

用户手册初稿

78K0R/Kx3-L

16 位微控制器

μ PD78F1000

μ PD78F1001

μ PD78F1002

μ PD78F1003

μ PD78F1004

μ PD78F1005

μ PD78F1006

μ PD78F1007

μ PD78F1008

μ PD78F1009

[备忘录]

CMOS 设备注意事项

① 输入引脚处的电压波形

输入噪音或一个反射波引起的波形失真可能导致错误发生。如果由于噪音等的影响使CMOS设备的输入电压范围保持在VIL(MAX)和VIH(MIN)之间，设备可能发生错误。在输入电平固定时以及输入电平从VIL(MAX)过渡到VIH(MIN)时的传输期间，要防止散射噪声影响设备。

② 未使用的输入引脚的处理

CMOS设备的输入端保持开路可能导致误操作。如果一个输入引脚未被连接，则由于噪音等原因可能会产生内部输入电平，从而导致误操作。CMOS设备的操作特性与Bipolar或NMOS设备不同。CMOS设备的输入电平必须借助上拉或下拉电路固定在高电平或低电平。每一个未使用引脚都应该通过附加电阻连接到VDD或GND。如果有可能尽量定义为输出引脚。对未使用引脚的处理因设备而异，必须遵循与设备相关的规定和说明。

③ ESD防护措施

如果MOS设备周围有强电场，将会击穿氧化栅极，从而影响设备的运行。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。对于环境必须有适当的控制。如果空气干燥，应当使用增湿器。建议避免使用容易产生静电的绝缘体。半导体设备的存放和运输必须使用抗静电容器、抗静电屏蔽袋或导电材料容器。所有的测试和测量工具包括工作台和工作面必须良好接地。操作员应当佩戴静电消除手带以保证良好接地。不能用手直接接触半导体设备。对于装配有半导体设备的PW板也应采取类似的静电防范措施。

④ 初始化之前的状态

在上电时MOS设备的初始状态是不确定的。在刚刚上电之后，具有复位功能的MOS设备并没有被初始化。因此上电不能保证输出引脚的电平，I/O设置和寄存器的内容。设备在收到复位信号后才进行初始化。具有复位功能的设备在上电后必须立即进行复位操作。

⑤ 电源开关顺序

在一个设备的内部操作和外部接口使用不同的电源的情况下，按照规定，应先在接通内部电源之后再接通外部电源。当关闭电源时，按照规定，先关闭外部电源再关闭内部电源。如果电源开关顺序颠倒，可能会导致设备的内部组件过电压，产生异常电流，从而引起内部组件的误操作和性能的退化。对于每个设备电源的正确开关顺序必须依据设备的规范说明分别进行判断。

⑥ 电源关闭状态下的输入信号

不要向没有加电的设备输入信号或提供I/O上拉电源。因为输入信号或提供I/O上拉电源将引起电流注入，从而引起设备的误操作，并产生异常电流，从而使内部组件退化。每个设备电源关闭时的信号输入必须依据设备的规范说明分别进行判断。

Windows 和 WindowsNT 是 Microsoft Corporation 在美国及其他国家的注册商标和商标。
PC/AT 是 International Business Machines Corporation 的商标。
EEPROM 是 NEC Electronics Corporation 的商标
SuperFlash®是 Silicon Storage Technology, Inc.的一个注册商标, 已经在美国和日本等几个国家使用。

注意事项: 该产品使用的SuperFlash®技术获得了Silicon Storage Technology, Inc.公司的授权。

- 本文档所登载的内容有效期截止至 2009 年 3 月, 信息先于产品的生产周期发布。将来可能未经预先通知而更改。在实际进行生产设计时, 请参阅各产品最新的数据表或数据手册等相关资料以获取本公司产品的最新规格。
- 并非所有的产品和/或型号都向每个国家供应。请向本公司销售代表查询产品供应及其他信息。
- 未经本公司事先书面许可, 禁止复制或转载本文件中的内容。否则因本文件所登载内容引发的错误, 本公司概不负责。
- 本公司对于因使用本文件中列明的本公司产品而引起的, 对第三者的专利、版权以及其它知识产权的侵权行为概不负责。本文件登载的内容不应视为本公司对本公司或其他人所有的专利、版权以及其它知识产权作出任何明示或默示的许可及授权。
- 本文件中的电路、软件以及相关信息仅用以说明半导体产品的运作和应用实例。用户如在设备设计中应用本文件中的电路、软件以及相关信息, 应自行负责。对于用户或其他人因使用了上述电路、软件以及相关信息而引起的任何损失, 本公司概不负责。
- 虽然本公司致力于提高半导体产品的质量及可靠性, 但用户应同意并知晓, 我们仍然无法完全消除出现产品缺陷的可能。为了最大限度地减少因本公司半导体产品故障而引起的对人身、财产造成损害(包括死亡)的危险, 用户务必在其设计中采用必要的安全措施, 如冗余度、防火和防故障等安全设计。
- 本公司产品质量分为:
“标准等级”、“专业等级”以及“特殊等级”三种质量等级。

“特殊等级”仅适用于为特定用途而根据用户指定的质量保证程序所开发的日电电子产品。另外, 各种日电电子产品的推荐用途取决于其质量等级, 详见如下。用户在选用本公司的产品时, 请事先确认产品的质量等级。

“标准等级”: 计算机, 办公自动化设备, 通信设备, 测试和测量设备, 音频·视频设备, 家电, 加工机械以及产业用机器人。

“专业等级”: 运输设备(汽车、火车、船舶等), 交通信号控制设备, 防灾装置, 防止犯罪装置, 各种安全装置以及医疗设备(不包括专门为维持生命而设计的设备)。

“特殊等级”: 航空器械, 宇航设备, 海底中继设备, 原子能控制系统, 为了维持生命的医疗设备、用于维持生命的装置或系统等。

除在本公司半导体产品的数据表或数据手册等资料中另有特别规定以外, 本公司半导体产品的质量等级均为“标准等级”。如果用户希望在本公司设计意图以外使用本公司半导体产品, 务必事先与本公司销售代表联系以确认本公司是否同意为该应用提供支持。

(注)

- (1) 本声明中的“本公司”是指日本电气电子株式会社(NEC Electronics Corporation)及其控股公司。
- (2) 本声明中的“本公司产品”是指所有由日本电气电子株式会社开发或制造的产品或为日本电气电子株式会社(定义如上)开发或制造的产品。

M5 02.11-1

前言

读者 本手册旨在使用户了解 78K0R/Kx3-L 的功能及设计开发应用系统。
以下为目标设备产品。

- 78K0R/KC3-L: μ PD78F1000, 78F1001, 78F1002, 78F1003
- 78K0R/KD3-L: μ PD78F1004, 78F1005, 78F1006
- 78K0R/KE3-L: μ PD78F1007, 78F1008, 78F1009

目的 本手册旨使用户对功能描述的了解，以下为手册的**组织结构**。

组织 78K0R/Kx3-L 手册分为两部分：本手册和指令部分 (与 78K0R 微控制器通用)。

78K0R/Kx3-L 用户手册 (本手册)	78K0R 微控制器 指令用户手册

• 引脚功能 • 内部寄存器功能 • 中断 • 片上外围功能 • 电气特性 (目标值)	• CPU 功能 • 指令设置 • 各指令说明

如何阅读本手册 在阅读本手册前，读者应掌握电子工程、逻辑电路和微控制器等电子工程方面的基础知识。

- <R>**
- 要掌握全部功能:
→ 请按**目录**顺序阅读本手册 标注“<R>”表示主要修改部分。 修订部分可以很方便地通过在 PDF 文件中拷贝 “<R>” 并通过在 “查找:” 中指定来搜索查询。 field.
 - 如何解释寄存器格式:
→ 尖括号(<>)中的二进制位名称在 RA78K0R 中被定义为保留字，并且在 CC78K0R 中用#pragma sfr 指令定义为一个 sfr 变量.
 - 要了解 78K0R 微控制器指令功能的详细内容
→ 请参考 **78K0R 微控制器指令用户手册(U17792E)**

规则	数据规则:	数据的高位部分在左边，低位部分在右边
	有效低电平表示法:	xxx xxx (在引脚和信号名称上加划一条线)
	注:	文中用 注 标注的相关术语的脚注
	注意事项:	需要特别关注的信息
	备注:	补充信息
	数值的表示:	二进制 ...xxxx 或 xxxxB 十进制 ...xxxx 十六进制 ...xxxxH

相关文档 本手册中提到的相关文档可能包括有初稿版本。 但是，初稿版本没有特别注明。

设备相关文档

文档名称	文档编号
78K0R/Kx3-L 用户手册	本手册
78K0R 微控制器指令用户手册	U17792E

开发工具相关文档 (软件) (用户手册)

文档名称		文档编号
CC78K0R Ver. 2.00 C 编译器	操作篇	U18549E
	语言篇	U18548E
RA78K0R Ver. 1.20 汇编包	操作篇	U18547E
	语言篇	U18546E
SM+ 系统仿真器	操作篇	U18010E
PM+ Ver. 6.30		U18416E
ID78K0R-QB Ver. 3.20 综合调试器	操作篇	U17839E

开发工具相关文档 (硬件) (用户手册)

文档名称	文档编号
QB-MINI2 具有编程功能的片上调试仿真器	U18371E
QB-78K0RIX3 片上仿真器	准备中

Flash存储器编程相关文档

文档名称	文档编号
PG-FP4 存储器编程器用户手册	U15260E
PG-FP5 存储器编程器用户手册	U18865E

注意事项: 以上列出的相关文档可能会在无任何声明条件下修改。 开发设计时，请使用各文档的最新版本。

其他文档

文档名称	文档编号
半导体选择指南 - 产品和封装-	X13769X
半导体设备装配手册	注:
NEC 半导体设备的质量等级	C11531E
NEC 半导体设备可靠性/质量控制系统	C10983E
半导体设备防静电 ESD 指南	C11892E

注: 可参阅“半导体设备装配手册”网站 (<http://www.necel.com/pkg/en/mount/index.html>)。

注意事项: 以上列出的相关文档可能会在无任何声明条件下修改。 开发设计时, 请使用各文档的最新版本。

目录

第一章 概述	18
1.1 特性	18
1.2 应用	19
1.3 订货信息	19
1.4 引脚配置（顶视图）	20
1.4.1 78K0R/KC3-L	20
1.4.2 78K0R/KD3-L	22
1.4.3 78K0R/KE3-L	23
1.5 引脚识别	25
1.6 程序块框图	26
1.6.1 78K0R/KC3-L	26
1.6.2 78K0R/KD3-L	28
1.6.3 78K0R/KE3-L	29
1.7 功能概述	30
第二章 引脚功能	32
2.1 引脚功能列表	32
2.1.1 78K0R/KC3-L	33
2.1.2 78K0R/KD3-L	38
2.1.3 78K0R/KE3-L	43
2.2 引脚功能说明	48
2.2.1 P00, P01（端口 0）	48
2.2.2 P10 至 P17（端口 1）	49
2.2.3 P20 至 P27（端口 2）	50
2.2.4 P30 至 P33（端口 3）	50
2.2.5 P40 至 P43（端口 4）	52
2.2.6 P50 至 P53（端口 5）	53
2.2.7 P60 和 P61（端口 6）	54
2.2.8 P70 至 P77（端口 7）	55
2.2.9 P80 至 P83（端口 8）	56
2.2.10 P120 至 P124（端口 12）	57
2.2.11 P140, P141（端口 14）	58
2.2.12 P150 至 P153（端口 15）	58
2.2.13 AVREF, AVSS, VDD, EVDD, VSS, EVSS	59
2.2.14 RESET	60
2.2.15 REGC	60
2.2.16 FLMD0	60
2.3 引脚 I/O 电路以及未使用引脚连接建议	61

第三章 CPU 架构	67
3.1 存储空间	67
3.1.1 内部编程存储器空间	73
3.1.2 镜像区域	75
3.1.3 内部数据存储器空间	76
3.1.3 特殊功能寄存器（SFR）区域	77
3.1.5 扩展的特殊功能寄存器（第二 SFR：第二特殊功能寄存器）区域	77
3.1.6 数据存储器寻址	78
3.2 处理器寄存器	82
3.2.1 控制寄存器	82
3.2.2 通用寄存器	84
3.2.3 ES 和 CS 寄存器	86
3.2.4 特殊功能寄存器（SFR）	87
3.2.5 扩展特殊功能寄存器（第二 SFR：第二特殊功能寄存器）	92
3.3 指令地址寻址	97
3.3.1 相对寻址	97
3.3.2 直接寻址	97
3.3.3 表间接寻址	98
3.3.4 寄存器直接寻址	99
3.4 处理数据地址寻址	100
3.4.1 隐式寻址	100
3.4.2 寄存器寻址	100
3.4.3 直接寻址	101
3.4.4 短直接寻址	102
3.4.5 SFR 寻址	103
3.4.6 寄存器间接寻址	104
3.4.7 基址寻址	105
3.4.8 基址变址寻址	108
3.4.9 堆栈寻址	109
第四章 端口功能	110
4.1 端口功能	110
4.2 端口配置	113
4.2.1 端口 0	114
4.2.2 端口 1	116
4.2.3 端口 2	119
4.2.4 端口 3	121
4.2.5 端口 4	125
4.2.6 端口 5	128
4.2.7 端口 6	132

4.2.8 端口 7.....	133
4.2.9 端口 8.....	138
4.2.10 端口 12.....	141
4.2.11 端口 14.....	145
4.2.12 端口 15.....	147
4.3 控制端口功能的寄存器.....	149
4.4 端口功能操作.....	162
4.4.1 写入 I/O 端口.....	162
4.4.2 从 I/O 端口读取.....	162
4.4.3 I/O 端口的操作.....	162
4.4.4 与具有不同电位的外部设备（2.5 V、3 V）相连接.....	163
4.5 使用复用功能时端口寄存器和输出锁存器的设置.....	165
4.6 端口寄存器 n（Pn）的 1 位操作指令的注意事项.....	168
第五章 时钟发生器.....	169
5.1 时钟发生器的功能.....	169
5.2 时钟发生器的配置.....	170
5.3 控制时钟发生器的寄存器.....	172
5.4 系统时钟振荡器.....	186
5.4.1 X1 振荡器.....	186
5.4.2 XT1 振荡器.....	186
5.4.3 内部高速振荡器 ^注	190
5.4.4 20 MHz 内部高速振荡器 ^注	190
5.4.5 内部低速振荡器.....	190
5.4.6 预分频器.....	190
5.5 时钟发生器的操作.....	191
5.6 控制时钟.....	197
5.6.1 控制高速系统时钟的示例.....	197
5.6.2 控制内部高速振荡时钟的示例.....	200
5.6.3 控制副系统时钟的示例.....	202
5.6.4 控制内部低速振荡时钟的示例.....	204
5.6.5 CPU 时钟状态转换框图.....	205
CPU 时钟改变前的条件和改变后的处理.....	212
5.6.7 CPU 时钟和副系统时钟转换所需的时间.....	214
5.6.8 时钟振荡停止前的条件.....	215
第六章 定时器阵列部件 TAUS.....	216
6.1 定时器阵列部件 TAUS 的功能.....	217
6.1.1 独立工作时各通道的功能.....	217
6.1.2 与其它通道同时工作时各通道的功能.....	218

6.1.3 LIN 总线支持功能（仅通道 7）	218
6.2 定时器阵列部件 TAUS 的配置	219
6.3 控制定时器阵列部件的寄存器	225
6.4 通道输出（TO0n 引脚）控制	247
6.4.1 TO0n 引脚输出电路配置	247
6.4.2 TO0n 引脚输出设置	248
6.4.3 通道输出操作时的注意事项	248
6.4.4 TO0n 位的集合操作	252
6.4.5 操作开始时的定时器中断和 TO0n 引脚输出	253
6.5 通道输入（TI0n 引脚）控制	254
6.5.1 TI0n 边沿检测电路	254
6.6 定时器阵列部件 TAUS 的基本功能	255
6.6.1 单一操作功能和组合操作功能的概述	255
6.6.2 组合操作功能的基本规则	255
6.6.3 组合操作功能基本规则的可用范围	256
6.7 定时器阵列部件 TAUS 作为独立通道的操作	257
6.7.1 作为间隔定时器/方波输出的操作	257
6.7.2 作为外部事件计数器的操作	263
6.7.3 作为分频器（仅限于 78K0R/KD3-L 和 78K0R/KE3-L 通道 0）的操作	266
6.7.4 用作输入脉冲测量的操作	270
6.7.5 用作输入信号高/低电平宽度测量的操作	274
6.8 定时器阵列部件 TAUS 复通道的操作	278
6.8.1 用作 PWM 功能的操作	278
6.8.2 作为单次脉冲输出功能的操作	285
6.8.3 用作复用 PWM 输出功能的操作	292
第七章 实时计数器	299
7.1 实时计数器的功能	299
7.2 实时计数器的配置	299
7.3 控制实时计数器的寄存器	301
7.4 实时计数器操作	316
7.4.1 实时计数器的启动操作	316
7.4.2 启动操作后转换至 STOP 模式	317
7.4.3 读取/写入实时计数器	318
7.4.4 实时计数器的报警设置	320
7.4.5 实时计数器的 1 Hz 输出设置	321
7.4.6 实时计数器的 32.768 kHz 输出设置	321
7.4.7 实时计数器的 512 Hz, 16.384 kHz 输出设置	321
7.4.8 实时计数器钟表错误校正的举例	322

第八章 比较器/可编程增益放大器	327
8.1 比较器和可编程增益放大器的特性	327
8.2 比较器和可编程增益放大器的配置	329
8.3 控制比较器和可编程增益放大器的寄存器.....	329
8.4 比较器和可编程增益放大器的操作	335
8.4.1 启动比较器和可编程增益放大器的操作	335
8.4.2 停止比较器和可编程增益放大器的操作	340
第九章 时钟输出/蜂鸣器输出控制器.....	341
9.1 时钟输出/蜂鸣器输出控制器的功能	341
9.2 时钟输出/蜂鸣器输出控制器的配置	343
9.3 控制时钟输出/蜂鸣器输出控制器的寄存器.....	343
9.4 时钟输出/蜂鸣器输出控制器的操作	345
9.4.1 作为输出引脚的操作	345
第十章 看门狗定时器	346
10.1 看门狗定时器的功能	346
10.2 看门狗定时器的配置	347
10.3 控制看门狗定时器的寄存器	348
10.4 看门狗定时器的操作	349
10.4.1 看门狗定时器的控制操作	349
10.4.2 看门狗定时器的溢出时间设置.....	350
10.4.3 看门狗定时器窗口开启周期的设置.....	351
10.4.4 看门狗定时器间隔中断的设置.....	352
第十一章 A/D 转换器	353
11.1 A/D 转换器的功能.....	353
11.2 A/D 转换器的配置.....	354
11.3 A/D 转换器中使用的寄存器.....	356
11.4 A/D 转换器操作	369
11.4.1 A/D 转换器的基本操作	369
11.4.2 输入电压和转换结果	371
11.4.3 A/D 转换器操作模式	372
11.5 如何读取 A/D 转换器参数表.....	375
11.6 A/D 转换器的注意事项.....	377
第十二章 串行阵列部件.....	381
12.1 串行阵列部件的功能	381
12.1.1 3-线串行 I/O (CSI00、CSI01、CSI10)	381
12.1.2 UART (UART0、UART1)	382

12.1.3 简易的 I ² C (IIC10)	382
12.2 串行阵列部件的配置	383
12.3 控制串行阵列部件的寄存器	387
12.4 操作停止模式	409
12.4.1 由部件停止操作	409
12.4.2 由通道停止操作	410
12.5 3-线串行 I/O (CSI00、CSI01、CSI10) 通信的操作	411
12.5.1 主机发送	412
12.5.2 主机接收	421
12.5.3 主机发送/接收	427
12.5.4 从机发送	435
12.5.5 从机接收	444
12.5.6 从机发送/接收	450
12.5.7 计算传输时钟频率	459
12.6 UART (UART0、UART1) 通信的操作	461
12.6.1 UART 发送	462
12.6.2 UART 接收	472
12.6.3 LIN 发送	479
12.6.4 LIN 接收	482
12.6.5 计算波特率	487
12.7 简易 I²C (IIC10) 通信的操作	491
12.7.1 地址域发送	492
12.7.2 数据发送	497
12.7.3 数据接收	500
12.7.4 产生停止条件	504
12.7.5 计算传输速率	505
12.8 发生错误时的处理步骤	507
12.9 寄存器设置和引脚之间的关系	509
第十三章 串行接口 IICA	513
13.1 串行接口 IICA 的功能	513
13.2 串行接口 IICA 的配置	516
13.3 控制串行接口 IICA 的寄存器	519
13.4 I²C 总线模式功能	532
13.4.1 引脚配置	532
13.4.2 使用 IICWL 和 IICWH 寄存器设置传输时钟	533
13.5 I²C 总线的定义和控制方法	534
13.5.1 开始条件	534
13.5.2 地址	535
13.5.3 传输方向指定	535

13.5.4 响应信号 (ACK)	536
13.5.5 停止条件	537
13.5.6 等待状态	538
13.5.7 取消等待	540
13.5.8 中断请求 (INTIICA) 产生时序和等待控制	541
13.5.9 地址匹配检测方法	542
13.5.10 错误检测	542
13.5.11 扩展码	542
13.5.12 仲裁	543
13.5.13 唤醒功能	545
13.5.14 通信保留	548
13.5.15 注意事项	552
13.5.16 通信操作	553
13.5.17 I ² C 中断请求 (INTIICA) 发生的时序	561
13.6 时序图	582
第十四章 乘法器/除法器	589
14.1 乘法器/除法器的功能	589
14.2 乘法器/除法器的配置	589
14.3 控制乘法器/除法器的寄存器	594
14.4 乘法器/除法器的操作	595
14.4.1 乘法操作	595
14.4.2 除法操作	596
第十五章 DMA 控制器	598
15.1 DMA 控制器的功能	598
15.2 DMA 控制器的配置	599
15.3 控制 DMA 控制器的寄存器	602
15.4 DMA 控制器的操作	605
15.4.1 操作步骤	605
15.4.2 传输模式	606
15.4.3 DMA 传输终止	606
15.5 DMA 控制器设置举例	607
15.5.1 CSI 连续发送	607
15.5.2 连续捕获 A/D 转换结果	609
15.5.3 UART 连续接收+ ACK 发送	611
15.5.4 由 DWAITn 保持 DMA 传输挂起	613
15.5.5 用软件强制终止	614
15.6 使用 DMA 控制器的注意事项	615

第十六章 中断功能	617
16.1 中断功能类型	617
16.2 中断源和配置	617
16.3 控制中断功能的寄存器	622
16.4 中断服务操作	632
16.4.1 可屏蔽中断请求响应.....	632
16.4.2 软件中断请求响应	635
16.4.3 多重中断服务	635
16.4.4 中断请求保持	639
第十七章 按键中断功能	640
17.1 按键中断功能	640
17.2 按键中断的配置	640
17.3 控制按键中断的寄存器	642
第十八章 待机功能	643
18.1 待机功能和配置	643
18.1.1 待机功能	643
18.1.2 控制待机功能的寄存器.....	644
18.2 待机功能操作	647
18.2.1 HALT 模式.....	647
18.2.2 STOP 模式	652
第十九章 复位功能	657
19.1 确认复位源的寄存器	666
第二十章 上电清零电路	667
20.1 上电清零电路的功能	667
20.2 上电清零电路的配置	668
20.3 上电清零电路的操作	668
20.4 上电清零电路的注意事项	671
第二十一章 低电压检测器	673
21.1 低电压检测器的功能	673
21.2 电压检测器的配置	674
21.3 控制低电压检测器的寄存器	674
21.4 低电压检测器的操作	679
21.4.1 用作复位时	680
21.4.2 用作中断时	686
21.5 有关低电压检测器的注意事项	692

第二十二章 稳压器	696
22.1 稳压器概述	696
22.2 控制稳压器的寄存器	696
第二十三章 选项字节	698
23.1 选项字节的功能	698
23.1.1 用户选项字节（000C0H 至 000C2H/010C0H 至 010C2H）	698
23.1.2 片上调试选项字节（000C3H/ 010C3H）	699
23.2 用户选项字节的格式	699
23.3 片上调试选项字节的格式	701
23.4 选项字节的设置	702
第二十四章 FLASH 存储器	703
24.1 用 Flash 存储器编程器写入	703
24.2 编程环境	708
24.3 通信模式	708
24.4 板上引脚的连接	709
24.4.1 FLMD0 引脚	709
24.4.2 TOOL0 引脚	710
24.4.3 RESET 引脚	710
24.4.4 端口引脚	711
24.4.5 REGC 引脚	711
24.4.6 X1 和 X2 引脚	711
24.4.7 电源供应	711
24.5 控制 Flash 存储器的寄存器	712
24.6 编程方法	712
24.6.1 控制 flash 存储器	712
24.6.2 Flash 存储器编程模式	713
24.6.3 选择通信模式	713
24.6.4 通信命令	714
24.7 安全设置	715
24.8 用自编程进行 Flash 存储器编程	717
24.8.1 引导区交换功能	719
24.8.2 Flash 屏蔽窗口功能	721
第二十五章 片上调试功能	722
25.1 将 QB-MINI2 连接到 78K0R/Kx3-L	722
25.2 片上调试安全 ID	723
25.3 用户资源保护	723

第二十六章 BCD 校正电路	725
26.1 BCD 校正电路功能.....	725
26.2 用于 BCD 校正电路的寄存器.....	725
26.3 BCD 校正电路操作.....	726
第二十七章 指令集.....	728
27.1 操作列表中的使用惯例	728
27.1.1 操作数标识符和指定方法	728
27.1.2 操作栏的描述	729
27.1.3 标志操作栏的描述	730
27.1.4 PREFIX 指令	730
27.2 操作列表.....	731
第二十八章 电气特性 （目标）	748
第二十九章 封装图.....	797
29.1 78K0R/KC3-L （44-引脚产品）	797
29.2 78K0R/KC3-L （48-引脚产品）	798
29.3 78K0R/KD3-L	799
29.4 78K0R/KE3-L.....	800
附录 A 开发工具	804
A.1 软件包	807
A.2 语言处理软件	807
A.3 Flash 存储器编程工具	808
A.3.1 使用 flash 存储器编程器 PG-FP5, FL-PR5, PG-FP4 and FL-PR4 时	808
A.3.2 使用带编程功能的片上调试仿真器 QB-MINI2 时	808
A.4 调试工具（硬件）	809
A.4.1 使用在线仿真器 QB-78K0RIX3 时	809
A.4.2 使用带编程功能的片上调试仿真器 QB-MINI2 时	810
A.5 调试工具（软件）	810
附录 B 修订历史	811
B.1 本版中主要修订内容	811

第一章 概述

1.1 特性

- 最短指令执行时间可以从高速（0.05 μ s： @ 20 MHz，使用高速系统时钟工作）变为超低速（61 μ s： @ 32.768 kHz，使用副系统时钟工作）
- 通用寄存器：8 位 $\times$ 32 寄存器（8 位 $\times$ 8 寄存器 $\times$ 4 组）
- ROM、RAM 容量

Flash ROM	RAM	78K0R/KC3-L		78K0R/KD3-L	78K0R/KE3-L
		44 引脚	48 引脚	52 引脚	64 引脚
64 KB	3 KB ^{‡1}	μ PD78F1003 ^{‡2}		μ PD78F1006 ^{‡2}	μ PD78F1009 ^{‡2}
48 KB	2 KB	μ PD78F1002 ^{‡2}		μ PD78F1005 ^{‡2}	μ PD78F1008 ^{‡2}
32 KB	1.5 KB	μ PD78F1001 ^{‡2}		μ PD78F1004 ^{‡2}	μ PD78F1007 ^{‡2}
16 KB	1 KB	μ PD78F1000 ^{‡2}	—	—	—

注 1. 当使用自编程功能时为 2 KB。

2. 开发中

- 片上内部高速振荡时钟
 - 20 MHz 内部高速振荡时钟：20 MHz $\pm$ 1 %（目标值）
 - 8 MHz 内部高速振荡时钟：8 MHz $\pm$ 1 %（目标值）
 - 1 MHz 内部高速振荡时钟：1 MHz $\pm$ 5 %
- 片上单供电电源闪存（具有禁止芯片擦除/块擦除/写功能）
- 自编程（具有引导区互换功能/Flash 屏蔽窗口功能）
- 片上调试功能
- 片上上电清除（POC）电路和低电压检测器（LVI）
- 片上看门狗定时器（使用专用的内部低速振荡时钟工作）
- 片上乘法器/除法器（16 位 $\times$ 16 位，32 位 $\div$ 32 位）
- 片上按键中断功能
- 片上时钟输出/蜂鸣器输出控制器[‡]
- 片上 BCD 调整
- I/O 端口：37 至 55（N 沟道漏极开路：2[‡]）
- 定时器：10 通道
 - 16 位定时器：8 通道
 - 看门狗定时器：1 通道
 - 实时计数器：1 通道
- 片上比较器/可编程增益放大器功能
- 串行接口
 - CSI：2 通道/UART（受支持的 LIN 总线）：1 通道
 - CSI：1 通道/UART：1 通道/简易的 I²C：1 通道
 - I²C：1 通道[‡]
- 10 位分辨率 A/D 转换器（AV_{REF} = 1.8 至 5.5 V）：10 至 12 通道
- 电源供电电压：V_{DD} = 1.8 至 5.5 V
- 工作环境温度：T_A = -40 至 +85°C

注 78K0R/KC3-L 的 44-引脚产品未配备。

1.2 应用

- 音视频设备
- 家用电器
- 工业设备

1.3 订货信息

- 闪存版本（无铅产品）

78K0R/Kx3 微控制器	封装	型号
78K0R/KC3-L	44-引脚LQFP塑料封装 (10 × 10)	μ PD78F1000GB-GAF-AX ^註 、78F1001GB-GAF-AX ^註 、78F1002GB-GAF-AX ^註 、78F1003GB-GAF-AX ^註
	48-引脚LQFP塑料封装(密间距) (7 × 7)	μ PD78F1001GA-HAA-AX ^註 、78F1002GA-HAA-AX ^註 、78F1003GA-HAA-AX ^註
78K0R/KD3-L	52-引脚LQFP塑料封装(10 × 10)	μ PD78F1004GB-GAG-AX ^註 、78F1005GB-GAG-AX ^註 、78F1006GB-GAG-AX ^註
78K0R/KE3-L	64-引脚LQFP塑料封装(12 × 12)	μ PD78F1007GK-GAJ-AX ^註 、8F1008GK-GAJ-AX ^註 、78F1009GK-GAJ-AX ^註
	64-引脚LQFP塑料封装(密间距) (10 × 10)	μ PD78F1007GB-GAH-AX ^註 、78F1008GB-GAH-AX ^註 、78F1009GB-GAH-AX ^註
	64-引脚LQFP塑料封装(密间距) (7 × 7)	μ PD78F1007GA-HAB-AX ^註 、78F1008GA-HAB-AX ^註 、78F1009GA-HAB-AX ^註
	64-引脚FBGA塑料封装 (5 × 5)	μ PD78F1007F1-AN1-A ^註 、78F1008F1-AN1-A ^註 、78F1009F1-AN1-A ^註

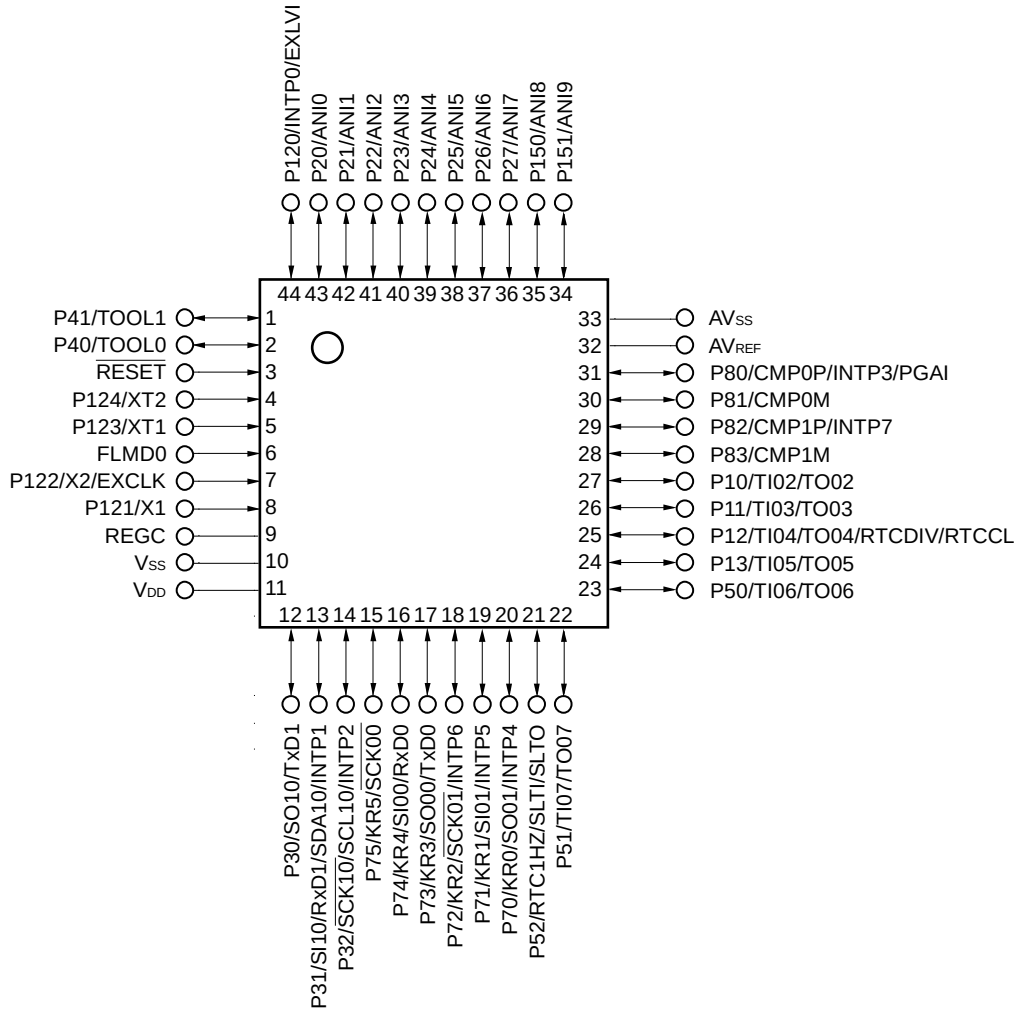
注 开发中

注意事项 78K0R/Kx3-L 具有片上调试功能，用于开发和评估。请不要在量产时使用该产品的片上调试功能，因为使用该功能时，可能会超出 flash 内存能保证的可重写次数，因此产品的可靠性得不到保证。NEC 电子对使用片上调试功能时产生的问题概不负责。

1.4 引脚配置（顶视图）

1.4.1 78K0R/KC3-L

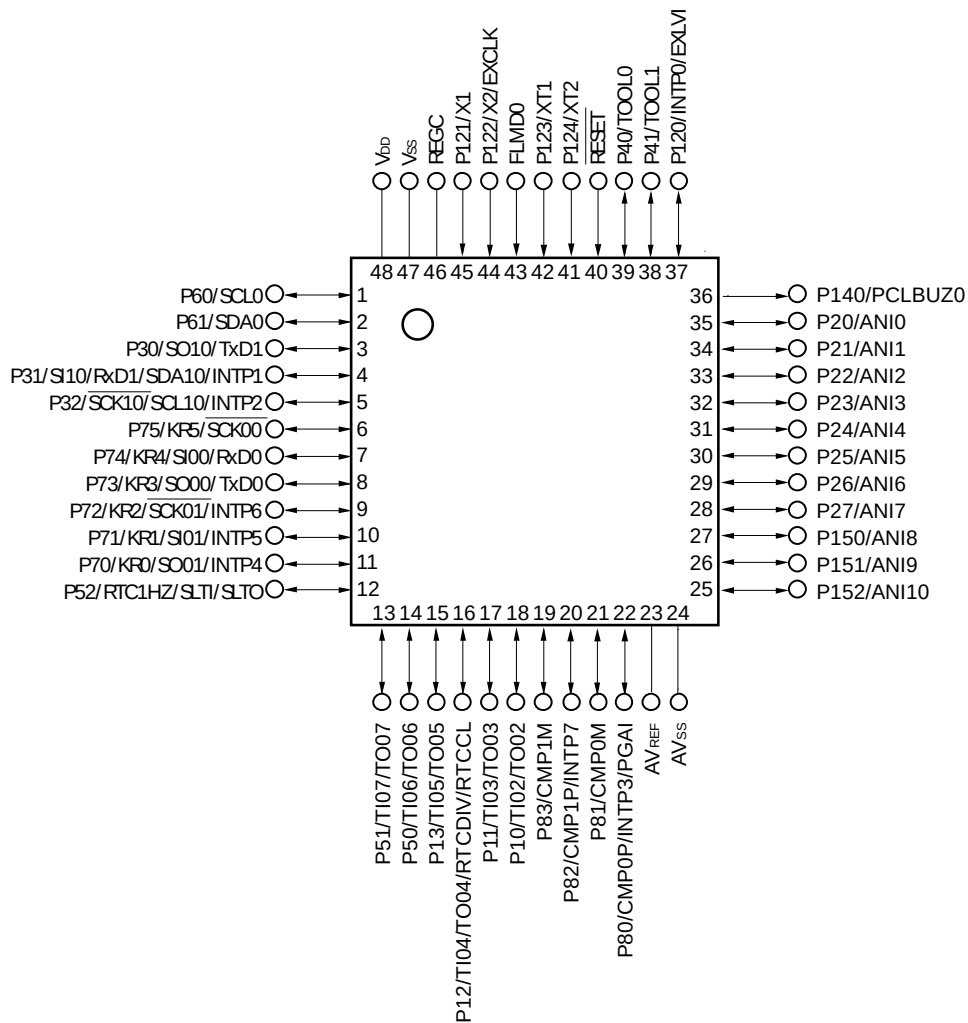
- 44-引脚 LQFP 塑料封装（10 × 10）



- 注意事项**
1. 使 AV_{SS} 和 V_{SS} 保持相同电平。
 2. 通过电容（0.47 至 1 μ F：目标）将 REGC 引脚连接到 V_{SS}。

备注 有关引脚识别，参见 1.5 引脚识别。

- 48-引脚 TQFP 塑料封装（密间距）（7 × 7）



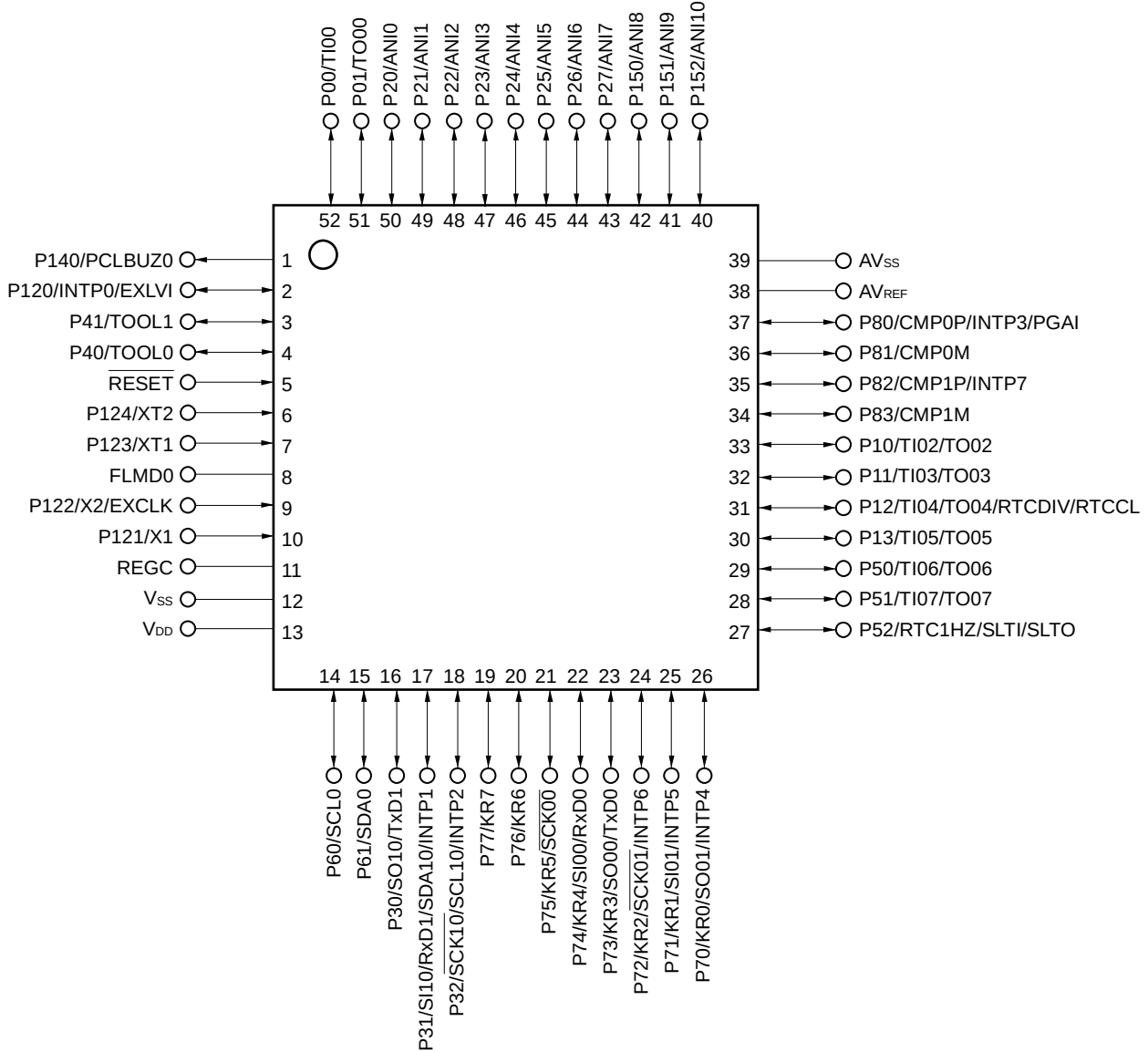
注意事项 1. 使 AVSS 和 VSS 保持相同电平。

2. 通过电容（0.47 至 1 μF ：目标）将 REGC 引脚连接到 VSS。

备注 有关引脚识别，参见 1.5 引脚识别。

1.4.2 78K0R/KD3-L

- 52-引脚 LQFP 塑料封装 (10 × 10)

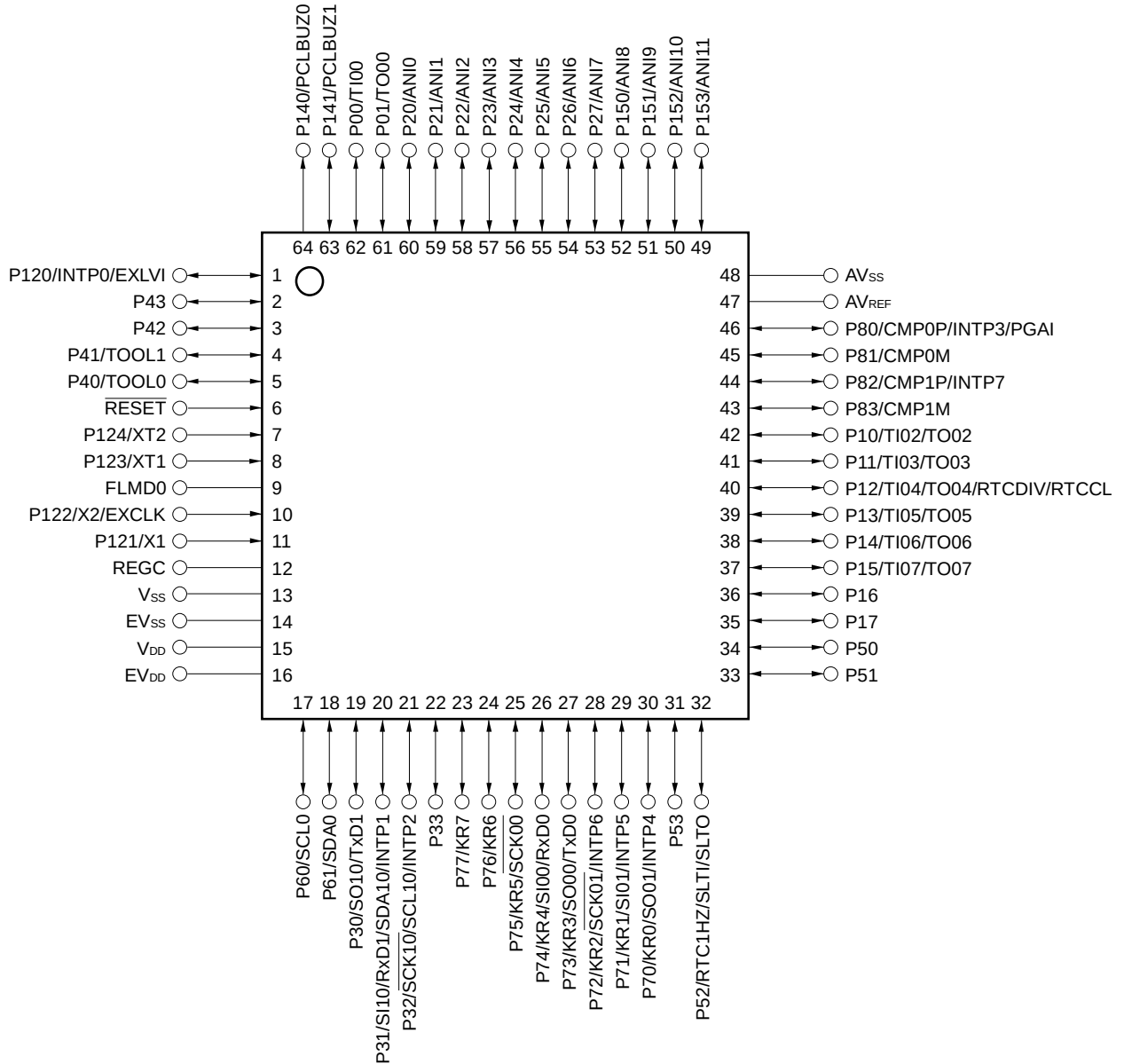


- 注意事项**
1. 使 AV_{SS} 和 V_{SS} 保持相同电平。
 2. 通过电容 (0.47 至 1 μ F: 目标) 将 REGC 引脚连接到 V_{SS}。

备注 有关引脚识别, 参见 1.5 引脚识别。

1.4.3 78K0R/KE3-L

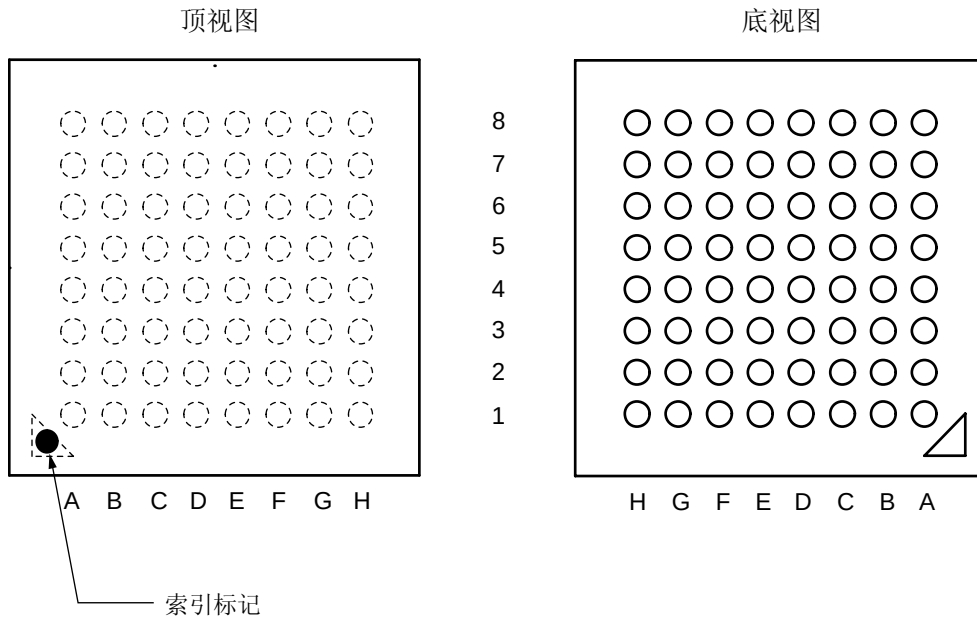
- 64-引脚 LQFP 塑料封装 (12 × 12)
- 64-引脚 LQFP 塑料封装 (密间距) (10 × 10)
- 64-引脚 LQFP 塑料封装 (密间距) (7 × 7)



- 注意事项**
1. 使 AV_{SS} 和 EV_{SS} 保持与 V_{SS} 相同的电平。
 2. 使 EV_{DD} 和 V_{DD} 保持相同电平。
 3. 通过电容 (0.47 至 1 μ F; 目标) 将 REGC 引脚连接到 V_{SS}。

备注 有关引脚识别, 参见 1.5 引脚识别。

• 64-引脚 FBGA 塑料封装 (5 × 5)



引脚号	名称	引脚号	名称	引脚号	名称	引脚号	名称
A1	P53	C1	P50	E1	P83/CMP1M	G1	AV _{REF}
A2	P52/RTC1HZ/SLTI/SLTO	C2	P71/KR1/SI01/INTP5	E2	P12/TI04/TO04/RTCDIV/RTCCL	G2	P151/ANI9
A3	P72/KR2/SCK01/INTP6	C3	P74/KR4/SI00/RxD0	E3	P11/TI03/TO03	G3	P150/ANI8
A4	P75/KR5/SCK00	C4	P17	E4	P10/TI02/TO02	G4	P26/ANI6
A5	P77/KR7	C5	P33	E5	P21/ANI1	G5	P23/ANI3
A6	P61/SDA0	C6	P31/SI10/RxD1/SDA10/INTP1	E6	P41/TOOL1	G6	P20/ANI0
A7	P60/SCL0	C7	V _{SS}	E7	RESET	G7	P00/TI00
A8	EV _{DD}	C8	P121/X1	E8	FLMD0	G8	P124/XT2
B1	P51	D1	P15/TI07/TO07	F1	P80/CMP0P/INTP3/PGAI	H1	AV _{SS}
B2	P70/KR0/SO01/INTP4	D2	P14/TI06/TO06	F2	P81/CMP0M	H2	P153/ANI11
B3	P73/KR3/SO00/TxD0	D3	P13/TI05/TO05	F3	P82/CMP1P/INTP7	H3	P152/ANI10
B4	P76/KR6	D4	P16	F4	P25/ANI5	H4	P27/ANI7
B5	P32/SCK10/SCL10/INTP2	D5	P42	F5	P22/ANI2	H5	P24/ANI4
B6	P30/SO10/TxD1	D6	P40/TOOL0	F6	P43	H6	P141/PCLBUZ1
B7	V _{DD}	D7	REGC	F7	P01/TO00	H7	P140/PCLBUZ0
B8	EV _{SS}	D8	P122/X2/EXCLK	F8	P123/XT1	H8	P120/INTP0/EXLVI

注 开发中

- 注意事项**
1. 使 AV_{SS} 和 EV_{SS} 保持与 V_{SS} 相同的电平。
 2. 使 EV_{DD} 和 V_{DD} 保持相同电平。
 3. 通过电容 (0.47 至 1 μ F; 目标) 将 REGC 引脚连接到 V_{SS}。

备注 有关引脚识别, 参见 1.5 引脚识别。

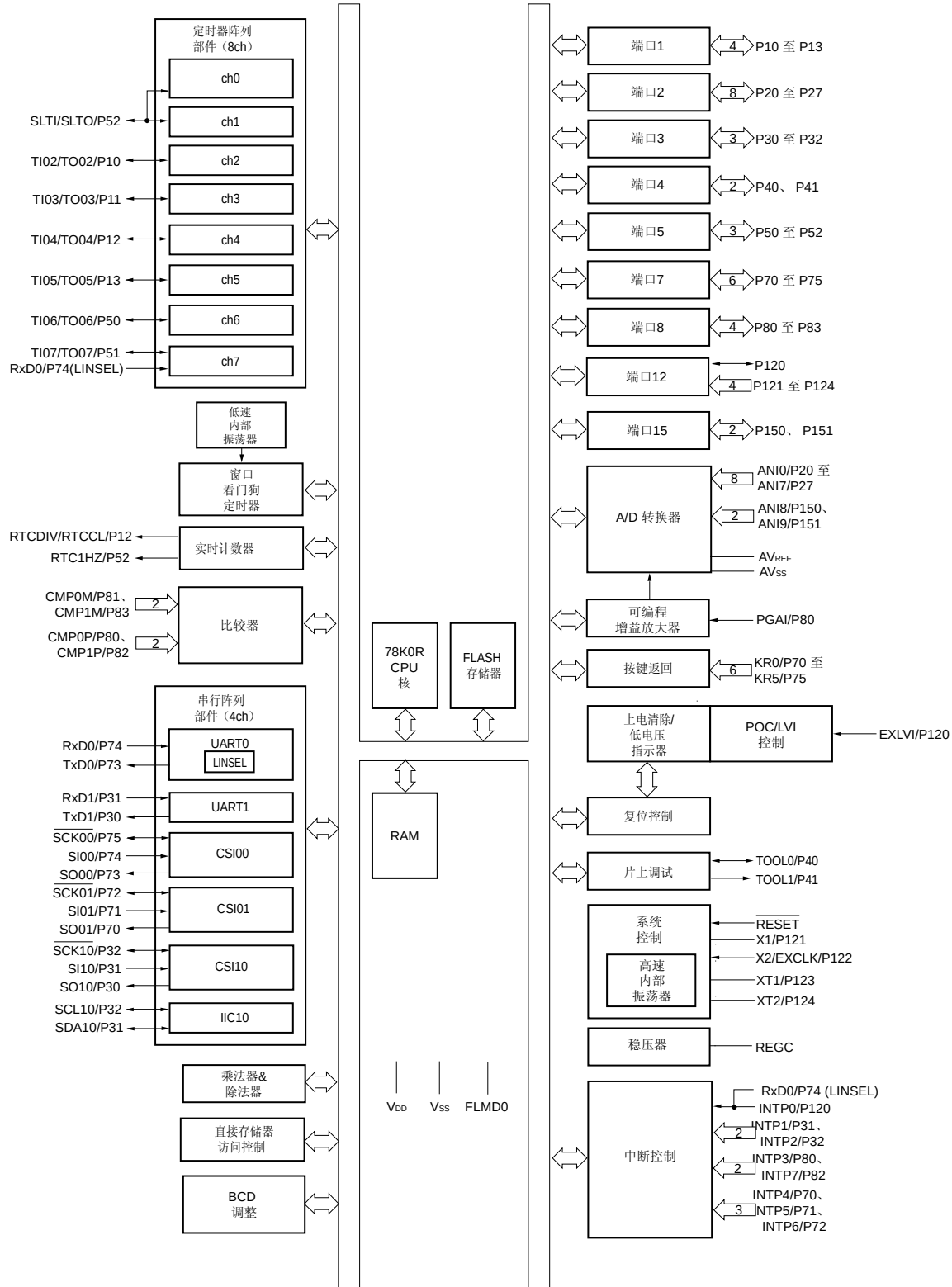
1.5 引脚识别

ANIO 至 ANI11:	模拟输入	PCLBUZ0、PCLBUZ1:	可编程的时钟输出/蜂鸣器输出
AVREF:	模拟参考电压	PGAI:	可编程增益放大器输入
AVSS :	模拟接地	REGC:	稳压器电容
CMP0M、CMP1M:	比较器输入（负的）	RESET:	复位
CMP0P、CMP1P:	比较器输入（正的）	RTC1HZ:	实时计数器校正时钟（1 Hz）输出
EVDD:	供应端口的电源	RTCCL:	实时计数器时钟（32 kHz 原始振荡）输出
EVSS:	端口接地	RTCDIV:	实时计数器时钟（32 kHz 分频）输出
EXCLK:	外部时钟输入（主系统时钟）	RxD0、RxD1:	接收数据
EXLVI:	低电压检测器的外部电位输入	SCK00、SCK01、SCK10:	串行时钟输入/输出
FLMD0:	Flash 可编程模式	SCL0、SCL10:	串行时钟输入/输出
INTP0 至 INTP7:	外部中断输入	SDA0、SDA10:	串行数据输入/输出
KR0 至 KR7:	按键返回	SI00、SI01、SI10:	串行数据输入
P00、P01:	端口 0	SLTI:	可选的定时器输入
P10 至 P17:	端口 1	SLTO:	可选的定时器输出
P20 至 P27:	端口 2	SO00、SO01、SO10:	串行数据输出
P30 至 P33:	端口 3	TI00、TI02 至 TI07:	定时器输出
P40 至 P43:	端口 4	TO00、TO02 至 TO07:	定时器输出
P50 至 P53:	端口 5	TOOL0:	工具的数据输入/输出
P60、P61:	端口 6	TOOL1:	工具的时钟输出
P70 至 P77:	端口 7	TxD0、TxD1:	发送数据
P80 至 P83:	端口 8	VDD:	供电电源
P120 至 P124:	端口 12	VSS:	接地
P140、P141:	端口 14	X1、X2:	晶体振荡器（主系统时钟）
P150 至 P153:	端口 15	XT1、XT2:	晶体振荡器（副系统时钟）

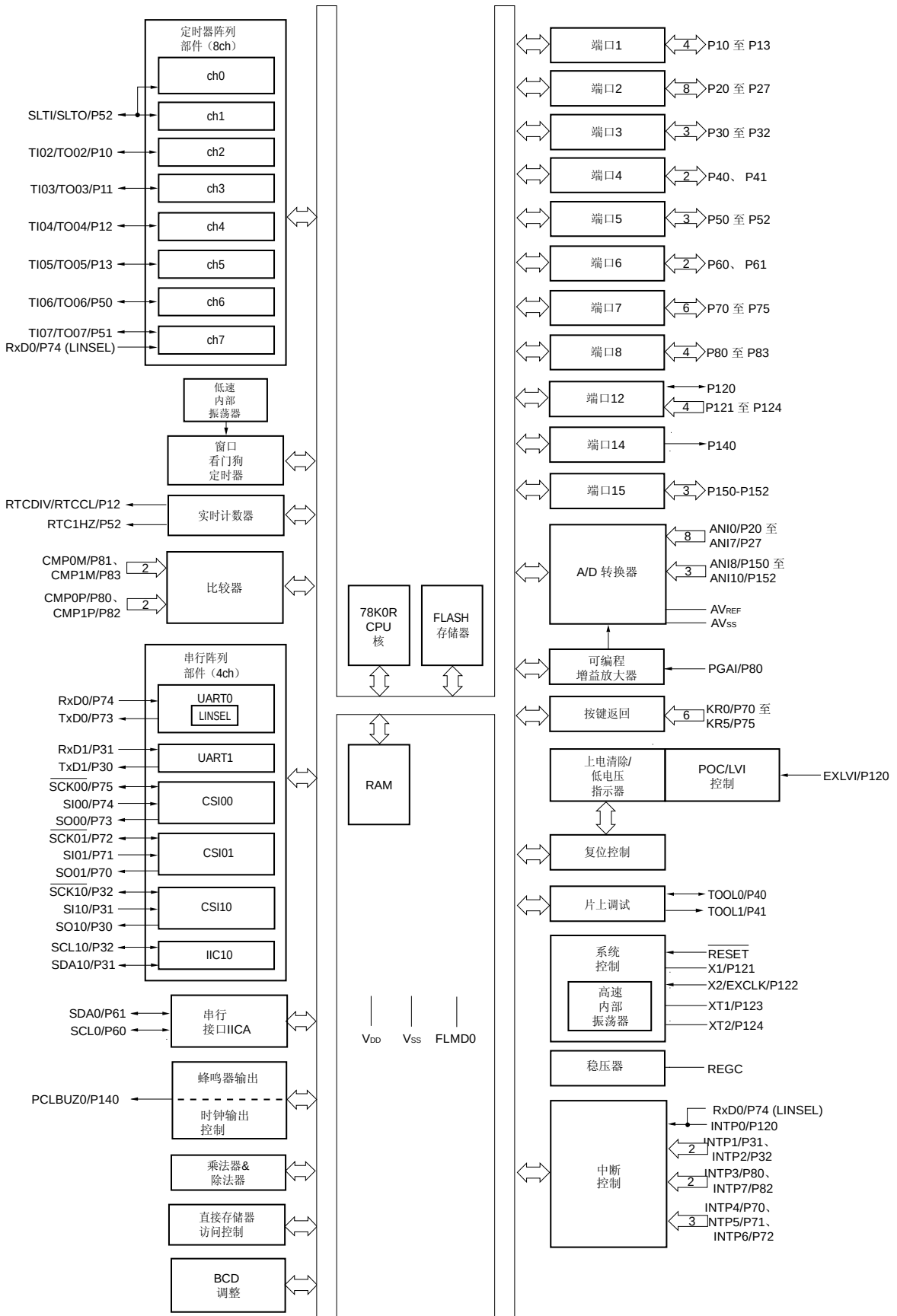
1.6 程序块框图

1.6.1 78K0R/KC3-L

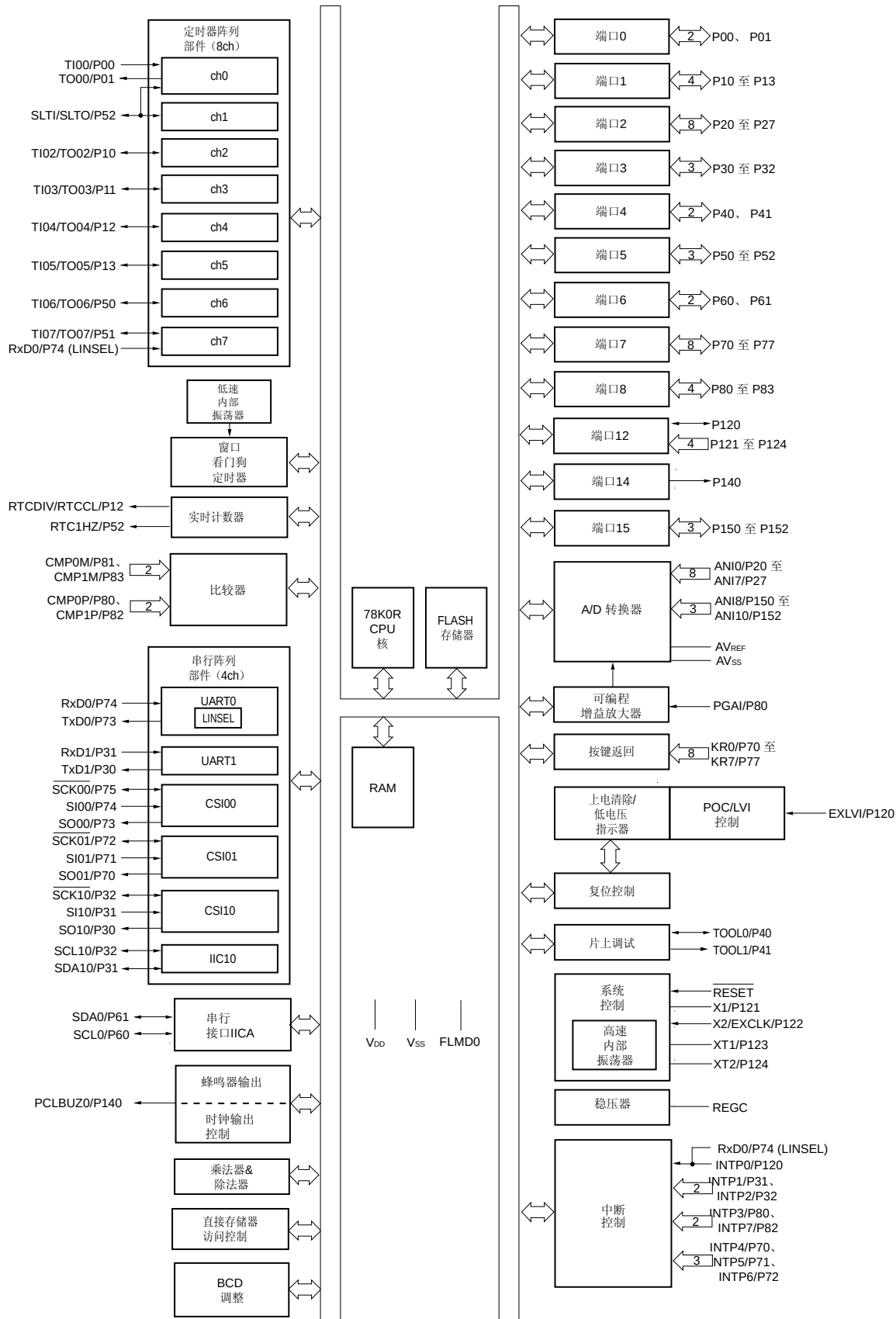
• 44-引脚产品



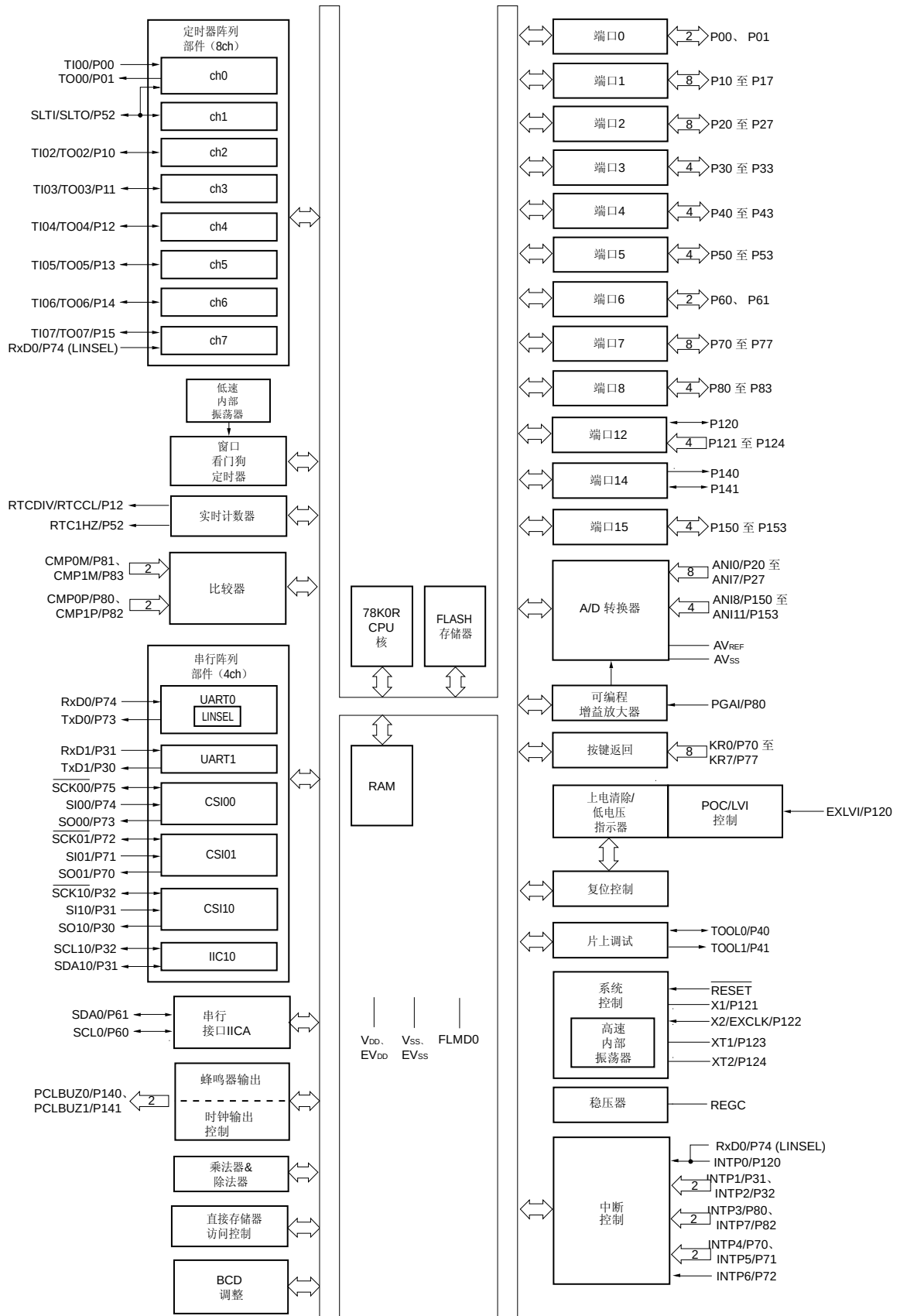
• 48-引脚产品



1.6.2 78K0R/KD3-L



1.6.3 78K0R/KE3-L



1.7 功能概述

(1/2)

项目		78K0R/KC3-L (44-引脚) (μ PD78F100 ^{註1} 、y=0至3)				78K0R/KC3-L (48-引脚) (μ PD78F100 ^{註1} 、y=1至3)			78K0R/KD3-L (μ PD78F100 ^{註1} 、y=4至6)			78K0R/KE3-L (μ PD78F100 ^{註1} 、y=7至9)			
内部存储器	Flash 存储器 (KB)	16	32	48	64	32	48	64	32	48	64	32	48	64	
	RAM (KB)	1	1.5	2	3/2 註2	1.5	2	3/2 註2	1.5	2	3/2 註2	1.5	2	3/2 註2	
存储空间		1 MB													
主系统时钟	高速系统时钟	X1 (晶体/陶瓷)振荡、外部主系统时钟输入(EXCLK) 2 至 20 MHz: V _{DD} = 2.7 至 5.5 V、2 至 5 MHz: V _{DD} = 1.8 至 5.5 V													
	内部高速振荡时钟	内部振荡 1 MHz ±5%、8 MHz ±1% (目标): V _{DD} = 1.8 至 5.5 V													
	20 MHz 内部高速振荡时钟	内部振荡 20 MHz ±1% (目标): V _{DD} = 2.7 至 5.5 V													
副系统时钟		XT1 (晶体)振荡 32.768 kHz (TYP.): V _{DD} = 1.8 至 5.5 V													
内部低速振荡时钟(专用于 WDT)		内部振荡 30 kHz (TYP.): V _{DD} = 1.8 至 5.5 V													
通用寄存器		8 位 × 32 寄存器 (8 位 × 8 寄存器 × 4 组)													
最短指令执行时间		0.05 μ S (高速系统时钟: f _{MX} = 20 MHz 工作)													
		61 μ S (副系统时钟: f _{SUB} = 32.768 kHz 工作)													
指令集		8 位操作、16 位操作 - 乘法 (8 位 × 8 位) - 位处理(设置、复位、测试和逻辑操作)等等													
I/O 端口	总数	37				41			45			55			
	CMOS I/O	33				34			38			48			
	CMOS 输入	4				4			4			4			
	CMOS 输出	—				1			1			1			
	N 沟道漏极开路 I/O (6 V 耐压)	—				2			2			2			
定时器		16 位定时器: 8 通道 - 看门狗定时器: 1 通道 - 实时计数器(RTC): 1 通道													
		定时器输出	8 (PWM 输出: 7 ^{註3})												
		RTC 输出	2 1 Hz (副系统时钟: f_{SUB} = 32.768 kHz) 512 Hz、16.384 kHz 或 32.768 kHz (副系统时钟: f_{SUB} = 32.768 kHz)												

- 注
1. 开发中
 2. 使用自编程功能时为 2 KB。
 3. 输出数目根据设置的不同而不同。

(2/2)

项目		78K0R/KC3-L (44-引脚) (μ PD78F100) ^{※1} 、y=0 至 3)	78K0R/KC3-L (48-引脚) (μ PD78F100) ^{※1} 、y=1 至 3)	78K0R/KD3-L (μ PD78F100) ^{※1} 、y=4 至 6)	78K0R/KE3-L (μ PD78F100) ^{※1} 、y=7 至 9)
时钟输出/蜂鸣器输出		—	1	1	2
		• 2.44 kHz、4.88 kHz、9.76 kHz、1.25 MHz、2.5 MHz、5 MHz、10 MHz (外围硬件时钟: f _{MAIN} = 20 MHz 工作) • 256 Hz、512 Hz、1.024 kHz、2.048 kHz、4.096 kHz、8.192 kHz、16.384 kHz、32.768 kHz (副系统时钟: f _{SUB} = 32.768 kHz 工作)			
10 位分辨率 A/D 转换器 (AV _{REF} = 1.8 至 5.5 V)		10 通道	11 通道	11 通道	12 通道
串行接口		• CSI: 2 通道/UART (受支持的LIN总线): 1 通道 • CSI: 1 通道/UART: 1 通道/简易的I ² C: 1 通道			
		I ² C总线	—	1 通道	1 通道
乘法器/除法器		• 16 位 × 16 位= 32 位(乘法) • 32 位÷ 32 位= 32 位(除法)			
DMA 控制器		2 通道			
矢量中断源	内部	24	25	25	25
	外部	9			
按键中断		6 通道 (KR0 至 KR5)		8 通道 (KR0 至 KR7)	
复位		• 由RESET引脚复位 • 看门狗定时器的内部复位 • 上电清除的内部复位 • 低电压检测器的内部复位 • 执行非法指令 ^{※2} 的内部复位			
上电清零电路		• 上电复位: 1.61 ±0.09 V • 下电复位: 1.59 ±0.09 V			
低电压检测器		1.91 V 至 4.22 V (16 级)			
片上调试功能		提供			
电源供电电压		V _{DD} = 1.8 至 5.5 V			
工作环境温度		T _A = -40 至 +85 °C			

