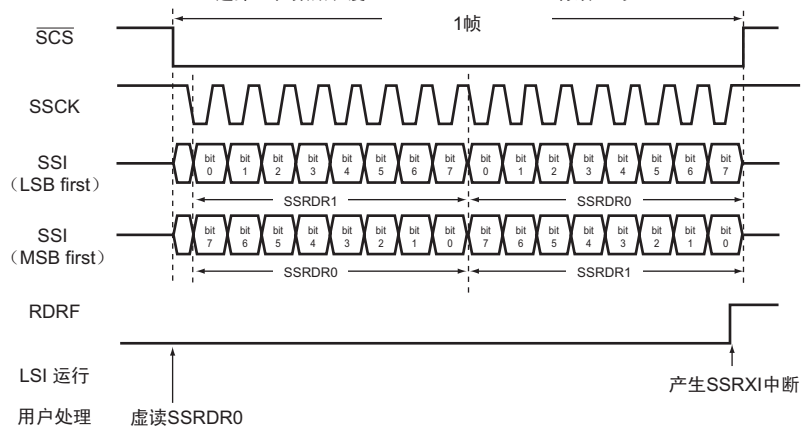
接收 1 帧的数据后，SSSR 的 RDRF 置 1，并在 SSRDR 保存接收数据。此时，如果 SSER 的 RIE 置 1，则产生 SSRXI 中断请求。如果读取 SSRDR，则 RDRF 自动清 0。

在 SSU 的从属接收模式连续接收数据时，必须在开始下一次接收前（与外部连接的主器件开始下一次发送前）读取 SS 接收数据寄存器 (SSRDR)。SS 状态寄存器 (SSSR) 的接收数据满 (RDRF) 位置 1 后，如果在读取 SSRDR 前开始下一次接收，并在接收完 1 帧数据前读取 SSRDR，则在接收完成后将 SSSR 的竞争 / 未完成错误 (CE) 位置 1。另外，RDRF 置 1 后，如果在读取 SSRDR 前开始下一次接收，且在接收完 1 帧数据前未读取 SSRDR，即使 SSSR 的 CE 位或溢出错误位 (ORER) 未置位，也会丢弃接收数据。

(1) CPOS=0, CPHS=0、选择8位数据长度 (SSRDR0有效) 时



(2) CPOS=0, CPHS=0、选择16位数据长度 (SSRDR0、SSRDR1有效) 时



(3) CPOS=0, CPHS=0、选择32位数据长度 (SSRDR0、SSRDR1、SSRDR2、SSRDR3有效) 时

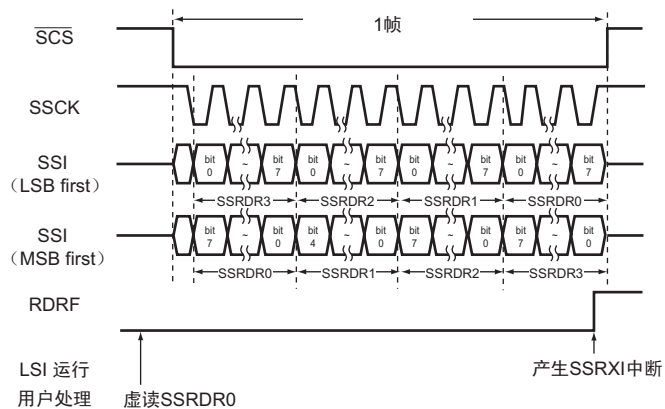


图 16.7 接收时的运行例 (SSU 模式)

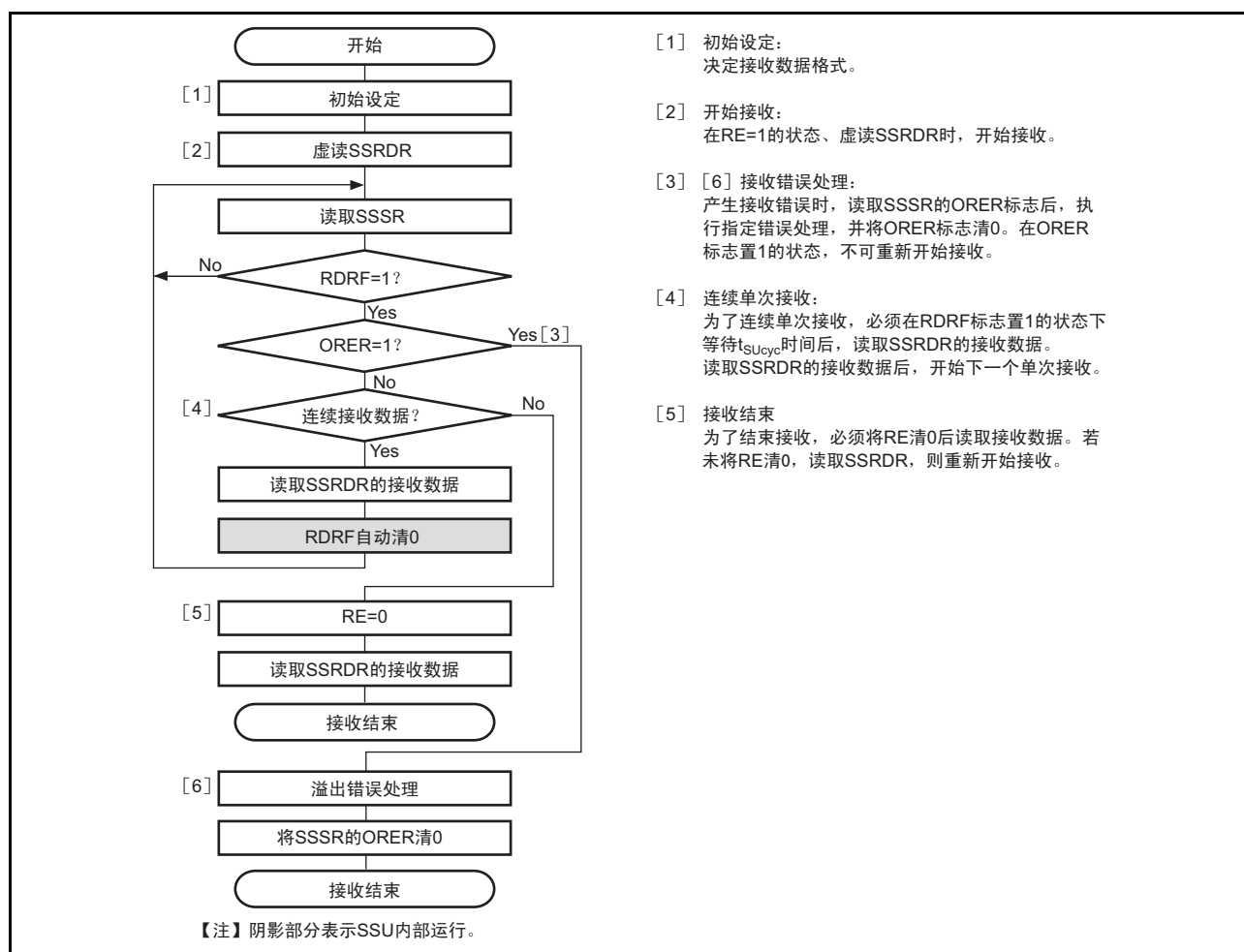


图 16.8 接收数据的流程图的例子 (SSU 模式)

(4) 发送 / 接收数据

发送 / 接收数据同时运行的流程图的例子如图 16.9 所示。发送 / 接收数据为发送数据与接收数据的混合运行。在 $TE=RE=1$ 的状态, 通过对 SSTDR 写入发送数据开始发送 / 接收数据。在 $RDRF=1$ 的状态, 如果第 8 个时钟上升, 则 SSSR 的 ORER 置 1, 并产生溢出错误 (SSERI), 同时停止发送 / 接收。在 $ORER=1$ 的状态不可发送 / 接收数据, 因此重新开始发送 / 接收时, 必须将 ORER 清 0。

从发送模式 ($TE=1$) 或接收模式 ($RE=1$) 转换为发送 / 接收模式 ($TE=RE=1$) 时, 必须先将 TE 、 RE 清 0 一次。另外, 确认 $TEND$ 、 $RDRF$ 及 $ORER$ 清 0 后, 将 TE 及 RE 置 1。

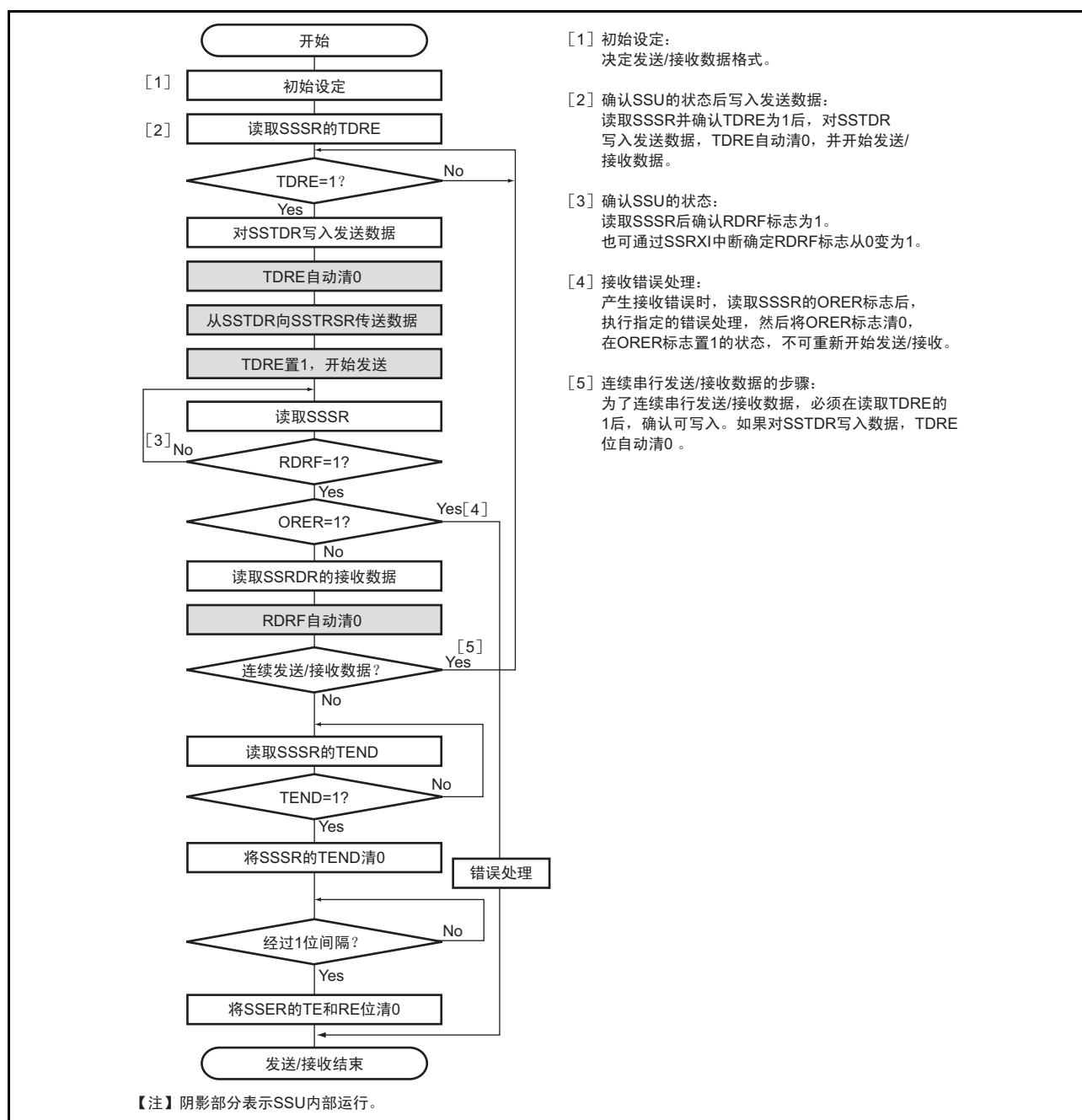


图 16.9 发送 / 接收数据同时运行的流程图的例子 (SSU 模式)

16.4.6 $\overline{\text{SCS}}$ 引脚控制与竞争错误

设定 SSCRH 的 CSS1、CSS0=10、SSCRL 的 SSUMS=0 时，在 SSCRH 的 MSS 置 1 后开始串行传送前与传送结束后， $\overline{\text{SCS}}$ 引脚为输入引脚 (Hi-Z)，并检测竞争错误。在此期间，如果从 $\overline{\text{SCS}}$ 引脚输入低电平，则为竞争错误，SSSR 的 CE 位置位，并清除 MSS。

【注】 在竞争错误置位的状态，不可执行以后的发送 / 接收。开始发送 / 接收前，必须将 CE 清 0。

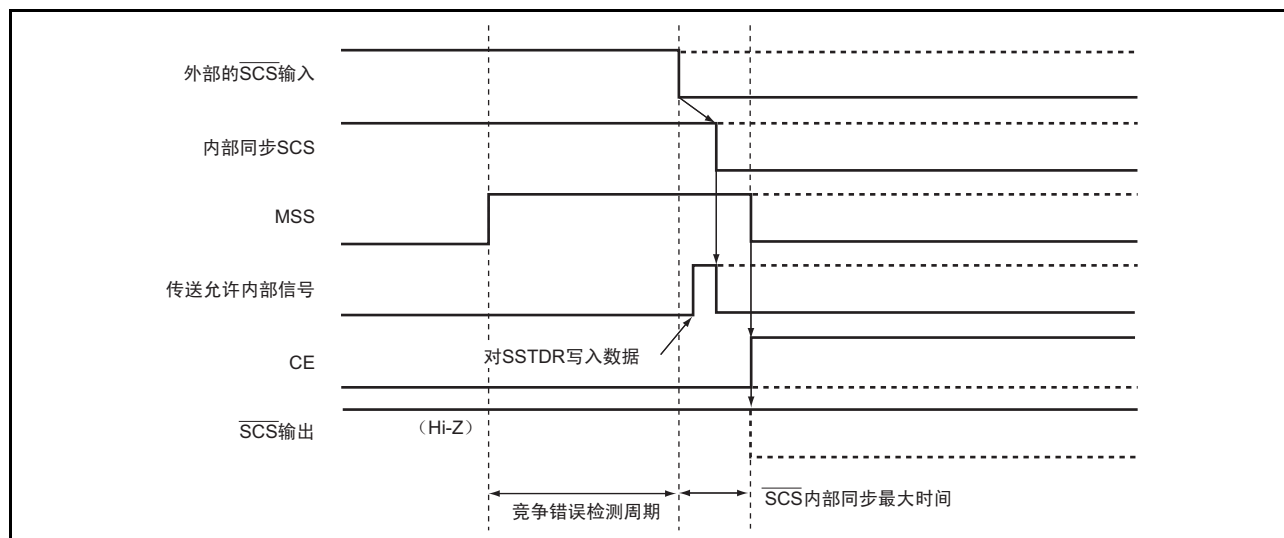


图 16.10 竞争错误检测时序 (传送前)

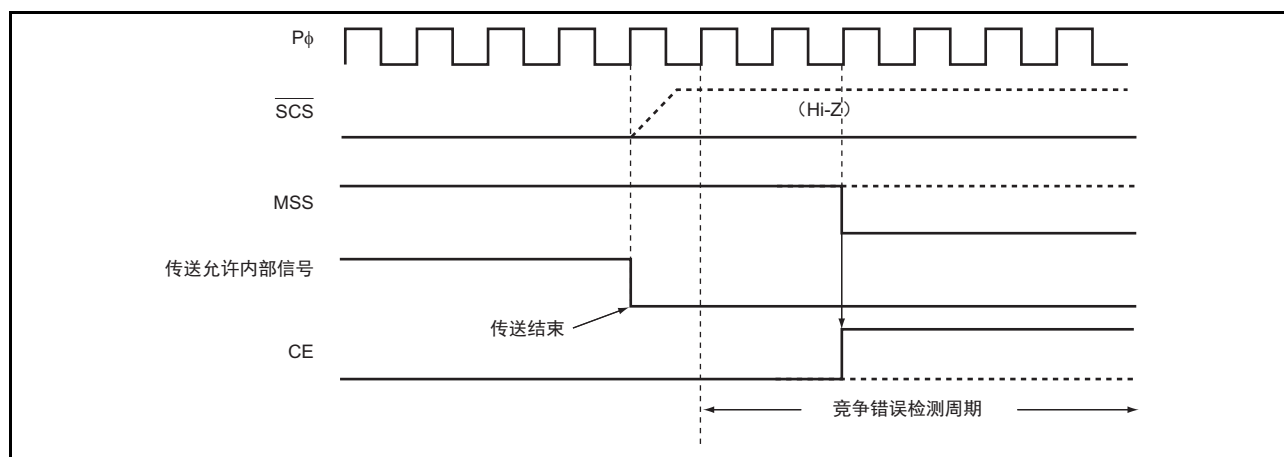


图 16.11 竞争错误检测时序 (传送结束后)

16.4.7 时钟同步通信模式

时钟同步通信模式使用时钟线 (SSCK)、数据输入线 (SSI) 及数据输出线 (SSO) 等 3 条总线执行数据通信。

(1) 时钟同步通信模式的初始设定

时钟同步通信模式的初始设定例如图 16.12 所示。发送 / 接收数据前，必须先将 SSER 的 TE 及 RE 清 0，再执行初始设定。

【注】 更改运行模式、通信格式时，必须先将 TE 及 RE 清 0。注意：如果 TE 清 0，则 TDRE 置 1，但 RE 清 0 时，仍保持 RDRF、ORER 的各标志及 SSRDR 的内容。

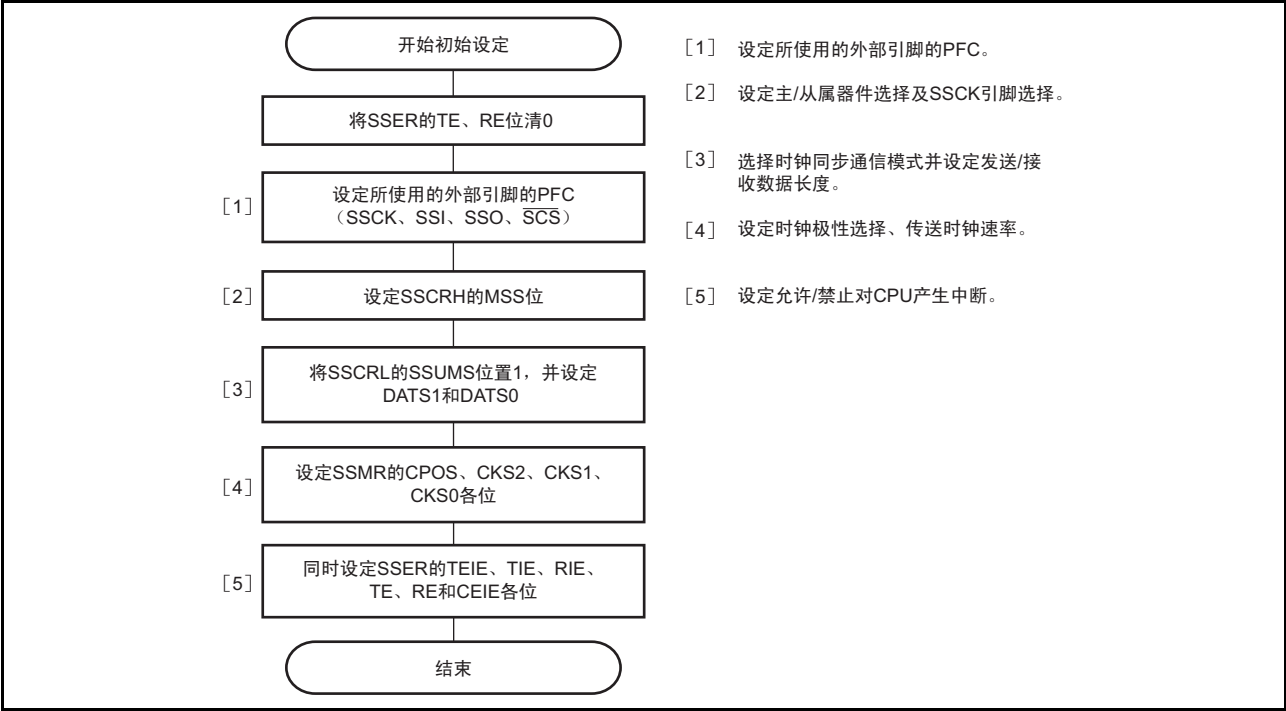


图 16.12 时钟同步通信模式的初始设定例

(2) 发送数据

发送时的运行例如图 16.13 所示，发送数据的流程图的例子如图 16.14 所示。发送数据时，时钟同步通信模式如下运行：

将 SSU 设定为主器件时，输出传送时钟与数据；设定为从属器件时，如果从 SSCK 引脚输入传送时钟，则与该传送时钟同步输出数据。

SSU 在 SSER 的 TE 置 1 后，如果对 SSTDR 写入发送数据，则 SSSR 的 TDRE 自动清 0，并从 SSTDR 向 SSTRSR 传送数据。之后，TDRE 置 1，并开始发送。此时，如果 SSER 的 TIE 置 1，则产生发送数据空时的 SSTXI 中断请求。

在 TDRE=0 的状态，如果传送完 1 帧的数据，则从 SSTDR 向 SSTRSR 传送数据，并开始发送下一帧。在 TDRE=1 的状态，如果发送第 8 位，则 SSSR 的 TEND 置 1，并保持状态。此时，如果 SSER 的 TEIE 置 1，则产生发送结束时的 SSTXI 中断请求。

在 SSSR 的 ORER 置 1 的状态，不可执行发送。发送前必须先确认 ORER 已清 0。

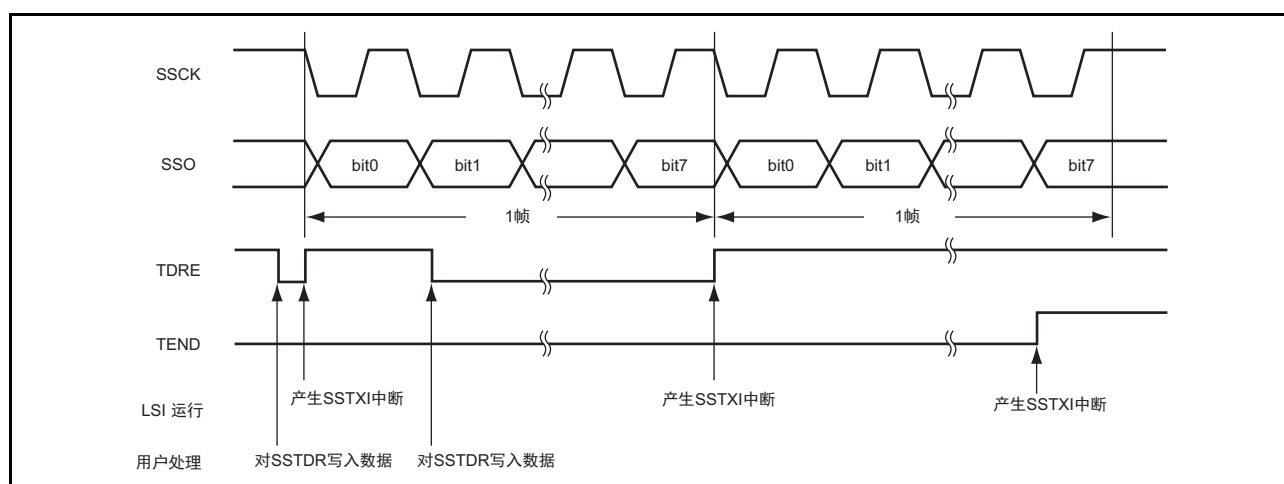


图 16.13 发送时的运行例（时钟同步通信模式）

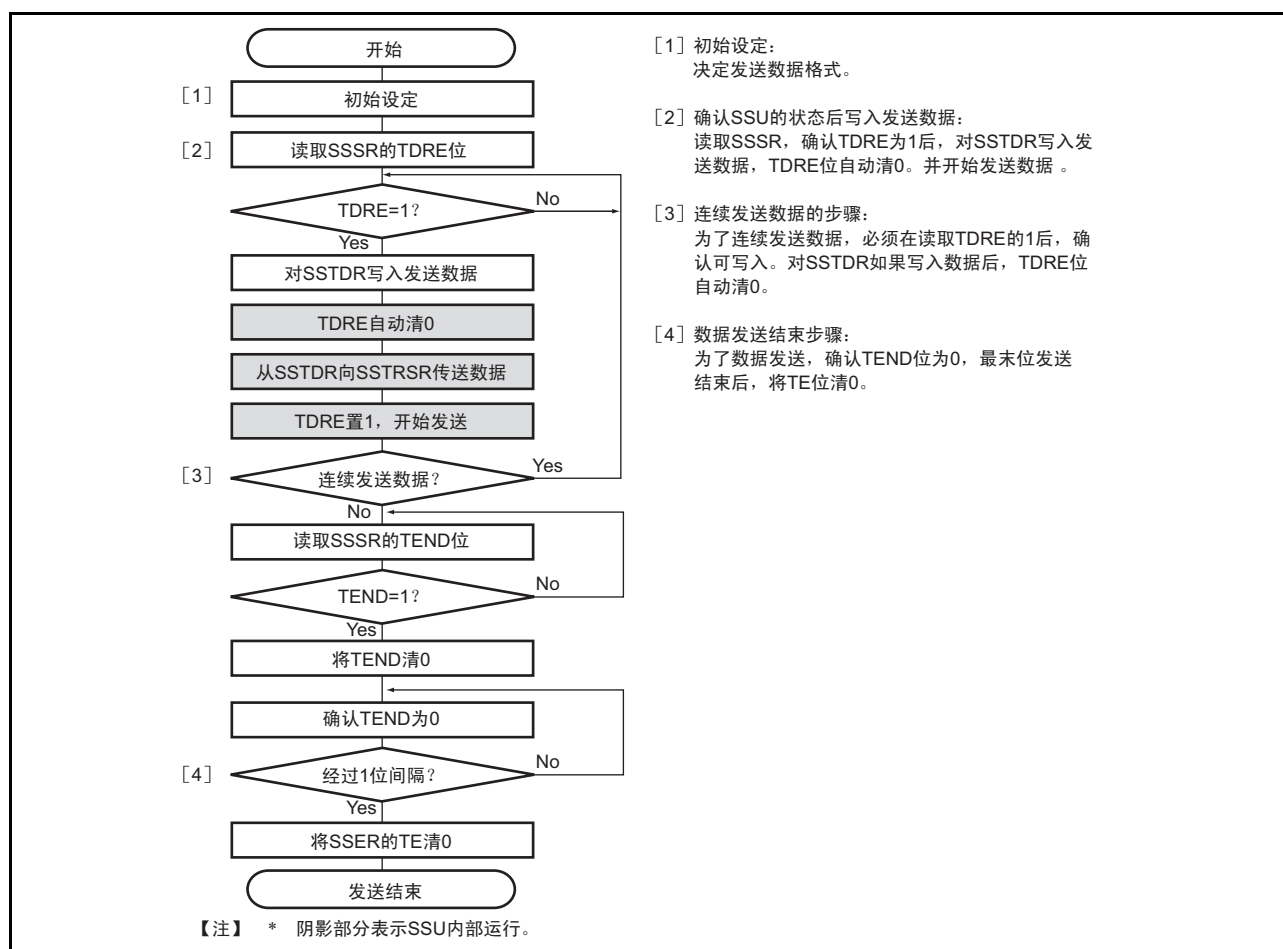


图 16.14 发送数据的流程图的例子（时钟同步通信模式）

(3) 接收数据

接收时的运行例如图 16.15 所示，接收数据的流程图的例子如图 16.16 所示。接收数据时，SSU 如下运行：

如果 SSER 的 RE 置 1，则 SSU 开始接收运行。

将 SSU 设定为主器件时，输出传送时钟，并输入接收数据；设定为从属器件时，如果从 SSCK 引脚输入传送时钟，则与该传送时钟同步输入接收数据。

接收 1 帧的数据后，将 SSSR 的 RDRF 置 1，并将接收数据保存至 SSRDR。此时，如果 SSER 的 RIE 置 1，则产生 SSRXI 中断请求；如果读取 SSRDR，则 RDRF 自动清 0。

在 RDRF=1 的状态，如果第 8 个时钟上升，则 SSSR 的 ORER 置 1，并产生溢出错误（SSERI），同时停止接收。在 ORER=1 的状态，不得接收数据，因此，重新开始接收时，必须将 ORER 清 0。

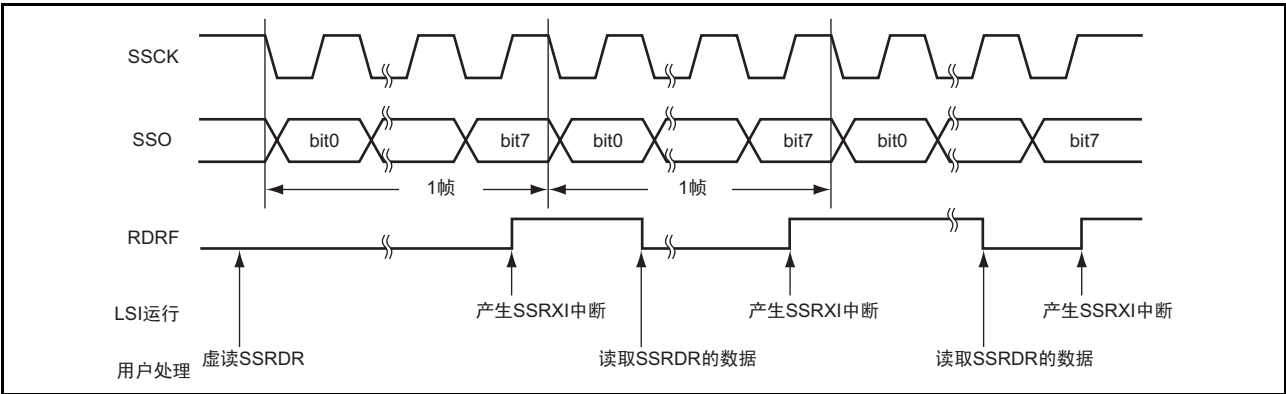


图 16.15 接收时的运行例（时钟同步通信模式）

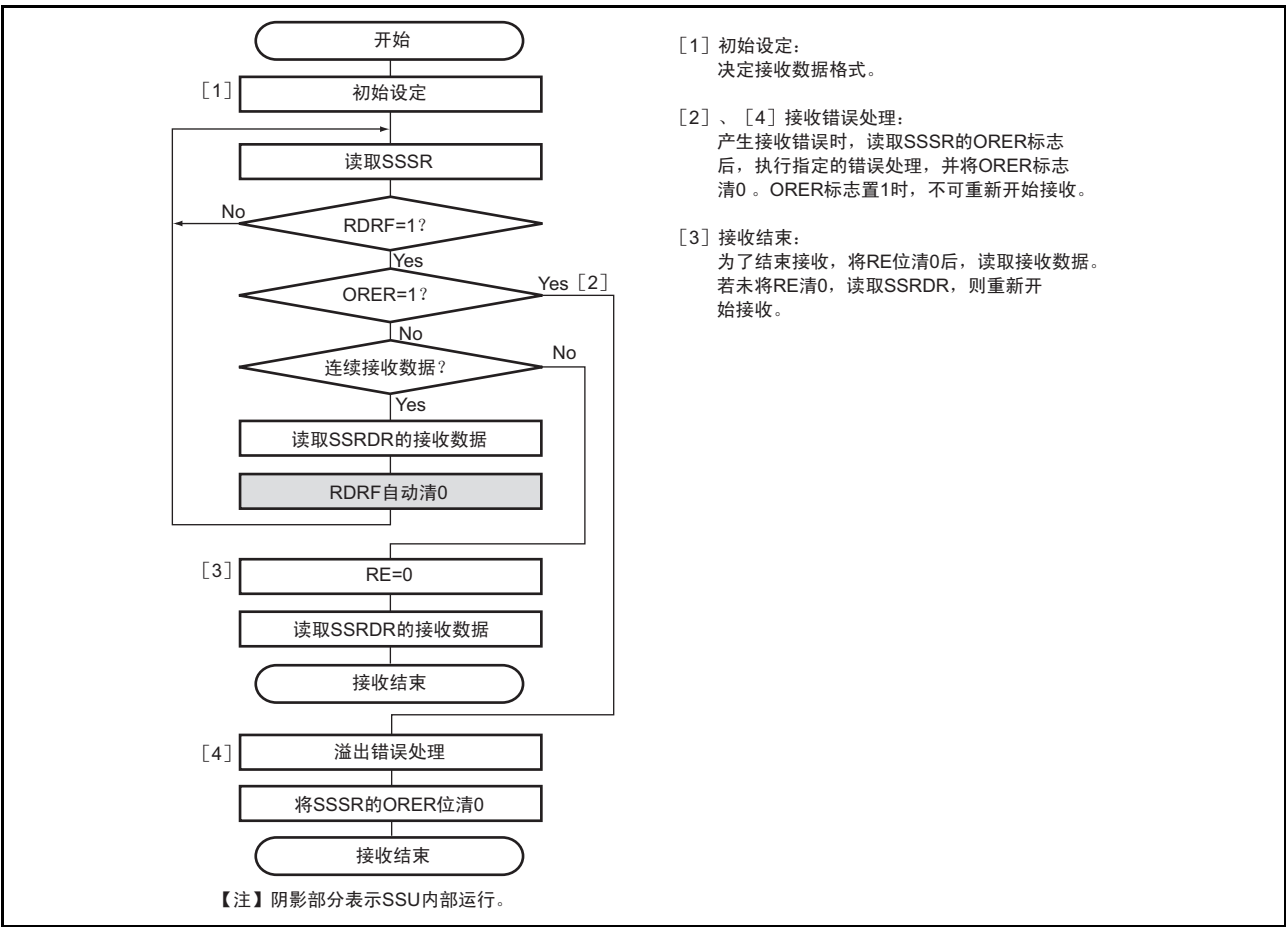


图 16.16 接收数据的流程图的例子（时钟同步通信模式）

(4) 发送 / 接收数据

发送 / 接收数据同时运行的流程图的例子如图 16.17 所示。发送 / 接收数据为发送数据与接收数据的混合运行。在 $TE=RE=1$ 的状态，通过对 SSTDR 写入发送数据开始发送 / 接收数据。在 $RDRF=1$ 的状态，如果第 8 个时钟上升，则 SSSR 的 ORER 置 1，并产生溢出错误 (SSERI)，同时停止发送 / 接收。在 $ORER=1$ 的状态，不得发送 / 接收数据，因此，重新开始发送 / 接收时，必须将 ORER 清 0。

从发送模式 ($TE=1$) 或接收模式 ($RE=1$) 转换为发送 / 接收模式 ($TE=RE=1$) 时，必须先将 TE、RE 清 0 一次。另外，确认 TEND、RDRF、ORER 清 0 后，将 TE 及 RE 置 1。

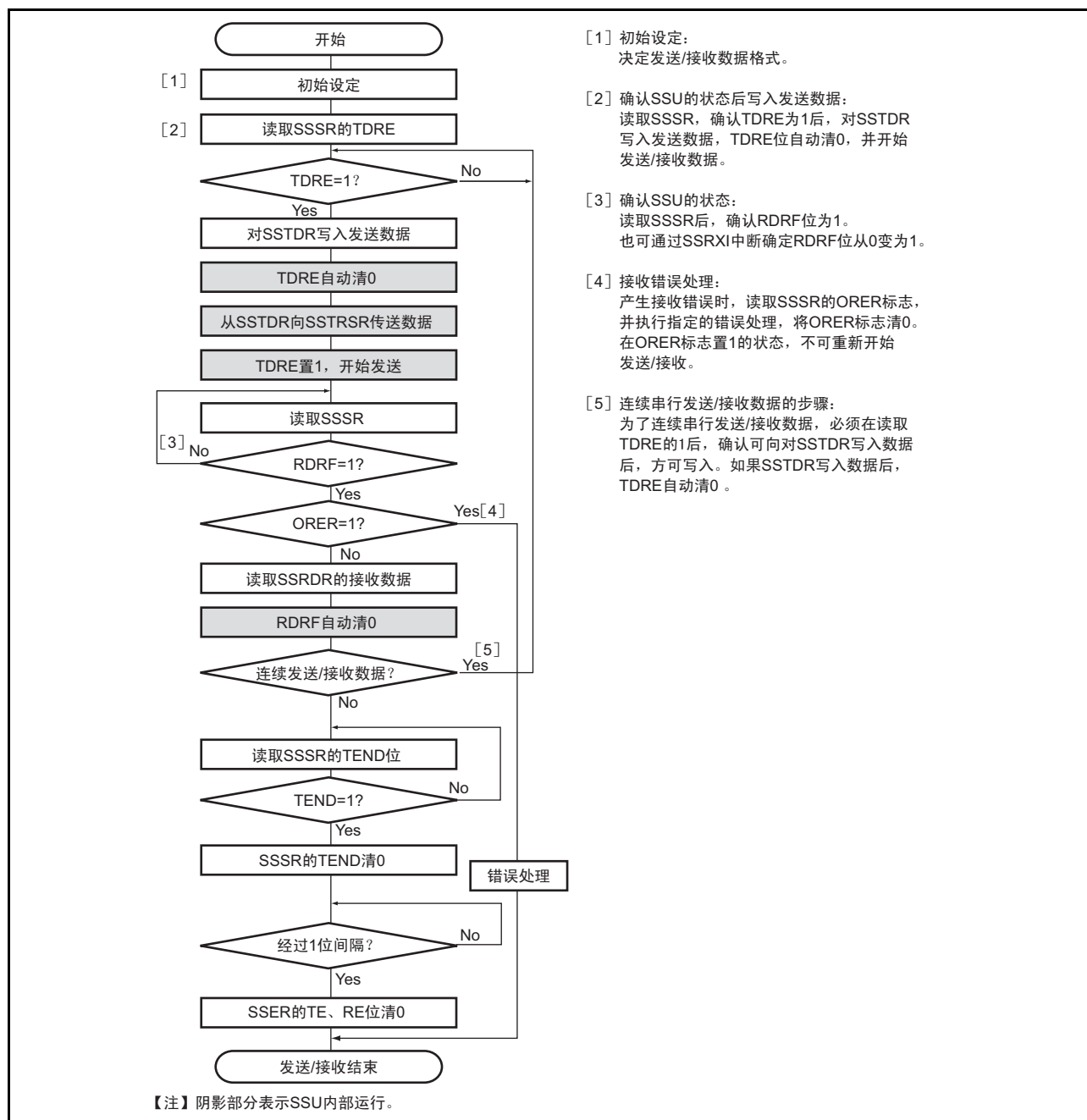


图 16.17 发送 / 接收数据同时运行的流程图的例子 (时钟同步通信模式)

16.5 SSU 的中断源与 DMAC

SSU 的中断请求有溢出错误、竞争错误、接收数据满、发送数据空及发送结束中断等。可通过接收数据满、发送数据空的中断请求启动 DMAC 来执行数据传送。

溢出错误、竞争错误的中断请求被分配至 SSERI 的向量地址；发送数据空、发送结束的中断请求被分配至 SSTXI 的向量地址，因此需根据标志判断中断源，中断源如表 16.8 所示。

表 16.8 所示的中断条件成立时，产生中断请求。必须通过 CPU 或 DMAC 的数据传送清除中断源。

表 16.8 SSU 中断源

名称	中断源	中断条件	DMAC 的启动
SSERI	溢出错误	$(RIE=1) \cdot (ORER=1) + (CEIE=1) \cdot (CE=1)$	—
	竞争错误		
SSRXI	接收数据满	$(RIE=1) \cdot (RDRF=1)$	○
SSTXI	发送数据空	$(TIE=1) \cdot (TDRE=1) + (TEIE=1) \cdot (TEND=1)$	○
	发送结束		

16.6 使用时的注意事项

16.6.1 模块待机模式的设定

SSU 可通过待机控制寄存器设定禁止 / 允许本模块的运行。待机控制寄存器为初始值时，SSU 停止运行。通过解除模块待机模式，可存取寄存器。详情参阅“第 28 章 低功耗模式”。

16.6.2 在 SSU 从属模式连续发送 / 接收时的注意事项

在 SSU 从属模式连续发送 / 接收时，必须将每一帧的 $\overline{SCS}$ 引脚设置为无效（高电平）。如果有超过 1 帧的 $\overline{SCS}$ 引脚被设置为有效（低电平），则无法正确发送 / 接收。

第 17 章 I²C 总线接口 3 (IIC3)

I²C 总线接口 3 以飞利浦公司提倡的 I²C 总线 (Inter IC Bus) 接口方式为依据, 配备了子集功能。但需注意, 控制 I²C 总线的寄存器结构与飞利浦公司的产品有一部分不同。

IIC3 有 4 个通道。

17.1 特点

- 可选择 I²C 总线格式或时钟同步串行格式
- 可连续发送/接收
移位寄存器、发送数据寄存器及接收数据寄存器各自独立, 因此可连续发送/接收

I²C 总线格式

- 在主模式自动生成开始条件、停止条件
- 接收时, 可选择应答的输出电平
- 发送时, 自动加载应答位
- 内置位同步功能
在主模式按位监控 SCL 的状态, 并自动取得同步。在未完成传送准备时, 将 SCL 设置为低电平使其处于待机状态。
- 中断源: 6 种
发送数据空 (包含从属地址匹配时)、发送结束、接收数据满 (包含从属地址匹配时)、仲裁失败、检测 NACK、检测停止条件。
- 可通过发送数据空中断与接收数据满中断启动直接存储器存取控制器 (DMAC), 执行数据传送。
- 可直接驱动总线
选择总线驱动功能时, SCL 引脚与 SDA 引脚为 NMOS 漏极开路输出

时钟同步串行格式

- 中断源: 4 种
发送数据空、发送结束、接收数据满、溢出错误
- 可通过发送数据空中断与接收数据满中断启动直接存储器存取控制器 (DMAC), 执行数据传送。

I²C 总线接口 3 框图如图 17.1 所示。

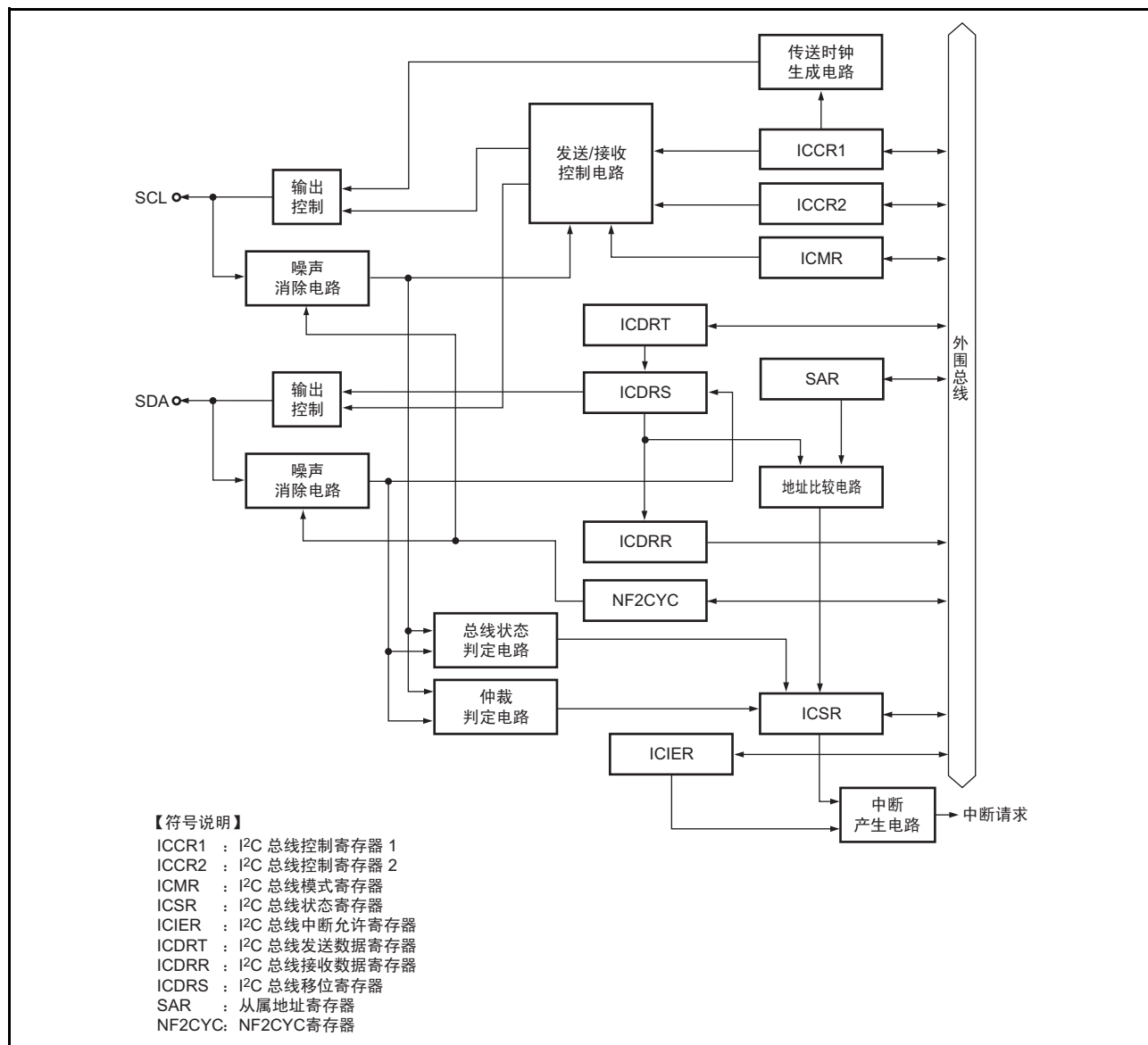


图 17.1 I²C 总线接口 3 框图

17.2 输入 / 输出引脚

用于 I²C 总线接口 3 的引脚的结构如表 17.1 所示。

表 17.1 引脚结构

名称	引脚名称	输入 / 输出	功 能
串行时钟引脚	SCL0 ~ SCL3	输入 / 输出	I ² C 串行时钟输入 / 输出引脚
串行数据引脚	SDA0 ~ SDA3	输入 / 输出	I ² C 串行数据输入 / 输出引脚

输入 / 输出引脚的外部电路连接例如图 17.2 所示。

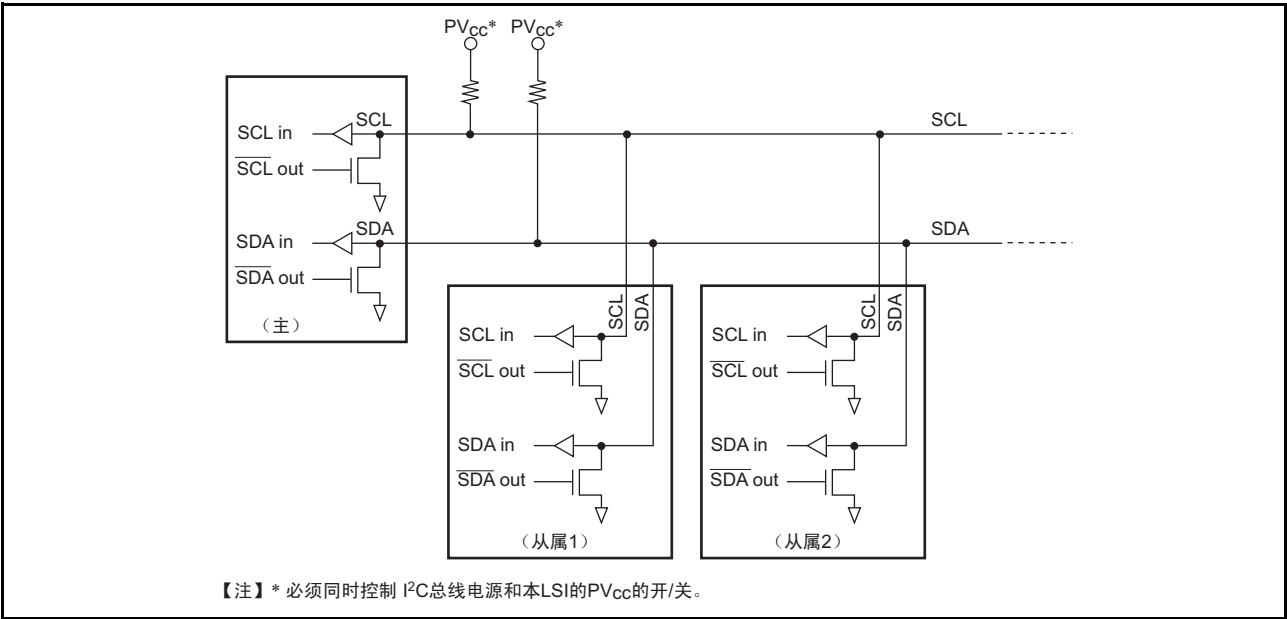


图 17.2 输入 / 输出引脚的外部电路连接例

17.3 寄存器说明

I²C 总线接口 3 有以下寄存器。

表 17.2 寄存器结构

通道	寄存器名称	简称	R/W	初始值	地址	存取长度
0	I ² C 总线控制寄存器 1	ICCR1_0	R/W	H'00	H'FFFEE000	8
	I ² C 总线控制寄存器 2	ICCR2_0	R/W	H'7D	H'FFFEE001	8
	I ² C 总线模式寄存器	ICMR_0	R/W	H'38	H'FFFEE002	8
	I ² C 总线中断允许寄存器	ICIER_0	R/W	H'00	H'FFFEE003	8
	I ² C 总线状态寄存器	ICSR_0	R/W	H'00	H'FFFEE004	8
	从属地址寄存器	SAR_0	R/W	H'00	H'FFFEE005	8
	I ² C 总线发送数据寄存器	ICDRT_0	R/W	H'FF	H'FFFEE006	8
	I ² C 总线接收数据寄存器	ICDRR_0	R/W	H'FF	H'FFFEE007	8
	NF2CYC 寄存器	NF2CYC_0	R/W	H'00	H'FFFEE008	8
1	I ² C 总线控制寄存器 1	ICCR1_1	R/W	H'00	H'FFFEE400	8
	I ² C 总线控制寄存器 2	ICCR2_1	R/W	H'7D	H'FFFEE401	8
	I ² C 总线模式寄存器	ICMR_1	R/W	H'38	H'FFFEE402	8
	I ² C 总线中断允许寄存器	ICIER_1	R/W	H'00	H'FFFEE403	8
	I ² C 总线状态寄存器	ICSR_1	R/W	H'00	H'FFFEE404	8
	从属地址寄存器	SAR_1	R/W	H'00	H'FFFEE405	8
	I ² C 总线发送数据寄存器	ICDRT_1	R/W	H'FF	H'FFFEE406	8
	I ² C 总线接收数据寄存器	ICDRR_1	R/W	H'FF	H'FFFEE407	8
	NF2CYC 寄存器	NF2CYC_1	R/W	H'00	H'FFFEE408	8
2	I ² C 总线控制寄存器 1	ICCR1_2	R/W	H'00	H'FFFEE800	8
	I ² C 总线控制寄存器 2	ICCR2_2	R/W	H'7D	H'FFFEE801	8
	I ² C 总线模式寄存器	ICMR_2	R/W	H'38	H'FFFEE802	8
	I ² C 总线中断允许寄存器	ICIER_2	R/W	H'00	H'FFFEE803	8
	I ² C 总线状态寄存器	ICSR_2	R/W	H'00	H'FFFEE804	8
	从属地址寄存器	SAR_2	R/W	H'00	H'FFFEE805	8
	I ² C 总线发送数据寄存器	ICDRT_2	R/W	H'FF	H'FFFEE806	8
	I ² C 总线接收数据寄存器	ICDRR_2	R/W	H'FF	H'FFFEE807	8
	NF2CYC 寄存器	NF2CYC_2	R/W	H'00	H'FFFEE808	8
3	I ² C 总线控制寄存器 1	ICCR1_3	R/W	H'00	H'FFFEEC00	8
	I ² C 总线控制寄存器 2	ICCR2_3	R/W	H'7D	H'FFFEEC01	8
	I ² C 总线模式寄存器	ICMR_3	R/W	H'38	H'FFFEEC02	8
	I ² C 总线中断允许寄存器	ICIER_3	R/W	H'00	H'FFFEEC03	8
	I ² C 总线状态寄存器	ICSR_3	R/W	H'00	H'FFFEEC04	8
	从属地址寄存器	SAR_3	R/W	H'00	H'FFFEEC05	8
	I ² C 总线发送数据寄存器	ICDRT_3	R/W	H'FF	H'FFFEEC06	8
	I ² C 总线接收数据寄存器	ICDRR_3	R/W	H'FF	H'FFFEEC07	8
	NF2CYC 寄存器	NF2CYC_3	R/W	H'00	H'FFFEEC08	8

17.3.1 I²C 总线控制寄存器 1 (ICCR1)

ICCR1 为 8 位可读取 / 写入的寄存器，用于选择 I²C 总线接口 3 的运行 / 停止、发送 / 接收控制、主模式 / 从属模式、发送 / 接收及主模式传送时钟频率。

位:	7	6	5	4	3	2	1	0
	ICE	RCVD	MST	TRS	CKS[3:0]			
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7	ICE	0	R/W	I ² C 总线接口 3 允许 0: 本模块为功能停止状态 1: 本模块为可传送状态 (SCL/SDA 为总线驱动状态)
6	RCVD	0	R/W	接收禁止 在 TRS=0 的状态, 读取 ICDRR 时, 设定继续 / 禁止下一次运行。 0: 继续下一次接收 1: 禁止下一次接收
5 4	MST TRS	0 0	R/W R/W	选择主模式 / 从属模式 选择发送 / 接收 在 I ² C 总线格式的主模式, 如果总线竞争失败, 则由硬件复位 MST、及 TRS, 并变为从属接收模式。必须在传送帧之间更改 TRS。 另外, 在从属接收模式, 如果开始条件后的 7 位地址与对 SAR 设定的从属地址匹配, 且第 8 位为 1, 则 TRS 自动置 1。在时钟同步串行格式的主接收模式产生溢出错误时, 则 MST 清 0, 并变为从属接收模式。 通过组合 MST 与 TRS, 变为以下运行模式。另外, 选择时钟同步串行格式时, 如果 MST=1, 则变为时钟输出。 00: 从属接收模式 01: 从属发送模式 10: 主接收模式 11: 主发送模式
3 ~ 0	CKS[3:0]	0000	R/W	选择传送时钟 主模式时, 必须对照所需传送速率 (参照表 17.3) 进行设定。

表 17.3 传送速率

bit3	bit2	bit1	bit0	时钟	传送速率 (kHz)				
CKS[3]	CKS[2]	CKS[1]	CKS[0]		P ϕ =16.7MHz	P ϕ =20.0MHz	P ϕ =25.0MHz	P ϕ =26.7MHz	P ϕ =33.3MHz
0	0	0	0	P ϕ /44	379	455	568	606	758
			1	P ϕ /52	321	385	481	513	641
		1	0	P ϕ /64	260	313	391	417	521
			1	P ϕ /72	231	278	347	370	463
	1	0	0	P ϕ /84	198	238	298	317	397
			1	P ϕ /92	181	217	272	290	362
		1	0	P ϕ /100	167	200	250	267	333
			1	P ϕ /108	154	185	231	247	309
1	0	0	0	P ϕ /176	94.7	114	142	152	189
			1	P ϕ /208	80.1	96.2	120	128	160
		1	0	P ϕ /256	65.1	78.1	97.7	104	130
			1	P ϕ /288	57.9	69.4	86.8	92.6	116
	1	0	0	P ϕ /336	49.6	59.5	74.4	79.4	99.2
			1	P ϕ /368	45.3	54.3	67.9	72.5	90.6
		1	0	P ϕ /400	41.7	50.0	62.5	66.7	83.3
			1	P ϕ /432	38.6	46.3	57.9	61.7	77.2

【注】 设定时必须满足外部规格。

17.3.2 I²C 总线控制寄存器 2 (ICCR2)

ICCR2 为 8 位可读/写入的寄存器，控制开始/停止条件的发行、SDA 引脚的操作、SCL 引脚的监控及 I²C 总线控制部的复位等。

位:	7	6	5	4	3	2	1	0
	BBSY	SCP	SDAO	SDAOP	SCLO	-	IICRST	-
初始值:	0	1	1	1	1	1	0	1
R/W:	R/W	R/W	R/W	R/W	R	R	R/W	R

位	位名称	初始值	R/W	说 明
7	BBSY	0	R/W	总线忙 具有表示 I ² C 总线占用/释放状态的标志功能及发行主模式开始/停止条件的功能。时钟同步串行格式时，本位的读取值总是为 0；I ² C 总线格式时，在 SCL= 高电平的状态，如果 SDA 从高电平变为低电平，则识别为已发行开始条件，并将此位置 1。在 SCL= 高电平的状态，如果 SDA 从低电平变为高电平，则识别为已发行停止条件，并将此位清 0。发行开始条件时，对 BBSY 写入 1、SCP 写入 0。再次发送开始条件时也同样执行此操作。发行停止条件时，则对 BBSY 写入 0、SCP 写入 0。
6	SCP	1	R/W	禁止发行开始/停止条件 SCP 位控制主模式下开始条件/停止条件的发行。发行开始条件时，对 BBSY 写入 1、SCP 写入 0。再次发送开始条件时也同样执行此操作。发行停止条件时，对 BBSY、SCP 写入 0。本位的读取值总是为 1。即使写入 1，也不保存数据。
5	SDAO	1	R/W	控制 SDA 输出值 更改 SDA 的输出电平时，SDAO 与 bit4 的 SDAOP 组合后使用。另外，不得在传送过程中操作本位。 0: 读取时，SDA 引脚输出为低电平 写入时，SDA 引脚输出变为低电平 1: 读取时，SDA 引脚输出为高电平 写入时，SDA 引脚输出变为 Hi-Z（由外部上拉电阻输出高电平）
4	SDAOP	1	R/W	SDAO 写入保护 通过改写 SDAOP 位，控制 SDA 引脚输出电平的变化。更改输出电平时，需对 SDAOP 及 SDAOP 写入 0，或对 SDAOP 写入 1、SDAOP 写入 0。本位的读取值总是为 1。
3	SCLO	1	R	SCL 输出电平 SCLO 监控 SCL 的输出电平。读取时，如果 SCLO 为 1，则 SCL 引脚输出为高电平；如果 SCLO 为 0，则 SCL 引脚输出为低电平。
2	—	1	R	保留位 读取值、写入值总是为 1。
1	IICRST	0	R/W	IIC 控制部复位 IICRST 复位 ICMR 寄存器的 BC[2:0] 位及 IIC3 的内部电路。在 I ² C 总线的运行过程中，因通信故障等原因而挂起时，如果 IICRST 位置 1，则可复位 ICMR 寄存器的 BC[2:0] 位及 IIC3 的内部电路。
0	—	1	R	保留位 读取值、写入值总是为 1。

17.3.3 I²C 总线模式寄存器 (ICMR)

ICMR 为 8 位可读取 / 写入寄存器，用于选择 MSB first/LSB first 及传送位数。
由 ICCR2 的 IICRST 将 BC[2:0] 初始化为 H'0。

位:	7	6	5	4	3	2	1	0
	MLS	-	-	-	BCWP	BC[2:0]		
初始值:	0	0	1	1	1	0	0	0
R/W:	R/W	R	R	R	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7	MLS	0	R/W	选择 MSB first/LSB first 0: MSB first 1: LSB first 用于 I ² C 总线格式时，必须设定为 0。
6	—	0	R	保留位 读取值、写入值总是为 0。
5、4	—	均为 1	R	保留位 读取值、写入值总是为 1。
3	BCWP	1	R/W	BC 写入保护 控制 BC[2:0] 的写入。改写 BC[2:0] 时，必须将本位设定为 0。另外，不得在时钟同步串行格式改写 BC[2:0]。 0: 写入时，设定 BC[2:0] 的值 1: 读取时，总是读取 1 写入时，BC[2:0] 的设定值无效
2 ~ 0	BC[2:0]	000	R/W	位计数器 指定下一次传送数据的位数。如果读取此位，则可获知剩余的传送位数。在 I ² C 总线格式，对数据加上应答 1 位并执行传送。必须在传送帧之间执行设定。另外，必须在 SCL 为低电平状态，设定除 B'000 之外的值。在包含应答的数据传送结束后，这些位自动返回至 B'000。同时，检测出停止条件后，自动变为 B'111。通过上电复位、软件待机模式、模块待机模式及将 ICCR2 的 IICRST 置 1，来清除本位。选择时钟同步串行格式时，不得改写本位。 I ² C 总线格式

17.3.4 I²C 总线中断允许寄存器 (ICIER)

ICIER 为 8 位可读取 / 写入的寄存器，用于允许各种中断源、选择应答的有效 / 无效、设定发送应答及确认接收应答。

位:	7	6	5	4	3	2	1	0
	TIE	TEIE	RIE	NAKIE	STIE	ACKE	ACKBR	ACKBT
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W

位	位名称	初始值	R/W	说 明
7	TIE	0	R/W	发送中断允许 ICSR 的 TDRE 置位时，允许 / 禁止发送数据空中断 (TXI)。 0: 禁止发送数据空中断请求 (TXI) 1: 允许发送数据空中断请求 (TXI)
6	TEIE	0	R/W	发送结束中断允许 在 ICSR 的 TDRE 为 1 的状态，且第 9 个时钟为上升时，TEIE 选择允许 / 禁止发送结束中断 (TEI)。另外，可通过将 TEND 或 TEIE 清 0，解除 TEI。 0: 禁止发送结束中断请求 (TEI) 1: 允许发送结束中断请求 (TEI)
5	RIE	0	R/W	接收中断允许 在接收数据从 ICDRS 传送至 ICDDR，且 ICSR 的 RDRF 置 1 时，RIE 选择允许 / 禁止接收数据满中断请求 (RXI) 及允许 / 禁止时钟同步格式时的溢出错误中断请求 (ERI)。另外，可通过将 RDRF 清 0 或 RIE 清 0，解除 RXI。 0: 禁止接收数据满中断请求 (RXI) 1: 允许接收数据满中断请求 (RXI)
4	NAKIE	0	R/W	NACK 接收中断允许 ICSR 的 NACKF 或 AL/OVE 置位时，NAKIE 选择允许 / 禁止 NACK 检测中断请求 (NAKI) 及允许 / 禁止时钟同步格式时的溢出错误 (ICSR 的 OVE 置位) 中断请求 (ERI)。另外，可通过将 NACKF、AL/OVE 清 0 或 NAKIE 清 0，解除 NAKI。 0: 禁止 NACK 接收中断请求 (NAKI) 1: 允许 NACK 接收中断请求 (NAKI)
3	STIE	0	R/W	停止条件检测中断允许 ICSR 的 STOP 置位时，STIE 选择允许 / 禁止停止条件检测中断请求 (STPI)。 0: 禁止停止条件检测中断请求 (STPI) 1: 允许停止条件检测中断请求 (STPI)
2	ACKE	0	R/W	选择应答位判断 0: 忽略接收应答的内容后执行连续传送 1: 接收应答位为 1 时，中止传送
1	ACKBR	0	R	接收应答 发送模式时，本位保存从接收器件接收的应答位的内容。写入无效。 另外，通过将 ICCR2 的 BBSY 置 1，清除本位。 0: 接收应答 = 0 1: 接收应答 = 1
0	ACKBT	0	R/W	发送应答 接收模式时，在应答时序设定发送位。 0: 在应答时序发送 0 1: 在应答时序发送 1

17.3.5 I²C 总线状态寄存器 (ICSR)

ICSR 为 8 位可读取 / 写入的寄存器，用于确认各中断请求标志及状态。

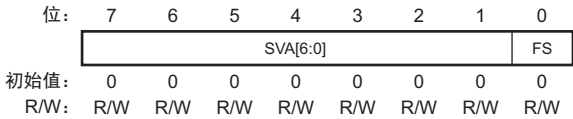
位:	7	6	5	4	3	2	1	0
	TDRE	TEND	RDRF	NACKF	STOP	AL/OVE	AAS	ADZ
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7	TDRE	0	R/W	发送数据空 [清除条件] • 读取 1 后写入 0 时 • 对 ICDRT 写入数据时 [置位条件] • 从 ICDRT 向 ICDRS 传送数据，且 ICDRT 为空时 • 置位 TRS 时 • 发行开始条件（含重新发送）时 • 从属模式下从接收模式变为发送模式时
6	TEND	0	R/W	发送结束 [清除条件] • 读取 1 后写入 0 时 • 对 ICDRT 写入数据时 [置位条件] • I ² C 总线格式时，在 TDRE 为 1 的状态，SCL 的第 9 个时钟上升时 • 在时钟同步串行格式，传送发送帧的最末位时
5	RDRF	0	R/W	接收数据满 [清除条件] • 读取 1 后写入 0 时 • 读取 ICDRR 时 [置位条件] • 从 ICDRS 向 ICDRR 传送接收数据时
4	NACKF	0	R/W	无应答检测标志 [清除条件] • 读取 1 后写入 0 时 [置位条件] • 在 ICIER 的 ACKE=1 的状态，发送时不存在来自接收器件的应答时

位	位名称	初始值	R/W	说 明
3	STOP	0	R/W	停止条件检测标志 [清除条件] • 读取 1 后写入 0 时 [置位条件] • 在结束帧传送后, 检测出停止条件时
2	AL/OVE	0	R/W	仲裁失败标志 / 溢出错误标志 在 I ² C 总线格式, AL/OVE 表示在主模式总线竞争失败; 在时钟同步串行格式, AL/OVE 表示在 RDRF=1 的状态接收到最末位。 表示多个主器件试图同时占用总线时, I ² C 总线接口 3 监控 SDA, 如果与自身发送的数据不同, 则 AL 标志置 1, 并由其它主器件占用总线。 [清除条件] • 读取 1 后写入 0 时 [置位条件] • 在主发送模式, 内部 SDA 与 SDA 引脚的电平在 SCL 的上升沿不匹配时 • 在主模式, 检测出开始条件, SDA 引脚为高电平时 • 在时钟同步串行格式, RDRF=1 的状态下接收到最末位时
1	AAS	0	R/W	从属地址识别标志 在从属接收模式, 开始条件后的第一帧与 SAR 的 SVA6 ~ SVA0 匹配时, 此标志置位。 [清除条件] • 读取 1 后写入 0 时 [置位条件] • 在从属接收模式检测出从属地址时 • 在从属接收模式检测出一般调用地址时
0	ADZ	0	R/W	一般调用地址识别标志 在 I ² C 总线格式的从属接收模式时有效 [清除条件] • 读取 1 后写入 0 时 [置位条件] • 在从属接收模式且检测出一般调用地址时

17.3.6 从属地址寄存器 (SAR)

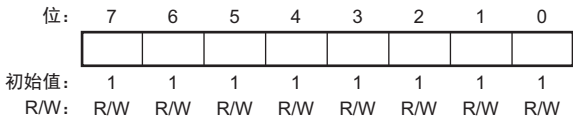
SAR 为 8 位可读取 / 写入的寄存器，用于选择格式及设定从属地址。在 I²C 总线格式的从属模式，开始条件后发送过来的第 1 帧的高 7 位与 SAR 的高 7 位匹配时，SAR 作为从属器件运行。



位	位名称	初始值	R/W	说 明
7 ~ 1	SVA[6:0]	0000000	R/W	从属地址 设定与连接至 I ² C 总线的其他从属地址不同的唯一地址。
0	FS	0	R/W	选择格式 0: 选择 I ² C 总线格式 1: 选择时钟同步串行格式

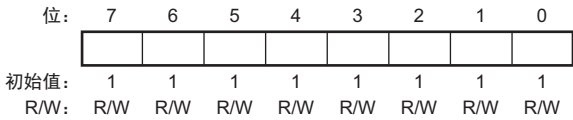
17.3.7 I²C 总线发送数据寄存器 (ICDRT)

ICDRT 为 8 位保存发送数据的可读取 / 写入的寄存器，检测出移位寄存器 (ICDRS) 为空时，将写入 ICDRT 的发送数据传送至 ICDRS，开始发送数据。发送 ICDRS 数据的过程中，如果将下次发送的数据预先写入 ICDRT，则可连续发送。



17.3.8 I²C 总线接收数据寄存器 (ICDRR)

ICDRR 为 8 位保存接收数据的寄存器。接受完 1 字节的数据时，将接收到的数据从 ICDRS 传送至 ICDRR，并设置为可接收下一个数据。ICDRR 为接收专用寄存器，不可由 CPU 写入。



17.3.9 I²C 总线移位寄存器 (ICDRS)

ICDRS 为发送 / 接收数据的寄存器。发送时，将发送数据从 ICDRT 传送至 ICDRS，并从 SDA 引脚发送。接收时，接受完 1 字节的数据时，将数据从 ICDRS 传送至 ICDRR。另外，不可直接由 CPU 读取本寄存器。

位:	7	6	5	4	3	2	1	0
初始值:	-	-	-	-	-	-	-	-
R/W:	-	-	-	-	-	-	-	-

17.3.10 NF2CYC 寄存器 (NF2CYC)

NF2CYC 为 8 位可读取 / 写入的寄存器，可选择 SCL 引脚、SDA 引脚的噪声消除范围。详情参阅“17.4.7 噪声消除电路”。

位:	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	PRS	NF2 CYC
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W

位	位名称	初始值	R/W	说 明
7 ~ 2	—	均为 0	R	保留位 读取值、写入值总是为 0。
1	PRS	0	R/W	选择脉宽比 指定 SCL 的高电平脉宽与低电平脉宽的比率。但 ICCR1 的 CKS[3:0]=H'7 或 H'F 时，不得设定 PRS=1。 0: High: Low=0.5 : 0.5 1: High: Low≈0.4 : 0.6
0	NF2CYC	0	R/W	选择噪声消除范围 0: 可消除 1 个外围时钟周期内的噪声 1: 可消除 2 个外围时钟周期内的噪声

17.4 运行说明

I²C 总线接口 3 通过设定 SAR 的 FS，能以 I²C 总线格式与时钟同步串行格式执行通信。

17.4.1 I²C 总线格式

I²C 总线格式如图 17.3 所示。I²C 总线时序如图 17.4 所示。开始条件后的第 1 帧必须由 8 位构成。

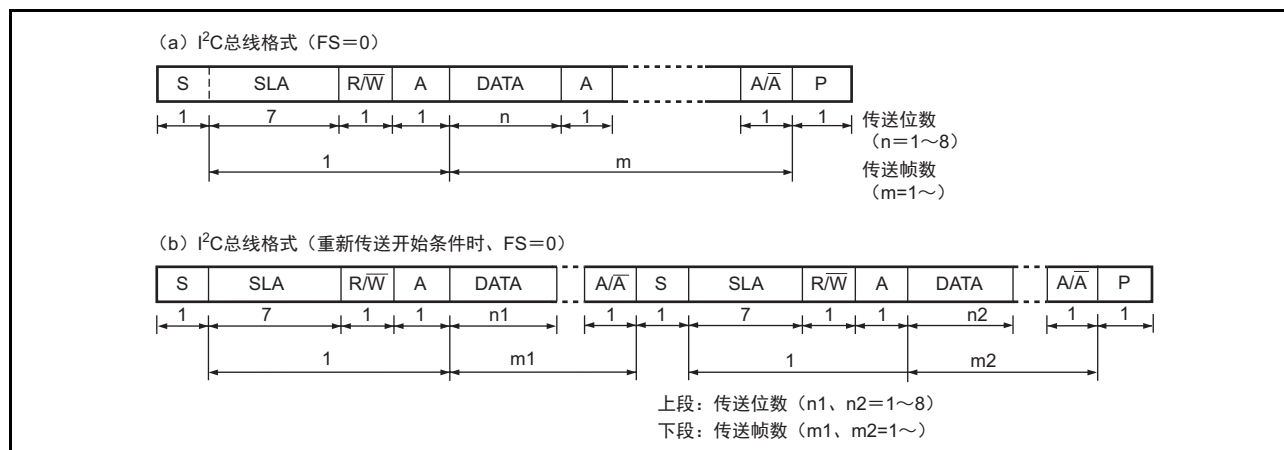


图 17.3 I²C 总线格式

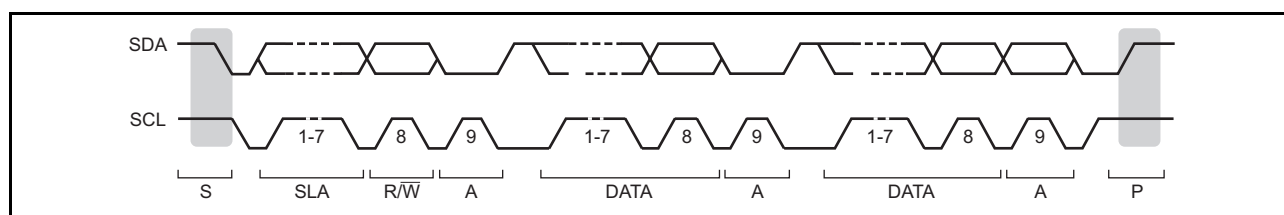


图 17.4 I²C 总线时序

【符号说明】

- S : 开始条件。在 SCL= 高电平的状态，主器件将 SDA 从高电平变为低电平。
- SLA : 从属地址
- R/W : 表示发送 / 接收方向。为 1 时，将数据从从属器件发送至主器件；为 0 时，将数据从主器件发送至从属器件。
- A : 应答。接收器件将 SDA 设置为低电平。
- DATA: 发送 / 接收数据
- P : 停止条件。在 SCL= 高电平的状态，主器件将 SDA 从低电平变为高电平。

17.4.2 主发送运行

在主发送模式，主器件输出发送时钟与发送数据，从属器件返回应答。有关主发送模式的运行时序，参照图 17.5 与图 17.6。主发送模式的发送步骤与运行如下所示：

1. 将 ICCR1 的 ICE 位设定为 1，并设定 ICCR1 的 CKS[3:0] 等（初始设定）。
2. 读取 ICCR2 的 BBSY 标志，并在确认总线处于释放状态后，将 ICCR1 的 MST、TRS 设定为主发送模式。然后，写入 BBSY=1、SCP=0（发行开始条件）。由此生成开始条件。
3. 在识别出 ICSR 的 TDRE 置位后，对 ICDRT 写入发送数据（第 1 字节为表示从属地址与 R/W 的数据）。此时，TDRE 自动清 0，从 ICDRT 向 ICDRS 传送数据，并再次置位 TDRE。
4. 在 TDRE 被置位的状态，结束 1 字节发送，且在发送时钟的第 9 个时钟的上升沿，ICSR 的 TEND 置位。读取 ICIER 的 ACKBR，并识别出选择的从属器件后，对 ICDRT 写入第 2 字节的数据。ACKBR 为 1 时无法识别从属器件，因此发行停止条件。通过写入 BBSY=0、SCP=0，发行停止条件。另外，在准备好数据或发行停止条件之前，SCL 固定为低电平。
5. 每当 TDRE 置位时，就对 ICDRT 写入第 2 字节及其后的发送数据。
6. 如果对 ICDRT 写入发送的字节数，则在 TDRE 被置位的状态，等待直到 TEND（发送完最后的字节）置位。或在 ICIER 的 ACKE 置位的状态，等待来自接收器件的 NACK（ICSR 的 NACKF=1）。然后，发行停止条件，并清除 TEND 或 NACKF。
7. 如果 ICSR 的 STOP 置位，则返回从属接收模式。

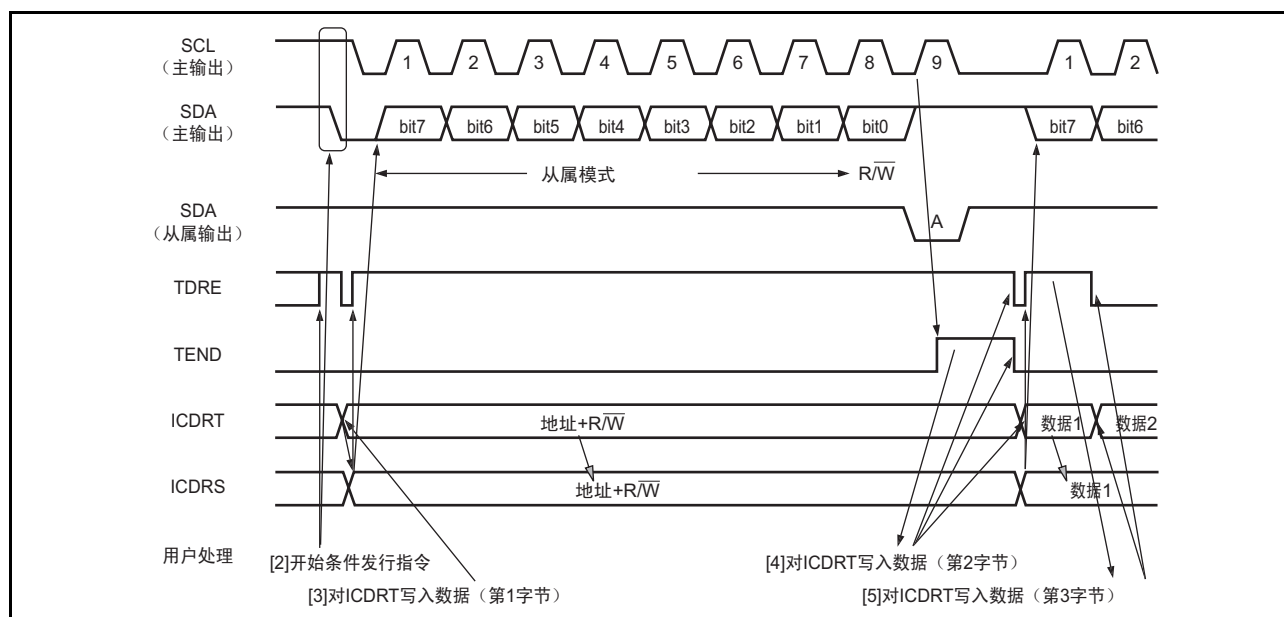


图 17.5 主发送模式的运行时序（1）

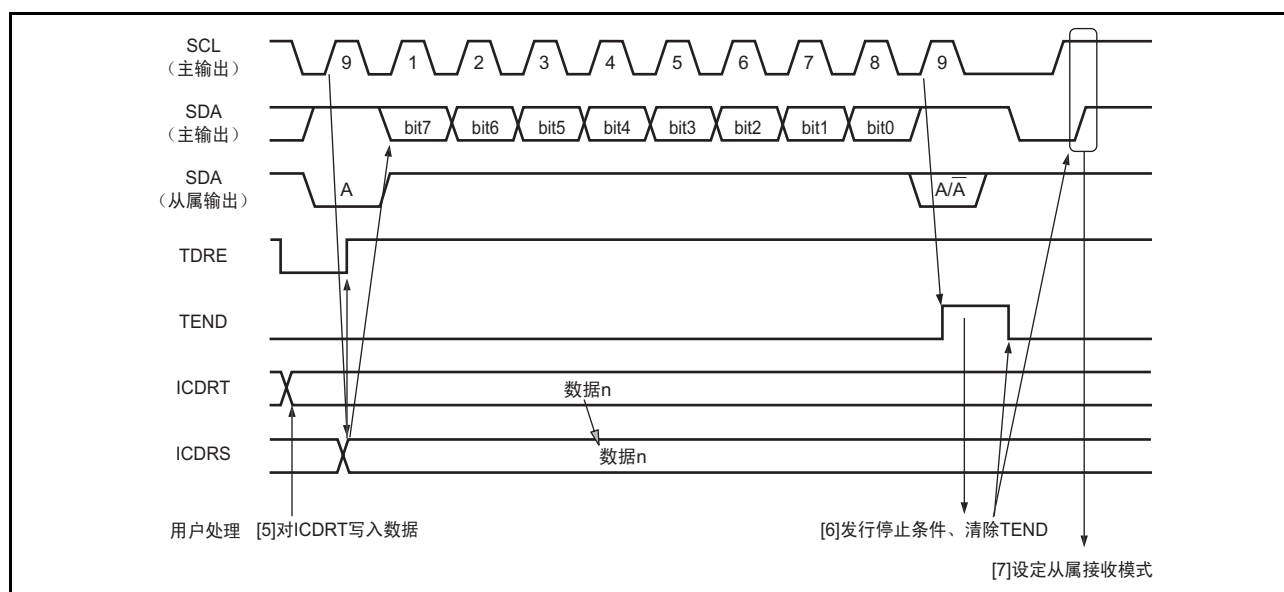


图 17.6 主发送模式的运行时序 (2)

17.4.3 主接收运行

在主接收模式，主器件输出接收时钟，接收来自从属器件的数据，并返回应答。有关主接收模式的运行顺序，参照图 17.7 与图 17.8。

主接收模式的接收步骤与运行如下所示：

1. 清除ICSR的TEND后，清除ICCR1的TRS，并从主发送模式转换至主接收模式。然后清除TDRE。
2. 如果虚读ICDRR，则开始*接收，与内部时钟同步，输出接收时钟并接收数据。主器件在第9个接收时钟向SDA输出在ICIER的ACKBT设定的电平。
3. 接收完1帧的数据，在第9个接收时钟的上升沿，置位ICSR的RDRF。此时，可通过读取ICDRR，读取已接收的数据，同时清除RDRF。
4. 每当RDRF置位时，可通过读取ICDRR连续接收数据。另外，由于其他处理，在RDRF置位的状态，延迟读取ICDRR且第8个时钟下降时，在读取ICDRR之前，将SCL固定为低电平。
5. 如果下一个接收数据为最后的帧，则在读取ICDRR之前，置位ICCR1的RCVD。由此，在下次接收后，进入可发行停止条件的状态。
6. 如果在第9个接收时钟的上升沿RDRF置位，则发行停止条件。
7. 如果ICSR的STOP置位，则读取ICDRR，然后清除RCVD。
8. 返回从属接收模式。

【注】 * 只想接收 1 字节时，必须在 ICCR1 的 RCVD 置位后，虚读 ICDRR。

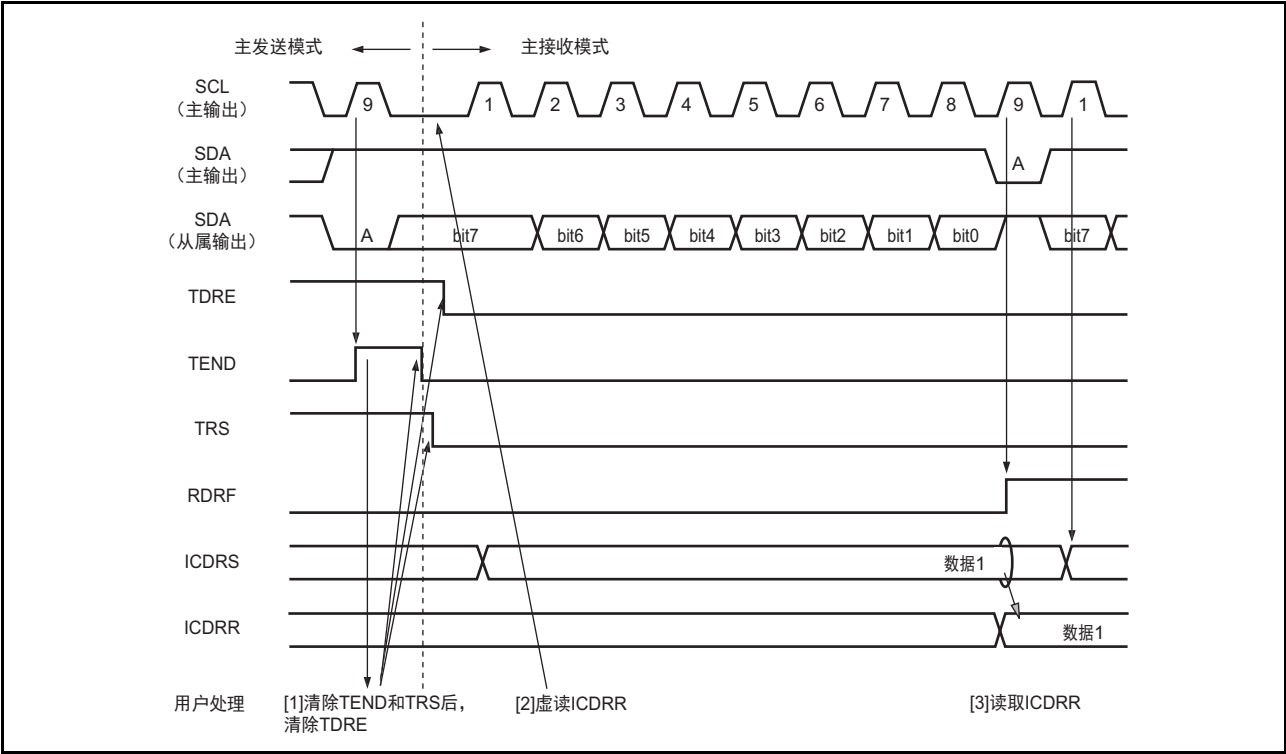


图 17.7 主接收模式的运行时序 (1)

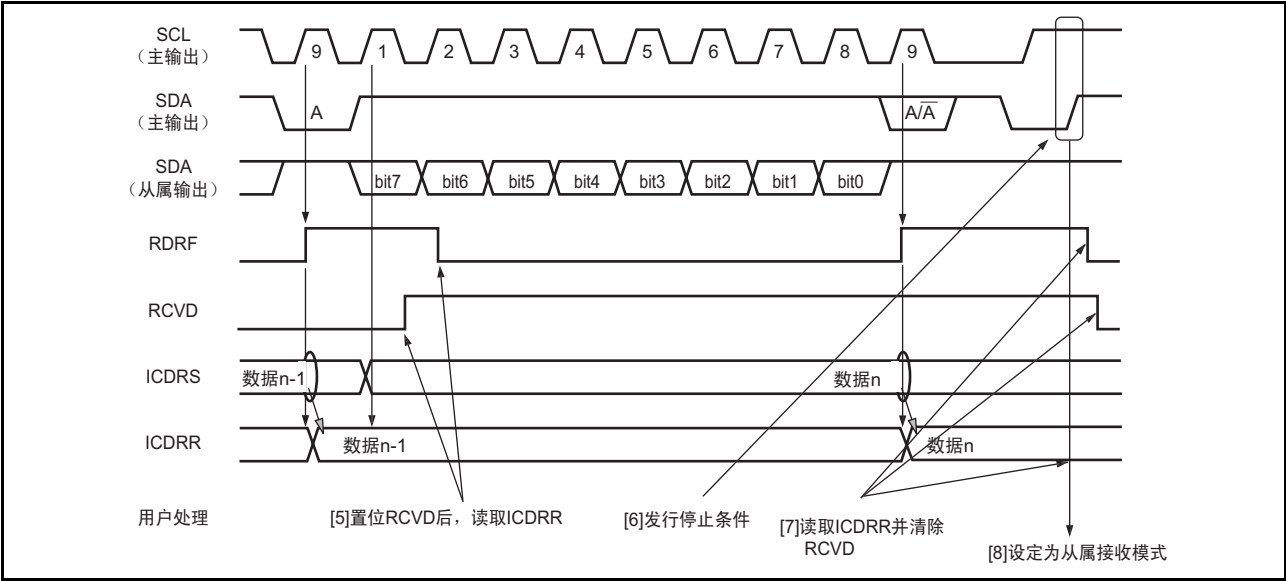


图 17.8 主接收模式的运行时序 (2)

17.4.4 从属发送运行

在从属发送模式，从属器件输出发送数据，主器件输出接收时钟，并返回应答。有关从属发送模式的运行时序，参照图 17.9 与图 17.10。

从属发送模式的发送步骤与运行如下所示：

1. 将ICCR1的ICE位置1。设定ICCR1的CKS[3:0]等（初始设定）。将ICCR1的MST、TRS设置为从属接收模式，并等待直到从属地址匹配。
2. 检测出开始条件后的第1帧且从属地址匹配时，从属器件在第9个时钟的上升沿向SDA输出在ICIER的ACKBT设定的电平。此时，第8位的数据（R/W）为1时，ICCR1的TRS与ICSR的TDRE置位，并自动转换为从属发送模式。每当TDRE置位时，如果对ICDRT写入发送数据，则可连续发送。
3. 对ICDRT写入最后的发送数据后，如果TDRE置位，则在TDRE=1的状态，等待直到ICSR的TEND置位。如果TEND置位，则清除TEND。
4. 为了结束处理清除TRS，并虚读ICDRR。由此来释放SCL。
5. 清除TDRE。

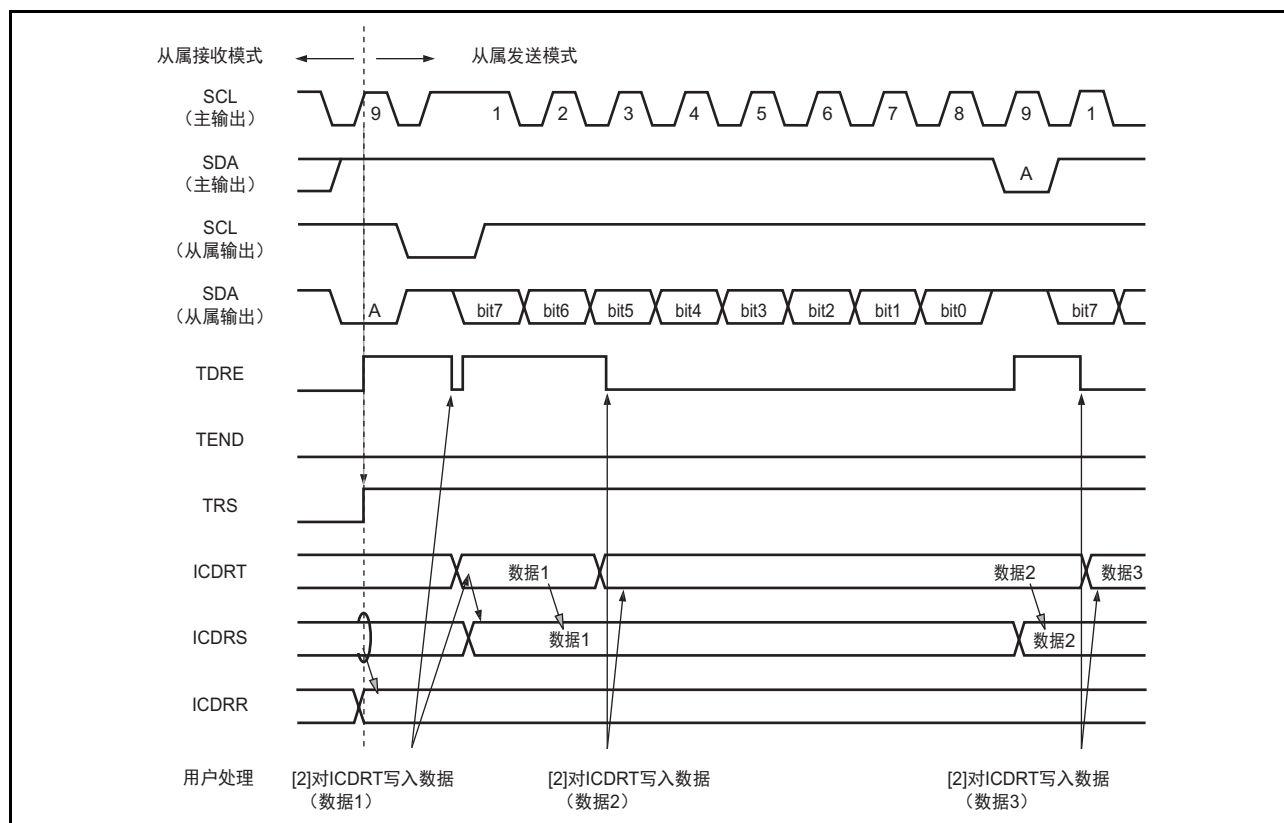


图 17.9 从属发送模式的运行时序 (1)

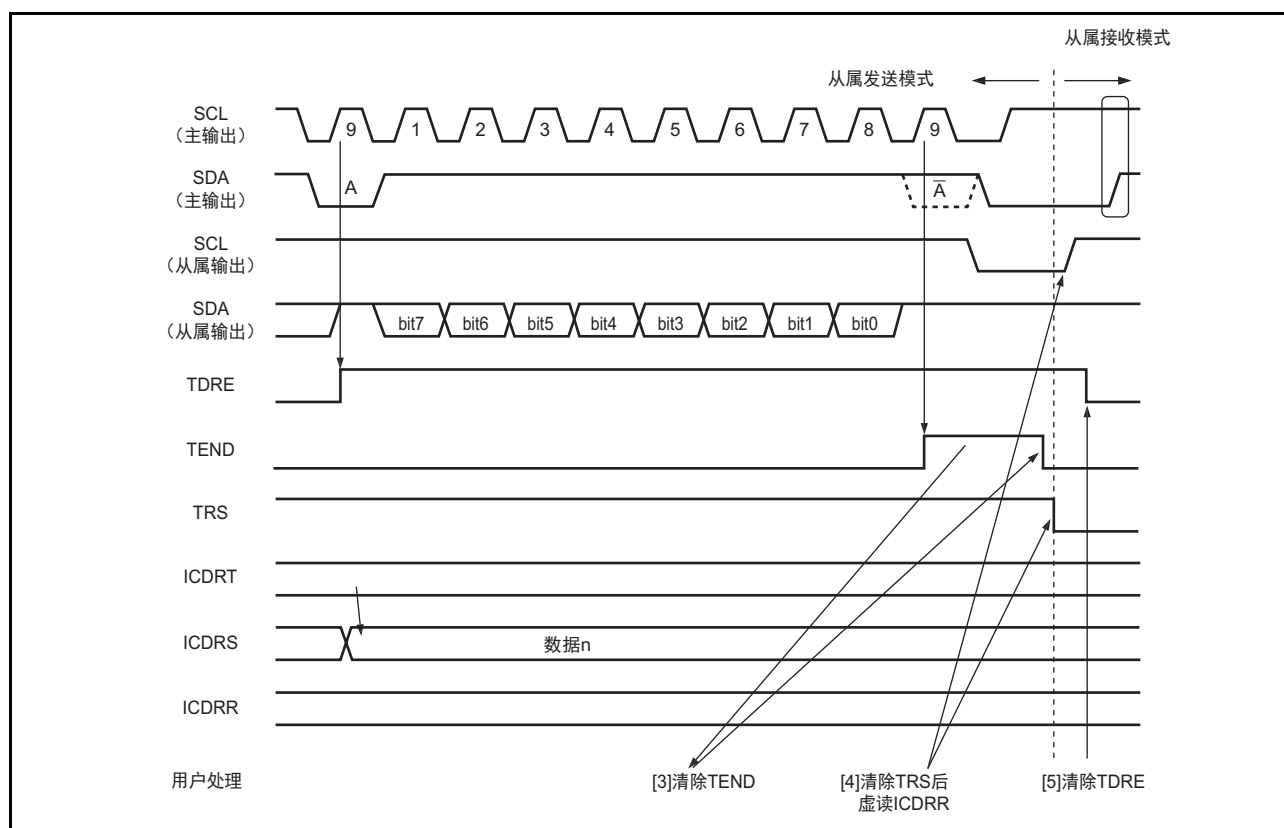


图 17.10 从属发送模式的运行时序 (2)

17.4.5 从属接收运行

在从属接收模式，主器件输出发送时钟及发送数据，且从属器件返回应答。有关从属运行模式的时序，参照图 17.11 与图 17.12。从属接收模式的接收步骤与运行如下所示：

1. 将ICCR1的ICE位置1。另外，设定ICCR1的CKS[3:0]等（初始设定）。将ICCR1的MST、TRS设置为从属接收模式，并等待直到从属地址匹配。
2. 检测出开始条件后的第一帧且从属地址匹配时，从属器件在第9个时钟的上升沿，对SDA输出在ICIER的ACKBT设定的电平。同时，因为ICSR的RDRF置位，所以虚读ICDRR（读取数据表示从属地址+R/W，因此不需要）。
3. 每当置位RDRF，则读取ICDRR。在RDRF置位的状态，如果第8个时钟下降，则在读取ICDRR之前，将SCL固定成低电平。应答（读取ICDRR前返回主器件的应答）设定的更改反映在下一个传送帧。
4. 同样通过读取ICDRR来读取最后字节。

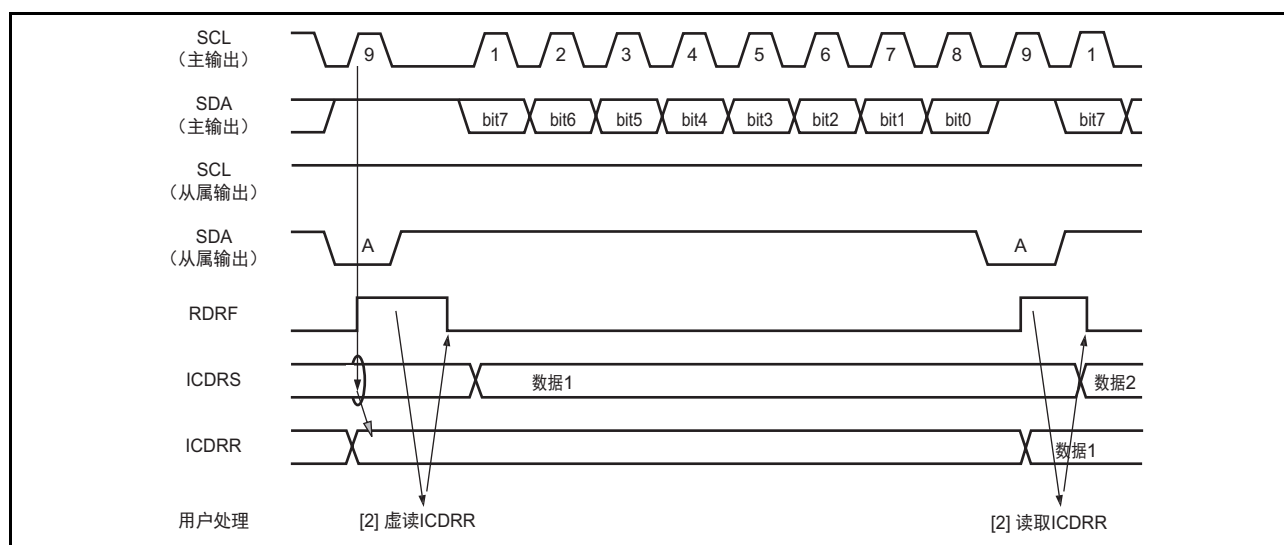


图 17.11 从属接收模式的运行时序 (1)

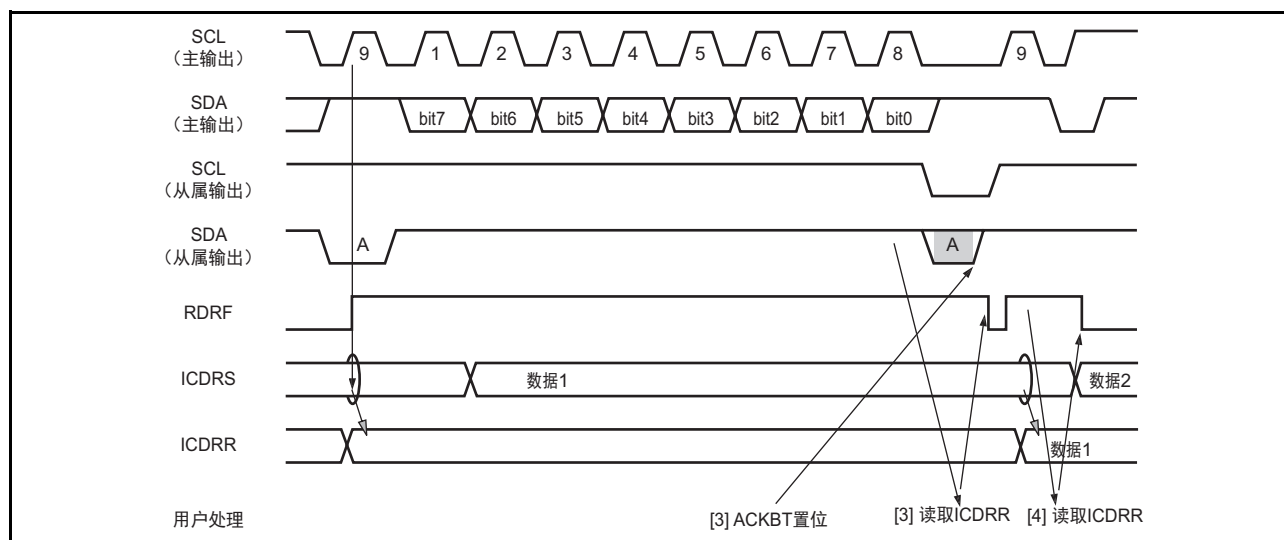


图 17.12 从属接收模式的运行时序 (2)

17.4.6 时钟同步串行格式

通过将 SAR 的 FS 置 1，I²C 总线接口 3 能够以时钟同步串行格式运行。ICCR1 的 MST=1 时，从 SCL 输出传送时钟；MST=0 时，则输入外部时钟。

(1) 数据传送格式

时钟同步串行传送格式如图 17.13 所示。

从 SCL 时钟的一个下降沿开始到下一个下降沿为止输出传送数据，保证 SCL 时钟上升沿的数据。通过 ICMR 的 MLS，可选择数据的传送顺序为 MSB first 或 LSB first。另外，在传送等待期间，可通过 ICCR2 的 SDAO 更改 SDA 的输出电平。

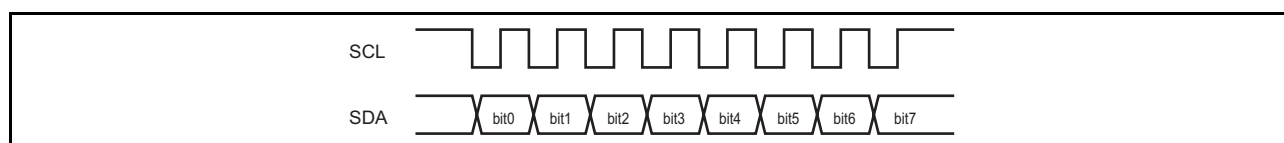


图 17.13 时钟同步串行格式的传送格式

(2) 发送运行

在发送模式，与传送时钟的下降同步，从 SDA 输出发送数据。ICCR1 的 MST=1 时，输出传送时钟；MST=0 时，输入传送时钟。有关发送模式的运行时序，参照图 17.14。发送模式的步骤和运行如下所示：

1. 将 ICCR1 的 ICE 位置 1。另外，设定 ICCR1 的 MST、CKS[3:0] 等（初始设定）。
2. 设定 ICCR1 的 TRS，并设置为发送模式，由此 ICSR 的 TDRE 置位。
3. 识别出 TDRE 置位后，对 ICDRT 写入发送数据。从 ICDRT 向 ICDRS 传送数据，并自动置位 TDRE。每当 TDRE 置位时，如果对 ICDRT 写入数据，则可连续发送。从发送模式转换为接收模式时，必须在 TDRE 置位的状态清除 TRS。

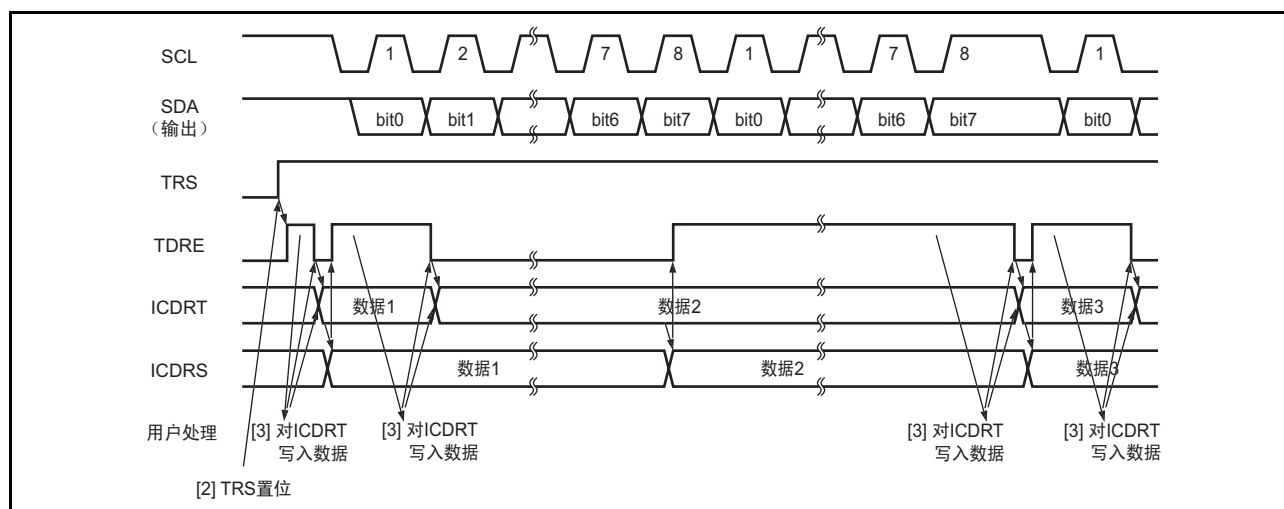


图 17.14 发送模式的运行时序

(3) 接收运行

接收模式时，在传送时钟的上升沿锁存数据。ICCR1 的 MST=1 时，输出传送时钟；MST=0 时，输入传送时钟。有关接收模式的运行时序，参照图 17.15。接收模式的步骤和运行如下所示：

1. 将 ICCR1 的 ICE 位置 1，设定 ICCR1 的 CKS[3:0] 等（初始设定）。
2. 输出传送时钟时，设置 MST=1。由此开始输出接收时钟。
3. 接收结束时，从 ICDRS 向 ICDRR 传送数据，并置位 ICSR 的 RDRF。MST=1 时，因为下一字节处于可接收状态，所以连续输出时钟。每当 RDRF 置位时，通过读取 ICDRR，可连续接收。在 RDRF 置位的状态，如果第 8 个时钟上升，则检测溢出，并置位 ICSR 的 AL/OVE。此时，ICDRR 的值保持之前的接收数据。
4. MST=1 时，为了停止接收，在置位 ICCR1 的 RCVD 后，读取 ICDRR。由此可在接收完下一字节后，将 SCL 固定为高电平。

【注】 MST=1 时，如果只想接收 1 字节，则必须按照以下步骤执行。有关运行时序，参照图 17.16。

1. 将 ICCR1 的 ICE 位置 1，设定 ICCR1 的 CKS[3:0] 等（初始设定）。
2. 在 ICCR1 的 RCVD 位为 0 的状态，设置 MST=1。由此开始输出接收时钟。
3. 在确认 ICMR 的 BC[2] 位置 1 后，设置 ICCR1 的 RCVD=1。由此可在输出 1 字节的接收时钟后，将 SCL 固定为高电平。

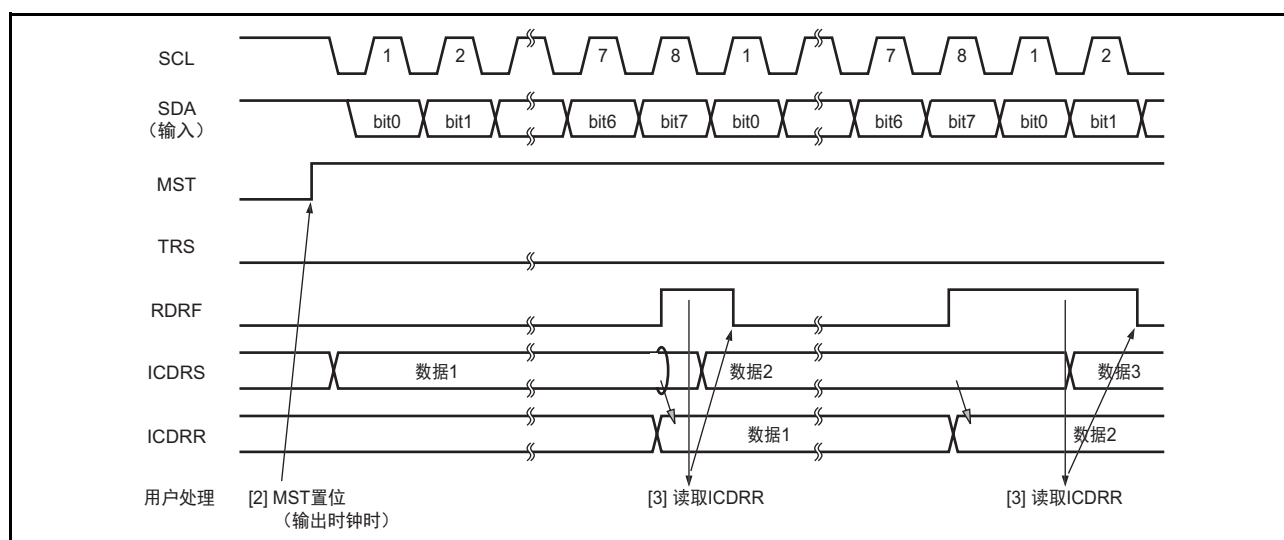


图 17.15 接收模式的运行时序

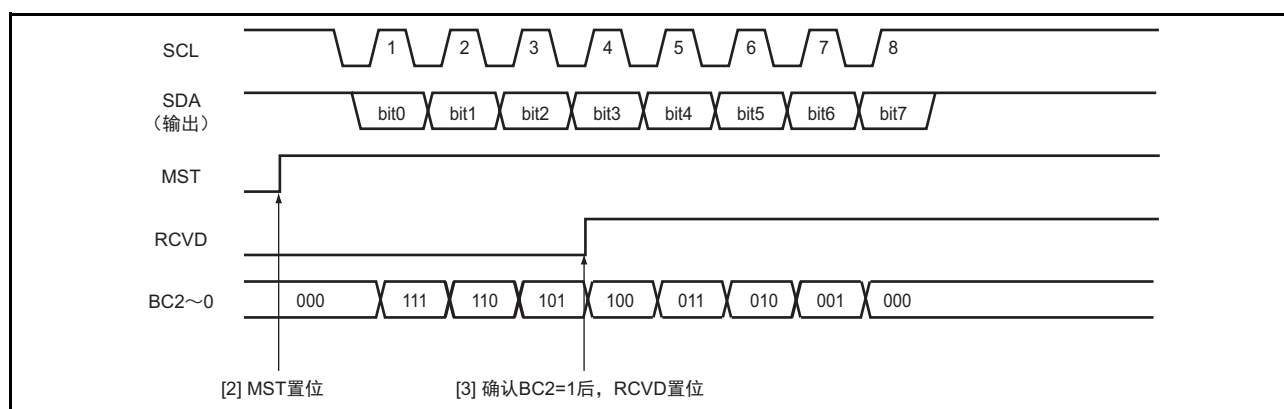


图 17.16 1 字节接收运行的时序 (MST=1)

17.4.7 噪声消除电路

SCL 引脚及 SDA 引脚的状态经过噪声消除电路被输入至内部。噪声消除电路框图如图 17.17 所示。

噪声消除电路由 3 段串联的锁存电路与匹配检测电路构成。通过外围时钟采样 SCL 引脚的输入信号（或 SDA 引脚的输入信号），并在 NF2CYC 寄存器为“0”，且 2 个锁存器输出匹配时，向后段电路传递电平。NF2CYC 寄存器为“1”，且 3 个锁存器输出匹配时，向后段电路传递电平。如果不匹配，则保持之前的值。

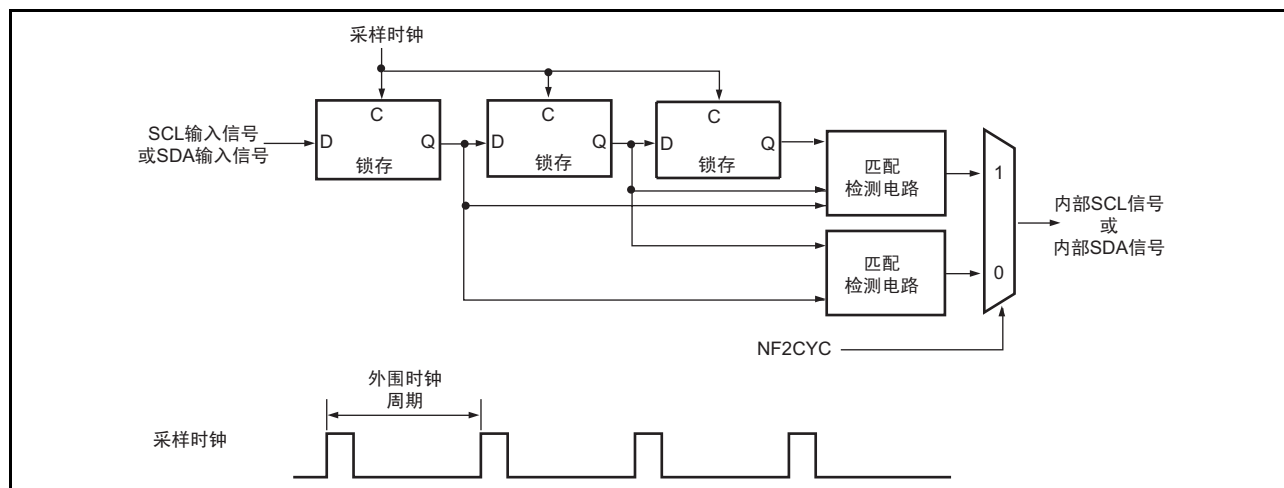


图 17.17 噪声消除电路框图

17.4.8 使用例

使用 I²C 总线接口 3 时的各模式流程图例如图 17.18 ~ 图 17.21 所示。

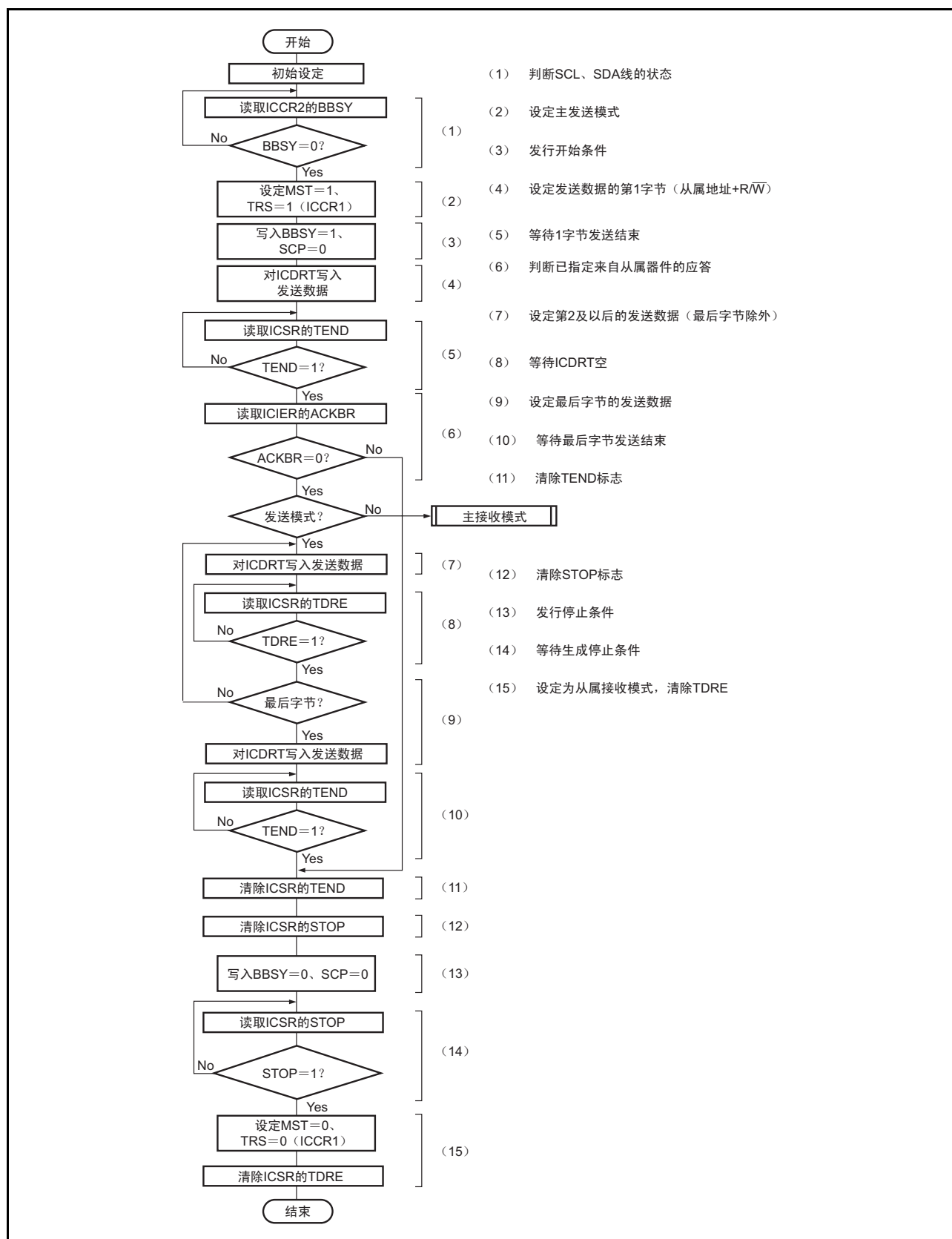


图 17.18 主发送模式的流程图例

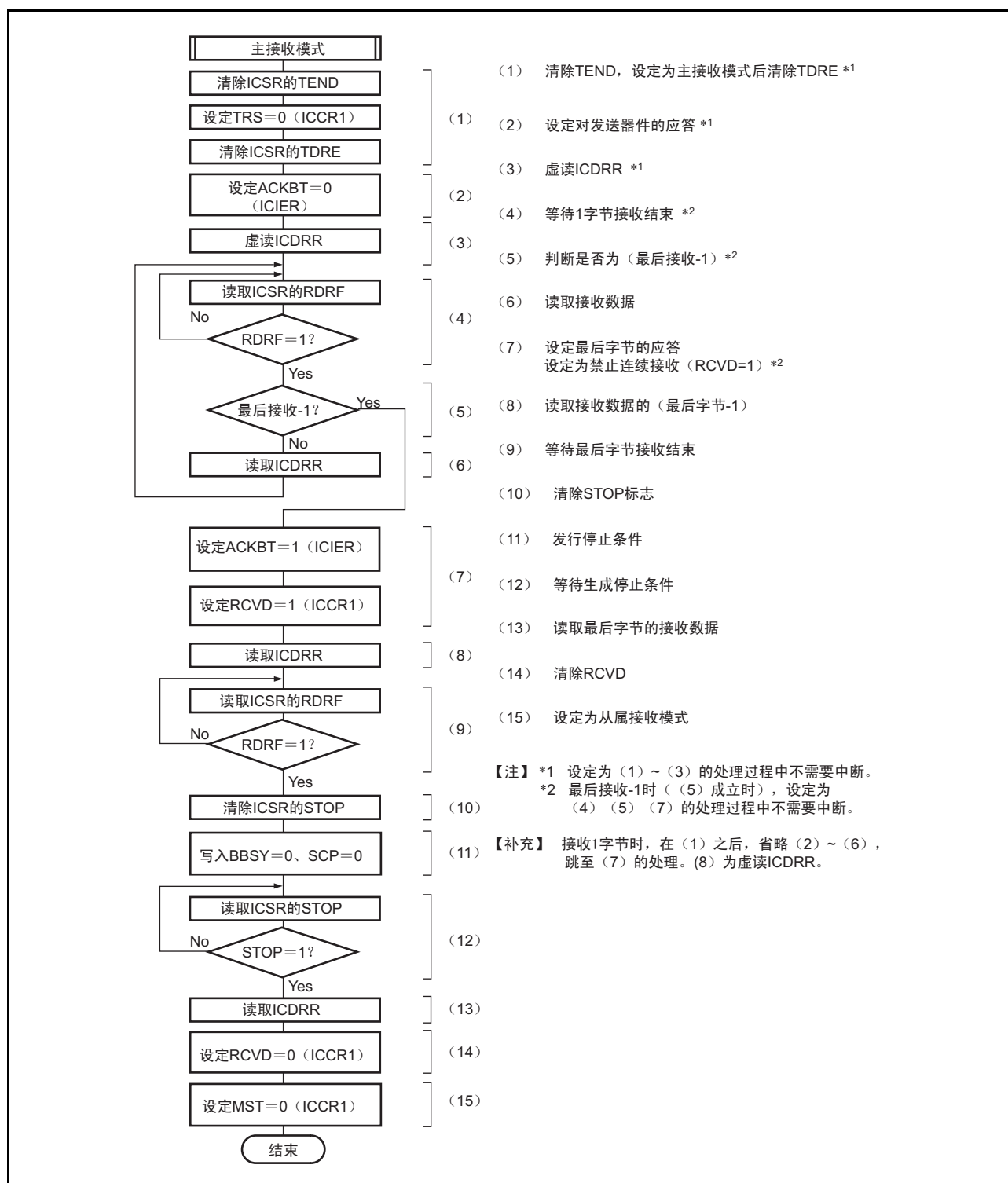


图 17.19 主接收模式的流程图例

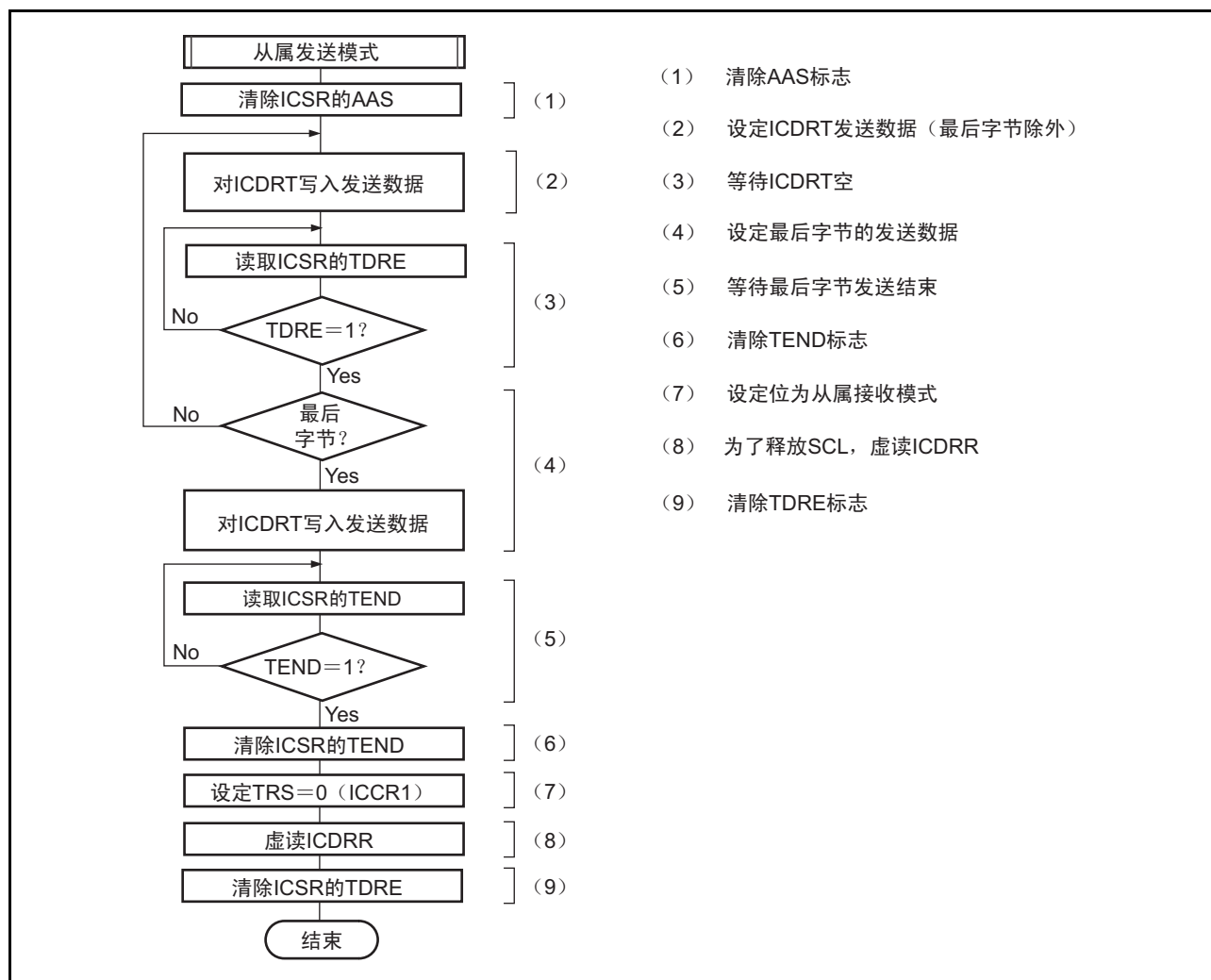


图 17.20 从属发送模式的流程图例

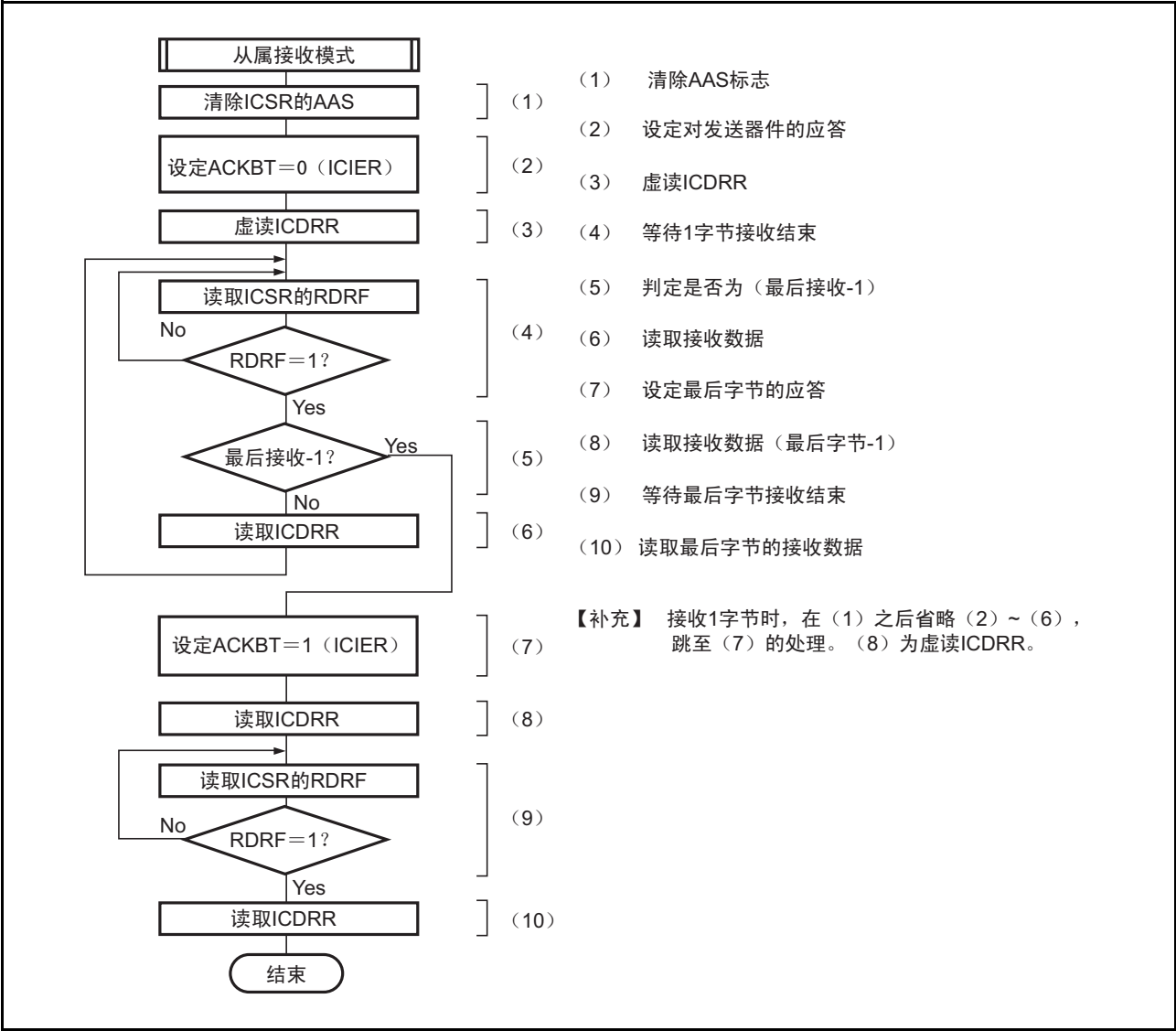


图 17.21 从属接收模式的流程图例

17.5 中断请求

I²C 总线接口 3 的中断请求有 6 种，即：发送数据空、发送结束、接收数据满、检测 NACK、检测停止条件、仲裁失败 / 溢出错误。各中断请求的内容如表 17.4 所示。

表 17.4 中断请求一览表

中断请求	简称	中断条件	I ² C 总线格式	时钟同步串行格式
发送数据空	TXI	(TDRE=1) • (TIE=1)	○	○
发送结束	TEI	(TEND=1) • (TEIE=1)	○	○
接收数据满	RXI	(RDRF=1) • (RIE=1)	○	○
检测停止条件	STPI	(STOP=1) • (STIE=1)	○	×
检测 NACK	NAKI	{(NACKF=1)+(AL=1)} • (NAKIE=1)	○	×
仲裁失败 / 溢出错误			○	○

表 17.4 的中断条件为 1 时，CPU 执行中断异常处理。执行 DMAC 的启动设定时，TXI 与 RXI 可启动 DMAC。此时，不对 CPU 产生中断。必须在异常处理中清除各中断源。但须注意：通过对 ICDRT 写入发送数据，可自动清除 TDRE 与 TEND；通过读取 ICDRR，可自动清除 RDRF。特别是对 ICDRT 写入发送数据的同时，TDRE 再次置位，且在清除 TDRE 时，有可能多发送 1 字节数据。

17.6 位同步电路

设定主模式时，I²C 总线接口 3 有以下 2 种情况：

- 由从属器件将 SCL 拉为低电平时
- 由于 SCL 线的负载（负载电容、上拉电阻），SCL 的上升变得缓慢时

在这两种状态，高电平期间可能变短，因此需监控 SCL，并按位在取得同步的同时执行通信。

位同步电路的时序如图 17.22 所示，将 SCL 从低电平输出设置为 Hi-Z，到监控 SCL 为止的时间如表 17.5 所示。

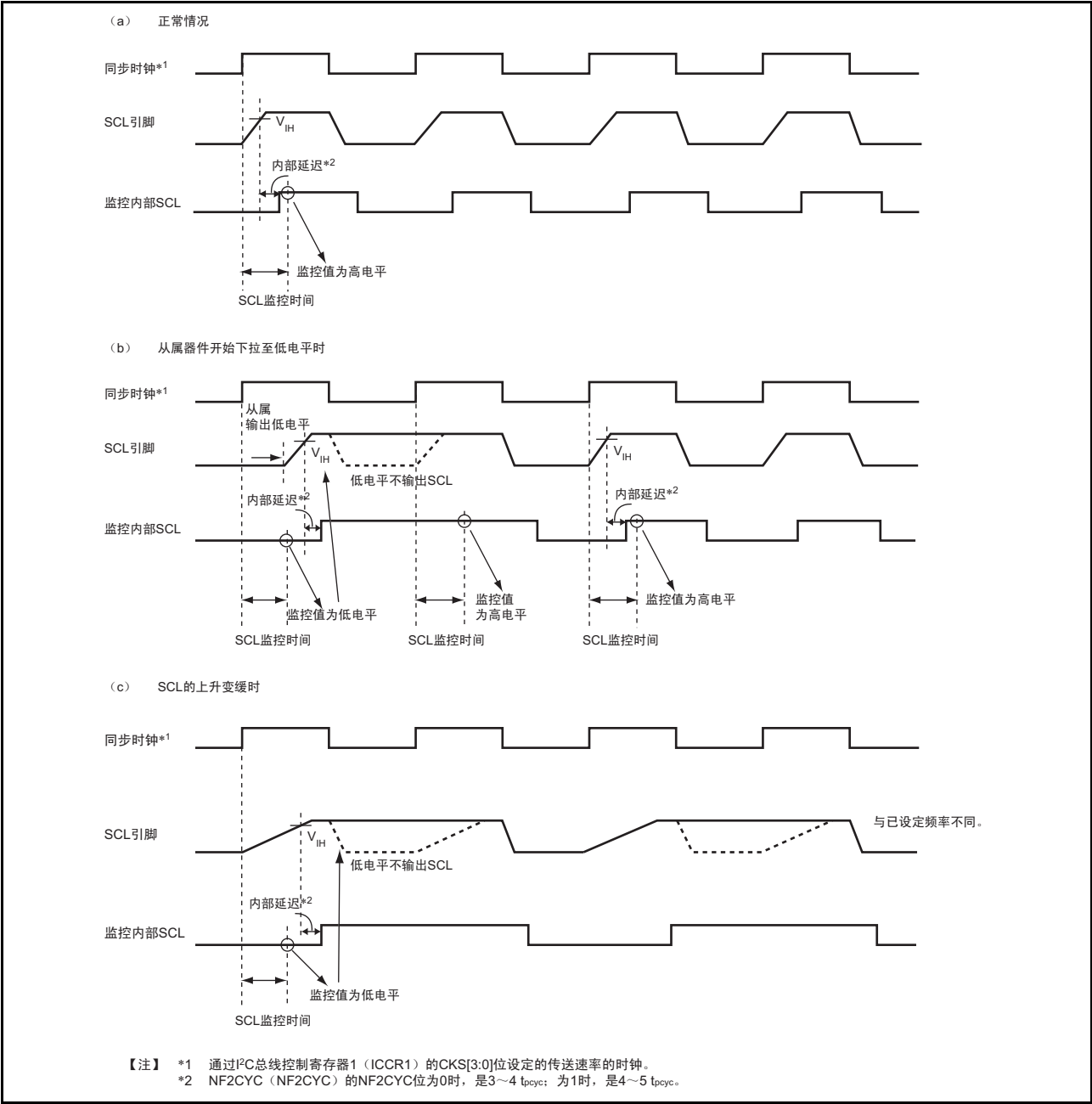


图 17.22 位同步电路的时序

表 17.5 监控 SCL 的时间

CKS[3]	CKS[2]	监控 SCL 的时间
0	0	9tpcyc*
	1	21tpcyc*
1	0	33tpcyc*
	1	81tpcyc*

【注】 * tpcyc 表示外围时钟的 (Pφ) 同步。

17.7 使用时的注意事项

17.7.1 设定 ICCR1.CKS[3:0] 时的注意事项

不得设定 ICCR1.CKS[3:0]=H'7 或 H'F、NF2CYC.PRS=1。

17.7.2 用作多主器件时的注意事项

用作多主器件，本 LSI 的 IIC 传送速率的设定 (ICCR1 CKS[3:0]) 比其他主器件慢时，偶尔对 SCL 输出预期宽度之外的 SCL。

需将传送速率设定为大于等于其他主器件最快传送速率的 1/1.8。

17.7.3 主接收模式时的注意事项

在第 8 个时钟的下降沿附近读取 ICDRR 时，有可能无法获取接收数据。

另外，接收缓冲器满且在第 8 个时钟的下降沿附近设定 RCVD=1 时，有可能无法发行停止条件。

请使用以下 1 或 2 的方法解决：

1. 主接收模式时，必须在第 8 个时钟的上升沿之前读取 ICDRR。
2. 主接收模式时，设置 RCVD=1，按每 1 字节执行通信处理。

17.7.4 主接收模式、ACKBT 设定时的注意事项

主接收模式运行时，必须在连续传送最后数据的第 8 个 SCL 下降之前，设定 ACKBT。从属发送的器件有可能会产生溢出。

17.7.5 仲裁失败时 MST 与 TRN 位状态的注意事项

用作多主器件时，如果将 MST、TRS 按顺序执行位操作并设定为主发送，则根据 TRS 的位操作指令执行过程中产生仲裁失败的时序，可能出现类似 ICSR 的 AL=1 且为主发送模式 (MST=1、TRS=1) 的矛盾。

以下方法可避免产生这些矛盾：

- 用作多主器件时，必须使用 MOV 指令设定 MST、TRS。
- 仲裁失败时，必须确认 MST=0、TRS=0。

不是 MST=0、TRS=0 的状态时，则必须重新设定 MST=0、TRS=0。

第 18 章 串行音频接口 (SSI)

串行音频接口模块 (SSI) 是发送 / 接收与飞利浦格式兼容的各种器件及音频数据的模块。除其他通用格式之外, 也支持多通道模式。

18.1 特点

- 通道数: 4 个通道
- 运行模式: 非压缩模式
非压缩模式支持分配至通道的串行音频流。
- 即可作为发送器也可作为接收器
- 可使用串行总线格式
- 数据缓冲器和移位寄存器之间的传送为异步传送
- 可选择串行总线接口使用的时钟分频比
- 通过 DMAC 或中断, 可控制发送 / 接收数据
- 可从以下引脚选择过采样时钟

EXTAL、XTAL (时钟运行模式 0、1: 10 ~ 33.33MHz)

CKIO (时钟运行模式 2: 40 ~ 50MHz*)

AUDIO_CLK (1 ~ 40MHz)

AUDIO_X1、AUDIO_X2 (连接晶体谐振器时: 10 ~ 40MHz; 输入外部时钟时: 1 ~ 40MHz)

【注】 * 在时钟运行模式 2, 使用超过 50MHz 的 CKIO 频率时, 不得将 CKIO 设定为过采样时钟的提供源。

SSI 模块的概况如图 18.1 所示。

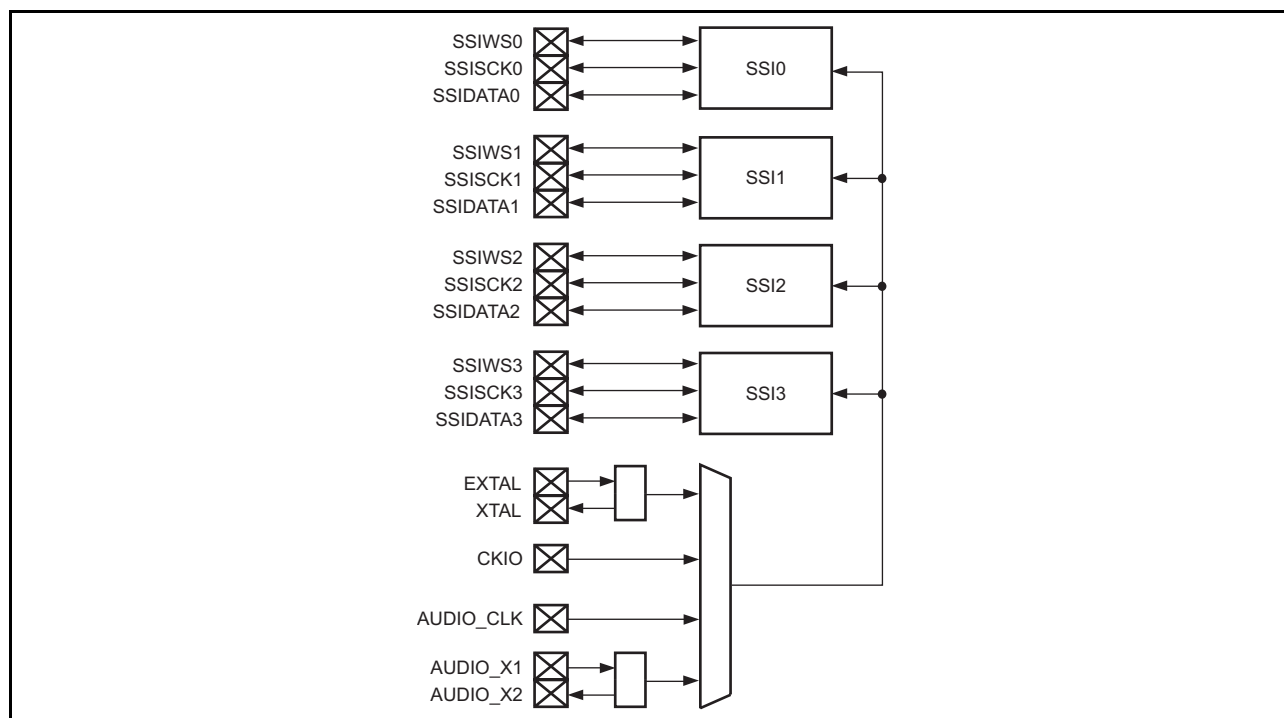


图 18.1 SSI 模块的概况图

SSI 框图如图 18.2 所示。

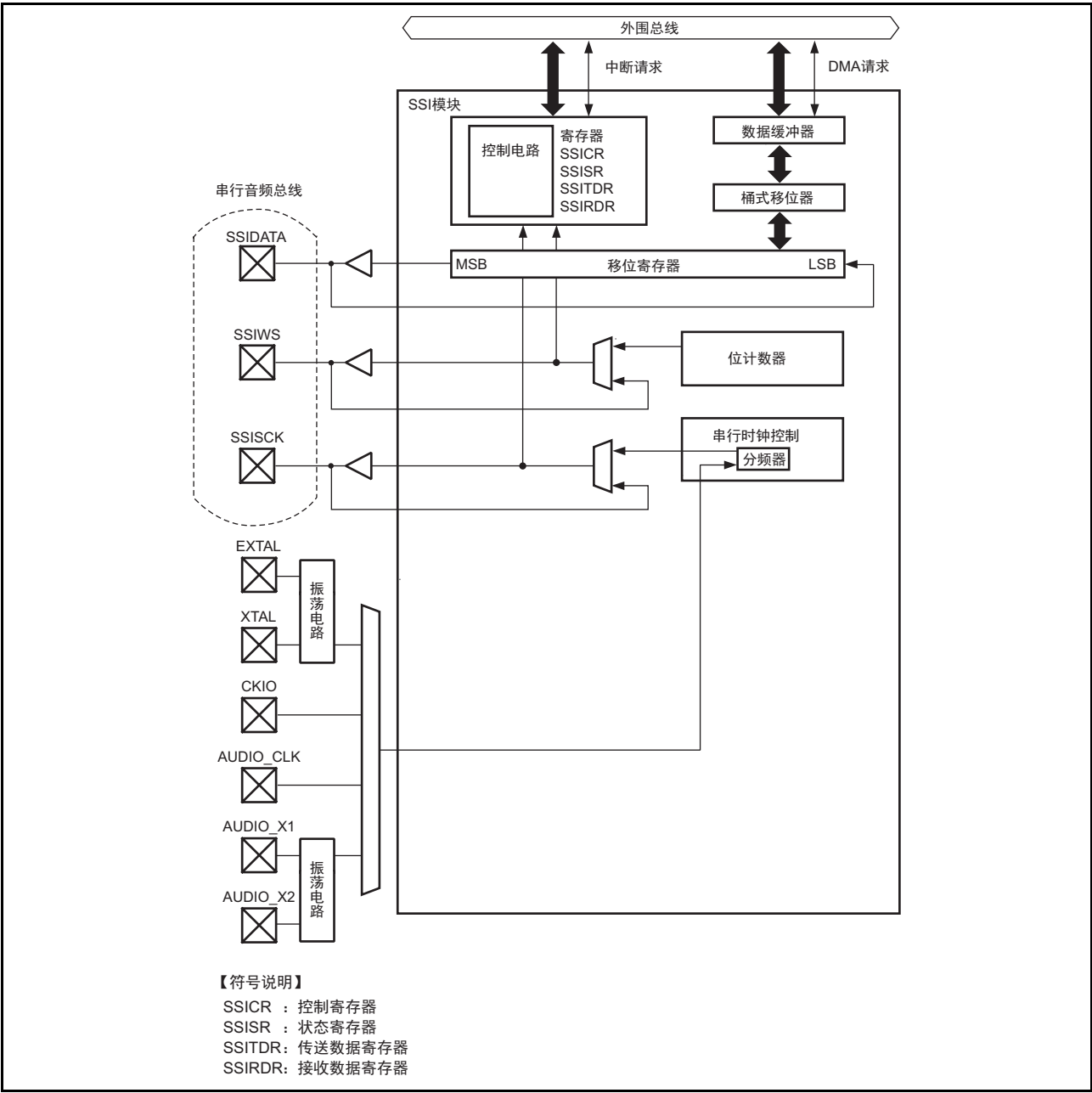


图 18.2 SSI 框图

18.2 输入 / 输出引脚

与 SSI 模块有关的引脚结构如表 18.1 所示。

表 18.1 引脚结构

名称	数量	输入 / 输出	功 能
SSISCK0	1	输入 / 输出	串行位时钟
SSIWS0	1	输入 / 输出	字选择
SSIDATA0	1	输入 / 输出	串行数据输入 / 输出
SSISCK1	1	输入 / 输出	串行位时钟
SSIWS1	1	输入 / 输出	字选择
SSIDATA1	1	输入 / 输出	串行数据输入 / 输出
SSISCK2	1	输入 / 输出	串行位时钟
SSIWS2	1	输入 / 输出	字选择
SSIDATA2	1	输入 / 输出	串行数据输入 / 输出
SSISCK3	1	输入 / 输出	串行位时钟
SSIWS3	1	输入 / 输出	字选择
SSIDATA3	1	输入 / 输出	串行数据输入 / 输出
AUDIO_CLK	1	输入	用于音频的外部时钟 (输入过采样时钟)
AUDIO_X1	1	输入	用于音频的晶体谐振器 (输入过采样时钟)
AUDIO_X2	1	输出	

18.3 寄存器说明

SSI 有以下寄存器。本文中，省略通道间的区别，进行简单说明。

表 18.2 寄存器结构

通道	寄存器名称	简称	R/W	初始值	地址	存取长度
0	控制寄存器 0	SSICR_0	R/W	H'00000000	H'FFFC000	32
	状态寄存器 0	SSISR_0	R/W*	H'02000003	H'FFFC004	32
	发送数据寄存器 0	SSITDR_0	R/W	H'00000000	H'FFFC008	32
	接收数据寄存器 0	SSIRDR_0	R	H'00000000	H'FFFC00C	32
1	控制寄存器 1	SSICR_1	R/W	H'00000000	H'FFFC800	32
	状态寄存器 1	SSISR_1	R/W*	H'02000003	H'FFFC804	32
	发送数据寄存器 1	SSITDR_1	R/W	H'00000000	H'FFFC808	32
	接收数据寄存器 1	SSIRDR_1	R	H'00000000	H'FFFC80C	32
2	控制寄存器 2	SSICR_2	R/W	H'00000000	H'FFFD000	32
	状态寄存器 2	SSISR_2	R/W*	H'02000003	H'FFFD004	32
	发送数据寄存器 2	SSITDR_2	R/W	H'00000000	H'FFFD008	32
	接收数据寄存器 2	SSIRDR_2	R	H'00000000	H'FFFD00C	32
3	控制寄存器 3	SSICR_3	R/W	H'00000000	H'FFFD800	32
	状态寄存器 3	SSISR_3	R/W*	H'02000003	H'FFFD804	32
	发送数据寄存器 3	SSITDR_3	R/W	H'00000000	H'FFFD808	32
	接收数据寄存器 3	SSIRDR_3	R	H'00000000	H'FFFD80C	32

【注】 * 本寄存器的 bit27 和 bit26 可读取 / 写入，其他位为只读位。详情参阅“18.3.2 状态寄存器 (SSISR)”。

18.3.1 控制寄存器 (SSICR)

SSICR 为 32 位可读取 / 写入的寄存器，控制 IRQ、选择各极性状态，并设定运行模式。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	DMEN	UIEN	OIEN	IIEN	DIEN	CHNL[1:0]		DWL[2:0]			SWL[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SCKD	SWSD	SCKP	SWSP	SPDP	SDTA	PDTA	DEL	-		CKDV[2:0]		MUEN	-	TRMD	EN
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R	R/W	R/W

位	位名称	初始值	R/W	说 明
31 ~ 29	—	均为 0	R	保留位 读取值、写入值总是为 0。
28	DMEN	0	R/W	DMA 允许 允许 / 禁止 DMA 请求。 0: 禁止 DMA 请求 1: 允许 DMA 请求
27	UIEN	0	R/W	下溢中断允许 0: 禁止下溢中断 1: 允许下溢中断
26	OIEN	0	R/W	上溢中断允许 0: 禁止上溢中断 1: 允许上溢中断
25	IIEN	0	R/W	空闲模式中断允许 0: 禁止空闲模式中断 1: 允许空闲模式中断
24	DIEN	0	R/W	数据中断允许 0: 禁止数据中断 1: 允许数据中断
23、22	CHNL[1:0]	00	R/W	通道 表示各系统字的通道数。 00: 各系统字由 1 个通道构成。 01: 各系统字由 2 个通道构成。 10: 各系统字由 3 个通道构成。 11: 各系统字由 4 个通道构成。
21 ~ 19	DWL[2:0]	000	R/W	数据字长 表示数据字的位数。 000: 8 位 001: 16 位 010: 18 位 011: 20 位 100: 22 位 101: 24 位 110: 32 位 111: 禁止设定

位	位名称	初始值	R/W	说 明															
18 ~ 16	SWL[2:0]	000	R/W	系统字长 表示系统字的位数。 000: 8 位 001: 16 位 010: 24 位 011: 32 位 100: 48 位 101: 64 位 110: 128 位 111: 256 位															
15	SCKD	0	R/W	串行位时钟方向 0: 串行位时钟输入、从属模式 1: 串行位时钟输出、主模式 【注】 仅可设定（SCKD、SWSD）=（0、0）和（1、1），禁止其他的设定															
14	SWSD	0	R/W	串行 WS 方向 0: 串行字选择输入、从属模式 1: 串行字选择输出、主模式 【注】 仅可设定（SCKD、SWSD）=（0、0）和（1、1），禁止其他的定。															
13	SCKP	0	R/W	串行位时钟极性 0: SSIWS 和 SSIDATA 在 SSISCK 下降沿变化 （在 SCK 的上升沿采样） 1: SSIWS 和 SSIDATA 在 SSISCK 上升沿变化 （在 SCK 的下降沿采样） <table border="1"><tr><td></td><td>SCKP=0</td><td>SCKP=1</td></tr><tr><td>接收时（TRMD=0），SSIDATA 输入采样时序</td><td>SSISCK 上升沿</td><td>SSISCK 下降沿</td></tr><tr><td>发送时（TRMD=1），SSIDATA 输出改变时序</td><td>SSISCK 下降沿</td><td>SSISCK 上升沿</td></tr><tr><td>从属模式时（SWSD=0），SSIWS 输入采样时序</td><td>SSISCK 上升沿</td><td>SSISCK 下降沿</td></tr><tr><td>主模式时（SWSD=1），SSIWS 输出改变时序</td><td>SSISCK 下降沿</td><td>SSISCK 上升沿</td></tr></table>		SCKP=0	SCKP=1	接收时（TRMD=0），SSIDATA 输入采样时序	SSISCK 上升沿	SSISCK 下降沿	发送时（TRMD=1），SSIDATA 输出改变时序	SSISCK 下降沿	SSISCK 上升沿	从属模式时（SWSD=0），SSIWS 输入采样时序	SSISCK 上升沿	SSISCK 下降沿	主模式时（SWSD=1），SSIWS 输出改变时序	SSISCK 下降沿	SSISCK 上升沿
	SCKP=0	SCKP=1																	
接收时（TRMD=0），SSIDATA 输入采样时序	SSISCK 上升沿	SSISCK 下降沿																	
发送时（TRMD=1），SSIDATA 输出改变时序	SSISCK 下降沿	SSISCK 上升沿																	
从属模式时（SWSD=0），SSIWS 输入采样时序	SSISCK 上升沿	SSISCK 下降沿																	
主模式时（SWSD=1），SSIWS 输出改变时序	SSISCK 下降沿	SSISCK 上升沿																	
12	SWSP	0	R/W	串行 WS 极性 0: SSIWS 在第 1 通道为低电平，在第 2 通道为高电平 1: SSIWS 在第 1 通道为高电平，在第 2 通道为低电平															
11	SPDP	0	R/W	串行填充极性 0: 填充位为低电平 1: 填充位为高电平 【注】 MUEN=1 时，填充位为低电平。（优先静噪功能）															

位	位名称	初始值	R/W	说 明
10	SDTA	0	R/W	串行数据对齐 0: 按照串行数据、填充位的顺序发送 / 接收 1: 按照填充位、串行数据的顺序发送 / 接收
9	PDTA	0	R/W	并行数据对齐 数据字长为 32、16、8 位时, 该位无意义。 本位适用于接收模式时的 SSIRD R 和发送模式时的 SSITDR。 0: 向左对齐并行数据 (SSITDR、SSIRD R) 1: 向右对齐并行数据 (SSITDR、SSIRD R) • DWL=000 (数据的字长为 8 位) 时, 忽略 PDTA 设定 在串行音频总线可使用 SSIRD R 或 SSITDR 的所有数据位。在各 32 位存取时发送 / 接收 4 个数据字。第 1 个数据字保存在 bit7 ~ 0、第 2 个数据字保存在 bit15 ~ 8、第 3 个数据字保存在 bit23 ~ 16、第 4 个数据字保存在 bit31 ~ 24。 • DWL=001 (数据的字长为 16 位) 时, 忽略 PDTA 设定 在串行音频总线可使用 SSIRD R 或 SSITDR 的所有数据位。在各 32 位存取时发送 / 接收 2 个数据字。第 1 个数据字保存在 bit15 ~ 0、第 2 个数据字保存在 bit31 ~ 16。 • DWL=010、011、100、101 (数据的字长为 18、20、22、24 位时)、PDTA=0 (向左对齐) 在 SSIRD R 或 SSITDR 的数据位使用以下位: bit31 ~ bit (通过 32-DWL 设定的数据字长的位数) 即, DWL=011 时, 数据的字长为 20 位, 可使用 SSIRD R 或 SSITDR 的 bit31 ~ 12。其他所有的位被忽略或为保留位。 • DWL=010、011、100、101 (数据的字长为 18、20、22、24 位时) PDTA=1 (向右对齐) SSIRD R 或 SSITDR 的数据位使用以下位: bit (通过 DWL 设定的数据字长的位数 -1) ~ bit0 即, DWL=011 时, 数据的字长为 20 位, 可使用 SSIRD R 或 SSITDR 的 bit19 ~ 0。其他所有的位被忽略或为保留位。 • DWL=110 (数据的字长为 32 位) 时, 忽略 PDTA 设定 在串行音频总线可使用 SSIRD R 或 SSITDR 的所有数据位。
8	DEL	0	R/W	串行数据延迟 0: SSIWS 和 SSIDATA 之间延迟 1 个时钟周期 1: SSIWS 和 SSIDATA 之间无延迟
7	—	0	R	保留位 读取值不定, 写入值为 0。

位	位名称	初始值	R/W	说 明
6 ~ 4	CKDV[2:0]	000	R/W	串行过采样时钟分频比 设定过采样时钟 * 与串行位时钟的分频比。SCKD=0 时, 忽略此位。 在移位寄存器可使用串行位时钟, 并由 SSISCK 引脚提供。 000: 串行位时钟频率 = 过采样时钟频率 /1 001: 串行位时钟频率 = 过采样时钟频率 /2 010: 串行位时钟频率 = 过采样时钟频率 /4 011: 串行位时钟频率 = 过采样时钟频率 /8 100: 串行位时钟频率 = 过采样时钟频率 /16 101: 串行位时钟频率 = 过采样时钟频率 /6 110: 串行位时钟频率 = 过采样时钟频率 /12 111: 禁止设定 【注】* 根据 PFC 内 SCSR 的设定, 选择过采样时钟。 详情参阅 “第 25 章 引脚功能控制器 (PFC)”。
3	MUEN	0	R/W	静噪允许 0: SSI 模块不为静噪状态 1: SSI 模块为静噪状态
2	—	0	R/W	保留位 读取值不定, 写入值为 0。
1	TRMD	0	R/W	选择发送 / 接收模式 0: SSI 模块为接收模式 1: SSI 模块为发送模式
0	EN	0	R/W	SSI 模块允许 0: 禁止 SSI 模块运行 1: 允许 SSI 模块运行

18.3.2 状态寄存器 (SSISR)

SSISR 由表示 SSI 模块运行状态的标志及表示当前通道编号与字编号的位构成。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	DMRQ	UIRQ	OIRQ	IIRQ	DIRQ	-	-	-	-	-	-	-	-
初始值:	0	0	0	0	0	0	1	0	不定	不定	不定	不定	不定	不定	不定	不定
R/W:	R	R	R	R	R/W*	R/W*	R	R	R	R	R	R	R	R	R	R

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	CHNO[1:0]	SWNO	IDST	
初始值:	不定	不定	不定	不定	不定	不定	不定	不定	不定	不定	不定	不定	0	0	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

【注】* 可读取或写入。写入0时初始化该位，写入1时忽略。

位	位名称	初始值	R/W	说 明
31 ~ 29	—	均为 0	R	保留位 读取值不定，写入值为 0
28	DMRQ	0	R	DMA 请求状态标志 根据本状态标志，CPU 可得知 SSI 模块的 DMA 请求状态。 [TRMD=0 (接收模式) 时] - DMRQ=1 时，SSIRD 中有未读数据。 - 读取 SSIRD 时，到出现下一个未读数据之前，DMRQ=0。 [TRMD=1 (发送模式) 时] - DMRQ=1 时，为了串行音频总线的发送能够继续，SSITDR 请求写入数据。 - 对 SSITDR 写入数据时，到产生下一个发送数据请求之前，DMRQ=0。
27	UIRQ	0	R/W*	下溢错误中断状态标志 本状态标志表示用低于请求速率的速率提供数据。 与 UIEN 位的设定无关，将本位置 1。清 0 时，必须写入 0。 UIRQ=1 且 UIEN=1 时，产生中断。 [TRMD=0 (接收模式) 时] UIRQ=1 时，在表示 DMRQ 或 DIRQ 位存在新的未读数据前，表示已读取 SSIRD。此时，根据主机不同，可能保存 2 次相同的接收数据，并可能破坏多通道数据。 [TRMD=1 (发送模式) 时] UIRQ=1 时，表示发送前不可向 SSITDR 写入发送数据。因此，可能不止 1 次发送相同数据，并破坏多通道数据。结果是输出错误的 SSI 数据，因此该错误比接收模式的下溢更严重。 【注】产生下溢错误时，到写入下一个数据前，发送数据缓冲器的数据。

位	位名称	初始值	R/W	说 明
26	OIRQ	0	R/W*	上溢错误中断状态标志 本状态标志表示用高于请求速率的速率提供数据。 与 OIEN 位的设定无关，将本位置 1。清 0 时，必须写入 0。 OIRQ=1 且 OIEN=1 时，产生中断。 [TRMD=0 (接收模式) 时] OIRQ=1 时，表示向 SSIRD R 写入新的未读数据前，不可读取以前的未 读数据。因此，可能丢失数据，并破坏多通道数据。 【注】 产生上溢错误时，从 SSI 接口发送来的下一个数据重写数据缓冲器中的数据。 TRMD=1 (发送模式) 时] OIRQ=1 时，表示 SSITDR 的数据传送到移位寄存器前，已向 SSITDR 写入数据。因此有可能丢失数据，并破坏多通道数据。
25	IIRQ	1	R	空闲模式中断状态标志 本中断状态标志表示 SSI 模块是否为空闲状态。为了可查询，与 I IEN 位的设定无关，将本位置 1。 将 I IEN 位清 0，可屏蔽中断，但即使对本位写入 0，也无法清除中断。 I IIRQ=1 且 I IEN=1 时，产生中断。 0: SSI 模块不为空闲状态 1: SSI 模块为空闲状态
24	DIRQ	0	R	数据中断状态标志 本状态标志表示 SSI 模块需读取或写入数据。 为了可查询，与 DIEN 位的设定无关，将本位置 1。 将 DIEN 位清 0，可屏蔽中断，但即使对该位写入 0，也无法清除中断。 DIRQ=1 且 DIEN=1 时，产生中断。 [TRMD=0 (接收模式) 时] 0: SSIRD R 无未读数据 1: SSIRD R 有未读数据 [TRMD=1 (发送模式) 时] 0: 发送缓冲器满 1: 发送缓冲器空，请求向 SSITDR 写入数据
23 ~ 4	—	不定	R	保留位 读取值不定，写入值为 0。
3、2	CHNO [1:0]	00	R	通道编号 表示当前的通道。 [TRMD=0 (接收模式) 时] 本位表示当前 SSIRD R 内的数据属于哪个通道。通过移位寄存器的传送更新 SSIRD R 的数据，该值发生变化。 [TRMD=1 (发送模式) 时] 本位表示应向 SSITDR 写入哪个通道的数据。数据被复制到移位寄存器后，无论是否写入 SSITDR，该值发生变化。

位	位名称	初始值	R/W	说 明
1	SWNO	1	R	串行字编号 表示当前的字编号。 [TRMD=0 (接收模式) 时] 本位表示当前 SSIRD 内的数据为哪个系统字, 与是否读取 SSIRD 无关, 通过移位寄存器的传送更新 SSIRD 的数据, 该值发生变化。 [TRMD=1 (发送模式) 时] 本位表示应向 SSITDR 写入哪个系统字。数据被复制到移位寄存器时, 无论是否写入 SSITDR 无关, 该值发生变化。
0	IDST	1	R	空闲模式状态标志 本状态标志表示串行总线为停止状态。 EN=1 且串行总线正在运行时, 清除本位。 本位在以下条件自动置 1。 [SSI 为主传送器 (SWSD=1 且 TRMD=1) 时] 清除 EN 位, 且写入 SSITDR 的数据从串行数据输入 / 输出引脚 (SSIDATA) 完成输出 (完成系统字长的输出) 后, 本位置 1。 [SSI 为主接收器 (SWSD=1 且 TRMD=0) 时] 清除 EN 位, 当前的系统字结束后, 本位置 1。 [SSI 为从属传送器 / 接收器 (SWSD=0) 时] 清除 EN 位, 当前的系统字结束后, 本位置 1。 【注】 当前的系统字结束前, 如果外部器件停止串行总线时钟, 则本位不置位。

【注】 * 可读取 / 写入。写入 0 时本位初始化, 写入 1 时被忽略。

18.3.3 发送数据寄存器 (SSITDR)

SSITDR 为 32 位寄存器, 保存发送数据。

有发送请求时, 将写入本寄存器的数据传送至移位寄存器。如果数据字长不足 32 位, 根据 SSICR 的 PDTA 控制位的设定对齐。

读取本寄存器, 可得到缓冲器的数据。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

18.3.4 接收数据寄存器 (SSIRD)

SSIRD 为 32 位寄存器，保存接收信息。

每次接收数据字时，从移位寄存器传送本寄存器的数据。数据字长不足 32 位时，根据 SSICR 的 PDTA 控制位的设定对齐。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

18.4 运行说明

18.4.1 总线格式

SSI 模块可作为发送器运行，也可作为接收器运行，任何模式均可使用多个串行总线格式。总线格式可从 8 个主要模式选择，如表 18.3 所示。

表 18.3 SSI 模块的总线格式

	TRMD	SCKD	SWSD	EN	MJEN	DIEN	IJEN	OJEN	UIJEN	DEL	PDTA	SDTA	SPDP	SWSP	SCKP	SWL [2:0]	DWL [2:0]	CHNL [1:0]
非压缩从属接收器	0	0	0	控制位						结构位								
非压缩从属发送器	1	0	0															
非压缩主接收器	0	1	1															
非压缩主发送器	1	1	1															

18.4.2 非压缩模式

非压缩模式支持分配至通道的串行音频流，除了飞利浦、索尼或松下模式外，还支持多个改良版本。

(1) 从属接收器

在该模式可从其他器件接收串行数据。从外部器件提供用于串行数据流的时钟与字选择信号。这些信号与 SSI 模块所设定的格式不匹配时，无法保证运行。

(2) 从属发送器

在该模式可向其他器件发送串行数据。从外部器件提供用于串行数据流的时钟与字选择信号。这些信号与 SSI 模块所设定的格式不匹配时，无法保证运行。

(3) 主接收器

在该模式可从其他器件接收串行数据。过采样时钟内部生成时钟与字选择信号。这些信号的格式取决于 SSI 模块的设定。从其他器件发送的数据与设定的格式不匹配时，无法保证运行。

(4) 主发送器

在该模式可向其他器件发送串行数据。从过采样时钟内部生成时钟与字选择信号。这些信号的格式取决于 SSI 模块结构位的设定。

(5) 运行设定 - 字长度的关系

在非压缩模式，SSICR 的字长度相关的所有位均有效。SSI 模块可支持多个结构，有关飞利浦、索尼、松下格式：

- 飞利浦格式

有 / 无填充的飞利浦格式分别如图 18.3、图 18.4 所示。数据字长度比系统字长度短时，产生填充。

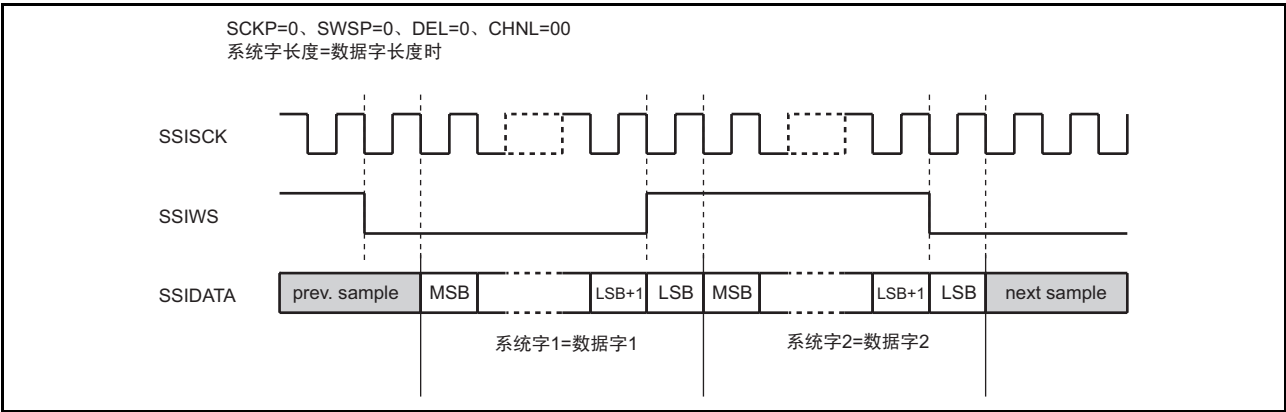


图 18.3 飞利浦格式（无填充）

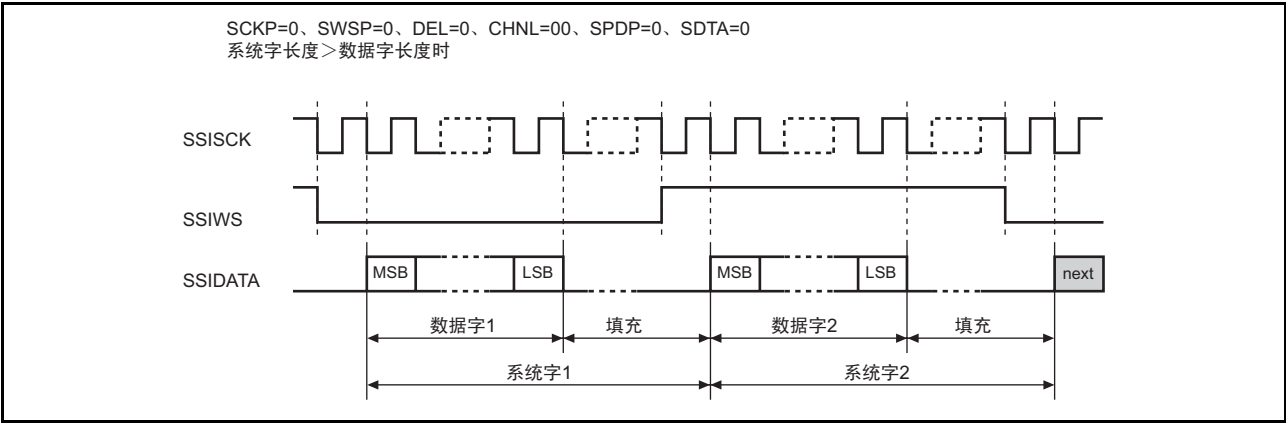


图 18.4 飞利浦格式（有填充）

索尼、松下的格式分别如图 18.5、图 18.6 所示，均为有填充的例子，但系统的字长度与数据的字长度相同时，会出现无填充的情况。

• 索尼格式

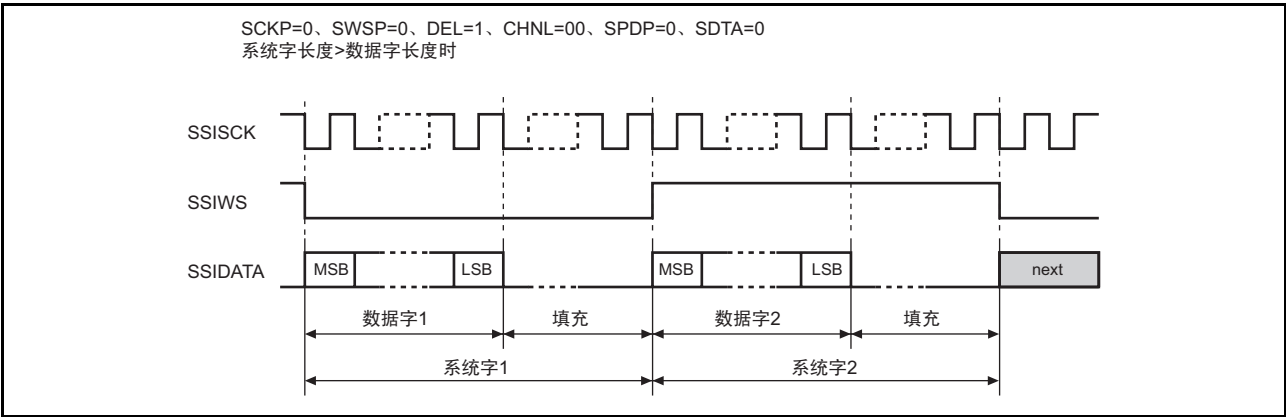


图 18.5 索尼格式（按照串行数据、填充位的顺序发送 / 接收）

• 松下格式

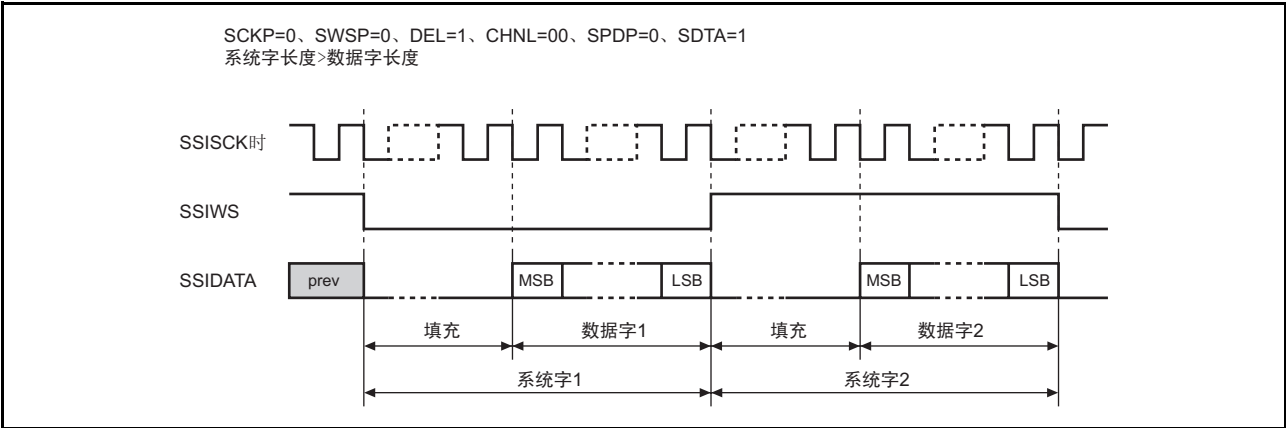


图 18.6 松下格式（按照填充位、串行数据的顺序发送 / 接收）

(6) 多通道格式

有些器件可扩展飞利浦规格的定义，在 2 个系统字执行超过 2 个通道的传送。

SSI 模块使用 CHNL、SWL 及 DWL 位执行 4、6、8 个通道的传送，但仅限于系统字长度（SWL）大于等于数据字长度（DWL）与通道数（CHNL）的乘积的情况。

有效设定与填充位数如表 18.4 所示。设定无效用 “—” 代替数字来表示。

表 18.4 有效设定和填充位数

每个系统字的填充位数			DWL[2:0]	000	001	010	011	100	101	110
CHNL [1:0]	解码每个系统字的通道	SWL [2:0]	解码后的 字长度	8	16	18	20	22	24	32
00	1	000	8	0	—	—	—	—	—	—
		001	16	8	0	—	—	—	—	—
		010	24	16	8	6	4	2	0	—
		011	32	24	16	14	12	10	8	0
		100	48	40	32	30	28	26	24	16
		101	64	56	48	46	44	42	40	32
		110	128	120	112	110	108	106	104	96
		111	256	248	240	238	236	234	232	224
01	2	000	8	—	—	—	—	—	—	—
		001	16	0	—	—	—	—	—	—
		010	24	8	—	—	—	—	—	—
		011	32	16	0	—	—	—	—	—
		100	48	32	16	12	8	4	0	—
		101	64	48	32	28	24	20	16	0
		110	128	112	96	92	88	84	80	64
		111	256	240	224	220	216	212	208	192
10	3	000	8	—	—	—	—	—	—	—
		001	16	—	—	—	—	—	—	—
		010	24	0	—	—	—	—	—	—
		011	32	8	—	—	—	—	—	—
		100	48	24	0	—	—	—	—	—
		101	64	40	16	10	4	—	—	—
		110	128	104	80	74	68	62	56	32
		111	256	232	208	202	196	190	184	160
11	4	000	8	—	—	—	—	—	—	—
		001	16	—	—	—	—	—	—	—
		010	24	—	—	—	—	—	—	—
		011	32	0	—	—	—	—	—	—
		100	48	16	—	—	—	—	—	—
		101	64	32	0	—	—	—	—	—
		110	128	96	64	56	48	40	32	0
		111	256	224	192	184	176	168	160	128

SSI 模块作为发送器运行时，写入 SSITDR 的各字，按照写入顺序发送至串行音频总线；SSI 模块作为接收器运行时，串行音频总线接收的各字，按照从 SSIRD R 接收的顺序读取。

4、6、8 个通道的数据如何传送到串行音频总线，如图 18.7 ~ 图 18.9 所示。图 18.7、图 18.8、图 18.9 分别表示没有填充位、向左对齐及向右对齐的情况，均为任意的例子。

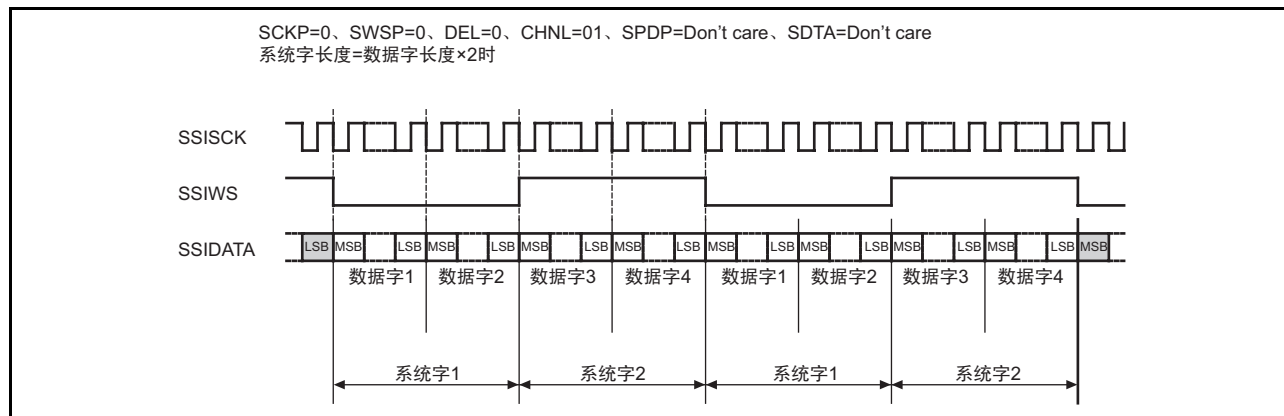


图 18.7 多通道格式 (4 个通道、无填充)

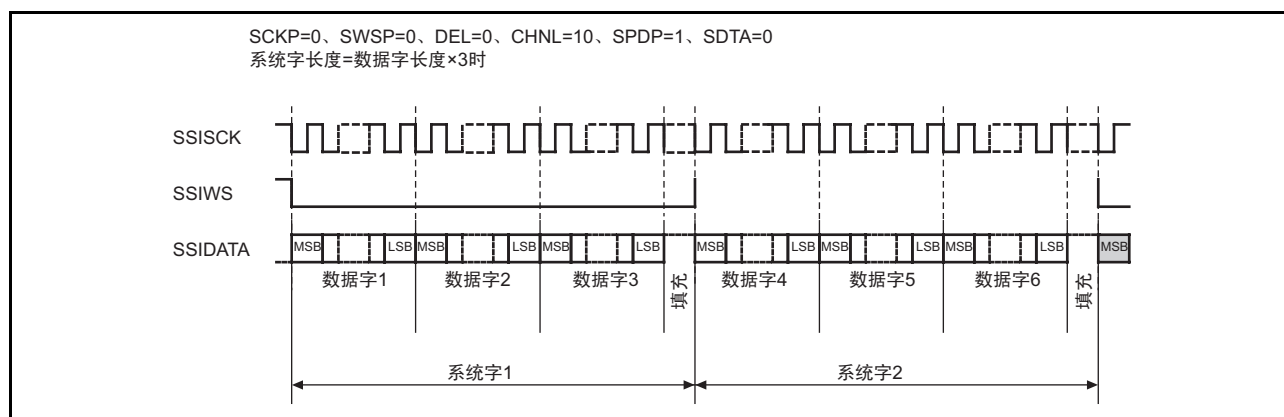


图 18.8 多通道格式 (6 个通道、High 填充)

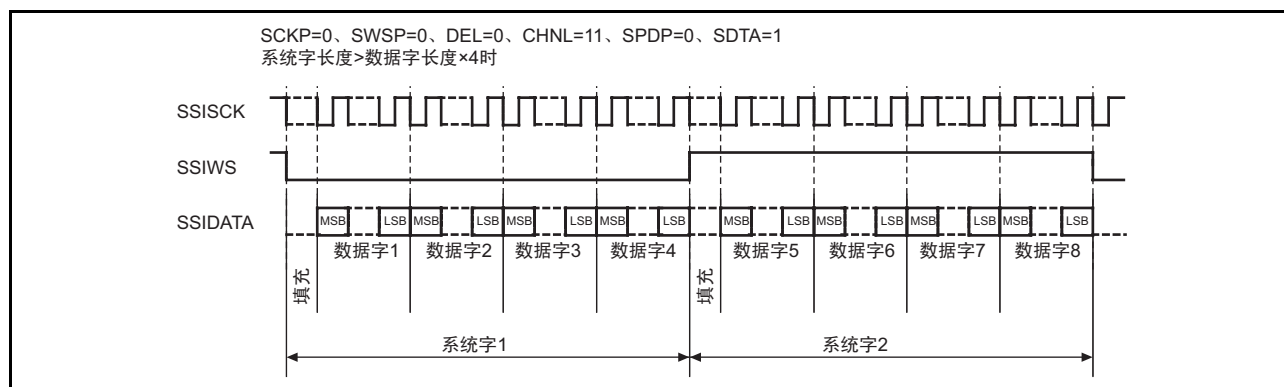


图 18.9 多通道格式
(8 个通道、按照填充位、串行数据的顺序发送 / 接收、有填充)

(7) 运行设定格式设定位

非压缩模式的其他结构位如下所示。这些位不相互排斥，但组合使用时，可能有的设定不实用。
参照图 18.10 的基本格式例，对这些结构位进行如下说明。

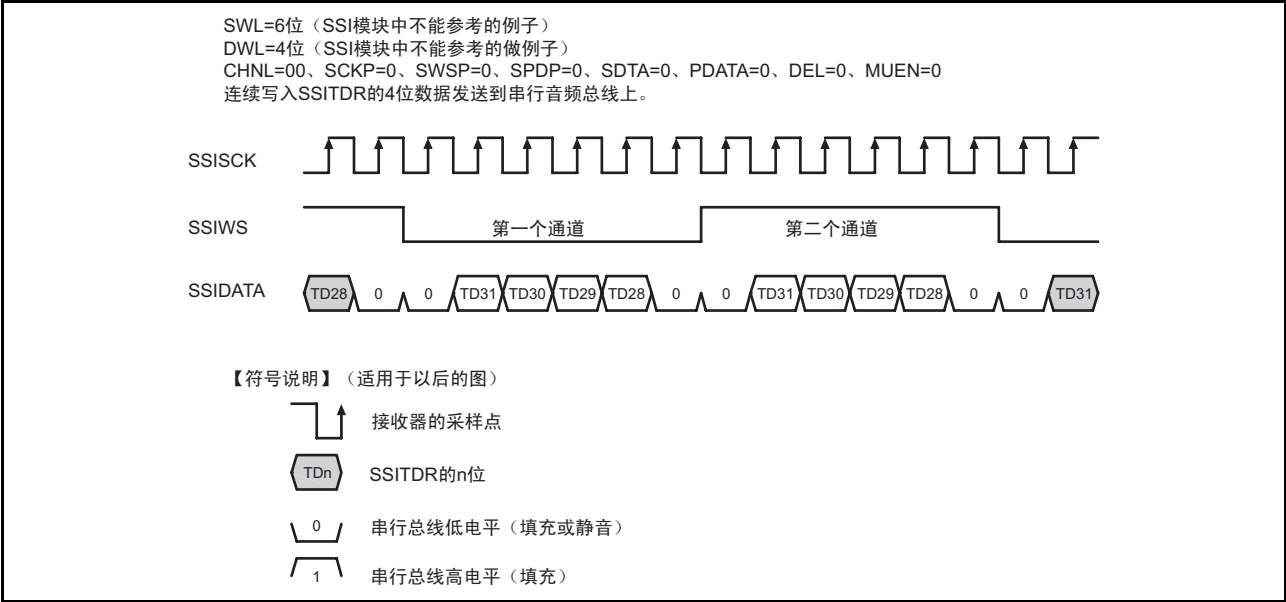


图 18.10 基本格式例 (发送模式、任意的系统 / 数据字长度)

在图 18.10，使用 6 位系统字与 4 位数据字。这些设定在 SSI 模块中无法实现，在此举例是为了说明其他的设定位。

- 取反时钟

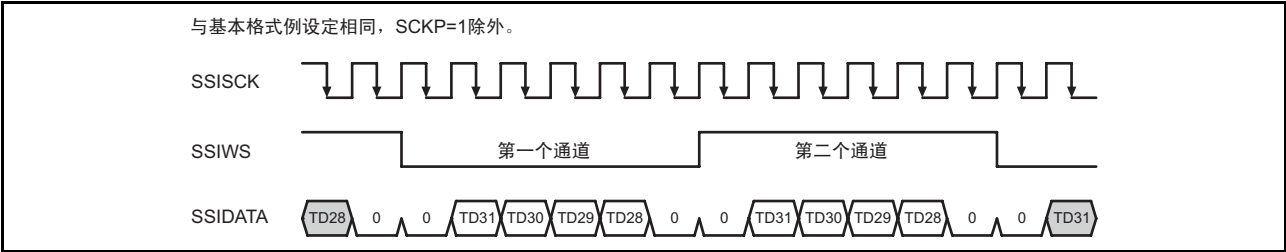


图 18.11 取反时钟

- 取反字选择信号

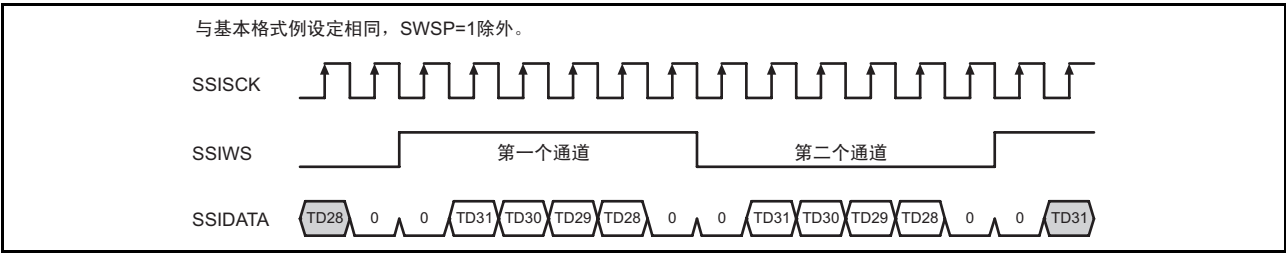


图 18.12 取反字选择信号

- 取反填充极性

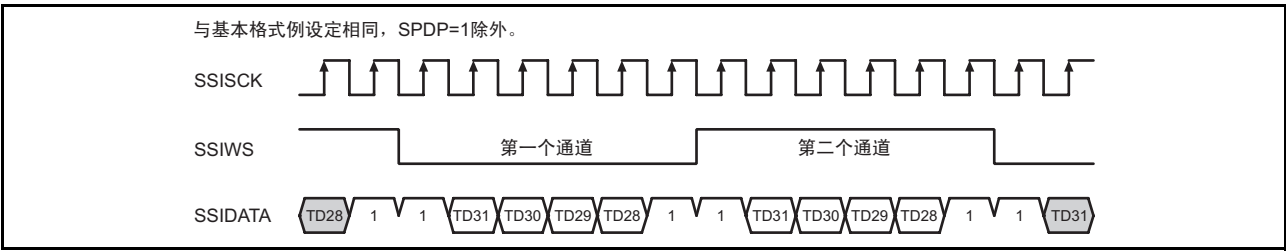


图 18.13 取反填充极性

- 按照填充位、串行数据的顺序发送/接收、有延迟

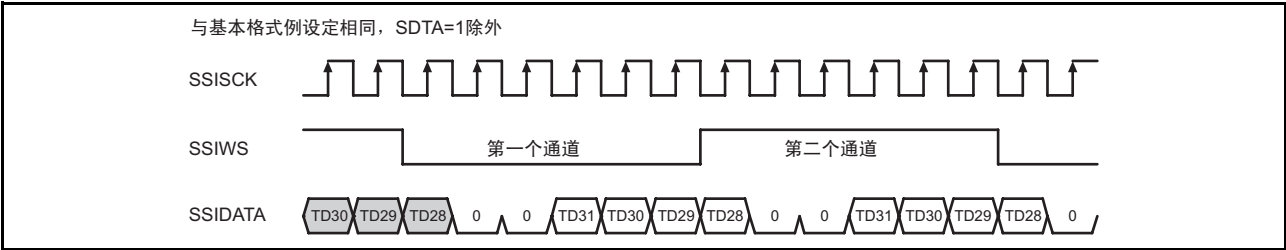


图 18.14 按照填充位、串行数据的顺序发送 / 接收、有延迟

- 按照填充位、串行数据的顺序发送/接收、无延迟

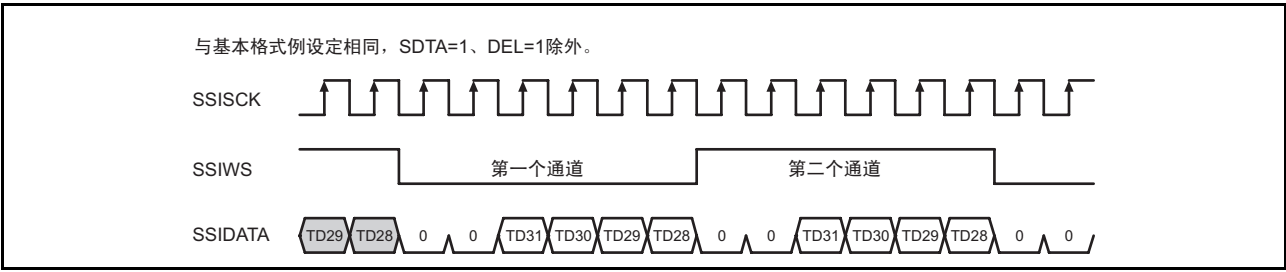


图 18.15 按照填充位、串行数据的顺序发送 / 接收、无延迟

- 按照串行数据、填充位的顺序发送/接收、无延迟

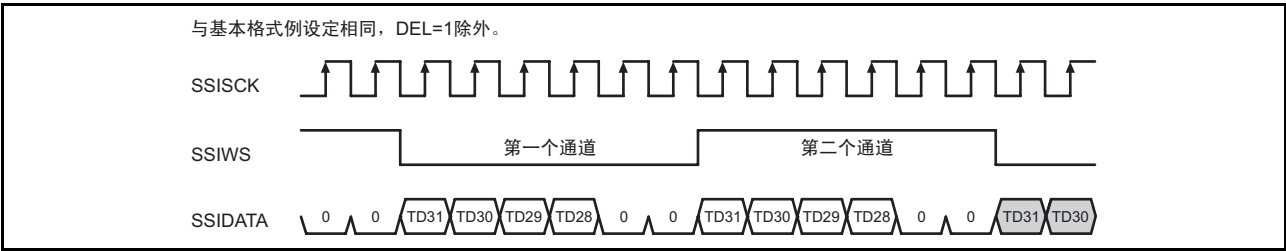


图 18.16 按照串行数据、填充位的顺序发送 / 接收、无延迟

- 向右对齐并行数据、有延迟

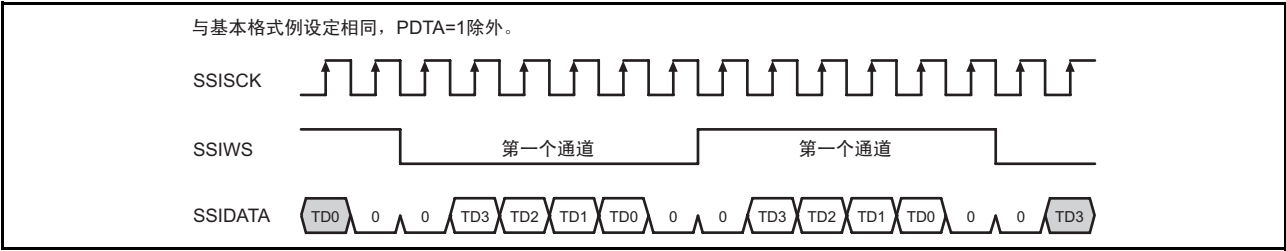


图 18.17 向右对齐并行数据，有延迟

- 静音有效

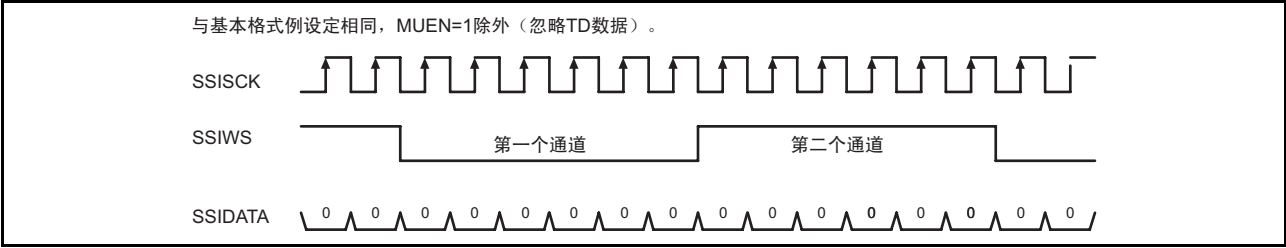


图 18.18 静音有效

18.4.3 运行模式

运行模式有结构、有效及无效等 3 种。运行模式的转移图如图 18.19 所示。

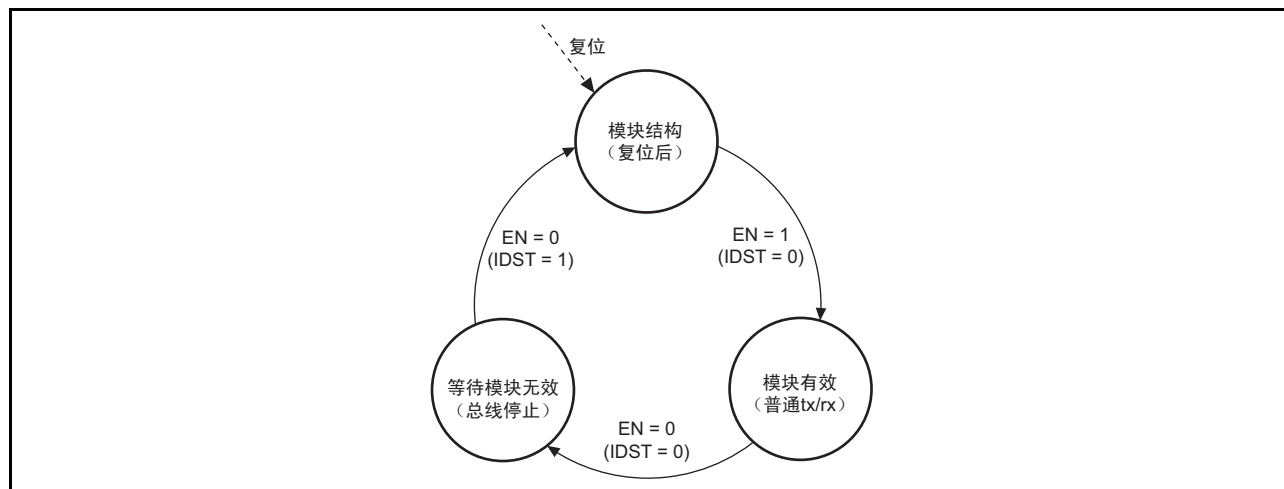


图 18.19 运行模式的转移图

(1) 结构模式

复位解除后进入该模式。SSI 模块在通过 EN 位置位有效前，必须在该模式对控制寄存器进行必要设定。如果 EN 置位，则 SSI 模块转移至模块有效模式。

(2) 模块有效模式

该模式的运行取决于所选择的运行模式。详情参阅“18.4.4 发送运行”及“18.4.5 接收运行”。

18.4.4 发送运行

可通过 DMA 或中断控制发送。

DMA 控制有利于减少 CPU 的负荷。在 DMA 控制模式，产生数据的下溢或上溢时，或 DMAC 的传送结束可通过中断得知。

另一种控制方法是根据需要，SSI 模块生成中断以提供数据。由于 SSI 模块仅为双缓冲结构，且至少需要在每个系统字写入数据，所以会在中断控制模式产生高负荷。

设定 SSI 模块无效时，必须持续提供 SSI 时钟 *，直到 IIRQ 位显示 SSI 为空闲状态。

DMA 控制模式的发送运行如图 18.20 所示，中断控制模式的发送运行如图 18.21 所示。

【注】 * SCKD=0 时，SSISCK 引脚的输入时钟
SCKD=1 时，过采样时钟

(1) 使用 DMA 控制器的发送

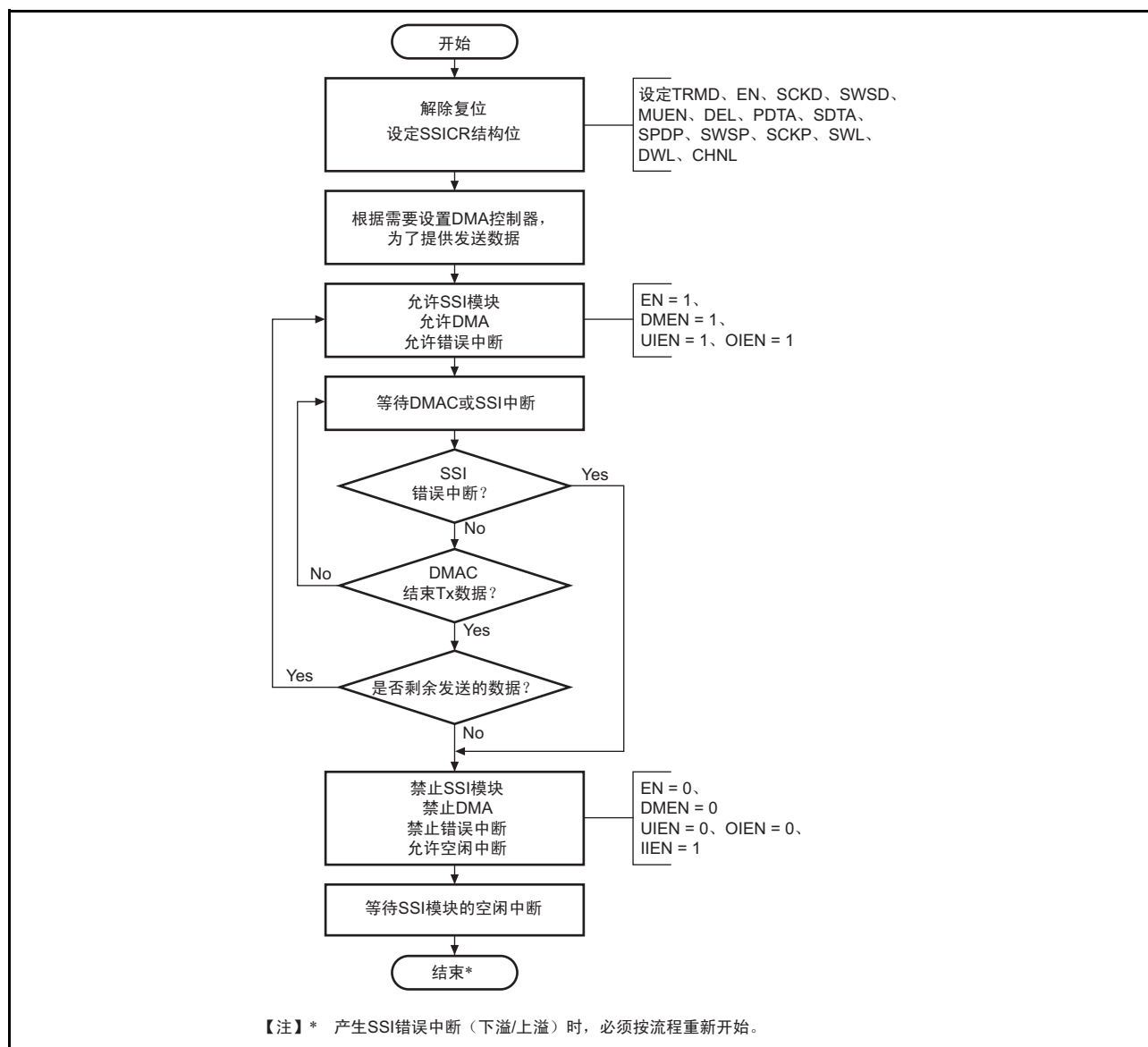


图 18.20 使用 DMA 控制器的发送

(2) 使用中断数据流程控制的发送

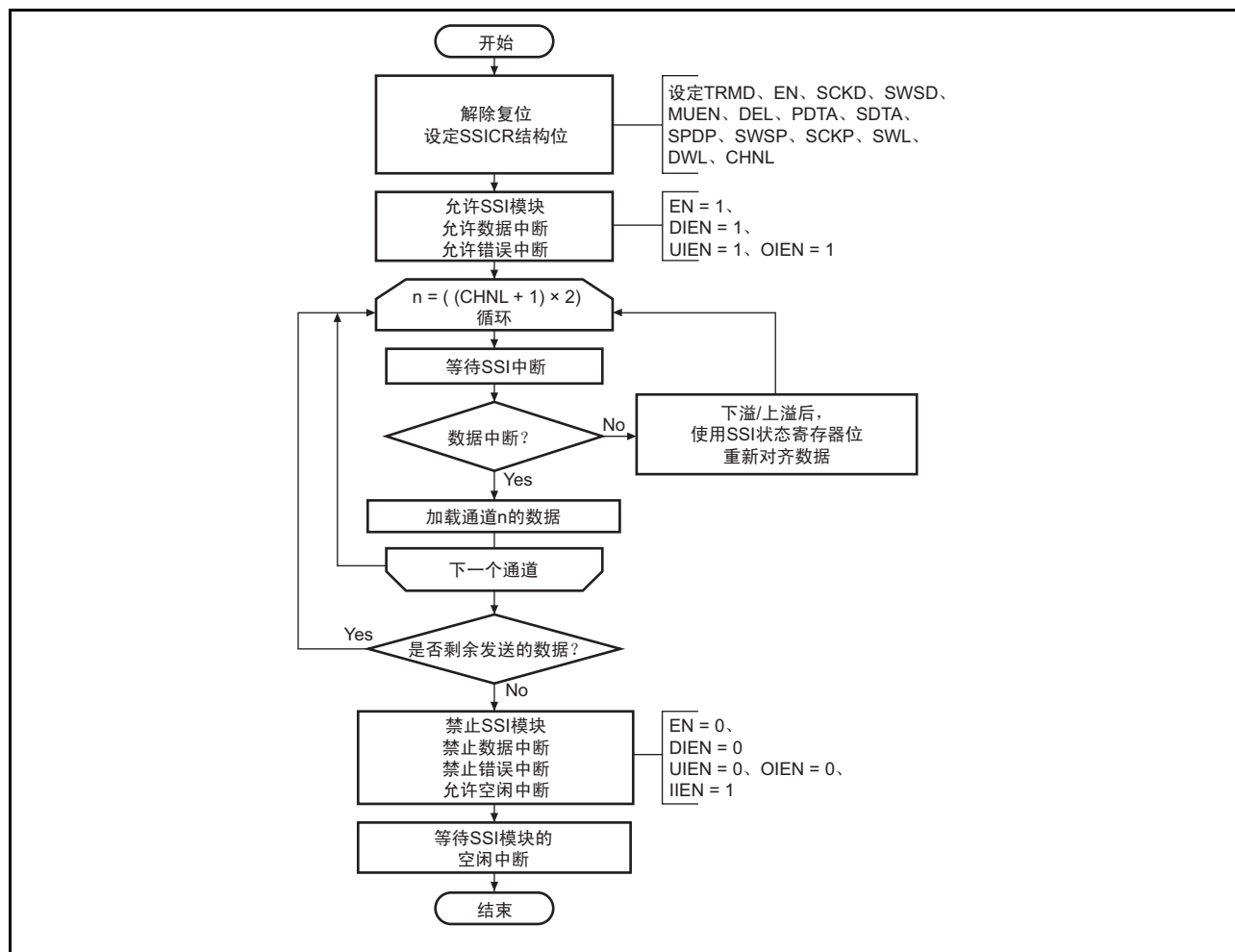


图 18.21 使用中断数据流程控制的发送

18.4.5 接收运行

与发送相同，可通过 DMA 或中断控制接收。

各运行的流程图分别如图 18.22、图 18.23 所示。

设定 SSI 模块无效时，必须持续提供 SSI 时钟 *，直到 IIRQ 位显示 SSI 为空闲状态。

【注】 * SCKD=0 时，SSISCK 引脚的输入时钟
SCKD=1 时，过采样时钟

(1) 使用 DMA 控制器的接收

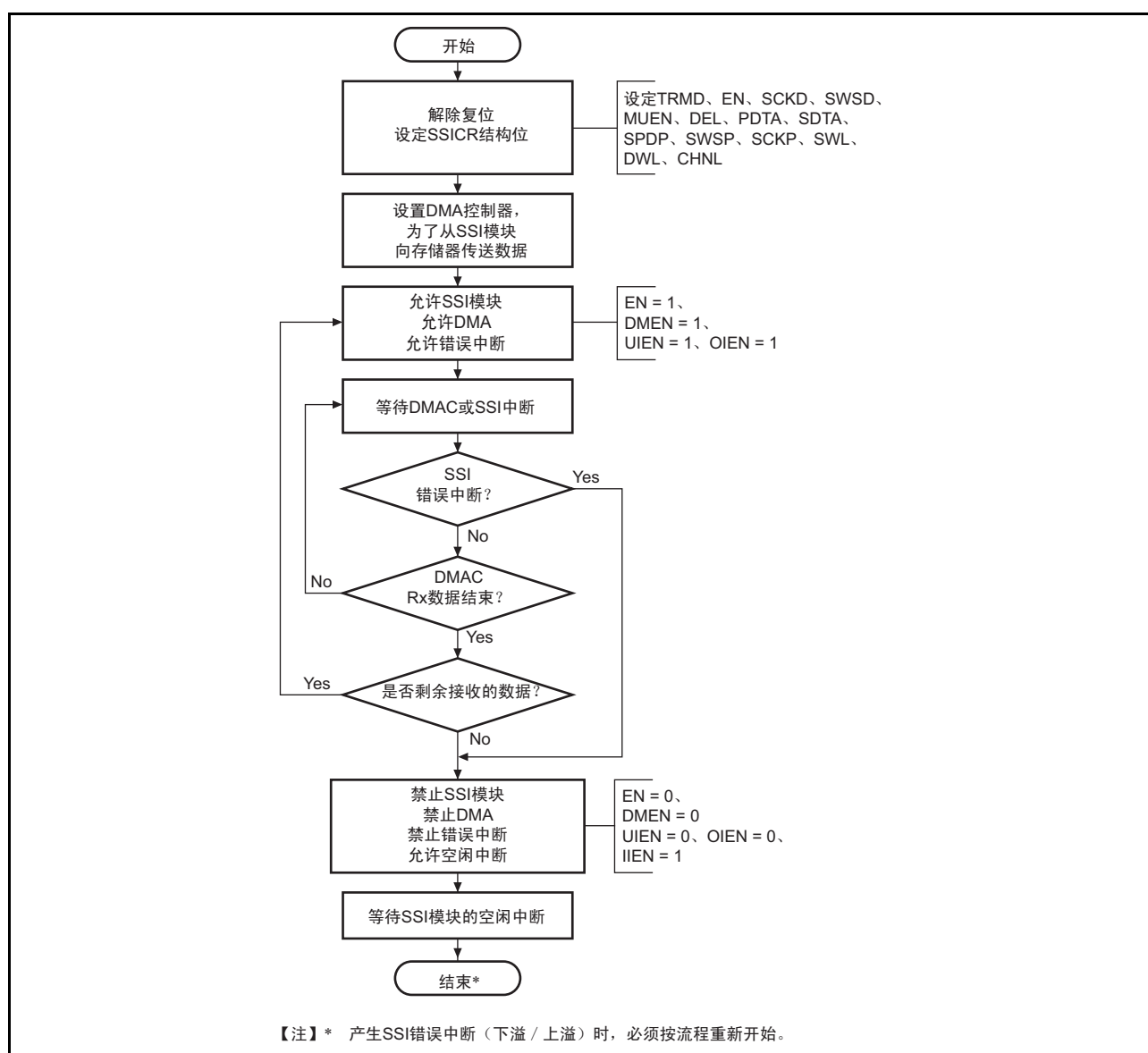


图 18.22 使用 DMA 控制器的接收

(2) 使用中断数据流程控制的接收

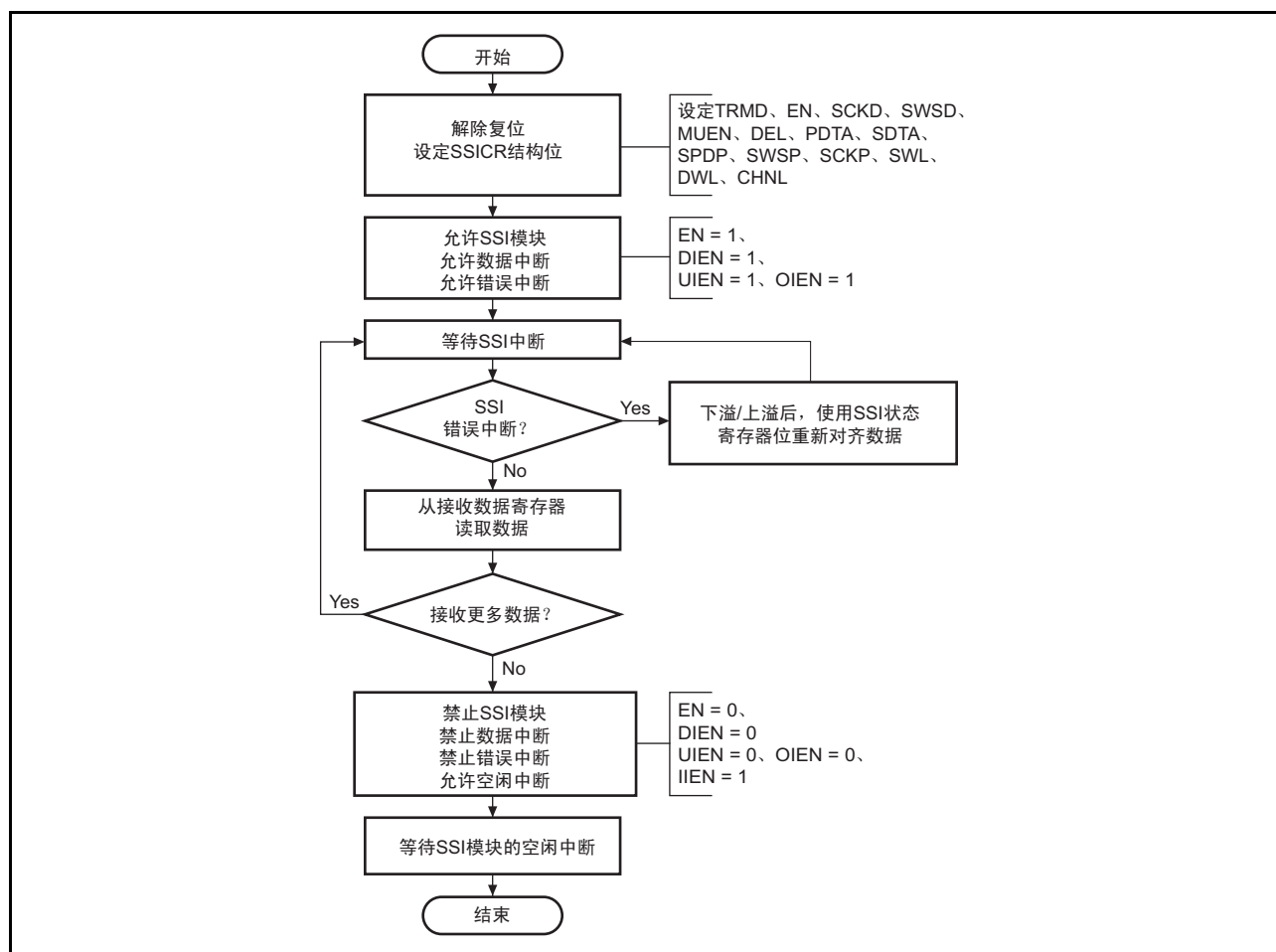


图 18.23 使用中断数据流控制的接收

下溢或上溢条件匹配时, 可使用 CHNO[1:0] 位与 SWNO 位使 SSI 模块恢复到匹配前的状态。如果产生下溢或上溢, 则可通过主机 CPU 读取通道数与系统字数, 得知串行音频流已到达的位置。作为发送器运行时, 在 SSI 模块到达下一次预计发送的数据前, 主机 CPU 可跳过发送数据。由此可与音频数据流再次同步。作为接收器运行时, SSI 模块在可保存表示下一次接收的数据前, 通过主机 CPU 保存无效数据, 得到接收数据的一致性, 与音频数据流再次同步。

