

本产品仅在中国大陆和香港以及印度销售

μPD79F7023, 79F7024

用户手册 硬件篇

8 位 单片上微控制器

本资料中包含的所有信息，包括产品和产品规格，仅代表文件发布时的产品信息，随时可能被瑞萨电子公司更改，恕不另行通知。请通过各种方式查阅瑞萨电子公司发布的最新信息，包括瑞萨电子公司网站(<http://www.renesas.com>)。

Notice

1. 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.
2. 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.
3. 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.
4. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from such alteration, modification, copy or otherwise misappropriation of Renesas Electronics product.
5. Renesas Electronics products are classified according to the following two quality grades: "Standard" and "High Quality". The recommended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below.
"Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots etc.
"High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; and safety equipment etc.

Renesas Electronics products are neither intended nor authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems, surgical implantations etc.), or may cause serious property damages (nuclear reactor control systems, military equipment etc.). 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 for which it is not intended. 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 which the product is not intended by Renesas Electronics.

6. 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.
7. 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 systems manufactured by you.
8. 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.
9. 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. You should not use Renesas Electronics products or 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. When exporting the Renesas Electronics 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.
10. It is the responsibility of the buyer or distributor of Renesas Electronics products, who distributes, disposes of, or otherwise places the product with a third party, to notify such third party in advance of the contents and conditions set forth in this document, Renesas Electronics assumes no responsibility for any losses incurred by you or third parties as a result of unauthorized use of Renesas Electronics products.
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.

注意事项

1. 本文档中所记载的关于电路、软件和其他相关信息仅用于说明半导体产品的操作和应用实例。用户如在设备设计中应用本文档中的电路、软件和相关信息，请自行负责。对于用户或第三方因使用上述电路、软件或信息而遭受的任何损失，瑞萨电子不承担任何责任。
2. 在准备本文档所记载的信息的过程中，瑞萨电子已尽量做到合理注意，但是，瑞萨电子并不保证这些信息都是准确无误的。用户因本文档中所记载的信息的错误或遗漏而遭受的任何损失，瑞萨电子不承担任何责任。
3. 对于因使用本文档中的瑞萨电子产品或技术信息而造成的侵权行为或因此而侵犯第三方的专利、版权或其他知识产权的行为，瑞萨电子不承担任何责任。本文档所记载的内容不应视为对瑞萨电子或其他人所有的专利、版权或其他知识产权作出任何明示、默示或其它方式的许可及授权。
4. 用户不得更改、修改、复制或以其他方式部分或全部地非法使用瑞萨电子的任何产品。对于用户或第三方因上述更改、修改、复制或以其他方式非法使用瑞萨电子产品的行为而遭受的任何损失，瑞萨电子不承担任何责任。
5. 瑞萨电子产品根据其质量等级分为两个等级：“标准等级”和“高质量等级”。每种瑞萨电子产品的推荐用途均取决于产品的质量等级，如下所示：

标准等级： 计算机、办公设备、通讯设备、测试和测量设备、视听设备、家用电器、机械工具、个人电子设备以及工业机器人等。

高质量等级： 运输设备（汽车、火车、轮船等）、交通控制系统、防灾系统、预防犯罪系统以及安全设备等。

瑞萨电子产品无意用于且未被授权用于可能对人类生命造成直接威胁的产品或系统及可能造成人身伤害的产品或系统（人工生命维持装置或系统、植埋于体内的装置等）中，或者可能造成重大财产损失的产品或系统（核反应堆控制系统、军用设备等）中。在将每种瑞萨电子产品用于某种特定应用之前，用户应先确认其质量等级。不得将瑞萨电子产品用于超出其设计用途之外的任何应用。对于用户或第三方因将瑞萨电子产品用于其设计用途之外而遭受的任何损害或损失，瑞萨电子不承担任何责任。

6. 使用本文档中记载的瑞萨电子产品时，应在瑞萨电子指定的范围内，特别是在最大额定值、电源工作电压范围、移动电源电压范围、热辐射特性、安装条件以及其他产品特性的范围内使用。对于在上述指定范围之外使用瑞萨电子产品而产生的故障或损失，瑞萨电子不承担任何责任。
7. 虽然瑞萨电子一直致力于提高瑞萨电子产品的质量和可靠性，但是，半导体产品有其自身的具体特性，如一定的故障发生率以及在某些使用条件下会发生故障等。此外，瑞萨电子产品均未进行防辐射设计。所以请采取安全保护措施，以避免当瑞萨电子产品在发生故障而造成火灾时导致人身事故、伤害或损害的事故。例如进行软硬件安全设计（包括但不限于冗余设计、防火控制以及故障预防等）、适当的老化处理或其他适当的措施等。由于难于对微机软件单独进行评估，所以请用用户自行对最终产品或系统进行安全评估。
8. 关于环境保护方面的详细内容，例如每种瑞萨电子产品的环境兼容性等，请与瑞萨电子的营业部门联系。使用瑞萨电子产品时，请遵守对管制物质的使用或含量进行管理的所有相应法律法规（包括但不限于《欧盟 RoHS 指令》）。对于因用户未遵守相应法律法规而导致的损害或损失，瑞萨电子不承担任何责任。
9. 不可将瑞萨电子产品和技术用于或者嵌入日本国内或海外相应的法律法规所禁止生产、使用及销售的任何产品或系统中。也不可将本文档中记载的瑞萨电子产品或技术用于与军事应用或者军事用途有关的任何目的（如大规模杀伤性武器的开发等）。在将本文档中记载的瑞萨电子产品或技术进行出口时，应当遵守相应的出口管制法律法规，并按照上述法律法规所规定的程序进行。
10. 向第三方分销或处分产品或以其他方式将产品置于第三方控制之下的瑞萨电子产品买方或分销商，有责任事先向上述第三方通知本文档规定的内容和条件；对于用户或第三方因非法使用瑞萨电子产品而遭受的任何损失，瑞萨电子不承担任何责任。
11. 在事先未得到瑞萨电子书面认可的情况下，不得以任何形式部分或全部转载或复制本文档。
12. 如果对本文档所记载的信息或瑞萨电子产品有任何疑问，或者用户有任何其他疑问，请向瑞萨电子的营业部门咨询。

（注 1） 瑞萨电子：在本文档中指瑞萨电子株式会社及其控股子公司。

（注 2） 瑞萨电子产品：指瑞萨电子开发或生产的任何产品。

关于 CMOS 器件的注意事项

① 输入引脚处的施加电压波形

输入噪声或由反射波引起的波形失真可能导致故障发生。如果由于噪声等影响，使CMOS器件的输入电压范围处于在 V_{IL} (MAX) 和 V_{IH} (MIN) 之间，器件可能发生故障。在输入电平固定时以及输入电平从 V_{IL} (MAX) 到 V_{IH} (MIN) 的过渡期间，要谨防颤振噪声进入器件。

② 未使用的输入引脚的处理

CMOS器件上未连接的输入端可能是故障源。如果一个输入引脚未被连接，则由于噪声等原因可能会产生内部输入电平，从而导致故障。CMOS器件的工作方式与双极性或NMOS器件不同。CMOS器件的输入电平必须借助上拉或下拉电路固定于高电平或低电平。每一个未使用引脚只要有可能成为输出引脚时，都应该通过附加电阻连接到 V_{DD} 或GND。对未使用引脚的处理因器件而异，必须遵循与器件相关的规格和说明。

③ ESD防护措施

如果MOS器件周围有强电场，将会击穿氧化栅极，降低器件的工作性能。因此必须采取措施，尽可能防止静电产生。一旦有静电，必须立即释放。环境必须控制适当。如果空气干燥，应当使用加湿器。建议避免使用容易产生静电的绝缘体。半导体器件的存放和运输必须使用抗静电容器、静电屏蔽袋或导电材料包装。所有包括工作台和工作面的测试和测量工具必须良好接地。操作员应当佩戴手腕带以保证良好接地。不能用手直接接触半导体器件。对装配有半导体器件的PW板也应采取类似的静电防范措施。

④ 初始化之前的状态

上电并不一定定义MOS器件的初始状态。刚接通电源时，具有复位功能的MOS器件并没有被初始化。因此上电不能保证输出引脚的电平、输入/输出设置和寄存器的内容。器件在收到复位信号后才进行初始化。具有复位功能的器件在上电后必须立即进行复位操作。

⑤ 电源上电/断电序列

器件内部工作和外部接口使用不同电源的情况下，原则上应先在接通内部电源之后再接通外部电源。当关闭电源时，原则上先关闭外部电源再关闭内部电源。如果电源开关顺序相反，可能会对器件的内部元件施加电压，从而由于异常电流的流过而造成故障和降低元件的性能。须视具体器件和支配器件的相关规格来单独决定正确的上电/断电序列。

⑥ 断电状态期间的信号输入

不要在器件断电时输入信号或输入/输出上拉电源。因为输入信号或提供输入/输出上拉电源将引起电流注入，从而引起器件的误操作，并且此时流过器件的异常电流引起内部元件性能劣化。须视具体器件和支配器件的相关规格来单独决定断电状态期间的信号输入。

本手册的使用方法

对 象

本手册以理解此系列的功能并且设计和开发其应用系统和程序的用户工程师为对象。
对象产品如下：

- μ PD79F7023, 79F7024

目 的

本手册以帮助用户理解下述结构中所示的功能为目的。

构 成

μ PD79F7023, 79F7024 的用户手册分为用户手册硬件篇和用户手册指令篇（78K/0 系列通用）共 2 本。

μ PD79F7023, 79F7024
用户手册
硬件篇

78K/0 系列
用户手册
指令篇

- 引脚功能
- 内部块功能
- 中断
- 其他的内部外围功能
- 电特性
- CPU 功能
- 指令集
- 指令的说明

阅读方法

阅读本手册的读者应具备电气、逻辑电路以及微控制器等的基础知识。

- 要理解全部功能时
→ 请按照**目录**的顺序阅读本手册。
- 寄存器格式的阅读方法
→ 关于(<>)符号内的位号，其位名称在 RA78K0 中被定义为保留字，而在 CC78K0 中被 # pragma sfr 指令定义为 sfr 变量。
- 要详细了解 78K/0 系列的指令功能时
→ 请参照另一本手册 **78K/0 系列用户手册 (U12326E)**。

凡 例

数据表示法:	左侧为高位，右侧为低位
有效低电平表示法:	xxx (在引脚或者信号名称上标注上划线)
注:	正文中加“注”的说明
注意事项:	需要留心阅读的内容
备注:	正文的补充说明
数制表示法:	二进制 ...xxxx 或者 xxxxB 十进制 ...xxxx 十六进制 ...xxxxH

相关资料 相关资料中可能包括暂定版。但是，在以下资料中并未特别注明“暂定版”，请谅解。

产品的相关资料

资料名称	资料号
μPD79F7023, 79F7024 用户手册 硬件篇	本手册
78K0 系列 用户手册 指令	U12326C
78K0/Kx2 Flash 存储器编程器 (编程器) 应用笔记	U17739C
78K0/Kx2 Flash Memory Self Programming User's Manual	U17516E
78K0 Microcontrollers Self Programming Library Type01 User's Manual	U18274E

闪存编程器的相关资料 (用户手册)

资料名称	资料号
PG-FP5 Flash Memory Programmer User's Manual	R20UT0008E
QB-MINI2 On-Chip Debug Emulator with Programming User's Manual	R20UT0449E
QB-Programmer Programming GUI Operation	U18527E

开发工具的相关资料 (硬件) (用户手册)

文档名称	文档编号
QB-MINI2 On-Chip Debug Emulator with Programming User's Manual	R20UT0449E

注意事项 上述相关资料的内容如有变更，恕不另行通知。设计等时请使用最新版本的资料。

开发工具的相关资料 (软件)

资料名称		资料号
RA78K0 Ver.3.80 Assembler Package User's Manual ^{註1}	Operation	U17199E
	Language	U17198E
	Structured Assembly Language	U17197E
78K0 Assembler Package RA78K0 Ver.4.01 Operating Precautions (Notification Document) ^{註1}		ZUD-CD-07-0181-E
CC78K0 Ver.3.70 C Compiler User's Manual ^{註2}	Operation	U17201E
	Language	U17200E
78K0 C Compiler CC78K0 Ver. 4.00 Operating Precautions (Notification Document) ^{註2}		ZUD-CD-07-0103-E
ID78K0-QB Ver.2.94 Integrated Debugger User's Manual	Operation	U18330E
ID78K0-QB Ver.3.00 Integrated Debugger User's Manual	Operation	U18492E
PM plus Ver.5.20 ^{註3} User's Manual		U16934E
PM+ Ver.6.30 ^{註4} User's Manual		U18416E

- 注 1.** 此文档在安装 RA78K0 Ver.4.01 时, 与工具一起安装到 PC 内。关于“78K0 汇编包 RA78K0 Ver.4.01 操作上的注意点”中没有记述的内容, 请参阅 RA78K0 Ver.3.80 的用户手册。
- 2.** 此文档在安装 CC78K0 Ver.4.00 时, 与工具一起安装到 PC 内。关于“78K0 C 汇编 CC78K0 Ver.4.00 操作上的注意点”中没有记述的内容, 请参阅 CC78K0 Ver.3.70 的用户手册。
- 3.** PM plus Ver.5.20 是附属在 RA78K0 Ver.3.80 的集成开发环境。
- 4.** PM+ Ver.6.30 是附属在 RA78K0 Ver.4.01 的集成开发环境。可管理不同版本的软件工具 (汇编程序, C 编译程序, 调试器和仿真器) 产品。

其他资料

资料名称	资料号
Semiconductor Package Mount Manual	注
Quality Grades on NEC Semiconductor Devices	C11531E
Guide to Prevent Damage for Semiconductor Devices by Electrostatic Discharge (ESD)	C11892E
Semiconductor Reliability Handbook	R51ZZ0001E

注 请参阅“半导体封装安装手册”网页。
<http://cn.renesas.com/products/package/manual/index.jsp>

注意事项 上述相关资料的内容如有变更, 恕不另行通知。设计等时请使用最新版本的资料。

所有商标及注册商标分别归属于其所有者。

本用户手册仅为参考译文, 对应的日文版和英文版具有正式效力。

EEPROM是瑞萨电子株式会社的注册商标。

SuperFlash是美国Silicon Storage Technology, Inc.在美国以及日本等国的注册商标。

注意: 本产品使用已获得Silicon Storage Technology, Inc.授权的 SuperFlash®。

目录

第一章 概述	1
1.1 特点.....	1
1.2 订购信息	2
1.3 引脚配置 (顶视图).....	3
1.4 框图.....	4
1.5 功能概述	5
第二章 引脚功能.....	6
2.1 引脚功能表.....	6
2.2 引脚功能描述.....	9
2.2.1 P20 至 P27 (端口 2).....	9
2.2.2 P30 至 P34 (端口 3).....	9
2.2.3 P121、P122、P125 (端口 12).....	10
2.2.4 AVREF、VDD、VSS	11
2.2.5 REGC.....	11
2.3 有关连接引脚 I/O 电路和未使用引脚的建议	13
第三章 CPU 架构	16
3.1 存储器空间.....	16
3.1.1 内部程序存储器空间	19
3.1.2 内部数据存储器空间	21
3.1.3 特殊功能寄存器(SFR: Special Function Register)区域	21
3.1.4 数据存储器寻址	22
3.2 处理程序寄存器	24
3.2.1 控制寄存器	24
3.2.2 通用寄存器	28
3.2.3 特殊功能寄存器 (SFR: Special Function Register)	29
3.3 指令地址寻址.....	34
3.3.1 相对寻址.....	34
3.3.2 立即寻址	35
3.3.3 表间接寻址	36
3.3.4 寄存器寻址	37
3.4 操作数地址寻址	37
3.4.1 隐含寻址.....	37
3.4.2 寄存器寻址	38
3.4.3 直接寻址.....	39
3.4.4 短直接寻址	40
3.4.5 特殊功能寄存器(SFR)寻址	41
3.4.6 寄存器间接寻址	42
3.4.7 基址寻址.....	43
3.4.8 寻址变址寻址.....	44
3.4.9 堆栈寻址	45
第四章 端口功能.....	46

4.1 端口功能	46
4.2 端口配置	48
4.2.1 端口 2	49
4.2.2 端口 3	59
4.2.3 端口 12	65
4.3 寄存器控制端口功能	68
4.4 端口功能操作	72
4.4.1 写入 I/O 端口	72
4.4.2 读取 I/O 端口	72
4.4.3 操作 I/O 端口	72
4.5 使用复用功能时，端口模式寄存器和输出锁存器的设置	73
4.6 端口寄存器 n(Pn)1 位操作指令的注意事项	75
第五章 时钟发生电路	76
5.1 时钟发生电路功能	76
5.2 时钟发生电路的配置	77
5.3 时钟发生电路的控制寄存器	79
5.4 系统时钟振荡电路	87
5.4.1 X1 振荡电路	87
5.4.2 内部高速振荡电路	89
5.4.3 内部低速振荡电路	89
5.4.4 预分频器	89
5.5 时钟发生电路操作	90
5.6 控制时钟	93
5.6.1 控制高速系统时钟示例	93
5.6.2 控制内部高速振荡时钟示例	96
5.6.3 控制内部低速振荡时钟示例	99
5.6.4 CPU 和外围硬件所采用时钟	100
5.6.5 CPU 时钟状态转换图	101
5.6.6 CPU 时钟的切换前条件和切换后处理	104
5.6.7 主系统时钟切换所需时间	105
5.6.8 时钟振荡停止前的条件	106
5.6.9 外围硬件和源时钟	106
第六章 16 位定时器/事件计数器 00	107
6.1 16 位定时器/事件计数器 00 功能	107
6.2 16 位定时器/事件计数器 00 配置	108
6.3 16 位定时器/事件计数器 00 的控制寄存器	114
6.4 16 位定时器/事件计数器 00 操作	122
6.4.1 间隔定时器操作	122
6.4.2 方波输出操作	125
6.4.3 外部事件计数器操作	128
6.4.4 通过 TI000 引脚有效沿输入进入清零并启动模式	132
6.4.5 自由运行定时器操作	145
6.4.6 PPG 输出操作	154
6.4.7 单脉冲输出操作	158
6.4.8 脉冲宽度测量操作	163
6.5 TM00 特殊用途	171
6.5.1 TM00 操作期间 CR010 的重写	171

6.5.2 LVS00 和 LVR00 的设置	171
6.6 16 位定时器/事件计数器 00 注意事项.....	173
第七章 8 位定时器/事件计数器 51	178
7.1 8 位定时器/事件计数器 51 功能.....	178
7.2 8 位定时器/事件计数器 51 配置.....	179
7.3 8 位定时器/事件计数器 51 控制寄存器	181
7.4 8 位定时器/事件计数器 51 的操作	183
7.4.1 用作间隔定时器	183
7.4.2 用作外部事件计数器	185
7.4.3 方波输出操作.....	186
7.4.4 PWM 输出操作	187
7.5 8 位定时器/事件计数器 51 注意事项.....	191
第八章 8 位定时器 H1	192
8.1 8 位定时器 H1 功能	192
8.2 8 位定时器 H1 配置	192
8.3 寄存器控制 8 位定时器 H1.....	195
8.4 8 位定时器 H1 操作	199
8.4.1 用作间隔定时器/方波输出	199
8.4.2 用作 PWM 输出	202
8.4.3 载波发生器操作	208
第九章 看门狗定时器	215
9.1 看门狗定时器功能.....	215
9.2 看门狗定时器配置.....	216
9.3 寄存器控制看门狗定时器.....	217
9.4 看门狗定时器操作.....	218
9.4.1 看门狗定时器的操作控制.....	218
9.4.2 看门狗定时器的溢出时间设置.....	219
9.4.3 看门狗定时器的窗口打开期间设置.....	220
第十章 A/D 转换器.....	222
10.1 A/D 转换器功能	222
10.2 A/D 转换器配置	224
10.3 A/D 转换器寄存器.....	226
10.4 A/D 转换器操作	233
10.4.1 A/D 转换器基本操作	233
10.4.2 输入电压和转换结果	235
10.4.3 A/D 转换器操作模式	236
10.5 A/D 转换器特征表的阅读方法.....	238
10.6 使用 A/D 转换器的注意事项	240
第十一章 运算放大器	244
11.1 运算放大器功能	244
11.2 运算放大器配置	245
11.3 运算放大器使用的寄存器	246
11.4 运算放大器操作	250

第十二章 比较器	251
12.1 比较器功能	251
12.2 比较器配置	251
12.3 寄存器控制比较器	252
12.4 比较器操作	256
12.4.1 启动比较器操作(参考电压使用内部参考电压).....	256
12.4.2 启动比较器操作 (参考电压使用 CMPCOM 引脚输入电压).....	257
12.4.3 停止比较器操作	257
第十三章 串行接口 UART0	258
13.1 串行接口 UART0 功能	258
13.2 串行接口 UART0 配置	259
13.3 寄存器控制串行接口 UART0	262
13.4 串行接口 UART0 操作	267
13.4.1 操作停止模式.....	267
13.4.2 异步串行接口(UART)模式	268
13.4.3 专用波特率发生器.....	274
13.4.4 波特率计算	275
第十四章 中断功能	279
14.1 中断功能类型	279
14.2 中断源和配置	279
14.3 寄存器控制中断功能	284
14.4 中断服务操作	291
14.4.1 可屏蔽中断响应	291
14.4.2 软件中断请求响应.....	294
14.4.3 多重中断服务.....	294
14.4.4 保留中断请求.....	297
第十五章 待机功能	298
15.1 待机功能和配置	298
15.1.1 待机功能.....	298
15.1.2 寄存器控制待机功能	299
15.2 待机功能操作	302
15.2.1 HALT 模式	302
15.2.2 STOP 模式	306
第十六章 复位功能	313
16.1 确认复位源的寄存器	321
第十七章 上电清除电路	322
17.1 上电清除电路功能	322
17.2 上电清除电路的配置	323
17.3 上电清除电路操作	323
17.4 上电清除电路注意事项	326
第十八章 低电压检测电路	328

18.1 低电压检测电路功能	328
18.2 低电压检测器配置	329
18.3 寄存器控制低电压检测器	330
18.4 低电压检测电路操作	333
18.4.1 用于复位	334
18.4.2 用于中断	337
18.5 低电压检测器注意事项	340
第十九章 稳压器	343
19.1 稳压器概述	343
19.2 寄存器控制稳压器	343
19.3 自编程注意事项	344
第二十章 选项字节	345
20.1 选项字节功能	345
20.2 选项字节格式	346
第二十一章 闪存	351
21.1 内存容量切换寄存器	351
21.2 闪存寄存器编程器的写入方法	352
21.3 编程环境	353
21.4 板上引脚连接	354
21.4.1 TOOL 引脚	354
21.4.2 $\overline{\text{RESET}}$ 引脚	355
21.4.3 端口引脚	355
21.4.4 REGC 引脚	355
21.4.5 其他信号引脚	355
21.4.6 电源	355
21.4.7 连接晶体/陶瓷谐振器时的板上写入	356
21.5 编程方法	357
21.5.1 控制闪存	357
21.5.2 闪存编程模式	357
21.5.3 通信命令	357
21.6 安全设置	359
21.7 通过自编程进行闪存编程	361
21.7.1 自编程模式控制寄存器	362
21.7.2 自编程流程(重写闪存寄存器)	362
21.7.3 引导交换功能	364
第二十二章 On-chip 调试功能	366
22.1 QB-MINI2 与 μ PD79F7023, 79F7024 的连接	366
22.2 On-chip 调试安全 ID	369
22.3 用户资源保留	370
第二十三章 指令集	371
23.1 应用操作表	371
23.1.1 操作标识符和指定方法	371
23.1.2 操作栏描述	372

23.1.3 标志操作列描述	372
23.2 操作表	373
23.3 按寻址分类列出的指令	381
第二十四章 电特性	384
第二十五章 封装尺寸图	403
第二十六章 待机注意事项	404
26.1 待机注意事项	404
26.2 产生等待的外围硬件	405
附录 A 开发工具	406
A.1 软件包	408
A.2 语言处理软件	408
A.3 闪存编程器的工具	409
A.3.1 使用闪存编程器 PG-FP5 和 FL-PR5	409
A.3.2 使用具备编程功能的片上调试仿真器 QB-MINI2	409
A.4 调试工具（硬件）	410
A.5 调试工具（软件）	410
修订记录	i

第一章 概述

1.1 特点

- 78K0 CPU 内核
- I/O 端口, ROM 和 RAM 容量

项目 产品	I/O 端口	程序存储器 (闪存)	数据存储器 (内部高速 RAM)
μPD79F7023	16 (CMOS I/O: 13, CMOS 输入: 3)	8 KB	512 字节
μPD79F7024		16 KB	768 字节

- 低功耗
 - 内部高速振荡器模式: 140 μA (典型值) ($V_{DD} = 3.0\text{ V}$, $f_{CPU} = 1\text{ MHz}$ 操作)
 - STOP 模式: 2.0 μA (典型值) ($V_{DD} = 5.0\text{ V}$ 操作)
- 时钟
 - 高速系统时钟 ... 可从以下三种源中进行选择
 - 陶瓷/晶体谐振器: 1 至 10 MHz
 - 外部时钟: 1 至 10 MHz
 - 内部高速振荡器: 4 MHz ± 3 % ($T_A = -40$ 至 + 85 °C), 4 MHz ± 2 % ($T_A = -20$ 至 + 70 °C)
 - 低速系统振荡器 240 kHz ± 10 % ... 看门狗定时器, 间歇工作时的定时器时钟
- 内置上电清除 (POC) 电路
- 内置低电压检测电路 (LVI) (达到检测电压时, 将产生断电/复位 (可选))
 - 检测电压: 可从 2.7 至 4.24V 之间的 11 个级别中选择
- 单电源闪存
 - 可自编程
 - 软件保护功能 ... 防止第三方复制(无 flash 读取指令)
- 安全功能
 - 可利用独立于 CPU 的时钟运行的看门狗定时器
 - ...即使系统时钟停止, 也可以检测出失控。
 - 可通过 LVI 检测出电源电压下降
 - ... 可在电源电压低于工作电压之前, 执行相应处理。
 - 备有选项字节功能
 - ...通过硬件设置重要系统运行设定。

○ 定时器

- 16 位定时器/事件计数器 ... PPG 输出, 捕捉输入, 外部事件计数器输入
- 8 位定时器 H ... PWM 输出, 可利用低速内部振荡时钟操作
- 8 位定时器/事件计数器 5 ... 外部事件计数器输入
- 看门狗定时器 ... 可利用低速内部振荡时钟操作

产品 \ 项目	16 位定时器/ 事件计数器	8 位定时器	看门狗定时器
μPD79F7023	1 通道	定时器 H: 1 通道	1 通道
μPD79F7024		定时器 5: 1 通道	

○ 串行接口

- UART ... 异步 2 线串行接口

产品 \ 项目	UART
μPD79F7023	1 通道
μPD79F7024	

○ 8 位分辨率 A/D 转换器: 5 通道

○ 运算放大器: 2 通道

○ 比较器: 1 通道

○ On-chip 调试功能 ... 可控制目标器件和参考存储器。

○ 支持汇编程序和 C 语言

○ 开发工具

- 提供简易仿真器(MINICUBE2)

○ 电源电压: $V_{DD} = 2.7$ 至 5.5 V

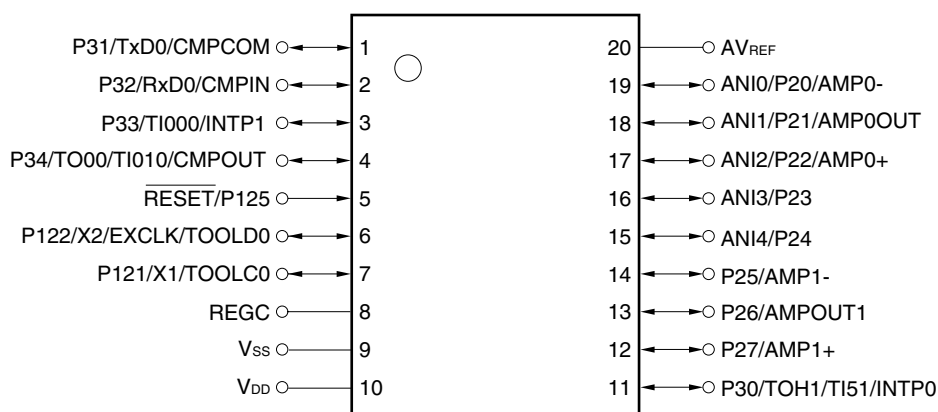
○ 工作环境温度: $T_A = -40$ 至 $+85^{\circ}\text{C}$

1.2 订购信息

引脚计数	封装	ROM	RAM	半导体部材	部件编号
20 针	20 针塑封 SSOP (7.62 mm (300))	8 KB	512 B	无铅产品 (外部引脚为 Ni/Pd/Au 涂装)	μPD79F7023MC-CAA-AX
		16 KB	768 B		μPD79F7024MC-CAA-AX

备注 所有产品均为无铅产品。

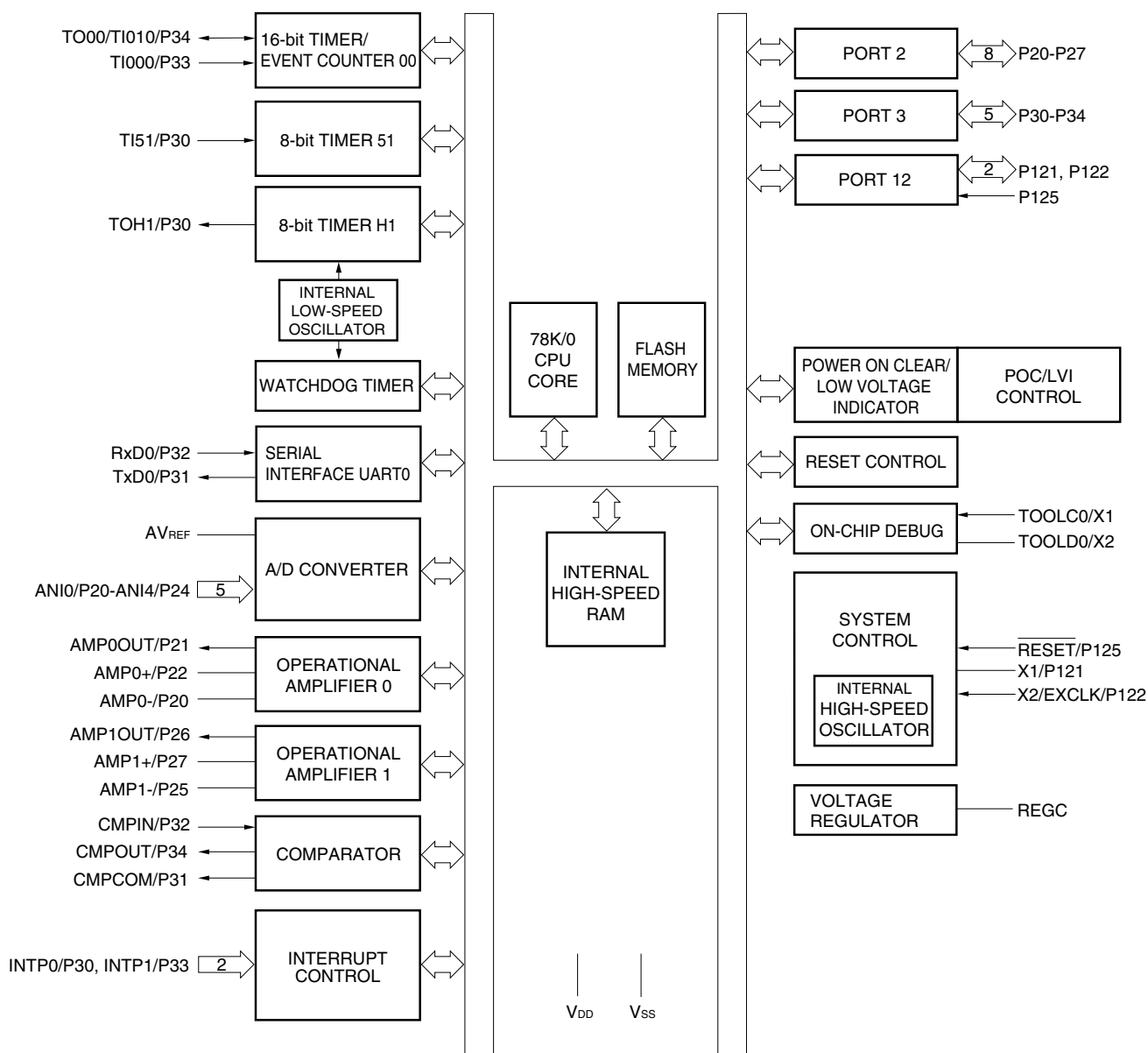
1.3 引脚配置 (顶视图)



AMP0-, AMP0+,	放大器输入	P30 至 P34 :	端口 3
AMP1-, AMP1+ :		P121, P122, P125 :	端口 12
AMPOUT, AMP1OUT :	放大器输出	REGC :	稳压器电容
ANI0 至 ANI4 :	模拟输入	RESET :	复位
AVREF :	模拟参考电压	RxD0 :	接收数据
		TI000, TI010, TI51 :	定时器输入
CMPCOM :	比较器共用输入	TO00, TOH1 :	定时器输出
		TOOLC0 :	工具用时钟输入
CMPIN	比较器输入	TOOLD0 :	工具用数据输入/输出
CMPOUT	比较器输出	TxD0 :	发送数据
EXCLK :	外部时钟输入	VDD :	电源
	(主系统时钟)	Vss :	接地
INTP0, INTP1 :	外部中断输入	X1, X2 :	晶体谐振器(主系统时钟)
P20 至 P27 :	端口 2		

- 注意事项**
1. Vss 可兼用作 A/D 转换器的接地电位。必须将 Vss 连接到稳定的 GND (= 0 V) 上。
 2. 通过电容器, 连接 REGC 引脚至 Vss (0.47 至 1 μF)。
 3. 复位解除后, ANI0/P20/AMP0-、ANI1/P21/AMPOUT、ANI2/P22/AMP0+、ANI3/P23、ANI4/P24、P31/TxD0/CMPCOM 和 P32/RxD0/CMPIN 为模拟输入模式。
 4. 复位解除后, RESET/P125 设置为外部复位输入。

1.4 框图



- 注意事项
1. V_{SS} 可兼用作 A/D 转换器的接地电位。必须将 V_{SS} 连接到稳定的 GND (= 0 V) 上。
 2. 通过电容器，连接 REGC 引脚至 V_{SS} (0.47 至 1 μF)。
 3. 复位解除后，ANI0/P20/AMP0-、ANI1/P21/AMP0OUT/PGAIN、ANI2/P22/AMP0+、ANI3/P23 和 ANI4/P24 为模拟输入模式。
 4. 复位解除后，RESET/P125 设置为外部复位输入。

1.5 功能概述

项目		μPD79F7023	μPD79F7024
内部存储器	闪存 (支持自编程)	8 KB	16 KB
	高速 RAM	512 字节	768 字节
存储空间		64 KB	
时钟	主时钟	高速系统 水晶/陶瓷振荡器, 外部时钟输入	1 至 10 MHz: $V_{DD} = 2.7$ 至 5.5 V
		内部高速振荡	4 MHz $\pm$ 3 % ($T_A = -40$ 至 $+85$ °C), 4 MHz $\pm$ 2 % ($T_A = -20$ 至 $+70$ °C)
	内部低速振荡		240 kHz $\pm$ 10 %
通用寄存器		8 位 $\times$ 32 个寄存器(8 位 $\times$ 8 个寄存器 $\times$ 4 个存储体)	
指令集		• 8 位运算, 16 位运算 • 乘/除运算(8 位 $\times$ 8 位, 16 位 $\div$ 8 位) • 位操作(设置, 复位, 测试和布尔操作) • BCD 校正等	
I/O 端口 (总计)		16	
	CMOS I/O	15	
	CMOS 输入	1	
	CMOS 输出	1	
定时器	16 位 (TMO)	1 通道(PPG 输出: 1, 捕捉输入: 2)	
	8 位 (TM5)	1 通道	
	8 位 (TMH)	1 通道 (PWM 输出: 1)	
	看门狗 (WDT)	1 通道	
串行接口		UART	1 通道
8 位 A/D 转换器 $AV_{REF} = 2.7$ 至 5.5 V		5 通道	
运算放大器 (产品附带运算放大器)		2 通道 ($V_{DD} = 2.2$ 至 5.5 V)	
比较器		1 通道	
向量中断内部源	外部	3	
	内部	8	
复位		• 使用 $\overline{RESET}$ 引脚的复位 • 使用看门狗定时器的内部复位 • 使用上电清除的内部复位 • 使用低电压检测电路的内部复位	
On-chip 调试功能		提供	
电源电压		$V_{DD} = 2.7$ 至 5.5 V	
工作环境温度		$T_A = -40$ 至 $+85$ °C	
封装		20 针塑封 SSOP (7.62 mm (300))	

第二章 引脚功能

2.1 引脚功能表

引脚 I/O 缓冲电源分为两类：AV_{REF} 和 V_{DD}。电源和引脚之间的关系分别如下所示：

表 2-1. 引脚 I/O 缓冲电源

电源	对应引脚
AV _{REF}	P20 至 P27
V _{DD}	P20 至 P27 除外的引脚

(1) 端口功能

功能名称	I/O	功能	复位后	复用功能
P20	I/O	端口 2 8 位 I/O 端口 可按 1 位单位指定输入/输出。	模拟输入	ANI0/AMP0-
P21				ANI1/AMP0OUT
P22				ANI2/AMP0+
P23				ANI3
P24				ANI4
P25				AMP1-
P26				AMP1OUT
P27				AMP1+
P30	I/O	端口 3 5 位 I/O 端口 可按 1 位单位指定输入/输出。 通过软件设置，可使用内置上拉电阻。	输出端口	TOH1/TI51/INTP0
P31			模拟输入	TxD0/CMPCOM
P32				RxD0/CMPIN
P33			输入端口	TI000/INTP1
P34				TO00/TI010/CMPOUT
P121	I/O	端口 12 P121,P122 为 2 位 I/O 端口 P125 为 1 位输入专用端口	输出端口	X1/TOOLC0
P122				X2/EXCLK/TOOLD0
P125	输入	仅限 P125 通过软件设置，使用内置上拉电阻。	复位输入	RESET

(2) 非端口功能

功能名称	I/O	功能	复位后	复用功能
ANI0	输入	A/D 转换器的模拟输入	模拟输入	P20/AMP0-
ANI1				P21/AMP0OUT
ANI2				P22/AMP0+
ANI3				P23
ANI4				P24
AMP0-	输入	运算放大器 0 输入	模拟输入	P20/ANI0
AMP0+				P22/ANI2
AMP0OUT	输出	运算放大器 0 输出		P21/ANI1
AMP1-	输入	运算放大器 1 输入	模拟输入	P25
AMP1+				P27
AMP1OUT	输出	运算放大器 1 输出		P26
CMPCOM	输入	比较器公用输入	模拟输入	P31/TxD0
CMPIN	输出	比较器输出		P32/RxD0
CMPOUT	输出	比较器输出	输入端口	P34/TO00/TI010
INTP0	输入	可指定有效沿的外部中断请求输入（上升沿、下降沿、或双边沿）。	输出端口	P30/TOH1/TI51
INTP1				P33/TI000
REGC	—	连接内部操作稳压输出(1.9 V/2.5 V)稳定电容。将电容器连接到 V _{SS} (0.47 至 1 μF)。	—	—
RESET	输入	系统复位输入	复位输入	P125
RxD0	输入	UART0 输入串行数据	输出端口	P32/CMPIN
TxD0	输出	UART0 串行数据输出		P31/CMPCOM
TI000	输入	输入外部计数时钟至 16 位定时器/事件计数器 00 输入捕捉触发至 16 位定时器/事件计数器 00 的捕捉寄存器(CR000, CR010)	输出端口	P33/INTP1
TI010		输入捕捉触发至 16 位定时器/事件计数器 00 的捕捉寄存器 (CR000)		P34/TO00/CMPOUT
TI51	输入	输入外部计数时钟至 8 位定时器/事件计数器 51	输出端口	P30/TOH1/INTP0
TO00	输出	16 位定时器/事件计数器 00 输出	输出端口	P34/TI010/CMPOUT
TOH1	输出	8 位定时器 H1 输出	输出端口	P30/TI51/INTP0
X1	—	连接主系统时钟用谐振器	输出端口	P121/TOOLC0
X2				P122/EXCLK/TOOLD0
EXCLK	输入	主系统时钟用输入外部时钟	输出端口	P122/X2/TOOLD0
V _{DD}	—	端口 2 除外的正电源引脚	—	—
A _V REF		A/D 转换器的参考电压输入和端口 2 和 A/D 转换器的正电源		
V _{SS}	—	接地电位	—	—
TOOLC0	输入	闪存编程/On-chip 调试的时钟输入	输出端口	P121/X1
TOOLD0	I/O	闪存编程/On-chip 调的数据 I/O		P122/X2/EXCLK

2.2 引脚功能描述

2.2.1 P20 至 P27 (端口 2)

P20 至 P27 可用作 I/O 端口，也可用作 A/D 转换器的模拟输入的引脚和运算放大器 I/O。

可按 1 位单位指定以下运行模式。

(1) 端口模式

P20 至 P27 可用作 I/O 端口。使用端口模式寄存器 2(PM2)，可按 1 位单位指定输入端口或输出端口。

(2) 控制模式

P20 至 P27 可用作 A/D 转换器的模拟输入和运算放大器 I/O。

(a) ANI0 至 ANI4

A/D 转换器的模拟输入引脚。用作模拟输入引脚时，参阅 10.6 使用 A/D 转换器的注意事项的(5)ANI0/P20 至 ANI4/P24 引脚。

(b) AMP0+, AMP0-

运算放大器 0 的输入引脚。

(c) AMP0OUT

运算放大器 0 的输出引脚。

(d) AMP1+, AMP1-

运算放大器 1 的输入引脚。

(e) AMP1OUT

运算放大器 1 的输出引脚。

注意事项 复位解除后，ANI0/P20 至 ANI4/P24 转换为模拟输入模式。

2.2.2 P30 至 P34 (端口 3)

P30 至 P34 即可用作 I/O 端口，也可用作外部中断请求输入、定时器 I/O、串行接口的时钟 I/O 和数据 I/O 的引脚。

可按 1 位单位指定以下运行模式。

(1) 端口模式

P30 至 P34 可用作 I/O 端口。通过设置端口模式寄存器 3(PM3)，可按 1 位单元指定输入端口或输出端口。可通过设置上拉电阻选择寄存器 3 (PU3)，使用内置上拉电阻。

(2) 控制模式

P30 至 P34 用作外部中断请求输入、定时器 I/O、时钟 I/O 和串行接口的数据 I/O、比较器 I/O。

(a) INTP0, INTP1

可指定有效沿的外部中断请求输入（上升沿、下降沿或双边沿）。

(b) TI000

16 位定时器/事件计数器 00 的外部计数时钟输入引脚和 16 位定时器/事件计数器 00 的捕捉寄存器(CR000, CR010)的捕捉触发信号输入引脚。

(c) TI010

16 位定时器/事件计数器 00 的捕捉寄存器(CR000)的捕捉触发信号输入引脚。

(d) TO00

16 位定时器/事件计数器 00 的时间输出引脚。

(e) TI51

8 位定时器/事件计数器 51 的外部计数时钟引脚。

(f) TOH1

8 位定时器 H1 的时间输出引脚。

(g) RxD0

串行接口 UART0 的串行数据输入引脚。

(h) TxD0

串行接口 UART0 的串行数据输出引脚。

(i) CMPCOM

比较器共用输入引脚。

(j) CMPIN

比较器输入引脚。

(k) CMPOUT

比较器输出引脚。

2.2.3 P121、P122、P125 (端口 12)

P121、P122 为 I/O 端口。P125 为输出端口。还具有外部低电压检测电位输入、主系统时钟的谐振器连接、主系统时钟的外部时钟输入、外部复位输入、闪存编程器/On-chip 调试的时钟输入和数据 I/O 的功能。

P125/ $\overline{\text{RESET}}$ 作为输入端口使用时，复位端口模式寄存器(RSRMASK)的位 5(RSTM)置“1”；用作外部复位输入时，则将 RSTM 清零。

可按 1 位单位指定以下运行模式。

(1) 端口模式

P121、P122 和 P125 为 I/O 端口。通过设置端口模式寄存器 12(PM12)，可指定 P120 至 P125 的输入或输出端口。只有 P125 可通过设置上拉电阻选择寄存器 12(PU12)，使用内置上拉电阻。

(2) 控制模式

P121、P122 和 P125 具有主系统时钟的谐振器连接、主系统时钟的外部时钟输入、外部复位输入以及 flash 编程器/On-chip 调试的时钟输入和数据 I/O 的功能。

(a) X1, X2

主系统时钟的谐振器连接引脚。

(b) EXCLK

主系统时钟的外部时钟输入引脚。

(c) RESET

低态有效系统复位输入引脚。

(d) TOOLC0

闪存编程器/On-chip 调试的时钟输入引脚。

(e) TOOLD0

闪存编程器/On-chip 调试的数据 I/O 引脚。

注意事项 复位解除后， $\overline{\text{RESET}}$ /P125 设置为外部复位输入。因此，若低电平输入期间复位，则复位状态将会持续到恢复高电平输入为止。

备注 有关将 TOOLC0/X1 和 TOOLD0/X2 连接至 flash 编程器的方法,参阅第二十一章 闪存。有关将 TOOLC0/X1 和 TOOLD0/X2 连接至 On-chip 调试程序仿真器, 参阅第二十二章 On-chip 调试功能。

2.2.4 AVREF、VDD、VSS

电源 / 接地引脚。

(a) AVREF

A/D 转换器的参考电压输入以及端口 2 和 A/D 转换器的正电源引脚。

不使用 A/D 转换器时, 请直接连接到 $V_{DD}^{\#}$ 。

注 端口 2 作为数字端口使用时, 将 AVREF 引脚设置为与 V_{DD} 引脚相同的电位。

(b) VDD

V_{DD} 为正电源引脚。

(c) VSS

V_{SS} 为接地电位引脚[‡]。

注 V_{SS} 兼备 A/D 转换器的接地电位功能。必须将 V_{SS} 连接到稳定的 GND (= 0 V) 上。

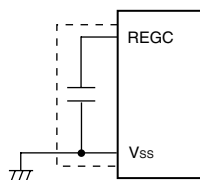
2.2.5 REGC

稳压器的稳定输出电容引脚和内部连接引脚。

(a) REGC

内部运行稳压器输出(1.9 V/2.5 V) 稳定电容连接端口。通过电容器连接至 V_{SS} 。但是，高速内置谐振时钟、外部主系统时钟运行中使用 **STOP** 模式时，推荐使用 $0.47\mu\text{F}$ 。

另外，为了稳定内部电压，请使用性能良好的电容器。



注意事项 请将上图的虚线部分接线尽量缩短到最小。

2.3 有关连接引脚 I/O 电路和未使用引脚的建议

引脚 I/O 电路分类和未使用引脚连接的建议如表 2-1 所示。

各类 I/O 电路配置参阅图 2-1。

表 2-2. 引脚 I/O 电路分类

引脚名称	I/O 电路分类	I/O	未使用引脚的推荐连接方法
ANI0/P20/AMP0-	11-P	I/O	<数字输入设置> 通过寄存器独立连接至AV _{REF} 或V _{SS} 。 <数字输出设置和模拟输入设置> 保持开路状态。 ^{注1}
ANI1/P21/AMP0OUT	11-S		
ANI2/P22/AMP0+	11-N		
ANI3/P23	11-G		
ANI4/P24			
P25/AMP1-	11-P		
P26/AMP1OUT	11-S		
P27/AMP1+	11-N		
P30/TOH1/TI51/INTP0	5-AH		输入： 通过寄存器独立连接至 V _{DD} 或 V _{SS} 。 输出： 保持开路状态。
P31/TxD0/CMPCOM	5-BD		<数字输入设置> 通过寄存器独立连接至AV _{REF} 或V _{SS} 。 <数字输出设置和模拟输入设置> 保持开路状态。 ^{注1}
P32/RxD0/CMPIN	5-BB		
P33/TI000/INTP1	5-AH		输入： 通过寄存器独立连接至 V _{DD} 或 V _{SS} 。 输出： 保持开路状态。
P34/TO00/TI010/CMPOUT			
P121/X1/TOOLC0 ^{注2}	37 (H)		
P122/X2/EXCLK/TOOLD0 ^{注2}			
RESET/P125	42-B	输入	通过寄存器或直接连接至 V _{DD} 。
AV _{REF}			直接连接至 V _{DD} 。

注 1. 该引脚在模拟输入设置下打开时，则会导致输入电压不稳定。建议在数字输出设定下打开。

2. 不使用这些引脚时，建议采用上述输入端口模式（参阅图 5-2 时钟操作模式选择寄存器(OSCCTL)的格式）的连接方法。

注意事项 1. 复位解除后，ANI0/P20/AMP0-、ANI1/P21/AMP0OUT、ANI2/P22/AMP0+、ANI3/P23ANI4/P24 为模拟输入模式。

2. 复位解除后，RESET/P125 设置为外部复位输入，因此，若低电平输入期间复位，则复位状态将会持续到恢复高电平输入为止。

图 2-2. 引脚 I/O 电路表 (1/2)

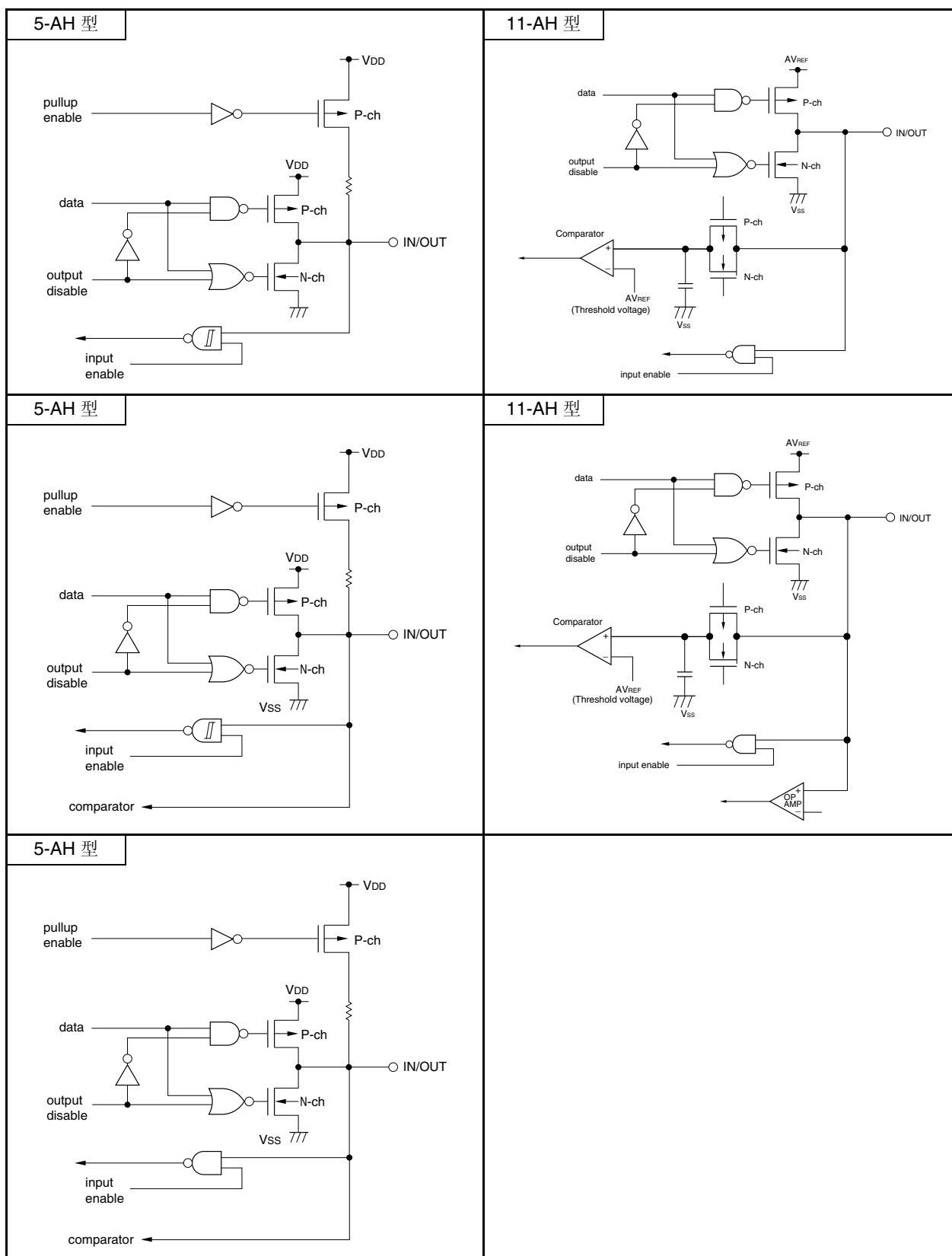
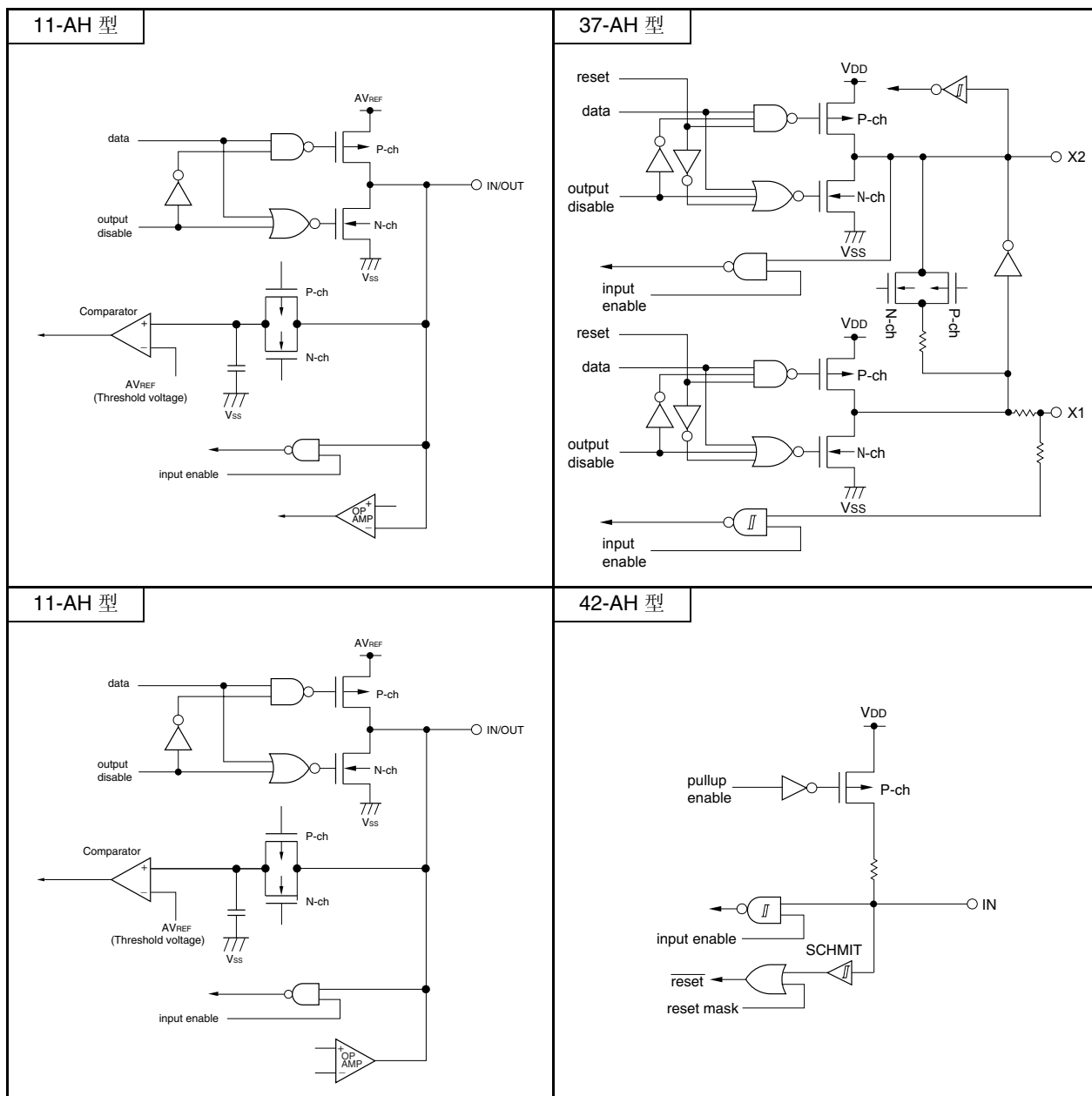


图 2-2. 引脚 I/O 电路表 (2/2)



第三章 CPU 架构

3.1 存储器空间

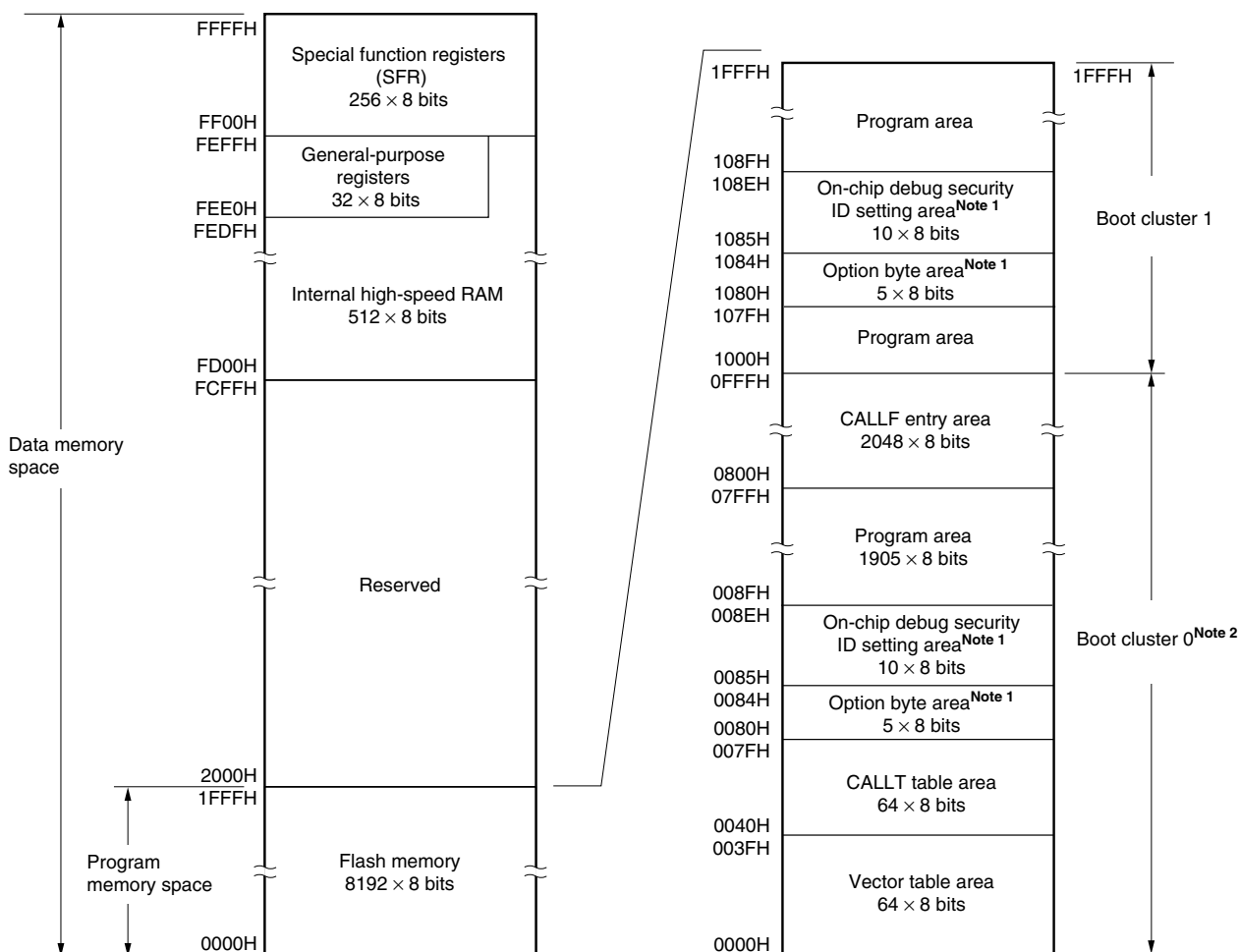
μPD79F7023, 79F7024 产品可以访问 64KB 存储空间。存储器映射图如图 3-1、3-2 所示。

注意事项 由于复位时 ROM 区域的设置值不确定，因此，复位解除后，必须在每个产品上设置以下值。

表 3-1. 存储器容量切换寄存器(IMS)的设置值

产品	IMS	ROM 容量	内部高速 RAM 容量
μPD79F7023	42H	8 KB	512 字节
μPD79F7024	04H	16 KB	768 字节

图 3-1. 存储器映射图 (μPD79F7023)



- 注 1.** 不使用引导交换功能时: 在 0080H 至 0084H 之间设置选项字节, 在 0085H 至 008EH 之间设置 On-chip 调试安全 ID。
- 使用引导交换功能时: 在 0080H 至 0084H 之间和 1080H 至 1084H 之间设置选项字节, 在 0085H 至 008EH 之间和 1085H 至 108EH 之间设置 On-chip 调试安全 ID。
- 2.** 通过安全设置, 可以禁止写入引导集群 0。(参阅 21.6 安全设置)。

备注 闪存分成若干个单元块(1 块=1KB)。有关地址值和块号, 参阅表 3-2 闪存地址值与块号之间的对应。

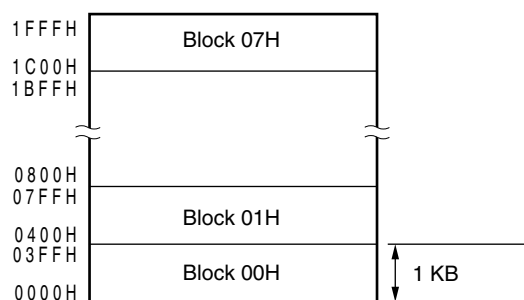
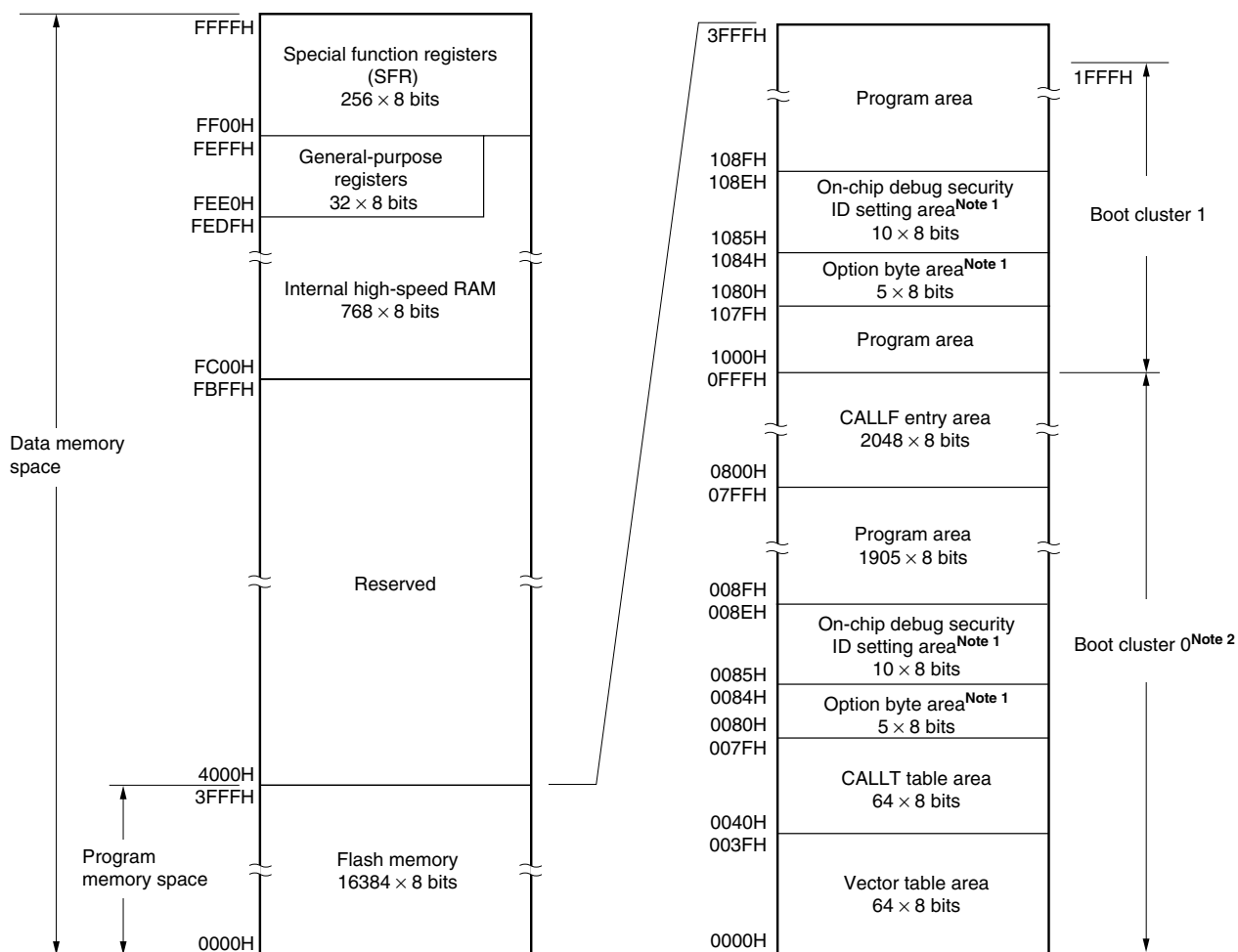


图 3-2. 存储器映射图 (μPD79F7024)

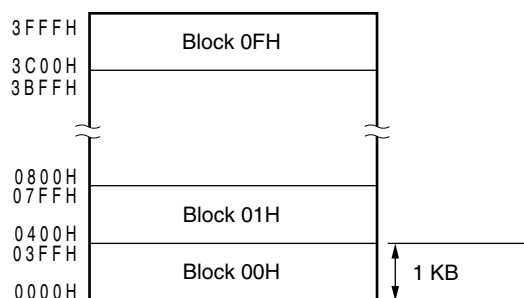


注 1. 不使用引导交换功能时: 在 0080H 至 0084H 之间设置选项字节, 在 0085H 与 008EH 之间设置 On-chip 调试安全 ID。

使用引导交换功能时: 在 0080H 至 0084H 之间和 1080H 至 1084H 之间设置选项字节, 在 0085H 至 008EH 之间和 1085H 至 108EH 之间设置 On-chip 调试安全 ID。

2. 通过安全设置,可以禁止写入引导集群 0。(参阅 21.6 安全设置)。

备注 闪存分成若干个单元块(1 块=1KB)。有关地址值和块号, 参阅表 3-2 闪存地址值与块号之间的对应。



闪存地址值与块号之间的对应如下所示。

表 3-2. 闪存地址值与块号之间的对应

地址值	块号
0000H 至 03FFH	00H
0400H 至 07FFH	01H
0800H 至 0BFFH	02H
0C00H 至 0FFFH	03H
1000H 至 13FFH	04H
1400H 至 17FFH	05H
1800H 至 1BFFH	06H
1C00H 至 1FFFH	07H
2000H 至 23FFH	08H
2400H 至 27FFH	09H
2800H 至 2BFFH	0AH
2C00H 至 2FFFH	0BH
3000H 至 33FFH	0CH
3400H 至 37FFH	0DH
3800H 至 3BFFH	0EH
3C00H 至 3FFFH	0FH

备注 μPD79F7023: 块号 00H 至 07H
 μPD79F7024: 块号 00H 至 0FH

3.1.1 内部程序存储器空间

内部程序存储器空间用于存储程序和表数据。一般由程序计数器(PC)进行寻址。

μPD79F7023, 79F7024 内置的内部 ROM(闪存)如下所示。

表 3-3.内部 ROM 容量

产品	内部 ROM	
	结构	容量
μPD79F7023	闪存	8192 × 8 位(0000H 至 1FFFH)
μPD79F7024		16384 × 8 位 (0000H 至 3FFFH)

内部程序存储器空间分为以下区域。

(1) 向量表区域

0000H 至 003FH 的 64 字节区域是预留的向量表区域。向量表区域根据复位或产生的各种中断请求进行转移的程序起始地址。

在 16 位地址中，低 8 位用作偶数地址，高 8 位用作奇数地址。

表 3-4. 向量表

向量表地址	中断源
0000H	RESET 输入, POC, LVI, WDT
0004H	INTLVI
0006H	INTP0
0008H	INTP1
000AH	INTCMP
0014H	INTSR0
0016H	INTST0
001AH	INTTMH1
0020H	INTTM000
0022H	INTTM010
0024H	INTAD
002AH	INTTM51
003EH	BRK

(2) CALLT 指令表区域

0040H 至 007FH 的 64 字节区域，可存放 1 字节调用指令(CALLT)的子程序入口地址。

(3) 选项字节区域

0080H 至 0084H、1080H 至 1084H 的各 5 字节区域可用作选项字节区域。不使用引导交换功能时，在 0080H 至 0084H 设置选项字节；使用引导交换功能时，则在 0080H 至 0084H 和 1080H 至 1084H 设置选项字节。详阅第二十章选项字节。

(4) On-chip 调试安全 ID 设置区域

0085H 至 008EH、1085H 至 108EH 的各 10 字节可用作 On-chip 调试安全 ID 设定区域。不使用引导交换功能时，在 0085H 至 008EH 区域设置 10 字节的 On-chip 调试安全 ID；使用引导交换功能时，则在 0085H 至 008EH 和 1085H 至 108EH 区域设置 10 字节的 On-chip 调试安全 ID。详情参阅第二十二章 On-chip 调试功能。

(5) CALLT 指令入口区域

0800H 至 0FFFH 的区域，通过一个 2 字节的调用指令(CALLF)可直接进行子程序调用。

3.1.2 内部数据存储器空间

μPD79F7023, 79F7024 内置以下 RAM。

(1) 内部高速 RAM

表 3-5. 内部高速 RAM 容量

产品	内部高速 RAM 容量
μPD79F7023	512 × 8 位 (FD00H 至 FEFH)
μPD79F7024	768 × 8 位 (FC00H 至 FEFH)

FEE0H 至 FEFH 的 32 字节区域分配给 4 个通用寄存器存储体，每个都由 8 台 8 字节寄存器组成。

该区域禁止用作程序区域进行指令的写入。

堆栈存储器使用内部高速 RAM。

3.1.3 特殊功能寄存器(SFR: Special Function Register)区域

On-chip 外围硬件的特殊功能寄存器(SFR)分配在 FF00H 至 FFFFH 的区域(参阅 3.2.3 特殊功能寄存器(SFR: Special Function Register) 的表 3-6 特殊功能寄存器)。

注意事项 禁止访问 SFRs 未分配的地址。

3.1.4 数据存储寻址

指定将要执行的下一条指令地址，或者指定执行指令相关的寄存器或存储器地址的方法，称之为寻址。

μPD79F7023, 79F7024 微控制器考虑到存储器执行指令寻址的可操作性，提供了多种寻址模式。尤其在内置数据存储区域，可利用特殊功能寄存器(SFR)和通用寄存器的特殊寻址方法。数据存储与寻址的对应如图 3-3 和 3-4 所示。有关各种寻址模式，详情参阅 3.4 操作数地址寻址。

图 3-3. 数据存储与寻址之间的对应(μPD79F7023)

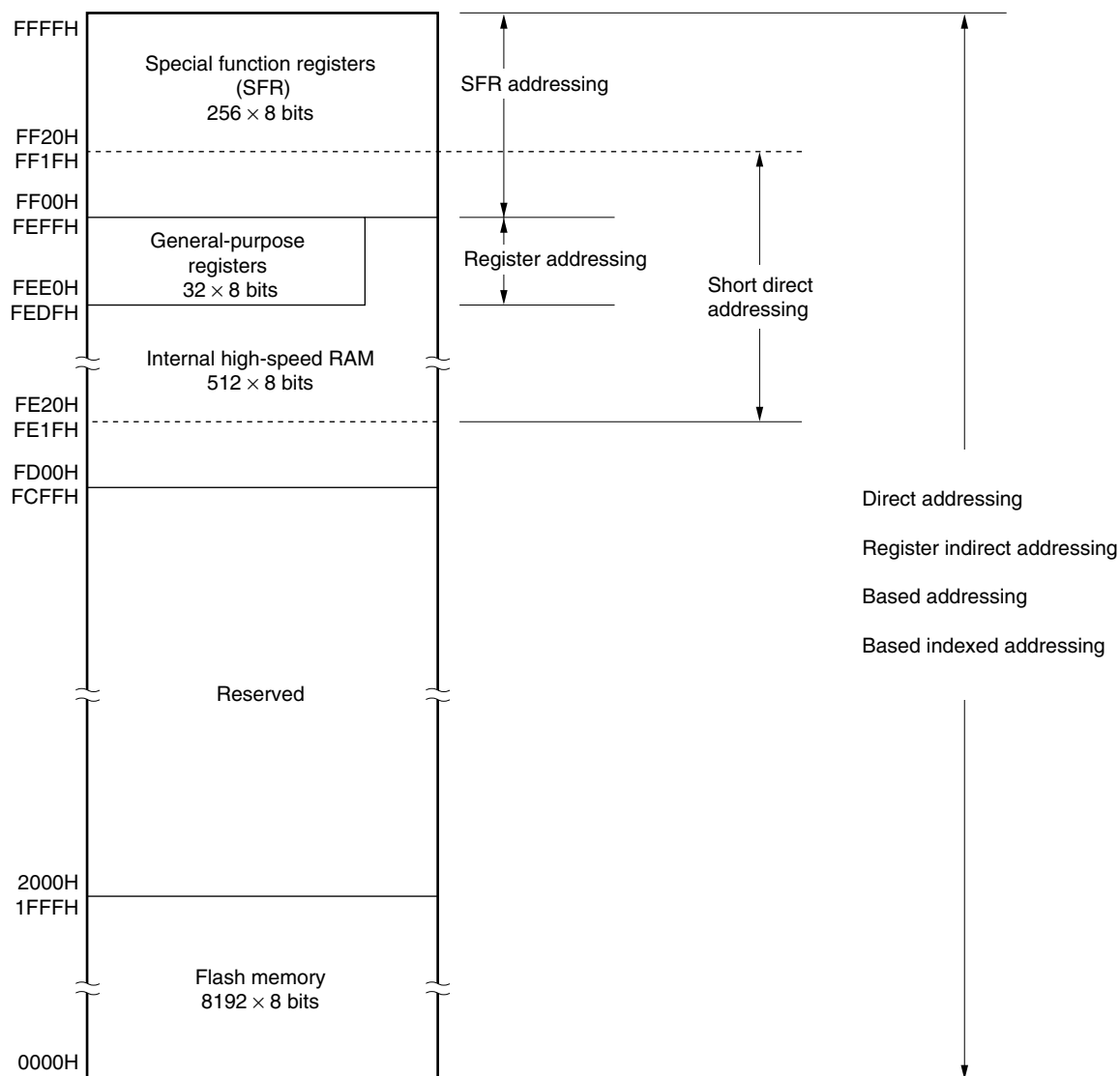
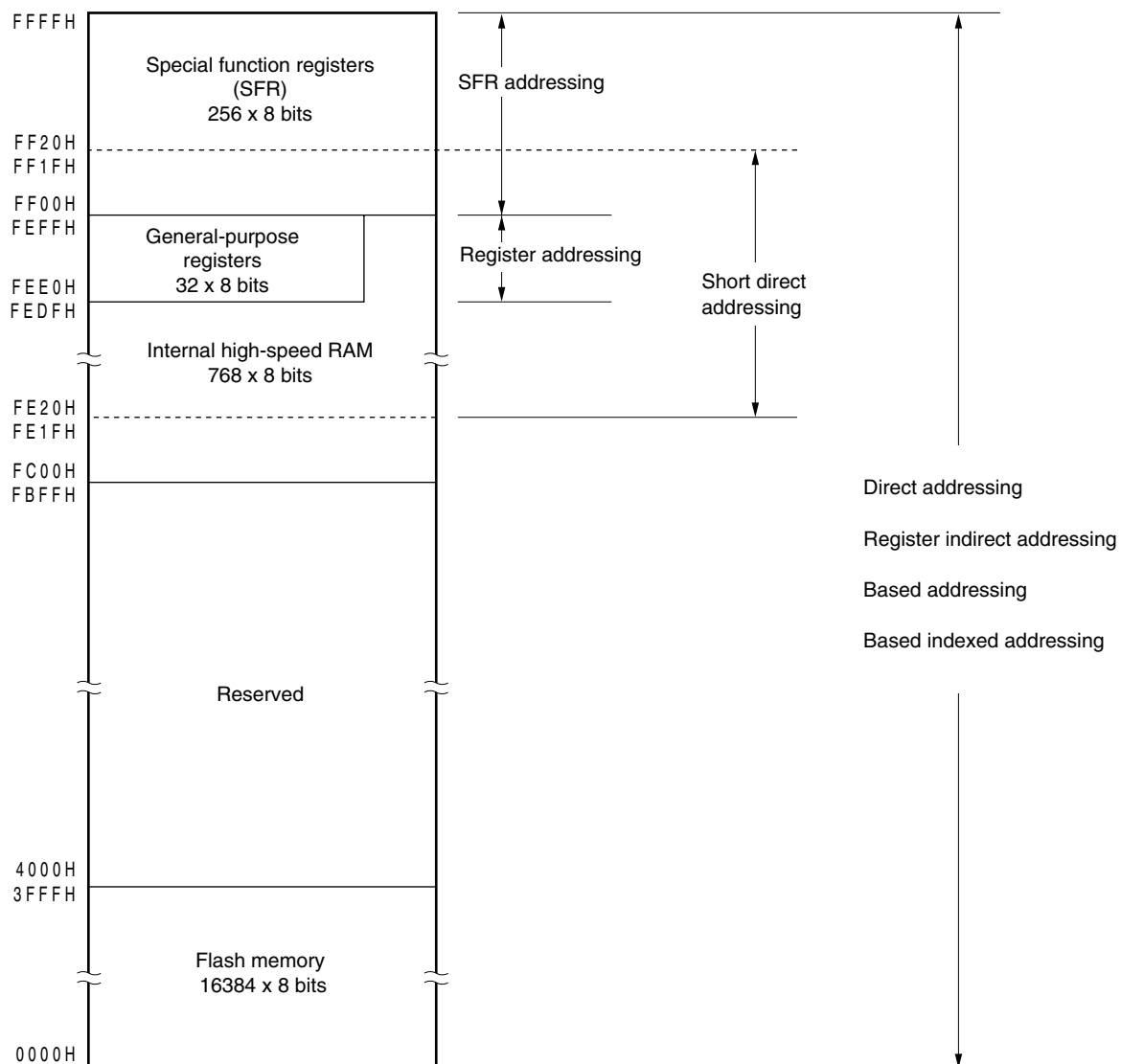


图 3-4. 数据存储器与寻址之间的对应 (μPD79F7024)



3.2 处理程序寄存器

μPD79F7023, 79F7024 内置于以下处理程序寄存器。

3.2.1 控制寄存器

控制寄存器具有控制程序序列、状态和堆栈存储的专用功能。控制寄存器是由程序计数器(PC)、程序状态字(PSW)和堆栈指针(SP)组成。

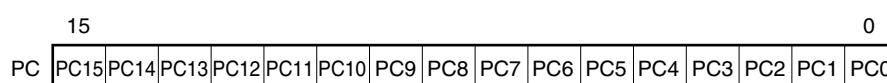
(1) 程序计数器(PC)

程序计数器为 16 位寄存器，用于存放将要执行的下一条程序地址信息。

正常操作下，根据获取的指令位数，程序计数器(PC)的值会自动累加。当执行转移指令时，设置立即数和寄存器内容。

复位信号产生时，程序计数器设置为 0000H 和 0001H 地址的复位向量表的值。

图 3-5. 程序计数器的格式



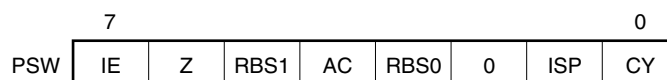
(2) 程序状态字 (PSW)

程序状态字是 8 位寄存器，由通过指令执行设置或复位的各种标识组成。

通过响应向量中断请求或执行 PUSH PSW 指令，将程序状态字的内容入栈；通过执行 RETB、RETI 和 POP PSW 指令，将程序状态字的内容出栈。

复位信号产生时，程序状态字置为“02H”。

图 3-6. 程序状态字的格式



(a) 中断允许标志 (IE)

该标志用于控制 CPU 响应中断请求操作。

IE 标志为 0 时，表示禁止中断(DI)状态，即禁止所有可屏蔽中断请求。

IE 标志为 1 时，表示允许中断(EI)状态。通过控制优先服务标志(ISP)、各种中断源相对应的中断屏蔽标志和优先级指定标志，响应中断请求。

执行 DI 指令或中断请求得以响应时，该标志将复位(0)；执行 EI 指令时，该标志将设置(1)。

(b) 零标志(Z)

操作结果为 0 时，该标志将设置(1)。其他情况下，该标志将复位(0)。

(c) 寄存器存储体选择标志(RBS0, RBS1)

用于选择 4 个寄存器存储体中的一个 2 位标志。

标志位中 2 位信息用来指定执行 SEL RBn 指令时所选的寄存器存储体。

(d) 辅助进位标志 (AC)

若操作结果在第 3 位有进位或在第 3 位有借位，则该标志置“1”。其他情况下，该标志清“0”。

(e) 优先服务标志(ISP)

该标志用来管理可屏蔽向量中断响应的优先级。该标志为 0 时，不响应通过优先级指定标志寄存器 (PR0L, PR0H, PR1L)指定的低级别向量中断请求 (参阅 14.3(3)优先级指定标志寄存器 (PR0L, PR0H, PR1L))。由中断允许标志(IE)控制请求的实际响应。

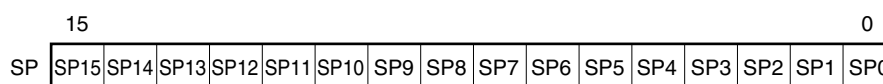
(f) 进位标志(CY)

存放执行加减运算指令时出现的进位和错位。存放执行循环指令中的转移位，并可在执行位操作指令中用作位累加器。

(3) 堆栈指针 SP

堆栈指针为 16 位寄存器，用于存放存储器堆栈区的起始地址。只有内部高速 RAM 区域可设置为堆栈区。

图 3-7. 堆栈指针的格式



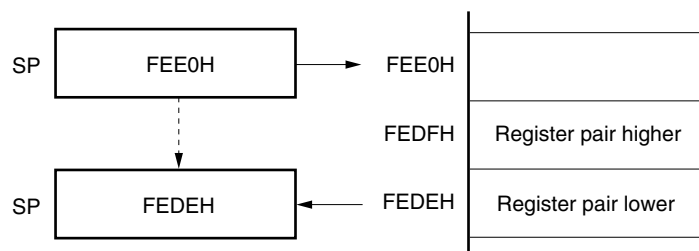
将数据写入(存储)堆栈时，堆栈指针将递减；而从堆栈中读取(恢复)数据时，堆栈指针将累加。

通过各堆栈操作的存储/恢复数据如图 3-8 和 3-9 所示。

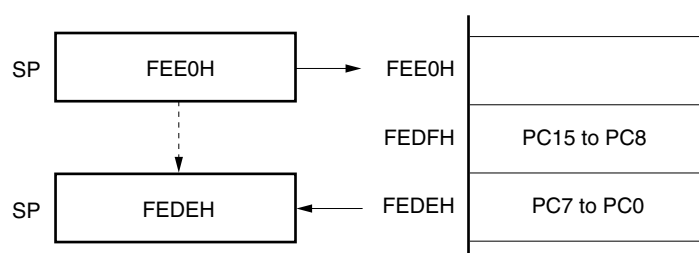
注意事项 因为复位信号产生时的 SP 内容不确定，所以必须在使用堆栈之前进行 SP 初始化。

图 3-8. 将数据存入堆栈存储器

(a) PUSH rp 指令 (SP = FEE0H 时)



(b) CALL, CALLF, CALLT 指令 (SP = FEE0H 时)



(c) 中断, BRK 指令 (SP = FEE0H 时)

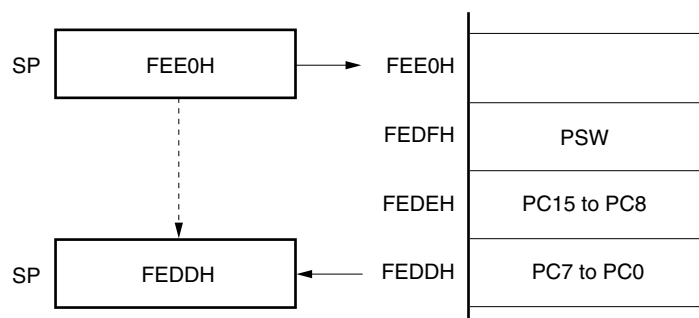
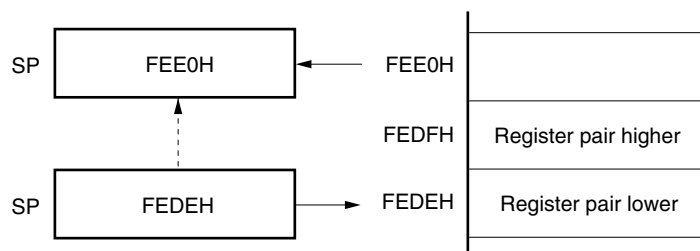
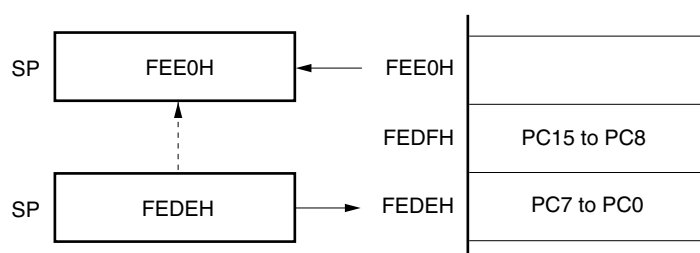


图 3-9. 从堆栈存储器读取数据

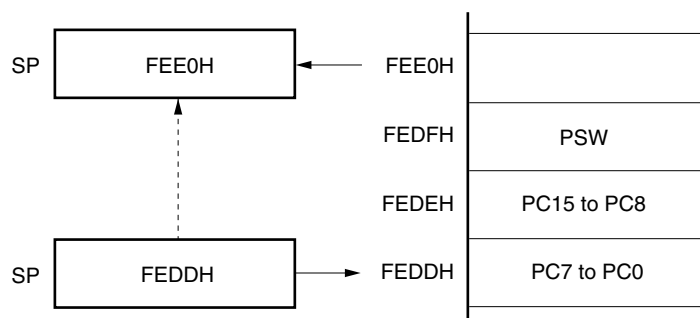
(a) POP rp 指令 (SP = FEDEH 时)



(b) RET 指令 (SP = FEDEH 时)



(c) RETI, RETB 指令 (SP = FEDDH 时)



3.2.2 通用寄存器

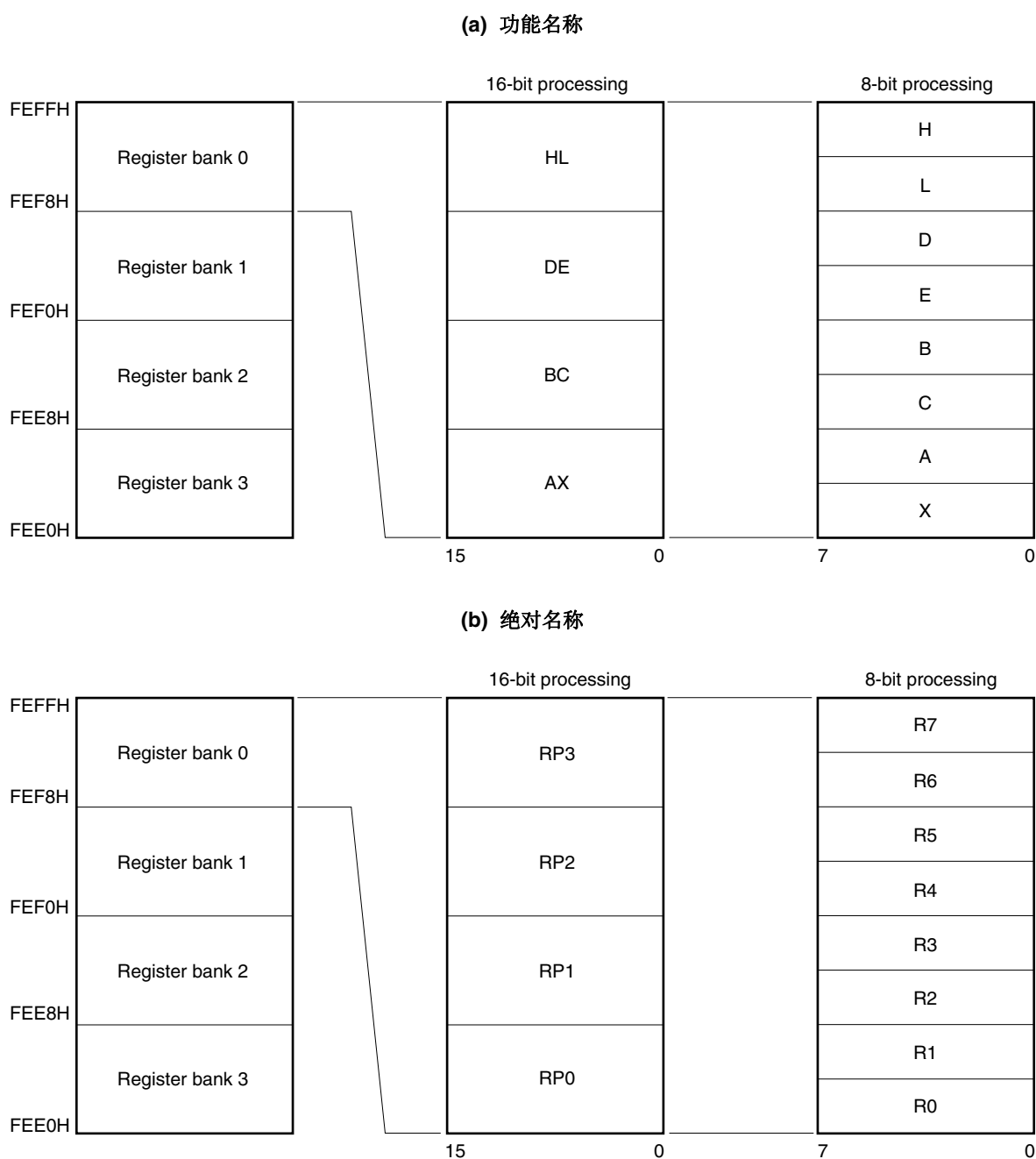
通用寄存器映射到数据存储器的特定地址(FEE0H 至 FEFH)。通用寄存器有 4 个存储体，每个存储体由 8 台 8 位寄存器 (X, A, C, B, E, D, L 和 H)组成。

每台寄存器可用作一台 8 位寄存器，一对可用作 16 位寄存器(AX, BC, DE 和 HL)。

描述通用寄存器时，可使用功能名称 (X, A, C, B, E, D, L, H, AX, BC, DE 和 HL)和绝对名称(R0 至 R7 和 RP0 至 RP3)。

由 CPU 控制指令(SEL RBn)可设置用于指令执行的寄存器存储体。4 台寄存器存储体结构，通过将一台正常处理的寄存器切换成一台存储体中断处理的寄存器，可创建一个高效程序。

图 3-10. 通用寄存器结构



3.2.3 特殊功能寄存器 (SFR: Special Function Register)

特殊功能寄存不同于通用寄存器，具有各种特定功能。

特殊功能寄存器分配在 FF00H 至 FFFFH 字节的区域。

特殊功能寄存器可以与通用寄存器一样用运行指令、传送指令以及位操作指令进行操作。可操作位单位根据特殊功能寄存器的类型不同而不同，可以是 1 位、8 位或 16 位。

各种位单位操作指定如下。

- 1 位操作

描述在 1 位操作指令的操作数(sfr.bit)中使用汇编程序保留的符号。

也可指定地址。

- 8 位操作

描述在 8 位操作指令的操作数(sfr)中使用汇编程序保留的符号。

也可指定地址。

- 16 位操作

描述在 16 位操作指令的操作数(sfrp)中使用汇编程序保留的符号。

指定地址时，描述为偶地址。

表 3-6 特殊功能寄存器一览表表格中各项目的含义如下：

- 符号

符号表示特殊功能寄存器地址。在 RA78K0，用作保留字；在 CC78K0，指定为 #pragma sfr 指令用作 sfr 变量。用作 RA78K0、ID78K0-QB 和系统仿真器时，符号可描述为指令操作数。

- R/W

表示该特殊功能寄存器可否读/写。

R/W: 可读/写

R: 只读

W: 只写

- 可操作位单位

表示可操作位单位(1、8 或 16)。“-”表示不可操作位单位。

- 复位后

表示复位信号发生时各寄存器的状态。

表 3-6. 特殊功能寄存器 (1/4)

地址	符号	位编号								R/W	可操作位单位编号			复位后	参见页码
		7	6	5	4	3	2	1	0		1	8	16		
FF00H	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
FF01H	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
FF02H	P2	P27	P26	P25	P24	P23	P22	P21	P20	R/W	√	√	—	00H	
FF03H	P3	0	0	0	P34	P33	P32	P31	P30	R/W	√	√	—	00H	
FF04H 至 FF08H	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
FF09H	ADCRH	—	—	—	—	—	—	—	—	R	—	√	—	00H	
FF0AH	RXB0	—	—	—	—	—	—	—	—	R	—	√	—	FFH	
FF0BH	TXS0	—	—	—	—	—	—	—	—	R/W	—	√	—	FFH	
FF0CH	P12	0	0	P125	0	0	P122	P121	0	R/W	√	√	—	00H	
FF0DH	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
FF0EH	ADS	0	0	0	0	0	<ADS2>	<ADS1>	<ADS0>	R/W	√	√	—	00H	
FF0FH	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
FF10H	TM00	—	—	—	—	—	—	—	—	R	—	—	√	0000H	
FF11H		—	—	—	—	—	—	—	—		—	—	—	—	—
FF12H	CR000	—	—	—	—	—	—	—	—	R/W	—	—	√	0000H	
FF13H		—	—	—	—	—	—	—	—		—	—	—	—	—
FF14H	CR010	—	—	—	—	—	—	—	—	R/W	—	—	√	0000H	
FF15H		—	—	—	—	—	—	—	—		—	—	—	—	—
FF16H 至 FF19H	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
FF1AH	CMP01	—	—	—	—	—	—	—	—	R/W	—	√	—	00H	
FF1BH	CMP11	—	—	—	—	—	—	—	—	R/W	—	√	—	00H	
FF1CH 至 FF1EH	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
FF1FH	TM51	—	—	—	—	—	—	—	—	R	—	√	—	00H	
FF20H	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
FF21H	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
FF22H	PM2	PM27	PM26	PM25	PM24	PM23	PM22	PM21	PM20	R/W	√	√	—	FFH	
FF23H	PM3	1	1	1	PM34	PM33	PM32	PM31	PM30	R/W	√	√	—	FFH	
FF24H 至 FF2AH	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

备注 标有(<>)符号的位名,在 RA78K0 指定为保留字; 在 CC78K0 指定为#pragma sfr 用作 sfr 变量。

表 3-6. 特殊功能寄存器 (2/4)

地址	符号	位编号								R/W	可操作位单位编号			复位后	参见页码
		7	6	5	4	3	2	1	0		1	8	16		
FF2BH	FPCTL	0	0	0	0	0	0	0	<FLMD PUP>	R/W	√	√	—	00H	
FF2CH	PM12	1	1	PM125	1	1	PM122	PM121	1	R/W	√	√	—	FFH	
FF2DH	RSTMASK	0	0	RSTM	0	0	0	0	0	R/W	√	√	—	00H	
FF2EH 至 FF32H	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
FF33H	PU3	0	0	0	PU34	PU33	PU32	PU31	PU30	R/W	√	√	—	00H	
FF34H 至 FF3BH	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
FF3CH	PU12	0	0	PU125	0	0	0	0	0	R/W	√	√	—	20H	
FF3DH	RMC	—	—	—	—	—	—	—	—	R/W	—	√	—	00H	
FF3EH 至 FF47H	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
FF48H	EGP	0	0	0	0	0	EGP2	EGP1	EGP0	R/W	√	√	—	00H	
FF49H	EGN	0	0	0	0	0	EGN2	EGN1	EGN0	R/W	√	√	—	00H	
FF4AH 至 FF4FH	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
FF50H	ASIM0	<POWE R0>	<TXE0>	<RXE0>	PS01	PS00	CL0	SL0	1	R/W	√	√	—	01H	
FF51H	BRGC0	TPS01	TPS00	0	MDL04	MDL03	MDL02	MDL01	MDL00	RW	—	√	—	FFH	
FF52H	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
FF53H	ASIS0	0	0	0	0	0	PE0	FE0	OVE0	R	—	√	—	00H	
FF54H 至 FF5FH	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
FF60H	AMPM	<OPA MP0E>	0	0	0	<OPA MP1E>	0	0	0	R/W	√	√	—	00H	
FF61H	CMPCTL	<CMP 0EN>	<CDFS 1>	<CDFS 0>	<CMPO UTEN>	<CRE GSEL>	<CFLG>	<COUT EN>	<CINV>	RW	√	√	—	00H	
FF62H	CMPPC	0	0	0	0	0	0	<CMPC OMPC>	<CMPIN PC>	R/W	√	√	—	00H	
FF63H 至 FF6FH	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

备注 标有(<>)符号的位名,在 RA78K0 指定为保留字; 在 CC78K0 指定为#pragma sfr 用作 sfr 变量。

表 3-6. 特殊功能寄存器 (3/4)

地址	符号	位编号								R/W	可操作位单位编号			复位后	参见页码
		7	6	5	4	3	2	1	0		1	8	16		
FF70H	TMHMD1	<TMHE1>	CKS12	CKS11	CKS10	TMMD11	TMMD10	<TOLEV1>	<TOEN1>	R/W	√	√	—	00H	
FF71H	TMCYC1	0	0	0	0	0	RMC1	NRZB1	<NRZ1>	R/W	√	√	—	00H	
FF72H 至 FF85H	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
FF86H	TMC00	0	0	0	0	TMC003	TMC002	TMC001	<OVF00>	R/W	√	√	—	00H	
FF87H	PRM00	ES110	ES100	ES010	ES000	0	0	PRM001	PRM000	R/W	√	√	—	00H	
FF88H	CRC00	0	0	0	0	0	CRC002	CRC001	CRC000	R/W	√	√	—	00H	
FF89H	TOC00	0	<OSPT00>	<OSPE00>	TOC004	<LVS00>	<LVR00>	TOC001	<TOE00>	R/W	√	√	—	00H	
FF8AH 至 FF8FH	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
FF90H	ADM	<ADCS>	0	FR2	FR1	FR0	LV1	LV0	<ADCE>	R/W	√	√	—	00H	
FF91H 至 FF96H	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
FF97H	ADPC	ADPC7	ADPC6	ADPC5	ADPC4	ADPC3	ADPC2	ADPC1	ADPC0	R/W	√	√	—	00H	
FF98H	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
FF99H	WDTE	—	—	—	—	—	—	—	—	RW	—	√	—	1AH/ 9AH ^{Note1}	
FF9AH 至 FF9EH	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
FF9FH	OSCCTL	<EXCLK>	<OSCSEL>	0	0	0	0	0	0	RW	√	√	—	00H	
FFA0H	RCM	<RSTS>	0	0	0	0	0	<LSRSTOP>	<RSTOP>	RW	√	√	—	80H ^{Note2}	
FFA1H	MCM	0	0	0	0	0	<XSEL>	<MCS>	<MCM0>	R/W	√	√	—	00H	
FFA2H	MOC	<MSTOP>	0	0	0	0	0	0	0	RW	√	√	—	80H	
FFA3H	OSTC	0	0	0	MOST11	MOST13	MOST14	MOST15	MOST16	R	√	√	—	00H	
FFA4H	OSTS	0	0	0	0	0	OSTS2	OSTS1	OSTS0	RW	—	√	—	05H	
FFA5H 至 FFABH	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

注 1. WDTE 复位值取决于选项字节的设置。

2. 复位解除时，该寄存器值为 00H；等到内置高速振荡器的振荡恢复准确稳定时，自动切换到 80H。

备注 标有(<>)符号的位名,在 RA78K0 指定为保留字；在 CC78K0 指定为#pragma sfr 用作 sfr 变量。

表 3-6. 特殊功能寄存器 (4/4)

地址	符号		位编号								R/W	可操作位单位编号			复位后	参见页码
			7	6	5	4	3	2	1	0		1	8	16		
FFACH	RESF		0	0	0	WDTRF	0	0	0	LVIRF	R	—	√	—	00H ^{Note1}	
FFADH 至 FFB0H	—		—	—	—	—	—	—	—	—	—	—	—	—	—	—
FFB1H	CR51		—	—	—	—	—	—	—	—	R/W	—	√	—	00H	
FFB2H	TCL51		0	0	0	0	0	TCL512	TCL511	TCL510	R/W	√	√	—	00H	
FFB3H	TMC51		<TCE51>	TMC516	0	0	0	0	TMC511	0	R/W	√	√	—	00H	
FFB4H 至 FFBDH	—		—	—	—	—	—	—	—	—	—	—	—	—	—	—
FFBEH	LVIM		<LVION>	0	0	0	0	0	<LVIMD>	<LVIF>	R/W	√	√	—	00H ^{Note2}	
FFBFH	LVIS		0	0	0	0	LVIS3	LVIS2	LVIS1	LVIS0	R/W	√	√	—	00H ^{Note3}	
FFC0H 至 FFDFH	—		—	—	—	—	—	—	—	—	—	—	—	—	—	—
FFE0H	IF0	IF0L	0	0	0	0	<CMPIF>	<PIF1>	<PIF0>	<LVIIF>	R/W	√	√	√	00H	
FFE1H		IF0H	<TMIF010>	<TMIF000>	0	0	<TMIFH1>	0	<STIF0>	<SRIF0>	R/W	√	√		00H	
FFE2H	IF1	IF1L	0	0	0	0	<TMIF51>	0	0	<ADIF>	R/W	√	√	√	00H	
FFE3H		—	0	0	0	0	0	0	0	0	—	—	—		00H	
FFE4H	MK0	MK0L	1	1	1	1	<CMPMK>	<PMK1>	<PMK0>	<LVIMK>	R/W	√	√	√	FFH	
FFE5H		MK0H	<TMMK010>	<TMMK000>	1	1	<TMMKH1>	1	<STMK0>	<SRMK0>	R/W	√	√		FFH	
FFE6H	MK1	MK1L	1	1	1	1	<TMMK51>	1	1	<ADMK>	R/W	√	√	√	FFH	
FFE7H		—	1	1	1	1	1	1	1	1	—	—	—		FFH	
FFE8H	PR0	PR0L	1	1	1	1	<CMPPR>	<PPR1>	<PPR0>	<LVIPR>	R/W	√	√	√	FFH	
FFE9H		PR0H	<TMPR010>	<TMPR000>	1	1	<TMPRH1>	1	<STPR0>	<SRPR0>	R/W	√	√		FFH	
FFEAH	PR1	PR1L	1	1	1	1	<TMPR51>	1	1	<ADPR>	R/W	√	√	√	FFH	
FFEBH		—	1	1	1	1	1	1	1	1	—	—	—		FFH	
FFECH 至 FFEFH	—		—	—	—	—	—	—	—	—	—	—	—	—	—	—
FFF0H	IMS		RAM2	RAM1	RAM0	0	ROM3	ROM2	ROM1	ROM0	R/W	—	√	—	CFH ^{Note4}	
FFF1H 至 FFFAH	—		—	—	—	—	—	—	—	—	—	—	—	—	—	—
FFFBH	PCC		0	0	0	0	0	PCC2	PCC1	PCC0	R/W	√	√	—	01H	

- 注 1. RESF 复位值取决于复位源。
2. LVIM 复位值取决于复位源和选项字节的设置。
3. LVIS 复位值取决于复位源。
4. 复位信号产生时, ROM 区域处于不稳定状态。因此, 复位解除后, 必须按照表 3-1 设置每个产品的对应值。

备注 标有(<>)符号的位名,在 RA78K0 指定为保留字; 在 CC78K0 指定为#pragma sfr 用作 sfr 变量。

3.3 指令地址寻址

指令地址取决于程序计数器(PC)的内容。一般情况下，根据执行指令时所获取的下一条指令字节数，程序计数器(PC)的内容自动递增(每个字节+1)。执行转移指令时，转移目的信息设置在程序计数器(PC)上，并按以下寻址转移，参阅 **78K/0 系列 用户手册 指令篇(U12326E)**。

3.3.1 相对寻址

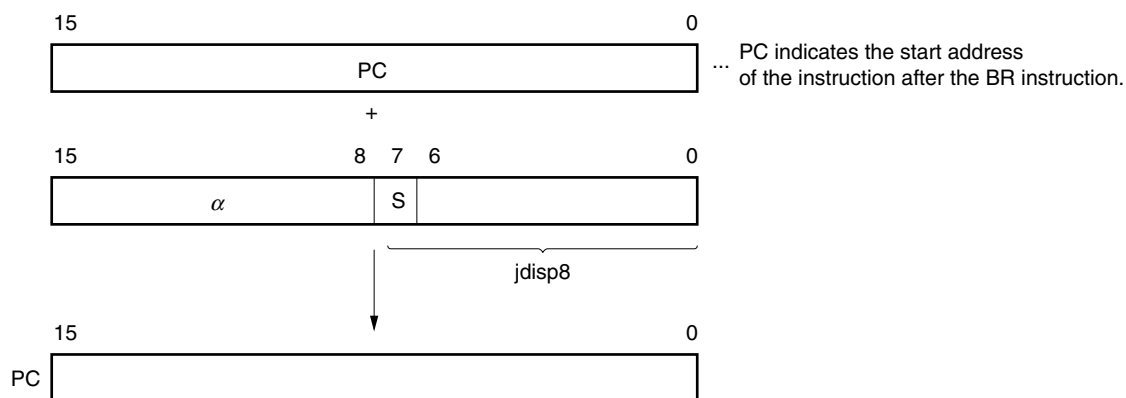
[功能]

将一条指令码的 8 位立即数（偏移量：jdisp8）与下一条指令的起始地址相加，结果赋给程序计数器(PC)后，并转移到其相应的地址。偏移量是带 2 个符号的补码数(−128 至 +127)，其中第 7 位是符号位。

换言之，在相对寻址中，相对转移值是从下一条指令的起始地址开始，范围在 −128 至 +127 之间。

当执行 BR \$addr16 指令或条件转移指令时，将执行该功能。

[图示]



When S = 0, all bits of α are 0.
When S = 1, all bits of α are 1.