- 注
1. 开发中
 2. 该非法指令在执行指令代码 FFH 时产生。
在线仿真器或片上调试仿真器不能产生通过执行非法指令的复位。

第二章 引脚功能

2.1 引脚功能列表

引脚 I/O 缓存器的电源取决于产品。电源和引脚之间的关系如下所示。

表 2-1. 引脚 I/O 缓存器电源 (AV_{REF} , V_{DD})

- 78K0R/KC3-L: 44-引脚塑封 LQFP (10x10)
48 引脚塑封 TQFP (密间距) (7x7)
- 78K0R/KD3-L: 52-引脚塑封 LQFP (10x10)

电源	相应引脚
AV_{REF}	P20 至 P27, P150 至 P152 [※] , P80 至 P83
EV_{DD}	• 除 P20 至 P27, P150 至 P152[※], P80 至 P83 之外的端口引脚 • 除端口引脚之外的引脚

注 78K0R/KC3-L 的 44-引脚产品不具有 P152 引脚。

表 2-2. 引脚 I/O 缓存器电源 (AV_{REF} , EV_{DD} , V_{DD})

- 78K0R/KE3-L: 64-引脚塑封 FBGA (5x5)
64-引脚塑封 TQFP (密间距) (7x7)
64-引脚塑封 LQFP (密间距) (10x10)
64-引脚塑封 LQFP (12x12)

电源	相应引脚
AV_{REF}	P20 至 P27, P150 至 P153, P80 至 P83
EV_{DD}	• 除 P20 至 P27, P150 至 P153, P80 至 P83 以及 P121 至 P124 之外的引脚 • RESET 引脚和 FLMD0 引脚
V_{DD}	• P121 至 P124 • 端口引脚之外的引脚 (除 RESET 引脚和 FLMD0 引脚之外的引脚)

2.1.1 78K0R/KC3-L

(1) 端口功能 (1/2) : 78K0R/KC3-L

功能名称	I/O	功能	复位后	复用功能
P10	I/O	端口 1。 4 位 I/O 端口。 输入/输出可按 1 位为单元指定。 片上上拉电阻的使用可通过软件设置进行指定。	输入端口	TI02/TO02
P11				TI03/TO03
P12				TI04/TO04/ RTCDIV/RTCCL
P13				TI05/TO05
P20 至 P27	I/O	端口 2。 8 位 I/O 端口。 输入/输出可按 1 位为单元指定。	数字输入端口	ANI0 至 ANI7
P30	I/O	端口 3。 3 位 I/O 端口。 P31 和 P32 的输入可设置至 TTL 缓存器。 P30 至 P32 的输出可设置至 N-通道漏极开路输出 (V _{DD} 耐压)。 输入/输出可按 1 位为单元指定。 片上上拉电阻的使用可通过软件设置进行指定。	输入端口	SO10/TxD1
P31				SI10/RxD1/SDA10/ INTP1
P32				SCK10/SCL10/ INTP2
P40 ^{※1}	I/O	端口 4。 2 位 I/O 端口。 输入/输出可按 1 位为单元指定。 片上上拉电阻的使用可通过软件设置进行指定。	输入端口	TOOL0
P41				TOOL1
P50	I/O	端口 5。 3 位 I/O 端口。 输入/输出可按 1 位为单元指定。 片上上拉电阻的使用可通过软件设置进行指定。	输入端口	TI06/TO06
P51				TI07/TO07
P52				RTC1HZ/SLTI/ SLTO
P60 ^{※2}	I/O	端口 6。 2 位 I/O 端口 P60 和 P61 的输出为 N-通道漏极开路输出(6 V 耐压)。 输入/输出可按 1 位为单元指定。	输入端口	SCL0 ^{※2}
P61 ^{※2}				SDA0 ^{※2}
P70	I/O	端口 7。 6-位 I/O 端口。 P71, P72, P74 和 P75 的输入可设置至 TTL 缓存器。 P70, P72, P73 和 P75 的输出可设置至 N-通道漏极开路输出 (V _{DD} 耐压)。 输入/输出可按 1 位为单元指定。 片上上拉电阻的使用可通过软件设置进行指定。	输入端口	KR0/SO01/INTP4
P71				KR1/SI01/INTP5
P72				KR2/SCK01/INTP6
P73				KR3/SO00/TxD0
P74				KR4/SI00/RxD0
P75				KR5/SCK00

- 注
1. 如果通过使用选项字节使能片上调试, 则一定要在外部上拉 P40/TOOL0 引脚。
 2. 仅限于 48 引脚产品。

(1) 端口功能 (2/2) : 78K0R/KC3-L

功能名称	I/O	功能	复位后	复用功能
P80	I/O	端口 8。 4 位 I/O 端口。 输入/输出可以 1 位为单元来指定。 P80 至 P83 的输入可设为比较器输入或可编程增益放大器输入。	模拟输入	CMP0P/INTP3/ PGAI
P81				CMP0M
P82				CMP1P/INTP7
P83				CMP1M
P120	I/O	端口 12。 1 位 I/O 端口和 4 位输入端口。 仅对于 P120, 输入/输出可按位为单元指定。 仅对于 P120, 片上上拉电阻的使用可通过软件设置进行指定。	输入端口	INTP0/EXLVI
P121	输入			X1
P122				X2/EXCLK
P123				XT1
P124				XT2
P140 [‡]	输出	端口 14。 1 位输出端口。	输出端口	PCLBUZ0 [‡]
P150, P151, P152 [‡]	I/O	端口 15。 3 位 I/O 端口。 输入/输出可按 1 位为单元指定。	数字输入端 口	ANI8, ANI9, ANI10 [‡]

注 仅限于 48 引脚产品。

(2) 非端口功能 (1/3) : 78K0R/KC3-L

功能名称	I/O	功能	复位后	复用功能
ANI0 至 ANI7	输入	A/D 转换器模拟输入	模拟输入端口	P20 至 P27
ANI8, ANI9, ANI10 [※]	输入			P150, P151, P152 [※]
CMP0M	输入	在比较器 0 的(-)端的输入电压	模拟输入	P81
CMP0P	输入	在比较器 0 的(+)端的输入电压		P80/INTP3/PGAI
CMP1M	输入	在比较器 1 的(-)端的输入电压		P83
CMP1P	输入	在比较器 1 的(+)端的输入电压		P82/INTP7
EXLVI	输入	外部低电压检测的电压输入	输入端口	P120/INTP0
INTP0	输入	可被指定的有效边沿（上升沿、下降沿或双边沿）的外部中断请求输入。	输入端口	P120/EXLVI
INTP1				P31/SI10/RxD1/ SDA10
INTP2				P32/SCK10/SCL10
INTP3			模拟输入	P80/CMP0P/PGAI
INTP4			输入端口	P70/KR0/SO01
INTP5				P71/KR1/SI01
INTP6				P72/KR2/SCK01
INTP7			模拟输入	P82/CMP1P
KR0	输入	按键中断输入	输入端口	P70/SO01/INTP4
KR1				P71/SI01/INTP5
KR2				P72/SCK01/INTP6
KR3				P73/SO00/TxD0
KR4				P74/SI00/RxD0
KR5				P75/SCK00
PCLBUZ0 [※]	输出	时钟输出/蜂鸣器输出	输出端口	P140 [※]
PGAI	输入	可编程增益放大器输入	模拟输入	P80/CMP0P/INTP3
REGC	—	为内部运算连接稳压器输出（2.4 V）稳定电容。 通过电容连接到 Vss (0.47 至 1 μ F: 目标).	—	—
RTCDIV	输出	实时计数器时钟（32 kHz 分频）输出	输入端口	P12/TI04/TO04/ RTCCL
RTCCL	输出	实时计数器时钟（32 kHz 原始振荡）输出	输入端口	P12/TI04/TO04/ RTCDIV
RTC1HZ	输出	实时计数器校正时钟（1 Hz）输出	输入端口	P52/SLTI/SLTO
RESET	输入	系统复位输入	—	—
RxD0	输入	串行数据输入至 UART0	输入端口	P74/KR4/SI00
RxD1		串行数据输入至 UART1		P31/SI10/SDA10/ INTP1
SCK00	I/O	CSI00 的时钟输入/输出	输入端口	P75/KR5
SCK01		CSI01 的时钟输入/输出		P72/KR2/INTP6
SCK10		CSI10 的时钟输入/输出		P32/SCL10/INTP2

注 仅限于 48 引脚产品。

(2) 非端口功能 (2/3) : 78K0R/KC3-L

功能名称	I/O	功能	复位后	复用功能
SCL0 [※]	I/O	I ² C 的时钟输入/输出	输入端口	P60 [※]
SCL10	I/O	简易的 I ² C 的时钟输入/输出	输入端口	P32/SCK10/INTP2
SDA0 [※]	I/O	I ² C 的串行数据 I/O	输入端口	P61 [※]
SDA10	I/O	简易的 I ² C 的串行数据 I/O	输入端口	P31/SI10/RxD1/ INTP1
SI00	输入	串行数据输入至 CSI00	输入端口	P74/KR4/RxD0
SI01		串行数据输入至 CSI01		P71/KR1/INTP5
SI10		串行数据输入至 CSI10		P31/RxD1/SDA10/ INTP1
SLTI	输入	16 位定时器 00, 01 输入	输入端口	P52/RTC1HZ/SLTO
SLTO	输出	16 位定时器 00, 01 输出	输入端口	P52/RTC1HZ/SLTI
SO00	输出	来自 CSI00 的串行数据输出	输入端口	P73/KR3/TxD0
SO01		来自 CSI01 的串行数据输出		P70/KR0/INTP4
SO10		来自 CSI10 的串行数据输出		P30/TxD1
TI02	输入	外部计数时钟输入至 16 位定时器 02	输入端口	P10/TO02
TI03		外部计数时钟输入至 16 位定时器 03		P11/TO03
TI04		外部计数时钟输入至 16 位定时器 04		P12/TO04/ RTCDIV/RTCCL
TI05		外部计数时钟输入至 16 位定时器 05		P13/TO05
TI06		外部计数时钟输入至 16 位定时器 06		P50/TO06
TI07		外部计数时钟输入至 16 位定时器 07		P51/TO07
TO02	输出	16 位定时器 02 输出	输入端口	P10/TI02
TO03		16 位定时器 03 输出		P11/TI03
TO04		16 位定时器 04 输出		P12/TI04/ RTCDIV/RTCCL
TO05		16 位定时器 05 输出		P13/TI05
TO06		16 位定时器 06 输出		P50/TI06
TO07		16 位定时器 07 输出		P51/TI07
TxD0	输出	来自 UART0 的串行数据输出	输入端口	P73/KR3/SO00
TxD1		来自 UART1 的串行数据输出		P30/SO10
X1	—	主系统时钟的谐振器连接	输入端口	P121
X2	—		输入端口	P122/EXCLK
XT1	—	副系统时钟的谐振器连接	输入端口	P123
XT2	—		输入端口	P124
EXCLK	输入	主系统时钟的外部时钟输入	输入端口	P122/X2
V _{DD}	—	供电电源正极 (除 P20 至 P27, P80 至 P83, P150, P151, P152 [※] 之外的端口引脚和其它端口)	—	—
AV _{REF}	—	- A/D 转换器和比较器的参考电压输入 - P20 至 P27, P150, P151, P152[※], P80 至 P83, A/D 转换器, 可编程增益放大器和比较器的供电电源正极	—	—
V _{SS}	—	接地电压 (除 P20 至 P27, P80 至 P83, P150, P151, P152 [※] 之外的端口引脚和其它端口)	—	—

注 仅限于 48 引脚产品。

(2) 非端口功能（3/3）：78K0R/KC3-L

功能名称	I/O	功能	复位后	复用功能
AVss	—	A/D 转换器，可编程增益放大器，比较器，P20 至 P27，P150，P151，P152 ^注 和 P80 至 P83 的接地电压	—	—
FLMD0	—	Flash 存储器编程模式设置	—	—
TOOL0	I/O	Flash 存储器编程器/调试器的数据 I/O	输入端口	P40
TOOL1	输出	调试器的时钟输出	输入端口	P41

注 仅限于 48 引脚产品。

2.1.2 78K0R/KD3-L

(1) 端口功能 (1/2) : 78K0R/KD3-L

功能名称	I/O	功能	复位后	复用功能
P00	I/O	端口 0。 2 位 I/O 端口。 输入/输出可按 1 位为单元指定。 片上上拉电阻的使用可通过软件设置进行指定。	输入端口	TI00
P01				TO00
P10	I/O	端口 1。 4 位 I/O 端口。 输入/输出可按 1 位为单元指定。 片上上拉电阻的使用可通过软件设置进行指定。	输入端口	TI02/TO02
P11				TI03/TO03
P12				TI04/TO04/ RTCDIV/RTCCL
P13				TI05/TO05
P20 至 P27	I/O	端口 2。 8 位 I/O 端口。 输入/输出可按 1 位为单元指定。	数字输入端口	ANI0 至 ANI7
P30	I/O	端口 3。 3 位 I/O 端口。 P31 和 P32 的输入可设置至 TTL 缓存器。 P30 至 P32 的输出可设置至 N-通道漏极开路输出 (V _{DD} 耐压)。 输入/输出可按 1 位为单元指定。 片上上拉电阻的使用可通过软件设置进行指定。	输入端口	SO10/TxD1
P31				SI10/RxD1/SDA10/ INTP1
P32				SCK10/SCL10/ INTP2
P40 ^注	I/O	端口 4。 2 位 I/O 端口 输入/输出可按 1 位为单元指定。 片上上拉电阻的使用可通过软件设置进行指定。	输入端口	TOOL0
P41				TOOL1
P50	I/O	端口 5。 3 位 I/O 端口。 输入/输出可按 1 位为单元指定。 片上上拉电阻的使用可通过软件设置进行指定。	输入端口	TI06/TO06
P51				TI07/TO07
P52				RTC1HZ/SLTI/ SLTO
P60	I/O	端口 6。 2 位 I/O 端口。 P60 和 P61 的输出为 N-通道漏极开路输出(6 V 耐压)。 输入/输出可按 1 位为单元指定。	输入端口	SCL0
P61				SDA0

注 如果通过使用选项字节使能片上调试, 则一定要在外部上拉 P40/TOOL0 引脚。

(1) 端口功能 (2/2) : 78K0R/KD3-L

功能名称	I/O	功能	复位后	复用功能
P70	I/O	端口 7。 8 位 I/O 端口。 P71, P72, P74 和 P75 的输入可设置至 TTL 缓存器。 P70, P72, P73 和 P75 的输出可设置至 N-通道漏极开路输出（V _{DD} 耐压）。 输入/输出可按 1 位为单元指定。 片上上拉电阻的使用可通过软件设置进行指定。	输入端口	KR0/SO01/INTP4
P71				KR1/SI01/INTP5
P72				KR2/SCK01/INTP6
P73				KR3/SO00/TxD0
P74				KR4/SI00/RxD0
P75				KR5/SCK00
P76				KR6
P77				KR7
P80	I/O	端口 8。 4 位 I/O 端口 输入/输出可以 1 位为单元来指定。 P80 至 P83 的输入可设为比较器输入或可编程增益放大器输入。	模拟输入	CMP0P/INTP3/PGA1
P81				CMP0M
P82				CMP1P/INTP7
P83				CMP1M
P120	I/O	端口 12。 1 位 I/O 端口和 4 位输入端口。 仅对于 P120, 输入/输出可按位为单元指定。 仅对于 P120, 片上上拉电阻的使用可通过软件设置进行指定。	输入端口	INTP0/EXLVI
P121	输入			X1
P122				X2/EXCLK
P123				XT1
P124				XT2
P140	输出	端口 14。 1 位输出端口。	输出端口	PCLBUZ0
P150 至 P152	I/O	端口 15。 3 位 I/O 端口。 输入/输出可按 1 位为单元指定。	数字输入端口	ANI8 至 ANI10

(2) 非端口功能 (1/3) : 78K0R/KD3-L

功能名称	I/O	功能	复位后	复用功能
ANI0 至 ANI7	输入	A/D 转换器和模拟输入	数字输入端口	P20 至 P27
ANI8 至 ANI10	输入			P150 至 P152
CMP0M	输入	在比较器 0 的(-)端的输入电压	模拟输入	P81
CMP0P	输入	在比较器 0 的(+)端的输入电压		P80/INTP3/PGAI
CMP1M	输入	在比较器 1 的(-)端的输入电压		P83
CMP1P	输入	在比较器 1 的(+)端的输入电压		P82/INTP7
EXLVI	输入	外部低电压检测的电压输入	输入端口	P120/INTP0
INTP0	输入	可被指定的有效边沿（上升沿、下降沿或双边沿）的外部中断请求输入。	输入端口	P120/EXLVI
INTP1				P31/SI10/RxD1/ SDA10
INTP2				P32/ $\overline{\text{SCK10}}$ /SCL10
INTP3			模拟输入	P80/CMP0P/PGAI
INTP4			输入端口	P70/KR0/SO01
INTP5				P71/KR1/SI01
INTP6				P72/KR2/ $\overline{\text{SCK01}}$
INTP7			模拟输入	P82/CMP1P
KR0	输入	按键中断输入	输入端口	P70/SO01/INTP4
KR1				P71/SI01/INTP5
KR2				P72/ $\overline{\text{SCK01}}$ /INTP6
KR3				P73/SO00/TxD0
KR4				P74/SI00/RxD0
KR5				P75/ $\overline{\text{SCK00}}$
KR6				P76
KR7				P77
PCLBUZ0	输出	时钟输出/蜂鸣器输出	输出端口	P140
PGAI	输入	可编程增益放大器输入	模拟输入	P80/CMP0P/INTP3
REGC	-	为内部运算连接稳压器输出（2.4 V）稳定电容。 通过电容（0.47 至 1 μF ：目标）连接至 V_{SS} 。	-	-
RTCDIV	输出	实时计数器时钟（32 kHz 分频）输出	输入端口	P12/TI04/TO04/ RTCCL
RTCCL	输出	实时计数器时钟（32 kHz 原始振荡）输出	输入端口	P12/TI04/TO04/ RTCDIV
RTC1HZ	输出	实时计数器校正时钟（1 Hz）输出	输入端口	P52/SLTI/SLTO
RESET	输入	系统复位输入	-	-

(2) 非端口功能 (2/3) : 78K0R/KD3-L

功能名称	I/O	功能	复位后	复用功能
RxD0	输入	串行数据输入至 UART0	输入端口	P74/KR4/SI00
RxD1		串行数据输入至 UART1		P31/SI10/SDA10/ INTP1
$\overline{\text{SCK00}}$	I/O	CSI00 的时钟输入/输出	输入端口	P75/KR5
$\overline{\text{SCK01}}$		CSI01 的时钟输入/输出		P72/KR2/INTP6
$\overline{\text{SCK10}}$		CSI10 的时钟输入/输出		P32/SCL10/INTP2
SCL0	I/O	I ² C 的时钟输入/输出	输入端口	P60
SCL10	I/O	简易的 I ² C 的时钟输入/输出	输入端口	P32/ $\overline{\text{SCK10}}$ /INTP2
SDA0	I/O	I ² C 的串行数据 I/O	输入端口	P61
SDA10	I/O	简易的 I ² C 的串行数据 I/O	输入端口	P31/SI10/RxD1/ INTP1
SI00	输入	串行数据输入至 CSI00	输入端口	P74/KR4/RxD0
SI01		串行数据输入至 CSI01		P71/KR1/INTP5
SI10		串行数据输入至 CSI10		P31/RxD1/SDA10/ INTP1
SLTI	输入	16 位定时器 00, 01 输入	输入端口	P52/RTC1HZ/SLTO
SLTO	输出	16 位定时器 00, 01 输出	输入端口	P52/RTC1HZ/SLTI
SO00	输出	来自 CSI00 的串行数据输出	输入端口	P73/KR3/TxD0
SO01		来自 CSI01 的串行数据输出		P70/KR0/INTP4
SO10		来自 CSI10 的串行数据输出		P30/TxD1
TI00	输入	外部计数时钟输入至 16 位定时器 00	输入端口	P00
TI02		外部计数时钟输入至 16 位定时器 02		P10/TO02
TI03		外部计数时钟输入至 16 位定时器 03		P11/TO03
TI04		外部计数时钟输入至 16 位定时器 04		P12/TO04/ RTCDIV/RTCCL
TI05		外部计数时钟输入至 16 位定时器 05		P13/TO05
TI06		外部计数时钟输入至 16 位定时器 06		P50/TO06
TI07		外部计数时钟输入至 16 位定时器 07		P51/TO07
TO00	输出	16 位定时器 00 输出	输入端口	P01
TO02		16 位定时器 02 输出		P10/TI02
TO03		16 位定时器 03 输出		P11/TI03
TO04		16 位定时器 04 输出		P12/TI04/ RTCDIV/RTCCL
TO05		16 位定时器 05 输出		P13/TI05
TO06		16 位定时器 06 输出		P50/TI06
TO07		16 位定时器 07 输出		P51/TI07
TxD0	输出	来自 UART0 的串行数据输出	输入端口	P73/KR3/SO00
TxD1		来自 UART1 的串行数据输出		P30/SO10
X1	—	主系统时钟的谐振器连接	输入端口	P121
X2	—		输入端口	P122/EXCLK
XT1	—	副系统时钟的谐振器连接	输入端口	P123
XT2	—		输入端口	P124
EXCLK	输入	主系统时钟的外部时钟输入	输入端口	P122/X2

(2) 非端口功能 (3/3) : 78K0R/KD3-L

功能名称	I/O	功能	复位后	复用功能
V _{DD}	—	供电电源正极（除 P20 至 P27，P80 至 P83，P150 至 P152 之外的端口引脚和其它端口）	—	—
AV _{REF}	—	- A/D 转换器和比较器的参考电压输入 - P20 至 P27，P150，P151 至 P152，P80 至 P83，A/D 转换器，可编程增益放大器和比较器的供电电源正极	—	—
V _{SS}	—	接地电压（除 P20 至 P27，P80 至 P83，P150 至 P152 之外的端口引脚和其它端口）	—	—
AV _{SS}	—	A/D 转换器，可编程增益放大器，比较器，P20 至 P27，P150，P151 至 P152 和 P80 至 P83 的接地电压	—	—
FLMD0	—	Flash 存储器编程模式设置	—	—
TOOL0	I/O	Flash 存储器编程器/调试器的数据 I/O	输入端口	P40
TOOL1	输出	调试器的时钟输出	输入端口	P41

2.1.3 78K0R/KE3-L

(1) 端口功能 (1/2) : 78K0R/KE3-L

功能名称	I/O	功能	复位后	复用功能
P00	I/O	端口 0。 2 位 I/O 端口。 输入/输出可按 1 位为单元指定。 片上上拉电阻的使用可通过软件设置进行指定。	输入端口	TI00
P01				TO00
P10	I/O	端口 1。 8 位 I/O 端口。 输入/输出可按 1 位为单元指定。 片上上拉电阻的使用可通过软件设置进行指定。	输入端口	TI02/TO02
P11				TI03/TO03
P12				TI04/TO04/ RTCDIV/RTCCL
P13				TI05/TO05
P14				TI06/TO06
P15				TI07/TO07
P16				—
P17				—
P20 至 P27	I/O	端口 2。 8 位 I/O 端口。 输入/输出可按 1 位为单元指定。	数字输入端口	ANI0 至 ANI7
P30	I/O	端口 3。 4 位 I/O 端口。 P31 和 P32 的输入可设置至 TTL 缓存器。 P30 至 P32 的输出可设置至 N-通道漏极开路输出 (V _{DD} 耐压)。 输入/输出可按 1 位为单元指定。 片上上拉电阻的使用可通过软件设置进行指定。	输入端口	SO10/TxD1
P31				SI10/RxD1/SDA10/ INTP1
P32				SCK10/SCL10/ INTP2
P33				—
P40 [※]	I/O	端口 4。 4 位 I/O 端口。 输入/输出可按 1 位为单元指定。 片上上拉电阻的使用可通过软件设置进行指定。	输入端口	TOOL0
P41				TOOL1
P42				—
P43				—
P50	I/O	端口 5。 4 位 I/O 端口。 输入/输出可按 1 位为单元指定。 片上上拉电阻的使用可通过软件设置进行指定。	输入端口	—
P51				—
P52				RTC1HZ/SLTI/ SLTO
P53				—
P60	I/O	端口 6。 2 位 I/O 端口。 P60 和 P61 的输出为 N-通道漏极开路输出(6 V 耐压)。 输入/输出可按 1 位为单元指定。	输入端口	SCL0
P61				SDA0

注 如果通过使用选项字节使能片上调试，则一定要在外部上拉 P40/TOOL0 引脚。

(1) 端口功能 (2/2) : 78K0R/KE3-L

功能名称	I/O	功能	复位后	复用功能
P70	I/O	端口 7。 8 位 I/O 端口。 P71, P72, P74 和 P75 的输入可设置至 TTL 缓存器。 P70, P72, P73 和 P75 的输出可设置至 N-通道漏极开路输出（V _{DD} 耐压）。 输入/输出可按 1 位为单元指定。 片上上拉电阻的使用可通过软件设置进行指定。	输入端口	KR0/SO01/INTP4
P71				KR1/SI01/INTP5
P72				KR2/SCK01/INTP6
P73				KR3/SO00/TxD0
P74				KR4/SI00/RxD0
P75				KR5/SCK00
P76				KR6
P77				KR7
P80	I/O	端口 8。 4 位 I/O 端口。 输入/输出可以 1 位为单元来指定。 P80 至 P83 的输入可设为比较器输入或可编程增益放大器输入。	模拟输入	CMP0P/INTP3/ PGAI
P81				CMP0M
P82				CMP1P/INTP7
P83				CMP1M
P120	I/O	端口 12。 1 位 I/O 端口和 4 位输入端口。 仅对于 P120, 输入/输出可按位为单元指定。 仅对于 P120, 片上上拉电阻的使用可通过软件设置进行指定。	输入端口	INTP0/EXLVI
P121	输入			X1
P122				X2/EXCLK
P123				XT1
P124				XT2
P140	输出	端口 14。	输出端口	PCLBUZ0
P141	I/O	1 位输出端口和 1 位 I/O 端口。 仅对 P141, 可指定输入/输出。 仅对于 P141, 片上上拉电阻的使用可通过软件设置进行指定。	输入端口	PCLBUZ1
P150 至 P153	I/O	端口 15。 4 位 I/O 端口。 输入/输出可按 1 位为单元指定。	数字输入端 口	ANI8 至 ANI11

(2) 非端口功能 (1/3) : 78K0R/KE3-L

功能名称	I/O	功能	复位后	复用功能
ANI0 至 ANI7	输入	A/D 转换器和模拟输入	数字输入端口	P20 至 P27
ANI8 至 ANI11	输入			P150 至 P153
CMP0M	输入	在比较器 0 的(-)端的输入电压	模拟输入	P81
CMP0P	输入	在比较器 0 的(+)端的输入电压		P80/INTP3/PGAI
CMP1M	输入	在比较器 1 的(-)端的输入电压		P83
CMP1P	输入	在比较器 1 的(+)端的输入电压		P82/INTP7
EXLVI	输入	外部低电压检测的电压输入	输入端口	P120/INTP0
INTP0	输入	可被指定的有效边沿（上升沿、下降沿或双边沿）的外部中断请求输入。	输入端口	P120/EXLVI
INTP1				P31/SI10/RxD1/ SDA10
INTP2				P32/ $\overline{\text{SCK10}}$ /SCL10
INTP3			模拟输入	P80/CMP0P/PGAI
INTP4			输入端口	P70/KR0/SO01
INTP5				P71/KR1/SI01
INTP6				P72/KR2/ $\overline{\text{SCK01}}$
INTP7			模拟输入	P82/CMP1P
KR0	输入	按键中断输入	输入端口	P70/SO01/INTP4
KR1				P71/SI01/INTP5
KR2				P72/ $\overline{\text{SCK01}}$ /INTP6
KR3				P73/SO00/TxD0
KR4				P74/SI00/RxD0
KR5				P75/ $\overline{\text{SCK00}}$
KR6				P76
KR7				P77
PCLBUZ0	输出	时钟输出/蜂鸣器输出	输出端口	P140
PCLBUZ1			输入端口	P141
PGAI	输入	可编程增益放大器输入	模拟输入	P80/CMP0P/INTP3
REGC	—	为内部运算连接稳压器输出（2.4 V）稳定电容。 通过电容（0.47 至 1 μF ：目标）连接至 V_{SS} 。	—	—
RTCDIV	输出	实时计数器时钟（32 kHz 分频）输出	输入端口	P12/TI04/TO04/ RTCCL
RTCCL	输出	实时计数器时钟（32 kHz 原始振荡）输出	输入端口	P12/TI04/TO04/ RTCDIV
RTC1HZ	输出	实时计数器校正时钟（1 Hz）输出	输入端口	P52/SLTI/SLTO
$\overline{\text{RESET}}$	输入	系统复位输入	—	—

(2) 非端口功能 (2/3) : 78K0R/KE3-L

功能名称	I/O	功能	复位后	复用功能
RxD0	输入	串行数据输入至 UART0	输入端口	P74/KR4/SI00
RxD1		串行数据输入至 UART1		P31/SI10/SDA10/ INTP1
$\overline{\text{SCK00}}$	I/O	CSI00 的时钟输入/输出	输入端口	P75/KR5
$\overline{\text{SCK01}}$		CSI01 的时钟输入/输出		P72/KR2/INTP6
$\overline{\text{SCK10}}$		CSI10 的时钟输入/输出		P32/SCL10/INTP2
SCL0	I/O	I ² C 的时钟输入/输出	输入端口	P60
SCL10	I/O	简易的 I ² C 的时钟输入/输出	输入端口	P32/SCK10/INTP2
SDA0	I/O	I ² C 的串行数据 I/O	输入端口	P61
SDA10	I/O	简易的 I ² C 的串行数据 I/O	输入端口	P31/SI10/RxD1/ INTP1
SI00	输入	串行数据输入至 CSI00	输入端口	P74/KR4/RxD0
SI01		串行数据输入至 CSI01		P71/KR1/INTP5
SI10		串行数据输入至 CSI10		P31/RxD1/SDA10/ INTP1
SLTI	输入	16 位定时器 00, 01 输入	输入端口	P52/RTC1HZ/SLTO
SLTO	输出	16 位定时器 00, 01 输出	输入端口	P52/RTC1HZ/SLTI
SO00	输出	来自 CSI00 的串行数据输出	输入端口	P73/KR3/TxD0
SO01		来自 CSI01 的串行数据输出		P70/KR0/INTP4
SO10		来自 CSI10 的串行数据输出		P30/TxD1
TI00	输入	外部计数时钟输入至 16 位定时器 00	输入端口	P00
TI02		外部计数时钟输入至 16 位定时器 02		P10/TO02
TI03		外部计数时钟输入至 16 位定时器 03		P11/TO03
TI04		外部计数时钟输入至 16 位定时器 04		P12/TO04/ RTCDIV/RTCCL
TI05		外部计数时钟输入至 16 位定时器 05		P13/TO05
TI06		外部计数时钟输入至 16 位定时器 06		P14/TO06
TI07		外部计数时钟输入至 16 位定时器 07		P15/TO07
TO00	输出	16 位定时器 00 输出	输入端口	P01
TO02		16 位定时器 02 输出		P10/TI02
TO03		16 位定时器 03 输出		P11/TI03
TO04		16 位定时器 04 输出		P12/TI04/ RTCDIV/RTCCL
TO05		16 位定时器 05 输出		P13/TI05
TO06		16 位定时器 06 输出		P14/TI06
TO07		16 位定时器 07 输出		P15/TI07
TxD0	输出	来自 UART0 的串行数据输出	输入端口	P73/KR3/SO00
TxD1		来自 UART1 的串行数据输出		P30/SO10
X1	—	主系统时钟的谐振器连接	输入端口	P121
X2	—		输入端口	P122/EXCLK
XT1	—	副系统时钟的谐振器连接	输入端口	P123
XT2	—		输入端口	P124
EXCLK	输入	主系统时钟的外部时钟输入	输入端口	P122/X2

(2) 非端口功能 (3/3) : 78K0R/KE3-L

功能名称	I/O	功能	复位后	复用功能
V _{DD}	—	供电电源正极 (P121 至 P124 和端口之外的引脚 (除 $\overline{\text{RESET}}$ 引脚和 FLMD0 引脚之外))	—	—
EV _{DD}	—	端口供电电源正极 (除 P20 至 P27, P150 至 P153, P80 至 P83 和 P121 至 P124 之外), $\overline{\text{RESET}}$ 引脚和 FLMD0 引脚	—	—
AV _{REF}	—	- A/D 转换器和比较器的参考电压输入 - P20 至 P27, P150 至 P153, P80 至 P83, A/D 转换器, 可编程增益放大器和比较器的供电电源正极	—	—
V _{SS}	—	接地电压 (P121 至 P124 和端口之外的引脚 (除 $\overline{\text{RESET}}$ 引脚和 FLMD0 引脚之外))	—	—
EV _{SS}	—	端口接地电压 (除 P20 至 P27, P150 至 P153 和 P121 至 P124 之外), $\overline{\text{RESET}}$ 引脚和 FLMD0 引脚	—	—
AV _{SS}	—	A/D 转换器, 可编程增益放大器, 比较器, P20 至 P27, P150 至 P153 以及 P80 至 P83 的接地电压	—	—
FLMD0	—	Flash 存储器编程模式设置	—	—
TOOL0	I/O	Flash 存储器编程器/调试器的数据 I/O	输入端口	P40
TOOL1	输出	调试器的时钟输出	输入端口	P41

2.2 引脚功能说明

备注 根据产品配置这些引脚。 参见 **1.4 引脚配置（顶视图）** 和 **2.1 引脚功能列表**。

2.2.1 P00, P01（端口 0）

P00 和 P01 用作 I/O 端口。 这些引脚也可用作定时器 I/O。

	78K0R/KC3-L (44-引脚) (μ PD78F100y: y = 0 至 3)	78K0R/KC3-L (48 引脚) (μ PD78F100y: y = 1 至 3)	78K0R/KD3-L (μ PD78F100y: y = 4 至 6)	78K0R/KE3-L (μ PD78F100y: y = 7 至 9)
P00/ TI00	—	—	√	√
P11/TO00	—	—	√	√

备注 √: 已配备

以下工作模式可按 1 位为单元指定。

(1) 端口模式

P00 和 P01 用作 I/O 端口。 P00 至 P01 可用端口模式寄存器 0（PM0）设置为以 1 位为单元的输入或输出端口。片上上拉电阻的使用可通过上拉电阻选项寄存器 0（PU0）指定。

(2) 控制模式

P00 和 P01 用作定时器 I/O。

(a) TI00

该引脚用于向 16-定时器 00 输入外部计数时钟/捕获触发。

(b) TO00

该引脚为 16-定时器 00 的定时器输出引脚。

2.2.2 P10 至 P17（端口 1）

P10 和 P17 用作 I/O 端口。这些引脚还可以用作定时器 I/O 和实时计数器时钟输出。

	78K0R/KC3-L (44-引脚) (μ PD78F100y: y = 0 至 3)	78K0R/KC3-L (48 引脚) (μ PD78F100y: y = 1 至 3)	78K0R/KD3-L (μ PD78F100y: y = 4 至 6)	78K0R/KE3-L (μ PD78F100y: y = 7 至 9)
P10/TI02/TO02	√	√	√	√
P11/TO00/TI03/ TO03	√	√	√	√
P12/TI04/TO04/ RTCDIV/RTCCL	√	√	√	√
P13/TI05/TO05	√	√	√	√
P14/TI06/TO06	— ^注	— ^注	— ^注	√
P15/TI07/TO07	— ^注	— ^注	— ^注	√
P16	—	—	—	√
P17	—	—	—	√

注 在非 78K0R/KE3-L 产品中，TI06/TO06 和 TI07/TO07 分别与 P50 和 P51 复用。

备注 √: 已配备

以下工作模式可按 1 位为单元指定。

(1) 端口模式

P10 和 P17 用作 I/O 端口。P10 至 P17 可用端口模式寄存器 1 (PM1) 设置为以 1 位为单元的输入或输出端口。片上上拉电阻的使用可通过上拉电阻选项寄存器 1 (PU1) 指定。

(2) 控制模式

P10 至 P17 可以用作定时器 I/O 和实时计数器时钟输出。

(a) TI02 至 TI07

这些引脚用于向 16 位定时器 02 至 07 输入外部计数时钟/捕获触发。

(b) TO02 至 TO07

这些引脚为 16 位定时器 02 至 07 的定时器输出引脚。

(c) RTCDIV

该引脚为实时计数器时钟 (32 kHz 分频) 输出引脚。

(d) RTCCL

该引脚为实时计数器时钟 (32 kHz 原始振荡) 输出引脚。

2.2.3 P20 至 P27（端口 2）

P20 和 P27 用作 I/O 端口。 这些引脚还可用作 A/D 转换器模拟输入引脚。

	78K0R/KC3-L (44-引脚) (μ PD78F100y: y = 0 至 3)	78K0R/KC3-L (48 引脚) (μ PD78F100y: y = 1 至 3)	78K0R/KD3-L (μ PD78F100y: y = 4 至 6)	78K0R/KE3-L (μ PD78F100y: y = 7 至 9)
P20/ANI0	√	√	√	√
P21/ANI1	√	√	√	√
P22/ANI2	√	√	√	√
P23/ANI3	√	√	√	√
P24/ANI4	√	√	√	√
P25/ANI5	√	√	√	√
P26/ANI6	√	√	√	√
P27/ANI7	√	√	√	√

备注 √: 已配备

以下工作模式可按 1 位为单元指定。

(1) 端口模式

P20 和 P27 用作 I/O 端口。 P20 至 P27 可用端口模式寄存器 2（PM2）设置为以 1 位为单元的输入或输出端口。

(2) 控制模式

P20 至 P27 用作 A/D 转换器的模拟输入引脚（ANI0 至 ANI7）。 当将这些引脚用作模拟输入引脚时，参见 **11.6 (5) ANI0/P20 至 ANI7/P27, ANI8/P150 至 ANI11/P153。**

注意事项 复位解除之后，在数字输入（通用端口）模式下设置 ANI0/P20 至 ANI7/P27 引脚。

2.2.4 P30 至 P33（端口 3）

P30 和 P33 用作 I/O 端口。 这些引脚还可用作串行接口数据 I/O，时钟 I/O 以及外部中断请求的输入。

P30 和 P31 引脚的输入可通过标准输入缓存器或 TTL 输入缓存器按 1 位为单位进行指定，指定时使用端口输入模式寄存器 3（PIM3）。

P30 至 P32 引脚的输出可按 1 位为单位指定为标准 CMOS 输出或 N-通道漏极开路输出（V_{DD} 耐压），指定时使用端口输出模式寄存器 3（POM3）。

	78K0R/KC3-L (44-引脚) (μ PD78F100y: y = 0 至 3)	78K0R/KC3-L (48 引脚) (μ PD78F100y: y = 1 至 3)	78K0R/KD3-L (μ PD78F100y: y = 4 至 6)	78K0R/KE3-L (μ PD78F100y: y = 7 至 9)
P30/SO10/TxD1	√	√	√	√
P31/SI10/RxD1/ SDA10/INTP1	√	√	√	√
P32/SCK10/ SCL10/INTP2	√	√	√	√
P33	—	—	—	√

备注 √: 已配备

以下工作模式可按 1 位为单元指定。

(1) 端口模式

P30 至 P33 用作 I/O 端口。 P30 至 P33 可用端口模式寄存器 3 (PM3) 设置为以 1 位为单元的输入或输出端口。
片上上拉电阻的使用可通过上拉电阻选项寄存器 3 (PU3) 指定。

(2) 控制模式

P30 至 P33 可用作串行接口数据 I/O, 时钟 I/O 以及外部中断请求的输入。

(a) SI10

这是串行接口 CSI10 的串行数据输入引脚。

(b) SO10

这是串行接口 CSI10 的串行数据输出引脚。

(c) SCK10

这是串行接口 CSI10 的串行时钟 I/O 引脚。

(d) TxD1

这是串行接口 CUART1 的串行数据输出引脚。

(e) RxD1

这是串行接口 UART1 的串行数据输入引脚。

(f) SDA10

这是简易 I²C 的串行接口的串行数据 I/O 引脚。

(g) SCL10

这是简易 I²C 的串行接口的串行时钟 I/O 引脚。

(h) INTP1, INTP2

这些是指定有效沿（上升沿、下降沿或双边沿）的外部中断请求输入引脚。

注意事项 要将 P30/SO10/TxD1 和 P32/SCK10/SCL10/INTP2 用作通用端口，需将串行通信操作设置寄存器 02（SCR02）设置至默认状态（0087H）。此外，需将端口输出模式寄存器（POM3）清为 00H。

2.2.5 P40 至 P43（端口 4）

P40 至 P43 用作 I/O 端口。这些引脚还可用作 flash 存储器编程器/调试器的数据 I/O 和时钟输出。

	78K0R/KC3-L (44-引脚) (μPD78F100y: y = 0 至 3)	78K0R/KC3-L (48 引脚) (μPD78F100y: y = 1 至 3)	78K0R/KD3-L (μPD78F100y: y = 4 至 6)	78K0R/KE3-L (μPD78F100y: y = 7 至 9)
P40/TOOL0	√	√	√	√
P41/TOOL1	√	√	√	√
P42	—	—	—	√
P43	—	—	—	√

备注 √: 已配备

以下工作模式可按 1 位为单元指定。

(1) 端口模式

P40 至 P43 用作 I/O 端口。P40 至 P43 可用端口模式寄存器 4（PM4）设置为以 1 位为单元的输入或输出端口。片上上拉电阻的使用可通过上拉电阻选项寄存器 4（PU4）指定。
当允许片上调试（通过使用选项字节使能）时，务必给 P40 连接一个外部上拉电阻。

(2) 控制模式

P40 至 P43 可用作 flash 存储器编程器/调试器的数据 I/O 和时钟输出。

(a) TOOL0

这是 flash 存储器编程器/调试器的数据 I/O 引脚。

当允许片上调试（禁止将其下拉）时，务必要在外部上拉该引脚。

(b) TOOL1

这是调试器的时钟输出引脚。

当使用片上调试功能时，可通过调试器的模式设置使用 P41/TOOL1 引脚，如下所示。

1-线模式：可用作一个端口（P41）。

2-线模式：用作 TOOL1 引脚且不能用作端口（P41）。

注意事项 P40/TOOL0 引脚的功能不同在以下（a）至（c）中说明。

在（b）或（c）情况下，需进行指定的连接。

(a) 在正常操作模式下且通过选项字节（000C3H）禁止片上调试（OCDENSET = 0）时
=> 该引脚用作端口引脚（P40）。

(b) 在正常操作模式下且通过选项字节（000C3H）使能片上调试（OCDENSET = 1）时
=> 通过外部电阻连接该引脚，且通常在复位解除之前给该引脚输入高电平。

(c) 当使用片上调试功能时，或处于 flash 存储器编程器的写入模式时
=> 该引脚用作 TOOL0。

直接将该引脚连接至片上调试仿真器或 flash 存储器编程器，或通过外部电阻连接至 V_{DD} 将其上拉。

2.2.6 P50 至 P53（端口 5）

P50 至 P53 用作 I/O 端口。这些引脚还可以用作实时计数器校正时钟输出和定时器 I/O。

	78K0R/KC3-L (44-引脚) (μPD78F100y: y = 0 至 3)	78K0R/KC3-L (48 引脚) (μPD78F100y: y = 1 至 3)	78K0R/KD3-L (μPD78F100y: y = 4 至 6)	78K0R/KE3-L (μPD78F100y: y = 7 至 9)
P50/TI06/TO06	√	√	√	P50 [※]
P51/TI07/TO07	√	√	√	P51 [※]
P52/RTC1HZ/ SLTI/SLTO	√	√	√	√
P53	—	—	—	√

注 TI06/TO06 和 TI07/TO07 仅能在 78K0R/KC3-L 和 78K0R/KD3-L 中共享。78K0R/KE3-L 不具有共享功能。

备注 √: 已配备

以下工作模式可按 1 位为单元指定。

(1) 端口模式

P50 至 P53 用作 I/O 端口。P50 至 P53 可用端口模式寄存器 5（PM5）设置为以 1 位为单元的输入或输出端口。片上上拉电阻的使用可通过上拉电阻选项寄存器 5（PU5）指定。

(2) 控制模式

P50 至 P53 用作实时计数器校正时钟输出和定时器 I/O。

(a) RTC1HZ

这是实时计数器校正时钟（1 Hz）输出引脚。

(b) SLTI

通过设置输入切换控制寄存器（ISC），将该引脚用作外部计数时钟的输入引脚或 16 位定时器 00 和 01 的捕获触发。

(c) SLTO

通过设置输入切换控制寄存器（ISC），该引脚用作 16 位定时器 00 和 01 的定时器输出引脚。

(d) TI06, TI07

这些引脚用于向 16-定时器 06 和 07 输入外部计数时钟/捕获触发。

(e) TO06, TO07

这些引脚为 16-定时器 06 和 07 的定时器输出引脚。

2.2.7 P60 和 P61（端口 6）

P60 和 P61 用作 I/O 端口。这些引脚还可用作串行接口 IICA 数据 I/O 和时钟 I/O。

	78K0R/KC3-L (44-引脚) (μ PD78F100y: y = 0 至 3)	78K0R/KC3-L (48 引脚) (μ PD78F100y: y = 1 至 3)	78K0R/KD3-L (μ PD78F100y: y = 4 至 6)	78K0R/KE3-L (μ PD78F100y: y = 7 至 9)
P60/SCL0	–	√	√	√
P61/SDA0	–	√	√	√

备注 √: 已配备

以下工作模式可按 1 位为单元指定。

(1) 端口模式

P60 和 P61 用作 I/O 端口。P60 和 P61 可用端口模式寄存器 6（PM6）设置为以 1 位为单元的输入或输出端口。P60 和 P61 的输出为 N-通道漏极开路输出(6 V 耐压)。

(2) 控制模式

P60 和 P61 还可用作串行接口 IICA 数据 I/O 和时钟 I/O。

(a) SDA0

这是串行接口 IICA 的串行数据 I/O 引脚。

(b) SCL0

这是串行接口 IICA 的串行时钟 I/O 引脚。

2.2.8 P70 至 P77（端口 7）

P70 至 P77 用作 I/O 端口。这些引脚还可用作按键中断输入、串行接口数据 I/O、时钟 I/O 以及外部中断请求的输入。

P71、P72、P74 和 P75 引脚的输入可通过标准输入缓存器或 TTL 输入缓存器按 1 位为单位进行指定，指定时使用端口输入模式寄存器 7（PIM7）。

P71、P72、P74 和 P75 引脚的输出可按 1 位为单位指定为标准 CMOS 输出或 N-通道漏极开路输出（V_{DD} 耐压），指定时使用端口输出模式寄存器 7（POM7）。

	78K0R/KC3-L (44-引脚) (μ PD78F100y: y = 0 至 3)	78K0R/KC3-L (48 引脚) (μ PD78F100y: y = 1 至 3)	78K0R/KD3-L (μ PD78F100y: y = 4 至 6)	78K0R/KE3-L (μ PD78F100y: y = 7 至 9)
P70/KR0/SO01/ INTP4	√	√	√	√
P71/KR1/SI01/ INTP5	√	√	√	√
P72/KR2/ SCK01/INTP6	√	√	√	√
P73/KR3/SO00/ TxD0	√	√	√	√
P74/KR4/SI00/ RxD0	√	√	√	√
P75/KR5/ $\overline{\text{SCK00}}$	√	√	√	√
P76/KR6	—	—	√	√
P77/KR7	—	—	√	√

备注 √: 已配备

以下工作模式可按 1 位为单元指定。

(1) 端口模式

P70 至 P77 用作 I/O 端口。P70 至 P77 可用端口模式寄存器 7（PM7）设置为以 1 位为单元的输入或输出端口。片上上拉电阻的使用可通过上拉电阻选项寄存器 7（PU7）指定。

(2) 控制模式

P70 至 P77 可用作按键中断输入、串行接口数据 I/O、时钟 I/O 以及外部中断请求的输入。

(a) KR0 至 KR7

这些为按键中断输入引脚。

(b) SI00, SI01

这些引脚为串行接口 CSI00 和 CSI01 的串行数据输入引脚。

(c) SO00, SO01

这些引脚为串行接口 CSI00 和 CSI01 的串行数据输出引脚。

(d) $\overline{\text{SCK00}}$, $\overline{\text{SCK01}}$

这些引脚为串行接口 CSI00 和 CSI01 的串行数据 I/O 引脚。

(e) RxD0

这是串行接口 UART0 的串行数据输入引脚。

(f) TxD0

这是串行接口 UART0 的串行数据输出引脚。

(g) INTP4 至 INTP6

这些是指定有效沿（上升沿、下降沿或双边沿）的外部中断输入引脚。

注意事项 要将 P70/KR0/SO01/INTP4、P72/KR2/SCK01/INTP6、P73/KR3/SO00/TxD0 和 P75/KR5/SCK00 用作通用端口，需将串行通信操作设置寄存器 00 和 01（SCR00 和 SCR01）设置至默认状态（0087H）。此外，需将端口输出模式寄存器 7（POM7）清为 00H。

2.2.9 P80 至 P83（端口 8）

P80 至 P83 用作 I/O 端口。这些引脚还可以用作比较器 0 和 1（+）侧的输入电压、比较器 0 和 1（-）侧的输入电压、外部中断请求输入和可编程增益放大器输入。

P80 至 P83 引脚的输入必须用端口输入模式寄存器 8（PIM8）按 1 位为单位使能或禁止。

	78K0R/KC3-L (44-引脚) (μ PD78F100y: y = 0 至 3)	78K0R/KC3-L (48 引脚) (μ PD78F100y: y = 1 至 3)	78K0R/KD3-L (μ PD78F100y: y = 4 至 6)	78K0R/KE3-L (μ PD78F100y: y = 7 至 9)
P80/CMP0P/ INTP3/PGAI	√	√	√	√
P81/CMP0M	√	√	√	√
P82/CMP1P/ INTP7	√	√	√	√
P83/CMP1M	√	√	√	√

备注 √: 已配备

以下工作模式可按 1 位为单元指定。

(1) 端口模式

P80 至 P83 用作 I/O 端口。P80 至 P83 可用端口模式寄存器 8（PM8）设置为以 1 位为单元的输入或输出端口。

(2) 控制模式

P80 至 P83 可以用作比较器 0 和 1（+）侧的输入电压、比较器 0 和 1（-）侧的输入电压、外部中断请求输入和可编程增益放大器输入。

(a) CMP0P, CMP1P

这些是比较器 0 和 1（+）侧的输入电压引脚。

(b) CMP0M, CMP1M

这些是比较器 0 和 1（-）侧的输入电压引脚。

(c) INTP3, INTP7

这些是指定有效沿（上升沿、下降沿或双边沿）的外部中断输入引脚。

(d) PGAI

这是可编程增益放大器的输入引脚。

2.2.10 P120 至 P124（端口 12）

P120 用作 I/O 端口。P121 至 P124 用作输入端口。这些引脚还可用作外部中断输入、外部低电压检测的电压输入、主系统时钟的连接谐振器、主系统时钟的连接谐振器以及主系统时钟的外部时钟输入。

	78K0R/KC3-L (44-引脚) (μ PD78F100y: y = 0 至 3)	78K0R/KC3-L (48 引脚) (μ PD78F100y: y = 1 至 3)	78K0R/KD3-L (μ PD78F100y: y = 4 至 6)	78K0R/KE3-L (μ PD78F100y: y = 7 至 9)
P120/INTP0/ EXLVI	√	√	√	√
P121/X1	√	√	√	√
P122/X2/ EXCLK	√	√	√	√
P123/XT1	√	√	√	√
P124/XT2	√	√	√	√

备注 √: 已配备

以下工作模式可按 1 位为单元指定。

(1) 端口模式

P120 用作 I/O 端口。P120 还可用端口模式寄存器 12（PM12）设置至输入或输出端口。片上上拉电阻的使用可通过上拉电阻选项寄存器 12（PU12）指定。

P121 至 P124 用作输入端口。

(2) 控制模式

P120 至 P124 用作外部中断输入、外部低电压检测的电压输入、主系统时钟的连接谐振器、主系统时钟的连接谐振器以及主系统时钟的外部时钟输入。

(a) INTP0

这是可被指定的有效边沿（上升沿、下降沿或双边沿）的外部中断请求输入引脚。

(b) EXLVI

这是外部低电压检测的电压输入。

(c) X1, X2

这些是为主系统时钟连接谐振器的引脚。

(d) EXCLK

这是主系统时钟的外部时钟输入引脚。

(e) XT1, XT2

这些是为副系统时钟连接谐振器的引脚。

2.2.11 P140, P141（端口 14）

P140 用作 1 位输出端口。 P141 用作 1 位 I/O 端口。 这些引脚还可用作时钟/蜂鸣器输出。

	78K0R/KC3-L (44-引脚) (μ PD78F100y: y = 0 至 3)	78K0R/KC3-L (48 引脚) (μ PD78F100y: y = 1 至 3)	78K0R/KD3-L (μ PD78F100y: y = 4 至 6)	78K0R/KE3-L (μ PD78F100y: y = 7 至 9)
P140/PCLBUZ0	–	√	√	√
P141/PCLBUZ1	–	–	–	√

备注 √: 已配备

以下工作模式可按 1 位为单元指定。

(1) 端口模式

P140 用作 1 位输出端口。

P141 用作 1 位 I/O 端口。 P141 可用端口模式寄存器 14（PM14）设置为以 1 位为单元的输入或输出端口。 片上上拉电阻的使用可通过上拉电阻选项寄存器 14（PU14）指定。

(2) 控制模式

P140 和 P141 用作时钟/蜂鸣器输出。

(a) PCLBUZ0, PCLBUZ1

这些是时钟/蜂鸣器输出引脚。

2.2.12 P150 至 P153（端口 15）

P150 至 P153 用作 I/O 端口。 这些引脚还可用作 A/D 转换器模拟输入引脚。

	78K0R/KC3-L (44-引脚) (μ PD78F100y: y = 0 至 3)	78K0R/KC3-L (48 引脚) (μ PD78F100y: y = 1 至 3)	78K0R/KD3-L (μ PD78F100y: y = 4 至 6)	78K0R/KE3-L (μ PD78F100y: y = 7 至 9)
P150/ANI8	√	√	√	√
P151/ANI9	√	√	√	√
P152/ANI10	–	√	√	√
P153/ANI11	–	–	–	√

备注 √: 已配备

以下工作模式可按 1 位为单元指定。

(1) 端口模式

P150 至 P153 用作 I/O 端口。 P150 至 P153 可用端口模式寄存器 15（PM15）设置为以 1 位为单元的输入或输出端口。

(2) 控制模式

P150 至 P153 用作 A/D 转换器的模拟输入引脚（ANI8 至 ANI11）。当将这些引脚用作模拟输入引脚时，参见 11.6 (5) ANI0/P20 至 ANI7/P27 和 ANI8/P150 至 ANI11/P153。

注意事项 复位解除之后，在数字输入（通用端口）模式下设置 ANI8/P150 至 ANI11/P153 引脚。

2.2.13 AVREF, AVSS, VDD, EVDD, VSS, EVSS

	78K0R/KC3-L (44-引脚) (μ PD78F100y: y = 0 至 3)	78K0R/KC3-L (48 引脚) (μ PD78F100y: y = 1 至 3)	78K0R/KD3-L (μ PD78F100y: y = 4 至 6)	78K0R/KE3-L (μ PD78F100y: y = 7 至 9)
AVREF	√	√	√	√
AVSS	√	√	√	√
VDD	√	√	√	√
EVDD	—	—	—	√
VSS	√	√	√	√
EVSS	—	—	—	√

(1) AVREF

这是 A/D 转换器和比较器参考电压输入引脚以及 P20 至 P27、P150 至 P153、P80 至 P83、A/D 转换器、可编程增益放大器和比较器的电源正极。

当端口 2、端口 15 和端口 18 的所有引脚都用作模拟端口引脚时，使得 AVREF 的电压为 $1.8\text{ V} \leq \text{AVREF} \leq \text{VDD}$ 。当端口 2、端口 15 和端口 8 的一个或一个以上的引脚用作数字端口引脚时，或不使用 A/D 转换器、可编程增益放大器和比较器时，使 AVREF 与 EVDD 或 VDD 的电压相同。

(2) AVSS

这是 A/D 转换器、可编程增益放大器、比较器、P20 至 P27、P150 至 P153 以及 P80 至 P83 的接地电压。即使不使用 A/D 转换器、可编程增益放大器和比较器时，通常也使该引脚与 EVSS 或 VSS 电压相同。

(3) VDD, EVDD

VDD 是 P121 至 P124 和非端口的电源正极引脚（除 RESET 引脚和 FLMD0 引脚之外）[※]。

EVDD 为除 P20 至 P27、P150 至 P153、P80 至 P83 和 P121 至 P124 之外端口的电源正极引脚，也是 RESET 引脚和 FLMD0 引脚的电源正极引脚。

注 对于不提供 EVDD 引脚的产品，使用 VDD 作为除 P20 至 P27、P150 至 P153 和 P80 至 P83 之外的端口引脚的电源正极引脚，还用其作为端口之外的引脚的电源正极引脚。

(4) VSS, EVSS

VSS 是 P121 至 P124 和非端口的接地电压引脚（除 RESET 引脚和 FLMD0 引脚之外）[※]。

EVSS 为除 P20 至 P27、P150 至 P153、P80 至 P83 和 P121 至 P124 之外端口的接地电压引脚，也是 RESET 引脚和 FLMD0 引脚的接地电压引脚。

注 对于不提供 EVSS 引脚的产品，使用 VSS 作为除 P20 至 P27、P150 至 P153 和 P80 至 P83 之外的端口引脚的接地电压引脚，还用其作为端口之外的引脚的接地电压引脚。

<R> 2.2.14 RESET

此为低有效系统复位输入引脚。

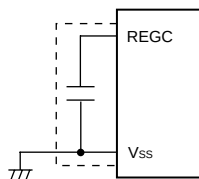
当不使用外部复位引脚时，通过电阻或直接该引脚连接至 V_{DD} 。

当使用外部复位引脚时，要基于 V_{DD} 设计电路。

2.2.15 REGC

此为内部操作连接稳压器输出（2.4 V）稳定电容的引脚。通过电容（0.47 至 1 μF ：目标）将该引脚连接至 V_{SS} 。

另外，需使用具有优良特性的电容，因为它用来稳定内部电压。



注意事项 使上图虚线框部分的接线长度尽可能地最短。

2.2.16 FLMD0

此为设置 flash 存储器编程模式的引脚。

进行以下的处理。

(a) 正常操作模式中

正常操作期间，建议将该引脚保持开路。

复位解除前 FLMD0 引脚必须保持 V_{SS} 的电压，但因为通过复位会在内部下拉该引脚。所以不必外部下拉该引脚。然而，必须用后台事件控制寄存器（BECTL）的位 7（FLMDPUP）保持选中将其下拉（参见 24.5 (1) 后台事件控制寄存器）。要在外部将其下拉，则需使用 200 k Ω 或更小的电阻。

通过直接将该引脚连接至 V_{SS} 引脚，可使用硬件禁止自编程和带有编程器的 flash 存储器的重写。

(b) 自编程模式中

使用自编程功能时，建议将该引脚保持开路。要在外部将其下拉，则需使用 100 k Ω 至 200 k Ω 的电阻。

在自编程模式下，在自编程库中将设置切换至上拉。

(c) flash 存储器编程模式中

当数据由 flash 存储器编程器写入时，直接将该引脚连接至 flash 存储器编程器。这给 FLMD0 引脚提供 V_{DD} 电平的写入电压。

因为 FLMD0 引脚可通过复位内部下拉，所以不必在外部将其下拉。要在外部将其下拉，则需使用 1 k Ω 至 200 k Ω 的电阻。

2.3 引脚 I/O 电路以及未使用引脚连接建议

表 2-3 显示了引脚 I/O 电路的类型以及未使用引脚的连接建议。

表 2-3. 未使用引脚的连接 (1/3)

引脚名称	I/O 电路类型	I/O	未使用引脚的连接建议
P00/TI00	8-R	I/O	输入： 通过电阻单独连接至 EV _{DD} 或 EV _{SS} 输出： 保持开路。
P01/TO00	5-AG		
P10/TI02/TO02	8-R		
P11/TI03/TO03			
P12/TI04/TO04/RTCDIV/ RTCCL			
P13/TI05/TO05			
P14/TI06/TO06 ^{註1}			
P15/TI07/TO07 ^{註1}			
P16			
P17			
P20/ANI0 至 P27/ANI7 ^{註2}	11-G	输入： 通过电阻单独连接至 AV _{REF} 或 AV _{SS} 。 输出： 保持开路	
P30/SO10/TxD1	5-AG	输入： 通过电阻单独连接至 EV _{DD} 或 EV _{SS} 。 输出： 保持开路。	
P31/SI10/RxD1/SDA10/ INTP1	5-AN	<N 通道漏极开路时> 输出 - 将端口输出锁存器设为 0： 保持开路。 - 将端口输出锁存器设为 1： 通过电阻单独连接至 EV_{DD} 或 EV_{SS}。	
P32/SCK10/SCL10/INTP2			
P33	5-AG	输入： 通过电阻单独连接至 EV _{DD} 或 EV _{SS} 。 输出： 保持开路	
P40/TOOL0	8-R	<使能片上调试时> 上拉该引脚（禁止下拉）。 <禁止片上调试时> 输入： 通过电阻单独连接至 EV _{DD} 或 EV _{SS} 。 输出： 保持开路。	
P41/TOOL1			
P42	5-AG	输入： 通过电阻单独连接至 EV _{DD} 或 EV _{SS} 。 输出： 保持开路。	
P43			
P50/TI06/TO06 ^{註3}	8-R		
P51/TI07/TO07 ^{註3}			
P52/RTC1HZ/SLTI/SLTO			
P53	5-AG		

- 注
1. 在非 78K0R/KE3-L 产品中，TI06/TO06 和 TI07/TO07 分别与 P50 和 P51 复用。
 2. 复位解除之后，在数字输入模式下设置 P20/ANI0 至 P27/ANI7 引脚。
 3. 在 78K0R/KE3-L 中，TI06/TO06 和 TI07/TO07 分别与 P14 和 P15 复用。

备注 对于不提供 EV_{DD} 或 EV_{SS} 引脚的产品，用 V_{DD} 代替 EV_{DD}，或用 V_{SS} 代替 EV_{SS}。

表 2-3. 未使用引脚的连接 (2/3)

引脚名称	I/O 电路类型	I/O	未使用引脚的连接建议
P60/SCL0	13-R	I/O	输入： 通过电阻单独连接至 EV _{DD} 或 EV _{SS} 。 输出 • 将端口输出锁存器设为 0： 保持开路。 • 将端口输出锁存器设为 1： 通过电阻单独连接至 EV _{DD} 或 EV _{SS} 。
P61/SDA0			
P70/KR0/SO01/INTP4	8-R		输入： 通过电阻单独连接至 EV _{DD} 或 EV _{SS} 。 输出： 保持开路。 <N 通道漏极开路时> 输出 • 将端口输出锁存器设为 0： 保持开路。 • 将端口输出锁存器设为 1： 通过电阻单独连接至 EV _{DD} 或 EV _{SS} 。
P71/KR1/SI01/INTP5	5-AN		输入： 通过电阻单独连接至 EV _{DD} 或 EV _{SS} 。 输出： 保持开路。
P72/KR2/SCK01/INTP6			输入： 通过电阻单独连接至 EV _{DD} 或 EV _{SS} 。 输出： 保持开路。 <N 通道漏极开路时> 输出 • 将端口输出锁存器设为 0： 保持开路。 • 将端口输出锁存器设为 1： 通过电阻单独连接至 EV _{DD} 或 EV _{SS} 。
P73/KR3/SO00/TxD0	8-R		
P74/KR4/SI00/RxD0	5-AN		输入： 通过电阻单独连接至 EV _{DD} 或 EV _{SS} 。 输出： 保持开路。
P75/KR5/SCK00			输入： 通过电阻单独连接至 EV _{DD} 或 EV _{SS} 。 输出： 保持开路。 <N 通道漏极开路时> 输出 • 将端口输出锁存器设为 0： 保持开路。 • 将端口输出锁存器设为 1： 通过电阻单独连接至 EV _{DD} 或 EV _{SS} 。
P76/KR6	8-R		输入： 通过电阻单独连接至 EV _{DD} 或 EV _{SS} 。 输出： 保持开路。
P77/KR7			

备注 对于不提供 EV_{DD} 或 EV_{SS} 引脚的产品，用 V_{DD} 代替 EV_{DD}，或用 V_{SS} 代替 EV_{SS}。

表 2-3. 未使用引脚的连接 (3/3)

引脚名称	I/O 电路类型	I/O	未使用引脚的连接建议
P80/CMP0P/INTP3/PGAI	11-J	I/O	输入： 通过电阻单独连接至 AVREF 或 AVSS。 输出： 保持开路。
P81/CMP0M	11-H		
P82/CMP1P/INTP7	11-I		
P83/CMP1M	11-H		
P120/INTP0/EXLVI	8-R		输入： 通过电阻单独连接至 EVDD 或 EVSS。 输出： 保持开路。
P121/X1 ^{註1}	37-C	输入	通过电阻单独连接至 EVDD 或 EVSS。
P122/X2/EXCLK ^{註1}			
P123/XT1 ^{註1}	37-B		
P124/XT2 ^{註1}			
P140/PCLBUZ0	3-C	输出	保持开路。
P141/PCLBUZ1	5-AG	I/O	输入： 通过电阻单独连接至 EVDD 或 EVSS。 输出： 保持开路。
P150/ANI8 至 P153/ANI11 ^{註2}	11-G		输入： 通过电阻单独连接至 AVREF 或 AVSS。 输出： 保持开路。
AVREF	—	—	<P20 至 P27、P150 至 P153 或 P80 至 P83 中的一个或一个以上设为数字端口时> 使该引脚与 EVDD 或 VDD 电压相同。 <P20 至 P27、P150 至 P153 以及 P80 至 P83 全都设为模拟端口时> 使该引脚电压具有 1.8 V ≤ AVREF ≤ VDD 的范围。
AVSS	—	—	使该引脚与 EVSS 或 VSS 电压相同。
FLMD0	2-W	—	保持开路或通过 100 kΩ或更大的电阻连接至 VSS。
RESET	2	输入	直接或通过电阻连接至 VDD。
REGC	—	—	通过电容（0.47 至 1 μF：目标）连接至 VSS。

- 注 1. 不使用这些引脚时，在输入端口模式下使用上述建议的连接（参见 图 5-2 时钟工作模式控制寄存器（CMC）的格式）。
2. 复位解除之后，在数字输入模式下设置 P150/ANI8 至 P153/ANI11 引脚。

备注 对于不提供 EV_{DD} 或 EV_{SS} 引脚的产品，用 V_{DD} 代替 EV_{DD} ，或用 V_{SS} 代替 EV_{SS} 。

图 2-1. 引脚 I/O 电路列表 (1/3)

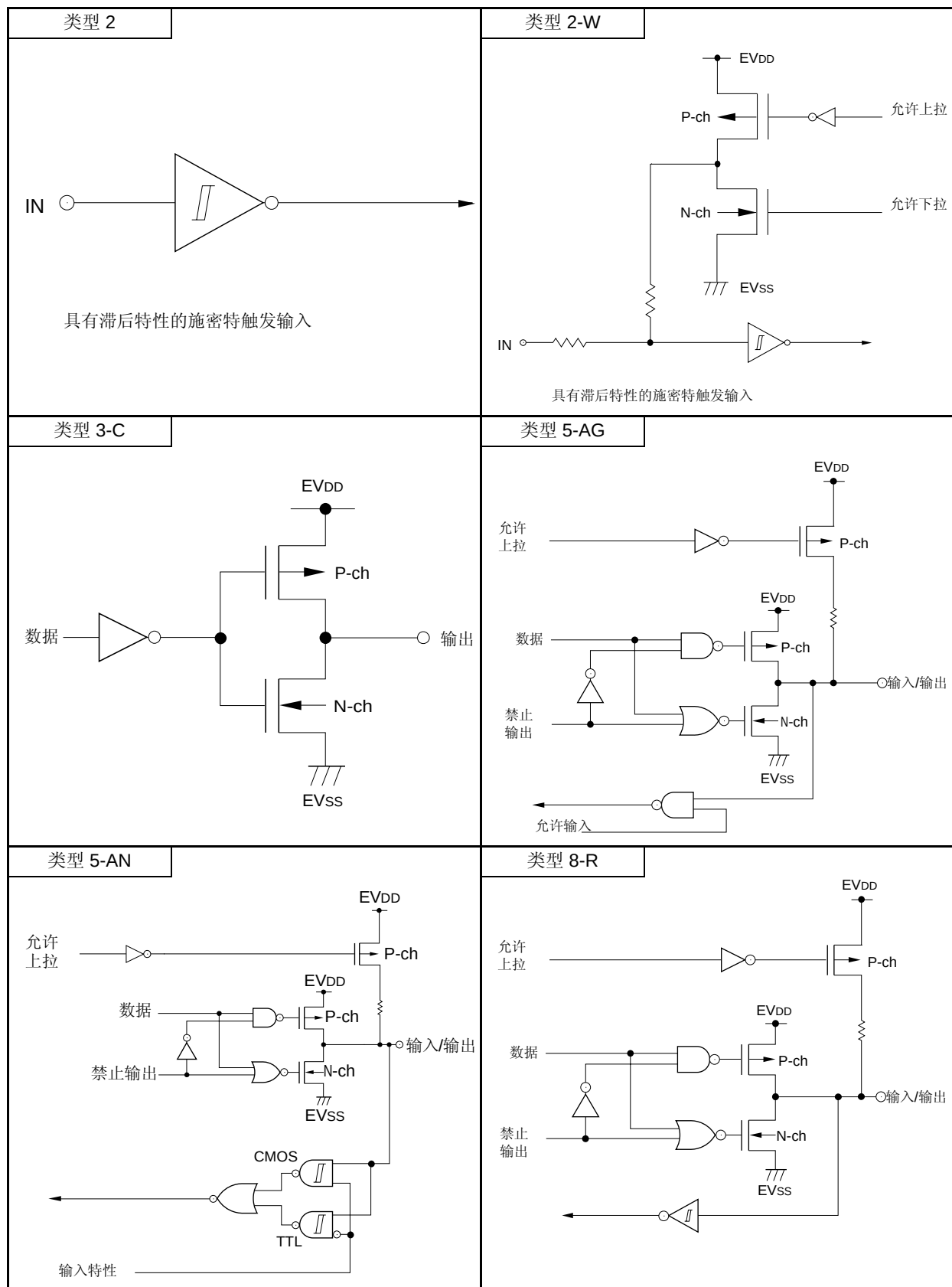


图 2-1. 引脚 I/O 电路列表 (2/3)