18.4.6 发送时暂时停止、重新开始步骤

可通过以下步骤实现：

(1) 不重新设定 DMAC 而重复传送、停止的步骤

1. 为了停止DMA传送，设定SSICR.DMEN=0（禁止DMA请求）。
2. 通过查询或中断等，等待SSISR.DIRQ=1（发送模式：发送缓冲器空）。
3. SSICR.EN=0（禁止SSI模块运行）时，停止传送。
4. 重新开始传送前，确认SSISR.IDST=1。
5. 设定SSICR.EN=1（允许SSI模块运行）。
6. 通过查询或中断等，等待SSISR.DIRQ=1。
7. 通过设定SSICR.DMEN=1（允许DMA请求），可重新开始DMA传送。

(2) SSI 停止后，重新设定 DMAC 时的传送步骤

1. 为了停止DMA传送，设定SSICR.DMEN=0（禁止DMA请求）。
2. 通过查询或中断等，等待SSISR.DIRQ=1（发送模式：发送缓冲器空）。
3. SSICR.EN=0（禁止SSI模块运行）时，停止传送。
4. 通过DMAC的CHCR，停止DMAC。
5. 重新开始传送前，确认SSISR.IDST=1。
6. 设定SSICR.EN=1（允许SSI模块运行）。
7. 设定DMAC的各寄存器，并开始传送。
8. 通过设定SSICR.DMEN=1（允许DMA请求），重新开始DMA传送。

18.4.7 串行位时钟控制

使用串行位时钟功能，控制及选择用于串行总线接口的时钟。

串行位时钟的方向设定为输入（SCKD=0）时，SSI 模块为时钟从属模式，使用移位寄存器的位时钟为输入至 SSISCK 引脚的时钟。

串行位时钟的方向设定为输出（SCKD=1）时，SSI 模块为时钟主模式，使用移位寄存器的位时钟为过采样时钟或将位时钟分频的时钟。过采样时钟以通过 SSICR 的串行过采样时钟分频比（CKDV）位设定的比率分频后，用作移位寄存器的位时钟。

在上述任何情况，SSISCK 引脚的输出均与位时钟相同。

18.5 使用时的注意事项

18.5.1 接收 DMA 运行过程中产生上溢时的限制事项

在接收 DMA 运行过程中产生上溢时，需重新启动模块。SSI 内的接收缓冲器由 L 通道与 R 通道共用的 32 位寄存器构成。因此，将控制寄存器 (SSICR) 的数据字长度 (DWL2 ~ DWL0) 与系统字长度 (SWL2 ~ DWL0) 设定为 32 位时，如果产生上溢，则本应在 L 通道接收的数据可能会在 R 通道接收。

在这里，通过上溢错误中断或上溢错误状态标志（SSISR 的 OIRQ 位）确认上溢时，必须通过对 SSICR 的 EN 位及 DMEN 位写入 0，禁止 SSI 模块的 DMA，并停止运行（此时也必须停止设定 DMA 控制器）。此后，对 OIRQ 位写入 0，清除上溢状态，重新设定 DMA 后，重新开始传送。

第 19 章 控制器局域网 (RCAN-TL1)

控制器局域网 (RCAN-TL1:Renesas CAN Time Trigger Level 1) 是控制以汽车及工业器件系统等的实时通信为目的的 CAN (Controller Area Network) 的模块。

本章说明 RCAN-TL1 的编程接口。

有关 CAN 的数据链路控制器功能, 请参阅以下参考文献。

[参考文献]

1. CAN Specification Version 2.0 part A, Robert Bosch GmbH, 1991
2. CAN Specification Version 2.0 part B, Robert Bosch GmbH, 1991
3. Implementation Guide for the CAN Protocol, CAN Specification 2.0 Addendum, CAN In Automation, Erlangen, Germany, 1997
4. Road vehicles - Controller area network (CAN): Part 1: Data link layer and physical signaling (ISO-11898-1, 2003)
5. Road vehicles - Controller area network (CAN): Part 4: Time triggered communication (ISO-11898-4, 2004)

19.1 特点

19.1.1 RCAN-TL1 的特点

- 支持 CAN2.0B 标准
- 位时序依据 ISO-11898 标准
- 32 个邮箱
- 时钟频率: 16 ~ 33MHz
- 31 个可编程的发送/接收邮箱及 1 个接收邮箱
- 检测低功耗的 CAN 睡眠模式及 CAN 总线激活后, 自动解除 CAN 睡眠模式
- 所有邮箱均支持可编程接收过滤器屏蔽功能 (标准 ID 及扩展 ID)
- 最大 1Mbps 的可编程 CAN 数据速率
- 为了防止实时应用程序引起的优先顺序颠倒, 具备内部区分优先顺序的传送信息队列
- 丰富的中断源
- 内置测试功能 (只听模式、错误被动模式)
- 16 位自由运行定时器 (多种时钟源、预分频器、3 个定时器比较匹配寄存器)
- 用于时间触发发送的 6 位周期计数器 (Basic Cycle)
- 因定时器比较匹配寄存器产生的中断
- 定时器计数器可清除及设定
- 执行时间触发的寄存器: Local_Time、Cycle_time、Ref_Mark、Tx_Enable_Window、Ref_Trigger_Offset
- 发送/接收均支持在 SOF 的灵活时戳功能
- 不仅支持事件触发发送, 还支持时间触发发送及定期发送
- 可将周期计数器 (Basic Cycle) 嵌入 CAN 帧后发送

19.1.2 本单片机的特点

- 装载32个邮箱RCAN×2个通道 (RCAN0、1)
- 通过设定引脚功能控制器 (PFC)，RCAN的2个通道可如下连接使用
32个邮箱×2个通道 (RCAN0、1)
64个邮箱×1个通道 (RCAN0、1)
- RCAN0、1均可通过邮箱0的接收信息启动DMAC

【注】* 64个邮箱结构均有使用时的注意事项，详情参阅“19.12 使用时的注意事项”。

19.2 结构

19.2.1 框图

RCAN-TL1 提供了构成并控制 CAN 帧（支持 CAN2.0B Active 和 ISO-11898）的灵活且简练的方法。RCAN-TL1 在功能上由微处理器接口 (MPI)、邮箱、邮箱控制、定时器及 CAN 接口等 5 种块构成。RCAN-TL1 框图如图 19.1 所示。

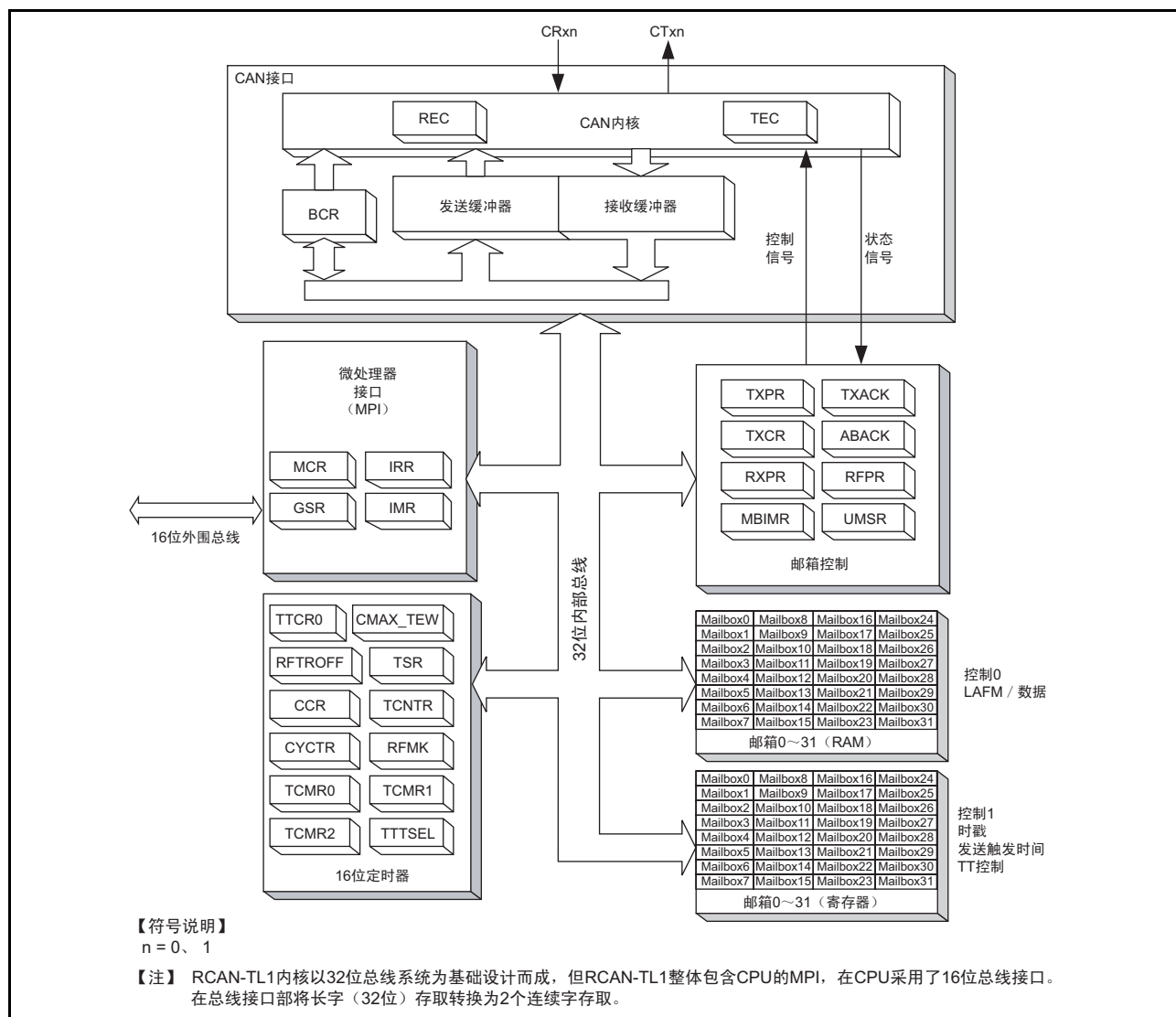


图 19.1 RCAN-TL1 框图 (1 个通道)

19.2.2 各块的功能

(1) 微处理器接口 (MPI)

MPI 使 CPU 与 RCAN-TL1 的寄存器或邮箱之间的通信成为可能，并控制存储器接口。MPI 具备检测 CAN 总线激活，并将其通知给 MPI 自身或 RCAN-TL1 的其他模块的唤醒控制逻辑，因此 RCAN-TL1 可自动解除 CAN 睡眠模式。MPI 包括 MCR、IRR、GSR 及 IMR 等寄存器。

(2) 邮箱

邮箱作为信息缓冲器排列于 RAM 及寄存器。

RAM 及寄存器内分别有 32 个邮箱，保存以下信息。

[RAM]

- CAN 信息控制 (ID、RTR、IDE 等)
- CAN 信息数据 (用于 CAN 数据帧)
- 用于接收的局部接收过滤器屏蔽 (LAFM)

[寄存器]

- CAN 信息控制 (DLC)
- 用于信息发送/接收的时戳
- 3 位邮箱结构、自动再发送无效位、用于远程请求的自动发送位、新信息控制位
- 发送触发时间

(3) 邮箱控制

邮箱控制有以下功能：

- 接收信息时比较 ID，生成将 CAN 接口的信息保存至邮箱的 RAM 地址和数据，并设置/清除对应的寄存器。
- 发送事件触发信息时，执行内部仲裁以选择正确优先顺序的信息，并将信息从邮箱加载至 CAN 接口的发送缓冲器。之后，设置或清除对应的寄存器。发送时间触发时，通过发送触发时间的比较匹配加载信息。
- 执行 CPU 与邮箱控制之间的邮箱存取仲裁。
- 寄存器有 TXPR、TXCR、TXACK、ABACK、RXPR、RFPR、MBIMR 及 UMSR。

(4) 定时器

本定时器是在特定时间范围发送信息、并记录结果的功能块。作为 16 位自由运行递增计数器，可由 CPU 控制。本定时器有 1 个与本地时间比较的 16 位比较匹配寄存器和 2 个与周期时间比较的比较匹配寄存器。这些比较匹配寄存器可产生中断信号、清除计数器。本定时器的时钟可从系统时钟生成的多个时钟周期选择，也可通过编程以 CAN 总线的 1 位时序计数。本定时器的寄存器有 TCNTR、TTCR0、CMAX_TEW、RETROFF、TSR、CCR、CYCTR、RFMK、TCMR0、TCMR1、TCMR2 及 TTTSEL。

(5) CAN 接口

本块支持参考文献 [2] 和 [4] 的 CAN 总线数据链路控制器规格，满足由 OSI 模型规定的链路控制器的全部功能，并且向 CAN 总线提供特殊的寄存器及逻辑。具体有接收错误计数器、发送错误计数器、位结构寄存器及各种测试模式。还有保存 CAN 数据链路控制器发送 / 接收的功能。

19.2.3 引脚结构

RCAN-TL1 的引脚结构如表 19.1 所示。

表 19.1 引脚结构

名称	引脚名称	输入 / 输出	功 能
发送数据引脚	CTxn	输出	CAN 总线发送引脚。
接收数据引脚	CRxn	输入	CAN 总线接收引脚。

【注】 n=0、1

19.2.4 存储器映射

RCAN-TL1 的存储器映射如图 19.2 所示。

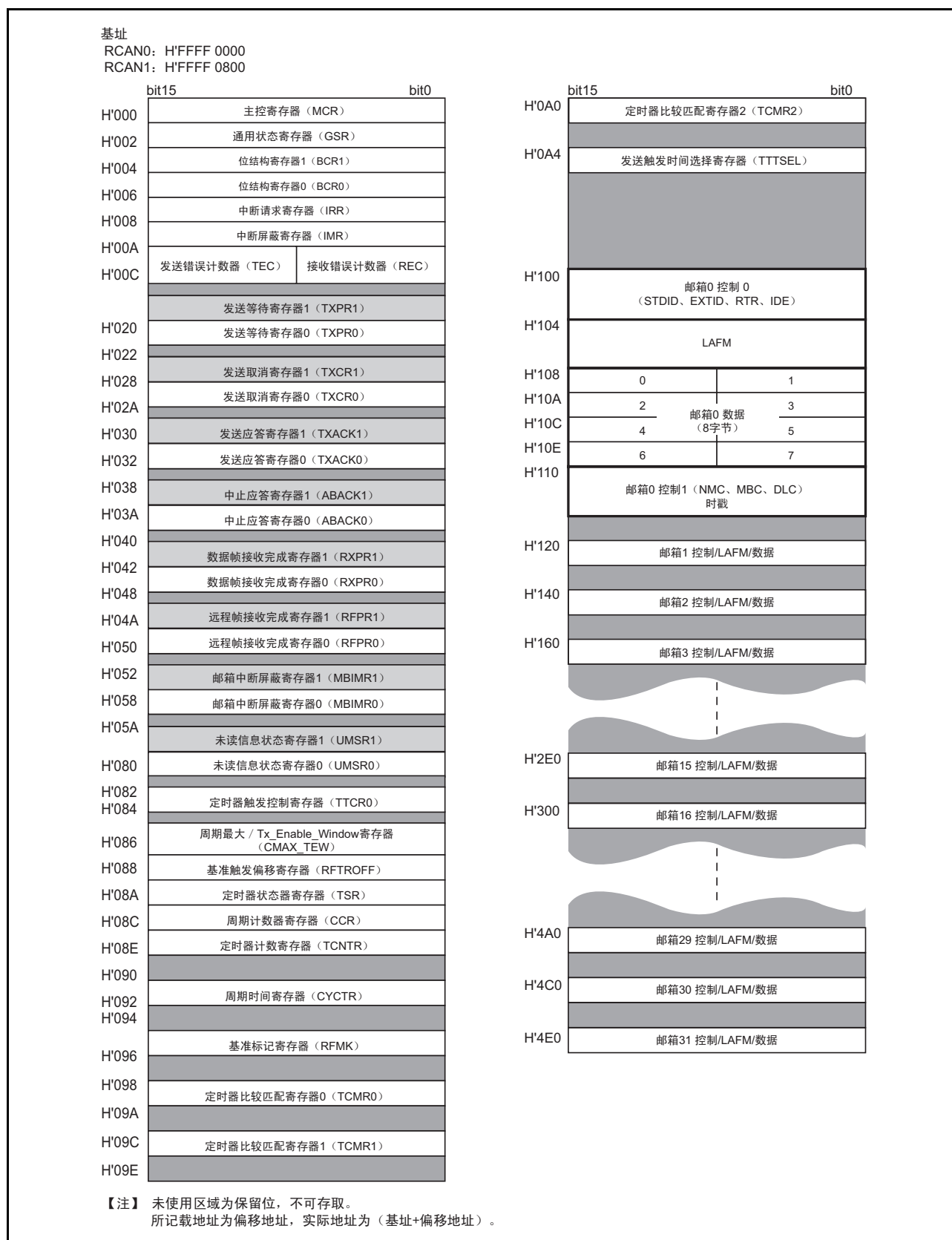


图 19.2 RCAN-TL1 的存储器映射（每 1 个通道）

19.3 邮箱

19.3.1 邮箱结构

邮箱用作发送 / 接收 CAN 帧的信息缓冲器。各邮箱由信息控制、局部接收过滤器屏蔽 (LAFM)、信息数据等 3 个保存字段构成。另外, 部分邮箱有时戳、时间触发结构及时间触发控制字段。

各邮箱的信息控制、LAFM、信息数据、时戳、发送触发时间及时时间触发控制的地址映射如表 19.2 所示。

表 19.2 各邮箱的地址映射 (每 1 个通道)

邮箱	地址						
	控制 0	LAFM	数据	控制 1	时戳	触发时间	TT 控制
	4 字节	4 字节	8 字节	2 字节	2 字节	2 字节	2 字节
0 (仅接收)	100 – 103	104 – 107	108 – 10F	110 – 111	112 – 113	—	—
1	120 – 123	124 – 127	128 – 12F	130 – 131	132 – 133	—	—
2	140 – 143	144 – 147	148 – 14F	150 – 151	152 – 153	—	—
3	160 – 163	164 – 167	168 – 16F	170 – 171	172 – 173	—	—
4	180 – 183	184 – 187	188 – 18F	190 – 191	192 – 193	—	—
5	1A0 – 1A3	1A4 – 1A7	1A8 – 1AF	1B0 – 1B1	1B2 – 1B3	—	—
6	1C0 – 1C3	1C4 – 1C7	1C8 – 1CF	1D0 – 1D1	1D2 – 1D3	—	—
7	1E0 – 1E3	1E4 – 1E7	1E8 – 1EF	1F0 – 1F1	1F2 – 1F3	—	—
8	200 – 203	204 – 207	208 – 20F	210 – 211	212 – 213	—	—
9	220 – 223	224 – 227	228 – 22F	230 – 231	232 – 233	—	—
10	240 – 243	244 – 247	248 – 24F	250 – 251	252 – 253	—	—
11	260 – 263	264 – 267	268 – 26F	270 – 271	272 – 273	—	—
12	280 – 283	284 – 287	288 – 28F	290 – 291	292 – 293	—	—
13	2A0 – 2A3	2A4 – 2A7	2A8 – 2AF	2B0 – 2B1	2B2 – 2B3	—	—
14	2C0 – 2C3	2C4 – 2C7	2C8 – 2CF	2D0 – 2D1	2D2 – 2D3	—	—
15	2E0 – 2E3	2E4 – 2E7	2E8 – 2EF	2F0 – 2F1	2F2 – 2F3	—	—
16	300 – 303	304 – 307	308 – 30F	310 – 311	—	—	—
17	320 – 323	324 – 327	328 – 32F	330 – 331	—	—	—
18	340 – 343	344 – 347	348 – 34F	350 – 351	—	—	—
19	360 – 363	364 – 367	368 – 36F	370 – 371	—	—	—
20	380 – 383	384 – 387	388 – 38F	390 – 391	—	—	—
21	3A0 – 3A3	3A4 – 3A7	3A8 – 3AF	3B0 – 3B1	—	—	—
22	3C0 – 3C3	3C4 – 3C7	3C8 – 3CF	3D0 – 3D1	—	—	—

邮箱	地址						
	控制 0	LAFM	数据	控制 1	时戳	触发时间	TT 控制
	4 字节	4 字节	8 字节	2 字节	2 字节	2 字节	2 字节
23	3E0 – 3E3	3E4 – 3E7	3E8 – 3EF	3F0 – 3F1	—	—	—
24	400 – 403	404 – 407	408 – 40F	410 – 411	—	414 – 415	416 – 417
25	420 – 423	424 – 427	428 – 42F	430 – 431	—	434 – 435	436 – 437
26	440 – 443	444 – 447	448 – 44F	450 – 451	—	454 – 455	456 – 457
27	460 – 463	464 – 467	468 – 46F	470 – 471	—	474 – 475	476 – 477
28	480 – 483	484 – 487	488 – 48F	490 – 491	—	494 – 495	496 – 497
29	4A0 – 4A3	4A4 – 4A7	4A8 – 4AF	4B0 – 4B1	—	4B4 – 4B5	4B6 – 4B7
30	4C0 – 4C3	4C4 – 4C7	4C8 – 4CF	4D0 – 4D1	4D2–4D3 (局部时间)	4D4 – 4D5	—
31	4E0 – 4E3	4E4 – 4E7	4E8 – 4EF	4F0 – 4F1	4F2–4F3 (局部时间)	—	—

邮箱 0 为接收专用。其他所有邮箱通过设定信息控制的 MBC（邮箱结构）位可发送、接收。邮箱结构详情如图 19.3 ~ 图 19.5 所示。

表 19.3 邮箱的作用

	事件触发		时间触发		备注	
	发送	接收	发送	接收	时戳寄存器	发送触发时间寄存器
MB31	可设定	可设定	—	接收时间基准	有	—
MB30	可设定	可设定	时间主模式时发送时间基准	在时间从属模式接收	有	有
MB29 ~ MB24	可设定	可设定	OK	OK	—	有
MB23 ~ MB16	可设定	可设定	— (ET)	OK	—	—
MB15 ~ MB1	可设定	可设定	— (ET)	OK	有	—
MB0	—	可设定	—	OK	有	—

【注】 ET: 在时间触发模式的合并仲裁窗内, 可执行发送。

• MB23~16 (邮箱: 无时戳)

地址	数据总线																存取长度	字段名称
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
H'100+N*32	IDE	RTR	0	STDID[10:0]											EXTID[17:16]		16/32	控制0
H'102+N*32	EXTID[15:0]																16	
H'104+N*32	IDE_LAFM	0	0	STDID_LAFM[10:0]											EXTID_LAFM[17:16]		16/32	LAFM
H'106+N*32	EXTID_LAFM[15:0]																16	
H'108+N*32	MSG_DATA_0（最初的Rx/Tx字节）								MSG_DATA_1								8/16/32	数据
H'10A+N*32	MSG_DATA_2								MSG_DATA_3								8/16	
H'10C+N*32	MSG_DATA_4								MSG_DATA_5								8/16/32	
H'10E+N*32	MSG_DATA_6								MSG_DATA_7								8/16	
H'110+N*32	0	0	NMC	ATX	DART	MBC[2:0]			0	0	0	0	DLC[3:0]			8/16	控制1	

• MB29~24 (在时间触发器模式, 发送时间触发器)

地址	数据总线																存取长度	字段名称	
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
H'100+N*32	IDE	RTR	0	STDID[10:0]												EXTID[17:16]		16/32	控制0
H'102+N*32	EXTID[15:0]																16		
H'104+N*32	IDE_LAFM	0	0	STDID_LAFM[10:0]												EXTID_LAFM[17:16]		16/32	LAFM
H'106+N*32	EXTID_LAFM[15:0]																16		
H'108+N*32	MSG_DATA_0（最初的Rx/Tx 字节）								MSG_DATA_1								8/16/32	数据	
H'10A+N*32	MSG_DATA_2								MSG_DATA_3								8/16		
H'10C+N*32	MSG_DATA_4								MSG_DATA_5								8/16/32		
H'10E+N*32	MSG_DATA_6								MSG_DATA_7								8/16		
H'110+N*32	0	0	NMC	ATX	DART	MBC[2:0]			0	0	0	0	DLC[3:0]			8/16	控制1		
H'112+N*32	保留位																—	—	
H'114+N*32	发送触发时间（TTT）																16	触发器时间	
H'116+N*32	TTW[1:0]		偏移						0	0	0	0	0	Rep Factor			16	TT控制	

【注】1. 灰色部分表示的位为保留位。写入值必须为0。读取值未必为0, 且不能保证。
2. MCR15位的初始值为1, 因此, 信息控制及LAFM的STDID、RTR、IDE、EXTID的顺序与HCAN2不同。

图 19.4 邮箱 (16 ~ 29) 的结构

• MB30 (在时间触发模式, 发送时间基准)

地址	数据总线																存取长度	字段名称	
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
H'100+N*32	IDE	RTR	0	STDID[10:0]												EXTID[17:16]		16/32	控制0
H'102+N*32	EXTID[15:0]																16		
H'104+N*32	IDE_LAFM	0	0	STDID_LAFM[10:0]												EXTID_LAFM[17:16]		16/32	LAFM
H'106+N*32	EXTID_LAFM[15:0]																16		
H'108+N*32	MSG_DATA_0（最初的Rx/Tx字节）								MSG_DATA_1								8/16/32	数据	
H'10A+N*32	MSG_DATA_2								MSG_DATA_3								8/16		
H'10C+N*32	MSG_DATA_4								MSG_DATA_5								8/16/32		
H'10E+N*32	MSG_DATA_6								MSG_DATA_7								8/16		
H'110+N*32	0	0	NMC	ATX	DART	MBC[2:0]			0	0	0	0	DLC[3:0]			8/16	控制1		
H'112+N*32	TimeStamp[15:0]（SOF中的TCNTR值）																16	时戳	
H'114+N*32	作为时间基准的发送触发时间（TTT）																16	触发时间	

• MB31 (在时间触发模式, 接收时间基准)

地址	数据总线																存取长度	字段名称
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
H'100+N*32	IDE	RTR	0	STDID[10:0]										EXTID[17:16]			16/32	控制0
H'102+N*32	EXTID[15:0]																16	
H'104+N*32	IDE_LAFM	0	0	STDID_LAFM[10:0]										EXTID_LAFM[17:16]			16/32	LAFM
H'106+N*32	EXTID_LAFM[15:0]																16	
H'108+N*32	MSG_DATA_0（最初的Rx/Tx字节）									MSG_DATA_1							8/16/32	数据
H'10A+N*32	MSG_DATA_2									MSG_DATA_3							8/16	
H'10C+N*32	MSG_DATA_4									MSG_DATA_5							8/16/32	
H'10E+N*32	MSG_DATA_6									MSG_DATA_7							8/16	
H'110+N*32	0	0	NMC	ATX	DART	MBC[2:0]			0	0	0	0	DLC[3:0]			8/16	控制1	
H'112+N*32	TimeStamp[15:0]（SOF中的TCNTR值）																16	时戳

【注】1. 灰色部分表示的位为保留位。写入值必须为0。读取值未必为0, 且不能保证。
2. MCR15位的初始值为1, 因此信息控制及LAFM的STDID、RTR、IDE、EXTID顺序与HCAN2不同。

图 19.5 邮箱 (30、31) 的结构

19.3.2 信息控制字段

寄存器名称	地址	位	位名称	说 明
MB[x]. CONTROL0H	H'100+N*32	15	IDE	ID 扩展 区分 CAN 数据帧和远程帧为标准格式或扩展格式。 0: 标准格式 1: 扩展格式
		14	RTR	远程发送请求 区分数据帧和远程帧。通过数据帧或远程帧, 接收 CAN 帧改写本位。 【重点】MBC=B'001 时, 置位数据帧自动发送 (ATX) 位, 则 RTR 位不可置位。接收远程帧时, 通过对应的 RFPR 位或 IRR2 (远程帧接收中断) 位, 可通知 CPU, 但 RCAN-TL1 需将当前信息作为数据帧发送, 因此 RTR 位不变。 【重点】MBC=B'001 时, 设定 ATX=1, 接收远程帧的邮箱自动进行发送设定。此时为了发送数据帧, RTR 必须设定为 0。 0: 数据帧 1: 远程帧
		13	—	保留位 初始值不定, 写入值总是为 0。
		12 ~ 2	STDID [10:0]	标准 ID 设定数据帧和远程帧的 ID (标准 ID) 的位。
		1、0	EXTID [17:16]	扩展 ID 设定数据帧和远程帧的 ID (扩展 ID) 的位。
MB[x]. CONTROL0L	H'102+N*32	15 ~ 0	EXTID [15:0]	

• 邮箱 0

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	NMC	0	0	MBC[2:0]			0	0	0	0	DLC[3:0]			
初始值:	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R	R	R/W	R	R/W	R	R	R	R	R/W	R/W	R/W	R/W

【注】MBC [1] 的值总是为 1。

• 邮箱 1 ~ 31

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	NMC	ATX	DART	MBC[2:0]			0	0	0	0	DLC[3:0]			
初始值:	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

寄存器名称	地址	位	位名称	说 明
MB[x]. CONTROL1	H'110+N*32	15、14	—	保留位 读取值、写入值总是为 0。
		13	NMC	新信息控制 此位置 0 时，已设置 RXPR 或 RFPR 的邮箱不保存新信息，仍保持之前的信息，并置位 UMSR 的对应位。 此位置 1 时，已设置 RXPR 或 RFPR 的邮箱重写新信息，并置位 UMSR 的对应位。 【重点】如果以数据帧改写远程帧，或相反的情况时，RXPR 及 RFPR 标志（与 UMSR 同时）可能在相同邮箱置位。此时邮箱控制字段内的 RTR 位也被重写。 【重点】使用时间触发模式时，邮箱 31 的 NMC 必须置 1，即使未清除 RXPR[31]，也可接收所有的基准信息。 0：溢出模式 1：重写模式
MB[x]. CONTROL1	H'110+N*32	12	ATX	数据帧自动发送 本位置 1 时，如果在邮箱接收远程帧，则保存 DLC，TXPR 自动置位。使用被改写的 DLC，从相同邮箱自动发送数据帧。设定为自动发送的邮箱，根据信息发送优先权位（MCR2）的设定，调度为 ID 优先顺序或邮箱优先顺序。要使用本功能，必须将 MBC[2:0] 设定为 B'001。使用本功能发送时，所使用的数据长度代码（DLC）为已接收的数据长度代码。实际应用时，远程帧的 DLC 必须与请求的数据帧的 DLC 对应。 【重点】1. 使用 ATX 且 MBC 为 B'001 时，远程帧的 ID 与响应信息时一样，必须与数据帧的 ID 完全相同，因此不可使用 IDE 位的过滤器。 2. 使用本功能时，无论是否接收远程帧，RTR 位都不置位。接收远程帧时，由已置位的 RFPR 通知 CPU，但 RCAN-TL1 必须将当前信息作为数据帧发送，因此 RTR 位不变。 3. 有时不能开始远程帧的自动发送，必须注意溢出状态（NMC=0 时 UMSR 置位）。 4. 设定为 ATX=1 的邮箱因远程帧而成为溢出状态时，有时会接受旧信息的自动发送请求。 0：数据帧的自动发送无效 1：数据帧的自动发送有效 【注】 本位仅存在于邮箱 1～31，在邮箱 0 为保留位。
		11	DART	自动再发送无效 本位置 1 时，产生 CAN 总线错误事件或 CAN 总线仲裁失败，信息自动再发送无效。实际使用本功能，在最初发送时，对应的 TXCR 位自动置位。本位置 0 时，RCAN-TL1 由 TXCR 取消发送请求或请求发送直至发送正常结束。 0：再发送有效 1：再发送无效 【注】 本位仅存在于邮箱 1～31，在邮箱 0 为保留位。

寄存器名称	地址	位	位名称	说 明
MB[x]. CONTROL1	H'110+N*32	10 ~ 8	MBC[2:0]	邮箱结构 这些位在各邮箱的功能设定如表 19.4。MBC 为 B'111 时，邮箱未激活。即，与 TXPR 或其他任何设定无关，邮箱不发送 / 接收信息。 禁止设定 MBC 为 B'100、B'101、B'110。MBC 设定为 B'000 或除此之外的值时，可以使用 LAFM 字段。 MBC 设定为接收时，禁止设置 TXPR。无硬件保护功能，TXPR 保持置位。邮箱 0 的 MBC[1] 为接收专用，因此由硬件固定为 1。
		7 ~ 4	—	保留位 读取值、写入值总是为 0。
		3 ~ 0	DLC[3:0]	数据长度代码 编码以数据帧发送的数据的字节数 (0 ~ 8)。 0000: 数据长度 0 字节 0001: 数据长度 1 字节 0010: 数据长度 2 字节 0011: 数据长度 3 字节 0100: 数据长度 4 字节 0101: 数据长度 5 字节 0110: 数据长度 6 字节 0111: 数据长度 7 字节 1xxx: 数据长度 8 字节 【注】 x: Don't care

表 19.4 邮箱功能的设定

MBC2	MBC1	MBC0	发送数据帧	发送远程帧	接收数据帧	接收远程帧	说明
0	0	0	可	可	不可	不可	- 邮箱 0 不可使用 - 发送时间触发可使用
0	0	1	可	可	不可	可	- 可通过 ATX 使用 * - 邮箱 0 不可使用 - LAFM 可使用
0	1	0	不可	不可	可	可	- 邮箱 0 可使用 - LAFM 可使用
0	1	1	不可	不可	可	不可	- 邮箱 0 可使用 - LAFM 可使用
1	0	0	禁止设定				
1	0	1	禁止设定				
1	1	0	禁止设定				
1	1	1	邮箱停止 (初始值)				

【注】 * 为了支持自动再发送，MBC 为 B'001 且 ATX 为 1 时，RTR 必须设定为 0。ATX 设定为 1 时，禁止使用 IDE 过滤器。

19.3.3 局部接收过滤器屏蔽（LAFM）

MBC 值为 B'001、B'010、B'011 时，该字段作为接收用的 LAFM。LAFM 允许邮箱接受多个接收 ID。如图 19.6 所示，LAFM 由 2 个 16 位可读取 / 写入区域构成。

寄存器名称	地址	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	存取长度	字段名称	
MB[0].LAFMH	H'104+N*32	IDE LAFM	0	0	STDID_LAFM[10:0]												EXTID LAFM[17:16]		16/32	LAFM
MB[0].LAFML	H'106+N*32	EXTID_LAFM[15:0]																16		

图 19.6 局部接收过滤器屏蔽（LAFM）

如果 1 个位设定为 LAFM，通过与 RCAN-TL1 匹配的 CAN-ID 搜索邮箱时，忽略已接收 CAN 的 ID 的对应位。清除位时，已接收 CAN 的 ID 的对应位必须与保存的邮箱设定的 STDID/IDE/EXTID 匹配。LAFM 的结构与邮箱的信息控制相同。不用该功能时，必须为 0。

- 【注】
1. RCAN-TL1 从邮箱 31 至邮箱 0 搜索匹配的 ID。RCAN-TL1 检测出匹配的 ID 时，无论其信息为 NMC 还是 RXPR/RFPD 标志，都立即结束搜索。这表示即使使用 LAFM，接收信息也仅保存至 1 个邮箱。
 2. 接收一条信息且找到匹配的邮箱时，信息保存至邮箱。使用 LAFM 时，STDID、RTR、IDE、EXTID 更新为已接收信息的 STDID、RTR、IDE、EXTID，因此可能与接收前设定的不同。

寄存器名称	地址	位	位名称	说 明
MB[x].LAFMH	H'104+N*32	15	IDE_LAFM	IDE 位的过滤器屏蔽位 0: 对应的 IDE 位有效 1: 对应的 IDE 位无效
		14、13	—	保留位 初始值不定，写入值为 0。
		12 ~ 2	STDID_LAFM [10:0]	STDID[10:0] 的过滤器屏蔽位 0: 对应的 STDID 位有效 1: 对应的 STDID 位无效
		1、0	EXTID_LAFM [17:16]	EXTID[17:0] 的过滤器屏蔽位 0: 对应的 EXTID 位有效 1: 对应的 EXTID 位无效
MB[x].LAFML	H'106+N*32	15 ~ 0	EXTID_LAFM [15:0]	

19.3.4 信息数据字段

保存发送 / 接收的 CAN 信息。MSG_DATA_0 对应发送 / 接收的第一个数据字节。CAN 总线的位排列顺序为 bit7 至 bit0。

• 发送时间触发时的注意事项

通过 CMAX≠B'111、MBC[30]=B'000 将 TXPR[30] 设定为 1 时，邮箱 30 设定为发送时间基准。此时，DLC 必须设定为大于 0 的值，并且将 RTR 设定为 0（根据 TTCAN Level 1 的规定）。邮箱 30 开始发送时，将周期计数器值（CCR）嵌入数据字段的起始字节 MSG_DATA_0[5:0] 后再发送信息。

MSG_DATA_0[7:6] 发送保存在邮箱的值。

需发送 Next_is_Gap 时，用户可将 MSG_DATA_0[7] 设定为 1。

请注意此时 CCR 的值插入可发送的帧，并不保存在邮箱 30 的信息数据字段。

设定 CMAX≠B'111、MBC[31]=B'011 时，邮箱 31 设定为接收时间基准。接收有效基准信息时（ID 匹配、DLC > 0），RCAN-TL1 通过改写 RFMK 及周期计数器（CCR），使 Cycle_Time 与 Basic_Cycle 同步。



图 19.7 信息数据字段

19.3.5 时戳

保存发送 / 接收信息时记录的时戳。时戳用于监控是否按计划发送 / 接收信息。

- 时戳

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TS15	TS14	TS13	TS12	TS11	TS10	TS9	TS8	TS7	TS6	TS5	TS4	TS3	TS2	TS1	TS0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

(1) 接收信息

通过设定时间触发控制寄存器 0 (TTCR0) 的 bit14, 由接收信息的 SOF 捕捉 CYCTR[15:0] (周期时间寄存器) 的值或 CCR[5:0] (周期计数器) + CYCTR[15:6] (周期时间寄存器) 的值, 并保存至邮箱 0 ~ 15 的接收信息的时戳。

由接收信息的 SOF 捕捉定时器计数器寄存器 (TCNTR) 的值, 保存至邮箱 30、31 的接收信息的时戳。

(2) 发送信息

通过设定时间触发控制寄存器 0 (TTCR0) 的 bit14, 由发送信息的 SOF 捕捉 CYCTR[15:0] (周期时间寄存器) 的值或 CCR[5:0] (周期计数器) + CYCTR[15:6] (周期时间寄存器) 的值, 并保存至邮箱 1 ~ 15 的发送信息的时戳。

由发送信息的 SOF 捕捉定时器计数器寄存器 (TCNTR) 的值, 并保存至邮箱 30、31 的发送信息的时戳。

【重点】时戳保存在暂存器。正常发送或接收时, 其值复制至邮箱字段。并且, 由于溢出, 在 UMSR[N] 置位的同时, CPU 清除 RXPR[N]/RFPR[N] 后, 可能仅更新时戳。因此, 可读取清除 RXPR[N]/RFPR[N] 前的正确时戳值。

19.3.6 发送触发时间（TTT）与时间触发控制

邮箱 24 ~ 29 中，在时间触发模式（CMA_X≠B'111）将 MBC 设定为 B'000 时，发送触发时间有决定时间窗间边界的 Time_Mark 功能。

发送触发时间（TTT）与时间触发控制（TT 控制）由以下 2 个 16 位读取 / 写入寄存器构成。

邮箱 30 没有时间触发控制（TT 控制），而有 Time_Ref 功能。

邮箱 24 ~ 30 不用于时间触发模式发送时，可用作接收邮箱。但使用时间触发模式时，不会成为事件触发发送的对象。

- 发送触发时间（TTT）

发送触发时间（TTT）指定在周期时间内开始发送信息的时间。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TTT15	TTT14	TTT13	TTT12	TTT11	TTT10	TTT9	TTT8	TTT7	TTT6	TTT5	TTT4	TTT3	TTT2	TTT1	TTT0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- 时间触发控制（TT 控制）

时间触发控制（TT 控制）指定时间窗的属性、开始发送的系统矩阵内的周期计数（Basic Cycle）及定期发送的频率。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TTW[1:0]		Offset[5:0]						0	0	0	0	0	rep_factor[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

支持时间触发模式的所有邮箱之间的差异如图 19.8 所示。

MB29~24																	
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
H'114+N*32	发送触发时间（周期时间）															16位	触发时间
H'116+N*32	TTW[1:0]		Offset[5:0]					0	0	0	0	0	rep_factor[2:0]		16位	TT控制	

MB30																	
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
H'114+N*32	发送触发时间（周期时间）															16位	触发时间

图 19.8 发送触发控制字段

- TTW[1:0]（时间触发窗）

表示时间窗的属性。通过设定 TTW=B'10 开始的合并仲裁窗（Merged Arbitrating Window），必须设定 TTW=B'11 结束。合并仲裁窗开始至结束期间，可使用多个 TTW=B'10 设定的信息。

TTW[1]	TTW[0]	说 明
0	0	独占窗 : exclusive window（初始值）
0	1	仲裁窗 : arbitrating window
1	0	开始合并仲裁窗 : Start of Merged arbitrating window
1	1	结束合并仲裁窗 : End of Merged arbitrating window

TTT 寄存器的值与 CYCTR 的值匹配、Offset 的值与 CCR 的值匹配时，从对应邮箱尝试发送。为了使该功能有效，必须设定 CMAX≠B'111、运行（TTCR0 的 bit15 为 1）定时器（TCNTR）、设定对应邮箱的 MBC=B'000、对应的 TXPR 位设定为 1。由软件置位 TXPR 位后，为了持续定期发送，RCAN-TL1 不清除 TXPR 位（邮箱 24 ~ 30）。

为了停止定期发送，必须由 TXCR 清除 TXPR。此时，发送完成后如果未立即清除 TXACK，则同一邮箱的 TXACK 和 ABACK 可能同时置位。（参照图 19.9）

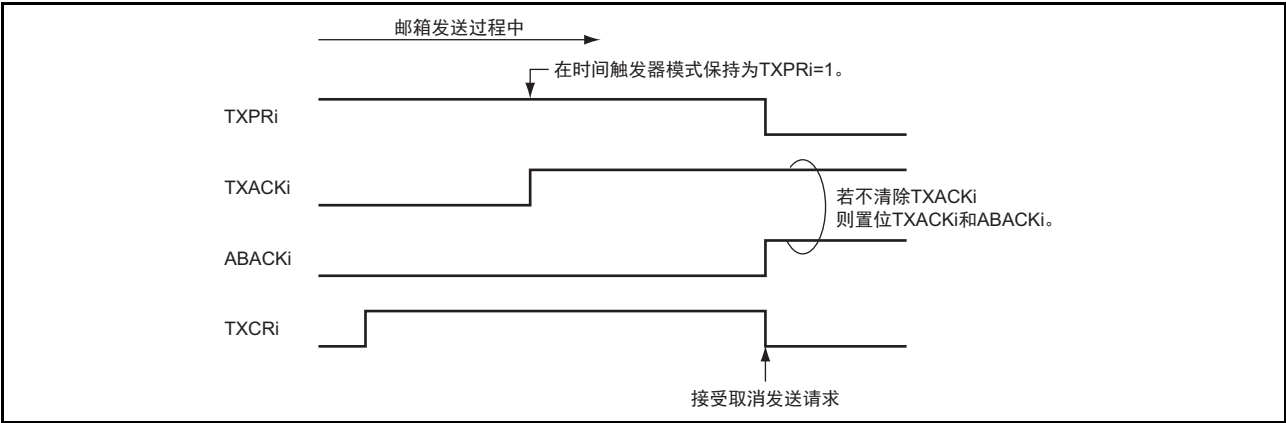


图 19.9 发送时间触发时的 TXACK 和 ABACK

另外，在邮箱 30，固定为 TTW=B'01、Offset =B'000000、rep_factor =B'000。rep_factor 与 Offset 的组合如下表所示：

rep_factor	说 明
B'000	各周期计数（初始值）
B'001	每 2 个周期计数
B'010	每 4 个周期计数
B'011	每 8 个周期计数
B'100	每 16 个周期计数
B'101	每 32 个周期计数
B'110	每 64 个周期计数（每个系统矩阵 1 次）
B'111	保留位

Offset 字段决定时间触发邮箱可开始发送信息的第 1 个周期计数器值。

Offset	说 明
B'000000	偏移（Offset）= 第 1 个周期计数（初始值）
B'000001	偏移（Offset）= 第 2 个周期计数
B'000010	偏移（Offset）= 第 3 个周期计数
B'000011	偏移（Offset）= 第 4 个周期计数
B'000100	偏移（Offset）= 第 5 个周期计数
...	...
B'111110	偏移（Offset）= 第 63 个周期计数
B'111111	偏移（Offset）= 第 64 个周期计数

必须满足以下条件：
 $Cycle_Count_Maximum + 1 \geq Repeat_Factor > Offset$
 $Cycle_Count_Maximum = 2^{C_{MAX}} - 1$
 $Repeat_Factor = 2^{rep_factor}$

C_{MAX}、Repeat_Factor 及 Offset 为寄存器值。

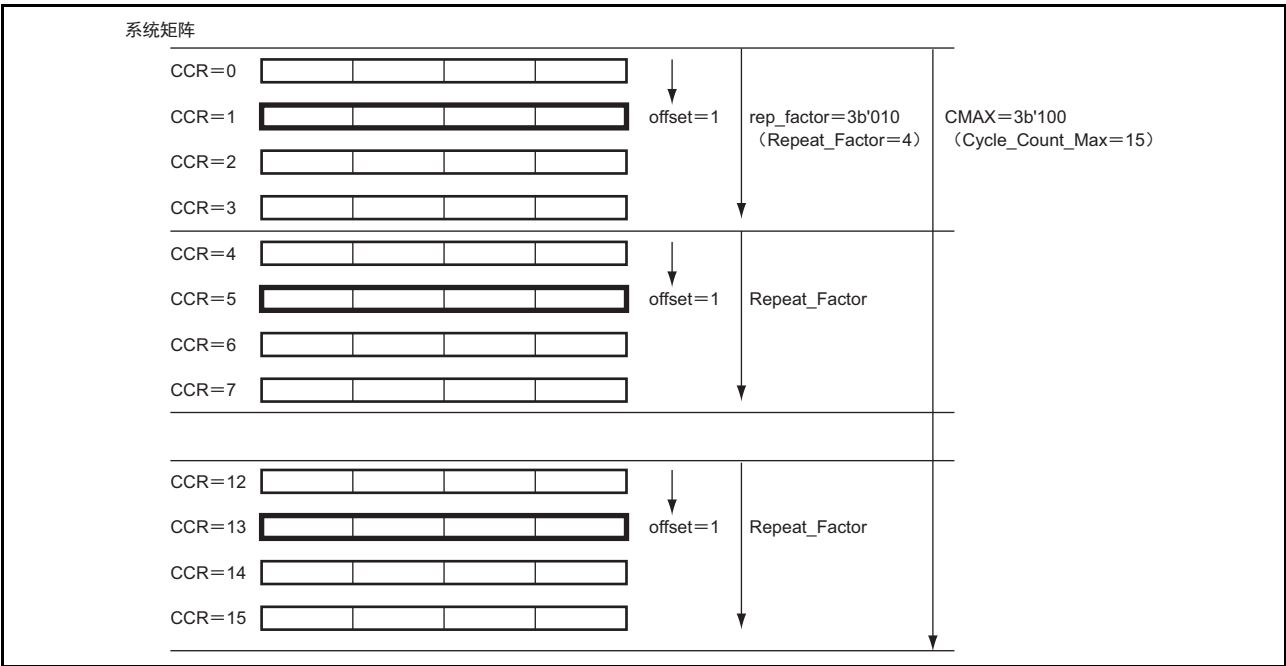


图 19.10 系统矩阵

发送触发时间必须设定为升序，必须确保发送触发时间之间的差为最小 TEW 宽度。

19.4 RCAN-TL1 的控制寄存器

以下说明 RCAN-TL1 的控制寄存器，本寄存器仅能以字长度（16 位）存取。

RCAN-TL1 的控制寄存器如表 19.5 所示。

表 19.5 RCAN-TL1 的控制寄存器的结构

寄存器名称	简称	地址	存取长度（位）
主控寄存器	MCR	H'000	16
通用状态寄存器	GSR	H'002	16
位结构寄存器 1	BCR1	H'004	16
位结构寄存器 0	BCR0	H'006	16
中断请求寄存器	IRR	H'008	16
中断屏蔽寄存器	IMR	H'00A	16
发送错误计数器 / 接收错误计数器	TEC/REC	H'00C	16

19.4.1 主控寄存器（MCR）

MCR 为控制 RCAN-TL1 的 16 位可读取 / 写入的寄存器。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MCR15	MCR14	—	—	—	TST[2:0]		MCR7	MCR6	MCR5	—	—	MCR2	MCR1	MCR0	
初始值:	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15	MCR15	1	R/W	ID 排序 本位可设定与 HCAN2 兼容的信息控制及 LAFM 的 STDID、RTR、IDE、EXTID 的顺序，仅可在复位模式更改。有关 ID 排序的顺序，请参照图 19.11。 0: RCAN-TL1 与 HCAN2 顺序相同 1: RCAN-TL1 与 HCAN2 顺序不同
14	MCR14	0	R/W	总线断开自动停机 同时设定本位和 MCR6 时，如果 RCAN-TL1 为总线断开状态，则 MCR1 立即自动置位。本位仅可在复位模式更改。 0: 在通常的返回顺序（128×11 个隐性位），RCAN-TL1 保持总线断开状态。 1: MCR6 置位时，RCAN-TL1 在总线断开状态后，立即进入停机模式。
13 ~ 11	—	均为 0	R	保留位。 读取值、写入值总是为 0。

位	位名称	初始值	R/W	说 明
10 ~ 8	TST[2:0]	000	R/W	测试模式 本位设定测试模式有效 / 无效。启动测试模式前，请注意必须将 RCAN-TL1 设定为停机模式或复位模式。这样可避免向测试模式的转移影响正在执行的发送 / 接收。详情参阅 “19.7.2 测试模式的设定”。 测试模式仅用于诊断及测试，RCAN-TL1 通常运行时，不可使用该模式。 000: 普通模式 001: 只听模式（接收专用模式） 010: 自测试模式 1（外部） 011: 自测试模式 2（内部） 100: 写入错误计数器 101: 错误被动模式 110: 禁止设定 111: 禁止设定
7	MCR7	0	R/W	自动唤醒模式 本位设定 CAN 睡眠模式的自动唤醒模式有效 / 无效。设定本位时，RCAN-TL1 检测出 CAN 总线激活（显性位）后，自动解除 CAN 睡眠模式（MCR5）。如果清除本位，则 RCAN-TL1 不自动解除 CAN 睡眠模式。RCAN-TL1 不可保存将其唤醒的信息。 0: 通过 CAN 总线激活的自动唤醒模式无效 1: 通过 CAN 总线激活的自动唤醒模式有效 【注】 在 CAN 睡眠模式不可更改 MCR7 位。
6	MCR6	0	R/W	总线断开时停机 本位设定在总线断开时 MCR1 置位后，立即进入停机模式有效或无效。仅可由复位或停机模式更改本位。注意如果总线断开时进入停机模式，则 CAN 控制器也会立即返回错误主动模式。 0: 总线断开时不进入停机模式，等待返回顺序结束 1: 总线断开时根据 MCR1 的设定，使转移至停机模式有效
5	MCR5	0	R/W	CAN 睡眠模式 本位设定向 CAN 睡眠模式的转移有效 / 无效。如果 RCAN-TL1 为停机模式且本位置位，则向 CAN 睡眠模式的转移有效。进入停机模式后，允许设定本位。2 个错误计数器（REC、TEC）在 CAN 睡眠模式不变化。解除 CAN 睡眠模式有 2 种方法。 - 向 MCR5 位写入 0 - MCR7 有效时，检测 CAN 总线上的显性位 如果自动睡眠解除模式无效，则在结束 CAN 睡眠模式前，RCAN-TL1 忽略所有的 CAN 总线运行。解除 CAN 睡眠模式时，RCAN-TL1 在 CAN 总线运行前，通过检测 11 个隐性位，可与 CAN 总线同步。这意味着如果使用第 2 种方法，则 RCAN-TL1 不可接收第 1 个信息。同样，CAN 收发器在待机模式结束时，也不可接收第 1 个信息，因此必须考虑该方法设计软件。 在 CAN 睡眠模式仅可存取 MCR、GSR、IRR、IMR 寄存器，详情参阅 “19.7.1(3) CAN 睡眠模式”。 0: 解除 CAN 睡眠模式 1: 向 CAN 睡眠模式的转移有效 【注】 先设定为停机模式，RCAN-TL1 可转移至睡眠模式。因此，转移至 CAN 睡眠模式前，CPU 可清除等待状态的所有中断。清除完所有中断的同时，RCAN-TL1 必须从停机模式转移至 CAN 睡眠模式（具体是同时向 MCR5 写入 1、向 MCR1 写入 0）。

位	位名称	初始值	R/W	说 明
4、3	—	均为 0	R	保留位 读取值、写入值总是为 0。
2	MCR2	0	R/W	信息发送优先权 本位选择等待发送数据的顺序。置 1 时，按照发送等待寄存器 (TXPR) 位的顺序发送数据。 邮箱 31 最先开始发送，并持续发送到邮箱 1 (邮箱设定为用于发送时)。 注意本功能不可用于邮箱 24 ~ 30 的时间触发发送。如果清除本位，则所有的发送信息 (由于启动内部仲裁) 均按照 ID 优先顺序发送。最先的信息有最小数字的仲裁字段 (STDID + IDE 位 + EXTID (IDE = 1 时) + RTR 位)，可首先发送。 内部仲裁包括 RTR 位及 IDE 位 (内部仲裁方法与同时开始转移 2 个 CAN 节点之间的 CAN 总线仲裁时的方法相同)。 仅可在复位或停机模式更改本位。 0: 按照信息 ID 优先顺序发送 1: 按照邮箱编号顺序 (邮箱 31→ 邮箱 1) 发送
1	MCR1	0	R/W	停机请求 设置本位时，CAN 控制器结束当前运行后进入停机模式 (在此从 CAN 总线切断)。清除本位前，RCAN-TL 保持停机模式。 在停机模式，CAN 接口与 CAN 总线运行无关，也不保存或发送信息。除向 CPU 通知停机状态的 IRR0 及 GSR4 之外，用户寄存器 (包括邮箱内容及 TEC/REC) 的内容保持不变。 CAN 总线为空闲或间歇状态时，与 MCR6 无关，RCAN-TL1 在 1 位时间内为停机模式。设置 MCR6 时，总线断开时的停机请求也在 1 位时间内运行。此外，在总线断开返回顺序结束前，不进入停机模式。由 IRR0 及 GSR4 通知进入停机模式。 如果设置 MCR14 及 MCR6，则 RCAN-TL1 转移为总线断开状态之后，本位立即自动设置。 RCAN-TL1 在停机模式与总线运行无关，因此除了设定位时序之外，可更改其结构。再次加入 CAN 总线运行则需要将本位清 0。清除后，RCAN-TL1 等待检测出 11 个隐性位后，加入 CAN 总线。 0: 清除停机模式请求 1: 停机模式转移请求 【注】 1. 发行停机请求后，在完成向停机模式转移之前 (由 IRR0 及 GSR4 通知)，CPU 不可存取 TXPR 和 TXCR 及清除本位。设置 MCR1 后，仅可在进入停机模式后或通过复位 (软件或硬件) 解除该状态。 2. 仅在 BCR1 及 BCR0 寄存器设定了适当的波特率时，可向停机模或从该模式转移。

位	位名称	初始值	R/W	说 明
0	MCR0	1	R/W	复位请求 本位控制 RCAN-TL1 模块的复位。从 0 变为 1 时，RCAN-TL1 控制器进入复位程序，初始化内部逻辑，设置 GSR3 及 IRR0 以通知复位模式。所有的用户寄存器均被初始化。 设置本位期间，可再次配置 RCAN-TL1。加入 CAN 总线则必须对本位写入 0 后清除。清除后，RCAN-TL1 模块等待检测出 11 个隐性位后，加入 CAN 总线。 采样 CAN 总线上的值则必须设置适当的波特率值。上电复位后，总是设置该位与 GSR3。表示请求复位，需要配置 RCAN-TL1。 复位请求相当于上电复位，但由软件控制。 0：清除复位模式请求 [清除条件] 复位 RCAN-TL1 后写入 0 时 1：CAN 接口的复位模式转移请求

• MCR15 (ID排序) = 0

地址	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	存取长度	字段名称
H'100+N*32	0	STDID[10:0]											RTR	IDE	EXTID[17:16]		16/32	控制0
H'102+N*32	EXTID[15:0]															16		
H'104+N*32	0	STDID_LAFM[10:0]											0	IDE_LAFM	EXTID_LAFM[17:16]		16/32	LAFM字段
H'106+N*32	EXTID_LAFM[15:0]															16		

• MCR15 (ID排序) = 1

地址	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	存取长度	字段名称
H'100+N*32	IED	RTR	0	STDID[10:0]										EXTID[17:16]		16/32	控制0	
H'102+N*32	EXTID[15:0]															16		
H'104+N*32	IDE_LAFM	0	0	STDID_LAFM[10:0]										EXTID_LAFM[17:16]		16/32	LAFM字段	
H'106+N*32	EXTID_LAFM[15:0]															16		

图 19.11 ID 排序

19.4.2 通用状态寄存器 (GSR)

GSR 为 16 位只读寄存器，表示 RCAN-TL1 的状态。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	GSR5	GSR4	GSR3	GSR2	GSR1	GSR0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名称	初始值	R/W	说 明
15 ~ 6	—	均为 0	R	保留位 读取值、写入值总是为 0。
5	GSR5	0	R	错误被动状态 表示 CAN 接口是否为错误被动。RCAN-TL1 为错误被动状态时，本位立即置 1，再次为该状态时则清除。说明在错误被动及总线断开时，GSR5 保持为 1。因此，为了得知正确的状态，必须检查 GSR5 及 GSR0 双方的状态。 0: RCAN-TL1 为非错误被动或总线断开状态 [清除条件] RCAN-TL1 为错误主动状态 1: RCAN-TL1 为错误被动 (但 GSR0=0 时) 或总线断开 (但 GSR0=1 时) [置位条件] TEC ≥ 128 或 REC ≥ 128 或测试模式时选择错误被动模式
4	GSR4	0	R	停机 / 睡眠状态 表示 CAN 控制器是否为停机 / 睡眠状态。注意本标志的清除时间与 IRR12 的设定时间不同。该标志反映 CAN 控制器的状态，不能全部反映 RCAN-TL1 的状态。如果结束 CAN 睡眠模式、清除 MCR5，则 RCAN-TL1 可存取。经过传送时钟的 2 个位之后，CAN 控制器的 CAN 睡眠模式结束。 0: RCAN-TL1 既非停机模式，也非 /CAN 睡眠模式 1: 停机模式 (MCR1=1 时) 或 CAN 睡眠模式 (MCR5=1 时) [置位条件] MCR1 置位、CAN 总线为间歇 / 空闲时，或 MCR5 置位、RCAN-TL1 为停机模式时，或 MCR14 与 MCR6 均置位、RCAN-TL1 向总线断开转移时。
3	GSR3	1	R	复位状态 表示 RCAN-TL1 是否为复位状态。 0: RCAN-TL1 为非复位状态 1: RCAN-TL1 为复位状态 [置位条件] RCAN-TL1 软件或硬件复位后

位	位名称	初始值	R/W	说 明
2	GSR2	1	R	正在发送信息标志 本标志向 CPU 显示 RCAN-TL1 是总线断开状态，还是正在发送信息或是正在发送因发送过程中检测出的错误源而导致错误 / 超载的标志。设置 TXACK 的时序与清除 GSR2 的时序不同。在 EOF 的第 7 个位设定 TXACK；无等待发送的信息时，在间歇的第 3 个位设定 GSR2。另外即使仲裁失败、总线空闲、接收、复位或停机转移也可设定 GSR2。 0: RCAN-TL1 为总线断开状态或正在发送信息 1: [置位条件] 非总线断开状态或未发送信息
1	GSR1	0	R	发送 / 接收警告标志 表示错误警告的标志。 0: [清除条件] $TEC < 96$ 且 $REC < 96$ 或总线断开时 1: [置位条件] $96 \leq TEC < 256$ 或 $96 \leq REC < 256$ 时 【注】 为了计数总线断开返回顺序所需 11 个隐性位的重复次数，在总线断开时增加 REC，但不置位本位。
0	GSR0	0	R	总线断开标志 表示 RCAN-TL1 为总线断开状态。 0: [清除条件] 从总线断开状态返回或硬件 / 软件复位后 1: [置位条件] $TEC \geq 256$ （总线断开状态）

19.4.3 位结构寄存器 0、1（BCR0、BCR1）

BCR0、BCR1 均为 16 位可读取 / 写入的寄存器，设定 CAN 位时序参数及 CAN 接口的波特率预分频器。按照下面的方法定义时间量。

$$\text{时间量} = 2 \times \text{BRP} / f_{\text{clk}}$$

BRP（波特率预分频器）为 BCR0 保存值 +1 的值。 f_{clk} 为外围总线时钟频率。

(1) BCR1

TSEG1 和 TSEG2 的设置参照表 19.5。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG1[3:0]				—	TSG2[2:0]			—	—	SJW[1:0]		—	—	—	BSP
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R	R/W

位	位名称	初始值	R/W	说 明
15 ~ 12	TSG1[3:0]	0000	R/W	时间段 1 这些位用于设定段 TSEG1 (= PRSEG + PHSEG1)，以通过正相位错误补偿 CAN 总线上的边沿。可设定 4 ~ 16 个时间量。 0000: 禁止设定 0001: 禁止设定 0010: 禁止设定 0011: PRSEG + PHSEG1=4 个时间量 0100: PRSEG + PHSEG1=5 个时间量 : : 1111: PRSEG + PHSEG1=16 个时间量
11	—	0	R	保留位 读取值、写入值总是为 0。
10 ~ 8	TSG2[2:0]	000	R/W	时间段 2 这些位用于设定段 TSEG2 (=PRSEG2)，以通过反相位错误补偿 CAN 总线上的边沿。可设定 2 ~ 8 时间量。 000: 禁止设定 001: PHSEG2=2 个时间量 (根据条件禁止设定。参照表 19.5) 010: PHSEG2=3 个时间量 011: PHSEG2=4 个时间量 100: PHSEG2=5 个时间量 101: PHSEG2=6 个时间量 110: PHSEG2=7 个时间量 111: PHSEG2=8 个时间量
7、6	—	均为 0	R	保留位 读取值、写入值总是为 0。
5、4	SJW[1:0]	00	R/W	再次同步跳转宽度 设定同步跳转宽度。 00: 同步跳转宽度 =1 个时间量 01: 同步跳转宽度 =2 个时间量 10: 同步跳转宽度 =3 个时间量 11: 同步跳转宽度 =4 个时间量
3 ~ 1	—	均为 0	R	保留位 读取值、写入值总是为 0。
0	BSP	0	R/W	位采样点 本位设定采样数据点。 0: 在 1 处执行位采样 (时间段 1 的最后) 1: 在 3 处执行位采样 (PHSEG1 的最后 3 个时钟周期的上升沿)

(2) BCR0

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	BRP[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15 ~ 8	—	均为 0	R	保留位 读取值、写入值总是为 0。
7 ~ 0	BRP[7:0]	0000000	R/W	波特率预分频器 这些位设定 1 个时间量对应的外围总线时钟数。 0000000: 2× 外围总线时钟 0000001: 4× 外围总线时钟 0000010: 6× 外围总线时钟 : 2× (寄存器值 +1) × 外围总线时钟 1111111: 512 × 外围总线时钟

• 关于位结构寄存器

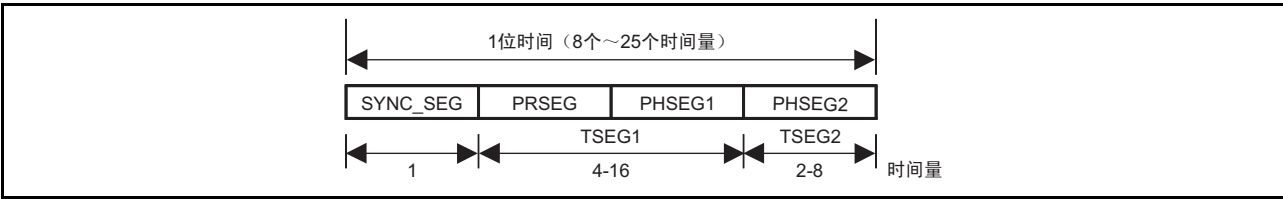


图 19.12 1 位时间的结构

- SYNC_SEG: 同步 CAN 总线节点的段
(在该段产生通常的位边沿转移)
- PRSEG : 调整网络间物理性延迟的段
- PHSEG1 : 相位漂移 (正向) 的缓冲段 (同步或再次同步时扩展)
- PHSEG2 : 相位漂移 (反向) 的缓冲段 (同步或再次同步时缩小)
- TSEG1 : TSG1+1
- TSEG2 : TSG2+1

RCAN-TL1 位速率的计算公式如下:

$$\text{位速率} = \text{fclk} / (2 \times (\text{BRP} + 1) \times (\text{TSEG1} + \text{TSEG2} + 1))$$

以寄存器值赋值 BRP，TSEG1 及 TSEG2 为根据上表计算出的值，并非实际设定值。上述计算公式中时间段的“+1”取决于 SYNC_SEG 的 1 个时间量。

f_{clk} = 外围时钟

BCR 设定时的限制事项

$TSEG1 \text{ (Min.)} > TSEG2 \geq SJW \text{ (Max.)}$ (SJW=1 ~ 4)

$8 \leq TSEG1 + TSEG2 + 1 \leq 25$ 个时间量 (TSEG1 + TSEG2 + 1=7 不可)

$TSEG2 \geq 2$

如果位结构寄存器 TSEG1 及 TSEG2 值在表 19.6 所示的设定值范围内，则满足上述限制事项。表 19.6 中的值为 SJW 的可设定值。“不可”表示 TSEG1 与 TSEG2 不可组合。

表 19.6 TSG 与 TSEG 的设定

		001	010	011	100	101	110	111	TSG2
		2	3	4	5	6	7	8	TSEG2
TSG1	TSEG1								
0011	4	不可	1-3	不可	不可	不可	不可	不可	
0100	5	1-2	1-3	1-4	不可	不可	不可	不可	
0101	6	1-2	1-3	1-4	1-4	不可	不可	不可	
0110	7	1-2	1-3	1-4	1-4	1-4	不可	不可	
0111	8	1-2	1-3	1-4	1-4	1-4	1-4	不可	
1000	9	1-2	1-3	1-4	1-4	1-4	1-4	1-4	
1001	10	1-2	1-3	1-4	1-4	1-4	1-4	1-4	
1010	11	1-2	1-3	1-4	1-4	1-4	1-4	1-4	
1011	12	1-2	1-3	1-4	1-4	1-4	1-4	1-4	
1100	13	1-2	1-3	1-4	1-4	1-4	1-4	1-4	
1101	14	1-2	1-3	1-4	1-4	1-4	1-4	1-4	
1110	15	1-2	1-3	1-4	1-4	1-4	1-4	1-4	
1111	16	1-2	1-3	1-4	1-4	1-4	1-4	1-4	

例 1: 如果 f_{clk} 为 32MHz，将位速率设定为 500kbps 时，需满足 BRP=1、TSEG1=11、TSEG2=4 的条件。此时，对 BCR1 写入 H'A300、对 BCR0 写入 H'0001。

例 2: 如果 f_{clk} 为 20MHz，将位速率设定为 500kbps 时，需满足 BRP=1、TSEG1=6、TSEG2=3 的条件。此时，对 BCR1 写入 H'5200、对 BCR0 写入 H'0001。

19.4.4 中断请求寄存器（IRR）

IRR 为 16 位可读取 / 写入的寄存器，由各中断源的状态标志构成。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IRR15	IRR14	IRR13	IRR12	IRR11	IRR10	IRR9	IRR8	IRR7	IRR6	IRR5	IRR4	IRR3	IRR2	IRR1	IRR0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R/W

位	位名称	初始值	R/W	说 明
15	IRR15	0	R/W	定时器比较匹配中断 1 表示已产生定时器比较匹配寄存器 1（TCMR1）的比较匹配。如果 TCMR1 的设定值与周期时间匹配（TCMR1=CYCTR），则本位置位。 0: 未产生 TCMR1 定时器的比较匹配 [清除条件] 写入 1 1: 产生 TCMR1 定时器的比较匹配 [置位条件] TCMR1 与周期时间匹配（TCMR1=CYCTR）
14	IRR14	0	R/W	定时器比较匹配中断 0 表示已产生定时器比较匹配寄存器 0（TCMR0）的比较匹配。如果 TCMR0 的设定值与局部时间匹配（TCMR0=TCNTR），则本位置位。 0: 未产生 TCMR0 定时器的比较匹配 [清除条件] 写入 1 1: 产生 TCMR0 定时器的比较匹配 [置位条件] TCMR0 与定时器值匹配（TCMR0=TCNTR）
13	IRR13	0	R/W	定时器溢出中断 /Next_is_Gap 接收中断 / 信息错误中断 本中断根据 RCAN-TL1 的模式而有不同的含义，表示以下状态： - 以事件触发模式运行时（包含测试模式），产生定时器 (TCNTR) 溢出。 - 以时间触发模式运行时（包含测试模式），接收已设置 Next_is_Gap 的时间基准信息。 - 在测试模式产生信息错误。但在测试模式即使产生信息超载条件，也不可置位本位。 0: 在事件触发模式（包含测试模式）未产生定时器（TCNTR）溢出 在时间触发模式（包含测试模式）未接收带 Next_is_Gap 的时间基准信息。 在测试模式未产生信息错误。 [清除条件] 写入 1 1: [置位条件] 在事件触发模式（包含测试模式），产生定时器（TCNTR）溢出，并从 H'FFFF 变为 H'0000 在时间触发模式（包含测试模式）接收带 Next_is_Gap 的时间基准信息 在测试模式产生信息错误。

位	位名称	初始值	R/W	说 明
12	IRR12	0	R/W	CAN 睡眠模式时总线激活 本位表示存在 CAN 总线激活。RCAN-TL1 为 CAN 睡眠模式时，如果在 CAN 总线检测出显性位，则本位置位。如果向本位写入 1 则清除中断，写入 0 则忽略。未使用自动唤醒功能、且未接收本中断时，则必须通过对应的屏蔽位使本中断无效。如果未使用自动唤醒功能时请求本中断，则必须从 CAN 睡眠模式返回后清除本位。以上可防止因接收行的新下降沿引起的再次设置中断。 必须注意本中断的设置时间与 GSR4 的清除时间不同。 0: 总线空闲状态 [清除条件] 写入 1 1: 在 CAN 睡眠模式检测出 CAN 总线激活 [置位条件] 在 CAN 睡眠模式检测出 CRxn 向显性的位状态变化
11	IRR11	0	R/W	定时器比较匹配中断 2 表示已产生定时器比较匹配寄存器 2(TCMR2) 的比较匹配。如果 TCMR2 的设定值与局部时间匹配 (TCMR2=CYCTR)，则本位置位。 0: 不产生 TCMR2 的定时器比较匹配 [清除条件] 写入 1 1: 产生 TCMR2 的定时器比较匹配 [置位条件] TCMR2 与周期时间匹配 (TCMR2=CYCTR)
10	IRR10	0	R/W	开始系统矩阵中断 表示已开始下一个系统矩阵。CCR=0 时，若完成发送 / 接收时间基准信息，则本位置位。CMAX=0 时，本中断位在每个周期计数 (Basic Cycle) 时置位。 0: 不在新系统矩阵的起始 [清除条件] 写入 1 1: 周期计数器为 0 [置位条件] CMAX!=B'111 且 CCR=0 时，完成发送 / 接收时间基准信息
9	IRR9	0	R	信息溢出 / 重写中断标志 本位表示虽然已接收信息，但是因对应的 RXPR 或 RFPR 已置 1 而不可读取邮箱内的信息，也未被 CPU 清除，所以新接收的信息根据 NMC 位的设定，被丢弃 (上溢) 或重写 (overwrite)。对 UMSR 寄存器的对应位写入 1，可清除本位。另外，对 MBIMR 对应的所有位写入 1，也可清除本位。对本位写入无效。 0: 无信息溢出 / 重写通知 [清除条件] UMSR 的所有位清除或已置位的所有 UMSR 对应的 MBIMR 置位 1: 因产生溢出而丢弃或重写接收信息 [置位条件] 对应的 PXPR 或 RFPR=1 且 MBIMR=0 时接收信息

位	位名称	初始值	R/W	说 明
8	IRR8	0	R	邮箱空中断标志 1 条设定为发送的信息正常发送（对应的 TXACK 标志置位）或发送中止（已取消发送的信息对应的 ABACK 标志置位）时本位置位。此时在事件触发模式清除对应的 TXPR，并且邮箱可接收下一个需发送的信息数据。在时间触发模式，为了持续发送每个已编程的周期计数（Basic Cycle），发送完成后不清除邮箱 24 ~ 30 的 TXPR。 实际上本位因未被 MBIMR 标志屏蔽的 TXACK 及获取 ABACK 位的 OR 的信号而置位。因此，所有的 TXACK 和 ABACK 位被清除时，本位自动清除。另外，对 MBIMR 对应的所有位写入 1，也可清除。对本位写入无效。 0：未执行发送或取消发送的信息 [清除条件] 所有的 TXACK 及 ABACK 位清除或置位后对应的 MBIMR 置位 1：发送信息或中止发送（取消发送），即可保存下一个信息（在时间触发模式，邮箱 24 ~ 30 仅在中止时可保存下一个信息） [置位条件] TXACK 或 ABACK 位置位时（对应的 MBIMR=0 时）
7	IRR7	0	R/W	超载帧 本标志表示 RCAN-TL1 检测出发送超载帧。对 IRR7 写入 1 时清除，写入 0 时忽略。 0：[清除条件] 写入 1 1：[置位条件] 检测出超载条件
6	IRR6	0	R/W	总线断开中断标志 RCAN-TL1 为总线断开状态或从该状态返回错误主动状态时，本位置位。因此节点 $TEC \geq 256$、总线断开返回顺序结束（连续接收 128 次 11 个隐性位）或从总线断开向停止状态转移（自动或手动）均为中断源。 RCAN-TL1 在总线断开解除后，本位仍保持置位状态，因此需由软件清除。必须由软件读取 GSR0，判断 RCAN-TL1 是否处于总线断开或错误主动状态。即使节点仍处于总线断开状态，写入 1 后也可清除本位，写入 0 则忽略。 0：[清除条件] 写入 1 1：因发送错误而转移至总线断开状态或从总线断开状态返回错误主动状态 [置位条件] $TEC \geq 256$ 或连续接收 128 次 11 个隐性位后结束总线断开或从总线断开向停止状态转移
5	IRR5	0	R/W	错误被动中断标志 表示因接收或发送错误计数器导致的错误被动状态，或因测试模式而强制成为错误被动状态。写入 1 时清除本位，写入 0 时忽略。即使清除本位，节点可能仍保持错误被动状态。需由软件检查 GSR0 及 GSR5，判断 RCAN-TL1 是否为错误被动或总线断开状态。 0：[清除条件] 写入 1 1：因发送 / 接收错误导致错误被动状态 [置位条件] 使用 $TEC \geq 128$ 或 $REC \geq 128$ 或错误被动测试模式

位	位名称	初始值	R/W	说 明
4	IRR4	0	R/W	接收错误计数器警告中断标志 RCAN-TL1 不为总线断开状态时, 如果接收错误计数器 (REC) 的值大于 95, 则本位置位。对本位写入 1 时清除中断, 写入 0 时忽略。 0: [清除条件] 写入 1 1: 因接收错误导致错误警告状态 [置位条件] RCAN-TL1 不为总线断开状态且 $REC \geq 96$
3	IRR3	0	R/W	发送错误计数器警告中断标志 RCAN-TL1 不为总线断开状态时, 如果发送错误计数器 (TEC) 的值大于 95, 则本位置位。对本位写入 1 时清除中断, 写入 0 时忽略。 0: [清除条件] 写入 1 1: 因发送错误导致错误警告状态 [置位条件] $TEC \geq 96$
2	IRR2	0	R	远程帧接收中断标志 表示邮箱已接收到远程帧。如果对应的 MBIMR 未置位的至少 1 个邮箱保存了远程帧发送请求时, 本位置位。清除远程帧接收完成寄存器 (RFPR) 的所有位时, 可清除本位。另外, 向 MBIMR 的所有对应位写入 1, 也可清除。对本位写入无效。 0: [清除条件] 清除 RFPR 的所有位 1: 至少 1 个远程帧请求为等待处理状态 [置位条件] 对应的 MBIMR=0 时接收远程帧
1	IRR1	0	R	数据帧接收中断标志 表示存在等待处理的接收数据帧。本位为 1 时, 至少有 1 个邮箱保存了未处理的信息。清除数据帧接收完成寄存器 (RXPR) 的所有位时, 清除本位 (任何接收邮箱均无未处理的信息)。设定 MBIMR 不为 0 的各接收邮箱 RXPR 标志的逻辑“或”。 另外, 对 MBIMR 的所有对应位写入 1, 可清除本位。对本位写入无效。 0: [清除条件] 清除 RXPR 的所有位 1: 接收数据帧并保存于邮箱 [置位条件] 对应的 MBIMR=0 时接收数据
0	IRR0	1	R/W	复位 / 停机 / 睡眠中断标志 本标志因以下 3 个中断源而置位: • 软件复位 (MCR0) 或硬件复位后, 转移至复位模式 • 停机请求 (MCR1) 后, 转移至停机模式 • 在停机模式发行睡眠请求 (MCR5) 后, 转移至 CAN 睡眠模式。 本位置位后读取 GSR, 可得知 RCAN-TL1 的状态。 【注】 需发行睡眠模式请求时, 必须预先使用停机模式。详情参阅 MCR5 的说明及图 19.8。GSR3 或 GSR4 从 0 变为 1 时, 或从停机模式转移至 CAN 睡眠模式时, IRR0 置位。因此, 如果 RCAN-TL1 解除停机模式后再次转移至该模式, 则因未清除 GSR4, 而使 IRR0 不可置位。同样, 直接从 CAN 睡眠模式转移至停机模式时 IRR0 也不可置位。从停机 / CAN 睡眠模式转移至发送 / 接收运行时, 在清除 GSR4 前需 [1 位时间 - TSEG2] ~ [1 位时间 $\times 2$ - TSEG2] 的时间。 复位模式时 IRR0 可置位, 但是通过初始化 IMR0 自动置位, 因此对 CPU 的中断无效。 0: [清除条件] 写入 1 1: 转移至软件复位模式或停机模式或 CAN 睡眠模式 [置位条件] 请求复位 (MCR0 或硬件)、停机模式 (MCR1) 或 CAN 睡眠模式 (MCR5) 后, 完成向复位 / 停机 / CAN 睡眠模式的转移

19.4.5 中断屏蔽寄存器 (IMR)

IMR 为 16 位可读取 / 写入的寄存器，屏蔽中断请求寄存器 (IRR) 各位对应的中断 IRQ 输出信号的生成。如果位设定为 1，则屏蔽对应的中断请求。IMR 直接控制 IRQ 的生成，但不禁止 IRR 的对应位置位。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IMR15	IMR14	IMR13	IMR12	IMR11	IMR10	IMR9	IMR8	IMR7	IMR6	IMR5	IMR4	IMR3	IMR2	IMR1	IMR0
初始值:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15 ~ 0	IMR[15:0]	均为 1	R/W	IRR[15:0] 的各位对应的中断源屏蔽 本位置位时，即使 IRR 的对应位置位，也不能生成中断信号。 0: 不屏蔽对应的 IRR (产生中断源则生成 IRQ) 1: 屏蔽 IRR 的对应中断

19.4.6 发送错误计数器 (TEC) / 接收错误计数器 (REC)

TEC 及 REC 为 16 位可读取 / 带条件写入寄存器，是表示 CAN 接口中的发送 / 接收信息错误数的计数器。在前述参考文献的 [1]、[2]、[3]、[4] 中规定了计数值。除写入错误计数器测试模式之外，本寄存器为只读，仅可通过 CAN 接口改写。通过复位请求 (MCR0) 或转移至总线断开状态，清除本寄存器。

在写入错误计数器测试模式 (TST[2:0]=B'100) 可写入本寄存器。对 TEC 及 REC 仅可写入相同的值。写入 TEC 的值可设定给 TEC 及 REC。写入本寄存器时，RCAN-TL1 需设定为停机模式。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TEC7	TEC6	TEC5	TEC4	TEC3	TEC2	TEC1	TEC0	REC7	REC6	REC5	REC4	REC3	REC2	REC1	REC0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*

【注】 * 由 MCR 的 TST[2:0]=B'100 仅可在测试模式时写入。在总线断开状态，REC 计数总线断开返回顺序所需的 11 个隐性位的接收次数。

位	位名称	初始值	R/W	说 明
15 ~ 8	TEC[7:0]	均为 0	R/W*	发送错误计数器 发送过程中根据 CAN 规格检测出错误时，本寄存器的值递增。
7 ~ 0	REC[7:0]	均为 0	R/W*	接收错误计数器 接收过程中根据 CAN 规格检测出错误时，本寄存器的值递增。

19.5 RCAN-TL1 邮箱寄存器

说明 RCAN-TL1 邮箱寄存器。该寄存器可控制各邮箱并表示邮箱的状态。

RCAN-TL1 邮箱寄存器如表 19.7 所示。

【重点】长字存取为 2 个连续字的存取。

表 19.7 RCAN-TL1 邮箱寄存器

寄存器名称	简称	地址	存取长度 (位)
发送等待寄存器 1	TXPR1	H'020	32
发送等待寄存器 0	TXPR0	H'022	—
		H'024	
		H'026	
发送取消寄存器 1	TXCR1	H'028	16/32
发送取消寄存器 0	TXCR0	H'02A	16
		H'02C	
		H'02E	
发送应答寄存器 1	TXACK1	H'030	16/32
发送应答寄存器 0	TXACK0	H'032	16
		H'034	
		H'036	
中止应答寄存器 1	ABACK1	H'038	16/32
中止应答寄存器 0	ABACK0	H'03A	16
		H'03C	
		H'03E	
数据帧接收完成寄存器 1	RXPR1	H'040	16/32
数据帧接收完成寄存器 0	RXPR0	H'042	16
		H'044	
		H'046	
远程帧接收完成寄存器 1	RFPR1	H'048	16/32
远程帧接收完成寄存器 0	RFPR0	H'04A	16
		H'04C	
		H'04E	
邮箱中断屏蔽寄存器 1	MBIMR1	H'050	16/32
邮箱中断屏蔽寄存器 0	MBIMR0	H'052	16
		H'054	
		H'056	
未读信息状态寄存器 1	UMSR1	H'058	16/32
未读信息状态寄存器 0	UMSR0	H'05A	16
		H'05C	
		H'05E	

19.5.1 发送等待寄存器 1、0 (TXPR1、TXPR0)

TXPR1 与 TXPR0 连接构成了保存 CAN 模块发送等待标志的 32 位寄存器。16 位总线接口时，长字存取为 2 个连续的字存取。

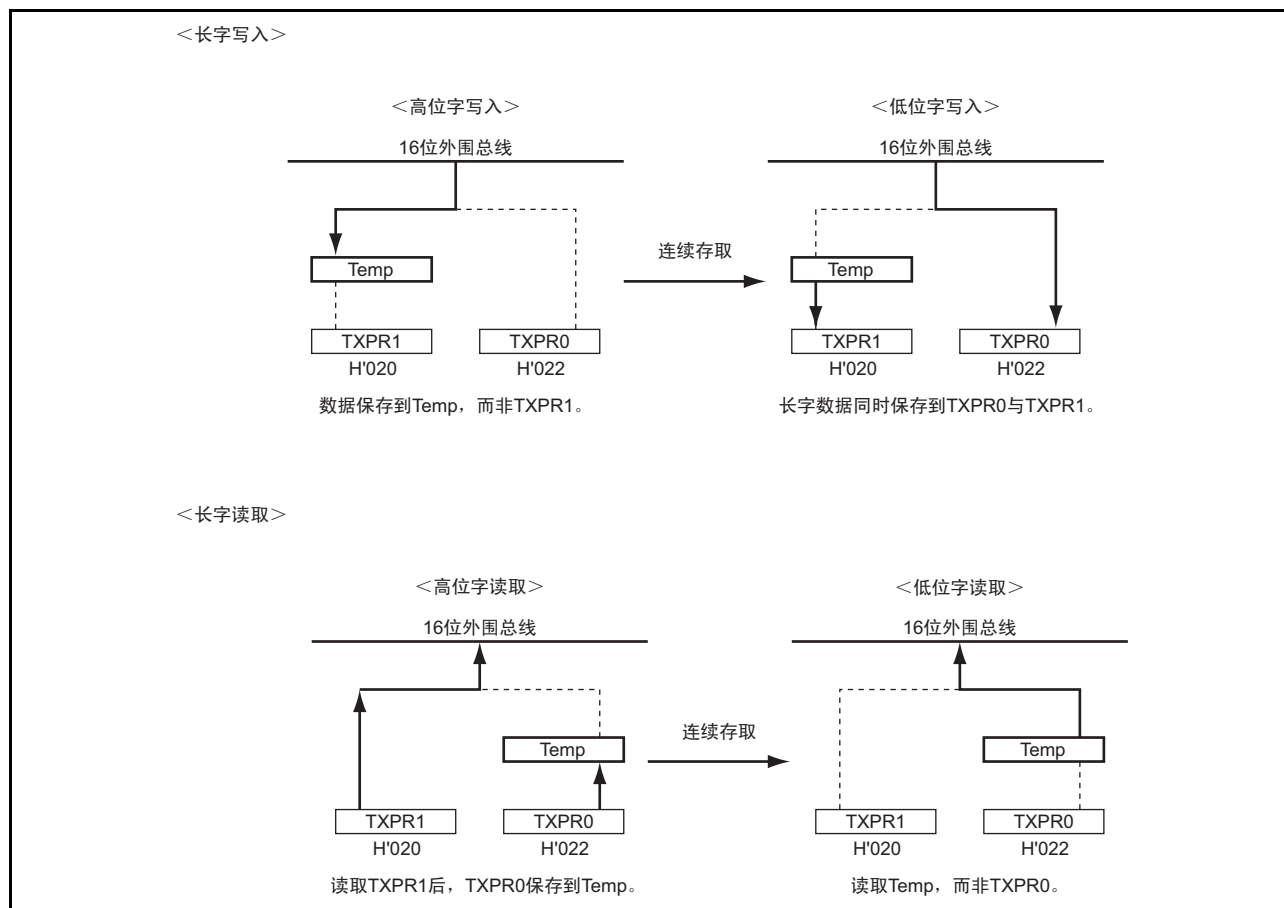


图 19.13 16 位总线接口时的长字存取

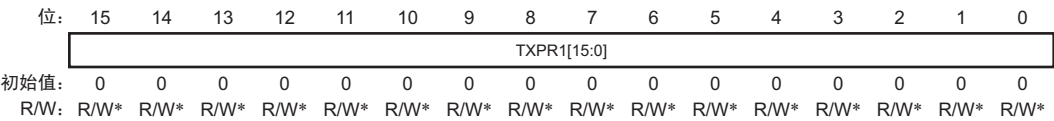
TXPR1 寄存器控制邮箱 31 ~ 16，TXPR0 寄存器控制邮箱 15 ~ 1。CPU 通过对位写入 1，即可对相应的发送信息进行操作。写入 0 无效，为了清除 TXPR，必须将 TXCR 的对应位置位。CPU 读取 TXPR 后，即可得知发送等待的邮箱及正在发送的邮箱。实际上除邮箱 0 之外，所有的邮箱均存在发送等待位。另外，禁止对未设定为发送的邮箱位写入 1。

在事件触发模式，正常发送信息后或执行 TXCR 的发送中止后，RCAN-TL1 清除对应的发送等待标志。在时间触发模式，为了持续发送已编程的周期计数（Basic Cycle），发送完成后不清除邮箱 30 ~ 24 的 TXPR。由于 CAN 节点仲裁失败或 CAN 总线产生错误，未发送信息时，不清除 TXPR 标志。此时，如果对应邮箱的信息控制字段的 DART（自动再发送无效）位未置位，则 RCAN-TL1 自动尝试重新发送。如果 DART 置位，则清除发送，通过邮箱空中断标志（IRR8）与中止应答寄存器（ABACK）的对应位通知 CPU。

TXPR 的状态变化时，即使总线仲裁失败或在 CAN 总线产生错误，也总是发送显示 ID 优先顺序（MCR2=0）最高的信息。详情参阅“19.7 运行说明”。

RCAN-TL1 在 TXPR 位的状态变为 0 时，可能生成邮箱空中断标志（IRR8）。这表示邮箱发送正常结束或中止。信息发送正常结束时，反映于 TXACK；中止时，反映于 ABACK。通过检查这些寄存器，可将对应邮箱的信息数据字段内容改写为下次发送内容。

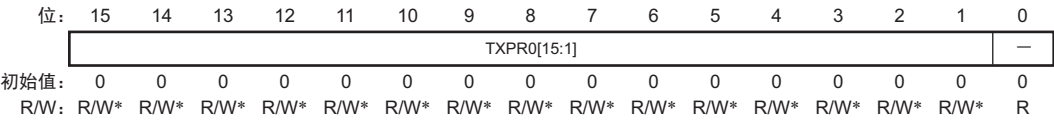
(1) TXPR1



【注】 * 对设定为发送的邮箱的对应位仅可写入 1。

位	位名称	初始值	R/W	说 明
15 ~ 0	TXPR1[15:0]	H'0000	R/W	请求向对应邮箱发送 CAN 帧。bit15 ~ 0 对应邮箱 31 ~ 16。多个位置位时的发送，可根据 MCR2 位设定的信息 ID 优先顺序或邮箱编号顺序发送。 0: 对应的邮箱为发送信息空闲状态 [清除条件] 信息发送结束（事件触发信息时）或信息发送中止（自动清除） 1: 在对应的邮箱产生发送请求

(2) TXPR0



【注】 * 对设定为发送的邮箱的对应位仅可写入 1。
必须以长字存取进行 TXPR1/TXPR0 的读取 / 写入。

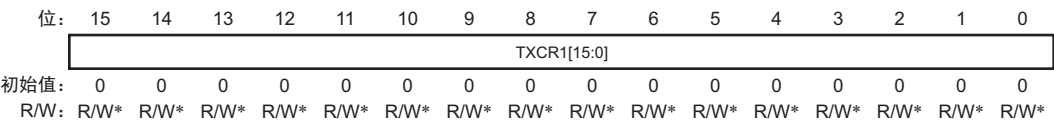
位	位名称	初始值	R/W	说 明
15 ~ 1	TXPR0[15:1]	H'0000	R/W	表示在对应邮箱产生 CAN 帧发送请求。bit15 ~ 1 对应邮箱 15 ~ 1。多个位置位时的发送，可根据 MCR2 位设定的信息 ID 优先顺序或邮箱编号顺序发送。 0: 对应的邮箱为发送信息空闲状态 [清除条件] 信息发送结束（事件触发信息时）或信息发送中止（自动清除） 1: 在对应的邮箱产生发送请求
0	—	0	R	保留位 为接收专用邮箱，因此本位总是为 0，写入 1 时忽略，读取值总是为 0。

19.5.2 发送取消寄存器 1、0（TXCR1、TXCR0）

TXCR1 与 TXCR0 为 16 位可读取 / 带条件写入的寄存器。TXCR1 控制邮箱 31 ~ 16；TXCR0 控制邮箱 15 ~ 1。CPU 使用 TXCR 取消 TXPR 的发送请求。清除 TXPR 位则必须由 CPU 向对应的 TXCR 位写入 1，写入 0 则忽略。

正常执行中止时，CAN 控制器清除对应的 TXPR 和 TXCR 位，并置位 ABACK 位。但邮箱开始发送后，通过本寄存器的位不可取消发送。这种情况，发送正常结束后，CAN 控制器清除 TXPR 和 TXCR 位，并置位 TXACK 位。但因仲裁失败或总线产生错误而导致不能发送时，CAN 控制器清除对应的 TXPR 和 TXCR 位，并置位对应的 ABACK 位。即使 CPU 试图清除非发送等待状态的邮箱发送，仍忽略。此时，CPU 不可置位 TXCR 标志。

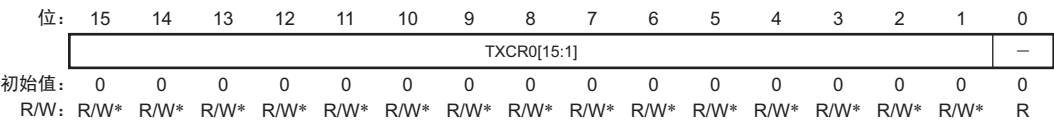
(1) TXCR1



【注】 * 接受发送请求的发送邮箱的对应位仅可写入 1。

位	位名称	初始值	R/W	说 明
15 ~ 0	TXCR1[15:0]	H'0000	R/W	对列入发送队列的邮箱请求取消发送。bit15 ~ 0 对应邮箱 31 ~ 16（及 TXPR1[15:0]）。 0: 对应的邮箱为取消发送信息空闲状态 [清除条件] 取消发送信息结束（自动清除） 1: 向对应的邮箱请求取消发送

(2) TXCR0



【注】 * 接受发送请求的发送邮箱的对应位仅可写入 1。

位	位名称	初始值	R/W	说 明
15 ~ 1	TXCR0[15:1]	H'0000	R/W	对列入发送队列的邮箱请求取消发送。bit15 ~ 1 对应邮箱 15 ~ 1（及 TXPR0[15:1]）。 0: 对应的邮箱为取消发送信息空闲状态 [清除条件] 取消发送信息结束（自动清除） 1: 向对应的邮箱请求取消发送
0	—	0	R	保留位 为接收专用邮箱，因此本位总是为 0，写入 1 则忽略，读取值总是为 0。

19.5.3 发送应答寄存器 1、0（TXACK1、TXACK0）

TXACK1 与 TXACK0 为 16 位可读取 / 带条件写入寄存器，向 CPU 通知邮箱已正常发送。正常发送时，RCAN-TL1 置位 TXACK 寄存器的对应位。写入 1 后，CPU 可清除 TXACK 位，写入 0 则忽略。

(1) TXACK1

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TXACK1[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*

【注】 * 为了清除位，仅可写入 1。

位	位名称	初始值	R/W	说 明
15 ~ 0	TXACK1[15:0]	H'0000	R/W	通知对应邮箱，请求的发送已正常执行。bit15 ~ 0 对应邮箱 31 ~ 16。 0: [清除条件] 写入 1 1: 已正常发送对应邮箱的信息（数据帧或远程帧） [置位条件] 对应邮箱的信息发送结束

(2) TXACK0

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TXACK0[15:1]															—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	—

【注】 * 为了清除位，仅可写入 1。

位	位名称	初始值	R/W	说 明
15 ~ 1	TXACK0[15:1]	H'0000	R/W	通知对应邮箱，请求的发送已正常执行。bit15 ~ 1 对应邮箱 15 ~ 1。 0: [清除条件] 写入 1 1: 已正常发送对应邮箱的信息（数据帧或远程帧） [置位条件] 对应邮箱的信息发送结束
0	—	0	R	保留位 为接收专用邮箱，因此本位总是为 0，写入 1 则忽略，读取值总是为 0。

19.5.4 中止应答寄存器 1、0（ABACK1、ABACK0）

ABACK1 与 ABACK0 为 16 位可读取 / 带条件写入寄存器，根据请求向 CPU 通知邮箱发送已中止。中止时，RCAN-TL1 置位 ABACK 寄存器的对应位。写入 1 后，CPU 可清除 ABACK 位，写入 0 则忽略。RCAN-TL1 通过置位 ABACK 位，由对应的 TXCR 位识别已清除 TXPR 位。

(1) ABACK1

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ABACK1[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*

【注】* 为了清除位，仅可写入 1。

位	位名称	初始值	R/W	说 明
15 ~ 0	ABACK1[15:0]	H'0000	R/W	通知对应邮箱，请求的取消发送已正常执行。bit15 ~ 0 对应邮箱 31 ~ 16。 0: [清除条件] 写入 1 1: 已取消发送对应邮箱的信息（数据帧或远程帧） [置位条件] 取消对应邮箱的信息发送

(2) ABACK0

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ABACK0[15:1]															—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R

【注】* 为了清除位，仅可写入 1。

位	位名称	初始值	R/W	说 明
15 ~ 1	ABACK0[15:1]	H'0000	R/W	通知对应邮箱，请求的取消发送已正常执行。bit15 ~ 1 对应邮箱 15 ~ 1。 0: [清除条件] 写入 1 1: 已取消发送对应邮箱的信息（数据帧或远程帧） [置位条件] 取消对应邮箱的信息发送
0	—	0	R	保留位 为接收专用邮箱，因此本位总是为 0，写入 1 则忽略，读取值总是为 0。

19.5.5 数据帧接收完成寄存器 1、0（RXPR1、RXPR0）

RXPR1 与 RXPR0 为 16 位可读取 / 带条件写入寄存器，由表示设定为接收的邮箱已接收数据帧的标志构成。CAN 数据帧正常保存至接收邮箱时，置位 RXPR 的对应位。写入 1 时清除对应位，写入 0 则无效。但仅在由 MBC（邮箱结构）设定为邮箱接收数据帧时置位。置位 RXPR 位时，如果未置位对应的 MBIMR，IRR1（数据帧接收中断标志）也置位，如果 IMR1 未置位，则生成中断信号。本寄存器的位仅因接收数据帧而设置，接收远程帧时不置位。

(1) RXPR1

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RXPR1[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*

【注】 * 为了清除位，仅可写入 1。

位	位名称	初始值	R/W	说 明
15 ~ 0	RXPR1[15:0]	H'0000	R/W	bit15 ~ 0 对应设定为接收的邮箱 31 ~ 16。 0: [清除条件] 写入 1 1: 对应邮箱已接收 CAN 数据帧 [置位条件] 对应邮箱的数据帧接收结束

(2) RXPR0

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RXPR0[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*

【注】 * 为了清除位，仅可写入 1。

位	位名称	初始值	R/W	说 明
15 ~ 0	RXPR0[15:0]	H'0000	R/W	bit15 ~ 0 对应设定为接收的邮箱 15 ~ 0。 0: [清除条件] 写入 1 1: 对应邮箱已接收 CAN 数据帧 [置位条件] 对应邮箱的数据帧接收结束

19.5.6 远程帧接收完成寄存器 1、0（RFPR1、RFPR0）

RFPR1 与 RFPR0 为 16 位可读/带条件写入寄存器，由表示设定为接收的邮箱已接收远程帧的标志构成。CAN 远程帧正常保存至接收邮箱时，置位 RFPR 的对应位。写入 1 时清除对应位，写入 0 则无效。所有的邮箱均存在位，但仅在由 MBC（邮箱结构）设定为邮箱接收远程帧时置位。置位 RFPR 位时，如果未置位对应的 MBIMR，则 IRR2（远程帧接收中断标志）也置位，如果 IMR2 未置位，则生成中断信号。本寄存器的位仅因接收远程帧而置位，接收数据帧时不置位。

(1) RFPR1

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFPR1[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*

【注】 * 为了清除位，仅可写入 1。

位	位名称	初始值	R/W	说 明
15 ~ 0	RFPR1[15:0]	H'0000	R/W	邮箱 31 ~ 16 的远程请求接收标志。 0: [清除条件] 写入 1 1: 对应邮箱已接收 CAN 远程帧 [置位条件] 对应邮箱的远程帧接收结束

(2) RFPR0

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFPR0[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*

【注】 * 为了清除位，仅可写入 1。

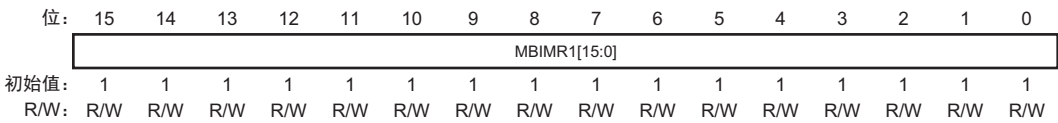
位	位名称	初始值	R/W	说 明
15 ~ 0	RFPR0[15:0]	H'0000	R/W	邮箱 15 ~ 0 的远程请求接收标志。 0: [清除条件] 写入 1 1: 对应邮箱已接收 CAN 远程帧 [置位条件] 对应邮箱的远程帧接收结束

19.5.7 邮箱中断屏蔽寄存器 1、0（MBIMR1、MBIMR0）

MBIMR1 与 MBIMR0 为 16 位可读取 / 写入的寄存器。MBIMR 为屏蔽与邮箱运行相关的 IRR（IRR1：数据帧接收中断、IRR2：远程帧接收中断、IRR8：邮箱空中断、IRR9：信息溢出 / 重写中断）的寄存器。邮箱设定为接收时，根据接收中断标志（IRR1、IRR2、IRR9）屏蔽中断。不影响 RXPR、RFPR、UMSR 位的置位。邮箱设定为发送时，由发送或发送中止（IRR8）屏蔽中断或邮箱空中断。不影响因发送产生的 TXPR / TXCR 位的清除和 TXACK 位的置位，及因发送中止产生的 TXPR / TXCR 位的清除和 ABACK 位的置位。

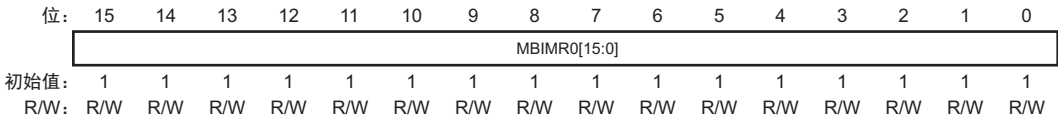
向屏蔽邮箱对应的位写入 1 可设定屏蔽。复位时屏蔽所有的邮箱中断。

(1) MBIMR1



位	位名称	初始值	R/W	说 明
15 ~ 0	MBIMR1[15:0]	H'FFFF	R/W	允许 / 禁止邮箱 31 ~ 16 的中断请求。 0: 允许 IRR1、IRR2、IRR8、IRR9 的中断请求 1: 禁止 IRR1、IRR2、IRR8、IRR9 的中断请求

(2) MBIMR0



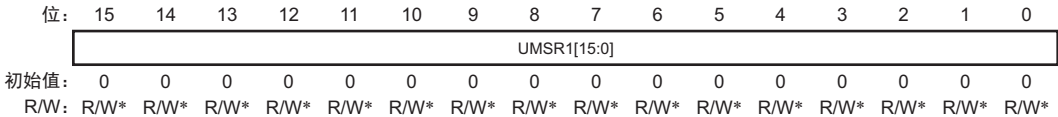
位	位名称	初始值	R/W	说 明
15 ~ 0	MBIMR0[15:0]	H'FFFF	R/W	允许 / 禁止邮箱 15 ~ 0 的中断请求。 0: 允许 IRR1、IRR2、IRR8、IRR9 的中断请求 1: 禁止 IRR1、IRR2、IRR8、IRR9 的中断请求

19.5.8 未读信息状态寄存器 1、0（UMSR1、UMSR0）

UMSR1 与 UMSR0 为 16 位可读取 / 带条件写入寄存器，接收新信息时，记录未由 CPU/DMAC 存取内容的接收邮箱。接收邮箱的新信息时，如果未由 CPU 清除 RXPR 或 RFPR 的对应位，则置位 UMSR 位。写入 1 时清除，写入 0 则无效。

邮箱设定为发送时，不可置位对应的 UMSR 位。

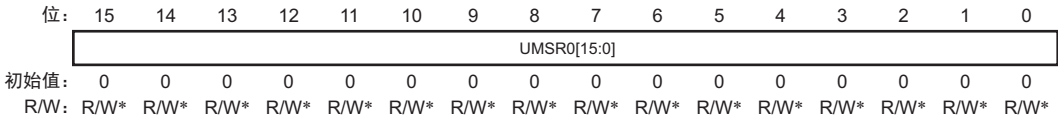
(1) UMSR1



【注】 * 为了清除位，仅可写入 1。

位	位名称	初始值	R/W	说 明
15 ~ 0	UMSR1[15:0]	H'0000	R/W	表示已重写邮箱 31 ~ 16 的未读接收信息或产生溢出。 0: [清除条件] 写入 1 1: 以新信息重写未读接收信息或产生溢出 [置位条件] 清除 RXPR 或 RFPR 前接收新信息

(2) UMSR0



【注】 * 为了清除位，仅可写入 1。

位	位名称	初始值	R/W	说 明
15 ~ 0	UMSR0[15:0]	H'0000	R/W	表示已重写邮箱 15 ~ 0 的未读接收信息或产生溢出。 0: [清除条件] 写入 1 1: 以新信息重写未读接收信息或产生溢出 [置位条件] 清除 RXPR 或 RFPR 前接收新信息

19.6 定时器寄存器

RCAN-TL1 的定时器为 16 位定时器，支持多种时钟源。通过预分频计数器可降低时钟的速度。另外，还备有 3 个比较匹配寄存器（TCMR2、TCMR1、TCMR0）。地址映射如下所示：

【重点】定时器寄存器均仅可字（16 位）存取。

表 19.8 RCAN-TL1 定时器寄存器的结构

寄存器名称	简称	地址	存取长度（位）
时间触发控制寄存器 0	TTCR0	H'080	16
最大周期 / Tx_Enable_Window 寄存器	CMAX_TEW	H'084	16
基准触发偏移寄存器	RFTROFF	H'086	16
定时器状态寄存器	TSR	H'088	16
周期计数器寄存器	CCR	H'08A	16
定时器计数器寄存器	TCNTR	H'08C	16
周期时间寄存器	CYCTR	H'090	16
基准标记寄存器	RFMK	H'094	16
定时器比较匹配寄存器 0	TCMR0	H'098	16
定时器比较匹配寄存器 1	TCMR1	H'09C	16
定时器比较匹配寄存器 2	TCMR2	H'0A0	16
发送触发时间选择寄存器	TTTSEL	H'0A4	16

19.6.1 时间触发控制寄存器 0（TTCR0）

TTCR0 为 16 位可读取 / 写入的寄存器。

控制定时器的运行。必须设定本寄存器、使定时器运行后再设定定期发送或监控寄存器。

位：	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TCR15	TCR14	TCR13	TCR12	TCR11	TCR10	—	—	—	TCR6	TPSC5	TPSC4	TPSC3	TPSC2	TPSC1	TPSC0
初始值：	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W：	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15	TCR15	0	R/W	定时器允许 设置本位时，TCNTR 运行，清除时，清除 TCNTR 与 CCR。 0：清除 TCNTR 与 CCR，停止运行 1：定时器运行
14	TCR14	0	R/W	时戳值 指定使用周期时间（CYCTR）或 CCR[5:0] + CYCTR[15:6] 作为邮箱 15 ~ 0 的发送及接收时戳。本功能便于监控时间触发模式的 Rx_Trigger。 本寄存器不影响邮箱 30 及 31 的时戳。 0：邮箱 15 ~ 0 时戳使用 CYCTR[15:0] 1：邮箱 15 ~ 0 时戳使用 CCR[5:0] + CYCTR[15:6]

位	位名称	初始值	R/W	说 明
13	TCR13	0	R/W	通过 TCMR2 取消 如果本位及 bit12 为已设置状态、RCAN-TL1 在停机模式之外产生比较匹配, 则该 TXPR 位对应的所有 TXCR 位自动置位, 这样可取消发送队列内的信息。 0: 禁止通过 TCMR2 的比较匹配取消发送 1: 允许通过 TCMR2 的比较匹配取消发送
12	TCR12	0	R/W	比较匹配允许 置位本位时, 通过 TCMR2 的比较匹配置位 IRR11。 0: 不通过 TCMR2 的比较匹配置位 IRR11 1: 通过 TCMR2 的比较匹配置位 IRR11
11	TCR11	0	R/W	比较匹配允许 置位本位时, 通过 TCMR1 的比较匹配置位 IRR15。 0: 不通过 TCMR1 的比较匹配置位 IRR15 1: 通过 TCMR1 的比较匹配置位 IRR15
10	TCR10	0	R/W	比较匹配允许 置位本位时, 通过 TCMR0 的比较匹配置位 IRR14。 0: 不通过 TCMR0 的比较匹配置位 IRR14 1: 通过 TCMR0 的比较匹配置位 IRR14
9 ~ 7	—	均为 0	R	保留位 读取值、写入值总是为 0。
6	TCR6	0	R/W	通过 TCMR0 控制定时器清除 TCNTR 与 TCMR0 匹配时, 指定是否将定时器清除为 H'0000。另外, 通过 IRR14, TCMR0 也可对 CPU 产生中断信号。 【注】 RCAN-TL1 在 TTCAN 模式运行时 (CMAX≠B'111), 必须设定本位为 0, 以避免清除本地时间。 0: 不通过 TCMR0 清除定时器 1: 通过 TCMR0 清除定时器
5 ~ 0	TPSC5 ~ TPSC0	均为 0	R/W	定时器预分频器 通过设定本位, 可分频使用定时器的源时钟 (4×[RCAN-TL1 系统时钟])。预分频器仅在事件触发模式有效。在 TTCAN 模式 (CMAX≠B'111) 自动选择 1 位时序 (=CAN 总线的 1 位长度) 作为 TCNTR 的源时钟。 源时钟周期与定时器周期之间的关系如下所示: 000000: 1X 源时钟 000001: 2X 源时钟 000010: 3X 源时钟 000011: 4X 源时钟 000100: 5X 源时钟 : 111111: 64X 源时钟

19.6.2 最大周期 /Tx_Enable_Window 寄存器 (CMAX_TEW)

CMAX_TEW 为 16 位可读取 / 写入的寄存器。

CMAX 指定时间触发发送的周期计数器 (CCR) 的最大值, 由此设定系统矩阵的周期数 (Basic Cycle)。周期计数器为最大值 (CCR=CMAX) 时, 1 个周期计数后周期计数器清 0, 产生 IRR10 中断。

TEW 指定 Tx_Enable_Window 宽度。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	CMAX[2:0]			—	—	—	—	TEW[3:0]			
初始值:	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15 ~ 11	—	均为 0	R	保留位 读取值、写入值总是为 0。
10 ~ 8	CMAX[2:0]	111	R/W	周期计数器最大值 表示 CCR 的最大值。 周期计数器最大值 = $2^{CMAX} - 1$ (CMAX = 寄存器值) 时间触发发送的矩阵周期内可计数的周期数为 (周期计数器最大值 + 1)。 设定 CMAX ≠ B'111 时, RCAN-TL1 为时间触发模式, 可使用时间触发功能。 设定 CMAX = B'111 时, RCAN-TL1 为事件触发模式。 000: 周期计数器最大值 = 0 001: 周期计数器最大值 = 1 010: 周期计数器最大值 = 3 011: 周期计数器最大值 = 7 100: 周期计数器最大值 = 15 101: 周期计数器最大值 = 31 110: 周期计数器最大值 = 63 111: RCAN-TL1 为事件触发模式 (清除 CCR) 【注】使用事件触发模式时, 必须设定为 CMAX = B'111。
7 ~ 4	—	均为 0	R	保留位 读取值、写入值总是为 0。
3 ~ 0	TEW[3:0]	0000	R/W	Tx_Enable_Window 设定 Tx_Enable_Window 宽度。 TEW = B'0000 时, 窗口宽度为 1 位时序。 可设定 B'0000 ~ B'1111 的所有值。 0000: Tx_Enable_Window 宽度 = 1 0001: Tx_Enable_Window 宽度 = 2 0010: Tx_Enable_Window 宽度 = 3 0011: Tx_Enable_Window 宽度 = 4 : 1111: Tx_Enable_Window 宽度 = 16 【注】RCAN-TL1 的 CAN 数据链路控制器虽然是从发送请求开始发送, 但是总是需要 CAN 总线 1 ~ 2 位时序。以上值并未此考虑该精确度。

19.6.3 基准触发偏移寄存器（RFTROFF）

RFTROFF 为 8 位可读取 / 写入的寄存器。

可对邮箱 30 的发送触发时间设定 -127 ~ +127 的偏移。邮箱 30 的 TTT 在附加 RFTROFF 寄存器值的符号扩展值后，与 CYCTR 比较。

但不可更改 TTT 值。此偏移值仅影响邮箱 30。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFTROFF[7:0]								—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

位	位名称	初始值	R/W	说 明
15 ~ 8	RFTROFF [7:0]	均为 0	R/W	表示基准触发偏移值。 00000000: ref_trigger_offset =+0 00000001: ref_trigger_offset =+1 00000010: ref_trigger_offset =+2 : 01111111: ref_trigger_offset =+127 : 11111111: ref_trigger_offset =-1 11111110: ref_trigger_offset =-2 : 10000001: ref_trigger_offset =-127
7 ~ 0	—	均为 0	R	保留位 读取值、写入值总是为 0。

19.6.4 定时器状态寄存器 (TSR)

TSR 为 16 位只读寄存器，通过本寄存器 CPU 可监控定时器比较匹配状态及定时器溢出状态。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	TSR4	TSR3	TSR2	TSR1	TSR0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名称	初始值	R/W	说 明
15 ~ 5	—	均为 0	R	保留位 读取值、写入值总是为 0。
4	TSR4	0	R*	开始系统矩阵 表示下一个系统矩阵已开始。 CCR=0 时，时间基准信息发送 / 接收完成后，本位置位。 0: 不为下一个系统矩阵的起始 [清除条件] 对 IRR10 (周期计数器上溢中断标志) 写入 1 1: 周期计数器为 0 [置位条件] 周期计数器的值从最大值变为 (CMAX) H'0 CMAX!=B'111 且 CCR=0 时，时间基准信息的发送 / 接收完成
3	TSR3	0	R*	定时器比较匹配标志 2 表示已产生定时器比较匹配寄存器 2 (TCMR2) 的比较匹配。 TCMR2 的设定值与周期时间寄存器匹配 (TCMR2=CYCTR) 时，如果 TTCR0 的 bit12 设定为 1，则本位置位。本位为只读，清除 IRR11 (定时器比较匹配中断 2 标志) 后本位被清除。 0: 不产生 TCMR2 定时器的比较匹配 [清除条件] 对 IRR11 (定时器比较匹配中断 2 标志) 写入 1 1: 产生 TCMR2 定时器的比较匹配 [置位条件] TTCR0 的 bit12 为 1 时，TCMR2 与周期时间匹配 (TCMR2=CYCTR)
2	TSR2	0	R*	定时器比较匹配标志 1 表示已产生定时器比较匹配寄存器 1 (TCMR1) 的比较匹配。 TCMR1 的设定值与周期时间寄存器匹配 (TCMR1=CYCTR) 时，如果 TTCR0 的 bit11 设定为 1，则本位置位。本位为只读，清除 IRR15 (定时器比较匹配中断 1 标志) 后本位被清除。 0: 不产生 TCMR1 定时器的比较匹配 [清除条件] IRR15 (定时器比较匹配中断 1 标志) 写入 1 1: 产生 TCMR1 定时器的比较匹配 [置位条件] TTCR0 的 bit11 为 1 时，TCMR1 与周期时间匹配 (TCMR1=CYCTR)

位	位名称	初始值	R/W	说 明
1	TSR1	0	R*	定时器比较匹配标志 0 表示已产生定时器比较匹配寄存器 0 (TCMR0) 的比较匹配。TCMR0 的设定值与时间值匹配 (TCMR0=TCNTR) 时, 如果 TTCR0 的 bit10 设定为 1, 则本位置位。本位为只读, 清除 IRR14 (定时器比较匹配中断 0 标志) 后本位被清除。 0: 不产生 TCMR0 定时器的比较匹配 [清除条件] IRR14 (定时器比较匹配中断 0 标志) 写入 1 1: 产生 TCMR0 定时器的比较匹配 [置位条件] TTCR0 的 bit10 为 1 时, TCMR0 与时间值匹配 (TCMR0=TCNTR)
0	TSR0	0	R*	定时器溢出 /Next_is_Gap 接收 / 信息错误 本标志有 3 种不同功能, 分别表示在事件触发模式产生定时器溢出、在时间触发模式接收设置 Next_is_Gap 的时间基准信息及在测试模式的 CAN 总线检测出错误。测试模式的优先级高于其他设定。 0: 在事件触发模式不产生定时器 (TCNTR) 溢出 在时间触发模式不接收带 Next_is_Gap 的时间基准信息 在测试模式不产生信息错误 [清除条件] 对 IRR13 写入 1 1: [置位条件] 在事件触发模式产生定时器 (TCNTR) 溢出, 并从 H'FFFF 变为 H'0000 在时间触发模式接收带 Next_is_Gap 的时间基准信息 在测试模式产生信息错误

【注】 * 本位为只读, CPU 可监控周期计数器、定时器及比较匹配寄存器的状态。写入本位无效。

19.6.5 周期计数器寄存器（CCR）

CCR 为 16 位可读取 / 写入的寄存器。
表示时间触发发送的周期计数值（Basic Cycle）。

RCAN-TL1 作为潜在时间主控运行运行或作为时间从动运行，CCR 值的更新方法也不同，如下更新 CCR：

- RCAN-TL1 作为潜在时间主控运行时：
- 周期时间（CYCTR）与邮箱 30 的发送触发时间每匹配一次，则增加 +1
- 或
- 每接收一次有效基准信息，重写邮箱 31 的 MSG_DATA_0[5:0] 值

- RCAN-TL1 作为时间从动运行时：
- 每接收一次有效基准信息时，用邮箱 31 的 MSG_DATA_0[5:0] 值重写

- CMAX=B'111 且 TTCR0[15]=0 时：
- CCR 的值总是为 B'000000

位：	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CCR[5:0]					
初始值：	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W：	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15 ~ 6	—	均为 0	R	保留位 读取值、写入值总是为 0。
5 ~ 0	CCR[5:0]	均为 0	R/W	周期计数器寄存器 表示计数时间触发发送的矩阵周期的当前周期数。

19.6.6 定时器计数器寄存器（TCNTR）

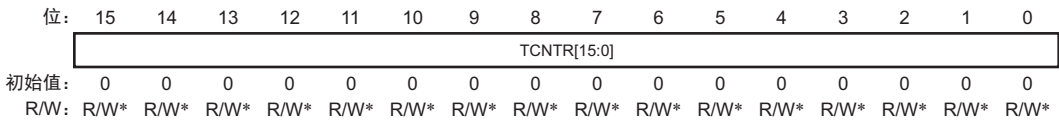
TCNTR 为 16 位可读取 / 写入的寄存器，是自由运行定时器计数器。设定 TTCR0[15]=1，定时器开始计数。定时器计数器的预分频器根据 RCAN-TL1 的触发模式而改变。

- 在事件触发模式（C_{MAX}=B'111）：TTCR0[5:0]的TPSC 值
- 在时间触发模式（C_{MAX}≠B'111）：CAN 总线 1 位时序

设定 TTCR0[6]=1 时，如果 TCNTR 的值与定时器比较匹配寄存器 0（TCMR0）匹配，则 TCNTR 清除为 H'0000，并递增计数。

在时间触发模式，TCNTR 作为本地时间（Local time）使用。因此，设定为 TTCR0[6]=0，并作为自由运行定时器运行。

- 【注】 1. TTCR0[15]=0 时，TCNTR 总是为 H'0000。
2. 设定定时器允许（TTCR0[15]=1）后到 TCNTR 开始计数，有多个时钟周期延迟，这是由预分频器的内部逻辑引起的。



- 【注】 * 仅在设定定时器允许（TTCR0[15]=1）后，可写入本寄存器。
- 在时间触发模式（C_{MAX} 不为 B'111）禁止写入。

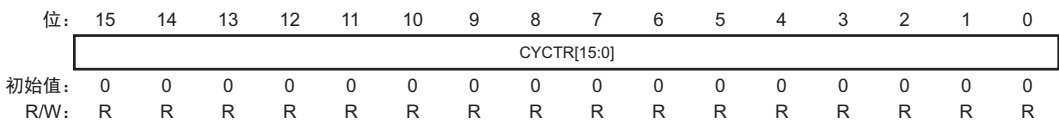
位	位名称	初始值	R/W	说 明
15 ~ 0	TCNTR[15:0]	H'0000	R/W	表示自由运行定时器的值。

19.6.7 周期时间寄存器（CYCTR）

CYCTR 为表示周期时间的 16 位只读寄存器。

Cycle_Time（CYCTR）=Local_Time（TCNTR）– Reference_Mark（RFMK）

在事件触发模式，RFMK 固定为 0，因此本寄存器为 TCNTR 的副本。



位	位名称	初始值	R/W	说 明
15 ~ 0	CYCTR[15:0]	H'0000	R	表示周期时间。

19.6.8 基准标记寄存器（RFMK）

RFMK 为 16 位只读寄存器。正常发送或接收基准信息时，本寄存器捕捉在基准信息 SOF 的本地时间（TCNTR）。在事件触发模式不使用本寄存器，总是清 0。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFMK[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名称	初始值	R/W	说 明
15 ~ 0	RFMK[15:0]	H'0000	R	基准标记寄存器 表示在时间基准信息 SOF 的 TCNTR 值。

19.6.9 定时器比较匹配寄存器 0 ~ 2（TCMR0 ~ 2）

TCMR0 ~ 2 为 16 位可读取 / 写入的寄存器。

可产生中断信号、清除定时器值（仅支持 TCMR0）、清除发送请求（仅支持 TCMR2）。

TCMR0 与 TCNTR 比较，TCMR1、TCMR2 与 CYCTR 比较。各比较匹配寄存器的比较值可独立设定。置位比较匹配的标志时，需置位 TTCR0 的 bit12 ~ 10。在时间触发模式，TCMR0 用于 Init_Watch_Trigger、TCMR2 用于 Watch_Trigger。

(1) 中断功能

设定 TTCR0 的 bit12、bit11、bit10，允许各寄存器的中断标志。

产生比较匹配时，置位 IRR 对应的中断标志（bit11、bit15、bit14）。

可由 IMR 的 bit11、bit15、bit14 抑制中断信号的产生。

产生比较匹配时，如果置位 IRR11（或 IRR15、IRR14），则定时器状态寄存器（TSR）的 bit3、bit2 和 bit1 随之置位。并且，清除 IRR 位时，也清除对应的 TSR 位。

(2) 定时器清除功能

仅 TCMR0 有通过比较匹配清除定时器值（TCNTR）的功能。设定 TTCR0[6]=1 时有效。TCMR1 与 TCMR2 无此功能。

(3) 取消已请求发送信息的功能

设定 TTCR0[13:12]=B'11，在停机模式之外产生 TCMR2 的比较匹配时，TCMR2 设定 TXPR 设定对应的 TXCR。因此，取消发送等待请求（正在发送的信息除外）。

TCMR1 与 TCMR0 无此功能。

• TCMR0

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TCMR0[15:0]															
初始值:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15 ~ 0	TCMR0[15:0]	H'FFFF	R/W	定时器比较匹配寄存器 表示产生比较匹配时的 TCNTR 值。

• TCMR1

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TCMR1[15:0]															
初始值:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15 ~ 0	TCMR1 [15:0]	H'FFFF	R/W	定时器比较匹配寄存器 表示产生比较匹配时的 CYCTR 值。

• TCMR2

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TCMR2[15:0]															
初始值:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15 ~ 0	TCMR2[15:0]	H'FFFF	R/W	定时器比较匹配寄存器 表示产生比较匹配时的 CYCTR 值。

19.6.10 发送触发时间选择寄存器（TTTSEL）

TTTSEL 为 16 位可读取 / 写入的寄存器，指定与周期时间比较匹配的发送触发时间。仅可置位 1 位，不可置位多位，并且必须清除所有位。本寄存器仅可在结构模式更改，更改时的算法如图 19.14 所示。

TTTSEL 为测试及诊断专用，通常运行时不得写入，并且不保证读取值。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	TTTSEL[14:8]							—	—	—	—	—	—	—	—
初始值:	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

【注】 仅可置位 1 位。

位	位名称	初始值	R/W	说 明
15	—	0	R	保留位 读取值、写入值总是为 0。
14 ~ 8	TTTSEL [14:8]	B'1000000	R/W	指定与 CYCTR 比较匹配的发送触发时间。bit14 ~ 8 对应邮箱 30 ~ 24。
7 ~ 0	—	均为 0	R	保留位 读取值、写入值总是为 0。

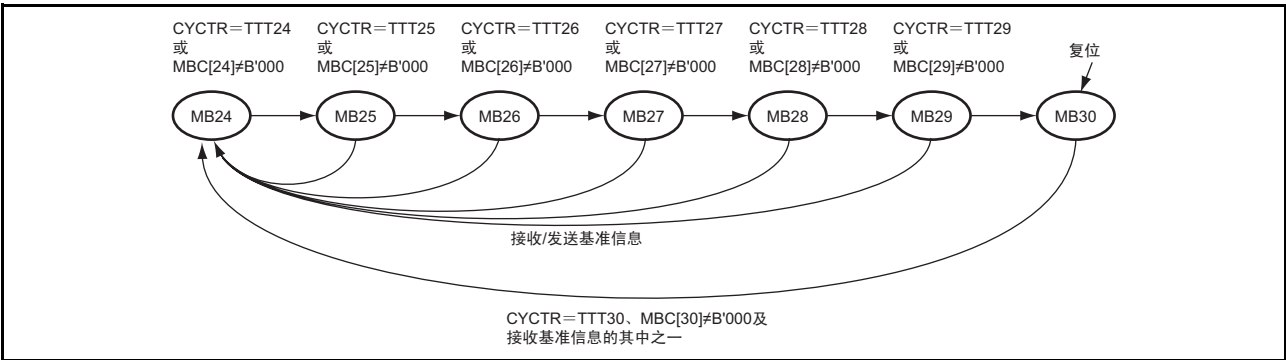


图 19.14 TTTSEL 更改算法

19.7 运行说明

19.7.1 RCAN-TL1 的设置

在本节说明硬件复位（上电复位）或软件复位（MCR0）后的结构模式及停机模式时的 RCAN-TL1 设定。无论哪种情况，RCAN-TL1 均不可加入 CAN 总线激活。并且，更改 RCAN-TL1 的设定不影响 CAN 总线的通信。

(1) 复位顺序

软件复位或硬件复位后的 RCAN-TL1 设定步骤如图 19.15 所示。复位后初始化所有寄存器。因此，加入 CAN 总线激活前需设定 RCAN-TL1，详情参照图中的“注”。

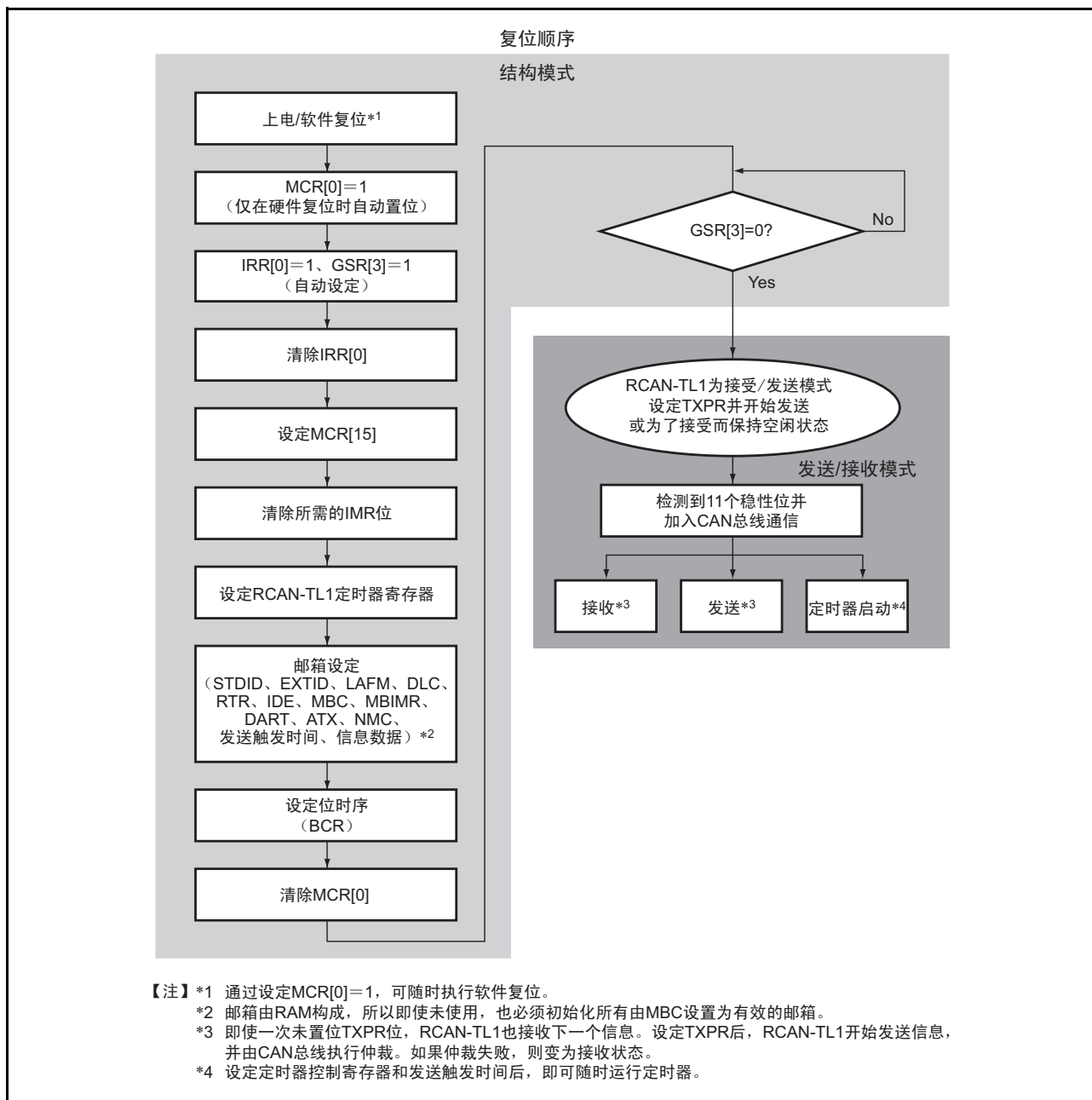


图 19.15 复位顺序

(2) 停机模式

RCAN-TL1 为停机模式时, 不可加入 CAN 总线激活。因此, 用户可在不影响 CAN 总线通信的情况下更改必要的寄存器设定。在此关键的是更改寄存器之前, 必须等待 RCAN-TL1 成为停机模式。不必立即向停机模式转移 (CAN 总线为空闲或暂停状态时转移)。RCAN-TL1 转移至停机模式后 GSR4 位置位。

寄存器设定结束后, 需解除停机请求。解除后 (MCR1=0 或 GSR4=0), RCAN-TL1 检测出 CAN 总线的 11 个隐性位后, 加入 CAN 总线激活。

(3) CAN 睡眠模式

在 CAN 睡眠模式, 停止 RCAN-TL1 的主要模块时钟以降低消耗电流, 仅可存取 MCR、GSR、IRR 及 IMR 寄存器。并且, 不可清除与发送和接收相关的中断, 因此必须预先清除。

RCAN-TL1 的 CAN 睡眠模式流程图如图 19.16 所示。

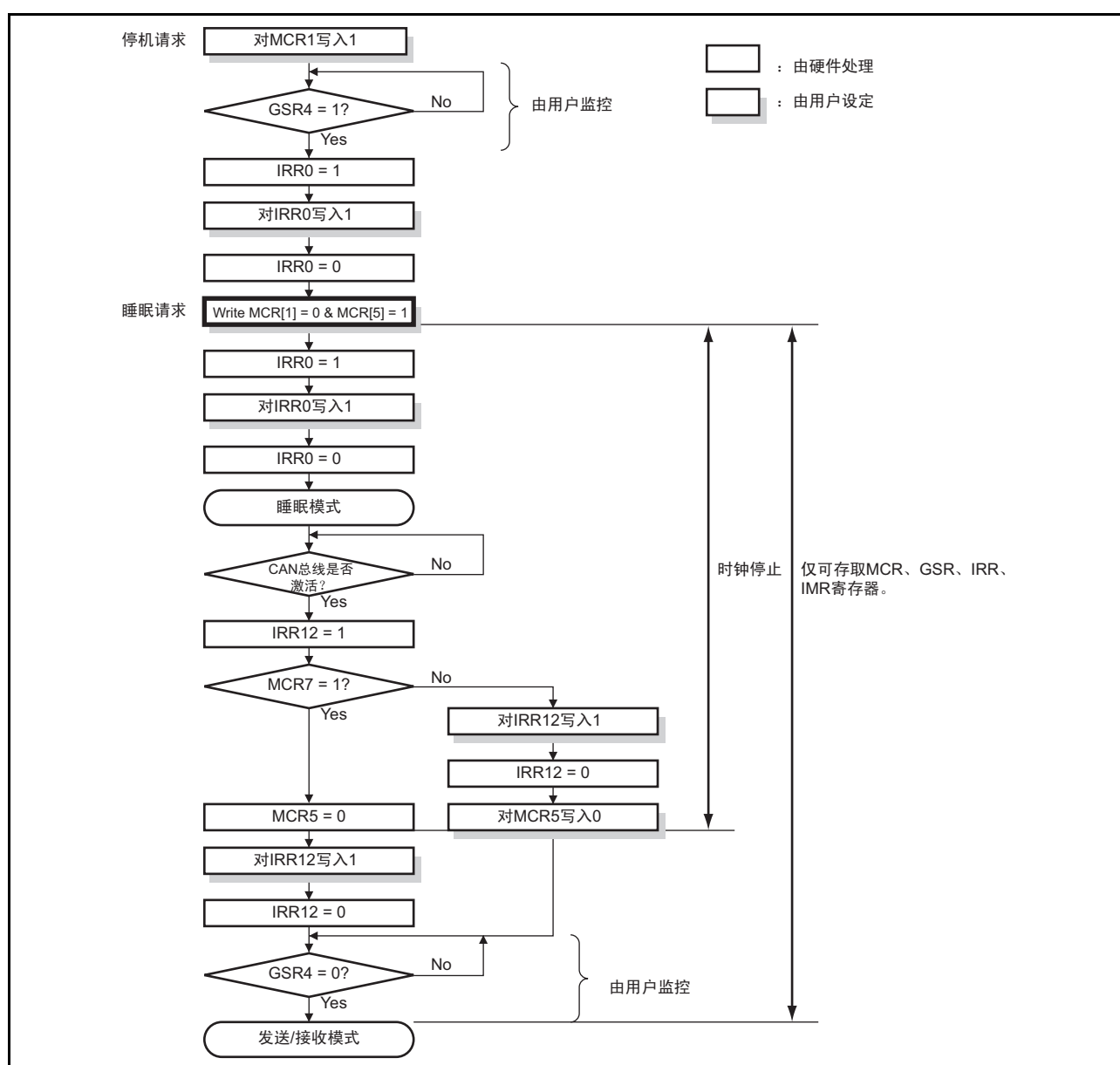


图 19.16 CAN 睡眠模式的流程图

可能的状态转移如图 19.17 所示。
除停机模式之外，不得置位 MCR5（CAN 睡眠模式）位。
并且，置位 MCR1 位后，清除 MCR1 前必须确认 GSR4 置位、并将 RCAN-TL1 设置为停机模式。

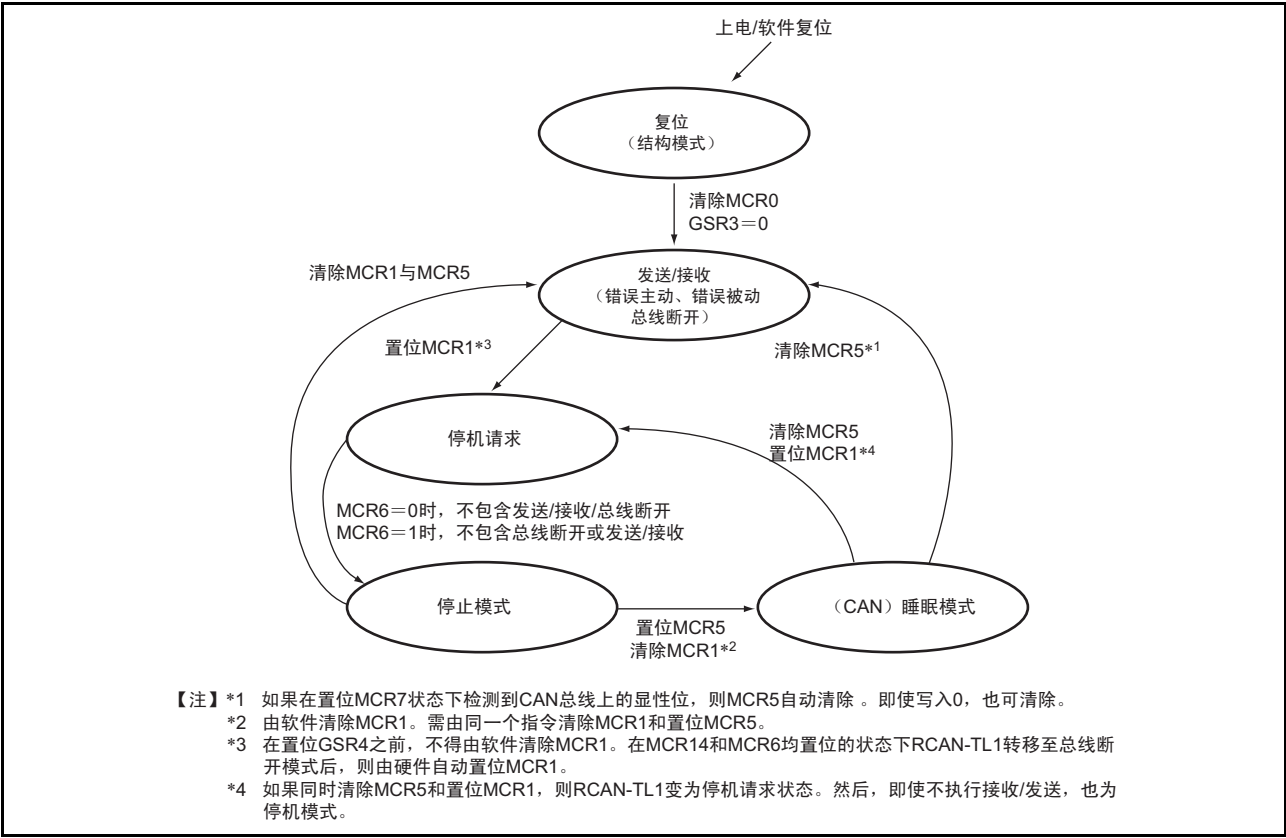


图 19.17 状态转移图

在各模式允许存取的条件如表 19.9 所示。

表 19.9 可存取的寄存器

状态模式	寄存器										
	MCR、 GSR	IRR、 IMR	BCR	MBIMR 定时器 TT 寄存器	标志 寄存器	邮箱（控制 0、 LAFM）		邮箱 （数据）	邮箱 （控制 1）		邮箱（触 发时间、 TT 控制）
复位	Yes	Yes	Yes	Yes	Yes	Yes		Yes	Yes		Yes
发送 / 接收	Yes	Yes	No	Yes	Yes	No	Yes *	Yes*	No	Yes *	Yes*
停机请求	Yes	Yes	No	Yes	Yes	No	Yes *	Yes*	No	Yes*	Yes*
停机模式	Yes	Yes	No	Yes	Yes	Yes		Yes	Yes		Yes
CAN 睡眠模 式	Yes	Yes	No	No	No	No		No	No		No

【符号说明】
Yes: 允许存取的寄存器
No: 禁止存取的寄存器
【注】 * TXPR0 未置位时

19.7.2 测试模式的设定

RCAN-TL1 有各种测试模式。由 MCR 寄存器的 TST[2:0] 位选择测试模式。初始值（默认）时，RCAN-TL1 以普通模式运行。

测试模式的设定如表 19.10 所示。

仅可在结构模式选择测试模式。执行已选的测试模式时，必须在选择后解除结构模式（确认 BCR0/BCR1 已设定）。

表 19.10 测试模式的设定

TST2	TST1	TST0	模式
0	0	0	普通模式
0	0	1	只听模式（只接收模式）
0	1	0	自测试模式 1（外部）
0	1	1	自测试模式 2（内部）
1	0	0	写错误计数器
1	0	1	错误被动模式
1	1	0	禁止设定
1	1	1	禁止设定

- 普通模式
RCAN-TL1 普通运行。
- 只听模式
该模式为 ISO-11898 标准要求的检测波特率的模式。错误计数器在清除后禁止运行，且 TEC/REC 的值不增加。另外，禁止 CTxn（n=0、1）输出，抑制由 RCAN-TL1 生成的错误帧或响应位。产生信息错误时，IRR13 置位。
- 自测试模式（外部）
RCAN-TL1 自身生成响应位，需要时将信息保存至 RCAN-TL1 接收邮箱。CRxn/CTxn（n=0、1）引脚必须连接至 CAN 总线。
- 自测试模式（内部）
RCAN-TL1 自身生成响应位，需要时将信息保存至 RCAN-TL1 接收邮箱。内部 CTxn（n=0、1）被循环反馈至内部 CRxn（n=0、1），因此无需将 CRxn/CTxn（n=0、1）引脚连接至 CAN 总线及其他外部器件。CTxn（n=0、1）引脚仅输出隐性位，CRxn（n=0、1）引脚无效。
- 写错误计数器
可在本模式写入 TEC/REC。通过对写错误计数器写入大于 127 的值，可强行将 RCAN-TL1 设置为错误被动模式。由于写入 TEC 的值被写入 REC，所以 TEC 与 REC 被设定为相同的值。同样，通过写入大于 95 的值，可强行将 RCAN-TL1 设置为错误警告模式。
写入 TEC/REC 时，RCAN-TL1 必须为停机模式（错误计数器写入时 MCR1=1）。并且根据 CAN 规格，必须在解除停机模式之前解除本测试模式，并重新运行 TEC/REC。
- 错误被动模式
可强行将 RCAN-TL1 设置为错误被动模式。
执行错误被动模式不改变 REC 的值，但是在该模式运行时，如果接收错误，则 REC 的值照常增加。
在本模式，如果 TEC 的值达到 256，则 RCAN-TL1 为总线断开状态，但是使用本模式时，RCAN-TL1 不能为错误主动状态。因此，RCAN-TL1 在总线断开返回顺序的最后，转移至错误被动状态，而非错误主动状态。

产生信息错误时，在所有测试模式 IRR13 位均置位。

19.7.3 信息发送顺序

(1) 信息发送请求

在总线发送 CAN 帧的顺序的例子如图 19.18 所示。

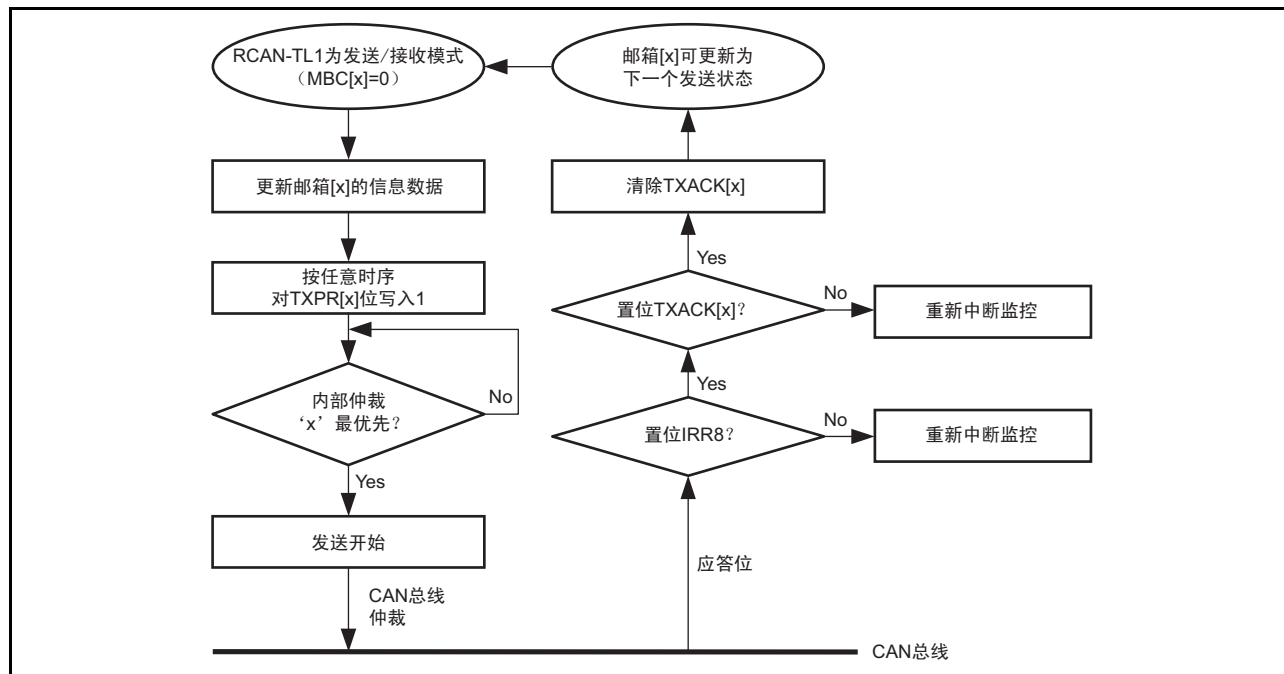


图 19.18 信息发送请求

注意：根据在寄存器章节的说明，如果 TXACK 或 ABACK 位的任意一位置位，则 IRR8 也置位。这意味着任意邮箱完成发送或发送中止，并可更新为下一个发送状态。另外，GSR2 表示当前未产生发送请求（所有 TXPR 标志均未置位）。

(2) 用于发送的内部仲裁

图 19.19 说明 RCAN-TL1 如何根据信息 ID 的顺序调度已请求发送的信息。通过内部仲裁获取优先级最高的已请求发送的信息。

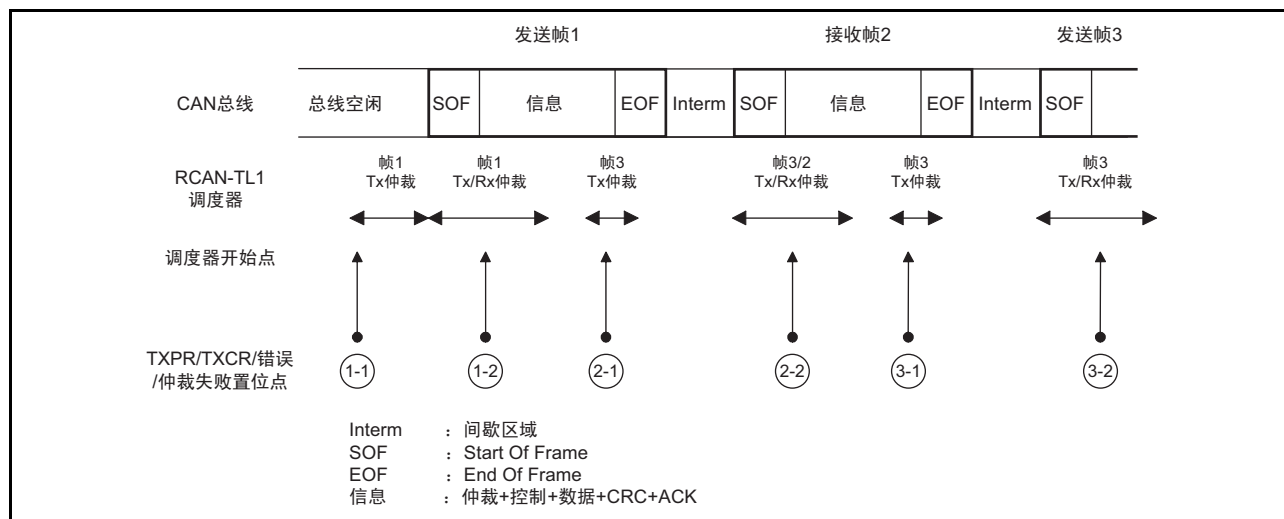


图 19.19 用于发送的内部仲裁

1-1: CAN 总线为空闲状态时，如果 TXPR 位置位，则立即执行内部仲裁，开始发送。

1-2: 通过 SOF 开始发送与接收。由于不为接收帧，所以 RCAN-TL1 执行发送。

2-1: 由 CRC 的定界符开始内部仲裁，并搜索下一条发送信息。

2-2: 通过 SOF 开始发送与接收。为高优先级的接收帧，所以 RCAN-TL1 执行接收，因此，不发送帧 3 而接收。

3-1: 由 CRC 的定界符开始内部仲裁，搜索下一条发送信息。

3-2: 通过 SOF 开始发送与接收。由于发送帧的优先级比接收帧高，所以 RCAN-TL1 执行发送。

在 CAN 总线检测出错误时，也可在各错误定界符的起始点执行用于下一次发送的内部仲裁，还可在超载帧后的错误定界符的起始点执行内部仲裁。

可 CRC 的定界符执行发送仲裁，因此 ATX=1 的邮箱在接收远程帧请求时，根据该请求，发送的信息加入用于发送的仲裁时序，为此后的总线空闲、CRC 定界符或错误定界符。

根据 CAN 总线状态，可执行从 TXCR 置位后到最大延迟 1CAN 帧后对应信息的中止处理。

(3) 时间触发发送

RCAN-TL1 通过硬件支持符合 ISO-11898-4 TTCAN Level 1 标准的时间触发模式通信。
本节说明使用时间触发模式的基本步骤。

• 时间触发模式的设定

设置时间触发模式需要以下设定：

- 将 CMAX_TEW 的 CMAX 设定为 B'111 以外的值
- 置位 TTCR0 的 bit15，开始 TCNTR 运行
- 清除 TTCR0 的 bit6，以防止通过比较匹配清除 TCNTR
- 为了定期发送，不可清除时间触发发送邮箱的 TXPR。

• 各寄存器的作用

使用 RCAN-TL1 的用户寄存器，可处理 TTCAN 协议要求的主要功能。

TCNTR	Local_Time
RFMK	Ref_Mark
CYCTR	周期时间 = TCNTR - RFMK
RFTROFF	邮箱 30 的 Ref_Trigger_Offset
邮箱 31	时间基准信息专用接收邮箱
邮箱 30	时间基准信息专用发送邮箱 (作为潜在时间主控运行时)
邮箱 29 ~ 24	时间触发发送对应的邮箱
邮箱 23 ~ 16	无时戳的接收对应的邮箱 (也可用作事件触发发送邮箱。)
邮箱 15 ~ 0	带时戳的接收对应的邮箱 (也可用作事件触发发送邮箱。)
Tx-Trigger Time	指定发送信息时序的 Time_Mark
CMAX	指定计数作为潜在时间主控运行时的周期 (Basic Cycle) 的最大值
TEW	指定 Tx_Enable_Window 宽度
TCMR0	Init_Watch_Trigger (与本地时间比较匹配)
TCMR1	与用于用户指定事件监控的周期时间比较匹配
TCMR2	Watch_Trigger (与周期时间比较匹配) 可设定为全部中止处于等待状态的发送。
TTW	指定用于发送的时间窗的属性
TTTSEL	指定下一个发送等待邮箱

• 时间主控模式/时间从动模式

RCAN-TL1 可设定为网络的潜在时间主控模式或时间从动模式运行。各模式所需的设定与 RCAN-TL1 自动执行的处理如下表所示：

模式	设定	功 能
时间从动模式	$\text{TXPR}[30]=0$ $\&$ $\text{MBC}[30]\neq\text{B}'000$ $\&$ $\text{CMAX}\neq\text{B}'111$ $\&$ $\text{MBC}[31]=\text{B}'011$	每次在 CAN 总线检测出 SOF 时，采样 TCNTR，并保存至内部寄存器。邮箱 31 接收到有效时间基准信息时，将 TCNTR 的值（由 SOF 保存）复制到 Ref_Mark（RFMK）。 将接收的基准信息的周期计数器值（CCR）复制到自节点的周期计数器寄存器（CCR）。 如果 Next_is_Gap=1 则置位 IRR13。
（潜在） 时间主控模式	$\text{TXPR}[30]=1$ $\&$ $\text{MBC}[30]=\text{B}'000$ $\&$ $\text{DLC}[30]>0$ $\&$ $\text{CMAX}\neq\text{B}'111$ $\&$ $\text{MBC}[31]=\text{B}'011$	有以下两种情况： （1）邮箱 31 接收到有效的的时间基准信息时，将通过 SOF 保存至内部寄存器的 TCNTR 值复制到 Ref_Mark（RFMK），接收基准信息的周期计数器值（CCR）复制到自节点的周期计数器寄存器（CCR），如果 Next_is_Gap=1 则置位 IRR13。 （2）从邮箱 30 发送时间基准信息时，将通过 SOF 保存至内部寄存器的 TCNTR 值复制到 Ref_Mark（RFMK）。邮箱 30 的 TTT 与 CYCTR 匹配时，周期计数器（CCR）递增。 周期计数器（CCR）嵌入时间基准信息的起始数据字节（Data0[7:6]、CCR[5:0]）。

• 发送触发时间的设定

必须按照以下升序设定发送触发时间（TTT）。TTT 之间的差异需满足以下公式，公式中的 TEW 为寄存器值。

$$\begin{aligned}
 &\text{TTT (邮箱 24)} < \text{TTT (邮箱 25)} < \text{TTT (邮箱 26)} < \\
 &\text{TTT (邮箱 27)} < \text{TTT (邮箱 28)} < \text{TTT (邮箱 29)} < \\
 &\text{TTT (邮箱 30)} \\
 &\text{且} \\
 &\text{TTT (邮箱 } i) - \text{TTT (邮箱 } i-1) > \text{TEW} + \text{最大帧长度} + 9
 \end{aligned}$$

作为潜在时间主控模式运行时，TTT（邮箱 24）～TTT（邮箱 29）对应 Time_Mark；TTT（邮箱 30）对应表示 Basic_Cycle length 的 Time_Ref。

上述规定仅适用于被设定为时间触发发送的邮箱。

【重点】 根据发送触发时间设定的限制，1 个时间窗仅有 1 个可分配的邮箱。

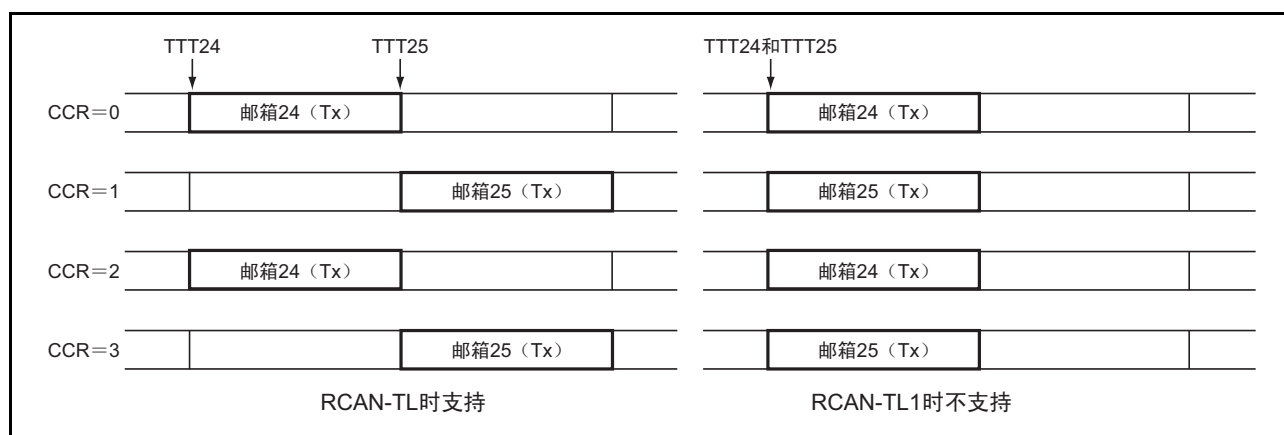


图 19.20 发送触发时间的限制

作为 Watch_Trigger 的 TCMR2 的值必须大于表示 1Basic_Cycle length 的 TTT (邮箱 30)。

(潜在) 时间主控及时间从动的设定例如图 19.21、图 19.22 所示, 图中的 L 为时间基准信息的时间长度。

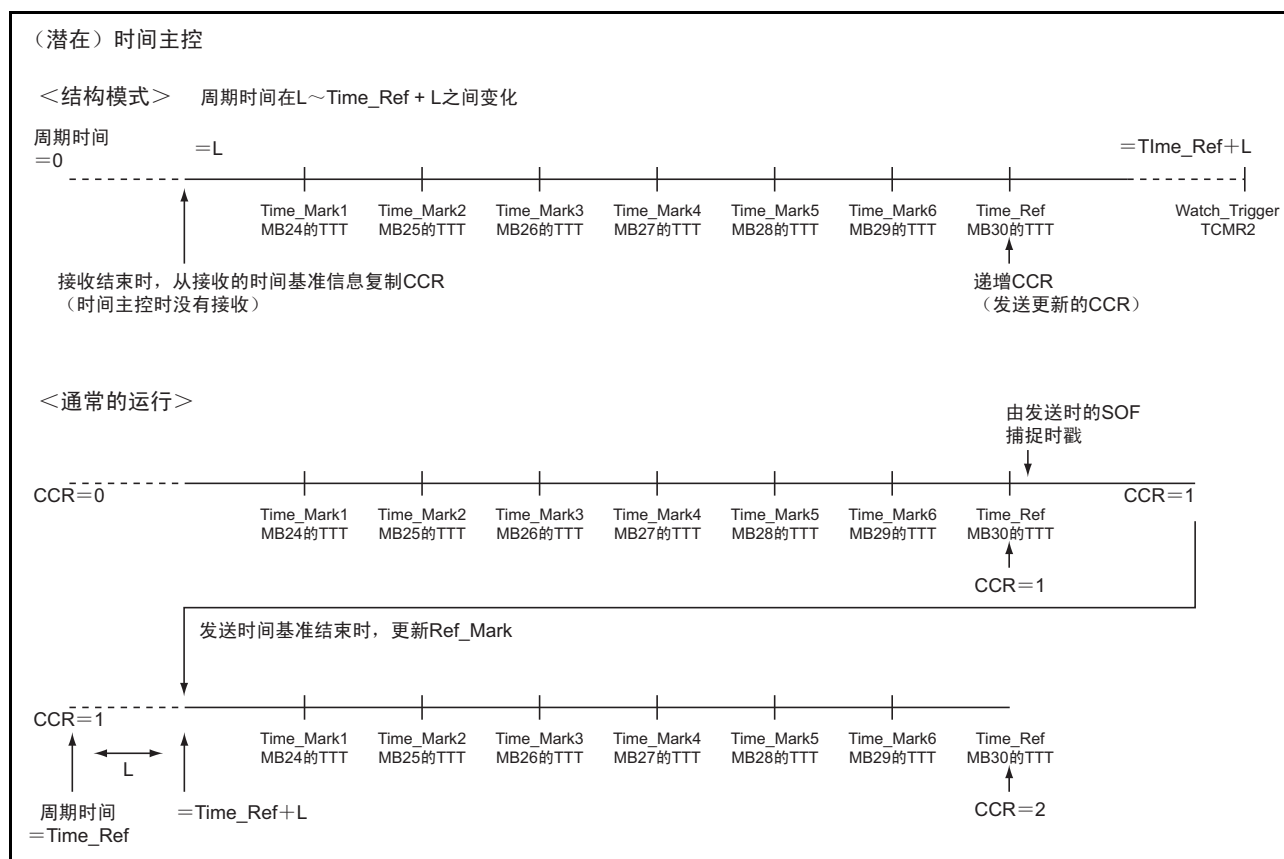


图 19.21 (潜在) 时间主控

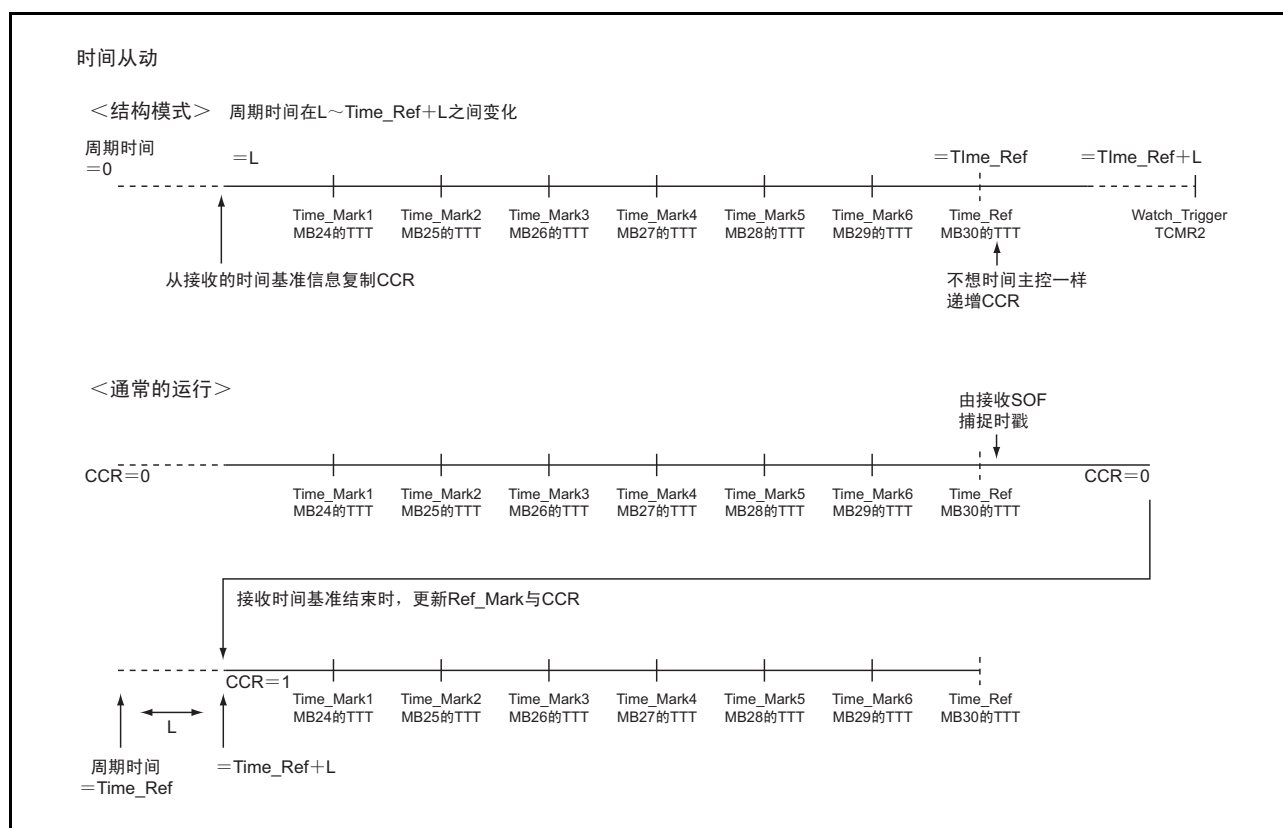


图 19.22 时间从动

- 通过软件安装的功能

TTCAN 的功能中有需要通过软件安装的功能。主要功能如下所示，详情参阅 ISO-11898-4。

- 从 Init_Watch_Trigger 更改为 Watch_Trigger

作为 Init_Watch_Trigger 及 Watch_Trigger 的硬件支持，RCAN-TL1 分别备有 TCMR0 寄存器与 TCMR2 寄存器。在 CAN 总线检测出初始基准信息之前，设置 TCMR0 为有效、TCMR2 为无效，之后由软件控制 TCMR0 无效、TCMR2 有效（日程同步状态机）。

仅支持 Next_is_Gap 接收中断。通过应用程序将对应的 TXCR 标志置位，当前的 Basic_Cycle 结束时，需停止所有发送。

主 / 从属模式控制

仅支持自动周期时间同步与 CCR 递增。

- 信息状态计数

需由软件计数独占窗的定期信息相关的日程错误。

- 时间触发通信的信息发送请求

使用时间触发模式时，必须满足 ISO11898-4 标准要求的事项。

必须执行以下步骤：

1. 将RCAN-TL1设置为复位或停机模式
2. 将Init_Watch_Trigger (H'FFFF)设定至TCMR0
3. 通过TTCR0的bit10，设定TCMR0产生的比较匹配为有效
4. 将指定的Watch_Trigger值设定至TCMR2
5. TTCR0的bit12保持为0，设定TCMR2产生的比较匹配无效
6. 对CMAx设定所需的值（除B'111之外）
7. 对TEW设定所需的值
8. 将邮箱设定为时间触发发送及接收
9. 对低3位设定邮箱31的LAFM
10. 对MCR、BCR1、BCR0设定所需的值
11. 作为潜在时间主控模式运行时设定如下：
 - 对RFTROFF设定所需的Init_Ref_Offset值
 - 置位邮箱30的TXPR
 - 对TTTSEL写入H'4000
12. 通过TTCR0的bit15允许定时器TCNTR
13. 转移至普通模式
14. 等待接收或发送有效基准信息，或等待TCMR0的比较匹配
15. 如果本地时间（TCNTR）与TCMR0的值匹配，则表示到达Init_Watch_Trigger，并通过应用程序将邮箱30的TXCR置位后重新开始
16. 如果发送基准信息（置位TXACK[30]），则将RFTROFF设定为0
17. 如果接收有效基准信息（置位RXPR[31]），则执行以下操作：
 - 如果邮箱31的ID低3位的优先级比邮箱30的低3位高（作为潜在时间主控模式运行时），则将RFTROFF的值保持为Init_Ref_Offset
 - 如果邮箱31的ID低3位的优先级比邮箱30的低3位低（作为潜在时间主控模式运行时），则RFTROFF的值递减1
18. 清除TTCR0的bit10，使TCMR0产生的比较匹配无效
19. 置位TTCR0的bit12，使TCMR2产生的比较匹配有效
20. 在CAN总线检测出2个基准信息（发送或接收）后，可通过应用程序置位其他时间触发邮箱的TXPR。

在 CAN 总线未检测出基准信息，且周期时间 CYCTR 到达 TCMR2 值时，RCAN-TL1 随时自动中止所有等待状态的发送（包含基准信息）。

在时间触发模式再次请求发送的顺序如下所示：

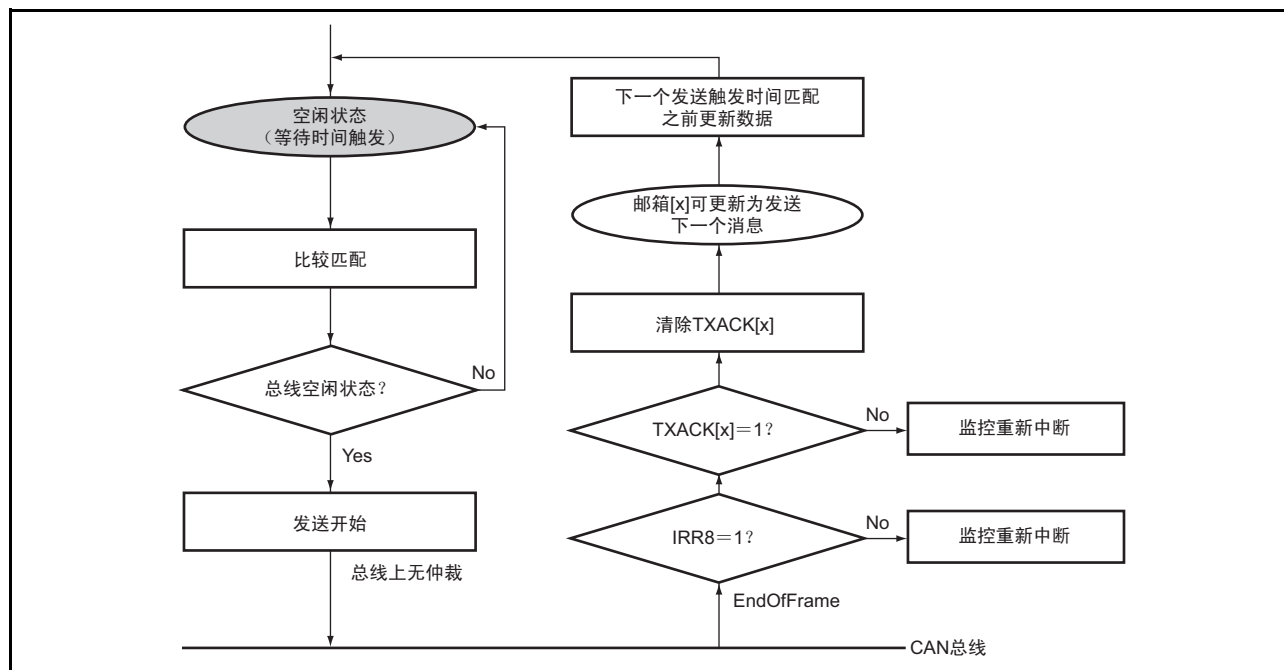


图 19.23 信息发送请求

产生发送触发之前，软件必须正确更新信息。CYCTR 到达邮箱的 TTT（发送触发时间）、CCR 与已编程的发送周期匹配时，RCAN-TL1 立即向发送缓冲器传送信息。

此时，RCAN-TL1 在指定的 Tx_Enable Window 中尝试发送。如果错过此时间槽，则定期发送（邮箱 24～30）时，RCAN-TL1 保持对应的 TXPR 位为 1，并在下一个发送触发之前保留发送请求。

RCAN-TL1 错过上述时间槽的原因有 3 种。

1. CAN 总线正在使用
2. 发送时间触发信息时，CAN 总线产生错误
3. 发送时间触发信息时产生仲裁失败

合并仲裁窗（Merged Arbitrating Window）时，此发送槽从已开始窗（TTW=B'10）邮箱的发送触发开始，到结束窗（TTW=B'11）邮箱的 TEW 末尾为止。TXPR 可随时更改。RCAN-TL1 总是按正确日程发送时间触发信息，但为了保证正确的日程，还规定了以下重要规则：

- 在结构模式可更改 TTT（发送触发时间）。
- 不可设定超过指定 Basic_Cycle length 周期计数器（Basic Cycle）长度的 Time_Ref 的 TTT，否则调度会出现问题。
- 定期发送时，不自动清除 TXPR。需取消定期发送时，通过应用程序置位对应的 TXCR 位。

• 时间触发系统的例子

使用时间从动模式的 RCAN-TL1，运行时间触发系统的简例如下图所示：

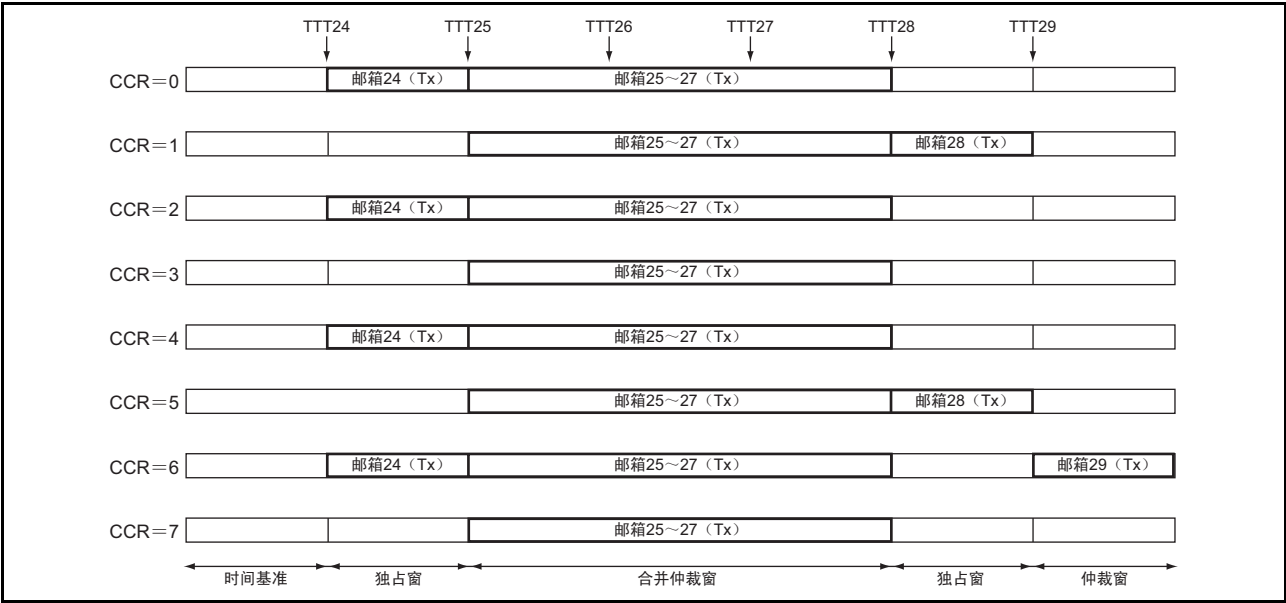


图 19.24 作为时间从动模式运行的时间触发系统例

图 19.24 中使用的值如表 19.11 所示。

表 19.11 例中使用的设定值

	rep_factor（寄存器）	Offset		TTW[1:0]	MBC[2:0]
邮箱 24	B'001	B'000000		B'00	B'000
邮箱 25	B'000	B'000000		B'10	B'000
邮箱 26	B'000	B'000000		B'10	B'000
邮箱 27	B'000	B'000000		B'11	B'000
邮箱 28	B'010	B'000001		B'00	B'000
邮箱 29	B'011	B'000110		B'01	B'000
邮箱 30	—	—		—	B'111
邮箱 31	—	—		—	B'011

【注】 CMAX=B'011,TXPR[30]=0

合并仲裁窗中，时间触发发送请求通过 FCFS（First Come First Served）进行处理。例如：邮箱 25 不可在发送触发时间 25（TTT25）与发送触发时间 26（TTT26）之间发送时，在 TTT26-TTT28 之间，邮箱 25 的优先级高于邮箱 26。

将 MBC 设定为 B'111，可使时间触发发送无效。RCAN-TL1 为时间主控模式时，必须设定 MBC[30]=B'000，时间基准窗自动识别为仲裁窗。

• 定时器运行

定时器的时序如图 19.25 所示。设定发送触发时间 =n 时，时间触发发送在 CYCTR= n+2 ~ CYCTR= n+3 之间开始。

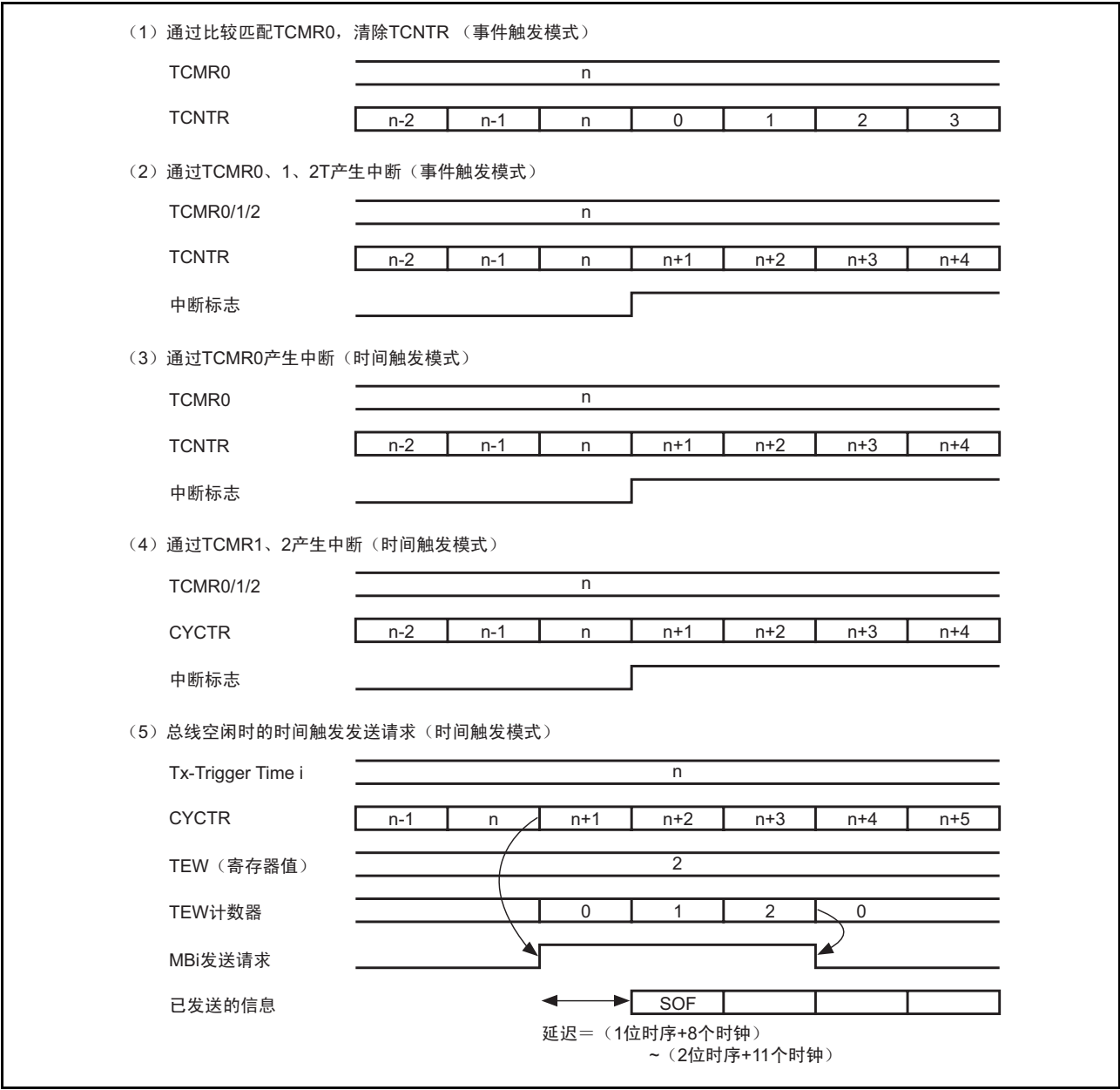


图 19.25 定时器的时序图

合并仲裁窗中，在时间触发送完成后处理事件触发送。例如：邮箱 25 的发送完成时，如果 CYCTR 未到达 TTT26，则通过 MCR2 所指定信息发送的优先顺序，开始事件触发送。发送完成后，不清除时间触发送的 TXPR，但清除事件触发送的 TXPR。

关闭合并仲裁窗的邮箱的 TXPR 未置位时，在此邮箱的 TTT 之后的 TEW 末尾，关闭合并仲裁窗。

详情参阅“19.3.1 邮箱结构”的表 19.3。

19.7.4 信息接收顺序

信息接收顺序如图 19.26 所示。

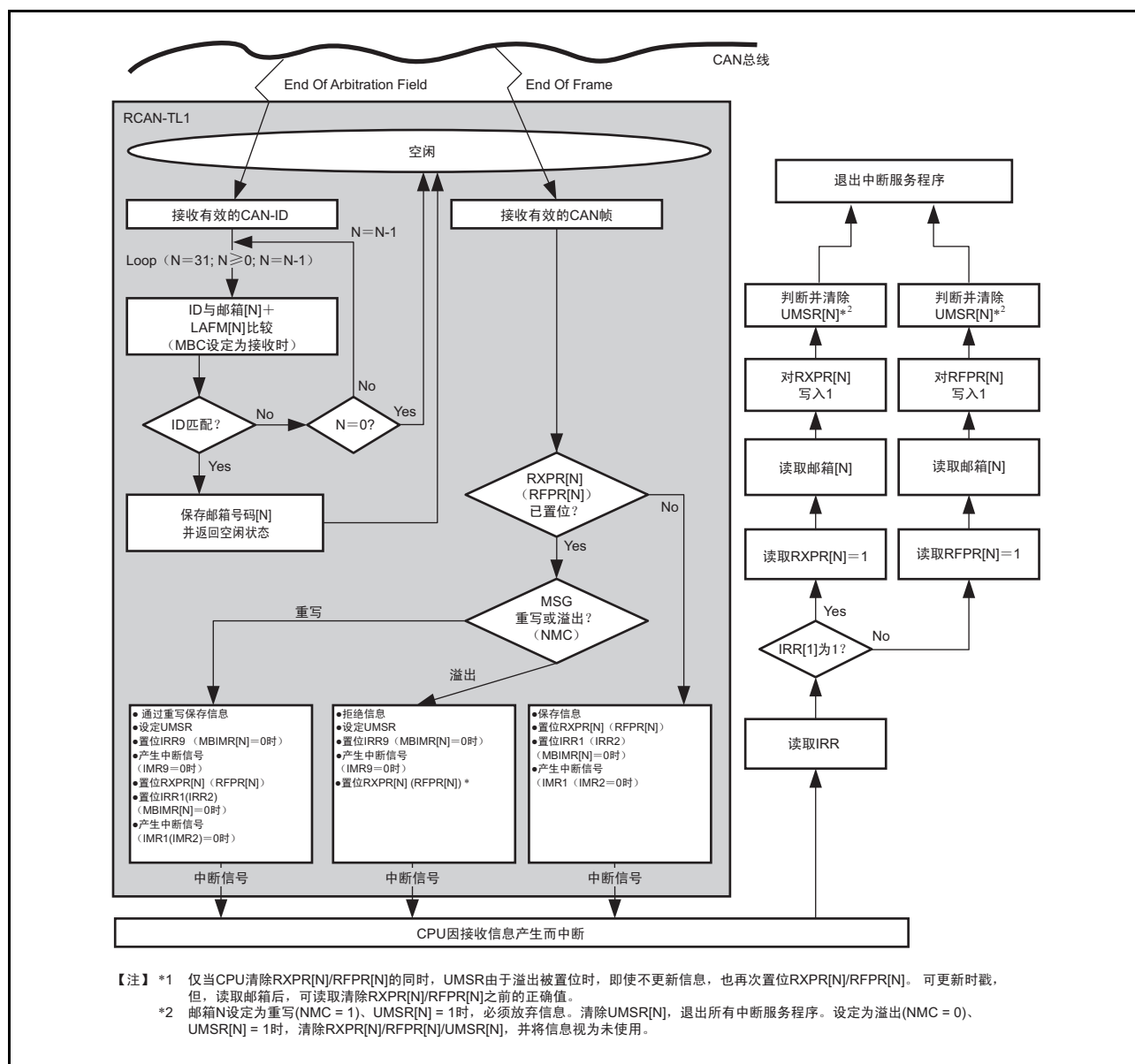


图 19.26 信息接收顺序

正在接收信息时，如果 RCAN-TL1 识别到仲裁字段的末尾，则开始比较已接收的 ID 与邮箱所设定的 ID。比较顺序为从邮箱 31 到邮箱 0。首先检查 MBC，确认邮箱是否设定为接收。之后读取 LAFM，接着读取邮箱 31（设定为接收时）的 ID 并与已接收的 ID 比较。如果不匹配，则对邮箱 30（设定为接收时）执行相同的检查。发现匹配的 ID 时，RCAN-TL1 将其邮箱编号（N）保存至内部缓冲器，停止搜索后，返回空闲状态，等待 EOF（End Of Frame）。通知 EOF 的第 6 位时，根据 NMC 位的设定写入或废除接收信息。

正在通信时不可更改 RCAN-TL1 的信息 ID 和 LAFM 的设定。停机模式及结构模式为更改其设定的方法之一。将接收信息写入对应的邮箱时，其中包含信息 ID，因此使用 LAFM 时，可能由不同的 CAN-ID 重写。同时，也说明在已接收信息的 ID 与多个邮箱的 ID + LAFM 匹配时，通常将接收信息保存在编号最大的邮箱，较小编号的邮箱不可保存信息。因此，必须仔细选择 ID 与 LAFM 的设定值。

如图 19.11 所示，接收数据及远程帧，并读取 IRR 后，清除 UMSR 标志，这是由于检测出执行中断服务程序时信息被保存在相同邮箱的新信息重写（NMC 为 1 时）。UMSR 的最终检查时如果检测出重写，则需废弃此信息重新读取。

邮箱设定为溢出（NMC=0）时，UMSR 置位时的信息有效。但 CAN 总线所监控的并非最新信息，而是旧信息。信息的读取必须在清除相关 RXPR/RFPR 标志前进行。

请注意：由数据帧重写已接收的远程帧时，远程帧接收中断（IRR2）及数据帧接收中断（IRR1）置位，接收标志（RXPR、RFPR）也置位。同样，由远程帧重写数据帧时，IRR2 与 IRR1 也置位。