3.3.2 立即寻址

[功能]

将指令字中的立即数赋给程序计数器(PC)，并转移到其相应地址。

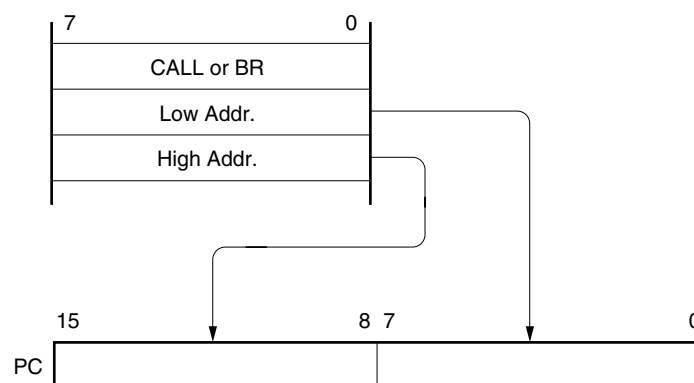
当执行 CALL !addr16、BR !addr16 或 CALLF !addr11 指令时，执行该功能。

CALL !addr16 和 BR !addr16 可转移到所有的存储器空间。

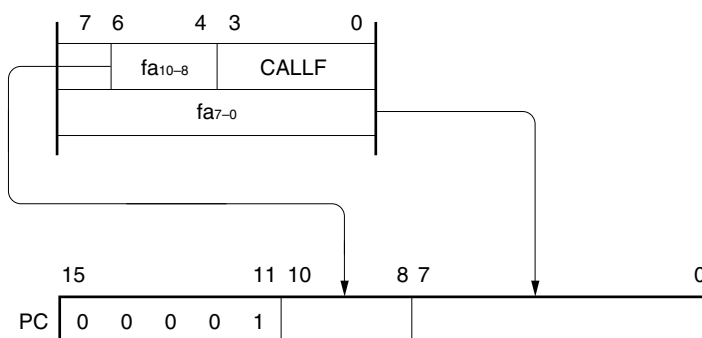
CALLF !addr11 指令可转移到 0800H 至 0FFFH 区域。

[图示]

CALL !addr16 和 BR !addr16 指令示例



CALLF !addr11 指令示例



3.3.3 表间接寻址

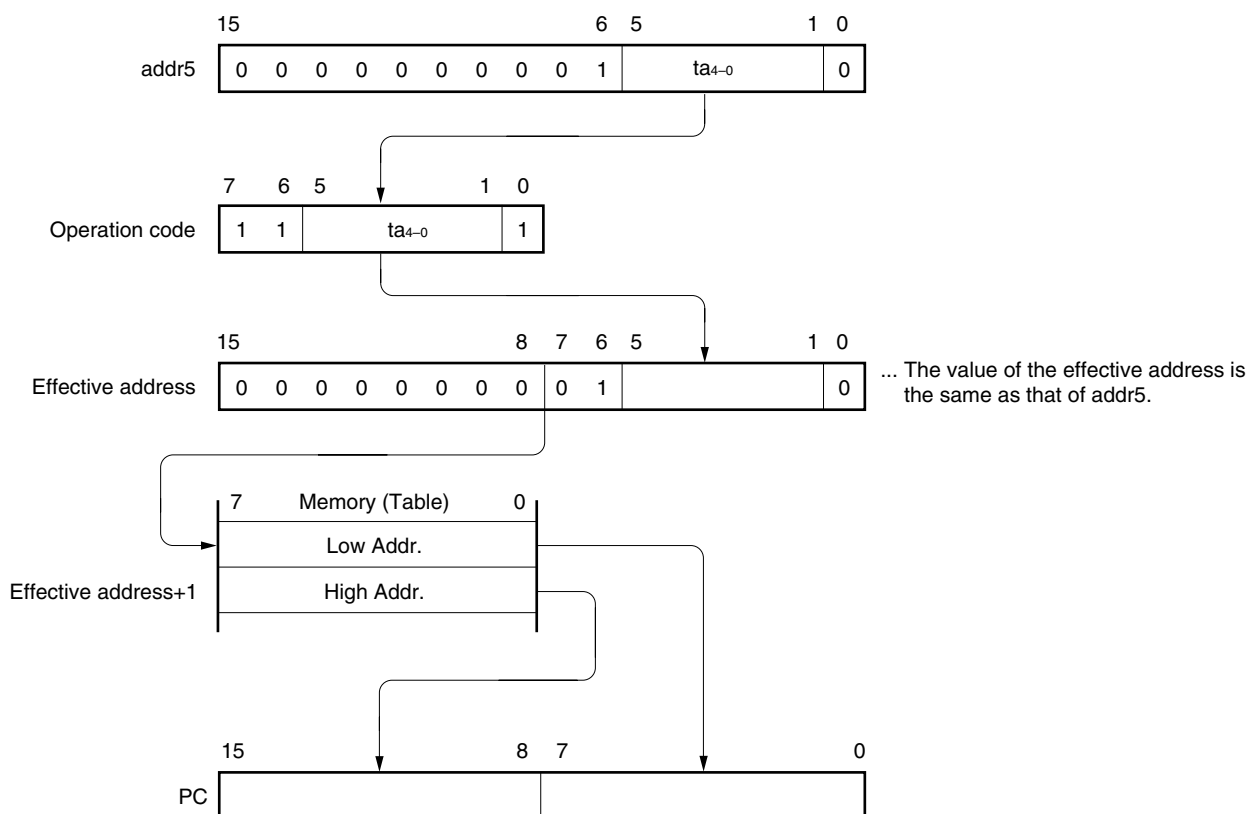
[功能]

用指令码第 1 位至第 5 位的立即数，在特定区域中被寻址的表内容（转移目的地址）赋给程序计数器(PC)，并转移到其相对应的地址。

当执行 CALLT [addr5]指令时，执行该功能。

该指令的基准地址指定为 addr5，并存放在 0040H 至 007FH 之间的存储表，并允许转移到整个存储器空间。

[图示]

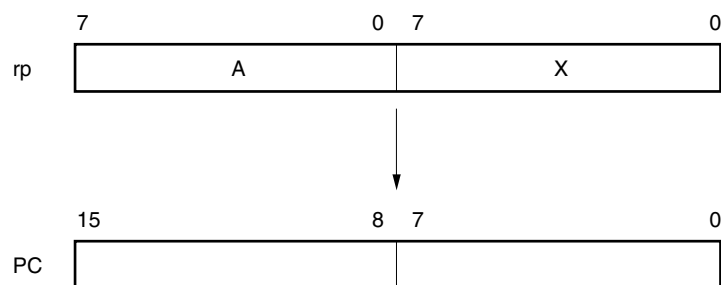


3.3.4 寄存器寻址

[功能]

将指令字指定的寄存器对(**AX**)的内容赋给程序计数器(**PC**), 并转移到其相对应的地址。
当执行 **BR AX** 指令时, 执行该功能。

[图示]



3.4 操作数地址寻址

执行指令期间, 寄存器和存储器的指定方法(寻址)如下。

3.4.1 隐含寻址

[功能]

该寄存器用作通用寄存器区域的累加器(**A** 和 **AX**), 进行自动(隐含)寻址。
在 μPD79F7023, 79F7024 微控制器的命令集, 使用的隐含寻址 指令如下所示。

指令	隐含寻址所指定的寄存器
MULU	A 寄存器存放被乘数, AX 寄存器存放积。
DIVUW	AX 寄存器存放被除数和商。
ADJBA/ADJBS	A 寄存器存放十进制调整后的数值
ROR4/ROL4	A 寄存器存放数字循环的数字数据

[操作数格式]

因为指令自动采用隐含寻址方式, 所以没有特定的操作数格式。

[说明示例]

MULU X 示例

一条 8 位× 8 位的乘法运算指令, **A** 寄存器和 **X** 寄存器之乘积存放在 **AX** 寄存器中。在此例中, 由隐含寻址指定了 **A** 寄存器和 **AX** 寄存器。

3.4.2 寄存器寻址

[功能]

通用寄存器用作操作数进行访问的寻址方式，由寄存器存储体选择标志(RBS0 至 RBS1)和指令码中的寄存器指定码，指定寻址的通用寄存器。

执行以下操作格式时，执行寄存器寻址方式。指定 8 位寄存器时，用指令码中的 3 位指定的 8 台寄存器中的 1 台。

[操作数格式]

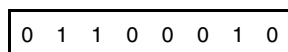
标识符	描述
r	X, A, C, B, E, D, L, H
rp	AX, BC, DE, HL

“r”和“rp”可用功能名称(X, A, C, B, E, D, L, H, AX, BC, DE 和 HL)，也可用绝对名称(R0 至 R7 和 RP0 至 RP3)进行描述。

[说明示例]

MOV A, C; 当选择 C 寄存器作为 r 时:

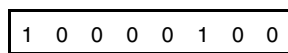
指令码



寄存器指定码

INCW DE; 当选择 DE 寄存器对作为 rp 时:

指令码



寄存器指定码

3.4.3 直接寻址

[功能]

对指令字中立即数给出的存储器进行直接寻址。
可在所有存储器空间进行寻址。

[操作数格式]

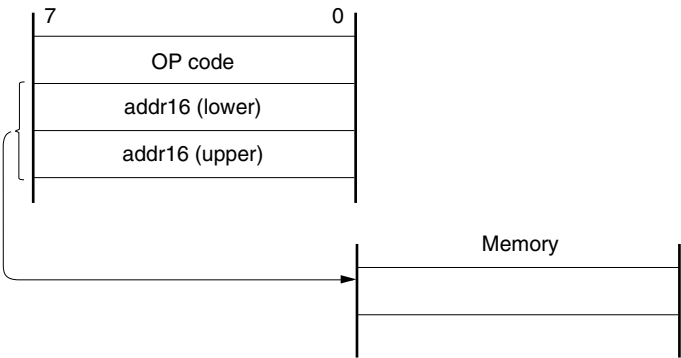
标识符	描述
addr16	标号或 16 位立即数

[说明示例]

MOV A, !0FE00H; 当设置!addr16 为 FE00H 时

指令码	1 0 0 0 1 1 1 0	OP 码
	0 0 0 0 0 0 0 0	00H
	1 1 1 1 1 1 1 0	FEH

[图示]



3.4.4 短直接寻址

[功能]

用指令字中的 8 位立即数对存储器的固定区域进行直接寻址。

寻址范围为 FE20H 至 FF1FH 的 256 字节区域。内部高速 RAM 和特殊功能寄存器(SFRs)分别映射在 FE20H 至 FEFFH 和 FF00H 至 FF1FH 的区域。

只有特殊功能寄存器(SFR)的 FF00H 至 FF1FH 区域可进行短直接寻址。程序中经常访问的端口、定时器/事件计数器的比较和捕捉寄存器都映射在该区域，可用很少的字节数和时钟数操作这些特殊功能寄存器。

当 8 位立即数在 20H 至 FFh 之间时，一条有效地址的第 8 位置“0”。在 00H 至 1FH 之间时，第 8 位置“1”。参阅以下[图示]。

[操作数格式]

标识符	描述
saddr	标号或 FE20H 至 FF1FH 之间的立即数
saddrp	标号或 FE20H 至 FF1FH 之间的立即数（仅限偶地址）

[说明示例]

LB1 EQU 0FE30H; 由 LB1 定义 FE30H。

:

MOV LB1, A; LB1 表示 saddr 区域的 FE30H，并且将 A 寄存器的值传送给该地址时

指令码

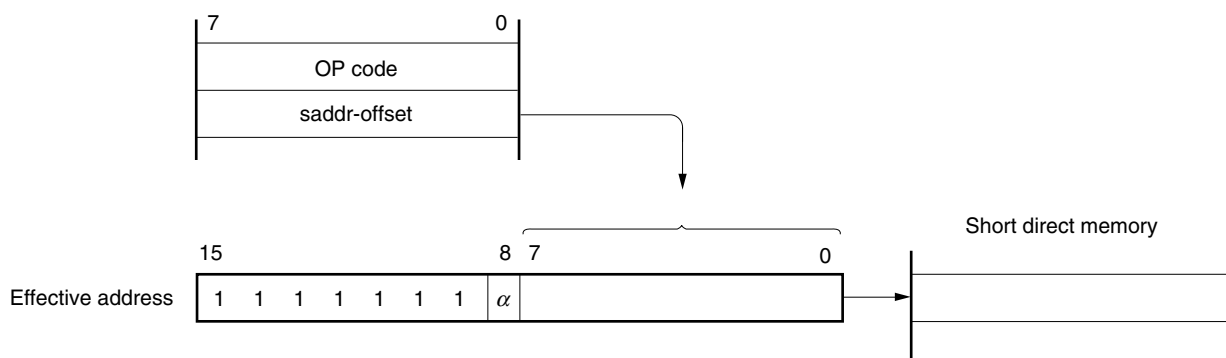
1 1 1 1 0 0 1 0

OP 码

0 0 1 1 0 0 0 0

30H (saddr-offset)

[图示]



当 8 位立即数为 20H 至 FFH 时， $\alpha = 0$

当 8 位立即数为 00H 至 1FH 时， $\alpha = 1$

3.4.5 特殊功能寄存器(SFR)寻址

[功能]

用指令字中的 8 位立即数对存储器映射的特殊功能寄存器(SFR)进行寻址。

寻址范围为 FF00H 至 FFCFH 和 FFE0H 至 FFFFH 的 240 字节区域。但是，映射在 FF00H 至 FF1FH 之间的特殊功能寄存器，则要用短直接寻址进行访问。

[操作数格式]

标识符	描述
sfr	特殊功能寄存器名称
sfrp	16 位可操作特殊功能寄存器名称（仅限偶地址）

[说明示例]

MOV PM0, A; 当选择 PM0 (FF20H)作为 sfr 时:

指令码

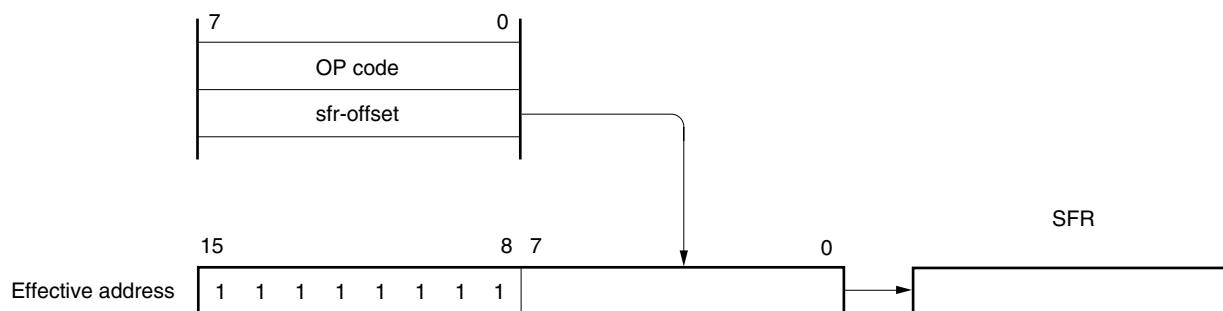
1 1 1 1 0 1 1 0

OP 码

0 0 1 0 0 0 0 0

20H (sfr-offset)

[图示]



3.4.6 寄存器间接寻址

[功能]

用寄存器对的内容作为操作数，对存储器进行寻址。由寄存器存储体选择标志 (RBS0 和 RBS1) 和指令字中的寄存器对指定码指定被访问的寄存器对。

可在所有存储器空间进行寻址。

[操作数格式]

标识符	描述
—	[DE], [HL]

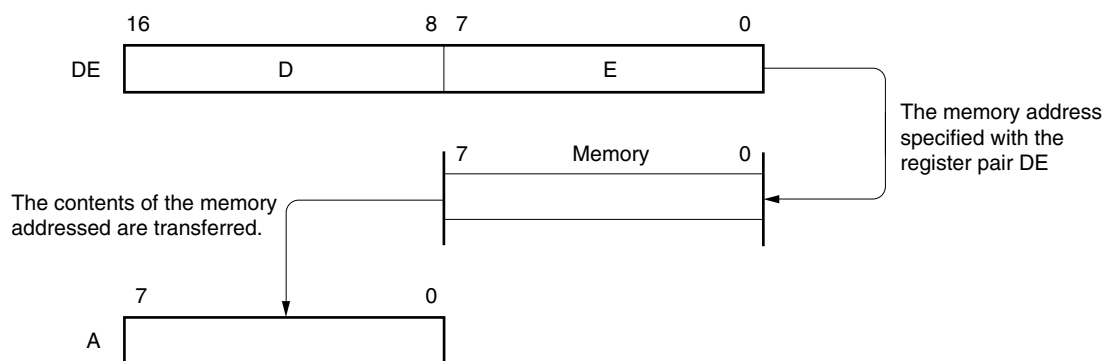
[说明示例]

MOV A, [DE]; 当选择[DE]作为寄存器对时:

指令码

1	0	0	0	0	1	0	1
---	---	---	---	---	---	---	---

[图示]



3.4.7 基址寻址

[功能]

将 8 位立即数加到作为基址寄存器的 HL 寄存器对的内容，根据其相加结果对存储器进行寻址。由寄存器存储体选择标志(RBS0 和 RBS1)指定访问的 HL 寄存器对。通过将偏移量扩展到 16 位正数，执行加法运算。第 16 位进位可忽略不计。

可在所有存储器空间进行寻址。

[操作数格式]

标识符	描述
—	[HL + byte]

[说明示例]

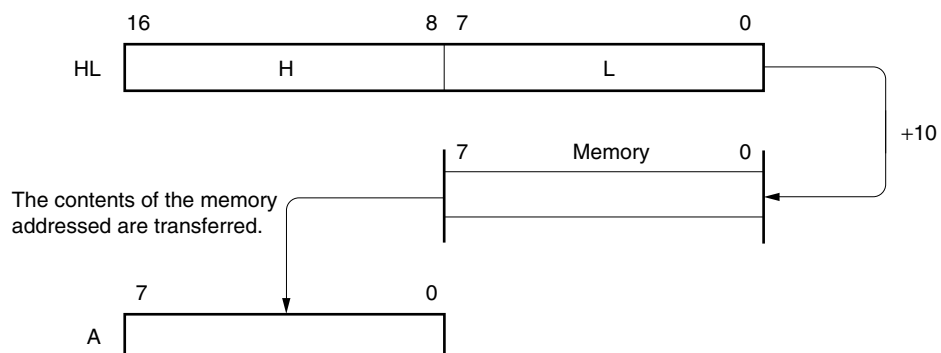
MOV A, [HL + 10H]; 当字节置为 10H 时:

指令码

1 0 1 0 1 1 1 0

0 0 0 1 0 0 0 0

[图示]



3.4.8 寻址变址寻址

[功能]

将 B 或 C 寄存器的内容相加到用作基址寄存器的 HL 寄存器对的内容，根据其相加结果对存储器进行寻址。由寄存器存储体选择标志(RBS0 至 RBS1)指定被访问的 HL 寄存器对。通过将 B 或 C 寄存器扩展到 16 位正数，执行加法运算。第 16 位进位可忽略不计。

可在所有存储器空间进行寻址。

[操作数格式]

标识符	描述
–	[HL + B], [HL + C]

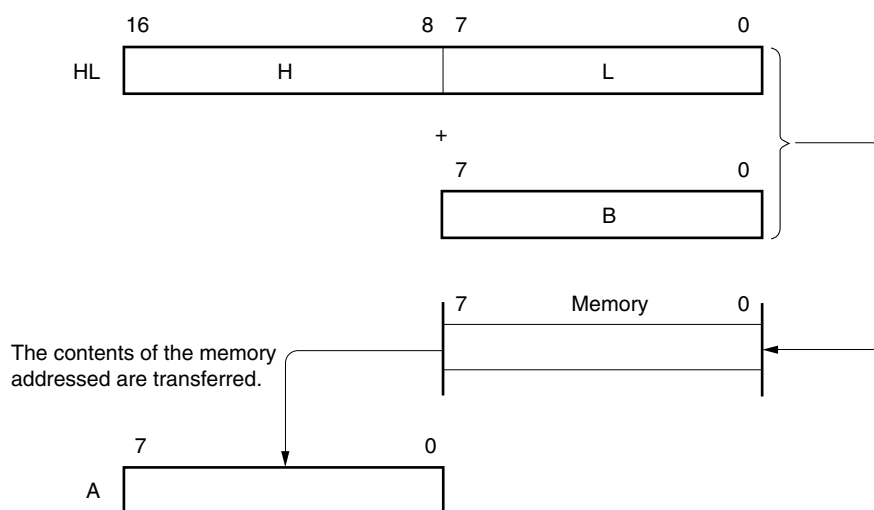
[说明示例]

MOV A, [HL + B]; 当选择 B 寄存器时:

指令码

1	0	1	0	1	0	1	1
---	---	---	---	---	---	---	---

[图示]



3.4.9 堆栈寻址

[功能]

用堆栈(SP)内容对堆栈区域进行间接寻址。

当执行 **PUSH**、**POP**、子程序调用和返回指令，或者因中断请求产生保存或恢复寄存器操作时,该寻址方法将被自动执行。

堆栈寻址仅限于内部高速 **RAM** 区域。

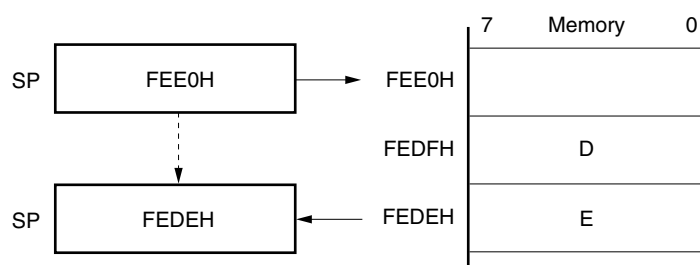
[说明示例]

PUSH DE，当存入 **DE** 寄存器时：

指令码

1	0	1	1	0	1	0	1
---	---	---	---	---	---	---	---

[图示]



第四章 端口功能

4.1 端口功能

端口引脚 I/O 缓冲电源分为两种类型：AV_{REF} 和 V_{DD}。电源和引脚关系分别如下图所示：

表 4-1. 各引脚 I/O 缓冲电源

电源	对应引脚
AV _{REF}	P20 至 P27
V _{DD}	P20 至 P27 除外的引脚

提供的数字 I/O 端口,支持多种控制操作。各端口功能如表 4-2 所示。

除数字 I/O 端口功能之外, 还有各种复用功能。有关详细的复用功能, 参阅第二章 引脚功能。

表 4-2. 端口功能

功能名称	I/O	功能	复位时	复用功能
P20	I/O	端口 2 8 位 I/O 端口 可按 1 位单元指定输入/输出。	模拟输入	ANI0/AMP0-
P21				ANI1/AMP0OUT
P22				ANI2/AMP0+
P23				ANI3
P24				ANI4
P25				AMP1-
P26				AMP1OUT
P27				AMP1+
P30	I/O	端口 3 5 位 I/O 端口 可按 1 位单元指定输入/输出。 通过软件设置，可使用内置上拉电阻。	输出端口	TOH1/TI51/INTP0
P31			模拟输入	TxD0/CMPCOM
P32				RxD0/CMPIN
P33			输入端口	TI000/INTP1
P34				TO00/TI010/CMPOUT
P121	I/O	端口 12 P121,P122 为 2 位 I/O 端口 P125 为 1 位输入专用端口	输出端口	X1/TOOLC0
P122				X2/EXCLK/TOOLD0
P125	输入	仅限 P125 可通过软件设置，使用内置上拉电阻。	复位输入	RESET

4.2 端口配置

端口包含如下硬件。

表 4-3. 端口配置

项目	配置
控制寄存器	端口模式寄存器 (PMxx): PM2, PM3, PM12 端口寄存器 (Pxx): P2, P3, P12 上拉电阻选择寄存器(PUxx): PU3, PU12 复位引脚模式寄存器(RSTMASK) A/D 端口配置寄存器(ADPC)
端口	共计: 16 个 (CMOS I/O:15 个, CMOS 输入:1 个)
上拉电阻	共计: 6 个

4.2.1 端口 2

端口 2 是带有输出锁存功能的 I/O 端口。通过端口模式寄存器 2(PM2),可按 1 位单元指定输入模式或输出模式。该端口也可兼用作 A/A 转换模拟输入和运算放大器 I/O。

使用 P20/AMP0-/ANI0 至 P24/ANI4 时,可以根据使用的引脚功设置寄存器。(参阅表 4-4 至 4-8)。

P20/AMP0-/ANI0 至 P24/ANI4 用作数字输入或数字输出时,建议从离 AV_{REF} (P24/ANI4 引脚)最远的引脚开始。
P20/AMP0-/ANI0 至 P24/ANI4 用作数字输入或数字输出时,建议从离 V_{SS} (P24/ANI4 引脚)最近的引脚开始。

复位信号产生时,端口 2 设置为模拟输入。

端口 2 系统图如图 4-1 至 4-7 所示。

注意事项 端口 2 用作数字端口时,请将 AV_{REF} 引脚设置与 V_{DD} 引脚相同的电位。

表 4-4. P20/ANI0/AMP0-和 P22/ANI2/AMP0+引脚的设置功能

ADPC寄存器	PM2寄存器	OPAMP0E位	ADS 寄存器 (n = 0, 2)	P20/ANI0/AMP0-和 P22/ANI2/AMP0+引脚
数字I/O选择	输入模式	—	选择ANIn	禁止设置
			禁止选择ANIn	数字输入
	输出模式	—	选择ANIn	禁止设置
			禁止选择ANIn	数字输出
模拟输入选择	输入模式	0	选择ANIn	模拟输入 (转换为数字模式)
			禁止选择ANIn	模拟输入 (禁止转换为数字模式)
		1	选择ANIn	模拟输入 (转换为数字模式), 运算放大器
			禁止选择ANIn	运算放大器0输入
	输出模式	—	—	禁止设置

备注

ADPC: A/D 端口配置寄存器

PM2: 端口模式寄存器 2

OPAMP0E: 运算放大器控制寄存器(AMPM)的第 7 位

ADS: 模拟输入通道指定寄存器

表 4-5. P21/ANI1/AMP0OUT 引脚功能的设置

ADPC寄存器	PM2寄存器	OPAMP0E位	ADS寄存器	P21/ANI1/AMP0OUT引脚
数字I/O选择	输入模式	0	选择ANI1	禁止设置
			禁止选择ANI1	数字输入
		1	—	禁止设置
	输出模式	0	选择ANI1	禁止设置
			禁止选择ANI1	数字输出
		1	—	禁止设置
模拟I/O选择	输入模式	0	选择ANI1	模拟输入（转换为数字模式）
			禁止选择ANI1	模拟输入（禁止转换为数字模式）
		1	选择ANI1	运算放大器（转换为数字模式）
			禁止选择ANI1	运算放大器 0输出（转换为数字模式）
	输出模式	—	—	禁止设置

表 4-6. P23/ANI3, P24/ANI4 引脚的设置功能

ADPC寄存器	PM2寄存器	ADS寄存器(n = 3, 4)	P23/ANI3和P24/ANI4引脚
数字I/O选择	输入模式	选择ANIn	禁止设置
		禁止选择ANIn	数字输入
	输出模式	选择ANIn	禁止设置
		禁止选择ANIn	数字输出
模拟输入选择	输入模式	选择ANIn	模拟输入（转换为数字模式）
		禁止选择ANIn	模拟输入（禁止转换为数字模式）
	输出模式	—	禁止设置

备注 ADPC: A/D 端口配置寄存器
 PM2: 端口模式寄存器 2
 OPAMP0E: 运算放大器控制寄存器(AMPM)的第 7 位
 ADS: 模拟输入通道指定寄存器

表 4-7. P25/AMP1-和 P27/AMP1+引脚的设置功能

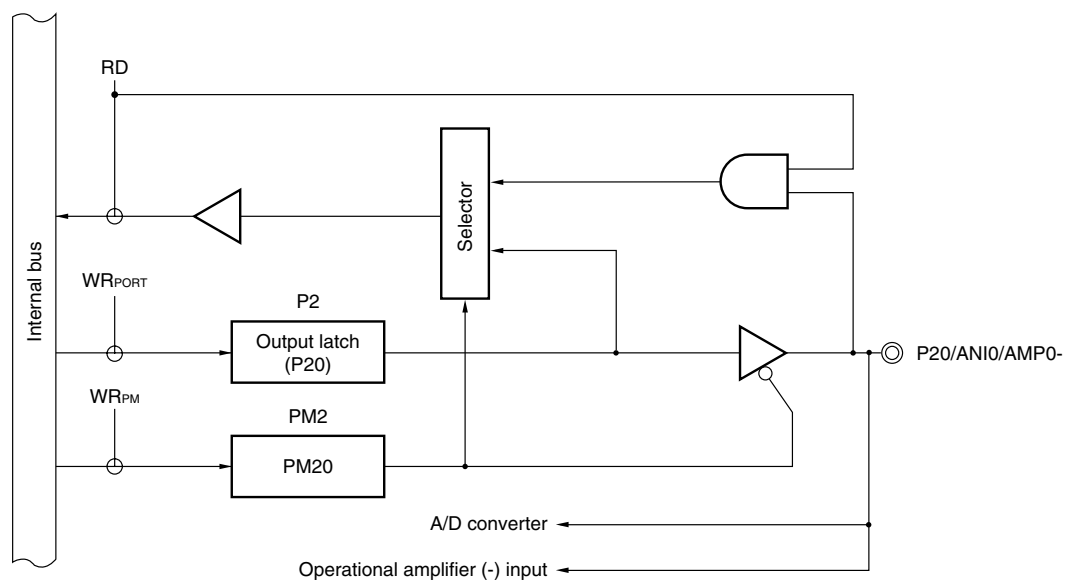
ADPC寄存器	PM2寄存器	OPAMP1E位	P25/AMP1-和P27/AMP1+引脚
数字I/O选择	输入模式	—	数字输入
	输出模式	—	数字输出
模拟输入选择	输入模式	0	模拟输入（转换为数字模式）
		1	运算放大器1输入
	输出模式	—	禁止设置

表 4-8. P26/AMP1OUT 引脚的设置功能

ADPC寄存器	PM2寄存器	OPAMP1E位	P23/ANI3和P24/ANI4引脚
数字I/O选择	输入模式	—	数字输入
	输出模式	—	数字输出
模拟输入选择	输入模式	0	模拟输入（转换为数字模式）
		1	运算放大器 1输出
	输出模式	—	禁止设置

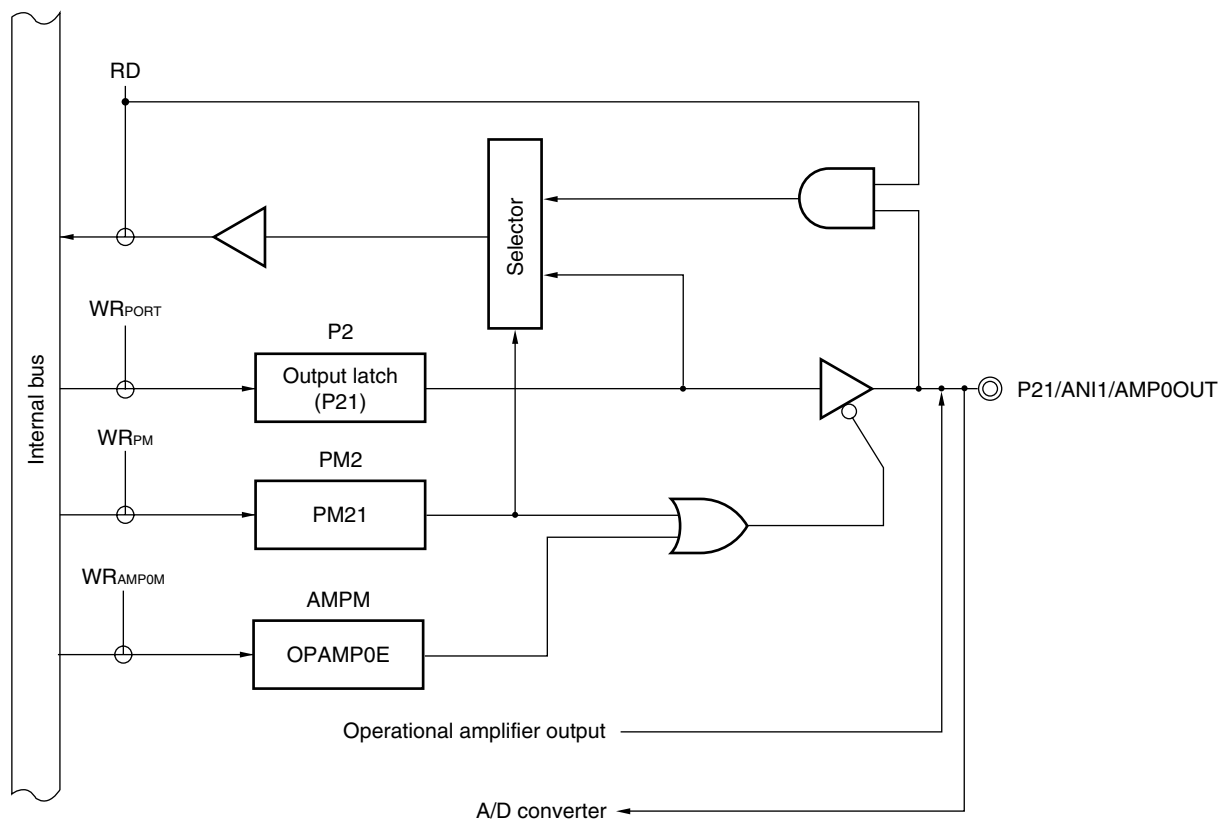
备注 ADPC: A/D 端口配置寄存器
 PM2: 端口模式寄存器 2
 OPAMP0E: 运算放大器控制寄存器(AMPM)的第 7 位

图 4-1. P20 系统图



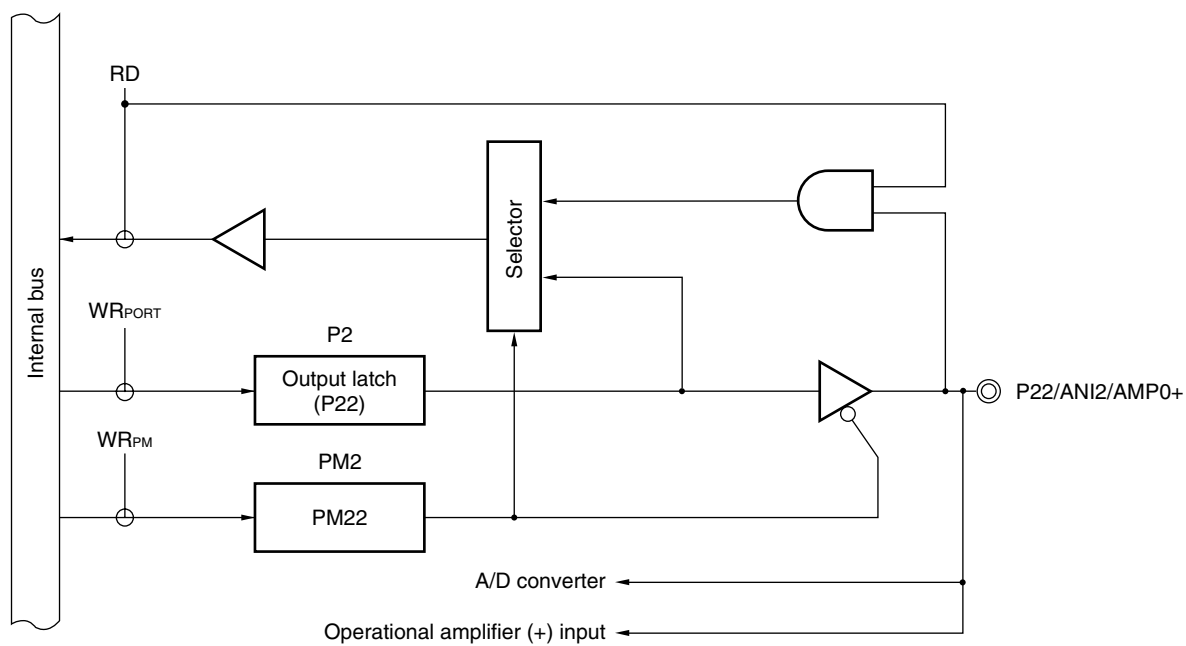
P2: 端口寄存器 2
 PM2: 端口模式寄存器 2
 RD: 读取信号
 WR_{xx}: 写入信号

图 4-2. P21 系统图



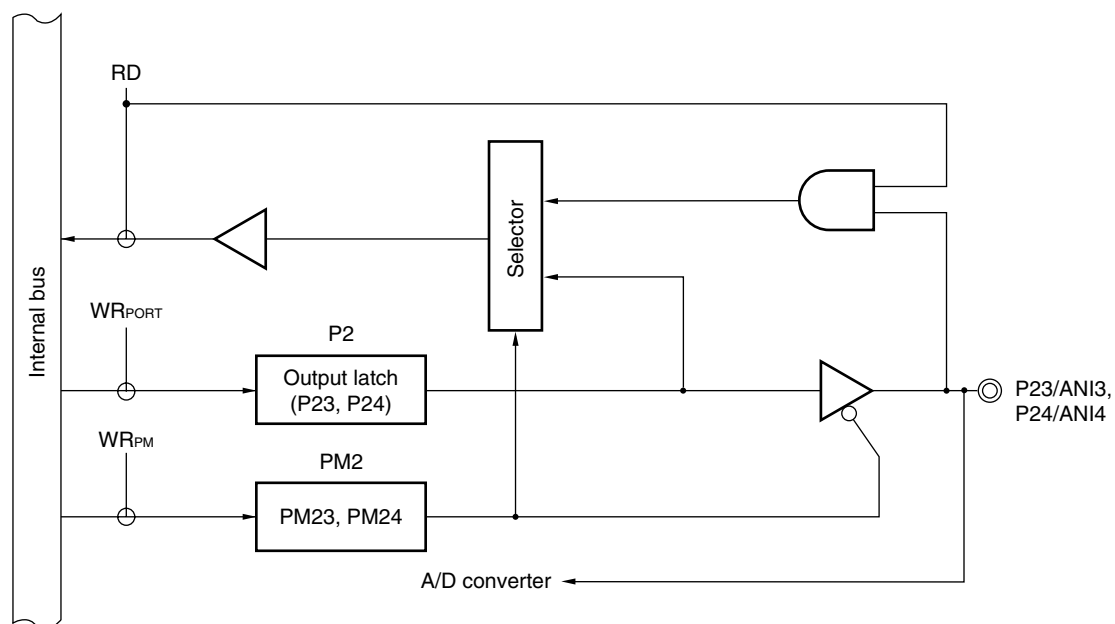
P2: 端口寄存器 2
 PM2: 端口模式寄存器 2
 RD: 读取信号
 WR_{xx}: 写入信号

图 4-3. P22 系统图



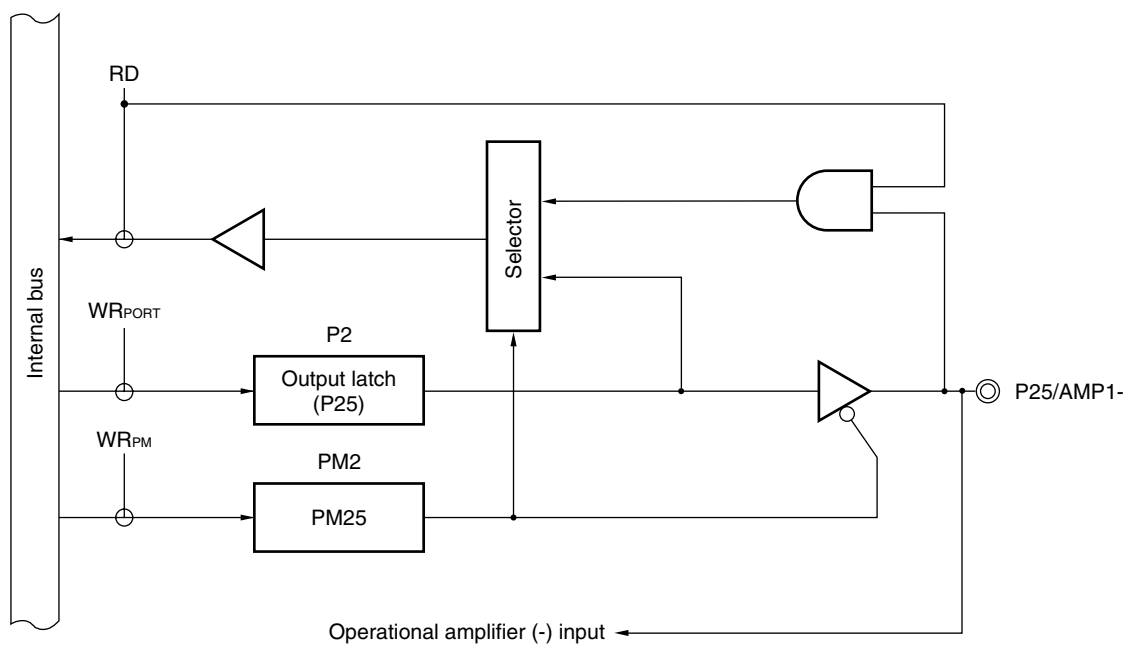
P2: 端口寄存器 2
 PM2: 端口模式寄存器 2
 RD: 读取信号
 WR_{xx}: 写入信号

图 4-4. P23 和 P24 系统图



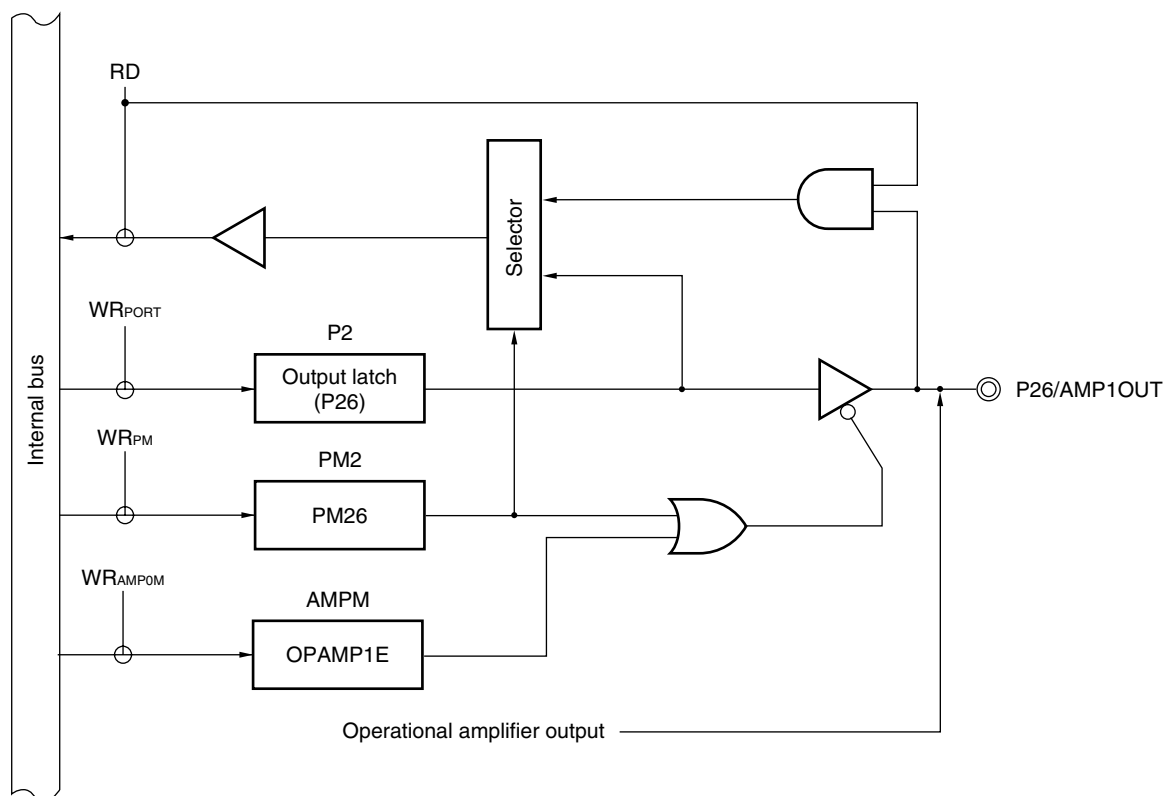
P2:	端口寄存器 2
PM2:	端口模式寄存器 2
RD:	读取信号
WR _{xx} :	写入信号

图 4-5. P25 系统图



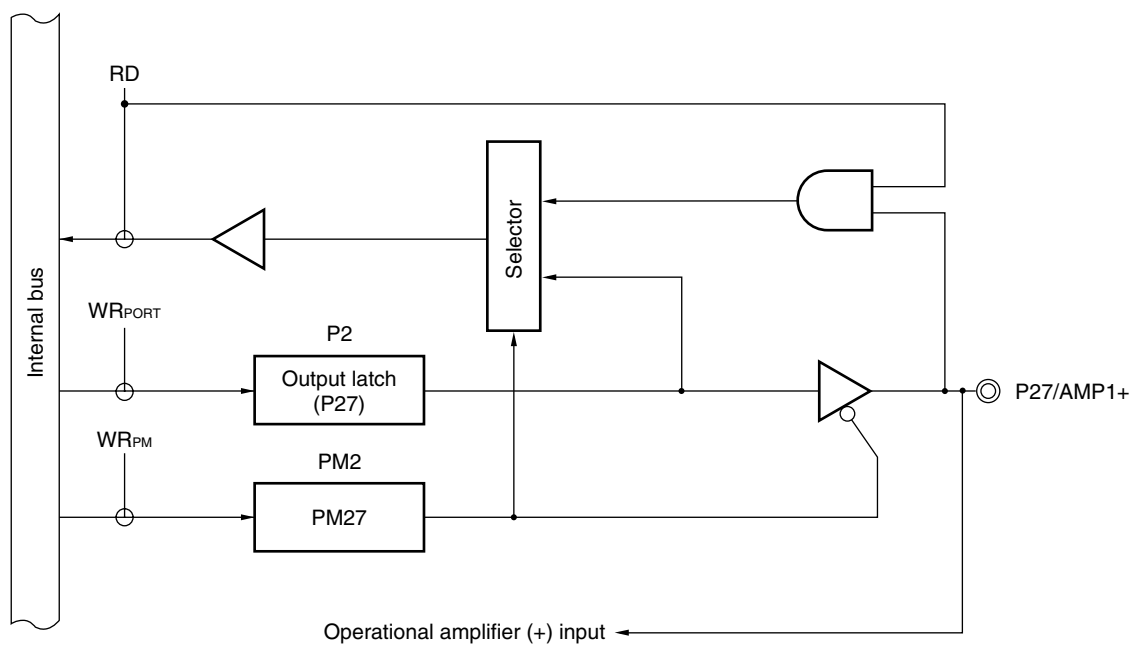
P2: 端口寄存器 2
 PM2: 端口模式寄存器 2
 RD: 读取信号
 WR_{xx}: 写入信号

图 4-6. P26 系统图



P2: 端口寄存器 2
 PM2: 端口模式寄存器 2
 RD: 读取信号
 WR_{xx}: 写入信号

图 4-7. P27 系统图



P2: 端口寄存器 2
 PM2: 端口模式寄存器 2
 RD: 读取信号
 WR_{xx}: 写入信号

4.2.2 端口 3

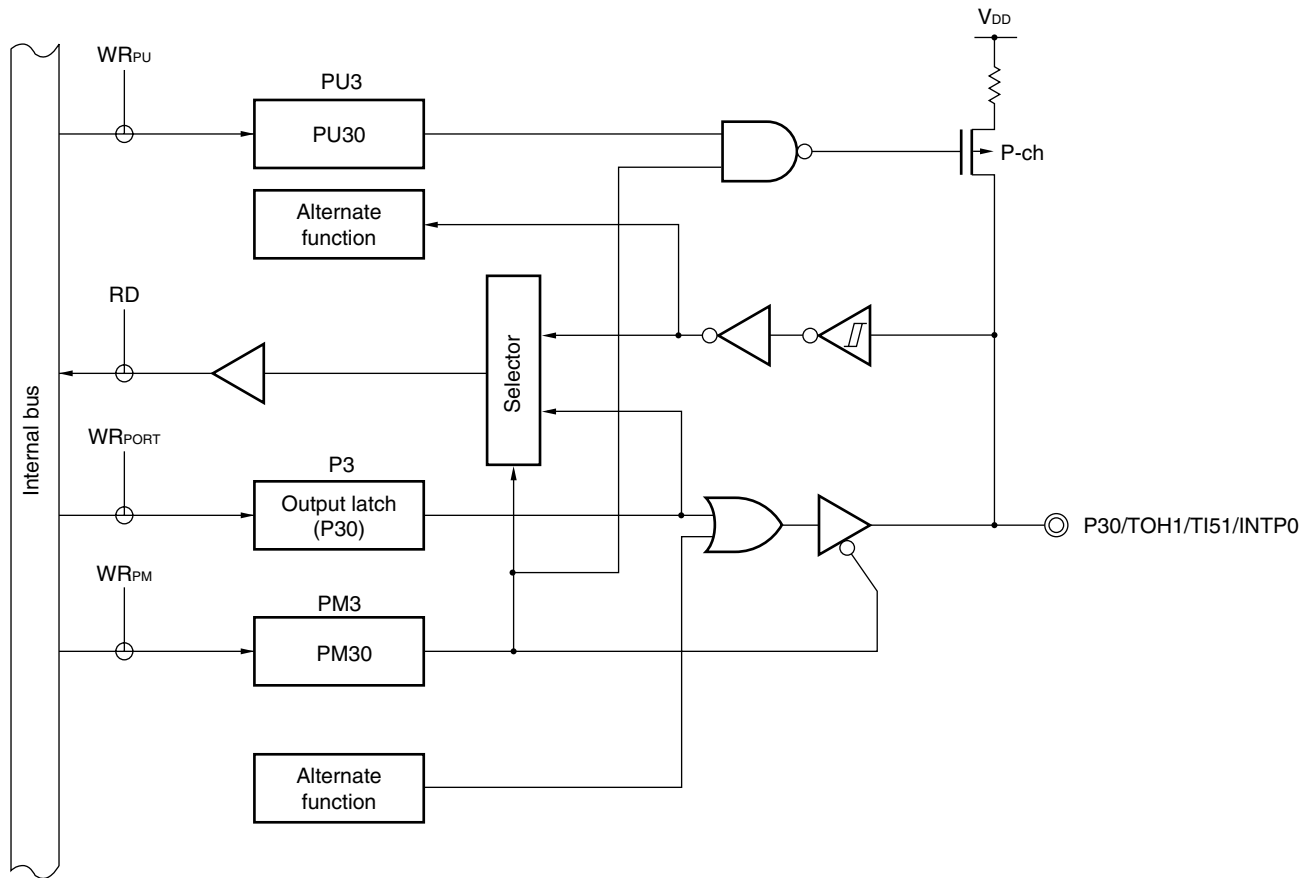
端口 3 是具有输出锁存功能的 I/O 端口。通过端口模式寄存器 3(PM3)，可按 1 位单元指定输入模式或输出模式。P30-P34 引脚用作输入端口时，可通过上拉电阻选择寄存器 3(PU3)按 1 位单元指定内置上拉电阻。

也可兼用作外部中断请求输入、定时器的输出、闪存编程器/片上调试器的时钟输出和数据输出、串行接口的时钟输出和数据输出。

复位信号产生时，端口 3 设置为输入模式。

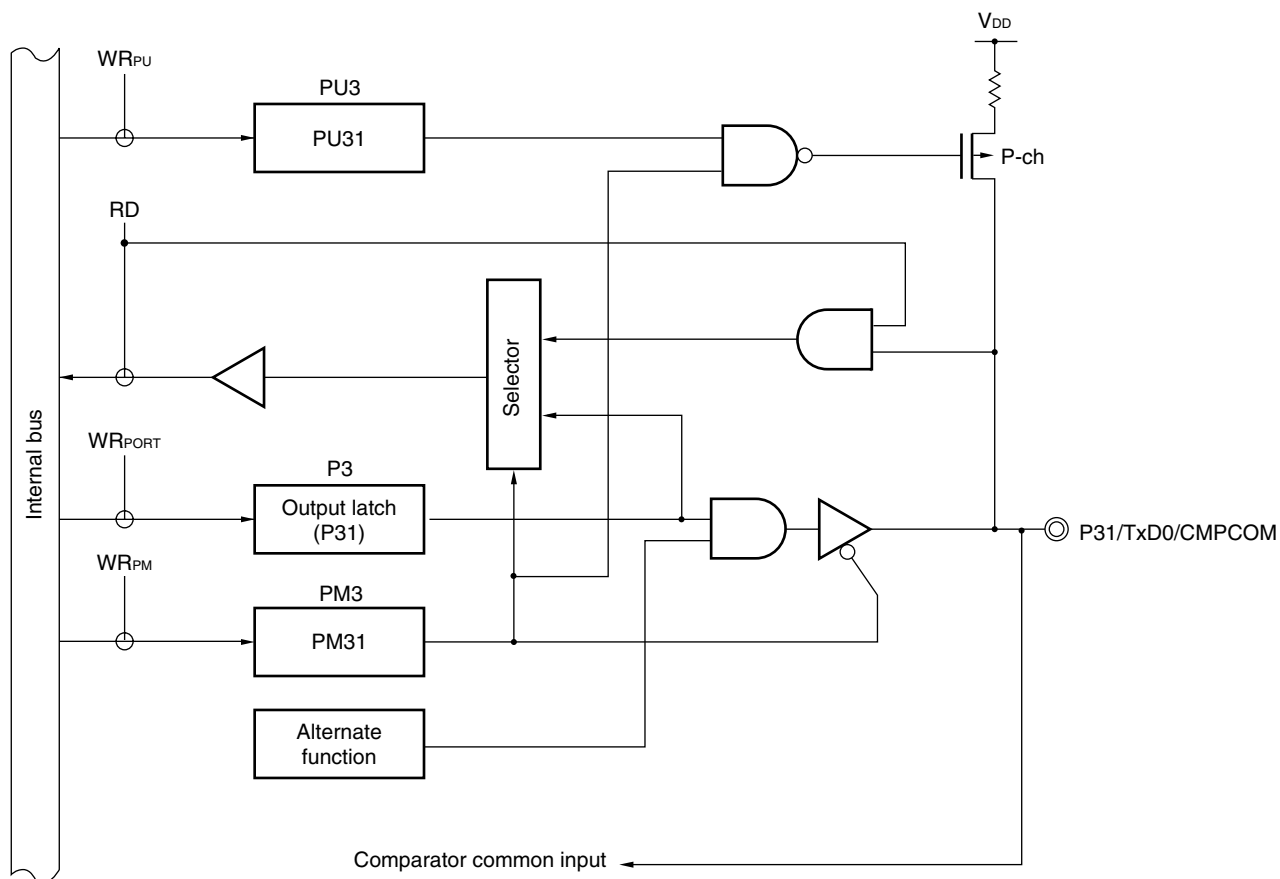
端口 3 系统图如图 4-8 至 4-12 所示。

图 4-8. P30 系统图



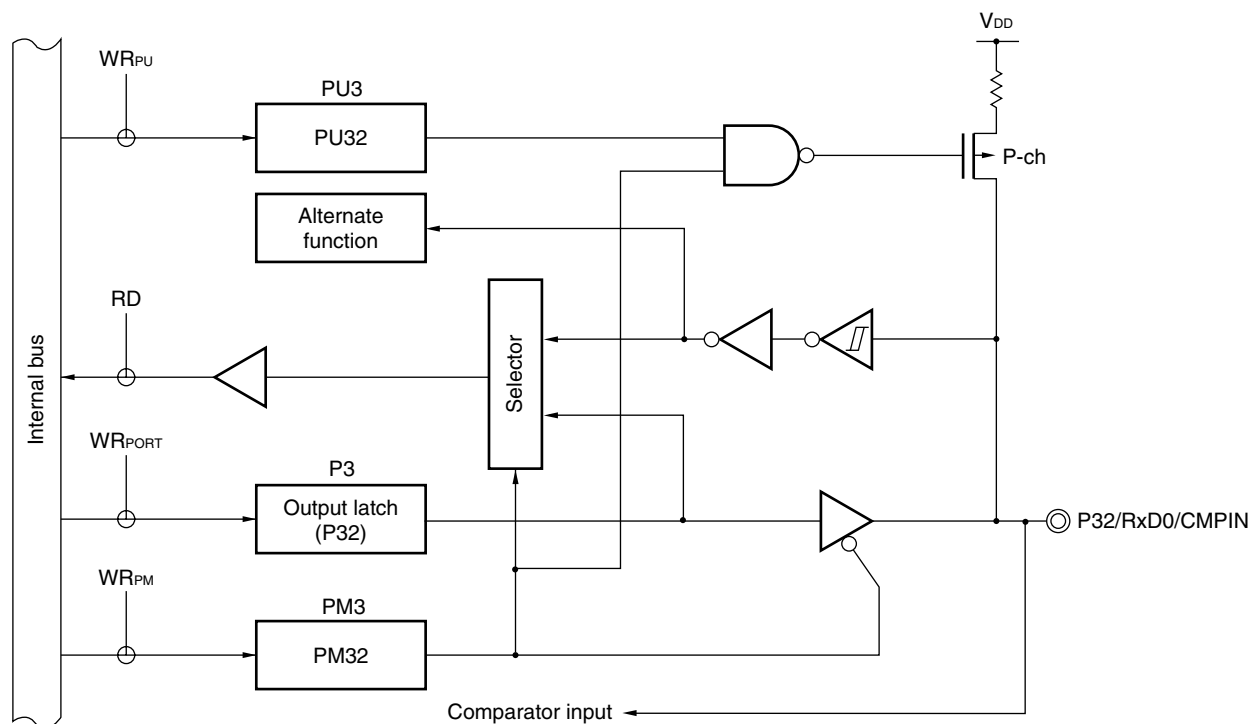
- P3:** 端口寄存器 3
PU3: 上拉电阻选择寄存器 3
PM3: 端口模式寄存器 3
RD: 读取信号
WR_{xx}: 写入信号

图 4-9. P31 系统图



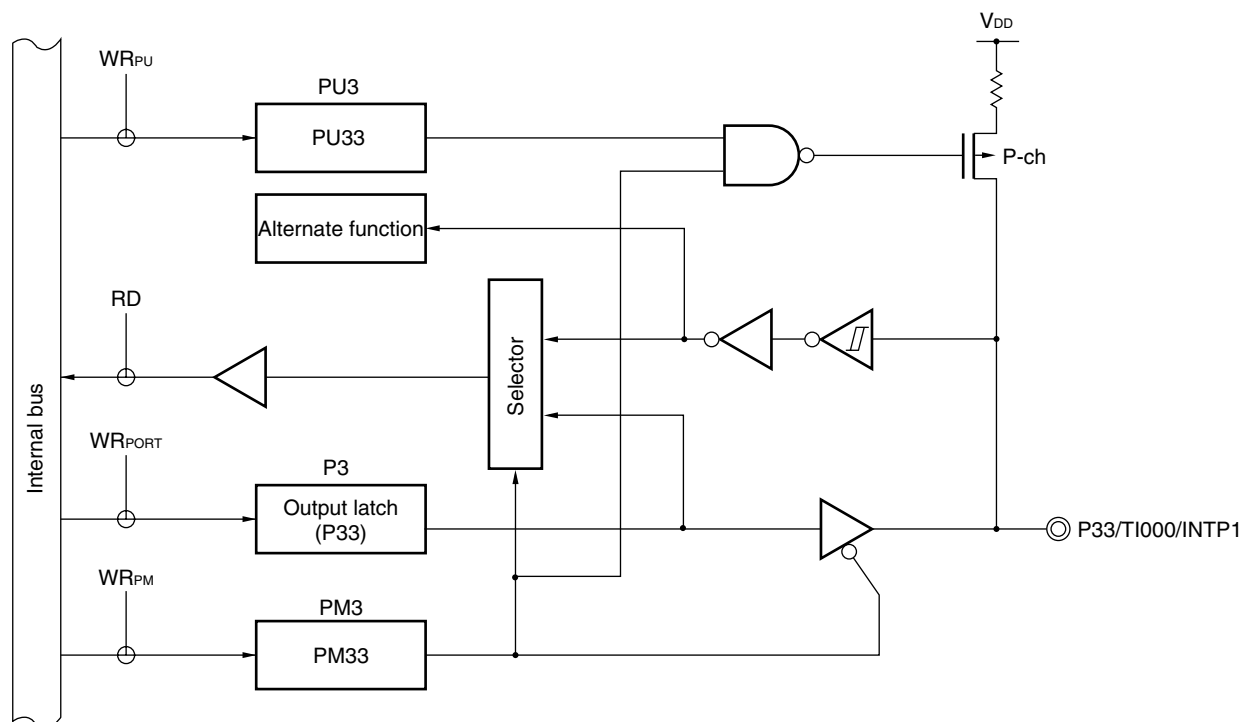
- P3:** 端口寄存器 3
PU3: 上拉电阻选择寄存器 3
PM3: 端口模式寄存器 3
RD: 读取信号
WR_{xx}: 写入信号

图 4-10. P32 系统图



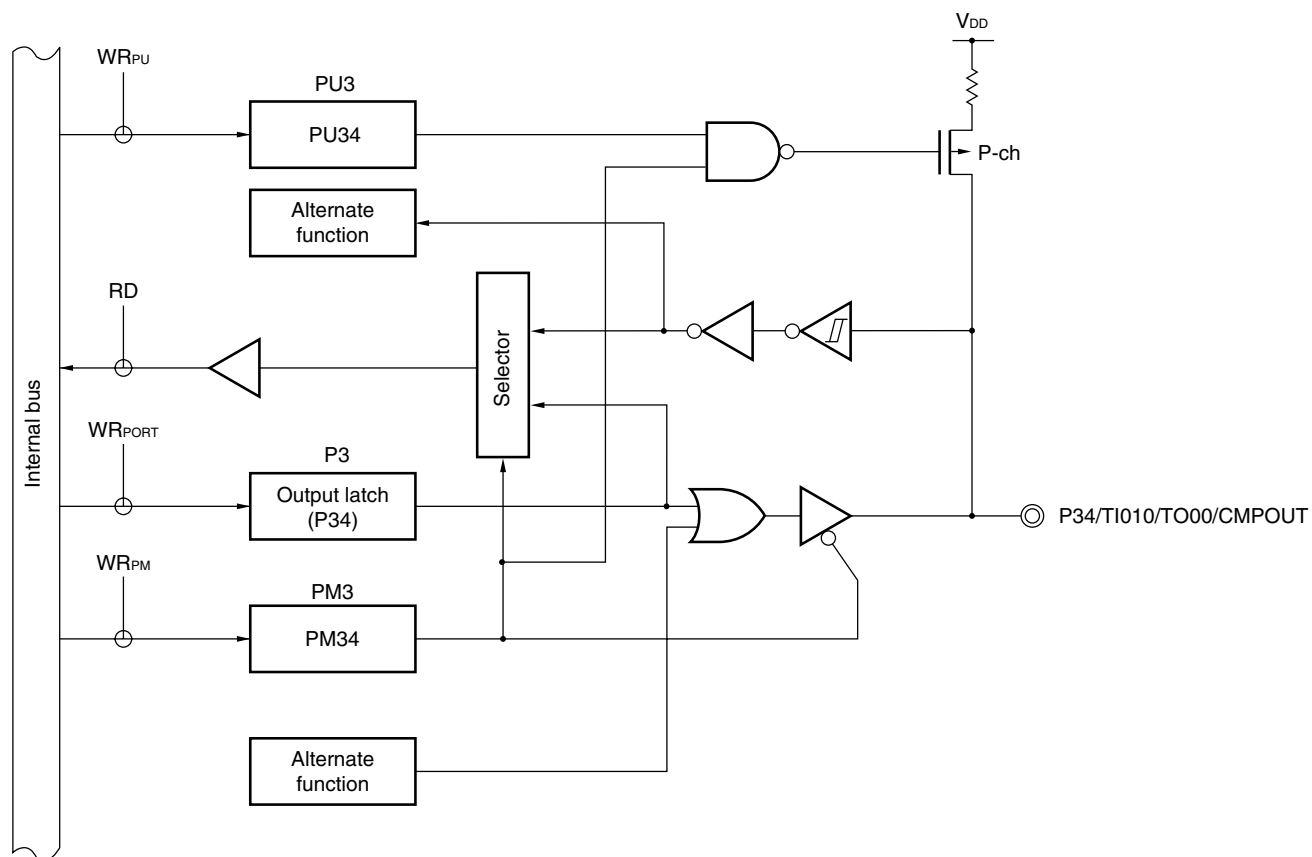
- P3: 端口寄存器 3
- PU3: 上拉电阻选择寄存器 3
- PM3: 端口模式寄存器 3
- RD: 读取信号
- WR_{xx} : 写入信号

图 4-11. P33 系统图



- P3: 端口寄存器 3
 PU3: 上拉电阻选择寄存器 3
 PM3: 端口模式寄存器 3
 RD: 读取信号
 WR_{xx} : 写入信号

图 4-12. P34 系统图



- P3:** 端口寄存器 3
PU3: 上拉电阻选择寄存器 3
PM3: 端口模式寄存器 3
RD: 读取信号
WR_{xx}: 写入信号

4.2.3 端口 12

P121 和 P122 是具有输出锁存功能的 I/O 端口。通过端口模式寄存器 12(PM12)，端口 125 可按 1 位单元设置输入或输出模式。

P125 为输出端口。

P121-P122 引脚用作输入端口时，可通过上拉电阻选择寄存器 12(PU12)指定内置上拉电阻。

还可用作主系统时钟的谐振器连接、主系统时钟的外部时钟输入、外部复位输入功能和闪存编程器/内置调试程序的时钟输入和数据 I/O 的引脚。

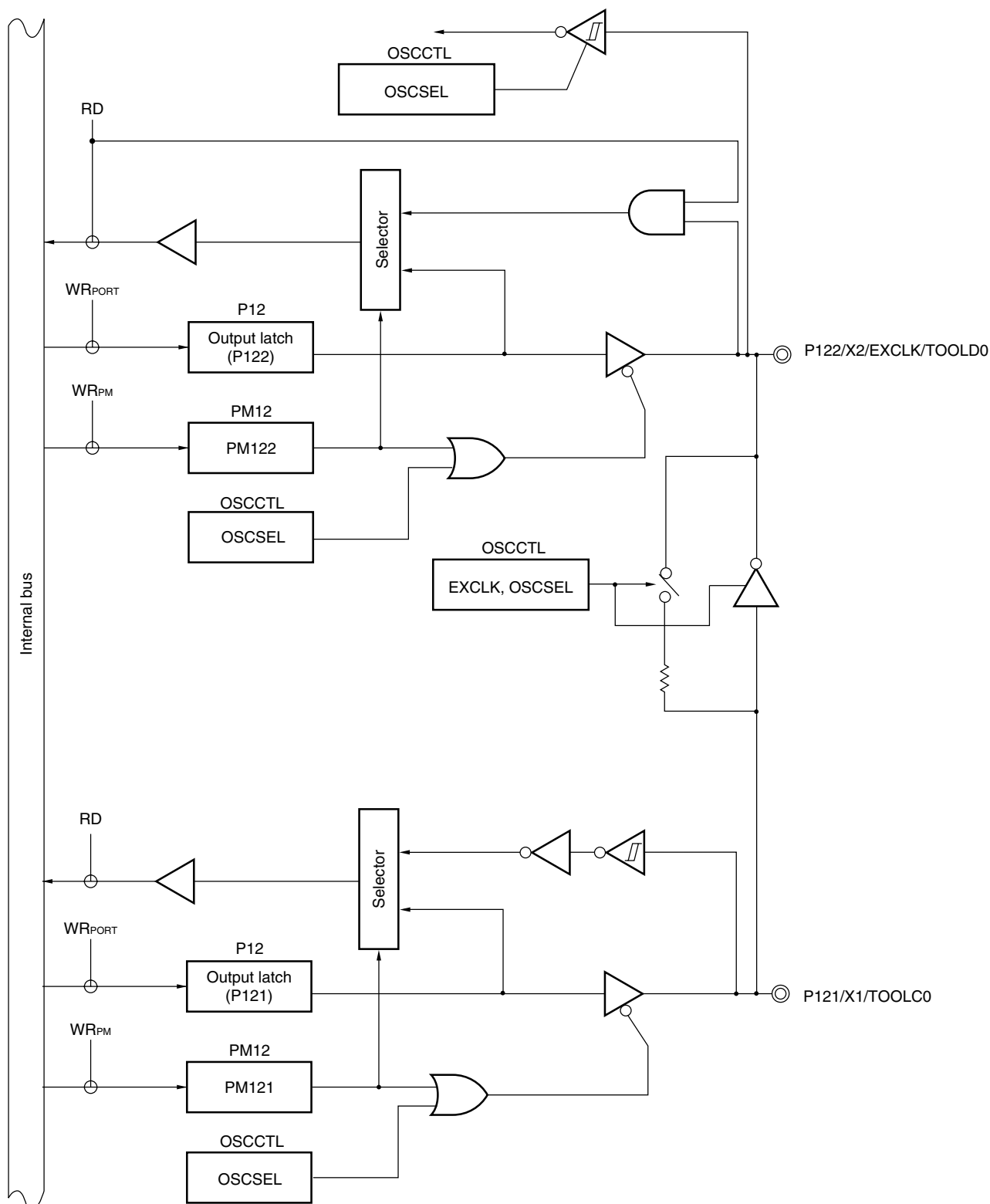
P125/RESET 作为输入端口使用时，复位端口模式寄存器(RSRMASK)的第 5 位(RSTM)置“1”；作为外部复位输入使用时，则把 RSTM 置“0”。

复位信号产生时，端口 12 设置为输入模式。

端口 12 系统图如图 4-13 和 4-14 所示。

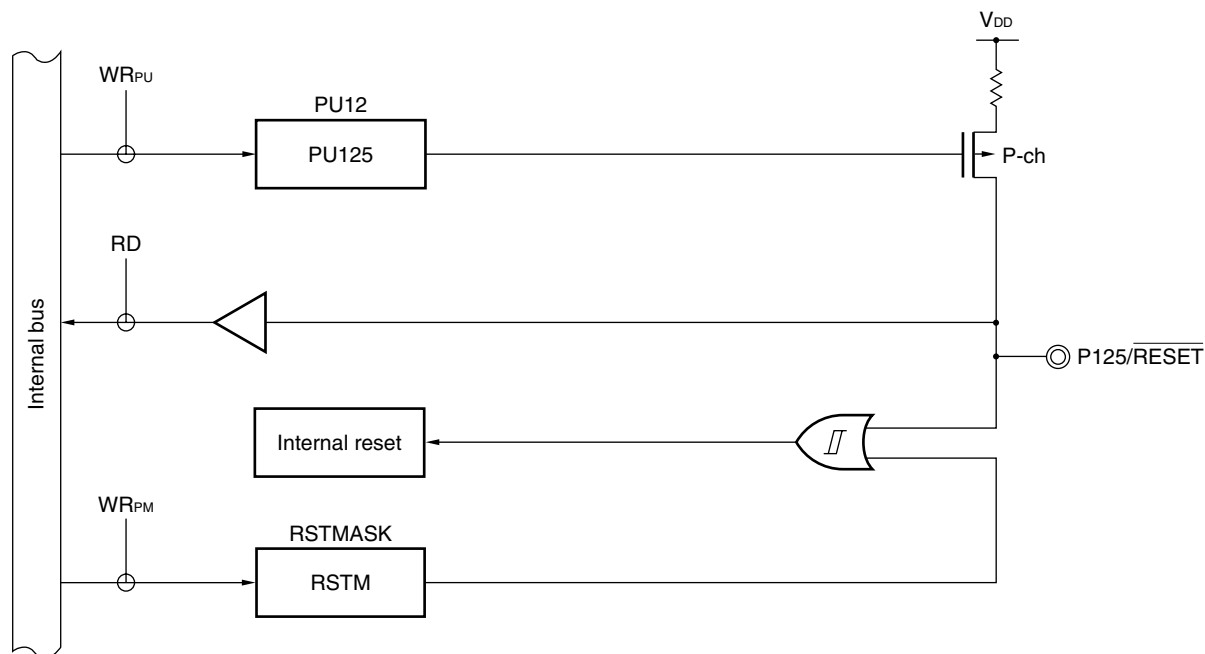
- 注意事项**
1. 使用 P121 和 P122 引脚连接谐振器到主系统时钟(X1,X2)器或输入外部时钟到主系统时钟(EXCLK)时，必须使用时钟操作模式寄存器(OSCCTL)设置 X1 振荡模式或外部时钟输入模式(详阅 5.3(1)时钟操作模式选择寄存器(OSCCTL))。OSCCTL 复位值为 00H (所有 P121 和 P122 引脚为输入端口引脚)。
 2. 复位解除后，RESET/P125 设置为外部复位输入。
 3. 复位释放后的 RESET/P125 设置为外部复位输入，因此，低电平输入期间复位时，复位状态将会持续到高电平输入恢复为止。

图 4-13. P121 和 P122 系统图



- P12: 端口寄存器 12
 PU12: 上拉电阻选择寄存器 12
 PM12: 端口模式寄存器 12
 OSCCTL: 时钟操作模式选择寄存器
 RD: 读取信号
 WR_{xx}: 写入信号

图 4-14. P125 系统图



PU12: 上拉电阻选择寄存器 12

RD: 读取信号

WR_{xx}: 写入信号

RSTMASK: 复位引脚模式寄存器

注意事项 复位释放后的 $\overline{\text{RESET}}/\text{P125}$ 设置为外部复位输入，因此，低电平输入期间复位时，复位状态将会持续到高电平输入恢复为止。

备注 复位后，可使用外部复位功能和上拉电阻($\text{RSTM} = 0, \text{PU125} = 1$)。用作端口功能时，将 RSTM 置“1”。

4.3 寄存器控制端口功能

通过以下五种类型寄存器控制端口功能。

- 端口模式寄存器(PMxx)
- 端口寄存器(Pxx)
- 上拉电阻选择寄存器(PUxx)
- 复位引脚模式寄存器(RSTMASK)
- A/D 端口配置寄存器(ADPC)

(1) 端口模式寄存器(PMxx)

这些寄存器可按 1 位单元指定端口的输入或输出模式。

可以由 1 位或 8 位存储器操作指令设置这些寄存器。

复位信号产生时，这些寄存器设置为 FFH。

端口引脚使用复用功能时，参照 4.5 使用复用功能时，端口模式寄存器和输出锁存器的设置，设置端口模式寄存器。

图 4-15. 端口模式寄存器格式

符号	7	6	5	4	3	2	1	0	地址	复位后	R/W
PM2	PM27 ^{注1}	PM26 ^{注1}	PM25 ^{注1}	PM24 ^{注1}	PM23 ^{注1}	PM22 ^{注1}	PM21 ^{注1}	PM20 ^{注1}	FF22H	FFH	R/W
PM3	1	1	1	PM34	PM33 ^{注2}	PM32 ^{注2}	PM31	PM30	FF23H	FFH	R/W
PM12	1	1	1	1	1	PM122	PM121	1	FF2CH	FFH	R/W

PMmn	Pmn 引脚 I/O 模式选择 (m = 2, 3, 12; n = 0 至 7)
0	输出模式 (输出缓冲打开)
1	输入模式 (输出缓冲关闭)

注 1. 该引脚使用 ADPC0 寄存器设置为模拟输入时，确保设置为输入模式。

2. 该引脚使用 CMPPC 寄存器设置为模拟输入时，确保设置为输入模式。

注意事项 确保 PM3 的第 5 位至第 7 位、PM12 的第 0 位和第 3 位至第 7 位置“1”。

(2) 端口寄存器(Pxx)

端口输出时，用于写入片上输出数据的寄存器。

若输入模式下读取数据时，则读取的是引脚电平。若输出模式下读取数据时，则读取的是输出锁存器的值。

可用 1 位或 8 位存储器操作指令设置这些寄存器。

复位信号产生时，这些寄存器清为 00H。

图 4-16. 端口寄存器格式

符号	7	6	5	4	3	2	1	0	地址	复位后	R/W
P2	P27 ^{注1}	P26 ^{注1}	P25 ^{注1}	P24 ^{注1}	P23 ^{注1}	P22 ^{注1}	P21 ^{注1}	P20 ^{注1}	FF02H	00H (输出锁存器)	R/W
P3	0	0	0	P34	P33 ^{注1}	P32 ^{注1}	P31	P30	FF03H	00H (输出锁存器)	R/W
P12	0	0	P125	0	0	P122 ^{注2}	P121 ^{注2}	0	FF0CH	00H (输出锁存器)	R/W

Pmn	m = 2, 3, 12; n = 0 至 7	
	输出数据控制 (输出模式下)	输入数据读取 (输入模式下)
0	输出 0	输入低电平
1	输出 1	输入高电平

注 1. 该引脚设置为模拟输入和输出模式的情况下，禁止访问输出锁存器。

2. 在 X1 振荡器模式或外部时钟输入模式下，输出锁存器引脚的值始终为“0”。

(3) 上拉电阻选择寄存器 (PUxx)

这些寄存器指定是否使用内置上拉电阻。在上拉电阻选择寄存器中，已指定内置上拉电阻的端口，只限于已设置输入模式的引脚，可按位单元使用内置上拉电阻。即使这些寄存器已被设置，内置上拉电阻也不能连接设置为输出模式和用作复用功能的输出引脚。

可用 1 位或 8 位存储器操作指令设置这些寄存器。

复位信号产生时，这些寄存器清零 (00H) (仅限 PU12 至 20H 的设置)。

图 4-17. 上拉电阻选择寄存器

符号	7	6	5	4	3	2	1	0	地址	复位后	R/W
PU3	0	0	0	PU34	PU33	PU32	PU31	PU30	FF33H	00H	R/W
PU12	0	0	PU125	0	0	0	0	0	FF3CH	20H	R/W

PUmn	Pmn 引脚内置上拉电阻选择 (m = 3, 12; n = 0 至 5)
0	没有连接内置上拉电阻
1	连接内置上拉电阻

(4) 复位引脚模式寄存器(RSTMASK)

该寄存器设置 RESET/P125 的引脚功能（外部复位输入/输入专用端口）。

可用 1 位或 8 位存储器操作指令设置该寄存器。

复位信号产生时，该寄存器清为 00H。

图 4-18. 复位引脚模式寄存器 (RSTMASK) 格式

地址: FF2DH 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
RSTMASK	0	0	RSTM	0	0	0	0	0

RSTM	RESET/P125 引脚功能选择
0	用作外部复位输入(RESET)
1	用作输入专用端口(P125)

(5) A/D 端口配置寄存器(ADPC)

ADPC 将 P20/AMP0-/ANI0 至 P27/AMP1+引脚切换到数字 I/O 端口或模拟 I/O 端口。可按 1 位单元指定 ADPC 每个位相对应的端口 2 的一个引脚。

可用 1 位或 8 位存储器操作指令设置该寄存器。

复位信号产生时，ADPC 清为 00H。

图 4-19. A/D 端口配置寄存器(ADPC)格式

地址: FF97H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
ADPC	ADPC7	ADPC6	ADPC5	ADPC4	ADPC3	ADPC2	ADPC1	ADPC0

ADPCn	数字 I/O 或模拟 I/O 选择(n = 0 至 7)
0	模拟 I/O
1	数字 I/O

- 注意事项**
1. 通过端口模式寄存器 2(PM12)，将设置为模拟输入的引脚转换成输入模式。
 2. 若将数据写入 ADPC，则产生一个等待周期。当外部硬件时钟停止时，禁止将数据写入 ADPC。详阅第二十六章 等待注意事项。

4.4 端口功能操作

端口操作取决于是否已设置输入或输出模式，具体如下所示。

4.4.1 写入 I/O 端口

(1) 输出模式

通过传送指令将数值写入输出锁存器，并将输出锁存器的内容从引脚输出。

数据一旦被写入输出锁存器，就可以保存到写入新的数据被写入为止。

复位信号产生时，输出锁存器的数据清零。

(2) 输入模式

通过传送指令将数值写入输出锁存器时，由于输出缓冲器处于关闭状态，引脚状态不会改变。

数据一旦被写入输出锁存器，就能够保存到新的数据被写入为止。

复位信号产生时，输出锁存器的数据清零。

4.4.2 读取 I/O 端口

(1) 输出模式

通过传送指令读取输出锁存器的内容。不会改变输出锁存器的内容。

(2) 输入模式

通过传送指令读取引脚状态。不会改变输出锁存器的内容。

4.4.3 操作 I/O 端口

(1) 输出模式

对输出锁存器的内容执行操作时，结果写入输出锁存器。从引脚输出锁存器的内容。

数据一旦被写入输出锁存器，就可以保存到新的数据被写入为止。

复位信号产生时，输出锁存器的数据。

(2) 输入模式

读取引脚电平，对其内容执行操作。操作结果写入输出锁存器，但由于输出缓冲器处于关闭状态，不会改变引脚状态。

复位信号产生时，输出锁存器的数据清零。

4.5 使用复用功能时，端口模式寄存器和输出锁寄存器的设置

用作端口引脚的复用功能时，应按表 4-9 所示对端口模式寄存器和输出锁寄存器进行设置。

表 4-9. 使用复用功能时，端口模式寄存器和输出锁寄存器的设置(1/2)

引脚名称	复用功能		PM _{xx}	P _{xx}
	功能名称	I/O		
P20	ANI0 [※]	输入	1	P20
	AMP0- [※]	输入	1	
P21	ANI1 [※]	输入	1	P21
	AMP0OUT [※]	输出	1	
P22	ANI2 [※]	输入	1	P22
	AMP0+ [※]	输入	1	
P23, P24	ANI3, ANI4 [※]	输入	1	P23, P24
P25	AMP1-	输入	1	P25
P26	AMP1OUT	输出	1	P26
P27	AMP1+	输入	1	P27
P30	INTP0	输入	1	P30
	TI51	输入	1	
	TOH1	输出	0	
P31	TxD0	输出	0	P31
	CMPCOM	输入	1	
P32	RxD0	输入	1	P32
	CMPIN	输入	1	
P33	INTP1	输入	1	P33
	TI000	输入	1	
P34	TO00	输入	1	P34
	TI010	输出	0	
	CMPOUT	输出	1	

备注 可通过使用 ADPC 寄存器、PM2 寄存器、ADS 寄存器和 AMPH 寄存器选择引脚功能。参阅 4.2.1 端口 2 的表 4-4 至 4-8。

备注 ×: 忽略
 PM_{xx}: 端口模式寄存器
 P_{xx}: 端口输锁寄存器

表 4-9. 使用复用功能时，端口模式寄存器和输出锁存器的设置(2/2)

引脚名称	复用功能		PM _{xx}	P _{xx}
	功能名称	I/O		
P121	X1 ^{註1}	—	×	×
	TOOLC0	输入	×	×
P122	X2 ^{註1}	—	×	×
	EXCLK ^{註1}	输入	×	×
	TOOLD0	I/O	×	×
P125	RESET ^{註2}	输入	×	×

- 注 1. 使用 P121 和 P122 引脚连接谐振器到主系统时钟(X1,X2)器或输入外部时钟到主系统时钟(EXCLK)时，必须使用时钟操作模式寄存器(OSCCTL)设置 X1 振荡模式或外部时钟输入模式(详阅 5.3(1)时钟操作模式选择寄存器(OSCCTL))。OSCCTL 复位值为 00H (P121 和 P122 引脚均为输入端口引脚)。
2. 当 P125 用作外部复位输入($\overline{\text{RESET}}$)时，RSTM 位 (RSTMASK 寄存器第 5 位) 清零。

备注

×: 忽略

PM_{xx}: 端口模式寄存器

P_{xx}: 端口输出锁存器

4.6 端口寄存器 n(Pn)1 位操作指令的注意事项

在兼有输入和输出功能的端口执行 1 位操作指令时，目标位被写入的同时，非目标操作的输入端口的输出锁存值也有可能被改写。

因此，当端口由输入模式切换为输出模式时，建议重写输出锁存器。

<例> 当 P20 为输出端口，P21 至 P27 为输入端口（所有引脚状态为高电平），端口 1 的端口锁存值为 00H 时，若通过 1 位操作指令将输出端口 P20 的输出从低电平跳变到高电平时，则端口 2 的输出锁存值为 FFH。

说明：PMnm 位=1 端口 Pn 寄存器的写入目标为输出锁存，读取目标为引脚状态。
在 μPD79F7023, 79F7024 微控制器，按以下顺序执行 1 位操作指令。

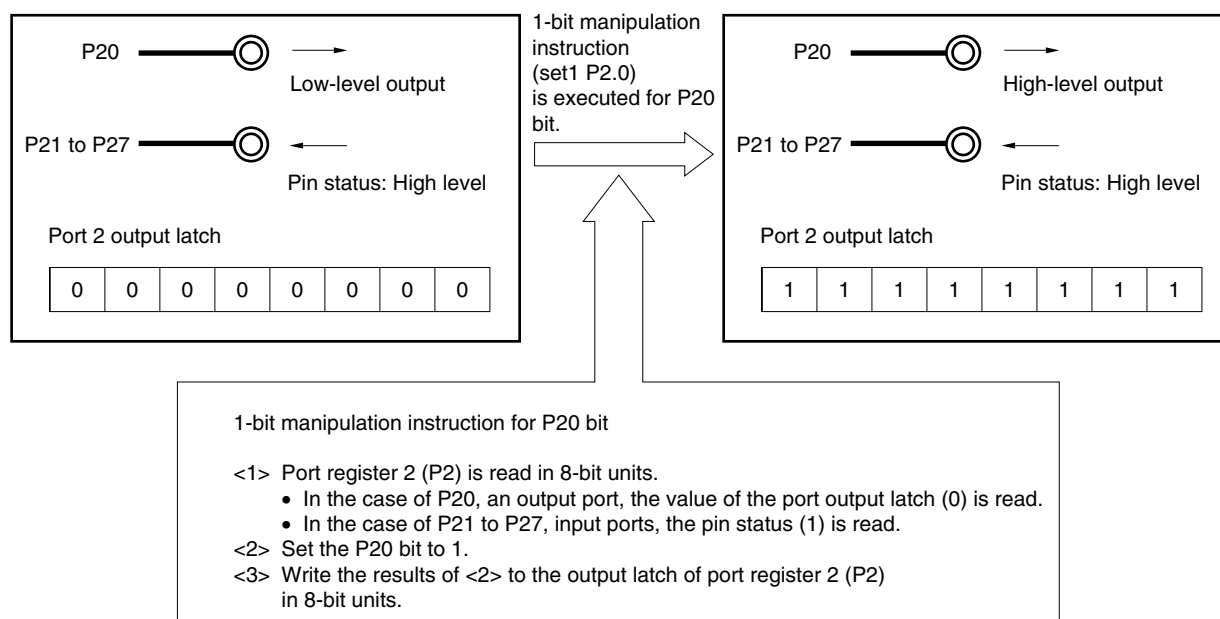
- <1> 按 8 位单元读取 Pn 寄存器。
- <2> 操作目标的 1 位。
- <3> 按 8 位单元写入 Pn 寄存器。

在步骤<1>中，读取输出端口 P20 的输出锁存值的同时，也能读取输入端口 P21 至 P27 的引脚状态。此时，若 P21 至 P27 的引脚状态是高电平，则读出的值将为 FEH。

通过步骤<2>操作把值改为 FFH。

在步骤 <3>时，通过位操作将 FFH 写入输出锁存器。

图 4-20 1 位操作指令(P20)



备注 以下指令为 1 位操作指令。

- MOV1, AND1, OR1, XOR1, SET1, CLR1, NOT1

第五章 时钟发生电路

5.1 时钟发生电路功能

通过时钟发生电路将产生的时钟提供给 CPU 和外围硬件。

以下三种系统时钟和系统振荡器可供选择。

(1) 主系统时钟

<1> X1 振荡电路

通过连接一台振荡器到 X1 和 X2, 该振荡电路产生 $f_x = 1$ 至 10 MHz 的时钟。

通过执行 STOP 指令或使用主 OSC 控制寄存器(MOC), 可以停止其振荡。

<2> 内部高速振荡电路

该振荡电路产生 $f_{IH} = 4$ MHz (典型值)的时钟。复位解除后, CPU 始终使用该内部高速振荡时钟开始操作。通过执行 STOP 指令或使用内部振荡模式寄存器(RCM), 可以停止其振荡。

通过 EXCLK/X2/P122 引脚, 可以提供外部主系统时钟($f_{EXCLK} = 1$ 至 10 MHz)。通过执行 STOP 指令或使用 RCM, 可以禁止外部主系统时钟输入。

用作主系统时钟时, 可以由主时钟模式(MCM)选择高速系统时钟(X1 时钟或外部主系统时钟)或内部高速振荡时钟。

备注	f_x :	X1 时钟振荡频率
	f_{IH} :	内部高速振荡时钟频率
	f_{EXCLK} :	外部主系统时钟频率

(2) 内部低速振荡时钟(看门狗定时器时钟)

• 内部低速振荡电路

该振荡电路产生 $f_{IL} = 240$ kHz(典型值)的时钟。复位解除后, 始终使用内部低速振荡时钟开始操作。

通过选项字节设置“使用软件停止内部低速振荡器”时, 可以用内部振荡模式寄存器(RCM)停止振荡。

内部低速振荡时钟不可用作 CPU 时钟。使用内部低速振荡时钟操作以下硬件。

- 看门狗定时器
- 8 位定时器 H1(选择 f_{IL} , $f_{IL}/2^7$ 或 $f_{IL}/2^9$ 时)

备注	f_{IL} : 内部低速振荡时钟频率
----	-----------------------

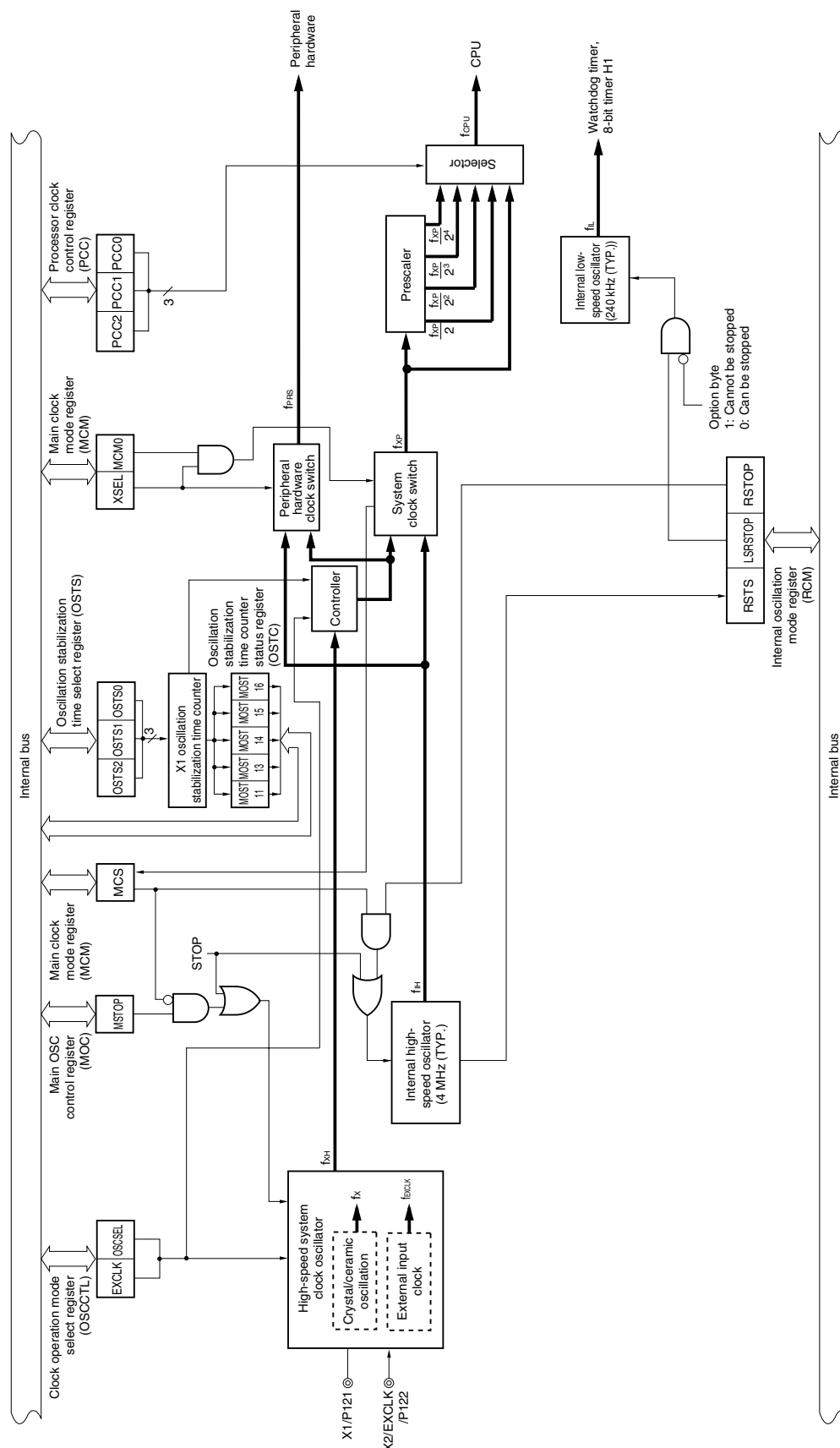
5.2 时钟发生电路的配置

时钟发生电路包含以下硬件。

表 5-1. 时钟发生电路的配置

项目	配置
控制寄存器	时钟操作模式选择寄存器(OSCCTL) 处理器时钟控制寄存器(PCC) 内部振荡模式寄存器(RCM) 主 OSC 控制寄存器(MOC) 主时钟模式寄存器(MCM) 振荡稳定时间计数器状态寄存器(OSTC) 振荡稳定时间选择寄存器(OSTS)
振荡电路	X1 振荡器 内部高速振荡器 内部低速振荡器

图 5-1.时钟发生器系统图



备注	fx:	X1 时钟振荡频率
	fIH:	内部高速振荡时钟频率
	fEXCLK:	外部主系统时钟频率
	fxH:	高速系统时钟频率
	fXP:	主系统时钟频率
	fPRS:	外围硬件时钟频率
	fCPU:	CPU 时钟频率
	fIL:	内部低速振荡时钟频率

5.3 时钟发生电路的控制寄存器

通过以下 7 台寄存器控制时钟发生电路。

- 时钟操作模式选择寄存器 (OSCCTL)
- 处理器时钟控制寄存器(PCC)
- 内部振荡模式寄存器(RCM)
- 主 OSC 控制寄存器(MOC)
- 主时钟模式寄存器(MCM)
- 振荡稳定时间计数器状态寄存器(OSTS)
- 振荡稳定时间选择寄存器(OSTS)

(1) 时钟操作模式选择寄存器(OSCCTL)

该寄存器用来选择高速系统时钟的操作模式。

可按 1 位或 8 位存储器操作指令设置 OSCCTL。

复位信号产生时，该寄存器清为 00H。

图 5-2. 时钟操作模式选择寄存器(OSCCTL)的格式

地址: FF9FH 复位后: 00H R/W

符号	<7>	<6>	5	4	3	2	1	0
OSCCTL	EXCLK	OSCSEL	0	0	0	0	0	0

EXCLK	OSCSEL	高速系统时钟引脚操作模式	P121/X1 引脚	P122/X2/EXCLK 引脚
0	0	输入端口模式	输入端口	
0	1	X1 振荡模式	连接晶体/陶瓷谐振器	
1	0	输入端口模式	输入端口	
1	1	外部时钟输入模式	输入端口	外部时钟输入

- 注意事项**
- 若要修改 EXCLK 和 OSCSEL，则必须确保主 OSC 控制寄存器(MOC)的第 7 位(MSTOP)置“1”(X1 振荡器停止或禁止使用来自 EXCLK 引脚的外部时钟)。
 - 确保第 0 位至第 5 位清零。

(2) 处理器时钟控制寄存器(PCC)

该寄存器用于选择 CPU 时钟和分频比。

可按 1 位或 8 位存储器操作指令设置 PCC。

复位信号产生时, PCC 设置为 01H。

图 5-3. 处理器时钟控制寄存器(PCC)格式

地址: FFFBH 复位后: 01H R/W

符号	7	6	5	4	3	2	1	0
PCC	0	0	0	0	0	PCC2	PCC1	PCC0

PCC2	PCC1	PCC0	CPU 时钟(f_{CPU})选择
0	0	0	f_{XP}
0	0	1	$f_{XP}/2$ (默认)
0	1	0	$f_{XP}/2^2$
0	1	1	$f_{XP}/2^3$
1	0	0	$f_{XP}/2^4$
其它			禁止设置

- 注意事项**
1. 必须将第 3 位至第 7 位清零。
 2. 设置 PCC 的分频比时,外围硬件时钟(f_{PRS})不分频。

备注 f_{XP} : 主系统时钟振荡频率

在 μPD79F7023, 79F7024 微微控制器中最快的指令执行时间在 2 个 CPU 时钟以内。

CPU 时钟 f_{CPU} 与最短指令执行时间之间的关系如表 5-2 所示。

表 5-2. CPU 时钟与最短指令执行时间之间的关系

CPU 时钟(f_{CPU})	最短指令执行时间: $2/f_{CPU}$	
	主系统时钟	
	高速系统时钟 ^注	内部高速振荡时钟 ^注
	操作频率: 10 MHz	操作频率: 4 MHz (典型值) 操作
f_{XP}	0.2 μs	0.5 μs (典型值)
$f_{XP}/2$	0.4 μs	1.0 μs (典型值)
$f_{XP}/2^2$	0.8 μs	2.0 μs (典型值)
$f_{XP}/2^3$	1.6 μs	4.0 μs (典型值)
$f_{XP}/2^4$	3.2 μs	8.0 μs (典型值)

注 主时钟模式寄存器(MCM)用于设置提供给 CPU 时钟的主系统时钟(高速系统时钟/内部高速振荡时钟)(参阅图 5-3)。

(3) 内部振荡模式寄存器(RCM)

该寄存器设置内部振荡器的操作模式。

可按 1 位或 8 位存储器操作指令设置 RCM。

复位信号产生时, 该寄存器设置为 80H^{注1}。

图 5-4. 内部振荡模式寄存器(RCM)的格式

地址: FFA0H 复位后: 80H^{注1} R/W^{注2}

符号	<7>	6	5	4	3	2	<1>	<0>
RCM	RSTS	0	0	0	0	0	LSRSTOP	RSTOP

RSTS	内部高速振荡器状态
0	等待内部高速振荡器精确稳定
1	内部高速振荡器稳定操作

LSRSTOP	内部低速振荡器振荡/停止
0	内部低速振荡器振荡
1	内部低速振荡器停止

RSTOP	内部高速振荡器振荡/停止
0	内部高速振荡器振荡
1	内部高速振荡器停止

注 1. 复位解除后的寄存器值改变为 00H, 但等到内部高速振荡器稳定后会自动恢复到 80H。

2. 第 7 位只读。

注意事项 当 RSTOP 置“1”时,必须确保 CPU 时钟使用内部高速振荡时钟以外的时钟进行操作。特别在以下条件下:

- 当 MCS = 1 时 (CPU 使用高速系统时钟)

此外, RSTOP 置“1”之前, 应停止正在使用内部高速振荡时钟的外围硬件。

(4) 主 OSC 控制寄存器(MOC)

该寄存器用来选择高速系统时钟的操作模式。

当 CPU 使用高速系统时钟以外的时钟进行操作时，该寄存器用来停止 X1 振荡器或禁止来自 EXCLK 引脚的外部时钟输入。

可按 1 位或 8 位存储器操作指令设置 MOC。

复位信号产生时，该寄存器设置为 80H。

图 5-5.主 OSC 控制寄存器(MOC)的格式

地址：FFA2H 复位后：80H R/W

符号	<7>	6	5	4	3	2	1	0
MOC	MSTOP	0	0	0	0	0	0	0

MSTOP	高速系统时钟操作控制	
	X1 振荡模式	外部时钟输入模式
0	X1 振荡器操作	允许来自 EXCLK 引脚的外部时钟
1	X1 振荡器停止	禁止来自 EXCLK 引脚的外部时钟

- 注意事项**
1. 当寄存器模式控制寄存器(RMC)为 00H 时，MSTOP 清零。
 2. 当 RSTOP 置“1”时,必须确保 CPU 使用内部高速振荡时钟以外的时钟进行操作。特别在以下条件下：
 - 当 MCS = 0 时 (CPU 使用内部高速振荡时钟)
 此外，RSTOP 置“1”之前，应停止正在使用高速系统时钟的外围硬件。
 3. 当时钟操作模式选择寄存器(OSCCTL)的第 6 位(OSCSEL)为 0（输入端口模式）时，禁止 MSTOP 清零。
 4. 当外围硬件时钟停止时，不能操作外围硬件。外围硬件时钟停止后，若要恢复外围硬件的操作，则需要进行外围硬件的初始化。

(5) 主时钟模式寄存器(MCM)

该寄存器用来选择提供给 CPU 时钟的主系统时钟和外围硬件时钟的时钟。

可按 1 位或 8 位存储器操作指令设置 MCM。

复位信号产生时, 该寄存器清为 00H。

图 5-6. 主时钟模式寄存器(MCM)的格式

地址: FFA1H 复位后: 00H R/W[※]

符号	7	6	5	4	3	<2>	<1>	<0>
MCM	0	0	0	0	0	XSEL	MCS	MCM0

XSEL	MCM0	提供给主系统时钟和外围硬件的时钟选择	
		主系统时钟(f _{XP})	外围硬件时钟(f _{PRS})
0	0	内部高速振荡时钟 (f _{IH})	内部高速振荡时钟 (f _{IH})
0	1		高速系统时钟(f _{XH})
1	0		
1	1	高速系统时钟(f _{XH})	

MCS	主系统时钟状态
0	使用内部高速振荡时钟
1	使用高速系统时钟

备注 第 1 位只读。

注意事项 1. 复位解除后, XSEL 只能修改一次。

2. 无论 XSEL 和 MCM0 的设置如何, 时钟 (f_{PRS} 除外) 提供以下外围功能。

- 看门狗定时器 (使用内部低速振荡时钟)
- 选择“f_{IL}”, “f_{IL}/2”或“f_{IL}/2⁹”用作 8 位定时器H1 的计数时钟 (使用内部低速振荡时钟)
- 选择外部时钟用作外围硬件的时钟源
(选择 TM00 外部计数时钟 (TI000 引脚有效沿) 时除外)

(6) 振荡稳定时间计数器状态寄存器(OSTC)

该寄存器显示 X1 时钟振荡稳定时间计数器的计数状态。CPU 时钟使用内部高速振荡时钟开始 X1 时钟振荡时，可以检测出 X1 时钟振荡稳定时间。

可按 1 位或 8 位存储器操作指令设置 OSTC。

复位解除后(通过 RESET 输入、POC、LVI 和 WDT 进行复位)，通过 STOP 指令和 MSTOP(MOC 寄存器第 7 位) = 1，OSTC 清为 00H。

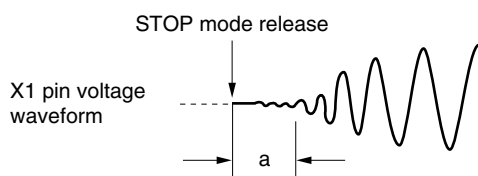
图 5-7. 振荡稳定时间计数器状态寄存器(OSTS)的格式

地址: FFA3H 复位后: 00H R

符号	7	6	5	4	3	2	1	0
OSTC	0	0	0	MOST11	MOST13	MOST14	MOST15	MOST16

MOST11	MOST13	MOST14	MOST15	MOST16	振荡稳定时间状态	
						$f_x = 10 \text{ MHz}$
1	0	0	0	0	$2^{11}/f_x \text{ min.}$	204.8 $\mu\text{s min.}$
1	1	0	0	0	$2^{13}/f_x \text{ min.}$	819.2 $\mu\text{s min.}$
1	1	1	0	0	$2^{14}/f_x \text{ min.}$	1.64 ms min.
1	1	1	1	0	$2^{15}/f_x \text{ min.}$	3.27 ms min.
1	1	1	1	1	$2^{16}/f_x \text{ min.}$	6.55 ms min.

- 注意事项**
1. 经过上述时间后，从 MOST11 开始位逐次置“1”，并一直保持为“1”。
 2. 振荡稳定时间计数器只能检测出由 OSTC 设置的振荡稳定时间。内部高速振荡时钟用作 CPU 时钟时，若进入 STOP 模式，则解除其模式时，需按以下方式设置振荡稳定时间。
 - 预期 OSTC 振荡稳定时间 $\leq$ 由 OSTC 设置的振荡稳定时间
 因此，应注意 STOP 模式解除后的 OSTC，只能达到 OSTC 设置的振荡稳定时间状态。
 3. X1 时钟振荡稳定等待时间不包括时钟振荡状态之前的时间（下图中“a”所表示的部分）。



备注 f_x : X1 时钟振荡频率

(7) 振荡稳定时间选择寄存器(OSTS)

STOP 模式解除时，该寄存器用来选择 X1 时钟的振荡稳定等待时间。

当 CPU 时钟选用 X1 时钟时，STOP 模式解除后，需等待由 OSTS 所设置的时间。

当 CPU 时钟选用内部高速振荡时钟时，STOP 模式解除后，使用 OSTC 确认是否已达到所需的振荡稳定时间。通过 OSTC 设置时间，可检测出振荡稳定时间。

可按 1 位或 8 位存储器操作指令设置 OSTS。

复位信号产生时，OSTS 设置为 05H。

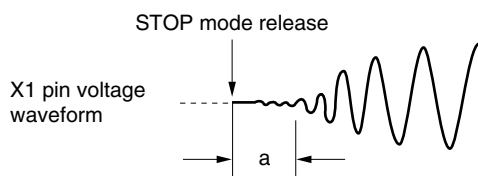
图 5-8. 振荡稳定时间选择寄存器(OSTS)的格式

地址: FFA4H 复位后: 05H R/W

符号	7	6	5	4	3	2	1	0
OSTS	0	0	0	0	0	OSTS2	OSTS1	OSTS0

OSTS2	OSTS1	OSTS0	振荡稳定时间选择	
				$f_x = 10 \text{ MHz}$
0	0	1	$2^{11}/f_x$	204.8 μs
0	1	0	$2^{13}/f_x$	819.2 μs
0	1	1	$2^{14}/f_x$	1.64 ms
1	0	0	$2^{15}/f_x$	3.27 ms
1	0	1	$2^{16}/f_x$	6.55 ms
其它			禁止设置	

- 注意事项
1. 若 CPU 使用 X1 时钟时，设置 STOP 模式，则需执行 STOP 指令之前设置 OSTS。
 2. X1 时钟振荡稳定时间内，不要改变 OSTS 的值。
 3. 振荡稳定时间计数器只能检测出由 OSTS 设置的振荡稳定时间。内部高速振荡时钟用作 CPU 时钟时，若进入 STOP 模式，则解除其模式时，需按以下方式设置振荡稳定时间。
 - 预期 OSTC 振荡稳定时间 $\leq$ 由 OSTS 设置的振荡稳定时间
 因此，应注意 STOP 模式解除后的 OSTS，只能达到 OSTS 所设置的振荡稳定时间的状态。
 4. 1 时钟振荡稳定等待时间不包括时钟振荡状态之前的时间（下图中“a”所表示的部分）。



备注 f_x : X1 时钟振荡频率

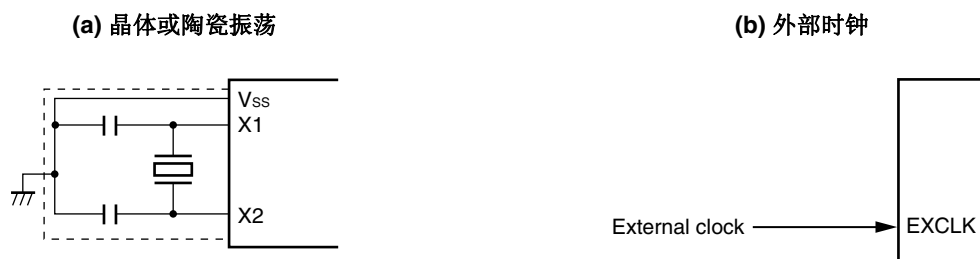
5.4 系统时钟振荡电路

5.4.1 X1 振荡电路

X1 振荡电路通过连接到 X1 和 X2 引脚的晶体谐振器或陶瓷谐振器(1 至 10 MHz)进行振荡。也可输入外部时钟。此时, 从 EXCLK 引脚输入时钟信号。

X1 振荡电路的外部电路示例如图 5-9 所示。

图 5-9.X1 振荡器的外部电路示例



注意事项 使用 X1 振荡器时, 在图 5-10 所示的虚线框内进行布线, 从而避免布线电容的副作用。

- 布线要尽量短。
- 振荡器接线不与其它信号线交叉。布线应远离高脉冲电流信号线。
- 始终保持振荡器电容的接地点与 VSS 相同。禁止连接电容的接地点至高电流的接地模式。
- 禁止从振荡器获取信号。

谐振器错误连接示例如图 5-10 所示。

图 5-10. 谐振器错误连接示例 (1/2)

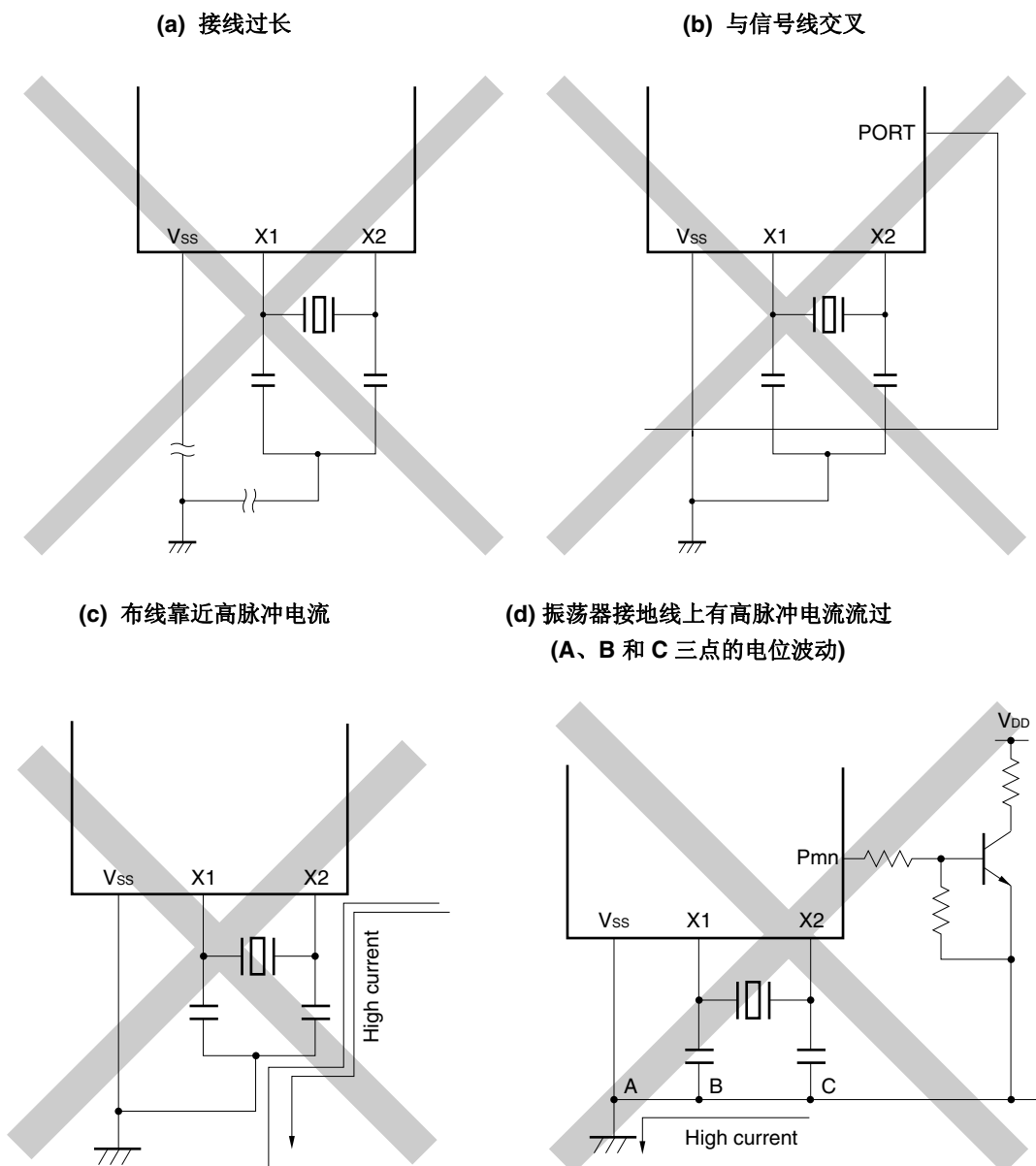
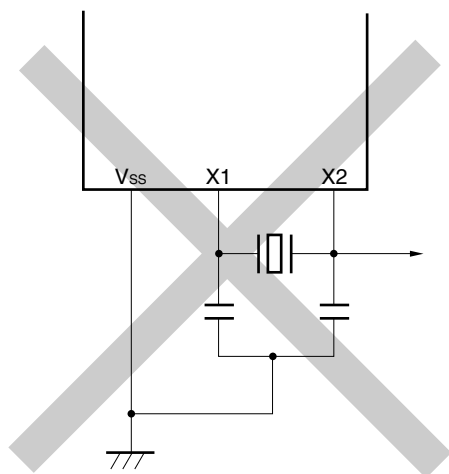


图 5-10. 谐振器错误连接示例 (2/2)

(e) 获取信号



5.4.2 内部高速振荡电路

μPD79F7023, 79F7024 内置内部高速振荡电路。由内部振荡模式寄存器(RCM)控制振荡。
复位解除后, 内部高速振荡电路自动开始振荡。

5.4.3 内部低速振荡电路

μPD79F7023, 79F7024 内置内部低速振荡电路。

内部低速振荡时钟只能用作看门狗定时器和 8 位定时器 H1 时钟。内部低速振荡时钟不能用作 CPU 时钟。

通过选项字节可以选择“可由软件停止”或“不可停止”。设置为“可由软件停止”时, 可用内部振荡模式寄存器(RCM)控制振荡。

复位解除后, 内部低速振荡器自动开始振荡, 若已使用选项字节设置“看门狗定时器操作”, 则启动看门狗定时器 (240 kHz (典型值))。

5.4.4 预分频器

CPU 使用主系统时钟时, 通过预分频器分频主系统时钟, 产生时钟。

5.5 时钟发生电路操作

时钟生成电路产生以下时钟,并控制 CPU 的操作模式,例如,待机模式(参阅图 5-1)。

- 主系统时钟 f_{XP}
 - 高速系统时钟 f_{XH}
 - X1 时钟 f_X
 - 外部主系统时钟 f_{EXCLK}
 - 内部高速振荡时钟 f_{IH}
- 内部低速振荡时钟 f_{IL}
- CPU 时钟 f_{CPU}
- 外围硬件时钟 f_{PRS}

在 μ PD79F7023, 79F7024 微微控制器中,当复位解除后,CPU 通过内部高速振荡电路的输出开始操作,并具有以下特点。

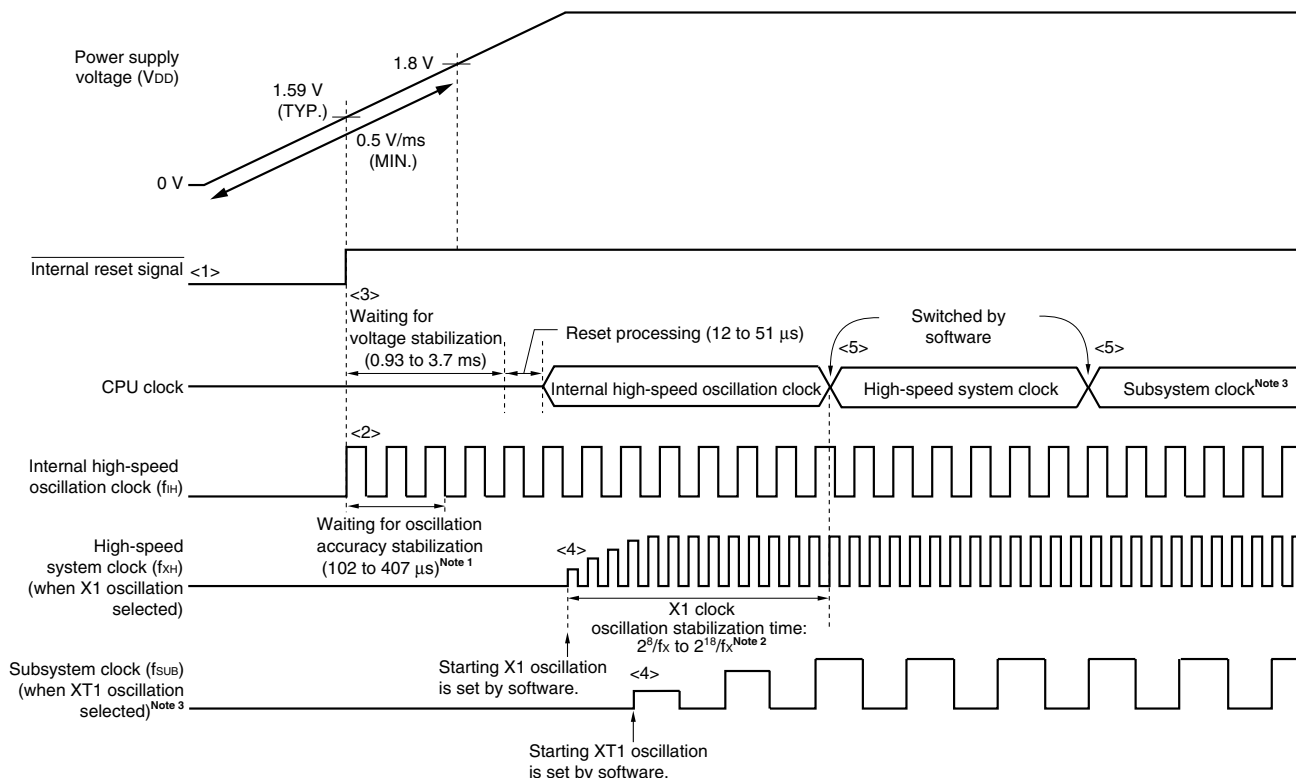
(1) 强化安全功能

X1 时钟默认为 CPU 时钟时,若因损坏或连接不良等原因导致不能操作 X1 时钟时,则复位解除后也不能启动设备。但若 CPU 初始时钟为内部高速振荡时钟,则复位解除后可以用内部高速振荡时钟启动设备。因此,只需执行最少操作,便可以安全关闭系统。例如,发生故障时,使用软件或执行安全程序,应答复位源。

(2) 性能改善

启动 CPU 无需等到 X1 时钟振荡稳定时间结束,总体性能得以进一步改善。
上电时,时钟发生电路的操作如图 5-11 和 5-12 所示。

图 5-11. 上电时的时钟发生电路操作
(LVI 默认启动功能设置为停止模式 (选项字节:LVISTART = 0))



- <1> 上电时，通过上电清除(POC)电路产生内部复位信号。
- <2> 供电电压超过 1.59 V(典型值)时，解除复位后，内部高速振荡器自动开始振荡。
- <3> 供电电压上升斜率为 0.5V/ms(最小值)时，复位解除并且等到电源和稳压器的电压稳定时间结束后，CPU 开始启动内部高速振荡时钟，然后执行复位处理。
- <4> 通过软件设置 X1 时钟开始振荡。(参阅 5.6.1 控制高速系统时钟示例之(1))。
- <5> CPU 时钟切换到 X1 时钟时，等到时钟振荡稳定后，通过软件设置切换(参阅 5.6.1 控制高速系统时钟示例之(3))。

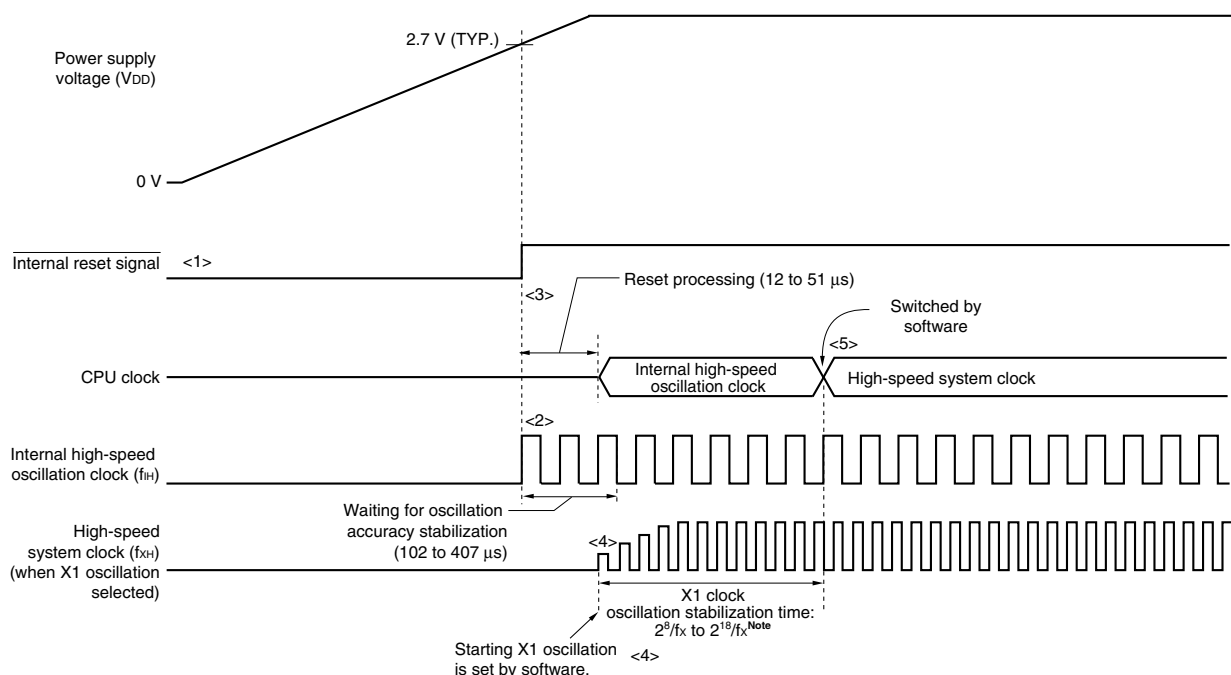
- 注 1. 内部电压稳定时间包括内部高速振荡时钟振荡的精确稳定时间。
2. 复位解除（如上图所示）或 CPU 正在执行内部高速振荡时钟过程中解除 STOP 模式时，使用振荡稳定事件计数器的状态寄存器(OSTC)确认 X1 时钟的振荡稳定时间。若 CPU 执行高速系统时钟(X1 振荡)，则使用振荡稳定时间选择寄存器(OSTS)设置解除 STOP 模式后的振荡稳定时间。

注意事项 1. 启动电压达到 2.7V 之前, 若电压上升斜率低于 0.5V/ms(最小值)时, 则输入低电平到 $\overline{\text{RESET}}$ 引脚直至电压达到 2.7V 为止, 或通过选项字节(LVISTART = 1)设置 LVI 默认启动功能(参阅图 5-12)。输入低电平至 $\overline{\text{RESET}}$ 引脚直至电压达到 2.7V 时, 通过 $\overline{\text{RESET}}$ 引脚解除复位后, CPU 的操作与图 5-11 的<2>以下时序相同。

2. 使用 EXCLK 引脚的外部时钟输入时, 无需等到振荡稳定时间结束。

备注 微微控制器运行时, 通过软件设置可以停止时钟 (CPU 时钟除外)。通过执行 STOP 指令, 可以停止内部高速振荡时钟和高速系统时钟 (参阅 5.6.1 控制高速系统时钟示例之(4) 和 5.6.2 控制内部高速振荡时钟之(3))。

图 5-12. 上电时的时钟发生电路操作
(LVI 默认启动功能设置为启动模式 (选项字节: LVISTART = 1))



<1> 上电时, 通过上电清除(POC)电路产生内部复位信号。

<2> 供电电压超过 2.7 V(典型值)时, 复位解除, 内部高速振荡器自动开始振荡。

<3> 复位解除并执行复位处理后, CPU 开始执行内部高速振荡时钟。

<4> 通过软件设置 X1 时钟开始振荡。(参阅 5.6.1 控制高速系统时钟示例之(1))。

<5> CPU 时钟切换到 X1 时钟时, 等到时钟振荡稳定后, 通过软件设置切换(参阅 5.6.1 控制高速系统时钟示例之(3))。

注 复位解除（如上图所示）或 CPU 正在执行内部高速振荡时钟过程中解除 STOP 模式时，使用振荡稳定事件计数器的状态寄存器(OSTC)确认 X1 时钟的振荡稳定时间。若 CPU 执行高速系统时钟(X1 振荡)，则使用振荡稳定时间选择寄存器(OSTS)设置解除 STOP 模式后的振荡稳定时间。

注意事项 1. 电源电压达到 1.59V(典型值)后，所需电压振荡稳定时间为 0.93 至 3.7ms。若在电压稳定等待时间内电源电压从 1.59V(典型值)上升到 2.7V(典型值)，则在复位处理之前自动生成电源稳定等待时间。
2. 使用 EXCLK 引脚的外部时钟输入时，无需等到振荡稳定时间结束。

备注 微微控制器运行时，通过软件设置可以停止时钟（CPU 时钟除外）。通过执行 STOP 指令，可以停止内部高速振荡时钟和高速系统时钟（参阅 5.6.1 控制高速系统时钟示例之(4) 和 5.6.2 控制内部高速振荡时钟示例之(3)）。

5.6 控制时钟

5.6.1 控制高速系统时钟示例

提供以下两种高速系统时钟。

- X1 时钟：晶体/陶瓷谐振器连接到 X1 和 X2 引脚。
- 外部主系统时钟：外部时钟输入到 EXCLK 引脚。

不使用高速系统时钟时，X1/P121 和 X2/EXCLK/P122 引脚可用作输入端口引脚。

注意事项 复位解除后，X1/P121 和 X2/EXCLK/P122 引脚处于输入端口模式。

在下列情况下的设置步骤示例如下所述。

- (1) X1 时钟振荡
- (2) 使用外部主系统时钟
- (3) 高速系统时钟用作 CPU 时钟和外围软件时钟
- (4) 高速系统时钟停止

(1) X1 时钟振荡时的设置步骤示例

<1> 设置 P121/X1 和 P122/X2/EXCLK 引脚，选择 X1 时钟或外部时钟(OSCCTL 寄存器)

当 EXCLK 清零和 OSCSEL 置“1”时，从端口模式切换到 X1 振荡模式。

EXCLK	OSCSEL	高速系统时钟引脚的操作模式	P121/X1 引脚	P122/X2/EXCLK 引脚
0	1	X1 振荡模式	连接晶体/陶瓷谐振器	

<2> 控制 X1 时钟振荡(MOC 寄存器)

若 MSTOP 清零，则 X1 振荡器开始振荡。

<3> 等待 X1 时钟振荡稳定

检测 OSTC 寄存器并等待所需的时间。

在等待时间内，可以使用内部高速振荡时钟执行其它软件处理。

注意事项 1. 在 X1 时钟运行时，禁止改变 EXCLK 和 OSCSEL 的值。

2. 电源电压达到时钟所需的工作电压后，设置 X1 时钟(参阅第二十四章 电特性)。

(2) 使用外部主系统时钟时的设置步骤示例

<1> 设置 P121/X1 和 P122/X2/EXCLK 引脚，选择操作模式(OSCCTL 寄存器)

当 EXCLK 和 OSCSEL 置“1”时，从端口模式切换到外部时钟输入模式。

EXCLK	OSCSEL	高速系统时钟引脚的操作模式	P121/X1 引脚	P122/X2/EXCLK 引脚
1	1	外部时钟输入模式	输入端口	外部时钟输入

<2> 控制外部主系统时钟输入(MOC 寄存器)

MSTOP 清零时，允许外部指系统时钟输入。

注意事项 1. 在运行外部主系统时钟时，禁止改变 EXCLK 和 OSCSEL 的值。

2. 电源电压达到时钟所需的工作电压后，设置外部主系统时钟(参阅第二十四章 电特性)。

(3) 高速系统时钟用作 CPU 时钟和外围软件时钟时的设置步骤示例

<1> 设置高速系统时钟振荡[※]

(参阅 5.6.1 (1) X1 时钟振荡时的设置步骤示例 和 (2) 使用外部主系统时钟时的设置步骤示例。)

注 运行高速系统时钟时,不需置“1”。

<2> 设置高速系统时钟为主系统时钟(MCM 寄存器)

XSEL 和 MCM0 置“1”时，高速系统时钟用作主系统时钟和外部硬件时钟。

XSEL	MCM0	选择主系统时钟和外围硬件时钟	
		主系统时钟 (f _{XP})	外围硬件时钟(f _{PRS})
1	1	高速系统时钟 (f _{XH})	高速系统时钟 (f _{XH})

注意事项 若主系统时钟选用高速系统时钟，则只有高速系统时钟可用作外部硬件时钟。

<3> 设置主系统时钟为 CPU 时钟并选择分频比(PCC 寄存器)

主系统时钟提供给 CPU。可使用 PCC0、PCC1 和 PCC2 选择 CPU 时钟分频比。

PCC2	PCC1	PCC0	CPU 时钟(f _{CPU})选择
0	0	0	f _{XP}
0	0	1	f _{XP} /2 (默认)
0	1	0	f _{XP} /2 ²
0	1	1	f _{XP} /2 ³
1	0	0	f _{XP} /2 ⁴
其它			禁止设置

(4) 停止高速系统时钟时的设置步骤示例

可以使用以下两种方法，停止高速系统时钟。

- 执行 STOP 指令，停止 X1 振荡（若使用外部时钟，则禁止时钟输入）。
- MSTOP 置“1”，停止 X1 振荡（若使用外部时钟，则禁止时钟输入）。

(a) 执行 STOP 指令

<1> 停止外围硬件的设置

停止 STOP 模式中不能使用的的所有外围硬件（有关停止 STIO 模式中不能使用的的所有外围硬件，参阅第五章 待机功能）

<2> 待机解除后，设置 X1 时钟振荡稳定时间。

CPU 使用 X1 时钟时，在执行 STOP 指令之前设置 OSTS 寄存器的值。

<3> 执行 STOP 指令

执行 STOP 指令时，设置 STOP 模式并停止 X1 振荡（禁止外部时钟输入）。

(b) 通过 MSTOP 置“1”停止 X1 振荡(禁止外部时钟输入)。

<1> 确认 CPU 时钟状态(PCC 和 MCM 寄存器)

通过 CLS 和 MCS 确认 CPU 使用高速系统时钟之外的时钟。

CLS = 0 和 MCS = 1 时, 高速系统时钟提供给 CPU, 因此, CPU 时钟应使用高速系统时钟之外的时钟。

MCS	CPU 时钟状态
0	内部高速振荡时钟
1	高速系统时钟

<2> 停止高速系统时钟(MOC 寄存器)

MSTOP 置“1”时, 停止 X1 振荡(禁止外部时钟输入)。

注意事项 MSTOP 置“1”时, 确保 MCS = 0 或 CLS = 1。此外, 应停止正在使用高速系统时钟的外围硬件。

5.6.2 控制内部高速振荡时钟示例

在下列情况时的时钟设置步骤示例如下所述。

(1) 内部高速振荡时钟重新振荡。

(2) 内部高速振荡时钟用作 CPU 时钟时, 内部高速振荡时钟或高速系统时钟用作外围硬件时钟。

(3) 停止内部高速系统时钟

(1) 内部高速振荡时钟重新振荡时的设置步骤示例^{注1}

(1) 重新启动内部高速振荡时钟振荡的设置。(RCM 寄存器)

RSTOP 清零时, 内部高速振荡时钟开始运行。

<2> 等待内部高速振荡时钟的振荡精确稳定时间(RCM 寄存器)

等到 RSTS 置“1”^{注2}。

注 1. 复位解除后, 内部高速振荡器自动开始振荡, 内部高速振荡时钟选作 CPU 时钟。

2. 若 CPU 时钟和外围硬件时钟不需要精确时间, 则可省略等待时间。

(2) 内部高速振荡时钟用作 CPU 时钟以及内部高速振荡时钟或高速系统时钟用作外围硬件时钟时的设置步骤示例。

<1> • 重新启动内部高速振荡时钟的振荡[※]。

(参阅 5.6.2 (1) 内部高速振荡时钟重新振荡时的设置步骤示例)。

• 高速系统时钟振荡[※]

(高速系统时钟用作外围硬件时钟时, 应进行该设置) (参阅 5.6.1 (1) X1 时钟振荡时的设置步骤示例 和 (2) 使用外部主系统时钟时的设置步骤示例。)

注 内部高速振荡时钟或高速系统时钟已经运行时, 不需要置“1”。

<2> 提供给主系统时钟和外围硬件时钟的时钟选择(MCM 寄存器)

由 XSEL 和 MCM0 设置主系统时钟和外围硬件时钟。

XSEL	MCM0	选择主系统时钟和外围硬件时钟	
		主系统时钟 (f _{XP})	外围硬件时钟(f _{PRS})
0	0	内部高速振荡时钟(f _{IH})	内部高速振荡时钟(f _{IH})
0	1		
1	0		高速系统时钟 (f _{XH})

<3> 选择 CPU 时钟分频比(PCC 寄存器)

主系统时钟提供给 CPU。使用 PCC0、PCC1 和 PCC2, 选择 CPU 时钟分频比。

PCC2	PCC1	PCC0	CPU 时钟(f _{CPU})选择
0	0	0	f _{XP}
0	0	1	f _{XP} /2 (默认)
0	1	0	f _{XP} /2 ²
0	1	1	f _{XP} /2 ³
1	0	0	f _{XP} /2 ⁴
其它			禁止设置

(3) 停止内部高速振荡时钟的设置步骤示例

可以使用以下两种方法，停止内部高速振荡时钟。

- 执行 STOP 指令，设置 STOP 模式。
- RSTOP 置“1”，停止内部高速振荡时钟。

(a) 执行 STOP 指令**<1> 外围硬件的设置**

停止 STOP 模式中不能使用的的所有外围硬件（有关停止 STIO 模式中不能使用的的所有外围硬件，参阅第十章 待机功能）

<2> 待机解除后，设置 X1 时钟振荡稳定时间。

CPU 使用 X1 时钟时，执行 STOP 指令之前设置 OSTS 寄存器的值。STOP 模式解除后，立即操作 CPU 时，MCM0 清零，CPU 时钟切换到内部高速振荡时钟，并确认 RSTS 置“1”。

<3> 执行 STOP 指令

执行 STOP 指令时，设置 STOP 模式并停止内部高速振荡时钟。

(b) 通过 RSTOP 置“1”，停止内部高速振荡时钟**<1> 确认 CPU 时钟状态(PCC 和 MCM 寄存器)**

通过 CLS 和 MCS 确认 CPU 使用高速系统时钟之外的时钟。

CLS = 0 和 MCS = 0 时，内部高速振荡时钟提供给 CPU，因此，CPU 时钟应使用内部高速振荡时钟之外的时钟。

MCS	CPU 时钟状态
0	内部高速振荡时钟
1	高速系统时钟

<2> 停止内部高速振荡时钟(RCM 寄存器)

RSTOP 置“1”时，内部高速振荡时钟停止运行。

注意事项 RSTOP 置“1”时，确保 MCS = 1 或 CLS = 1。此外，停止正在使用内部高速振荡时钟的外围硬件。

5.6.3 控制内部低速振荡时钟示例

内部低速振荡时钟不能用作 CPU 时钟。

该时钟只能操作以下外围硬件。

- 看门狗定时器
- 8 位定时器 H1（若 f_{IL} 选作计数时钟）

此外，可以通过选项字节选择以下操作模式。

- 禁止停止内部低速振荡器。
- 禁止使用软件停止内部振荡器。

复位解除后,内部低速振荡器自动开始振荡，若使用选项字节已设置“看门狗定时器操作”，则启动看门狗定时器 (240 kHz (典型值))。

(1) 停止内部低速振荡时钟的设置步骤示例

<1> LSRSTOP 置“1”(RCM 寄存器)

LSRSTOP 置“1”时，内部低速振荡时钟停止运行。

(2) 内部低速振荡时钟重新振荡时的设置步骤示例

<1> LSRSTOP 清零(RCM 寄存器)