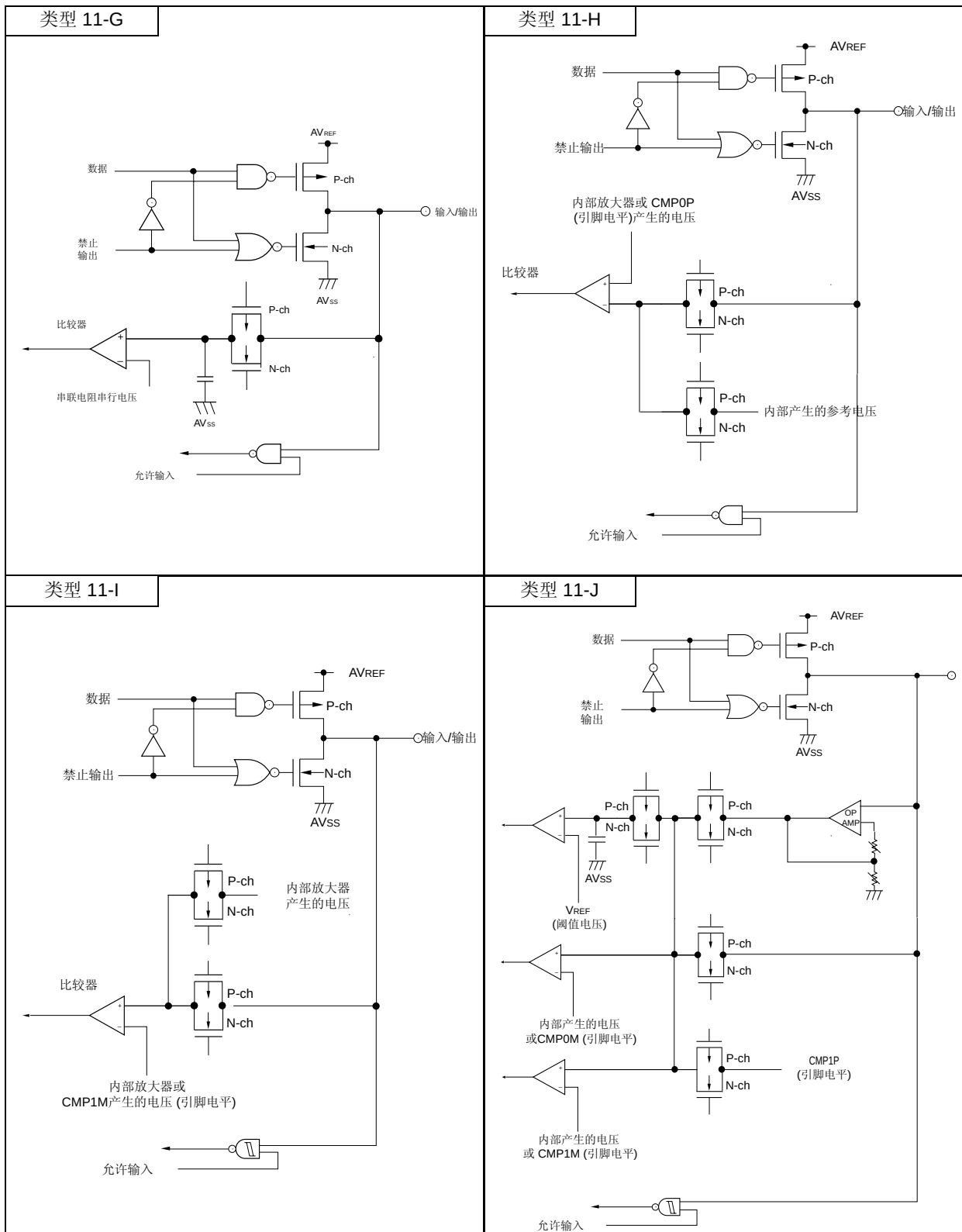
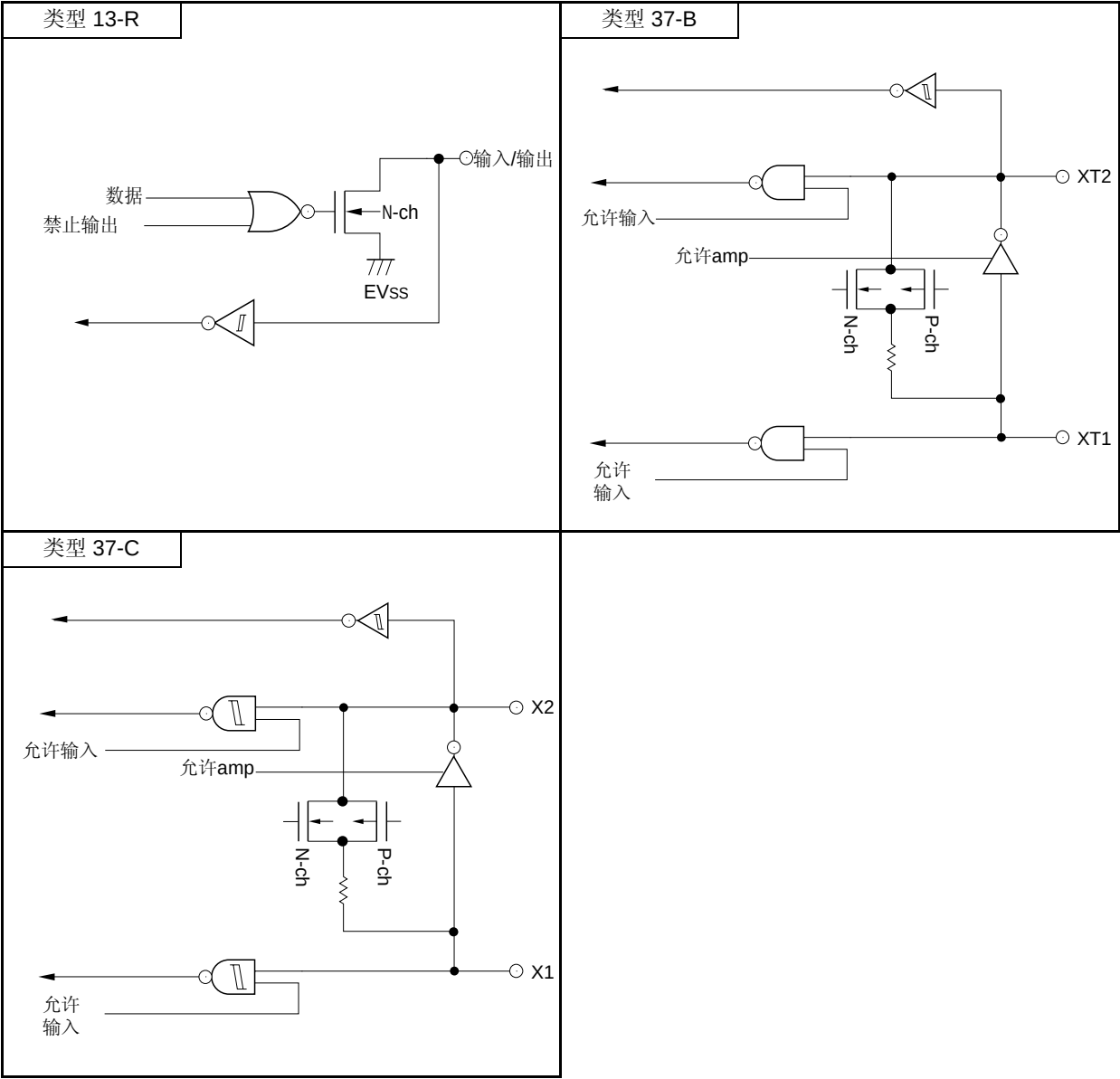


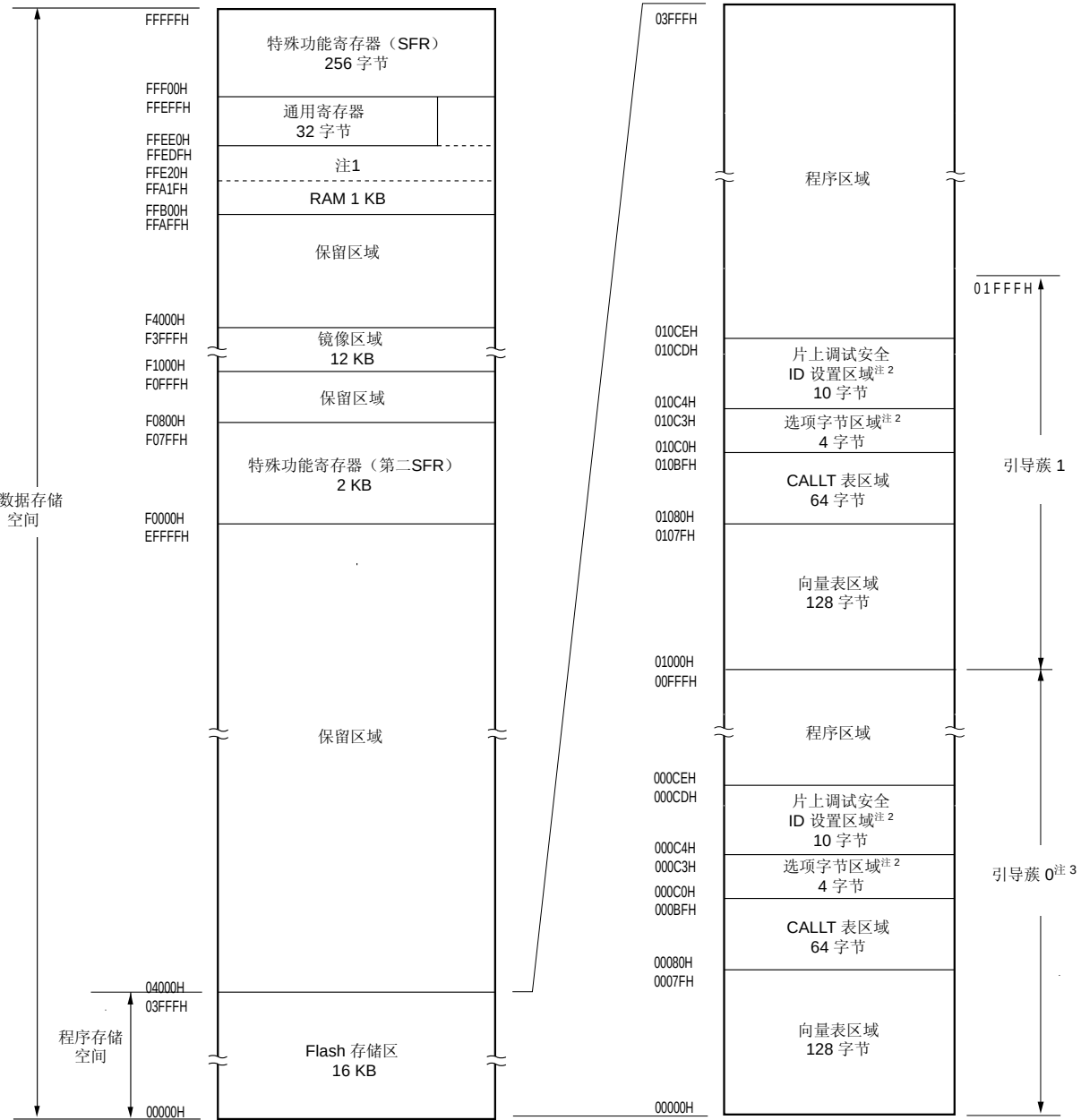
图 2-1. 引脚 I/O 电路列表 (3/3)



3.1 存储空间

78K0R/Kx3-L 中的产品可访问 1 MB 的存储空间。图 3-1 至 3-4 显示了存储空间的映射。

图 3-1. 存储器映射 (μPD78F1000)



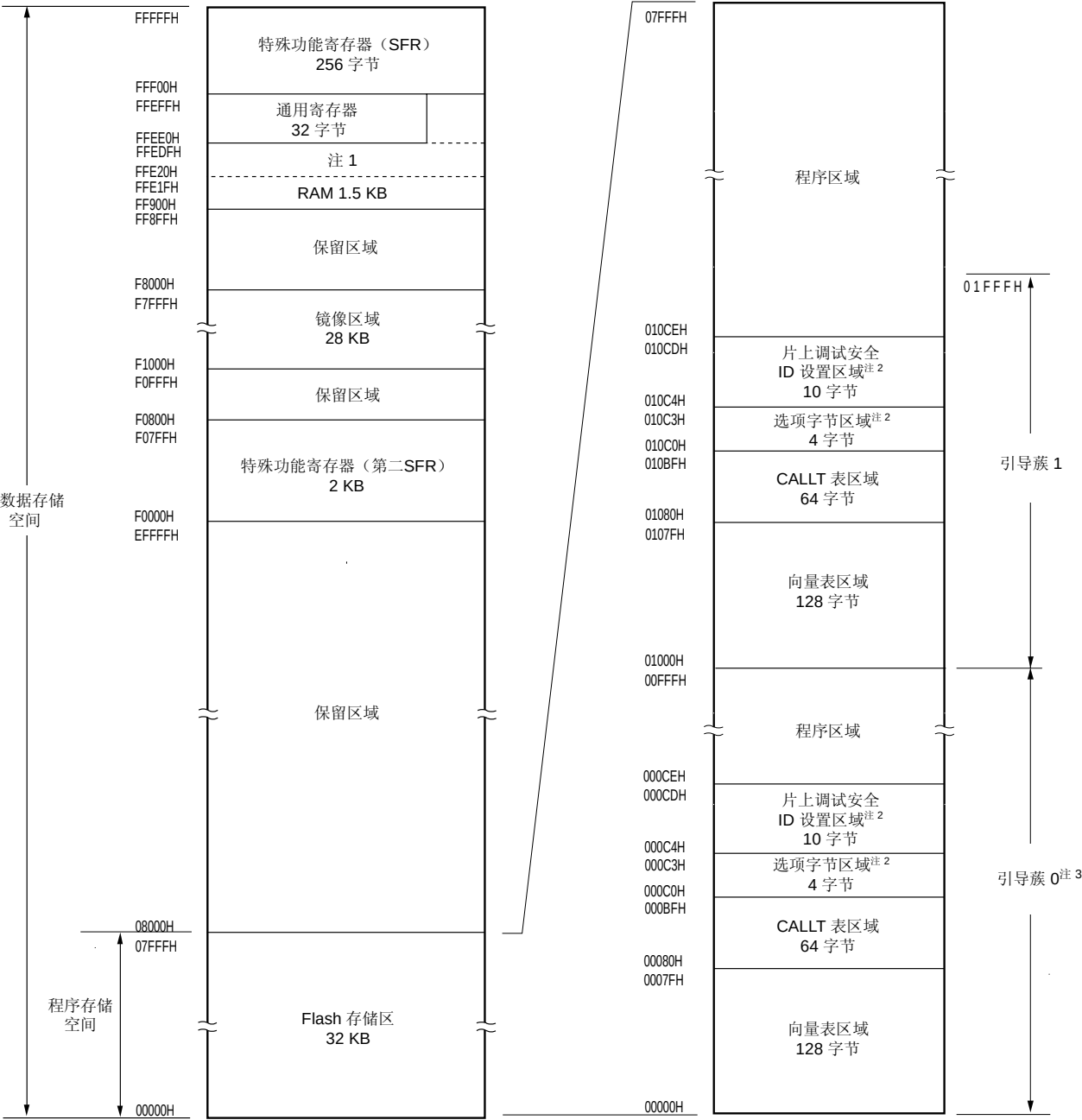
注 1. 使用自编程功能时禁止使用区域 FFE20H 至 FFEEDFH。因为该区域用于自编程库。

2. 不使用引导区交换时：将选项字节设置至 000C0H 至 000C3H，并将片上调试安全 ID 设置至 000C4H 至 000CDH。

使用引导区交换时：将选项字节设置至 000C0H 至 000C3H 以及 000C0H 至 000C3H，并将片上调试安全 ID 设置至 000C4H 至 000CDH 以及 010C4H 至 010CDH。

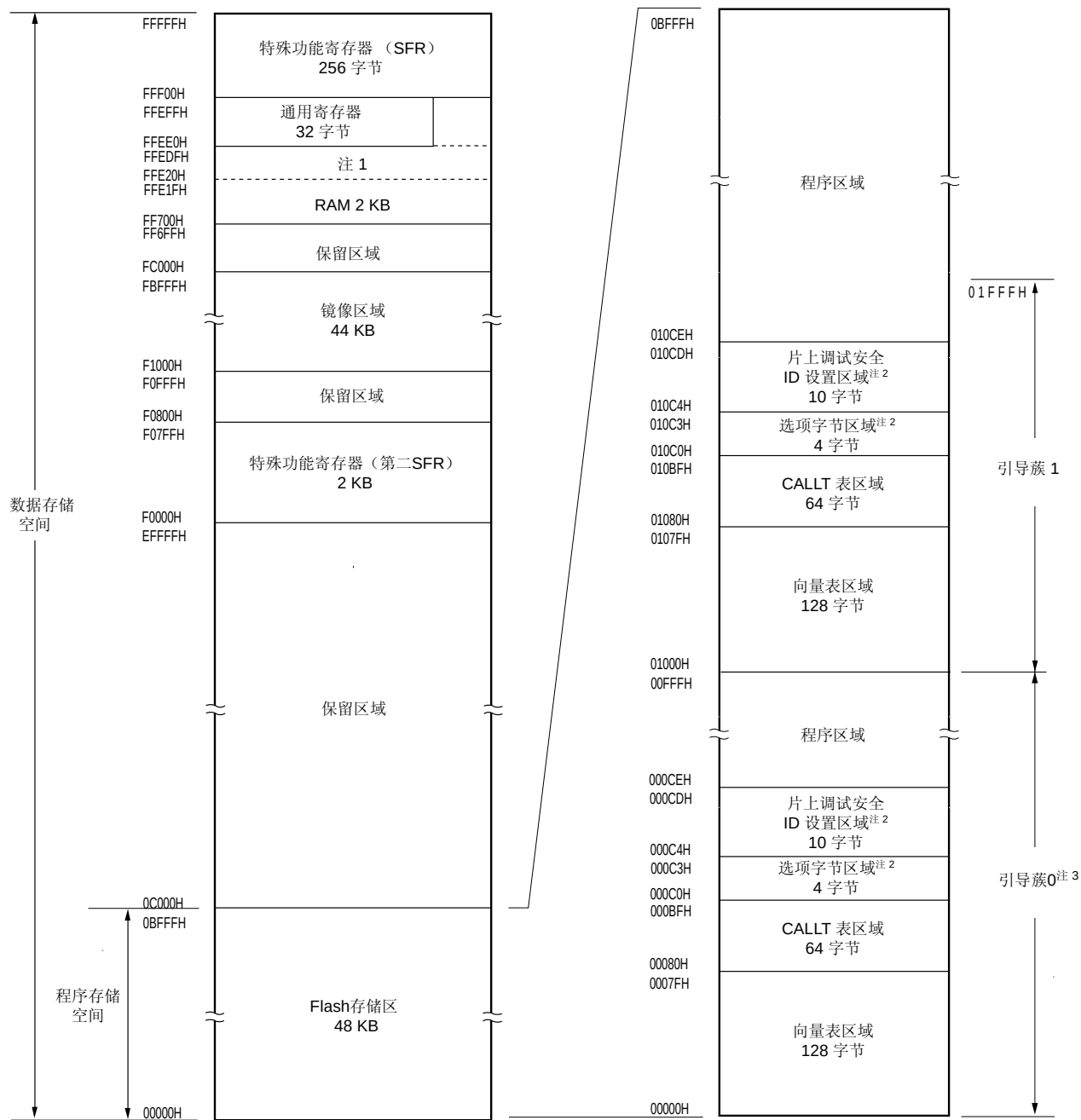
3. 根据安全设置可禁止写入引导区簇 0（参见 24.7 安全设置）。

图 3-2. 存储器映射 (μPD78F1001, 78F1004, 78F1007)

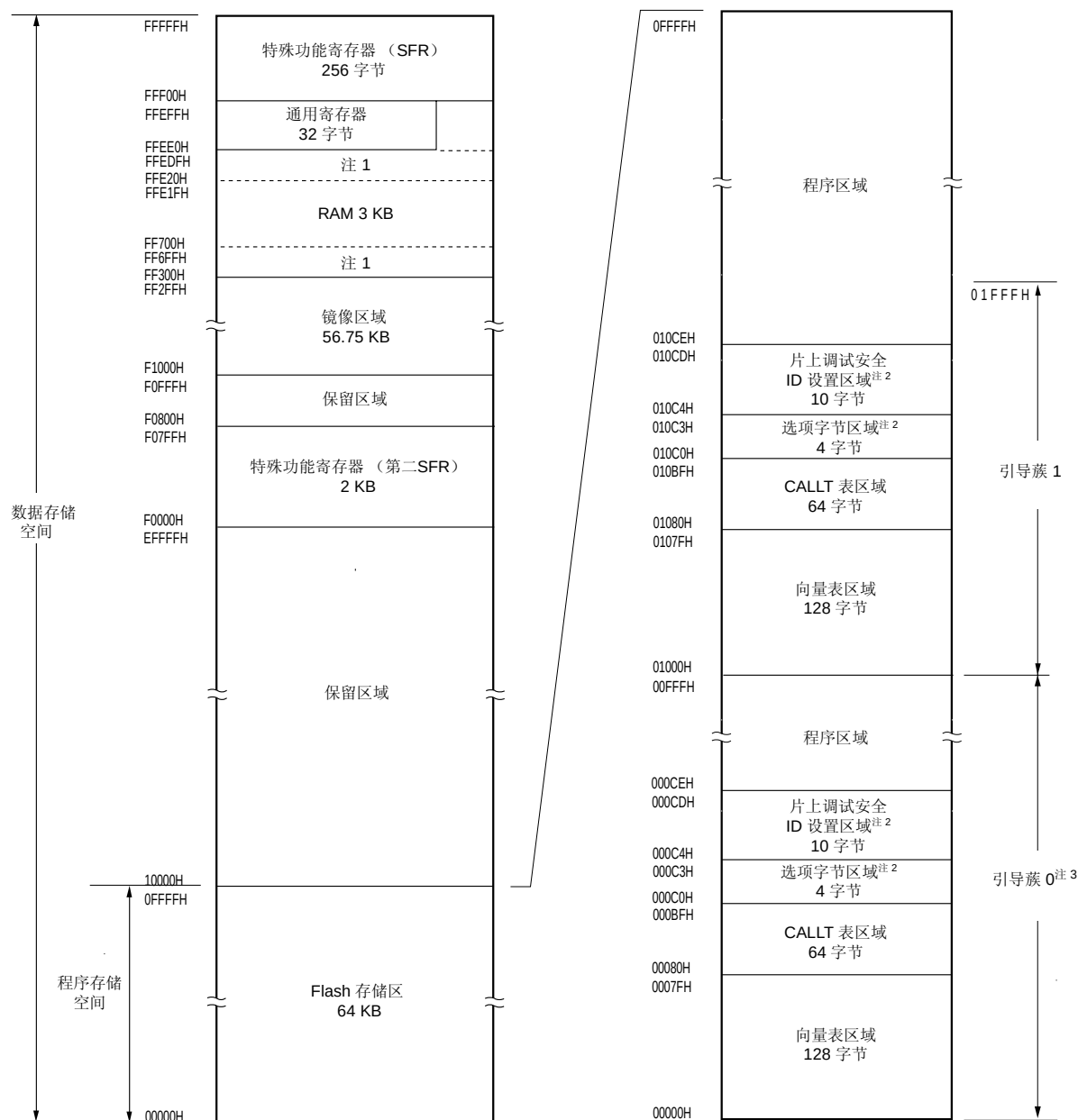


- 注
1. 使用自编程功能时禁止使用区域 FFE20H 至 FFEDFH。因为该区域用于自编程库。
 2. 不使用引导区交换时：将选项字节设置至 000C0H 至 000C3H，并将片上调试安全 ID 设置至 000C4H 至 000CDH。
使用引导区交换时：将选项字节设置至 000C0H 至 000C3H 以及 000C0H 至 000C3H，并将片上调试安全 ID 设置至 000C4H 至 000CDH 以及 010C4H 至 010CDH。
 3. 根据安全设置可禁止写入引导区簇 0（参见 24.7 安全设置）。

图 3-3. 存储器映射 (μPD78F1002, 78F1005, 78F1008)



- 注
1. 使用自编程功能时禁止使用区域 FFE20H 至 FFEDFH。因为该区域用于自编程库。
 2. 不使用引导区交换时：将选项字节设置至 000C0H 至 000C3H，并将片上调试安全 ID 设置至 000C4H 至 000CDH。
使用引导区交换时：将选项字节设置至 000C0H 至 000C3H 以及 000C0H 至 000C3H，并将片上调试安全 ID 设置至 000C4H 至 000CDH 以及 010C4H 至 010CDH。
 3. 根据安全设置可禁止写入引导区簇 0（参见 24.7 安全设置）。

图 3-4. 存储器映射 (μ PD78F1003, 78F1006, 78F1009)

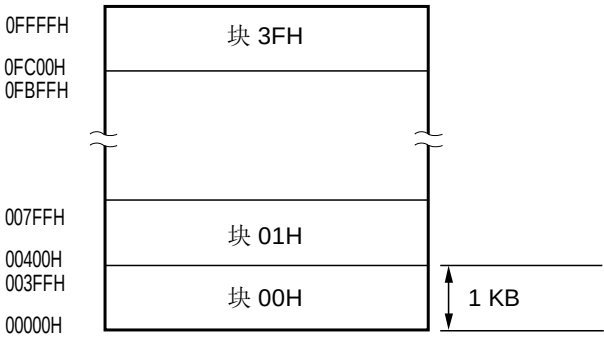
注 1. 使用自编程功能时禁止使用区域 FFE20H 至 FFEDFH 以及区域 FF300H 至 FF6FFH。 因为该区域用于自编程库。

2. 不使用引导区交换时：将选项字节设置至 000C0H 至 000C3H，并将片上调试安全 ID 设置至 000C4H 至 000CDH。

使用引导区交换时：将选项字节设置至 000C0H 至 000C3H 以及 000C0H 至 000C3H，并将片上调试安全 ID 设置至 000C4H 至 000CDH 以及 010C4H 至 010CDH。

3. 根据安全设置可禁止写入引导区簇0（参见 24.7 安全设置）。

备注 Flash 存储器被划分成块（1 块= 1 KB）。 有关地址值和块编号，参见表 3-1 flash 存储器中地址值和块编号的对应关系。



Flash 存储器中地址值和块编号的对应关系如下所示。

表 3-1. Flash 存储器中地址值和块编号的对应关系

地址值	块编号	地址值	块编号	地址值	块编号	地址值	块编号
00000H 至 003FFH	00H	04000H 至 043FFH	10H	08000H 至 083FFH	20H	0C000H 至 0C3FFH	30H
00400H 至 007FFH	01H	04400H 至 047FFH	11H	08400H 至 087FFH	21H	0C400H 至 0C7FFH	31H
00800H 至 00BFFH	02H	04800H 至 04BFFH	12H	08800H 至 08BFFH	22H	0C800H 至 0CBFFH	32H
00C00H 至 00FFFH	03H	04C00H 至 04FFFH	13H	08C00H 至 08FFFH	23H	0CC00H 至 0CFFFH	33H
01000H 至 013FFH	04H	05000H 至 053FFH	14H	09000H 至 093FFH	24H	0D000H 至 0D3FFH	34H
01400H 至 017FFH	05H	05400H 至 057FFH	15H	09400H 至 097FFH	25H	0D400H 至 0D7FFH	35H
01800H 至 01BFFH	06H	05800H 至 05BFFH	16H	09800H 至 09BFFH	26H	0D800H 至 0DBFFH	36H
01C00H 至 01FFFH	07H	05C00H 至 05FFFH	17H	09C00H 至 09FFFH	27H	0DC00H 至 0DFFFH	37H
02000H 至 023FFH	08H	06000H 至 063FFH	18H	0A000H 至 0A3FFH	28H	0E000H 至 0E3FFH	38H
02400H 至 027FFH	09H	06400H 至 067FFH	19H	0A400H 至 0A7FFH	29H	0E400H 至 0E7FFH	39H
02800H 至 02BFFH	0AH	06800H 至 06BFFH	1AH	0A800H 至 0ABFFH	2AH	0E800H 至 0EBFFH	3AH
02C00H 至 02FFFH	0BH	06C00H 至 06FFFH	1BH	0AC00H 至 0AFFFH	2BH	0EC00H 至 0EFFFH	3BH
03000H 至 033FFH	0CH	07000H 至 073FFH	1CH	0B000H 至 0B3FFH	2CH	0F000H 至 0F3FFH	3CH
03400H 至 037FFH	0DH	07400H 至 077FFH	1DH	0B400H 至 0B7FFH	2DH	0F400H 至 0F7FFH	3DH
03800H 至 03BFFH	0EH	07800H 至 07BFFH	1EH	0B800H 至 0BBFFH	2EH	0F800H 至 0FBFFH	3EH
03C00H 至 03FFFH	0FH	07C00H 至 07FFFH	1FH	0BC00H 至 0BFFFH	2FH	0FC00H 至 0FFFFH	3FH

备注 μ PD78F1000: 块编号 00H 至 0FH
 μ PD78F1001, 78F1004, 78F1007: 块编号 00H 至 1FH
 μ PD78F1002, 78F1005, 78F1008: 块编号 00H 至 2FH
 μ PD78F1003, 78F1006, 78F1009: 块编号 00H 至 3FH

3.1.1 内部编程存储器空间

内部编程存储器空间存储程序和表数据。

78K0R/Kx3-L 产品包括内部 ROM（flash 存储器），如下所示。

表 3-2. 内部 ROM 容量

元件编号	内部 ROM	
	结构	容量
μ PD78F1000	Flash 存储器	16384 $\times$ 8 位 (00000H 至 03FFFFH)
μ PD78F1001, 78F1004, 78F1007		32768 $\times$ 8 位 (00000H 至 07FFFFH)
μ PD78F1002, 78F1005, 78F1008		49152 $\times$ 8 位 (00000H 至 0BFFFFH)
μ PD78F1003, 78F1006, 78F1009		65536 $\times$ 8 位 (00000H 至 0FFFFFFH)

内部编程存储器空间划分为以下区域。

(1) 向量表区域

128-字节区域 00000H 至 0007FH 保留为向量表区域。复位或发生各中断请求时的分支程序起始地址存储于向量表区域。

16 位地址中，低 8 位存储在偶地址而高 8 位存储在奇地址。

表 3-3. 向量表

向量表地址	中断源	向量表地址	中断源
00000H	RESET 输入, POC, LVI, WDT, TRAP	00026H	INTSR1
		00028H	INTSRE1
00004H	INTWDTI	0002AH	INTICA [※]
00006H	INTLVI	0002CH	INTTM00
00008H	INTP0	0002EH	INTTM01
0000AH	INTP1	00030H	INTTM02
0000CH	INTP2	00032H	INTTM03
0000EH	INTP3	00034H	INTAD
00010H	INTP4	00036H	INTRTC
00012H	INTP5	00038H	INTRTCI
00016H	INTCMP0	0003AH	INTKR
00018H	INTCMP1	00040H	INTMD
0001AH	INTDMA0	00042H	INTTM04
0001CH	INTDMA1	00044H	INTTM05
0001EH	INTST0/INTCSI00	00046H	INTTM06
00020H	INTSR0/INTCSI01	00048H	INTTM07
00022H	INTSRE0	0004AH	INTP6
00024H	INTST1/INTCSI10/INTIIC10	0004CH	INTP7

注 78K0R/KC3-L 的 44 引脚产品未配置此项。

(2) CALLT 指令表区域

64-字节区域 00080H 至 000BFH 可存储一个 2 字节调用指令 (CALLT) 的子程序入口地址。将子程序入口地址设为 00000H 至 0FFFFH 范围内的值 (因为地址代码为 2 个字节)。

要使用引导区交换功能, 还需在 01080H 至 010BFH 设置 CALLT 指令表。

(3) 选项字节区域

000C0H 至 000C3H 中 4-字节区域可用作选项字节区域。使用引导区交换时需在 010C0H 至 010C3H 设置选项字节。有关细节, 参见 **第二十三章 选项字节**。

(4) 片上调试安全 ID 设置区域

000C4H 至 000CDH 和 010C4H 至 010CDH 的 10-字节区域可用作片上调试安全 ID 设置区域。不使用引导区交换功能时在 000C4H 至 000CDH 设置 10 字节的片上调试安全 ID, 而使用引导区交换功能时在 000C4H 至 000CDH 和 010C4H 至 010CDH 设置 10 字节的片上调试安全 ID。有关细节, 参见 **第 25 章 片上调试功能**。

<R> 3.1.2 镜像区域

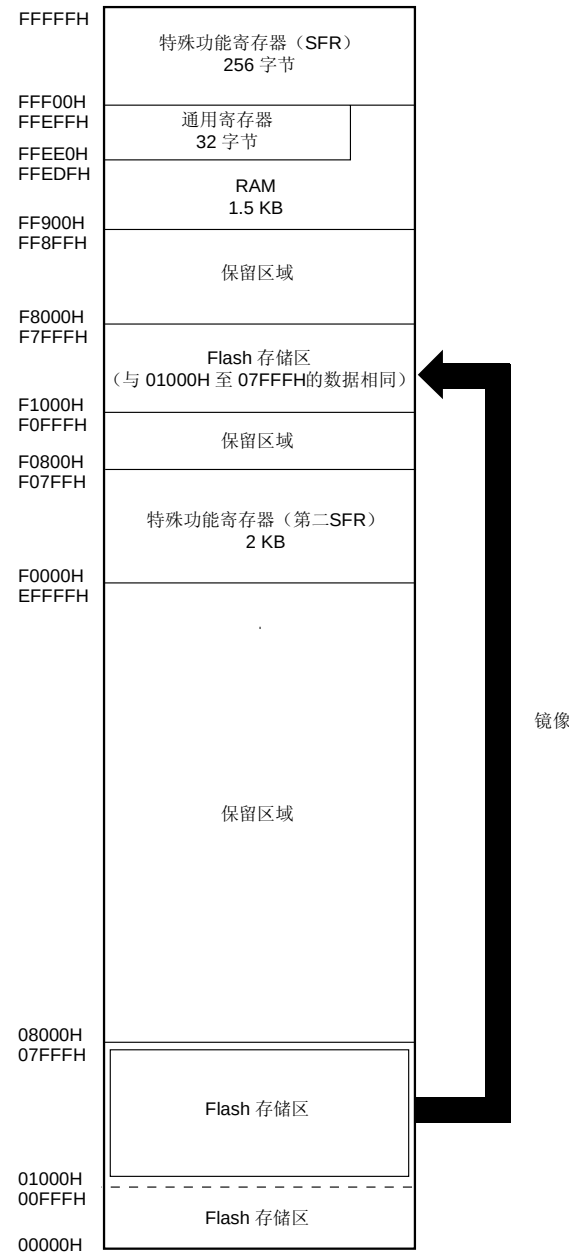
78K0R/Kx3-L 将 00000H 至 0FFFFH 的数据 flash 区域映射到 F0000H 至 FFFFFH（待镜像的数据 flash 区域由处理器模式控制寄存器（PMC）设置）。

通过从 F0000H 至 FFFFFH 读取数据，可使用不具有 ES 寄存器作为操作数的指令，因此可用较短的指令读取数据 flash 存储区的内容。但是，数据 flash 区域不会映射至 SFR、扩展 SFR、RAM 并使用禁止的区域。

有关各产品的镜像区域，参见 3.1 存储空间。
镜像区域为只读区域且不能从该区域取指令。

下面给出示例。

示例 μ PD78F1001, 78F1004, 78F1007（Flash 存储器: 32 KB, RAM: 1.5 KB）



PMC 寄存器说明如下。

<R> • 处理器模式控制寄存器（PMC）

该寄存器设置 flash 存储器空间，使其从 F0000H 映射至 FFFFFFFH。
PMC 可由 1 位或 8 位存储器操作指令进行设置。
复位信号产生后该寄存器被设为 00H。

图 3-5. 处理器模式控制寄存器（PMC）的配置格式

地址: FFFFEH 复位后: 00H R/W

符号	7	6	5	4	3	2	1	<0>
PMC	0	0	0	0	0	0	0	MAA

MAA	选择映射至区域F0000H 至 FFFFFH的flash存储空间
0	00000H 至 0FFFFH 映射到 F0000H 至 FFFFFH
1	禁止设置

- 注意事项
1. 务必将该寄存器的位 0（MAA）设为 0（默认值）。
 2. 在初始化设置优先于 DMA 控制器操作期间仅设置 PMC 一次。禁止在非初始化设置期间重写 PMC。
 3. 设置 PMC 之后，需等待至少一个指令然后再访问镜像区域。

3.1.3 内部数据存储器空间

78K0R/Kx3-L 产品包含以下 RAM。

表 3-4. 内部 RAM 容量

元件编号	内部 RAM
μPD78F1000	1024 × 8 位 (FFB00H 至 FFEFFH)
μPD78F1001, 78F1004, 78F1007	1536 × 8 位 (FF900H 至 FFEFFH)
μPD78F1002, 78F1005, 78F1008	2048 × 8 位 (FF700H 至 FFEFFH)
μPD78F1003, 78F1006, 78F1009	3072 × 8 位 (FF300H 至 FFEFFH)

32 字节区域 FFEE0H 至 FFEFFH 指定为 4 个通用寄存器组，每组包括 8 个 8 位寄存器。
该区域可用作写入并执行指令的程序空间。但是，在通用寄存器中禁止执行指令。
内部高速 RAM 也可用作堆栈存储器。

- 注意事项
1. 禁止使用通用寄存器（FFEE0H 至 FFEFFH）空间取指令，并禁止用作堆栈区域。
 2. 使用自编程功能时，不能将区域 FFE20H 至 FFEFFH 用作堆栈存储区。此外，μPD78F1003、78F1006 和 78F1009 的区域 FF300H 至 FF6FFH 也不能用作堆栈存储区。

3.1.4 特殊功能寄存器（SFR）区域

片上外设硬件特殊功能寄存器（SFR）分配在区域 FFF00H 至 FFFFFH（参见 3.2.4 中的 表 3-5 特殊功能寄存器（SFR））。

注意事项 不要访问未分配 SFR 的地址。

3.1.5 扩展的特殊功能寄存器（第二 SFR：第二特殊功能寄存器）区域

片上外设硬件特殊功能寄存器（2nd SFR）分配在区域 F0000H 至 F07FFH（参见 3.2.5 中的 表 3-6 扩展特殊功能寄存器（第二 SFR：第二特殊功能寄存器））。

除 SFR 区域（FFF00H 至 FFFFFH）的 SFR 之外的 SFR 被分配至该区域。但是，访问扩展 SFR 区域的指令比访问 SFR 区域的指令长一个字节。

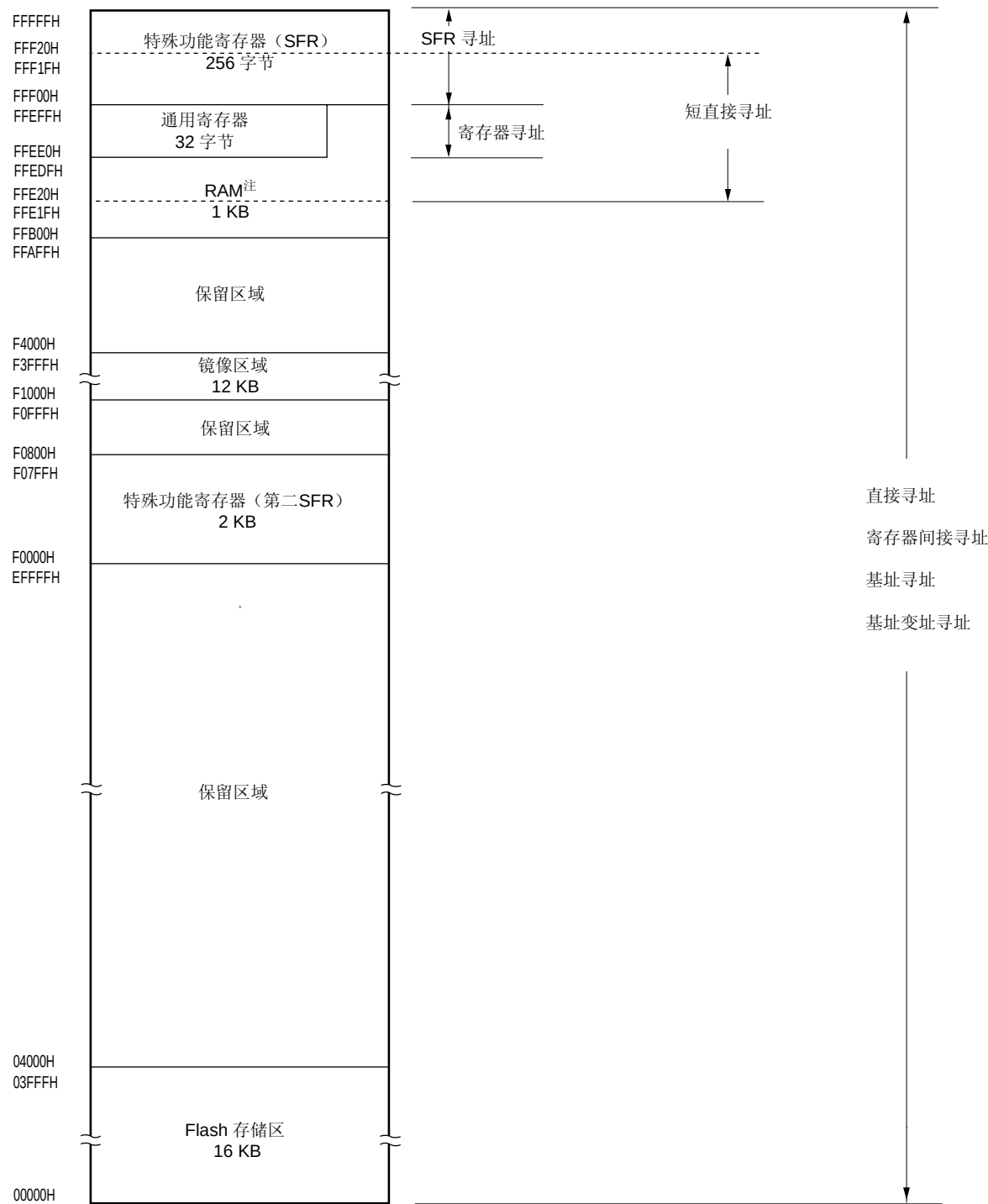
注意事项 不要访问未分配扩展 SFR 的地址。

3.1.6 数据存储器寻址

寻址涉及下一个将要执行指令的指定方法或与指令执行相关的寄存器或存储器的地址。

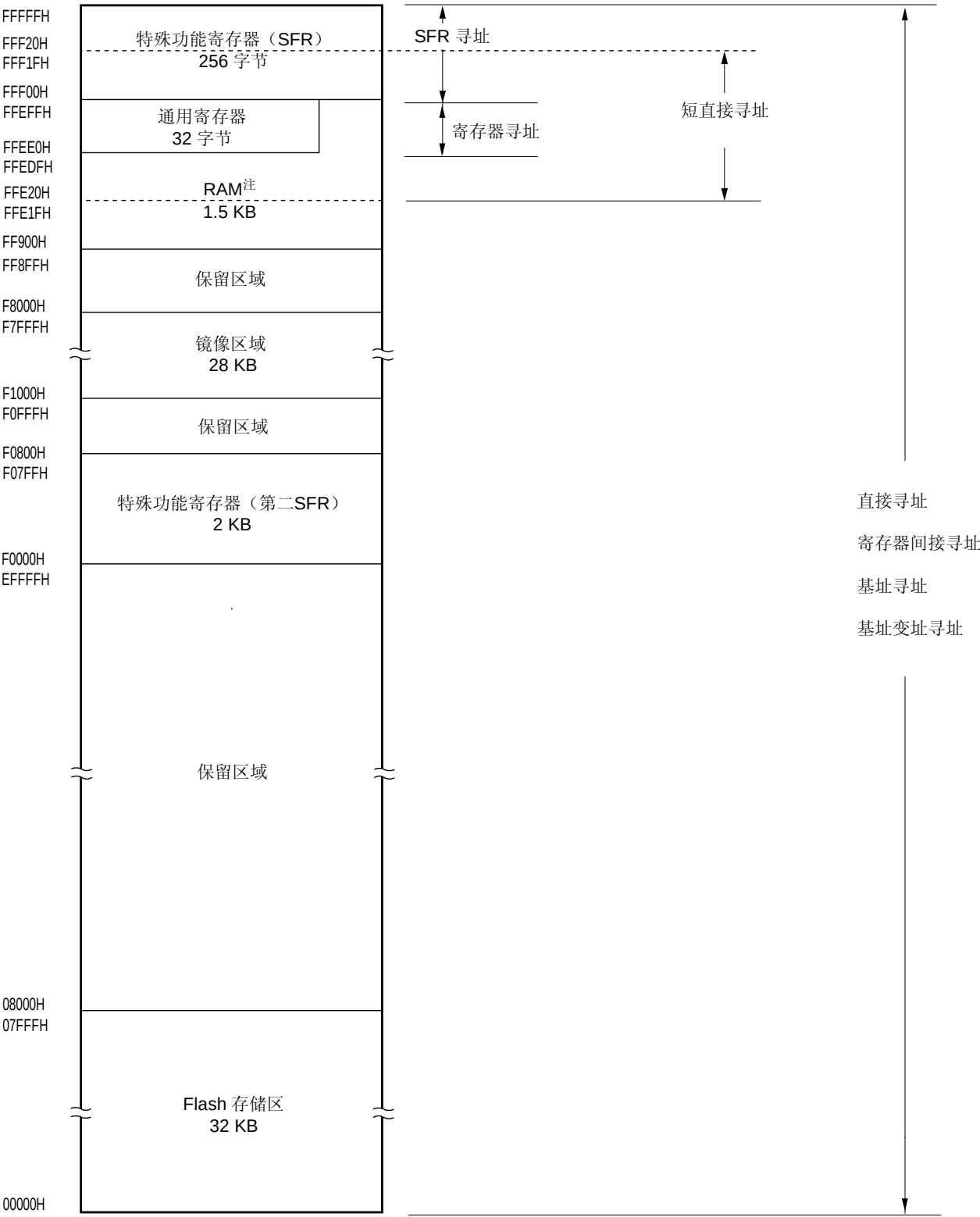
基于可操作性和其它原因，在 78K0R/Kx3-L 寻址与指令执行相关的存储器时提供多种寻址模式。尤其对于包含数据存储器的区域，可使用专为特殊功能寄存器（SFR）和通用寄存器设计的专用寻址方法。图 3-5 至 3-8 给出了数据存储器寻址和寻址之间的对应关系。

图 3-6. 数据存储器与寻址之间的对应关系（μPD78F1000）



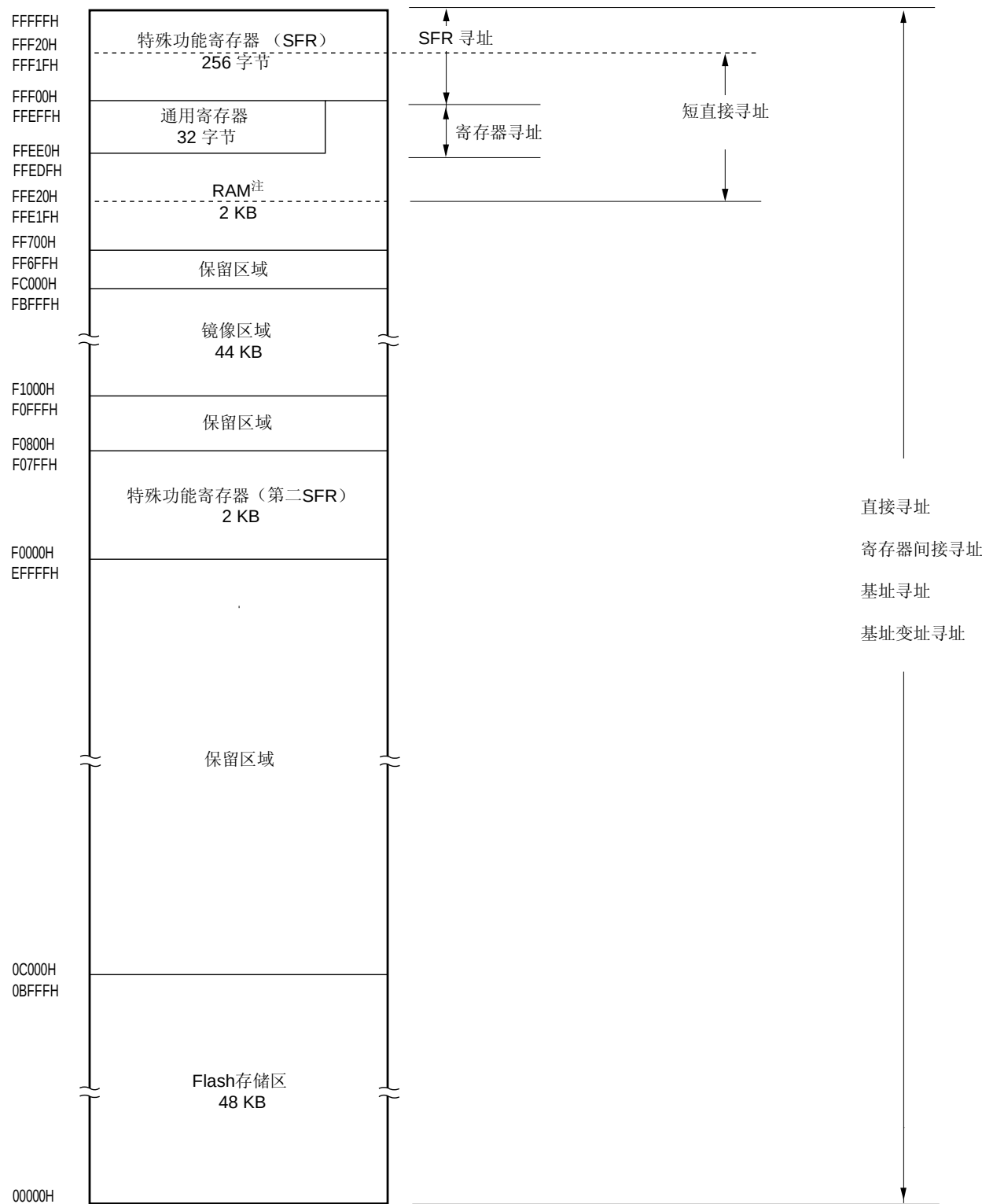
注 使用自编程功能时禁止使用区域 FFE20H 至 FFEDFH。 因为该区域用于自编程库。

图 3-7. 数据存储器与寻址之间的对应关系（ μ PD78F1001, 78F1004, 78F1007）



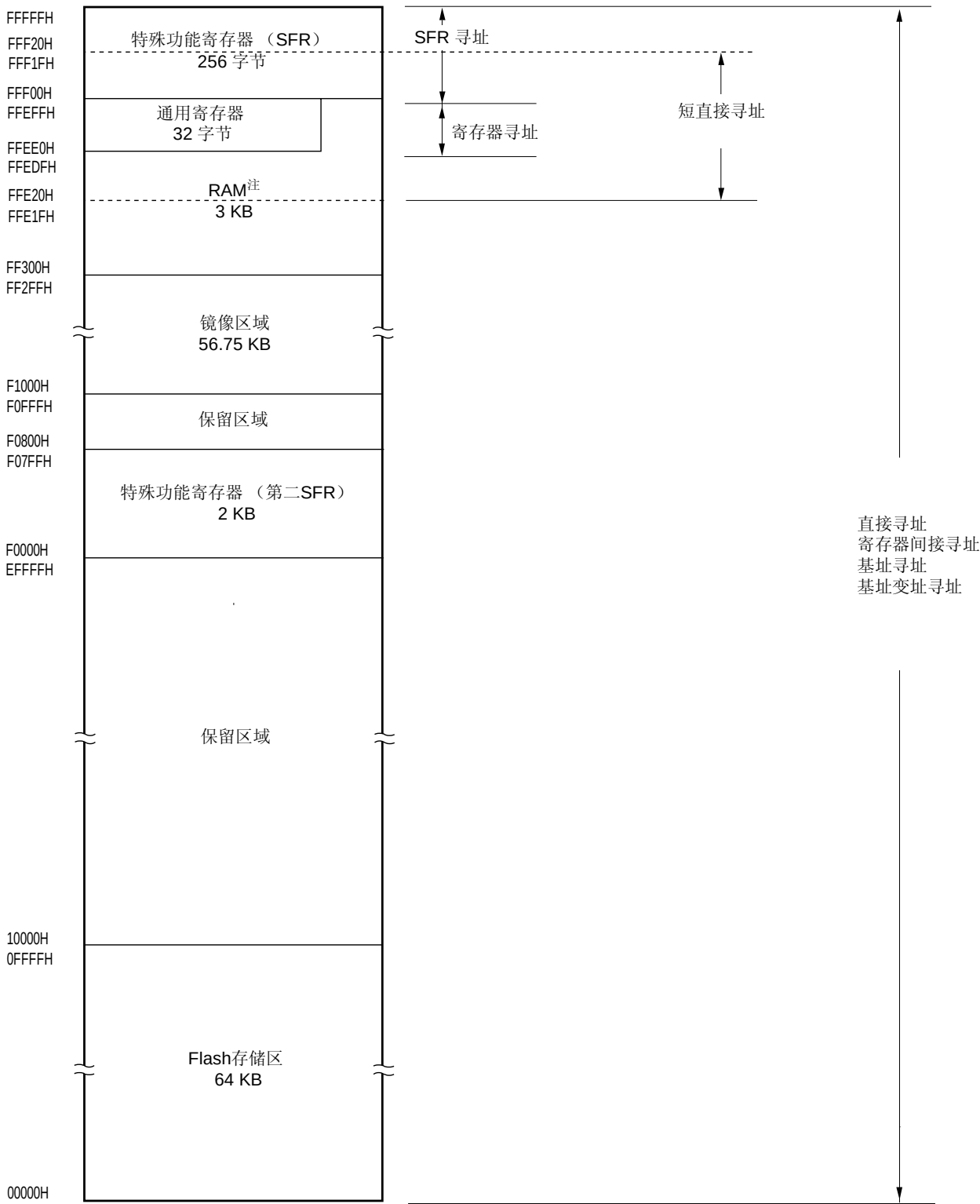
注 使用自编程功能时禁止使用区域 FFE20H 至 FFE1FH。 因为该区域用于自编程库。

图 3-8. 数据存储器与寻址之间的对应关系（ μ PD78F1002, 78F1005, 78F1008）



注 使用自编程功能时禁止使用区域 FFE20H 至 FFEDFH。 因为该区域用于自编程库。

图 3-9. 数据存储与寻址之间的对应关系（ μ PD78F1003, 78F1006, 78F1009）



注 使用自编程功能时禁止使用区域 FFE20H 至 FFEDFH 以及区域 FF300H 至 FF6FFH。 因为该区域用于自编程库。

3.2 处理器寄存器

78K0R/Kx3-L 产品包含以下的处理器寄存器。

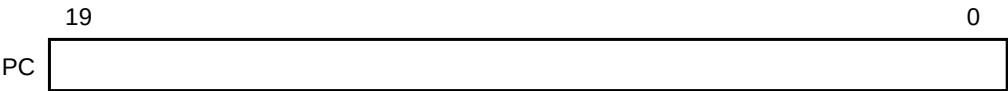
3.2.1 控制寄存器

控制寄存器控制程序顺序、状态和堆栈存储器。 控制寄存器由程序计数器（PC）、程序状态字（PSW）和堆栈指针（SP）组成。

(1) 程序计数器（PC）

程序计数器是 20 位寄存器，用以存储将要执行的下一个程序的地址信息。
正常工作下，程序计数器根据取回指令的字节数自动增加。 执行转移指令时，立即数和寄存器的内容被设置。
复位信号的产生会将地址 0000H 和 0001H 处的复位向量表的值设置至程序计数器。

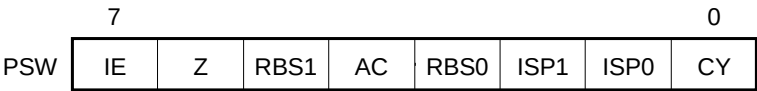
图 3-10. 程序计数器的格式



(2) 程序状态字（PSW）

程序状态字是一个 8 位寄存器，由不同的标志组成，执行指令对这些标志进行设置/复位。
向量中断请求被响应或执行 PUSH PSW 指令时，程序状态字内容可自动入栈；执行 RETB, RETI 和 POP PSW 指令时，程序状态字复位。 产生复位信号将 PSW 设为 06H。

图 3-11. 程序状态字的格式



(a) 中断使能标志（IE）

该标志控制 CPU 的中断请求识别操作。
为 0 时，IE 标志被设为中断禁止（DI）状态，并禁止所有的可屏蔽中断请求。
为 1 时，IE 标志被设为中断使能（EI）状态，中断请求应答由服务中优先级标志（ISP1，ISP0）、不同中断源的中断屏蔽标志和优先级指示标志控制。
执行 DI 指令或中断被响应时 IE 标志复位（为 0）；执行 EI 指令时 IE 被置位（为 1）。

(b) 零标志（Z）

操作结果为零时，该标志被置位（1）。 其它所有情况下该标志复位（0）。

(c) 寄存器组选择标志（RBS0，RBS1）

此标志为 2-位标志用以选择 4 组寄存器中的一组。
表示寄存器组由 SEL RBn 指令执行选中的 2-位信息存储于这些标志中。

(d) 辅助进位标志 (AC)

如果运算结果产生自位 3 进位或在位 3 借位，此标志被置位 (1)。所有其它情况下该标志复位 (0)。

(e) 服务中优先级标志 (ISP1, ISP0)

该标志管理可响应可屏蔽向量中断的优先级。通过优先级指示标志寄存器 (PRn0L, PRn0H, PRn1L, PRn1H, PRn2L, PRn2H) (参见 16.3 (3)) 指定的低于 ISP0 和 ISP1 值的向量中断请求不能被应答。实际的请求应答由中断使能标志 (IE) 控制。

备注 n = 0, 1

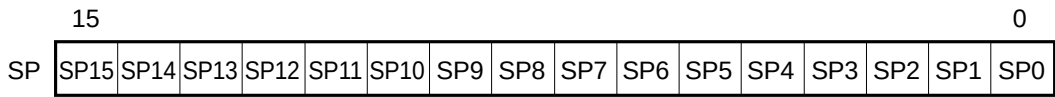
(f) 进位标志 (CY)

该标志在执行加/减指令时存储上溢或下溢。执行循环指令时，该标志存储移出值；执行位操作指令期间用作位累加器。

(3) 堆栈指针 (SP)

此为 16 位寄存器用以存放存储器堆栈区域的起始地址。只有内部 RAM 区才可设置为堆栈区域。

图 3-12. 堆栈指针的格式



写入 (存储) 堆栈存储器之前 SP 减 1；读取 (恢复) 堆栈存储器之后 SP 加 1。

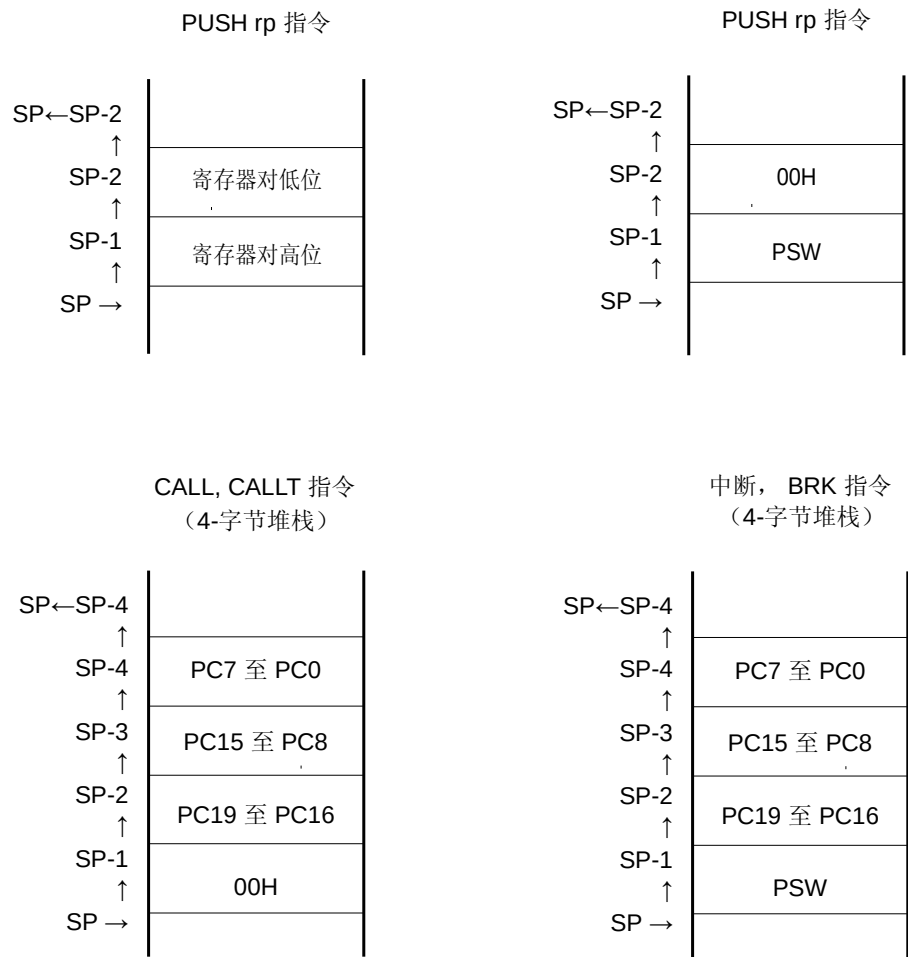
各堆栈操作存储如 图 3-13 所示的数据。

注意事项1. 因为复位信号的产生使得 SP 的内容不确定，所以使用堆栈之前务必要初始化 SP。

2. 禁止使用通用寄存器 (FFEE0H 至 FFEFFH) 空间用作堆栈区域。

3. 使用自编程功能时，不能将区域 FFE20H 至 FFEFFH 用作堆栈存储区。此外，μPD78F1003、78F1006 和 78F1009 的区域 FF300H 至 FF6FFH 也不能用作堆栈存储区。

图 3-13. 要存储到堆栈存储器的数据



3.2.2 通用寄存器

通用寄存器被映射到数据存储器的特殊地址（FFEE0H 至 FFEFFH）。通用寄存器由 4 个寄存器组组成，每组包含 8 个 8 位寄存器（X, A, C, B, E, D, L 和 H）。

每个寄存器可用作一个 8 位寄存器；两个 8 位寄存器还可以成对使用作为 16 位寄存器（AX, BC, DE 和 HL）。

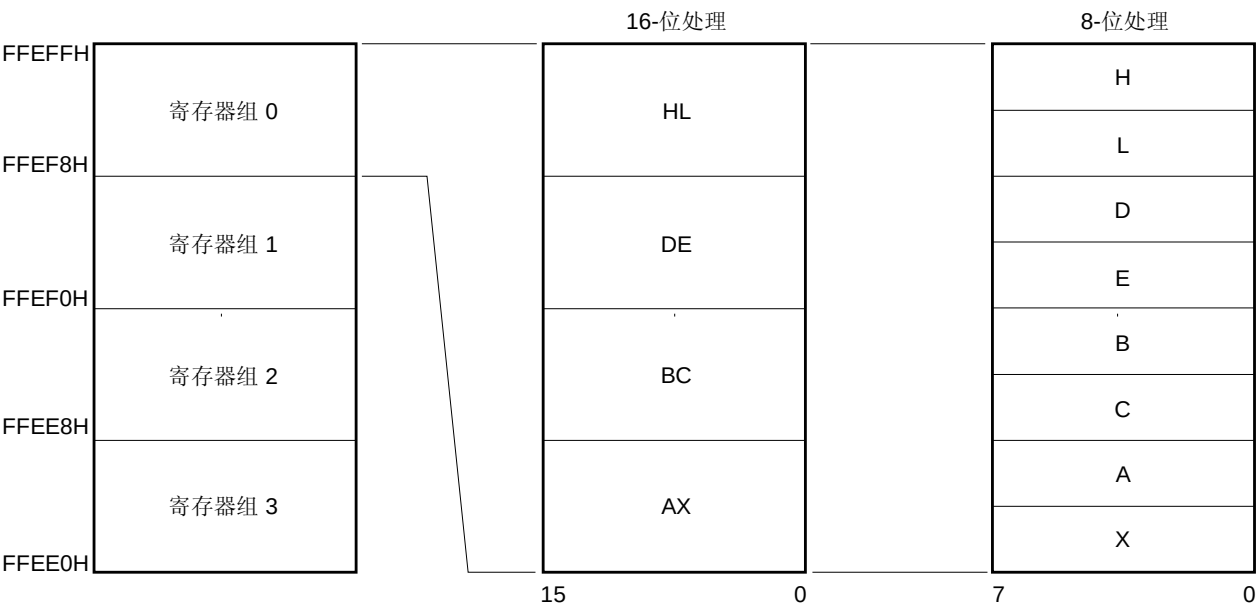
这些寄存器可按照功能名称（X, A, C, B, E, D, L, H, AX, BC, DE 和 HL）和绝对名称（R0 至 R7 和 RP0 至 RP3）说明。

用于指令执行的寄存器组由 CPU 控制指令（SEL RBn）设置。由于 4-寄存器组配置，可通过在正常处理寄存器和每组中断的寄存器之间切换创建高效的程序。

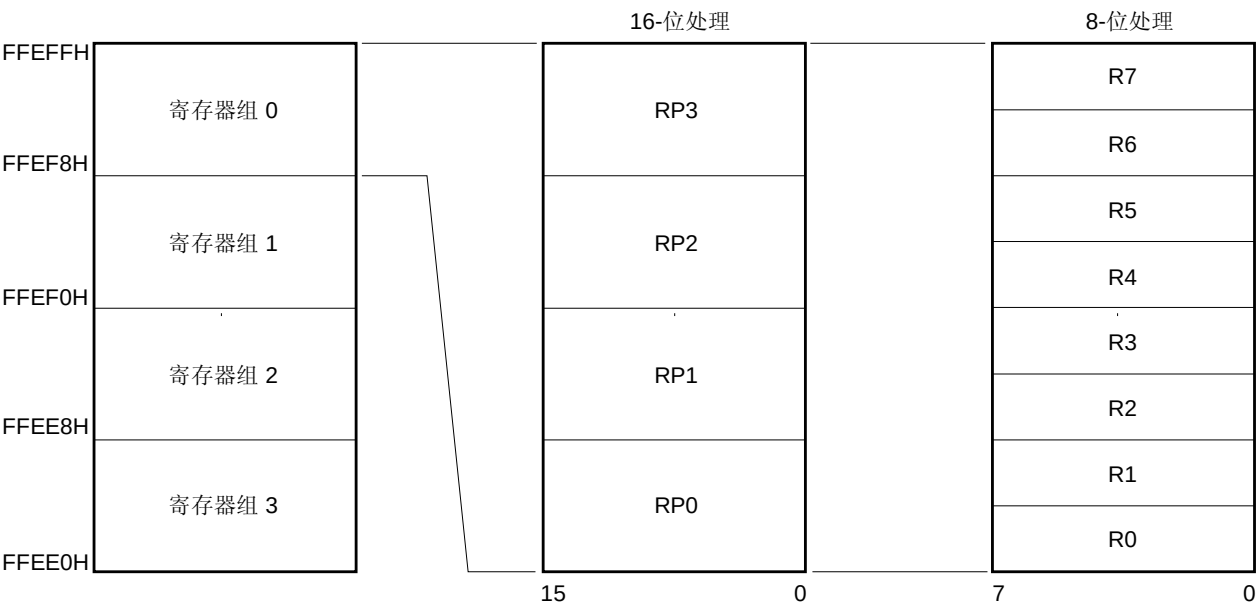
注意事项 禁止使用通用寄存器（FFEE0H 至 FFEFFH）空间取指令，并禁止用作堆栈区域。

图 3-14. 通用寄存器的配置

(a) 功能名称



(b) 绝对名称



3.2.3 ES 和 CS 寄存器

ES 寄存器用于数据存取，CS 寄存器用来在执行转移指令时指定高地址。
复位后，ES 寄存器的默认值为 0FH，CS 寄存器的默认值为 00H。

图 3-15. ES 和 CS 寄存器的配置

	7	6	5	4	3	2	1	0
ES	0	0	0	0	ES3	ES2	ES1	ES0

	7	6	5	4	3	2	1	0
CS	0	0	0	0	CS3	CP2	CP1	CP0

3.2.4 特殊功能寄存器（SFR）

与通用功能寄存器不同，每个 SFR 都具有一个特殊功能。

SFR 被分配至 FFF00H 至 FFFFFH 区域。

可使用运算、发送和位操作指令对 SFR 进行与通用寄存器相同的操作。根据 SFR 的类型可操作的位单元为 1，8 和 16。

各操作位单元可被指定如下。

- 1 位操作
为 1 位操作指令操作数（sfr.位）说明由汇编器保留的符号。该操作也可以由一个地址进行指定。
- 8 位操作
为 8 位操作指令操作数（sfr）说明由汇编器保留的符号。该操作也可以由一个地址进行指定。
- 16 位操作
为 16 位操作指令操作数（sfrp）说明由汇编器保留的符号。指定地址时，需说明一个偶地址。

表 3-5 给出了 SFR 的列表。表中各项的含义如下。

- 符号
符号表示特殊功能寄存器的地址。它在 RA78K0R 中是保留字，而在 CC78K0R 中则使用 #pragma sfr 伪指令将其定义为 sfr 变量。当给 78K0R 使用 RA78K0R，ID78K0R-QB 和 SM+ 时，符号可作为指令操作数写入。
- R/W
表明相应的 SFR 是否可以读写。
R/W: 读/写使能
R: 只读
W: 只写
- 操作位单元
“√”表示可操作位单元（1, 8 或 16）。“-”表示操作不可用的位单元。
- 复位后
表示复位信号产生时各寄存器的状态。

注意事项 不要访问未分配扩展 SFR 的地址。

备注 有关扩展的 SFR（第二 SFR），参见 3.2.5 扩展特殊功能寄存器（第二 SFR: 第二特殊功能寄存器）。

表 3-5. SFR 列表 (1/4)

地址	特殊功能寄存器（SFR）名称	符号		R/W	可操作位范围			复位后	78K0R/KC3-L(44 引脚)	78K0R/KC3-L(48 引脚)	78K0R/KD3-L	78K0R/KE3-L
					1 位	8 位	16 位					
FFF00H	端口寄存器 0	P0		R/W	√	√	—	00H	—	—	√	√
FFF01H	端口寄存器 1	P1		R/W	√	√	—	00H	√	√	√	√
FFF02H	端口寄存器 2	P2		R/W	√	√	—	00H	√	√	√	√
FFF03H	端口寄存器 3	P3		R/W	√	√	—	00H	√	√	√	√
FFF04H	端口寄存器 4	P4		R/W	√	√	—	00H	√	√	√	√
FFF05H	端口寄存器 5	P5		R/W	√	√	—	00H	√	√	√	√
FFF06H	端口寄存器 6	P6		R/W	√	√	—	00H	—	√	√	√
FFF07H	端口寄存器 7	P7		R/W	√	√	—	00H	√	√	√	√
FFF08H	端口寄存器 8	P8		R/W	√	√	—	00H	√	√	√	√
FFF0CH	端口寄存器 12	P12		R/W	√	√	—	未定义	√	√	√	√
FFF0EH	端口寄存器 14	P14		R/W	√	√	—	00H	—	√	√	√
FFF0FH	端口寄存器 15	P15		R/W	√	√	—	00H	√	√	√	√
FFF10H	串行数据寄存器 00	TXD0/ SIO00	SDR00	R/W	—	√	√	0000H	√	√	√	√
FFF11H		—			—	—			√	√	√	√
FFF12H	串行数据寄存器 01	RXD0/ SIO01	SDR01	R/W	—	√	√	0000H	√	√	√	√
FFF13H		—			—	—			√	√	√	√
FFF18H	定时器数据寄存器 00	TDR00		R/W	—	—	√	0000H	√	√	√	√
FFF19H												
FFF1AH	定时器数据寄存器 01	TDR01		R/W	—	—	√	0000H	√	√	√	√
FFF1BH												
FFF1EH	10-位 A/D 转换结果寄存器	ADCR		R	—	—	√	0000H	√	√	√	√
FFF1FH	8 位 A/D 转换结果寄存器	ADCRH		R	—	√	—	00H	√	√	√	√
FFF20H	端口模式寄存器 0	PM0		R/W	√	√	—	FFH	—	—	√	√
FFF21H	端口模式寄存器 1	PM1		R/W	√	√	—	FFH	√	√	√	√
FFF22H	端口模式寄存器 2	PM2		R/W	√	√	—	FFH	√	√	√	√
FFF23H	端口模式寄存器 3	PM3		R/W	√	√	—	FFH	√	√	√	√
FFF24H	端口模式寄存器 4	PM4		R/W	√	√	—	FFH	√	√	√	√
FFF25H	端口模式寄存器 5	PM5		R/W	√	√	—	FFH	√	√	√	√
FFF26H	端口模式寄存器 6	PM6		R/W	√	√	—	FFH	—	√	√	√
FFF27H	端口模式寄存器 7	PM7		R/W	√	√	—	FFH	√	√	√	√
FFF28H	端口模式寄存器 8	PM8		R/W	√	√	—	FFH	√	√	√	√
FFF2CH	端口模式寄存器 12	PM12		R/W	√	√	—	FFH	√	√	√	√
FFF2EH	端口模式寄存器 14	PM14		R/W	√	√	—	FEH	—	—	—	√
FFF2FH	端口模式寄存器 15	PM15		R/W	√	√	—	FFH	√	√	√	√
FFF30H	A/D 转换器模式寄存器	ADM		R/W	√	√	—	00H	√	√	√	√

表 3-5. SFR 列表 (2/4)

地址	特殊功能寄存器 (SFR) 名称	符号	R/W	可操作位范围			复位后	78K0R/KC3-L(44 引脚)	78K0R/KC3-L(48 引脚)	78K0R/KD3-L	78K0R/KE3-L
				1 位 t	8 位	16 位					
FFF31H	模拟输入通道指定寄存器	ADS	R/W	√	√	—	00H	√	√	√	√
FFF37H	按键返回模式寄存器	KRM	R/W	√	√	—	00H	√	√	√	√
FFF38H	外部中断上升沿使能寄存器 0	EGP0	R/W	√	√	—	00H	√	√	√	√
FFF39H	外部中断下降沿使能寄存器 0	EGN0	R/W	√	√	—	00H	√	√	√	√
FFF3CH	输入切换控制寄存器	ISC	R/W	√	√	—	00H	√	√	√	√
FFF3EH	定时器输入选择寄存器 0	TIS0	R/W	√	√	—	00H	√	√	√	√
FFF44H	串行数据寄存器 02	TXD1/ SIO10	SDR02	—	√	√	0000H	√	√	√	√
FFF45H		—		—	—	—		√	√	√	√
FFF46H	串行数据寄存器 03	RXD1	SDR03	—	√	√	0000H	√	√	√	√
FFF47H		—		—	—	—		√	√	√	√
FFF50H	IICA 移位寄存器	IICA	R/W	—	√	—	00H	—	√	√	√
FFF51H	IICA 状态寄存器	IICS	R	√	√	—	00H	—	√	√	√
FFF52H	IICA 标志寄存器	IICF	R/W	√	√	—	00H	—	√	√	√
FFF64H	定时器数据寄存器 02	TDR02	R/W	—	—	√	0000H	√	√	√	√
FFF65H				—	—	—		—	—	—	—
FFF66H	定时器数据寄存器 03	TDR03	R/W	—	—	√	0000H	√	√	√	√
FFF67H				—	—	—		—	—	—	—
FFF68H	定时器数据寄存器 04	TDR04	R/W	—	—	√	0000H	√	√	√	√
FFF69H				—	—	—		—	—	—	—
FFF6AH	定时器数据寄存器 05	TDR05	R/W	—	—	√	0000H	√	√	√	√
FFF6BH				—	—	—		—	—	—	—
FFF6CH	定时器数据寄存器 06	TDR06	R/W	—	—	√	0000H	√	√	√	√
FFF6DH				—	—	—		—	—	—	—
FFF6EH	定时器数据寄存器 07	TDR07	R/W	—	—	√	0000H	√	√	√	√
FFF6FH				—	—	—		—	—	—	—
FFF90H	减法计数器寄存器	RSUBC	R	—	—	√	0000H	√	√	√	√
FFF91H				—	—	—		—	—	—	—
FFF92H	秒计数器寄存器	SEC	R/W	—	√	—	00H	√	√	√	√
FFF93H	分计数器寄存器	MIN	R/W	—	√	—	00H	√	√	√	√
FFF94H	小时计数器寄存器	HOURL	R/W	—	√	—	12H [※]	√	√	√	√
FFF95H	星期计数器寄存器	WEEK	R/W	—	√	—	00H	√	√	√	√
FFF96H	天计数器寄存器	DAY	R/W	—	√	—	01H	√	√	√	√
FFF97H	月计数器寄存器	MONTH	R/W	—	√	—	01H	√	√	√	√