接收信息后保存至邮箱时，将所有未接收的数据字段作为 0 保存，接收标准 ID 时也相同。如果为扩展 ID（EXTID[17:0]）则写入 0。

19.7.5 邮箱的重新设定

需重新设定邮箱时，请按以下步骤进行：

(1) 更改发送邮箱的设定

有以下 2 种情况：

- 更改 ID、RTR、IDE、LAFM、数据、DLC、NMC、ATX、DART
仅可在 MBC=B'000 时更改。必须确认对应的 TXPR 位未置位。可随时更改设定（MBC 除外）。
- 从发送邮箱更改为接收邮箱
必须确认对应的 TXPR 位未置位。仅可在停机模式或复位状态更改。RCAN-TL1 正在接收或发送信息时，转移至停机状态可能需要花费时间（等待接收/发送结束后才转移至停机状态）。另外，必须注意在停机状态不可接收/发送信息。
RCAN-TL1 为总线断开状态时，根据 MCR 寄存器的 bit6 及 bit14 的设定，向停机状态转移。

(2) 更改接收邮箱的 ID、RTR、IDE、LAFM、数据、DLC、NMC、ATX、DART、MBC 的设定及从接收邮箱更改为发送邮箱

仅可在停机模式更改设定。信息存在于 CAN 总线且 RCAN-TL1 为接收模式时不会丢失。RCAN-TL1 完成当前接收后转移至停机模式。RCAN-TL1 正在接收或发送信息时，虽可转移至停机状态但需要时间（等待接收 / 发送结束后才转移至停机状态）。另外，必须注意在停机状态不可接收 / 发送信息。

RCAN-TL1 为总线断开状态时，根据 MCR 寄存器的 bit6 及 bit14 的设定，向停机状态转移。

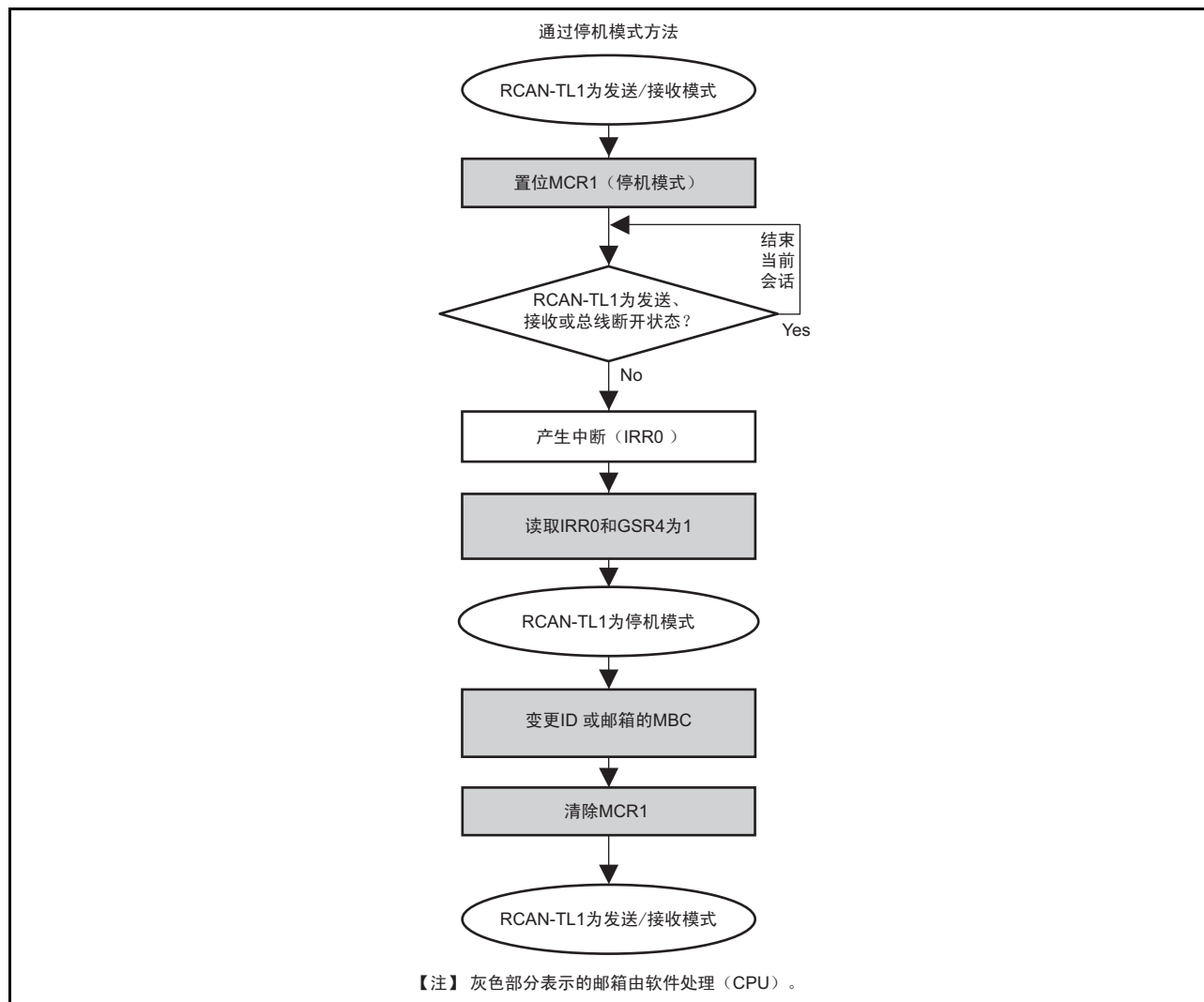


图 19.27 接收邮箱的 ID 更改 / 从接收邮箱更改为发送邮箱

19.8 中断源

RCAN-TL1 的中断源如表 19.12 所示。可使用邮箱中断屏蔽寄存器（MBIMR）及中断屏蔽寄存器（IMR）屏蔽这些中断源。有关各中断请求的中断向量，详情参阅“第 6 章 中断控制器（INTC）”。

表 19.12 RCAN-TL1 的中断源

模块名称	名称	中断源	中断标志	DMAC 的启动
RCANn*1	ERSn*1	错误被动（ $TEC \geq 128$ 或 $REC \geq 128$ ）	IRR5	不可
		总线断开（ $TEC \geq 256$ ）/ 从总线断开返回	IRR6	
		错误警告（ $TEC \geq 96$ ）	IRR3	
		错误警告（ $REC \geq 96$ ）	IRR4	
	OVRn*1	复位 / 停机 / CAN 睡眠转移	IRR0	
		发送超载帧	IRR7	
		未读信息的重写（溢出）	IRR9	
		开始系统矩阵	IRR10	
		TCMR2 比较匹配	IRR11	
		在 CAN 睡眠时检测出 CAN 总线运行	IRR12	
		定时器溢出 / Next_is_Gap / 信息错误	IRR13	
		TCMR0 比较匹配	IRR14	
		TCMR1 比较匹配	IRR15	
	RMn0*1*2	接收数据帧	IRR1*3	可 *4
	RMn1*1*2	接收远程帧	IRR2*3	
	SLEn*1	发送信息 / 取消发送（空槽）	IRR8	不可

【注】 *1 n=0、1

*2 RM0 为由邮箱 0 的远程帧接收标志（RFPR0[0]）或数据帧接收标志（RXPR0[0]）产生的中断，RM1 为由邮箱 n（n=1～31）的远程帧接收标志（RFPR0[n]）或数据帧接收标志（RXPR0[n]）产生的中断。

*3 IRR1 为邮箱 0～31 的数据帧接收标志，IRR2 为邮箱 0～31 的远程帧接收标志。

*4 仅 RMn0 中断可启动 DMAC。

19.9 DMAC 接口

RCAN-TL1 的邮箱 0 接收信息时, 可启动 DMAC。并且, 设定 DMAC 启动后, 通过 DMAC 传送结束时, 自动清除 RXPR0 与 RFPR0 寄存器标志。此时为来自 RCAN-TL1 的接收中断, 不向 CPU 产生中断。DMAC 的传送流程如图 19.28 所示。

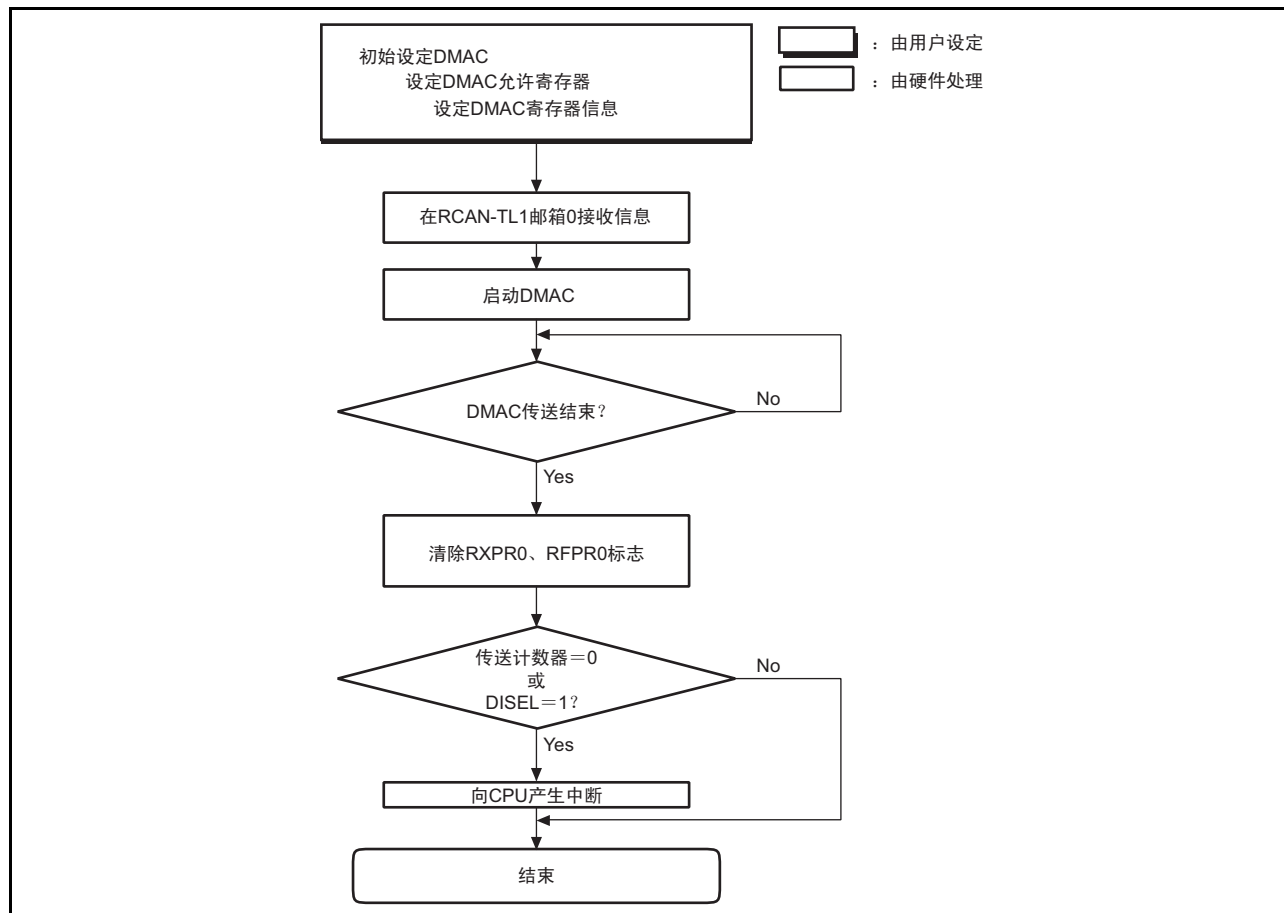


图 19.28 DMAC 的传送流程图

19.10 CAN 总线接口

本 LSI 与 CAN 总线的连接需要总线收发器 IC，推荐使用瑞萨 HA13721。使用其他产品时，必须与 HA13721 兼容。连接例如图 19.29 所示。

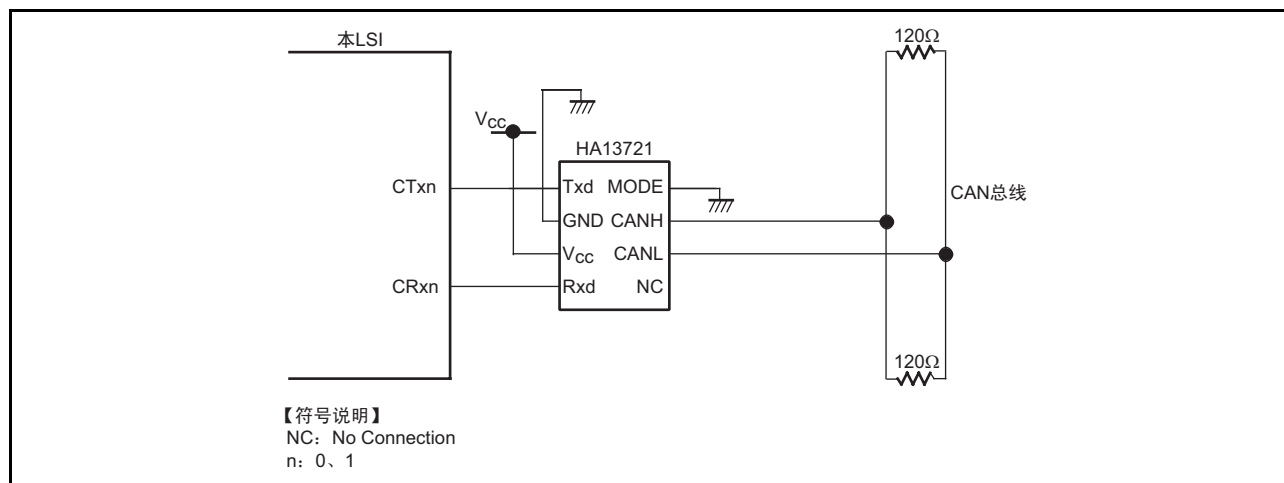


图 19.29 使用 HA13721 的高速 CAN 接口

19.11 RCAN-TL1 引脚端口设定

必须在结构模式或在此之前设定 RCAN-TL1 端口，有关端口设定方法，详情参阅“第 25 章 引脚功能控制器 (PFC)”。本 LSI 内置 3 个通道的 RCAN-TL1，有 2 种使用方法。

- 2 个通道的 32 个邮箱 (RCAN0、1)
- 1 个通道的 64 个邮箱 (RCAN0+1)

【注】 使用 64 个邮箱时，必须阅读“19.12.1 设定 1 个通道 64 个邮箱的端口时的注意事项”。

各端口设定的连接例如图 19.30、图 19.31 所示。

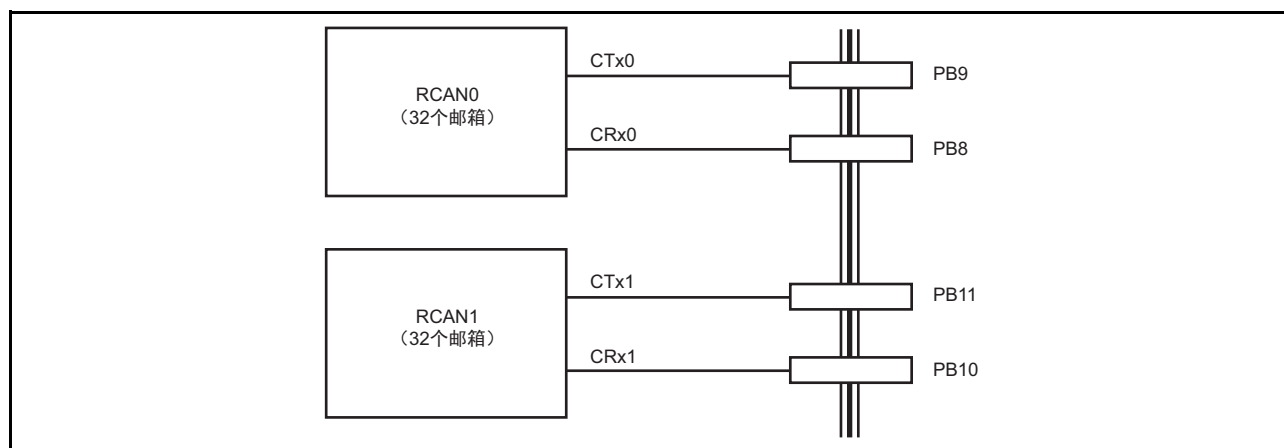


图 19.30 RCAN0、1 作为个别通道使用时的连接例

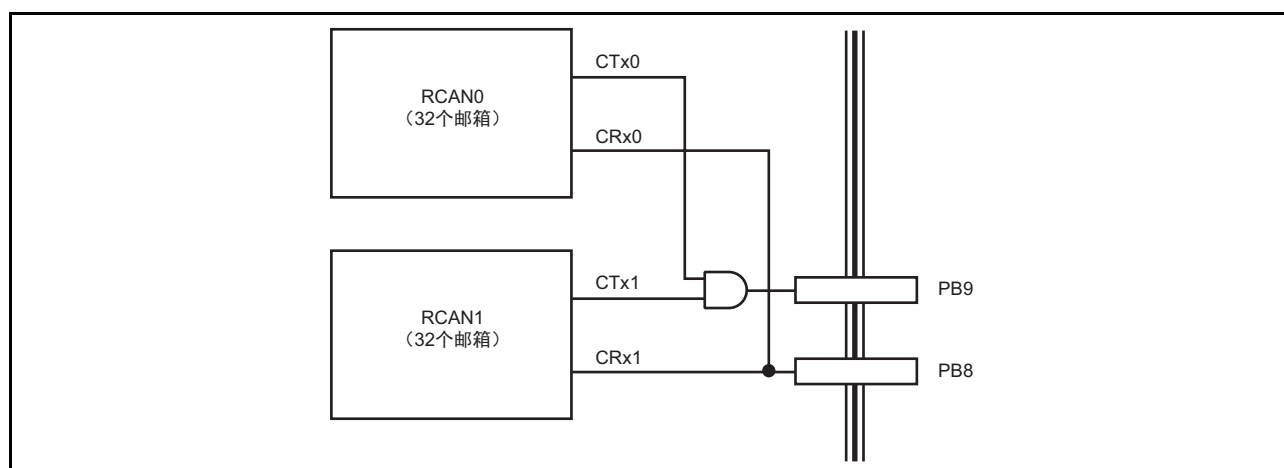


图 19.31 RCAN0、1 作为 64 个邮箱的 1 个通道使用时的连接例

19.12 使用时的注意事项

19.12.1 设定 1 个通道 64 个邮箱的端口时的注意事项

本 LSI 内置 2 个通道的 RCAN-TL1。设定为 1 个通道 64 个邮箱使用时，有以下注意事项：

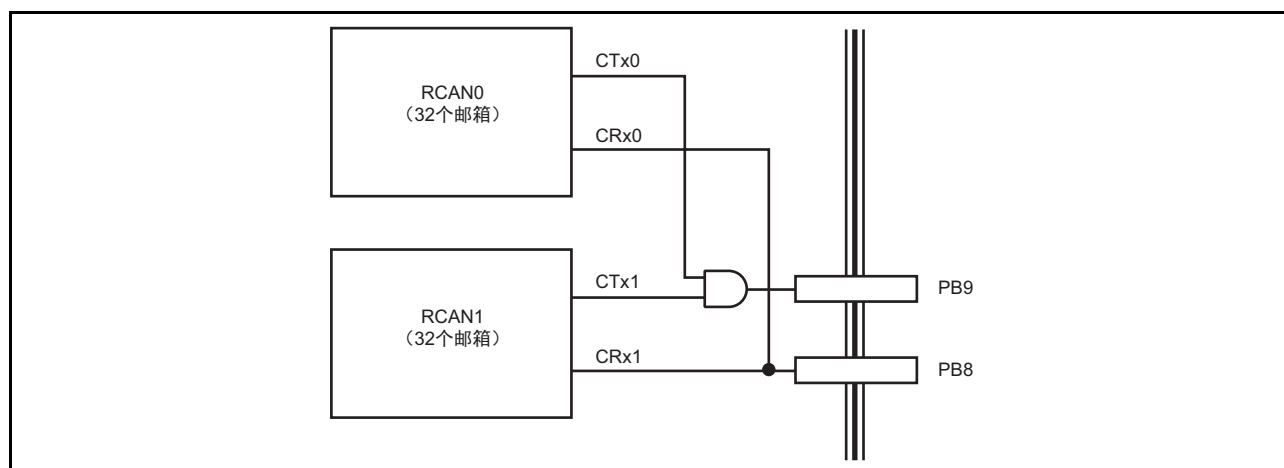


图 19.32 RCAN0、1 作为 64 个邮箱的 1 个通道使用时的连接例

1. 在 CAN 总线未连接其他节点的状态发送信息时，不产生 ACK 错误。这是由于从上图的 RCAN0 发送信息时，RCAN1 在 ACK 字段发送 ACK。RCAN1 接收 CAN 总线的信息，根据 CAN 协议，在 ACK 字段发送 ACK，RCAN0 接收此 ACK。
对应的方法为，将不发送信息的 RCAN1 设置为测试模式的只听模式（TST[2:0]=B'001）或复位状态（设定 MCR0=1）。这样就不会从不发送信息的通道发送 ACK。
2. 在 RCAN0 和 RCAN1 独立实施决定发送顺序的内部仲裁。可发送的缓冲器，在 RCAN-TL1 有 31 个邮箱 / 通道，但在 62 个缓冲器范围内不执行内部仲裁。
3. 不得在 RCAN0 和 RCAN1 设定相同的发送信息 ID。执行 CAN 总线上的仲裁后，变为从 2 个通道发送信息。

第 20 章 A/D 转换器 (ADC)

本 LSI 内置逐次逼近方式的 10 位 A/D 转换器，最多可选择 8 个通道的模拟输入。

20.1 特点

- 分辨率：10 位
- 输入通道：8 个通道
- 最小转换时间：每通道 3.9 μ s（P ϕ 时钟以 33MHz 运行时）
- 绝对精度： ± 4 LSB
- 运行模式：3 种

单通道模式：1 个通道的 A/D 转换

多通道模式：1~4 个通道或 1~8 个通道的 A/D 转换

扫描模式：1~4 个通道或 1~8 个通道的连续 A/D 转换

- 数据寄存器：8 个

各通道对应的 16 位数据寄存器保存转换结果

- 带采样&保持功能
- A/D 转换开始方法：3 种

软件

通过多功能定时器脉冲单元 2（MTU2）的触发开始转换

外部触发信号

- 中断源
- A/D 转换结束时，可产生 A/D 转换结束中断（ADI）请求
- 可设定模块待机模式

A/D 转换器框图如图 20.1 所示：

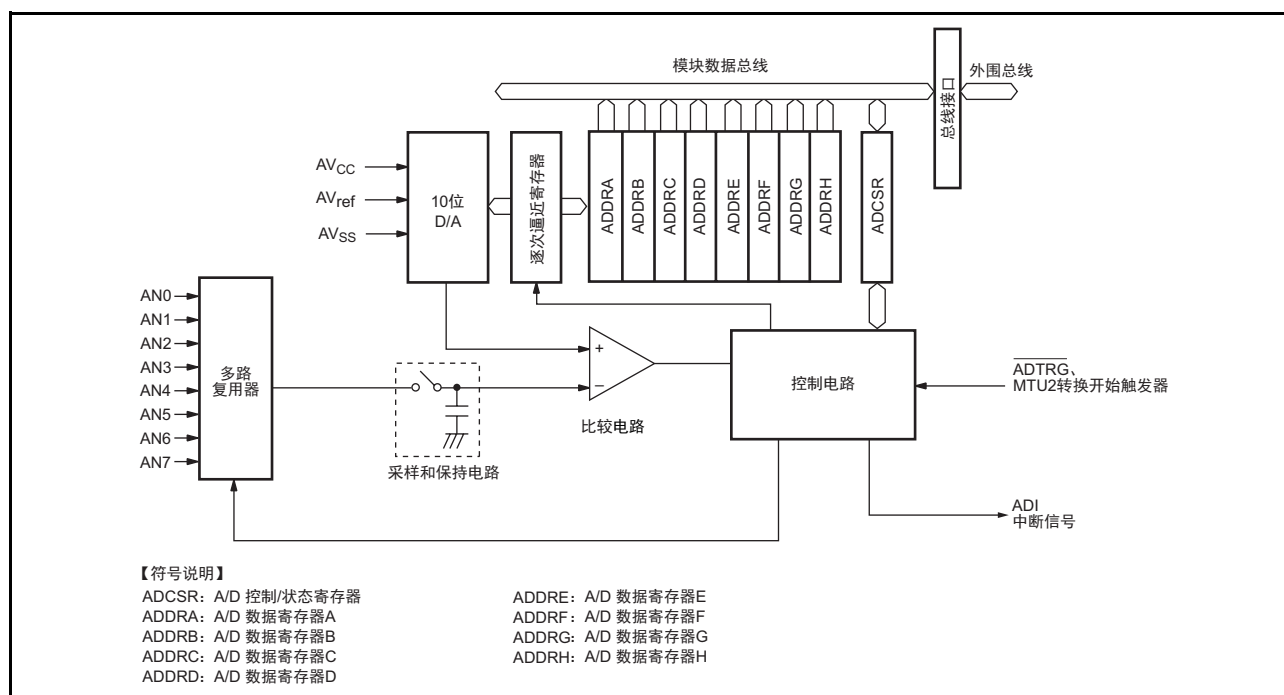


图 20.1 A/D 转换器框图

20.2 输入 / 输出引脚

A/D 转换器使用的输入引脚如表 20.1 所示:

表 20.1 引脚结构

名称	引脚名称	输入 / 输出	功 能
模拟电源引脚	AVcc	输入	模拟部分的电源引脚
模拟接地引脚	AVss	输入	模拟部分的接地引脚及 A/D 转换的基准接地
模拟基准电压引脚	AVref	输入	A/D 转换器的基准电压引脚
模拟输入引脚 0	AN0	输入	模拟输入
模拟输入引脚 1	AN1	输入	
模拟输入引脚 2	AN2	输入	
模拟输入引脚 3	AN3	输入	
模拟输入引脚 4	AN4	输入	
模拟输入引脚 5	AN5	输入	
模拟输入引脚 6	AN6	输入	
模拟输入引脚 7	AN7	输入	
A/D 外部触发输入引脚	$\overline{\text{ADTRG}}$	输入	输入为了开始 A/D 转换的外部触发

20.3 寄存器说明

A/D 转换器有以下寄存器。

表 20.2 寄存器结构

寄存器名称	简称	R/W	初始值	地址	存取长度
A/D 数据寄存器 A	ADDRA	R	H'0000	H'FFFE5800	16
A/D 数据寄存器 B	ADDRB	R	H'0000	H'FFFE5802	16
A/D 数据寄存器 C	ADDRC	R	H'0000	H'FFFE5804	16
A/D 数据寄存器 D	ADDRD	R	H'0000	H'FFFE5806	16
A/D 数据寄存器 E	ADDRE	R	H'0000	H'FFFE5808	16
A/D 数据寄存器 F	ADDRF	R	H'0000	H'FFFE580A	16
A/D 数据寄存器 G	ADDRG	R	H'0000	H'FFFE580C	16
A/D 数据寄存器 H	ADDRH	R	H'0000	H'FFFE580E	16
A/D 控制 / 状态寄存器	ADCSR	R/W	H'0040	H'FFFE5820	16

20.3.1 A/D 数据寄存器 A ~ H（ADDRA ~ ADDRH）

ADDR 为保存 A/D 转换结果的 16 位只读寄存器，ADDRA ~ ADDRH 共有 8 个。

A/D 转换后的数据为 10 位，传送并保存到已转换通道对应的 ADDR 的 bit15 ~ 6，bit5 ~ 0 的读取值总是为 0。

禁止以 8 位为单位存取 ADDR，必须以 16 位为单位存取。

模拟输入通道与 ADDR 的对应关系如表 20.3 所示。

位：	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
											-	-	-	-	-	-
初始值：	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W：	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名称	初始值	R/W	说 明
15 ~ 6		均为 0	R	位数据（10 位）
5 ~ 0	—	均为 0	R	保留位 读取值、写入值总是为 0。

表 20.3 模拟输入通道与 ADDR 的对应关系

模拟输入通道	保存转换结果的 A/D 数据寄存器
AN0	ADDRA
AN1	ADDRB
AN2	ADDRC
AN3	ADDRD
AN4	ADDRE
AN5	ADDRF
AN6	ADDRG
AN7	ADDRH

20.3.2 A/D 控制 / 状态寄存器 (ADCSR)

ADCSR 为 16 位可读取 / 写入的寄存器。可选择运行模式、控制 A/D 转换，以及选择允许 / 禁止因外部触发输入引起的 A/D 转换开始。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADF	ADIE	ADST	-	TRGS[3:0]				CKS[1:0]		MDS[2:0]		CH[2:0]			
初始值:	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
R/W:	R/(W)*	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

【注】* 为了清除标志，仅可在读取1后写入0。

位	位名称	初始值	R/W	说 明
15	ADF	0	R/(W)*	A/D 结束标志 表示 A/D 转换结束的状态标志。 [清除条件] • 在 ADF=1 的状态读取 ADF 标志后，向 ADF 标志写入 0 时 • 由 ADI 中断启动 DMAC，并读取 ADDR 时 [置位条件] • 在单通道模式下，A/D 转换结束时 • 在多通道模式下，所选通道的 A/D 转换全部结束时 • 在扫描模式下，所选通道的 A/D 转换全部结束时
14	ADIE	0	R/W	A/D 中断允许 选择允许 / 禁止因 A/D 转换结束引起的中断 (ADI) 请求。必须在转换停止时设定 ADIE 位。 0: 禁止因 A/D 转换结束引起的中断 (ADI) 请求 1: 允许因 A/D 转换结束引起的中断 (ADI) 请求
13	ADST	0	R/W	A/D 转换开始 选择 A/D 转换的开始或停止。在 A/D 转换过程中保持为 1。 0: 停止 A/D 转换 1: 单通道模式时: 开始 A/D 转换。指定通道的 A/D 转换结束后自动清除。 多通道模式时: 开始 A/D 转换。指定的所有通道执行完一轮 A/D 转换后自动清 0。 扫描模式时: 开始 A/D 转换。通过向软件、上电复位、深度待机模式、软件待机模式或模块待机模式转移来清 0 之前，持续 A/D 转换。
12	—	0	R	保留位 读取值、写入值总是为 0。
11 ~ 8	TRGS[3:0]	0000	R/W	定时器触发选择 允许 / 禁止通过触发信号开始 A/D 转换。 0000: 禁止通过外部触发开始 A/D 转换 0001: 通过 MTU2 的转换触发 TRGAN 开始 A/D 转换 0010: 通过 MTU2 的转换触发 TRG0N 开始 A/D 转换 0011: 通过 MTU2 的转换触发 TRG4AN 开始 A/D 转换 0100: 通过 MTU2 的转换触发 TRG4BN 开始 A/D 转换 1001: 通过 ADTRG 开始 A/D 转换 上述以外: 禁止设定

位	位名称	初始值	R/W	说 明																																				
7、 6	CKS[1:0]	01	R/W	时钟选择 必须在转换停止时（ADST=0），设定 A/D 转换时间。 00: 转换时间 =138 个状态（最大值） 以 Pφ/4 运行 01: 转换时间 =274 个状态（最大值） 以 Pφ/8 运行 10: 转换时间 =546 个状态（最大值） 以 Pφ/16 运行 11: 禁止设定																																				
5 ~ 3	MDS[2:0]	000	R/W	多通道扫描模式 选择 A/D 转换的运行模式。 0xx: 单通道模式 100: 多通道模式。 1 ~ 4 个通道的 A/D 转换 101: 多通道模式。 1 ~ 8 个通道的 A/D 转换 110: 扫描模式。 1 ~ 4 个通道的 A/D 转换 111: 扫描模式。 1 ~ 8 个通道的 A/D 转换																																				
2 ~ 0	CH[2:0]	000	R/W	通道选择 与 ADCSR 的 MDS 位共同选择模拟输入。 <table> <tr> <th></th> <th>MDS2=1、 MDS2=0 时</th> <th>MDS2=1、 MDS0=0 时</th> <th>MDS2=1、 MDS0=1 时</th> </tr> <tr> <td>000: AN0</td> <td>000: AN0</td> <td>000: AN0</td> <td>000: AN0</td> </tr> <tr> <td>001: AN1</td> <td>001: AN0</td> <td>001: AN0、 AN1</td> <td>001: AN0、 AN1</td> </tr> <tr> <td>010: AN2</td> <td>010: AN0 ~ AN2</td> <td>010: AN0 ~ AN2</td> <td>010: AN0 ~ AN2</td> </tr> <tr> <td>011: AN3</td> <td>011: AN0 ~ AN3</td> <td>011: AN0 ~ AN3</td> <td>011: AN0 ~ AN4</td> </tr> <tr> <td>100: AN4</td> <td>100: AN4</td> <td>100: AN4</td> <td>100: AN0 ~ AN4</td> </tr> <tr> <td>101: AN5</td> <td>101: AN4、 AN5</td> <td>101: AN4、 AN5</td> <td>101: AN0 ~ AN5</td> </tr> <tr> <td>110: AN6</td> <td>110: AN4 ~ AN6</td> <td>110: AN4 ~ AN6</td> <td>110: AN0 ~ AN6</td> </tr> <tr> <td>111: AN7</td> <td>111: AN4 ~ AN7</td> <td>111: AN4 ~ AN7</td> <td>111: AN0 ~ AN7</td> </tr> </table>		MDS2=1、 MDS2=0 时	MDS2=1、 MDS0=0 时	MDS2=1、 MDS0=1 时	000: AN0	000: AN0	000: AN0	000: AN0	001: AN1	001: AN0	001: AN0、 AN1	001: AN0、 AN1	010: AN2	010: AN0 ~ AN2	010: AN0 ~ AN2	010: AN0 ~ AN2	011: AN3	011: AN0 ~ AN3	011: AN0 ~ AN3	011: AN0 ~ AN4	100: AN4	100: AN4	100: AN4	100: AN0 ~ AN4	101: AN5	101: AN4、 AN5	101: AN4、 AN5	101: AN0 ~ AN5	110: AN6	110: AN4 ~ AN6	110: AN4 ~ AN6	110: AN0 ~ AN6	111: AN7	111: AN4 ~ AN7	111: AN4 ~ AN7	111: AN0 ~ AN7
	MDS2=1、 MDS2=0 时	MDS2=1、 MDS0=0 时	MDS2=1、 MDS0=1 时																																					
000: AN0	000: AN0	000: AN0	000: AN0																																					
001: AN1	001: AN0	001: AN0、 AN1	001: AN0、 AN1																																					
010: AN2	010: AN0 ~ AN2	010: AN0 ~ AN2	010: AN0 ~ AN2																																					
011: AN3	011: AN0 ~ AN3	011: AN0 ~ AN3	011: AN0 ~ AN4																																					
100: AN4	100: AN4	100: AN4	100: AN0 ~ AN4																																					
101: AN5	101: AN4、 AN5	101: AN4、 AN5	101: AN0 ~ AN5																																					
110: AN6	110: AN4 ~ AN6	110: AN4 ~ AN6	110: AN0 ~ AN6																																					
111: AN7	111: AN4 ~ AN7	111: AN4 ~ AN7	111: AN0 ~ AN7																																					

【符号说明】x: Don't care

【注】 * 为了清除标志, 仅可在读取 1 后写入 0。

20.4 运行说明

A/D 转换器为逐次逼近方式，其分辨率为 10 位。运行模式有单通道模式、多通道模式及扫描模式。为了防止误操作，必须在 ADCSR 的 ADST 位为 0 时，转换运行模式及模拟输入通道。ADST 置位的同时可更改运行模式及模拟输入通道。

20.4.1 单通道模式

仅 1 个通道 A/D 转换时，选择单通道模式。

在单通道模式，指定的一个通道的模拟输入如下执行 1 次 A/D 转换：

1. 由软件、MTU2 或外部触发输入将 ADCSR 的 ADST 位置 1 后，所选通道开始 A/D 转换。
2. A/D 转换结束后，将 A/D 转换结果传送至相应通道的 A/D 数据寄存器。
3. A/D 转换结束后，ADCSR 的 ADF 位置 1。此时，如果 ADIE 位置 1，则产生 ADI 中断请求。
4. ADST 位在 A/D 转换过程中保持 1，转换结束时自动清 0，A/D 转换器为待机状态。

在 A/D 转换过程中为了避免误操作，必须在 ADST 位清 0、停止 A/D 转换的状态下，切换模式或模拟输入通道。更新后，ADST 位置 1(模式及通道的转换与 ADST 置位可同时进行)，再次开始 A/D 转换。

在单通道模式选择通道 1 (AN1) 时的运行例如下所示。此时的运行时序如图 20.2 所示 (本运行例中指定的位属于 ADCSR 寄存器)。

1. 设定运行模式为单通道模式，输入通道为 AN1 (CH[2:0]=001) 并允许 A/D 中断请求 (ADIE=1)，开始 A/D 转换 (ADST=1)。
2. A/D 转换结束后，将 A/D 转换结果传送至 ADDR1。同时，ADF=1、ADST=0，A/D 转换器为转换等待。
3. ADF=1、ADIE=1，产生 ADI 中断请求。
4. 开始 A/D 中断处理程序。
5. 读取 ADF=1 后，向 ADF 写入 0。
6. 读取并处理 A/D 转换结果 (ADDR1)。
7. 结束 A/D 中断处理程序之后，ADST 位置 1 后，开始 A/D 转换，执行步骤 2.~7.。

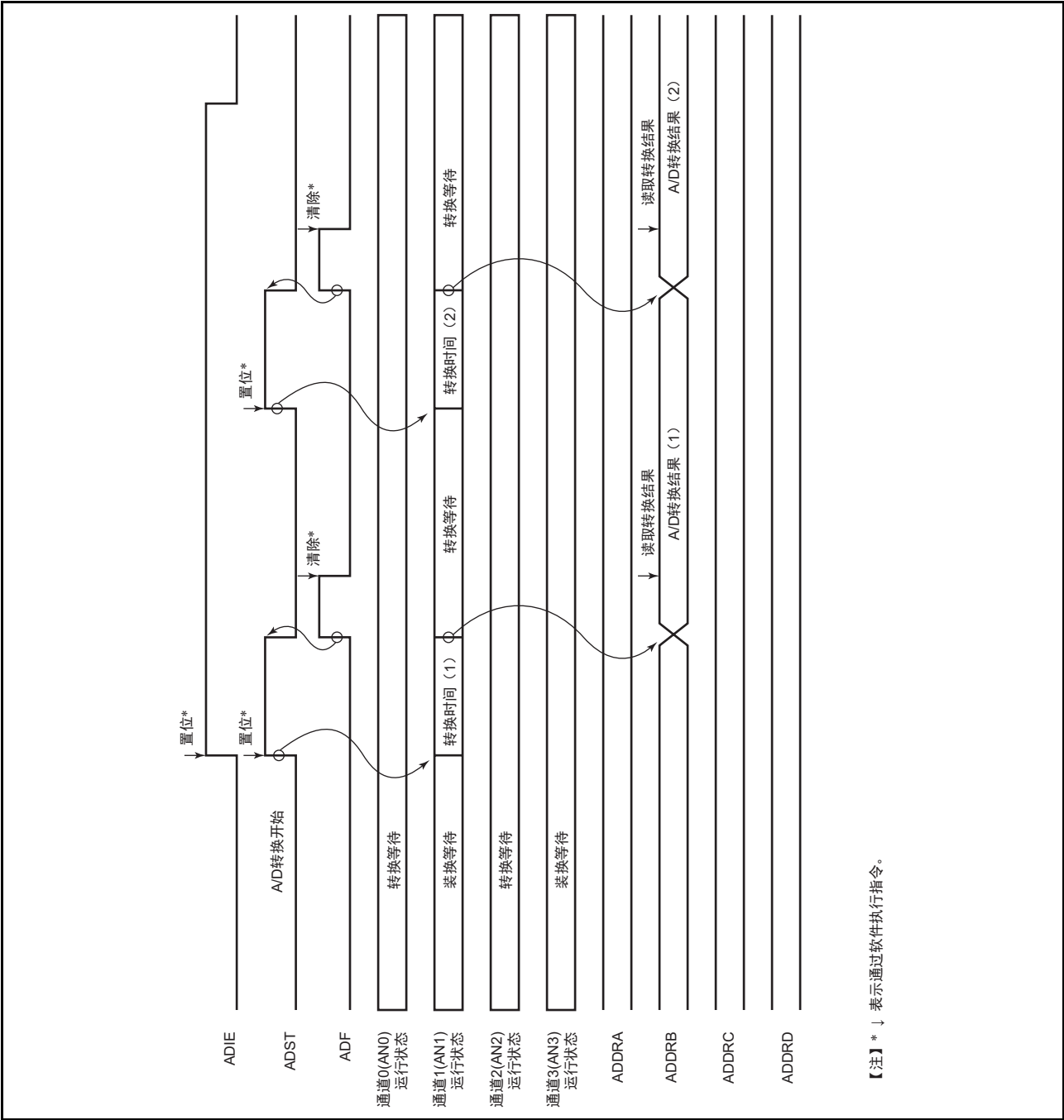


图 20.2 A/D 转换器运行例 (选择单通道模式、通道 1 时)

20.4.2 多通道模式

在多通道模式下，分别依次转换 1 次多个通道（包含 1 个）的模拟输入。

在多通道模式下，将指定的最多 8 个通道的模拟输入如下执行 1 次 A/D 转换。

1. 由软件、MTU2 或外部触发输入将 ADCSR 的 ADST 位置 1 时，则按照模拟输入通道编号从小到大的顺序（例如 AN0、AN1……AN3）执行 A/D 转换。
2. 各通道的所有 A/D 转换结束时，A/D 转换结果依次传送至相应通道的 A/D 数据寄存器。
3. 所选通道的 A/D 转换结束时，ADCSR 的 ADF 位置 1。此时，如果 ADIE 位置 1，则产生 ADI 中断请求。
4. ADST 位在 A/D 转换过程中保持 1，转换结束时自动清 0，A/D 转换器为待机状态下。A/D 切换过程中如果 ADST 位清 0，则停止转换，A/D 转换器为待机状态。读取 ADF=1 后，再对 ADF 位写入 0 即可清除。

在指定的所有通道执行一轮 A/D 转换，其结果传送并保存至各通道对应的 ADDR。

在 A/D 转换过程中，为了防止误操作，必须在 ADST 位清 0、停止 A/D 转换的状态，转换模式及模拟输入通道。更改后，ADST 位置 1 时（模式及通道的更改与 ADST 位的置位可同时进行），选择第 1 个通道，再次开始 A/D 转换。

在多通道模式下选择 3 个通道（AN0 ~ AN2）执行 A/D 转换的运行例如下所示，此时的运行时序如图 20.3 所示。

1. 运行模式设定为多通道模式（MDS[2]=1、MDS[1]=0），模拟输入通道设定为 AN0 ~ AN2（CH[2:0]=010）后，开始 A/D 转换（ADST=1）。
2. 开始第 1 个通道（AN0）的 A/D 转换。转换结束时，其结果传送至 ADDR0。
3. 其次自动选择第 2 个通道（AN1），开始 A/D 转换。
4. 同样地执行第 3 个通道（AN2）的转换。
5. 如果所选通道（AN0 ~ AN2）的转换均已结束，则在 ADF=1，ADST 位设定为 0 下，结束转换。
6. 此时如果 ADIE 位为 1，则 A/D 转换结束后产生 ADI 中断。

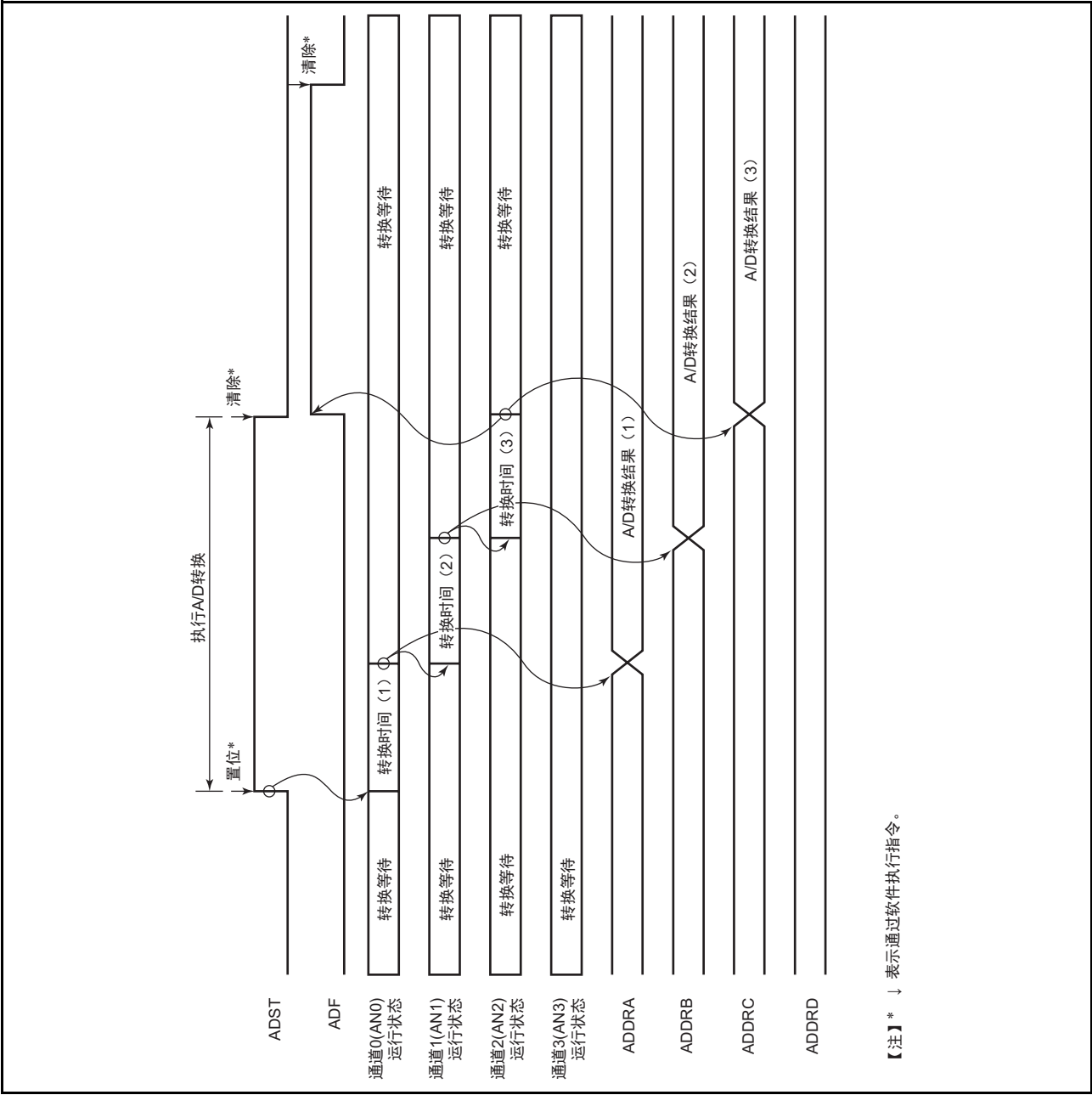


图 20.3 A/D 转换器的运行例（选择多通道模式、AN0 ~ AN2 等 3 个通道时）

20.4.3 扫描模式

扫描模式适用于总是监控多个通道（包含 1 个通道）的模拟输入的系统。在扫描模式下，将指定的最多 8 个通道的模拟输入如下依次连续进行 A/D 转换。

1. 由软件、MTU2 或外部触发输入将 ADCSR 的 ADST 位置 1 时，按照模拟输入通道编号从小到大的顺序（例如 AN0、AN1……AN3）进行 A/D 转换。
2. 各通道的 A/D 转换结束时，A/D 转换结果被依次传送至相应通道的 A/D 数据寄存器。
3. 所选通道的 A/D 转换结束时，ADCSR 的 ADF 位置 1。此时，如果 ADIE 位置 1，则产生 ADI 中断请求。A/D 转换器再次按照通道编号从小到大的顺序开始 A/D 转换。
4. ADST 位不自动清除，置 1 期间重复执行步骤 2.~3.。如果 ADST 位清 0，则停止 A/D 转换，A/D 转换器为待机状态。读取 ADF=1 后，对 ADF 位写入 0 即可清除。

在 A/D 转换过程中，为了防止误操作，必须在 ADST 位清 0、停止 A/D 转换的状态下，切换模式及模拟输入通道。更改后，ADST 位置 1 时（模式及通道的更改与 ADST 位的置位可同时进行），选择第 1 个通道，再次开始 A/D 转换。

在扫描模式选择 3 个通道（AN0 ~ AN2）执行 A/D 转换的运行例如下所示，此时的运行时序如图 20.4 所示。

1. 设定运行模式设定为扫描模式（MDS[2]=1、MDS[1]=1），模拟输入通道设定为 AN0 ~ AN2（CH[2:0]=010）后，开始 A/D 转换（ADST=1）。
2. 开始第 1 个通道（AN0）的 A/D 转换。转换结束时，其结果传送至 ADDRA。
3. 其次自动选择第 2 个通道（AN1），开始 A/D 转换。
4. 同样地，执行第 3 个通道（AN2）的转换。
5. 如果所选通道（AN0 ~ AN2）的转换均已结束，则 ADF=1，再次选择第 1 个通道（AN0），连续执行转换。此时如果 ADIE 位为 1，则 A/D 转换（第 3 个通道的转换）结束后产生 ADI 中断。
6. ADST 位不自动清除，置 1 期间重复步骤 2.~4.，重复期间保持 ADF=1。ADST 位清 0 时，停止 A/D 转换。读取 ADF=1 后，对 ADF 位写入 0 即可清除。

重复步骤 2. ~ 4. 期间 ADF=1、ADIE=1 时，通常产生 ADI 中断。如果第 3 通道的转换结束时需产生中断，则在中断产生后，必须将 ADF 位清 0。

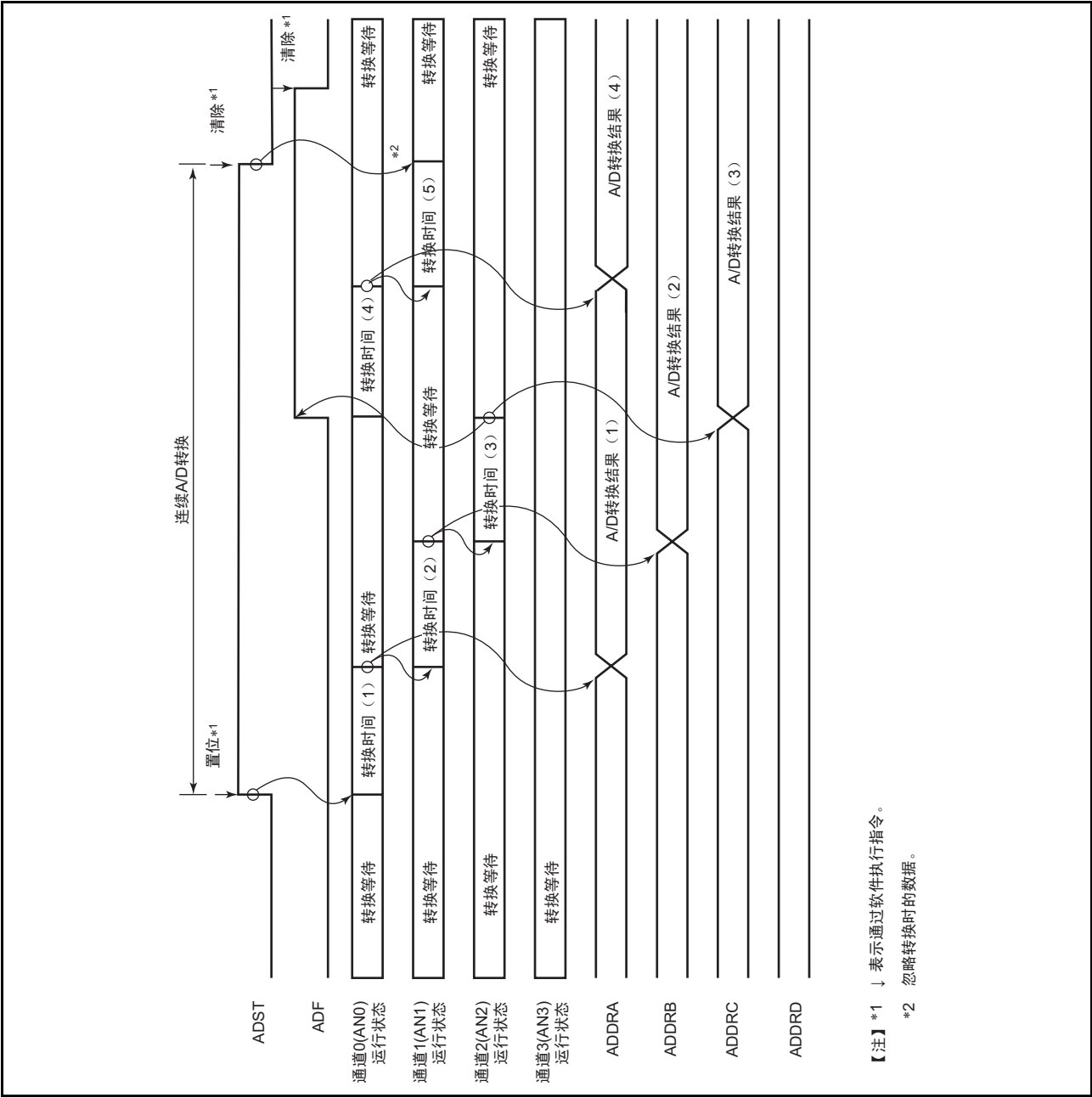


图 20.4 A/D 转换器的运行例 (选择扫描模式、AN0 ~ AN2 等 3 个通道时)

20.4.4 由外部触发、MTU2 启动 A/D 转换器

可根据外部触发、MTU2 的 A/D 转换请求，独立启动 A/D 转换器。由外部触发、MTU2 启动 A/D 转换器时，设定 A/D 触发允许位 (TRGS3 ~ TRGS0)。在此状态下，如果产生外部触发、MTU2 的 A/D 转换请求，则将 ADST 位置 1，由此开始 A/D 转换。由 ADCSR 的 CH2 ~ CH0 位决定转换的通道。ADST 位置 1 后到开始 A/D 转换前的时序，与软件对 ADST 位写入 1 的情况相同。

20.4.5 输入采样和 A/D 转换时间

A/D 转换器内置采样 & 保持电路。ADCSR 的 ADST 位置 1 后，经过 A/D 转换开始延迟时间 (t_D) 后，A/D 转换器执行输入采样，之后开始转换。A/D 转换的时序如图 20.5、A/D 转换时间如表 20.4 所示。

A/D 转换时间 (t_{CONV}) 包含 t_D 和输入采样时间 (t_{SPL})，如图 20.5 所示。 t_D 由写入 ADCSR 的时序决定，不是固定值。因此，转换时间在表 20.4 所示范围内变化。

表 20.4 所示值为多通道模式和扫描模式的第 1 次转换时间，表 20.5 所示值为第 2 次及以后的转换时间。

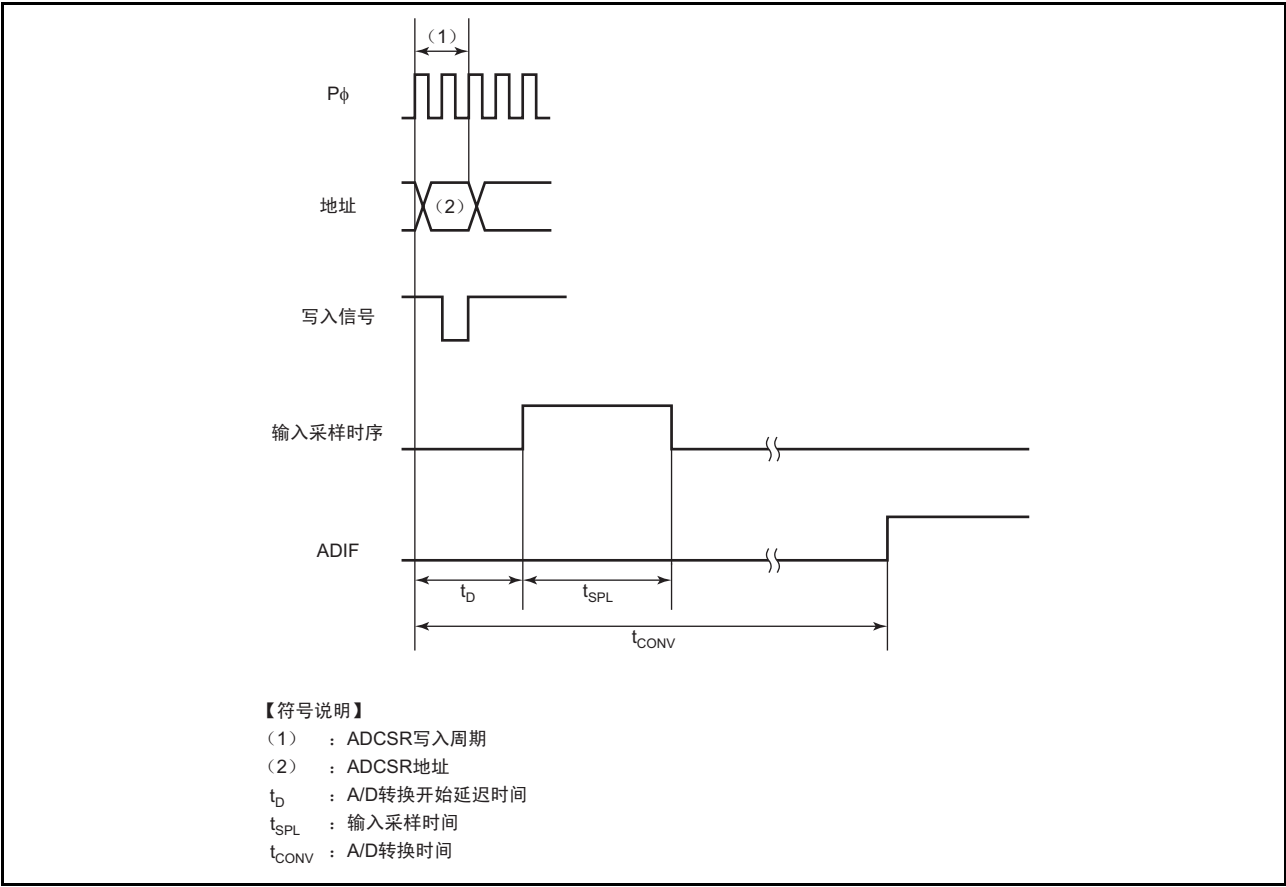


图 20.5 A/D 转换时序

表 20.4 A/D 转换时间（单通道模式）

项目	符号	CKS1=0						CKS1=1		
		CKS0=0			CKS0=1			CKS0=0		
		Min.	Typ.	Max.	Min.	Typ.	Max.	Min.	Typ.	Max.
A/D 转换开始延迟时间	tD	11	—	14	19	—	26	35	—	50
输入采样时间	tSPL	—	33	—	—	65	—	—	129	—
A/D 转换时间	tCONV	135	—	138	267	—	274	531	—	546

【注】 表中数值的单位为状态。

表 20.5 A/D 转换时间（多通道模式 / 扫描模式）

CKS1	CKS0	转换时间（状态）
0	0	128（固定）
	1	256（固定）
1	0	512（固定）

【注】 表中数值的单位为状态。

20.4.6 外部触发输入时序

可由外部触发输入开始 A/D 转换。外部触发输入是指在 ADCSR 的 TRGS3 ~ TRGS0 位设置为 B'1001 时，从 ADTRG 引脚输入外部触发。在 ADTRG 的下降沿，ADCSR 的 ADST 位置 1，开始 A/D 转换。其他运行与单通道模式 / 多通道模式 / 扫描模式无关，均与通过软件将 ADST 位置 1 时相同，外部出发输入时序如图 20.6 所示。

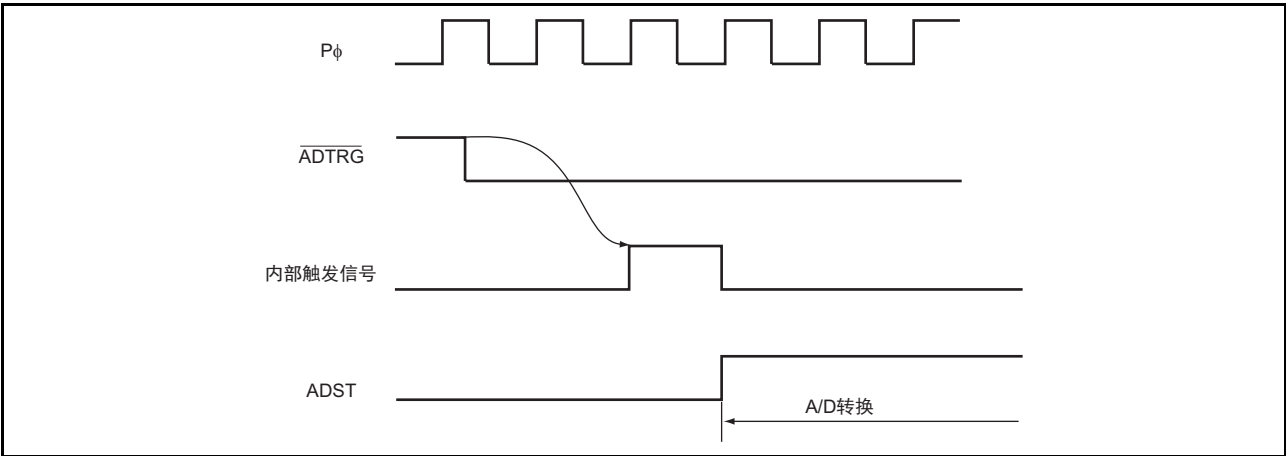


图 20.6 外部触发输入时序

20.5 中断源和 DMAC 传送请求

A/D 转换结束时，A/D 转换器产生 A/D 转换中断 (ADI)。A/D 转换结束后，ADCSR 的 ADF 位置 1，此时如果 ADIE 位置 1，则产生 ADI 中断请求。另外，根据直接存储器存取控制器 (DMAC) 的设定，ADI 中断可启动 DMAC，此时不向 CPU 产生中断请求。未设定启动 DMAC 时，则向 CPU 产生中断请求。使用 DMAC 读取因 ADI 中断转换的数据时，可实现连续转换且无软件负荷。

在单通道模式，必须设定由 ADI 中断引起的 DMA 传送为 1 次。对于扫描模式或多通道模式的多通道 A/D 转换，设定 DMA 传送次数为 1 次时，仅传送一个通道的数据后即结束 DMA 传送。由 DMAC 传送所有转换数据时，必须将传送源地址设定为保存 A/D 转换数据的 ADDR，将传送次数设定为转换通道数，并将 DMA 通道控制寄存器 (CHCR) 的 TC 位置 1。

如果 ADI 启动 DMAC，由 DMAC 传送数据时自动清除 ADCSR 的 ADF 位。

表 20.6 中断源和 DMAC 传送请求的关系

名称	中断源	中断标志	DMAC 的启动
ADI	A/D 转换结束	ADCSR 的 ADF	可

20.6 A/D 转换精度的定义

A/D 转换器比较从模拟输入通道输入的模拟值与模拟基准电压，同时转换为 10 位的数字值。此时，A/D 转换的绝对精度，即输入模拟值与输出数字值的偏差包含以下误差：

1. 偏移误差
2. 满刻度误差
3. 量化误差
4. 非线性误差

根据图 20.7，说明上述 1. ~ 4. 的误差。为了使图简单易懂，将 10 位 A/D 转换器简化为 3 位 A/D 转换器。偏移误差是指数字输出值从最小值 (0 电压) B'0000000000 (图中为 000) 变为 B'0000000001 (图中为 001) 时，实际 A/D 转换特性与理想 A/D 转换特性的偏差 (图 20.7 (1))。满刻度误差是指数字输出值从 B'1111111110 (图中为 110) 变为最大值 (满刻度电压) B'1111111111 (图中为 111) 变化时，实际 A/D 转换特性与理想 A/D 转换特性的偏差 (图 20.7 (2))。量化误差是指 A/D 转换器固有的偏差，表示为 1/2LSB (图 20.7 (3))。非线性误差是指从 0 电压到满刻度电压之间，实际 A/D 转换特性与理想 A/D 转换特性的偏差 (图 20.7 (4))，不包含偏移误差、满刻度误差及量化误差。

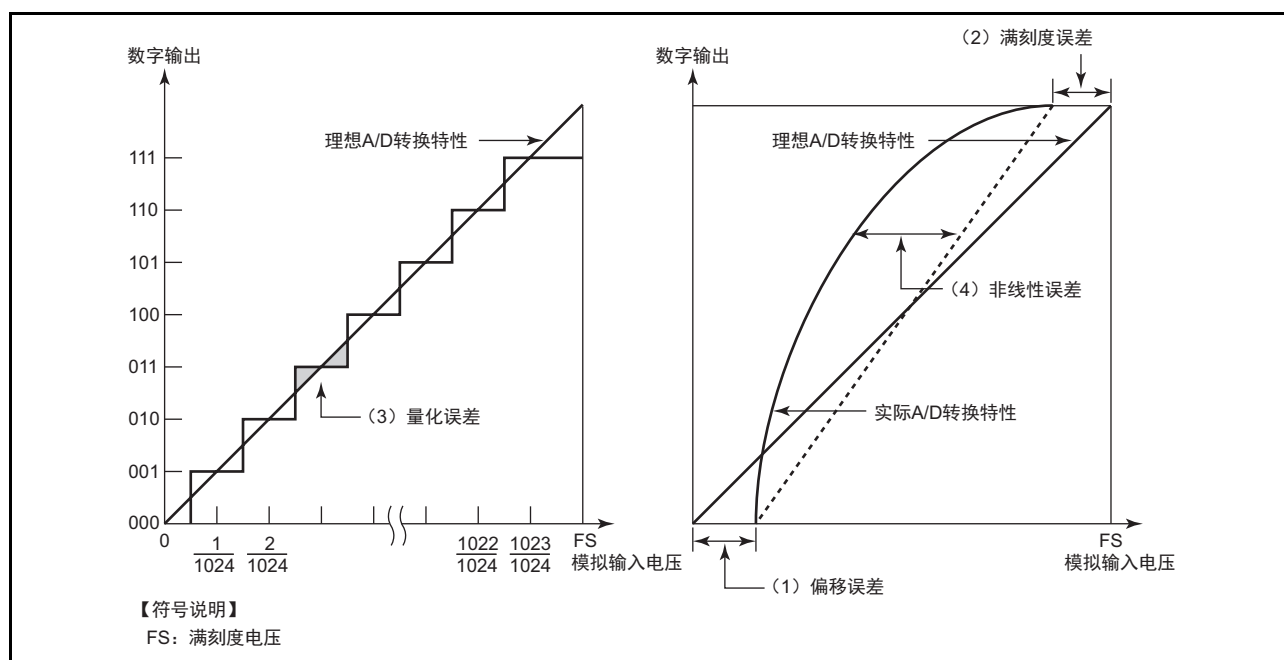


图 20.7 A/D 转换精度的定义

20.7 使用时的注意事项

使用 A/D 转换器时，必须注意以下事项：

20.7.1 模块待机模式的设定

通过待机控制寄存器，可设定允许 / 禁止 A/D 转换器的运行。为初始值时，A/D 转换器停止运行。通过解除模块待机模式可存取寄存器。详情参阅“第 28 章 低功耗模式”。

20.7.2 模拟电压的设定

如果超出以下所示电压的设定范围使用 LSI 时，可能影响其可靠性。

1. 模拟输入电压的范围

在 A/D 转换过程中，必须将外加给模拟输入引脚 AN_n 的电压设定在 $AV_{SS} \leq AN_n \leq AV_{CC}$ 的范围内。
($n=0 \sim 7$)。

2. AV_{CC} 、 AV_{SS} 输入电压

AV_{CC} 、 AV_{SS} 的输入电压必须设定为 $PV_{CC} - 0.3V \leq AV_{CC} \leq PV_{CC}$ 、 $AV_{SS} = PV_{SS}$ 。并且，不使用 A/D 转换器或 D/A 转换器时，或即使在软件待机模式， AV_{CC} 、 AV_{SS} 引脚也不得设置为开路。不使用时，必须将 AV_{CC} 连接至电源 (PV_{CC})， AV_{SS} 连接至接地 (PV_{SS})。

3. AV_{ref} 的设定范围

AV_{ref} 引脚的基准电压范围必须设定为 $3.0V \leq AV_{ref} \leq AV_{CC}$ 。

20.7.3 电路板设计的注意事项

设计电路板时，必须尽可能将数字电路和模拟电路分开。布线时应极力避免使数字电路的信号线和模拟电路的信号线交叉或靠近。否则，会因感应等引起模拟电路的误运行，或影响 A/D 转换值。

模拟输入信号 ($AN0 \sim AN7$)、模拟基准电压 (AV_{ref})、模拟电源 (AV_{cc}) 必须通过模拟接地 (AV_{ss}) 与数字电路分离。并且，模拟接地 (AV_{ss}) 必须单点连接至电路板上稳定的数字接地 (PV_{ss})。

20.7.4 模拟输入引脚的处理

为了防止因过大电涌等异常电压对模拟输入引脚（AN0～AN7）的破坏，必须如图 20.8 所示的保护电路进行连接。该图的电路兼有 CR 过滤器功能（可抑制噪声引起的误差）。图中的电路为设计例，必须根据实际使用条件决定电路常数。

模拟输入引脚的等效电路如图 20.9 所示，模拟输入引脚的规格如表 20.7 所示。

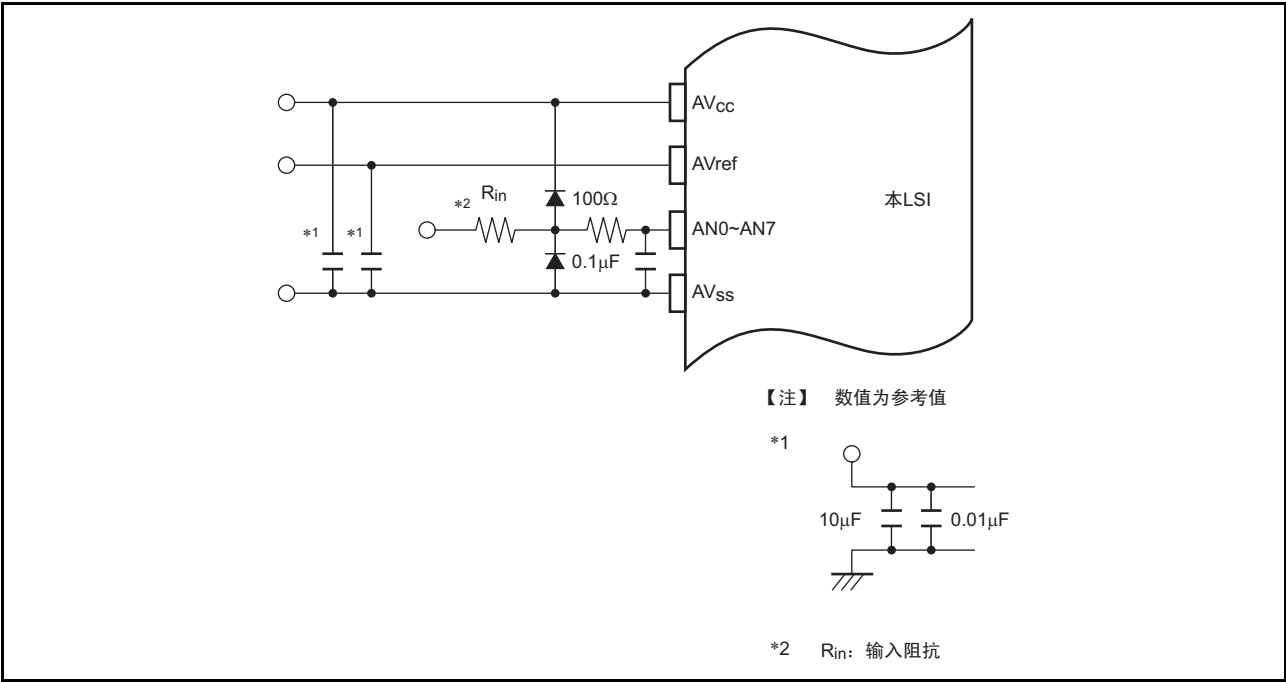


图 20.8 模拟输入引脚的保护电路例

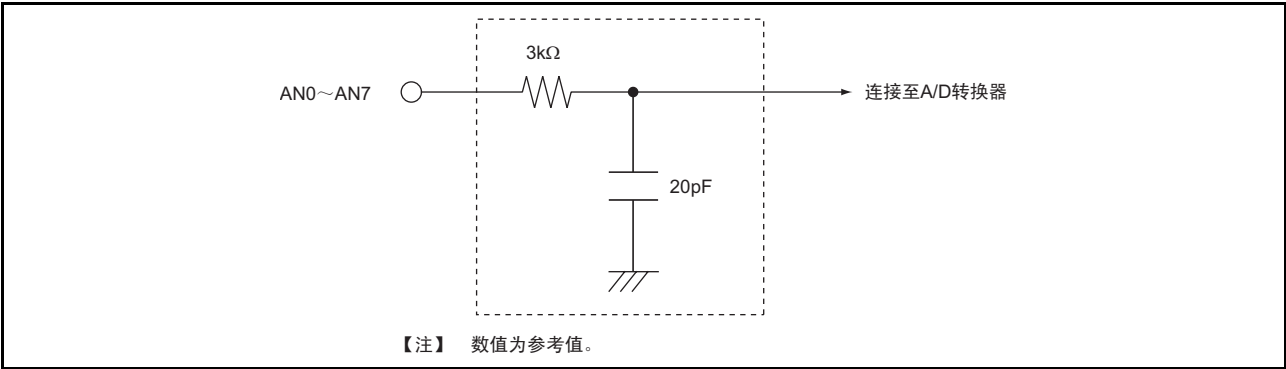


图 20.9 模拟输入引脚的等效电路

表 20.7 模拟输入引脚的规格

项目	Min.	Max.	单位
模拟输入电容	—	20	pF
容许信号源阻抗	—	5	kΩ

20.7.5 容许信号源阻抗

对于信号源阻抗不超过 $5\text{k}\Omega$ 的输入信号，本 LSI 的设计可保证模拟输入的转换精度。这是为了在采样时间内对 A/D 转换器的采样 & 保持电路的输入电容进行充电而制定的规格。传感器的输出阻抗超过 $5\text{k}\Omega$ 时，产生充电不足，无法保证 A/D 转换精度。在单通道模式转换且外置大电容时，输入负载实际只是 $3\text{k}\Omega$ 的内部输入电阻，因此信号源阻抗可忽略不计。但是，由于形成低通滤波器，可能无法跟踪大微分系数的模拟信号（例如大于等于 $5\text{mV}/\mu\text{s}$ ）（图 20.10）。转换高速模拟信号或在扫描模式转换时，必须加载低阻抗缓冲器。

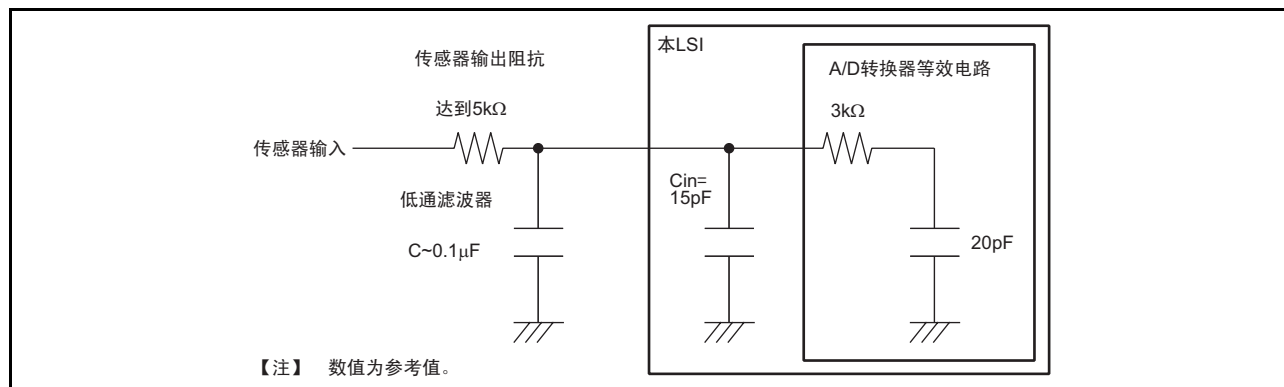


图 20.10 模拟输入电路的例子

20.7.6 对绝对精度的影响

附加电容会导致接受与 GND 的耦合。如果 GND 有噪声，可能降低绝对精度，因此必须确保 AV_{SS} 等连接至稳定的 GND。

另外，必须避免滤波器电路在安装电路板上干扰数字信号或充当天线。

20.7.7 深度待机模式时的 A/D 转换

转移至深度待机模式时，必须将 ADST 位清 0、禁止 A/D 转换。允许 A/D 转换时，如果本 LSI 为深度待机模式，则不保证 A/D 的引脚状态。

20.7.8 使用扫描模式及多通道模式时的注意事项

扫描模式及多通道模式停止后，立即更改为单通道模式并开始转换时，可在单通道模式显示错误的转换结果。

连续执行单通道模式转换时，设定 ADST=0 后，必须经过 1 个通道的 A/D 转换时间后再启动（ADST=1）（1 个通道的转换时间根据 ADC 分频寄存器的设定而不同）。

第 21 章 D/A 转换器 (DAC)

21.1 特点

- 分辨率：8 位
- 输出通道：2 个通道
- 最小转换时间：10μs（负载电容 20pF 时）
- 输出电压：0V ~ AVref
- 软件待机模式时的 D/A 输出保持功能
- 可设定模块待机模式

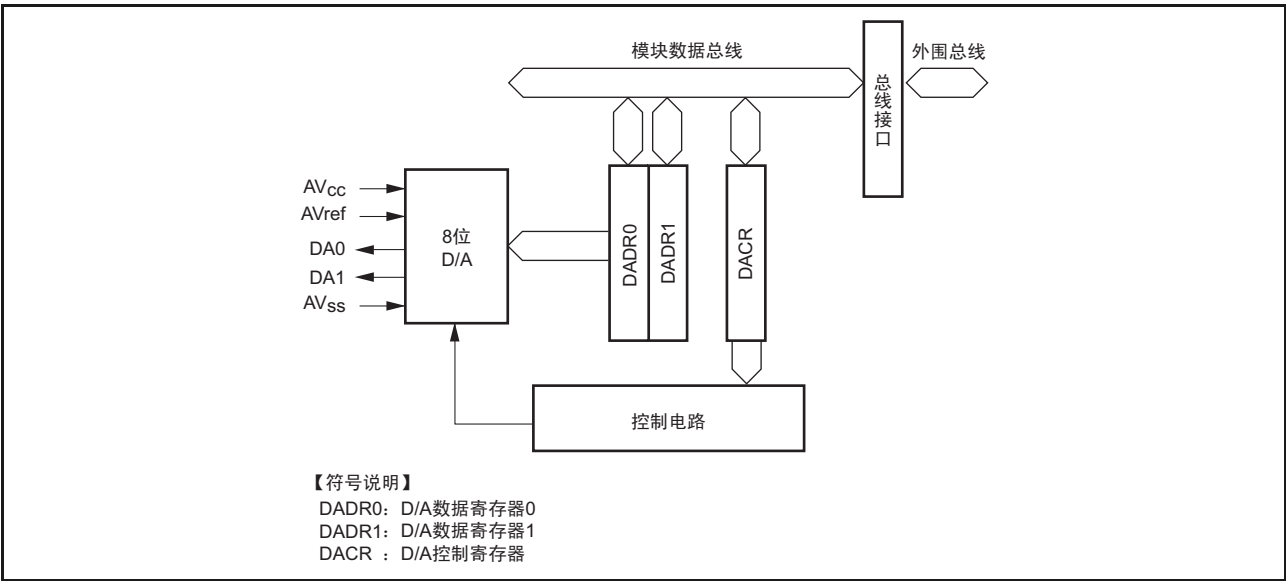


图 21.1 D/A 转换器框图

21.2 输入 / 输出引脚

D/A 转换器使用的输入 / 输出引脚，如表 21.1 所示。

表 21.1 引脚结构

名称	引脚名称	输入 / 输出	功能
模拟电源引脚	AVcc	输入	模拟部的电源引脚
模拟接地引脚	AVss	输入	模拟部的接地引脚
模拟基准电压引脚	AVref	输入	D/A 转换器的基准电压引脚
模拟输出引脚 0	DA0	输出	通道 0 的模拟输出
模拟输出引脚 1	DA1	输出	通道 1 的模拟输出

21.3 寄存器说明

D/A 转换器有以下寄存器。

表 21.2 寄存器结构

寄存器名称	简称	R/W	初始值	地址	存取长度
D/A 数据寄存器 0	DADR0	R/W	H'00	H'FFFE6800	8、16
D/A 数据寄存器 1	DADR1	R/W	H'00	H'FFFE6801	8、16
D/A 控制寄存器	DACR	R/W	H'1F	H'FFFE6802	8、16

21.3.1 D/A 数据寄存器 0、1 (DADR0、DADR1)

DADR 为 8 位可读取 / 写入的寄存器，用于保存执行 D/A 转换的数据。如果允许模拟输出，则转换 DADR 的值并输出至模拟输出引脚。

在上电复位或模块待机模式下，DADR 初始化为 H'00。

位:	7	6	5	4	3	2	1	0
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

21.3.2 D/A 控制寄存器 (DACR)

DACR 为 8 位可读取 / 写入的寄存器，控制 D/A 转换器运行。

在上电复位及模块待机模式，DACR 初始化为 H'1F。

位:	7	6	5	4	3	2	1	0
	DAOE1	DAOE0	DAE	-	-	-	-	-
初始值:	0	0	0	1	1	1	1	1
R/W:	R/W	R/W	R/W	-	-	-	-	-

位	位名称	初始值	R/W	说 明
7	DAOE1	0	R/W	D/A 输出允许 1 控制通道 1 的 D/A 转换与模拟输出。 0: 禁止通道 1 的模拟输出 (DA1) 1: 允许通道 1 的 D/A 转换及模拟输出 (DA1)
6	DAOE0	0	R/W	D/A 输出允许 0 控制通道 0 的 D/A 转换与模拟输出。 0: 禁止通道 0 的模拟输出 (DA0) 1: 允许通道 0 的 D/A 转换及模拟输出 (DA0)
5	DAE	0	R/W	D/A 允许 与 DAOE0、DAOE1 位组合，控制 D/A 转换。转换结果的输出由 DAOE0、DAOE1 位控制。参照表 21.3。 0: 单独控制通道 0 与通道 1 的 D/A 转换 1: 统一控制通道 0 与通道 1 的 D/A 转换
4 ~ 0	—	均为 1	—	保留位 读取值总是为 1。写入无效。

表 21.3 D/A 转换的控制

bit5	bit7	bit6	说 明
DAE	DAOE1	DAOE0	
0	0	0	禁止 D/A 转换
		1	允许通道 0 的 D/A 转换，禁止通道 1 的 D/A 转换
	1	0	允许通道 1 的 D/A 转换，禁止通道 0 的 D/A 转换
		1	允许通道 0、1 的 D/A 转换
1	0	0	禁止 D/A 转换
		1	允许通道 0、1 的 D/A 转换
	1	0	
		1	

21.4 运行说明

可分别单独执行 2 个通道的 D/A 转换。如果 DACR 的 DAOE 位置 1，则允许 D/A 转换并输出转换结果。执行通道 0 的 D/A 转换时的运行例如下所示。此时的运行时序如图 21.2 所示。

1. 对 DADR0 写入转换数据。
2. DACR 的 DAOE0 位置 1 时，开始 D/A 转换，经过 $t_{\text{D CONV}}$ 时间后，通过模拟输出引脚 DA0 输出转换结果。该转换结果持续输出直到改写 DADR0 或 DAOE0 位清 0。输出值用以下公式表示：

$$\frac{\text{DADR 的内容}}{256} \times \text{AVref}$$

3. 改写 DADR0 后立即开始转换，经过 $t_{\text{D CONV}}$ 时间后，输出转换结果。
4. 如果 DAOE0 位清 0，则禁止模拟输出。

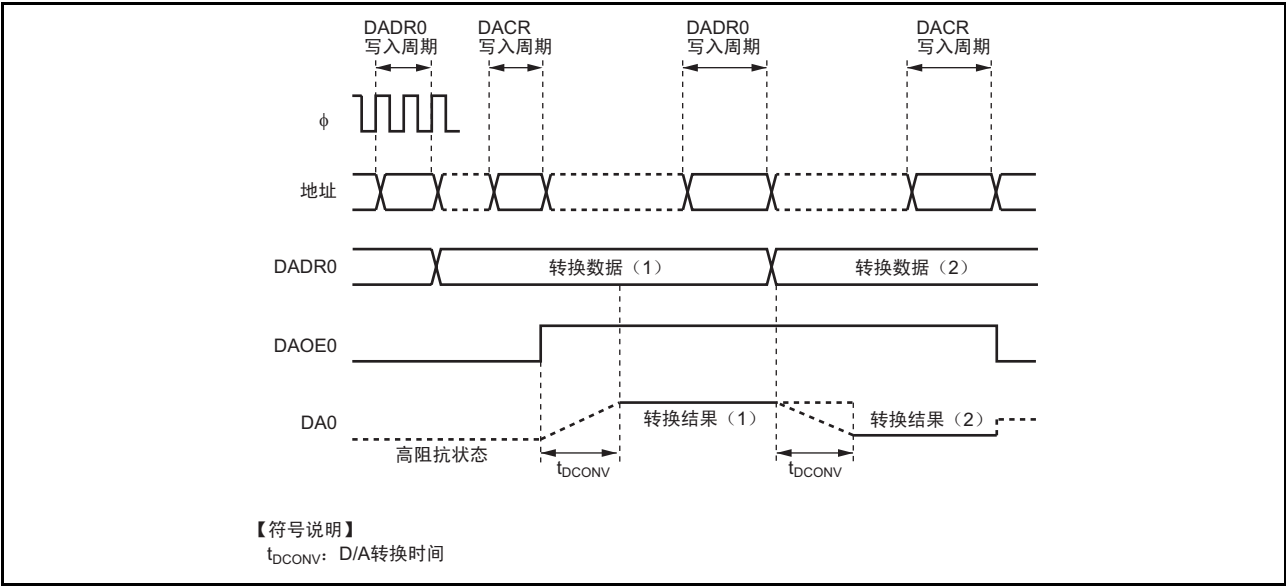


图 21.2 D/A 转换器的运行例

21.5 使用时的注意事项

21.5.1 模块待机模式的设定

通过待机控制寄存器，可禁止允许 D/A 转换器运行。为初始值时，D/A 转换停止运行。通过解除模块待机模式，可存取寄存器。详情参阅“第 28 章 低功耗模式”。

21.5.2 软件待机模式时的 D/A 输出保持功能

在允许 D/A 转换的状态下，如果本 LSI 进入软件待机模式，则保持 D/A 输出，且模拟电源电流与 D/A 转换过程中的电流相同。在软件待机模式需要降低模拟电源电流时，必须将 DAOE0、DAOE1、DAE 位全部清 0 并禁止 D/A 输出。

21.5.3 模拟输入电压的设定

超过以下所示电压的设定范围使用 LSI 时，会影响本 LSI 的可靠性。

1. AV_{CC} 、 AV_{SS} 输入电压

AV_{CC} 、 AV_{SS} 输入电压必须满足 $PV_{CC} - 0.3V \leq AV_{CC} \leq PV_{CC}$ 、 $AV_{SS} = PV_{SS}$ 。不使用 A/D 转换器及 D/A 转换器或处于软件待机模式时，必须将 AV_{CC} 、 AV_{SS} 引脚设置为开路。不使用时，必须将 AV_{CC} 连接到电源 (PV_{CC})，将 AV_{SS} 接地 (PV_{SS})。

2. AV_{ref} 的设定范围

AV_{ref} 引脚的基准电压范围必须设定为 $3.0V \leq AV_{ref} \leq AV_{CC}$ 。

21.5.4 深度待机模式时的 D/A 转换

转移至深度待机模式时，必须将 DAOE0、DAOE1、DAE 位全部清 0 并禁止 D/A 转换。在允许 D/A 转换的状态下，如果本 LSI 进入深度待机模式，则无法保证 D/A 引脚的状态。

第 22 章 AND/NAND 闪存控制器 (FLCTL)

AND/NAND 闪存控制器 (FLCTL) 提供外接 AND 型闪存与 NAND 型闪存的存储器接口。还具有 ECC 码生成及错误检测功能, 以解决闪存特有的读取问题。

【注】 本 LSI 不支持多层单元 (MLC) 产品的闪存。

22.1 特点

(1) AND 型闪存的存储器接口

- 可直接连结 AND 型闪存的接口
- 以扇区 (512+16 字节) 为单位读取/写入。执行 ECC 处理。
在 AND 型闪存数据表, 还有某些产品将 2048+64 字节的存取单位记述为页, 但本手册中, 将 512+16 字节统一为 1 个扇区, 将 1 页 2048+64 字节的产品以 512+16 字节单位 (每页 4 个扇区) 进行分割处理。
- 字节单位的读取/写入
- 扩展为 5 字节地址后, 可支持超过 2G 位的地址

(2) NAND 型闪存存储器的接口

- 可直接连结 NAND 型闪存的接口
- 以扇区 (512+16 字节) 为单位读取/写入。执行 ECC 处理。
在 NAND 型闪存数据表中, 还有某些产品将 2048+64 字节的存取单位记述为页, 但本手册中, 将 512+16 字节统一为 1 个扇区。将 1 页 2048+64 字节的产品以 512+16 字节单位 (每页 4 个扇区) 进行分割处理。
- 以字节单位读取/写入
- 扩展为 5 字节地址后, 可支持超过 2G 位的地址

(3) 存取模式: FLCTL 可选择以下 2 种存取模式:

- 命令存取模式:
在寄存器指定本 FLCTL 向闪存发行的命令、地址、输入/输出数据长度, 可执行连续的存取。据此可读取/写入及擦除无 ECC 处理的数据。
- 扇区存取模式:
指定扇区, 执行以扇区为单位的读取/写入, 控制 ECC 码生成/检查。指定扇区数, 可读取/写入连续扇区。

(4) 扇区与管理码

- 1 个扇区是存取的基本单位, 由 512 字节的数据和 16 字节的管理码构成。管理码包含 8 字节的 ECC。
- 能以 4 字节为单位, 指定管理码的 ECC 嵌入位置。
- 可对除 ECC 之外的管理码写入用户信息。

(5) ECC

- 对扇区（数据：512字节+管理码：16字节）生成8字节的ECC码并检查错误。
（但是，16字节的管理码中生成ECC并检查错误的字节数因设定而不同）
- 纠错能力：最多纠正任意3处错误。
- 写入时，数据及ECC之前的管理码为生成ECC码的对象，ECC后的管理码不属于ECC对象。
- 读取时，数据及ECC之前的管理码为ECC错误判定对象。FIFO内管理码的ECC并非从闪存读取的ECC码，而替换为通过ECC电路判定的结果。
- 产生ECC错误时不纠错，必须进行软件处理。

(6) 数据错误时

- 产生编程/擦除错误时，错误反映在错误源标志。可指定独立源的中断。
- 产生读取错误时，管理码的ECC不为0。此时，读取错误反映在ECC错误源标志。
- 产生ECC错误时，执行纠错，指定替代扇区，并根据需要将块的内容复制到替代扇区。

(7) 用于数据传送的 FIFO 和数据寄存器

- 内置用于传送闪存数据的224字节数据FIFO寄存器（FLDTFIFO）
- 内置用于传送管理码数据的32字节管理码FIFO寄存器（FLECFIFO）

(8) DMA 传送

- 通过在DMA控制器分别指定闪存数据和管理码传送目标，可向不同区域传送数据和管理码。

(9) 存取时间

- 通过分频外围时钟（Pφ），使用AND/NAND型闪存引脚的运行时钟（FCLK）。分频比可由共用控制寄存器（FLCMNCR）的FCKSEL位和QTSEL位指定。
- 必须在通过模块停止功能使FLCTL停止的状态下，更改CPG设定。
- 在NAND型闪存中，FSC引脚、FWE引脚以FCLK频率运行。在AND型闪存中，FSC引脚以FCLK频率运行，FWE引脚以1/2FCLK频率运行。设定时，不可超过连接存储器的最大工作频率。

FLCTL 框图，如图 22.1 所示。

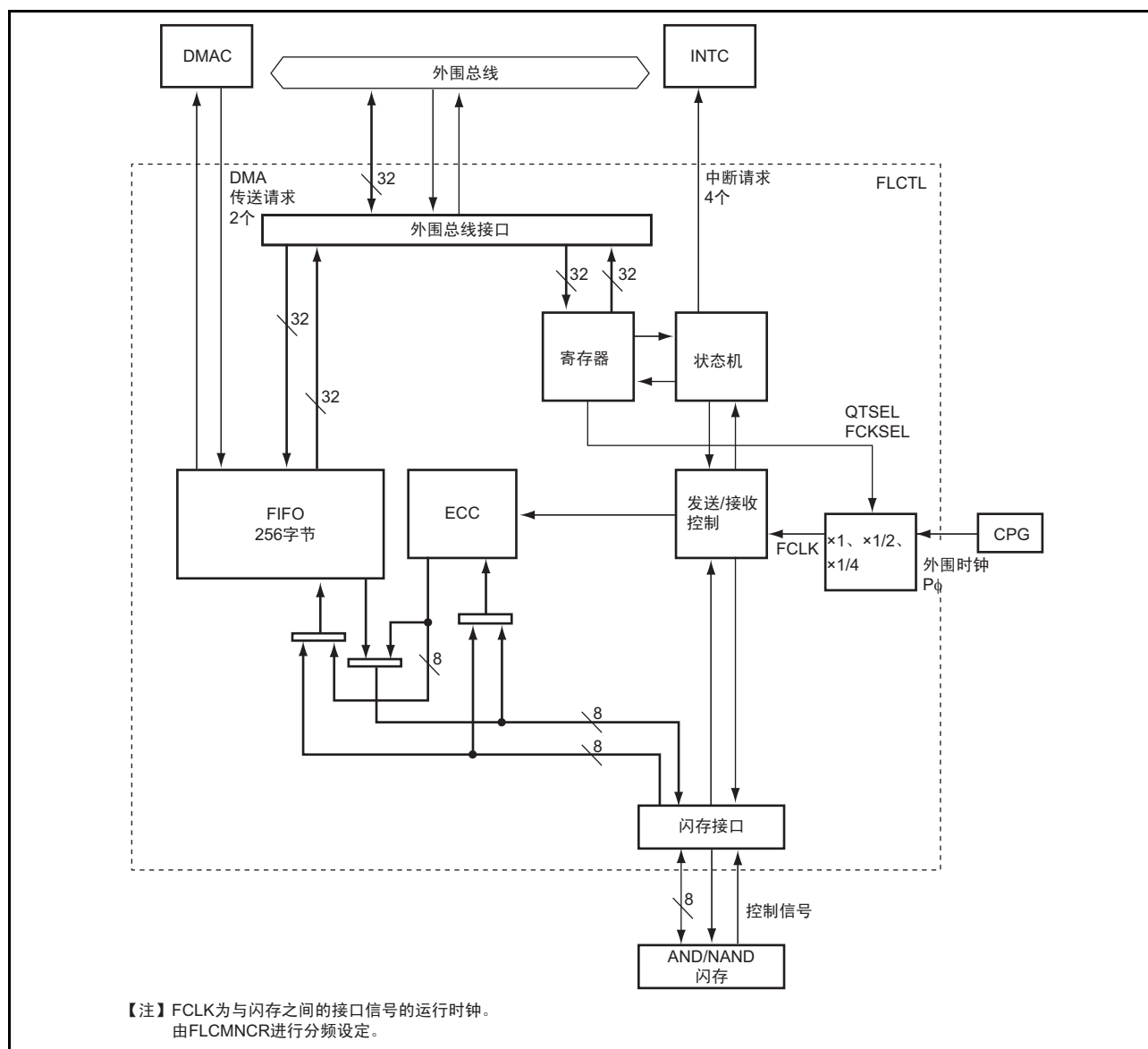


图 22.1 FLCTL 框图

22.2 输入 / 输出引脚

FLCTL 的引脚结构如表 22.1 所示。

表 22.1 引脚结构

引脚名称	输入 / 输出	对应的闪存引脚		功 能
		NAND 型	AND 型	
$\overline{\text{FCE}}$	输出	$\overline{\text{CE}}$	$\overline{\text{CE}}$	芯片选通 允许本 LSI 连接的闪存。
NAF7 ~ 0	输入 / 输出	I/O7 ~ I/O0	I/O7 ~ I/O0	数据输入 / 输出 命令、地址、数据的输入 / 输出引脚。
FCDE	输出	CLE	$\overline{\text{CDE}}$	命令锁定允许 (CLE) 命令输出时有效。 命令数据允许 ($\overline{\text{CDE}}$) 命令输出时有效。
FOE	输出	ALE	$\overline{\text{OE}}$	地址锁定允许 (ALE) 地址输出时有效。 数据输入 / 输出时无效。 输出允许 ($\overline{\text{OE}}$) 数据输入时 / 状态读时有效。
FSC	输出	$\overline{\text{RE}}$	SC	读取允许 ($\overline{\text{RE}}$) 在 $\overline{\text{RE}}$ 的下降沿读取数据。 串行时钟 (SC) 与 SC 同步输入 / 输出数据
$\overline{\text{FWE}}$	输出	$\overline{\text{WE}}$	$\overline{\text{WE}}$	写入允许 闪存在 $\overline{\text{WE}}$ 的上升沿锁定命令、地址及数据。
FRB	输入	R/ $\overline{\text{B}}$	R/ $\overline{\text{B}}$	就绪 / 忙 高电平表示就绪状态，低电平表示忙状态。
—*	—	$\overline{\text{WP}}$	$\overline{\text{RES}}$	写入保护 / 复位 在低电平，因上电 / 断电偶发的擦除 / 编程时进行保护。
—*	—	$\overline{\text{SE}}$	—	备用区域允许 可存取备用区域。使用扇区存取模式时，必须固定为低电平。

【注】 * 本 LSI 不支持。

22.3 寄存器说明

FLCTL 的寄存器结构如表 22.2 所示。

表 22.2 寄存器结构

寄存器名称	简称	R/W	初始值	地址	存取长度
共用控制寄存器	FLCMNCR	R/W	H'00000000	H'FFFFFF00	32
命令控制寄存器	FLCMDCR	R/W	H'00000000	H'FFFFFF04	32
命令码寄存器	FLCMCDR	R/W	H'00000000	H'FFFFFF08	32
地址寄存器	FLADR	R/W	H'00000000	H'FFFFFF0C	32
地址寄存器 2	FLADR2	R/W	H'00000000	H'FFFFFF03C	32
数据寄存器	FLDATAR	R/W	H'00000000	H'FFFFFF10	32
数据计数器寄存器	FLDTCNTR	R/W	H'00000000	H'FFFFFF14	32
中断 DMA 控制寄存器	FLINTDMACR	R/W	H'00000000	H'FFFFFF18	32
就绪 / 忙超时设定寄存器	FLBSYTMR	R/W	H'00000000	H'FFFFFF1C	32
就绪 / 忙超时计数器	FLBSCNT	R	H'00000000	H'FFFFFF20	32
数据 FIFO 寄存器	FLDTFIFO	R/W	H'xxxxxxxx	H'FFFFFF50	32
管理码 FIFO 寄存器	FLECFIFO	R/W	H'xxxxxxxx	H'FFFFFF60	32
传送控制寄存器	FLTRCR	R/W	H'00	H'FFFFFF2C	8

22.3.1 共用控制寄存器 (FLCMNCR)

FLCMNCR 为 32 位可读取 / 写入的寄存器，指定闪存类型 (AND/NAND) 及存取模式等。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	SNAND	QT SEL	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FCK SEL	-	ECCPOS[1:0]	ACM[1:0]	NAND WF	-	-	-	-	-	-	CE	-	-	TYPE SEL	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R/W	R	R	R/W

位	位名称	初始值	R/W	说 明
31 ~ 19	—	均为 0	R	保留位 读取值、写入值总是为 0。
18	SNAND	0	R/W	大容量 NAND 闪存选择位 本位用于在超过 1G 位的闪存中，指定 1 页的结构为 2048+64 字节的 NAND 闪存及超过 1G 位的 AG-AND 闪存。 0: 用于 1 页结构为 512+16 字节的闪存及 AND 型闪存时，必须设定为 0 1: 用于 1 页结构为 2048+64 字节的 NAND 型闪存及超过 1G 位的 AG-AND 闪存时，必须设定为 1 【注】 TYPESEL=0 时，禁止设定为 1。

位	位名称	初始值	R/W	说 明
31 ~ 19	—	均为 0	R	保留位 读取值、写入值总是为 0。
18	SNAND	0	R/W	大容量 NAND 闪存选择位 本位用于在超过 1G 位的闪存中，指定 1 页的结构为 2048+64 字节的 NAND 闪存及超过 1G 位的 AG-AND 闪存。 0: 用于 1 页结构为 512+16 字节的闪存及 AND 型闪存时，必须设定为 0 1: 用于 1 页结构为 2048+64 字节的 NAND 型闪存及超过 1G 位的 AG-AND 闪存时，必须设定为 1 【注】 TYPESEL=0 时，禁止设定为 1。
17	QTSEL	0	R/W	闪存时钟分频选择位 本位为闪存内使用时钟 FCLK 的分频选择位，配合 FCKSEL 使用。 QTSEL=0、FCKSEL=0: 将 CPG 提供的时钟 (Pφ) 1/2 分频，作为 FCLK 使用 QTSEL=0、FCKSEL=1: 将 CPG 提供的时钟 (Pφ) 直接作为 FCLK 使用。 QTSEL=1、FCKSEL=0: 将 CPG 提供的时钟 (Pφ) 1/4 分频，作为 FCLK 使用。 QTSEL=1、FCKSEL=1: 禁止设定
16	—	0	R	保留位 读取值、写入值总是为 0。
15	FCKSEL	0	R/W	闪存时钟选择位 本位为闪存内使用时钟 FCLK 的分频选择位，配合 QTSEL 使用，详情参阅 QTSEL 的说明。
14	—	0	R	保留位 读取值、写入值总是为 0。
13、12	ECCPOS[1:0]	00	R/W	ECC 嵌入位置指定位 1、0 本位 指定管理码区域内 ECC 的嵌入位置 (第 0/4/8 字节)。 00: 在管理码区域的第 0 ~ 7 字节配置 ECC 01: 在管理码区域的第 4 ~ 11 字节配置 ECC 10: 在管理码区域的第 8 ~ 15 字节配置 ECC 11: 禁止设定
11、10	ACM[1:0]	00	R/W	存取模式指定位 1、0 本位指定存取模式。 00: 命令存取模式 01: 扇区存取模式 10: 禁止设定 11: 禁止设定
9	NANDWF	0	R/W	NAND 等待插入运行位 0: 以 1 个 FCLK 周期输入 / 输出地址、数据 1: 以 2 个 FCLK 周期输入 / 输出地址、数据
8 ~ 4	—	均为 0	R	保留位 读取值、写入值总是为 0。

位	位名称	初始值	R/W	说 明
3	CE	0	R/W	芯片允许位 0: 禁止 (向 $\overline{\text{FCE}}$ 引脚输出高电平。) 1: 允许 (向 $\overline{\text{FCE}}$ 引脚输出低电平。)
2、1	—	均为 0	R	保留位 读取值、写入值总是为 0。
0	TYPESEL	0	R/W	存储器选择位 0: 选择 AND 型闪存。 1: 选择 NAND 型闪存、AG-AND。

22.3.2 命令控制寄存器 (FLCMDCR)

FLCMDCR 为 32 位可读取 / 写入的寄存器，可指定命令存取模式时是否发行命令、地址及数据的输入 / 输出地址；在扇区存取模式可指定扇区传送次数。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ADR CNT2	SCTCNT[19:16]					ADR MD	CDS RC	DOSR	-	-	SEL RW	DOA DR	ADRCNT[1:0]	DOC MD2	DOC MD1
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SCTCNT[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
31	ADRCNT2	0	R/W	地址发行字节数指定位 2 本位指定在地址阶段发行的地址数据的字节数，配合 ADRCNT[1:0] 位使用。 0: 仅发行 ADRCNT[1:0] 指定字节数的地址。 1: 发行 5 字节的地址 ADRCNT[1:0] 必须设定为 00。
30 ~ 27	SCTCNT [19:16]	均为 0	R/W	扇区传送次数指定位 [19:16] 本位为是扇区传送次数指定位 SCTCNT[15:0] 的扩展位。 SCTCNT[19:16] 与 SCTCNT[15:0] 配合，作为 SCTCNT[19:0] 的 20 位计数器运行。
26	ADRMD	0	R/W	扇区存取地址指定位 在命令存取模式，本位无效。仅在扇区存取模式有效。 0: 地址寄存器的值作为物理扇区编号处理。 扇区存取时，通常使用该值。 1: 地址寄存器的值直接作为闪存的地址输出。 【注】 连续扇区存取时，必须设定为 0。
25	CDSRC	0	R/W	数据缓冲器指定位 本位在命令存取模式时指定数据阶段读取 / 写入的数据缓冲器。 0: 指定 FLDATAR 作为数据缓冲器 1: 指定 FLDTFIFO 作为数据缓冲器

位	位名称	初始值	R/W	说 明
24	DOSR	0	R/W	状态读取检查位 本位在命令存取模式时发行第 2 个命令后, 指定是否读取状态。 0: 不读取状态 1: 读取状态
23、22	—	均为 0	R	保留位 读取值、写入值总是为 0。
21	SELRW	0	R/W	数据读取 / 写入指定位 本位指定在数据阶段的读取 / 写入方向。 0: 读取 1: 写入
20	DOADR	0	R/W	地址阶段执行指定位 本位在命令存取模式时, 指定是否执行地址阶段。 0: 不执行地址阶段 1: 执行地址阶段
19、18	ADRCNT[1:0]	00	R/W	地址发行字节数指定位 1、0 本位指定在地址阶段发行的地址数据的字节数。 00: 发行 1 字节的地址 01: 发行 2 字节的地址 10: 发行 3 字节的地址 11: 发行 4 字节的地址
17	DOCMD2	0	R/W	第 2 命令阶段执行指定位 本位在命令存取模式时, 指定是否执行第 2 命令阶段。 0: 不执行第 2 命令阶段 1: 执行第 2 命令阶段
16	DOCMD1	0	R/W	第 1 命令阶段执行指定位 本位在命令存取模式时, 指定是否执行第 1 命令阶段。 0: 不执行第 1 命令阶段 1: 执行第 1 命令阶段
15 ~ 0	SCTCNT[15:0]	均为 0	R/W	扇区传送次数指定位 [15:0] 指定在扇区存取模式时连续读取的扇区数。每结束 1 扇区的传送即递减计数, 为 0 时停止。与 SCTCNT[19:16] 配合使用。在命令存取模式下, 运行时为 H'00001。

22.3.3 命令码寄存器 (FLCMCDR)

FLCMCDR 为 32 位可读取 / 写入的寄存器，指定命令存取、扇区存取时发行的命令值。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMD2[7:0]								CMD1[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
31 ~ 16	—	均为 0	R	保留位 读取值、写入值总是为 0。
15 ~ 8	CMD2[7:0]	均为 0	R/W	第 2 命令数据位 本位指定在第 2 命令阶段发行的命令码。
7 ~ 0	CMD1[7:0]	均为 0	R/W	第 1 命令数据位 本位指定在第 1 命令阶段发行的命令码。

22.3.4 地址寄存器 (FLADR)

FLADR 为 32 位可读取 / 写入的寄存器，指定作为地址的输出值。从 ADR1 依次将命令寄存器指定的字节数以字节为单位作为地址输出。由命令控制寄存器的扇区存取地址指定位 (ADRMD)，可指定是否输出地址数据位指定的扇区编号转换为地址后的值。

- ADRMD=1 时

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ADR4[7:0]								ADR3[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADR2[7:0]								ADR1[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
31 ~ 24	ADR4[7:0]	均为 0	R/W	第 4 地址数据位 ADRMD=1 时，本位指定第 4 次输出至闪存作为地址的数据。
23 ~ 16	ADR3[7:0]	均为 0	R/W	第 3 地址数据位 ADRMD=1 时，本位指定第 3 次输出至闪存作为地址的数据。
15 ~ 8	ADR2[7:0]	均为 0	R/W	第 2 地址数据位 ADRMD=1 时，本位指定第 2 次输出至闪存作为地址的数据。
7 ~ 0	ADR1[7:0]	均为 0	R/W	第 1 地址数据位 ADRMD=1 时，本位指定第 1 次输出至闪存作为地址的数据。

- ADRMD=0 时

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	ADR[25:16]									
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADR[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
31 ~ 26	—	均为 0	R	保留位 读取值、写入值总是为 0。
25 ~ 0	ADR[25:0]	均为 0	R/W	扇区地址指定位 ADRMD=0 时, 本位指定存取的扇区编号。扇区编号转换为地址后, 输出至闪存。FLCMDRCR 的 ADRCNT2 位为 1 时, ADR[25:0] 有效; 为 0 时, ADR[17:0] 有效, 详情参照图 22.15。 • 大块产品 (2048+64 字节) ADR[25:2] 可指定页地址, ADR[1:0] 可指定扇区单位的列地址。 ADR[1:0]=00: 第 0 字节 (0 扇区) ADR[1:0]=01: 第 512+16 字节 (1 扇区) ADR[1:0]=10: 第 1024+32 字节 (2 扇区) ADR[1:0]=11: 第 1536+48 字节 (3 扇区) • 小块产品 (512+16 字节) 仅指定页地址。

22.3.5 地址寄存器 2 (FLADR2)

FLADR2 为 32 位可读取 / 写入的寄存器, FLCMDRCR 的 ADRCNT2 位为 1 时有效, 在命令存取模式指定作为地址的输出值。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	ADR5[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
31 ~ 8	—	均为 0	R	保留位 读取值、写入值总是为 0。
7 ~ 0	ADR5[7:0]	均为 0	R/W	第 5 地址数据位 ADRMD=1 时, 本位指定第 5 次输出至闪存作为地址的数据。

22.3.6 数据计数器寄存器 (FLDTCNTR)

FLDTCNTR 为 32 位可读取 / 写入的寄存器，在命令存取模式时，指定读取 / 写入的字节数。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECFLW[7:0]								DTFLW[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	DTCNT[11:0]											
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
31 ~ 24	ECFLW[7:0]	均为 0	R	FLECFIFO 存取数位 本位表示 FLECFIFO 可读取 / 写入的长字数。 可用于 CPU 读取 / 写入 FLECFIFO 时。 读取 FLECFIFO 时，表示 FLECFIFO 内可读取数据的长字数。 写入 FLECFIFO 时，表示 FLECFIFO 内可写入的空长字数。
23 ~ 16	DTFLW[7:0]	均为 0	R	FLDTFIFO 存取数位 本位表示 FLDTFIFO 可读取 / 写入的长字数。 可用于 CPU 读取 / 写入 FLDTFIFO 时。 读取 FLDTFIFO 时，表示 FLDTFIFO 内可读取数据的长字数。 写入 FLDTFIFO 时，表示 FLDTFIFO 内可写入的空长字数。
15 ~ 12	—	均为 0	R	保留位 读取值、写入值总是为 0。
11 ~ 0	DTCNT[11:0]	均为 0	R/W	数据数指定位 本位在命令存取模式下指定读取 / 写入数据的字节数（最大可指定 2048+64 字节）。

22.3.7 数据寄存器 (FLDATAR)

FLDATAR 为 32 位可读取 / 写入的寄存器。

本寄存器保存输入 / 输出数据，该数据在命令存取模式设定 FLCMDCR 的 CDSRC 位为 0 时使用。读取 / 写入超过 5 字节的连续数据时不可使用。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DT4[7:0]								DT3[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DT2[7:0]								DT1[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
31 ~ 24	DT4[7:0]	均为 0	R/W	第 4 数据位 本位指定从 NAF7 ~ 0 第 4 个输入 / 输出的数据。 写入时: 指定写入数据。 读取时: 保存读取数据。
23 ~ 16	DT3[7:0]	均为 0	R/W	第 3 数据位 本位指定从 NAF7 ~ 0 第 3 个输入 / 输出的数据。 写入时: 指定写入数据。 读取时: 保存读取数据。
15 ~ 8	DT2[7:0]	均为 0	R/W	第 2 数据位 本位指定从 NAF7 ~ 0 第 2 个输入 / 输出的数据。 写入时: 指定写入数据。 读取时: 保存读取数据。
7 ~ 0	DT1[7:0]	均为 0	R/W	第 1 数据位 本位指定从 NAF7 ~ 0 第 1 个输入 / 输出的数据。 写入时: 指定写入数据。 读取时: 保存读取数据。

22.3.8 中断 DMA 控制寄存器 (FLINTDMACR)

FLINTDMACR 为 32 位可读取 / 写入的寄存器，设定允许 / 禁止 DMA 传送请求及中断。各存取模式开始运行后，产生 FLCTL 对 DMAC 的传送请求。

bit9 ~ 5 是表示闪存存取时的各种错误及是否有来自 FIFO 的传送请求的标志位，仅可写入 0。清除标志时，必须对需清除的标志位写入 0，其他标志位写入 1。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	ECER INTE	-	-	FIFOTRG [1:0]	AC1 CLR	AC0 CLR	DREQ1 EN	DREQ0 EN	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	EC ERB	ST ERB	BTO ERB	TRR EQF1	TRR EQF0	STER INTE	RBER INTE	TE INTE	TR INTE1	TR INTE0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/W	R/W	R/W	R/W	R/W

【注】* 仅在写入0时有效。

位	位名称	初始值	R/W	说 明
31 ~ 25	—	均为 0	R	保留位 读取值、写入值总是为 0。
24	ECERINTE	0	R/W	ECC 错误中断允许位 0: 禁止产生 ECC 错误时的中断 1: 允许产生 ECC 错误时的中断
23、22	—	均为 0	R	保留位 读取值、写入值总是为 0。
21、20	FIFOTRG[1:0]	00	R/W	FIFO 触发设定位 切换 FLDTFIFO、FLECFIFO 的传送请求产生条件。 - 读取闪存时 00: FLDTFIFO 已保存 4 字节时，向 CPU 产生中断，或产生 DMA 传送请求。 01: FLDTFIFO 已保存 16 字节时，向 CPU 产生中断，或产生 DMA 传送请求。 10: FLDTFIFO 已保存 128 字节时，向 CPU 产生中断，或已保存 4 字节时，产生 DMA 传送请求。 11: FLDTFIFO 已保存 128 字节时，向 CPU 产生中断，或已保存 16 字节时，产生 DMA 传送请求。 - 写入闪存时 00: FLDTFIFO 有超过 4 字节空间时，向 CPU 产生中断。（不可设定 DMA 传送） 01: FLDTFIFO 有超过 16 字节空间时，向 CPU 产生中断，或产生 DMA 传送请求。 10: FLDTFIFO 有超过 128 字节空间时，向 CPU 产生中断。（不可设定 DMA 传送） 11: FLDTFIFO 有超过 128 字节空间时，向 CPU 产生中断，或有超过 16 字节空间时，产生 DMA 传送请求。 【注】* 仅可用 FIFOTRG[0] 的值转换 FLECFIFO 0: FLECFIFO 已保存 4 字节时，向 CPU 产生中断，或产生 DMA 传送请求。 1: FLECFIFO 已保存 16 字节时，向 CPU 产生中断，或产生 DMA 传送请求。 【注】* 仅可用 FIFOTRG[0] 的值转换 FLECFIFO 0: FLECFIFO 有超过 4 字节空间时，向 CPU 产生中断。（不可设定 DMA 传送） 1: FLECFIFO 有超过 16 字节空间时，向 CPU 产生中断，或产生 DMA 传送请求。

位	位名称	初始值	R/W	说 明
19	AC1CLR	0	R/W	FLECFIFO 清除位 本位清除 FLECFIFO。更改读取 / 写入方向时, 必须清除 FIFO。 0: 保持 FLECFIFO 的值。闪存存取时, 必须设定为 0。 1: 清除 FLECFIFO。清除后, 必须设定为 0。
18	AC0CLR	0	R/W	FLDTFIFO 清除位 本位清除 FLDTFIFO。更改读取 / 写入方向时, 必须清除 FIFO。 0: 保持 FLDTFIFO 的值。闪存存取时, 必须设定为 0。 1: 清除 FLDTFIFO 的值。清除后, 必须设定为 0。
17	DREQ1EN	0	R/W	FLECFIFODMA 请求允许位 本位选择允许 / 禁止发行 FLECFIFO 区域的 DMA 传送请求。 0: 禁止发行 FLECFIFO 区域的 DMA 传送请求 1: 允许发行 FLECFIFO 区域的 DMA 传送请求
16	DREQ0EN	0	R/W	FLDTFIFODMA 请求允许位 本位选择允许 / 禁止发行 FLDTFIFO 区域的 DMA 传送请求。 0: 禁止发行 FLDTFIFO 区域的 DMA 传送请求 1: 允许发行 FLDTFIFO 区域的 DMA 传送请求
15 ~ 10	—	均为 0	R	保留位 读取值、写入值总是为 0。
9	ECERB	0	R/(W)*	ECC 错误位 本位显示 ECC 错误检测结果。在扇区存取模式下读取闪存时, 如果产生 ECC 错误, 则本位置 1。 本位为标志位, 不可写入 1。为了清除标志, 仅可写入 0。 0: 无 ECC 错误 (表示输入的 ECC 均为 0) 1: 表示产生 ECC 错误。
8	STERB	0	R/(W)*	状态错误位 本位表示状态读取的结果。读取状态时, 如果 FLBSYCNT 的 STAT[7:0] 位的特定定位为 1, 则本位置 1。 本位为标志位, 不能写入 1。为了清除标志, 仅可写入 0。 0: 无状态错误 (表示 FLBSYCNT 的 STAT[7:0] 位的特定定位为 0) 1: 表示产生状态错误 有关特定定位, 详情参阅 “22.4.7 状态读取”。
7	BTOERB	0	R/(W)*	R/B 超时错误位 产生 R/B 超时错误时 (FLBSYCNT 的 RBTIMCNT[19:0] 位在递减计数后为 0), 本位置 1。 本位为标志位, 不能写入 1。为了清除标志, 仅可写入 0。 0: 无 R/B 超时错误 1: 表示产生 R/B 超时错误。
6	TRREQF1	0	R/(W)*	FLECFIFO 传送请求标志位 本位表示产生 FLECFIFO 的传送请求。 本位为标志位, 不能写入 1。为了清除标志, 仅可写入 0。 0: 表示未产生 FLECFIFO 的传送请求 1: 表示产生 FLECFIFO 的传送请求
5	TRREQF0	0	R/(W)*	FLDTFIFO 传送请求标志位 本位表示产生 FLDTFIFO 的传送请求。 本位为标志位, 不能写入 1。为了清除标志, 仅可写入 0。 0: 表示未产生 FLDTFIFO 的传送请求 1: 表示产生 FLDTFIFO 的传送请求

位	位名称	初始值	R/W	说 明
4	STERINTE	0	R/W	产生状态错误时的中断允许位 本位选择允许 / 禁止因状态错误而对 CPU 产生中断。 0: 禁止因状态错误对 CPU 产生中断 1: 允许因状态错误对 CPU 产生中断
3	RBERINTE	0	R/W	产生 R/B 超时错误时的中断允许位 本位选择允许 / 禁止因 R/B 超时错误而对 CPU 产生中断。 0: 禁止因 R/B 超时错误对 CPU 产生中断 1: 允许因 R/B 超时错误对 CPU 产生中断
2	TEINTE	0	R/W	传送结束中断允许位 本位选择允许 / 禁止因传送结束 (FLTRCR 的 TREND 位) 而对 CPU 产生中断。 0: 禁止因传送结束对 CPU 产生中断 1: 允许因传送结束对 CPU 产生中断
1	TRINTE1	0	R/W	对 CPU 的 FLECFIFO 传送请求允许位 本位选择允许 / 禁止因 FLECFIFO 的传送请求而对 CPU 产生中断。 0: 禁止因 FLECFIFO 的传送请求对 CPU 产生中断 1: 允许因 FLECFIFO 的传送请求对 CPU 产生中断 允许 DMA 传送时, 必须将本位设定为 0。
0	TRINTE0	0	R/W	对 CPU 的 FLDTFIFO 传送请求允许位 本位选择允许 / 禁止因 FLDTFIFO 的传送请求而对 CPU 产生中断。 0: 禁止因 FLDTFIFO 的传送请求对 CPU 产生中断 1: 允许因 FLDTFIFO 的传送请求对 CPU 产生中断 允许 DMA 传送时, 必须将本位设定为 0。

【注】 * 仅写入 0 有效。

22.3.9 就绪 / 忙超时设定寄存器 (FLBSYTMR)

FLBSYTMR 为 32 位可读取 / 写入的寄存器, 设定 FRB 引脚为忙状态时的超时时间。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	RBTMOU[19:16]			
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RBTMOU[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
31 ~ 20	—	均为 0	R	保留位 读取值、写入值总是为 0。
19 ~ 0	RBTMOU[19:0]	均为 0	R/W	就绪 / 忙超时位 (以 P _φ 的时钟数) 设定忙状态超时之前的时间。设定为 0 时, 不超时。

22.3.10 就绪 / 忙超时计数器 (FLBSYCNT)

FLBSYCNT 为 32 位只读寄存器。

通过因状态读取运行，将读取的闪存状态保存至 STAT[7:0]。

FRB 引脚为忙状态时，将 FLBSYTMCR 的 RBTMOUT[19:0] 位设定的超时时间复制到 RBTIMCNT[19:0] 位，开始递减计数。RBTIMCNT[19:0] 位的值为 0 时，FLINTDMACR 的 BTOERB 位置 1，并通知产生超时错误。此时如果 FLINTDMACR 的 RBERINTE 位允许中断，则发行 FLSTE 中断。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	STAT[7:0]							-	-	-	-	RBTIMCNT[19:16]				
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RBTIMCNT[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名称	初始值	R/W	说 明
31 ~ 24	STAT[7:0]	H'00	R	本位表示从闪存读取状态的值。
23 ~ 20	—	均为 0	R	保留位 读取值为 0。
19 ~ 0	RBTIMCNT[19:0]	均为 0	R	就绪 / 忙超时计数器位 FRB 引脚为忙状态时，FLBSYTMCR 的 RBTMOUT[19:0] 位的设定值被复制到本位。 之后 FRB 引脚为忙状态期间，本位的值递减计数，为 0 时，产生超时错误。

22.3.11 数据 FIFO 寄存器 (FLDTFIFO)

FLDTFIFO 是对数据 FIFO 区域读取 / 写入的寄存器。

DMA 传送时，必须将本寄存器指定为数据的传送目标（传送源）。16 字节 DMA 传送时，必须从 16 字节地址边界处的地址存取到 FLDTFIFO。

必须使 FLCMDCR 的 SELRW 位指定的读取 / 写入方向与本寄存器读取 / 写入方向一致。另外，更改读取 / 写入方向时，必须由 FLINTDMACR 的 AC0CLR 位清除 FLDTFIFO 后再使用。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DTFO[31:16]															
初始值:	不定	不定	不定	不定	不定	不定	不定	不定	不定	不定	不定	不定	不定	不定	不定	不定
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTFO[15:0]															
初始值:	不定	不定	不定	不定	不定	不定	不定	不定	不定	不定	不定	不定	不定	不定	不定	不定
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

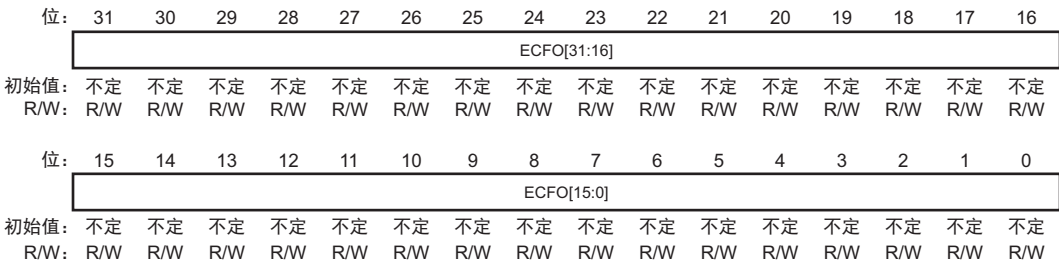
位	位名称	初始值	R/W	说 明
31 ~ 0	DTFO[31:0]	H'xxxxxxxx	R/W	对数据 FIFO 区域用于读取 / 写入的寄存器 写入时：将数据写入数据 FIFO 区域。 读取时：读取数据 FIFO 区域的数据。

22.3.12 管理码 FIFO 寄存器（FLECFIFO）

FLECFIFO 为读取 / 写入的管理码 FIFO 区域的寄存器。

DMA 传送时，必须将本寄存器指定为数据的传送目标（传送源）。传送 16 字节 DMA 时，必须从 16 字节地址边界处的地址存取 FLECFIFO。

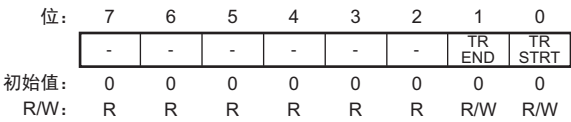
必须使 FLCMDCR 的 SELRW 位指定的读取 / 写入方向与本寄存器读取 / 写入方向一致。另外，更改读取 / 写入方向时，必须由 FLINTDMACR 的 AC1CLR 位清除 FLECFIFO 后再使用。



位	位名称	初始值	R/W	说 明
31 ~ 0	ECFO[31:0]	H'xxxxxxxx	R/W	对管理码 FIFO 区域用于读取 / 写入的寄存器 写入时：将数据写入管理码 FIFO 区域。 读取时：读取管理码 FIFO 区域的数据。

22.3.13 传送控制寄存器（FLTRCR）

将 TRSTRT 位置 1，开始存取闪存。通过 TREND 位可确认存取是否完成。正在传送（从 TRSTRT 位置 1 到 TREND 位置 1 期间）时不可强制结束（TRSTRT 位置 0）。在读取闪存完成时，TREND 置位。如果 FIFO 保留读取数据，则不可在 FIFO 读取结束前强制结束。



位	位名称	初始值	R/W	说 明
7 ~ 2	—	均为 0	R	保留位 读取值、写入值总是为 0。
1	TREND	0	R/W	处理结束标志位 本位表示指定的存取模式处理已结束。写入时，必须写入 0。
0	TRSTRT	0	R/W	传送开始位 TREND 为 0 时，通过将 TRSTR 从 0 置 1，开始存取模式指定位 ACM[1:0] 指定的存取模式处理。 0：停止传送 1：开始传送

22.4 运行说明

22.4.1 存取步骤

FLCTL 的存取是将一系列的存取步骤分割为独立的阶段。

例如，AND 型闪存的编程由以下 5 个阶段构成。

- 第1命令发送阶段（写入设置命令）
- 地址发送阶段（写入地址）
- 数据阶段（输出）
- 第2命令发送阶段（写入开始命令）
- 状态读取

按照以上 5 个阶段的顺序可实现一系列的存取，最后阶段（此例中为状态读取）结束时，完成对闪存的存取。

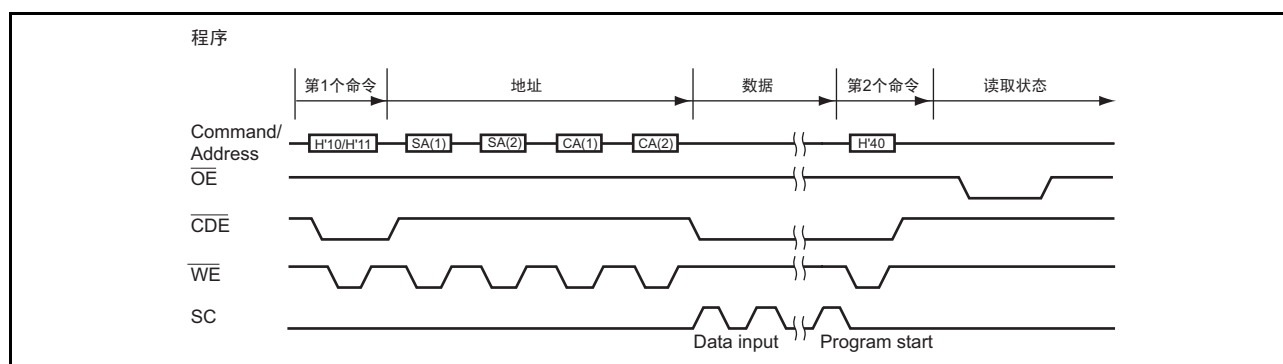


图 22.2 AND 型闪存的写入操作按阶段区分的例子

有关 AND 型闪存的读取、NAND 型闪存的读取 / 写入，详情参阅“22.4.4 命令存取模式”。

22.4.2 运行模式

运行模式有以下 2 种：

- 命令存取模式
- 扇区存取模式

在扇区存取模式执行 ECC 的生成 / 错误检查。

22.4.3 寄存器设定步骤

存取闪存所必需的寄存器设定流程如图 22.3 所示。

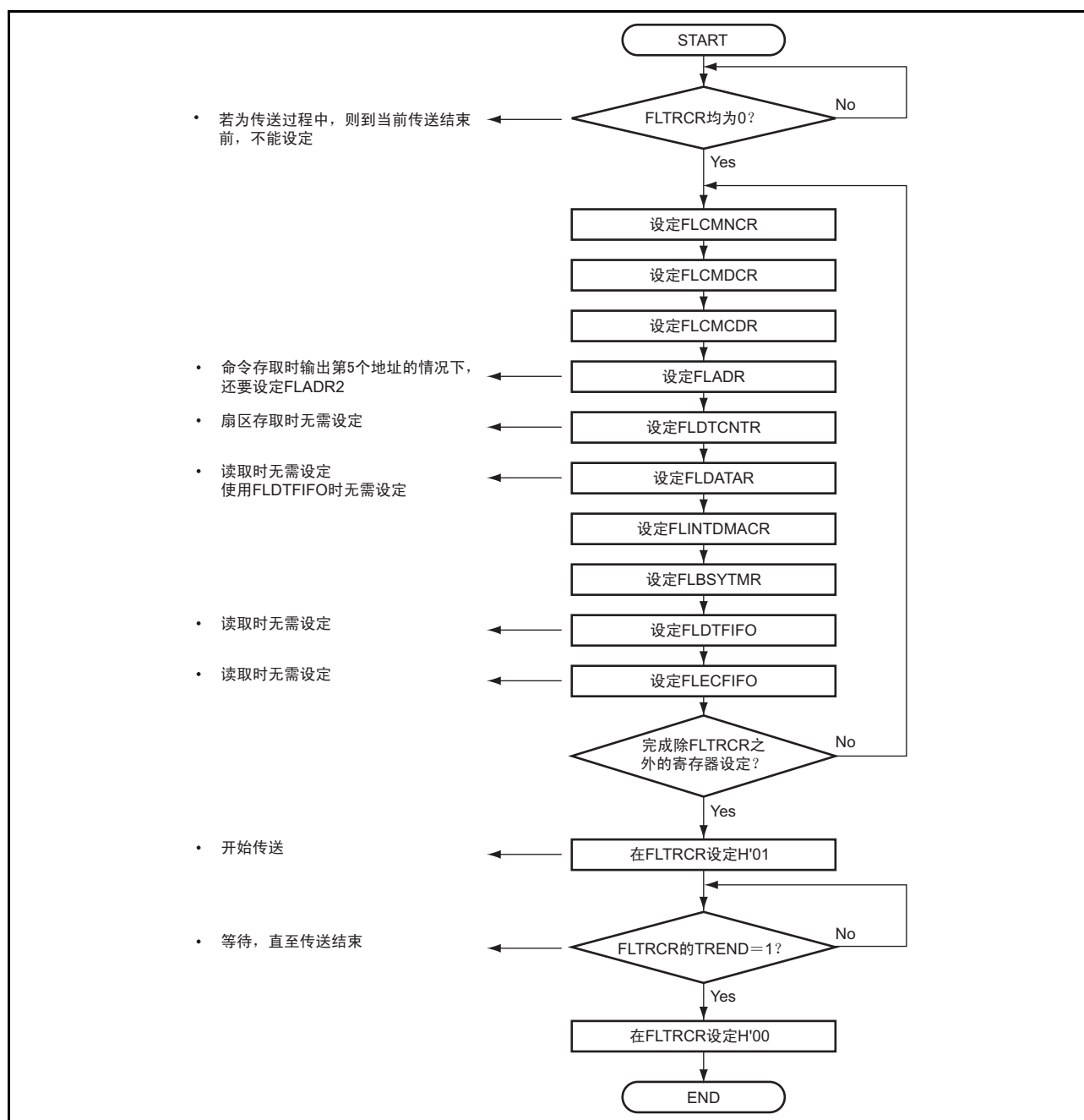


图 22.3 寄存器设定流程

22.4.4 命令存取模式

在命令存取模式下，通过对寄存器设定向闪存发行的命令、地址、数据、读取 / 写入方向及次数等存取闪存。输入 / 输出数据可使用 FLDTFIFO 进行 DMA 传送。

(1) AND 型闪存的存取

AND 型闪存的读取例如图 22.4、图 22.5 所示。在本例，第 1 命令指定为 H'00，地址数据长度指定为 2 字节的 SA1、SA2（仅指定 SA1、2，不指定 CA1、2），数据计数器的读取字节数指定为 4 字节，第 2 命令指定为 H'FF。

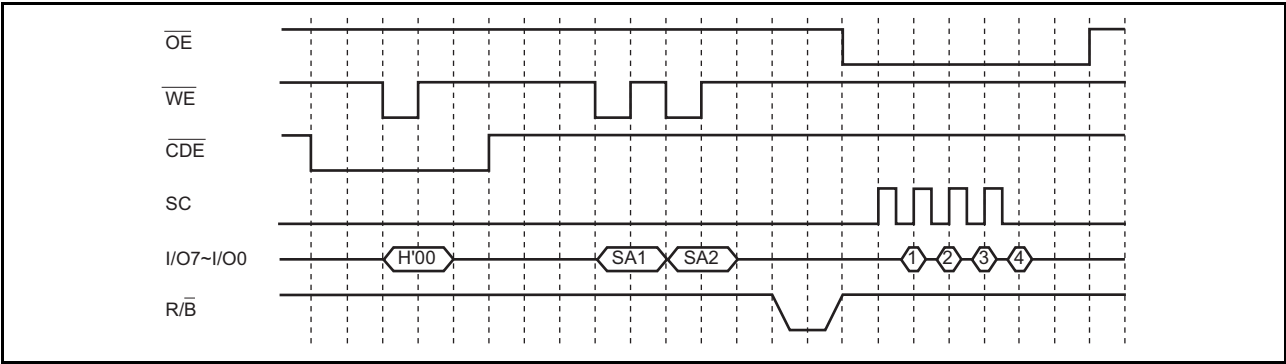


图 22.4 AND 型闪存的读取操作时序 (1)

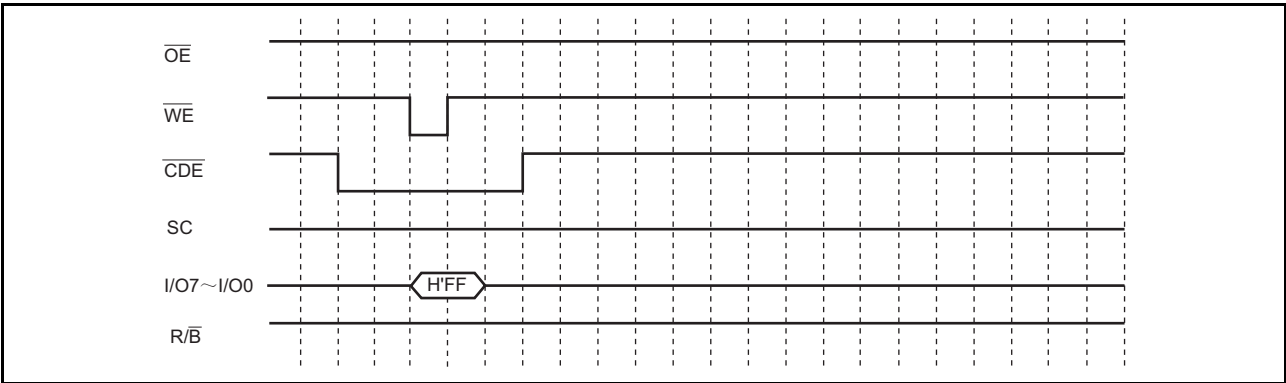


图 22.5 AND 型闪存的读取操作时序 (2)

对 AND 型闪存进行写入操作的程波形如图 22.6、图 22.7 所示。

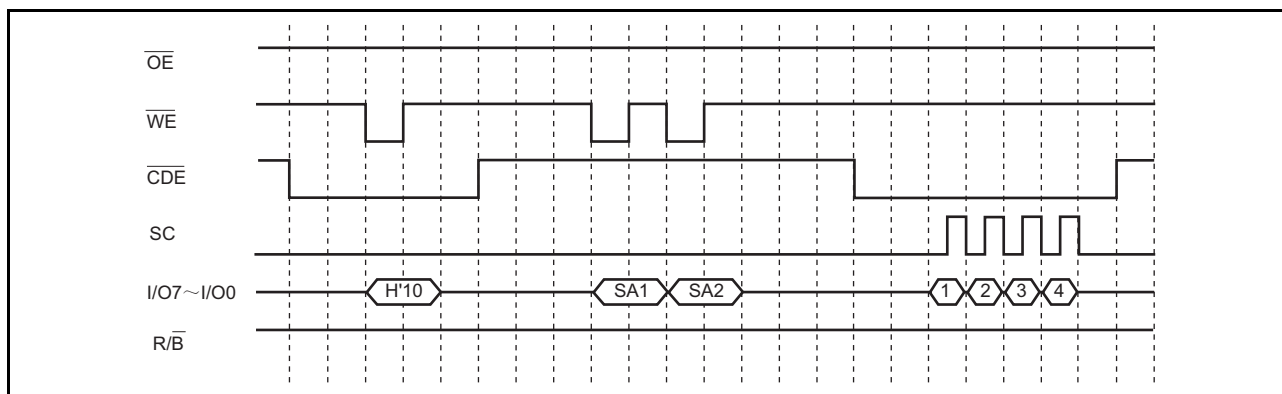


图 22.6 AND 型闪存写入操作时序 (1)

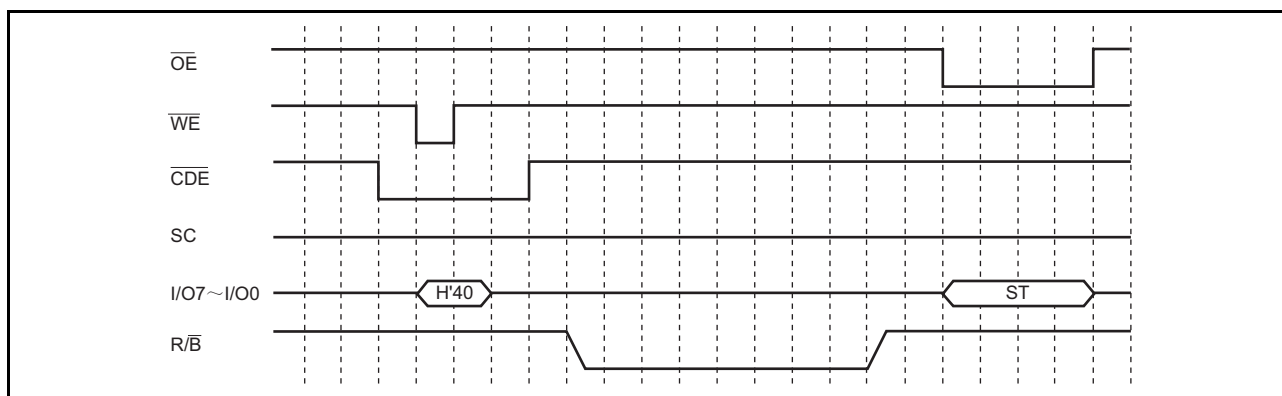


图 22.7 AND 型闪存写入操作时序 (2)

(2) NAND 型闪存的存取

NAND 型闪存的读取例如图 22.8 所示。本例中指定第 1 命令为 H'00，地址数据长度为 3 字节，数据计数器的读取字节数为 8 字节。

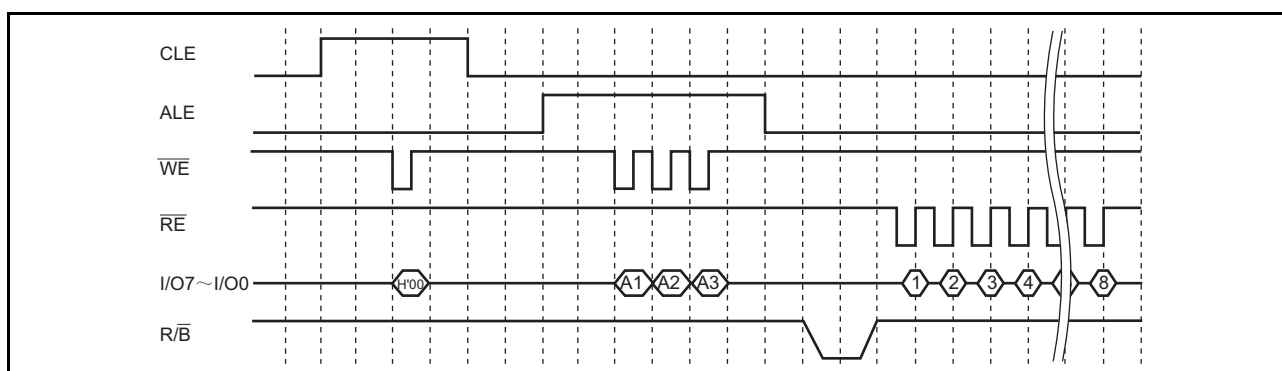


图 22.8 NAND 型闪存的读取操作时序

对 NAND 型闪存进行写入操作的波形如图 22.9、图 22.10 所示。

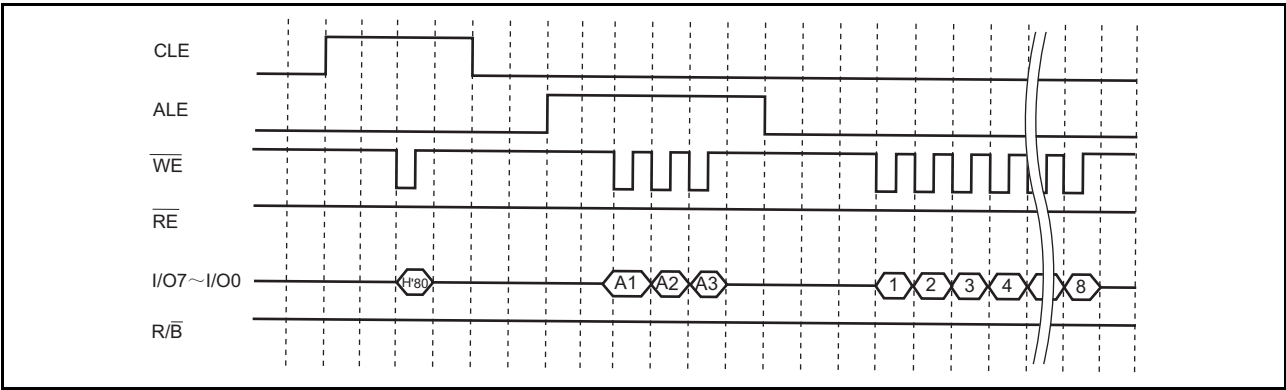


图 22.9 NAND 型闪存的写入操作时序（1）

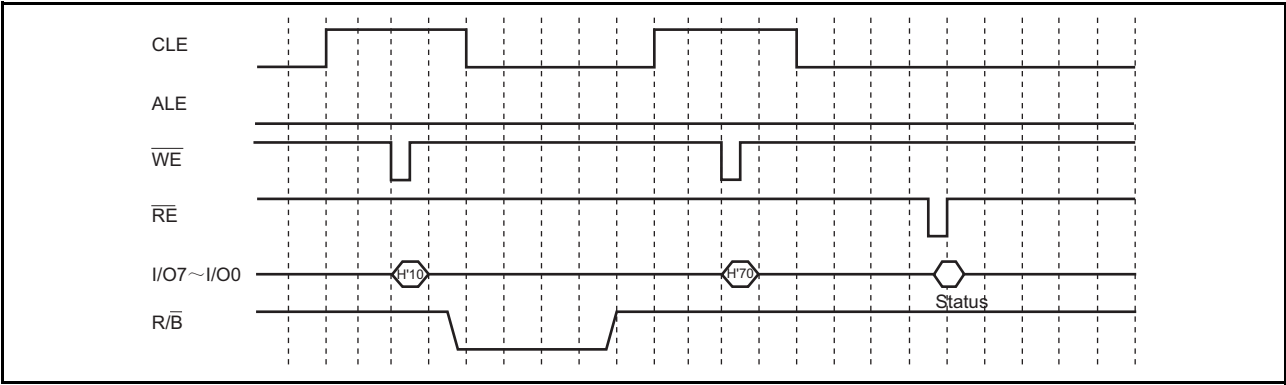


图 22.10 NAND 型闪存的写入操作编程时序（2）

(3) NAND 型闪存（2048+64 字节）的存取

NAND 型闪存（2048+64 字节）的读取例如图 22.11 所示。本例中指定第 1 命令为 H'00，第 2 命令为 H'30，地址数据长度为 4 字节，数据计数器的读取字节数为 4 字节。

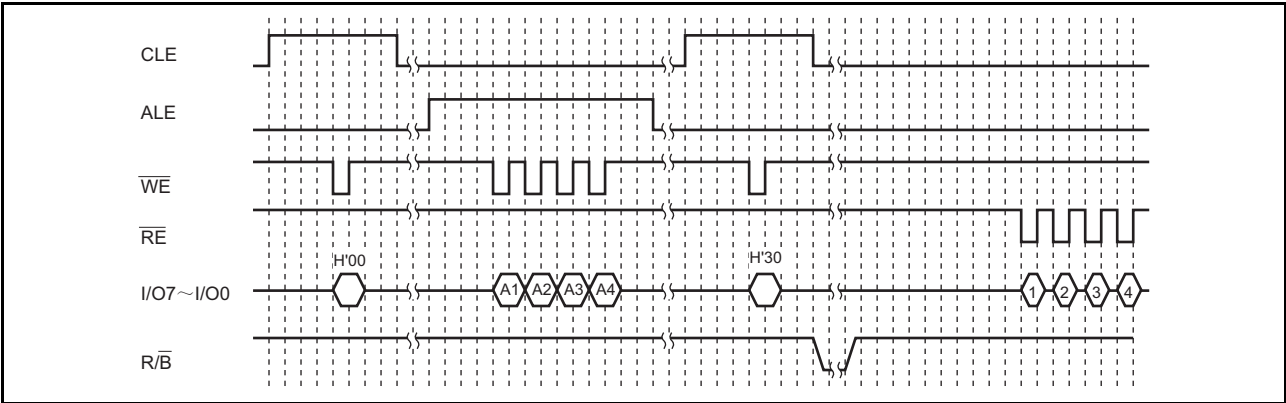


图 22.11 NAND 型闪存读取操作时序

对 NAND 型闪存 (2048+64 字节) 进行写入操作的波形如图 22.12、图 22.13 所示。

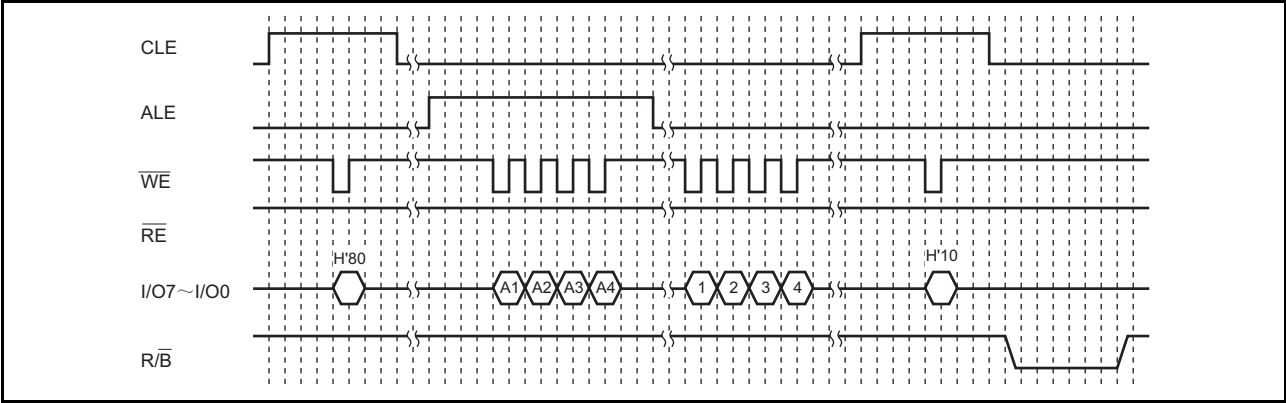


图 22.12 NAND 型闪存的写入操作时序 (1)

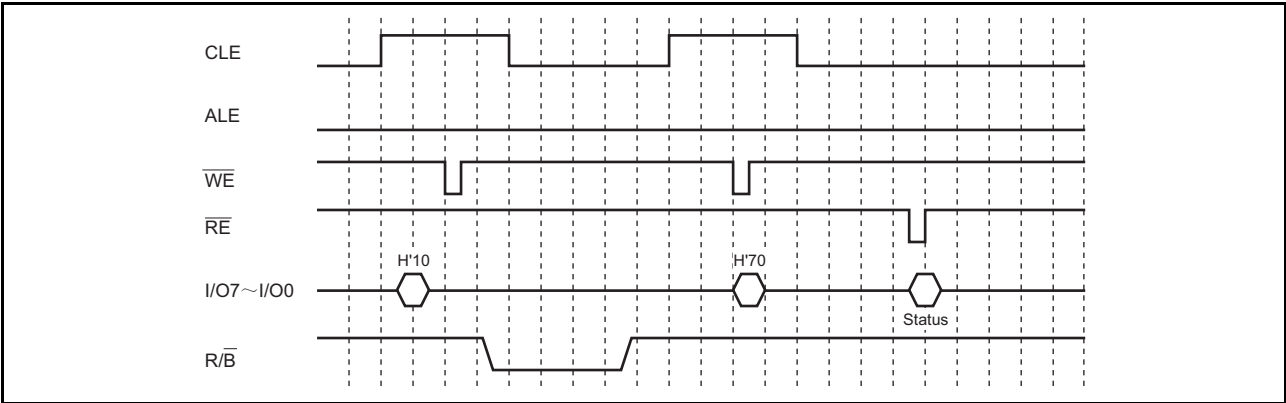


图 22.13 NAND 型闪存的写入操作时序 (2)

22.4.5 扇区存取模式

在扇区存取模式，通过指定存取的物理扇区编号，就能以扇区为单位读取 / 写入。写入时，附加 ECC；读取时，执行 ECC 错误检查（检测）处理。

512 字节的数据保存在 FLDTFIFO，16 字节的管理码保存在 FLECFIFO，因此可设定 FLINTDMACR 的 DREQ1EN、DREQ0EN，并分别执行 DMA 传送。

闪存内的扇区（数据 + 管理码）与地址空间的存储器之间的 DMA 传送关系，如图 22.14 所示。

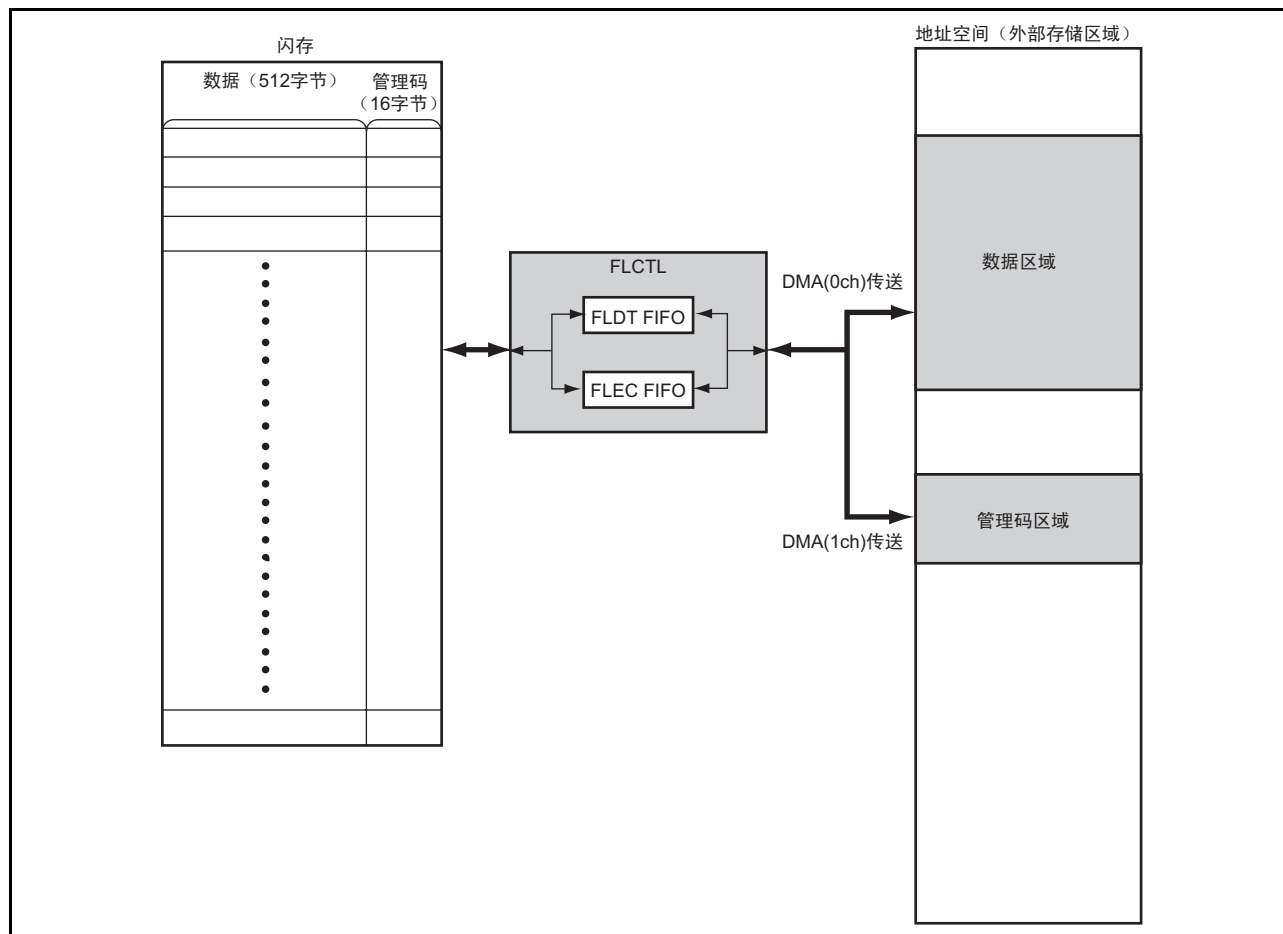


图 22.14 DMA 传送与扇区（数据、管理码）及存储器与 DMA 的关联模式图

(1) 物理扇区

AND/NAND 型闪存的物理扇区地址与闪存地址的关系，如图 22.15 所示。

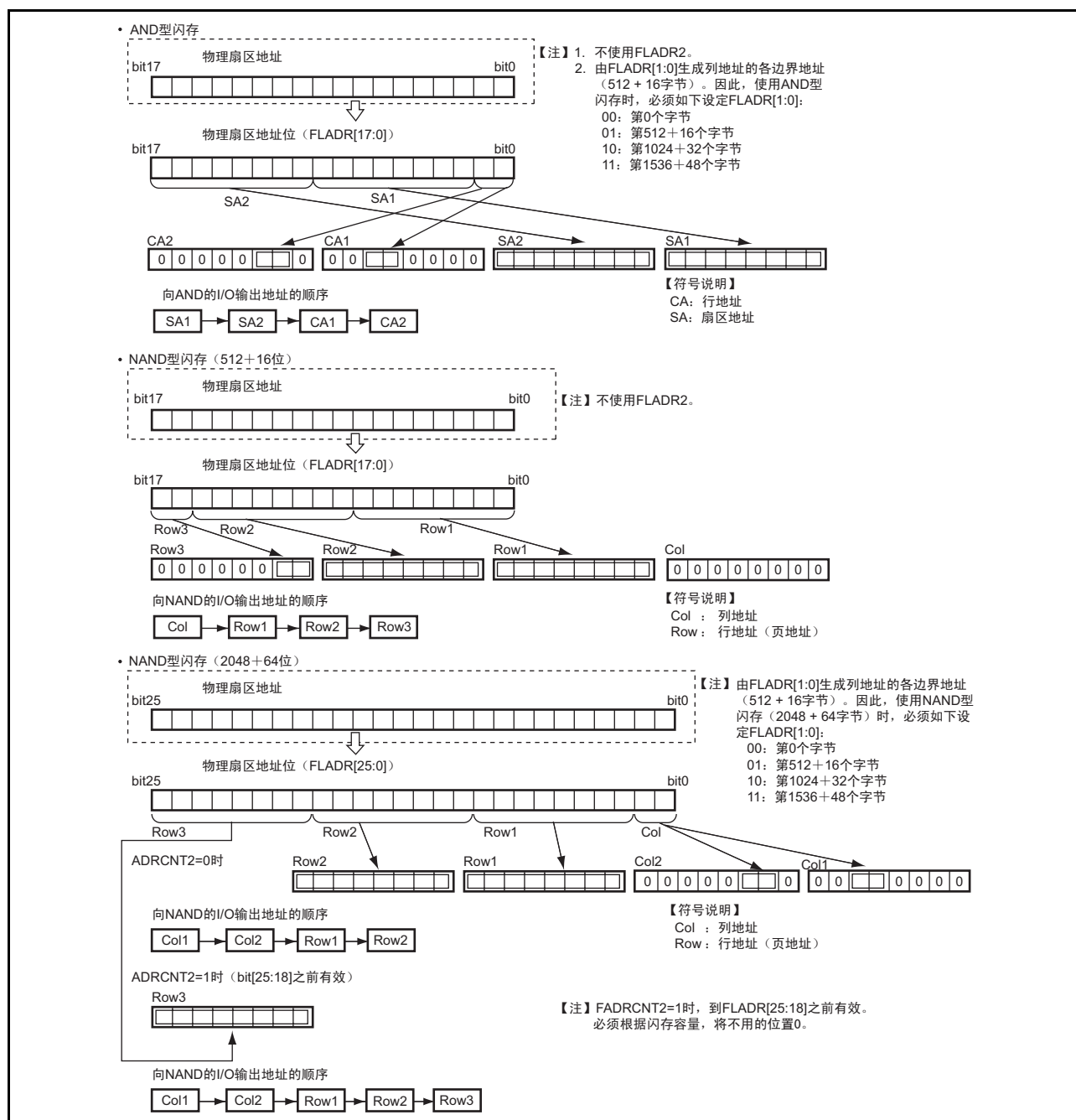


图 22.15 扇区编号与 AND/NAND 型闪存地址的扩展例

(2) 连续扇区存取

通过指定 NAND 型闪存的起始物理扇区与扇区传送次数，可连续读取 / 写入物理扇区。传送的 0 ~ 40 的逻辑扇区（中途存在不良扇区、而使物理扇区不连续）时，设定物理扇区指定寄存器和扇区传送次数指定寄存器的例子如图 22.16 所示。

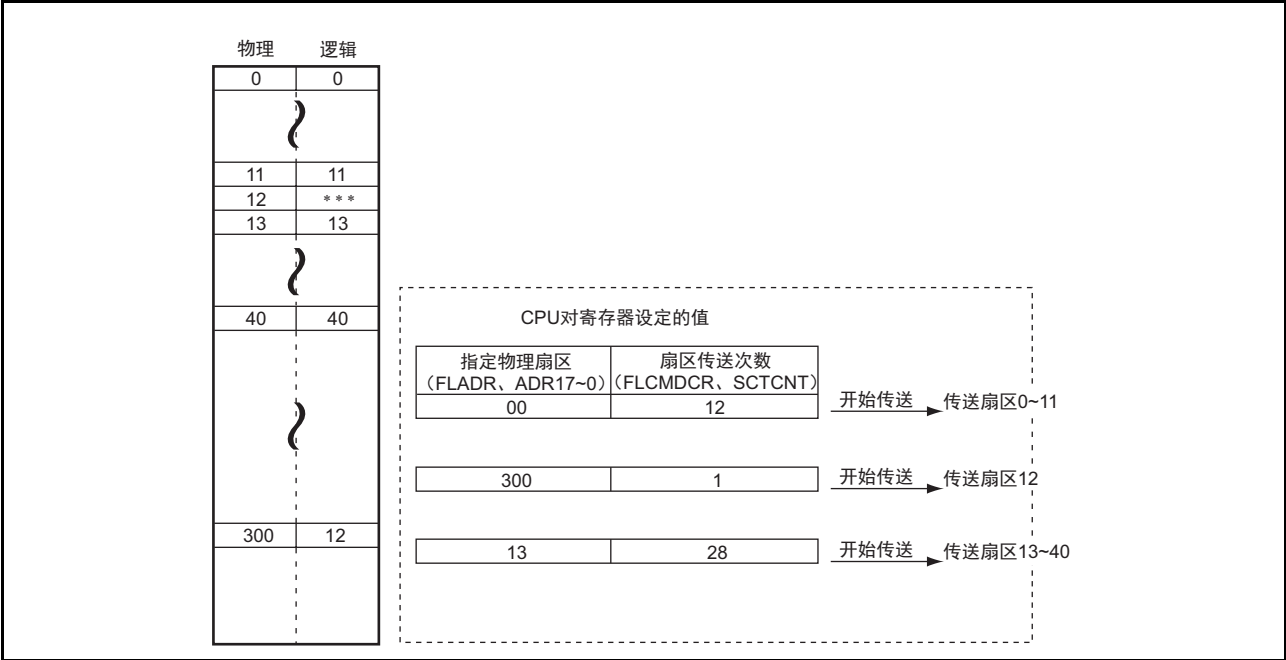


图 22.16 存在不良扇区时的存取例

22.4.6 ECC 的纠错

本 FLCTL 在扇区存取模式写入时，生成及附加 ECC 码，读取时检查 ECC 错误，但不纠错。必须由软件进行纠错。

22.4.7 状态读取

FLCTL 可读取 AND 型闪存、NAND 型闪存的状态寄存器的值。AND/NAND 闪存的状态寄存器的值从 I/O7 ~ 0 输入，保存至 FLBSYCNT 的 STAT[7:0] 位。FLBSYCNT 的 STAT[7:0] 位可从 CPU 读取。状态寄存器的值保存在 FLBSYCNT 的 STAT[7:0] 位时，如果检测出编程或擦除错误，则 FLINTDMACR 的 STERB 位置 1，允许 FLINTDMACR 的 STERINTE 位时，对 CPU 产生中断。

(1) AND 型闪存的状态读取

AND 型闪存的状态读取可通过设置输出允许信号 $\overline{OE}$ 有效 ($\overline{OE}=0$) 实现。将 FLCMDCR 的 DOSR 位置 1，并以命令存取模式或扇区存取模式编程，FLCTL 自动使 $\overline{OE}$ 信号有效，读取状态。读取 AND 型闪存状态时，从 I/O7 ~ 0 输入的状态寄存器各位的含义如表 22.3 所示。

表 22.3 AND 型闪存的状态读取

I/O	状态 (definition)	说 明
I/O7	就绪 / 忙	0: 忙状态 1: 就绪状态
I/O6	保留	—
I/O5	擦除 (Erase) 检查	0: Pass (擦除) 1: Fail (擦除失败)
I/O4	编程 (程序) 检查	0: Pass (编程) 1: Fail (编程失败)
I/O3 ~ 0	保留	—

(2) NAND 型闪存的状态读取

NAND 型闪存的状态读取可通过对 NAND 型闪存输入命令 H'70 实现。将 FLCMDCR 的 DOSR 位置 1，并以命令存取模式或扇区存取模式编程时，FLCTL 自动将命令 H'70 输入 NAND 型闪存，读取状态。读取 NAND 型闪存状态时，从 I/O7 ~ 0 输入的状态寄存器各位的含义如表 22.4 所示。

表 22.4 NAND 型闪存的状态读取

I/O	状态 (definition)	说 明
I/O7	写入保护	0: 不可写入 1: 可写入
I/O6	就绪 / 忙	0: 忙状态 1: 就绪状态
I/O5 ~ 1	保留	—
I/O0	编程 / 擦除	0: Pass (成功) 1: Fail (失败)

22.5 中断处理

FLCTL 有 6 种中断源：状态错误、就绪 / 忙超时错误、ECC 错误、传送结束、FIFO0 传送请求、FIFO1 传送请求。所有的中断源均有独立的中断标志，如果通过中断允许位允许中断，则向 CPU 发出独立的中断请求。状态错误、就绪 / 忙超时错误、ECC 错误对 CPU 使用通用的 FLSTE 中断。

表 22.5 FLCTL 的中断请求

中断源	中断标志	允许位	含义	优先顺序
FLSTE 中断	STERB	STERINTE	状态错误	高 ↑
	BTOERB	RBERINTE	就绪 / 忙超时错误	
	ECERB	ECERINTE	ECC 错误	
FLTEND 中断	TREND	TEINTE	传送结束	↓ 低
FLTRQ0 中断	TRREQF0	TRINTE0	FIFO0 传送请求	
FLTRQ1 中断	TRREQF1	TRINTE1	FIFO1 传送请求	

22.6 DMA 传送设定

FLCTL 可分别从数据区域 FLDTFIFO 及管理码区域 FLECFIFO 发出 DMA 传送请求。在各存取模式可否进行 DMA 传送，如表 22.6 所示。

表 22.6 DMA 传送设定

	扇区存取模式	命令存取模式
FLDTFIFO	可	可
FLECFIFO	可	不可

使用小端法时，如果使用 16 字节 DMA 传送，则以长字单位替换数据，因此禁止使用。
有关 DMAC 的设定，详情参阅“第 10 章 直接存储器存取控制器（DMAC）”。

第 23 章 USB2.0 主机 / 功能模块 (USB)

本模块为支持 USB2.0 标准的高速传送及全速传送，并具备 USB 主机 / 功能的模块。本模块内置 *USB 收发器，支持 USB 标准所定义的所有传送类型。

内置用于数据传送的 8K 字节缓冲存储器，最多可使用 8 个管道。此外，可对管道 1 ~ 7，分配符合执行通信的外围器件及用户系统的任意端点编号。

【注】* 使用本模块时，需先设定内部 USB 收发器。详情参阅“23.5.2 USB 收发器的设定步骤”。

23.1 特点

(1) 内置支持 USB 高速运行的主机控制器和功能控制器

- 内置USB主机控制器和功能控制器
- 通过设定寄存器，可转换USB主机控制器和功能控制器
- 支持高速传送（480Mbps）和全速传送（12Mbps）
- 内置高速/全速USB收发器（主机/功能通用）

(2) 可减少外接器件并节省安装空间

- 内置D+上拉电阻（功能运行时）
- 内置D+、D- 的下拉电阻（主机运行时）
- 内置D+、D- 终端电阻（高速运行时）
- 内置D+、D- 输出电阻（全速运行时）

(3) 支持 USB 通信的所有数据传送类型

- 控制传送
- 批量传送
- 中断传送（不支持High Bandwidth传送）
- 同步传送（不支持High Bandwidth传送）

(4) 内部总线接口

- 内置2个通道的DMA接口

(5) 管道结构

- 内置8K字节的用于USB通信的缓冲存储器
- 最多可选择8个管道（含默认控制管道）
- 可编程的管道结构
- 可给管道1～7分配任意端点编号
- 各管道可设定的传送条件如下：
管道0：控制传送专用管道、连续传送模式、256字节固定单缓冲
管道1、2：可选择批量或同步传送的管道、连续传送模式、缓冲器大小可编程（最大为2K字节，可指定双缓冲）
管道3～5：批量传送专用管道、连续传送模式、缓冲器大小可编程（最大为2K字节，可指定双缓冲）
管道6、7：中断传送专用管道、64字节固定单缓冲

【注】 使用同步 OUT 传送时，请参阅“23.5.1 使用同步 OUT 传送时的注意事项”。

(6) 选择主机控制器时的特点

- 可与外围器件进行1对1连接的通信
- 自动安排SOF、信息包的发送安排
- 同步传送、中断传送的传送间隔设定功能

(7) 选择功能控制器时的特点

- 控制传送阶段的管理功能
- 器件状态管理功能
- 对SET_ADDRESS请求的自动响应功能
- NAK响应中断功能（NRDY）

(8) 其他功能

- 通过复位信号交换自动响应自动识别高速运行或全速运行
- 通过处理计数结束传送的功能
- DMA传送结束功能
- SOF插值功能
- DMA传送结束时的Zero-Length信息包附加功能（DEZPM）
- BRDY中断事件通知时序更改功能（BFRE）
- 在DnFIFO（n=0、1）端口读取指定管道的数据后，自动清除缓冲存储器的功能（DCLRM）。
- 传送结束（SHTNAK）后，响应PID的NAK设定功能。

23.2 输入 / 输出引脚

USB 的引脚结构及使用 USB 时的引脚功能，如表 23.1 所示。

不使用本模块时，须如下处理：

- 必须外加电源。
- 必须将 DP、DM、VBUS 连接到 USBDPV_{SS}。
- 通过 5.6kΩ±1% 的电阻，将 REFRIN 连接到 USBAPV_{CC}。
- 关于 USB_X1、USB_X2，详情参阅“4.3 时钟运行模式”。

表 23.1 USB 引脚结构

区分	名称	引脚名称	输入 / 输出	功 能
USB 总线接口	USB D+ 数据	DP	输入 / 输出	USB 内部收发器 D+ 输入 / 输出引脚。 必须连接到 USB 总线的 D+ 引脚。
	USB D - 数据	DM	输入 / 输出	USB 内部收发器 D - 输入 / 输出引脚。 必须连接到 USB 总线的 D - 引脚。
VBUS 监控 输入	VBUS 输入	VBUS	输入	USB 电缆连接监控引脚。 必须连接到 USB 总线的 Vbus。可检测出 Vbus 的连接 / 断开。未连接到 USB 总线的 Vbus 时，固定为 5V。选择主机控制器功能时，也提供 5V。 【注】不可向连接的器件提供 Vbus。
基准电阻	基准输入	REFRIN	输入	基准电阻连接引脚。 必须通过 5.6kΩ±1% 电阻连接到 USBAPV _{SS} 。
时钟	用于 USB 的晶体 谐振器 / 外部时钟	USB_X1	输入	连接用于 USB 的晶体谐振器。USB_X1 引脚也可输入外部时钟。
		USB_X2	输出	
电源	收发器部 模拟引脚电源	USBAPV _{CC}	输入	引脚电源。
	收发器部 模拟引脚接地	USBAPV _{SS}	输入	引脚接地。
	收发器部 数字引脚电源	USBDPV _{CC}	输入	引脚电源。
	收发器部 数字引脚接地	USBDPV _{SS}	输入	引脚接地。
	收发器部 模拟内核电源	USBAV _{CC}	输入	内核电源。
	收发器部 模拟内核接地	USBAV _{SS}	输入	内核接地。
	收发器部 数字内核电源	USBDV _{CC}	输入	内核电源。
	收发器部 数字内核接地	USBDV _{SS}	输入	内核接地。

23.3 寄存器说明

USB 的寄存器结构如表 23.2 所示。

表 23.2 寄存器结构

寄存器名称	简称	R/W	初始值	地址	存取长度
系统结构控制寄存器	SYSCFG	R/W	H'0000	H'FFFC1C00	16
系统结构状态寄存器	SYSSTS	R	H'040X	H'FFFC1C02	16
器件状态控制寄存器	DVSTCTR	R/W	H'0000	H'FFFC1C04	16
测试模式寄存器	TESTMODE	R/W	H'0100	H'FFFC1C06	16
CPU-FIFO 总线结构寄存器	CFBCFG	R/W	H'000F	H'FFFC1C0A	16
DMA0-FIFO 总线结构寄存器	D0FBCFG	R/W	H'000F	H'FFFC1C0C	16
DMA1-FIFO 总线结构寄存器	D1FBCFG	R/W	H'000F	H'FFFC1C0E	16
CFIFO 端口寄存器	CFIFO	R/W	H'00000000	H'FFFC1C10	8、16、32
D0FIFO 端口寄存器	D0FIFO	R/W	H'00000000	H'FFFC1C14	8、16、32
D1FIFO 端口寄存器	D1FIFO	R/W	H'00000000	H'FFFC1C18	8、16、32
CFIFO 端口选择寄存器	CFIFOSEL	R/W	H'0000	H'FFFC1C1E	16
CFIFO 端口控制寄存器	CFIFOCTR	R/W	H'0000	H'FFFC1C20	16
CFIFO 端口 SIE 寄存器	CFIFOSIE	R/W	H'0000	H'FFFC1C22	16
D0FIFO 端口选择寄存器	D0FIFOSEL	R/W	H'0000	H'FFFC1C24	16
D0FIFO 端口控制寄存器	D0FIFOCTR	R/W	H'0000	H'FFFC1C26	16
D0 处理计数器寄存器	D0FIFOTRN	R/W	H'0000	H'FFFC1C28	16
D1FIFO 端口选择寄存器	D1FIFOSEL	R/W	H'0000	H'FFFC1C2A	16
D1FIFO 端口控制寄存器	D1FIFOCTR	R/W	H'0000	H'FFFC1C2C	16
D1 处理计数器寄存器	D1FIFOTRN	R/W	H'0000	H'FFFC1C2E	16
中断允许寄存器 0	INTENB0	R/W	H'0000	H'FFFC1C30	16
中断允许寄存器 1	INTENB1	R/W	H'0000	H'FFFC1C32	16
BRDY 中断允许寄存器	BRDYENB	R/W	H'0000	H'FFFC1C36	16
NRDY 中断允许寄存器	NRDYENB	R/W	H'0000	H'FFFC1C38	16
BEMP 中断允许寄存器	BEMPENB	R/W	H'0000	H'FFFC1C3A	16
中断状态寄存器 0	INTSTS0	R/W	H'00x0	H'FFFC1C40	16
中断状态寄存器 1	INTSTS1	R/W	H'0000	H'FFFC1C42	16
BRDY 中断状态寄存器	BRDYSTS	R/W	H'0000	H'FFFC1C46	16
NRDY 中断状态寄存器	NRDYSTS	R/W	H'0000	H'FFFC1C48	16
BEMP 中断状态寄存器	BEMPSTS	R/W	H'0000	H'FFFC1C4A	16
帧编号寄存器	FRMNUM	R/W	H'0000	H'FFFC1C4C	16

寄存器名称	简称	R/W	初始值	地址	存取长度
μ 帧编号寄存器	UFRMNUM	R/W	H'0000	H'FFFC1C4E	16
USB 地址寄存器	USBADDR	R	H'0000	H'FFFC1C50	16
USB 请求类型寄存器	USBREQ	R	H'0000	H'FFFC1C54	16
USB 请求值寄存器	USBVAL	R	H'0000	H'FFFC1C56	16
USB 请求变址寄存器	USBINDX	R	H'0000	H'FFFC1C58	16
USB 请求长度寄存器	USBLENG	R	H'0000	H'FFFC1C5A	16
DCP 结构寄存器	DCPCFG	R/W	H'0000	H'FFFC1C5C	16
DCPMax 包大小寄存器	DCPMAXP	R/W	H'0040	H'FFFC1C5E	16
DCP 控制寄存器	DCPCTR	R/W	H'0040	H'FFFC1C60	16
管道窗口选择寄存器	PIPESEL	R/W	H'0000	H'FFFC1C64	16
管道结构寄存器	PIPECFG	R/W	H'0000	H'FFFC1C66	16
管道缓冲器指定寄存器	PIPEBUF	R/W	H'0000	H'FFFC1C68	16
管道 Max 包大小寄存器	PEMAXP	R/W	H'0XXX	H'FFFC1C6A	16
管道周期控制寄存器	PIPEPERI	R/W	H'0000	H'FFFC1C6C	16
管道 1 控制寄存器	PIPE1CTR	R/W	H'0000	H'FFFC1C70	16
管道 2 控制寄存器	PIPE2CTR	R/W	H'0000	H'FFFC1C72	16
管道 3 控制寄存器	PIPE3CTR	R/W	H'0000	H'FFFC1C74	16
管道 4 控制寄存器	PIPE4CTR	R/W	H'0000	H'FFFC1C76	16
管道 5 控制寄存器	PIPE5CTR	R/W	H'0000	H'FFFC1C78	16
管道 6 控制寄存器	PIPE6CTR	R/W	H'0000	H'FFFC1C7A	16
管道 7 控制寄存器	PIPE7CTR	R/W	H'0000	H'FFFC1C7C	16
USB AC 特性转换寄存器	USBACSWR	R/W	H'00000000	H'FFFC1C84	32

23.3.1 系统结构控制寄存器 (SYSCFG)

SYSCFG 是设置高速运行允许、选择主机控制器功能或功能控制器功能、控制 DP、DM 引脚、控制全速接收器、并控制本模块软件复位的寄存器。

由上电复位初始化本寄存器。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	HSE	DCFM	DMRPD	DPRPU	—	FSRPC	—	USBE
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R	R/W	R	R/W

位	位名称	初始值	R/W	说 明
15 ~ 8	—	均为 0	R	保留位 读取值、写入值总是为 0。
7	HSE	0	R/W	高速运行允许 0: 禁止高速运行 (仅允许全速运行) 1: 允许高速运行 (本模块检测)
6	DCFM	0	R/W	控制器功能选择 选择主机控制器功能或功能控制器功能。 0: 选择功能控制器功能 1: 选择主机控制器功能
5	DMRPD	0	R/W	D - 线电阻控制
4	DPRPU	0	R/W	D + 线电阻控制 设定 D - , D + 线的电阻。设定本位前, 必须设定 HSE 位以及 DCFM 位。 00: D - 为开路, D + 为开路 01: D - 为开路, D + 为上拉 10: D - 为下拉, D + 为下拉 11: D - 为下拉, D + 为上拉
3	—	0	R	保留位 读取值、写入值总是为 0。
2	FSRPC	0	R/W	全速接收器运行允许 允许全速接收器的运行。 0: 硬件控制全速接收器的运行 1: 由软件允许全速接收器的运行
1	—	0	R	保留位 读取值、写入值总是为 0。
0	USBE	0	R/W	USB 块运行允许 通过本位复位本模块的软件。设定 USBE=0 时, 将软件复位初始化对象寄存器复位为初始设定值。设定 USBE=0 时, 不写入软件复位初始化对象寄存器及位。软件复位后, 必须设定 USBE=1, 允许本模块运行。 0: 禁止 USB 块运行 (软件复位) 1: 允许 USB 块运行

23.3.2 系统结构状态寄存器（SYSSTS）

SYSSTS 监控 USB 数据总线的线状态（D+ 及 D- 线）。

由上电复位、软件复位及 USB 复位初始化本寄存器。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	SOFEN	—	—	—	LNST[1:0]	
初始值:	0	0	0	0	0	1	0	0	0	0	0	0	0	0	*	*
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名称	初始值	R/W	说 明
15 ~ 11	—	均为 0	R	保留位 读取值、写入值总是为 0。
10	—	1	R	保留位 读取值不定，写入无效。
9 ~ 6	—	均为 0	R	保留位 读取值、写入值总是为 0。
5	SOFEN	0	R	SOF 发行允许 表示主机模式运行时，软件对 DVSTCTR 寄存器的 UACT 位写入后，本模块的内部电路为可发行 SOF 或 SOF 发行停止状态。 0: 不对 USB 端口发行 SOF 1: 对 USB 端口发行 SOF
4 ~ 2	—	均为 0	R	保留位 读取值、写入值总是为 0。
1、0	LNST[1:0]	—	R	USB 数据线状态 USB 数据总线的线状态如表 23.3 所示。通过设定本位，监控 USB 数据总线的线状态（D+ 线及 D- 线）。 用全速接收器确认线状态。通过提供 USBCLK，本模块自动进行全速接收器的控制。但是如果设定 SYSCFG 寄存器的 FSRPC 位，即使不提供 USBCLK，也可由软件允许全速接收器的控制。上电复位后，在提供 USBCLK 前确认 D+ 及 D- 线的状态时，将 FSRPC 位置 1。提供 USBCLK 后，无需软件设定。

【注】 * 取决于 D+ 及 D- 线的状态。

表 23.3 USB 数据总线的线状态表

LNST1	LNST0	全速运行时	高速运行时	Chirp 运行时
0	0	SE0	Squelch	Squelch
0	1	J-State	not Squelch	Chirp J
1	0	K-State	Invalid	Chirp K
1	1	SE1	Invalid	Invalid

【符号说明】

Chirp: 在高速运行允许状态（SYSCFG.HSE=1），正在执行复位信号交换协议（RHSP）

Squelch: SE0 或空闲状态

not Squelch: 高速 J-State 或高速 K-State

Chirp J: Chirp J-State

Chirp K: Chirp K-State

Invalid: 无效

23.3.3 器件状态控制寄存器 (DVSTCTR)

DVSTCTR 控制及确认 USB 数据总线状态。

由上电复位初始化本寄存器。由软件复位初始化除 WKUP 位之外的位，WKUP 位变为不定；通过 USB 总线复位，初始化 WKUP 位，RESUME 位变为不定。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	JACKKEY0	—	—	JACKKEY1	—	—	—	WKUP	RWUPE	USBRST	RESUME	UACT	—	—	RHST[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R

位	位名称	初始值	R/W	说 明
15	JACKKEY0	0	R/W	USBAC 键 0 为了写入测试寄存器 HOSTPCC 位，必须设定此位。详情参阅 “23.5.2 USB 收发器的设定步骤”。
14、13	—	均为 0	R	保留位 读取值、写入值总是为 0。
12	JACKKEY1	0	R/W	USBAC 键 1 为了写入测试寄存器 HOSTPCC 位，必须设定此位。详情参阅 “23.5.2 USB 收发器的设定步骤”。
11 ~ 9	—	均为 0	R	保留位 读取值、写入值总是为 0。
8	WKUP	0	R/W	唤醒输出 本位控制向 USB 总线输出远程唤醒信号。本模块管理远程唤醒信号的输出时间。本位置 1 时，本模块输出 10ms 的 K-State 后，将本位置 0。 根据 USB 规格，发送远程唤醒信号前，至少需要 5ms 的 USB 总线空闲状态。因此，即使本模块检测出挂起状态后立即对本位写入 1，也需等待 2ms 后输出 K-State。 0: 不输出 1: 输出远程唤醒信号 【注】 除器件状态为挂起 (INTSTS0.DVSQ=1xx) 并且 USB 主机已允许远程唤醒之外，不得向本位写入 1。本位设定为 1 时，即使为挂起也不得停止 USBCLK。
7	RWUPE	0	R/W	唤醒检测允许 本位设定为 1，检测出远程唤醒信号时，向下行端口输出恢复信号。此时，本模块将 RESUME 位置 1。 0: 禁止下行端口唤醒 1: 允许下行端口唤醒 【注】 本位设定为 1 时，即使为挂起也不得停止 USBCLK。

位	位名称	初始值	R/W	说 明
6	USBRST	0	R/W	总线复位输出 本位设定为 1，输出 USB 总线复位信号。USB 总线复位信号输出时间需由软件管理。USB 总线复位时间结束后，将本位设定为 0。 0：停止 USB 总线复位信号输出 1：输出 USB 总线复位信号
5	RESUME	0	R/W	输出恢复 本位设定为 1，向 USB 总线输出恢复信号。 0：停止恢复信号输出 1：输出恢复信号
4	UACT	0	R/W	USB 总线允许 控制向 USB 总线发送 SOF 或 μ SOF 包。本模块管理 SOF 包的发送间隔。对本位写入 0 时，发送下一个 SOF 后进入总线空闲状态。 0：禁止下行端口（禁止 SOF / μ SOF 发送） 1：允许下行端口（允许 SOF / μ SOF 发送）
3、2	—	均为 0	R	保留位 读取值、写入值总是为 0。
1、0	RHST[1:0]	00	R	复位信号交换 参照本位可确认与已连接的主机控制器的通信速度（通信位速率）。 高速运行设定为禁止状态（SYSCFG.HSE=0）时，本模块不执行复位信号交换而全速运行。高速运行设定为允许状态时，本模块执行复位信号交换（执行过程中 RHST=01），并将结果反映至本位（高速运行时 RHST=11 或全速运行时 RHST=10）。 选择主机控制器功能时，解除 USB 总线复位后，反映到本位的结果是以下两个时序。一个是在全速模式时，由 USB 总线复位输出，从 SE0-State 变为 J-State 的时序。另一个是在高速模式时，通过复位信号交换将终端电阻转换为高速模式的时序。但需注意：在任意一个时序，所连接的外围器件支持全速运行时，反映结果前需等待时间。 00：通信速度不定 01：正在处理复位信号交换 10：确定全速运行 11：确定高速运行 【注】 USB 总线复位结束后（设定 USBRST=0 后），经过足够等待时间也未确定 RHST 时，在 USB 总线复位过程中，USB 电缆可能被切断。此时，必须通过 LNST 位确认 USB 总线状态。

【注】 选择功能控制器功能时，RWUPE、USBRST、RESUME 及 UACT 位的值均设定为 0。
选择主机控制器功能时，WKUP 位的值必须设定为 0。

23.3.4 测试模式寄存器（TESTMODE）

TESTMODE 控制高速运行时的 USB 测试信号输出及本模块内置的 USB 收发器。

由上电复位初始化本寄存器。软件复位时 UTST 位被初始化。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HOST PCC	—	—	—	—	—	—	—	—	—	—	—	UTST[3:0]			
初始值:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15	HOSTPCC	0	R/W	断开检测功率的转换 设定 USB 收发器。* 要设定本位，需设定器件控制寄存器的 UACKEY0、UACKEY1 位。
14 ~ 9	—	均为 0	R	保留位 读取值、写入值总是为 0。
8	—	1	R	保留位 本位的读取值取决于 UACKEY0、UACKEY1 位。 UACKEY0=0、UACKEY1=0 时，读取值总是为 1，写入值无效。 UACKEY0=1、UACKEY1=1 时，读取值总是为 0、此时，写入值也总是为 0。
7 ~ 4	—	均为 0	R	保留位 读取值、写入值总是为 0。
3 ~ 0	UTST[3:0]	0000	R/W	测试模式 本模块的测试模式运行如表 23.4 所示。由本位控制高速运行时 USB 测试信号输出。 [选择主机控制器功能时] 选择主机控制器功能时，写入 DCFM=1、DRPD=1 后，可设定本位。 通过 写入本位，本模块高速终端运行。 本位设定步骤如下： (1) 上电复位 (2) DCFM=1，DPRD=1（无需设定 HSE=1） (3) USBE=1 (4) 对本位设定测试内容对应的值。 本位更改步骤如下： (1) （在上述（4）的状态）USBE=0 (2) USBE=1 (3) 对本位设定测试内容对应值 【注】 设定 Test_SE0_NAK (1011) 时，即使设定 UACT=1，本模块也不输出 SOF 包。设定 Test_Force_Enable (1101) 时，如果 UACT=1，本模块输出 SOF 包。 设定 UTST 位时，设定所有管道的 PID 位为 NAK。设定测试模式后，执行通常的 USB 通信时，必须执行上电复位。 [选择功能控制器功能时] 选择功能控制器功能时，按照高速通信时 USB 主机的 SetFeature 请求写入本位。 【注】 本位设定为 0001 ~ 0100 时，本模块不会转移至挂起状态。

【注】 * 详情参阅“23.5.2 USB 收发器的设定步骤”。

表 23.4 测试模式运行表

测试模式	UTST 位设定	
	选择功能控制器功能时	选择主机控制器功能时
正常运行	0000	0000
Test_J	0001	1001
Test_K	0010	1010
Test_SE0_NAK	0011	1011
Test_Packet	0100	1100
Reserved	0101 ~ 0111	1101 ~ 1111

23.3.5 FIFO 端口结构寄存器 (CFBCFG、D0FBCFG、D1FBCFG)

本寄存器控制 FIFO 端口的存取。FIFO 端口包括 CPU-FIFO、DMA0-FIFO 及 DMA1-FIFO 等 3 个。各由各端口对应的结构寄存器控制存取。

由上电复位初始化本寄存器。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TENDE	FEND	—	—	—	—	FWAIT[3:0]			
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
R/W:	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15 ~ 10	—	均为 0	R	保留位 读取值、写入值总是为 0。
9	TENDE	0	R/W	DMA 传送结束采样允许 控制 DMA 传送结束时, 是否接收从直接存储器存取控制器 (DMAC) 发送的 DMA 传送结束信号。 0: 不采样 DMA 传送结束信号 1: 采样 DMA 传送结束信号
8	FEND	0	R/W	FIFO 端口字节序 设定在 FIFO 端口存取时的字节序。字节序运行如表 23.5 ~ 表 23.7 所示。本 LSI 为大端法运行。发送 / 接收字节序不同的数据时, 必须设定本位。
7 ~ 4	—	均为 0	R	保留位 读取值、写入值总是为 0。
3 ~ 0	FWAIT[3:0]	1111	R/W	指定 FIFO 端口存取等待 指定相应 FIFO 端口的存取等待数。FIFO 端口的最小存取周期为 2 个周期。 0000: 0 个等待 (2 个存取周期) : 0010: 2 个等待 (4 个存取周期) : 0100: 4 个等待 (6 个存取周期) : 1111: 15 个等待 (17 个存取周期)

【注】 TEND 位只存在于 D0FBCFG 及 D1FBCFG。

表 23.5 字节序运行 (32 位宽度存取时)

FEND	bit31 ~ bit24	bit23 ~ bit16	bit15 ~ bit8	bit7 ~ bit0
0	地址 N+0	地址 N+1	地址 N+2	地址 N+3
1	地址 N+3	地址 N+2	地址 N+1	地址 N+0

表 23.6 字节序运行 (16 位宽度存取时)

FEND	bit31 ~ bit24	bit23 ~ bit16	bit15 ~ bit8	bit7 ~ bit0
0	偶数地址	奇数地址	写入: 无效 读取: 禁止 *	
1	写入: 无效 读取: 禁止 *		奇数地址	偶数地址

【注】 * 禁止以字为单位读取无效寄存器。

表 23.7 字节序运行 (8 位宽度存取时)

FEND	bit31 ~ bit24	bit23 ~ bit16	bit15 ~ bit8	bit7 ~ bit0
0	写入: 有效 读取: 有效	写入: 无效 读取: 无效 *		
1	写入: 无效 读取: 禁止 *			写入: 有效 读取: 有效

【注】 * 禁止以字节为单位读取无效寄存器。

23.3.6 FIFO 端口寄存器 (CFIFO、D0FIFO、D1FIFO)

本寄存器为读取 / 写入 FIFO 缓冲存储器数据的端口寄存器。

FIFO 端口包括 CFIFO、D0FIFO 及 D1FIFO 等 3 个。各 FIFO 端口与读取 / 写入至缓冲存储器的端口寄存器、选择分配到 FIFO 端口的管道的选择寄存器、控制寄存器及端口功能专用寄存器 (CFIFO 端口专用的 SIE 寄存器和 DnFIFO 端口专用的处理计数器寄存器) 构成的寄存器群相关联。

由上电复位及软件复位初始化本寄存器。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	FIFOPORT[31:16]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FIFOPORT[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
31 ~ 0	FIFOPORT [31:0]	均为 0	R/W	FIFO 端口 从缓冲存储器读取接收数据或向缓冲存储器写入发送数据。

- 【注】
- 只有 DCP 为 CFIFO 端口时，才可存取缓冲存储器。
 - 只有 D0FIFO 端口及 D1FIFO 端口才可通过 DMA 传送存取缓冲存储器。
 - 通过 CPU 存取 DnFIFO 端口，也要按照 DnFIFO 端口的功能及限制 (使用处理计数器等) 进行。
 - 使用 FIFO 端口固有的功能时，不可使用选择管道 (使用处理计数器等)。
 - 构成 FIFO 端口的寄存器群，不影响其他 FIFO 端口。
 - 不得将相同管道分配给不同的 FIFO 端口。
 - 缓冲存储器的存取权可能在 CPU 或 SIE。缓冲存储器的存取权在 SIE 时，无法由 CPU 正确存取。
 - 通过 FIFO 端口选择的管道结构不得更改。

23.3.7 FIFO 端口选择寄存器（CFIFOSEL、D0FIFOSEL、D1FIFOSEL）

选择分配至 FIFO 端口的管道，控制对相应端口的存取。

不可对 CFIFOSEL、D0FIFOSEL 及 D1FIFOSEL 寄存器的 CURPIPE 位指定相同管道。D0FIFOSEL 及 D1FIFOSEL 寄存器的 CURPIPE 位设定为 B'000 时，无需指定管道。

在 DMA 传送允许状态时，不可更改管道编号。

由上电复位及软件复位初始化本寄存器。

(1) CFIFOSEL

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RCNT	REW	—	—	MBW[1:0]	—	—	—	—	ISEL	—	—	—	—	—	CURPIPE[2:0]
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W*1	R	R	R/W	R/W	R	R	R	R	R/W	R	R	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15	RCNT	0	R/W	读取计数模式 0: 通过读取所有接收数据，清除 DTLN 位 1: 通过读取接收数据，DTLN 位递减计数
14	REW	0	R/W*1	缓冲器指针回绕 0: 缓冲器指针不回绕 1: 缓冲器指针回绕
13、12	—	均为 0	R	保留位 读取值、写入值总是为 0。
11、10	MBW[1:0]	00	R/W	FIFO 端口存取位宽度 00: 8 位宽度 01: 16 位宽度 10: 32 位宽度 11: 禁止设定 选择 CURPIPE 为缓冲存储器读取方向时，本位设定请按照以下步骤之一执行。 • 与设定 CURPIPE 位同时写入。 • 设定 DCP（CURPIPE=000）时，与设定 ISEL 位同时写入。 详情参阅“23.4.4 缓冲存储器”。 【注】一旦开始缓冲存储器的读取处理，在所有数据读取结束前，不可更改 FIFO 端口存取位宽度。对缓冲存储器写入时，不能从 8 位宽度转换为 16 位宽度 / 32 位宽度，也不能从 16 位宽度转换为 32 位宽度。
9 ~ 6	—	均为 0	R	保留位 读取值、写入值总是为 0。

位	位名称	初始值	R/W	说 明
5	ISEL	0	R/W	DCP 选择时的 FIFO 端口存取方向 0: 选择缓冲存储器读取 1: 选择缓冲存储器写入 仅通过 CURPIPE 位选择 DCP 时, 本位有效。 请按以下步骤之一设定本位: • CURPIPE 位的 DCP 设定 (CURPIPE=000) 和 ISEL 位的设定同时写入 • CURPIPE 位的 DCP 设定 (CURPIPE=000) 后, 等待 200ns 再设定 ISEL 位 详情参阅 “23.4.4 缓冲存储器”。
4、3	—	均为 0	R	保留位 读取值、写入值总是为 0。
2 ~ 0	CURPIPE [2:0]	000	R/W	指定 FIFO 端口存取管道 000: DCP 001: 管道 1 010: 管道 2 011: 管道 3 100: 管道 4 101: 管道 5 110: 管道 6 111: 管道 7

【注】 *1 仅读取 0 后写入 1 有效。

*2 在连续更改 ISEL 位或 CURPIPE 位时, 至少需要 120ns+5 个总线周期的存取周期。

(2) D0FIFOSEL、D1FIFOSEL

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RCNT	REW	DCLRM	DREQE	MBW[1:0]	TRENB	TRCLR	DEZPM	—	—	—	—	—	CURPIPE[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W* ¹	R/W	R/W	R/W	R/W	R/W	R/W* ¹	R/W	R	R	R	R	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15	RCNT	0	R/W	读取计数模式 0: 通过读取所有接收数据, 清除 DTLN 位 1: 通过读取接收数据, DTLN 位递减计数
14	REW	0	R/W* ¹	缓冲器指针回绕 0: 缓冲器指针不回绕 1: 缓冲器指针回绕
13	DCLRM	0	R/W	读取指定管道的数据后, 为自动缓冲存储器清除模式 仅在设定 CURPIPE 位指定的管道为接收方向 (缓冲存储器读取) 时, 本位有效。 0: 禁止自动缓冲清除模式 1: 允许自动缓冲清除模式