LSRSTOP 清零时，内部低速振荡时钟重新振荡。

注意事项 若通过选项字节选择“禁止停止内部低速振荡”，则不能控制内部低速振荡时钟的振荡。

5.6.4 CPU 和外围硬件所采用时钟

提供给 CPU 与外围硬件的时钟之间关系和寄存器的设置如下表所示。

表 5-3. 提供给 CPU 与外围硬件的时钟和寄存器的设置

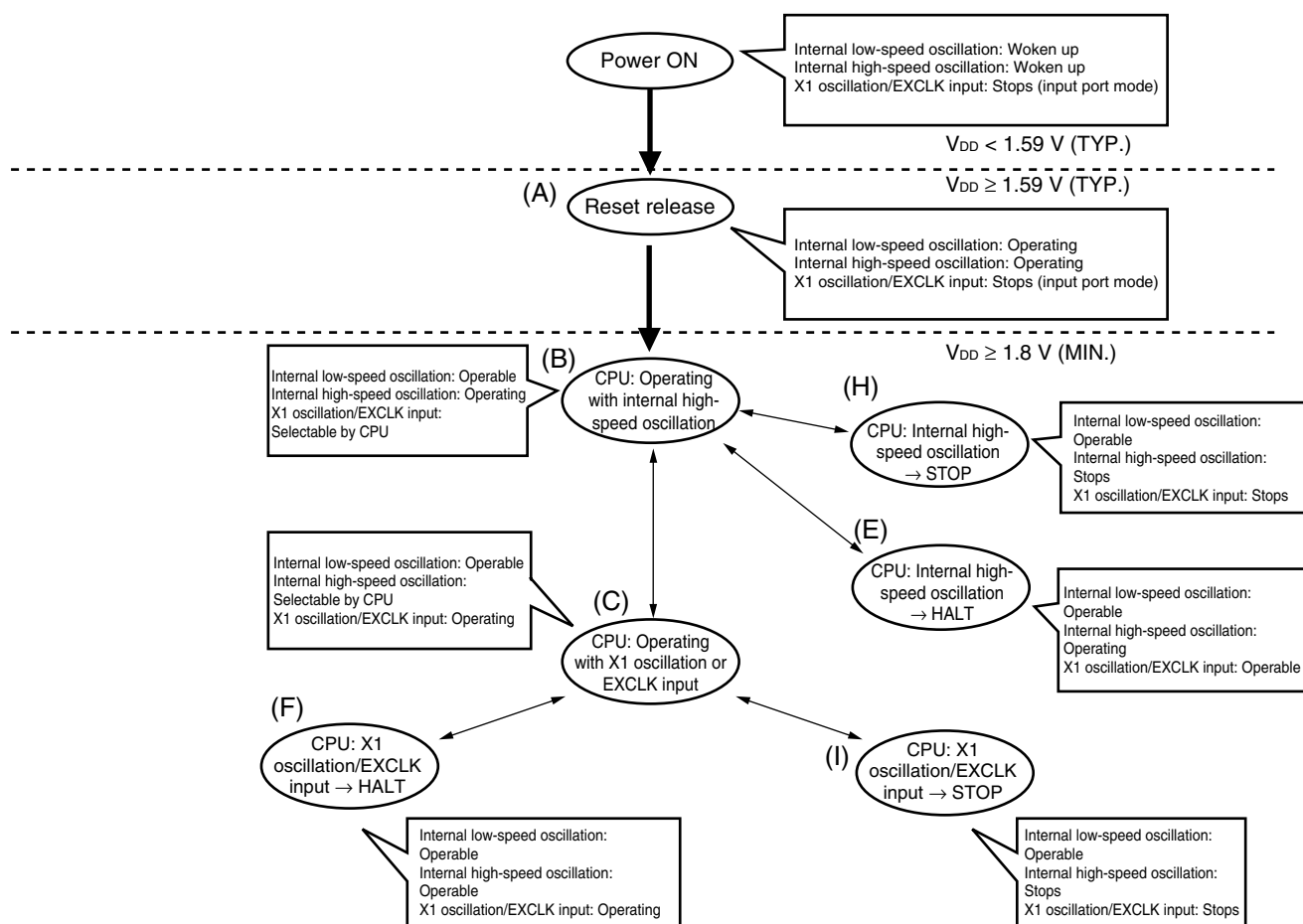
提供时钟		XSEL	MCM0	EXCLK
提供 CPU 时钟	提供外围硬件时钟			
内部高速振荡时钟		0	×	×
内部高速振荡时钟	X1 时钟	1	0	0
	外部主系统时钟	1	0	1
X1 时钟		1	1	0
外部主系统时钟		1	1	1

备注 XSEL: 主时钟模式寄存器(MCM)第 2 位
MCM0: MCM 第 0 位
EXCLK: 时钟操作模式选择寄存器(OSCCTL)第 7 位
×: 忽略

5.6.5 CPU 时钟状态转换图

本产品的 CPU 时钟状态转换图如图 5-13 所示。

图 5-13. CPU 时钟状态转换图
(已设置停止 LVI 默认启动模式功能 (选项字节:LVISTART = 0))



注意事项 允许设置 LVI 默认启动功能(选项字节:LVISTART = 1)时, 上电后, 若电源电压超过 2.7V(典型值), 则 CPU 时钟状态切换为上图所示的(A), 并在复位处理后切换到(B) (12 至 51 μs)。

CPU 时钟切换过程和 SFR 寄存器设置示例如表 5-4 所示。

表 5-4. CPU 时钟切换和 SFR 寄存器设置示例 (1/2)

(1) 复位解除(A)后,CPU 使用内部高速振荡时钟(B)。

状态转换	SFR 寄存器设置
(A) → (B)	无需设置 SFR 寄存器 (复位解除后默认状态)

(2) 复位解除(A)后,CPU 使用高速系统时钟(C)。

(复位解除后, CPU 立即执行内部高速振荡时钟(B)。)

(SFR 寄存器的设置序列) →						
(SFR 寄存器的设置标志)	EXCLK	OSCSEL	MSTOP	OSTC 寄存器	XSEL	MCM0
状态转换						
(A) → (B) → (C) (X1 时钟)	0	1	0	必须检测	1	1
(A) → (B) → (C) (外部主系统时钟)	1	1	0	不必检测	1	1

注意事项 电源电压达到时钟所需的工作电压后, 设置时钟(参阅第二十四章 电特性)。

(3) CPU 时钟从内部高速振荡时钟(B)切换到高速系统时钟(C)

(SFR 寄存器的设置序列) →						
(SFR 寄存器的设置标志)	EXCLK	OSCSEL	MSTOP	OSTC 寄存器	XSEL [※]	MCM0
状态转换						
(B) → (C) (X1 时钟)	0	1	0	必须检测	1	1
(B) → (C) (外部主系统时钟)	1	1	0	不必检测	1	1

若已设置这些寄存器,则不需要。 若 CPU 正在执行高速系统时钟, 则不需要。

注 复位解除后, 该标志值只能修改一次。若已设置该项, 则无需再设置。

注意事项 电源电压达到时钟所需的工作电压后, 设置时钟(参见第二十四章 电特性(目标))。

- 备注 1.** 表 5-4 中的(A)至(I)对应图 5-13 的(A)至(I)。
2. EXCLK, OSCSEL: 时钟操作模式选择寄存器(OSCCTL)第 7 位和第 6 位
MSTOP: 主 OSC 控制寄存器(MOC)第 7 位
XSEL, MCM0: 主时钟模式寄存器(MCM)第 2 位和第 0 位

表 5-4. CPU 时钟切换和 SFR 寄存器设置示例 (2/2)

(4) CPU 时钟从高速系统时钟(C)切换到内部高速振荡时钟(B)

(SFR 寄存器的设置序列) →			
(SFR 寄存器的设置标志)	RSTOP	RSTS	MCM0
状态转换			
(C) → (B)	0	确认该标志为“1”。	0

若 CPU 执行内部高速振荡时钟,则不需要。

(5) • CPU 正在执行内部高速振荡时钟(B)期间, 设置 HALT 模式(E)。

- CPU 正在执行高速系统时钟期间, 设置 HALT 模式(F)。

状态转换	设置
(B) → (E)	执行 HALT 指令
(C) → (F)	

(6) • CPU 正在执行内部高速振荡时钟(B)期间, 设置 STOP 模式(H)。

- CPU 正在执行高速系统时钟(C)期间, 设置 STOP 模式(I)。

(设置序列) →		
状态转换	设置	
(B) → (H)	禁止操作 STOP 模式, 停止外围功能	执行 STOP 指令
(C) → (I)		

- 备注 1. 表 5-4 中的(A)至(I)对应图 5-13 的(A)至(I)。
2. MCM0: 主时钟模式寄存器(MCM)第 0 位
RSTS, RSTOP: 内部振荡模式寄存器(RCM)第 7 位和第 0 位

5.6.6 CPU 时钟的切换前条件和切换后处理

CPU 时钟的切换前条件和切换后处理如下所示。

表 5-5.CPU 时钟切换

CPU 时钟		切换前条件	切换后处理
切换前	切换后		
内部高速振荡时钟	X1 时钟	X1 振荡稳定 • MSTOP = 0, OSCSEL = 1, EXCLK = 0 • 振荡稳定结束后	允许停止内部高速振荡器(RSTOP = 1)。
	外部主系统时钟	允许从 EXCLK 引脚的外部时钟输入 • MSTOP = 0, OSCSEL = 1, EXCLK = 1	允许停止内部高速振荡器(RSTOP = 1)。
X1 时钟	内部高速振荡时钟	内部高速振荡器振荡 • RSTOP = 0	可以停止 X1 振荡(MSTOP = 1)。
外部主系统时钟			禁止外部主系统时钟输入(MSTOP = 1)。

5.6.7 主系统时钟切换所需时间

通过设置处理器的时钟控制寄存器(PCC)的第 0 位至第 2 位(PCC0 至 PCC2)，可以切换主系统时钟的分频率。
重写 PCC 后，不能立即执行实际切换操作；将继续执行几个时钟的切换前时钟 (参阅表 5-6)。

表 5-6. 主系统时钟分频比切换所需时间

切换前设置值			切换后设置值														
PCC2	PCC1	PCC0	PCC2	PCC1	PCC0	PCC2	PCC1	PCC0	PCC2	PCC1	PCC0	PCC2	PCC1	PCC0	PCC2	PCC1	PCC0
			0	0	0	0	0	1	0	1	0	0	1	1	1	0	0
0	0	0				16 个时钟			16 个时钟			16 个时钟			16 个时钟		
0	0	1				8 个时钟			8 个时钟			8 个时钟			8 个时钟		
0	1	0	4 个时钟			4 个时钟			4 个时钟			4 个时钟			4 个时钟		
0	1	1	2 个时钟			2 个时钟			2 个时钟			2 个时钟			2 个时钟		
1	0	0	1 个时钟			1 个时钟			1 个时钟			1 个时钟			1 个时钟		

备注 表 5-6 给出的时钟数为切换前的 CPU 时钟数。

通过设置主时钟模式寄存器(MCM)的第 0 位(MCM0)，可以切换主系统时钟（在内部高速振荡时钟和高速系统时钟之间）。

重写 MCM0 后，不能立即执行实际切换操作；将继续执行几个时钟的切换前时钟 (参阅表 5-7)。

通过 MCM 的第 1 位可以确定 CPU 是否执行内部高速振荡时钟或高速系统时钟。

表 5-7. 主系统时钟切换所需的最长时间

切换前设置值	切换后设置值	
MCM0	MCM0	
	0	1
0	$1 + 2f_{IH}/f_{XH}$ 时钟	
1		

注意事项 内部高速振荡时钟切换到高速系统时钟时，MCM 的第 2 位必须预先置“1”。复位解除后，该 XSEL 值只能修改一次。

- 备注 1. 表 5-7 给出的时钟数为切换前的主系统时钟数。
2. 通过舍去小数用整数计算表 5-7 的时钟数。

示例 主系统时钟从内部高速振荡时钟切换到高速系统时钟(@振荡频率 $f_{IH} = 8\text{ MHz}$, $f_{XH} = 10\text{ MHz}$)
 $1 + 2f_{IH}/f_{XH} = 1 + 2 \times 8/10 = 1 + 2 \times 0.8 = 1 + 1.6 = 2.6 \rightarrow 2$ 个时钟

5.6.8 时钟振荡停止前的条件

以下列出停止时钟振荡的寄存器标志设置（禁止外部时钟输入）和时钟振荡停止前的条件。

表 5-8. 时钟振荡停止前的条件和标志设置

时钟	时钟振荡停止前的条件 (禁止外部时钟输入)	SFR 寄存器的设置标志
内部高速振荡时钟	MCS = 1 (CPU 执行高速系统时钟)	RSTOP = 1
X1 时钟	MCS = 0 (CPU 执行内部高速振荡时钟)	MSTOP = 1
外部主系统时钟		

5.6.9 外围硬件和源时钟

以下列出了 μPD79F7023, 79F7024 微控制器中包括的外围硬件和源时钟。

表 5-9. 外围硬件和源时钟

S 源时钟		外围硬件时钟(f _{PRS})	内部低速振荡时钟(f _{IL})	外围硬件引脚的外部时钟
外围硬件				
16 位定时器/事件计数器 00		Y	N	Y (TI000 pin) [※]
8 位定时器/事件计数器 00	51	Y	N	Y (TI51 pin) [※]
8 位定时器	H1	Y	Y	N
看门狗定时器		N	Y	N
A/D 转换器		Y	N	N
串行接口	UART0	Y	N	N

注 STOP 模式下，禁止通过外围硬件引脚的外部时钟启动外围硬件操作。

备注 Y:可选, N:不可选

第六章 16 位定时器/事件计数器 00

6.1 16 位定时器/事件计数器 00 功能

16 位定时器/事件计数器 00 具有以下功能。

(1) 间隔定时器

通过预置的时间间隔产生中断请求。

(2) 方波输出

可以输出任意频率的方波。

(3) 外部事件计数器

可以测量外部输入信号的脉冲数。

(4) 单触发脉冲输出

可以输出任意脉冲宽度的单脉冲。

(5) PPG 输出

可以输出任意频率和输出脉冲宽度的方波。

(6) 脉冲宽度测量

可以测量外部输入信号的脉冲宽度。

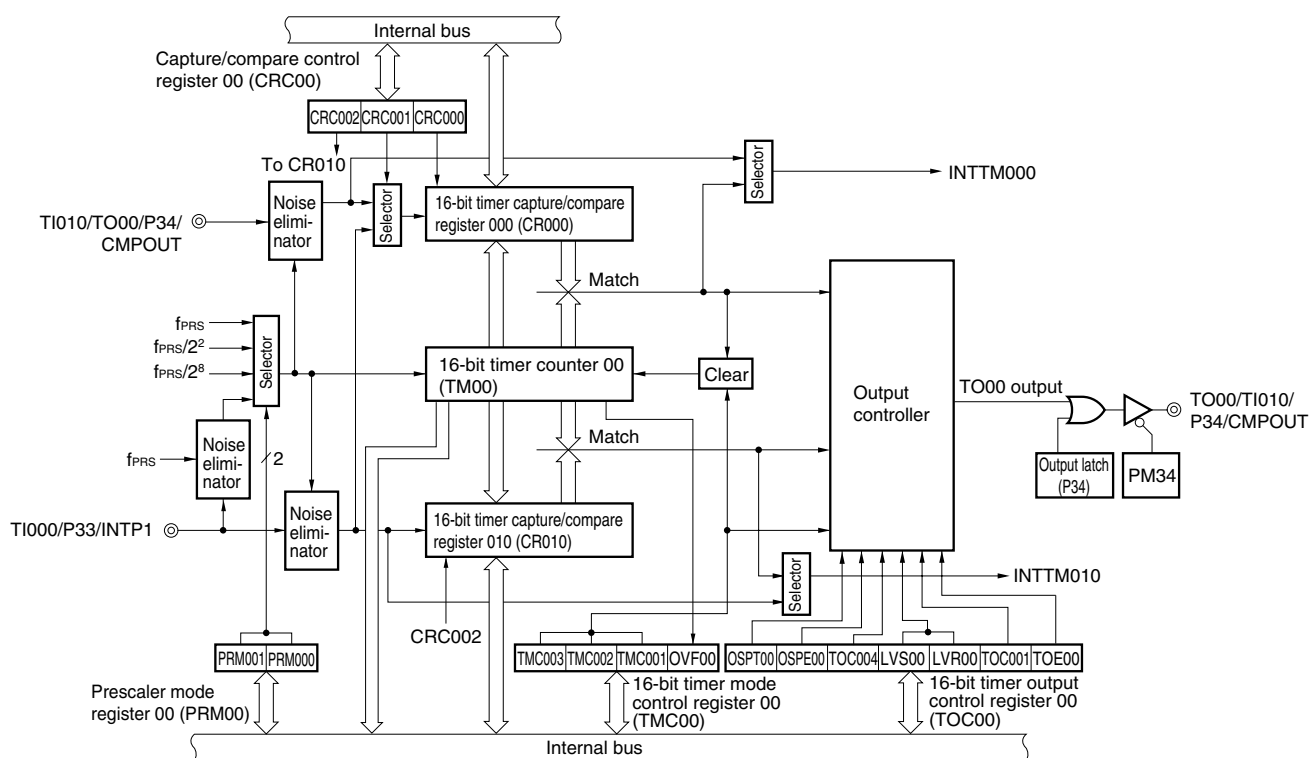
6.2 16 位定时器/事件计数器 00 配置

16 位定时器/事件计数器 00 包括以下硬件。

表 6-1. 16 位定时器/事件计数器 00 配置

项目	配置
定时器/计数器	16 位定时器/事件计数器 00 (TM00)
寄存器	16 位定时器捕捉/比较寄存器 000, 010 (CR000, CR010)
定时器输入	TI000, TI010
定时器输出	TO00, 输出控制器
控制寄存器	16 位定时器模式控制器寄存器 00 (TMC00) 16 位定时器捕捉/比较控制寄存器 00 (CRC00) 16 位定时器输出控制寄存器 00 (TOC00) 预分频器模式寄存器 00 (PRM00) 端口模式寄存器 3 (PM3) 端口寄存器 3 (P3)

图 6-1. 16 位定时器/事件计数器 00 的框图



- 注意事项**
1. P34 引脚不能同时使用 TI010 有效沿和定时器输出(TO00)。只能选择其中一项功能。
 2. 若 16 位定时器模式控制寄存器 00(TMC00)的第 3 位和第 2 位(TMC003 和 TMC002)清零(00)时段与输入捕捉触发输入相冲突, 则捕捉数据变化不定。
 3. 若把捕捉模式切换到比较模式, 则先将 TMC003 和 TMC002 位清为 00, 然后切换设置。
在器件复位前捕捉的值保存在 CR000 中。若模式已切换成比较模式, 则必须设置比较值。

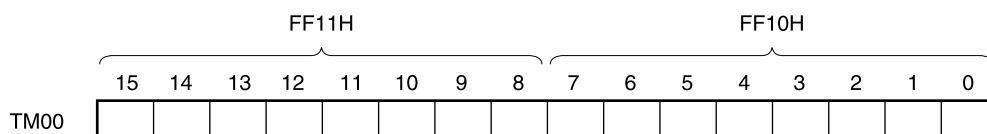
(1) 16 位定时器/计数器 00 (TM00)

TM00 是 16 位只读寄存器, 用于脉冲计数。

计数器随计数时钟的上升沿同步递增。

图 6-2. 16 位定时器/计数器 00 (TM00)格式

地址: FF10H, FF11H 复位后: 0000H R



16 位定时器模式控制寄存器 00(TMC00)的第 3 位和第 2 位没有清为 00 时, 可通过读取 TM00 来读取 TM00 的计算值。若 TMC003 和 TMC002 = 00, 则 TM00 的读取值为 0000H。

在以下情况下, 计算值为 0000H。

- 复位信号产生时
- TMC003 和 TMC002 清零(00)
- TI000 引脚输入有效沿进入清零并启动模式时, TI000 引脚输入有效沿
- TM00 和 CR00 相等进入清零并启动模式时, TM00 和 CR00 相等
- 单脉冲输出模式下, OSPT00 置“1”或 TI000 引脚输入有效沿

注意事项 即使 TM00 已被读取, 也不能通过 CR010 捕捉到该值。

(2) 16 位定时器捕捉/比较寄存器 000(CR000), 16 位定时器捕捉/比较寄存器 010(CR010)

CR000 和 CR010 是 16 位寄存器，通过 CRC00 进行捕捉功能或比较功能的选择。

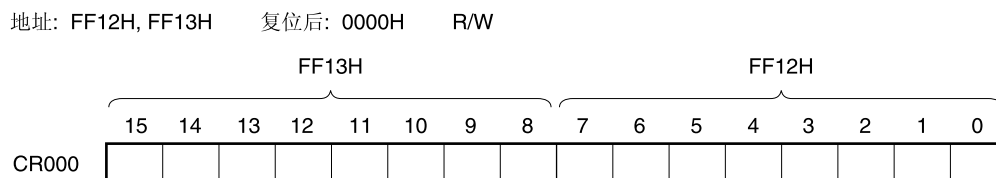
当即定时器停止(TMC003 和 TMC002 = 00)时，重写为 CR000 值。

若已按指定方式设置值，则操作期间可以重写 CR010 值。详阅 6.5.1 TM00 操作期间的 CR010 重写。

这些寄存器可以按 16 位单元读取或写入。

复位信号产生后，这些寄存器清为 00H。

图 6-3. 16 位定时器捕捉/比较寄存器 000(CR000)格式

**(i) CR000 用作比较寄存器**

CR000 的设定值不断与 TM00 计数值进行比较，如果两者匹配就会生成中断请求信号(INTTM000)。该值将一直保存到 CR000 被重写。

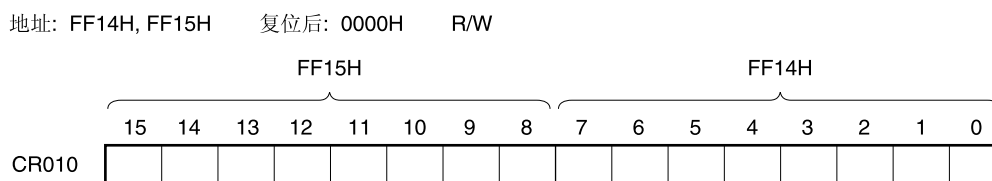
注意事项 CR000 设置为比较模式时，即使输入捕捉触发，也不执行捕捉操作。

(i) CR000 用作捕捉寄存器

输入捕捉触发时，TM00 计数值被捕捉到 CR000。

用作捕捉触发，可以通过使用 CRC00 和 PRM00 选择 TI000 引脚的反向沿或 TI010 引脚的有效沿。

图 6-4. 16 位定时器捕捉/比较寄存器 010(CR010)格式

**(i) CR010 用作比较寄存器**

CR010 的设定值不断与 TM00 计数值进行比较, 如果两者匹配就会生成中断请求信号(INTTM010)。

注意事项 CR010 设置为比较模式时, 即使输入捕捉触发, 也不执行捕捉操作。

(ii) CR010 用作捕捉寄存器

输入捕捉触发时, TM00 计数值被捕捉到 CR010。

可以用作捕捉触发, 选择 TI000 引脚的有效沿。通过 PRM00, 设置 TI000 引脚有效沿。

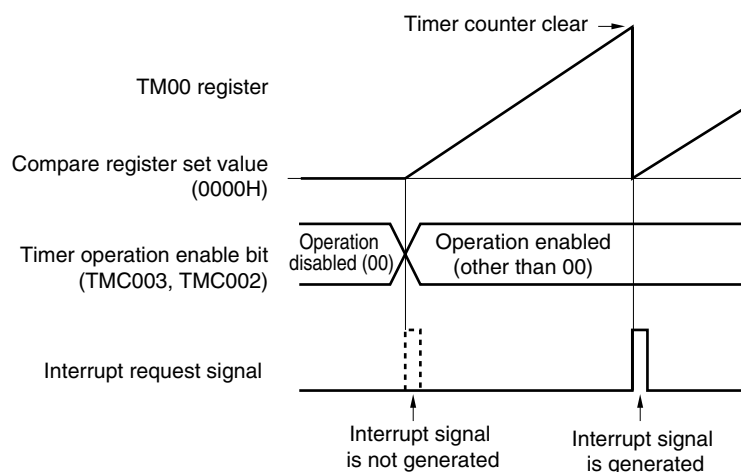
(iii) CR000 或 CR010 用作比较寄存器时, 设置范围。

CR000 或 CR010 用作比较寄存器时, 将其设置如下。

操作	CR000寄存器设置范围	CR010寄存器设置范围
用作间隔定时器	0000H < N ≤ FFFFH	0000H [※] ≤ M ≤ FFFFH
用作方波		一般不使用此设置。屏蔽匹配中断信号 (INTTM010)
用作外部事件计数器		
通过TI000引脚有效沿输入, 进入清零并启动模式	0000H [※] ≤ N ≤ FFFFH	0000H [※] ≤ M ≤ FFFFH
用作自由运行定时器		
用作PPG输出	M < N ≤ FFFFH	0000H [※] ≤ M < N
用作单脉冲输出	0000H [※] ≤ N ≤ FFFFH (N ≠ M)	0000H [※] ≤ M ≤ FFFFH (M ≠ N)

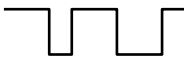
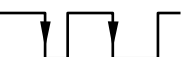
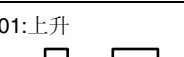
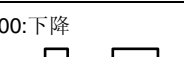
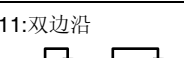
注 设置为 0000H 时, 定时器运行后不会立即产生匹配中断, 也不会改变定时器输出, 最初的匹配时序如下所示。定时器计数器(TM00)从 0000H 变化到 0001H 时, 产生匹配中断。

- 因溢出而使定时器计数器清零时。
- 因 TI000 引脚有效沿而使定时器计数器清零时(通过 TI000 引脚有效沿输入进入清零并启动模式时)。
- 因比较匹配而使定时器计数器清零(通过 TM00 和 CR000 匹配(CR000 = 0000H 除外的值, CR010 = 0000H), 进入清零并启动模式)。



- 备注
1. N:CR000 寄存器设定值， M:C0BRP 寄存器设定值
 2. 有关详细的操作允许位(TMC00 的第 3 位和第 2 位(TMC003 和 TMC002)，参阅 **6.3 (1) 16 位定时器模式控制寄存器 00(TMC00)**

表 6-2. CR000 和 CR010 的捕捉操作

外部输入 信号		TI000 引脚输入 		TI010 引脚输入 	
捕获操作	CR000捕捉操作	CRC001 = 1 TI000引脚输入 (反相) 	设置ES010和ES000 捕捉沿位置	CRC001位 = 0 TI010引脚输入 	设置ES110和ES100 捕捉沿位置
			01:上升 		01:上升 
			00:下降 		00:下降 
			11:双边沿 (不能捕捉)		11:双边沿 
		中断信号	即使捕捉到计数值，也不产生INTTM000信号。	中断信号 每当捕捉到时间值时，产生INTTM000信号。	
捕获操作	CR010捕捉操作	TI000 引脚输入 [※] 	设置ES010和ES000 捕捉沿位置		
			01:上升 		
			00:下降 		
			11:双边沿 		
		中断信号	每当捕捉到时间值时，产生INTTM010信号。		

注 CRC001 位设置不影响 CR010 的捕捉操作。

注意事项 若使用输入到 TI000 引脚的反相来捕捉 TM00 寄存器并放入 CR000 寄存器，则捕捉计数值后不产生中断请求信号(INTTM000)。若操作期间检测出 TI010 引脚的有效值，则不执行捕捉操作，但产生的 INTTM000 信号用作外部中断信号。不使用外部中断时，屏蔽 INTTM000 信号。

备注 CRC001:参阅 6.3 (2) 捕捉/比较控制寄存器 00(CRC00)。

ES110, ES100, ES010, ES000:参阅 6.3 (4) 预分频模式寄存器 00(PRM00)。

6.3 16 位定时器/事件计数器 00 的控制寄存器

16 位定时器/事件计数器 00 的控制寄存器如下所示。

- 16 位定时器模式控制寄存器 00 (TMC00)
- 捕捉/比较控制寄存器 00 (CRC00)
- 16 位定时器输出控制寄存器 00 (CR010)
- 预分频模式寄存器 00 (PRM00)
- 端口模式寄存器 3(PM3)
- 端口寄存器 3 (P3)

(1) 16 位定时器模式控制寄存器 00 (TMC00)

TMC00 是 8 位寄存器，用于设置 16 位定时器/事件计数器 00 操作模式、TM00 清零模式、输出时序和溢出检测。

操作期间禁止重写 TMC（TMC003 和 TMC002 = 00 除外的值）。但若 TMC003 和 TMC002 清为 00（停止操作）以及 OVF00 清零，则可以重写。

可按 1 位或 8 位存储器操作指令设置 TMC00。

复位信号产生时，TMC00 清为 00H。

注意事项 TMC003 和 TMC00216 设置为 00(操作停止模式)之外的值时，16 位定时器/事件计数器开始操作。通过 TMC003 和 TMC002 清为 00，停止操作。

图 6-5. 16 位定时器模式控制寄存器 00(TMC00)格式

地址: FF86H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	<0>
TMC00	0	0	0	0	TMC003	TMC002	TMC001	OVF00

TMC003	TMC002	允许操作16位定时器/事件计数器00
0	0	禁止操作16位定时器/事件计数器00停止供应操作时钟16位定时器/计数器清为00(TM00)
0	1	自由运行定时器模式
1	0	通过TI000引脚有效沿输入 ^注 , 进入清零并启动模式
1	1	TM00与CR000相匹配时, 进入清零并启动模式。

TMC001	反向定时器输入(TO00)条件
0	• TM00与CR000相匹配或TM00与CR010相匹配
1	• TM00与CR000相匹配或TM00与CR010相匹配 • TI000引脚有效沿的触发输入

OVF00	TM00溢出标志
清除 (0)	OVF00清零或TMC003和TMC002 = 00
设置 (1)	产生溢出
所有操作模式下 (自由运行定时器模式、通过TI000引脚有效沿输入进入的清零并启动模式以及TM00与CR000相匹配进入的清零并启动模式), TM00由FFFFH变为0000H时, OVF00置“1”。 也可以将“1”写入OVF00, 置“1”。	

注 通过预分频器模式寄存器 00(PRM00)的第 5 和 4 位(ES001 和 ES000)设置 TI000 引脚的有效沿。

(2) 捕捉/比较控制寄存器 00 (CRC00)

CRC00 用来控制 CR000 与 CR010 的操作。

操作期间, 禁止重写 CRC00 (TMC003 和 TMC002 = 00 除外的值)

可以由 1 位或 8 位存储器操作指令设置 CRC00。

复位信号产生时, CRC00 清为 00H。

图 6-6. 捕捉/比较控制寄存器 00(CRC00)格式

地址: FF88H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
CRC00	0	0	0	0	0	CRC002	CRC001	CRC000

CRC002	CR010 操作模式选择
0	用作比较寄存器
1	用作捕捉寄存器

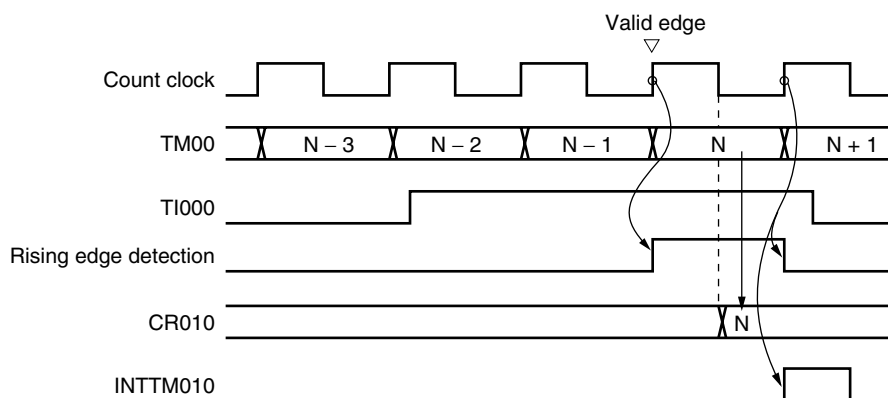
CRC001	CR000 捕捉触发选择
0	TI010 引脚有效沿捕捉
1	TI000 引脚有效沿反相捕捉 [※]
通过 PRM00 设置 TI010 和 TI000 引脚的有效沿。 CRC001 置“1”时, 若 ES010 和 EX000 置“11”, 则检测不出 TI000 有效沿。	

CRC000	CR000 操作模式选择
0	用作比较寄存器
1	用作捕捉寄存器
若 TMC003 和 TMC002 置“11”(TM00 和 CR000 相匹配进入清零并启动模式), 则应确保 CRC000 清零。	

注 若检测出 TI010 引脚的有效值, 则不执行捕捉操作, 但产生的 INTTM000 信号用作外部中断信号。

注意事项 为了正确执行捕捉操作,应确保捕捉触发所需的脉冲宽度应大于预分频器模式寄存器 00(PRM00)选择的计数时钟的两个周期。

图 6-7. CR010 捕捉操作示例 (指定上升沿)



(3) 16 位定时器输出控制寄存器 00 (TOC00)

TAB1IOC4 是 8 位寄存器，用于控制 TO00 输出。

只有 OSPT00 操作时，才可重写 TOC00(TMC003 和 TMC002= 00 意外值)。操作期间，禁止重写其他位。

但是，使用定时器操作重写 CR010 时，可以重写 TOC004(参阅 6.5.1 TM00 操作期间 CR010 的重写)。

可按 1 位或 8 位存储器操作指令设置 TOC00。

复位信号产生时，TOC00 清为 00H。

注意事项 必须按照以下步骤设置 TOC00。

<1> TOC004 和 TOC001 置“1”。

<2> 仅将 TOE00 置“1”。

<3> LVS00 或 LVR00 置“1”。

图 6-8. 16 位定时器输出控制寄存器 00(TOC00)格式

地址: FF89H 复位后: 00H R/W

符号	7	<6>	<5>	4	<3>	<2>	1	<0>
TOC00	0	OSPT00	OSPE00	TOC004	LVS00	LVR00	TOC001	TOE00

OSPT00	通过软件进行单脉冲输出触发
0	—
1	单发脉冲输出
该位读取值始终为“1”。单脉冲数触模式除外的模式下，禁止将该位置“1”。 若该位置“1”，则 TM00 清零并启动。	

OSPE00	单脉冲输出操作控制
0	连续脉冲输出
1	单发脉冲输出
在自由运行定时器模式或通过 TI000 引脚有效沿输入进入的清零并启动模式下，可以正确输入单脉冲。 在 TM00 与 CR000 相匹配进入的清零并启动模式下，不能输入单脉冲。	

TOC004	CR010 和 TM00 相匹配时 TO00 输出控制
0	禁止反转操作
1	允许反转操作
即使 TOC004 = 0，也产生中断信号(INTTM010)。	

LVS00	LVR00	TO00 输出状态设置
0	0	无变化
0	1	TO00 输出初始值为低电平(TO00 输出清零)。
1	0	TO00 输出初始值为高电平(TO00 输出置“1”)。
1	1	禁止设置
- LVS00 与 LVR00 可以用来设置 TO00 输出电平的初始值。若不设置初始值，则 LVS00 和 LVR00 保持为 00。 - TOE00 = 1 时，必须设置 LVS00 和 LVR00。 禁止同时将 LVS00、LVR00 和 TOE00 置“1”。 - LVS00 和 LVR00 是触发位。通过将这俩位置“1”，可以设置 TO00 输出电平的初始值。即使这两位清零，也不会影响 TO00 输出。 - LVS00 和 LVR00 的读取值始终为“0”。 - 有关 LVS00 和 LVR00 的设置，参阅 6.5.2 LVS00 和 LVR00 的设置。 - 除 TO00 输出之外，实际的 TO00/TI010/P01 引脚输出取决于 PM01 和 P01。		

TOC001	CR000 和 TM00 相匹配时的 TO00 输出控制
0	禁止反转操作
1	允许反转操作
即使 TOC001 = 0，也产生中断信号(INTTM000)。	

TOE00	TO00 输出控制
0	禁止输出 (TO00 输出恒为低电平)
1	允许输出

(4) 预分频模式寄存器 00(PRM00)

PRM00 寄存器用来设置 TM00 计数时钟以及 TI000 和 TI010 引脚输入的有效沿。

操作期间禁止重写 PRM00 (TMC003 和 TMC002 = 00 除外的值)

可按 1 位或 8 位存储器操作指令设置 PRM00。

复位信号产生后, PRM00 清为 00H。

注意事项 1. PRM001 和 PRM000 置“11” (指定 TI000 引脚的有效沿用作计数时钟)时, 禁止进行以下设置。

- 通过 TI000 引脚有效沿进入清零并启动模式。
 - 设置 TI000 引脚为捕捉触发。
2. 若 TI000 或 TI010 引脚处于高电平,并且上升沿或双边沿指定为 TI000 或 TI010 引脚的有效沿时, 允许 16 位定时器/事件计数器 00 的操作, 则检测出的 TI000 或 TI010 的高电平为上升沿。上拉 TI000 和 TI010 引脚时要特别注意。但若一旦停止定时器操作后允许重新启动, 则检测不出上升沿。
3. P34 引脚不能同时使用 TI010 有效沿和定时器输出(TO00)。只能选择其中一项功能。

图 6-9. 预分频模式寄存器 00(PRM00)格式

地址: FF87H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
PRM00	ES110	ES100	ES010	ES000	0	0	PRM001	PRM000

ES110	ES100	TI010 引脚有效沿选择
0	0	下降沿
0	1	上升沿
1	0	禁止设置
1	1	上升和下降双边沿

ES010	ES000	TI000 引脚有效沿选择
0	0	下降沿
0	1	上升沿
1	0	禁止设置
1	1	上升和下降双边沿

PRM001	PRM000	计数时钟选择		
			f _{PRS} = 2 MHz	f _{PRS} = 5 MHz
0	0	f _{PRS}	2 MHz	5 MHz
0	1	f _{PRS} /2 ²	500 kHz	1.25 MHz
1	0	f _{PRS} /2 ⁶	7.81 kHz	19.53 kHz
1	1	TI000 有效沿 ^{注1,2}		

注 1. TI000 端口的外部时钟所需的脉冲宽度应大于外围硬件时钟(f_{PRS})的两个周期。

2. STOP 模式下, 禁止通过 TI000 引脚的外部时钟启动定时器操作。

备注 f_{PRS}:外部硬件时钟频率

(5) 端口模式寄存器 3(PM3)

该寄存器按 1 位设置端口 3 的输入和输出。

P34/TO00/TI010/CMPOUT引脚用作定时器输出时，设置PM34 和P34 的输出锁存清零。

P33/TI000/INTP1 和P34/TI010/TO00/CMPOUT引脚用作定时器输入时，PM33 和PM34 置“1”。此时，P33 和P34 的输出锁存即可清零也可置“1”。

可以由 1 位或 8 位存储器操作指令设置 PM3。

复位信号产生时，PM3 设置为 FFH。

图 6-10. 端口模式寄存器 3(PM3)格式

地址: FF23H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PM0	1	1	1	PM34	PM33	PM32	PM31	PM30

PM3n	P3n 引脚 I/O 模式选择 (n = 0 至 4)
0	输出模式 (输出缓冲打开)
1	输入模式 (输出缓冲关闭)

6.4 16 位定时器/事件计数器 00 操作

6.4.1 间隔定时器操作

若 16 位定时器模式控制寄存器(TMC00) 的第 3 位和第 2 位(TMC003 和 TMC002)置“1”(TM00 和 CR000 相匹配进入清零并启动模式)，则将与计数时钟同步启动计数操作。

随后，TM00 值和 CR00 值相匹配时，TM00 清为 0000H，并产生匹配中断信号 INTTM000)。INTTM000 信号允许 TM00 用作间隔定时器进行操作。

- 备注**
1. 有关 I/O 引脚的设置，参阅 6.3 (5) 端口模式寄存器 3(PM3)
 2. 有关如何允许 INTTM000 中断，参阅第十四章 中断功能

图 6-11. 间隔定时器系统图

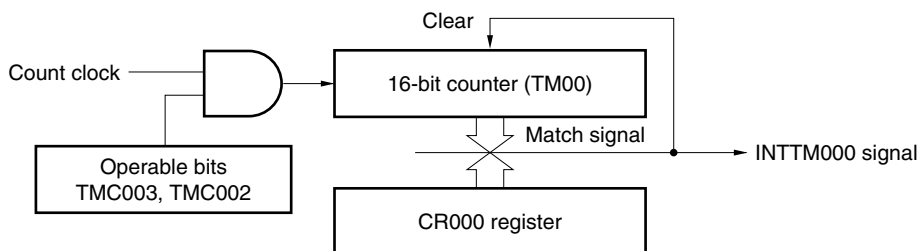


图 6-12. 间隔定时器操作的基本时序示例

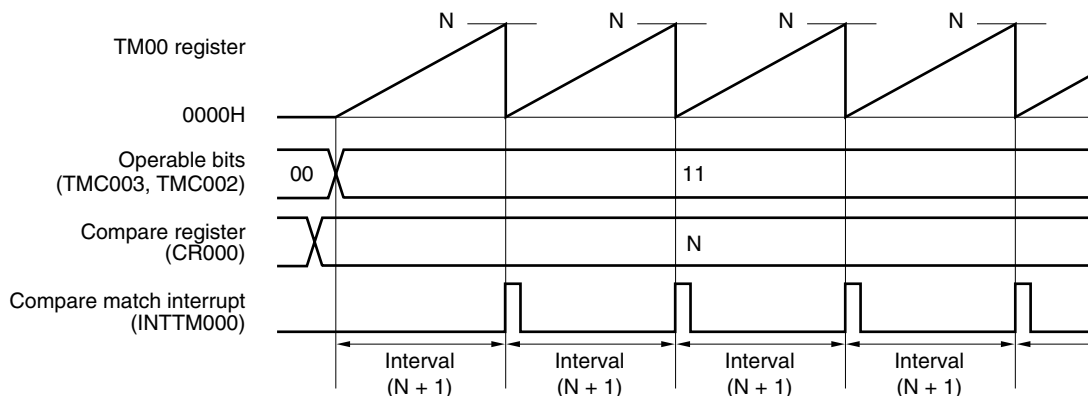
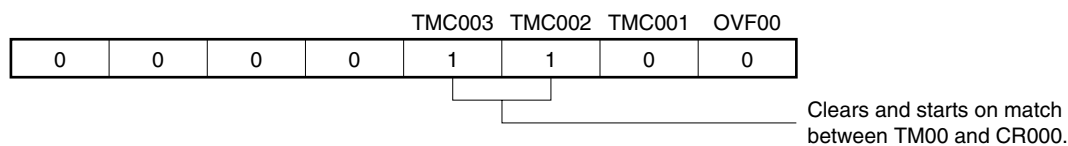
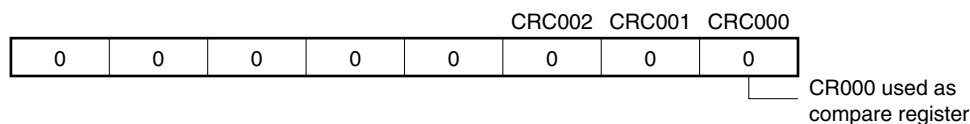


图 6-13. 间隔定时器操作的寄存器设置示例

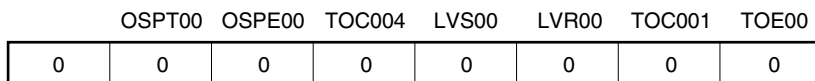
(a) 16 位定时器模式控制寄存器 00(TMC00)



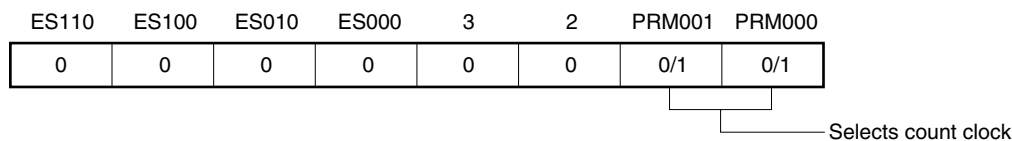
(b) 捕捉/比较控制寄存器 00(CRC00)



(c) 16 位定时器输出控制寄存器 00(TOC00)



(d) 预分频模式寄存器 00(PRM00)



(e) 16 位定时器/计数器 00(TM00)

通过读取 TM00，获取计数值。

(f) 16 位定时器捕捉/比较寄存器 000(CR000)

若 M 设置为 CR000，则间隔时间如下：

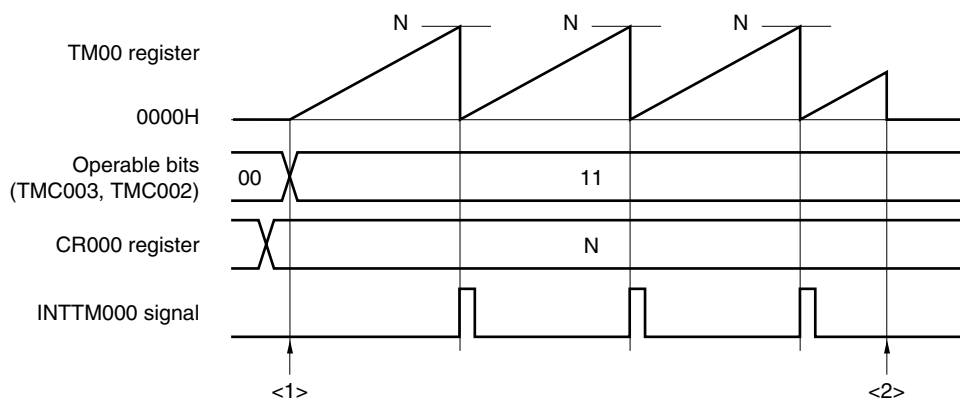
- 间隔时间 = $(M + 1) \times$ 计数时钟周期

禁止将 CR000 设置为 0000H。

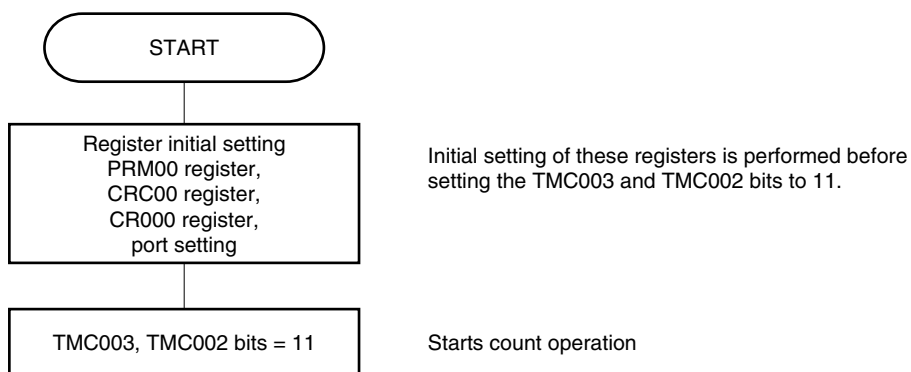
(g) 16 位捕捉/比较寄存器 010(CR010)

CR010 一般不用作间隔定时器。但若 CR010 设定值与 TM00 值相匹配，则会产生比较匹配中断(INTTM010)。因此，通过使用中断屏蔽标志屏蔽标志(TMMK010)屏蔽中断请求。

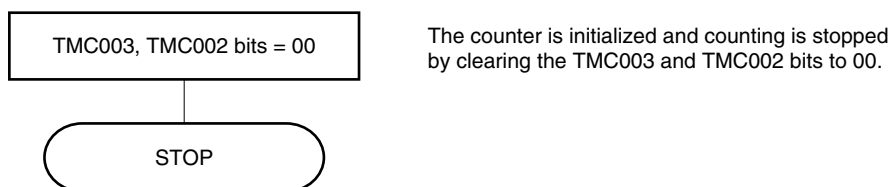
图 6-14. 间隔定时器功能的软件处理示例



<1> Count operation start flow



<2> Count operation stop flow



6.4.2 方波输出操作

16 位定时器/事件计数器 00 用作间隔定时器（参阅 6.4.1）时，通过将 16 位定时器输出控制寄存器 00(TOC00)设置为 03H，可以从 TO00 引脚输出方波。

若 TMC003 和 TMC002 置“11” (TM00 和 CR000 相匹配进入清零并启动模式)，则将与计数时钟同步启动计数操作。

随后，TM00 值和 CR00 值相匹配时，TM00 清为“0000H”，并产生匹配中断信号 INTTM000)。反转的 TO00 输出在恒定间隔允许 TO0n 输出方波。

- 备注 1. 有关 I/O 引脚的设置，参阅 6.3 (5) 端口模式寄存器 3(PM3)
2. 有关如何允许 INTTM000 信号中断，参阅第十四章 中断功能

图 6-15. 方波输出操作系统图

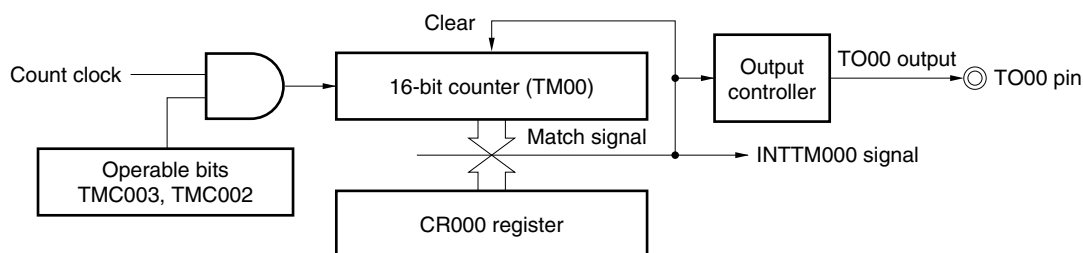


图 6-16. 方波输出操作的基本时序示例

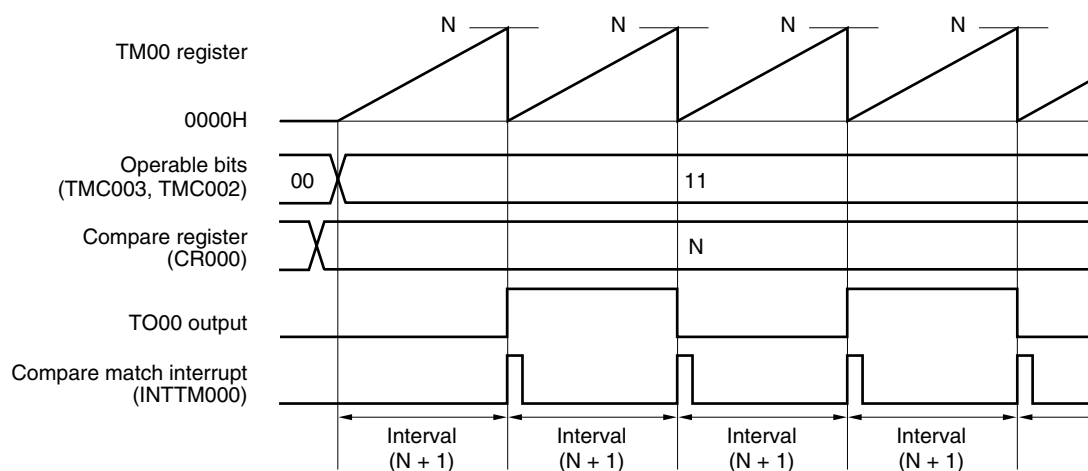
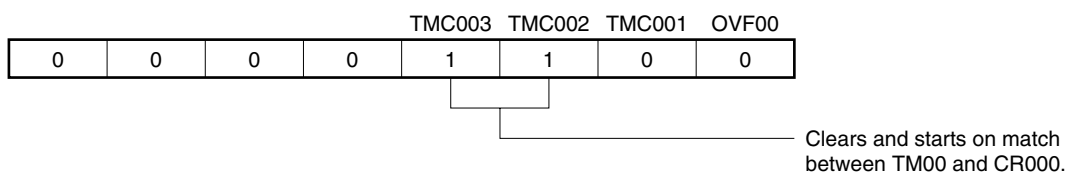
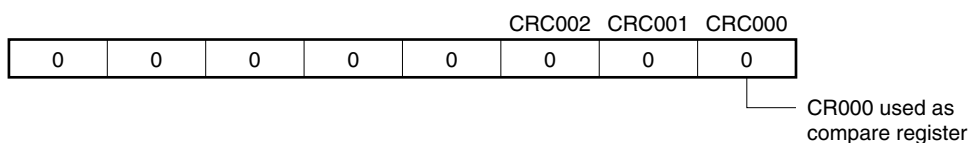


图 6-17. 方波输出操作的寄存器设置示例

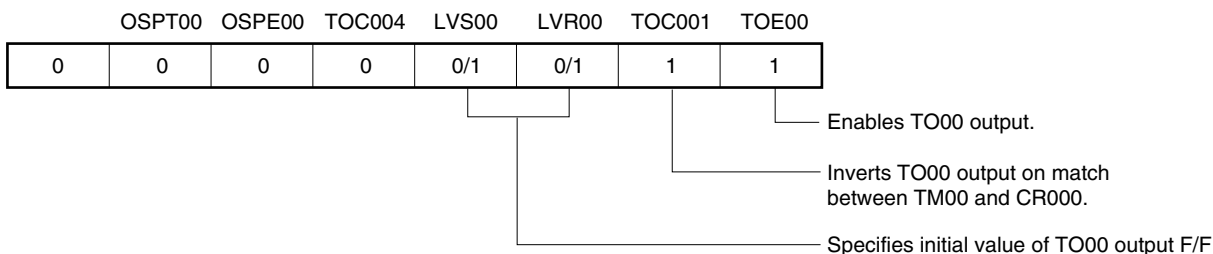
(a) 16 位定时器模式控制寄存器 00(TMC00)



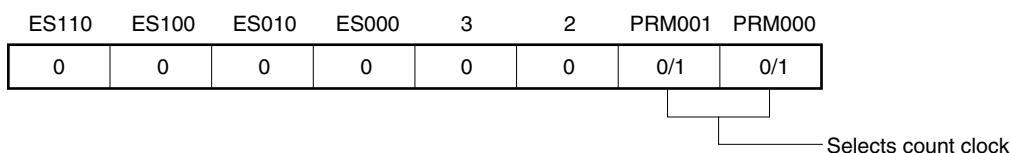
(b) 捕捉/比较控制寄存器 00(CRC00)



(c) 16 位定时器输出控制寄存器 00(TOC00)



(d) 预分频模式寄存器 00(PRM00)



(e) 16 位定时器/计数器 00(TM00)

通过读取 TM00，获取计数值。

(f) 16 位定时器捕捉/比较寄存器 000(CR000)

若 M 设置为 CR000，则间隔时间如下：

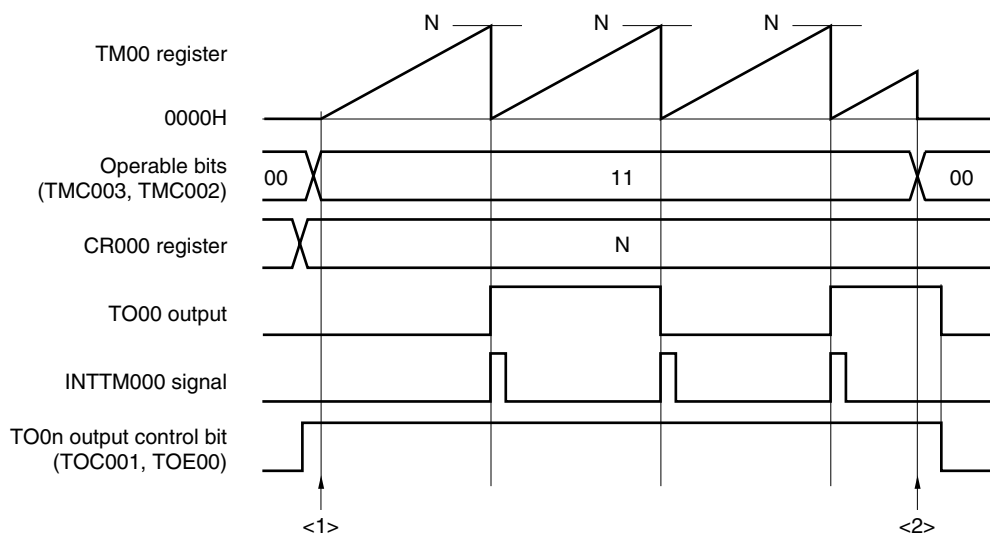
$$\bullet \text{ 方波频率} = 1 / [2 \times (M + 1) \times \text{计数时钟周期}]$$

禁止将 CR000 设置为 0000H。

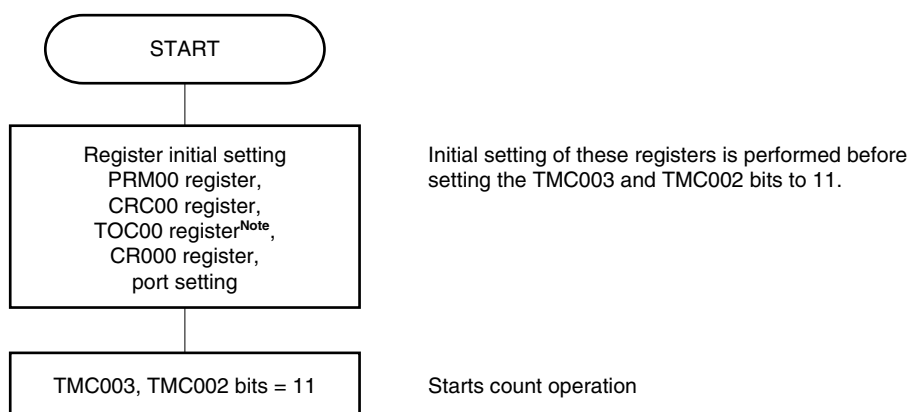
(g) 16 位捕捉/比较寄存器 010(CR010)

CR010 一般不用作方波输出。但是，CR010 设定值与 TM00 值相匹配时，会产生比较匹配中断(INTTM010)。因此，通过使用中断屏蔽标志屏蔽标志(TMMK010)屏蔽中断请求。

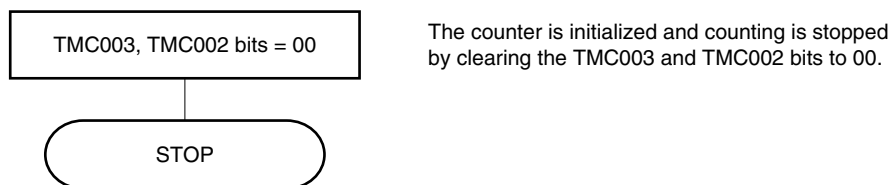
图 6-18. 方波输出功能的软件处理示例



<1> Count operation start flow



<2> Count operation stop flow



注 设置 TOC00 时，应特别注意。参阅 6.3 (3) 16 位定时器输出控制寄存器 00(TOC00)。

6.4.3 外部事件计数器操作

预分频器 00(PRM00)的第 1 位和第 0 位(PRM001 和 PRM000)置“11”(使用 TI000 引脚的有效沿计数), 16 位定时器模式控制寄存器 00(TMC00)的第 3 位和第 2 位(TMC003 和 TMC002)置“11”时, 开始对外部事件输入的有效沿进行计数, 并产生与 TM00 和 CR000(INTTM000)相匹配的指定中断信号。

使用 TI00 引脚输入外部事件。因此, 通过 TI00 引脚有效沿输入(TMC003 和 TMC002 = 10 时)进入清零并启动模式下, 定时器/事件计数器不能用作外部事件计数器。

按以下时序产生 INTTM000 信号。

- INTTM000 信号产生时序(第二次或之后)
= 外部事件输入的有效沿检测次数 × (CR000 设定值 + 1)

但是, 定时器/事件计数器开始运行后, 按以下时序立即产生第 1 次中断。

- INTTM000 信号产生时序(仅在第 1 次)
= 外部事件输入的有效沿检测次数 × (CR000 设定值 + 2)

若要检测有效沿, 应在 f_{PRS} 时钟周期内对 TI000 引脚输入信号进行采样。通过连续两次检测, 才能检测出有效沿。因此, 可以消除窄脉冲宽度的噪音。

- 备注 1.** 有关 I/O 引脚的设置, 参阅 **6.3 (5) 端口模式寄存器 3(PM3)**
- 2.** 有关如何允许 INTTM000 信号中断, 参阅**第十四章 中断功能**

图 6-19. 外部事件计数器操作系统图

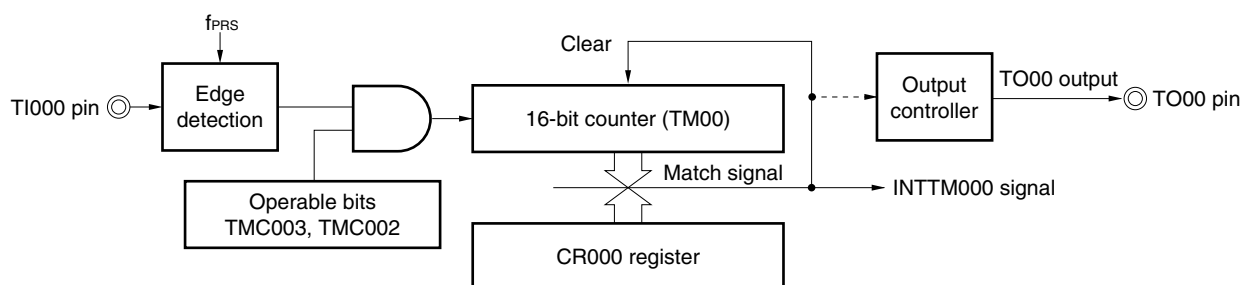
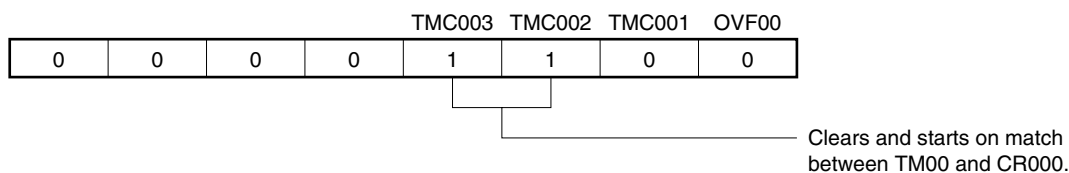
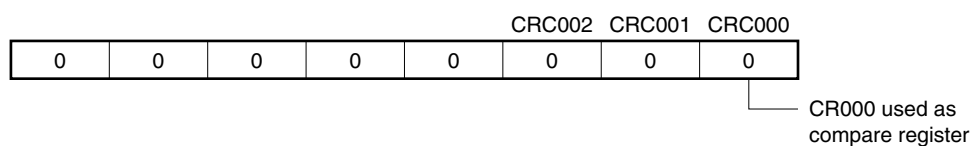


图 6-20. 外部事件计数模式下的寄存器设置示例 (1/2)

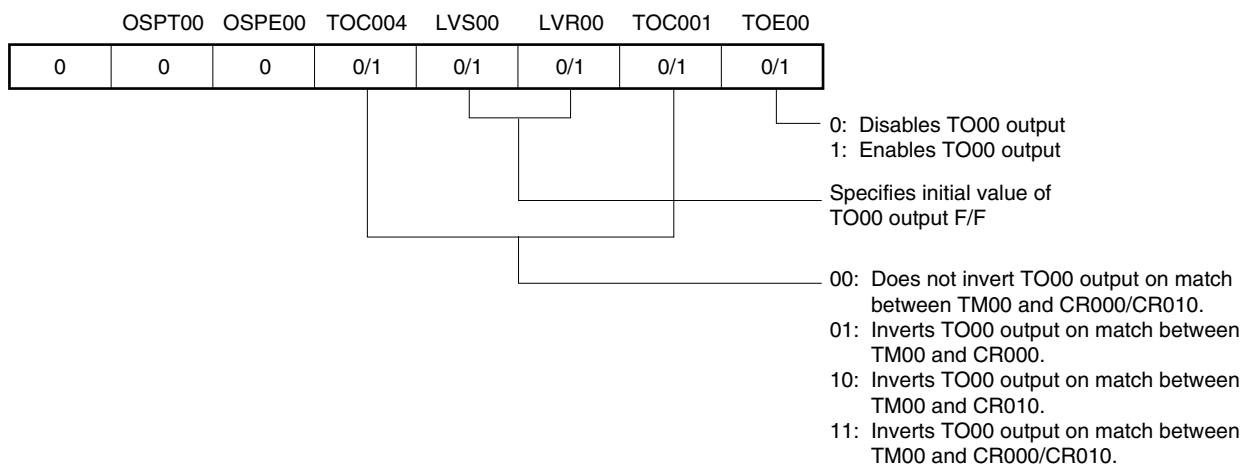
(a) 16 位定时器模式控制寄存器 00(TMC00)



(b) 捕捉/比较控制寄存器 00(CRC00)



(c) 16 位定时器输出控制寄存器 00(TOC00)



(d) 预分频模式寄存器 00(PRM00)

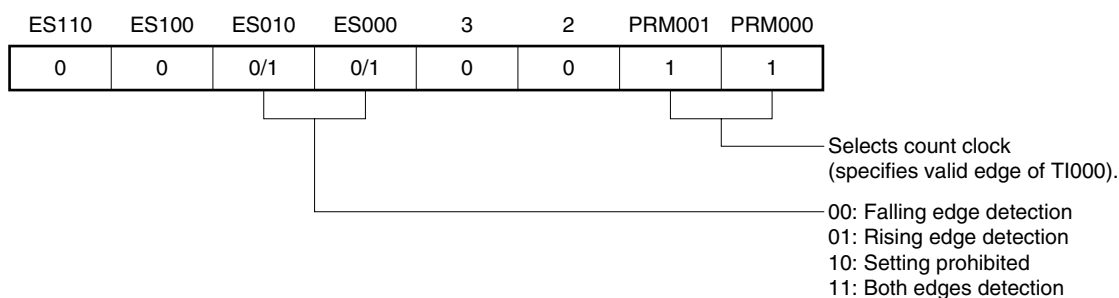


图 6-20. 外部事件计数模式下的寄存器设置示例 (2/2)

(e) 16 位定时器/计数器 00(TM00)

通过读取 TM00，获取计数值。

(f) 16 位定时器捕捉/比较寄存器 000(CR000)

若在 CR000 设置为 M，则达到外部事件数(M + 1)时，产生中断信号(INTTM000)。

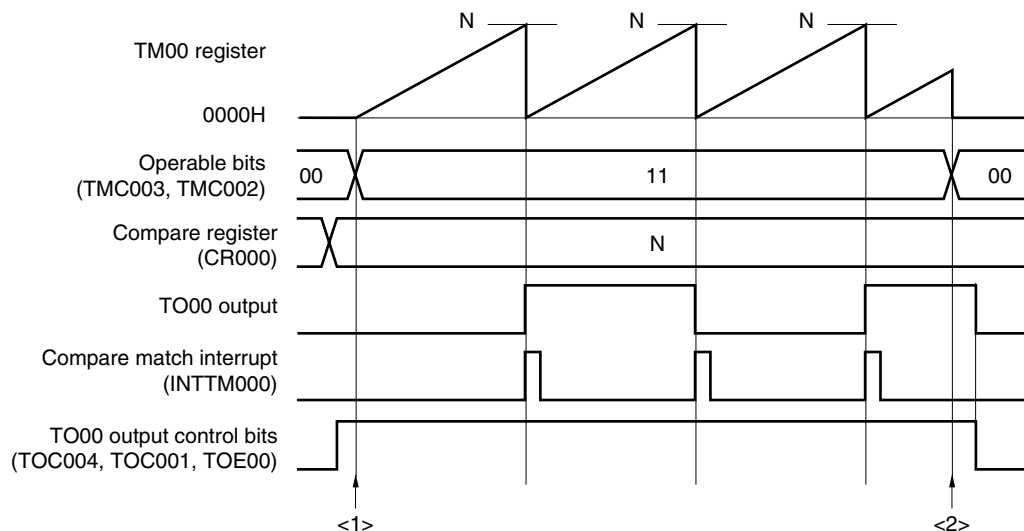
禁止将 CR000 设置为 0000H。

(g) 16 位捕捉/比较寄存器 010(CR010)

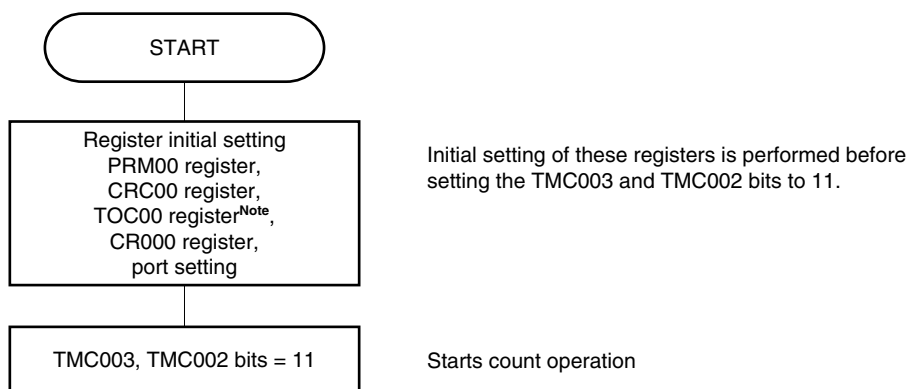
外部事件计数模式中，一般不使用 CR010。但若 CR010 设定值与 TM00 值相匹配，则会产生比较匹配中断(INTTM010)。

因此，通过使用中断屏蔽标志屏蔽标志(TMMK010)屏蔽中断请求。

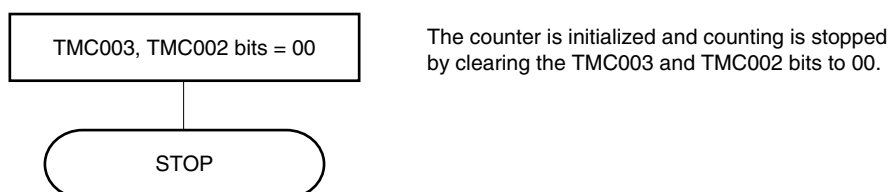
图 6-21. 外部事件计数模式下的软件处理示例



<1> Count operation start flow



<2> Count operation stop flow



注 设置 TOC00 时, 应特别注意。详情参阅 6.3 (3) 16 位定时器输出控制寄存器 00(TOC00)。

6.4.4 通过 TI000 引脚有效沿输入进入清零并启动模式

16 位定时器模式控制寄存器 00(TMC00)的第 3 位和第 2 位 (TMC003 和 TMC002)置“10”(通过 TI000 引脚有效沿输入进入清零并启动模式), 并向定时器/事件计数器提供计数时钟(由 PRM00 设置)时, TM00 开始计数。技术操作期间检测出 TI000 引脚的有效沿时, TM00 清为 0000H 并重新开始计数。若没有检测出 TI000 引脚的有效沿, 则 TM00 溢出并继续计数。

TI000 引脚的有效沿将 TM00 清零。操作开始后, 不能立即启动计数器。

CR000 和 CR010 可用作比较寄存器和捕捉寄存器。

(a) CR000 和 CR010 用作比较寄存器

TM00 的值与 CR000 和 CR010 的值相匹配时, 产生 INTTM000 和 INTTM010 的信号。

(b) CR000 和 CR010 用作捕捉寄存器

有效沿输入到 TI010 引脚 (或反向有效沿输入到 TI000 引脚) 时, TM00 的计数值被捕捉到 CR000, 并产生 INTTM000 信号。

有效沿输入到 TI000 引脚时, TM00 的计数值被捕捉到 CR010, 并产生 INTTM010 信号。一旦捕捉到计数值, 计数器清为 0000H。

注意事项 禁止将计数时钟用作 TI000 引脚的有效沿(PRM001 和 PRM000 = 11)。PRM001 和 PRM000 = 11 时, TM00 清零。

备注 1. 有关 I/O 引脚的设置, 参阅 6.3 (5) 端口模式寄存器 3(PM3)

2. 有关如何允许 INTTM000 信号中断, 参阅第十四章 中断功能

(1) 通过输入 TI000 引脚的有效沿, 进入清零并启动模式

(CR000: 比较寄存器, CR010: 比较寄存器)

图 6-22. 通过输入 TI000 引脚有效沿, 进入清零并启动模式的框图
(CR000: 比较寄存器, CR010: 比较寄存器)

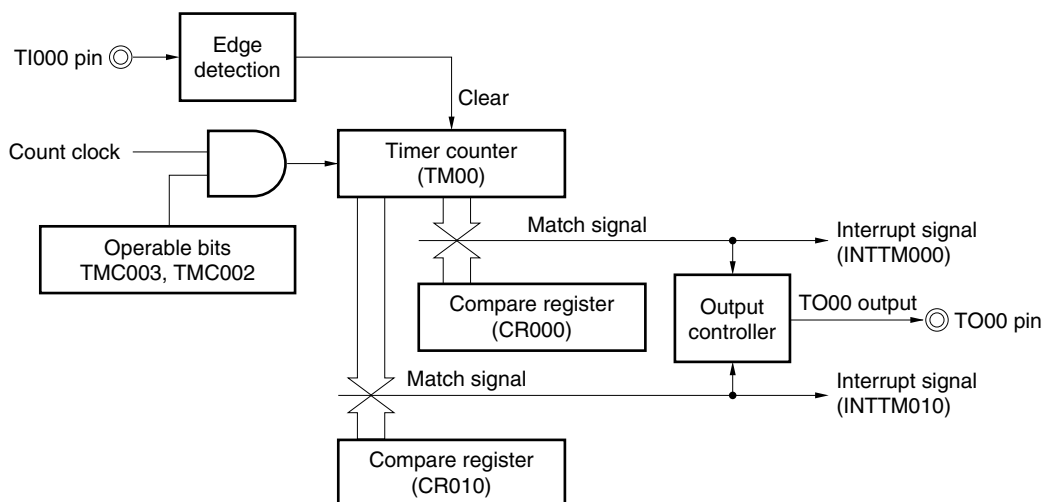
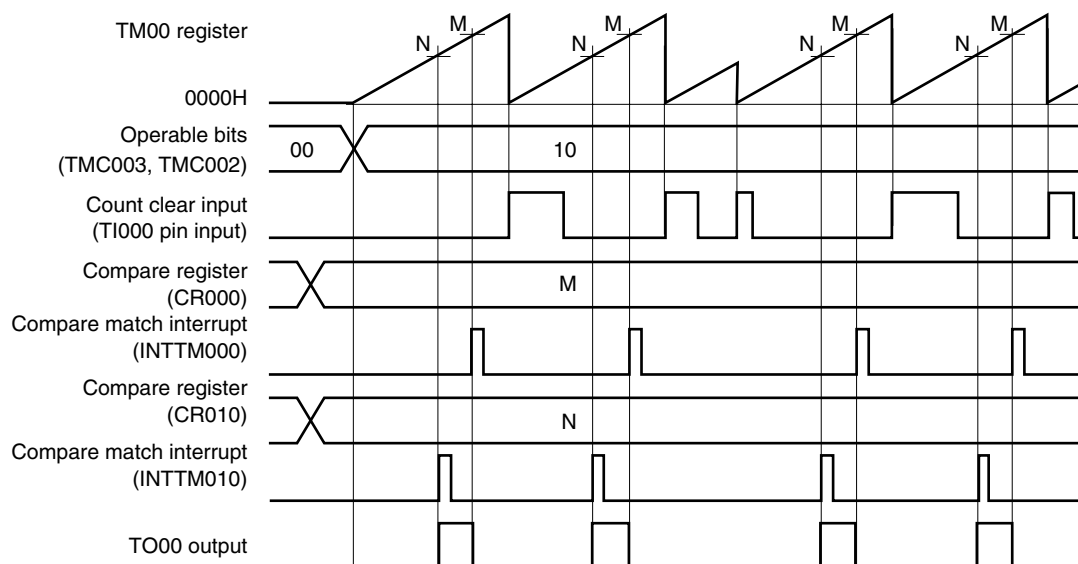
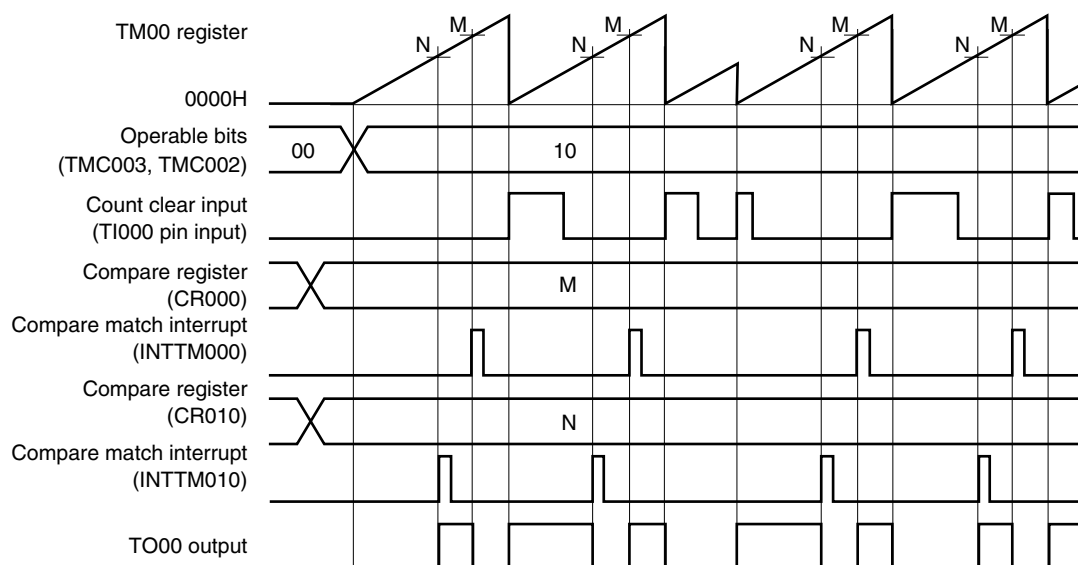


图 6-23. 通过输入 TI000 引脚的有效沿, 进入清零并启动模式的时序示例
(CR000: 比较寄存器, CR010: 比较寄存器)

(a) TOC00 = 13H, PRM00 = 10H, CRC00 = 00H, TMC00 = 08H



(b) TOC00 = 13H, PRM00 = 10H, CRC00 = 00H, TMC00 = 0AH



根据 16 位定时器模式控制器 00 (TMC00) 第 1 位的设置不同, (a)和(b)有如下变化。

- (a) TM00 与比较寄存器相匹配时 4, 反转 TO00 输出电平。
- (b) TM00 与比较寄存器相匹配或检测出 TI000 引脚的有效沿时, 反转 TO00 输出电平。

- (2) 通过输入 TI000 引脚的有效沿，进入清零并启动模式
(CR000: 比较寄存器, CR010: 捕捉寄存器)

图 6-24. 通过输入 TI000 引脚有效沿，进入清零并启动模式的框图
(CR000:比较寄存器, CR010:捕捉寄存器)

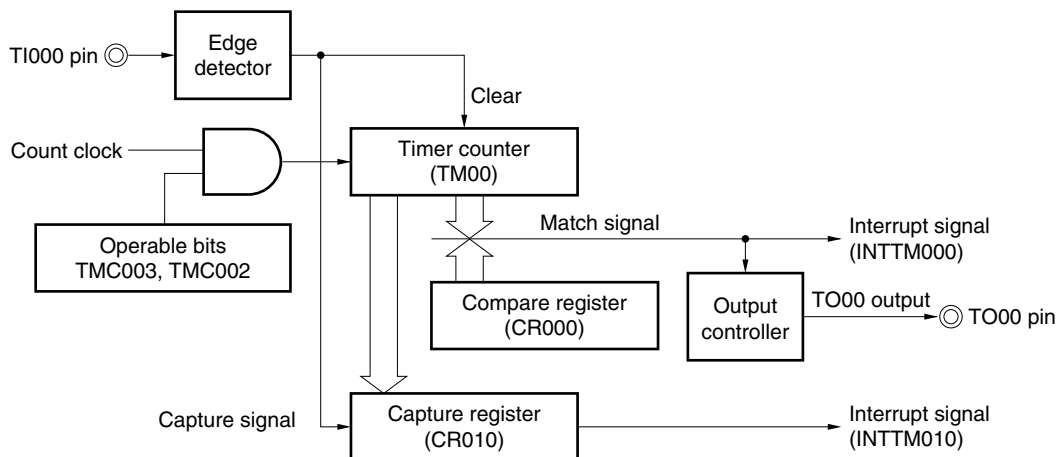
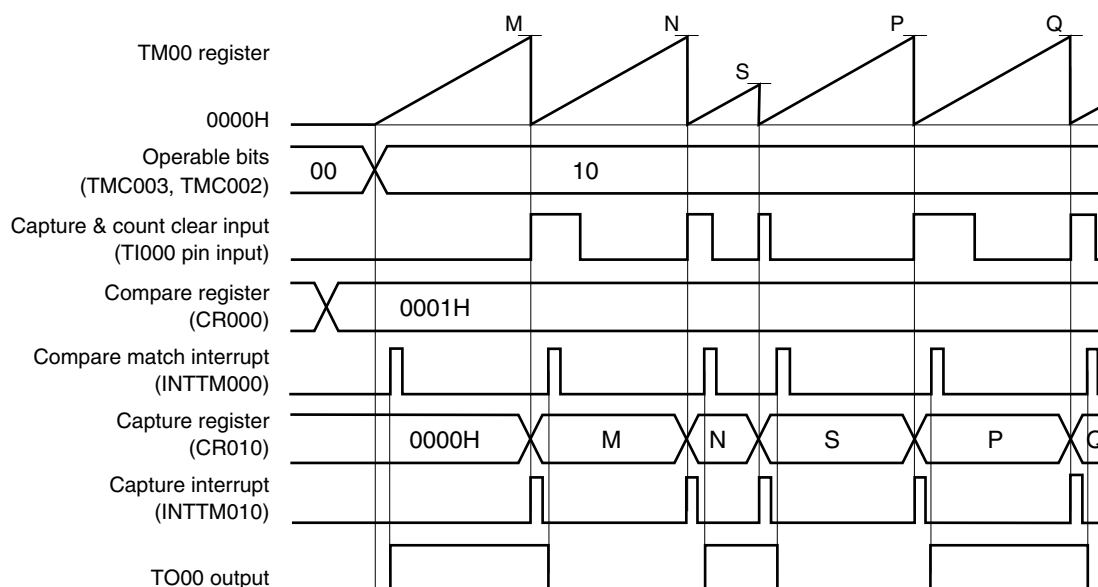


图 6-25. 通过输入 TI000 引脚的有效沿，进入清零并启动模式的时序示例
(CR000:比较寄存器, CR010:捕捉寄存器) (1/2)

(a) TOC00 = 13H, PRM00 = 10H, CRC00 = 04H, TMC00 = 08H, CR000 = 0001H

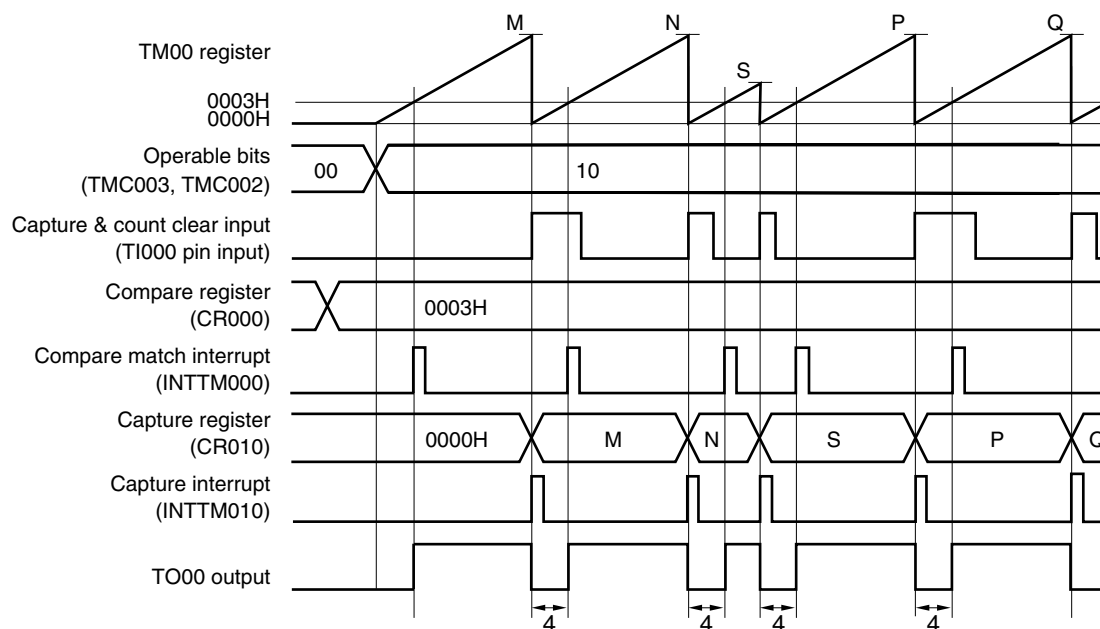


应用示例：捕捉到计数值并清零时，反转 TO00 输出电平。

检测出 TI000 引脚的有效沿时，捕捉计数值存入 CR010，并将 TM00 清零(0000H)。TM00 计算值为“0001H”时，产生比较匹配中断信号(INTTM000)，并反转 TO00。

图 6-25. 通过输入 TI000 引脚的有效沿，进入清零并启动模式的时序示例
(CR000:比较寄存器, CR010:捕捉寄存器) (2/2)

(b) TOC00 = 13H, PRM00 = 10H, CRC00 = 04H, TMC00 = 0AH, CR000 = 0003H



应用示例：捕捉计算值并清零时，从 TO00 输出 CR000（本例使用 4 时钟）设置的宽度。

检测出 TI000 引脚的有效沿时，捕捉的计数值存入 CR010，产生捕捉中断信号(INTTM010)，TM00 清为 0000H，并反转 TO00 输出电平。TM00 计算值为“0003H”（已经计数 4 个时钟）时，产生比较匹配中断信号(INTTM000)，并反转 TO00。

- (3) 通过输入 TI000 引脚有效沿，进入清零并启动模式
(CR000: 捕捉寄存器, CR010: 比较寄存器)

图 6-26. 通过输入 TI000 引脚有效沿，进入清零并启动模式的框图
(CR000:捕捉寄存器, CR010:比较寄存器)

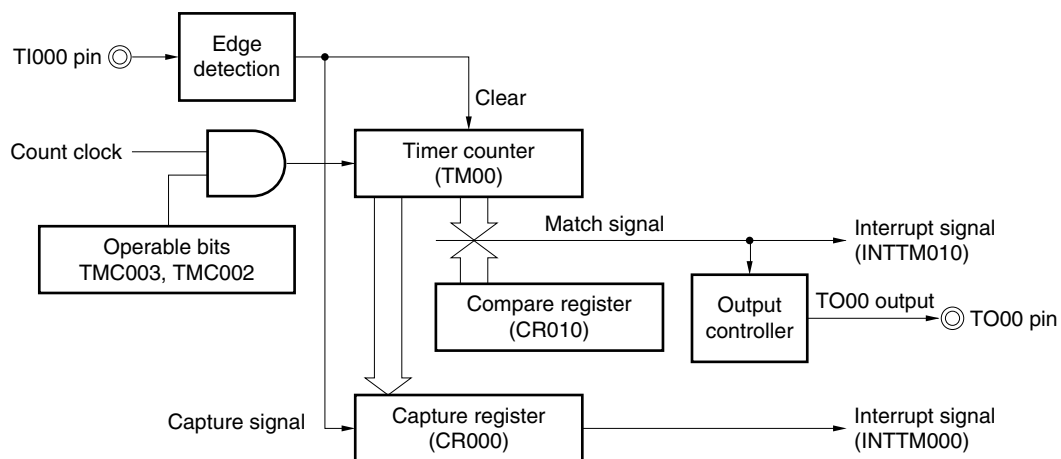
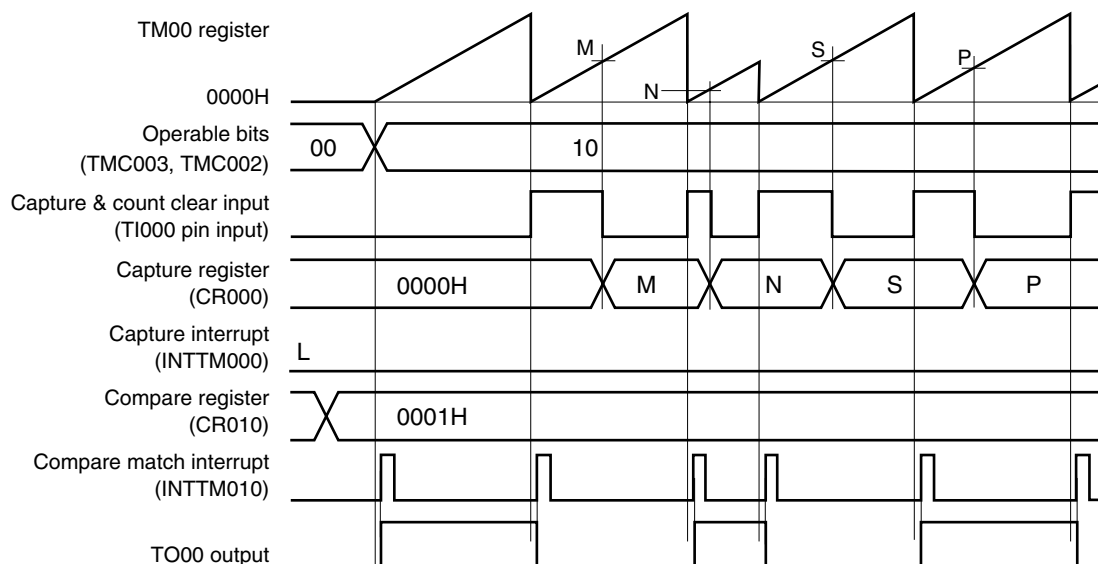


图 6-27. 通过输入 TI000 引脚的有效沿，进入清零并启动模式的时序示例
(CR000:捕捉寄存器, CR010:比较寄存器) (1/2)

(a) TOC00 = 13H, PRM00 = 10H, CRC00 = 03H, TMC00 = 08H, CR010 = 0001H



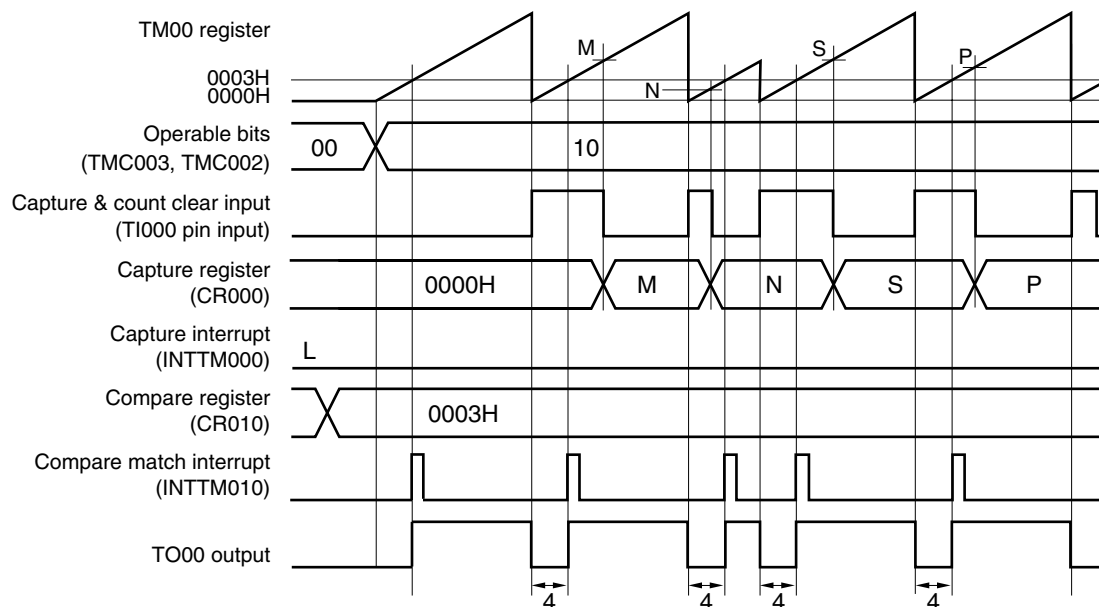
应用示例：捕捉到计数值并清零时，反转 TO00 输出电平。

检测出 TI000 引脚的上升沿时，TM00 清零；检测出 TI000 引脚的下降沿时，捕捉的计数值存入 CR000。

捕捉/比较控制寄存器 00(CRC00)的第 1 位(CRC001)置“1”时，用 TI000 引脚输入信号的反向沿捕捉 TM00 的计数值并存入 CR000，但不产生捕捉中断信号(INTTM000)。但检测出 TI010 引脚的有效边沿时，将会产生 INTTM000 信号。若不使用 INTTM000 信号，则将其屏蔽。

图 6-27. 通过输入 TI000 引脚的有效沿，进入清零并启动模式的时序示例
(CR000:捕捉寄存器, CR010:比较寄存器) (2/2)

(b) TOC00 = 13H, PRM00 = 10H, CRC00 = 03H, TMC00 = 0AH, CR010 = 0003H



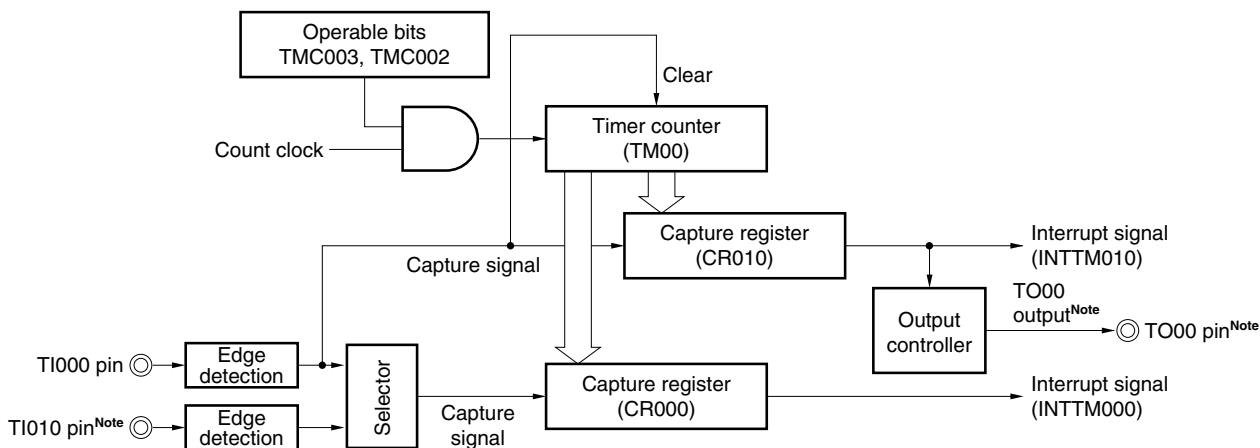
应用示例：捕捉计算值并清零时，从 TO00 输出 CR010（本例使用 4 时钟）设置的宽度。

检测出 TI000 引脚的上升沿时，TM00 清零(0000H)；检测出 TI000 引脚的下降沿时，捕捉的计数值存入 CR000。检测出 TI000 引脚的有效沿或 TM00 与比较寄存器(CR010)相匹配，TO00 清为 0000H 时，反转 TO00 输入电平。

捕捉/比较控制寄存器 00(CRC00)的第 1 位(CRC001)置“1”时，用 TI000 引脚输入信号的反向沿捕捉 TM00 的计数值并存入 CR000，但不产生捕捉中断信号(INTTM000)。但检测出 TI010 引脚的有效边沿时，将会产生 INTTM000 中断。若不使用 INTTM000 信号，则将其屏蔽。

- (4) 通过输入 TI000 引脚的有效沿, 进入清零并启动模式
(CR000: 捕捉寄存器, CR010: 比较寄存器)

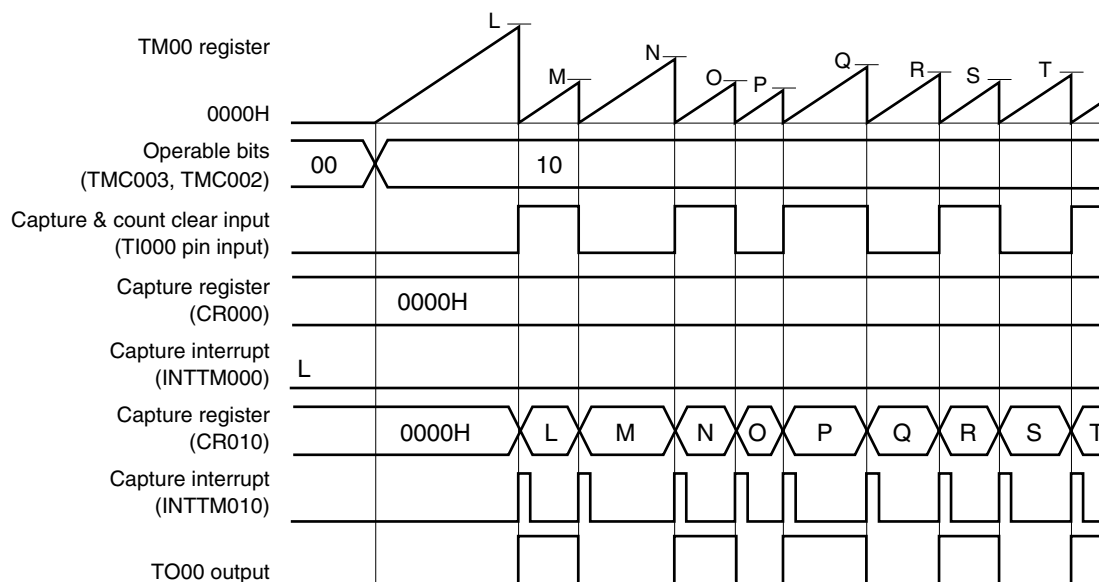
图 6-28. 通过输入 TI000 引脚有效沿, 进入清零并启动模式的框图
(CR000:捕捉寄存器, CR010:捕捉寄存器)



注 使用 TI010 引脚检测有效沿时, 禁止使用定时器输出(TO00)。

图 6-29. 通过输入 TI000 引脚的有效沿, 进入清零并启动模式的时序示例
(CR000:捕捉寄存器, CR010:捕捉寄存器) (1/3)

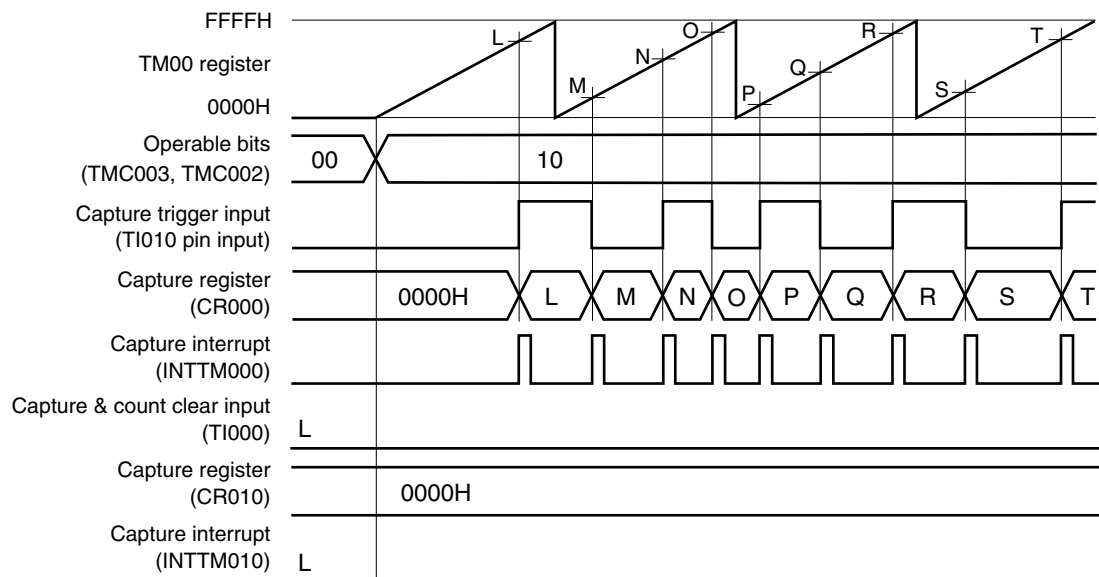
(a) TOC00 = 13H, PRM00 = 30H, CRC00 = 05H, TMC00 = 0AH



使用示例: 检测出 TI000 端口的上升沿或下降沿时, 捕捉计数值并存入 CR010, TM00 清零, 并反转 TO00 输出。
检测出 TI010 引脚的脉冲沿时, 产生中断信号(INTTM000)。若不使用 INTTM000 信号, 则将其屏蔽。

图 6-29. 通过输入 TI000 引脚的有效沿，进入清零并启动模式的时序示例
(CR000:捕捉寄存器, CR010:捕捉寄存器) (2/3)

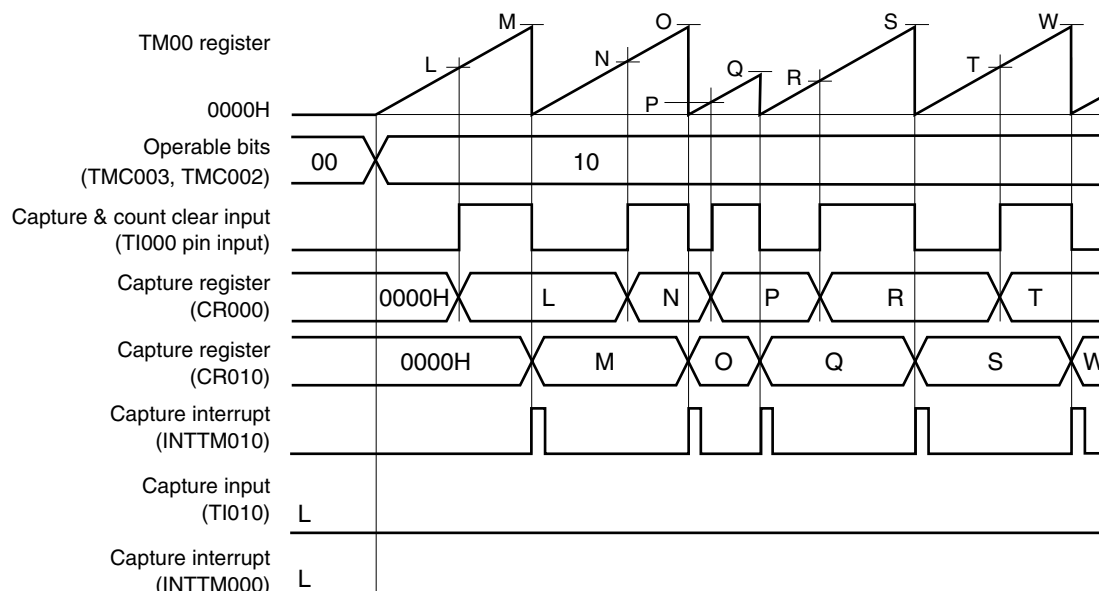
(b) TOC00 = 13H, PRM00 = C0H, CRC00 = 05H, TMC00 = 0AH



应用示例：检测出 TI010 引脚的上升沿或下降沿时，捕捉计数值并存入 CR000，脉冲沿不输入到 TI000 引脚。

图 6-29. 通过输入 TI000 引脚的有效沿，进入清零并启动模式的时序示例
(CR000:捕捉寄存器, CR010:捕捉寄存器) (3/3)

(c) TOC00 = 13H, PRM00 = 00H, CRC00 = 07H, TMC00 = 0AH



应用示例：检测 TI000 引脚输入信号的脉冲宽度。

通过设置 CRC00，在 TI000 引脚下降沿的反向沿(即，上升沿)捕捉计数值并存入 CR000；而在 TI000 引脚下降沿则捕捉计数值并存入 CR010。

通过以下表达式可以计算得出输入脉冲的高电平和低电平宽度。

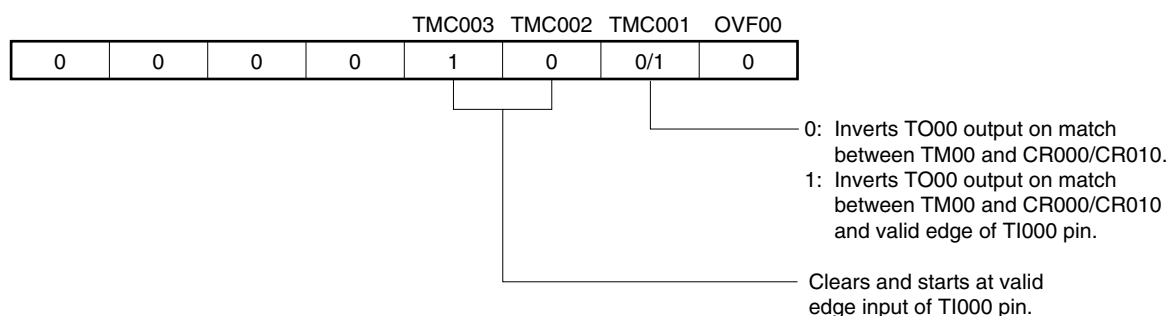
- 高电平宽度 = [CR010 值] - [CR000 值] × [计数时钟周期]
- 低电平宽度 = [CR000 值] × [计数时钟周期]

若 TI000 引脚的反向沿用作触发，捕捉计数值并存入 CR000，则不产生 INTTM000 信号。产生 INTTM010 信号后，立即读取 CR000 和 010，测量脉冲宽度。

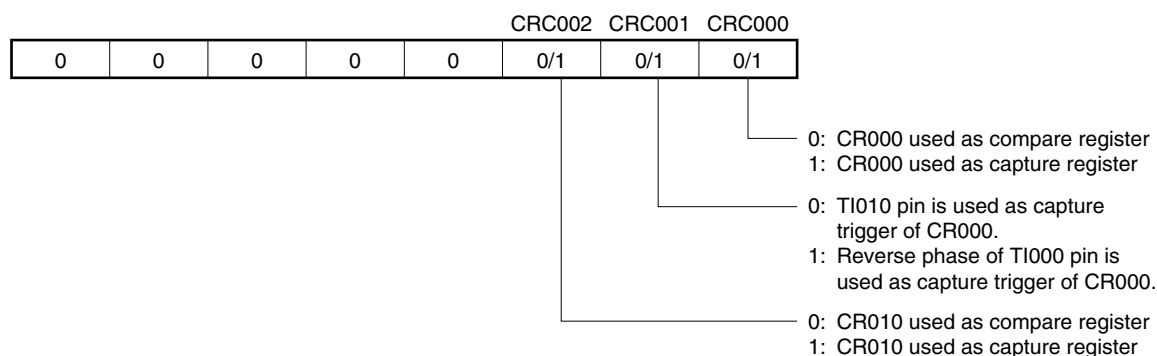
但是，若通过预分频模式寄存器 00(PRM00)的第 6 位和第 5 位(ES110 和 ES100)指定有效沿输入 TI010 引脚，则不能捕捉计数值，但产生 INTTM000 信号。测量 TI000 引脚的脉冲宽度时，若不使用 INTTM000 则将其屏蔽。

图 6-30. 通过输入 TI000 引脚有效沿, 进入清零并启动模式下的寄存器设置示例 (1/2)

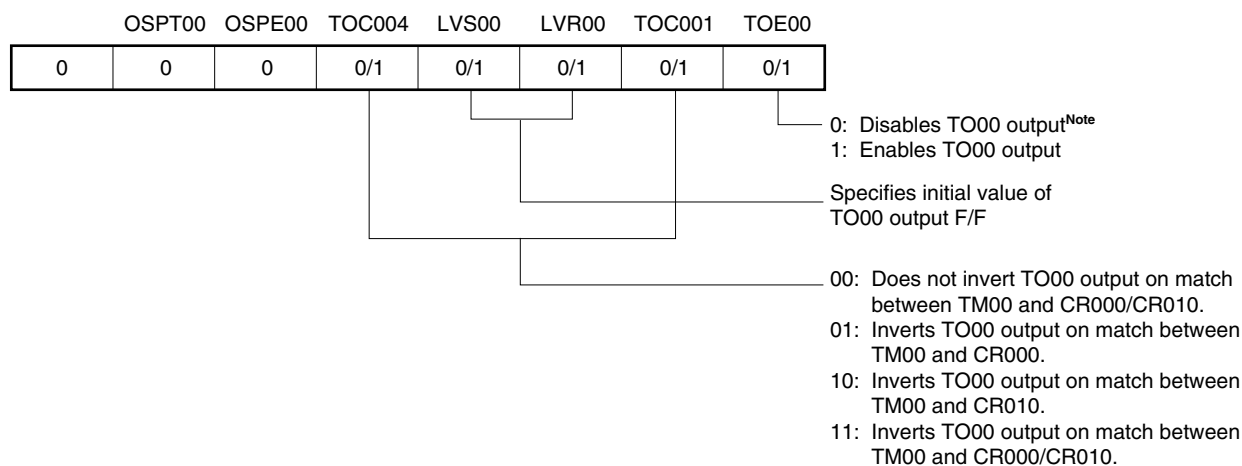
(a) 16 位定时器模式控制寄存器 00(TMC00)



(b) 捕捉/比较控制寄存器 00(CRC00)



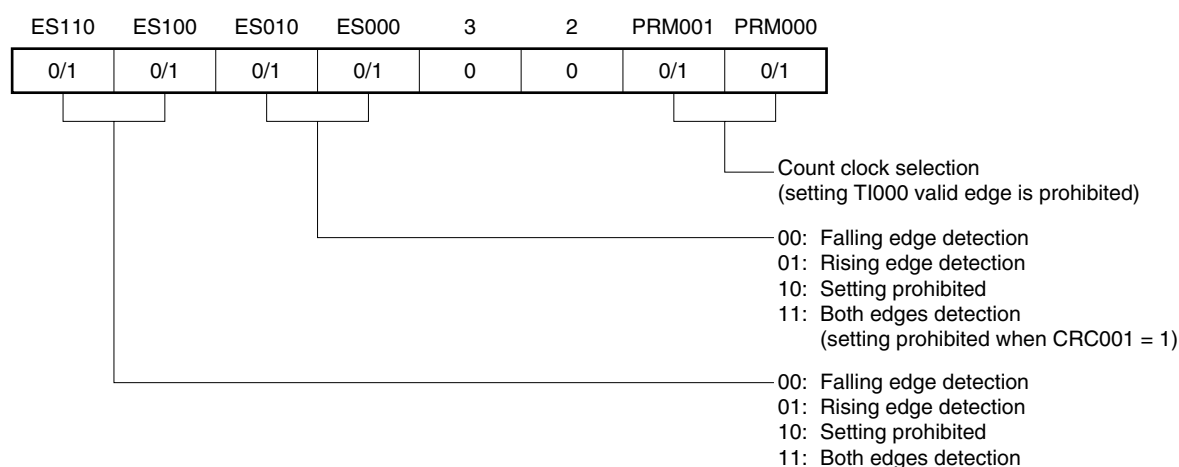
(c) 16 位定时器输出控制寄存器 00(TOC00)



注 使用 TI010 引脚检测有效沿时, 禁止使用定时器输出(TO00)。

图 6-30. 通过输入 TI000 引脚有效沿，进入清零并启动模式下的寄存器设置示例 (2/2)

(d) 预分频模式寄存器 00(PRM00)



(e) 16 位定时器/计数器 00(TM00)

通过读取 TM00，获取计数值。

(f) 16 位定时器捕捉/比较寄存器 000(CR000)

该寄存器用作比较寄存器并与 TM00 的值相匹配时，产生中断信号(INTTM000)。TM00 不能清零。

若该寄存器用作捕捉寄存器，则选择 TI000 或 TI010 引脚^注输入作为捕捉触发。检测出捕捉触发的有效沿时，将 TM00 计数值存入 CR000。

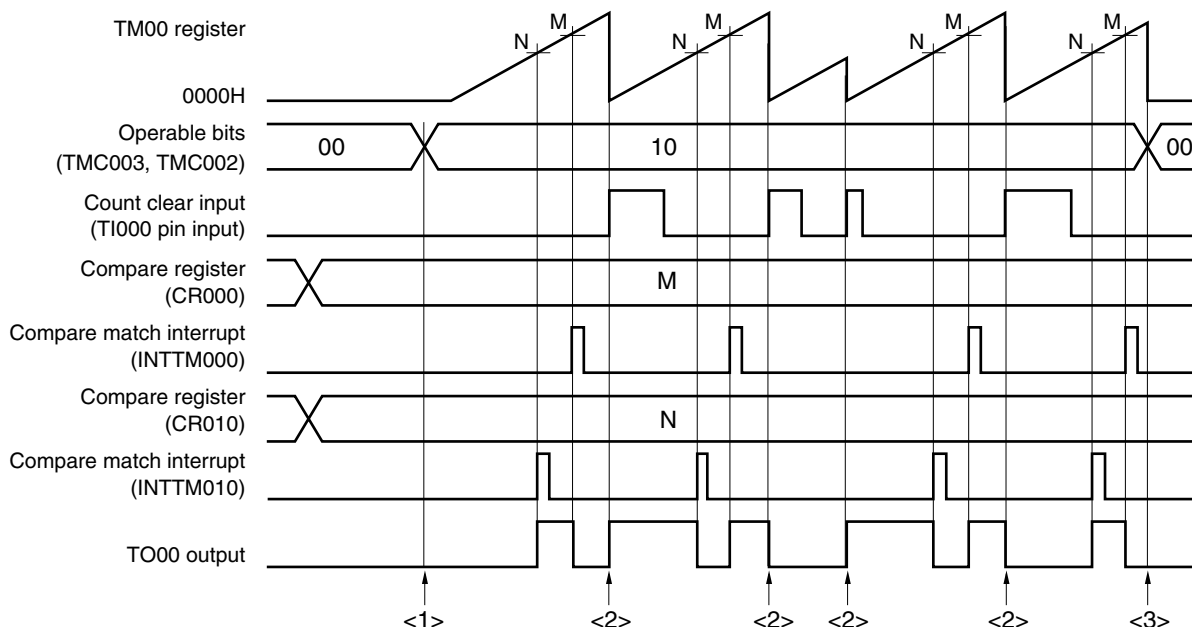
注 使用 TI010 引脚检测有效沿时，禁止使用定时器输出(TO00)。

(g) 16 位捕捉/比较寄存器 010(CR010)

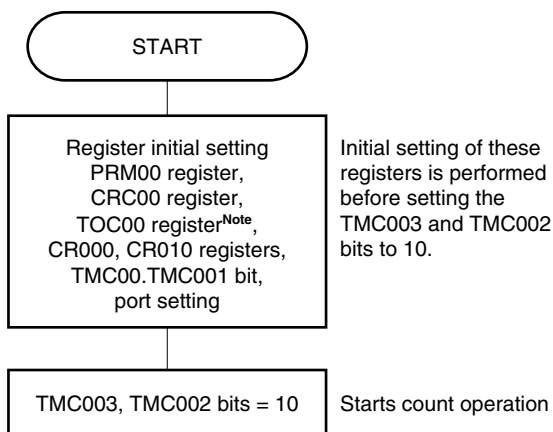
该寄存器用作比较寄存器并与 TM00 的值相匹配时，产生中断信号(INTTM010)。TM00 不能清零。

该寄存器用作捕捉寄存器时，TI000 引脚输入作为捕捉触发。检测出捕捉触发的有效沿时，将 TM00 计数值存入 CR010。

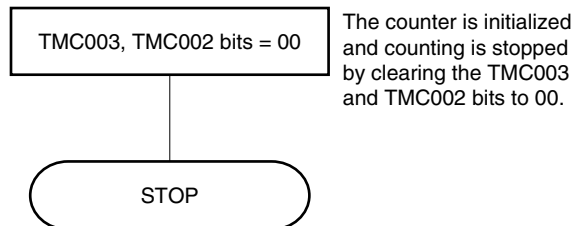
图 6-31. 通过输入 TI000 引脚有效沿, 进入清零并启动模式的软件处理示例



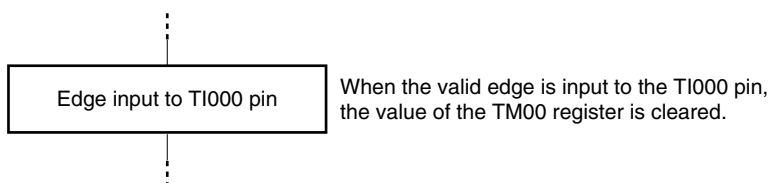
<1> Count operation start flow



<3> Count operation stop flow



<2> TM00 register clear & start flow



注 设置 TOC00 时, 应特别注意。详情参阅 6.3 (3) 16 位定时器输出控制寄存器 00(TOC00)。

6.4.5 自由运行定时器操作

16 位定时器模式控制寄存器的第 3 位和第 2 位置“01”（自由运行定时器模式）时，16 位定时器/事件计数器 00 与计数时钟同步连续计数。计数到 FFFFH 时，在下一个时钟将溢出标志 (OVF00) 置“1”，TM00 清零并继续计数。通过软件执行 CLR 指令将 OVF00 清零。

提供以下三种自由运行定时器操作。

- CR000 和 CR010 均用作比较寄存器
- CR000 和 CR010，其中一个用作比较寄存器，另一个用作捕捉寄存器。
- CR000 和 CR010 均用作捕捉寄存器

- 备注 1. 有关 I/O 引脚的设置，参阅 6.3 (5) 端口模式寄存器 3(PM3)
2. 有关如何允许 INTTM000 信号中断，参阅第十四章 中断功能

(1) 自由运行定时器操作

(CR000: 捕捉寄存器, CR010: 比较寄存器)

图 6-32. 自由运行定时器模式系统图
(CR000:比较寄存器, CR010:比较寄存器)

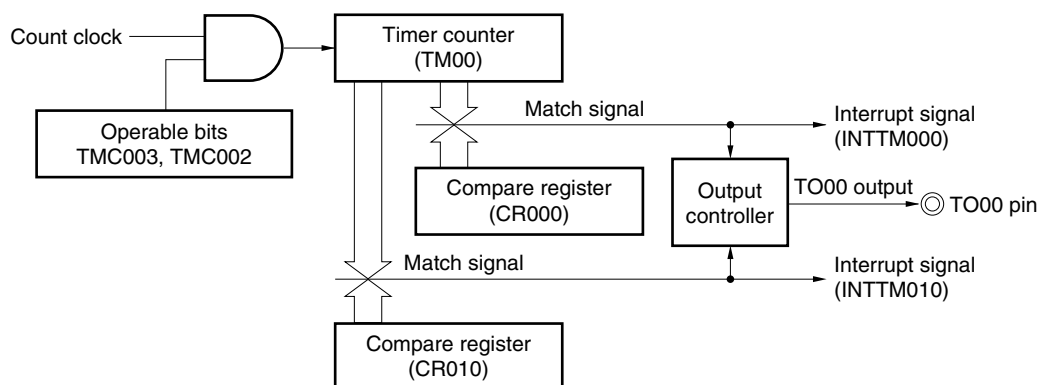
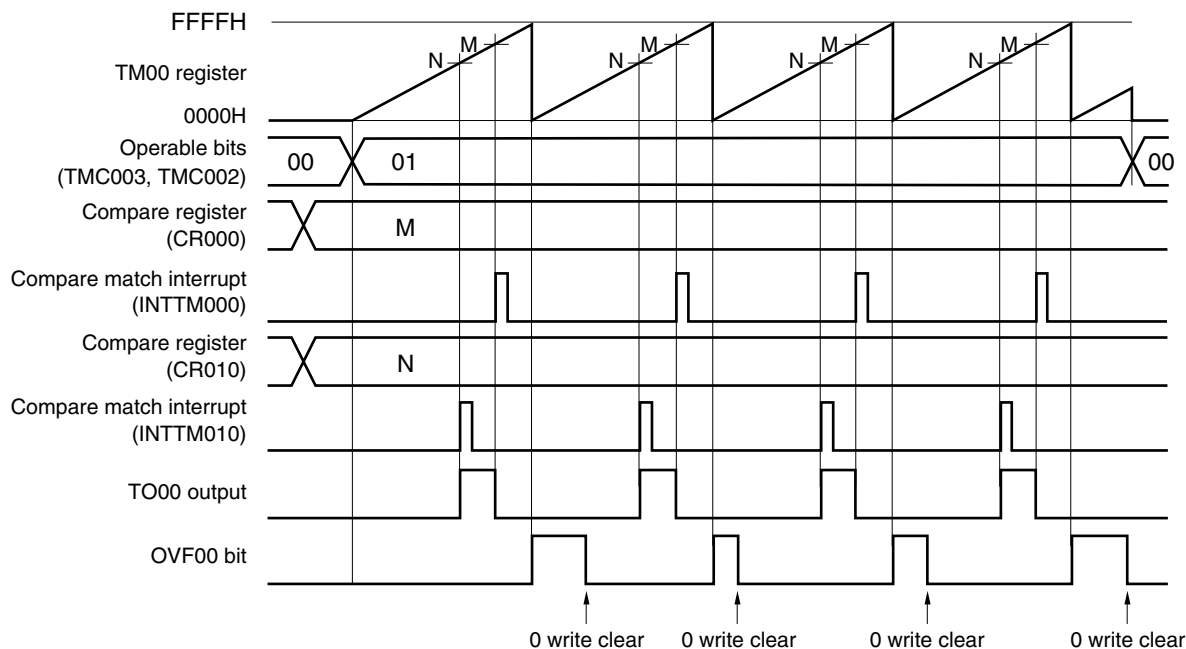


图 6-33. 自由运行定时器模式的时序示例
(CR000:比较寄存器, CR010:比较寄存器)

• TOC00 = 13H, PRM00 = 00H, CRC00 = 00H, TMC00 = 04H



应用示例：在自由运行定时器模式下，使用 2 个比较寄存器。

每当 TM00 的计数值与 CR000 或 CR010 的值相匹配时，反转 TO00 引脚的输出电平。计数值与寄存器值相匹配时，产生 INTTM000 或 INTTM010 信号。

(2) 自由运行定时器操作

(CR000: 比较寄存器, CR010: 捕捉寄存器)

图 6-34. 自由运行定时器模式的框图
(CR000:比较寄存器, CR010:捕捉寄存器)