注 复位后如果 AMPM 位 (RTCC0 寄存器的位 3) 设置为 1, 则该寄存器的值为 00H。

表 3-5. SFR 列表 (3/4)

地址	特殊功能寄存器 (SFR) 名称	符号	R/W	可操作位范围			复位后	78K0R/KC3-L(44 引脚)	78K0R/KC3-L(48 引脚)	78K0R/KD3-L	78K0R/KE3-L
				1 位	8 位	16 位					
FFF98H	年计数器寄存器	YEAR	R/W	–	√	–	00H	√	√	√	√
FFF99H	钟表错误校正寄存器	SUBCUD	R/W	–	√	–	00H	√	√	√	√
FFF9AH	分报警寄存器	ALARMWWM	R/W	–	√	–	00H	√	√	√	√
FFF9BH	小时报警寄存器	ALARMWH	R/W	–	√	–	12H	√	√	√	√
FFF9CH	星期报警寄存器	ALARMWW	R/W	–	√	–	00H	√	√	√	√
FFF9DH	实时计数器控制寄存器 0	RTCC0	R/W	√	√	–	00H	√	√	√	√
FFF9EH	实时计数器控制寄存器 1	RTCC1	R/W	√	√	–	00H	√	√	√	√
FFF9FH	实时计数器控制寄存器 2	RTCC2	R/W	√	√	–	00H	√	√	√	√
FFFA0H	时钟工作模式控制寄存器	CMC	R/W	–	√	–	00H	√	√	√	√
FFFA1H	时钟工作状态控制寄存器	CSC	R/W	√	√	–	C0H	√	√	√	√
FFFA2H	振荡稳定时间计数器状态寄存器	OSTC	R	√	√	–	00H	√	√	√	√
FFFA3H	振荡稳定时间选择寄存器	OSTS	R/W	–	√	–	07H	√	√	√	√
FFFA4H	时钟控制寄存器	CKC	R/W	√	√	–	09H	√	√	√	√
FFFA5H	时钟输出选择寄存器 0	CKS0	R/W	√	√	–	00H	√	√	√	√
FFFA6H	时钟输出选择寄存器 1	CKS1	R/W	√	√	–	00H	√	√	√	√
FFFA8H	复位控制标志寄存器	RESF	R	–	√	–	未定义 ^{注1}	√	√	√	√
FFFA9H	低电压检测寄存器	LVIM	R/W	√	√	–	00H ^{注2}	√	√	√	√
FFFAAH	低电压检测电平选择寄存器	LVIS	R/W	√	√	–	0EH ^{注3}	√	√	√	√
FFFABH	看门狗定时器使能寄存器	WDTE	R/W	–	√	–	1A/9A ^{注4}	√	√	√	√
FFFB0H	DMA SFR 地址寄存器 0	DSA0	R/W	–	√	–	00H	√	√	√	√
FFFB1H	DMA SFR 地址寄存器 1	DSA1	R/W	–	√	–	00H	√	√	√	√
FFFB2H	DMA RAM 地址寄存器 0L	DRA0L	DRA0	R/W	–	√	00H	√	√	√	√
FFFB3H	DMA RAM 地址寄存器 0H	DRA0H		R/W	–	√	00H	√	√	√	√
FFFB4H	DMA RAM 地址寄存器 1L	DRA1L	DRA1	R/W	–	√	00H	√	√	√	√
FFFB5H	DMA RAM 地址寄存器 1H	DRA1H		R/W	–	√	00H	√	√	√	√
FFFB6H	DMA 字节计数寄存器 0L	DBC0L	DBC0	R/W	–	√	00H	√	√	√	√
FFFB7H	DMA 字节计数寄存器 0H	DBC0H		R/W	–	√	00H	√	√	√	√
FFFB8H	DMA 字节计数寄存器 1L	DBC1L	DBC1	R/W	–	√	00H	√	√	√	√
FFFB9H	DMA 字节计数寄存器 1H	DBC1H		R/W	–	√	00H	√	√	√	√
FFFBABH	DMA 模式控制寄存器 0	DMC0	R/W	√	√	–	00H	√	√	√	√
FFFBABH	DMA 模式控制寄存器 1	DMC1	R/W	√	√	–	00H	√	√	√	√
FFFBCH	DMA 操作控制寄存器 0	DRC0	R/W	√	√	–	00H	√	√	√	√
FFFBDBH	DMA 操作控制寄存器 1	DRC1	R/W	√	√	–	00H	√	√	√	√

- 注
1. RESF 的复位值根据复位源的不同而不同。
 2. LVIM 的复位值根据复位源和选择字节设置的不同而不同。
 3. LVIS 的复位值根据复位源的不同而不同。
 4. WDTE 的复位值取决于选项字节的设置。

表 3-5. SFR 列表 (4/4)

地址	特殊功能寄存器（SFR）名称	符号		R/W	可操作位范围			复位后	78K0R/KC3-L(44 引脚)	78K0R/KC3-L(48 引脚)	78K0R/KD3-L	78K0R/KE3-L
					1 位	8 位	16 位					
FFFBEH	后台事件控制寄存器	BECTL		R/W	√	√	—	00H	√	√	√	√
FFFC0H	—	PFCMD [※]		—	—	—	—	未定义	√	√	√	√
FFFC2H	—	PFS [※]		—	—	—	—	未定义	√	√	√	√
FFFC4H	—	FLPMC [※]		—	—	—	—	未定义	√	√	√	√
FFFD0H	中断请求标志寄存器 2	IF2L	IF2	R/W	√	√	√	0000H	√	√	√	√
FFFD1H		—			—	—			√	√	√	√
FFFD4H	中断屏蔽标志寄存器 2	MK2L	MK2	R/W	√	√	√	FFFFH	√	√	√	√
FFFD5H		—			—	—			√	√	√	√
FFFD8H	优先级指示标志寄存器 02	PR02L	PR02	R/W	√	√	√	FFFFH	√	√	√	√
FFFD9H		—			—	—			√	√	√	√
FFFDCH	优先级指示标志寄存器 12	PR12L	PR12	R/W	√	√	√	FFFFH	√	√	√	√
FFFDH		—			—	—			√	√	√	√
FFFE0H	中断请求标志寄存器 0L	IF0L	IF0	R/W	√	√	√	00H	√	√	√	√
FFFE1H	中断请求标志寄存器 0H	IF0H		R/W	√	√		00H	√	√	√	√
FFFE2H	中断请求标志寄存器 1L	IF1L	IF1	R/W	√	√	√	00H	√	√	√	√
FFFE3H	中断请求标志寄存器 1H	IF1H		R/W	√	√		00H	√	√	√	√
FFFE4H	中断屏蔽标志寄存器 0L	MK0L	MK0	R/W	√	√	√	FFH	√	√	√	√
FFFE5H	中断屏蔽标志寄存器 0H	MK0H		R/W	√	√		FFH	√	√	√	√
FFFE6H	中断屏蔽标志寄存器 1L	MK1L	MK1	R/W	√	√	√	FFH	√	√	√	√
FFFE7H	中断屏蔽标志寄存器 1H	MK1H		R/W	√	√		FFH	√	√	√	√
FFFE8H	优先级指定标志寄存器 00L	PR00L	PR00	R/W	√	√	√	FFH	√	√	√	√
FFFE9H	优先级指定标志寄存器 00H	PR00H		R/W	√	√		FFH	√	√	√	√
FFFEAH	优先级指定标志寄存器 01L	PR01L	PR01	R/W	√	√	√	FFH	√	√	√	√
FFFEBH	优先级指定标志寄存器 01H	PR01H		R/W	√	√		FFH	√	√	√	√
FFFECH	优先级指定标志寄存器 10L	PR10L	PR10	R/W	√	√	√	FFH	√	√	√	√
FFFEDH	优先级指定标志寄存器 10H	PR10H		R/W	√	√		FFH	√	√	√	√
FFFEEH	优先级指定标志寄存器 11L	PR11L	PR11	R/W	√	√	√	FFH	√	√	√	√
FFFEFH	优先级指定标志寄存器 11H	PR11H		R/W	√	√		FFH	√	√	√	√
FFFF0H	乘法/除法数据寄存器 A (L)	MDAL/MULA		R/W	—	—	√	0000H	√	√	√	√
FFFF1H												
FFFF2H	乘法/除法数据寄存器 A (H)	MDAH/MULB		R/W	—	—	√	0000H	√	√	√	√
FFFF3H												
FFFF4H	乘法/除法数据寄存器 B (H)	MDBH/MULOH		R/W	—	—	√	0000H	√	√	√	√
FFFF5H												
FFFF6H	乘法/除法数据寄存器 B (L)	MDBL/MULOL		R/W	—	—	√	0000H	√	√	√	√
FFFF7H												
FFFFEH	处理器模式控制寄存器	PMC		R/W	√	√	—	00H	√	√	√	√

注 不要直接操作该 SFR，因为它用于自编程库中。

备注 有关扩展 SFR（第二 SFR），参见表 3-6 扩展 SFR（第二 SFR）列表。

3.2.5 扩展特殊功能寄存器（第二 SFR：第二特殊功能寄存器）

与通用功能寄存器不同，每个扩展的 SFR（第二 SFR）都具有一个特殊功能。

扩展的 SFR 被分配至 F0000H 至 F07FFH 区域。除 SFR 区域（FFF00H 至 FFFFFH）的 SFR 之外的 SFR 被分配至该区域。但是，访问扩展 SFR 区域的指令比访问 SFR 区域的指令长一个字节。

可使用运算、发送和位操作指令对扩展 SFR 进行与通用寄存器相同的操作。根据 SFR 的类型可操作的位单元为 1, 8 和 16。

各操作位单元可被指定如下。

- 1 位操作
为 1 位操作指令操作数（!addr16.bit）说明由汇编器保留的符号。该操作也可以由一个地址进行指定。
- 8 位操作
为 8 位操作指令操作数（!addr16）说明由汇编器保留的符号。该操作也可以由一个地址进行指定。
- 16 位操作
为 16 位操作指令操作数（!addr16）说明由汇编器保留的符号。指定地址时，需说明一个偶地址。

表 3-6 给出了扩展 SFR 的列表。表中各项的含义如下。

- 符号
符号指示扩展 SFR 的地址。它在 RA78K0R 中是保留字，而在 CC78K0R 中则使用 #pragma sfr 伪指令将其定义为 sfr 变量。当给 78K0R 使用 RA78K0R, ID78K0R-QB 和 SM+ 时，符号可作为指令操作数写入。
- R/W
表明相应的扩展 SFR 是否可以读写。
R/W: 读/写使能
R: 只读
W: 只写
- 操作位单元
“√”表示可操作位单元（1, 8 或 16）。“—”表示操作不可用的位单元。
- 复位后
表示复位信号产生时各寄存器的状态。

注意事项 不要访问未分配扩展 SFR 的地址。

备注 有关 SFR 区域的 SFR，参见 3.2.4 特殊功能寄存器（SFR）。

表 3-6. 扩展 SFR (第二 SFR) 列表 (1/4)

地址	特殊功能寄存器 (SFR) 名称	符号	R/W	可操作位范围			复位后	78K0R/KC3-L(44 引脚)	78K0R/KC3-L(48 引脚)	78K0R/KD3-L	78K0R/KE3-L
				1 位	8 位	16 位					
F0017H	A/D 端口配置寄存器	ADPC	R/W	—	√	—	10H	√	√	√	√
F0030H	上拉电阻选项寄存器 0	PU0	R/W	√	√	—	00H	—	—	√	√
F0031H	上拉电阻选项寄存器 1	PU1	R/W	√	√	—	00H	√	√	√	√
F0033H	上拉电阻选项寄存器 3	PU3	R/W	√	√	—	00H	√	√	√	√
F0034H	上拉电阻选项寄存器 4	PU4	R/W	√	√	—	00H	√	√	√	√
F0035H	上拉电阻选项寄存器 5	PU5	R/W	√	√	—	00H	√	√	√	√
F0037H	上拉电阻选项寄存器 7	PU7	R/W	√	√	—	00H	√	√	√	√
F003CH	上拉电阻选项寄存器 12	PU12	R/W	√	√	—	00H	√	√	√	√
F003EH	上拉电阻选项寄存器 14	PU14	R/W	√	√	—	00H	—	—	—	√
F0043H	端口输入模式寄存器 3	PIM3	R/W	√	√	—	00H	√	√	√	√
F0047H	端口输入模式寄存器 7	PIM7	R/W	√	√	—	00H	√	√	√	√
F0048H	端口输入模式寄存器 8	PIM8	R/W	√	√	—	00H	√	√	√	√
F0053H	端口输出模式寄存器 3	POM3	R/W	√	√	—	00H	√	√	√	√
F0057H	端口输出模式寄存器 7	POM7	R/W	√	√	—	00H	√	√	√	√
F0060H	噪声滤波器使能寄存器 0	NFEN0	R/W	√	√	—	00H	√	√	√	√
F0061H	噪声滤波器使能寄存器 1	NFEN1	R/W	√	√	—	00H	√	√	√	√
F0062H	噪声滤波器使能寄存器 2	NFEN2	R/W	√	√	—	00H	√	√	√	√
F00E0H	乘法/除法数据寄存器 C (L)	MDCL	R	—	—	√	0000H	√	√	√	√
F00E2H	乘法/除法数据寄存器 C (H)	MDCH	R	—	—	√	0000H	√	√	√	√
F00E8H	乘法/除法控制寄存器	MDUC	R/W	√	√	—	00H	√	√	√	√
F00F0H	外设使能寄存器 0	PER0	R/W	√	√	—	00H	√	√	√	√
F00F1H	外设使能寄存器 1	PER1	R/W	√	√	—	00H	√	√	√	√
F00F2H	外设使能寄存器 2	PER2	R/W	√	√	—	00H	√	√	√	√
F00F3H	运行速度模式控制寄存器	OSMC	R/W	—	√	—	00H	√	√	√	√
F00F4H	稳压器模式控制寄存器	RMC	R/W	—	√	—	00H	√	√	√	√
F00F6H	20 MHz 内部高速振荡控制寄存器	DSCCTL	R/W	√	√	—	00H	√	√	√	√
F00FEH	BCD 调整结果寄存器	BCDADJ	R	—	√	—	00H	√	√	√	√
F0100H	串口状态寄存器 00	SSR00L	SSR00	R	—	√	0000H	√	√	√	√
F0101H		—			—	—		√	√	√	√
F0102H	串口状态寄存器 01	SSR01L	SSR01	R	—	√	0000H	√	√	√	√
F0103H		—			—	—		√	√	√	√
F0104H	串口状态寄存器 02	SSR02L	SSR02	R	—	√	0000H	√	√	√	√
F0105H		—			—	—		√	√	√	√
F0106H	串口状态寄存器 03	SSR03L	SSR03	R	—	√	0000H	√	√	√	√
F0107H		—			—	—		√	√	√	√
F0108H	串口标志清除触发寄存器 00	SIR00L	SIR00	R/W	—	√	0000H	√	√	√	√
F0109H		—			—	—		√	√	√	√
F010AH	串口标志清除触发寄存器 01	SIR01L	SIR01	R/W	—	√	0000H	√	√	√	√
F010BH		—			—	—		√	√	√	√

表 3-6. 扩展 SFR（第二 SFR）列表（2/4）

地址	特殊功能寄存器（SFR）名称	符号		R/W	可操作位范围			复位后	78K0R/KC3-L(44 引脚)	78K0R/KC3-L(48 引脚)	78K0R/KD3-L	78K0R/KE3-L
					1 位	8 位	16 位					
F010CH	串口标志清除触发寄存器 02	SIR02L	SIR02	R/W	—	√	√	0000H	√	√	√	√
F010DH		—			—	—			√	√	√	√
F010EH	串口标志清除触发寄存器 03	SIR03L	SIR03	R/W	—	√	√	0000H	√	√	√	√
F010FH		—			—	—			√	√	√	√
F0110H	串口模式寄存器 00	SMR00		R/W	—	—	√	0020H	√	√	√	√
F0111H												
F0112H	串口模式寄存器 01	SMR01		R/W	—	—	√	0020H	√	√	√	√
F0113H												
F0114H	串口模式寄存器 r 02	SMR02		R/W	—	—	√	0020H	√	√	√	√
F0115H												
F0116H	串口模式寄存器 r 03	SMR03		R/W	—	—	√	0020H	√	√	√	√
F0117H												
F0118H	串行通信操作设置寄存器 00	SCR00		R/W	—	—	√	0087H	√	√	√	√
F0119H												
F011AH	串行通信操作设置寄存器 01	SCR01		R/W	—	—	√	0087H	√	√	√	√
F011BH												
F011CH	串行通信操作设置寄存器 02	SCR02		R/W	—	—	√	0087H	√	√	√	√
F011DH												
F011EH	串行通信操作设置寄存器 03	SCR03		R/W	—	—	√	0087H	√	√	√	√
F011FH												
F0120H	串行通道使能状态寄存器 0	SE0L	SE0	R	√	√	√	0000H	√	√	√	√
F0121H		—			—	—			√	√	√	√
F0122H	串行通道开始寄存器 0	SS0L	SS0	R/W	√	√	√	0000H	√	√	√	√
F0123H		—			—	—			√	√	√	√
F0124H	串行通道停止寄存器 0	ST0L	ST0	R/W	√	√	√	0000H	√	√	√	√
F0125H		—			—	—			√	√	√	√
F0126H	串行时钟选择寄存器 0	SPS0L	SPS0	R/W	—	√	√	0000H	√	√	√	√
F0127H		—			—	—			√	√	√	√
F0128H	串行输出寄存器 0	SO0		R/W	—	—	√	0F0FH	√	√	√	√
F0129H												
F012AH	串行输出使能寄存器 0	SOE0L	SOE0	R/W	√	√	√	0000H	√	√	√	√
F012BH		—			—	—			√	√	√	√
F0134H	串行输出电平寄存器 0	SOL0L	SOL0	R/W	—	√	√	0000H	√	√	√	√
F0135H		—			—	—			√	√	√	√
F0180H	定时器计数器寄存器 00	TCR00		R	—	—	√	FFFFH	√	√	√	√
F0181H												
F0182H	定时器计数器寄存器 01	TCR01		R	—	—	√	FFFFH	√	√	√	√
F0183H												
F0184H	定时器计数器寄存器 02	TCR02		R	—	—	√	FFFFH	√	√	√	√
F0185H												

表 3-6. 扩展 SFR（第二 SFR）列表（3/4）

地址	特殊功能寄存器（SFR）名称	符号 I	R/W	可操作位范围			复位后 t	78K0R/KC3-L(44 引脚)	78K0R/KC3-L(48 引脚)	78K0R/KD3-L	78K0R/KE3-L
				1 位	8 位	16 位					
F0186H	定时器计数器寄存器 03	TCR03	R	—	—	√	FFFFH	√	√	√	√
F0187H				—	—	√					
F0188H	定时器计数器寄存器 04	TCR04	R	—	—	√	FFFFH	√	√	√	√
F0189H				—	—	√					
F018AH	定时器计数器寄存器 05	TCR05	R	—	—	√	FFFFH	√	√	√	√
F018BH				—	—	√					
F018CH	定时器计数器寄存器 06	TCR06	R	—	—	√	FFFFH	√	√	√	√
F018DH				—	—	√					
F018EH	定时器计数器寄存器 07	TCR07	R	—	—	√	FFFFH	√	√	√	√
F018FH				—	—	√					
F0190H	定时器模式寄存器 00	TMR00	R/W	—	—	√	0000H	√	√	√	√
F0191H				—	—	√					
F0192H	定时器模式寄存器 01	TMR01	R/W	—	—	√	0000H	√	√	√	√
F0193H				—	—	√					
F0194H	定时器模式寄存器 02	TMR02	R/W	—	—	√	0000H	√	√	√	√
F0195H				—	—	√					
F0196H	定时器模式寄存器 03	TMR03	R/W	—	—	√	0000H	√	√	√	√
F0197H				—	—	√					
F0198H	定时器模式寄存器 04	TMR04	R/W	—	—	√	0000H	√	√	√	√
F0199H				—	—	√					
F019AH	定时器模式寄存器 05	TMR05	R/W	—	—	√	0000H	√	√	√	√
F019BH				—	—	√					
F019CH	定时器模式寄存器 06	TMR06	R/W	—	—	√	0000H	√	√	√	√
F019DH				—	—	√					
F019EH	定时器模式寄存器 07	TMR07	R/W	—	—	√	0000H	√	√	√	√
F019FH				—	—	√					
<R>	定时器状态寄存器 00	TSR00L	TSR00	R	—	√	√	0000H	√	√	√
		—			—	—					
<R>	定时器状态寄存器 01	TSR01L	TSR01	R	—	√	√	0000H	√	√	√
		—			—	—					
<R>	定时器状态寄存器 02	TSR02L	TSR02	R	—	√	√	0000H	√	√	√
		—			—	—					
<R>	定时器状态寄存器 03	TSR03L	TSR03	R	—	√	√	0000H	√	√	√
		—			—	—					
<R>	定时器状态寄存器 04	TSR04L	TSR04	R	—	√	√	0000H	√	√	√
		—			—	—					
<R>	定时器状态寄存器 05	TSR05L	TSR05	R	—	√	√	0000H	√	√	√
		—			—	—					
<R>	定时器状态寄存器 06	TSR06L	TSR06	R	—	√	√	0000H	√	√	√
		—			—	—					

表 3-6. 扩展 SFR（第二 SFR）列表（4/4）

	地址	特殊功能寄存器（SFR）名称	符号		R/W	可操作位范围			复位后	78K0R/KC3-L(44 引脚)	78K0R/KC3-L(48 引脚)	78K0R/KD3-L	78K0R/KE3-L
						1 位	8 位	16 位					
<R>	F01AEH	定时器状态寄存器 07	TSR07L	TSR07	R	—	√	√	0000H	√	√	√	√
	F01AFH		—			—							
<R>	F01B0H	定时器通道使能状态寄存器 0	TE0L	TE0	R	√	√	√	0000H	√	√	√	√
	F01B1H		—			—							
<R>	F01B2H	定时器通道开始寄存器 0	TS0L	TS0	R/W	√	√	√	0000H	√	√	√	√
	F01B3H		—			—							
<R>	F01B4H	定时器通道停止寄存器 0	TT0L	TT0	R/W	√	√	√	0000H	√	√	√	√
	F01B5H		—			—							
<R>	F01B6H	定时器时钟选择寄存器 0	TPS0L	TPS0	R/W	—	√	√	0000H	√	√	√	√
	F01B7H		—			—							
<R>	F01B8H	定时器输出寄存器 0	TO0L	TO0	R/W	—	√	√	0000H	√	√	√	√
	F01B9H		—			—							
<R>	F01BAH	定时器输出使能寄存器 0	TOE0L	TOE0	R/W	√	√	√	0000H	√	√	√	√
	F01BBH		—			—							
<R>	F01BCH	定时器输出电平寄存器 0	TOL0L	TOL0	R/W	—	√	√	0000H	√	√	√	√
	F01BDH		—			—							
<R>	F01BEH	定时器输出模式寄存器 0	TOM0L	TOM0	R/W	—	√	√	0000H	√	√	√	√
	F01BFH		—			—							
	F0230H	IICA 控制寄存器 0	IICCTL0		R/W	√	√	—	00H	—	√	√	√
	F0231H	IICA 控制寄存器 1	IICCTL1		R/W	√	√	—	00H	—	√	√	√
	F0232H	IICA 低电平宽度设置寄存器	IICWL		R/W	—	√	—	FFH	—	√	√	√
	F0233H	IICA 高电平宽度设置寄存器	IICWH		R/W	—	√	—	FFH	—	√	√	√
	F0234H	从设备地址寄存器	SVA		R/W	—	√	—	00H	—	√	√	√
	F0240H	可编程增益放大器控制寄存器	OAM		R/W	√	√	—	00H	√	√	√	√
	F0241H	比较器 0 控制寄存器	C0CTL		R/W	√	√	—	00H	√	√	√	√
	F0242H	比较器 0 内部参考电压设置寄存器	C0RVM		R/W	√	√	—	00H	√	√	√	√
	F0243H	比较器 1 控制寄存器 r	C1CTL		R/W	√	√	—	00H	√	√	√	√
	F0244H	比较器 1 内部参考电压设置寄存器	C1RVM		R/W	√	√	—	00H	√	√	√	√

备注 有关 SFR 区域中的 SFR，参见 表 3-5 SFR 列表。

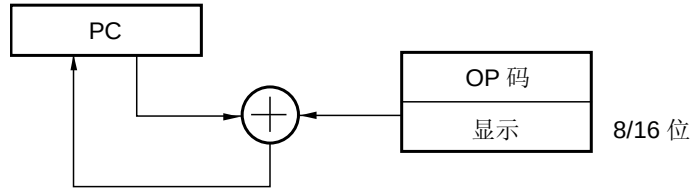
3.3 指令地址寻址

3.3.1 相对寻址

[功能]

相对寻址在程序计数器（PC）中将指令字（有符号补码数据：-128 至 +127 或 -32768 至 +32767）中包含的加上偏移值的结果存储至程序计数器（PC）的值（下一条指令的起始地址），并指定将要用作转移目的地的程序地址。相对寻址仅用于转移指令。

图 3-16. 相对寻址略图



3.3.2 直接寻址

[功能]

直接寻址在程序计数器中存储指令字的立即数，并指定将要用作转移目的地的程序地址。

对于立即寻址，CALL !!addr20 或 BR !!addr20 用于指定 20-位地址，CALL !addr16 或 BR !addr16 用于指定 16 位地址。指定 16 位地址时将高 4 位设为 0000。

图 3-17. CALL !!addr20/BR !!addr20 的示例

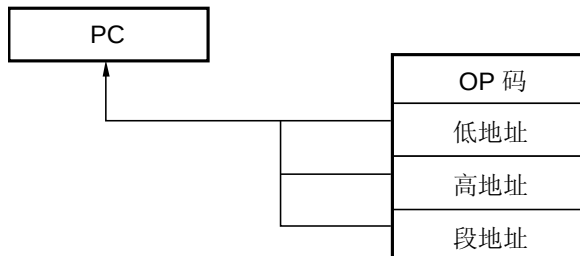
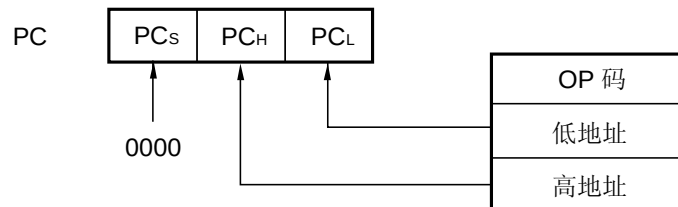


图 3-18. CALL !addr16/BR !addr16 的示例



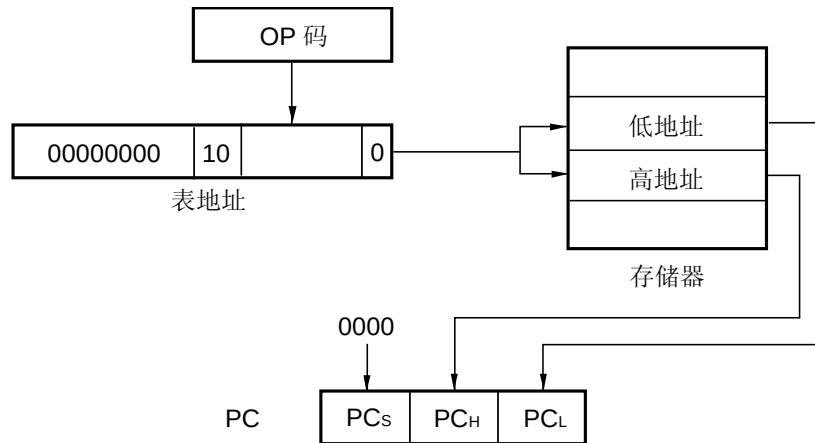
3.3.3 表间接寻址

[功能]

表间接寻址用指令字中的 5-位立即数在 CALLT 表区域（0080H 至 00BFH）指定表地址、将程序计数器（PC）中的表地址和下一个地址存储为 16 位数据并指定程序地址。表间接寻址仅用于 CALLT 指令。

78K0R 微控制器中，分支仅可启用从 00000H 至 0FFFFH 的 64 KB 空间。

图 3-19. 表间接寻址略图

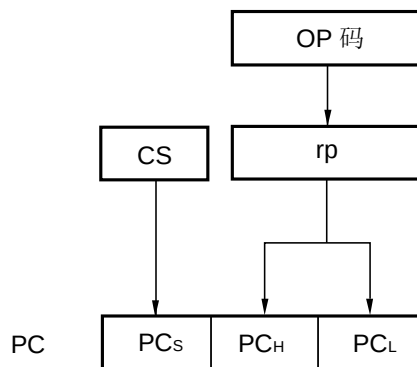


3.3.4 寄存器直接寻址

[功能]

寄存器直接寻址在程序计数器（PC）中将通用寄存器对（AX/BC/DE/HL）和用指令字指定的当前寄存器 CS 寄存器的内容存储为 20-位数据，并指定程序地址。寄存器直接寻址仅用于 CALL AX、BC、DE、HL 和 BR AX 指令。

图 3-20. 寄存器直接寻址略图

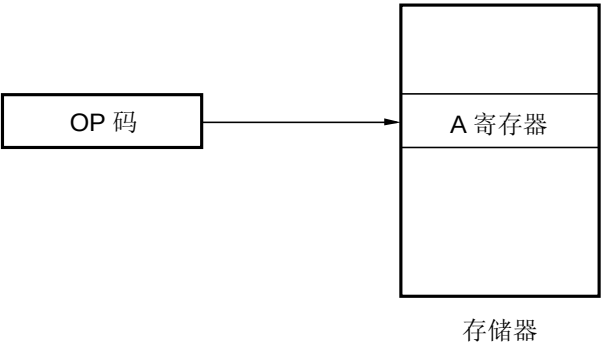


3.4 处理数据地址寻址

3.4.1 隐式寻址

- [功能]
访问具有专用功能的寄存器（例如累加器）的指令直接由指令字指定，不需在指令字中使用任何寄存器指定字段。
- [操作数格式]
因为隐式寻址可自动用于指令，所以不需要特殊的操作数格式。
隐式寻址仅可用于 MULU X。

图 3-21. 隐式寻址略图

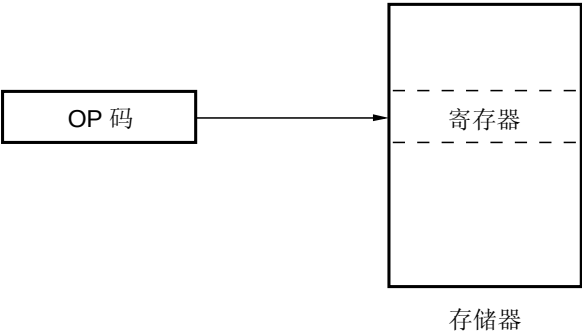


3.4.2 寄存器寻址

- [功能]
寄存器寻址将通用寄存器视为操作数进行访问。 3-位长度的指令字用于选择 8 位寄存器；2-位长度的指令字用于选择 16 位寄存器。
- [操作数格式]

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

图 3-22. 寄存器寻址略图



3.4.3 直接寻址

[功能]

直接寻址使用指令字中的立即数作为操作数地址直接指定目标地址。

[操作数格式]

标识符	说明
ADDR16	标记或16位立即数（仅 F0000H 至 FFFFFH 的空间可指定）
ES: ADDR16	标记或16位立即数（高 4-位地址由 ES 寄存器指定）

图 3-23. ADDR16 的示例

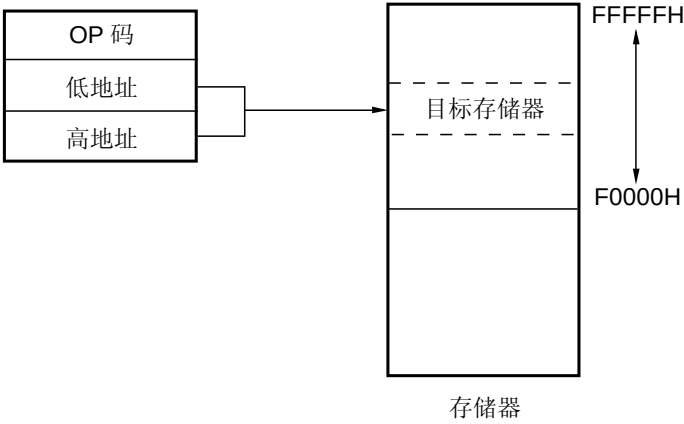
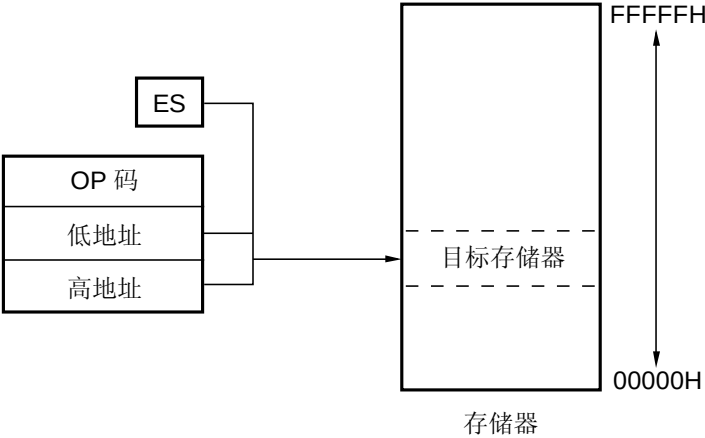


图 3-24. ES:ADDR16 的示例



3.4.4 短直接寻址

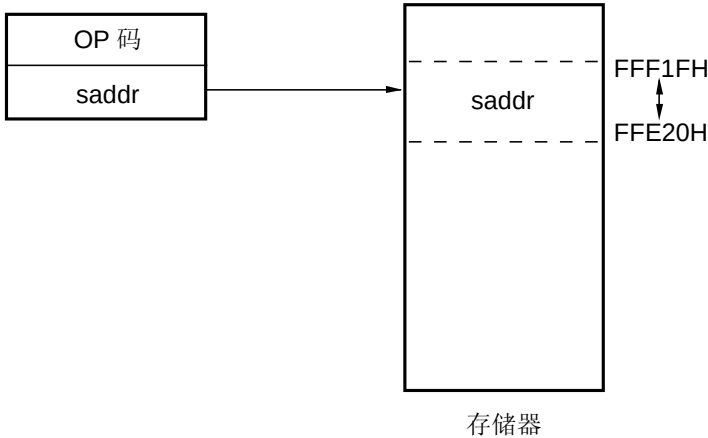
[功能]

在指令字中，短直接寻址使用 8 位数据直接指定目标地址。 该寻址类型仅用于从 FFE20H 至 FFF1FH 的空间。

[操作数格式]

标识符	说明
SADDR	标记、FFE20H 至 FFF1FH 立即数或 0FE20H 至 0FF1FH 立即数 (仅 FFE20H 至 FFF1FH 的空间可指定)
SADDRP	标记、FFE20H 至 FFF1FH 立即数或 0FE20H 至 0FF1FH 立即数 (仅偶地址) (仅 FFE20H 至 FFF1FH 的空间可指定)

图 3-25. 短直接寻址略图



备注 SADDR 和 SADDRP 用于使用 16 位的立即数（实际地址的高 4 位被忽略）说明地址 FE20H 至 FF1FH 的值，且使用 20-位的立即数说明地址 FFE20H 至 FFF1FH 的值。
无论是否使用 SADDR 或 SADDRP，都要为存储器指定 FFE20H 至 FFF1FH 空间内的地址。

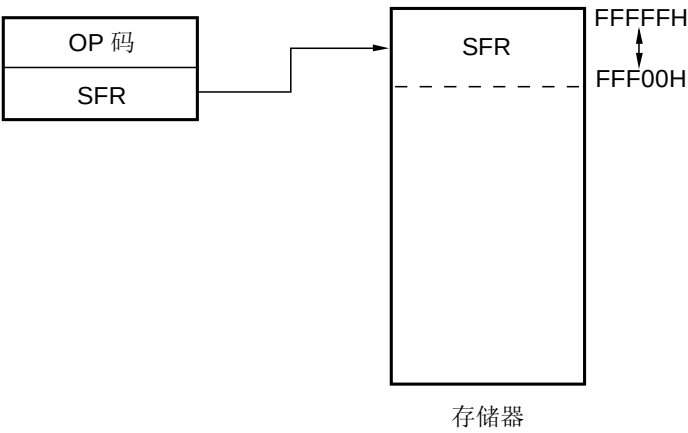
3.4.5 SFR 寻址

[功能]
在指令字中，SFR 寻址使用 8 位数据直接指定目 SFR 标地址。 该寻址类型仅用于从 FFF00H 至 FFFFFH 的空间。

[操作数格式]

标识符	说明
SFR	SFR 名称
SFRP	16位可操作 SFR 名称（仅偶地址）

图 3-26. SFR 寻址略图



3.4.6 寄存器间接寻址

[功能]

寄存器间接寻址使用指定为操作数地址的寄存器对的内容直接指定目标地址。

[操作数格式]

标识符	说明
–	[DE], [HL]（仅 F0000H 至 FFFFFH 的空间可指定）
–	ES:[DE], ES:[HL]（高4-位地址由 ES 寄存器指定）

图 3-27. [DE], [HL]的示例

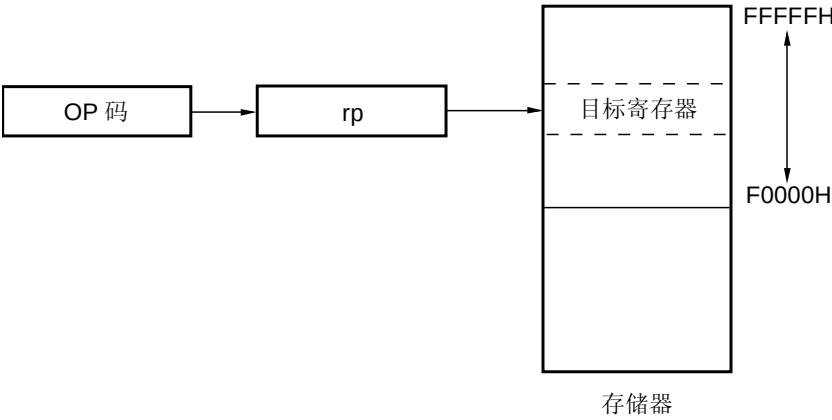
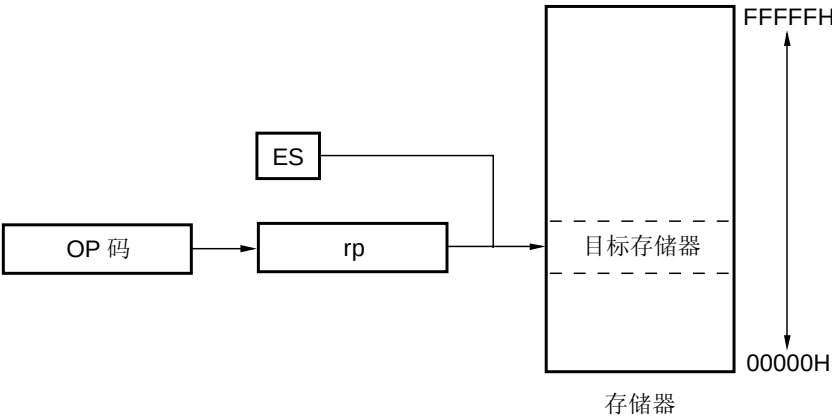


图 3-28. ES: [DE], ES: [HL]的示例



3.4.7 基址寻址

[功能]

基址寻址使用指令字指定的寄存器对的内容作为基地址，8 位立即数或 16 位立即数作为偏移地址。 这些值的和用来指定目标地址。

[操作数格式]

标识符	说明
–	[HL + 字节], [DE + 字节], [SP + 字节]（仅F0000H 至 FFFFFH的空间可指定）
–	字[B], 字[C]（仅F0000H 至 FFFFFH的空间可指定）
–	字[BC]（仅F0000H 至 FFFFFH 的空间可指定）
–	ES:[HL + 字节], ES:[DE + 字节]（高4-位地址由 ES 寄存器指定）
–	ES:字[B], ES:字[C]（高4-位地址由 ES 寄存器指定）
–	ES:字[BC]（高4-位地址由 ES 寄存器指定）

图 3-29. [SP+字节]的示例

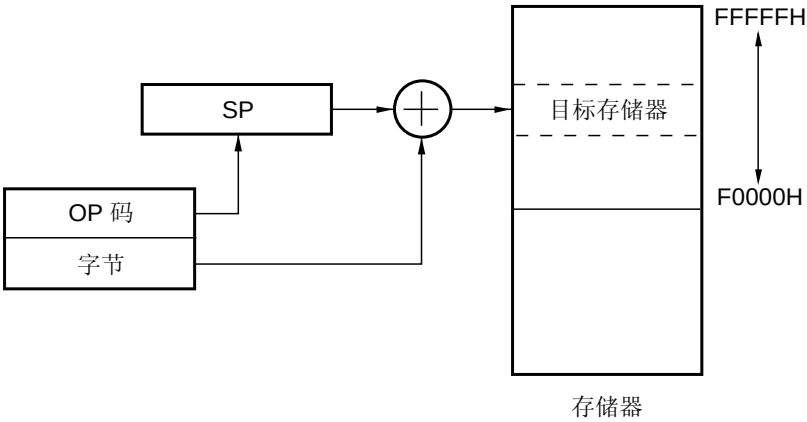


图 3-30. [HL + 字节], [DE + 字节]的示例

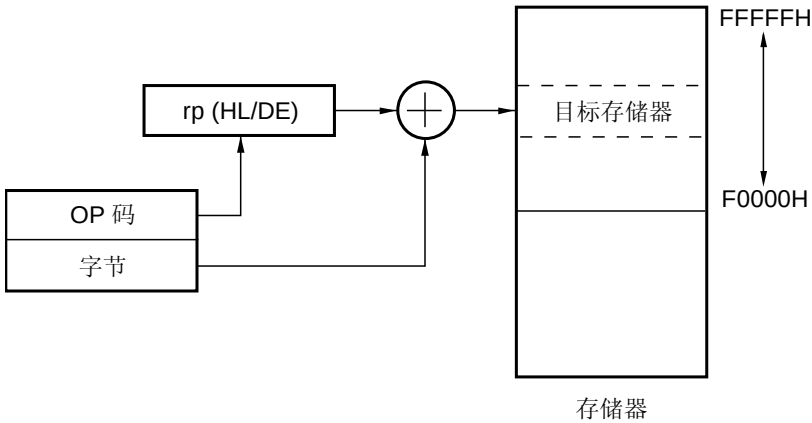


图 3-31. 字[B], 字[C]的示例

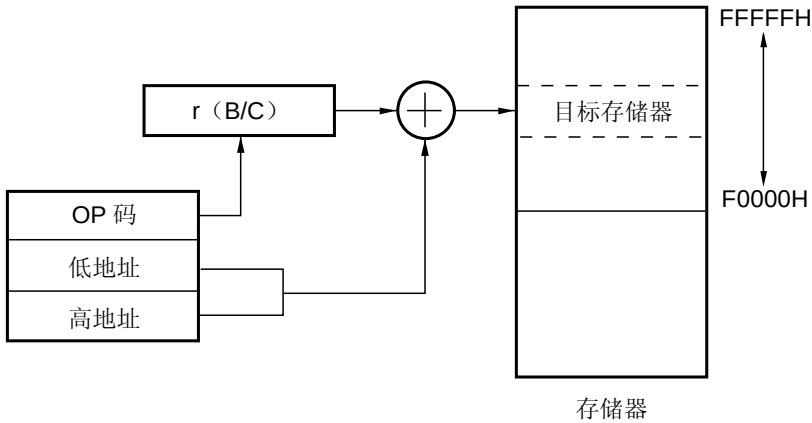


图 3-32. 字[BC]的示例

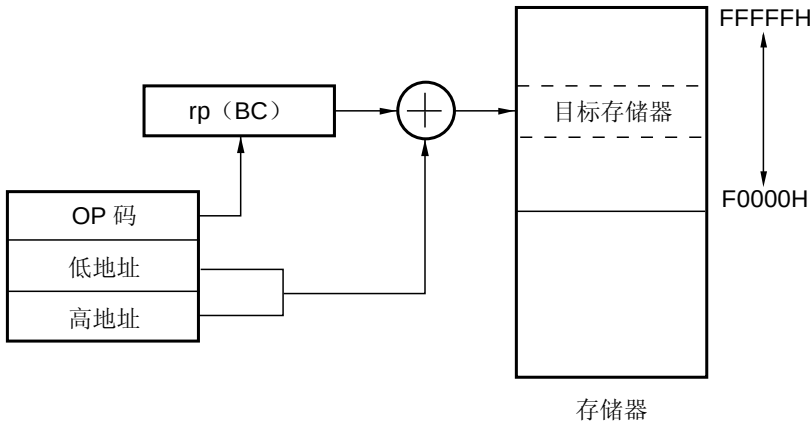


图 3-33. ES: [HL + 字节], ES: [DE + 字节]的示例

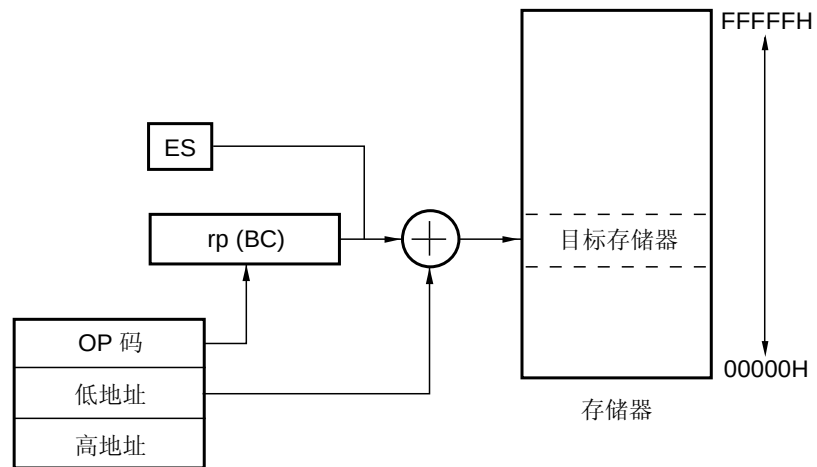


图 3-34. ES: 字[B], ES: 字[C]的示例

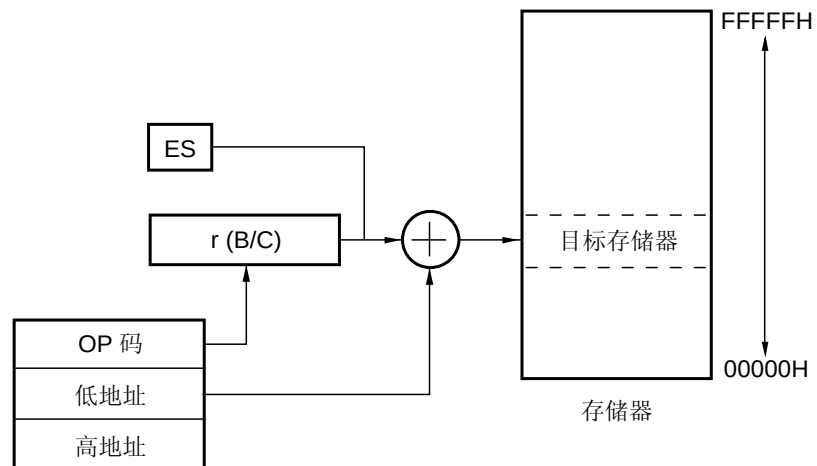
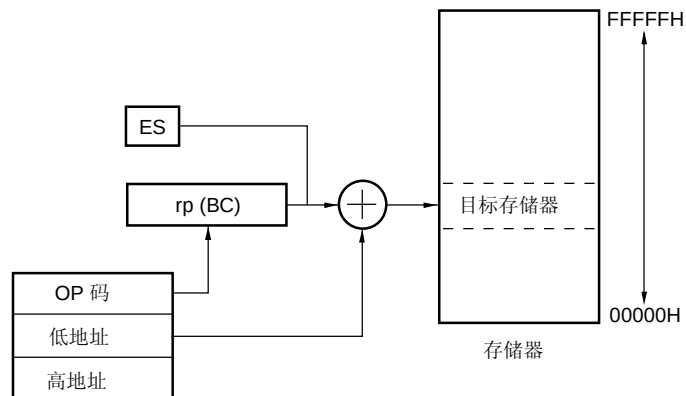


图 3-35. ES: 字[BC]的示例



3.4.8 基址变址寻址

[功能]

基址变址寻址使用指令字指定的寄存器对的内容作为基地址，同样地使用由指令字指定的寄存器 B 或 C 的内容作为偏移地址。 这些值的和用来指定目标地址。

[操作数格式]

标识符	说明
—	[HL+B], [HL+C]（仅 F0000H 至 FFFFFH 的空间可指定）
—	ES:[HL+B], ES:[HL+C]（高 4-位地址由 ES 寄存器指定）

图 3-36. [HL+B], [HL+C]的示例

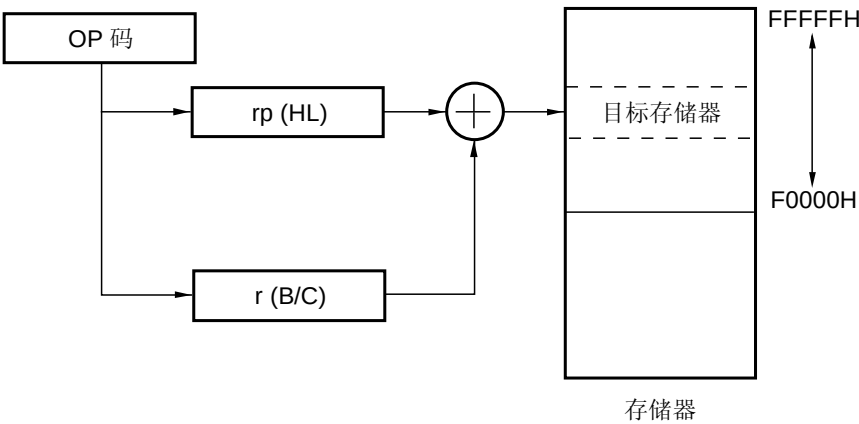
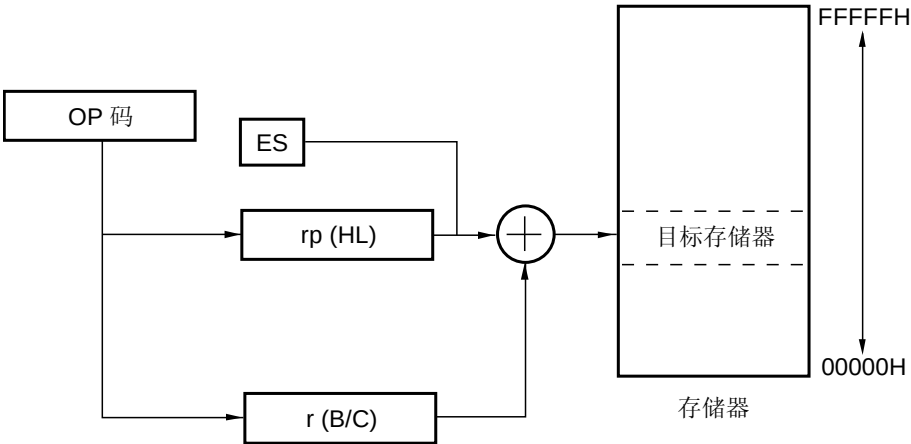


图 3-37. ES: [HL+B], ES: [HL+C]的示例



3.4.9 堆栈寻址

[功能]

堆栈区域用堆栈指针（SP）的内容间接进行寻址。 当执行 PUSH、POP、子程序调用和返回指令时，或产生中断请求时保持/恢复寄存器时自动使用该寻址。

堆栈寻址仅用于内部 RAM 区域。

[操作数格式]

标识符	说明
—	PUSH AX/BC/DE/HL POP AX/BC/DE/HL CALL/CALLT RET BRK RETB (产生的中断请求) RETI

第四章 端口功能

4.1 端口功能

引脚 I/O 缓存器电源由产品决定。 电源和引脚之间的关系如下所示。

表 4-1. 引脚 I/O 缓存器电源 (AVREF, VDD)

- 78K0R/KC3-L: 44 引脚塑封 LQFP (10x10)
48 引脚塑封 TQFP (密间距) (7x7)
- 78K0R/KD3-L: 52 引脚塑封 LQFP (10x10)

电源	相应的引脚
AVREF	P20 至 P27、P150 至 P152 [※] 、P80 至 P83
VDD	• 除 P20 至 P27、P150 至 P152 [※] 、P80 至 P83 之外的端口引脚 • 除端口引脚之外的引脚

注 78K0R/KC3-L 的 44 引脚产品不具有 P152 引脚。

表 4-2. 引脚 I/O 缓存器电源 (AVREF, EVDD, VDD)

- 78K0R/KE3-L: 64-引脚塑封 FBGA (5x5)
64-引脚塑封 TQFP (密间距) (7x7)
64-引脚塑封 LQFP (密间距) (10x10)
64-引脚塑封 LQFP (12x12)

电源	相应的引脚
AVREF	P20 至 P27、P150 至 P153、P80 至 P83
EVDD	• 除 P20 至 P27、P150 至 P153、P80 至 P83 和 P121 至 P124 之外的端口引脚 • RESET 引脚和 FLMD0 引脚
VDD	• P121 至 P124 • 除端口引脚之外的引脚 (不包括 RESET 引脚和 FLMD0 引脚)

78K0R/Kx3-L 微控制器提供数字 I/O 端口，允许多种控制操作。 各端口的功能如表 4-3 所示。
除了用作数字 I/O 端口，这些端口还具有多种复用功能。 有关复用功能的详情，参见 第二章 引脚功能。

表 4-2. 端口功能 (1/2)

KC3-L(44 引脚)	KC3-L(48 引脚)	KD3-L	KE3-L	功能名称	I/O	功能	复位后	复用功能
—	—	√	√	P00	I/O	端口 0。 I/O 端口。 输入/输出可按 1 位为单元指定。 片上上拉电阻的使用可通过软件设置进行指定。	输入端口	TI00
—	—	√	√	P01				TO00
√	√	√	√	P10	I/O	端口 1。 I/O 端口。 输入/输出可按 1 位为单元指定。 片上上拉电阻的使用可通过软件设置进行指定。	输入端口	TI02/TO02
√	√	√	√	P11				TI03/TO03
√	√	√	√	P12				TI04/TO04/ RTCDIV/RTCCL
√	√	√	√	P13				TI05/TO05
—	—	—	√	P14				TI06/TO06
—	—	—	√	P15				TI07/TO07
—	—	—	√	P16				—
—	—	—	√	P17				—
√	√	√	√	P20 至 P27	I/O	端口 2。 I/O 端口。 输入/输出可按 1 位为单元指定。	数字输入端口	ANI0 to ANI7
√	√	√	√	P30	I/O	端口 3。 I/O 端口。 P31 和 P32 的输入可设置至 TTL 缓存器。 P30 至 P32 的输出可设置至 N-通道漏极开路输出 (V _{DD} 耐压)。 输入/输出可按 1 位为单元指定。 片上上拉电阻的使用可通过软件设置进行指定。	输入端口	SO10/TxD1
√	√	√	√	P31				SI10/RxD1/SDA10 /INTP1
√	√	√	√	P32				SCK10/SCL10/ INTP2
—	—	—	√	P33				—
√	√	√	√	P40 ^{※2}	I/O	端口 4。 I/O 端口。 输入/输出可按 1 位为单元指定。 片上上拉电阻的使用可通过软件设置进行指定。	输入端口	TOOL0
√	√	√	√	P41				TOOL1
—	—	—	√	P42				—
—	—	—	√	P43				—
√	√	√	√	P50	I/O	端口 5。 I/O 端口。 输入/输出可按 1 位为单元指定。 片上上拉电阻的使用可通过软件设置进行指定。	输入端口	TI06/TO06 ^{※1}
√	√	√	√	P51				TI07/TO07 ^{※1}
√	√	√	√	P52				RTC1HZ/SLTI/ SLTO
—	—	—	√	P53				—

- 注 1. TI06/TO06 和 TI07/TO07 仅能在 78K0R/KC3-L 和 78K0R/KD3-L 中共享。 78K0R/KE3-L 不具有共享功能。
2. 如果通过使用选项字节使能片上调试, 则一定要在外部上拉 P40/TOOL0 引脚 (参见 2.2.5 P40 至 P43 (端口 4) 中的 注意事项)。

表 4-2. 端口功能 (2/2)

KC3-L(44 引 脚)	KC3-L(48 引 脚)	KD3-L	KE3-L	功能名称	I/O	功能	复位后	复用功能
—	√	√	√	P60	I/O	端口 6。 I/O 端口。 P60 和 P61 的输出为 N-通道漏极开路输出(6 V 耐压)。 输入/输出可按 1 位为单元指定。	输入端口	SCL0
—	√	√	√	P61				SDA0
√	√	√	√	P70	I/O	端口 7。 I/O 端口。 P71, P72, P74 和 P75 的输入可设置至 TTL 缓存器。 P70, P72, P73 和 P75 的输出可设置至 N-通道漏极开路输出 (V _{DD} 耐压)。 输入/输出可按 1 位为单元指定。 片上上拉电阻的使用可通过软件设置进行指定。	输入端口	KR0/SO01/INTP4
√	√	√	√	P71				KR1/SI01/INTP5
√	√	√	√	P72				KR2/SCK01/INTP6
√	√	√	√	P73				KR3/SO00/TxD0
√	√	√	√	P74				KR4/SI00/RxD0
√	√	√	√	P75				KR5/SCK00
—	—	√	√	P76				KR6
—	—	√	√	P77				KR7
√	√	√	√	P80	I/O	端口 8。 I/O 端口。 输入/输出可以 1 位为单元来指定。 P80 至 P83 的输入可设为比较器输入或可编程增益放大器输入。	模拟输入	CMP0P/INTP3/PGAI
√	√	√	√	P81				CMP0M
√	√	√	√	P82				CMP1P/INTP7
√	√	√	√	P83				CMP1M
√	√	√	√	P120	I/O	端口 12。 I/O 端口和输入端口。 仅对于 P120, 输入/输出可按位为单元指定。 仅对于 P120, 片上上拉电阻的使用可通过软件设置进行指定。	输入端口	INTP0/EXLVI
√	√	√	√	P121				X1
√	√	√	√	P122				X2/EXCLK
√	√	√	√	P123				XT1
√	√	√	√	P124				XT2
—	√	√	√	P140	输出	端口 14。	输出端口	PCLBUZ0
—	—	—	√	P141	I/O	输出端口和 I/O 端口。 仅对 P141, 可指定输入/输出。 仅对于 P141, 片上上拉电阻的使用可通过软件设置进行指定。	输入端口	PCLBUZ1
√	√	√	√	P150	I/O	端口 15。 I/O 端口。 输入/输出可按 1 位为单元指定。	数字输入端口	ANI8
√	√	√	√	P151				ANI9
—	√	√	√	P152				ANI10
—	—	—	√	P153				ANI11

4.2 端口配置

端口包括以下硬件。

表 4-4. 端口配置

项目	配置
控制寄存器	• 78K0R/KC3-L (44 引脚产品) - 端口模式寄存器 (PM1 至 PM5、PM7、PM8、PM12、PM15) - 端口寄存器 (P1 至 P5、P7、P8、P12、P15) - 上拉电阻选项寄存器 (PU1、PU3 至 PU5、PU7、PU12) - 端口输入模式寄存器 (PIM3、PIM7、PIM8) - 端口输出模式寄存器 (POM3、POM7) - A/D 端口配置寄存器 (ADPC) • 78K0R/KC3-L (48 引脚产品) - 端口模式寄存器 (PM1 至 PM8、PM12、PM15) - 端口寄存器 (P1 至 P8、P12、P14、P15) - 上拉电阻选项寄存器 (PU1、PU3 至 PU5、PU7、PU12) - 端口输入模式寄存器 (PIM3、PIM7、PIM8) - 端口输出模式寄存器 (POM3、POM7) - A/D 端口配置寄存器 (ADPC) • 78K0R/KD3-L - 端口模式寄存器 (PM0 至 PM8、PM12、PM15) - 端口寄存器 (P0 至 P8、P12、P14、P15) - 上拉电阻选项寄存器 (PU0、PU1、PU3 至 PU5、PU7、PU12) - 端口输入模式寄存器 (PIM3、PIM7、PIM8) - 端口输出模式寄存器 (POM3、POM7) - A/D 端口配置寄存器 (ADPC) • 78K0R/KE3-L - 端口模式寄存器 (PM0 至 PM8、PM12、PM14、PM15) - 端口寄存器 (P0 至 P8、P12、P14、P15) - 上拉电阻选项寄存器 (PU0、PU1、PU3 至 PU5、PU7、PU12、PU14) - 端口输入模式寄存器 (PIM3、PIM7、PIM8) - 端口输出模式采集器 (POM3、POM7) - A/D 端口配置寄存器 (ADPC)
端口	• 78K0R/KC3-L (44 引脚产品) - 总计: 37 (CMOS I/O: 33、CMOS 输入: 4) • 78K0R/KC3-L (48 引脚产品) - 总计: 41 (CMOS I/O: 34、CMOS 输入: 4、CMOS 输出: 1、N-通道漏极开路 I/O: 2) • 78K0R/KD3-L - 总计: 45 (CMOS I/O: 38、CMOS 输入: 4、CMOS 输出: 1、N-通道漏极开路 I/O: 2) • 78K0R/KE3-L - 总计: 55 (CMOS I/O: 48、CMOS 输入: 4、CMOS 输出: 1、N-通道漏极开路 I/O: 2)
上拉电阻	• 78K0R/KC3-L (44 引脚产品) 总计: 19 • 78K0R/KC3-L (48 引脚产品) 总计: 19 • 78K0R/KD3-L 总计: 23 • 78K0R/KE3-L 总计: 32

4.2.1 端口 0

	78K0R/KC3-L (44引脚) (μ PD78F100y: y = 0 至 3)	78K0R/KC3-L (48引脚) (μ PD78F100y: y = 1 至 3)	78K0R/KD3-L (μ PD78F100y: y = 4 至 6)	78K0R/KE3-L (μ PD78F100y: y = 7 至 9)
P00/TI00	–	–	√	√
P11/TO00	–	–	√	√

端口 0 是带有输出锁存器的 I/O 端口。端口 0 可用端口模式寄存器 0 (PM0) 设置为以 1 位为单元的输入或输出模式。当 P00 和 P01 引脚用作输入端口时，片上上拉电阻可用上拉电阻选项寄存器 0 (PU0) 按 1 位为单元指定。

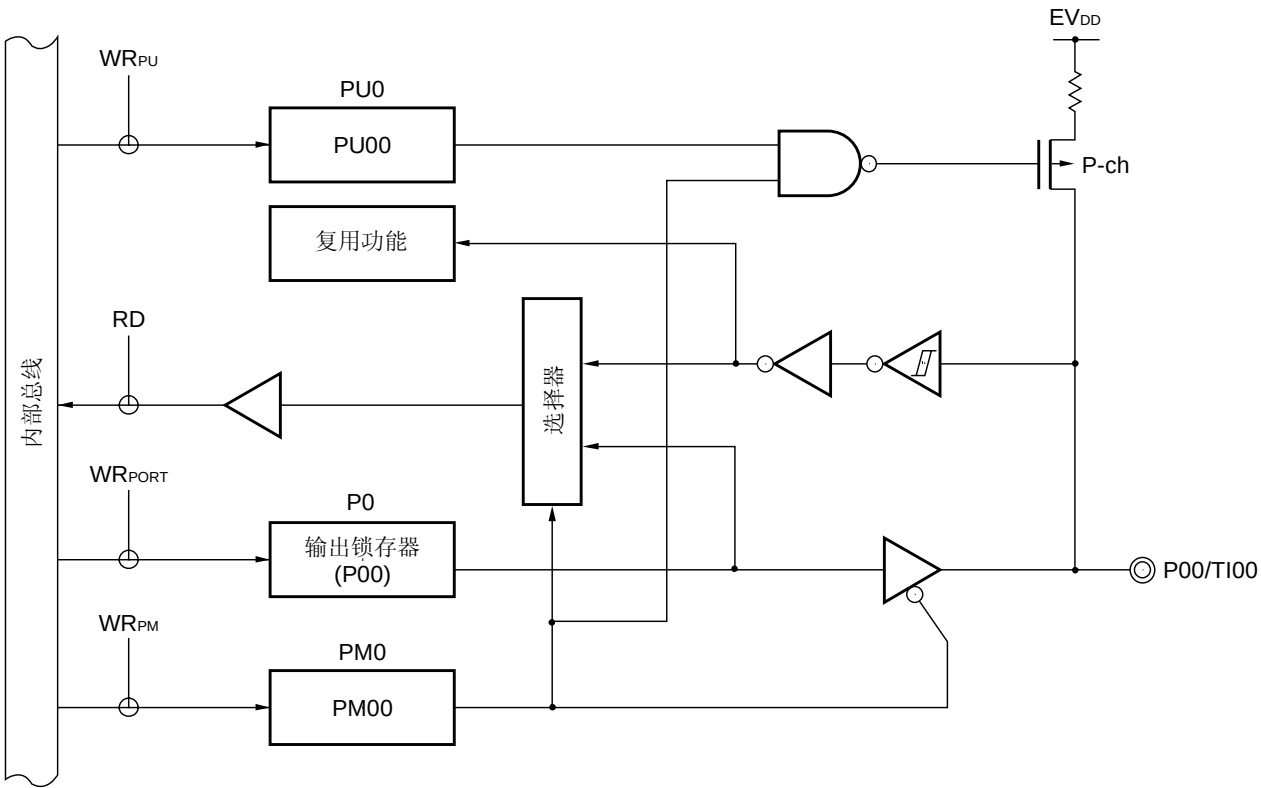
该端口还可以用作定时器 I/O。

产生复位信号后将端口 0 设为输入模式。

图 4-1 和 4-2 显示了端口 0 的框图。

注意事项 要将 P01/TO00 用作通用端口，则需将定时器输出寄存器 0 (TO0) 的位 0 (TO00) 和定时器输出使能寄存器 (TOE0) 的位 0 (TOE00) 设为“0”，与它们的默认状态设置相同。

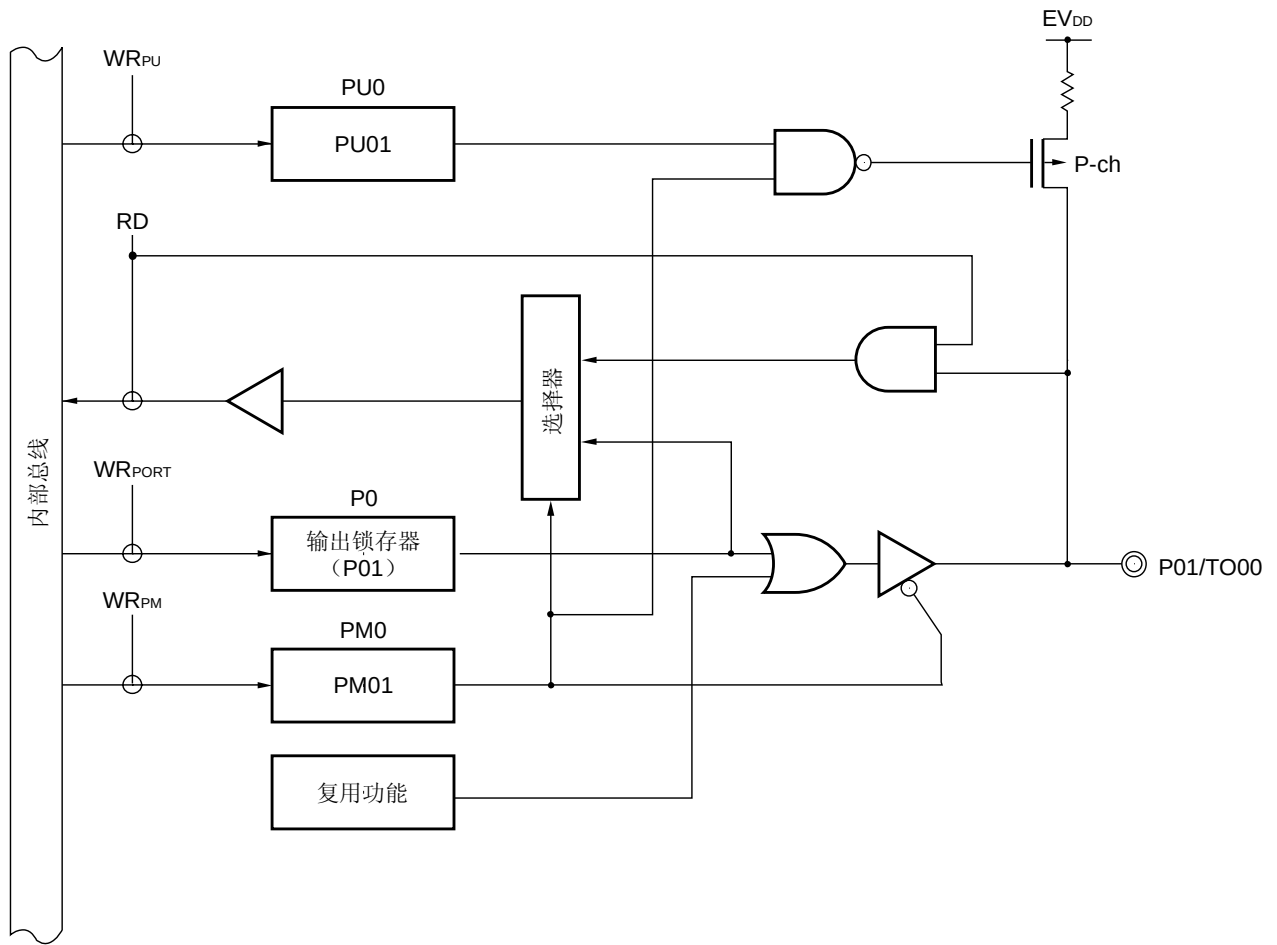
图 4-1. P00 的框图



P0: 端口寄存器 0
PU0: 上拉电阻选项寄存器 0
PM0: 端口模式寄存器 0
RD: 读取信号
WRxx: 写入信号

备注 对于不提供 EVDD 或 EVSS 引脚的产品，用 VDD 代替 EVDD，或用 VSS 代替 EVSS。

图 4-2. P01 的框图



P0: 端口寄存器 0
 PU0: 上拉电阻选项寄存器 0
 PM0: 端口模式寄存器 0
 RD: 读取信号
 WRxx: 写入信号

备注 对于不提供 EV_{DD} 或 EV_{SS} 引脚的产品，用 V_{DD} 代替 EV_{DD}，或用 V_{SS} 代替 EV_{SS}。

4.2.2 端口 1

	78K0R/KC3-L (44引脚) (μ PD78F100y: y = 0 至 3)	78K0R/KC3-L (48引脚) (μ PD78F100y: y = 1 至 3)	78K0R/KD3-L (μ PD78F100y: y = 4 至 6)	78K0R/KE3-L (μ PD78F100y: y = 7 至 9)
P10/TI02/TO02	√	√	√	√
P11/TI03/TO03	√	√	√	√
P12/TI04/TO04/ RTCDIV/RTCCL	√	√	√	√
P13/TI05/TO05	√	√	√	√
P14/TI06/TO06	— ^注	— ^注	— ^注	√
P15/TI07/TO07	— ^注	— ^注	— ^注	√
P16	—	—	—	√
P17	—	—	—	√

注 在非 78K0R/KE3-L 产品中，TI06/TO06 和 TI07/TO07 分别与 P50 和 P51 复用。

备注 √: 已配备

端口 1 是带有输出锁存器的 I/O 端口。端口 1 可用端口模式寄存器 1 (PM1) 设置为以 1 位为单元的输入或输出模式。当 P10 和 P17 引脚用作输入端口时，片上上拉电阻可用上拉电阻选项寄存器 1 (PU1) 按 1 位为单元指定。