位	位名称	初始值	R/W	说 明
12	DREQE	0	R/W	DMA 传送请求允许 0: 禁止 DMA 传送请求 1: 允许 DMA 传送请求
11、10	MBW[1:0]	00	R/W	FIFO 端口存取位宽度 00: 8 位宽度 01: 16 位宽度 10: 32 位宽度 11: 禁止设定 选择 CURPIPE 为缓冲存储器读取方向时, 必须同时设定本位与 CURPIPE 位。 详情参阅“23.4.4 缓冲存储器”。 【注】一旦开始缓冲存储器的读取处理, 在所有数据读取结束前, 不可更改 FIFO 端口存取位宽度。写入缓冲存储器时, 不能从 8 位宽度转换为 16 位宽度 /32 位宽度, 也不能从 16 位宽度转换为 32 位宽度。
9	TRENB	0	R/W	处理计数器允许 仅在设定 CURPIPE 位指定的管道为接收方向 (缓冲存储器读取) 时, 本位有效。 0: 处理计数器功能无效 1: 处理计数器功能有效
8	TRCLR	0	R/W*1	处理计数器清除 仅在设定 CURPIPE 位指定的管道为接收方向 (缓冲存储器读取) 时, 本位有效。 0: 无效 1: 清除当前计数器
7	DEZPM	0	R/W	Zero-Length 包附加模式 仅在设定 CURPIPE 位指定的管道为发送方向 (缓冲存储器写入) 时, 本位有效。 0: 无付加 1: 有付加
6 ~ 3	—	均为 0	R	保留位 读取值、写入值总是为 0。
2 ~ 0	CURPIPE [2:0]	000	R/W	指定 FIFO 端口存取管道 *2 000: 无指定 001: 管道 1 010: 管道 2 011: 管道 3 100: 管道 4 101: 管道 5 110: 管道 6 111: 管道 7

【注】 *1 仅读取 0 后写入 1 有效。

*2 在连续更改 CURPIPE 位时, 至少需要 120ns+5 个总线周期的存取周期。

23.3.8 FIFO 端口控制寄存器 (CFIFOCTR、D0FIFOCTR、D1FIFOCTR)

本寄存器是设定缓冲存储器的写入是否结束、是否清除 CPU 侧的缓冲器及可否存取 FIFO 端口的寄存器。本寄存器中对应各 FIFO 端口的有 CFIFOCTR、D0FIFOCTR 及 D1FIFOCTR。

由上电复位及软件复位初始化本寄存器。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BVAL	BCLR	FRDY	—	DTLN[11:0]											
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*1	R/W*2	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名称	初始值	R/W	说 明
15	BVAL	0	R/W*1	缓冲存储器有效标志 在数据包发送方向时 (缓冲存储器写入时), 对本位写入 1 有效。为接收方向时, 必须设定 BVAL=0。 0: 无效 1: 写入结束
14	BCLR	0	R/W*2	CPU 缓冲器清除 *3 通过管道结构设定为管道无效状态 (PID=NAK) 后, 由本位清除缓冲器。 0: 无效 1: 清除 CPU 侧的缓冲存储器
13	FRDY	0	R	FIFO 端口就绪 确认本位时, 在选择管道后至少需要 450ns 存取周期。 0: 不可存取 FIFO 端口 1: 可存取 FIFO 端口
12	—	0	R	保留位 读取值、写入值总是为 0。
11 ~ 0	DTLN [11:0]	H'000	R	接收数据长度 *4 可确认接收数据长度。

【注】 *1 仅写入 1 有效。

*2 仅读取 0 后写入 1 有效。

*3 选择除 DCP 之外的管道时, BCLR 位对 CPU 侧的缓冲存储器有效。必须在确认 FRDY=1 后, 设定 BCLR=1。选择 DCP 为管道时, 也清除 SIE 侧的缓冲存储器。此时, 无需确认 FRDY=1。

*4 DTLN 位对 CPU 侧的缓冲存储器有效。必须确认 FRDY=1 后, 再参照 DTLN 位。

23.3.9 CFIFO 端口 SIE 寄存器 (CFIFOSIE)

本寄存器是控制 CFIFO 端口的 SIE 功能，转换 CPU 侧与 SIE 侧的存取权、清除 SIE 侧的缓冲存储器以及确认 SIE 侧的缓冲器忙状态的寄存器。选择 DCP 时，本寄存器不起作用。

由上电复位及软件复位初始化本寄存器。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TGL	SCLR	SBUSY	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*	R/W*	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名称	初始值	R/W	说 明
15	TGL	0	R/W*	转换存取权 本位的功能是将 SIE 侧的缓冲存储器设置到 CPU 侧。设定 PID=NAK，并由 SBUSY 位确认 SIE 未存取缓冲器 (SBUSY=0) 后，写入 TGL 位 (交替操作)。本位仅对设定为接收方向 (缓冲存储器读取) 的管道有效。 0: 无效 1: 转换存取权
14	SCLR	0	R/W*	SIE 缓冲器清除 本位的功能是清除 SIE 侧的缓冲存储器。设定 PID=NAK，并由 SBUSY 位确认 SIE 侧未存取缓冲器 (SBUSY=0) 后，清除缓冲器。本位仅对设定为发送方向 (缓冲存储器写入) 的管道有效。 0: 无效 1: 清除 SIE 侧的缓冲存储器
13	SBUSY	0	R	SIE 缓冲器忙 0: SIE 未存取状态 1: SIE 存取状态
12 ~ 0	—	均为 0	R	保留位 读取值、写入值总是为 0。

【注】 * 仅读取 0 后写入 1 有效。

23.3.10 处理计数器寄存器 (D0FIFOTRN、D1FIFOTRN)

设定 DMA 传送的处理次数及读取处理次数。

由上电复位及软件复位初始化本寄存器。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TRNCNT[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15 ~ 0	TRNCNT [15:0]	H'0000	R/W	处理计数器 本位仅在从缓冲存储器读取数据时有效。 仅在 DnFIFOSEL 寄存器的 TRENb 位为 1 时, 才可读取计数时的处理次数。TRENb=0 时, 可读取已设定的处理次数。 W: 设定 DMA 传送的处理次数 R: 读取处理次数

23.3.11 中断允许寄存器 0 (INTENB0)

本寄存器指定各中断屏蔽、器件状态转移中断源屏蔽 (URST 位、SADR 位、SCFG 位及 SUSP 位) 及控制传送阶段中断源屏蔽 (WDST 位、RDST 位、CMPL 位及 SERR)。

由上电复位或软件复位初始化本寄存器。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VBSE	RSME	SOFE	DVSE	CTRE	BEMPE	NRDYE	BRDYE	URST	SADR	SCFG	SUSP	WDST	RDST	CMPL	SERR
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15	VBSE	0	R/W	VBUS 中断允许 0: 禁止中断输出 1: 允许中断输出
14	RSME	0	R/W	恢复中断允许 0: 禁止中断输出 1: 允许中断输出
13	SOFE	0	R/W	帧编号更新中断允许 0: 禁止中断输出 1: 允许中断输出
12	DVSE	0	R/W	器件状态转移中断允许 0: 禁止中断输出 1: 允许中断输出
11	CTRE	0	R/W	控制传送阶段转移中断允许 0: 禁止中断输出 1: 允许中断输出

位	位名称	初始值	R/W	说 明
10	BEMPE	0	R/W	缓冲器空中断允许 0: 禁止中断输出 1: 允许中断输出
9	NRDYE	0	R/W	缓冲器未就绪响应中断允许 0: 禁止中断输出 1: 允许中断输出
8	BRDYE	0	R/W	缓冲器就绪中断允许 0: 禁止中断输出 1: 允许中断输出
7	URST	0	R/W	默认状态转移通知允许 0: 默认状态转移时, 禁止 DVST 中断 1: 默认状态转移时, 允许 DVST 中断
6	SADR	0	R/W	地址状态转移通知允许 0: 地址状态转移时, 禁止 DVST 中断 1: 地址状态转移时, 允许 DVST 中断
5	SCFG	0	R/W	结构状态转移通知允许 0: 结构状态转移时, 禁止 DVST 中断 1: 结构状态转移时, 允许 DVST 中断
4	SUSP	0	R/W	挂起状态转移通知允许 0: 挂起状态转移时, 禁止 DVST 中断 1: 挂起状态转移时, 允许 DVST 中断
3	WDST	0	R/W	控制写入阶段转移通知允许 0: 控制写入阶段转移时, 禁止 CTRT 中断 1: 控制写入阶段转移时, 允许 CTRT 中断
2	RDST	0	R/W	控制读取阶段转移通知允许 0: 控制读取阶段转移时, 禁止 CTRT 中断 1: 控制读取阶段转移时, 允许 CTRT 中断
1	CMPL	0	R/W	控制传送结束通知允许 0: 检测出控制传送结束时, 禁止 CTRT 中断 1: 检测出控制传送结束时, 允许 CTRT 中断
0	SERR	0	R/W	控制传送顺序错误通知允许 0: 检测出控制传送顺序错误时, 禁止 CTRT 中断 1: 检测出控制传送顺序错误时, 允许 CTRT 中断

【注】 从清除各中断状态到允许 / 禁止相应的中断, 必须空出至少 80ns 的间隔。

23.3.12 中断允许寄存器 1 (INTENB1)

本寄存器指定各中断屏蔽，并控制各管道的 BRDY 中断状态清除时序。

由上电复位初始化本寄存器。此外，由软件复位初始化除 BRDYM 之外的位。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	BCHGE	—	DTCHE	—	—	—	—	—	—	SIGNE	SACKE	—	BRDYM	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R/W	R	R	R	R	R	R	R/W	R/W	R	R/W	R	R

位	位名称	初始值	R/W	说 明
15	—	0	R	保留位 读取值、写入值总是为 0。
14	BCHGE	0	R/W	USB 总线变化中断允许 0: 禁止中断输出 1: 允许中断输出
13	—	0	R	保留位 读取值、写入值总是为 0。
12	DTCHE	0	R/W	全速运行时断开检测中断允许 仅在选择主机控制器功能且全速运行时，由本位检测断开有效。高速运行时，必须由软件检测断开（检测功能控制器无响应等）。 0: 禁止中断输出 1: 允许中断输出 【注】 复位信号交换后，判断为高速运行 (RHST=11) 时，高速通信时，必须设定 DTCHE=0。
11 ~ 6	—	均为 0	R	保留位 读取值、写入值总是为 0。
5	SIGNE	0	R/W	设置处理错误中断允许 0: 禁止中断输出 1: 允许中断输出
4	SACKE	0	R/W	设置处理正常响应中断允许 0: 禁止中断输出 1: 允许中断输出
3	—	0	R	保留位 读取值、写入值总是为 0。
2	BRDYM	0	R/W	控制各管道的 BRDY 中断状态清除时序 0: 软件清除状态 1: 通过读取 FIFO 缓冲器或对 FIFO 缓冲器写入，本模块清除状态
1、0	—	均为 0	R	保留位 读取值、写入值总是为 0。

23.3.13 BRDY 中断允许寄存器 (BRDYENB)

本寄存器允许各管道的 BRDY 中断。
由上电复位或软件复位初始化本寄存器。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	PIPE7 BRDYE	PIPE6 BRDYE	PIPE5 BRDYE	PIPE4 BRDYE	PIPE3 BRDYE	PIPE2 BRDYE	PIPE1 BRDYE	PIPE0 BRDYE
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15 ~ 8	—	均为 0	R	保留位 读取值、写入值总是为 0。
7	PIPE7BRDYE	0	R/W	管道 7 的 BRDY 中断允许 0: 禁止中断输出 1: 允许中断输出
6	PIPE6BRDYE	0	R/W	管道 6 的 BRDY 中断允许 0: 禁止中断输出 1: 允许中断输出
5	PIPE5BRDYE	0	R/W	管道 5 的 BRDY 中断允许 0: 禁止中断输出 1: 允许中断输出
4	PIPE4BRDYE	0	R/W	管道 4 的 BRDY 中断允许 0: 禁止中断输出 1: 允许中断输出
3	PIPE3BRDYE	0	R/W	管道 3 的 BRDY 中断允许 0: 禁止中断输出 1: 允许中断输出
2	PIPE2BRDYE	0	R/W	管道 2 的 BRDY 中断允许 0: 禁止中断输出 1: 允许中断输出
1	PIPE1BRDYE	0	R/W	管道 1 的 BRDY 中断允许 0: 禁止中断输出 1: 允许中断输出
0	PIPE0BRDYE	0	R/W	管道 0 的 BRDY 中断允许 0: 禁止中断输出 1: 允许中断输出

【注】 清除各中断状态后，允许 / 禁止相应中断时，必须至少空出 80ns 的间隔。

23.3.14 NRDY 中断允许寄存器 (NRDYENB)

本寄存器允许各管道 NRDY 的中断。
由上电复位或软件复位初始化本寄存器。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	PIPE7 NRDYE	PIPE6 NRDYE	PIPE5 NRDYE	PIPE4 NRDYE	PIPE3 NRDYE	PIPE2 NRDYE	PIPE1 NRDYE	PIPE0 NRDYE
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15 ~ 8	—	均为 0	R	保留位 读取值、写入值总是为 0。
7	PIPE7NRDYE	0	R/W	管道 7 的 NRDY 中断允许 0: 禁止中断输出 1: 允许中断输出
6	PIPE6NRDYE	0	R/W	管道 6 的 NRDY 中断允许 0: 禁止中断输出 1: 允许中断输出
5	PIPE5NRDYE	0	R/W	管道 5 的 NRDY 中断允许 0: 禁止中断输出 1: 允许中断输出
4	PIPE4NRDYE	0	R/W	管道 4 的 NRDY 中断允许 0: 禁止中断输出 1: 允许中断输出
3	PIPE3NRDYE	0	R/W	管道 3 的 NRDY 中断允许 0: 禁止中断输出 1: 允许中断输出
2	PIPE2NRDYE	0	R/W	管道 2 的 NRDY 中断允许 0: 禁止中断输出 1: 允许中断输出
1	PIPE1NRDYE	0	R/W	管道 1 的 NRDY 中断允许 0: 禁止中断输出 1: 允许中断输出
0	PIPE0NRDYE	0	R/W	管道 0 的 NRDY 中断允许 0: 禁止中断输出 1: 允许中断输出

【注】 清除各中断状态后，允许 / 禁止相应中断时，必须至少空出 80ns 的间隔。

23.3.15 BEMP 中断允许寄存器 (BEMPENB)

本寄存器允许各管道 BEMP 的中断。
由上电复位或软件复位初始化本寄存器。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	PIPE7 BEMPE	PIPE6 BEMPE	PIPE5 BEMPE	PIPE4 BEMPE	PIPE3 BEMPE	PIPE2 BEMPE	PIPE1 BEMPE	PIPE0 BEMPE
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15 ~ 8	—	均为 0	R	保留位 读取值、写入值总是为 0。
7	PIPE7BEMPE	0	R/W	管道 7 的 BEMP 中断允许 0: 禁止中断输出 1: 允许中断输出
6	PIPE6BEMPE	0	R/W	管道 6 的 BEMP 中断允许 0: 禁止中断输出 1: 允许中断输出
5	PIPE5BEMPE	0	R/W	管道 5 的 BEMP 中断允许 0: 禁止中断输出 1: 允许中断输出
4	PIPE4BEMPE	0	R/W	管道 4 的 BEMP 中断允许 0: 禁止中断输出 1: 允许中断输出
3	PIPE3BEMPE	0	R/W	管道 3 的 BEMP 中断允许 0: 禁止中断输出 1: 允许中断输出
2	PIPE2BEMPE	0	R/W	管道 2 的 BEMP 中断允许 0: 禁止中断输出 1: 允许中断输出
1	PIPE1BEMPE	0	R/W	管道 1 的 BEMP 中断允许 0: 禁止中断输出 1: 允许中断输出
0	PIPE0BEMPE	0	R/W	管道 0 的 BEMP 中断允许 0: 禁止中断输出 1: 允许中断输出

【注】 清除各中断状态后，允许 / 禁止相应中断时，必须至少空出 80ns 的间隔。

23.3.16 中断状态寄存器 0 (INTSTS0)

本寄存器为确认各中断状态的寄存器。

由上电复位或软件复位初始化本寄存器。通过 USB 总线复位时初始化 DVSQ[2:0] 位。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VBINT	RESM	SOFR	DVST	CTRT	BEMP	NRDY	BRDY	VBSTS	DVSQ[2:0]			VALID	CTSQ[2:0]		
初始值:	0	0	0	0	0	0	0	0	*3	*4	*4	*4	0	0	0	0
R/W:	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R	R	R	R	R	R	R	R/W*1	R	R	R

位	位名称	初始值	R/W	说 明
15	VBINT	0	R/W*1	VBUS 中断状态 *2 0: 不产生 VBUS 中断 1: 产生 VBUS 中断
14	RESM	0	R/W*1	恢复中断状态 *2 0: 不产生恢复中断 1: 产生恢复中断
13	SOFR	0	R/W*1	帧编号更新中断状态 *2 0: 不产生 SOF 中断 1: 产生 SOF 中断
12	DVST	0	R/W*1	器件状态转移中断状态 *2 0: 不产生器件状态转移中断 1: 产生器件状态转移中断
11	CTRT	0	R/W*1	控制传送阶段转移中断状态 *2 0: 不产生控制传送阶段转移中断 1: 产生控制传送阶段转移中断
10	BEMP	0	R	缓冲器空中断状态 清除 BEMPSTS 寄存器的所有位时, 清除本位。 0: 不产生 BEMP 中断 1: 产生 BEMP 中断
9	NRDY	0	R	缓冲器未就绪中断状态 清除 NRDYSTS 寄存器的所有位时, 清除本位。 0: 不产生 NRDY 中断 1: 产生 NRDY 中断
8	BRDY	0	R	缓冲器就绪中断状态 清除 BRDYSTS 寄存器的所有位时, 清除本位。 0: 不产生 BRDY 中断 1: 产生 BRDY 中断
7	VBSTS	*3	R	VBUS 输入状态 本位监控 VBUS 引脚的状态。但需要通过控制用程序去除震颤。 0: VBUS 引脚为低电平 1: VBUS 引脚为高电平

位	位名称	初始值	R/W	说 明
6 ~ 4	DVSQ[2:0]	*4	R	器件状态 000: 上电状态 001: 默认状态 010: 地址状态 011: 结构状态 1xx: 挂起状态
3	VALID	0	R/W*1	设置包接收 0: 未检测 1: 设置包接收
2 ~ 0	CTSQ[2:0]	000	R	控制传送阶段 000: 空闲或设置阶段 001: 控制读取数据阶段 010: 控制读取状态阶段 011: 控制写入数据阶段 100: 控制写入状态阶段 101: 控制写入 (NoData) 状态阶段 110: 控制传送顺序错误 111: 禁止设定

【注】 *1 仅写入 0 有效。

*2 如果 VBINT 位、RESM 位、SOFR 位、DVST 位及 CTRT 位中产生多个源，连续而非同时清除各位时，则 140ns+ 至少需要 3 个总线时钟周期的存取周期。

*3 VBUS 引脚为高电平时置 1；低电平时置 0。

*4 上电复位、软件复位时为 B'000；USB 总线复位时，为 B'001。

23.3.17 中断状态寄存器 1 (INTSTS1)

本寄存器为确认各中断状态的寄存器。SOFR、BEMP、NRDY 及 BRDY 位为 INTSTS0 寄存器的镜像位。读取时，可读出与 INTSTS0 寄存器同一个位相同的值；写入时，写入的值反映到 INTSTS0 与 INTSTS1 两个寄存器。

用 INTSTS0 寄存器选择外围控制器功能、用 INTSTS1 寄存器选择主机控制器功能时，通过参照任何一方的寄存器，可确认中断的产生。

由上电复位或软件复位初始化本寄存器。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	BCHG	SOFR	DTCH	—	BEMP	NRDY	BRDY	—	—	SIGN	SACK	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W*	R/W*	R/W*	R	R	R	R	R	R	R/W*	R/W*	R	R	R	R

位	位名称	初始值	R/W	说 明
15	—	0	R	保留位 读取值、写入值总是为 0。
14	BCHG	0	R/W*	USB 总线变化中断状态 0: 不产生 BCHG 中断 1: 产生 BCHG 中断

位	位名称	初始值	R/W	说 明
13	SOFR	0	R/W*	帧编号更新中断状态 0: 不产生 SOF 中断 1: 产生 SOF 中断
12	DTCH	0	R/W*	全速运行时中断断开检测状态 由本位检测断开, 仅在选择主机控制器功能且全速运行时有效。高速运行时, 必须由软件检测断开 (检测外围器件无响应等)。 0: 不产生 DTCH 中断 1: 产生 DTCH 中断 【注】 判断在复位信号交换协议后为高速运行 (RGST=11) 时, 高速通信过程中, 必须设定 DTCHE=0。此外, 高速通信过程中, 本位可能置 1, 因此, 完成高速通信后, 本位必须清 0。
11	—	0	R	保留位 读取值、写入值总是为 0。
10	BEMP	0	R	缓冲器空中断状态 0: 不产生 BEMP 中断 1: 产生 BEMP 中断
9	NRDY	0	R	缓冲器未就绪中断状态 0: 不产生 NRDY 中断 1: 产生 NRDY 中断
8	BRDY	0	R	缓冲器就绪中断状态 0: 不产生 BRDY 中断 1: 产生 BRDY 中断
7、6	—	均为 0	R	保留位 读取值、写入值总是为 0。
5	SIGN	0	R/W*	设置处理错误中断状态 0: 不产生 SIGN 中断 1: 产生 SIGN 中断
4	SACK	0	R/W*	设置处理正常响应中断状态 0: 不产生 SACK 中断 1: 产生 SACK 中断
3 ~ 0	—	均为 0	R	保留位 读取值、写入值总是为 0。

【注】 * 仅写入 0 有效。

23.3.18 BRDY 中断状态寄存器 (BRDYSTS)

本寄存器为确认各管道的 BRDY 中断状态的寄存器。

由上电复位或软件复位初始化本寄存器。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	PIPE7 BRDY	PIPE6 BRDY	PIPE5 BRDY	PIPE4 BRDY	PIPE3 BRDY	PIPE2 BRDY	PIPE1 BRDY	PIPE0 BRDY
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1

位	位名称	初始值	R/W	说 明
15 ~ 8	—	均为 0	R	保留位 读取值、写入值总是为 0。
7	PIPE7BRDY	0	R/W*1	管道 7 的 BRDY 中断状态 *2 0: 不产生中断 1: 产生中断
6	PIPE6BRDY	0	R/W*1	管道 6 的 BRDY 中断状态 *2 0: 不产生中断 1: 产生中断
5	PIPE5BRDY	0	R/W*1	管道 5 的 BRDY 中断状态 *2 0: 未产生中断 1: 产生中断
4	PIPE4BRDY	0	R/W*1	管道 4 的 BRDY 中断状态 *2 0: 未产生中断 1: 产生中断
3	PIPE3BRDY	0	R/W*1	管道 3 的 BRDY 中断状态 *2 0: 不产生中断 1: 产生中断
2	PIPE2BRDY	0	R/W*1	管道 2 的 BRDY 中断状态 *2 0: 不产生中断 1: 产生中断
1	PIPE1BRDY	0	R/W*1	管道 1 的 BRDY 中断状态 *2 0: 不产生中断 1: 产生中断
0	PIPE0BRDY	0	R/W*1	管道 0 的 BRDY 中断状态 *2 0: 不产生中断 1: 产生中断

【注】 *1 仅写入 0 有效。

*2 产生多个管道的中断源，连续而非同时清除时，140ns+ 至少需要 3 个总线时钟周期的存取周期。

23.3.19 NRDY 中断状态寄存器 (NRDYSTS)

本寄存器为确认各管道的 NRDY 中断状态的寄存器。

由上电复位或软件复位初始化本寄存器。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	PIPE7 NRDY	PIPE6 NRDY	PIPE5 NRDY	PIPE4 NRDY	PIPE3 NRDY	PIPE2 NRDY	PIPE1 NRDY	PIPE0 NRDY
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1

位	位名称	初始值	R/W	说 明
15 ~ 8	—	均为 0	R	保留位 读取值、写入值总是为 0。
7	PIPE7NRDY	0	R/W*1	管道 7 的 NRDY 中断状态 *2 0: 不产生中断 1: 产生中断
6	PIPE6NRDY	0	R/W*1	管道 6 的 NRDY 中断状态 *2 0: 不产生中断 1: 产生中断
5	PIPE5NRDY	0	R/W*1	管道 5 的 NRDY 中断状态 *2 0: 未产生中断 1: 产生中断
4	PIPE4NRDY	0	R/W*1	管道 4 的 NRDY 中断状态 *2 0: 不产生中断 1: 产生中断
3	PIPE3NRDY	0	R/W*1	管道 3 的 NRDY 中断状态 *2 0: 不产生中断 1: 产生中断
2	PIPE2NRDY	0	R/W*1	管道 2 的 NRDY 中断状态 *2 0: 不产生中断 1: 产生中断
1	PIPE1NRDY	0	R/W*1	管道 1 的 NRDY 中断状态 *2 0: 未产生中断 1: 产生中断
0	PIPE0NRDY	0	R/W*1	管道 0 的 NRDY 中断状态 *2 0: 不产生中断 1: 产生中断

【注】 *1 仅写入 0 有效。

*2 产生多个管道的中断源，连续而非同时清除时，140ns+ 至少需要 3 个总线时钟周期的存取周期。

23.3.20 BEMP 中断状态寄存器 (BEMPSTS)

本寄存器为确认各管道的 BEMP 中断状态的寄存器。

由上电复位或软件复位初始化本寄存器。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	PIPE7 BEMP	PIPE6 BEMP	PIPE5 BEMP	PIPE4 BEMP	PIPE3 BEMP	PIPE2 BEMP	PIPE1 BEMP	PIPE0 BEMP
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1

位	位名称	初始值	R/W	说 明
15 ~ 8	—	均为 0	R	保留位 读取值、写入值总是为 0。
7	PIPE7BEMP	0	R/W*1	管道 7 的 BEMP 中断状态 *2 0: 不产生中断 1: 产生中断
6	PIPE6BEMP	0	R/W*1	管道 6 的 BEMP 中断状态 *2 0: 不产生中断 1: 产生中断
5	PIPE5BEMP	0	R/W*1	管道 5 的 BEMP 中断状态 *2 0: 不产生中断 1: 产生中断
4	PIPE4BEMP	0	R/W*1	管道 4 的 BEMP 中断状态 *2 0: 不产生中断 1: 产生中断
3	PIPE3BEMP	0	R/W*1	管道 3 的 BEMP 中断状态 *2 0: 不产生中断 1: 产生中断
2	PIPE2BEMP	0	R/W*1	管道 2 的 BEMP 中断状态 *2 0: 不产生中断 1: 产生中断
1	PIPE1BEMP	0	R/W*1	管道 1 的 BEMP 中断状态 *2 0: 不产生中断 1: 产生中断
0	PIPE0BEMP	0	R/W*1	管道 0 的 BEMP 中断状态 *2 0: 不产生中断 1: 产生中断

【注】 *1 仅写入 0 有效。

*2 产生多个管道的中断源，连续而非同时清除时，140ns+ 至少需要 3 个总线时钟周期的存取周期。

23.3.21 帧编号寄存器 (FRMNUM)

本寄存器为判别同步错误通知源、选择 SOFR 中断运行模式及表示帧编号的寄存器。
由上电复位或软件复位初始化本寄存器。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OVRN	CRCE	—	—	SOFRM	FRNM[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*1	R/W*1	R	R	R/W	R	R	R	R	R	R	R	R	R	R	R

位	位名称	初始值	R/W	说 明
15	OVRN	0	R/W*1	上溢 / 下溢 *2 0: 无错误 1: 产生错误 表示因管道 (同步传送中) 的 NRDY 中断引起的错误通知源为数据缓冲器错误。详情参照表 23.8 及表 23.9。
14	CRCE	0	R/W*1	接收数据错误 *2 0: 无错误 1: 产生错误 表示因管道 (同步传送中) 的 NRDY 中断引起的错误通知源为包错误。详情参照表 23.8 及表 23.9。
13、12	—	均为 0	R	保留位 读取值、写入值总是为 0。
11	SOFRM	0	R/W	帧编号更新中断输出模式 • 选择功能控制器功能时 0: 接收 SOF、定时器插值时中断有效 1: SOF 破损、脱落时中断有效 • 选择主机控制器功能时 0: 发送 SOF 时有效 1: 禁止设定 在检测出 UFRMNUM.UFRNM=000 以外的 μ SOF 包时, 不产生帧编号更新中断。
10 ~ 0	FRNM [10:0]	H'000	R	帧编号 可确认帧编号。 选择功能控制器功能时, 本模块在 SOF 包接收时序更新帧编号。由于包破损等原因模块无法检测出 SOF 包时, 在接收新的 SOF 包前, 保持 FRNM 的值。 不通过 SOF 插值定时器更新 FRNM 位。

【注】 *1 仅写入 0 有效。

*2 产生 OVRN 和 CRCE 的源, 连续而非同时清各位除时, 140ns+ 至少需要 3 个总线时钟周期的存取周期。

表 23.8 同步 OUT 传送中产生 NRDY 中断时的错误信息

位状态	产生时序	产生条件	错误检测	运行
OVRN=1	接收数据包	缓冲存储器读取结束前接收到新的数据包	接收数据缓冲器的上溢出	废弃接收数据
CRCE=1	接收数据包	检测出 CRC 错误或位填充错误	接收包错误	废弃接收数据

表 23.9 同步 IN 传送中产生 NRDY 中断时的错误信息

位状态	产生时序	产生条件	错误检测	运行
OVRN=1	接收 IN-Token	缓冲存储器写入结束前已接收 IN-Token	发送数据缓冲器的下溢出	发送 Zero-Length 包
CRCE=1	不产生			

23.3.22 μ 帧编号寄存器 (UFRMNUM)

本寄存器为表示 μ 帧编号的寄存器。

由上电复位或软件复位初始化本寄存器。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	UFRNM[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名称	初始值	R/W	说 明
15 ~ 3	—	均为 0	R	保留位 读取值、写入值总是为 0。
2 ~ 0	UFRNM[2:0]	000	R	μ 帧 可确认 μ 帧编号。 通过接收 uSOF 包, UFRNM 位递增。 全速运行时, 本位总是读出 B'000。

23.3.23 USB 地址寄存器 (USBADDR)

本寄存器表示 USB 地址。

本寄存器仅在选择功能控制器功能时有效；选择主机控制器功能时，必须使用 PIPEMAXP 寄存器的 DEVSEL 位设定外围地址。

由上电复位、软件复位及 USB 总线复位初始化本寄存器。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	USBADDR[6:0]						
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名称	初始值	R/W	说 明
15 ~ 7	—	均为 0	R	保留位 读取值、写入值总是为 0。
6 ~ 0	USBADDR[6:0]	H'00	R	USB 地址 表示 USB 地址。

23.3.24 USB 请求类型寄存器 (USBREQ)

本寄存器为保存控制传送的设置请求的寄存器。

选择功能控制器功能时，保存接收的 bRequest 及 bmRequestType 值；选择主机控制器功能时，设定发送的 bRequest 及 bmRequestType 值。

由上电复位、软件复位及 USB 总线复位初始化本寄存器。

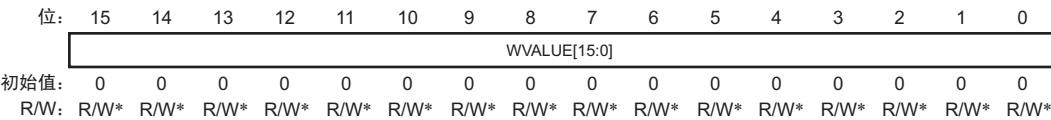
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BREQUEST[7:0]								BMREQUESTTYPE[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*

位	位名称	初始值	R/W	说 明
15 ~ 8	BREQUEST [7:0]	H'00	R/W*	请求 保存 USB 请求 bRequest 的值。
7 ~ 0	BMREQUEST TYPE[7:0]	H'00	R/W*	请求类型 保存 USB 请求 bmRequestType 的值。

【注】 * 选择功能控制器功能时，仅可读取；选择主机控制器功能时，可读取 / 写入。

23.3.25 USB 请求值寄存器 (USBVAL)

本寄存器为保存控制传送设置请求的寄存器。
 选择外围控制器功能时，保存接收的 wValue 值；选择主机控制器功能时，设定发送的 wValue 值。
 由上电复位、软件复位及 USB 总线复位初始化本寄存器。

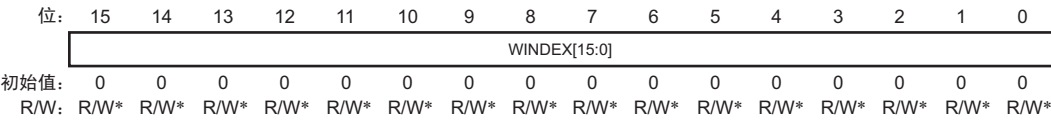


位	位名称	初始值	R/W	说 明
15 ~ 0	WVALUE[15:0]	H'0000	R/W*	值 保存 USB 请求 wValue 的值。

【注】 * 选择功能控制器功能时，仅可读取；选择主机控制器功能时，可读取 / 写入。

23.3.26 USB 请求变址寄存器 (USBINDX)

本寄存器是保存控制传送的设置请求的寄存器。
 选择功能控制器功能时，保存接收的 wIndex 值；选择主机控制器功能时，设定发送的 wIndex 值。
 由上电复位、软件复位及 USB 总线复位初始化本寄存器。

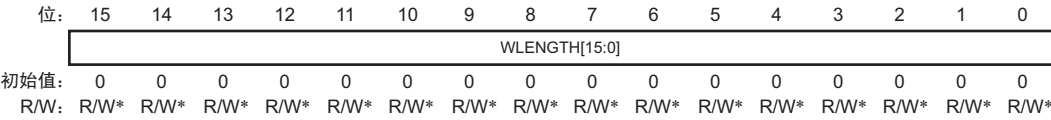


位	位名称	初始值	R/W	说 明
15 ~ 0	WINDEX[15:0]	H'0000	R/W*	变址 保存 USB 请求 wIndex 的值。

【注】 * 选择功能控制器功能时，仅可读取；选择主机控制器功能时，可读取 / 写入。

23.3.27 USB 请求长度寄存器 (USBLENG)

本寄存器是保存控制传送的设置请求的寄存器。
 选择外围控制器功能时，保存接收的 wLength 值；选择主机控制器功能时，设定发送的 wLength 值。
 由上电复位、软件复位及 USB 总线复位初始化本寄存器。



位	位名称	初始值	R/W	说 明
15 ~ 0	WLENGTH[15:0]	H'0000	R/W*	长度 保存 USB 请求 wLength 的值。

【注】 * 选择功能控制器功能时，仅可读取；选择主机控制器功能时，可读取 / 写入。

23.3.28 DCP 结构寄存器 (DCPCFG)

对于默认控制管道 (DCP)，本寄存器选择连续传送模式或非连续传送模式、是否继续 PIPE、传送结束时是否禁止管道运行。

由上电复位及软件复位初始化本寄存器。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	CNTMD	SHTNAK	—	—	DIR	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R	R	R/W	R	R	R	R

位	位名称	初始值	R/W	说 明
15 ~ 9	—	均为 0	R	保留位 读取值、写入值总是为 0。
8	CNTMD	0	R/W	连续传送模式 0: 非连续传送模式 1: 连续传送模式 由控制读取传送及控制写入传送共同使用 DCP 缓冲存储器，所以无论为任何传送方向，CNTMD 位为通用位。
7	SHTNAK	0	R/W	DCP 的传送结束时，管道禁止模式 0: 传送结束时，管道继续 1: 传送结束时，管道禁止 (响应 PID 为 NAK)
6、5	—	0	R	保留位 读取值、写入值总是为 0。
4	DIR	0	R/W	传送方向 选择主机控制器功能时，设定控制传送的数据阶段及状态阶段的传送方向。选择功能控制器功能时，必须将本位置 0。 0: 数据接收方向 1: 数据发送方向
3 ~ 0	—	均为 0	R	保留位 读取值、写入值总是为 0。

23.3.29 DCP Max 包大小寄存器 (DCPMAXP)

本寄存器对 DCP 指定 Max 包大小。

由上电复位及软件复位初始化本寄存器。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DEVSEL[1:0]		—	—	—	—	—	—	—	MXPS[6:0]						
初始值:	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R*	R*	R*

位	位名称	初始值	R/W	说 明
15、14	DEVSEL[1:0]	00	R/W	选择器件 指定选择主机控制器功能时的通信对象的器件地址；选择功能控制器功能时，必须将本位的值设定为 B'00。 00: 地址 00 01: 地址 01 10: 地址 10 11: 地址 11
13 ~ 7	—	均为 0	R	保留位 读取值、写入值总是为 0。
6 ~ 0	MXPS[6:0]	H'40	R/W*	Max 包大小 指定 DCP 的 Max 包大小。 不得进行 USB 规格以外的设定。MXPS[2:0] 固定为 0。

【注】 * 对 MXPS[2:0] 的写入无效。

23.3.30 DCP 控制寄存器 (DCPCTR)

本寄存器对 DCP 进行缓冲存储器状态确认、数据 PID 顺序位的更改和确认及响应 PID 的设定。

由上电复位及软件复位初始化本寄存器。由 USB 总线复位初始化 CCPL 和 PID[1:0] 位。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BSTS	SUREQ	—	—	—	—	—	SQCLR	SQSET	SQMON	—	—	—	CCPL	PID[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
R/W:	R	R/W*2	R	R	R	R	R	R*1/ W*2	R*1/ W*2	R	R	R	R	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15	BSTS	0	R	缓冲器状态 0: 不可存取缓冲器 1: 可存取缓冲器 根据 CFIFOSEL.ISEL 位的设定值决定缓冲器存取方向是写入还是读取。
14	SUREQ	0	R/W*2	发送设置令牌 通过将本位置 1，发送设置包。设置处理结束后，本模块将本位清 0。本位的值为 1 期间，不得写入 USBREQ、USBVAL、USBINDX 及 USBLENG 寄存器。 0: 无效 1: 发送设置包

位	位名称	初始值	R/W	说 明
13 ~ 9	—	均为 0	R	保留位 读取值、写入值总是为 0。
8	SQCLR	0	R*1/W*2	交替位清除 *3*4 0: 无效 1: 指定 DATA0
7	SQSET	0	R*1/W*2	交替位置位 *3*4 0: 无效 1: 指定 DATA1
6	SQMON	1	R	交替位确认 0: DATA0 1: DATA1 选择功能控制器功能时, 在接收控制传送的 SETUP 令牌后, 立即通过本模块将本位初始化为 1。
5 ~ 3	—	均为 0	R	保留位 读取值、写入值总是为 0。
2	CCPL	0	R/W	控制传送结束允许 0: 无效 1: 控制传送结束 选择功能控制器功能时, 在接收 SETUP 令牌后本位立即清 0; 选择主机控制器功能时, 必须将本位置 0。
1、0	PID[1:0]	00	R/W	响应 PID 00: NAK 响应 01: BUF 响应 (根据缓冲器状态) 10: STALL 响应 11: STALL 响应 选择功能控制器功能时, 本位在接收 SETUP 令牌后立即清除为 B'00。产生传送错误等时, 通过本控制器设定本位, 结束传送。

【注】 *1 仅读取 0 有效。

*2 仅写入 1 有效。

*3 不得同时将 SQCLR 位及 SQSET 位置 1。任何一位操作前必须设定 PID=NAK。

*4 连续更改 SQSET 位, 以及 PIPEnCTR 寄存器的 SQSET 位或 SQCLR 位时 (连续更改多个管道的数据 PID 顺序交替位时), 至少需要 120ns+5 个总线时钟周期的存取周期。

23.3.31 管道窗口选择寄存器 (PIPESEL)

本寄存器在管道 1 ~ 7，选择使用的管道。选择管道后，对 PIPECFG 寄存器、PIPEBUF 寄存器、PIPEMAXP 寄存器及 PIPEPERI 寄存器设定各管道的功能。PIPEnCTR 寄存器可设定为与通过 PIPESEL 选择管道无关。

上电复位、软件复位及 USB 总线复位时，不仅选择的管道，其他管道的寄存器的相应位均被初始化。
由上电复位或软件复位初始化本寄存器。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	PIPESEL[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15 ~ 3	—	均为 0	R	保留位 读取值、写入值总是为 0。
2 ~ 0	PIPESEL [2:0]	000	R/W	管道窗口选择 000: 未选择 001: 管道 1 010: 管道 2 011: 管道 3 100: 管道 4 101: 管道 5 110: 管道 6 111: 管道 7 设定 PIPESEL=000 时，对 PIPECFG 寄存器、PIPEBUF 寄存器、PIPEMAXP 寄存器、PIPEPERI 寄存器及 PIPEnCTR 寄存器的各位均读出 0。

23.3.32 管道结构寄存器 (PIPECFG)

对管道 1 ~ 7 指定传送类型、缓冲存储器的存取方向及端点编号，选择是连续传送模式或是非连续传送模式、是单缓冲器或双缓冲器及是否禁止传送结束时的管道运行。

由上电复位及软件复位初始化本寄存器。由 USB 总线复位初始化 TYPE[1:0]。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TYPE[1:0]		—	—	—	BFRE	DBLB	CNTMD	SHT NAK	—	—	DIR	EPNUM[3:0]			
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15、14	TYPE[1:0]	00	R/W	传送类型 - 管道 1、2 时 00: 不可使用管道 01: 批量传送 10: 禁止设定 11: 同步传送 * - 管道 3 ~ 5 时 00: 不可使用管道 01: 批量传送 10: 禁止设定 11: 禁止设定 - 管道 6、7 时 00: 不可使用管道 01: 禁止设定 10: 中断传送 11: 禁止设定
13 ~ 11	—	均为 0	R	保留位 读取值、写入值总是为 0。
10	BFRE	0	R/W	BRDY 中断运行指定 0: 通过发送 / 接收数据，指定 BRDY 中断 1: 通过读取数据，指定 BRDY 中断 设定 BFRE=1，且为数据写入方向时不产生 BRDY 中断。

位	位名称	初始值	R/W	说 明
9	DBLB	0	R/W	双缓冲器模式 0: 单缓冲器 1: 双缓冲器 本位在选择管道 1 ~ 5 时有效。 对某管道动态转换本位的设定时, 必须遵守以下步骤。 • 从单缓冲器转换为双缓冲器时 (DBLB=0→DBLB=1) 1. 将相应管道的响应 PID 设定为 NAK 2. PIPEnCTR.ACLRM=1 3. 通过软件等待 100ns 4. ACLRM=0 5. 更改 DBLB 位 6. 对 BUF 设定响应 PID • 从双缓冲器转换为单缓冲器时 (DBLB=1→DBLB=0) 1. 将相应管道的响应 PID 设定为 NAK 2. 更改 DBLB 位 3. PIPEnCTR.ACLRM=1 4. 对软件等待 100ns 5. ACLRM=0 6. 对 BUF 设定响应 PID
8	CNTMD	0	R/W	连续传送模式 本位在管道 1 ~ 5 选择批量传送时 (TYPE=01) 有效。选择同步传送时 (TYPE=11) 不可设定 CNTMD=1。另外, 已选择管道 6、7 时, 不可设定 CNTMD=1。 0: 非连续传送模式 1: 连续传送模式
7	SHTNAK	0	R/W	传送结束时的管道禁止 0: 传送结束时管道继续 1: 传送结束时管道禁止
6、5	—	均为 0	R	保留位 读取值、写入值总是为 0。
4	DIR	0	R/W	传送方向 0: 接收方向 1: 发送方向
3 ~ 0	EPNUM[3:0]	0000	R/W	端点编号 指定该管道的端点编号。

【注】 * 使用同步 OUT 传送时, 详情参阅 “23.5.1 使用同步 OUT 传送时的注意事项”。

23.3.33 管道缓冲器指定寄存器 (PIPEBUF)

本寄存器对管道 1 ~ 7 指定缓冲器大小及缓冲器编号。
由上电复位及软件复位初始化本寄存器。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	BUFSIZE[4:0]					—	—	—	BUFNMB[6:0]						
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15	—	0	R	保留位 读取值、写入值总是为 0。
14 ~ 10	BUFSIZE [4:0]	H'00	R/W	缓冲器大小 指定相应管道的缓冲器大小。 (从 0 : 64 字节到 H'1F : 2K 字节) 根据 PIPESEL 寄存器的 PIPESEL 位选择的管道, 本位可设定的值有所不同。 管道 1 ~ 5 时: 设定 BUFSIZE=H'00 ~ H'1F。 管道 6、7 时: 设定 BUFSIZE=H'00。
9 ~ 7	—	均为 0	R	保留位 读取值、写入值总是为 0。
6 ~ 0	BUFNMB [6:0]	H'00	R/W	缓冲器编号 指定相应管道的缓冲器编号 (H'04 ~ H'7F)。 选择管道 1 ~ 5 时, 可配合系统设定 BUFNMB 位。 BUFNMB=0 ~ 3 为 DCP 专用。BUFNMB=4、5 被配置到管道 6、7。 管道 1 ~ 5: 设定 BUFNMB=H'06 ~ H'7F。但是, 不使用 PIPE7 时, 可设定 BUFNMB=H'05 ~ H'7F。 不使用管道 6、7 时, 可设定 BUFNMB=H'04 ~ H'7F。 管道 6: 对本位写入无效, 读取值总是为 BUFNMB=4。 管道 7: 对本位写入无效, 读取值总是为 BUFNMB=5。

23.3.34 管道 Max 包大小寄存器（PIPEMAXP）

本寄存器对管道 1 ～ 7 指定 Max 包大小。
由上电复位及软件复位初始化本寄存器。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DEVSEL[1:0]		—	—	—	MXPS[10:0]										
初始值:	0	0	0	0	0	*	*	*	*	*	*	*	*	*	*	*
R/W:	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15、14	DEVSEL[1:0]	00	R/W	器件选择 本位在选择主机控制器功能时指定外围器件地址，选择功能控制器功能时，设定本位的值为 B'00。 00: 地址 00 01: 地址 01 10: 地址 10 11: 地址 11
13 ～ 11	—	均为 0	R	保留位 读取值、写入值总是为 0。
10 ～ 0	MXPS[10:0]	*	R/W	Max 包大小 本位指定相应管道的 Max 包大小。 本位按照传送类型，设定 USB 规格定义范围内的值。

【注】 * 不用 PIPESEL 寄存器的 PIPESEL 位选择管道时初始值为 H'000，用 PIPESEL 寄存器的 PIPESEL 位选择时初始值为 H'040。

23.3.35 管道周期控制寄存器 (PIPEPERI)

本寄存器对管道 1、2，选择是否通过同步 IN 传送时的间隔错误使缓冲器清除功能运行，并设定间隔错误的检测间隔。

由上电复位及软件复位初始化本寄存器。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	IFIS	—	—	—	—	—	—	—	—	—	IITV[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15 ~ 13	—	均为 0	R	保留位 读取值、写入值总是为 0。
12	IFIS	0	R/W	同步 IN 缓冲器清除 0: 不清除缓冲器 1: 清除缓冲器 本位仅在选择功能控制器功能时有效。 使用本位时，需按照以下步骤设定。 - 同步 IN 传送开始时 - IFIS=1 - PIPEnCTR.PID[1:0]=01 (BUF) - 将发送数据写入 Iso-IN PIPE 的 FIFO 缓冲器 【注】不使用 IFIS 传送时，无需上述步骤。 - 同步 IN 传送结束时 - PID[1:0]=00 (NAK) - PIPEnCTR.ACLR=1 - 至少等待 100ns - ACLR=0 【注】不使用 IFIS 传送时，无需设定 ACLR。
11 ~ 3	—	均为 0	R	保留位 读取值、写入值总是为 0。
2 ~ 0	IITV[2:0]	000	R/W	间隔错误检测间隔 本位按照帧时序的 2 的 n 次方指定间隔时序。 本位仅在选择功能控制器功能、同步传送时有效。即，只有选择管道 1、2 时才能设定。 设定 OUT 方向时: 在本位设定的间隔，如果不接收来自主机的 OUT 令牌，则通过 NRDY 中断检测出间隔错误，并产生 NRDY 中断。 设定 IN 方向时: 在本位设定的间隔，如果不接收来自主机的 IN 令牌，则检测出间隔错误，并且 IFIS=1 就进行缓冲器清除。 选择主机控制器功能时，本位对同步传送和中断传送有效。

23.3.36 管道 n 控制寄存器 (PIPEnCTR) (n=1 ~ 7)

本寄存器对管道 1 ~ 7，确认相应管道的缓冲存储器状态、更改和确认数据 PID 顺序位、选择是否设定为自动响应模式、选择是否设定为自动缓冲器清除模式及设定响应 PID。本设定与通过 PIPESEL 寄存器选择管道无关。

由上电复位或软件复位初始化本寄存器。由 USB 总线复位初始化 PID[1:0] 位。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BSTS	INBUFM	—	—	—	AT REPM	ACLRM	SQCLR	SQSET	SQMON	—	—	—	—	PID[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W*1	R/W*1	R	R	R	R	R	R/W	R/W

位	位名称	初始值	R/W	说 明
15	BSTS	0	R	缓冲器状态 0: 不可存取缓冲器 1: 可存取缓冲器 由 PIPECFG 寄存器的 DIR 位决定缓冲器存取方向为写入或读取。详情参阅“23.4 运行说明”
14	INBUFM	0	R	发送缓冲器监控 本位表示将相应管道设定为发送方向时有效的值。 0: 缓冲存储器中没有可发送的数据 1: 缓冲存储器中有可发送的数据 【注】本位对管道 1 ~ 5 有效。
13 ~ 11	—	均为 0	R	保留位 读取值、写入值总是为 0。
10	ATREPM	0	R/W	自动响应模式 0: 普通模式 1: 自动响应模式 【注】本位对管道 1 ~ 5 有效。
9	ACLRM	0	R/W	自动缓冲器清除模式 0: 禁止 1: 允许 (初始化所有缓冲器) 不可对 CFIFOSEL/DnFIFOSEL 寄存器的 CURPIPE 位选择的管道设定 ACLRM=1。
8	SQCLR	0	R/W*1	交替位清除 *2*3 0: 无效 1: 指定 DATA0
7	SQSET	0	R/W*1	交替位置位 *2*3 0: 无效 1: 指定 DATA1
6	SQMON	0	R	交替位确认 0: DATA0 1: DATA1
5 ~ 2	—	均为 0	R	保留位 读取值、写入值总是为 0。

位	位名称	初始值	R/W	说 明
1、0	PID[1:0]	00	R/W	响应 PID*3 00: NAK 响应 01: BUF 响应 (根据缓冲器状态) 10: STALL 响应 11: STALL 响应 选择主机控制器功能时, 如果未设定为 PID=BUF, 则不发行令牌。产生传送错误时等, 由控制器设定 PID 位并结束传送。

【注】 *1 仅读取 0 后写入 1 有效。

*2 连续更改 SQSET 位或 SQCLR 位及 DCPCTR 寄存器的 SQSET 位或 SQCLR 位时 (连续更改多个管道的数据 PID 的顺序交替位时), 至少需要 120ns+5 个总线时钟周期的存取周期。

*3 不可同时将 SQCLR 位及 SQSET 位置 1。另外, 任何一位操作前必须设定 PID=NAK。传送类型设定为同步传送 (TYPE=11) 时, 写入 SQSET 位无效。

23.3.37 USB AC 特性转换寄存器 (USBACSWR)

USBACSWR 设定本模块内置的 USB 收发器。

由上电复位初始化本寄存器。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	UACS23	-	-	-	-	-	-	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名称	初始值	R/W	说 明
31 ~ 24	—	均为 0	R	保留位 读取值、写入值总是为 0。
23	UACS23	0	R/W	USB AC 特性开关 设定 USB 收发器。*
22 ~ 0	—	均为 0	R	保留位 读取值、写入值总是为 0。

【注】 * 使用本模块时, 必须对本位写入 1。详情参阅 “23.5.2 USB 收发器的设定步骤”

23.4 运行说明

23.4.1 系统控制

本节说明本模块的初始设定所需的寄存器操作及功耗控制所需寄存器。

(1) 复位

表 23.10 所示为本模块的复位类型一览表。此外，有关各复位运行后的寄存器初始化状态，详情参阅“23.3 寄存器说明”。

表 23.10 复位类型一览表

名称	操 作
上电复位	从 $\overline{\text{RST}}$ 引脚输入低电平
软件复位	使用 SYSCFG 寄存器的 USBE 位进行操作
USB 总线复位	选择功能控制器功能时，本模块从 D+、D- 线自动检测

(2) 控制器功能的选择设定

本模块可选择主机控制器功能或功能控制器功能。控制器功能的选择必须由 SYSCFG 寄存器的 DCFM 位进行。

(3) 高速运行的允许

本模块可由软件设定 USB 通信速度（通信位速率）是高速运行或是全速运行。允许本模块高速运行时，必须将 SYSCFG 寄存器的 HSE 位置 1。必须在上电复位后的初始设定或 D+ 上拉禁止状态（DPRPU=0），更改 HSE 位。

允许高速运行时，本模块执行复位信号交换协议，并自动设定 USB 通信速度。复位信号交换的结果可由 DVSTCTR 寄存器的 RHST 位确认。

禁止高速运行时，本模块仅可全速运行。

(4) USB 数据总线电阻控制

本模块与 USB 连接器的连接如图 23.1 所示。

本模块内置 D+ 信号的上拉电阻和 D+、D- 信号的下拉电阻。必须由 SYSCFG 寄存器的 DPRPU、DMRPD 位设定各信号的上拉、下拉。

本模块控制 D+、D- 信号高速运行时的终端电阻和全速运行时的输出电阻。本模块通过复位信号交换协议、挂起、恢复检测，自动转换主机控制器或与外围器件连接后的内部电阻。如果检测出从主机控制器或外围器件断开时，必须由软件复位（SYSCFG.USBE=0），将本模块置为初始化状态。

另外，选择功能控制器功能，并且与主机控制器通信时设定 SYSCFG 寄存器的 DPRPU 位为 0 时，因禁止 USB 数据线的上拉电阻（或终端电阻），所以在连接 USB 电缆的状态，可由软件控制器件的断开和连接。

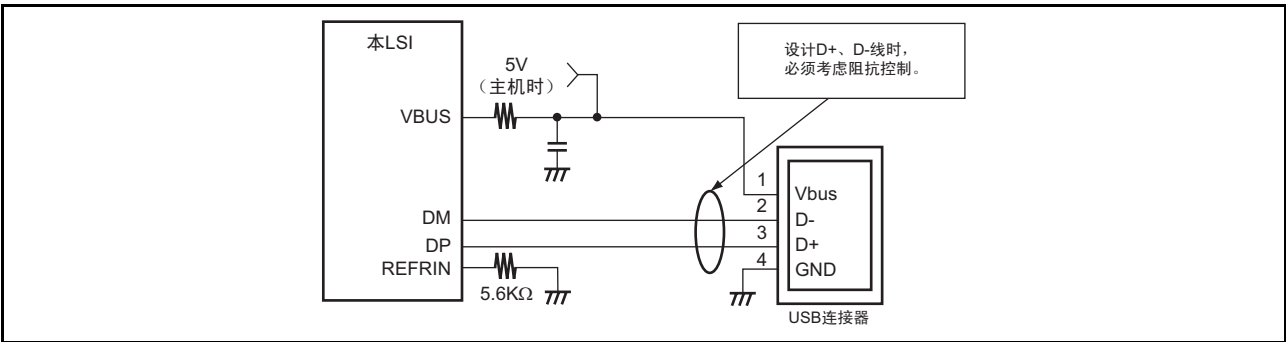


图 23.1 USB 连接器的连接图

23.4.2 中断功能

表 23.11 所示为本模块的中断产生条件一览表。

这些中断产生条件成立，且通过相应的中断允许寄存器设定允许中断输出时，本模块向中断控制器 (INTC) 输出 USB 中断请求信号。

表 23.11 中断产生条件一览表

位	中断名称	中断产生条件	产生的功能	相关状态
VBINT	VBUS 中断	检测出 VBUS 输入引脚的状态变化时 (L→H、H→L 双向变化)	主机、功能	VBSTS
RESM	恢复中断	在挂起状态，检测出 USB 总线的状态变化时 (J-State→K-State 或 J-State→SE0)	功能	—
SOFR	帧编号更新中断	[选择主机控制器功能时] - 发送与帧编号不同的 SOF 包时 [选择功能控制器功能时] - SOFRM=0: 接收与帧编号不同的 SOF 包时 - SOFRM=1: 由于 μ 帧编号为 0 时的 SOF 包破损等原因而不可接收时	主机、功能	—
DVST	器件状态转移中断	- 检测出器件状态转移时 - 检测出 USB 总线复位 - 检测出挂起状态 - 接收 Set Address 请求 - 接收 Set Configuration 请求	功能	DVSQ
CTRT	控制传送阶段转移中断	- 检测出控制传送阶段转移时 - 设置阶段结束 - 控制写入传送状态阶段转移 - 控制读取传送状态阶段转移 - 控制传送结束 - 产生控制传送顺序错误	功能	CTSQ
BEMP	缓冲器空中断	- 发送缓冲存储器中的全部数据后，缓冲器为空时 - 接收的包超过 Max 包大小时	主机、功能	BEMPSTS. PIPEBEMP
NRDY	缓冲器未就绪中断	[选择主机控制器功能时] - 对于发行的令牌，接收到来自功能控制器的 STALL 时 - 对于发行的令牌，没有来自功能控制器的响应时 (无响应) - 在同步传送中发生上溢 / 下溢时 [选择功能控制器功能时] - 接收 IN 令牌时，没有可向缓冲存储器发送的数据时 - 接收 OUT 令牌时，由于缓冲存储器中没有数据保存区域，而不可接收时 - 同步传送时，发生 CRC 错误、位填充错误时	主机、功能	NRDYSTS. PIPENRDY
BRDY	缓冲器就绪中断	为缓冲器就绪 (可读取或可写入状态) 时	主机、功能	BRDYSTS. PIPEBRDY
BCHG	总线变化中断	检测出 USB 总线状态变化时	主机、功能	—
DTCH	全速运行时的断开检测	全速运行时检测出功能控制器断开时	主机	—
SACK	SETUP 正常	接收到设置处理的正常响应 (ACK) 时	主机	—
SIGN	SETUP 错误	检测出设置处理的错误 (无响应、ACK 包破损) 时	主机	—

图 23.2 所示为与 USB 模块中断相关的图。

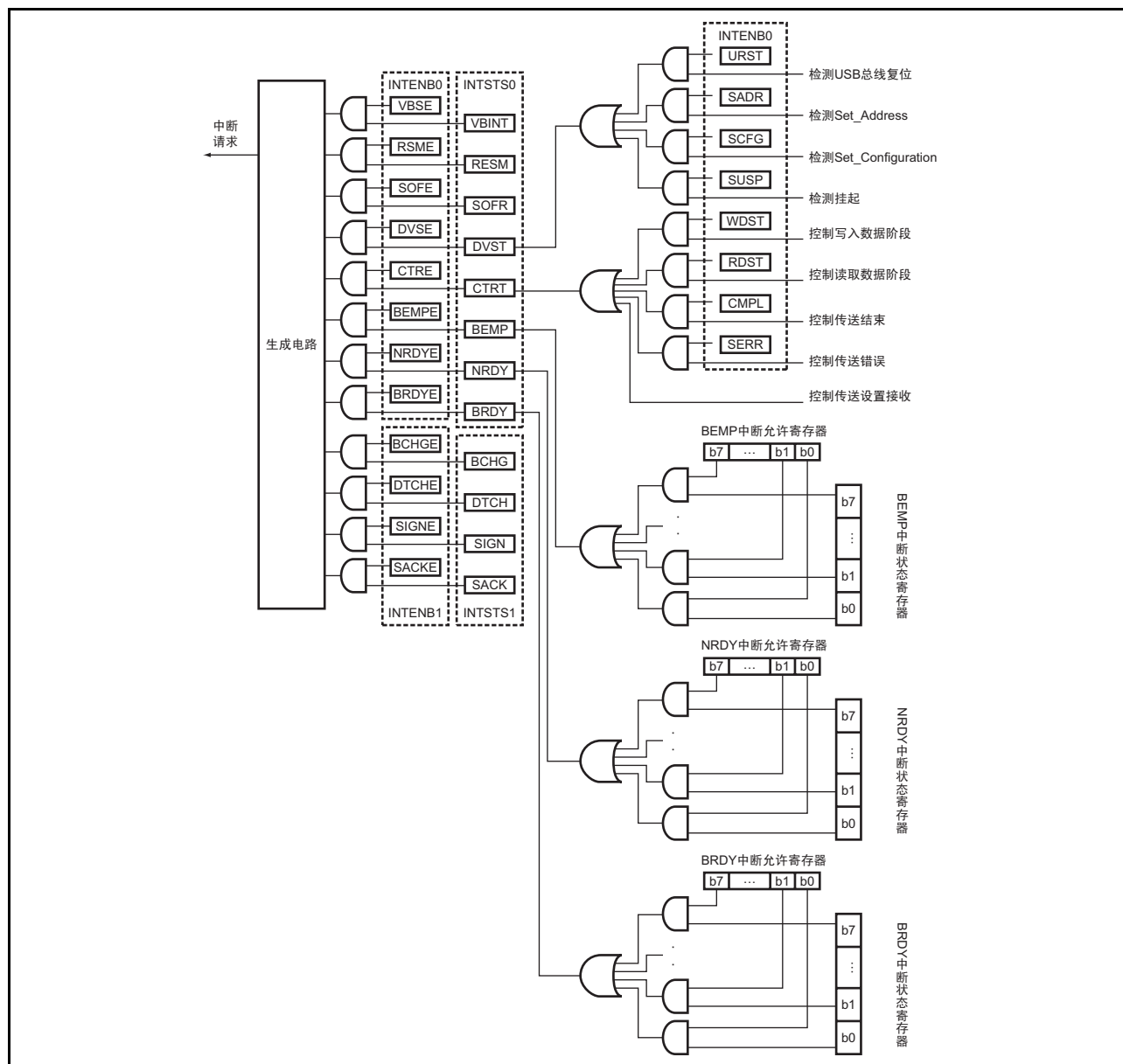


图 23.2 中断的相关图

(1) BRDY 中断

无论选择主机控制器还是功能控制器功能，均会产生 BRDY 中断。各管道满足表 23.12 的条件时，本模块将 BRDYSTS 寄存器的相应位置 1。如果软件将该管道对应的 BRDYENB 寄存器的 PIPEBRDYE 位设定为 1，且将 INTENB0 寄存器的 BRDYE 位设定为 1，则本模块产生 BRDY 中断。

BRDY 中断产生时序如图 23.3 所示。

本模块清除 INTSTS0 寄存器的 BRDY 位的条件，因 INTENB1 寄存器的 BRDYM 位的设定值而不同。

BRDY 位清除条件如表 23.13 所示。

另外，选择功能控制器功能时，通过以下条件 1 总是对 IN 令牌发送 Zero-Length 包，且不产生 BRDY 中断。

1. 设定传送类型为批量 IN 传送、PID=BUF，且设定该管道的 PIPEnCTR 寄存器的 ATREPM 位为 H'01 时

表 23.12 BRDY 中断产生条件表

存取方向	传送方向	管道	BFRE	DBLB	BRDY 中断产生条件
读取	接收	DCP	—	—	进行以下任意接收时 - 接收含 Zero-Length 的短包 - 接收数据直至缓冲器满
		1 ~ 7	0	0	进行以下任意接收时 - 接收含 Zero-Length 的短包 - 接收数据直到缓冲器满 * - 缓冲器未满，但处理计数器结束
				1	满足以下任何一个条件时 - 缓冲器两侧均为等待接收的状态时，用 1. ~ 3. 的任一种方法进行接收 1. 接收含 Zero-Length 的短包 2. 接收数据直到缓冲器的一侧满 * 3. 缓冲器未满，但处理计数器结束 - 缓冲器两侧均为等待读取状态时，读完一侧的接收数据 - 缓冲器两侧均为等待读取状态时，由软件设置 BCLR=1，清除一侧的接收数据 - 设定为连续传送模式时 (PIPECFG.CNTMD=1)，在 SIE 侧的缓冲器含数据的状态，设定 CFIFOSIE.TGL=1
			1	Don't care	满足以下任何一个条件时 - 接收 Zero-Length 包 - 接收短包，且完成该短包的数据读取 - 处理计数器结束，且完成最后包的数据读取

【注】 * 该缓冲器满即：设定为非连续传送 (CNTMD=0) 时接收 Max 包大小的数据；设定为连续传送 (CBTMD=1) 时接收与缓冲器大小的数据。

存取方向	传送方向	管道	BFRE	DBLB	BRDY 中断产生条件
写入	发送	DCP	—	—	不产生
		1 ~ 7	0	0	满足以下任何一个条件时 - 由软件将传送方向从接收更改为发送 - 在缓冲器有可发送数据的状态，完成该数据的发送 - 在缓冲器有可发送数据的状态，软件设定 ACLRM=1 - 在缓冲器有可发送数据的状态，软件设定 SCLR=1
				1	满足以下任何一个条件时 - 由软件将传送方向从接收更改为发送 - 在缓冲器没有可发送数据的状态，用以下 1. ~ 3. 的任何一种方法使数据为可发送状态 - 将 n 倍 Max 包大小的数据写入缓冲器，缓冲器为满（非连续传送时：n=1） - 软件设定 DnFIFOCTR.BVAL=1，使缓冲器可发送 - 通过 DMA 传送结束写入 - 在缓冲器两侧有可发送数据的状态，完成一侧数据的发送 - 在缓冲器两侧有可发送数据的状态，软件设定 ACLRM=1 - 在缓冲器两侧有可发送数据的状态，软件设定 SCLR=1
			1	Don't care	不产生

接收到 Zero-Length 包时，BRDYSTS 寄存器的相应位为 1，但不可读取该包的数据。清除 BRDYSTS 寄存器之后，必须清除缓冲器（BCLR=1）。

同时，在管道 1 ~ 7，如果在读取方向使用 DMA 传送，则可通过将 PIPECFG 寄存器的 BFRE 位置 1，按传送单位产生中断。

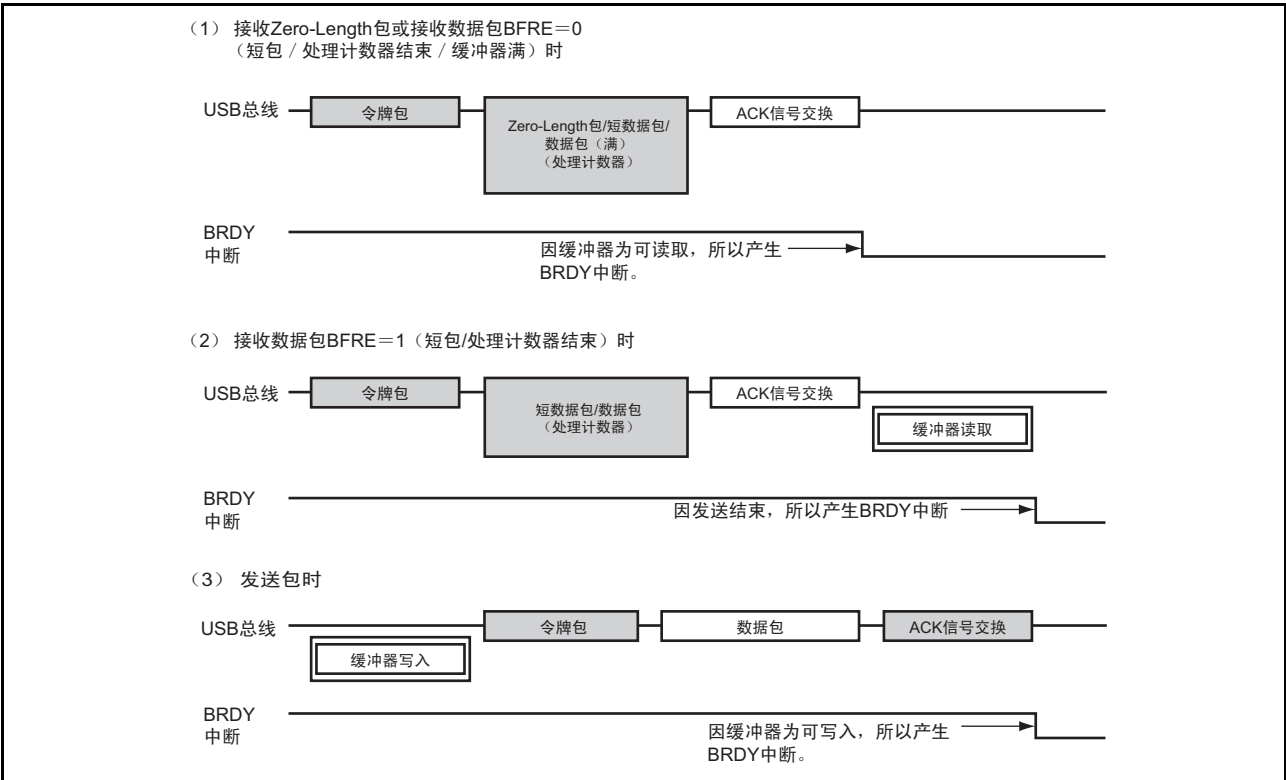


图 23.3 BRDY 中断产生时序图

表 23.13 通过本模块清除 BRDY 位的条件表

BRDYM	BRDY 位清除条件
0	如果软件清除 BRDYSTS 寄存器的所有位，则本模块清除 INTSTS0 寄存器的 BRDY 位。
1	所有管道的 BSTS 位为 0 时，本模块清除 INTSTS0 寄存器的 BRDY 位。

(2) NRDY 中断

各管道满足以下条件时，本模块将 NRDYSTS 寄存器的相应位置 1。如果软件将对应该管道的 NRDYENB 寄存器的 PIPENRDYE 位设定为 1，且将 INTENB0 寄存器的 NRDYE 位也设定为 1，则本模块产生 NRDY 中断。软件清除 NRDYSTS 寄存器的所有位时，本模块清除 INTSTS0 寄存器的 NRDY 位。

(a) 选择主机控制器功能时

以下任一条件，均可产生 NRDY 中断。此时，硬件设定 PID 位并停止发行令牌。有关 PID 位的运行，详情参阅“23.4.3(4) 响应 PID”。

- 对于发行的令牌，接收来自外围侧的 STALL 时
- 对于发行的令牌，没有来自外围侧的响应时 (无响应)
- 同步传送中产生上溢/下溢错误时

但在设置处理中不可接收外围侧的 ACK 时，产生 SIGN 中断。

(b) 选择功能控制器功能时

通过以下条件产生 NRDY 中断

1. 发送数据时

- 在 PIPEnCTR 寄存器的 PID 位为 PID=BUF、且缓冲存储器中无发送数据的状态，接收到 IN 令牌（数据下溢）时

2. 接收数据时

- 在 PIPEnCTR 寄存器的 PID 位为 PID=BUF、且缓冲存储器中无保存接收数据区域的状态，接收到 OUT 令牌或 PING 令牌（数据上溢）时
- 传送类型为批量传送时，在未设定 Max 包大小（MXPS=0）的管道接收到 OUT 令牌或 PING 令牌时
- 同步传送时，产生 CRC 错误、位填充错误时
- 同步传送时，在间隔帧以外接收到令牌时（间隔错误）

选择功能控制器功能时的 NRDY 中断产生时序如图 23.4 所示：

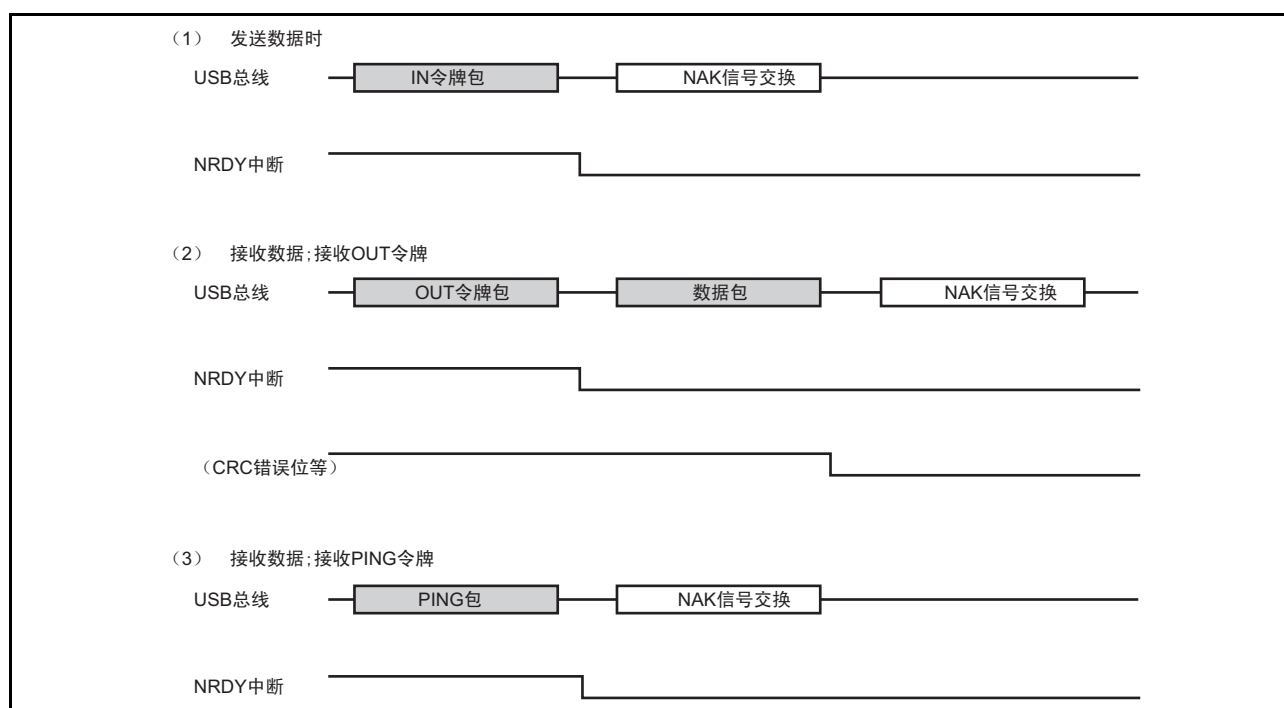


图 23.4 选择功能控制器功能时的 NRDY 中断产生时序图

(3) BEMP 中断

各管道满足以下条件时，本模块将 BEMPSTS 寄存器的相应位置 1。如果软件将该管道对应的 BEMPENB 寄存器的 PIPEBEMPE 位置 1 且将 INTENB0 寄存器的 BEMPE 位置 1，则本模块产生 BEMP 中断。如果软件清除 BEMPSTS 寄存器的所有位，则本模块清除 INTSTS0 寄存器的 BEMP 位。

1. 设定传送方向（写入缓冲存储器）时

传送保存在缓冲存储器的所有数据时。

但是缓冲存储器用作双缓冲器时，遵从以下条件。

- 一侧的缓冲器为空状态，且另一侧缓冲器的数据传送结束时，产生 BEMP 中断。
- 写入一侧的缓冲器的数据不满 8 字节且另一侧的缓冲器数据传送结束时，产生 BEMP 中断。

2. 设定接收方向（读取缓冲存储器）时

接收的数据包大小超过设定的 Max 包大小时。

此时，Max 包大小未设定为 0（MXPS=0）时，本模块将该管道的 PID 位设定为 STALL。

选择功能控制器功能时的 BEMP 中断产生时序如图 23.5 所示。

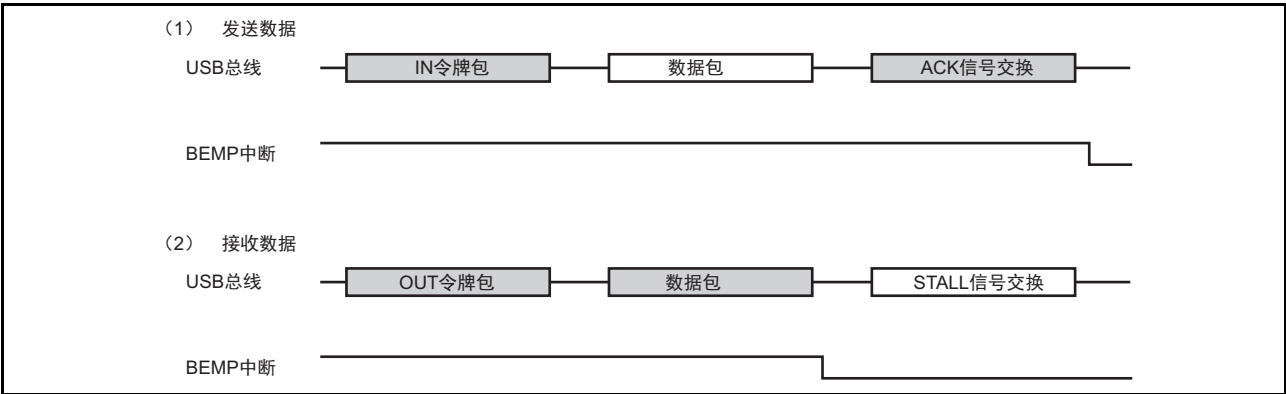


图 23.5 选择功能控制器功能时的 BEMP 中断产生时序图

(4) 器件状态转移中断

本模块的器件状态转移如图 23.6 所示。本模块控制器件状态，并产生器件状态转移中断。但是，由恢复中断检测出从挂起状态的返回（恢复信号检测）。由 INTENB0 寄存器可个别禁止或允许器件状态转移中断，并可由 INTSTS0 寄存器的 DVSQ 位，确认转移的器件状态。

转移到默认状态时，在复位信号交换协议结束后，产生器件状态转移中断。

仅在选择功能控制器功能时，可控制器件状态，产生器件状态转移中断。

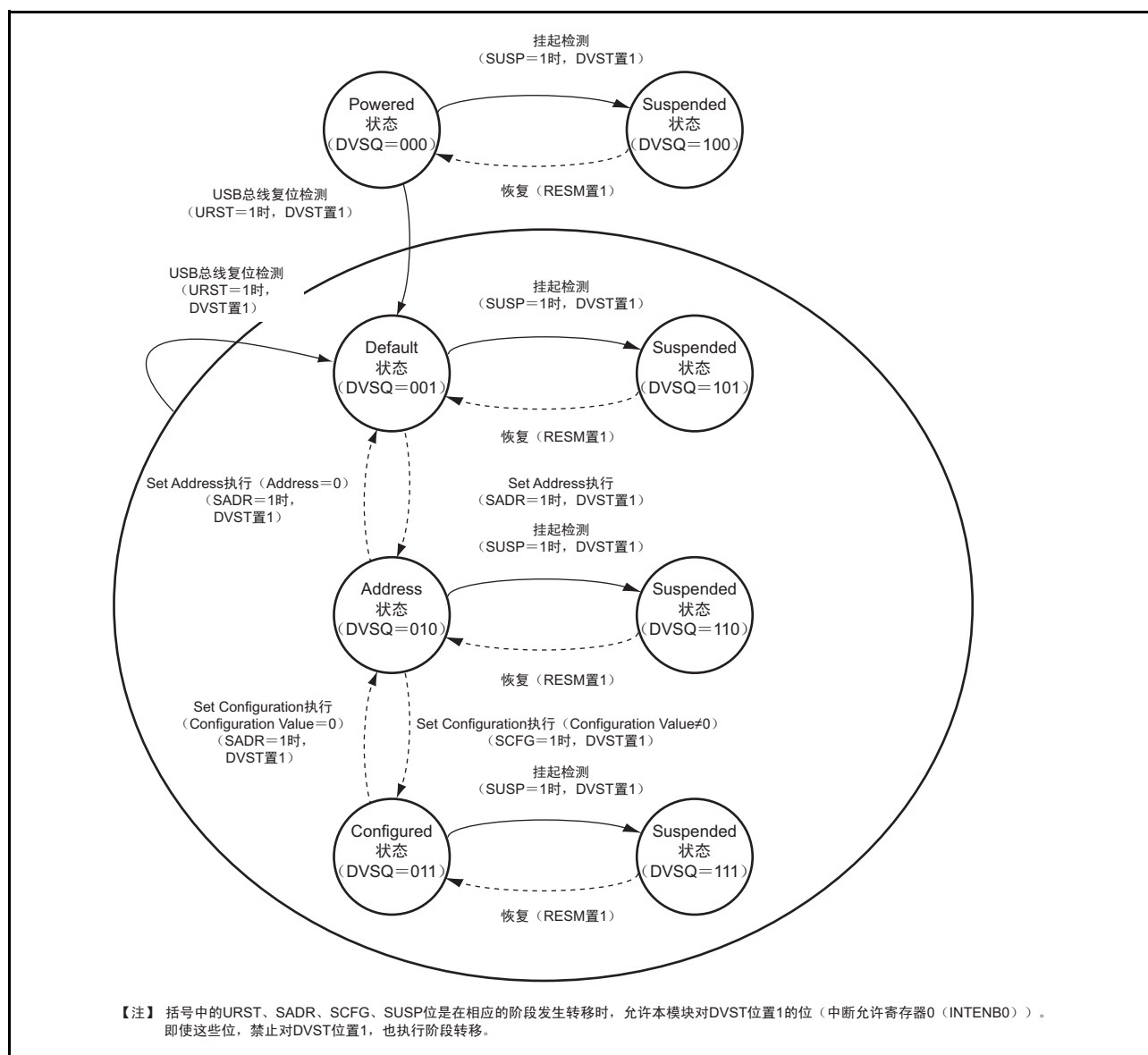


图 23.6 器件状态转移图

(5) 控制传送阶段转移中断

本模块的控制传送阶段转移如图 23.7 所示。本模块管理控制传送顺序，并产生控制传送阶段转移中断。由 INTENB0 寄存器可个别允许或禁止控制传送阶段转移中断，并可由 INTSTS0 寄存器的 CTSQ 位，确认转移的传送阶段。

控制传送的顺序错误如下所示。产生错误时，DCPCTR 寄存器的 PID 位为 B'1x (STALL 响应)。

1. 控制读取传送时

- 在数据没有传送一次的状态，对数据阶段的 IN 令牌，接收 OUT 或 PING 令牌
- 在状态阶段接收 IN 令牌
- 在状态阶段接收 DATA PID=DATA0 的数据包

2. 控制写入传送时

- 在没有一次 ACK 响应的状态，对数据阶段的 OUT 令牌，接收 IN 令牌
- 在数据状态阶段接收最初的 DATA PID=DATA0 的数据包
- 在状态阶段接收 OUT 或 PING 令牌

3. 控制写入无数据控制传送时

- 在状态阶段接收 OUT 或 PING 令牌

另外，在控制写入传送阶段，接收数据数超过 USB 请求的 wLength 值时，无法识别为控制传送顺序错误。在控制读取传送状态阶段，接收除 Zero-Length 包之外的包时，进行 ACK 响应并正常结束。

由于顺序错误产生 CTRT 中断时 (设定 SERR=1)，从系统写入 CTRT=0 (清除中断状态) 之前保持 CTSQ=110 的值。因此，即使在保持 CTSQ=110 的状态接收新的 USB 请求，也不产生设置阶段完成的 CTRT 中断 (本模块保持设置阶段结束，并由软件清除中断状态后，产生设置阶段完成中断)。

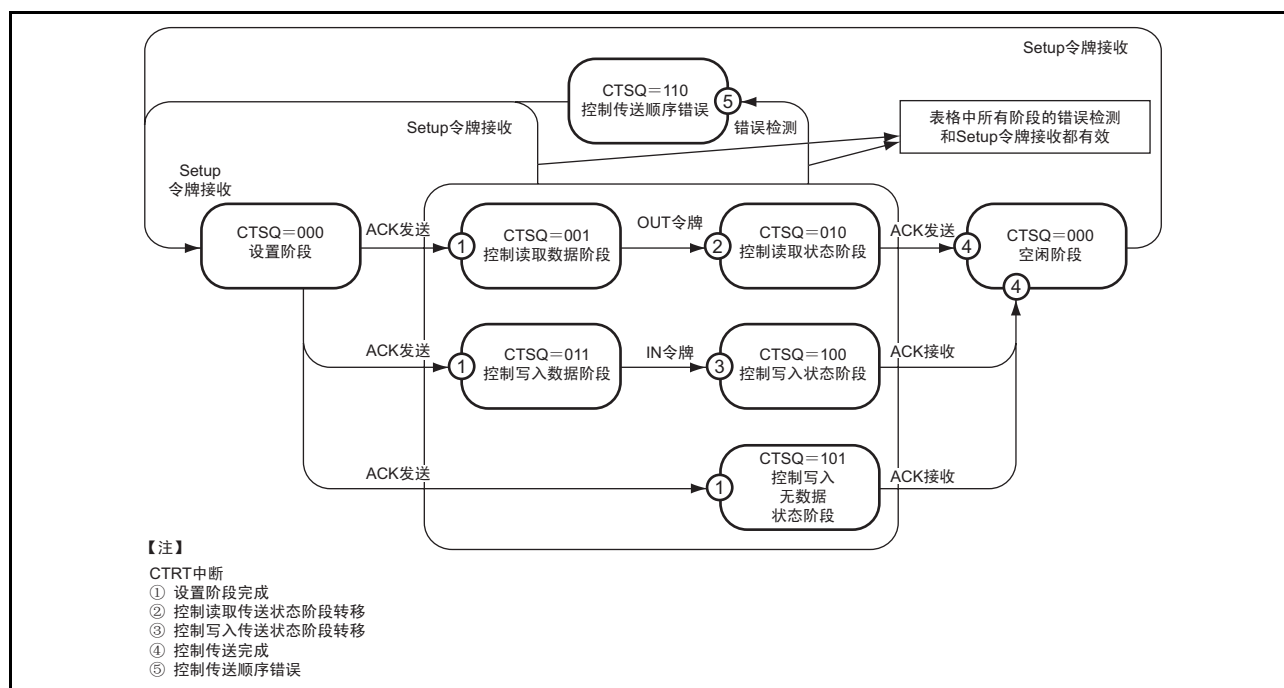


图 23.7 控制传送阶段转移图

(6) 帧更新中断

本模块的 SOFR 中断输出时序例如图 23.8 所示。

选择主机控制器功能时，在更新帧编号时序产生中断；选择功能控制器功能时，在更新帧编号后或检测出 SOF 包破损时，产生 SOFR 中断。必须由 FRMNUM 寄存器的 SOFRM 位指定中断运行。

但是，选择主机控制器功能时，不得设定 SOFRM=1。

1. 选择 SOFRM=0 时

在帧编号更新时序（约 1ms 的间隔）产生 SOFR 中断。SOF 包破损或脱落时，通过内部插值功能产生中断。高速通信过程中，在帧编号更新时序（约 1ms 的间隔）产生中断。

2. 选择 SOFRM=1 时

SOF 包破损或脱落时产生 SOFR 中断。在高速通信过程中，仅在有相同帧编号的 μ SOF 包的第一个包破损或脱落时产生中断。

（通过 SOF 插值功能识别 SOF 破损或脱落。详情参阅“23.4.9 SOF 插值功能”）。

选择功能控制器功能时，如果本模块在全速运行时检测出新的 SOF 包，则更新帧编号并产生 SOFR 中断。但是，如果高速运行中不为 μ SOF 锁定状态，则不更新帧编号且不产生 SOFR 中断。同时，SOF 的插值功能也不运行。所谓 μ SOF 锁定状态，即连续 2 次准确接收不同帧编号的 μ SOF 包。

μ SOF 锁定监控开始条件及停止条件如下所示：

1. μ SOF 锁定监控开始条件

USB_E=1

2. μ SOF 锁定监控停止条件

USB_E=0（软件复位）、接收 USB 总线复位或检测出挂起

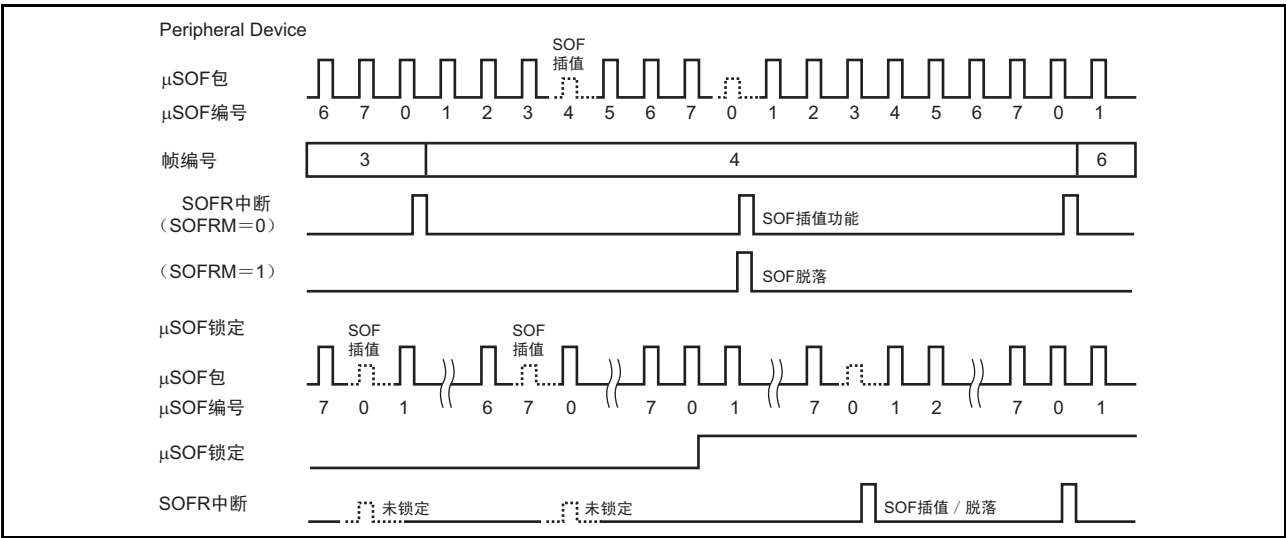


图 23.8 SOFR 中断输出时序的例子

(7) VBUS 中断

如果 VBUS 引脚发生变化则产生 VBUS 中断。可由 INTSTS0 寄存器的 VBSTS 位确认 VBUS 引脚的电平, 由 VBUS 中断确认主机控制器是否连接。在主机控制器连接的状态启动系统时, 因 VBUS 引脚不变, 所以不产生最初的 VBUS 中断。

(8) 恢复中断

选择功能控制器功能时, 器件状态为挂起、USB 总线状态 (J-State→K-State 或 J-State→SE0) 变化时, 产生恢复中断。从挂起状态的返回由此中断检测。

选择主机控制器功能时, 不产生恢复中断。必须使用 BCHG 中断检测 USB 总线的变化。

(9) BCHG 中断

USB 总线状态发生变化时, 产生 BCHG 中断。该中断用于检测是否连接功能器件 (选择主机控制器功能时) 和远程唤醒。选择主机控制器功能或功能控制器功能时, 均产生 BCHG 中断。

(10) DTCH 中断

选择主机控制器功能时, 如果在全速运行中检测出器件断开, 则产生 DTCH 中断。注意: 高速模式时不可使用该中断。在高速模式必须设定 DTCHE=0。在高速模式检测断开时, 需定期执行标准请求的控制传送, 及在无外围响应时需要断开和判断等处理。

必须使用 BCHG 中断检测挂起中的分离。

(11) SACK 中断

如果选择主机控制器功能, 对已发送的设置包在接收到外围的 ACK 响应时产生 SACK 中断。可由 SACK 中断确认设置处理是否正常结束。

(12) SIGN 中断

如果选择主机控制器功能, 对已发送的设置包在不可接收到外围的 ACK 响应时产生 SIGN 中断。该中断可用于检测外围未发送 ACK (无响应) 或 ACK 包破损。

23.4.3 管道控制

本模块的管道设定项目一览表如表 23.14 所示。USB 数据传送通过称为端点的逻辑管道执行数据通信。本模块有 8 个传送数据的管道。必须根据系统规格设定各管道。

表 23.14 管道设定项目一览表

寄存器名称	位名称	设定内容	备 注
DCPCFG PIPECFG	TYPE	指定传送类型	参阅 “23.4.3(1) 传送类型”。
	BFRE	选择 BRDY 中断模式	管道 1 ~ 5: 可设定
	DBLB	选择单缓冲器或双缓冲器	管道 1 ~ 5: 可设定
	CNTMD	选择连续传送或非连续传送	DCP: 可设定 管道 1、2: 可设定 (仅在选择批量传送时可设定) 管道 3 ~ 5: 可设定 连续发送 / 接收时设定缓冲器大小为负载的整数倍
	DIR	选择传送方向 (读取或写入)	可设定 IN 或 OUT
	EPNUM	端点编号	参阅 “23.4.3(2) 端点编号”。
	SHTNAK	选择管道禁止 (传送结束时)	管道 1、2: 可设定 (仅在选择批量传送时可设定) 管道 3 ~ 5: 可设定
PIPEBUF	BUFSIZE	缓冲存储器大小	DCP: 不可设定 (固定 256 字节) 管道 1 ~ 5: 可设定 (以 64 字节为单位最大可指定 2K 字节) 管道 6、7: 不可设定 (固定 64 字节)
	BUFNMB	缓冲存储器编号	DCP: 不可设定 (固定 H'0 ~ H'3 区域) 管道 1 ~ 5: 可设定 (可在 H'6 ~ H'7F 区域指定) 管道 6、7: 不可设定 (固定 H'4、H'5 区域)
DCPMAXP PIPEMAXP	MXPS	Max 包大小	参阅 “23.4.3(3) Max 包大小设定”。
PIPEPERI	IFIS	缓冲器清除	管道 1、2: 可设定 (仅在选择同步传送时) 管道 3 ~ 7: 不可设定
	IITV	间隔计数器	管道 1、2: 可设定 (仅在选择同步传送时) 管道 3 ~ 7: 不可设定
DCPCTR PIPEnCTR	BSTS	缓冲器状态	与 DIR/ISEL 位也有关系。
	INBUFM	IN 缓冲器监控	与 DIR/ISEL 位也有关系。
	ACLRM	自动缓冲器清除	设定缓冲存储器读取时, 可设定允许 / 禁止该功能。
	SQCLR	顺序清除	数据交替位清除
	SQSET	顺序设置	数据交替位置位
	SQMON	顺序确认	数据交替位确认
	PID	响应 PID	参阅 “23.4.3(4) 响应 PID”。

(1) 传送类型

由 PIPECFG 寄存器的 TYPE 位设定各管道的传送类型。各管道可设定的传送类型如下所示:

- DCP: 无须设定 (固定为控制传送)。
- 管道 1、2: 设定批量传送或同步传送。
- 管道 3~5: 设定批量传送。
- 管道 6、7: 设定中断传送。

(2) 端点编号

由 PIPECFG 寄存器的 EPNUM 位设定各管道的端点编号。DCP 固定为端点 0, 其他管道可设定为端点 1~端点 15。

- DCP: 无须设定 (固定为端点 0)。
- 管道 1~7: 选择 (1~15) 并设定。

不得使 DIR 位和 EPNUM 位的组合重复。

(3) Max 包大小设定

由 DCPMAXP 寄存器及 PIPEMAXP 寄存器的 MXPS 位设定各管道的 Max 包大小。DCP 及管道 1~5 设定为 USB 规格定义的所有 Max 包大小。管道 6、7 的 Max 包大小上限为 64 字节。必须在传送之前 (PID=BUF) 设定 Max 包大小。

- DCP: 高速运行时设定 64。
- DCP: 全速运行时, 选择 8、16、32 或 64 并设定。
- 管道 1~5: 全速批量传送时, 设定 0 或 512。
- 管道 1~5: 全速批量传送时, 选择 0、8、16、32 或 64 并设定。
- 管道 1、2: 高速同步传送时, 设定 1~1024 的值。
- 管道 1、2: 全速同步传送时, 设定 1~1023 的值。
- 管道 6、7: 设定 1~64 的值。

不支持中断传送及同步传送的 High Bandwidth。

(4) 响应 PID

由 DCPCTR 寄存器及 PIPEnCTR 寄存器的 PID 位设定各管道的响应 PID。
设定时, 本模块的运行如下所示:

1. 设定选择主机控制器功能时的响应 PID

响应 PID 指定处理的执行。

- NAK 设定: 管道禁止状态, 不执行处理。
- BUF 设定: 根据缓冲存储器的状态, 执行处理。
为 OUT 方向时, 如果缓冲存储器有发送数据, 则发送 OUT 令牌。
为 IN 方向时, 如果缓冲存储器有空间可接收数据, 则发送 IN 令牌。
- STALL 设定: 管道禁止状态, 不执行处理。

【注】 由 SUREQ 位设定 DCP 的设置处理。

2. 设定选择功能控制器功能时的响应PID

响应PID指定对主机处理的响应。

- NAK 设定：对产生的处理，总是执行 NAK 响应。
- BUF 设定：根据缓冲存储器的状态，响应处理。
- STALL 设定：对产生的处理，总是执行 STALL 响应。

【注】 对于设置处理，与 PID 设定无关，总是执行 ACK 响应，并将 USB 请求保存在寄存器。

根据处理结果，由本模块可能写入 PID 位。在以下情况，可由本模块写入 PID 位：

1. 选择主机控制器功能时，硬件设定响应PID时

- NAK 设定：在以下情况，PID=NAK，自动停止令牌发行。
除同步传送之外的传送，且对发送的令牌无响应时
发送的令牌接收到破损包时
在控制读取传送的数据阶段，接收到短包时
批量传送时，将 PIPECFG 寄存器的 SHTNAK 位设定为 1，接收到短包时
批量传送时，将 SHTNAK 位设定为 1，处理计数器结束时
- BUF 设定：本模块无 BUF 写入。
- STALL 设定：在以下情况，PID=STALL，自动停止令牌发行。
已发送的令牌接收到 STALL 时
已接收的数据包超过 Max 包大小时

2. 选择功能控制器功能时，硬件设定响应PID时

- NAK 设定：在以下情况，PID=NAK，且对处理总是进行 NAK 响应。
正常接收 SETUP 令牌时（仅 DCP）
批量传送时，将 PIPECFG 寄存器的 SHTNAK 位设定 1 时，处理计数器结束或接收到短包时
- BUF 设定：本模块无 BUF 写入。
- STALL 设定：在以下情况，PID=STALL，且对处理总是进行 STALL 响应。
检测出在接收数据包 Max 包大小溢出错误时
检测出控制传送顺序错误时

(5) 在 USB 通信允许 (PID=BUF) 状态禁止设定的寄存器

- CFIFOSEL 寄存器的 ISEL 位（仅选择 DCP 时）
- CFIFOSIE 寄存器的 TGL 位、SCLR 位
- DnFIFOSEL 寄存器的 DCLRM 位、TRENb 位、TRCLR 位、DEZPM 位
- DnFIFOTRN 寄存器的 TRNCNT 位
- DCPCFG 寄存器的各位
- DCPMAXP 寄存器的各位
- DCPCTR 寄存器的各位（除 CCPL 位之外）
- PIPECFG 寄存器的各位
- PIPEBUF 寄存器的各位
- PIPEMAXP 寄存器的各位
- PIPEPERI 寄存器的各位
- PIPEnCTR 寄存器的各位

(6) 数据 PID 顺序位

在控制传送的数据阶段、批量传送、中断传送正常执行数据传送时，数据 PID 的顺序位通过本模块自动交替运行。下一个被发送的数据 PID 的顺序位，可通过 DCPCTR 寄存器及 PIPEnCTR 寄存器的 SQMON 位确认。发送数据时，在 ACK 信号交换时序转换顺序位。接收数据时，在 ACK 信号交换发送时序转换顺序位。同时，由 DCPCTR 寄存器及 PIPEnCTR 寄存器的 SQCLR 位、SQSET 位，更改数据 PID 顺序位。

在选择功能控制器功能时的控制传送，阶段转移时，本模块自动设定顺序位。设置阶段结束时为 DATA0 响应，状态阶段为 DATA1 响应，因此，无须软件设定。在选择主机控制器功能时的控制传送，阶段转移时需由软件设定顺序位。

注意：无论选择主机控制器或是功能控制器功能，发送或接收 ClearFeature 请求等时，必须由软件设定数据 PID 顺序位。

另外，同步传送设定管道不可由 SQSET 位执行顺序位操作。

(7) 响应 PID=NAK 功能

通过将 PIPECFG 寄存器的位 SHTNAK 设定为 1，本模块可在接收到传送的最后一个（通过短包接收或处理计数器自动识别）数据包时，禁止管道运行（响应 PID=NAK）。

使用该功能，在缓冲存储器为双缓冲时，能以传送单位接收数据包。如果禁止管道运行，则需由软件重新设定管道允许（响应 PID=BUF）。

另外，本功能仅在批量传送时运行。

(8) 自动响应模式

在批量传送管道（管道 1～5），如果 PIPEnCTR 寄存器的 ATREPM 位置 1，则为自动响应模式。OUT 传送时（DIR=0）为 OUT-NAK 模式，IN 传送时（DIR=1）为 Null 自动响应模式。

• OUT-NAK 模式

在批量 OUT 传送管道，如果 ATREPM 位置 1，则对 OUT 令牌或 PING 令牌执行 NAK 响应，并输出 NRDY 中断。为了从普通模式转移至 OUT-NAK 模式，必须在管道运行禁止状态（响应 PID=NAK）设定 OUTNAK 模式后，并允许管道运行（响应 PID=BUF）。之后，OUT-NAK 模式有效。但是，在管道运行禁止前立即接收 OUT 令牌时，正常接收该令牌的数据，并对主机执行 ACK 响应。

为了从 OUT-NAK 模式转移至普通模式，必须在管道运行禁止状态（响应 PID=NAK）解除 OUT-NAK 模式，并允许管道运行（PID=BUF）。在普通模式可接收 OUT 数据，对于 PING 令牌，如果缓冲器可接收数据，则返回 ACK。

• Null 自动响应模式

在批量 IN 传送的管道，ATREPM 位置 1 时，连续发送 Zero-Length 包。

为了从普通模式转移至 Null 自动响应模式，必须在管道运行禁止状态（响应 PID=NAK）设定 Null 自动响应模式允许管道运行（响应 PID=BUF）。允许管道运行后，Null 自动响应模式有效。设定 Null 自动响应模式时，缓冲器必须为空，确认 INBUFM 位为 0。INBUFM 位为 1 时，缓冲器有数据，因此必须由 ACLRM 位清空缓冲器。设定 Null 自动响应模式时，不可从 FIFO 端口写入数据。

为了从 Null 自动响应模式转移至普通模式，必须在管道运行禁止状态（响应 PID=NAK）等待 Zero-Length 包发送期间（全速时：10μs，高速时：3μs）后，解除 Null 自动响应模式。在普通模式，可从 FIFO 端口写入数据，并可通过允许管道运行（响应 PID=BUF）向主机发送数据包。

23.4.4 缓冲存储器

(1) 缓冲存储器分配

本模块的缓冲存储器映射例如图 23.9 所示。缓冲存储器为 CPU 与本模块共用区域。缓冲存储器的存取权可能在系统 (CPU 侧) 也可能在本模块 (SIE 侧)。

缓冲存储器对各管道设定独立区域。以 64 字节为 1 块, 根据块起始编号和块数 (由 PIPEBUF 寄存器的 BUFNMB 位及 BUFSIZE 位指定) 设定存储区。对缓冲存储器的存取 (数据读取 / 写入) 使用 3 个 FIFO 端口。分配至 FIFO 端口的管道, 由 C/DnFIFOSEL 寄存器的 CURPIPE 位指定管道编号。

由 DCPCTR 寄存器及 PIPEnCTR 寄存器的 BSTS 位、INBUFM 位, 可确认各管道的缓冲器状态。并且, 由 C/DnFIFOCTR 寄存器的 FRDY 位, 可确认 FIFO 端口的存取权。

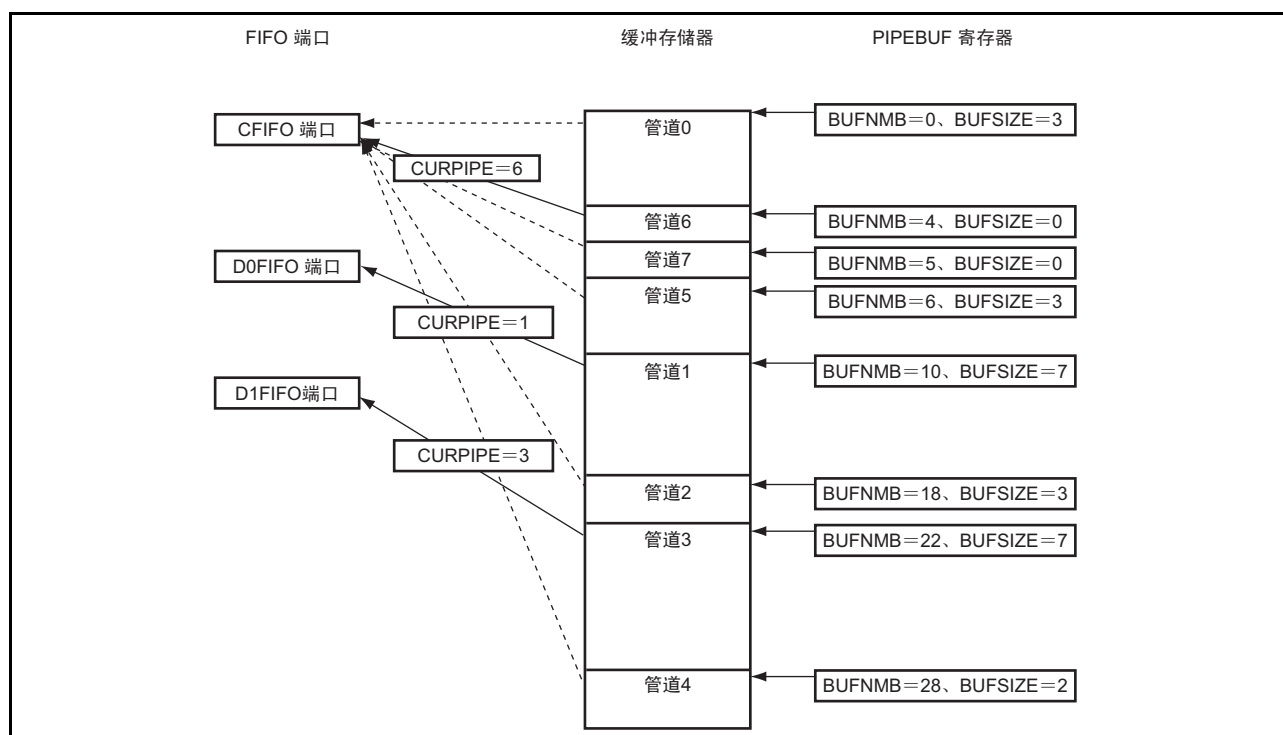


图 23.9 缓冲存储器映射例

(a) 缓冲器状态

本模块的缓冲器状态如表 23.15 及表 23.16 所示。由 DCPCTR.BSTS 位及 PIPEnCTR.INBUFM 位，可确认缓冲存储器状态。由 PIPEnCFG 寄存器的 DIR 位或 CFIFOSEL 寄存器的 ISEL 位（选择 DCP 时），指定缓冲存储器的存取方向。

INBUFM 位仅在发送方向的管道 1 ~ 5 有效。

发送的传送管道设定为双缓冲时，BSTS 位判断 CPU 的缓冲器状态；INBUFM 位判断 SIE 的缓冲器状态。此时，CPU（DMAC）对 FIFO 端口的写入延迟、切不可通过 BEMP 中断辨别缓冲器是否为空时，可由 INBUFM 位确认发送是否结束。

表 23.15 由 BSTS 位判断缓冲器状态表

ISEL 或 DIR	BSTS	缓冲存储器的状态
0（接收方向）	0	无接收数据，或正在接收 禁止从 FIFO 端口读取数据
0（接收方向）	1	有接收数据，或接收 Zero-Length 包 允许从 FIFO 端口读取数据 但是，接收 Zero-Length 包时不可读取，因此需清除缓冲器
1（发送方向）	0	未完成发送 禁止对 FIFO 端口写入数据
1（发送方向）	1	发送结束 可写入 CPU

表 23.16 由 INBUFM 位判断缓冲器状态表

DIR	INBUFM	缓冲存储器的状态
0（接收方向）	无效	无效
1（发送方向）	0	发送完可发送的数据 无可发送的数据
1（发送方向）	1	从 FIFO 端口写入可发送的数据， 有可发送的数据

(b) 缓冲器清除

通过本模块清除缓冲存储器如表 23.17 所示。由 BCLR、SCLR、DCLRM、ACLRM 位可清除缓冲存储器。

表 23.17 缓冲器清除一览表

位名称	BCLR	SCLR	DCLRM	ACLRM
寄存器	CFIFOCTR 寄存器 DnFIFOCTR 寄存器	CFIFOSIE 寄存器	DnFIFOSEL 寄存器	PIPEnCTR 寄存器
功能	清除 CPU 侧的缓冲存储器	清除 SIE 侧的缓冲存储器	读取指定管道的数据后， 自动清除缓冲存储器的模式。	废弃所有已接收的包的 自动缓冲器清除模式。
清除方法	通过写入 1 清除	通过写入 1 清除	1: 模式有效 0: 模式无效	1: 模式有效 0: 模式无效

(c) 缓冲区

本模块的缓冲存储器映射如表 23.18 所示。在缓冲存储器，有预先分配至管道的专用固定区域及用户可设定的用户区域。

DCP 的缓冲器为控制读取传送及控制写入传送时使用同一区域的专用固定区域。

预先分配管道 6、7 区域。不使用管道 6、7 时，分配至管道 1～5 作为用户区域。

此设定应确保各管道区域不重叠。必须注意：设定双缓冲时的区域是设定值的二倍。

不可用小于 Max 包大小的设定值指定缓冲器大小。

表 23.18 缓冲存储器映射

缓冲存储器编号	缓冲器大小	管道设定	备注
H'0 ～ H'3	256 字节	DCP 专用固定区域	单缓冲器、可连续传送
H'4	64 字节	管道 6 的固定区域	单缓冲器
H'5	64 字节	管道 7 的固定区域	单缓冲器
H'6 ～ H'7F	最大 7808 字节	管道 1 ～ 5 用户区域	可设定双缓冲器、可连续传送

(d) 自动缓冲器清除模式功能

本模块通过将 PIPEnCTR 寄存器的位 ACLRM 置 1，废弃所有已接收的数据包。如果接收正常的数据包，则 ACK 响应主机控制器。仅在缓冲存储器读取方向设定本功能。

如果 ACLRM 位先设定为 1 再设定为 0，可清除该管道的缓冲存储器，与存取方向无关。

作为硬件的内部顺序执行时间，对 ACLRM 位写入 1 与写入 0 的间隔至少需要 100ns。

(e) 缓冲存储器规格（单 / 双设定）

管道 1～5 可由 PIPEnCFG 寄存器的 DBLB 位选择单缓冲器或双缓冲器。双缓冲器具备对相同管道分配 2 个存储区（由 PIPEBUF 寄存器的 BUFSIZE 位指定）的功能。本模块的缓冲存储器设定例如图 23.10 所示。

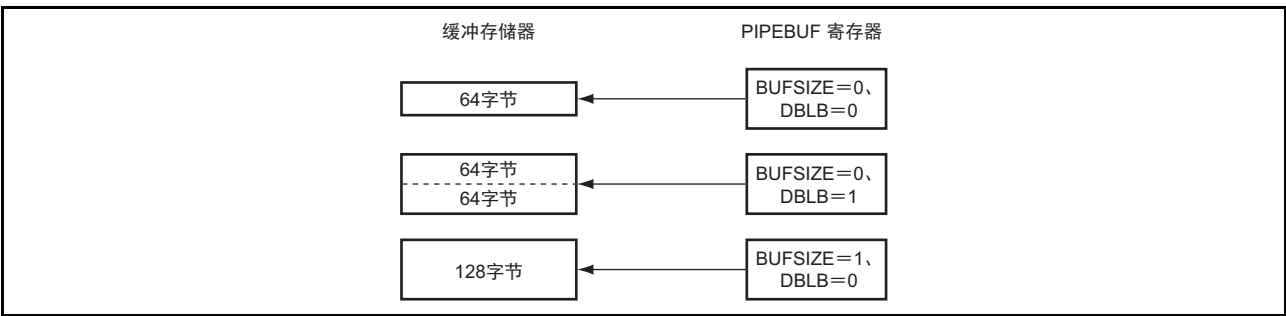


图 23.10 缓冲存储器设定例

(f) 缓冲存储器运行 (连续传送设定)

由 DCPCFG 寄存器及 PIPEnCFG 寄存器的 CNTMD 位可选择连续传送模式或非连续传送模式。该选择对管道 0 ~ 5 有效。

连续传送模式功能为连续发送 / 接收多个处理的功能。设定连续传送模式时, 可传送各管道所分配缓冲器大小的数据, 不对 CPU 产生中断。

在连续发送模式, 按照 Max 包大小分割写入数据后发送。发送小于缓冲器大小的数据 (短包或 Max 包大小的整数倍且小于缓冲器大小) 时, 写入发送数据后须设定 BVAL=1。

在连续接收模式, 接收相当于缓冲器大小的包、处理计数结束或接收短包前, 不产生中断。

本模块的缓冲存储器运行例如图 23.11 所示。

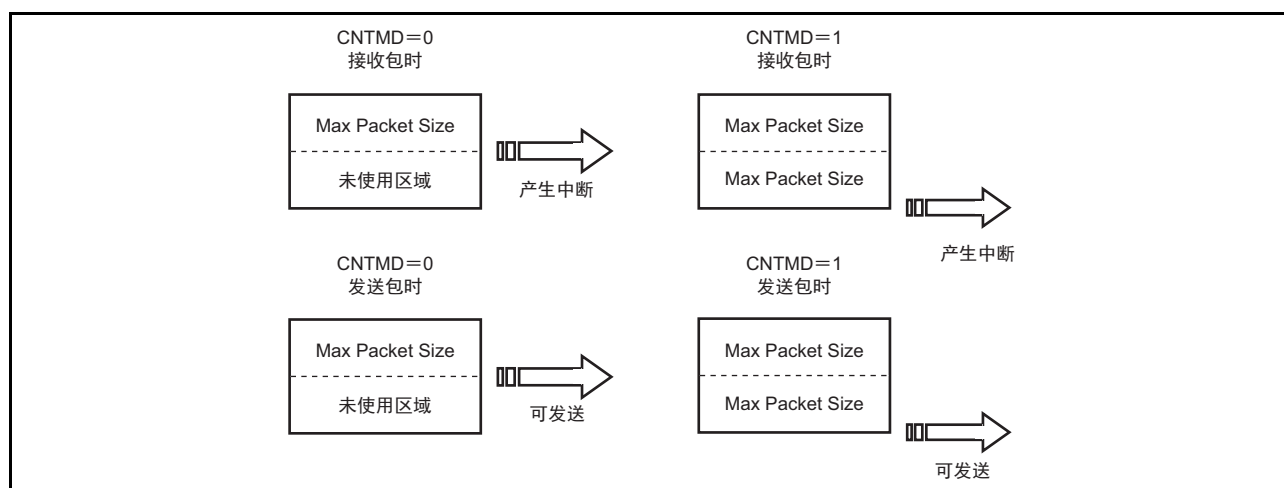


图 23.11 缓冲存储器运行例

(2) FIFO 端口功能

本模块的 FIFO 端口功能设定如表 23.19 所示。数据写入存取时, 如果执行写入直到缓冲器满 (非连续传送时为 Max 包大小), 则自动成为可发送状态。为了使并未达到缓冲器满 (非连续传送时为 Max 包大小) 的数据处于可发送状态时, 需由 C/DnFIFOCTR 寄存器的 BVAL 位设定写入结束。同时, Zero-Length 包的发送需由相同寄存器的 BCLR 位清除缓冲器后, 再由 BVAL 位设定写入结束。

读取存取时, 如果读取所有数据, 则自动成为可接收新包的状态。但接收 Zero-Length 包时 (DTLN=0), 无法读取数据, 因此需由相同寄存器的 BCLR 位清除缓冲器, 由 C/DnFIFOCTR 寄存器的 DTLN 位确认接收数据长度。

表 23.19 FIFO 端口功能设定表

寄存器名称	位名称	功 能	备注
C/DnFIFOSEL	REW	缓冲存储器回绕 (重新读取、重新写入)	
	DCLRM	读取指定管道的接收数据后自动清除	DnFIFO 专用
	DREQE	允许 DMA 传送	DnFIFO 专用
	MBW	FIFO 端口存取位宽度	
	TRENB	允许处理计数器运行	DnFIFO 专用
	TRCLR	清除当前处理次数	DnFIFO 专用
	DEZPM	Zero-Length 包附加模式	DMA 传送专用
	ISEL	FIFO 端口存取方向	DCP 专用
C/DnFIFOCTR	BVAL	缓冲存储器写入结束	
	BCLR*	清除 CPU 侧的缓冲存储器	
	DTLN	确认接收数据长度	
DnFIFOTRN	TRNCNT	设定接收处理计数	DnFIFO 专用
CFIFOSIE (除 DCP)	TGL	CPU/SIE 缓冲器交替	CFIFO 专用
	SCLR	清除 SIE 侧的缓冲存储器	CFIFO 专用

【注】 * 设定为 CFIFOSEL.CURPIPE=DCP 时, 如果设定为 CFIFOCTR.BCLR=1, 也可清除 SIE 侧的缓冲存储器。

(a) FIFO 端口选择

表 23.20 所示为各 FIFO 端口可选择的管道。由 C/DnFIFOSEL 寄存器的 CURPIPE 位选择存取的管道。选择管道后, 必须确认 FRDY=1, 再存取 FIFO 端口。

同时, 必须由 MBW 位选择存取的总线宽度。缓冲存储器存取方向取决于 PIPEnCFG 寄存器的 DIR 位。但是仅 DCP 由 ISEL 位决定。

表 23.20 各管道的 FIFO 端口存取表

管道	存取方法	可使用的端口
DCP	CPU 存取	CFIFO 端口寄存器
管道 1 ~ 7	CPU 存取	CFIFO 端口寄存器 D0FIFO/D1FIFO 端口寄存器
	DMA 存取	D0FIFO/D1FIFO 端口寄存器

(b) REW 位

可暂时中断当前的管道存取后，存取其他管道，之后再继续处理当前管道。该处理使用 C/DnFIFOSEL 寄存器的 REW 位。

如果在设定 C/DnFIFOSEL 寄存器的 CURPIPE 位的同时，将 REW 位设定为 1 选择管道，可复位缓冲存储器的读取 / 写入指针，并从第一个字节读取 / 写入。如果将 REW 位设定为 0 选择管道，可不复位缓冲存储器的读取 / 写入指针，而紧接上次的选择继续读取 / 写入数据。

存取 FIFO 端口时，选择管道后必须确认 FRDY=1。

(c) SIE 侧 的缓冲存储器的读取 (CFIFO 端口读取方向)

在 FRDY=0 的状态，即使无法从缓冲存储器读取数据，本模块通过确认 CFIFOSIE 寄存器的 SBUSY 位，并设定 TGL 位为 1，即可读取 SIE 的数据。设定 PID=NAK 并确认 SBUSY=0 后，写入 TGL=1，模块可从 CFIFO 寄存器读取数据。本功能仅限用于缓冲存储器读取方向，并由 TGL 位操作产生 BRDY 中断。

不得在以下状态对 TGL 位写入 1。

- 选择 DCP 时
- 读取缓冲存储器时
- 缓冲存储器写入方向的管道

(d) SIE 的缓冲存储器清除 (CFIFO 端口写入方向)

本模块通过确认 CFIFOSIE 寄存器的 SBUSY 位，并将 SCLR 位设定为 1，可取消准备发送的数据。

必须设定 PID=NAK 并确认 SBUSY=0 后，写入 SCLR=1。本模块可从 CFIFO 寄存器写入新数据。本功能仅限用于缓冲存储器写入方向，并由 SCLR 位操作产生 BRDY 中断。

不得在以下状态对 SCLR 位写入 1：

- 选择 DCP 时
- 正在写入缓冲存储器时
- 缓冲存储器读取方向的管道

(e) 处理计数器 (D0FIFO/D1FIFO 端口读取方向)

在数据包接收方向，且指定次数的处理已结束，本模块可识别传送结束。由 D0FIFO/D1FIFO 端口选择的管道被设定为从缓冲存储器读取数据方向时，处理计数器运行。处理计数器有指定处理次数的 TRNCNT 寄存器和在内部对处理进行计数的当前计数器，如果当前计数器与指定次数匹配，则缓冲存储器为可读取状态。由 TRCLR 位可初始化计数器功能的当前计数器并可重新处理计数。根据 TRENb 位的设定不同，TRNCNT 寄存器读取的信息也不同。

- TRENb=0：可读取已设定的处理计数器值。
- TRENb=1：可读取在内部计数的当前计数器值。

CURPIPE 位的更改条件如下：

- 指定管道的处理结束前，不得更改 CURPIPE 位。
- 未清除当前计数器时，不得更改 CURPIPE 位。

TRCLR 位的操作条件如下：

- 如果正在计数处理且 PID=BUF，则不可清除当前计数器。
- 在缓冲器残留数据的状态，不可清除当前计数器。

(f) FIFO 端口存取等待指定

向本模块的 FIFO 端口的存取，有如下规定：

- 传送速度最大不超过 48MB/sec

为了取消总线时钟的频率限制，本模块具有由存取等待指定 (FWAIT) 位控制存取周期的功能。
可在各 FIFO 端口设定 FWAIT 位，并可根据 CPU 速度及传送目标的存取周期等进行更有效的设定。
FWAIT 位的设定例：

- 条件
存取方向 : 写入 FIFO
总线时钟频率 : 66MHz
MBW 位设定值 : 10 (32 位宽度)
存取类型 : 从内部存储器 (源) 读取传送数据后，写入 FIFO 端口。此时源的存取需 2 个时钟周期。
- 计算例
 $(2 + (FWAIT + 2)) \times 1 / 66\text{MHz} \geq 1 / 48\text{MHz} \times 4$ (32 位)
FWAIT=2 (4 个时钟周期)

(g) 存取 FIFO 端口尾数数据的方法

读取 FIFO 端口的数据，当读取数据宽度小于 FIFO 端口选择寄存器的 MBW 位指定的位宽度时，必须以 MBW 位指定的位宽度读取数据，并由软件删除不需要的数据。

向 FIFO 端口写入数据，当写入数据宽度小于 FIFO 端口选择寄存器的 MBW 位指定的位宽度时，必须按照以下例存取。例如，将 FIFO 端口宽度设定为 32 位 (MBW=10) 时，写入 24 位数据。

- 写入尾数数据例 1：以 16 位宽度写入 1 次，以 8 位宽度写入 1 次

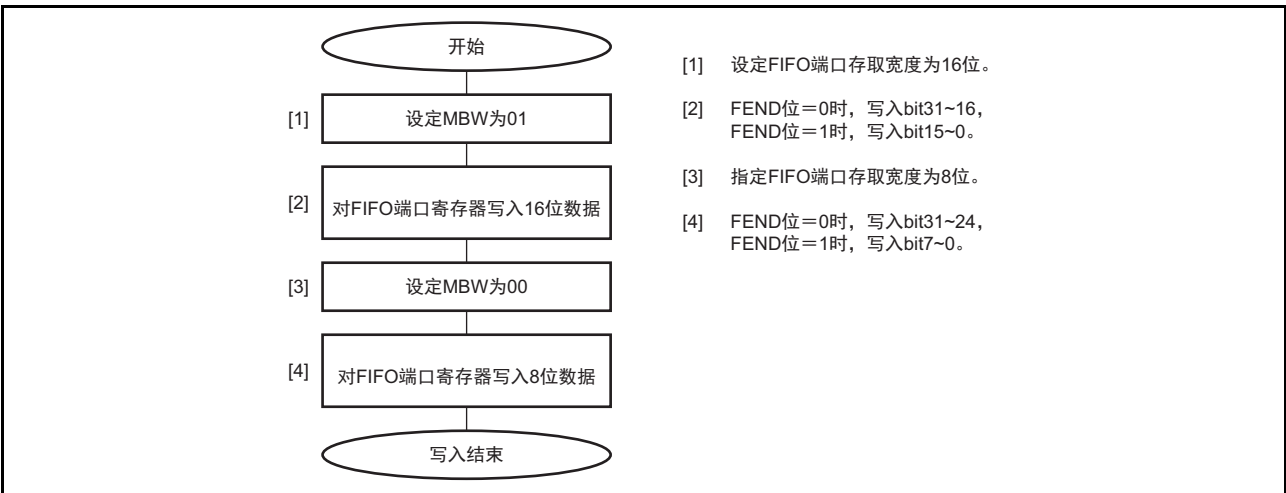


图 23.12 写入 FIFO 端口尾数数据例 1

- 写入尾数数据例2：以8位宽度写入3次

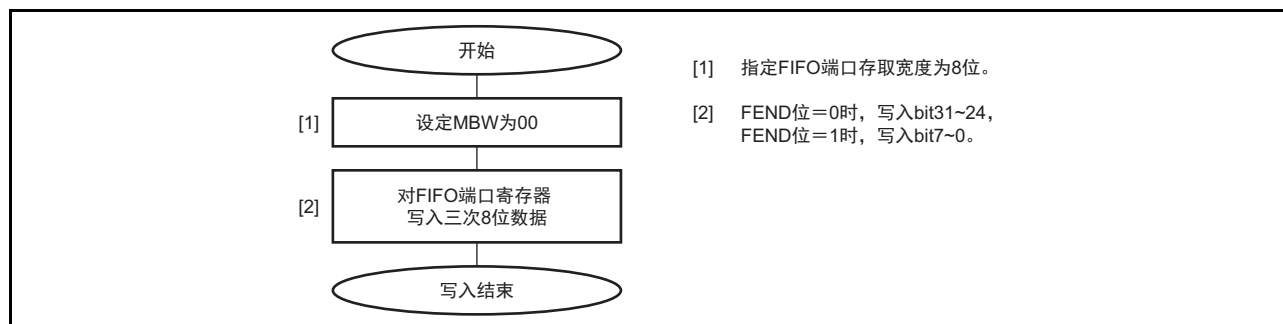


图 23.13 写入 FIFO 端口尾数数据例 2

(h) 选择 CURPIPE 在缓冲存储器读取方向时的 MBW 位的更改方法：

设定 CURPIPE 位的同时，必须写入 FIFO 端口选择寄存器（CFIFOSEL、D0FIFOSEL、D1FIFOSEL）的 MBW 位。由 CFIFO 寄存器设定 DCP 时（CURPIPE = 000），必须在设定 CURPIPE 或 ISEL 位的同时进行写入。

在正设定的管道更改 MBW 位时，请按以下步骤进行。但是一旦开始缓冲存储器的读取处理，则在所有数据读取结束前不得更改 MBW 位。

选择 CURPIPE 为缓冲存储器写入方向时，仅设定 MBW 即可更改。

但是一旦开始缓冲存储器的写入处理，则不得转换 8 位宽度～16 位宽度/32 位宽度或 16 位宽度～32 位宽度。

- DFIFO0、DFIFO1或CFIFO的CURPIPE不为DCP设定（000）时

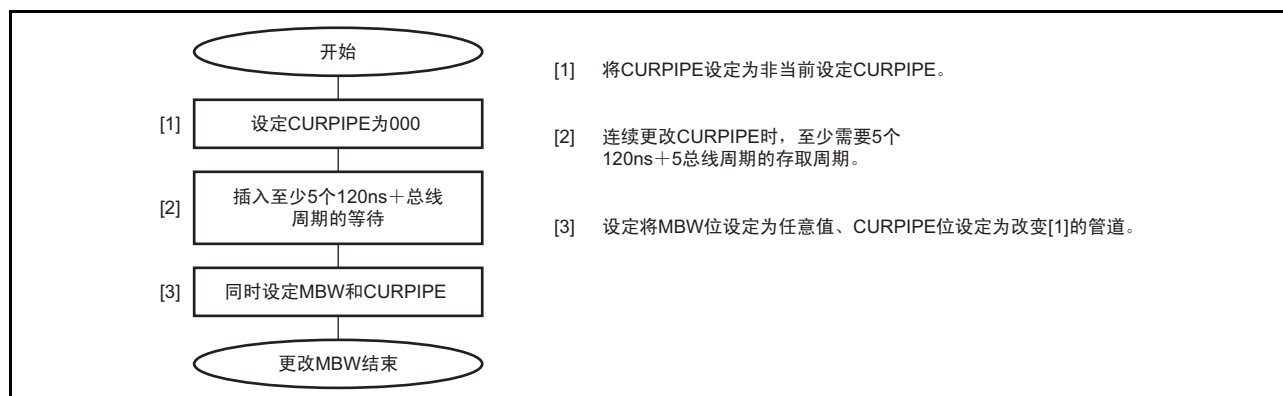


图 23.14 DFIFO0、DFIFO1 或 CFIFO 的 CURPIPE 不为 DCP 设定（000）时更改 MBW 的例子

(3) DMA 传送 (D0FIFO/D1FIFO 端口)

(a) DMA 传送概要

管道 1 ~ 7 可由 DMAC 存取 FIFO 端口。在 DMA 设定的管道的缓冲器可存取时, 输出 DMA 传送请求。

由 DnFIFOSEL 寄存器的 MBW 位, 选择向 FIFO 端口传送的单位, 由 CURPIPE 位, 选择 DMA 传送的管道。在 DMA 传送中不得更改所选管道。

(b) DMA 传送结束自动识别

通过控制 DMA 传送结束信号输入, 本模块可结束因 DMA 传送引起的 FIFO 数据写入。如果按直接存储器存取控制器 (DMAC) 的 DMA 传送计数器 (DMATCR) 设定的次数执行 DMA 传送, 则从 DMAC 输出 DMA 传送结束信号。采样 DMA 传送结束信号时, 使缓冲存储器进入可发送状态 (与设定 BVAL=1 相同的状态)。可由 DnFBCFG.TENDE 位设定是否采样 DMA 传送结束信号。

(c) Zero-Length 包附加模式 (D0FIFO/D1FIFO 端口写入方向)

将 DnFIFOSEL 寄存器的 DEZPM 位设定为 1, 采样 DMA 传送结束信号, 当写入缓冲存储器的数据的字节数为 Max 包大小的整数倍时, 发送所有数据后, 本模块可附加发送 1 个 Zero-Length 包。仅可在设定缓冲存储器写入方向时 (在 CURPIPE 位设定传送方向的管道时) 设定本功能。

(d) DnFIFO 自动清除模式 (D0FIFO/D1FIFO 端口读取方向)

将 DnFIFOSEL 寄存器的 DCLRM 位设定为 1, 在结束从缓冲存储器的数据读取时, 本模块自动清除该管道的缓冲存储器。

表 23.21 表示各设定中, 包接收和由软件引起缓冲存储器清除处理的关系。如表 23.21 所示, 根据 BFRE 位的设定值, 缓冲器清除条件不同。即使在必须清除的状态, 通过使用 DCLRM 位则无需由软件清除缓冲器, 可执行无需软件介入的 DMA 传送。

仅可在缓冲存储器读取方向设定本功能。

表 23.21 包接收和由软件引起缓冲存储器清除处理的关系表

寄存器设定 接收包时的缓冲器状态	DCLRM=0		DCLRM=1	
	BFRE=0	BFRE=1	BFRE=0	BFRE=1
缓冲器满	无需清除	无需清除	无需清除	无需清除
接收 Zero-Length 包	需要清除	需要清除	无需清除	无需清除
接收普通短包	无需清除	需要清除	无需清除	无需清除
处理计数结束	无需清除	需要清除	无需清除	无需清除

(e) BRDY 中断时序选择功能

设定 PIPECFG 寄存器的 BFRE 位，可在接收 Max 包大小的数据包时不产生 BRDY 中断。

使用 DMA 传送时，通过本功能仅在接收最终数据时产生中断。所谓最终数据，表示接收短包或处理计数结束。设定 BFRE=1 时，读取接收的数据后，产生 BRDY 中断。读取 DnFIFOCTR 寄存器的 DTLN 位，产生 BRDY 中断时，可确认最终接收的数据包的接收数据长度。

本模块的 BRDY 中断产生时序如表 23.22 所示。

表 23.22 BRDY 中断产生时序表

寄存器设定	BFRE=0	BFRE=1
接收包时的缓冲器状态		
缓冲器满（接收普通包）	接收包时	不产生
接收 Zero-Length 包	接收包时	接收包时
接收普通短包	接收包时	从缓冲存储器读取完接收数据时
处理计数结束	接收包时	从缓冲存储器读取完接收数据时

【注】 BFRE 位仅在从缓冲存储器读取方向有效。为写入方向时，BFRE 位必须固定为 0。

(4) FIFO 端口可存取时序

(a) 管道转换时的 FIFO 端口可存取时序

转换 FIFO 端口指定的管道（更改 C/DnFIFOSEL 寄存器的 CURPIPE 位）时，FRDY 位及 DTLN 位确定前的时序如图 23.15 所示。

更改 CURPIPE 位时，写入 C/DnFIFOSEL 寄存器后，等待 450ns+8 个总线时钟的时钟周期后，存取 FIFO 端口。

更改 ISEL 位时，CFIFO 端口也为相同时序。

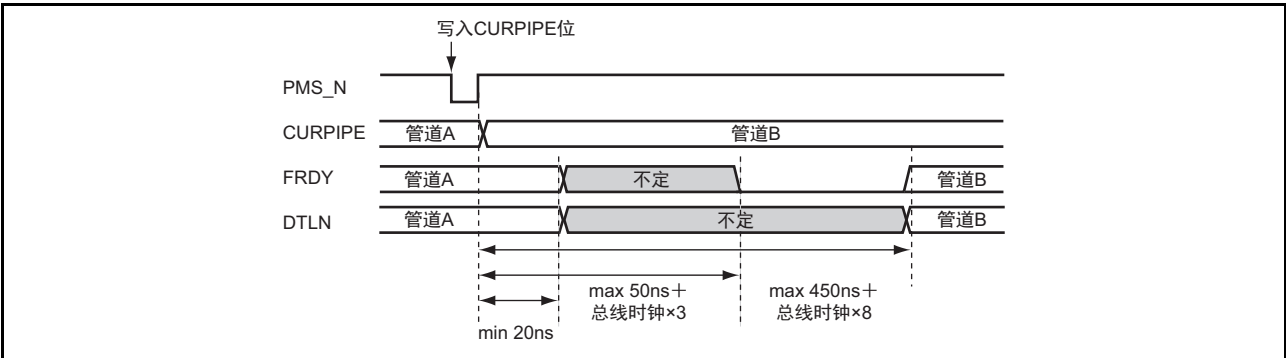


图 23.15 管道更改后的 FRDY、DTLN 的确定时序

(b) 双缓冲时时读取 / 写入结束后的 FIFO 端口可存取时序

对双缓冲器的管道，缓冲器读取 / 写入结束后，另一个缓冲器变为可存取状态前的时序如图 23.16 所示。使用双缓冲器时，请在交替前的存取后，等待 $300\text{ns}+6$ 个总线时钟的时钟周期，再存取 FIFO 端口。在 IN 方向的管道发送短包（通过设定 BVAL=1）时也为相同时序。

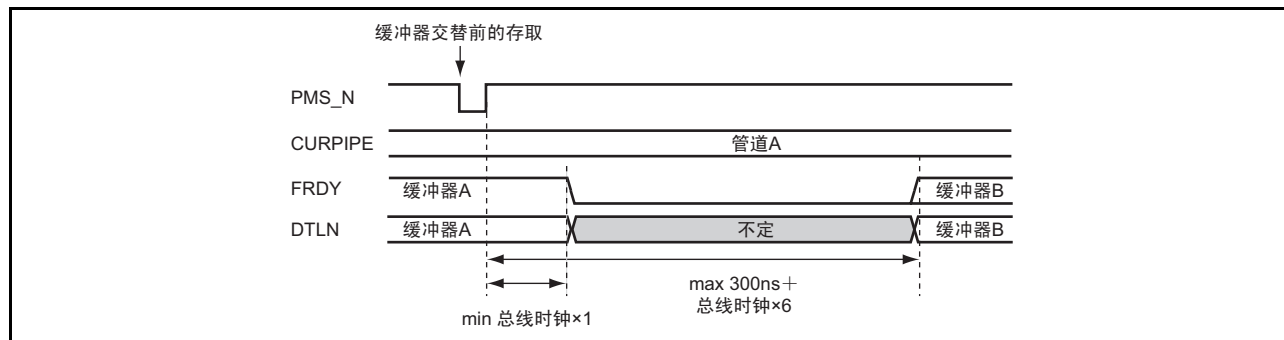


图 23.16 双缓冲器时的读取 / 写入结束后的 FRDY、DTLN 的确定时序

23.4.5 控制传送 (DCP)

控制传送的数据阶段的数据传送，使用默认控制管道 (DCP)。DCP 的缓冲存储器为控制读取及控制写入共用的固定区域，为 256 字节单缓冲器。仅 CFIFO 端口可存取缓冲存储器。

(1) 选择主机控制器功能时的控制传送

(a) 设置阶段

USBREQ、USBVAL、USBINDX 及 USBLENG 寄存器用于发送设置处理的 USB 请求。将设置包的数据写入寄存器，并给 DCPCTR 寄存器的 SUREQ 位写入 1，将设定数据作为设置处理发送。处理结束时，SUREQ 位清 0。SUREQ=1 时，不得运行上述 USB 请求寄存器。设置处理的器件地址由 DCPMAXP 寄存器的 DEVSEL 位指定。

发送处理时，根据外围的响应产生中断请求 (INTSTS1 寄存器的 SIGN 位及 SACK 位)。通过该中断请求，可确认设置处理结果。

总是发送 DATA0 的数据包 (USB 请求) 作为设置处理的数据包，与 DCPCTR 寄存器的 SQMON 位的内容无关。

(b) 数据阶段

使用 DCP 缓冲存储器传送数据。

存取 DCP 缓冲存储器时必须由 CFIFOSEL 寄存器的 ISEL 位指定方向。

数据阶段的第 1 个数据包必须是将数据 PID 作为 DATA1 进行通信。由 DCPCFG 寄存器的 SQSET 位将数据 PID 设定为 DATA1，并将 PID 位设定为 BUF 执行处理。通过 BRDY 及 BEMP 中断，检测数据传送结束。

通过指定连续传送，可执行多个包的数据传送。但是必须注意：在接收方向设定连续传送时，如果缓冲器满或不接收短包，则不会产生 BRDY 中断（为 Max 包大小的整数倍，且最多为 256 字节时）。

另外，控制写入传送时，如果发送数据为 Max 包大小的整数倍，则必须由软件控制在最后发送 Zero-Length 包。

如果为高速运行时的数据发送方向，则发送 PING 包。PING 包的控制与批量传送相同。

(c) 状态阶段

在本阶段传送与数据阶段相反方向的 Zero-Length 包的数据。与数据阶段相同, 使用 DCP 缓冲存储器传送数据。按照与数据阶段相同的步骤执行处理。

数据阶段的数据包必须是将数据 PID 作为 DATA1 进行通信。由 DCPCFG 寄存器的 SQSET 位将数据 PID 设定为 DATA1。

在产生 BRDY 中断后, 由 CFIFOCTR 寄存器的 DTLN 位确认接收数据长度并由 BCLR 位清除缓冲存储器后, 接收 Zero-Length 包。

如果为高速运行时的数据发送方向, 则发送 PING 包。PING 包的控制与批量传送相同。

(2) 选择功能控制器功能时的控制传送

(a) 设置阶段

对本模块来说属于正常的设置包时, 本模块必定对其进行 ACK 响应。设置阶段的本模块的运行如下所示:

1. 接收新设置包时, 本模块将以下位置位。
 - INTSTS0 寄存器的 VALID 位置 1
 - DCPCTR 寄存器的 PID 位设置为 NAK
 - DCPCTR 寄存器的 CCPL 位置 0
2. 在设置包后继续接收数据包, 本模块将 USB 请求的参数保存到 USBREQ、USBVAL、USBINDX 及 USBLENG 寄存器。

必须在设定 VALID=0 后, 处理控制传送的响应。VALID=1 时, 不可设定 PID=BUF, 且不可结束数据阶段。

通过 VALID 位的功能, 在控制传送过程中, 接收到新 USB 请求时, 本模块可中断正在处理的请求处理, 响应最新请求。

另外, 本模块自动辨别接收的 USB 请求的方向位 (bmRequestType 的 bit8) 及请求数据长度 (wLength), 并识别控制读取传送、控制写入传送及控制写入无数据传送, 以及管理阶段转移。对于错误顺序, 产生控制传送阶段转移中断的顺序错误, 并通知软件。有关本模块的阶段管理, 详情参照图 23.7。

(b) 数据阶段

必须使用 DCP 执行与接收到的 USB 请求对应的数据传送。存取 DCP 缓冲存储器前, 必须由 CFIFOSEL 寄存器的 ISEL 位指定存取方向。

传送数据大于 DCP 缓冲存储器的大小时, 必须在控制写入传送时使用 BRDY 中断、控制读取传送时使用 BEMP 中断传送数据。

高速运行时的控制写入传送时, 根据缓冲存储器的状态, 响应 NYET 信号交换。有关 NYET 信号交换, 详情参阅 “23.4.6(2) 选择功能控制器功能时的 NYET 信号交换控制”。

(c) 状态阶段

在 DCPCTR 寄存器的 PID 位为 PID=BUF 的状态, 将位 CCPL 置 1 结束控制传送。

通过上述设定, 按照在设置阶段确定的数据传送方向, 本模块自动执行状态阶段。具体如下:

- 控制读取传送时
接收 USB 主机的 Zero-Length 包, 发送 ACK 响应。
- 控制写入传送、无数据控制传送时
本模块发送 Zero-Length 包, 并接收 USB 主机的 ACK 响应。

(d) 控制传送自动响应功能

本模块自动响应正常的 SET_ADDRESS 请求。SET_ADDRESS 请求有以下错误时, 需由软件响应。

- 不为控制读取传送时: bmRequestType≠H'00
- 请求错误时: wIndex≠H'00
- 不为无数据控制传送时: wLength≠H'00
- 请求错误时: wValue > H'7F
- 器件状态错误的控制传送: DVSQ=011 (Configured)

除 SET_ADDRESS 之外的所有请求需由相应的软件响应。

23.4.6 批量传送 (管道 1 ~ 5)

批量传送可选择缓冲存储器的使用方法 (单 / 双缓冲器设定或连续 / 非连续传送模式设定)。缓冲存储器大小最大可设定到 2K 字节。本模块管理缓冲存储器的状态, 并自动响应 PING 包 / NYET 信号交换。设定 MXPS=0 时的中断规格与其他管道不同。详情参阅 “23.4.3(3) Max 包大小设定”。

(1) 选择主机控制器功能时的 PING 包控制

本模块自动发送 OUT 方向的 PING 包。

如下所示, 初始状态为 PING 包发送状态, 通过接收 ACK 信号交换发送 OUT 包。接收 NAK 或 NYET 时, 返回到 PING 发送状态。该控制在数据阶段、状态阶段也同样。

1. 设定 OUT 数据发送
2. 发送 PING 包
3. 接收 ACK 信号交换
4. 发送 OUT 数据包
5. 接收 ACK 信号交换
(重复步骤 4. 和 5.)
6. 发送 OUT 数据包
7. 接收 NAK/NYET 信号交换
8. 发送 PING 包

本控制器返回 PING 包发送的原因为上电复位、软件复位、NYET/NAK 信号交换接收、顺序交替位置位、清除 (SQSET、SQCLR) 及缓冲器清除设定 (ACLRM)。

(2) 选择功能控制器功能时的 NYET 信号交换控制

本模块的 NYET 信号交换响应如表 23.23 所示，NYET 响应遵从以下条件。但是接收短包时，不响应 NYET 包，而为 ACK 响应。控制写入传送的数据阶段也相同。

表 23.23 NYET 信号交换响应表

DCPCTR.PID 位的 设定值	缓冲存储器状态	令牌	响应	备 注
NAK/STALL	—	SETUP	ACK	—
	—	IN/OUT/PING	NAK/STALL	—
BUF	—	SETUP	ACK	—
	RCV-BRDY1	OUT/PING	ACK	接收 OUT 令牌时，接收数据包
	RCV-BRDY2	OUT	NYET	接收数据包、通知不可接收
	RCV-BRDY2	OUT (Short)	ACK	接收数据包、通知可接收
	RCV-BRDY2	PING	ACK	通知可接收
	RCV-NRDY	OUT/PING	NAK	通知不可接收
	TRN-BRDY	IN	DATA0/1	发送数据包
	TRN-NRDY	IN	NAK	TRN-NRDY

【符号说明】

RCV-BRDY1: 接收 OUT/PING 令牌时，缓冲存储器有不少于 2 个包的空白区

RCV-BRDY2: 接收 OUT 令牌时，缓冲存储器仅有 1 个包的空白区

RCV-NRDY: 接收 PING 令牌时，缓冲存储器无空白区

TRN-BRDY: 接收 IN 令牌时，缓冲存储器有发送数据

TRN-NRDY: 接收 IN 令牌时，缓冲存储器无发送数据

23.4.7 中断传送 (管道 6、7)

选择功能控制器功能时, 本模块按照主机控制器管理的周期进行中断传送。中断传送时, 忽略 PING 包 (无响应)。同时不发送 NYET 信号交换, 而响应 ACK、NAK、STALL。

选择主机控制器功能时, 可由间隔计数器设定令牌发送时序。即使为 OUT 方向的传送, 也不发送 PING 令牌而发送 OUT 令牌。如果从外围接收了 NYET 信号交换, 则为 ACK 接收运行。

另外, 本模块不支持中断传送的 High-Bandwidth 传送。

(1) 选择主机控制器功能时的中断传送的间隔计数器

执行中断传送时, 对 PIPEPERI 寄存器的 IITV 位设定处理间隔。本控制器按照设定的间隔发送中断传送的令牌。

(a) 计数器的初始化

本控制器初始化间隔计数器的条件如下:

- 上电复位
初始化 IITV 位。
- 软件复位
初始化 IITV 位。
- 由 ACLRM 位初始化缓冲存储器
不初始化 IITV 位但初始化计数。通过将 ACLRM 位置 0 从最初计数 IITV 的设定值。

必须注意: 在以下情况不初始化间隔计数器。

- USB 总线复位、USB 挂起
不初始化 IITV 位。通过将 UACT 位置 1, 从进入 USB 总线复位或 USB 挂起状态前的值开始计数。

(b) 在令牌的产生时序的不可发送 / 接收时的运行

以下情况时, 即使在令牌的产生时序也不产生令牌。此时在下一间隔尝试执行处理。

- 将 PID 设定为 NAK 或 STALL 时
- IN 方向 (接收) 的传送中, 在令牌的发送时序缓冲存储器无空白区域时
- OUT 方向 (发送) 的传送中, 在令牌的发送时序缓冲存储器无发送数据时

23.4.8 同步传送 (管道 1、2)

本模块对同步传送有以下功能:

- 通知同步传送错误信息
- 间隔计数器 (IITV 位指定)
- 同步 IN 传送数据设置控制 (IDLY 功能)
- 同步 IN 传送缓冲器清除功能 (IFIS 位指定)

本模块不支持同步传送的 High-Bandwidth 传送。

【注】 使用同步 OUT 传送时, 参阅 “23.5.1 使用同步 OUT 传送时的注意事项”。

(1) 同步传送的错误检测

为了使软件可管理同步传送错误, 本模块有以下错误信息检测功能。有关伴随本模块检测错误的优先顺序及错误检测产生的中断, 如表 23.24 及表 23.25 所示。

1. PID 错误
 - 接收包的 PID 错误时
2. CRC 错误、位填充错误
 - 接收包的 CRC 有错误或位填充错误时
3. 超出 Max 包大小
 - 接收包的数据大小超过 Max 包大小的设定值
4. 上溢、下溢错误
 - 选择主机控制器功能时
IN 方向 (接收) 传送时, 在令牌的发送时序缓冲存储器无空白区域时
OUT 方向 (发送) 传送时, 在令牌的发送时序缓冲存储器无数据时
 - 选择功能控制器功能时
IN 方向 (发送) 传送时, 接收 IN 令牌时缓冲存储器无数据时
OUT 方向 (接收) 传送时, 虽接收 OUT 令牌但缓冲存储器无空白区域时
5. 间隔错误

选择功能控制器功能时, 以下为间隔错误。

 - 同步 IN 传送时, 在间隔帧不可接收 IN 令牌时
 - 同步 OUT 传送时, 除间隔帧之外, 接收 OUT 令牌时

表 23.24 接收令牌时的错误检测

检测的优先顺序	错误	产生的中断和状态
1	PID 错误	选择主机 / 功能控制器功能时, 均不产生中断 (忽略为破损包)。
2	CRC 错误、位填充错误	选择主机 / 功能控制器功能时, 均不产生中断 (忽略为破损包)。
3	上溢、下溢错误	选择主机 / 功能控制器功能时, 均产生 NRDY 中断, 并将 OVRN 位置位。选择主机控制器功能时, 不发送令牌。 选择功能控制器功能时, 对 IN 令牌发送 Zero-Length 包。对 OUT 令牌, 不接收数据包。
4	间隔错误	选择功能控制器功能时, 产生 NRDY 中断。选择主机控制器功能时不产生。

表 23.25 接收数据包时的错误检测

检测的优先顺序	错误	产生的中断和状态
1	PID 错误	不产生中断 (忽略为破损包)。
2	CRC 错误、位填充错误	选择主机 / 功能控制器功能时, 均产生 NRDY 中断, 并置位 CRCE 位。
3	Max 包大小超出错误	选择主机 / 功能控制器功能时, 均产生 BEMP 中断, 并将 PID 设置为 STALL。

(2) DATA-PID

本模块不支持 High-Bandwidth 传送。选择功能控制器功能时, 对接收的 PID 做如下处理:

1. IN 方向

- DATA0: 作为数据包的PID发送。
- DATA1: 不发送。
- DATA2: 不发送。
- mData: 不发送。

2. OUT 方向 (全速运行时)

- DATA0: 作为数据包的PID正常接收。
- DATA1: 作为数据包的PID正常接收。
- DATA2: 忽略包。
- mData: 忽略包。

3. OUT 方向 (高速运行时)

- DATA0: 作为数据包的PID正常接收。
- DATA1: 作为数据包的PID正常接收。
- DATA2: 作为数据包的PID正常接收。
- mData: 作为数据包的PID正常接收。

(3) 间隔计数器

由 PIPEPERI 寄存器的 IITV 位，可设定同步传送的间隔。选择功能控制器功能时，间隔计数器可实现表 23.26 所示的功能。选择主机控制器功能时，生成令牌的发送时序。选择主机控制器功能时，间隔计数器的运行与中断传送运行相同。

表 23.26 选择功能控制器功能时的间隔计数器功能

传送方向	功能	检测条件
IN	发送缓冲器清除功能	同步 IN 传送中，在间隔帧不可正常接收 IN 令牌。
OUT	通知未接收令牌	同步 OUT 传送中，在间隔帧不可正常接收 OUT 令牌。

由接收 SOF 或插值 SOF 时进行间隔计数，即使 SOF 损坏也可保持同步性。可设定的帧间隔为 2^{IITV} 帧或 $2^{IITV}\mu$ 帧。

(a) 选择功能控制器功能时计数器的初始化

在以下条件，本模块初始化间隔计数器。

- 上电复位
初始化 IITV 位。
- 软件复位
初始化 IITV 位。
- USB 总线复位
不初始化 IITV 位，但是初始化计数值。
- 由 ACLRM 初始化缓冲存储器
虽然不初始化 IITV 位，但是初始化计数值。通过将 ACLRM 位置 0，从 IITV 的设定值开始计数。

间隔计数器初始化后，正常传送信息包后，按以下条件 1. 或 2. 开始间隔计数。

1. 在 PID=BUF 的状态，对 IN 令牌发送数据后接收 SOF
2. 在 PID=BUF 的状态，接收 OUT 令牌的数据后接收 SOF

另外，以下条件，不可初始化间隔计数器。

1. 设定 PID 位为 NAK 或 STALL 时
不停止间隔定时器。尝试在下一间隔执行处理。
2. USB 挂起
不初始化 IITV 位。接收 SOF 后，从接收前的值开始计数。

(4) 选择功能控制器功能时的同步传送发送数据设置

选择功能控制器功能，在本模块的同步数据发送时，对缓冲存储器写入数据后，在检测出 SOF 包的下一个帧发送数据包。本功能称为同步传送发送数据设置功能。通过本功能可指定开始发送的帧。

缓冲存储器用于双缓冲器时，即使两个缓冲器的写入都结束，可传送状态的缓冲存储器也仅是先结束写入的那个缓冲器。因此在同一帧即使接收到多个 IN 令牌，发送的缓冲存储器也仅有 1 个包大小。

接收 IN 令牌时，如果缓冲存储器为可发送状态，则传送数据并正常响应。如果缓冲存储器为不可发送状态，则发送 Zero-Length 包，为下溢错误。

在本模块设定 IITV=0（每帧）时，通过使用同步传送发送数据设置功能发送的例子如图 23.17 所示。图中阴影 Null 表示发送 Zero-Length 包。

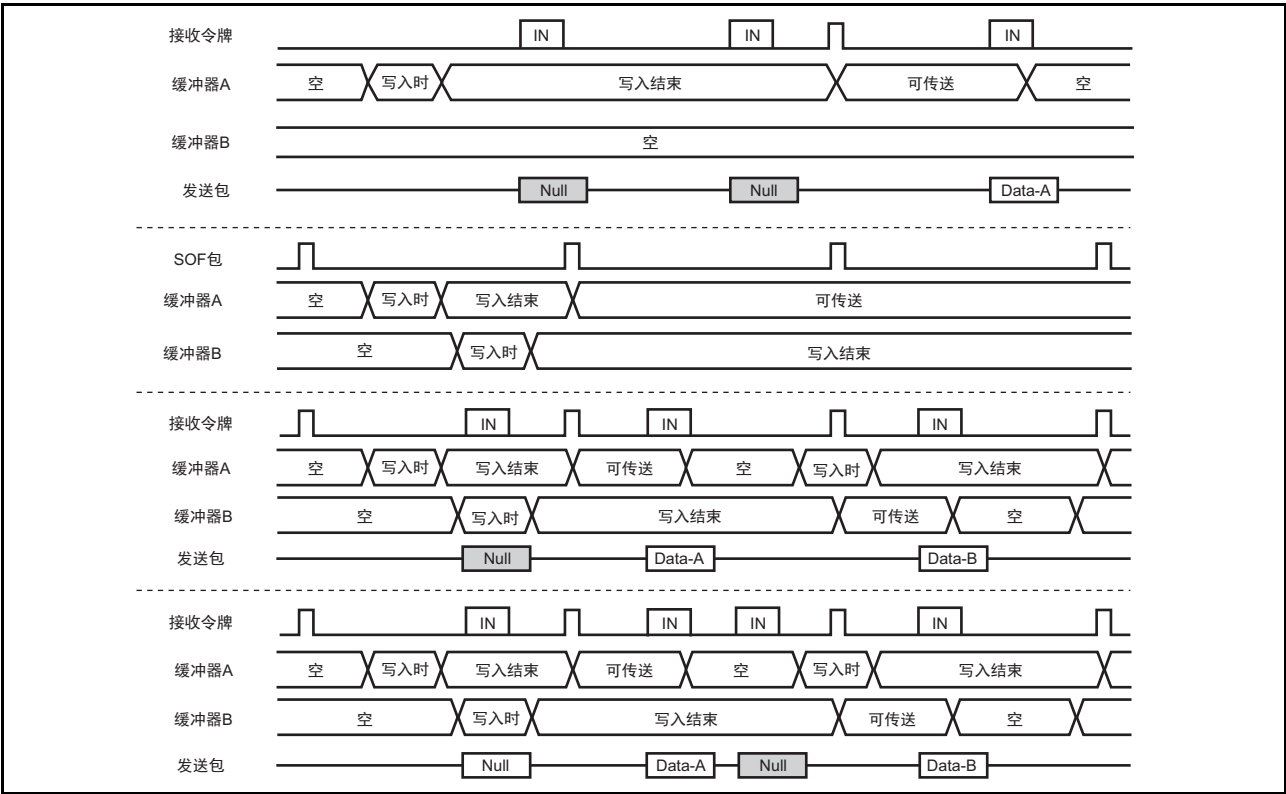


图 23.17 数据设置功能运行例

(5) 选择功能控制器功能时的同步传送发送缓冲器清除

选择功能控制器功能时，本模块不在同步数据发送时，在间隔帧接收 IN 令牌，如果接收下一个帧的 SOF 或 μ SOF 包，作为 IN 令牌破损处理并清除可发送状态的缓冲器，使该缓冲器进入可写入状态。

此时，缓冲存储器为双缓冲器，两个缓冲器的写入都结束时，将废弃的缓冲存储器视为在相同间隔帧发送的数据，SOF 或 μ SOF 包接收时，没有废弃的缓冲存储器为可传送状态。

因 IITV 位设定值不同，缓冲器清除功能的运行时序不同。

- IITV=0 时
从管道有效后的下一帧开始运行缓冲器清除。
- 除 IITV=0 之外时
在第一个正常处理后运行缓冲器清除。

本模块的缓冲器清除功能运行例如图 23.18 所示。但在设定的间隔以外（间隔帧之前的令牌），根据数据设置状态，作为写入数据或下溢错误发送 Zero-Length 包。



图 23.18 缓冲器清除功能运行例

本模块产生间隔错误例如图 23.19 所示。间隔错误有以下 5 种。在图中①的时序，产生间隔错误并运行缓冲器清除功能。

IN 传送时，如果发生间隔错误，则运行缓冲器清除功能。OUT 传送时，如果发生间隔错误，则产生 NRDY 中断。

必须由 OVRN 位判断 NRDY 中断（接收包错误等）和溢出错误的区别。
根据缓冲存储器状态，响应图中阴影部分的令牌。

1. IN 方向
 - 如果为缓冲器可传送状态，则传送数据并正常响应
 - 如果为缓冲器禁止传送状态，则发送 Zero-Length 包并产生下溢错误
2. OUT 方向
 - 如果为缓冲器可接收状态，则接收数据并正常响应
 - 如果为缓冲器禁止接收状态，则废弃数据并产生上溢错误

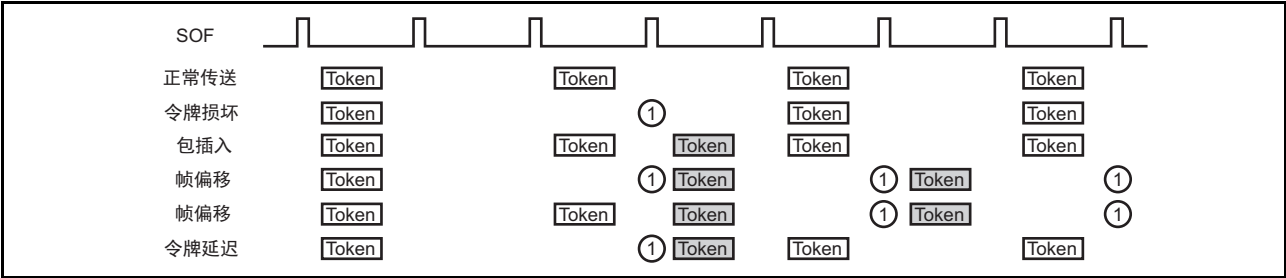


图 23.19 IITV=1 时的间隔错误产生例

23.4.9 SOF 插值功能

选择功能控制器功能时，由于 SOF 包破损或脱落，在 1ms（全速运行时）或 125μs（高速运行时）的间隔不可接收 SOF 包时，本模块插值 SOF。SYSCFG.USB=1 且接收 SOF 包时，运行 SOF 插值功能。同时，在以下条件初始化插值功能。

- 上电复位
- 软件复位
- USB 总线复位
- 挂起检测

同时，SOF 插值按照以下规格运行。

- 帧间隔（125μs/1ms）遵从复位信号交换协议的结果。
- 接收 SOF 包前，不运行插值功能。
- 接收第一个 SOF 包后，以 USB 用时钟的 48MHz 计数 125μs 或 1ms 并插值。
- 接收第 2 次之后的 SOF 包后，使用上次的接收间隔进行插值。
- 挂起状态及接收 USB 总线复位时，不执行插值。
（在高速运行的挂起转移时，从接收到最终包开始 3ms 后继续插值）

本模块以 SOF 包的接收为基础，运行以下功能。SOF 包脱落时会执行 SOF 插值，因此可继续正常运行。

- 帧编号及微帧编号的更新
- SOFR 中断时序及 μSOF 锁定
- 同步传送间隔计数

如果全速运行时 SOF 包脱落，则不更新 FRMNUM 寄存器的 FRNM 位。如果高速运行时 μSOF 包脱落，则更新 UFRMNUM 寄存器的 UFRNM 位。但是，如果 UFRNM=000 的 μSOF 包脱落，则不更新 FRNM 位。此时，即使继续正常接收 UFRNM=000 以外的 μSOF 包，也不更新 FRNM 位。

23.4.10 管道安排

(1) 处理发送条件

选择主机控制器功能时，设定 UACT=1 后，本模块按照表 23.27 所示的条件发送处理。

表 23.27 处理发送条件

处理	发送条件				
	DIR	PID	IITV0	缓冲器的状态	SUREQ
设置	—*1	—*1	—*1	—*1	1 设定
控制传送的数据阶段、 状态阶段、批量传送	IN	BUF	无效	有接收区域	—*1
	OUT	BUF	无效	有发送数据	—*1
中断传送	IN	BUF	有效	有接收区域	—*1
	OUT	BUF	有效	有发送数据	—*1
同步传送	IN	BUF	有效	*2	—*1
	OUT	BUF	有效	*3	—*1

【注】 *1 表中的“—”，表示与令牌发送无关的条件。有效表示在中断传送和同步传送中，仅根据间隔计数器的传送帧产生处理。无效表示条件的产生与间隔计数器无关。

*2 产生处理与有无接收区域无关。但是，无接收区域时，废弃接收数据。

*3 产生处理与有无发送数据无关。但是，无发送数据时，发送 Zero-Length 包。

(2) 传送安排

以下说明本模块帧内的传送安排。发送 SOF 后，本模块按照以下顺序传送。

- 1. 执行周期传送
按管道 1→管道 2→管道 6→管道 7 的顺序搜索，如果有可产生同步传送或中断传送处理的管道，则发送处理。
- 2. 控制传送的设置处理
确认 DCP，如果可设置处理，则发送。
- 3. 批量、控制传送数据阶段、状态阶段的执行
按照 DCP→管道 1→管道 2→管道 3→管道 4→管道 5 的顺序搜索管道。如果为可能产生批量 / 控制
传
送数据阶段、控制传送状态阶段处理的管道，则执行处理。
发送处理时，外围的响应无论是 ACK 还是 NAK，均转移至下一个管道处理。同时，如果帧内有执行
传
送的时间，则重复步骤 3。

(3) USB 通信允许

通过将 DVSTCTR 寄存器的 UACT 位置 1，开始 SOF 或 μ SOF 的发送，并可发送处理。

设定 UACT 位为 0 时，停止 SOF 或 μ SOF 的发送并进入挂起状态。如果设定 UACT 位为 1→0，则发送下一个 SOF 或 μ SOF 后停止处理。

23.5 使用时的注意事项

23.5.1 使用同步 OUT 传送时的注意事项

符合以下条件“①和②”时，请使用下表的管道设定，执行同步（以下 ISO）-OUT 传送。

- ① 在主机模式时使用
- ② 全速通信时

【注】 选择主机模式的高速传送或使用功能模式时（含全速通信时），本注意事项不适用。

	PIPE1	PIPE2	PIPE6
ISO-OUT 传送中使用 PIPE1 时	ISO-OUT	未使用或 BULK-IN/OUT	未使用或 INT-IN/OUT
ISO-OUT 传送中使用 PIPE2 时	未使用或 ISO-IN 或 BULK-IN/OUT	ISO-OUT	未使用

【注】 不可同时在 PIPE1、PIPE2（2 个管道）执行 ISO-OUT 传送。需在 2 个管道执行 ISO-OUT 传送时，必须使用高速通信。

23.5.2 USB 收发器的设定步骤

使用本模块前，需预先设定内部 USB 收发器。设定方法如下所示。以下步骤的程序例如图 23.20 所示。

(1) 对 USBAC 特性转换寄存器 (USBACSWR) 的 UACS23 位写入 1。

(2) 对测试模式寄存器 (TESTMODE) 的 HOSTPCC 位写入 1。

为了防止该位被错误改写，可通过特殊顺序写入，写入顺序如下所示：

1. 对器件状态控制寄存器 (DVSTCTR) 的 UACKEY0、UACKEY1 位写入 1。
2. 对测试模式寄存器 (TESTMODE) 的 HOSTPCC 位写入 1。
3. 对器件状态控制寄存器 (DVSTCTR) 的 UACKEY0、UACKEY1 位写入 0

```

;初始化程序
;设定USBE=1
    MOVI20  #H'FFFC1C00, R0
    MOV.W   #H'0001, R1
    MOV.W   R1, @R0

;(1) 设定UACS23=1
    MOVI20  #H'FFFC1C84, R0
    MOV.L   #H'00800000, R1
    MOV.L   R1, @R0

;(2) 设定HOSTPCC=1
;1. UACKEY0、UACKEY1=1
    MOVI20  #H'FFFC1C04, R0
    MOV.W   #H'9000, R1
    MOV.W   R1, @R0
;2. HOSTPCC=1
    MOVI20  #H'FFFC1C06, R0
    MOV.W   #H'8000, R1
    MOV.W   R1, @R0
;3. UACKEY0、UACKEY1=0
    MOVI20  #H'FFFC1C04, R0
    MOV.W   #H'0000, R1
    MOV.W   R1, @R0
    .
    .
    .
    .

```

图 23.20 USB 收发器的设定步骤

23.5.3 中断源的清除时序

必须通过中断异常服务程序清除中断源标志。从清除中断源标志到实际撤消对 CPU 的中断源需一定的时间。因此，为了防止再次错误接收应清除的中断源，需在清除后读取 3 次中断源标志，之后执行 RTE 指令。

第 24 章 LCD 控制器 (LCDC)

LCD 控制器 (LCDC) 采用可将显示的图像保存至系统存储器的统一存储器体系结构。LCDC 模块从系统存储器读取数据, 并使用调色板存储器决定颜色后, 传送至 LCD 面板。可连接除单片机总线接口方式、NTSC/PAL 方式、LVDS 接口的液晶模块 * 之外的液晶模块。

【注】 * 通过连接 LVDS 转换 LSI, 可连接 LVDS 接口。

24.1 特点

LCDC 有以下特点:

- 面板接口
串行接口方式
支持 STN/Dual STN/TFT 面板 (8/12/16/18 位总线宽度) 的数据格式 *1
- 支持 4/8/15/16bpp (位每像素) 彩色模式
- 支持 1/2/4/6bpp 灰度模式
- 支持 16×1 ~ 1024×1024 的液晶面板 *2
- 24 位彩色调色板存储器 (24 位中 16 位有效 R:5/G:6/B:5)
- 通过 RGB 的 8 位、24 位的空间调制 FRC, 可在容易产生闪烁、阴影的 STN/DSTN 面板实现闪烁较少的 65536 彩色控制
- 连接 CPU 的同步 DRAM (区域 3) 的一部分作为保存 LCDC 显示数据的 VRAM 使用, 无需显示专用的存储器
- 通过 2.4K 字节的大尺寸行缓冲器, 实现稳定显示
- 支持适合液晶面板信号极性的输出信号、输出信号的电平反转功能
- 支持由寄存器选择各种数据格式 (字节内的字节序设定、反像素方式)
- 可在用户指定的位置产生中断 (通过控制 VRAM 的更新开始时序, 避免闪烁)
- 支持硬件旋转模式 (可在横向液晶面板支持纵向液晶面板) (旋转前的宽度不可超过 320 像素, 详情参照表 24.5)

【注】 *1 连接 18 位总线宽度的 TFT 面板时, 未连接的线的低位信号必须连接 GND 或连接输出数据的最低位。

*2 详情参阅 “24.4.1 关于 LCDC 可显示的液晶模块尺寸”。

LCDC 框图如图 24.1 所示:

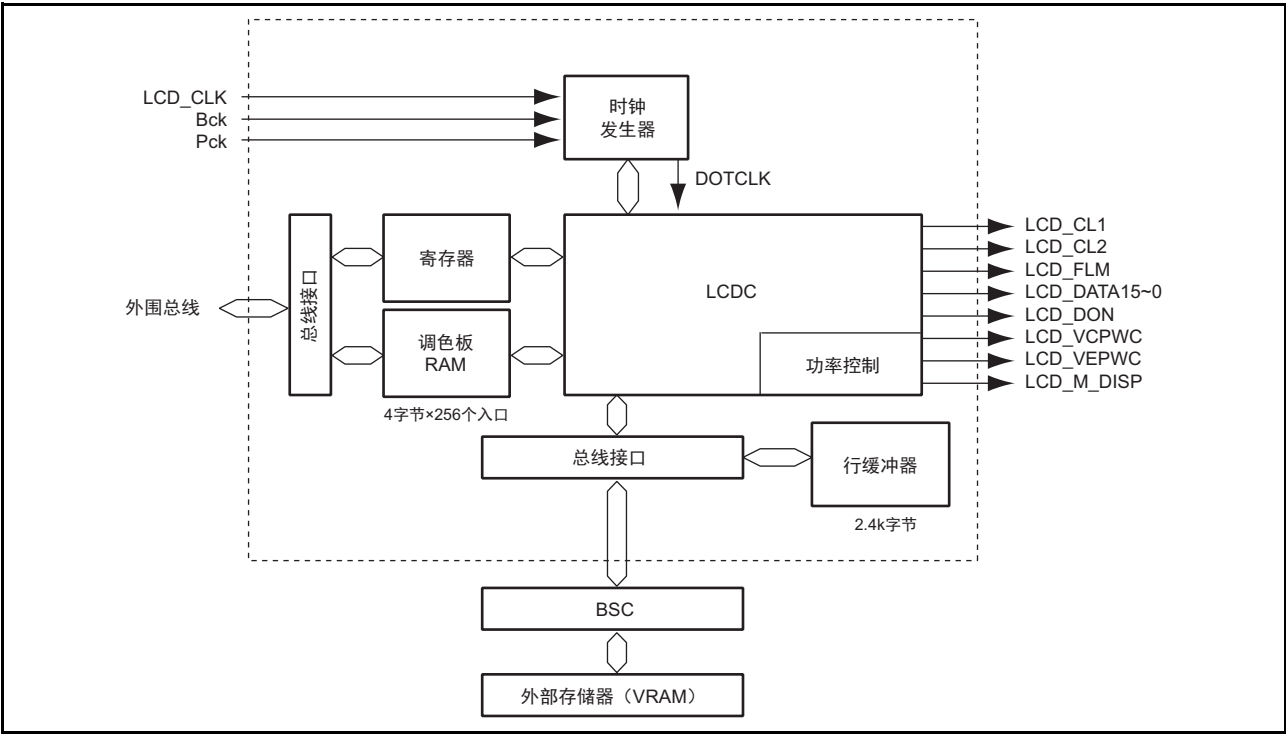


图 24.1 LCDC 框图

24.2 输入 / 输出引脚

LCDC 引脚结构如表 24.1 所示:

表 24.1 引脚结构

引脚名称	输入 / 输出	功 能
LCD_DATA15 ~ 0	输出	用于 LCD 面板的数据
LCD_DON	输出	显示开始信号 (DON)
LCD_CL1	输出	移位时钟 1 (STN/DSTN) / 水平同步信号 (HSYNC) (TFT)
LCD_CL2	输出	移位时钟 2 (STN/DSTN) / 点时钟 (DONCLK) (TFT)
LCD_M_DISP	输出	液晶交流信号 / DISP 信号
LCD_FLM	输出	首行标记 / 垂直同步信号 (VSYNC) (TFT)
LCD_VCPWC	输出	液晶模块电源控制 (VCC)
LCD_VEPWC	输出	液晶模块电源控制 (VEE)
LCD_CLK	输入	LCD 时钟源输入

【注】 必须仔细确认 “24.5 时钟和 LCD 数据信号例” 及液晶模块的规格后, 决定与液晶模块的接线规格。

24.3 寄存器说明

LCDC 有以下寄存器。有关这些寄存器的地址及各处理状态中寄存器的状态，详情参阅“第 30 章 寄存器一览表”。

表 24.2 寄存器结构

寄存器名称	简称	R/W	初始值	地址	存取长度
LCDC 输入时钟寄存器	LDICKR	R/W	H'0101	H'FFFFFFC00	16
LCDC 模块类型寄存器	LDMTR	R/W	H'0109	H'FFFFFFC02	16
LCDC 数据格式寄存器	LDDFR	R/W	H'000C	H'FFFFFFC04	16
LCDC 扫描模式寄存器	LDSMR	R/W	H'0000	H'FFFFFFC06	16
LCDC 上部显示面板用取数据起始地址寄存器	LDSARU	R/W	H'0C000000	H'FFFFFFC08	32
LCDC 下部显示面板用取数据起始地址寄存器	LDSARL	R/W	H'0C000000	H'FFFFFFC0C	32
LCDC 显示面板用取数据行地址偏移量寄存器	LDLAOR	R/W	H'0280	H'FFFFFFC10	16
LCDC 调色板控制寄存器	LDPALCR	R/W	H'0000	H'FFFFFFC12	16
调色板数据寄存器 00 ~ FF	LDPR00 ~ FF	R/W	—	H'FFFFFF800 ~ H'FFFFFFBFC	32
LCDC 水平字符数寄存器	LDHCNR	R/W	H'4F52	H'FFFFFFC14	16
LCDC 水平同步信号寄存器	LDHSYNR	R/W	H'0050	H'FFFFFFC16	16
LCDC 垂直显示行数寄存器	LDVDLNR	R/W	H'01DF	H'FFFFFFC18	16
LCDC 垂直总行数寄存器	LDVTLNR	R/W	H'01DF	H'FFFFFFC1A	16
LCDC 垂直同步信号寄存器	LDVSYNR	R/W	H'01DF	H'FFFFFFC1C	16
LCDC AC 调制信号交替行数寄存器	LDACLNR	R/W	H'000C	H'FFFFFFC1E	16
LCDC 中断控制寄存器	LDINTR	R/W	H'0000	H'FFFFFFC20	16
LCDC 电源管理模式寄存器	LDPMMR	R/W	H'0010	H'FFFFFFC24	16
LCDC 电源顺序期间寄存器	LDPSPR	R/W	H'F60F	H'FFFFFFC26	16
LCDC 控制寄存器	LDCNTR	R/W	H'0000	H'FFFFFFC28	16
LCDC 用户指定中断控制寄存器	LDUINTR	R/W	H'0000	H'FFFFFFC34	16
LCDC 用户指定中断行数寄存器	LDUINLNR	R/W	H'004F	H'FFFFFFC36	16
LCDC 存储器存取间隔数寄存器	LDLIRNR	R/W	H'0000	H'FFFFFFC40	16

24.3.1 LCDC 输入时钟寄存器 (LDICKR)

LCDC 可选择总线时钟、外围时钟或外部时钟作为运行时钟供给源。并且，内置 1/1 ~ 1/32 的分频器，分频的时钟可用作 LCDC 运行时钟 (DOTCLK)。LCDC 输出的时钟从本寄存器选择的运行时钟生成液晶面板的同步时钟输出 (LCD_CL2)。为 TFT 面板时，输出 LCD_CL2=DOTCLK 频率的时钟；为 STN、DSTN 面板时，输出 LCD_CL2= (DOTCLK/ 向液晶面板输出的数据总线宽度) 频率的时钟。设定 LDICKR 时，与 LCD_CL2 无关，LCDC 的输入时钟必须小于等于 66MHz。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	ICKSEL[1:0]		-	-	-	-	-	-	DCDR[5:0]					
初始值:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1
R/W:	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15、14	—	均为 0	R	保留位 读取值、写入值总是为 0。
13、12	ICKSEL[1:0]	00	R/W	输入时钟选择 设定 DOTCLK 的供给源。 00: 选择总线时钟 (Bck) 01: 选择外围时钟 (Pck) 10: 选择外部时钟 (LCD_CLK) 11: 禁止设定
11 ~ 9	—	均为 0	R	保留位 读取值、写入值总是为 0。
8	—	1	R	保留位 读取值、写入值总是为 1。
7、6	—	均为 0	R	保留位 读取值、写入值总是为 0。
5 ~ 0	DCDR[5:0]	000001	R/W	时钟分频比 设定输入时钟分频比。 设定详情参照表 24.3。

表 24.3 输入 / 输出时钟频率与分频比

DCDR[5:0]	时钟分频比	输入 / 输出时钟频率 (MHz)		
		50.000	60.000	66.000
000001	1/1	50.000	60.000	66.000
000010	1/2	25.000	30.000	33.000
000011	1/3	16.667	20.000	22.000
000100	1/4	12.500	15.000	16.500
000110	1/6	8.333	10.000	11.000
001000	1/8	6.250	7.500	8.250
001100	1/12	4.167	5.000	5.500
010000	1/16	3.125	3.750	4.125
011000	1/24	2.083	2.500	2.750
100000	1/32	1.563	1.875	2.063

【注】 为除上述之外的设定时，时钟分频比为 1/1 (初始值)。

24.3.2 LCDC 模块类型寄存器 (LDMTR)

LDMTR 根据连接的液晶模块的信号极性，设定从 LCDC 输出的控制信号及数据信号的极性。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FLM POL	CL1 POL	DISP POL	DPOL	-	MCNT	CL1CNT	CL2CNT	-	-	MIFTYP[5:0]					
初始值:	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	1
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15	FLMPOL	0	R/W	FLM（垂直同步）极性选择 选择液晶模块的 LCD_FLM（垂直同步信号、首行标记）极性。 0: LCD_FLM 脉冲为高电平有效 1: LCD_FLM 脉冲为低电平有效
14	CL1POL	0	R/W	CL1（水平同步）极性选择 选择液晶模块的 LCD_CL1（水平同步信号）极性。 0: LCD_CL1 脉冲为高电平有效 1: LCD_CL1 脉冲为低电平有效
13	DISPPOL	0	R/W	DISP（允许显示）极性选择 选择液晶模块的 LCD_M_DISP（允许显示）极性。 0: LCD_M_DISP 为高电平有效 1: LCD_M_DISP 为低电平有效
12	DPOL	0	R/W	显示数据极性选择 选择液晶模块的 LCD_DATA（显示数据）极性，支持液晶模块的反映。 0: LCD_DATA 为高电平有效。透射式液晶面板 1: LCD_DATA 为低电平有效。反射式液晶面板
11	—	0	R	保留位 读取值、写入值总是为 0。
10	MCNT	0	R/W	M 信号控制 设定液晶模块的液晶交流信号的输出。 0: 输出 M（AC 行调制）信号 1: 不输出 M 信号
9	CL1CNT	0	R/W	CL1（水平同步）控制 设定垂直回扫线期间的 LCD_CL1 输出。 0: 垂直回扫线期间输出 LCD_CL1 1: 垂直回扫线期间不输出 LCD_CL1
8	CL2CNT	1	R/W	CL2（液晶模块的点时钟）控制 设定垂直水平回扫线期间的 LCD_CL2 输出。 0: 垂直水平回扫线期间输出 LCD_CL2 1: 垂直水平回扫线期间不输出 LCD_CL2
7、6	—	均为 0	R	保留位 读取值、写入值总是为 0。

位	位名称	初始值	R/W	说 明
5 ~ 0	MIFTYP[5:0]	001001	R/W	模块接口类型选择 设定液晶面板的类型和向液晶面板输出的数据总线宽度。液晶面板的类型可选择 STN、DSTN、TFT 等 3 种。向液晶面板输出的数据总线宽度可选择 4 位、8 位、12 位、16 位。TFT 液晶面板的请求数据总线宽度超过 16 位时，必须连接对应面板上存在的数据总线。与 TFT 不同，在 STN、DSTN 的液晶面板，显示色数、显示分辨率和输出数据总线宽度的设定并不一一对应，因此即使是 16bpp 的显示色数，也可能有 8 位数据总线宽度、是 4bpp 的显示色数，也可能有 12 位数据总线宽度。这是因为由数据总线的数据载入方法决定 STN、DSTN 的显示色数，而并非由数据总线的位数。有关 STN、DSTN 时的数据规格，详情参阅所使用的液晶面板的规格书。输出数据总线宽度根据液晶面板的机械接口规格设定。 选择 STN 或 DSTN 类型的液晶面板时，与色彩显示、灰度显示的灰度设定无关，通过 LCDC 内置的 RGB 各 8 位的 24 位空间调制 FRC 控制显示。因此 STN 或 DSTN 显示从 1600 万颜色中选择的 DSPCOLOR 指定的颜色、灰度。使用调色板时，显示在调色板设定的颜色。 000000: STN 单色 4 位数据总线模块 000001: STN 单色 8 位数据总线模块 001000: STN 彩色 4 位数据总线模块 001001: STN 彩色 8 位数据总线模块 001010: STN 彩色 12 位数据总线模块 001011: STN 彩色 16 位数据总线模块 010001: DSTN 单色 8 位数据总线模块 010011: DSTN 单色 16 位数据总线模块 011001: DSTN 彩色 8 位数据总线模块 011010: DSTN 彩色 12 位数据总线模块 011011: DSTN 彩色 16 位数据总线模块 101011: TFT 彩色 16 位数据总线模块 除上述之外: 禁止设定

24.3.3 LCDC 数据格式寄存器 (LDDFR)

LDDFR 为了符合显示驱动软件的规格，需设定 1 字节内的数据的位调整及用于显示的数据型号和色数。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	PABD	-	DSPCOLOR[6:0]						
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15 ~ 9	—	均为 0	R	保留位 读取值、写入值总是为 0。
8	PABD	0	R/W	字节数据像素调整 设定 1 字节数据的像素数据的调整种类。已调整的每个像素的数据内容均相同，与该位的内容无关。例如，数据 H'05 不选择 2 进制的 B'0101 或 B'1010，而作为 CPU 通过 MOV 指令进行普通处理的 H'05 (B'0101)。 0: 将字节数据设定为大端法 1: 将字节数据设定为小端法
7	—	0	R	保留位 读取值、写入值总是为 0。
6 ~ 0	DSPCOLOR [6:0]	0001100	R/W	显示色彩选择 设定显示器的显示色数 (对调色板上 4、5、6bpp 高位写入 0)。 关于标有 (通过调色板) 的显示色彩，实际上调色板设定的颜色可通过显示数据选择并显示。 根据显示分辨率限制旋转显示时可支持的色数，参照表 24.5。 0000000: 单色、2 灰度、1bpp (通过调色板) 0000001: 单色、4 灰度、2bpp (通过调色板) 0000010: 单色、16 灰度、4bpp (通过调色板) 0000100: 单色、64 灰度、6bpp (通过调色板) 0001010: 彩色、16 色、4bpp (通过调色板) 0001100: 彩色、256 色、8bpp (通过调色板) 0011101: 彩色、32k 色 (RGB: 5-5-5)、15bpp 0101101: 彩色、64k 色 (RGB: 5-6-5)、16bpp 除上述之外: 禁止设定

24.3.4 LCDC 扫描模式寄存器 (LDSMR)

LDSMR 指定旋转液晶面板所使用的硬件旋转功能的 ON/OFF，并对用于确保显示的 VRAM（区域 3 的同步 DRAM）指定突发长度。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	ROT	-	-	-	AU[1:0]	-	-	-	-	-	-	-	-	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R	R	R	R/W	R/W	R	R	R	R	R	R	R	R

位	位名称	初始值	R/W	说 明
15、14	—	均为 0	R	保留位 读取值、写入值总是为 0。
13	ROT	0	R/W	选择旋转模块 选择由硬件显示的旋转运行，旋转时有以下限制 - 模块类型为 STN 或 TFT。不可为 DSTN - 液晶面板的水平方向（液晶面板内部的扫描方向）宽度最大为 320 - 超过 LDLAOR 显示尺寸，设定 2 乘方的值（320×240 面板旋转为 240×320 面板，显示图像的宽度为 240 字节时，必须选择 256） 0: 不旋转 1: 向右旋转 90 度（显示图像的左侧在液晶模块的上面）
12 ~ 10	—	均为 0	R	保留位 读取值、写入值总是为 0。
9、8	AU[1:0]	00	R/W	存取单元选择 选择 VRAM 的存取单位。仅在 ROT=1（旋转时）时本位有效，ROT=0 时，进行 16 突发运行，与 AU 的设定无关。 00: 4 突发 01: 8 突发 10: 16 突发 11: 32 突发 【注】 1. 以上所示为 32 位总线宽度时的突发长度，16 位总线宽度时翻倍。 2. 根据连接 SDRAM 的列地址位数和总线宽度不同，旋转显示时的突发长度有不同限制，详情参照表 24.4 及表 24.5。
7 ~ 0	—	均为 0	R	保留位 读取值、写入值总是为 0。

24.3.5 LCDC 上部显示面板用取数据起始地址寄存器 (LDSARU)

LDSARU 指定将液晶面板所显示的数据取入 LCDC 的起始地址。使用 DSTN 型的液晶面板时，本寄存器指定上部面板取入的起始地址。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	SAU25	SAU24	SAU23	SAU22	SAU21	SAU20	SAU19	SAU18	SAU17	SAU16
初始值:	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SAU15	SAU14	SAU13	SAU12	SAU11	SAU10	SAU9	SAU8	SAU7	SAU6	SAU5	SAU4	-	-	-	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

位	位名称	初始值	R/W	说 明
31 ~ 28	—	均为 0	R	保留位 读取值、写入值总是为 0。
27、26	—	均为 1	R	保留位 读取值、写入值总是为 1。
25 ~ 4	SAU25 ~ SAU4	均为 0	R/W	取上部面板显示数据的起始地址 在区域 3 的同步 DRAM 区域设定取显示数据的起始地址。
3 ~ 0	—	均为 0	R	保留位 读取值、写入值总是为 0。

- 【注】 1. 不使用硬件旋转功能时，LDSARU 的最小调整单位为 512 字节，低 9 位必须设定为 0；使用硬件旋转功能时，必须设定 LDSARU 的值，使图像左上的地址在 512 字节边界。
2. 使用硬件旋转功能 (ROT=1) 时，必须在本寄存器设定根据显示的图像尺寸计算出的图像的左下地址。设定图像为 240×320、LDLAOR=256 时，可进行以下计算。LDSARU 的值并非根据面板的尺寸，而是根据显示图像的存储器大小计算。此时必须注意 LDLAOR 至少为图像水平尺寸的 2 乘方。此时如果使用 LDSARU 进行倒算，为 LDSARU-256 (LDLAOR 的值) × (320-1)，因此必须确认图像左上的地址在 512 字节边界。
LDSARU=图像左上的地址+256 (LDLAOR 的值) ×319 (行)。

24.3.6 LCDC 下部显示用面板取数据起始地址寄存器 (LDSARL)

使用 DSTN 型的液晶面板时，LDSARL 指定取下部面板数据的起始地址。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	SAL25	SAL24	SAL23	SAL22	SAL21	SAL20	SAL19	SAL18	SAL17	SAL16
初始值:	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SAL15	SAL14	SAL13	SAL12	SAL11	SAL10	SAL9	SAL8	SAL7	SAL6	SAL5	SAL4	-	-	-	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

位	位名称	初始值	R/W	说 明
31 ~ 28	—	均为 0	R	保留位 读取值、写入值总是为 0。
27、26	—	均为 1	R	保留位 读取值、写入值总是为 1。
25 ~ 4	SAL25 ~ SAL4	均为 0	R/W	取下部面板显示数据的起始地址 在区域 3 的同步 DRAM 区域设定取显示数据的起始地址。 STN、TFT: 不使用 DSTN: 取下部面板对应的显示数据的起始地址
3 ~ 0	—	均为 0	R	保留位 读取值、写入值总是为 0。

24.3.7 LCDC 显示面板取数据行地址偏移量寄存器 (LDLAOR)

LDLAOR 指定 LCDC 读取图形驱动器识别的图像的 Y 坐标递增的地址宽度。当 Y 坐标方向增加 1 时，指定从存储器读取数据需移动多少字节地址，无需与液晶面板的水平宽度相同。通过 $Ax+By+C$ 计算 2 维图像上的点 (X、Y) 的存储器地址时，本寄存器等同于公式中的 B。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LAO15	LAO14	LAO13	LAO12	LAO11	LAO10	LAO9	LAO8	LAO7	LAO6	LAO5	LAO4	LAO3	LAO2	LAO1	LAO0
初始值:	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15 ~ 10	LAO15 ~ LAO10	均为 0	R/W	行地址偏移量 最小调整单位为 16 字节。以 16 字节为单位处理，因此必须设定各寄存器写入值的低 4 位为 0。读取寄存器值时，可读取低 4 位为 0。初始值为可使 VGA (640×480 点) 显示数据不跳过地址而连续、密集地配置在行之间的设定值 (×分辨率=640)，详情参照表 24.4 及表 24.5。 考虑到软件的运行速度，推荐 LDLAOR 的值大于等于图像水平宽度的 2 乘方的值。另外，使用硬件旋转功能时，并非液晶面板的宽度 (例如: 320×240 面板的 320)，必须为大于等于图像宽度 (旋转后为 240×320 时的 240) 的 2 乘方的值 (本例为 256)。
9	LAO9	1	R/W	
8	LAO8	0	R/W	
7	LAO7	1	R/W	
6 ~ 0	LAO6 ~ LAO0	均为 0	R/W	

24.3.8 LCDC 调色板控制寄存器 (LDPALCR)

LDPALCR 选择由 CPU 或 LCDC 存取调色板存储器。使用调色板存储器显示运行时，设定为通常显示模式；改写调色板存储器的内容时，设定为彩色调色板设定模式。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	PALS	-	-	-	PALEN
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

位	位名称	初始值	R/W	说 明
15 ~ 5	—	均为 0	R	保留位 读取值、写入值总是为 0。
4	PALS	0	R	调色板状态 表示调色板存取权的状态。 0: LCDC 使用调色板。通常显示模式 1: 主机 (CPU) 使用调色板。彩色调色板设定模式
3 ~ 1	—	均为 0	R	保留位 读取值、写入值总是为 0。
0	PALEN	0	R/W	调色板读取 / 写入允许 请求调色板存取权。 0: 请求转移至通常显示模式 1: 请求转移至彩色调色板设定模式

24.3.9 调色板数据寄存器 00 ~ FF (LDPR00 ~ LDPRFF)

LDPR 存取直接配置在存储器空间 (4 字节 × 256 地址) 的调色板数据。存取调色板存储器时，必须存取本寄存器 (LDPR00 ~ LDPRFF) 中相应的寄存器。各调色板寄存器均为 RGB 各有 8 位区域的 32 位寄存器。有关本调色板的规格，详情参阅 “24.4.3 彩色调色板规格”。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	PALDnn23	PALDnn22	PALDnn21	PALDnn20	PALDnn19	PALDnn18	PALDnn17	PALDnn16
初始值:	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PALDnn15	PALDnn14	PALDnn13	PALDnn12	PALDnn11	PALDnn10	PALDnn9	PALDnn8	PALDnn7	PALDnn6	PALDnn5	PALDnn4	PALDnn3	PALDnn2	PALDnn1	PALDnn0
初始值:	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
31 ~ 24	—	—	R	保留位
23 ~ 0	PALDnn23 ~ PALDnn0	—	R/W	调色板数据 bit18 ~ 16、9、8 及 2 ~ 0 为 RGB 各调色板的保留位，不可设定，但根据高位可扩展后使用。

【注】 nn=H'00 ~ H'FF

24.3.10 LCDC 水平字符数寄存器 (LDHCNR)

LDHCNR 指定液晶模块的水平方向（扫描方向）尺寸及含水平回扫线期间的所有扫描宽度。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HDCN7	HDCN6	HDCN5	HDCN4	HDCN3	HDCN2	HDCN1	HDCN0	HTCN7	HTCN6	HTCN5	HTCN4	HTCN3	HTCN2	HTCN1	HTCN0
初始值:	0	1	0	0	1	1	1	1	0	1	0	1	0	0	1	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15	HDCN7	0	R/W	水平显示字符数 设定水平画面方向的显示字符数（字符=8点单位）。 必须设定（显示字符数）-1 的值。 （例）使用宽度 640pixel 的液晶模块时 HDCN=(640/8)-1=79=H'4F
14	HDCN6	1	R/W	
13	HDCN5	0	R/W	
12	HDCN4	0	R/W	
11	HDCN3	1	R/W	
10	HDCN2	1	R/W	
9	HDCN1	1	R/W	
8	HDCN0	1	R/W	
7	HTCN7	0	R/W	水平总字符数 设定水平画面方向的总字符数（字符=8点）。 必须设定（总字符数）-1 的值。 但最小的水平回扫线期间为 3 个字符（24 点）。 （例）使用宽度 640pixel 的液晶模块时， HTCN=[(640/8)-1]+3=82=H'52 此时，水平总点数为 664 点，水平回扫线期间为 24 点。
6	HTCN6	1	R/W	
5	HTCN5	0	R/W	
4	HTCN4	1	R/W	
3	HTCN3	0	R/W	
2	HTCN2	0	R/W	
1	HTCN1	1	R/W	
0	HTCN0	0	R/W	

【注】 1. HDCN、HTCN 的设定值必须满足 $HTCN \geq HDCN$ 的关系。另外，HTCN 的总字符数必须设定为偶数（设定值设定为 -1 后的值，所以为奇数）。

2. 根据所使用的显示器的分辨率，HDCN 的设定如下所示：

1bpp 时：（16 的倍数）-1 [1 行为 128pixel 的倍数]

2bpp 时：（8 的倍数）-1 [1 行为 64pixel 的倍数]

4bpp 时：（4 的倍数）-1 [1 行为 32pixel 的倍数]

6bpp/8bpp 时：（2 的倍数）-1 [1 行为 16pixel 的倍数]

24.3.11 LCDC 水平同步信号寄存器 (LDHSYNR)

LDHSYNR 是指定液晶面板模块的水平方向（扫描方向）的同步信号时序的寄存器。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HSYNW3	HSYNW2	HSYNW1	HSYNW0	-	-	-	-	HSYNP7	HSYNP6	HSYNP5	HSYNP4	HSYNP3	HSYNP2	HSYNP1	HSYNP0
初始值:	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15	HSYNW3	0	R/W	水平同步信号宽度
14	HSYNW2	0	R/W	设定水平画面方向同步信号（CL1、Hsync）宽度（字符=8点）。
13	HSYNW1	0	R/W	
12	HSYNW0	0	R/W	必须设定（水平同步信号宽度）-1 的值。 （例）水平同步信号宽度为 8 点时， HSYNW=（8 点 / 8 点 / 字符）-1=0=H'0
11 ~ 8	—	均为 0	R	保留位 读取值、写入值总是为 0。
7	HSYNP7	0	R/W	水平同步信号输出位置
6	HSYNP6	1	R/W	设定水平画面方向的同步信号的输出位置（字符=8 点）。
5	HSYNP5	0	R/W	必须设定（水平同步信号输出位置）-1 的值。
4	HSYNP4	1	R/W	（例）使用宽度 640pixel 的液晶模块时， HSYNP=[(640/8)+1]-1=80=H'50
3	HSYNP3	0	R/W	
2	HSYNP2	0	R/W	此时，从第 648 点到第 655 点的水平同步信号有效。
1	HSYNP1	0	R/W	
0	HSYNP0	0	R/W	

【注】 必须满足：HTCN ≥ HSYNP+HSYNW+1
HSYNP ≥ HDCN+1 的关系。

24.3.12 LCDC 垂直显示行数寄存器 (LDVDLNR)

LDVDLNR 是指定液晶面板模块纵向（与扫描方向垂直的方向）尺寸的寄存器。为 DSTN 时，必须指定不小于面板模块的纵向尺寸的偶数，与上下面板的尺寸无关。（例如：640×480 的面板时为 480）。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	VDLN10	VDLN9	VDLN8	VDLN7	VDLN6	VDLN5	VDLN4	VDLN3	VDLN2	VDLN1	VDLN0
初始值:	0	0	0	0	0	0	0	1	1	1	0	1	1	1	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15 ~ 11	—	均为 0	R	保留位 读取值、写入值总是为 0。
10	VDLN10	0	R/W	垂直显示行数 设定垂直画面方向的显示行数（单位：行）。 必须设定（显示行数）-1 的值。 （例）使用 480 行的液晶模块时， VDLN=480-1=479=H'1DF
9	VDLN9	0	R/W	
8	VDLN8	1	R/W	
7	VDLN7	1	R/W	
6	VDLN6	1	R/W	
5	VDLN5	0	R/W	
4	VDLN4	1	R/W	
3	VDLN3	1	R/W	
2	VDLN2	1	R/W	
1	VDLN1	1	R/W	
0	VDLN0	1	R/W	

24.3.13 LCDC 垂直总行数寄存器 (LDVTLNR)

LDVTLNR 是指定包含液晶面板模块的垂直回扫线期间的所有纵向长度的寄存器。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	VTLN10	VTLN9	VTLN8	VTLN7	VTLN6	VTLN5	VTLN4	VTLN3	VTLN2	VTLN1	VTLN0
初始值:	0	0	0	0	0	0	0	1	1	1	0	1	1	1	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15 ~ 11	—	均为 0	R	保留位 读取值、写入值总是为 0。
10	VTLN10	0	R/W	垂直总行数 设定垂直画面方向的总行数（单位：行）。 必须设定为（总行数）-1 的值。 最小的垂直总行数为 2 行。 必须满足 $VTLN \geq VDLN$ 、 $VTLN \geq 1$ 。 （例）使用 480 行的液晶模块且垂直回扫线期间为 0 行时， $VTLN=(480+0)-1=479=H'1DF$
9	VTLN9	0	R/W	
8	VTLN8	1	R/W	
7	VTLN7	1	R/W	
6	VTLN6	1	R/W	
5	VTLN5	0	R/W	
4	VTLN4	1	R/W	
3	VTLN3	1	R/W	
2	VTLN2	1	R/W	
1	VTLN1	1	R/W	
0	VTLN0	1	R/W	