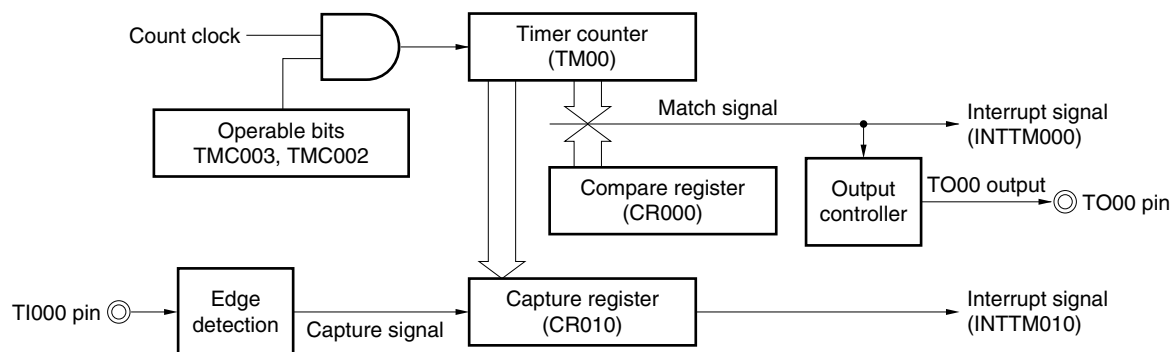
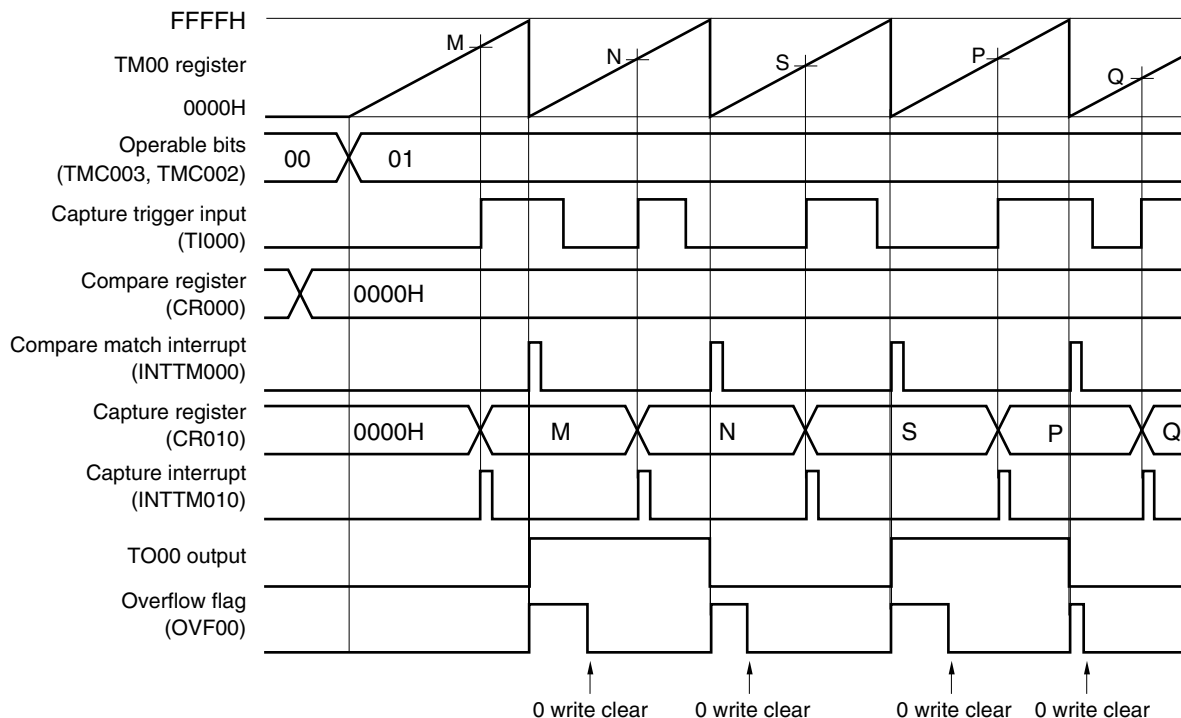


图 6-35. 自由运行定时器模式的时序示例
(CR000:比较寄存器, CR010:捕捉寄存器)

• TOC00 = 13H, PRM00 = 10H, CRC00 = 04H, TMC00 = 04H



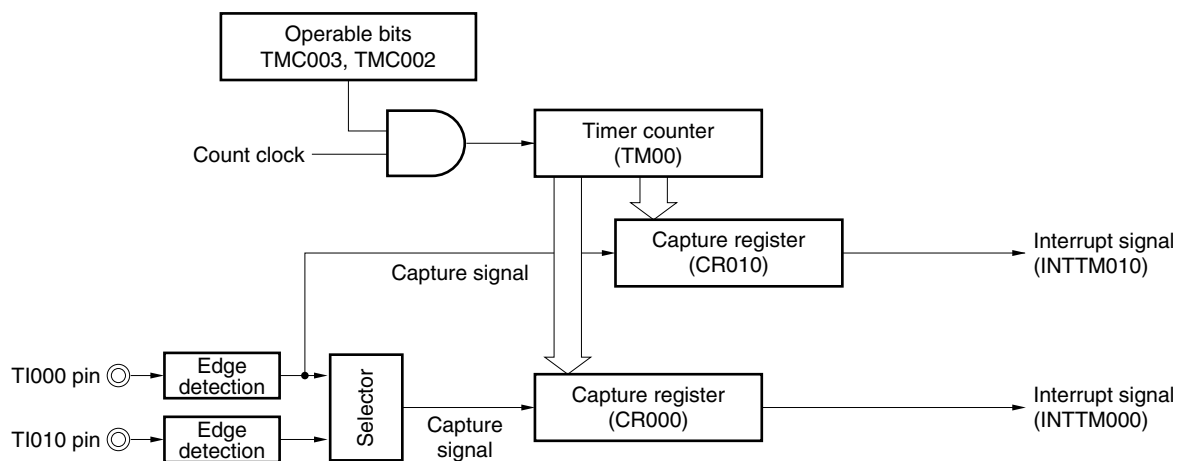
应用示例：在自由运行定时器模式下，同时使用一个比较寄存器和一个捕捉寄存器。

在本例中，每当 TM00 的计数值和 CR000 的计数值（比较寄存器）相匹配时，产生 INTTM000 信号，反转 TO00 输出电平。另外，每当检测出 TI000 引脚时，产生 INTTM010 信号，捕捉 TM00 的计数值并存入 CR010。

(3) 自由运行定时器操作

(CR000: 捕捉寄存器, CR010: 比较寄存器)

图 6-36. 自由运行定时器模式的框图
(CR000:捕捉寄存器, CR010:捕捉寄存器)

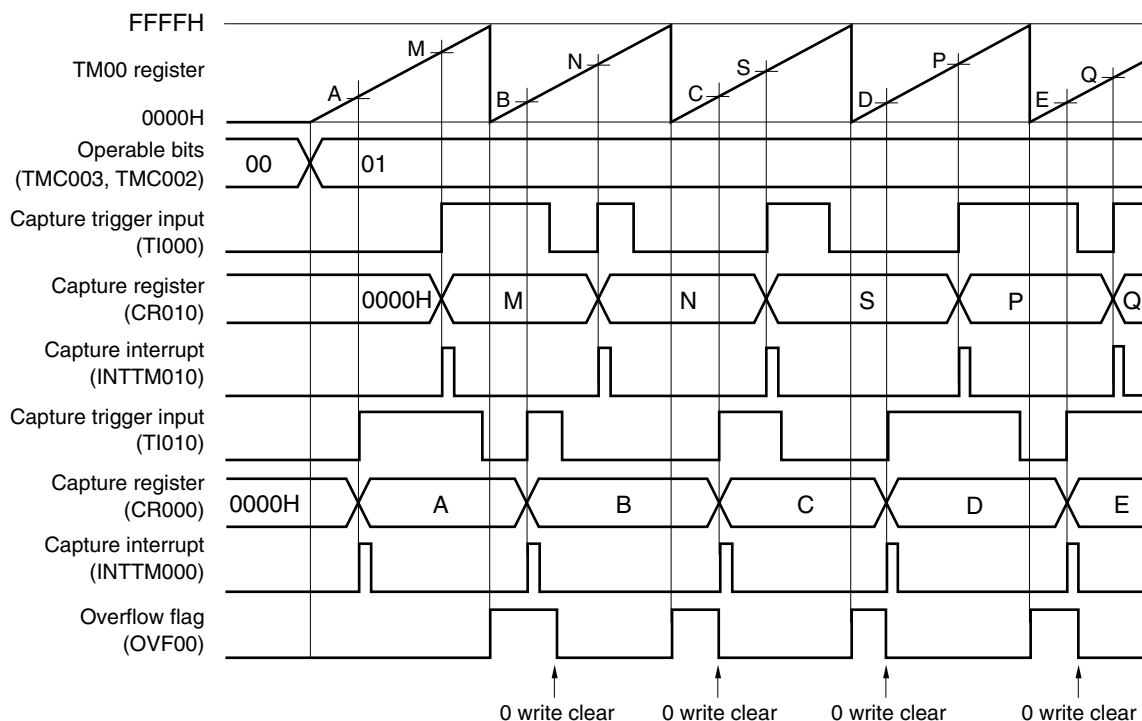


备注 自由运行定时器模式下，若 CR000 和 CR010 均用作捕捉寄存器，则不反转 TO00 输出电平。

但若 16 位定时器模式控制寄存器 00(TMC00)的第 1 位置“1”，则每次检测出的 TI000 引脚的有效沿可以反转。

图 6-37. 自由运行定时器模式的时序示例
(CR000:捕捉寄存器, CR010:捕捉寄存器) (1/2)

(a) TOC00 = 13H, PRM00 = 50H, CRC00 = 05H, TMC00 = 04H

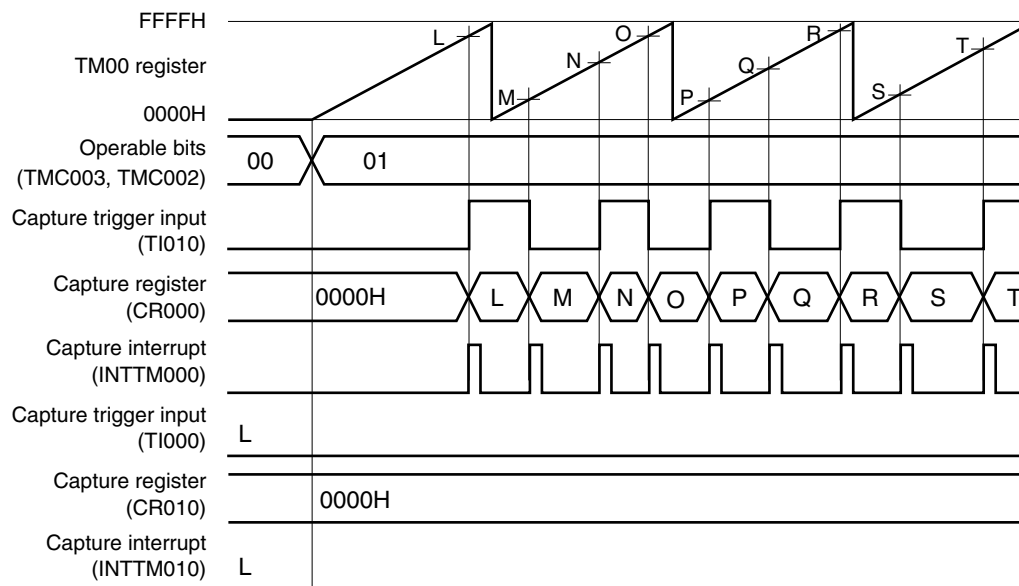


应用示例：自由运行定时器模式下，在各自捕捉触发输入的有效沿捕捉的计数值存入各自的捕捉寄存器中。

检测出 TI000 引脚输入的有效沿时，捕捉计数值并存入 CR010；检测出 TI010 引脚输入的有效沿时，捕捉计数值并存入 CR000。

图 6-37. 自由运行定时器模式的时序示例
(CR000:捕捉寄存器, CR010:捕捉寄存器) (2/2)

(b) TOC00 = 13H, PRM00 = C0H, CRC00 = 05H, TMC00 = 04H

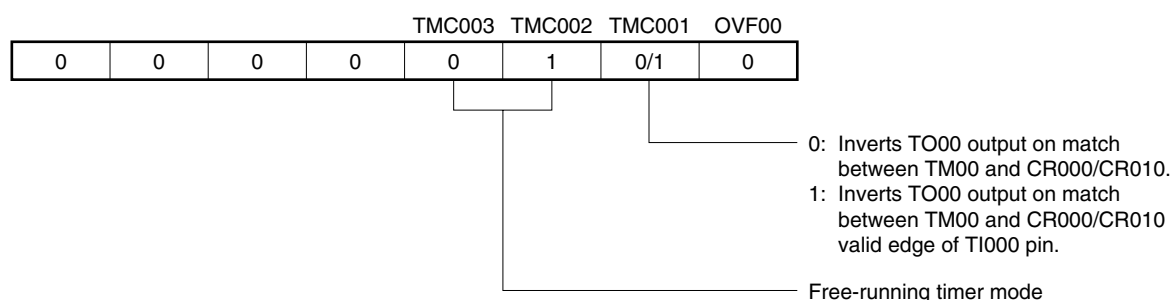


应用示例：自由运行定时器模式下，检测出 TI010 引脚的双边沿，捕捉计数值并存入 CR000。

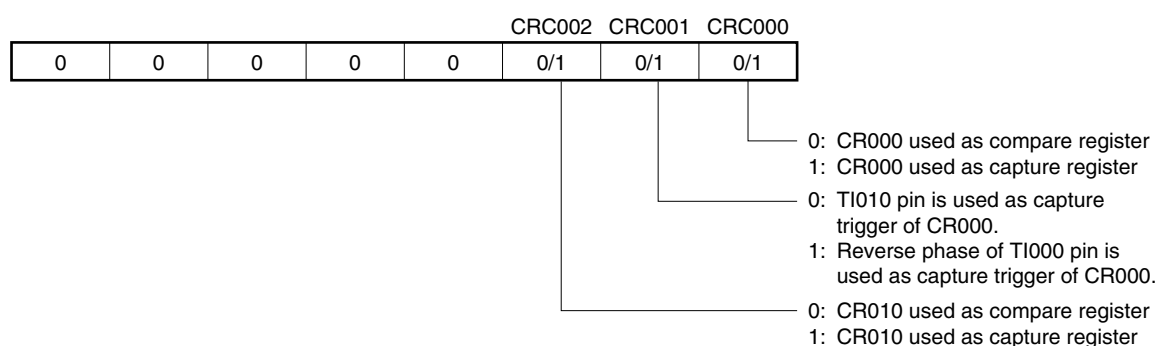
CR000 和 CR010 均用作捕捉寄存器并且只能检测出 TI010 引脚的有效沿时，捕捉不到计数值并存入 CR010。

图 6-38. 自由运行定时器模式下的寄存器设置示例(1/2)

(a) 16 位定时器模式控制寄存器 00(TMC00)



(b) 捕捉/比较控制寄存器 00(CRC00)



(c) 16 位定时器输出控制寄存器 00(TOC00)

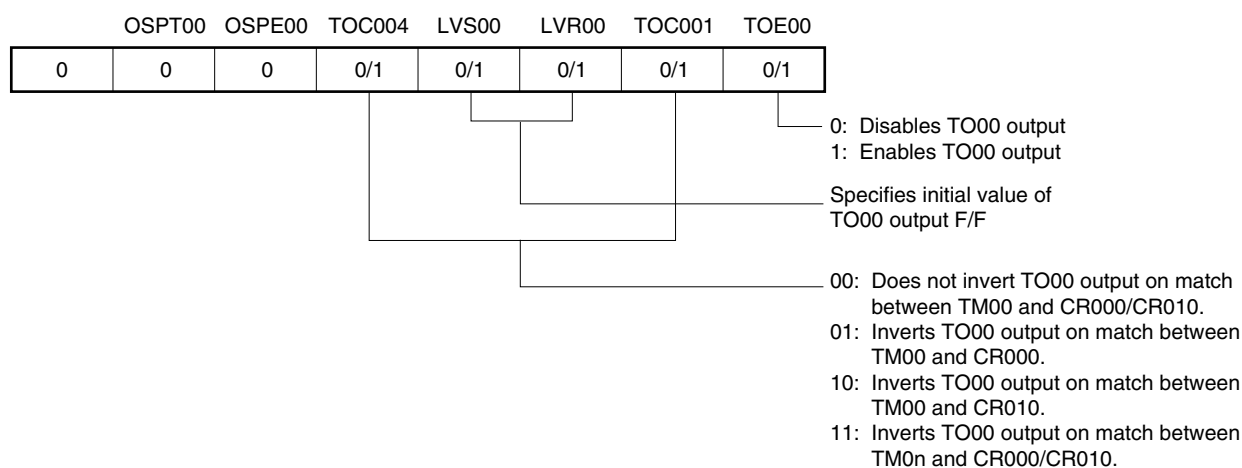
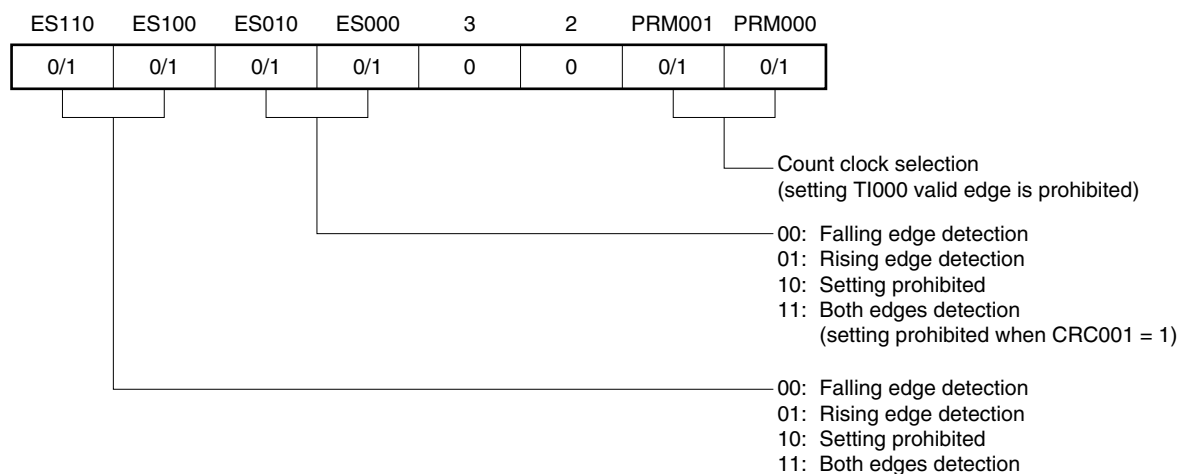


图 6-38. 自由运行定时器模式下的寄存器设置示例(2/2)

(d) 预分频模式寄存器 00(PRM00)



(e) 16 位定时器/计数器 00(TM00)

通过读取 TM00，获取计数值。

(f) 16 位定时器捕捉/比较寄存器 000(CR000)

该寄存器用作比较寄存器并与 TM00 的值相匹配时，产生中断信号(INTTM000)。TM00 不能清零。

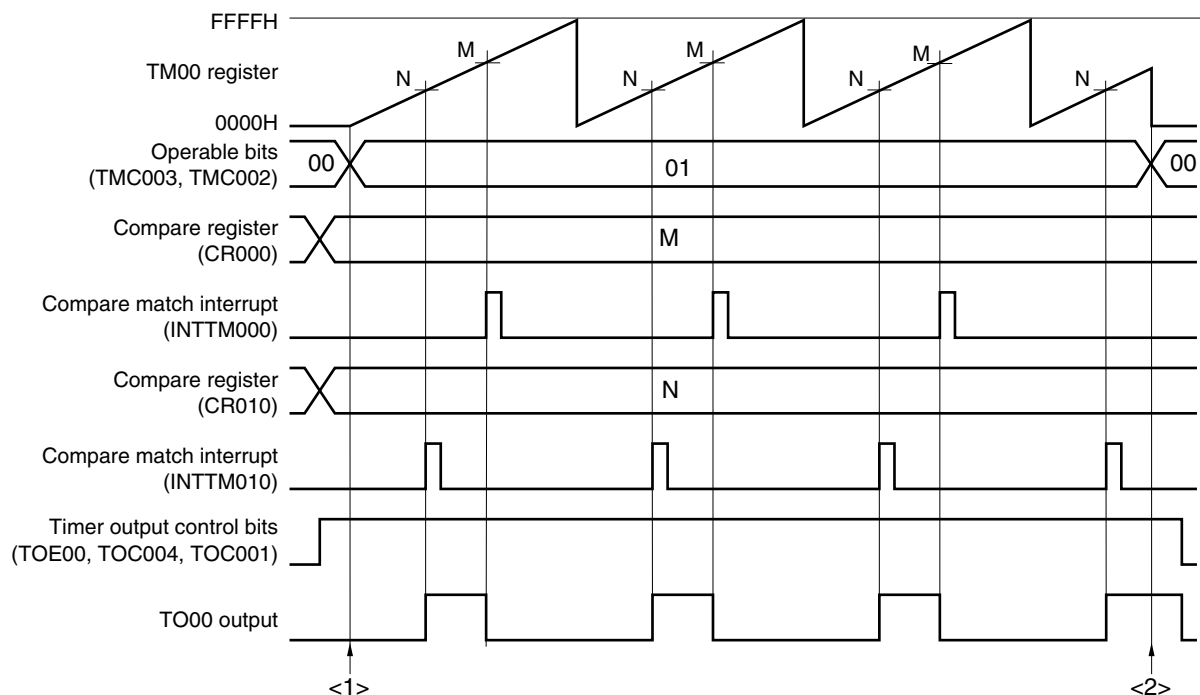
若该寄存器用作捕捉寄存器，则选择 TI000 或 TI010 引脚输入作为捕捉触发。检测出捕捉触发的有效沿时，将 TM00 计数值存入 CR000。

(g) 16 位捕捉/比较寄存器 010(CR010)

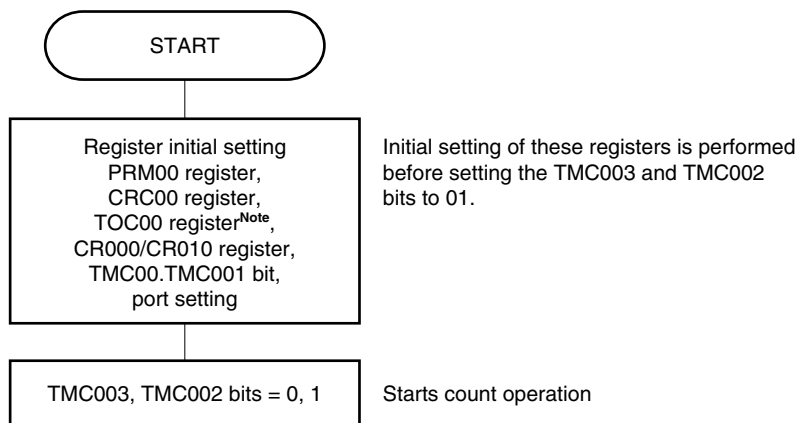
该寄存器用作比较寄存器并与 TM00 的值相匹配时，产生中断信号(INTTM010)。TM00 不能清零。

该寄存器用作捕捉寄存器时，TI000 引脚输入作为捕捉触发。检测出捕捉触发的有效沿时，将 TM00 计数值存入 CR010。

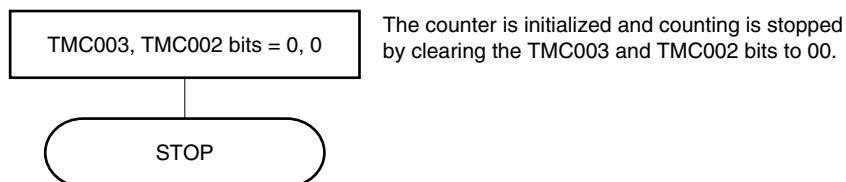
图 6-39. 自由运行定时器模式的软件处理示例



<1> Count operation start flow



<2> Count operation stop flow



注 设置 TOC00 时，应特别注意。参阅 6.3 (3) 16 位定时器输出控制寄存器 00(TOC00)。

6.4.6 PPG 输出操作

16 位定时器模式控制寄存器 00(TMC00)的第 3 位和第 2 位(TMC003 和 TMC002) 置“11”(TM00 和 CR000 相匹配时, 清零并启动)时, 在 CR000 的设定值周期内, 从 TO00 引脚通过 CR010 将预先设有脉冲宽度的方波用作 PPG (可编程脉冲发生器)信号。

PPG 输出产生的脉冲周期和占空比如下所示。

- 脉冲周期 = (CR000 设定值 + 1) × 计数时钟周期
- 占空比 = (CR010 设定值 + 1) / (CR000 设定值 + 1)

注意事项 有关操作期间修改占空比(CR010 值), 参阅 6.5.1 TM00 操作期间 CR010 的重写。

- 备注**
1. 有关 I/O 引脚的设置, 参阅 6.3 (5) 端口模式寄存器 3(PM3)
 2. 有关如何允许 INTTM000 信号中断, 参阅第十四章 中断功能

图 6-40.PPG 输出操作系统图

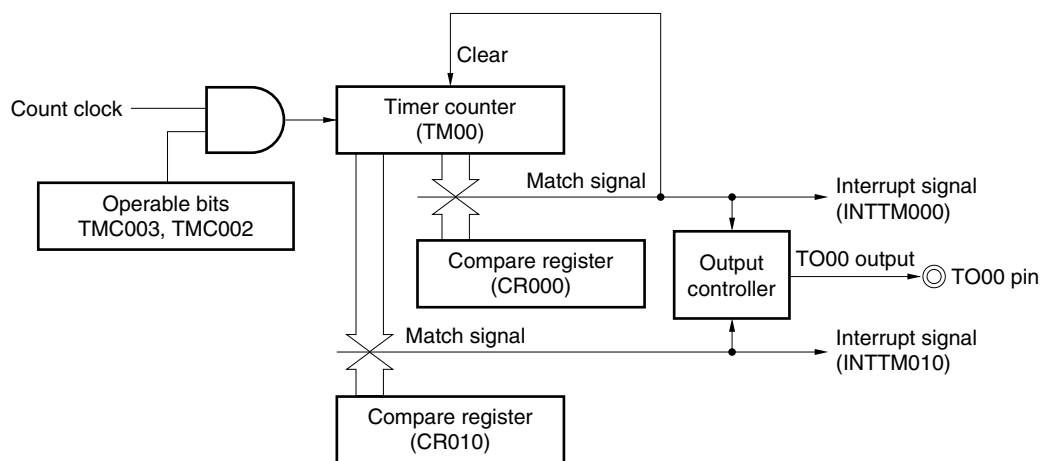
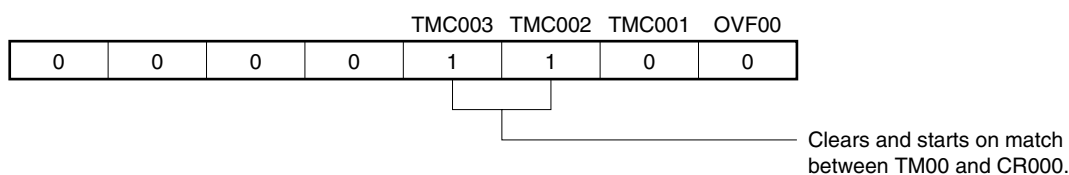
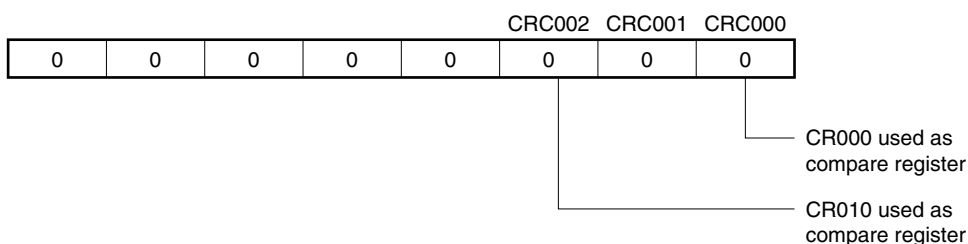


图 6-41. PPG 输出操作的寄存器设置示例(1/2)

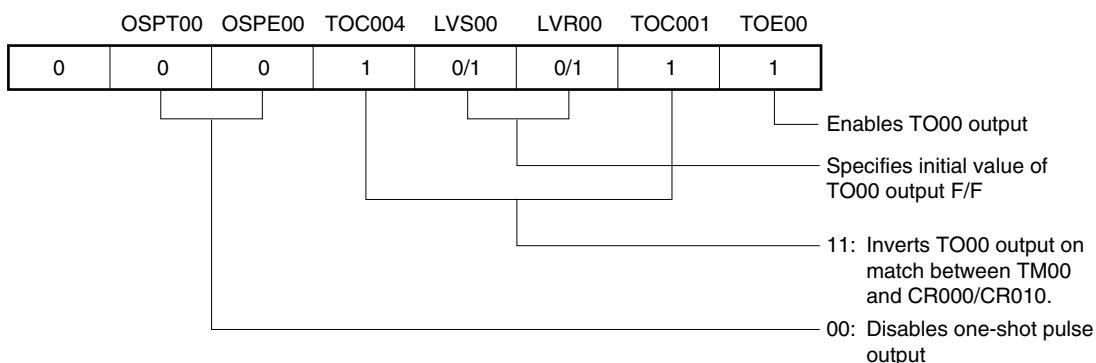
(a) 16 位定时器模式控制寄存器 00(TMC00)



(b) 捕捉/比较控制寄存器 00(CRC00)



(c) 16 位定时器输出控制寄存器 00(TOC00)



(d) 预分频模式寄存器 00(PRM00)

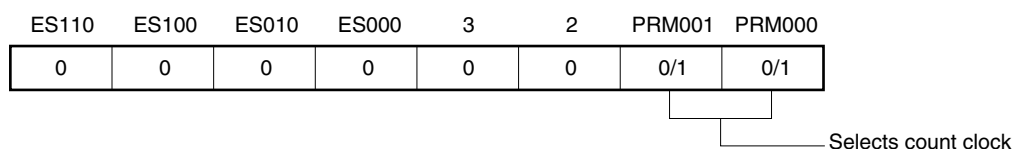


图 6-41. PPG 输出操作的寄存器设置示例(2/2)

(e) 16 位定时器/计数器 00(TM00)

通过读取 TM00，获取计数值。

(f) 16 位定时器捕捉/比较寄存器 000(CR000)

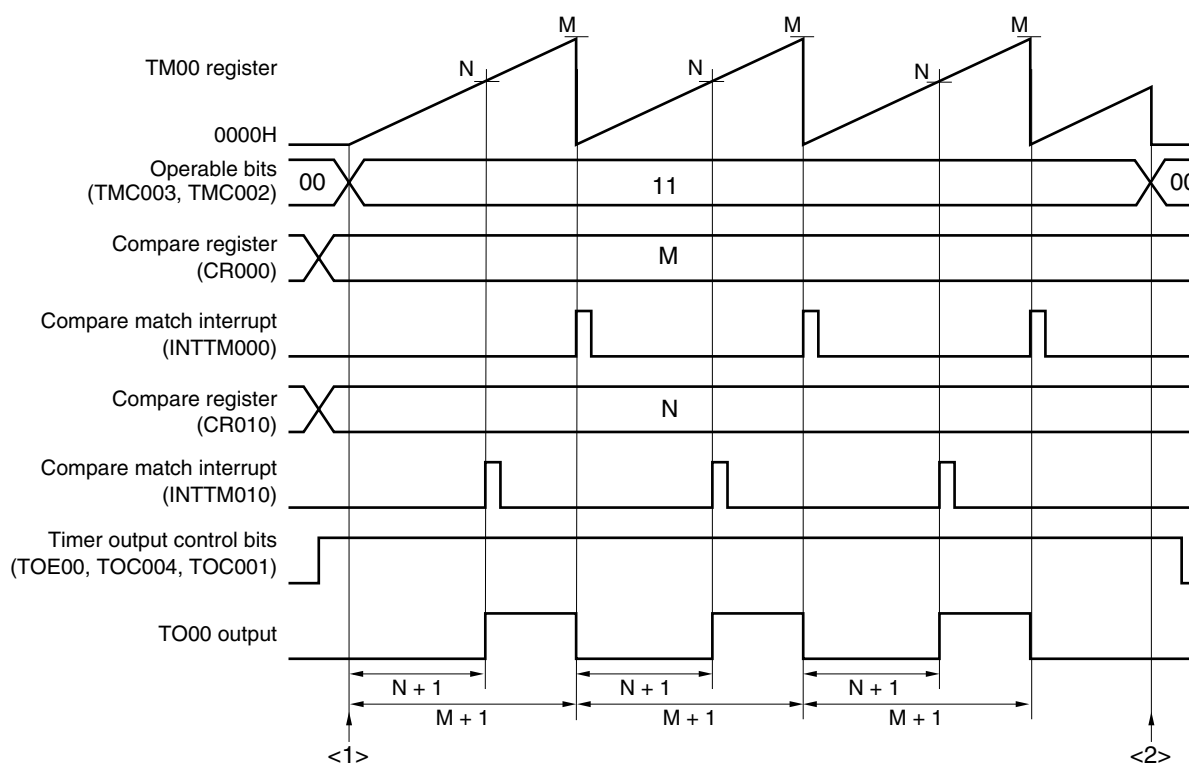
该寄存器的值与 TM00 的计数值相匹配时，产生中断信号(INTTM000)。TM00 的计数值清零。

(g) 16 位捕捉/比较寄存器 010(CR010)

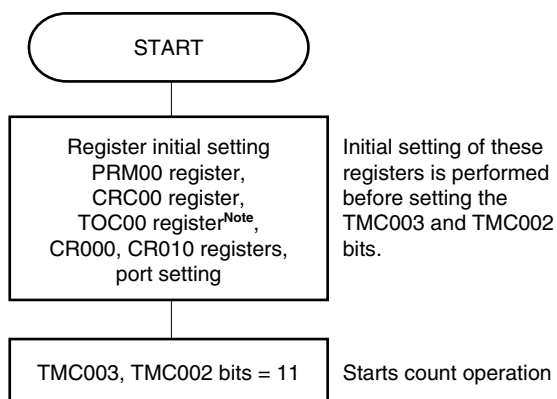
该寄存器的值与 TM00 的计数值相匹配时，产生中断信号(INTTM010)。TM00 不能清零。

注意事项 在 CR000 和 CR010 设置满足条件 $0000H \leq CR010 < CR000 \leq FFFFH$ 的值。

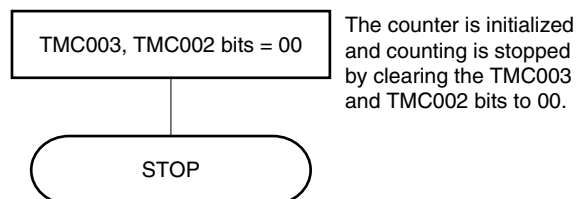
图 6-42. PPG 输出操作的软件处理示例



<1> Count operation start flow



<2> Count operation stop flow



注 设置 TOC00 时，应特别注意。详情参阅 6.3 (3) 16 位定时器输出控制寄存器 00(TOC00)。

注意事项 PPG 脉冲周期 = $(M + 1) \times$ 计数时钟

PPG 占空比 = $(N + 1)/(M + 1)$

6.4.7 单脉冲输出操作

通过 16 位寄存器模式控制寄存器 00(TMC00)的第 3 位和第 2 位 (TMC003 和 TMC002)置“01”（自由运行定时器模式）或置“10”（通过 TI000 引脚的有效沿进入清零并启动模式），以及 16 位定时器输出控制寄存器 00(TOC00)的第 5 位(OSPE00)置“1”，可以输出单脉冲。

定时器运行期间，TOC00 的第 6 位(OSPT00)置“1”或有效沿输入 TI000 引脚时，触发 TM00 清零并启动，并从 TO00 引脚只输出一 CR000 和 CR010 之间的不同脉冲。

- 注意事项**
1. 在单脉冲输出时，禁止再次输入触发（OSPT00 置“1”或检测 TI000 引脚的有效沿）。当前单脉冲输出完成后产生触发，才可再次输出单脉冲。
 2. 仅将 OSPT00 置“1”来触发单脉冲输出时，禁止改变 TI000 引脚或其复用功能端口引脚的电平。否则，可能出现脉冲输出异常。

- 备注**
1. 有关 I/O 引脚的设置，参阅 6.3 (5) 端口模式寄存器 3(PM3)
 2. 有关如何允许 INTTM000 信号中断，参阅第十四章 中断功能

图 6-43. 单脉冲输出操作系统图

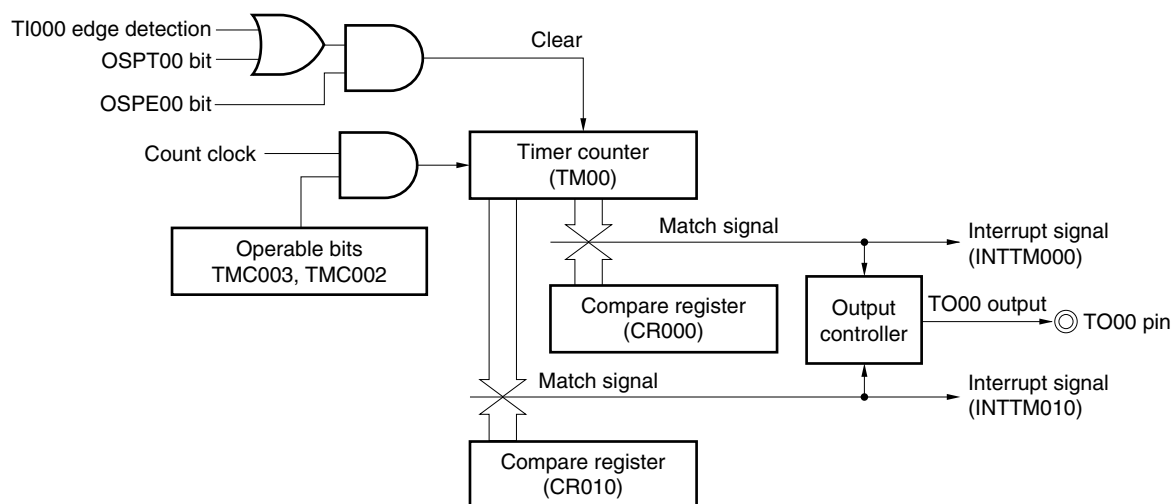


图 6-44. 单脉冲输出操作的寄存器设置示例(1/2)

(a) 16 位定时器模式控制寄存器 00(TMC00)

TMC003				TMC002		TMC001	OVF00
0	0	0	0	0/1	0/1	0	0

01: Free running timer mode
10: Clear and start mode by valid edge of TI000 pin.

(b) 捕捉/比较控制寄存器 00(CRC00)

CRC002				CRC001		CRC000	
0	0	0	0	0	0	0	0

CR000 used as compare register

CR010 used as compare register

(c) 16 位定时器输出控制寄存器 00(TOC00)

OSPT00	OSPE00	TOC004	LVS00	LVR00	TOC001	TOE00	
0	0/1	1	1	0/1	0/1	1	1

Enables TO00 output

Specifies initial value of TO00 output

Inverts TO00 output on match between TM00 and CR000/CR010.

Enables one-shot pulse output

Software trigger is generated by writing 1 to this bit (operation is not affected even if 0 is written to it).

(d) 预分频模式寄存器 00(PRM00)

ES110	ES100	ES010	ES000	3	2	PRM001	PRM000
0	0	0	0	0	0	0/1	0/1

Selects count clock

图 6-44. 单脉冲输出操作的寄存器设置示例(2/2)

(e) 16 位定时器/计数器 00(TM00)

通过读取 TM00，获取计数值。

(f) 16 位定时器捕捉/比较寄存器 000(CR000)

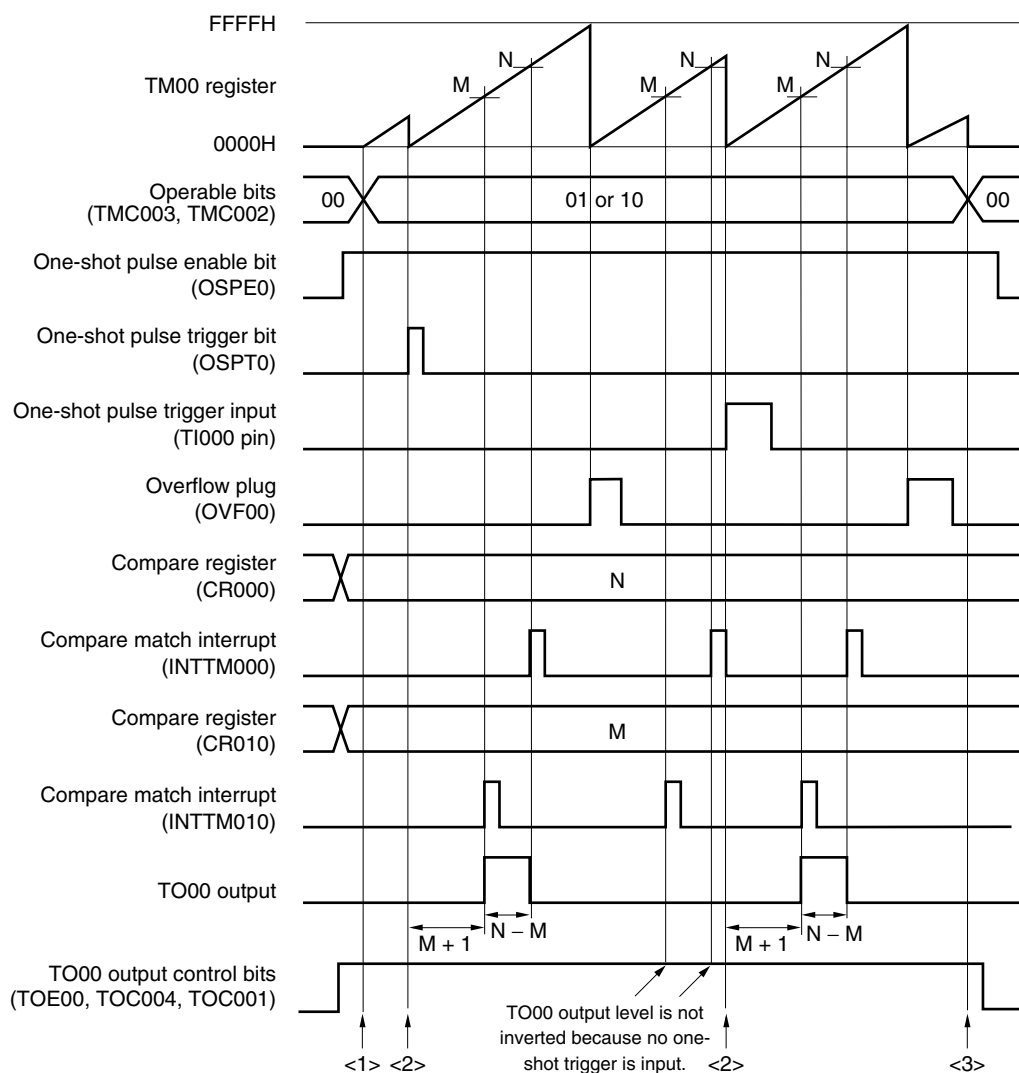
单脉冲输出时，该寄存器用作比较寄存器。TM00 的值与 CR000 的值相匹配时，产生匹配信号(INTTM000)，并反转 TO00 的输出电平。

(g) 16 位捕捉/比较寄存器 010(CR010)

单脉冲输出时，该寄存器用作比较寄存器。TM00 的值与 CR010 的值相匹配时，产生匹配信号(INTTM010)，并反转 TO00 的输出电平。

注意事项 禁止将 CR000 和 CR010 设置为相同的值。

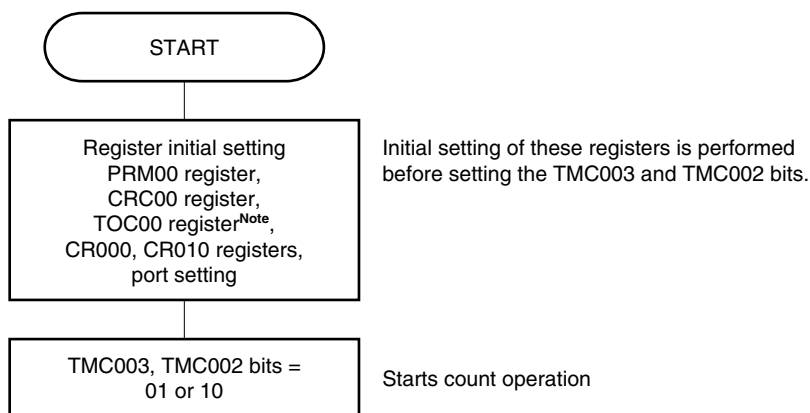
图 6-45. 单脉冲输出操作的软件处理示例(1/2)



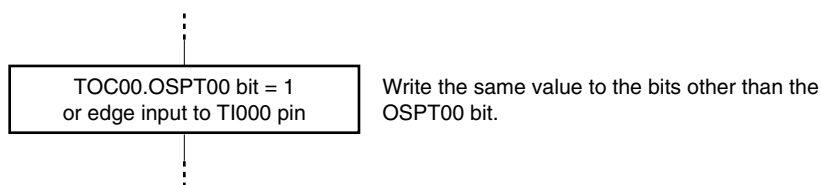
- 从输入单脉冲触发到输出单脉冲的时间
= $(M + 1) \times$ 计数时钟周期
- 单脉冲输出活动电平宽度
= $(N - M) \times$ 计数时钟周期

图 6-45. 单脉冲输出操作的软件处理示例(2/2)

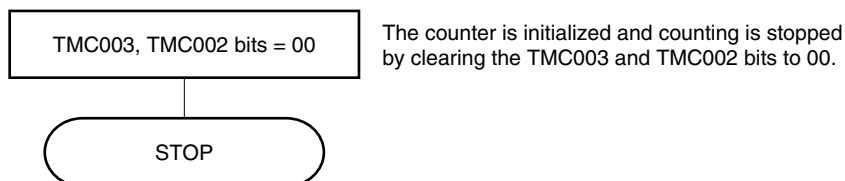
<1> Count operation start flow



<2> One-shot trigger input flow



<3> Count operation stop flow



注 设置 TOC00 时，应特别注意。参阅 6.3 (3) 16 位定时器输出控制寄存器 00(TOC00)。

6.4.8 脉冲宽度测量操作

TM00 可用来测量输入到 TI000 和 TI010 引脚的信号脉冲宽度。

在自由运行定时器模式下，可以通过操作 16 位定时器/时间计数器 0 或者 TI000 引脚输入信号的有效沿同步启动定时器，进行测量。

产生中断时，读取有效捕捉寄存器的有效值，并测量脉冲宽度。检测 16 位定时器模式控制寄存器 00(TMC00)第 0 位 (OVF00) 若该值置“1”，则同过软件将其清零。

图 6-46. 脉冲宽度测量(自由运行定时器模式)系统图

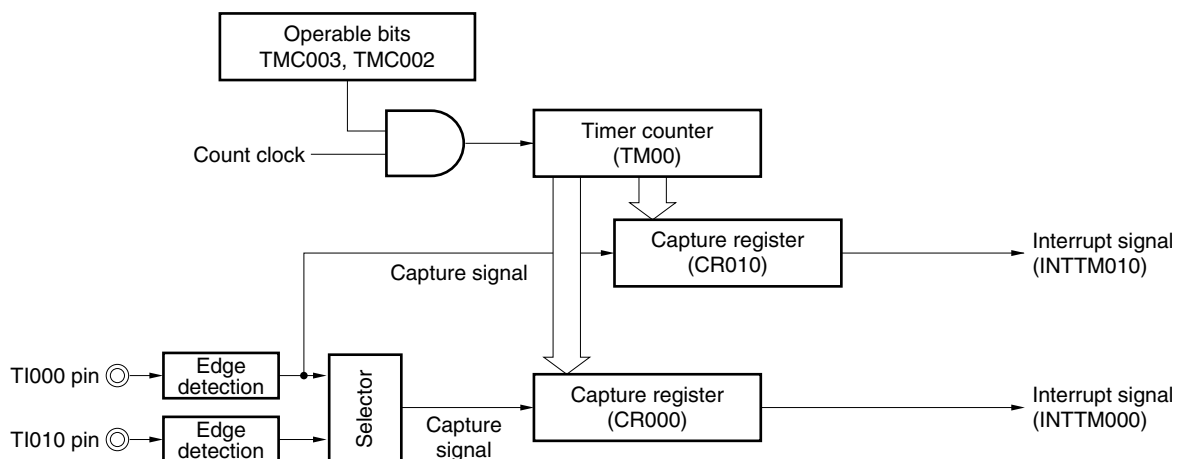
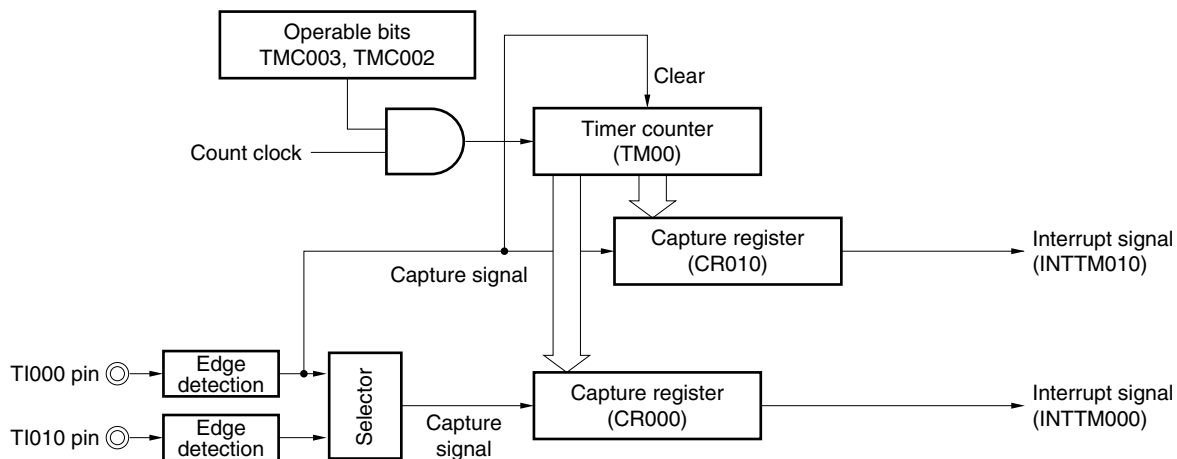


图 6-47. 脉冲宽度测量的框图
(通过输入 TI000 引脚有效沿，进入清零并启动模式)



通过以下三种方法测量脉冲宽度。

- 通过使用 TI000 与 TI010 引脚的两个输入信号测量脉冲宽度(自由运行定时器模式)
- 通过使用 TI000 引脚的输入信号测量脉冲宽度(自由运行定时器模式)
- 通过使用 TI000 引脚的输入信号测量脉冲宽度(通过 TI000 引脚的有效沿输入进入清零并启动模式))

- 备注 1.** 有关 I/O 引脚的设置, 参阅 6.3 (5) 端口模式寄存器 3(PM3)
- 2.** 有关如何允许 INTTM000 信号中断, 参阅第十四章 中断功能

(1) 通过使用 TI000 与 TI010 引脚的两个输入信号测量脉冲宽度(自由运行定时器模式)

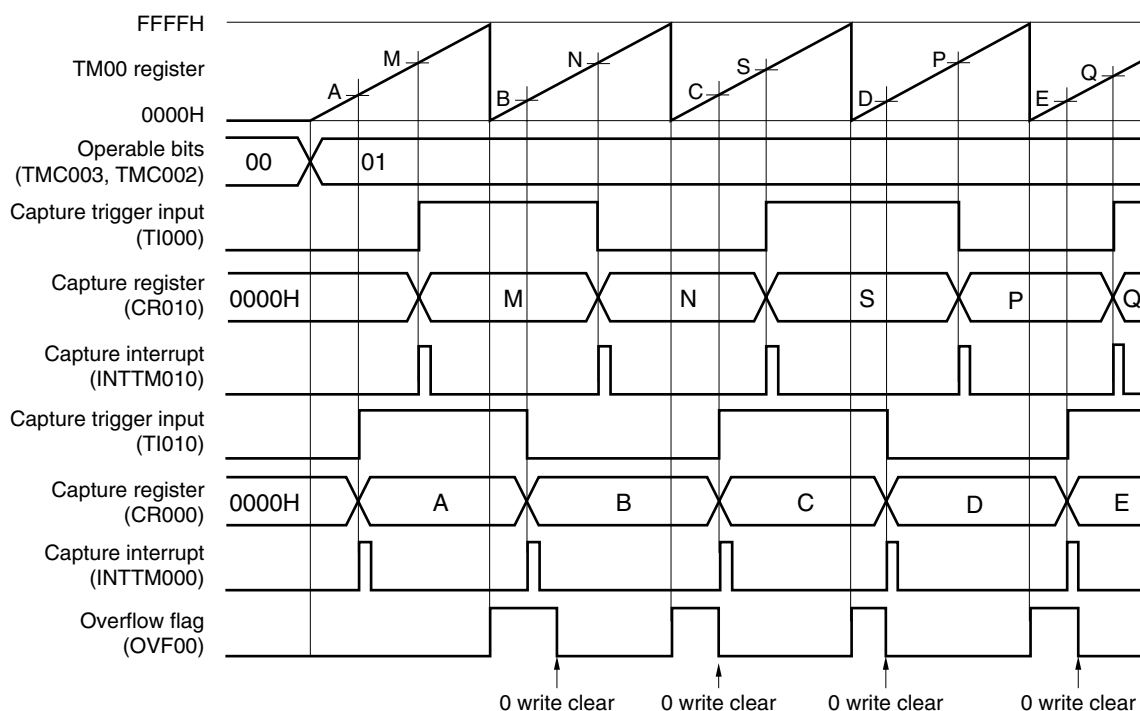
设置自由运行定时器模式(TMC003 和 TMC002 = 01)。检测出 TI000 引脚的有效沿时, 捕捉 TM00 的计数值并存入 CR010。检测出 TI010 引脚的有效沿时, 捕捉 TM00 的计数值并存入 CR000。指定检测 TI000 和 TI010 引脚的双边沿。

使用此方法, 从各输入信号的脉冲沿捕捉的计数值中减去预设值。因此, 事先将先前捕捉值存入其他的寄存器中。

产生溢出时, 若简单地从当前捕捉值减去先前捕捉值后的计数值为负, 则产生借位。因此, 将程序状态字(PSW)的第 0 位(CY)置“1”。若发生这种情况, 忽略 CY 并将计算值用作脉冲宽度。另外, 将检测 16 位定时器模式控制寄存器 00(TMC00)第 0 位(OVF00)清零。

图 6-48. 测量脉冲宽度的时序示例 (1)

• TMC00 = 04H, PRM00 = F0H, CRC00 = 05H



(2) 通过使用 TI000 引脚的输入信号测量脉冲宽度(自由运行定时器模式)

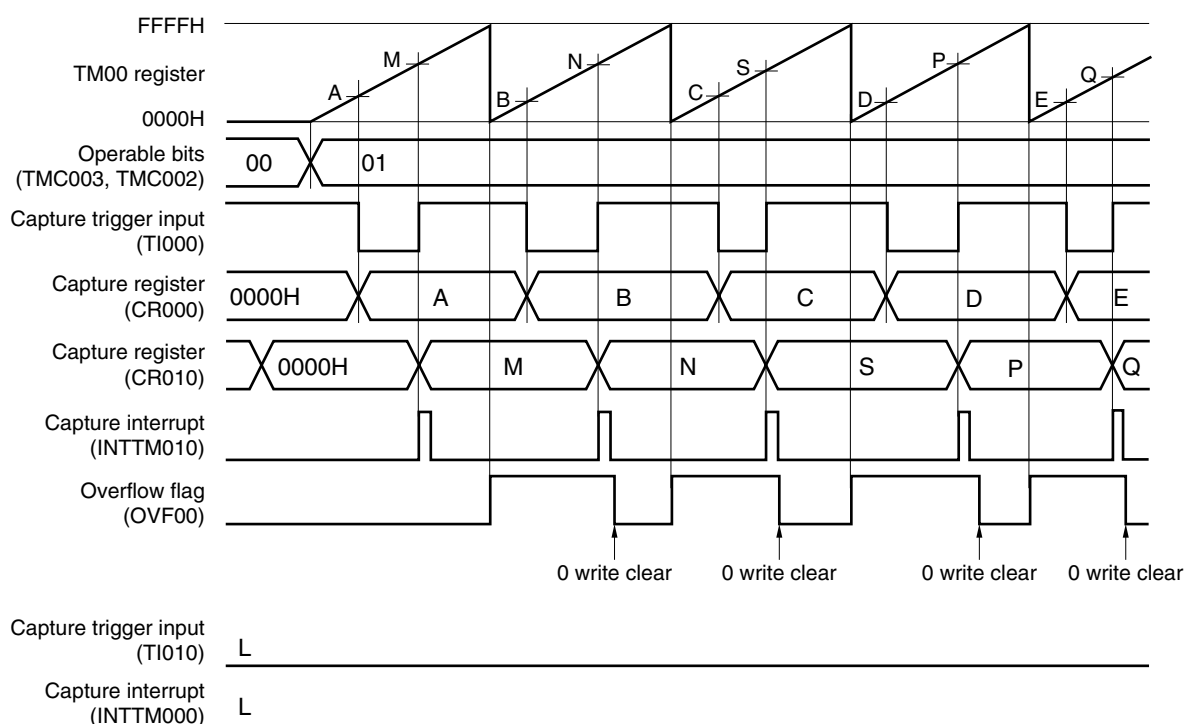
设置自由运行定时器模式(TMC003 和 TMC002 = 01)。TI000 引脚检测出反向沿时, 捕捉 TM00 的计数值并存入 CR00。检测出 TI000 引脚的有效沿时, 捕捉 TM00 的计数值并存入 CR010。

使用此方法, 可以测量从一个脉冲沿到另一个脉冲沿的宽度时, 计数值存入各自的捕捉计器。因此, 不必保存捕捉值。通过两个捕捉寄存器的计数值相减, 可以算出高电平宽度、低电平宽度和周期。

产生溢出时, 若只从当前捕捉值减去先前捕捉值后的计数值为负, 则产生借位。因此, 将程序状态字(PSW)的第 0 位(CY)置“1”。若发生这种情况, 忽略 CY 并将计算值用作脉冲宽度。另外, 将检测 16 位定时器模式控制寄存器 00(TMC00)第 0 位(OVF00)清零。

图 6-49. 测量脉冲宽度的时序示例 (2)

• TMC00 = 04H, PRM00 = 10H, CRC00 = 07H



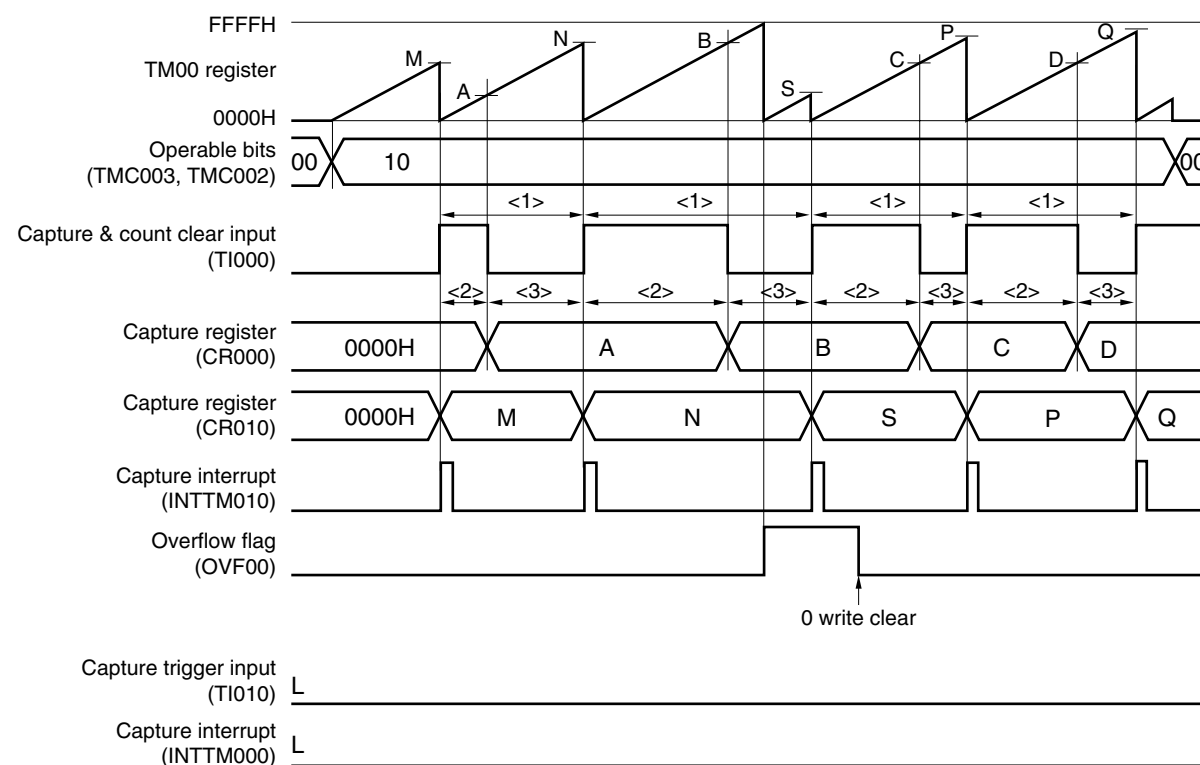
通过使用 TI000 引脚的输入信号测量脉冲宽度(通过输入 TI000 引脚的有效沿, 进入清零并启动模式))

通过 TI000 引脚的有效沿输入设置清零并启动模式(TMC003 和 TMC002 = 10)。在 TI000 引脚检测出反向的有效沿时, 捕捉 TM00 的计数值并存入 CR000; 检测出 TI000 引脚的有效沿时, 捕捉 TM00 的计数值捕捉并存入 CR010, TM00 清零(0000H)。因此, 若 TM00 没有溢出, 则在 CR010 中存入一个周期。

若产生溢出, 则将 CR010 的值加上 10000H, 其结果作为一个周期。16 位定时器模式控制寄存器 00(TMC00)第 0 位 (OVF00)清零。

图 6-50. 测量脉冲宽度的时序示例 (3)

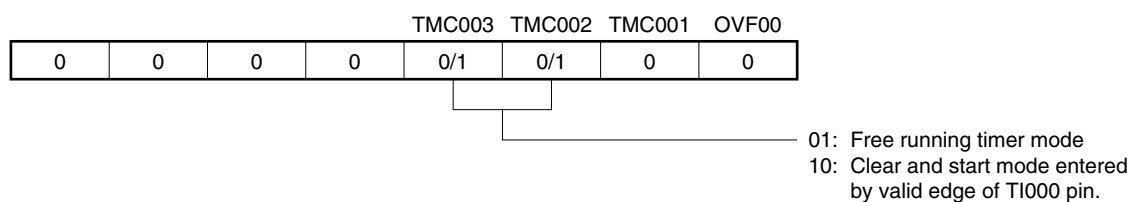
• TMC00 = 08H, PRM00 = 10H, CRC00 = 07H



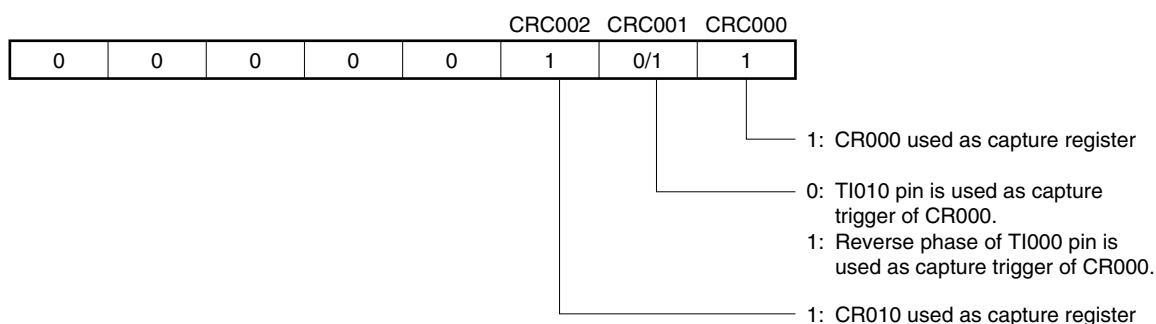
- <1> 脉冲周期 = $(10000H \times \text{OVF00 位被置“1”的次数} + \text{CR010 捕捉值}) \times \text{计数时钟周期}$
- <2> 高电平脉冲宽度 = $(10000H \times \text{OVF00 位被置“1”的次数} + \text{CR000 捕捉值}) \times \text{计数时钟周期}$
- <3> 低电平脉冲宽度 = $(\text{脉冲周期} - \text{高电平脉冲宽度})$

图 6-51. 脉冲宽度测量的寄存器设置示例 (1/2)

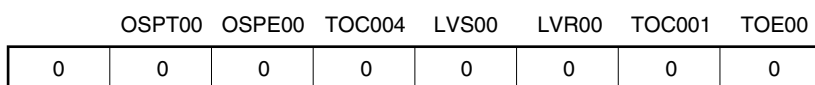
(a) 16 位定时器模式控制寄存器 00(TMC00)



(b) 捕捉/比较控制寄存器 00(CRC00)



(c) 16 位定时器输出控制寄存器 00(TOC00)



(d) 预分频模式寄存器 00(PRM00)

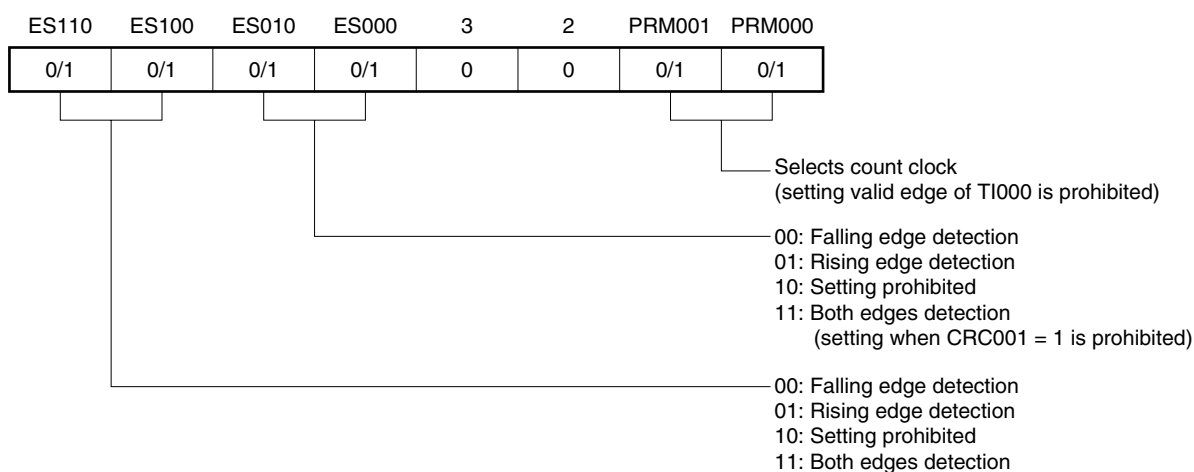


图 6-51. 脉冲宽度测量的寄存器设置示例 (2/2)

(e) 16 位定时器/计数器 00(TM00)

通过读取 TM00，获取计数值。

(f) 16 位定时器捕捉/比较寄存器 000(CR000)

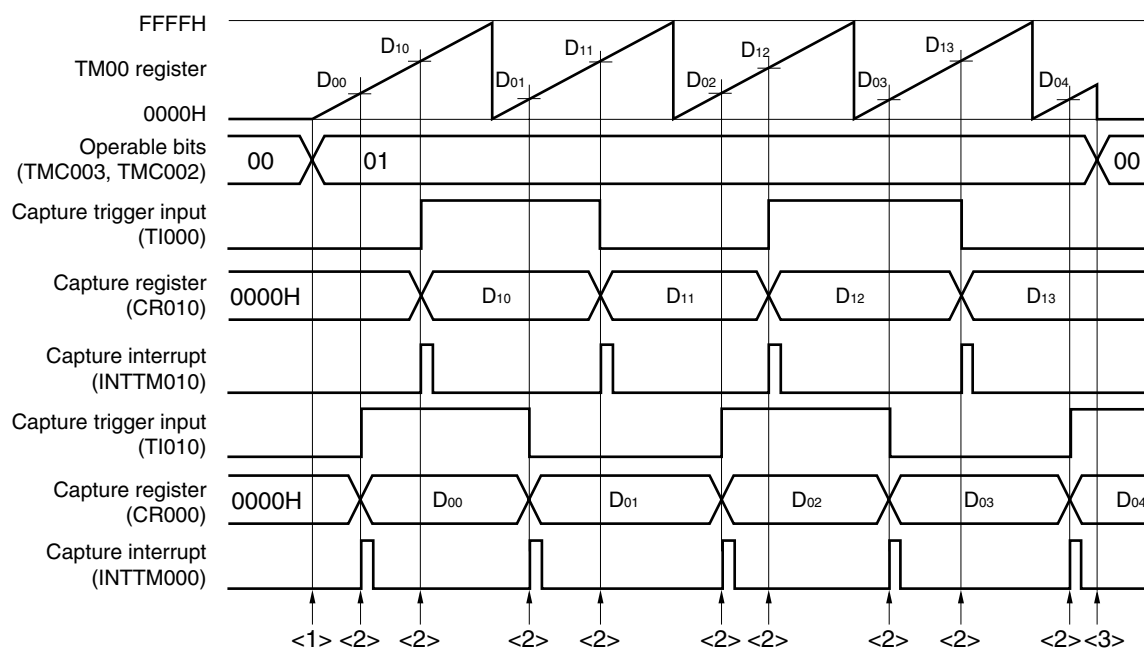
该寄存器用作捕捉寄存器选择 TI000 或 TI010 引脚用作捕捉触发。检测出捕捉触发的指定有效沿时，将 TM00 计数值存入 CR000。

(g) 16 位捕捉/比较寄存器 010(CR010)

该寄存器用作捕捉寄存器 TI000 引脚的输入信号用作捕捉触发。检测出捕捉触发时，将 TM00 计数值存入 CR010。

图 6-52. 测量脉冲宽度的软件处理示例 (1/2)

(a) 自由运行定时器模式示例



(b) 通过输入 TI000 引脚，进入清零并启动模式

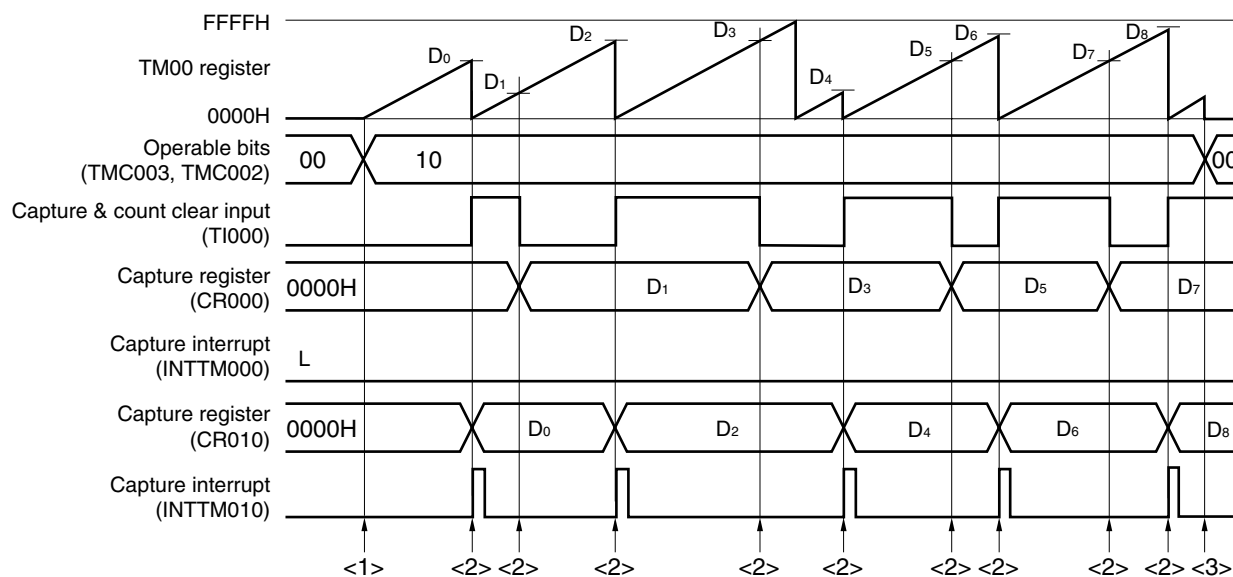
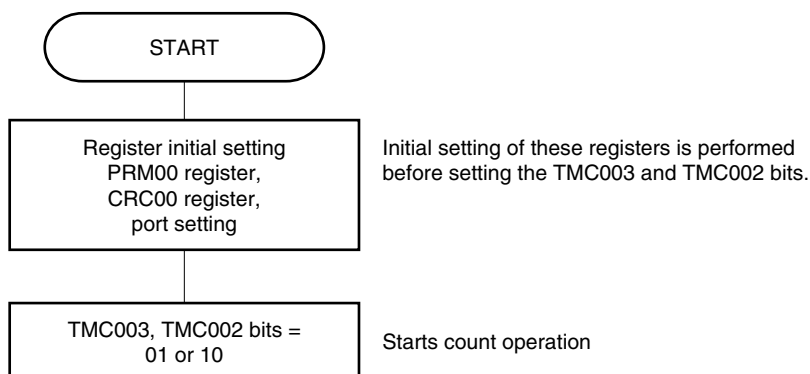
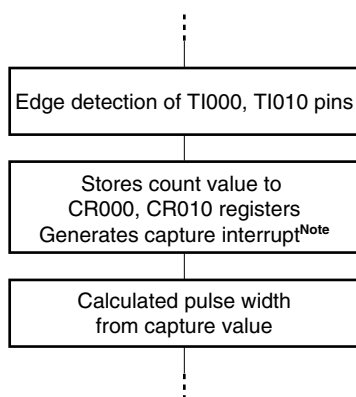


图 6-52. 测量脉冲宽度的软件处理示例 (2/2)

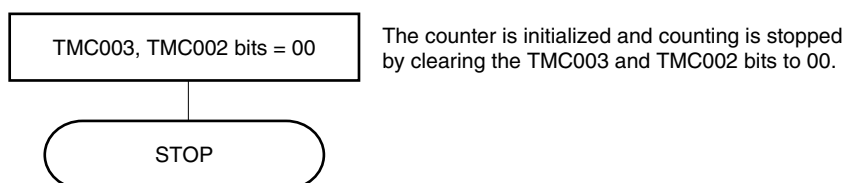
<1> Count operation start flow



<2> Capture trigger input flow



<3> Count operation stop flow



注 选择 TI000 引脚输入的反向沿用作 CR000 的有效沿时，不产生捕捉中断信号(INTTM000)。

6.5 TM00 特殊用途

6.5.1 TM00 操作期间 CR010 的重写

原则上，TM00 操作(TMC003 和 C002 = 00 除外的值)时，禁止将 CR000 和 CR010 用作比较寄存器进行重写。

但是，若仅将 CR010 用作 PPG 输出修改占空比时，则可以按以下步骤在 TM00 操作期间修改 CR010。（若 CR010 修改值小于当前值，则 CR010 与 00 相匹配后会立即重写。若 CR010 修改值大于当前值，则 CR00 与 00 相匹配后可以立即重写。若在 CR010 与 TM00 或 CR000 与 TM00 相匹配之前重写，则可能产生异常操作）。

修改 CR010 值的步骤：

- <1> 禁止中断 INTTM010(TMMK010 = 1)。
- <2> TM00 值与 CR010 值相匹配(TOC004 = 0)时，禁止反转定时器输出。
- <3> 修改 CR010 值。
- <4> 等待一个 TM00 计数时钟周期。
- <5> TM00 的值与 CR010 的值相匹配(TOC004 = 1)时，允许反转定时器输出。
- <6> INTTM010(TMIF010 = 0)中断标志清零。
- <7> 允许中断 INTTM010(TMMK010 = 0)。

备注 有关 TMIF010 和 TMMK010，参阅第十四章 中断功能。

6.5.2 LVS00 和 LVR00 的设置

(1) LVS00 和 LVR00 的设置

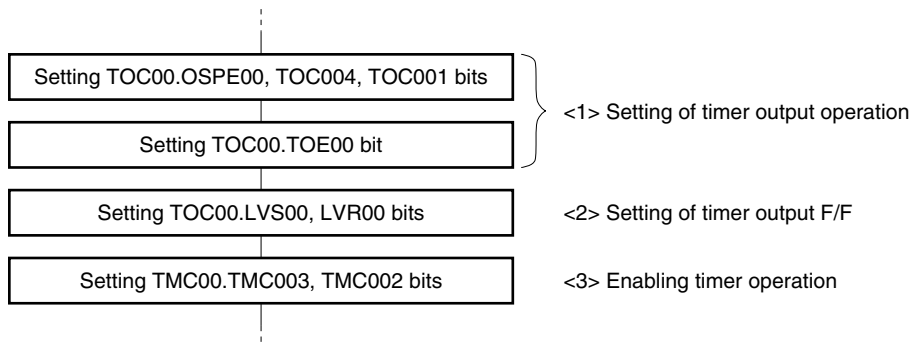
LVS00 和 LVR00 用于设置 TO00 引脚输出的默认值并可以在不需要定时器操作时(TMC003 与 TMC002 = 00)对定时器输出进行反转。不需要软件控制时，LVS00 和 LVR00 清零(00)(默认值：低电平输出)

LVS00	LVR00	定时器输出状态
0	0	无修改（低电平输出）
0	1	清除（低电平输出）
1	0	设置（高电平输出）
1	1	禁止设置

(2) 设置 LVS00 和 LVR00

按照以下步骤设置 LVS00 和 LVR00。

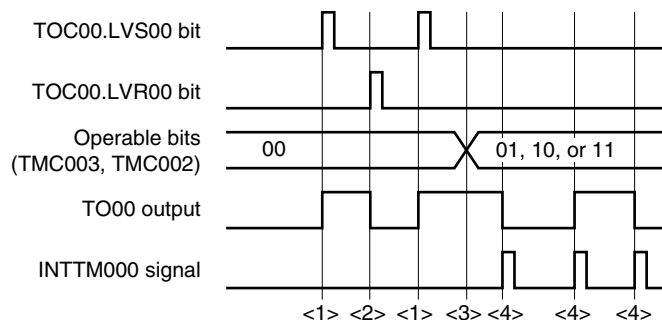
图 6-53. LVS00 和 LVR00 位的设置流程示例



注意事项 必须按上述步骤<1>、<2>和<3>设置 LVS00 和 LVR00。

可以在<1>之后、<3>之后，执行步骤<2>。

图 6-54. LVR00 和 LVS00 时序示例



- <1> LVS00 和 LVR00 = 10 时，TO00 引脚输出高电平。
- <2> LVS00 和 LVR00 = 01 时，TO00 引脚输出低电平(即使 LVS00 与 LVR00 清零(00)，引脚输出恒为高电平)。
- <3> TMC002 置“01”、“10”或“11”时，定时器开始操作。由于在开始操作前 LVS00 与 LVR00 置“10”，所以 TO00 引脚输出从高电平开始。定时器开始操作后。一直到 TMC003 和 TMC002 = 00，才可以设置 LVS00 和 LVR00 输出。
- <4> 每当产生中断信号(INTTM000)时，反转 TO00 引脚的输出电平。

6.6 16 位定时器/事件计数器 00 注意事项

(1) 16 位定时器/事件寄存器清零 00 各通道的限制条件

各通道的限制条件如表 6-3 所示。

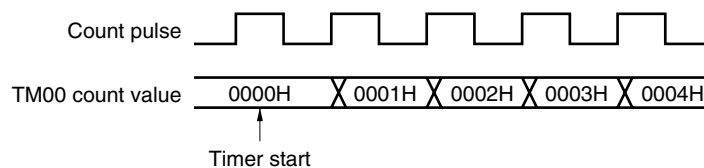
表 6-3. 16 位定时器/事件计数器 00 各通道的限制条件

操作	约束
用作间隔定时器	—
用作方波输出	
用作外部事件计数器	
通过TI000引脚有效沿输入进入清零并启动模式	检测出TI010引脚的有效沿时，禁止使用定时器输出(TO00)。(TOC00 = 00H)
用作自由运行定时器	—
用作PPG输出	$0000H \leq CP010 < CR000 \leq FFFFH$
单脉冲输出	禁止设置与CR000和CP010相同的值。
用作脉冲宽度测量	禁止使用定时器输出(TO00)(TOC00 = 00H)

(2) 定时器启动错误

定时器启动后，由于启动 TM00 计数操作与计数脉冲不同步，可能导致相匹配信号期间产生最长一个时钟的误差。

图 6-55. TM00 计数的启动时序



(3) CR000 和 CR010 的设置(TM00 与 CR000 相匹配时，进入清零并启动模式)

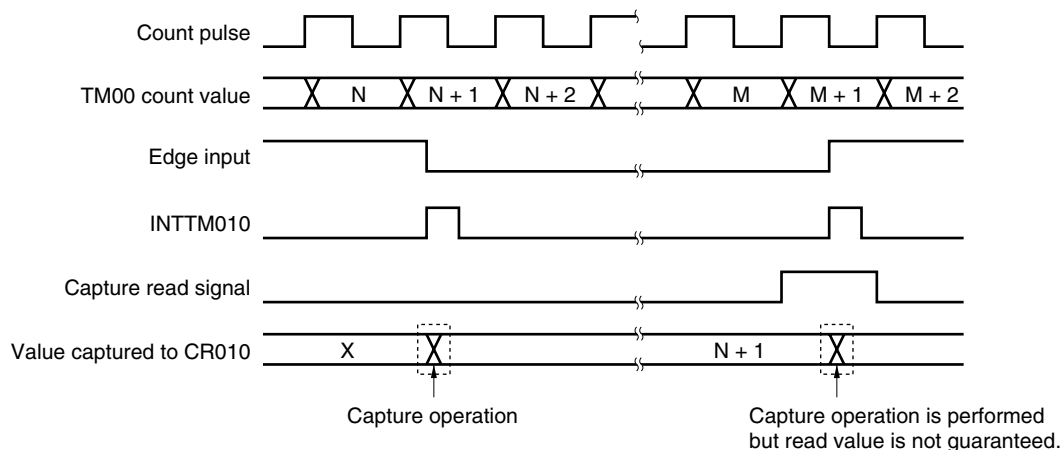
CR000 和 CR010 设置 0000H 除外的值(TM00 用作外部事件计数器时，不能进行单脉冲计数)。

(4) 捕捉寄存器的保持数据时序

- (a) 读取 CR000/CR010 期间，有效沿输入到 TI000/TI010 引脚并检测出 TI000 引脚的反向沿时，CR010 执行捕捉操作，但不能保证 CR000/CR010 的读取值。此时，若检测出 TI000/TI010 引脚的有效沿，则产生中断信号 (INTTM000/INTTM010)(若检测出 TI000 引脚的反向沿，则不产生中断信号)。

若因检测出 TI000/TI010 引脚的有效沿而捕捉到计数值，则应在产生 INTTM000/INTTM010 信号后读取 CR000/CR010 的值。

图 6-56. 捕捉寄存器的保持数据时序



- (b) 16 位定时器/事件计数器 00 停止后，不能确保 CR000 和 CR010 的值。

(5) 有效沿的设置

定时器停止(TMC003 和 TMC002 = 00)期间，使用 ES000 和 ES001 设置 TI000 引脚的有效沿。

(6) 单脉冲的重新触发

在单脉冲输出模式中输出有效电平期间，必须确保不产生触发。确保输出当前活动电平后，输入下一个触发。

(7) OVF00 标志的操作**(a) OVF00 标志的设置(1)**

在以下情况（TM00 溢出除外）下，OVF00 标志置“1”。

TM00 与 CR000 相匹配时，选择清零并启动模式。

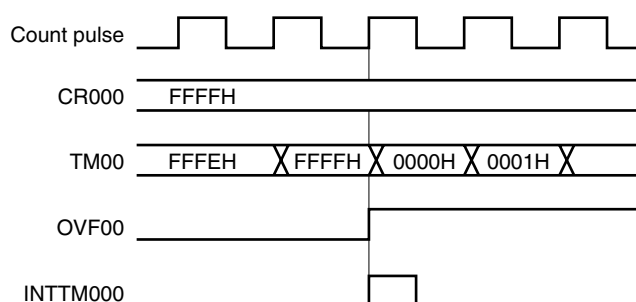
↓

设置 CR000 为 FFFFH。

↓

TM00 和 CR000 相匹配时，TM00 从“FFFFH”清为“0000H”。

图 6-57. OVF00 标志的操作时序

**(b) OVF00 标志清除**

即使 TM00 溢出后，下一个计数时钟被计数之前(TM00 的值变为 0001H 之前)，将 OVF00 标志清零(0)，OVF00 也会重新置“1”，并且清零无效。

(8) 单触发脉冲输出

在自由运行定时器模式或通过 TI000 引脚的有效沿输入进入的清零并启动模式下，可以进行正确的单脉冲输出操作。因 TM00 与 CR000 相匹配而进入的清零并启动模式下，不能输入单脉冲。

(9) 捕捉操作**(a) 指定 TI000 有效沿用作计数时钟**

指定 TI000 有效沿用作计数时钟时，指定用作 TI000 触发的捕捉寄存器不能正确操作。

(b) 通过 TI010 和 TI000 引脚的输入信号准确捕捉计数值的脉冲宽度

为了准确捕捉计数值，用作捕捉触发的 TI000 和 TI010 引脚的输入脉冲宽度必须大于两个计数时钟（用 PRM00 选择）（参阅图 6-7）。

(c) 中断信号产生

在计数时钟的下降沿执行捕捉操作，但中断信号(INTTM000 和 INTTM010)则产生在下一计数时钟的上升沿（参阅图 6-7）。

(d) CRC001 (捕捉/比较控制寄存器 00(CRC00)的第 1 位) 置“1”时的注意事项

若使用输入到 TI000 引脚的反向沿来捕捉 TM00 寄存器并放入 CR000 寄存器，则捕捉计数值后不产生中断请求信号(INTTM000)。若操作期间检测出 TI010 引脚的有效值，则不执行捕捉操作，但产生的 INTTM000 信号用作外部中断信号。若不使用外部中断，则屏蔽 INTTM000 信号。

(10) 脉冲沿检测**(a) 复位后指定有效沿**

若 TI000 或 TI010 引脚处于高电平,并且上升沿或双边沿指定为 TI000 或 TI010 引脚的有效沿时，允许 16 位定时器/事件计数器 00 的操作，则 TI000 或检测出的 TI010 引脚高电平为上升沿。上拉 TI000 和 TI010 引脚时要特别注意。但若定时器操作一旦停止后允许重新启动，则检测不出上升沿。

(b) 消除噪声用采样时钟

TI000 的有效沿用作计数时钟和用作捕捉触发时，所用于消除噪声的采样时钟有所不同。前者的采样时钟恒为 f_{PRS} 。后者选择 PRM00 用作计数时钟进行采样。

采样到 TI000 引脚的信号输出并且连续 2 次检测出有效电平时，才能判断出有效沿，从而消除窄脉冲宽度的噪声（参阅图 6-7）。

(11) 定时器操作

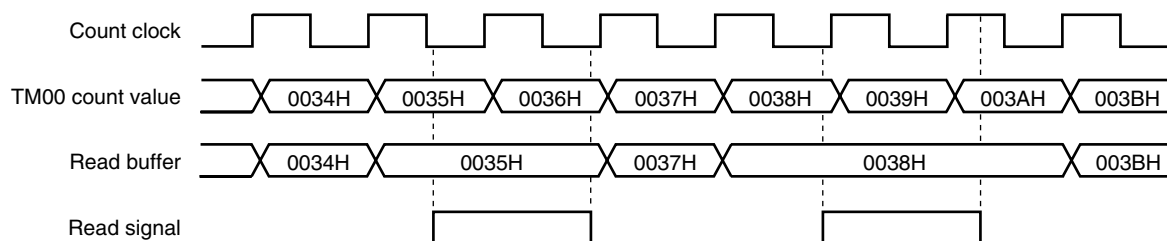
无论 CPU 模式处于何种操作模式，只要定时器停止，则不响应 TI000/TI010 引脚的输入信号。

备注 f_{PRS} :外部硬件时钟频率

(12) 16 位定时器/计数器 00(TM00)读取

因为 TM00 固定捕捉到缓冲器的计数值后才进行读取，所以在没有停止实际计数的情况下可以进行读取。但由于缓冲器是按计数器计数时序进行更新，因此，在计数器计数之前的读取值会出现没有更新的情况。

图 6-58. 16 位定时器/计数器 00(TM00)读取时间



第七章 8 位定时器/事件计数器 51

7.1 8 位定时器/事件计数器 51 功能

8 位定时器/事件计数器 51 具有以下功能。

- (1) 间隔定时器
- (2) 外部事件计数器
- (3) 方波输出
- (4) PWM 输出

备注 方波输出和 PWM 输出仅可用作 UART0 的基本时钟。

7.2 8 位定时器/事件计数器 51 配置

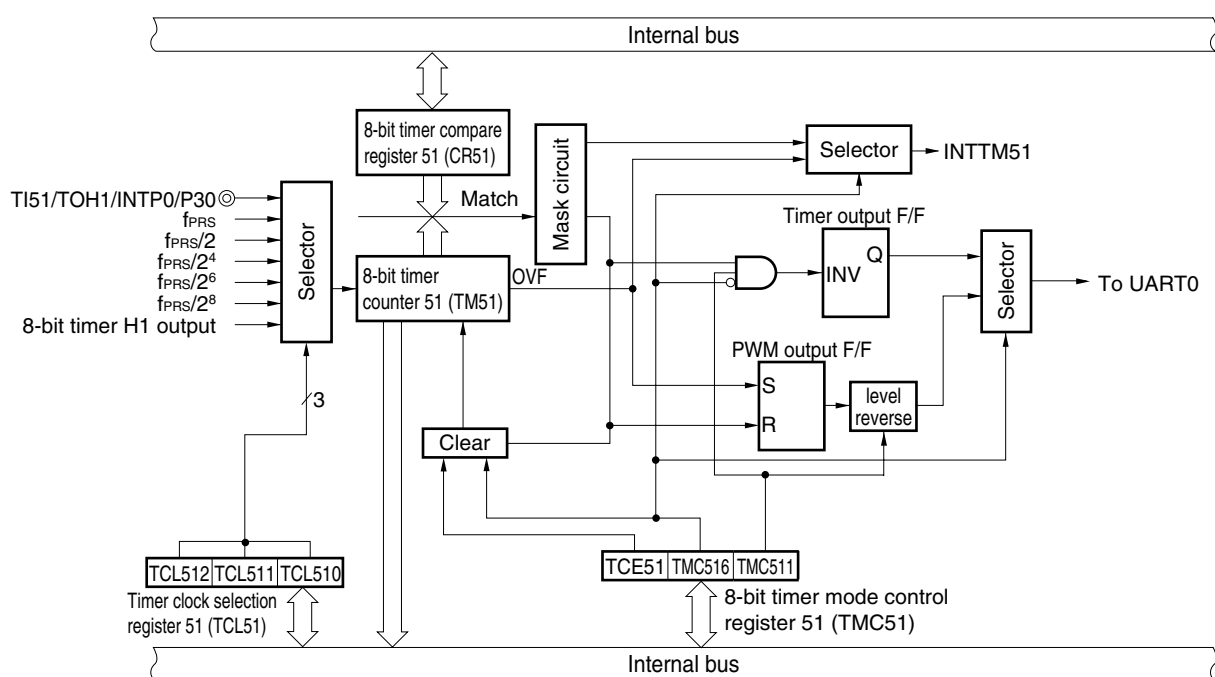
8 位定时器/事件计数器 51 包括以下硬件。

表 7-1. 8 位定时器/事件计数器 51 配置

项目	配置
定时器寄存器	8 位定时器/计数器 51 (TM51)
定时器输入	TI51
寄存器	8 位定时器比较寄存器 51 (CR51)
控制寄存器	定时器时钟选择寄存器 51(TCL51) 8 位定时器模式控制寄存器 51(TMC51) 端口模式寄存器 3(PM3) 端口寄存器 3(P3)

8 位定时器/事件计数器 51 的框图如图 7-1 所示。

图 7-1. 8 位定时器事件计数器 51 的框图



(1) 8 位定时器/计数器 51(TM51)

TM51 是 8 位只读寄存器，用于对计数脉冲进行计数。

计数器随计数时钟的上升沿同步递增。

图 7-2. 8 位定时器/计数器 51(TM51)的格式

地址：FF1FH 复位后：00H R

符号	7	6	5	4	3	2	1	0
TM51								

在以下情况下，计数值清零（00）。

<1> 产生复位信号

<2> TCE51 清零

<3> TM51 和 CR51 相匹配进入清零并启动模式时，TM51 和 CR51 相匹配。

(2) 8 位定时器比较寄存器 51(CR51)

可按 8 位存储器操作指令读写 CR51。

设置在 CR51 的值不断与 8 位定时器计数器的计数值进行比较，若两个值相匹配则产生中断请求信号(INTTM51)。

CR51 可以设置在 00H 至 FFH 之间。

复位信号产生时，CR51 清为 00H。

图 7-3. 8 位定时器比较寄存器 51(CR51)的格式

地址：FFB1H 复位后：00H R/W

符号	7	6	5	4	3	2	1	0
CR51								

注意事项 因 TM51 与 CR51 相匹配而进入清零并启动模式(TMC516 = 0) 时，操作中禁止写入 CR51 以外的值。

7.3 8 位定时器/事件计数器 51 控制寄存器

以下 4 个寄存器用来控制 8 位定时器/事件计数器 51。

- 定时器时钟选择寄存器 51(TCL51)
- 8 位定时器模式控制寄存器 51(TMC51)
- 端口模式寄存器 3(PM3)
- 端口寄存器 3 (P3)

(1) 定时器时钟选择寄存器 51 (TCL51)

该寄存器用来设置 8 为定时器/事件计数器 51 的计数时钟和 TI51 引脚输入的有效沿。

可以由 1 位或 8 位存储器操作指令设置 TCL51。

复位信号产生时，TCL51 清零(00H)。

图 7-4. 定时器时钟选择寄存器 51(TCL51)的格式

地址:	FFB2H	复位后:	00H	R/W								
符号	7	6	5	4	3	2	1	0				
TCL51	0	0	0	0	0	TCL512	TCL511	TCL510				

TCL512	TCL511	TCL510	计数时钟选择			
				f _{PRS} = 2 MHz	f _{PRS} = 5 MHz	f _{PRS} = 10 MHz
0	0	0	TI51 引脚下降沿 ^注			
0	0	1	TI51 引脚上升沿 ^注			
0	1	0	f _{PRS}	2 MHz	5 MHz	10 MHz
0	1	1	f _{PRS} /2	1 MHz	2.5 MHz	5 MHz
1	0	0	f _{PRS} /2 ⁴	125 kHz	312.5 kHz	625 kHz
1	0	1	f _{PRS} /2 ⁶	31.25 kHz	78.13 kHz	156.25 kHz
1	1	0	f _{PRS} /2 ⁸	7.81 kHz	19.53 kHz	39.06 kHz
1	1	1	TMH1 输出			

注 STOP 模式下，禁止使用从 TI51 引脚的外部时钟启动定时器操作。

注意事项 1. 若要向 TCL51 写入其它数据，则必须事先停止定时器操作。

2. 确保第 3 位至第 7 位清“0”。

备注 f_{PRS}: 外部硬件时钟频率

(2) 8 位定时器模式控制寄存器 51(TMC51)

TMC51 寄存器用来执行 8 位定时器 51(TM51)的计数操作控制。

可以由 1 位或 8 位存储器操作指令设置 TMC51。

复位信号产生时，该寄存器清为 00H。

图 7-5. 8 位定时器模式控制寄存器 51(TMC51)的格式

地址:	FFB3H	复位后:	00H	R/W						
符号	<7>	6	5	4	3	2	1	0		
TMC51	TCE51	TMC516	0	0	0	0	TMC511	0		
	TCE51	TM51 计数操作控制								
	0	清零后，禁止计数操作（计数器停止）								
	1	开始计数操作								
	TMC516	TM51 操作模式选择								
	0	TM51 和 CR51 相匹配时，进入清零并启动模式。								
	1	PWM(自由运行)模式								
	TMC511	其他模式(TMC516 = 0)				PWM 模式 (TMC516 = 1)				
		定时器 F/F 控制				有效电平选择				
	0	禁止反转操作				有效高电平				
	1	允许反转操作				有效低电平				

注意事项 1. 按照以下<1>至<2>的步骤进行设置，禁止同时设置。

<1> 设置 TMC511, TMC516: 工作模式设置

<2> 设置 TCE51

2. TCE51 = 1 时，禁止设置 TMC51 以外的位。

备注 PWM 模式下，通过 TCE 清零使 PWM 输出无效。

(4) 端口模式寄存器 3(PM3)

该寄存器按 1 位设置端口 3 的输入和输出。

可以由 1 位或 8 位存储器操作指令设置 PM3。

复位信号产生时，该寄存器设置为 FFH。

P30/TI51/TOH1/INTP0 引脚用作定时器输入时，PM30 置“1”。此时，P30 的输出锁存即可是“0”也可能是“1”。

图 7-6. 端口模式寄存器 3(PM3)的格式

地址:	FF23H	复位后:	FFH	R/W						
符号	7	6	5	4	3	2	1	0		
PM3	1	1	1	PM34	PM33	PM32	PM31	PM30		
	PM3n	P3n 引脚 I/O 模式选择 (n = 0 至 4)								
	0	输出模式（输出缓冲打开）								
	1	输入模式（输出缓冲关闭）								

7.4 8 位定时器/事件计数器 51 的操作

7.4.1 用作间隔定时器

以 8 位定时器比较寄存器 51(CR51)中预置的计数值作为间隔，用作产生反复中断要求的间隔定时器。

8 位定时器计数器 51(TM51)的计数值与 CR51 的值相匹配时，TM51 清零并继续计数，同时产生中断请求信号 (INTTM51)。

根据定时器时钟选择寄存器 51(TCL51)的第 0 位至第 2 位(TCL510 至 TCL512)，选择 TM51 的计数时钟。

设置

<1> 设置寄存器。

- TCL51: 选择计数时钟。
- CR51: 比较值
- TMC51: 停止计数操作，通过 TM51 和 CR51 相匹配，选择清零和启动模式。
(TMC51 = 0000xxx0B x = 忽略)

<2> 设置 TCE51 = 1 后，开始计数操作。

<3> 若 TM51 与 CR51 的值相匹配时，产生 INTTM51(TM51 清零(00H))。

<4> 以相同的时间间隔重复产生 INTTM51。

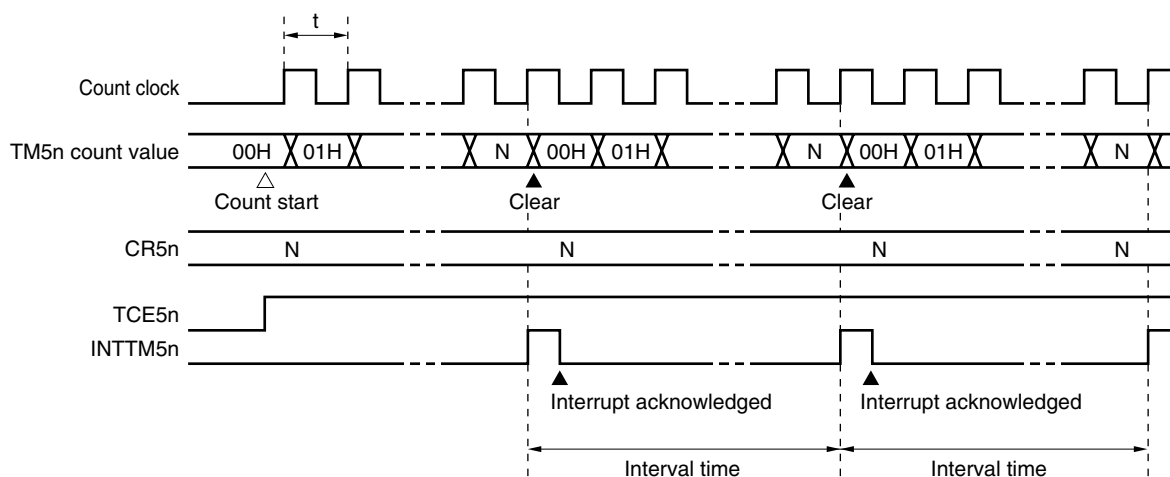
通过 TCE51 清零，停止计数操作。

注意事项 操作期间禁止向 CR51 输入其他值。

备注 有关如何允许 INTTM000 信号中断，参阅第十四章 中断功能。

图 7-7. 间隔定时器操作时序 (1/2)

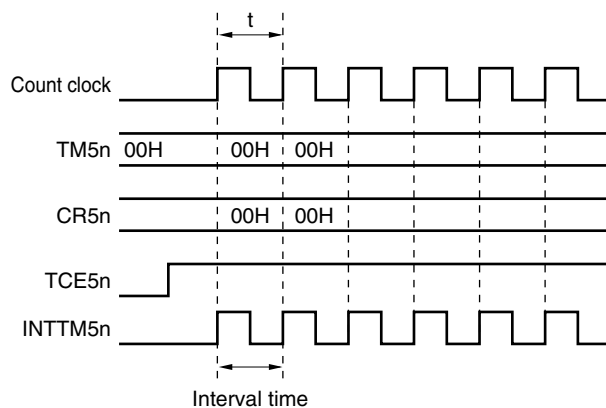
(a) 基本操作



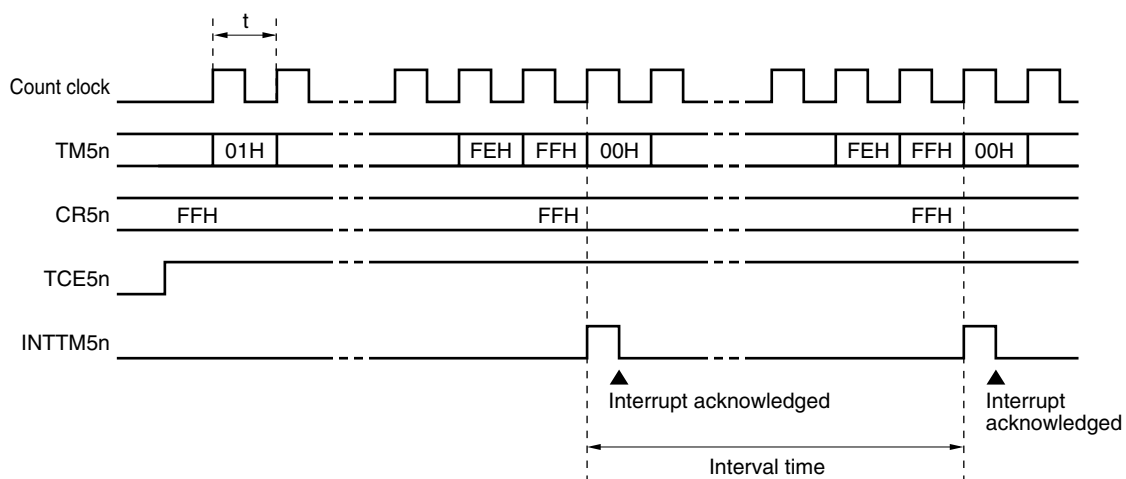
- 备注**
1. 间隔时间 = $(N + 1) \times t$, $N = 01H$ 至 FFH
 2. $n = 1$

图 7-7. 间隔定时器操作时序 (2/2)

(b) 当 CR51 = 00H 时



(c) 当 CR51 = FFH 时



备注 n = 1

7.4.2 用作外部事件计数器

通过 8 位定时器 51(TM51)，外部事件计数器对输入到 TI51 引脚的外部时钟脉冲进行计数。

由定时器时钟选择寄存器 51(TCL51)指定的有效沿输入时，TM51 随其输入次数递增。

可选上升沿或下降沿。

TM51 的计数值与 8 位定时器比较寄存器 51(CR51)的值相匹配时，TM51 清零，并产生中断请求信号(INTTM5n)。

随后，每当 TM51 的值与 CR51 的值相匹配时，将产生 INTTM51。

设置

<1> 设置各寄存器。

- 端口模式寄存器(PM30)置“1”。
- TCL51: 选择 TI51 引脚输入沿。
TI51 引脚下降沿 → TCL51 = 00H
TI51 引脚上升沿 → TCL51 = 01H
- CR51: 比较值
- TMC51: 停止计数，通过 TM51 与 CR51 相匹配选择清零和启动模式，禁止定时器 F/F 反转操作，禁止定时器输出。
(TMC51 = 00000000B)

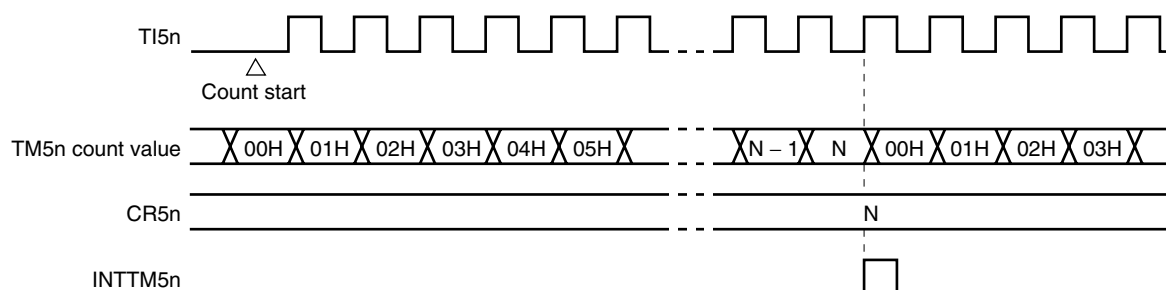
<2> 设置 TCE51 = 1 时，对 TI51 引脚输入的脉冲数进行计数。

<3> 若 TM51 与 CR51 的值相匹配时，产生 INTTM51(TM51 清零(00H))。

<4> 经以上设置，每当 TM51 的值与 CR51 的值相匹配时,将产生 INTTM51。

备注 有关如何允许 INTTM000 信号中断，参阅第十四章 中断功能。

图 7-8. 外部事件计数器操作（使用指定上升沿）



- 备注**
1. N = 00H 至 FFH
 2. n = 1

7.4.3 方波输出操作

以 8 位定时器比较寄存器 51(CR51)中预置的计数值为间隔，输出任意频率的方波。

通过 8 位定时器模式控制寄存器 51(TMC51)的第 0 位(TOE51)置“1”，在 CR51 的预置值决定的间隔时间内反转 TO51 引脚输出状态，可以输出任选频率的方波(占空比=50%)。

备注 方波输出仅可用作 UART0 的基本时钟。

设置方法

<1> 设置各寄存器。

- 端口输出锁存(P30)和端口模式寄存器(PM30)清零。
- TCL51 : 选择计数时钟。
- CR51 : 比较值
- TMC51 : 停止计数操作，通过 TM51 和 CR51 相匹配，选择清零和启动模式。
(TMC51 = 00000010B)

<2> 设置 TCE51 = 1 后，开始计数操作。

<3> 通过 TM51 和 CR51 相匹配，反转定时器输出 F/F。另外，AfINTTM51 产生后，TM51 清为 00H。

<4> 经以上设置，在相同间隔内反转定时器输出 F/F，并输出方波。

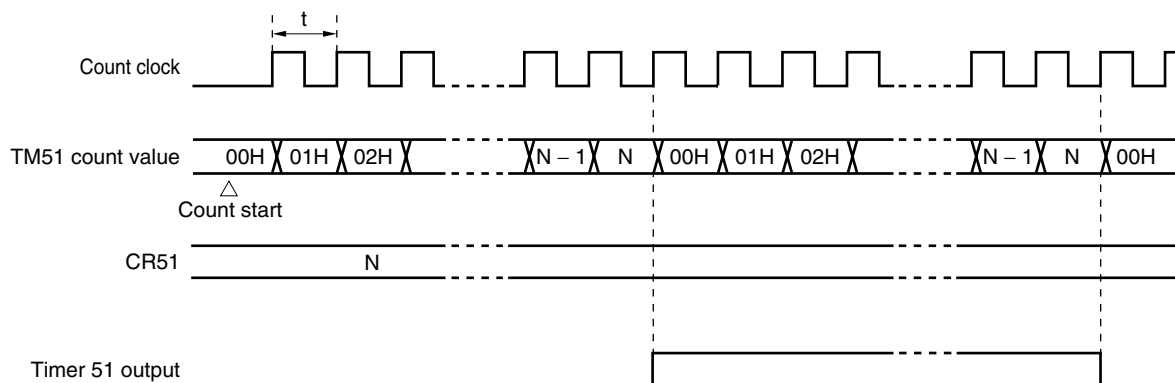
频率如下。

- 频率 = $1/2t(N + 1)$
(N: 00H 至 FFH)

注意事项 操作期间禁止向 CR51 输入其他值。

备注 有关 INTTM51 信号的中断允许，参阅第十四章 中断功能。

图 7-9. 方波输出操作的时序



7.4.4 PWM 输出操作

8 位定时器模式控制寄存器 51(TMC51)的第 6 位(TMC516)置“1”时，8 位定时器/事件计数器 51 用作 PWM 输出。输出由 8 位定时器比较寄存器 51(CR51)设定值决定的占空比脉冲。

设置 CR51 为 PWM 脉冲的有效电平宽度；可通过 TMC51 的第 1 位(TMC511)选择有效电平。

根据定时器时钟选择寄存器 51(TCL51)的第 0 位至第 2 位(TCL510 至 TCL512)，选择计数时钟。

注意事项 PWM 模式下，重写 CR51 周期应在 3 个计数时钟(由 TCL51 选择时钟)以上。

备注 PWM 输出仅可用作 UART0 的基本时钟。

(1) PWM 输出基本操作**设置**

<1> 设置各寄存器。

- 端口输出锁存(P30)和端口模式寄存器(PM30)清零。
- TCL51: 选择计数时钟。
- CR51: 比较值
- TMC51: 停止计数操作, 选择 PWM 模式。

TMC511	有效电平选择
0	有效高电平
1	有效低电平

(TMC51 = 01000000B 或 01000010B)

<2> TCE51 = 1 时, 开始计数操作。

通过 TCE51 清零, 停止计数操作。

PWM 输出操作

<1> PWM 输出(TO51 输出)一直输出非有效电平, 直到产生溢出为止。

<2> 产生溢出时, 输出有效电平。有效电平一直输出, 直至 CR51 与 8 位定时器计数器 51 的计数值相匹配为止。

<3> 若 CR51 与计数值相匹配, 则输出非有效电平直至再次发生溢出。

<4> 重复<2>和<3>操作, 直至计数停止。

<5> TCE51 = 0 停止计数时, PWM 开始输出非有效电平。

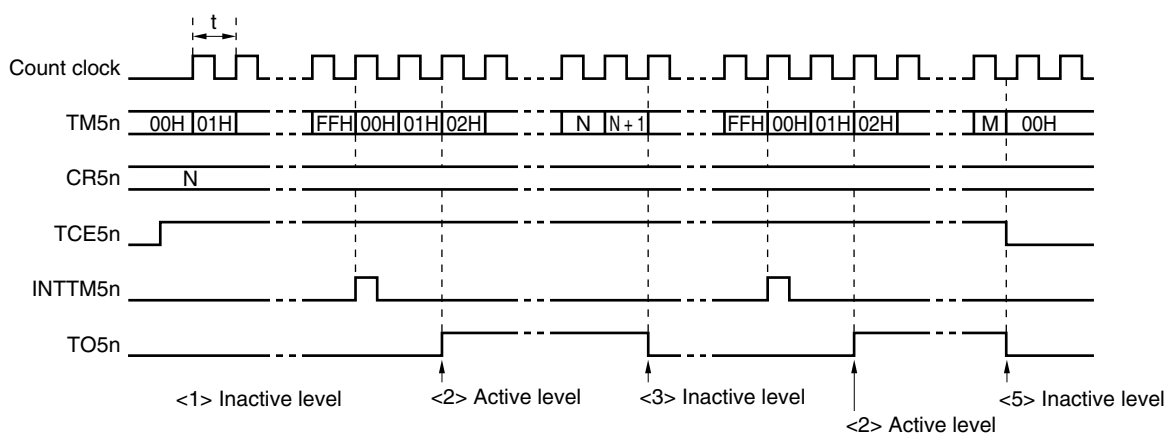
有关详细的时序, 参阅图 7-10 和 7-11。

周期、有效电平宽度和占空比如下。

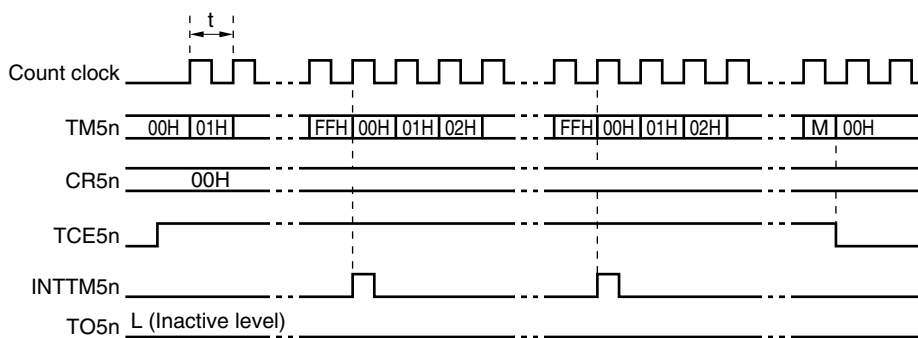
- 周期 = $2^8 t$
 - 有效电平宽度 = Nt
 - 占空比 = $N/2^8$
- (N = 00H 至 FFH)

图 7-10. PWM 输出操作的时序

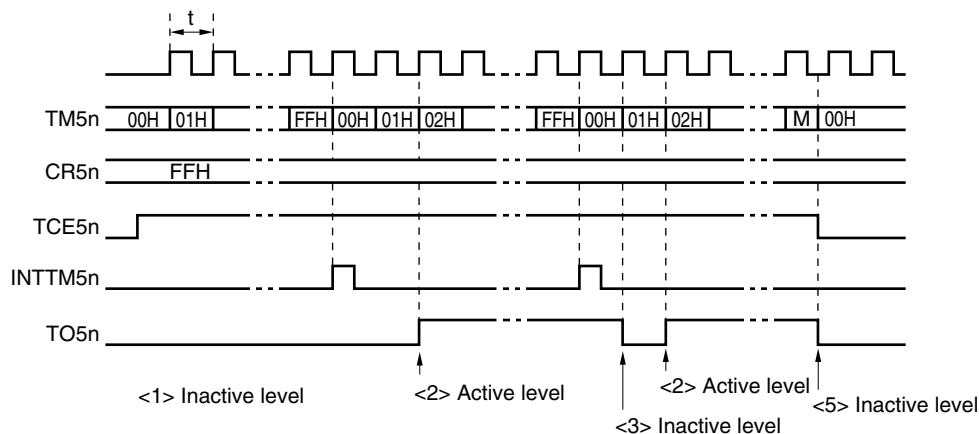
(a) 基本操作 (有效电平 = H)



(b) CR5n = 00H



(c) CR5n = FFH

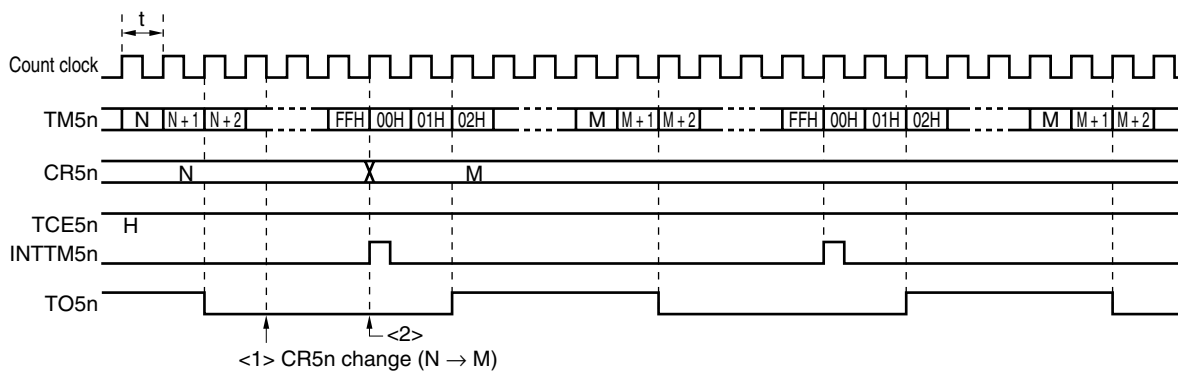


- 备注 1. 图 8-14 (a)和(c)中的<1>至<3>和<5>与 7.4.4 (1) PWM 输出基本操作的 PWM 输出操作中<1>至<3>和<5>相对应。
2. n = 1

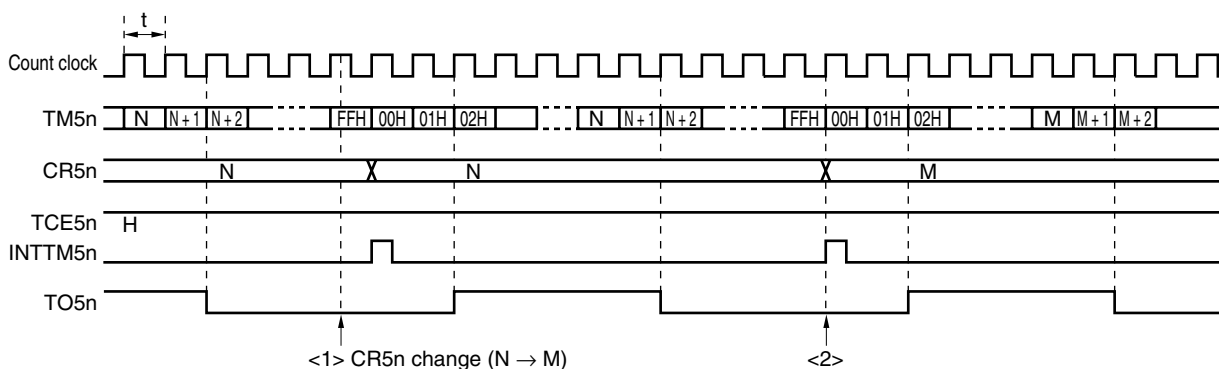
(2) 改变 CR51 的操作

图 7-11. 改变 CR51 的操作时序

(a) FFH 的时钟上升沿出现之前，CR51 值从 N 改变为 M 的情况下，
→改变后，出现溢出时立即将值传送到 CR51。



(b) FFH 的时钟上升沿出现之前，CR51 值从 N 改变为 M 的情况下，
→改变后，产生溢出时立即将值传送到 CR51。



注意事项 图 8-15<1>与<2>之间读取 CR51 时，读取的值与实际值不同(读取值:M,CR51 实际值:N)。

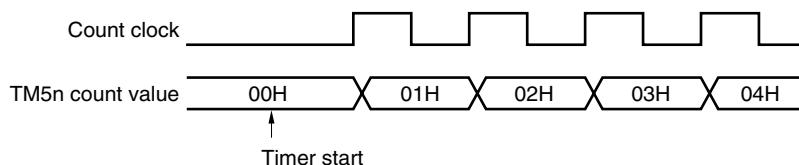
备注 n = 1

7.5 8 位定时器/事件计数器 51 注意事项

(1) 定时器启动错误

定时器启动后，产生相匹配信号期间可能发生最长一个时钟的误差。这是因为计数时钟与 8 位定时器计数器 51(TM51) 没有同步启动而导致的。

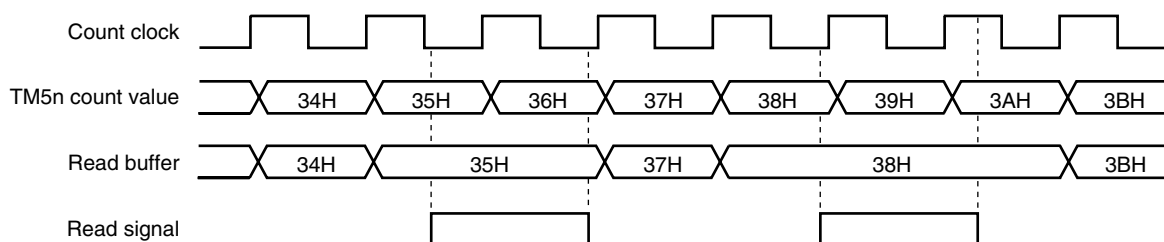
图 7-12. 8 位定时器计数器 51(TM51)读取时序



(2) 8 位定时器计数器 51(TM51)的读取

因为将捕捉到缓冲器的计数值固定后读取，所以 TM51 在没有停止实际计数的情况下可以进行读取。但是，缓冲器由计数器计数时序进行更新，所以在计数器计数之前读取的值会出现没有更新的情况。