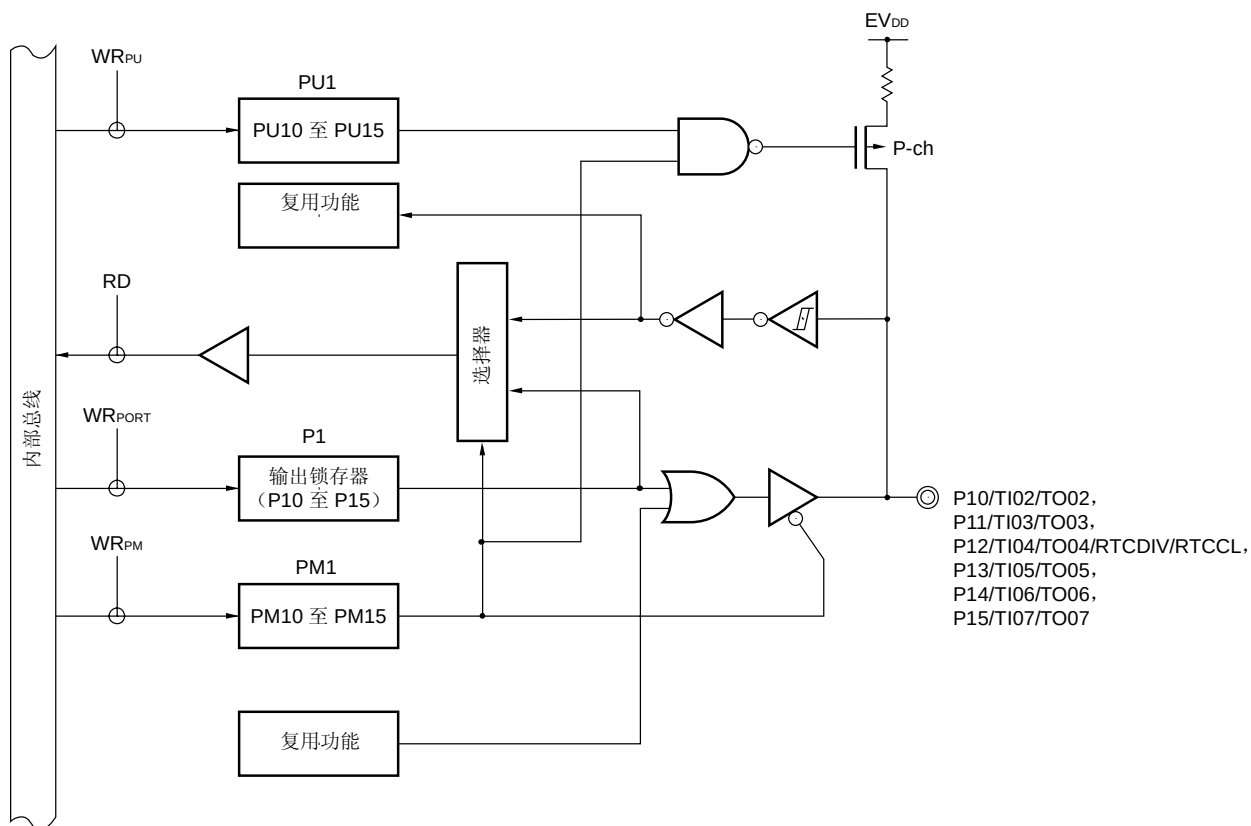
该引脚还可以用于定时器 I/O 和实时计数器时钟输出。

产生复位信号后将端口 1 设为输入模式。

图 4-3 和 4-4 显示了端口 1 的框图。

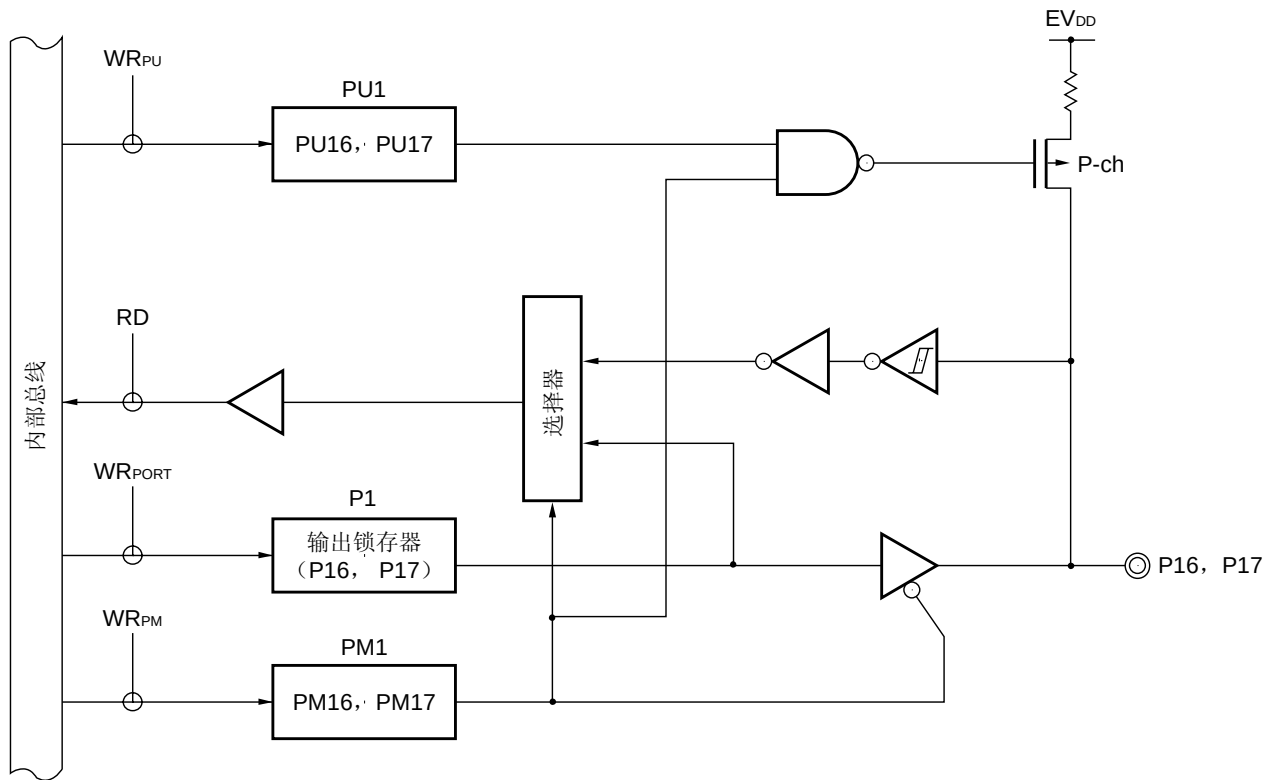
注意事项 要使 P10/TI02/TO02、P11/TI03/TO03、P12/TI04/TO04/RTCDIV/RTCCL、P13/TI05/TO05、P14/TI06/TO06 或 P15/TI07/TO07 用作通用端口，需将定时器输出寄存器 0 (TO0) 的位 2 至 7 (TO02 至 TO07) 和定时器输出使能寄存器 (TOE0) 的位 2 至 7 (TOE02 至 TOE07) 设为“0”，与它们的默认状态设置相同。

图 4-3. P10 至 P15 的框图



P1: 端口寄存器 1
PU1: 上拉电阻选项寄存器 1
PM1: 端口模式寄存器 1
RD: 读取信号
WR_{xx}: 写入信号

备注 对于不提供 EV_{DD} 或 EV_{SS} 引脚的产品，用 V_{DD} 代替 EV_{DD}，或用 V_{SS} 代替 EV_{SS}。



P1: 端口寄存器 1
PU1: 上拉电阻选项寄存器 1
PM1: 端口模式寄存器 1
RD: 读取信号
WR_{xx}: 写入信号

4.2.3 端口 2

	78K0R/KC3-L (44引脚) (μ PD78F100y: y = 0 至 3)	78K0R/KC3-L (48引脚) (μ PD78F100y: y = 1 至 3)	78K0R/KD3-L (μ PD78F100y: y = 4 至 6)	78K0R/KE3-L (μ PD78F100y: y = 7 至 9)
P20/ANI0	√	√	√	√
P21/ANI1	√	√	√	√
P22/ANI2	√	√	√	√
P23/ANI3	√	√	√	√
P24/ANI4	√	√	√	√
P25/ANI5	√	√	√	√
P26/ANI6	√	√	√	√
P27/ANI7	√	√	√	√

备注 √: 已配备

端口 2 是带有输出锁存器的 I/O 端口。 端口 2 可用端口模式寄存器 2 (PM2) 设置为以 1 位为单元的输入或输出模式。

该端口还可用于 A/D 转换器模拟输入。

要将 P20/ANI0 至 P27/ANI7 用作数字输入引脚，在数字 I/O 模式下需用 A/D 端口配置寄存器 (ADPC) 对它们进行设置；在输入模式下需用 PM2 对它们进行设置。 从低位开始使用这些引脚。

要将 P20/ANI0 至 P27/ANI7 用作数字输出引脚，在数字 I/O 模式下需用 ADPC 对它们进行设置；在输出模式下需用 PM2 对它们进行设置。

要将 P20/ANI0 至 P27/ANI7 用作模拟输入引脚，在模拟输入模式下需用 A/D 端口配置寄存器 (ADPC) 对它们进行设置；在输入模式下需用 PM2 对它们进行设置。 从高位开始使用这些引脚。

表 4-5. P20/ANI0 to P27/ANI7 引脚的功能设置

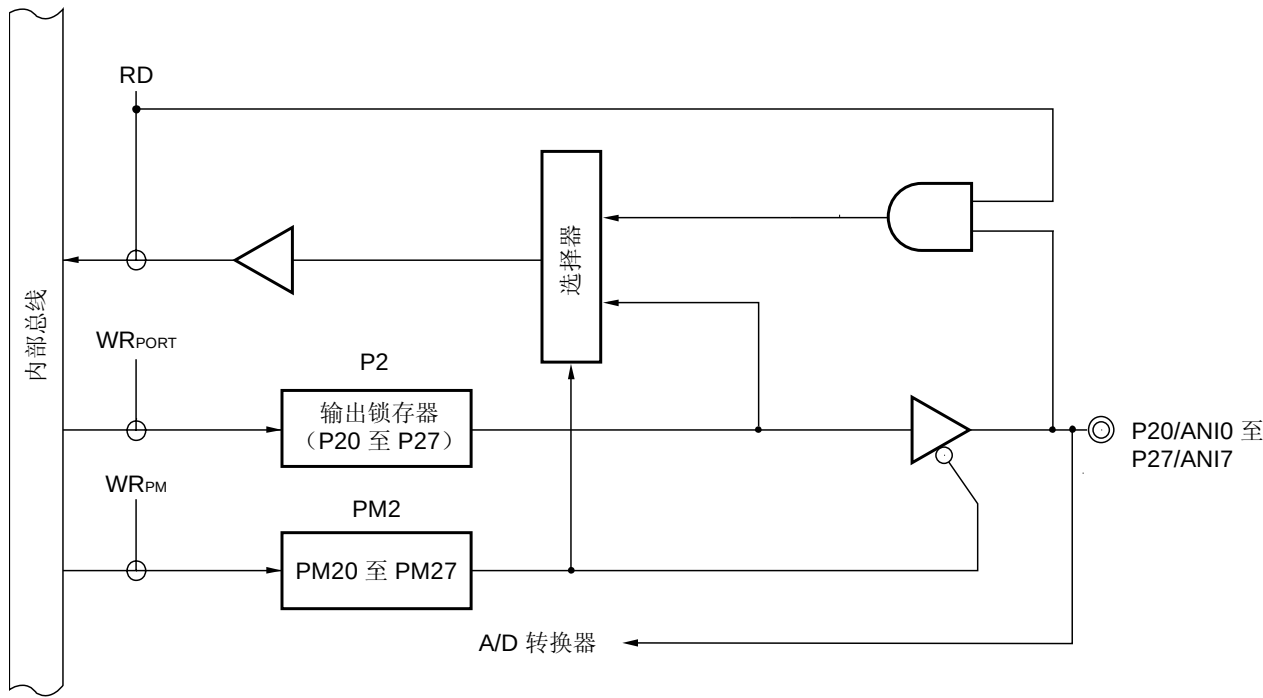
ADPC	PM2	ADS	P20/ANI0 至 P27/ANI7 引脚
数字 I/O 选择	输入模式	—	数字输入
	输出模式	—	数字输出
模拟输入选择	输入模式	选择 ANI.	模拟输入 (待转换)
		不选择 ANI.	模拟输入 (不会被转换)
	输出模式	选择 ANI.	禁止设置
		不选择 ANI.	

产生复位信号时，在数字输入模式下设置 P20/ANI0 至 P27/ANI7 的所有引脚。

图 4-5 显示了端口 2 的框图。

注意事项 将端口 2 用作数字端口时，应使引脚 AVREF 的电压与 VDD 引脚的电压相同。

图 4-5. P20 至 P27 的框图



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

4.2.4 端口 3

	78K0R/KC3-L (44引脚) (μ PD78F100y: y = 0 至 3)	78K0R/KC3-L (48引脚) (μ PD78F100y: y = 1 至 3)	78K0R/KD3-L (μ PD78F100y: y = 4 至 6)	78K0R/KE3-L (μ PD78F100y: y = 7 至 9)
P30/SO10/TxD1	√	√	√	√
P31/SI10/RxD1/ SDA10/INTP1	√	√	√	√
P32/SCK10/ SCL10/INTP2	√	√	√	√
P33	—	—	—	√

备注 √: 已配备

端口 3 是带有输出锁存器的 I/O 端口。端口 3 可用端口模式寄存器 3 (PM3) 设置为以 1 位为单元的输入或输出模式。当 P30 和 P33 引脚用作输入端口时，片上上拉电阻可用上拉电阻选项寄存器 3 (PU3) 按 1 位为单元指定。

P31 和 P32 引脚的输入可通过标准输入缓存器或 TTL 输入缓存器按 1 位为单位进行指定，指定时使用端口输入模式寄存器 3 (PIM3)。

P30 至 P32 引脚的输出可按 1 位为单位指定为 N-通道漏极开路输出 (V_{DD} 耐压)，指定时使用端口输出模式寄存器 3 (POM3)。

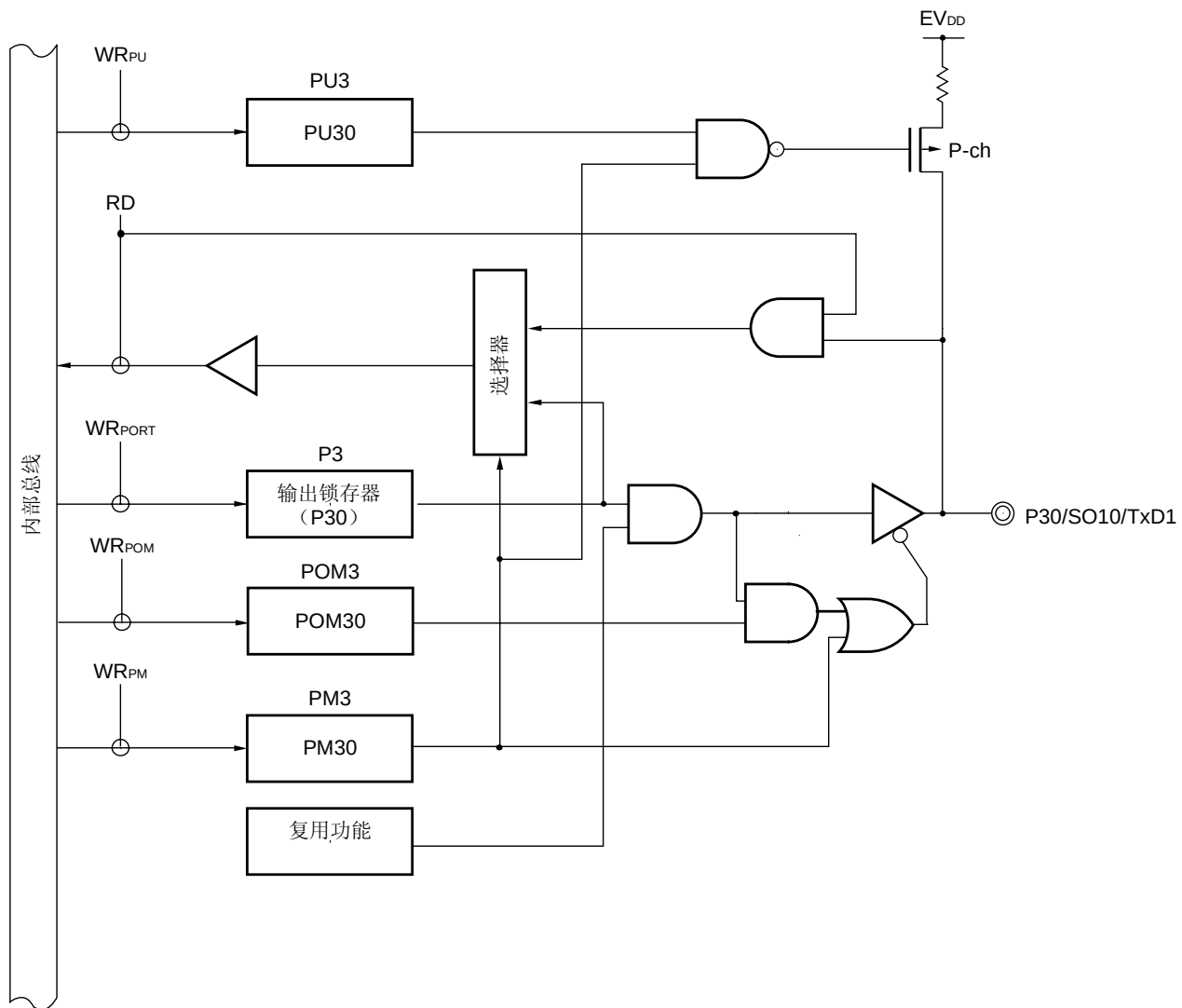
该端口还可用作串行接口数据 I/O、时钟 I/O 以及外部中断请求的输入。

产生复位信号后将端口 3 设为输入模式。

图 4-6 至 4-8 显示了端口 3 的框图。

注意事项 要将 P30/SO10/TxD1、P31/SI10/RxD1/SDA10/INTP1、P32/SCK10/SCL10/INTP2 用作通用端口，则需注意串行阵列部件的设置。有关细节，请参见表 12-7 寄存器设置和引脚之间的关系（通道 2: CSI10、UART1 发送、IIC10）和表 12-8 寄存器设置和引脚之间的关系（通道 3: UART1 接收）。

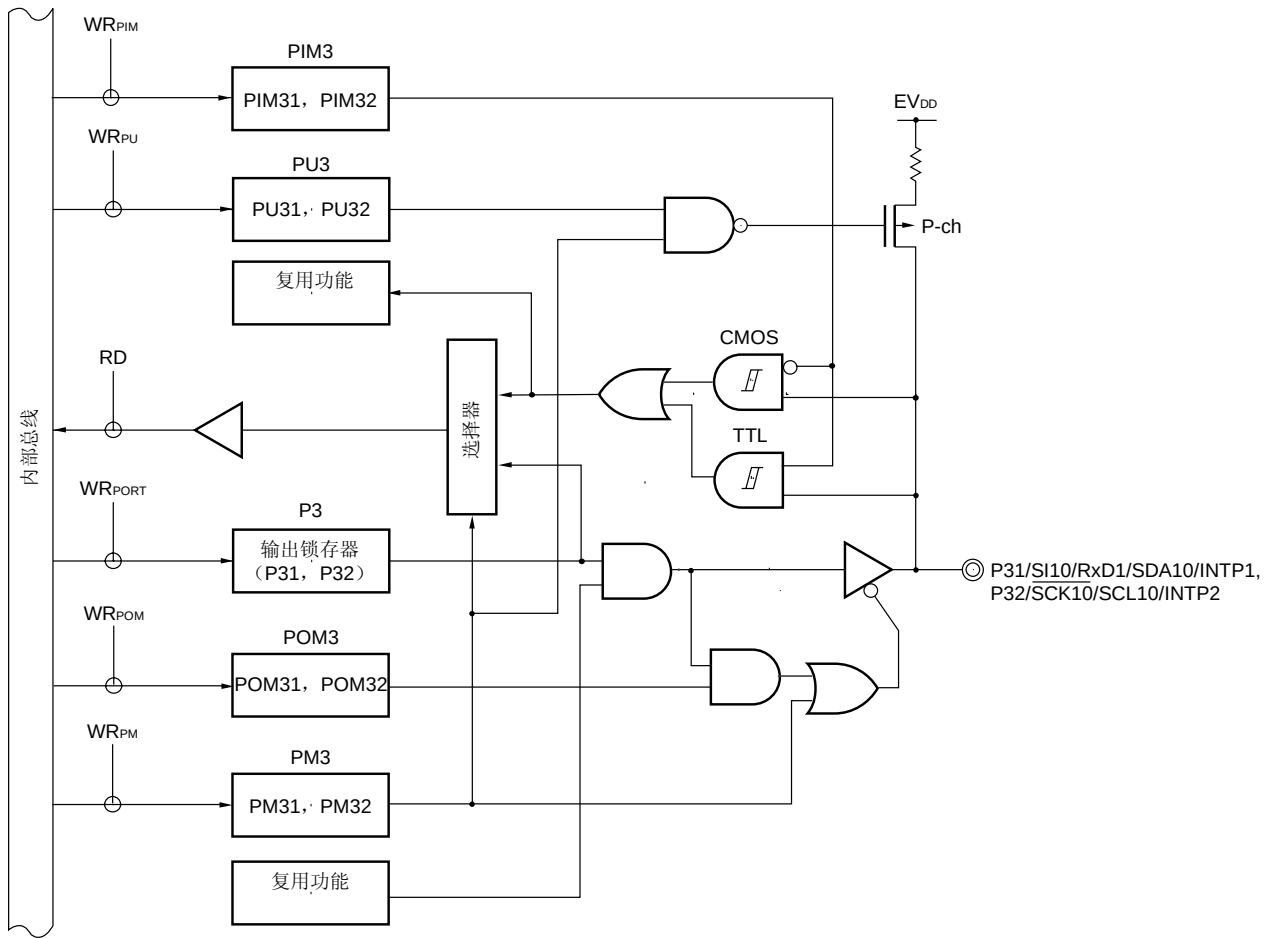
图 4-6. P30 的框图



P3: 端口寄存器 3
 PU3: 上拉电阻选项寄存器 3
 POM3: 端口输出模式寄存器 3
 PM3: 端口模式寄存器 3
 RD: 读取信号
 WR_{xx}: 写入信号

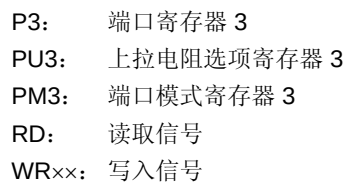
备注 对于不提供 EV_{DD} 或 EV_{SS} 引脚的产品，用 V_{DD} 代替 EV_{DD}，或用 V_{SS} 代替 EV_{SS}。

图 4-7. P31 和 P32 的框图



P3: 端口寄存器 3
 PU3: 上拉电阻选项寄存器 3
 PIM3: 端口输入模式寄存器 3
 POM3: 端口输出模式寄存器 3
 PM3: 端口模式寄存器 3
 RD: 读取信号
 WR_{xx} : 写入信号

备注 对于不提供 EV_{DD} 或 EV_{SS} 引脚的产品，用 V_{DD} 代替 EV_{DD} ，或用 V_{SS} 代替 EV_{SS} 。



4.2.5 端口 4

	78K0R/KC3-L (44引脚) (μPD78F100y: y = 0 至 3)	78K0R/KC3-L (48引脚) (μPD78F100y: y = 1 至 3)	78K0R/KD3-L (μPD78F100y: y = 4 至 6)	78K0R/KE3-L (μPD78F100y: y = 7 至 9)
P40/TOOL0	√	√	√	√
P41/ TOOL1	√	√	√	√
P42	—	—	—	√
P43	—	—	—	√

备注 √: 已配备

端口 4 是带有输出锁存器的 I/O 端口。 端口 4 可用端口模式寄存器 4（PM4）设置为以 1 位为单元的输入或输出模式。 当 P40 和 P43 引脚用作输入端口时，片上上拉电阻可用上拉电阻选项寄存器 4（PU4）按 1 位为单元指定。

该端口还可用作 flash 存储器编程器/调试器数据 I/O 和时钟输出。

产生复位信号后将端口 4 设为输入模式。

图 4-9 和 4-10 显示了端口 4 的框图。

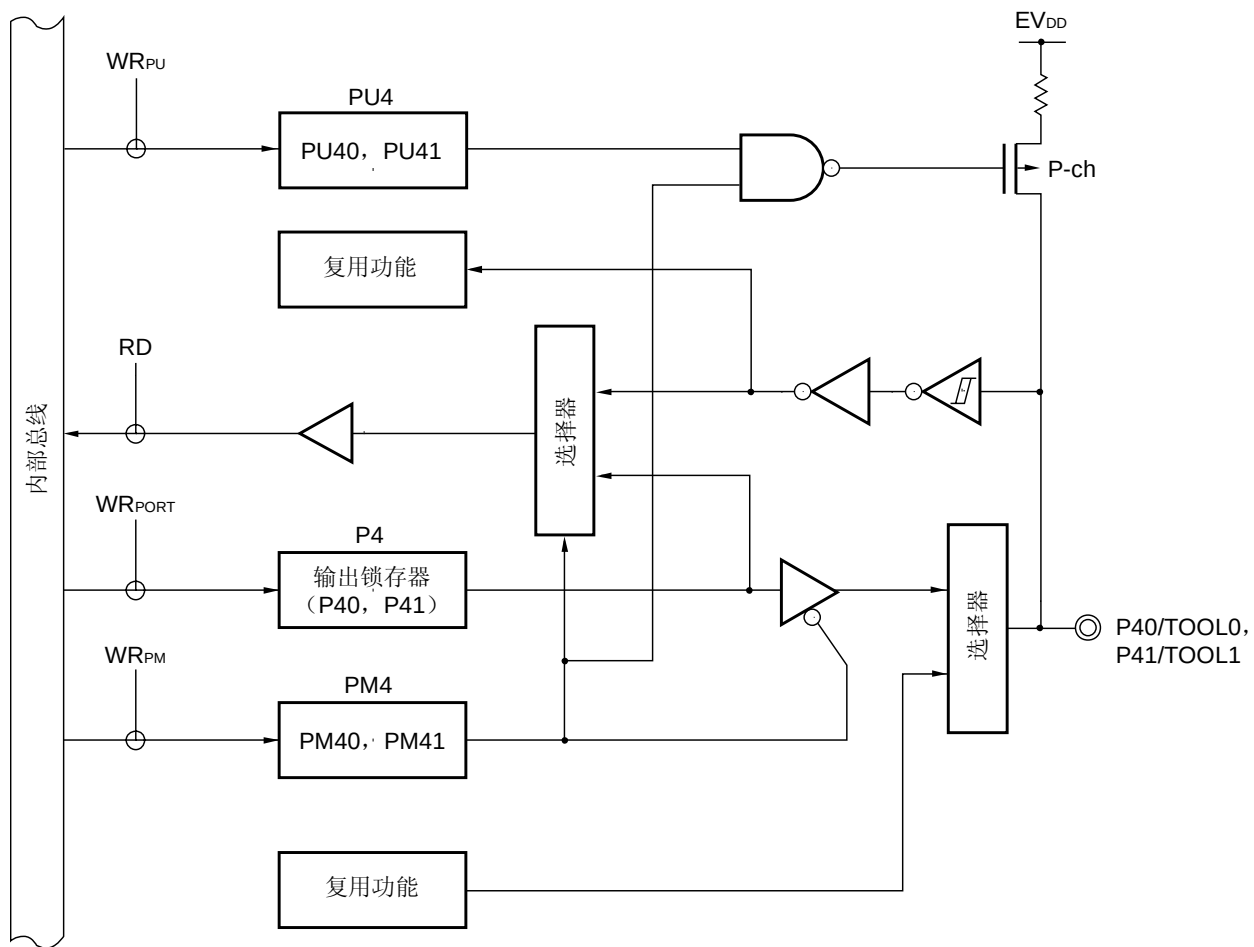
备注 当 P40 和 P41 引脚与工具相连时，不能连接至上拉电阻。

注意事项 当 P40 与工具相连时，不能用作端口引脚。

当使用片上调试功能时，可通过调试器的模式设置使用 P41 引脚，如下所示。

- 1-线模式： 可用作端口（P41）。
- 2-线模式： 用作 TOOL1 引脚且不能用作端口（P41）。

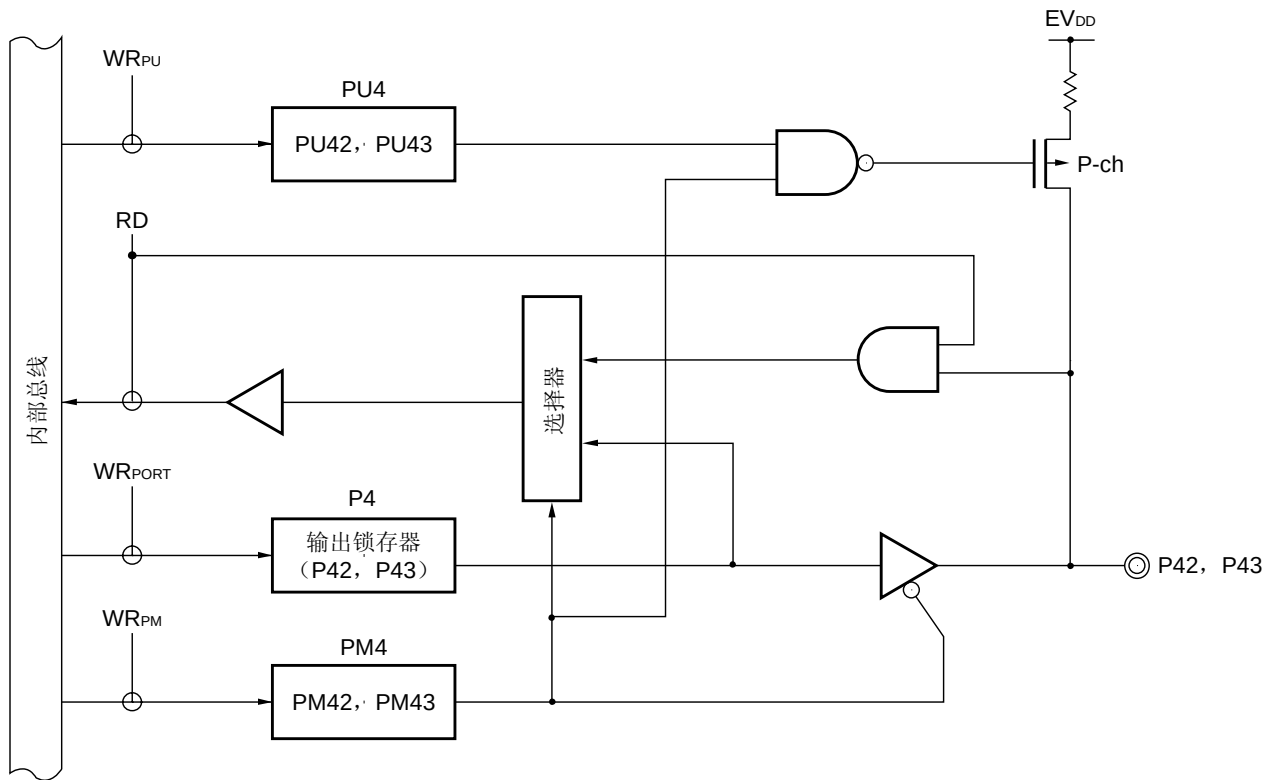
图 4-9. P40 和 P41 的框图



P4: 端口寄存器 4
 PU4: 上拉电阻选项寄存器 4
 PM4: 端口模式寄存器 4
 RD: 读取信号
 WR_{xx} : 写入信号

备注 对于不提供 EV_{DD} 或 EV_{SS} 引脚的产品，用 V_{DD} 代替 EV_{DD} ，或用 V_{SS} 代替 EV_{SS} 。

图 4-10. P42 和 P43 的框图



P4: 端口寄存器 4
PU4: 上拉电阻选项寄存器 4
PM4: 端口模式寄存器 4
RD: 读取信号
WR_{xx}: 写入信号

4.2.6 端口 5

	78K0R/KC3-L (44引脚) (μ PD78F100y: y = 0 至 3)	78K0R/KC3-L (48引脚) (μ PD78F100y: y = 1 至 3)	78K0R/KD3-L (μ PD78F100y: y = 4 至 6)	78K0R/KE3-L (μ PD78F100y: y = 7 至 9)
P50/TI06/TO06	√	√	√	P50 [※]
P51/TI07/TO07	√	√	√	P51 [※]
P52/RTC1HZ/ SLTI/SLTO	√	√	√	√
P53	—	—	—	√

注 TI06/TO06 和 TI07/TO07 仅能在 78K0R/KC3-L 和 78K0R/KD3-L 中共享。78K0R/KE3-L 不具有共享功能。

备注 √: 已配备

端口 5 是带有输出锁存器的 I/O 端口。端口 5 可用端口模式寄存器 5 (PM5) 设置为以 1 位为单元的输入或输出模式。当 P50 和 P53 引脚用作输入端口时，片上上拉电阻可用上拉电阻选项寄存器 5 (PU5) 按 1 位为单元指定。

该端口还可用作实时计数器校正时钟输出和定时器 I/O。

产生复位信号后将端口 5 设为输入模式。

图 4-11 至 4-13 显示了端口 5 的框图。

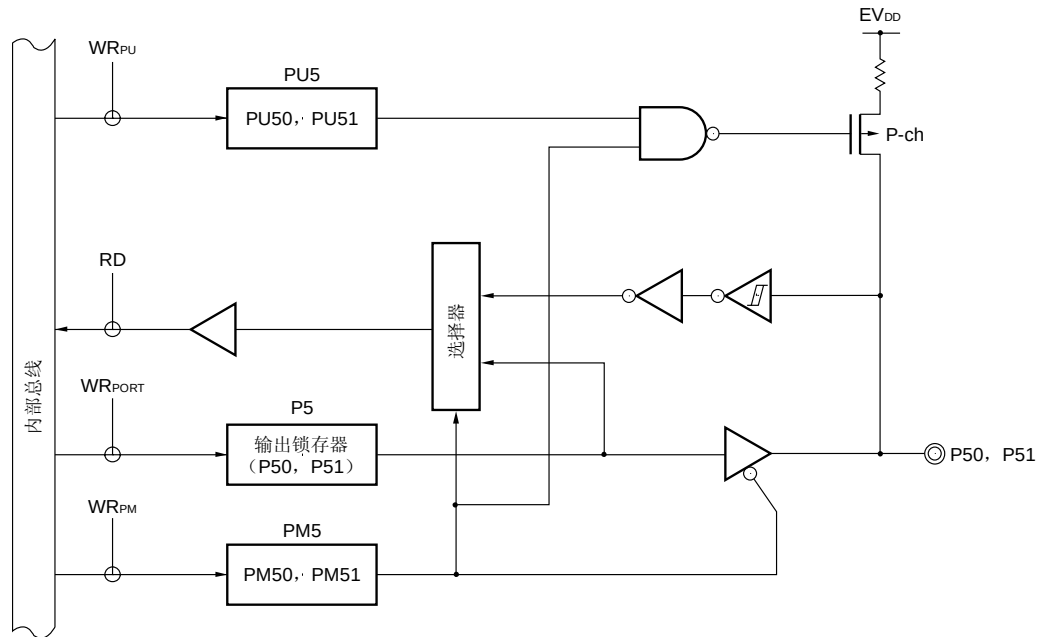
注意事项 1. 要将 P50/TI06/TO06 和 P51/TI07/TO07 用作通用端口，则需将定时器输出寄存器 0 (TO0) 的位 6 和 7 (TO06 和 TO07) 和定时器输出使能寄存器 (TOE0) 的位 6 和 7 (TOE06 和 TOE07) 设为“0”，与它们的默认状态设置相同。”

2. 要将 P52/RTC1HZ/SLTI/SLTO 用作通用端口，需在输入切换控制寄存器 (ISC) 设置中检查选择了哪个寄存器的哪个通道 n。另外，将定时器输出寄存器 (TO0) 的位 n (TO0n) 和定时器输出使能寄存器 (TOE0) 的位 n (TOE0n) 设为“0”，与各个初始状态的设置相同。

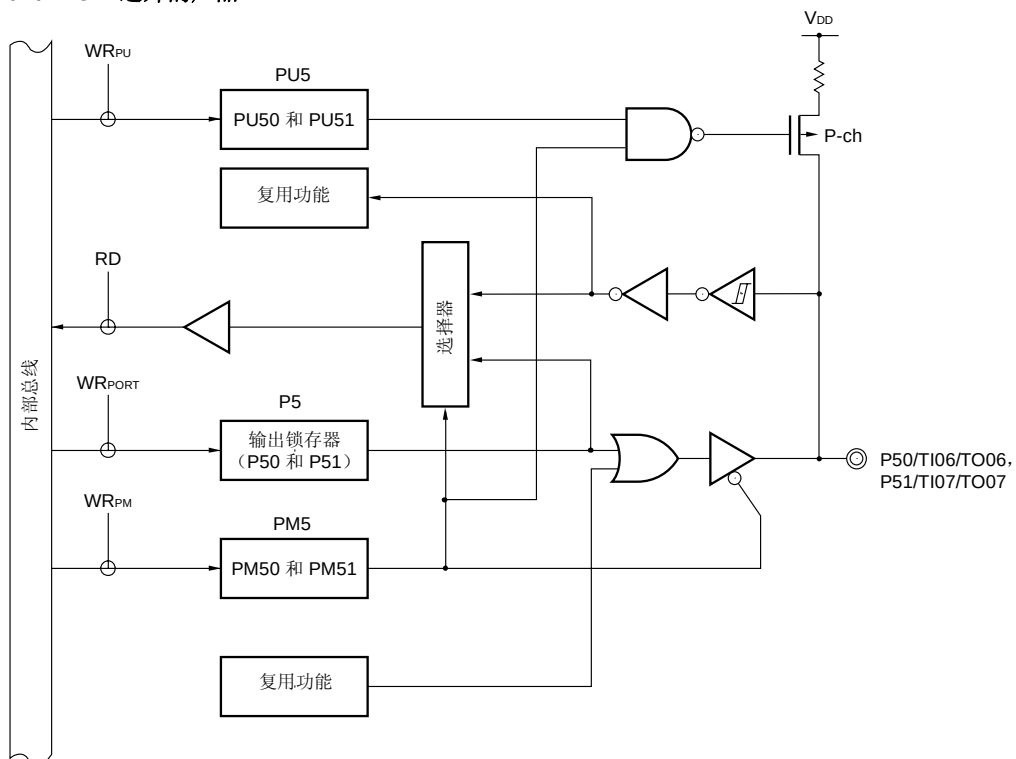
备注 n = 0, 1

图 4-11. P50 和 P51 的框图

(1) 78K0R/KE3-L

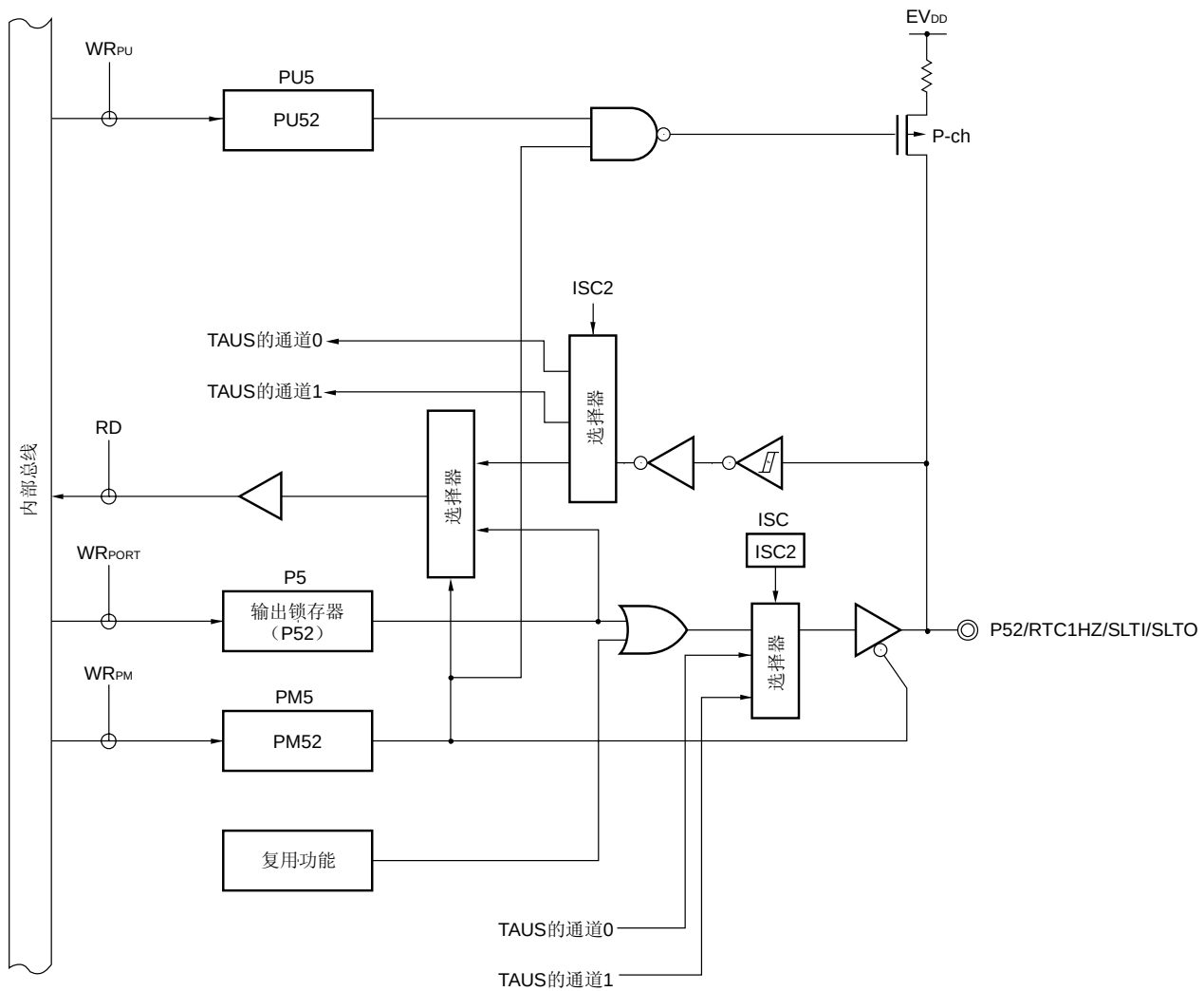


(2) 除 78K0R/KE3-L 之外的产品



P5: 端口寄存器 5
 PU5: 上拉电阻选项寄存器 5
 PM5: 端口模式寄存器 5
 RD: 读取信号
 WRxx: 写入信号

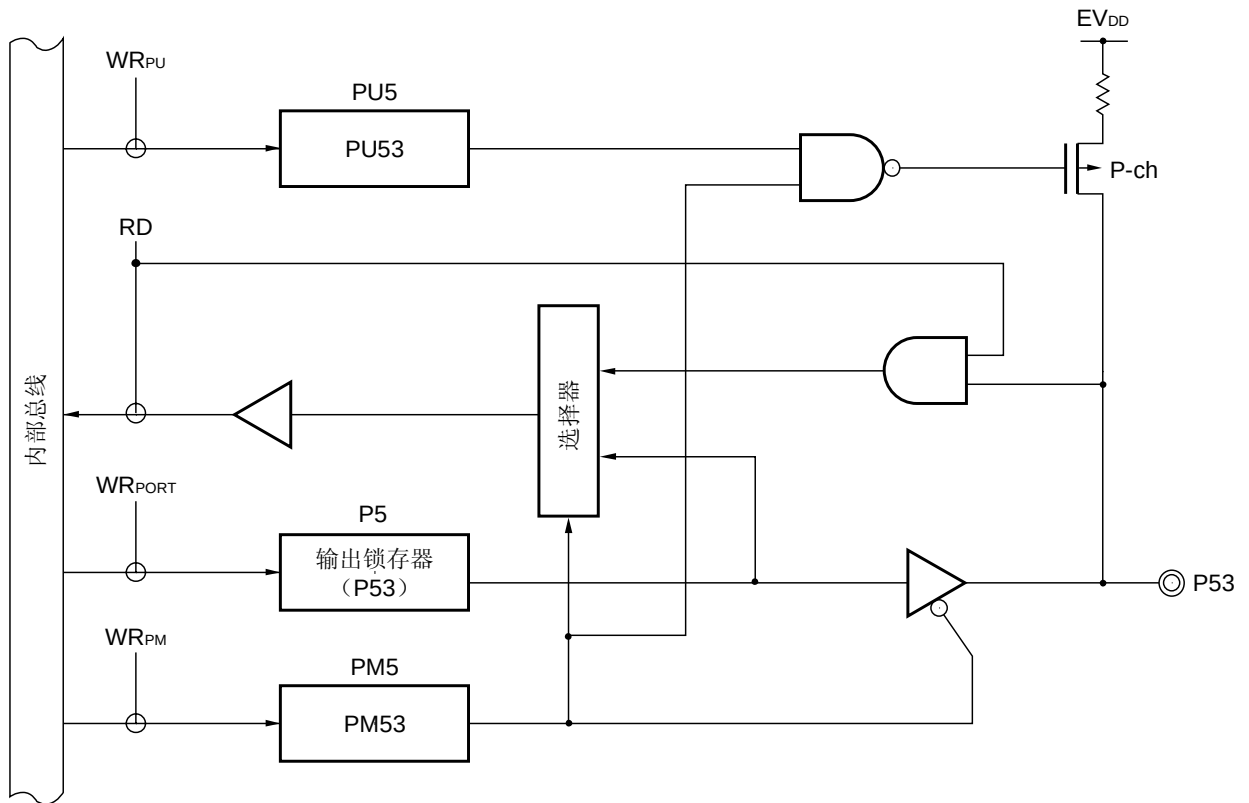
图 4-12. P52 的框图



P5: 端口寄存器 5
PU5: 上拉电阻选项寄存器 5
PM5: 端口模式寄存器 5
RD: 读取信号
 WR_{xx} : 写入信号
ISC: 输入转换控制寄存器

备注 对于不提供 EV_{DD} 或 EV_{SS} 引脚的产品，用 V_{DD} 代替 EV_{DD} ，或用 V_{SS} 代替 EV_{SS} 。

图 4-13. P53 的框图



P5: 端口寄存器 5
 PU5: 上拉电阻选项寄存器 5
 PM5: 端口模式寄存器 5
 RD: 读取信号
 WR_{xx}: 写入信号

4.2.7 端口 6

	78K0R/KC3-L (44引脚) (μ PD78F100y: y = 0 至 3)	78K0R/KC3-L (48引脚) (μ PD78F100y: y = 1 至 3)	78K0R/KD3-L (μ PD78F100y: y = 4 至 6)	78K0R/KE3-L (μ PD78F100y: y = 7 至 9)
P60/SCL0	–	√	√	√
P61/SDA0	–	√	√	√

备注 √: 已配备

端口 6 是带有输出锁存器的 I/O 端口。 端口 6 可用端口模式寄存器 6 (PM6) 设置为以 1 位为单元的输入或输出模式。

P60 和 P61 引脚的输出为 N-通道漏极开路输出(6 V 耐压)。

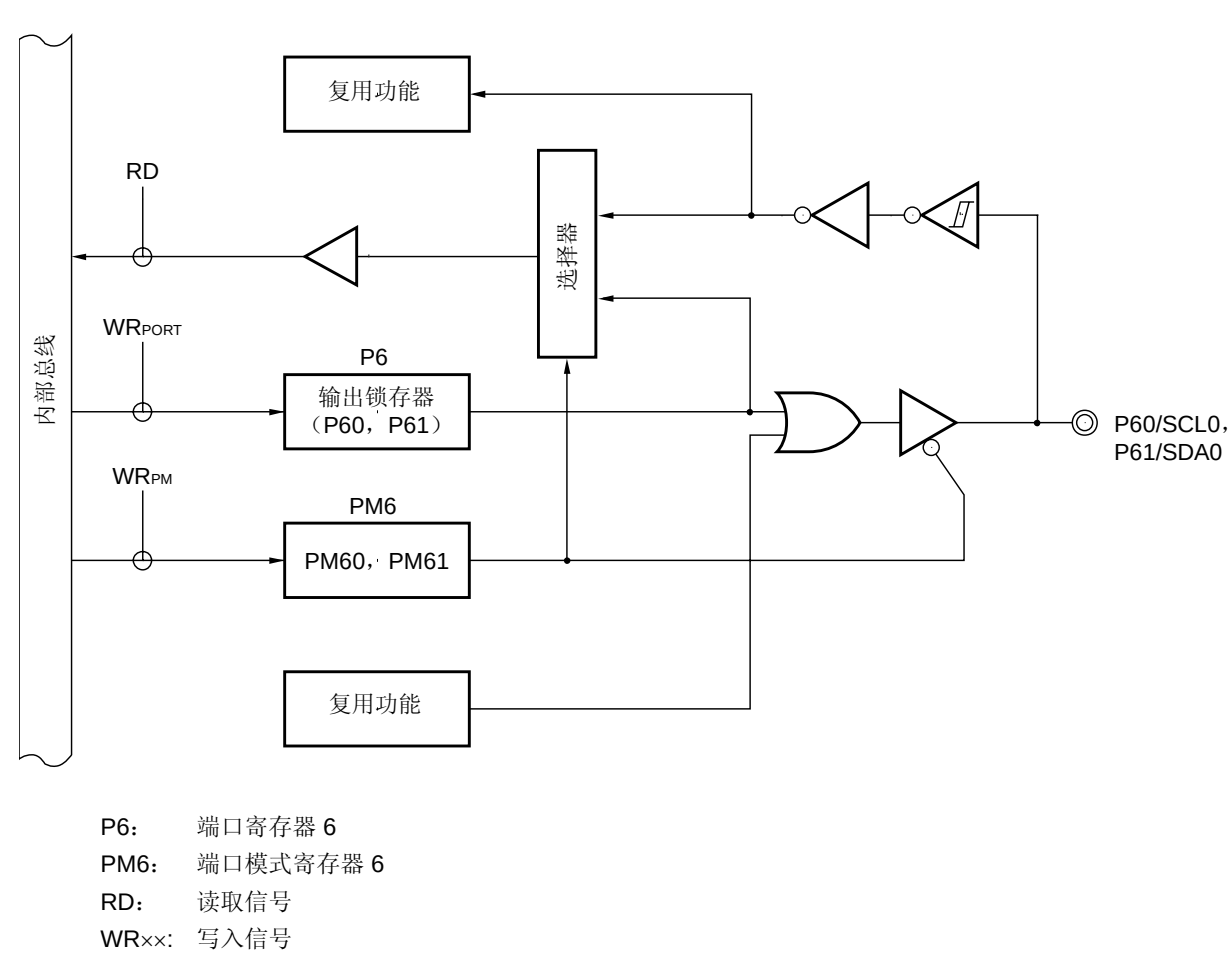
该端口还可用于串行接口数据 I/O 和时钟 I/O。

产生复位信号后将端口 6 设为输入模式。

图 4-14 显示了端口 6 的框图。

注意事项 使用 P60/SCL0 或 P61/SDA0 作为通用端口时，应停止串行接口 IICA 的操作。

图 4-14. P60 和 P61 的框图



4.2.8 端口 7

	78K0R/KC3-L (44引脚) (μ PD78F100y: y = 0 至 3)	78K0R/KC3-L (48引脚) (μ PD78F100y: y = 1 至 3)	78K0R/KD3-L (μ PD78F100y: y = 4 至 6)	78K0R/KE3-L (μ PD78F100y: y = 7 至 9)
P70/KR0/SO01/ INTP4	√	√	√	√
P71/KR1/SI01/ INTP5	√	√	√	√
P72/KR2/ SCK01/INTP6	√	√	√	√
P73/KR3/SO00/ TxD0	√	√	√	√
P74/KR4/SI00/ RxD0	√	√	√	√
P75/KR5/SCK00	√	√	√	√
P76/KR6	—	—	√	√
P77/KR7	—	—	√	√

备注 √: 已配备

端口 7 是带有输出锁存器的 I/O 端口。端口 7 可用端口模式寄存器 7 (PM7) 设置为以 1 位为单元的输入或输出模式。当 P70 和 P77 引脚用作输入端口时，片上上拉电阻可用上拉电阻选项寄存器 7 (PU7) 按 1 位为单元指定。

P71、P72、P74 和 P75 引脚的输入可通过标准输入缓存器或 TTL 输入缓存器按 1 位为单位进行指定，指定时使用端口输入模式寄存器 7 (PIM7)。

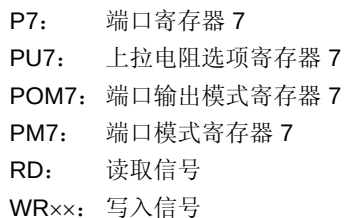
P70、P72、P73 和 P75 引脚的输出可按 1 位为单位指定为 N-通道漏极开路输出 (V_{DD} 耐压)，指定时使用端口输出模式寄存器 7 (POM7)。

该端口还可用作按键中断输入、串行接口数据 I/O、时钟 I/O 以及外部中断请求的输入。

产生复位信号后将端口 7 设为输入模式。

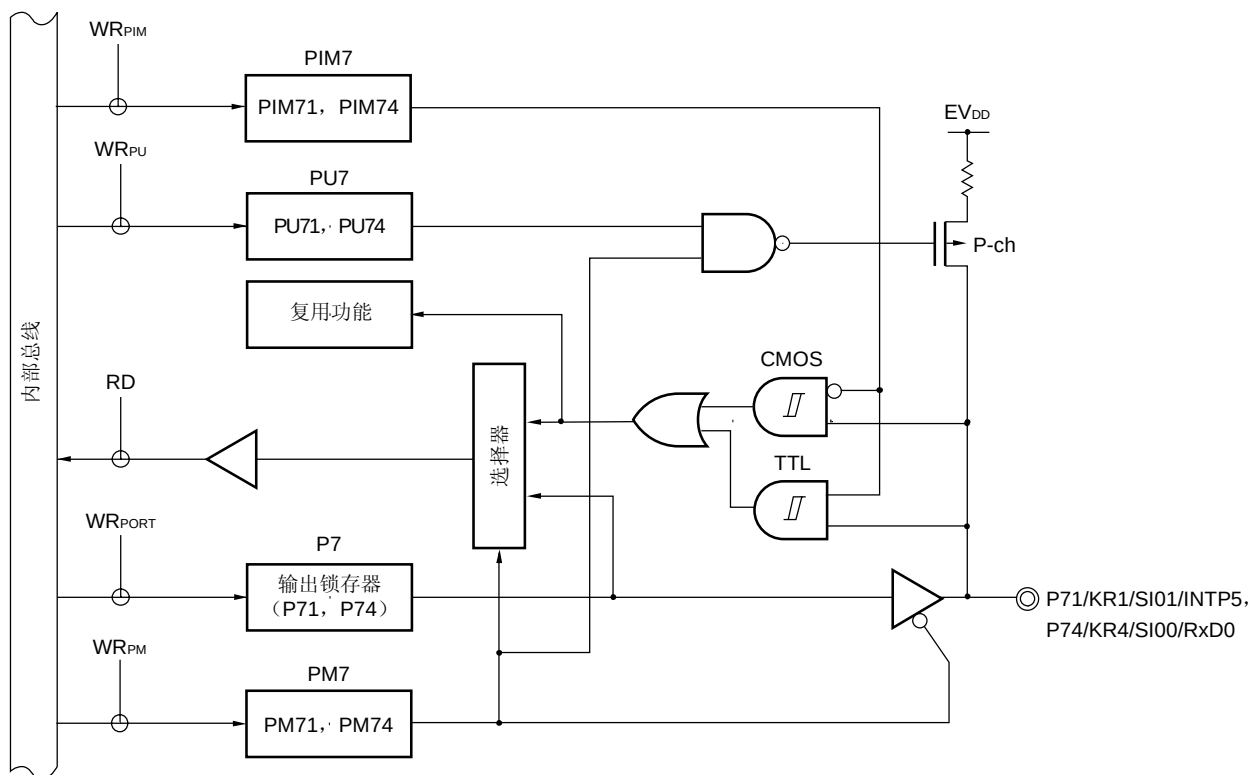
图 4-15 至 4-18 显示了端口 7 的框图。

注意事项 要将 P70/KR0/SO01/INTP4、P71/KR1/SI01/INTP5、P72/KR2/SCK01/INTP6、P73/KR3/SO00/TxD0、P74/KR4/SI00/RxD0 和 P75/KR5/SCK00 用作通用端口，应注意串行阵列部件的设置。有关细节，请参见表 12-5 寄存器设置和引脚之间的关系（通道 0: CSI00、UART0 发送）和表 12-6 寄存器设置和引脚之间的关系（通道 1: CSI01, UART0 接收）。



134 用户手册初稿 U19291CA2V0UD

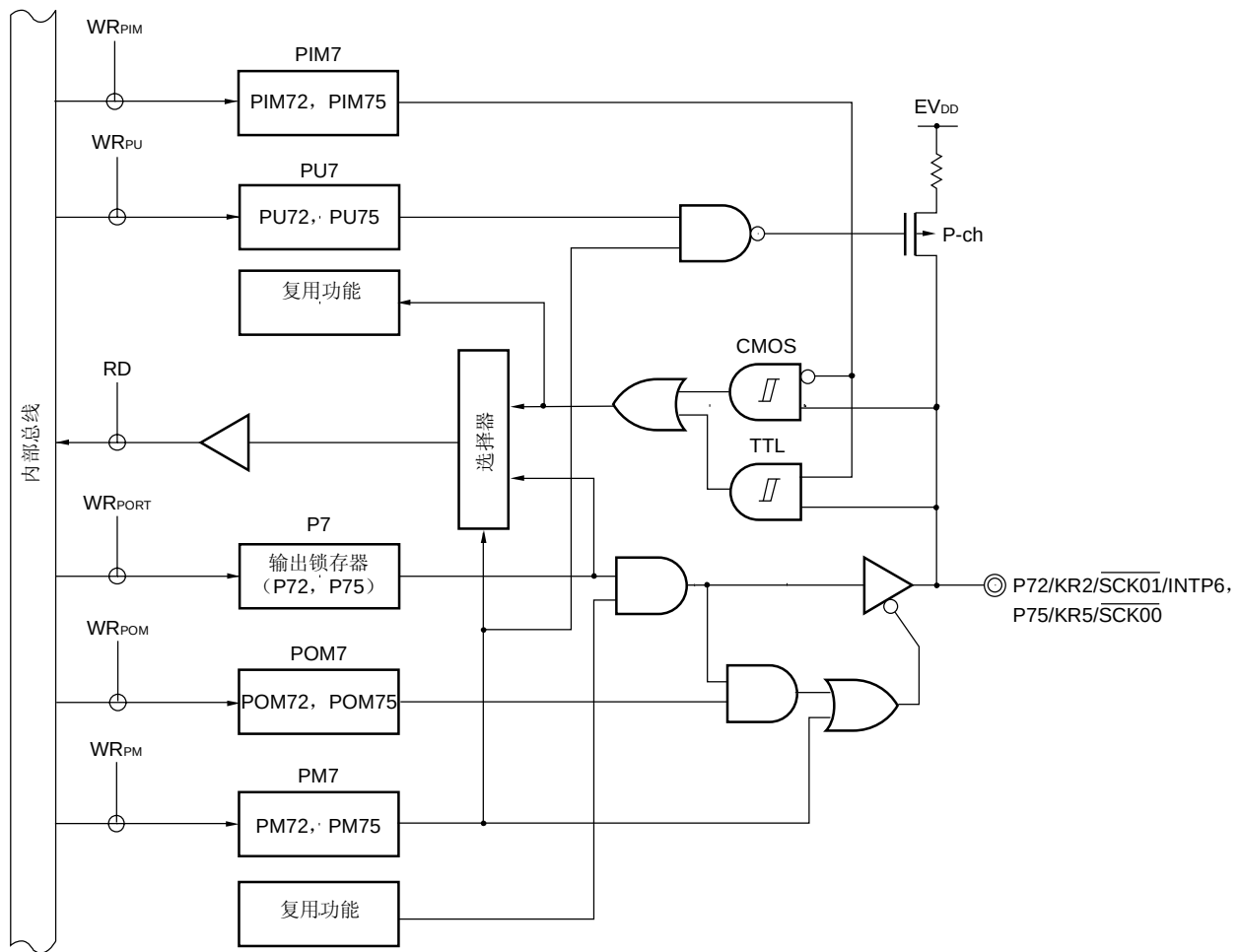
图 4-16. P71 和 P74 的框图



P7: 端口寄存器 7
 PU7: 上拉电阻选项寄存器 7
 PM7: 端口模式寄存器 7
 PIM7: 端口输入模式寄存器 7
 RD: 读取信号
 WR_{xx}: 写入信号

备注 对于不提供 EV_{DD} 或 EV_{SS} 引脚的产品，用 V_{DD} 代替 EV_{DD}，或用 V_{SS} 代替 EV_{SS}。

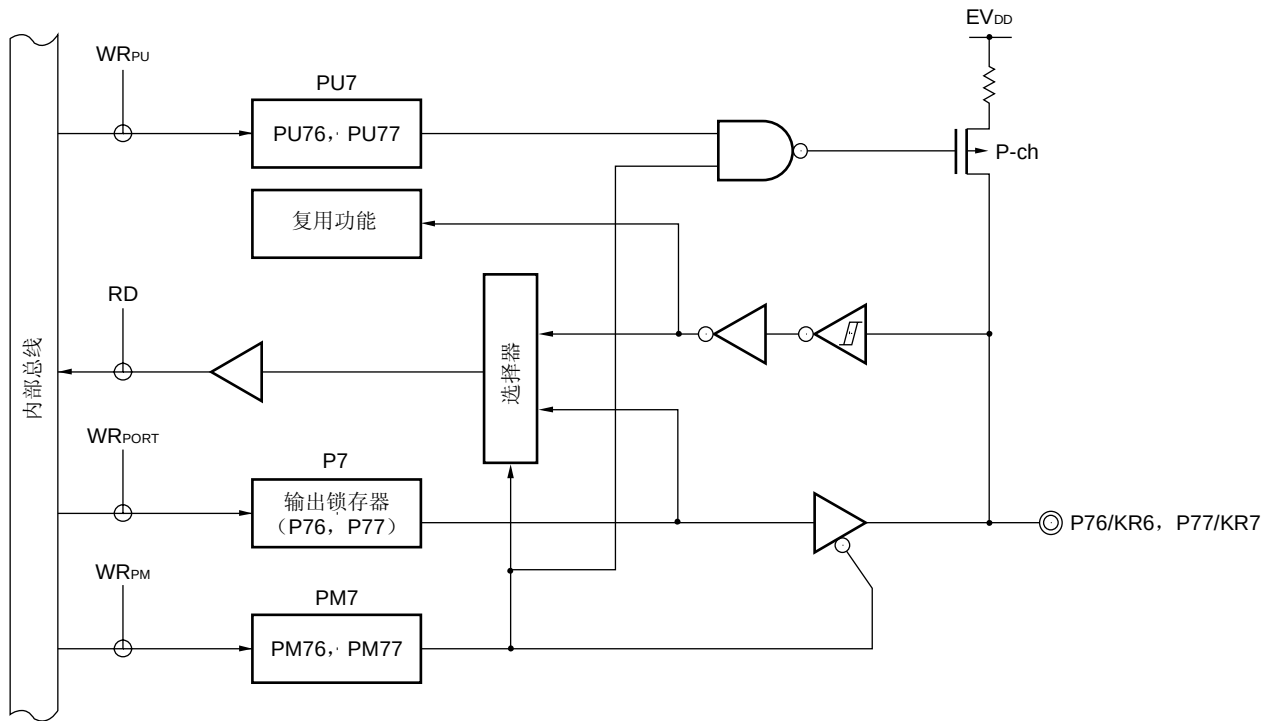
图 4-17. P72 和 P75 的框图



P7: 端口寄存器 7
 PU7: 上拉电阻选项寄存器 7
 PM7: 端口模式寄存器 7
 PIM7: 端口输入模式寄存器 7
 POM7: 端口输出模式寄存器 7
 RD: 读取信号
 WR_{xx} : 写入信号

备注 对于不提供 EV_{DD} 或 EV_{SS} 引脚的产品，用 V_{DD} 代替 EV_{DD} ，或用 V_{SS} 代替 EV_{SS} 。

图 4-18. P76 和 P77 的框图



P7: 端口寄存器 7
 PU7: 上拉电阻选项寄存器 7
 PM7: 端口模式寄存器 7
 RD: 读取信号
 WR_{xx}: 写入信号

备注 对于不提供 EV_{DD} 或 EV_{SS} 引脚的产品，用 V_{DD} 代替 EV_{DD}，或用 V_{SS} 代替 EV_{SS}。

4.2.9 端口 8

	78K0R/KC3-L (44引脚) (μ PD78F100y: y = 0 至 3)	78K0R/KC3-L (48引脚) (μ PD78F100y: y = 1 至 3)	78K0R/KD3-L (μ PD78F100y: y = 4 至 6)	78K0R/KE3-L (μ PD78F100y: y = 7 至 9)
P80/CMP0P/ INTP3/PGAI	√	√	√	√
P81/CMP0M	√	√	√	√
P82/CMP1P/ INTP7	√	√	√	√
P83/CMP1M	√	√	√	√

备注 √: 已配备

端口 8 是带有输出锁存器的 I/O 端口。 端口 8 可用端口模式寄存器 8 (PM8) 设置为以 1 位为单元的输入或输出模式。

P80 至 P83 引脚的输入必须用端口输入模式寄存器 8 (PIM8) 按 1 位为单位使能或禁止。

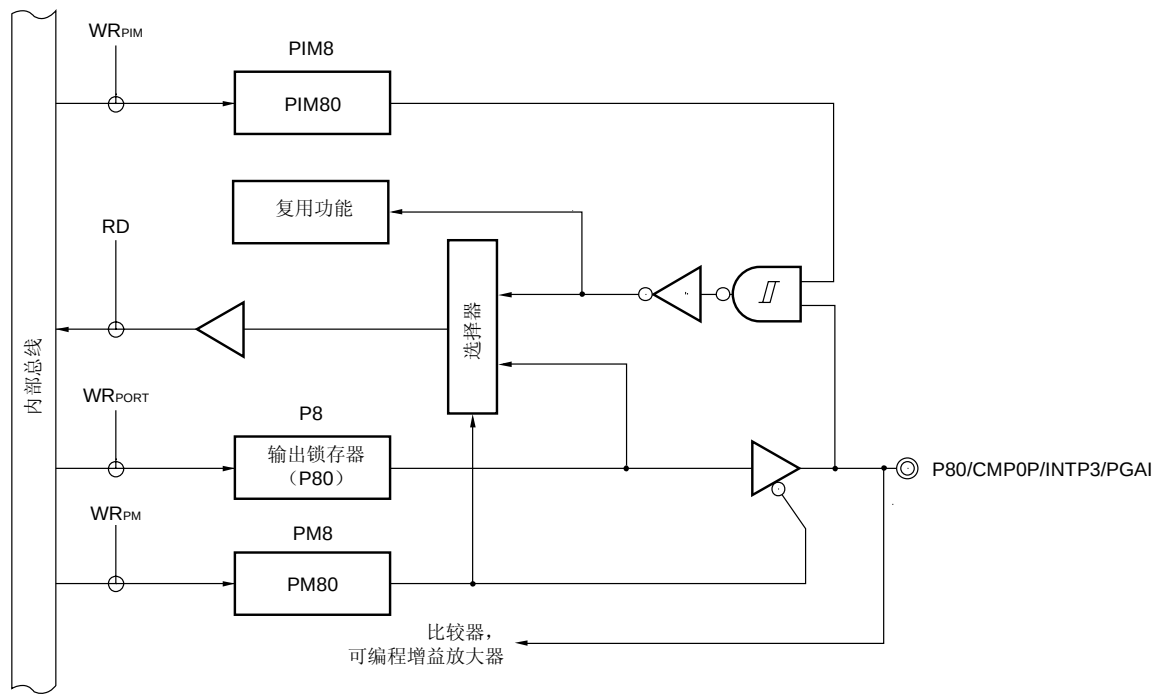
该端口还可以用作比较器 0 和 1 (+) 侧的输入电压、比较器 0 和 1 (-) 侧的输入电压、外部中断请求输入和可编程增益放大器输入。

产生复位信号后将端口 8 设为输入模式。

图 4-19 至 4-21 显示了端口 8 的框图。

<R>

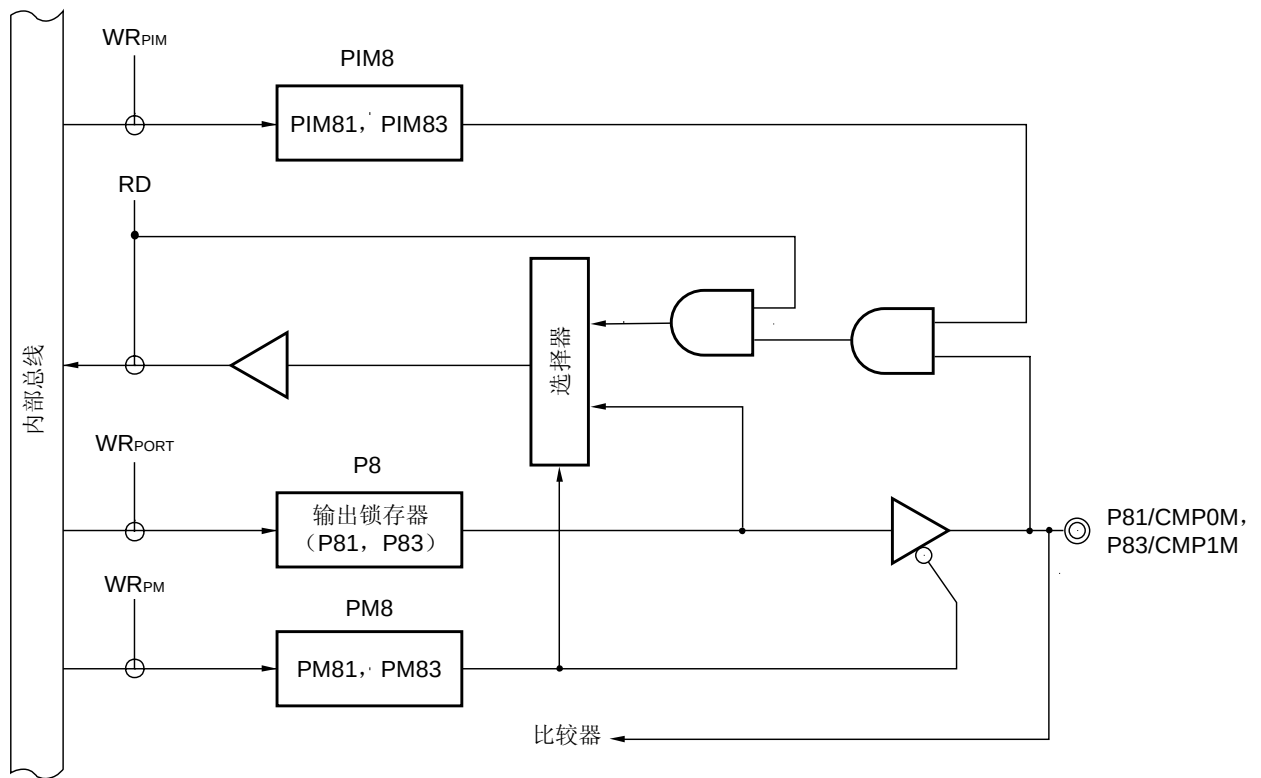
图 4-19. P80 的框图



- P8: 端口寄存器 8
- PM8: 端口模式寄存器 8
- PIM8: 端口输入模式寄存器 8
- RD: 读取信号
- WR_{xx} : 写入信号

<R>

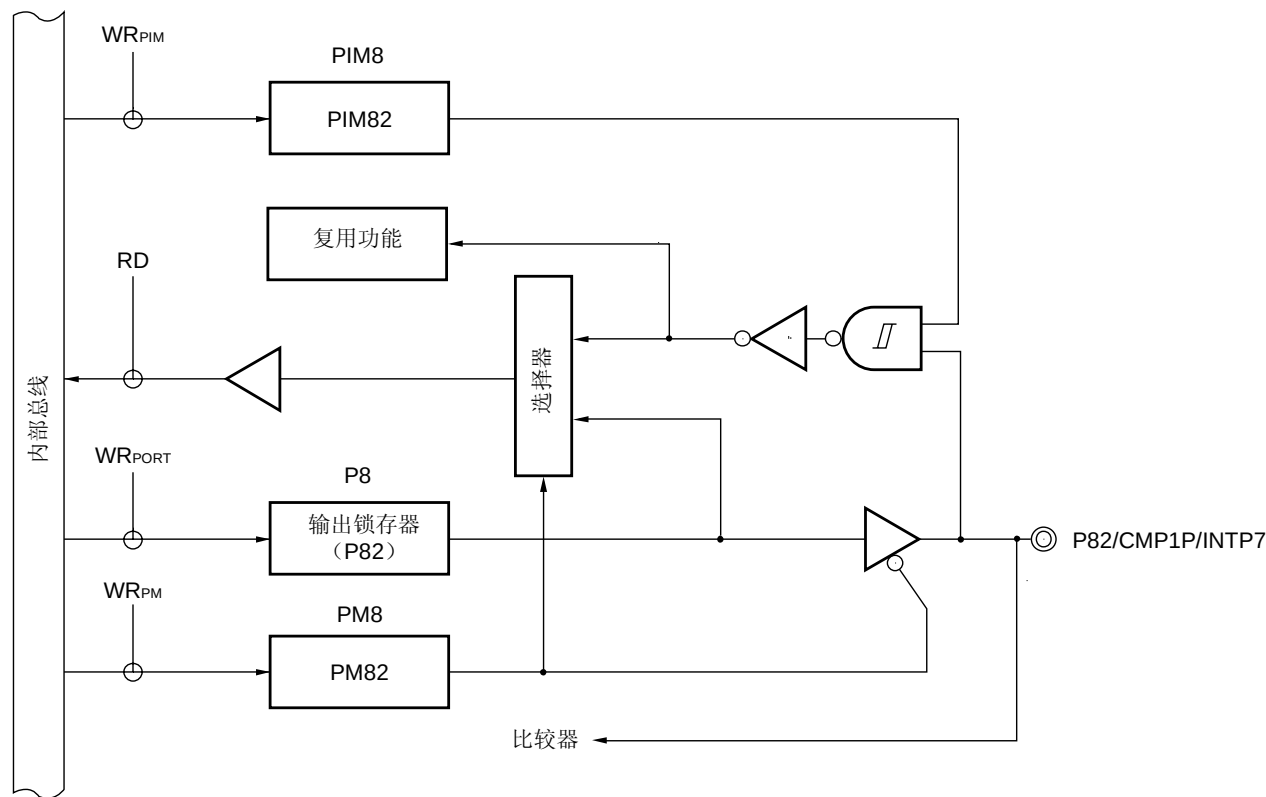
图 4-20. P81 和 P83 的框图



P8: 端口寄存器 8
 PM8: 端口模式寄存器 8
 PIM8: 端口输入模式寄存器 8
 RD: 读取信号
 WR_{xx}: 写入信号

<R>

图 4-21. P82 的框图



- P8: 端口寄存器 8
PM8: 端口模式寄存器 8
PIM8: 端口输入模式寄存器 8
RD: 读取信号
WR_{xx}: 写入信号

4.2.10 端口 12

	78K0R/KC3-L (44引脚) (μ PD78F100y: y = 0 至 3)	78K0R/KC3-L (48引脚) (μ PD78F100y: y = 1 至 3)	78K0R/KD3-L (μ PD78F100y: y = 4 至 6)	78K0R/KE3-L (μ PD78F100y: y = 7 至 9)
P120/INTP0/ EXLVI	√	√	√	√
P121/X1	√	√	√	√
P122/X2/ EXCLK	√	√	√	√
P123/XT1	√	√	√	√
P124/XT2	√	√	√	√