24.3.14 LCDC 垂直同步信号寄存器 (LDVSYNR)

LDVSYNR 是指定液晶模块的纵向（与扫描方向垂直的方向）的同步信号时序的寄存器。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VSYNW3	VSYNW2	VSYNW1	VSYNW0	-	VSYNP10	VSYNP9	VSYNP8	VSYNP7	VSYNP6	VSYNP5	VSYNP4	VSYNP3	VSYNP2	VSYNP1	VSYNP0
初始值:	0	0	0	0	0	0	0	1	1	1	0	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15	VSYNW3	0	R/W	垂直同步信号宽度 设定垂直画面方向的同步信号（FLM、Vsync）宽度（单位：行）。 必须设定为（垂直同步信号宽度）-1 的值。 （例）垂直同步信号宽度为 1 行时， $VSYNW=(1-1)=0=H'0$
14	VSYNW2	0	R/W	
13	VSYNW1	0	R/W	
12	VSYNW0	0	R/W	
11	—	0	R	保留位 读取值、写入值总是为 0。
10	VSYNP10	0	R/W	垂直同步信号输出位置 设定垂直画面方向的同步信号（FLM、Vsync）的输出位置（单位：行）。 必须设定为（垂直同步信号输出位置）-2 的值。 为 DSTN 时，必须设定为奇数值。作为（设定值 +1）/2 处理。 （例）使用 480 行的液晶模块且回扫线期间为 0 行，即： $VTLN=479$ 时，在第 1 行设置垂直同步信号为有效时 - 单显示器时 $VSYNP=[(1-1)+VTLN] \bmod (VTLN+1)=[(1-1)+479] \bmod (479+1)$ $=479 \bmod 480=479$ $=H'1DF$ - 双地址时 $VSYNP=[(1-1) \times 2 + VTLN] \bmod (VTLN+1)=[(1-1) \times 2 + 479] \bmod (479+1)$ $=479 \bmod 480=479$ $=H'1DF$
9	VSYNP9	0	R/W	
8	VSYNP8	1	R/W	
7	VSYNP7	1	R/W	
6	VSYNP6	1	R/W	
5	VSYNP5	0	R/W	
4	VSYNP4	1	R/W	
3	VSYNP3	1	R/W	
2	VSYNP2	1	R/W	
1	VSYNP1	1	R/W	
0	VSYNP0	1	R/W	

24.3.15 LCDC AC 调制信号交替行数寄存器 (LDACLNR)

LDACLNR 是指定交替液晶模块的 AC 调制信号 (液晶交流信号) 时序的寄存器。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	ACLN4	ACLN3	ACLN2	ACLN1	ACLN0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15 ~ 5	—	均为 0	R	保留位 读取值、写入值总是为 0。
4	ACLN4	0	R/W	AC 行数 设定交替液晶模块交流信号的行数 (单位: 行)。 必须设定 (交替行数) -1 的值。 (例) 每 13 行交替时, ACLN=13-1=12=H'0C
3	ACLN3	1	R/W	
2	ACLN2	1	R/W	
1	ACLN1	0	R/W	
0	ACLN0	0	R/W	

【注】 面板的总行数为偶数时, 必须设定偶数以便在奇数行交替。

24.3.16 LCDC 中断控制寄存器 (LDINTR)

LDINTR 是指定 Vsync 中断开始点的寄存器。有关中断, 详情参阅 “24.3.20 LCDC 用户指定中断控制寄存器 (LDUINTR)” 及 “24.3.21 LCDC 用户指定中断行数寄存器 (LDUINTLNR)”。另外, 通过本寄存器设定的运行与通过 LCDC 用户指定中断控制寄存器 (LDUINTR) 设定的运行是相互独立的。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MINT EN	FINT EN	VSINT EN	VEINT EN	MINTS	FINTS	VSINTS	VEINTS	-	-	-	-	-	-	-	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

位	位名称	初始值	R/W	说 明
15	MINTEN	0	R/W	存储器存取中断允许 设定通过 LCDC 存取 VRAM 的垂直回扫线期间的开始点是否产生中断。 0: 在 VRAM 存取的垂直回扫线期间的开始点不产生中断。 1: 在 VRAM 存取的垂直回扫线期间的开始点产生中断。
14	FINTEN	0	R/W	帧结束中断允许 设定向液晶面板输出帧的最后像素时是否产生中断。 0: 向液晶面板输出帧的最后像素时不产生中断。 1: 向液晶面板输出帧的最后像素时产生中断。
13	VSINTEN	0	R/W	Vsync 开始中断允许 设定 LCDC Vsync 开始时是否产生中断。 0: LCDC Vsync 开始时不产生中断。 1: LCDC Vsync 开始时产生中断。

位	位名称	初始值	R/W	说 明
12	VEINTEN	0	R/W	Vsync 结束中断允许 设定 LCDC Vsync 结束时是否产生中断。 0: LCDC Vsync 结束时不产生中断。 1: LCDC Vsync 结束时产生中断。
11	MINTS	0	R/W	存储器存取中断状态 表示存储器存取中断的处理状态。 本位在产生 LCDC 存储器存取中断时显示为 1 (置位状态)。存储器存取中断的处理程序中, 必须对寄存器写入 0, 清除本位。 0: 表示 LCDC 未产生存储器存取中断或接收到产生存储器存取中断的处理结束通知。 1: 表示 LCDC 产生存储器存取中断, 未接收到处理结束通知。
10	FINTS	0	R/W	帧结束中断状态 表示帧结束中断的处理状态。 本位在产生 LCDC 帧结束中断时显示为 1 (置位状态)。帧结束中断的处理程序中, 必须对寄存器写入 0, 清除本位。 0: 表示 LCDC 未产生帧结束中断或接收到产生帧结束中断的处理结束通知。 1: 表示 LCDC 产生帧结束中断, 未接收到处理结束通知。
9	VSINTS	0	R/W	Vsync 开始中断状态 表示 LCDC Vsync 开始中断的处理状态。 本位在产生 LCDC Vsync 开始中断时显示为 1 (置位状态)。Vsync 开始中断的处理程序中, 必须对寄存器写入 0, 清除本位。 0: 表示 LCDC 未产生 Vsync 开始中断或接收到产生 Vsync 开始中断的处理结束通知。 1: 表示 LCDC 产生 Vsync 开始中断, 未接收到处理结束通知。
8	VEINTS	0	R/W	Vsync 结束中断状态 表示 LCDC Vsync 结束中断的处理状态。 该位在产生 LCDC Vsync 结束中断时显示为 1 (置位状态)。Vsync 结束中断的处理程序中, 必须对寄存器写入 0, 清除本位。 0: 表示 LCDC 未产生 Vsync 结束中断或接收到产生 Vsync 结束中断的处理结束通知。 1: 表示 LCDC 产生 Vsync 结束中断, 未接收到处理结束通知。
7 ~ 0	—	均为 0	R	保留位 读取值、写入值总是为 0。

24.3.17 LCDC 电源管理模式寄存器 (LDPMMR)

LDPMMR 设定控制电源电路向液晶面板模块提供电源的功能，并设定是否使用 LCD_VCPWC 和 LCD_VEPWC 这 2 种电源控制引脚及上电功能的 ON/OFF。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ONC3	ONC2	ONC1	ONC0	OFFD3	OFFD2	OFFD1	OFFD0	-	VCPE	VEPE	DONE	-	-	-	LPS[1:0]
初始值:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R	R

位	位名称	初始值	R/W	说 明
15 14 13 12	ONC3 ONC2 ONC1 ONC0	0 0 0 0	R/W R/W R/W R/W	LCDC 上电顺序期间 在 LCD 模块的上电顺序中，以帧周期为单位设定从 LCD_VEPWC 引脚有效到 LCD_DON 引脚有效期间。必须设定（期间）-1 的值。 相当于图 24.4 ~ 图 24.7 中“电源控制顺序和液晶模块的运行状态”的（c）期间，有关详细方法，参照表 24.6（ONA、ONB、OFFD、OFFE、OFFF 各寄存器通用设定方法）。
11 10 9 8	OFFD3 OFFD2 OFFD1 OFFD0	0 0 0 0	R/W R/W R/W R/W	LCDC 断电顺序期间 在 LCD 模块的断电顺序中，以帧周期为单位设定从 LCD_DON 引脚无效到 LCD_VEPWC 引脚无效期间。必须设定（期间）-1 的值。 相当于图 24.4 ~ 图 24.7 中“电源控制顺序和液晶模块的运行状态”的（d）期间。
7	—	0	R	保留位 读取值、写入值总是为 0。
6	VCPE	0	R/W	LCD_VCPWC 引脚允许 设定是否有使用 LCD_VCPWC 引脚的电源控制顺序处理。 0:（无处理）屏蔽 LCD_VCPWC 引脚输出并固定为低电平 1:（有处理）按照规定的顺序，使 LCD_VCPWC 引脚输出有效或无效
5	VEPE	0	R/W	LCD_VEPWC 引脚允许 设定是否有使用 LCD_VEPWC 引脚的电源控制顺序处理。 0:（无处理）屏蔽 LCD_VEPWC 引脚输出并固定为低电平 1:（有处理）按照规定的顺序，使 LCD_VEPWC 引脚输出有效或无效
4	DONE	1	R/W	LCD_DON 引脚允许 设定是否有使用 LCD_DON 引脚的电源控制顺序处理。 0:（无处理）屏蔽 LCD_DON 引脚输出并固定为低电平 1:（有处理）按照规定的顺序，使 LCD_DON 引脚输出有效或无效
3、2	—	均为 0	R	保留位 读取值、写入值总是为 0。
1、0	LPS[1:0]	00	R	液晶模块电源输入状态 表示使用电源、控制功能时的液晶模块的上电状态。 00: 液晶模块断电 11: 液晶模块上电

24.3.18 LCDC 上电顺序期间寄存器 (LDPSPR)

LDPSPR 设定控制电源电路向液晶面板模块提供电源的功能，并指定伴随 LCD_VEPWC、LCD_VCPWC 引脚的时序信号的开始输出时序。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ONA3	ONA2	ONA1	ONA0	ONB3	ONB2	ONB1	ONB0	OFFE3	OFFE2	OFFE1	OFFE0	OFFF3	OFFF2	OFFF1	OFFF0
初始值:	1	1	1	1	0	1	1	0	0	0	0	0	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15 14 13 12	ONA3 ONA2 ONA1 ONA0	1 1 1 1	R/W R/W R/W R/W	LCDC 上电顺序期间 在 LCD 模块的上电顺序中，以帧周期为单位设定从 LCD_VCPWC 引脚有效到开始输出显示数据 (LCD_DATA) 和时序信号 (LCD_FLM、LCD_CL1、LCD_CL2、LCD_M_DISP) 的期间。必须设定 (期间) -1 的值。 相当于图 24.4 ~ 图 24.7 中“电源控制顺序和液晶模块的运行状态”的 (a) 期间。
11 10 9 8	ONB3 ONB2 ONB1 ONB0	0 1 1 0	R/W R/W R/W R/W	LCDC 上电顺序期间 在 LCD 模块的上电顺序中，以帧周期为单位设定从开始输出显示数据 (LCD_DATA) 和时序信号 (LCD_FLM、LCD_CL1、LCD_CL2、LCD_M_DISP) 到 LCD_VEPWC 引脚有效的期间。必须设定 (期间) -1 的值。 相当于图 24.4 ~ 图 24.7 中“电源控制顺序和液晶模块的运行状态”的 (b) 期间。
7 6 5 4	OFFE3 OFFE2 OFFE1 OFFE0	0 0 0 0	R/W R/W R/W R/W	LCDC 断电顺序期间 在液晶模块的断电顺序中，以帧周期为单位设定从 LCD_VEPWC 引脚无效到停止输出显示数据 (LCD_DATA) 和时序信号 (LCD_FLM、LCD_CL1、LCD_CL2、LCD_M_DISP) 的期间。必须设定 (期间) -1 的值。 相当于图 24.4 ~ 图 24.7 中“电源控制顺序和液晶模块的运行状态”的 (e) 期间。
3 2 1 0	OFFF3 OFFF2 OFFF1 OFFF0	1 1 1 1	R/W R/W R/W R/W	LCDC 断电顺序期间 在液晶模块的断电顺序中，以帧周期为单位设定从停止输出显示数据 (LCD_DATA) 和时序信号 (LCD_FLM、LCD_CL1、LCD_CL2、LCD_M_DISP) 到 LCD_VCPWC 引脚无效的期间。必须设定 (期间) -1 的值。 相当于图 24.4 ~ 图 24.7 中“电源控制顺序和液晶模块的运行状态”的 (f) 期间。

24.3.19 LCDC 控制寄存器 (LDCNTR)

LDCNTR 指定 LCDC 开始 / 结束显示运行。

对 DON2 位和 DON 位分别写入 1 时，LCDC 开始显示。之后，按照在 LDPMMR 及 LDPSPR 设定的顺序，对液晶模块上电。LPS1、LPS0 位从 B'00 变为 B'11 时，规定的顺序结束。在规定的顺序结束之前，不可进行下一个 DON 位的操作。

LCDC 显示结束时，DON 位设定为 0。按照在 LDPMMR 及 LDPSPR 设定的顺序，断开液晶模块的电源，LCDC 停止运行。LPS[1:0] 从 B'11 变为 B'00 时，规定的顺序结束。在规定的顺序结束之前，不可进行下一个 DON 位的操作。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	DON2	-	-	-	DON
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

位	位名称	初始值	R/W	说 明
15 ~ 5	—	均为 0	R	保留位 读取值、写入值总是为 0。
4	DON2	0	R/W	显示器开 2 指定 LCDC 开始显示运行。 0: LCDC 运行 / 结束时 1: 开始运行 读取值总是为 0。必须在开始显示运行时写入 1。除开始显示运行之外时，如果写入 1，则无法保证运行。写入 1 时自动恢复为 0，因此该位无需通过写入 0 来清除 1。
3 ~ 1	—	均为 0	R	保留位 读取值、写入值总是为 0。
0	DON	0	R/W	显示器开 指定 LCDC 开始 / 结束显示运行。 控制顺序的状态可通过参照 LDPMMR 的 LPS[1:0] 值进行确认。 0: LCDC 停止。显示关模式 1: LCDC 运行。显示开模式

- 【注】
1. 显示开始时将 H'0011 写入 LDCNTR，显示结束时将 H'0000 写入 LDCNTR。禁止写入除此之外的值。
 2. 对 DON2 位写入 1 时，调色板 RAM 数据不定，因此，对 DON2 位写入 1 后，必须设定调色板 RAM 数据。
 3. 写入 LDCNTR 后，在实际显示开始 / 结束运行之前需要时间。因此，写入 LDCNTR 后，存取 LCDC 的其他寄存器前，必须虚读 1 次 LDCNTR。

24.3.20 LCDC 用户指定中断控制寄存器 (LDUINTR)

LDUINTR 设定是否产生用户指定中断，并显示其状态。LCDC 从 VRAM 读取完在 LCDC 用户指定中断行数寄存器 (LDUINTLNR) 设定的行的图像数据时，产生本中断。

本 LCDC 发行的中断 (LCDCI) 为：通过本寄存器的用户指定中断、通过 LCDC 中断控制寄存器 (LDINTR) 的存储器存取中断及与 Vsync 中断的 OR 输出。另外，本寄存器和 LCDC 中断控制寄存器 (LDINTR) 的设定，对中断运行独立作用。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	UINTEN	-	-	-	-	-	-	-	UINTS
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

位	位名称	初始值	R/W	说 明
15 ~ 9	—	均为 0	R	保留位 读取值、写入值总是为 0。
8	UINTEN	0	R/W	用户指定中断允许 设定是否产生 LCDC 用户指定中断。 0: 不产生 LCDC 用户指定中断。 1: 产生 LCDC 用户指定中断。
7 ~ 1	—	均为 0	R	保留位 读取值、写入值总是为 0。
0	UINTS	0	R/W	用户指定中断状态 本位在产生 LCDC 用户指定中断时显示为 1 (置位状态)。用户指定中断的处理程序中，必须对寄存器写入 0，清除本位。 0: 表示 LCDC 未产生用户指定中断或接收到产生用户指定中断的处 理结束通知。 1: 表示 LCDC 产生用户指定中断，未接收到处理结束通知。

【注】 中断处理流程

1. 输入中断信号
2. 读取 LDINTR
3. 如果 MINTS、FINTS、VSINTS、VEINTS=1，则其中断为存储器存取、帧结束中断、Vsync 上升沿中断或 Vsync 下降沿中断。处理各中断。
4. 如果 MINTS、FINTS、VSINTS、VEINTS=0，则其中断并非存储器存取、帧结束中断、Vsync 上升沿中断或 Vsync 下降沿中断。
5. 读取 UINTS
6. 如果 UINTS=1，则其中断为用户指定中断。处理用户指定中断。
7. 如果 UINTS=0，则其中断并非用户指定中断。进行其他处理。

24.3.21 LCDC 用户指定中断行数寄存器 (LDUINTLNR)

LDUINTLNR 是以水平行为单位设定产生用户指定中断位置的寄存器。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	UINTLN10	UINTLN9	UINTLN8	UINTLN7	UINTLN6	UINTLN5	UINTLN4	UINTLN3	UINTLN2	UINTLN1	UINTLN0
初始值:	0	0	0	0	0	0	0	0	0	1	0	0	1	1	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15 ~ 11	—	均为 0	R	保留位 读取值、写入值总是为 0。
10	UINTLN10	0	R/W	用户指定中断产生行数 设定产生用户指定中断的行 (单位: 行)。 必须设定 (中断产生行数) -1 的值。 (例) 第 80 行产生用户指定中断时: HINTLN=160/2-1=79=H'04F
9	UINTLN9	0	R/W	
8	UINTLN8	0	R/W	
7	UINTLN7	0	R/W	
6	UINTLN6	1	R/W	
5	UINTLN5	0	R/W	
4	UINTLN4	0	R/W	
3	UINTLN3	1	R/W	
2	UINTLN2	1	R/W	
1	UINTLN1	1	R/W	
0	UINTLN0	1	R/W	

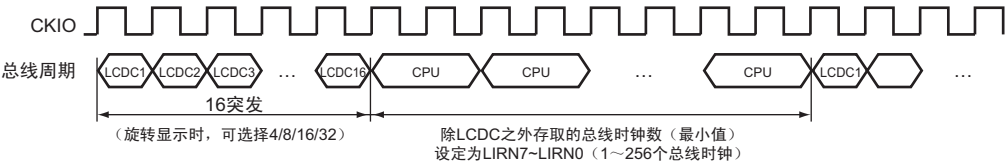
- 【注】
1. 使用 STN/TFT 的液晶模块时, 本寄存器的设定值必须小于等于 LDVDLNR 的垂直显示行数 (VDLN)。
 2. 使用 DSTN 液晶模块时, 本寄存器的设定值必须小于等于 LDVDLNR 垂直显示行数 (VDLN) 的 1/2。此时, LCDC 从 VRAM 读取完本寄存器设定行数的下部画面的图像数据时, 产生用户指定中断。

24.3.22 LCDC 存储器存取间隔数寄存器 (LDLIRNR)

LDLIRNR 控制 LCDC 读取 VRAM 时的总线时钟间隔。在 LDLIRNR 设定的总线时钟期间，LCDC 不存取 VRAM，因此可在此期间由 CPU/DMAC 存取外部总线。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	LIRN7	LIRN6	LIRN5	LIRN4	LIRN3	LIRN2	LIRN1	LIRN0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15 ~ 8	—	均为 0	R	保留位 读取值、写入值总是为 0。
7 ~ 0	LIRN7 ~ LIRN0	均为 0	R/W	LCDC 读取 VRAM 的总线时钟间隔 指定 LCDC 读取 VRAM 的突发总线周期期间插入的总线时钟数。 H'00: 1 个总线时钟 H'01: 2 个总线时钟 : H'FF: 256 个总线时钟



24.4 运行说明

24.4.1 关于 LCDC 可显示的液晶模块尺寸

LCDC 可显示 1024×1024 点、16bpp（比特每像素）。显示的图像保存在与 CPU 共用的 VRAM，显示前本 LCDC 需从 VRAM 读取数据。

本 LSI 内置最大 32 突发的存储器读取和 2.4K 字节的行缓冲器，不容易产生显示失败，但由于组合的原因，有时会难于显示。帧速率为 60Hz 时，推荐尺寸为：16bpp 时 320×240 点或 8bpp 时 640×480 点。

标准：以下所示总线占用率不得超过 40%。

开销系数×显示像素总数 ((HDCN+1) ×8× (VDLN+1))

×帧速率 (Hz) ×色数 (bpp)

总线占有率 (%) = ————— ×100

CKIO (Hz) ×总线宽度 (bit)

在 CL2 的 SDRAM 为 32 位总线时开销系数为 1.375，16 位总线时开销系数为 1.188。
有效显示和回扫线期间如图 24.2 所示。

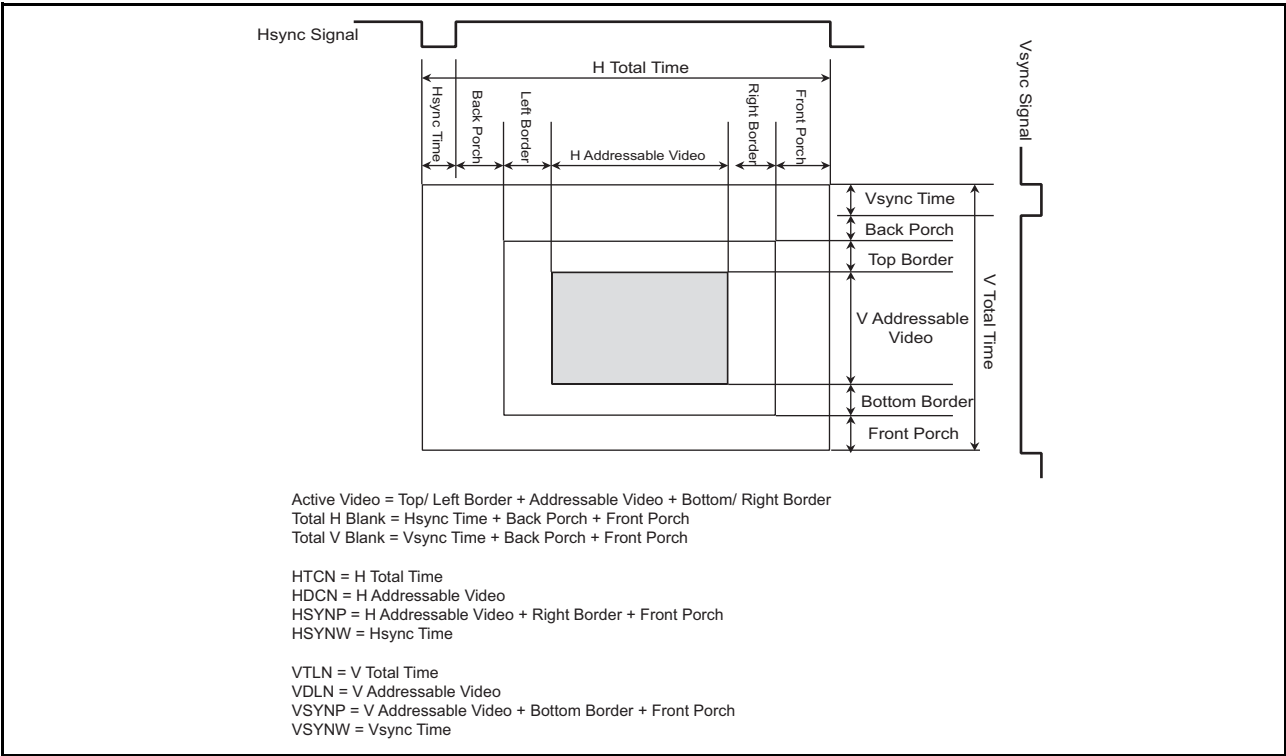


图 24.2 有效显示和回扫线期间

24.4.2 旋转显示的分辨率 / 突发长度及连接存储器 (SDRAM) 的限制事项

本 LCDC 可将纵向显示图像旋转 90 度，并显示在对应的横向液晶模块。各分辨率仅可显示以下色数。为了连续读取 SDRAM，对 SDRAM 的尺寸（列地址位数）及 LCDC 的突发长度有所限制。

显示色数、SDRAM 列地址数及 LCDC 的突发长度限制如表 24.4 及表 24.5 所示。

另外，显示单色显示图像需单色液晶模块，彩色显示图像需彩色液晶模块。

表 24.4 旋转显示的分辨率 / 突发长度及连接存储器的限制（32 位宽度的 SDRAM 时）

存储器的显示图像 (X 分辨率 × Y 分辨率)	LCDC 模块 (X 分辨率 × Y 分辨率)	显示色数		所用 SDRAM 的 列地址位数	LCDC 的突发长度 (LDSMR*)
240×320	320×240	单色	4bpp (packed)	8 位	小于等于 8 突发
				9 位	小于等于 16 突发
				10 位	—
			4bpp (unpacked)	8 位	4 突发
				9 位	小于等于 8 突发
				10 位	小于等于 16 突发
			6bpp	8 位	4 突发
				9 位	小于等于 8 突发
				10 位	小于等于 16 突发
		彩色	8bpp	8 位	4 突发
				9 位	小于等于 8 突发
				10 位	小于等于 16 突发
			16bpp	8 位	不可使用
				9 位	4 突发
				10 位	小于等于 8 突发
234×320	320×234	单色	6bpp	8 位	4 突发
				9 位	小于等于 8 突发
				10 位	小于等于 16 突发
		彩色	16bpp	8 位	不可使用
				9 位	4 突发
				10 位	小于等于 8 突发
80×160	160×80	单色	2bpp	8 位	—
				9 位	—
				10 位	—
			4bpp (packed)	8 位	小于等于 16 突发
				9 位	—
				10 位	—
			4bpp (unpacked)	8 位	小于等于 8 突发
				9 位	小于等于 16 突发
				10 位	—
			6bpp	8 位	小于等于 8 突发
				9 位	小于等于 16 突发
				10 位	—

存储器的显示图像 (X 分辨率 × Y 分辨率)	LCDC 模块 (X 分辨率 × Y 分辨率)	显示色数		所用 SDRAM 的 列地址位数	LCDC 的突发长度 (LDSMR*)
80×160	160×80	彩色	4bpp (packed)	8 位	小于等于 16 突发
				9 位	—
				10 位	—
			4bpp (unpacked)	8 位	小于等于 8 突发
				9 位	小于等于 16 突发
				10 位	—
			8bpp	8 位	小于等于 8 突发
				9 位	小于等于 16 突发
				10 位	—
			16bpp	8 位	4 突发
				9 位	小于等于 8 突发
				10 位	小于等于 16 突发
64×128	128×64	单色	1bpp	8 位	—
				9 位	—
				10 位	—
			2bpp	8 位	—
				9 位	—
				10 位	—
			4bpp (packed)	8 位	—
				9 位	—
				10 位	—
			4bpp (unpacked)	8 位	小于等于 16 突发
				9 位	—
				10 位	—
			6bpp	8 位	小于等于 16 突发
				9 位	—
				10 位	—
		彩色	4bpp (packed)	8 位	—
				9 位	—
				10 位	—
			4bpp (unpacked)	8 位	小于等于 16 突发
				9 位	—
				10 位	—
			8bpp	8 位	小于等于 16 突发
				9 位	—
				10 位	—

【注】 * 设定数据时，必须使设定为突发长度的行数的数据存储在 SDRAM 的相同 ROW 地址。

表 24.5 旋转显示分辨率 / 突发长度及连接存储器的限制 (SDRAM16 位宽度时)

存储器的显示图像 (X 分辨率 × Y 分辨率)	LCDC 模块 (X 分辨率 × Y 分辨率)	显示色数		所用 SDRAM 的 列地址位数	LCDC 的突发长度 (LDSMR*)
240×320	320×240	单色	4bpp (packed)	8 位	小于等于 4 突发
				9 位	小于等于 8 突发
				10 位	小于等于 16 突发
			4bpp (unpacked)	8 位	不可使用
				9 位	4 突发
				10 位	小于等于 8 突发
			6bpp	8 位	不可使用
				9 位	4 突发
				10 位	小于等于 8 突发
		彩色	8bpp	8 位	不可使用
				9 位	4 突发
				10 位	小于等于 8 突发
			16bpp	8 位	不可使用
				9 位	不可使用
				10 位	4 突发
234×320	320×234	单色	6bpp	8 位	不可使用
				9 位	4 突发
				10 位	小于等于 8 突发
		彩色	16bpp	8 位	不可使用
				9 位	不可使用
				10 位	4 突发
80×160	160×80	单色	2bpp	8 位	小于等于 16 突发
				9 位	—
				10 位	—
			4bpp (packed)	8 位	小于等于 8 突发
				9 位	小于等于 16 突发
				10 位	—
			4bpp (unpacked)	8 位	4 突发
				9 位	小于等于 8 突发
				10 位	小于等于 16 突发
			6bpp	8 位	4 突发
				9 位	小于等于 8 突发
				10 位	小于等于 16 突发

存储器的显示图像 (X 分辨率 × Y 分辨率)	LCDC 模块 (X 分辨率 × Y 分辨率)	显示色数		所用 SDRAM 的 列地址位数	LCDC 的突发长度 (LDSMR*)
80×160	160×80	彩色	4bpp (packed)	8 位	小于等于 8 突发
				9 位	小于等于 16 突发
				10 位	—
			4bpp (unpacked)	8 位	4 突发
				9 位	小于等于 8 突发
				10 位	小于等于 16 突发
			8bpp	8 位	4 突发
				9 位	小于等于 8 突发
				10 位	小于等于 16 突发
			16bpp	8 位	不可使用
				9 位	4 突发
				10 位	小于等于 8 突发
64×128	128×64	单色	1bpp	8 位	—
				9 位	—
				10 位	—
			2bpp	8 位	—
				9 位	—
				10 位	—
			4bpp (packed)	8 位	小于等于 16 突发
				9 位	—
				10 位	—
			4bpp (unpacked)	8 位	小于等于 8 突发
				9 位	小于等于 16 突发
				10 位	—
			6bpp	8 位	小于等于 8 突发
				9 位	小于等于 16 突发
				10 位	—
		彩色	4bpp (packed)	8 位	小于等于 16 突发
				9 位	—
				10 位	—
			4bpp (unpacked)	8 位	小于等于 8 突发
				9 位	小于等于 16 突发
				10 位	—
			8bpp	8 位	小于等于 8 突发
				9 位	小于等于 16 突发
				10 位	—

【注】 * 设定数据时，必须使设定为突发长度的行数的数据存储在 SDRAM 的相同 ROW 地址。

24.4.3 彩色调色板规格

(1) 调色板寄存器

本 LCDC 有通过每个入口输出 24 位数据，可同时使用 256 个入口的调色板。使用该调色板可同时显示 16M 色中的 256 色。

按照以下步骤，用户可随时设定本调色板。

1. LDPALCR 的 PALEN 位=0（初始值）：通常显示运行
2. 存取 LDPALCR，设定 PALEN=1：在外围时钟的 3 个周期后，转移至彩色调色板设定模式。
3. 存取 LDPALCR，确认 PALS=1
4. 存取 LDPR00～FF，在 PALD00～FF 写入必要的值
5. 存取 LDPALCR，设定 PALEN=0：在外围时钟的 1 个周期后，转移至通常显示模式

另外，在 LDPALCR 的 PALS=1 期间，LCDC 显示数据输出（LCD_DATA）为 0 输出。

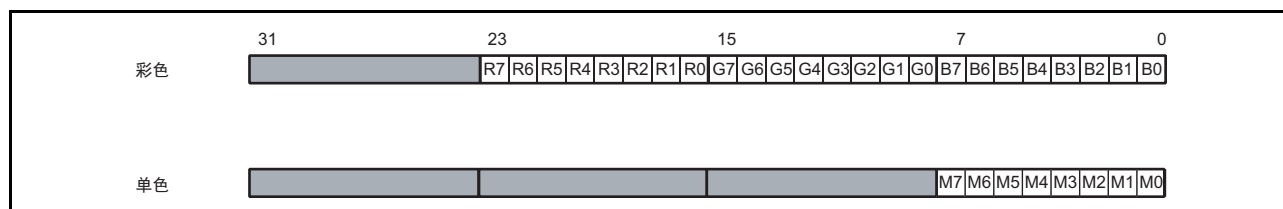


图 24.3 彩色调色板数据格式

必须如上设定 PALDnn 的色彩 / 灰度数据。

彩色显示时，在 PALDnn[23:16] 设定 R 数据，在 PALDnn[15:8] 设定 G 数据，在 PALDnn[7:0] 设定 B 数据。PALDnn[18:16]、PALDnn[9:8]、PALDnn[2:0] 中存在寄存器的位，但不存在对应的存储器。因此，PALDnn[18:16]、PALDnn[9:8]、PALDnn[2:0] 不可保存调色板数据，R:5 位、G:6 位、B:5 位有效。实际使用时，必须写入 24 位（R:8 位、G:8 位、B:8 位）的数据。PALDnn[23:19]、PALDnn[15:10]、PALDnn[7:3] 的值不为 0 时，对 PALDnn[18:16]、PALDnn[9:8]、PALDnn[2:0] 写入 0 或 1，PALDnn[23:19]、PALDnn[15:10]、PALDnn[7:3] 的值为 0 时，对 PALDnn[18:16]、PALDnn[9:8]、PALDnn[2:0] 写入 0，扩展为 24 位。

单色显示时，在 PALDnn[7:3] 设定灰度数据，Don't care PALDnn[23:8]。PALDnn[7:3] 的值不为 0 时，对 PALDnn[2:0] 写入 1；PALDnn[7:3] 的值为 0 时，对 PALDnn[2:0] 写入 0，扩展为 8 位。

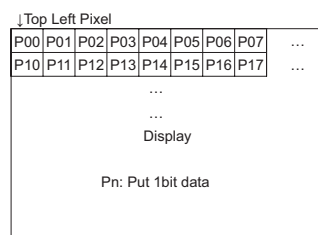
【符号说明】nn: H'00 ～ H'FF

24.4.4 数据格式

1. Packed 1bpp (Pixel Alignment in Byte is Big Endian) [Windows CE Recommended Format]

Address	MSB							LSB	[Bit]
	7	6	5	4	3	2	1	0	
+00	P00	P01	P02	P03	P04	P05	P06	P07	(Byte0)
+01	P08								(Byte1)
+02									
+03									
...									
+LAO+00	P10	P11	P12	P13	P14	P15	P16	P17	
+LAO+01	P18								
+LAO+02									
+LAO+03									
...									

Display Memory

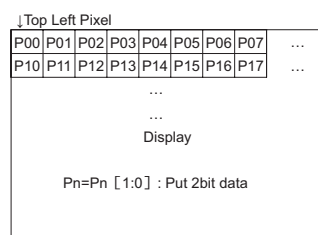


LAO: Line Address Offset
 -Unused bits should be 0

2. Packed 2bpp (Pixel Alignment in Byte is Big Endian) [Windows CE Recommended Format]

Address	MSB							LSB	[Bit]
	7	6	5	4	3	2	1	0	
+00	P00		P01		P02		P03		(Byte0)
+01	P04		P05		P06		P07		(Byte1)
+02									
+03									
...									
+LAO+00	P10		P11		P12		P13		
+LAO+01	P14		P15		P16		P17		
+LAO+02									
+LAO+03									
...									

Display Memory

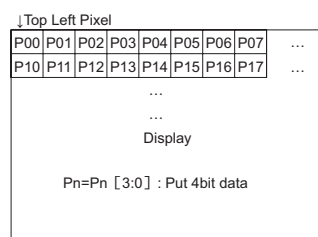


LAO: Line Address Offset
 -Unused bits should be 0

3. Packed 4bpp (Pixel Alignment in Byte is Big Endian) [Windows CE Recommended Format]

Address	MSB							LSB	[Bit]
	7	6	5	4	3	2	1	0	
+00	P00								(Byte0)
+01	P02								(Byte1)
+02	P04								(Byte2)
+03									
...									
+LAO+00	P10								
+LAO+01	P12								
+LAO+02	P14								
+LAO+03									
...									

Display Memory

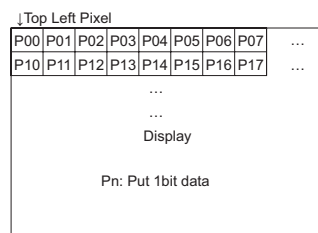


LAO: Line Address Offset
 -Unused bits should be 0

4. Packed 1bpp (Pixel Alignment in Byte is Little Endian)

Address	MSB							LSB	[Bit]
	7	6	5	4	3	2	1	0	
+00	P07	P06	P05	P04	P03	P02	P01	P00	(Byte0)
+01								P08	(Byte1)
+02									
+03									
...									
+LAO+00	P17	P16	P15	P14	P13	P12	P11	P10	
+LAO+01								P18	
+LAO+02									
+LAO+03									
...									

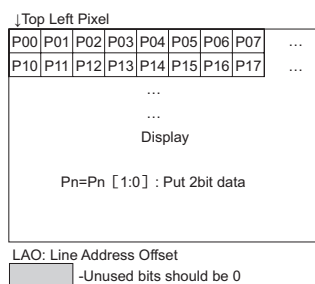
Display Memory



LAO: Line Address Offset
 -Unused bits should be 0

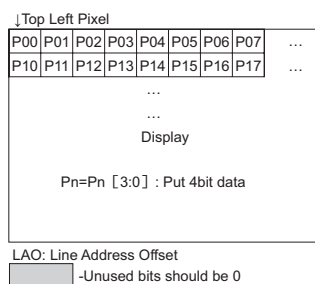
5. Packed 2bpp (Pixel Alignment in Byte is Little Endian)

	MSB							LSB	
Address	7	6	5	4	3	2	1	0	[Bit]
+00	P03		P02		P01		P00		(Byte0)
+01	P07		P06		P05		P04		(Byte1)
+02									
+03									
...									
+LAO+00	P13		P12		P11		P10		
+LAO+01	P17		P16		P15		P14		
+LAO+02									
+LAO+03									
...									
	Display Memory								



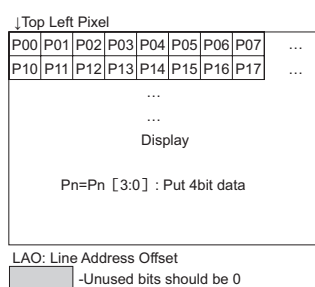
6. Packed 4bpp (Pixel Alignment in Byte is Little Endian)

	MSB							LSB		
Address	7	6	5	4	3	2	1	0	[Bit]	
+00	P01				P00				(Byte0)	
+01	P03				P02				(Byte1)	
+02	P05				P04				(Byte2)	
+03										
...	...									
+LAO+00	P11				P10					
+LAO+01	P13				P12					
+LAO+02	P15				P14					
+LAO+03	...									
	Display Memory									



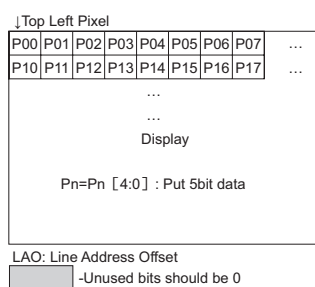
7. Unpacked 4bpp [Windows CE Recommended Format]

	MSB				LSB				
Address	7	6	5	4	3	2	1	0	[Bit]
+00								P00	(Byte0)
+01								P01	(Byte1)
+02								P02	(Byte2)
+03									
...									
+LAO+00								P10	
+LAO+01								P11	
+LAO+02								P12	
+LAO+03									
...									
	Display Memory								



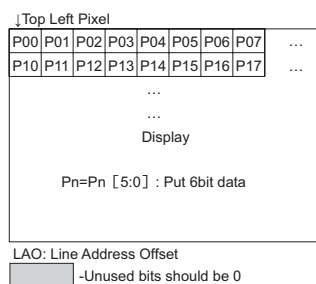
8. Unpacked 5bpp [Windows CE Recommended Format]

	MSB				LSB				
Address	7	6	5	4	3	2	1	0	[Bit]
+00								P00	(Byte0)
+01								P01	(Byte1)
+02								P02	(Byte2)
+03									
...									
+LAO+00								P10	
+LAO+01								P11	
+LAO+02								P12	
+LAO+03									
...									
	Display Memory								



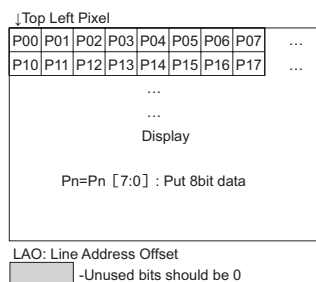
9.Unpacked 6bpp [Windows CE Recommended Format]

	MSB							LSB		
Address	7	6	5	4	3	2	1	0	[Bit]	
+00									(Byte0)	
+01									(Byte1)	
+02									(Byte2)	
+03										
...										
+LAO+00									P10	
+LAO+01									P11	
+LAO+02									P12	
+LAO+03										
...										
	Display Memory									



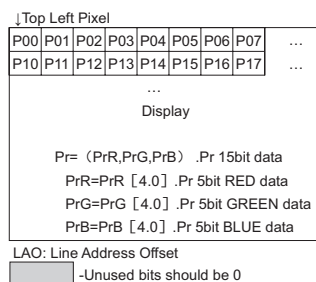
10.Packed 8bpp [Windows CE Recommended Format]

	MSB							LSB		
Address	7	6	5	4	3	2	1	0	[Bit]	
+00	P00								(Byte0)	
+01	P01								(Byte1)	
+02	P02								(Byte2)	
+03										
...	...									
+LAO+00	P10									
+LAO+01	P11									
+LAO+02	P12									
+LAO+03	...									
	Display Memory									



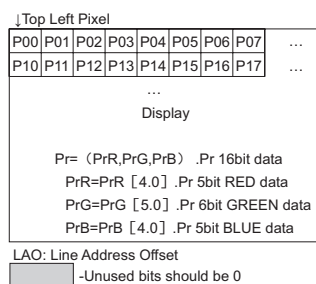
11.Unpacked color 15bpp (RGB 555) [Windows CE Recommended Format]

	MSB															LSB		
Address	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	[Bit]	
+00			P00R				P00G				P00B				(Word0)			
+02			P01R				P01G				P01B				(Word2)			
+04			P02R				P02G				P02B				(Word4)			
+06																		
...	...																	
+LAO+00			P10R				P10G				P10B							
+LAO+02			P11R				P11G				P11B							
+LAO+04			P12R				P12G				P12B							
+LAO+06																		
...	...																	
	Display Memory																	



12.Packed color 16bpp (RGB 565) [Windows CE Recommended Format]

	MSB															LSB		
Address	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	[Bit]	
+00	P00R					P00G					P00B					(Word0)		
+02	P01R					P01G					P01B					(Word2)		
+04	P02R					P02G					P02B					(Word4)		
+06																		
...	...																	
+LAO+00	P10R					P10G					P10B							
+LAO+02	P11R					P11G					P11B							
+LAO+04	P12R					P12G					P12B							
+LAO+06																		
...	...																	
	Display Memory																	



24.4.5 显示分辨率的设定

由 LDHCNR、LDHSYNR、LDVDLNR、LDVTLNR、LDVSYNR 设定显示分辨率。由 LDACLNR 设定 STN 或 DSTN 显示时的液晶交流周期。这些寄存器的初始值为 VGA (640×480 点)、STN 或 DSTN 显示的典型的分辨率设定值。

由 LDICKR 设定使用的时钟。通过在与尺寸有关的寄存器设定的 1 个显示画面 + 回扫线期间 (非显示期间) 和所用时钟频率, 决定液晶模块的帧速率。另外, 本 LCDC 具有在每个垂直回扫线期间的开始点 (确切地说, 是最后显示行的下一行的开始点) 产生中断的 Vsync 中断功能。使用 LDINTR 设定该功能。

24.4.6 电源控制顺序处理

通常液晶模块需处理与上电断电有关的特定顺序。通过设定 LDPMMR、LDPSPR、LDCNTR, 使用液晶电源控制引脚 (LCD_VCPWC、LCD_VEPWC、LCD_DON), 对应液晶模块请求可执行多种电源控制顺序处理。

电源控制顺序的简略时序如图 24.4 ~ 图 24.7 所示, 可设定的电源控制顺序期间说明如表 24.6 所示。

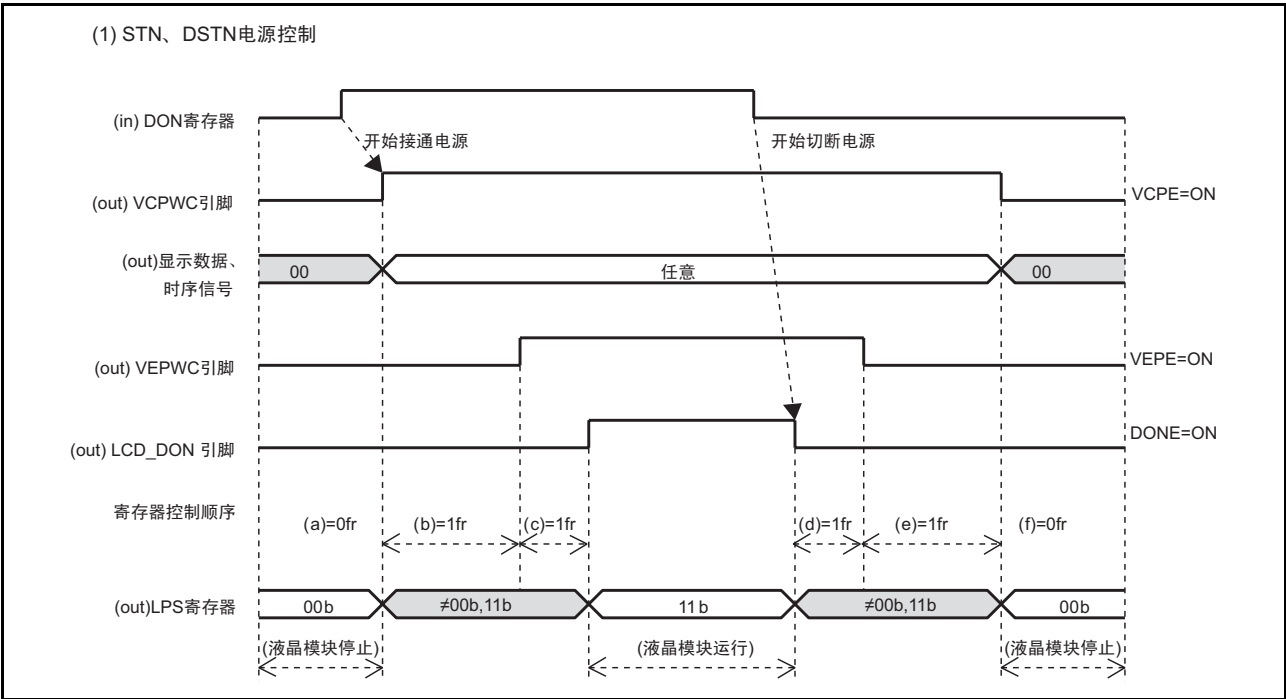


图 24.4 电源控制顺序和液晶模块的运行状态

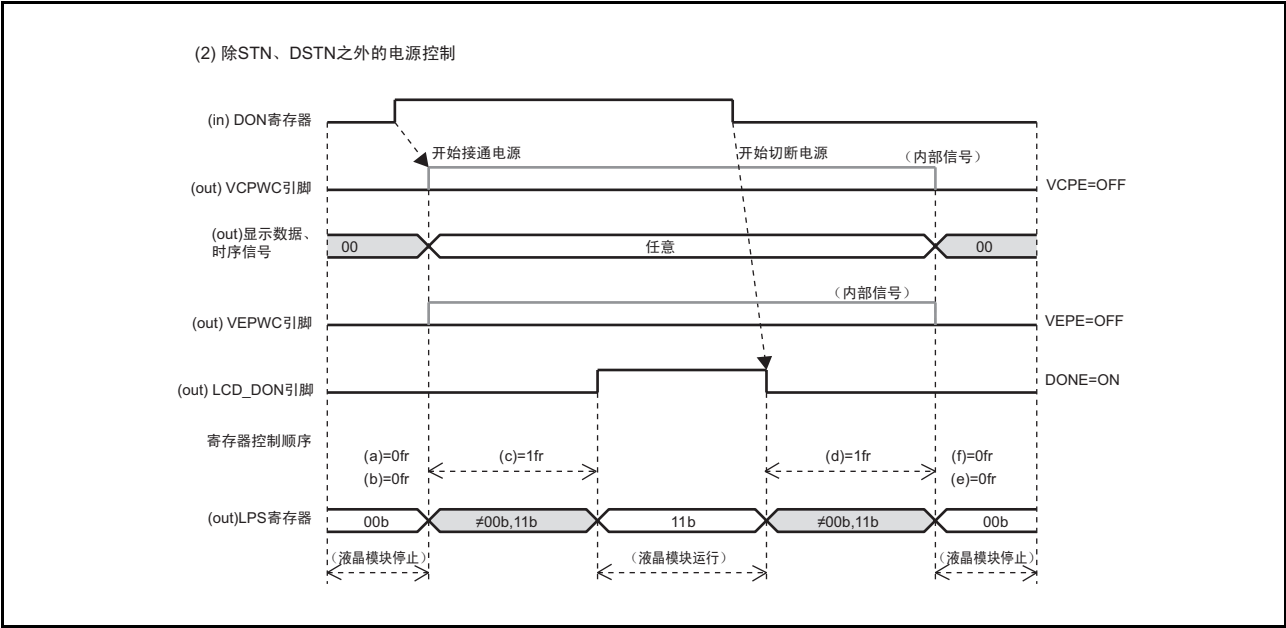


图 24.5 电源控制顺序和液晶模块的运行状态

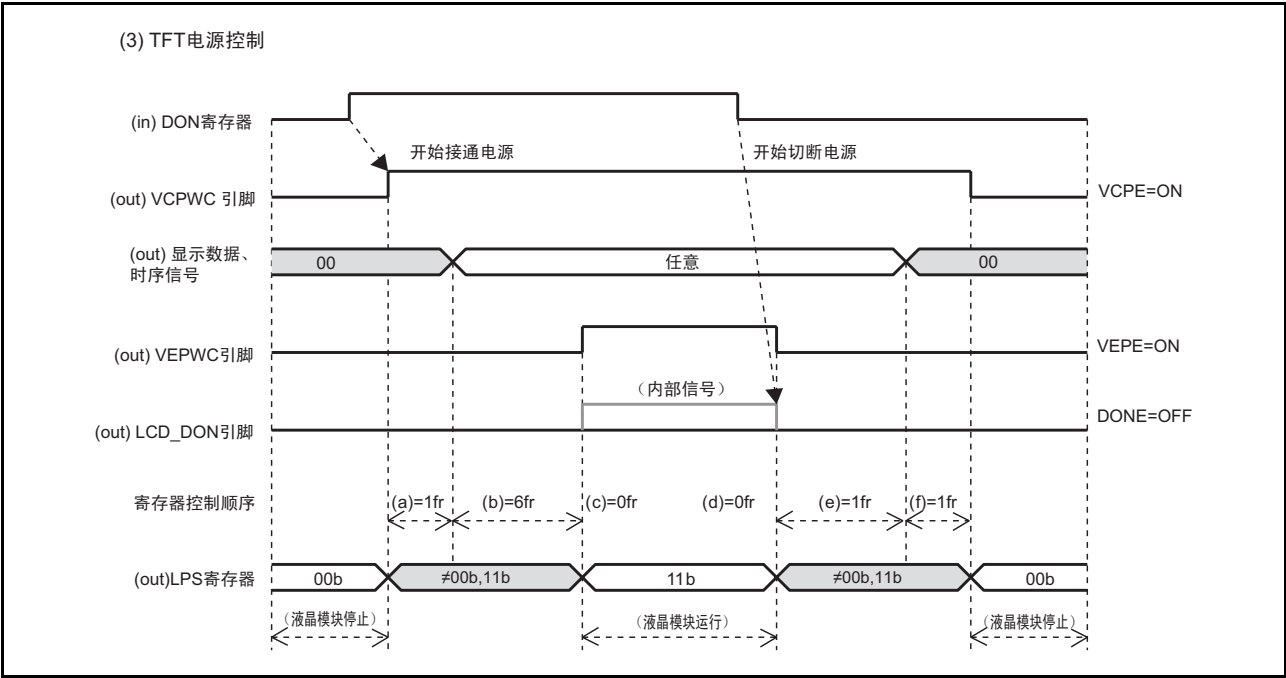


图 24.6 电源控制顺序和液晶模块的运行状态

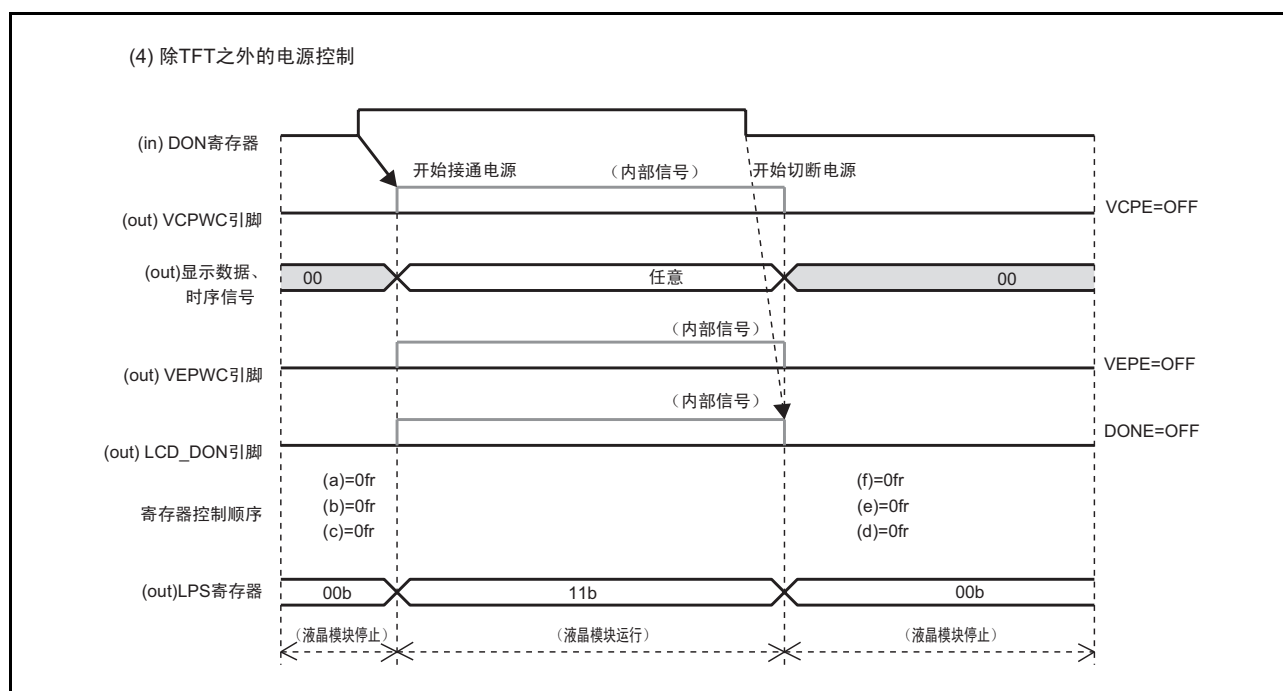


图 24.7 电源控制顺序和液晶模块的运行状态

表 24.6 典型帧速率中可设定的电源控制顺序期间

ONX、OFFX 寄存器设定值	帧速率	
	120Hz	60Hz
H'F	$(-1+1)/120 = 0.00(\text{ms})$	$(-1+1)/60 = 0.00(\text{ms})$
H'0	$(0+1)/120 = 8.33(\text{ms})$	$(0+1)/60 = 16.67(\text{ms})$
H'1	$(1+1)/120 = 16.67(\text{ms})$	$(1+1)/60 = 33.33(\text{ms})$
H'2	$(2+1)/120 = 25.00(\text{ms})$	$(2+1)/60 = 50.00(\text{ms})$
H'3	$(3+1)/120 = 33.33(\text{ms})$	$(3+1)/60 = 66.67(\text{ms})$
H'4	$(4+1)/120 = 41.67(\text{ms})$	$(4+1)/60 = 83.33(\text{ms})$
H'5	$(5+1)/120 = 50.00(\text{ms})$	$(5+1)/60 = 100.00(\text{ms})$
H'6	$(6+1)/120 = 58.33(\text{ms})$	$(6+1)/60 = 116.67(\text{ms})$
H'7	$(7+1)/120 = 66.67(\text{ms})$	$(7+1)/60 = 133.33(\text{ms})$
H'8	$(8+1)/120 = 75.00(\text{ms})$	$(8+1)/60 = 150.00(\text{ms})$
H'9	$(9+1)/120 = 83.33(\text{ms})$	$(9+1)/60 = 166.67(\text{ms})$
H'A	$(10+1)/120 = 91.67(\text{ms})$	$(10+1)/60 = 183.33(\text{ms})$
H'B	$(11+1)/120 = 100.00(\text{ms})$	$(11+1)/60 = 200.00(\text{ms})$
H'C	$(12+1)/120 = 108.33(\text{ms})$	$(12+1)/60 = 216.67(\text{ms})$
H'D	$(13+1)/120 = 116.67(\text{ms})$	$(13+1)/60 = 233.33(\text{ms})$
H'E	$(14+1)/120 = 125.00(\text{ms})$	$(14+1)/60 = 250.00(\text{ms})$

ONA、ONB、ONC、OFFD、OFFE、OFFF 等寄存器以帧周期为单位，可设定 0 ~ 15 帧的电源控制顺序期间。寄存器设定为 -1 后的值，H'0 ~ H'E 设定表示 1 ~ 15 帧，H'F 设定表示 0 帧。

实际的顺序时间取决于寄存器设定值和显示帧频率。下表为典型液晶模块所使用的显示帧频率的电源控制顺序期间。

- 显示帧频率为 120Hz，ONB 寄存器设定为 H'6 时
显示帧频率为 120Hz，1 帧的时间为 $8.33(\text{ms})=1/120(\text{sec})$ 。
ONB 寄存器设定为 -1 后的值，上电顺序期间为 7 帧。
因此，此时的顺序时间为 $58.33(\text{ms})=8.33(\text{ms})\times 7$ 。

表 24.7 LCDC 运行模式

模式		功 能
显示 ON (LCDC 运行)	寄存器设定: DON=1	根据规定的分辨率、色数，向液晶模块输出显示数据、时序信号
显示 OFF (LCDC 停止)	寄存器设定: DON=0	可存取寄存器。 根据规定的分辨率、色数，不向液晶模块输出显示数据、时序信号

表 24.8 液晶模块电源状态

(STN、DSTN 模块时)

状态	逻辑电源	显示数据时序信号	高压电源	DON 信号
相应的控制引脚	LCD_VCPWC	LCD_CL2、LCD_CL1、 LCD_FLM、LCD_M_DISP、 LCD_DATA	LCD_VEPWC	LCD_DON
运行状态	提供	提供	提供	提供
(过渡状态)	提供	提供	提供	
	提供	提供		
	提供			
停止状态				

(TFT 模块时)

状态	逻辑电源	显示数据时序信号	高压电源
相应的控制引脚	LCD_VCPWC	LCD_CL2、LCD_CL1、LCD_FLM、 LCD_M_DISP、LCD_DATA	LCD_VEPWC
运行状态	提供	提供	提供
(过渡状态)	提供	提供	
	提供		
停止状态			

以上为典型液晶模块在运行及停止状态，电源、显示数据及时序信号的供给状态。根据模块不同，也可能在模块内部从逻辑电源电压生成高压电源，因此不一定需要提供电源。

- 显示 OFF 模式 (LCDC 停止) 的注意事项

通过本 LCDC 使用液晶模块电源控制顺序处理时，如果在显示 ON 模式状态断电，则无法保证 LCDC 的正常运行。最坏的可能是损坏连接的液晶模块。

24.4.7 硬件旋转运行说明

以下是硬件旋转运行的说明。硬件旋转模式的使用设想为“将有横向画面的液晶面板纵向放置，代替纵向液晶面板使用”。面板的形状无论是横向或纵向均可，但宽度不能超过 320。

硬件旋转时，必须更改以下 5 项无硬件旋转设定（以下例子是 8bpp 时的情况。16bpp 时，每点的存储器尺寸为 2 倍，因此图像的尺寸及旋转时寄存器的设定值不同）。

1. 图像具备已旋转面板显示的形状（如果旋转后为 240×320 像素，则需具备未旋转的 240×320 尺寸的数据）
2. 更改与图像地址有关的寄存器设定（LDSARU、LDLAOR）。
3. LDLAOR 为 2 乘方（旋转后宽度为 240 时，LDLAOR 为 256）。
4. 图形软件也符合 3. 的设定。
5. 将 LDSARU 从图像的左上地址更改为左下地址。

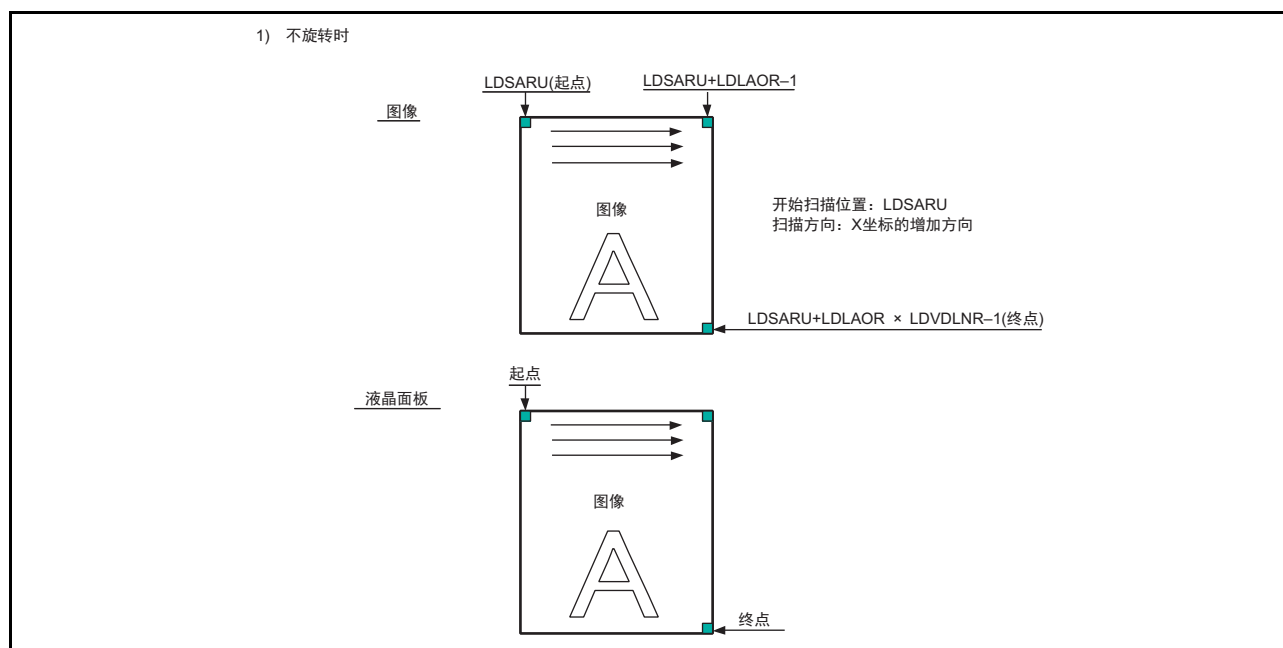


图 24.8 硬件旋转运行（不旋转时）

例如，对尺寸为 320×240 的液晶面板，已设定显示从 LDSARU=0x0c001000 开始的横向图像（320×240）寄存器、并完成图形驱动软件。在此状态使用硬件旋转，并尝试更改为 240×320 显示。此时，如果 LDLAOR=512，则图形驱动软件也应将计算图像 Y 坐标地址的偏移设置为 2 的乘方。ROT=1 前，为了可用于 240×320，需重画图像。另外，由于尺寸不同，LDLAOR=256，也要更改图形驱动软件。LDSARU 从左上变为左下，因此更改为 LDSARU=0x0c001000+256*319。

【注】 硬件旋转功能可将液晶面板旋转 90 度使用，液晶面板自身的设定需符合旋转前的液晶面板。另外，旋转时可不考虑图形驱动软件的绘画处理，但图形驱动软件管理图像的尺寸及地址偏移值需符合实际图像。

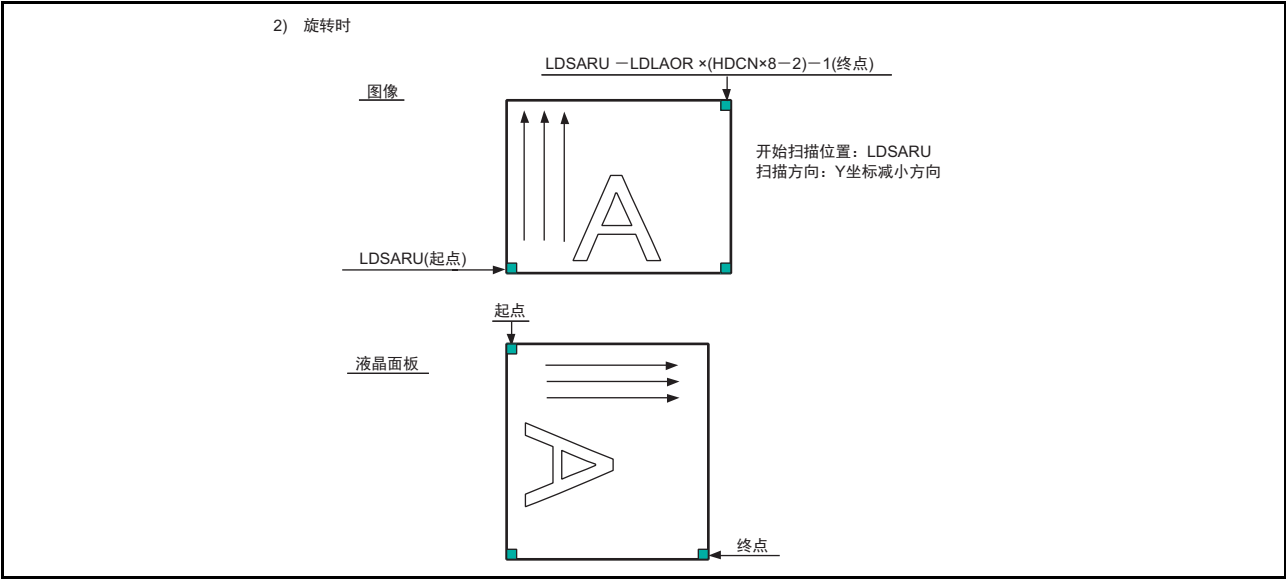


图 24.9 硬件旋转运行 (旋转时)

24.5 时钟和 LCD 数据信号例

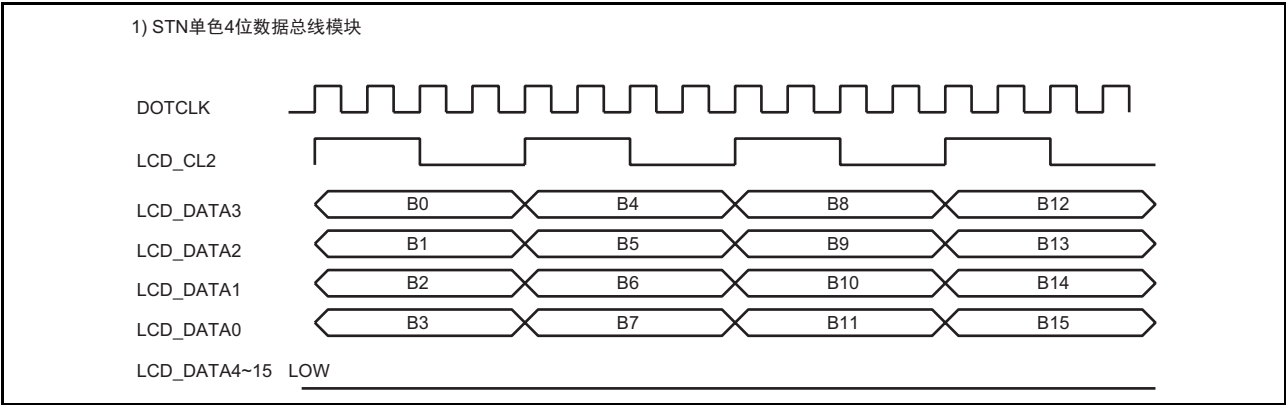


图 24.10 时钟和 LCD 数据信号例 (STN 单色 4 位数据总线模块)

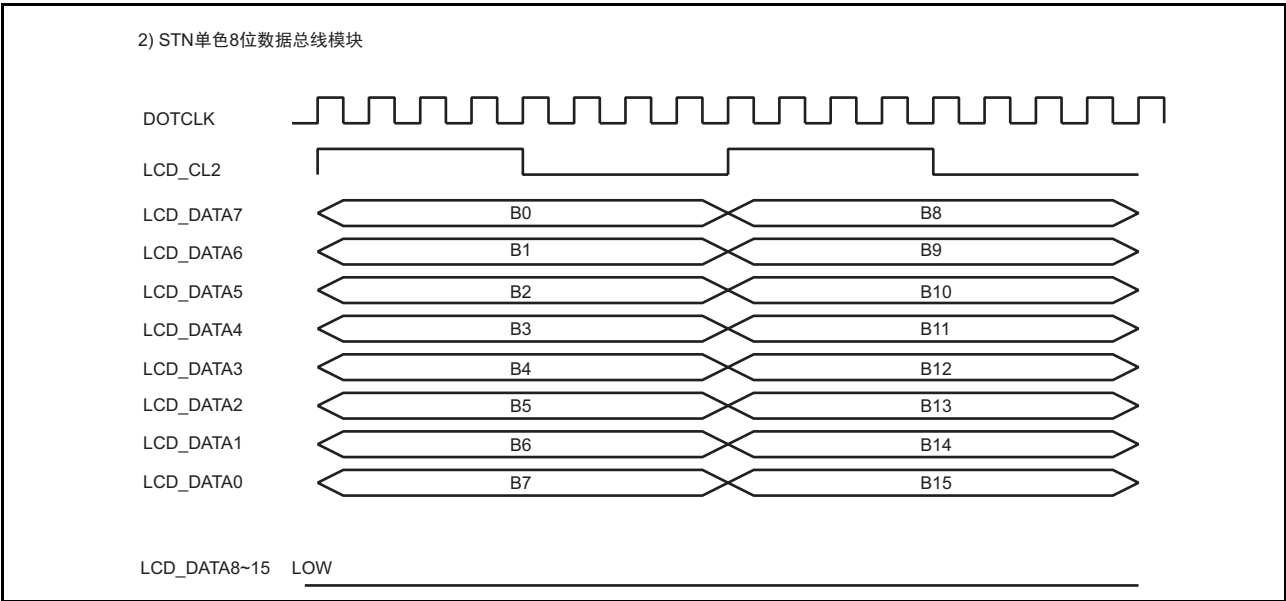


图 24.11 时钟和 LCD 数据信号例 (STN 单色 8 位数据总线模块)

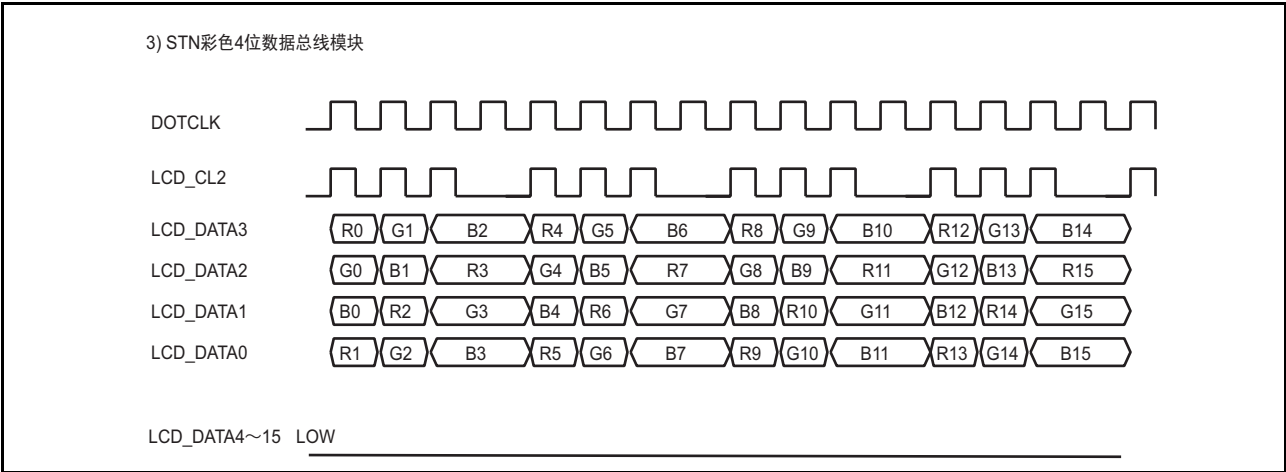


图 24.12 时钟和 LCD 数据信号例 (STN 彩色 4 位数据总线模块)

4) STN彩色8位数据总线模块

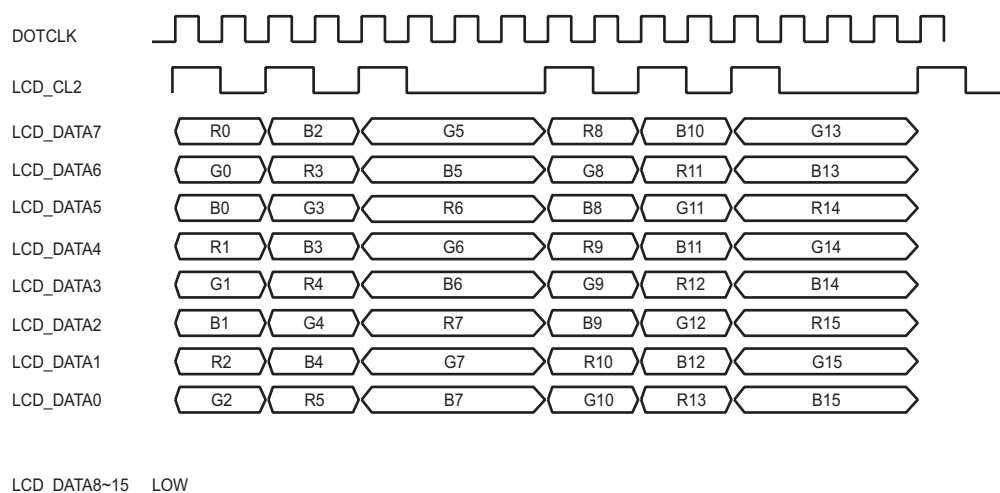


图 24.13 时钟和 LCD 数据信号例 (STN 彩色 8 位数据总线模块)

5) STN彩色12位数据总线模块

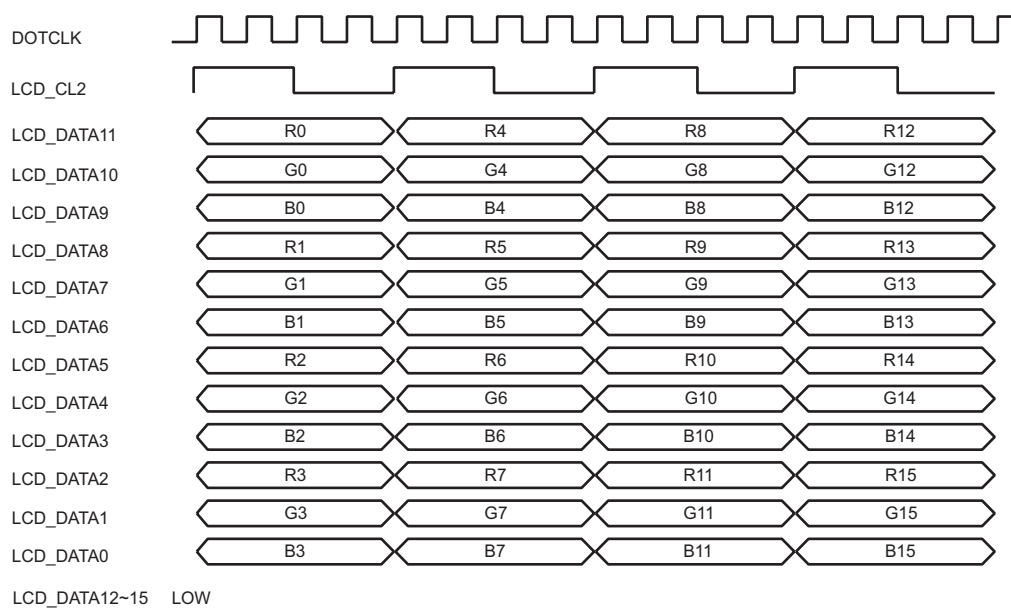


图 24.14 时钟和 LCD 数据信号例 (STN 彩色 12 位数据总线模块)

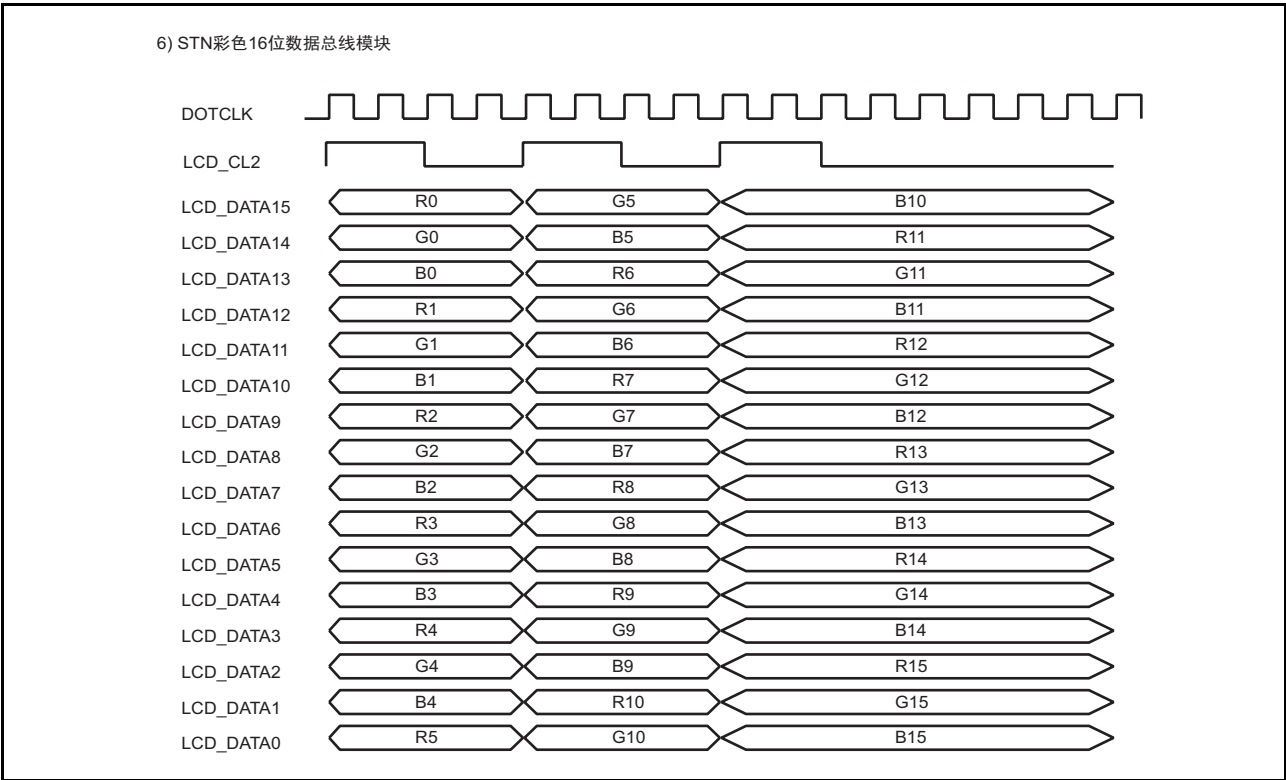


图 24.15 时钟和 LCD 数据信号例 (STN 彩色 16 位数据总线模块)

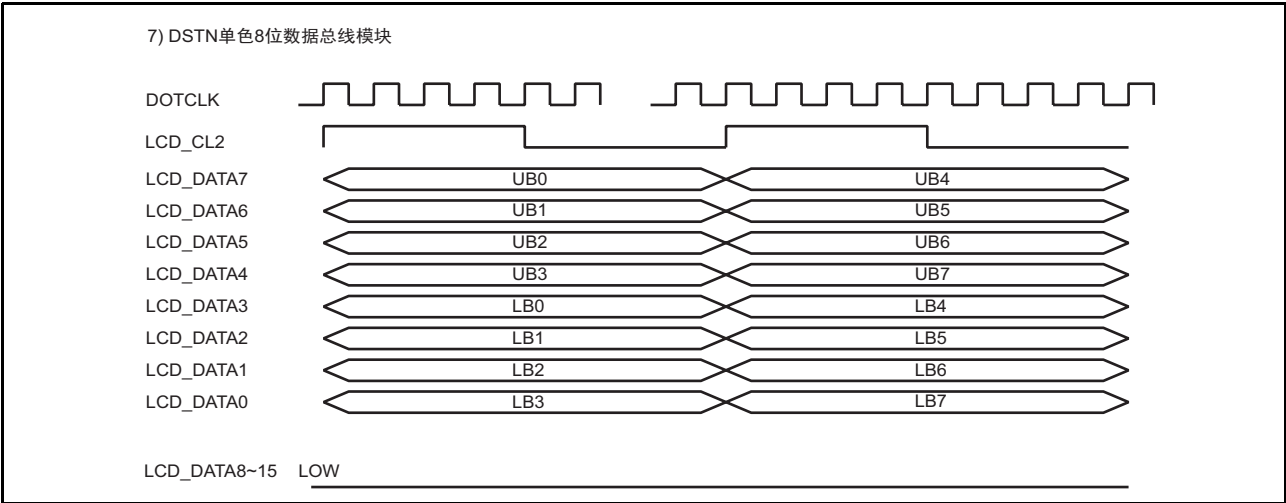


图 24.16 时钟和 LCD 数据信号例 (DSTN 单色 8 位数据总线模块)

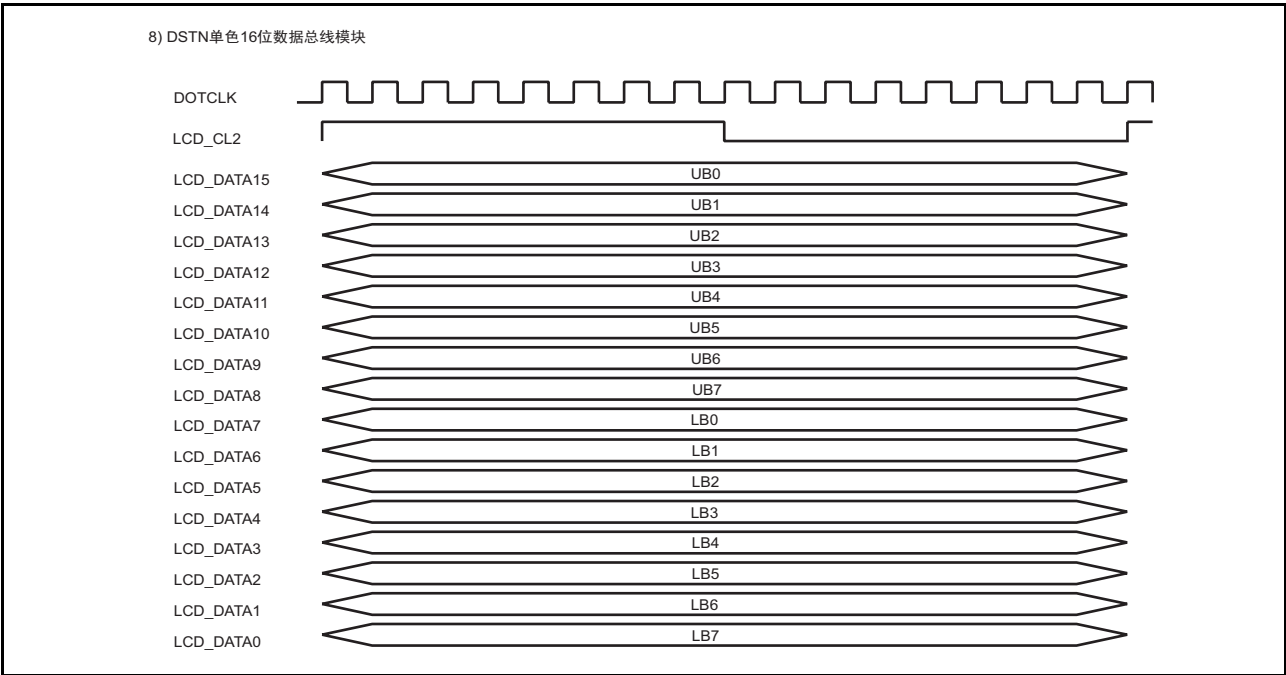


图 24.17 时钟和 LCD 数据信号例 (DSTN 单色 16 位数据总线模块)

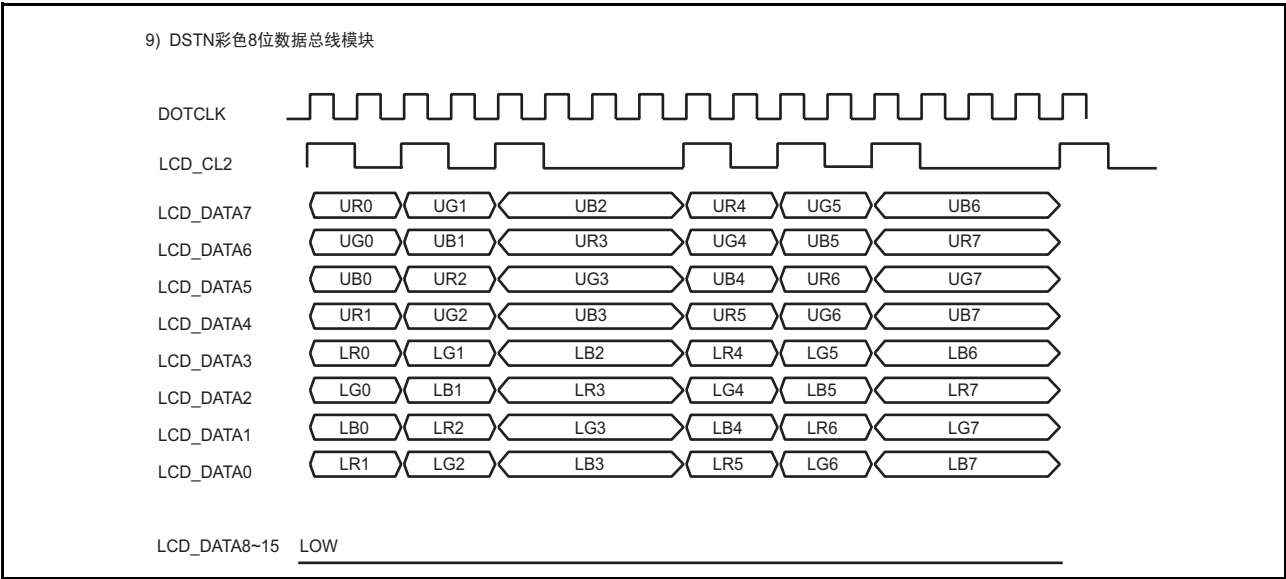


图 24.18 时钟和 LCD 数据信号例 (DSTN 彩色 8 位数据总线模块)

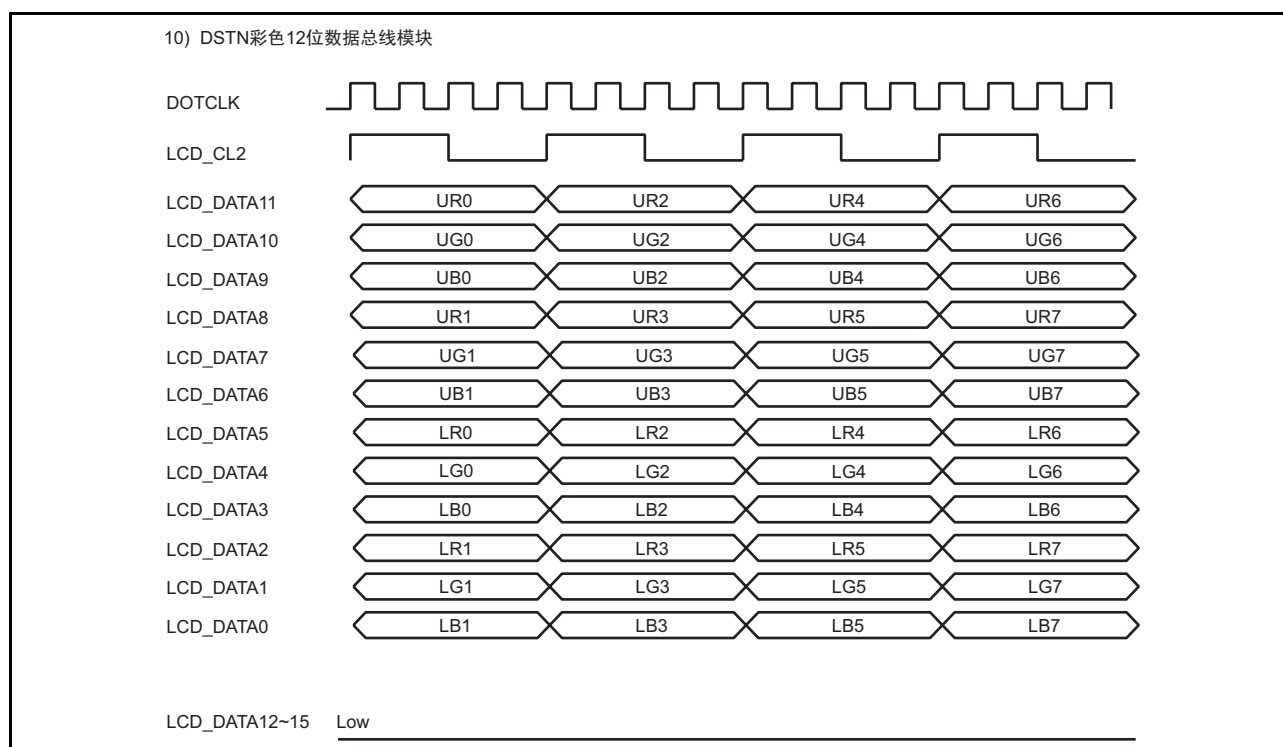


图 24.19 时钟和 LCD 数据信号例 (DSTN 彩色 12 位数据总线模块)

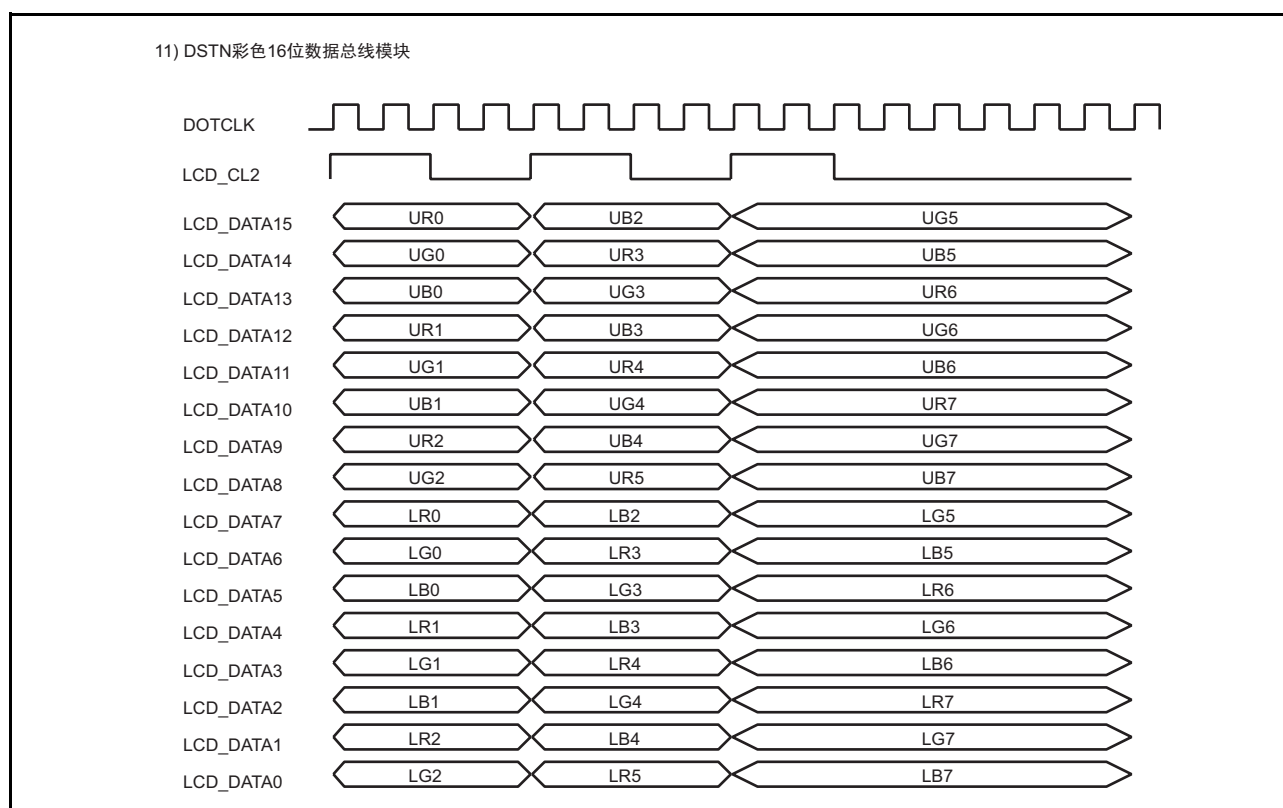


图 24.20 时钟和 LCD 数据信号例 (DSTN 彩色 16 位数据总线模块)

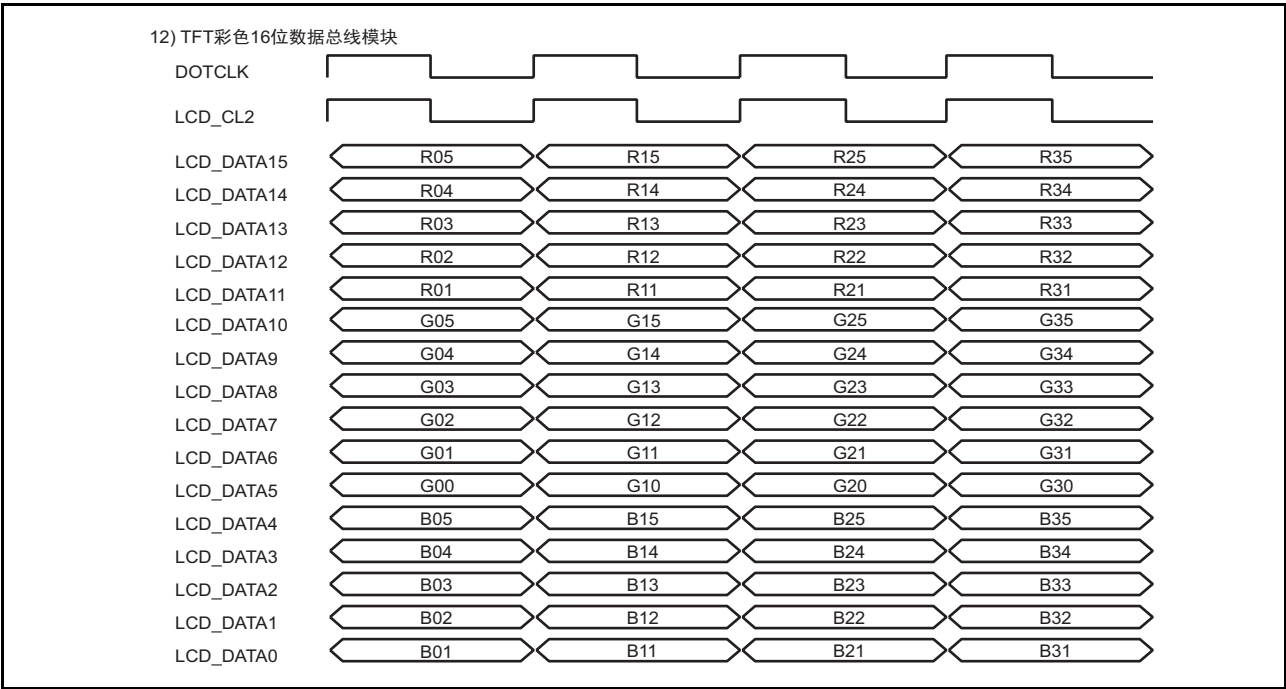


图 24.21 时钟和 LCD 数据信号例 (TFT 彩色 16 位数据总线模块)

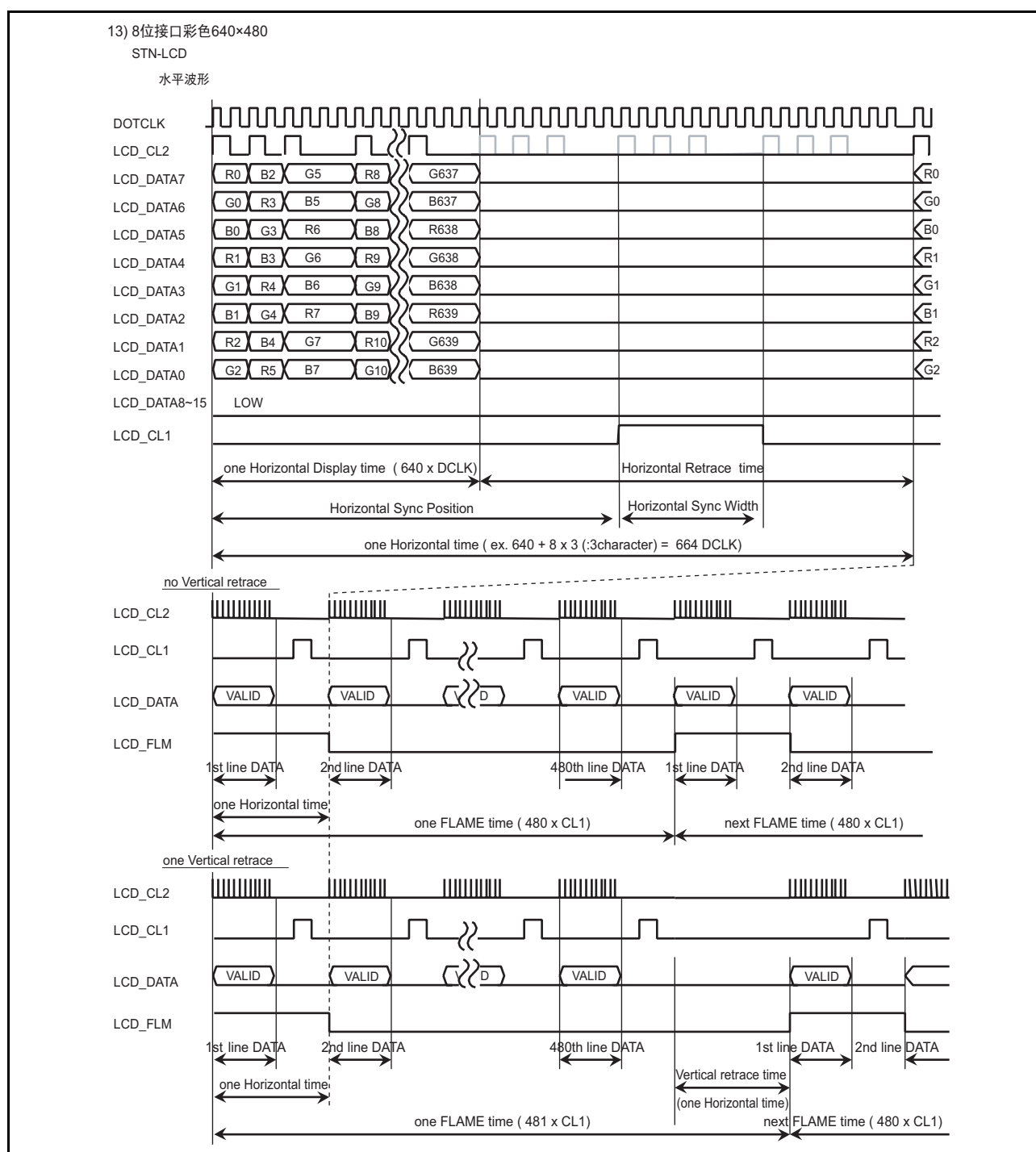


图 24.22 时钟和 LCD 数据信号例 (8 位接口彩色 640×480)

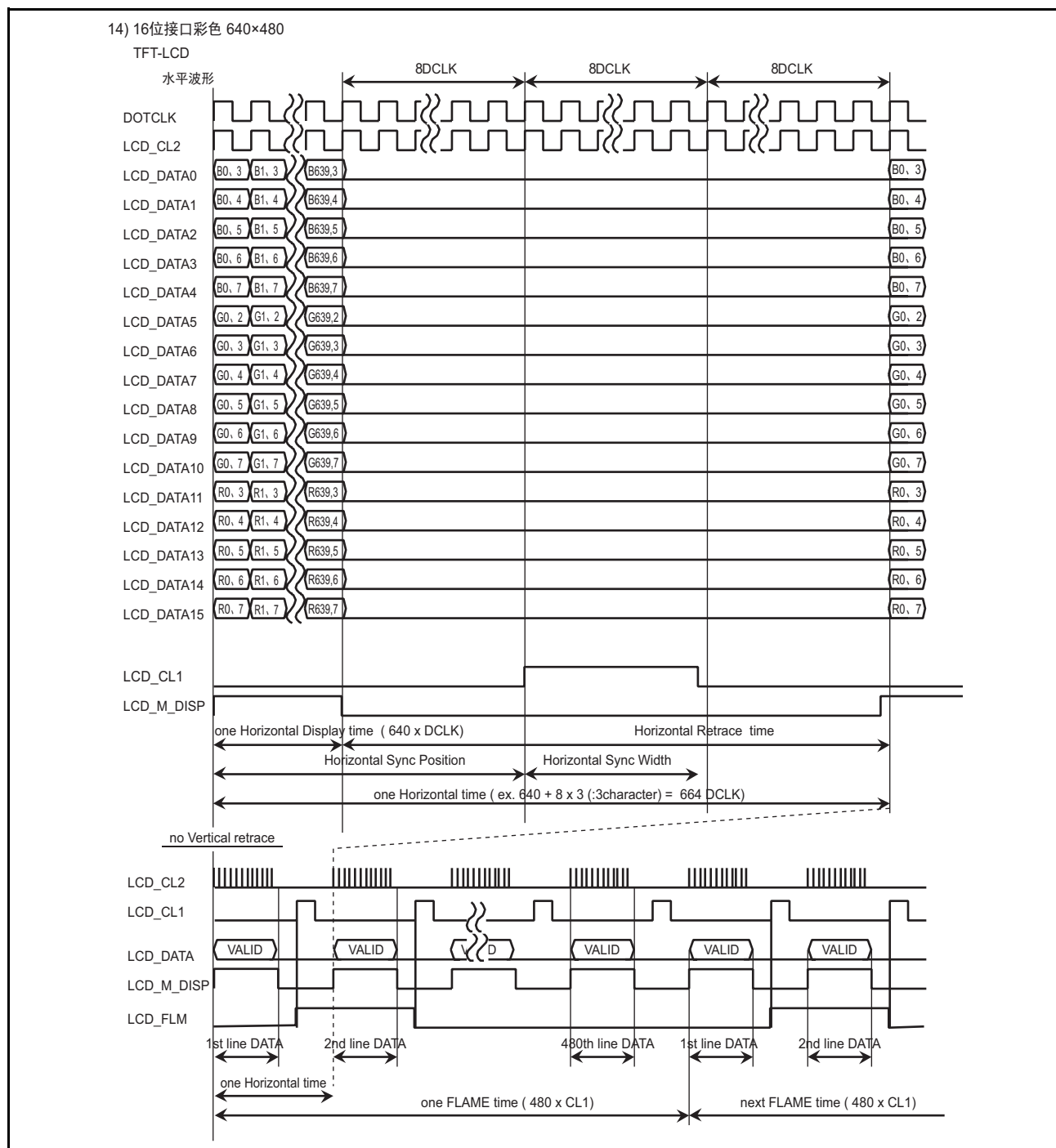


图 24.23 时钟和 LCD 数据信号例 (16 位接口彩色 640×480)

24.6 使用时的注意事项

24.6.1 存取保存显示数据的 VRAM（区域 3 的同步 DRAM）的停止步骤

存取保存显示数据的 VRAM（区域 3 的同步 DRAM）的停止步骤，必须按照以下流程。

- 存取保存显示数据的 VRAM 的停止步骤
 1. 确认当前状态为 LDPMMR 的 LPS1 和 LPS0 位为 1。
 2. LDCNTR 的 DON 位设定为 0（显示关闭模式）。
 3. 确认 LDPMMR 的 LPS1 和 LPS0 位为 0。
 4. 等待 1 帧的显示时间。

保存显示数据的 VRAM（区域 3 的同步 DRAM）自刷新前，或进入待机模式、模块待机前，需执行该停止步骤。

第 25 章 引脚功能控制器（PFC）

引脚功能控制器（PFC）由选择多路复用引脚的功能与其输入 / 输出方向的寄存器构成。本 LSI 的多路复用引脚如表 25.1 ～表 25.6 所示。

表 25.1 多路复用引脚一览表（端口 A）

端口	功能 1（相关模块）	功能 2（相关模块）	功能 3（相关模块）
A	PA7 输入（端口）	AN7 输入（ADC）	DA1 输出（DAC）
	PA6 输入（端口）	AN6 输入（ADC）	DA0 输出（DAC）
	PA5 输入（端口）	AN5 输入（ADC）	—
	PA4 输入（端口）	AN4 输入（ADC）	—
	PA3 输入（端口）	AN3 输入（ADC）	—
	PA2 输入（端口）	AN2 输入（ADC）	—
	PA1 输入（端口）	AN1 输入（ADC）	—
	PA0 输入（端口）	AN0 输入（ADC）	—

【注】 可在通用输入功能、A/D 转换器模拟输入功能及 D/A 转换器模拟输出功能间自动转换，因此 PFC 中不存在用于设定的寄存器。

表 25.2 多路复用一览表 (端口 B)

设定 寄存器	模式位 (PBnMD[1:0]) 设定值			
	00	01	10	11
	功能 1 (相关模块)	功能 2 (相关模块)	功能 3 (相关模块)	功能 4 (相关模块)
PBCRL4	PB12 输出 (端口)	WDTOVF 输出 (WDT)	IRQOUT/REFOUT 输出 (INTC/BSC)	UBCTRG 输出 (UBC)
PBCRL3	PB11 输入 / 输出 (端口)	CTx1 输出 (RCAN-TL1)	—	—
	PB10 输入 / 输出 (端口)	CRx1 输入 (RCAN-TL1)	—	—
	PB9 输入 / 输出 (端口)	CTx0 输出 (RCAN-TL1)	CTx0&CTx1 输出 (RCAN-TL1)	—
	PB8 输入 / 输出 (端口)	CRx0 输入 (RCAN-TL1)	CRx0/CRx1 输入 (RCAN-TL1)	—
PBCRL2	PB7 输入 (端口)	SDA3 输入 / 输出 (IIC3)	PINT7 输入 (INTC)	IRQ7 输入 (INTC)
	PB6 输入 (端口)	SCL3 输入 / 输出 (IIC3)	PINT6 输入 (INTC)	IRQ6 输入 (INTC)
	PB5 输入 (端口)	SDA2 输入 / 输出 (IIC3)	PINT5 输入 (INTC)	IRQ5 输入 (INTC)
	PB4 输入 (端口)	SCL2 输入 / 输出 (IIC3)	PINT4 输入 (INTC)	IRQ4 输入 (INTC)
PBCRL1	PB3 输入 (端口)	SDA1 输入 / 输出 (IIC3)	PINT3 输入 (INTC)	IRQ3 输入 (INTC)
	PB2 输入 (端口)	SCL1 输入 / 输出 (IIC3)	PINT2 输入 (INTC)	IRQ2 输入 (INTC)
	PB1 输入 (端口)	SDA0 输入 / 输出 (IIC3)	PINT1 输入 (INTC)	IRQ1 输入 (INTC)
	PB0 输入 (端口)	SCL0 输入 / 输出 (IIC3)	PINT0 输入 (INTC)	IRQ0 输入 (INTC)

表 25.3 多路复用一览表 (端口 C)

设定 寄存器	模式位 (PCnMD[1:0]) 设定值			
	00	01	10	11
	功能 1 (相关模块)	功能 2 (相关模块)	功能 3 (相关模块)	—
PCCRL4	PC14 输入 / 输出 (端口)	WAIT 输入 (BSC)	—	—
	PC13 输入 / 输出 (端口)	RDWR 输出 (BSC)	—	—
	PC12 输入 / 输出 (端口)	CKE 输出 (BSC)	—	—
PCCRL3	PC11 输入 / 输出 (端口)	CASU 输出 (BSC)	BREQ 输入 (BSC)	—
	PC10 输入 / 输出 (端口)	RASU 输出 (BSC)	BACK 输出 (BSC)	—
	PC9 输入 / 输出 (端口)	CASL 输出 (BSC)	—	—
	PC8 输入 / 输出 (端口)	RASL 输出 (BSC)	—	—
PCCRL2	PC7 输入 / 输出 (端口)	WE3/DQMUU/AH/ICIOWR 输出 (BSC)	—	—
	PC6 输入 / 输出 (端口)	WE2/DQMUL/ICIORD 输出 (BSC)	—	—
	PC5 输入 / 输出 (端口)	WE1/DQMLU/WE 输出 (BSC)	—	—
	PC4 输入 / 输出 (端口)	WE0/DQMLL 输出 (BSC)	—	—
PCCRL1	PC3 输入 / 输出 (端口)	CS3 输出 (BSC)	—	—
	PC2 输入 / 输出 (端口)	CS2 输出 (BSC)	—	—
	PC1 输入 / 输出 (端口)	A1 输出 (地址)	—	—
	PC0 输入 / 输出 (端口)	A0 输出 (地址)	CS7 输出 (BSC)	—

表 25.4 多路复用一览表 (端口 D)

设定 寄存器	模式位 (PDnMD[2:0]) 设定值						
	000	001	010	011	100	101	110/111
	功能 1 (相关模块)	功能 2 (相关模块)	功能 3 (相关模块)	功能 4 (相关模块)	功能 5 (相关模块)	功能 6 (相关模块)	—
PDCRL4	PD15 输入 / 输出 (端口)	D31 输入 / 输出 (数据)	PINT7 输入 (INTC)	—	ADTRG 输入 (ADC)	TIOC4D 输入 / 输出 (MTU2)	—
	PD14 输入 / 输出 (端口)	D30 输入 / 输出 (数据)	PINT6 输入 (INTC)	—	—	TIOC4C 输入 / 输出 (MTU2)	—
	PD13 输入 / 输出 (端口)	D29 输入 / 输出 (数据)	PINT5 输入 (INTC)	—	TEND1 输出 (DMAC)	TIOC4B 输入 / 输出 (MTU2)	—
	PD12 输入 / 输出 (端口)	D28 输入 / 输出 (数据)	PINT4 输入 (INTC)	—	DACK1 输出 (DMAC)	TIOC4A 输入 / 输出 (MTU2)	—
PDCRL3	PD11 输入 / 输出 (端口)	D27 输入 / 输出 (数据)	PINT3 输入 (INTC)	—	DREQ1 输入 (DMAC)	TIOC3D 输入 / 输出 (MTU2)	—
	PD10 输入 / 输出 (端口)	D26 输入 / 输出 (数据)	PINT2 输入 (INTC)	—	TEND0 输出 (DMAC)	TIOC3C 输入 / 输出 (MTU2)	—
	PD9 输入 / 输出 (端口)	D25 输入 / 输出 (数据)	PINT1 输入 (INTC)	—	DACK0 输出 (DMAC)	TIOC3B 输入 / 输出 (MTU2)	—
	PD8 输入 / 输出 (端口)	D24 输入 / 输出 (数据)	PINT0 输入 (INTC)	—	DREQ0 输入 (DMAC)	TIOC3A 输入 / 输出 (MTU2)	—
PDCRL2	PD7 输入 / 输出 (端口)	D23 输入 / 输出 (数据)	IRQ7 输入 (INTC)	SCS1 输入 / 输出 (SSU)	TCLKD 输入 (MTU2)	TIOC2B 输入 / 输出 (MTU2)	—
	PD6 输入 / 输出 (端口)	D22 输入 / 输出 (数据)	IRQ6 输入 (INTC)	SSO1 输入 / 输出 (SSU)	TCLKC 输入 (MTU2)	TIOC2A 输入 / 输出 (MTU2)	—
	PD5 输入 / 输出 (端口)	D21 输入 / 输出 (数据)	IRQ5 输入 (INTC)	SSI1 输入 / 输出 (SSU)	TCLKB 输入 (MTU2)	TIOC1B 输入 / 输出 (MTU2)	—
	PD4 输入 / 输出 (端口)	D20 输入 / 输出 (数据)	IRQ4 输入 (INTC)	SSCK1 输入 / 输出 (SSU)	TCLKA 输入 (MTU2)	TIOC1A 输入 / 输出 (MTU2)	—
PDCRL1	PD3 输入 / 输出 (端口)	D19 输入 / 输出 (数据)	IRQ3 输入 (INTC)	SCS0 输入 / 输出 (SSU)	DACK3 输出 (DMAC)	TIOC0D 输入 / 输出 (MTU2)	—
	PD2 输入 / 输出 (端口)	D18 输入 / 输出 (数据)	IRQ2 输入 (INTC)	SSO0 输入 / 输出 (SSU)	DREQ3 输入 (DMAC)	TIOC0C 输入 / 输出 (MTU2)	—
	PD1 输入 / 输出 (端口)	D17 输入 / 输出 (数据)	IRQ1 输入 (INTC)	SSI0 输入 / 输出 (SSU)	DACK2 输出 (DMAC)	TIOC0B 输入 / 输出 (MTU2)	—
	PD0 输入 / 输出 (端口)	D16 输入 / 输出 (数据)	IRQ0 输入 (INTC)	SSCK0 输入 / 输出 (SSU)	DREQ2 输入 (DMAC)	TIOC0A 输入 / 输出 (MTU2)	—

表 25.5 多路复用一览表 (端口 E)

设定 寄存器	模式位 (PEnMD[2:0]) 设定值					
	000	001	010	011	100	101/110/111
	功能 1 (相关模块)	功能 2 (相关模块)	功能 3 (相关模块)	功能 4 (相关模块)	功能 5 (相关模块)	—
PECRL4	PE15 输入 / 输出 (端口)	IOIS16 输入 (BSC)	—	RTS3 输入 / 输出 (SCIF)	—	—
	PE14 输入 / 输出 (端口)	CS1 输出 (BSC)	—	CTS3 输入 / 输出 (SCIF)	—	—
	PE13 输入 / 输出 (端口)	—	—	TxD3 输出 (SCIF)	—	—
	PE12 输入 / 输出 (端口)	—	—	RxD3 输入 (SCIF)	—	—
PECRL3	PE11 输入 / 输出 (端口)	CS6/CE1B 输出 (BSC)	IRQ7 输入 (INTC)	—	TEND1 输出 (DMAC)	—
	PE10 输入 / 输出 (端口)	CE2B 输出 (BSC)	IRQ6 输入 (INTC)	—	TEND0 输出 (DMAC)	—
	PE9 输入 / 输出 (端口)	CS5/CE1A 输出 (BSC)	IRQ5 输入 (INTC)	SCK3 输入 / 输出 (SCIF)	—	—
	PE8 输入 / 输出 (端口)	CE2A 输出 (BSC)	IRQ4 输入 (INTC)	SCK2 输入 / 输出 (SCIF)	—	—
PECRL2	PE7 输入 / 输出 (端口)	FRAME 输出 (BSC)	IRQ3 输入 (INTC)	TxD2 输出 (SCIF)	DACK1 输出 (DMAC)	—
	PE6 输入 / 输出 (端口)	A25 输出 (地址)	IRQ2 输入 (INTC)	RxD2 输入 (SCIF)	DREQ1 输入 (DMAC)	—
	PE5 输入 / 输出 (端口)	A24 输出 (地址)	IRQ1 输入 (INTC)	TxD1 输出 (SCIF)	DACK0 输出 (DMAC)	—
	PE4 输入 / 输出 (端口)	A23 输出 (地址)	IRQ0 输入 (INTC)	RxD1 输入 (SCIF)	DREQ0 输入 (DMAC)	—
PECRL1	PE3 输入 / 输出 (端口)	A22 输出 (地址)	—	SCK1 输入 / 输出 (SCIF)	—	—
	PE2 输入 / 输出 (端口)	A21 输出 (地址)	—	SCK0 输入 / 输出 (SCIF)	—	—
	PE1 输入 / 输出 (端口)	CS4 输出 (BSC)	MRES 输入 (系统控制)	TxD0 输出 (SCIF)	—	—
	PE0 输入 / 输出 (端口)	BS 输出 (BSC)	—	RxD0 输入 (SCIF)	ADTRG 输入 (ADC)	—

表 25.6 多路复用一览表 (端口 F)

设定 寄存器	模式位 (PFnMD[1:0]) 设定值			
	00	01	10	11
	功能 1 (相关模块)	功能 2 (相关模块)	功能 3 (相关模块)	功能 4 (相关模块)
PFCRH4	PF30 输入 / 输出 (端口)	AUDIO_CLK 输入 (SSI)	—	—
	PF29 输入 / 输出 (端口)	SSIDATA3 输入 / 输出 (SSI)	—	—
	PF28 输入 / 输出 (端口)	SSIWS3 输入 / 输出 (SSI)	—	—
PFCRH3	PF27 输入 / 输出 (端口)	SSISCK3 输入 / 输出 (SSI)	—	—
	PF26 输入 / 输出 (端口)	SSIDATA2 输入 / 输出 (SSI)	—	—
	PF25 输入 / 输出 (端口)	SSIWS2 输入 / 输出 (SSI)	—	—
	PF24 输入 / 输出 (端口)	SSISCK2 输入 / 输出 (SSI)	—	—
PFCRH2	PF23 输入 / 输出 (端口)	SSIDATA1 输入 / 输出 (SSI)	LCD_VEPWC 输出 (LCDC)	—
	PF22 输入 / 输出 (端口)	SSIWS1 输入 / 输出 (SSI)	LCD_VCPWC 输出 (LCDC)	—
	PF21 输入 / 输出 (端口)	SSISCK1 输入 / 输出 (SSI)	LCD_CLK 输入 (LCDC)	—
	PF20 输入 / 输出 (端口)	SSIDATA0 输入 / 输出 (SSI)	LCD_FLM 输出 (LCDC)	—
PFCRH1	PF19 输入 / 输出 (端口)	SSIWS0 输入 / 输出 (SSI)	LCD_M_DISP 输出 (LCDC)	—
	PF18 输入 / 输出 (端口)	SSISCK0 输入 / 输出 (SSI)	LCD_CL2 输出 (LCDC)	—
	PF17 输入 / 输出 (端口)	$\overline{\text{FCE}}$ 输出 (FLCTL)	LCD_CL1 输出 (LCDC)	—
	PF16 输入 / 输出 (端口)	FRB 输入 (FLCTL)	LCD_DON 输出 (LCDC)	—
PFCRL4	PF15 输入 / 输出 (端口)	NAF7 输入 / 输出 (FLCTL)	LCD_DATA15 输出 (LCDC)	—
	PF14 输入 / 输出 (端口)	NAF6 输入 / 输出 (FLCTL)	LCD_DATA14 输出 (LCDC)	—
	PF13 输入 / 输出 (端口)	NAF5 输入 / 输出 (FLCTL)	LCD_DATA13 输出 (LCDC)	—
	PF12 输入 / 输出 (端口)	NAF4 输入 / 输出 (FLCTL)	LCD_DATA12 输出 (LCDC)	—
PFCRL3	PF11 输入 / 输出 (端口)	NAF3 输入 / 输出 (FLCTL)	LCD_DATA11 输出 (LCDC)	—
	PF10 输入 / 输出 (端口)	NAF2 输入 / 输出 (FLCTL)	LCD_DATA10 输出 (LCDC)	—
	PF9 输入 / 输出 (端口)	NAF1 输入 / 输出 (FLCTL)	LCD_DATA9 输出 (LCDC)	—
	PF8 输入 / 输出 (端口)	NAF0 输入 / 输出 (FLCTL)	LCD_DATA8 输出 (LCDC)	—
PFCRL2	PF7 输入 / 输出 (端口)	FSC 输出 (FLCTL)	LCD_DATA7 输出 (LCDC)	$\overline{\text{SCS1}}$ 输入 / 输出 (SSU)
	PF6 输入 / 输出 (端口)	FOE 输出 (FLCTL)	LCD_DATA6 输出 (LCDC)	SSO1 输入 / 输出 (SSU)
	PF5 输入 / 输出 (端口)	FCDE 输出 (FLCTL)	LCD_DATA5 输出 (LCDC)	SSI1 输入 / 输出 (SSU)
	PF4 输入 / 输出 (端口)	$\overline{\text{FWE}}$ 输出 (FLCTL)	LCD_DATA4 输出 (LCDC)	SSCK1 输入 / 输出 (SSU)
PFCRL1	PF3 输入 / 输出 (端口)	TCLKD 输入 (MTU2)	LCD_DATA3 输出 (LCDC)	$\overline{\text{SCS0}}$ 输入 / 输出 (SSU)
	PF2 输入 / 输出 (端口)	TCLKC 输入 (MTU2)	LCD_DATA2 输出 (LCDC)	SSO0 输入 / 输出 (SSU)
	PF1 输入 / 输出 (端口)	TCLKB 输入 (MTU2)	LCD_DATA1 输出 (LCDC)	SSI0 输入 / 输出 (SSU)
	PF0 输入 / 输出 (端口)	TCLKA 输入 (MTU2)	LCD_DATA0 输出 (LCDC)	SSCK0 输入 / 输出 (SSU)

25.1 特点

- 通过设定控制寄存器，选择多路复用引脚的功能
- 选择通用输入/输出功能或 MTU2 的 TIOC 输入/输出功能时，通过设定 IO 寄存器选择输入/输出方向
- 端口 A 通过设定 A/D 转换器的 A/D 控制/状态寄存器 (ADCSR)、D/A 转换器的 D/A 控制寄存器 (DACR) 来转换功能

25.2 寄存器的说明

PFC 有以下寄存器。

表 25.7 寄存器结构

寄存器名称	简称	R/W	初始值	地址	存取长度
端口 B/IO 寄存器 L	PBIORL	R/W	H'0000	H'FFFE3886	8、16
端口 B 控制寄存器 L4	PBCRL4	R/W	H'0001	H'FFFE3890	8*1、16、32
端口 B 控制寄存器 L3	PBCRL3	R/W	H'0000	H'FFFE3892	8、16
端口 B 控制寄存器 L2	PBCRL2	R/W	H'0000	H'FFFE3894	8、16、32
端口 B 控制寄存器 L1	PBCRL1	R/W	H'0000	H'FFFE3896	8、16
IRQOUT 功能控制寄存器	IFCR	R/W	H'0000	H'FFFE38A2	8、16
端口 C/IO 寄存器 L	PCIORL	R/W	H'0000	H'FFFE3906	8、16
端口 C 控制寄存器 L4	PCCRL4	R/W	H'0000	H'FFFE3910	8、16、32
端口 C 控制寄存器 L3	PCCRL3	R/W	H'0000	H'FFFE3912	8、16
端口 C 控制寄存器 L2	PCCRL2	R/W	H'0000	H'FFFE3914	8、16、32
端口 C 控制寄存器 L1	PCCRL1	R/W	H'0000/ H'0010*2	H'FFFE3916	8、16
端口 D/IO 寄存器 L	PDIORL	R/W	H'0000	H'FFFE3986	8、16
端口 D 控制寄存器 L4	PDCRL4	R/W	H'0000/ H'1111*2	H'FFFE3990	8、16、32
端口 D 控制寄存器 L3	PDCRL3	R/W	H'0000/ H'1111*2	H'FFFE3992	8、16
端口 D 控制寄存器 L2	PDCRL2	R/W	H'0000/ H'1111*2	H'FFFE3994	8、16、32
端口 D 控制寄存器 L1	PDCRL1	R/W	H'0000/ H'1111*2	H'FFFE3996	8、16
端口 E/IO 寄存器 L	PEIORL	R/W	H'0000	H'FFFE3A06	8、16
端口 E 控制寄存器 L4	PECRL4	R/W	H'0000	H'FFFE3A10	8、16、32
端口 E 控制寄存器 L3	PECRL3	R/W	H'0000	H'FFFE3A12	8、16
端口 E 控制寄存器 L2	PECRL2	R/W	H'0000	H'FFFE3A14	8、16、32
端口 E 控制寄存器 L1	PECRL1	R/W	H'0000	H'FFFE3A16	8、16
端口 F/IO 寄存器 H	PFIORH	R/W	H'0000	H'FFFE3A84	8、16、32
端口 F/IO 寄存器 L	PFIORL	R/W	H'0000	H'FFFE3A86	8、16
端口 F 控制寄存器 H4	PFCRH4	R/W	H'0000	H'FFFE3A88	8、16、32
端口 F 控制寄存器 H3	PFCRH3	R/W	H'0000	H'FFFE3A8A	8、16
端口 F 控制寄存器 H2	PFCRH2	R/W	H'0000	H'FFFE3A8C	8、16、32

寄存器名称	简称	R/W	初始值	地址	存取长度
端口 F 控制寄存器 H1	PFCRH1	R/W	H'0000	H'FFFE3A8E	8、16
端口 F 控制寄存器 L4	PFCRL4	R/W	H'0000	H'FFFE3A90	8、16、32
端口 F 控制寄存器 L3	PFCRL3	R/W	H'0000	H'FFFE3A92	8、16
端口 F 控制寄存器 L2	PFCRL2	R/W	H'0000	H'FFFE3A94	8、16、32
端口 F 控制寄存器 L1	PFCRL1	R/W	H'0000	H'FFFE3A96	8、16
SSI 时钟选择寄存器	SCSR	R/W	H'0000	H'FFFE3AA2	8、16

【注】 *1 8 位存取时，可读取但不可写入。

*2 初始值因 LSI 的运行模式而不同。

25.2.1 端口 B/I/O 寄存器 L (PBIORL)

PBIORL 为 16 位可读取 / 写入的寄存器，选择端口 B 引脚的输入 / 输出方向。PB11IOR ~ PB8IOR 位分别对应 PB11/CTx1 ~ PB8/CRx0/CRx0/CRx1 引脚。端口 B 的引脚功能为通用输入 / 输出 (PB11 ~ PB8) 时，PBIORL 有效；为其他功能时无效。如果将 PBIORL 位设置为 1，则对应的引脚为输出；设置为 0 时为输入。

PBIORL 的 bit15 ~ 12、7 ~ 0 为保留位。读取值、写入值总是为 0。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	PB11 IOR	PB10 IOR	PB9 IOR	PB8 IOR	-	-	-	-	-	-	-	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

25.2.2 端口 B 控制寄存器 L1 ~ L4 (PBCRL1 ~ PBCRL4)

PBCRL1 ~ PBCRL4 为 16 位可读取 / 写入的寄存器，选择端口 B 多路复用引脚的功能。

(1) 端口 B 控制寄存器 L4 (PBCRL4)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	PB12MD[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

【注】 写入 PBCRL4 时，必须将 bit15 ~ 8 的值设定为 H'5A，以 16/32 位存取写入。
禁止以 8 位存取写入。

位	位名称	初始值	R/W	说 明
15 ~ 2	—	均为 0	R	保留位 读取值总是为 0。
1、0	PB12MD[1:0]	01	R/W	PB12 模式 控制 PB12/WDTOVF/IRQOUT/REFOUT/UBCTRG 引脚的功能。 00: PB12 输出 (端口) 01: WDTOVF 输出 (WDT) 10: IRQOUT/REFOUT 输出 (INTC/BSC) 11: UBCTRG 输出 (UBC)

(2) 端口 B 控制寄存器 L3 (PBCRL3)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	PB11MD0	-	-	-	PB10MD0	-	-	PB9MD[1:0]	-	-	-	PB8MD[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R/W	R/W	R	R	R/W	R/W

位	位名称	初始值	R/W	说 明
15 ~ 13	—	均为 0	R	保留位 读取值、写入值总是为 0。
12	PB11MD0	0	R/W	PB11 模式 控制 PB11/CTx1 引脚的功能。 0: PB11 输入 / 输出 (端口) 1: CTx1 输出 (RCAN-TL1)
11 ~ 9	—	均为 0	R	保留位 读取值、写入值总是为 0。
8	PB10MD0	0	R/W	PB10 模式 控制 PB10/CRx1 引脚的功能。 0: PB10 输入 / 输出 (端口) 1: CRx1 输入 (RCAN-TL1)
7、6	—	均为 0	R	保留位 读取值、写入值总是为 0。
5、4	PB9MD[1:0]	00	R/W	PB9 模式 控制 PB9/CTx0/CTx0&CTx1 引脚的功能。 00: PB9 输入 / 输出 (端口) 01: CTx0 输出 (RCAN-TL1) 10: CTx0&CTx1 输出 (RCAN-TL1) 11: 禁止设定
3、2	—	均为 0	R	保留位 读取值、写入值总是为 0。
1、0	PB8MD[1:0]	00	R/W	PB8 模式 控制 PB8/CRx0/CRx0/CRx1 引脚的功能。 00: PB8 输入 / 输出 (端口) 01: CRx0 输入 (RCAN-TL1) 10: CRx0/CRx1 输入 (RCAN-TL1) 11: 禁止设定

(3) 端口 B 控制寄存器 L2 (PBCRL2)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	PB7MD[1:0]	-	-	PB6MD[1:0]	-	-	PB5MD[1:0]	-	-	PB4MD[1:0]	-	-	-	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

位	位名称	初始值	R/W	说 明
15、14	—	均为 0	R	保留位 读取值、写入值总是为 0。
13、12	PB7MD[1:0]	00	R/W	PB7 模式 控制 PB7/SDA3/PINT7/IRQ7 引脚的功能。 00: PB7 输入 (端口) 01: SDA3 输入 / 输出 (IIC3) 10: PINT7 输入 (INTC) 11: IRQ7 输入 (INTC)
11、10	—	均为 0	R	保留位 读取值、写入值总是为 0。
9、8	PB6MD[1:0]	00	R/W	PB6 模式 控制 PB6/SCL3/PINT6/IRQ6 引脚的功能。 00: PB6 输入 (端口) 01: SCL3 输入 / 输出 (IIC3) 10: PINT6 输入 (INTC) 11: IRQ6 输入 (INTC)
7、6	—	均为 0	R	保留位 读取值、写入值总是为 0。
5、4	PB5MD[1:0]	00	R/W	PB5 模式 控制 PB5/SDA2/PINT5/IRQ5 引脚的功能。 00: PB5 输入 (端口) 01: SDA2 输入 / 输出 (IIC3) 10: PINT5 输入 (INTC) 11: IRQ5 输入 (INTC)
3、2	—	0	R	保留位 读取值、写入值总是为 0。
1、0	PB4MD[1:0]	00	R/W	PB4 模式 控制 PB4/SCL2/PINT4/IRQ4 引脚的功能。 00: PB4 输入 (端口) 01: SCL2 输入 / 输出 (IIC3) 10: PINT4 输入 (INTC) 11: IRQ4 输入 (INTC)

(4) 端口 B 控制寄存器 L1 (PBCRL1)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	PB3MD[1:0]	-	-	PB2MD[1:0]	-	-	PB1MD[1:0]	-	-	PB0MD[1:0]	-	-	-	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

位	位名称	初始值	R/W	说 明
15、14	—	均为 0	R	保留位 读取值、写入值总是为 0。
13、12	PB3MD[1:0]	00	R/W	PB3 模式 控制 PB3/SDA1/PINT3/IRQ3 引脚的功能。 00: PB3 输入 (端口) 01: SDA1 输入 / 输出 (IIC3) 10: PINT3 输入 (INTC) 11: IRQ3 输入 (INTC)
11、10	—	均为 0	R	保留位 读取值、写入值总是为 0。
9、8	PB2MD[1:0]	00	R/W	PB2 模式 控制 PB2/SCL1/PINT2/IRQ2 引脚的功能。 00: PB2 输入 (端口) 01: SCL1 输入 / 输出 (IIC3) 10: PINT2 输入 (INTC) 11: IRQ2 输入 (INTC)
7、6	—	均为 0	R	保留位 读取值、写入值总是为 0。
5、4	PB1MD[1:0]	00	R/W	PB1 模式 控制 PB1/SDA0/PINT1/IRQ1 引脚的功能。 00: PB1 输入 (端口) 01: SDA0 输入 / 输出 (IIC3) 10: PINT1 输入 (INTC) 11: IRQ1 输入 (INTC)
3、2	—	0	R	保留位 读取值、写入值总是为 0。
1、0	PB0MD[1:0]	00	R/W	PB0 模式 控制 PB0/SCL0/PINT0/IRQ0 引脚的功能 00: PB0 输入 (端口) 01: SCL0 输入 / 输出 (IIC3) 10: PINT0 输入 (INTC) 11: IRQ0 输入 (INTC)

25.2.3 端口 C/I/O 寄存器 L (PCIORL)

PCIORL 为 16 位可读取 / 写入的寄存器，选择端口 C 引脚的输入 / 输出方向。PC14IOR ~ PC0IOR 位分别对应 PC14/WAIT ~ PC0/A0/CS7 引脚。端口 C 的引脚功能为通用输入 / 输出 (PC14 ~ PC0) 时，PCIORL 有效；为其他功能时无效。如果将 PCIORL 的位设置为 1，则对应的引脚为输出；设置为 0 时为输入。

PCIORL 的 bit15 为保留位。读取值、写入值总是为 0。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PC14 IOR	PC13 IOR	PC12 IOR	PC11 IOR	PC10 IOR	PC9 IOR	PC8 IOR	PC7 IOR	PC6 IOR	PC5 IOR	PC4 IOR	PC3 IOR	PC2 IOR	PC1 IOR	PC0 IOR
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

25.2.4 端口 C 控制寄存器 L1 ~ L4 (PCCRL1 ~ PCCRL4)

PCCRL1 ~ PCCRL4 均为 16 位可读取 / 写入的寄存器，选择端口 C 多路复用引脚的功能。

(1) 端口 C 控制寄存器 L4 (PCCRL4)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	PC14 MD0	-	-	-	PC13 MD0	-	-	-	PC12 MD0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

位	位名称	初始值	R/W	说 明
15 ~ 9	—	均为 0	R	保留位 读取值、写入值总是为 0。
8	PC14MD0	0	R/W	PC14 模式 控制 PC14/WAIT 引脚的功能。 0: PC14 输入 / 输出 (端口) 1: WAIT 输入 (BSC)
7 ~ 5	—	均为 0	R	保留位 读取值、写入值总是为 0。
4	PC13MD0	0	R/W	PC13 模式 控制 PC13/RDWR 引脚的功能。 0: PC13 输入 / 输出 (端口) 1: RDWR 输出 (BSC)
3 ~ 1	—	均为 0	R	保留位 读取值、写入值总是为 0。
0	PC12MD0	0	R/W	PC12 模式 控制 PC12/CKE 引脚的功能。 0: PC12 输入 / 输出 (端口) 1: CKE 输出 (BSC)

(2) 端口 C 控制寄存器 L3 (PCCRL3)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	PC11MD[1:0]	-	-	PC10MD[1:0]	-	-	-	PC9MD0	-	-	-	PC8MD0		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R	R/W	R	R	R	R/W

位	位名称	初始值	R/W	说 明
15、14	—	均为 0	R	保留位 读取值、写入值总是为 0。
13、12	PC11MD[1:0]	00	R/W	PC11 模式 控制 PC11/CASU/BREQ 引脚的功能。 00: PC11 输入 / 输出 (端口) 01: CASU 输出 (BSC) 10: BREQ 输入 (BSC) 11: 禁止设定
11、10	—	均为 0	R	保留位 读取值、写入值总是为 0。
9、8	PC10MD[1:0]	00	R/W	PC10 模式 控制 PC10/RASU/BACK 引脚的功能。 00: PC10 输入 / 输出 (端口) 01: RASU 输出 (BSC) 10: BACK 输出 (BSC) 11: 禁止设定
7 ~ 5	—	均为 0	R	保留位 读取值、写入值总是为 0。
4	PC9MD0	0	R/W	PC9 模式 控制 PC9/CASL 引脚的功能。 0: PC9 输入 / 输出 (端口) 1: CASL 输出 (BSC)
3 ~ 1	—	均为 0	R	保留位 读取值、写入值总是为 0。
0	PC8MD0	0	R/W	PC8 模式 控制 PC8/RASL 引脚的功能。 0: PC8 输入 / 输出 (端口) 1: RASL 输出 (BSC)

(3) 端口 C 控制寄存器 L2 (PCCRL2)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	PC7 MD0	-	-	-	PC6 MD0	-	-	-	PC5 MD0	-	-	-	PC4 MD0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

位	位名称	初始值	R/W	说 明
15 ~ 13	—	均为 0	R	保留位 读取值、写入值总是为 0。
12	PC7MD0	0	R/W	PC7 模式 控制 $\overline{\text{PC7/WE3/DQMUU/AH/ICIOWR}}$ 引脚的功能。 0: $\overline{\text{PC7}}$ 输入 / 输出 (端口) 1: $\overline{\text{WE3/DQMUU/AH/ICIOWR}}$ 输出 (BSC)
11 ~ 9	—	均为 0	R	保留位 读取值、写入值总是为 0。
8	PC6MD0	0	R/W	PC6 模式 控制 $\overline{\text{PC6/WE2/DQMUL/ICIORD}}$ 引脚的功能。 0: $\overline{\text{PC6}}$ 输入 / 输出 (端口) 1: $\overline{\text{WE2/DQMUL/ICIORD}}$ 输出 (BSC)
7 ~ 5	—	均为 0	R	保留位 读取值、写入值总是为 0。
4	PC5MD0	0	R/W	PC5 模式 控制 $\overline{\text{PC5/WE1/DQMLU/WE}}$ 引脚的功能。 0: $\overline{\text{PC5}}$ 输入 / 输出 (端口) 1: $\overline{\text{WE1/DQMLU/WE}}$ 输出 (BSC)
3 ~ 1	—	均为 0	R	保留位 读取值、写入值总是为 0。
0	PC4MD0	0	R/W	PC4 模式 控制 $\overline{\text{PC4/WE0/DQMLL}}$ 引脚的功能。 0: $\overline{\text{PC4}}$ 输入 / 输出 (端口) 1: $\overline{\text{WE0/DQMLL}}$ 输出 (BSC)

(4) 端口 C 控制寄存器 L1 (PCCRL1)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	PC3 MD0	-	-	-	PC2 MD0	-	-	-	PC1 MD0	-	-	PC0MD[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0/1*	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R/W	R/W

【注】* 初始值因 LSI 的运行模式而不同。

位	位名称	初始值	R/W	说 明
15 ~ 13	—	均为 0	R	保留位 读取值、写入值总是为 0。
12	PC3MD0	0	R/W	PC3 模式 控制 PC3/ $\overline{\text{CS3}}$ 引脚的功能。 0: PC3 输入 / 输出 (端口) 1: $\overline{\text{CS3}}$ 输出 (BSC)
11 ~ 9	—	均为 0	R	保留位 读取值、写入值总是为 0。
8	PC2MD0	0	R/W	PC2 模式 控制 PC2/ $\overline{\text{CS2}}$ 引脚的功能。 0: PC2 输入 / 输出 (端口) 1: $\overline{\text{CS2}}$ 输出 (BSC)
7 ~ 5	—	均为 0	R	保留位 读取值、写入值总是为 0。
4	PC1MD0	0/1*	R/W	PC1 模式 控制 PC1/A1 引脚的功能。 • 区域 0 : 32 位模式 0: PC1 输入 / 输出 (端口) (初始值) 1: A1 输出 (地址) • 区域 0 : 16 位模式 0: 禁止设定 1: A1 输出 (地址) (初始值)
3、2	—	均为 0	R	保留位 读取值、写入值总是为 0。
1、0	PC0MD[1:0]	00	R/W	PC0 模式 控制 PC0/A0/ $\overline{\text{CS7}}$ 引脚的功能。 00: PC0 输入 / 输出 (端口) 01: A0 输出 (地址) 10: $\overline{\text{CS7}}$ 输出 (BSC) 11: 禁止设定

【注】* 初始值因 LSI 的运行模式而不同。

25.2.5 端口 D/I/O 寄存器 L (PDIORL)

PDIORL 为 16 位可读取 / 写入的寄存器，选择端口 D 引脚的输入 / 输出方向。PD15IOR ~ PD0IOR 位分别对应 PD15/D31/PINT7/ADTRG/TIOC4D ~ PD0/D16/IRQ0/SSCK0/DREQ2/TIOC0A 引脚。端口 D 的引脚功能为通用输入 / 输出 (PD15 ~ PD0) 或 MTU2 的 TIOC 输入 / 输出时，PDIORL 有效；为其他功能时无效。如果将 PDIORL 位设置为 1，则对应的引脚为输出；设置为 0 时为输入。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PD15 IOR	PD14 IOR	PD13 IOR	PD12 IOR	PD11 IOR	PD10 IOR	PD9 IOR	PD8 IOR	PD7 IOR	PD6 IOR	PD5 IOR	PD4 IOR	PD3 IOR	PD2 IOR	PD1 IOR	PD0 IOR
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

25.2.6 端口 D 控制寄存器 L1 ~ L4 (PDCRL1 ~ PDCRL4)

PDCRL1 ~ PDCRL4 均为 16 位可读取 / 写入的寄存器，选择端口 D 多路复用引脚的功能。

(1) 端口 D 控制寄存器 L4 (PDCRL4)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PD15MD[2:0]		-	PD14MD[2:0]		-	PD13MD[2:0]		-	PD12MD[2:0]		-	PD11MD[2:0]		-
初始值:	0	0	0	0/1*	0	0	0	0/1*	0	0	0	0/1*	0	0	0	0/1*
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

【注】* 初始值因 LSI 的运行模式而不同。

位	位名称	初始值	R/W	说 明
15	—	0	R	保留位 读取值、写入值总是为 0。
14 ~ 12	PD15MD[2:0]	000/ 001*	R/W	PD15 模式 控制 PD15/D31/PINT7/ADTRG/TIOC4D 引脚的功能。 - 区域 0 : 32 位模式 000: 禁止设定100: 禁止设定 001: D31 输入 / 输出（数据）（初始值）101: 禁止设定 010: 禁止设定110: 禁止设定 011: 禁止设定111: 禁止设定 - 区域 0 : 16 位模式 000: PD15 输入 / 输出（端口）（初始值）100: ADTRG 输入（ADC） 001: D31 输入 / 输出（数据）101: TIOC4D 输入 / 输出（MTU2） 010: PINT7 输入（INTC）110: 禁止设定 011: 禁止设定111: 禁止设定
11	—	0	R	保留位 读取值、写入值总是为 0。

位	位名称	初始值	R/W	说 明
10 ~ 8	PD14MD[2:0]	000/ 001*	R/W	PD14 模式 控制 PD14/D30/PINT6/TIOC4C 引脚的功能。 • 区域 0 : 32 位模式 000: 禁止设定 001: D30 输入 / 输出 (数据) (初始值) 010: 禁止设定 011: 禁止设定 • 区域 0 : 16 位模式 000: PD14 输入 / 输出 (端口) (初始值) 001: D30 输入 / 输出 (数据) 010: PINT6 输入 (INTC) 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: 禁止设定 111: 禁止设定 100: 禁止设定 101: TIOC4C 输入 / 输出 (MTU2) 110: 禁止设定 111: 禁止设定
7	—	0	R	保留位 读取值、写入值总是为 0。
6 ~ 4	PD13MD[2:0]	000/ 001*	R/W	PD13 模式 控制 PD13/D29/PINT5/TEND1/TIOC4B 引脚的功能。 • 区域 0 : 32 位模式 000: 禁止设定 001: D29 输入 / 输出 (数据) (初始值) 010: 禁止设定 011: 禁止设定 • 区域 0 : 16 位模式 000: PD13 输入 / 输出 (端口) (初始值) 001: D29 输入 / 输出 (数据) 010: PINT5 输入 (INTC) 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: 禁止设定 111: 禁止设定 100: TEND1 输出 (DMAC) 101: TIOC4B 输入 / 输出 (MTU2) 110: 禁止设定 111: 禁止设定
3	—	0	R	保留位 读取值、写入值总是为 0。
2 ~ 0	PD12MD[2:0]	000/ 001*	R/W	PD12 模式 控制 PD12/D28/PINT4/DACK1/TIOC4A 引脚的功能。 • 区域 0 : 32 位模式 000: 禁止设定 001: D28 输入 / 输出 (数据) (初始值) 010: 禁止设定 011: 禁止设定 • 区域 0 : 16 位模式 000: PD12 输入 / 输出 (端口) (初始值) 001: D28 输入 / 输出 (数据) 010: PINT4 输入 (INTC) 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: 禁止设定 111: 禁止设定 100: DACK1 输出 (DMAC) 101: TIOC4A 输入 / 输出 (MTU2) 110: 禁止设定 111: 禁止设定

【注】 * 初始值因 LSI 的运行模式而不同。

(2) 端口 D 控制寄存器 L3 (PDCRL3)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PD11MD[2:0]			-	PD10MD[2:0]			-	PD9MD[2:0]			-	PD8MD[2:0]		
初始值:	0	0	0	0/1*	0	0	0	0/1*	0	0	0	0/1*	0	0	0	0/1*
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

【注】* 初始值因LSI的运行模式而不同。

位	位名称	初始值	R/W	说 明
15	—	0	R	保留位 读取值、写入值总是为 0。
14 ~ 12	PD11MD[2:0]	000/ 001*	R/W	PD11 模式 控制 PD11/D27/PINT3/DREQ1/TIOC3D 引脚的功能。 - 区域 0 : 32 位模式 000: 禁止设定 001: D27 输入 / 输出 (数据) (初始值) 010: 禁止设定 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: 禁止设定 111: 禁止设定 - 区域 0 : 16 位模式 000: PD11 输入 / 输出 (端口) (初始值) 001: D27 输入 / 输出 (数据) 010: PINT3 输入 (INTC) 011: 禁止设定 100: DREQ1 输入 (DMAC) 101: TIOC3D 输入 / 输出 (MTU2) 110: 禁止设定 111: 禁止设定
11	—	0	R	保留位 读取值、写入值总是为 0。
10 ~ 8	PD10MD[2:0]	000/ 001*	R/W	PD10 模式 控制 PD10/D26/PINT2/TEND0/TIOC3C 引脚的功能。 - 区域 0 : 32 位模式 000: 禁止设定 001: D26 输入 / 输出 (数据) (初始值) 010: 禁止设定 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: 禁止设定 111: 禁止设定 - 区域 0 : 16 位模式 000: PD10 输入 / 输出 (端口) (初始值) 001: D26 输入 / 输出 (数据) 010: PINT2 输入 (INTC) 011: 禁止设定 100: TEND0 输出 (DMAC) 101: TIOC3C 输入 / 输出 (MTU2) 110: 禁止设定 111: 禁止设定
7	—	0	R	保留位 读取值、写入值总是为 0。

位	位名称	初始值	R/W	说 明
6 ~ 4	PD9MD[2:0]	000/ 001*	R/W	PD9 模式 控制 PD9/D25/PINT1/DACK0/TIOC3B 引脚的功能。 • 区域 0 : 32 位模式 000: 禁止设定100: 禁止设定 001: D25 输入 / 输出 (数据) (初始值)101: 禁止设定 010: 禁止设定110: 禁止设定 011: 禁止设定111: 禁止设定 • 区域 0 : 16 位模式 000: PD9 输入 / 输出 (端口) (初始值)100: DACK0 输出 (DMAC) 001: D25 输入 / 输出 (数据)101: TIOC3B 输入 / 输出 (MTU2) 010: PINT1 输入 (INTC)110: 禁止设定 011: 禁止设定111: 禁止设定
3	—	0	R	保留位 读取值、写入值总是为 0。
2 ~ 0	PD8MD[2:0]	000/ 001*	R/W	PD8 模式 控制 PD8/D24/PINT0/DREQ0/TIOC3A 引脚的功能。 • 区域 0 : 32 位模式 000: 禁止设定100: 禁止设定 001: D24 输入 / 输出 (数据) (初始值)101: 禁止设定 010: 禁止设定110: 禁止设定 011: 禁止设定111: 禁止设定 • 区域 0 : 16 位模式 000: PD8 输入 / 输出 (端口) (初始值)100: DREQ0 输入 (DMAC) 001: D24 输入 / 输出 (数据)101: TIOC3A 输入 / 输出 (MTU2) 010: PINT0 输入 (INTC)110: 禁止设定 011: 禁止设定111: 禁止设定

【注】 * 初始值因 LSI 的运行模式而不同。

(3) 端口 D 控制寄存器 L2 (PDCRL2)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PD7MD[2:0]			-	PD6MD[2:0]			-	PD5MD[2:0]			-	PD4MD[2:0]		
初始值:	0	0	0	0/1*	0	0	0	0/1*	0	0	0	0/1*	0	0	0	0/1*
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

【注】* 初始值因LSI的运行模式而不同。

位	位名称	初始值	R/W	说 明
15	—	0	R	保留位 读取值、写入值总是为 0。
14 ~ 12	PD7MD[2:0]	000/ 001*	R/W	PD7 模式 控制 PD7/D23/IRQ7/ <u>SCS1</u> /TCLKD/TIOC2B 引脚的功能。 • 区域 0 : 32 位模式 000: 禁止设定 001: D23 输入 / 输出 (数据) (初始值) 010: 禁止设定 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: 禁止设定 111: 禁止设定 • 区域 0 : 16 位模式 000: PD7 输入 / 输出 (端口) (初始值) 001: D23 输入 / 输出 (数据) 010: IRQ7 输入 (INTC) 011: <u>SCS1</u> 输入 / 输出 (SSU) 100: TCLKD 输入 (MTU2) 101: TIOC2B 输入 / 输出 (MTU2) 110: 禁止设定 111: 禁止设定
11	—	0	R	保留位 读取值、写入值总是为 0。
10 ~ 8	PD6MD[2:0]	000/ 001*	R/W	PD6 模式 选择 PD6/D22/IRQ6/SSO1/TCLKC/TIOC2A 引脚的功能。 • 区域 0 : 32 位模式 000: 禁止设定 001: D22 输入 / 输出 (数据) (初始值) 010: 禁止设定 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: 禁止设定 111: 禁止设定 • 区域 0 : 16 位模式 000: PD6 输入 / 输出 (端口) (初始值) 001: D22 输入 / 输出 (数据) 010: IRQ6 输入 (INTC) 011: SSO1 输入 / 输出 (SSU) 100: TCLKC 输入 (MTU2) 101: TIOC2A 输入 / 输出 (MTU2) 110: 禁止设定 111: 禁止设定
7	—	0	R	保留位 读取值、写入值总是为 0。

位	位名称	初始值	R/W	说 明
6 ~ 4	PD5MD[2:0]	000/ 001*	R/W	PD5 模式 选择 PD5/D21/IRQ5/SSI1/TCLKB/TIOC1B 引脚的功能。 • 区域 0 : 32 位模式 000: 禁止设定 001: D21 输入 / 输出 (数据) (初始值) 010: 禁止设定 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: 禁止设定 111: 禁止设定 • 区域 0 : 16 位模式 000: PD5 输入 / 输出 (端口) (初始值) 001: D21 输入 / 输出 (数据) 010: IRQ5 输入 (INTC) 011: SSI1 输入 / 输出 (SSU) 100: TCLKB 输入 (MTU2) 101: TIOC1B 输入 / 输出 (MTU2) 110: 禁止设定 111: 禁止设定
3	—	0	R	保留位 读取值、写入值总是为 0。
2 ~ 0	PD4MD[2:0]	000/ 001*	R/W	PD4 模式 选择 PD4/D20/IRQ4/SSCK1/TCLKA/TIOC1A 引脚的功能。 • 区域 0 : 32 位模式 000: 禁止设定 001: D20 输入 / 输出 (数据) (初始值) 010: 禁止设定 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: 禁止设定 111: 禁止设定 • 区域 0 : 16 位模式 000: PD4 输入 / 输出 (端口) (初始值) 001: D20 输入 / 输出 (数据) 010: IRQ4 输入 (INTC) 011: SSCK1 输入 / 输出 (SSU) 100: TCLKA 输入 (MTU2) 101: TIOC1A 输入 / 输出 (MTU2) 110: 禁止设定 111: 禁止设定

【注】 * 初始值因 LSI 的运行模式而不同。

(4) 端口 D 控制寄存器 L1 (PDCRL1)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PD3MD[2:0]			-	PD2MD[2:0]			-	PD1MD[2:0]			-	PD0MD[2:0]		
初始值:	0	0	0	0/1*	0	0	0	0/1*	0	0	0	0/1*	0	0	0	0/1*
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

【注】* 初始值因LSI的运行模式而不同。

位	位名称	初始值	R/W	说 明
15	—	0	R	保留位 读取值、写入值总是为 0。
14 ~ 12	PD3MD[2:0]	000/ 001*	R/W	PD3 模式 选择 PD3/D19/IRQ3/ $\overline{\text{SCS0}}$ /DACK3/TIOC0D 引脚的功能。 • 区域 0 : 32 位模式 000: 禁止设定 001: D19 输入 / 输出 (数据) (初始值) 010: 禁止设定 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: 禁止设定 111: 禁止设定 • 区域 0 : 16 位模式 000: PD3 输入 / 输出 (端口) (初始值) 001: D19 输入 / 输出 (数据) 010: IRQ3 输入 (INTC) 011: $\overline{\text{SCS0}}$ 输入 / 输出 (SSU) 100: DACK3 输出 (DMAC) 101: TIOC0D 输入 / 输出 (MTU2) 110: 禁止设定 111: 禁止设定
11	—	0	R	保留位 读取值、写入值总是为 0。
10 ~ 8	PD2MD[2:0]	000/ 001*	R/W	PD2 模式 选择 PD2/D18/IRQ2/SSO0/DREQ3/TIOC0C 引脚的功能。 • 区域 0 : 32 位模式 000: 禁止设定 001: D18 输入 / 输出 (数据) (初始值) 010: 禁止设定 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: 禁止设定 111: 禁止设定 • 区域 0 : 16 位模式 000: PD2 输入 / 输出 (端口) (初始值) 001: D18 输入 / 输出 (数据) 010: IRQ2 输入 (INTC) 011: SSO0 输入 / 输出 (SSU) 100: DREQ3 输入 (DMAC) 101: TIOC0C 输入 / 输出 (MTU2) 110: 禁止设定 111: 禁止设定
7	—	0	R	保留位 读取值、写入值总是为 0。

位	位名称	初始值	R/W	说 明
6 ~ 4	PD1MD[2:0]	000/ 001*	R/W	PD1 模式 选择 PD1/D17/IRQ1/SSIO/DACK2/TIOC0B 引脚的功能。 • 区域 0 : 32 位模式 000: 禁止设定 001: D17 输入 / 输出 (数据) (初始值) 010: 禁止设定 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: 禁止设定 111: 禁止设定 • 区域 0 : 16 位模式 000: PD1 输入 / 输出 (端口) (初始值) 001: D17 输入 / 输出 (数据) 010: IRQ1 输入 (INTC) 011: SSIO 输入 / 输出 (SSU) 100: DACK2 输出 (DMAC) 101: TIOC0B 输入 / 输出 (MTU2) 110: 禁止设定 111: 禁止设定
3	—	0	R	保留位 读取值、写入值总是为 0。
2 ~ 0	PD0MD[2:0]	000/ 001*	R/W	PD0 模式 选择 PD0/D16/IRQ0/SSCK0/DREQ2/TIOC0A 引脚的功能。 • 区域 0 : 32 位模式 000: 禁止设定 001: D16 输入 / 输出 (数据) (初始值) 010: 禁止设定 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: 禁止设定 111: 禁止设定 • 区域 0 : 16 位模式 000: PD0 输入 / 输出 (端口) (初始值) 001: D16 输入 / 输出 (数据) 010: IRQ0 输入 (INTC) 011: SSCK0 输入 / 输出 (SSU) 100: DREQ2 输入 (DMAC) 101: TIOC0A 输入 / 输出 (MTU2) 110: 禁止设定 111: 禁止设定

【注】 * 初始值因 LSI 的运行模式而不同。

25.2.7 端口 E/IO 寄存器 L (PEIORL)

PEIORL 为 16 位可读取 / 写入的寄存器, 选择端口 E 引脚的输入 / 输出方向。PE15IOR ~ PE0IOR 位分别对应 PE15/IOIS16/RTS3 ~ PE0/BS/RxD0/ADTRG 引脚。端口 E 的引脚功能为通用输入 / 输出 (PE15 ~ PE0) 时, PEIORL 有效; 为其他功能无效。如果将 PEIORL 位设置为 1, 则对应的引脚为输出; 设置为 0 时为输入。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PE15 IOR	PE14 IOR	PE13 IOR	PE12 IOR	PE11 IOR	PE10 IOR	PE9 IOR	PE8 IOR	PE7 IOR	PE6 IOR	PE5 IOR	PE4 IOR	PE3 IOR	PE2 IOR	PE1 IOR	PE0 IOR
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

25.2.8 端口 E 控制寄存器 L1 ~ L4 (PECRL1 ~ PECRL4)

PECRL1 ~ PECRL4 均为 16 位可读取 / 写入寄存器，选择端口 E 多路复用引脚的功能。

(1) 端口 E 控制寄存器 L4 (PECRL4)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	PE15MD[1:0]	-	-	PE14MD[1:0]	-	-	PE13MD[1:0]	-	-	PE12MD[1:0]	-	-	-	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

位	位名称	初始值	R/W	说 明
15、14	—	均为 0	R	保留位 读取值、写入值总是为 0。
13、12	PE15MD[1:0]	00	R/W	PE15 模式 控制 PE15/ $\overline{\text{IOIS16}}$ /RTS3 引脚的功能。 00: PE15 输入 / 输出 (端口) 01: $\overline{\text{IOIS16}}$ 输入 (BSC) 10: 禁止设定 11: RTS3 输入 / 输出 (SCIF)
11、10	—	均为 0	R	保留位 读取值、写入值总是为 0。
9、8	PE14MD[1:0]	00	R/W	PE14 模式 控制 PE14/ $\overline{\text{CS1}}$ / $\overline{\text{CTS3}}$ 引脚的功能。 00: PE14 输入 / 输出 (端口) 01: $\overline{\text{CS1}}$ 输出 (BSC) 10: 禁止设定 11: $\overline{\text{CTS3}}$ 输入 / 输出 (SCIF)
7、6	—	均为 0	R	保留位 读取值、写入值总是为 0。
5、4	PE13MD[1:0]	00	R/W	PE13 模式 控制 PE13/TxD3 引脚的功能。 00: PE13 输入 / 输出 (端口) 01: 禁止设定 10: 禁止设定 11: TxD3 输出 (SCIF)
3、2	—	均为 0	R	保留位 读取值、写入值总是为 0。
1、0	PE12MD[2:0]	00	R/W	PE12 模式 选择 PE12/RxD3 引脚的功能。 00: PE12 输入 / 输出 (端口) 01: 禁止设定 10: 禁止设定 11: RxD3 输入 (SCIF)

(2) 端口 E 控制寄存器 L3 (PECRL3)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PE11MD[2:0]			-	PE10MD[2:0]			-	-	PE9MD[1:0]		-	-	PE8MD[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

位	位名称	初始值	R/W	说 明
15	—	0	R	保留位 读取值、写入值总是为 0。
14 ~ 12	PE11MD[2:0]	000	R/W	PE11 模式 控制 PE11/CS6/CE1B/IRQ7/TEND1 引脚的功能。 000: PE11 输入 / 输出 (端口) 100: TEND1 输出 (DMAC) 001: CS6/CE1B 输出 (BSC) 101: 禁止设定 010: IRQ7 输入 (INTC) 110: 禁止设定 011: 禁止设定 111: 禁止设定
11	—	0	R	保留位 读取值、写入值总是为 0。
10 ~ 8	PE10MD[2:0]	000	R/W	PE10 模式 控制 PE10/CE2B/IRQ6/TEND0 引脚的功能。 000: PE10 输入 / 输出 (端口) 100: TEND0 输出 (DMAC) 001: CE2B 输出 (BSC) 101: 禁止设定 010: IRQ6 输入 (INTC) 110: 禁止设定 011: 禁止设定 111: 禁止设定
7、6	—	均为 0	R	保留位 读取值、写入值总是为 0。
5、4	PE9MD[1:0]	00	R/W	PE9 模式 控制 PE9/CS5/CE1A/IRQ5/SCK3 引脚的功能。 00: PE9 输入 / 输出 (端口) 01: CS5/CE1A 输出 (BSC) 10: IRQ5 输入 (INTC) 11: SCK3 输入 / 输出 (SCIF)
3、2	—	均为 0	R	保留位 读取值、写入值总是为 0。
1、0	PE8MD[1:0]	00	R/W	PE8 模式 控制 PE8/CE2A/IRQ4/SCK2 引脚的功能。 00: PE8 输入 / 输出 (端口) 01: CE2A 输出 (BSC) 10: IRQ4 输入 (INTC) 11: SCK2 输入 / 输出 (SCIF)

(3) 端口 E 控制寄存器 L2 (PECRL2)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PE7MD[2:0]			-	PE6MD[2:0]			-	PE5MD[2:0]			-	PE4MD[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15	—	0	R	保留位 读取值、写入值总是为 0。
14 ~ 12	PE7MD[2:0]	000	R/W	PE7 模式 控制 PE7/FRAME/IRQ3/TxD2/DACK1 引脚的功能。 000: PE7 输入 / 输出 (端口) 100: DACK1 输出 (DMAC) 001: FRAME 输出 (BSC) 101: 禁止设定 010: IRQ3 输入 (INTC) 110: 禁止设定 011: TxD2 输出 (SCIF) 111: 禁止设定
11	—	0	R	保留位 读取值、写入值总是为 0。
10 ~ 8	PE6MD[2:0]	000	R/W	PE6 模式 控制 PE6/A25/IRQ2/RxD2/DREQ1 引脚的功能。 000: PE6 输入 / 输出 (端口) 100: DREQ1 输入 (DMAC) 001: A25 输出 (地址) 101: 禁止设定 010: IRQ2 输入 (INTC) 110: 禁止设定 011: RxD2 输入 (SCIF) 111: 禁止设定
7	—	0	R	保留位 读取值、写入值总是为 0。
6 ~ 4	PE5MD[2:0]	000	R/W	PE5 模式 选择 PE5/A24/IRQ1/TxD1/DACK0 引脚的功能。 000: PE5 输入 / 输出 (端口) 100: DACK0 输出 (DMAC) 001: A24 输出 (地址) 101: 禁止设定 010: IRQ1 输入 (INTC) 110: 禁止设定 011: TxD1 输出 (SCIF) 111: 禁止设定
3	—	0	R	保留位 读取值、写入值总是为 0。
2 ~ 0	PE4MD[2:0]	000	R/W	PE4 模式 控制 PE4/A23/IRQ0/RxD1/DREQ0 引脚的功能。 000: PE4 输入 / 输出 (端口) 100: DREQ0 输入 (DMAC) 001: A23 输出 (地址) 101: 禁止设定 010: IRQ0 输入 (INTC) 110: 禁止设定 011: RxD1 输入 (SCIF) 111: 禁止设定

(4) 端口 E 控制寄存器 L1 (PECRL1)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	PE3MD[1:0]	-	-	PE2MD[1:0]	-	-	PE1MD[1:0]	-	PE0MD[2:0]					
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15、14	—	均为 0	R	保留位 读取值、写入值总是为 0。
13、12	PE3MD[1:0]	00	R/W	PE3 模式 控制 PE3/A22/SCK1 引脚的功能。 00: PE3 输入 / 输出 (端口) 01: A22 输出 (地址) 10: 禁止设定 11: SCK1 输入 / 输出 (SCIF)
11、10	—	均为 0	R	保留位 读取值、写入值总是为 0。
9、8	PE2MD[1:0]	00	R/W	PE2 模式 控制 PE2/A21/SCK0 引脚的功能。 00: PE2 输入 / 输出 (端口) 01: A21 输出 (地址) 10: 禁止设定 11: SCK0 输入 / 输出 (SCIF)
7、6	—	均为 0	R	保留位 读取值、写入值总是为 0。
5、4	PE1MD[1:0]	00	R/W	PE1 模式 控制 PE1/CS4/MRES/TxD0 引脚的功能。 00: PE1 输入 / 输出 (端口) 01: CS4 输出 (BSC) 10: MRES 输入 (系统控制) 11: TxD0 输出 (SCIF)
3	—	0	R	保留位 读取值、写入值总是为 0。
2 ~ 0	PE0MD[2:0]	000	R/W	PE0 模式 控制 PE0/BS/RxD0/ADTRG 引脚的功能。 000: PE0 输入 / 输出 (端口) 100: ADTRG 输入 (ADC) 001: BS 输出 (BSC) 101: 禁止设定 010: 禁止设定 110: 禁止设定 011: RxD0 输入 (SCIF) 111: 禁止设定

25.2.9 端口 F/I/O 寄存器 H、L (PFIORH、PFIORL)

PFIORH、PFIORL 均为 16 位可读取 / 写入的寄存器，选择端口 F 引脚的输入 / 输出方向。PF30IOR ~ PF0IOR 位分别对应 PF30/AUDIO_CLK ~ PF0/TCLKA/LCD_DATA0/SSCK0 引脚。端口 F 的引脚功能为通用输入 / 输出 (PF30 ~ PF0) 时，PFIORH、PFIORL 有效；在其他功能无效。如果将 PFIORH/PFIORL 位置 1，则对应的引脚为输出；清 0 时为输入。

PFIORH 的 bit15 为保留位。读取值、写入值总是为 0。

(1) 端口 F/I/O 寄存器 H (PFIORH)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PF30 IOR	PF29 IOR	PF28 IOR	PF27 IOR	PF26 IOR	PF25 IOR	PF24 IOR	PF23 IOR	PF22 IOR	PF21 IOR	PF20 IOR	PF19 IOR	PF18 IOR	PF17 IOR	PF16 IOR
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

(2) 端口 F/I/O 寄存器 L (PFIORL)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PF15 IOR	PF14 IOR	PF13 IOR	PF12 IOR	PF11 IOR	PF10 IOR	PF9 IOR	PF8 IOR	PF7 IOR	PF6 IOR	PF5 IOR	PF4 IOR	PF3 IOR	PF2 IOR	PF1 IOR	PF0 IOR
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

25.2.10 端口 F 控制寄存器 H1 ~ H4、L1 ~ L4 (PFCRH1 ~ PFCRH4、PFCRL1 ~ PFCRL4)

PFCRH1 ~ PFCRH4、PFCRL1 ~ PFCRL4 均为 16 位可读取 / 写入的寄存器，选择端口 F 多路复用引脚的功能。

(1) 端口 F 控制寄存器 H4 (PFCRH4)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	PF30 MD0	-	-	-	PF29 MD0	-	-	-	PF28 MD0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

位	位名称	初始值	R/W	说 明
15 ~ 9	—	均为 0	R	保留位 读取值、写入值总是为 0。
8	PF30MD0	0	R/W	PF30 模式 控制 PF30/AUDIO_CLK 引脚的功能。 0: PF30 输入 / 输出 (端口) 1: AUDIO_CLK 输入 (SSI)
7 ~ 5	—	均为 0	R	保留位 读取值、写入值总是为 0。

位	位名称	初始值	R/W	说 明
4	PF29MD0	0	R/W	PF29 模式 控制 PF29/SSIDATA3 引脚的功能。 0: PF29 输入 / 输出 (端口) 1: SSIDATA3 输入 / 输出 (SSI)
3 ~ 1	—	均为 0	R	保留位 读取值、写入值总是为 0。
0	PF28MD0	0	R/W	PF28 模式 控制 PF28/SSIWS3 引脚的功能。 0: PF28 输入 / 输出 (端口) 1: SSIWS3 输入 / 输出 (SSI)

(2) 端口 F 控制寄存器 H3 (PFCRH3)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	PF27 MD0	-	-	-	PF26 MD0	-	-	-	PF25 MD0	-	-	-	PF24 MD0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

位	位名称	初始值	R/W	说 明
15 ~ 13	—	均为 0	R	保留位 读取值、写入值总是为 0。
12	PF27MD0	0	R/W	PF27 模式 控制 PF27/SSISCK3 引脚的功能。 0: PF27 输入 / 输出 (端口) 1: SSISCK3 输入 / 输出 (SSI)
11 ~ 9	—	均为 0	R	保留位 读取值、写入值总是为 0。
8	PF26MD0	0	R/W	PF26 模式 控制 PF26/SSIDATA2 引脚的功能。 0: PF26 输入 / 输出 (端口) 1: SSIDATA2 输入 / 输出 (SSI)
7 ~ 5	—	均为 0	R	保留位 读取值、写入值总是为 0。
4	PF25MD0	0	R/W	PF25 模式 控制 PF25/SSIWS2 引脚的功能。 0: PF25 输入 / 输出 (端口) 1: SSIWS2 输入 / 输出 (SSI)
3 ~ 1	—	均为 0	R	保留位 读取值、写入值总是为 0。
0	PF24MD0	0	R/W	PF24 模式 选择 PF24/SSISCK2 引脚的功能。 0: PF24 输入 / 输出 (端口) 1: SSISCK2 输入 / 输出 (SSI)

(3) 端口 F 控制寄存器 H2 (PFCRH2)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	PF23MD[1:0]	-	-	PF22MD[1:0]	-	-	PF21MD[1:0]	-	-	PF20MD[1:0]	-	-	PF20MD[1:0]	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

位	位名称	初始值	R/W	说 明
15、14	—	均为 0	R	保留位 读取值、写入值总是为 0。
13、12	PF23MD[1:0]	00	R/W	PF23 模式 控制 PF23/SSIDATA1/LCD_VEPWC 引脚的功能。 00: PF23 输入 / 输出 (端口) 01: SSIDATA1 输入 / 输出 (SSI) 10: LCD_VEPWC 输出 (LCDC) 11: 禁止设定
11、10	—	均为 0	R	保留位 读取值、写入值总是为 0。
9、8	PF22MD[1:0]	00	R/W	PF22 模式 控制 PF22/SSIWS1/LCD_VCPWC 引脚的功能。 00: PF22 输入 / 输出 (端口) 01: SSIWS1 输入 / 输出 (SSI) 10: LCD_VCPWC 输出 (LCDC) 11: 禁止设定
7、6	—	均为 0	R	保留位 读取值、写入值总是为 0。
5、4	PF21MD[1:0]	00	R/W	PF21 模式 控制 PF21/SSISCK1/LCD_CLK 引脚的功能。 00: PF21 输入 / 输出 (端口) 01: SSISCK1 输入 / 输出 (SSI) 10: LCD_CLK 输入 (LCDC) 11: 禁止设定
3、2	—	均为 0	R	保留位 读取值、写入值总是为 0。
1、0	PF20MD[1:0]	00	R/W	PF20 模式 控制 PF20/SSIDATA0/LCD_FLM 引脚的功能。 00: PF20 输入 / 输出 (端口) 01: SSIDATA0 输入 / 输出 (SSI) 10: LCD_FLM 输出 (LCDC) 11: 禁止设定

(4) 端口 F 控制寄存器 H1 (PFCRH1)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	PF19MD[1:0]	-	-	PF18MD[1:0]	-	-	PF17MD[1:0]	-	-	PF16MD[1:0]	-	-	-	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

位	位名称	初始值	R/W	说 明
15、14	—	均为 0	R	保留位 读取值、写入值总是为 0。
13、12	PF19MD[1:0]	00	R/W	PF19 模式 控制 PF19/SSIWS0/LCD_M_DISP 引脚的功能。 00: PF19 输入 / 输出 (端口) 01: SSIWS0 输入 / 输出 (SSI) 10: LCD_M_DISP 输出 (LCDC) 11: 禁止设定
11、10	—	均为 0	R	保留位 读取值、写入值总是为 0。
9、8	PF18MD[1:0]	00	R/W	PF18 模式 控制 PF18/SSISCK0/LCD_CL2 引脚的功能。 00: PF18 输入 / 输出 (端口) 01: SSISCK0 输入 / 输出 (SSI) 10: LCD_CL2 输出 (LCDC) 11: 禁止设定
7、6	—	均为 0	R	保留位 读取值、写入值总是为 0。
5、4	PF17MD[1:0]	00	R/W	PF17 模式 控制 PF17/FCE/LCD_CL1 引脚的功能。 00: PF17 输入 / 输出 (端口) 01: FCE 输出 (FLCTL) 10: LCD_CL1 输出 (LCDC) 11: 禁止设定
3、2	—	均为 0	R	保留位 读取值、写入值总是为 0。
1、0	PF16MD[1:0]	00	R/W	PF16 模式 控制 PF16/FRB/LCD_DON 引脚的功能。 00: PF16 输入 / 输出 (端口) 01: FRB 输入 (FLCTL) 10: LCD_DON 输出 (LCDC) 11: 禁止设定

(5) 端口 F 控制寄存器 L4 (PFCRL4)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	PF15MD[1:0]	-	-	PF14MD[1:0]	-	-	PF13MD[1:0]	-	-	PF12MD[1:0]	-	-	-	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

位	位名称	初始值	R/W	说 明
15、14	—	均为 0	R	保留位 读取值、写入值总是为 0。
13、12	PF15MD[1:0]	00	R/W	PF15 模式 控制 PF15/NAF7/LCD_DATA15 引脚的功能。 00: PF15 输入 / 输出 (端口) 01: NAF7 输入 / 输出 (FLCTL) 10: LCD_DATA15 输出 (LCDC) 11: 禁止设定
11、10	—	均为 0	R	保留位 读取值、写入值总是为 0。
9、8	PF14MD[1:0]	00	R/W	PF14 模式 控制 PF14/NAF6/LCD_DATA14 引脚的功能。 00: PF14 输入 / 输出 (端口) 01: NAF6 输入 / 输出 (FLCTL) 10: LCD_DATA14 输出 (LCDC) 11: 禁止设定
7、6	—	均为 0	R	保留位 读取值、写入值总是为 0。
5、4	PF13MD[1:0]	00	R/W	PF13 模式 控制 PF13/NAF5/LCD_DATA13 引脚的功能。 00: PF13 输入 / 输出 (端口) 01: NAF5 输入 / 输出 (FLCTL) 10: LCD_DATA13 输出 (LCDC) 11: 禁止设定
3、2	—	均为 0	R	保留位 读取值、写入值总是为 0。
1、0	PF12MD[1:0]	00	R/W	PF12 模式 控制 PF12/NAF4/LCD_DATA12 引脚的功能。 00: PF12 输入 / 输出 (端口) 01: NAF4 输入 / 输出 (FLCTL) 10: LCD_DATA12 输出 (LCDC) 11: 禁止设定

(6) 端口 F 控制寄存器 L3 (PFCRL3)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	PF11MD[1:0]	-	-	PF10MD[1:0]	-	-	PF9MD[1:0]	-	-	PF8MD[1:0]				
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

位	位名称	初始值	R/W	说 明
15、14	—	均为 0	R	保留位 读取值、写入值总是为 0。
13、12	PF11MD[1:0]	00	R/W	PF11 模式 控制 PF11/NAF3/LCD_DATA11 引脚的功能。 00: PF11 输入 / 输出 (端口) 01: NAF3 输入 / 输出 (FLCTL) 10: LCD_DATA11 输出 (LCDC) 11: 禁止设定
11、10	—	均为 0	R	保留位 读取值、写入值总是为 0。
9、8	PF10MD[1:0]	00	R/W	PF10 模式 控制 PF10/NAF2/LCD_DATA10 引脚的功能。 00: PF10 输入 / 输出 (端口) 01: NAF2 输入 / 输出 (FLCTL) 10: LCD_DATA10 输出 (LCDC) 11: 禁止设定
7、6	—	均为 0	R	保留位 读取值、写入值总是为 0。
5、4	PF9MD[1:0]	00	R/W	PF9 模式 控制 PF9/NAF1/LCD_DATA9 引脚的功能。 00: PF9 输入 / 输出 (端口) 01: NAF1 输入 / 输出 (FLCTL) 10: LCD_DATA9 输出 (LCDC) 11: 禁止设定
3、2	—	均为 0	R	保留位 读取值、写入值总是为 0。
1、0	PF8MD[1:0]	00	R/W	PF8 模式 控制 PF8/NAF0/LCD_DATA8 引脚的功能。 00: PF8 输入 / 输出 (端口) 01: NAF0 输入 / 输出 (FLCTL) 10: LCD_DATA8 输出 (LCDC) 11: 禁止设定

(7) 端口 F 控制寄存器 F2 (PFCRL2)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	PF7MD[1:0]	-	-	PF6MD[1:0]	-	-	PF5MD[1:0]	-	-	PF4MD[1:0]	-	-	PF3MD[1:0]	-
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

位	位名称	初始值	R/W	说 明
15、14	—	均为 0	R	保留位 读取值、写入值总是为 0。
13、12	PF7MD[1:0]	00	R/W	PF7 模式 控制 PF7/FSC/LCD_DATA7/SCS1 引脚的功能。 00: PF7 输入 / 输出 (端口) 01: FSC 输出 (FLCTL) 10: LCD_DATA7 输出 (LCDC) 11: SCS1 输入 / 输出 (SSU)
11、10	—	均为 0	R	保留位 读取值、写入值总是为 0。
9、8	PF6MD[1:0]	00	R/W	PF6 模式 控制 PF6/FOE/LCD_DATA6/SSO1 引脚的功能。 00: PF6 输入 / 输出 (端口) 01: FOE 输出 (FLCTL) 10: LCD_DATA6 输出 (LCDC) 11: SSO1 输入 / 输出 (SSU)
7、6	—	均为 0	R	保留位 读取值、写入值总是为 0。
5、4	PF5MD[1:0]	00	R/W	PF5 模式 控制 PF5/FCDE/LCD_DATA5/SSI1 引脚的功能。 00: PF5 输入 / 输出 (端口) 01: FCDE 输出 (FLCTL) 10: LCD_DATA5 输出 (LCDC) 11: SSI1 输入 / 输出 (SSU)
3、2	—	均为 0	R	保留位 读取值、写入值总是为 0。
1、0	PF4MD[1:0]	00	R/W	PF4 模式 控制 PF4/FWE/LCD_DATA4/SSCK1 引脚的功能。 00: PF4 输入 / 输出 (端口) 01: FWE 输出 (FLCTL) 10: LCD_DATA4 输出 (LCDC) 11: SSCK1 输入 / 输出 (SSU)

(8) 端口 F 控制寄存器 L1 (PFCRL1)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	PF3MD[1:0]	-	-	PF2MD[1:0]	-	-	PF1MD[1:0]	-	-	PF0MD[1:0]				
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

位	位名称	初始值	R/W	说 明
15、14	—	均为 0	R	保留位 读取值、写入值总是为 0。
13、12	PF3MD[1:0]	00	R/W	PF3 模式 控制 PF3/TCLKD/LCD_DATA3/SCS0 引脚的功能。 00: PF3 输入 / 输出 (端口) 01: TCLKD 输入 (MTU2) 10: LCD_DATA3 输出 (LCDC) 11: SCS0 输入 / 输出 (SSU)
11、10	—	均为 0	R	保留位 读取值、写入值总是为 0。
9、8	PF2MD[1:0]	00	R/W	PF2 模式 控制 PF2/TCLKC/LCD_DATA2/SSO0 引脚的功能。 00: PF2 输入 / 输出 (端口) 01: TCLKC 输入 (MTU2) 10: LCD_DATA2 输出 (LCDC) 11: SSO0 输入 / 输出 (SSU)
7、6	—	均为 0	R	保留位 读取值、写入值总是为 0。
5、4	PF1MD[1:0]	00	R/W	PF1 模式 控制 PF1/TCLKB/LCD_DATA1/SSI0 引脚的功能。 00: PF1 输入 / 输出 (端口) 01: TCLKB 输入 (MTU2) 10: LCD_DATA1 输出 (LCDC) 11: SSI0 输入 / 输出 (SSU)
3、2	—	均为 0	R	保留位 读取值、写入值总是为 0。
1、0	PF0MD[1:0]	00	R/W	PF0 模式 控制 PF0/TCLKA/LCD_DATA0/SSCK0 引脚的功能。 00: PF0 输入 / 输出 (端口) 01: TCLKA 输入 (MTU2) 10: LCD_DATA0 输出 (LCDC) 11: SSCK0 输入 / 输出 (SSU)

25.2.11 IRQOUT 功能控制寄存器 (IFCR)

IFCR 为 16 位可读取 / 写入的寄存器，由端口 B 控制寄存器 L4 (PBCRL4) 设定多路复用功能为 $\overline{\text{IRQOUT}}$ / $\overline{\text{REFOUT}}$ 输出时，使用该寄存器控制其输出。PBCRL4 的设定为其他功能时，该寄存器的设定不影响引脚的功能。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	PB12IRQ[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

位	位名称	初始值	R/W	说 明
15 ~ 2	—	均为 0	R	保留位 读取值、写入值总是为 0。
1、0	PB12IRQ[1:0]	00	R/W	PB12IRQOUT 模式 选择设定 PBCRL4 的 bit1、0 (PB12MD[1:0]) 为 (1、0) 时的 $\overline{\text{IRQOUT}}$ / $\overline{\text{REFOUT}}$ 引脚功能。 00: 输出中断请求接受信号 01: 输出刷新信号 10: 输出中断请求接受信号或输出刷新信号 (输出的信号取决于运行状态) 11: 总是为高电平输出

25.2.12 SSI 过采样时钟选择寄存器 (SCSR)

SCSR 为 16 位可读取 / 写入的寄存器，选择 SSI 使用的过采样时钟的提供源及分频比。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	SSI3CKS[2:0]			-	SSI2CKS[2:0]			-	SSI1CKS[2:0]			-	SSI0CKS[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15	—	0	R	保留位 读取值、写入值总是为 0。
14 ~ 12	SSI3CKS[2:0]	000	R/W	选择 SSI ch3 时钟 选择 SSI ch3 使用的过采样时钟的提供源。 参照表 25.8。
11	—	0	R	保留位 读取值、写入值总是为 0。
10 ~ 8	SSI2CKS[2:0]	000	R/W	选择 SSI ch2 时钟 选择 SSI ch2 使用的过采样时钟的提供源。 参照表 25.8。
7	—	0	R	保留位 读取值、写入值总是为 0。
6 ~ 4	SSI1CKS[2:0]	000	R/W	选择 SSI ch1 时钟 选择 SSI ch1 使用的过采样时钟的提供源。 参照表 25.8
3	—	0	R	保留位 读取值、写入值总是为 0。
2 ~ 0	SSI0CKS[2:0]	000	R/W	选择 SSI ch0 时钟 选择 SSI ch0 使用的过采样时钟的提供源。 参照表 25.8。

表 25.8 通过 SSInCKS 位选择过采样时钟提供源

SSInCKS[2:0]*1 设定值	时钟运行模式		
	0 或 1	2	3
000	AUDIO_X1 输入		
001	AUDIO_X1 输入 /4		
010	AUDIO_CLK 输入 *2		
011	AUDIO_CLK 输入 *2/4		
100	EXTAL 输入	CKIO 输入	禁止设定
101	EXTAL 输入 /4	CKIO 输入 /4	禁止设定
110	EXTAL 输入 /2	CKIO 输入 /2	禁止设定
111	EXTAL 输入 /8	CKIO 输入 /8	禁止设定

【注】 *1 n=0 ~ 3

*2 使用 AUDIO_CLK 时，必须将端口 F 控制寄存器 H4 (PFCRH4) 的 PF30MD0 位设定为 1。

25.3 有关端口 A 的功能转换

端口 A 与 A/D 转换器的模拟输入引脚、D/A 转换器的模拟输出引脚多路复用。通过设定 A/D 转换器的 A/D 控制 / 状态寄存器、D/A 转换器的 D/A 控制寄存器，自动转换引脚功能（详情参阅“第 20 章 A/D 转换器 (ADC)”、“第 21 章 D/A 转换器 (DAC)”）。

表 25.9 PA6/AN6/DA0 及 PA7/AN7/DA1 的功能转换

DACR 设定值 (DAE、DAOE0、DAE1)	ADCSR 设定值		引脚功能		备注
	CH[2:0]	MDS[2]	PA6/AN6/DA0	PA7/AN7/DA1	
(x、0、0)	110	x	AN6	PA7	
	111	0	PA6	AN7	
		1	AN6	AN7	
(0、1、0)	110	x	AN6/DA0	PA7	禁止设定
	111	0	DA0	AN7	
		1	AN6/DA0	AN7	禁止设定
(0、0、1)	110	x	AN6	DA1	
	111	0	PA6	AN7/DA1	禁止设定
		1	AN6	AN7/DA1	禁止设定
(x、1、1) / (1、0、1) / (1、1、0)	110	x	AN6/DA0	DA1	禁止设定
	111	0	DA0	AN7/DA1	禁止设定
		1	AN6/DA0	AN7/DA1	禁止设定

【符号说明】

x: Don't care

【注】 因为对 PA6 或 PA7 引脚同时选择 A/D 转换器与 D/A 转换器的功能，所以为“禁止设定”。

25.4 使用时的注意事项

表 25.1 ~ 表 25.6 所示的多路复用一览表中，除 PA0 ~ PA7、PB0 ~ PB7 之外所有引脚的 I/O 缓冲器均带有弱保持电路，以便抑止引脚电位变为中间电位状态。但需注意，在高阻抗状态保持的电位，会因噪声等的影响而发生变化。

第 26 章 I/O 端口

本 LSI 的端口由 A、B、C、D、E、F 共 6 个端口构成。

各端口的引脚均为兼备其他功能的多路复用引脚，由引脚功能控制器（PFC）选择其功能。

各端口分别有保存引脚数据的数据寄存器和读取引脚值的端口寄存器。

26.1 特点

1. 端口总数：99 个（输入/输出 82 个、输入 16 个、输出 1 个）

- 端口 A：输入 8 个
- 端口 B：输入/输出 4 个、输入 8 个、输出 1 个
- 端口 C：输入/输出 15 个
- 端口 D：输入/输出 16 个
- 端口 E：输入/输出 16 个
- 端口 F：输入/输出 31 个

2. 带弱保持的引脚

本 LSI 的以下 I/O 引脚附带有抑制引脚电位变为中间电位状态的弱保持电路。

- 端口 B：PB8～PB12
- 端口 C：PC0～PC14
- 端口 D：PD0～PD15
- 端口 E：PE0～PE15
- 端口 F：PF0～PF30

弱保持电路内置于 I/O 引脚，未从外部驱动 I/O 引脚时，将输入固定为高电平或低电平。一般的 CMOS 产品中，未使用的输入引脚需通过外部上拉或下拉电阻固定输入电平，而本 LSI 带弱保持的 I/O 引脚无需这些外置电路，因此可减少部件个数。

另外，需通过上拉或下拉电阻固定引脚电平时，电阻值必须小于等于 10kΩ。

26.2 端口 A

端口 A 是有 8 个引脚的输入端口，如图 26.1 所示。

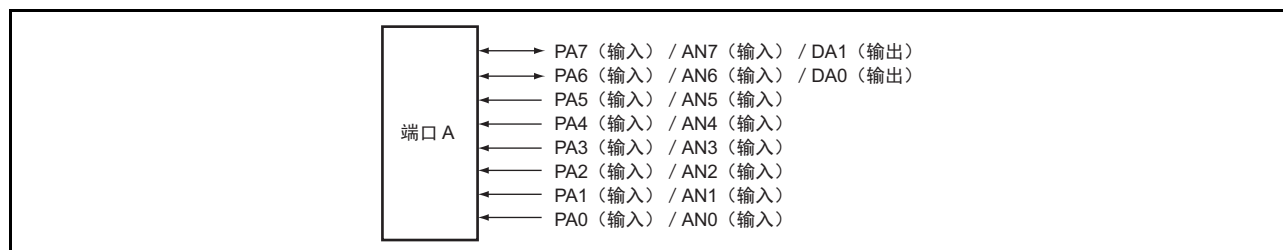


图 26.1 端口 A

26.2.1 寄存器说明

端口 A 的寄存器如表 26.1 所示。

表 26.1 寄存器结构

寄存器名称	简称	R/W	初始值	地址	存取长度
端口 A 数据寄存器 L	PADRL	R	H'00xx	H'FFFE3802	8、16

26.2.2 端口 A 数据寄存器 L（PADRL）

PADRL 为 16 位只读寄存器，PA7DR ~ PA0DR 位分别对应 PA7/AN7/DA1 ~ PA0/AN0 引脚。PA7 ~ PA0 的通用输入功能仅在 A/D 转换器及 D/A 转换器停止时有效。

即使对这些位写入值也被忽略，不影响引脚状态。读取时，直接读取引脚状态而非位的值，但不可在 A/D 转换器及 D/A 转换器运行时读取。PADRL 的读取 / 写入如表 26.2 所示。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	PA7 DR	PA6 DR	PA5 DR	PA4 DR	PA3 DR	PA2 DR	PA1 DR	PA0 DR
初始值:	0	0	0	0	0	0	0	0	*	*	*	*	*	*	*	*
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

【注】* 取决于外部引脚的状态。

位	位名称	初始值	R/W	说明
15 ~ 8	—	均为 0	R	保留位 读取值、写入值总是为 0。
7	PA7DR	引脚状态	R	参照表 26.2
6	PA6DR	引脚状态	R	
5	PA5DR	引脚状态	R	
4	PA4DR	引脚状态	R	
3	PA3DR	引脚状态	R	
2	PA2DR	引脚状态	R	
1	PA1DR	引脚状态	R	
0	PA0DR	引脚状态	R	

表 26.2 端口 A 数据寄存器 L（PADRL）的读取 / 写入

- PADRL 的 bit7 ~ 0

引脚功能	读取	写入
通用输入	引脚状态	忽略（不影响引脚状态）
ANn 输入 / DAn 输出	禁止	忽略（不影响引脚状态）

【符号说明】n=7 ~ 0。但 DA 仅为 DA0 及 DA1。

26.3 端口 B

端口 B 是有 13 个引脚的输入 / 输出端口，如图 26.2 所示。

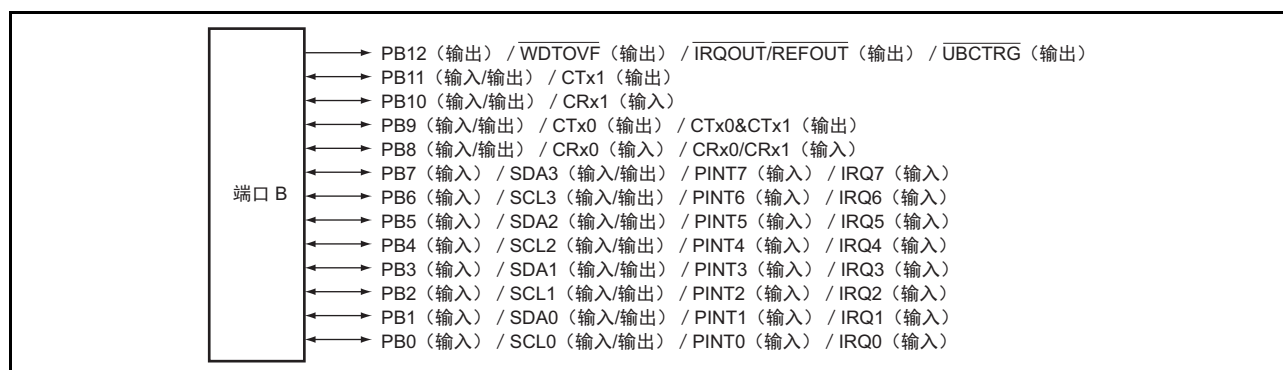


图 26.2 端口 B

26.3.1 寄存器说明

端口 B 的寄存器如表 26.3 所示。

表 26.3 寄存器结构

寄存器名称	简称	R/W	初始值	地址	存取长度
端口 B 数据寄存器 L	PBDRL	R/W	H'00xx	H'FFFE3882	8、16
端口 B 端口寄存器 L	PBPRL	R	H'xxxx	H'FFFE389E	8、16

26.3.2 端口 B 数据寄存器 L（PBDRL）

PBDRL 为 16 位可读取 / 写入的寄存器，保存端口 B 的数据。PB12DR ~ PB0DR 位分别对应 PB12/WDTOVF/IRQOUT/REFOUT/UBCTRG ~ PB0/SCL0/PINT0/IRQ0 引脚。

引脚功能为通用输出时，如果对 PBDRL 写入值，则从引脚输出该值；读取时，与引脚状态无关，可直接读取寄存器的值。

引脚功能为通用输入时，如果读取 PBDRL，则直接读取引脚状态而非寄存器的值；写入时，可对 PBDRL 写入值，不影响引脚状态。PBDRL 的读取 / 写入如表 26.4 所示。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	PB12 DR	PB11 DR	PB10 DR	PB9 DR	PB8 DR	PB7 DR	PB6 DR	PB5 DR	PB4 DR	PB3 DR	PB2 DR	PB1 DR	PB0 DR
初始值:	0	0	0	0	0	0	0	0	*	*	*	*	*	*	*	*
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

【注】* 取决于外部引脚的状态。

位	位名称	初始值	R/W	说明
15 ~ 13	—	均为 0	R	保留位 读取值、写入值总是为 0。
12	PB12DR	0	R/W	参照表 26.4
11	PB11DR	0	R/W	
10	PB10DR	0	R/W	
9	PB9DR	0	R/W	
8	PB8DR	0	R/W	
7	PB7DR	引脚状态	R	
6	PB6DR	引脚状态	R	
5	PB5DR	引脚状态	R	
4	PB4DR	引脚状态	R	
3	PB3DR	引脚状态	R	
2	PB2DR	引脚状态	R	
1	PB1DR	引脚状态	R	
0	PB0DR	引脚状态	R	

表 26.4 端口 B 数据寄存器 L (PBDRL) 的读取 / 写入

- PBDRL 的 bit12

引脚功能	读取	写入
通用输出	PBDRL 的值	从引脚输出写入值
非通用输出	PBDRL 的值	可写入 PBDRL, 不影响引脚状态

- PBDRL 的 bit11 ~ 8

PBIORL	引脚功能	读取	写入
0	通用输入	引脚状态	可写入 PBDRL, 不影响引脚状态
	非通用输入	引脚状态	可写入 PBDRL, 不影响引脚状态
1	通用输出	PBDRL 的值	从引脚输出写入值
	非通用输出	PBDRL 的值	可写入 PBDRL, 不影响引脚状态

- PBDRL 的 bit7 ~ 0

引脚功能	读取	写入
通用输入	引脚状态	无效
非通用输入	引脚状态	无效

26.3.3 端口 B 端口寄存器 L (PBPR L)

PBPR L 为 16 位只读寄存器，PB11PR ~ PB0PR 位分别对应 PB11/CTx1 ~ PB0/SCL0/PINT0/IRQ0 引脚。与 PFC 的设定无关，PBPR L 总是可读取引脚的值。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	PB11 PR	PB10 PR	PB9 PR	PB8 PR	PB7 PR	PB6 PR	PB5 PR	PB4 PR	PB3 PR	PB2 PR	PB1 PR	PB0 PR
初始值:	0	0	0	0	*	*	*	*	*	*	*	*	*	*	*	*
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

【注】* 取决于外部引脚的状态。

位	位名称	初始值	R/W	说明
15 ~ 12	—	均为 0	R	保留位 读取值、写入值总是为 0。
11	PB11PR	引脚状态	R	与 PFC 的设定无关，可读取引脚状态，写入无效。
10	PB10PR	引脚状态	R	
9	PB9PR	引脚状态	R	
8	PB8PR	引脚状态	R	
7	PB7PR	引脚状态	R	
6	PB6PR	引脚状态	R	
5	PB5PR	引脚状态	R	
4	PB4PR	引脚状态	R	
3	PB3PR	引脚状态	R	
2	PB2PR	引脚状态	R	
1	PB1PR	引脚状态	R	
0	PB0PR	引脚状态	R	

26.4 端口 C

端口 C 是有 15 个引脚的输入 / 输出端口，如图 26.3 所示。

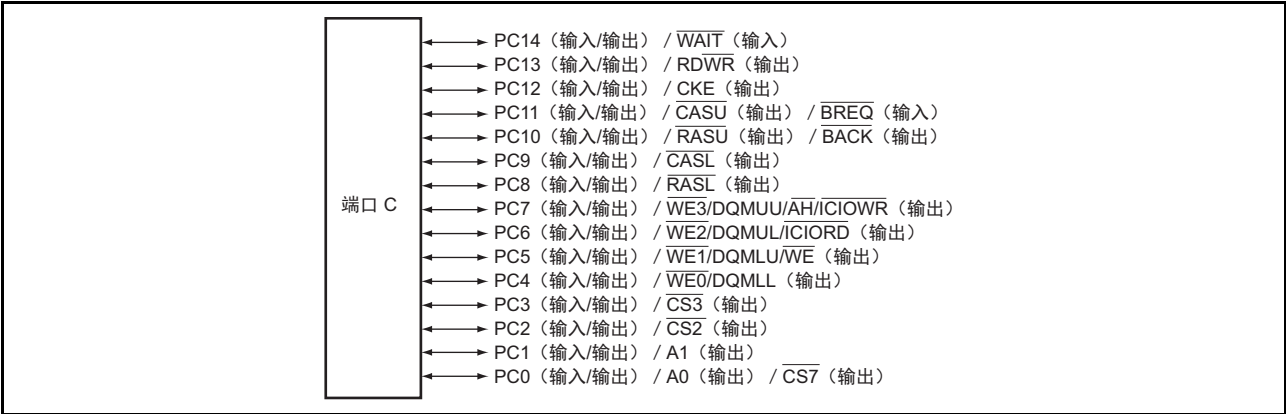


图 26.3 端口 C

26.4.1 寄存器说明

端口 C 的寄存器如表 26.5 所示。

表 26.5 寄存器结构

寄存器名称	简称	R/W	初始值	地址	存取长度
端口 C 数据寄存器 L	PCDRL	R/W	H'xxxx	H'FFFE3902	8、16
端口 C 端口寄存器 L	PCPRL	R	H'xxxx	H'FFFE391E	8、16