图 7-13. 8 位定时器/计数器 51(TM51)读取时间



备注 n = 1

第八章 8 位定时器 H1

8.1 8 位定时器 H1 功能

8 位定时器 H1 包含以下功能。

- 间隔定时器
- 方波输出
- PWM 输出
- 载波发生器

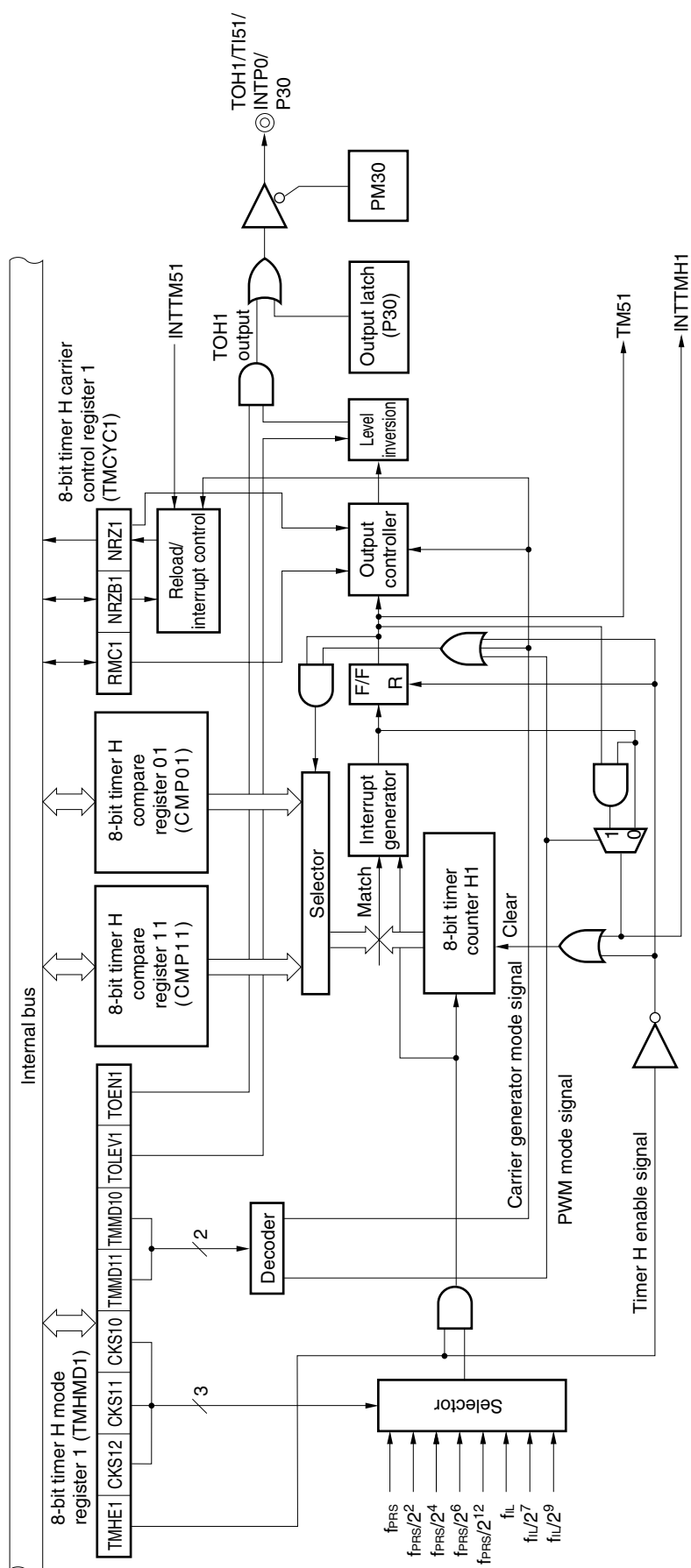
8.2 8 位定时器 H1 配置

8 位定时器 H1 包含下列硬件。

表 8-1. 8 位定时器 H1 配置

项目	配置
定时器寄存器	8 位定时器 H1
寄存器	8 位定时器 H 比较寄存器 01(CMP01) 8 位定时器 H 比较寄存器 11(CMP11)
定时器输出	TOH1, 输出控制器
控制寄存器	8 位定时器 H 模式寄存器 n(TMMD1) 8 位定时器 H 载波控制寄存器 1(TMCYC1) 端口模式寄存器 3(PM3) 端口寄存器 3(P3)

图 8-1.8 位定时器 H1 的框图



(1) 8 位定时器 H 比较寄存器 01(CMP01)

可用 8 位存储器操作指令写入该寄存器。该寄存器适用于所有定时器操作模式。

该寄存器不断地将 CMP01 的值和 8 位定时器计数器 H1 的计数值进比较，两个值相匹配时，产生中断请求信号 (INTTMH1) 并反转 TOH1 的输出电平。

当定时器停止(TMHE1 = 0)时，重写 CR01 的值。

复位信号产生时，该寄存器清为 00H。

图 8-2. 8 位定时器 H 比较寄存器 01(CMP01)的格式

地址: FF1AH 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
CMP01								

注意事项 定时器计数操作期间，不能写入 CMP01。定时器计数操作期间，可以刷新 CMP01(写入相同的值)

(2) 8 位定时器 H 比较寄存器 11(CMP11)

可用 8 位存储器操作指令写入该寄存器。该寄存器用作 PWM 输出模式和载波发生器模式。

PWM 输出模式下，该寄存器不断地将 CMP11 的值和 8 位定时器计数器 H1 的计数值进比较，两个值相匹配时，反转 TOH1 的输出电平。产生不中断请求信号。

载波发生器模式下，CMP11 寄存器不断地将 CMP11 的值和 8 位定时器计数器 H1 的计数值进比较，两个值相匹配时，产生中断请求信号(INTTMH1)。同时，计数值清零。

定时器计数操作期间，可以刷新 CMP11(写入相同的值)并重写。

若在定时器操作期间重写 CMP11 的值，则当定时器当前值与 CMP11 以前值相匹配时，锁存新值并传送给 CMP11,然后用新值取代 CMP 旧值。若计数值与 CMP11 的值相匹配操作与 CMP 的写操作冲突时，则 CMP11 的值保持不变。

复位信号产生时，该寄存器清为 00H。

图 8-3. 8 位定时器 H 比较寄存器 11(CMP11)的格式

地址: FF1BH 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
CMP11								

注意事项 PWM 输出模式和载波发生器模式下，设置定时器计数停止 (TMHE1 = 0)后，启动定时器计数操作(TMHE1 = 1) 时，必须设置 CMP11 (CMP11) (即使 CMP11 的设定值相同，也必须重新设置)。

8.3 寄存器控制 8 位定时器 H1

以下 4 个寄存器用来控制 8 位定时器 H1。

- 8 位定时器 H 模式寄存器 1(TMhMD1)
- 8 位定时器H载波控制寄存器 1(TMCYC1)
- 端口模式寄存器 3(PM3)
- 端口寄存器 3(P3)

(1) 8 位定时器 H 模式寄存器 n(TMhMD1)

该寄存器控制定时器 H 的模式。

可用 1 位或 8 位存储器操作指令设置该寄存器。

复位信号产生时，该寄存器清为 00H。

图 8-4. 8 位定时器 H 模式寄存器 1(TMHMD1)的格式

地址: FF6CH 复位后: 00H R/W

	<7>	6	5	4	3	2	<1>	<0>
TMHMD1	TMHE1	CKS12	CKS11	CKS10	TMMD11	TMMD10	TOLEV1	TOEN1

TMHE1	允许定时器操作
0	停止定时器计数操作 (计数器清零)
1	允许定时器计数操作 (通过输入时钟开始计数操作)

CKS12	CKS11	CKS10	计数时钟选择			
				f _{PRS} = 2 MHz	f _{PRS} = 5 MHz	f _{PRS} = 10 MHz
0	0	0	f _{PRS}	2 MHz	5 MHz	10 MHz
0	0	1	f _{PRS} /2 ²	500 kHz	1.25 MHz	2.5 MHz
0	1	0	f _{PRS} /2 ⁴	125 kHz	312.5 kHz	625 kHz
0	1	1	f _{PRS} /2 ⁶	31.25 kHz	78.13 kHz	156.25 kHz
1	0	0	f _{PRS} /2 ¹²	0.49 kHz	1.22 kHz	2.44 kHz
1	0	1	f _{IL} /2 ⁷	1.88 kHz (典型值)		
1	1	0	f _{IL} /2 ⁹	0.47 kHz (典型值)		
1	1	1	f _{IL}	240 kHz (典型值)		

TMMD11	TMMD10	定时器操作模式
0	0	间隔定时器模式
0	1	载波发生器模式
1	0	PWM输出模式
1	1	禁止设置

TOLEV1	定时器输出电平控制 (默认状态)
0	低电平
1	高电平

TOEN1	定时器输出控制
0	禁止输出
1	允许输出

- 注意事项**
1. **TMHE1 = 1** 时,不能设置 **TMHMD** 以外的位。但是,可刷新 **TMHMD1** (写入相同值)
 2. **PWM** 输出模式和载波发生器模式下, 设置定时器计数停止(**TMHE1 = 1**)后, 启动定时器计数操作时, 必须设定 **8** 位定时器 **H** 比较寄存器 **11 (CMP11)** (即使 **CMP11** 的设定值相同, 也必须重新设置)。
 3. 使用载波发生器模式时, 将 **TMH1** 的计数时钟频率设置到 **TM51** 计数时钟频率的 **8** 倍以上。
 4. 除 **TOH1** 输出之外, 实际的 **TOH1/TI51/INTP0/P30** 引脚输出取决于 **PM30** 和 **P30**。

- 备注**
1. **fPRS**: 外部硬件时钟频率
 2. **fIL**: 内部低速振荡时钟频率

(2) 8 位定时器 H 载波控制寄存器 1(TMCYC1)

该寄存器控制 **8** 位定时器 **H1** 的远控输出和载波脉冲输出状态。

可用 **1** 位或 **8** 位存储器操作指令设置该寄存器。

复位信号产生时, 该寄存器清为 **00H**。

图 8-5. 8 位定时器 H 载波控制寄存器 1(TMCYC1)的格式

地址: **FF71H** 复位后: **00H** R/W^{Note}

	7	6	5	4	3	2	1	<0>
TMCYC1	0	0	0	0	0	RMC1	NRZB1	NRZ1

RMC1	NRZB1	远控输出
0	0	低电平输出
0	1	INTTM51信号输入上升沿的高电平输出
1	0	低电平输出
1	1	INTTM51信号输入上升沿的载波脉冲输出

NRZ1	载波脉冲输出状态标志
0	载波输出禁止状态(低电平状态)
1	载波输出允许状态 (RMC1 = 1: 载波脉冲输出, RMC1 = 0: 高电平状态)

备注 第 **0** 位只读。

注意事项 **TMHE = 1** 时, 禁止重写 **RMC1**。但是, 可刷新 **TMCYC1** (写入相同值)。

(3) 端口模式寄存器 3(PM3)

该寄存器按 1 位设置端口 3 的输入和输出。

P30/TOH1/TI51/INTP0 引脚用作定时器输出时，PM30 和P30 的输出锁存清零。

可用 1 位或 8 位存储器操作指令设置 PM3。

复位信号产生时，该寄存器设置为 FFH。

图 8-6. 端口模式寄存器 3(PM3)的格式

地址: FF23H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PM3	1	1	1	PM34	PM33	PM32	PM31	PM30

PM3n	P3n 引脚 I/O 模式选择(n = 0 至 4)
0	输出模式（输出缓冲打开）
1	输入模式（输出缓冲关闭）

8.4 8 位定时器 H1 操作

8.4.1 用作间隔定时器/方波输出

8 位定时器计数器 H1 和比较寄存器 01(CMP01)相匹配时，产生中断请求信号(INTTMH1)，同时 8 位定时器计数器 H1 清为 00H。

通常在间隔定时器模式下不能使用比较寄存器 11(CMP11)。因此，即使设置 CMP11 寄存器，也不能检测出 8 位定时器计数器 H1 和 CMP11 寄存器相匹配，因此，不影响定时器输出。

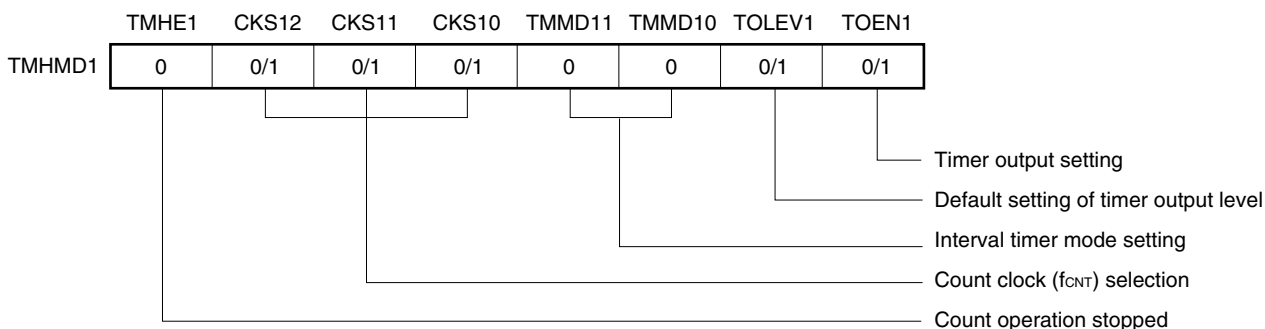
通过设置定时器 H 模式寄存器(TMHMD1)的第 0 位(TOEN1)置“1”，从 TOH1 输出任意形式频率（占空比 50%）的巨型波。

设置

<1> 设置各寄存器。

图 8-7. 在间隔定时器/方波输出操作期间设置寄存器

(i) 设置定时器 H 模式寄存器 1(TMHMD1)



(ii) CMP01 寄存器设置

若 N 设置为比较值时，则间隔时间如下所示。

- 间隔时间 = (N + 1)/fCNT

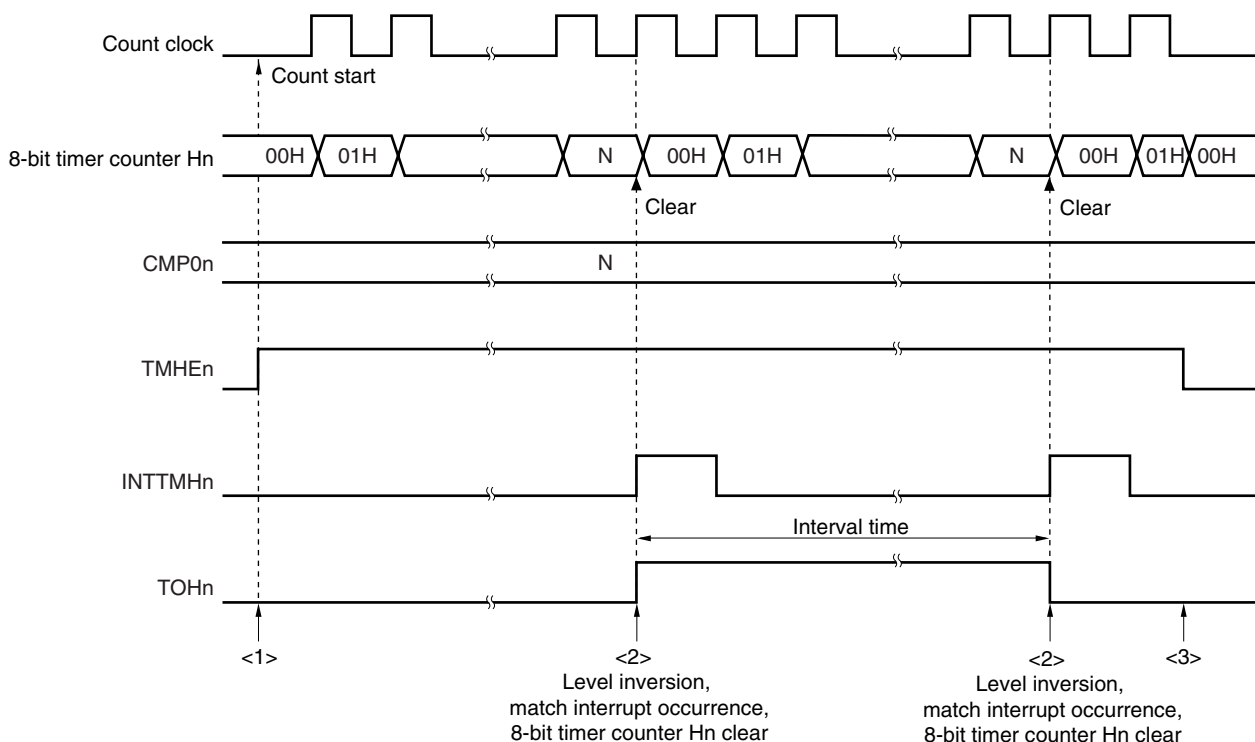
<2> TMHE1 = 1 时，启动计数操作。

<3> 8 位定时器计数器 H1 和 CMP01 寄存器的值相匹配时，产生 INTTMH1 信号，同时 8 位定时器计数器 H1 清为 00H。

<4> 随后,以相同间隔时间重复产生 INTTMH1 信号。若计数操作停止，则 TMHE1 清零。

- 备注 1. 有关输出引脚的设置，参阅 8.3 (3) 端口模式寄存器 3(PM3)。
2. 有关如何允许 INTTMH1 信号中断，参阅第十四章 中断功能。

图 8-8. 间隔定时器/方波输出操作的时序 (1/2)

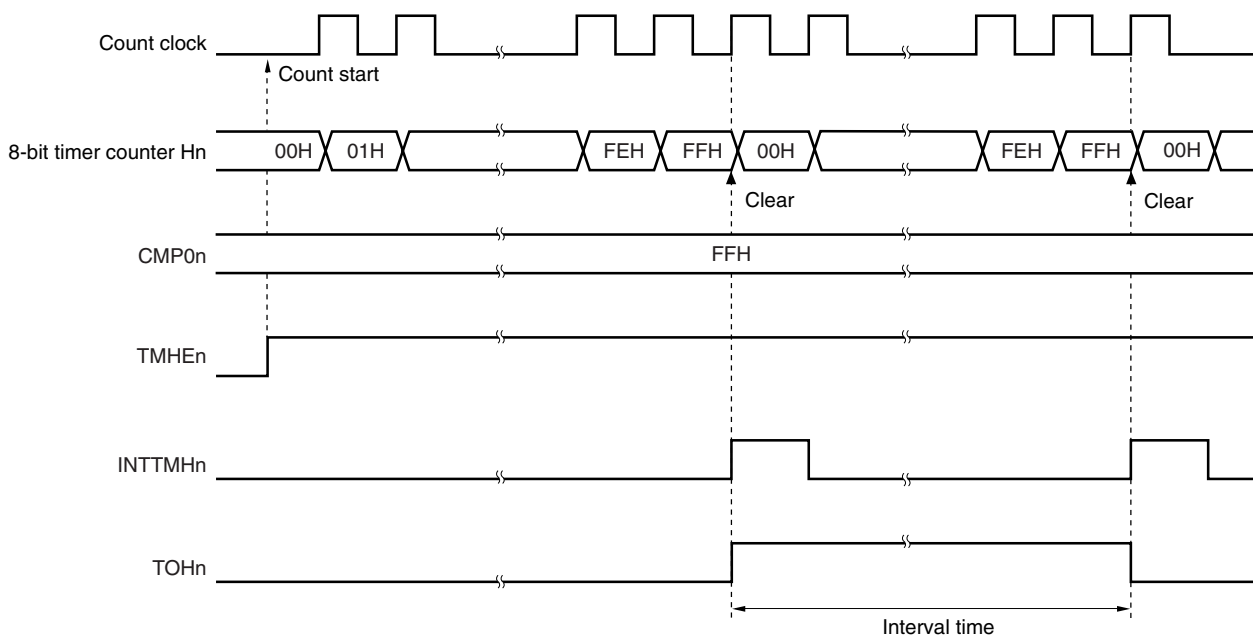
(a) 基本操作($01H \leq CMP01 \leq FEH$ 时的操作)

- <1> 通过将 **TMHE1** 位置“1”，允许计数操作。允许操作后,计数时钟最迟 1 个时钟内开始计数。
- <2> 8 位定时器计数器 **H1** 的值与 **CMP01** 寄存器的值相匹配时，将该定时器计数器清零并反转 **TOH1** 的输出电平。另外，在计数时钟的上升沿输出 **INTTMH1** 信号。
- <3> 若在定时器 **H** 操作期间将 **TMHE1** 位清零，则 **INTTMH1** 信号和 **TOH1** 输出设置为默认电平。若 **TMHE1** 位在清零 (0)之前已经是默认电平，则电平保持不变。

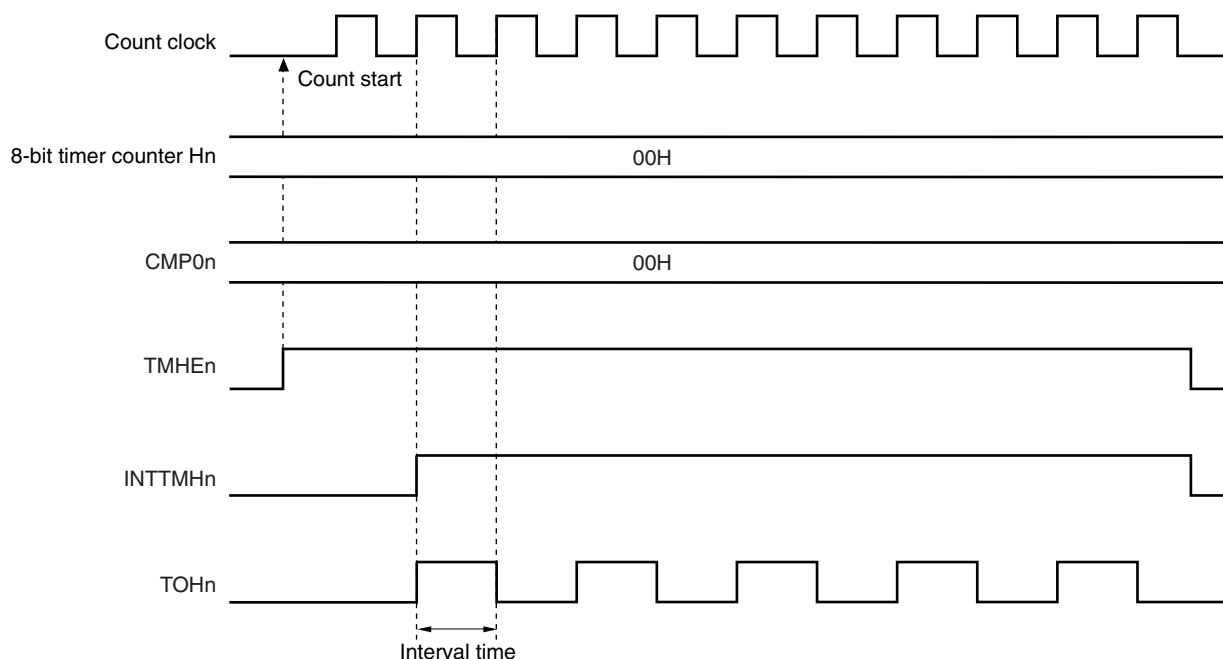
备注 1. $01H \leq N \leq FEH$
 2. $n = 1$

图 8-8. 间隔定时器/方波输出操作的时序 (2/2)

(b) CMP01 = FFH 时的操作



(c) CMP01 = 00H 时的操作



备注 n = 1

8.4.2 用作 PWM 输出

PWM 输出模式中，可以输出任意的占空比和周期脉冲。

8 位定时器比较寄存器 01 (CMP01) 用于控制定时器输出 (TOH1) 的周期。定时器操作期间，禁止重写 CMP01 寄存器。

8 位定时器比较寄存器 11 (CMP11) 用于控制定时器输出 (TOH1) 的占空比。定时器操作期间，允许重写 CMP11 寄存器。

PWM 输出模式下的操作如下所示。

定时器开始计数后，8 位定时器计数器 H1 与 CMP01 寄存器的值相匹配时，PWM 输出(TOH1 输出)输出有效电平，8 位定时器计数器 H1 清零。8 位定时器计数器 H1 和 CMP11 寄存器相匹配时，PWM 输出(TOH1 输出)输出有效电平。

设置

<1> 设置各寄存器。

图 8-9. PWM 输出模式下的寄存器设置

(i) 设置定时器 H 模式寄存器 1 (TMHMD1)



(ii) 设置 CMP01 寄存器

- 比较值(N): 周期设置

(iii) 设置 CMP11 寄存器

- 比较值(M): 占空比设置

备注 $00H \leq \text{CMP11 (M)} < \text{CMP01 (N)} \leq \text{FFH}$

<2> TMHE1 = 1 时，开始计数操作。

<3> 允许计数器操作后，最初用于比较的比较寄存器是 CMP01 寄存器。8 位定时器计数器 H1 与 CMP01 寄存器的值相匹配时，8 为定时器计数器 H1 清零，产生中断请求信号(INTTMH1)，并输出有效电平。同时，将与 8 位定时器计数器 H1 相匹配的比较寄存器从 CMP01 切换到 CMP11 寄存器。

<4> 8 位定时器计数器 H1 与 CMP11 寄存器相匹配时，输出无效电平，并将与 8 位定时器计数器 H1 相匹配的比较寄存器从 CMP11 切换到 CMP01 寄存器。此时，8 位定时器计数器 H1 不能清零，而且不产生 INTTMH1 信号。

<5> 通过重复执行上述<3>和<4>步骤，可获取任意的占空比脉冲。

<6> 若计数操作停止，则设置 **TMHE1** = 0。

若 **CMP01** 寄存器的值设置为 **N**，**CMP11** 寄存器的值设置为 **M**，计数时钟频率设置为 **f_{CNT}**，则 **PWM** 脉冲输入周期和占空比如下所示。

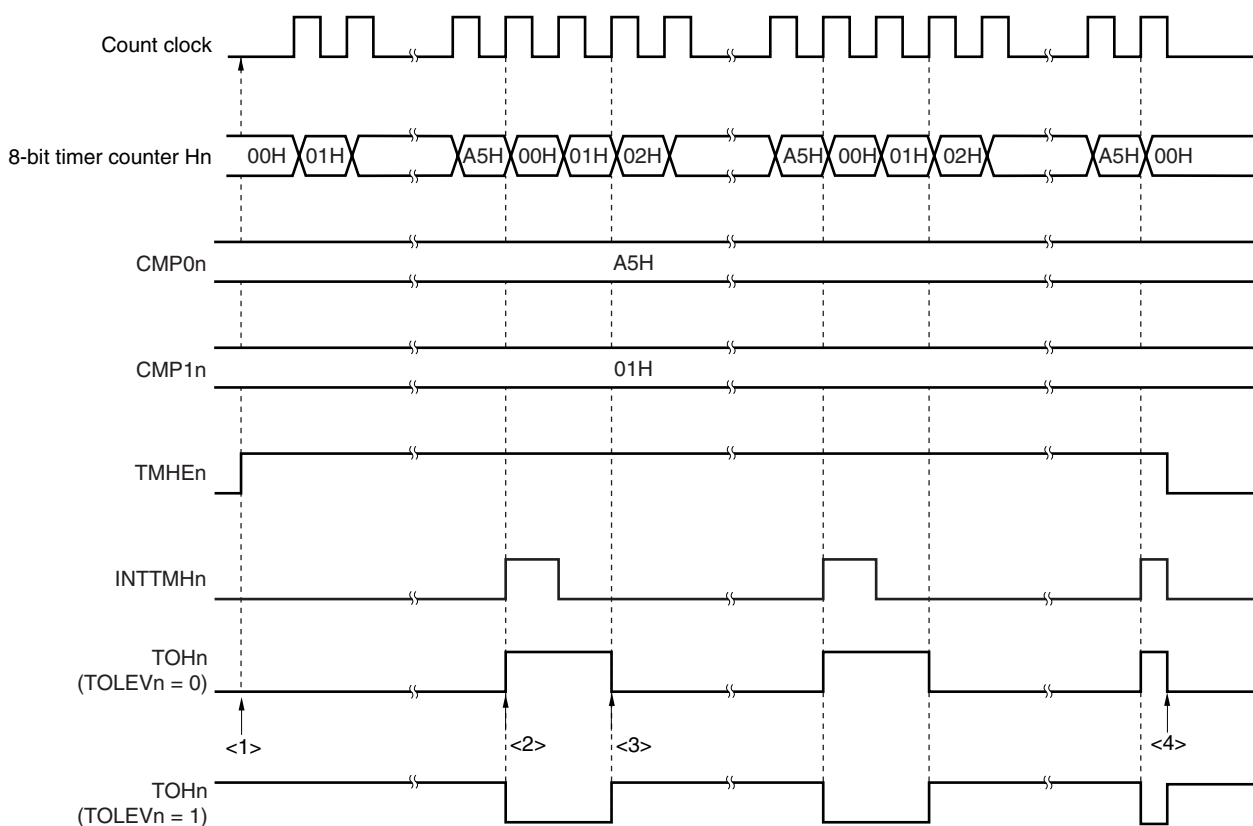
- PWM 脉冲输出周期 = $(N + 1)/f_{CNT}$
- 占空比 = $(M + 1)/(N + 1)$

- 注意事项**
1. 定时器计数器操作期间，可以修改 **CMP11** 寄存器的设定值。但若 **CMP11** 寄存器的值经修改后传送到寄存器，则需要 3 个操作时钟(通过 **TMHMD1** 的 **CKSn2** 至 **CKSn0** 的位选择信号)
 2. 设置定时器计数停止(**TMHE1** = 0)后，启动定时器计数操作(**TMHE1** = 1) 时，必须设置 **CMP11** 寄存器（即使 **CMP11** 的设定值相同，也必须重新设置）。
 3. **CMP11** 寄存器的设定值(**M**)和 **CMP01** 寄存器的设定值(**N**)必须在以下范围内。
00H ≤ CMP11 (M) < CMP01 (N) ≤ FFH

- 备注**
1. 有关输出引脚的设置，参阅 **8.3 (3) 端口模式寄存器 3(PM3)**。
 2. 有关如何允许 **INTTMH1** 信号中断的详细内容，参阅**第十四章 中断功能**。

图 8-10. PWM 输出模式下的操作时序 (1/4)

(a) 基本操作

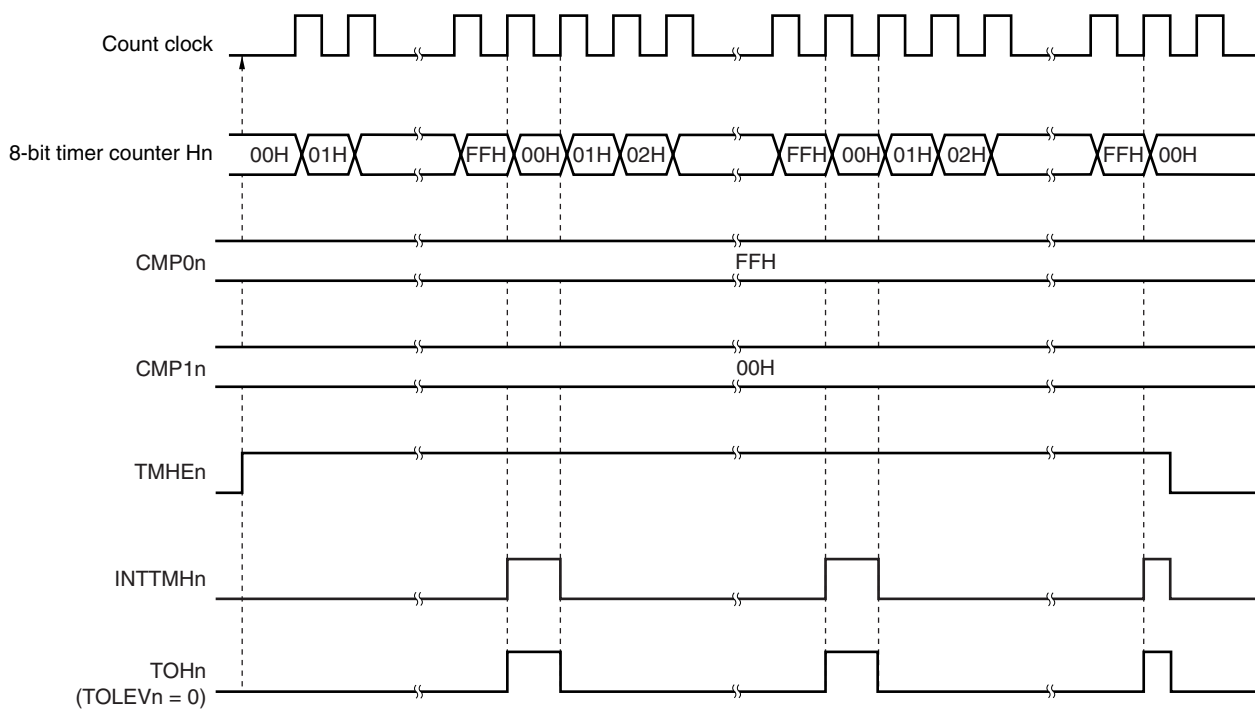


- <1> 通过将 **TMHE1** 位置“1”，允许计数操作。同过屏蔽计数时钟的 1 个时钟，启动 8 位定时器计数器 **H1**，开始计数。此时，**PWM** 输出无效电平。
- <2> 8 为定时器计数器 **H1** 与 **CMP01** 寄存器的值相匹配时，输出有效电平。此时，8 位定时器计数器 **H1** 清零，并输出 **INTTMH1** 信号。
- <3> 8 为定时器计数器 **H1** 与 **CMP11** 寄存器的值相匹配时，输出无效电平。此时，8 位定时器计数器的值不能清零，而且不产生 **INTTMH1** 信号。
- <4> 通过定时器 **H1** 操作期间 **TMHE1** 位清零，**NTTMH1** 信号设置为默认，**PWM** 输出无效电平。

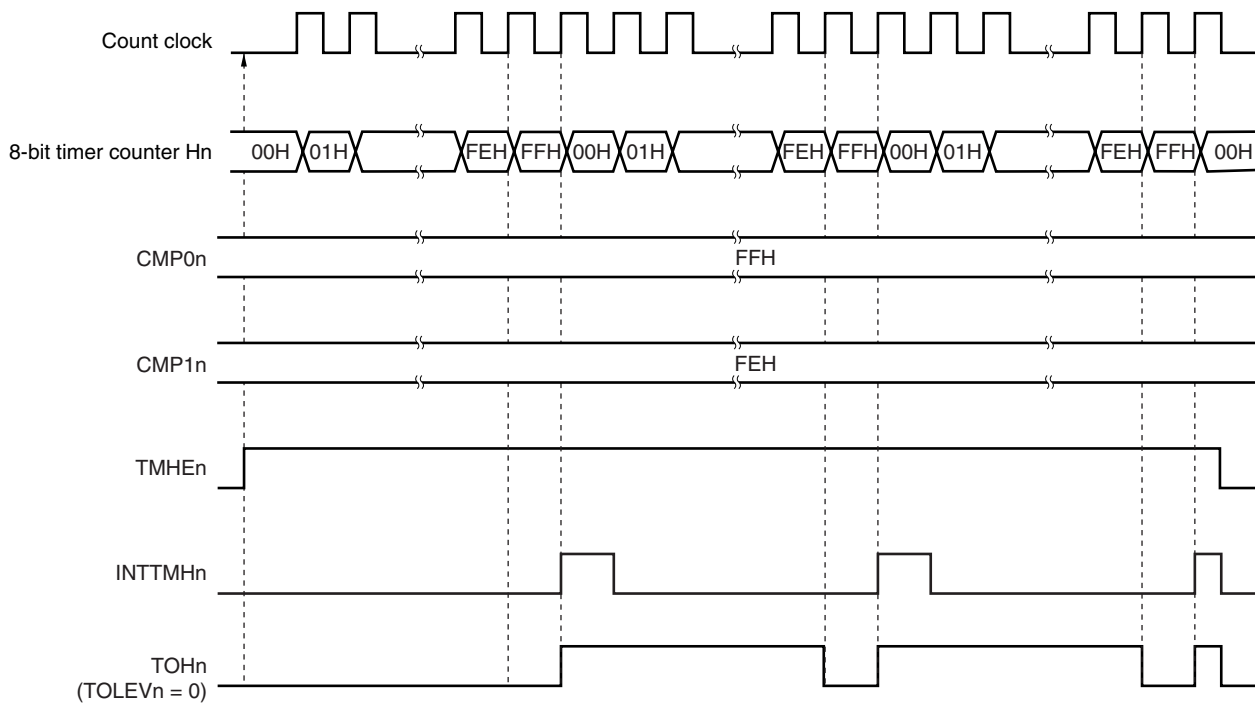
备注 n = 1

图 8-10. PWM 输出模式下的操作时序 (2/4)

(b) CMP01 = FFH, CMP11 = 00H 时的操作



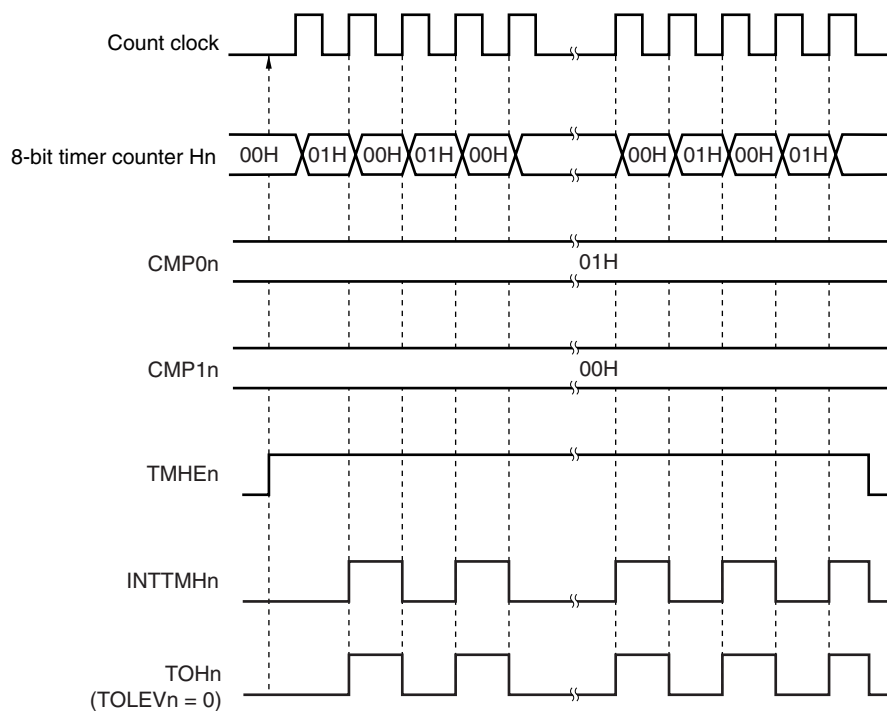
(c) CMP01 = FFH, CMP11 = FEH 时的操作



备注 n = 1

图 8-10. PWM 输出模式下的操作时序 (3/4)

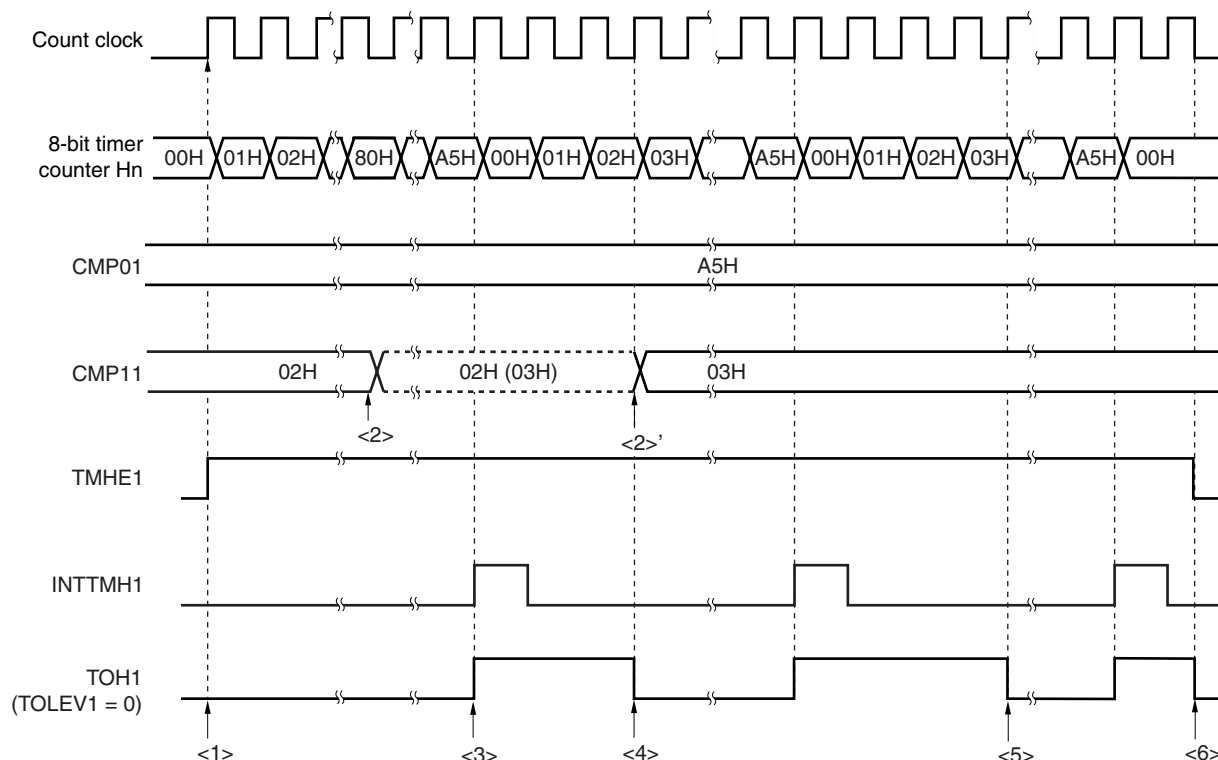
(d) CMP01 = 01H, CMP11 = 00H 时的操作



备注 n = 1

图 8-10. PWM 输出模式下的操作时序 (4/4)

(e) 通过 CMP11 (CMP11 = 02H → 03H, CMP01 = A5H)修改的操作



- <1> 通过将 TMHE1 位置“1”，允许计数操作。同过屏蔽计数时钟的 1 个时钟，启动 8 位定时器计数器 H1，开始计数。此时，PWM 输出无效电平。
- <2> 定时器操作期间,允许修改 CMP11 寄存器值。此操作不与计数时钟同步。
- <3> 8 位定时器计数器 H1 与 CMP01 寄存器的值相匹配时，8 为定时器计数器 H1 清零，并输出有效电平和 NTTMH1 信号。
- <4> 即使修改 CMP11 寄存器的值，该值也被锁存并不传送给寄存器。8 位定时器计数器 H1 与 CMP11 寄存器的值在修改前相匹配时，传送到 CMP11 寄存器并修改 CMP11 寄存器的值(<2>')。
- 但若 CMP11 寄存器的值修改后传送到寄存器，则需要 3 个操作时钟以上。若 3 个时钟内信号相匹配，则修改的值无法传送到寄存器。
- <5> 8 为定时器计数器 H1 的值修改后与 CMP11 寄存器的值相匹配时，输出无效电平。8 位定时器计数器 H1 不能清零，而且不产生 INTTMH1 信号。
- <6> 通过定时器 H1 操作期间 TMHE1 位清零，NTTMH1 信号设置为默认，PWM 输出无效电平。

备注 n = 1

8.4.3 载波发生器操作

载波发生器模式下，8 位定时器 H1 用作红外线遥控机的载波信号产生载波信号，并将 8 位定时器/事件计数器 51 用于产生红外线遥控机信号（时间计数）。

由 8 位定时器/时间计数器 51 的设定周期，输出 8 位定时器 H1 产生的载波时钟。

载波发生器模式下，通过 8 位定时器/时间计数器 51 控制 8 位定时器 H1 的载波脉冲输出量，并从 TOH1 输出载波脉冲。

(1) 载波生成

载波发生器模式下，8 位定时器 H 比较寄存器 01（CMP01）产生低电平宽度载波脉冲波形，而 8 位定时器 H 比较寄存器 11（CMP11）产生高电平宽度载波脉冲波形。

8 位定时器 H1 操作期间，允许重写 CMP11，但禁止重写 CMP01。

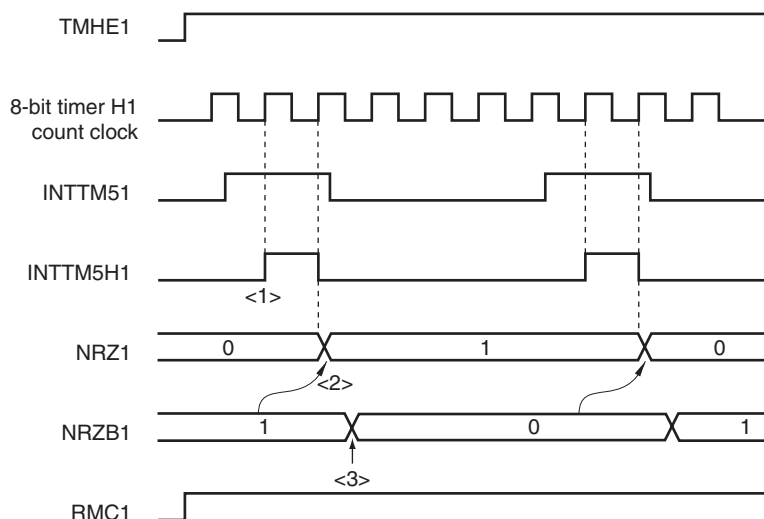
(2) 载波输出控制

由 8 位定时器/事件计数器 51 的中断请求信号（INTTM51）和 8 位定时器 H 载波控制寄存器（TMCYC1）的 NRZB1 和 RMC1 的位控制载波输出。输出之间的关系如下所示。

RMC1 Bit	NRZB1 Bit	输出
0	0	低电平输出
0	1	在 INTTM51 信号输出上升沿的高电平输出
1	0	低电平输出
1	1	在 INTTM51 信号输出上升沿的载波脉冲输出

为了在计数操作期间控制载波脉冲输出，TMCYC1 寄存器的 NRZ1 和 NRZB1 的位由主位和从位组成。NRZ1 位只读，而 NRZB1 位可读写。INTTM51 信号与 8 位定时器 H1 计数时钟同步，并用作 INTTM5H1 信号进行输出。INTTM5H1 信号成为 NRZ1 位的数据传送信号，并将 NRZB1 位的值传送到 NRZ1 位。NRZB1 位至 NRZ1 位的传送时序如下所示。

图 8-11. 传送时序



- <1> INTTM51 信号与 8 位定时器 H1 计数时钟同步，并用作 INTTM5H1 信号进行输出。
- <2> 在 INTTM5H1 信号上升沿的第 2 个时钟处将 NRZB1 位传送到 NRZ1 位。
- <3> 通过 INTTM5H1 中断启动的中断服务程序中，或轮询中断请求标志检测出时序后，将下一个值写入 NRZB1 位。将下一次计数值写入 CR51 寄存器。

注意事项

1. 重写 NRZB1 位后，禁止在第 2 个时钟之前再次重写，否则，不能保证从 NRZB1 位传送到 NRZ1 位。
2. 8 位定时器/计时器 51 用作载波发生器模式时，在 <1>时序下发生中断。8 为定时器/实践计数器 51 用作载波发生器模式以外的模式时，中断发生时序则不同。

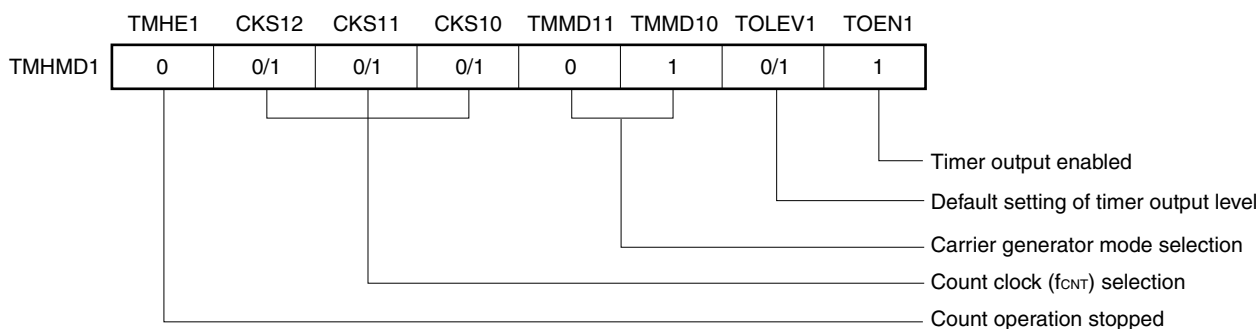
备注 INTTM5H1 是内部信号，不是中断源。

设置

<1> 设置每个寄存器。

图 8-12. 载波发生器模式下的寄存器设置

(i) 设置 8 位定时器 H 模式寄存器 1 (TMHMD1)



(ii) CMP01 寄存器设置

- 比较值

(iii) CMP11 寄存器设置

- 比较值

(iv) TMCYC1 寄存器设置

- RMC1 = 1 ...远控输出允许位
- NRZB1 = 0/1 ... 载波输出允许位

(v) TCL51 和 TMC51 寄存器设置

- 参阅 7.3 8 位定时器/事件计数器 51 控制寄存器

<2> TMHE1 = 1 时, 8 位定时器 H1 开始计数。

<3> 8 位定时器模式控制寄存器 51(TMC51)的 TCE51 置“1”时, 8 位定时器/事件计数器 51 开始计数。

<4> 允许计数操作后, 最初用于比较的比较寄存器是 CMP01 寄存器。8 位定时器计数器 H1 和 CMP01 寄存器的计数值相匹配时, 产生 INTTMH1 信号, 同时 8 位定时器计数器 H1 清零。同时, 将与 8 位定时器计数器 H1 相匹配的比较寄存器从 CMP01 切换到 CMP11 寄存器。

<5> 8 位定时器计数器 H1 和 CMP11 寄存器的值相匹配时, 产生 INTTMH1 信号, 同时 8 位定时器计数器 H1 清零 (00H)。同时, 将与 8 位定时器计数器 H1 相匹配的比较寄存器从 CMP11 切换到 CMP01 寄存器。

<6> 通过重复执行<4>和<5>步骤, 产生载波时钟。

<7> INTTM51 信号与 8 位定时器 H1 计数时钟同步, 并用作 INTTM5H1 信号进行输出。INTTM5H1 信号成为 NRZB1 位的数据传送信号, 并将 NRZB1 位的值传送到 NRZ1 位。

<8> 通过 INTTM5H1 中断启动的中断服务程序中, 或轮询中断请求标志检测出时序后, 将下一个值写入 NRZB1 位。将下一次计数的值写入 CR51 寄存器。

<9> NRZ1 位为高电平时, 由 TOH1 输出载波时钟。

<10> 通过执行上述步骤，获取任意形式的载波时钟。若计数操作停止，则 TMHE1 清零。

若 CMP01 寄存器的值置“N”，CMP11 寄存器的值置“M”，计数时钟频率置“f_{CNT}”，则载波时钟输出周期和占空比如下所示。

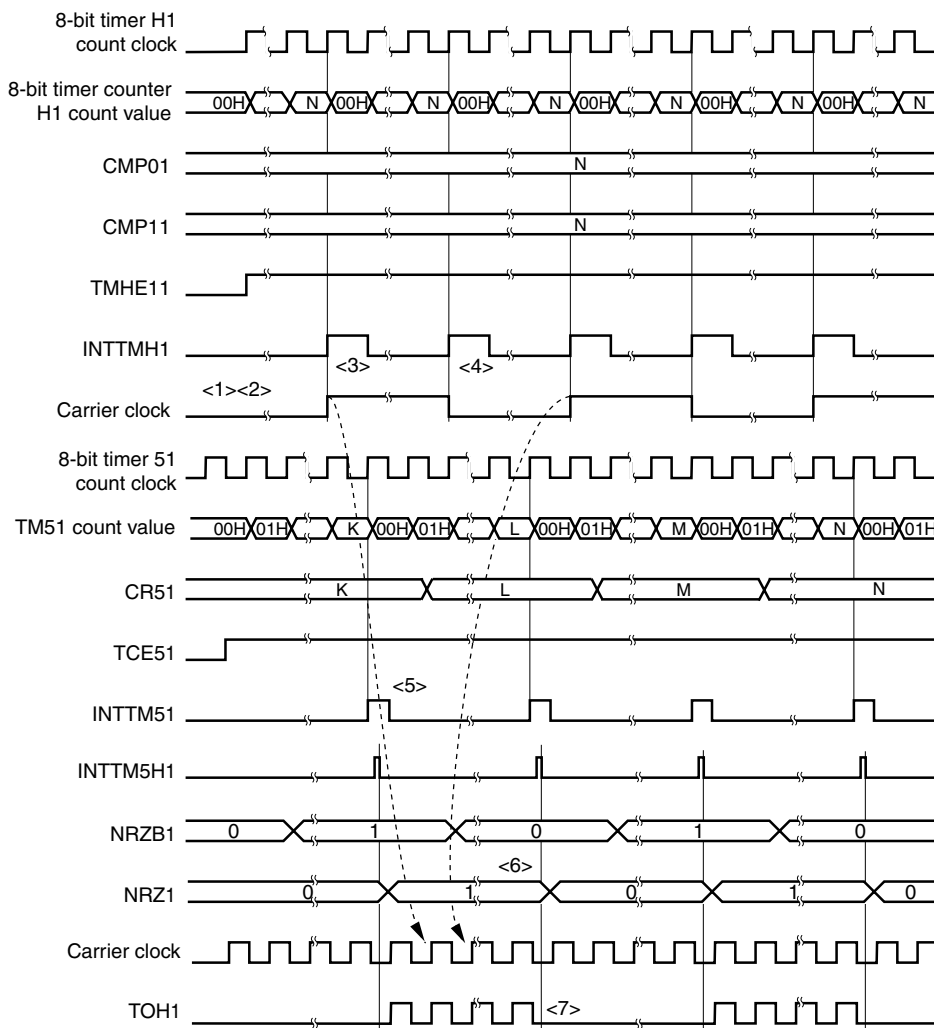
- 载波时钟输出周期 = $(N + M + 2)/f_{CNT}$
- 占空比 = 高电平宽度/载波时钟输出宽度 = $(M + 1)/(N + M + 2)$

- 注意事项**
1. 设置定时器计数停止(TMHE1 = 0)后，启动定时器计数操作(TMHE1 = 1) 时，必须设置 CMP11 寄存器（即使 CMP11 的设定值相同，也必须重新设置）。
 2. 将 TMH1 的计数时钟频率设置到 TM51 计数时钟频率的 6 倍以上。
 3. 在 01H 至 FFH 的范围内设置 CMP01 和 CMP11 寄存器的值。
 4. 定时器计数器操作期间，可以修改 CMP11 寄存器的设定值。但若 CMP11 寄存器的值经修改后传送到寄存器，则需要 3 各操作时钟(通过 TMHMD1 的 CKS12 至 CKS10 的位选择信号)。
 5. 开始计数操作之前,确保设置 RMC1 位。

- 备注**
1. 有关输出引脚的设置，参阅 8.3 (3) 端口模式寄存器 3(PM3)。
 2. 有关如何允许 INTTMH1 信号中断，参阅第十四章 中断功能。

图 8-13. 载波发生器模式操作时序 (1/3)

(a) CMP01 = N, CMP11 = N 时的操作

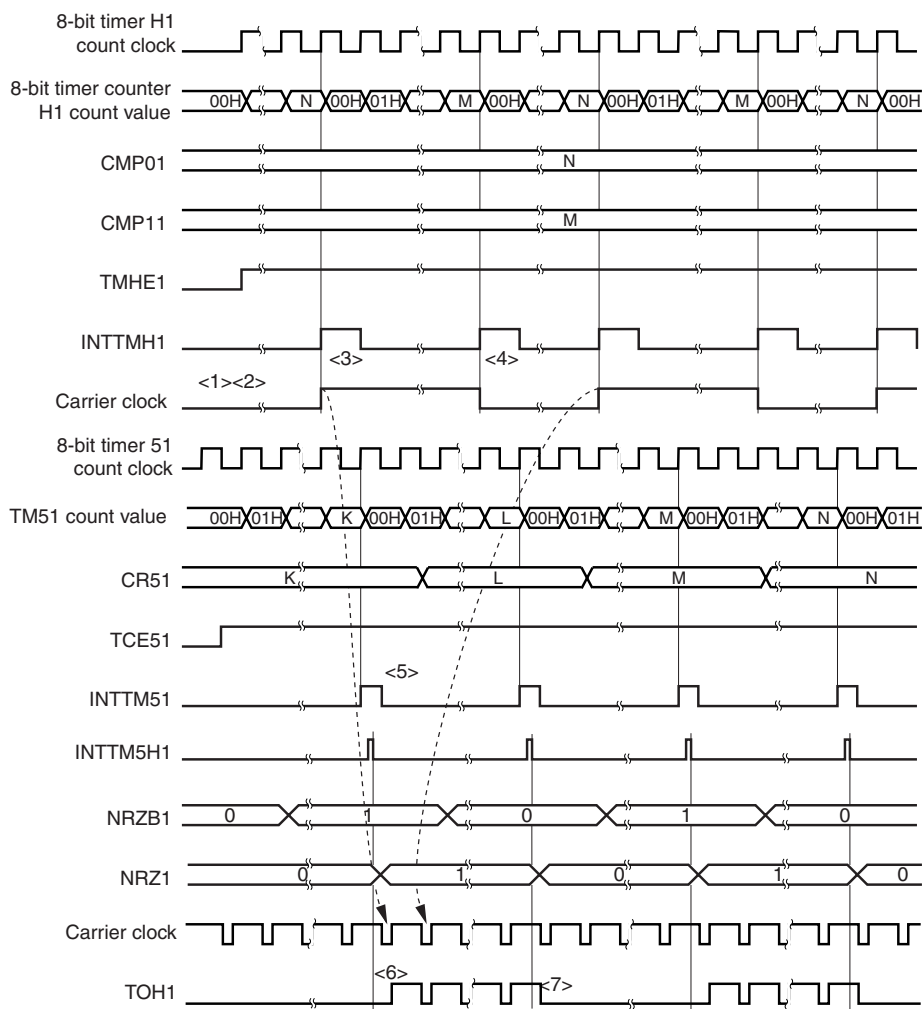


- <1> TMHE1 = 0 和 TCE51 = 0 时，停止 8 位定时器计数器 H1 的操作。
- <2> 设置 TMHE1 = 1 时，8 位定时器 H1 开始计数操作。此时，载波时钟保持默认。
- <3> 8 位定时器计数器 H1 的计数值与 CMP01 寄存器的值相匹配时，产生最初的 INTTMH1 信号，反转载波时钟信号，并将与 8 位定时器计数器 H1 相匹配的比较寄存器从 CMP01 切换到 CMP11 寄存器。8 位定时器计数器 H1 清零(00H)。
- <4> 8 位定时器计数器 H1 的计数值与 CMP11 寄存器的值相匹配时，产生 INTTMH1 信号，反转载波时钟信号，并将与 8 位定时器计数器 H1 相匹配的比较寄存器从 CMP11 切换到 CMP01 寄存器。8 位定时器计数器 H1 清零(00H)。通过重复执行<3>和<4>步骤，产生固定占空比（50%除外）的载波时钟。
- <5> INTTM51 信号产生时，与 8 位定时器 H1 计数时钟同步后，用作 INTTM5H1 信号进行输出。
- <6> INTTM5H1 信号成为 NRZB1 位的数据传送信号，并将 NRZB1 位的值传送到 NRZ1 位。
- <7> 设置 NRZ1 = 0 时，TOH1 输出为低电平。

备注 INTTM5H1 是内部信号，不是中断源。

图 8-13. 载波发生器模式操作时序 (2/3)

(b) CMP01 = N, CMP11 = M 时的操作

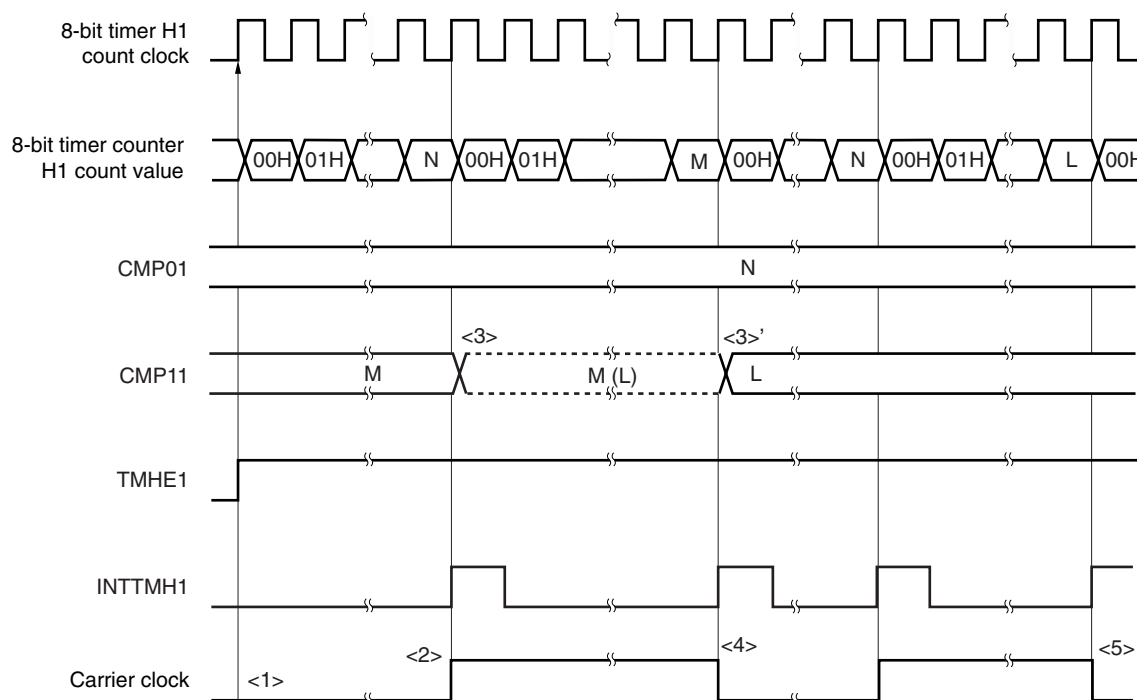


- <1> TMHE1 = 0 和 TCE51 = 0 时, 停止 8 位定时器计数器 H1 的操作。
- <2> 设置 TMHE1 = 1 时, 8 位定时器 H1 开始计数操作。此时, 载波时钟保持默认。
- <3> 8 位定时器计数器 H1 的计数值与 CMP01 寄存器的值相匹配时, 产生最初的 INTTMH1 信号, 反转载波时钟信号, 并将与 8 位定时器计数器 H1 相匹配的比较寄存器从 CMP01 切换到 CMP11 寄存器。8 位定时器计数器 H1 清零(00H)。
- <4> 8 位定时器计数器 H1 的计数值与 CMP11 寄存器的值相匹配时, 产生 INTTMH1 信号, 反转载波时钟信号, 并将与 8 位定时器计数器 H1 相匹配的比较寄存器从 CMP11 切换到 CMP01 寄存器。8 位定时器计数器 H1 清零(00H)。通过重复执行<3>和<4>步骤, 产生固定占空比(50%除外)的载波时钟。
- <5> INTTM51 信号产生时, 与 8 位定时器 H1 计数时钟同步后, 用作 INTTM5H1 信号进行输出。
- <6> 若 NRZ1 置“1”时, 则在载波时钟的最初上升沿输出载波信号。
- <7> NRZ1 = 0 时, 载波时钟的高电平期间, TOH1 输出保持高电平, 不切换到低电平(在<6>和<7>步骤中, 保证载波时钟波形的高电平宽度)。

备注 INTTM5H1 是内部信号, 不是中断源。

图 8-13. 载波发生器模式操作时序 (3/3)

(c) CMP11 时的操作



- <1> 设置 $TMHE1 = 1$ 时, 8 位定时器 H1 开始计数操作。此时, 载波时钟保持默认。
- <2> 8 位定时器计数器 H1 的值与 CMP01 寄存器的值相匹配时, 输出 INTTMH1 信号, 反转 TOH1 的输出电平, 并将该定时器计数器清零(00H)。同时, 将与 8 位定时器计数器 H1 的值相匹配的比较寄存器的值从 CMP01 切换到 CMP11 寄存器。
- <3> CMP11 寄存器不与计数时钟同步, 并且在 8 位定时器 H1 操作期间可以修改其值。修改后的新值(L)被锁存。8 位定时器计数器 H1 的计数值与 CMP11 寄存器修改前的值 (M) 相匹配 (<3>') 时, 修改 CMP11 的值(<3>')。但是, CMP11 寄存器的值修改后传送到寄存器, 需要 3 个操作时钟以上。即使在 3 个时钟内信号产生匹配信号时, 新值将无法传送到寄存器。
- <4> 8 位定时器计数器 H1 的计算值与 CMP11 寄存器的值(M)在修改之前相匹配时, 输出 INTTMH1 信号, 反转载波信号, 并将该定时器计数器清(00H)。同时, 将与 8 位定时器计数器 H1 相匹配的比较寄存器从 CMP11 切换到 CMP01 寄存器。
- <5> 由修改后的值(L)指定 8 位定时器计数器 H1 的计算值与 CMP11 寄存器的值再次匹配时的时序。

第九章 看门狗定时器

9.1 看门狗定时器功能

看门狗定时器使用内部振荡时钟。

看门狗定时器用于检测程序意外挂起。若检测出程序挂起，则产生内部复位信号。

在以下情况下，判断为程序挂起。

- 看门狗定时器计数器溢出。
- 在看门狗定时器允许寄存器(WDTE)执行 1 位操作指令。
- 将“ACH”以外的数据写入 WDTE。
- 窗口关闭期间，将 WDTE 以外的数据写入 WDTE。
- 从未经 IMS 寄存器设置的区域获取指令。
- 通过执行读写指令，访问未经 IMS 寄存器设置的区域（FB00H 至 FFFFH 除外）（CPU 程序挂起期间，检测出异常访问）

因看门狗定时器产生复位时，将复位控制标志寄存器(RESF)置“1”。有关详细的 RESF，参阅第十六章 复位功能。

9.2 看门狗定时器配置

看门狗定时器包含以下硬件。

表 9-1. 看门狗定时器配置

项目	配置
控制寄存器	看门狗定时器允许寄存器 (WDTE)

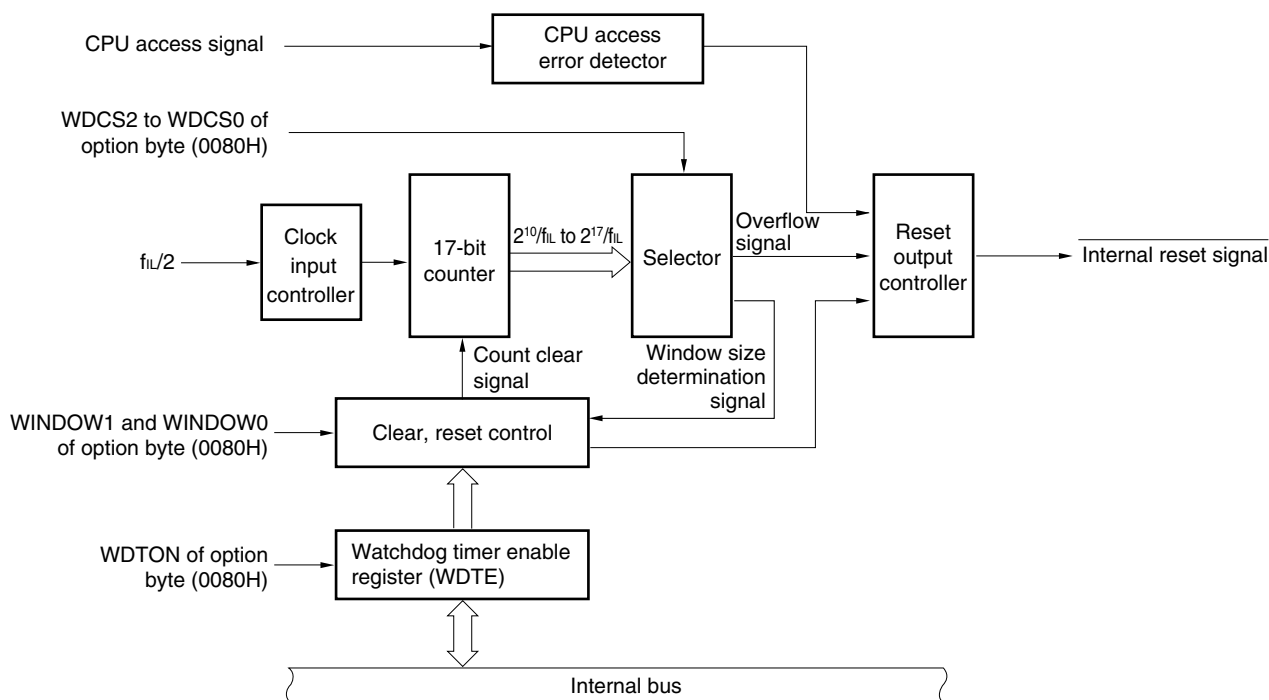
通过选项字节，设置计数器操作的控制、溢出时间、窗口打开期间。

表 9-2. 设置选项字节和看门狗定时器

设置看门狗定时器	选项字节(0080H)
窗口打开期间	第6位和第5位 (窗口1, 窗口0)
看门狗定时器的计数器操作控制	第4位 (WDTON)
看门狗定时器溢出时间	第3位至第1位 (WDACS2 至 WDACS0)

备注 选项字节的详情参阅第二十章 选项字节。

图 9-1. 看门狗定时器的框图



9.3 寄存器控制看门狗定时器

由看门狗定时器允许寄存器(WDTE)控制看门狗定时器。

(1) 看门狗定时器允许寄存器 (WDTE)

通过将“ACH”写入 WDTE，看门狗定时器计数器清零，并重新开始计数。

可用 8 位存储器操作指令设置该寄存器。

复位信号产生时，该寄存器设置为 9AH 或 1AH[※]。

图 9-2. 看门狗定时器允许寄存器 (WDTE) 格式

地址: FF99H	复位后: 9AH/1AH ^{Note}	R/W						
符号	7	6	5	4	3	2	1	0
WDTE								

注 WDTE 复位值取决于选项字节的设定值(0080H)。通过操作看门狗定时器，WDTON 置“1”。

WDTON设定值	WDTE复位值
0 (禁止看门狗定时器计数操作)	1AH
1 (允许看门狗定时器计数操作)	9AH

- 注意事项
1. 若将“ACH”以外的值写入 WDTE，则产生内部复位信号。但若看门狗定时器的源时钟停止，则重新启动看门狗计时器源时钟的同时，产生内部复位信号。
 2. 若对 WDTE 寄存器执行 1 位存储器操作指令，则产生内部复位信号。但若看门狗定时器的源时钟停止，则重新启动看门狗计时器源时钟的同时，产生内部复位信号。
 3. WDTE 的读取值为 9AH/1AH(与写入值(ACH)不同)。

9.4 看门狗定时器操作

9.4.1 看门狗定时器的操作控制

- 使用看门狗定时器时，由选项字节(0080H)指定其操作。
 - 通过设置选项字节(0080H)的第 4 位(WDTON)=1，允许看门狗定时器的计数操作(复位解除后，计数器开始计数)(详阅第二十章)。

WDTON	看门狗定时器计数器/非法访问检测的操作控制
0	禁止计数器操作(复位后停止计数),禁止非法访问检测操作
1	允许计数器操作(复位后开始计数),允许非法访问检测操作

- 通过选项字节(0080H)的第 3 位和第 1 位(WDCS2 至 WDCS0)，设置溢出时间（详阅 9.4.2 和第二十章）。
 - 通过选项字节(0080H)的第 6 位和第 5 位(WINDOW1 和 WINDOW0)，设置窗口打开期间（详阅 9.4.3 和第二十章）。
- 复位解除后，看门狗计数器开始计数。
 - 计数器开始计数后，选项字节设置溢出时间之前，将“ACH”写入 WDTE 时，看门狗定时器清零并重新开始计数。
 - 经上述步骤之后，复位解除后对 WDTE 进行第 2 次之后的写操作，应在窗口打开期间进行操作。若在窗口关闭期间对 WDTE 进行写入时，则产生内部复位信号。
 - 若溢出时间经过后，还没有将“ACH”写入 WDTE，则产生内部复位信号。
在下列情况下，产生内部复位信号。

- 在看门狗定时器允许寄存器(WDTE)执行 1 位操作指令。
- 将“ACH”以外的数据写入 WDTE。
- 从未经 IMS 寄存器设置的区域获取指令。(CPU 程序挂起期间，检测出的无效校验)
- 通过 CPU 的读写执行指令，访问未经 IMS 寄存器设置的区域（FB00H 至 FFFFH 除外）（CPU 程序挂起期间，检测出异常访问）

- 注意事项**
- 复位解除后对 WDTE 进行第一次写操作时，若是在到达溢出时间之前，则无论时序如何，看门狗定时器都重新开始计数。
 - 若通过将“ACH”写入 WDTE 执行看门狗定时器清零，则实际的溢出时间可能与由选项字节设置的溢出时间之间产生最长 2/fRL 秒的误差。
 - 计数值溢出(FFFFH)之前，看门狗定时器可以立即清零。
 - 如下所示，看门狗定时器的操作在 HALT 和 STOP 模式下，根据选项字节第 0 位(LSROSC)设定值的不同而不同。

	LSROSC = 0 (允许通过软件停止内部低速振荡器)	LSROSC = 1 (禁止停止内部低速振荡器)
HALT模式下	看门狗定时器操作停止。	看门狗定时器操作持续。
STOP模式下		

若 LSROSC = 0，则 HALT 或 STOP 模式复位后，看门狗定时器恢复计数。此时，计数器不清零，而从计数器停止时的值开始继续计数。

若在 LSROSC = 0 时，通过设置 LSRSTOP(内部振荡模式寄存器(RCM)的第 1 位=1)停止内部低速振荡器的振荡，则看门狗定时器停止操作。此时，寄存器不清零。

- 闪存自编程期间，看门狗定时器将持续运行。但在此处理期间，中断响应时间会出现延迟，因此，在设置溢出时间和窗口尺寸时，应充分考虑延迟时间。

9.4.2 看门狗定时器的溢出时间设置

通过选项字节(0080H)的第3位置“1”(WDCS2至WDCS0)，设置看门狗定时器的溢出时间。

若出现溢出，则产生内部复位信号。达到溢出时间之前的窗口打开期间，通过将“ACH”写入WDTE，可将当前计数值清零，并且看门狗定时器重新开始计数。

溢出时间的设置如下所示。

表 9-3. 看门狗定时器的溢出时间设置

WDCS2	WDCS1	WDCS0	看门狗定时器的溢出时间
0	0	0	$2^{10}/f_{IL}$ (3.88 ms)
0	0	1	$2^{11}/f_{IL}$ (7.76 ms)
0	1	0	$2^{12}/f_{IL}$ (15.52 ms)
0	1	1	$2^{13}/f_{IL}$ (31.03 ms)
1	0	0	$2^{14}/f_{IL}$ (62.06 ms)
1	0	1	$2^{15}/f_{IL}$ (124.12 ms)
1	1	0	$2^{16}/f_{IL}$ (248.24 ms)
1	1	1	$2^{17}/f_{IL}$ (496.48 ms)

注意事项 1. 禁止 WDCS2 = WDCS1 = WDCS0 = 0 和 WINDOW1 = WINDOW0 = 0 的组合设置。

2. 闪存自编程期间，看门狗定时器将持续运行。但在此处理期间，中断响应时间会出现延迟，因此，在设置溢出时间和窗口尺寸时，应充分考虑延迟时间。

备注 1. f_{IL} : 内部低速振荡时钟频率

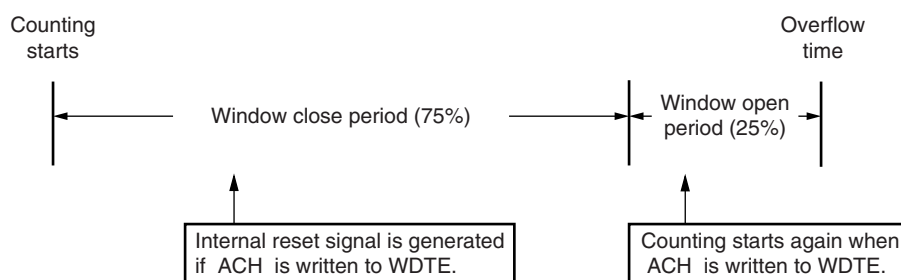
2. (): $f_{IL} = 33 \text{ kHz (MAX.)}$

9.4.3 看门狗定时器的窗口打开期间设置

通过选项字节(0080H)的第 6 位和第 6 位(WINDOW1 和 WINDOW0)，设置看门狗定时器的窗口打开期间。窗口概述如下。

- 若在窗口打开期间将“ACH”写入 WDTE，则看门狗定时器清零并重新开始计数。
- 在窗口关闭期间，即使将“ACH”写入 WDTE，也会检测出异常并产生内部复位信号。

示例的窗口打开周期为 25%时



注意事项 复位解除后对 WDTE 进行第一次写操作时，若是在到达溢出时间之前，则无论时序如何，看门狗定时器都重新开始计数。

窗口打开期间的设置如下所示。

表 9-4. 看门狗定时器的窗口打开期间设置

WINDOW1	WINDOW0	看门狗定时器的窗口打开期间
0	0	25%
0	1	50%
1	0	75%
1	1	100%

- 注意事项**
1. 禁止 $WDCS2 = WDCS1 = WDCS0 = 0$ 和 $WINDOW1 = WINDOW0 = 0$ 的组合设置。
 2. 闪存自编程期间，看门狗定时器将持续运行。但在此处理期间，中断响应时间会出现延迟，因此，在设置溢出时间和窗口尺寸时，应充分考虑延迟时间。

备注 若将溢出时间设置为 $2^{11}/f_{IL}$ ，则窗口关闭时间和打开时间如下所示。

	窗口打开期间的设置			
	25%	50%	75%	100%
窗口关闭时间	0 至 7.11 ms	0 至 4.74 ms	0 至 2.37 ms	无
窗口打开时间	7.11 至 7.76 ms	4.74 至 7.76 ms	2.37 至 7.76 ms	0 至 7.76 ms

<窗口打开周期为 25%时>

- 溢出时间:
 $2^{11}/f_{IL} \text{ (MAX.)} = 2^{11}/264 \text{ kHz (MAX.)} = 7.76 \text{ ms}$
- 窗口关闭时间:
 $0 \text{ 至 } 2^{11}/f_{IL} \text{ (MIN.)} \times (1 - 0.25) = 0 \text{ 至 } 2^{11}/216 \text{ kHz (MIN.)} \times 0.75 = 0 \text{ 至 } 7.11 \text{ ms}$
- 窗口打开时间:
 $2^{11}/f_{IL} \text{ (MIN.)} \times (1 - 0.25) \text{ 至 } 2^{11}/f_{IL} \text{ (MAX.)} = 2^{11}/216 \text{ kHz (MIN.)} \times 0.75 \text{ 至 } 2^{11}/264 \text{ kHz (MAX.)}$
 $= 7.11 \text{ 至 } 7.76 \text{ ms}$

第十章 A/D 转换器

10.1 A/D 转换器功能

A/D 转换器备有将模拟输入信号转换到数字信号的 8 位分辨率，最多可以控制 5 个通道(ANIO 至 ANI4)模拟输入。

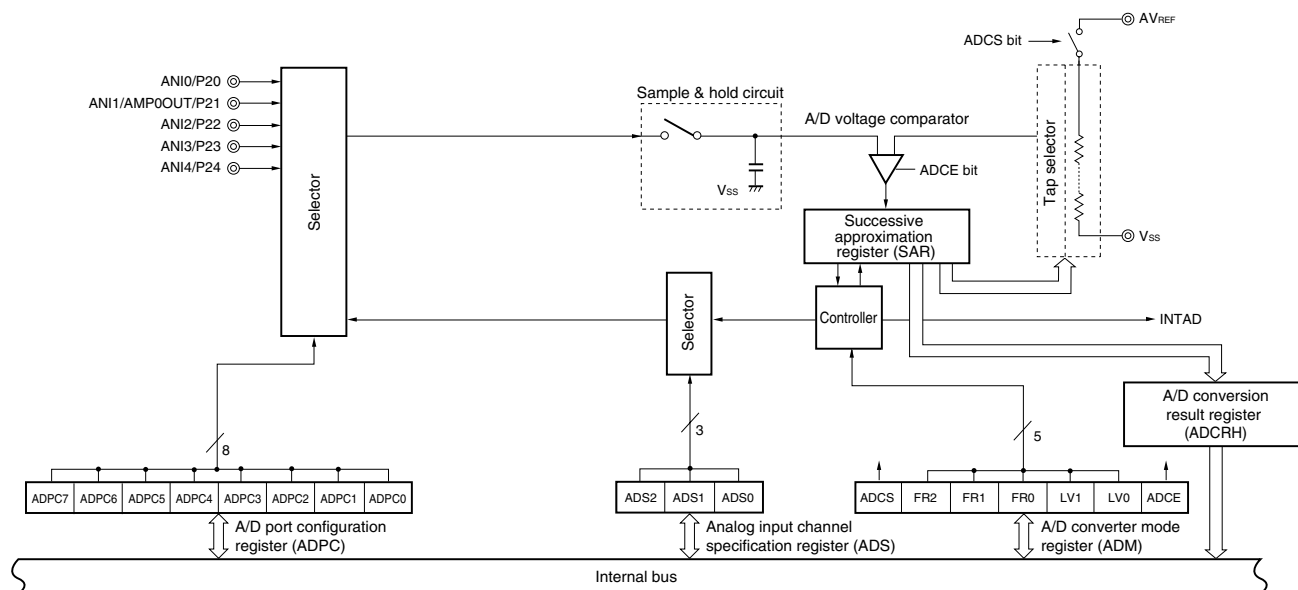
ANI1 还可用作放大器 0 输出(AMP0OUT)的引脚功能。因此，可将放大器 0 输出用作模拟输入源。

A/D 转换器具有以下功能。

- 8 位分辨率的 A/D 转换

使用放大器 0 输出从 ANIO 至 ANI4 选择一个模拟输入频道，重复 8 位分辨率的 A/D 转换动作。每次完成 A/D 转换后，都会产生中断请求（INTAD）。

图 10-1. A/D 转换器的框图



注意事项 V_{SS} 兼备 A/D 转换器的接地电位功能。必须连接 V_{SS} 至稳定的 GND (= 0 V) 上。

10.2 A/D 转换器配置

A/D 转换器包括下列硬件。

(1) ANI0 至 ANI4 引脚

5 个通道 A/D 转换器的模拟输入引脚。将输入模拟信号转换成数字信号。除选作模拟输入的引脚之外，其他引脚均可用作 I/O 端口引脚。

(2) AMP0OUT 引脚

AMP0OUT 是放大器 0 的输出引脚。

AMP0OUT 可兼作 ANI1。A/D 转换器将放大器 0 的输入信号作为模拟输入源，进行 A/D 转换。

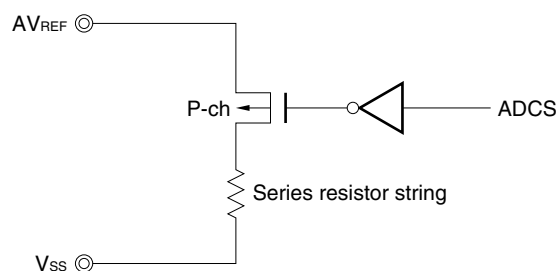
(3) 采样和保持电路

启动 A/D 转换器时，采样和保持电路对选择器选择的模拟输入的输入电压进行采样，并在 A/D 转换期间保持所采样的电压值。

(4) 串联电阻字符串

串联电阻串用于连接 AVREF 和 AVSS，与采样到的电压值进行比较后，产生电压。

图 10-2. 串行电阻字符串的电路配置



(5) 电压比较器

该电压比较器用于比较采样电压值和串行电阻字符串的输出值。

(6) 逐次逼近寄存器(SAR)

该寄存器从最高有效位开始转换电压比较器比较的结果。

直到最低有效位(LSB)转换成数字值(A/D 转换结束)时，SAR 寄存器的内容将传送到 A/D 转换结果寄存器(ADCR)储存。

(7) 8 位 A/D 转换结果寄存器 H(ADCRH)

每次 A/D 转换结束时，将 A/D 转换结果从逐次逼近寄存器(SAR)传送至 ADCRH 寄存器储存。

注意事项 从 ADCRH 读取数据时，产生等待周期。外围硬件时钟(f_{PRS})停止时，禁止读取 ADCRH 的数据。详阅第二十六章 待机注意事项。

(8) 控制器

控制输入模拟信号转换成数字信号的转换时间，转换操作的启动/停止等。A/D 转换完成时，产生 INTAD。

(9) AV_{REF} 引脚

该引脚用于输入 A/D 转换器的模拟电源/参考电压。端口 2 用作数字端口时，该引脚的电位与 V_{DD} 引脚的电位保持一致。

根据 AV_{REF} 和 V_{SS} 的电压，将输入到 ANI0 至 ANI4 的信号转换成数字信号。

(10) V_{SS} 引脚

该引脚是接地电位引脚。 V_{SS} 兼用作 A/D 转换器的接地电位。必须连接 V_{SS} 至稳定的 GND (= 0 V) 上。

10.3 A/D 转换器寄存器

A/D 转换器使用以下 5 个寄存器。

- A/D 转换器模式寄存器(ADM)
- 8 位 A/D 转换结果寄存器 H(ADCRH)
- 模拟输入通道指定寄存器(ADS)
- A/D 端口配置寄存器(ADPC)
- 端口模式寄存器 2(PM2)

(1) A/D 转换器模式寄存器(ADM)

该寄存器设置模拟输入的 A/D 转换时间和启动/停止的转换。

可用 1 位或 8 位存储器操作指令设置 ADM。

复位信号产生时，该寄存器清为 00H。

图 10-3. A/D 转换模式寄存器(ADM)的格式

地址: FF90H 复位后: 00H R/W

符号	<7>	6	5	4	3	2	1	<0>
ADM	ADCS	0	FR2 ^{注1}	FR1 ^{注1}	FR0 ^{注1}	LV1 ^{注1}	LV0 ^{注1}	ADCE

ADCS	A/D 转换操作控制
0	停止转换操作
1	允许转换操作

ADCE	电压比较器操作控制 ^{注2}
0	停止电压比较器操作
1	允许电压比较器操作

注 1. 有关 FR2 至 FR0、LV1、LV0 和 A/D 的转换，详阅表 10-2 A/D 转换时间选择。

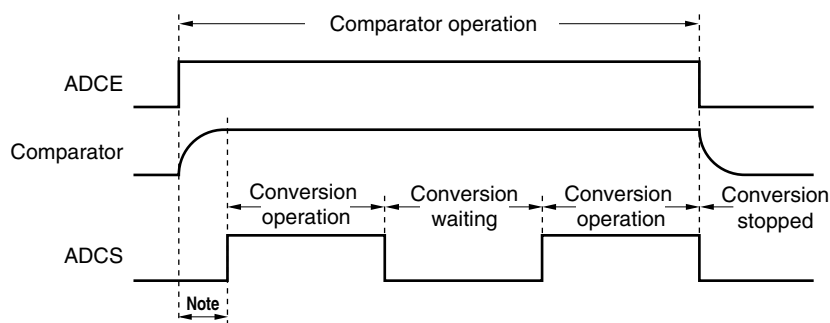
2. 通过 ADCS 和 ADCE 控制电压比较器，从开始操作到稳定需要 1 μs。因此，从 ADCE 置“1”开始经过 1 μs 后，将 ADCS 置“1”时，最初的转换结果开始生效；1 μs 之前将 ADCS 置“1”时，则忽略最初的转换数据。

表 10-1. ADCS 和 ADCE 的设置

ADCS	ADCE	A/D 转换操作
0	0	停止状态（不存在直流功耗）
0	1	转换等待模式（只有电压比较器产生功耗）
1	0	转换模式（停止比较器操作 ^注 ）
1	1	转换模式（电压比较器操作）

注 忽略最初的转换数据。

图 10-4. 使用比较器时的时序图



注 为了稳定内部电路，从 ADCS 置“1”到 ADCS 置“1”所需时间在 1 μs 以上。

- 注意事项
1. 将 FR0 至 FR2、LV1 和 LV0 重写成其他值之前，必须停止 A/D 转换。
 2. 若将数据写入 ADM,则产生一个等待周期。外围硬件时钟(f_{PRS})停止时，禁止将数据写入 ADM。详阅第二十六章 待机注意事项。

表 10-2. A/D 转换时间选择

(1) $2.7\text{ V} \leq \text{AV}_{\text{REF}} \leq 5.5\text{ V}$ ($\text{LV0} = 0$)

A/D转换器模式寄存器(ADM)						转换时间选择		转换时钟(f _{AD})
FR2	FR1	FR0	LV1	LV0		f _{PRS} = 2 MHz	f _{PRS} = 10 MHz	
0	0	0	0	0	264/f _{PRS}	禁止设置	26.4 μs	f _{PRS} /12
0	0	1	0	0	176/f _{PRS}		17.6 μs	f _{PRS} /8
0	1	0	0	0	132/f _{PRS}		13.2 μs	f _{PRS} /6
0	1	1	0	0	88/f _{PRS}		8.8 μs [‡]	f _{PRS} /4
1	0	0	0	0	66/f _{PRS}	33.0 μs	6.6 μs [‡]	f _{PRS} /3
1	0	1	0	0	44/f _{PRS}	22.0 μs	禁止设置	f _{PRS} /2
其它								

注 仅在 $4.0\text{ V} \leq \text{AV}_{\text{REF}} \leq 5.5\text{ V}$ 时，才能设置。

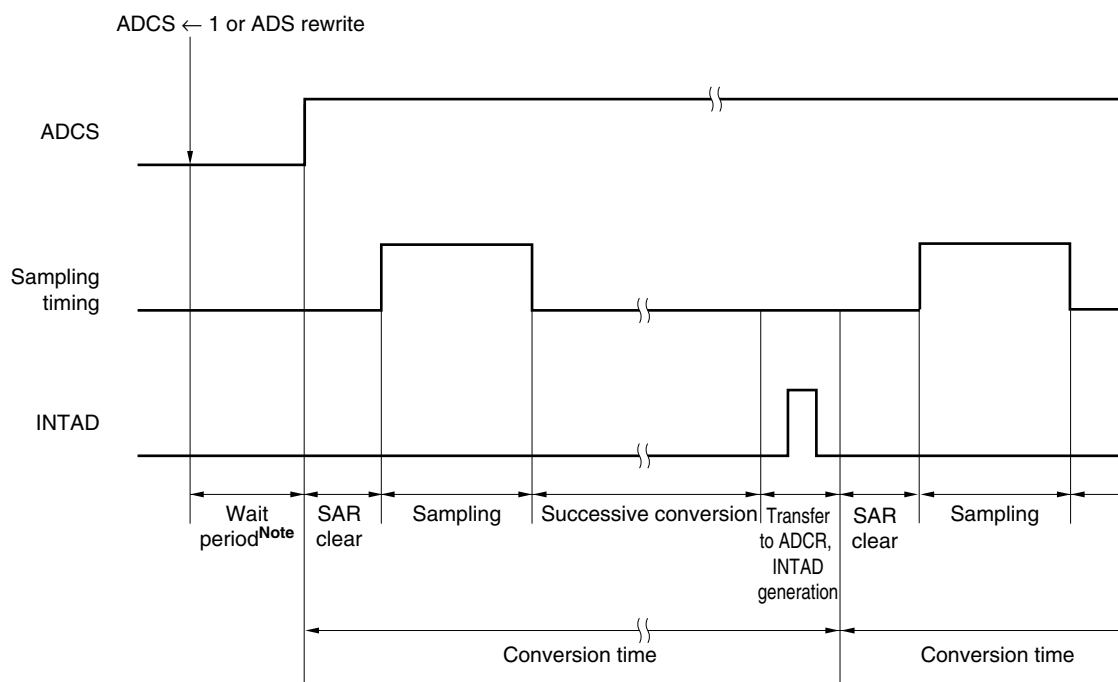
注意事项 1. 在以下条件下，设置转换时间。

 $4.0\text{ V} \leq \text{AV}_{\text{REF}} \leq 5.5\text{ V}$ 时: $f_{\text{AD}} = 0.6\text{ V}$ 至 3.6 MHz $2.7\text{ V} \leq \text{AV}_{\text{REF}} < 4.0\text{ V}$ 时: $f_{\text{AD}} = 0.6\text{ V}$ 至 1.8 MHz 2. FR2 至 FR0、LV1 和 LV0 重写成其他值时，预先停止 A/D 转换($\text{ADCS} = 0$)。

3. 上述转换时间不包括时钟频率误差。选择转换时间时，要考虑时钟频率误差。

备注 f_{PRS} : 外部硬件时钟频率

图 10-5. A/D 转换采样和 A/D 转换时序



注 有关等待期间的详情参阅第二十六章 待机注意事项。

(2) 8 位 A/D 转换结果寄存器 H(ADCRH)

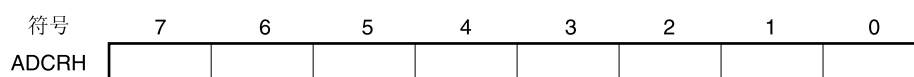
该 8 位寄存器用来存储 A/D 转换结果。

可按 8 位存储器操作指令设置 ADCRH。

复位信号产生时，该寄存器清零(00H)。

图 10-6. 8 位 A/D 转换结果寄存器(ADCRH)格式

地址: FF09H 复位后: 00H R



- 注意事项**
1. 对 A/D 转换器模式寄存器(ADM)、模拟输入频道指定寄存器(ADS)和 A/D 端口配置寄存器(ADPC)进行赋值时，ADCRH 的内容可能不确定。转换结束后读取转换结果，赋值给 ADM、ADS 和 ADPC。若没有使用上述时序，则有可能读取不正确的转换结果。
 2. 若读取 ADCRH 的数据，则产生等待周期。外围硬件时钟(f_{PRS})停止时，禁止读取 ADCRH 的数据。详阅第二十六章 待机注意事项。

(3) 模拟输入通道指定寄存器(ADS)

该寄存器指定模拟电压进行 A/D 转换的输入通道。

可用 1 位或 8 位存储器操作指令设置该 ADS。

复位信号产生时，该寄存器清为 00H。

图 10-7. 模拟输入通道指定寄存器(ADS)格式

地址: FF0EH 复位后: 00H R/W

符号	7	6	5	4	3	<2>	<1>	<0>
ADS	0	0	0	0	0	ADS2	ADS1	ADS0

ADS2	ADS1	ADS0	模拟输入通道	输入源
0	0	0	ANI0	P20/ANI0引脚
0	0	1	ANI1	P21/ANI1引脚或 运算放大器0 输出信号
0	1	0	ANI2	P22/ANI2引脚
0	1	1	ANI3	P23/ANI3引脚
1	0	0	ANI4	P24/ANI4引脚
上述除外的部分			禁止设置	

- 注意事项**
1. 确保第 3 位至第 7 位清零。
 2. 通过端口模式寄存器 2(PM2)的输入模式下，选择 A/D 转换使用的通道。
 3. 若将数据写入 ADS，则产生一个等待周期。外围硬件时钟(f_{PRS})停止时，禁止将数据写入 ADS。详阅第二十六章 待机注意事项。

(4) A/D 端口配置寄存器(ADPC)

ADPC 将 P20/AMP0-/ANI0 至 P24/ANI4 引脚切换到数字 I/O 端口或模拟 I/O 端口。可按 1 位单元指定 ADPC 每个位相对应的端口 2 的一个引脚。

可用 1 位或 8 位存储器操作指令设置该寄存器。

复位信号产生时，ADPC 清为 00H。

图 10-8. A/D 端口配置寄存器(ADPC)

地址: FF97H 复位后:00H R/W

符号	7	6	5	4	3	2	1	0
ADPC	ADPC7	ADPC6	ADPC5	ADPC4	ADPC3	ADPC2	ADPC1	ADPC0

ADPCn	数字 I/O 或模拟 I/O 选择(n = 0 至 7)
0	模拟 I/O
1	数字 I/O

- 注意事项**
1. 通过端口模式寄存器 2(PM12)，选择模拟 I/O 引脚的输入方式。
 2. 若将数据写入 ADPC，则产生一个等待周期。当外部硬件时钟停止时，禁止将数据写入 ADPC。详阅第二十六章 待机注意事项。

(5) 端口模式寄存器 (PM2)

ANIO/AMP0-/P20 至 ANI4/P24 引脚用作模拟输入端口时，PM20 至 PM24 分别置“1”。此时，20 至 P24 的输出锁存即可是“0”也可能是“1”。

若 PM20 至 PM24 分别置“0”，则不能用作模拟输入端口引脚。

可用 1 位或 8 位存储器操作指令设置 PM2。

复位信号产生时，该寄存器置为“FFH”。

图 10-9. 端口模式寄存器 2(PM2) 的格式

地址: FF22H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PM2	PM27	PM26	PM25	PM24	PM23	PM22	PM21	PM20

PM2n	P2n 引脚 I/O 模式选择 (n = 0 至 7)
0	输出模式 (输出缓冲打开)
1	输入模式 (输出缓冲关闭)

使用 P20/AMP0-/ANI0 至 P24/ANI4 时, 根据使用的引脚功能设置寄存器(参阅表 10-3 至 10-5)。

表 10-3. P20/ANI0/AMP0-和 P22/ANI2/AMP0+引脚设置功能

ADPC寄存器	PM2寄存器	OPAMP0E位	ADS 寄存器 (n = 0, 2)	P20/ANI0/AMP0-和 P22/ANI2/AMP0+引脚
模拟输入选择	输入模式	0	选择ANIn	模拟输入 (转换为数字模式)
			禁止选择ANIn	模拟输入 (禁止转换为数字模式)
		1	选择ANIn	禁止设置
			禁止选择ANIn	运算放大器 0 输入
	输出模式	—	—	禁止设置
数字 I/O 选择	输入模式	—	选择ANIn	禁止设置
			禁止选择ANIn	数字输入
	输出模式	—	选择ANIn	禁止设置
			禁止选择ANIn	数字输出

备注 ADPC: A/D 端口配置寄存器
 PM2: 端口模式寄存器 2
 OPAMP0E: 运算放大器控制寄存器(AMP0)的第 7 位
 ADS: 模拟输入通道指定寄存器

表 10-4. P21/ANI1/AMP0OUT 引脚设置功能

ADPC寄存器	PM2寄存器	OPAMP0E位	ADS寄存器	P21/ANI1/AMP0OUT引脚
模拟I/O选择	输入模式	0	选择ANI1	模拟输入（转换为数字模式）
			禁止选择ANI1	模拟输入（禁止转换为数字模式）
		1	选择ANI1	运算放大器（转换为数字模式）
			禁止选择ANI1	运算放大器 0输出（转换为数字模式）
	输出模式	—		禁止设置
数字I/O选择	输入模式	0	选择ANI1	禁止设置
			禁止选择ANI1	数字输入
		1	—	禁止设置
	输出模式	0	选择ANI1	禁止设置
			禁止选择ANI1	数字输出
		1	—	禁止设置

表 10-5. P23/ANI3 至 P24/ANI4 引脚功能设置

ADPC寄存器	PM2寄存器	ADS寄存器 (n = 3, 4)	P23/ANI3 至 P24/ANI4引脚
模拟输入选择	输入模式	选择ANIn	模拟输入（转换为数字模式）
		禁止选择ANIn	模拟输入（禁止转换为数字模式）
	输出模式	—	禁止设置
数字I/O选择	输入模式	选择ANIn	禁止设置
		禁止选择ANIn	数字输入
	输出模式	选择ANIn	禁止设置
		禁止选择ANIn	数字输出

备注 ADPC: A/D 端口配置寄存器
 PM2: 端口模式寄存器 2
 OPAMP0E: 运算放大器控制寄存器(AMPM)的第 7 位
 ADS: 模拟输入通道指定寄存器

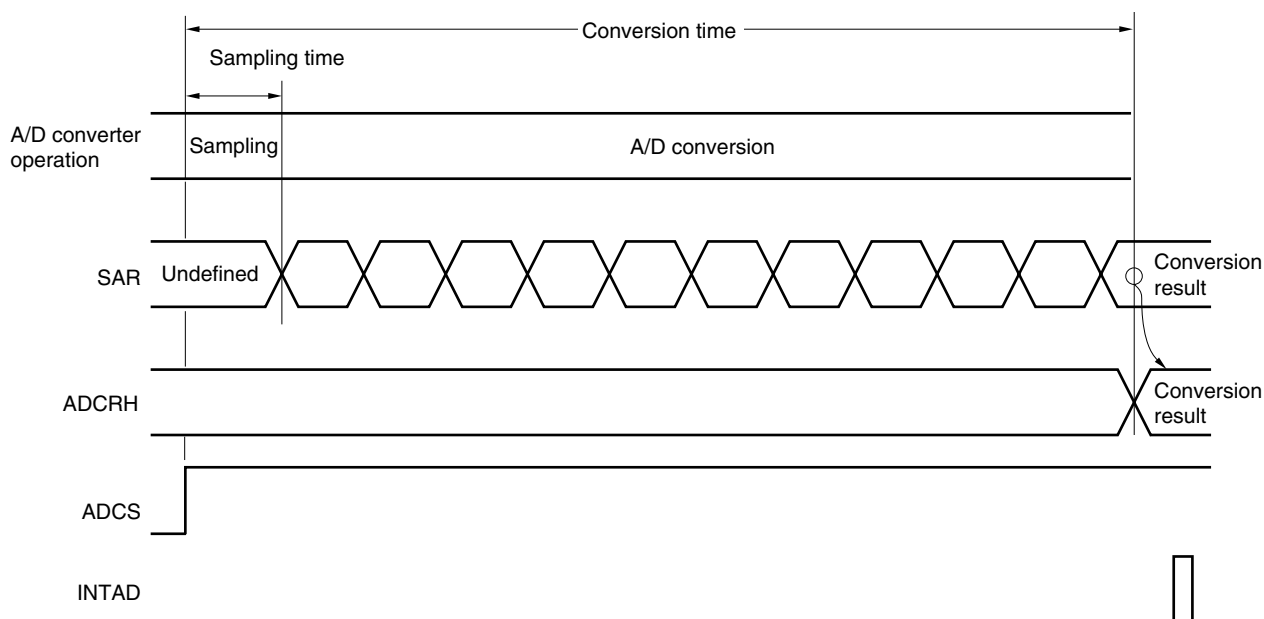
10.4 A/D 转换器操作

10.4.1 A/D 转换器基本操作

- <1> 通过使用 A/D 转换器模式寄存器(ADM)的第 5 位至第 1 位(FR2 至 FR0、LV1 和 LV0)设置 A/D 转换时间和操作模式。
- <2> ADM 的第 0 位(ADCE)置“1”，开始电压比较器的操作。
- <3> 通过 A/D 端口配置寄存器设置 A/D 转换的模拟 I/O 通道，并且通过端口模式寄存器 2(PM2)设置输入模式。
- <4> 通过模拟输入通道指定寄存器(ADS)选择 A/D 转换通道。
- <5> ADM 的第 7 位(ADCS)置“1”，启动转换操作。
(通过软件执行<6>至<13>的操作。)
- <6> 通过采样&保持电路，对所选模拟输入通道的输入电压进行采样。
- <7> 经过一定时间的采样后，采样&保持电路处于保持状态，并在 A/D 转换操作结束前一直保持此状态。
- <8> 设置逐次逼近寄存器(SAR)的第 7 位。通过分接选择器将串联电阻字符串的分接电压置为 $(1/2)AV_{REF}$ 。
- <9> 由电压比较器比较串联电阻字符串的分接电压与采样电压。若模拟输入电压高于 $(1/2)AV_{REF}$ ，则 SAR 的 MSB 置“1”；若模拟输入电压低于 $(1/2)AV_{REF}$ ，则 SAR 的 MSB 清零。
- <10> 随后，SAR 的第 6 位自动置“1”，并进入下一个比较操作过程。根据第 9 位的预置值选择串联电阻字符串的分接电压，描述如下。
- 第 7 位 = 1: $(3/4)AV_{REF}$
 - 第 7 位 = 0: $(1/4)AV_{REF}$
- 比较分接电压与采样电压后，用其结果对 SAR 的第 6 位进行以下操作。
- 模拟输入电压 $\geq$ 分接电压: 第 6 位 = 1
 - 模拟输入电压 $<$ 分接电压: 第 6 位 = 0
- <11> 按此方式继续进行比较，直至 SAR 的第 0 位。
- <12> 完成全部 8 位比较后，在 SAR 中保留一个有效数值结果，将结果值传送到 A/D 转换结果寄存器(ADCRH)并锁存。同时，产生 A/D 转换结束中断(INTAD)。
- <13> 重复<6>至<12>步骤，直到 ADCS 清零。
若停止计数操作，则将 ADCS 清零。
- ADCE = 1 状态下，重新启动 A/D 转换操作时，应从步骤<5>开始。ADCE = 0 状态下，重新启动 A/D 转换操作时，设置 ADCE=1，等待至少 1 μ s 后，从步骤<5>开始。若改变 A/D 转换的通道，则从步骤<4>开始。

- 注意事项 1.** 必须确保<1>至<5>的操作时间至少为 1 μ s。
- 2.** 若<2>时序在<4>之前，则可以在任何时间执行<2>。

图 10-10. A/D 转换器的基本操作



连续执行 A/D 转换操作，直到用软件将 A/D 转换器模式寄存器(ADM)的第 7 位(ADCS)复位(0)。

在 A/D 转换期间，若对模拟输入通道指定寄存器(ADS)进行写操作，则转换操作将初始化；若 ADCS 置“1”，则重新开始转换操作。

复位信号产生时，A/D 转换结果寄存器(ADCRH)清为 00H。

10.4.2 输入电压和转换结果

输入到模拟输入引脚(AN10 至 AN14)的模拟输入电压与理论上的 A/D 转换结果(8 位 A/D 转换结果寄存器(ADCRH))之间的关系如下所示。

$$\text{ADCR} = \text{INT} \left(\frac{V_{\text{AIN}}}{AV_{\text{REF}}} \times 256 + 0.5 \right)$$

或

$$(\text{ADCR} - 0.5) \times \frac{AV_{\text{REF}}}{256} \leq V_{\text{AIN}} < (\text{ADCR} + 0.5) \times \frac{AV_{\text{REF}}}{256}$$

INT(): 该函数返回()中值的整数部分。

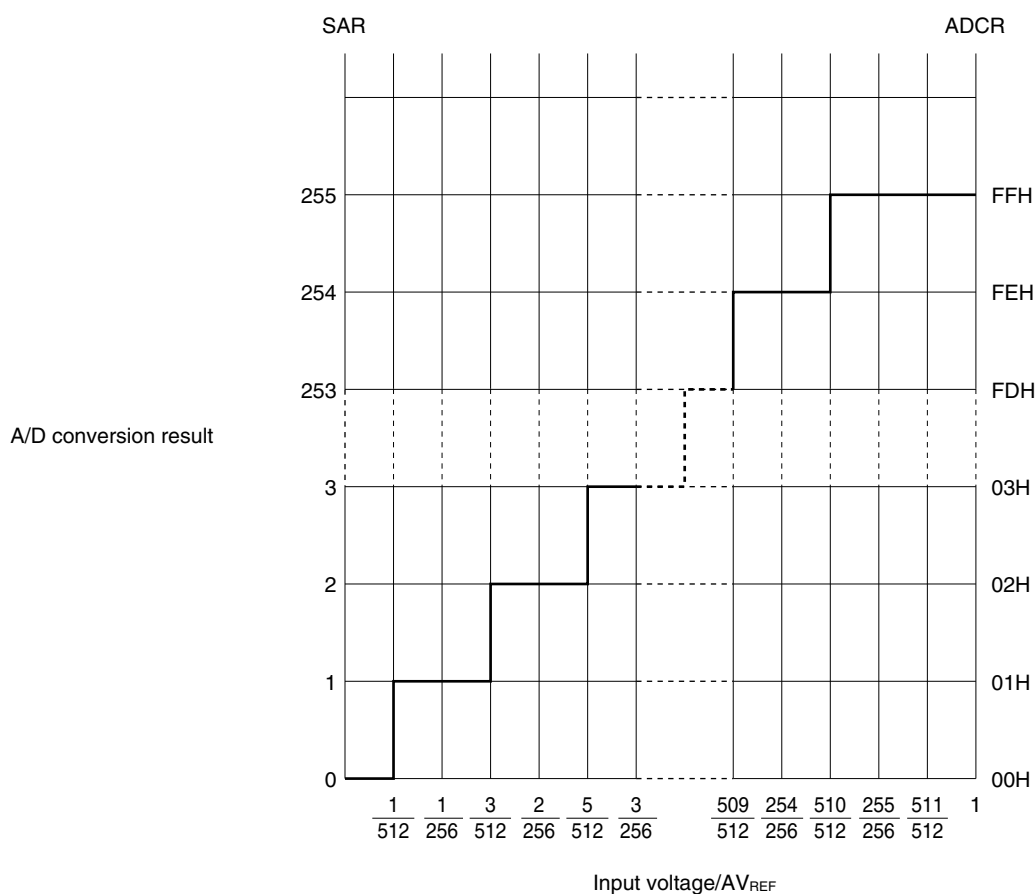
V_{AIN} : 模拟输入电压

AV_{REF} : AV_{REF} 引脚电压

ADCR: 8 位 A/D 转换结果寄存器(ADCRH)值

模拟输入电压和 A/D 转换结果之间的关系如图 10-11 所示。

图 10-11. 模拟输入电压和 A/D 转换结果之间的关系



10.4.3 A/D 转换器操作模式

通过模拟输入通道指定寄存器(ADS)，从 ANI0 至 ANI4 之间选择 1 个模拟输入通道，进行 A/D 转换。

(1) A/D 转换操作

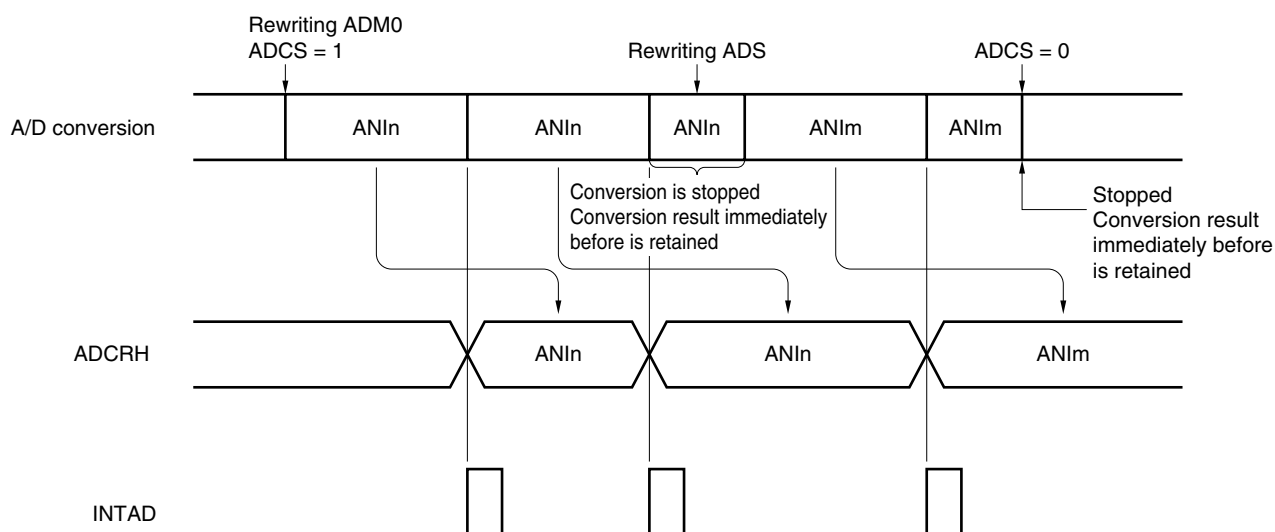
通过将 A/D 转换器模式寄存器(ADM)的第 7 位(ADCS)置“1”，启动由模拟输入通道指定寄存器(ADS)指定的模拟输入引脚外加电压的 A/D 转换操作。

A/D 转换完成时，将 A/D 转换结果存入 A/D 转换结果寄存器(ADCRH)中，并产生中断请求信号(INTAD)。当一个 A/D 转换结束时，将立即开始下一个 A/D 转换操作。

若在转换期间重写 ADS 的值，则停止执行中的 A/D 转换操作，重新开始。

若在 A/D 转换期间 ADCS 置“0”，则 A/D 转换立即停止。此时，保存其停止之前的转换结果。

图 10-12. A/D 转换操作



- 备注 1. n = 0 至 4
2. m = 0 至 4

设置方法如下所述。

- <1> 通过使用 A/D 转换器模式寄存器(ADM)的第 5 位至第 1 位(FR2 至 FR0、LV1 和 LV0)设置 A/D 转换时间和操作模式。
- <<2> ADM 的第 0 位(ADCE))置“1”。
- <3> 通过 A/D 培植寄存器(ADPC)和端口模式寄存器 2(PM2)，设置用于模拟输入的通道。
- <<4> 通过模拟输入通道指定寄存器(ADS)选择通道。
- <<5> ADCS 的第 7 位(ADCE)置“1”，开始 A/D 转换。
- <6> A/D 转换完成后，会产生一个中断请求信号（INTAD）。
- <7> 将 A/D 转换数据传送到 A/D 转换结果寄存器(ADCRH)。
- <改换通道>
 - <8> 中断屏蔽标志寄存器 1L(MK1L)的第 0 位(ADMK)置“1”^注。
 - <9> 通过 ADS 改换通道，启动 A/D 转换。
 - <10> 中断请求标志寄存器 1L(IF1L) 的第 0 位(ADIF)清零。
 - <11> ADMK 清零^注。
 - <12> A/D 转换完成后，会产生一个中断请求信号（INTAD）。
 - <13> 将 A/D 转换数据传送到 A/D 转换结果寄存器(ADCRH)。
- <完成 A/D 转换>
 - <14> ADCS 清零。
 - <15> ADCE 清零。

注 仅限在 A/D 转换中使用中断处理时，执行此项。

- 注意事项**
1. 必须确保<1>至<5>的操作时间至少为 1 μs。
 2. 若<2>时序在<4>之前，则可以在任何时间执行<2>。
 3. 可省略<2>。但在这种情况下，忽略<5>之后的第一个转换结果。
 4. <6>至<12>的期间不同于由 ADM 的第 5 位至第 1 位(FR2 至 FR0, LV1, LV0)设置的转换时间。<9>至<12>的时间是由 FR2 至 FR0、LV1、LV0 设置的转换时间。

10.5 A/D 转换器特征表的阅读方法

以下介绍 A/D 转换器中的专用术语。

(1) 分辨率

可分辨的最小模拟输入电压。即，1 位数字输出的模拟输入电压的百分比，称为 1LSB(最低有效位)。1LSB 的满量程比率表示为%FSR（满度量程）。

分辨率为 8 位时：

$$\begin{aligned} 1\text{LSB} &= 1/2^8 = 1/256 \\ &= 0.391\%\text{FSR} \end{aligned}$$

精确度与分辨率无关，取决于总误差。

2: 总误差

总误差是指实际测量值与理论值之间的最大误差。

零度误差、满度误差、积分线性误差和微分线性误差等合称总误差。

但量化误差不属于特征表中总误差的范围。

(3) 量化误差

当模拟值转换成数字值时，通常会产生 $\pm 1/2\text{LSB}$ 的误差。在 A/D 转换器中，将 $\pm 1/2\text{LSB}$ 范围的模拟输入电压转换成相同的数字代码，因此，量化误差是不可避免。

但量化误差不属于参数表中的总体误差、零误差、满度误差、积分线性误差或微分线性误差的范围。

图 10-13. 总误差

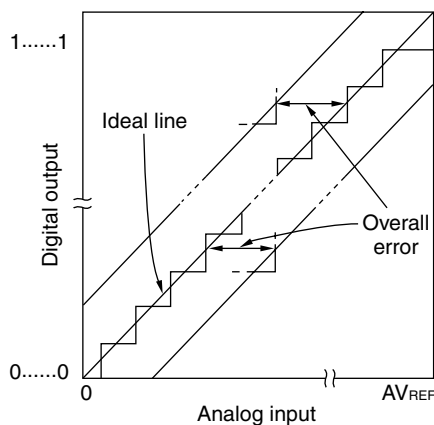
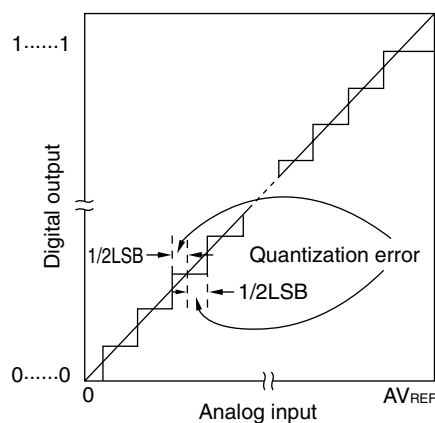


图 10-14. 量化误差



(4) 零误差

当数字输出从 0.....000 改变到 0.....001 时，表示模拟输入电压的实测值与理论值(1/2LSB) 之差。

实测值大于理论值的情况下，数字输出从 0.....001 改变到 0.....010 时，表示模拟输入电压的实测值与理论值(3/2LSB) 之差。

(5) 满度误差

当数字输出从 1.....110 改变到 1.....111 时，模拟输入电压的实测值与理论值(满度- 3/2LSB)之差。

(6) 积分线性误差

表示转换特性偏离理想线性关系的程度。表示当零误差和满量程误差均为 0 时，实测值与其理论值之间的最大误差。

(7) 微分线性误差

当代码输出的理想宽度为 1LSB 时，实测值与理想值之差。

图 10-15. 零误差

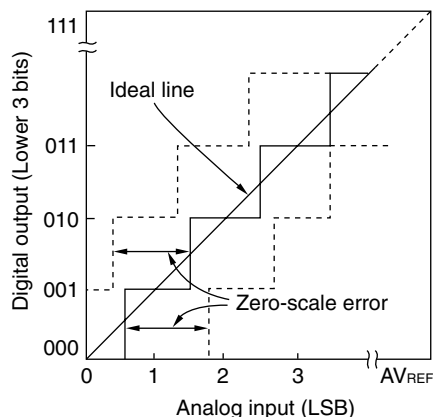


图 10-16. 满度误差

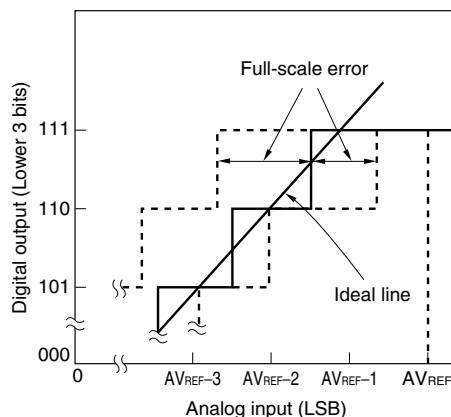


图 10-17. 微分线性误差

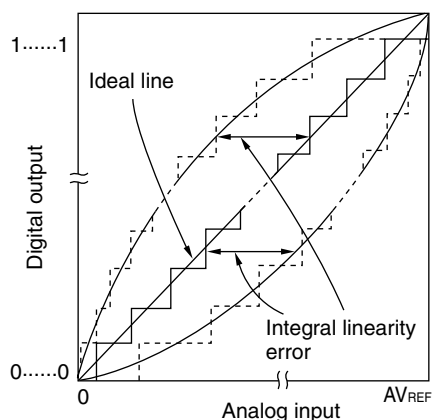
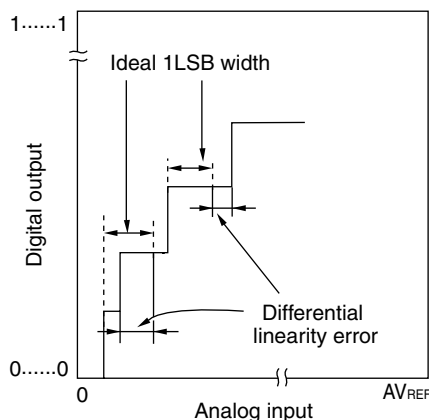


图 10-18. 微分线性误差



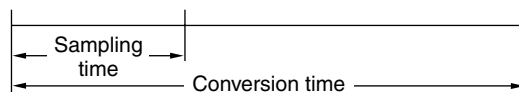
(8) 转换时间

表示从开始采样到获取数字输出所经历的时间。

采样时间属于特征表中的转换时间。

(9) 采样时间

表示通过采样和保持电路，将模拟电压模拟开关打开到对模拟电压进行采样所需的时间。



10.6 使用 A/D 转换器的注意事项

(1) STOP 模式下的操作电流

为了满足 STOP 模式下的电源电流的 DC 特性时，在执行 STOP 指令之前，将 A/D 转换器模式寄存器(ADM)的第 7 位(ADCS)和第 0 位置“1”。

若从待机状态重新启动，则将中断请求标志寄存器 1L(IF1L)的第 0 位(ADIF)清零，并开始操作。

(2) ANI0 至 ANI4 的输入范围

应在额定范围内使用 ANI0 至 ANI4 输入电压。若输入大于 AV_{REF} 或小于 V_{SS} 的电压(即使在绝对最大额定范围之内)，则该通道的转换值将变得不确定。而且还会影响到其他通道的转换值。

(3) 冲突操作

<1> 转换结束后，A/D 转换结果寄存器(ADCRH)的写入与通过指令对 ADCRH 的读取之间的冲突。

优先读取 ADCRH。执行读取操作，将新的转换结果写入 ADCRH。

<2> 转换结束后，ADCRH 写入与 A/D 转换器模式寄存器(ADM)写入、模拟输入通道指定寄存器(ADS)或 A/D 端口配置寄存器(ADPC)写入之间的冲突。