备注 √: 已配备

P120 是带有输出锁存器的 I/O 端口。端口 12 可用端口模式寄存器 12 (PM12) 设置为以 1 位为单元的输入或输出模式。当用作输入端口时, 片上上拉电阻的使用可通过上拉电阻选项寄存器 12 (PU012) 指定。

P121 至 P124 位输入端口。

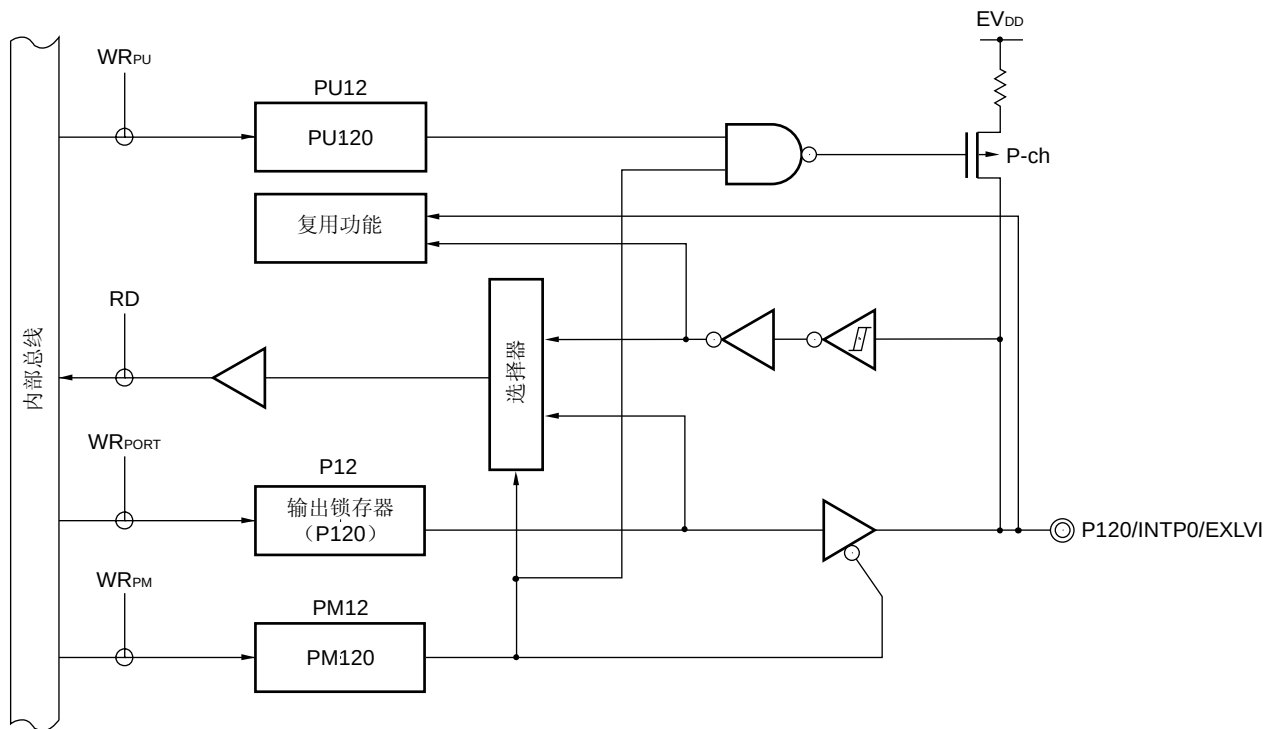
该端口还可用作外部中断输入、外部低电压检测的电压输入、主系统时钟的连接谐振器、主系统时钟的连接谐振器以及主系统时钟的外部时钟输入。

产生复位信号后将端口 12 设为输入模式。

图 4-22 至 4-24 显示了端口 12 的框图。

注意事项 P121 至 P124 的功能设置仅能在复位解除之后有效。端口一旦设为与振荡器相连, 则除非执行复位其不能再用作输入端口。

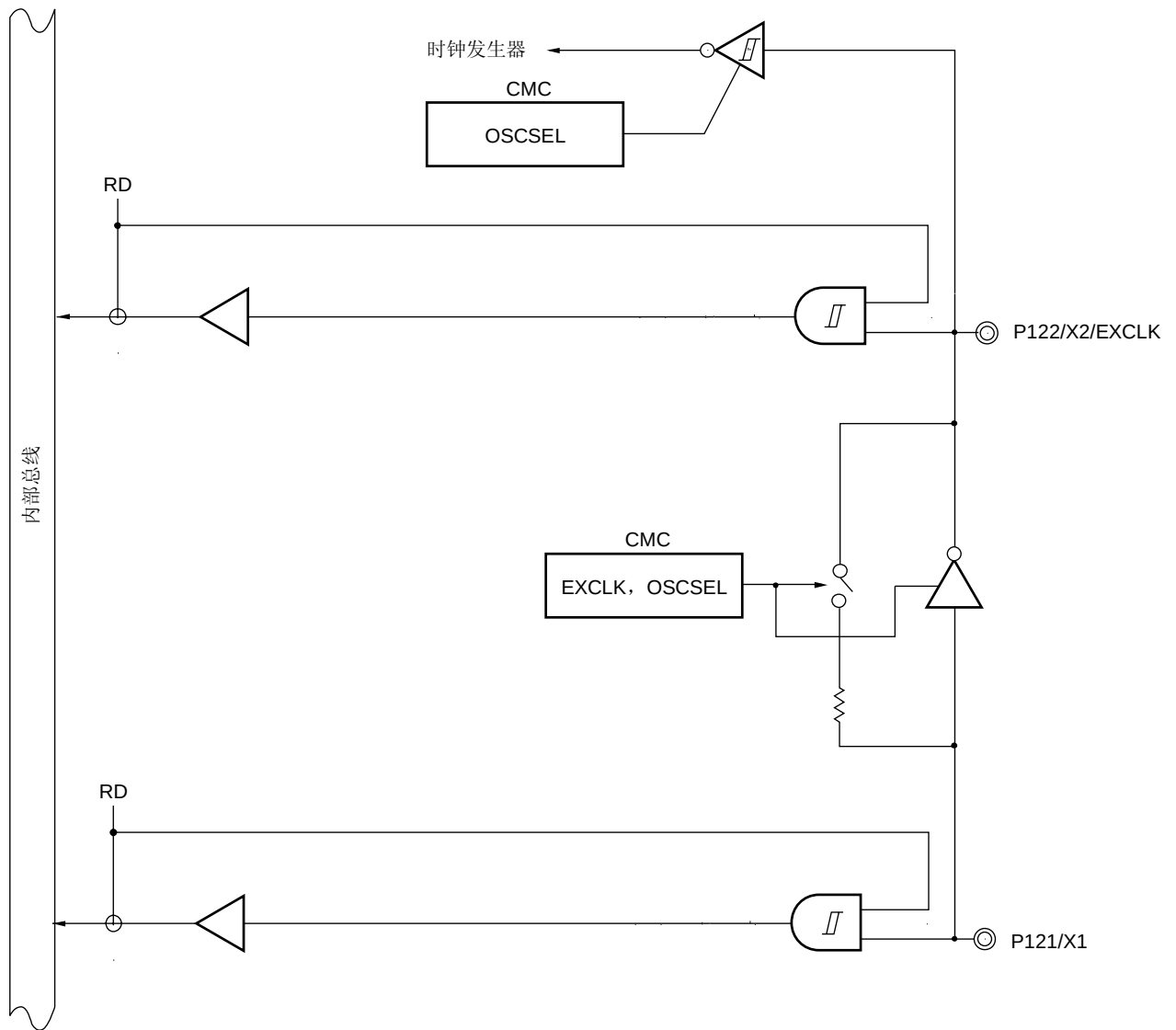
图 4-22. P120 的框图



P12: 端口寄存器 12
 PU12: 上拉电阻选项寄存器 12
 PM12: 端口模式寄存器 12
 RD: 读取信号
 WR_{xx}: 写入信号

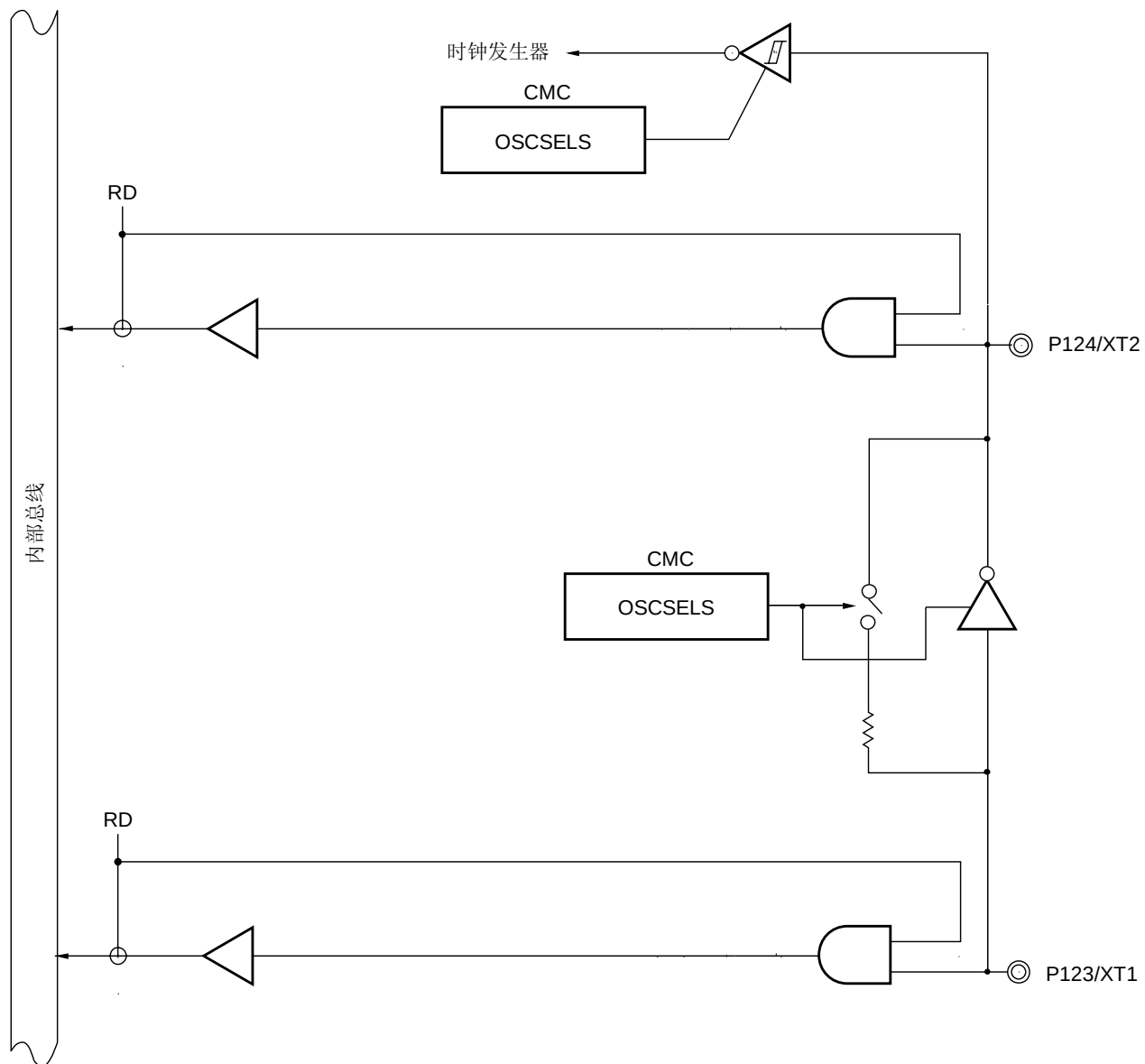
备注 对于不提供 EV_{DD} 或 EV_{SS} 引脚的产品，用 V_{DD} 代替 EV_{DD}，或用 V_{SS} 代替 EV_{SS}。

图 4-23. P121 和 P122 的框图



CMC: 时钟工作模式控制寄存器
RD: 读取信号

图 4-24. P123 和 P124 的框图



CMC: 时钟工作模式控制寄存器

RD: 读取信号

4.2.11 端口 14

	78K0R/KC3-L (44引脚) (μ PD78F100y: y = 0 至 3)	78K0R/KC3-L (48引脚) (μ PD78F100y: y = 1 至 3)	78K0R/KD3-L (μ PD78F100y: y = 4 至 6)	78K0R/KE3-L (μ PD78F100y: y = 7 至 9)
P140/PCLBUZ0	–	√	√	√
P141/ PCLBUZ1	–	–	–	√

备注 √: 已配备

P140 是专用于输出的端口并提供输出锁存器。

P141 是带有输出锁存器的 I/O 端口。 P141 可用端口模式寄存器 14（PM14）设置为以 1 位为单元的输入或输出模式。 当 P141 引脚用作输入端口时，片上上拉电阻可用上拉电阻选项寄存器 14（PU14）按 1 位为单元指定。

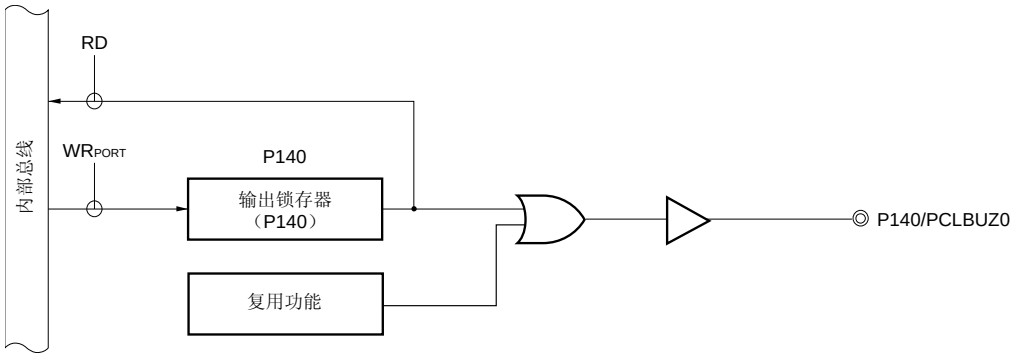
该端口还可用作时钟/蜂鸣器输出。

产生复位信号后将 P141 设为输入模式。

图 4-25 和 4-26 显示了端口 14 的框图。

注意事项 要使 P140/PCLBUZ0 和 P141/PCLBUZ1 用作通用端口，则需将时钟输出选择寄存器 0 和 1（CKS0，CKS1）的位 7 设为“0”，与它们的默认状态设置相同。

图 4-25. P140 的框图



P14: 端口寄存器 14
RD: 读取信号
WRxx: 写入信号

备注 当 P140 引脚用作端口功能引脚且复位有效时，P140 引脚输出低电平。 如果将 P140 设为输出高电平，则 P140 的输出信号可视为 CPU 复位信号而伪输出。

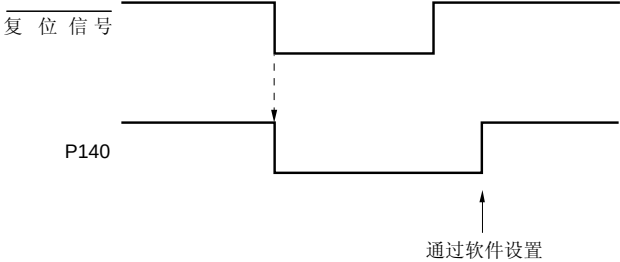
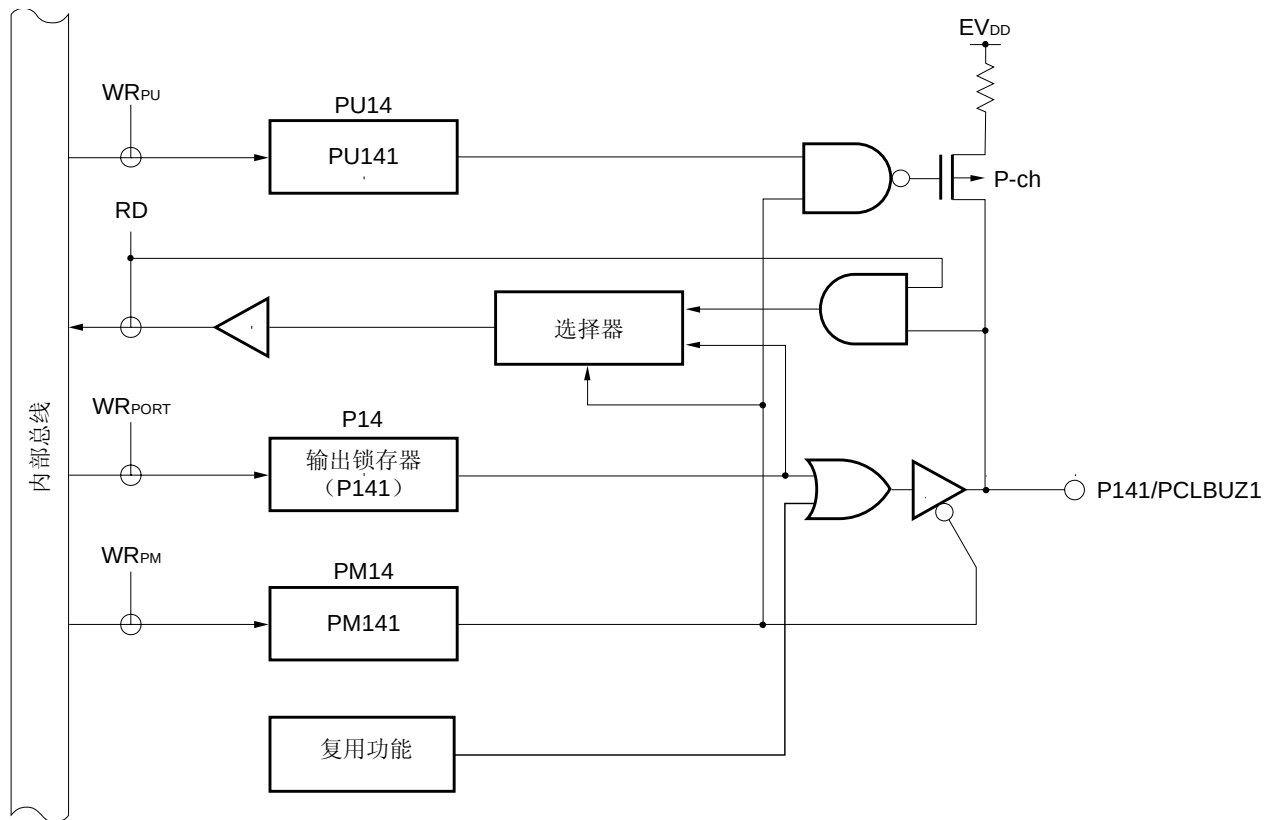


图 4-26. P141 的框图



P14: 端口寄存器 14
 PU14: 上拉电阻选项寄存器 14
 PM14: 端口模式寄存器 14
 RD: 读取信号
 WR_{xx} : 写入信号

4.2.12 端口 15

	78K0R/KC3-L (44引脚) (μPD78F100y: y = 0 至 3)	78K0R/KC3-L (48引脚) (μPD78F100y: y = 1 至 3)	78K0R/KD3-L (μPD78F100y: y = 4 至 6)	78K0R/KE3-L (μPD78F100y: y = 7 至 9)
P150/ANI8	√	√	√	√
P151/ANI9	√	√	√	√
P152/ANI10	–	√	√	√
P153/ANI11	–	–	–	√

备注 √: 已配备

端口 15 是带有输出锁存器的 I/O 端口。 端口 15 可用端口模式寄存器 15（PM15）设置为以 1 位为单元的输入或输出模式。

该端口还可用于 A/D 转换器模拟输入。

要将 P150/ANI8 至 P153/ANI11 用作数字输入引脚，在数字 I/O 模式下需用 A/D 端口配置寄存器（ADPC）对它们进行设置；在输入模式下需用 PM15 对它们进行设置。 从低位开始使用这些引脚。

要将 P150/ANI8 至 P153/ANI11 用作数字输出引脚，则在数字 I/O 模式下使用 ADPC 对 它们进行设置；在输出模式下使用 PM15 对它们进行设置。

表 4-6. P150/ANI8 至 P153/ANI11 引脚的功能设置

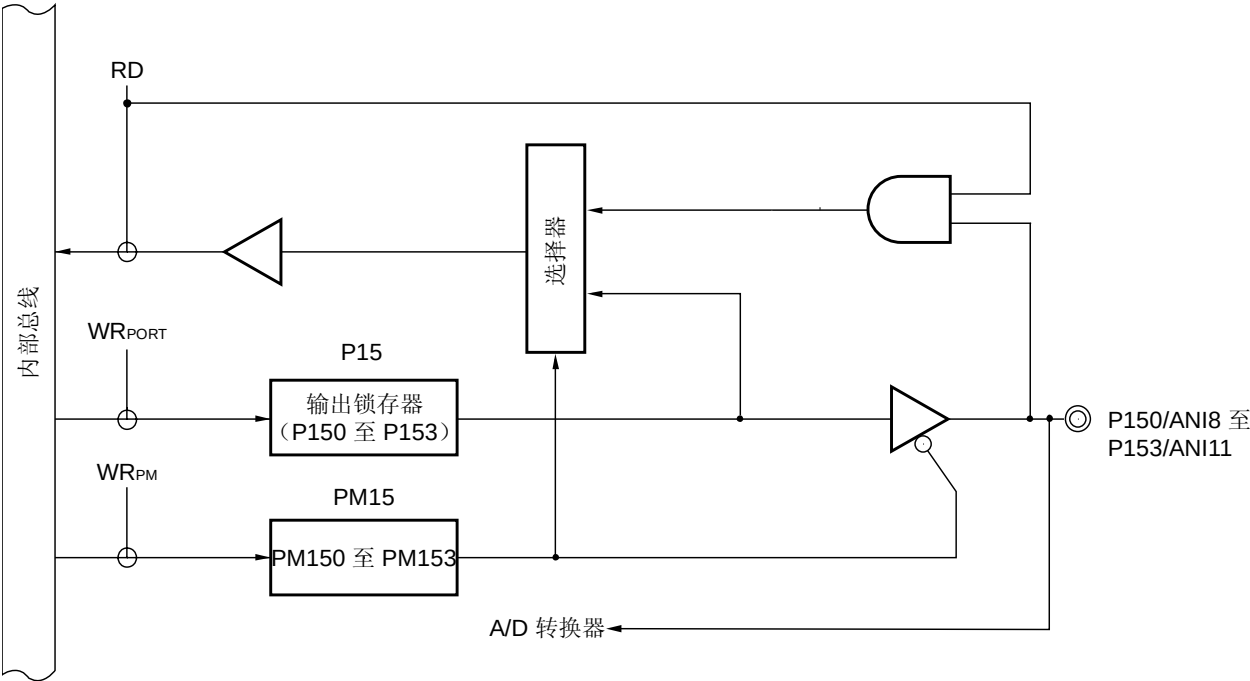
ADPC	PM15	ADS	P150/ANI8 至 P153/ANI11 Pins
数字 I/O 选择	输入模式	–	数字输入
	输出模式	–	数字输出
模拟输入选择	输入模式	选择ANI.	模拟输入（待转换）
		不选择 ANI.	模拟输入（不会被转换）
	输出模式	选择ANI.	禁止设置
		不选择ANI.	

产生复位信号时，在数字输入模式下设置 P150/ANI8 至 P153/ANI11 的所有引脚。

图 4-27 显示了端口 15 的框图。

注意事项 将端口 15 用作数字端口时，使引脚 AVREF 的电压与 VDD 引脚的电压相同。

图 4-27. P150 至 P153 的框图



P15: 端口寄存器 15
PM15: 端口模式寄存器 15
RD: 读取信号
WR_{xx}: 写入信号

4.3 控制端口功能的寄存器

端口功能由以下六种类型的寄存器控制。

- 端口模式寄存器 (PMxx)
- 端口寄存器 (Pxx)
- 上拉电阻选项寄存器 (PUxx)
- 端口输入模式寄存器 (PIM3、PIM7、PIM8)
- 端口输出模式寄存器 (POM3、POM7)
- A/D 端口配置寄存器 (ADPC)

(1) 端口模式寄存器 (PMxx)

这些寄存器按位单元为端口指定输入或者输出模式。

这些寄存器可由 1 位或 8 位存储器操作指令进行设置。

产生复位信号将这些寄存器设为 FFH (PM13 被设为 FEH)。

当端口引脚用作复用功能引脚时，应通过参考 **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 4-28. 端口模式寄存器的格式（78K0R/KC3-L）

符号	7	6	5	4	3	2	1	0	地址	复位后	R/W
PM1	1	1	1	1	PM13	PM12	PM11	PM10	FFF21H	FFH	R/W
PM2	PM27 PM26 PM25 PM24 PM23 PM22 PM21 PM20								FFF22H	FFH	R/W
PM3	1	1	1	1	1	PM32	PM31	PM30	FFF23H	FFH	R/W
PM4	1	1	1	1	1	1	PM41	PM40	FFF24H	FFH	R/W
PM5	1	1	1	1	1	PM52	PM51	PM50	FFF25H	FFH	R/W
PM6 [※]	1	1	1	1	1	1	PM61 [※]	PM60 [※]	FFF26H	FFH	R/W
PM7	1	1	PM75	PM74	PM73	PM72	PM71	PM70	FFF27H	FFH	R/W
PM8	1	1	1	1	PM83	PM82	PM81	PM80	FFF28H	FFH	R/W
PM12	1	1	1	1	1	1	1	PM120	FFF2CH	FFH	R/W
PM15	1	1	1	1	1	PM152 [※]	PM151	PM150	FFF2FH	FFH	R/W
PMmn	Pmn 引脚 I/O 模式选择 (m = 1 至 8、12、15; n = 0 至 7)										
0	输出模式 (输出缓存器开启)										
1	输入模式 (输出缓存器关闭)										

注 仅限于 48 引脚产品。

注意事项 务必将 PM1 的位 4 至 位 7、PM3 的位 3 至 位 7、PM4 的位 2 至位 7、PM5 的位 3 至 位 7、PM6 的位 2 至 位 7、PM6 的位 6 和 位 7、PM8 的位 4 至 位 7、PM12 的位 1 至 位 7 以及 PM15 的位 1 至 位 7 设为 1。

图 4-29. 端口模式寄存器的格式（78K0R/KD3-L）

符号	7	6	5	4	3	2	1	0	地址	复位后	R/W
PM0	1	1	1	1	1	1	PM01	PM00	FFF20H	FFH	R/W
PM1	1	1	1	1	PM13	PM12	PM11	PM10	FFF21H	FFH	R/W
PM2	PM27	PM26	PM25	PM24	PM23	PM22	PM21	PM20	FFF22H	FFH	R/W
PM3	1	1	1	1	1	PM32	PM31	PM30	FFF23H	FFH	R/W
PM4	1	1	1	1	1	1	PM41	PM40	FFF24H	FFH	R/W
PM5	1	1	1	1	1	PM52	PM51	PM50	FFF25H	FFH	R/W
PM6	1	1	1	1	1	1	PM61	PM60	FFF26H	FFH	R/W
PM7	PM77	PM76	PM75	PM74	PM73	PM72	PM71	PM70	FFF27H	FFH	R/W
PM8	1	1	1	1	PM83	PM82	PM81	PM80	FFF28H	FFH	R/W
PM12	1	1	1	1	1	1	1	PM120	FFF2CH	FFH	R/W
PM15	1	1	1	1	1	PM152	PM151	PM150	FFF2FH	FFH	R/W
PMmn	Pmn 引脚 I/O 模式选择 (m = 0 至 8、12、15; n = 0 至 7)										
0	输出模式 (输出缓存器开启)										
1	输入模式 (输出缓存器关闭)										

注意事项 务必将 **PM0** 的位 2 至 位 7、**PM1** 的位 4 至 位 7、**PM3** 的位 3 至 位 7、**PM4** 的位 2 至 位 7、**PM5** 的位 3 至 位 7、**PM6** 的位 2 和 位 7、**PM8** 的位 4 至 位 7、**PM12** 的位 1 至 位 7 以及 **PM15** 的位 3 至 位 7 设为 1。

图 4-30. 端口模式寄存器的格式（78K0R/KE3-L）

符号	7	6	5	4	3	2	1	0	地址	复位后	R/W
PM0	1	1	1	1	1	1	PM01	PM00	FFF20H	FFH	R/W
PM1	PM17	PM16	PM15	PM14	PM13	PM12	PM11	PM10	FFF21H	FFH	R/W
PM2	PM27	PM26	PM25	PM24	PM23	PM22	PM21	PM20	FFF22H	FFH	R/W
PM3	1	1	1	1	PM33	PM32	PM31	PM30	FFF23H	FFH	R/W
PM4	1	1	1	1	PM43	PM42	PM41	PM40	FFF24H	FFH	R/W
PM5	1	1	1	1	PM53	PM52	PM51	PM50	FFF25H	FFH	R/W
PM6	1	1	1	1	1	1	PM61	PM60	FFF26H	FFH	R/W
PM7	PM77	PM76	PM75	PM74	PM73	PM72	PM71	PM70	FFF27H	FFH	R/W
PM8	1	1	1	1	PM83	PM82	PM81	PM80	FFF28H	FFH	R/W
PM12	1	1	1	1	1	1	1	PM120	FFF2CH	FFH	R/W
PM14	1	1	1	1	1	1	PM141	0	FFF2EH	FEH	R/W
PM15	1	1	1	1	PM153	PM152	PM151	PM150	FFF2FH	FFH	R/W
PMmn	Pmn 引脚 I/O 模式选择 (m = 0 至 8、12、14、15; n = 0 至 7)										
0	输出模式（输出缓存器开启）										
1	输入模式（输出缓存器关闭）										

注意事项 务必将 PM0 的位 2 至 位 7、PM3 的位 4 至 位 7、PM4 的位 4 至 位 7、PM5 的位 4 至 位 7、PM6 的位 2 至 位 7、PM8 的位 4 至 位 7、PM12 的位 1 至 位 7、PM14 的位 2 至 位 7 以及 PM15 的位 4 至 位 7 设为 1。另外，务必要将 PM14 的位 0 设为 0。

(2) 端口寄存器 (Pxx)

这些寄存器设置端口的输出锁存器的值。
如果在输入模式中读取数据，则读取引脚电平。 如果在输出模式中读取，则读取输出锁存值^注。
这些可由 1 位或 8 位存储器操作指令设置。
产生复位信号将这些寄存器清除为 00H。

注 当 P2 和 P15 设置为 A/D 转换器的模拟输入功能时，如果在输入模式期间读取端口，则值总是为 0 且决不是读出的引脚电平。

图 4-31. 端口寄存器的格式 (78K0R/KC3-L)

符号	7	6	5	4	3	2	1	0	地址	复位后	R/W
P1	0	0	0	0	P13	P12	P11	P10	FFF01H	00H(输出锁存器)	R/W
P2	P27	P26	P25	P24	P23	P22	P21	P20	FFF02H	00H(输出锁存器)	R/W
P3	0	0	0	0	0	P32	P31	P30	FFF03H	00H(输出锁存器)	R/W
P4	0	0	0	0	0	0	P41	P40	FFF04H	00H(输出锁存器)	R/W
P5	0	0	0	0	0	P52	P51	P50	FFF05H	00H(输出锁存器)	R/W
P6 ^{注1}	0	0	0	0	0	0	P61 ^{注1}	P60 ^{注1}	FFF06H	00H(输出锁存器)	R/W
P7	0	0	P75	P74	P73	P72	P71	P70	FFF07H	00H(输出锁存器)	R/W
P8	0	0	0	0	P83	P82	P81	P80	FFF08H	00H(输出锁存器)	R/W
P12	0	0	0	P124	P123	P122	P121	P120	FFF0CH	未定义	R/W ^{注2}
P14 ^{注1}	0	0	0	0	0	0	0	P140 ^{注1}	FFF0EH	00H(输出锁存器)	R/W
P15	0	0	0	0	0	P152 ^{注1}	P151	P150	FFF0FH	00H(输出锁存器)	R/W
Pmn	m = 1 至 8、12、14、15; 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. P121 至 P124 为只读端口。
2. 仅限于 48 引脚产品。

图 4-32. 端口寄存器的格式（78K0R/KD3-L）

符号	7	6	5	4	3	2	1	0	地址	复位后	R/W
P0	0	0	0	0	0	0	P01	P00	FFF00H	00H(输出锁存器)	R/W
P1	0	0	0	0	P13	P12	P11	P10	FFF01H	00H(输出锁存器)	R/W
P2	P27	P26	P25	P24	P23	P22	P21	P20	FFF02H	00H(输出锁存器)	R/W
P3	0	0	0	0	0	P32	P31	P30	FFF03H	00H(输出锁存器)	R/W
P4	0	0	0	0	0	0	P41	P40	FFF04H	00H(输出锁存器)	R/W
P5	0	0	0	0	0	P52	P51	P50	FFF05H	00H(输出锁存器)	R/W
P6	0	0	0	0	0	0	P61	P60	FFF06H	00H(输出锁存器)	R/W
P7	P77	P76	P75	P74	P73	P72	P71	P70	FFF07H	00H(输出锁存器)	R/W
P8	0	0	0	0	P83	P82	P81	P80	FFF08H	00H(输出锁存器)	R/W
P12	0	0	0	P124	P123	P122	P121	P120	FFF0CH	未定义	R/W ^注
P14	0	0	0	0	0	0	0	P140	FFF0EH	00H(输出锁存器)	R/W
P15	0	0	0	0	0	P152	P151	P150	FFF0FH	00H(输出锁存器)	R/W
Pmn		m = 0 至 8、12、14、15; 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 P121 至 P124 为只读端口。

图 4-33. 端口寄存器的格式（78K0R/KE3-L）

符号	7	6	5	4	3	2	1	0	地址	复位后	R/W
P0	0	0	0	0	0	0	P01	P00	FFF00H	00H(输出锁存器)	R/W
P1	P17	P16	P15	P14	P13	P12	P11	P10	FFF01H	00H(输出锁存器)	R/W
P2	P27	P26	P25	P24	P23	P22	P21	P20	FFF02H	00H(输出锁存器)	R/W
P3	0	0	0	0	P33	P32	P31	P30	FFF03H	00H(输出锁存器)	R/W
P4	0	0	0	0	P43	P42	P41	P40	FFF04H	00H(输出锁存器)	R/W
P5	0	0	0	0	P53	P52	P51	P50	FFF05H	00H(输出锁存器)	R/W
P6	0	0	0	0	0	0	P61	P60	FFF06H	00H(输出锁存器)	R/W
P7	P77	P76	P75	P74	P73	P72	P71	P70	FFF07H	00H(输出锁存器)	R/W
P8	0	0	0	0	P83	P82	P81	P80	FFF08H	00H(输出锁存器)	R/W
P12	0	0	0	P124	P123	P122	P121	P120	FFF0CH	未定义	R/W ^注
P14	0	0	0	0	0	0	P141	P140	FFF0EH	00H(输出锁存器)	R/W
P15	0	0	0	0	P153	P152	P151	P150	FFF0FH	00H(输出锁存器)	R/W
Pmn	m = 0 至 8、12、14、15; 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 P121 至 P124 为只读端口。

(3) 上拉电阻选项寄存器（Puxx）

这些寄存器指定是否要使用片上上拉电阻。片上上拉电阻仅能以 1 位单元用于引脚输入模式的位设置，引脚上拉电阻的使用已在这些寄存器中设置。无论这些寄存器的设置怎样，都不能将片上上拉电阻连接至设为输出模式的位和用作复用功能输出引脚的位。

这些寄存器可由 1 位或 8 位存储器操作指令设置。

产生复位信号将这些寄存器清除为 00H。

图 4-34. 上拉电阻选项寄存器的格式（78K0R/KC3-L）

符号	7	6	5	4	3	2	1	0	地址	复位后	R/W
PU1	0	0	0	0	PU13	PU12	PU11	PU10	F0031H	00H	R/W
PU3	0	0	0	0	0	PU32	PU31	PU30	F0033H	00H	R/W
PU4	0	0	0	0	0	0	PU41	PU40	F0034H	00H	R/W
PU5	0	0	0	0	0	PU52	PU51	PU50	F0035H	00H	R/W
PU7	0	0	PU75	PU74	PU73	PU72	PU71	PU70	F0037H	00H	R/W
PU12	0	0	0	0	0	0	0	PU120	F003CH	00H	R/W
PUmn	Pmn 引脚片上上拉电阻的选择 (m = 1、3 至 5、7、12 ; n = 0 至 5)										
0	未连接片上上拉电阻										
1	已接了片上上拉电阻										

图 4-35. 上拉电阻选项寄存器的格式（978K0R/KD3-L）

符号	7	6	5	4	3	2	1	0	地址	复位后	R/W
PU0	0	0	0	0	0	0	PU01	PU00	F0030H	00H	R/W
PU1	0	0	0	0	PU13	PU12	PU11	PU10	F0031H	00H	R/W
PU3	0	0	0	0	0	PU32	PU31	PU30	F0033H	00H	R/W
PU4	0	0	0	0	0	0	PU41	PU40	F0034H	00H	R/W
PU5	0	0	0	0	0	PU52	PU51	PU50	F0035H	00H	R/W
PU7	PU77	PU76	PU75	PU74	PU73	PU72	PU71	PU70	F0037H	00H	R/W
PU12	0	0	0	0	0	0	0	PU120	F003CH	00H	R/W
PUmn	Pmn 引脚片上上拉电阻的选择 (m = 0、1、3 至 5、7、12；n = 0 至 7)										
0	未连接片上上拉电阻										
1	已连接片上上拉电阻										

图 4-36. 上拉电阻选项寄存器的格式（78K0R/KE3-L）

符号	7	6	5	4	3	2	1	0	地址	复位后	R/W
PU0	0	0	0	0	0	0	PU01	PU00	F0030H	00H	R/W
PU1	PU17	PU16	PU15	PU14	PU13	PU12	PU11	PU10	F0031H	00H	R/W
PU3	0	0	0	0	PU33	PU32	PU31	PU30	F0033H	00H	R/W
PU4	0	0	0	0	PU43	PU42	PU41	PU40	F0034H	00H	R/W
PU5	0	0	0	0	PU53	PU52	PU51	PU50	F0035H	00H	R/W
PU7	PU77	PU76	PU75	PU74	PU73	PU72	PU71	PU70	F0037H	00H	R/W
PU12	0	0	0	0	0	0	0	PU120	F003CH	00H	R/W
PU14	0	0	0	0	0	0	PU141	0	F003EH	00H	R/W
PUmn	Pmn 引脚片上上拉电阻的选择 (m = 0、1、3 至 5、7、12、14 ; n = 0 至 7)										
0	未连接片上上拉电阻										
1	已连接片上上拉电阻										

(4) 端口输入模式寄存器（PIM3、PIM7、PIM8）

PIM3 和 PIM7 寄存器按位单元设置 P31、P32、P71、P72、P74 或 P75 的输入缓冲器。

与不同电位的外部设备串行通信期间，可以选择 TTL 输入缓冲器。

PIM8 用于使能或禁止以 1 位为单元向 P80 至 P83 的数字输入。当使用比较器或者可编程增益放大器时，通过软件处理禁止数字输入（用作模拟输入）。要使用端口功能和复用功能，必须使能数字输入，因为默认情况下它们被禁止（用作模拟输入）。

这些寄存器可由 1 位或 8 位存储器操作指令设置。

产生复位信号将这些寄存器清除为 00H。

图 4-37. 输入模式寄存器的格式

符号	7	6	5	4	3	2	1	0	地址	复位后	R/W
PIM3	0	0	0	0	0	PIM32	PIM31	0	F0043H	00H	R/W
PIM7	0	0	PIM75	PIM74	0	PIM72	PIM71	0	F0047H	00H	R/W
PIM8	0	0	0	0	PIM83	PIM82	PIM81	PIM80	F0048H	00H	R/W
PIMmn	Pmn 引脚输入缓存器选择 (m = 3 和 7; n = 1、2、4、5)										
0	标准输入缓存器										
1	TTL 输入缓存器										
PIM8n	P8n 引脚数字输入缓存器选择 (n = 0 至 3)										
0	禁止数字输入（用作模拟输入）										
1	允许数字输入										

(5) 端口输出模式寄存器（POM3、POM7）

这些寄存器按 1 位为单元设置 P30 至 P32、P70、P72、P73 或 P75 的输出模式。
与不同电位的外部设备串行通信期间，可以选择 N 沟道漏极开路输出（V_{DD} 耐压）模式；与相同电位的外部设备进行简化的 I²C 通信期间，为 SDA10 引脚。
这些寄存器可由 1 位或 8 位存储器操作指令设置。
产生复位信号将这些寄存器清除为 00H。

图 4-38. 端口输出模式寄存器的格式

符号	7	6	5	4	3	2	1	0	地址	复位后	R/W
POM3	0	0	0	0	0	POM32	POM31	POM30	F0053H	00H	R/W
POM7	0	0	POM75	0	POM73	POM72	0	POM70	F0057H	00H	R/W
POMmn	Pmn 引脚输出模式选择 (m = 3 和 7; n = 0 至 3 和 5)										
0	标准输出模式										
1	N 沟道漏极开路输出（V _{DD} 耐压）模式										

(6) A/D 端口配置寄存器 (ADPC)

该寄存器将 ANI0/P20 至 ANI7/P27 和 ANI8/P150 至 ANI11/P153 引脚转换为端口的数字 I/O 或 A/D 转换器的模拟输入。

ADPC 可由 8 位存储器操作指令设置。

复位信号产生后该寄存器被设为 10H。

图 4-39. A/D 端口配置寄存器 (ADPC) 的格式

地址: F0017H 复位后: 10H R/W

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

ADPC4	ADPC3	ADPC2	ADPC1	ADPC0	模拟输入 (A) / 数字 I/O (D) 转换											
					端口 15						端口 2					
					ANI11 /P153	ANI10 /P152	ANI9 /P151	ANI8 /P150	ANI7 /P27	ANI6 /P26	ANI5 /P25	ANI4 /P24	ANI3 /P23	ANI2 /P22	ANI1 /P21	ANI0 /P20
0	0	0	0	0	A	A	A	A	A	A	A	A	A	A	A	A
0	0	0	0	1	A	A	A	A	A	A	A	A	A	A	A	D
0	0	0	1	0	A	A	A	A	A	A	A	A	A	A	D	D
0	0	0	1	1	A	A	A	A	A	A	A	A	A	D	D	D
0	0	1	0	0	A	A	A	A	A	A	A	A	D	D	D	D
0	0	1	0	1	A	A	A	A	A	A	A	D	D	D	D	D
0	0	1	1	0	A	A	A	A	A	A	D	D	D	D	D	D
0	0	1	1	1	A	A	A	A	A	D	D	D	D	D	D	D
0	1	0	0	0	A	A	A	A	D	D	D	D	D	D	D	D
0	1	0	0	1	A	A	A	D	D	D	D	D	D	D	D	D
0	1	0	1	0	A	A	D	D	D	D	D	D	D	D	D	D
0	1	0	1	1	A	D	D	D	D	D	D	D	D	D	D	D
1	0	0	0	0	D	D	D	D	D	D	D	D	D	D	D	D
其它					禁止设置											

- 注意事项**
1. 在输入模式下，使用短路模式寄存器 2 和 15 (PM2、PM15) 为 A/D 转换设置一个通道。
 2. 不要通过模拟输入通道指定寄存器 (ADS) 将 ADPC 设置的引脚设为数字 I/O。

备注

P20/ANI0-P27/ANI7、P150/ANI8 和 P151/ANI9: 78K0R/KC3-L (44 引脚)

P20/ANI0-P27/ANI7、P150/ANI8 至 P152/ANI10: 78K0R/KC3-L (48 引脚)、78K0R/KD3-L

P20/ANI0-P27/ANI7、P150/ANI8 至 P153/ANI11: 78K0R/KE3-L

4.4 端口功能操作

端口操作取决于是否设置了输入或输出模式，如下所示。

4.4.1 写入 I/O 端口

(1) 输出模式

通过一个传输指令将一个值写入输出锁存器，且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，则会一直保持直到再一次数据写入输出锁存器。

产生复位信号时，输出锁存器的内容会被清除。

(2) 输入模式

通过传输指令将一个值写入输出锁存器，但是因为输出缓存器关闭所以引脚状态不发生改变。

数据一旦被写入输出锁存器，则会一直保持直到再一次数据写入输出锁存器。

产生复位信号时，输出锁存器的内容会被清除。

4.4.2 从 I/O 端口读取

(1) 输出模式

输出锁存器的内容由传输指令读取。输出锁存器的内容不发生改变。

(2) 输入模式

引脚状态由传输指令读取。输出锁存器的内容不发生改变。

4.4.3 I/O 端口的操作

(1) 输出模式

在输出锁存器的内容中执行操作，结果写入输出锁存器。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，则会一直保持直到再一次数据写入输出锁存器。

产生复位信号时，输出锁存器的内容会被清除。

(2) 输入模式

读取引脚电平且在其内容中执行操作。操作结果被写入输出锁存器，但是因为输出缓存器关闭，所以引脚状态不发生改变。

产生复位信号时，输出锁存器的内容会被清除。

4.4.4 与具有不同电位的外部设备（2.5 V、3 V）相连接

当端口 3 和 7 在 $V_{DD} = 4.0\text{ V}$ 至 5.5 V 条件下工作时，I/O 可与工作在 2.5V 或 3 V 供电电压的外部设备连接。

对于输入，可通过端口输入模式寄存器 3 和 7（PIM3 和 PIM7），基于逐位进行标准输入（CMOS）/TTL 转换。

此外，对于输出，通过使用端口输出模式寄存器 3 和 7（POM3 和 POM7）将输出缓存器转换为 N 沟道漏极开路（ V_{DD} 耐压）来提供不同的电位。

(1) 使用 UART0、UART1、CSI00、CSI01 和 CSI1 功能的 I/O 引脚时的设置步骤

(a) 用作 2.5V 或 3 V 输入端口

<1> 复位后，端口模式为输入模式（高阻）。

<2> 如果需要上拉，则在外部上拉要使用的引脚（不能使用片上上拉电阻）。

UART0 的情况：P74

UART1 的情况：P31

CSI00 的情况：P74、P75

CSI01 的情况：P71、P72

CSI10 的情况：P31、P32

<3> 将 PIM3 和 PIM7 寄存器的相应位设为 1，以转换至 TTL 输入缓存器。

<4> V_{IH}/V_{IL} 在 2.5V 或 3 V 的工作电压工作。

(b) 用作 2.5V 或 3 V 输出端口

<1> 复位后，端口模式变为输入模式（高阻）。

<2> 在外部上拉要使用的引脚（不能使用片上上拉电阻）。

UART0 的情况：P73

UART1 的情况：P30

CSI00 的情况：P73、P75

CSI01 的情况：P70、P72

CSI10 的情况：P30、P32

<3> 将相应端口的输出锁存器设为 1。

<4> 将 POM3 和 POM7 寄存器的相应位设为 1，以设置 N 沟道漏极开路输出（ V_{DD} 耐压）模式。

<5> 通过操作 PM3 和 PM7 寄存器设置输出模式。

此时，输出数据为高电平，因此引脚处于高阻状态。

<6> 通过设置串行阵列部件开始通信。

(2) 使用简化的 IIC10 功能的 I/O 引脚时的设置步骤

- <1> 复位后，端口模式为输入模式（高阻）。
- <2> 在外部上拉要使用的引脚（不能使用片上上拉电阻）。

简化的 IIC10 的情况： P31、P32

- <3> 将相应端口的输出锁存器设为 1。
- <4> 将 POM3 寄存器的相应位设为 1，以设置 N 沟道漏极开路输出（V_{DD} 耐压）模式。
- <5> 将 PM3 寄存器的相应位设为输出模式（数据 I/O 在输出模式下可用）。
此时，输出数据为高电平，因此引脚处于高阻状态。
- <6> 使能串行阵列部件的操作并将模式设为简化的 I²C 模式。

4.5 使用复用功能时端口寄存器和输出锁存器的设置

要使用端口引脚的复用功能，需如表 4-7 所示设置端口模式寄存器和输出锁存器。

表 4-7. 使用复用功能时端口寄存器和输出锁存器的设置 (1/2)

引脚名称	复用功能		PM _{xx}	P _{xx}
	功能名称	I/O		
P00	TI00	输入	1	×
P01	TO00	输出	0	0
P10	TI02	输入	1	×
	TO02	输出	0	0
P11	TI03	输入	1	×
	TO03	输出	0	0
P12	TI04	输入	1	×
	TO04	输出	0	0
	RTCDIV	输出	0	0
	RTCCL	输出	0	0
P13	TI05	输入	1	×
	TO05	输出	0	0
P14 ^{※1}	TI06 ^{※1}	输入	1	×
	TO06 ^{※1}	输出	0	0
P15 ^{※1}	TI07 ^{※1}	输入	1	×
	TO07 ^{Note 1}	输出	0	0
P20 to P27 ^{※2}	ANI0 to ANI7 ^{※2}	输入	1	×
P30	SO10	输出	0	1
	TxD1	输出	0	1
P31	SI10	输入	1	×
	RxD1	输入	1	×
	SDA10	I/O	0	1
	INTP1	输入	1	×
P32	SCK10	输入	1	×
		输出	0	1
	SCL10	I/O	0	1
	INTP2	输入	1	×
P40	TOOL0	I/O	×	×
P41	TOOL1	输出	×	×
P50 ^{※1}	TI06 ^{※1}	输入	1	×
	TO06 ^{※1}	输出	0	0
P51 ^{※1}	TI07 ^{※1}	输入	1	×
	TO07 ^{※1}	输出	0	0

备注 ×: 忽略
PM_{xx}: 端口模式寄存器
P_{xx}: 端口输出锁存器

(注 在下页的下一页上列出。)

表 4-7. 使用复用功能时端口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		
P52	RTC1HZ	输出	0	0
	SLTI	输入	1	×
	SLTO	输出	0	0
P60	SCL0	I/O	0	0
P61	SDA0	I/O	0	0
P70	KR0	输入	1	×
	SO01	输出	0	1
	INTP4	输入	1	×
P71	KR1	输入	1	×
	SI01	输入	1	×
	INTP5	输入	1	×
P72	KR2	输入	1	×
	$\overline{\text{SCK01}}$	输入	1	×
		输出	0	1
	INTP6	输入	1	×
P73	KR3	输入	1	×
	SO00	输出	0	1
	TxD0	输出	0	1
P74	KR4	输入	1	×
	SI00	输入	1	×
	RxD0	输入	1	×
P75	KR5	输入	1	×
	$\overline{\text{SCK00}}$	输入	1	×
		输出	0	1
P76	KR6	输入	1	×
P77	KR7	输入	1	×
P80 ^{‡2}	CMP0P	输入	1	×
	INTP3	输入	1	×
	PGAI ^{‡2}	输入	1	×
P81	CMP0M	输入	1	×
P82	CMP1P	输入	1	×
	INTP7	输入	1	×
P83	CMP1M	输入	1	×
P120	INTP0	输入	1	×
	EXLVI	输入	1	×
P140	PCLBUZ0	输出	0	0
P141	PCLBUZ1	输出	0	0
P150 至 P153 ^{‡2}	ANI8 至 ANI11 ^{‡2}	输入	1	×

备注 ×: 忽略

PM_{xx}: 端口模式寄存器P_{xx}: 端口输出锁存器

(注 在下一页上列出。)

- 注 1. TI06/TO06 和 TI07/TO07 引脚共用的端口根据产品不同而不同。
 78K0R/KC3-L、78K0R/KD3-L: P50/TI06/TO06、P51/TI07/TO07
 78K0R/KE3-L: P14/TI06/TO06、P15/TI07/TO07
2. ANI0/P20 至 ANI7/P27、ANI8/P150 至 ANI11/P153 和 PGAI/P80 引脚的功能可使用 A/D 端口配置寄存器 (ADPC)、模拟输入通道指定寄存器 (ADS)、PM2、PM15 和 PM8 选择。

表 4-8. ANI0/P20 至 ANI7/P27、ANI8/P150 至 ANI11/P153 和 PGAI/P80 引脚的功能设置

ADPC	PM2、PM15、PM8	ADS	ANI0/P20 至 ANI7/P27, ANI8/P150 至 ANI11/P153, 和 PGAI/P80 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	选择 ANI.	模拟输入 (待转换)
		不选择 ANI.	模拟输入 (不会转换)
	输出模式	选择 ANI.	禁止设置
		不选择 ANI.	

- 备注 P20/ANI0-P27/ANI7、P150/ANI8 和 P151/ANI9: 78K0R/KC3-L (44 引脚)
 P20/ANI0-P27/ANI7、P150/ANI8 至 P152/ANI10: 78K0R/KC3-L (48 引脚)、78K0R/KD3-L
 P20/ANI0-P27/ANI7、P150/ANI8 至 P153/ANI11: 78K0R/KE3-L

4.6 端口寄存器 n（Pn）的 1 位操作指令的注意事项

在同时具有输入和输出功能的端口执行 1 位操作指令时，除了正常写入目标位外，还可能对操作对象之外的输入端口的输出锁存器写入不需要的值。

因此，当端口由输入模式转换为输出模式时，建议重写输出锁存器。

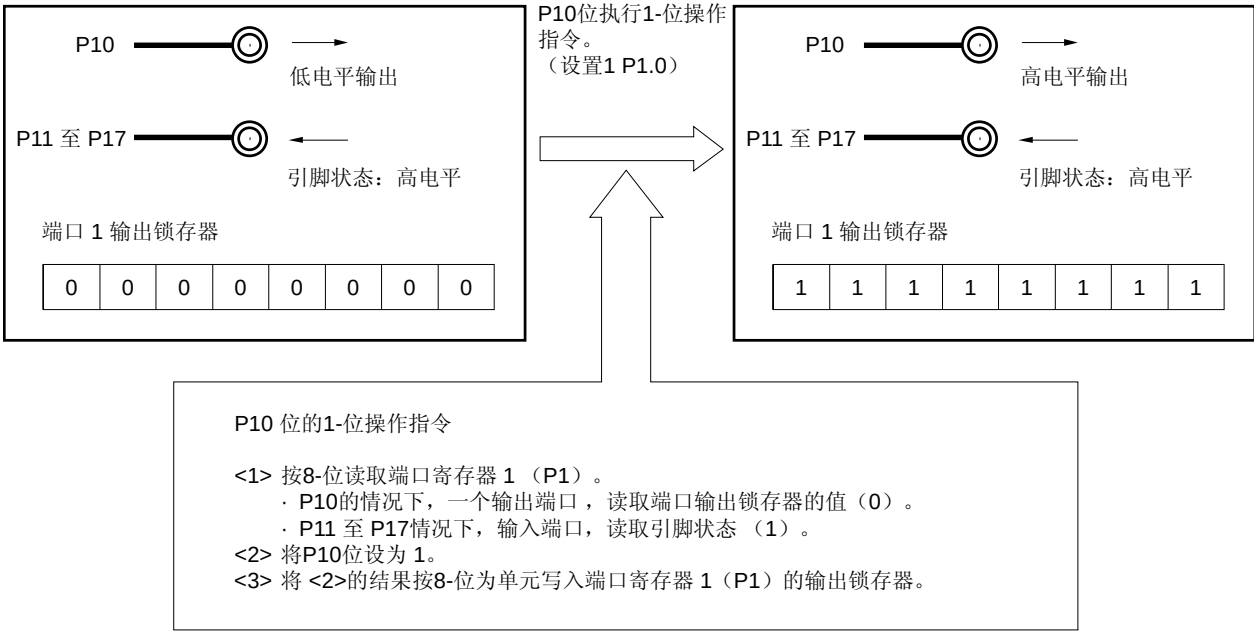
<示例> 当 P10 为输出端口，端口 P11 至 P17 为输入端口（所有的引脚状态为高电平），且端口 1 的输出锁存器的值为 00H 时，如果通过位操作指令将输出端口 P10 的输出由低电平变为高电平，则端口 1 的输出锁存器的值为 FFH。

解释： 写入和读取 PMnm 位为 1 的端口的 Pn 寄存器的目标分别为输出锁存和引脚状态。
78K0R/Kx3-L 单片机按以下次序执行位处理指令：

- <1> 按 8 位读取 Pn 寄存器。
- <2> 处理目标位。
- <3> 按 8 位写入 Pn 寄存器。

在步骤 <1>中，当读取作为输入端口的 P11 至 P17 引脚状态时，作为输出端口的 P10 引脚的输出锁存值（0）被读取。 如果此时 P11 至 P17 的引脚状态为高电平，则读取的值为 FEH。
通过在<2>中的操作该值变成 FFH。
通过在<3>中的操作 FFH 被写入输出锁存器。

图 4-40. 位操作指令（P10）



5.1 时钟发生器的功能

时钟发生器产生供应 CPU 和外围硬件的时钟。
可选择以下三种系统时钟和时钟振荡器。

(1) 主系统时钟

<1> X1 振荡器

通过将一个谐振器连接到 X1 和 X2，该电路按 $f_x = 2 \sim 20 \text{ MHz}$ 的时钟振荡。
通过执行 STOP 指令或设置 MSTOP 可以停止振荡（时钟运行状态控制寄存器（CSC）的位 7）。

<2> 内部高速振荡器[※]

该电路按 $f_{IH} = 1$ 和 8 MHz (TYP.) 的时钟振荡。复位解除后，CPU 总是在该内部高速振荡时钟下开始运行。
通过执行 STOP 指令或者设置 HIOSTOP（CSC 的位 0）可以停止振荡。

<3> 20 MHz 内部高速振荡时钟振荡器[※]

该电路按 $f_{IH20} = 20 \text{ MHz (TYP.)}$ 的时钟振荡。可通过在 $V_{DD} \geq 2.7 \text{ V}$ 时将 20 MHz 内部高速振荡控制寄存器（DSCCTL）的位 0（DSCON）设为 1 开始振荡。将 DSCON 设为 0 停止振荡。

注 要使用 1、8 或 20 MHz 内部高速振荡时钟，需预先使用选项字节来设置频率（有关详情，请参见第二十三章 选项字节）。另外，在复位解除后内部高速振荡器自动开始振荡。要使用 20 MHz 内部高速振荡操作微控制器，需通过将 20 MHz 内部高速振荡控制寄存器（DSCCTL）的位 0（DSCON）设为 1 开始振荡。

也能从 EXCLK/X2/P122 引脚提供一个外部主系统时钟（ $f_{EX} = 2 \sim 20 \text{ MHz}$ ）。通过执行 STOP 指令或者设置 MSTOP，可以禁止外部主系统时钟输入。
作为主系统时钟，通过设置 MCM0（系统时钟控制寄存器（CKC）的位 4）可以选择高速系统时钟（X1 时钟或者外部主系统时钟）或者内部高速振荡时钟。

(2) 副系统时钟

● XT1 时钟振荡器

通过将一个 32.768 kHz 的谐振器连接到 XT1 和 XT2，该电路按 $f_{SUB} = 32.768 \text{ kHz}$ 的时钟振荡。通过设置 XTSTOP（CSC 的位 6）可以停止振荡。

备注

- f_x : X1 时钟振荡频率
- f_{IH} : 内部高速振荡时钟频率
- f_{IH20} : 20 MHz 内部高速振荡时钟频率
- f_{EX} : 外部主系统时钟频率
- f_{SUB} : 副系统时钟频率

(3) 内部低速振荡时钟（专用于看门狗定时器的时钟）

- **内部低速振荡器**

该电路按 $f_{IL} = 30 \text{ kHz (TYP.)}$ 的时钟振荡。

内部低速振荡时钟不能用作 CPU 时钟。 在内部低速振荡时钟下工作的仅有的硬件是看门狗定时器。

当看门狗定时器停止时振荡停止。

备注

1. f_{IL} : 内部低速振荡时钟频率
2. 看门狗定时器在以下情况下停止运行。
 - 选项字节 (000C0H) 的位 4 (WDTON) = 0 时
 - 当选项字节 (000C0H) 的位 4 (WDTON) = 1 且位 0 (WDSTBYON) = 0 时，若执行 HALT 或 STOP 指令

5.2 时钟发生器的配置

时钟发生器包含以下硬件。

表 5-1. 时钟发生器的配置

项目	配置
控制寄存器	时钟运行模式控制寄存器 (CMC) 时钟运行状态控制寄存器 (CSC) 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) 系统时钟控制寄存器 (CKC) 20 MHz 内部高速振荡控制寄存器 (DSCCTL) 外设使能寄存器 0, 1, 2 (PER0, PER1, PER2) 运行速度模式控制寄存器 (OSMC)
振荡器	X1 振荡器 XT1 振荡器 内部高速振荡器 内部低速振荡器

(注和备注在下一页中列出。)



注 78K0R/KC3-L 的 44 引脚产品未配置此项。

备注	f _x :	X1 时钟振荡频率
	f _{ih} :	内部高速振荡时钟频率
	f _{ih20} :	20 MHz 内部高速振荡时钟频率
	f _{ex} :	外部主系统时钟频率
	f _{mx} :	高速系统时钟频率
	f _{main} :	主系统时钟频率
	f _{xt} :	XT1 时钟振荡频率
	f _{sub} :	副系统时钟频率
	f _{clk} :	CPU/外设硬件时钟频率
	f _{il} :	内部低速振荡时钟频率

5.3 控制时钟发生器的寄存器

以下八种寄存器用于控制时钟发生器。

- 时钟工作模式控制寄存器 (CMC)
- 时钟运行状态控制寄存器 (CSC)
- 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)
- 系统时钟控制寄存器 (CKC)
- 20 MHz 内部高速振荡控制寄存器 (DSCCTL)
- 外设使能寄存器 0、1、2 (PER0、PER1、PER2)
- 运行速度模式控制寄存器 (OSMC)

(1) 时钟工作模式控制寄存器 (CMC)

该寄存器用于设置 X1/P121、X2/EXCLK/P122、XT1/P123 和 XT2/P124 引脚的工作模式，并选择振荡器的增益。复位之后，CMC 仅可使用 8 位存储器操作指令写入一次。该寄存器可由 1 位或 8 位存储器操作指令读取。产生复位信号将该寄存器清除为 00H。

图 5-2. 时钟工作模式控制寄存器（CMC）的格式

地址： FFFA0H 复位后： 00H R/W

符号	7	6	5	4	3	2	1	0
CMC	EXCLK	OSCSEL	0	OSCSELS	0	AMPHS1	AMPHS0	AMPH

EXCLK	OSCSEL	高速系统时钟引脚工作模式	X1/P121 引脚	X2/EXCLK/P122 引脚
0	0	输入端口模式	输入端口	
0	1	X1 振荡模式	晶体/陶瓷谐振器连接	
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

OSCSELS	副系统时钟引脚工作模式	XT1/P123 引脚	XT2/P124 引脚
0	输入端口模式	输入端口	
1	XT1 振荡模式	晶体谐振器连接	

AMPHS1	AMPHS0	XT1 振荡器振荡模式选择
0	0	低功耗振荡（默认）
0	1	标准振荡
1	0	超低功耗振荡
1	1	

AMPH	高速系统时钟振荡频率的控制
0	$2\text{ MHz} \leq f_{\text{MX}} \leq 10\text{ MHz}$
1	$10\text{ MHz} < f_{\text{MX}} \leq 20\text{ MHz}$

- 注意事项
1. 复位之后，CMC 仅可使用 8 位存储器操作指令写入一次。
 2. 复位之后且在 X1 或 XT1 振荡开始前，按照设置时钟工作状态控制寄存器（CSC）设置 CMC。
 3. 如果 X1 时钟振荡频率超过了 10 MHz，则务必将 AMPH 设为 1。
 4. 当在默认值（00H）情况下使用 CMC 时，务必在复位后将该寄存器设为 00H，以在程序循环期间防止误操作。
 5. XT1 振荡器是低放大倍数来达到低功耗的电路。设计电路时需注意以下几点。
 - 引脚和电路包括附加电容。因此，要用实际使用的电路板进行振荡评估并确认不存在问题。
 - 当使用超低功耗振荡（AMPHS1 = 1）作为 XT1 的振荡模式时，要使用第二十八章电气规范（目标）中说明的推荐的谐振器。
 - 使 XT1 和 XT2 引脚与谐振器之间的连线尽可能最短，且附加电容和接线电阻最小。选择超低功耗振荡（AMPHS1 = 1）时尤其要注意该点。

（注意事项 和 备注在下页给出）。

- 使用线电阻很小的材料配置电路板的电路。
- 在 XT1 振荡器附近，尽可能多地放置与 V_{SS} 电压相同的接地点。
- 确保 XT1 和 XT2 引脚与谐振器之间的连线不与其它信号线交叉。布线时不要将该连线靠近通过大波动电流的信号线。
- 在电路板处于高湿度环境下或露水凝缩时，由于电路板吸湿，XT1 和 XT2 引脚之间的阻抗可能会变小且振荡也可能失常。当在这样的环境下使用电路板时，应采取措
施，诸如涂层来防止电路板受潮。
- 给电路板涂层时，应使用不会引起 XT1 和 XT2 引脚电容或泄漏的材料。

备注 f_{MX}: 高速系统时钟频率

(2) 时钟工作状态控制寄存器 (CSC)

该寄存器用于控制高速系统时钟、内部高速振荡时钟和副系统时钟（除 20 MHz 内部高速振荡时钟和内部低速振荡时钟）的操作。

CSC 可由 1 位或 8 位存储器操作指令设置。

复位信号产生后该寄存器被设为 C0H。

图 5-3. 时钟工作模式状态控制寄存器 (CSC) 的格式

地址: FFFA1H 复位后: C0H R/W

符号	<7>	<6>	5	4	3	2	1	<0>
CSC	MSTOP	XTSTOP	0	0	0	0	0	HIOSTOP

MSTOP	高速系统时钟工作控制		
	X1 振荡模式	外部时钟输入模式	输入端口模式
0	X1 振荡器运行	来自 EXCLK 引脚的外部时钟有效。	输入端口
1	X1 振荡器停止	来自 EXCLK 引脚的外部时钟无效。	

XTSTOP	副系统时钟工作控制	
	XT1 振荡模式	输入端口模式
0	XT1 振荡器运行	输入端口
1	XT1 振荡器停止	

HIOSTOP	内部高速振荡时钟工作控制	
	0	内部高速振荡器运行
	1	内部高速振荡器停止

(注意事项 在下一页中列出。)

- 注意事项**
1. 复位后，在设置 **CSC** 之前设置时钟工作模式控制寄存器（**CMC**）。
 2. 要开始由 **MSTOP** 所设置的 **X1** 振荡，需使用振荡稳定时间计数器状态寄存器（**OSTC**）检查 **X1** 时钟的振荡稳定时间。
 3. 不要停止由 **OSC** 寄存器为 **CPU** 外设硬件时钟（**f_{CLK}**）选中的时钟。
 4. 停止时钟振荡（使得外部时钟输入无效）的寄存器的标志设置和时钟振荡将要停止前的条件如下所示。

表 5-2. 停止时钟振荡之前的条件和标志设置

时钟	停止时钟之前的条件 (使外部时钟输入无效)	CSC 寄存器标志 的设置
X1 时钟	CPU 和外设硬件时钟不使用高速系统时钟工作。 (CLS = 0 且 MCS = 0, 或 CLS = 1)	MSTOP = 1
外部主系统时钟		
副系统时钟	CPU 和外设硬件时钟不使用主系统时钟工作。 (CLS = 0)	XTSTOP = 1
内部高速振荡时钟	CPU 和外设硬件时钟不使用内部高速振荡时钟工作。 (CLS = 0 且 MCS = 1, 或 CLS = 1)	HIOSTOP = 1

(3) 振荡稳定时间计数器状态寄存器（OSTC）

这是指示 X1 时钟振荡稳定时间计数器的计数状态的寄存器。

可以在下列情况中检测 X1 时钟振荡时间。

- 如果 X1 时钟在内部高速振荡时钟或副系统时钟被用作 CPU 时钟期间开始振荡。
- 如果在内部高速振荡时钟用作 CPU 时钟且随 X1 时钟振荡期间输入 STOP 模式，然后解除。

OSTC 可由 1 位或 8 位存储器操作指令设置。

当产生复位信号时，STOP 指令和 MSTOP（CSC 寄存器的位 7）= 1 将 OSTC 清为 00H。

备注 振荡稳定时间计数器在以下情况下开始计数。

- X1 时钟的振荡开始（EXCLK, OSCSEL = 0, 1 → MSTOP = 0）时
- STOP 模式解除时

图 5-4. 振荡稳定时间计数器状态寄存器（OSTC）的格式

地址：FFFA2H 复位后：00H R

符号

OSTC

	7	6	5	4	3	2	1	0
	MOST	MOST	MOST	MOST	MOST	MOST	MOST	MOST
	8	9	10	11	13	15	17	18

MOST	MOST	MOST	MOST	MOST	MOST	MOST	MOST	指定稳定时间状态		
									$f_x = 10 \text{ MHz}$	$f_x = 20 \text{ MHz}$
0	0	0	0	0	0	0	0	最大 $2^8/f_x$	最大 $25.6 \mu\text{s}$	最大 $12.8 \mu\text{s}$
1	0	0	0	0	0	0	0	最小 $2^9/f_x$	最小 $25.6 \mu\text{s}$	最小 $12.8 \mu\text{s}$
1	1	0	0	0	0	0	0	最小 $2^9/f_x$	最小 $51.2 \mu\text{s}$	最小 $25.6 \mu\text{s}$
1	1	1	0	0	0	0	0	最小 $2^{10}/f_x$	最小 $102.4 \mu\text{s}$	最小 $51.2 \mu\text{s}$
1	1	1	1	0	0	0	0	最小 $2^{11}/f_x$	最小 $204.8 \mu\text{s}$	最小 $102.4 \mu\text{s}$
1	1	1	1	1	0	0	0	最小 $2^{13}/f_x$	最小 $819.2 \mu\text{s}$	最小 $409.6 \mu\text{s}$
1	1	1	1	1	1	0	0	最小 $2^{15}/f_x$	最小 3.27 ms	最小 1.64 ms
1	1	1	1	1	1	1	0	最小 $2^{17}/f_x$	最小 13.11 ms	最小 6.55 ms
1	1	1	1	1	1	1	1	最小 $2^{18}/f_x$	最小 26.21 ms	最小 13.11 ms

注意事项 1. 以上的时间过去之后，按照从 MOST8 开始的顺序将位设为 1 并保持为 1。

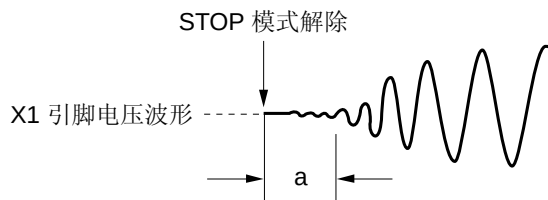
2. 振荡稳定时间计数器通过 OSTC 设置为振荡稳定时间向上计数。

在下列情况下，振荡开始后，将 OSTC 的振荡稳定时间设为大于 OSTC 寄存器要检查的计数值的值。

- 如果 X1 时钟在内部高速振荡时钟或副系统时钟被用作 CPU 时钟期间开始振荡。
- 如果在内部高速振荡时钟用作 CPU 时钟且随 X1 时钟振荡期间输入 STOP 模式，然后解除。

（因此，要注意，在解除 STOP 模式之后仅将 OSTC 设置振荡稳定时间之前的状态设置至 OSTC。）

3. X1 时钟振荡稳定等待时间不包括时钟振荡开始之前的时间（“a” 以下）。



备注 f_x : X1 时钟振荡频率

(4) 振荡稳定时间选择寄存器 (OSTS)

该寄存器用于选择解除 STOP 模式时的 X1 时钟振荡稳定等待时间。

X1 时钟选作为 CPU 时钟时，在 STOP 模式解除后操作会自动地等待使用 OSTS 的计时装置。

内部高速振荡时钟选作为 CPU 时钟时，由 OSTC 确认在 STOP 模式解除后所需的振荡稳定时间已流逝。可以一直检测振荡稳定时间到使用 OSTC 的计时装置。