26.4.2 端口 C 数据寄存器 L (PCDRL)

PCDRL 为 16 位可读取 / 写入的寄存器，保存端口 C 的数据。PC14DR ~ PC0DR 位分别对应 PC14/WAIT ~ PC0/A0/CS7 引脚。

引脚功能为通用输出时，如果对 PCDRL 写入值，则从引脚输出该值；读取时，与引脚状态无关，可直接读取寄存器的值。

引脚功能为通用输入时，如果读取 PCDRL，则直接读取引脚状态而非寄存器的值；写入时，可对 PCDRL 写入值，不影响引脚状态。PCDRL 的读取 / 写入如表 26.6 所示。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PC14 DR	PC13 DR	PC12 DR	PC11 DR	PC10 DR	PC9 DR	PC8 DR	PC7 DR	PC6 DR	PC5 DR	PC4 DR	PC3 DR	PC2 DR	PC1 DR	PC0 DR
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说明
15	—	0	R	保留位 读取值、写入值总是为 0。
14	PC14DR	0	R/W	参照表 26.6
13	PC13DR	0	R/W	
12	PC12DR	0	R/W	
11	PC11DR	0	R/W	
10	PC10DR	0	R/W	
9	PC9DR	0	R/W	
8	PC8DR	0	R/W	
7	PC7DR	0	R/W	
6	PC6DR	0	R/W	
5	PC5DR	0	R/W	
4	PC4DR	0	R/W	
3	PC3DR	0	R/W	
2	PC2DR	0	R/W	
1	PC1DR	0	R/W	
0	PC0DR	0	R/W	

表 26.6 端口 C 数据寄存器 L（PCDRL）的读取 / 写入

• PCDRL 的 bit14 ~ 0

PCIORL	引脚功能	读取	写入
0	通用输入	引脚状态	可写入 PCDRL，不影响引脚状态
	非通用输入	引脚状态	可写入 PCDRL，不影响引脚状态
1	通用输入	PCDRL 的值	从引脚输出写入值
	非通用输入	PCDRL 的值	可写入 PCDRL，不影响引脚状态

26.4.3 端口 C 端口寄存器 L（PCPRL）

PCPRL 为 16 位只读寄存器，PC14PR ~ PC0PR 位分别对应 PC14/ $\overline{\text{WAIT}}$ ~ PC0/A0/ $\overline{\text{CS}}$ 7 引脚。与 PFC 的设置无关，PCPRL 通常可读取引脚的值。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PC14 PR	PC13 PR	PC12 PR	PC11 PR	PC10 PR	PC9 PR	PC8 PR	PC7 PR	PC6 PR	PC5 PR	PC4 PR	PC3 PR	PC2 PR	PC1 PR	PC0 PR
初始值:	0	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

【注】* 取决于外部引脚的状态。

位	位名称	初始值	R/W	说明
15	—	0	R	保留位 读取值、写入值总是为 0。
14	PC14PR	引脚状态	R	与 PFC 的设置无关，可读取引脚状态，写入无效。
13	PC13PR	引脚状态	R	
12	PC12PR	引脚状态	R	
11	PC11PR	引脚状态	R	
10	PC10PR	引脚状态	R	
9	PC9PR	引脚状态	R	
8	PC8PR	引脚状态	R	
7	PC7PR	引脚状态	R	
6	PC6PR	引脚状态	R	
5	PC5PR	引脚状态	R	
4	PC4PR	引脚状态	R	
3	PC3PR	引脚状态	R	
2	PC2PR	引脚状态	R	
1	PC1PR	引脚状态	R	
0	PC0PR	引脚状态	R	

26.5 端口 D

端口 D 是有 16 个引脚的输入 / 输出端口，如图 26.4 所示。

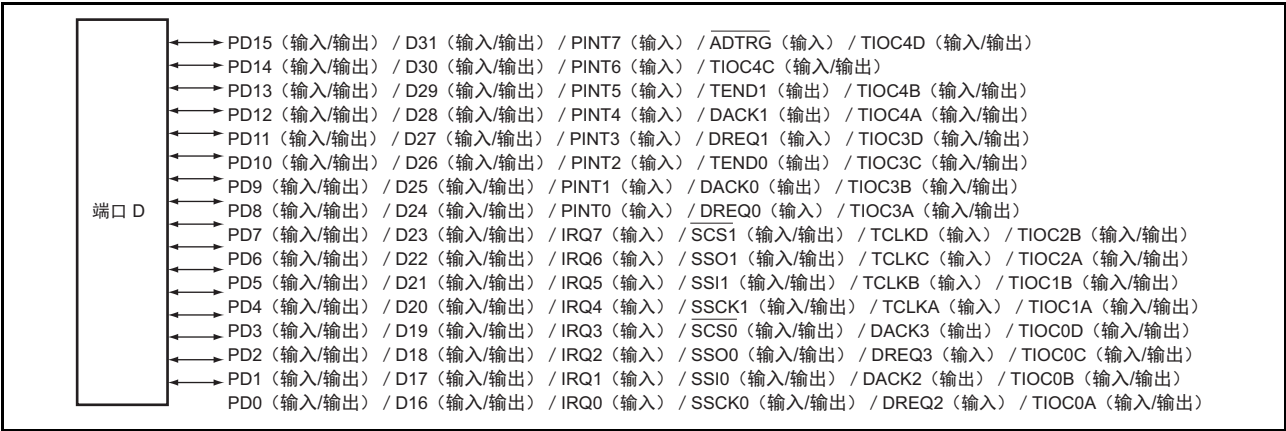


图 26.4 端口 D

26.5.1 寄存器说明

端口 D 的寄存器如表 26.7 所示。

表 26.7 寄存器结构

寄存器名称	简称	R/W	初始值	地址	存取长度
端口 D 数据寄存器 L	PDDRL	R/W	H'0000	H'FFFE3982	8、16
端口 D 端口寄存器 L	PDPRL	R	H'xxxx	H'FFFE399E	8、16

26.5.2 端口 D 数据寄存器 L（PDDRL）

PDDRL 为 16 位可读取 / 写入的寄存器，保存端口 D 的数据。PD15DR ~ PD0DR 位分别对应 PD15/D31/PINT7/ADTRG/TIOC4D ~ PD0/D16/IRQ0/SSCK0/DREQ2/TIOC0A 引脚。

引脚功能为通用输出时，如果对 PDDRL 写入值，则从引脚输出该值；读取时，与引脚状态无关，可直接读取寄存器的值。

引脚功能为通用输入时，如果读取 PDDRL，则直接读取引脚状态而非寄存器的值；写入时，可对 PDDRL 写入值，不影响引脚状态。PDDRL 的读取 / 写入如表 26.8 所示。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PD15 DR	PD14 DR	PD13 DR	PD12 DR	PD11 DR	PD10 DR	PD9 DR	PD8 DR	PD7 DR	PD6 DR	PD5 DR	PD4 DR	PD3 DR	PD2 DR	PD1 DR	PD0 DR
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说明
15	PD15DR	0	R/W	参照表 26.8
14	PD14DR	0	R/W	
13	PD13DR	0	R/W	
12	PD12DR	0	R/W	
11	PD11DR	0	R/W	
10	PD10DR	0	R/W	
9	PD9DR	0	R/W	
8	PD8DR	0	R/W	
7	PD7DR	0	R/W	
6	PD6DR	0	R/W	
5	PD5DR	0	R/W	
4	PD4DR	0	R/W	
3	PD3DR	0	R/W	
2	PD2DR	0	R/W	
1	PD1DR	0	R/W	
0	PD0DR	0	R/W	

表 26.8 端口 D 数据寄存器 L（PDDRL）的读取 / 写入

- PDDRL 的 bit15 ~ 0

PDIORL	引脚功能	读取	写入
0	通用输入	引脚状态	可写入 PDDRL，不影响引脚状态
	非通用输入	引脚状态	可写入 PDDRL，不影响引脚状态
1	通用输入	PDDRL 的值	从引脚输出写入值
	非通用输入	PDDRL 的值	可写入 PDDRL，不影响引脚状态

26.5.3 端口 D 端口寄存器 L（PDPRL）

PDPRL 为 16 位只读寄存器，PD15PR ~ PD0PR 位分别对应 PD15/D31/PINT7/ADTRG/TIOC4D ~ PD0/D16/IRQ0/SSCK0/DREQ2/TIOC0A 引脚。与 PFC 的设定无关，PDPRL 总是可读取引脚的值。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PD15 PR	PD14 PR	PD13 PR	PD12 PR	PD11 PR	PD10 PR	PD9 PR	PD8 PR	PD7 PR	PD6 PR	PD5 PR	PD4 PR	PD3 PR	PD2 PR	PD1 PR	PD0 PR
初始值:	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

【注】* 取决于外部引脚的状态。

位	位名称	初始值	R/W	说明
15	PD15PR	引脚状态	R	与 PFC 的设定无关，可读取引脚状态，写入无效。
14	PD14PR	引脚状态	R	
13	PD13PR	引脚状态	R	
12	PD12PR	引脚状态	R	
11	PD11PR	引脚状态	R	
10	PD10PR	引脚状态	R	
9	PD9PR	引脚状态	R	
8	PD8PR	引脚状态	R	
7	PD7PR	引脚状态	R	
6	PD6PR	引脚状态	R	
5	PD5PR	引脚状态	R	
4	PD4PR	引脚状态	R	
3	PD3PR	引脚状态	R	
2	PD2PR	引脚状态	R	
1	PD1PR	引脚状态	R	
0	PD0PR	引脚状态	R	

26.6 端口 E

端口 E 是有 16 个引脚的输入 / 输出端口，如图 26.5 所示。



图 26.5 端口 E

26.6.1 寄存器说明

端口 E 的寄存器如表 26.9 所示。

表 26.9 寄存器结构

寄存器名称	简称	R/W	初始值	地址	存取长度
端口 E 数据寄存器 L	PEDRL	R/W	H'0000	H'FFFE3A02	8、16
端口 E 端口寄存器 L	PEPRL	R	H'xxxx	H'FFFE3A1E	8、16

26.6.2 端口 E 数据寄存器 L (PEDRL)

PEDRL 为 16 位可读取 / 写入的寄存器，保存端口 E 的数据。PE15DR ~ PE0DR 位分别对应 PE15/IOIS16/RTS3 ~ PE0/BS/RxD0/ADTRG 引脚。

引脚功能为通用输出时，如果对 PEDRL 写入值，则从引脚输出该值；读取时，与引脚状态无关，可直接读取寄存器的值。

引脚功能为通用输入时，如果读取 PEDRL，则直接读取引脚状态而非寄存器的值；写入时，可对 PEDRL 写入值，不影响引脚状态。PEDRL 的读取 / 写入如表 26.10 所示。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PE15 DR	PE14 DR	PE13 DR	PE12 DR	PE11 DR	PE10 DR	PE9 DR	PE8 DR	PE7 DR	PE6 DR	PE5 DR	PE4 DR	PE3 DR	PE2 DR	PE1 DR	PE0 DR
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说明
15	PE15DR	0	R/W	参照表 26.10
14	PE14DR	0	R/W	
13	PE13DR	0	R/W	
12	PE12DR	0	R/W	
11	PE11DR	0	R/W	
10	PE10DR	0	R/W	
9	PE9DR	0	R/W	
8	PE8DR	0	R/W	
7	PE7DR	0	R/W	
6	PE6DR	0	R/W	
5	PE5DR	0	R/W	
4	PE4DR	0	R/W	
3	PE3DR	0	R/W	
2	PE2DR	0	R/W	
1	PE1DR	0	R/W	
0	PE0DR	0	R/W	

表 26.10 端口 E 数据寄存器 L (PEDRL) 的读取 / 写入

- PEDRL 的 bit15 ~ 0

PEIORL	引脚功能	读取	写入
0	通用输入	引脚状态	可写入 PEDRL，不影响引脚状态
	非通用输入	引脚状态	可写入 PEDRL，不影响引脚状态
1	通用输入	PEDRL 的值	从引脚输出写入值
	非通用输入	PEDRL 的值	可写入 PEDRL，不影响引脚状态

26.6.3 端口 E 端口寄存器 L (PEPRL)

PEPRL 为 16 位只读寄存器，PE15PR ~ PE0PR 位分别对应 PE15/IOIS16/RTS3 ~ PE0/BS/RxD0/ADTRG 引脚。与 PFC 的设定无关，PEPRL 总是可读取引脚的值。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PE15 PR	PE14 PR	PE13 PR	PE12 PR	PE11 PR	PE10 PR	PE9 PR	PE8 PR	PE7 PR	PE6 PR	PE5 PR	PE4 PR	PE3 PR	PE2 PR	PE1 PR	PE0 PR
初始值:	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

【注】* 取决于外部引脚的状态。

位	位名称	初始值	R/W	说明
15	PE15PR	引脚状态	R	与 PFC 的设定无关，可读取引脚状态，写入无效。
14	PE14PR	引脚状态	R	
13	PE13PR	引脚状态	R	
12	PE12PR	引脚状态	R	
11	PE11PR	引脚状态	R	
10	PE10PR	引脚状态	R	
9	PE9PR	引脚状态	R	
8	PE8PR	引脚状态	R	
7	PE7PR	引脚状态	R	
6	PE6PR	引脚状态	R	
5	PE5PR	引脚状态	R	
4	PE4PR	引脚状态	R	
3	PE3PR	引脚状态	R	
2	PE2PR	引脚状态	R	
1	PE1PR	引脚状态	R	
0	PE0PR	引脚状态	R	

26.7 端口 F

端口 F 是有 31 个引脚的输入/输出端口，如图 26.6 所示。

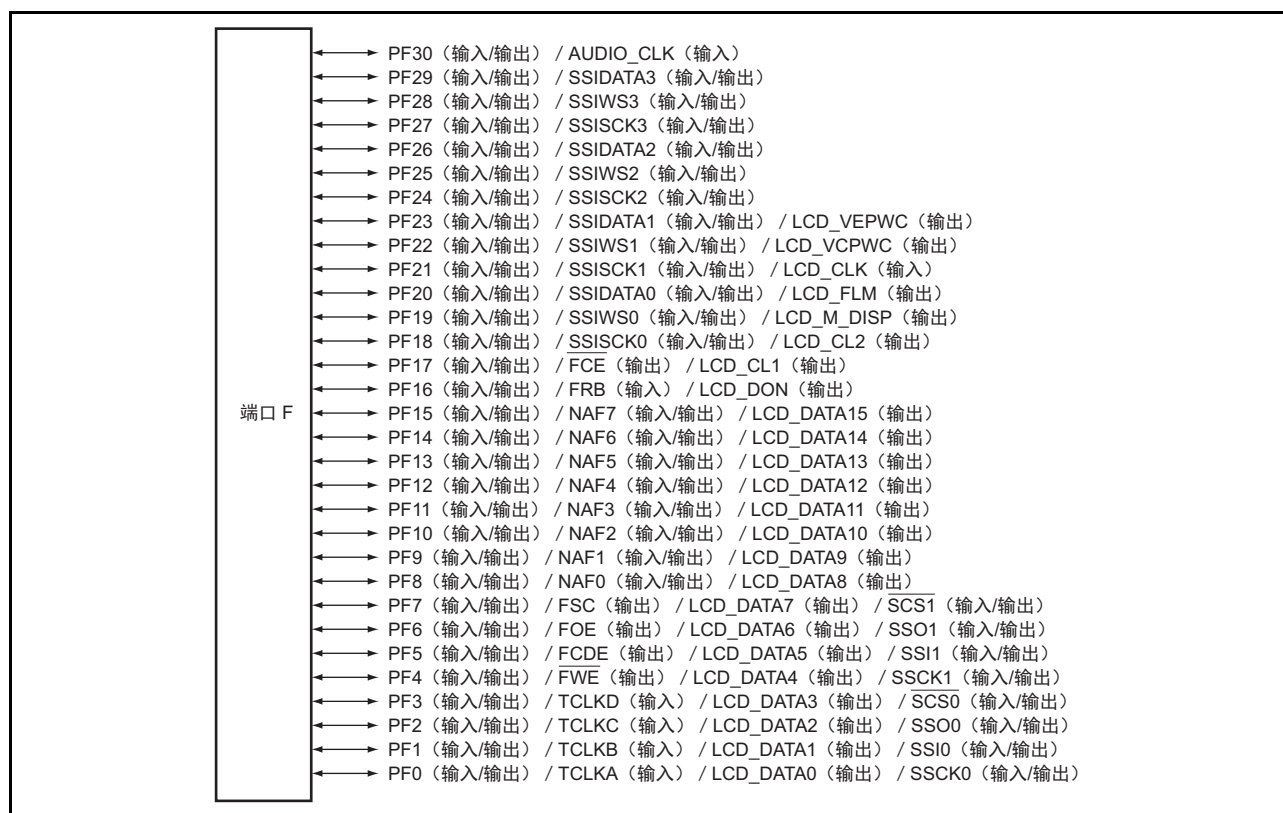


图 26.6 端口 F

26.7.1 寄存器说明

端口 F 的寄存器如表 26.11 所示。

表 26.11 寄存器结构

寄存器名称	简称	R/W	初始值	地址	存取长度
端口 F 数据寄存器 H	PFDRH	R/W	H'0000	H'FFFE3A80	8、16、32
端口 F 数据寄存器 L	PFDRL	R/W	H'0000	H'FFFE3A82	8、16
端口 F 端口寄存器 H	PFPRH	R	H'xxxx	H'FFFE3A9C	8、16、32
端口 F 端口寄存器 L	PFPRL	R	H'xxxx	H'FFFE3A9E	8、16

26.7.2 端口 F 数据寄存器 H、L（PFDRH、PFDRL）

PFDRH、PFDRL 为 16 位可读取 / 写入的寄存器，保存端口 F 的数据。PF30DR ~ PF0DR 位分别对应 PF30/AUDIO_CLK ~ PF0/TCLKA/LCD_DATA0/SSCK0 引脚。

引脚功能为通用输出时，如果对 PFDRH、PFDRL 写入值，则从引脚输出该值；读取时，与引脚状态无关，可直接读取寄存器的值。

引脚功能为通用输入时，如果读取 PFDRH、PFDRL，则直接读取引脚状态而非寄存器的值；写入时，可对 PFDRH、PFDRL 写入值，不影响引脚状态。PFDRH/PFDRL 的读取 / 写入如表 26.12 所示。

(1) 端口 F 数据寄存器 H（PFDRH）

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PF30 DR	PF29 DR	PF28 DR	PF27 DR	PF26 DR	PF25 DR	PF24 DR	PF23 DR	PF22 DR	PF21 DR	PF20 DR	PF19 DR	PF18 DR	PF17 DR	PF16 DR
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说明
15	—	0	R	保留位 读取值不定，写入值固定为 0。
14	PF30DR	0	R/W	参照表 26.12
13	PF29DR	0	R/W	
12	PF28DR	0	R/W	
11	PF27DR	0	R/W	
10	PF26DR	0	R/W	
9	PF25DR	0	R/W	
8	PF24DR	0	R/W	
7	PF23DR	0	R/W	
6	PF22DR	0	R/W	
5	PF21DR	0	R/W	
4	PF20DR	0	R/W	
3	PF19DR	0	R/W	
2	PF18DR	0	R/W	
1	PF17DR	0	R/W	
0	PF16DR	0	R/W	

(2) 端口 F 数据寄存器 L (PFDRL)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PF15 DR	PF14 DR	PF13 DR	PF12 DR	PF11 DR	PF10 DR	PF9 DR	PF8 DR	PF7 DR	PF6 DR	PF5 DR	PF4 DR	PF3 DR	PF2 DR	PF1 DR	PF0 DR
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说明
15	PF15DR	0	R/W	参照表 26.12
14	PF14DR	0	R/W	
13	PF13DR	0	R/W	
12	PF12DR	0	R/W	
11	PF11DR	0	R/W	
10	PF10DR	0	R/W	
9	PF9DR	0	R/W	
8	PF8DR	0	R/W	
7	PF7DR	0	R/W	
6	PF6DR	0	R/W	
5	PF5DR	0	R/W	
4	PF4DR	0	R/W	
3	PF3DR	0	R/W	
2	PF2DR	0	R/W	
1	PF1DR	0	R/W	
0	PF0DR	0	R/W	

表 26.12 端口 F 数据寄存器 H、L (PFDRH、PFDRL) 的读取 / 写入

- PFDRH 的 bit14 ~ 0 及 PFDRL 的 bit15 ~ 0

PFIORH、L	引脚功能	读取	写入
0	通用输入	引脚状态	可写入 PFIORH、L，不影响引脚状态
	非通用输入	引脚状态	可写入 PFIORH、L，不影响引脚状态
1	通用输入	PFIORH、L 的值	从引脚输出写入值
	非通用输入	PFIORH、L 的值	可写入 PFIORH、L，不影响引脚状态

26.7.3 端口 F 端口寄存器 H、L（PFPRH、PFPLR）

PFPRH、PFPLR 为 16 位只读寄存器，PF30PR ~ PF0PR 位分别对应 PF30/AUDIO_CLK ~ PF0/TCLKA/LCD_DATA0/SSCK0 引脚。与 PFC 的设定无关，PFPRH、PFPLR 总是可读取引脚的值。

(1) 端口 F 端口寄存器 H（PFPRH）

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PF30 PR	PF29 PR	PF28 PR	PF27 PR	PF26 PR	PF25 PR	PF24 PR	PF23 PR	PF22 PR	PF21 PR	PF20 PR	PF19 PR	PF18 PR	PF17 PR	PF16 PR
初始值:	0	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

【注】* 取决于外部引脚的状态。

位	位名称	初始值	R/W	说明
15	—	0	R	保留位 读取值、写入值总是为 0。
14	PF30PR	引脚状态	R	与 PFC 的设定无关，可读取引脚状态，写入无效。
13	PF29PR	引脚状态	R	
12	PF28PR	引脚状态	R	
11	PF27PR	引脚状态	R	
10	PF26PR	引脚状态	R	
9	PF25PR	引脚状态	R	
8	PF24PR	引脚状态	R	
7	PF23PR	引脚状态	R	
6	PF22PR	引脚状态	R	
5	PF21PR	引脚状态	R	
4	PF20PR	引脚状态	R	
3	PF19PR	引脚状态	R	
2	PF18PR	引脚状态	R	
1	PF17PR	引脚状态	R	
0	PF16PR	引脚状态	R	

(2) 端口 F 端口寄存器 L (PFPR L)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PF15 PR	PF14 PR	PF13 PR	PF12 PR	PF11 PR	PF10 PR	PF9 PR	PF8 PR	PF7 PR	PF6 PR	PF5 PR	PF4 PR	PF3 PR	PF2 PR	PF1 PR	PF0 PR
初始值:	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

【注】* 取决于外部引脚的状态。

位	位名称	初始值	R/W	说明
15	PF15PR	引脚状态	R	与 PFC 的设定无关，可读取引脚状态，写入无效。
14	PF14PR	引脚状态	R	
13	PF13PR	引脚状态	R	
12	PF12PR	引脚状态	R	
11	PF11PR	引脚状态	R	
10	PF10PR	引脚状态	R	
9	PF9PR	引脚状态	R	
8	PF8PR	引脚状态	R	
7	PF7PR	引脚状态	R	
6	PF6PR	引脚状态	R	
5	PF5PR	引脚状态	R	
4	PF4PR	引脚状态	R	
3	PF3PR	引脚状态	R	
2	PF2PR	引脚状态	R	
1	PF1PR	引脚状态	R	
0	PF0PR	引脚状态	R	

26.8 使用时的注意事项

由 PFC 选择以下引脚功能时，不可在存取数据寄存器及端口寄存器后读取引脚状态。

- A25～A21、A1、A0（地址总线）
- D31～D16（数据总线）
- BS
- CS7、CS4～CS1、CS5/CE1A、CS6/CE1B、CE2A、CE2B
- RD/WR
- WE3/DQMUU/AH/ICIOWR、WE2/DQMUL/ICIORD、WE1/DQMLU/WE、WE0/DQMLL
- RASU、RASL、CASU、CASL
- CKE
- FRAME
- BREQ
- BACK
- IOIS16
- MRES
- NAF7～0

第 27 章 内部 RAM

本 LSI 内置可高速存取的高速 RAM 及在深度待机模式可保持数据的 RAM，可保存指令或数据。

高速内部 RAM 通过 RAM 允许及写入允许，可禁止存储器的运行及写入。

用于保持的内部 RAM，可按页选择在深度待机模式是否保持数据。

27.1 特点

- 存储器映射
内部 RAM 分配在表 27.1、表 27.2 所示的地址空间。

表 27.1 内部 RAM（高速）地址空间

页	地址
页 0	H'FFF80000 ~ H'FFF83FFF
页 1	H'FFF84000 ~ H'FFF87FFF
页 2	H'FFF88000 ~ H'FFF8BFFF
页 3	H'FFF8C000 ~ H'FFF8FFFF

表 27.2 内部 RAM（用于保持）地址空间

页	地址
页 0	H'FFFF8000 ~ H'FFFF8FFF
页 1	H'FFFF9000 ~ H'FFFF9FFF
页 2	H'FFFFA000 ~ H'FFFFAFFF
页 3	H'FFFFB000 ~ H'FFFFBFFF

- 端口
高速内部 RAM 的各页有 2 个独立的读取/写入端口，可与内部 DMA 总线（ID 总线）、CPU 取指令总线（F 总线）及 CPU 存储器存取总线（M 总线）连接（但是，F 总线仅连接在读取端口）。F 总线及 M 总线用于 CPU 的存取。ID 总线用于 DMAC 的存取。
用于保持的内部 RAM 有 1 个读取/写入端口，连接外围总线。
- 优先顺序
对高速内部 RAM 的相同页从不同总线同时产生存取请求时，按照优先顺序处理存取。优先顺序从高到低为 ID 总线、M 总线、F 总线。

27.2 使用时的注意事项

27.2.1 页竞争

高速内部 RAM 相同页同时产生不同总线的存取请求时，产生页竞争。即使各存取正常结束，但此竞争将降低存储器存取性能。因此，建议使用软件等对策，尽量避免引起这些竞争。例如，将各总线存取不同页，则不产生竞争。

27.2.2 关于 RAME 位、RAMWE 位

对高速内部 RAM，禁止 RAME 位及 RAMWE 位的设定时，在设定 RAME 位及 RAMWE 位之前，必须对各页任意的相同地址执行读取 / 写入。否则，各页最后写入的数据可能无法写入 RAM。

```
//页0
MOV.L #H'FFF80000, R0
MOV.L @R0, R1
MOV.L R1, @R0

//页1
MOV.L #H'FFF84000, R0
MOV.L @R0, R1
MOV.L R1, @R0

//页2
MOV.L #H'FFF88000, R0
MOV.L @R0, R1
MOV.L R1, @R0

//页3
MOV.L #H'FFF8C000, R0
MOV.L @R0, R1
MOV.L R1, @R0
```

图 27.1 执行例

27.2.3 指令配置禁止区域

从用于保持的内部 RAM 的最终地址起 16 字节内（地址 H'FFFFBFF0 ~ H'FFFFBFFF），不得配置指令。如果配置，则 CPU 在地址 H'FFFC000 之后的内部外围模块空间因溢出而取指令，并产生地址错误。

第 28 章 低功耗模式

本 LSI 支持的低功耗模式有：睡眠模式、软件待机模式、深度待机模式和模块待机功能。在低功耗模式，通过停止 CPU、时钟、内部存储器、部分内部外围模块等的功能或关闭电源，可降低功耗。由复位或中断等可解除该模式。

28.1 特点

28.1.1 低功耗模式的种类

低功耗模式有以下模式、功能：

1. 睡眠模式
2. 软件待机模式
3. 深度待机模式
4. 模块待机功能

从程序执行状态转移至各模式的条件、各模式的 CPU 和外围模块等的状态及各模式的解除步骤如表 28.1 所示：

表 28.1 低功耗模式的状态

低功耗模式	转移条件	状 态*1									解除方法
		CPG	CPU	CPU 寄存器	内部RAM (高速) 闪存	内部RAM (保持用)	内部外围 模块	RTC	电源	外部存储器	
睡眠模式	STBCR的STBY位为0时，执行 SLEEP指令	运行	停止	保持	运行	运行	运行	运行*2	运行	必须设置为自动刷新	• 中断 • 手动复位 • 上电复位 • DMA地址错误
软件待机模式	STBCR的STBY位为1、DEEP位为0时，执行SLEEP指令	停止	停止	保持	停止 (保持内容*5*6)	停止 (保持内容*5)	停止	运行*2	运行	必须设置为自刷新	• NMI中断 • IRQ中断 • 手动复位 • 上电复位
深度待机模式	STBCR的STBY位与DEEP位为1时，执行 SLEEP指令	停止	停止	停止	停止 (不保持内容)	停止 (保持内容*3)	停止	运行*2	停止	必须设置为自刷新	• NMI中断*4 • IRQ中断*4 • 手动复位*4 • 上电复位*4
模块待机模式	STBCR2、STBCR3、STBCR4、STBCR5、STBCR6的MSTP位置1	运行	运行	保持	运行	运行	指定模块 停止	停止	运行	必须设置为自动刷新	• MSTP位清0 • 上电复位 (仅限H-UDI、UBC、DMAC)

【注】 *1 引脚状态为保持或高阻抗，详情参阅“附录 A. 引脚状态”。

*2 RCR2 寄存器的 START 位为 1 时，RTC 运行，详情参阅“第 14 章 实时时钟（RTC）”。

*3 如果 RAMKP 寄存器的 RAMKP3 ~ RAMKP0 位置 1，则可在转移至深度待机模式时，保持内部 RAM 对象区域的内容。但是，由上电复位解除深度待机模式时，保存的内容被初始化。

*4 由中断（NMI、IRQ）及复位（手动复位、上电复位）可解除深度待机模式。但由 NMI 中断或 IRQ 中断解除深度待机模式时，不执行中断异常处理，而是执行上电复位异常处理。手动复位时也相同。

*5 由上电复位解除软件待机模式时，保存的内容被初始化。

*6 通过禁止 SYSCR2 寄存器的 RAMWE 位或 SYSCR1 寄存器的 RAME 位，即使由上电复位解除软件待机模式，仍可继续保持内部 RAM（高速）。

28.2 寄存器说明

低功耗模式所使用的寄存器如下：

表 28.2 寄存器结构

寄存器名称	简称	R/W	初始值	地址	存取长度
待机控制寄存器	STBCR	R/W	H'00	H'FFFE0014	8
待机控制寄存器 2	STBCR2	R/W	H'00	H'FFFE0018	8
待机控制寄存器 3	STBCR3	R/W	H'7E	H'FFFE0408	8
待机控制寄存器 4	STBCR4	R/W	H'FF	H'FFFE040C	8
待机控制寄存器 5	STBCR5	R/W	H'FF	H'FFFE0410	8
待机控制寄存器 6	STBCR6	R/W	H'FF	H'FFFE0414	8
系统控制寄存器 1	SYSCR1	R/W	H'FF	H'FFFE0402	8
系统控制寄存器 2	SYSCR2	R/W	H'FF	H'FFFE0404	8
系统控制寄存器 3	SYSCR3	R/W	H'00	H'FFFE0418	8
深度待机控制寄存器	DSCTR	R/W	H'00	H'FFFF2800	8
深度待机控制寄存器 2	DSCTR2	R/W	H'00	H'FFFF2802	8
深度待机解除源选择寄存器	DSSSR	R/W	H'0000	H'FFFF2804	16
深度待机解除源标志寄存器	DSFR	R/W	H'0000	H'FFFF2808	16
用于保持的内部 RAM 调整寄存器	DSRTR	R/W	H'00	H'FFFF280C	8

28.2.1 待机控制寄存器（STBCR）

STBCR 为 8 位可读取 / 写入的寄存器，指定低功耗模式的状态。仅字节存取有效。

【注】 写入本寄存器时，参阅“28.4 使用时的注意事项”。

位：	7	6	5	4	3	2	1	0
	STBY	DEEP	-	-	-	-	-	-
初始值：	0	0	0	0	0	0	0	0
R/W：	R/W	R/W	R	R	R	R	R	R

位	位名称	初始值	R/W	说 明
7	STBY	0	R/W	软件待机、深度待机 指定向软件待机模式、深度待机模式的转移。 0x: 通过执行 SLEEP 指令，向睡眠模式转移 10: 通过执行 SLEEP 指令，向软件待机模式转移 11: 通过执行 SLEEP 指令，向深度待机模式转移
6	DEEP	0	R/W	
5 ~ 0	—	均为 0	R	保留位 读取值、写入值总是为 0。

【符号说明】 x: Don't care

28.2.2 待机控制寄存器 2（STBCR2）

STBCR2 为 8 位可读取 / 写入的寄存器，控制低功耗模式时各模块的运行。仅字节存取有效。

【注】 写入本寄存器时，参阅“28.4 使用时的注意事项”。

位:	7	6	5	4	3	2	1	0
	MSTP 10	MSTP 9	MSTP 8	MSTP 7	-	-	-	-
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R	R

位	位名称	初始值	R/W	说 明
7	MSTP10	0	R/W	模块停止 10 MSTP10 位置 1 时，停止向 H-UDI 提供时钟。 0: H-UDI 运行 1: 停止向 H-UDI 提供时钟
6	MSTP9	0	R/W	模块停止 9 MSTP9 位置 1 时，停止向 UBC 提供时钟。 0: UBC 运行 1: 停止向 UBC 提供时钟
5	MSTP8	0	R/W	模块停止 8 MSTP8 位置 1 时，停止向 DMAC 提供时钟。 0: DMAC 运行 1: 停止向 DMAC 提供时钟
4	MSTP7	0	R/W	模块停止 7 MSTP7 位置 1 时，停止向 FPU 提供时钟。MSTP7 位置 1 后，不可通过写入 0 清除。即：一旦 MSTP7 位置 1，停止向 FPU 提供时钟后，就不能将 MSTP7 位清 0，重新开始向 FPU 提供时钟。 停止向 FPU 提供时钟后重新开始提供时，必须将本 LSI 进行上电复位。 0: FPU 运行 1: 停止向 FPU 提供时钟
3 ~ 0	—	均为 0	R	保留位 读取值、写入值总是为 0。

28.2.3 待机控制寄存器 3（STBCR3）

STBCR3 为 8 位可读取 / 写入的寄存器，控制低功耗模式时各模块的运行。仅字节存取有效。

【注】 写入本寄存器时，参阅“28.4 使用时的注意事项”。

位:	7	6	5	4	3	2	1	0
	HIZ	-	MSTP 35	-	-	MSTP 32	MSTP 31	MSTP 30
初始值:	0	1	1	1	1	1	1	0
R/W:	R/W	R	R/W	R	R	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7	HIZ	0	R/W	端口高阻抗 选择软件待机模式及深度待机模式时保持特定的输出引脚状态或设置为高阻抗。控制哪个引脚，详情参阅“附录 A. 引脚状态”。 WDT 的 WTSCR 的 TME 位为 1 时，不可设定本位。需将输出引脚的状态设置为高阻抗时，必须在 TME 位为 0 时，将 HIZ 位置位。 0: 软件待机模式及深度待机模式时保持引脚状态 1: 软件待机模式及深度待机模式时引脚状态为高阻抗
6	—	1	R	保留位 读取值、写入值总是为 1。
5	MSTP35	1	R/W	模块停止 35 MSTP35 位置 1 时，停止向 MTU2 提供时钟。 0: MTU2 运行 1: 停止向 MTU2 提供时钟
4、3	—	均为 1	R	保留位 读取值、写入值总是为 1。
2	MSTP32	1	R/W	模块停止 32 MSTP32 位置 1 时，停止向 ADC 提供时钟。 0: ADC 运行 1: 停止向 ADC 提供时钟
1	MSTP31	1	R/W	模块停止 31 MSTP31 位置 1 时，停止向 DAC 提供时钟。 0: DAC 运行 1: 停止向 DAC 提供时钟
0	MSTP30	0	R/W	模块停止 30 MSTP30 位置 1 时，停止向 RTC 提供时钟。 0: RTC 运行 1: 停止向 RTC 提供时钟

28.2.4 待机控制寄存器 4 (STBCR4)

STBCR4 为 8 位可读取 / 写入的寄存器，控制低功耗模式时各模块的运行。仅字节存取有效。

【注】 写入本寄存器时，参阅“28.4 使用时的注意事项”。

位:	7	6	5	4	3	2	1	0
	MSTP 47	MSTP 46	MSTP 45	MSTP 44	-	MSTP 42	MSTP 41	MSTP 40
初始值:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7	MSTP47	1	R/W	模块停止 47 MSTP47 位置 1 时，停止向 SCIF0 提供时钟。 0: SCIF0 运行 1: 停止向 SCIF0 提供时钟
6	MSTP46	1	R/W	模块停止 46 MSTP46 位置 1 时，停止向 SCIF1 提供时钟。 0: SCIF1 运行 1: 停止向 SCIF1 提供时钟
5	MSTP45	1	R/W	模块停止 45 MSTP45 位置 1 时，停止向 SCIF2 提供时钟。 0: SCIF2 运行 1: 停止向 SCIF2 提供时钟
4	MSTP44	1	R/W	模块停止 44 MSTP44 位置 1 时，停止向 SCIF3 提供时钟。 0: SCIF3 运行 1: 停止向 SCIF3 提供时钟
3	—	1	R	保留位 读取值、写入值总是为 1。
2	MSTP42	1	R/W	模块停止 42 MSTP42 位置 1 时，停止向 CMT 提供时钟。 0: CMT 运行 1: 停止向 CMT 提供时钟
1	MSTP41	1	R/W	模块停止 41 MSTP41 位置 1 时，停止向 LCDC 提供时钟。 0: LCDC 运行 1: 停止向 LCDC 提供时钟
0	MSTP40	1	R/W	模块停止 40 MSTP40 位置 1 时，停止向 FLCTL 提供时钟。 0: FLCTL 运行 1: 停止向 FLCTL 提供时钟

28.2.5 待机控制寄存器 5 (STBCR5)

STBCR5 为 8 位可读取 / 写入的寄存器，控制低功耗模式时各模块的运行。仅字节存取有效。

【注】 写入本寄存器时，参阅“28.4 使用时的注意事项”。

位:	7	6	5	4	3	2	1	0
	MSTP 57	MSTP 56	MSTP 55	MSTP 54	MSTP 53	MSTP 52	MSTP 51	MSTP 50
初始值:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7	MSTP57	1	R/W	模块停止 57 MSTP57 位置 1 时，停止向 IIC3-0 提供时钟。 0: IIC3-0 运行 1: 停止向 IIC3-0 提供时钟
6	MSTP56	1	R/W	模块停止 56 MSTP56 位置 1 时，停止向 IIC3-1 提供时钟。 0: IIC3-1 运行 1: 停止向 IIC3-1 提供时钟
5	MSTP55	1	R/W	模块停止 55 MSTP55 位置 1 时，停止向 IIC3-2 提供时钟。 0: IIC3-2 运行 1: 停止向 IIC3-2 提供时钟
4	MSTP54	1	R/W	模块停止 54 MSTP54 位置 1 时，停止向 IIC3-3 提供时钟。 0: IIC3-3 运行 1: 停止向 IIC3-3 提供时钟
3	MSTP53	1	R/W	模块停止 53 MSTP53 位置 1 时，停止向 RCAN0 提供时钟。 0: RCAN0 运行 1: 停止向 RCAN0 提供时钟
2	MSTP52	1	R/W	模块停止 52 MSTP52 位置 1 时，停止向 RCAN1 提供时钟。 0: RCAN1 运行 1: 停止向 RCAN1 提供时钟
1	MSTP51	1	R/W	模块停止 51 MSTP51 位置 1 时，停止向 SSU0 提供时钟。 0: SSU0 运行 1: 停止向 SSU0 提供时钟
0	MSTP50	1	R/W	模块停止 50 MSTP50 位置 1 时，停止向 SSU1 提供时钟。 0: SSU1 运行 1: 停止向 SSU1 提供时钟

28.2.6 待机控制寄存器 6（STBCR6）

STBCR6 为 8 位可读取 / 写入的寄存器，控制低功耗模式时各模块的运行。仅字节存取有效。

【注】 写入本寄存器时，参阅“28.4 使用时的注意事项”

位:	7	6	5	4	3	2	1	0
	MSTP 67	MSTP 66	MSTP 65	MSTP 64	-	-	-	MSTP 60
初始值:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R	R	R	R/W

位	位名称	初始值	R/W	说 明
7	MSTP67	1	R/W	模块停止 67 MSTP67 位置 1 时，停止向 SSI0 提供时钟。 0: SSI0 运行 1: 停止向 SSI0 提供时钟
6	MSTP66	1	R/W	模块停止 66 MSTP66 位置 1 时，停止向 SSI1 提供时钟。 0: SSI1 运行 1: 停止向 SSI1 提供时钟
5	MSTP65	1	R/W	模块停止 65 MSTP65 位置 1 时，停止向 SSI2 提供时钟。 0: SSI2 运行 1: 停止向 SSI2 提供时钟
4	MSTP64	1	R/W	模块停止 64 MSTP64 位置 1 时，停止向 SSI3 提供时钟。 0: SSI3 运行 1: 停止向 SSI3 提供时钟
3 ~ 1	—	均为 1	R	保留位 读取值、写入值总是为 1。
0	MSTP60	1	R/W	模块停止 60 MSTP60 位置 1 时，停止向 USB 提供时钟。 0: USB 运行 1: 停止向 USB 提供时钟

28.2.7 系统控制寄存器 1（SYSCR1）

SYSCR1 为 8 位可读取 / 写入的寄存器，设定允许 / 禁止存取内部 RAM（高速）。仅字节存取有效。

RAME 位置 1 时内部 RAM（高速）有效；清 0 时，不可存取 RAM（高速）。从内部 RAM（高速）读取或取指令时读取值不定，写入 RAM（高速）被忽略。初始值为 1。

禁止设定 RAME 位时，设定前必须对各页读取 / 写入任意一个相同地址。否则，最后写入各页的数据可能无法写入内部 RAM（高速）。不可在紧随 SYSCR1 的写入指令后设置存取内部 RAM（高速）的指令，否则，无法保证正常存取。

将本位置 1 使内部 RAM（高速）有效时，必须设置 SYSCR1 的读取指令。如果在紧随 SYSCR1 的写入指令后设置内部 RAM（高速）存取指令，则无法保证正常存取。

【注】 写入本寄存器时，参阅“28.4 使用时的注意事项”。

位:	7	6	5	4	3	2	1	0
	-	-	-	-	RAME3	RAME2	RAME1	RAME0
初始值:	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7 ~ 4	—	均为 1	R	保留位 读取值、写入值总是为 1。
3	RAME3	1	R/W	RAM 允许 3（对象：内部 RAM（高速）的页 3*） 0：内部 RAM（高速）无效 1：内部 RAM（高速）有效
2	RAME2	1	R/W	RAM 允许 2（对象：内部 RAM（高速）的页 2*） 0：内部 RAM（高速）无效 1：内部 RAM（高速）有效
1	RAME1	1	R/W	RAM 允许 1（对象：内部 RAM（高速）的页 1*） 0：内部 RAM（高速）无效 1：内部 RAM（高速）有效
0	RAME0	1	R/W	RAM 允许 0（对象：内部 RAM（高速）的页 0*） 0：内部 RAM（高速）无效 1：内部 RAM（高速）有效

【注】 * 有关各页的地址，详情参阅“第 27 章 内部 RAM”。

28.2.8 系统控制寄存器 2（SYSCR2）

SYSCR2 为 8 位可读取 / 写入的寄存器，设定允许 / 禁止写入内部 RAM（高速）。仅字节存取有效。

RAMWE 位置 1 时，写入内部 RAM（高速）有效；清 0 时，不可写入内部 RAM（高速）。此时，忽略向内部 RAM（高速）的写入。初始值为 1。

禁止设定 RAMWE 位时，设定前必须对各页读取 / 写入任意一个相同地址。否则，最后写入的数据可能无法写入内部 RAM（高速）。不可在紧随 SYSCR2 的写入指令后设置存取内部 RAM（高速）的指令，否则，无法保证正常存取。

将本位置 1 使写入内部 RAM（高速）有效，必须设置 SYSCR2 的读取指令。如果在紧随 SYSCR2 的写入指令后设置存取内部 RAM（高速）的指令，则无法保证正常存取。

【注】 写入本寄存器时，参阅“28.4 使用时的注意事项”。

位:	7	6	5	4	3	2	1	0
	-	-	-	-	RAM WE3	RAM WE2	RAM WE1	RAM WE0
初始值:	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7 ~ 4	—	均为 1	R	保留位 读取值、写入值总是为 1。
3	RAME3	1	R/W	RAM 写入允许 3（对象：内部 RAM（高速）的页 3*） 0：内部 RAM（高速）写入无效 1：内部 RAM（高速）写入有效
2	RAME2	1	R/W	RAM 写入允许 2（对象：内部 RAM（高速）的页 2*） 0：内部 RAM（高速）写入无效 1：内部 RAM（高速）写入有效
1	RAME1	1	R/W	RAM 写入允许 1（对象：内部 RAM（高速）的页 1*） 0：内部 RAM（高速）写入无效 1：内部 RAM（高速）写入有效
0	RAME0	1	R/W	RAM 写入允许 0（对象：内部 RAM（高速）的页 0*） 0：内部 RAM（高速）写入无效 1：内部 RAM（高速）写入有效

【注】 * 有关各页的地址，详情参阅“第 27 章 内部 RAM”。

28.2.9 系统控制寄存器 3（SYSCR3）

SYSCR3 为 8 位可读取 / 写入的寄存器，控制 SSI0 ~ 3 的软件复位。仅字节存取有效。

【注】 写入本寄存器时，参阅“28.4 使用时的注意事项”。

位:	7	6	5	4	3	2	1	0
	AXT ALE	-	-	-	SSI3 SRST	SSI2 SRST	SSI1 SRST	SSI0 SRST
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7	AXTALE	0	R/W	音频用晶体谐振器允许 指定音频用晶体谐振器功能有效 / 无效。 0: 音频用晶体谐振器功能有效 1: 音频用晶体谐振器功能无效
6 ~ 4	—	均为 0	R	保留位 读取值、写入值总是为 0。
3	SSI3SRST	0	R/W	SSI3 软件复位 由软件控制 SSI3 的复位。 0: 解除 SSI3 复位 1: SSI3 为复位状态
2	SSI2SRST	0	R/W	SSI2 软件复位 由软件控制 SSI2 的复位。 0: 解除 SSI2 的复位 1: SSI2 为复位状态
1	SSI1SRST	0	R/W	SSI1 软件复位 由软件控制 SSI1 的复位。 0: 解除 SSI1 的复位 1: SSI1 为复位状态
0	SSI0SRST	0	R/W	SSI0 软件复位 由软件控制 SSI0 的复位。 0: 解除 SSI0 的复位 1: SSI0 为复位状态

28.2.10 深度待机控制寄存器（DSCTR）

DSCTR 为 8 位可读取 / 写入的寄存器，设定深度待机模式时是否保持相应内部 RAM（保持用）区域的内容。仅字节存取有效。

RRAMKP3 ~ 0 位置 1 时，在深度待机模式时保持相应内部 RAM（保持用）区域的内容，清 0 时不保持。

【注】 写入本寄存器时，参阅“28.4 使用时的注意事项”。

位:	7	6	5	4	3	2	1	0
	-	-	-	-	RRAM KP3	RRAM KP2	RRAM KP1	RRAM KP0
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7 ~ 4	—	均为 0	R	保留位 读取值、写入值总是为 0。
3	RRAMKP3	0	R/W	保持内部 RAM 的保持区域 3（对象：内部 RAM（保持用）的页 3*） 0：深度待机模式时，不保持内部 RAM（保持用） 1：深度待机模式时，保持内部 RAM（保持用）
2	RRAMKP2	0	R/W	保持内部 RAM 保持区域 2（对象：内部 RAM（保持用）的页 2*） 0：深度待机模式时，不保持内部 RAM（保持用） 1：深度待机模式时，保持内部 RAM（保持用）
1	RRAMKP1	0	R/W	保持内部 RAM 保持区域 1（对象：内部 RAM（保持用）的页 1*） 0：深度待机模式时，不保持内部 RAM（保持用） 1：深度待机模式时，保持内部 RAM（保持用）
0	RRAMKP0	0	R/W	保持内部 RAM 保持区域 0（对象：内部 RAM（保持用）的页 0*） 0：深度待机模式时，不保持内部 RAM（保持用） 1：深度待机模式时，保持内部 RAM（保持用）

【注】 * 有关各页的地址，详情参阅“第 27 章 内部 RAM”。

28.2.11 深度待机控制寄存器 2（DSCTR2）

DSCTR2 为 8 位可读取 / 写入的寄存器，控制从深度待机模式返回时，外部总线控制引脚的状态和启动方法。仅字节存取有效。

【注】 写入本寄存器时，参阅“28.4 使用时的注意事项”。

位:	7	6	5	4	3	2	1	0
	CS0 KEEPE	RAM BOOT	-	-	-	-	-	-
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R	R	R

位	位名称	初始值	R/W	说 明
7	CS0KEEPE	0	R/W	外部总线控制引脚状态保持 0: 从深度待机模式返回时，不保持外部总线控制引脚的状态 1: 从深度待机模式返回时，保持外部总线控制引脚的状态
6	RAMBOOT	0	R/W	选择从深度待机模式返回后的启动方法 在由 MRES、NMI、IRQ 解除深度待机模式时的上电复位异常处理中，从以下地址取出程序计数器（PC）和堆栈指针（SP）： 0: 从地址 H'00000000、H'00000004 取出 1: 从地址 H'FFFF8000、H'FFFF8004 取出
5 ~ 0	—	均为 0	R	保留位 读取值、写入值总是为 0。

28.2.12 深度待机解除源选择寄存器 (DSSSR)

DSSSR 为 16 位可读取 / 写入的寄存器，由选择通过哪种中断解除深度待机模式的位构成。仅在引脚分配至 IRQ0 ~ 7, PE4 ~ PE11 才有效。仅字节存取有效。

【注】 写入本寄存器时，参阅“28.4 使用时的注意事项”。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	MRES	IRQ7	IRQ6	IRQ5	IRQ4	IRQ3	IRQ2	IRQ1	IRQ0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
15 ~ 9	—	均为 0	R	保留位 读取值、写入值总是为 0。
8	MRES	0	R/W	通过手动复位解除 0: 不通过手动复位解除 1: 通过手动复位解除
7	IRQ7	0	R/W	通过 IRQ7 解除 (PE11) 0: 不通过 IRQ7 解除 1: 通过 IRQ7 解除
6	IRQ6	0	R/W	通过 IRQ6 解除 (PE10) 0: 不通过 IRQ6 解除 1: 通过 IRQ6 解除
5	IRQ5	0	R/W	通过 IRQ5 解除 (PE9) 0: 不通过 IRQ5 解除 1: 通过 IRQ5 解除
4	IRQ4	0	R/W	通过 IRQ4 解除 (PE8) 0: 不通过 IRQ4 解除 1: 通过 IRQ4 解除
3	IRQ3	0	R/W	通过 IRQ3 解除 (PE7) 0: 不通过 IRQ3 解除 1: 通过 IRQ3 解除
2	IRQ2	0	R/W	通过 IRQ2 解除 (PE6) 0: 不通过 IRQ2 解除 1: 通过 IRQ2 解除
1	IRQ1	0	R/W	通过 IRQ1 解除 (PE5) 0: 不通过 IRQ1 解除 1: 通过 IRQ1 解除
0	IRQ0	0	R/W	通过 IRQ0 解除 (PE4) 0: 不通过 IRQ0 解除 1: 通过 IRQ0 解除

28.2.13 深度待机解除源标志寄存器（DSFR）

DSFR 为 16 位可读取 / 写入的寄存器，由确认通过哪种中断解除深度待机模式的标志及解除深度待机模式后解除引脚状态的位构成。由中断（NMI、IRQ）及手动复位解除深度待机模式时，DSFR 处理上电复位异常，但本寄存器保持此之前的值；由上电复位解除深度待机模式时，本寄存器初始化为 H'0000。仅字节存取有效。

转移至深度待机模式前，需清除所有的标志。

【注】 写入本寄存器时，参阅“28.4 使用时的注意事项”。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IO KEEP	-	-	-	-	-	MRESF	NMIF	IRQ7F	IRQ6F	IRQ5F	IRQ4F	IRQ3F	IRQ2F	IRQ1F	IRQ0F
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/(W)*	R	R	R	R	R	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*

【注】* 为了清除标志，仅可在读取1后写入0。

位	位名称	初始值	R/W	说 明
15	IOKEEP	0	R/(W)*	引脚状态保持解除 本位为解除深度待机模式时，解除引脚状态保持的位。 0: 不保持引脚状态 [清除条件] • 读取 1 后写入 0 1: 保持引脚状态 [置位条件] • 转移至深度待机模式时
14 ~ 10	—	均为 0	R	保留位 读取值、写入值总是为 0。
9	MRESF	0	R/(W)*	MRES 标志 0: 在 MRES 引脚无中断 1: 在 MRES 引脚有中断
8	NMIF	0	R/(W)*	NMI 标志 0: 在 NMI 引脚无中断 1: 在 NMI 引脚有中断
7	IRQ7F	0	R/(W)*	IRQ7 标志 0: 在 IRQ7（PE11）引脚无中断 1: 在 IRQ7（PE11）引脚有中断
6	IRQ6F	0	R/(W)*	IRQ6 标志 0: 在 IRQ6（PE10）引脚无中断 1: 在 IRQ6（PE10）引脚有中断
5	IRQ5F	0	R/(W)*	IRQ5 标志 0: 在 IRQ5（PE9）引脚无中断 1: 在 IRQ5（PE9）引脚有中断
4	IRQ4F	0	R/(W)*	IRQ4 标志 0: 在 IRQ4（PE8）引脚无中断 1: 在 IRQ4（PE8）引脚有中断

位	位名称	初始值	R/W	说 明
3	IRQ3F	0	R/(W)*	IRQ3 标志 0: 在 IRQ3 (PE7) 引脚无中断 1: 在 IRQ3 (PE7) 引脚有中断
2	IRQ2F	0	R/(W)*	IRQ2 标志 0: 在 IRQ2 (PE6) 引脚无中断 1: 在 IRQ2 (PE6) 引脚有中断
1	IRQ1F	0	R/(W)*	IRQ1 标志 0: 在 IRQ1 (PE5) 引脚无中断 1: 在 IRQ1 (PE5) 引脚有中断
0	IRQ0F	0	R/(W)*	IRQ0 标志 0: 在 IRQ0 (PE4) 引脚无中断 1: 在 IRQ0 (PE4) 引脚有中断

【注】 * 为了清除标志，仅可在读取 1 后写入 0。

28.2.14 保持用内部 RAM 调整寄存器 (DSRTR)

DSRTR 为 8 位可读取 / 写入的寄存器。

本寄存器在深度待机模式调整保持用内部 RAM 的待机电流。仅字节存取有效。

在深度待机模式保持内部 RAM (保持用) 的数据时，必须向本寄存器写入 H'09 后，转移至深度待机模式。

RES 引脚有效或解除深度待机模式后，初始化本寄存器。

【注】 写入本寄存器时，参阅“28.4 使用时的注意事项”。

位:	7	6	5	4	3	2	1	0
	-	TRMD[6:0]						
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名称	初始值	R/W	说 明
7	—	0	R	保留位 读取值、写入值总是为 0。
6 ~ 0	TRMD[6:0]	均为 0	R/W	保持用内部 RAM 调整数据 本位可在深度待机模式时调整保持用内部 RAM 的待机电流。

28.3 运行说明

28.3.1 睡眠模式

(1) 向睡眠模式转移

在 STBCR 的 STBY 位为 0 的状态执行 SLEEP 指令时，从程序执行状态转移至睡眠模式。执行 SLEEP 指令后，CPU 停止运行，但保持 CPU 寄存器的内容。内部外围模块继续运行。在时钟模式 0、1、3，可设定从 CKIO 引脚连续输出时钟。

(2) 解除睡眠模式

由中断（NMI、IRQ、内部外围）、DMA 地址错误及复位（手动复位、上电复位）解除睡眠模式。

- 由中断解除

产生 NMI、IRQ 及内部外围中断时，解除睡眠模式、执行中断异常处理。产生中断的优先级小于等于 CPU 状态寄存器（SR）设定的中断屏蔽级时，或在模块由内部外围模块禁止中断时，不接受中断请求、不解除睡眠模式。

- 由 DMA 地址错误解除

产生 DMA 地址错误时，解除睡眠模式，并执行 DMA 地址错误异常处理。

- 由复位解除

由上电复位及手动复位解除睡眠模式。

28.3.2 软件待机模式

(1) 向软件待机模式转移

STBCR 的 STBY 位为 1、DEEP 位为 0 时执行 SLEEP 指令，可从程序执行状态转移至软件待机模式。在软件待机模式时，CPU、时钟和内部外围模块均停止运行。在时钟模式 0、1、3 时，停止从 CKIO 引脚输出时钟。

保持 CPU 及高速缓存的寄存器内容。部分内部外围模块的寄存器被初始化。有关软件待机模式时外围模块的寄存器状态，详情参阅“30.3 各运行模式的寄存器状态一览表”。

CPU 在 1 个周期结束对 STBCR 的写入后，执行下一条指令处理。但实际的写入超过 1 个周期，为了将 CPU 对 STBCR 写入的值准确反映在 SLEEP 指令，必须读取 STBCR 后执行 SLEEP 指令。

向软件待机模式转移的步骤如下：

1. 将 WDT 的定时器控制寄存器（WTCSR）的 TME 位清 0，停止 WDT 运行。
2. 将 WDT 的定时器计数器（WTCNT）置 0，设定 WTCSR 寄存器的 CKS[2:0] 位的值大于等于振荡稳定时间。
3. 将 STBCR 寄存器的 STBY 位设定为 1、DEEP 位设定为 0 后，读取 STBCR 寄存器，再执行 SLEEP 指令。

(2) 解除软件待机模式

由中断（NMI、IRQ）及复位（手动复位、上电复位）解除软件待机模式。在时钟模式 0、1、3 时，从 CKIO 引脚输出时钟。

- 由中断解除

如果检测出 NMI 引脚的下降沿或上升沿（由中断控制器（INTC）的中断控制寄存器 0（ICR0）的 NMI 边沿选择位（NMIE）选择）、IRQ 引脚（IRQ7～IRQ0）的下降沿或上升沿（由中断控制器（INTC）的中断控制寄存器 1（ICR1）的 IRQn 检测选择位（IRQn1S～IRQn0S）选择），则开始时钟振荡。本时钟仅提供计算振荡稳定时间的振荡稳定计数器（WDT）。

转移至软件待机模式前，如果经过 WDT 的看门狗定时器控制/状态寄存器（WTCSR）的时钟选择位（CKS[2:0]）设定的时间，则产生 WDT 上溢。根据产生上溢判断时钟稳定，向本 LSI 整体提供时钟。据此解除软件待机模式，并执行 NMI 中断异常处理（IRRQ 时，为 IRQ 中断异常处理）。

由 NMI 中断或 IRQ 中断解除软件待机模式时，设定 CKS[2:0] 位，使 WDT 的上溢周期大于等于振荡稳定时间。

从检测出中断到解除软件待机模式期间，CKIO 引脚时钟输出的相位可能不稳定。如果通过在下降沿设定的 NMI 引脚解除软件待机模式，进入软件待机模式时（时钟停止时）的 NMI 引脚必须为高电平，而且解除软件待机模式时（启动振荡稳定后的时钟时）的 NMI 引脚必须为低电平；如果通过在上升沿设定的 NMI 引脚解除软件待机模式，进入软件待机模式时（时钟停止时）的 NMI 引脚必须为低电平，且解除软件待机模式时（启动振荡稳定后的时钟时）的 NMI 引脚必须为高电平（IRQ 引脚时也相同）。

- 由复位解除

RES 引脚为低电平时，解除软件待机模式，并转移至上电复位状态。之后，如果 $\overline{\text{RES}}$ 引脚为高电平，则开始上电复位异常处理。

MRES 引脚为低电平时，解除软件待机模式，并转移至手动复位状态。之后，如果 $\overline{\text{MRES}}$ 引脚为高电平，则开始手动复位异常处理。

时钟振荡稳定前， $\overline{\text{RES}}$ 引脚或 $\overline{\text{MRES}}$ 引脚必须保持低电平。

在时钟模式 0、1、3，可继续从 CKIO 引脚输出内部时钟。

(3) 软件待机模式转移时的注意事项

由中断（NMI、IRQ）及复位（手动复位、上电复位）解除软件待机模式，如果同时产生 SLEEP 指令和除 NMI、IRQ 之外的中断，则可能接受中断，解除软件待机模式。

向软件待机模式转移时，必须设定为不接受中断，再执行 SLEEP 指令。

28.3.3 软件待机模式的应用例

本节说明在 NMI 信号的下降沿转移至软件待机模式，以及在 NMI 信号的上升沿解除的例子，本例的时序如图 28.1 所示。

中断控制寄存器 0（ICR0）的 NMI 边沿选择位（NMIE）置 0（下降沿检测）时，如果 NMI 引脚从高电平变为低电平，则接受 NMI 中断。通过 NMI 异常服务程序将 NMIE 位置 1（上升沿检测），STBCR 的 STBY 位为 1、DEEP 位为 0 时，执行 SLEEP 指令，转移至软件待机模式。之后，如果 NMI 引脚从低电平变为高电平，则解除软件待机模式。

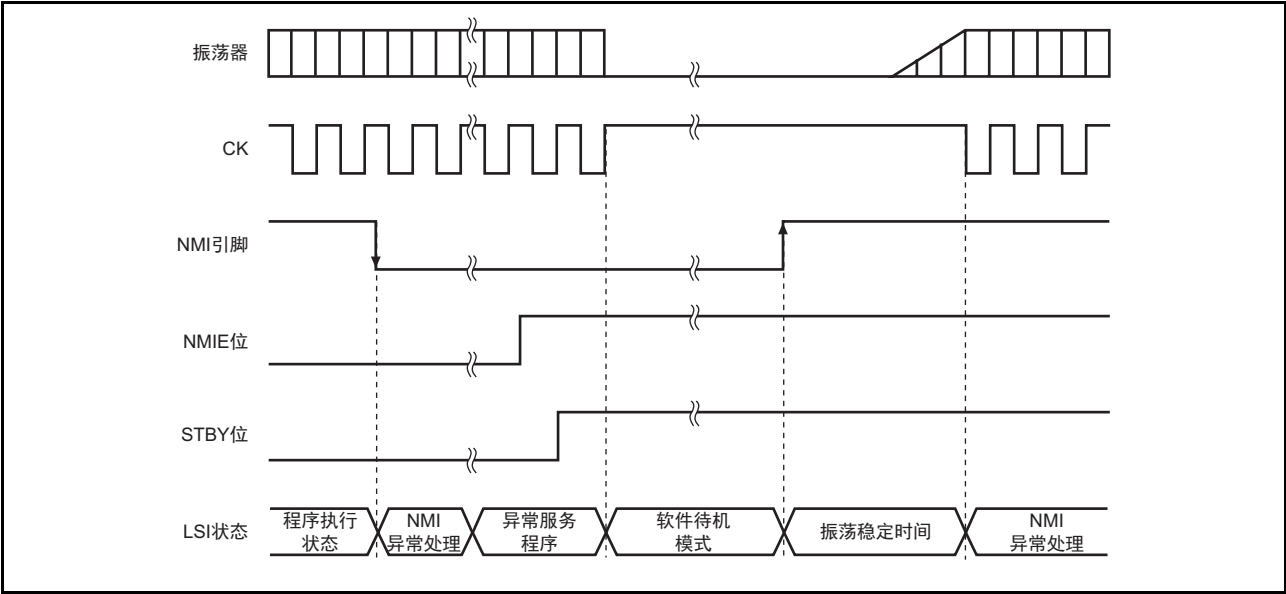


图 28.1 软件待机模式的 NMI 时序（应用例）

28.3.4 深度待机模式

(1) 向深度待机模式转移

在 STBCR 的 STBY 位和 DEEP 位置 1 的状态下，执行 SLEEP 指令时，从程序执行状态转移至深度待机模式。在深度待机模式，CPU、时钟及内部外围模块均停止运行。因 DSCTR 寄存器的 RRAMKP3 ~ RRAMKP0 位的设定而保持的内部 RAM（保持用）区域、RTC 之外的电源也关闭，这样可大幅度降低功耗。因此，也不保持 CPU 高速缓存寄存器内容及内部外围模块的寄存器的值，但引脚状态可保持转移至深度待机模式前的状态。

CPU 在 1 个周期对 DSCTR 的写入结束后，执行下一条处理指令。但实际的写入超过 1 个周期，因此，为了将 CPU 对 DSCTR 写入的值准确反映在 SLEEP 指令，必须在读取 DSCTR 后执行 SLEEP 指令。

向深度待机模式转移的步骤如下，流程如图 28.2 所示：

1. 深度待机模式时在内部 RAM（保持用）区域保持数据时，并在 DSRTR 寄存器设定 H'09。
2. 对需保持的内部 RAM（保持用）区域设定 DSCTR 寄存器的 RRAMKP3 ~ RRAMKP0 位，并将需保持的程序传送至设定的内部 RAM（保持用）区域。
3. 由中断解除深度待机模式时，设定由哪个引脚解除并设定 DSSSR 的相应位。同时设定解除引脚的输入信号检测模式（由中断控制器（INTC）的中断控制寄存器 0、1（ICR0、ICR1）设定）。在深度待机模式时，仅上升沿或下降沿的设定有效（设定为 IRQ 的低电平及双边沿检测时，不可解除）。
4. 对保持内部 RAM（保持用）的各页执行任意相同地址的读取/写入。否则，最后写入的数据可能无法写入内部 RAM（保持用），之后，向内部 RAM（保持用）写入时，最后写入内部 RAM（保持用）后必须执行本处理。
5. 将 STBCR 寄存器的 STBY 位和 DEEP 位设定为 1。
6. 清除 DSFR 寄存器的标志后，读取 DSFR 寄存器之后，执行 SLEEP 指令。

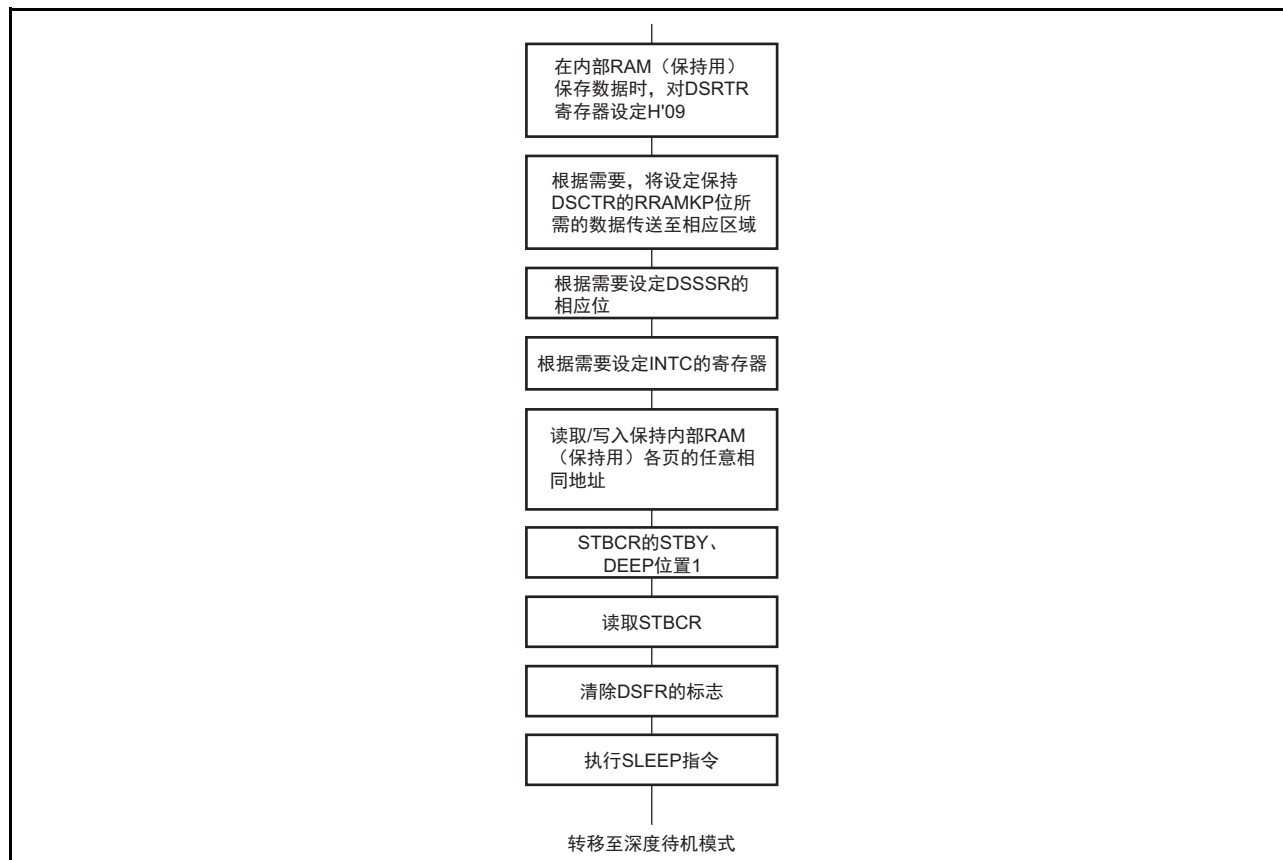


图 28.2 深度待机模式转移流程

(2) 深度待机模式的解除

由中断（NMI、分配至 PE11 ~ PE4 的 IRQ）及复位（手动复位、上电复位）解除深度待机模式。由 NMI 中断或 IRQ 中断解除时，不执行中断异常处理，而执行上电复位异常处理，手动复位时也相同。解除深度待机模式的流程如图 28.3 所示。

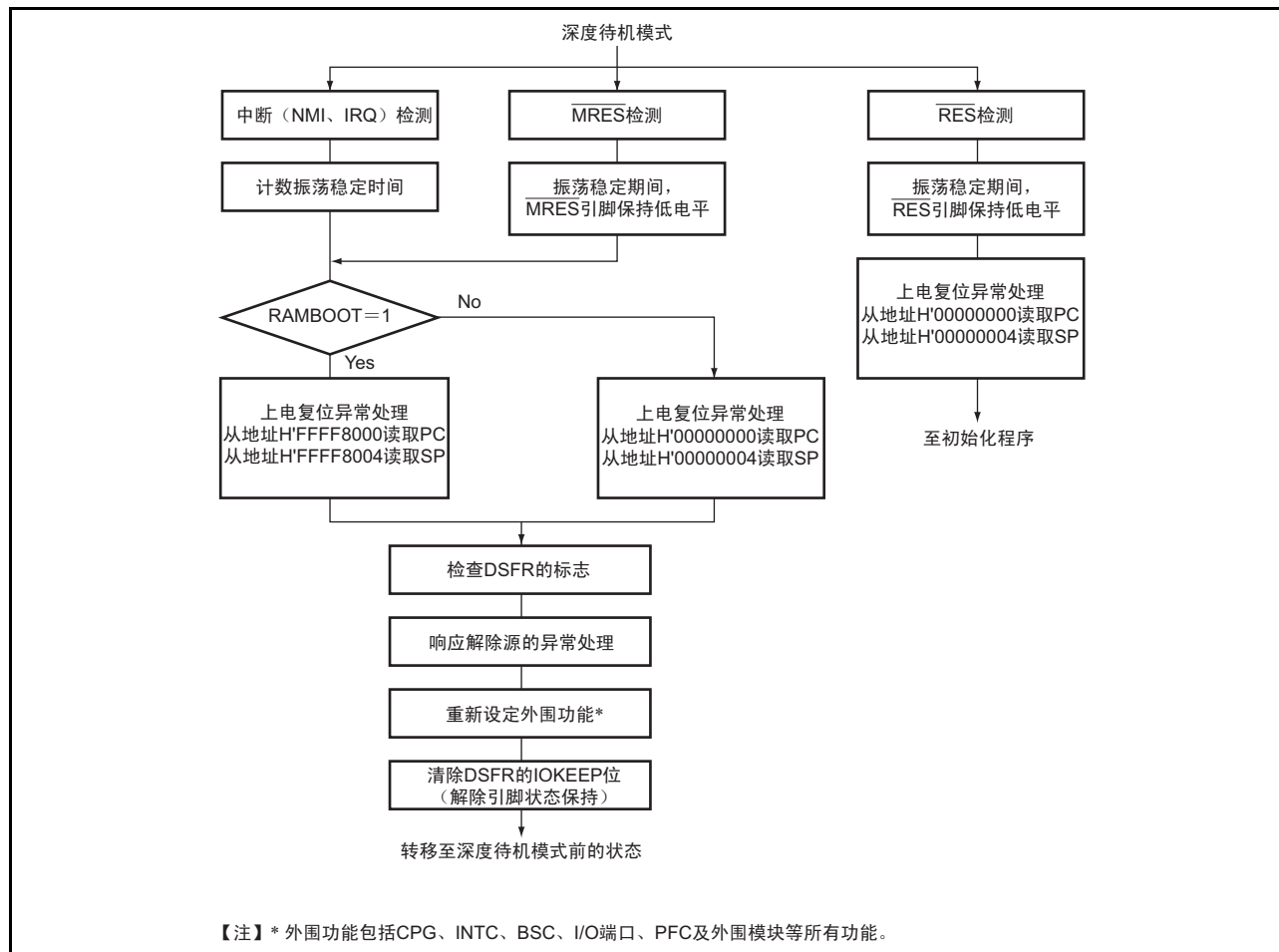


图 28.3 深度待机模式解除流程

- 由中断解除

如果检测出 NMI 引脚的下降沿或上升沿（由中断控制器（INTC）的中断控制寄存器 0（ICR0）的 NMI 边沿选择位（NMIE）选择）、IRQ 引脚（IRQ7～IRQ0 被分配至 PE11～PE4）的下降沿或上升沿（由中断控制器（INTC）的中断控制寄存器 1（ICR1）的 IRQn 检测选择位（IRQn1S～IRQn0S）选择），则在电源稳定等待时间后开始时钟振荡。经过振荡稳定时间后，解除深度待机模式，执行上电复位异常处理。

从检测出中断至解除深度待机模式期间，CKIO 引脚时钟输出的相位可能不稳定。如果通过在下降沿设定的 NMI 引脚解除深度待机模式，进入深度待机模式时（时钟停止时）的 NMI 引脚必须为高电平，且解除深度待机模式时（启动振荡稳定后的时钟时）的 NMI 引脚必须为低电平；如果通过在上升沿设定的 NMI 引脚解除深度待机模式，进入深度待机模式时（时钟停止时）的 NMI 引脚必须为低电平，且解除深度待机模式时（启动振荡稳定后的时钟时）的 NMI 引脚必须为高电平（IRQ 引脚时也相同）。

- 由复位解除

$\overline{\text{RES}}$ 引脚为低电平时，解除深度待机模式，并转移至上电复位状态。之后，如果 $\overline{\text{RES}}$ 引脚为高电平，则开始上电复位异常处理。在时钟模式 0、1、3 时，如果 $\overline{\text{RES}}$ 引脚为低电平，就从 CKIO 引脚开始输出内部时钟。

$\overline{\text{MRES}}$ 引脚为低电平时，解除深度待机模式，并转移至上电复位。之后，如果 $\overline{\text{MRES}}$ 引脚为高电平，则开始上电复位异常处理。在时钟模式 0、1、3 时，如果 $\overline{\text{MRES}}$ 引脚为高电平，就从 CKIO 引脚开始输出内部时钟。

时钟振荡稳定前， $\overline{\text{RES}}$ 引脚或 $\overline{\text{MRES}}$ 引脚必须保持低电平状态。

(3) 解除深度待机模式后的运行

通过设定 DSCTR2 寄存器的 RAMBOOT 位，可选择启动外部总线或内部 RAM（保持用）。通过设定 CS0KEEPE 位，即使解除深度待机模式后，仍可保持外部总线控制引脚的状态。由各位解除深度待机模式后的引脚状态如表 28.3 所示，外部总线控制引脚一览表如表 28.4 所示。

表 28.3 通过 DSCTR2 寄存器设定解除深度待机后的引脚状态和启动方法

CS0KEEPE 位	RAMBOOT 位	启动方法	解除深度待机后的引脚状态
0	0	外部总线	不保持外部总线控制引脚。 清除 IOKEEP 位时，解除其他引脚的保持状态。
0	1	内部 RAM （保持用）	不保持外部总线控制引脚。 解除深度待机模式后，解除外部总线控制引脚的保持状态。 清除 IOKEEP 位时，解除其他引脚的保持状态。
1	0	—	禁止设定
1	1	内部 RAM （保持用）	保持外部总线控制引脚。 清除 IOKEEP 位时，解除外部总线控制引脚及其他引脚的保持状态。

表 28.4 外部总线控制引脚一览表

运行模式 0（启动外部 16 位总线）	运行模式 1（启动外部 32 位总线）
$A[20:1]$ $D[15:0]$ $\overline{CS0}$ 、 $\overline{RD}$ 、 $\overline{CKIO}$	$A[20:2]$ $D[31:0]$ $\overline{CS0}$ 、 $\overline{RD}$ 、 $\overline{CKIO}$

由中断（NMI、IRQ）及手动复位解除深度待机模式时，可由深度待机解除源标志寄存器（DSFR）确认通过哪种中断解除。

转移至深度待机模式时，引脚保持之前的状态。但在外部总线启动模式，解除深度待机模式后，可解除保存的外部总线控制引脚的状态，并取出程序。解除深度待机模式后，读取 DSFR 寄存器的 IOKEEP 位的 1 后到写入 0 前，可继续保持其他引脚的状态。在内部 RAM（保持用）启动模式，解除深度待机模式后，读取 DSFR 寄存器的 IOKEEP 位的 1 后到写入 0 前，可继续保持外部总线控制引脚及其他引脚的状态。返回深度待机模式转移前的状态需重新设定外围功能。外围功能包含 CPG、INTC、BSC、I/O 端口、PFC、外围模块等所有功能。重新设定后，通过读取 IOKEEP 位的 1 后写入 0，可解除引脚保持状态，并成为转移至深度待机模式前的状态。

(4) 深度待机模式转移时的注意事项

设定深度待机模式后，除深度待机解除源选择寄存器设定的中断之外，其他中断均被屏蔽。仅接受深度待机解除源选择寄存器设定的中断，但设定并输入多个解除源时，多个解除源标志置位。

深度待机模式转移的 SLEEP 指令和 NMI、IRQ 的中断及手动复位同时产生时，可能接受中断，并解除深度待机模式。

28.3.5 模块待机功能

(1) 向模块待机功能转移

将待机控制寄存器的各 MSTP 位置 1，可停止向相应的内部外围模块提供时钟。使用本功能可降低程序执行状态及睡眠模式时的功耗。转移前必须将该内部外围模块设置为深度待机状态后，再设置为模块待机状态。另外，不可对模块待机状态的模块执行寄存器存取。

有关寄存器的状态，详情参阅“30.3 各运行模式的寄存器状态一览表”。

(2) 模块待机功能的解除

通过将各 MSTP 位清 0 或上电复位（仅限 RTC、H-UDI、UBC、DMAC），来解除模块待机功能。将各 MSTP 位清 0 解除模块待机功能时，必须确认读取相应位后清 0。

28.4 使用时的注意事项

28.4.1 写入寄存器时的注意事项

写入与低功耗模式相关的寄存器时，必须注意以下事项。

从 CPU 写入与低功耗模式相关的寄存器时，CPU 执行写入指令后，不等待实际寄存器的写入结束即执行后续指令。

执行后续指令时，如果需通过写入寄存器反映更改，必须在寄存器写入指令与后续指令之间虚读相同寄存器。

28.4.2 有关深度待机控制寄存器 2（DSCTR2）的注意事项

由 $\overline{\text{RES}}$ 引脚解除上电复位后，如果在深度待机控制寄存器 2（DSCTR2）的 bit7（CS0KEEPE）及 bit6（RAMBOOT）写入 1 时转移至深度待机模式，之后则不可写入 0。为了可写入 0，需设定 $\overline{\text{RES}}$ 引脚为低电平有效。

28.4.3 上电复位异常处理相关注意事项

- 由 $\overline{\text{RES}}$ 引脚解除上电复位后，如果在深度待机控制寄存器 2（DSCTR2）的 bit6（RAMBOOT）写入 1 时转移至深度待机模式并解除后，由 $\overline{\text{RES}}$ 重新执行上电复位前产生 WDT 引起的上电复位/H-UDI 复位时，上述复位异常处理的运行如表 28.5 所示。符合上述运行时，必须在相应的保持 RAM 区域保持 PC 及 SP。

表 28.5 复位异常处理

取出程序计数器（PC）的地址	取出堆栈指针（SP）的地址
H'FFFF8000	H'FFFF8004

- 由 $\overline{\text{RES}}$ 引脚解除上电复位后，转移至深度待机模式并解除，再次由 $\overline{\text{RES}}$ 引脚执行上电复位前，可能产生 WDT 引起的上电复位或 H-UDI 复位，此时，解除深度待机模式后，必须在深度待机解除源标志寄存器（DSFR）的 bit15（IOKEEP）及 bit9～0 全部清 0 后（为 1 时必须读取 1 后写入 0），设定 WDT 或 H-UDI。

在 IOKEEP 位不为 0 的状态下设定 WDT 及 H-UDI，由 $\overline{\text{RES}}$ 引脚执行上电复位前，产生 WDT 引起的上电复位/H-UDI 复位时，表 28.4 所示之外的在深度待机模式保持的所有引脚均继续保持。另外，如果深度待机控制寄存器 2（DSCTR2）的 bit7（CS0KEEPE）设定为 1，表 28.4 所示的外部总线控制引脚也继续保持。

在 bit9～0 的所有标志均不为 0 的状态下设定 WDT 及 H-UDI，由 $\overline{\text{RES}}$ 引脚执行上电复位前，产生 WDT 引起的上电复位/H-UDI 复位时，不清除内部的解除源信息，之后需再次转移至深度待机模式时会错误地解除。

第 29 章 用户调试接口 (H-UDI)

为了支持仿真器，本 LSI 内置用户调试接口 (H-UDI)。

29.1 特点

用户调试接口 (H-UDI) 具有复位及中断请求功能。

本 LSI 的 H-UDI 用于连接仿真器。

与仿真器的连接方法，参阅仿真器手册。

H-UDI 框图，如图 29.1 所示。

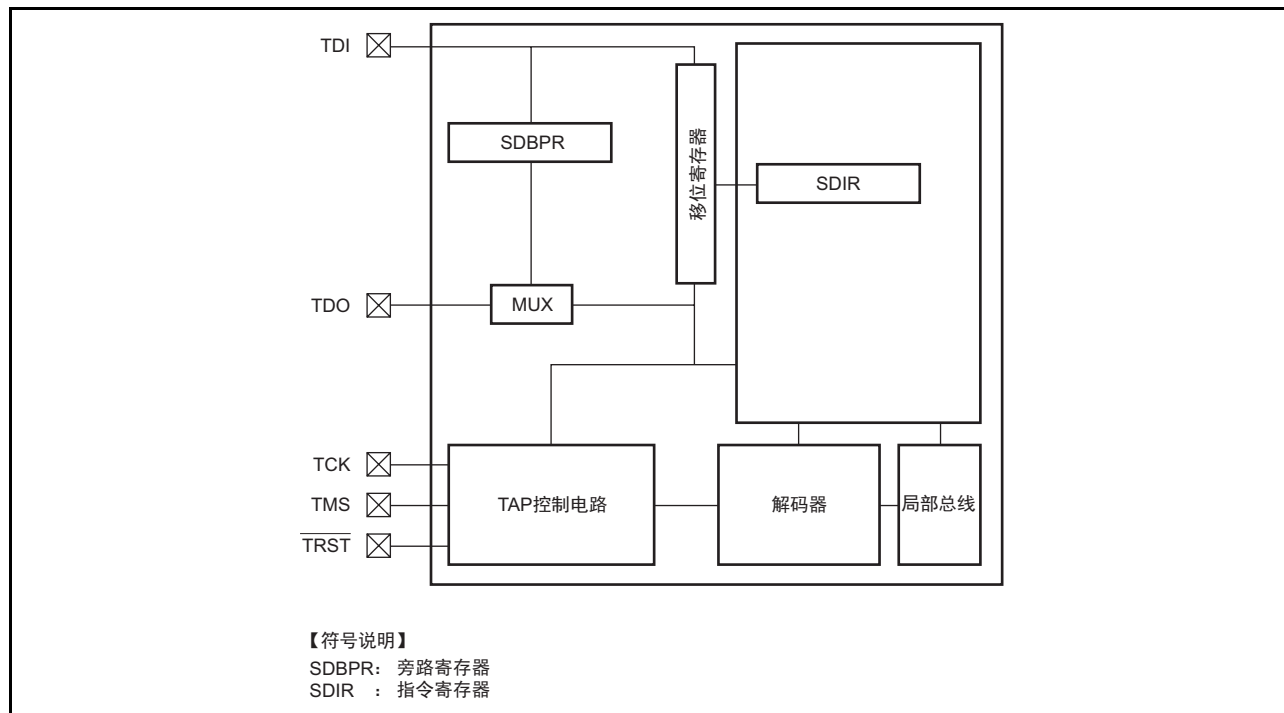


图 29.1 H-UDI 框图

29.2 输入 / 输出引脚

表 29.1 引脚结构

名称	引脚名称	输入 / 输出	功能
H-UDI 串行数据输入 / 输出时钟引脚	TCK	输入	与该时钟同步，从数据输入引脚（TDI）向 H-UDI 串行提供数据，并从数据输出引脚（TDO）输出。
模式选择输入引脚	TMS	输入	通过与 TCK 同步后改变该信号，决定 TAP 控制电路的状态。协议参照图 29.2。
H-UDI 复位输入引脚	$\overline{\text{TRST}}$	输入	与 TCK 异步，接受输入，并由低电平复位 H-UDI。与是否利用 H-UDI 功能无关，必须在上电时使 $\overline{\text{TRST}}$ 保持一定时间的低电平。有关复位结构，详情参阅“29.4.2 复位结构”。
H-UDI 串行数据输入引脚	TDI	输入	通过与 TCK 同步后改变该引脚，向 H-UDI 发送数据。
H-UDI 串行数据输出引脚	TDO	输出	通过与 TCK 同步后读取该引脚，从 H-UDI 读取数据。数据输出时序的初始值为下降沿同步，但通过对 SDIR 输入“TDO 变化时序转换”命令可更改为上升沿同步。详情参阅“29.4.3 TDO 输出时序”。
ASE 模式选择引脚	$\overline{\text{ASEMD}}^*$	输入	在 RES 引脚有效期间，如果对 $\overline{\text{ASEMD}}$ 引脚输入低电平，则变为 ASE 模式；如果输入高电平，则变为产品芯片模式。在 ASE 模式，可使用仿真器专用功能。向 $\overline{\text{ASEMD}}$ 引脚输入的电平，在 RES 引脚无效后，必须至少保持 1 个周期。

【注】 * 不使用仿真器时，必须固定为高电平。

29.3 寄存器说明

H-UDI 有以下寄存器。

表 29.2 寄存器结构

寄存器名称	简称	R/W	初始值	地址	存取长度
旁路寄存器	SDBPR	—	—	—	—
指令寄存器	SDIR	R	H'EFFD	H'FFFE2000	16

29.3.1 旁路寄存器（SDBPR）

SDBPR 为 CPU 不可存取的 1 位寄存器。SDIR 设置为 BYPASS 模式时，SDBPR 被连接于 H-UDI 引脚的 TDI 与 TDO 之间。初始值不定。

29.3.2 指令寄存器（SDIR）

SDIR 为 16 位只读寄存器。 $\overline{\text{TRST}}$ 有效或在 TAP 的 Test-Logic-Reset 状态时，SDIR 被初始化。另外，与 CPU 的模式无关，可从 H-UDI 写入 SDIR。在该寄存器设定保留的命令时，不保证运行。初始值为 H'EFDD。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TI[7:0]							-	-	-	-	-	-	-	-	-
初始值:	1*	1*	1*	0*	1*	1*	1*	1*	1	1	1	1	1	1	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

【注】* TI[7:0]位的初始值为保留值，设定命令时，TI[7:0]位必须设定为非保留值。

位	位名称	初始值	R/W	说 明
15 ~ 8	TI[7:0]	11101111*	R	测试指令 通过 TDI 的串行输入，将 H-UDI 的指令传送至 SDIR。 有关命令，参照表 29.3。
7 ~ 2	—	均为 1	R	保留位 读取值总是为 1。
1	—	0	R	保留位 读取值总是为 0。
0	—	1	R	保留位 读取值总是为 1。

表 29.3 H-UDI 命令

bit15 ~ 8								说明
TI7	TI6	TI5	TI4	TI3	TI2	TI1	TI0	
0	1	1	0	—	—	—	—	H-UDI 复位无效
0	1	1	1	—	—	—	—	H-UDI 复位有效
1	0	0	1	1	1	0	0	TDO 变化时序转换
1	0	1	1	—	—	—	—	H-UDI 中断
1	1	1	1	—	—	—	—	BYPASS 模式
上述以外								保留

29.4 运行说明

29.4.1 TAP 控制器

TAP 控制器的内部状态如图 29.2 所示。

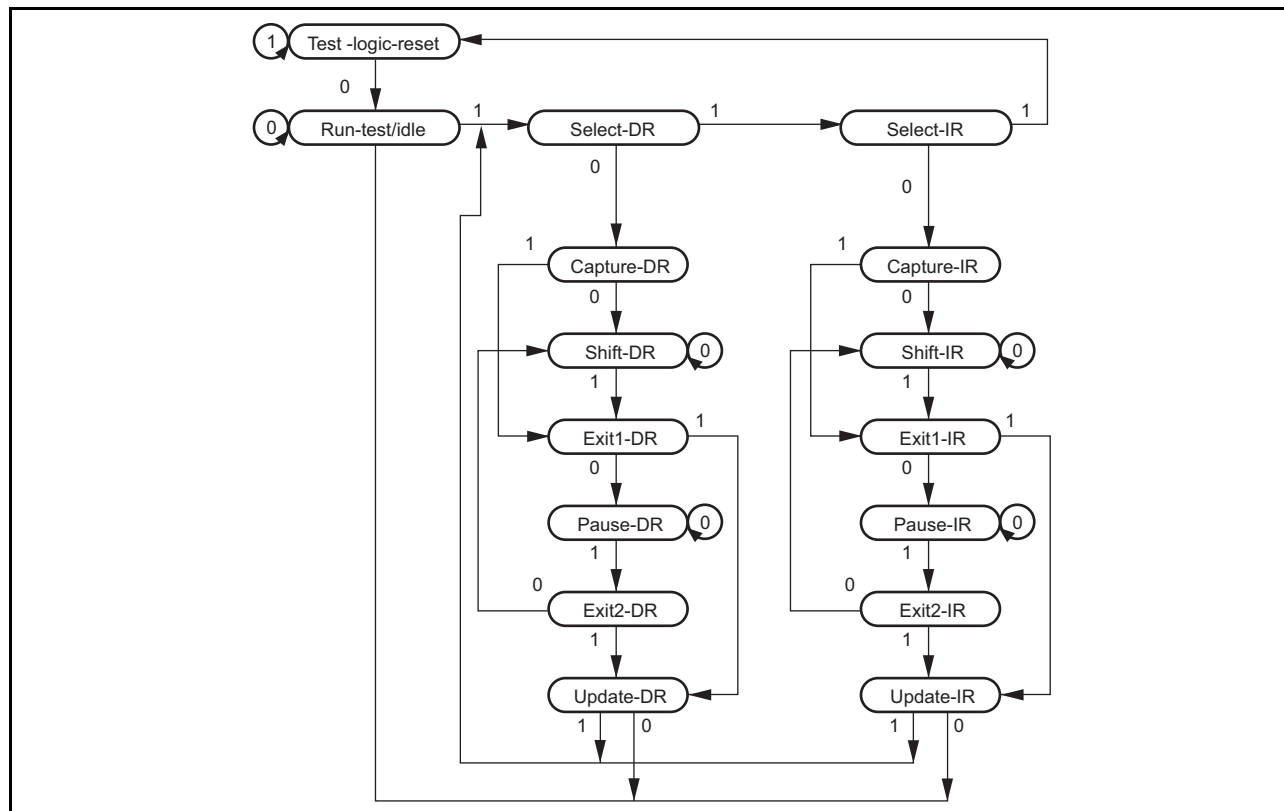


图 29.2 TAP 控制器状态转移图

【注】 转移条件为 TCK 上升沿的 TMS 值。在 TCK 的上升沿采样 TDI 值，并在 TCK 的下降沿移位 TDI 值。有关 TDO 值的变化时序，详情参阅“29.4.3 TDO 输出时序”。在 Shift-DR、Shift-IR 以外的状态，TDO 为高阻抗状态。
TRST 有效后，与 TCK 异步，转移至 Test-Logic-Reset 状态。

29.4.2 复位结构

表 29.4 复位结构

$\overline{\text{ASEMD}}^{*1}$	$\overline{\text{RES}}$	$\overline{\text{TRST}}$	芯片状态
H	L	L	上电复位及 H-UDI 的复位
		H	上电复位
	H	L	仅复位 H-UDI
		H	通常运行
L	L	L	复位保持 *2
		H	上电复位
	H	L	仅复位 H-UDI
		H	通常运行

【注】 *1 选择产品芯片模式或 ASE 模式。

$\overline{\text{ASEMD}}=\text{H}$ 、产品芯片模式

$\overline{\text{ASEMD}}=\text{L}$ 、ASE 模式

*2 在 ASE 模式， $\overline{\text{RES}}$ 引脚无效时，如果 $\overline{\text{TRST}}$ 引脚为低电平则进入进入复位保持状态，且在该状态时不启动 CPU。此后，如果 $\overline{\text{TRST}}$ 设定为高电平，则 H-UDI 运行有效，但不启动 CPU。由上电复位解除复位保持状态。

29.4.3 TDO 输出时序

TDO 的变化时序在初始值时与 TCK 的下降沿同步输出。但是，由 H-UDI 引脚在 SDIR 设置 “TDO 变化时序转换” 命令并通过 Update-IR 后，TDO 的变化时序与 TCK 的上升沿同步。此后，TDO 的变化时序设定与在 TCK 的下降沿同步输出时，需要在上电复位的同时，使 $\overline{\text{TRST}}$ 引脚有效。通过 $\overline{\text{RES}}$ 引脚上电复位时，在 $\overline{\text{RES}}$ 引脚无效后，同步复位仍会在芯片内部运行一定时间。因此，在 $\overline{\text{RES}}$ 引脚无效后立即使 $\overline{\text{TRST}}$ 引脚有效时，清除 “TDO 变化时序转换” 命令，TDO 的变化时序变为与 TCK 的下降沿同步输出。为防止出现该情况， $\overline{\text{RES}}$ 引脚与 $\overline{\text{TRST}}$ 引脚之间的信号变化间隔必须大于等于 20tcyc。

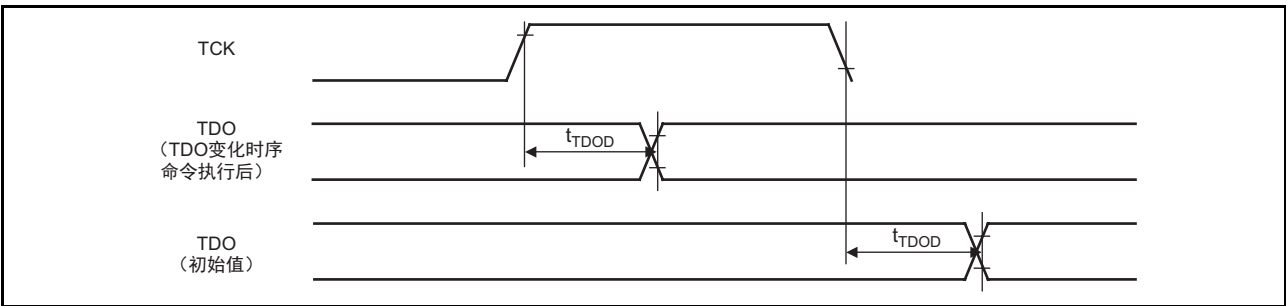


图 29.3 H-UDI 数据传送时序

29.4.4 H-UDI 复位

通过向 SDIR 设置 H-UDI 复位有效命令，产生 H-UDI 复位。H-UDI 复位与上电复位相同。通过设置 H-UDI 复位无效命令解除 H-UDI 复位。H-UDI 复位有效命令与 H-UDI 复位无效命令之间所需的间隔时间与上电复位时 RES 引脚保持低电平的时间相同。

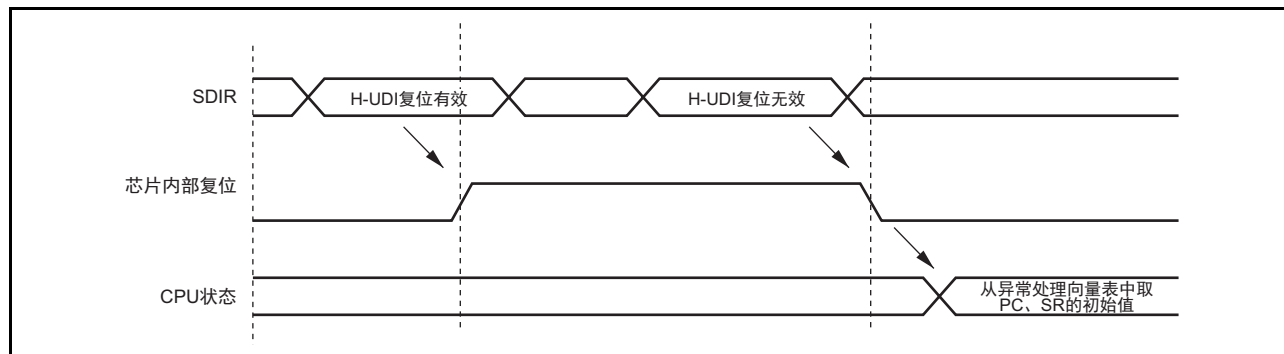


图 29.4 H-UDI 复位

29.4.5 H-UDI 中断

通过向 SDIR 设置来自 H-UDI 的命令，可产生 H-UDI 中断。H-UDI 中断为一般异常 / 中断运行，从异常处理向量表取出异常服务程序的起始地址，并跳转至该地址后执行程序。该中断请求的优先顺序固定为 15。

即使在睡眠模式也可接受 H-UDI 中断，但在软件待机模式不可接受 H-UDI 中断。

29.5 使用时的注意事项

1. H-UDI 命令一旦设置便不可更改，除非从 H-UDI 重新设置其他命令。连续设置相同命令时，必须先设置不影响芯片运行的命令（BYPASS 模式等）后，重新设置命令。
2. 在软件待机模式、H-UDI 模块待机状态，不可使用 H-UDI 的全部功能。同时，为了在待机模式前后保持 TAP 状态，转移至待机模式时，必须预先将 TCK 设置为高电平。
3. 与是否使用 H-UDI 无关，上电时，通过 RES 引脚有效解除深度待机时，必须将 TRST 引脚设置为低电平并初始化 H-UDI。
4. 设置“TDO 变化时序转换”命令，并在 RES 引脚无效后立即使 TRST 引脚有效时，有可能清除“TDO 变化时序转换”命令。为防止出现该情况，在设置“TDO 变化时序转换”命令时，RES 引脚与 TRST 引脚间的信号变化间隔必须大于等于 20tcyc。详情参阅“29.4.3 TDO 输出时序”。
5. TRST 引脚无效后启动 TAP 控制器时，必须空出大于等于 200ns 的间隔。

第 30 章 寄存器一览表

本章按照以下结构，说明本 LSI 内部 I/O 寄存器信息。

(1) 寄存器地址一览表（按各功能模块、手册章节编号顺序）

- 以各功能模块、手册章节编号顺序记载寄存器。
- 不得存取寄存器地址一览表未记载的保留地址。
- 地址为 16 位或 32 位时，以大端法为前提，记载 MSB 侧的地址。

(2) 寄存器位一览表

- 以“寄存器地址一览表（按各功能模块、手册章节编号顺序）”的顺序，记载各寄存器的位结构。
- 保留位在位名称栏用“—”表示。
- 表中位名称栏为空白的部分，表示该寄存器全部分配给计数器或数据。

(3) 各运行模式的寄存器状态一览表

- 以“寄存器地址一览表（按各功能模块、手册章节编号顺序）”的顺序，记载寄存器的状态。
- 有关各位的初始状态，详情参阅对应章节的寄存器说明。
- 表示基本运行模式的寄存器状态。有内部外围模块固有的复位时，详情参阅内部外围模块章节。

(4) 写入内部外围模块的寄存器时的注意事项

存取内部外围模块寄存器时，经过内部总线需不少于外围模块时钟（Pφ）的 2 个周期。由 CPU 写入内部外围寄存器时，CPU 不等寄存器写入结束，即执行后续指令。

以降低功耗而转移至软件待机模式为例说明。为了执行转移，需将 STBCR 寄存器的 STBY 位置 1，再执行 SLEEP 指令，但是，执行 SLEEP 指令前，必须虚读 STBCR 寄存器。如果不虚读，因 STBY 位置 1 前，CPU 执行 SLEEP 指令，因此导致不转移至软件待机模式，而转移至睡眠模式。为了等待写入 STBY 位，需对 STBCR 寄存器进行虚读。

如同此例，执行后续指令时，如果需通过内部外围寄存器反映变化，必须在寄存器写入指令后虚读同一寄存器，再执行后续指令。

30.1 寄存器地址一览表（按各功能模块、手册章节编号顺序）

模块名称	寄存器名称	简称	位数	地址	存取长度
CPG	频率控制寄存器	FRQCR	16	H'FFFE0010	16
INTC	中断控制寄存器 0	ICR0	16	H'FFFE0800	16、32
	中断控制寄存器 1	ICR1	16	H'FFFE0802	16、32
	中断控制寄存器 2	ICR2	16	H'FFFE0804	16、32
	IRQ 中断请求寄存器	IRQRR	16	H'FFFE0806	16、32
	PINT 中断允许寄存器	PINTER	16	H'FFFE0808	16、32
	PINT 中断请求寄存器	PIRR	16	H'FFFE080A	16、32
	存储体控制寄存器	IBCR	16	H'FFFE080C	16、32
	存储体编号寄存器	IBNR	16	H'FFFE080E	16、32
	中断优先级设定寄存器 01	IPR01	16	H'FFFE0818	16、32
	中断优先级设定寄存器 02	IPR02	16	H'FFFE081A	16、32
	中断优先级设定寄存器 05	IPR05	16	H'FFFE0820	16、32
	中断优先级设定寄存器 06	IPR06	16	H'FFFE0C00	16、32
	中断优先级设定寄存器 07	IPR07	16	H'FFFE0C02	16、32
	中断优先级设定寄存器 08	IPR08	16	H'FFFE0C04	16、32
	中断优先级设定寄存器 09	IPR09	16	H'FFFE0C06	16、32
	中断优先级设定寄存器 10	IPR10	16	H'FFFE0C08	16、32
	中断优先级设定寄存器 11	IPR11	16	H'FFFE0C0A	16、32
	中断优先级设定寄存器 12	IPR12	16	H'FFFE0C0C	16、32
	中断优先级设定寄存器 13	IPR13	16	H'FFFE0C0E	16、32
	中断优先级设定寄存器 14	IPR14	16	H'FFFE0C10	16、32
	中断优先级设定寄存器 15	IPR15	16	H'FFFE0C12	16、32
	中断优先级设定寄存器 16	IPR16	16	H'FFFE0C14	16、32
	中断优先级设定寄存器 17	IPR17	16	H'FFFE0C16	16、32
UBC	断点地址寄存器 _0	BAR_0	32	H'FFFC0400	32
	断点地址屏蔽寄存器 _0	BAMR_0	32	H'FFFC0404	32
	断点数据寄存器 _0	BDR_0	32	H'FFFC0408	32
	断点数据屏蔽寄存器 _0	BDMR_0	32	H'FFFC040C	32
	断点地址寄存器 _1	BAR_1	32	H'FFFC0410	32
	断点地址屏蔽寄存器 _1	BAMR_1	32	H'FFFC0414	32
	断点数据寄存器 _1	BDR_1	32	H'FFFC0418	32
	断点数据屏蔽寄存器 _1	BDMR_1	32	H'FFFC041C	32
	断点总线周期寄存器 _0	BBR_0	16	H'FFFC04A0	16

模块名称	寄存器名称	简称	位数	地址	存取长度
UBC	断点总线周期寄存器 _1	BBR_1	16	H'FFFC04B0	16
	断点控制寄存器	BRCR	32	H'FFFC04C0	32
高速缓存	高速缓存控制寄存器 1	CCR1	32	H'FFFC1000	32
	高速缓存控制寄存器 2	CCR2	32	H'FFFC1004	32
BSC	共用控制寄存器	CMNCR	32	H'FFFC0000	32
	CS0 空间总线控制寄存器	CS0BCR	32	H'FFFC0004	32
	CS1 空间总线控制寄存器	CS1BCR	32	H'FFFC0008	32
	CS2 空间总线控制寄存器	CS2BCR	32	H'FFFC000C	32
	CS3 空间总线控制寄存器	CS3BCR	32	H'FFFC0010	32
	CS4 空间总线控制寄存器	CS4BCR	32	H'FFFC0014	32
	CS5 空间总线控制寄存器	CS5BCR	32	H'FFFC0018	32
	CS6 空间总线控制寄存器	CS6BCR	32	H'FFFC001C	32
	CS7 空间总线控制寄存器	CS7BCR	32	H'FFFC0020	32
	CS0 空间等待控制寄存器	CS0WCR	32	H'FFFC0028	32
	CS1 空间等待控制寄存器	CS1WCR	32	H'FFFC002C	32
	CS2 空间等待控制寄存器	CS2WCR	32	H'FFFC0030	32
	CS3 空间等待控制寄存器	CS3WCR	32	H'FFFC0034	32
	CS4 空间等待控制寄存器	CS4WCR	32	H'FFFC0038	32
	CS5 空间等待控制寄存器	CS5WCR	32	H'FFFC003C	32
	CS6 空间等待控制寄存器	CS6WCR	32	H'FFFC0040	32
	CS7 空间等待控制寄存器	CS7WCR	32	H'FFFC0044	32
	SDRAM 控制寄存器	SDCR	32	H'FFFC004C	32
	刷新定时器控制 / 状态寄存器	RTCSR	32	H'FFFC0050	32
	刷新定时器计数器	RTCNT	32	H'FFFC0054	32
	刷新时间常数寄存器	RTCOR	32	H'FFFC0058	32
DMAC	DMA 源地址寄存器 _0	SAR0	32	H'FFFE1000	16、32
	DMA 目标地址寄存器 _0	DAR0	32	H'FFFE1004	16、32
	DMA 传送计数寄存器 _0	DMATCR0	32	H'FFFE1008	16、32
	DMA 通道控制寄存器 _0	CHCR0	32	H'FFFE100C	8、16、32
	DMA 重加载源地址寄存器 _0	RSAR0	32	H'FFFE1100	16、32
	DMA 重加载目标地址寄存器 _0	RDAR0	32	H'FFFE1104	16、32
	DMA 重加载传送计数寄存器 _0	RDMATCR0	32	H'FFFE1108	16、32
	DMA 源地址寄存器 _1	SAR1	32	H'FFFE1010	16、32
	DMA 目标地址寄存器 _1	DAR1	32	H'FFFE1014	16、32

模块名称	寄存器名称	简称	位数	地址	存取长度
DMAC	DMA 传送计数寄存器 _1	DMATCR1	32	H'FFFE1018	16、32
	DMA 通道控制寄存器 _1	CHCR1	32	H'FFFE101C	8、16、32
	DMA 重加载源地址寄存器 _1	RSAR1	32	H'FFFE1110	16、32
	DMA 重加载目标地址寄存器 _1	RDAR1	32	H'FFFE1114	16、32
	DMA 重加载传送计数寄存器 _1	RDMATCR1	32	H'FFFE1118	16、32
	DMA 源地址寄存器 _2	SAR2	32	H'FFFE1020	16、32
	DMA 目标地址寄存器 _2	DAR2	32	H'FFFE1024	16、32
	DMA 传送计数寄存器 _2	DMATCR2	32	H'FFFE1028	16、32
	DMA 通道控制寄存器 _2	CHCR2	32	H'FFFE102C	8、16、32
	DMA 重加载源地址寄存器 _2	RSAR2	32	H'FFFE1120	16、32
	DMA 重加载目标地址寄存器 _2	RDAR2	32	H'FFFE1124	16、32
	DMA 重加载传送计数寄存器 _2	RDMATCR2	32	H'FFFE1128	16、32
	DMA 源地址寄存器 _3	SAR3	32	H'FFFE1030	16、32
	DMA 目标地址寄存器 _3	DAR3	32	H'FFFE1034	16、32
	DMA 传送计数寄存器 _3	DMATCR3	32	H'FFFE1038	16、32
	DMA 通道控制寄存器 _3	CHCR3	32	H'FFFE103C	8、16、32
	DMA 重加载源地址寄存器 _3	RSAR3	32	H'FFFE1130	16、32
	DMA 重加载目标地址寄存器 _3	RDAR3	32	H'FFFE1134	16、32
	DMA 重加载传送计数寄存器 _3	RDMATCR3	32	H'FFFE1138	16、32
	DMA 源地址寄存器 _4	SAR4	32	H'FFFE1040	16、32
	DMA 目标地址寄存器 _4	DAR4	32	H'FFFE1044	16、32
	DMA 传送计数寄存器 _4	DMATCR4	32	H'FFFE1048	16、32
	DMA 通道控制寄存器 _4	CHCR4	32	H'FFFE104C	8、16、32
	DMA 重加载源地址寄存器 _4	RSAR4	32	H'FFFE1140	16、32
	DMA 重加载目标地址寄存器 _4	RDAR4	32	H'FFFE1144	16、32
	DMA 重加载传送计数寄存器 _4	RDMATCR4	32	H'FFFE1148	16、32
	DMA 源地址寄存器 _5	SAR5	32	H'FFFE1050	16、32
	DMA 目标地址寄存器 _5	DAR5	32	H'FFFE1054	16、32
	DMA 传送计数寄存器 _5	DMATCR5	32	H'FFFE1058	16、32
	DMA 通道控制寄存器 _5	CHCR5	32	H'FFFE105C	8、16、32
	DMA 重加载源地址寄存器 _5	RSAR5	32	H'FFFE1150	16、32
	DMA 重加载目标地址寄存器 _5	RDAR5	32	H'FFFE1154	16、32
	DMA 重加载传送计数寄存器 _5	RDMATCR5	32	H'FFFE1158	16、32
	DMA 源地址寄存器 _6	SAR6	32	H'FFFE1060	16、32

模块名称	寄存器名称	简称	位数	地址	存取长度
DMAC	DMA 目标地址寄存器 _6	DAR6	32	H'FFFE1064	16、32
	DMA 传送计数寄存器 _6	DMATCR6	32	H'FFFE1068	16、32
	DMA 通道控制寄存器 _6	CHCR6	32	H'FFFE106C	8、16、32
	DMA 重加载源地址寄存器 _6	RSAR6	32	H'FFFE1160	16、32
	DMA 重加载目标地址寄存器 _6	RDAR6	32	H'FFFE1164	16、32
	DMA 重加载传送计数寄存器 _6	RDMATCR6	32	H'FFFE1168	16、32
	DMA 源地址寄存器 _7	SAR7	32	H'FFFE1070	16、32
	DMA 目标地址寄存器 _7	DAR7	32	H'FFFE1074	16、32
	DMA 传送计数寄存器 _7	DMATCR7	32	H'FFFE1078	16、32
	DMA 通道控制寄存器 _7	CHCR7	32	H'FFFE107C	8、16、32
	DMA 重加载源地址寄存器 _7	RSAR7	32	H'FFFE1170	16、32
	DMA 重加载目标地址寄存器 _7	RDAR7	32	H'FFFE1174	16、32
	DMA 重加载传送计数寄存器 _7	RDMATCR7	32	H'FFFE1178	16、32
	DMA 操作寄存器	DMAOR	16	H'FFFE1200	8、16
	DMA 扩展资源选择器 0	DMARS0	16	H'FFFE1300	16
	DMA 扩展资源选择器 1	DMARS1	16	H'FFFE1304	16
	DMA 扩展资源选择器 2	DMARS2	16	H'FFFE1308	16
	DMA 扩展资源选择器 3	DMARS3	16	H'FFFE130C	16
MTU2	定时器控制寄存器 _0	TCR_0	8	H'FFFE4300	8
	定时器模式寄存器 _0	TMDR_0	8	H'FFFE4301	8
	定时器 I/O 控制寄存器 H_0	TIORH_0	8	H'FFFE4302	8
	定时器 I/O 控制寄存器 L_0	TIORL_0	8	H'FFFE4303	8
	定时器中断允许寄存器 _0	TIER_0	8	H'FFFE4304	8
	定时器状态寄存器 _0	TSR_0	8	H'FFFE4305	8
	定时器计数器 _0	TCNT_0	16	H'FFFE4306	16
	定时器通用寄存器 A_0	TGRA_0	16	H'FFFE4308	16
	定时器通用寄存器 B_0	TGRB_0	16	H'FFFE430A	16
	定时器通用寄存器 C_0	TGRC_0	16	H'FFFE430C	16
	定时器通用寄存器 D_0	TGRD_0	16	H'FFFE430E	16
	定时器通用寄存器 E_0	TGRE_0	16	H'FFFE4320	16
	定时器通用寄存器 F_0	TGRF_0	16	H'FFFE4322	16
	定时器中断允许寄存器 2_0	TIER2_0	8	H'FFFE4324	8
	定时器状态寄存器 2_0	TSR2_0	8	H'FFFE4325	8
	定时器缓冲运行传送模式寄存器 _0	TBTM_0	8	H'FFFE4326	8

模块名称	寄存器名称	简称	位数	地址	存取长度
MTU2	定时器控制寄存器 _1	TCR_1	8	H'FFFE4380	8
	定时器模式寄存器 _1	TMDR_1	8	H'FFFE4381	8
	定时器 I/O 控制寄存器 _1	TIOR_1	8	H'FFFE4382	8
	定时器中断允许寄存器 _1	TIER_1	8	H'FFFE4384	8
	定时器状态寄存器 _1	TSR_1	8	H'FFFE4385	8
	定时器计数器 _1	TCNT_1	16	H'FFFE4386	16
	定时器通用寄存器 A_1	TGRA_1	16	H'FFFE4388	16
	定时器通用寄存器 B_1	TGRB_1	16	H'FFFE438A	16
	定时器输入捕捉控制寄存器	TICCR	8	H'FFFE4390	8
	定时器控制寄存器 _2	TCR_2	8	H'FFFE4000	8
	定时器模式寄存器 _2	TMDR_2	8	H'FFFE4001	8
	定时器 I/O 控制寄存器 _2	TIOR_2	8	H'FFFE4002	8
	定时器中断允许寄存器 _2	TIER_2	8	H'FFFE4004	8
	定时器状态寄存器 _2	TSR_2	8	H'FFFE4005	8
	定时器计数器 _2	TCNT_2	16	H'FFFE4006	16
	定时器通用寄存器 A_2	TGRA_2	16	H'FFFE4008	16
	定时器通用寄存器 B_2	TGRB_2	16	H'FFFE400A	16
	定时器控制寄存器 _3	TCR_3	8	H'FFFE4200	8
	定时器模式寄存器 _3	TMDR_3	8	H'FFFE4202	8
	定时器 I/O 控制寄存器 H_3	TIORH_3	8	H'FFFE4204	8
	定时器 I/O 控制寄存器 L_3	TIORL_3	8	H'FFFE4205	8
	定时器中断允许寄存器 _3	TIER_3	8	H'FFFE4208	8
	定时器状态寄存器 _3	TSR_3	8	H'FFFE422C	8
	定时器计数器 _3	TCNT_3	16	H'FFFE4210	16
	定时器通用寄存器 A_3	TGRA_3	16	H'FFFE4218	16
	定时器通用寄存器 B_3	TGRB_3	16	H'FFFE421A	16
	定时器通用寄存器 C_3	TGRC_3	16	H'FFFE4224	16
	定时器通用寄存器 D_3	TGRD_3	16	H'FFFE4226	16
	定时器缓冲运行传送模式寄存器 _3	TBTM_3	8	H'FFFE4238	8
	定时器控制寄存器 _4	TCR_4	8	H'FFFE4201	8
	定时器模式寄存器 _4	TMDR_4	8	H'FFFE4203	8
	定时器 I/O 控制寄存器 H_4	TIORH_4	8	H'FFFE4206	8
	定时器 I/O 控制寄存器 L_4	TIORL_4	8	H'FFFE4207	8
	定时器中断允许寄存器 _4	TIER_4	8	H'FFFE4209	8

模块名称	寄存器名称	简称	位数	地址	存取长度
MTU2	定时器状态寄存器 _4	TSR_4	8	H'FFFE422D	8
	定时器计数器 _4	TCNT_4	16	H'FFFE4212	16
	定时器通用寄存器 A_4	TGRA_4	16	H'FFFE421C	16
	定时器通用寄存器 B_4	TGRB_4	16	H'FFFE421E	16
	定时器通用寄存器 C_4	TGRC_4	16	H'FFFE4228	16
	定时器通用寄存器 D_4	TGRD_4	16	H'FFFE422A	16
	定时器缓冲运行传送模式寄存器 _4	TBTM_4	8	H'FFFE4239	8
	定时器 A/D 转换开始请求控制寄存器	TADCR	16	H'FFFE4240	16
	定时器 A/D 转换开始请求周期设定寄存器 A_4	TADCORA_4	16	H'FFFE4242	16
	定时器 A/D 转换开始请求周期设定寄存器 B_4	TADCORB_4	16	H'FFFE4244	16
	定时器 A/D 转换开始请求周期设定缓冲寄存器 A_4	TADCOBRA_4	16	H'FFFE4246	16
	定时器 A/D 转换开始请求周期设定缓冲寄存器 B_4	TADCOBRB_4	16	H'FFFE4248	16
	定时器开始寄存器	TSTR	8	H'FFFE4280	8
	定时器同步寄存器	TSYR	8	H'FFFE4281	8
	定时器读取 / 写入允许寄存器	TRWER	8	H'FFFE4284	8
	定时器输出主控允许寄存器	TOER	8	H'FFFE420A	8
	定时器输出控制寄存器 1	TOCR1	8	H'FFFE420E	8
	定时器输出控制寄存器 2	TOCR2	8	H'FFFE420F	8
	定时器门控寄存器	TGCR	8	H'FFFE420D	8
	定时器周期数据寄存器	TCDR	16	H'FFFE4214	16
	定时器空载时间数据寄存器	TDDR	16	H'FFFE4216	16
	定时器副计数器	TCNTS	16	H'FFFE4220	16
	定时器周期缓冲寄存器	TCBR	16	H'FFFE4222	16
	定时器中断跳过设定寄存器	TITCR	8	H'FFFE4230	8
	定时器中断跳过次数计数器	TITCNT	8	H'FFFE4231	8
	定时器缓冲传送设定寄存器	TBTER	8	H'FFFE4232	8
	定时器空载时间允许寄存器	TDER	8	H'FFFE4234	8
	定时器波形控制寄存器	TWCR	8	H'FFFE4260	8
	定时器输出电平缓冲寄存器	TOLBR	8	H'FFFE4236	8
CMT	比较匹配定时器开始寄存器	CMSTR	16	H'FFFE0000	16
	比较匹配定时器控制 / 状态寄存器 _0	CMCSR0	16	H'FFFE0002	16
	比较匹配计数器 _0	CMCNT0	16	H'FFFE0004	8、16

模块名称	寄存器名称	简称	位数	地址	存取长度
CMT	比较匹配常数寄存器 _0	CMCOR0	16	H'FFFE0006	8、16
	比较匹配定时器控制 / 状态寄存器 _1	CMCSR1	16	H'FFFE0008	16
	比较匹配计数器 _1	CMCNT1	16	H'FFFE000A	8、16
	比较匹配常数寄存器 _1	CMCOR1	16	H'FFFE000C	8、16
WDT	看门狗定时器控制 / 状态寄存器	WTCNT	8	H'FFFE0002	8、16
	看门狗定时器计数器	WTCSR	8	H'FFFE0000	8、16
	看门狗定时器复位控制 / 状态寄存器	WRCSR	8	H'FFFE0004	8、16
RTC	64Hz 计数器	R64CNT	8	H'FFFF2000	8
	秒计数器	RSECNT	8	H'FFFF2002	8
	分计数器	RMINCNT	8	H'FFFF2004	8
	小时计数器	RHRCNT	8	H'FFFF2006	8
	星期计数器	RWKCNT	8	H'FFFF2008	8
	日计数器	RDAYCNT	8	H'FFFF200A	8
	月计数器	RMONCNT	8	H'FFFF200C	8
	年计数器	RYRCNT	16	H'FFFF200E	16
	秒报警寄存器	RSECAR	8	H'FFFF2010	8
	分报警寄存器	RMINAR	8	H'FFFF2012	8
	小时报警寄存器	RHRAR	8	H'FFFF2014	8
	星期报警寄存器	RWKAR	8	H'FFFF2016	8
	日报警寄存器	RDAYAR	8	H'FFFF2018	8
	月报警寄存器	RMONAR	8	H'FFFF201A	8
	年报警寄存器	RYRAR	16	H'FFFF2020	16
	RTC 控制寄存器 1	RCR1	8	H'FFFF201C	8
	RTC 控制寄存器 2	RCR2	8	H'FFFF201E	8
	RTC 控制寄存器 3	RCR3	8	H'FFFF2024	8
SCIF	串行模式寄存器 _0	SCSMR_0	16	H'FFFE8000	16
	位速率寄存器 _0	SCBRR_0	8	H'FFFE8004	8
	串行控制寄存器 _0	SCSCR_0	16	H'FFFE8008	16
	发送 FIFO 数据寄存器 _0	SCFTDR_0	8	H'FFFE800C	8
	串行状态寄存器 _0	SCFSR_0	16	H'FFFE8010	16
	接收 FIFO 数据寄存器 _0	SCFRDR_0	8	H'FFFE8014	8
	FIFO 控制寄存器 _0	SCFCR_0	16	H'FFFE8018	16
	FIFO 数据计数设置寄存器 _0	SCFDR_0	16	H'FFFE801C	16
	串行端口寄存器 _0	SCSPTR_0	16	H'FFFE8020	16

模块名称	寄存器名称	简称	位数	地址	存取长度
SCIF	线状态寄存器 _0	SCLSR_0	16	H'FFFE8024	16
	串行扩展模式寄存器 _0	SCEMR_0	16	H'FFFE8028	16
	串行模式寄存器 _1	SCSMR_1	16	H'FFFE8800	16
	位速率寄存器 _1	SCBRR_1	8	H'FFFE8804	8
	串行控制寄存器 _1	SCSCR_1	16	H'FFFE8808	16
	发送 FIFO 数据寄存器 _1	SCFTDR_1	8	H'FFFE880C	8
	串行状态寄存器 _1	SCFSR_1	16	H'FFFE8810	16
	接收 FIFO 数据寄存器 _1	SCFRDR_1	8	H'FFFE8814	8
	FIFO 控制寄存器 _1	SCFCR_1	16	H'FFFE8818	16
	FIFO 数据计数设置寄存器 _1	SCFDR_1	16	H'FFFE881C	16
	串行端口寄存器 _1	SCSPTR_1	16	H'FFFE8820	16
	行状态寄存器 _1	SCLSR_1	16	H'FFFE8824	16
	串行扩展模式寄存器 _1	SCEMR_1	16	H'FFFE8828	16
	串行模式寄存器 _2	SCSMR_2	16	H'FFFE9000	16
	位速率寄存器 _2	SCBRR_2	8	H'FFFE9004	8
	串行控制寄存器 _2	SCSCR_2	16	H'FFFE9008	16
	发送 FIFO 数据寄存器 _2	SCFTDR_2	8	H'FFFE900C	8
	串行状态寄存器 _2	SCFSR_2	16	H'FFFE9010	16
	接收 FIFO 数据寄存器 _2	SCFRDR_2	8	H'FFFE9014	8
	FIFO 控制寄存器 _2	SCFCR_2	16	H'FFFE9018	16
	FIFO 数据计数设置寄存器 _2	SCFDR_2	16	H'FFFE901C	16
	串行端口寄存器 _2	SCSPTR_2	16	H'FFFE9020	16
	线状态寄存器 _2	SCLSR_2	16	H'FFFE9024	16
	串行扩展模式寄存器 _2	SCEMR_2	16	H'FFFE9028	16
	串行模式寄存器 _3	SCSMR_3	16	H'FFFE9800	16
	位速率寄存器 _3	SCBRR_3	8	H'FFFE9804	8
	串行控制寄存器 _3	SCSCR_3	16	H'FFFE9808	16
	发送 FIFO 数据寄存器 _3	SCFTDR_3	8	H'FFFE980C	8
	串行状态寄存器 _3	SCFSR_3	16	H'FFFE9810	16
	接收 FIFO 数据寄存器 _3	SCFRDR_3	8	H'FFFE9814	8
	FIFO 控制寄存器 _3	SCFCR_3	16	H'FFFE9818	16
	FIFO 数据计数设置寄存器 _3	SCFDR_3	16	H'FFFE981C	16
	串行端口寄存器 _3	SCSPTR_3	16	H'FFFE9820	16
	线状态寄存器 _3	SCLSR_3	16	H'FFFE9824	16
	串行扩展模式寄存器 _3	SCEMR_3	16	H'FFFE9828	16

模块名称	寄存器名称	简称	位数	地址	存取长度
SSU	SS 控制寄存器 H_0	SSCRH_0	8	H'FFFE7000	8、16
	SS 控制寄存器 L_0	SSCRL_0	8	H'FFFE7001	8
	SS 模式寄存器 _0	SSMR_0	8	H'FFFE7002	8、16
	SS 允许寄存器 _0	SSER_0	8	H'FFFE7003	8
	SS 状态寄存器 _0	SSSR_0	8	H'FFFE7004	8、16
	SS 控制寄存器 2_0	SSCR2_0	8	H'FFFE7005	8
	SS 发送数据寄存器 0_0	SSTDR0_0	8	H'FFFE7006	8、16
	SS 发送数据寄存器 1_0	SSTDR1_0	8	H'FFFE7007	8
	SS 发送数据寄存器 2_0	SSTDR2_0	8	H'FFFE7008	8、16
	SS 发送数据寄存器 3_0	SSTDR3_0	8	H'FFFE7009	8
	SS 接收数据寄存器 0_0	SSRDR0_0	8	H'FFFE700A	8、16
	SS 接收数据寄存器 1_0	SSRDR1_0	8	H'FFFE700B	8
	SS 接收数据寄存器 2_0	SSRDR2_0	8	H'FFFE700C	8、16
	SS 接收数据寄存器 3_0	SSRDR3_0	8	H'FFFE700D	8
	SS 控制寄存器 H_1	SSCRH_1	8	H'FFFE7800	8、16
	SS 控制寄存器 L_1	SSCRL_1	8	H'FFFE7801	8
	SS 模式寄存器 _1	SSMR_1	8	H'FFFE7802	8、16
	SS 允许寄存器 _1	SSER_1	8	H'FFFE7803	8
	SS 状态寄存器 _1	SSSR_1	8	H'FFFE7804	8、16
	SS 控制寄存器 2_1	SSCR2_1	8	H'FFFE7805	8
	SS 发送数据寄存器 0_1	SSTDR0_1	8	H'FFFE7806	8、16
	SS 发送数据寄存器 1_1	SSTDR1_1	8	H'FFFE7807	8
	SS 发送数据寄存器 2_1	SSTDR2_1	8	H'FFFE7808	8、16
	SS 发送数据寄存器 3_1	SSTDR3_1	8	H'FFFE7809	8
	SS 接收数据寄存器 0_1	SSRDR0_1	8	H'FFFE780A	8、16
	SS 接收数据寄存器 1_1	SSRDR1_1	8	H'FFFE780B	8
	SS 接收数据寄存器 2_1	SSRDR2_1	8	H'FFFE780C	8、16
	SS 接收数据寄存器 3_1	SSRDR3_1	8	H'FFFE780D	8
IIC3	I2C 总线控制寄存器 1_0	ICCR1_0	8	H'FFFEE000	8
	I2C 总线控制寄存器 2_0	ICCR2_0	8	H'FFFEE001	8
	I2C 总线模式寄存器 _0	ICMR_0	8	H'FFFEE002	8
	I2C 总线中断允许寄存器 _0	ICIER_0	8	H'FFFEE003	8
	I2C 总线状态寄存器 _0	ICSR_0	8	H'FFFEE004	8
	从属地址寄存器 _0	SAR_0	8	H'FFFEE005	8
	I2C 总线发送数据寄存器 _0	ICDRT_0	8	H'FFFEE006	8

模块名称	寄存器名称	简称	位数	地址	存取长度
IIC3	I2C 总线接收数据寄存器 _0	ICDRR_0	8	H'FFFEE007	8
	NF2CYC 寄存器 _0	NF2CYC_0	8	H'FFFEE008	8
	I2C 总线控制寄存器 1_1	ICCR1_1	8	H'FFFEE400	8
	I2C 总线控制寄存器 2_1	ICCR2_1	8	H'FFFEE401	8
	I2C 总线模式寄存器 _1	ICMR_1	8	H'FFFEE402	8
	I2C 总线中断允许寄存器 _1	ICIER_1	8	H'FFFEE403	8
	I2C 总线状态寄存器 _1	ICSR_1	8	H'FFFEE404	8
	从属地址寄存器 _1	SAR_1	8	H'FFFEE405	8
	I2C 总线发送数据寄存器 _1	ICDRT_1	8	H'FFFEE406	8
	I2C 总线接收数据寄存器 _1	ICDRR_1	8	H'FFFEE407	8
	NF2CYC 寄存器 _1	NF2CYC_1	8	H'FFFEE408	8
	I2C 总线控制寄存器 1_2	ICCR1_2	8	H'FFFEE800	8
	I2C 总线控制寄存器 2_2	ICCR2_2	8	H'FFFEE801	8
	I2C 总线模式寄存器 _2	ICMR_2	8	H'FFFEE802	8
	I2C 总线中断允许寄存器 _2	ICIER_2	8	H'FFFEE803	8
	I2C 总线状态寄存器 _2	ICSR_2	8	H'FFFEE804	8
	从属地址寄存器 _2	SAR_2	8	H'FFFEE805	8
	I2C 总线发送数据寄存器 _2	ICDRT_2	8	H'FFFEE806	8
	I2C 总线接收数据寄存器 _2	ICDRR_2	8	H'FFFEE807	8
	NF2CYC 寄存器 _2	NF2CYC_2	8	H'FFFEE808	8
	I2C 总线控制寄存器 1_3	ICCR1_3	8	H'FFFEEC00	8
	I2C 总线控制寄存器 2_3	ICCR2_3	8	H'FFFEEC01	8
	I2C 总线模式寄存器 _3	ICMR_3	8	H'FFFEEC02	8
	I2C 总线中断允许寄存器 _3	ICIER_3	8	H'FFFEEC03	8
	I2C 总线状态寄存器 _3	ICSR_3	8	H'FFFEEC04	8
	从属地址寄存器 _3	SAR_3	8	H'FFFEEC05	8
	I2C 总线发送数据寄存器 _3	ICDRT_3	8	H'FFFEEC06	8
	I2C 总线接收数据寄存器 _3	ICDRR_3	8	H'FFFEEC07	8
	NF2CYC 寄存器 _3	NF2CYC_3	8	H'FFFEEC08	8
SSI	控制寄存器 _0	SSICR_0	32	H'FFFFC000	32
	状态寄存器 _0	SSISR_0	32	H'FFFFC004	32
	发送数据寄存器 _0	SSITDR_0	32	H'FFFFC008	32
	接收数据寄存器 _0	SSIRDR_0	32	H'FFFFC00C	32
	控制寄存器 _1	SSICR_1	32	H'FFFFC800	32
	状态寄存器 _1	SSISR_1	32	H'FFFFC804	32

模块名称	寄存器名称	简称	位数	地址	存取长度
SSI	发送数据寄存器 _1	SSITDR_1	32	H'FFFC808	32
	接收数据寄存器 _1	SSIRDR_1	32	H'FFFC80C	32
	控制寄存器 _2	SSICR_2	32	H'FFFD000	32
	状态寄存器 _2	SSISR_2	32	H'FFFD004	32
	发送数据寄存器 _2	SSITDR_2	32	H'FFFD008	32
	接收数据寄存器 _2	SSIRDR_2	32	H'FFFD00C	32
	控制寄存器 _3	SSICR_3	32	H'FFFD800	32
	状态寄存器 _3	SSISR_3	32	H'FFFD804	32
	发送数据寄存器 _3	SSITDR_3	32	H'FFFD808	32
	接收数据寄存器 _3	SSIRDR_3	32	H'FFFD80C	32
RCAN-TL1	主控制寄存器 _0	MCR_0	16	H'FFF0000	16
	通用状态寄存器 _0	GSR_0	16	H'FFF0002	16
	位结构寄存器 1_0	BCR1_0	16	H'FFF0004	16
	位结构寄存器 0_0	BCR0_0	16	H'FFF0006	16
	中断请求寄存器 _0	IRR_0	16	H'FFF0008	16
	中断屏蔽寄存器 _0	IMR_0	16	H'FFF000A	16
	发送 / 接收错误计数器 _0	TEC_REC_0	16	H'FFF000C	8、16
	发送等待寄存器 1_0	TXPR1_0	16	H'FFF0020	32
	发送等待寄存器 0_0	TXPR0_0	16	H'FFF0022	16
	发送取消寄存器 1_0	TXCR1_0	16	H'FFF0028	16
	发送取消寄存器 0_0	TXCR0_0	16	H'FFF002A	16
	发送应答寄存器 1_0	TXACK1_0	16	H'FFF0030	16
	发送应答寄存器 0_0	TXACK0_0	16	H'FFF0032	16
	中止应答寄存器 1_0	ABACK1_0	16	H'FFF0038	16
	中止应答寄存器 0_0	ABACK0_0	16	H'FFF003A	16
	数据帧接收完成寄存器 1_0	RXPR1_0	16	H'FFF0040	16
	数据帧接收完成寄存器 0_0	RXPR0_0	16	H'FFF0042	16
	远程帧接收完成寄存器 1_0	RFPR1_0	16	H'FFF0048	16
	远程帧接收完成寄存器 0_0	RFPR0_0	16	H'FFF004A	16
	邮箱中断屏蔽寄存器 1_0	MBIMR1_0	16	H'FFF0050	16
	邮箱中断屏蔽寄存器 0_0	MBIMR0_0	16	H'FFF0052	16
	未读信息状态寄存器 1_0	UMSR1_0	16	H'FFF0058	16
	未读信息状态寄存器 0_0	UMSR0_0	16	H'FFF005A	16
	定时器触发控制寄存器 _0	TTCR0_0	16	H'FFF0080	16

模块名称	寄存器名称	简称	位数	地址	存取长度
RCAN-TL1	最大周期 / Tx_Enable_Window 寄存器_0	CMAX_TEW_0	16	H'FFFF0084	16
	基准触发偏移寄存器_0	RFTROFF_0	16	H'FFFF0086	16
	定时器状态寄存器_0	TSR_0	16	H'FFFF0088	16
	周期计数器寄存器_0	CCR_0	16	H'FFFF008A	16
	定时器计数器寄存器_0	TCNTR_0	16	H'FFFF008C	16
	周期时间寄存器_0	CYCTR_0	16	H'FFFF0090	16
	基准标记寄存器_0	RFMK_0	16	H'FFFF0094	16
	时间比较匹配寄存器 0_0	TCMR0_0	16	H'FFFF0098	16
	时间比较匹配寄存器 1_0	TCMR1_0	16	H'FFFF009C	16
	时间比较匹配寄存器 2_0	TCMR2_0	16	H'FFFF00A0	16
	发送触发时间选择寄存器_0	TTTSEL_0	16	H'FFFF00A4	16
	邮箱 n 控制 0H_0 (n=0 ~ 31)	MBn_CONTROL0_H_0 (n=0 ~ 31)	16	H'FFFF0100+n×32	16、32
	邮箱 n 控制 0L_0 (n=0 ~ 31)	MBn_CONTROL0_L_0 (n=0 ~ 31)	16	H'FFFF0102+n×32	16
	邮箱 n 局部接收过滤器屏蔽 0_0 (n=0 ~ 31)	MBn_LAFM0_0 (n=0 ~ 31)	16	H'FFFF0104+n×32	16、32
	邮箱 n 局部接收过滤器屏蔽 1_0 (n=0 ~ 31)	MBn_LAFM1_0 (n=0 ~ 31)	16	H'FFFF0106+n×32	16
	邮箱 n 数据 01_0 (n=0 ~ 31)	MBn_DATA_01_0 (n=0 ~ 31)	16	H'FFFF0108+n×32	8、16、32
	邮箱 n 数据 23_0 (n=0 ~ 31)	MBn_DATA_23_0 (n=0 ~ 31)	16	H'FFFF010A+n×32	8、16
	邮箱 n 数据 45_0 (n=0 ~ 31)	MBn_DATA_45_0 (n=0 ~ 31)	16	H'FFFF010C+n×32	8、16、32
	邮箱 n 数据 67_0 (n=0 ~ 31)	MBn_DATA_67_0 (n=0 ~ 31)	16	H'FFFF010E+n×32	8、16
	邮箱 n 控制 1_0 (n=0 ~ 31)	MBn_CONTROL1_0 (n=0 ~ 31)	16	H'FFFF0110+n×32	8、16
	邮箱 n 时戳_0 (n=0 ~ 15、30、31)	MBn_TIMESTAMP_0 (n=0 ~ 15、30、31)	16	H'FFFF0112+n×32	16
	邮箱 n 触发时间_0 (n=24 ~ 30)	MBn_TTT_0 (n=24 ~ 30)	16	H'FFFF0114+n×32	16
	邮箱 nTT 控制_0 (n=24 ~ 29)	MBn_TTCONTROL_0 (n=24 ~ 29)	16	H'FFFF0116+n×32	16

模块名称	寄存器名称	简称	位数	地址	存取长度
RCAN-TL1	主控制寄存器 _1	MCR_1	16	H'FFFF0800	16
	通用状态寄存器 _1	GSR_1	16	H'FFFF0802	16
	位结构寄存器 1_1	BCR1_1	16	H'FFFF0804	16
	位结构寄存器 0_1	BCR0_1	16	H'FFFF0806	16
	中断请求寄存器 _1	IRR_1	16	H'FFFF0808	16
	中断屏蔽寄存器 _1	IMR_1	16	H'FFFF080A	16
	发送 / 接收错误计数器 _1	TEC_REC_1	16	H'FFFF080C	8、16
	发送等待寄存器 1_1	TXPR1_1	16	H'FFFF0820	32
	发送等待寄存器 0_1	TXPR0_1	16	H'FFFF0822	16
	发送取消寄存器 1_1	TXCR1_1	16	H'FFFF0828	16
	发送取消寄存器 0_1	TXCR0_1	16	H'FFFF082A	16
	发送应答寄存器 1_1	TXACK1_1	16	H'FFFF0830	16
	发送应答寄存器 0_1	TXACK0_1	16	H'FFFF0832	16
	中止应答寄存器 1_1	ABACK1_1	16	H'FFFF0838	16
	中止应答寄存器 0_1	ABACK0_1	16	H'FFFF083A	16
	数据帧接收完成寄存器 1_1	RXPR1_1	16	H'FFFF0840	16
	数据帧接收完成寄存器 0_1	RXPR0_1	16	H'FFFF0842	16
	远程帧接收完成寄存器 1_1	RFPR1_1	16	H'FFFF0848	16
	远程帧接收完成寄存器 0_1	RFPR0_1	16	H'FFFF084A	16
	邮箱中断屏蔽寄存器 1_1	MBIMR1_1	16	H'FFFF0850	16
	邮箱中断屏蔽寄存器 0_1	MBIMR0_1	16	H'FFFF0852	16
	未读信息状态寄存器 1_1	UMSR1_1	16	H'FFFF0858	16
	未读信息状态寄存器 0_1	UMSR0_1	16	H'FFFF085A	16
	定时器触发控制寄存器 _1	TTCR0_1	16	H'FFFF0880	16
	最大周期 / Tx_Enable_Window 寄存器 _1	CMAX_TEW_1	16	H'FFFF0884	16
	基准触发偏移寄存器 _1	RFTROFF_1	16	H'FFFF0886	16
	定时器状态寄存器 _1	TSR_1	16	H'FFFF0888	16
	周期计数器寄存器 _1	CCR_1	16	H'FFFF088A	16
	定时器计数器寄存器 _1	TCNTR_1	16	H'FFFF088C	16
	周期时间寄存器 _1	CYCTR_1	16	H'FFFF0890	16
	基准标记寄存器 _1	RFMK_1	16	H'FFFF0894	16
	时间比较匹配寄存器 0_1	TCMR0_1	16	H'FFFF0898	16

模块名称	寄存器名称	简称	位数	地址	存取长度
RCAN-TL1	时间比较匹配寄存器 1_1	TCMR1_1	16	H'FFFF089C	16
	时间比较匹配寄存器 2_1	TCMR2_1	16	H'FFFF08A0	16
	发送触发时间选择寄存器_1	TTTSEL_1	16	H'FFFF08A4	16
	邮箱 n 控制 0H_1 (n=0 ~ 31)	MBn_CONTROL0_H_1 (n=0 ~ 31)	16	H'FFFF0900+n×32	16、32
	邮箱 n 控制 0L_1 (n=0 ~ 31)	MBn_CONTROL0_L_1 (n=0 ~ 31)	16	H'FFFF0902+n×32	16
	邮箱 n 局部接收过滤器屏蔽 0_1 (n=0 ~ 31)	MBn_LAFM0_1 (n=0 ~ 31)	16	H'FFFF0904+n×32	16、32
	邮箱 n 局部接收过滤器屏蔽 1_1 (n=0 ~ 31)	MBn_LAFM1_1 (n=0 ~ 31)	16	H'FFFF0906+n×32	16
	邮箱 n 数据 01_1 (n=0 ~ 31)	MBn_DATA_01_1 (n=0 ~ 31)	16	H'FFFF0908+n×32	8、16、32
	邮箱 n 数据 23_1 (n=0 ~ 31)	MBn_DATA_23_1 (n=0 ~ 31)	16	H'FFFF090A+n×32	8、16
	邮箱 n 数据 45_1 (n=0 ~ 31)	MBn_DATA_45_1 (n=0 ~ 31)	16	H'FFFF090C+n×32	8、16、32
	邮箱 n 数据 67_1 (n=0 ~ 31)	MBn_DATA_67_1 (n=0 ~ 31)	16	H'FFFF090E+n×32	8、16
	邮箱 n 控制 1_1 (n=0 ~ 31)	MBn_CONTROL1_1 (n=0 ~ 31)	16	H'FFFF0910+n×32	8、16
	邮箱 n 时戳_1 (n=0 ~ 15、30、31)	MBn_TIMESTAMP_1 (n=0 ~ 15、30、31)	16	H'FFFF0912+n×32	16
	邮箱 n 触发时间_1 (n=24 ~ 30)	MBn_TTT_1 (n=24 ~ 30)	16	H'FFFF0914+n×32	16
	邮箱 nTT 控制_1 (n=24 ~ 29)	MBn_TTCONTROL_1 (n=24 ~ 29)	16	H'FFFF0916+n×32	16
ADC	A/D 数据寄存器 A	ADDRA	16	H'FFFE5800	16
	A/D 数据寄存器 B	ADDRB	16	H'FFFE5802	16
	A/D 数据寄存器 C	ADDRC	16	H'FFFE5804	16
	A/D 数据寄存器 D	ADDRD	16	H'FFFE5806	16
	A/D 数据寄存器 E	ADDRE	16	H'FFFE5808	16
	A/D 数据寄存器 F	ADDRF	16	H'FFFE580A	16
	A/D 数据寄存器 G	ADDRG	16	H'FFFE580C	16
	A/D 数据寄存器 H	ADDRH	16	H'FFFE580E	16
	A/D 控制 / 状态寄存器	ADCSR	16	H'FFFE5820	16
DAC	D/A 数据寄存器 0	DADR0	8	H'FFFE6800	8、16
	D/A 数据寄存器 1	DADR1	8	H'FFFE6801	8、16
	D/A 控制寄存器	DACR	8	H'FFFE6802	18、16

模块名称	寄存器名称	简称	位数	地址	存取长度
FLCTL	共用控制寄存器	FLCMNCR	32	H'FFFFF000	32
	命令控制寄存器	FLCMDCR	32	H'FFFFF004	32
	命令码寄存器	FLCMCDR	32	H'FFFFF008	32
	地址寄存器	FLADR	32	H'FFFFF00C	32
	地址寄存器 2	FLADR2	32	H'FFFFF03C	32
	数据寄存器	FLDATAR	32	H'FFFFF010	32
	数据计数器寄存器	FLDTCNTR	32	H'FFFFF014	32
	中断 DMA 控制寄存器	FLINTDMACR	32	H'FFFFF018	32
	就绪忙超时设定寄存器	FLBSYTMR	32	H'FFFFF01C	32
	就绪忙超时计数器	FLBSYCNT	32	H'FFFFF020	32
	数据 FIFO 寄存器	FLDTFIFO	32	H'FFFFF050	32
	管理码 FIFO 寄存器	FLECFIFO	32	H'FFFFF060	32
	传送控制寄存器	FLTRCR	8	H'FFFFF02C	8
USB	系统结构控制寄存器	SYSCFG	16	H'FFFC1C00	16
	系统结构状态寄存器	SYSSTS	16	H'FFFC1C02	16
	器件状态控制寄存器	DVSTCTR	16	H'FFFC1C04	16
	测试模式寄存器	TESTMODE	16	H'FFFC1C06	16
	CPU-FIFO 总线结构寄存器	CFBCFG	16	H'FFFC1C0A	16
	DMA0-FIFO 总线结构寄存器	D0FBCFG	16	H'FFFC1C0C	16
	DMA1-FIFO 总线结构寄存器	D1FBCFG	16	H'FFFC1C0E	16
	CFIFO 端口寄存器	CFIFO	32	H'FFFC1C10	8、16、32
	D0FIFO 端口寄存器	D0FIFO	32	H'FFFC1C14	8、16、32
	D1FIFO 端口寄存器	D1FIFO	32	H'FFFC1C18	8、16、32
	CFIFO 端口选择寄存器	CFIFOSEL	16	H'FFFC1C1E	16
	CFIFO 端口控制寄存器	CFIFOCTR	16	H'FFFC1C20	16
	CFIFO 端口 SIE 寄存器	CFIFOSIE	16	H'FFFC1C22	16
	D0FIFO 端口选择寄存器	D0FIFOSEL	16	H'FFFC1C24	16
	D0FIFO 端口控制寄存器	D0FIFOCTR	16	H'FFFC1C26	16
	D0 处理计数器寄存器	D0FIFOTRN	16	H'FFFC1C28	16
	D1FIFO 端口选择寄存器	D1FIFOSEL	16	H'FFFC1C2A	16
	D1FIFO 端口控制寄存器	D1FIFOCTR	16	H'FFFC1C2C	16
	D1 处理计数器寄存器	D1FIFOTRN	16	H'FFFC1C2E	16
	中断允许寄存器 0	INTENB0	16	H'FFFC1C30	16
	中断允许寄存器 1	INTENB1	16	H'FFFC1C32	16
	BRDY 中断允许寄存器	BRDYENB	16	H'FFFC1C36	16

模块名称	寄存器名称	简称	位数	地址	存取长度
USB	NRDY 中断允许寄存器	NRDYENB	16	H'FFFC1C38	16
	BEMP 中断允许寄存器	BEMPENB	16	H'FFFC1C3A	16
	中断状态寄存器 0	INTSTS0	16	H'FFFC1C40	16
	中断状态寄存器 1	INTSTS1	16	H'FFFC1C42	16
	BRDY 中断状态寄存器	BRDYSTS	16	H'FFFC1C46	16
	NRDY 中断状态寄存器	NRDYSTS	16	H'FFFC1C48	16
	BEMP 中断状态寄存器	BEMPSTS	16	H'FFFC1C4A	16
	帧编号寄存器	FRMNUM	16	H'FFFC1C4C	16
	μ 帧编号寄存器	UFRMNUM	16	H'FFFC1C4E	16
	USB 地址寄存器	USBADDR	16	H'FFFC1C50	16
	USB 请求类型寄存器	USBREQ	16	H'FFFC1C54	16
	USB 请求值寄存器	USBVAL	16	H'FFFC1C56	16
	USB 请求变址寄存器	USBINDX	16	H'FFFC1C58	16
	USB 请求长度寄存器	USBLENG	16	H'FFFC1C5A	16
	DCP 结构寄存器	DCPCFG	16	H'FFFC1C5C	16
	DCP Max 包大小寄存器	DCPMAXP	16	H'FFFC1C5E	16
	DCP 控制寄存器	DCPCTR	16	H'FFFC1C60	16
	管道窗选择寄存器	PIPESEL	16	H'FFFC1C64	16
	管道结构寄存器	PIPECFG	16	H'FFFC1C66	16
	管道缓冲指定寄存器	PIPEBUF	16	H'FFFC1C68	16
	管道 Max 包大小寄存器	PEPEMAXP	16	H'FFFC1C6A	16
	管道周期控制寄存器	PIPEPERI	16	H'FFFC1C6C	16
	管道 1 控制寄存器	PIPE1CTR	16	H'FFFC1C70	16
	管道 2 控制寄存器	PIPE2CTR	16	H'FFFC1C72	16
	管道 3 控制寄存器	PIPE3CTR	16	H'FFFC1C74	16
	管道 4 控制寄存器	PIPE4CTR	16	H'FFFC1C76	16
	管道 5 控制寄存器	PIPE5CTR	16	H'FFFC1C78	16
	管道 6 控制寄存器	PIPE6CTR	16	H'FFFC1C7A	16
	管道 7 控制寄存器	PIPE7CTR	16	H'FFFC1C7C	16
	USB AC 特性转换寄存器	USBACSWR	32	H'FFFC1C84	32
LCDC	LCDC 输入时钟寄存器	LDICKR	16	H'FFFFFFC00	16
	LCDC 模块类型寄存器	LDMRT	16	H'FFFFFFC02	16
	LCDC 数据格式寄存器	LDDFR	16	H'FFFFFFC04	16
	LCDC 扫描模式寄存器	LDSMR	16	H'FFFFFFC06	16

模块名称	寄存器名称	简称	位数	地址	存取长度
LCDC	LCDC 上部显示面板用取数据起始地址寄存器	LDSARU	32	H'FFFFFFC08	32
	LCDC 下部显示面板用取数据起始地址寄存器	LDSARL	32	H'FFFFFFC0C	32
	LCDC 显示面板用取数据行地址偏移量寄存器	LDLAOR	16	H'FFFFFFC10	16
	LCDC 调色板控制寄存器	LDPALCR	16	H'FFFFFFC12	16
	调色板数据寄存器 00 ~ FF	LDPR00 ~ FF	32	H'FFFFFF800 ~ H'FFFFFFBFC	32
	LCDC 水平字符数寄存器	LDHCNR	16	H'FFFFFFC14	16
	LCDC 水平同步信号寄存器	LDHSYNR	16	H'FFFFFFC16	16
	LCDC 垂直显示行数寄存器	LDVDLNR	16	H'FFFFFFC18	16
	LCDC 垂直总行数寄存器	LDVTLNR	16	H'FFFFFFC1A	16
	LCDC 垂直同步信号寄存器	LDVSYNR	16	H'FFFFFFC1C	16
	LCDC AC 调制信号交替行数寄存器	LDACLNR	16	H'FFFFFFC1E	16
	LCDC 中断控制寄存器	LDINTR	16	H'FFFFFFC20	16
	LCDC 电源管理模式寄存器	LDPMMR	16	H'FFFFFFC24	16
	LCDC 上电顺序期间寄存器	LDPSPR	16	H'FFFFFFC26	16
	LCDC 控制寄存器	LDCNTR	16	H'FFFFFFC28	16
	LCDC 用户指定中断控制寄存器	LDUINTR	16	H'FFFFFFC34	16
	LCDC 用户指定中断行数寄存器	LDUINTLNR	16	H'FFFFFFC36	16
	LCDC 存储器存取间隔数寄存器	LDLIRNR	16	H'FFFFFFC40	16
PFC	端口 B/I/O 寄存器 L	PBIORL	16	H'FFFE3886	8、16
	端口 B 控制寄存器 L4	PBCRL4	16	H'FFFE3890	16、32
	端口 B 控制寄存器 L3	PBCRL3	16	H'FFFE3892	8、16
	端口 B 控制寄存器 L2	PBCRL2	16	H'FFFE3894	8、16、32
	端口 B 控制寄存器 L1	PBCRL1	16	H'FFFE3896	8、16
	IRQOUT 功能控制寄存器	IFCR	16	H'FFFE38A2	8、16
	端口 C/I/O 寄存器 L	PCIORL	16	H'FFFE3906	8、16
	端口 C 控制寄存器 L4	PCCRL4	16	H'FFFE3910	8、16、32
	端口 C 控制寄存器 L3	PCCRL3	16	H'FFFE3912	8、16
	端口 C 控制寄存器 L2	PCCRL2	16	H'FFFE3914	8、16、32
	端口 C 控制寄存器 L1	PCCRL1	16	H'FFFE3916	8、16
	端口 D/I/O 寄存器 L	PDIORL	16	H'FFFE3986	8、16
	端口 D 控制寄存器 L4	PDCRL4	16	H'FFFE3990	8、16、32
	端口 D 控制寄存器 L3	PDCRL3	16	H'FFFE3992	8、16

模块名称	寄存器名称	简称	位数	地址	存取长度
PFC	端口 D 控制寄存器 L2	PDCRL2	16	H'FFFE3994	8、16、32
	端口 D 控制寄存器 L1	PDCRL1	16	H'FFFE3996	8、16
	端口 E/IO 寄存器 L	PEIORL	16	H'FFFE3A06	8、16
	端口 E 控制寄存器 L4	PECRL4	16	H'FFFE3A10	8、16、32
	端口 E 控制寄存器 L3	PECRL3	16	H'FFFE3A12	8、16
	端口 E 控制寄存器 L2	PECRL2	16	H'FFFE3A14	8、16、32
	端口 E 控制寄存器 L1	PECRL1	16	H'FFFE3A16	8、16
	端口 F/IO 寄存器 H	PFIORH	16	H'FFFE3A84	8、16、32
	端口 F/IO 寄存器 L	PFIORL	16	H'FFFE3A86	8、16
	端口 F 控制寄存器 H4	PFCRH4	16	H'FFFE3A88	8、16、32
	端口 F 控制寄存器 H3	PFCRH3	16	H'FFFE3A8A	8、16
	端口 F 控制寄存器 H2	PFCRH2	16	H'FFFE3A8C	8、16、32
	端口 F 控制寄存器 H1	PFCRH1	16	H'FFFE3A8E	8、16
	端口 F 控制寄存器 L4	PFCRL4	16	H'FFFE3A90	8、16、32
	端口 F 控制寄存器 L3	PFCRL3	16	H'FFFE3A92	8、16
	端口 F 控制寄存器 L2	PFCRL2	16	H'FFFE3A94	8、16、32
	端口 F 控制寄存器 L1	PFCRL1	16	H'FFFE3A96	8、16
	SSI 时钟选择寄存器	SCSR	16	H'FFFE3AA2	8、16
I/O 端口	端口 A 数据寄存器 L	PADRL	16	H'FFFE3802	8、16
	端口 B 数据寄存器 L	PBDRL	16	H'FFFE3882	8、16
	端口 B 端口寄存器 L	PBPRL	16	H'FFFE389E	8、16
	端口 C 数据寄存器 L	PCDRL	16	H'FFFE3902	8、16
	端口 C 端口寄存器 L	PCPRL	16	H'FFFE391E	8、16
	端口 D 数据寄存器 L	PDDRL	16	H'FFFE3982	8、16
	端口 D 端口寄存器 L	PDPRL	16	H'FFFE399E	8、16
	端口 E 数据寄存器 L	PEDRL	16	H'FFFE3A02	8、16
	端口 E 端口寄存器 L	PEPRL	16	H'FFFE3A1E	8、16
	端口 F 数据寄存器 H	PFDRH	16	H'FFFE3A80	8、16、32
	端口 F 数据寄存器 L	PFDRL	16	H'FFFE3A82	8、16
	端口 F 端口寄存器 H	PFPRH	16	H'FFFE3A9C	8、16、32
	端口 F 端口寄存器 L	PFPRL	16	H'FFFE3A9E	8、16
低功耗模式	待机控制寄存器	STBCR	8	H'FFFE0014	8
	待机控制寄存器 2	STBCR2	8	H'FFFE0018	8
	待机控制寄存器 3	STBCR3	8	H'FFFE0408	8
	待机控制寄存器 4	STBCR4	8	H'FFFE040C	8

模块名称	寄存器名称	简称	位数	地址	存取长度
低功耗模式	待机控制寄存器 5	STBCR5	8	H'FFFE0410	8
	待机控制寄存器 6	STBCR6	8	H'FFFE0414	8
	系统控制寄存器 1	SYSCR1	8	H'FFFE0402	8
	系统控制寄存器 2	SYSCR2	8	H'FFFE0404	8
	系统控制寄存器 3	SYSCR3	8	H'FFFE0418	8
	深度待机控制寄存器	DSCTR	8	H'FFFF2800	8
	深度待机控制寄存器 2	DSCTR2	8	H'FFFF2802	8
	深度待机控制解除源选择寄存器	DSSSR	16	H'FFFF2804	16
	深度待机解除源标志寄存器	DSFR	16	H'FFFF2808	16
	保持用内部 RAM 调整寄存器	DSRTR	8	H'FFFF280C	8
H-UDI	指令寄存器	SDIR	16	H'FFFE2000	16

30.2 寄存器位一览表

模块名称	寄存器 简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
CPG	FRQCR	—	CKOEN2	CKOEN[1]	CKOEN[0]	—	—	STC[1]	STC[0]
		—	—	—	IFC	—	PFC[2]	PFC[1]	PFC[0]
INTC	ICR0	NMIL	—	—	—	—	—	—	NMIE
		—	—	—	—	—	—	—	—
	ICR1	IRQ71S	IRQ70S	IRQ61S	IRQ60S	IRQ51S	IRQ50S	IRQ41S	IRQ40S
		IRQ31S	IRQ30S	IRQ21S	IRQ20S	IRQ11S	IRQ10S	IRQ01S	IRQ00S
	ICR2	—	—	—	—	—	—	—	—
		PINT7S	PINT6S	PINT5S	PINT4S	PINT3S	PINT2S	PINT1S	PINT0S
	IRQRR	—	—	—	—	—	—	—	—
		IRQ7F	IRQ6F	IRQ5F	IRQ4F	IRQ3F	IRQ2F	IRQ1F	IRQ0F
	PINTER	—	—	—	—	—	—	—	—
		PINT7E	PINT6E	PINT5E	PINT4E	PINT3E	PINT2E	PINT1E	PINT0E
	PIRR	—	—	—	—	—	—	—	—
		PINT7R	PINT6R	PINT5R	PINT4R	PINT3R	PINT2R	PINT1R	PINT0R
	IBCR	E15	E14	E13	E12	E11	E10	E9	E8
		E7	E6	E5	E4	E3	E2	E1	—
	IBNR	BE[1]	BE[0]	BOVE	—	—	—	—	—
		—	—	—	—	BN[3]	BN[2]	BN[1]	BN[0]
	IPR01								
	IPR02								
	IPR05								
	IPR06								
	IPR07								
	IPR08								
	IPR09								
	IPR10								

模块名称	寄存器 简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
INTC	IPR11								
	IPR12								
	IPR13								
	IPR14								
	IPR15								
	IPR16								
	IPR17								
UBC	BAR_0	BA31	BA30	BA29	BA28	BA27	BA26	BA25	BA24
		BA23	BA22	BA21	BA20	BA19	BA18	BA17	BA16
		BA15	BA14	BA13	BA12	BA11	BA10	BA9	BA8
		BA7	BA6	BA5	BA4	BA3	BA2	BA1	BA0
	BAMR_0	BAM31	BAM30	BAM29	BAM28	BAM27	BAM26	BAM25	BAM24
		BAM23	BAM22	BAM21	BAM20	BAM19	BAM18	BAM17	BAM16
		BAM15	BAM14	BAM13	BAM12	BAM11	BAM10	BAM9	BAM8
		BAM7	BAM6	BAM5	BAM4	BAM3	BAM2	BAM1	BAM0
	BBR_0	—	—	UBID	DBE	—	—	CP[1]	CP[0]
		CD[1]	CD[0]	ID[1]	ID[0]	RW[1]	RW[0]	SZ[1]	SZ[0]
	BDR_0	BD31	BD30	BD29	BD28	BD27	BD26	BD25	BD24
		BD23	BD22	BD21	BD20	BD19	BD18	BD17	BD16
		BD15	BD14	BD13	BD12	BD11	BD10	BD9	BD8
		BD7	BD6	BD5	BD4	BD3	BD2	BD1	BD0
	BDMR_0	BDM31	BDM30	BDM29	BDM28	BDM27	BDM26	BDM25	BDM24
		BDM23	BDM22	BDM21	BDM20	BDM19	BDM18	BDM17	BDM16
		BDM15	BDM14	BDM13	BDM12	BDM11	BDM10	BDM9	BDM8
		BDM7	BDM6	BDM5	BDM4	BDM3	BDM2	BDM1	BDM0

模块名称	寄存器 简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
UBC	BAR_1	BA31	BA30	BA29	BA28	BA27	BA26	BA25	BA24
		BA23	BA22	BA21	BA20	BA19	BA18	BA17	BA16
		BA15	BA14	BA13	BA12	BA11	BA10	BA9	BA8
		BA7	BA6	BA5	BA4	BA3	BA2	BA1	BA0
	BAMR_1	BAM31	BAM30	BAM29	BAM28	BAM27	BAM26	BAM25	BAM24
		BAM23	BAM22	BAM21	BAM20	BAM19	BAM18	BAM17	BAM16
		BAM15	BAM14	BAM13	BAM12	BAM11	BAM10	BAM9	BAM8
		BAM7	BAM6	BAM5	BAM4	BAM3	BAM2	BAM1	BAM0
	BBR_1	—	—	UBID	DBE	—	—	CP[1]	CP[0]
		CD[1]	CD[0]	ID[1]	ID[0]	RW[1]	RW[0]	SZ[1]	SZ[0]
	BDR_1	BD31	BD30	BD29	BD28	BD27	BD26	BD25	BD24
		BD23	BD22	BD21	BD20	BD19	BD18	BD17	BD16
		BD15	BD14	BD13	BD12	BD11	BD10	BD9	BD8
		BD7	BD6	BD5	BD4	BD3	BD2	BD1	BD0
	BDMR_1	BDM31	BDM30	BDM29	BDM28	BDM27	BDM26	BDM25	BDM24
		BDM23	BDM22	BDM21	BDM20	BDM19	BDM18	BDM17	BDM16
		BDM15	BDM14	BDM13	BDM12	BDM11	BDM10	BDM8	BDM9
		BDM7	BDM6	BDM5	BDM4	BDM3	BDM2	BDM1	BDM0
	BRCCR	—	—	—	—	—	—	—	—
		—	—	—	—	UTOD1	UTOD0	CKS[1]	CKS[0]
		SCMFC0	SCMFC1	SCMFD0	SCMFD1	—	—	—	—
		—	PCB1	PCB0	—	—	—	—	—
高速缓存	CCR1	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	ICF	—	—	ICE
		—	—	—	—	OCF	—	WT	OCE
	CCR2	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	LE
		—	—	—	—	—	—	W3LOAD	W3LOCK
		—	—	—	—	—	—	W2LOAD	W2LOCK

模块 名称	寄存器 简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
BSC	CMNCR	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	BLOCK	DPRTY[1]	DPRTY[0]	DMAIW[2]
		DMAIW[1]	DMAIW[0]	DMAIWA	—	—	—	HIZMEM	HIZCNT
	CS0BCR	—	IWW[2]	IWW[1]	IWW[0]	IWRWD[2]	IWRWD[1]	IWRWD[0]	IWRWS[2]
		IWRWS[1]	IWRWS[0]	IWRRD[2]	IWRRD[1]	IWRRD[0]	IWRRS[2]	IWRRS[1]	IWRRS[0]
		—	TYPE[2]	TYPE[1]	TYPE[0]	ENDIAN	BSZ[1]	BSZ[0]	—
		—	—	—	—	—	—	—	—
	CS1BCR	—	IWW[2]	IWW[1]	IWW[0]	IWRWD[2]	IWRWD[1]	IWRWD[0]	IWRWS[2]
		IWRWS[1]	IWRWS[0]	IWRRD[2]	IWRRD[1]	IWRRD[0]	IWRRS[2]	IWRRS[1]	IWRRS[0]
		—	TYPE[2]	TYPE[1]	TYPE[0]	ENDIAN	BSZ[1]	BSZ[0]	—
		—	—	—	—	—	—	—	—
	CS2BCR	—	IWW[2]	IWW[1]	IWW[0]	IWRWD[2]	IWRWD[1]	IWRWD[0]	IWRWS[2]
		IWRWS[1]	IWRWS[0]	IWRRD[2]	IWRRD[1]	IWRRD[0]	IWRRS[2]	IWRRS[1]	IWRRS[0]
		—	TYPE[2]	TYPE[1]	TYPE[0]	ENDIAN	BSZ[1]	BSZ[0]	—
		—	—	—	—	—	—	—	—
	CS3BCR	—	IWW[2]	IWW[1]	IWW[0]	IWRWD[2]	IWRWD[1]	IWRWD[0]	IWRWS[2]
		IWRWS[1]	IWRWS[0]	IWRRD[2]	IWRRD[1]	IWRRD[0]	IWRRS[2]	IWRRS[1]	IWRRS[0]
		—	TYPE[2]	TYPE[1]	TYPE[0]	ENDIAN	BSZ[1]	BSZ[0]	—
		—	—	—	—	—	—	—	—
	CS4BCR	—	IWW[2]	IWW[1]	IWW[0]	IWRWD[2]	IWRWD[1]	IWRWD[0]	IWRWS[2]
		IWRWS[1]	IWRWS[0]	IWRRD[2]	IWRRD[1]	IWRRD[0]	IWRRS[2]	IWRRS[1]	IWRRS[0]
		—	TYPE[2]	TYPE[1]	TYPE[0]	ENDIAN	BSZ[1]	BSZ[0]	—
		—	—	—	—	—	—	—	—
	CS5BCR	—	IWW[2]	IWW[1]	IWW[0]	IWRWD[2]	IWRWD[1]	IWRWD[0]	IWRWS[2]
		IWRWS[1]	IWRWS[0]	IWRRD[2]	IWRRD[1]	IWRRD[0]	IWRRS[2]	IWRRS[1]	IWRRS[0]
		—	TYPE[2]	TYPE[1]	TYPE[0]	ENDIAN	BSZ[1]	BSZ[0]	—
		—	—	—	—	—	—	—	—

模块名称	寄存器简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
BSC	CS6BCR	—	IWW[2]	IWW[1]	IWW[0]	IWRWD[2]	IWRWD[1]	IWRWD[0]	IWRWS[2]
		IWRWS[1]	IWRWS[0]	IWRRD[2]	IWRRD[1]	IWRRD[0]	IWRRS[2]	IWRRS[1]	IWRRS[0]
		—	TYPE[2]	TYPE[1]	TYPE[0]	ENDIAN	BSZ[1]	BSZ[0]	—
		—	—	—	—	—	—	—	—
	CS7BCR	—	IWW[2]	IWW[1]	IWW[0]	IWRWD[2]	IWRWD[1]	IWRWD[0]	IWRWS[2]
		IWRWS[1]	IWRWS[0]	IWRRD[2]	IWRRD[1]	IWRRD[0]	IWRRS[2]	IWRRS[1]	IWRRS[0]
		—	TYPE[2]	TYPE[1]	TYPE[0]	ENDIAN	BSZ[1]	BSZ[0]	—
		—	—	—	—	—	—	—	—
	CS0WCR*1	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	SW[1]	SW[0]	WR[3]	WR[2]	WR[1]
		WR[0]	WM	—	—	—	—	HW[1]	HW[0]
	CS0WCR*2	—	—	—	—	—	—	—	—
		—	—	BST[1]	BST[0]	—	—	BW[1]	BW[0]
		—	—	—	—	—	W[3]	W[2]	W[1]
		W[0]	WM	—	—	—	—	—	—
	CS0WCR*3	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	BW[1]	BW[0]
		—	—	—	—	—	W[3]	W[2]	W[1]
		W[0]	WM	—	—	—	—	—	—
	CS1WCR*4	—	—	—	—	—	—	—	—
		—	—	—	BAS	—	WW[2]	WW[1]	WW[0]
		—	—	—	SW[1]	SW[0]	WR[3]	WR[2]	WR[1]
		WR[0]	WM	—	—	—	—	HW[1]	HW[0]
	CS2WCR*1	—	—	—	—	—	—	—	—
		—	—	—	BAS	—	—	—	—
		—	—	—	—	—	WR[3]	WR[2]	WR[1]
		WR[0]	WM	—	—	—	—	—	—
	CS2WCR*2	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	A2CL1
		A2CL0	—	—	—	—	—	—	—

模块名称	寄存器简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
BSC	CS3WCR*1	—	—	—	—	—	—	—	—
		—	—	—	BAS	—	—	—	—
		—	—	—	—	—	WR[3]	WR[2]	WR[1]
		WR[0]	WM	—	—	—	—	—	—
	CS3WCR*5	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	WTRP[1]	WTRP[0]	—	WTRCD[1]	WTRCD[0]	—	A3CL1
		A3CL0	—	—	TRWL[1]	TRWL[0]	—	WTRC[1]	WTRC[0]
	CS4WCR*1	—	—	—	—	—	—	—	—
		—	—	—	BAS	—	WW[2]	WW[1]	WW[0]
		—	—	—	SW[1]	SW[0]	WR[3]	WR[2]	WR[1]
		WR[0]	WM	—	—	—	—	HW[1]	HW[0]
	CS4WCR*2	—	—	—	—	—	—	—	—
		—	—	BST[1]	BST[0]	—	—	BW[1]	BW[0]
		—	—	—	SW[1]	SW[0]	W[3]	W[2]	W[1]
		W[0]	WM	—	—	—	—	HW[1]	HW[0]
	CS5WCR*1	—	—	—	—	—	—	—	—
		—	—	SZSEL	MPXW/ BAS	—	WW[2]	WW[1]	WW[0]
		—	—	—	SW[1]	SW[0]	WR[3]	WR[2]	WR[1]
		WR[0]	WM	—	—	—	—	HW[1]	HW[0]
	CS5WCR*6	—	—	—	—	—	—	—	—
		—	—	SA[1]	SA[0]	—	—	—	—
		—	TED[3]	TED[2]	TED[1]	TED[0]	PCW[1]	PCW[0]	PCW[1]
		PCW[0]	WM	—	—	TEH[3]	TEH[2]	TEH[1]	TEH[0]
	CS6WCR*1	—	—	—	—	—	—	—	—
		—	—	—	BAS	—	—	—	—
		—	—	—	SW[1]	SW[0]	WR[3]	WR[2]	WR[1]
		WR[0]	WM	—	—	—	—	HW[1]	HW[0]
	CS6WCR*7	—	—	—	—	—	—	—	—
		—	—	MPXAW[1]	MPXAW[0]	MPXMD	—	BW[1]	BW[0]
		—	—	—	—	—	W[3]	W[2]	W[1]
		W[0]	WM	—	—	—	—	—	—

模块名称	寄存器 简称	bit 31/23/15/ 7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
BSC	CS6WCR*6	—	—	—	—	—	—	—	—
		—	—	SA[1]	SA[0]	—	—	—	—
		—	TED[3]	TED[2]	TED[1]	TED[0]	PCW[1]	PCW[0]	PCW[1]
		PCW[0]	WM	—	—	TEH[3]	TEH[2]	TEH[1]	TEH[0]
	CS7WCR*4	—	—	—	—	—	—	—	—
		—	—	—	BAS	—	WW[2]	WW[1]	WW[0]
		—	—	—	SW[1]	SW[0]	WR[3]	WR[2]	WR[1]
		WR[0]	WM	—	—	—	—	HW[1]	HW[0]
	SDCR	—	—	—	—	—	—	—	—
		—	—	—	A2ROW[1]	A2ROW[0]	—	A2COL[1]	A2COL[0]
		—	—	DEEP	SLOW	RFSH	RMODE	PDOWN	BACTV
		—	—	—	A3ROW[1]	A3ROW[0]	—	A3COL[1]	A3COL[0]
	RTCSR	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		CMF	CMIE	CKS[2]	CKS[1]	CKS[0]	RRC[2]	RRC[1]	RRC[0]
	RTCNT	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
	RTCOR	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—

模块名称	寄存器简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
DMAC	SAR0								
	DAR0								
	DMATCR0	—	—	—	—	—	—	—	—
	CHCR0	TC	—	RLDSAR	RLDDAR	—	—	—	—
		DO	TL	—	TEMASK	HE	HIE	AM	AL
		DM[1]	DM[0]	SM[1]	SM[0]	RS[3]	RS[2]	RS[1]	RS[0]
		DL	DS	TB	TS[1]	TS[0]	IE	TE	DE
	RSAR0								
	RDAR0								
	RDMATCR0	—	—	—	—	—	—	—	—
	SAR1								
	DAR1								

模块名称	寄存器简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
DMAC	DMATCR1	—	—	—	—	—	—	—	—
	CHCR1	TC	—	RLDSAR	RLDDAR	—	—	—	—
		DO	TL	—	TEMASK	HE	HIE	AM	AL
		DM[1]	DM[0]	SM[1]	SM[0]	RS[3]	RS[2]	RS[1]	RS[0]
		DL	DS	TB	TS[1]	TS[0]	IE	TE	DE
	RSAR1								
	RDAR1								
	RDMATCR1	—	—	—	—	—	—	—	—
	SAR2								
	DAR2								
	DMATCR2	—	—	—	—	—	—	—	—
	CHCR2	TC	—	RLDSAR	RLDDAR	—	—	—	—
		DO	—	—	TEMASK	HE	HIE	AM	AL
		DM[1]	DM[0]	SM[1]	SM[0]	RS[3]	RS[2]	RS[1]	RS[0]
		DL	DS	TB	TS[1]	TS[0]	IE	TE	DE

模块名称	寄存器简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
DMAC	RSAR2								
	RDAR2								
	RDMATCR2	—	—	—	—	—	—	—	—
	SAR3								
	DAR3								
	DMATCR3	—	—	—	—	—	—	—	—
	CHCR3	TC	—	RLDSAR	RLDDAR	—	—	—	—
		DO	—	—	TEMASK	HE	HIE	AM	AL
		DM[1]	DM[0]	SM[1]	SM[0]	RS[3]	RS[2]	RS[1]	RS[0]
		DL	DS	TB	TS[1]	TS[0]	IE	TE	DE
	RSAR3								
	RDAR3								

模块名称	寄存器简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
DMAC	RDMATCR3	—	—	—	—	—	—	—	—
	SAR4								
	DAR4								
	DMATCR4	—	—	—	—	—	—	—	—
	CHCR4	TC	—	RLDSAR	RLDDAR	—	—	—	—
		—	—	—	TEMASK	HE	HIE	—	—
		DM[1]	DM[0]	SM[1]	SM[0]	RS[3]	RS[2]	RS[1]	RS[0]
		—	—	TB	TS[1]	TS[0]	IE	TE	DE
	RSAR4								
	RDAR4								
	RDMATCR4	—	—	—	—	—	—	—	—
	SAR5								

模块名称	寄存器简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
DMAC	DAR5								
	DMATCR5	—	—	—	—	—	—	—	—
	CHCR5	TC	—	RLDSAR	RLDDAR	—	—	—	—
		—	—	—	TEMASK	HE	HIE	—	—
		DM[1]	DM[0]	SM[1]	SM[0]	RS[3]	RS[2]	RS[1]	RS[0]
		—	—	TB	TS[1]	TS[0]	IE	TE	DE
	RSAR5								
	RDAR5								
	RDMATCR5	—	—	—	—	—	—	—	—
	SAR6								
	DAR6								
	DMATCR6	—	—	—	—	—	—	—	—

模块名称	寄存器简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
DMAC	CHCR6	TC	—	RLDSAR	RLDDAR	—	—	—	—
		—	—	—	TEMASK	HE	HIE	—	—
		DM[1]	DM[0]	SM[1]	SM[0]	RS[3]	RS[2]	RS[1]	RS[0]
		—	—	TB	TS[1]	TS[0]	IE	TE	DE
	RSAR6								
	RDAR6								
	RDMATCR6	—	—	—	—	—	—	—	—
	SAR7								
	DAR7								
	DMATCR7	—	—	—	—	—	—	—	—
	CHCR7	TC	—	RLDSAR	RLDDAR	—	—	—	—
		—	—	—	TEMASK	HE	HIE	—	—
		DM[1]	DM[0]	SM[1]	SM[0]	RS[3]	RS[2]	RS[1]	RS[0]
		—	—	TB	TS[1]	TS[0]	IE	TE	DE
	RSAR7								

模块名称	寄存器简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
DMAC	RDAR7								
	RDMATCR7	—	—	—	—	—	—	—	—
	DMAOR	—	—	CMS[1]	CMS[0]	—	—	PR[1]	PR[0]
		—	—	—	—	—	AE	NMIF	DME
	DMARS0	CH1MID[5]	CH1MID[4]	CH1MID[3]	CH1MID[2]	CH1MID[1]	CH1MID[0]	CH1RID[1]	CH1RID[0]
		CH0MID[5]	CH0MID[4]	CH0MID[3]	CH0MID[2]	CH0MID[1]	CH0MID[0]	CH0RID[1]	CH0RID[0]
	DMARS1	CH3MID[5]	CH3MID[4]	CH3MID[3]	CH3MID[2]	CH3MID[1]	CH3MID[0]	CH3RID[1]	CH3RID[0]
		CH2MID[5]	CH2MID[4]	CH2MID[3]	CH2MID[2]	CH2MID[1]	CH2MID[0]	CH2RID[1]	CH2RID[0]
	DMARS2	CH5MID[5]	CH5MID[4]	CH5MID[3]	CH5MID[2]	CH5MID[1]	CH5MID[0]	CH5RID[1]	CH5RID[0]
		CH4MID[5]	CH4MID[4]	CH4MID[3]	CH4MID[2]	CH4MID[1]	CH4MID[0]	CH4RID[1]	CH4RID[0]
	DMARS3	CH7MID[5]	CH7MID[4]	CH7MID[3]	CH7MID[2]	CH7MID[1]	CH7MID[0]	CH7RID[1]	CH7RID[0]
		CH6MID[5]	CH6MID[4]	CH6MID[3]	CH6MID[2]	CH6MID[1]	CH6MID[0]	CH6RID[1]	CH6RID[0]
MTU2	TCR_0	CCLR[2]	CCLR[1]	CCLR[0]	CKEG[1]	CKEG[0]	TPSC[2]	TPSC[1]	TPSC[0]
	TMDR_0	—	BFE	BFB	BFA	MD[3]	MD[2]	MD[1]	MD[0]
	TIORH_0	IOB[3]	IOB[2]	IOB[1]	IOB[0]	IOA[3]	IOA[2]	IOA[1]	IOA[0]
	TIORL_0	IOD[3]	IOD[2]	IOD[1]	IOD[0]	IOC[3]	IOC[2]	IOC[1]	IOC[0]
	TIER_0	TTGE	—	—	TCIEV	TGIED	TGIEC	TGIEB	TGIEA
	TSR_0	TCFD	—	—	TCFV	TGFD	TGFC	TGFB	TGFA
	TCNT_0								
	TGRA_0								
	TGRB_0								
	TGRC_0								
	TGRD_0								
	TGRE_0								

模块名称	寄存器简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
MTU2	TGRF_0								
	TIER2_0	TTGE2	—	—	—	—	—	TGIEF	TGIEE
	TSR2_0	—	—	—	—	—	—	TGFF	TGFE
	TBTM_0	—	—	—	—	—	TTSE	TTSB	TTSA
	TCR_1	—	CCLR[1]	CCLR[0]	CKEG[1]	CKEG[0]	TPSC[2]	TPSC[1]	TPSC[0]
	TMDR_1	—	—	—	—	MD[3]	MD[2]	MD[1]	MD[0]
	TIOR_1	IOB[3]	IOB[2]	IOB[1]	IOB[0]	IOA[3]	IOA[2]	IOA[1]	IOA[0]
	TIER_1	TTGE	—	TCIEU	TCIEV	—	—	TGIEB	TGIEA
	TSR_1	TCFD	—	TCFU	TCFV	TGFD	TGFC	TGFB	TGFA
	TCNT_1								
	TGRA_1								
	TGRB_1								
	TICCR	—	—	—	—	I2BE	I2AE	I1BE	I1AE
	TCR_2	—	CCLR[1]	CCLR[0]	CKEG[1]	CKEG[0]	TPSC[2]	TPSC[1]	TPSC[0]
	TMDR_2	—	—	—	—	MD[3]	MD[2]	MD[1]	MD[0]
	TIOR_2	IOB[3]	IOB[2]	IOB[1]	IOB[0]	IOA[3]	IOA[2]	IOA[1]	IOA[0]
	TIER_2	TTGE	—	TCIEU	TCIEV	—	—	TGIEB	TGIEA
	TSR_2	TCFD	—	TCFU	TCFV	TGFD	TGFC	TGFB	TGFA
	TCNT_2								
	TGRA_2								
	TGRB_2								
	TCR_3	CCLR[2]	CCLR[1]	CCLR[0]	CKEG[1]	CKEG[0]	TPSC[2]	TPSC[1]	TPSC[0]
	TMDR_3	—	—	BFB	BFA	MD[3]	MD[2]	MD[1]	MD[0]
	TIORH_3	IOB[3]	IOB[2]	IOB[1]	IOB[0]	IOA[3]	IOA[2]	IOA[1]	IOA[0]
	TIORL_3	IOD[3]	IOD[2]	IOD[1]	IOD[0]	IOC[3]	IOC[2]	IOC[1]	IOC[0]
	TIER_3	TTGE	—	—	TCIEV	TGIED	TGIEC	TGIEB	TGIEA
	TSR_3	TCFD	—	—	TCFV	TGFD	TGFC	TGFB	TGFA
	TCNT_3								

模块名称	寄存器简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
MTU2	TGRA_3								
	TGRB_3								
	TGRC_3								
	TGRD_3								
	TBTM_3	—	—	—	—	—	—	TTSB	TTSA
	TCR_4	CCLR[2]	CCLR[1]	CCLR[0]	CKEG[1]	CKEG[0]	TPSC[2]	TPSC[1]	TPSC[0]
	TMDR_4	—	—	BFB	BFA	MD[3]	MD[2]	MD[1]	MD[0]
	TIORH_4	IOB[3]	IOB[2]	IOB[1]	IOB[0]	IOA[3]	IOA[2]	IOA[1]	IOA[0]
	TIORL_4	IOD[3]	IOD[2]	IOD[1]	IOD[0]	IOC[3]	IOC[2]	IOC[1]	IOC[0]
	TIER_4	TTGE	TTGE2	—	TCIEV	TGIED	TGIEC	TGIEB	TGIEA
	TSR_4	TCFD	—	—	TCFV	TGFD	TGFC	TGFB	TGFA
	TCNT_4								
	TGRA_4								
	TGRB_4								
	TGRC_4								
	TGRD_4								
	TBTM_4	—	—	—	—	—	—	TTSB	TTSA
	TADCR	BF[1]	BF[0]	—	—	—	—	—	—
		UT4AE	DT4AE	UT4BE	DT4BE	ITA3AE	ITA4VE	ITB3AE	ITB4VE
	TADCORA_4								
	TADCORB_4								
	TADCOBRA_4								
	TADCOBRB_4								

模块名称	寄存器简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
MTU2	TSTR	CST4	CST3	—	—	—	CST2	CST1	CST0
	TSYR	SYNC4	SYNC3	—	—	—	SYNC2	SYNC1	SYNC0
	TRWER	—	—	—	—	—	—	—	RWE
	TOER	—	—	OE4D	OE4C	OE3D	OE4B	OE4A	OE3B
	TOCR1	—	PSYE	—	—	TOCL	TOCS	OLSN	PLSP
	TOCR2	BF[1]	BF[0]	OLS3N	OLS3P	OLS2N	OLS2P	OLS1N	OLS1P
	TGCR	—	BDC	N	P	FB	WF	VF	UF
	TCDR								
	TDDR								
	TCNTS								
	TCBR								
	TITCR	T3AEN	3ACOR[2]	3ACOR[1]	3ACOR[0]	T4VEN	4VCOR[2]	4VCOR[1]	4VCOR[0]
	TITCNT	—	3ACNT[2]	3ACNT[1]	3ACNT[0]	—	4VCNT[2]	4VCNT[1]	4VCNT[0]
	TBTER	—	—	—	—	—	—	BTE[1]	BTE[0]
	TDER	—	—	—	—	—	—	—	TDER
	TWCR	CCE	—	—	—	—	—	—	WRE
	TOLBR	—	—	OLS3N	OLS3P	OLS2N	OLS2P	OLS1N	OLS1P
CMT	CMSTR	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	STR1	STR0
	CMCSR0	—	—	—	—	—	—	—	—
		CMF	CMIE	—	—	—	—	CKS[1]	CKS[0]
	CMCNT0								
	CMCOR0								
	CMCSR1	—	—	—	—	—	—	—	—
		CMF	CMIE	—	—	—	—	CKS[1]	CKS[0]
	CMCNT1								
	CMCOR1								

模块名称	寄存器简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
WDT	WTCNT								
	WTCSR	IOVF	WT/IT	TME	—	—	CKS[2]	CKS[1]	CKS[0]
	WRCSR	WOVF	RSTE	RSTS	—	—	—	—	—
RTC	R64CNT	—	1Hz	2Hz	4Hz	8Hz	16Hz	32Hz	64Hz
	RSECCNT	—	10 秒 [2]	10 秒 [1]	10 秒 [0]	1 秒 [3]	1 秒 [2]	1 秒 [1]	1 秒 [0]
	RMINCNT	—	10 分 [2]	10 分 [1]	10 分 [0]	1 分 [3]	1 分 [2]	1 分 [1]	1 分 [0]
	RHRCNT	—	—	10 小时 [1]	10 小时 [0]	1 小时 [3]	1 小时 [2]	1 小时 [1]	1 小时 [0]
	RWKCNT	—	—	—	—	—	星期 [2]	星期 [1]	星期 [0]
	RDAYCNT	—	—	10 日 [1]	10 日 [0]	1 日 [3]	1 日 [2]	1 日 [1]	1 日 [0]
	RMONCNT	—	—	—	10 月	1 月 [3]	1 月 [2]	1 月 [1]	1 月 [0]
	RYRCNT	1000 年 [3]	1000 年 [2]	1000 年 [1]	1000 年 [0]	100 年 [3]	100 年 [2]	100 年 [1]	100 年 [0]
		10 年 [3]	10 年 [2]	10 年 [1]	10 年 [0]	1 年 [3]	1 年 [2]	1 年 [1]	1 年 [0]
	RSECAR	ENB	10 秒 [2]	10 秒 [1]	10 秒 [0]	1 秒 [3]	1 秒 [2]	1 秒 [1]	1 秒 [0]
	RMINAR	ENB	10 分 [2]	10 分 [1]	10 分 [0]	1 分 [3]	1 分 [2]	1 分 [1]	1 分 [0]
	RHRAR	ENB	—	10 小时 [1]	10 小时 [0]	1 小时 [3]	1 小时 [2]	1 小时 [1]	1 小时 [0]
	RWKAR	ENB	—	—	—	—	星期 [2]	星期 [1]	星期 [0]
	RDAYAR	ENB	—	10 日 [1]	10 日 [0]	1 日 [3]	1 日 [2]	1 日 [1]	1 日 [0]
	RMONAR	ENB	—	—	10 月	1 月 [3]	1 月 [2]	1 月 [1]	1 月 [0]
	RYRAR	1000 年 [3]	1000 年 [2]	1000 年 [1]	1000 年 [0]	100 年 [3]	100 年 [2]	100 年 [1]	100 年 [0]
		10 年 [3]	10 年 [2]	10 年 [1]	10 年 [0]	1 年 [3]	1 年 [2]	1 年 [1]	1 年 [0]
	RCR1	CF	—	—	CIE	AIE	—	—	AF
	RCR2	PEF	PES[2]	PES[1]	PES[0]	RTCEN	ADJ	RESET	START
	RCR3	ENB	—	—	—	—	—	—	—
SCIF	SCSMR_0	—	—	—	—	—	—	—	—
		C/A	CHR	PE	O/E	STOP	—	CKS[1]	CKS[0]
	SCBRR_0								
	SCSCR_0	—	—	—	—	—	—	—	—
		TIE	RIE	TE	RE	REIE	—	CKE[1]	CKE[0]
	SCFTDR_0								
	SCFSR_0	PER[3]	PER[2]	PER[1]	PER[0]	FER[3]	FER[2]	FER[1]	FER[0]
		ER	TEND	TDFE	BRK	FER	PER	RDF	DR
	SCFRDR_0								
	SCFCR_0	—	—	—	—	—	RSTRG[2]	RSTRG[1]	RSTRG[0]
		RTRG[1]	RTRG[0]	TTRG[1]	TTRG[0]	MCE	TFRST	RFRST	LOOP
	SCFDR_0	—	—	—	T[4]	T[3]	T[2]	T[1]	T[0]
		—	—	—	R[4]	R[3]	R[2]	R[1]	R[0]

模块名称	寄存器简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
SCIF	SCSPTR_0	—	—	—	—	—	—	—	—
		—	—	—	—	SCKIO	SCKDT	SPB2IO	SPB2DT
	SCLSR_0	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	ORER
	SCEMR_0	—	—	—	—	—	—	—	—
		BGDM	—	—	—	—	—	—	ABCS
	SCSMR_1	—	—	—	—	—	—	—	—
		C/A	CHR	PE	O/E	STOP	—	CKS[1]	CKS[0]
	SCBRR_1								
	SCSCR_1	—	—	—	—	—	—	—	—
		TIE	RIE	TE	RE	REIE	—	CKE[1]	CKE[0]
	SCFTDR_1								
	SCFSR_1	PER[3]	PER[2]	PER[1]	PER[0]	FER[3]	FER[2]	FER[1]	FER[0]
		ER	TEND	TDFE	BRK	FER	PER	RDF	DR
	SCFRDR_1								
	SCFCR_1	—	—	—	—	—	RSTRG[2]	RSTRG[1]	RSTRG[0]
		RTRG[1]	RTRG[0]	TTRG[1]	TTRG[0]	MCE	TFRST	RFRST	LOOP
	SCFDR_1	—	—	—	T[4]	T[3]	T[2]	T[1]	T[0]
		—	—	—	R[4]	R[3]	R[2]	R[1]	R[0]
	SCSPTR_1	—	—	—	—	—	—	—	—
		—	—	—	—	SCKIO	SCKDT	SPB2IO	SPB2DT
	SCLSR_1	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	ORER
	SCEMR_1	—	—	—	—	—	—	—	—
		BGDM	—	—	—	—	—	—	ABCS
	SCSMR_2	—	—	—	—	—	—	—	—
		C/A	CHR	PE	O/E	STOP	—	CKS[1]	CKS[0]
	SCBRR_2								
	SCSCR_2	—	—	—	—	—	—	—	—
		TIE	RIE	TE	RE	REIE	—	CKE[1]	CKE[0]
	SCFTDR_2								
	SCFSR_2	PER[3]	PER[2]	PER[1]	PER[0]	FER[3]	FER[2]	FER[1]	FER[0]
		ER	TEND	TDFE	BRK	FER	PER	RDF	DR
	SCFRDR_2								
	SCFCR_2	—	—	—	—	—	RSTRG[2]	RSTRG[1]	RSTRG[0]
		RTRG[1]	RTRG[0]	TTRG[1]	TTRG[0]	MCE	TFRST	RFRST	LOOP

模块名称	寄存器简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
SCIF	SCFDR_2	—	—	—	T[4]	T[3]	T[2]	T[1]	T[0]
		—	—	—	R[4]	R[3]	R[2]	R[1]	R[0]
	SCSPTR_2	—	—	—	—	—	—	—	—
		—	—	—	—	SCKIO	SCKDT	SPB2IO	SPB2DT
	SCLSR_2	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	ORER
	SCEMR_2	—	—	—	—	—	—	—	—
		BGDM	—	—	—	—	—	—	ABCS
	SCSMR_3	—	—	—	—	—	—	—	—
		C/A	CHR	PE	O/E	STOP	—	CKS[1]	CKS[0]
	SCBRR_3								
	SCSCR_3	—	—	—	—	—	—	—	—
		TIE	RIE	TE	RE	REIE	—	CKE[1]	CKE[0]
	SCFTDR_3								
	SCFSR_3	PER[3]	PER[2]	PER[1]	PER[0]	FER[3]	FER[2]	FER[1]	FER[0]
		ER	TEND	TDFE	BRK	FER	PER	RDF	DR
	SCFRDR_3								
	SCFCR_3	—	—	—	—	—	RSTRG[2]	RSTRG[1]	RSTRG[0]
		RTRG[1]	RTRG[0]	TTRG[1]	TTRG[0]	MCE	TFRST	RFRST	LOOP
	SCFDR_3	—	—	—	T[4]	T[3]	T[2]	T[1]	T[0]
		—	—	—	R[4]	R[3]	R[2]	R[1]	R[0]
	SCSPTR_3	—	—	—	—	—	—	—	—
		RTSIO	RTSDT	CTSIO	CTSDT	SCKIO	SCKDT	SPB2IO	SPB2DT
	SCLSR_3	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	ORER
	SCEMR_3	—	—	—	—	—	—	—	—
		BGDM	—	—	—	—	—	—	ABCS
SSU	SSCRH_0	MSS	BIDE	—	SOL	SOLP	—	CSS[1]	CSS[0]
	SSCRL_0	—	SSUMS	SRES	—	—	—	DATS[1]	DATS[0]
	SSMR_0	MLS	CPOS	CPHS	—	—	CKS[2]	CKS[1]	CKS[0]
	SSER_0	TE	RE	—	—	TEIE	TIE	RIE	CEIE
	SSSR_0	—	ORER	—	—	TEND	TDRE	RDRF	CE
	SSCR2_0	—	—	—	TENDSTS	SCSATS	SSODTS	—	—
	SSTD0_0								
	SSTD1_0								
	SSTD2_0								
	SSTD3_0								

模块名称	寄存器简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
SSU	SSRDR0_0								
	SSRDR1_0								
	SSRDR2_0								
	SSRDR3_0								
	SSCRH_1	MSS	BIDE	—	SOL	SOLP	—	CSS[1]	CSS[0]
	SSCRL_1	—	SSUMS	SRES	—	—	—	DATS[1]	DATS[0]
	SSMR_1	MLS	CPOS	CPHS	—	—	CKS[2]	CKS[1]	CKS[0]
	SSER_1	TE	RE	—	—	TEIE	TIE	RIE	CEIE
	SSSR_1	—	ORER	—	—	TEND	TDRE	RDRF	CE
	SSCR2_1	—	—	—	TENDSTS	SCSATS	SSODTS	—	—
	SSTDR0_1								
	SSTDR1_1								
	SSTDR2_1								
	SSTDR3_1								
	SSRDR0_1								
	SSRDR1_1								
	SSRDR2_1								
	SSRDR3_1								
IIC3	ICCR1_0	ICE	RCVD	MST	TRS	CKS[3]	CKS[2]	CKS[1]	CKS[0]
	ICCR2_0	BBSY	SCP	SDAO	SDAOP	SCL	—	IICRST	—
	ICMR_0	MLS	—	—	—	BCWP	BC[2]	BC[1]	BC[0]
	ICIER_0	TIE	TEIE	RIE	NAKIE	STIE	ACKE	ACKBR	ACKBT
	ICSR_0	TDRE	TEND	RDRF	NACKF	STOP	AL/OVE	AAS	ADZ
	SAR_0	SVA[6]	SVA[5]	SVA[4]	SVA[3]	SVA[2]	SVA[1]	SVA[0]	FS
	ICDRT_0								
	ICDRR_0								
	NF2CYC_0	—	—	—	—	—	—	PRS	NF2CYC
	ICCR1_1	ICE	RCVD	MST	TRS	CKS[3]	CKS[2]	CKS[1]	CKS[0]
	ICCR2_1	BBSY	SCP	SDAO	SDAOP	SCL	—	IICRST	—
	ICMR_1	MLS	—	—	—	BCWP	BC[2]	BC[1]	BC[0]
	ICIER_1	TIE	TEIE	RIE	NAKIE	STIE	ACKE	ACKBR	ACKBT
	ICSR_1	TDRE	TEND	RDRF	NACKF	STOP	AL/OVE	AAS	ADZ
	SAR_1	SVA[6]	SVA[5]	SVA[4]	SVA[3]	SVA[2]	SVA[1]	SVA[0]	FS
	ICDRT_1								
	ICDRR_1								
	NF2CYC_1	—	—	—	—	—	—	PRS	NF2CYC