优先写入 ADM、ADS 和 ADPC。不执行 ADCRH 写入，而且不产生转换结束中断信号(INTAD)。

(4) 噪声解决对策

为了保持 8 位分辨率，必须注意输入到 AV_{REF} 引脚和 ANI0 至 ANI4 引脚的噪声。

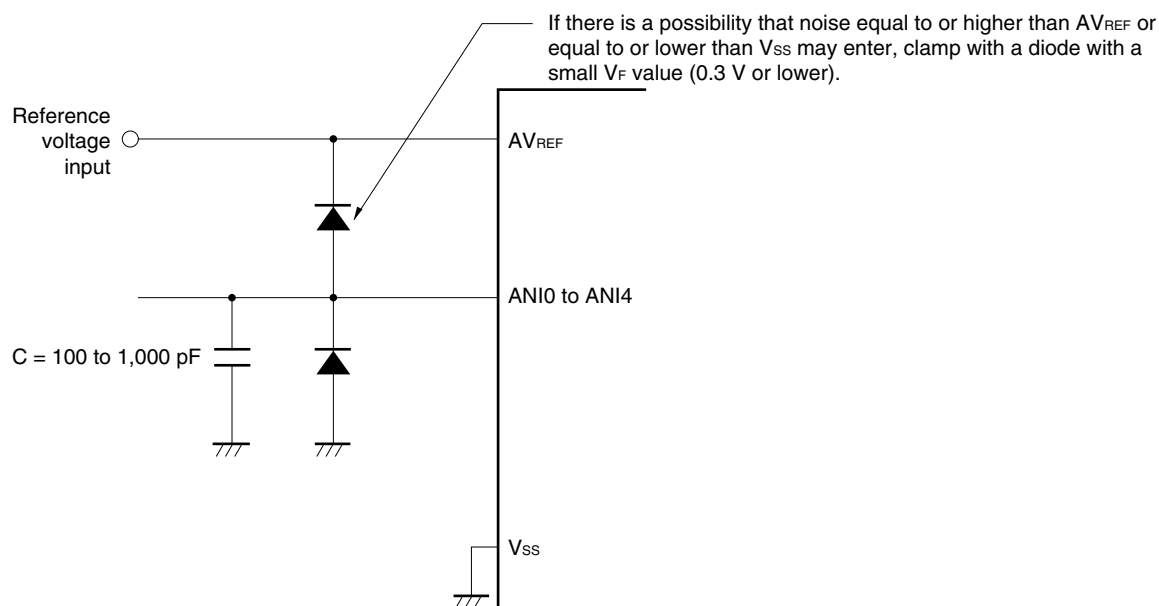
<1> 电源连接低等效电阻和优质频率相应的电容。

<2> 模拟输入源的输出阻抗越高，影响就越大。为了减小噪声，建议连接如图 10-19 所示的外部 C。

<3> 转换期间不切换其他引脚。

<4> 在转换开始后立即设置 HALT 模式，则可以改善精度。

图 10-19. 模拟输入引脚的连接



(5) ANI0/P20 至 ANI4/P24 引脚

- <1> 模拟输入引脚(ANI0 至 ANI4)也可用作数字 I/O 端口引脚(P20 至 P24)。选择 ANI0 至 ANI4 中的任意一个通道执行 A/D 转换时, 转换过程期间禁止访问 P20 至 P27; 否则, 会降低转换分辨率。
- <2> ANI0/P20 至 ANI4/P24 引脚用作数字 I/O 端口时, 建议从离 AV_{REF} (ANI4/P24 引脚)最远的引脚开始。建议模拟输入从离 V_{SS} (ANI0/P20 引脚)最近的引脚开始。
- <3> 若在 A/D 转换中的引脚相邻的引脚外加数字脉冲, 则由于噪声耦合, 有可能得不到预期的 A/D 转换值。因此, 不要在进行 A/D 转换引脚的邻近引脚上施加脉冲。

(6) ANI0 至 ANI4 引脚输入阻抗

采样期间, A/D 转换器对内部采样电容进行充电后进行采样。

因此, 不进行采样时仅有漏电流, 而在采样期间则时电容充电的电流。由此得出, 输入电阻在采样中和其他状态下进行变动。

为了使采样有效, 建议模拟输入源的输出阻抗小于等于 $10k\Omega$, 并将 $100pF$ 左右的电容连接到 ANI0 至 ANI4 的引脚上 (参阅图 10-19)。

(7) AV_{REF} 引脚输入抗租

AV_{REF} 和 V_{SS} 的引脚之间连接几十 $k\Omega$ 的串联电阻字符串。

因此, 参考电压源的输出阻抗高时, 串联连接到 AV_{REF} 与 AV_{SS} 引脚之间的串联电阻字符串, 会导致较大的参考电压误差。

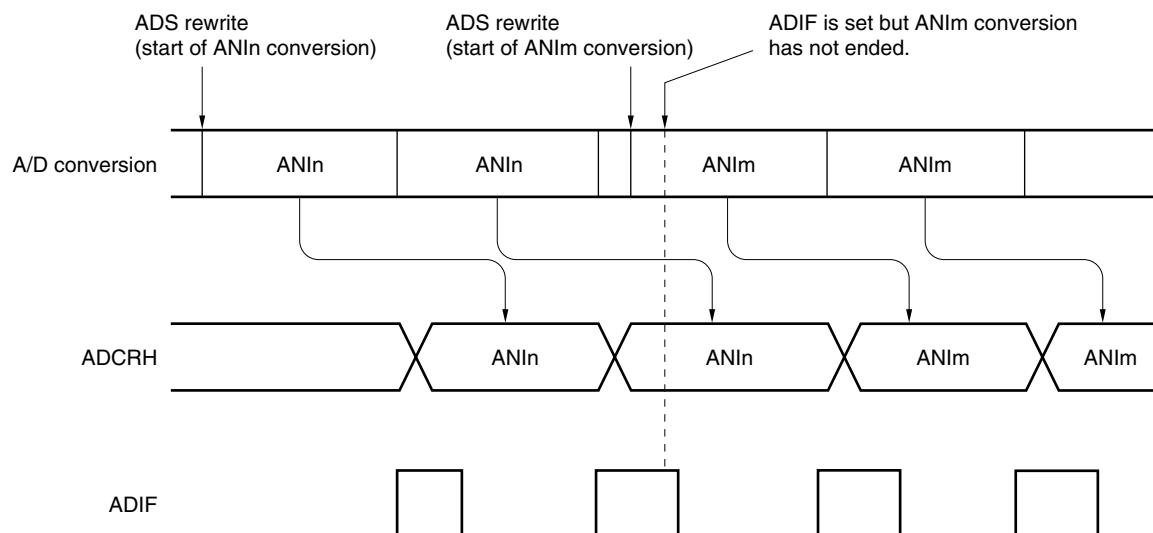
(8) 中断请求标志 (ADIF)

即使改变模拟输入通道指定寄存器(ADS), 中断请求标志 (ADIF) 也不会清零。

因此, 若 A/D 转换期间模拟输入引脚发生变化, 则在 ADS 重写之前, 模拟输入通道的 A/D 转换结果和 ADIF 有可能已被设置。需要注意的是, 若 ADS 重写后立即读取 ADIF, 则重写模拟输入的 A/D 转换尚未结束前, ADIF 可能被设置。

一旦 A/D 停止转换后又重新开始时, 在启动前先将 ADIF 清零。

图 10-20. A/D 转换结束中断请求的发生时序



备注 1. $n = 0$ 至 4

2. $m = 0$ 至 4

(9) A/D 转换刚开始后的转换结果

若在 ADCE 置“1”后, 在 $1\mu\text{s}$ 内将 ADCS 置“1”, 或者 ADCE=0 时 ADCS 置“1”, 则 A/D 转换开始后的 A/D 转换值可能不在额定范围内。可采取悬挂 A/D 转换结束中断请求(INTAD), 并删除第 1 次转换结果的措施。

(10) A/D 转换结果寄存器(ADCRH)的读取操作

对 A/D 转换器模式寄存器(ADM)、模拟输入频道指定寄存器(ADS)和 A/D 端口配置寄存器(ADPC)进行写入操作时, ADCRH 的内容可能不确定。转换结束后读取转换结果, 赋值给 ADM、ADS 和 ADPC。若没有使用上述时序, 则有可能读取不正确的转换结果。

<11> 间隔等效电路

模拟输入模块的等效 电路如下所示。

图 10-21. ANIn 引脚的内部等效电路

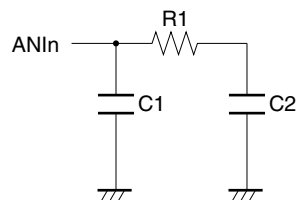


表 10-6. 等效电路的电阻和电容（参考值）

AV_{REF}	R1	C1	C2
$2.7\text{ V} \leq AV_{REF} \leq 5.5\text{ V}$	11.5 kΩ	8.0 pF	8.0 pF

第十一章 运算放大器

11.1 运算放大器功能

μPD79F7023, 79F7024 配置运算放大器 0 和 1。

配有 2 个输入引脚 (AMPn-引脚和 AMPn+引脚)和 1 个输出引脚(AMPnOUT 引脚), 可用于与外部相联接的单电源放大器。

AMP0OUT 引脚可兼作 A/D 转换器的模拟输入引脚, 因此, 增幅电压可用作 A/D 转换器的模拟输入。

备注 n = 0, 1

11.2 运算放大器配置

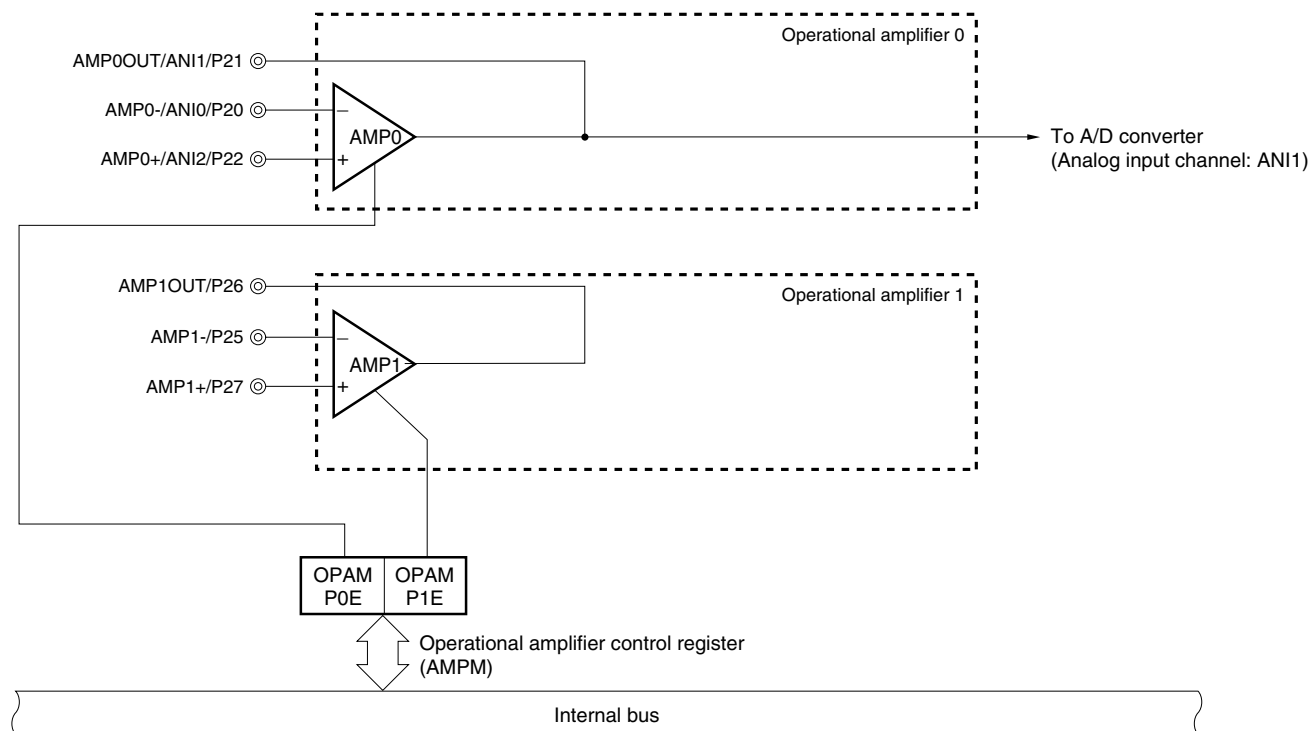
运算放大器包括以下硬件。

表 11-1. 运算放大器配置

项目	配置
运算放大器输入	AMPn-引脚, AMPn+引脚
运算放大器输出	AMPnOUT引脚
控制寄存器	运算放大器控制寄存器(AMPM) A/D配置寄存器(ADPC) 模拟输入通道指定寄存器(ADS) 端口模式寄存器 2(PM2)

备注 n = 0, 1

图 11-1. 运算放大器的框图



11.3 运算放大器使用的寄存器

运算放大器使用以下 4 个寄存器。

- 运算放大器控制寄存器(AMPM)
- A/D 端口配置寄存器(ADPC)
- 模拟输入通道指定寄存器(ADS)
- 端口模式寄存器 2(PM2)

(1) 运算放大器控制寄存器(AMPM)

控制运算放大器 0 和 1 的操作。

可用 1 位或 8 位存储器操作指令设置 AMPM。

复位信号产生时，该寄存器清为 00H。

图 11-2. 运算放大器控制寄存器(AMPM)的格式

地址: FF60H 复位后: 00H R/W

符号	<7>	6	5	4	<3>	2	1	0
AMPM	OPAMP0E	0	0	0	OPAMP1E	0	0	0

OPAMP0E	运算放大器 0 操作控制
0	停止运算放大器 0 操作
1	允许运算放大器 0 操作

注意事项 使用运算放大器 0 时，将不用于运算放大器 0 的端口 2 引脚用作数字输入的情况下，为了防止 A/D 转换分辨率下降，应固定数字输入端口的输入电平。

OPAMP1E	运算放大器 1 操作控制
0	停止运算放大器 1 操作
1	允许运算放大器 1 操作

(2) A/D 端口配置寄存器(ADPC)

ADPC 将 P20/AMP0-/ANI0 至 P24/AMP4 之间的引脚切换到数字 I/O 端口或模拟 I/O 端口。可按 1 位单元指定 ADPC 每个位相对应的端口 2 的一个引脚。

复位信号产生时, ADPC 清为 00H。

图 11-3. A/D 端口配置寄存器(ADPC)

地址: FF97H 复位后:00H R/W

符号	7	6	5	4	3	2	1	0
ADPC	ADPC7	ADPC6	ADPC5	ADPC4	ADPC3	ADPC2	ADPC1	ADPC0

ADPCn	数字 I/O 或模拟 I/O 选择(n = 0 至 7)
0	模拟 I/O
1	数字 I/O

- 注意事项**
1. 通过端口模式寄存器 2(PM2), 将设置为模拟 I/O 的引脚选为输入模式。
 2. 若将数据写入 ADPC,则产生一个等待周期。当外部硬件时钟停止时, 禁止将数据写入 ADPC。详阅第二十六章 等待注意事项。

(3) 模拟输入通道指定寄存器(ADS)

该寄存器指定模拟电压进行 A/D 转换的输入通道。

可用 1 位或 8 位存储器操作指令设置 ADS。

复位信号产生时，该寄存器清为 00H。

图 11-4. 模拟输入通道指定寄存器(ADS)的格式

地址: FF0EH 复位后: 00H R/W

符号	7	6	5	4	3	<2>	<1>	<0>
ADS	0	0	0	0	0	ADS2	ADS1	ADS0

ADS2	ADS1	ADS0	模拟输入通道	输入源
0	0	0	ANI0	P20/ANI0引脚
0	0	1	ANI1	P21/ANI1引脚或 运算放大器0 输出信号
0	1	0	ANI2	P22/ANI2引脚
0	1	1	ANI3	P23/ANI3引脚
1	0	0	ANI4	P24/ANI4引脚
上述除外的部分			禁止设置	

注意事项 1. 确保第 3 位至第 7 位清零。

2. 通过端口模式寄存器 2(PM2)的输入模式下，选择 A/D 转换使用的通道。

(4) 端口模式寄存器 2(PM2)

运算放大器 0 使用AMP0-/ANI0/P20、AMP0OUT/ANI1/P21 和AMP0+/ANI2/P22 的引脚时，将PM20 至PM22 分别置“1”。

运算放大器 1 使用AMP1-/P25、AMP1OUT/P26 和AMP1+/P27 的引脚时，将PM25 至PM27 分别置“1”。

此时，P20 至 P22 和 P25 至 P27 的输出锁存即可为“0”也可为“1”。

若 PM20 至 PM22 和 PM25 至 PM27 分别置“0”，则禁止作运算放大器 0 和 1 的引脚。

可用 1 位或 8 位存储器操作指令设置 PM2。

复位信号产生时，该寄存器设置为 FFH。

图 11-5. 端口模式寄存器 2(PM2)的格式

地址: FF22H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PM2	PM27	PM26	PM25	PM24	PM23	PM22	PM21	PM20

PM2n	P2n 引脚 I/O 模式选择 (n = 0 至 7)
0	输出模式 (输出缓冲打开)
1	输入模式 (输出缓冲关闭)

使用 P20/ANI0/AMP0-, P21/ANI1/AMP0OUT 和 P22/ANI2/AMP0+时, 根据使用的引脚功能设置寄存器。(参阅表 11-2 和 11-3)。

表 11-2. P20/ANI0/AMP0-和 P22/ANI2/AMP0+引脚设置功能

ADPC寄存器	PM2寄存器	OPAMP0E位	ADS 寄存器 (n = 0, 2)	P20/ANI0/AMP0-和 P22/ANI2/AMP0+引脚
模拟输入选择	输入模式	0	选择ANIn	模拟输入 (转换为数字模式)
			禁止选择ANIn	模拟输入 (禁止转换为数字模式)
		1	选择ANIn	禁止设置
			禁止选择ANIn	运算放大器输入
	输出模式	—	—	禁止设置
数字I/O选择	输入模式	—	选择ANIn	禁止设置
			禁止选择ANIn	数字输入
	输出模式	—	选择ANIn	禁止设置
			禁止选择ANIn	数字输出

表 11-3. P21/ANI1/AMP0OUT 引脚设置功能

ADPC寄存器	PM2寄存器	OPAMP0E位	ADS寄存器	P21/ANI1/AMP0OUT引脚
模拟I/O选择	输入模式	0	选择ANI1	模拟输入 (转换为数字模式)
			禁止选择ANI1	模拟输入 (禁止转换为数字模式)
		1	选择ANI1	运算放大器输出 (转换为数字模式)
			禁止选择ANI1	运算放大器 0输出 (转换为数字模式)
	输出模式	—	—	禁止设置
数字I/O选择	输入模式	0	选择ANI1	禁止设置
			禁止选择ANI1	数字输入
		1	—	禁止设置
			—	禁止设置
	输出模式	0	选择ANI1	禁止设置
			禁止选择ANI1.	数字输出
		1	—	禁止设置

备注 ADPC: A/D 端口配置寄存器
 PM2: 端口模式寄存器 2
 OPAMP0E: 运算放大器控制寄存器(AMPM)的第 7 位
 ADS: 模拟输入通道指定寄存器

11.4 运算放大器操作

配有 2 个输入引脚 AMPn-引脚和 AMPn+引脚)和 1 个输出引脚(AMPnOUT 引脚)，可用作与外部相联接的单电源放大器。

AMP0OUT 引脚兼备 A/D 转换器的模拟输入引脚，因此，增幅电压可用作 A/D 转换器的模拟输入。

开始操作步骤如下所述。

- <1> 通过 ADPC 寄存器将使用的引脚(AMPn-, AMPn+, AMPnOUT)设置为 I/O。
- <2> 通过 PM2 寄存器将使用的引脚 (AMPn-, AMPn+, AMPnOUT) 设置为输入模式。
- <3> OPAMPnE 位置“1”，并允许操作。

注意事项 增幅电压用作 A/D 转换器的输入时，通过 ADS 寄存器选择模拟输入通道之前，允许操作。

备注 n = 0, 1

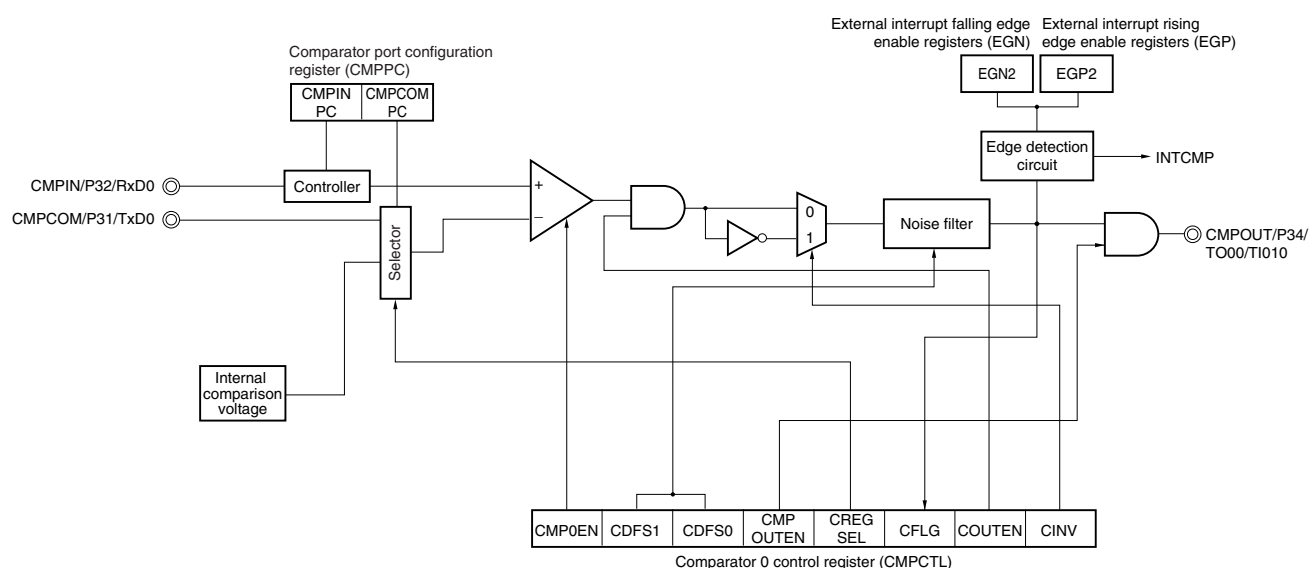
第十二章 比较器

12.1 比较器功能

比较器具有以下功能。

- 可选用以下参考电压。
 - <1> 内部参考电压
 - <2> 比较器公共引脚(CMPCOM)的输入电压
- 通过检测出比较器输出的有效沿，可产生中断信号。可通过 EGP2 和 EGN2 位设置有效沿(参阅第十四章 中断功能)。
- 可选择噪声消除数字过滤器消除宽度。

图 12-1. 比较器的框图



12.2 比较器配置

比较器包括以下硬件。

表 12-1. 比较器配置

项目	配置
控制寄存器	比较器控制寄存器(CMPCTL) 比较器端口配置寄存器(CMPPC) 端口模式寄存器 3(PM3) 端口寄存器 3 (P3)

12.3 寄存器控制比较器

比较器采用以下 3 个寄存器。

- 比较器控制寄存器(CMPCTL)
- 比较器端口配置寄存器(CMPPEC)
- 端口模式寄存器 3PM3)

(1) 比较器控制寄存器(CMPCTL)

CMPCTL 用于设置比较器的操作控制、允许/禁止比较器输出、反转输出、噪声消除宽度和参考电压。

可用 1 位或 8 位存储器操作指令设置 CMPCTL。

复位信号产生时, CMPCTL 清为 00H。

图 12-2.比较器控制寄存器(CMPCTL)的格式 (1/2)

地址: FF61H 复位后: 00H R/W

符号	<7>	<6>	<5>	<4>	<3>	2	<1>	<0>
CMPCTL	CMP0EN	CDFS1	CDFS0	CMPOUTEN	CREGSEL	CFLG	COUTEN	CINV
CMP0EN		比较器0操作控制						
0		停止操作						
1		允许操作 允许输入比较器0的+侧外部引脚						
CDFS1	CDFS0	噪声消除宽度设置						
0	0	不使用噪声过滤器						
0	1	$2/F_{prs}$						
1	0	$2^2/F_{prs}$						
1	1	$2^3/F_{prs}$						
CMPOUTEN		比较器输出引脚控制						
0		禁止比较器输出的外部引脚输出						
1		允许比较器输出的外部引脚输出						
CREGSEL		参考电压选择						
0		外部参考电压CMPCOM						
1		内部参考电压(稳压器电压:参阅第十九章 稳压器)						
CFLG		比较器输出标志监视器						
0		比较器输出值为“0”						
1		比较器输出值为“1”						
COE		允许或禁止比较器输出						
0		禁止输出(输出信号=固定低电平)						
1		允许输出						

图 12-2. 比较器控制寄存器(CMPCTL)的格式 (2/2)

CINV	反向输出设置
0	正向
1	反向

- 注意事项**
1. 设置比较器操作为禁止状态(CMP0EN = 0)后,重写 CDFS1、CDFS0、CMPOUTEN、CREGSEL、COUTEN、CINV。
 2. 噪声消除宽度与设定值相比,有可能会多删除外围硬件时钟频率(fPRS)的一个时钟。
 3. 比较器输出噪声的间隔设置为“设置的噪声消除宽度 + 1 个时钟”时,可能会输出无效波形。

备注 fPRS: 外围硬件时钟频率

(2) 比较器端口配置寄存器(CMPPC)

CMPPC 将比较器输入引脚切换到数字 I/O 或模拟输入。

可用 1 位或 8 位存储器操作指令设置 CMPPC。

复位信号产生时, CMPPC 清为 00H。

图 12-3. 比较器输出标志寄存器(CMPPC)格式

地址: FF62H 复位后: 00H R/W

符号	7	6	5	4	3	2	<1>	<0>
CMPPC	0	0	0	0	0	0	CMPCOM PC	CMPINPC

CMPCOM PC	P31/TxD0/CMPCOM的模拟(A)或数字(D)选择
0	模拟(A)(默认)
1	数字(D)

CMPINPC	P32/RxD0/CMPIN的模拟(A)或数字(D)选择
0	模拟(A)(默认)
1	数字(D)

(3) 端口模式寄存器 3(PM3)

PM3 用来按 1 位单元设置端口 3 输入或输出。

CMPIN/P32/RxD0 引脚和 CMPCOM/P31/TxD0 引脚分别用作比较器引脚和比较器公共输入时，PM31 和 PM32 位分别置“1”。

此时，P31 和 P32 的输出锁存即可为“0”也可为“1”。

可用 1 位或 8 位存储器操作指令设置 PM3。

复位信号产生时，PM3 设置为 FFH。

图 12-4. 端口模式寄存器 3(PM3)的格式

地址: FF23H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PM3	1	1	1	PM34	PM33	PM32	PM31	PM30

注意事项 确保 PM3 的第 5 位至第 7 位设置为“1”。

PM3n	P3n 引脚 I/O 模式选择(n = 0 至 4)
0	输出模式 (输出缓冲打开)
1	输入模式 (输出缓冲关闭)

注意事项 正在使用内部参考电压时，在输入模式下 CMPCOM 引脚可用作端口功能。但禁止在输出模式下使用。并且，禁止访问端口寄存器 3(P3)。

使用 P32/RxD0/CMPIN 和 P31/TxD0/CMPCOM 时, 根据使用的引脚功能设置寄存器(参阅表 12-2 至 12-3)。

表 12-2. P32/RxD0/CMPIN 引脚设置功能

CMPPC 寄存器	PM3 寄存器	CMP0EN 位	P32/RxD0/CMPIN 引脚
数字 I/O 选择	输入模式	0	数字输入
		1	禁止设置
	输出模式	0	数字输出
		1	禁止设置
模拟输入选择	输入模式	0	比较器输入 (禁止)
		1	比较器输入 (允许)
	输出模式	—	禁止设置

备注 CMPPC: 比较器端口配置寄存器
 PM3: 端口模式寄存器 3
 CMP0EN: 比较器控制寄存器第 7 位(CMPCTL)
 ADS: 模拟输入通道指定寄存器

表 12-3.P31/TxD0/CMPCOM 引脚设置功能

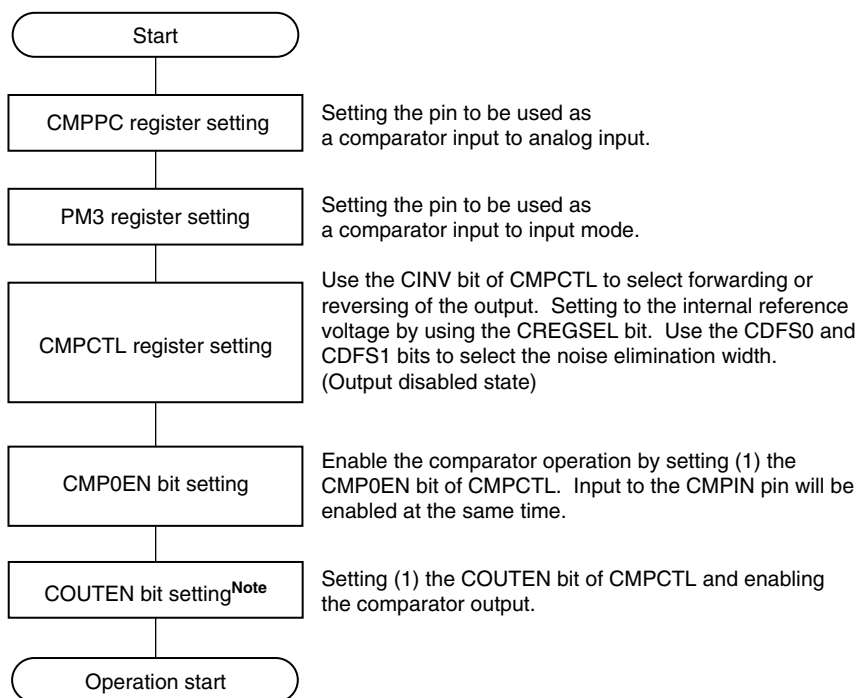
CMPPC 寄存器	PM3 寄存器	CMP0EN 位	CREGSEL 位	P31/TxD0/CMPCOM 引脚
数字 I/O 选择	输入模式	0	0	数字输入
			1	
		1	0	禁止设置
			1	数字输入/比较器公共输入 (禁止)
	输出模式	0	0	数字输出
			1	
		1	0	禁止设置
			1	数字输出/比较器公共输入 (禁止)
模拟输入选择	输入模式	0	0	比较器公共输入 (禁止)
			1	
		1	0	比较器公共输入 (禁止)
			1	比较器公共输入 (允许)
	输出模式	—		禁止设置

备注 CMPPC: 比较器端口配置寄存器
 PM3: 端口模式寄存器 3
 CMP0EN: 比较器控制寄存器第 7 位(CMPCTL)
 ADS: 模拟输入通道指定寄存器

12.4 比较器操作

12.4.1 启动比较器操作(参考电压使用内部参考电压)

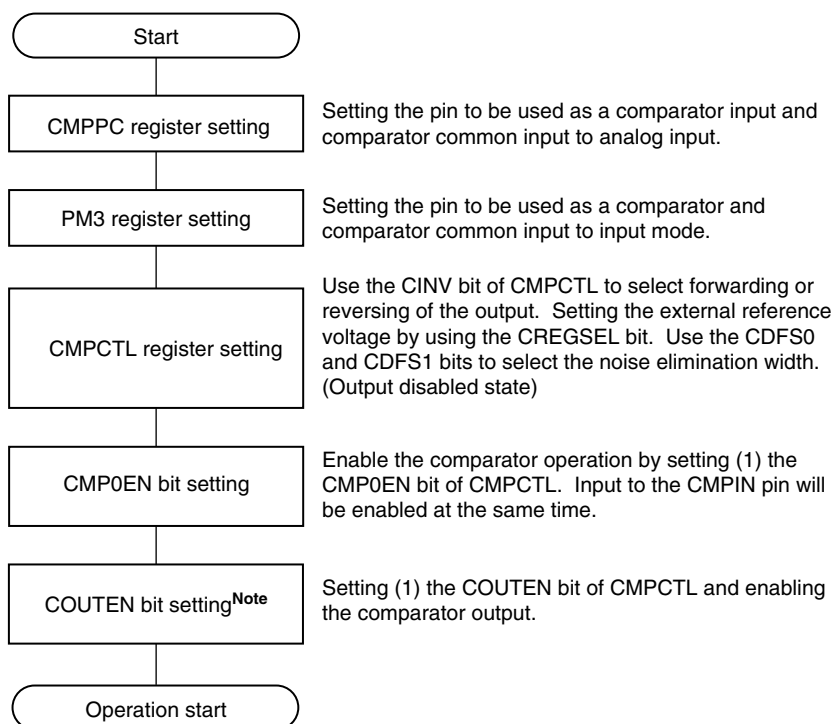
图 12-5. 启动比较器动作的设置步骤示例
(内部参考电压用作参考电压)



注 设置 CREGSEL 位后，应至少经过 20 μs 后进行设置。

12.4.2 启动比较器操作 (参考电压使用 **CMPCOM** 引脚输入电压)

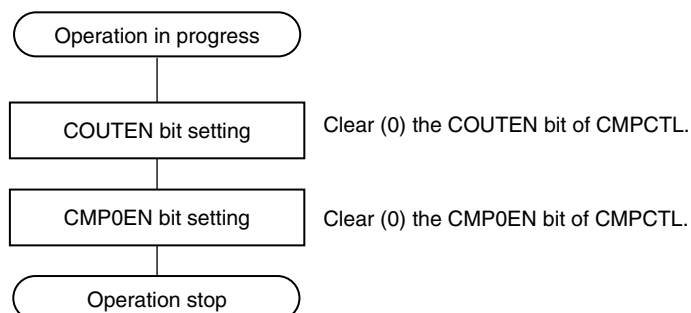
图 12-6. 启动比较器动作的设置步骤示例
(参考电压使用比较器公共引脚(**CMPCOM**)的输入电压)



注 设置 **CMPEN** 位后，应至少经过 1 μs 后将 **COE** 置“1”。

12.4.3 停止比较器操作

图 12-7. 停止比较器动作的设置步骤示例



第十三章 串行接口 UART0

13.1 串行接口 UART0 功能

串行接口 UART0 具有以下两种模式。

(1) 操作停止模式

不执行串行通信时使用该模式，从而降低功耗。

详情参阅 **13.4.1 操作停止模式**。

(2) 异步串行接口(UART)模式

该功能简述如下。

详情参阅 **13.4.2 异步串行接口(UART)模式**和 **13.4.3 专用波特率发生器**。

- 最大传送速率：625 kbps
- 2 引脚配置 TxD0:传送数据输出引脚
RxD0:接收数据输入引脚
- 可选 7 或 8 位的通讯数据长度。
- 通过内置的专用 5 位波特率发生器，可设置任意波特率。
- 可以单独操作发射机和接收机 (全双工操作)。
- 固定为 LSB (最低有效位) 的通信。

- 注意事项**
1. 若串行接口 **UART0** 使用的时钟没有停止(例如,HALT 模式下),则可继续正常操作.若串行接口 **UART0** 使用的时钟停止(例如,STOP 模式下),则各寄存器立即保存时钟停止前的数据后停止操作。TxD0 引脚也同样将时钟停止之前的数据立即保存并输出。但不能保证在时钟重新启动后的正常操作。因此，必须通过电路复位，使 **POWER0 = 0**、**RXE0 = 0** 和 **TXE0 = 0**。
 2. 通过先设置 **POWER0 = 1**，再设置 **TXE0 = 1** (传送)或 **RXE0 = 1** (接收)后，开始通信。
 3. 可通过 **BRGC0** 设置(**f_{XCLK0}**)，使 **TXE0** 与 **RXE0** 同步。为了重新开始传送或接收，应在 **TXE0** 或 **RXE0** 清零后，至少要等待 2 个基本时钟后，再将 **TXE0** 或 **RXE0** 置“1”。若在 2 个基本时钟内设置 **TXE0** 或 **RXE0**，则 **TXE0** 或 **RXE0** 有可能不被初始化。
 4. 设置 **TXE0 = 1** 后，应至少等待 1 个基本时钟(**f_{XCLK0}**)，才能将数据传送到 **TXS0**。

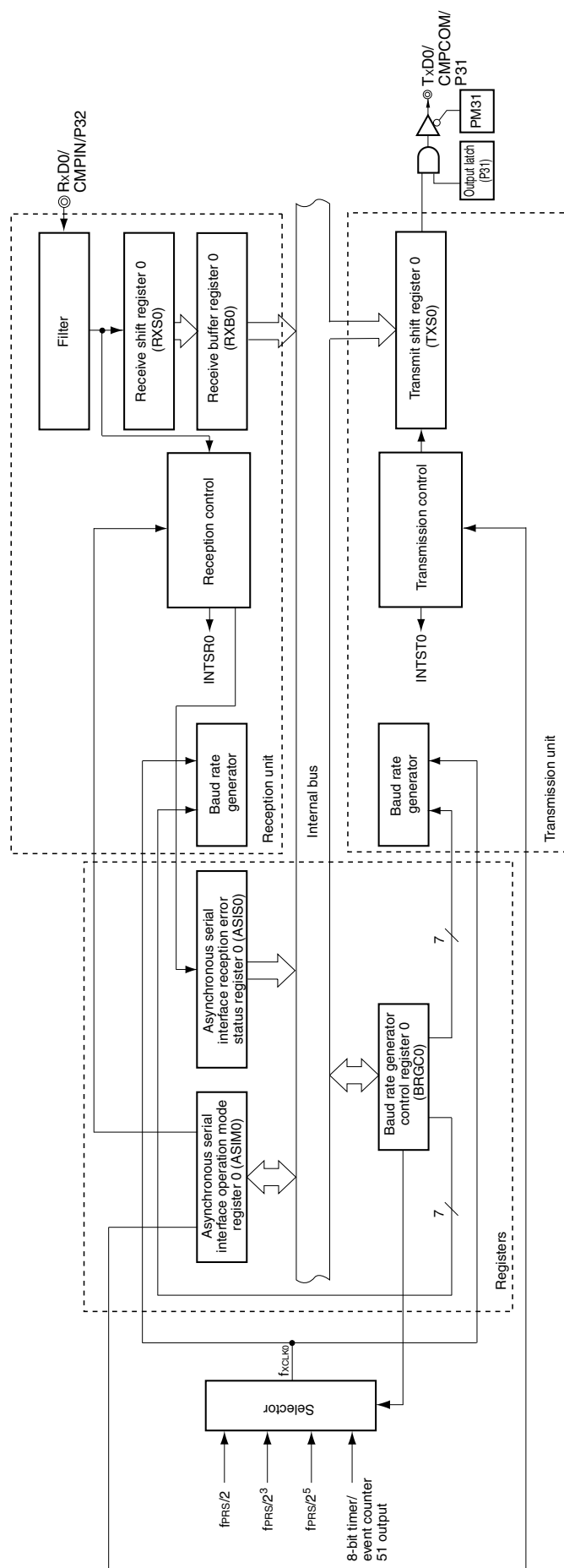
13.2 串行接口 UART0 配置

串行接口 UART0 包括以下硬件。

表 13-1. 串行接口 UART0 配置

项目	配置
寄存器	接收缓冲寄存器 0(RXB0) 接收移位寄存器 0(RXS0) 传送移位寄存器 0(TXS0)
控制寄存器	异步串行接口操作模式寄存器 0(ASIM0) 异步串行接口接收错误状态寄存器 0(ASIS0) 波特率生成器控制寄存器 0(BRGC0) 端口模式寄存器 3(PM3) 端口寄存器 3(P3)

图 13-1. 串行接口 UART0 的框图



(1) 接收缓冲寄存器 0(RXB0)

该 8 位寄存器用于储存通过接收移位寄存器 0(RXS0)转换的并行数据。

每次接收到 1 字节的数据时，新的接收数据就从接收移位寄存器 0(RXS0)传送到该寄存器。

若数据宽度置“1”，则将接收的数据传送到 RXB0 的第 0 位至第 6 位，RXB0 的 MSB 恒为“0”。

若产生溢出错误(OVE0)，则接收的数据不传送给 RXB0。

可按 8 位存储器操作指令读取 RXB0。该寄存器禁止写入。

产生复位信号并且 POWER0 = 0 时，该寄存器设置为 FFH。

(2) 接收移位寄存器 0(RXS0)

该寄存器将输入到 RxD0 引脚的串行数据转换成平行数据。

不能通过程序直接操作 RXS0。

(3) 传送移位寄存器 0(TXS0)

该寄存器用于设置传送数据。通过将数据写入 TXS0，启动发射操作，并从 TxD0 引脚传送串行数据。可按 8 位存储器操作指令写入 TXS0。该寄存器禁止读取。

复位信号产生并且 POWER0 = 0 和 TXE0 = 0 时，该寄存器设置为 FFH。

注意事项 1. 设置 TXE0 = 1 后，应至少等待 1 个基本时钟(f_{CLK0})，才能将数据传送到 TXS0。

2. 传送完成中断信号(INTST0)产生之前，禁止写入下一条传送数据。

13.3 寄存器控制串行接口 UART0

通过以下 5 个寄存器控制串行接口 UART0。

- 异步串行接口操作模式寄存器 0(ASIM0)
- 异步串行接口接收错误状态寄存器 0 (ASIS0)
- 波特率发生器控制寄存器 0(BRGC0)
- 端口模式寄存器 3(PM3)
- 端口寄存器 3 (P3)

(1) 异步串行接口操作模式寄存器 0(ASIM0)

该 8 位寄存器用于控制串行接口 UART0 的串行通信操作。

可用 1 位或 8 位存储器操作指令设置该寄存器。

复位信号产生时，该寄存器设置为 01H。

图 13-2. 异步串行接口操作模式寄存器 0(ASIM0)格式 (1/2)

地址: FF50H 复位后: 01H R/W

符号	<7>	<6>	<5>	4	3	2	1	0
ASIM0	POWER0	TXE0	RXE0	PS01	PS00	CL0	SL0	1

POWER0	允许/禁止操作内部运行时钟
0 ^{#1}	禁止内部操作时钟的操作（时钟恒为低电平），并异步复位内部电路 ^{#2} 。
1	允许操作内部运行时钟

TXE0	允许/禁止传送
0	禁止传送（同步复位传送电路）
1	允许传送

RXE0	允许/禁止接收
0	禁止接收（同步复位接收电路）
1	允许接收。

- 注 1. POWER0 = 0 时，从 RxD0 引脚的输入恒为高电平。
2. 将异步串行接口接收错误状态寄存器 0(ASIS0)、传送移位寄存器 0(TXS0)和接收缓冲寄存器 0(RXB0)复位。

图 13-2. 异步串行接口操作模式寄存器 0(ASIM0)格式 (2/2)

PS01	PS00	传送操作	接收操作
0	0	禁止输出奇偶校验位。	无奇偶校验时接收
0	1	输出 0 奇偶校验。	作为奇偶校验接收 ^注
1	0	输出奇校验。	判断为奇校验。
1	1	输出偶校验。	判断为偶校验。

CL0	指定传送/接收数据的字符长度
0	数据的字符长度 = 7 位
1	数据的字符长度 = 8 位

SL0	指定传送数据的停止位编码
0	停止位编码 = 1
1	停止位编码 = 2

注 若设置“作为 0 奇偶校验接收”，则无法进行奇偶校验。因此，由于没有设置异步串行接口接收错误状态寄存器 0(ASIS0) 的第 2 位(PE0)，不会发生错误中断。

- 注意事项
1. 开始传送时,先设置 **POWER0** 为“1”后,再设置 **TXE0** 为“1”。停止传送时,先设置 **TXE0** 为“0”后,再设置 **POWER0** 为“0”。
 2. 开始接收时,先设置 **POWER0** 为“1”后,再设置 **RXE0** 为“1”。停止接收时,先设置 **RXE0** 为“0”后,再设置 **POWER0** 为“0”。
 3. 将高电平输入 **RxD0** 引脚期间,先设置 **POWER0** 为“1”后,再设置 **RXE0** 为“1”。若低电平输入期间将 **POWER0** 和 **RXE0** 按先后顺序置“1”,则会开始接收。
 4. 可通过 **BRGC0** 设置(**fxCLK0**) , 使 **TXE0** 与 **RXE0** 同步。为了重新开始传送或接收,应在 **TXE0** 或 **RXE0** 清零后,至少要等待 2 个基本时钟后,再将 **TXE0** 或 **RXE0** 置“1”。若在 2 个基本时钟内设置 **TXE0** 或 **RXE0**, 则 **TXE0** 或 **RXE0** 有可能不被初始化。
 5. 设置 **TXE0** = 1 后,应至少等待 1 个基本时钟(**fxCLK0**)才能传送数据到 **TXS0**。
 6. 重入 **PS01**、**PS00** 和 **CL0** 之前, **TXE0** 和 **RXE0** 清零。
 7. 重写 **SL0** 位时,应确保 **TXE0** = 0。因为始终以“停止位的个数= 1”的格式执行接收操作,所以不受 **SL0** 设定值的影响。
 8. 确保第 0 位置“1”。

(2) 异步串行接口接收错误状态寄存器 0(ASIS0)

该寄存器用于指定串行接口 **UART0** 完成接收时的错误状态。包括 3 位的错误标志(PE0, FE0, OVE0)。

可用 8 位存储器操作指令读取该寄存器。

通过复位信号产生、ASIM0 的第 7 位(POWER0)和第 5 位(RXE0)分别清零, 使该寄存器清零。读取该寄存器时, 读取值为“00H”。若产生接收错误, 则读取 ASIS0 后, 读取接收缓冲寄存器 0(RXB0)并清除错误标志。

图 13-3. 异步串行接口接收错误状态寄存器 0(ASIS0)的格式

地址: FF53H 复位后: 00H R

符号	7	6	5	4	3	2	1	0
ASIS0	0	0	0	0	0	PE0	FE0	OVE0

PE0	显示奇偶校验错误的状态标志
0	POWER0 = 0 或 RXE0 = 0 时; 或 ASIS0 寄存器的读取
1	接收结束后, 传送数据的奇偶校验与奇偶校验位不匹配

FE0	显示帧错误的状态标志
0	POWER0 = 0 或 RXE0 = 0 时; 或 ASIS0 寄存器的读取
1	接收结束后, 没有检测出停止位。

OVE0	显示溢出错误的状态标志
0	POWER0 = 0 和 RXE0 = 0 时; 或 ASIS0 寄存器的读取
1	接收数据被设置在 RXB0 寄存器, 并且在读取其数据之前完成下一个接收动作时

- 注意事项**
1. PE0 位的操作根据异步串行接口操作模式寄存器(ASIM0)的 PS01 和 PS00 位的设定值不同而不同。
 2. 无论停止位数如何, 接收数据的停止位只检测接收数据的第 1 位。
 3. 若产生溢出错误, 则下一个接收数据不写入接收缓冲寄存器 0(RXB0), 而被忽略。
 4. 若从 ADPC 读取数据, 则产生一个等待周期。外围硬件时钟(fPRS)停止时, 禁止从 ASIS0 读取数据。详情参阅第二十六章 待机注意事项。

(3) 波特率发生器控制寄存器 0(BRGC0)

该寄存器用于选择串行接口 UART0 的基本时钟和 5 位计数器的分频值。

可按 8 位存储器操作指令设置 BRGC0。

复位信号产生时, 该寄存器设置为 1FH。

图 13-4. 波特率发生器控制寄存器 0(BRGC0)的格式

地址: FF51H 复位后: 1FH R/W

符号	7	6	5	4	3	2	1	0
BRGC0	TPS01	TPS00	0	MDL04	MDL03	MDL02	MDL01	MDL00

TPS01	TPS00	基准时钟 (f _{CLK0})选择			
		f _{PRS} = 2 MHz		f _{PRS} = 5 MHz	f _{PRS} = 10 MHz
0	0	TM51 输出 ^注			
0	1	f _{PRS} /2	1 MHz	2.5 MHz	5 MHz
1	0	f _{PRS} /2 ³	250 kHz	625 kHz	1.25 MHz
1	1	f _{PRS} /2 ⁵	62.5 kHz	156.25 kHz	312.5 kHz

MDL04	MDL03	MDL02	MDL01	MDL00	k	选择 5 位计数器输出时钟
0	0	×	×	×	×	禁止设置
0	1	0	0	0	8	f _{CLK0} /8
0	1	0	0	1	9	f _{CLK0} /9
0	1	0	1	0	10	f _{CLK0} /10
•	•	•	•	•	•	•
•	•	•	•	•	•	•
•	•	•	•	•	•	•
•	•	•	•	•	•	•
•	•	•	•	•	•	•
1	1	0	1	0	26	f _{CLK0} /26
1	1	0	1	1	27	f _{CLK0} /27
1	1	1	0	0	28	f _{CLK0} /28
1	1	1	0	1	29	f _{CLK0} /29
1	1	1	1	0	30	f _{CLK0} /30
1	1	1	1	1	31	f _{CLK0} /31

注 选择 TM51 输出用作基本时钟时, 应注意以下几点。

- TM51 与 CR51 相匹配时, 计数时钟进入清零并启动的模式(TMC516 = 0)
先启动 8 位定时器/事件计数器 51 后, 允许定时器 F/F 进行反转操作(TMC511 = 1)。
- PWM 模式(TMC516 = 1)
先启动 8 位定时器/事件计数器 51 后, 设置计数时钟, 使占空比 = 50%。

注意事项 1. 重写 MDL04 至 MDL00 位时, 必须确保 ASIM0 的第 6 位(TXE0)和第 5 位(RXE0) = 0。
2. 重写 TPS01 和 TPS00 位时, 必须确保 ASIM0 的第 7 位(POWER0) = 0。
3. 波特率为 5 位计数器输出时钟的 1/2。

备注 1. f_{CLK0}: 通过 TPS01 和 TPS00 位选择基本时钟频率
2. f_{PRS}: 外部硬件时钟频率
3. k: 由 MDL04 至 MDL00 位设置的值 (k = 8, 9, 10, ..., 31)
4. ×: 忽略

5. TMC516: 8 位定时器模式控制寄存器 51(TMC51)的第 6 位
TMC511: TMC51 的第 1 位

(4) 端口模式寄存器 3(PM3)

该寄存器按 1 位设置端口 3 的输入和输出。

P31/TxD0/CMPCOM 引脚用作串行接口的数据输出时，PM31 清零并将 P31 输出锁存置“1”。

P32/RxD0/CMPIN 引脚用作串行接口的数据输入时，将 PM32 置“1”。此时，P32 的输出锁存即可为“0”也可“1”。

可用 1 位或 8 位存储器操作指令设置 PM3。

复位信号产生时，该寄存器设置为 FFH。

图 13-5. 端口模式寄存器 3(PM3)的格式

地址: FF23H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PM3	1	1	1	PM34	PM33	PM32	PM31	PM30

PM3n	P3n 引脚 I/O 模式选择(n = 0 至 4)
0	输出模式 (输出缓冲打开)
1	输入模式 (输出缓冲关闭)

13.4 串行接口 UART0 操作

串行接口 UART0 具有以下两种模式。

- 运行停止模式
- 异步串行接口(UART)模式

13.4.1 操作停止模式

在该模式下，不执行串行通信，从而降低功耗。此外，引脚可用作通用端口引脚。操作停止模式下，ASIM0 的第 7 位、第 6 位和第 5 位(POWER0、TXE0 和 RXE0)清零。

(1) 使用寄存器

通过异步串行接口操作模式寄存器 0(ASIM0)设置操作停止模式。

可用 1 位或 8 位存储器操作指令设置 ASIM0。

复位信号产生时，该寄存器设置为 01H。

地址：FF50H 复位后：01H R/W

符号	<7>	<6>	<5>	4	3	2	1	0
ASIM0	POWER0	TXE0	RXE0	PS01	PS00	CL0	SL0	1
POWER0		允许/禁止操作内部运行时钟						
0 ^{註1}		禁止内部操作时钟的操作（时钟恒为低电平），并异步复位内部电路 ^{註2} 。						
TXE0		允许/禁止传送						
0		禁止传送（同步复位传送电路）						
RXE0		允许/禁止接收						
0		禁止接收（同步复位接收电路）						

注 1. POWER0 = 0 时，从 RxD0 引脚的输入恒为高电平。

2. 将异步串行接口接收错误状态寄存器 0(ASIS0)、传送移位寄存器 0(TXS0)和接收缓冲寄存器 0(RXB0)复位。

注意事项 操作停止模式下，先将 TXE0 和 RXE0CI 分别清零后，再将 POWER0 清零。

开始通信时，先设置 POWER0 为“1”后，再设置 TXE0 为“1”。

注意事项 RxD0/CMPIN/P32 和 TxD0/CMPCOM/P31 引脚用作通用端口引脚时，参阅第四章 端口功能。

13.4.2 异步串行接口(UART)模式

可将起始位后的 1 字节数据用作传送/接收模式，执行全双工操作。

内置 UART 专用波特率发生器，适用于较大范围内任意波特率的通信。

(1) 使用的寄存器

- 异步串行接口操作模式寄存器 0(ASIM0)
- 异步串行接口接收错误状态寄存器 0 (ASIS0)
- 波特率发生器控制寄存器 0(BRGC0)
- 端口模式寄存器 3(PM3)
- 端口寄存器 3 (P3)

UART 模式的基本操作设置步骤示例如下。

- <1> 设置 BRGC0 寄存器 (参阅图 13-4)。
- <2> 设置 ASIM0 寄存器的第 1 位至第 4 位(SL0, CL0, PS00 和 PS01)(参阅图 13-2)。
- <3> ASIM0 寄存器第 7 位(POWER0)置“1”。
- <4> ASIM0 寄存器第 6 位(TXE0)置“1”。→允许传送。
ASIM0 寄存器第 5 位(RXE0)置“1”。→允许接收。
- <5> 写入数据至 TXS0 寄存器。→开始数据传送。

注意事项 设置端口模式寄存器和端口寄存器时，应考虑与通信另一方的关系。

寄存器设置与引脚之间的关系如下所示。

表 13-2. 寄存器设置和引脚之间的关系

POWER0	TXE0	RXE0	PM31	P31	PM32	P32	UART0 操作	引脚功能	
								TxD0/CMPCOM/P31	RxD0/CMPIN/P32
0	0	0	× ^注	× ^注	× ^注	× ^注	停止	CMPCOM/P31	CMPIN/P32
1	0	1	× ^注	× ^注	1	×	接收	CMPCOM/P31	RxD0
	1	0	0	1	× ^注	× ^注	传送	TxD0	CMPIN/P32
	1	1	0	1	1	×	传送/接收	TxD0	RxD0

注 可用作端口功能和比较器。

备注

×: 忽略

POWER0: 异步串行接口操作模式寄存器 0(ASIM0)第 7 位

TXE0: ASIM0 第 6 位

RXE0: ASIM0 第 5 位

PM3×: 端口模式寄存器

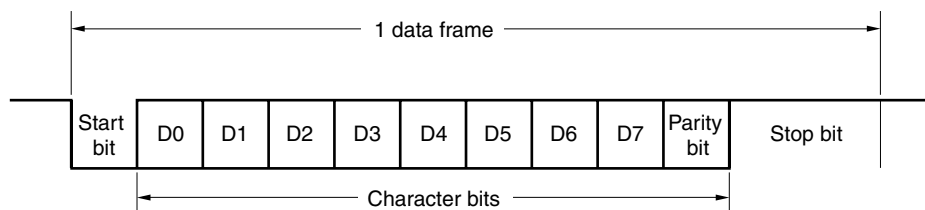
P3×: 端口输出锁存器

(2) 通信操作

(a) 正常传送/接收数据的格式和波形示例

正常传送/接收数据的格式和波形示例如图 13-6 和 13-7 所示。

图 13-6. 正常 UART 传送/接收数据的格式



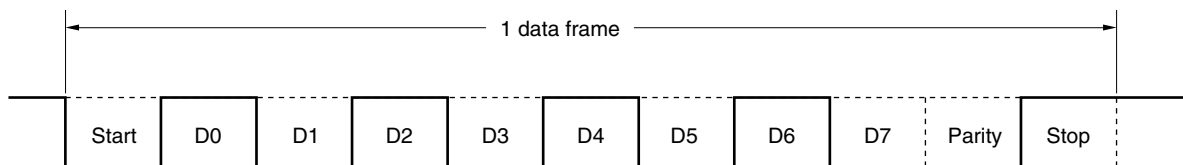
一个数据帧由以下各位组成

- 起始位 ...1 位
- 字符位 ...7 或 8 位(LSB first)
- 奇偶校验位 ...偶校验/奇校验/零校验/无校验
- 停止位 ...1 或 2 位

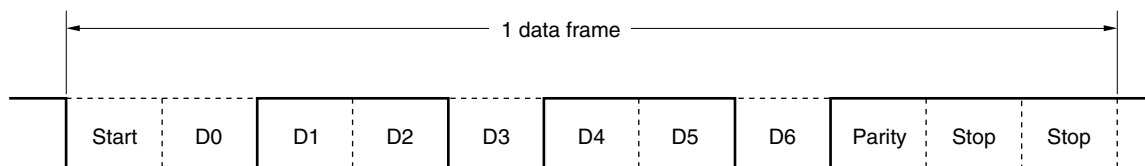
由异步串行接口操作模式寄存器 0(ASIM0)指定一个数据帧中的字符位宽度、校验方式和停止位宽度。

图 13-7. 正常 UART 传送/接收数据波形实例

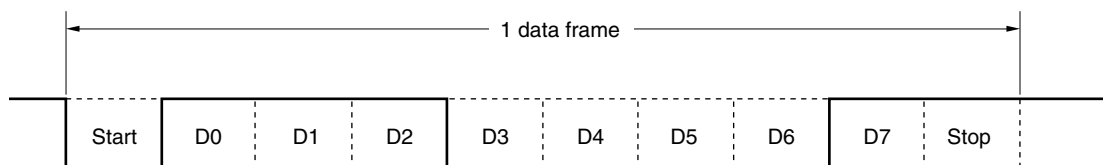
1. 数据长度：8 位，奇偶校验：偶校验，停止位：1 位，通信数据：55H



2. 数据长度：7 位，奇偶校验：偶校验，停止位：2 位，通信数据：36H



3. 数据长度：8 位，奇偶校验：无，停止位：1 位，通信数据：87H



(b) 奇偶校验类型和操作

奇偶校验位用于检测通信数据中的位错误。通常，发送和接收两端均可采用相同的校验方式。采用偶校验和奇校验可检测出 1 位（奇数）错误。采用零校验和无校验则检测不出错误。

(i) 偶校验

- 发送

通过控制包括奇偶校验位在内的发送数据，使值中“1”的个数为偶数。

校验位取值如下。

若发送数据有奇数个“1”： 1

若发送数据有偶数个“1”： 0

- 接收

计算包括校验位在内的接收数据中“1”的个数。若是奇数，则产生校验错误。

(ii) 奇校验

- 发送

与偶校验方式不同，通过控制包括奇偶校验位在内的发送数据，使值中“1”的个数为奇数。

发送数据有奇数个“1”： 0

发送数据有偶数个“1”： 1

- 接收

计算包括校验位在内的接收数据中“1”的个数。若是偶数，则产生校验错误。

(iii) 0 奇偶校验

无论发送数据为何值，发送时校验位清零。

接收数据时检测不出校验位。因此，无论校验位是“0”或“1”，均不会产生奇偶错误。

(iv) 无奇偶校验

发送数据没有奇偶位。

接收数据时，当作无奇偶位进行接收。因没有校验位，不产生校验错误。

(c) 传送

若异步串行接口操作模式寄存器 0(ASIM0)的第 7 位(POWER0)=1 和第 6 位(TXE0)置“1”，则允许传送。通过传送数据写入传送移位寄存器 0(TXS0)，可启动传送。起始位、校验位以及停止位自动添加到数据中。

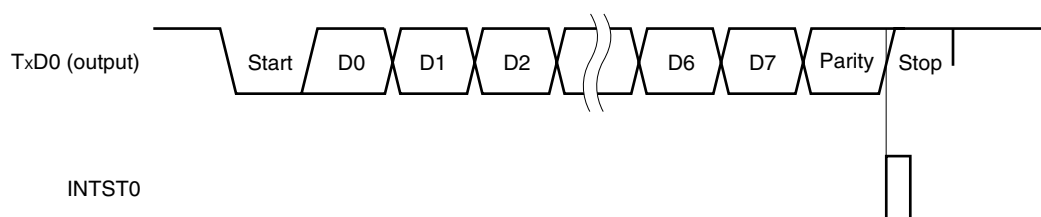
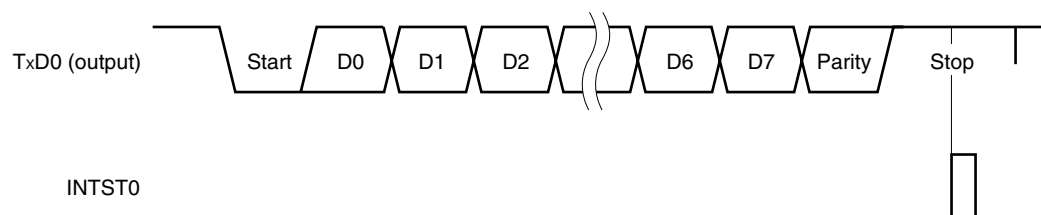
传送开始时，先从 TxD0 引脚输出起始位，然后从 LSB 开始依次传送其它位数据。传送结束后，由 ASIM0 设置的校验位和停止位添加到数据中，并产生传送完成中断请求(INTST0)。

一直到待发数据写入 TXS0 时，才结束传送操作。

传送完成中断请求(INTST0)的时序如图 13-8 所示。最后停止位一旦输出就产生中断。

注意事项 将待传送数据写入 TXS0 后，在传送完成中断信号(INTST0)产生之前，禁止写入下一个传送数据。

图 13-8. 传送完成中断请求时序

1. 停止位长度：1**2. 停止位长度：2**

(d) 接收

异步串行接口操作模式寄存器 0(ASIM0)的第 7 位(POWER0)置“1”后, 再将 ASIM0 的第 5 位(RXE0)置 1 时, 允许接收并对 RxD0 引脚输入进行采样。

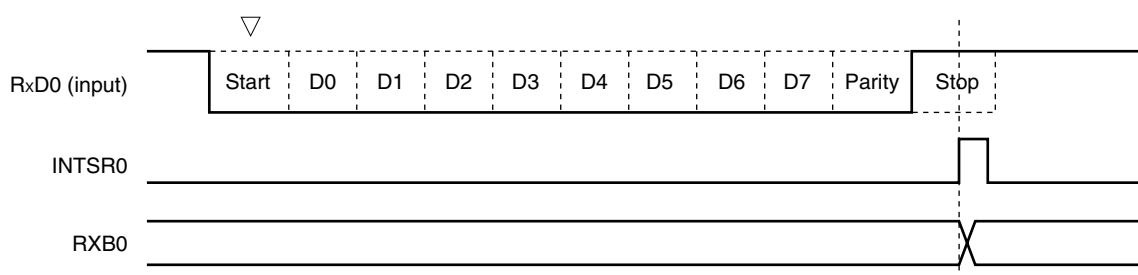
检测出 RxD0 引脚输入的下降沿时, 波特率发生器的 5 位计数器开始计数。计算出波特率发生器控制寄存器 0(BRGC0)的设置值时, 再次采样 RxD0 的输入信号(如图 14-9 中的)。此时, 若 RxD0 为低电平, 则被认为起始位。

检测出起始位时, 开始接收, 以设置的波特率将串行数据依次存入接收移位寄存器 0(RXS0)。当接收到停止位时, 会产生接收完成中断信号(INTSR0), 并且将 RXS0 的数据写入接收缓冲寄存器 0(RXB0)。若产生溢出错误(OVE0), 则接收的数据不写入 RXB0。

即使在接收过程中产生校验错误(PE0), 也继续接收, 直至接收到停止位, 而在接收完成后会产生一个接收错误中断(INTSR0)。

接收操作完成且存在接收错误时, 产生 INTSR0。

图 13-9. 接收完成中断请求时序



注意事项 1. 若出现接收错误, 则读取异步串行接口接收错误状态寄存器 0(ASIS0)后, 读取接收缓冲寄存器 0(RXB0)并清除错误标志。

否则, 接收到下一个数据时会产生溢出错误, 而且将持续接收错误状态。

2. 始终按照“停止位的个数 = 1”执行接收。忽略第二个停止位。

(e) 接收错误

接收期间可能出现的三种类型错误：校验错误、帧错误或溢出错误。若已设置异步串行接口接收错误状态寄存器 0(ASIS0)的错误标志位，作为数据接收的结果，则将会产生接收错误中断(INTSR0)。

通过在执行接收错误中断(INTSR0)处理中读取 ASIS0 内容，可检测出接收期间产生的何种错误(参阅图 13-3)

通过读取 ASIS0，可将其内容清零。

表 13-3. 接收错误原因

接收错误	原因
奇偶校验错误	传送时指定的校验位与接收数据的校验位不匹配
帧错误	没有检测到停止位
溢出错误	在从接收缓冲寄存器(RXB0)读出数据之前，已完成下一个数据接收

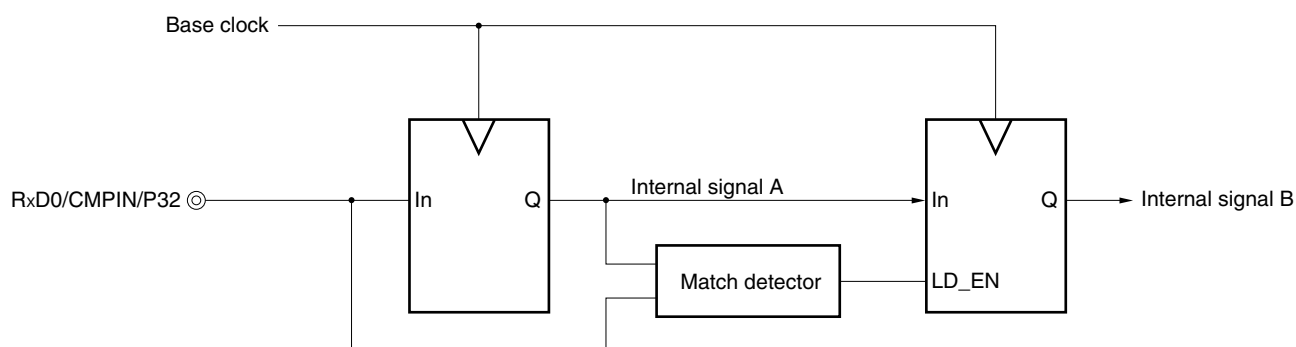
(f) 接收数据噪声滤波器

通过分频器，使用基本时钟输出，采样 RxD0 信号。

若两次采样值相同，则匹配检测器的输出发生变化，并将其采样的数据作为输入数据。

电路配置如图 12-10 所示，接收操作的内部处理与外部信号状态相比延迟 2 个时钟。

图 13-10. 噪声滤波器电路



13.4.3 专用波特率发生器

专用波特率发生器由 1 台钟源选择器和 1 台 5 位可编程计数器组成，在 UARTCn 传送/接收期间产生串行时钟。
传送和接收分别使用不同的 5 位计数器。

(1) 波特率发生器配置

- 基准时钟

异步串行接口操作模式寄存器 0(ASIM0)的第 7 位(POWER0)置“1”时，可将波特率发生器控制寄存器 0(BRGC0)的第 7 位和第 6 位(TPS01 和 TPS00)选择的时钟提供给每个模块。该时钟称为基本时钟,其频率用 f_{XCLK0} 表示。
POWER0 = 0 时，基本时钟恒为低电平。

- 传送计数器

当异步串行接口操作模式寄存器 0(ASIM0)的第 7 位(POWER0)或第 6 位(TXE0) = 0 时，该计数器停止操作，并清零。

POWER0 = 1 且 TXE0 = 1 时，计数器开始计数。

第 1 个传送数据已写入传送移位寄存器 0(TXS0)时，计数器清零。

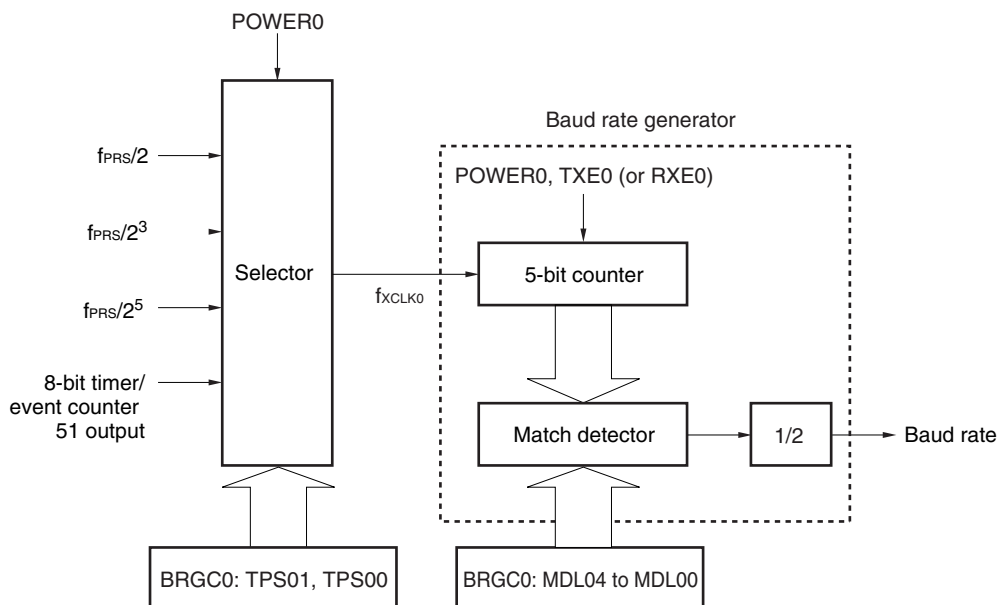
- 接收计数器

异步串行接口操作模式寄存器 0(ASIM0)的第 7 位(POWER0)或第 5 位(TXE0)置“0”时，该计数器停止操作，并清零。

检测出起始位时，计数器开始计数。

接收到 1 帧数据后，直到检测出下一个起始位为止，计数器停止计数。

图 13-11. 波特率发生器配置



备注

POWER0: 异步串行接口操作模式寄存器 0(ASIM0)第 7 位
 TXE0: ASIM0 第 6 位
 RXE0: ASIM0 第 5 位
 BRGC0: 波特率发生器控制寄存器 0

(2) 串行时钟发生器

由波特率发生器控制寄存器 0(BRGC0)指定产生的串行时钟。

根据 BRGC0 的第 7 位和第 6 位((TPS01 和 TPS00)，选择输入给 5 位计数器的时钟。

根据 BRGC0 的第 4 位至第 0 位(MDL04 至 MDL00)，选择 5 位计数器的分频值($f_{XCLK0}/8$ 至 $f_{XCLK0}/31$)。

13.4.4 波特率计算**(1) 波特率计算公式**

可通过下列公式计算波特率。

$$\bullet \text{ 波特率} = \frac{f_{XCLK0}}{2 \times k} [\text{bps}]$$

f_{XCLK0} : 通过 BRGC0 寄存器的 TPS01 和 TPS00 位选择基本时钟的频率。

k: 由 BRGC0 寄存器的 MDL04 至 MDL00 位设置的值 ($k = 8, 9, 10, \dots, 31$)

表 13-4. 设置 TPS01 和 TPS00 值

TPS01	TPS00	基准时钟 (f_{XCLK0})选择			
			$f_{PRS} = 2 \text{ MHz}$	$f_{PRS} = 5 \text{ MHz}$	$f_{PRS} = 10 \text{ MHz}$
0	0	TM51 输出 ^注			
0	1	$f_{PRS}/2$	1 MHz	2.5 MHz	5 MHz
1	0	$f_{PRS}/2^3$	250 kHz	625 kHz	1.25 MHz
1	1	$f_{PRS}/2^5$	62.5 kHz	156.25 kHz	312.5 kHz

注 选择 TM51 输出用作基本时钟时，应注意以下几点。

- TM51 与 CR51 相匹配时，计数时钟进入清零并启动的模式($TMC516 = 0$)
启动 8 位定时器/事件计数器 51 后，允许定时器 F/F 进行反转操作($TMC511 = 1$)。
- PWM 模式($TMC516 = 1$)
启动 8 位定时器/事件计数器 51 后，设置计数时钟，使占空比 = 50%。

(2) 波特率错误

可通过以下公式计算波特率误差。

$$\bullet \text{ 误差}(\%) = \left(\frac{\text{实际波特率 (有误差的波特率)}}{\text{理想波特率 (正确波特率)}} - 1 \right) \times 100 [\%]$$

- 注意事项**
1. 传送过程中的波特率误差必须在接收方的允许误差范围内。
 2. 接收过程中的波特率误差必须满足“(4)接收时的允许波特率范围”中所指定的范围。

示例: 基准时钟频率 = 2.5 MHz = 2,500,000 Hz
 BRGC0 寄存器 MDL04 至 MDL00 位的设定值 = 10000B (k = 16)
 目标波特率 = 76,800 bps

$$\begin{aligned} \text{波特率} &= 2.5 \text{ M}/(2 \times 16) \\ &= 2,500,000/(2 \times 16) = 78,125 [\text{bps}] \end{aligned}$$

$$\begin{aligned} \text{错误} &= (78,125/76,800 - 1) \times 100 \\ &= 1.725 [\%] \end{aligned}$$

(3) 设置波特率实例

表 13-5. 波特率发生器的数据设置

波特率 (bps)	f _{PRS} = 2.0 MHz				f _{PRS} = 5.0 MHz				f _{PRS} = 10.0 MHz			
	TPS01, TPS00	k	计算值	ERR [%]	TPS01, TPS00	k	计算值	ERR [%]	TPS01, TPS00	k	计算值	ERR [%]
4800	2H	26	4808	0.16	3H	16	4883	1.73	—	—	—	—
9600	2H	13	9615	0.16	3H	8	9766	1.73	3H	16	9766	1.73
10400	2H	12	10417	0.16	2H	30	10417	0.16	3H	15	10417	0.16
19200	1H	26	19231	0.16	2H	16	19531	1.73	3H	8	19531	1.73
24000	1H	21	23810	-0.79	2H	13	24038	0.16	2H	26	24038	0.16
31250	1H	16	31250	0	2H	10	31250	0	2H	20	31250	0
33600	1H	15	33333	-0.79	2H	9	34722	3.34	2H	19	32895	-2.1
38400	1H	13	38462	0.16	2H	8	39063	1.73	2H	16	39063	1.73
56000	1H	9	55556	-0.79	1H	22	56818	1.46	2H	11	56818	1.46
62500	1H	8	62500	0	1H	20	62500	0	2H	10	62500	0
76800	—	—	—	—	1H	16	78125	1.73	2H	8	78125	1.73
115200	—	—	—	—	1H	11	113636	-1.36	1H	22	113636	-1.36
153600	—	—	—	—	1H	8	156250	1.73	1H	16	156250	1.73
312500	—	—	—	—	—	—	—	—	1H	8	312500	0
625000	—	—	—	—	—	—	—	—	—	—	—	—

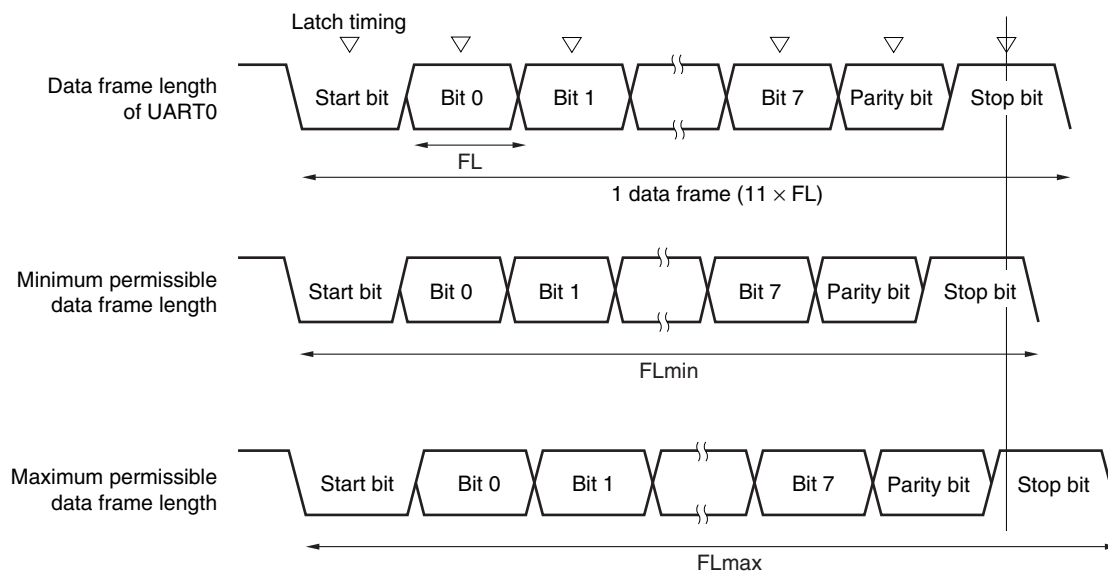
备注 TPS01, TPS00: 波特率发生器控制寄存器 0(BRGC0)的第 7 位和第 6 位 (基本时钟(f_{CLK0})的设置)
 k: 由 BRGC0 的 MDL04 至 MDL00 位设置的值 (k = 8, 9, 10, ..., 31)
 f_{PRS}: 外部硬件时钟频率
 ERR 波特率错误

(4) 接收期间波特率允许范围

接收期间来自发送端的波特率允许误差范围如下所示。

注意事项 通过以下公式计算，必须确保接收期间波特率误差在允许误差范围内。

图 13-12. 接收期间波特率的允许范围



如图 13-12 所示，检测出起始位时，通过波特率发生器控制寄存器 0(BRGC0)设置计数器来确定接收数据的锁存时序。若数据的最后一位(停止位)满足该锁存时序，则正确接收该数据。假设接收数据为 11 位，则各项理论值计算如下。

$$FL = (\text{Brate})^{-1}$$

Brate: UART0 波特率

k: BRGC0 设定值

FL: 1 位数据长度

锁存时序容限: 2 个时钟

$$\text{最小允许数据帧宽度: } FL_{\min} = 11 \times FL - \frac{k-2}{2k} \times FL = \frac{21k+2}{2k} FL$$

因此，发送端可接收的最大波特率如下所示。

$$BR_{\max} = (FL_{\min}/11)^{-1} = \frac{22k}{21k+2} Brate$$

同样，最大允许数据帧宽度的计算如下。

$$\frac{10}{11} \times FL_{\max} = 11 \times FL - \frac{k+2}{2 \times k} \times FL = \frac{21k-2}{2 \times k} FL$$

$$FL_{\max} = \frac{21k-2}{20k} FL \times 11$$

因此，发射端可接收的最小波特率如下所示。

$$BR_{\min} = (FL_{\max}/11)^{-1} = \frac{20k}{21k-2} Brate$$

通过上述最小和最大波特率公式，计算得出的 UART0 与传送端之间的允许波特率误差，如下所示。

表 13-6. 波特率最大/最小允许误差

分频比(k)	波特率最大允许误差	波特率最小允许误差
8	+3.53%	-3.61%
16	+4.14%	-4.19%
24	+4.34%	-4.38%
31	+4.44%	-4.47%

- 备注 1.** 接收时的允许误差取决于每帧的位数、输入时钟频率和分频比(k)。基准时钟频率和分频比(k)越高，则允许的误差就越大。
- 2.** k:BRGC0 设定值

第十四章 中断功能

14.1 中断功能类型

使用以下两种中断功能。

(1) 可屏蔽中断功能

接收屏蔽控制的中断。通过设置优先级制定标志寄存器(PR0L, PR0H, PR1L)，将可屏蔽中断分为高优先级中断组和低优先级中断组。

高优先级中断产生时，可以对低优先级中断进行多级中断。另外，若同时产生两个以上相同优先级的中断请求，则根据向量中断服务的优先级进行处理。优先级顺序参阅表 14-1。

待机解除信号产生时，解除 STOP 和 HALT 模式。

可屏蔽中断分为外部中断请求和内部中断请求。

(2) 软件中断

通过执行 BRK 指令，产生向量中断。即使禁止中断时也响应。软件中断不受中断优先级控制。

14.2 中断源和配置

中断源包括可屏蔽中断和软件中断。另外，还用 4 种复位源（参阅表 14-1）。

表 14-1. 中断源表 (1/2)

中断 类型	内部/ 外部	基本 配置 类型 ^{注1}	默认 优先 级 ^{注2}	中断源		向量表地 址
				名称	触发器	
可屏蔽	内部	(A)	0	INTLVI	低电压检测 ^{注3}	0004H
	外部	(B)	1	INTP0	引脚输入沿检测	0006H
			2	INTP1		0008H
			3	INTCMP	比较器沿检测	000AH
	-	-	4	-	-	000CH
			5	-		000EH
			6	-		0010H
			7	-		0012H
	内部	(A)	8	INTSR0	UART0 接收结束/产生接收错误	0014H
			9	INTST0	UART0 传送结束	0016H
	-	-	10	-	-	0018H
	内部	(A)	11	INTTMH1	TMH1 和 CMP01 相匹配 (指定比较寄存器)	001AH
	-	-	12	-	-	001CH
			13			001EH
	内部	(A)	14	INTTM000	TM00 和 CR000 相匹配 (指定比较寄存器), TI010 引脚有效沿检测 (指定捕捉寄存器)	0020H
			15	INTTM010	TM00 和 CR010 相匹配 (指定比较寄存器), TI000 引脚有效沿检测 (指定捕捉寄存器)	0022H
			16	INTAD	A/D 转换结束	0024H
	-	-	17	-	-	0026H
			18			0028H
	内部	(A)	19	INTTM51 ^{注4}	TM51 和 CR51 相匹配 (指定比较寄存器)	002AH
	-	-	20	-	-	002CH

- 注 1. 基本配置类型(A)至(C)分别与图 14-1 中的(A)至(C)相对应。
2. 同时产生 2 个以上可屏蔽中断时，默认优先级决定向量中断的处理顺序。“0”为最高优先级，“28”为最低优先级。
3. 低电压检测寄存器(LVIM)的第 1 位(LVIMD)清零。
4. 载波发生器模式下，使用 8 位定时器/计数器 51 时，在 INTTM5H1 信号产生时序中产生中断（参阅图 8-11 传送时序）。

表 14-1. 中断源表 (2/2)

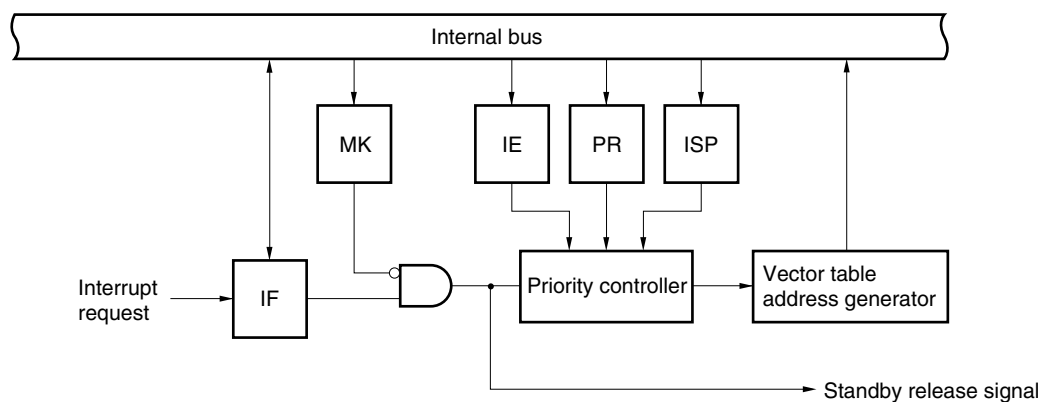
中断 类型	内部/ 外部	基本 配置 类型 ^{注1}	默认 优先 级 ^{注2}	中断源		向量表地 址
				名称	触发器	
可屏蔽	-	-	21	-	-	002EH
			22			0030H
			23			0032H
			24			0034H
			25			0036H
			26			0038H
			27			003AH
			28			003CH
软件	-	(C)	-	BRK	执行 BRK 指令	003EH
复位	-	-	-	RESET	复位输入	0000H
				POC	上电清除	
				LVI	低电压检测 ^{注3}	
				WDT	WDT 溢出	

注 1. 基本配置类型(A)至(C)分别与图 14-1 中的(A)至(C)相对应。

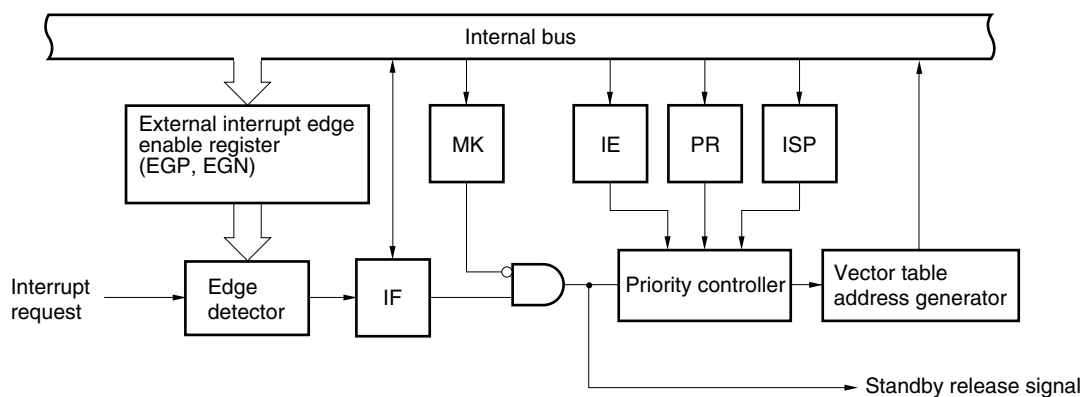
- 同时产生 2 个以上可屏蔽中断时，默认优先级决定向量中断的处理顺序。“0”为最高优先级，“28”为最低优先级。
- 低电压检测寄存器(LVIM)的第 1 位(LVIMD)置“1”。

图 14-1. 中断功能的基本配置 (1/2)

(A) 内部可屏蔽中断



(B) 外部可屏蔽中断(INTPn, INTCMP)

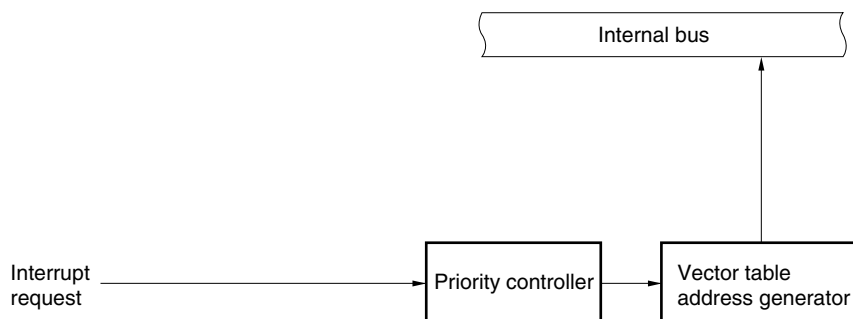


备注 n = 0, 1

IF: 中断请求标志
IE: 中断允许标志
ISP: 服务中优先级标志
MK: 中断屏蔽标志
PR: 优先级指定标志

图 14-1. 中断功能的基本配置 (2/2)

(C) 软件中断



IF: 中断请求标志
 IE: 中断允许标志
 ISP: 服务中优先级标志
 MK: 中断屏蔽标志
 PR: 优先级指定标志

14.3 寄存器控制中断功能

以下 6 种寄存器用来控制中断功能。

- 中断请求标志寄存器(IF0L, IF0H, IF1L)
- 中断屏蔽标志寄存器(MK0L, MK0H, MK1L)
- 优先级指定标志寄存器(PR0L, PR0H, PR1L)
- 允许外部中断上升沿寄存器(EGP)
- 允许外部中断下降沿寄存器(EGN)
- 程序状态字(PSW)

对应中断请求源的中断请求标志、中断可屏蔽标志和优先级指定标志名称如表 14-2 所示。

表 14-2. 对应中断请求源的标志

中断源	中断请求标志		中断屏蔽标志		优先级指定标志	
		寄存器		寄存器		寄存器
INTLVI	LVIIIF	IF0L	LVIMK	MK0L	LVIPR	PR0L
INTP0	PIF0		PMK0		PPR0	
INTP1	PIF1		PMK1		PPR1	
INTCMP	CMPIF		CMPMK		CMPPR	
INTSR6	SRIF0	IF0H	SRMK0	MK0H	SRPR0	PR0H
INTST0	STIF0		STMK0		STPR0	
INTTMH1	TMIFH1		TMMKH1		TMPRH1	
INTTM000	TMIF000		TMMK000		TMPR000	
INTTM010	TMIF010		TMMK010		TMPR010	
INTAD	ADIF	IF1L	ADMK	MK1L	ADPR	PR1L
INTTM51 ^注	TMIF51		TMMK51		TMPR51	

注 载波发生器模式下，使用 8 位定时器/计数器 51 时，在 INTTM5H1 信号产生时序中产生中断（参阅图 8-11 传送时序）。

(1) 中断请求标志寄存器(IF0L, IF0H, IF1L)

对应中断请求产生或指令执行时，中断请求标志置“1”。响应中断请求的复位信号产生或指令执行时，这些标志清零。

响应中断时，中断请求标志自动清零，然后进入中断程序。

可用 1 位或 8 位存储器操作指令设置 IF0L、IF0H 和 IF1L。

IF0L 与 IF0H 组合成 16 位寄存器 IF0 时，可用 16 位存储器操作指令设置这些寄存器。

复位信号产生时，这些寄存器清为 00H。

- 注意事项**
1. 待机模式解除后，操作定时器、串行接口或 A/D 转换器时，必须将中断请求标志清零之后进行操作。噪声有可能产生中断请求标志。
 2. 修改中断请求标志寄存器中的标志时，操作使用 1 位存储器指令(CLR1)。使用 C 语言描述时，由于编译的汇编程序必须是 1 位存储器操作指令(CLR1)，所以应使用“IF0L.0 = 0;”或“_asm(“clr1 IF0L, 0”);”等 1 位操作指令。
若使用 C 语言描述一条 8 位存储器操作指令，例如“IF0L &= 0xfe;”，则编译后将成为 3 条汇编指令。

```
mov a, IF0L  
和 a, #0FEH  
mov IF0L, a
```

在这种情况下，当处于“mov a, IF0L”和“mov IF0L, a”之间的时序时，即使同一中断请求标志寄存器(IF0L)的其他位的请求标志置“1”，也被“mov IF0L, a”清零。因此，在 C 语言中使用 8 位存储器操作指令时必须注意。

图 14-2. 中断请求标志寄存器(IF0L, IF0H, IF1L)的格式

地址:	FFE0H	复位后:	00H	R/W				
符号	7	6	5	4	<3>	<2>	<1>	<0>
IF0L	0	0	0	0	CMPIF	PIF1	PIF0	LVIIIF

地址:	FFE1H	复位后:	00H	R/W				
符号	<7>	<6>	5	4	<3>	2	<1>	<0>
IF0H	TMIF010	TMIF000	0	0	TMIFH1	0	STIF0	SRIF0

地址:	FFE2H	复位后:	00H	R/W				
符号	7	6	5	4	<3>	2	1	<0>
IF1L	0	0	0	0	TMIF51	0	0	ADIF

XXIFX	中断请求标志
0	产生不中断请求信号。
1	产生中断请求, 中断请求状态

注意事项 确保 IF0L 的第 4 位至第 7 位、IF0H 的第 2 位、第 4 位和第 5 位以及 IF1L 的第 1 位、第 2 位、第 4 位至第 7 位置“0”。

(2) 中断屏蔽标志寄存器(MK0L, MK0H, MK1L)

中断屏蔽标志用于允许/禁止相关的可屏蔽中断服务。

可用 1 位或 8 位存储器操作指令设置 MK0L、MK0H 和 MK1L。MK0L 和 MK0H 组合成 16 位寄存器 MK0 时，可用 16 位存储器操作指令设置这些寄存器。

复位信号产生时，这些寄存器设置为 FFH。

图 14-3. 中断屏蔽标志寄存器(MK0L, MK0H, MK1L) 的格式

地址: FFE4H 复位后: FFH R/W

符号	7	6	5	4	<3>	<2>	<1>	<0>
MK0L	1	1	1	1	CMPMK	PMK1	PMK0	LVIMK

地址: FFE5H 复位后: FFH R/W

符号	<7>	<6>	5	4	<3>	2	<1>	<0>
MK0H	TMMK010	TMMK000	1	1	TMMKH1	1	STMK0	SRMK0

地址: FFE6H 复位后: FFH R/W

符号	7	6	5	4	<3>	2	1	<0>
MK1L	1	1	1	1	TMMK51	1	1	ADMK

XXMKX	服务中断控制
0	允许服务中断
1	禁止服务中断

注意事项 确保 MK0L 的第 4 位至第 7 位、MK0H 的第 2 位、第 4 位和第 5 位以及 MK1L 的第 1 位、第 2 位、第 4 位至第 7 位设置为 1。

(3) 优先级指定标志寄存器(PROL, PROH, PR1L)

优先级指定标志寄存器用于设置相关的可屏蔽中断优先级次序。

可用 1 位或 8 位存储器操作指令设置 PROL、PROH 和 PR1L。PROL 和 PROH 组合成 16 位寄存器 PR0 时，可用 16 位存储器操作指令设置这些寄存器。

复位信号产生时，这些寄存器设置为 FFH。

图 14-4. 优先级指定标志寄存器(PROL, PROH, PR1L)的格式

地址:	FFE8H	复位后:	FFH	R/W				
符号	7	6	5	4	<3>	<2>	<1>	<0>
PROL	1	1	1	1	CMPPR	PPR1	PPR0	LVIPR

地址:	FFE9H	复位后:	FFH	R/W				
符号	<7>	<6>	5	4	<3>	2	<1>	<0>
PROH	TMPR010	TMPR000	1	1	TMPRH1	1	STPR0	SRPR0

地址:	FFEAH	复位后:	FFH	R/W				
符号	7	6	5	4	<3>	2	1	<0>
PR1L	1	1	1	1	TMPR51	1	1	ADPR

XXPRX	优先级选择
0	高优先级
1	低优先级

注意事项 确保 PROL 的第 4 位至第 7 位、PROH 的第 2 位、第 4 位和第 5 位以及 PR1L 的第 1 位、第 2 位、第 4 位至第 7 位置“1”。

(4) 允许外部中断上升沿寄存器(EGP)，允许外部中断下降沿寄存器(EGN)

这些寄存器用来指定 INTP_n 有效沿。

可用 1 位或 8 位存储器操作指令设置 EGP 和 EGN。

复位信号产生时，这些寄存器被清为 00H。

图 14-5. 允许外部中断上升沿寄存器(EGP)和允许外部中断下降沿寄存器(EGN)的格式

地址: FF48H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
EGP	0	0	0	0	0	EGP2	EGP1	EGP0

地址: FF49H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
EGN	0	0	0	0	0	EGN2	EGN1	EGN0

EGP _n	EGN _n	INTP _n 引脚有效沿选择
0	0	禁止沿检测
0	1	下降沿
1	0	上升沿
1	1	双边沿

注意事项 确保 EGP 的第 3 位至第 7 位和 EGN 清零。

备注 n = 0 至 2

表 14-3 给出 EGP_n 和 EGN_n 相对应的端口。

表 14-3. 与 EGP_n 和 EGN_n 相对应端口

检测允许寄存器		沿检测端口	中断请求信号 I
EGP0	EGN0	P30	INTP0
EGP1	EGN1	P33	INTP1
EGP2	EGN2	-	INTCMP

注意事项 外部中断功能切换到端口功能时，有可能检测出脉冲沿,因此,通过 EGP_n 和 EGN_n 置“0”，选择端口模式。

Remark n = 0 至 2

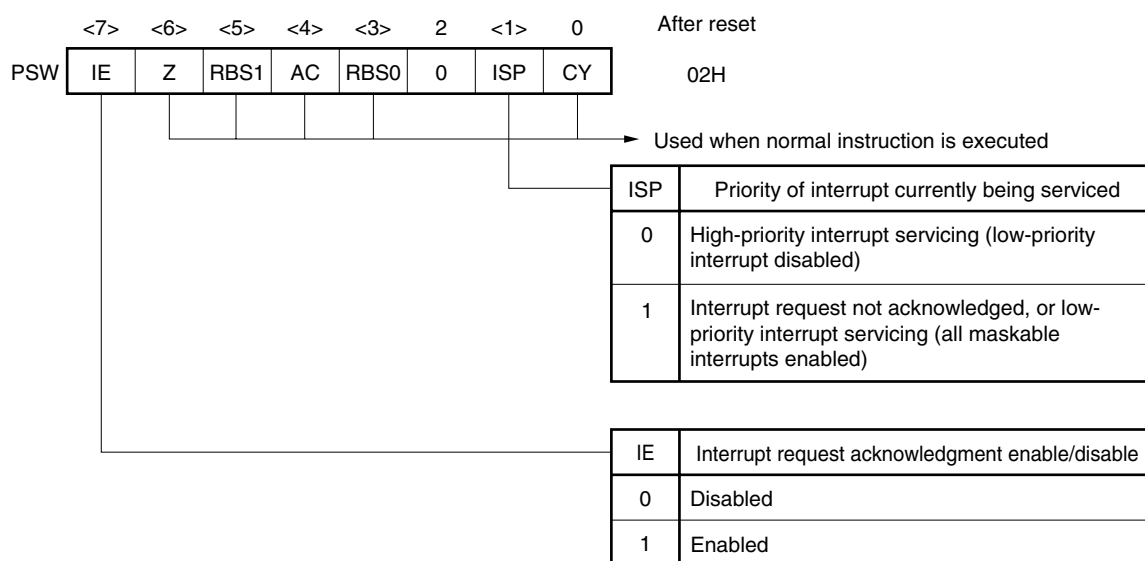
(5) 程序状态字(PSW)

程序状态字是用来保存指令执行结果和中断请求的当前状态的寄存器。PSW 映射设置/禁止可屏蔽中断的 IE 标志和控制操多级中断服务的 ISP 标志。

除了 8 位读/写操作指令，还可使用位操作指令和专用指令(EI 和 DI)来操作该寄存器。响应向量中断请求时，若执行 BRK 指令，则 PWS 内容将自动保存到堆栈中，并且 IE 标志复位为“0”。若响应可屏蔽中断请求，则将被响应中断的优先级指定标志内容转移到 ISP 标志中。可通过执行 PUSH PSW 指令将 PSW 内容保存到堆栈中。也可通过 RETI、RETB 和 POP PSW 指令将 PSW 内容从堆栈中恢复。

复位信号产生时，程序状态字设置为 02H。

图 14-6. 程序状态字的格式



14.4 中断服务操作

14.4.1 可屏蔽中断响应

若处于中断允许状态（IE 标志置“1”），则响应向量中断请求。但在较高优先级中断请求服务期间（ISP 标志复位为“0”），不响应低优先级中断请求。

从可屏蔽中断请求产生到向量中断服务执行所需时间如下表 17-4 所示。

有关中断请求响应时序，参阅图 14-8 和 14-9。

表 14-4. 从可屏蔽中断产生到服务所需时间

	最短时间	最长时间 ^注
xxPR = 0	7 个时钟	32 个时钟
xxPR = 1	8 个时钟	33 个时钟

注 若在除法指令执行之前产生一个中断请求，则等待更长时间。

备注 1 个时钟：1/fCPU (fCPU: CPU 时钟)

若同时产生两个以上的可屏蔽中断请求，则先响应优先级指定标志中优先级别高的请求。若两个以上的中断请求的优先级相同，则先响应最高默认优先级的中断请求。

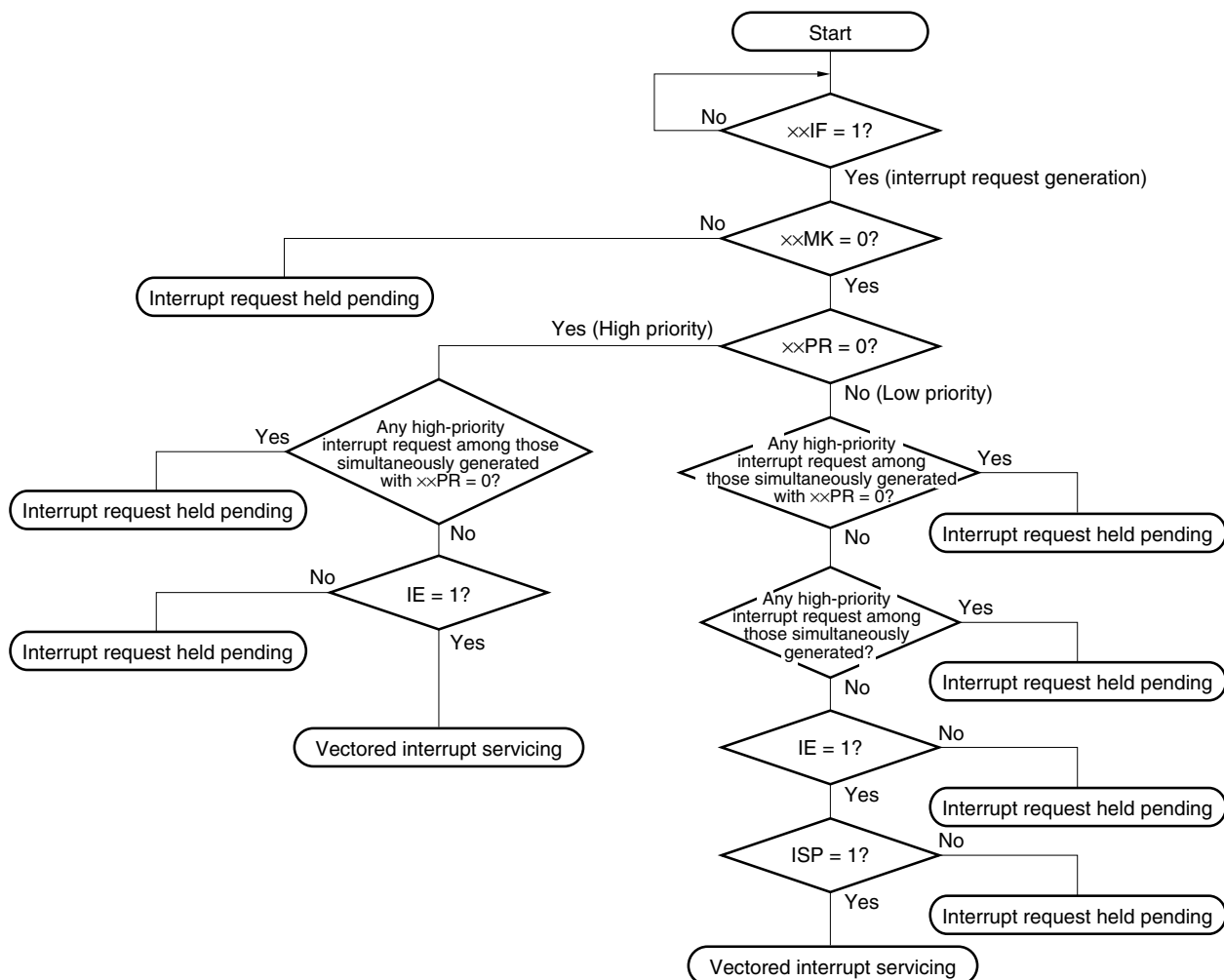
允许响应中断请求时，从等待状态转换成响应。

图 14-7 给出中断请求响应的算法。

若响应可屏蔽中断请求，则 PSW、PC 的内容依次保存到堆栈中，然后将 IE 标志复位为“0”，并将被响应中断对应的优先级指定标志的内容传送到 ISP 标志。将确定每个中断请求的向量表数据传送到 PC 并转移。

通过执行 RETI 指令，可从中断返回。

图 14-7. 中断请求响应处理的算法



xxIF: 中断请求标志

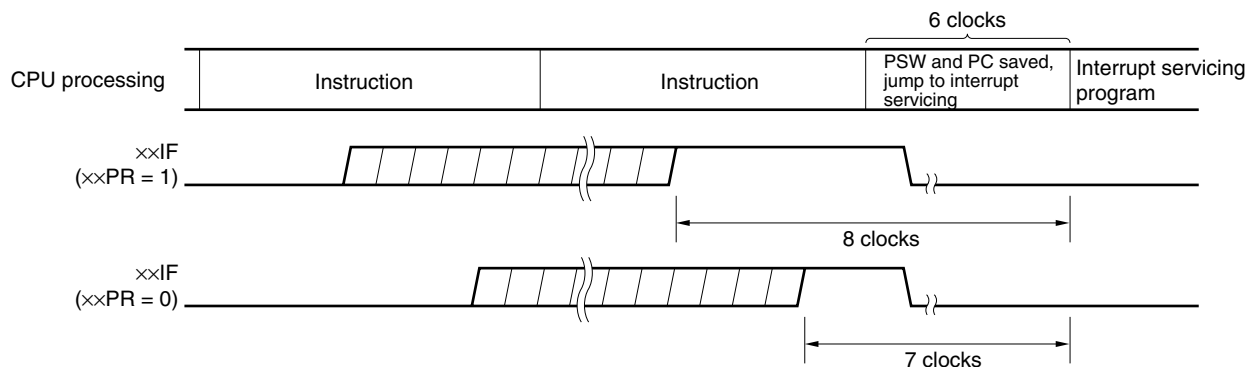
xxMK: 中断屏蔽标志

xxPR: 优先级指定标志

IE: 控制可屏蔽中断请求响应的标志 (1 = 允许、0 = 禁止)

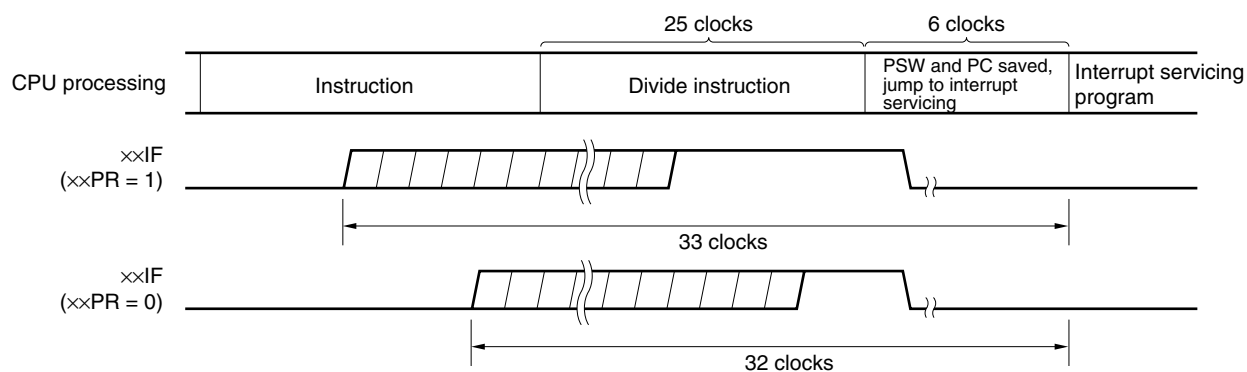
ISP: 指示正在服务的中断优先级别的标志 (0 = 高优先级中断服务, 1 = 没有响应的中断请求, 或低优先级中断服务)

图 14-8. 中断请求响应时序（最短时间）



备注 1 个时钟: $1/f_{CPU}$ (f_{CPU} : CPU 时钟)

图 14-9. 中断请求响应时序（最长时间）



备注 1 个时钟: $1/f_{CPU}$ (f_{CPU} : CPU 时钟)

14.4.2 软件中断请求响应

通过执行 **BRK** 指令响应软件中断。不可禁止软件中断。

若响应软件中断请求，则将程序状态字(PSW)和程序计数器(PC)的内容依次保存到堆栈中，然后将 **IE** 标志复位为“0”，并将向量的内容(003EH, 003FH)装入 PC 并转移。

通过执行 **RETB** 指令，可从中断返回。

注意事项 禁止使用 **RETI** 指令从软件中断返回。

14.4.3 多重中断服务

多重中断服务是指在执行一个中断服务程序时，又响应另一个优先级别更高的中断请求的服务。除非选择中断请求响应允许状态(**IE** = 1)，否则不会产生多重中断服务。响应中断请求时，禁止响应其他中断请求(**IE** = 0)。因此，若要允许多重中断服务，则必须在中断服务期间，执行 **EI** 指令将 **IE** 标志置“1”，才可以允许响应其他中断请求。

此外，即使允许中断，也不一定允许多重中断服务，这是因为受到中断优先级的控制。中断控制分为两种：默认优先级控制和可编程优先级控制。可编程优先级控制用于多重中断服务。

在中断允许状态中，若产生的中断请求的优先级与当前处理中的中断请求相等或高于它，则响应多重中断服务。若低于当前处理中的中断请求的优先级，则不响应多重中断服务。因禁止中断或优先级低而被禁止的中断请求将处于等待状态。当前中断结束服务时，至少执行一条主程序指令后才会响应等待状态中的中断请求。

表 14-5 给出多重中断服务允许的中断请求关系，图 14-10 给出多重中断服务示例。

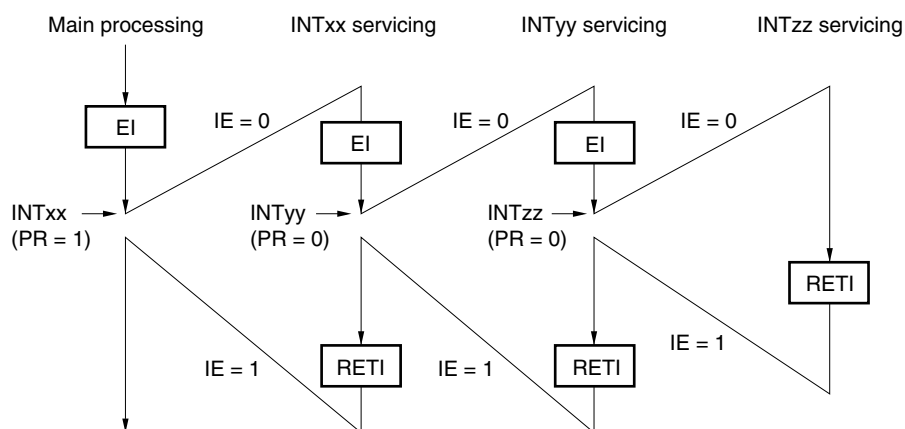
表 14-5. 中断服务期间多重中断服务允许的中断请求之间关系

多重中断请求 服务中断		可屏蔽中断请求				软件中断请求
		PR = 0		PR = 1		
		IE = 1	IE = 0	IE = 1	IE = 0	
可屏蔽中断	ISP = 0	○	×	×	×	○
	ISP = 1	○	×	○	×	○
软件中断		○	×	○	×	○

- 备注 1.** ○: 允许多重中断服务
- 2.** ×: 禁止多重中断服务
- 3.** **ISP** 和 **IE** 为 **PSW** 中的标志位。
ISP = 0: 高优先级中断服务中。
ISP = 1: 没有响应的中断请求，或服务中的低优先级中断。
IE = 0: 禁止响应中断请求。
IE = 1: 允许响应中断请求。
- 4.** **PR** 为 **PR0L**、**PR0H** 和 **PR1L** 的标志位。
PR = 0: 较高优先级别
PR = 1: 较低优先级别

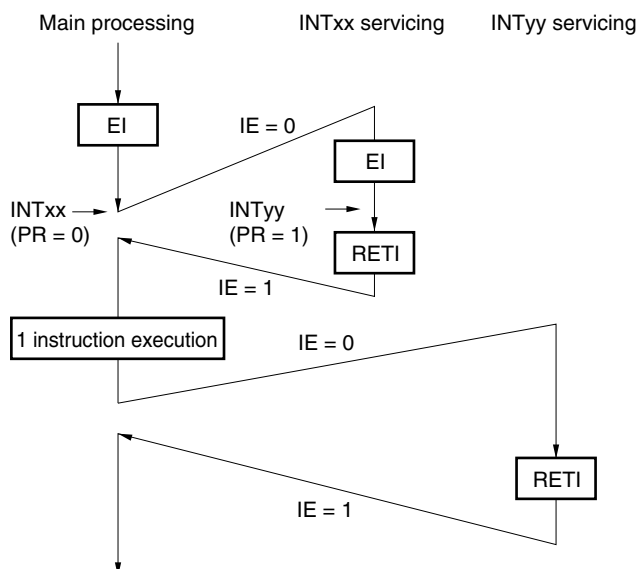
图 14-10. 多重中断服务示例 (1/2)

示例 1. 产生 2 次多重中断服务



INTxx 中断服务期间，响应 INTyy 和 INTzz 两个中断请求，产生多重中断服务。响应每个中断请求之前，必须先执行 EI 指令，才允许响应中断请求。

示例 2. 由于优先级控制没有产生的多重中断服务



中断 INTxx 服务中产生的中断请求 INTyy，由于其优先级低于 INTxx，不产生多重中断服务。中断请求 INTyy 处于等待状态，执行一条主程序指令后响应该中断请求。

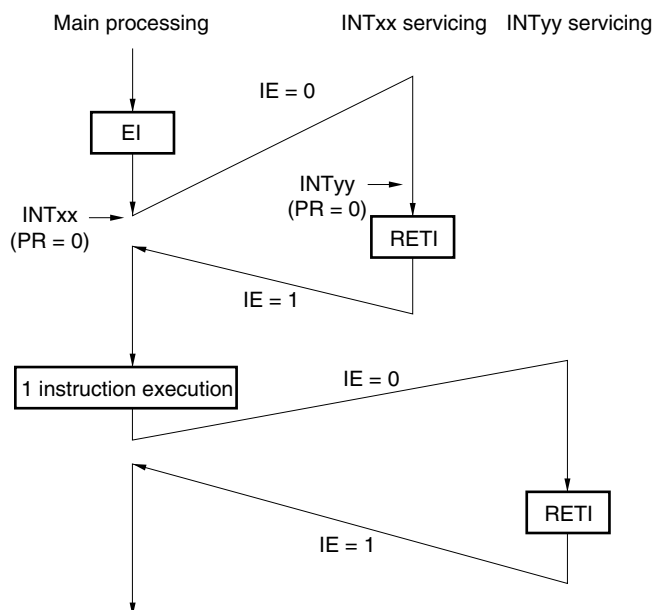
PR = 0: 较高优先级

PR = 1: 较低优先级

IE = 0: 禁止响应中断请求

图 14-10. 多重中断服务示例 (2/2)

示例 3. 由于不允许中断没有产生多重中断服务



在 INTxx 中断服务期间禁止中断（不执行 EI 指令），因此不响应中断请求 INTyy，不产生多重中断服务。中断请求 INTyy 处于等待状态，执行一条主程序指令后响应该中断请求。

PR = 0: 较高优先级别

IE = 0: 禁止响应中断请求

14.4.4 保留中断请求

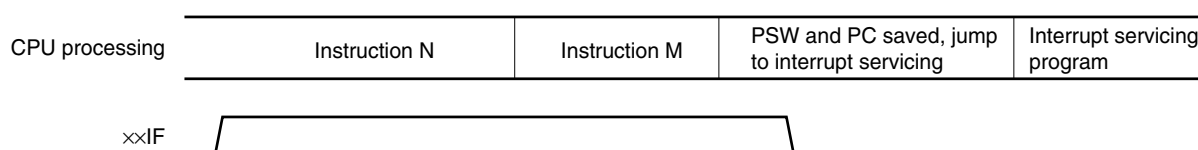
在某些指令执行期间，即使出现中断请求，请求响应也要保留等待状态，直到下一条指令执行结束。这类指令（中断请求的保留指令）如下所示。

- MOV PSW, #byte
- MOV A, PSW
- MOV PSW, A
- MOV1 PSW. bit, CY
- MOV1 CY, PSW. bit
- AND1 CY, PSW. bit
- OR1 CY, PSW. bit
- XOR1 CY, PSW. bit
- SET1 PSW. bit
- CLR1 PSW. bit
- RETB
- RETI
- PUSH PSW
- POP PSW
- BT PSW. bit, \$addr16
- BF PSW. bit, \$addr16
- BTCLR PSW. bit, \$addr16
- EI
- DI
- 用于 IF0L, IF0H, IF1L, MK0L, MK0H, MK1L, PR0L, PR0H 和 PR1L 寄存器的多重指令。

注意事项 **BRK** 指令不属于上述列出的中断请求保持指令。但通过执行 **BRK** 指令激活的软件中断，可将 **IE** 标志清零。因此，即使在执行 **BRK** 指令期间产生的可屏蔽中断请求，也不会被响应。

图 14-11 给出等待状态中的中断请求时序。

图 14-11. 保留中断请求



- 备注 1.** 指令 N: 保留中断请求指令
- 2.** 指令 M: 除中断请求保持指令之外的指令
- 3.** xxPR（优先级别）的值不会影响 IF（中断请求）的操作。

第十五章 待机功能

15.1 待机功能和配置

15.1.1 待机功能

待机功能用于降低系统的工作电流。提供以下两种模式：

(1) HALT 模式

通过执行 HALT 指令设置 HALT 模式。HALT 模式下，CPU 操作时钟停止。若设置 HALT 模式之前，运行高速系统时钟振荡器、内部高速振荡器、内部低速振荡器或副系统时钟振荡器，则设置后每个时钟会继续振荡。虽然 HALT 模式的功耗高于 STOP 模式，但可以非常有效地进行对中断请求产生后立即重启操作和频繁的间歇操作。

(2) STOP 模式

通过执行 STOP 指令设置 STOP 模式。在 STOP 模式中，通过停止高速系统时钟振荡电路和内部高速振荡电路，来停止整个系统操作，从而大幅度降低 CPU 功耗。

通过中断请求清除该模式，因此，可执行间歇操作。但选择 X1 时钟时，STOP 模式解除后，为了确保振荡器振荡稳定需要一段等待时间，因此，若产生中断请求后需要立即处理，则应选择 HALT 模式。

在这两种模式下，可以保留待机模式设置之前的所有寄存器、标志和数据存储器的内容。也可以保留 I/O 端口输出锁存器和输出缓冲器的状态。

- 注意事项**
1. 切换到 STOP 模式时，在执行 STOP 指令之前必须停止使用主系统时钟的外围硬件的操作。
 2. 使用待机功能时，建议通过以下步骤降低 A/D 转换器的工作电流：首先，通过 A/D 转换器模式寄存器 0(ADM0)的第 7 位(ADCS)和第 0 位(ADCE)清零，停止 A/D 转换器操作后，执行 STOP 指令。
 3. 执行 STOP 指令之前，停止运算放大器。

15.1.2 寄存器控制待机功能

由以下 2 个寄存器控制待机功能。

- 振荡稳定时间计数状态寄存器(OSTS)
- 振荡稳定时间选择寄存器(OSTS)

注意事项 有关控制操作/停止、切换时钟的寄存器,参阅第五章 时钟发生电路。

(1) 振荡稳定时间计数状态寄存器(OSTC)

该寄存器显示 X1 时钟振荡稳定时间计数器的计数状态。CPU 时钟使用内部高速振荡时钟开始 X1 时钟振荡时,可以检测出 X1 时钟振荡稳定时间。

可用 1 位或 8 位存储器操作指令设置 OSTC。

复位解除后(通过 RESET 输入、POC、LVI 和 WDT 进行复位),通过 STOP 指令和 MSTOP(MOC 寄存器第 7 位) = 1, 将 OSTC 清零(00H)。

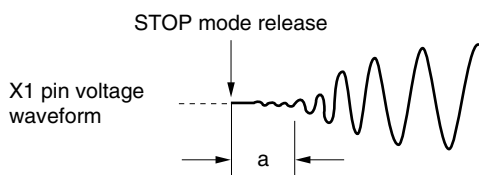
图 15-1. 振荡稳定时间计数状态寄存器(OSTS)的格式

地址: FFA3H 复位后: 00H R

符号	7	6	5	4	3	2	1	0
OSTC	0	0	0	MOST11	MOST13	MOST14	MOST15	MOST16

MOST11	MOST13	MOST14	MOST15	MOST16	振荡稳定时间状态	
						$f_x = 10 \text{ MHz}$
1	0	0	0	0	$2^{11}/f_x \text{ min.}$	204.8 $\mu\text{s min.}$
1	1	0	0	0	$2^{13}/f_x \text{ min.}$	819.2 $\mu\text{s min.}$
1	1	1	0	0	$2^{14}/f_x \text{ min.}$	1.64 ms min.
1	1	1	1	0	$2^{15}/f_x \text{ min.}$	3.27 ms min.
1	1	1	1	1	$2^{16}/f_x \text{ min.}$	6.55 ms min.