OSTS 可由 8 位存储器操作指令设置。

产生复位信号会将 OSTS 设为 07H。

图 5-5. 振荡稳定时间选择寄存器（OSTS）的格式

地址：FFFA3H 复位后：07H 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	
				$f_x = 10 \text{ MHz}$	$f_x = 20 \text{ MHz}$
0	0	0	$2^8/f_x$	$25.6 \mu\text{s}$	禁止设置
0	0	1	$2^9/f_x$	$51.2 \mu\text{s}$	$25.6 \mu\text{s}$
0	1	0	$2^{10}/f_x$	$102.4 \mu\text{s}$	$51.2 \mu\text{s}$
0	1	1	$2^{11}/f_x$	$204.8 \mu\text{s}$	$102.4 \mu\text{s}$
1	0	0	$2^{13}/f_x$	$819.2 \mu\text{s}$	$409.6 \mu\text{s}$
1	0	1	$2^{15}/f_x$	3.27 ms	1.64 ms
1	1	0	$2^{17}/f_x$	13.11 ms	6.55 ms
1	1	1	$2^{18}/f_x$	26.21 ms	13.11 ms

注意事项 1. 要在 X1 时钟用作 CPU 时钟时设置 STOP 模式，则需在执行 STOP 指令之前设置 OSTS 寄存器。

2. 禁止将振荡稳定时间设为 $20 \mu\text{s}$ 或小于 $20 \mu\text{s}$ 。

3. 要改变 OSTS 寄存器的设置，务必要确认已完成了 OSTC 寄存器的计数操作。

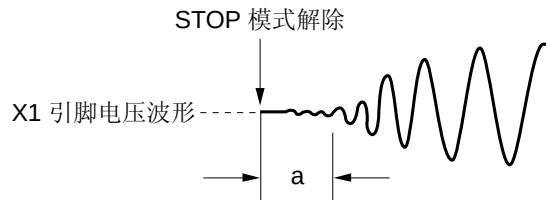
4. 在 X1 时钟振荡稳定时间内不要改变 OSTS 寄存器的值。

5. 振荡稳定时间计数器通过 OSTS 设置为振荡稳定时间向上计数。

在下列情况下，振荡开始后，将 OSTS 的振荡稳定时间设为大于 OSTC 寄存器要检查的计数值的值。

- 如果 X1 时钟在内部高速振荡时钟或副系统时钟被用作 CPU 时钟期间开始振荡。
- 如果在内部高速振荡时钟用作 CPU 时钟且随 X1 时钟振荡期间输入 STOP 模式，然后解除。（因此，要注意，在解除 STOP 模式之后仅将 OSTS 设置振荡稳定时间之前的状态设置至 OSTC。）

6. X1 时钟振荡稳定等待时间不包括时钟振荡开始之前的时间（“a” 以下）。



备注 fx: X1 时钟振荡频率

(5) 系统时钟控制寄存器 (CKC)

该寄存器用于选择 CPU/外围硬件时钟和标度比。

CKC 可由 1 位或 8 位存储器操作指令设置。

复位信号产生后该寄存器被设为 09H。

图 5-6. 系统时钟控制寄存器 (CKC) 的格式

地址: FFFA4H 复位后: 09H R/W^{注1}

符号	<7>	<6>	<5>	<4>	3	2	1	0
CKC	CLS	CSS	MCS	MCM0	1	MDIV2	MDIV1	MDIV0

CLS	CPU/外设硬件时钟 (f _{CLK}) 的状态
0	主系统时钟 (f _{MAIN})
1	副系统时钟 (f _{SUB})

MCS	主系统时钟 (f _{MAIN}) 的状态
0	内部高速振荡时钟 (f _{IH})
1	高速系统时钟 (f _{MX})

CSS	MCM0	MDIV2	MDIV1	MDIV0	CPU/外设硬件时钟 (f _{CLK}) 的选择
0	0	0	0	0	f _{IH}
		0	0	1	f _{IH} /2 (默认)
		0	1	0	f _{IH} /2 ²
		0	1	1	f _{IH} /2 ³
		1	0	0	f _{IH} /2 ⁴
		1	0	1	f _{IH} /2 ⁵
0	1	0	0	0	f _{MX}
		0	0	1	f _{MX} /2
		0	1	0	f _{MX} /2 ²
		0	1	1	f _{MX} /2 ³
		1	0	0	f _{MX} /2 ⁴
		1	0	1	f _{MX} /2 ⁵ ^{注2}
1 ^{注3}	× ^{注3}	×	×	×	f _{SUB} /2
其它					禁止设置

- 注**
1. 位 7 和位 5 为只读位。
 2. 当 f_{MX} < 4 MHz 时禁止设置。
 3. CSS 设为 1 期间禁止改变 MCM0 位的值。

- 备注**
1. f_{IH}: 内部高速振荡时钟频率
f_{MX}: 高速系统时钟频率
f_{SUB}: 副系统时钟频率
 2. ×: 忽略

(注意事项 1 至 3 在下页列出。)

- 注意事项
1. 务必将位 3 设置为 1。
 2. 由 CSS、MCM0 和 MDIV2 至 MDIV0 设置的时钟提供给 CPU 和外设硬件。 所以，如果改变 CPU 时钟，则同时还会改变提供给外设硬件（除实时计数器、时钟输出/蜂鸣器输出和看门狗定时器之外）的时钟。 因此，当改变 CPU/外设硬件时钟时要停止各个外设功能。
 3. 如果将外设硬件时钟用作副系统时钟，则不能保证 A/D 转换器和 IICA 的操作。 有关外设硬件的工作特性，请参见说明各种不同外设硬件的章节以及 第二十八章 电气规范（目标）。

在 78K0R/Kx3-L 中，最快的指令可在 CPU 时钟的一个时钟周期内执行。因此，CPU 时钟（f_{CLK}）和最短指令执行时间之间的关系如表 5-3 中所示。

表 5-3. CPU 时钟和最短指令执行时间之间的关系

CPU 时钟 (由 MDIV2 至 MDIV0 位所设置的 值)	最短指令执行时间: $1/f_{CLK}$			
	主系统时钟 (CSS = 0)			副系统时钟 (CSS = 1)
	高速系统时钟 (MCM0 = 1)		内部高速振荡时钟 (MCM0 = 0)	
	工作于 10 MHz	工作于 20 MHz	工作于 8 MHz (TYP.)	
f _{MAIN}	0.1 μ s	0.05 μ s	0.125 μ s (TYP.)	—
f _{MAIN} /2	0.2 μ s	0.1 μ s	0.25 μ s (TYP.) (默认)	—
f _{MAIN} /2 ²	0.4 μ s	0.2 μ s	0.5 μ s (TYP.)	—
f _{MAIN} /2 ³	0.8 μ s	0.4 μ s	1.0 μ s (TYP.)	—
f _{MAIN} /2 ⁴	1.6 μ s	0.8 μ s	2.0 μ s (TYP.)	—
f _{MAIN} /2 ⁵	3.2 μ s	1.6 μ s	4.0 μ s (TYP.)	—
f _{SUB} /2	—		—	61 μ s

备注 f_{MAIN}: 主系统时钟频率 (f_{IH} 或 f_{MX})

f_{SUB}: 副系统时钟频率

(6) 20 MHz 内部高速振荡控制寄存器（DSCCTL）

该寄存器控制 20 MHz 内部高速振荡时钟（DSC）功能。
它可用于选择是否将 20 MHz 内部高速振荡时钟（f_{IH20}）用为支持 20 MHz 的外设硬件时钟。
DSCCTL 可由 1 位或 8 位存储器操作指令设置。
产生复位信号将该寄存器清除为 00H。

图 5-7. 20 MHz 内部高速振荡控制寄存器（DSCCTL）的格式

地址: F00F6H 复位后: 00H R/W[※]

符号	7	6	5	4	<3>	<2>	1	<0>
DSCCTL	0	0	0	0	DSCS	SELDSC	0	DSCON

DSCS	20 MHz 内部高速振荡提供状态标志
0	不提供
1	提供

SELDSC	CPU/外设硬件时钟（f _{CLK} ）的 20 MHz 内部高速振荡选择
0	不选择 20 MHz 内部高速振荡（为 f _{CLK} 提供 CKC 寄存器所选的时钟）
1	选择 20 MHz 内部高速振荡（为 f _{CLK} 提供 20 MHz 内部高速振荡）

DSCON	20 MHz 内部高速振荡时钟（f _{IH20} ）操作使能/禁止
0	禁止操作。
1	允许操作。

注 位 3 为只读位。

- <R>
- 注意事项
1. 如果 V_{DD} ≥ 2.7 V，则仅能使用 20 MHz 内部振荡。

2. 在 V_{DD} ≥ 2.7 V 情况下设置 DSCON 之后经过 100μs 时，设置 SELDSC。

3. 内部高速振荡器必须在 DSCON = 1 时操作（HIOSTOP = 0）。
- <R>

(7) 外设使能寄存器 0、1、2 (PER0、PER1、PER2)

这些寄存器用于启动或禁止各外设硬件宏功能的使用。供应未使用的硬件的时钟也会停止工作，以便减少功耗和噪声。

PER0、PER1、PER2 可由 1 位或 8 位存储器操作指令设置。

产生复位信号将这些寄存器清除为 00H。

图 5-8. 外设使能寄存器的格式 (1/2)

地址: F00F0H 复位后: 00H R/W

符号	<7>	6	<5>	<4>	3	<2>	1	0
PER0	RTCEN	0	ADCEN	IICAEN ^{#1}	0	SAU0EN	0	0

地址: F00F1H 复位后: 00H R/W

符号	7	6	5	4	<3>	2	1	0
PER1	0	0	0	0	OACMPEN	0	0	0

地址: F00F2H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	<0>
PER2	0	0	0	0	0	0	0	TAU0EN

RTCEN	实时计数器 (RTC) 输入时钟提供 ^{#2} 的控制
0	停止提供输入时钟。 - 不能写入由实时计数器 (RTC) 使用的 SFR (可被读取)。 - 继续实时计数器 (RTC) 的操作。
1	提供输入时钟。 能够读写由实时计数器 (RTC) 使用的 SFR。

ADCEN	A/D 转换器输入时钟提供的控制
0	停止提供输入时钟。 - 不能写入由 A/D 转换器使用的 SFR。 - A/D 转换器处于复位状态。
1	提供输入时钟。 可以读写由 A/D 转换器使用的 SFR。

- 注**
1. 78K0R/KC3-L 的 44 引脚产品未配置此项。
 2. 当从 CPU 访问实时计数器 (RTC) 所用的寄存器时，应使用可由 RTCEN 控制的输入时钟。RTCEN 不能控制提供给 RTC 的工作时钟 (f_{SUB})。

注意事项 务必将 PER0 寄存器的位 0、1、3 和 6 (44 引脚产品为：位 0、1、3、4 和 6)、PER1 寄存器的位 0 至 2 和位 4、以及 PER2 寄存器的位 1 至 7 清为 0。

图 5-8. 外设使能寄存器的格式 (2/2)

地址: F00F0H 复位后: 00H R/W

符号	<7>	6	<5>	<4>	3	<2>	1	0
PER0	RTCEN	0	ADCEN	IICAEN ^注	0	SAU0EN	0	0

地址: F00F1H 复位后: 00H R/W

符号	7	6	5	4	<3>	2	1	0
PER1	0	0	0	0	OACMPEN	0	0	0

地址: F00F2H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	<0>
PER2	0	0	0	0	0	0	0	TAU0EN

IICAEN	串行接口 IICA 输入时钟提供的控制
0	停止提供输入时钟。 • 不能写入串行接口 IICA 所使用的 SFR。 • 串行接口 IICA 处于复位状态。
1	提供输入时钟。 • 可以读写串行接口 IICA 所使用的 SFR。

SAU0EN	串行阵列部件输入时钟提供的控制
0	停止提供输入时钟。 • 不能写入串行阵列部件所使用的 SFR。 • 串行阵列部件处于复位状态。
1	提供输入时钟。 • 可以读写串行阵列部件所使用的 SFR。

OACMPEN	比较器和可编程增益放大器输入时钟提供的控制
0	停止提供输入时钟。 • 不能写入由比较器和可编程增益放大器所使用的 SFR。 • 比较器和可编程增益放大器处于复位状态。
1	提供输入时钟。 • 可以读写比较器和可编程增益放大器所使用的 SFR。

TAU0EN	定时器阵列部件 TAUS 输入时钟提供的控制
0	停止提供输入时钟。 • 不能写入定时器阵列部件 TAUS 所使用的 SFR。 • 定时器阵列部件 TAUS 处于复位状态。
1	提供输入时钟。 • 可以读写定时器阵列部件 TAUS 所使用的 SFR。

注 78K0R/KC3-L 的 44 引脚产品未配置此项。

注意事项 务必将 PER0 寄存器的位 0、1、3 和 6（44 引脚产品为：位 0、1、3、4 和 6）、PER1 寄存器的位 0 至 2 和位 4、以及 PER2 寄存器的位 1 至 7 清为 0。

(8) 运行速度模式控制寄存器 (OSMC)

FLPC 和 FSEL 用于控制高速工作时 flash 存储器的升压电路。

如果微控制器工作在小于 10 MHz 的系统时钟下的低速状态，则通过将该寄存器设置为默认值 00H 可以降低功耗。

此外，当系统时钟工作于 1 MHz 时，可通过将 FLPC 设为 1 更进一步地降低功耗。

RTCLPC 可用于设置副系统时钟 HALT 模式下的操作。

OSMC 可由 8 位存储器操作指令设置。

产生复位信号将该寄存器清除为 00H。

图 5-9. 运行速度模式控制寄存器 (OSMC) 的格式

地址: F00F3H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
OSMC	RTCLPC	0	0	0	0	0	FLPC	FSEL

FLPC	FSEL	f _{CLK} 频率选择
0	0	工作于 10 MHz 或更低的频率（默认）。
0	1	工作于高于 10 MHz 的频率。
1	0	工作于 1 MHz 的频率。
1	1	禁止设置

RTCLPC	副系统时钟 HALT 模式下的设置
0	允许给外设功能提供副系统时钟。 (参见表 18-1 操作被允许的外设的功能。)
1	停止给除实时计数器之外的外设功能提供副系统时钟。

注意事项 1. 在以下两种操作之前将“1”写入 FSEL。

- 将 f_{CLK} 分频至除 f_{IH} 之外的时钟前先改变时钟。
- 运行 DMA 控制器。

2. 当将“1”写入 FSEL 标志时，CPU 等待。

f_{CLK} = f_{IH} 时，等待时间为 15 μs 至 20 μs（目标）；f_{CLK} = f_{IH}/2 时，等待时间为 30 μs 至 40 μs（目标）。

然而，即使在 CPU 等待期间，仍能继续计数 f_x 的振荡稳定时间。

3. 要使 f_{CLK} 提高至 10 MHz 或更高，需将 FSEL 设为“1”，然后在经过两个或两个以上的时钟之后改变 f_{CLK}。

4. 在设置 FSEL = 0 之前确定时钟工作于 10 MHz 或更低的频率。

5. V_{DD} ≤ 2.7 V 期间，设置 FSEL = 0 以切换至 STOP 模式。

(注意事项在下页给出。)

<R>

<R>

6. 使用副系统时钟时，可通过将 **RTCLPC** 设为 **1** 减小 **HALT** 模式的电流。但是，在副系统时钟 **HALT** 期间，不能为除实时计数器之外的外设功能提供时钟。在设置副系统时钟 **HALT** 模式之前，将 **PER0** 寄存器的位 **7** (**RTCEN**) 设为 **1**，并将 **PER0** 的位 **0** 至 **6**、**PER2** 的位 **0** 至 **7** 的所有位设为 **0**。
7. 一旦将 **FLPC** 从 **0** 设为 **1**，则除非复位禁止将其从 **1** 设回 **0**。

5.4 系统时钟振荡器

5.4.1 X1 振荡器

X1 振荡器使用连接至 X1 和 X2 引脚的晶体谐振器或陶瓷谐振器（2 至 20 MHz）振荡。

也可以输入外部时钟。 这种情况下，将时钟信号输入至 EXCLK 引脚。

要使用 X1 振荡器，需按照以下设置时钟工作模式控制寄存器（CMC）的位 7 和 6（EXCLK、OSCSEL）。

- 晶体或陶瓷振荡：EXCLK, OSCSEL = 0, 1
- 外部时钟输入：EXCLK, OSCSEL = 1, 1

当不使用 X1 振荡器时，设置输入端口模式（EXCLK, OSCSEL = 0, 0）。

当引脚也不用作输入端口引脚时，请参见 表 2-2 未使用引脚的连接。

图 5-10 显示了 X1 振荡器外部电路的示例。

图 5-10. X1 振荡器外部电路的示例



注意事项在下一页中列出。

5.4.2 XT1 振荡器

XT1 振荡器使用连接至 XT1 和 XT2 引脚的晶体谐振器（标准：32.768 kHz）振荡。

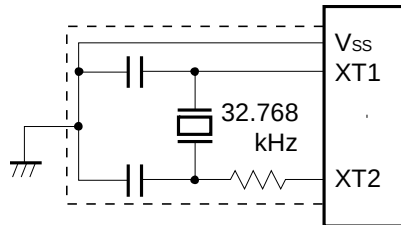
要使用 XT1 振荡器，需将时钟工作模式控制寄存器（CMC）的位 4（OSCSELS）设为 1。

当不使用 XT1 振荡器时，设置输入端口模式（OSCSELS = 0）。

当引脚也不用作输入端口引脚时，请参见 表 2-2 未使用引脚的连接。

图 5-11 显示了 XT1 振荡器外部电路的示例。

图 5-11. XT1 振荡器（晶体振荡）外部电路的示例



注意事项在下一页中列出。

注意事项 1. 当使用 X1 振荡器和 XT1 振荡器时，为了避免连接电容引起的不利影响，应如下连接图 5-10 和 5-11 中虚线包围的区域。

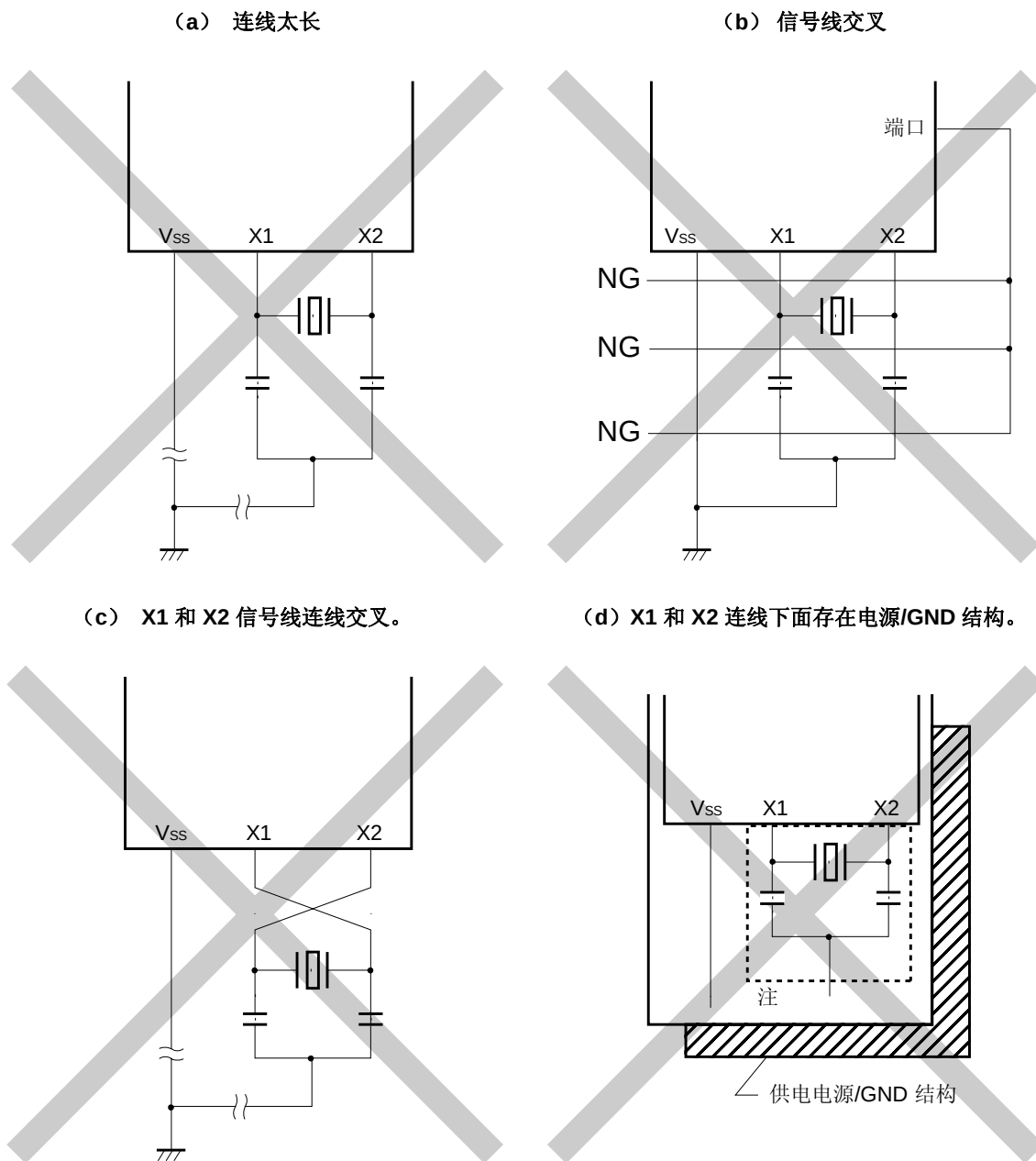
- 保证连线尽可能最短。
- 不要将该连线和其它信号线交叉。布线时不要将该连线靠近通过大波动电流的信号线。
- 总是使振荡器电容的接地点与 V_{SS} 的电位相同。不要将电容连接到一个大电流通过的接地点
- 不要从振荡器获取信号

XT1 振荡器是低放大倍数来达到低功耗的电路。设计电路时需注意以下几点。

- 引脚和电路包括附加电容。因此，要用实际使用的电路板进行振荡评估并确认不存在问题。
- 当使用超低功耗振荡（AMPHS1 = 1）作为 XT1 的振荡模式时，要使用第二十八章 电气规范（目标）中说明的推荐的谐振器。
- 使 XT1 和 XT2 引脚与谐振器之间的连线尽可能最短，且附加电容和接线电阻最小。选择超低功耗振荡（AMPHS1 = 1）时尤其要注意该点。
- 使用线电阻很小的材料配置电路板的电路。
- 在 XT1 振荡器附近，尽可能多地放置与 V_{SS} 电压相同的接地点。
- 确保 XT1 和 XT2 引脚与谐振器之间的连线不与其他信号线交叉。布线时不要将该连线靠近通过大波动电流的信号线。
- 在电路板处于高湿度环境下或露水凝缩时，由于电路板吸湿，XT1 和 XT2 引脚之间的阻抗可能会变小且振荡也可能失常。当在这样的环境下使用电路板时，应采取措施，诸如涂层来防止电路板受潮。
- 给电路板涂层时，应使用不会引起 XT1 和 XT2 引脚电容或泄漏的材料。

图 5-12 显示了不正确的谐振器连接的示例。

图 5-12. 不正确的谐振器连接的示例 (1/2)

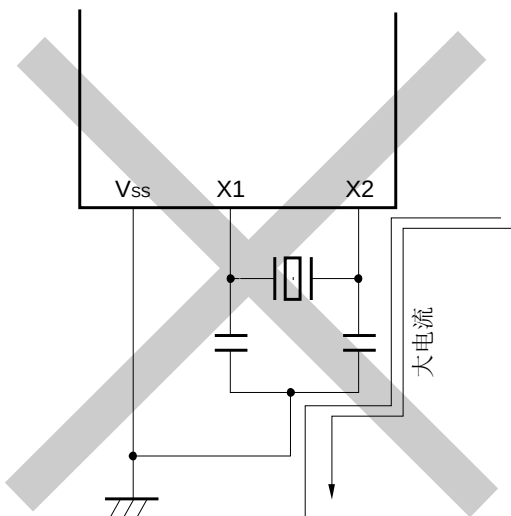
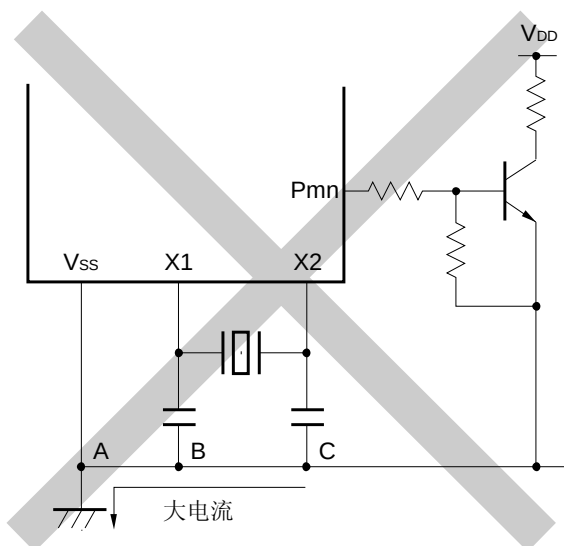


注 不要将电压/GND 结构置于 X1 和 X2 引脚的连线部分（图中虚线专门括起的部分）的下方，以及多层板或双面板当中谐振器的下方。
不要配置会导致电容元件且影响振荡特性的线路图。

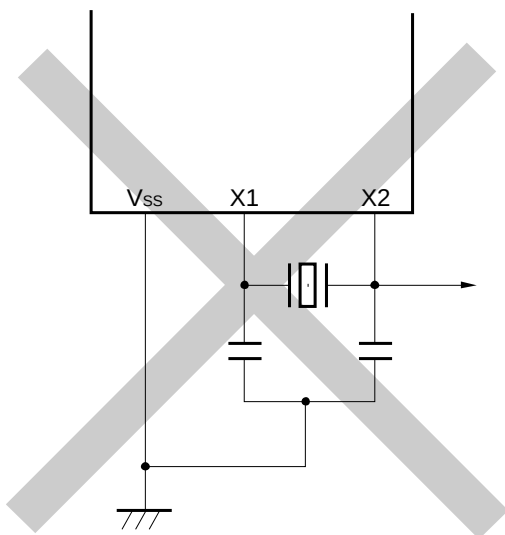
备注 使用副系统时钟时，分别使用 XT1 和 XT2 代替 X1 和 X2。另外，在 XT2 侧连续地插入电阻。

图 5-12. 不正确的谐振器连接的示例 (2/2)

(e) 大交变电流附近的连线

(f) 电流流经振荡器的接地线
(点 A、B 和 C 处的波动电压)

(g) 获取信号



备注 使用副系统时钟时，分别使用 XT1 和 XT2 代替 X1 和 X2。另外，在 XT2 侧连续地插入电阻。

注意事项 2. 当 X2 和 XT1 并联时，由于误动作可能会导致 X2 的串扰噪声随 XT1 增加。

5.4.3 内部高速振荡器[‡]

78K0R/Kx3-L (1, 8 MHz (TYP.)) 中包含内部高速振荡器。振荡可由时钟工作状态控制寄存器 (CSC) 的位 0 (HIOSTOP) 控制。

5.4.4 20 MHz内部高速振荡器[‡]

78K0R/Kx3-L (20 MHz (TYP.)) 中包含 20 MHz 内部高速振荡器。振荡可通过 20 MHz 内部高速振荡控制寄存器 (DSCCTL) 位 0 (DSCON) 在 $V_{DD} \geq 2.7\text{ V}$ 下控制。

复位解除之后, 通过将 DSCCTL 寄存器的位 0 (DSCON) 设为 1 使 20 MHz 内部高速振荡器开始振荡。

注 要使用 1、8 或 20 MHz 内部高速振荡时钟, 需预先使用选项字节来设置频率 (有关详情, 请参见 **第二十三章 选项字节**)。另外, 在复位解除后内部高速振荡器自动开始振荡。要使用 20 MHz 内部高速振荡操作微控制器, 需通过将 20 MHz 内部高速振荡控制寄存器 (DSCCTL) 的位 0 (DSCON) 设为 1 在 $V_{DD} \geq 2.7\text{ V}$ 下开始振荡。

5.4.5 内部低速振荡器

78K0R/Kx3-L 中包含内部低速振荡器。

内部低速振荡时钟仅用作看门狗定时器时钟。内部低速振荡时钟不能用作 CPU 时钟。

复位解除之后, 内部低速振荡器自动开始振荡, 且如果通过选项字节允许看门狗定时器的操作, 会驱动看门狗定时器 (30 kHz (TYP.))。

除了看门狗定时器停止的情况, 内部低速振荡器继续振荡。看门狗定时器工作时, 即使在程序循环的情况下内部低速时钟也不会停止。

5.4.6 预分频器

预分频器通过分频主系统时钟和副系统时钟产生 CPU/外设硬件时钟。

5.5 时钟发生器的操作

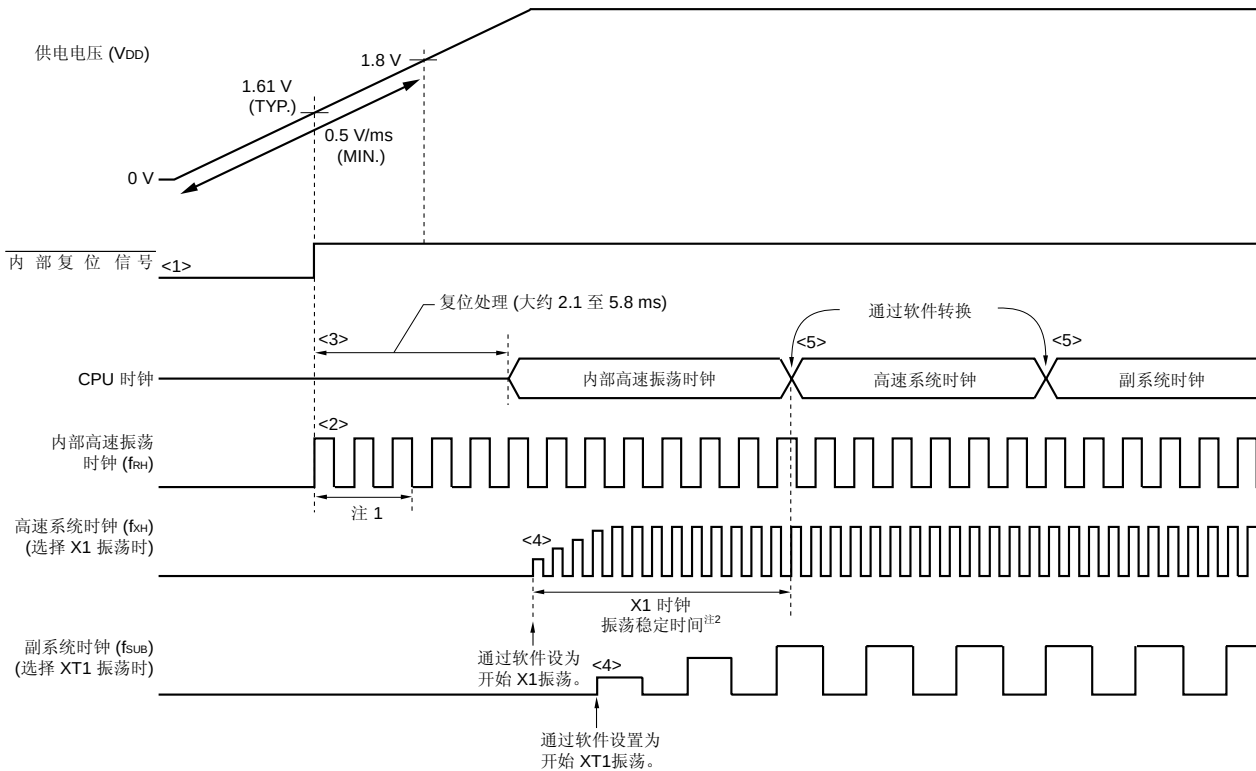
时钟发生器产生以下时钟并控制 CPU 的工作模式，例如待机模式（参见图 5-1）。

- 主系统时钟 f_{MAIN}
 - 高速系统时钟 f_{MX}
 - X1 时钟 f_{x}
 - 外部主系统时钟 f_{EX}
 - 内部高速振荡时钟 f_{IH}
 - 20 MHz 内部高速振荡时钟 f_{IH20}
- 副系统时钟 f_{SUB}
- 内部低速振荡时钟 f_{IL}
- CPU/外设硬件时钟 f_{CLK}

在 78K0R/Kx3-L 中，CPU 在复位解除后内部高速振荡器开始输出时开始工作。

图 5-13 至 图 5-16 中显示了供电电压开启时时钟发生器的操作。

图 5-13. 供电电压开启时时钟发生器的操作
(设置了停止 LVI 默认开始功能(选项字节: LVIOFF = 1) 时)



- <1> 打开电源时，通过加电清除（POC）电路产生内部复位信号。
- <2> 当供电电压超过 1.61 V（TYP.）时，复位解除且内部高速振荡器自动开始振荡。
- <3> 当供电电压以 0.5 V/ms（MIN.）的斜率上升时，CPU 在执行，诸如在复位解除后等待供电电源或稳压器的电压稳定的复位处理之后基于内部高速振荡时钟开始工作。
- <4> 通过软件设置 X1 或 XT1 时钟振荡的开始（参见 5.6.1 控制高速系统时钟的示例 中的（1）以及 5.6.3 控制副系统时钟的示例 中的（1））。
- <5> 当 CPU 时钟切换至 X1 或 XT1 时钟时，待时钟振荡稳定然后再通过软件设置切换（参见 5.6.1 控制高速系统时钟的示例 中的（3）以及 5.6.3 控制副系统时钟的示例 中的（3））。

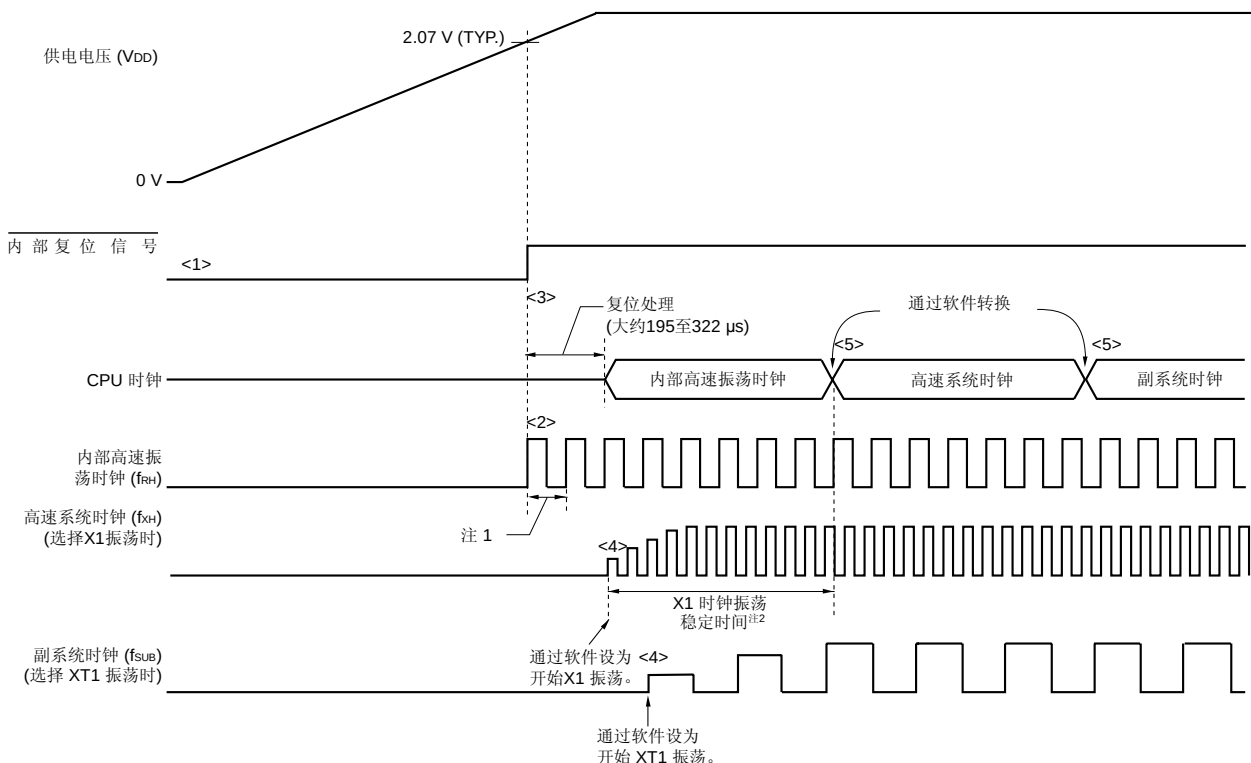
注 1. 内部复位处理时间包括内部高速振荡时钟的振荡精确稳定时间。

2. CPU 基于内部高速振荡时钟工作期间解除复位（上图）或解除 STOP 模式时，使用振荡稳定时间计数器状态寄存器（OSTC）确认 X1 时钟的振荡稳定时间。如果 CPU 基于高速系统时钟（X1 振荡）工作，则在使用振荡稳定时间选择寄存器（OSTS）解除 STOP 模式时设置振荡稳定时间。

- 注意事项**
1. 如果电压从供电电源开始以小于 0.5 V/ms (MIN.) 的斜率上升至 1.8 V , 则在电压达到 1.8 V 之前从供电电源输入一个低电平至 $\overline{\text{RESET}}$ 引脚, 或使用选项字节将 LVI 默认开始功能设为停止 ($\text{LVIOFF} = 0$) (参见图 5-14)。通过这样, CPU 按照与<2>相同的时序工作, 且此后按照图 5-13 中通过 $\overline{\text{RESET}}$ 引脚解除复位后的时序工作。
 2. 当使用通过 EXCLK 引脚输入的外部时钟时, 不必等待振荡稳定时间。

备注 微控制器运行期间, 可通过软件设置停止不用作 CPU 时钟的时钟。可通过执行 STOP 指令停止内部高速振荡时钟和高速系统时钟 (参见 5.6.1 控制高速系统时钟的示例 中的 (4)、5.6.2 控制内部高速振荡时钟的示例 中的 (3) 以及 5.6.3 控制副系统时钟的示例 中的 (3))。

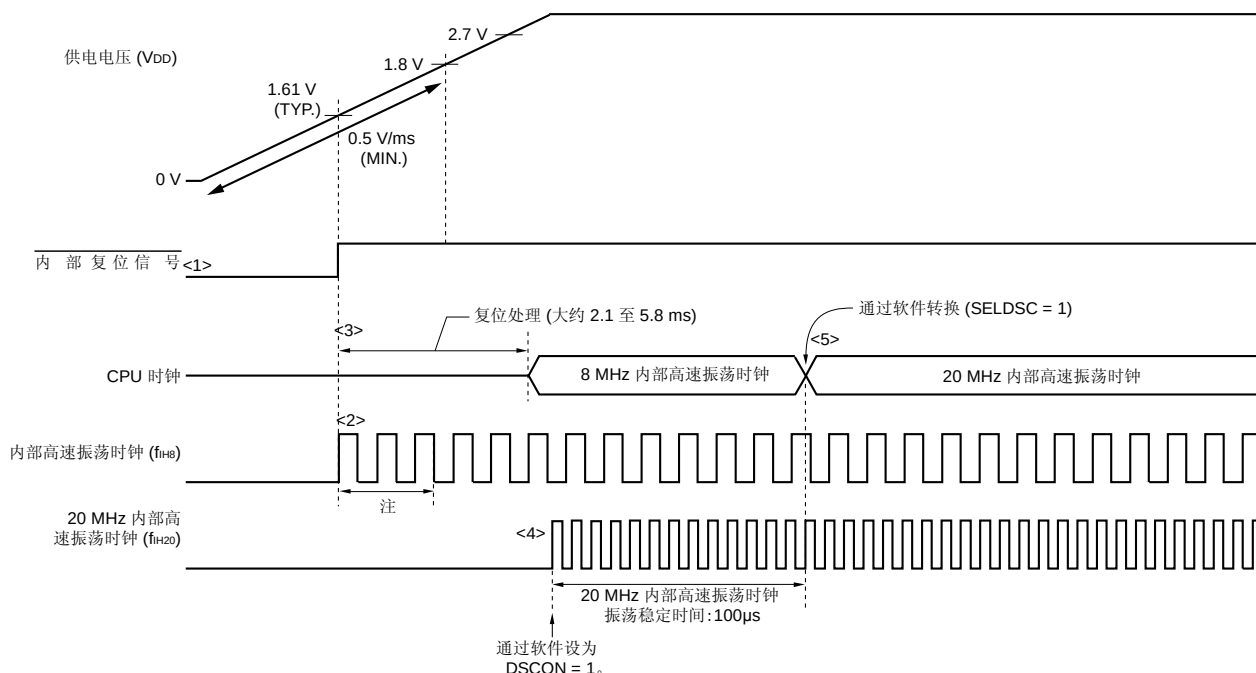
图 5-14. 供电电压开启时时钟发生器的操作
(设置了使能 LVI 默认开始功能 (选项字节: $\text{LVIOFF} = 0$) 时)



- <1> 打开电源时, 通过加电清除 (POC) 电路产生内部复位信号。
- <2> 当供电电压超过 2.07 V (TYP.) 时, 复位解除且内部高速振荡器自动开始振荡。
- <3> 复位解除且执行复位处理之后, CPU 基于内部高速振荡时钟开始工作。
- <4> 通过软件设置 X1 或 XT1 时钟振荡的开始 (参见 5.6.1 控制高速系统时钟的示例 中的 (1) 以及 5.6.3 控制副系统时钟的示例 中的 (1))。
- <5> 当 CPU 时钟切换至 X1 或 XT1 时钟时, 待时钟振荡稳定然后再通过软件设置切换 (参见 5.6.1 控制高速系统时钟的示例 中的 (3) 以及 5.6.3 控制副系统时钟的示例 中的 (3))。

- 注**
1. 内部复位处理时间包括内部高速振荡时钟的振荡精确稳定时间。
 2. CPU 基于内部高速振荡时钟工作期间解除复位（上图）或解除 STOP 模式时，使用振荡稳定时间计数器状态寄存器（OSTC）确认 X1 时钟的振荡稳定时间。如果 CPU 基于高速系统时钟（X1 振荡）工作，则在使用振荡稳定时间选择寄存器（OSTS）解除 STOP 模式时设置振荡稳定时间。
- 注意事项**
1. 供电电源达到 1.61 V（TYP.）后需要一段电压稳定时间（大约为 2.0 至 5.8 ms）。如果供电电压从 1.61 V（TYP.）升至 2.07 V（TYP.）的时间比电压稳定时间短，则电压稳定时间过后会进入复位处理。
 2. 当使用通过 EXCLK 引脚输入的外部时钟时，不必等待振荡稳定时间。
- 备注**
- 微控制器运行期间，可通过软件设置停止不用作 CPU 时钟的时钟。可通过执行 STOP 指令停止内部高速振荡时钟和高速系统时钟（参见 5.6.1 控制高速系统时钟的示例 中的（4）、5.6.2 控制内部高速振荡时钟的示例 中的（3）以及 5.6.3 控制副系统时钟的示例 中的（3））。

图 5-15. 供电电压开启时时钟发生器的操作
(设置了停止 LVI 默认开始功能时 (选项字节: LVIOFF = 1) 且变为 20 MHz 内部高速振荡时钟)

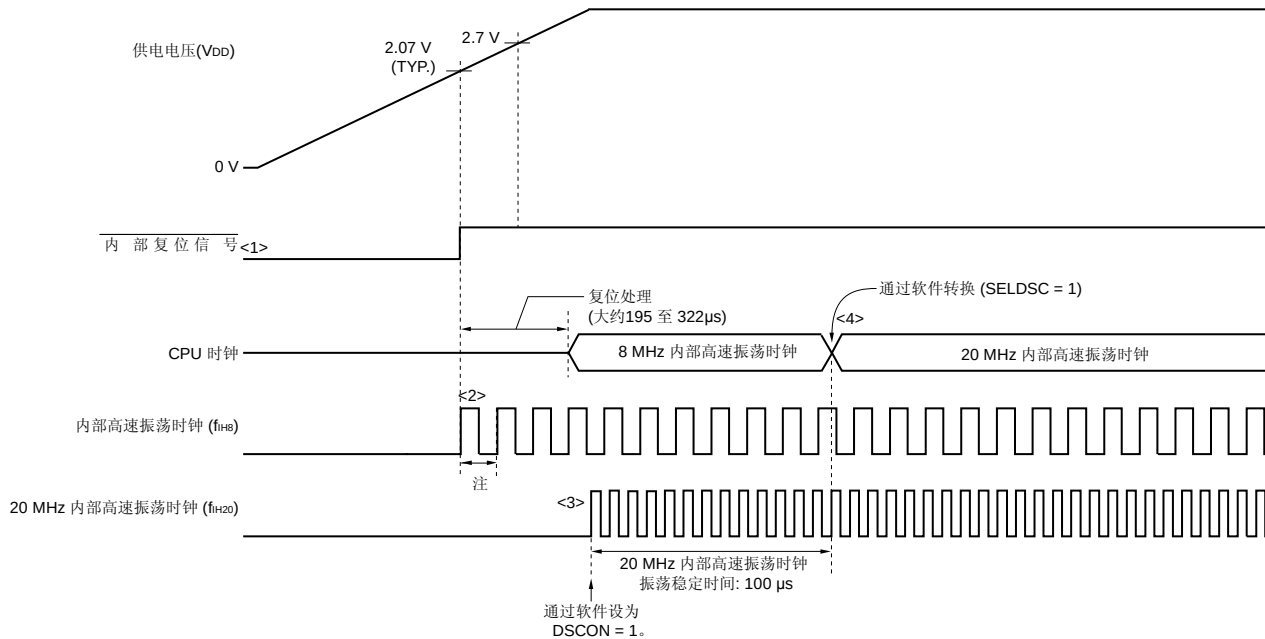


- <1> 打开电源时，通过加电清除（POC）电路产生内部复位信号。
- <2> 当供电电压超过 1.61 V（TYP.）时，复位解除且内部高速振荡器自动开始振荡。
- <3> 当供电电压以 0.5 V/ms（MIN.）的斜率上升时，CPU 在执行，诸如在复位解除后等待供电电源或稳压器的电压稳定的复位处理之后基于内部高速振荡时钟开始工作。
- <4> 检查供电电源为 2.7 V 或更高并通过软件设置 DSCON = 1。
- <5> 等待 100 μs 之后，通过软件设置 SELDSC = 1 来切换时钟。

注 内部复位处理时间包括内部高速振荡时钟的振荡精确稳定时间。

- 注意事项
1. 要使用 20 MHz 内部高速振荡时钟，需预先使用选项字节（000C1H）的位 2 和位 1（FRQSEL2 和 FRQSEL1）将频率设为 20 MHz（有关详情，请参见第二十三章 选项字节）。
 2. 如果电压从供电电源开始以小于 0.5 V/ms（MIN.）的斜率上升至 1.8 V，则在电压达到 1.8 V 之前从供电电源输入一个低电平至 RESET 引脚，或使用选项字节将 LVI 默认开始功能设为停止（LVIOFF = 0）（参见图 5-16）。这样一来，CPU 按照与<2>相同的时序工作，且此后按照图 5-15 中通过 RESET 引脚解除复位后的时序工作。

图 5-16. 供电电压开启时时钟发生器的操作
(设置了使能 LVI 默认开始功能 (选项字节: LVIOFF = 0) 且变为 20 MHz 内部高速振荡时钟时)



- <1> 打开电源时, 通过加电清除 (POC) 电路产生内部复位信号。
- <2> 当供电电压超过 2.07 V (TYP.) 时, 复位解除且内部高速振荡器自动开始振荡。
- <3> 复位解除且执行复位处理之后, CPU 基于内部高速振荡时钟开始工作。
- <4> 检查供电电源为 2.7 V 或更高并通过软件设置 DSCON = 1。
- <5> 等待 100 μs 之后, 通过软件设置 SELDSC = 1 来切换时钟。

注 内部复位处理时间包括内部高速振荡时钟的振荡精确稳定时间。

- 注意事项
1. 要使用 20 MHz 内部高速振荡时钟, 需预先使用选项字节 (000C1H) 的位 2 和位 1 (FRQSEL2 和 FRQSEL1) 将频率设为 20 MHz (有关详情, 请参见 第二十三章 选项字节)。
 2. 供电电源达到 1.61 V (TYP.) 后需要一段电压稳定时间 (大约为 2.0 至 5.8 ms)。如果供电电压从 1.61 V (TYP.) 升至 2.07 V (TYP.) 的时间比电压稳定时间短, 则电压稳定时间过后会进入复位处理。

5.6 控制时钟

5.6.1 控制高速系统时钟的示例

高速系统时钟的以下两种类型可用。

- X1 时钟：将晶体/陶瓷谐振器连接至 X1 和 X2 引脚。
- 外部主系统时钟：外部时钟被输入至 EXCLK 引脚。

当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 复位解除之后，X1/P121 和 X2/EXCLK/P122 引脚处于输入端口模式。

下面说明以下情况下设置步骤的示例。

- (1) 振荡 X1 时钟时
- (2) 使用外部主系统时钟时
- (3) 使用高速系统时钟作为 CPU/外设硬件时钟时
- (4) 停止高速系统时钟时

(1) 振荡 X1 时钟时设置步骤的举例

<1> 设置 P121/X1 和 P122/X2/EXCLK 引脚并设置振荡频率（CMC 寄存器）

- $2\text{ MHz} \leq f_x \leq 10\text{ MHz}$

EXCLK	OSCSEL	0	OSCSELS	0	AMPHS1	AMPHS0	AMPH
0	1	0	0/1	0	0/1	0/1	0

- $10\text{ MHz} < f_x \leq 20\text{ MHz}$

EXCLK	OSCSEL	0	OSCSELS	0	AMPHS1	AMPHS0	AMPH
0	1	0	0/1	0	0/1	0/1	1

- 备注**
1. f_x : X1 时钟振荡频率
 2. 有关 P123/XT1 和 P124/XT2 引脚的设置,参见 5.6.3 控制副系统时钟的示例。

<2> 控制 X1 时钟的振荡（CSC 寄存器）

如果 MSTOP 被清为 0，则 X1 振荡器开始振荡。

<3> 等待 X1 时钟振荡的稳定

检查 OSTC 寄存器并等待必要的时间。

等待期间，可使用内部高速振荡时钟执行软件处理。

注意事项 1. 复位之后，CMC 仅可使用 8 位存储器操作指令写入一次。

因此，还有必要同时设置 OSCSELS 位的值。 有关 OSCSELS 位，请参见 5.6.3 控制副系统时钟的示例。

2. 供电电压达到要使用的时钟的可用电压之后，设置 X1 时钟（参见 第二十八章 电气规范（目标））。

(2) 使用外部主系统时钟时设置步骤的示例

<1> 设置 P121/X1 和 P122/X2/EXCLK 引脚（CMC 寄存器）

EXCLK	OSCSEL	0	OSCSELS	0	AMPHS1	AMPHS0	AMPH
1	1	0	0/1	0	0/1	0/1	0/1

备注 有关 P123/XT1 和 P124/XT2 引脚的设置,参见 5.6.3 (1) 振荡副系统时钟时设置步骤的示例。

<2> 控制外部主系统时钟输入（CSC 寄存器）

当 MSTOP 被清为 0 时，允许外部主系统时钟的输入。

注意事项 1. 复位之后，CMC 仅可使用 8 位存储器操作指令写入一次。

因此，还有必要同时设置 OSCSELS 位的值。 有关 OSCSELS 位，请参见 5.6.3 控制副系统时钟的示例。

2. 供电电压达到要使用的时钟的可用电压之后，设置外部主系统时钟（参见 第二十八章 电气规范（目标））。

(3) 使用高速系统时钟作为 CPU/外设硬件时钟时设置步骤的示例

<1> 设置高速系统时钟振荡[‡]

（参见 5.6.1 (1) 振荡 X1 时钟时设置步骤的示例 和 (2) 使用外部主系统时钟时设置步骤的示例。）

注 高速系统时钟已经工作时，<1>的设置不是必须的。

<2> 将高速系统时钟设为 CPU/外设硬件时钟的源时钟并设定设置时钟（CKC 寄存器）的分频系数

MCM0	MDIV2	MDIV1	MDIV0	CPU/外设硬件时钟（f _{CLK} ）的选择
1	0	0	0	f _{MX}
	0	0	1	f _{MX} /2
	0	1	0	f _{MX} /2 ²
	0	1	1	f _{MX} /2 ³
	1	0	0	f _{MX} /2 ⁴
	1	0	1	f _{MX} /2 ⁵ [‡]

注 当 f_{MX} < 4 MHz 时禁止设置。

<3> 如果不使用某些外设硬件宏，则可停止提供各个硬件宏的输入时钟。
(PER0 寄存器)

RTCEN	0	ADCEN	IICAEN ^注	0	SAU0EN	0	0

(PER1 寄存器)

0	0	0	0	OACMPEN	0	0	0

(PER2 寄存器)

0	0	0	0	0	0	0	TAU0EN

xxxEN	输入时钟控制
0	停止提供输入时钟。
1	提供输入时钟。

注 78K0R/KC3-L 的 44 引脚产品未配置此项。

注意事项 务必将以下位清除为 0。

- PER0 寄存器的位 0、1、3 和 6
(对于 78K0R/KC3-L 的 44 引脚产品，则清除位 0、1、3、4 和 6)
- PER1 寄存器的位 0 至 2 和位 4 至 7
- PER2 寄存器的位 1 至 7

备注 RTCEN: 实时计数器输入时钟的控制
ADCEN: A/D 转换器输入时钟的控制
IICAEN: 串行接口 IICA 输入时钟的控制
SAU0EN: 串行阵列部件输入时钟的控制
OACMPEN: 可编程增益放大器输入时钟的控制
TAU0EN: 定时器阵列部件 TAUS 输入时钟的控制

(4) 停止高速系统时钟时设置步骤的示例

可按照以下两种方式停止高速系统时钟（如果使用外部时钟则禁止时钟输入）。

- 执行 STOP 指令
- 将 MSTOP 设为 1

(a) 执行 STOP 指令

- <1> 设置为停止外设硬件
停止不能用于 STOP 模式的外设硬件（有关不能用于 STOP 模式的外设硬件，参见 第十八章 待机功能）。
- <2> 解除 STOP 模式之后设置 X1 时钟振荡稳定时间
如果 X1 时钟在进入 STOP 模式之前振荡，则在执行 STOP 指令前设置 OSTS 寄存器的值。
- <3> 执行 STOP 指令
当执行 STOP 指令时，系统处于 STOP 模式且 X1 振荡停止（禁止外部时钟的输入）。

(b) 通过将 MSTOP 设为 1 停止 X1 振荡（禁止外部时钟输入）**<1> 确认 CPU 时钟状态（CKC 寄存器）**

使用 CLS 和 MCS 确认 CPU 基于除高速系统时钟之外的时钟工作。

当 CLS = 0 且 MCS = 1 时，高速系统时钟被提供给 CPU，所以应将 CPU 时钟变为副系统时钟或内部高速振荡时钟。

CLS	MCS	CPU 时钟状态
0	0	内部高速振荡时钟或 20 MHz 内部高速振荡时钟
0	1	高速系统时钟
1	×	副系统时钟

<2> X1 时钟振荡重新开始后 X1 振荡稳定时间的设置[※]

重新开始 X1 时钟振荡之后，先将“1”设置至 MSTOP，再将 OSTC 寄存器设为大于计数值的值，该值使用 OSTC 寄存器确认。

<3> 停止高速系统时钟（CSC 寄存器）

当将 MSTOP 设为 1 时，停止 X1 振荡（禁止外部时钟的输入）。

注 当高速系统时钟处于 X1 振荡模式时，需要该设置恢复 X1 时钟振荡。
外部时钟输入模式下不需要该设置。

注意事项 当 MSTOP 设为 1 时，务必保证 MCS = 0 或 CLS = 1。此外，停止基于高速系统时钟工作的外设硬件。

5.6.2 控制内部高速振荡时钟的示例

下面说明以下情况下时钟设置步骤的示例。

- (1) 内部高速振荡时钟重新开始振荡时
- (2) 使用高速振荡时钟作为 CPU/外设硬件时钟时
- (3) 停止内部高速振荡时钟时

(1) 重新开始内部高速振荡时钟的振荡时设置步骤的示例[※]**<1> 设置重新开始内部高速振荡时钟的振荡（CSC 寄存器）**

当 HIOSTOP 被清为 0 时，内部高速振荡时钟重新开始振荡。

注 复位解除后，内部高速振荡器自动开始振荡且选择内部高速振荡时钟作为 CPU/外设硬件时钟。

(2) 使用内部高速振荡时钟作为 CPU/外设硬件时钟时设置步骤的示例**<1> 重新开始内部高速振荡时钟的振荡[※]**

（参见 5.6.2 (1) 重新开始内部高速振荡时钟时设置步骤的示例）。

注 内部高速振荡时钟工作时，<1>的设置不是必须的。

<2> 将内部高速振荡时钟设为 CPU/外设硬件时钟的源时钟并设定设置时钟（CKC 寄存器）的分频系数

MCM0	MDIV2	MDIV1	MDIV0	CPU/外设硬件时钟（f _{CLK} ）的选择
0	0	0	0	f _{IH}
	0	0	1	f _{IH} /2
	0	1	0	f _{IH} /2 ²
	0	1	1	f _{IH} /2 ³
	1	0	0	f _{IH} /2 ⁴
	1	0	1	f _{IH} /2 ⁵

注意事项 如果要在重新开始内部高速振荡时钟之后将 CPU/外设硬件时钟从高速系统时钟切换为内部高速振荡时钟，则应经过 10 μs 或更长的时间后再进行切换。

如果在重新开始内部高速振荡时钟之后立即进行切换，则内部高速振荡的精度不能保证为 10 μs。

(3) 停止内部高速振荡时钟时设置步骤的示例

内部高速振荡时钟不能以以下两种方式停止。

- 执行 STOP 指令
- 将 HIOSTOP 设为 1

(a) 执行 STOP 指令

<1> 外设硬件的设置

停止不能用于 STOP 模式的外设硬件（有关不能用于 STOP 模式的外设硬件，参见 第十八章 待机功能）。

<2> 解除 STOP 模式之后设置 X1 时钟振荡稳定时间

如果 X1 时钟在进入 STOP 模式之前振荡，则在执行 STOP 指令前设置 OSTC 寄存器的值。

<3> 执行 STOP 指令

当执行 STOP 指令时，系统处于 STOP 模式并停止内部高速振荡时钟。

(b) 通过将 HIOSTOP 设为 1 停止内部高速振荡时钟

<1> 确认 CPU 时钟状态（CKC 寄存器）

使用 CLS 和 MCS 确认 CPU 基于除内部高速振荡时钟之外的时钟工作。

当 CLS = 0 且 MCS = 0 时，内部高速振荡时钟被提供给 CPU，所以应将 CPU 时钟变为高速系统时钟或副系统时钟。

CLS	MCS	CPU 时钟状态
0	0	内部高速振荡时钟或 20 MHz 内部高速振荡时钟
0	1	高速系统时钟
1	×	副系统时钟

<2> 停止内部高速振荡时钟（CSC 寄存器）

当 HIOSTOP 被设为 1 时，停止内部高速振荡时钟。

注意事项 当 HIOSTOP 设为 1 时，务必保证 MCS = 1 或 CLS = 1。此外，停止基于内部高速振荡时钟工作的外设硬件。

5.6.3 控制副系统时钟的示例

可通过将晶体振荡器连接至 XT1 和 XT2 引脚使副系统时钟振荡。

当不使用副系统时钟时，XT1/P123 和 XT2/P124 引脚可用作输入端口引脚。

注意事项 复位解除之后，XT1/P123 和 XT2/P124 引脚处于输入端口模式。

下面说明以下情况下设置步骤的示例。

- (1) 振荡副系统时钟时
- (2) 使用副系统时钟作为 CPU 时钟时
- (3) 停止副系统时钟时

注意事项 当副系统时钟用作 CPU 时钟时，副系统时钟还可以提供给外设硬件（除实时计数器、时钟输出/蜂鸣器输出和看门狗定时器之外）。此时，不能保证 A/D 转换器和 IICA 的操作。有关外设硬件的工作特性，请参见说明各种不同外设硬件的章节以及第二十八章 电气规范（目标）。

(1) 振荡副系统时钟时设置步骤的举例

<1> 设置 P123/XT1 和 P124/XT2 引脚（CMC 寄存器）

EXCLK	OSCSEL	0	OSCSELS	0	AMPHS1	AMPHS0	AMPH
0/1	0/1	0	1	0	0/1	0/1	0/1

备注 有关 P121/X1 和 P122/X2 引脚的设置,参见 5.6.1 控制高速系统时钟的示例。

<2> 控制副系统时钟的振荡（CSC 寄存器）

如果 XTSTOP 被清为 0，则 XT1 振荡器开始振荡。

<3> 等待副系统时钟振荡的稳定

使用定时器功能通过软件等待副系统时钟的振荡稳定时间。

注意事项 复位之后，CMC 寄存器仅可使用 8-位存储器操作指令写入一次。

因此，还有必要同时设置 EXCLK 和 OSCSEL 位的值。有关 EXCLK 和 OSCSEL 位，参见 5.6.1

(1) 振荡 X1 时钟时设置步骤的示例 或 5.6.1 (2) 使用外部主系统时钟时设置步骤的示例。

(2) 用副系统时钟作为 CPU 时钟时设置步骤的示例

<1> 设置副系统时钟振荡^注
(参见 5.6.3 (1) 振荡副系统时钟时设置步骤的示例。)

注 副系统时钟运行期间，<1>的设置不是必须的。

<2> 将副系统时钟设为 CPU 时钟的源时钟 (CKC 寄存器)

CSS	CPU/外设硬件时钟 (f _{CLK}) 的选择
1	f _{SUB} /2

注意事项 当副系统时钟用作 CPU 时钟时，副系统时钟还可以提供给外设硬件（除实时计数器、时钟输出/蜂鸣器输出和看门狗定时器之外）。此时，不能保证 A/D 转换器和 IICA 的操作。有关外设硬件的工作特性，请参见说明各种不同外设硬件的章节以及第二十八章 电气规范（目标）。

(3) 停止副系统时钟时设置步骤的示例

<1> 确认 CPU 时钟状态 (CKC 寄存器)
使用 CLS 和 MCS 确认 CPU 基于除副系统时钟之外的时钟工作。
当 CLS = 1 时，副系统时钟被提供给 CPU，所以应将 CPU 时钟变为内部高速振荡时钟或高速系统时钟。

CLS	MCS	CPU 时钟状态
0	0	内部高速振荡时钟或 20 MHz 内部高速振荡时钟
0	1	高速系统时钟
1	×	副系统时钟

<2> 停止副系统时钟 (CSC 寄存器)
当 XTSTOP 设为 1 时，停止副系统时钟。

注意事项 1. 当将 XTSTOP 设为 1 时，务必保证 CLS = 0。此外，如果外设硬件基于副系统时钟工作，则停止外设硬件。
2. 副系统时钟振荡不能使用 STOP 指令停止。

5.6.4 控制内部低速振荡时钟的示例

内部低速振荡时钟不能用作 CPU 时钟。 仅能用作看门狗定时器时钟。

复位解除之后，内部低速振荡器自动开始振荡，且如果通过选项字节允许看门狗定时器的操作，会驱动看门狗定时器（30 kHz（TYP.））。

除了看门狗定时器停止的情况，内部低速振荡器继续振荡。 看门狗定时器工作时，即使在程序循环情况下内部低速时钟也不会停止。

(1) 停止内部低速振荡时钟时设置步骤的示例

内部低速振荡时钟不能以以下两种方式停止。

- 在 HALT/STOP 模式下，通过选项字节（000C0H 的位 0（WDSTBYON）= 0）以及执行 HALT 或 STOP 指令停止看门狗定时器。
- 通过选项字节（000C0H 的位 4（WDTON）= 0）停止看门狗定时器。

(2) 重新开始内部低速振荡时钟的振荡时设置步骤的示例

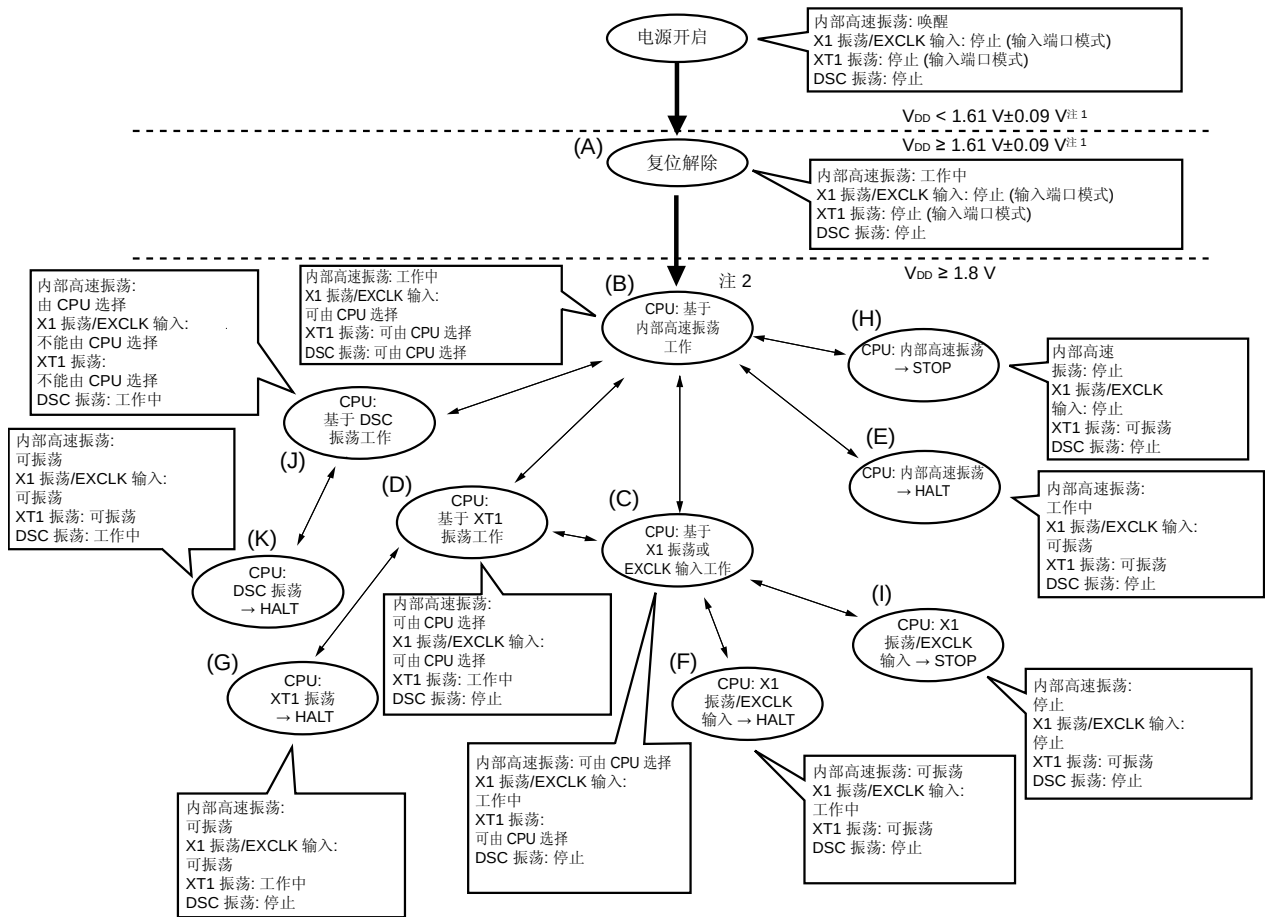
可按照以下重新开始内部低速振荡时钟。

- 解除 HALT 或 STOP 模式
（仅当在 HALT/STOP 模式下通过选项字节（000C0H 的位 0（WDSTBYON）= 0）停止看门狗定时器以及看门狗定时器由于执行 HALT 或 STOP 指令停止时）

5.6.5 CPU 时钟状态转换框图

图 5-17 显示了该产品 CPU 时钟的状态转换框图。

图 5-17. CPU 时钟状态转换框图



注 1. 初始值且易更改。

2. 因为复位之后，已通过将系统时钟控制寄存器（CKC）设为 09H 选中了 $f_{CLK} = f_{IH}/2$ ，所以可开始基于以下工作频率的操作。

- 已使用选项字节选择了 1 MHz 时：500 kHz（1 MHz/2）
- 已使用选项字节选择了 8 MHz 或 20 MHz 时：4 MHz（8 MHz/2）

备注

1. 如果低电压检测器（LVI）由默认的选项字节设置为 ON，则在供电电压（V_{DD}）超过 2.07 V ±0.2 V^{注 1}之前复位不会被解除。

复位操作之后，状态将切换至上图中的（B）。

2. DSC：20 MHz 内部高速振荡时钟

表 5-4 显示了 CPU 时钟的转换和设置 SFR 寄存器的示例。

表 5-4. CPU 时钟转换和 SFR 寄存器设置示例 (1/6)

(1) 复位解除 (A) 之后 CPU 使用内部高速振荡时钟工作 (B)

状态转换	SFR 寄存器设置
(A) → (B)	并非必须要设置 SFR 寄存器 (复位之后的默认状态)。

(2) 复位解除 (A) 之后 CPU 使用高速系统时钟工作 (C)

(CPU 在复位解除 (B) 之后立即使用内部高速振荡时钟工作。)

(SFR 寄存器的设置顺序)

设置 SFR 寄存器的标志 状态转换	CMC 寄存器 ^{注1}			CSC 寄存器	OSMC 寄存器	OSTC 寄存器	CKC 寄存器
	EXCLK	OSCSEL	AMPH	MSTOP	FSEL		MCM0
(A) → (B) → (C) (X1 时钟: $2\text{ MHz} \leq f_x \leq 10\text{ MHz}$)	0	1	0	0	0	必须进行检 查	1
(A) → (B) → (C) (X1 时钟: $10\text{ MHz} < f_x \leq 20\text{ MHz}$)	0	1	1	0	1 ^{注2}	必须进行检 查	1
(A) → (B) → (C) (外部主时钟)	1	1	0/1	0	0/1	必须进行检 查	1

注 1. 复位之后, CMC 寄存器仅可使用 8 位存储器操作指令写入一次。

2. $f_{\text{CLK}} > 10\text{ MHz}$ 时, $\text{FSEL} = 1$ 。

如果选择了分频时钟且 $f_{\text{CLK}} \leq 10\text{ MHz}$, 则即使 $f_x > 10\text{ MHz}$ 也可使用 $\text{FSEL} = 0$ 。

注意事项 供电电压达到要进行设置的时钟的可用电压之后, 设置 X1 时钟 (参见 第二十八章 电气规范 (目标))。

(3) 复位解除 (A) 之后 CPU 使用副系统时钟工作 (D)

(CPU 在复位解除 (B) 之后立即使用内部高速振荡时钟工作。)

(SFR 寄存器的设置顺序)

设置 SFR 寄存器的标志 状态转换	CMC 寄存器 ^注			CSC 寄存器	等待振荡 稳定	CKC 寄存器
	OSCSELS	AMPHS1	AMPHS0	XTSTOP		CSS
(A) → (B) → (D)	1	0/1	0/1	0	必要的	1

注 复位之后, CMC 寄存器仅可使用 8 位存储器操作指令写入一次。

备注 表 5-4 中的(A) 至 (K)对应于图 5-17 中的 (A) 至 (K)。

表 5-4. CPU 时钟转换和 SFR 寄存器设置示例 (2/6)

(4) 复位解除 (A) 之后 CPU 使用 20 MHz 内部高速振荡时钟工作 (J)

(CPU 在复位解除 (B) 之后立即使用内部高速振荡时钟工作。)

(SFR 寄存器的设置顺序)

状态转换	设置 SFR 寄存器的标志	DSCCTL 寄存器 ^注	等待振荡稳定	DSCCTL 寄存器
		DSCON		SELDSC
(A) → (B) → (J)		1	必要的 (100 μ s)	1

注 检查 $V_{DD} \geq 2.7$ V 并设置 DSCON = 1。

(5) CPU 时钟由内部高速振荡时钟 (B) 变为高速系统时钟 (C)

(SFR 寄存器的设置顺序)

状态转换	设置 SFR 寄存器的标志			CMC 寄存器 ^{注1}			OSTS 寄存器	CSC 寄存器	OSMC 寄存器	OSTC 寄存器	CKC 寄存器
				EXCLK	OSCSEL	AMPH		MSTOP	FSEL		
(B) → (C) (X1 时钟: $2 \text{ MHz} \leq f_X \leq 10 \text{ MHz}$)				0	1	0	注 2	0	0	必须进行 检查	1
(B) → (C) (X1 时钟: $10 \text{ MHz} < f_X \leq 20 \text{ MHz}$)				0	1	1	注 2	0	1 ^{注3}	必须进行 检查	1
(B) → (C) (外部主时钟)				1	1	0/1	注 2	0	0/1	务必不要 检查	1

如果这些寄存器已经设置则不必 如果 CPU 使用高速系统时钟工作则不必要
要

- 注
1. 复位解除后, CMC 寄存器仅能更改一次。 如果已经对其进行了设置则该设置不必要。
 2. 按照以下设置振荡稳定时间。
 - 期望的 OSTC 振荡稳定时间 $\leq$ OSTS 所设置的振荡稳定时间
 3. $f_{CLK} > 10 \text{ MHz}$ 时, FSEL = 1。
如果选择了分频时钟且 $f_{CLK} \leq 10 \text{ MHz}$, 则即使 $f_X > 10 \text{ MHz}$ 也可使用 FSEL = 0。