模块名称	寄存器简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
IIC3	ICCR1_2	ICE	RCVD	MST	TRS	CKS[3]	CKS[2]	CKS[1]	CKS[0]
	ICCR2_2	BBSY	SCP	SDAO	SDAOP	SCL	—	IICRST	—
	ICMR_2	MLS	—	—	—	BCWP	BC[2]	BC[1]	BC[0]
	ICIER_2	TIE	TEIE	RIE	NAKIE	STIE	ACKE	ACKBR	ACKBT
	ICSR_2	TDRE	TEND	RDRF	NACKF	STOP	AL/OVE	AAS	ADZ
	SAR_2	SVA[6]	SVA[5]	SVA[4]	SVA[3]	SVA[2]	SVA[1]	SVA[0]	FS
	ICDRT_2								
	ICDRR_2								
	NF2CYC_2	—	—	—	—	—	—	PRS	NF2CYC
	ICCR1_3	ICE	RCVD	MST	TRS	CKS[3]	CKS[2]	CKS[1]	CKS[0]
	ICCR2_3	BBSY	SCP	SDAO	SDAOP	SCL	—	IICRST	—
	ICMR_3	MLS	—	—	—	BCWP	BC[2]	BC[1]	BC[0]
	ICIER_3	TIE	TEIE	RIE	NAKIE	STIE	ACKE	ACKBR	ACKBT
	ICSR_3	TDRE	TEND	RDRF	NACKF	STOP	AL/OVE	AAS	ADZ
	SAR_3	SVA[6]	SVA[5]	SVA[4]	SVA[3]	SVA[2]	SVA[1]	SVA[0]	FS
	ICDRT_3								
	ICDRR_3								
	NF2CYC_3	—	—	—	—	—	—	PRS	NF2CYC
SSI	SSICR_0	—	—	—	DMEN	UIEN	OIEN	IIEN	DIEN
		CHNL[1]	CHNL[0]	DWL[2]	DWL[1]	DWL[0]	SWL[2]	SWL[1]	SWL[0]
		SCKD	SWSD	SCKP	SWSP	SPDP	SDTA	PDTA	DEL
		BREN	CKDV[2]	CKDV[1]	CKDV[0]	MUEN	CPEN	TRMD	EN
	SSISR_0	—	—	—	DMRQ	UIRQ	OIRQ	IIRQ	DIRQ
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	CHNO1	CHNO0	SWNO	IDST
	SSITDR_0								
	SSIRDR_0								

模块名称	寄存器简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
SSI	SSICR_1	—	—	—	DMEN	UIEN	OIEN	IIEN	DIEN
		CHNL[1]	CHNL[0]	DWL[2]	DWL[1]	DWL[0]	SWL[2]	SWL[1]	SWL[0]
		SCKD	SWSD	SCKP	SWSP	SPDP	SDTA	PDTA	DEL
		BREN	CKDV[2]	CKDV[1]	CKDV[0]	MUEN	CPEN	TRMD	EN
	SSISR_1	—	—	—	DMRQ	UIRQ	OIRQ	IIRQ	DIRQ
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	CHNO1	CHNO0	SWNO	IDST
	SSITDR_1								
	SSIRDR_1								
	SSICR_2	—	—	—	DMEN	UIEN	OIEN	IIEN	DIEN
		CHNL[1]	CHNL[0]	DWL[2]	DWL[1]	DWL[0]	SWL[2]	SWL[1]	SWL[0]
		SCKD	SWSD	SCKP	SWSP	SPDP	SDTA	PDTA	DEL
		BREN	CKDV[2]	CKDV[1]	CKDV[0]	MUEN	CPEN	TRMD	EN
	SSISR_2	—	—	—	DMRQ	UIRQ	OIRQ	IIRQ	DIRQ
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	CHNO1	CHNO0	SWNO	IDST
	SSITDR_2								
	SSIRDR_2								

模块名称	寄存器简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
SSI	SSICR_3	—	—	—	DMEN	UIEN	OIEN	IEN	DIEN
		CHNL[1]	CHNL[0]	DWL[2]	DWL[1]	DWL[0]	SWL[2]	SWL[1]	SWL[0]
		SCKD	SWSD	SCKP	SWSP	SPDP	SDTA	PDTA	DEL
		BREN	CKDV[2]	CKDV[1]	CKDV[0]	MUEN	CPEN	TRMD	EN
	SSISR_3	—	—	—	DMRQ	UIRQ	OIRQ	IIRQ	DIRQ
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	CHNO1	CHNO0	SWNO	IDST
	SSITDR_3								
	SSIRDR_3								
RCAN-TL1	MCR_0	MCR15	MCR14	—	—	—	TST[2]	TST[1]	TST[0]
		MCR7	MCR6	MCR5	—	—	MCR2	MCR1	MCR0
	GSR_0	—	—	—	—	—	—	—	—
		—	—	GSR5	GSR4	GSR3	GSR2	GSR1	GSR0
	BCR1_0	TSEG1[3]	TSEG1[2]	TSEG1[1]	TSEG1[0]	—	TSEG2[2]	TSEG2[1]	TSEG2[0]
		—	—	SJW[1]	SJW[0]	—	—	—	BSP
	BCR0_0	—	—	—	—	—	—	—	—
		BRP[7]	BRP[6]	BRP[5]	BRP[4]	BRP[3]	BRP[2]	BRP[1]	BRP[0]
	IRR_0	IRR15	IRR14	IRR13	IRR12	IRR11	IRR10	IRR9	IRR8
		IRR7	IRR6	IRR5	IRR4	IRR3	IRR2	IRR1	IRR0
	IMR_0	IMR15	IMR14	IMR13	IMR12	IMR11	IMR10	IMR9	IMR8
		IMR7	IMR6	IMR5	IMR4	IMR3	IMR2	IMR1	IMR0
	TEC_REC_0	TEC[7]	TEC[6]	TEC[5]	TEC[4]	TEC[3]	TEC[2]	TEC[1]	TEC[0]
		REC[7]	REC[6]	REC[5]	REC[4]	REC[3]	REC[2]	REC[1]	REC[0]
	TXPR1_0	TXPR1[15]	TXPR1[14]	TXPR1[13]	TXPR1[12]	TXPR1[11]	TXPR1[10]	TXPR1[9]	TXPR1[8]
		TXPR1[7]	TXPR1[6]	TXPR1[5]	TXPR1[4]	TXPR1[3]	TXPR1[2]	TXPR1[1]	TXPR1[0]
	TXPR0_0	TXPR0[15]	TXPR0[14]	TXPR0[13]	TXPR0[12]	TXPR0[11]	TXPR0[10]	TXPR0[9]	TXPR0[8]
		TXPR0[7]	TXPR0[6]	TXPR0[5]	TXPR0[4]	TXPR0[3]	TXPR0[2]	TXPR0[1]	—

模块名称	寄存器简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
RCAN-TL1	TXCR1_0	TXCR1[15]	TXCR1[14]	TXCR1[13]	TXCR1[12]	TXCR1[11]	TXCR1[10]	TXCR1[9]	TXCR1[8]
		TXCR1[7]	TXCR1[6]	TXCR1[5]	TXCR1[4]	TXCR1[3]	TXCR1[2]	TXCR1[1]	TXCR1[0]
	TXCR0_0	TXCR0[15]	TXCR0[14]	TXCR0[13]	TXCR0[12]	TXCR0[11]	TXCR0[10]	TXCR0[9]	TXCR0[8]
		TXCR0[7]	TXCR0[6]	TXCR0[5]	TXCR0[4]	TXCR0[3]	TXCR0[2]	TXCR0[1]	—
	TXACK1_0	TXACR1[15]	TXACR1[14]	TXACR1[13]	TXACR1[12]	TXACR1[11]	TXACR1[10]	TXACR1[9]	TXACR1[8]
		TXACR1[7]	TXACR1[6]	TXACR1[5]	TXACR1[4]	TXACR1[3]	TXACR1[2]	TXACR1[1]	TXACR1[0]
	TXACK0_0	TXACK0[15]	TXACK0[14]	TXACK0[13]	TXACK0[12]	TXACK0[11]	TXACK1[10]	TXACK0[9]	TXACK0[8]
		TXACK0[7]	TXACK0[6]	TXACK0[5]	TXACK0[4]	TXACK0[3]	TXACK0[2]	TXACK0[1]	—
	ABACK1_0	ABACK1[15]	ABACK1[14]	ABACK1[13]	ABACK1[12]	ABACK1[11]	ABACK1[10]	ABACK1[9]	ABACK1[8]
		ABACK1[7]	ABACK1[6]	ABACK1[5]	ABACK1[4]	ABACK1[3]	ABACK1[2]	ABACK1[1]	ABACK1[0]
	ABACK0_0	ABACK0[15]	ABACK0[14]	ABACK0[13]	ABACK0[12]	ABACK0[11]	ABACK0[10]	ABACK0[9]	ABACK0[8]
		ABACK0[7]	ABACK0[6]	ABACK0[5]	ABACK0[4]	ABACK0[3]	ABACK0[2]	ABACK0[1]	ABACK0[0]
	RXPR1_0	RXPR1[15]	RXPR1[14]	RXPR1[13]	RXPR1[12]	RXPR1[11]	RXPR1[10]	RXPR1[9]	RXPR1[8]
		RXPR1[7]	RXPR1[6]	RXPR1[5]	RXPR1[4]	RXPR1[3]	RXPR1[2]	RXPR1[1]	RXPR1[0]
	RXPR0_0	RXPR0[15]	RXPR0[14]	RXPR0[13]	RXPR0[12]	RXPR0[11]	RXPR0[10]	RXPR0[9]	RXPR0[8]
		RXPR0[7]	RXPR0[6]	RXPR0[5]	RXPR0[4]	RXPR0[3]	RXPR0[2]	RXPR0[1]	RXPR0[0]
	RFPR1_0	RFPR1[15]	RFPR1[14]	RFPR1[13]	RFPR1[12]	RFPR1[11]	RFPR1[10]	RFPR1[9]	RFPR1[8]
		RFPR1[7]	RFPR1[6]	RFPR1[5]	RFPR1[4]	RFPR1[3]	RFPR1[2]	RFPR1[1]	RFPR1[0]
	RFPR0_0	RFPR0[15]	RFPR0[14]	RFPR0[13]	RFPR0[12]	RFPR0[11]	RFPR0[10]	RFPR0[9]	RFPR0[8]
		RFPR0[7]	RFPR0[6]	RFPR0[5]	RFPR0[4]	RFPR0[3]	RFPR0[2]	RFPR0[1]	RFPR0[0]
	MBIMR1_0	MBIMR1[15]	MBIMR1[14]	MBIMR1[13]	MBIMR1[12]	MBIMR1[11]	MBIMR1[10]	MBIMR1[9]	MBIMR1[8]
		MBIMR1[7]	MBIMR1[6]	MBIMR1[5]	MBIMR1[4]	MBIMR1[3]	MBIMR1[2]	MBIMR1[1]	MBIMR1[0]
	MBIMR0_0	MBIMR0[15]	MBIMR0[14]	MBIMR0[13]	MBIMR0[12]	MBIMR0[11]	MBIMR0[10]	MBIMR0[9]	MBIMR0[8]
		MBIMR0[7]	MBIMR0[6]	MBIMR0[5]	MBIMR0[4]	MBIMR0[3]	MBIMR0[2]	MBIMR0[1]	MBIMR0[0]
	UMSR1_0	UMSR1[15]	UMSR1[14]	UMSR1[13]	UMSR1[12]	UMSR1[11]	UMSR1[10]	UMSR1[9]	UMSR1[8]
		UMSR1[7]	UMSR1[6]	UMSR1[5]	UMSR1[4]	UMSR1[3]	UMSR1[2]	UMSR1[1]	UMSR1[0]
	UMSR0_0	UMSR0[15]	UMSR0[14]	UMSR0[13]	UMSR0[12]	UMSR0[11]	UMSR0[10]	UMSR0[9]	UMSR0[8]
		UMSR0[7]	UMSR0[6]	UMSR0[5]	UMSR0[4]	UMSR0[3]	UMSR0[2]	UMSR0[1]	UMSR0[0]
	TTCR0_0	TCR15	TCR14	TCR13	TCR12	TCR11	TCR10	—	—
		—	TCR6	TPSC5	TPSC4	TPSC3	TPSC2	TPSC1	TPSC0
	CMAX_TEW_0	—	—	—	—	—	CMAX[2]	CMAX[1]	CMAX[0]
		—	—	—	TEW[4]	TEW[3]	TEW[2]	TEW[1]	TEW[0]

模块名称	寄存器简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
RCAN-TL1	RFTROFF_0	RTROFF[7]	RTROFF[6]	RTROFF[5]	RTROFF[4]	RTROFF[3]	RTROFF[2]	RTROFF[1]	RTROFF[0]
		—	—	—	—	—	—	—	—
	TSR_0	—	—	—	—	—	—	—	—
		—	—	—	TSR4	TSR3	TSR2	TSR1	TSR0
	CCR_0	—	—	—	—	—	—	—	—
		—	—	CCR[5]	CCR[4]	CCR[3]	CCR[2]	CCR[1]	CCR[0]
	TCNTR_0	TCNTR[15]	TCNTR[14]	TCNTR[13]	TCNTR[12]	TCNTR[11]	TCNTR[10]	TCNTR[9]	TCNTR[8]
		TCNTR[7]	TCNTR[6]	TCNTR[5]	TCNTR[4]	TCNTR[3]	TCNTR[2]	TCNTR[1]	TCNTR[0]
	CYCTR_0	CYCTR[15]	CYCTR[14]	CYCTR[13]	CYCTR[12]	CYCTR[11]	CYCTR[10]	CYCTR[9]	CYCTR[8]
		CYCTR[7]	CYCTR[6]	CYCTR[5]	CYCTR[4]	CYCTR[3]	CYCTR[2]	CYCTR[1]	CYCTR[0]
	RFMK_0	RFMK[15]	RFMK[14]	RFMK[13]	RFMK[12]	RFMK[11]	RFMK[10]	RFMK[9]	RFMK[8]
		RFMK[7]	RFMK[6]	RFMK[5]	RFMK[4]	RFMK[3]	RFMK[2]	RFMK[1]	RFMK[0]
	TCMR0_0	TCMR0[15]	TCMR0[14]	TCMR0[13]	TCMR0[12]	TCMR0[11]	TCMR0[10]	TCMR0[9]	TCMR0[8]
		TCMR0[7]	TCMR0[6]	TCMR0[5]	TCMR0[4]	TCMR0[3]	TCMR0[2]	TCMR0[1]	TCMR0[0]
	TCMR1_0	TCMR1[15]	TCMR1[14]	TCMR1[13]	TCMR1[12]	TCMR1[11]	TCMR1[10]	TCMR1[9]	TCMR1[8]
		TCMR1[7]	TCMR1[6]	TCMR1[5]	TCMR1[4]	TCMR1[3]	TCMR1[2]	TCMR1[1]	TCMR1[0]
	TCMR2_0	TCMR2[15]	TCMR2[14]	TCMR2[13]	TCMR2[12]	TCMR2[11]	TCMR2[10]	TCMR2[9]	TCMR2[8]
		TCMR2[7]	TCMR2[6]	TCMR2[5]	TCMR2[4]	TCMR2[3]	TCMR2[2]	TCMR2[1]	TCMR2[0]
	TTTSEL_0	—	TTTSEL[14]	TTTSEL[13]	TTTSEL[12]	TTTSEL[11]	TTTSEL[10]	TTTSEL[9]	TTTSEL[8]
		—	—	—	—	—	—	—	—
	MBn_CONT ROLO_H_0 (n=0 ~ 31)*8	—	STDID[10]	STDID[9]	STDID[8]	STDID[7]	STDID[6]	STDID[5]	STDID[4]
		STDID[3]	STDID[2]	STDID[1]	STDID[0]	RTR	IDE	EXTID[17]	EXTID[16]
	MBn_CONT ROLO_H_0 (n=0 ~ 31)*9	IDE	RTR	—	STDID[10]	STDID[9]	STDID[8]	STDID[7]	STDID[6]
		STDID[5]	STDID[4]	STDID[3]	STDID[2]	STDID[1]	STDID[0]	EXTID[17]	EXTID[16]
	MBn_CONT ROLO_L_0 (n=0 ~ 31)	EXTID[15]	EXTID[14]	EXTID[13]	EXTID[12]	EXTID[11]	EXTID[10]	EXTID[9]	EXTID[8]
		EXTID[7]	EXTID[6]	EXTID[5]	EXTID[4]	EXTID[3]	EXTID[2]	EXTID[1]	EXTID[0]
	MBn_LAFM0 _0 (n=0 ~ 31)*8	—	STDID_LAFM[10]	STDID_LAFM[9]	STDID_LAFM[8]	STDID_LAFM[7]	STDID_LAFM[6]	STDID_LAFM[5]	STDID_LAFM[4]
		STDID_LAFM[3]	STDID_LAFM[2]	STDID_LAFM[1]	STDID_LAFM[0]	—	IDE	EXTID_LAFM[17]	EXTID_LAFM[16]
	MBn_LAFM0 _0 (n= ~ 031)*9	IDE	—	—	STDID_LAFM[10]	STDID_LAFM[9]	STDID_LAFM[8]	STDID_LAFM[7]	STDID_LAFM[6]
		STDID_LAFM[5]	STDID_LAFM[4]	STDID_LAFM[3]	STDID_LAFM[2]	STDID_LAFM[1]	STDID_LAFM[0]	EXTID_LAFM[17]	EXTID_LAFM[16]
	MBn_LAFM1 _0 (n=0 ~ 31)	EXTID_LAFM[15]	EXTID_LAFM[14]	EXTID_LAFM[13]	EXTID_LAFM[12]	EXTID_LAFM[11]	EXTID_LAFM[10]	EXTID_LAFM[9]	EXTID_LAFM[8]
		EXTID_LAFM[7]	EXTID_LAFM[6]	EXTID_LAFM[5]	EXTID_LAFM[4]	EXTID_LAFM[3]	EXTID_LAFM[2]	EXTID_LAFM[1]	EXTID_LAFM[0]

模块名称	寄存器简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
RCAN-TL1	MBn_DATA_01_0 (n=0 ~ 31)	MSG_DATA0	MSG_DATA0	MSG_DATA0	MSG_DATA0	MSG_DATA0	MSG_DATA0	MSG_DATA0	MSG_DATA0
		MSG_DATA1	MSG_DATA1	MSG_DATA1	MSG_DATA1	MSG_DATA1	MSG_DATA1	MSG_DATA1	MSG_DATA1
	MBn_DATA_23_0 (n=0 ~ 31)	MSG_DATA2	MSG_DATA2	MSG_DATA2	MSG_DATA2	MSG_DATA2	MSG_DATA2	MSG_DATA2	MSG_DATA2
		MSG_DATA3	MSG_DATA3	MSG_DATA3	MSG_DATA3	MSG_DATA3	MSG_DATA3	MSG_DATA3	MSG_DATA3
	MBn_DATA_45_0 (n=0 ~ 31)	MSG_DATA4	MSG_DATA4	MSG_DATA4	MSG_DATA4	MSG_DATA4	MSG_DATA4	MSG_DATA4	MSG_DATA4
		MSG_DATA5	MSG_DATA5	MSG_DATA5	MSG_DATA5	MSG_DATA5	MSG_DATA5	MSG_DATA5	MSG_DATA5
	MBn_DATA_67_0 (n=0 ~ 31)	MSG_DATA6	MSG_DATA6	MSG_DATA6	MSG_DATA6	MSG_DATA6	MSG_DATA6	MSG_DATA6	MSG_DATA6
		MSG_DATA7	MSG_DATA7	MSG_DATA7	MSG_DATA7	MSG_DATA7	MSG_DATA7	MSG_DATA7	MSG_DATA7
	MBn_CONTROL1_0 (n=0)	—	—	NMC	—	—	MBC[2]	MBC[1]	MBC[0]
		—	—	—	—	DLD[3]	DLD[2]	DLD[1]	DLD[0]
	MBn_CONTROL1_0 (n=1 ~ 31)	—	—	NMC	ATX	DART	MBC[2]	MBC[1]	MBC[0]
		—	—	—	—	DLD[3]	DLD[2]	DLD[1]	DLD[0]
	MBn_TIMESAMP_0(n=0 ~ 15、30、31)	TS15	TS14	TS13	TS12	TS11	TS10	TS9	TS8
		TS7	TS6	TS5	TS4	TS3	TS2	TS1	TS0
	MBn_TTT_0 (n=24 ~ 30)	TTT15	TTT14	TTT13	TTT12	TTT11	TTT10	TTT9	TTT8
		TTT7	TTT6	TTT5	TTT4	TTT3	TTT2	TTT1	TTT0
	MBn_TTCONTROL_0 (n=24 ~ 29)	TTW[1]	TTW[0]	OFFSET[5]	OFFSET[4]	OFFSET[3]	OFFSET[2]	OFFSET[1]	OFFSET[0]
		—	—	—	—	—	REP_FACTOR[2]	REP_FACTOR[1]	REP_FACTOR[0]
	MCR_1	MCR15	MCR14	—	—	—	TST2	TST1	TST0
		MCR7	MCR6	MCR5	—	—	MCR2	MCR1	MCR0
	GSR_1	—	—	—	—	—	—	—	—
		—	—	GSR5	GSR4	GSR3	GSR2	GSR1	GSR0
	BCR1_1	TSEG13	TSEG12	TSEG11	TSEG10	—	TSEG22	TSEG21	TSEG20
		—	—	SJW1	SJW0	—	—	—	BSP
	BCR0_1	—	—	—	—	—	—	—	—
		BRP7	BRP6	BRP5	BRP4	BRP3	BRP2	BRP1	BRP0
	IRR_1	IRR15	IRR14	IRR13	IRR12	IRR11	IRR10	IRR9	IRR8
		IRR7	IRR6	IRR5	IRR4	IRR3	IRR2	IRR1	IRR0
	IMR_1	IMR15	IMR14	IMR13	IMR12	IMR11	IMR10	IMR9	IMR8
		IMR7	IMR6	IMR5	IMR4	IMR3	IMR2	IMR1	IMR0

模块名称	寄存器简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
RCAN- TL1	TEC_REC_1	TEC[7]	TEC[6]	TEC[5]	TEC[4]	TEC[3]	TEC[2]	TEC[1]	TEC[0]
		REC[7]	REC[6]	REC[5]	REC[4]	REC[3]	REC[2]	REC[1]	REC[0]
	TXPR1_1	TXPR1[15]	TXPR1[14]	TXPR1[13]	TXPR1[12]	TXPR1[11]	TXPR1[10]	TXPR1[9]	TXPR1[8]
		TXPR1[7]	TXPR1[6]	TXPR1[5]	TXPR1[4]	TXPR1[3]	TXPR1[2]	TXPR1[1]	TXPR1[0]
	TXPR0_1	TXPR0[15]	TXPR0[14]	TXPR0[13]	TXPR0[12]	TXPR0[11]	TXPR0[10]	TXPR0[9]	TXPR0[8]
		TXPR0[7]	TXPR0[6]	TXPR0[5]	TXPR0[4]	TXPR0[3]	TXPR0[2]	TXPR0[1]	—
	TXCR1_1	TXCR1[15]	TXCR1[14]	TXCR1[13]	TXCR1[12]	TXCR1[11]	TXCR1[10]	TXCR1[9]	TXCR1[8]
		TXCR1[7]	TXCR1[6]	TXCR1[5]	TXCR1[4]	TXCR1[3]	TXCR1[2]	TXCR1[1]	TXCR1[0]
	TXCR0_1	TXCR0[15]	TXCR0[14]	TXCR0[13]	TXCR0[12]	TXCR0[11]	TXCR0[10]	TXCR0[9]	TXCR0[8]
		TXCR0[7]	TXCR0[6]	TXCR0[5]	TXCR0[4]	TXCR0[3]	TXCR0[2]	TXCR0[1]	—
	TXACK1_1	TXACK1[15]	TXACK1[14]	TXACK1[13]	TXACK1[12]	TXACK1[11]	TXACK1[10]	TXACK1[9]	TXACK1[8]
		TXACK1[7]	TXACK1[6]	TXACK1[5]	TXACK1[4]	TXACK1[3]	TXACK1[2]	TXACK1[1]	TXACK1[0]
	TXACK0_1	TXACK0[15]	TXACK0[14]	TXACK0[13]	TXACK0[12]	TXACK0[11]	TXACK0[10]	TXACK0[9]	TXACK0[8]
		TXACK0[7]	TXACK0[6]	TXACK0[5]	TXACK0[4]	TXACK0[3]	TXACK0[2]	TXACK0[1]	—
	ABACK1_1	ABACK1[15]	ABACK1[14]	ABACK1[13]	ABACK1[12]	ABACK1[11]	ABACK1[10]	ABACK1[9]	ABACK1[8]
		ABACK1[7]	ABACK1[6]	ABACK1[5]	ABACK1[4]	ABACK1[3]	ABACK1[2]	ABACK1[1]	ABACK1[0]
	ABACK0_1	ABACK0[15]	ABACK0[14]	ABACK0[13]	ABACK0[12]	ABACK0[11]	ABACK0[10]	ABACK0[9]	ABACK0[8]
		ABACK0[7]	ABACK0[6]	ABACK0[5]	ABACK0[4]	ABACK0[3]	ABACK0[2]	ABACK0[1]	—
	RXPR1_1	RXPR1[15]	RXPR1[14]	RXPR1[13]	RXPR1[12]	RXPR1[11]	RXPR1[10]	RXPR1[9]	RXPR1[8]
		RXPR1[7]	RXPR1[6]	RXPR1[5]	RXPR1[4]	RXPR1[3]	RXPR1[2]	RXPR1[1]	RXPR1[0]
	RXPR0_1	RXPR0[15]	RXPR0[14]	RXPR0[13]	RXPR0[12]	RXPR0[11]	RXPR0[10]	RXPR0[9]	RXPR0[8]
		RXPR0[7]	RXPR0[6]	RXPR0[5]	RXPR0[4]	RXPR0[3]	RXPR0[2]	RXPR0[1]	RXPR0[0]
	RFPR1_1	RFPR1[15]	RFPR1[14]	RFPR1[13]	RFPR1[12]	RFPR1[11]	RFPR1[10]	RFPR1[9]	RFPR1[8]
		RFPR1[7]	RFPR1[6]	RFPR1[5]	RFPR1[4]	RFPR1[3]	RFPR1[2]	RFPR1[1]	RFPR1[0]
	RFPR0_1	RFPR0[15]	RFPR0[14]	RFPR0[13]	RFPR0[12]	RFPR0[11]	RFPR0[10]	RFPR0[9]	RFPR0[8]
		RFPR0[7]	RFPR0[6]	RFPR0[5]	RFPR0[4]	RFPR0[3]	RFPR0[2]	RFPR0[1]	RFPR0[0]
	MBIMR1_1	MBIMR1[15]	MBIMR1[14]	MBIMR1[13]	MBIMR1[12]	MBIMR1[11]	MBIMR1[10]	MBIMR1[9]	MBIMR1[8]
		MBIMR1[7]	MBIMR1[6]	MBIMR1[5]	MBIMR1[4]	MBIMR1[3]	MBIMR1[2]	MBIMR1[1]	MBIMR1[0]
	MBIMR0_1	MBIMR0[15]	MBIMR0[14]	MBIMR0[13]	MBIMR0[12]	MBIMR0[11]	MBIMR0[10]	MBIMR0[9]	MBIMR0[8]
		MBIMR0[7]	MBIMR0[6]	MBIMR0[5]	MBIMR0[4]	MBIMR0[3]	MBIMR0[2]	MBIMR0[1]	MBIMR0[0]
	UMSR1_1	UMSR1[15]	UMSR1[14]	UMSR1[13]	UMSR1[12]	UMSR1[11]	UMSR1[10]	UMSR1[9]	UMSR1[8]
		UMSR1[7]	UMSR1[6]	UMSR1[5]	UMSR1[4]	UMSR1[3]	UMSR1[2]	UMSR1[1]	UMSR1[0]
	UMSR0_1	UMSR0[15]	UMSR0[14]	UMSR0[13]	UMSR0[12]	UMSR0[11]	UMSR0[10]	UMSR0[9]	UMSR0[8]
		UMSR0[7]	UMSR0[6]	UMSR0[5]	UMSR0[4]	UMSR0[3]	UMSR0[2]	UMSR0[1]	UMSR0[0]

模块名称	寄存器简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
RCAN-TL1	TTCR0_1	TCR[15]	TCR[14]	TCR[13]	TCR[12]	TCR[11]	TCR[10]	—	—
		—	TCR[6]	TPSC[5]	TPSC[4]	TPSC[3]	TPSC[2]	TPSC[1]	TPSC[0]
	CMAX_TEW_1	—	—	—	—	—	CMAX[2]	CMAX[1]	CMAX[0]
		—	—	—	TEW[4]	TEW[3]	TEW[2]	TEW[1]	TEW[0]
	RFTROFF_1	RTROFF[7]	RTROFF[6]	RTROFF[5]	RTROFF[4]	RTROFF[3]	RTROFF[2]	RTROFF[1]	RTROFF[0]
		—	—	—	—	—	—	—	—
	TSR_1	—	—	—	—	—	—	—	—
		—	—	—	TSR[4]	TSR[3]	TSR[2]	TSR[1]	TSR[0]
	CCR_1	—	—	—	—	—	—	—	—
		—	—	CCR[5]	CCR[4]	CCR[3]	CCR[2]	CCR[1]	CCR[0]
	TCNTR_1	TCNTR[15]	TCNTR[14]	TCNTR[13]	TCNTR[12]	TCNTR[11]	TCNTR[10]	TCNTR[9]	TCNTR[8]
		TCNTR[7]	TCNTR[6]	TCNTR[5]	TCNTR[4]	TCNTR[3]	TCNTR[2]	TCNTR[1]	TCNTR[0]
	CYCTR_1	CYCTR[15]	CYCTR[14]	CYCTR[13]	CYCTR[12]	CYCTR[11]	CYCTR[10]	CYCTR[9]	CYCTR[8]
		CYCTR[7]	CYCTR[6]	CYCTR[5]	CYCTR[4]	CYCTR[3]	CYCTR[2]	CYCTR[1]	CYCTR[0]
	RFMK_1	RFMK[15]	RFMK[14]	RFMK[13]	RFMK[12]	RFMK[11]	RFMK[10]	RFMK[9]	RFMK[8]
		RFMK[7]	RFMK[6]	RFMK[5]	RFMK[4]	RFMK[3]	RFMK[2]	RFMK[1]	RFMK[0]
	TCMR0_1	TCMR0[15]	TCMR0[14]	TCMR0[13]	TCMR0[12]	TCMR0[11]	TCMR0[10]	TCMR0[9]	TCMR0[8]
		TCMR0[7]	TCMR0[6]	TCMR0[5]	TCMR0[4]	TCMR0[3]	TCMR0[2]	TCMR0[1]	TCMR0[0]
	TCMR1_1	TCMR1[15]	TCMR1[14]	TCMR1[13]	TCMR1[12]	TCMR1[11]	TCMR1[10]	TCMR1[9]	TCMR1[8]
		TCMR1[7]	TCMR1[6]	TCMR1[5]	TCMR1[4]	TCMR1[3]	TCMR1[2]	TCMR1[1]	TCMR1[0]
	TCMR2_1	TCMR2[15]	TCMR2[14]	TCMR2[13]	TCMR2[12]	TCMR2[11]	TCMR2[10]	TCMR2[9]	TCMR2[8]
		TCMR2[7]	TCMR2[6]	TCMR2[5]	TCMR2[4]	TCMR2[3]	TCMR2[2]	TCMR2[1]	TCMR2[0]
	TTTSEL_1	—	TTTSEL[14]	TTTSEL[13]	TTTSEL[12]	TTTSEL[11]	TTTSEL[10]	TTTSEL[9]	TTTSEL[8]
		—	—	—	—	—	—	—	—
	MBn_CONT ROL0_H_1 (n=0 ~ 31)*8	—	STDID[10]	STDID[9]	STDID[8]	STDID[7]	STDID[6]	STDID[5]	STDID[4]
		STDID[3]	STDID[2]	STDID[1]	STDID[0]	RTR	IDE	EXTID[17]	EXTID[16]
	MBn_CONT ROL0_H_1 (n=0 ~ 31)*9	IDE	RTR	—	STDID[10]	STDID[9]	STDID[8]	STDID[7]	STDID[6]
		STDID[5]	STDID[4]	STDID[3]	STDID[2]	STDID[1]	STDID[0]	EXTID[17]	EXTID[16]
	MBn_CONT ROL0_L_1 (n=0 ~ 31)	EXTID[15]	EXTID[14]	EXTID[13]	EXTID[12]	EXTID[11]	EXTID[10]	EXTID[9]	EXTID[8]
		EXTID[7]	EXTID[6]	EXTID[5]	EXTID[4]	EXTID[3]	EXTID[2]	EXTID[1]	EXTID[0]
	MBn_LAFM0 _1 (n=0 ~ 31)*8	—	STDID_ LAFM[10]	STDID_ LAFM[9]	STDID_ LAFM[8]	STDID_ LAFM[7]	STDID_ LAFM[6]	STDID_ LAFM[5]	STDID_ LAFM[4]
		STDID_ LAFM[3]	STDID_ LAFM[2]	STDID_ LAFM[1]	STDID_ LAFM[0]	—	IDE	EXTID_ LAFM[17]	EXTID_ LAFM[16]
	MBn_LAFM0 _1 (n=0 ~ 31)*9	IDE	—	—	STDID_ LAFM[10]	STDID_ LAFM[9]	STDID_ LAFM[8]	STDID_ LAFM[7]	STDID_ LAFM[6]
		STDID_ LAFM[5]	STDID_ LAFM[4]	STDID_ LAFM[3]	STDID_ LAFM[2]	STDID_ LAFM[1]	STDID_ LAFM[0]	EXTID_ LAFM[17]	EXTID_ LAFM[16]

模块名称	寄存器简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
RCAN-TL1	MBn_LAFM1_1 (n=0 ~ 31)	EXTID_LAFM[15]	EXTID_LAFM[14]	EXTID_LAFM[13]	EXTID_LAFM[12]	EXTID_LAFM[11]	EXTID_LAFM[10]	EXTID_LAFM[9]	EXTID_LAFM[8]
		EXTID_LAFM[7]	EXTID_LAFM[6]	EXTID_LAFM[5]	EXTID_LAFM[4]	EXTID_LAFM[3]	EXTID_LAFM[2]	EXTID_LAFM[1]	EXTID_LAFM[0]
	MBn_DATA_01_1 (n=0 ~ 31)	MSG_DATA0	MSG_DATA0	MSG_DATA0	MSG_DATA0	MSG_DATA0	MSG_DATA0	MSG_DATA0	MSG_DATA0
		MSG_DATA1	MSG_DATA1	MSG_DATA1	MSG_DATA1	MSG_DATA1	MSG_DATA1	MSG_DATA1	MSG_DATA1
	MBn_DATA_23_1 (n=0 ~ 31)	MSG_DATA2	MSG_DATA2	MSG_DATA2	MSG_DATA2	MSG_DATA2	MSG_DATA2	MSG_DATA2	MSG_DATA2
		MSG_DATA3	MSG_DATA3	MSG_DATA3	MSG_DATA3	MSG_DATA3	MSG_DATA3	MSG_DATA3	MSG_DATA3
	MBn_DATA_45_1 (n=0 ~ 31)	MSG_DATA4	MSG_DATA4	MSG_DATA4	MSG_DATA4	MSG_DATA4	MSG_DATA4	MSG_DATA4	MSG_DATA4
		MSG_DATA5	MSG_DATA5	MSG_DATA5	MSG_DATA5	MSG_DATA5	MSG_DATA5	MSG_DATA5	MSG_DATA5
	MBn_DATA_67_1 (n=0 ~ 31)	MSG_DATA6	MSG_DATA6	MSG_DATA6	MSG_DATA6	MSG_DATA6	MSG_DATA6	MSG_DATA6	MSG_DATA6
		MSG_DATA7	MSG_DATA7	MSG_DATA7	MSG_DATA7	MSG_DATA7	MSG_DATA7	MSG_DATA7	MSG_DATA7
	MBn_CONTROL1_1 (n=0)	—	—	NMC	—	—	MBC[2]	MBC[1]	MBC[0]
		—	—	—	—	DLD[3]	DLD[2]	DLD[1]	DLD[0]
	MBn_CONTROL1_1 (n=1 ~ 31)	—	—	NMC	ATX	DART	MBC[2]	MBC[1]	MBC[0]
		—	—	—	—	DLD[3]	DLD[2]	DLD[1]	DLD[0]
	MBn_TIMES_TAMP_1 (n=0 ~ 15、30、31)	TS15	TS14	TS13	TS12	TS11	TS10	TS9	TS8
		TS7	TS6	TS5	TS4	TS3	TS2	TS1	TS0
	MBn_TTT_1 (n=24 ~ 30)	TTT15	TTT14	TTT13	TTT12	TTT11	TTT10	TTT9	TTT8
		TTT7	TTT6	TTT5	TTT4	TTT3	TTT2	TTT1	TTT0
	MBn_TTCONTROL_1 (n=24 ~ 29)	TTW[1]	TTW[0]	OFFSET[5]	OFFSET[4]	OFFSET[3]	OFFSET[2]	OFFSET[1]	OFFSET[0]
		—	—	—	—	—	REP_FACTOR[2]	REP_FACTOR[1]	REP_FACTOR[0]
ADC	ADDRA								
				—	—	—	—	—	—
	ADDRB								
				—	—	—	—	—	—
	ADDRC								
				—	—	—	—	—	—
	ADDRD								
				—	—	—	—	—	—

模块名称	寄存器简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
ADC	ADDRE								
				—	—	—	—	—	—
	ADDRF								
				—	—	—	—	—	—
	ADDRG								
				—	—	—	—	—	—
	ADDRH								
				—	—	—	—	—	—
DAC	ADCSR	ADF	ADIE	ADST	—	TRGS[3]	TRGS[2]	TRGS[1]	TRGS[0]
		CKS[1]	CKS[0]	MDS[2]	MDS[1]	MDS[0]	CH[2]	CH[1]	CH[0]
	DADR0								
	DADR1								
FLCTL	DACR	DAOE1	DAOE0	DAE	—	—	—	—	—
	FLCMNCR	—	—	—	—	—	—	—	—
		—	—	—	—	—	SNAND	QTSEL	—
		FCKSEL	—	ECCPOS[1]	ECCPOS[0]	ACM[1]	ACM[0]	NANDWF	—
		—	—	—	—	CE	—	—	TYPESEL
	FLCMDRCR	ADRCNT2	SCTCNT[19]	SCTCNT[18]	SCTCNT[17]	SCTCNT[16]	ADRMD	CDSRC	DOSR
		—	—	SELRW	DOADR	ADRCNT[1]	ADRCNT[0]	DOCMD2	DOCMD1
		SCTCNT[15]	SCTCNT[14]	SCTCNT[13]	SCTCNT[12]	SCTCNT[11]	SCTCNT[10]	SCTCNT[9]	SCTCNT[8]
		SCTCNT[7]	SCTCNT[6]	SCTCNT[5]	SCTCNT[4]	SCTCNT[3]	SCTCNT[2]	SCTCNT[1]	SCTCNT[0]
		—	—	—	—	—	—	—	—
	FLCMCDR	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		CMD2[7]	CMD2[6]	CMD2[5]	CMD2[4]	CMD2[3]	CMD2[2]	CMD2[1]	CMD2[0]
		CMD1[7]	CMD1[6]	CMD1[5]	CMD1[4]	CMD1[3]	CMD1[2]	CMD1[1]	CMD1[0]
	FLADR*10	ADR4[7]	ADR4[6]	ADR4[5]	ADR4[4]	ADR4[3]	ADR4[2]	ADR4[1]	ADR4[0]
		ADR3[7]	ADR3[6]	ADR3[5]	ADR3[4]	ADR3[3]	ADR3[2]	ADR3[1]	ADR3[0]
		ADR2[7]	ADR2[6]	ADR2[5]	ADR2[4]	ADR2[3]	ADR2[2]	ADR2[1]	ADR2[0]
		ADR1[7]	ADR1[6]	ADR1[5]	ADR1[4]	ADR1[3]	ADR1[2]	ADR1[1]	ADR1[0]
	FLADR*11	—	—	—	—	—	—	ADR[25]	ADR[24]
		ADR[23]	ADR[22]	ADR[21]	ADR[20]	ADR[19]	ADR[18]	ADR[17]	ADR[16]
		ADR[15]	ADR[14]	ADR[13]	ADR[12]	ADR[11]	ADR[10]	ADR[9]	ADR[8]
		ADR[7]	ADR[6]	ADR[5]	ADR[4]	ADR[3]	ADR[2]	ADR[1]	ADR[0]
	FLADR2	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		ADR5[7]	ADR5[6]	ADR5[5]	ADR5[4]	ADR5[3]	ADR5[2]	ADR5[1]	ADR5[0]

模块名称	寄存器简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
FLCTL	FLDTCNTR	ECFLW[7]	ECFLW[6]	ECFLW[5]	ECFLW[4]	ECFLW[3]	ECFLW[2]	ECFLW[1]	ECFLW[0]
		DTFLW[7]	DTFLW[6]	DTFLW[5]	DTFLW[4]	DTFLW[3]	DTFLW[2]	DTFLW[1]	DTFLW[0]
		—	—	—	—	DTCNT[11]	DTCNT[10]	DTCNT[9]	DTCNT[8]
		DTCNT[7]	DTCNT[6]	DTCNT[5]	DTCNT[4]	DTCNT[3]	DTCNT[2]	DTCNT[1]	DTCNT[0]
	FLDATAR	DT4[7]	DT4[6]	DT4[5]	DT4[4]	DT4[3]	DT4[2]	DT4[1]	DT4[0]
		DT3[7]	DT3[6]	DT3[5]	DT3[4]	DT3[3]	DT3[2]	DT3[1]	DT3[0]
		DT2[7]	DT2[6]	DT2[5]	DT2[4]	DT2[3]	DT2[2]	DT2[1]	DT2[0]
		DT1[7]	DT1[6]	DT1[5]	DT1[4]	DT1[3]	DT1[2]	DT1[1]	DT1[0]
	FLINTDMACR	—	—	—	—	—	—	—	ECERINTE
		—	—	FIFOTRG [1]	FIFOTRG [0]	AC1CLR	AC0CLR	DREQ1EN	DREQ0EN
		—	—	—	—	—	—	ECERB	STERB
		BTOERB	TRREQF1	TRREQF0	STERINTE	RBERINTE	TEINTE	TRINTE1	TRINTE0
	FLBSYTMR	—	—	—	—	—	—	—	—
		—	—	—	—	RBTMOUT [19]	RBTMOUT [18]	RBTMOUT [17]	RBTMOUT [16]
		RBTMOUT [15]	RBTMOUT [14]	RBTMOUT [13]	RBTMOUT [12]	RBTMOUT [11]	RBTMOUT [10]	RBTMOUT [9]	RBTMOUT [8]
		RBTMOUT [7]	RBTMOUT [6]	RBTMOUT [5]	RBTMOUT [4]	RBTMOUT [3]	RBTMOUT [2]	RBTMOUT [1]	RBTMOUT [0]
	FLBSYCNT	STAT[7]	STAT[6]	STAT[5]	STAT[4]	STAT[3]	STAT[2]	STAT[1]	STAT[0]
		—	—	—	—	RBTIMCNT [19]	RBTIMCNT [18]	RBTIMCNT [17]	RBTIMCNT [16]
		RBTIMCNT [15]	RBTIMCNT [14]	RBTIMCNT [13]	RBTIMCNT [12]	RBTIMCNT [11]	RBTIMCNT [10]	RBTIMCNT [9]	RBTIMCNT [8]
		RBTIMCNT [7]	RBTIMCNT [6]	RBTIMCNT [5]	RBTIMCNT [4]	RBTIMCNT [3]	RBTIMCNT [2]	RBTIMCNT [1]	RBTIMCNT [0]
	FLDTFIFO	DTFO[31]	DTFO[30]	DTFO[29]	DTFO[28]	DTFO[27]	DTFO[26]	DTFO[25]	DTFO[24]
		DTFO[23]	DTFO[22]	DTFO[21]	DTFO[20]	DTFO[19]	DTFO[18]	DTFO[17]	DTFO[16]
		DTFO[15]	DTFO[14]	DTFO[13]	DTFO[12]	DTFO[11]	DTFO[10]	DTFO[9]	DTFO[8]
		DTFO[7]	DTFO[6]	DTFO[5]	DTFO[4]	DTFO[3]	DTFO[2]	DTFO[1]	DTFO[0]
	FLECFIFO	ECFO[31]	ECFO[30]	ECFO[29]	ECFO[28]	ECFO[27]	ECFO[26]	ECFO[25]	ECFO[24]
		ECFO[23]	ECFO[22]	ECFO[21]	ECFO[20]	ECFO[19]	ECFO[18]	ECFO[17]	ECFO[16]
		ECFO[15]	ECFO[14]	ECFO[13]	ECFO[12]	ECFO[11]	ECFO[10]	ECFO[9]	ECFO[8]
		ECFO[7]	ECFO[6]	ECFO[5]	ECFO[4]	ECFO[3]	ECFO[2]	ECFO[1]	ECFO[0]
	FLTRCR	—	—	—	—	—	—	TREND	TRSTRT

模块名称	寄存器简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
USB	SYSCFG	—	—	—	—	—	—	—	—
		HSE	DCFM	DMRPD	DPRPU	—	FSRPC	—	USBE
	SYSSTS	—	—	—	—	—	—	—	—
		—	—	SOFEN	—	—	—	LNST[1]	LNST[0]
	DVSTCTR	UACKEY0	—	—	UACKEY1	—	—	—	WKUP
		RWUPE	USBRST	RESUME	UACT	—	—	RHST[1]	RHST[0]
	TESTMODE	HOSTPCC	—	—	—	—	—	—	—
		—	—	—	—	UTST[3]	UTST[2]	UTST[1]	UTST[0]
	CFBCFG	—	—	—	—	—	—	—	—
		—	—	—	—	FWAIT[3]	FWAIT[2]	FWAIT[1]	FWAIT[0]
	D0FBCFG	—	—	—	—	—	—	TENDE	FEND
		—	—	—	—	FWAIT[3]	FWAIT[2]	FWAIT[1]	FWAIT[0]
	D1FBCFG	—	—	—	—	—	—	TENDE	FEND
		—	—	—	—	FWAIT[3]	FWAIT[2]	FWAIT[1]	FWAIT[0]
	CFIFO	FIFOPORT [31]	FIFOPORT [30]	FIFOPORT [29]	FIFOPORT [28]	FIFOPORT [27]	FIFOPORT [26]	FIFOPORT [25]	FIFOPORT [24]
		FIFOPORT [23]	FIFOPORT [22]	FIFOPORT [21]	FIFOPORT [20]	FIFOPORT [19]	FIFOPORT [18]	FIFOPORT [17]	FIFOPORT [16]
		FIFOPORT [15]	FIFOPORT [14]	FIFOPORT [13]	FIFOPORT [12]	FIFOPORT [11]	FIFOPORT [10]	FIFOPORT [9]	FIFOPORT [8]
		FIFOPORT [7]	FIFOPORT [6]	FIFOPORT [5]	FIFOPORT [4]	FIFOPORT [3]	FIFOPORT [2]	FIFOPORT [1]	FIFOPORT [0]
	D0FIFO	FIFOPORT [31]	FIFOPORT [30]	FIFOPORT [29]	FIFOPORT [28]	FIFOPORT [27]	FIFOPORT [26]	FIFOPORT [25]	FIFOPORT [24]
		FIFOPORT [23]	FIFOPORT [22]	FIFOPORT [21]	FIFOPORT [20]	FIFOPORT [19]	FIFOPORT [18]	FIFOPORT [17]	FIFOPORT [16]
		FIFOPORT [15]	FIFOPORT [14]	FIFOPORT [13]	FIFOPORT [12]	FIFOPORT [11]	FIFOPORT [10]	FIFOPORT [9]	FIFOPORT [8]
		FIFOPORT [7]	FIFOPORT [6]	FIFOPORT [5]	FIFOPORT [4]	FIFOPORT [3]	FIFOPORT [2]	FIFOPORT [1]	FIFOPORT [0]
	D1FIFO	FIFOPORT [31]	FIFOPORT [30]	FIFOPORT [29]	FIFOPORT [28]	FIFOPORT [27]	FIFOPORT [26]	FIFOPORT [25]	FIFOPORT [24]
		FIFOPORT [23]	FIFOPORT [22]	FIFOPORT [21]	FIFOPORT [20]	FIFOPORT [19]	FIFOPORT [18]	FIFOPORT [17]	FIFOPORT [16]
		FIFOPORT [15]	FIFOPORT [14]	FIFOPORT [13]	FIFOPORT [12]	FIFOPORT [11]	FIFOPORT [10]	FIFOPORT [9]	FIFOPORT [8]
		FIFOPORT [7]	FIFOPORT [6]	FIFOPORT [5]	FIFOPORT [4]	FIFOPORT [3]	FIFOPORT [2]	FIFOPORT [1]	FIFOPORT [0]
	CFIFOSEL	RCNT	REW	—	—	MBW[1]	MBW[0]	—	—
		—	—	ISEL	—	—	CURPIPE [2]	CURPIPE [1]	CURPIPE [0]

模块名称	寄存器简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
USB	CFIFOCTR	BVAL	BCLR	FRDY	—	DTLN[11]	DTLN[10]	DTLN[9]	DTLN[8]
		DTLN[7]	DTLN[6]	DTLN[5]	DTLN[4]	DTLN[3]	DTLN[2]	DTLN[1]	DTLN[0]
	CFIFOSIE	TGL	SCLR	SBUSY	—	—	—	—	—
		—	—	—	—	—	—	—	—
	D0FIFOSEL	RCNT	REW	DCLRM	DREQE	MBW[1]	MBW[0]	TRENB	TRCLR
		DEZPM	—	—	—	—	CURPIPE [2]	CURPIPE [1]	CURPIPE [0]
	D0FIFOCTR	BVAL	BCLR	FRDY	—	DTLN[11]	DTLN[10]	DTLN[9]	DTLN[8]
		DTLN[7]	DTLN[6]	DTLN[5]	DTLN[4]	DTLN[3]	DTLN[2]	DTLN[1]	DTLN[0]
	D0FIFOTRN	TRNCNT [15]	TRNCNT [14]	TRNCNT [13]	TRNCNT [12]	TRNCNT [11]	TRNCNT [10]	TRNCNT [9]	TRNCNT [8]
		TRNCNT [7]	TRNCNT [6]	TRNCNT [5]	TRNCNT [4]	TRNCNT [3]	TRNCNT [2]	TRNCNT [1]	TRNCNT [0]
	D1FIFOSEL	RCNT	REW	DCLRM	DREQE	MBW[1]	MBW[0]	TRENB	TRCLR
		DEZPM	—	—	—	—	CURPIPE [2]	CURPIPE [1]	CURPIPE [0]
	D1FIFOCTR	BVAL	BCLR	FRDY	—	DTLN[11]	DTLN[10]	DTLN[9]	DTLN[8]
		DTLN[7]	DTLN[6]	DTLN[5]	DTLN[4]	DTLN[3]	DTLN[2]	DTLN[1]	DTLN[0]
	D1FIFOTRN	TRNCNT [15]	TRNCNT [14]	TRNCNT [13]	TRNCNT [12]	TRNCNT [11]	TRNCNT [10]	TRNCNT [9]	TRNCNT [8]
		TRNCNT [7]	TRNCNT [6]	TRNCNT [5]	TRNCNT [4]	TRNCNT [3]	TRNCNT [2]	TRNCNT [1]	TRNCNT [0]
	INTENB0	VBSE	RSME	SOFE	DVSE	CTRE	BEMPE	NRDYE	BRDYE
		URST	SADR	SCFG	SUSP	WDST	RDST	CMPL	SERR
	INTENB1	—	BCHGE	—	DTCHE	—	—	—	—
		—	—	SIGNE	SACKE	—	BRDYM	—	—
	BRDYENB	—	—	—	—	—	—	—	—
		PIPE7 BRDYE	PIPE6 BRDYE	PIPE5 BRDYE	PIPE4 BRDYE	PIPE3 BRDYE	PIPE2 BRDYE	PIPE1 BRDYE	PIPE0 BRDYE
	NRDYENB	—	—	—	—	—	—	—	—
		PIPE7 NRDYE	PIPE6 NRDYE	PIPE5 NRDYE	PIPE4 NRDYE	PIPE3 NRDYE	PIPE2 NRDYE	PIPE1 NRDYE	PIPE0 NRDYE
	BEMPENB	—	—	—	—	—	—	—	—
		PIPE7 BEMPE	PIPE6 BEMPE	PIPE5 BEMPE	PIPE4 BEMPE	PIPE3 BEMPE	PIPE2 BEMPE	PIPE1 BEMPE	PIPE0 BEMPE
	INTSTS0	VBINT	RESM	SOFR	DVST	CTRT	BEMP	NRDY	BRDY
		VBSTS	DVSQ[2]	DVSQ[1]	DVSQ[0]	VALID	CTSQ[2]	CTSQ[1]	CTSQ[0]
	INTSTS1	—	BCHG	SOFR	DTCH	—	BEMP	NRDY	BRDY
		—	—	SIGN	SACK	—	—	—	—
	BRDYSTS	—	—	—	—	—	—	—	—
		PIPE7 BRDY	PIPE6 BRDY	PIPE5 BRDY	PIPE4 BRDY	PIPE3 BRDY	PIPE2 BRDY	PIPE1 BRDY	PIPE0 BRDY

模块名称	寄存器简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
USB	NRDYSTS	—	—	—	—	—	—	—	—
		PIPE7NRDY	PIPE6NRDY	PIPE5NRDY	PIPE4NRDY	PIPE3NRDY	PIPE2NRDY	PIPE1NRDY	PIPE0NRDY
	BEMPSTS	—	—	—	—	—	—	—	—
		PIPE7BEMP	PIPE6BEMP	PIPE5BEMP	PIPE4BEMP	PIPE3BEMP	PIPE2BEMP	PIPE1BEMP	PIPE0BEMP
	FRMNUM	OV RN	CRCE	—	—	SOFRM	FRNM[10]	FRNM[9]	FRNM[8]
		FRNM[7]	FRNM[6]	FRNM[5]	FRNM[4]	FRNM[3]	FRNM[2]	FRNM[1]	FRNM[0]
	UFRMNUM	—	—	—	—	—	—	—	—
		—	—	—	—	—	UFRNM[2]	UFRNM[1]	UFRNM[0]
	USBADDR	—	—	—	—	—	—	—	—
		—	USBADDR[6]	USBADDR[5]	USBADDR[4]	USBADDR[3]	USBADDR[2]	USBADDR[1]	USBADDR[0]
	USBREQ	BREQUEST[7]	BREQUEST[6]	BREQUEST[5]	BREQUEST[4]	BREQUEST[3]	BREQUEST[2]	BREQUEST[1]	BREQUEST[0]
		BMREQUESTTYPE[7]	BMREQUESTTYPE[6]	BMREQUESTTYPE[5]	BMREQUESTTYPE[4]	BMREQUESTTYPE[3]	BMREQUESTTYPE[2]	BMREQUESTTYPE[1]	BMREQUESTTYPE[0]
	USBVAL	WVALUE[15]	WVALUE[14]	WVALUE[13]	WVALUE[12]	WVALUE[11]	WVALUE[10]	WVALUE[9]	WVALUE[8]
		WVALUE[7]	WVALUE[6]	WVALUE[5]	WVALUE[4]	WVALUE[3]	WVALUE[2]	WVALUE[1]	WVALUE[0]
	USBINDX	WINDEX[15]	WINDEX[14]	WINDEX[13]	WINDEX[12]	WINDEX[11]	WINDEX[10]	WINDEX[9]	WINDEX[8]
		WINDEX[7]	WINDEX[6]	WINDEX[5]	WINDEX[4]	WINDEX[3]	WINDEX[2]	WINDEX[1]	WINDEX[0]
	USBLENG	WLENGTH[15]	WLENGTH[14]	WLENGTH[13]	WLENGTH[12]	WLENGTH[11]	WLENGTH[10]	WLENGTH[9]	WLENGTH[8]
		WLENGTH[7]	WLENGTH[6]	WLENGTH[5]	WLENGTH[4]	WLENGTH[3]	WLENGTH[2]	WLENGTH[1]	WLENGTH[0]
	DCPCFG	—	—	—	—	—	—	—	CNTMD
		SHTNAK	—	—	DIR	—	—	—	—
	DCPMAXP	DEVSEL[1]	DEVSEL[0]	—	—	—	—	—	—
		—	MXPS[6]	MXPS[5]	MXPS[4]	MXPS[3]	MXPS[2]	MXPS[1]	MXPS[0]
	DCPCTR	BSTS	SUREQ	—	—	—	—	—	SQCLR
		SQSET	SQMON	—	—	—	CCPL	PID[1]	PID[0]
	PIPESEL	—	—	—	—	—	—	—	—
		—	—	—	—	—	PIPESEL[2]	PIPESEL[1]	PIPESEL[0]
	PIPECFG	TYPE[1]	TYPE[0]	—	—	—	BFRE	DBLB	CNTMD
		SHTNAK	—	—	DIR	EPNUM[3]	EPNUM[2]	EPNUM[1]	EPNUM[0]

模块名称	寄存器简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
USB	PIPEBUF	—	BUFSIZE [4]	BUFSIZE[3]	BUFSIZE[2]	BUFSIZE[1]	BUFSIZE[0]	—	—
		—	BUFNMB [6]	BUFNMB[5]	BUFNMB[4]	BUFNMB[3]	BUFNMB[2]	BUFNMB [1]	BUFNMB [0]
	PIPEMAXP	DEVSEL [1]	DEVSEL [0]	—	—	—	MXPS[10]	MXPS[9]	MXPS[8]
		MXPS[7]	MXPS[6]	MXPS[5]	MXPS[4]	MXPS[3]	MXPS[2]	MXPS[1]	MXPS[0]
	PIPEPERI	—	—	—	IFIS	—	—	—	—
		—	—	—	—	—	IITV[2]	IITV[1]	IITV[0]
	PIPE1CTR	BSTS	INBUFM	—	—	—	ATREPM	ACLRM	SQCLR
		SQSET	SQMON	—	—	—	—	PID[1]	PID[0]
	PIPE2CTR	BSTS	INBUFM	—	—	—	ATREPM	ACLRM	SQCLR
		SQSET	SQMON	—	—	—	—	PID[1]	PID[0]
	PIPE3CTR	BSTS	INBUFM	—	—	—	ATREPM	ACLRM	SQCLR
		SQSET	SQMON	—	—	—	—	PID[1]	PID[0]
	PIPE4CTR	BSTS	INBUFM	—	—	—	ATREPM	ACLRM	SQCLR
		SQSET	SQMON	—	—	—	—	PID[1]	PID[0]
	PIPE5CTR	BSTS	INBUFM	—	—	—	ATREPM	ACLRM	SQCLR
		SQSET	SQMON	—	—	—	—	PID[1]	PID[0]
	PIPE6CTR	BSTS	INBUFM	—	—	—	—	ACLRM	SQCLR
		SQSET	SQMON	—	—	—	—	PID[1]	PID[0]
	PIPE7CTR	BSTS	INBUFM	—	—	—	—	ACLRM	SQCLR
		SQSET	SQMON	—	—	—	—	PID[1]	PID[0]
	USBACSWR	—	—	—	—	—	—	—	—
		UACS23	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
LCDC	LDICKR	—	—	ICKSEL1	ICKSEL0	—	—	—	—
		—	—	DCDR5	DCDR4	DCRD3	DCRD2	DCRD1	DCDR0
	LDMTR	FLMPOL	CL1POL	DISPPOL	DPOL	—	MCNT	CL1CNT	CL2CNT
		—	—	MIFTYP5	MIFTYP4	MIFTYP3	MIFTYP2	MIFTYP1	MIFTYP0
	LDDFR	—	—	—	—	—	—	—	PABD
		—	DSPCOLOR 6	DSPCOLOR 5	DSPCOLOR 4	DSPCOLOR 3	DSPCOLOR 2	DSPCOLOR 1	DSPCOLOR 0
	LDSMR	—	—	ROT	—	—	—	AU1	AU0
		—	—	—	—	—	—	—	—
	LDSARU	—	—	—	—	—	—	SAU25	SAU24
		SAU23	SAU22	SAU21	SAU20	SAU19	SAU18	SAU17	SAU16
		SAU15	SAU14	SAU13	SAU12	SAU11	SAU10	SAU9	SAU8
		SAU7	SAU6	SAU5	SAU4	—	—	—	—

模块名称	寄存器简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
LCDC	LDSARL	—	—	—	—	—	—	SAL25	SAL24
		SAL23	SAL22	SAL21	SAL20	SAL19	SAL18	SAL17	SAL16
		SAL15	SAL14	SAL13	SAL12	SAL11	SAL10	SAL9	SAL8
		SAL7	SAL6	SAL5	SAL4	—	—	—	—
	LDLAOR	LAO15	LAO14	LAO13	LAO12	LAO11	LAO10	LAO9	LAO8
		LAO7	LAO6	LAO5	LAO4	LAO3	LAO2	LAO1	LAO0
	LDPALCR	—	—	—	—	—	—	—	—
		—	—	—	PALS	—	—	—	PALEN
	LDPRnn (nn=00 ~ FF)	—	—	—	—	—	—	—	—
		PALDnn23	PALDnn22	PALDnn21	PALDnn20	PALDnn19	PALDnn18	PALDnn17	PALDnn16
		PALDnn15	PALDnn14	PALDnn13	PALDnn12	PALDnn11	PALDnn10	PALDnn9	PALDnn8
		PALDnn7	PALDnn6	PALDnn5	PALDnn4	PALDnn3	PALDnn2	PALDnn1	PALDnn0
	LDHCNR	HDCN7	HDCN6	HDCN5	HDCN4	HDCN3	HDCN2	HDCN1	HDCN0
		HTCN7	HTCN6	HTCN5	HTCN4	HTCN3	HTCN2	HTCN1	HTCN0
	LDHSYNR	HSYNW3	HSYNW2	HSYNW1	HSYNW0	—	—	—	—
		HSYNP7	HSYNP6	HSYNP5	HSYNP4	HSYNP3	HSYNP2	HSYNP1	HSYNP0
	LDVDLNR	—	—	—	—	—	VDLN10	VDLN9	VDLN8
		VDLN7	VDLN6	VDLN5	VDLN4	VDLN3	VDLN2	VDLN1	VDLN0
	LDVTLNR	—	—	—	—	—	VTLN10	VTLN9	VTLN8
		VTLN7	VTLN6	VTLN5	VTLN4	VTLN3	VTLN2	VTLN1	VTLN0
	LDVSYNR	VSYNW3	VSYNW2	VSYNW1	VSYNW0	—	VSYNP10	VSYNP9	VSYNP8
		VSYNP7	VSYNP6	VSYNP5	VSYNP4	VSYNP3	VSYNP2	VSYNP1	VSYNP0
	LDACLNR	—	—	—	—	—	—	—	—
		—	—	—	ACLN4	ACLN3	ACLN2	ACLN1	ACLN0
	LDINTR	MINTEN	FINTEN	VSINTEN	VEINTEN	MINTS	FINTS	VSINTS	VEINTS
		—	—	—	—	—	—	—	—
	LDPMMR	OCN3	OCN2	OCN1	OCN0	OFFD3	OFFD2	OFFD1	OFFD0
		—	VCPE	VEPE	DONE	—	—	LPS1	LPS0
	LDPSPR	ONA3	ONA2	ONA1	ONA0	ONB3	ONB2	ONB1	ONB0
		OFFE3	OFFE2	OFFE1	OFFE0	OFFF3	OFFF2	OFFF1	OFFF0
	LDCNTR	—	—	—	—	—	—	—	—
		—	—	—	DON2	—	—	—	DON
	LDUINTR	—	—	—	—	—	—	—	UINTEN
		—	—	—	—	—	—	—	UINTS
	LDUINTLNR	—	—	—	—	—	UINTLN10	UINTLN9	UINTLN8
		UINTLN7	UINTLN6	UINTLN5	UINTLN4	UINTLN3	UINTLN2	UINTLN1	UINTLN0
	LDLIRNR	—	—	—	—	—	—	—	—
		LIRN7	LIRN6	LIRN5	LIRN4	LIRN3	LIRN2	LIRN1	LIRN0

模块名称	寄存器简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
PFC	PBIORL	—	—	—	—	PB11IOR	PB10IOR	PB9IOR	PB8IOR
		—	—	—	—	—	—	—	—
	PBCRL4	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	PB12MD[1]	PB12MD[0]
	PBCRL3	—	—	—	PB11MD0	—	—	—	PB10MD0
		—	—	PB9MD[1]	PB9MD[0]	—	—	PB8MD[1]	PB8MD[0]
	PBCRL2	—	—	PB7MD[1]	PB7MD[0]	—	—	PB6MD[1]	PB6MD[0]
		—	—	PB5MD[1]	PB5MD[0]	—	—	PB4MD[1]	PB4MD[0]
	PBCRL1	—	—	PB3MD[1]	PB3MD[0]	—	—	PB2MD[1]	PB2MD[0]
		—	—	PB1MD[1]	PB1MD[0]	—	—	PB0MD[1]	PB0MD[0]
	IFCR	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	PB12IRQ1	PB12IRQ0
	PCIORL	—	PC14IOR	PC13IOR	PC12IOR	PC11IOR	PC10IOR	PC9IOR	PC8IOR
		PC7IOR	PC6IOR	PC5IOR	PC4IOR	PC3IOR	PC2IOR	PC1IOR	PC0IOR
	PCCRL4	—	—	—	—	—	—	—	PC14MD[0]
		—	—	—	PC13MD[0]	—	—	—	PC12MD[0]
	PCCRL3	—	—	PC11MD[1]	PC11MD[0]	—	—	PC10MD[1]	PC10MD[0]
		—	—	—	PC9MD[0]	—	—	—	PC8MD[0]
	PCCRL2	—	—	—	PC7MD[0]	—	—	—	PC6MD[0]
		—	—	—	PC5MD[0]	—	—	—	PC4MD[0]
	PCCRL1	—	—	—	PC3MD[0]	—	—	—	PC2MD[0]
		—	—	—	PC1MD[0]	—	—	PC0MD[1]	PC0MD[0]
	PDIORL	PD15IOR	PD14IOR	PD13IOR	PD12IOR	PD11IOR	PD10IOR	PD9IOR	PD8IOR
		PD7IOR	PD6IOR	PD5IOR	PD4IOR	PD3IOR	PD2IOR	PD1IOR	PD0IOR
	PDCRL4	—	PD15MD[2]	PD15MD[1]	PD15MD[0]	—	PD14MD[2]	PD14MD[1]	PD14MD[0]
		—	PD13MD[2]	PD13MD[1]	PD13MD[0]	—	PD12MD[2]	PD12MD[1]	PD12MD[0]
	PDCRL3	—	PD11MD[2]	PD11MD[1]	PD11MD[0]	—	PD10MD[2]	PD10MD[1]	PD10MD[0]
		—	PD9MD[2]	PD9MD[1]	PD9MD[0]	—	PD8MD[2]	PD8MD[1]	PD8MD[0]
	PDCRL2	—	PD7MD[2]	PD7MD[1]	PD7MD[0]	—	PD6MD[2]	PD6MD[1]	PD6MD[0]
		—	PD5MD[2]	PD5MD[1]	PD5MD[0]	—	PD4MD[2]	PD4MD[1]	PD4MD[0]
	PDCRL1	—	PD3MD[2]	PD3MD[1]	PD3MD[0]	—	PD2MD[2]	PD2MD[1]	PD2MD[0]
		—	PD1MD[2]	PD1MD[1]	PD1MD[0]	—	PD0MD[2]	PD0MD[1]	PD0MD[0]
	PEIORL	PE15IOR	PE14IOR	PE13IOR	PE12IOR	PE11IOR	PE10IOR	PE9IOR	PE8IOR
		PE7IOR	PE6IOR	PE5IOR	PE4IOR	PE3IOR	PE2IOR	PE1IOR	PE0IOR
	PECRL4	—	—	PE15MD[1]	PE15MD[0]	—	—	PE14MD[1]	PE14MD[0]
		—	—	PE13MD[1]	PE13MD[0]	—	—	PE12MD[1]	PE12MD[0]
	PECRL3	—	PE11MD[2]	PE11MD[1]	PE11MD[0]	—	PE10MD[2]	PE10MD[1]	PE10MD[0]
		—	—	PE9MD[1]	PE9MD[0]	—	—	PE8MD[1]	PE8MD[0]

模块名称	寄存器简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
PFC	PECRL2	—	PE7MD[2]	PE7MD[1]	PE7MD[0]	—	PE6MD[2]	PE6MD[1]	PE6MD[0]
		—	PE5MD[2]	PE5MD[1]	PE5MD[0]	—	PE4MD[2]	PE4MD[1]	PE4MD[0]
	PECRL1	—	—	PE3MD[1]	PE3MD[0]	—	—	PE2MD[1]	PE2MD[0]
		—	—	PE1MD[1]	PE1MD[0]	—	PE0MD[2]	PE0MD[1]	PE0MD[0]
	PFIORH	—	PF30IOR	PF29IOR	PF28IOR	PF27IOR	PF26IOR	PF25IOR	PF24IOR
		—	PF23IOR	PF22IOR	PF21IOR	PF20IOR	PF19IOR	PF18IOR	PF16IOR
	PFIORL	PF15IOR	PF14IOR	PF13IOR	PF12IOR	PF11IOR	PF10IOR	PF9IOR	PF8IOR
		PF7IOR	PF6IOR	PF5IOR	PF4IOR	PF3IOR	PF2IOR	PF1IOR	PF0IOR
	PFCRH4	—	—	—	—	—	—	—	PF30MD[0]
		—	—	—	PF29MD[0]	—	—	—	PF28MD[0]
	PFCRH3	—	—	—	PF27MD[0]	—	—	—	PF26MD[0]
		—	—	—	PF25MD[0]	—	—	—	PF24MD[0]
	PFCRH2	—	—	PF23MD[1]	PF23MD[0]	—	—	PF22MD[1]	PF22MD[0]
		—	—	PF21MD[1]	PF21MD[0]	—	—	PF20MD[1]	PF20MD[0]
	PFCRH1	—	—	PF19MD[1]	PF19MD[0]	—	—	PF18MD[1]	PF18MD[0]
		—	—	PF17MD[1]	PF17MD[0]	—	—	PF16MD[1]	PF16MD[0]
	PFCRL4	—	—	PF15MD[1]	PF15MD[0]	—	—	PF14MD[1]	PF14MD[0]
		—	—	PF13MD[1]	PF13MD[0]	—	—	PF12MD[1]	PF12MD[0]
	PFCRL3	—	—	PF11MD[1]	PF11MD[0]	—	—	PF10MD[1]	PF10MD[0]
		—	—	PF9MD[1]	PF9MD[0]	—	—	PF8MD[1]	PF8MD[0]
	PFCRL2	—	—	PF7MD[1]	PF7MD[0]	—	—	PF6MD[1]	PF6MD[0]
		—	—	PF5MD[1]	PF5MD[0]	—	—	PF4MD[1]	PF4MD[0]
	PFCRL1	—	—	PF3MD[1]	PF3MD[0]	—	—	PF2MD[1]	PF2MD[0]
		—	—	PF1MD[1]	PF1MD[0]	—	—	PF0MD[1]	PF0MD[0]
	SCSR	—	S3CKS2	S3CKS1	S3CKS0	—	S2CKS2	S2CKS1	S2CKS0
		—	S1CKS2	S1CKS1	S1CKS0	—	S0CKS2	S0CKS1	S0CKS0
I/O 端口	PADRL	—	—	—	—	—	—	—	—
		PA7DR	PA6DR	PA5DR	PA4DR	PA3DR	PA2DR	PA1DR	PA0DR
	PBDRL	—	—	—	PB12DR	PB11DR	PB10DR	PB9DR	PB8DR
		PB7DR	PB6DR	PB5DR	PB4DR	PB3DR	PB2DR	PB1DR	PB0DR
	PBPRL	—	—	—	—	PB11PR	PB10PR	PB9PR	PB8PR
		PB7PR	PB6PR	PB5PR	PB4PR	PB3PR	PB2PR	PB1PR	PB0PR
	PCDRL	—	PC14DR	PC13DR	PC12DR	PC11DR	PC10DR	PC9DR	PC8DR
		PC7DR	PC6DR	PC5DR	PC4DR	PC3DR	PC2DR	PC1DR	PC0DR
	PCPRL	—	PC14PR	PC13PR	PC12PR	PC11PR	PC10PR	PC9PR	PC8PR
		PC7PR	PC6PR	PC5PR	PC4PR	PC3PR	PC2PR	PC1PR	PC0PR
	PDDRL	PD15DR	PD14DR	PD13DR	PD12DR	PD11DR	PD10DR	PD9DR	PD8DR
		PD7DR	PD6DR	PD5DR	PD4DR	PD3DR	PD2DR	PD1DR	PD0DR

模块名称	寄存器简称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
I/O 端口	PDPRL	PD15PR	PD14PR	PD13PR	PD12PR	PD11PR	PD10PR	PD9PR	PD8PR
		PD7PR	PD6PR	PD5PR	PD4PR	PD3PR	PD2PR	PD1PR	PD0PR
	PEDRL	PE15DR	PE14DR	PE13DR	PE12DR	PE11DR	PE10DR	PE9DR	PE8DR
		PE7DR	PE6DR	PE5DR	PE4DR	PE3DR	PE2DR	PE1DR	PE0DR
	PEPRL	PE15PR	PE14PR	PE13PR	PE12PR	PE11PR	PE10PR	PE9PR	PE8PR
		PE7PR	PE6PR	PE5PR	PE4PR	PE3PR	PE2PR	PE1PR	PE0PR
	PFDRH	—	PF30DR	PF29DR	PF28DR	PF27DR	PF26DR	PF25DR	PF24DR
		PF23DR	PF22DR	PF21DR	PF20DR	PF19DR	PF18DR	PF17DR	PF16DR
	PFDRL	PF15DR	PF14DR	PF13DR	PF12DR	PF11DR	PF10DR	PF9DR	PF8DR
		PF7DR	PF6DR	PF5DR	PF4DR	PF3DR	PF2DR	PF1DR	PF0DR
	PFPRH	—	PF30PR	PF29PR	PF28PR	PF27PR	PF26PR	PF25PR	PF24PR
		PF23PR	PF22PR	PF21PR	PF20PR	PF19PR	PF18PR	PF17PR	PF16PR
低功耗模式	PFPRL	PF15PR	PF14PR	PF13PR	PF12PR	PF11PR	PF10PR	PF9PR	PF8PR
		PF7PR	PF6PR	PF5PR	PF4PR	PF3PR	PF2PR	PF1PR	PF0PR
	STBCR	STBY	DEEP	—	—	—	—	—	—
	STBCR2	MSTP10	MSTP9	MSTP8	MSTP7	—	—	—	—
	STBCR3	HIZ	—	MSTP35	—	—	MSTP32	MSTP31	MSTP30
	STBCR4	MSTP47	MSTP46	MSTP45	MSTP44	MSTP43	MSTP42	MSTP41	MSTP40
	STBCR5	MSTP57	MSTP56	MSTP55	MSTP54	MSTP53	MSTP52	MSTP51	MSTP50
	STBCR6	MSTP67	MSTP66	MSTP65	MSTP64	—	—	—	MSTP60
	SYSCR1	—	—	—	—	RAME3	RAME2	RAME1	RAME0
	SYSCR2	—	—	—	—	RAMWE3	RAMWE2	RAMWE1	RAMWE0
	SYSCR3	AXTALE	—	—	—	SSI3SRST	SSI2SRST	SSI1SRST	SSI0SRST
	DSCTR	—	—	—	—	RAMKP3	RAMKP2	RAMKP1	RAMKP0
	DSCTR2	CS0KEEPE	RAMBOOT	—	—	—	—	—	—
	DS SSR	—	—	—	—	—	—	—	MRES
		IRQ7	IRQ6	IRQ5	IRQ4	IRQ3	IRQ2	IRQ1	IRQ0
	DSFR	IOKEEP	—	—	—	—	—	MRESF	NMIF
		IRQ7F	IRQ6F	IRQ5F	IRQ4F	IRQ3F	IRQ2F	IRQ1F	IRQ0F
	DSRTR	—	TRMD[6]	TRMD[5]	TRMD[4]	TRMD[3]	TRMD[2]	TRMD[1]	TRMD[0]
H- UDI	SDIR	T1[7]	T1[6]	T1[5]	T1[4]	T1[3]	T1[2]	T1[1]	T1[0]
		—	—	—	—	—	—	—	—

【注】 *1 普通空间、带字节选择的 SRAM 及地址 / 数据多路复用 I/O(MPX-IO) 时, 指定存储器种类。

*2 指定存储器种类为突发 ROM (时钟异步) 时。

*3 指定存储器种类为突发 ROM (时钟同步) 时。

*4 指定存储器种类为普通空间、带字节选择的 SRAM 时。

*5 指定存储器种类为 SDRAM 时。

*6 指定存储器种类为 PCMCIA 时。

*7 指定存储器种类为突发 MPX-IO 时。

*8 MCR15=0 时。

*9 MCR15=1 时。

*10 命令存取模式时。

*11 扇区存取模式时。

30.3 各运行模式的寄存器状态一览表

模块	寄存器名称	上电复位	手动复位	深度待机	软件待机	模块待机	睡眠
CPG	FRQCR	初始化 *1	保持	初始化	保持	—	保持
INTC	IBNR	初始化	保持 *2	初始化	保持	—	保持
	除上述之外的所有寄存器	初始化	保持	初始化	保持	—	保持
UBC	所有寄存器	初始化	保持	初始化	保持	保持	保持
高速缓存	所有寄存器	初始化	保持	初始化	保持	—	保持
BSC	RTCSR	初始化	保持 *3	初始化	保持	—	保持 *3
	RTCNT	初始化	保持 *4	初始化	保持	—	保持 *4
	除上述之外的所有寄存器	初始化	保持	初始化	保持	—	保持
DMAC	所有寄存器	初始化	保持	初始化	保持	保持	保持 *5
MTU2	所有寄存器	初始化	保持	初始化	保持	初始化	保持
CMT	所有寄存器	初始化	保持	初始化	初始化	保持	保持
WDT	WRCSR	初始化 *1	保持	初始化	保持	—	保持
	除上述之外的所有寄存器	初始化	保持	初始化	保持	—	保持
RTC	R64CNT	保持 *4	保持 *4	保持 *4	保持 *4	保持	保持 *4
	RSECCNT						
	RMINCNT						
	RHRCNT						
	RWKCNT						
	RDAYCNT						
	RMONCNT						
	RYRCNT						
	RSECAR	初始化	保持	初始化	保持	保持	保持
	RMINAR						
	RHRAR						
	RWKAR						
	RDAYAR						
	RMONAR						
	RYRAR						
	RCR1	初始化	初始化	初始化	保持	保持	保持
	RCR2	初始化	初始化 *6	初始化	保持	保持	保持
	RCR3	初始化	保持	初始化	保持	保持	保持
	全部寄存器	初始化	保持	初始化	保持	保持	保持

模块	寄存器简称	上电复位	手动复位	深度待机	软件待机	模块待机	睡眠
SCIF	全部寄存器	初始化	保持	初始化	保持	保持	保持
SSU	全部寄存器	初始化	保持	初始化	初始化	初始化	保持
IIC3	ICMR_0、1、2、3	初始化	保持	初始化	保持 *7	保持 *7	保持
	除上述之外的所有寄存器	初始化	保持	初始化	保持	保持	保持
SSI	所有寄存器	初始化	保持	初始化	保持	保持	保持
RCAN-TL1	所有寄存器	初始化	保持	初始化	保持	保持	保持
ADC	所有寄存器	初始化	保持	初始化	初始化	初始化	保持
DAC	所有寄存器	初始化	保持	初始化	保持	初始化	保持
FLCTL	所有寄存器	初始化	保持	初始化	保持	保持	保持
USB	所有寄存器	初始化	保持	初始化	保持	保持	保持
LCDC	所有寄存器	初始化	保持	初始化	保持	保持	保持
PFC	所有寄存器	初始化	保持	初始化	保持	—	保持
I/O 端口	所有寄存器 *8	初始化	保持	初始化	保持	—	保持
低功耗模式	STBCR	初始化	保持	初始化	保持	—	保持
	STBCR2	初始化	保持	初始化	保持	—	保持
	SYSCR1	初始化	保持	初始化	保持	—	保持
	SYSCR2	初始化	保持	初始化	保持	—	保持
	SYSCR3	初始化	保持	初始化	保持	—	保持
	STBCR3	初始化	保持	初始化	保持	—	保持
	STBCR4	初始化	保持	初始化	保持	—	保持
	STBCR5	初始化	保持	初始化	保持	—	保持
	STBCR6	初始化	保持	初始化	保持	—	保持
	DSCTR	初始化	保持	初始化	保持	—	保持
	DSCTR2	初始化	保持	保持	保持	—	保持
	DSSSR	初始化	保持	初始化	保持	—	保持
	DSFR	初始化	保持	保持	保持	—	保持
	DSRTR	初始化 *10	保持	初始化	保持	—	保持
H-UDI*9	SDIR	保持	保持	初始化	保持	保持	保持

【注】 *1 由 WDT 引起内部上电复位时，为保持。

*2 初始化 BN[3:0] 位。

*3 继续执行标志处理。

*4 继续递增计数。

*5 可继续传送。

*6 RTCEN、START 位为保持。

*7 初始化 BC[3:0] 位。

*8 PADRL 及端口寄存器用于读取引脚状态，因此既不初始化也不保持。

*9 在 $\overline{\text{TRST}}$ 有效或 TAP 的 Test-Logic-Reset 状态时初始化。

*10 在 RES 有效时初始化，并在执行 H-UDI 复位有效命令或由 WDT 引起内部上电复位时保持。

第 31 章 电特性

31.1 绝对最大额定值

表 31.1 绝对最大额定值

项 目		符号	额定值	单位
电源电压 (I/O)		PV_{CC}	$-0.3 \sim 4.6$	V
电源电压 (内部)		V_{CC}	$-0.3 \sim 1.7$	V
PLL 电源电压		$PLL V_{CC}$	$-0.3 \sim 1.7$	V
模拟电源电压		AV_{CC}	$-0.3 \sim 4.6$	V
模拟基准电压		AV_{ref}	$-0.3 \sim AV_{CC}+0.3$	V
USB 收发器部模拟电源电压 (I/O)		$USBAPV_{CC}$	$-0.3 \sim 4.6$	V
USB 收发器部数字电源电压 (I/O)		$USBDPV_{CC}$	$-0.3 \sim 4.6$	V
USB 收发器部模拟电源电压 (内部)		$USBAV_{CC}$	$-0.3 \sim 1.7$	V
USB 收发器部数字电源电压 (内部)		$USBDV_{CC}$	$-0.3 \sim 1.7$	V
输入电压	模拟输入引脚	V_{AN}	$-0.3 \sim AV_{CC}+0.3$	V
	VBUS	V_{in}	$-0.3 \sim 5.5$	V
	其他输入引脚	V_{in}	$-0.3 \sim PV_{CC}+0.3$	V
工作温度		T_{opr}	$-20 \sim 85$	°C
保存温度		T_{stg}	$-55 \sim 125$	°C

【使用时的注意事项】

超过绝对最大额定值使用时，可能导致 LSI 的永久性破坏。

31.2 上电、断电顺序

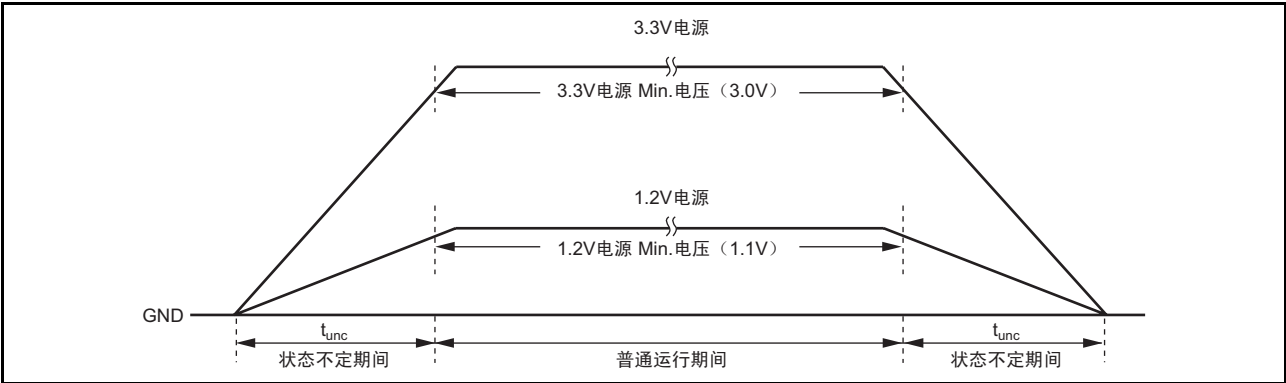


图 31.1 上电 / 断电顺序

表 31.2 上电 / 断电时间

项 目	符号	Min.	Max.	单位
状态不定时间	t_{unc}	—	100	ms

【注】 请尽量同时接通 / 切断 1.2V 电源（ V_{CC} 、 $PLL V_{CC}$ 、 $USBA V_{CC}$ 、 $USBD V_{CC}$ ）、3.3V 电源（ PV_{CC} 、 AV_{CC} 、 $USBA PV_{CC}$ 、 $USBD PV_{CC}$ ）。从开始上电到二者电源超过 Min. 电压，或某一方电源降至 Min. 电压以下直至为 0V 的期间为状态不定期间。此期间，输出引脚及输入 / 输出引脚的状态及内部状态为不定状态，因此，设计电源电路时应尽可能将此期间缩短。另外，系统设计时应避免因此状态引起整个系统的错误运行。

31.3 DC 特性

表 31.3 DC 特性 (1) 【通用项目】

条件: $V_{CC}=PLL V_{CC}=USBDV_{CC}=1.1 \sim 1.3V$ 、 $PV_{CC}=USBDPV_{CC}=3.0 \sim 3.6V$ 、 $AV_{CC}=3.0 \sim 3.6V$ 、
 $USBAV_{CC}=1.1 \sim 1.3V$ 、 $USBAPV_{CC}=3.0 \sim 3.6V$ 、
 $V_{SS}=PLL V_{SS}=PV_{SS}=AV_{SS}=USBDV_{SS}=USBAV_{SS}=USBDPV_{SS}=USBAPV_{SS}=0V$ 、 $T_a=-20 \sim 85^{\circ}C$

项 目		符号	Min.	Typ.	Max.	单位	测量条件
电源电压		PV_{CC}	3.0	3.3	3.6	V	
		V_{CC}	1.1	1.2	1.3	V	
PLL 电源电压		$PLL V_{CC}$	1.1	1.2	1.3	V	
模拟电源电压		AV_{CC}	3.0	3.3	3.6	V	
USB 电源电压		$USBAPV_{CC}$ $USBDPV_{CC}$	3.0	3.3	3.6	V	
		$USBAV_{CC}$ $USBDV_{CC}$	1.1	1.2	1.3	V	
消耗电流 *1	普通运行时	I_{CC}^{*2}	—	240	400	mA	$V_{CC}=1.2V$ $I_{\phi}=200.00MHz$
	睡眠模式时	I_{sleep}^{*2}	—	180	360	mA	$B_{\phi}=66.66MHz$ $P_{\phi}=33.33MHz$
	软件待机模式时	I_{sstby}^{*2}	—	12	120	mA	$T_a > 50^{\circ}C$ $V_{CC}=1.2V$
			—	4	40	mA	$T_a \leq 50^{\circ}C$ $V_{CC}=1.2V$
	深度待机模式时	I_{dstby}	—	5	30	μA	$T_a > 50^{\circ}C$ 1.2V 电源 *3=1.2V RAM 0KB 保持
			—	23	130	μA	$T_a > 50^{\circ}C$ 1.2V 电源 *3=1.2V RAM 4KB 保持
			—	41	230	μA	$T_a > 50^{\circ}C$ 1.2V 电源 *3=1.2V RAM 8KB 保持
			—	59	330	μA	$T_a > 50^{\circ}C$ 1.2V 电源 *3=1.2V RAM 12KB 保持
			—	77	430	μA	$T_a > 50^{\circ}C$ 1.2V 电源 *3=1.2V RAM 16KB 保持
			—	9	58	μA	$T_a > 50^{\circ}C$ 3.3V 电源 *4=3.3V
			—	11	12	μA	$T_a > 50^{\circ}C$ VBUS=5.0V

项 目		符号	Min.	Typ.	Max.	单位	测量条件
消耗电流 *1	深度待机模式时	I_{dstby}	—	2	10	μA	$T_a \leq 50^\circ\text{C}$ 1.2V 电源 *3=1.2V RAM 0KB 保持
			—	12	24	μA	$T_a \leq 50^\circ\text{C}$ 1.2V 电源 *3=1.2V RAM 8KB 保持
			—	22	38	μA	$T_a \leq 50^\circ\text{C}$ 1.2V 电源 *3=1.2V RAM 8KB 保持
			—	32	52	μA	$T_a \leq 50^\circ\text{C}$ 1.2V 电源 *3=1.2V RAM 12KB 保持
			—	42	66	μA	$T_a \leq 50^\circ\text{C}$ 1.2V 电源 *3=1.2V RAM 16KB 保持
			—	5	20	μA	$T_a \leq 50^\circ\text{C}$ 3.3V 电源 *4=3.3V
			—	11	12	μA	$T_a \leq 50^\circ\text{C}$ VBUS=5.0V
输入（泄）漏电流	所有输入引脚	$ I_{\text{in}} $	—	—	1.0	μA	$V_{\text{in}}=0.5 \sim \text{PV}_{\text{CC}}-0.5\text{V}$
三态（泄）漏电流	所有输入 / 输出引脚 （除 PB7 ~ PB0 及弱保持引脚） （断路状态）	$ I_{\text{STI}} $	—	—	1.0	μA	$V_{\text{in}}=0.5 \sim \text{PV}_{\text{CC}}-0.5\text{V}$
	PB7 ~ PB0		—	—	10	μA	
引脚电容	所有引脚	C_{in}	—	—	20	pF	
模拟电源电流	A/D、D/A 转换时	I_{CC}	—	2	4	mA	
	A/D、D/A 转换待机时		—	1	10	μA	
模拟基准电压电流		I_{ref}	—	2	4	mA	
USB 电源电压	USBAV _{CC} +USBDV _{CC}	I_{USBCC}	—	15	20	mA	USBAV _{CC} =USBDV _{CC} =1.2V
	USBAPV _{CC} +USBDPV _{CC}	I_{USBPCC}	—	40	50	mA	USBAPV _{CC} =USBDPV _{CC} =3.3V

【注意事项】 不使用 A/D 转换器及 D/A 转换器时，不得将 AV_{CC}、AV_{SS} 引脚置为开路。

【注】 *1 消耗电流是将所有输出引脚及带上拉的引脚置为无负载状态时的值。

*2 I_{CC} 、 I_{sleep} 、 I_{sstby} 为在 V_{CC}、PLL_{VCC} 系统消耗电流的合计值。

*3 I_{dstby} 的 1.2V 电流为在 V_{CC}、PLL_{VCC}、USBAV_{CC}、USBDV_{CC} 消耗电流的合计值。

*4 I_{dstby} 的 3.3V 电流为在 PV_{CC}、AV_{CC}、USBAPV_{CC}、USBDPV_{CC} 消耗电流的合计值。

表 31.3 DC 特性 (2) 【除 I²C、USB 相关引脚之外】

条件: $V_{CC}=PLL_{V_{CC}}=USBDV_{CC}=1.1 \sim 1.3V$ 、 $PV_{CC}=USBDPV_{CC}=3.0 \sim 3.6V$ 、 $AV_{CC}=3.0 \sim 3.6V$ 、
 $USBAV_{CC}=1.1 \sim 1.3V$ 、 $USBAPV_{CC}=3.0 \sim 3.6V$ 、
 $V_{SS}=PLL_{V_{SS}}=PV_{SS}=AV_{SS}=USBDV_{SS}=USBAV_{SS}=USBDPV_{SS}=USBAPV_{SS}=0V$ 、 $T_a = -20 \sim 85^{\circ}C$

项 目	符号	Min.	Typ.	Max.	单位	测量条件
输入高电平电压	RES、MRES、NMI、MD、 MD_CLK1、MD_CLK0、 ASEMD、TRST、EXTAL、 CKIO、AUDIO_X1、 RTC_X1	$PV_{CC}-0.5$	—	$PV_{CC}+0.3$	V	
	PA7 ~ PA0	2.2	—	$AV_{CC}+0.3$	V	
	其他输入引脚 (除施密特引脚之外)	2.2	—	$PV_{CC}+0.3$	V	
输入低电平电压	RES、MRES、NMI、MD、 MD_CLK1、MD_CLK0、 ASEMD、TRST、EXTAL、 CKIO、AUDIO_X1、 RTC_X1	-0.3	—	0.5	V	
	其他输入引脚 (除施密特引脚之外)	-0.3	—	0.8	V	
施密特触发输入特性	IRQ7 ~ IRQ0、 PINT7 ~ PINT0、 IOIS16、 DREQ3 ~ DREQ0、 TIOC0A ~ TIOC0D、 TIOC1A、TIOC1B、 TIOC2A、TIOC2B、 TIOC3A ~ TIOC3D、 TIOC4A ~ TIOC4D、 TCLKA ~ TCLKD、 SCK3 ~ SCK0、 RxD3 ~ RxD0、 CTS3、RTS3、 SSCK1、SSCK0、 SSI1、SSI0、 SSO1、SSO0、 SCS1、SCS0、 ADTRG、 PE15 ~ PE0、 PF7 ~ PF0	V_{T+}	$PV_{CC}-0.5$	—	V	
		V_{T-}	—	0.5	V	
		$V_{T+} - V_{T-}$	0.2	—	V	
输出高电平电压		V_{OH}	$PV_{CC}-0.5$	—	V	$I_{OH}=-200\mu A$
输出低电平电压		V_{OL}	—	0.4	V	$I_{OL}=1.6mA$
RAM 待机电压	软件待机模式时	V_{RAMS}	0.75	—	V	将 V_{CC} ($=PLL_{V_{CC}}$) 作为参数测量
	深度待机模式时 (仅限用于保持内部 RAM)	V_{RAMD}	1.1	—	V	

表 31.3 DC 特性 (3) 【I²C 相关引脚*】

条件: $V_{CC}=PLL_{V_{CC}}=USBDV_{CC}=1.1 \sim 1.3V$ 、 $PV_{CC}=USBDPV_{CC}=3.0 \sim 3.6V$ 、 $AV_{CC}=3.0 \sim 3.6V$ 、
 $USBAV_{CC}=1.1 \sim 1.3V$ 、 $USBAPV_{CC}=3.0 \sim 3.6V$ 、
 $V_{SS}=PLL_{V_{SS}}=PV_{SS}=AV_{SS}=USBDV_{SS}=USBAV_{SS}=USBDPV_{SS}=USBAPV_{SS}=0V$ 、 $T_a=-20 \sim 85^{\circ}C$

项 目	符号	Min.	Typ.	Max.	单位	测量条件
输入高电平电压	V_{IH}	$PV_{CC} \times 0.7$	—	$PV_{CC} + 0.3$	V	
输入低电平电压	V_{IL}	-0.3	—	$PV_{CC} \times 0.3$	V	
施密特触发输入特性	$V_{IH} - V_{IL}$	$PV_{CC} \times 0.05$	—	—	V	
输出低电平电压	V_{OL}	—	—	0.4	V	$I_{OL}=3.0mA$

【注】 * PB7/SDA3/PINT7/IRQ7 ~ PB0/SCL0/PINT0/IRQ0 引脚（开路漏极引脚）

表 31.3 DC 特性 (4) 【USB 相关引脚*】

条件: $V_{CC}=PLL_{V_{CC}}=USBDV_{CC}=1.1 \sim 1.3V$ 、 $PV_{CC}=USBDPV_{CC}=3.0 \sim 3.6V$ 、 $AV_{CC}=3.0 \sim 3.6V$ 、
 $USBAV_{CC}=1.1 \sim 1.3V$ 、 $USBAPV_{CC}=3.0 \sim 3.6V$ 、
 $V_{SS}=PLL_{V_{SS}}=PV_{SS}=AV_{SS}=USBDV_{SS}=USBAV_{SS}=USBDPV_{SS}=USBAPV_{SS}=0V$ 、 $T_a=-20 \sim 85^{\circ}C$

项 目	符号	Min.	Typ.	Max.	单位	测量条件
基准电阻	R_{REF}	5.6k Ω $\pm 1\%$				
输入高电平电压 (VBUS)	V_{IH}	4.02	—	5.25	V	
输入低电平电压 (VBUS)	V_{IL}	-0.3	—	0.5	V	
输入高电平电压 (USB_X1)	V_{IH}	$PV_{CC} - 0.5$	—	$PV_{CC} + 0.3$	V	
输入低电平电压 (USB_X1)	V_{IL}	-0.3	—	0.5	V	

【注】 * REFRIN、VBUS、USB_X1、USB_X2 引脚

表 31.3 DC 特性 (5) 【USB 相关引脚*（全速 / 高速通用项目）】

条件: $V_{CC}=PLL_{V_{CC}}=USBDV_{CC}=1.1 \sim 1.3V$ 、 $PV_{CC}=USBDPV_{CC}=3.0 \sim 3.6V$ 、 $AV_{CC}=3.0 \sim 3.6V$ 、
 $USBAV_{CC}=1.1 \sim 1.3V$ 、 $USBAPV_{CC}=3.0 \sim 3.6V$ 、
 $V_{SS}=PLL_{V_{SS}}=PV_{SS}=AV_{SS}=USBDV_{SS}=USBAV_{SS}=USBDPV_{SS}=USBAPV_{SS}=0V$ 、 $T_a=-20 \sim 85^{\circ}C$

项 目	符号	Min.	Typ.	Max.	单位	测量条件
DP 上拉电阻 (选择功能控制器功能时)	R_{pu}	0.900	—	1.575	k Ω	空闲时
		1.425	—	3.090	k Ω	发送 / 接收时
DP、DM 下拉电阻 (选择主机控制器功能时)	R_{pd}	14.25	—	24.80	k Ω	

【注】 * DP、DM 引脚

表 31.3 DC 特性 (6) 【USB 相关引脚 * (全速时)】

条件: $V_{CC}=PLL_{V_{CC}}=USBDV_{CC}=1.1 \sim 1.3V$ 、 $PV_{CC}=USBDPV_{CC}=3.0 \sim 3.6V$ 、 $AV_{CC}=3.0 \sim 3.6V$ 、
 $USBAV_{CC}=1.1 \sim 1.3V$ 、 $USBAPV_{CC}=3.0 \sim 3.6V$ 、
 $V_{SS}=PLL_{V_{SS}}=PV_{SS}=AV_{SS}=USBDV_{SS}=USBAV_{SS}=USBDPV_{SS}=USBAPV_{SS}=0V$ 、 $T_a=-20 \sim 85^{\circ}C$

项 目	符号	Min.	Typ.	Max.	单位	测量条件
输入高电平电压	V_{IH}	2.0	—	—	V	
输入低电平电压	V_{IL}	—	—	0.8	V	
差动输入灵敏度	V_{DI}	0.2	—	—	V	(DP)-(DM)
差动共模范围	V_{CM}	0.8	—	2.5	V	
输出高电平电压	V_{OH}	2.8	—	3.6	V	$I_{OH}=-200\mu A$
输出低电平电压	V_{OL}	0.0	—	0.3	V	$I_{OL}=2mA$
输出信号交叉电压	V_{CRS}	1.3	—	2.0	V	$C_L=50pF$

【注】 * DP、DM 引脚

表 31.3 DC 特性 (7) 【USB 相关引脚 * (高速时)】

条件: $V_{CC}=PLL_{V_{CC}}=USBDV_{CC}=1.1 \sim 1.3V$ 、 $PV_{CC}=USBDPV_{CC}=3.0 \sim 3.6V$ 、 $AV_{CC}=3.0 \sim 3.6V$ 、
 $USBAV_{CC}=1.1 \sim 1.3V$ 、 $USBAPV_{CC}=3.0 \sim 3.6V$ 、
 $V_{SS}=PLL_{V_{SS}}=PV_{SS}=AV_{SS}=USBDV_{SS}=USBAV_{SS}=USBDPV_{SS}=USBAPV_{SS}=0V$ 、 $T_a=-20 \sim 85^{\circ}C$

项 目	符号	Min.	Typ.	Max.	单位	测量条件
噪声抑制电路检测阈值电压 (差动电压)	V_{HSSQ}	100	—	150	mV	
共模电压范围	V_{HSCM}	-50	—	500	mV	
空闲状态	V_{HSOI}	-10.0	—	10.0	mV	
输出高电平电压	V_{HSOH}	360	—	440	mV	
输出低电平电压	V_{HSOL}	-10.0	—	10.0	mV	
Chirp J 输出电压 (差分)	V_{CHIRPJ}	700	—	1100	mV	
Chirp K 输出电压 (差分)	V_{CHIRPK}	-900	—	-500	mV	

【注】 * DP、DM 引脚

表 31.4 输出容许电流值

条件: $V_{CC}=PLL_{V_{CC}}=USBDV_{CC}=1.1 \sim 1.3V$ 、 $PV_{CC}=USBDPV_{CC}=3.0 \sim 3.6V$ 、 $AV_{CC}=3.0 \sim 3.6V$ 、
 $USBAV_{CC}=1.1 \sim 1.3V$ 、 $USBAPV_{CC}=3.0 \sim 3.6V$ 、
 $V_{SS}=PLL_{V_{SS}}=PV_{SS}=AV_{SS}=USBDV_{SS}=USBAV_{SS}=USBDPV_{SS}=USBAPV_{SS}=0V$ 、 $T_a=-20 \sim 85^{\circ}C$

项 目	符号	Min.	Typ.	Max.	单位
输出低电平容许电流 (1 个引脚)	PB7 ~ PB0	I_{OL}	—	10	mA
	其他输出引脚			2	mA
输出低电平容许电流 (总和)	ΣI_{OL}	—	—	150	mA
输出高电平容许电流 (每个引脚)	$-I_{OH}$	—	—	2	mA
输出高电平容许电流 (总和)	$\Sigma -I_{OH}$	—	—	50	mA

【注意事项】为了确保 LSI 的可靠性, 输出电流值不得超出表 31.4 所列的值。

31.4 AC 特性

本 LSI 的输入原则为时钟同步输入。无特殊理由，必须遵守各输入信号的准备 / 保持时间。

表 31.5 最大工作频率

条件: $V_{CC}=PLL_{VCC}=USBDV_{CC}=1.1 \sim 1.3V$ 、 $PV_{CC}=USBDPV_{CC}=3.0 \sim 3.6V$ 、 $AV_{CC}=3.0 \sim 3.6V$ 、
 $USBAV_{CC}=1.1 \sim 1.3V$ 、 $USBAPV_{CC}=3.0 \sim 3.6V$ 、
 $V_{SS}=PLL_{VSS}=PV_{SS}=AV_{SS}=USBDV_{SS}=USBAV_{SS}=USBDPV_{SS}=USBAPV_{SS}$ $T_a=-20 \sim 85^{\circ}C$

项 目	符号	Min.	Max.	单位	备注
工作频率	CPU (I_{ϕ})	80.00	200.00	MHz	
	内部总线、外部总 (B_{ϕ})	40.00	66.66	MHz	
	外围模块 (P_{ϕ})	6.66	33.33	MHz	

31.4.1 时钟时序

表 31.6 时钟时序

条件: $V_{CC}=PLL_{VCC}=USBDV_{CC}=1.1 \sim 1.3V$ 、 $PV_{CC}=USBDPV_{CC}=3.0 \sim 3.6V$ 、 $AV_{CC}=3.0 \sim 3.6V$ 、
 $USBAV_{CC}=1.1 \sim 1.3V$ 、 $USBAPV_{CC}=3.0 \sim 3.6V$ 、
 $V_{SS}=PLL_{VSS}=PV_{SS}=AV_{SS}=USBDV_{SS}=USBAV_{SS}=USBDPV_{SS}=USBAPV_{SS}=0V$ 、 $T_a=-20 \sim 85^{\circ}C$

项 目	符号	Min.	Max.	单位	参照图
EXTAL 时钟输入频率	f_{EX}	10.00	33.33	MHz	31.2
EXTAL 时钟输入周期时间	t_{EXcyc}	30	100	ns	
AUDIO_X1 时钟输入频率 (用于连接晶体振荡器)	f_{EX}	10	40	MHz	
AUDIO_X1 时钟输入周期时间 (用于连接晶体振荡器)	t_{EXcyc}	25	100	ns	
AUDIO_X1、AUDIO_CLK 时钟输入频率 (外部时钟输入时)	f_{EX}	1	40	MHz	
AUDIO_X1、AUDIO_CLK 时钟输入周期时间 (外部时钟输入时)	t_{EXcyc}	25	1000	ns	
USB_X1 时钟输入频率	f_{EX}	48MHz $\pm$ 100ppm			
EXTAL、AUDIO_X1、AUDIO_CLK、USB_X1 时钟输入低电平脉宽	t_{EXL}	0.4	0.6	t_{EXcyc}	
EXTAL、AUDIO_X1、AUDIO_CLK、USB_X1 时钟输入高电平脉宽	t_{EXH}	0.4	0.6	t_{EXcyc}	
EXTAL、AUDIO_X1、AUDIO_CLK、USB_X1 时钟输入上升时间	t_{EXr}	—	4	ns	
EXTAL、AUDIO_X1、AUDIO_CLK、USB_X1 时钟输入下降时间	t_{EXf}	—	4	ns	
CKIO 时钟输入频率	f_{CK}	40.00	66.66	MHz	31.3
CKIO 时钟输入周期时间	t_{CKIcyc}	15	25	ns	
CKIO 时钟输入低电平脉宽	t_{CKIL}	0.4	0.6	t_{CKIcyc}	
CKIO 时钟输入高电平脉宽	t_{CKIH}	0.4	0.6	t_{CKIcyc}	
CKIO 时钟输入上升时间	t_{CKIr}	—	3	ns	
CKIO 时钟输入下降时间	t_{CKIf}	—	3	ns	

项 目	符号	Min.	Max.	单位	参照图
CKIO 时钟输出频率	f_{OP}	40.00	66.66	MHz	31.4
CKIO 时钟输出周期时间	t_{cyc}	15	25	ns	
CKIO 时钟输出低电平脉宽	t_{CKOL}	0.4	0.6	t_{cyc}	
CKIO 时钟输出高电平脉宽	t_{CKOH}	0.4	0.6	t_{cyc}	
CKIO 时钟输出上升时间	t_{CKOr}	—	3	ns	
CKIO 时钟输出下降时间	t_{CKOf}	—	3	ns	
上电振荡稳定时间	t_{OSC1}	10	—	ms	31.5
待机返回振荡稳定时间 1	t_{OSC2}	10	—	ms	31.6
待机返回振荡稳定时间 2	t_{OSC3}	10	—	ms	31.7
RTC 时钟振荡稳定时间	t_{ROSC}	3	—	s	31.8

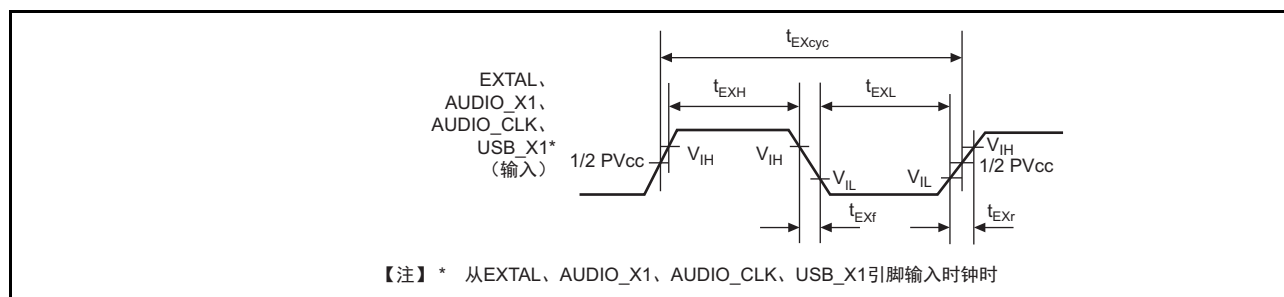


图 31.2 EXTAL、AUDIO_X1、AUDIO_CLK 及 USB_X1 时钟输入时序

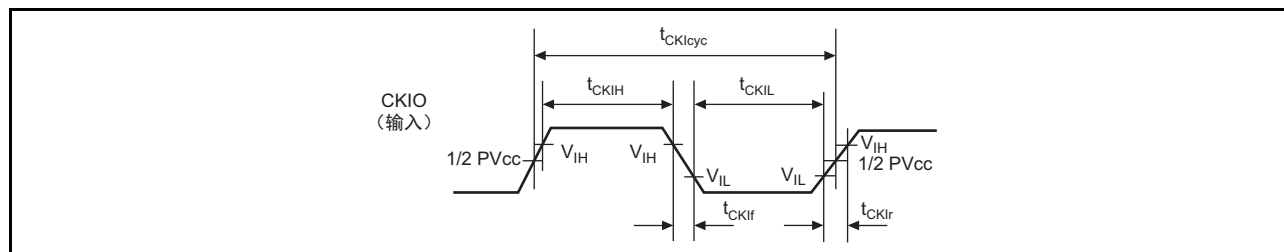


图 31.3 CKIO 时钟输入时序

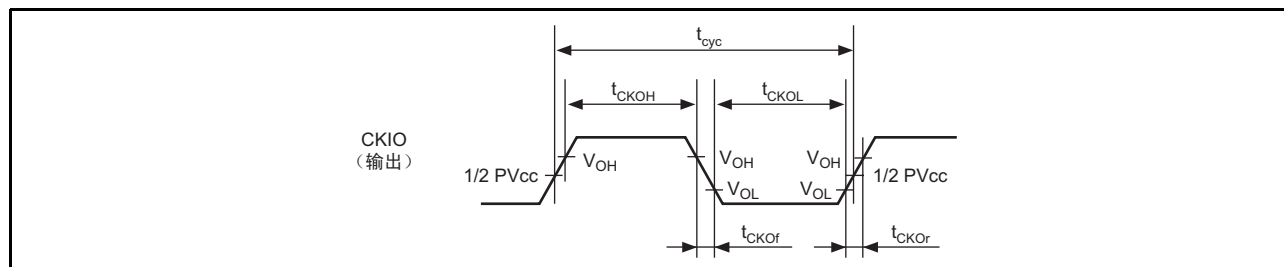


图 31.4 CKIO 时钟输出时序

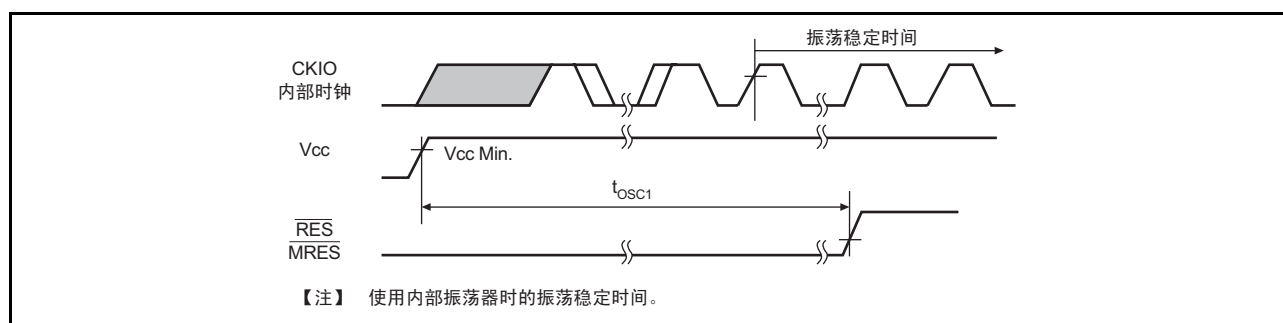


图 31.5 上电振荡稳定时间

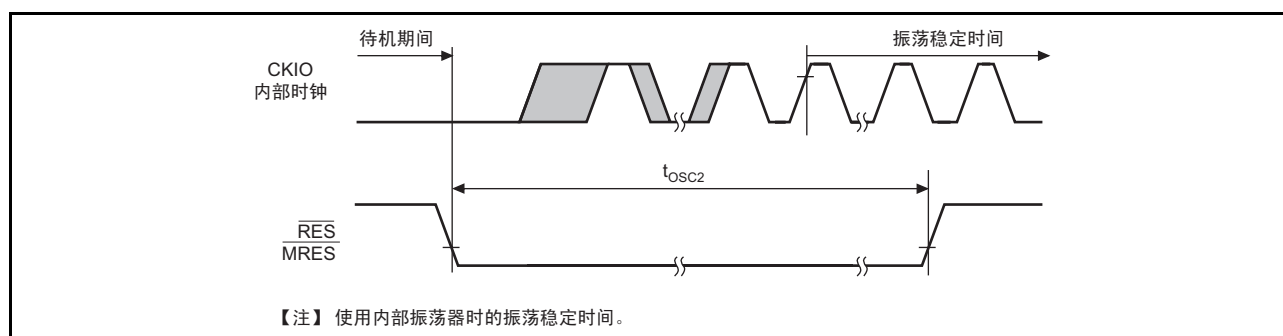


图 31.6 待机返回时的振荡稳定时间（通过复位返回）

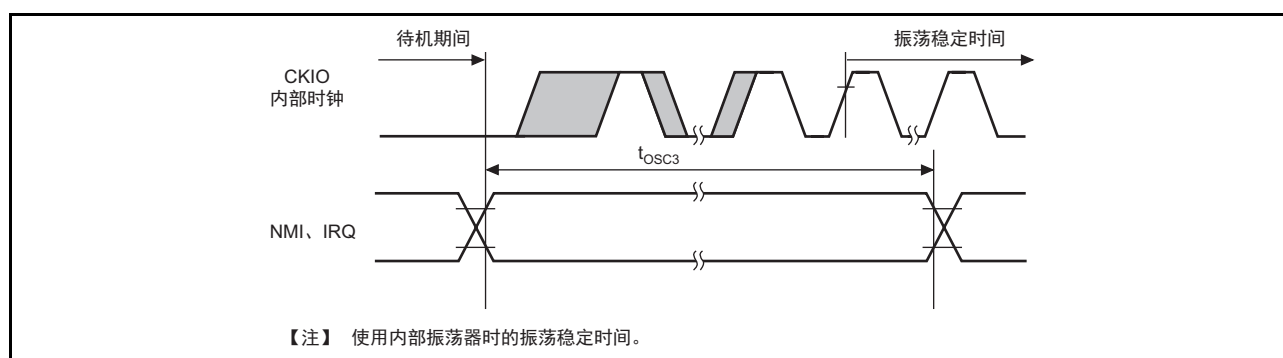


图 31.7 待机返回时的振荡稳定时间（通过 NMI、IRQ 返回）

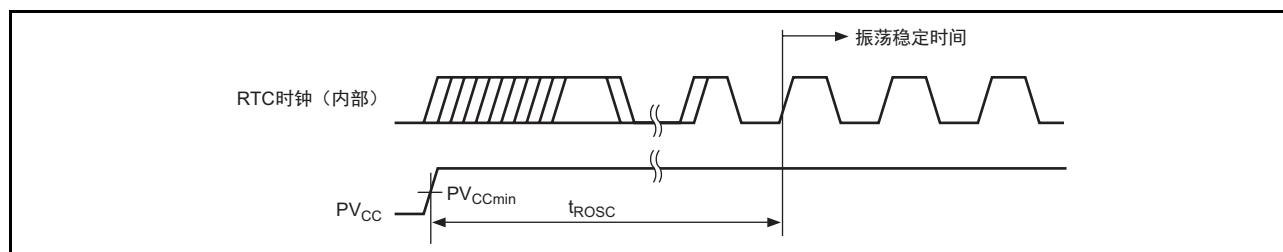


图 31.8 RTC 时钟振荡稳定时间

31.4.2 控制信号时序

表 31.7 控制信号时序

条件: $V_{CC}=PLL V_{CC}=USBDV_{CC}=1.1 \sim 1.3V$ 、 $PV_{CC}=USBDPV_{CC}=3.0 \sim 3.6V$ 、 $AV_{CC}=3.0 \sim 3.6V$ 、
 $USBAV_{CC}=1.1 \sim 1.3V$ 、 $USBAPV_{CC}=3.0 \sim 3.6V$ 、
 $V_{SS}=PLL V_{SS}=PV_{SS}=AV_{SS}=USBDV_{SS}=USBAV_{SS}=USBDPV_{SS}=USBAPV_{SS}=0V$ 、 $T_a=-20 \sim 85^{\circ}C$

项 目	符号	B ϕ =66.66MHz		单位	参照图
		Min.	Max.		
$\overline{RES}$ 脉宽	t_{RESW}	20*1	—	tcyc	31.9
$\overline{MRES}$ 脉宽	t_{MRESW}	20*2	—	tcyc	
NMI 脉宽	t_{NMIW}	20*3	—	tcyc	31.10
IRQ 脉宽	t_{IRQW}	20*3	—	tcyc	
PINT 脉宽	t_{PINTW}	20	—	tcyc	
$\overline{IRQOUT}/\overline{REFOUT}$ 输出延迟时间	t_{IRQOD}	—	100	ns	31.11
$\overline{BREQ}$ 准备时间	t_{BREQS}	$1/2t_{cyc}+7$	—	ns	31.12
$\overline{BREQ}$ 保持时间	t_{BREQH}	$1/2t_{cyc}+2$	—	ns	
BACK 延迟时间	t_{BACKD}	—	$1/2t_{cyc}+13$	ns	
总线缓冲器断开时间 1	t_{BOFF1}	—	15	ns	
总线缓冲器断开时间 2	t_{BOFF2}	—	15	ns	
总线缓冲器保持时间 1	t_{BON1}	—	15	ns	
总线缓冲器保持时间 2	t_{BON2}	—	15	ns	31.12
总线缓冲器断开时的 BACK 准备时间	t_{BACKS}	0	—	ns	

【注】 *1 待机模式时或时钟倍频率变化时，为 $t_{RESW}=t_{OSC2}$ （10ms）。

*2 待机模式时，为 $t_{MRESW}=t_{OSC2}$ （10ms）。

*3 待机模式时，为 $t_{NMIW}/t_{IRQW}=t_{OSC3}$ （10ms）。

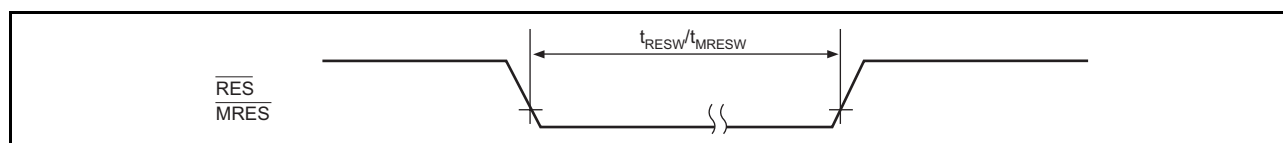


图 31.9 复位输入时序

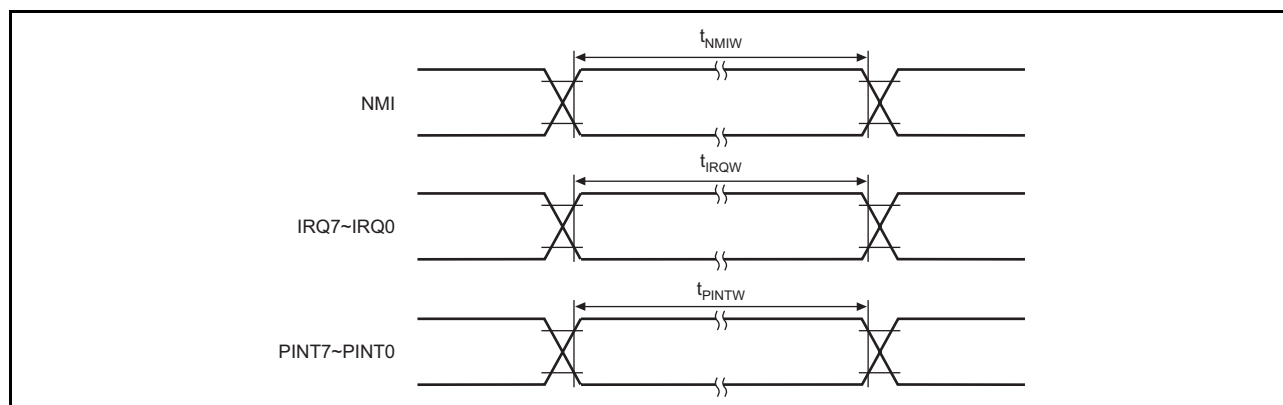


图 31.10 中断信号输入时序

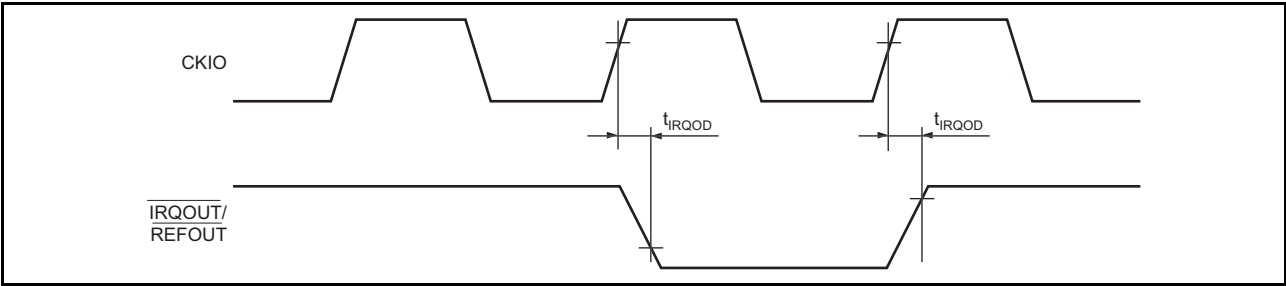


图 31.11 中断信号输出时序

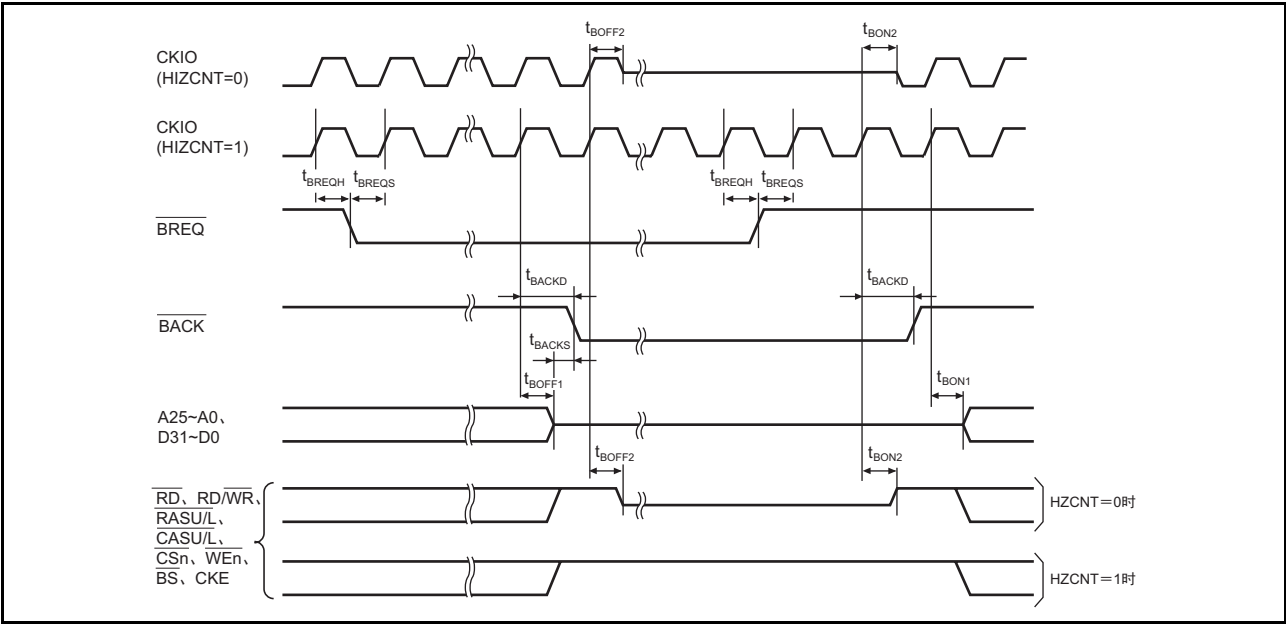


图 31.12 总线权释放时序

31.4.3 总线时序

表 31.8 总线时序

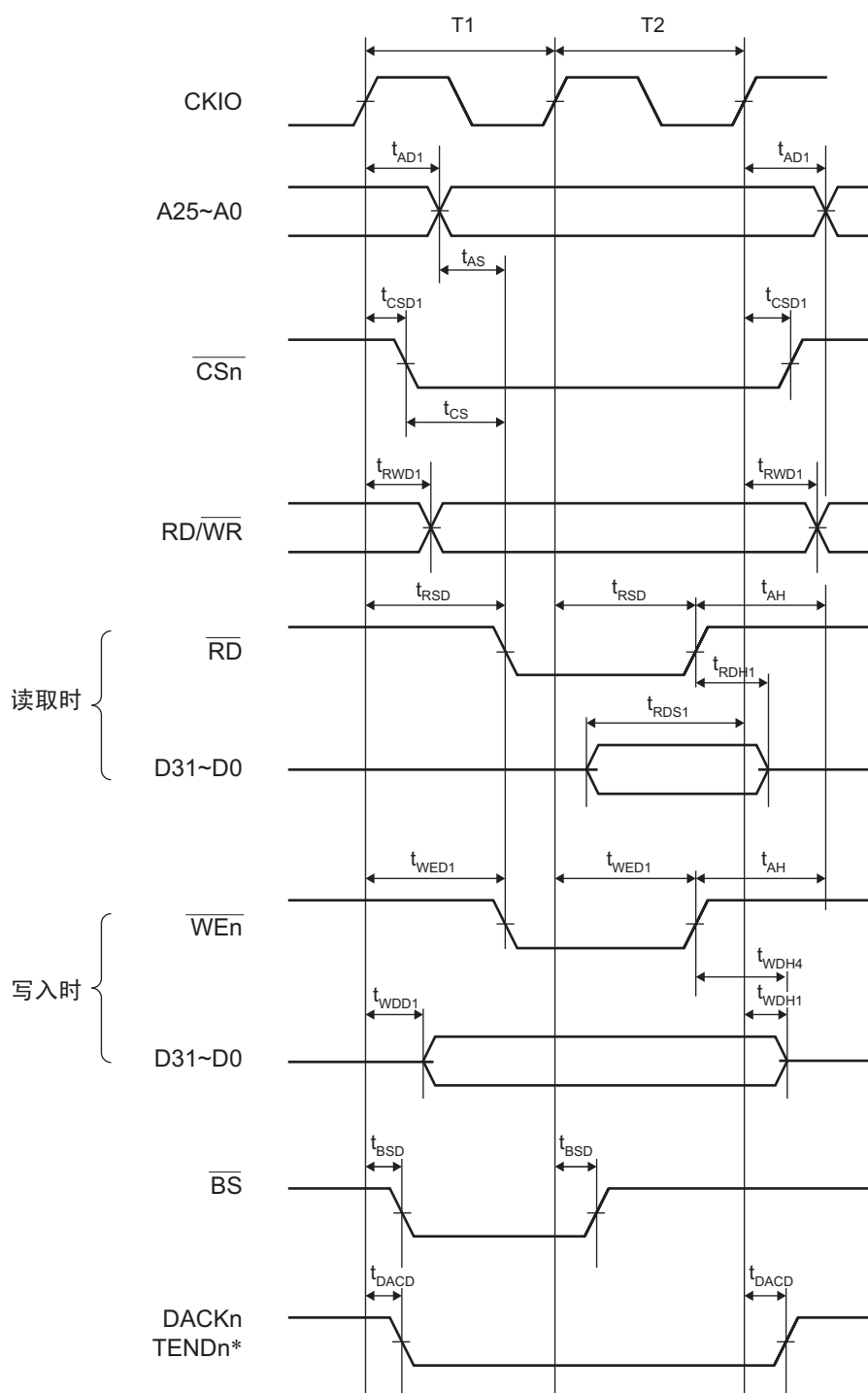
条件: $V_{CC}=PLL V_{CC}=USBDV_{CC}=1.1 \sim 1.3V$ 、 $PV_{CC}=USBDPV_{CC}=3.0 \sim 3.6V$ 、 $AV_{CC}=3.0 \sim 3.6V$ 、
 $USBAV_{CC}=1.1 \sim 1.3V$ 、 $USBAPV_{CC}=3.0 \sim 3.6V$ 、
 $V_{SS}=PLL V_{SS}=PV_{SS}=AV_{SS}=USBDV_{SS}=USBAV_{SS}=USBDPV_{SS}=USBAPV_{SS}=0V$ 、 $T_a=-20 \sim 85^{\circ}C$

项 目	符号	Bφ=66.66MHz*1,*2		单位	参照图
		Min.	Max.		
地址延迟时间 1	t_{AD1}	1	13	ns	31.13 ~ 31.38、 31.40 ~ 31.44
地址延迟时间 2	t_{AD2}	$1/2t_{cyc}$	$1/2t_{cyc}+13$	ns	31.21
地址延迟时间 3	t_{AD3}	$1/2t_{cyc}$	$1/2t_{cyc}+13$	ns	31.39、31.40
地址准备时间	t_{AS}	0	—	ns	31.13 ~ 31.16、31.21
芯片允许准备时间	t_{CS}	0	—	ns	31.13 ~ 31.16、31.21
地址保持时间	t_{AH}	0	—	ns	31.13 ~ 31.16
$\overline{BS}$ 延迟时间	t_{BSD}	—	13	ns	31.13 ~ 31.35、31.39、 31.41 ~ 31.44
$\overline{CS}$ 延迟时间 1	t_{CSD1}	1	13	ns	31.13 ~ 31.38、 31.41 ~ 31.44
$\overline{CS}$ 延迟时间 2	t_{CSD2}	$1/2t_{cyc}$	$1/2t_{cyc}+13$	ns	31.39、31.40
读取 / 写入延迟时间 1	t_{RWD1}	1	13	ns	31.13 ~ 31.38、 31.41 ~ 31.44
读取 / 写入延迟时间 2	t_{RWD2}	$1/2t_{cyc}$	$1/2t_{cyc}+13$	ns	31.39、31.40
读取选通延迟时间	t_{RSD}	$1/2t_{cyc}$	$1/2t_{cyc}+13$	ns	31.13 ~ 31.17、 31.19 ~ 31.21、31.41、 31.42
读取数据准备时间 1	t_{RDS1}	$1/2t_{cyc}+13$	—	ns	31.13 ~ 31.17、31.19、 31.20、31.41 ~ 31.44
读取数据准备时间 2	t_{RDS2}	8	—	ns	31.18、31.22 ~ 31.25、 31.30 ~ 31.32
读取数据准备时间 3	t_{RDS3}	$1/2t_{cyc}+13$	—	ns	31.21
读取数据准备时间 4	t_{RDS4}	$1/2t_{cyc}+13$	—	ns	31.39
读取数据保持时间 1	t_{RDH1}	0	—	ns	31.13 ~ 31.17、31.19、 31.20、31.41 ~ 31.44
读取数据保持时间 2	t_{RDH2}	2	—	ns	31.18、31.22 ~ 31.25、 31.30 ~ 31.32
读取数据保持时间 3	t_{RDH3}	0	—	ns	31.21
读取数据保持时间 4	t_{RDH4}	$1/2t_{cyc}+6$	—	ns	31.39
写入允许延迟时间 1	t_{WED1}	$1/2t_{cyc}$	$1/2t_{cyc}+13$	ns	31.13 ~ 31.17、31.18、 31.41、31.42
写入允许延迟时间 2	t_{WED2}	—	13	ns	31.20

项 目	符号	B ϕ =66.66MHz*1,*2		单位	参照图
		Min.	Max.		
写入数据延迟时间 1	t _{WDD1}	—	13	ns	31.13 ~ 31.20、 31.41 ~ 31.44
写入数据延迟时间 2	t _{WDD2}	—	13	ns	31.26 ~ 31.29、 31.33 ~ 31.35
写入数据延迟时间 3	t _{WDD3}	—	1/2t _{cyc} +13	ns	31.39
写入数据保持时间 1	t _{WDH1}	1	—	ns	31.13 ~ 31.20、 31.41 ~ 31.44
写入数据保持时间 2	t _{WDH2}	1	—	ns	31.26 ~ 31.29、 31.33 ~ 31.35
写入数据保持时间 3	t _{WDH3}	1/2t _{cyc}	—	ns	31.39
写入数据保持时间 4	t _{WDH4}	0	—	ns	31.13、31.17、31.41、31.43
WAIT 准备时间	t _{WTS}	1/2t _{cyc} +5.5	—	ns	31.14 ~ 31.21、31.42、 31.44
WAIT 保持时间	t _{WTH}	1/2t _{cyc} +4.5	—	ns	31.14 ~ 31.21、31.42、 31.44
IOIS16 准备时间	t _{IO16S}	1/2t _{cyc} +8	—	ns	31.44
IOIS16 保持时间	t _{IO16H}	1/2t _{cyc} +5	—	ns	31.44
RAS 延迟时间 1	t _{RASD1}	1	13	ns	31.22 ~ 31.38
RAS 延迟时间 2	t _{RASD2}	1/2t _{cyc}	1/2t _{cyc} +13	ns	31.39、31.40
CAS 延迟时间 1	t _{CASD1}	1	13	ns	31.22 ~ 31.38
CAS 延迟时间 2	t _{CASD2}	1/2t _{cyc}	1/2t _{cyc} +13	ns	31.39、31.40
DQM 延迟时间 1	t _{DQMD1}	1	13	ns	31.22 ~ 31.35
DQM 延迟时间 2	t _{DQMD2}	1/2t _{cyc}	1/2t _{cyc} +13	ns	31.39、31.40
CKE 延迟时间 1	t _{CKED1}	1	13	ns	31.37
CKE 延迟时间 2	t _{CKED2}	1/2t _{cyc}	1/2t _{cyc} +13	ns	31.40
AH 延迟时间	t _{AHD}	1/2t _{cyc}	1/2t _{cyc} +13	ns	31.17
多路复用地址延迟时间	t _{MAD}	—	13	ns	31.17
多路复用地址保持时间	t _{MAH}	1	—	ns	31.17
AH 对应的地址准备时间	t _{AVVH}	1/2t _{cyc} -2	—	ns	31.17
DACK、TEND 延迟时间	t _{DACD}	参照 DMAC 时序		ns	31.13 ~ 31.35、31.39、 31.41 ~ 31.44
FRAME 延迟时间	t _{FMD}	0	13	ns	31.18
ICIORD 延迟时间	t _{ICRSD}	—	1/2t _{cyc} +13	ns	31.43、31.44
ICIOWR 延迟时间	t _{ICWSD}	—	1/2t _{cyc} +13	ns	31.43、31.44

【注】 *1 研究 B ϕ （外部总线时钟）的 f_{max} 时，也要参照所使用的系统结构对应的等待数。

*2 延迟时间与准备、保持时间的 1/2t_{cyc} 表示从时钟上升开始的 1/2 周期，即表示以时钟的下降为基准。



【注】* DACKn、TENDn为指定低电平有效时的波形

图 31.13 普通空间基本总线周期（无等待）

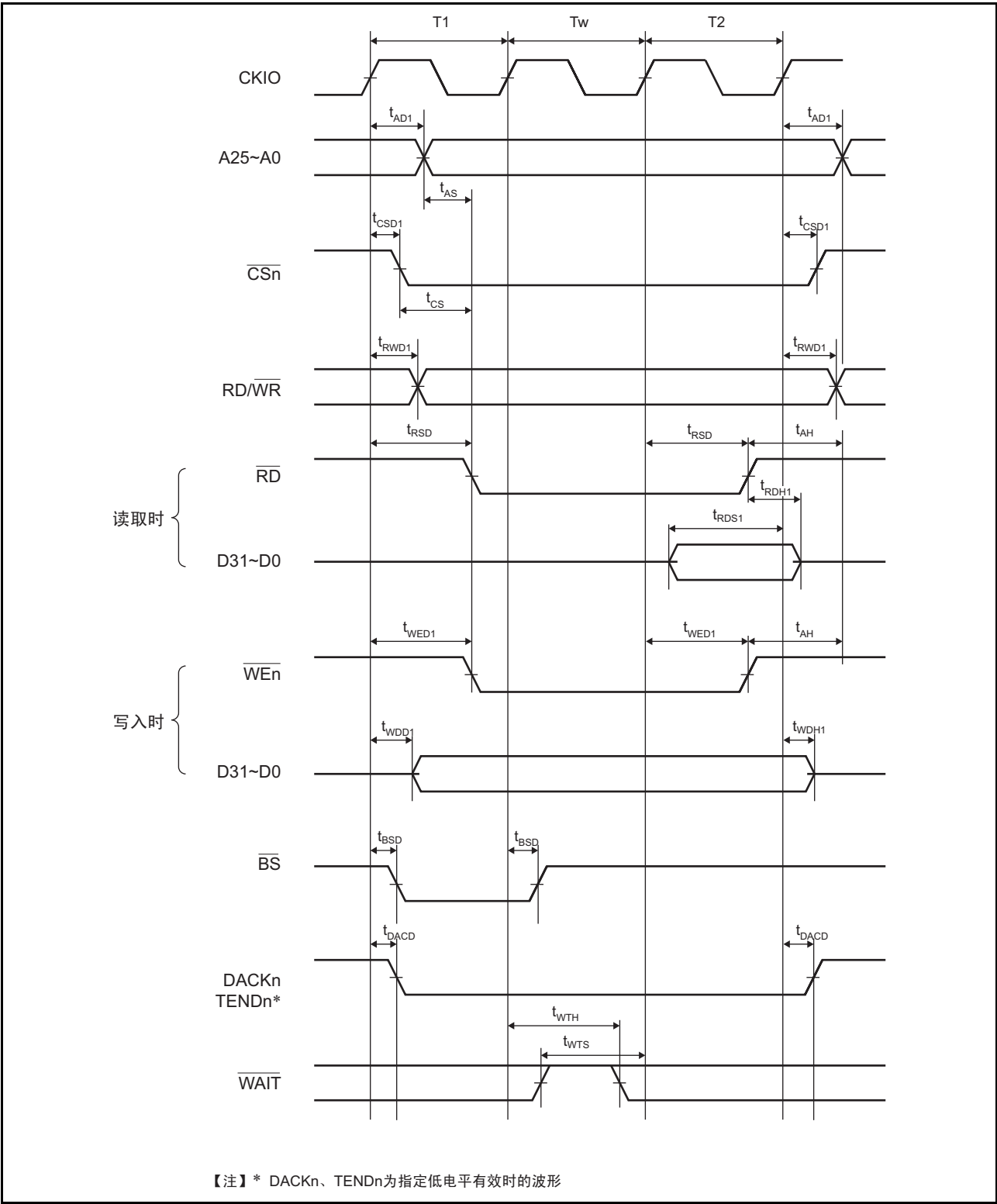


图 31.14 普通空间基本总线周期（1 个软件等待周期）

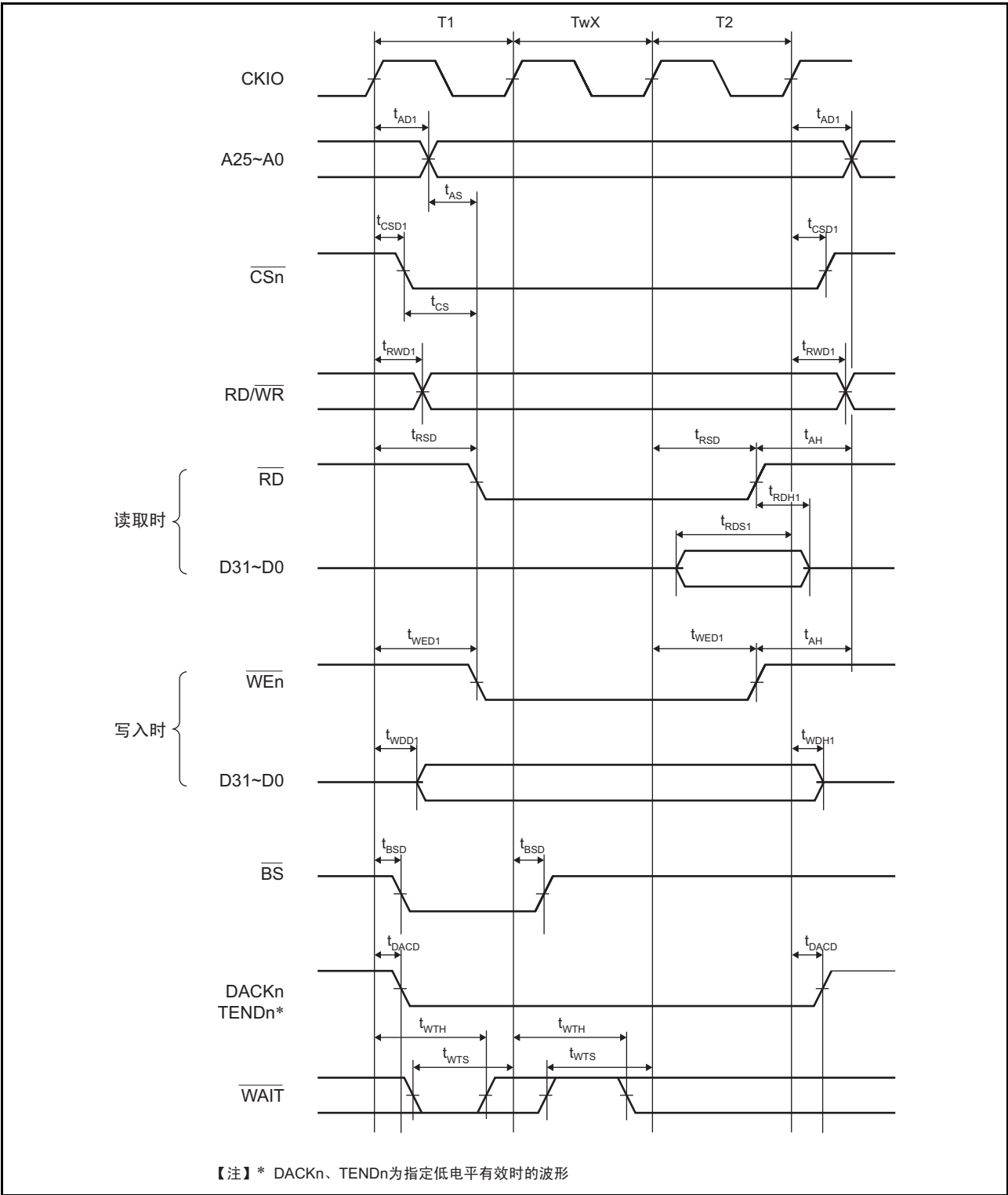


图 31.15 普通空间基本总线周期（插入 1 个外部等待周期）

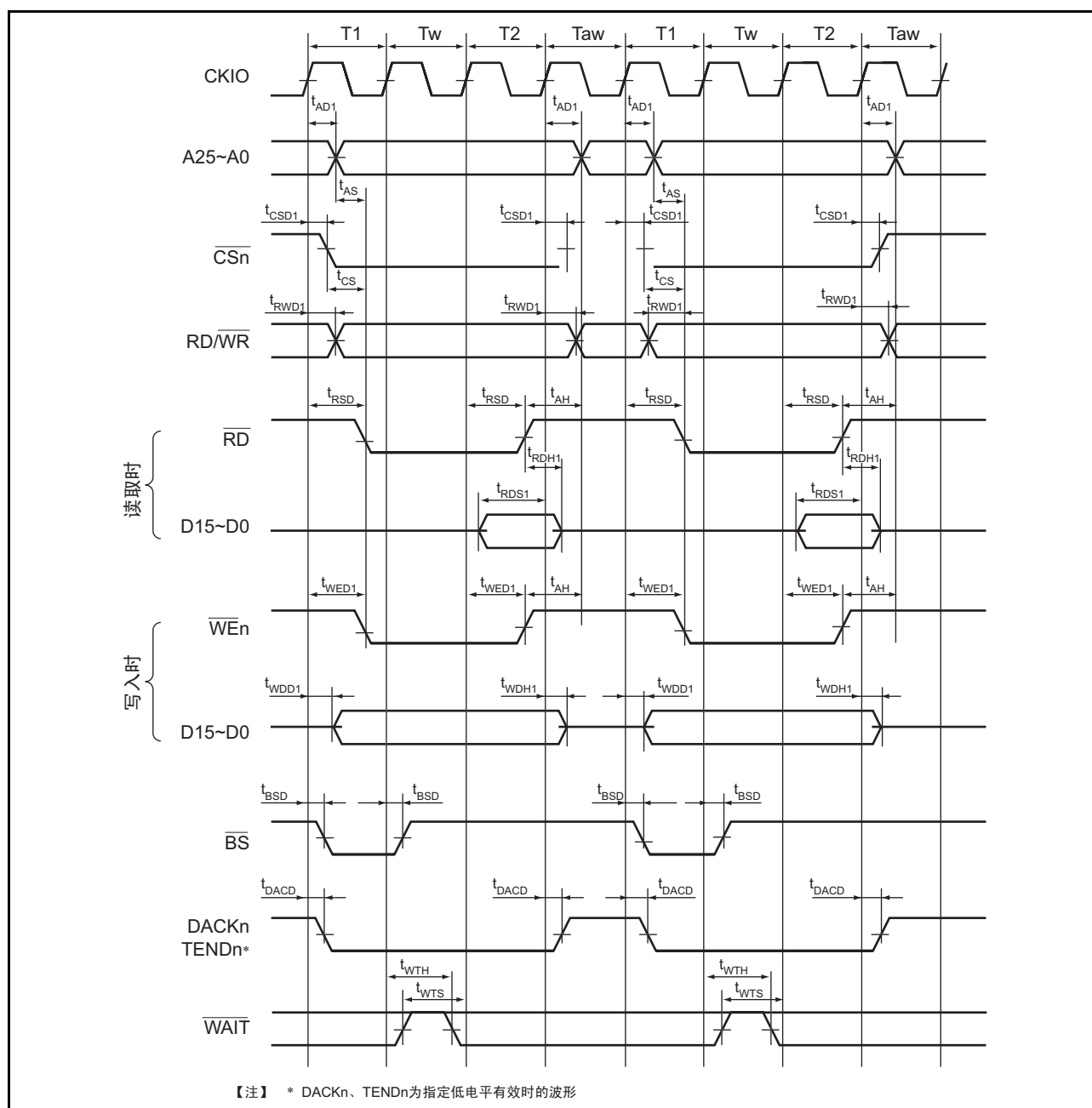


图 31.16 普通空间基本总线周期
(1 个软件等待周期、外部等待有效 (WM 位 = 0)、无空闲周期)

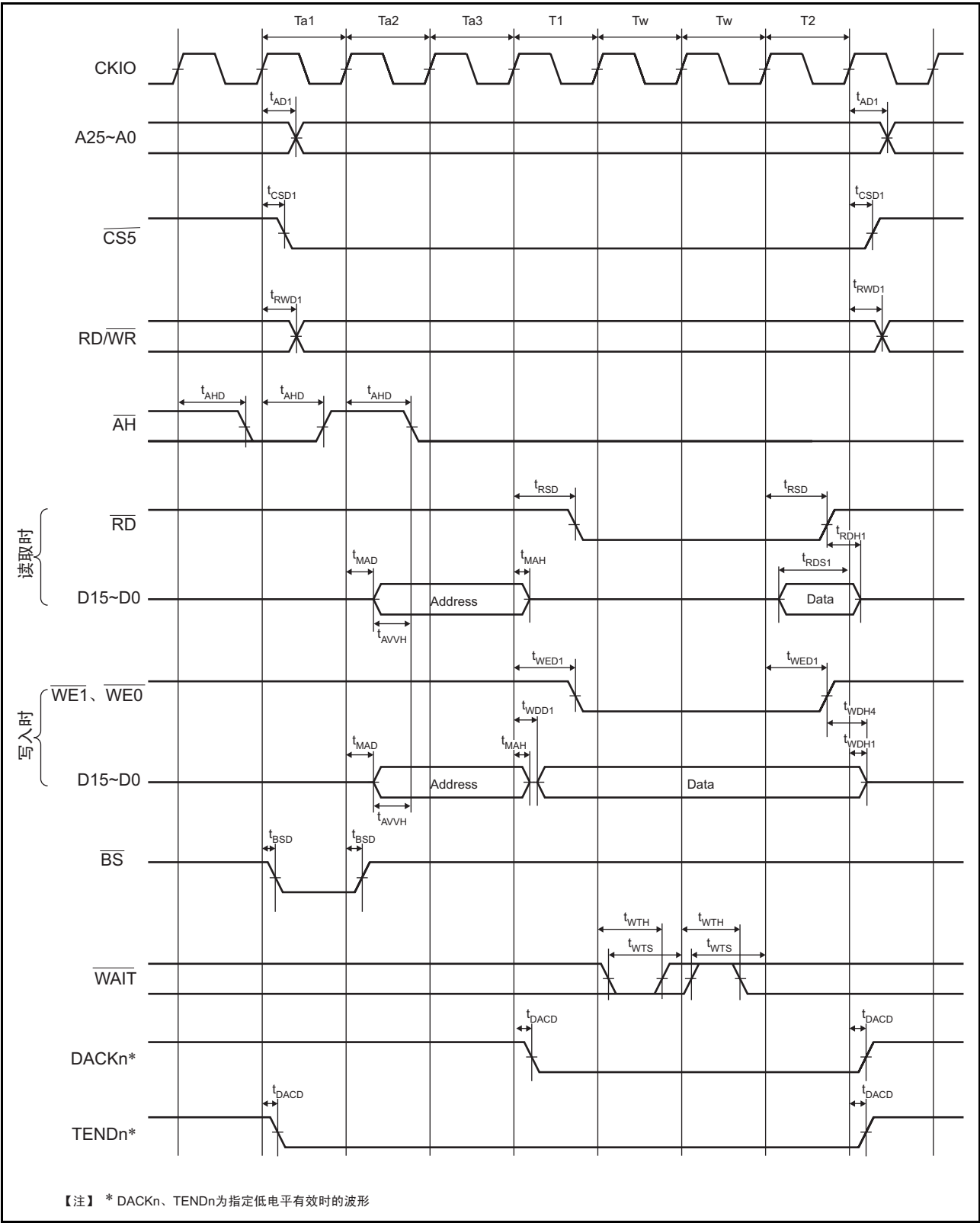


图 31.17 MPX-I/O 接口总线周期
(3 个地址周期、1 个软件等待周期、插入 1 个外部等待)

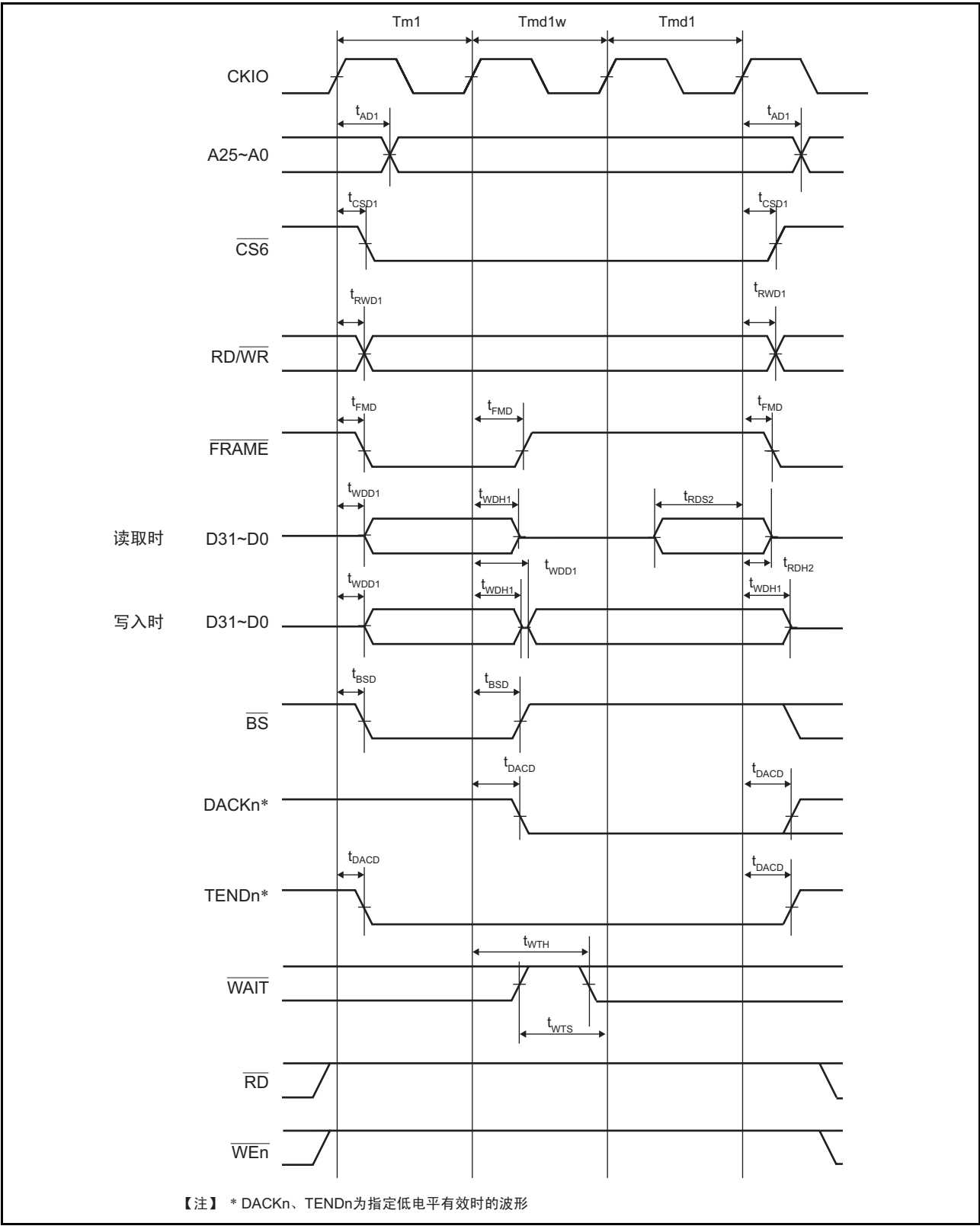


图 31.18 突发 MPX-I/O 接口总线周期单次读取 / 写入
(1 个地址周期、1 个软件等待周期)

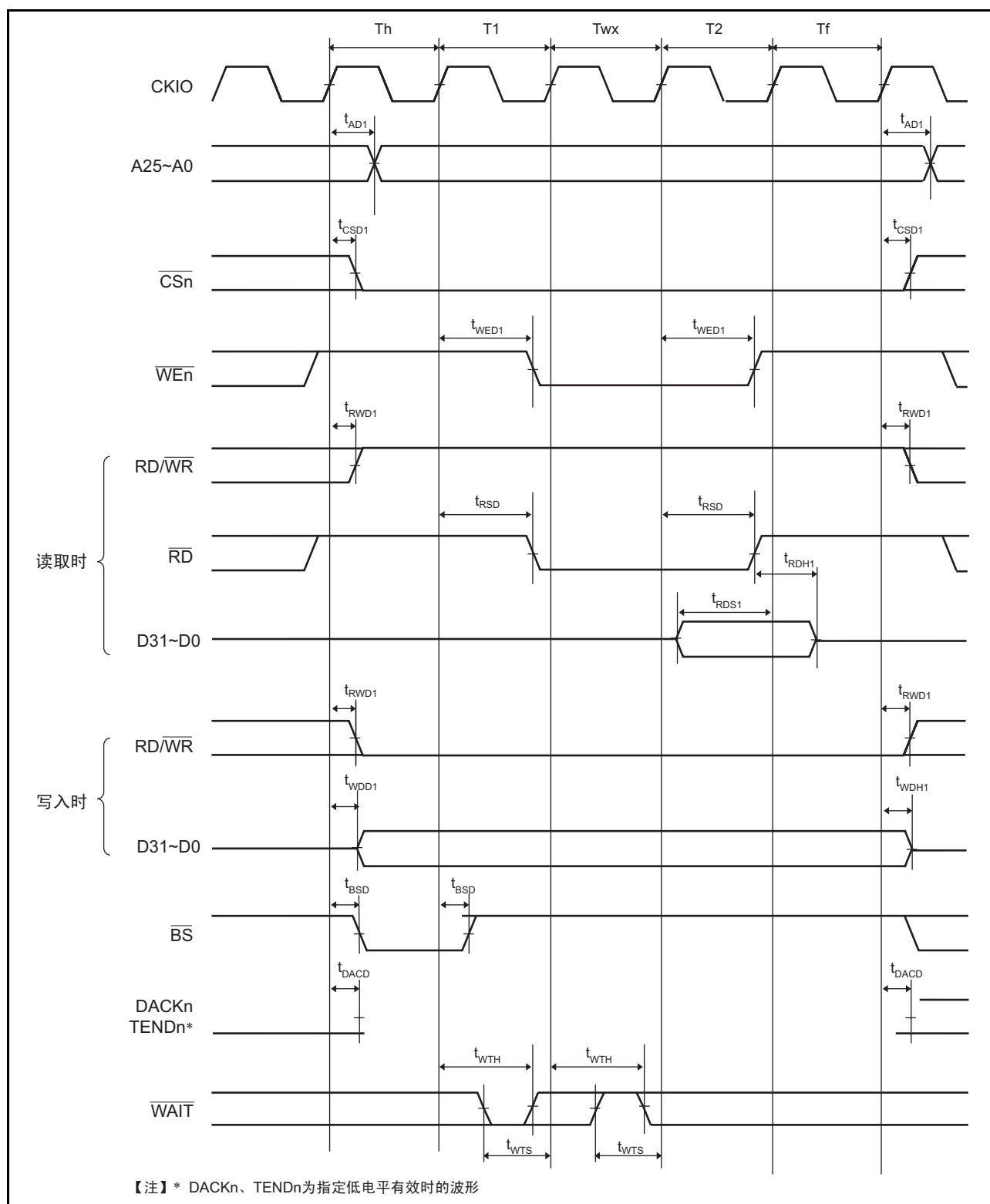


图 31.19 带字节选择的 SRAM 总线周期
(SW=1 个周期、HW=1 个周期、插入 1 个异步外部等待、BAS=0 (写入周期 UB/LB 控制))

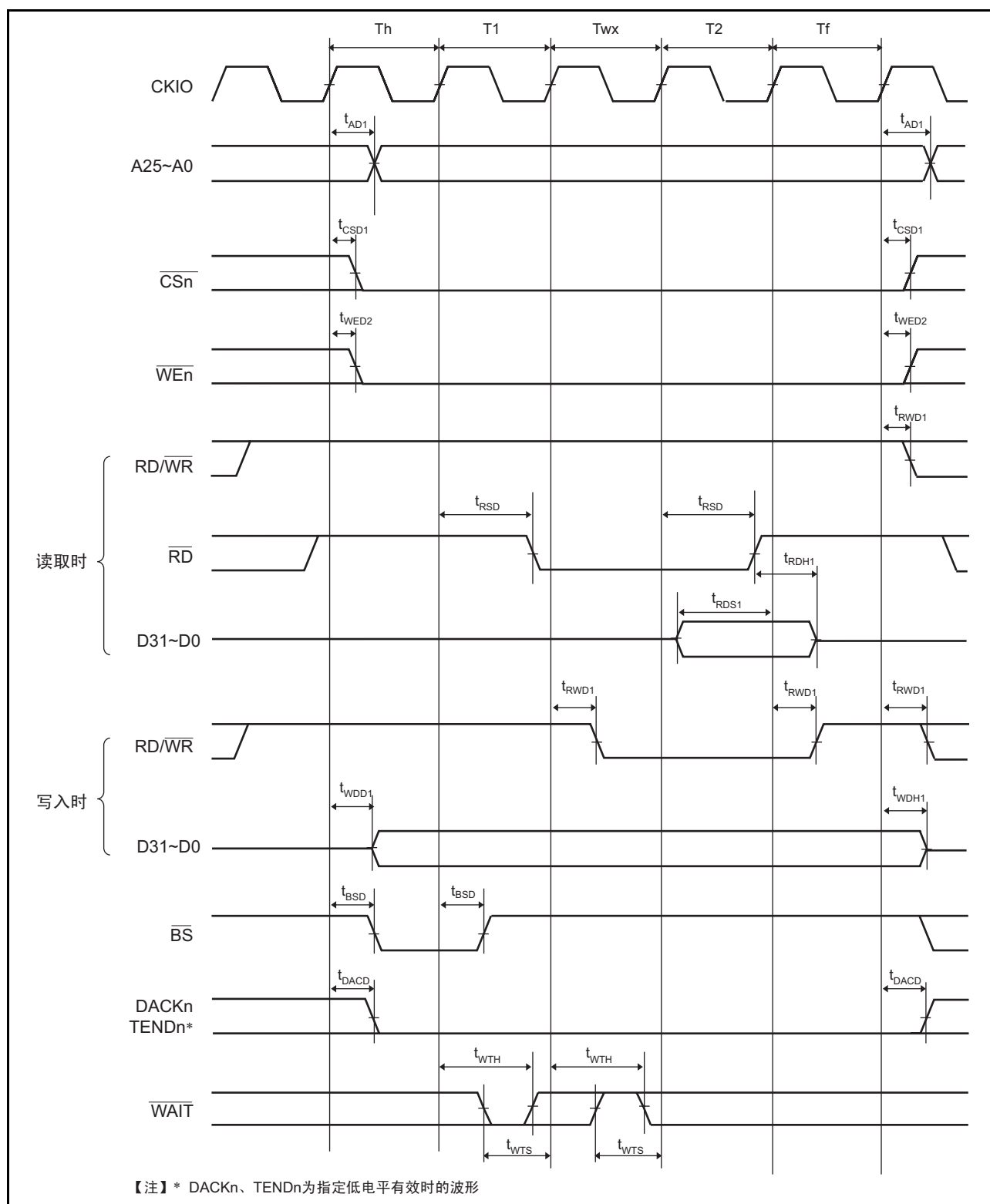


图 31.20 带字节选择的 SRAM 总线周期
(SW=1 个周期、HW=1 个周期、插入 1 个异步外部等待、BAS=1 (写入周期 WE 控制))

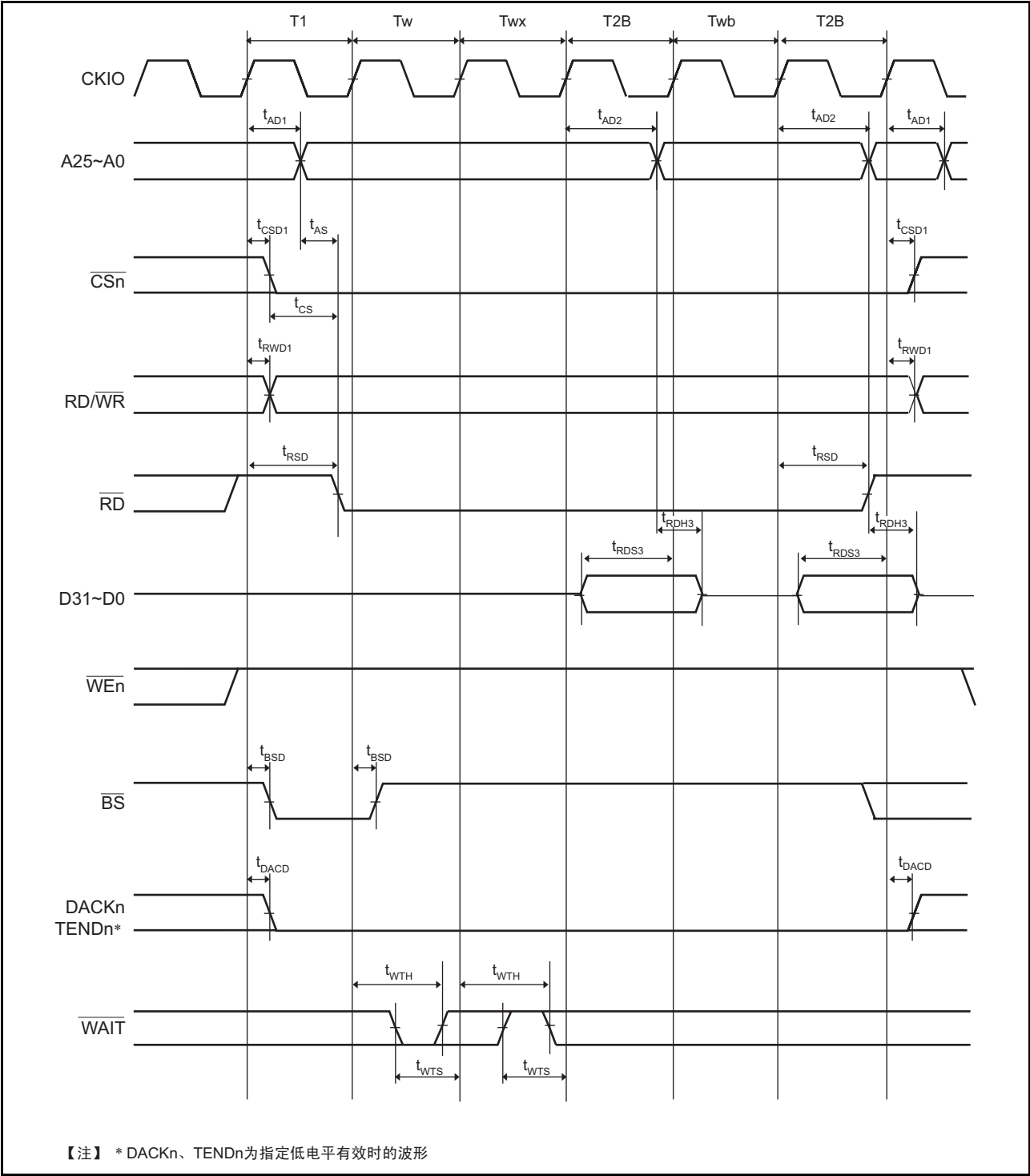


图 31.21 突发 ROM 读取周期
(1 个软件等待周期、插入 1 个异步外部等待、1 突发等待、2 突发)

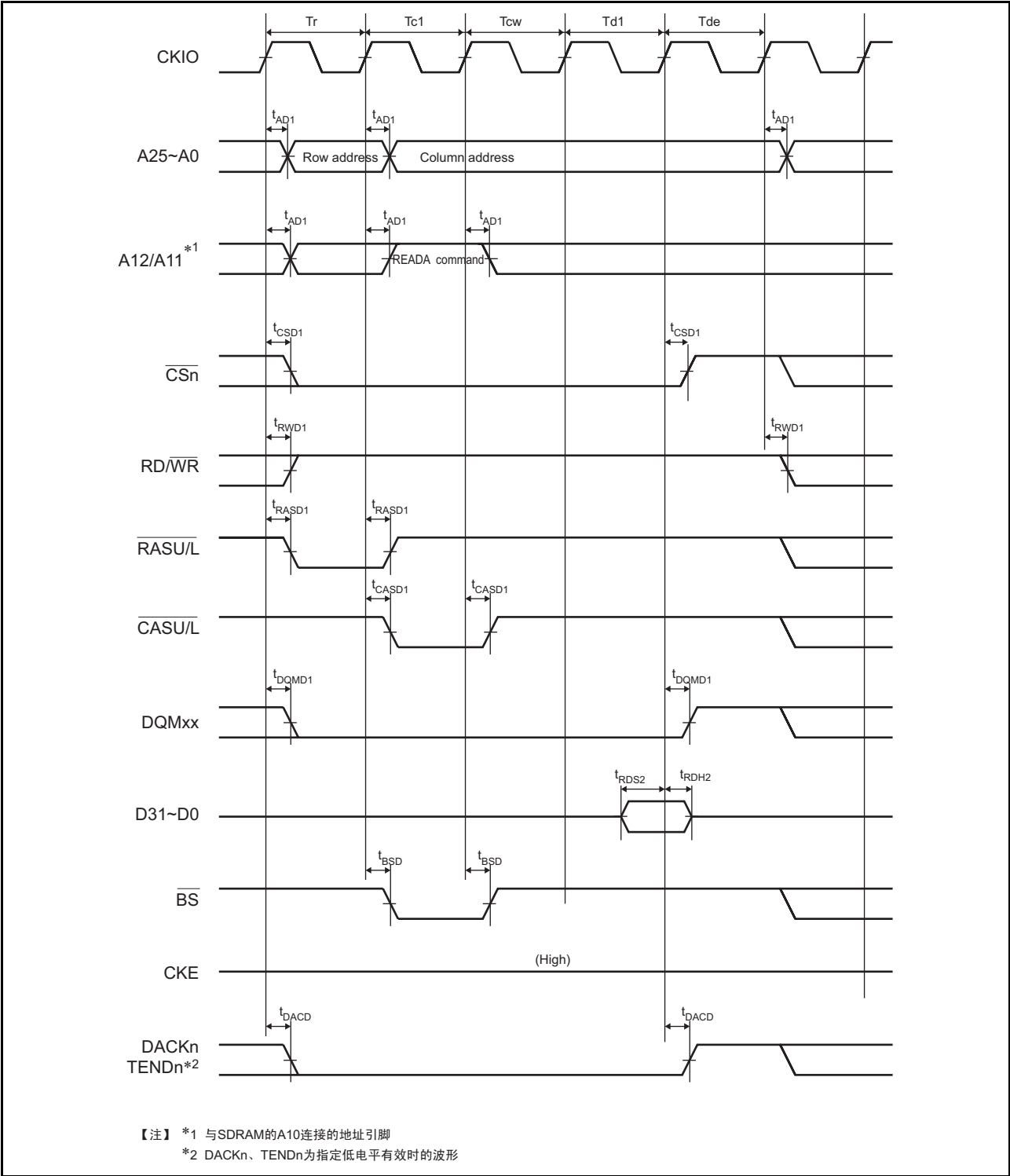


图 31.22 同步 DRAM 单次读取总线周期
(有自动预充电、2 个 CAS 等待时间、WTRCD=0 个周期、WTRP=0 个周期)

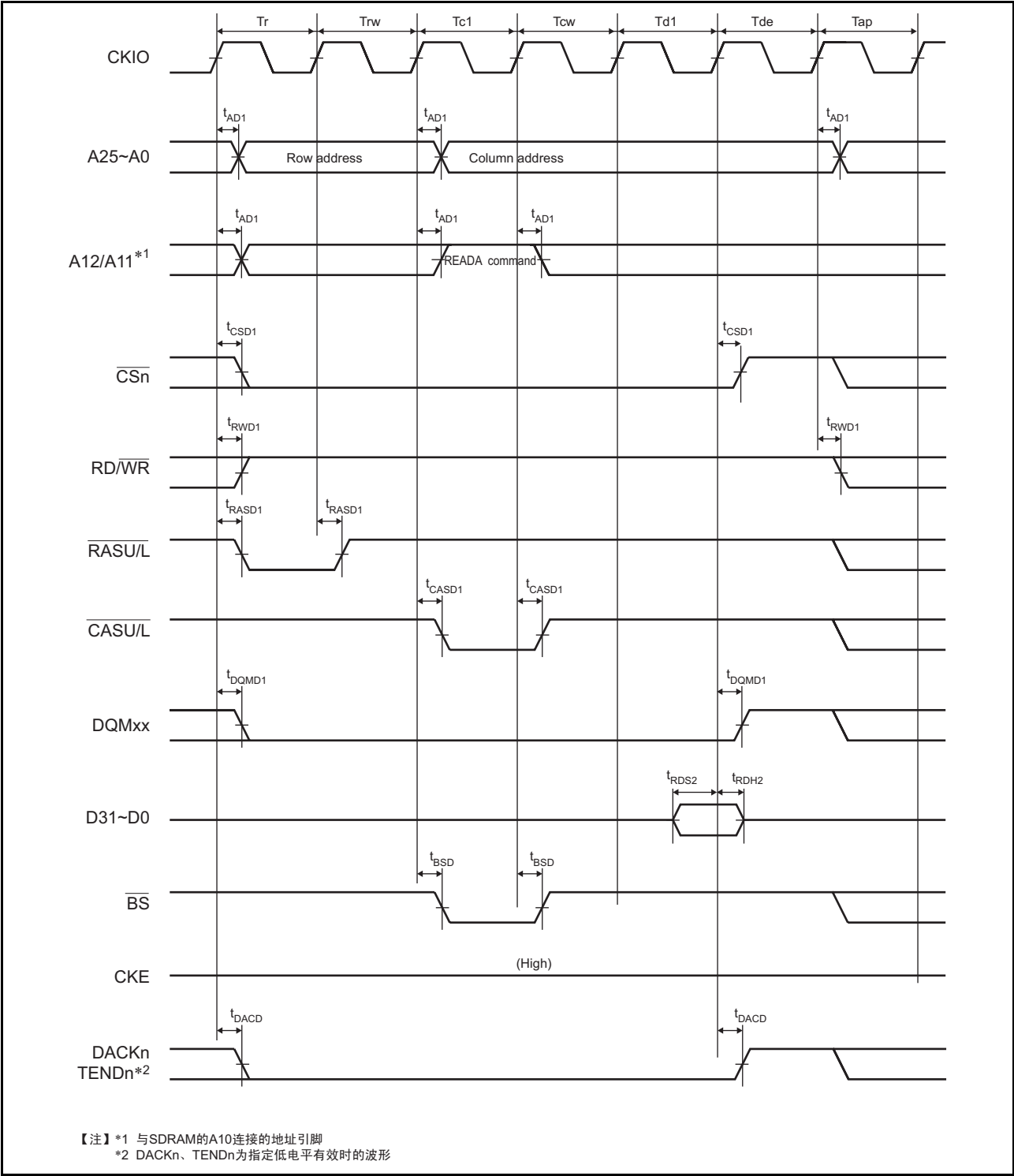


图 31.23 同步 DRAM 单次读取总线周期
(有自动预充电、2 个 CAS 等待时间、WTRCD=1 个周期、WTRP=1 个周期)

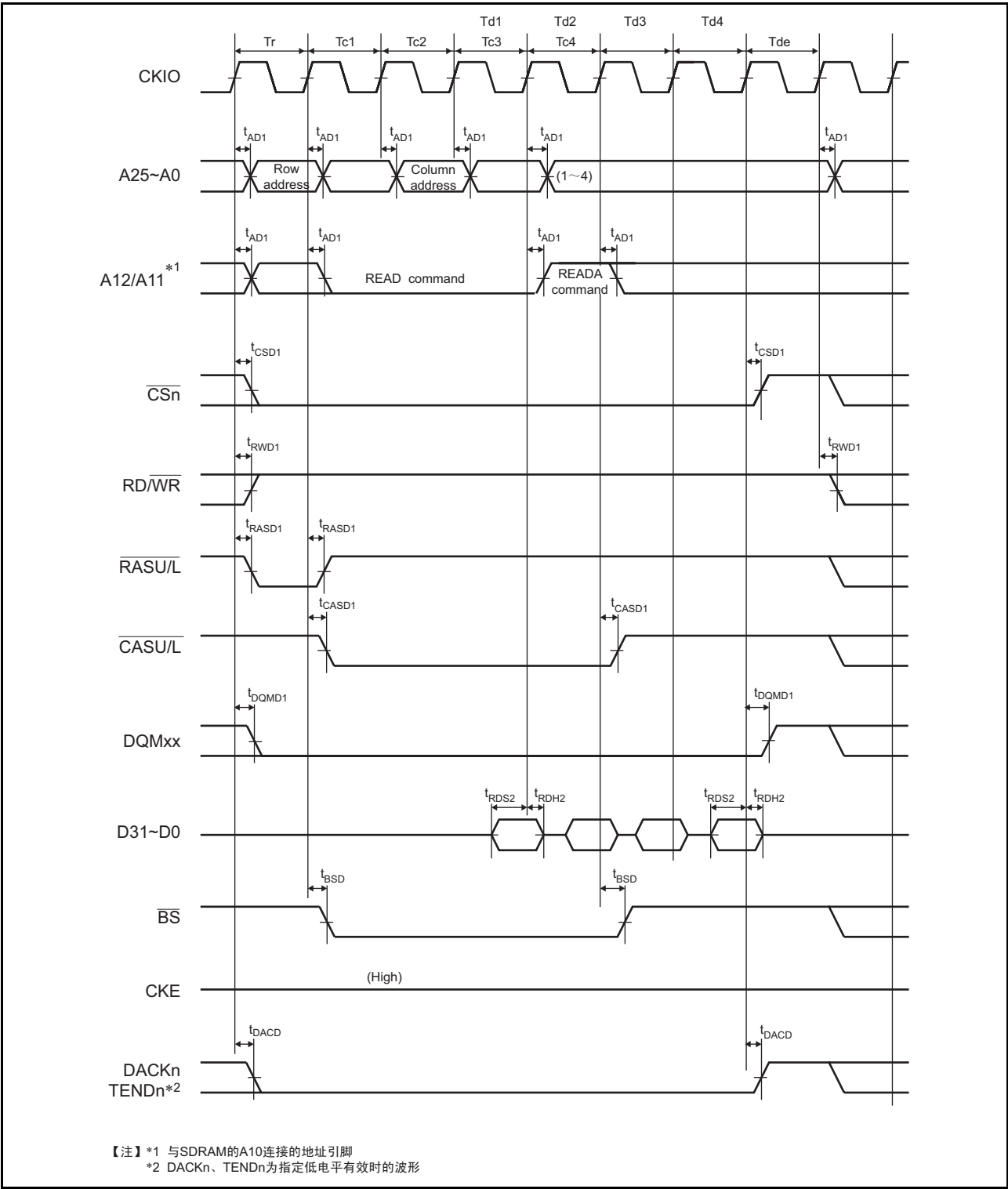


图 31.24 同步 DRAM 突发读取总线周期（读取 4 个周期）
（有自动预充电、2 个 CAS 等待时间、WTRCD=0 个周期、WTRP=1 个周期）

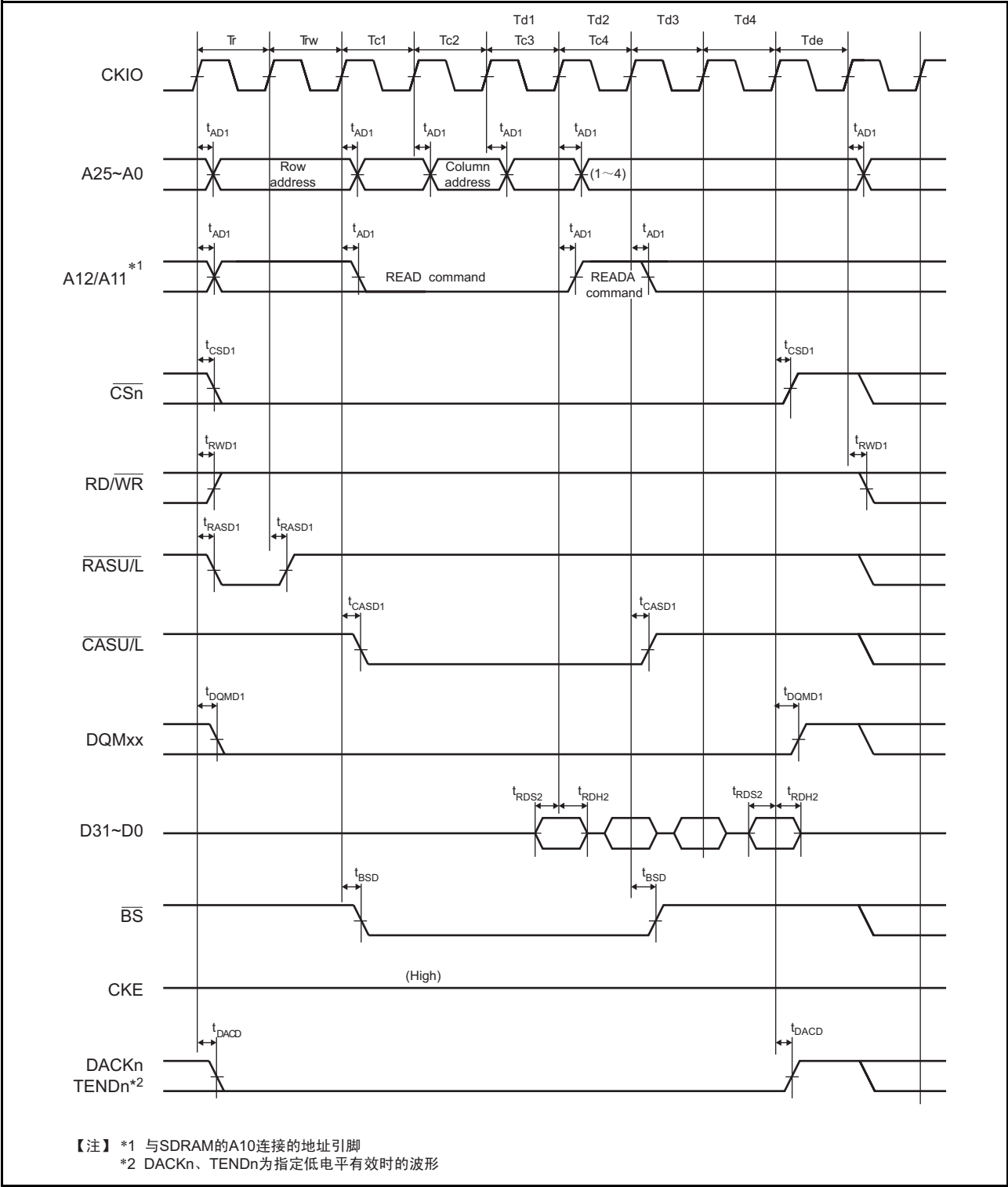


图 31.25 同步 DRAM 突发读取总线周期（读取 4 个周期）
（有自动预充电、2 个 CAS 等待时间、WTRCD=1 个周期、WTRP=0 个周期）

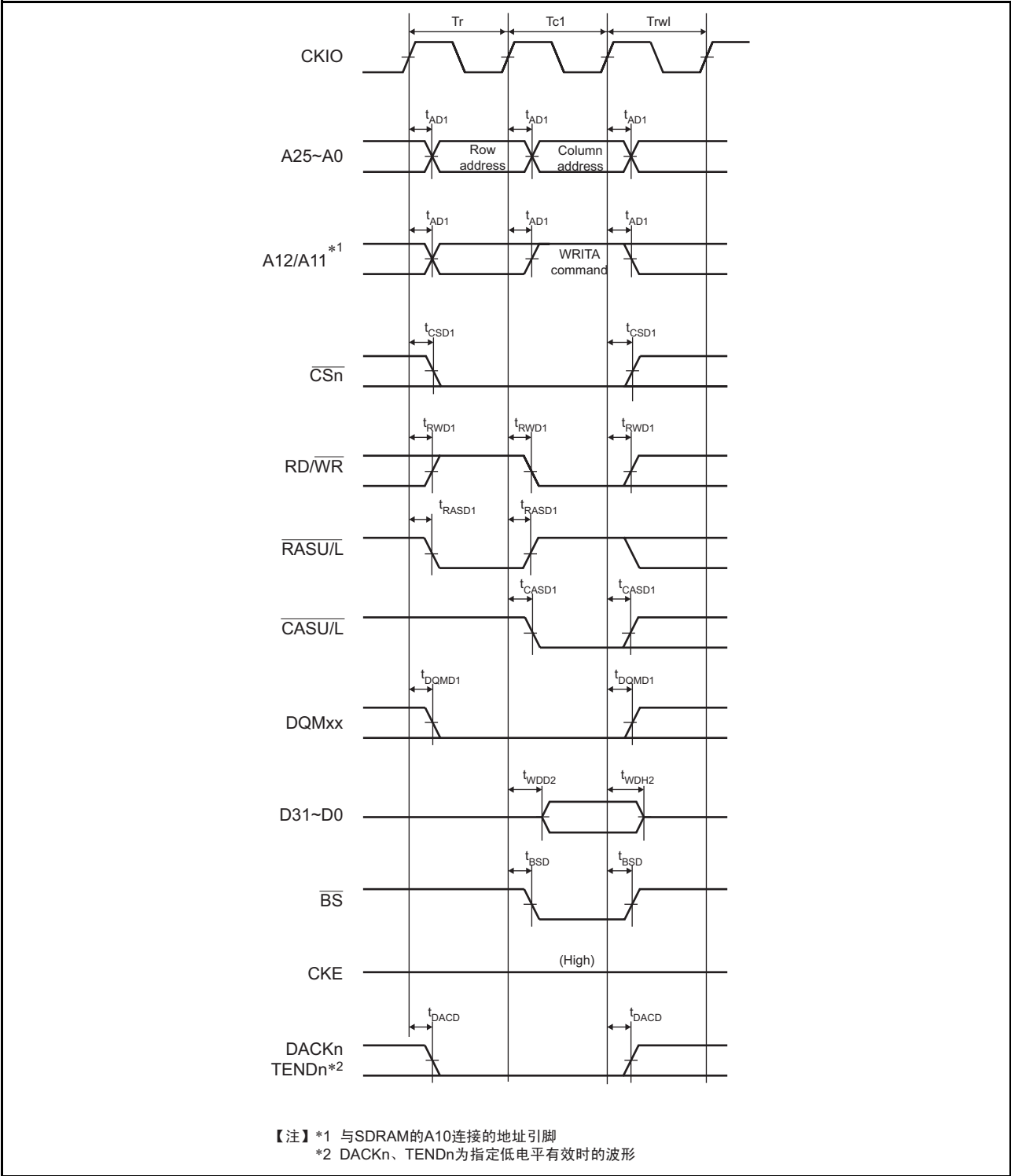
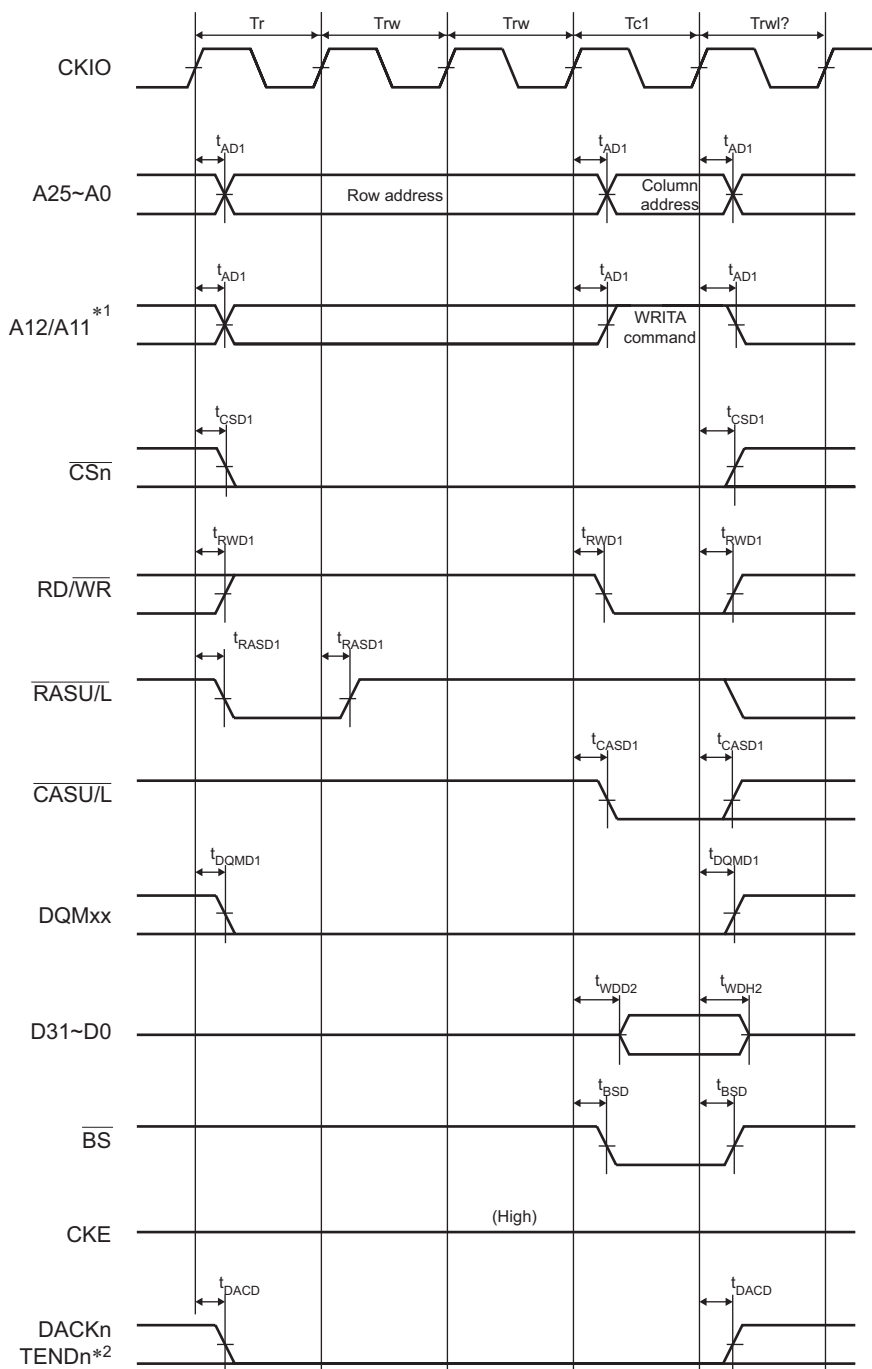


图 31.26 同步 DRAM 单次写入总线周期
(有自动预充电、TRWL=1 个周期)



【注】*1 与 SDRAM 的 A10 连接的地址引脚
 *2 DACKn、TENDn 为指定低电平有效时的波形

图 31.27 同步 DRAM 单次写入总线周期
 (有自动预充电、WTRCD=2 个周期、TRWL=1 个周期)

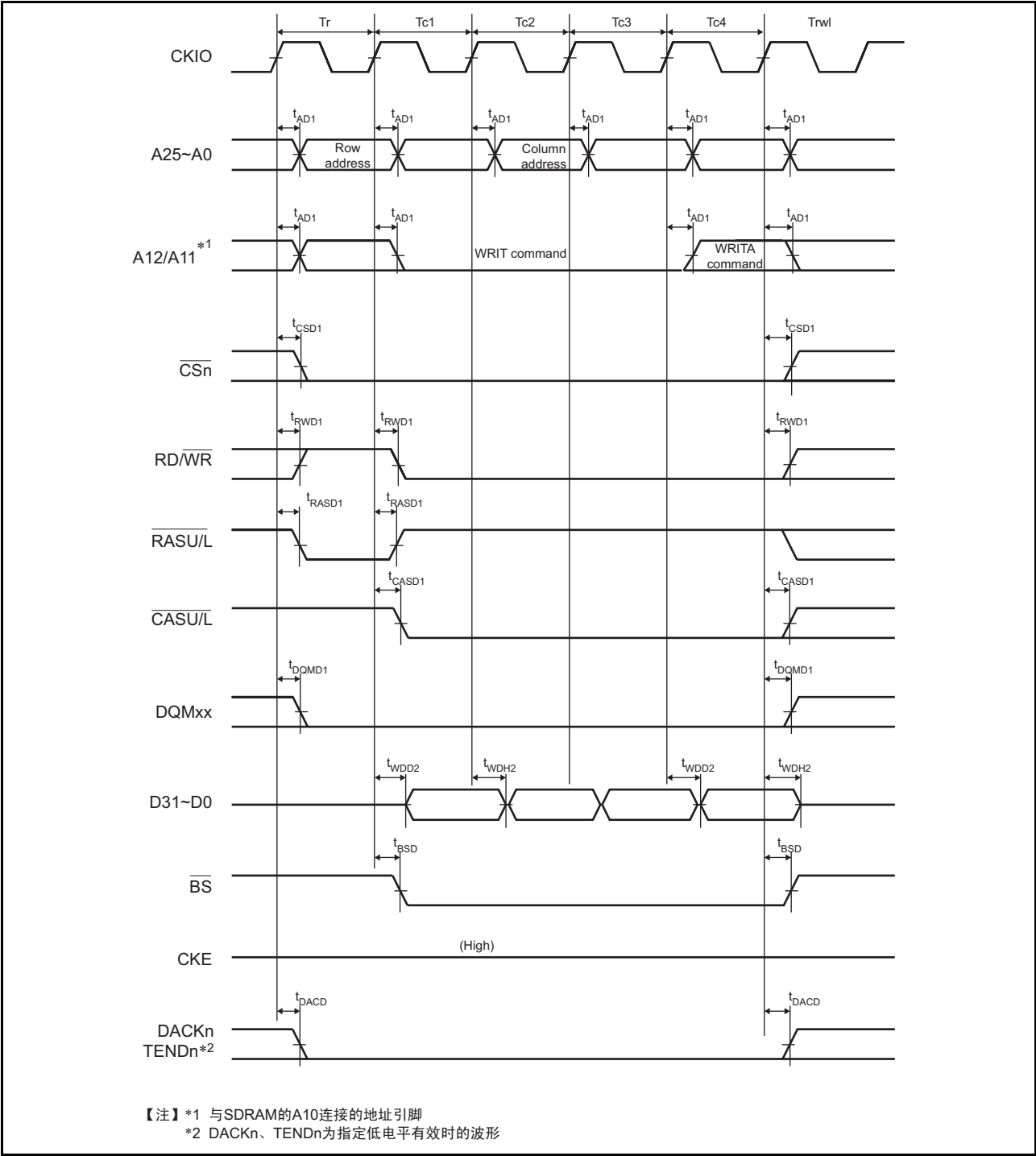


图 31.28 同步 DRAM 单次写入总线周期（写入 4 个周期）
（有自动预充电、WTRCD=0 个周期、TRWL=1 个周期）

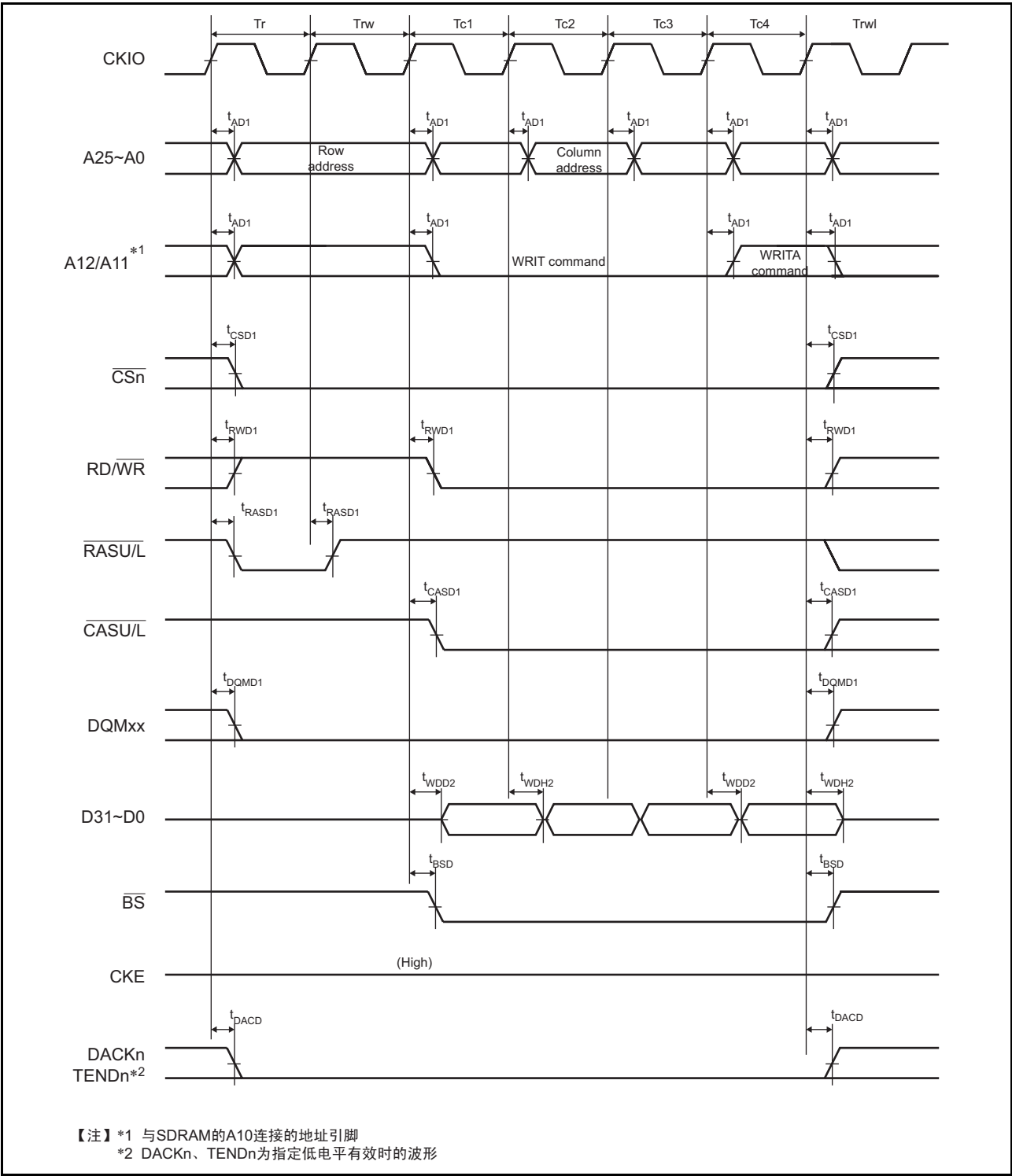


图 31.29 同步 DRAM 突发写入总线周期（写入 4 个周期）
（有自动预充电、WTRCD=1 个周期、TRWL=1 个周期）

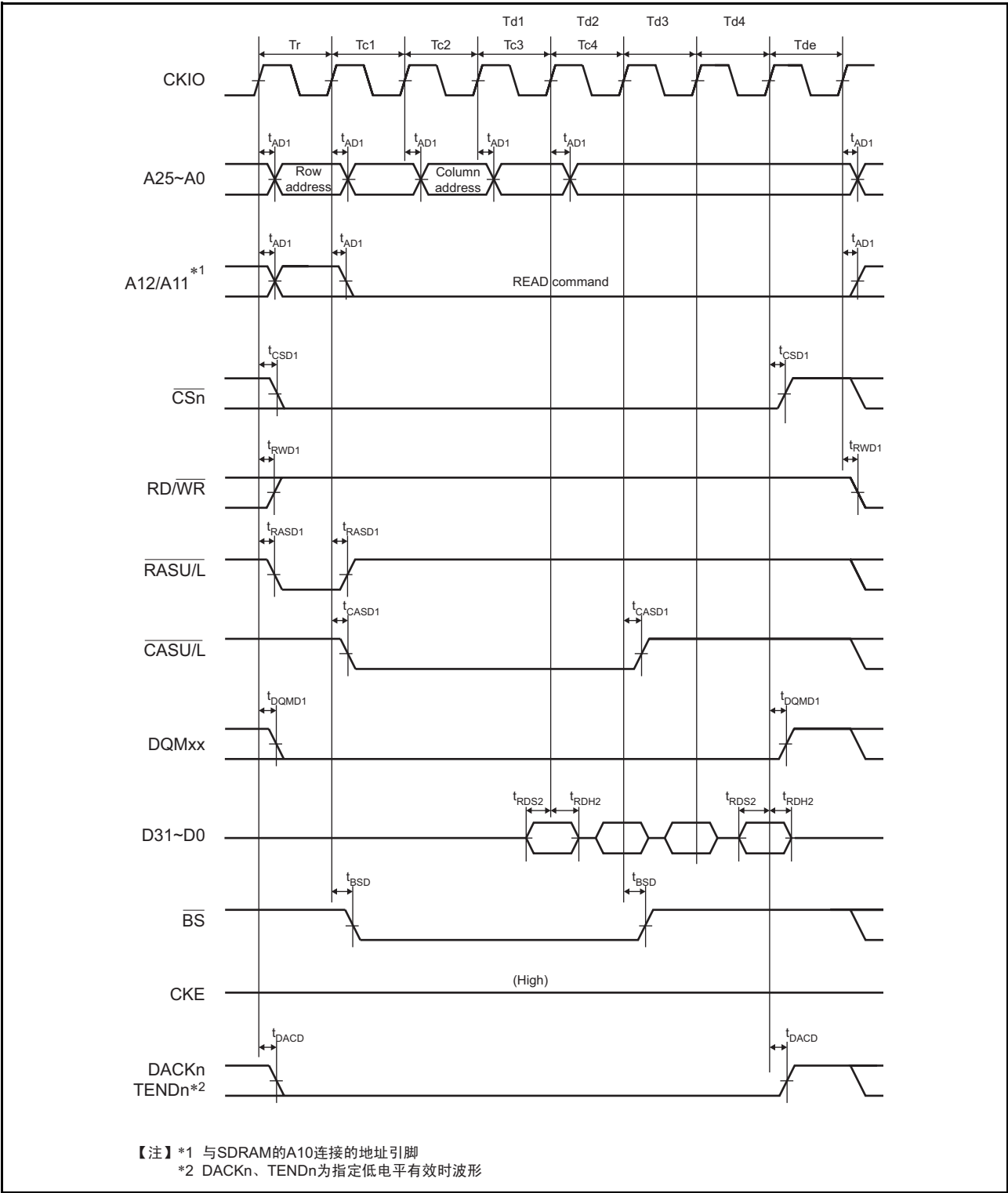


图 31.30 同步 DRAM 突发读取总线周期（读取 4 个周期）
（存储体激活模式：ACT+READ 命令、2 个 CAS 等待时间、WTRCD=0 个周期）