- 注意事项
1. 经过上述时间后, 从 **MOST11** 开始位逐次置“1”, 并一直保持为“1”。
 2. 振荡稳定时间计数器只能检测出由 **OSTS** 设置的振荡稳定时间。内部高速振荡时钟用作 **CPU** 时钟时, 若进入 **STOP** 模式, 则解除其模式时, 需按以下方式设置振荡稳定时间。
 - 预期 **OSTC** 振荡稳定时间 $\leq$ 通过 **OSTS** 设置的振荡稳定时间
 因此, 应注意 **STOP** 模式解除后的 **OSTS**, 只能达到 **OSTS** 设置的振荡稳定时间状态。
 3. **X1** 时钟振荡稳定等待时间不包括时钟振荡状态之前的时间 (下图中“a”所表示的部分)。

备注 f_x : X1 时钟振荡频率**(2) 振荡稳定时间选择寄存器(OSTS)**

STOP 模式解除时, 该寄存器用来选择 **X1** 时钟振荡稳定等待时间。

当 **CPU** 时钟选用 **X1** 时钟时, **STOP** 模式解除后, 需等待由 **OSTS** 所设置的时间。

当 **CPU** 时钟选用内部高速振荡时钟时, **STOP** 模式解除后, 通过 **OSTC** 确认是否已达到所需的振荡稳定时间。通过 **OSTC** 设置时间, 可检测出振荡稳定时间。

可按 1 位或 8 位存储器操作指令设置 **OSTS**。

复位信号产生时, **OSTS** 设置为 05H。

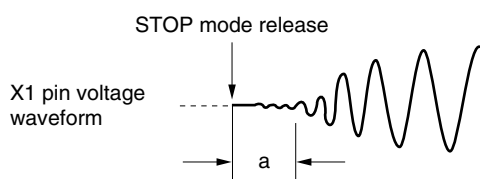
图 15-2. 振荡稳定时间选择寄存器(OSTS)的格式

地址: FFA4H 复位后: 05H R/W

符号	7	6	5	4	3	2	1	0
OSTS	0	0	0	0	0	OSTS2	OSTS1	OSTS0

OSTS2	OSTS1	OSTS0	振荡稳定时间选择	
				$f_x = 10 \text{ MHz}$
0	0	1	$2^{11}/f_x$	204.8 μs
0	1	0	$2^{13}/f_x$	819.2 μs
0	1	1	$2^{14}/f_x$	1.64 ms
1	0	0	$2^{15}/f_x$	3.27 ms
1	0	1	$2^{16}/f_x$	6.55 ms
其它			禁止设置	

- 注意事项
1. 若 CPU 使用 X1 时钟时, 设置 STOP 模式, 则需执行 STOP 指令之前设置 OSTS。
 2. X1 时钟振荡稳定时间内, 禁止改变 OSTS 的值。
 3. 振荡稳定时间计数器只能检测出由 OSTS 设置的振荡稳定时间。内部高速振荡时钟用作 CPU 时钟时, 若进入 STOP 模式, 则解除其模式时, 需按以下方式设置振荡稳定时间。
 - 预期 OSTC 振荡稳定时间 $\leq$ 通过 OSTS 设置的振荡稳定时间
 因此, 应注意 STOP 模式解除后的 OSTS, 只能达到 OSTS 设置的振荡稳定时间状态。
 4. X1 时钟振荡稳定等待时间不包括时钟振荡状态之前的时间 (下图中"a"所表示的部分)。



备注 fx:X1 时钟振荡频率

15.2 待机功能操作

15.2.1 HALT 模式

(1) HALT 模式

通过执行 HALT 指令，设置 HALT 模式。无论设置前的 CPU 时钟是高速系统时钟，还是内部高速振荡时钟，都可以设置 HALT 模式。

HALT 模式下的操作状态如下所示。

表 15-1. HALT 模式下运行状态

HALT 模式设置 项目			CPU 使用主系统时钟执行 HALT 指令时		
			CPU 使用内部高速振荡时钟 (f _{IH})时	CPU 使用 X1 时钟(f _X)时	当 CPU 使用内部主系统时钟 (f _{IH})时
系统时钟			停止 CPU 时钟		
主系统时钟	f _{IH}	操作继续(禁止停止)		保留设置 HALT 模式前的状态	
	f _X	保留设置 HALT 模式前的状态		操作继续(禁止停止)	保留设置 HALT 模式前的状态
	f _{EXCLK}	由外部时钟输入的操作或停止			操作继续(禁止停止)
	f _{IL}	保留设置 HALT 模式前的状态			
CPU			操作停止		
闪存					
RAM			保留设置 HALT 模式前的状态		
端口（锁存）					
16 位定时器/事件计数器 00			可操作		
8 位定时器/事件计数器 51					
8 位定时器 H1					
看门狗定时器			可操作通过选项字节设置“允许软件停止内部低速振荡器”时，停止提供给看门狗定时器的时钟。		
A/D 转换器			可操作		
运算放大器 0, 1					
比较器					
串行接口 UART0					
上电清除功能					
低电压检测功能					
外部中断					

备注 f_{IH}: 内部高速振荡时钟,
f_{EXCLK}: 外部主系统时钟,

f_X: X1 时钟
f_{IL}: 内部低速振荡时钟

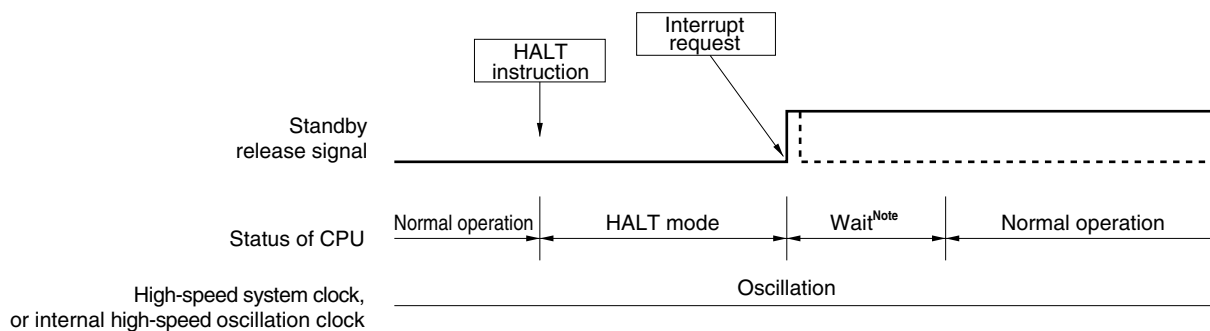
(2) HALT 模式解除

通过以下两种模式可解除 HALT 模式。

(a) 由非屏蔽中断请求解除

产生非屏蔽中断请求时，解除 HALT 模式。若允许中断响应，则执行向量中断服务。若禁止中断响应，则执行下一条地址指令。

图 15-3. 通过产生中断请求，解除 HALT 模式



注 等待时间如下：

- 执行向量中断服务：11 或 12 个时钟
- 不执行向量中断服务：4 或 5 个时钟

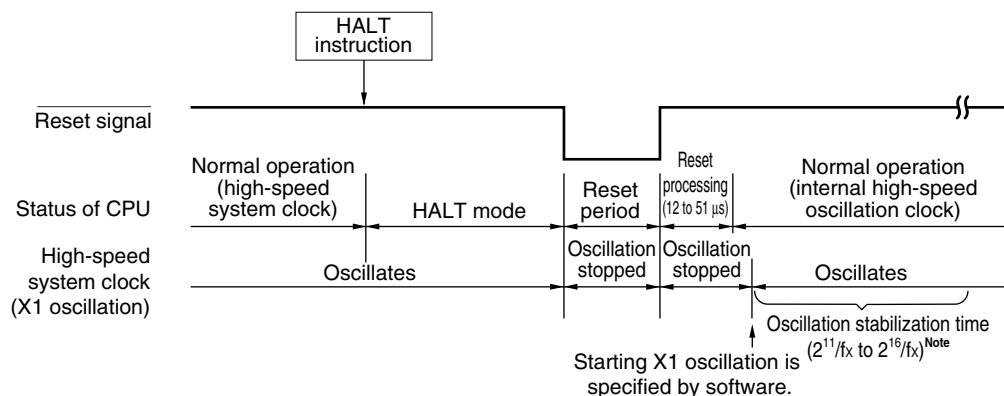
注意事项 虚线表示解除待机模式后中断请求的响应情况。

(b) 通过产生复位信号解除 HALT 模式

复位信号产生时，解除 HALT 模式，然后与正常的复位操作一样转移到复位向量地址后，开始执行程序。

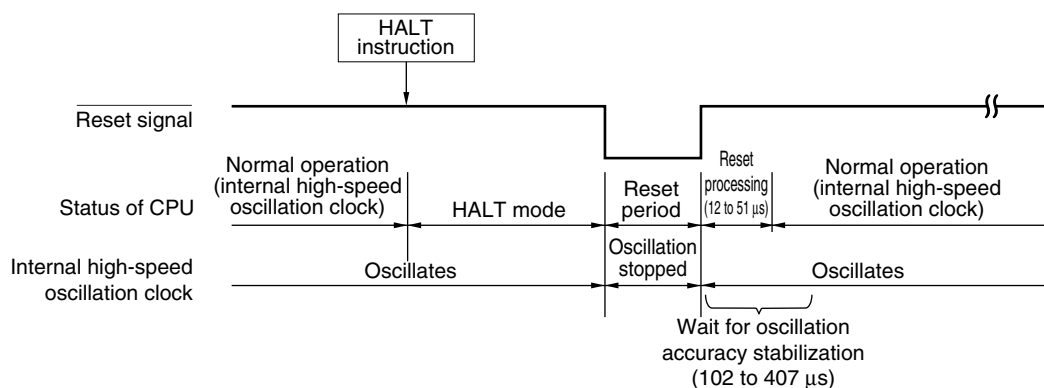
图 15-4. 通过复位解除 HALT 模式

(1) 高速系统时钟用作 CPU 时钟时



注 外部主系统时钟(f_{EXCLK})用作高速系统时钟时，不需要振荡稳定时间。

(2) 内部高速振荡时钟用作 CPU 时钟时



备注 f_x : X1 时钟振荡频率

表 15-2. HALT 模式下中断请求相对应的操作

解除源	MK _{xx}	PR _{xx}	IE	ISP	操作
可屏蔽中断请求	0	0	0	×	执行下一个指令地址
	0	0	1	×	执行中断服务
	0	1	0	1	执行下一个指令地址
	0	1	×	0	
	0	1	1	1	执行中断服务
	1	×	×	×	保留 HALT 模式
复位	—	—	×	×	复位处理

×:忽略

15.2.2 STOP 模式

(1) STOP 模式设置和运行状态

通过执行 STOP 指令设置 STOP 模式。仅在设置前的 CPU 时钟为主系统时钟时才可设置 STOP 模式。

注意事项 由于中断请求信号用于解除待机模式，若中断源的中断请求标志已设置且中断屏蔽标志已清零，则立即解除待机模式。因此，执行 STOP 指令后，立即将 STOP 模式复位到 HALT 模式，并且经过 OSTS 设置的等待时间后，系统返回操作模式。

STOP 模式下，运行状态如下所示。

表 15-3. STOP 模式下运行状态

STOP 模式设置 项目		CPU 使用主系统时钟执行 STOP 指令时		
		CPU 使用内部高速振荡时钟 (f _{IH})时	CPU 使用 X1 时钟(f _x)时	当 CPU 使用内部主系统时钟 (f _{IH})时
系统时钟		停止 CPU 时钟		
主系统时钟	f _{IH}	停止		
	f _x			
	f _{EXCLK}	无效输入		
	f _{IL}	保留设置 STOP 模式前的状态		
CPU		操作停止。		
闪存				
RAM		保留设置 STOP 模式前的状态		
端口（锁存）				
16 位定时器/事件计数器 00		操作停止。		
8 位定时器/事件计数器 51		TI51 用作计数时钟时，可操作。		
8 位定时器 H1		仅在选择f _{IL} , f _{IL} /2 ⁷ , f _{IL} /2 ⁹ 用作计数时钟时，可操作。		
看门狗定时器		可操作通过选项字节设置“允许软件停止内部低速振荡器”时，停止提供给看门狗定时器的时钟。		
A/D 转换器		操作停止。		
运算放大器 0,1		可操作		
比较器		禁止操作		
串行接口 UART0		在 8 位定时器/事件计数器 51 操作期间，仅在选择 TM51 输出用作串行时钟时，可操作		
上电清除功能		可操作		
低电压检测功能				
外部中断				

备注

f_{IH}: 内部高速振荡时钟，

f_x: X1 时钟

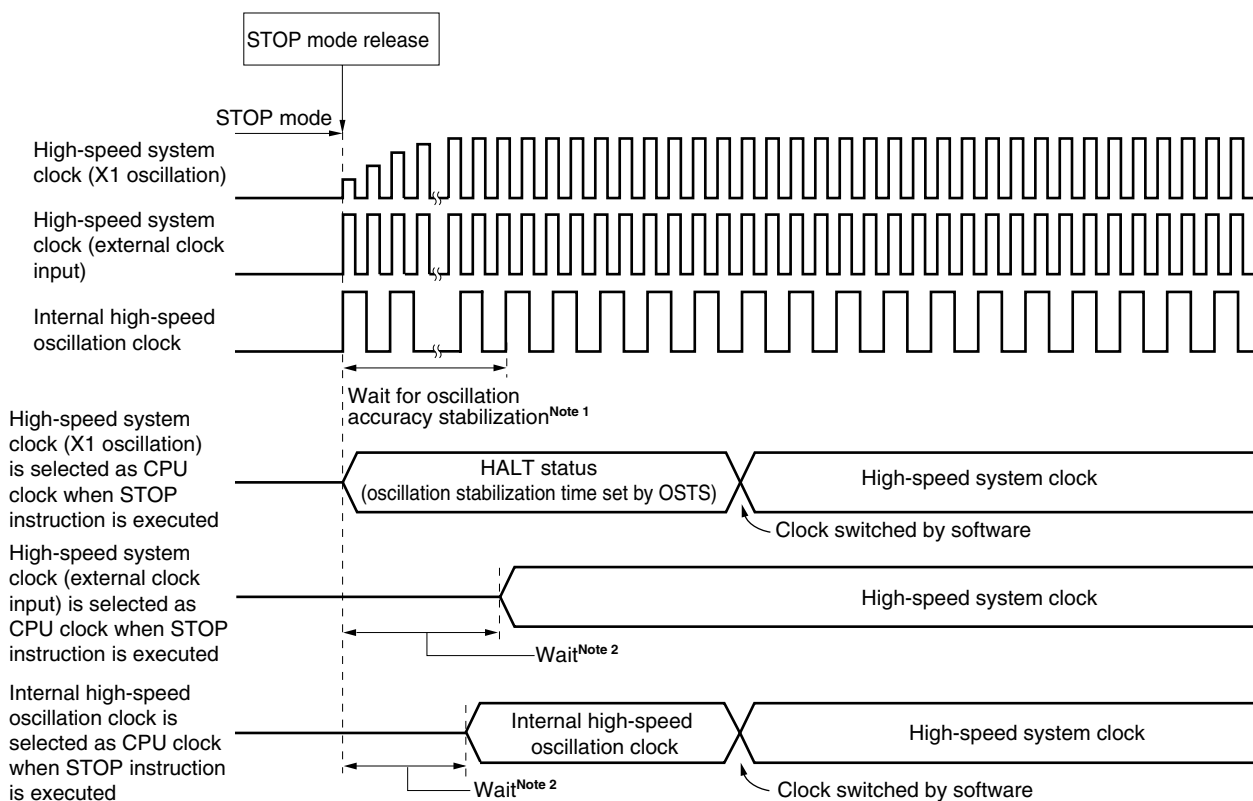
f_{EXCLK}: 外部主系统时钟，

f_{IL}: 内部低速振荡时钟

- 注意事项
1. 若使用 **STOP** 模式下停止操作的外围硬件，以及 **STOP** 模式解除后停止振荡时钟的外围硬件，则必须重新启动外围硬件。
 2. 即使通过选项字节选择“允许软件停止内部低速振荡器”，在 **STOP** 模式下的内部低速振荡时钟仍持续 **STOP** 模式设置前的状态。若在 **STOP** 模式下，通过软件停止内部低速振荡器后，执行 **STOP** 指令。
 3. 通过高速系统时钟(X1 振荡)进行 CPU 操作时，若要缩短 **STOP** 模式解除后的振荡稳定时间，则在执行 **STOP** 指令之前，应按照以下步骤将 CPU 时钟切换到内部高速振荡时钟。
<1> **RSTOP** 置“0”(内部高速振荡器开始振荡) → <2> **MCM0** 置“0”(将 CPU 从 X1 振荡切换到内部高速振荡) → <3> 确认 **MCS** 为“0”(确认 CPU 时钟) → <4> 确认 **RSTS** 为“1”(确认内部高速振荡操作) → <5> 执行 **STOP** 指令
STOP 模式解除后，将 CPU 时钟从内部高速振荡时钟切换到高速系统时钟(X1 振荡)之前，应使用振荡稳定时间计数器的状态寄存器(**OSTC**)检测振荡稳定时间。
 4. 必须在内部高速振荡器稳定操作(**RSTS** = 1)状态下，执行 **STOP** 指令。

(2) STOP 模式解除

图 15-5. STOP 模式解除时的操作时序（产生非屏蔽中断请求时）



注 1. 振荡精确稳定等待时间如下：

- RMC 寄存器 = 00H: 102 至 407 μs
- RMC 寄存器 = 56H/59H: 120 至 481 μs

2. 待机时间如下：

- 执行向量中断服务：17 或 18 个时钟
- 不执行向量中断服务：11 或 12 个时钟

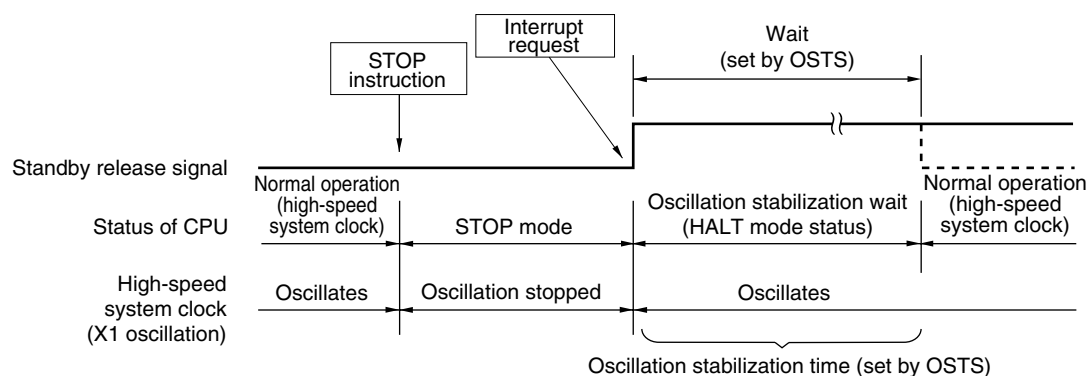
通过以下两种方式可解除 STOP 模式。

(a) 由非屏蔽中断请求解除

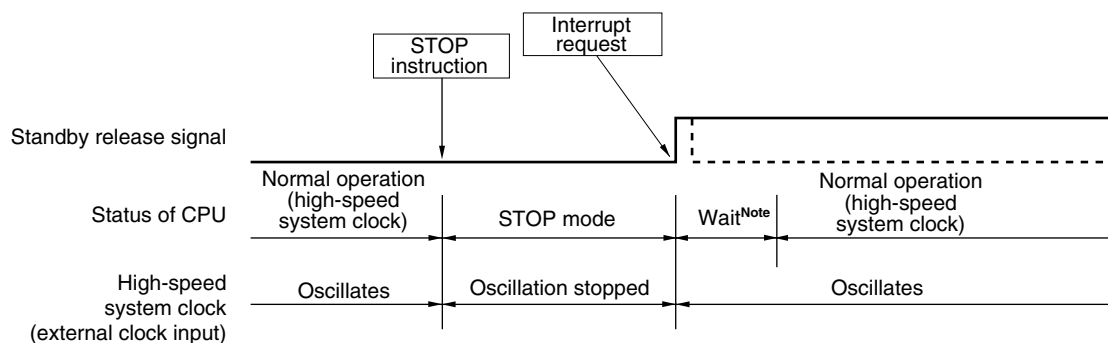
产生非屏蔽中断请求时，解除 STOP 模式。振荡稳定时间过后，若允许响应中断，则执行向量中断服务。若禁止中断响应，则执行下一条地址指令。

图 15-6. 通过产生中断请求解除 STOP 模式 (1/2)

(1) 高速系统时钟(X1 振荡) 用作 CPU 时钟时



(2) 高速系统时钟(外部时钟输入) 用作 CPU 时钟时



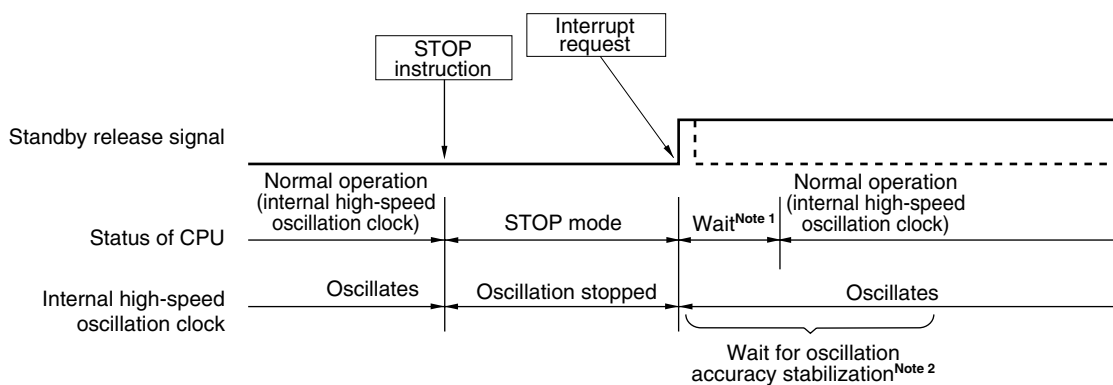
注 待机时间如下：

- 执行向量中断服务：17 或 18 个时钟
- 不执行向量中断服务：11 或 12 个时钟

注意事项 虚线表示解除待机模式后中断请求的响应情况。

图 15-6. 通过产生中断请求解除 STOP 模式 (2/2)

(3) 内部高速振荡时钟用作 CPU 时钟时



注 1. 待机时间如下。

- 执行向量中断服务: 17 或 18 个时钟
- 不执行向量中断服务: 11 或 12 个时钟

2. 振荡精确稳定等待时间如下:

- RMC 寄存器 = 00H: 102 至 407μs
- RMC 寄存器 = 56H/59H: 120 至 481μs

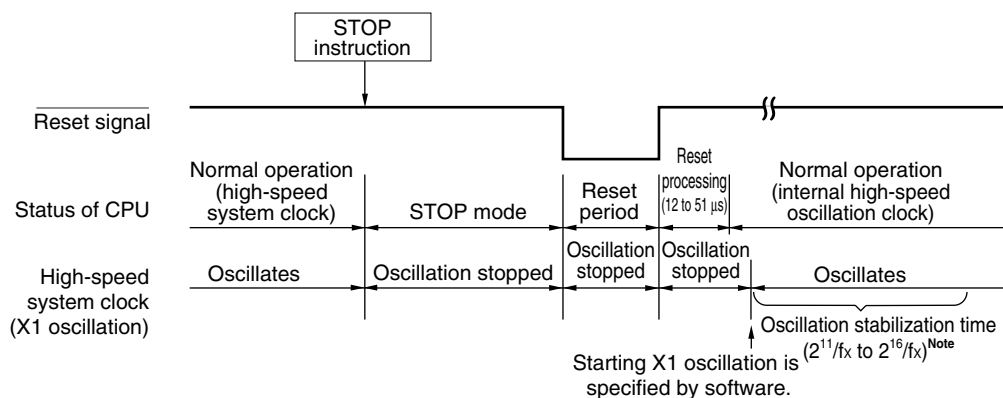
注意事项 虚线表示解除待机模式后中断请求的响应情况。

(b) 通过产生复位信号的接触

复位信号产生时, 解除 STOP 模式, 然后与正常的复位操作一样转移到复位向量地址后, 开始执行程序。

图 15-7. 通过复位解除 STOP 模式

(1) 高速系统时钟用作 CPU 时钟时



注 外部主系统时钟(f_{EXCLK})用作高速系统时钟时，不需要振荡稳定时间。

备注 f_x :X1 时钟振荡频率

(2) 内部高速振荡时钟用作 CPU 时钟时

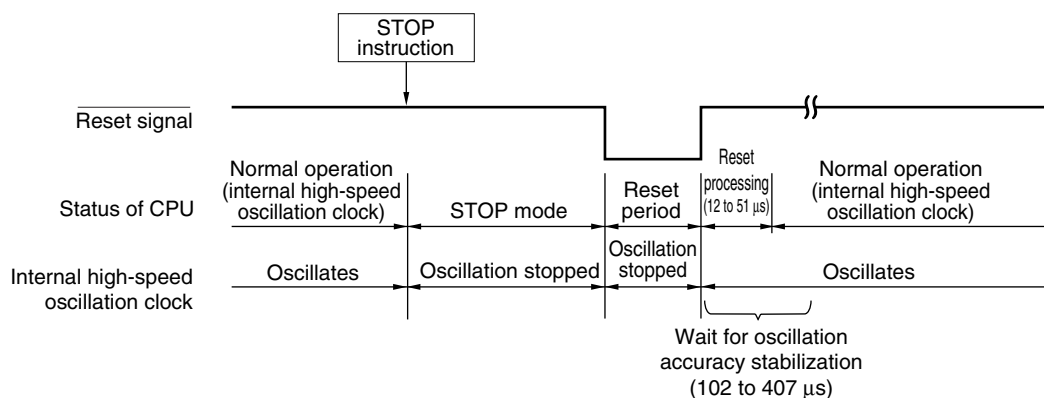


表 15-4. STOP 模式下中断请求相对应的操作

解除源	MK _{xx}	PR _{xx}	IE	ISP	操作
可屏蔽中断请求	0	0	0	×	执行下一个指令地址
	0	0	1	×	执行中断服务
	0	1	0	1	执行下一个指令地址
	0	1	×	0	
	0	1	1	1	执行中断服务
	1	×	×	×	保留 STOP 模式
复位	—	—	×	×	复位处理

×:忽略

第十六章 复位功能

以下 4 种操作用来产生复位信号。

- (1) 由 $\overline{\text{RESET}}$ 引脚产生的外部复位输入
- (2) 由看门狗定时器的程序环路检测产生的内部复位
- (3) 上电清除(POC)电路的电源电压与检测电压相比较产生的内部复位
- (4) 低电源检测电路(LVI)的电源电压与检测电压相比较产生的内部复位

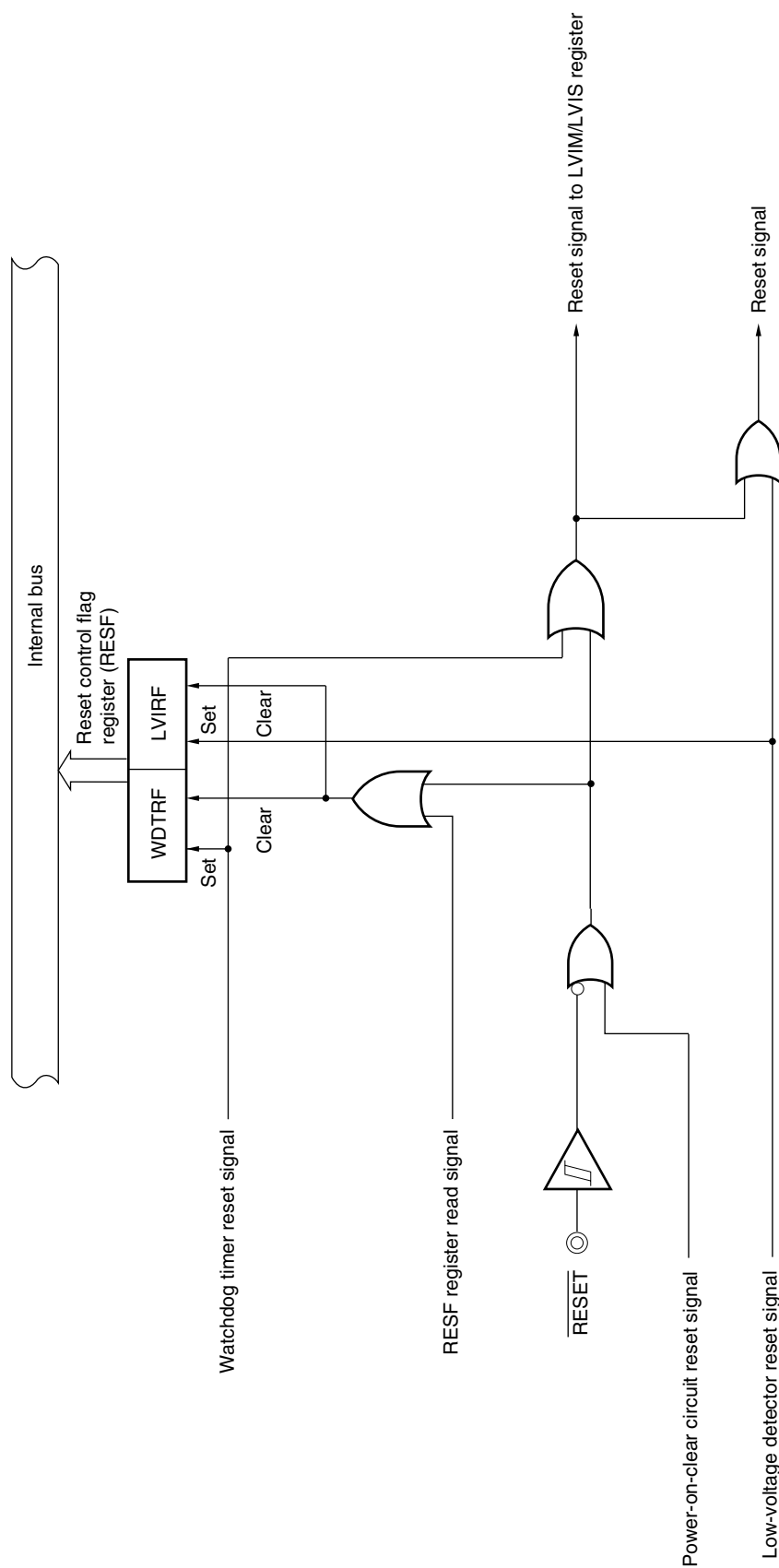
外部复位与内部复位一样，复位信号产生时，从地址 0000H 和 0001H 开始执行程序。

$\overline{\text{RESET}}$ 引脚为低电平、看门狗定时器溢出、或 POC 和 LVI 电路电压检测时，产生复位。各硬件的设置状态如表 16-1 和 16-2 所示。复位信号产生期间或复位解除后振荡稳定时间内，每个引脚均为高电阻。

$\overline{\text{RESET}}$ 引脚输入低电平时，器件复位。 $\overline{\text{RESET}}$ 引脚输入高电平产生的复位自动解除后，使用高速内部振荡时钟开始执行程序。由看门狗定时器产生的复位自动解除后，使用内部高速振荡时钟开始执行程序（参阅图 16-2 至 16-4）。在 $V_{DD} \geq V_{POR}$ 或 $V_{DD} \geq V_{LVI}$ 时，由 POC 和 LVI 电路的电源检测产生的复位自动解除后，使用内部高速振荡时钟开始执行程序（参阅第十七章 上电清除电路 和 第十八章 低电压检测电路）。

- 注意事项**
1. 外部复位时，输入 $\overline{\text{RESET}}$ 引脚的低电平应大于等于 10 μs 。
(若在上电时进行外部复位，则操作电压范围外($V_{DD} < 1.8 \text{ V}$)期间，在 10 μs 不进行计数。但在 POC 解除之前，持续低电平输入。)
 2. 复位信号产生期间，X1 时钟、内部高速振荡时钟和内部低速振荡时钟停止振荡。外部主系统时钟输入无效。
 3. 通过复位解除 STOP 模式时，复位输入期间保持 STOP 模式下的 RAM 内容。但由于各 SFR 的初始化，端口引脚变为高阻抗。

图 16-1. 复位功能的框图



注意事项 LVI 电路的内部复位时，LVI 电路不能复位。

备注 1. LVIM:低电压检测寄存器

2. LVIS:低电压检测电平选择寄存器

图 16-2. $\overline{\text{RESET}}$ 输入的复位时序

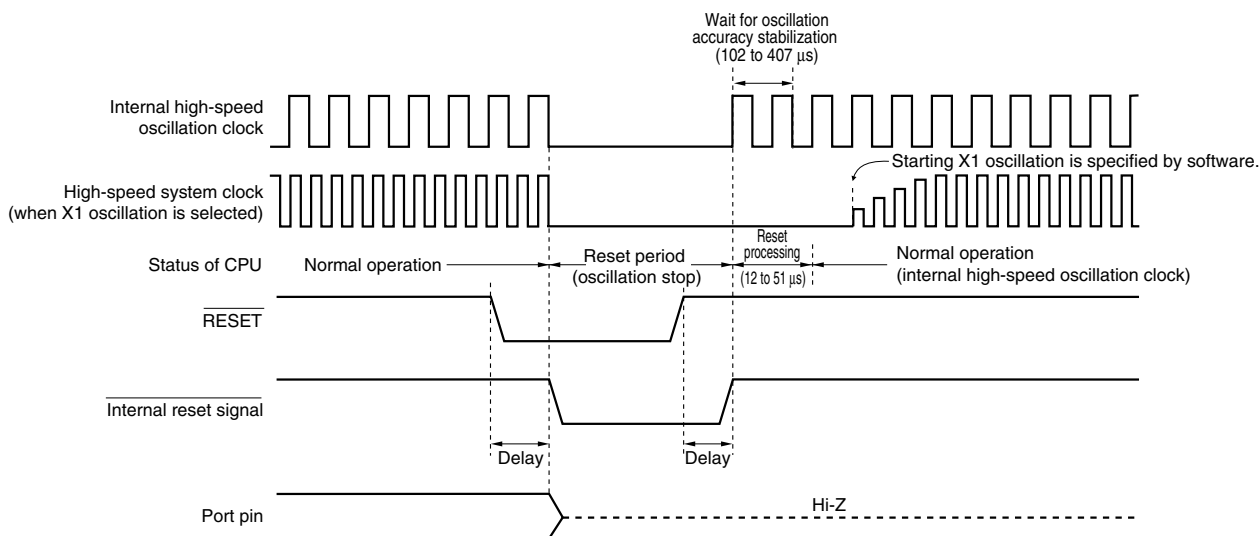
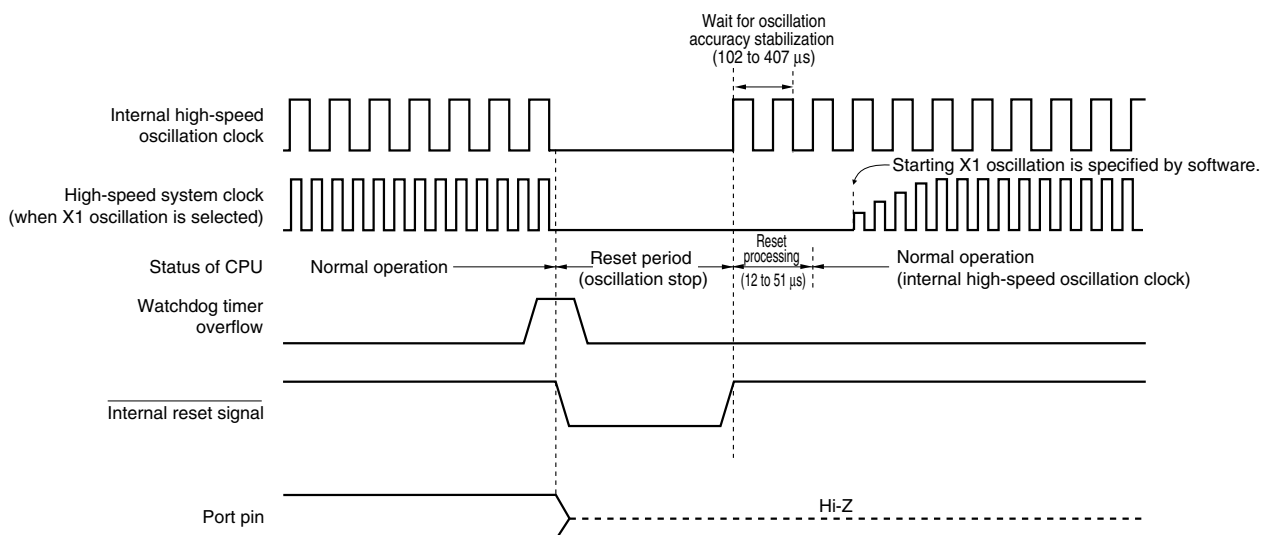
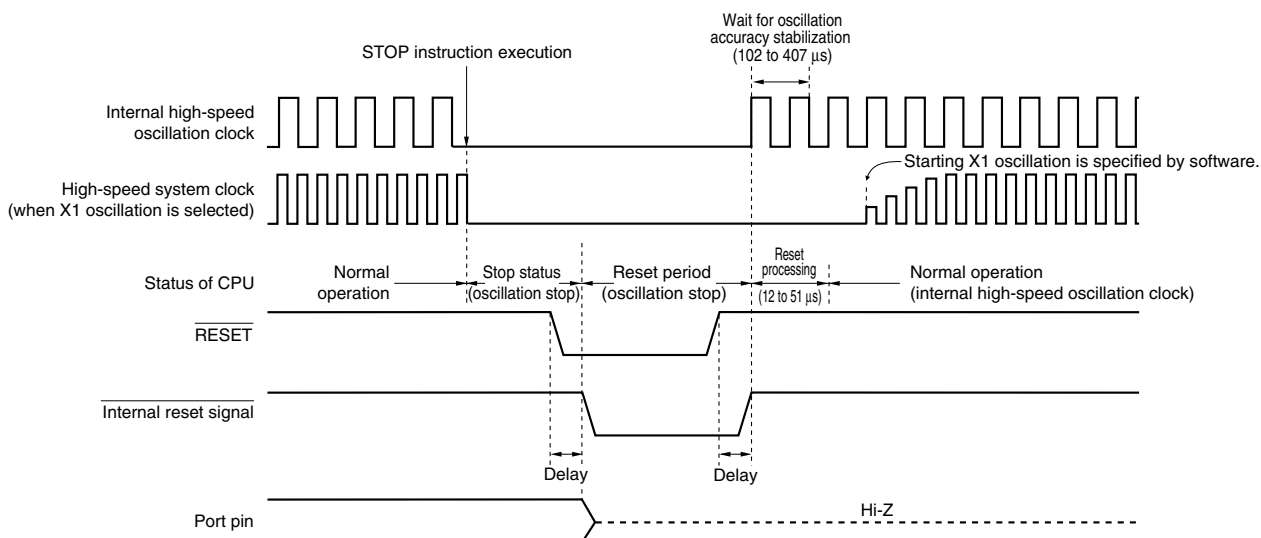


图 16-3. 由看门狗定时器溢出而产生复位的时序



注意事项 可以通过看门狗定时器的内部复位执行看门狗定时器的复位。

图 16-4. STOP 模式下通过 RESET 输入的复位时序



备注 有关上电清除电路和低电压检测电路的复位时序，参阅第十七章 上电清除电路 和 第十八章 低电压检测电路。

表 16-1. 复位期间运行状态

项目			复位期间
系统时钟			停止 CPU 时钟
主系统时钟	f _{IH}	停止操作	
	f _X	停止操作（X1 和 X2 引脚为输入端口模式）	
	f _{EXCLK}	时钟输入无效(EXCLK 引脚为输入端口模式)	
f _{IL}		停止操作	
CPU			
闪存			
RAM			
			停止操作(但高于上电清除检测电压时，保持原值。)
端口（锁存器）			停止操作
16 位定时器/事件计数器 00			
8 位定时器/事件计数器 51			
8 位定时器 H1			
看门狗定时器			
A/D 转换器			
运算放大器 0(AMP0)			
运算放大器 1(AMP1)			
比较器			
串行接口 UART0			
外部中断			
上电清除功能			允许操作
低电压检测功能			停止操作(但,LVI 复位时，持续操作)
On-chip 调试功能			停止操作

备注

f_{IH}: 内部高速振荡时钟,f_X: X1 时钟f_{EXCLK}: 外部主系统时钟,f_{IL}: 内部低速振荡时钟

表 16-2. 复位响应后软件状态 (1/3)

硬件		复位响应后状态 ^{注1}
程序计数器 (PC)		设置复位向量表的内容 (0000H, 0001H)
堆栈指针 (SP)		不确定
程序状态字 (PSW)		02H
RAM	数据寄存器	不确定 ^{注2}
	通用寄存器	不确定 ^{注2}
端口寄存器 2, 3, 12 (P2, P3, P12) (输出锁存器)		00H
端口模式寄存器 2, 3, 12 (PM2, PM3, PM12)		FFH
上拉电阻选择寄存器 3 (PU3)		00H
上拉电阻选择寄存器 12 (PU12)		20H
复位引脚模式寄存器 (RSTMASK)		00H
内部寄存器容量切换寄存器 (IMS)		CFH ^{注3}

注 1. 复位信号产生或振荡稳定时间等待期间的硬件状态中，只有 PC 内容不确定。复位后其他硬件状态保持不变。

2. 待机模式下复位后的状态保持原状。

3. 复位信号产生时，ROM 区域处于不稳定状态。因此，复位解除后，必须在每个产品上设置以下值。

产品	IMS	ROM 容量	内部高速 RAM 容量
μPD79F7023	42H	8 KB	512 字节
μPD79F7024	04H	16 KB	768 字节

表 16-2. 复位响应后软件状态 (2/3)

硬件		复位响应后状态 ^{注1}
时钟操作模式选择寄存器 (OSCCTL)		00H
处理器时钟控制寄存器(PCC)		01H
内部振荡模式寄存器(RCM)		80H
主 OSC 控制寄存器(MOC)		80H
主时钟模式寄存器(MCM)		00H
振荡稳定时间计数状态寄存器(OSTS)		00H
振荡稳定时间选择寄存器(OSTS)		05H
16 位定时器/事件计数器 00	定时器/计数器 00(TM00)	0000H
	捕捉/比较寄存器 000, 010 (CR000, CR010)	0000H
	模式控制寄存器 00(TMC00)	00H
	预定标模式寄存器 00 (PRM00)	00H
	捕捉/比较控制寄存器 00 (CRC00)	00H
	定时器输出控制寄存器 00(TOC00)	00H
8 位定时器/事件计数器 51	定时器/计数器 51(TM51)	00H
	比较寄存器 51 (CR51)	00H
	定时器时钟选择寄存器 51(TCL51)	00H
	模式控制寄存器 51(TMC51)	00H
8 位定时器 H1	比较寄存器 01, 11 (CMP01, CMP11)	00H
	模式寄存器(TMHMD1)	00H
	载波控制寄存器 1(TMCYC1)	00H
看门狗定时器	允许寄存器(WDTE)	1AH/9AH ^{注2}
A/D 转换器	8 位 A/D 转换结果寄存器 H(ADCRH)	00H
	模式寄存器(ADM)	00H
	模拟输入通道指定寄存器(ADS)	00H
	A/D 端口配置寄存器(ADPC)	00H
运算放大器 0(AMP0)	运算放大器控制寄存器(AMPM)	00H
运算放大器 1(AMP1)		
比较器	比较器控制寄存器(CMPCTL)	00H
	比较器端口配置寄存器(CMPPC)	00H
串行接口 UART0	接收缓冲寄存器 0(RXB0)	FFH
	发送缓冲寄存器 0(TXS0)	FFH
	异步串行接口操作模式寄存器 0(ASIM0)	01H
	异步串行接口接收错误状态寄存器 0 (ASIS0)	00H
	波特率发生器控制寄存器 0(BRGC0)	1FH

注 1. 复位信号产生或振荡稳定时间等待期间的硬件状态中，只有 PC 内容不明确。复位后其他硬件状态保持不变。

2. 通过设置选项字节来确定 WDTE 复位值。

表 16-2. 复位响应后软件状态 (3/3)

硬件		复位响应后状态 ^{注1}
复位功能	复位控制标志寄存器 (RESF)	00H ^{注2}
低电压检测电路	低电压检测寄存器(LVIM)	00H ^{注2}
	低电压检测电平选择寄存器(LVIS)	00H ^{注2}
中断	请求标志寄存器 0L, 0H, 1L (IF0L, IF0H, IF1L)	00H
	屏蔽标志寄存器 0L, 0H, 1L (IF0L, IF0H, IF1L)	FFH
	奇偶校验指定标志寄存器 0L, 0H, 1L (PR0L, PR0H, PR1L)	FFH
	外部中断上升沿允许寄存器(EGP)	00H
	外部中断下降沿允许寄存器(EGN)	00H
稳压器	稳压器模式控制寄存器(RMC)	00H

- 注 1. 复位信号产生或振荡稳定时间等待期间的硬件状态中，只有 PC 内容不明确。复位后其他硬件状态保持不变。
2. 地址值因产品的不同而有所差异。

复位源 寄存器		$\overline{\text{RESET}}$ 输入	通过 POC 复位	通过 WDT 复位	通过 LVI 复位 (LVI 默认启动功 能复位除外)	通过 LVI 默认启 动功能复位
RESF	WDTRF 标 志	清除 (0)	清除 (0)	置 (1)	保留	清除 (0)
	LVIRF 标志			保留	置 (1)	
LVIM		清除 (00H)	清除 (00H)	清除 (00H)	保留	清除 (00H)
LVIS						

16.1 确认复位源的寄存器

μPD79F7023, 79F7024 微控制器存在许多内部复位产生源。复位控制标志寄存器(RESF) 用于存储产生复位请求的复位源。

可按 8 位存储器操作指令读取 RESF。

$\overline{\text{RESET}}$ 输入、由上电清除(POC)产生的复位和读取 RESF 可将 RESF 清零(00H)。

图 16-5. 复位控制标志寄存器(RESF)格式

地址: FFACH 复位后: 00H^注 R

符号	7	6	5	4	3	2	1	0
RESF	0	0	0	WDTRF	0	0	0	LVIRF

WDTRF	看门狗定时器产生的内部复位请求(WDT)
0	不产生内部复位请求, RESF 清零。
1	产生内部复位请求

LVIRF	低电压检测电路产生的内部复位请求(LVI)
0	不产生内部复位请求, RESF 清零。
1	产生内部复位请求

备注 复位值因复位源的变化而不同。

注意事项 禁止通过 1 位存储器操作指令读取数据。

复位请求产生时的 RESF 状态如表 16-3 所示。

表 16-3. 复位请求产生时的 RESF 状态。

复位源 标志	$\overline{\text{RESET}}$ 输入	通过 POC 复位	通过 WDT 复位	通过 LVI 复位 (LVI 默认启动功能复位除外)	通过 LVI 默认启动功能复位
WDTRF	清除 (0)	清除 (0)	设置 (1)	保留	清除 (0)
LVIRF			保留	设置 (1)	

第十七章 上电清除电路

17.1 上电清除电路功能

上电清除电路(POC)具有以下功能。

(1) 1.59V POC 模式下(选项字节: LVISTART = 0)

- 上电时, 产生内部复位信号。供电电压(V_{DD})超过检测电压($V_{POC} = 1.59\text{ V} \pm 0.15\text{ V}$)时, 解除复位状态。
- 供电电压(V_{DD})和检测电压($V_{POC} = 1.59\text{ V} \pm 0.15\text{ V}$) 相匹配时, $V_{DD} < V_{POC}$ 时, 产生内部复位信号。 $V_{DD} \geq V_{POC}$ 时, 解除。

(2) 2.7 V/1.59 V POC 模式下(选项字节: LVISTART = 1)

- 上电时, 产生内部复位信号。供电电压(V_{DD})超过检测电压($V_{DDPOC} = 2.7\text{ V} \pm 0.2\text{ V}$)时, 解除复位状态。
- 供电电压(V_{DD})和检测电压($V_{POC} = 1.59\text{ V} \pm 0.15\text{ V}$) 相匹配时, 若 $V_{DD} < V_{POC}$, 则产生内部复位信号; 若 $V_{DD} \geq V_{DDPOC}$ 时, 则解除内部复位信号。

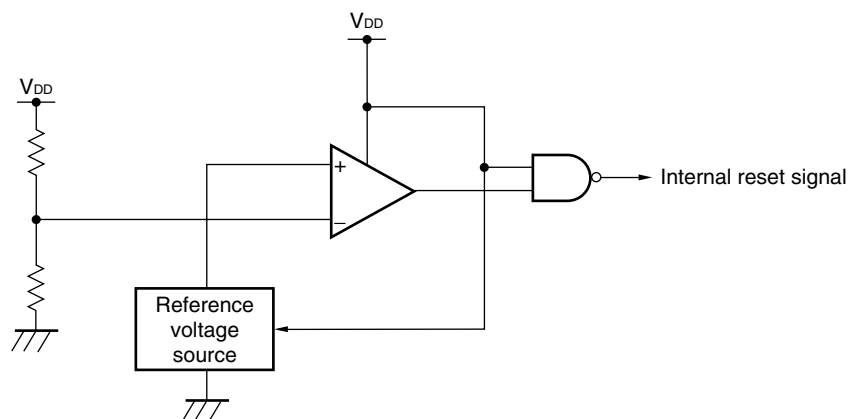
注意事项 若 POC 电路产生内部复位信号时, 复位控制标志寄存器(RESF)清零为 00H。

备注 μ PD79F7023, 79F7024 微控制器包括多种产生内部复位信号的硬件功能。由看门狗定时器(WDT)和低电压检测器(LVI)产生内部复位信号时, 表示其复位源的标志配置在复位控制标志寄存器(RESF)中。通过 WDT 或 LVI 产生内部复位信号时, RESF 不清为 00H, 标志位置“1”。有关 RESF 的详情, 参阅第十六章 复位功能。

17.2 上电清除电路的配置

上电清除电路系统图如图 17-1 所示。

图 17-1. 上电清除电路系统图



17.3 上电清除电路操作

- 上电时，产生内部复位信号。供电电压(V_{DD})超过检测电压($V_{POC} = 1.59\text{ V} \pm 0.15\text{ V}$)时，解除复位状态。

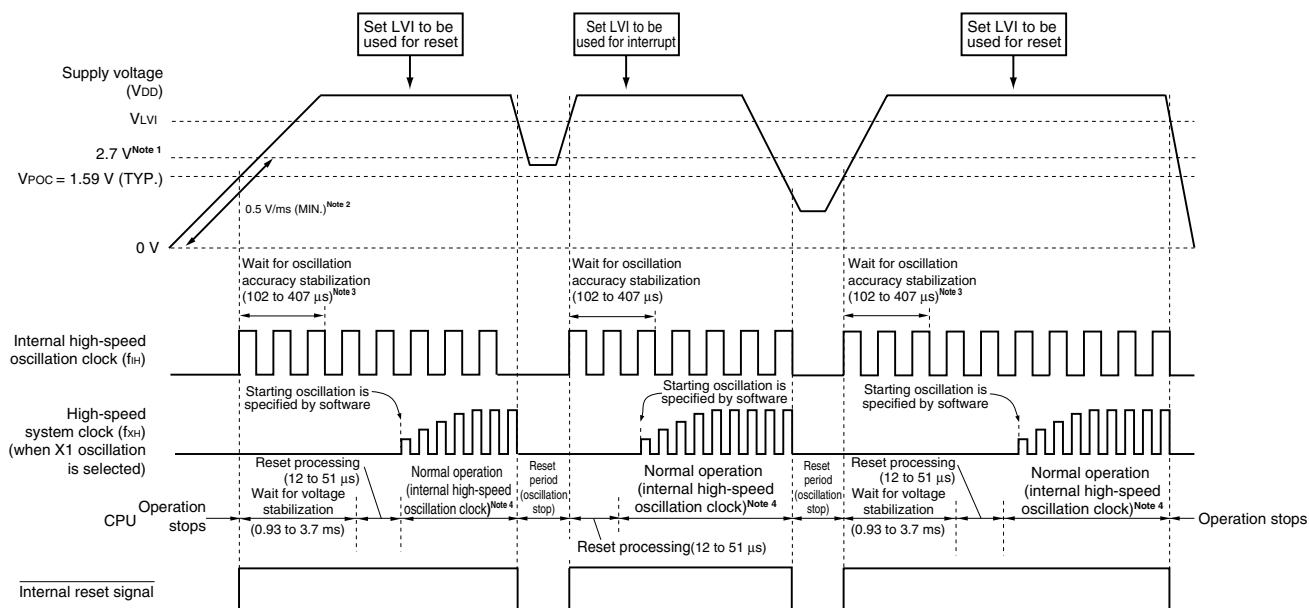
注意事项 若通过选项字节设置低电压检测(LVI)电路，则电源电压(V_{DD})超过 $2.7\text{ V} \pm 0.1\text{ V}$ 之前不能解除复位信号。

- 供电电压(V_{DD})和检测电压($V_{POC} = 1.59\text{ V} \pm 0.15\text{ V}$)相匹配时，并且 $V_{DD} < V_{PDR}$ 时，产生内部复位信号。

通过上电清除电路和低电压检测电路产生的内部复位信号时序如下所示。

图 17-2. 由上电清除电路和低电压检测电路产生的内部复位信号时序 (1/2)

(1) 1.59 V POC 模式(选项字节:LVISTART = 0)



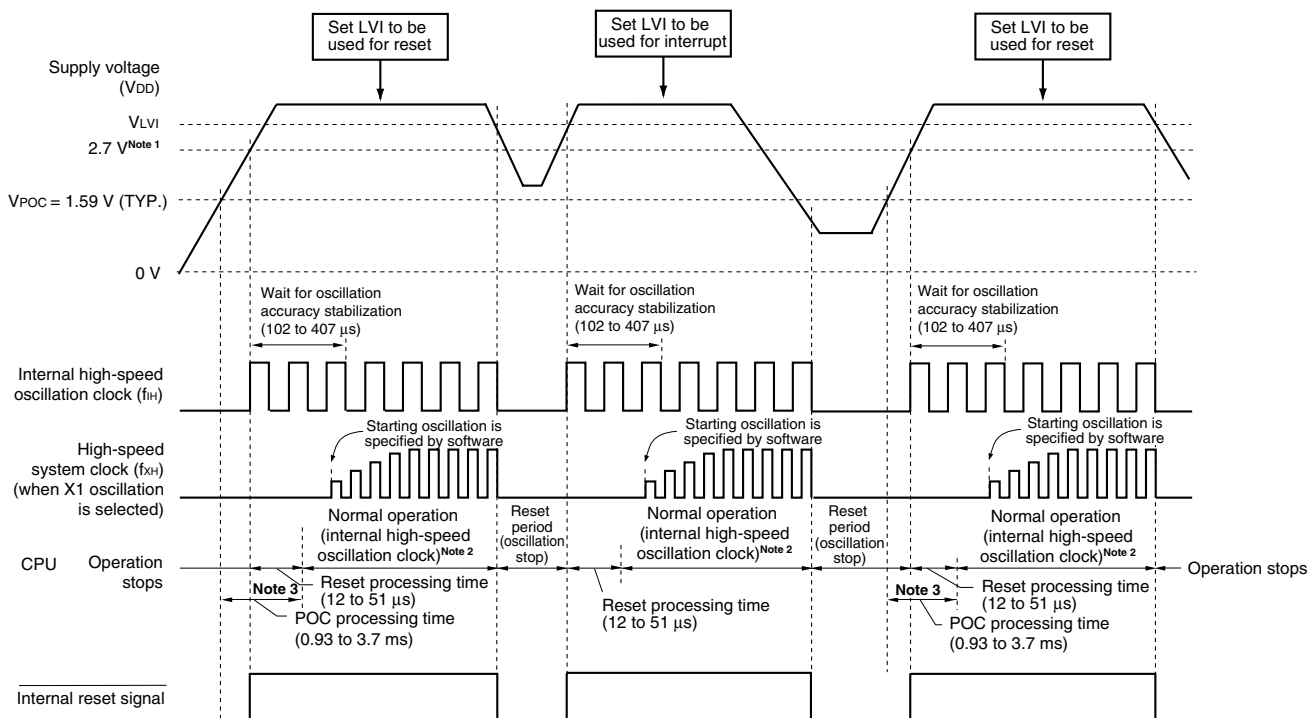
- 注 1. 操作保证范围在 $2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}$ 之间。电压下降时, 若要保持低于 2.7 V 的复位状态, 则使用低电压检测电路的复位功能, 或将低电平输入到 $\overline{\text{RESET}}$ 引脚。
2. 若上电后电压上升到 2.7 V 的速率小于 0.5 V/ms (最小值), 则上电后, 电压达到 2.7 V 之前将低电平输入到 $\overline{\text{RESET}}$ 引脚。
3. 内部电压稳定等待时间包括内部高速振荡时钟振荡的精确稳定时间。
4. 可选择内部高速振荡时钟或高速系统时钟用作 CPU 时钟。若使用 X1 时钟, 则通过 OSTC 寄存器确认是否已经过振荡稳定时间。

注意事项 复位状态解除后,通过软件设置低电压检测电路(参阅 第十八章 低电压检测电路)。

备注 VLVI: LVI 检测电压
VPOC: 检测电压

图 17-2. 通过上电清除电路和低电压检测电路产生的内部复位信号时序 (2/2)

(2) 2.7 V/1.59 V POC 模式下(选项字节:LVISTART = 1)



- 注 1. 操作保证范围为 $2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}$ 。电压下降时, 若要保持低于 2.7V 的复位状态, 则使用低电压检测电路的复位功能, 或将低电平输入到 $\overline{\text{RESET}}$ 引脚。
2. 可选择内部高速振荡时钟或高速系统时钟用作 CPU 时钟。若使用 X1 时钟, 则通过 OSTC 寄存器确认是否已经过振荡稳定时间。
3. 从检测电压(1.59 V(典型值))到开始正常操作所需时间如下所示。
- 1.59 V (典型值) → 2.7 V (典型值) 到达时间 < 3.7 ms:
1.59 V (典型值) → 正常操作需要大约 1.0~3.8 ms 的 POC 处理时间。
 - 1.59 V (典型值) → 2.7 V (典型值) 到达时间 > 3.7 ms:
2.7 V (典型值) → 正常操作需要大约 12~51 μs 的 POC 处理时间。

注意事项 复位状态解除后,通过软件设置低电压检测电路(参阅 第十八章 低电压检测电路)。

备注 V_{LVI}: LVI 检测电压
V_{POC}: 检测电压

17.4 上电清除电路注意事项

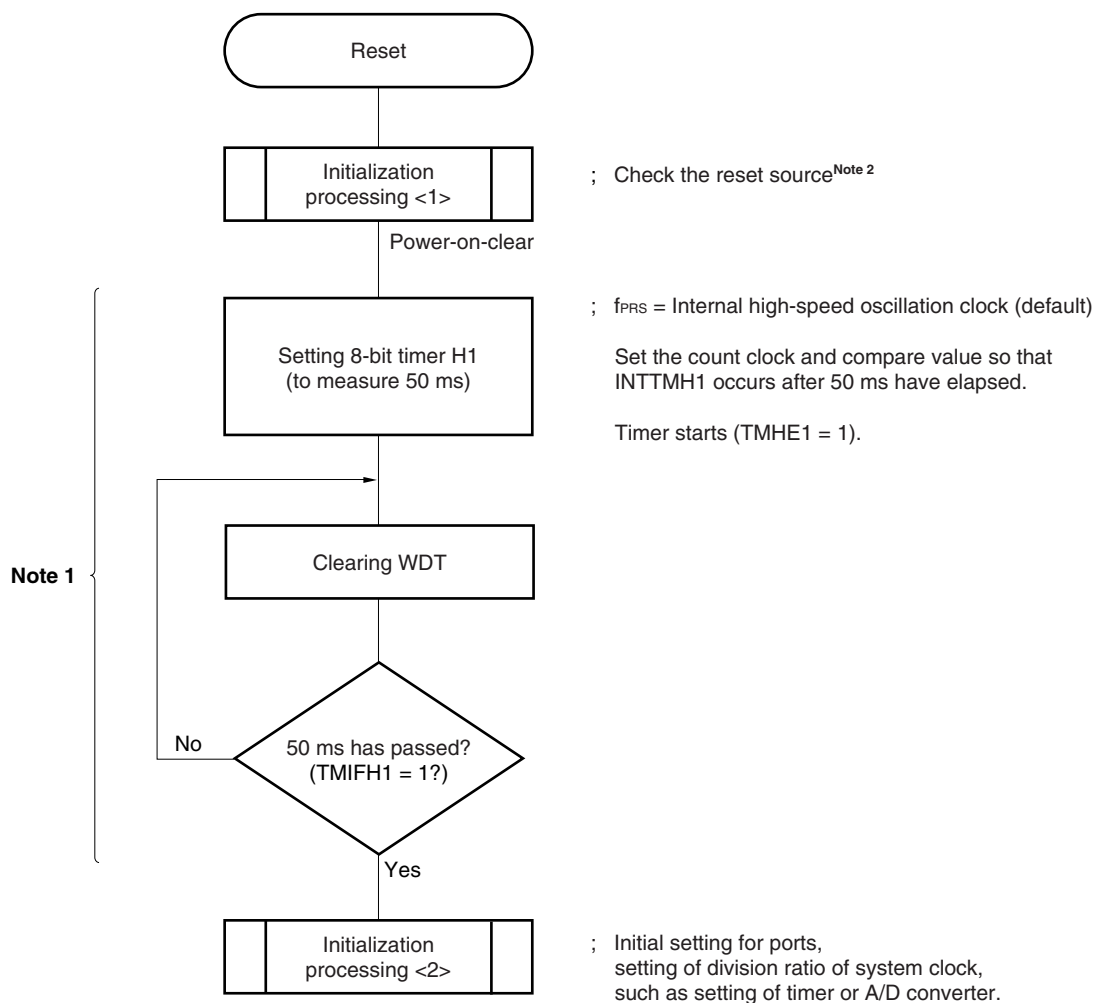
在系统中，若电源电压(V_{DD})接近检测电压(V_{Poc})时，产生一定时间的波动，则系统可能会反复复位和解除复位。在这种情况下，通过以下方法可以设置从复位解除到启动微控制器操作的所需时间。

<方法>

复位信号解除后，通过使用定时器等软件计数器等待系统电源电压的波动期，然后对端口进行初始化。

图 17-3. 复位解除的软件处理示例 (1/2)

- 接近检测电压的电源电压波动为 50ms 以下时

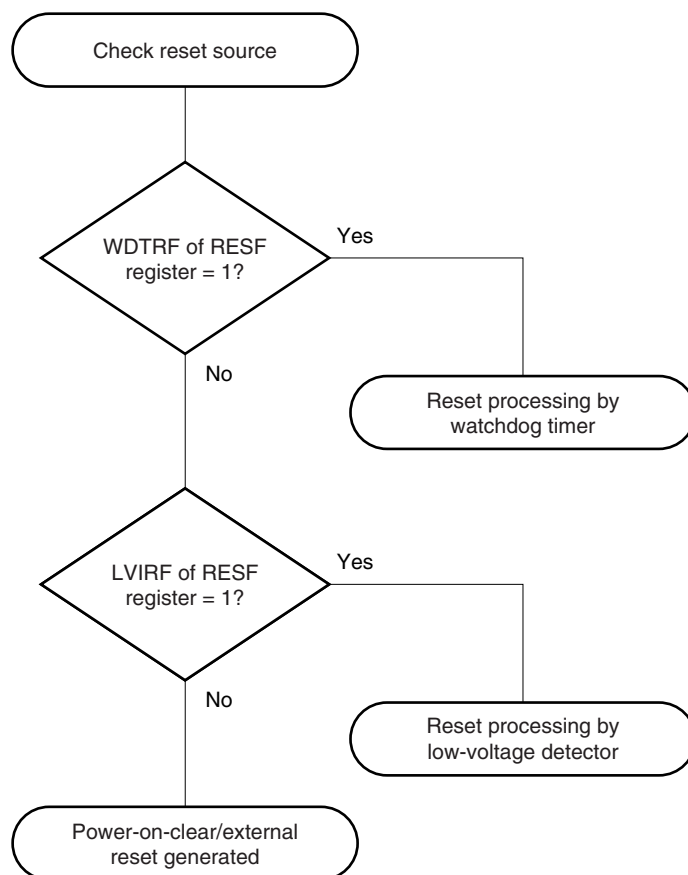


备注 1. 若在此期间再次产生复位，则不能执行初始化处理<2>。

2. 流程图如下一页所示。

图 17-3. 复位解除的软件处理示例 (2/2)

- 检测复位源



第十八章 低电压检测电路

18.1 低电压检测电路功能

低电压检测电路具有以下功能。

- 电源电压(V_{DD})与 LVI 检测电压(V_{LVI})相比较, 产生内部复位或内部中断信号。
- 通过选项字节将低电压检测电路(LVI)设置为 ON 的默认值。置“ON”, 若在检测电压 $V_{POC} = 1.59\text{ V}$ (典型值)时上电, 则当电源电压 (V_{DD}) < LVI 检测电压($V_{LVI} = 2.7\text{ V} \pm 0.1\text{ V}$)时, 将产生内部复位信号。此后, 当电源电压 (V_{DD}) < LVI 检测电压($V_{LVI} = 2.7\text{ V} \pm 0.1\text{ V}$)时, 将产生复位信号。
- 检测后, 可通过软件选择产生复位或中断信号。
- 可通过软件的检测等级分 11 个级别选择电源电压的检测电平(V_{LVI})。
- STOP 模式操作。

可通过软件选择产生以下复位和中断信号。

选择复位(LVIMD = 1)	选择中断(LVIMD = 0)
$V_{DD} < V_{LVI}$ 时, 产生内部复位信号; $V_{DD} \geq V_{LVI}$ 时, 解除复位信号。	V_{DD} 降低至低于 V_{LVI} ($V_{DD} < V_{LVI}$) 或升高至高于 ($V_{DD} \geq V_{LVI}$) 时, 产生内部中断信号。

备注 LVISEL: 低电压检测寄存器(LVIM)第 2 位
 LVIMD: LVIM 第 1 位

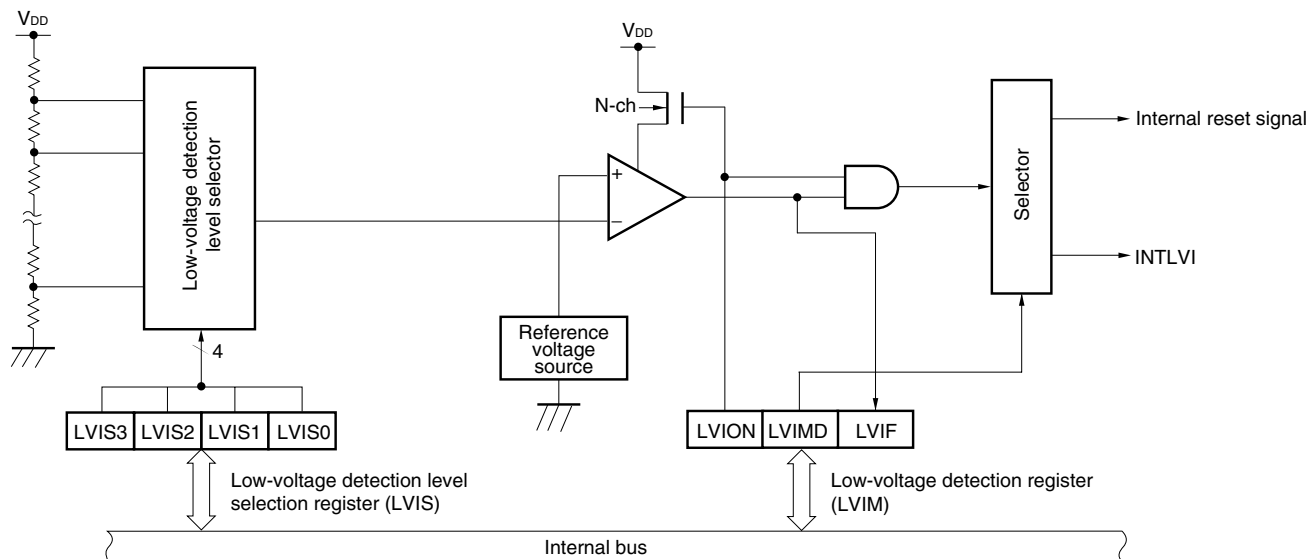
低电压检测电路处于操作时, 通过读取低电压检测标志(LVIF: LVIM 的第 0 位), 可以检测出电源电压或外部输入引脚的输入电压是否大于或小于检测等级。

低电压检测电路产生复位时, 复位控制标志寄存器(RESF)的第 0 位(LVIRF)置“1”。有关 RESF 的详情, 参阅第十六章复位功能。

18.2 低电压检测器配置

低电压检测器系统图如图 18-1 所示。

图 18-1. 低电压检测器的框图



18.3 寄存器控制低电压检测器

由下列寄存器控制低电压检测器。

- 低电压检测寄存器(LVIM)
- 低电压检测电平选择寄存器(LVIS)

(1) 低电压检测寄存器(LVIM)

该寄存器设置低电压检测和操作模式。

可用 1 位或 8 位存储器操作指令设置该寄存器。

产生 LVI 复位以外的复位信号时，清零。

图 18-2. 低电压检测寄存器(LVIM)的格式

地址: FFBEH 复位后: 00H^{注1} R/W^{注2}

符号	<7>	6	5	4	3	2	<1>	<0>
LVIM	LVION	0	0	0	0	0	LVIMD	LVIF

LVION ^{注3,4}	允许低电压检测操作
0	禁止操作
1	允许操作

LVIMD ^{注3}	低电压检测操作模式（中断/复位）选择
0	电源电压（V _{DD} ）降低至低于 LVI 电压（V _{LVI} ）（V _{DD} < V _{LVI} ）或升高至高于（V _{DD} ≥ V _{LVI} ）时，产生内部中断信号。
1	电源电压（V _{DD} ）< LVI 检测电压（V _{LVI} ）时，产生内部复位信号；V _{DD} ≥ V _{LVI} 时，解除复位信号。

LVIF	低电压检测标志
0	电源电压（V _{DD} ）≥ LVI 检测电压（V _{LVI} ），或禁止 LVI 操作时
1	电源电压（V _{DD} ）< LVI 检测电压（V _{LVI} ）

注 1. 复位值取决于复位源和选项字节的设置。

LVI 复位(LVI 默认启动功能复位除外)时,该寄存器不清为 00H。

其他复位时，该寄存器清为 00H。

2. 第 0 位只读。

3. LVI 复位以外的复位时，LVION 和 LVIMD 清零。LVI 复位时，不清零。

4. LVION 置“1”时，LVI 电路内的比较器开始操作。从 LVION 置“1”到操作稳定，使用软件等待稳定时间（10μs(最大值)）。操作稳定后，电压从低于 LVI 检测电压到 LVIF 置“1”，需要外部输入 200μs 以上(最小脉冲宽度：200μs)。

注意事项 1. 可通过执行以下任一步骤，停止 LVI。

- 使用 8 位操作指令时：写入 00H 至 LVIM。
- 使用 1 位操作指令时：LVION 清零。

2. 中断模式（LVIMD = 0）下使用 LVI 时，若电源电压（V_{DD}）≤ 检测电压（V_{LVI}）时禁止 LVI 操作（LVION 清零），则产生中断请求信号 (INTLVI)，并且 LVIF 可能置为“1”。

(2) 低电压检测电平选择寄存器(LVIS)

该电平选择低电压检测电平。

可用 1 位或 8 位存储器操作指令设置该寄存器。

复位信号产生时，该寄存器置“00H”。

图 18-3. 低电压检测电平选择寄存器(LVIM)格式

地址	FFBFH	复位后: 00H ^注	R/W					
符号	7	6	5	4	3	2	1	0
LVIS	0	0	0	0	LVIS3	LVIS2	LVIS1	LVIS0

LVIS3	LVIS2	LVIS1	LVIS0	检测电平
0	0	0	0	$V_{LV10} (4.24 \pm 0.1 \text{ V})$
0	0	0	1	$V_{LV11} (4.09 \pm 0.1 \text{ V})$
0	0	1	0	$V_{LV12} (3.93 \pm 0.1 \text{ V})$
0	0	1	1	$V_{LV13} (3.78 \pm 0.1 \text{ V})$
0	1	0	0	$V_{LV14} (3.62 \pm 0.1 \text{ V})$
0	1	0	1	$V_{LV15} (3.47 \pm 0.1 \text{ V})$
0	1	1	0	$V_{LV16} (3.32 \pm 0.1 \text{ V})$
0	1	1	1	$V_{LV17} (3.16 \pm 0.1 \text{ V})$
1	0	0	0	$V_{LV18} (3.01 \pm 0.1 \text{ V})$
1	0	0	1	$V_{LV19} (2.85 \pm 0.1 \text{ V})$
1	0	1	0	$V_{LV110} (2.70 \pm 0.1 \text{ V})$

注 复位值变化取决于复位源。

若由 LVI 复位（通过 LVI 默认启动功能的复位除外）时，LVIS 寄存器的值不能复位，保持当前值。其他复位时，复位为“00H”。其他复位时，该寄存器清为 00H。

注意事项 1. 确保第 4 位至第 7 位清零。

2. LVI 操作期间禁止改变 LVIS 值。

18.4 低电压检测电路操作

以下两种模式可以使用低电压检测电路。

(1) 用于复位(LVIMD = 1)

- 通过电源电压(V_{DD})和 LVI 检测电压(V_{LVI})相比较, $V_{DD} < V_{LVI}$ 时产生内部复位信号; $V_{DD} \geq V_{LVI}$ 时则解除内部复位。

备注 通过选项字节将低电压检测电路(LVI)设置为 ON 的默认值。置“ON”, 若在检测电压 $V_{POC} = 1.59\text{ V}$ (典型值) 时上电, 则电源电压 (V_{DD}) < 检测电压($V_{LVI} = 2.70\text{ V} \pm 0.1\text{ V}$)时, 产生内部复位信号。

(2) 用于中断 (LVIMD = 0)

- 电源电压(V_{DD})和 LVI 检测电压(V_{LVI})的比较 V_{DD} 降低至低于 V_{LVI} ($V_{DD} < V_{LVI}$) 或升高至高于 ($V_{DD} \geq V_{LVI}$) 时, 产生中断信号 (INTLVI)。

低电压检测电路处于操作时, 通过读取低电压检测标志(LVIF: LVIM 的第 0 位), 可以检测出电源电压的输入电压是否大于或小于检测等级。

备注 LVIMD: 低电压检测寄存器(LVIM)第 1 位。

18.4.1 用于复位

(1) 1.59 V POC 模式下(LVISTART = 0)

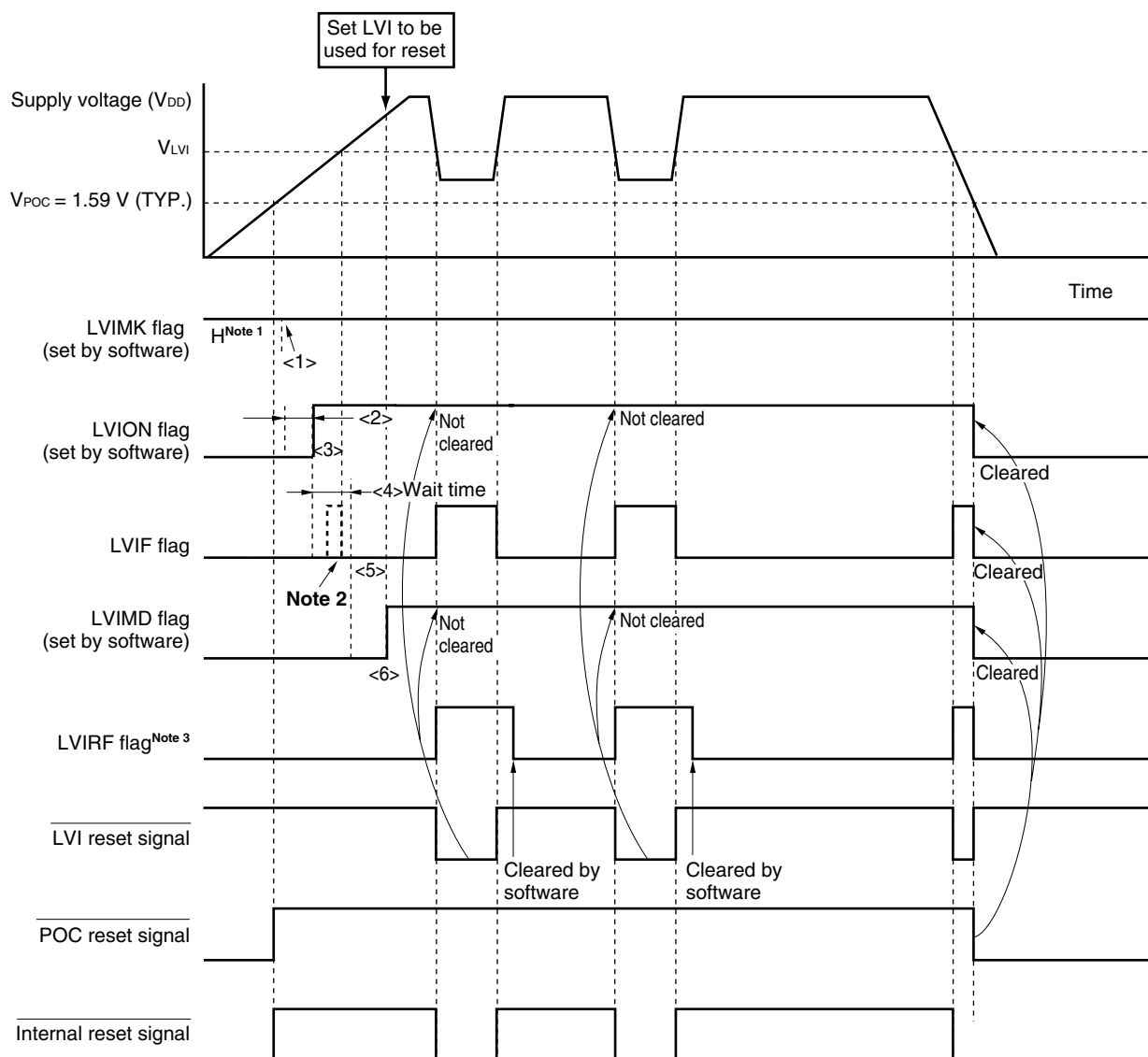
- 开始操作时
 - <1> 屏蔽 LVI 中断(LVIMK = 1)。
 - <2> 使用低电压检测电平选择寄存器(LVIS)的第 3 位至第 0 位(LVIS3 至 LVIS0)设置 LVI 检测电压。
 - <3> LVIM 的第 7 位(LVION)置“1”(允许 LVI 操作)。
 - <4> 使用软件等待操作稳定时间(10 μs(最大值))。
 - <5> 等待由 LVIM 的第 0 位确认出“电源电压(V_{DD}) ≥ LVI 检测电压 (V_{LVI})”。
 - <6> LVIM 的第 1 位(LVIMD)置“1”(检测出电平时产生复位)。

图 18-4 给出低电压检测电路内部复位信号产生的时序。该时序图中的标号与上面的<1>至<6>相对应。

注意事项 1. 必须执行<1>。LVIMK = 0 时，在程序 1<3>之后可能立即会产生中断。
2. LVIMD 至“1”时，若电源电压(V_{DD}) ≥ LVI 检测电压(V_{LVI})，则不产生复位信号。

- 停止操作时
必须执行以下任一步骤。
- 使用 8 位操作指令时：
写入 00H 至 LVIM。
- 使用 1 位操作指令时：
LVIMD 清零后，LVION 也清零。

图 18-4. 低电压检测电路内部复位信号产生的时序(LVISTART = 0)



注 1. 通过产生复位信号，使 LVIMK 标志置“1”。

2. 中断请求标志寄存器的 LVIF 标志和 LVIF 标志可能会置“1”。

3. LVIRF 是复位控制标志寄存器(RESF)的第 0 位。有关 RESF 的详情，参阅第十六章 复位功能。

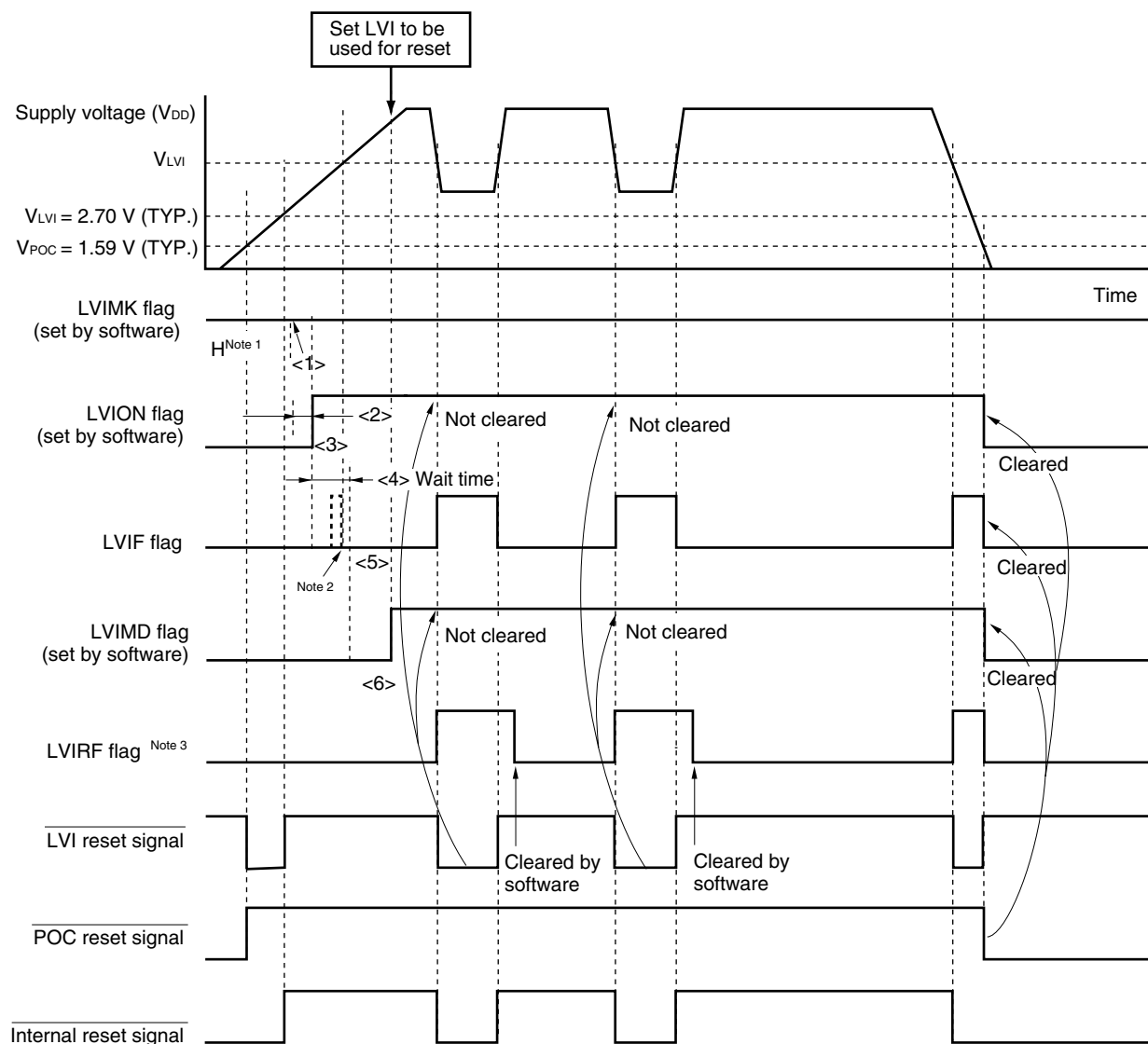
备注 1. 图 18-4 中的<1>至<6>，与 18.4.1 (1) 设置 LVI 默认启动功能停止 (LVISTART = 0) 中“启动操作时”的<1>至<6>描述相对应。

2. V_{POC}: 检测电压

(2) 2.7 V/1.59 V POC 模式下(LVISTART = 1)

启动操作和停止操作时的设置相同于 18.4.1(1)设置 LVI 默认启动功能停止时的描述。

图 18-5.低电压检测电路内部复位信号产生的时序(LVISTART = 1)



- 注 1. 通过产生复位信号，使 LVIMK 标志置“1”。
2. 中断请求标志寄存器的 LVIF 标志和 LVIF 标志可能会置“1”。
3. LVIRF 是复位控制标志寄存器(RESF)的第 0 位。
有关 RESF 的详情，参阅第十六章 复位功能。

- 备注 1. 图 18-5 中的<1>至<6>，与 18.4.1 (1) 设置 LVI 默认启动功能停止 (LVISTART = 0) 中“启动操作时”的<1>至<6>描述相对应。
2. V_{POC}: 检测电压

18.4.2 用于中断

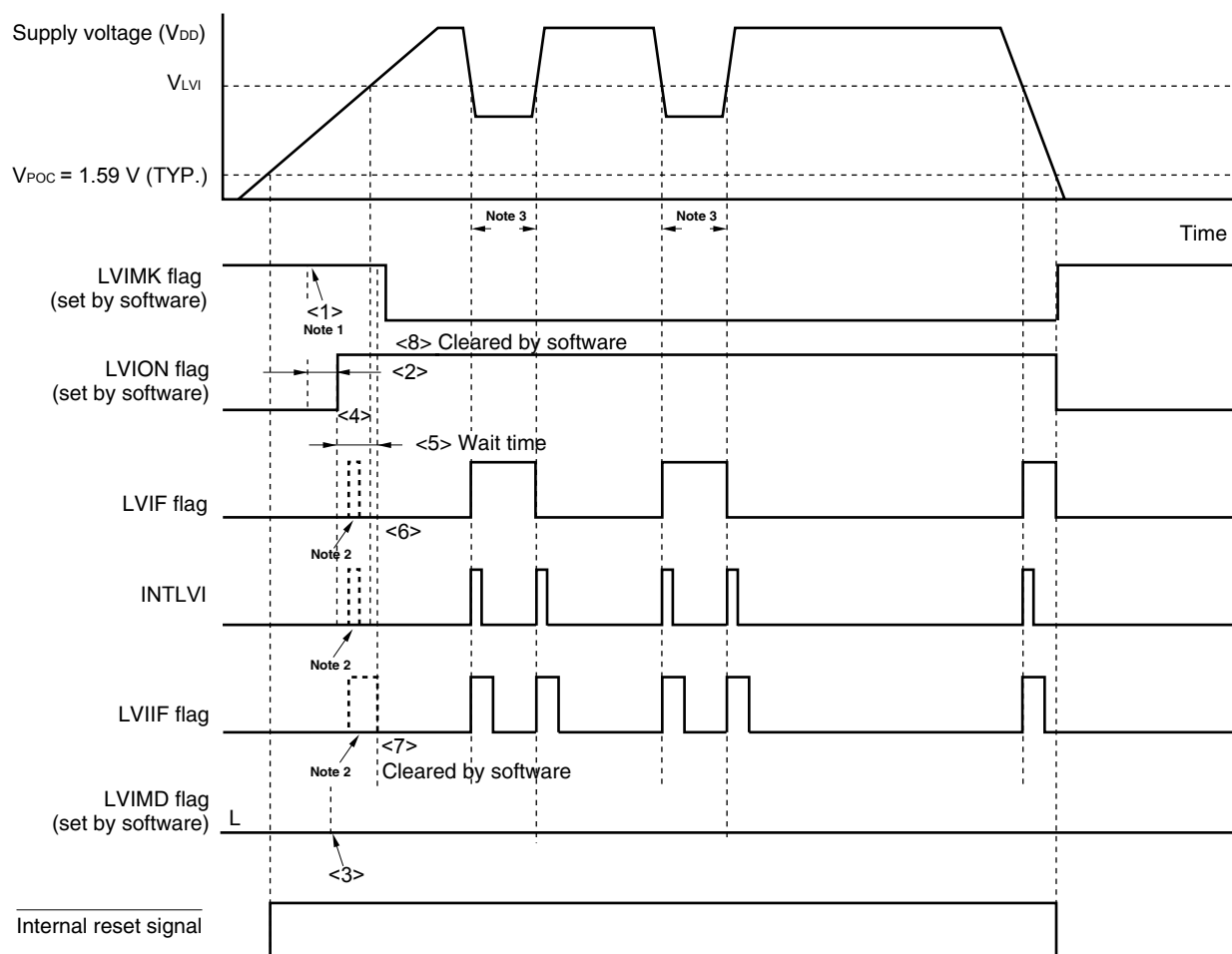
(1) 设置 LVI 默认功能操作停止时(LVISTART = 0)

- 开始操作时
 - <1> 屏蔽 LVI 中断(LVIMK = 1).
 - <2> 使用低电压检测电平选择寄存器(LVIS)的第 3 位至第 0 位(LVIS3 至 LVIS0)设置 LVI 检测电压。
 - <3> LVIM 的第 1 位(LVIMD)清零(检测出电平时产生复位) (默认值)。
 - <4> LVIM 的第 7 位(LVION)置“1”(允许 LVI 操作)。
 - <5> 使用软件等待操作稳定时间(10 μs(最大值))。
 - <6> 在 LVIM 的第 0 位(LVIF)检测 V_{DD} 的下降沿时,确认 “电源电压(V_{DD}) ≥ LVI 检测电压(V_{LVI})” ; 检测 V_{DD} 的上升沿时, 确认 “电源电压(V_{DD}) < LVI 检测电压(V_{LVI})” 。
 - <7> LVI(LVIF)的中断请求标志清零。
 - <8> LVI(LVIF)的中断请求标志清零。
 - <9> 执行 EI 指令(使用向量中断时)。

图 18-6 给出低电压检测电路中中断信号产生的时序。该时序图中的标号与上面的<1>至<8>相对应。

- 停止操作时:
 - 必须执行以下任一步骤。
- 使用 8 位操作指令时:
 - 写入 00H 至 LVIM。
- 使用 1 位操作指令时:
 - LVION 清零。

图 18-6. 低电压检测电路中断信号产生的时序(LVISTART = 0)



注 1. 通过产生复位信号，使 LVIMK 标志置“1”。

2. 产生中断请求信号(INTLVI)时, LVIF 和 LVIIF 可能会置“1”。

3. 若电源电压 (V_{DD}) $\leq$ 检测电压 (V_{LVI}) 时禁止 LVI 操作 (LVION 清零)，则产生中断请求信号 (INTLVI)，并且 LVIIF 可能置“1”。

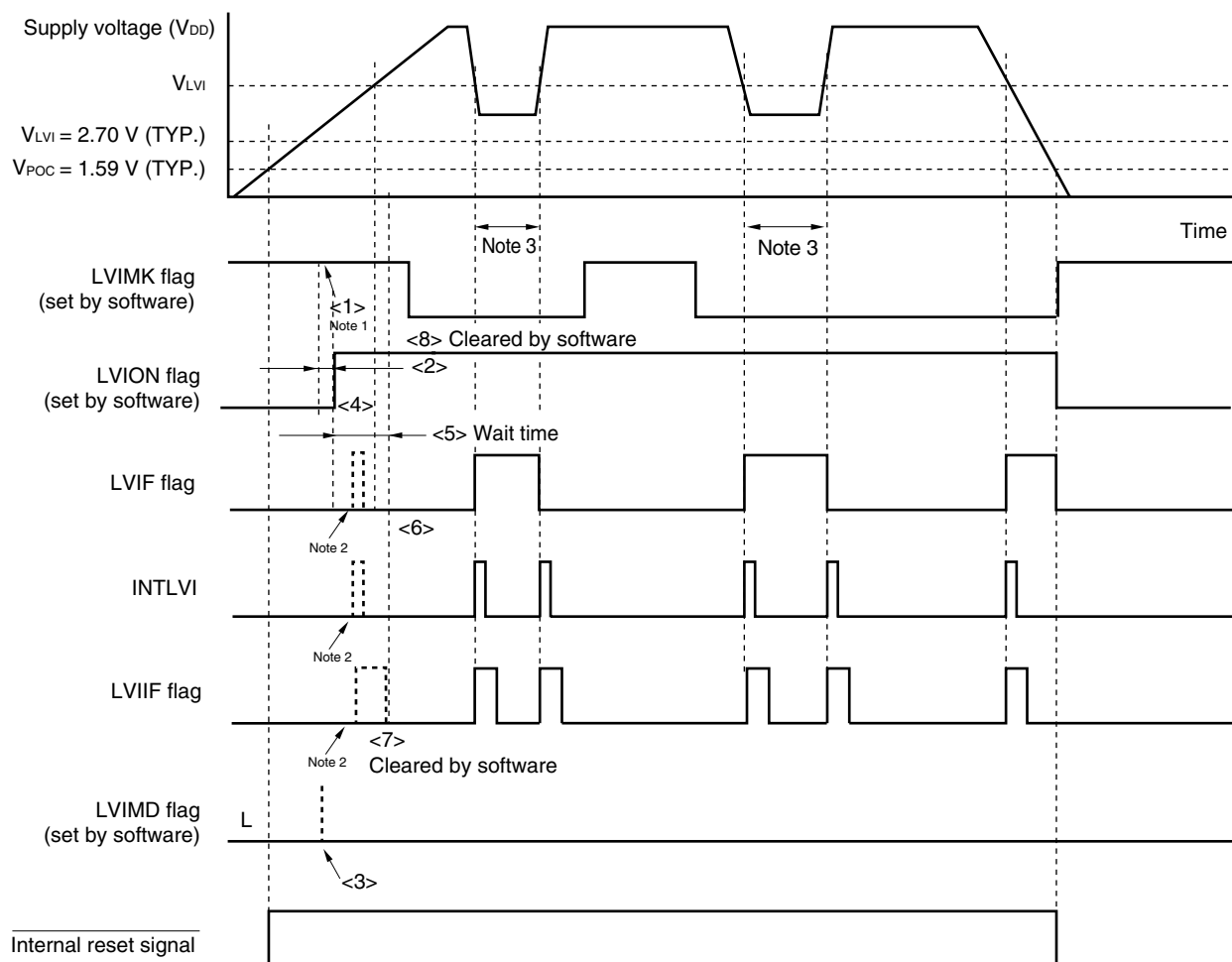
备注 1. 图 18-6 中的<1>至<8>，与 18.4.1 (1) 设置 LVI 默认启动功能停止 (LVISTART = 0) 中“启动操作时”的<2>至<8>描述相对应。

2. V_{POC} : 检测电压

(2) 设置 LVI 默认功能操作时(LVISTART = 1)

启动操作和停止操作时的设置相同于 18.4.1(2)设置 LVI 默认启动功能停止时的描述。

图 18-7. 低电压检测电路中断信号产生的时序(LVISTART = 1)



注 1. 通过产生复位信号，使 LVIMK 标志置“1”。

2. 中断请求标志寄存器的 LVIF 标志和 LVIF 标志可能会置“1”。

3. 若电源电压 (V_{DD}) ≤ 检测电压 (V_{LVI}) 时禁止 LVI 操作 (LVION 清零)，则产生中断请求信号 (INTLVI)，并且 LVIF 可能被置“1”。

备注 1. 图 18-7 中的<1>至<8>，与 18.4.1 (1) 设置 LVI 默认启动功能 (LVISTART = 1) 中“启动操作时”的<2>至<8>描述相对应。

2. V_{POC}:检测电压

18.5 低电压检测器注意事项

电源电压(V_{DD})接近 LVI 检测电压(V_{POC})时, 产生一定时间波动的系统, 根据低电压检测电路的使用方法, 进行如下操作。

操作示例 1: 用于复位时

系统可能会重复进行复位/复位解除。

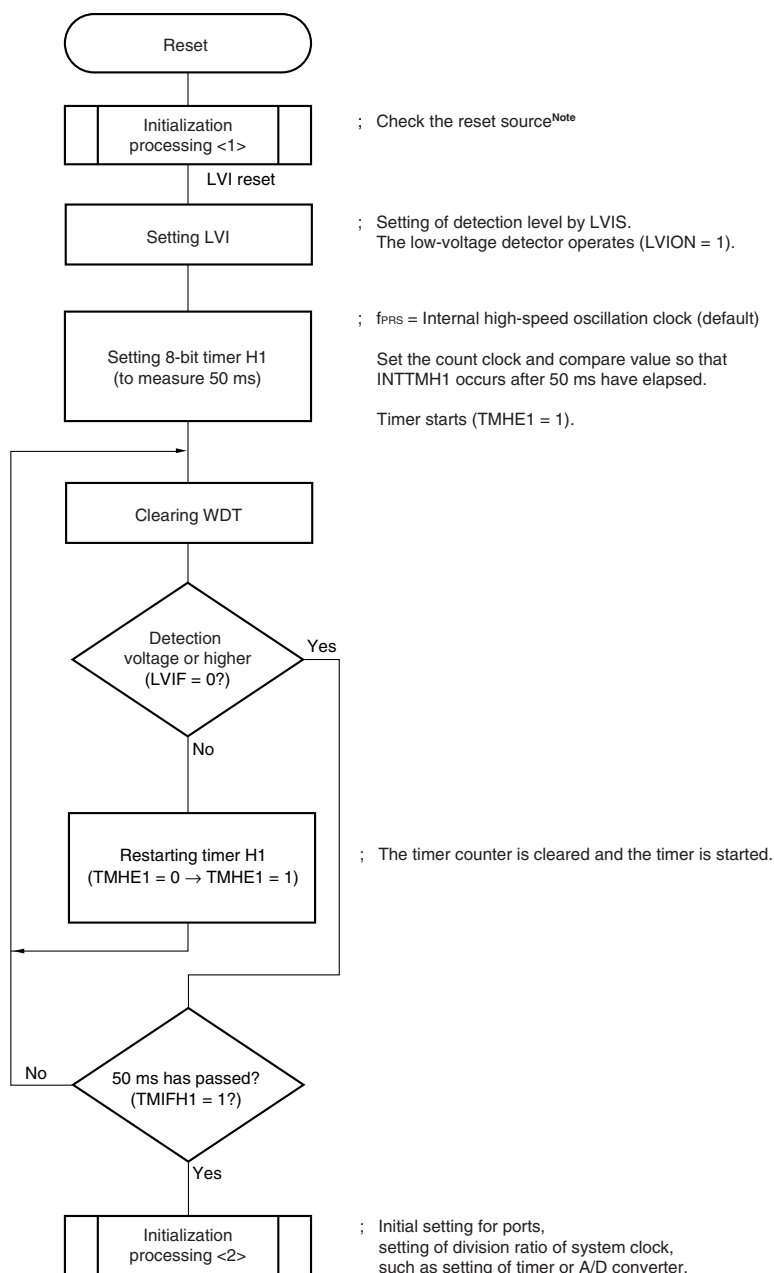
通过以下方法, 可任意设置复位解除到操作开始的时间。

<方法>

复位信号解除后, 通过使用定时器等软件计数器等待系统电源电压的波动期, 然后对端口进行初始化 (参阅图 18-8)。

图 18-8. 复位解除的软件处理示例 (1/2)

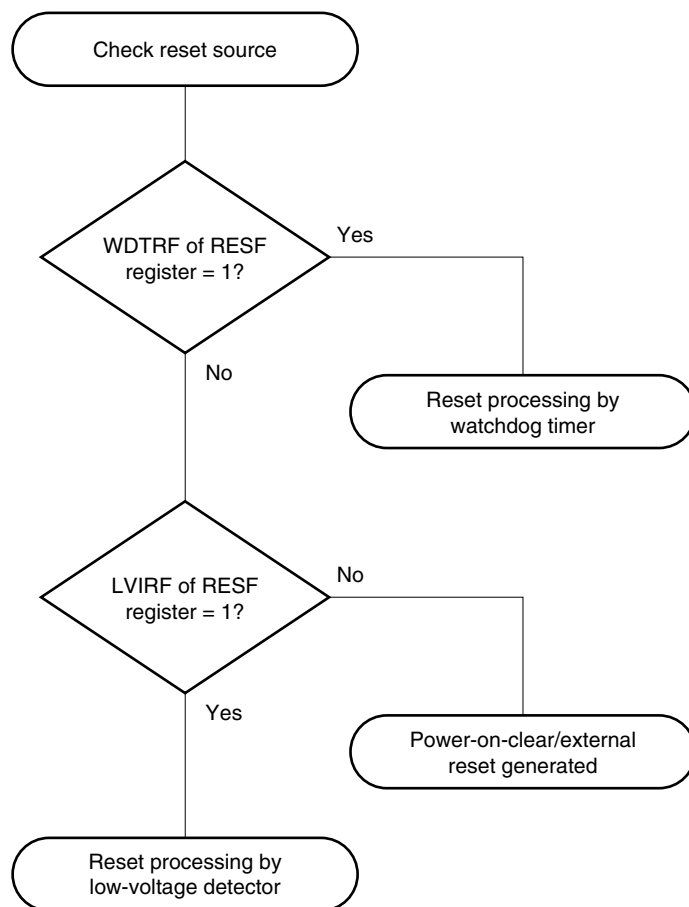
- 接近 LVI 检测电压的电源电压波动为 50ms 以下时



注 流程图如下一页所示。

图 18-8. 复位解除的软件处理示例 (2/2)

• 检测复位源



操作示例 2：用于中断时

可能会频繁产生中断请求。
采取以下方法。

<方法>

在 LVI 中断服务程序中通过使用低电压检测寄存器(LVIM) 的第 0 位(LVIF)，检测 V_{DD} 的下降沿时，确定“电源电压(V_{DD}) $\geq$ LVI 检测电压(V_{LVI})”；或检测 V_{DD} 的上升沿时，确定“电源电压(V_{DD}) $<$ LVI 检测电压(V_{LVI})”。中断请求标志寄存器 0L(IF0L) 的第 1 位(LVIIF)清零。

若系统在接近 LVI 检测电压时的电源电压波动时间长，则等待电源电压变化期间后采取上述方法。

第十九章 稳压器

19.1 稳压器概述

μPD79F7023, 79F7024 微控制器内置用于器件内部恒定电压操作的电路。此时，为了稳定稳压器的输出电压，通过电容器将 REGC 引脚连接到 V_{SS}(0.47 至 1 μF: 目标)。但在运行高速内置谐振时钟、外部主系统时钟中使用 STOP 模式时，推荐使用 0.47 μF。另外，为了使用稳定的内部电压，建议使用性能良好的电容器。

稳压器输出电压一般为 2.5 V(典型值)，低功耗模式下为 1.9V (典型值)。

19.2 寄存器控制稳压器

(1) 稳压器模式控制寄存器(RMC)

该寄存器设置内部振荡器的操作模式。该寄存器设置稳压器的输出电压。

可按 8 位存储器操作指令设置 RMC。

复位信号产生时，该寄存器清零(00H)。

图 19-1. 稳压器模式控制寄存器(RMC)的格式

地址: FF3DH 复位后:00H R/W

符号	7	6	5	4	3	2	1	0
RMC								

RMC[7:0]	控制稳压器输出电压
59H	低功耗模式(恒为 1.9 V) CPU 时钟($f_{CPU} \leq 4 \text{ MHz}$) 时，可设置。
56H	低功耗模式(恒为 1.9 V) CPU 时钟选择高速内置振荡时钟时，仅在 $f_{CPU} \leq 1 \text{ MHz}$ 或高速系统时钟停止的情况下，可设置 56H。
00H	正常电源模式 (固定在 2.5V)
其它	禁止设置

- 注意事项
1. 将 RMC 寄存器的设定值从 56H/59H 更改为 00H，使用 4MHz 以上的 CPU 操作频率时，设置 RMC 寄存器后，应经过 22μs 以上之后，再设置 PCC 和 RCM。
 2. 使用低功耗模式的固定设置时，可在以下情况下使用 RMC 寄存器。
<CPU 时钟选择高速内置振荡时钟，高速系统时钟停止时> $f_{CPU} \leq 1 \text{ MHz}$

19.3 自编程注意事项

1. 执行子编程时，应固定稳压器的输出电压模式。
2. 在正常功率模式下，重写闪存时，允许重写的电源电压范围在 $V_{DD} \geq 2.5\text{ V}$ 。另外，在正常功率模式下，可通过自编程库重写程序区域。
3. 在低功耗模式下，禁止重写闪存。

备注 有关详细的自编程功能和自编程库，参阅“**78K0 微控制器 用户手册 自编程库 01 型 (U18274J)**”和“**78K0 微控制器 自编程库 01 型 3.10 版 操作注意事项(文件) (ZUD-CD-09-0122)**”。

第二十章 选项字节

20.1 选项字节功能

在μPD79F7023, 79F7024 微控制机中, 闪存的 0080H、0081H 和 0084H 为选项字节区域。打开电源或从复位状态重启器件时, 器件自动参考选项字节, 并设置指定功能。使用本产品时, 必须使用选项字节设置下列功能。

在自编程期间使用引导交换功能时, 将 0080H、0081H 和 0084H 切换到 1080H、1081H 和 1084H。因此, 应事先将 0080H、0081H 和 0084H 至 1080H、1081H 和 1084H 设置为相同值。

(1) 0080H/1080H

- 内部低速振荡器操作
 - 可用软件停止
 - 禁止停止
- 看门狗定时器间隔时间设置
- 看门狗定时器计数器操作
 - 允许计数器操作
 - 禁止计数器操作
- 看门狗定时器窗口打开期间设置

注意事项 在引导交换操作中, 0080H 和 1080H 进行切换, 因此, 应预先将 0080H 与 1080H 设置相同的值。

(2) 0081H/1081H

- LVI 默认开始的操作控制
 - LVI 默认功能操作期间(LVISTART = 1)
器件从复位解除或上电开始直至供电电压达到 2.7V (典型值) 期间处于复位状态, 电压超过 2.7V (典型值) 时, 将解除复位状态。
从复位解除或上电后开始直至供电电压达到 2.7V 期间, 速率低于 0.5 V/ms (最小值), 建议使用 LVI 默认启动操作。
 - LVI 默认功能停止期间(LVISTART = 0)
器件从复位解除或上电开始直至供电电压达到 1.59 V (典型值) 期间处于复位状态, 电压超过 1.59 V (典型值) 时, 将解除复位状态。

注意事项 LVISTART 只能通过专用闪存编程器写入。不能通过自编程或自编程中的引导交换操作进行设置和改变。
但在引导交换中, 0081H 和 1081H 进行切换, 因此, 应预先将 0080H 与 1080H 设置相同的值。

(3) 0082H/1082H

必须设置为 01H。

注意事项 在引导交换操作中，0082H 和 1082H 进行切换，因此，应预先将 0082H 与 1082H 设置相同的值。

(4) 0083H/1083H

- On-chip 调试模式
 - 禁止设置 On-chip 调试模式
 - 强制性设置 On-chip 调试模式
- 执行 STOP 指令时，时钟供应给 UART0。
 - 提供
 - 停止

注意事项 在引导交换操作中，0083H 和 1083H 进行切换，因此，应预先将 0083H 与 1083H 设置相同的值。

(5) 0084H/1084H

- On-chip 调试操作控制
 - 禁止 On-chip 调试操作
 - 允许 On-chip 调试操作，On-chip 调试安全 ID 认证失败时，清除闪存数据。
 - 允许 On-chip 调试操作，On-chip 调试安全 ID 认证失败时，不清除闪存数据。

注意事项 在引导交换操作中，0084H 和 1084H 进行切换，因此，应预先将 0084H 与 1084H 设置相同的值。

20.2 选项字节格式

选项字节格式如下所示：

图 20-1. 选项字节格式 (1/3)

地址: 0080H/1080H[※]

7	6	5	4	3	2	1	0
0	WINDOW1	WINDOW0	WDTON	WDCS2	WDCS1	WDCS0	LSROSC

WINDOW1	WINDOW0	看门狗定时器窗口打开期间
0	0	25%
0	1	50%
1	0	75%
1	1	100%

WDTON	看门狗定时器计数器/非法访问检测的操作控制
0	禁止计数器操作(复位后停止计数)，禁止非法访问检测操作
1	允许计数器操作(复位后开始计数)，允许非法访问检测操作

WDCS2	WDCS1	WDCS0	看门狗定时器溢出时间
0	0	0	$2^{10}/f_{IL}$ (4.27 ms)
0	0	1	$2^{11}/f_{IL}$ (8.53 ms)
0	1	0	$2^{12}/f_{IL}$ (17.07 ms)
0	1	1	$2^{13}/f_{IL}$ (34.13 ms)
1	0	0	$2^{14}/f_{IL}$ (68.27 ms)
1	0	1	$2^{15}/f_{IL}$ (136.53 ms)
1	1	0	$2^{16}/f_{IL}$ (273.07 ms)
1	1	1	$2^{17}/f_{IL}$ (546.13 ms)

LSROSC	内部低速振荡器操作
0	可由软件停止(RCM 寄存器的第 1 位(LSRSTOP)写入“1”时，停止操作)
1	禁止停止(即使 LSRSTOP 位写入“1”，也不停止操作)

注 在引导交换操作中, 0080H 和 1080H 进行切换, 因此, 应预先将 0080H 与 1080H 设置相同的值。

- 注意事项
- 禁止 $WDCS2 = WDCS1 = WDCS0 = 0$ 和 $WINDOW1 = WINDOW0 = 0$ 的组合设置。
 - 闪存自编程期间, 看门狗定时器继续操作。处理期间, 延迟中断应答时间。设置溢出时间和窗口容量时, 应考虑延迟。
 - 若 $LSROSC = 0$ (可用软件停止振荡), 则无论内部振荡模式寄存器(RCM)的第 0 位(LSRSTOP)设置为何值, 在 HALT 和 STOP 模式下, 不提供计数时钟给看门狗定时器。
若 8 位定时器 H1 用内部低速振荡时钟, 则即使在 HALT/STOP 模式下, 也提供计数时钟给 8 位定时器 H1。
 - 确保第 7 位清零。

- 备注
- f_{IL} : 内部低速振荡时钟频率
 - (): $f_{IL} = 240\text{ kHz}$ (典型值)

图 20-1. 选项字节格式 (2/3)

地址: 0081H/1081H^{※1,2}

7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	LVISTART

LVISTART	LVI 默认开始操作控制
0	上电时, LVI 默认为 OFF(LVI 默认启动功能停止)
1	上电时, LVI 默认为 ON(LVI 默认启动功能启动)

- 注 1. LVISTART 只能通过专用闪存编程器写入。不能通过自编程或自编程中的引导交换操作进行设置。但在引导交换中, 0081H 和 1081 进行切换, 因此, 应预先将 0080H 与 1080H 设置相同的值。
2. 改变 LVI 默认启动的设置内容时, 批擦除 (片擦除) 闪存后, 重新设置 0081H 的值。存储器指定块擦除后不能更改设置。

确保第 7 位至第 1 位清零。

地址: 0082H/1082H[※]

7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	1

注 在引导交换操作中, 0082H 和 1082H 进行切换, 因此, 应预置 0083H 为 “1”。

注意 确保置 “01H”。

地址: 0083H/1083H[※]

7	6	5	4	3	2	1	0
0	0	0	OCDCKSTP	1	1	1	OCDONB

OCDCKSTP	On-chip 调试模式下, 执行 STOP 指令时, 向 UART0 提供时钟。
0	即使执行 STOP 命令, 也不能停止内部振荡器, 继续向 UART0 提供时钟。CPU 和外围电路停止。
1	停止内部振荡, 从而停止向 UART0 提供时钟。

OCDONB	On-chip 调试模式
0	禁止设置 On-chip 调试模式
1	强制性设置 On-chip 调试模式

注 在引导交换操作中, 0083H 和 1083H 进行切换, 因此, 应预先将 0083H 与 1083H 设置相同的值。

注意事项 确保第 7 至第 5 位和第 0 位清零, 第 3 至第 1 位置 “1”。

图 20-1.选项字节格式 (3/3)

地址: 0084H/1084H[※]

7	6	5	4	3	2	1	0
0	0	0	0	0	0	OCDEN1	OCDEN0

OCDEN1	OCDEN0	On-chip 调试操作控制
0	0	禁止操作
0	1	禁止设置
1	0	允许操作。On-chip 调试安全 ID 认证失败时，禁止清除闪存数据。
1	1	允许操作。On-chip 调试安全 ID 认证失败时，清除闪存数据。

注 在引导交换操作中，0084H 和 1084H 进行切换，因此，应预先将 0084H 与 1084H 设置相同的值。

注意事项 确保第 7 位至第 2 位清零。

备注 有关 On-chip 调试安全 ID，参阅第二十二章 On-chip 调试功能。

以下是选择子介设置的软件描述示例。

OPT	CSEG	AT 0080H	
选项:	DB	30H	: 允许看门狗定时器操作（非法访问检测操作）；
			: 看门狗定时器的窗口打开期间:50%;
			: 看门狗定时器的溢出时间: $2^{10}/f_{IL}$;
			: 禁止使用软件停止内部振荡器。
	DB	00H	: 停止 LVI 默认开始功能
	DB	01H	: 内部高速振荡时钟频率 4 MHz(典型值)
	DB	1EH	: 使用 TOOLC0/X1, TOOLD0/X2 引脚
	DB	02H	: 允许操作。On-chip 调试安全 ID 认证失败时，禁止清除闪存数据。

备注 复位处理期间，参考选项字节。有关复位处理时序的详情，参阅**第十六章 复位功能**。

第二十一章 闪存

μPD79F7023 和 79F7024 内置有闪存，安装在基板状态下可进行程序的写入、擦除和重写。

21.1 内存容量切换寄存器

通过存储器容量切换寄存器(IMS)选择内存容量。

可用 8 位存储器操作指令设置 IMS。

复位信号产生时，IMS 设置为 CFH。

注意事项 由于复位时 ROM 区域的设置不确定，因此，复位解除后，必须按照表 21-1 设置每个产品的对应值。

图 21-1. 存储器容量切换寄存器(IMS)的格式

地址：FFF0H 复位后：CFH R/W

符号	7	6	5	4	3	2	1	0
IMS	RAM2	RAM1	RAM0	0	ROM3	ROM2	ROM1	ROM0

RAM2	RAM1	RAM0	内部高速 RAM 容量选择
0	0	0	768 字节
0	1	0	512 字节
1	1	0	(默认值)
其它			禁止设置

ROM3	ROM2	ROM1	ROM0	内部 ROM 容量选择
0	0	1	0	8 KB
0	1	0	0	16 KB
1	1	1	1	(默认值)
其它				禁止设置

表 21-1. 内存容量切换寄存器(IMS)设定值

产品	IMS 设置
μPD79F7023	42H
μPD79F7024	04H

21.2 闪存寄存器编程器的写入方法

可通过专用闪存寄存器编程器进行在线写入或离线写入。

(1) 在线编程

在目标系统上安装 μPD79F7023, 79F7024 系列产品后, 可以重写闪存内容。必须将连接专用闪存编程器的连接器安装在目标系统上。

(2) 离线编程

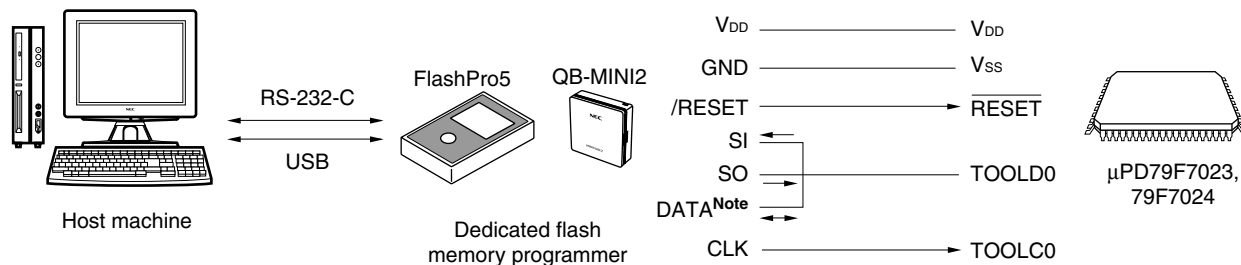
将μPD79F7023, 79F7024 系列产品安装到目标系统之前, 可以使用专用程序适配器(FA 系列)将数据写入闪存。

备注 FA 系列是株式会社 内藤电诚町田制作所(Naito Densei Machida Mfg. Co., Ltd.)的产品。

21.3 编程环境

μPD79F7023, 79F7024 闪存所需的编程环境如下所示。

图 21-2. 闪存程序写入环境



注 仅限 QB-MINI2

需要一台控制专用闪存编程器的主机。

专用闪存编程器与μPD79F7023, 79F7024 的接口使用 TOOLD0 引脚，通过专用单线 UART 进行写入和擦除等操作。在闪存进行离线写入时，必须使用专用程序适配器(FA 系列)。

表 21-2. 引脚连接

专用闪存编程器			PD79F7023, 79F7024 微控制器
信号名称	I/O	引脚功能	引脚名称
CLK	输出	时钟输入至 PD79F7023 和 79F7024 微控制器	TOOLC0/
SI	输入	接收信号	TOOLD0
SO	输出	发送信号	
DATA ^注	I/O	调试期间用于数据通信的输入输出信号	
/RESET	输出	复位信号	RESET
V _{DD}	I/O	V _{DD} 电压生成/电源监控	V _{DD}
GND	—	接地	V _{SS}

注 仅限 QB-MINI2

21.4 板上引脚连接

目标系统必须安装用于连接专用闪存编程器的连接器。电路板必须提供从正常操作模式切换到闪存编程模式的选择功能。

设置闪存编程模式时，不用于闪存编程的引脚状态与复位后的瞬时状态相同。因此，若外部器件不能立即识别复位后的状态，则必须采用以下方式连接引脚。

21.4.1 TOOL 引脚

闪存编程模式中使用的通信引脚如下所示。

表 21-3. 闪存编程模式中使用的通信引脚

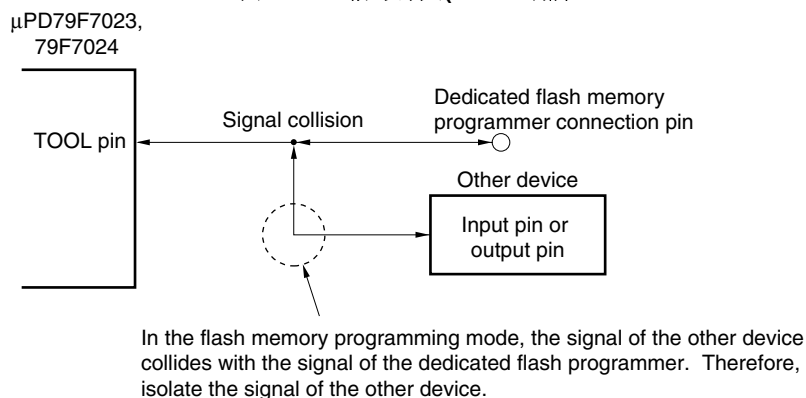
引脚名称	引脚连接
TOOLC0	直接连接专用闪存编程器，或者通过阻抗(10 kΩ)连接 VSS。
TOOLD0	直接连接专用闪存编程器，或者通过阻抗 (3 kΩ 至 10 kΩ)连接 VDD。

在线上已连接另一个器件，并连接专用闪存编程器时，应注意信号冲突和另一个器件的误操作等。

(1) 信号冲突

若与另一个器件连接的 TOOL 引脚再连接专用闪存编程器时，会产生信号冲突。为了避免这种情况，应隔离另一个连接器件，或使另一个器件处于高阻态。

图 21-3. 信号冲突(TOOL 引脚)

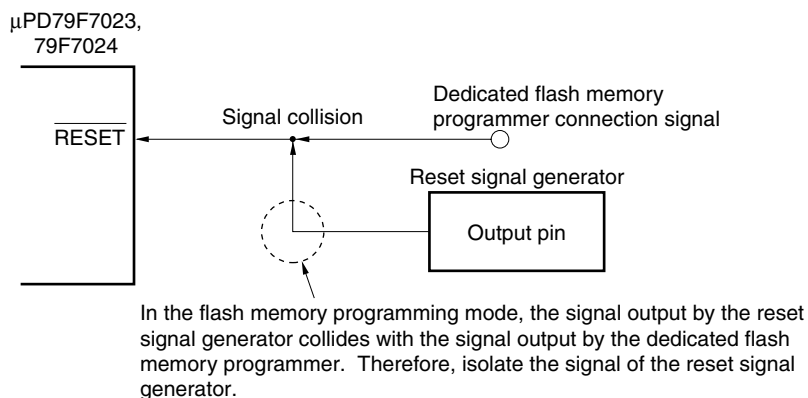


21.4.2 RESET 引脚

若将专用闪存编程器的复位信号接到 **RESET** 引脚（已连接到板上的复位信号发生器），则会产生信号冲突。为了避免这种情况，应隔离与复位信号发生器的连接。

闪存编程模式下，若从用户系统输入复位信号，则不能对闪存进行正确编程。因此，除了专用闪存编程器复位信号之外，禁止输入其他信号。

图 21-4. 信号冲突(RESET 引脚)



21.4.3 端口引脚

闪存编程模式下，不用于闪存编程的引脚状态与复位后的瞬时状态相同。若连接端口的外部器件不能立即识别复位后的状态，则必须通过电阻将端口引脚连接到 V_{DD} 或 V_{SS} 。

21.4.4 REGC 引脚

与正常操作相同的方式通过电容(0.47 至 1 μF)将 **REGC** 引脚连接到 **GND**。但在运行内部高速谐振时钟、外部主系统时钟中使用 **STOP** 模式时，推荐使用 0.47 μF 。另外，为了使用稳定的内部电压，请使用性能良好的电容器。

21.4.5 其他信号引脚

以正常操作模式下相同的状态连接 **X1**、**X2**。

备注 闪存编程模式下，使用内部高速振荡时钟(f_{IH})。

21.4.6 电源

若使用闪存编程器的电源输出，则将 **VDD** 引脚连接到闪存编程器的 **VDD**，**VSS** 引脚连接到闪存编程器的 **GND**。

若使用板上电源，则连接必须符合正常操作模式。

但由于通过闪存编程器进行电压监控，即使使用板上电源，也必须将 **VDD** 和 **VSS** 连接闪存编程器的 **VDD** 和 **GND**。

其他电源(**AVREF**, **VSS**)必须与正常操作模式的电源相同。

21.4.7 连接晶体/陶瓷谐振器时的板上写入

目标系统必须安装连接专用闪存编程器的连接器。电路板上必须提供从正常操作模式切换到闪存编程模式的选择功能。

设置闪存编程模式时，不用于闪存编程的引脚状态与复位后的瞬时状态相同。因此，若外部器件不能立即识别复位后的状态，则必须采用以下方式处理引脚。

自编程模式时的引脚状态与 HALT 模式时的相同。

X1(TOOLC0)和 X2(TOOLD0)引脚分别用作闪存编程时，若连接外部器件，则产生信号冲突。为防止信号间的冲突，隔离外部器件的连接。

同样，将 X1、X2 连接电容器时，由于通信中波形被改变，根据电容器容量出现不能通信的可能性。闪存编程期间，隔离与电容器的连接。

若系统时钟选择晶体/陶瓷谐振，则很难隔离电容器，装有谐振器状态的板上执行写入时，必须确定装有谐振器状态下的写入后，进行以下处理。

- 器件和谐振器之间尽量安装小型测试焊盘，通过测试焊盘连接编程器。布线尽量要短（参阅图 21-5 和表 21-4）。

图 21-5. 测试焊盘安装示例

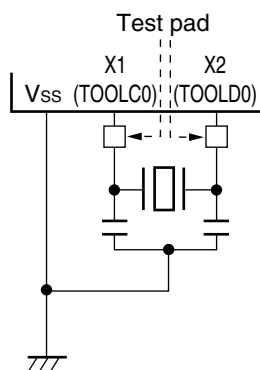


表 21-4. 使用时钟和测试焊盘安装

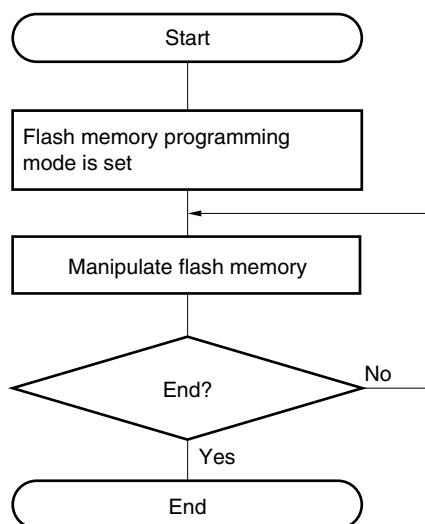
使用时钟		测试焊盘安装
高速内部振荡时钟		不需要
外部时钟		
晶体/陶瓷振荡时钟	安装谐振器之前	
	安装谐振器之后	需要

21.5 编程方法

21.5.1 控制闪存

闪存操作步骤如下图所示。

图 21-6. 闪存操作步骤



21.5.2 闪存编程模式

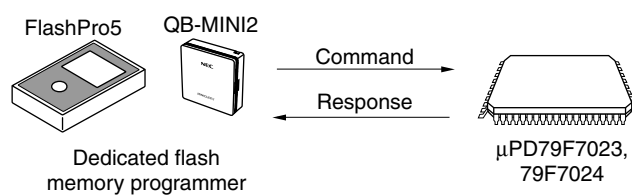
若通过专用闪存编程器重写闪存的内容，则必须将 μPD79F7023, 79F7024 设置为闪存编程模式。连接专用闪存编程器，开始通信时，系统切换到闪存编程模式。

执行线上写入时，使用跳线改变模式。

21.5.3 通信命令

μPD79F7023, 79F7024 使用命令与专用闪存编程器进行通信。从闪存编程器传送到 μPD79F7023, 79F7024 的信号成为命令，从 μPD79F7023, 79F7024 发往专用闪存编程器的命令成为响应命令。

图 21-7. 通信命令



μPD79F7023, 79F7024 的闪存控制命令如下表所示。所有这些命令都从编程器发出，μPD79F7023, 79F7024 根据相应的命令进行处理。

表 21-5. 闪存控制命令

类别	命令名称	功能
校验	Verify	比较闪存指定区域的内容与从编程器发送的数据。
擦除	Chip Erase	擦除整个闪存
	Block Erase	擦除闪存的指定区域
空白检查	Block Blank Check	验证闪存的指定块是否正确擦除
写入	Programming	将数据写入闪存的指定区域
获取信息	Silicon Signature	获取μPD79F7023 和 79F7024 信息(例如部品名称和闪存配置)。
	Version Get	获取μPD79F7023 和 79F7024 版本和固件版本。
	Checksum	获取指定区域的校验和数据。
安全	Security Set	设置安全信息
其他	Reset	用于检测通讯同步状态
	Baud Rate Set	设置 UART 时的波特率。

μPD79F7023, 79F7024 对专用闪存编程器发出的命令进行响应。μPD79F7023, 79F7024 发出的响应名称如下所示。

表 21-6. 对应名称

对应名称	功能
ACK	响应命令/数据
NAK	非响应命令/数据

21.6 安全设置

μPD79F7023, 79F7024 具有禁止重写闪存的用户程序的安全功能，因此，未经授权的第三方不能改变程序。通过使用安全设置命令，可以执行以下操作。在设置的下一个编程模式中，安全性设置有效。

- 禁止批擦除（片擦除）

在线/离线编程时，禁止对闪存所有块执行块擦除命令和批擦除(片擦除)命令。一旦禁止执行批擦除(片擦除)命令，则不能取消所有禁止设置(包括批擦除(片擦除)禁止)。

注意事项 一旦对批擦除进行安全性设置，则不能再对该设置执行擦除操作。另外，由于擦除命令无效，即使执行写命令，也不能写入与闪存存有数据不同的数据。

- 禁止块擦除

在线/离线编程期间，禁止对闪存内指定块擦进行块除命令。但是，使用自编程时可以执行块擦除。

- 禁止写入

在线/离线编程期间，禁止对闪存内所有块进行写入和块擦除命令。但是，使用自编程时可以执行块写入。

- 禁止重写引导集群 0

禁止对闪存中引导集群 0(0000H 至 0FFFH)进行批擦除命令和写命令。还禁止执行批擦除(片擦除)命令。

出厂时的初始状态下，允许所有批擦除（片擦除）/块擦除/写入/引导集群 0 的重写。通过在线/离线编程和自编程可以进行安全设置。安全设置可以相互结合使用。

通过执行批擦除（片擦除）命令，可以解除所有安全设置。

表 21-7 给出允许使用μPD79F7023 和 79F7024 安全功能时，擦除和写命令之间的关系。

表 21-7. 安全功能有效时与命令的关系

(1) 在线/离线编程期间

安全有效	执行命令		
	批擦除（芯片擦除）	块擦除	写入
禁止批擦除（片擦除）	禁止批擦除	禁止块擦除	允许执行
禁止块擦除	允许批擦除		允许执行
禁止写入			禁止执行
禁止重写引导集群 0	禁止批擦除	禁止擦除引导集群 0	禁止写入引导集群 0

注 确认写区域中没有写入数据。由于已设置的批擦除(片擦除)不能擦除，若数据没有擦除，则不要写入数据。

(2) 自编程期间

安全有效	执行命令	
	块擦除	写入
禁止批擦除（芯片擦除）	可块擦除	允许执行
禁止块擦除		
禁止写入		
禁止重写引导集群 0	禁止擦除引导集群 0	禁止写入引导集群 0

表 21-8 给出在各种编程模式中的安全设置方法

表 21-8. 在各种编程模式下的安全设置方法

(1) 在线/离线编程

安全	安全设置	禁止安全设置方法
禁止批擦除（芯片擦除）	通过专用闪存编程器 GUI 的设置	设置后不能解除
禁止块擦除		执行批擦除（片擦除）命令
禁止写入		
禁止重写引导集群 0		设置后不能解除

(2) 自编程

安全	安全设置	禁止安全设置方法
禁止批擦除（芯片擦除）	通过信息库的设置	设置后不能解除
禁止块擦除		在在线/离线编程期间执行批擦除（片擦除） （编程期间不能解除）
禁止写入		
禁止重写引导集群 0		

21.7 通过自编程进行闪存编程

μPD79F7023, 79F7024 支持自编程功能，即可以通过用户程序重写闪存。由于该功能允许用户应用程序使用 μPD79F7023, 79F7024 自编程库重写闪存，可用于现场程序升级。

若自编程期间产生中断，可以暂时停止自编程操作，并执行中断服务程序。若 EI 状态产生非屏蔽中断请求，则从自编程库直接转移到中断程序。随后再次恢复到自编程模式，重新开始自编程。但是，中断响应时间与正常操作模式下的不同。

- 注意事项**
1. 若自编程期间禁止中断，则必须采用与正常操作模式相同的方式，通过 DI 指令将 IE 标志清零（0）后，执行自编程库。若允许中断，则通过 EI 指令将 IE 标志置“1”，接收中断的中断屏蔽标志清零（0）后，执行自编程库。
 2. 执行子编程时，应固定稳压器的输出电压模式。
 3. 在正常功率模式下，重写闪存时，允许重写的电源电压范围在 $V_{DD} \geq 2.5V$ 。另外，在正常功率模式下，可通过自编程库重写程序区域。
 4. 低功耗模式下，禁止重写闪存。

备注 有关详细的自编程功能和自编程库，参阅“78K0 微控制器 用户手册 自编程库 01 型 (U18274J)”和“78K0 微控制器 自编程库 01 型 3.10 版 操作注意事项(文件) (ZUD-CD-09-0122)”。

21.7.1 自编程模式控制寄存器

自编程模式控制寄存器(FPCTL)控制自编程模式。
可用 1 位或 8 位存储器操作指令设置 FPCTL。
复位信号产生时，FPCTL 被清为 00H。

图 21-8. 自编程模式控制寄存器(FPCTL)的格式

地址: FF2BH	复位后: 00H	R/W						
符号	7	6	5	4	3	2	1	<0>
FPCTL	0	0	0	0	0	0	0	FLMDPUP 注

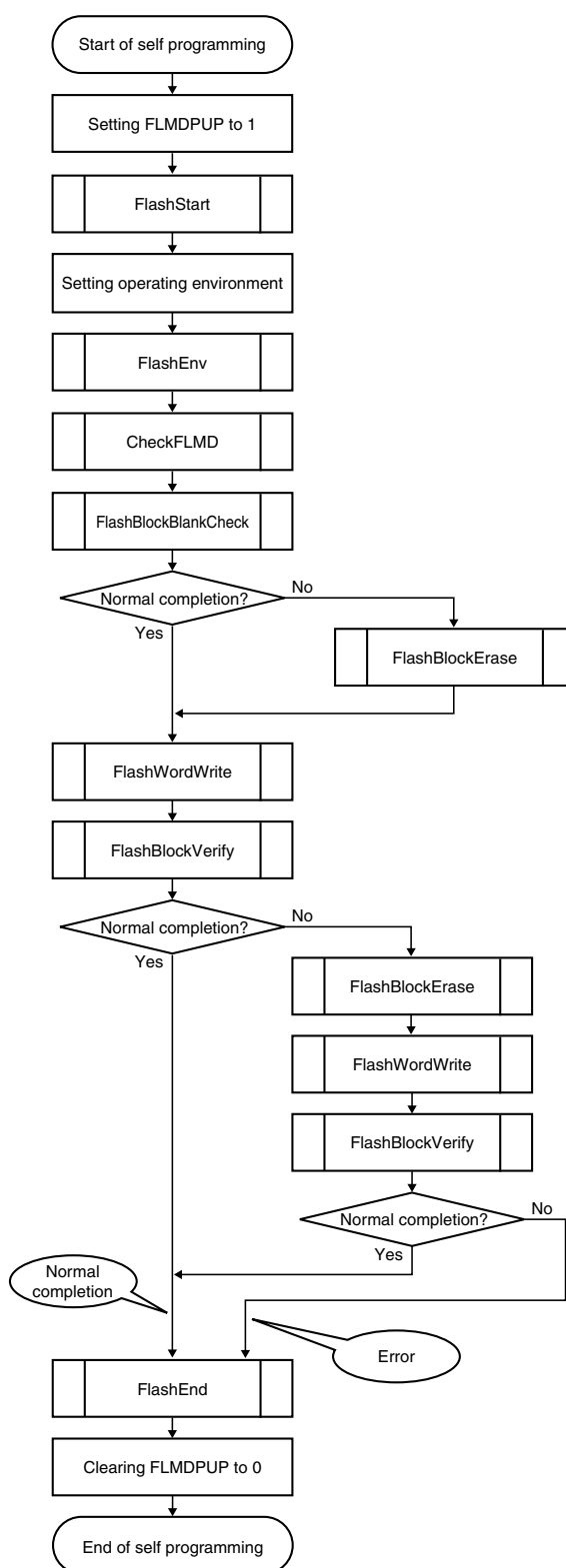
FLMDPUP 注	自编程模式控制
0	正常操作模式
1	自编程模式

注 FLMDPUP 位，在执行正常用户编程中必须置“0”（正常操作模式），在执行自编程中必须置“1”（自编程模式）。在正常操作模式下，由于闪存的重写电路不执行操作，因此即使固件和软件执行重写，也不执行实际重写操作。

21.7.2 自编程流程(重写闪存寄存器)

利用自编程库，执行闪存的重写流程如下所示。

图 21-9. 自编程（重写闪存寄存器）流程



备注 有关详细的自编程功能和自编程库，参阅“78K0 微控制器 用户手册 自编程库类型 01 (U18274E)”和“78K0 微控制器 自编程库类型 01 Ver. 3.10 操作注意事项(通报文件) (ZUD-CD-09-0122)”。

21.7.3 引导交换功能

在自编程的重写引导区域期间，若因瞬间断电或其它原因而致重写失败，则引导区域的数据将会丢失，无法通过复位重启程序或重写。

引导交换功能用于解决此类问题。

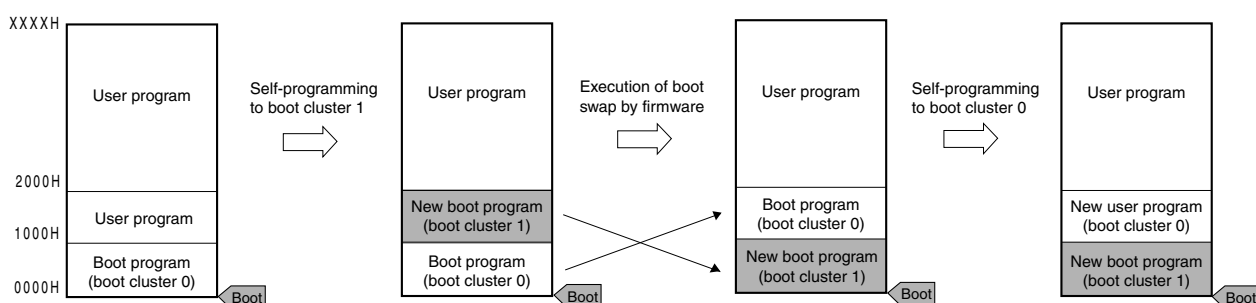
消除在自编程中引导程序区域的引导集群 0*之前，应预先将新的引导程序写入引导集群 1。该程序正常写入引导集群 1 后，通过μPD79F7023, 79F7024 微控制器固件的信息设置功能将引导集群 1 和引导集群 2 的内容进行交换，以便引导集群 1 用作引导区域。接着对初始引导程序区域的引导机群 0 执行擦除或写操作。

由于程序在复位重启时，从引导集群 1 进行引导交换，因此，在重写引导程序区域期间出现断电，也能够正确执行程序。

注 在 4KB 区域的引导集群中，通过引导交换功能可以交换引导集群 0 和引导集群 1 的内容。

注意事项 4KB ROM 规格的产品不能使用引导交换功能。

图 21-10. 引导交换功能

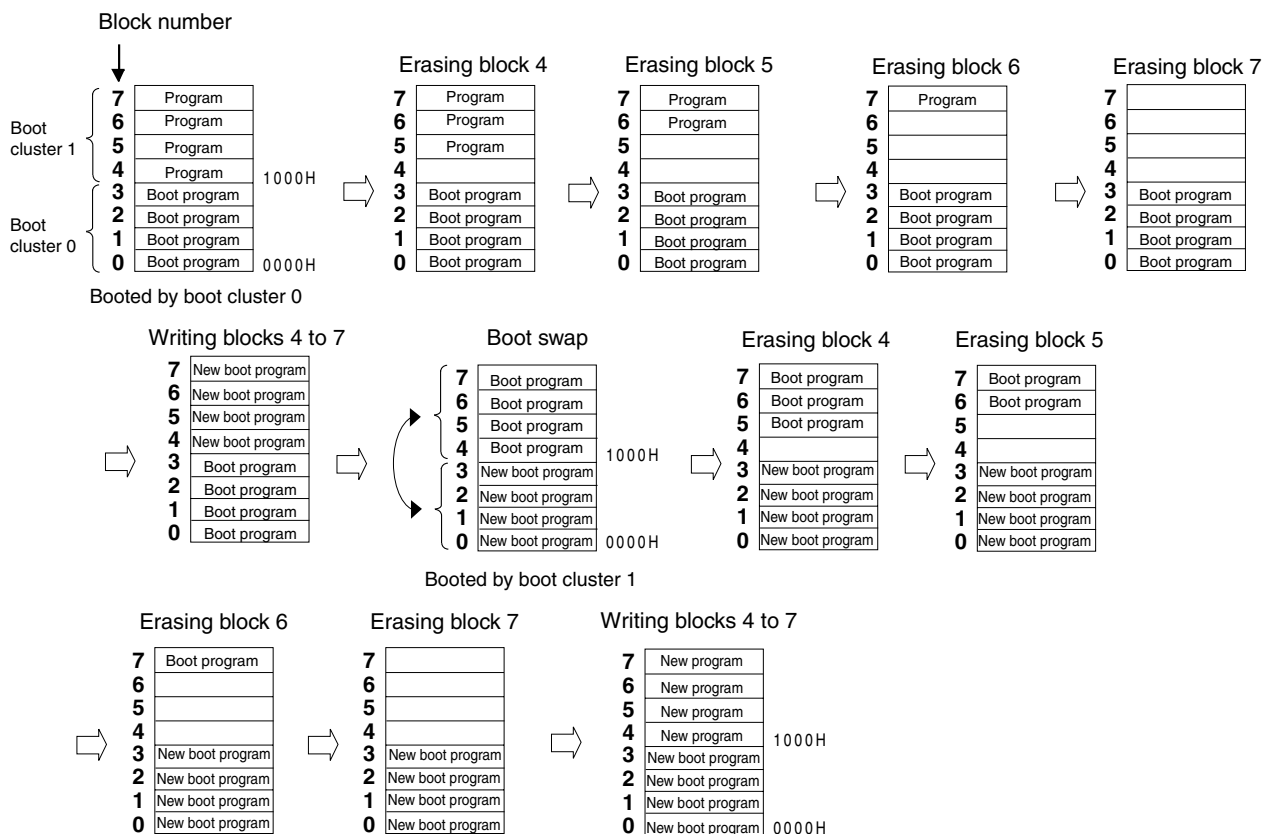


上图示例如下。

引导集群 0: 引导交换前的引导程序区域

引导集群 1: 引导交换后的引导程序区域

图 21-11. 引导交换的执行示例



第二十二章 On-chip 调试功能

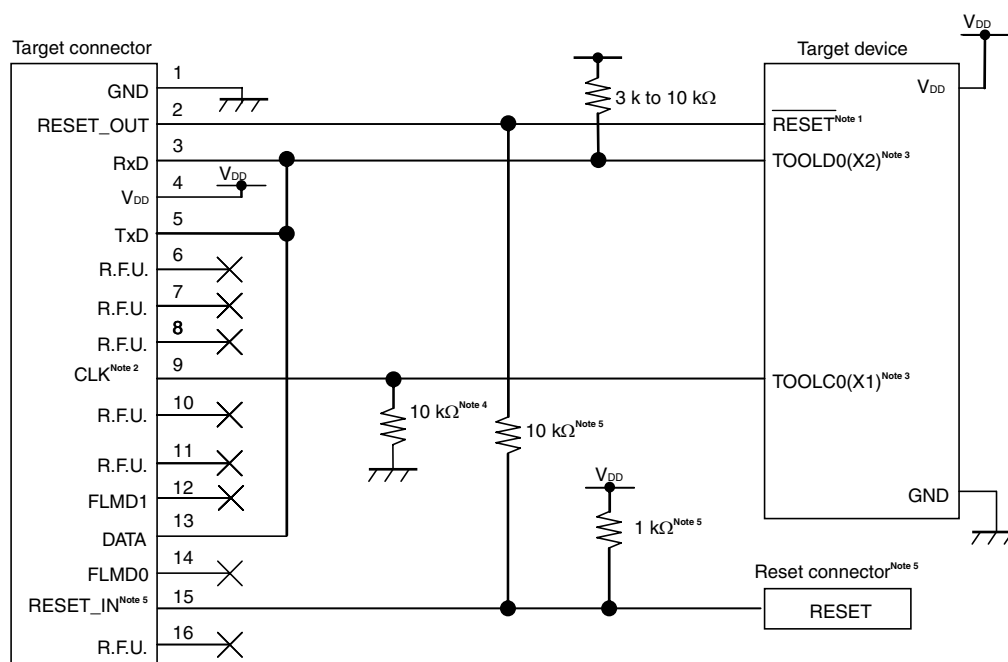
22.1 QB-MINI2 与μPD79F7023, 79F7024 的连接

μPD79F7023, 79F7024 通过 On-chip 调试仿真器(QB-MINI2), 使用 V_{DD} , $\overline{RESET}$, TOOLC0/X1, TOOLD0/X2 和 V_{SS} 引脚与主机进行通信。

- 注意事项
1. μPD79F7023, 79F7024 具有开发/评估选择的 On-chip 调试功能。使用 On-chip 调试功能时, 可能会出现超过闪存所定重写次数的现象, 从而不能保证产品的可靠性, 因此, 请勿将本功能用于量产化的产品。瑞萨电子公司不接受任何与 On-chip 调试功能相关的产品投诉。
 2. 将 On-chip 调试中转换到 STOP 模式时, 虽然内部高速振荡器继续振荡, 但不影响 On-chip 调试操作。

图 22-1. QB-MINI2 与μPD79F7023, 79F7024 的连接示例 (1/2)

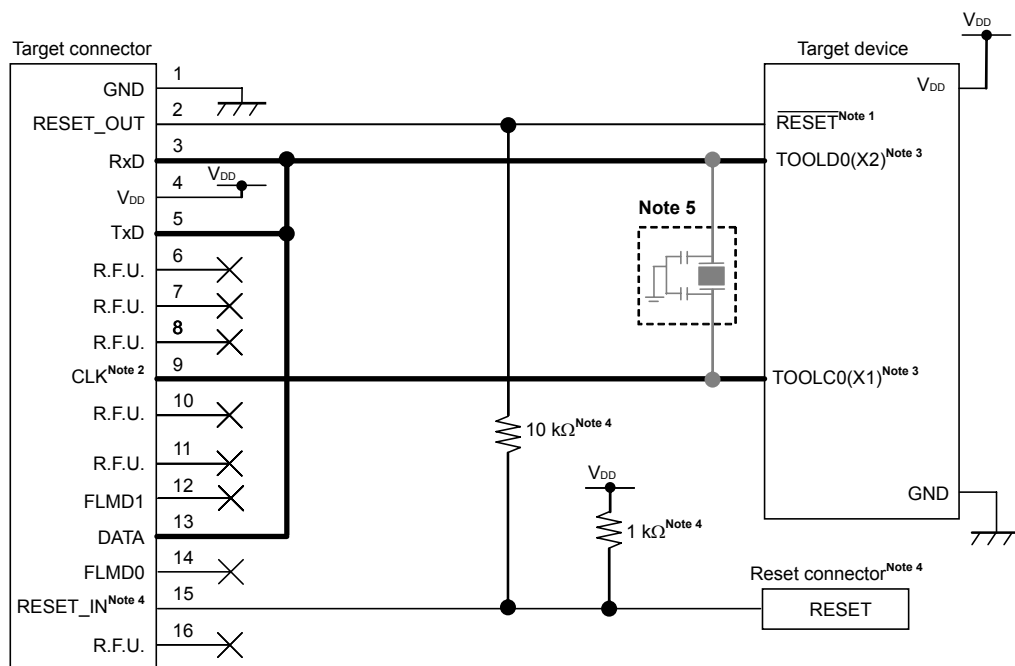
(1) 使用 TOOLC0 和 TOOLD0 引脚时(不使用 X1 振荡器或 EXCLK 输入时钟, 均执行调试和编程)



- 注 1. 若含有电容器等的电容元件, 则 On-chip 调试功能可能出现异常操作。
2. On-chip 调试期间, 78K0-OCD 板上提供的时钟、QB-MINI2 内部生成的 4/8MHz 时钟和器件内部高速振荡器可以用作目标器件的操作时钟。
闪存编程期间, 只能使用器件内部高速振荡器。
3. On-chip 调试期间, 由于这些引脚用作 On-chip 调试专用引脚, 忽略用户程序的设置。但若这些引脚设置为输入引脚, 则必须进行引脚处理(未连接 QB-MINI2 时, 处于开路状态)。
4. 操作目标器件时(未连接 QB-MINI2), 不使用引脚(输入电路开路)进行处理。(使用振荡电路时, 不需要。)
5. 该复位电路的输出假设为 N-ch 漏极开路缓冲器(输出电阻: 100Ω 以下)的电路连接。

图 22-1. QB-MINI2 和μPD79F7023, 79F7024 的连接实例(2/2)

(2) 使用 TOOLC0 和 TOOLD0 引脚时(使用 X1/X2 振荡器, 均执行调试和编程)



- 注 1. 若含有电容器等的电容元件, 则 On-chip 调试功能可能出现不能正常操作。
2. On-chip 调试期间, 78K0-OCD 板上提供的时钟、QB-MINI2 内部生成的 4/8MHz 时钟和器件内部高速振荡器可以用作目标器件的操作时钟。
flash 编程期间, 只能使用器件的内部高速振荡器。
3. On-chip 调试期间, 由于这些引脚用作 On-chip 调试专用引脚, 忽略用户程序的设置。但是, 若这些引脚设置为输入引脚, 则必须进行引脚处理 (未连接 QB-MINI2 时, 处于开路状态)。
4. 该复位电路的输出假设为 N-ch 漏极开路缓冲器 (输出电阻: 100 Ω 以下) 的电路连接。详阅配备 QB-MINI2 编程功能的 On-chip 调试仿真器用户手册(U18371J)的 4.1.3 复位引脚的连接。
5. On-chip 调试和闪存编程期间, 严禁安装振荡电路。
目标器件操作(未连接 QB-MINI2)时, 为了防止振荡电路因布线容量不能振动, 应采取设置跨接等隔离目标连接器和振荡电路的措施。
使用调试的下载程序, 在没有连接 QB-MINI2 时不能操作。

注意事项 图中用粗线标出的线路(TOOLD0 和 TOOLC0), 从器件引脚连接到 QB-MINI2 连接器的长度不得超过 30mm 以上, 或者进行 GND 屏蔽。

22.2 On-chip 调试安全 ID

在μPD79F7023, 79F7024 中,为了防止第三方读取存储器的内容,在内存的 0084H 预留 On-chip 调试操作控制位(参阅第二十章 选项字节)、0085H 至 008EH 之间预留 On-chip 调试安全 ID 的设置区域。

在自编程中使用引导交换功能时,进行 0084H、0085H 至 008EH 和 1083H 以及 1085H 至 108EH 的切换,因此,应预先设置与 1084H 和 1085H 至 108EH 相同的值。

有关详细的 On-chip 调试安全 ID,参阅配备 QB-MINI2 编程功能的 On-chip 调试仿真器 用户手册(U18371J)。

表 22-1. On-chip 调试安全 ID

地址	On-chip 调试安全 ID
0085H 至 008EH	任意 10 个字节的 ID 码
1085H 至 108EH	

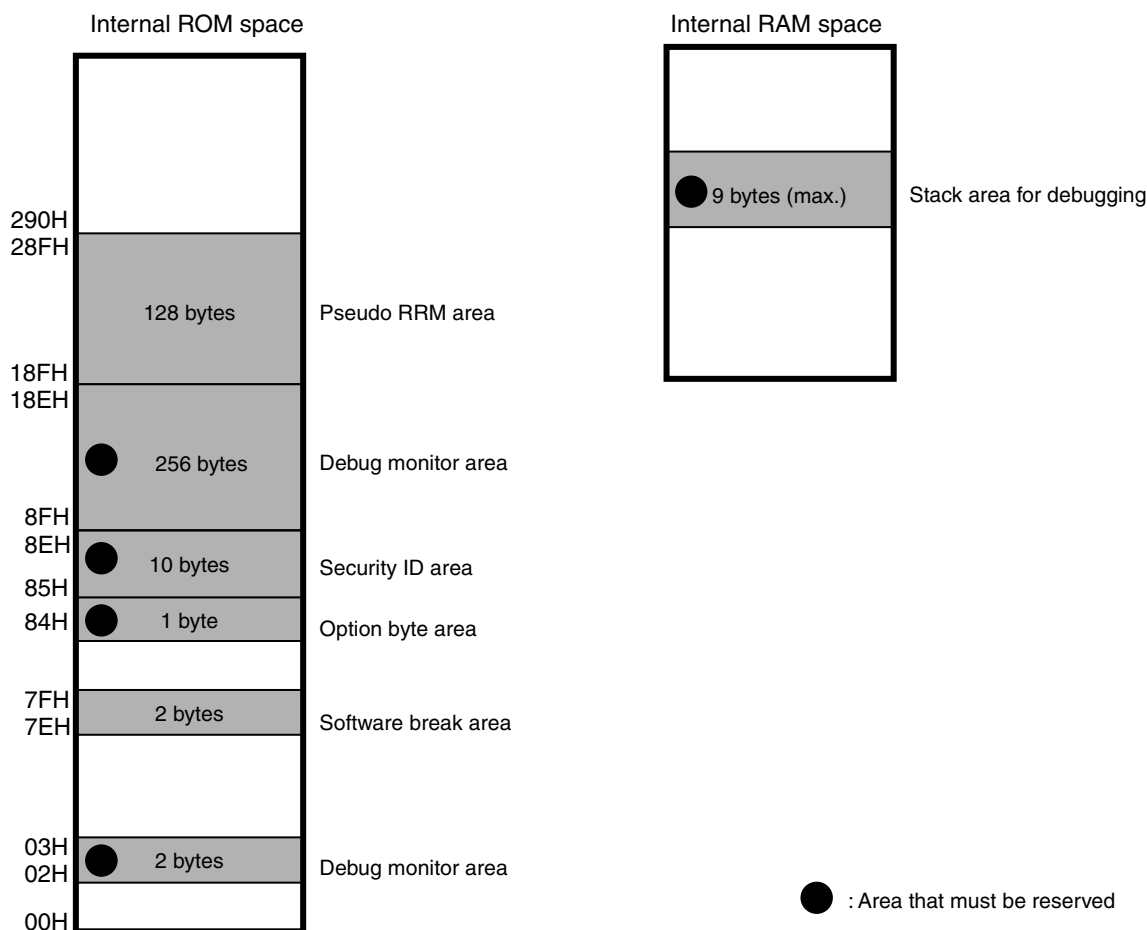
22.3 用户资源保留

为了实现与目标器件的通信或各器件功能，QB-MINI2 使用图 22-2 中表示的灰色部分的用户存储区域。图中(●) 的表示区域在调试时必须使用，其他区域根据调试功能进行使用。

通过用户程序或连接程序选择保留这些区域。

有关详细的这些区域保留，参阅配备 QB-MINI2 编程功能的 On-chip 调试仿真器 用户手册(U18371J)。

图 22-2. QB-MINI2 使用的预置区域



第二十三章 指令集

本章以表格形式列出了 μPD79F7023, 79F7024 的指令集。有关详细的各操作和操作码, 参阅附件 **78K/0 系列 用户手册 指令篇(U12326E)**。

23.1 应用操作表

23.1.1 操作标识符和指定方法

根据指令操作数的标识方法(详阅汇编程序编程规范), 在各指令的“操作数”栏列出操作数。若有两种或两种以上的标识方法, 则可选其中之一。大写字母和符号 #、!、\$、[] 为关键字, 必须按其原文写入。各种符号含义如下。

- #: 指定立即数
- !: 指定绝对地址
- \$: 指定相对地址
- []: 指定间接地址

立即数用来描述一个数值型数据或标号。使用标号时, 必须写入符号 #、!、\$ 和 []。

操作数寄存器的标识符 **r** 和 **rp**, 可以使用功能名称(X, A, C 等)或绝对名称(下括号内的名称, R0, R1, R2 等)进行标识。

表 23-1. 操作数标识符和指定方法

标识符	指定方法
r	X (R0), A (R1), C (R2), B (R3), E (R4), D (R5), L (R6), H (R7)
rp	AX (RP0), BC (RP1), DE (RP2), HL (RP3)
sfr	特殊功能寄存器符号 ^注
sfrp	特殊功能寄存器符号(仅限 16 位可操作寄存器偶地址) ^注
saddr	FE20H 至 FF1FH 立即数或标号
saddrp	FE20H 至 FF1FH 立即数或标号(仅限地址)
addr16	0000H 至 FFFFH 立即数或标号 (仅限 16 位数据传送指令地址)
addr11	0800H 至 0FFFH 立即数或标号
addr5	0040H 至 007FH 立即数或标号(仅限地址)
字	16 位立即数或标号
字节	8 位立即数或标号
位	3 位立即数或标号
RBn	RB0 至 RB3

注 不能使用这些操作数访问地址 FFD0H 至 FFD7H。

备注 有关特殊功能寄存器符号, 参阅表 3-6 特殊功能寄存器。

23.1.2 操作栏描述

A:	A 寄存器; 8 位累加器
X:	X 寄存器
B:	B 寄存器
C:	C 寄存器
D:	D 寄存器
E:	E 寄存器
H:	H 寄存器
L:	L 寄存器
AX:	AX 寄存器对; 16 位累加器
BC:	BC 寄存器对
DE:	DE 寄存器对
HL:	HL 寄存器对
PC:	程序计数器
SP:	堆栈指针
PSW:	程序状态字
CY:	载波标志
AC:	辅助载波标志
Z:	零标志
RBS:	寄存器储存体选择标志
IE:	中断请求允许标志
():	括号中的地址或寄存器的内容所指的存储内容
X _H , X _L :	16 位寄存器的高 8 位和低 8 位
∧:	逻辑积 (和)
∨:	逻辑和 (或)
⊕:	排他逻辑和 (异或)
—:	取反数据
addr16:	16 位立即数或标号
jdsp8:	带符号的 8 位数据 (位移值)

23.1.3 标志操作列描述

(Blank):	无影响
0:	清零
1:	置“1”
×	根据结果, 设置/清除
R:	恢复先前保存的值。

23.2 操作表

指令组	助记符	操作数	位	时钟		操作	标志 Z AC CY
				注 1	注 2		
8 位数据传送	MOV	r, #字节	2	4	—	$r \leftarrow \text{字节}$	
		saddr, #字节	3	6	7	$(saddr) \leftarrow \text{字节}$	
		sfr, #字节	3	—	7	$sfr \leftarrow \text{字节}$	
		A, r ^{注 3}	1	2	—	$A \leftarrow r$	
		r, A ^{注 3}	1	2	—	$r \leftarrow A$	
		A, saddr	2	4	5	$A \leftarrow (saddr)$	
		saddr, A	2	4	5	$(saddr) \leftarrow A$	
		A, sfr	2	—	5	$A \leftarrow sfr$	
		sfr, A	2	—	5	$sfr \leftarrow A$	
		A, !addr16	3	8	9	$A \leftarrow (addr16)$	
		!addr16, A	3	8	9	$(addr16) \leftarrow A$	
		PSW, #字节	3	—	7	$PSW \leftarrow \text{字节}$	x x x
		A, PSW	2	—	5	$A \leftarrow PSW$	
		PSW, A	2	—	5	$PSW \leftarrow A$	x x x
		A, [DE]	1	4	5	$A \leftarrow (DE)$	
		[DE], A	1	4	5	$(DE) \leftarrow A$	
		A, [HL]	1	4	5	$A \leftarrow (HL)$	
		[HL], A	1	4	5	$(HL) \leftarrow A$	
		A, [HL + 字节]	2	8	9	$A \leftarrow (HL + \text{字节})$	
		[HL + 字节], A	2	8	9	$(HL + \text{字节}) \leftarrow A$	
		A, [HL + B]	1	6	7	$A \leftarrow (HL + B)$	
		[HL + B], A	1	6	7	$(HL + B) \leftarrow A$	
		A, [HL + C]	1	6	7	$A \leftarrow (HL + C)$	
		[HL + C], A	1	6	7	$(HL + C) \leftarrow A$	
	XCH	A, r ^{注 3}	1	2	—	$A \leftrightarrow r$	
		A, saddr	2	4	6	$A \leftrightarrow (saddr)$	
		A, sfr	2	—	6	$A \leftrightarrow (sfr)$	
		A, !addr16	3	8	10	$A \leftrightarrow (addr16)$	
		A, [DE]	1	4	6	$A \leftrightarrow (DE)$	
		A, [HL]	1	4	6	$A \leftrightarrow (HL)$	
		A, [HL + 字节]	2	8	10	$A \leftrightarrow (HL + \text{字节})$	
		A, [HL + B]	2	8	10	$A \leftrightarrow (HL + B)$	
		A, [HL + C]	2	8	10	$A \leftrightarrow (HL + C)$	

- 注 1. 访问内部高速 RAM 时或没有无数据访问的指令
 2. 访问内部高速 RAM 以外的区域时
 3. “r = A” 除外

- 备注 1. 一个指令时钟周期是指由处理器时钟控制寄存器(PCC)选择的 CPU 时钟(f_{CPU})的一个周期。
 2. 该时钟周期用于内部 ROM 程序。

指令组	助记符	操作数	位	时钟		操作	标志
				注 1	注 2		Z AC CY
16 位数据 传送	MOVW	rp, #字	3	6	—	rp ← 字	
		saddrp, #字	4	8	10	(saddrp) ← 字	
		sfrp, #字	4	—	10	sfrp ← 字	
		AX, saddrp	2	6	8	AX ← (saddrp)	
		saddrp, AX	2	6	8	(saddrp) ← AX	
		AX, sfrp	2	—	8	AX ← sfrp	
		sfrp, AX	2	—	8	sfrp ← AX	
		AX, rp ^{注 3}	1	4	—	AX ← rp	
		rp, AX ^{注 3}	1	4	—	rp ← AX	
		AX, !addr16	3	10	12	AX ← (addr16)	
		!addr16, AX	3	10	12	(addr16) ← AX	
	XCHW	AX, rp ^{注 3}	1	4	—	AX ↔ rp	
8 位操作	ADD	A, #字节	2	4	—	A, CY ← A + 字节	× × ×
		saddr, #字节	3	6	8	(saddr), CY ← (saddr) + 字节	× × ×
		A, r ^{注 4}	2	4	—	A, CY ← A + r	× × ×
		r, A	2	4	—	r, CY ← r + A	× × ×
		A, saddr	2	4	5	A, CY ← A + (saddr)	× × ×
		A, !addr16	3	8	9	A, CY ← A + (addr16)	× × ×
		A, [HL]	1	4	5	A, CY ← A + (HL)	× × ×
		A, [HL + 字节]	2	8	9	A, CY ← A + (HL + 字节)	× × ×
		A, [HL + B]	2	8	9	A, CY ← A + (HL + B)	× × ×
		A, [HL + C]	2	8	9	A, CY ← A + (HL + C)	× × ×
	ADDC	A, #字节	2	4	—	A, CY ← A + 字节 + CY	× × ×
		saddr, #字节	3	6	8	(saddr), CY ← (saddr) + 字节 + CY	× × ×
		A, r ^{注 4}	2	4	—	A, CY ← A + r + CY	× × ×
		r, A	2	4	—	r, CY ← r + A + CY	× × ×
		A, saddr	2	4	5	A, CY ← A + (saddr) + CY	× × ×
		A, !addr16	3	8	9	A, CY ← A + (addr16) + C	× × ×
		A, [HL]	1	4	5	A, CY ← A + (HL) + CY	× × ×
		A, [HL + 字节]	2	8	9	A, CY ← A + (HL + 字节) + CY	× × ×
		A, [HL + B]	2	8	9	A, CY ← A + (HL + B) + CY	× × ×
		A, [HL + C]	2	8	9	A, CY ← A + (HL + C) + CY	× × ×

- 注 1. 访问内部高速 RAM 时或没有无数据访问的指令
 2. 访问内部高速 RAM 以外的区域时
 3. 仅在 rp = BC, DE 或 HL 时
 4. “r = A” 除外

- 备注 1. 一个指令时钟周期是指由处理器时钟控制寄存器(PCC)选择的 CPU 时钟(fCPU)的一个周期。
 2. 该时钟周期用于内部 ROM 程序。

指令组	助记符	操作数	位	时钟		操作	标志		
				注 1	注 2		Z	AC	CY
8 位操作	SUB	A, #字节	2	4	—	$A, CY \leftarrow A - \text{字节}$	×	×	×
		saddr, #字节	3	6	8	$(saddr), CY \leftarrow (saddr) - \text{字节}$	×	×	×
		A, r ^{注3}	2	4	—	$A, CY \leftarrow A - r$	×	×	×
		r, A	2	4	—	$r, CY \leftarrow r - A$	×	×	×
		A, saddr	2	4	5	$A, CY \leftarrow A - (saddr)$	×	×	×
		A, laddr16	3	8	9	$A, CY \leftarrow A - (addr16)$	×	×	×
		A, [HL]	1	4	5	$A, CY \leftarrow A - (HL)$	×	×	×
		A, [HL + 字节]	2	8	9	$A, CY \leftarrow A - (HL + \text{字节})$	×	×	×
		A, [HL + B]	2	8	9	$A, CY \leftarrow A - (HL + B)$	×	×	×
		A, [HL + C]	2	8	9	$A, CY \leftarrow A - (HL + C)$	×	×	×
	SUBC	A, #字节	2	4	—	$A, CY \leftarrow A - \text{字节} - CY$	×	×	×
		saddr, #字节	3	6	8	$(saddr), CY \leftarrow (saddr) - \text{字节} - CY$	×	×	×
		A, r ^{注3}	2	4	—	$A, CY \leftarrow A - r - CY$	×	×	×
		r, A	2	4	—	$r, CY \leftarrow r - A - CY$	×	×	×
		A, saddr	2	4	5	$A, CY \leftarrow A - (saddr) - CY$	×	×	×
		A, laddr16	3	8	9	$A, CY \leftarrow A - (addr16) - CY$	×	×	×
		A, [HL]	1	4	5	$A, CY \leftarrow A - (HL) - CY$	×	×	×
		A, [HL + 字节]	2	8	9	$A, CY \leftarrow A - (HL + \text{字节}) - CY$	×	×	×
		A, [HL + B]	2	8	9	$A, CY \leftarrow A - (HL + B) - CY$	×	×	×
		A, [HL + C]	2	8	9	$A, CY \leftarrow A - (HL + C) - CY$	×	×	×
	AND	A, #字节	2	4	—	$A \leftarrow A \wedge \text{字节}$	×		
		saddr, #字节	3	6	8	$(saddr) \leftarrow (saddr) \wedge \text{字节}$	×		
		A, r ^{注3}	2	4	—	$A \leftarrow A \wedge r$	×		
		r, A	2	4	—	$r \leftarrow r \wedge A$	×		
		A, saddr	2	4	5	$A \leftarrow A \wedge (saddr)$	×		
		A, laddr16	3	8	9	$A \leftarrow A \wedge (addr16)$	×		
		A, [HL]	1	4	5	$A \leftarrow A \wedge (HL)$	×		
		A, [HL + 字节]	2	8	9	$A \leftarrow A \wedge (HL + \text{字节})$	×		
		A, [HL + B]	2	8	9	$A \leftarrow A \wedge (HL + B)$	×		
		A, [HL + C]	2	8	9	$A \leftarrow A \wedge (HL + C)$	×		

- 注 1. 访问内部高速 RAM 时或没有无数据访问的指令
 2. 访问内部高速 RAM 以外的区域时
 3. “r = A” 除外

- 备注 1. 一个指令时钟周期是指由处理器时钟控制寄存器(PCC)选择的 CPU 时钟(f_{cpu})的一个周期。
 2. 该时钟周期用于内部 ROM 程序。

指令组	助记符	操作数	位	时钟		操作	标志
				注 1	注 2		Z AC CY
8 位操作	OR	A, #字节	2	4	—	$A \leftarrow A \vee \text{字节}$	×
		saddr, #字节	3	6	8	$(saddr) \leftarrow (saddr) \vee \text{字节}$	×
		A, r ^{注 3}	2	4	—	$A \leftarrow A \vee r$	×
		r, A	2	4	—	$r \leftarrow r \vee A$	×
		A, saddr	2	4	5	$A \leftarrow A \vee (saddr)$	×
		A, !addr16	3	8	9	$A \leftarrow A \vee (\text{addr16})$	×
		A, [HL]	1	4	5	$A \leftarrow A \vee (\text{HL})$	×
		A, [HL + 字节]	2	8	9	$A \leftarrow A \vee (\text{HL} + \text{字节})$	×
		A, [HL + B]	2	8	9	$A \leftarrow A \vee (\text{HL} + B)$	×
		A, [HL + C]	2	8	9	$A \leftarrow A \vee (\text{HL} + C)$	×
	XOR	A, #字节	2	4	—	$A \leftarrow A \oplus \text{字节}$	×
		saddr, #字节	3	6	8	$(saddr) \leftarrow (saddr) \oplus \text{字节}$	×
		A, r ^{注 3}	2	4	—	$A \leftarrow A \oplus r$	×
		r, A	2	4	—	$r \leftarrow r \oplus A$	×
		A, saddr	2	4	5	$A \leftarrow A \oplus (saddr)$	×
		A, !addr16	3	8	9	$A \leftarrow A \oplus (\text{addr16})$	×
		A, [HL]	1	4	5	$A \leftarrow A \oplus (\text{HL})$	×
		A, [HL + 字节]	2	8	9	$A \leftarrow A \oplus (\text{HL} + \text{字节})$	×
		A, [HL + B]	2	8	9	$A \leftarrow A \oplus (\text{HL} + B)$	×
		A, [HL + C]	2	8	9	$A \leftarrow A \oplus (\text{HL} + C)$	×
	CMP	A, #字节	2	4	—	$A - \text{字节}$	× × ×
		saddr, #字节	3	6	8	$(saddr) - \text{字节}$	× × ×
		A, r ^{注 3}	2	4	—	$A - r$	× × ×
		r, A	2	4	—	$r - A$	× × ×
		A, saddr	2	4	5	$A - (saddr)$	× × ×
		A, !addr16	3	8	9	$A - (\text{addr16})$	× × ×
		A, [HL]	1	4	5	$A - (\text{HL})$	× × ×
		A, [HL + 字节]	2	8	9	$A - (\text{HL} + \text{字节})$	× × ×
		A, [HL + B]	2	8	9	$A - (\text{HL} + B)$	× × ×
		A, [HL + C]	2	8	9	$A - (\text{HL} + C)$	× × ×

- 注 1. 访问内部高速 RAM 时或没有无数据访问的指令
 2. 访问内部高速 RAM 以外的区域时
 3. “r = A” 除外

- 备注 1. 一个指令时钟周期是指由处理器时钟控制寄存器(PCC)选择的 CPU 时钟(f_{cpu})的一个周期。
 2. 该时钟周期用于内部 ROM 程序。

指令组	助记符	操作数	位	时钟		操作	标志
				注 1	注 2		Z AC CY
16 位操作	ADDW	AX, #字	3	6	—	$AX, CY \leftarrow AX + \text{字}$	× × ×
	SUBW	AX, #字	3	6	—	$AX, CY \leftarrow AX - \text{字}$	× × ×
	CMPW	AX, #字	3	6	—	$AX - \text{字}$	× × ×
乘/除	MULU	X	2	16	—	$AX \leftarrow A \times X$	
	DIVUW	C	2	25	—	$AX \text{ (商)}, C \text{ (余数)} \leftarrow AX \div C$	
递增/递减	INC	r	1	2	—	$r \leftarrow r + 1$	× ×
		saddr	2	4	6	$(saddr) \leftarrow (saddr) + 1$	× ×
	DEC	r	1	2	—	$r \leftarrow r - 1$	× ×
		saddr	2	4	6	$(saddr) \leftarrow (saddr) - 1$	× ×
	INCW	rp	1	4	—	$rp \leftarrow rp + 1$	
	DECW	rp	1	4	—	$rp \leftarrow rp - 1$	
循环	ROR	A, 1	1	2	—	$(CY, A_7 \leftarrow A_0, A_{m-1} \leftarrow A_m) \times 1 \text{ 次}$	×
	ROL	A, 1	1	2	—	$(CY, A_0 \leftarrow A_7, A_{m+1} \leftarrow A_m) \times 1 \text{ 次}$	×
	RORC	A, 1	1	2	—	$(CY \leftarrow A_0, A_7 \leftarrow CY, A_{m-1} \leftarrow A_m) \times 1 \text{ 次}$	×
	ROLC	A, 1	1	2	—	$(CY \leftarrow A_7, A_0 \leftarrow CY, A_{m+1} \leftarrow A_m) \times 1 \text{ 次}$	×
	ROR4	[HL]	2	10	12	$A_{3-0} \leftarrow (HL)_{3-0}, (HL)_{7-4} \leftarrow A_{3-0},$ $(HL)_{3-0} \leftarrow (HL)_{7-4}$	
	ROL4	[HL]	2	10	12	$A_{3-0} \leftarrow (HL)_{7-4}, (HL)_{3-0} \leftarrow A_{3-0},$ $(HL)_{7-4} \leftarrow (HL)_{3-0}$	
BCD 调整	ADJBA		2	4	—	加法后的十进制调整	× × ×
	ADJBS		2	4	—	减法后的十进制调整	× × ×
位操作	MOV1	CY, saddr.位	3	6	7	$CY \leftarrow (saddr.\text{位})$	×
		CY, sfr.位	3	—	7	$CY \leftarrow \text{sfr.位}$	×
		CY, A.位	2	4	—	$CY \leftarrow A.\text{位}$	×
		CY, PSW.位	3	—	7	$CY \leftarrow \text{PSW.位}$	×
		CY, [HL].位	2	6	7	$CY \leftarrow (HL).\text{位}$	×
		saddr.位, CY	3	6	8	$(saddr.\text{位}) \leftarrow CY$	
		sfr.位, CY	3	—	8	$\text{sfr.位} \leftarrow CY$	
		A.位, CY	2	4	—	$A.\text{位} \leftarrow CY$	
		PSW.位, CY	3	—	8	$\text{PSW.位} \leftarrow CY$	× ×
		[HL].位, CY	2	6	8	$(HL).\text{位} \leftarrow CY$	

注 1. 访问内部高速 RAM 时或没有无数据访问的指令

2. 访问内部高速 RAM 以外的区域时

备注 1. 一个指令时钟周期是指由处理器时钟控制寄存器(PCC)选择的 CPU 时钟(f_{CPU})的一个周期。

2. 该时钟周期用于内部 ROM 程序。

指令组	助记符	操作数	位	时钟		操作	标志
				注 1	注 2		Z AC CY
位操作	AND1	CY, saddr.位	3	6	7	$CY \leftarrow CY \wedge (saddr.位)$	×
		CY, sfr.位	3	—	7	$CY \leftarrow CY \wedge sfr.位$	×
		CY, A.位	2	4	—	$CY \leftarrow CY \wedge A.位$	×
		CY, PSW.位	3	—	7	$CY \leftarrow CY \wedge PSW.位$	×
		CY, [HL].位	2	6	7	$CY \leftarrow CY \wedge (HL).位$	×
	OR1	CY, saddr.位	3	6	7	$CY \leftarrow CY \vee (saddr.位)$	×
		CY, sfr.位	3	—	7	$CY \leftarrow CY \vee sfr.位$	×
		CY, A.位	2	4	—	$CY \leftarrow CY \vee A.位$	×
		CY, PSW.位	3	—	7	$CY \leftarrow CY \vee PSW.位$	×
		CY, [HL].位	2	6	7	$CY \leftarrow CY \vee (HL).位$	×
	XOR1	CY, saddr.位	3	6	7	$CY \leftarrow CY \oplus (saddr.位)$	×
		CY, sfr.位	3	—	7	$CY \leftarrow CY \oplus sfr.位$	×
		CY, A.位	2	4	—	$CY \leftarrow CY \oplus A.位$	×
		CY, PSW.位	3	—	7	$CY \leftarrow CY \oplus PSW.位$	×
		CY, [HL].位	2	6	7	$CY \leftarrow CY \oplus (HL).位$	×
	SET1	saddr.位	2	4	6	$(saddr.位) \leftarrow 1$	
		sfr.位	3	—	8	$sfr.位 \leftarrow 1$	
		A.位	2	4	—	$A.位 \leftarrow 1$	
		PSW.位	2	—	6	$PSW.位 \leftarrow 1$	× × ×
		[HL].位	2	6	8	$(HL).位 \leftarrow 1$	
	CLR1	saddr.位	2	4	6	$(saddr.位) \leftarrow 0$	
		sfr.位	3	—	8	$sfr.位 \leftarrow 0$	
		A.位	2	4	—	$A.位 \leftarrow 0$	
		PSW.位	2	—	6	$PSW.位 \leftarrow 0$	× × ×
		[HL].位	2	6	8	$(HL).位 \leftarrow 0$	
	SET1	CY	1	2	—	$CY \leftarrow 1$	1
	CLR1	CY	1	2	—	$CY \leftarrow 0$	0
	NOT1	CY	1	2	—	$CY \leftarrow \overline{CY}$	×

- 注 1. 访问内部高速 RAM 时或没有无数据访问的指令
 2. 访问内部高速 RAM 以外的区域时

- 备注 1. 一个指令时钟周期是指由处理器时钟控制寄存器(PCC)选择的 CPU 时钟(f_{CPU})的一个周期。
 2. 该时钟周期用于内部 ROM 程序。

指令组	助记符	操作数	位	时钟		操作	标志 Z AC CY
				注 1	注 2		
调用/返回	CALL	!addr16	3	7	—	$(SP - 1) \leftarrow (PC + 3)_H, (SP - 2) \leftarrow (PC + 3)_L,$ $PC \leftarrow \text{addr16}, SP \leftarrow SP - 2$	
	CALLF	!addr11	2	5	—	$(SP - 1) \leftarrow (PC + 2)_H, (SP - 2) \leftarrow (PC + 2)_L,$ $PC_{15-11} \leftarrow 00001, PC_{10-0} \leftarrow \text{addr11},$ $SP \leftarrow SP - 2$	
	CALLT	[addr5]	1	6	—	$(SP - 1) \leftarrow (PC + 1)_H, (SP - 2) \leftarrow (PC + 1)_L,$ $PC_H \leftarrow (\text{addr5} + 1), PC_L \leftarrow (\text{addr5}),$ $SP \leftarrow SP - 2$	
	BRK		1	6	—	$(SP - 1) \leftarrow PSW, (SP - 2) \leftarrow (PC + 1)_H,$ $(SP - 3) \leftarrow (PC + 1)_L, PC_H \leftarrow (003FH),$ $PC_L \leftarrow (003EH), SP \leftarrow SP - 3, IE \leftarrow 0$	
	RET		1	6	—	$PC_H \leftarrow (SP + 1), PC_L \leftarrow (SP),$ $SP \leftarrow SP + 2$	
	RETI		1	6	—	$PC_H \leftarrow (SP + 1), PC_L \leftarrow (SP),$ $PSW \leftarrow (SP + 2), SP \leftarrow SP + 3$	R R R
	RETB		1	6	—	$PC_H \leftarrow (SP + 1), PC_L \leftarrow (SP),$ $PSW \leftarrow (SP + 2), SP \leftarrow SP + 3$	R R R
堆栈操作	PUSH	PSW	1	2	—	$(SP - 1) \leftarrow PSW, SP \leftarrow SP - 1$	
		rp	1	4	—	$(SP - 1) \leftarrow rp_H, (SP - 2) \leftarrow rp_L,$ $SP \leftarrow SP - 2$	
	POP	PSW	1	2	—	$PSW \leftarrow (SP), SP \leftarrow SP + 1$	R R R
		rp	1	4	—	$rp_H \leftarrow (SP + 1), rp_L \leftarrow (SP),$ $SP \leftarrow SP + 2$	
	MOVW	SP, #字	4	—	10	$SP \leftarrow \text{字}$	
		SP, AX	2	—	8	$SP \leftarrow AX$	
		AX, SP	2	—	8	$AX \leftarrow SP$	
无条件转移	BR	!addr16	3	6	—	$PC \leftarrow \text{addr16}$	
		\$addr16	2	6	—	$PC \leftarrow PC + 2 + \text{jdisp8}$	
		AX	2	8	—	$PC_H \leftarrow A, PC_L \leftarrow X$	
条件转移	BC	\$addr16	2	6	—	$PC \leftarrow PC + 2 + \text{jdisp8}$ if CY = 1	
	BNC	\$addr16	2	6	—	$PC \leftarrow PC + 2 + \text{jdisp8}$ if CY = 0	
	BZ	\$addr16	2	6	—	$PC \leftarrow PC + 2 + \text{jdisp8}$ if Z = 1	
	BNZ	\$addr16	2	6	—	$PC \leftarrow PC + 2 + \text{jdisp8}$ if Z = 0	

- 注 1. 访问内部高速 RAM 时或没有无数据访问的指令
2. 访问内部高速 RAM 以外的区域时

- 备注 1. 一个指令时钟周期是指由处理器时钟控制寄存器(PCC)选择的 CPU 时钟(f_{CPU})的一个周期。
2. 该时钟周期用于内部 ROM 程序。

指令组	助记符	操作数	位	时钟		操作	标志
				注 1	注 2		Z AC CY
条件转移	BT	saddr.位, \$addr16	3	8	9	PC ← PC + 3 + jdisp8 if (saddr.位) = 1	
		sfr.位, \$addr16	4	–	11	PC ← PC + 4 + jdisp8 if sfr.位 = 1	
		A.位, \$addr16	3	8	–	PC ← PC + 3 + jdisp8 if A.位 = 1	
		PSW.位, \$addr16	3	–	9	PC ← PC + 3 + jdisp8 if PSW.位 = 1	
		[HL].位, \$addr16	3	10	11	PC ← PC + 3 + jdisp8 if (HL).位 = 1	
	BF	saddr.位, \$addr16	4	10	11	PC ← PC + 4 + jdisp8 if (saddr.位) = 0	
		sfr.位, \$addr16	4	–	11	PC ← PC + 4 + jdisp8 if sfr.位 = 0	
		A.位, \$addr16	3	8	–	PC ← PC + 3 + jdisp8 if A.位 = 0	
		PSW.位, \$addr16	4	–	11	PC ← PC + 4 + jdisp8 if PSW.位 = 0	
		[HL].位, \$addr16	3	10	11	PC ← PC + 3 + jdisp8 if (HL).位 = 0	
	BTCLR	saddr.位, \$addr16	4	10	12	PC ← PC + 4 + jdisp8 if (saddr.位) = 1 then reset (saddr.位)	
		sfr.位, \$addr16	4	–	12	PC ← PC + 4 + jdisp8 if sfr.位 = 1 then reset sfr.位	
		A.位, \$addr16	3	8	–	PC ← PC + 3 + jdisp8 if A.位 = 1 then reset A.位	
		PSW.位, \$addr16	4	–	12	PC ← PC + 4 + jdisp8 if PSW.位 = 1 then reset PSW.位	× × ×
		[HL].位, \$addr16	3	10	12	PC ← PC + 3 + jdisp8 if (HL).位 = 1 then reset (HL).位	
	DBNZ	B, \$addr16	2	6	–	B ← B – 1, 然后 PC ← PC + 2 + jdisp8 if B ≠ 0	
		C, \$addr16	2	6	–	C ← C – 1, 然后 PC ← PC + 2 + jdisp8 if C ≠ 0	
		saddr, \$addr16	3	8	10	(saddr) ← (saddr) – 1, 然后 PC ← PC + 3 + jdisp8 if (saddr) ≠ 0	
CPU 控制	SEL	RBn	2	4	–	RBS1, 0 ← n	
	NOP		1	2	–	无操作	
	EI		2	–	6	IE ← 1 (允许中断)	
	DI		2	–	6	IE ← 0 (禁止中断)	
	HALT		2	6	–	设置 HALT 模式	
	STOP		2	6	–	设置 STOP 模式	

注 1. 访问内部高速 RAM 时或没有无数据访问的指令

2. 访问内部高速 RAM 以外的区域时

备注 1. 一个指令时钟周期是指由处理器时钟控制寄存器(PCC)选择的 CPU 时钟(fCPU)的一个周期。

2. 该时钟周期用于内部 ROM 程序。

23.3 按寻址分类列出的指令

(1) 8 位指令

MOV, XCH, ADD, ADDC, SUB, SUBC, AND, OR, XOR, CMP, MULU, DIVUW, INC, DEC, ROR, ROL, RORC, ROLC, ROR4, ROL4, PUSH, POP, DBNZ

第二操作数 第一操作数	#字节	A	r*	sfr	saddr	!addr16	PSW	[DE]	[HL]	[HL + 字节] [HL + B] [HL + C]	\$addr16	1	无
A	ADD ADDC SUB SUBC AND OR XOR CMP		MOV XCH ADD ADDC SUB SUBC AND OR XOR CMP	MOV XCH	MOV XCH ADD ADDC SUB SUBC AND OR XOR CMP	MOV XCH ADD ADDC SUB SUBC AND OR XOR CMP	MOV	MOV XCH	MOV XCH ADD ADDC SUB SUBC AND OR XOR CMP	MOV XCH ADD ADDC SUB SUBC AND OR XOR CMP		ROR ROL RORC ROLC	
r	MOV	MOV ADD ADDC SUB SUBC AND OR XOR CMP											INC DEC
B, C											DBNZ		
sfr	MOV	MOV											
saddr	MOV ADD ADDC SUB SUBC AND OR XOR CMP	MOV									DBNZ		INC DEC
!addr16		MOV											
PSW	MOV	MOV											PUSH POP
[DE]		MOV											
[HL]		MOV											ROR4 ROL4
[HL + 字节] [HL + B] [HL + C]		MOV											
X													MULU
C													DIVUW

注 “r = A” 除外

(2) 16 位指令

MOVW, XCHW, ADDW, SUBW, CMPW, PUSH, POP, INCW, DECW

第二操作数 第一操作数	#字	AX	rp [※]	sfrp	saddrp	!addr16	SP	无
AX	ADDW SUBW CMPW		MOVW XCHW	MOVW	MOVW	MOVW	MOVW	
rp	MOVW	MOVW [※]						INCW DECW PUSH POP
sfrp	MOVW	MOVW						
saddrp	MOVW	MOVW						
!addr16		MOVW						
SP	MOVW	MOVW						

注 仅在 rp = BC, DE, HL 时

(3) 位操作指令

MOV1, AND1, OR1, XOR1, SET1, CLR1, NOT1, BT, BF, BTCLR

第二操作数 第一操作数	A.位	sfr.位	saddr.位	PSW.位	[HL].位	CY	\$addr16	无
A.位						MOV1	BT BF BTCLR	SET1 CLR1
sfr.位						MOV1	BT BF BTCLR	SET1 CLR1
saddr.位						MOV1	BT BF BTCLR	SET1 CLR1
PSW.位						MOV1	BT BF BTCLR	SET1 CLR1
[HL].位						MOV1	BT BF BTCLR	SET1 CLR1
CY	MOV1 AND1 OR1 XOR1	MOV1 AND1 OR1 XOR1	MOV1 AND1 OR1 XOR1	MOV1 AND1 OR1 XOR1	MOV1 AND1 OR1 XOR1			SET1 CLR1 NOT1

(4) 调用指令/转移指令

CALL, CALLF, CALLT, BR, BC, BNC, BZ, BNZ, BT, BF, BTCLR, DBNZ

第二操作数 第一操作数	AX	!addr16	!addr11	[addr5]	\$addr16
基本指令	BR	CALL BR	CALLF	CALLT	BR BC BNC BZ BNZ
复合指令					BT BF BTCLR DBNZ

(5) 其他指令

ADJBA, ADJBS, BRK, RET, RETI, RETB, SEL, NOP, EI, DI, HALT, STOP

第二十四章 电特性

注意事项 μPD79F7023, 79F7024 微控制器具有开发/评估选择的 On-chip 调试功能。使用 On-chip 调试功能时，可能会出现超过闪存所定重写次数的现象，从而不能保证产品的可靠性，因此，不要将本功能用于量产化的产品。瑞萨电子公司不接受任何与 On-chip 调试功能相关产品的投诉。

最大绝对额定值(T_A = 25°C) (1/2)

项目	符号	条件	额定值	单位
供电电压	V _{DD}		-0.5 至 +6.5	V
	V _{SS}		-0.5 至 +0.3	V
	AV _{REF}		-0.5 至 V _{DD} + 0.3 ^{注1}	V
REGC引脚输入电压 ^{注2}	VI _{REGC}		-0.5 至 +3.6 和 -0.5 至 V _{DD} + 0.3	V
输入电压	VI1	P30 至 P34, P121, P122, P125, X1, X2, RESET	-0.3 至 V _{DD} + 0.3 ^{注1}	V
	VI2	P20 至 P27	-0.3 至 AV _{REF} + 0.3 ^{注1} 和 -0.3 至 V _{DD} + 0.3 ^{注1}	V
输出电压	VO1	P30 至 P34, P121, P122	-0.3 至 V _{DD} + 0.3 ^{注1}	V
	VO2	P20 至 P27	-0.3 至 AV _{REF} + 0.3 ^{注1}	V

注 1. 必须等于或低于 6.5V。

2. 通过电容器，连接 REGC 引脚至 V_{SS}(0.47 至 1 μF)。该值规定 REGC 引脚的最大绝对额定值。禁止使用外加电压。

注意事项 即使任何一项参数在瞬间超过最大绝对额定值，都会影响产品质量。也就是说，最大绝对额定值是产品濒临物理损坏边缘时的临界点，因此，必须确保在不超过最大绝对额定值的条件下使用本产品。

备注 在没有特别指定的情况下，复用功能引脚的特性与端口引脚的特性相同。

最大绝对额定值(T_A = 25°C) (2/2)

项目	符号	条件		额定值	单位
高电平输出电流	I _{OH1}	1 个引脚	P31 至 P34	-10	mA
		全部引脚总和		-25	mA
	I _{OH2}	1 个引脚	P30	-10	mA
		全部引脚总和		-10	mA
	I _{OH3}	1 个引脚	P20 至 P27	-0.5	mA
		全部引脚总和		-2	mA
	I _{OH4}	1 个引脚	P121, P122	-1	mA
		全部引脚总和		-4	mA
低电平输出电流	I _{OL1}	1 个引脚	P31 至 P34	30	mA
		全部引脚总和		55	mA
	I _{OL2}	1 个引脚	P30	30	mA
		全部引脚总和		30	mA
	I _{OL3}	1 个引脚	P20 至 P27	1	mA
		全部引脚总和		5	mA
	I _{OL4}	1 个引脚	P121, P122	4	mA
		全部引脚总和		10	mA
工作环境温度	T _A			-40 至 +85	°C
存储温度	T _{stg}			-65 至 +150	°C

- 注意事项**
1. 即使任何一项参数在瞬间超过最大绝对额定值，都会影响产品质量。也就是说，最大绝对额定值是产品濒临物理损坏边缘时的临界点，因此，必须确保在不超过最大绝对额定值的条件下使用本产品。
 2. 流经每个引脚的电流值必须符合每个引脚的电流值和所有引脚的电流总值。

备注 在没有特别指定的情况下，复用功能引脚的特性与端口引脚的特性相同。

X1 振荡器特性

($T_A = -40$ 至 $+85^{\circ}\text{C}$, $2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $V_{SS} = 0\text{ V}$)

谐振器	推荐电路	项目	条件	最小值	典型值	最大值	单位
陶瓷谐振器 /晶体谐振器		X1 时钟振荡频率 (f_x) [※]	$2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}$	1.0		10.0	MHz

注 仅可显示振荡器特性。指令执行时间参阅 **AC 特性**。

注意事项 1. 使用 X1 振荡器时，按照下面的要求在上图中虚线框起来的区域进行布线，从而避免布线电容的副作用。

- 布线要保证尽可能要短。
- 振荡器接线不要和其它的信号线互相交叉。
- 禁止将振荡器的布线靠近大波动电流经由的信号线。
- 始终保持振荡器电容的接地点与 **VSS** 相同。
禁止将电容的接地点连接在大电流经由的接地模式。
- 禁止从振荡器获取信号。

2. 复位解除后，使用高速振荡时钟启动 **CPU**，因此，用户应通过振荡稳定时间时钟状态寄存器(**OSTC**)确认 X1 时钟的振荡稳定时间。或者使用谐振器充分评估振荡稳定时间后，确定 **OSTC** 寄存器、振荡稳定时间选择寄存器(**OSTS**)的振荡稳定时间。

备注 有关谐振器的选择和振荡电路常数，需要用户自行评估振荡特性或者委托谐振器生产商提供评估。

内部高速振荡器特性

(T_A = -40 至 +85°C, 2.7 V ≤ V_{DD} ≤ 5.5 V, V_{SS} = 0 V)

谐振器	项目	条件		最小值	典型值	最大值	单位
内部高速振荡器	振荡频率 (f _{1H} = 4 MHz) [※]	RSTS = 1	T _A = -20 至 +70°C	3.92	4	4.08	MHz
			T _A = -40 至 +85°C	3.88	4	4.12	MHz

注 只能显示振荡器特性。指令执行时间参阅 AC 特性。

内部低速振荡器特性

(T_A = -40 至 +85°C, 2.7 V ≤ V_{DD} ≤ 5.5 V, V_{SS} = 0 V)

谐振器	项目	条件	最小值	典型值	最大值	单位
内部低速振荡器	振荡频率 f _{1L})	CREG = 2.5 V 模式	216	240	264	kHz
		CREG = 1.9 V 模式	192	240	264	kHz

DC 特性(1/5)

(T_A = -40 至 +85°C, 2.7 V ≤ V_{DD} ≤ 5.5 V, AV_{REF} ≤ V_{DD}, V_{SS} = 0 V)

项目	符号	条件	最小值	典型值	最大值	单位
输出电流, 高 ^{注1}	I _{OH1}	P31 至 P34 的 1 个引脚	4.0 V ≤ V _{DD} ≤ 5.5 V		-3.0	mA
			2.7 V ≤ V _{DD} < 4.0 V		-2.5	mA
		P31 至 P34 ^{注3} 之总和。	4.0 V ≤ V _{DD} ≤ 5.5 V		-6.0	mA
			2.7 V ≤ V _{DD} < 4.0 V		-4.5	mA
	I _{OH2}	P30 的 1 个引脚	4.0 V ≤ V _{DD} ≤ 5.5 V		-3.0	mA
			2.7 V ≤ V _{DD} < 4.0 V		-2.5	mA
	I _{OH3}	P20 至 P27 的 1 个引脚	AV _{REF} = V _{DD}		-100	μA
		P121 和 P122 的 1 个引脚			-100	μA
输出电流, 低 ^{注2}	I _{OL1}	P31 至 P34 的 1 个引脚	4.0 V ≤ V _{DD} ≤ 5.5 V		8.5	mA
			2.7 V ≤ V _{DD} < 4.0 V		5.0	mA
		P31 至 P34 ^{注3} 之总和。	4.0 V ≤ V _{DD} ≤ 5.5 V		15.0	mA
			2.7 V ≤ V _{DD} < 4.0 V		10.0	mA
	I _{OH2}	P30 的 1 个引脚	4.0 V ≤ V _{DD} ≤ 5.5 V		8.5	mA
			2.7 V ≤ V _{DD} < 4.0 V		5.0	mA
	I _{OH3}	P122 的 1 个引脚	4.0 V ≤ V _{DD} ≤ 5.5 V		8.5	mA
			2.7 V ≤ V _{DD} < 4.0 V		5.0	mA
	I _{OL4}	P20 至 P27 的 1 个引脚	AV _{REF} = V _{DD}		400	μA
		P121 的 1 个引脚			400	μA

- 注 1. 即使电流从 V_{DD} 流向输出引脚, 也能保证器件操作时的电流值。
2. 即使电流从输出引脚流向 GND, 也能保证器件操作时的电流值。
3. 占空比为 70% 条件 (指定时间为 t, 输出电流时间为 0.7 × t, 不输出电流时间为 0.3 × t 以下的规格。通过以下公式可以计算占空比不等于 70% 时的引脚总和的输出电流。
- I_{OH} 占空比为 n%: 引脚的总输出电流 = (I_{OH} × 0.7)/(n × 0.01)
- <示例> 假设占空比为 50%、I_{OH} = -20.0 mA,
- 则 引脚的总输出电流 = (-20.0 × 0.7)/(50 × 0.01) = -28.0 mA
- 但是, 允许流入每个引脚的电流不随占空比的变化而变化。高于最大绝对额定值的电流不流入引脚。

备注 在没有特别指定的情况下, 复用功能引脚的特性与端口引脚的特性相同。

DC 特性(2/5)

($T_A = -40$ 至 $+85^{\circ}\text{C}$, $2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $AV_{REF} \leq V_{DD}$, $V_{SS} = 0\text{ V}$)

项目	符号	条件		最小值	典型值	最大值	单位
高电平输入电压	V_{IH1}	P122 [※] , P31		$0.7V_{DD}$		V_{DD}	V
	V_{IH2}	P20 至 P27	$AV_{REF} = V_{DD}$	$0.7AV_{REF}$		AV_{REF}	V
	V_{IH3}	P121, P125		$0.7V_{DD}$		V_{DD}	V
	V_{IH4}	P30, P32 至 P34, $\overline{\text{RESET}}$, EXCLK		$0.8V_{DD}$		V_{DD}	V
	V_{IH5}	X1, X2		$V_{DD} - 0.1$		V_{DD}	V
低电平输入电压	V_{IL1}	P122 [※] , P31		0		$0.3V_{DD}$	V
	V_{IL2}	P20 至 P27	$AV_{REF} = V_{DD}$	0		$0.3AV_{REF}$	V
	V_{IL3}	P121, P125		0		$0.3V_{DD}$	V
	V_{IL4}	P30, P32 至 P34, $\overline{\text{RESET}}$, EXCLK		0		$0.2V_{DD}$	V
	V_{IL5}	X1, X2		0		0.1	V
高电平输出电压	V_{OH1}	P30 至 P34	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $I_{OH1} = -3.0\text{ mA}$	$V_{DD} - 0.7$			V
			$2.7\text{ V} \leq V_{DD} < 4.0\text{ V}$, $I_{OH1} = -2.5\text{ mA}$	$V_{DD} - 0.5$			V
	V_{OH2}	P20 至 P27	$AV_{REF} = V_{DD}$, $I_{OH2} = -100\text{ }\mu\text{A}$	$V_{DD} - 0.5$			V
		P121, P122	$I_{OH2} = -100\text{ }\mu\text{A}$	$V_{DD} - 0.5$			V
低电平输出电压	V_{OH1}	P30 至 P34, P122	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $I_{OL1} = 8.5\text{ mA}$			0.8	V
			$2.7\text{ V} \leq V_{DD} < 4.0\text{ V}$, $I_{OL1} = 5.0\text{ mA}$			0.7	V
	V_{OH2}	P20 至 P27	$AV_{REF} = V_{DD}$, $I_{OL2} = 400\text{ }\mu\text{A}$			0.4	V
		P121	$I_{OL2} = 400\text{ }\mu\text{A}$			0.4	V

注 输入端口模式和外部时钟模式下, P122/EXCLK 的 V_{IH}/V_{IL} 不同。

备注 在没有特别指定的情况下, 复用功能引脚的特性与端口引脚的特性相同。

DC 特性(3/5)**($T_A = -40$ 至 $+85^{\circ}\text{C}$, $2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $AV_{REF} \leq V_{DD}$, $V_{SS} = 0\text{ V}$)**

项目	符号	条件		最小值	典型值	最大值	单位	
高电平输入漏电流	I _{LIH1}	P30 至 P34, P125/ $\overline{\text{RESET}}$		V _I = V _{DD}		3	μA	
	I _{LIH2}	P20 至 P27		V _I = AV _{REF} = V _{DD}		3	μA	
	I _{LIH3}	P121, P122	V _I = V _{DD}	I/O 端口模式		3	μA	
		X1, X2		OSC 模式		20	μA	
低电平输入漏电流	I _{LIL1}	P30 至 P34, P125/ $\overline{\text{RESET}}$		V _I = V _{SS}		−3	μA	
	I _{LIL2}	P20 至 P27		V _I = V _{SS} , AV _{REF} = V _{DD}		−3	μA	
	I _{LIL3}	P121, P122	V _I = V _{SS}	I/O 端口模式		−3	μA	
		X1, X2		OSC 模式		−20	μA	
上拉电阻值	R _{PLU1}	P30 至 P34		V _I = V _{SS}	10	20	100	kΩ
	R _{PLU2}	P125/ $\overline{\text{RESET}}$			75	150	300	kΩ

备注 在没有特别指定的情况下，复用功能引脚的特性与端口引脚的特性相同。

DC 特性(4/5)

(T_A = -40 至 +85°C, 2.7 V ≤ V_{DD} ≤ 5.5 V, AV_{REF} ≤ V_{DD}, V_{SS} = 0 V)

项目	符号	条件			最小值	典型值	最大值	单位
供应电流 ^注	I _{DD1}	操作模式	f _{XH} = 10 MHz, RMC = 00H	输入方波		1.1	2.7	mA
				接谐振器连		1.3	3.2	mA
			f _{XH} = 5 MHz, RMC = 00H	输入方波		0.63	1.6	mA
				连接谐振器		0.8	2.0	mA
			f _{IH} = 4 MHz, RMC = 59H	仅限 CPU 操作		0.37	0.74	mA
				CPU 和外围操作		0.42	0.84	mA
			f _{IH} = 4 MHz, f _{CPU} = 1 MHz V _{DD} = 3.0 V, RMC = 56H	仅限 CPU 操作		0.14	0.25	mA
				CPU 和外围操作		0.18	0.36	mA
	I _{DD2}	HALT 模式	f _{XH} = 10 MHz, RMC = 00H	输入方波		0.13	1.3	mA
				连接谐振器		0.36	2.4	mA
			f _{XH} = 5 MHz, RMC = 00H	输入方波		0.09	0.65	mA
				连接谐振器		0.27	1.1	mA
			f _{IH} = 4 MHz, RMC = 59H			0.12	0.12	0.5
	I _{DD3}	STOP 模式	V _{DD} = 5.0 V, 仅限 CREG+POC	RMC = 00H		1.2	10	μA
				RMC = 56H/59H		2	10	μA

注 流经内部电源(V_{DD}, AV_{REF})的总电流, 包括输入引脚恒为 V_{DD} 或 V_{SS} 状态下的输入泄漏电流。但不包括端口的上拉电阻、下拉电阻和输出电流。典型值是指 V_{DD} = 3.0 V 并且 CPU 运行时的电流。最大值包括外围操作电流。但是, ADC、WWDT、LVI、AMP 和 CMP 停止。

备注 在没有特别指定的情况下, 复用功能引脚的特性与端口引脚的特性相同。

DC 特性(5/5)

(T_A = -40 至 +85°C, 2.7 V ≤ V_{DD} ≤ 5.5 V, AV_{REF} ≤ V_{DD}, V_{SS} = 0 V)

项目	符号	条件		最小值	典型值	最大值	单位
看门狗定时器工作电流 ^{注1}	I _{WDT}	V _{DD} = 5.0 V	240kHz 内部低速振荡时钟操作时		5	10	μA
LVI工作电流 ^{注2}	I _{LVI}				9	18	μA
A/D转换器工作电流 ^{注3}	I _{ADC}				0.86	1.9	mA
运算放大器工作电流 ^{注3}	I _{AMP}	备有 1 运算放大器操作	AV _{REF} = V _{DD} = 5.0 V		250	380	μA
			AV _{REF} = V _{DD} = 3.0 V		230	321	μA
比较器工作电流 ^{注3}	I _{CMP}	AV _{REF} = V _{DD} = 5.0 V			80	T.B.D	μA
		AV _{REF} = V _{DD} = 3.0 V			70	T.B.D	μA

注 1. 只流经看门狗定时器的电流（包括 240kHz 内部振荡器的操作电流）。看门狗定时器运行时，μPD79F7023，79F7024 微控制器的电流值等于 I_{DD1}、I_{DD2} 或 I_{DD3} 加上 I_{WDT} 的值。

2. 只流经率 LVI 电路的电流。LVI 电路擦操作时，μPD79F7023，79F7024 微控制器的电流值等于 I_{DD1}、I_{DD2} 或 I_{DD3} 和 I_{LVI} 加上 LVI 的值。

3. 只流经 A/D 转换器(AV_{REF})的电流。

AC 特性

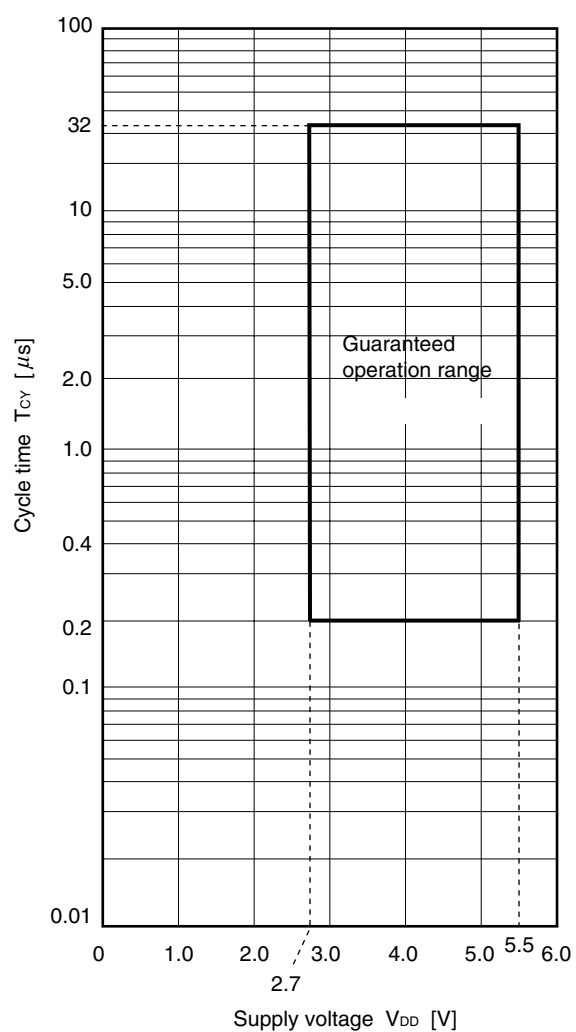
(1) 基本操作

($T_A = -40$ 至 $+85^{\circ}\text{C}$, $2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $V_{SS} = 0\text{ V}$)

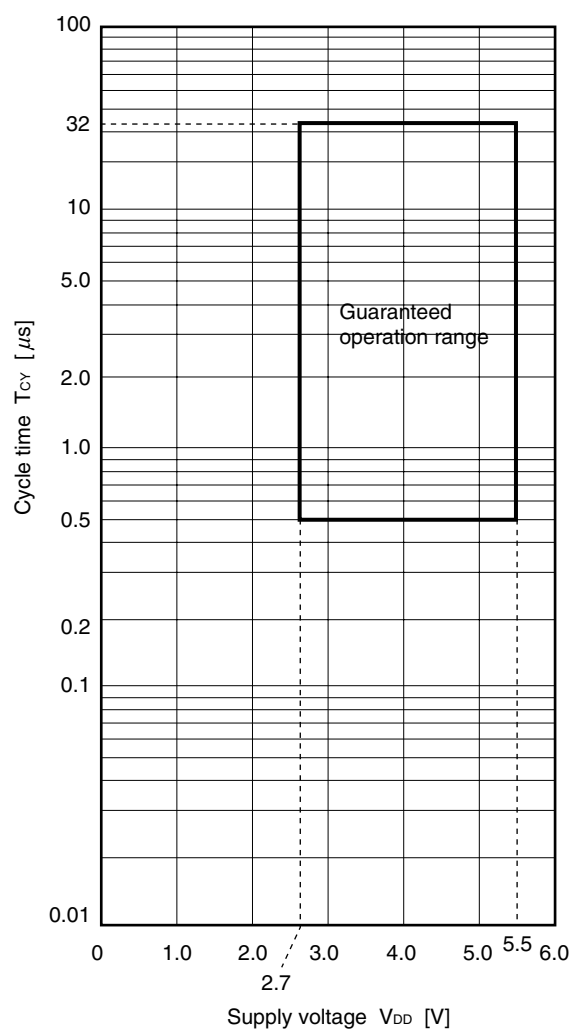
项目	符号	条件	最小值	典型值	最大值	单位
指令周期（最小指令执行时间）	T_{CY}	主系统时钟(f_{XP})操作	CREG = 2.5 V 模式	0.2	32	μs
			CREG = 1.9 V 模式	0.5	32	μs
外部硬件时钟频率	f_{PRS}	$f_{PRS} = f_{XP}$			10	MHz
		$f_{PRS} = f_{IH}$	3.8		4.2	MHz
外部主系统时钟频率	f_{EXCLK}	$2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}$	1.0		10.0	MHz
外部主系统时钟输入高电平宽度, 低电平宽度	t_{EXCLKH} , t_{EXCLKL}		$(1/f_{EXCLK} \times 1/2) - 1$			ns
TI000, TI010 输入高电平宽度, 低电平宽度	t_{TIH0} , t_{TIL0}	$2.7\text{ V} \leq V_{DD} < 4.0\text{ V}$	$2/f_{sam} + 0.2^{\#}$			μs
TI51 输入频率	f_{TI5}	$2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}$			10.0	MHz
TI51 输入高电平宽度, 低电平宽度	t_{TIH5}	$2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}$	50			ns
中断输入高电平宽度, 低电平宽度	t_{INTH} , t_{INTL}		1			μs
RESET 低电平宽度	t_{RSL}		10			μs

注 通过预分频器模式寄存器 00(PRM00)的第 0 位和第 1 位(PRM000, PRM001), 可以选择 $f_{sam} = f_{PRS}$ 、 $f_{PRS}/4$ 、 $f_{PRS}/256$ 。应注意, 若选择 TI000 有效沿用作计数时钟时, 则 $f_{sam} = f_{PRS}$ 。

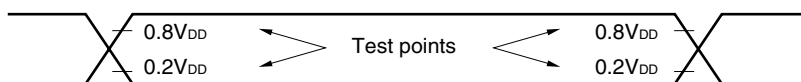
T_{CY} vs. V_{DD} (主系统时钟操作, CREG = 2.5 V 模式)



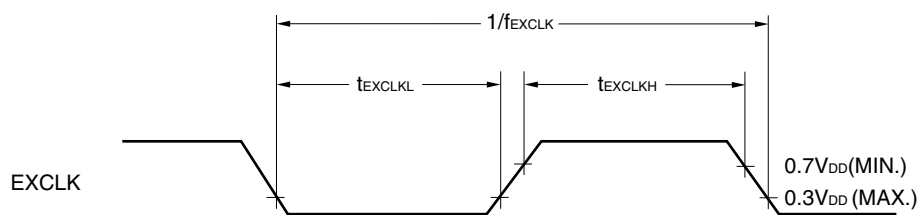
T_{CY} vs. V_{DD} (主系统时钟操作, CREG = 1.9 V 模式)



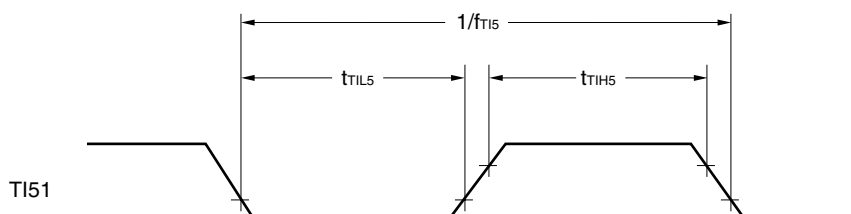
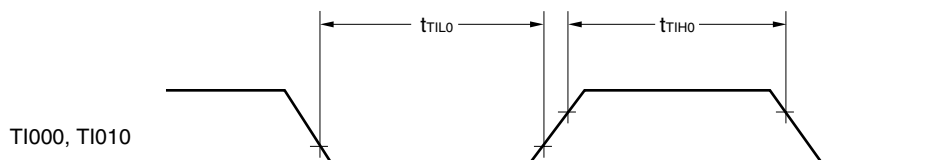
AC 调整测试点



外部主系统时钟调整



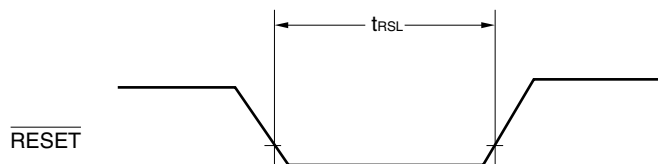
TI 调整



中断请求输入时序



RESET 输入调整



(2) 串行接口

($T_A = -40$ 至 $+85^{\circ}\text{C}$, $2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $V_{SS} = 0\text{ V}$)

(a) UART0 (专用波特率发生器输出)

项目	符号	条件	最小值	典型值	最大值	单位
传送速率:					312.5	kbps

(b) OCD (UART0)

项目	符号	条件	最小值	典型值	最大值	单位
传送速率:			$f_{CLK}/32$		$f_{CLK}/8$	bps
		OCD 模式下 ($f_{CLK} = 4\text{ M}$, $V_{DD} \geq 2.7\text{ V}$, $C_b = 50\text{ pF}$)	125/250/500			kbps
		写入模式下 ($f_{CLK} = 4\text{ M}$, $V_{DD} \geq 2.7\text{ V}$, $C_b = 50\text{ pF}$)	125/250/500			kbps
X2/输出频率	f_{OCDB}	$V_{DD} \geq 2.7\text{ V}$			10	MHz

模式特性

(1) A/D 转换器

($T_A = -40$ 至 $+85^{\circ}\text{C}$, $2.7\text{ V} \leq AV_{REF} \leq V_{DD} \leq 5.5\text{ V}$, $V_{SS} = 0\text{ V}$)

项目	符号	条件	最小值	典型值	最大值	单位
分辨率	R_{ES}				8	bit
总误差 ^{注1,2}	A_{INL}	$4.0\text{ V} \leq AV_{REF} \leq 5.5\text{ V}$			± 0.6	%FSR
		$2.7\text{ V} \leq AV_{REF} \leq 5.5\text{ V}$			± 0.8	%FSR
转换时间	t_{CONV}	$4.0\text{ V} \leq AV_{REF} \leq 5.5\text{ V}$	6.1		36.7	μs
		$2.7\text{ V} \leq AV_{REF} \leq 5.5\text{ V}$	6.1		36.7	μs
零误差 ^{注1,2}	E_{ZS}	$4.0\text{ V} \leq AV_{REF} \leq 5.5\text{ V}$			± 0.6	%FSR
		$2.7\text{ V} \leq AV_{REF} \leq 5.5\text{ V}$			± 0.8	%FSR
满度误差 ^{注1,2}	E_{FS}	$4.0\text{ V} \leq AV_{REF} \leq 5.5\text{ V}$			± 0.6	%FSR
		$2.7\text{ V} \leq AV_{REF} \leq 5.5\text{ V}$			± 0.8	%FSR
模拟输入电压	V_{AIN}		V_{SS}		AV_{REF}	V

注 1. 不包括量化误差 ($\pm 1/2$ LSB)。

2. 表示满度值的对应百分比(%FSR)。

(2) 运算放大器 0, 1

($T_A = -40$ 至 $+85^{\circ}\text{C}$, $2.7\text{ V} \leq V_{DD} \leq AV_{REF} \leq 5.5\text{ V}$, $V_{SS} = 0\text{ V}$, 输出负载: $R_L = 47\text{ k}\Omega$, $C_L = 50\text{ pF}$)

项目	符号	条件	最小值	典型值	最大值	单位
输入偏移电压 ^{注1}	V_{IOPO}	$V_{BIAS} = 1/2 V_{DD}$, $AV_{REF} = 3.0\text{ V}$			± 3	mV
电源电压抑制比	$PSRR_{OP0}$	$AV_{REF} = 3.0\text{ V}$		70		dB
高电平输出电压	V_{OHOP0}	$AV_{REF} = 3.0\text{ V}$, $I_{OH} = -500\text{ }\mu\text{A}$	$AV_{REF}-0.2$			V
低电平输出电压	V_{OLOP0}	$AV_{REF} = 3.0\text{ V}$, $I_{OL} = 500\text{ }\mu\text{A}$			0.1	V
共模输入电压	V_{ICMOP0}	$AV_{REF} = 3.0\text{ V}$	0		$AV_{REF}-0.6$	V
压摆率值	SR_{OP0}	$AV_{REF} = 3.0\text{ V}$		1.8		V/ μs
		$AV_{REF} = 5.0\text{ V}$		2.0		V/ μs
输入噪声密度 (Inoise)		$AV_{REF} = 3.0\text{ V}$, $V_{IN} = 0.1\text{ V}$, $f = 1\text{ kHz}$		73		nV/ $\sqrt{\text{Hz}}$
		$AV_{REF} = 3.0\text{ V}$, $V_{IN} = AV_{REF}/2$, $f = 1\text{ kHz}$		60		
		$AV_{REF} = 3.0\text{ V}$, $V_{IN} = AV_{REF} - 0.6\text{ V}$, $f = 1\text{ kHz}$		55		
相位裕度		$AV_{REF} = 3.0\text{ V}$		40		deg
大振幅电压增益	AV_{OP0}	$AV_{REF} = 3.0\text{ V}$		100		dB
增益带宽积	GBW_{OP0}	$AV_{REF} = 5.0\text{ V}/3.0\text{ V}$		3.0		MHz
操作稳定等待时间 ^{注2}	t_{OP0}	$AV_{REF} = 3.0\text{ V}$		10		μs

注 1. 特性评估的结果。没有做出货检验。

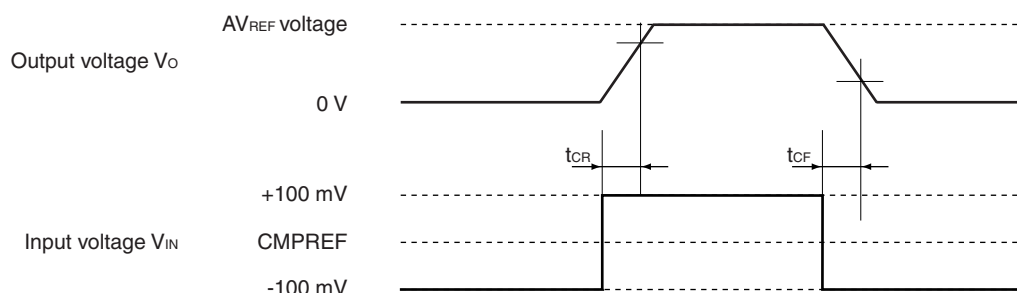
2. 允许放大器操作(OPAMP0E/OPAMP1E = 1)后, 为了满足放大器操作的 DC 和 AC 特性状态所需的时间。

(3) CMP

($T_A = -40$ 至 $+85^{\circ}\text{C}$, $2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $V_{SS} = 0\text{ V}$, 输出负载: $R_L = 47\text{ k}\Omega$, $C_L = 50\text{ pF}$)

项目	符号	条件	最小值	典型值	最大值	单位
输入偏移电压	V_{IOCOMP}			± 5	± 40	mV
输入电压范围	V_{ICMP}	CMPIN	0		AV_{REF}	V
		CMPCOM	0.045		$0.9AV_{REF}$	V
响应时间	t_{CR} , t_{CF}	输入振幅 $\pm 100\text{ mV}$		70	150	ns
操作稳定等待时间 [‡]	t_{CMP}				1	μs
比较器输入高电平宽度, 低电平宽度	t_{CMPL}		125			ns

注 允许比较器操作(OPAMP0E/OPAMP1E = 1)后, 比较器达到符合 DC 和 AC 特性的状态为止所需的时间。

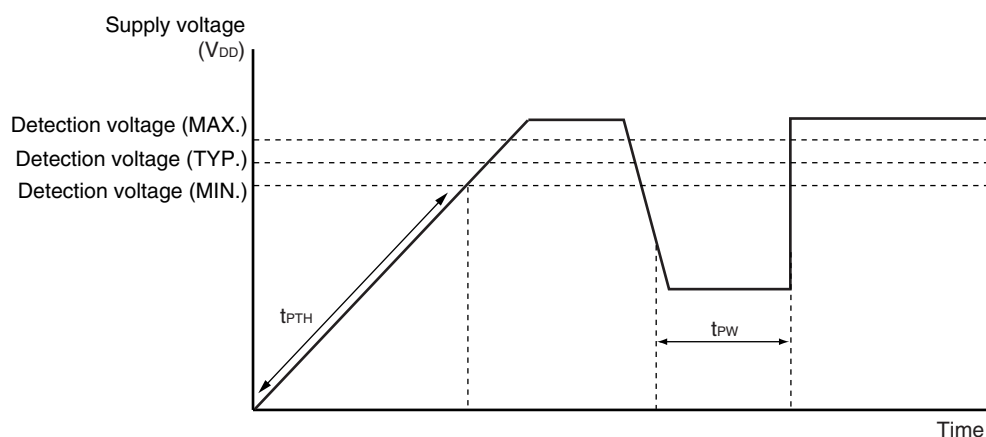


(4) POC

($T_A = -40$ 至 $+85^{\circ}\text{C}$, $V_{SS} = 0\text{ V}$)

项目	符号	条件	最小值	典型值	最大值	单位
检测电压	V_{POC0}		1.44	1.59	1.74	V
电源电压上升斜度	t_{PTH}	V_{DD} 变化高度: $0\text{ V} \rightarrow V_{POC0}$	0.5			V/ms
最小脉冲宽度	t_{PW}	电压降落	200			μs

POC 电路调整



(5) 供电电压上升时间

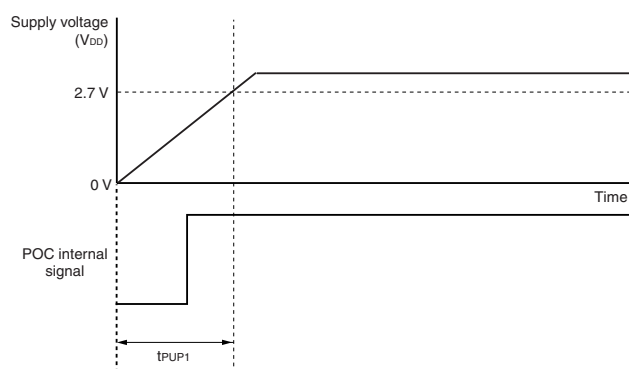
($T_A = -40$ 至 $+85^{\circ}\text{C}$, $V_{SS} = 0\text{ V}$)

项目	符号	条件	最小值	典型值	最大值	单位
上升至 $2.7\text{V}(V_{DD}(\text{MIN.}))$ 所需最长时间 ^注 ($V_{DD}: 0\text{ V} \rightarrow 2.7\text{ V}$)	t_{PUP1}	不使用 $\overline{\text{RESET}}$ 输入时, LVI 默认启动功能无效(LVISTART (选项字节) = 0)。			5.4	ms
上升至 $2.7\text{V}(V_{DD}(\text{MIN.}))$ 所需最长时间 ^注 (解除 $\overline{\text{RESET}}$ 输入 $\rightarrow V_{DD}: 2.7\text{ V}$)	t_{PUP2}	使用 $\overline{\text{RESET}}$ 输入时, LVI 默认启动功能无效(LVISTART (选项字节) = 0)。			1.9	ms

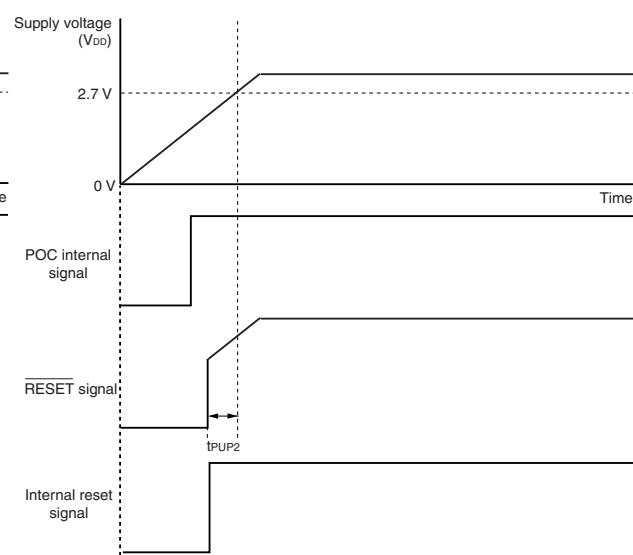
注 上电时间必须小于该时间。

电源电压上升时间的时序

- 不使用 $\overline{\text{RESET}}$ 引脚输入时



- 使用 $\overline{\text{RESET}}$ 引脚输出时(POC 接触后, 由 $\overline{\text{RESET}}$ 引脚解除外部复位时)



(6) 二次供电电压输出

($T_A = -40$ 至 $+85^{\circ}\text{C}$, $V_{SS} = 0\text{ V}$)

项目	符号	条件	最小值	典型值	最大值	单位
CREG 输出电压	V_{REGC}	$2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}$ NV 模式	2.3	2.5	2.7	V
		LV 模式	1.7	1.9	2.1	V

(7) LVI

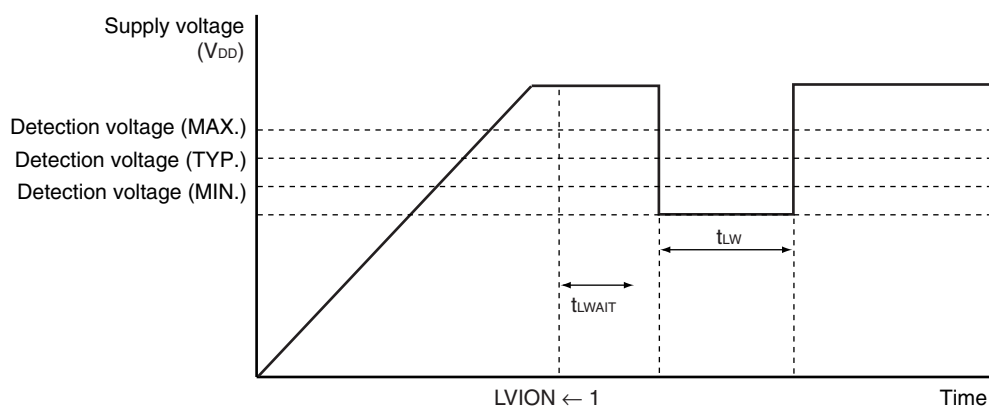
(T_A = -40 至 +85°C, V_{PDR} ≤ V_{DD} ≤ 5.5 V, V_{SS} = 0 V)

项目	符号	条件	最小值	典型值	最大值	单位
检测电压	供电电压电平	V _{LVI0}	4.24±0.1			V
		V _{LVI1}	4.09±0.1			V
		V _{LVI2}	3.93±0.1			V
		V _{LVI3}	3.78±0.1			V
		V _{LVI4}	3.62±0.1			V
		V _{LVI5}	3.47±0.1			V
		V _{LVI6}	3.32±0.1			V
		V _{LVI7}	3.16±0.1			V
		V _{LVI8}	3.01±0.1			V
		V _{LVI9}	2.85±0.1			V
		V _{LVI10}	2.70±0.1			V
	上电时的电源电压	V _{DDLVI}	LVI 默认功能操作操作时(LVISTART = 1)			V
最小脉冲宽度	t _{LW}		200			μs
操作稳定等待时间 [※]	t _{LWAIT}				10	μs

注 低电压检测寄存器(LVIM)的第7位(LVION)置“1”后, 至操作稳定为止所需时间。

备注 V_{LVI(n-1)} > V_{LVI n}: n = 1 至 10

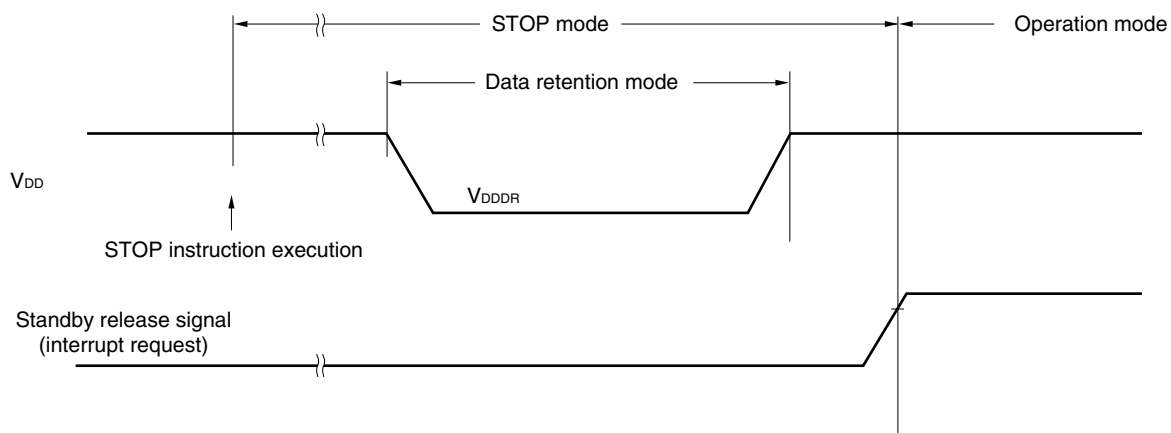
LVI 电路时序



数据存储器 STOP 模式下，低电源电压时的数据保持特性 ($T_A = -40$ 至 $+85^\circ\text{C}$)

项目	符号	条件	最小值	典型值	最大值	单位
数据保留供电电压	V_{DDDR}		1.44 [※]		5.5	V

注 该值取决于 POC 检测电压。电压降落时，该数据会一直保持到 POC 复位有效为止，但 POC 复位有效时数据将消失。



闪存编程特性

($T_A = -40$ 至 $+85^\circ\text{C}$, $2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $V_{SS} = 0\text{ V}$)

• 基本特性

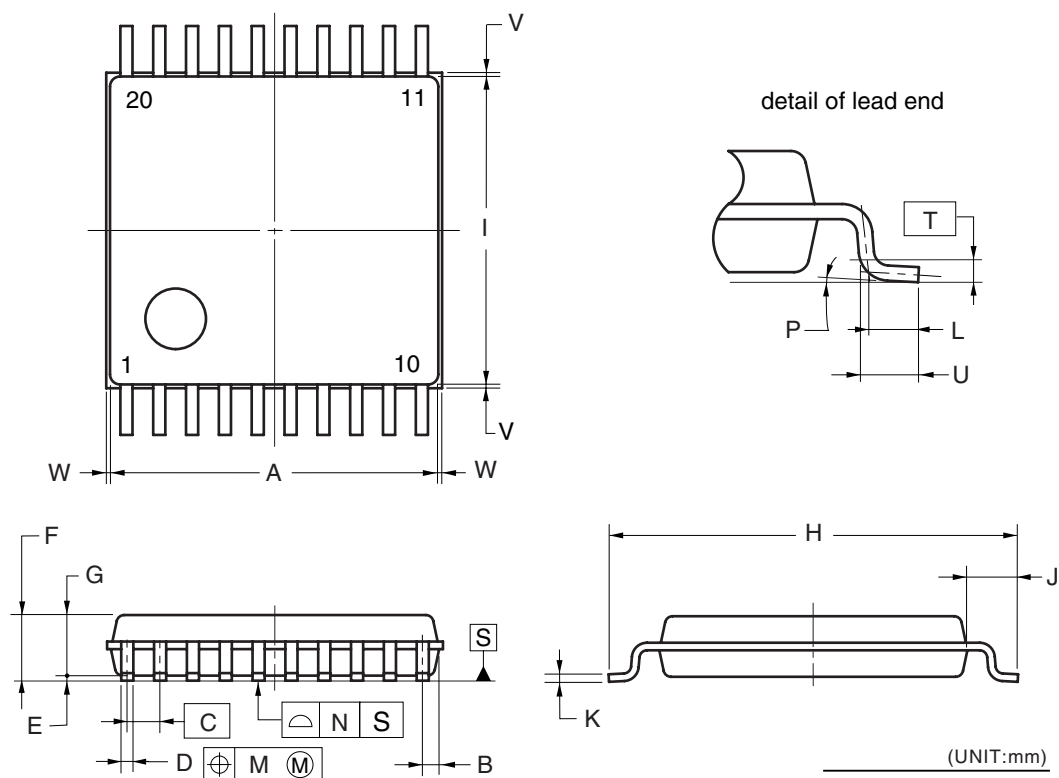
项目	符号	条件			最小值	典型值	最大值	单位
V_{DD} 供电电流	I_{DD}					4.5	11.0	mA
擦除时间	片单位	t_{eraca}				20	200	ms
	扇区单位	t_{erasa}				20	200	ms
写入时间 (8 位单位)	t_{wrwa}					10	100	μs
1 片重写次数 ^注	C_{erwr}	擦写 1 次 + 擦除后写 1 次 = 重写 1 次	使用闪存编程器时，或使用 Renesas Electronics 提供的 自编程库时	保存 15 年	1000			次
工作温度		使用闪存编程器时自编程期间，10 至 40°C ；-40 至 $+85^\circ\text{C}$						
工作电压范围		正常电源模式下 (RMC = 56/59H 以外)	使用闪存编程器时	2.7 至 5.5 V @ 8 MHz (最大值)				
			自编程期间					

注 对出厂后的产品进行第一次写入时，“擦除 → 写入”和“只写”均当作一次重写。

第二十五章 封装尺寸图

• μPD79F7023MC-CAA-AX, 79F7024MC-CAA-AX

20-PIN PLASTIC SSOP (7.62 mm (300))



NOTE

Each lead centerline is located within 0.13 mm of its true position (T.P.) at maximum material condition.

(UNIT:mm)	
ITEM	DIMENSIONS
A	6.50±0.10
B	0.325
C	0.65 (T.P.)
D	0.22 ^{+0.10} _{-0.05}
E	0.10±0.05
F	1.30±0.10
G	1.20
H	8.10±0.20
I	6.10±0.10
J	1.00±0.20
K	0.15 ^{+0.05} _{-0.01}
L	0.50
M	0.13
N	0.10
P	3° ^{+5°} _{-3°}
T	0.25(T.P)
U	0.60±0.15
V	0.25 MAX.
W	0.15 MAX.
P20MC-65-CAA	

第二十六章 待机注意事项

26.1 待机注意事项

本产品有两种内部系统总线。

一种是 CPU 总线，另一种是连接低速外围硬件接口的外围总线。

因为 CPU 总线时钟与外围总线时钟异步，所以对 CPU 的访问与外围硬件的访问之间发生冲突时，可能会产生预期不到的非法数据。

对可能会发生冲突的外围硬件进行访问时，应反复执行处理，直到 CPU 送出正确的数据。

因此，CPU 将不执行下一条指令的处理，而是进入等待状态。在这种情况下，指令执行所需的时钟数会以等待的时钟数而增加（有关详细的等待时钟数，参阅表 26-1）。在执行实时程序中，必须注意这一点。

26.2 产生等待的外围硬件

CPU 访问时，发出等待请求的寄存器和 CPU 等待的时钟数如表 21-1 所示。

表 26-1. 产生等待的寄存器和 CPU 等待时钟数

外围硬件	寄存器	访问	待机时钟数
串行接口 UART0	ASIS0	读取	1 个时钟（固定）
A/D 转换器	ADM0	写入	1 至 5 clocks (选择 $f_{AD} = f_{PRS}/2$ 时)
	ADS	写入	1 至 7 clocks (选择 $f_{AD} = f_{PRS}/3$ 时)
	ADPC	写入	1 至 9 clocks (选择 $f_{AD} = f_{PRS}/4$ 时)
	ADCRH	读取	2 至 13 clocks (选择 $f_{AD} = f_{PRS}/6$ 时)
			2 至 25 clocks (选择 $f_{AD} = f_{PRS}/12$ 时)
上述时钟数是选择相同的源时钟作为 f_{CPU} 和 f_{PRS} 时的值。可以根据以下公式和条件计算等待时钟的个数。 <计算等待时钟数> $\bullet$ 等待时钟数 = $\frac{2 f_{CPU}}{f_{AD}} + 1$ * 若等待时钟的个数 ≤ 0.5 ，则舍去小数部分；若等待时钟的个数 > 0.5 ，则进行四舍五入。 f_{AD} : A/D 转换时钟频率(f_{PRS} 至 $f_{PRS}/12$) f_{CPU} : CPU 时钟频率 f_{PRS} : 外部硬件时钟频率 f_{XP} : 主系统时钟频率 <最大/最小等待时钟数条件> $\bullet$ 最大个数: CPU 的最高速度 (f_{XP}),A/D 转换时钟的最低速度($f_{PRS}/12$) $\bullet$ 最小个数:CPU 的最低速度 (f_{SUB}), A/D 转换时钟的最大速度(f_{PRS})			

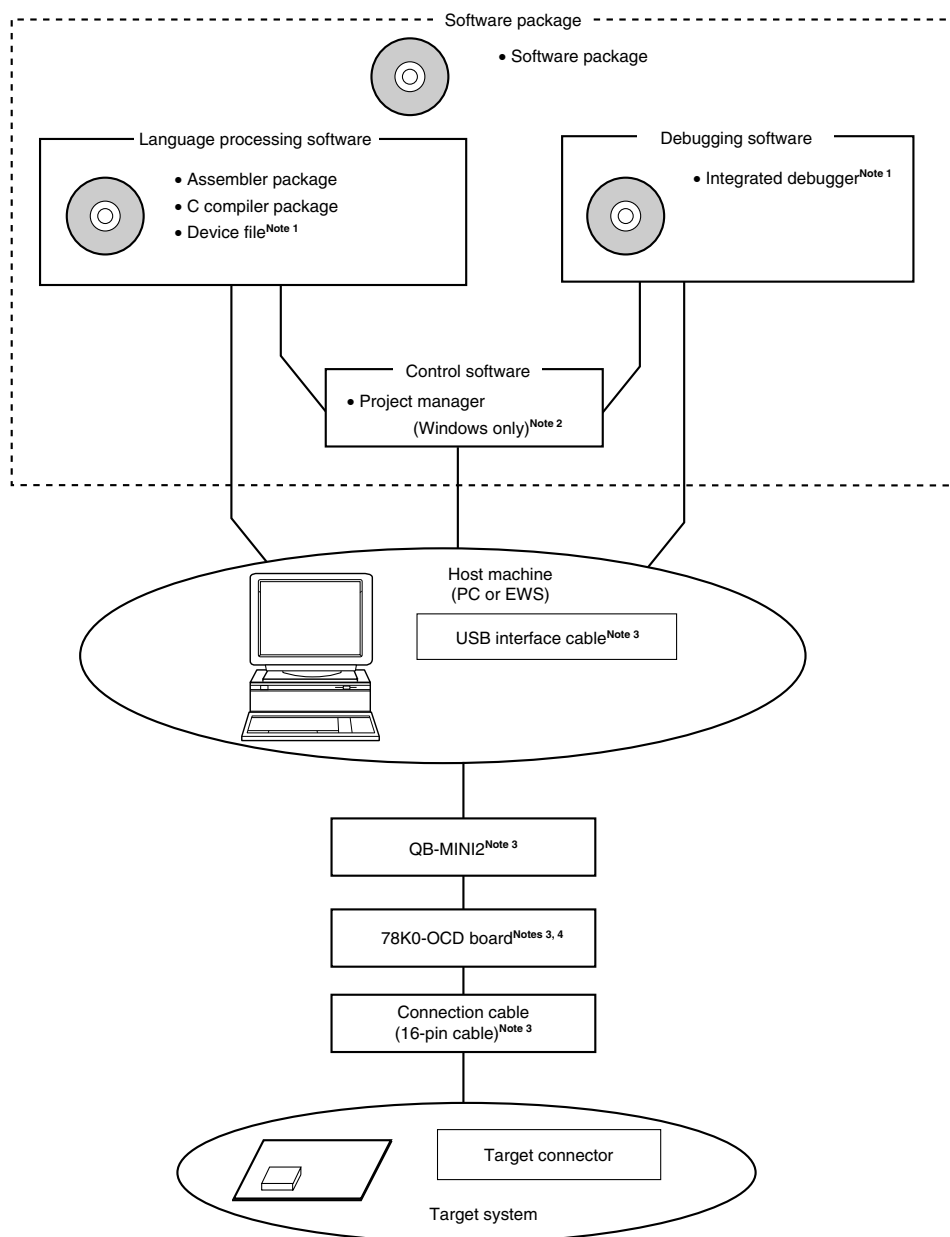
注意事项 外围硬件时钟(f_{PRS})停止时，禁止使用产生等待请求的访问方式访问上述寄存器。

备注 该时钟为 CPU 时钟(f_{CPU})。

附录 A 开发工具

以下开发工具可用来开发使用 μPD79F7023, 79F7024 微控制器的系统。
图 A-1 显示了开发工具的配置。

图 A-1. 开发工具的配置



- 注 1. 此外，请从开发工具的下载网站处下载操作μPD79F7023, 79F7024 的微控制器（开发中）的设备文件和集成调试器ID78K0-QB (<http://www2.renesas.com/micro/en/ods/index.html>)。
2. 项目管理器 PM+ 包含在汇编包内。
PM+ 只能在 Windows 环境下使用。
3. 片上调试仿真器 QB-MINI2 提供 USB 接口电缆、连接电缆（10 芯电缆、16 芯电缆）和 78K0-OCD 板。此外，请从开发工具的下载网站处下载操作QB-MINI2 的软件 (<http://www2.renesas.com/micro/en/ods/index.html>)。
4. 仅在使用 QB-MINI2 作为片上调试仿真器时使用。

A.1 软件包

SP78K0 78K0 微控制器的软件包	本软件包包括用于 78K0 微控制器的通用开发工具（软件）。
-------------------------	--------------------------------

A.2 语言处理软件

RA78K0 [※] 汇编包	本汇编器将用 mnemonics 指令写的程序转换为微控制器执行的目标代码。此外，本汇编器还具备能够自动生成符号表、优化转移指令的功能。它应与设备文件组合使用（开发中）。 <在 PC 环境中使用 RA78K0 时的注意点> 该汇编包是基于 DOS 的应用程序。通过在 Windows 上使用项目管理器 (PM+)，可以在 Windows 环境中使用。PM+ 包含在汇编包内。
CC78K0 [※] C 编译包	本编译器将用 C 语言写的程序转换为微控制器执行的目标代码。该编译器应和汇编包和设备文件（开发中）配合使用。 <在 PC 环境中使用 CC78K0 时的注意点> 该 C 编译包是基于 DOS 的应用程序。通过在 Windows 上使用项目管理器 (PM+)，可以在 Windows 环境中使用。PM+ 包含在汇编包内。
设备文件 (开发中)	本文件包含设备的特有信息。 该设备文件应和开发工具（RA78K0, CC78K0 和 ID78K0-QB）配合使用。 相应的主机和操作系统随所用工具的不同而异。

注 如果 RA78K0 和 CC78K0 版本为 4.00 或更高版本时，同一台设备上可以安装 RA78K0 和 CC78K0 的不同版本。

A.3 闪存编程器的工具

A.3.1 使用闪存编程器 PG-FP5 和 FL-PR5

PG-FP5, FL-PR5 闪存编程器	闪存编程器专用于内置闪存的微控制器。
FA-78F0567MC-CAA-RX 闪存编程适配器	闪存编程适配器。可连接闪存编程器来使用。

备注 1. FL-PR5 和 FA-78F0567MC-CAA-RX 为 Naito Densei Machida Mfg. Co., Ltd 的产品。

TEL: +81-42-750-4172 Naito Densei Machida Mfg. Co., Ltd.

2. 使用闪存编程适配器的最新版本。

A.3.2 使用具备编程功能的片上调试仿真器 QB-MINI2

QB-MINI2 具备变成功能的片上调试仿真器	这是一个带片上闪存的微控制器的专用闪存编程器。使用 μPD79F7023, 79F7024 微控制器进行应用系统的开发时，也可作为调试硬件和软件的片上调试仿真器来使用。当作为闪存编程器来使用时，需要把连接电缆 (16 芯电缆)和用来连接主机的 USB 接口电缆组合使用。
目标连接器的规格	16 针通用连接器 (2.54 mm 间距)

备注 请从开发工具的下载网站处下载操作QB-MINI2 的软件(<http://www2.renesas.com/micro/en/ods/index.html>)。

A.4 调试工具（硬件）

QB-MINI2 具备编程功能的片上调试仿真器	使用 μPD79F7023, 79F7024 微控制器进行应用系统的开发时，作为调试硬件和软件的片上调试仿真器来使用。也可作为带片上闪存的微控制器的专用闪存编程器来使用。当作为片上调试仿真器来使用时，需要把连接电缆 (16 芯电缆)、用来连接主机的 USB 接口电缆和 78K0-OCD 版一起组合使用。
目标连接器的规格	16 针通用连接器 (2.54 mm 间距)

备注 请从开发工具的下载网站处下载操作QB-MINI2 的软件(<http://www2.renesas.com/micro/en/ods/index.html>)。

A.5 调试工具（软件）

ID78K0-QB 集成调试器	本调试器支持 78K0 微控制器的在线仿真器。ID78K0-QB 是基于 Windows 操作系统的软件。 它具有改进的兼容 C 语言的调试功能，并且，使用一个与源程序相关的窗口集成功能来显示源程序追踪执行的结果，反汇编显示、带有追踪结果的存储显示。它应与设备文件组合使用（开发中）。
--------------------	---

修订记录	μPD79F7023, 79F7024 用户手册 硬件篇
------	------------------------------

版本	日期	说明	
1.00	2012.03.16	—	初次制作
1.10	2013.11.29	—	更改了封面的注意文本

μPD79F7023, 79F7024 用户手册 硬件篇

Publication Date: Rev.1.10 Nov 29, 2013

Published by: Renesas Electronics Corporation



SALES OFFICES

Renesas Electronics Corporation

<http://www.renesas.com>

Refer to "<http://www.renesas.com/>" for the latest and detailed information.

Renesas Electronics America Inc.

2880 Scott Boulevard Santa Clara, CA 95050-2554, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130

Renesas Electronics Canada Limited

1101 Nicholson Road, Newmarket, Ontario L3Y 9C3, Canada
Tel: +1-905-898-5441, Fax: +1-905-898-3220

Renesas Electronics Europe Limited

Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K.
Tel: +44-1628-651-700, Fax: +44-1628-651-804

Renesas Electronics Europe GmbH

Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-65030, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.

7th Floor, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100083, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.

Unit 204, 205, AZIA Center, No.1233 Lujiazui Ring Rd., Pudong District, Shanghai 200120, China
Tel: +86-21-5877-1818, Fax: +86-21-6887-7858 / -7898

Renesas Electronics Hong Kong Limited

Unit 1601-1613, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2886-9318, Fax: +852 2886-9022/9044

Renesas Electronics Taiwan Co., Ltd.

13F, No. 363, Fu Shing North Road, Taipei, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

Renesas Electronics Singapore Pte. Ltd.

80 Bendemeer Road, Unit #06-02 Hyflux Innovation Centre Singapore 339949
Tel: +65-6213-0200, Fax: +65-6213-0300

Renesas Electronics 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: +60-3-7955-9390, Fax: +60-3-7955-9510

Renesas Electronics Korea Co., Ltd.

11F., Samik Lavied' or Bldg., 720-2 Yeoksam-Dong, Kangnam-Ku, Seoul 135-080, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141

μPD79F7023, 79F7024



瑞萨电子株式会社

R01UH0312CJ0110