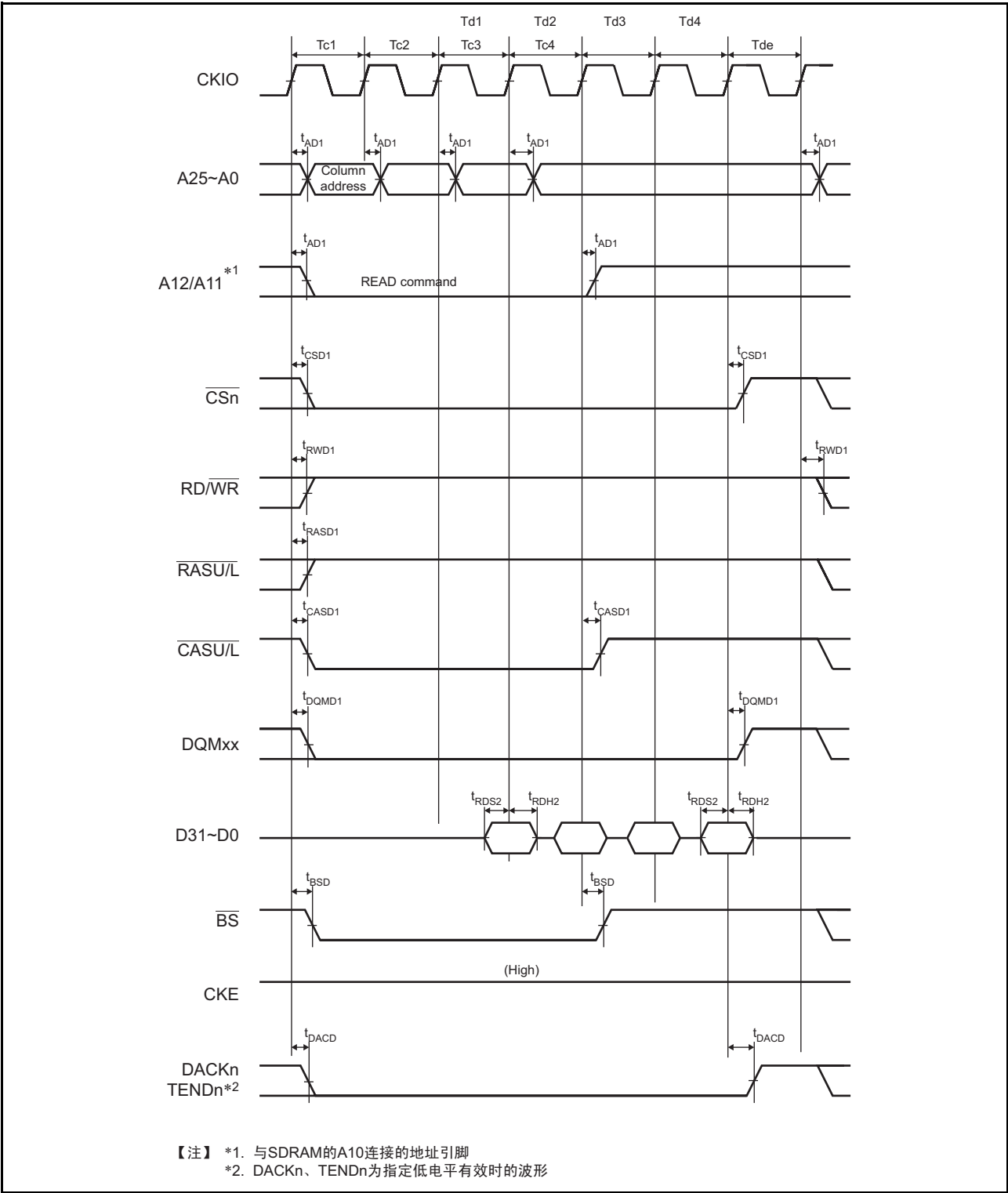


图 31.31 同步 DRAM 突发读取总线周期（读取 4 个周期）
（存储体激活模式：READ 命令、相同的行地址、2 个 CAS 等待时间、WTRCD=0 个周期）

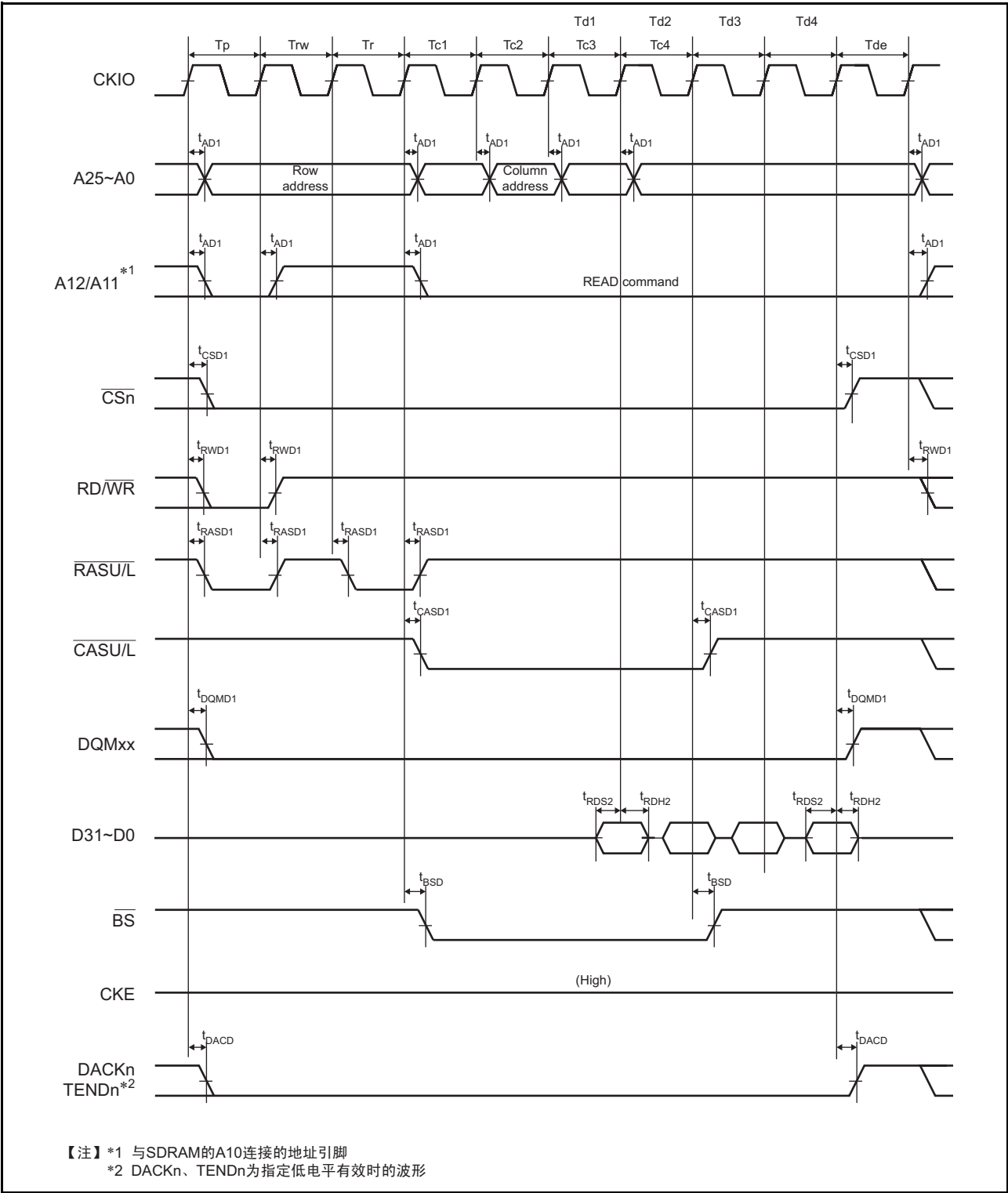


图 31.32 同步 DRAM 突发读取总线周期（读入 4 个周期）
（存储体激活模式：PRE+ACT+READ 命令、不同的行地址、2 个 CAS 等待时间、WTRCD=0 个周期）

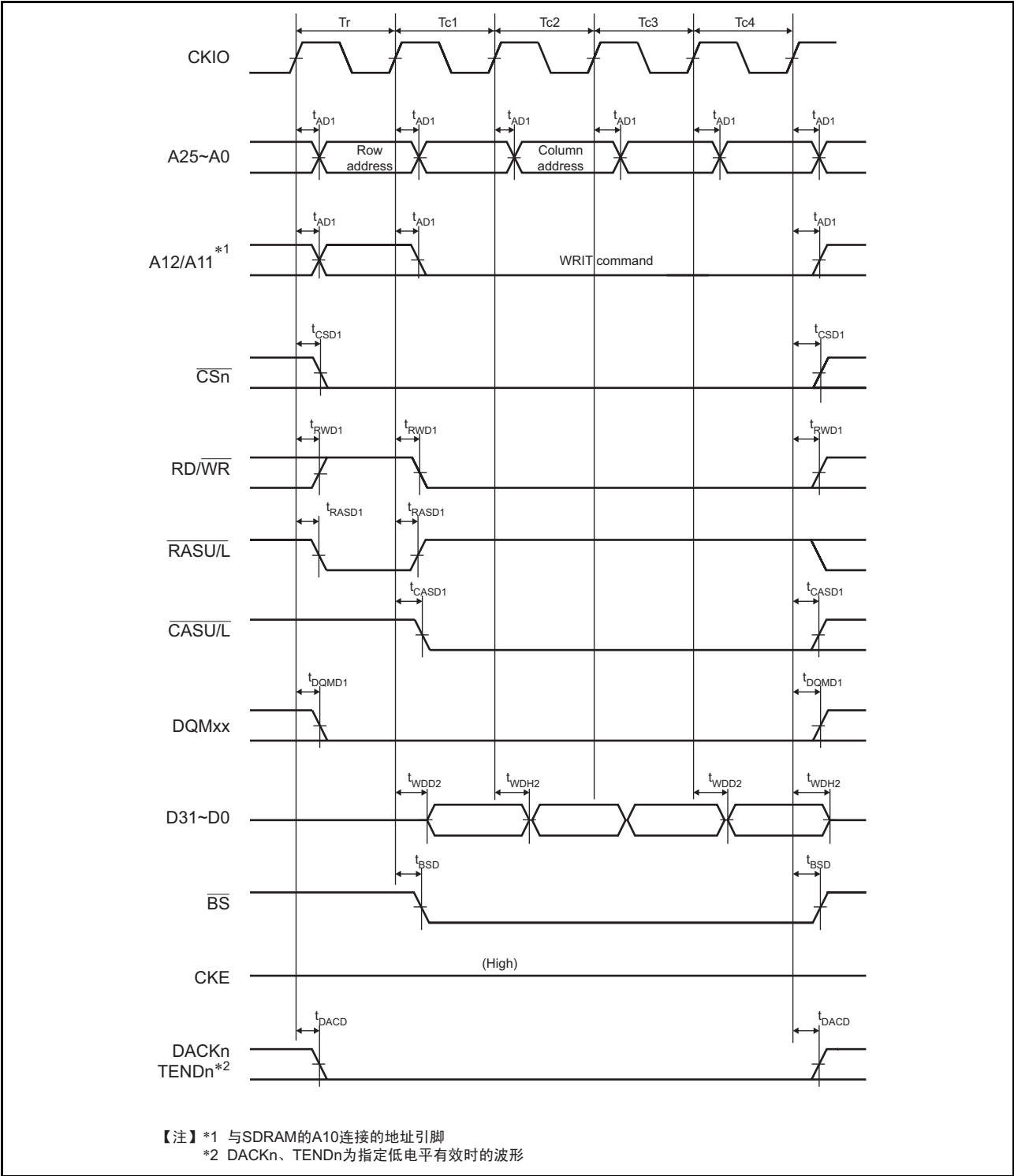


图 31.33 同步 DRAM 突发写入总线周期（写入 4 个周期）
 （存储体激活模式：ACT+WRITE 命令、WTRCD=0 个周期、TRWL=0 个周期）

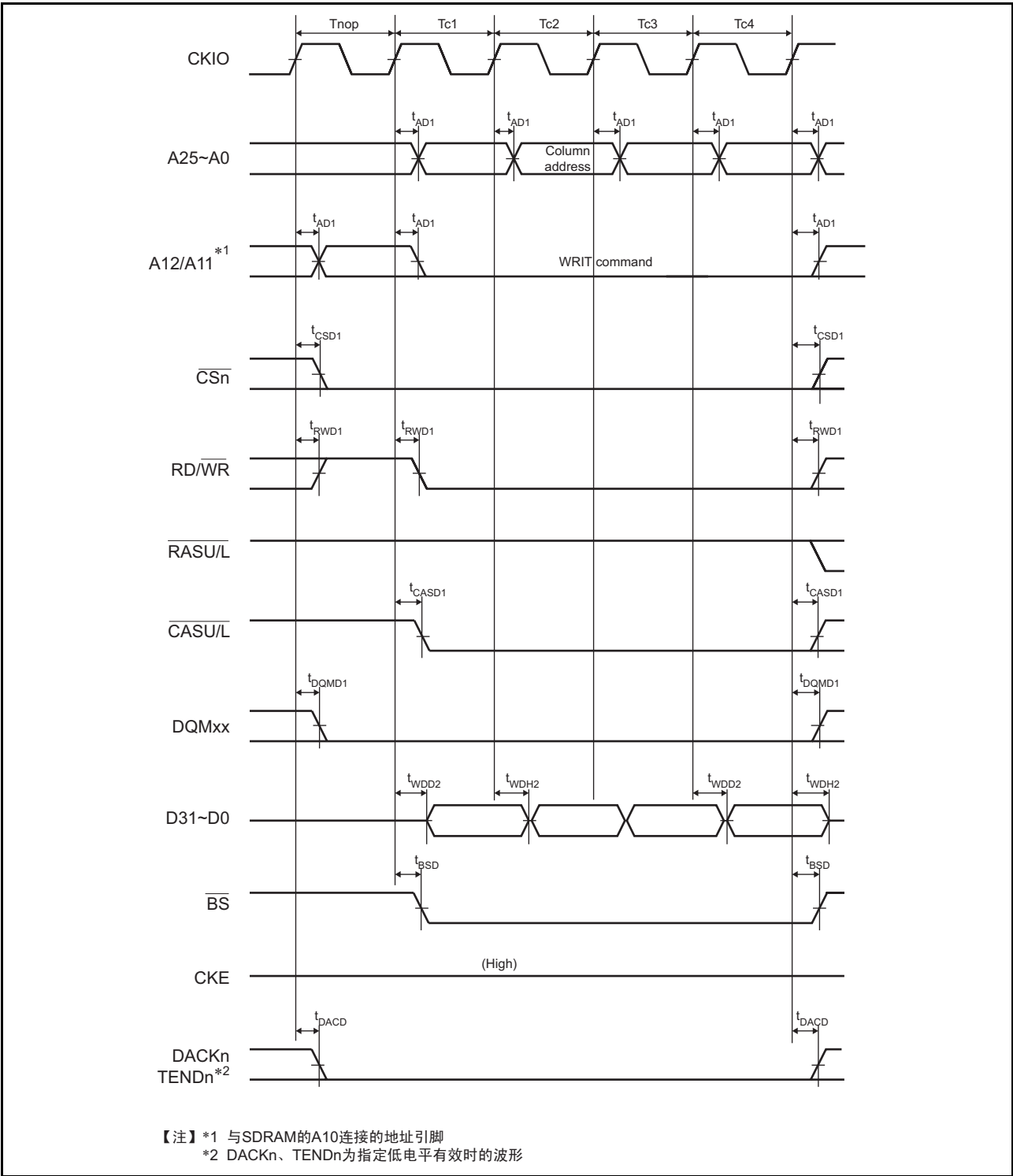


图 31.34 同步 DRAM 突发写入总线周期（写入 4 个周期）
（存储体激活模式：WRITE 命令、相同的行地址、WTRCD=0 个周期、TRWL=0 个周期）

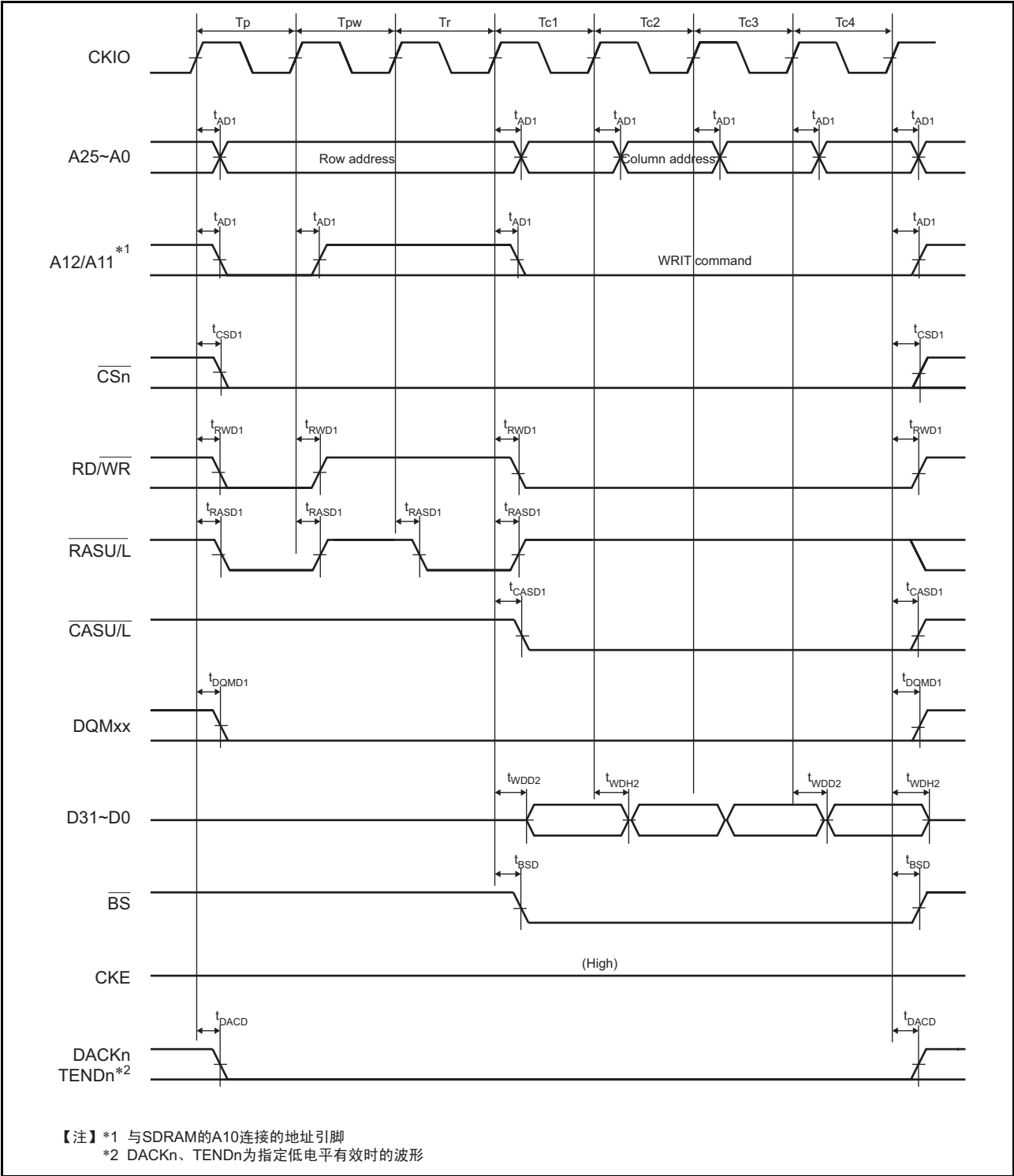


图 31.35 同步 DRAM 突发写入总线周期（写入 4 个周期）
（存储体激活模式：PRE+ACT+WRITE 命令、不同的行地址、WTRCD=0 个周期、TRWL=0 个周期）

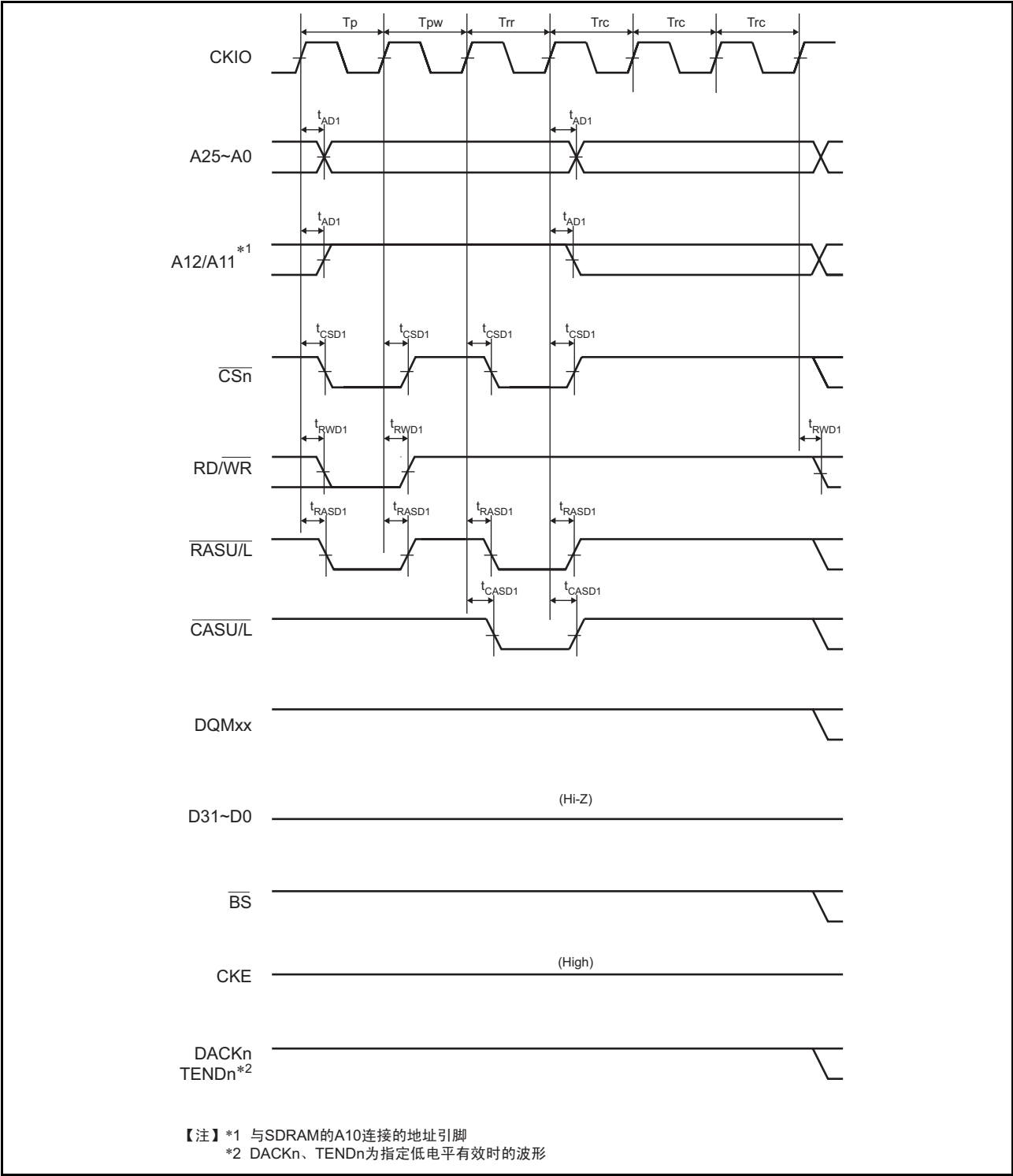


图 31.36 同步 DRAM 自刷新时序
(WTRP=1 个周期、WTRC=3 个周期)

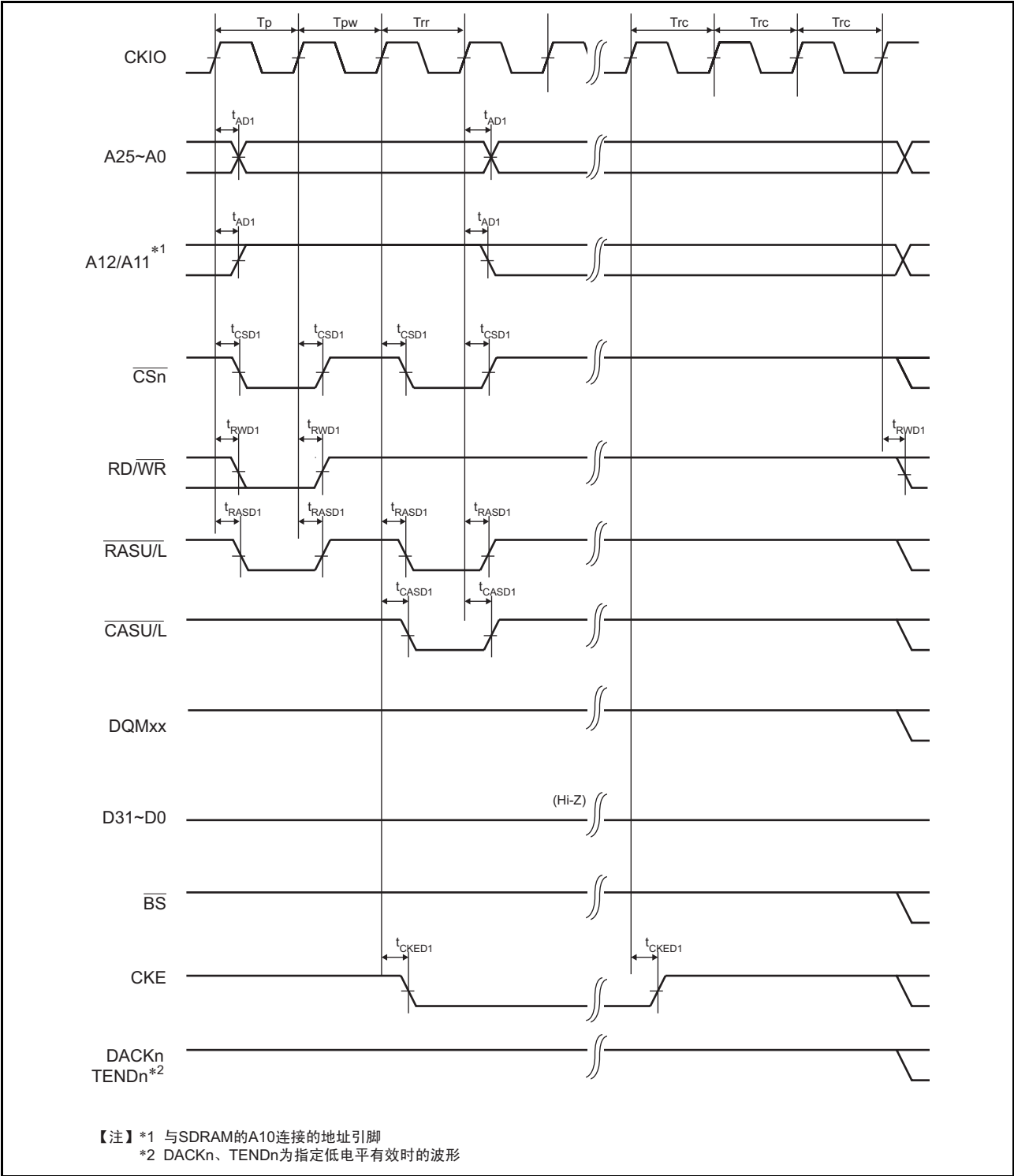


图 31.37 同步 DRAM 自刷新时序
(WTRP=1 个周期)

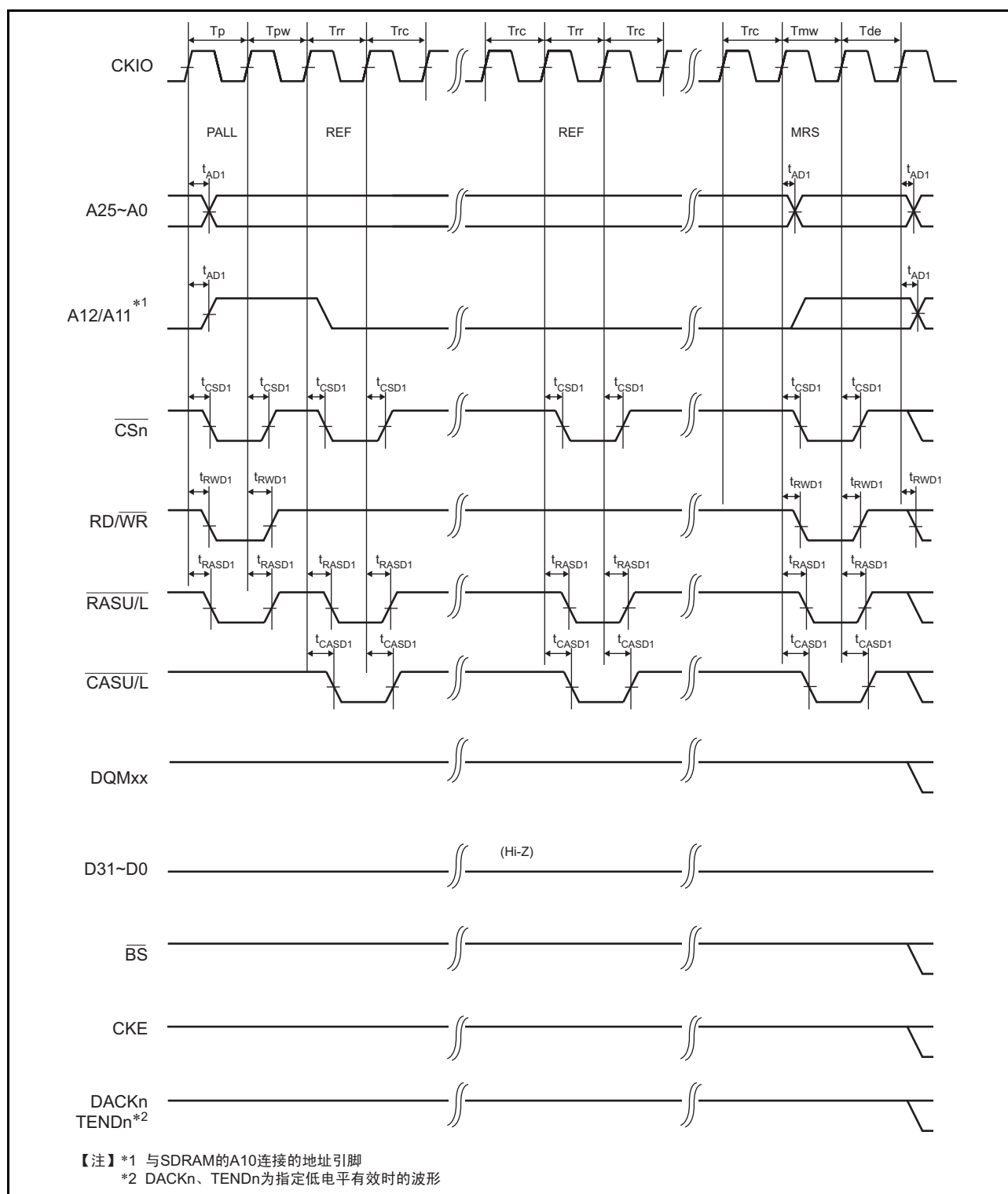


图 31.38 同步 DRAM 模式寄存器写入时序
(WTRP=1 个周期)

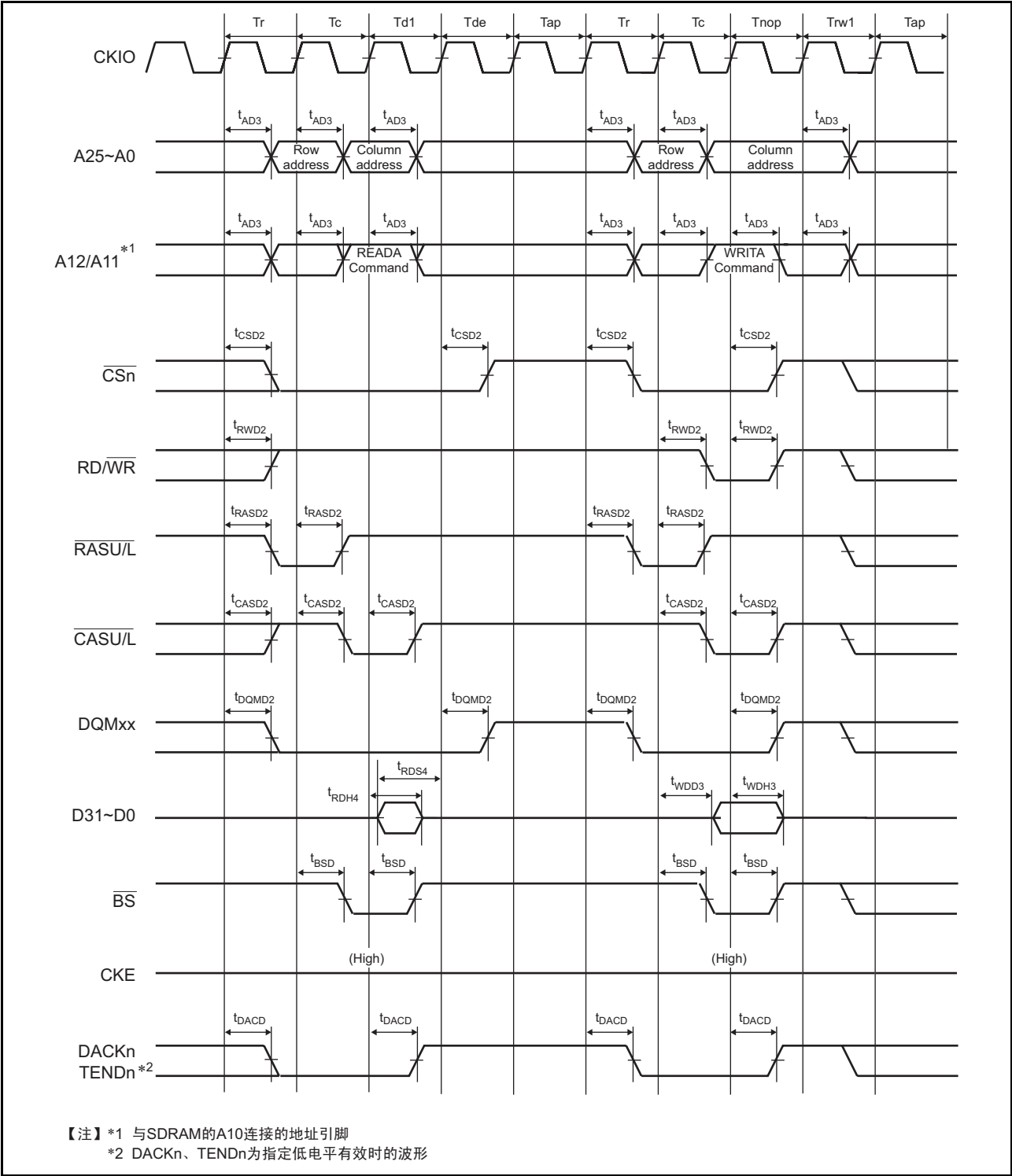


图 31.39 同步 DRAM 低频模式的存取时序
(自动预充电模式、TRWL=2 个周期)

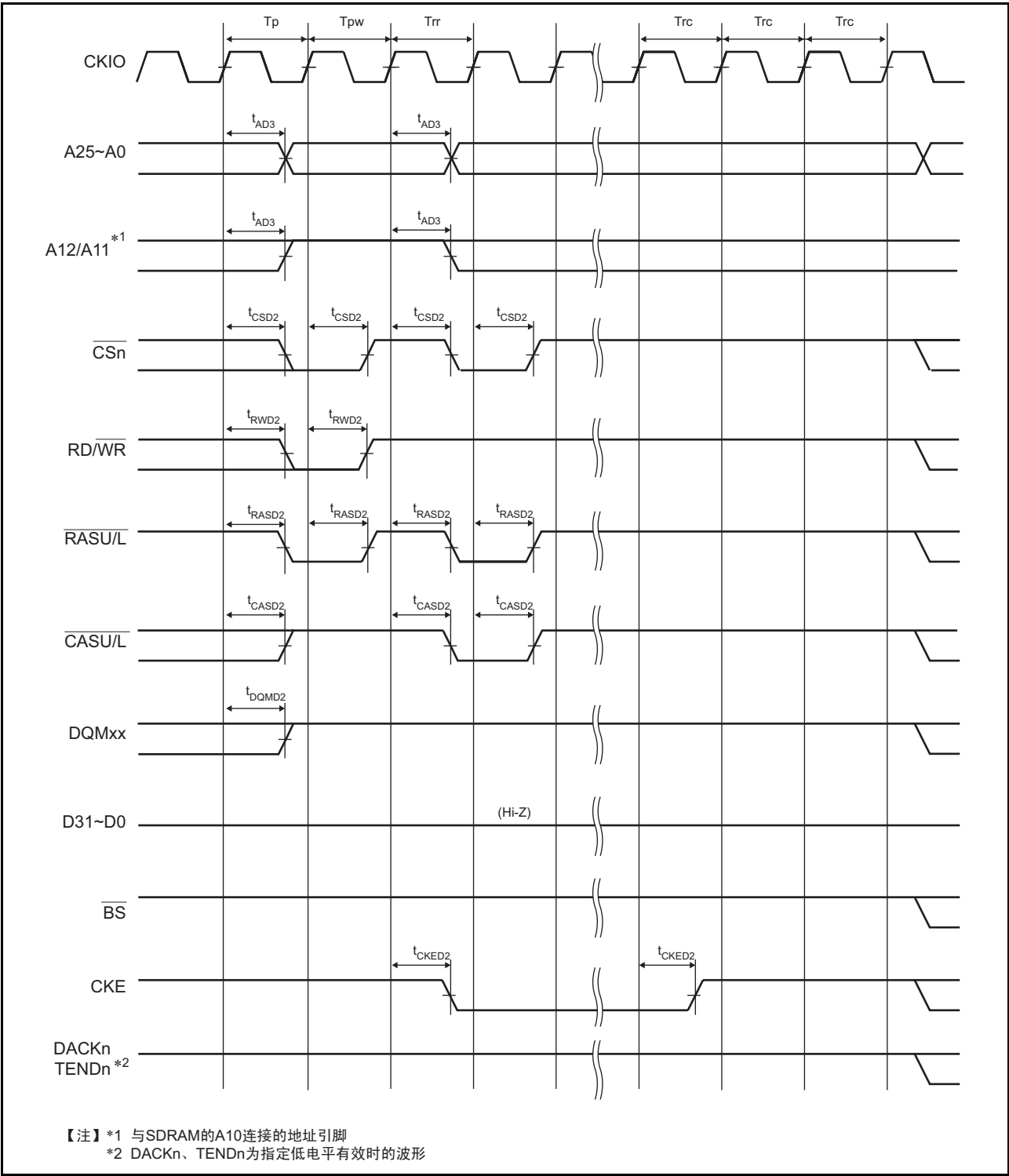


图 31.40 同步 DRAM 低频模式自刷新时序 (WTRP=2 个周期)

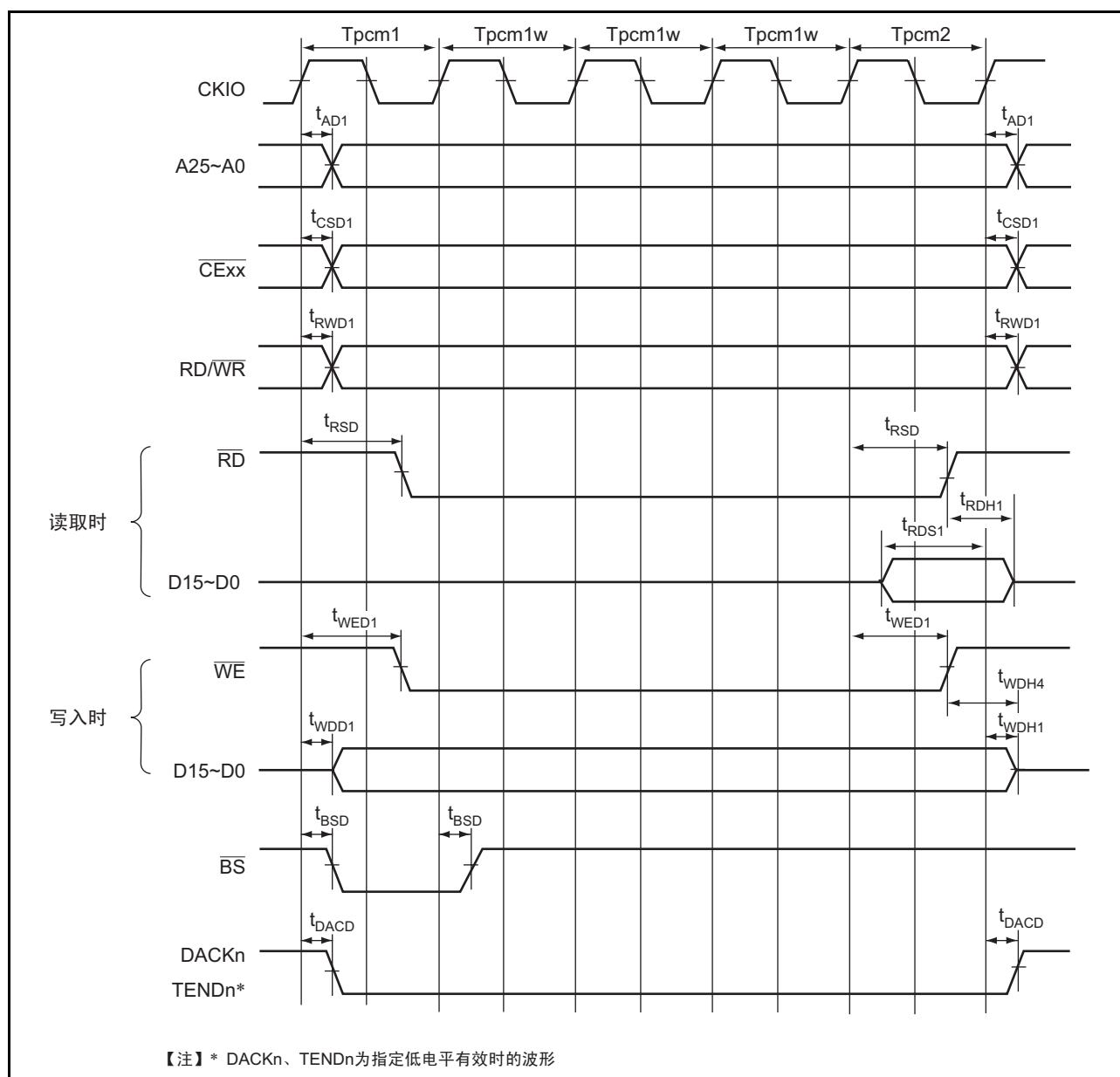


图 31.41 PCMCIA 存储卡总线周期
(TED=0 个周期、TEH=0 个周期、无等待)

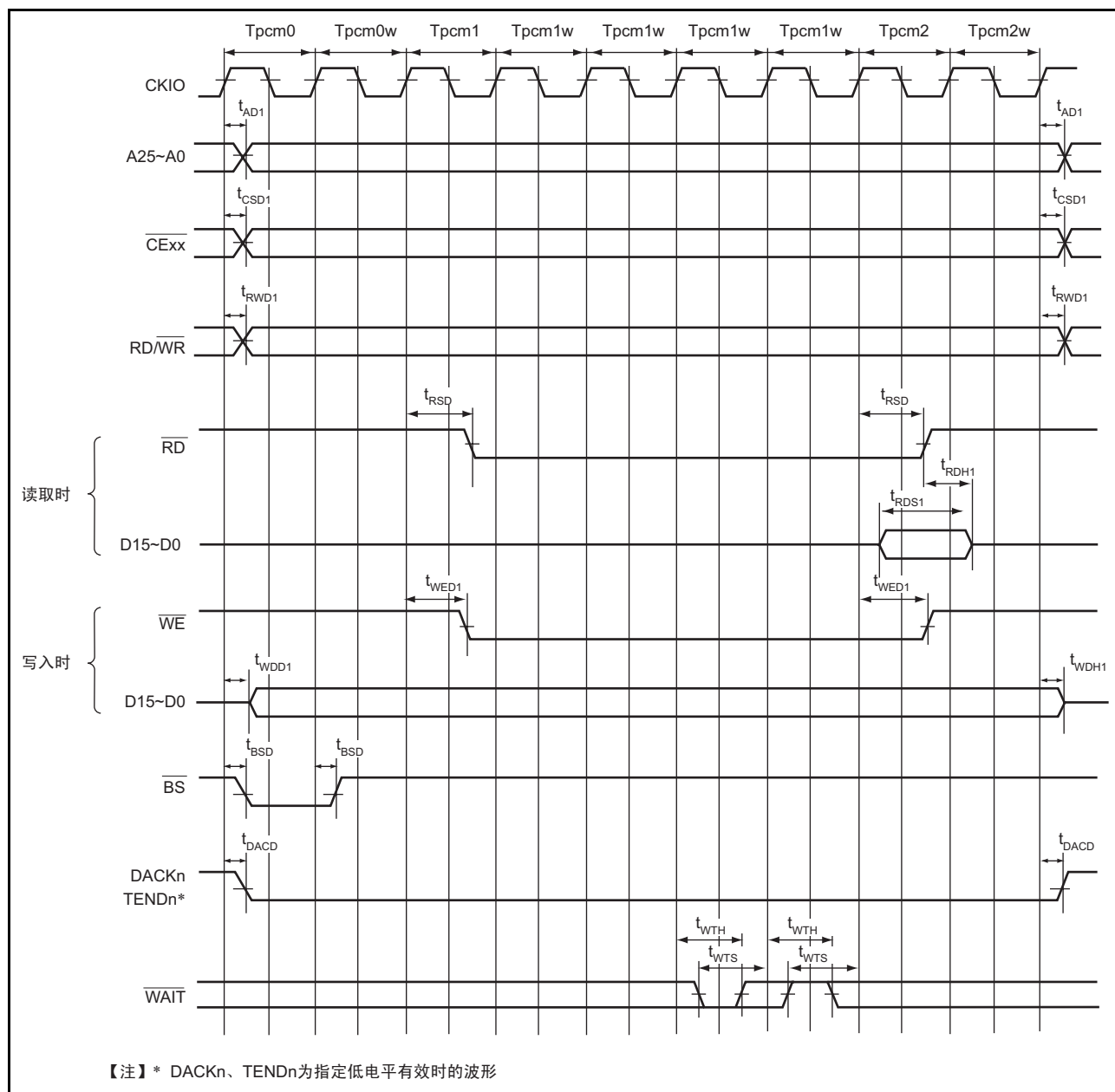


图 31.42 PCMCIA 存储卡总线周期
(TED=2 个周期、TEH=1 个周期、0 个软件等待、1 个硬件等待)

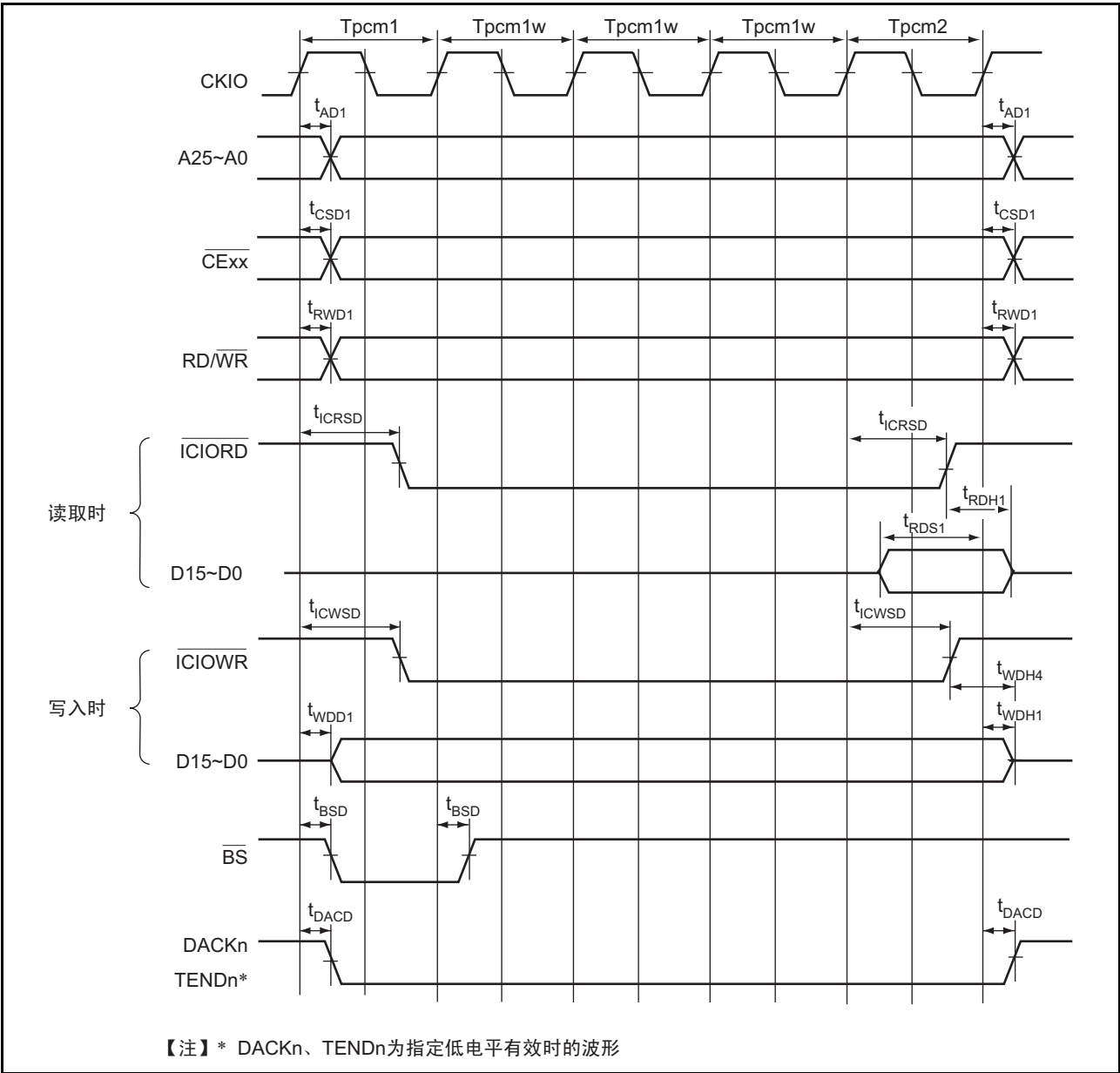


图 31.43 PCMCIA I/O 卡总线周期
(TED=0 个周期、TEH=0 个周期、无等待)

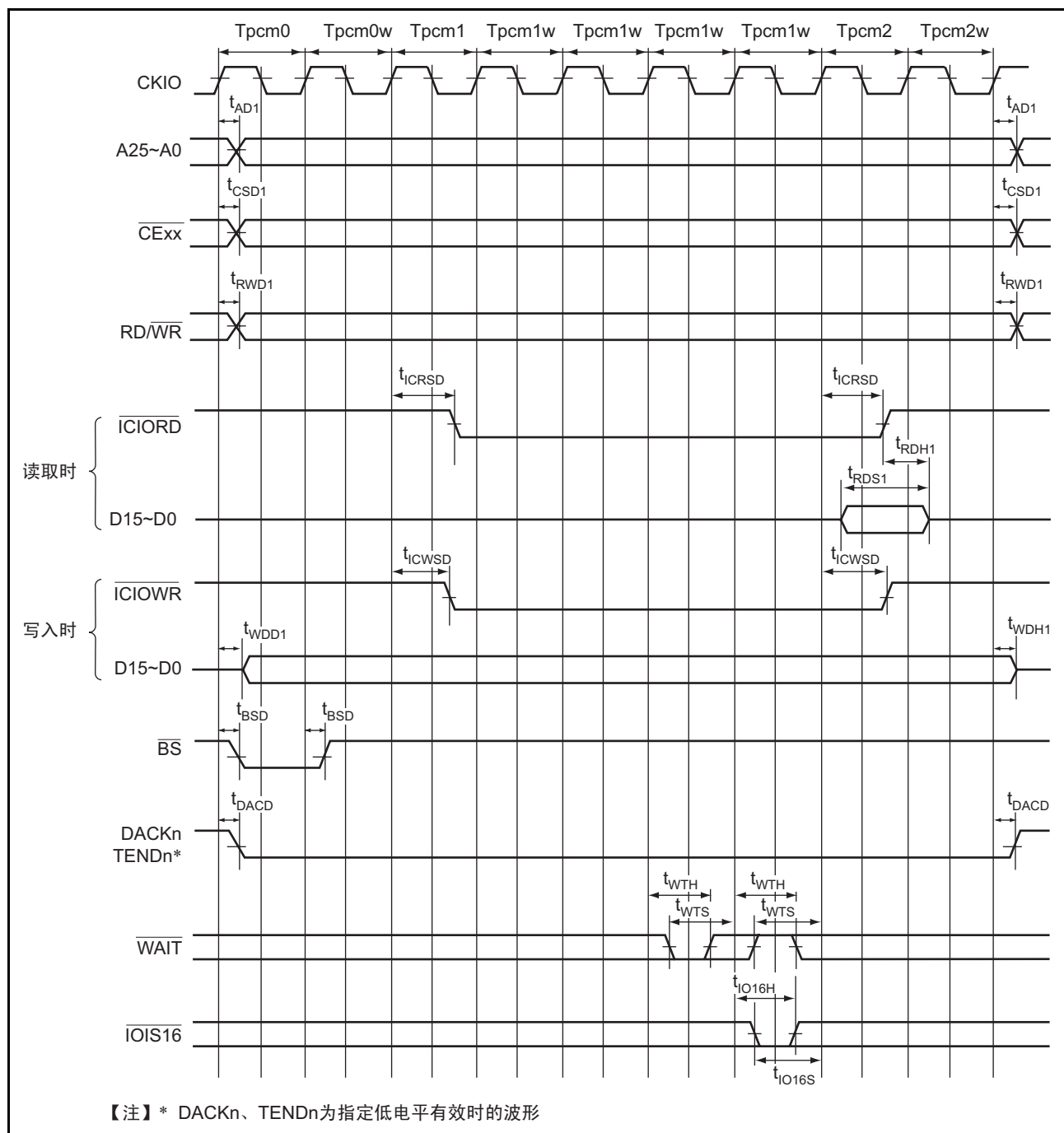


图 31.44 PCMCIA I/O 卡总线周期
(TED=2 个周期、TEH=1 个周期、0 个软件等待、1 个硬件等待)

31.4.4 UBC 时序

表 31.9 UBC 时序

条件: $V_{CC}=PLL_{VCC}=USBDV_{CC}=1.1 \sim 1.3V$ 、 $PV_{CC}=USBDPV_{CC}=3.0 \sim 3.6V$ 、 $AV_{CC}=3.0 \sim 3.6V$ 、
 $USBAV_{CC}=1.1 \sim 1.3V$ 、 $USBAPV_{CC}=3.0 \sim 3.6V$ 、
 $V_{SS}=PLL_{VSS}=PV_{SS}=AV_{SS}=USBDV_{SS}=USBAV_{SS}=USBAPV_{SS}=0V$ 、 $T_a=-20 \sim 85^{\circ}C$

项 目	符号	Min.	Max.	单位	参照图
UBCTRG 延迟时间	t_{UBCTGD}	—	14	ns	31.45

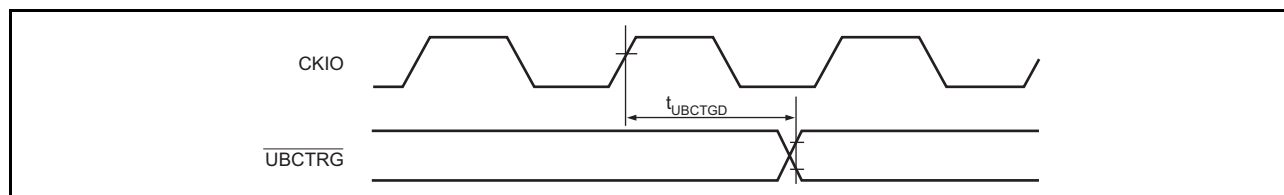


图 31.45 UBC 触发时序

31.4.5 DMAC 时序

表 31.10 DMAC 时序

条件: $V_{CC}=PLL_{VCC}=USBDV_{CC}=1.1 \sim 1.3V$ 、 $PV_{CC}=USBDPV_{CC}=3.0 \sim 3.6V$ 、 $AV_{CC}=3.0 \sim 3.6V$ 、
 $USBAV_{CC}=1.1 \sim 1.3V$ 、 $USBAPV_{CC}=3.0 \sim 3.6V$ 、
 $V_{SS}=PLL_{VSS}=PV_{SS}=AV_{SS}=USBDV_{SS}=USBAV_{SS}=USBAPV_{SS}=0V$ 、 $T_a=-20 \sim 85^{\circ}C$

项 目	符号	Min.	Max.	单位	参照图
DREQ 准备时间	t_{DRQS}	15	—	ns	31.46
DREQ 保持时间	t_{DRQH}	15	—		
DACK、TEND 延迟时间	t_{DACD}	0	13		31.47

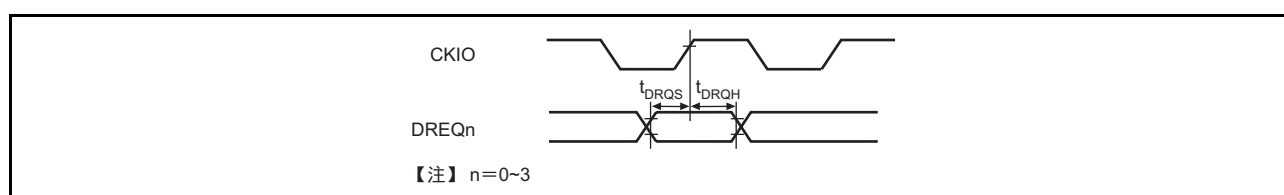


图 31.46 DREQ 输入时序

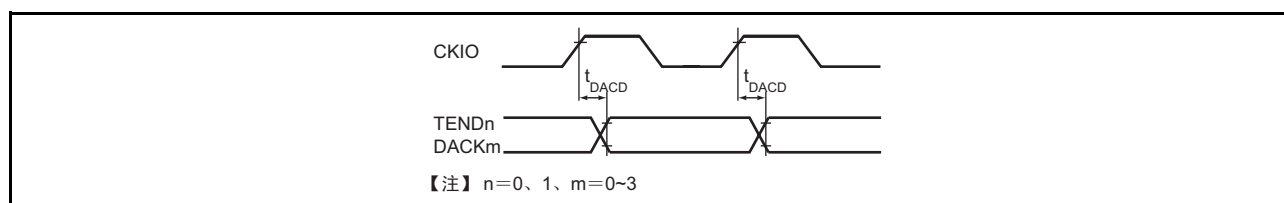


图 31.47 DACK、TEND 输出时序

31.4.6 MTU2 时序

表 31.11 MTU2 时序

条件： $V_{CC}=PLL_{V_{CC}}=USBDV_{CC}=1.1 \sim 1.3V$ 、 $PV_{CC}=USBDPV_{CC}=3.0 \sim 3.6V$ 、 $AV_{CC}=3.0 \sim 3.6V$ 、
 $USBAV_{CC}=1.1 \sim 1.3V$ 、 $USBAPV_{CC}=3.0 \sim 3.6V$ 、
 $V_{SS}=PLL_{V_{SS}}=PV_{SS}=AV_{SS}=USBDV_{SS}=USBAV_{SS}=USBDPV_{SS}=USBAPV_{SS}=0V$ 、 $T_a=-20 \sim 85^{\circ}C$

项 目	符号	Min.	Max.	单位	参照图
输出比较的输出延迟时间	t_{TOCD}	—	100	ns	31.48
输入捕捉的输入准备时间	t_{TICS}	$t_{cyc}/2+20$	—	ns	
定时器输入准备时间	t_{TCKS}	$t_{cyc}+20$	—	ns	31.49
定时器时钟脉宽（指定单边沿）	$t_{TCKWH/L}$	1.5	—	t_{pcyc}	
定时器时钟脉宽（指定双边沿）	$t_{TCKWH/L}$	2.5	—	t_{pcyc}	
定时器时钟脉宽（相位计数模式）	$t_{TCKWH/L}$	2.5	—	t_{pcyc}	

【注】 t_{pcyc} 表示外围时钟（ $P\phi$ ）的周期。

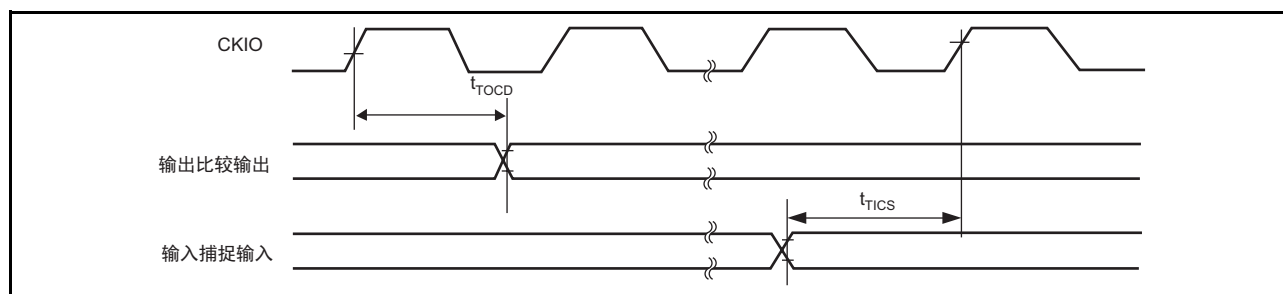


图 31.48 MTU2 输入 / 输出时序

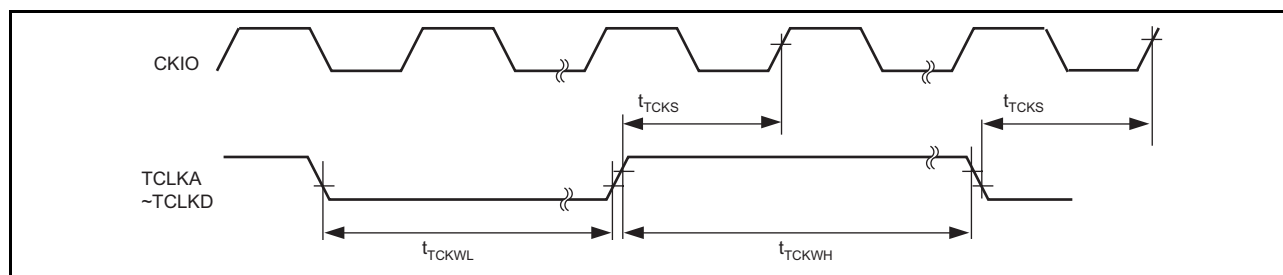


图 31.49 MTU2 时钟输入时序

31.4.7 WDT 时序

表 31.12 WDT 时序

条件: $V_{CC}=PLL_{VCC}=USBDV_{CC}=1.1 \sim 1.3V$ 、 $PV_{CC}=USBDPV_{CC}=3.0 \sim 3.6V$ 、 $AV_{CC}=3.0 \sim 3.6V$ 、
 $USBAV_{CC}=1.1 \sim 1.3V$ 、 $USBAPV_{CC}=3.0 \sim 3.6V$ 、
 $V_{SS}=PLL_{VSS}=PV_{SS}=AV_{SS}=USBDV_{SS}=USBAV_{SS}=USBDPV_{SS}=0V$ 、 $T_a = -20 \sim 85^{\circ}C$

项 目	符号	Min.	Max.	单位	参照图
WDTOVF 延迟时间	t_{WOVD}	—	100	ns	31.50

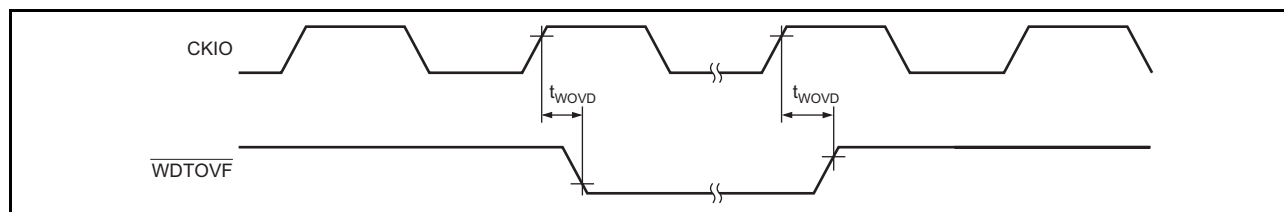


图 31.50 WDT 时序

31.4.8 SCIF 时序

表 31.13 SCIF 时序

条件: $V_{CC}=PLL_{VCC}=USBDV_{CC}=1.1 \sim 1.3V$ 、 $PV_{CC}=USBDPV_{CC}=3.0 \sim 3.6V$ 、 $AV_{CC}=3.0 \sim 3.6V$ 、
 $USBAV_{CC}=1.1 \sim 1.3V$ 、 $USBAPV_{CC}=3.0 \sim 3.6V$ 、
 $V_{SS}=PLL_{VSS}=PV_{SS}=AV_{SS}=USBDV_{SS}=USBAV_{SS}=USBDPV_{SS}=USBAV_{SS}=0V$ 、 $T_a = -20 \sim 85^{\circ}C$

项 目		符号	Min.	Max.	单位	参照图
输入时钟周期	时钟同步	t_{Scyc}	12	—	t_{pcyc}	31.51
	异步		4	—	t_{pcyc}	31.51
输入时钟上升时间		t_{SCKr}	—	1.5	t_{pcyc}	31.51
输入时钟下降时间		t_{SCKf}	—	1.5	t_{pcyc}	31.51
输入时钟宽度		t_{SCKW}	0.4	0.6	t_{Scyc}	31.51
发送数据延迟时间（时钟同步）		t_{TXD}	—	$3t_{pcyc}+15$	ns	31.52
接收数据准备时间（时钟同步）		t_{RXS}	$4t_{pcyc}+15$	—	ns	31.52
接收数据保持时间（时钟同步）		t_{RXH}	$1t_{pcyc}+15$	—	ns	31.52

【注】 t_{pcyc} 表示外围时钟 (P ϕ) 周期。

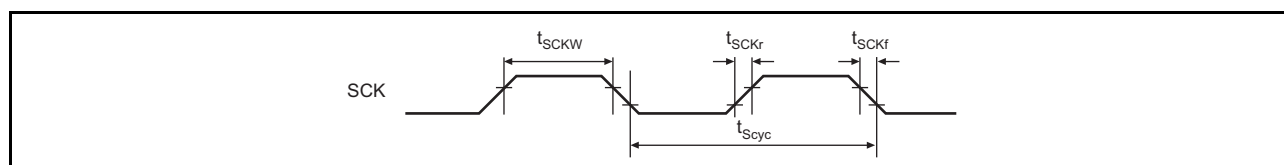


图 31.51 SCK 输入时钟时序

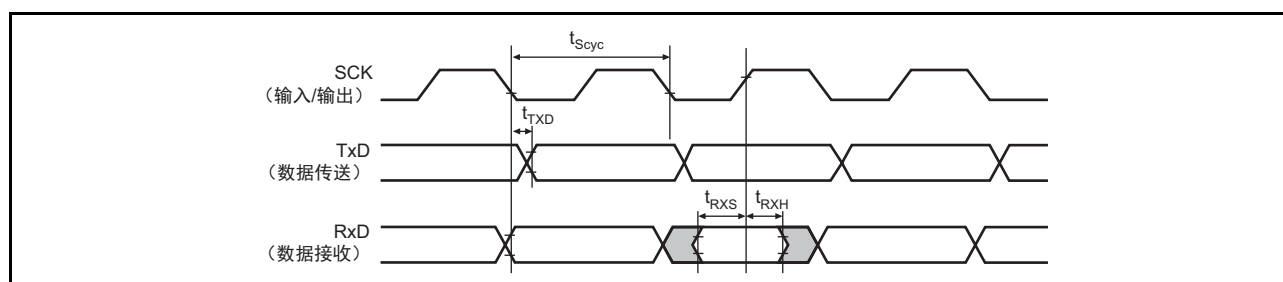


图 31.52 时钟同步模式时的 SCIF 输入 / 输出时序

31.4.9 SSU 时序

表 31.14 SSU 时序

条件: $V_{CC}=PLL_{VCC}=USBDV_{CC}=1.1 \sim 1.3V$ 、 $PV_{CC}=USBDPV_{CC}=3.0 \sim 3.6V$ 、 $AV_{CC}=3.0 \sim 3.6V$ 、
 $USBAV_{CC}=1.1 \sim 1.3V$ 、 $USBAPV_{CC}=3.0 \sim 3.6V$ 、
 $V_{SS}=PLL_{VSS}=PV_{SS}=AV_{SS}=USBDV_{SS}=USBAV_{SS}=USBDPV_{SS}=USBAPV_{SS}=0V$ 、 $T_a = -20 \sim 85^{\circ}C$

项 目		符号	Min.	Max.	单位	参照图
时钟周期	主模式	t_{SUcyc}	4	256	t_{pcyc}	31.53
	从属模式		4	256		31.54
时钟高电平脉宽	主模式	t_{HI}	48	—	ns	31.55
	从属模式		48	—		31.56
	主模式	t_{LO}	48	—	ns	
	从属模式		48	—		
时钟上升时间		t_{RISE}	—	12	ns	
时钟下降时间		t_{FALL}	—	12	ns	
数据输入准备时间	主模式	t_{SU}	30	—	ns	
	从属模式		20	—		
数据输入保持时间	主模式	t_{H}	0	—	ns	
	从属模式		20	—		
SCS 准备时间	主模式	t_{LEAD}	1.5	—	t_{pcyc}	
	从属模式		1.5	—		
SCS 保持时间	主模式	t_{LAG}	1.5	—	t_{pcyc}	
	从属模式		1.5	—		
数据输出延迟时间	主模式	t_{OD}	—	50	ns	
	从属模式		—	50		
数据输出保持时间	主模式	t_{OH}	0	—	ns	
	从属模式		0	—		
连续发送延迟时间	主模式	t_{TD}	1.5	—	t_{pcyc}	
	从属模式		1.5	—		
从属存取时间		t_{SA}	—	1	t_{pcyc}	31.55
从属输出开路时间		t_{REL}	—	1	t_{pcyc}	31.56

【注】 t_{pcyc} 表示外围时钟 (Pφ) 周期。

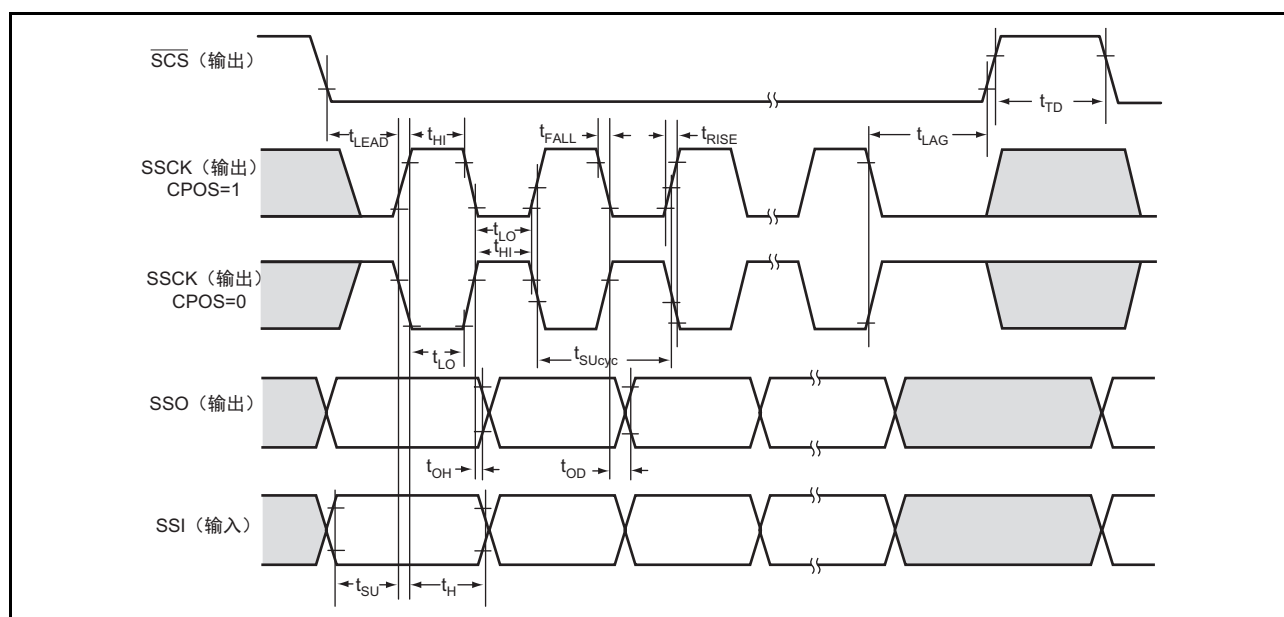


图 31.53 SSU 时序（主模式、CPHS=1）

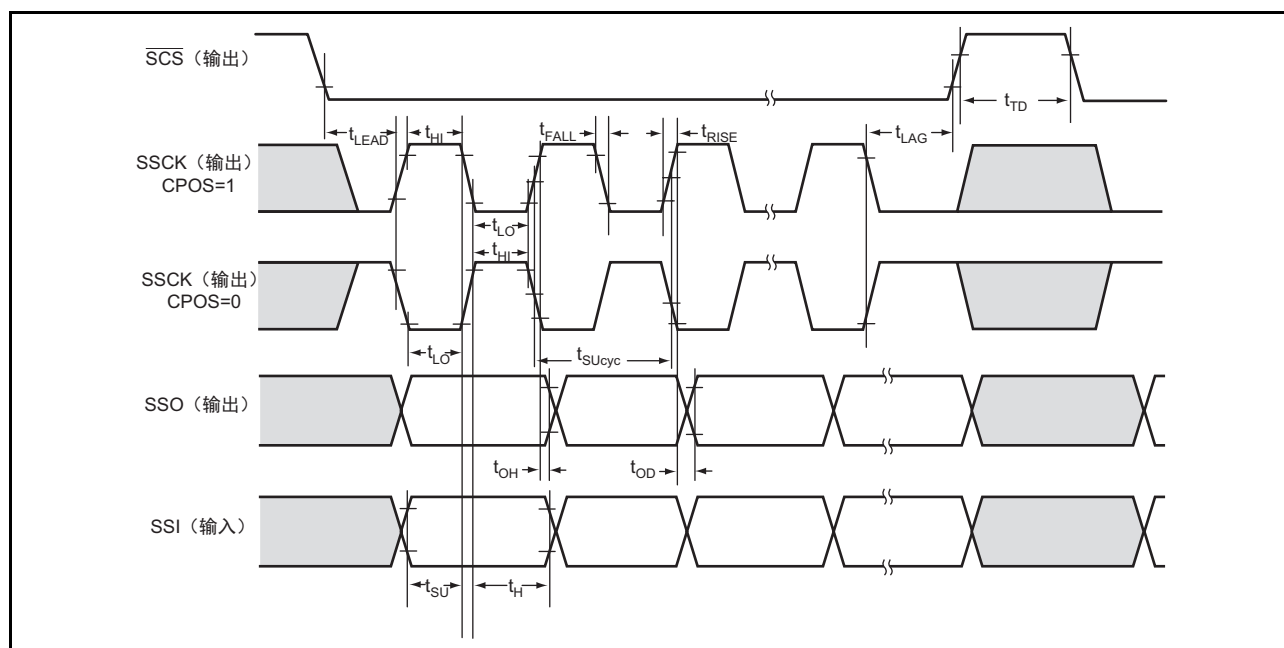


图 31.54 SSU 时序（主模式、CPHS=0）

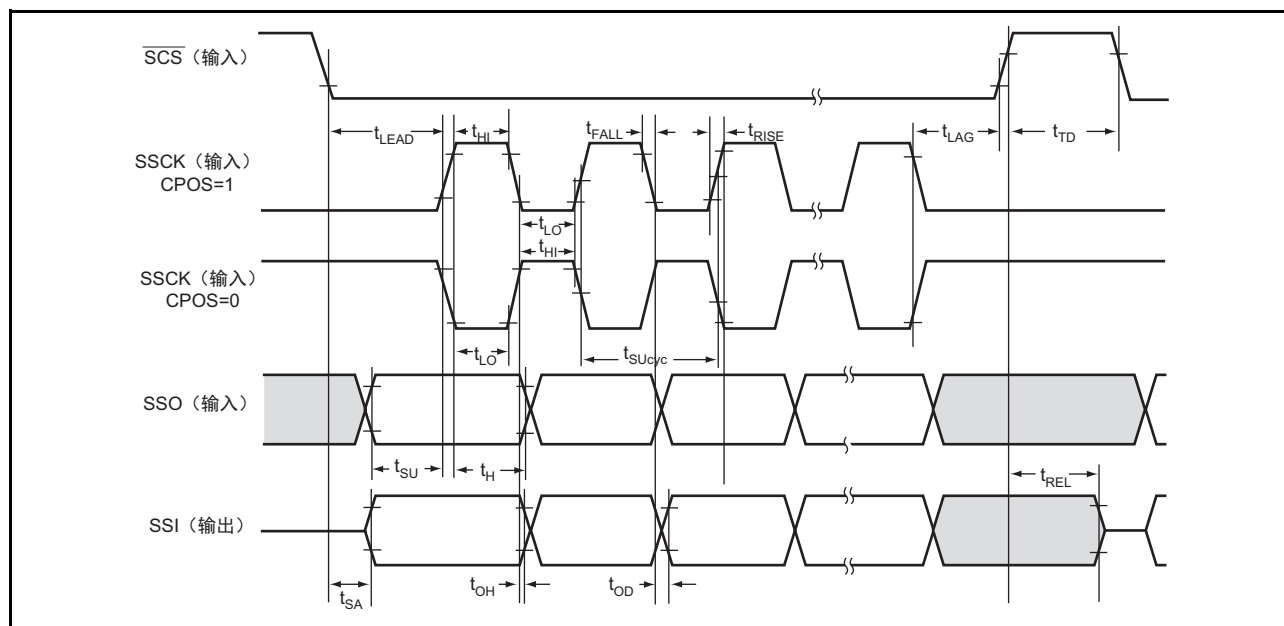


图 31.55 SSU 时序 (从属模式、CPHS=1)

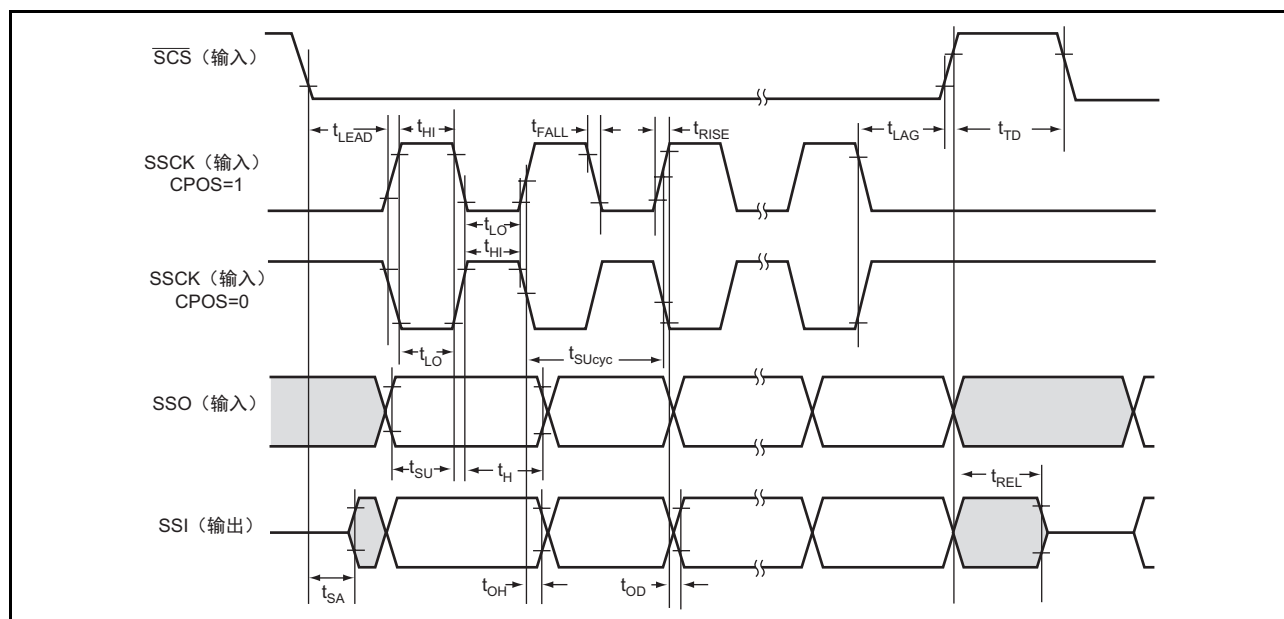


图 31.56 SSU 时序 (从属模式、CPHS=0)

31.4.10 IIC3 时序

表 31.15 IIC3 时序

条件: $V_{CC}=PLL_{VCC}=USBDV_{CC}=1.1 \sim 1.3V$ 、 $PV_{CC}=USBDPV_{CC}=3.0 \sim 3.6V$ 、 $AV_{CC}=3.0 \sim 3.6V$ 、
 $USBAV_{CC}=1.1 \sim 1.3V$ 、 $USBAPV_{CC}=3.0 \sim 3.6V$ 、
 $V_{SS}=PLL_{VSS}=PV_{SS}=AV_{SS}=USBDV_{SS}=USBAV_{SS}=USBAPV_{SS}=0V$ 、 $T_a = -20 \sim 85^\circ C$

项 目	符号	Min.	Max.	单位	参照图
SCL 输入周期时间	t_{SCL}	$12t_{pcyc}^{*1}+600$	—	ns	31.57
SCL 输入高电平脉宽	t_{SCLH}	$3t_{pcyc}^{*1}+300$	—	ns	
SCL 输入低电平脉宽	t_{SCLL}	$5t_{pcyc}^{*1}+300$	—	ns	
SCL、SDA 输入上升时间	t_{Sr}	—	300	ns	
SCL、SDA 输入下降时间	t_{Sf}	—	300	ns	
SCL、SDA 输入尖脉冲消除时间 *2	t_{SP}	—	1、2	t_{pcyc}^{*1}	
SDA 输入总线自由时间	t_{BUF}	5	—	t_{pcyc}^{*1}	
开始条件输入保持时间	t_{STAH}	3	—	t_{pcyc}^{*1}	
重新发送开始条件输入准备时间	t_{STAS}	3	—	t_{pcyc}^{*1}	
停止条件输入准备时间	t_{STOS}	3	—	t_{pcyc}^{*1}	
数据输入准备时间	t_{SDAS}	$1t_{pcyc}^{*1}+20$	—	ns	
数据输入保持时间	t_{SDAH}	0	—	ns	
SCL、SDA 的电容负载	Cb	0	400	pF	
SCL、SDA 输出下降沿时间 *3	t_{Sf}	—	250	ns	

【注】 *1 t_{pcyc} 表示外围时钟 ($P\phi$) 周期。

*2 取决于 NF2CYC 寄存器值。

*3 表示 I/O 缓冲器特性。

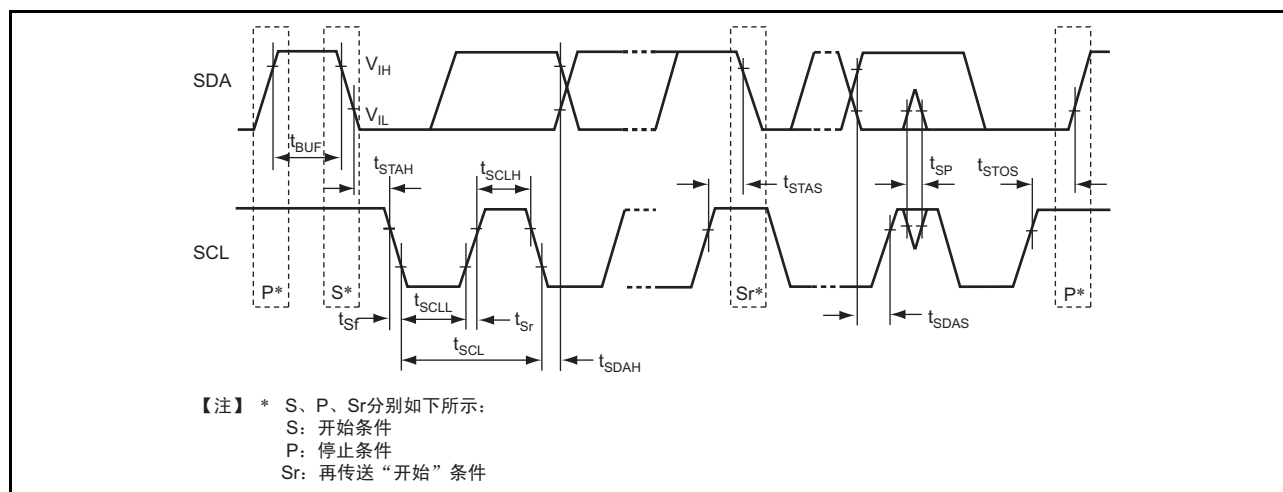


图 31.57 IIC3 输入 / 输出时序

31.4.11 SSI 时序

表 31.16 SSI 时序

条件: $V_{CC}=PLL V_{CC}=USBDV_{CC}=1.1 \sim 1.3V$ 、 $PV_{CC}=USBDPV_{CC}=3.0 \sim 3.6V$ 、 $AV_{CC}=3.0 \sim 3.6V$ 、
 $USBAV_{CC}=1.1 \sim 1.3V$ 、 $USBAPV_{CC}=3.0 \sim 3.6V$ 、
 $V_{SS}=PLL V_{SS}=PV_{SS}=AV_{SS}=USBDV_{SS}=USBAV_{SS}=USBDPV_{SS}=USBAPV_{SS}=0V$ 、 $T_a=-20 \sim 85^{\circ}C$

项 目	符号	Min.	Max.	单位	备注	参照图
输出时钟周期	t _O	80	64000	ns	输出	31.58
输入时钟周期	t _I	80	64000	ns	输入	
时钟高电平	t _{HC}	32	—	ns	双向	
时钟低电平	t _{LC}	32	—	ns		
时钟上升时间	t _{RC}	—	20	ns	输出 (100pF)	
延迟	t _{DTR}	−5	25	ns	发送	31.59、 31.60
准备时间	t _{SR}	25	—	ns	接收	
保持时间	t _{HTR}	5	—	ns	接收、发送	
AUDIO_CLK 输入频率	f _{AUDIO}	1	40	MHz		31.61

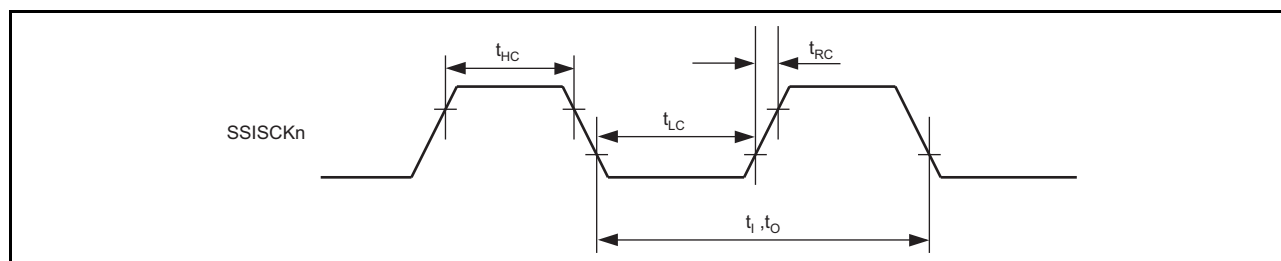


图 31.58 时钟输入 / 输出时序

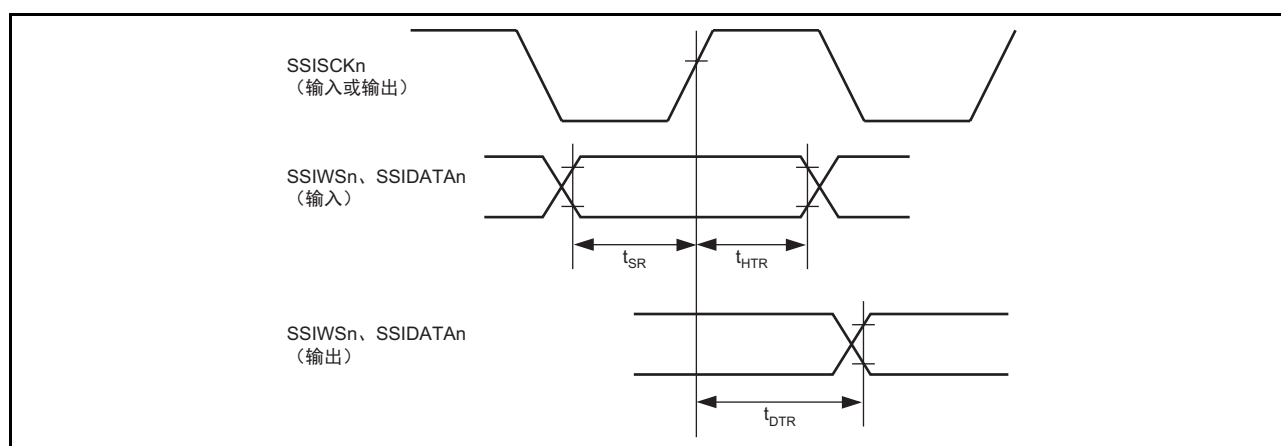


图 31.59 SSI 发送 / 接收时序 (SSISCKn 上升沿同步)

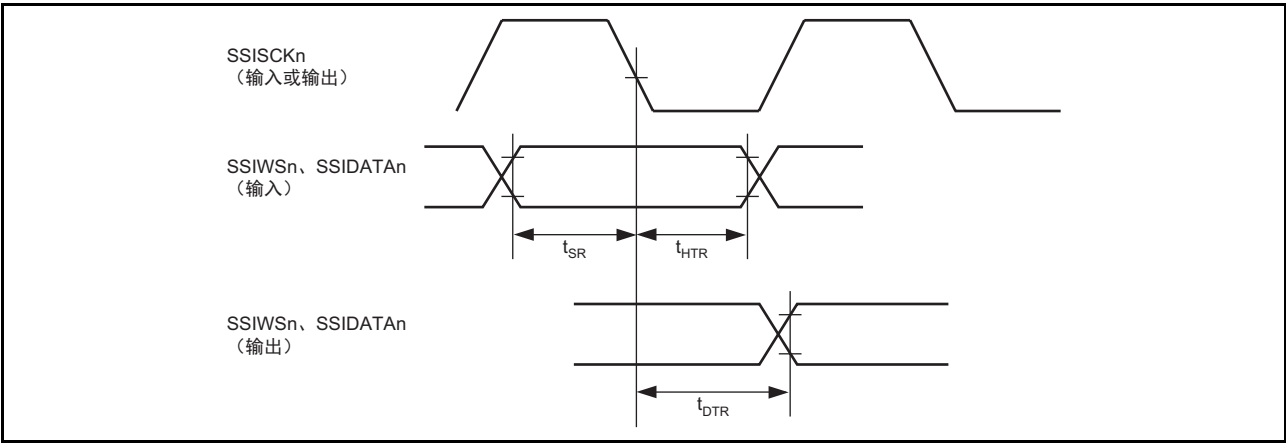


图 31.60 SSI 发送 / 接收时序 (SSISCKn 下降沿同步)

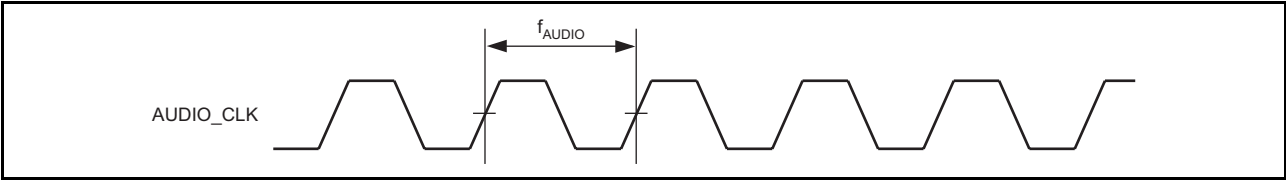


图 31.61 AUDIO_CLK 输入时序

31.4.12 RCAN-TL1 时序

表 31.17 RCAN-TL1 时序

条件: $V_{CC}=PLL V_{CC}=USBDV_{CC}=1.1 \sim 1.3V$ 、 $PV_{CC}=USBDPV_{CC}=3.0 \sim 3.6V$ 、 $AV_{CC}=3.0 \sim 3.6V$ 、
 $USBAV_{CC}=1.1 \sim 1.3V$ 、 $USBAPV_{CC}=3.0 \sim 3.6V$ 、
 $V_{SS}=PLL V_{SS}=PV_{SS}=AV_{SS}=USBDV_{SS}=USBAV_{SS}=USBDPV_{SS}=USBAPV_{SS}=0V$ 、 $T_a=-20 \sim 85^{\circ}C$

项 目	符号	Min.	Max.	单位	参照图
发送数据延迟时间	t_{CTXD}	—	100	ns	31.62
接收数据准备时间	t_{CRXS}	100	—		
接收数据保持时间	t_{CRXH}	100	—		

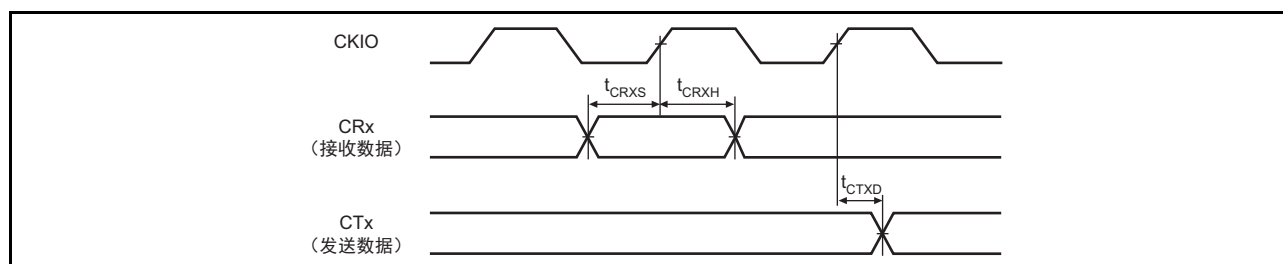


图 31.62 RCAN-TL1 输入 / 输出时序

31.4.13 ADC 时序

表 31.18 ADC 时序

条件: $V_{CC}=PLL V_{CC}=USBDV_{CC}=1.1 \sim 1.3V$ 、 $PV_{CC}=USBDPV_{CC}=3.0 \sim 3.6V$ 、 $AV_{CC}=3.0 \sim 3.6V$ 、
 $USBAV_{CC}=1.1 \sim 1.3V$ 、 $USBAPV_{CC}=3.0 \sim 3.6V$ 、
 $V_{SS}=PLL V_{SS}=PV_{SS}=AV_{SS}=USBDV_{SS}=USBAV_{SS}=USBDPV_{SS}=USBAPV_{SS}=0V$ 、 $T_a=-20 \sim 85^{\circ}C$

模块	项 目	符号	Min.	Max.	单位	参照图
A/D 转换器	触发输入准备时间	B : P 时钟比 = 1 : 1	t_{TRGS}	17	ns	31.63
		B : P 时钟比 = 2 : 1		$t_{cyc}+17$		
		B : P 时钟比 = 4 : 1		$3 \times t_{cyc}+17$		

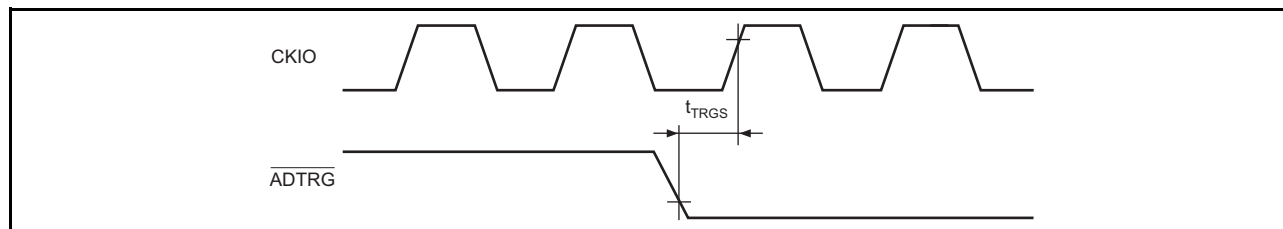


图 31.63 A/D 转换器外部触发输入时序

31.4.14 FLCTL 时序

表 31.19 AND 型闪存接口时序

条件: $V_{CC}=PLL V_{CC}=USBDV_{CC}=1.1 \sim 1.3V$ 、 $PV_{CC}=USBDPV_{CC}=3.0 \sim 3.6V$ 、 $AV_{CC}=3.0 \sim 3.6V$ 、
 $USBAV_{CC}=1.1 \sim 1.3V$ 、 $USBAPV_{CC}=3.0 \sim 3.6V$ 、
 $V_{SS}=PLL V_{SS}=PV_{SS}=AV_{SS}=USBDV_{SS}=USBAV_{SS}=USBDPV_{SS}=USBAPV_{SS}=0V$ 、 $T_a=-20 \sim 85^{\circ}C$

项 目	符号	Min.	Max.	单位	参照图
命令发行准备时间	t_{ACDS}	$2 \times t_{fcyc} - 10$	—	ns	31.64、31.68
命令发行保持时间	t_{ACDH}	$2 \times t_{fcyc} - 10$	—	ns	
数据输出准备时间	t_{ADOS}	$t_{fcyc} - 10$	—	ns	31.64、31.65、31.68
数据输出保持时间	t_{ADOH}	$t_{fcyc} - 10$	—	ns	
数据输出准备时间 2	t_{ADOS2}	$0.5 \times t_{fcyc} - 10$	—	ns	31.67
数据输出保持时间 2	t_{ADOH2}	$0.5 \times t_{fcyc} - 10$	—	ns	
$\overline{FWE}$ 周期时间	t_{ACWC}	$2 \times t_{fcyc} - 5$	—	ns	31.65
$\overline{FWE}$ 低电平脉宽	t_{AWP}	$t_{fcyc} - 5$	—	ns	31.64、31.65、31.68
$\overline{FWE}$ 高电平脉宽	t_{AWPH}	$t_{fcyc} - 5$	—	ns	
命令 - 地址转移时间	t_{ACAS}	$4 \times t_{fcyc}$	—	ns	31.66
地址 - 数据读取转移时间	t_{AADDR}	$32 \times t_{pcyc}$	—	ns	
地址 - 就序 / 忙转移时间	t_{AADRB}	—	$35 \times t_{pcyc}$	ns	
就序 / 忙 - 数据读取转移时间	t_{ARBDR}	$3 \times t_{fcyc}$	—	ns	31.66
数据读取准备时间	t_{ADRS}	$t_{fcyc} - 10$	—	ns	
FSC 周期时间	t_{ASCC}	$t_{fcyc} - 5$	—	ns	31.66、31.67
FSC 高电平脉宽	t_{ASP}	$0.5 \times t_{fcyc} - 5$	—	ns	
FSC 低电平脉宽	t_{ASPL}	$0.5 \times t_{fcyc} - 5$	—	ns	
读取数据准备时间	t_{ARDS}	24	—	ns	31.66、31.68
读取数据保持时间	t_{ARDH}	5	—	ns	
状态读取数据准备时间	t_{ASRDS}	$2 \times t_{fcyc} + 24$	—	ns	31.68
地址数据写入转移时间	t_{AADDW}	$4 \times t_{pcyc}$	—	ns	31.67
数据写入准备时间	t_{ADWS}	$50 \times t_{pcyc}$	—	ns	
FSC-FOE 保持时间	t_{ASOH}	$2 \times t_{fcyc} - 10$	—	ns	31.66

【注】 t_{fcyc} 表示 FLCTL 时钟的 1 个周期时间。

t_{pcyc} 表示外围时钟 ($P\phi$) 的 1 个周期时间。

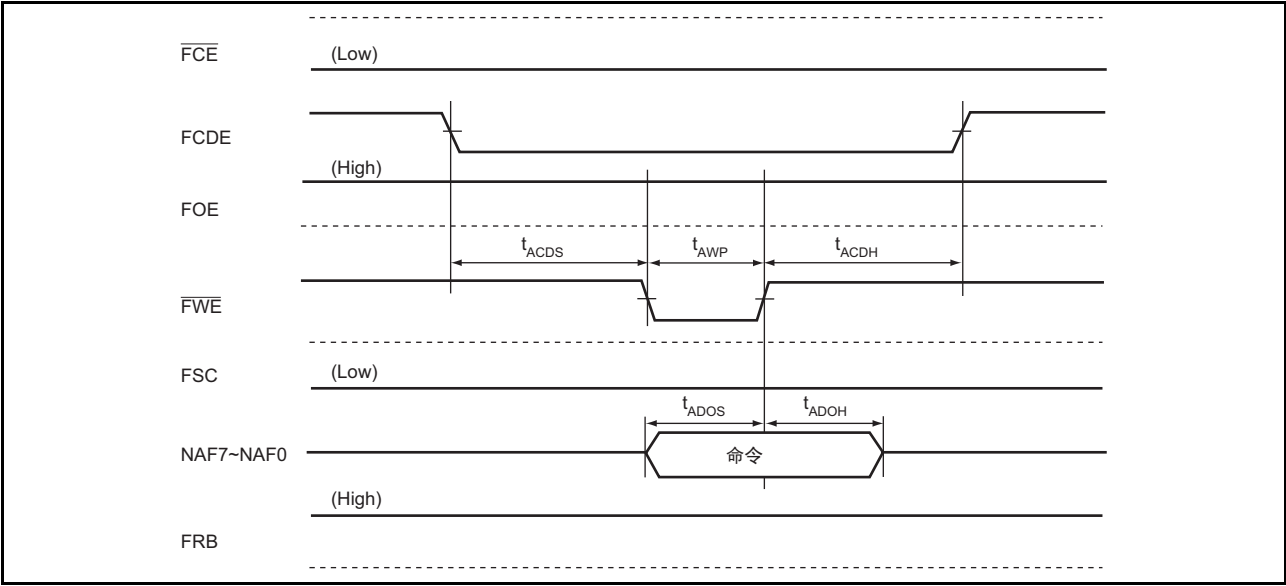


图 31.64 AND 型闪存命令发行时序

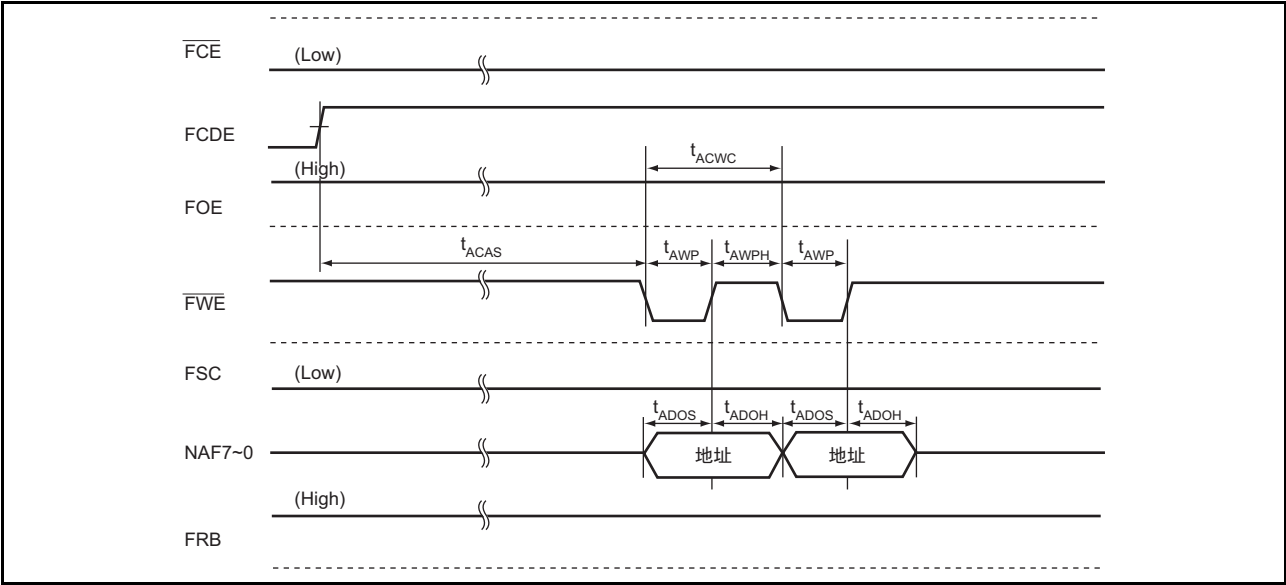


图 31.65 AND 型闪存地址发行时序

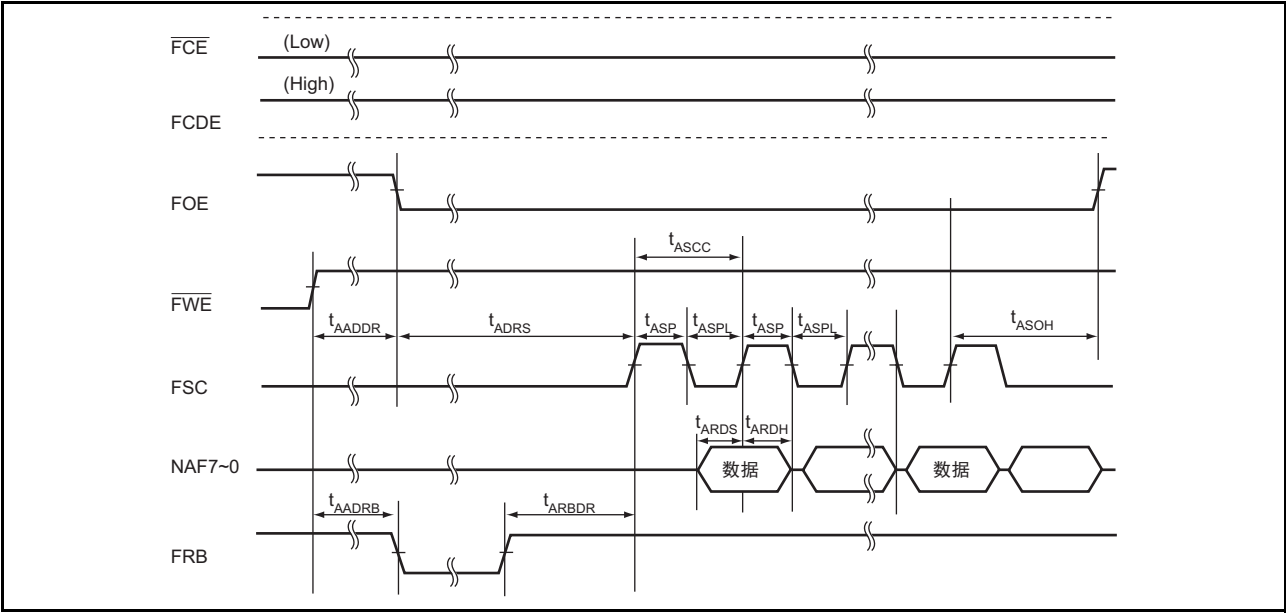


图 31.66 AND 型闪存数据读取时序

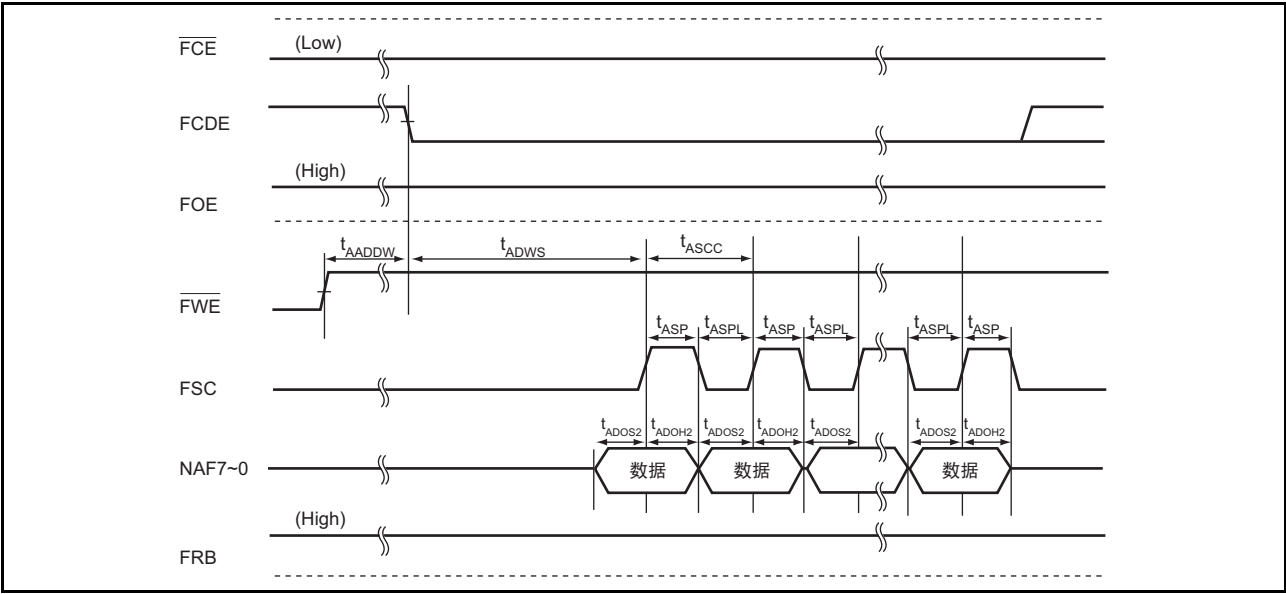


图 31.67 AND 型闪存数据写入时序

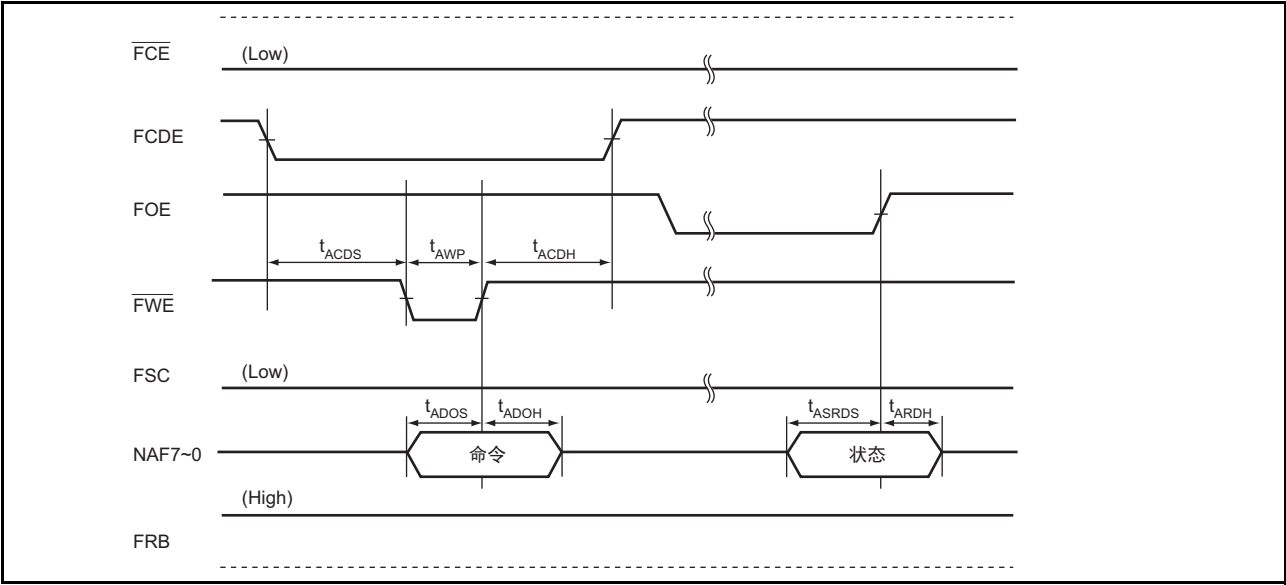


图 31.68 AND 型闪存状态读取时序

表 31.20 NAND 型闪存接口时序

条件: $V_{CC}=PLL V_{CC}=USBDV_{CC}=1.1 \sim 1.3V$ 、 $PV_{CC}=USBDPV_{CC}=3.0 \sim 3.6V$ 、 $AV_{CC}=3.0 \sim 3.6V$ 、
 $USBAV_{CC}=1.1 \sim 1.3V$ 、 $USBAPV_{CC}=3.0 \sim 3.6V$ 、
 $V_{SS}=PLL V_{SS}=PV_{SS}=AV_{SS}=USBDV_{SS}=USBAV_{SS}=USBDPV_{SS}=USBAPV_{SS}=0V$ 、 $T_a=-20 \sim 85^{\circ}C$

项 目	符号	Min.	Max.	单位	参照图
命令输出准备时间	t_{NCDS}	$2 \times t_{fcyc} - 10$	—	ns	31.69、31.73
命令输出保持时间	t_{NCDH}	$1.5 \times t_{fcyc} - 5$	—	ns	
数据输出准备时间	t_{NDOS}	$0.5 \times t_{wfcyc} - 5$	—	ns	31.69、31.70、 31.72、31.73
数据输出保持时间	t_{NDOH}	$0.5 \times t_{wfcyc} - 10$	—	ns	
命令 - 地址转移时间 1	t_{NCDAD1}	$1.5 \times t_{fcyc} - 10$	—	ns	31.69、31.70
命令 - 地址转移时间 2	t_{NCDAD2}	$2 \times t_{fcyc} - 10$	—	ns	31.70
$\overline{FWE}$ 周期时间	t_{NWC}	$t_{wfcyc} - 5$	—	ns	31.70、31.72
$\overline{FWE}$ 低电平脉宽	t_{NWP}	$0.5 \times t_{wfcyc} - 5$	—	ns	31.69、31.70、 31.72、31.73
$\overline{FWE}$ 高电平脉宽	t_{NWH}	$0.5 \times t_{wfcyc} - 5$	—	ns	31.70、31.72
地址 - 就序 / 忙转移时间	t_{NADRB}	—	$32 \times t_{pcyc}$	ns	31.70、31.71
命令 - 就序 / 忙转移时间	t_{NCDRB}	—	$10 \times t_{pcyc}$	ns	31.70、31.71
就序 / 忙 - 数据读取转移时间 1	t_{NRBDR1}	$1.5 \times t_{fcyc}$	—	ns	31.71
就序 / 忙 - 数据读取转移时间 2	t_{NRBDR2}	$32 \times t_{pcyc}$	—	ns	
FSC 周期时间	t_{NSCC}	$t_{wfcyc} - 5$	—	ns	
FSC 低电平脉宽	t_{NSP}	$0.5 \times t_{wfcyc} - 5$	—	ns	31.71、31.73
FSC 高电平脉宽	t_{NSPH}	$0.5 \times t_{wfcyc} - 5$	—	ns	31.71
读取数据准备时间	t_{NRDS}	24	—	ns	31.71、31.73
读取数据保持时间	t_{NRDH}	5	—	ns	31.71、31.73
数据写入准备时间	t_{NDWS}	$32 \times t_{pcyc}$	—	ns	31.72
命令 - 状态读取转移时间	t_{NCDSR}	$4 \times t_{fcyc}$	—	ns	31.73
命令输出关闭 - 状态读取转移时间	t_{NCDFSR}	$3.5 \times t_{fcyc}$	—	ns	
状态读取准备时间	t_{NSTS}	$2.5 \times t_{fcyc}$	—	ns	

【注】 t_{fcyc} 为 FLCTL 时钟的 1 个周期时间。

NANDWF 位为 0 时, t_{wfcyc} 为 FLCTL 时钟的 1 个周期时间; NANDWF 位为 1 时, t_{wfcyc} 为 FLCTL 时钟的 2 个周期时间。

t_{pcyc} 表示外围时钟 ($P\phi$) 的 1 个周期时间。

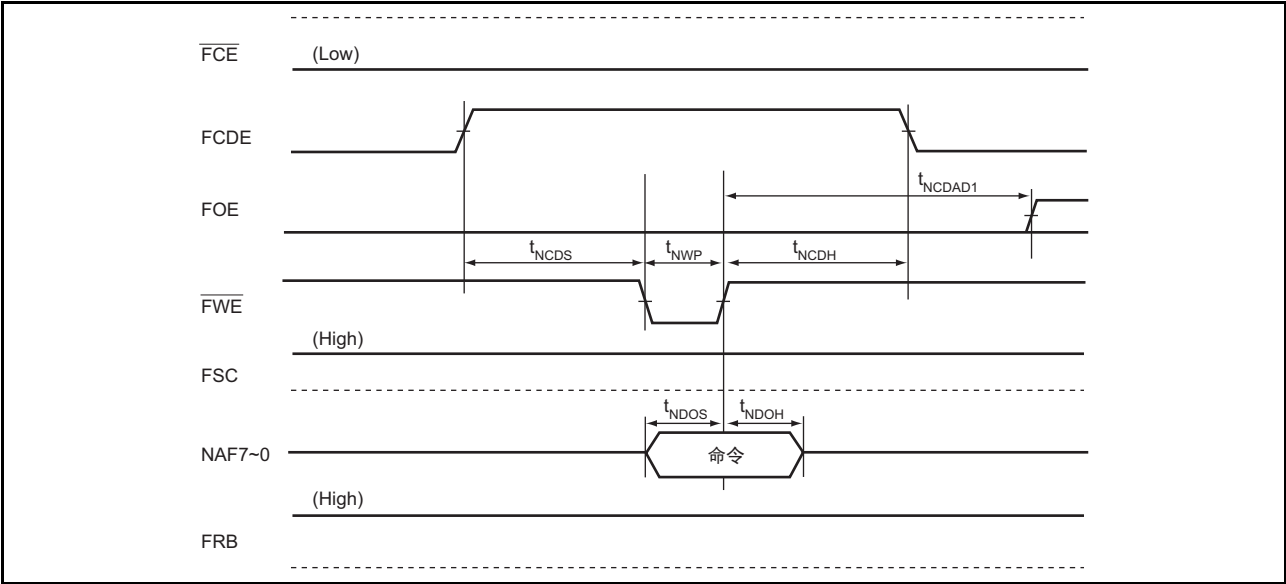


图 31.69 NAND 型闪存命令发行时序

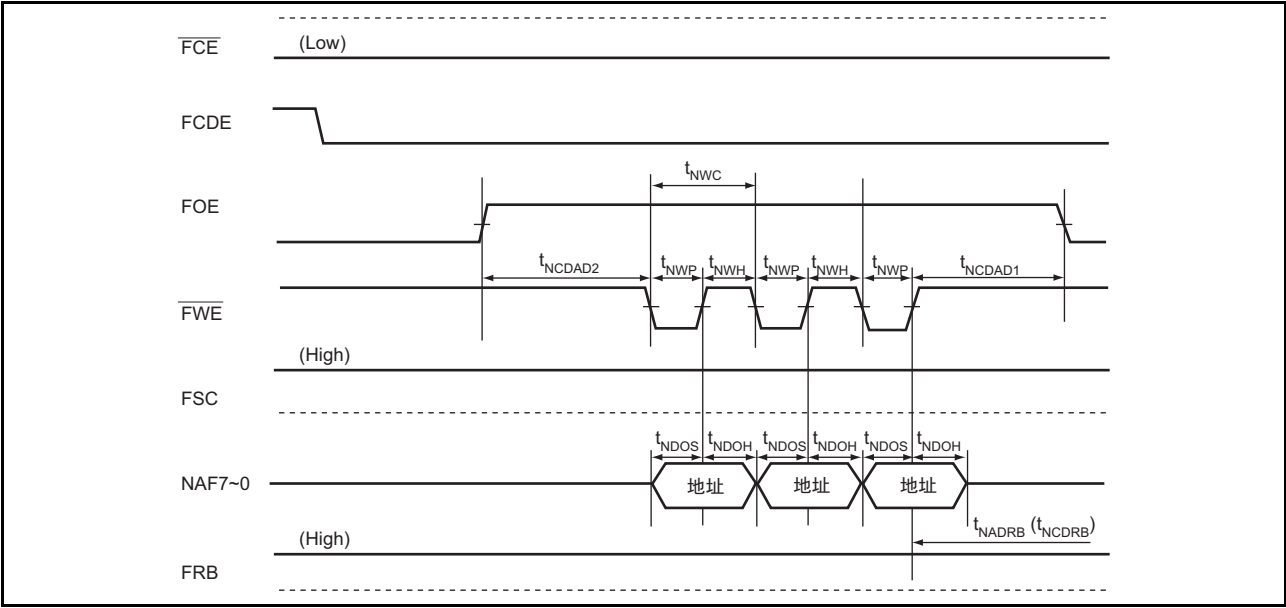


图 31.70 NAND 型闪存地址发行时序

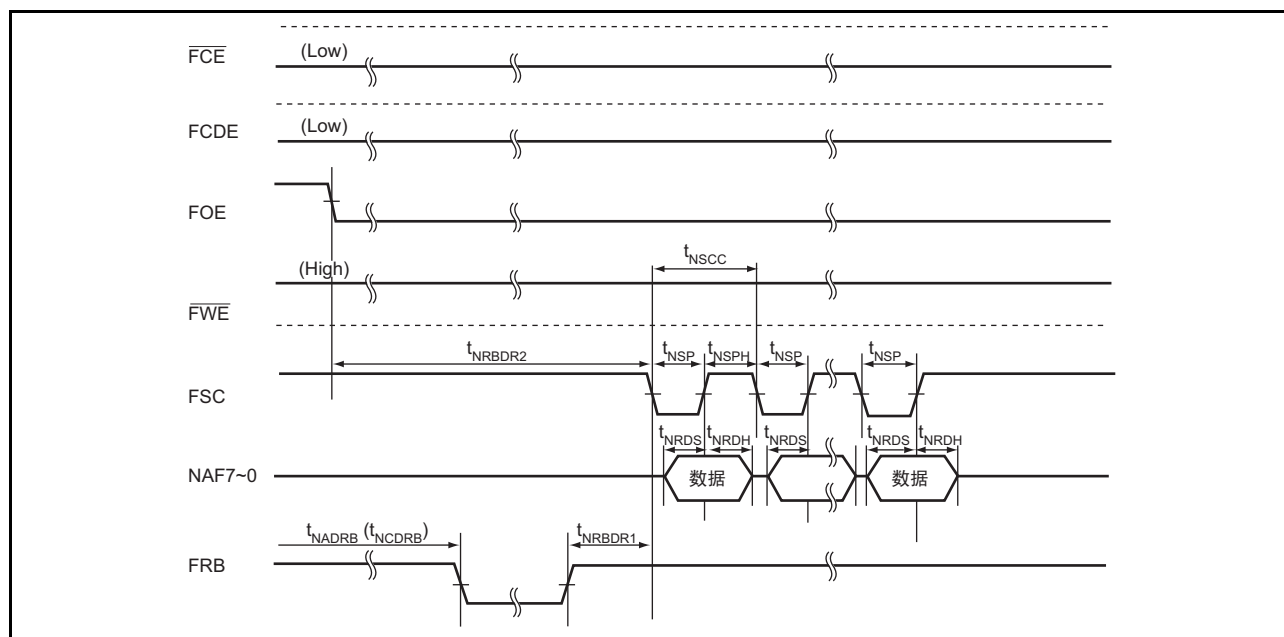


图 31.71 NAND 型闪存数据读取时序

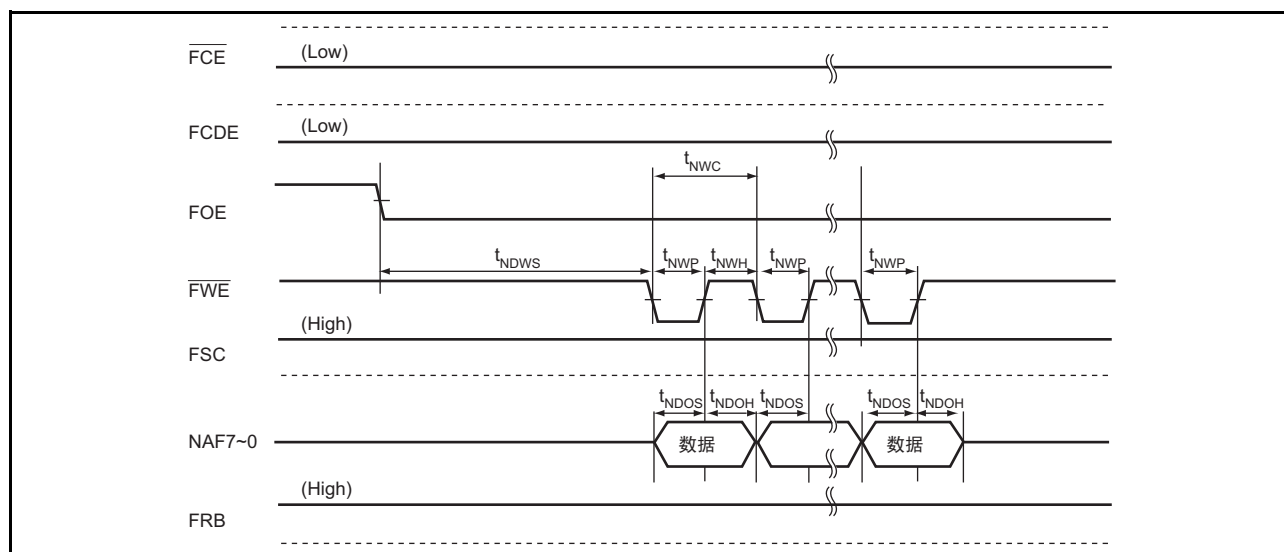


图 31.72 NAND 型闪存数据写入时序

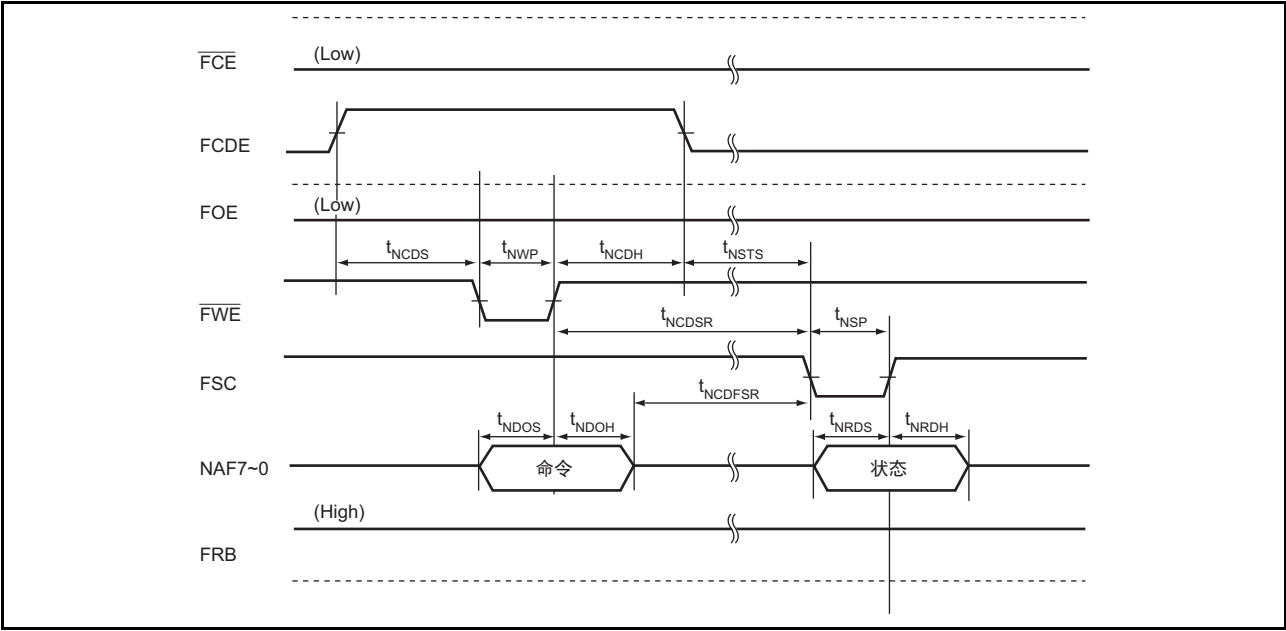


图 31.73 NAND 型闪存状态读取时序

31.4.15 USB 时序

表 31.21 USB 收发器时序（全速时）

条件: $V_{CC}=PLL V_{CC}=USBDV_{CC}=1.1 \sim 1.3V$ 、 $PV_{CC}=USBDPV_{CC}=3.0 \sim 3.6V$ 、 $AV_{CC}=3.0 \sim 3.6V$ 、
 $USBAV_{CC}=1.1 \sim 1.3V$ 、 $USBAPV_{CC}=3.0 \sim 3.6V$ 、
 $V_{SS}=PLL V_{SS}=PV_{SS}=AV_{SS}=USBDV_{SS}=USBAV_{SS}=USBDPV_{SS}=USBAPV_{SS}=0V$ 、 $T_a=-20 \sim 85^{\circ}C$

项 目	符号	Min.	Typ.	Max.	单位	参照图
上升时间	t_{FR}	4	—	20	ns	31.74
下降时间	t_{FF}	4	—	20	ns	
上升 / 下降时间比	t_{FR} / t_{FF}	90	—	111.11	%	

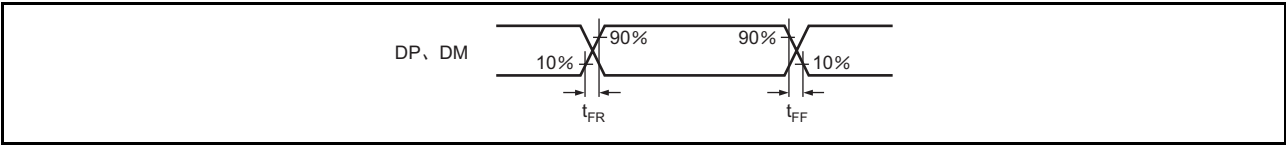


图 31.74 DP、DM 输出时序（全速时）

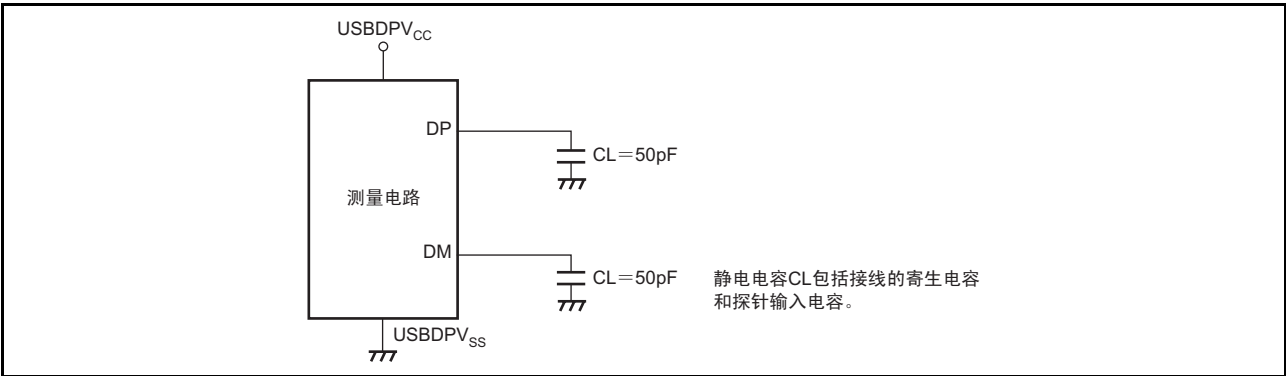


图 31.75 测量电路（全速时）

表 31.22 USB 收发器时序（高速时）

条件: $V_{CC}=PLL_{V_{CC}}=USB_{DV_{CC}}=1.1 \sim 1.3V$ 、 $PV_{CC}=USB_{DPV_{CC}}=3.0 \sim 3.6V$ 、 $AV_{CC}=3.0 \sim 3.6V$ 、
 $USB_{AV_{CC}}=1.1 \sim 1.3V$ 、 $USB_{APV_{CC}}=3.0 \sim 3.6V$ 、
 $V_{SS}=PLL_{V_{SS}}=PV_{SS}=AV_{SS}=USB_{DV_{SS}}=USB_{AV_{SS}}=USB_{DPV_{SS}}=USB_{APV_{SS}}=0V$ 、 $T_a = -20 \sim 85^{\circ}C$

项 目	符号	Min.	Typ.	Max.	单位	参照图
上升时间	t_{HSR}	500	—	—	ps	31.76
下降时间	t_{HSF}	500	—	—	ps	
输出驱动电阻	Z_{HSDRV}	40.5	—	49.5	Ω	

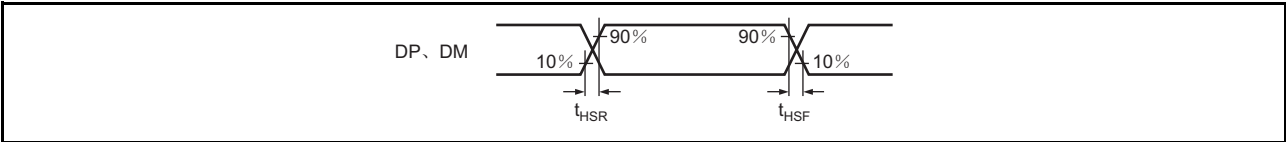


图 31.76 DP、DM 输出时序（高速时）

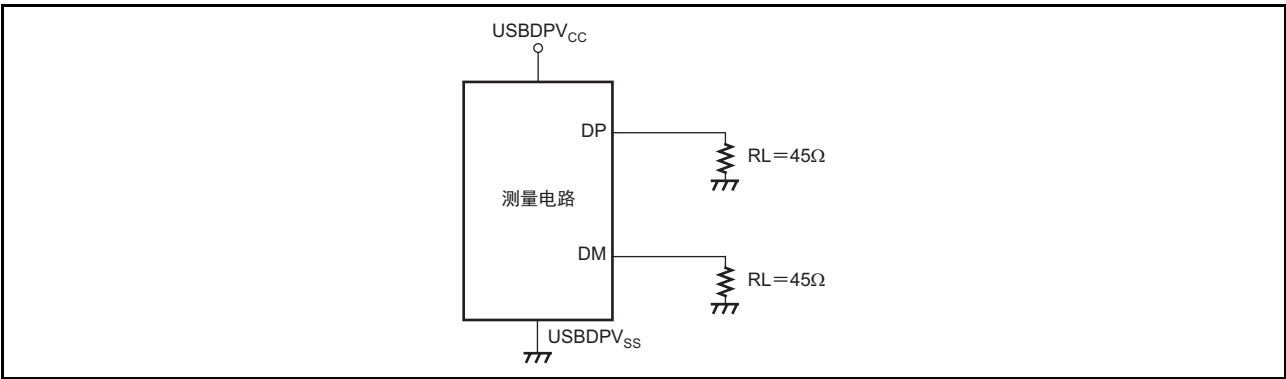


图 31.77 测量电路（高速时）

31.4.16 LCDC 时序

表 31.23 LCDC 时序

条件: $V_{CC}=PLL_{V_{CC}}=USBDV_{CC}=1.1 \sim 1.3V$ 、 $PV_{CC}=USBDPV_{CC}=3.0 \sim 3.6V$ 、 $AV_{CC}=3.0 \sim 3.6V$ 、
 $USBAV_{CC}=1.1 \sim 1.3V$ 、 $USBAPV_{CC}=3.0 \sim 3.6V$ 、
 $V_{SS}=PLL_{V_{SS}}=PV_{SS}=AV_{SS}=USBDV_{SS}=USBAV_{SS}=USBDPV_{SS}=USBAPV_{SS}=0V$ 、 $T_a=-20 \sim 85^{\circ}C$

项 目	符号	Min.	Max.	单位	参照图
LCD_CLK 输入时钟频率	t_{FREQ}	—	66.66	MHz	
LCD_CLK 输入时钟上升时间	t_r	—	3	ns	
LCD_CLK 输入时钟下降时间	t_f	—	3	ns	
LCD_CLK 输入时钟占空比	t_{DUTY}	90	110	%	
时钟 (LCD_CL2) 周期时间	t_{CC}	25	—	ns	31.78
时钟 (LCD_CL2) 高电平脉宽	t_{CHW}	7	—	ns	
时钟 (LCD_CL2) 低电平脉宽	t_{CLW}	7	—	ns	
时钟 (LCD_CL2) 转移时间 (上升 / 下降)	t_{CT}	—	3	ns	
数据 (LCD_DATA) 转移时间	t_{DD}	-3.5	3	ns	
显示允许 (LCD_M_DISP) 转移时间	t_{ID}	-3.5	3	ns	
水平同步信号 (LCD_CL1) 延迟时间	t_{HD}	-3.5	3	ns	
垂直同步信号 (LCD_FLM) 延迟时间	t_{VD}	-3.5	3	ns	

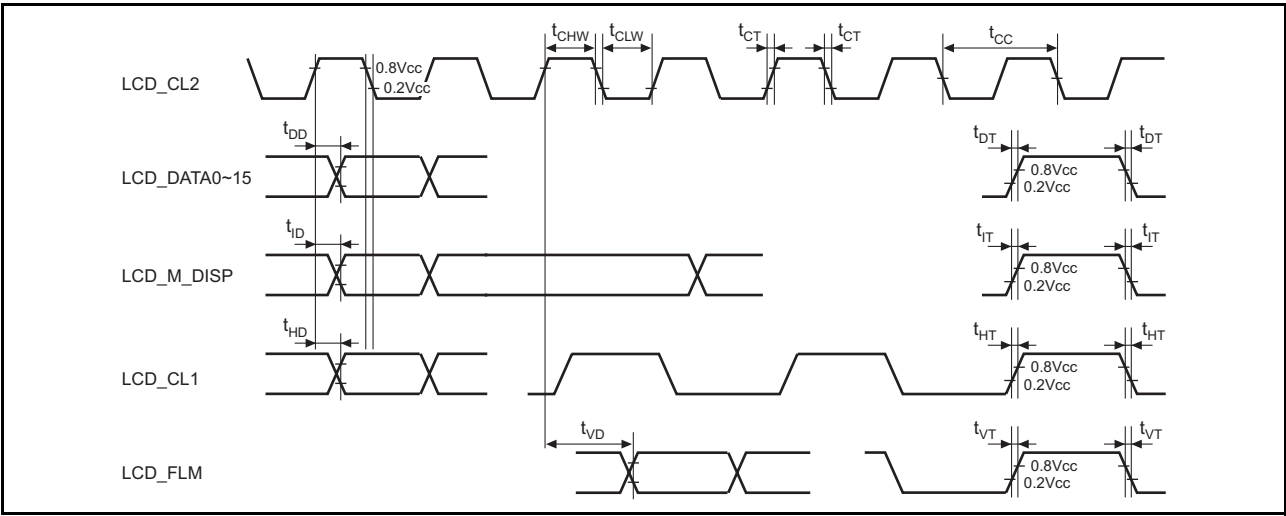


图 31.78 LCDC 模块时序

31.4.17 I/O 端口时序

表 31.24 I/O 端口时序

条件: $V_{CC}=PLL V_{CC}=USBDV_{CC}=1.1 \sim 1.3V$ 、 $PV_{CC}=USBDPV_{CC}=3.0 \sim 3.6V$ 、 $AV_{CC}=3.0 \sim 3.6V$ 、
 $USBAV_{CC}=1.1 \sim 1.3V$ 、 $USBAPV_{CC}=3.0 \sim 3.6V$ 、
 $V_{SS}=PLL V_{SS}=PV_{SS}=AV_{SS}=USBDV_{SS}=USBAV_{SS}=USBDPV_{SS}=USBAPV_{SS}=0V$ 、 $T_a=-20 \sim 85^{\circ}C$

项 目	符号	Min.	Max.	单位	参照图
输出数据延迟时间	t_{PORTD}	—	100	ns	31.79
输入数据准备时间	t_{PORTS}	100	—		
输入数据保持时间	t_{PORTH}	100	—		

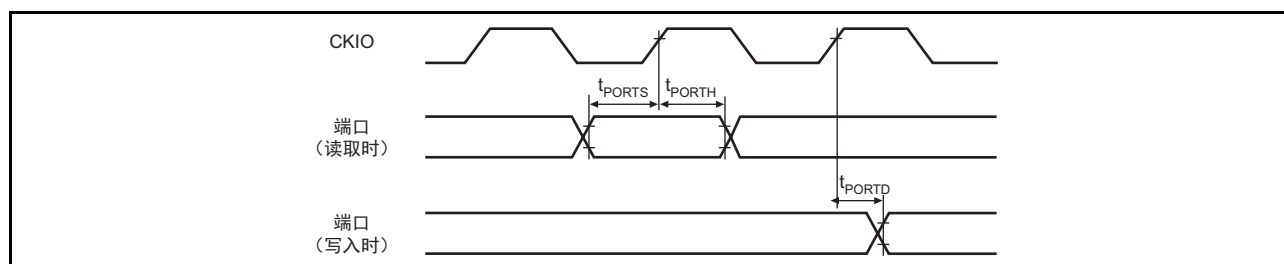


图 31.79 I/O 端口时序

31.4.18 H-UDI 时序

表 31.25 H-UDI 时序

条件: $V_{CC}=PLL V_{CC}=USBDV_{CC}=1.1 \sim 1.3V$ 、 $PV_{CC}=USBDPV_{CC}=3.0 \sim 3.6V$ 、 $AV_{CC}=3.0 \sim 3.6V$ 、
 $USBAV_{CC}=1.1 \sim 1.3V$ 、 $USBAPV_{CC}=3.0 \sim 3.6V$ 、
 $V_{SS}=PLL V_{SS}=PV_{SS}=AV_{SS}=USBDV_{SS}=USBAV_{SS}=USBDPV_{SS}=USBAPV_{SS}=0V$ 、 $T_a=-20 \sim 85^{\circ}C$

项 目	符号	Min.	Max.	单位	参照图
TCK 周期时间	t_{TCKcyc}	50*	—	ns	31.80
TCK 高电平脉宽	t_{TCKH}	0.4	0.6	t_{TCKcyc}	
TCK 低电平脉宽	t_{TCKL}	0.4	0.6	t_{TCKcyc}	
TDI 准备时间	t_{TDIS}	10	—	ns	31.81
TDI 保持时间	t_{TDIH}	10	—	ns	
TMS 准备时间	t_{TMSS}	10	—	ns	
TMS 保持时间	t_{TMSH}	10	—	ns	
TDO 延迟时间	t_{TDOD}	—	16	ns	

【注】 * 必须设置为大于外围时钟 ($P\phi$) 周期时间。

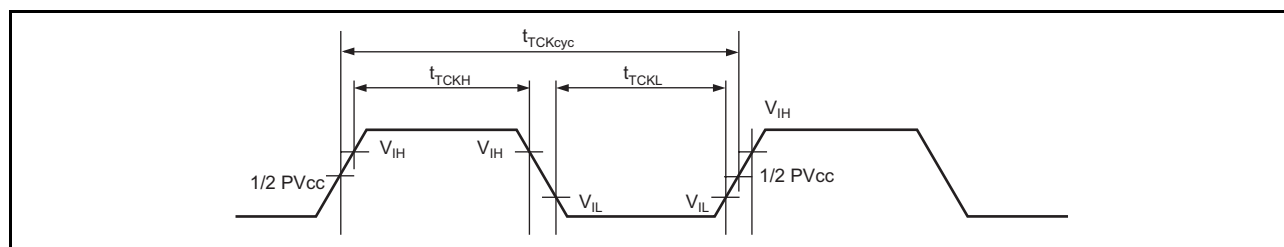


图 31.80 TCK 输入时序

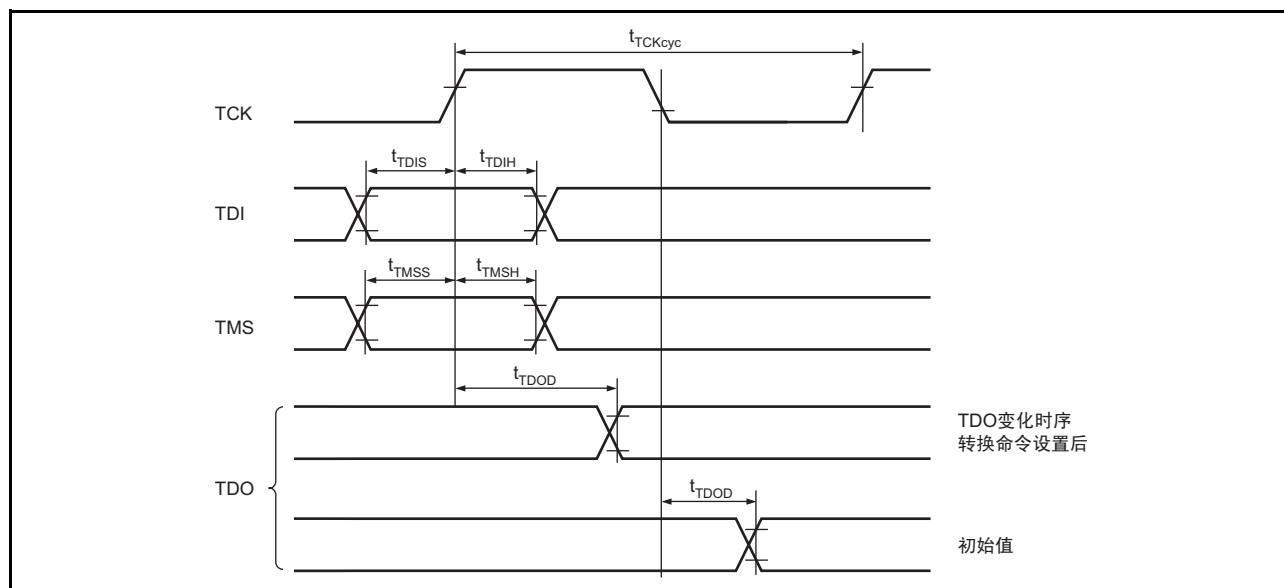


图 31.81 H-UDI 数据传送时序

31.4.19 AC 特性测量条件

- 输入/输出信号参考电平: $PV_{CC}/2$ ($PV_{CC}=3.0\sim 3.6V$ 、 $V_{CC}=1.1\sim 1.3V$)
- 输入脉冲电平: $PV_{SS}\sim 3.0V$ (但 $\overline{RES}$ 、 $\overline{MRES}$ 、 NMI 、 MD 、 MD_CLK1 、 MD_CLK0 、 $\overline{ASEMD}$ 、 $\overline{TRST}$ 及施密特触发器输入引脚为 $PV_{SS}\sim PV_{CC}$)
- 上升、下降时间: 1ns

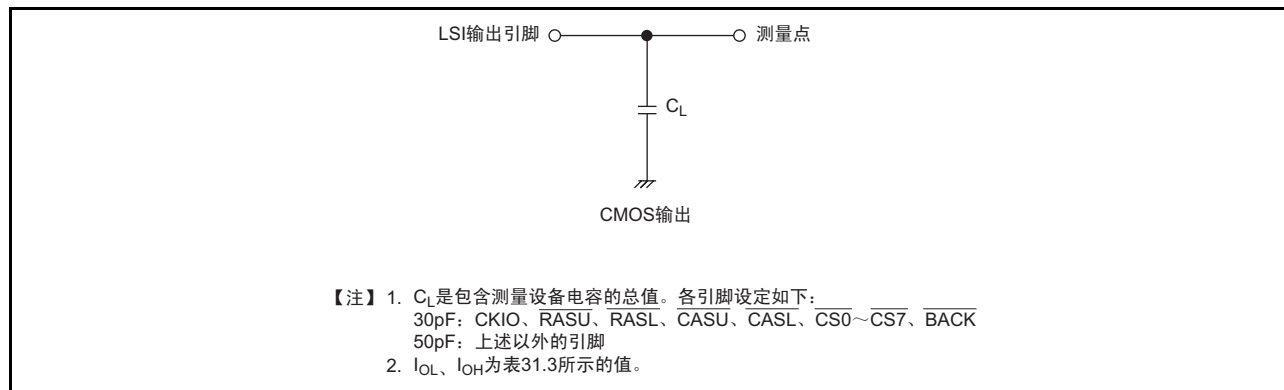


图 31.82 输出补充电路

31.5 A/D 转换器特性

表 31.26 A/D 转换器特性

条件: $V_{CC}=PLL V_{CC}=USBDV_{CC}=1.1 \sim 1.3V$ 、 $PV_{CC}=USBDPV_{CC}=3.0 \sim 3.6V$ 、 $AV_{CC}=3.0 \sim 3.6V$ 、
 $USBAV_{CC}=1.1 \sim 1.3V$ 、 $USBAPV_{CC}=3.0 \sim 3.6V$ 、
 $V_{SS}=PLL V_{SS}=PV_{SS}=AV_{SS}=USBDV_{SS}=USBAV_{SS}=USBDPV_{SS}=USBAPV_{SS}=0V$ 、 $T_a=-20 \sim 85^{\circ}C$

项 目	Min.	Typ.	Max.	单位
分辨率	10	10	10	位
转换时间	3.9	—	—	μs
模拟输入容量	—	—	20	pF
容许信号源阻抗	—	—	5	k Ω
非线性误差	—	—	$\pm 3.0^*$	LSB
偏移误差	—	—	$\pm 2.0^*$	LSB
满刻度误差	—	—	$\pm 2.0^*$	LSB
量化误差	—	—	$\pm 0.5^*$	LSB
绝对精度	—	—	± 4.0	LSB

【注】 * 参考值

31.6 D/A 转换器特性

表 31.27 D/A 转换器特性

条件: $V_{CC}=PLL V_{CC}=USBDV_{CC}=1.1 \sim 1.3V$ 、 $PV_{CC}=USBDPV_{CC}=3.0 \sim 3.6V$ 、 $AV_{CC}=3.0 \sim 3.6V$ 、
 $USBAV_{CC}=1.1 \sim 1.3V$ 、 $USBAPV_{CC}=3.0 \sim 3.6V$ 、
 $V_{SS}=PLL V_{SS}=PV_{SS}=AV_{SS}=USBDV_{SS}=USBAV_{SS}=USBDPV_{SS}=USBAPV_{SS}=0V$ 、 $T_a=-20 \sim 85^{\circ}C$

项 目	Min.	Typ.	Max.	单位	测量条件
分辨率	8	8	8	位	
转换时间	10	—	—	μs	负载电容 20pF
绝对精度	—	± 2.0	± 3.0	LSB	负载电阻 2M Ω
	—	—	± 2.5	LSB	负载电阻 4M Ω

31.7 使用时的注意事项

必须将多层陶瓷电容作为旁路电容装入每对电源引脚。旁路电容尽可能安装在 LSI 电源引脚附近。电容值必须为使用 $0.1\mu\text{F} \sim 0.33\mu\text{F}$ （推荐值）。与晶体振荡相关的电容，详情参阅“4.6 使用时的注意事项”。

外接电容配置例，如图 31.83 所示。

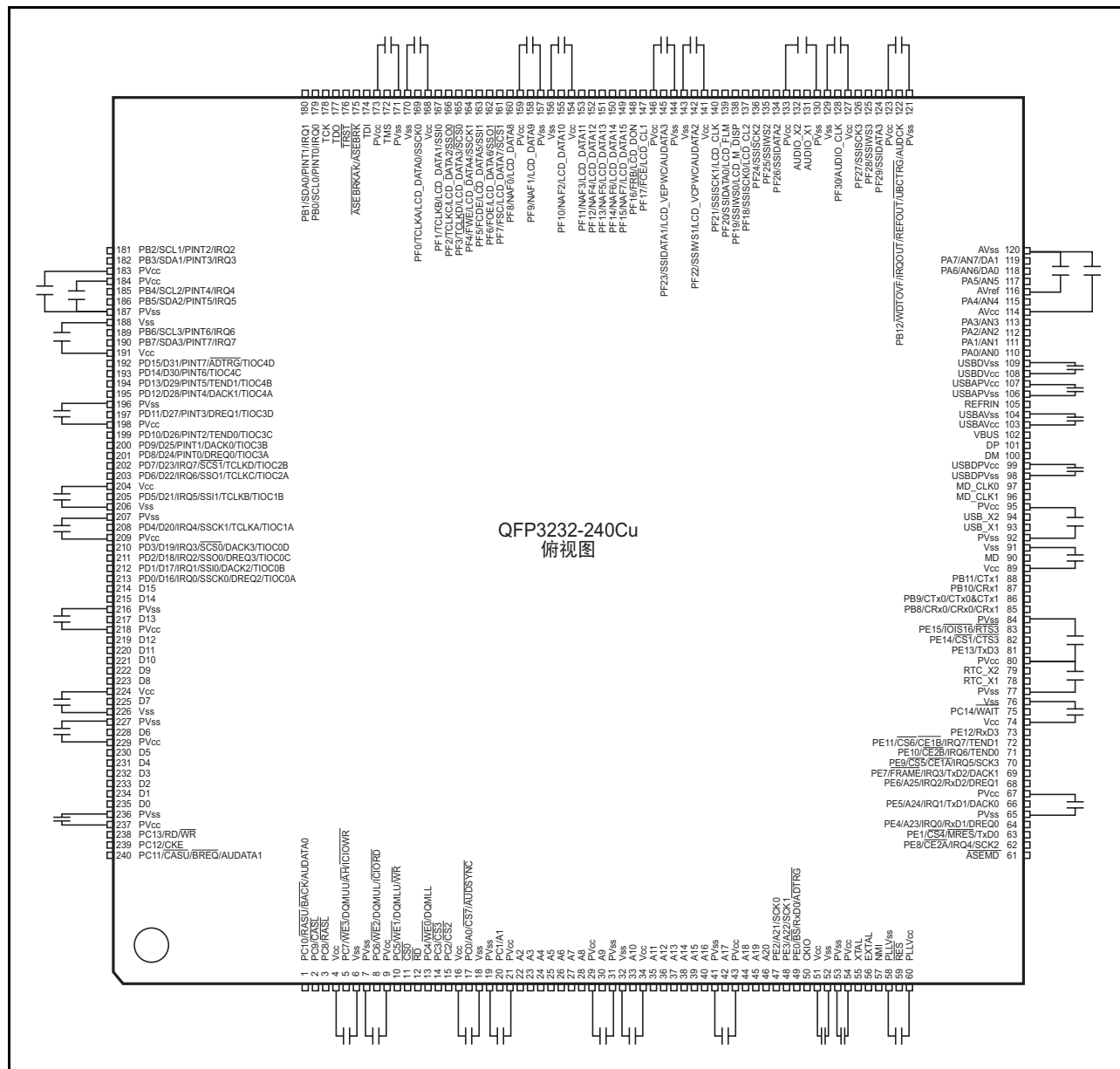


图 31.83 外接电容配置例

附录

附录 A. 引脚状态

表 A.1 引脚状态

引脚功能				引脚状态					
分类	引脚名称			普通状态 (右边情况除外)	复位状态		低功耗状态		总线权 释放状态
					上电复位 *1	保持引脚 状态 *2	深度待机 模式 *3	软件待机 模式	
时钟	EXTAL*4	时钟运行 模式	0、 1	I	I	I	Z	I	I
			2、 3	Z	Z	Z	Z	Z	Z
	XTAL*4			O	O	O	L	L	O
	CKIO	时钟运行 模式	0、 1、 3	O/Z*6	O	O/Z*6*13	O/Z*6	O/Z*6	O/Z*6
			2	I	I	I	Z	I	I
系统控制	RES			I	I	I	I	I	I
	MRES			I	—	I/Z*10	I/Z*10	I	I
	WDTOVF			O	H	H	H	H	O
	BREQ			I	—	Z	Z	Z	I
	BACK			O	—	Z	Z	Z	L
运行模式控制	MD			I	I	I	I	I	I
	MD_CLK1、 MD_CLK0			I	I	I	I	I	I
	ASEMD			I	I	I	I	I	I
中断	NMI			I	I	I	I	I	I
	IRQ7 ~ IRQ0 (PB7 ~ PB0)			I	—	I	I	I	I
	IRQ7 ~ IRQ0 (PD7 ~ PD0)			I	—	Z	Z	I	I
	IRQ7 ~ IRQ0 (PE11 ~ PE4)			I	—	I/Z*10	I/Z*10	I	I
	PINT7 ~ PINT0 (PB7 ~ PB0)			I	—	I	I	I	I
	PINT7 ~ PINT0 (PD7 ~ PD0)			I	—	Z	Z	Z	I
	IRQOUT			O	—	H/Z*7	H/Z*7	H/Z*7	O
UBC	UBCTRG			O	—	O/Z*7	O/Z*7	O/Z*7	O
地址总线	A25 ~ A21、 A0			O	—	O/Z*8	O/Z*8	O/Z*8	Z
	A20 ~ A2			O	O	O/Z*8*13	O/Z*8	O/Z*8	Z
	A1			O	O*5	O/Z*8*13	O/Z*8	O/Z*8	Z

引脚功能		引脚状态					
分类	引脚名称	普通状态 (右边情况除外)	复位状态		低功耗状态		总线权释放状态
			上电复位 *1	保持引脚 状态 *2	深度待机 模式 *3	软件待 机模式	
数据总线	D31 ~ D16	I/O/Z	Z*5	Z	Z	Z	Z
	D15 ~ D0	I/O/Z	Z	Z	Z	Z	Z
总线控制	$\overline{\text{CS0}}$	O	H	H/Z*8*13	H/Z*8	H/Z*8	Z
	$\overline{\text{CS7}} \sim \overline{\text{CS1}}$ 、 $\overline{\text{CE1A}}$ 、 $\overline{\text{CE1B}}$ 、 $\overline{\text{CE2A}}$ 、 $\overline{\text{CE2B}}$	O	—	H/Z*8	H/Z*8	H/Z*8	Z
	$\overline{\text{RD}}$	O	H	H/Z*8*13	H/Z*8	H/Z*8	Z
	$\overline{\text{RD}}/\overline{\text{WR}}$	O	—	H/Z*8	H/Z*8	H/Z*8	Z
	$\overline{\text{BS}}$	O	—	H/Z*8	H/Z*8	H/Z*8	Z
	$\overline{\text{FRAME}}$	O	—	H/Z*8	H/Z*8	H/Z*8	Z
	$\overline{\text{WAIT}}$	I	—	Z	Z	Z	Z
	$\overline{\text{WE3/DQMUU/ICIOWR/AH}}$ 、 $\overline{\text{WE2/DQMUL/ICIORD}}$ 、 $\overline{\text{WE1/DQMLU/WE}}$ 、 $\overline{\text{WE0/DQMLL}}$	O	—	H/Z*8	H/Z*8	H/Z*8	Z
	$\overline{\text{RASU}}$ 、 $\overline{\text{RASL}}$ 、 $\overline{\text{CASU}}$ 、 $\overline{\text{CASL}}$	O	—	O/Z*9	O/Z*9	O/Z*9	O/Z*9
	$\overline{\text{CKE}}$	O	—	O/Z*9	O/Z*9	O/Z*9	O/Z*9
	$\overline{\text{IOIS16}}$	I	—	Z	Z	Z	I
	$\overline{\text{REFOUT}}$	O	—	H/Z*7	H/Z*7	H/Z*7	O
DMAC	DREQ3 ~ DREQ0	I	—	Z	Z	Z	I
	DACK3 ~ DACK0	O	—	O/Z*7	O/Z*7	O/Z*7	O
	TEND1、TEND0	O	—	O/Z*7	O/Z*7	O/Z*7	O
MTU2	TCLKA、TCLKB、TCLKC、TCLKD	I	—	Z	Z	Z	I
	TIOC0A、TIOC0B、TIOC0C、TIOC0D	I/O	—	K/Z*7	K/Z*7	K/Z*7	I/O
	TIOC1A、TIOC1B	I/O	—	K/Z*7	K/Z*7	K/Z*7	I/O
	TIOC2A、TIOC2B	I/O	—	K/Z*7	K/Z*7	K/Z*7	I/O
	TIOC3A、TIOC3B、TIOC3C、TIOC3D	I/O	—	K/Z*7	K/Z*7	K/Z*7	I/O
	TIOC4A、TIOC4B、TIOC4C、TIOC4D	I/O	—	K/Z*7	K/Z*7	K/Z*7	I/O
RTC	RTC_X1*4	I/Z*11	I	I	I	I/Z*11	I/Z*11
	RTC_X2*4	O/H*11	O	O	O	O/H*11	O/H*11
SCIF	TxD3 ~ TxD0	O/Z	—	O/Z*7	O/Z*7	O/Z*7	O/Z
	RxD3 ~ RxD0	I	—	Z	Z	Z	I
	SCK3 ~ SCK0	I/O	—	K/Z*7	K/Z*7	K/Z*7	I/O
	$\overline{\text{RTS3}}$	I/O	—	K/Z*7	K/Z*7	K/Z*7	I/O
	$\overline{\text{CTS3}}$	I/O	—	K/Z*7	K/Z*7	K/Z*7	I/O

引脚功能		引脚状态					
分类	引脚名称	普通状态 (右边情况除外)	复位状态		低功耗状态		总线权释放状态
			上电复位 *1	保持引脚 状态 *2	深度待机 模式 *3	软件待机 模式	
SSU	SSO1、SSO0	I/O	—	Z	Z	Z	I/O
	SSI1、SSI0	I/O	—	Z	Z	Z	I/O
	SSCK1、SSCK0	I/O	—	Z	Z	Z	I/O
	$\overline{\text{SCS1}}$ 、 $\overline{\text{SCS0}}$	I/O	—	Z	Z	Z	I/O
IIC3	SCL3 ~ SCL0	I/O	—	I	I	I	I/O
	SDA3 ~ SDA0	I/O	—	I	I	I	I/O
SSI	SSIDATA3 ~ SSIDATA0	I/O	—	K/Z*7	K/Z*7	K/Z*7	I/O
	SSISCK3 ~ SSISCK0	I/O	—	K/Z*7	K/Z*7	K/Z*7	I/O
	SSIWS3 ~ SSIWS0	I/O	—	K/Z*7	K/Z*7	K/Z*7	I/O
	AUDIO_CLK	I	—	Z	Z	Z	I
	AUDIO_X1*4	I/Z*12	I	I	Z	Z	I/Z*12
	AUDIO_X2*4	O/L*12	O	O	L	L	O/L*12
RCAN-TL1	CTx1、CTx0	O	—	O/Z*7	O/Z*7	O/Z*7	O
	CRx1、CRx0	I	—	Z	Z	Z	I
ADC	AN7 ~ AN0	I	—	Z	Z	Z	I
	ADTRG	I	—	Z	Z	Z	I
DAC	DA1、DA0	O	—	Z	Z	O	O
FLCTL	FOE	O	—	O/Z*7	O/Z*7	O/Z*7	O
	FSC	O	—	O/Z*7	O/Z*7	O/Z*7	O
	$\overline{\text{FCE}}$	O	—	O/Z*7	O/Z*7	O/Z*7	O
	FCDE	O	—	O/Z*7	O/Z*7	O/Z*7	O
	FRB	I	—	Z	Z	Z	I
	$\overline{\text{FWE}}$	O	—	O/Z*7	O/Z*7	O/Z*7	O
	NAF7 ~ NAF0	I/O/Z	—	K/Z*7	K/Z*7	K/Z*7	I/O/Z
USB	DP、DM	I/O	I/O	I/O	I/O	I/O	I/O
	VBUS	I	I	I	I	I	I
	REFRIN	I	I	I	I	I	I
	USB_X1*4	I	I	I	Z	Z	I
	USB_X2*4	O	O	O	L	L	O

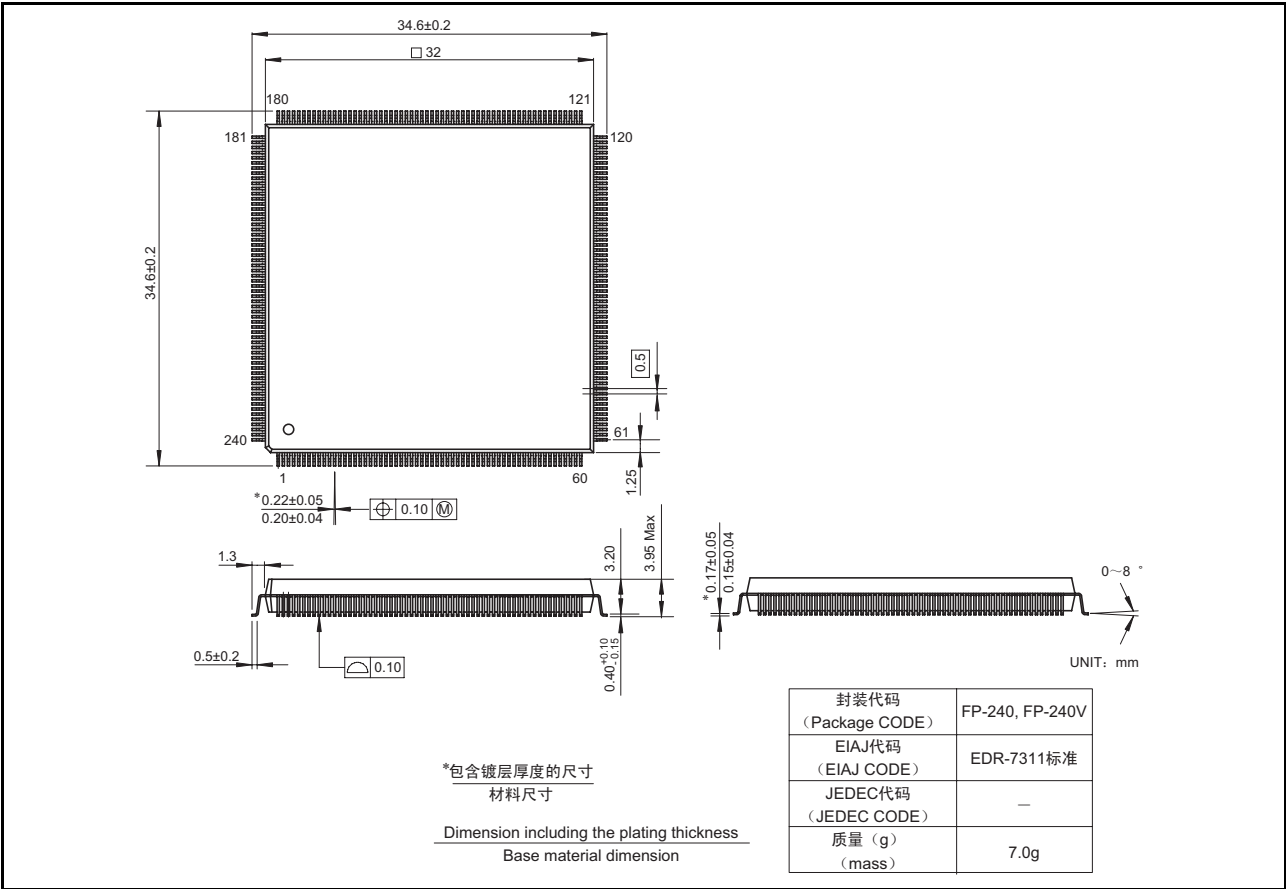
引脚功能		引脚状态					
分类	引脚名称	普通状态 (右边情况除外)	复位状态		低功耗状态		总线权释放状态
			上电复位 *1	保持引脚 状态 *2	深度待机 模式 *3	软件待机 模式	
LCDC	LCD_DATA15 ~ LCD_DATA0	O	—	O/Z*7	O/Z*7	O/Z*7	O
	LCD_DON	O	—	O/Z*7	O/Z*7	O/Z*7	O
	LCD_CL1、LCD_CL2	O	—	O/Z*7	O/Z*7	O/Z*7	O
	LCD_M_DISP	O	—	O/Z*7	O/Z*7	O/Z*7	O
	LCD_FLM	O	—	O/Z*7	O/Z*7	O/Z*7	O
	LCD_VCPWC、LCD_VEPWC	O	—	O/Z*7	O/Z*7	O/Z*7	O
	LCD_CLK	I	—	Z	Z	Z	I
I/O 端口	PA7 ~ PA0	I	Z	Z	Z	Z	I
	PB12	O	—	O/Z*7	O/Z*7	O/Z*7	O
	PB11 ~ PB8	I/O	Z	K/Z*7	K/Z*7	K/Z*7	I/O
	PB7 ~ PB0	I	I	I	I	I	I
	PC14 ~ PC2、PC0	I/O	Z	K/Z*7	K/Z*7	K/Z*7	I/O
	PC1	I/O	Z*5	K/Z*7	K/Z*7	K/Z*7	I/O
	PD15 ~ PD0	I/O	Z*5	K/Z*7	K/Z*7	K/Z*7	I/O
	PE15 ~ PE0	I/O	Z	K/Z*7	K/Z*7	K/Z*7	I/O
	PF30 ~ PF0	I/O	Z	K/Z*7	K/Z*7	K/Z*7	I/O
H-JDI	TRST	I	I	I	Z	I	I
	TCK	I	I	I	Z	I	I
	TDI	I	I	I	Z	I	I
	TDO	O/Z*14	O/Z*14	O/Z*14	O/Z*14	O/Z*14	O/Z*14
	TMS	I	I	I	Z	I	I
仿真器 *15	AUDSYNC	—	—	—	—	—	—
	AUDCK	—	—	—	—	—	—
	AUDATA3 ~ AUDATA0	—	—	—	—	—	—
	ASEBRKAK/ASEBRK	Z	Z	Z	Z	Z	Z

【符号说明】

- I : 输入
- O : 输出
- H : 高电平输出
- L : 低电平输出
- Z : 高阻抗
- K : 输入引脚为高阻抗, 输出引脚保持其状态

- 【注】 *1 通过向 $\overline{\text{RES}}$ 引脚输入低电平，指定上电复位。由 H-UDI 复位有效命令及 WDT 上溢引起上电复位时，各引脚的状态与初始功能普通运行时的引脚状态相同（详情参阅“第 25 章 引脚功能控制器（PFC）”）。
- *2 指定通过输入 $\overline{\text{NMI}}$ 、 $\overline{\text{MRES}}$ 、 $\text{IRQ7} \sim \text{IRQ0}$ 各引脚从深度待机模式返回后，到清除深度待机解除源标志寄存器（DSFR）的 IOKEEP 位之前的状态（详情参阅“第 28 章 低功耗模式”）。
- *3 内置于 I/O 引脚的弱保持电路处于关闭状态。
- *4 不使用连接晶体谐振器的引脚时，必须固定（上拉 / 下拉 / 接通电源 / 接地）输入引脚（EXTAL、RTC_X1、AUDIO_X1、USB_X1），并将输出引脚（XTAL、RTC_X2、AUDIO_X2、USB_X2）置为开路。
- *5 因区域 0 的数据总线宽度不同，初始功能各异（详情参阅“第 25 章 引脚功能控制器（PFC）”）。
- *6 取决于 CPG 的频率控制寄存器（FRQCR）的 CKOEN 位的设定（详情参阅“第 4 章 时钟脉冲振荡器（CPG）”）。
- *7 取决于待机控制寄存器 3（STBCR3）的 HIZ 位的设定（详情参阅“第 28 章 低功耗模式”）。
- *8 取决于 BSC 的共用控制寄存器（CMNCR）的 HIZMEM 位的设定（详情参阅“第 9 章 总线状态控制器（BSC）”）。
- *9 取决于 BSC 的共用控制寄存器（CMNCR）的 HIZCNT 位的设定（详情参阅“第 9 章 总线状态控制器（BSC）”）。
- *10 取决于深度待机解除源选择寄存器（DSSSR）各位的设定（详情参阅“第 28 章 低功耗模式”）。
- *11 取决于 RTC 的 RTC 控制寄存器（RCR2）的 RTCEN 位的设定（详情参阅“第 14 章 实时时钟（RTC）”）。
- *12 取决于待机控制寄存器（STBCR）的 AXTALE 位的设定（详情参阅“第 28 章 低功耗模式”）。
- *13 深度待机控制寄存器 2（DSCTR2）的 CS0KEEPE 位为 1 时，保持深度待机模式的引脚状态；CS0KEEPE 位为 0 时，为上电复位的引脚状态（详情参阅“第 28 章 低功耗模式”）。
- *14 H-UDI 的 TAP 控制器不是 Shift-DR、Shift-IR 状态时，为 Z。
- *15 产品芯片模式时（ASEMD=H）的引脚状态。有关 ASE 模式时（ASEMD=L）的引脚状态，详情参阅“仿真手册”。

附录 B. 封装尺寸图



索引

数字

16 位 /32 位位移量	35
---------------------	----

A

A/D 转换时间（单通道模式）	678
A/D 转换时序	677
A/D 转换精度的定义	679
A/D 转换开始请求延迟功能	388
A/D 转换器的启动	393
A/D 转换器特性	1063
A/D 转换器（ADC）	666
A/D 转换时间（多通道模式 / 扫描模式）	678
AC 特性	1002
AC 特性测量条件	1062
ADC 时序	1050
AND/NAND 闪存控制器（FLCTL）	687

B

BCHG 中断	771
BEMP 中断	766
BRDY 中断	763
报警功能的使用方法	462
倍频比的更改	78
比较匹配定时器（CMT）	433
不接受异常处理时	92

C

CAN 接口	595
CAN 睡眠模式	646
CAN 总线接口	664
CFIFOSEL	728
CMCNT 的写入与比较匹配的竞争	438
CMCNT 的字节写入与递增计数的竞争	439
CMCNT 的字写入与递增计数的竞争	438
CMCNT 计数时序	437
CPU	28
CSn 有效期间的扩展	199
彩色调色板规格	827
槽非法指令	91
测试模式的设定	648
产生紧随延迟转移指令之后的异常源	92
乘法 / 乘法累加运算	34
乘加低位寄存器（MACL）	31
乘加高位寄存器（MACH）	31
程序计数器（PC）	31
程序执行状态	61
处理发送条件	796
出栈	122
传送时钟	520
传送速率	542
串行位时钟控制	591

串行音频接口（SSI）	567
从存储体返回	122
从属发送运行	554
从属接收运行	555
存储器的数据格式	32
存储器分配高速缓存的结构	146
存储体保存、返回的运行	121
存储体的对象寄存器与输入 / 输出方式	120
存储体激活	217
存取长度与数据对齐	188
存取长度与突发数的关系	211
存取等待控制	197
存取周期之间的空闲周期	242

D

D/A 转换器特性	1063
D/A 转换器（DAC）	683
DC 特性	997
DMAC 的启动	393
DMAC 接口	663
DMAC 时序	1041
DMAC 使用时的限制事项	506
DMA 传送流程图	268
DREQ 引脚的采样时序	281
DTCH 中断	771
带 FIFO 的串行通信接口（SCIF）	464
待机控制电路	71
带字节选择的 SRAM 接口	230
单次读取	214
单次写入	215
单地址模式	278
单精度浮点格式	62
单通道模式	671
低功耗模式	904
低功耗状态	61
低功率 SDRAM	227
低频模式	223
地址错误	86
地址多路复用	204
地址映射	153
地址阵列	139, 146
电特性	995
电源控制顺序处理	831
掉电模式	224
读取地址阵列	146
读取数据阵列	147
断点的发送	506
断点的检测和处理	506
端口 A	883
端口 B	885
端口 C	887
端口 D	890
端口 E	893

端口 F	896
对绝对精度的影响	682
多功能定时器脉冲单元 2 (MTU2)	286
多路复用一览表 (端口 B)	847
多路复用一览表 (端口 C)	847
多路复用一览表 (端口 D)	848
多路复用一览表 (端口 E)	849
多路复用一览表 (端口 F)	850
多路复用引脚一览表 (端口 A)	846
多通道模式	673

E

ECC 错误检查	710
ECC 码	713

F

FLCTL 的中断请求	714
FLCTL 时序	1051
FPU 相关的 CPU 指令	59
FPU 异常处理	68
FPU 异常源	68
发送触发控制字段	607
发送触发时间 (TTT)	607
非规格化数	64
非数 (NaN)	64
非线性误差	679
非压缩模式	578
分频比的更改	78
分频器 1	71
分频器 2	71
浮点单元 (FPU)	62
浮点的范围	63
浮点寄存器	65
浮点异常	68
浮点运算指令	57, 92
复位顺序	645
复位同步 PWM 模式	358
复位状态	61

G

高速缓存	138
高速缓存的搜索	143
高速缓存与外部存储器的比配	146
高速缓存运行汇总	145
各运行模式的寄存器状态一览表	993
固定模式	272
管道安排	796
管道控制	772
过程寄存器 (PR)	31

H

H-UDI 复位	932
----------------	-----

H-UDI 命令	929
H-UDI 时序	1061
H-UDI 中断	107, 932
互补 PWM 模式	360
缓冲存储器	776
恢复中断	771
回写缓冲器 (仅限操作数高速缓存)	145

I

I/O 端口	883
I/O 端口时序	1061
I ² C 总线格式	550
I ² C 总线接口 3 (IIC3)	537

J

寄存器

ABACK0	629
ABACK1	629
ADCSR	669
ADDRA ~ ADDRH	668
BAMR	128
BAR	128
BBR	130
BCR0	615, 617
BCR1	615
BDMR	129
BDR	129
BEMPENB	738
BEMPSTS	744
BRCR	131
BRDYENB	736
BRDYSTS	742
CCR	640
CCR1	140
CCR2	141
CFBCFG	725
CFIFO	727
CFIFOCTR	731
CFIFOSEL	728
CFIFOSIE	732
CHCR	257
CMA _{MAX} _TEW	636
CMCNT	436
CMCOR	436
CMCSR	435
CMNCR	156
CMSTR	434
CS0WCR	161, 172, 183
CS1WCR	163
CS2WCR	165, 176
CS3WCR	165, 177
CS4WCR	166, 174
CS5WCR	168, 179
CS6WCR	170, 179, 181

CS7WCR	163	ICR0	99
CSnBCR (n=0 ~ 7)	158	ICR1	100
CYCTR	641	ICR2	101
D0FBCFG	725	ICSR	546
D0FIFO	727	IFCR	880
D0FIFOCTR	731	IMR	623
D0FIFOSEL	728, 729	INTENB0	733
D0FIFOTRN	733	INTENB1	735
D1FBCFG	725	INTSTS0	739
D1FIFO	727	INTSTS1	740
D1FIFOCTR	731	IPR01、IPR02、IPR05 ~ IPR17	98
D1FIFOSEL	728, 729	IRQRR	102
D1FIFOTRN	733	IRR	619
DACR	684	LDACLNR	814
DADR0	684	LDCNTR	818
DADR1	684	LDDFR	805
DAR	256	LDHCNR	810
DCPCFG	749	LDHSYNR	811
DCPCTR	750	LDICKR	802
DCPMAXP	750	LDINTR	814
DMAOR	262	LDLAOR	808
DMARS0 ~ DMARS3	265	LDLIRNR	821
DMATCR	256	LDMTR	803
DSCTR	914	LDPALCR	809
DSCTR2	915	LDPMMR	816
DSFR	917	LDPR	809
DSRTR	918	LDPSPR	817
DSSSR	916	LDSARL	808
DVSTCTR	722	LDSARU	807
FLADR	695	LDSMR	806
FLADR2	696	LDVDLNR	812
FLBSYCNT	702	LDUINTLNR	820
FLBSYTMR	701	LDUINTR	819
FLCMCDR	695	LDVSYNR	813
FLCMDCR	693	LDVTLNR	812
FLCMNCR	691	MBIMR0	632
FLCTL	687	MBIMR1	632
FLDATAR	698	MCR	610
FLDTCNTR	697	NF2CYC	549
FLDTFIFO	702	NRDYENB	737
FLECFIFO	703	NRDYSTS	743
FLINTDMACR	699	PADRL	884
FLTRCR	703	PBCRL1	855
FPSCR	66	PBCRL2	854
FPUL	67	PBCRL3	853
FRMNUM	745	PBCRL4	852
FRQCR	76	PBDRL	885
GSR	614	PBIORL	852
IBCR	105	PBPRL	887
IBNR	106	PCCRL1	859
ICCR1	541	PCCRL2	858
ICCR2	543	PCCRL3	857
ICDRR	548	PCCRL4	856
ICDRS	549	PCDRL	888
ICDRT	548	PCIORL	856
ICIER	545	PCPRL	889
ICMR	544	PDCRL1	866

PDCRL2	864	RSECCNT	451
PDCRL3	862	RTCNT	187
PDCRL4	860	RTCOR	187
PDDR1	891	RTCSR	186
PDIORL	860	RWKAR	456
PDPRL	892	RWKCNT	453
PECRL1	871	RXPR0	630
PECRL2	870	RXPR1	630
PECRL3	869	RYRAR	458
PECRL4	868	RYRCNT	454
PEDRL	894	SAR (DMAC)	256
PEIORL	867	SAR (IIC3)	548
PEPRL	895	SCEMR	488
PFCRH1	875	SCFCR	483
PFCRH2	874	SCFDR	485
PFCRH3	873	SCFRDR	467, 505
PFCRH4	872	SCFSR	472
PFCRL1	879	SCFTDR	468
PFCRL2	878	SCLSR	487
PFCRL3	877	SCRSR	467
PFCRL4	876	SCSCR	470
PFDRH	897	SCSMR	468, 488
PFDR1	897	SCSPTR	486
PFIORH	872	SCSR	881
PFIORL	872	SCTSR	467
PFPRH	899	SDBPR	929
PFPR1	899	SDCR	184
PINTER	103	SDIR	929
PIPEBUF	755	SSCR2	517
PIPECFG	753	SSCRH	511
PIPEMAXP	756	SSCRL	512
PIPEnCTR (n=1 ~ 7)	758	SSER	514
PIPEPER	757	SSICR	571
PIPEPERI	757	SSIRDR	578
PIPESEL	752	SSISR	575
PIRR	104	SSITDR	577
R64CNT	451	SSMR	513
RCR1	458	SSRDR0 ~ SSRDR3	519
RCR2	459	SSSR	515
RCR3	460	SSTDR0 ~ SSTDR3	518
RDAR	261	SSTRSR	519
RDAYAR	457	STBCR	905
RDAYCNT	453	STBCR2	906
RDMATCR	262	STBCR3	907
REC	623	STBCR4	908
RFMK	642	STBCR5	909
RFPR0	631	STBCR6	910
RFPR1	631	SYSCFG	720
RFTROFF	637	SYSCR1	911
RHRAR	456	SYSCR2	912
RHRCNT	452	SYSCR3	913
RMINAR	455	SYSSTS	721
RMINCNT	452	TADCOBRA_4	388
RMONAR	457	TADCOBRB_4	388
RMONCNT	454	TADCORA_4	388
RSAR	261	TADCORB_4	388
RSECAR	455	TADCR	314

TBTER.....	331
TBTM.....	312
TCBR.....	328
TCDR.....	327
TCMR0.....	642
TCMR1.....	643
TCMR2.....	643
TCNT.....	316
TCNTR.....	641
TCNTS.....	327
TCR.....	293
TDDR.....	327
TEC.....	623
TESTMODE.....	724
TGCR.....	326
TGR.....	316
TICCR.....	313
TIER.....	307
TIOR.....	298
TITCNT.....	330
TITCR.....	328
TMDR.....	296
TOCR1.....	321
TOCR2.....	323
TOER.....	320
TOLBR.....	325
TRWER.....	319
TSR.....	309, 638
TSTR.....	317
TSYR.....	318
TTCR0.....	634
TTTSEL.....	644
TWCR.....	333
TXACK0.....	628
TXACK1.....	628
TXCR0.....	627
TXCR1.....	627
TXPR0.....	625
TXPR1.....	625
UFRMNUM.....	746
UMSR0.....	633
UMSR1.....	633
USBACSWR.....	759
USBADDR.....	747
USBINDX.....	748
USBLENG.....	748
USBREQ.....	747
USBVAL.....	748
WRCSR.....	443
WTCNT.....	447
WTCSR.....	442
寄存器存储器.....	120
寄存器存储体.....	31
寄存器存储体错误.....	87
寄存器存储体错误异常处理.....	87, 122
寄存器存储体异常.....	122
寄存器的数据格式.....	32

寄存器地址一览表 (按各功能模块、手册章节编号顺序).....	934
寄存器位一览表.....	953
ID 排序.....	613
IIC3 时序.....	1047
级联运行.....	344
IRQ 中断.....	107
加载存储结构.....	33
间隔定时器模式的使用方法.....	446
晶体振荡器.....	71
竞争错误.....	530
就近舍入.....	67
局部接收过滤器屏蔽 (LAFM).....	604
决定空闲周期数的项目.....	243
绝对地址.....	35
绝对最大额定值.....	995

K

看门狗定时器模式的使用方法.....	445
看门狗定时器 (WDT).....	440
可支持的 DMA 传送.....	275
控制寄存器的初始值.....	31
控制器局域网 (RCAN-TL1).....	592
控制信号时序.....	1005
控制传送阶段转移中断.....	769

L

LCDC 时序.....	1060
LCD 控制器 (LCDC).....	799
LRU.....	139
立即数.....	34
立即数的数据格式.....	32
量化误差.....	679
轮询模式.....	272
逻辑运算指令.....	52

M

MPX-I/O 接口.....	200
MTU2 功能一览表.....	286
MTU2 时序.....	1042
MTU2 输出引脚的初始化方法.....	413
MTU2 中断源.....	392
满刻度误差.....	679
命令存取模式.....	704
模块待机功能.....	925
模块待机模式的设定.....	536
模拟电压的设定.....	680
模拟输入电压的设定.....	686
模拟输入引脚的处理.....	681
模拟输入引脚的规格.....	681

N

NMI 中断	107
NRDY 中断	765
NYET 信号交换响应表	789
内部 RAM (高速) 地址空间	902
内部 RAM (用于保持) 地址空间	902
内部外围模块请求模式	270
内部外围模块中断	108

P

PCMCIA 接口	234
PINT 中断	108
PLL 电路	71
PLL 振荡电路使用时的注意事项	79
PWM 模式	348
批量传送	788
偏移误差	679
频率更改步骤	445
频率更改方法	78
普通空间接口	149

Q

器件状态转移中断	767
清除中断源的时序	124
求取 SCBRR 设定值的公式	476
取指令周期的断点	133
全局基址寄存器 (GBR)	30

R

RCAN	596, 662
RCAN-TL1 的控制寄存器	610
RCAN-TL1 的存储器映射	596
RCAN-TL1 的存储器映射 (1 个通道)	596
RCAN-TL1 的设定	645
RCAN-TL1 的中断源	662
RCAN-TL1 定时器寄存器	634
RCAN-TL1 时序	1050
RCAN-TL1 引脚端口设定	664
RCAN-TL1 邮箱寄存器	624
RCAN-TL1 中断源	662
RISC 结构	33
容许信号源阻抗	682
软件待机模式	919
软件待机模式的解除步骤 (WDT)	445
软件待机模式时的 D/A 输出保持功能	686

S

SACK 中断	771
SCIF 时序	1043
SCIF 中断源	505
SDRAM 接口	149
SIGN 中断	771

SOF 插值功能	796
SSI 模块的总线格式	578
SSI 时序	1048
SSU 模式	523
SSU 中断源	536
SSU 时序	1044
扫描模式	675
扇区存取模式	704
上电 / 断电顺序	996
上电复位	84
上电顺序	224
上电、断电顺序	996
舍入	67
深度待机模式	922
深度掉电模式	228
时戳	606
时间触发发送	651
时间触发控制 (TT 控制)	607
时间触发系统的例子	657
时间从动	654
实时时钟 (RTC)	448
时钟脉冲振荡器 (CPG)	69
时钟频率控制电路	71
时钟时序	1002
时钟同步串行格式	557
时钟同步模式时的运行	498
时钟运行模式	73
手动复位	85
输出补充电路	1062
数据存取周期的断点	134
数据 PID 顺序位	775
数据阵列	139, 147
数据传送指令	47
刷新请求与总线周期的关系	223
双地址模式	276
双精度浮点格式	62
睡眠模式	919
算术运算指令	50

T

T 位	34
TAP 控制器	930
TDO 输出时序	931
TTW[1:0] (时间触发窗)	607
TSG 与 TSEG 的设定	618
跳转表基址寄存器 (TBR)	30
停机模式	646
同步串行通信单元 (SSU)	508
同步传送	791
通过绝对地址参照	35
通过立即数参照	34
通过位移量参照	35
通过中断请求信号传送数据	123
通用寄存器	28
通用寄存器的初始值	31
突发读取	211

突发 MPX-I/O 接口	238
突发模式	280
突发 ROM（时钟同步）接口	241
突发 ROM（时钟异步）接口	149
突发写入	214

U

UBC 时序	1041
USB2.0 主机 / 功能模块（USB）	715
USB 时序	1058
USB 数据总线电阻控制	760

V

VBUS 中断	771
---------------	-----

W

WDT 时序	1043
外部触发输入时序	678
外部请求模式	269
位操作指令	59
微处理器接口（MPI）	594
位同步电路	564
无延迟槽的无条件转移指令	34

X

系统寄存器的初始值	31
系统矩阵	609
系统控制指令	55
陷阱指令	90
显示分辨率的设定	831
显示 OFF 模式（LCDC 停止）的注意事项	834
向 0 舍入	67
向存储体保存	121
向量基址寄存器（VBR）	30
相位计数模式	352
写入地址阵列（无联想）	146
写入地址阵列（有联想）	146
写入数据阵列	147
信息发送请求	649
信息接收顺序	659
信息控制字段	601
信息数据字段	605
选择主机控制器功能时的控制传送	786
寻址方式	36

Y

压栈	122
延迟转移指令	33
液晶模块电源状态	834
页竞争	903
一般非法指令	91

异步模式的接收数据采样时序	506
异步模式时的运行	490
异常处理	80
异常处理的种类与优先顺序	80
异常处理后的堆栈状态	93
异常处理向量表	82
异常处理向量表地址的计算方法	83
异常处理状态	61
移位指令	53
引脚功能控制器（PFC）	846
硬件旋转运行	835
用户调试接口（H-UDI）	927
用户断点控制器（UBC）	125
用户断点运行的流程	133
用户断点中断	107
用于发送的内部仲裁	650
有关端口 A 的功能转换	882
邮箱	594, 597
邮箱的重新设定	660
邮箱的作用	598
邮箱功能的设定	603
邮箱控制	594
有效地址的计算方法	36
预取操作（仅限操作数高速缓存）	144

Z

在 USB 通信允许状态禁止设定的寄存器	774
噪声消除电路	559
帧更新中断	770
整数除法异常	91
直接存储器存取控制器（DMAC）	251
指令格式	40
指令特点	33
指令系统	43
指令引起的异常	90
中断控制器（INTC）	95
中断响应时间	115
中断异常处理	89
中断异常处理结束后的堆栈状态	115
中断优先顺序	89
中断传送	790
周期挪用模式	279
周期设定时的注意事项	403
主发送运行	551
主接收运行	552
转移指令	54
状态寄存器（SR）	29
自动请求模式	269
自动刷新	221
字节序	188
字数据的符号扩展	33
自刷新	222
总线权释放状态	61
总线时序	1007
总线仲裁	248
总线状态控制器（BSC）	149

修订记录	SH7203 群硬件手册
------	--------------

Rev.	发行日	修订内容	
		页	修订处
1.00	2009.03.27	—	初版发行

**瑞萨 32 位 RISC 单片机
硬件手册
SH7203 群**

Publication Date: Rev1.00, Mar. 27, 2009
Published by: Sales Strategic Planning Div.
Renesas Technology Corp.
Edited by: Customer Support Department
Global Strategic Communication Div.
Renesas Solutions Corp.

Renesas Technology Corp. Sales Strategic Planning Div. Nippon Bldg., 2-6-2, Ohte-machi, Chiyoda-ku, Tokyo 100-0004, Japan



RENESAS SALES OFFICES

<http://www.renesas.com>

Refer to "<http://www.renesas.com/en/network>" for the latest and detailed information.

Renesas Technology America, Inc.

450 Holger Way, San Jose, CA 95134-1368, U.S.A
Tel: <1> (408) 382-7500, Fax: <1> (408) 382-7501

Renesas Technology Europe Limited

Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K.
Tel: <44> (1628) 585-100, Fax: <44> (1628) 585-900

Renesas Technology (Shanghai) Co., Ltd.

Unit 204, 205, AZIA Center, No.1233 Lujiazui Ring Rd, Pudong District, Shanghai, China 200120
Tel: <86> (21) 5877-1818, Fax: <86> (21) 6887-7858/7898

Renesas Technology Hong Kong Ltd.

7th Floor, North Tower, World Finance Centre, Harbour City, Canton Road, Tsimshatsui, Kowloon, Hong Kong
Tel: <852> 2265-6688, Fax: <852> 2377-3473

Renesas Technology Taiwan Co., Ltd.

10th Floor, No.99, Fushing North Road, Taipei, Taiwan
Tel: <886> (2) 2715-2888, Fax: <886> (2) 3518-3399

Renesas Technology Singapore Pte. Ltd.

1 Harbour Front Avenue, #06-10, Keppel Bay Tower, Singapore 098632
Tel: <65> 6213-0200, Fax: <65> 6278-8001

Renesas Technology Korea Co., Ltd.

Kukje Center Bldg. 18th Fl., 191, 2-ka, Hangang-ro, Yongsan-ku, Seoul 140-702, Korea
Tel: <82> (2) 796-3115, Fax: <82> (2) 796-2145

Renesas Technology Malaysia Sdn. Bhd

Unit 906, Block B, Menara Amcorp, Amcorp Trade Centre, No.18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: <603> 7955-9390, Fax: <603> 7955-9510

SH7203群



瑞萨电子株式会社

RCJ09B0067-0100