注意事项 供电电压达到要进行设置的时钟的可用电压之后, 设置 X1 时钟 (参见 第二十八章 电气规范 (目标))。

备注 表 5-4 中的(A) 至 (K)对应于图 5-17 中的 (A) 至 (K)。

表 5-4. CPU 时钟转换和 SFR 寄存器设置示例 (3/6)

(6) CPU 时钟由内部高速振荡时钟 (B) 变为副系统时钟 (D)

(SFR 寄存器的设置顺序)

设置 SFR 寄存器的标志 状态转换	CMC 寄存器 ^注	CSC 寄存器	等待振荡稳定	CKC 寄存器
	OSCELS	XTSTOP		CSS
(B) → (D)	1	0	必要的	1

如果 CPU 使用副系统时钟工作则不必要

注 复位之后，CMC 寄存器仅可使用 8 位存储器操作指令写入一次。

(7) CPU 时钟由内部高速振荡时钟 (B) 变为 20 MHz 内部高速振荡时钟 (J)

(SFR 寄存器的设置顺序)

设置 SFR 寄存器的标志 状态转换	DSCCTL 寄存器 ^注	等待振荡稳定	DSCCTL 寄存器
	DSCON		SELDSC
(B) → (J)	1	必要 (100 μ s)	1

如果 CPU 使用 20 MHz 内部高速振荡时钟工作则不必要

注 检查 $V_{DD} \geq 2.7$ V 并设置 DSCON = 1。

(8) CPU 时钟由高速系统时钟 (C) 变为内部高速振荡钟 (B)

(SFR 寄存器的设置顺序)

设置 SFR 寄存器的标志 状态转换	CSC 寄存器	振荡精确稳定时间	CKC 寄存器
	HIOSTOP		MCM0
(C) → (B)	0	10 s	0

如果 CPU 使用内部高速振荡时钟工作则不必要

备注 表 5-4 中的(A) 至 (K)对应于图 5-17 中的 (A) 至 (K)。

表 5-4. CPU 时钟转换和 SFR 寄存器设置示例（4/6）

(9) CPU 时钟由高速系统时钟（C）变为副系统时钟（D）

（SFR 寄存器的设置顺序）

<R>

设置 SFR 寄存器的标志 状态转换	CSC 寄存器	等待振荡稳定	CKC 寄存器
	XTSTOP		CSS
(C) → (D)	0	必要的	1

如果 CPU 使用副系统时钟工作则不必要

(10) CPU 时钟由副系统时钟（D）变为内部高速振荡时钟（B）

（SFR 寄存器的设置顺序）

设置 SFR 寄存器的标志 状态转换	CSC 寄存器	CKC 寄存器	
	HIOSTOP	MCM0	CSS
(D) → (B)	0	0	0

如果 CPU 使用内部高速振荡时钟工作则不必要

如果已设置了该寄存器则不必要

备注 表 5-4 中的(A) 至 (K)对应于图 5-17 中的 (A) 至 (K)。

表 5-4. CPU 时钟转换和 SFR 寄存器设置示例 (5/6)

(11) CPU 时钟由副系统时钟 (D) 变为高速系统时钟 (C)

(SFR 寄存器的设置顺序) →

状态转换	设置 SFR 寄存器的标志	OSTS 寄存器	CSC 寄存器	OSMC 寄存器	OSTC 寄存器	CKC 寄存器	
			MSTOP	FSEL		MCM0	CSS
(D) → (C) (X1 时钟: $2 \text{ MHz} \leq f_x \leq 10 \text{ MHz}$)		注 1	0	0	必须进行检查	1	0
(D) → (C) (X1 时钟: $10 \text{ MHz} < f_x \leq 20 \text{ MHz}$)		注 1	0	1 ^{#2}	必须进行检查	1	0
(D) → (C) (外部主时钟)		注 1	0	0/1	务必不要检查	1	0

如果 CPU 使用高速系统时钟工作则不必要

如果这些寄存器已经设置则不必要

注 1. 按照以下设置振荡稳定时间。

- 期望的 OSTC 振荡稳定时间 ≤ OSTS 所设置的振荡稳定时间
2. $f_{\text{CLK}} > 10 \text{ MHz}$ 时, $\text{FSEL} = 1$ 。
如果选择了分频时钟且 $f_{\text{CLK}} \leq 10 \text{ MHz}$, 则即使 $f_x > 10 \text{ MHz}$ 也可使用 $\text{FSEL} = 0$ 。

注意事项 供电电压达到要进行设置的时钟的可用电压之后, 设置 X1 时钟 (参见 第二十八章 电气规范 (目标))。

(12) CPU 时钟由 20 MHz 内部高速振荡时钟 (J) 变为内部高速振荡时钟 (B)

(SFR 寄存器的设置顺序) →

状态转换	设置 SFR 寄存器的标志	DSCCTL 寄存器	
		SELDSC	DSCON
(J) → (B)		0	0

备注 表 5-4 中的 (A) 至 (K) 对应于图 5-17 中的 (A) 至 (K)。

表 5-4. CPU 时钟转换和 SFR 寄存器设置示例（6/6）

- (13) • CPU 使用内部高速振荡时钟（B）工作期间设为 HALT 模式（E）
- CPU 使用高速系统时钟（C）工作期间设为 HALT 模式（F）
 - CPU 使用副系统时钟（D）工作期间设为 HALT 模式（G）
 - CPU 使用 20 MHz 内部高速振荡时钟（J）工作期间设为 HALT 模式（K）

状态转换	设置
(B) → (E) (C) → (F) (D) → (G) (J) → (K)	执行 HALT 指令

- (14) • CPU 使用内部高速振荡时钟（B）工作期间设为 STOP 模式（H）
- CPU 使用高速系统时钟（C）工作期间设为 STOP 模式（I）

（设置顺序） →

<R>

状态转换		设置		
(B) → (H)		停止不能在 STOP 模式下工作的外设功能	—	执行 STOP 指令
(C) → (I)	X1 振荡中		设置 OSTS 寄存器	
	外部主系统时钟		—	

备注 表 5-4 中的(A) 至 (K)对应于图 5-17 中的 (A) 至 (K)。

CPU 时钟改变前的条件和改变后的处理

改变 CPU 时钟之前的条件以及改变 CPU 时钟之后的处理如下所示。

表 5-5. 改变 CPU 时钟（1/2）

CPU 时钟		改变前的条件	改变后的处理
改变前	改变后		
内部高速振荡时钟	X1 时钟	X1 振荡的稳定 • OSCSEL = 1、EXCLK = 0、MSTOP = 0 • 振荡稳定时间过后	可通过停止内部高速振荡器（HIOSTOP = 1）减小工作电流。
	外部主系统时钟	允许来自 EXCLK 引脚的外部时钟输入 • OSCSEL = 1、EXCLK = 1、MSTOP = 0	
	副系统时钟	XT1 振荡的稳定 • OSCSELS = 1、XTSTOP = 0 • 振荡稳定时间过后	
	20 MHz 内部高速振荡时钟	使用选项字节设置的 20 MHz 的 DSC 振荡的稳定 • $V_{DD} \geq 2.7\text{ V}$ • 设为 DSCON = 1 之后经过振荡稳定时间（100 μs ）后 • SELDSC = 1	—
X1 时钟	内部高速振荡时钟	内部高速振荡器的振荡 • HIOSTOP = 0	可停止 X1 振荡（MSTOP = 1）。
	外部主系统时钟	不可转换 （要改变时钟，应在执行一次复位之后再对其进行设置）	—
	副系统时钟	XT1 振荡的稳定 • OSCSELS = 1、XTSTOP = 0 • 振荡稳定时间过后	可停止 X1 振荡（MSTOP = 1）。
	20 MHz 内部高速振荡时钟	除非时钟变为内部高速振荡时钟，否则转换不能进行。	—
外部主系统时钟	内部高速振荡时钟	内部高速振荡器的振荡 • HIOSTOP = 0	可禁止外部主系统时钟输入（MSTOP = 1）。
	X1 时钟	不可转换 （要改变时钟，应在执行一次复位之后再对其进行设置）	—
	副系统时钟	XT1 振荡的稳定 • OSCSELS = 1、XTSTOP = 0 • 振荡稳定时间过后	可禁止外部主系统时钟输入（MSTOP = 1）。
	20 MHz 内部高速振荡时钟	除非时钟变为内部高速振荡时钟，否则不能进行转换。	—

表 5-5. 改变 CPU 时钟 (2/2)

CPU 时钟		改变前的条件	改变后的处理
改变前	改变后		
副系统时钟	内部高速振荡时钟	内部高速振荡器的振荡并选择内部高速振荡时钟作为主系统时钟 • HIOSTOP = 0、MCS = 0	可停止 XT1 振荡 (XTSTOP = 1)
	X1 时钟	X1 振荡的稳定并选择高速系统时钟作为主系统时钟 • OSCSEL = 1、EXCLK = 0、MSTOP = 0 • 振荡稳定时间过后 • MCS = 1	
	外部主系统时钟	允许 EXCLK 引脚的外部时钟输入并选择高速系统时钟作为主系统时钟 • OSCSEL = 1、EXCLK = 1、MSTOP = 0 • MCS = 1	
	20 MHz 内部高速振荡时钟	除非时钟变为内部高速振荡时钟，否则转换不能进行。	—
20 MHz 内部高速振荡时钟	内部高速振荡时钟	• SELDSC = 0 (改变时钟时设置)	可停止 20 MHz 内部高速振荡时钟 (DSCON = 0)
	X1 时钟	除非时钟变为内部高速振荡时钟，否则转换不能进行。	—
	外部主系统时钟	除非时钟变为内部高速振荡时钟，否则转换不能进行。	—
	副系统时钟	除非时钟变为内部高速振荡时钟，否则转换不能进行。	—

<R>

5.6.7 CPU 时钟和副系统时钟转换所需的时间

通过设置系统时钟控制寄存器（CKC）的位 0 至 2、4 和 6（MDIV0 至 MDIV2、MCM0、CSS），可转换 CPU 时钟（主系统时钟与副系统时钟之间）、转换主系统时钟（内部高速振荡时钟与高速系统时钟之间）并可以更改主系统时钟的分频系数。

重新写入 CKC 之后并不立即执行实际的转换操作；操作经过若干时钟的预转换时钟再继续进行（参见 表 5-6 至 表 5-9）。

可使用 CKC 的位 7（CLS）确定 CPU 是基于主系统时钟工作还是基于副系统时钟工作。 可使用 CKC 的位 5（MCS）确定主系统时钟是基于高速系统时钟工作还是基于内部高速振荡时钟工作。

当 CPU 时钟转换时，外设硬件时钟也会转换。

表 5-6. 主系统时钟转换所需的最长时间

时钟 A	转换方向	时钟 B	备注
f_{MAIN}	$\longleftrightarrow$ (改变分频系数)	f_{MAIN}	参见 表 5-7
f_{IH}	$\longleftrightarrow$	f_{MX}	参见 表 5-8
f_{MAIN}	$\longleftrightarrow$	f_{SUB}	参见 表 5-9

表 5-7. $f_{\text{MAIN}} \leftrightarrow f_{\text{MAIN}}$ 所需的最大时钟数目（改变分频系数）

转换前的设置值	转换后的设置值	
	时钟 A	时钟 B
时钟 A		$1 + f_{\text{A}}/f_{\text{B}}$ 时钟
时钟 B	$1 + f_{\text{B}}/f_{\text{A}}$ 时钟	

表 5-8. $f_{\text{IH}} \leftrightarrow f_{\text{MX}}$ 所需的最大时钟数目

转换前的设置值		转换后的设置值	
MCM0		MCM0	
		0 ($f_{\text{MAIN}} = f_{\text{IH}}$)	1 ($f_{\text{MAIN}} = f_{\text{MX}}$)
0 ($f_{\text{MAIN}} = f_{\text{IH}}$)	$f_{\text{MX}} > f_{\text{IH}}$		$1 + f_{\text{MX}}/f_{\text{IH}}$ 时钟
	$f_{\text{MX}} < f_{\text{IH}}$		$2f_{\text{IH}}/f_{\text{MX}}$ 时钟
1 ($f_{\text{MAIN}} = f_{\text{MX}}$)	$f_{\text{MX}} > f_{\text{IH}}$	$2f_{\text{MX}}/f_{\text{IH}}$ 时钟	
	$f_{\text{MX}} < f_{\text{IH}}$	$1 + f_{\text{MX}}/f_{\text{IH}}$ 时钟	

（备注 1 和 2 在下页列出。）

表 5-9. $f_{\text{MAIN}} \leftrightarrow f_{\text{SUB}}$ 所需的最大时钟数目

转换前的设置值		转换后的设置值	
CSS		CSS	
		0 ($f_{\text{CLK}} = f_{\text{MAIN}}$)	1 ($f_{\text{CLK}} = f_{\text{SUB}}$)
0 ($f_{\text{CLK}} = f_{\text{MAIN}}$)	$f_{\text{MAIN}} < f_{\text{SUB}}$		$2 + f_{\text{MAIN}}/f_{\text{SUB}}$ 时钟
	$f_{\text{MAIN}} > f_{\text{SUB}}$		$1 + 2f_{\text{MAIN}}/f_{\text{SUB}}$ 时钟
1 ($f_{\text{CLK}} = f_{\text{SUB}}$)	$f_{\text{MAIN}} < f_{\text{SUB}}$	$1 + 2f_{\text{SUB}}/f_{\text{MAIN}}$ 时钟	
	$f_{\text{MAIN}} > f_{\text{SUB}}$	$2 + f_{\text{SUB}}/f_{\text{MAIN}}$ 时钟	

- 备注**
1. 表 5-7 至 表 5-9 中列出的时钟数目是转换之前的 CPU 时钟数目。
 2. 通过删除小数部分计算表 5-7 至 表 5-9 中的时钟数。

示例 当主系统时钟由内部高速振荡时钟转换为高速系统时钟（@使用 $f_{\text{IH}} = 8 \text{ MHz}$, $f_{\text{MX}} = 10 \text{ MHz}$ 振荡）时

$$1 + f_{\text{IH}}/f_{\text{MX}} = 1 + 8/10 = 1 + 0.8 = 1.8 \rightarrow 2 \text{ 个时钟}$$

5.6.8 时钟振荡停止前的条件

以下列出了停止时钟振荡的寄存器标志设置（禁止外部时钟输入）以及停止时钟振荡之前的条件。

表 5-10. 停止时钟振荡之前的条件和标志设置

时钟	时钟振荡停止之前的条件 (禁止外部时钟输入)	SFR 寄存器的 标志设置
内部高速振荡时钟	$\text{MCS} = 1$ 或 $\text{CLS} = 1$ (CPU 基于除内部高速振荡时钟之外的时钟工作。)	$\text{HIOSTOP} = 1$
X1 时钟	$\text{MCS} = 0$ 或 $\text{CLS} = 1$ (CPU 基于除高速系统时钟之外的时钟工作。)	$\text{MSTOP} = 1$
外部主系统时钟		
副系统时钟	$\text{CLS} = 0$ (CPU 基于除副系统时钟之外的时钟工作。)	$\text{XTSTOP} = 1$
20 MHz 内部高速振荡时钟	$\text{SELDSC} = 0$ (主系统时钟基于除 20 MHz 内部高速振荡时钟之外的时钟工作。)	$\text{DSCON} = 0$

第六章 定时器阵列部件 TAUS

定时器阵列部件 TAUS 每个单元都有八个 16 位定时器。每个 16 位定时器称为一个通道，可作为一个独立的定时器使用。另外，两个或多个“通道”可用于创建一个高精度的定时器。

单一操作功能	组合操作功能
• 间隔定时器 • 方波输出 • 外部事件计数器 • 分配器功能 (仅限于 78K0R/KD3-L 和 78K0R/KE3-L 的通道 0) • 输入脉冲间隔测量 • 输入信号高/低电平宽度的测量	• PWM 输出 • 单次脉冲输出 • 多个 PWM 输出

通道 7 与串行阵列部件的 UART0 一起用于实现 LIN 总线接收处理。

各定时器阵列部件通道中是否存在定时器 I/O 引脚取决于产品。

定时器阵列部件通道	各产品的 I/O 引脚			
	KC3-L (44 引脚)	KC3-L (48 引脚)	KD3-L	KE3-L
通道 0	—	—	P00/TI00, P01/TO00	
通道 1	—	—	—	—
通道 2	P10/TI02/TO02			
通道 3	P11/TI03/TO03			
通道 4	P12/TI04/TO04			
通道 5	P13/TI05/TO05			
通道 6	P50/TI06/TO06			P14/TI06/TO06
通道 7	P51/TI07/TO07			P15/TI07/TO07

备注 通过设置输入开关控制寄存器（ISC），可将 P52/SLTI/SLTO 引脚分配至通道 0 和 1 的定时器 I/O。参见 **6.2 定时器阵列部件 TAUS 的配置** 中的“定时器 I/O 引脚配置”。

6.1 定时器阵列部件 TAUS 的功能

定时器阵列部件 TAUS 具有以下功能。

6.1.1 独立工作时各通道的功能

单一操作功能是指那些与其它通道的操作模式无关且能用于任何通道的功能（关于细节，参见 6.6.1 单一操作功能和组合操作功能的概述）。

(1) 间隔定时器

一个部件的各个定时器都可以用作按固定间隔生成中断（INTTM0n）的参考定时器。

(2) 方波输出

每次生成 INTTM0n 时都会执行切换操作，并从定时器输出引脚（TO0n、SLTO）输出占空比为 50% 的方波。

(3) 外部事件计数器

当输入至定时器输入引脚（TI0n、SLTI）的信号的有效沿数量达到特定值时，一个部件的各个定时器都可以用作生成中断的事件计数器。

(4) 分配器功能（仅通道 0）[※]

从定时器输入引脚（TI00）输入的时钟被划分，并从输出引脚（TO00）输出。

(5) 输入脉冲间隔测量

依据输入至定时器输入引脚（TI0n、SLTI）的脉冲信号的有效沿开始计数。在下一脉冲的有效沿捕获定时器的计数值。这样，就可以测量输入脉冲的间隔。

(6) 输入信号的高/低电平宽度测量

依据输入至定时器输入引脚（TI0n、SLTI）的信号的单边沿开始计数，并在另一边捕获计数值。这样，就可以测量输入信号的高电平或低电平宽度。

注 仅限于 78K0R/KD3-L 和 78K0R/KE3-L

备注 n: 通道编号（n = 0 至 7）

6.1.2 与其它通道同时工作时各通道的功能

组合操作功能是指那些通过组合使用主通道（主要是控制周期的参考定时器）和从通道（在主通道之后工作的定时器）而实现的功能（关于细节，参见 6.6.1 单一操作功能和组合操作功能的概述）。

(1) PWM（脉冲宽度调制）输出

两个通道作为一组使用以生成具有指定周期和指定占空比的脉冲。

(2) 单次脉冲输出

两个通道作为一组使用以生成具有指定延时和指定脉冲宽度的单次脉冲。

(3) 多 PWM（脉冲宽度调制）输出

通过扩展 PWM 功能以及使用一个主通道和两个或多个从通道，最多可以生成七种类型具有指定周期和指定占空比的 PWM 信号。

6.1.3 LIN 总线支持功能（仅通道 7）

(1) 唤醒信号的检测

定时器在输入至 UART0 的串行数据输入引脚（RxD0）的信号的下沿开始计数，并在上升沿处捕获定时器的计数值。这样，可以测量低电平宽度。如果低电平宽度大于指定值，则被视为唤醒信号。

(2) 同步中断域的检测

检测到唤醒信号之后，定时器在输入至 UART0 的串行数据输入引脚（RxD0）的信号的下沿开始计数，并在上升沿处捕获定时器的计数值。这样，可测量低电平宽度。如果低电平宽度大于指定值，则被视为同步中断域。

(3) 同步域的脉冲宽度测量

检测到同步中断域之后，测量输入至 UART0 的串行数据输入引脚（RxD0）的信号的低电平宽度和高电平宽度。从以这种方式测量的同步域的位间隔，计算出波特率。

6.2 定时器阵列部件 TAUS 的配置

定时器阵列部件 TAUS 包含以下硬件。

表 6-1 定时器阵列部件 TAUS 的配置

项目	配置
定时器/计数器	定时器计数器寄存器 0n (TCR0n)
寄存器	定时器数据寄存器 0n (TDR0n)
定时器输入	TI00 至 TI07、SLTI 引脚, RxD0 引脚 (对于 LIN 总线)
定时器输出	TO00 至 TO07、SLTO 引脚, 输出控制器
控制寄存器	<部件设置块的寄存器> • 外设使能寄存器 2 (PER2) • 定时器时钟选择寄存器 0 (TPS0) • 定时器通道使能状态寄存器 0 (TE0) • 定时器通道启动寄存器 0 (TS0) • 定时器通道停止寄存器 0 (TT0) • 定时器输入选择寄存器 0 (TIS0) • 定时器输出使能寄存器 0 (TOE0) • 定时器输出寄存器 0 (TO0) • 定时器输出电平寄存器 0 (TOL0) • 定时器输出模式寄存器 0 (TOM)
	<各个通道的寄存器> • 定时器模式寄存器 0n (TMR0n) • 定时器状态寄存器 0n (TSR0n) • 输入转换控制寄存器 (ISC) • 噪声滤波器使能寄存器 1, 2 (NFEN1, NFEN2) • 端口模式寄存器 0, 1, 5 (PM0, PM1, PM5) • 端口寄存器 0, 1, 5 (P0, P1, P5)

备注 n: 通道编号 (n = 0 至 7)

• 定时器 I/O 引脚配置

通过设置输入开关控制寄存器（ISC），可将 P52/SLTI/SLTO 引脚分配至通道 0 和 1 的定时器 I/O。（关于输入转换控制寄存器（ISC）的详情，参见 **6.3 (13) 输入转换控制寄存器（ISC）**。）

可以为通道 0 和 1 选择以下 I/O 引脚。

表 6-2. 可为通道 0 和 1 选择的 I/O 引脚

可被选择的I/O引脚所用的通道	输入引脚	输出引脚
通道 0	• P00/TI00 引脚[※] • P52/SLTI/SLTO 引脚	• P01/TO00 引脚[※] • P52/SLTI/SLTO 引脚
通道 1	P52/SLTI/SLTO 引脚	P52/SLTI/SLTO 引脚

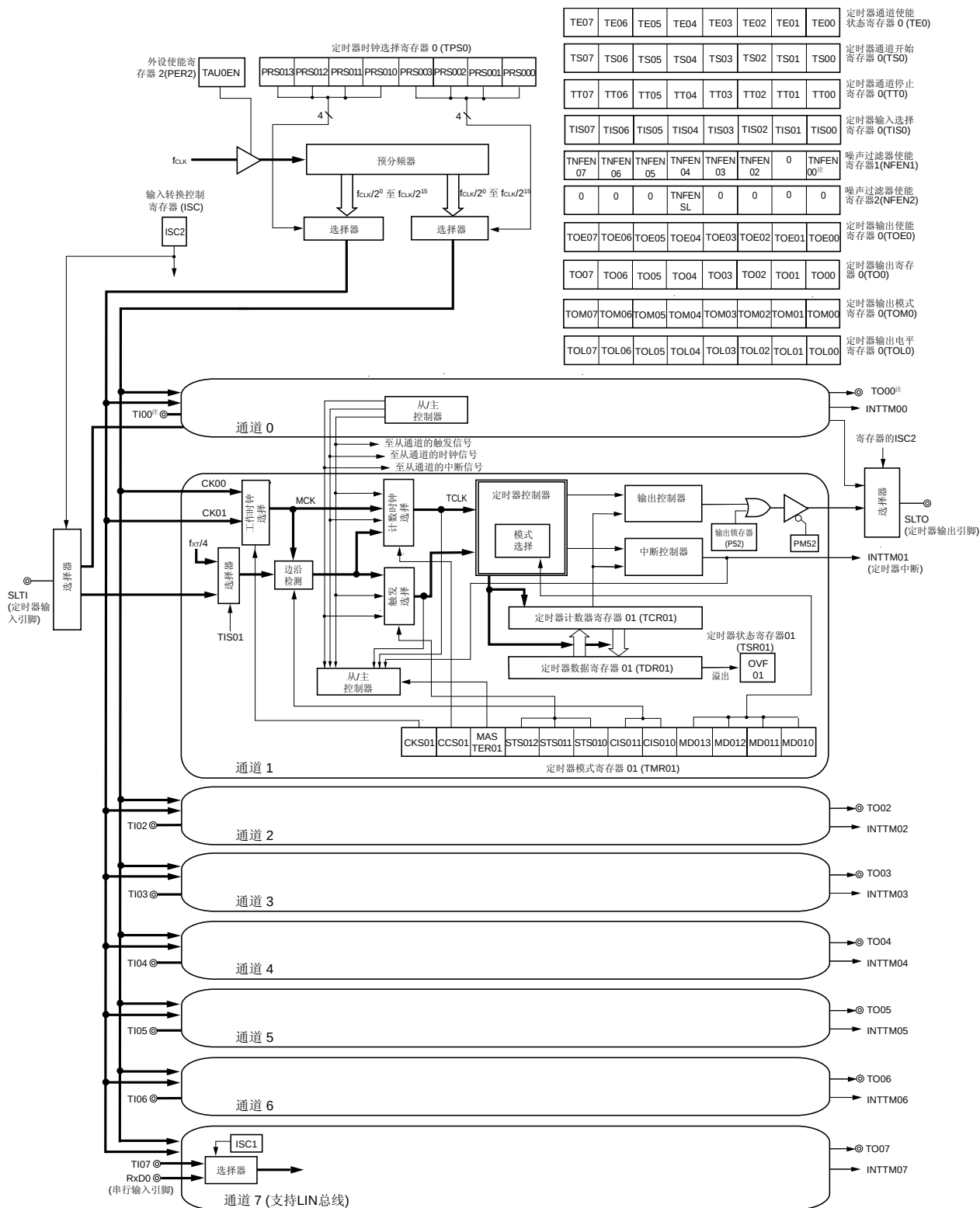
注 仅限于 78K0R/KD3-L 和 78K0R/KE3-L。在 78K0R/KC3-L 中，仅 P52/SLTI/SLTO 引脚可分配至通道 0 和 1。

注意事项 在下文中，定时器 I/O 引脚作为 TIn 和 TOn（n = xx）说明，也包括 SLTI 和 SLTO 引脚的选择。

- 备注
1. 当同一引脚共享定时器输入和定时器输出时，仅定时器输入或定时器输出之一可以使用。
 2. 仅上述提到的通道之一可以分配为 P52/SLTI/SLTO 引脚的定时器 I/O 引脚。
 3. 不能选择 SLTI 和 SLTO 引脚作为除上述的通道（通道 2 至 7）之外的通道的定时器 I/O。

图 6-1 显示了其框图。

图 6-1. 定时器阵列部件的框图



注 仅限于 78K0R/KD3-L 和 78K0R/KE3-L

(1) 定时器/计数器寄存器 0n (TCR0n)

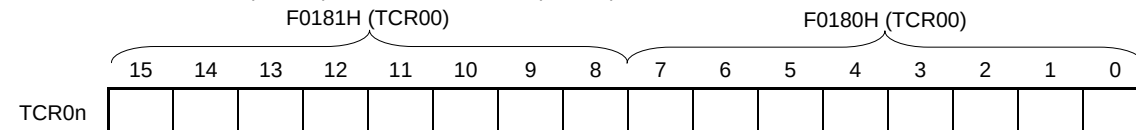
TCR0n 是 16 位的只读寄存器，用于计算时钟数。

该计数器的值与计数时钟的上升沿同步增加或减少。

计数器是增加还是减少取决于由 TMR0n 的 MD0n3 至 MD0n0 位选择的操作模式。

图 6-2. 定时器/计数器寄存器 0n (TCR0n) 的格式

地址: F0180H, F0181H (TCR00) 至 F018EH, F018FH (TCR07) 复位后: FFFFH R



(n = 0 至 7)

可通过读取 TCR0n 来读取计数值。

在以下情况时，计数值被设为 FFFFH。

- 复位信号产生时
- 外设使能寄存器 2（PER2）的 TAU0EN 位被清除时
- 在 PWM 输出模式下完成从通道的计数时
- 在单次脉冲输出模式下完成主/从通道的计数时
- 在多 PWM 输出模式下完成从通道的计数时

在以下情况时，计数值被清除为 0000H。

- 在捕获模式下输入起始触发时
- 捕获模式下完成了捕获时

注意事项 即使读取 TCR0n 时, 计数值也不会被捕获到 TDR0n。

TCR0n 寄存器的读取值根据操作模式变化和操作状态不同而不同，如下所示。

表 6-3. 各种不同操作模式下 TCR0n 寄存器的读取值

操作模式	计数模式	TCR0n 寄存器读取值 [※]			
		复位后操作模式变化	暂停计数操作后操作模式变化（TT0n = 1）	暂停计数操作后操作重新开始（TT0n = 1）	单次计数后开始触发器等待状态期间
间隔定时器模式	向下计数	FFFFH	未定义	Stop value	—
捕获模式	向上计数	0000H	未定义	Stop value	—
事件计数器模式	向下计数	FFFFH	未定义	Stop value	—
单次计数模式	向下计数	FFFFH	未定义	Stop value	FFFFH
捕获 & 单次计数模式	向上计数	0000H	未定义	Stop value	TDR0n 寄存器的捕获值 + 1

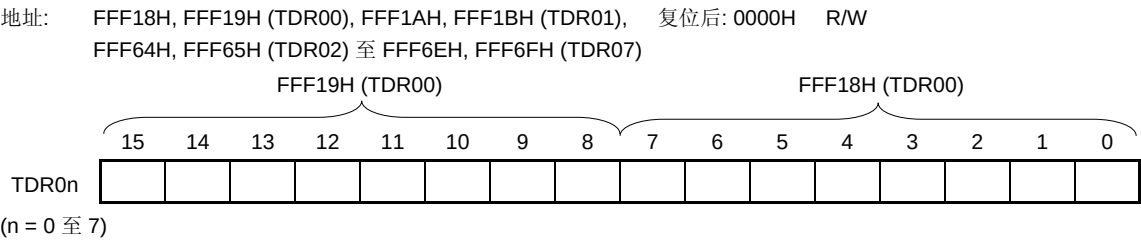
注 显示 TE0n = 0 期间，TS0n 设为“1”时 TCR0n 寄存器的读取值。该读取值一直保持在 TCR0n 寄存器中，直到计数操作开始。

备注 n = 0 至 7

(2) 定时器数据寄存器 0n (TDR0n)

这是一款 16 位的寄存器，从该寄存器可以选择捕获功能和比较功能。
通过使用 TMR0n 的 MD0n3 至 MD0n0 位选择操作模式可以转换捕获或比较功能。
TDR0n 的值可以在任意时刻进行更改。
该寄存器可按 16 位为单位进行读写。
产生复位信号将该寄存器清为 0000H。

图 6-3. 定时器数据寄存器 0n (TDR0n) 的格式



(i) TDR0n 用作比较寄存器时

从设置至 TDR0n 的值开始向下计数。 当计数值达到 0000H 时，产生一个中断信号 (INTTM0n)。
TDR0n 保持它的值直到将其重写

注意事项 当将 TDR0n 设为比较功能时，即使输入一个捕获触发它也不会执行捕获操作。

(ii) TDR0n 用作捕获寄存器时

当输入捕获触发时，TCR0n 的计数值被捕获至 TDR0n。
可选择 TI0n 引脚的有效沿选作为捕获触发。 由 TMR0n 进行该选择。

备注 n = 0 至 7

6.3 控制定时器阵列部件的寄存器

定时器阵列部件由以下寄存器控制。

- 外设使能寄存器 2 (PER2)
- 定时器时钟选择寄存器 0 (TPS0)
- 定时器模式寄存器 0n (TMR0n)
- 定时器状态寄存器 0n (TSR0n)
- 定时器通道使能状态寄存器 0 (TE0)
- 定时器通道开始寄存器 0 (TS0)
- 定时器通道停止寄存器 0 (TT0)
- 定时器输入选择寄存器 0 (TIS0)
- 定时器输出使能寄存器 0 (TOE0)
- 定时器输出寄存器 0 (TO0)
- 定时器输出电平寄存器 0 (TOL0)
- 定时器输出模式寄存器 0 (TOM0)
- 输入开关控制寄存器 (ISC)。
- 噪声过滤器使能寄存器 1、2 (NFEN1、NFEN2)
- 端口模式寄存器 0、1、5 (PM0、PM1、PM5)
- 端口寄存器 0、1、5 (P0、P1、P5)

备注 n = 0 至 7

(1) 外设使能寄存器 2 (PER2)

PER2 用于启动或禁止各外设硬件宏功能的使用。 停止供应给未使用的硬件宏的时钟，以减少功耗和噪声。
当使用定时器阵列部件 TAUS 时，确保将该寄存器的位 0 (TAU0EN) 设置为 1。
PER2 可由 1 位或 8 位存储器操作指令设置。
产生复位信号将该寄存器清除为 00H。

- 注意事项 1. 设置定时器阵列部件 TAUS 时，务必要先将 TAU0EN 设为 1。 如果 TAU0EN = 0，则会忽略定时器阵列部件 TAUS 的控制寄存器的写入，且所有读取值都为默认值。
2. 务必将 PER2 寄存器的位 1 至 7 清为 0。

图 6-4. 外设使能寄存器 2 (PER2) 的格式

地址: F00F2H	复位后: 00H	R/W						
符号	7	6	5	4	3	2	1	<0>
PER2	0	0	0	0	0	0	0	TAU0EN

TAU0EN	定时器阵列部件输入时钟供应的控制
0	停止提供输入时钟。 - 不能写入定时器阵列部件TAUS所使用的SFR。 - 定时器阵列部件TAUS处于复位状态。
1	提供输入时钟。 可以读写定时器阵列部件TAUS所使用的SFR。

(2) 定时器时钟选择寄存器 0 (TPS0)

TPS0 是 16 位寄存器，用于选择通常供应各个通道的工作时钟（CK00、CK01）的两种类型。CK01 由 TPS0 的位 7 至位 4 选择；CK00 由位 3 至位 0 选择。

定时器运行期间，仅在下列情况中可进行 TPS0 的重写。

重写 PRS000 至 PRS003 位： 仅当在操作停止状态（TE0n = 0）下将所有通道设为 CKS0n = 0 时可进行

重写 PRS010 至 PRS013 位： 仅当在操作停止状态（TE0n = 1）下将所有通道设为 CKS0n = 0 时可进行

TPS0 可由 16 位存储器操作指令设置。

TPS0 的低八位可使用 TPS0L 由 8 位存储器操作指令设置。

产生复位信号将该寄存器清除为 0000H。

图 6-5. 定时器时钟选择寄存器 0 (TPS0) 的格式

地址: F01B6H, F01B7H 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TPS0	0	0	0	0	0	0	0	0	PRS 013	PRS 012	PRS 011	PRS 010	PRS 003	PRS 002	PRS 001	PRS 000

PRS 0m3	PRS 0m2	PRS 0m1	PRS 0m0		工作时钟（CK0m）* 的选择			
					f _{CLK} = 2 MHz	f _{CLK} = 5 MHz	f _{CLK} = 10 MHz	f _{CLK} = 20 MHz
0	0	0	0	f _{CLK}	2 MHz	5 MHz	10 MHz	20 MHz
0	0	0	1	f _{CLK} /2	1 MHz	2.5 MHz	5 MHz	10 MHz
0	0	1	0	f _{CLK} /2 ²	500 kHz	1.25 MHz	2.5 MHz	5 MHz
0	0	1	1	f _{CLK} /2 ³	250 kHz	625 kHz	1.25 MHz	2.5 MHz
0	1	0	0	f _{CLK} /2 ⁴	125 kHz	312.5 kHz	625 kHz	1.25 MHz
0	1	0	1	f _{CLK} /2 ⁵	62.5 kHz	156.2 kHz	312.5 kHz	625 kHz
0	1	1	0	f _{CLK} /2 ⁶	31.25 kHz	78.1 kHz	156.2 kHz	312.5 kHz
0	1	1	1	f _{CLK} /2 ⁷	15.62 kHz	39.1 kHz	78.1 kHz	156.2 kHz
1	0	0	0	f _{CLK} /2 ⁸	7.81 kHz	19.5 kHz	39.1 kHz	78.1 kHz
1	0	0	1	f _{CLK} /2 ⁹	3.91 kHz	9.76 kHz	19.5 kHz	39.1 kHz
1	0	1	0	f _{CLK} /2 ¹⁰	1.95 kHz	4.88 kHz	9.76 kHz	19.5 kHz
1	0	1	1	f _{CLK} /2 ¹¹	976 Hz	2.44 kHz	4.88 kHz	9.76 kHz
1	1	0	0	f _{CLK} /2 ¹²	488 Hz	1.22 kHz	2.44 kHz	4.88 kHz
1	1	0	1	f _{CLK} /2 ¹³	244 Hz	610 Hz	1.22 kHz	2.44 kHz
1	1	1	0	f _{CLK} /2 ¹⁴	122 Hz	305 Hz	610 Hz	1.22 kHz
1	1	1	1	f _{CLK} /2 ¹⁵	61 Hz	153 Hz	305 Hz	610 Hz

注 改变 f_{CLK} 所选的时钟（通过改变系统时钟控制寄存器（CKC）的值进行）时，停止定时器阵列部件 TAUS（TT0 = 00FFH）。

注意事项 务必将位 15 至位 8 清为“0”。

备注 1. f_{CLK}: CPU/外设硬件时钟频率
2. m = 0, 1 n = 0 至 7

(3) 定时器模式寄存器 0n (TMR0n)

TMR0n 设置通道 n 的操作模式。它用于选择操作时钟 (MCK)、计数时钟、是作为主定时器还是从定时器操作、开始触发器和捕获触发器、定时器输入的有效沿以及操作模式 (间隔、捕获、事件计数器、单次计数或捕获 & 单次计数)。

当寄存器处于工作中 (TE0 = 1 时) 时禁止重写 TMR0n。但是, 即使寄存器正在使用某些功能工作 (TE0 = 1 时) 期间也可以重写位 7 和 6 (CIS0n1、CIS0n0) (关于细节, 参见 6.7 定时器阵列部件 TAUS 作为独立通道的操作 和 6.8 定时器阵列部件 TAUS 复通道的操作)。

TMR0n 可由 16 位存储器操作指令设置。

产生复位信号将该寄存器清除为 0000H。

图 6-6. 定时器模式寄存器 0n (TMR0n) 的格式 (1/3)

地址: F0190H, F0191H (TMR00) 至 F019EH, F019FH (TMR07) 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMR0n	CKS 0n	0	0	CCS 0n	MAST ER0n	STS 0n2	STS 0n1	STS 0n0	CIS 0n1	CIS 0n0	0	0	MD 0n3	MD 0n2	MD 0n1	MD 0n0

CKS 0n	通道n的工作时钟（MCK）的选择
0	工作时钟 CK00 由 TPS0 寄存器设置
1	工作工作时钟时钟 CK01 由 TPS0 寄存器设置
工作时钟MCK由边沿检测器使用。 根据CCS0n位的设置生成计数时钟（TCLK）和采样时钟。	

CCS 0n	通道 n 的计数时钟（TCLK）的选择
0	工作时钟MCK由CKS0n位指定
1	从TI0n 输入的输入信号的有效沿/子系统时钟被4倍分频率（fsub/4）
计数时钟 TCLK用于定时器/计数器、输出控制器和中断控制器。	

MAS TER 0n	选择单一操作功能中工作或在组合操作功能中作为从通道功能/在通道n的组合操作功能中作为主通道工作
0	在单次操作功能中工作或在组合操作功能中作为从通道工作
1	在组合操作功能中作为主通道工作。
仅偶通道可设为主通道（MASTER0n = 1）。 务必将奇通道用作从通道（MASTER0n = 0）。 对于使用单一操作功能的通道，将MASTER0n清为0。	

注意事项 务必将位 14、13、5 和 4 清为“0”。

备注 n = 0 至 7

图 6-6. 定时器模式寄存器 0n（TMR0n）的格式（2/3）

地址: F0190H, F0191H (TMR00) 至 F019EH, F019FH (TMR07) 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMR0n	CKS 0n	0	0	CCS 0n	MAST ER0n	STS 0n2	STS 0n1	STS 0n0	CIS 0n1	CIS 0n0	0	0	MD 0n3	MD 0n2	MD 0n1	MD 0n0

STS 0n2	STS 0n1	STS 0n0	通道n的开始触发或捕获触发的设置
0	0	0	仅软件触发开始有效（未选择其它触发源）。
0	0	1	TI0n引脚输入的有效沿既用作开始触发器也用作捕获触发。
0	1	0	TI0n引脚输入的两个边沿都用作开始触发和捕获触发。
1	0	0	使用主通道的中断信号（当使用组合操作功能将通道用作从通道时）。
其它			禁止设置

CIS 0n1	CIS 0n0	TI0n引脚输入有效沿的选择
0	0	下降沿
0	1	上升沿
1	0	双边沿（测量低电平宽度时） 开始触发：下降沿、捕获触发、上升沿
1	1	双边沿（测量高电平宽度时） 开始触发：上升沿、捕获触发、下降沿
如果在STS0n2 至 STS0n0位的值不是010B时指定两个边沿，则将CIS0n1 至 CIS0n0位设为10B。		

备注 n = 0 至 7

图 6-6. 定时器模式寄存器 0n (TMR0n) 的格式 (3/3)

地址: F0190H, F0191H (TMR00) 至 F019EH, F019FH (TMR07) 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMR0n	CKS 0n	0	0	CCS 0n	MAST ER0n	STS 0n2	STS 0n1	STS 0n0	CIS 0n1	CIS 0n0	0	0	MD 0n3	MD 0n2	MD 0n1	MD 0n0

MD 0n3	MD 0n2	MD 0n1	MD 0n0	通道 n 的工作模式	TCR的计数操作	独立操作
0	0	0	1/0	间隔定时器模式	向下计数	可行
0	1	0	1/0	捕获模式	向上计数	可行
0	1	1	0	事件计数器模式	向下计数	可行
1	0	0	1/0	单次计数模式	向下计数	不可行
1	1	0	0	捕获 & 单次计数模式	向上计数	可行
其它				禁止设置		
MD0n0位的操作根据各个操作模式的不同而不同（参见下表）。						

操作模式 (由 MD0n3 至 MD0n1 位设置的值 (参见上表))	MD 0n0	开始计数和中断的设置
- 间隔定时器模式 (0、0、0) - 捕获模式 (0、1、0)	0	当计数开始时不产生定时器中断 (定时器的输出也不改变)。
	1	当计数开始时产生定时器中断 (定时器的输出也会改变)
事件计数器模式 (0、1、1)	0	当计数开始时不产生定时器中断 (定时器的输出也不会改变)
单次计数模式^{#1} (1、0、0)	0	计数操作期间开始触发无效。 以此同时也不会产生中断。
	1	计数操作期间 ^{#2} 开始触发有效。 与此同时还会产生中断。
捕获 & 单次计数模式 (1、1、0)	0	当计数开始时不产生定时器中断 (定时器的输出也不会改变) 计数操作期间开始触发无效。 与此同时也不会产生中断。
其它		禁止设置

- 注
1. 在单次计数模式下, 开始计数操作时的中断输出 (INTTM0n) 和 Ton 输出得不到控制。
 2. 如果在操作期间生成开始触发 (TS0n = 1), 则计数器会被清除、产生一个中断并开始重新计数。

备注 n = 0 至 7

(4) 定时器状态寄存器 0n (TSR0n)

TSR0n 指示通道 n 的计数器的溢出状态。

TSR0n 仅在捕获模式 (MD0n3 至 MD0n1 = 010B) 和捕获&单次计数模式 (MD0n3 至 MD0n1 = 110B) 下有效。 在任何其它模式中都不能对其进行设置。 关于各个操作模式下 OVF 位的操作和设置/清除条件, 请参见表 6-4。

TSR0n 可由 16 位存储器操作指令读取。

TSR0n 的低八位可使用 TSR0nLL 由 8 位存储器操作指令设置。

产生复位信号将该寄存器清除为 0000H。

图 6-7. 定时器状态寄存器 0n (TSR0n) 的格式

地址: F01A0H, F01A1H (TSR00) 至 F01AEH, F01AFH (TSR07) 复位后: 0000H R

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TSR0n	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	OVF

OVF	通道n的计数器溢出状态
0	不发生溢出。
1	发生溢出。
OVF = 1时, 当在不溢出的情况下捕获下一个值时清除该标志 (OVF = 0)。	

表 6-4. 各个操作模式下 OVF 位的操作和设置/清除条件

定时器操作模式	OVF	设置/清除条件
• 捕获模式	清除	当未基于捕获发生溢出时
• 捕获 & 单次计数模式	设置	当基于捕获发生溢出时
• 间隔定时器模式	清除	— (禁止使用, 不设置也不清除)
• 事件计数器模式	设置	
• 单次计数模式		

备注 计数器发生溢出后, OVF 位并不立即发生改变, 而是在下一个捕获时发生改变。

(5) 定时器通道使能状态寄存器 0（TE0）

TE0 用于使能或停止各个通道的定时器操作。

当将定时器通道开始寄存器 0（TS0）的一位设为 1 时，则该寄存器的相应位设为 1。当将定时器通道停止寄存器 0（TT0）的一位设为 1 时，则将该寄存器的相应位清为 0。

TE0 可由 16 位存储器操作指令读取。

可通过 TE0L 使用 1 位或 8 位存储器操作指令设置 TE0 的低八位。

产生复位信号将该寄存器清除为 0000H。

图 6-8. 定时器通道使能状态寄存器（TE0）的格式

地址: F01B0H, F01B1H 复位后: 0000H R

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TE0	0	0	0	0	0	0	0	0	TE07	TE06	TE05	TE04	TE03	TE02	TE01	TE00

TE0n	通道n的操作使能/停止状态的指示
0	操作停止。
1	允许操作。

备注 n = 0 至 7

(6) 定时器通道开始寄存器 0 (TS0)

TS0 是一个触发器寄存器，用于清除定时器计数器 (TCR0n) 并开始各个通道的计数操作。
当该寄存器的一个位 (TS0n) 设为 1 时，定时器通道使能状态寄存器 0 (TE0) 的对应位 (TE0n) 被设为 1。
TS0n 位为触发位，当 TE0n = 1 时立即被清除。
TS0 可由 16 位存储器操作指令设置。
可通过 TS0L 使用 1 位或 8 位存储器操作指令设置 TS0 的低八位。
产生复位信号将该寄存器清除为 0000H。

图 6-9. 定时器通道开始寄存器 0 (TS0) 的格式

地址: F01B2H, F01B3H 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TS0	0	0	0	0	0	0	0	0	TS07	TS06	TS05	TS04	TS03	TS02	TS01	TS00

TS0n	通道n的操作使能（开始）触发
0	无触发操作
1	TE0n被设为1且计数操作变为允许。 计数操作使能状态下，TCR0n计数操作根据各个操作模式不同而不同（参见 表 6-5）。

- 注意事项** 务必将位 15 至位 8 清为“0”。
- 备注**
- 1. 读取 TS0 寄存器时，总是读取到 0。
 - 2. n = 0 至 7

表 6-5. 从计数操作使能状态到 TCR0n 计数开始的操作 (1/2)

定时器操作模式	设置为TS0n = 1时的操作
• 间隔定时器模式	从触发检测（TS0n=1）一直到计数时钟产生都不执行操作。 第一个计数时钟载入TDR0n至TCR0n的值；随后的计数时钟执行向下计数操作（参见 6.3 (6) (a) 间隔定时器模式中的开始时序）。
• 事件计数器模式	向TS0n位写入1载入TDR0n至TCR0n的值。 随后的计数时钟执行向下计数的操作。 在TMR0n 寄存器中，STS0n2至STS0n0位所选择的外部触发检测并不开始计数操作（参见 6.3 (6) (b) 事件计数器模式中的开始时序）。
• 捕获模式	从触发检测一直到计数时钟产生都不执行操作。 第一个计数时钟将0000H载入TCR0n；随后的计数时钟执行向上计数操作（参见 6.3 (6) (c) 捕获模式中的开始时序）。

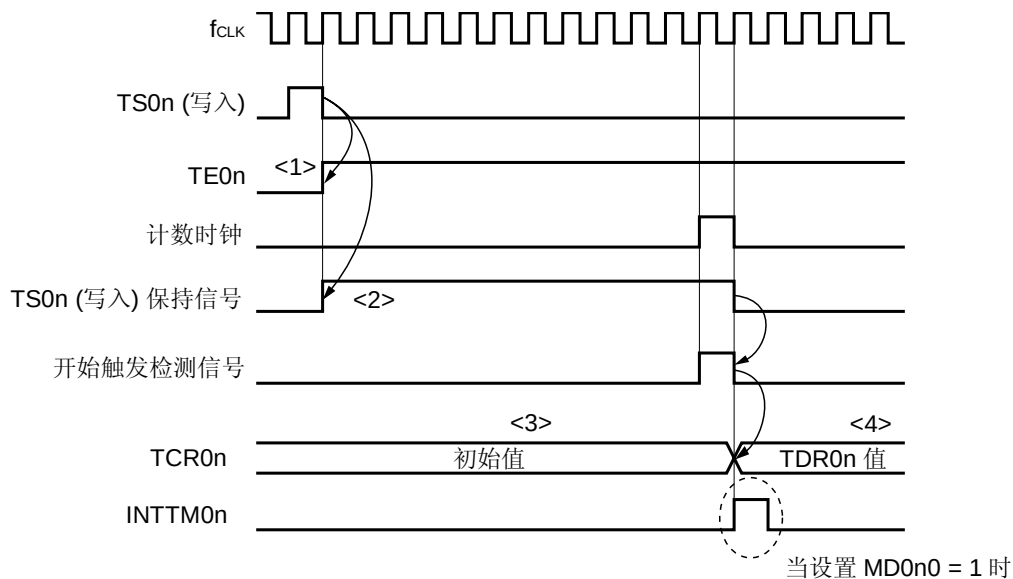
表 6-5. 从计数操作使能状态到 TCR0n 计数开始的操作 (2/2)

定时器操作模式	设置为TS0n = 1时的操作
• 单次计数模式	当 TS0n = 0 时，向 TS0n 位写入 1 设置开始触发等待状态。 从触发检测一直到计数时钟产生都不执行操作。 第一个计数时钟载入 TDR0n 至 TCR0n 的值；随后的计数时钟执行向下计数操作（参见 6.3 (6) (d) 单次计数模式中的开始时序）。
• 捕获 & 单次计数模式	当TS0n = 0时，向TS0n位写入1设置开始触发等待状态。 从触发检测一直到计数时钟产生都不执行操作。 第一个计数时钟将0000H载入TCR0n；随后的计数时钟执行向上计数操作（参见 6.3 (6) (e) 捕获 & 单次计数中的开始时序）。

(a) 间隔定时器模式中的开始时序

- <1> 向 TS0n 写入 1 设置 TE0n = 1。
 <2> 写入至 TS0n 的数据会一直保持到计数时钟产生。
 <3> TCR0n 在计数时钟产生之前保持初始值。
 <4> 计数时钟产生时，“TDR0n 值”被载入 TCR0n 并开始计数。

图 6-10. 开始时序（间隔定时器模式中）

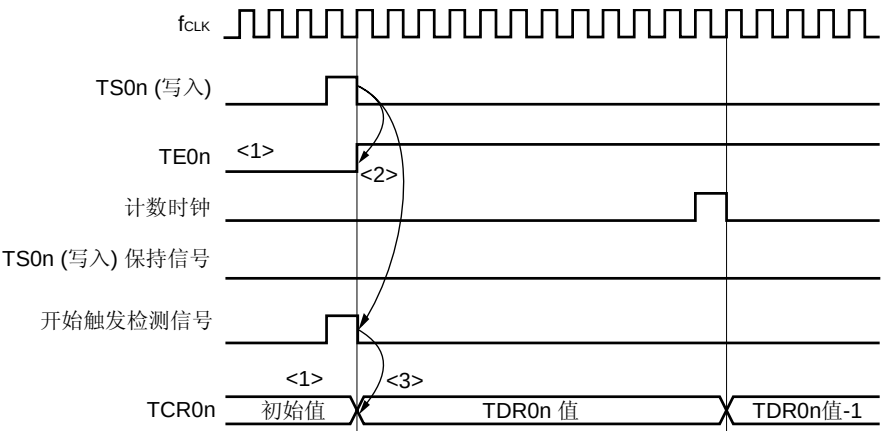


注意事项 写入 TS0n 之后，在计数时钟的第一个周期操作中，因为计数开始延迟到产生了计数时钟，所以会产生最大为一个时钟的误差。当需要计数开始时刻的信息时，可通过设置 MD0n0 = 1 在计数开始处产生一个中断。

(b) 事件计数器模式中的开始时序

- <1> TE0n 设为 0 期间，TCR0n 保持其初始值。
- <1> 向 TS0n 写入 1 将 TE0n 设为 1。
- <3> 一旦将 1 写入 TS0n 且将 1 设置至 TE0n，“TDR0n 值”立即被载入 TCR0n 以开始计数。
- <4> 计数开始之后，TCR0n 的值根据计数时钟向下计数。

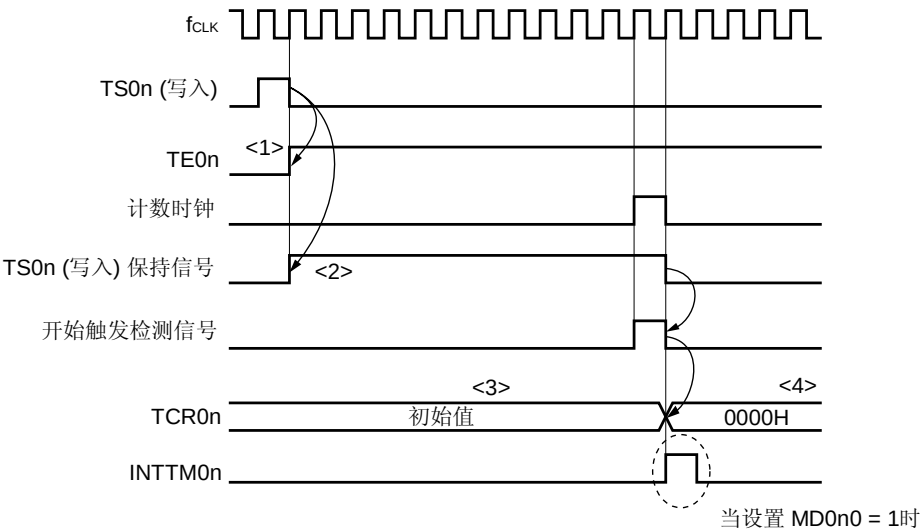
图 6-11. 开始时序（事件计数器模式中）



(c) 捕获模式中的开始时序

- <1> 向 TS0n 写入 1 设置 TE0n = 1。
- <2> 写入至 TS0n 的数据会一直保持到计数时钟产生。
- <3> TCR0n 在计数时钟产生之前保持初始值。
- <4> 计数时钟产生时，0000H 被载入 TCR0n 并开始计数。

图 6-12. 开始时序（捕获模式中）

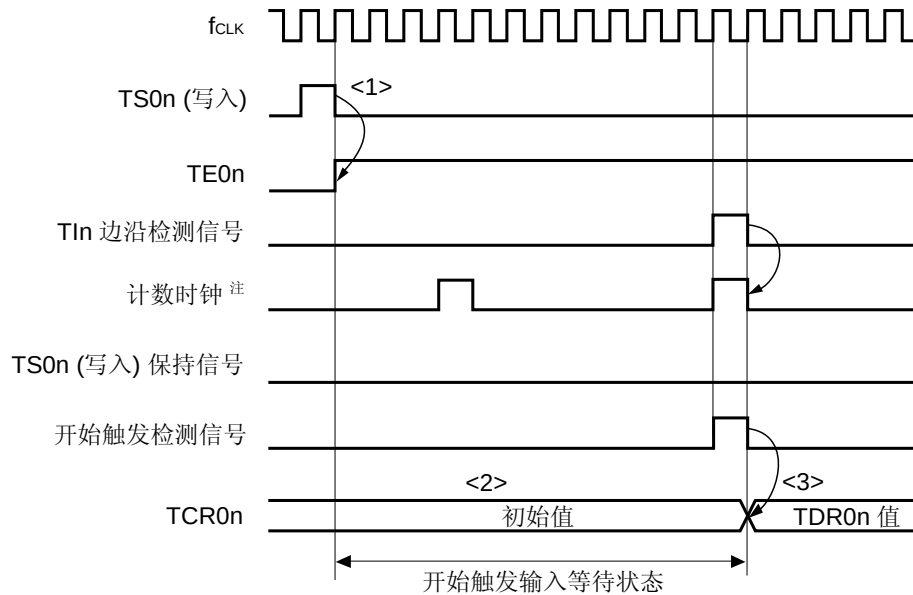


注意事项 写入 TS0n 之后，在计数时钟的第一个周期操作中，因为计数开始延迟到产生了计数时钟，所以产生最大为一个时钟的误差。 当需要计数开始时刻的信息时，可通过设置 MD0n0 = 1 在计数开始处产生一个中断。

(d) 单次计数模式中的开始时序

- <1> 向 TS0n 写入 1 设置 TE0n = 1。
- <2> 进入开始触发输入等待状态，且 TCR0n 保持其初始值。
- <4> 开始触发检测时，“TDR0n 值”被载入 TCR0n 并开始计数。

图 6-13. 开始时序（单次计数模式中）



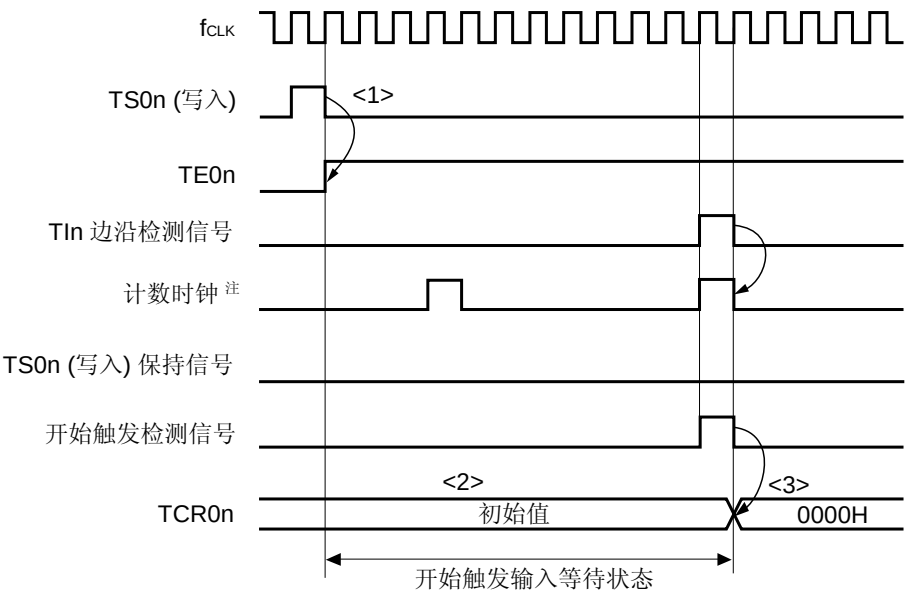
注 当设置单次计数模式时，选择工作时钟（MCK）作为计数时钟（CCS0n = 0）。

注意事项 因为操作在开始触发检测时开始，所以会产生输入信号采样误差（使用 TIn 时，误差为一个计数时钟）。

(e) 捕获&单次计数模式中的开始时序

- <1> 向 TS0n 写入 1 设置 TE0n = 1。
- <2> 进入开始触发输入等待状态，且 TCR0n 保持其初始值。
- <3> 开始触发检测时，0000H 被载入 TCR0n 并开始计数。

图 6-14. 开始时序（捕获&单次计数模式中）



注 当设置捕获&单次计数模式时，选择工作时钟（MCK）作为计数时钟（CCS0n = 0）。

注意事项 因为操作在开始触发检测时开始，所以会产生输入信号采样误差（使用 TIn 时，误差为一个计数时钟）。

(7) 定时器通道停止寄存器 0（TT0）

TT0 是一个触发器寄存器，用于清除定时器计数器（TCR0n）并开始各个通道的计数操作。
当该寄存器的一个位（TT0n）设为 1 时，定时器通道使能状态寄存器 0（TE0）的对应位（TE0n）被设为 0。
TT0n 位为触发位，当 TE0n = 0 时立即被清为 0。
TT0 可由 16 位存储器操作指令设置。
可通过 TT0L 使用 1 位或 8 位存储器操作指令设置 TT0 的低八位。
产生复位信号将该寄存器清除为 0000H。

图 6-15. 定时器通道停止寄存器 0（TT0）的格式

地址: F01B4H, F01B5H 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TT0	0	0	0	0	0	0	0	0	TT07	TT06	TT05	TT04	TT03	TT02	TT01	TT00

TT0n	通道n的操作停止触发
0	无触发操作
1	操作停止（产生操作触发）。

注意事项 务必将位 15 至位 8 清为“0”。

备注 1. 读取 TT0 寄存器时，总是读取到 0。
 2. n = 0 至 7

(8) 定时器输入选择寄存器 0（TIS0）

TIS0 用于选择输入至定时器输入引脚（TI0n）的信号或者 4 倍分频的副系统时钟（fxτ/4）对各个通道是否有效。
TIS0 可由 1 位或 8 位存储器操作指令进行设置。
产生复位信号将该寄存器清除为 00H。

图 6-16. 定时器输入选择寄存器 0（TIS0）的格式

地址: FFF3EH 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
TIS0	TIS07	TIS06	TIS05	TIS04	TIS03	TIS02	TIS01	TIS00

TIS0n	通道n所用的定时器输入/子系统时钟的选择
0	定时器输入引脚（TI0n）的输入信号
1	子系统时钟被4分频（f _{XT} /4）

(9) 定时器输出使能寄存器 0 (TOE0)

TOE0 用于使能或禁止各个通道的定时器输出。

已启动定时器输出的通道 n 将不能重写定时器输出寄存器 (TO0) 的 TO0n 位的值，该寄存器随后由软件描述，且通过计数操作反映定时器输出功能设置的值会从定时器输出引脚 (TO0n) 输出。

TOE0 可由 16 位存储器操作指令设置。

可通过 TOE0L 使用 1 位或 8 位存储器操作指令设置 TOE0 的低八位。

产生复位信号将该寄存器清除为 0000H。

图 6-17. 定时器输出使能寄存器 (TOE0) 的格式

地址: F01BAH, F01BBH 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TOE0	0	0	0	0	0	0	0	0	TOE07	TOE06	TOE05	TOE04	TOE03	TOE02	TOE01	TOE00

TOE0n	通道n的定时器输出使能/禁止
0	TO0n 操作通过计数操作停止（定时器通道输出位）。 允许写入 TO0n 位。 TO0n 引脚可用作数据输出引脚，并输出设置到 TO0n 位的电平。 TO0n 引脚的输出电平可通过软件操纵。
1	TO0n 操作由计数操作使能（定时器通道输出位）。 禁止写入 TO0n 位（写入被忽略）。 TO0n 引脚用作定时器输出引脚；TOE0n 根据定时器操作置位或者复位。 TO0n 引脚根据定时器操作输出方波或 PWM。

注意事项 务必将位 15 至位 8 清为“0”。

备注 n = 0 至 7

(10) 定时器输出寄存器 0 (TO0)

TO0 是各个通道定时器输出的缓冲器寄存器。
该寄存器中各个位的值会从各通道的定时器输出引脚 (TO0n) 输出。
仅当禁止定时器输出 (TOE0n = 0) 时, 该寄存器可以由软件重写。 当允许定时器输出 (TOE0n = 1) 时, 忽略由软件对该寄存器的重写, 且仅能由定时器操作对值进行更改。
要使用 P01/TO00、P10/TO02、P11/TO03、P12/TO04、P13/TO05、P14/TO06[※]、P15/TO07[※]或 P52/SLTO 作为端口功能引脚, 则需将对应的 TO0n 位设为“0”。
TO0 可由 16 位存储器操作指令设置。
TO0 的低八位可使用 TO0L 由 8 位存储器操作指令设置。
产生复位信号将该寄存器清除为 0000H。

注 在非 78K0R/KE3-L 产品中, TO06 和 TO07 分别与 P50 和 P51 复用 (P50/TO06、P51/TO07)。

图 6-18. 定时器输出寄存器 0 (TO0) 的格式

地址: F01B8H, F01B9H 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TO0	0	0	0	0	0	0	0	0	TO0 7	TO0 6	TO0 5	TO0 4	TO0 3	TO0 2	TO0 1	TO0 0

TO0 n	通道 n 的定时器输出
0	定时器输出值为“0”。
1	定时器输出值为“1”。

注意事项 务必将位 15 至位 8 清为“0”。

备注 n = 0 至 7

(11) 定时器输出电平寄存器 0（TOL0）

TOL0 是控制各个通道的定时器输出电平的寄存器。

当在组合操作模式（TOM0n = 1）中启动定时器输出（TOE0n = 1）时，由该寄存器进行的通道 n 的反向输出设置在定时器输出信号的设置或复位时序处得到反映。 在切换模式（TOM0n = 0）中，该寄存器的设置无效。

TOL0 可由 16 位存储器操作指令设置。

TOL0 的低八位可使用 TOL0L 由 8 位存储器操作指令设置。

产生复位信号将该寄存器清除为 0000H。

图 6-19. 定时器输出电平寄存器 0（TOL0）的格式

地址: F01BCH, F01BDH 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TOL0	0	0	0	0	0	0	0	0	TOL 07	TOL 06	TOL 05	TOL 04	TOL 03	TOL 02	TOL 01	TOL 00

TOL 0n	通道n定时器输出电平的控制														
0	正逻辑输出（高有效）														
1	反相输出（低有效）														

- 注意事项** 务必将位 15 至位 8 清为“0”。
- 备注**
1.

如果在定时器操作期间重写该寄存器的值，则当定时器输出信号下一次改变时定时器输出反相，而不是在重写寄存器值之后立即输出反相。
2.

n = 0 至 7

(12) 定时器输出模式寄存器 0 (TOM0)

TOM0 用于控制各个通道的定时器输出模式。

当一个通道用于单一操作功能时，将该通道的对应位设置为 0。

当一个通道用于组合操作功能（PWM 输出、单次脉冲输出或多次 PWM 输出）时，将主通道的对应位设置为 0，并将从通道的对应位设置为 1。

当使能定时器输出（TOE0n = 1）时，由该寄存器进行的各个通道 n 的设置 in 设置或复位定时器输出信号的时序中得到反映。

TOM0 可由 16 位存储器操作指令设置。

TOM0 的低八位可使用 TOM0L 由 8 位存储器操作指令设置。

产生复位信号将该寄存器清除为 0000H。

图 6-20. 定时器输出模式寄存器 0 (TOM0) 的格式

地址: F01BEH, F01BFH 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TOM0	0	0	0	0	0	0	0	0	TOM07	TOM06	TOM05	TOM04	TOM03	TOM02	TOM01	TOM00

TOM0n	通道n定时器输出模式的控制
0	切换操作模式（通过定时器中断请求信号（INTTM0n）产生切换输出）
1	组合操作模式（输出由主通道的定时器中断请求信号（INTTM0n）设置；复位由从通道的定时器中断请求信号（INTTM0m）设置）

注意事项 务必将位 15 至位 8 清为“0”。

备注 n: 通道编号，m: 从通道编号
 n = 0 至 7 （对于主通道，n = 0、2、4、6）
 n < m ≤ 7 （m 是大于 n 的连续整数）

(13) 输入开关控制寄存器（ISC）。

ISC 寄存器的 ISC1 和 ISC0 位用于使用通道 7 实现 LIN 总线通信操作与串行阵列部件的结合。 当将 ISC1 位设为 1 时，选择串行数据输入引脚（RxD0）的输入信号作为定时器输入信号。

当将 P52/SLTI/SLTO 引脚选作为定时器通道 0 和 1 的定时器 I/O 引脚时，设置 ISC2 位。

ISC 可由 1 位或 8 位存储器操作指令设置。

产生复位信号将该寄存器清除为 00H。

图 6-21. 输入转换控制寄存器（ISC）的格式

地址: FFF3CH 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
ISC	0	0	0	0	0	ISC2	ISC1	ISC0

ISC2	选择 P52/SLTI/SLTO 引脚作为定时器 I/O 引脚			
	通道 0		通道 1	
	输入引脚	输出引脚	输入引脚	输出引脚
0	P00/TI00 [#]	P01/TO00 [#]	P52/SLTI	P52/SLTO
1	P52/SLTI	P52/SLTO	—	—
其它	禁止设置			

ISC1	转换定时器阵列部件TAUS的通道7的输入
0	使用TI07引脚的输入信号作为定时器输入（正常操作）。
1	使用RxD0引脚的输入信号作为定时器输入（唤醒信号检测。）

ISC0	转换外部中断（INTP0）输入
0	使用INTP0引脚的输入信号作为外部中断（正常操作）。
1	使用RxD0引脚的输入信号作为外部中断 (以测量同步中断域和同步域的脉冲宽度)。

注 仅限于 78K0R/KD3-L 和 78KR/KE3-L。 在 78K0R/KC3-L 中，仅 P52/SLTI/SLTO 引脚可分配至通道 0 和 1。

注意事项 务必将位 7 至位 3 清为“0”。

备注 当使用 LIN 总线通信功能时，通过将 ISC1 设为 1 选择 RxD0 引脚的输入信号。

(14) 噪声过滤器使能寄存器 1、2 (NFEN1、NFEN2)

NFEN1 和 NFEN2 用于设置噪声过滤器能否用于各个通道的定时器输入信号。

通过在需要噪声抑止器的引脚上将对应位置为 1，来启动噪声过滤器。

当噪声过滤器为 ON 时，在 CPU/外围硬件时钟 (f_{CLK}) 下执行 2 个时钟的匹配检测和同步。当噪声过滤器为 OFF 时，仅在 CPU/外围硬件时钟 (f_{CLK}) 下执行同步。

NFEN1 和 NFEN2 可由 1 位或 8 位存储器操作指令设置。

产生复位信号将该寄存器清除为 00H。

图 6-22. 噪声过滤器使能寄存器 1、2（NFEN1、NFEN2）的格式

地址: F0061H

复位后: 00H

R/W

符号	7	6	5	4	3	2	1	0
NFEN1	TNFEN07	TNFEN06	TNFEN05	TNFEN04	TNFEN03	TNFEN02	0	TNFEN00 ^{#1}

地址: F0062H

复位后: 00H

R/W

符号	7	6	5	4	3	2	1	0
NFEN2	0	0	0	TNFENSL	0	0	0	0

TNFEN07	使用TI07/TO07/P15 (P51)引脚 ^{#2} 或RxD0/P74引脚输入信号 ^{#3} 的噪声过滤器使能/禁止
0	噪声过滤器为 OFF
1	噪声过滤器为 ON

TNFEN06	使用TI06/TO06/P14 (P50)引脚输入信号 ^{#2} 的噪声过滤器使能/禁止
0	噪声过滤器为 OFF
1	噪声过滤器为 ON

TNFEN05	使用TI05/TO05/P13引脚输入信号的噪声过滤器使能/禁止
0	噪声过滤器为 OFF
1	噪声过滤器为 ON

TNFEN04	使用TI04/TO04/RTCDIV/RTCCL/P引脚输入信号的噪声过滤器使能/禁止
0	噪声过滤器为 OFF
1	噪声过滤器为 ON

TNFEN03	使用TI03/TO03/P11引脚输入信号的噪声过滤器使能/禁止
0	噪声过滤器为 OFF
1	噪声过滤器为 ON

TNFEN02	使用TI02/TO02/P10引脚输入信号的噪声过滤器使能/禁止
0	噪声过滤器为 OFF
1	噪声过滤器为 ON

TNFEN00 ^{#1}	使用TI00/P00引脚输入信号的噪声过滤器使能/禁止
0	噪声过滤器为 OFF
1	噪声过滤器为 ON

TNFENSL	使用SLTI/SLTO/P52引脚输入信号的噪声过滤器使能/禁止
0	噪声过滤器为 OFF
1	噪声过滤器为 ON

注 1. 仅限于 78K0R/KD3-L 和 78K0R/KE3-L

2. 在非 78K0R/KE3-L 产品中，TI06/TO06 和 TI07/TO07 分别与 P50 和 P51 复用。

3. 通过设置 ISC 寄存器的 ISC1 可转换可用的引脚。

ISC1 = 0: 可以选择是否使用 TI07 引脚的噪声过滤器。

ISC1 = 1: 可以选择是否使用 RxD0 引脚的噪声过滤器。

(15) 端口模式寄存器 0[※]、1、5（PM0[※]、PM1、PM5）

这些寄存器以 1 位为单位设置端口 0[※]、1 和 5 的输入/输出。
当使用将与定时器输出引脚共用的端口（例如，P01/TO00 和 P10/TO02/TI02）作为定时器输出时，将对应于各通道端口模式寄存器（PMxx）位和端口寄存器（Pxx）位设为 0。

示例：当 P10/TO02/TI02 用于定时器输出时
将端口模式寄存器 1 的 PM10 位设为 0。
将端口寄存器 1 的 P10 位设为 0。

当使用将与定时器输出引脚共用的端口（例如，P00/TI00 和 P10/TO02/TI02）作为定时器输入时，将对应于各通道的端口模式寄存器（PMxx）位设为 1。此时，端口寄存器（Pxx）位可能为 0 或 1。

示例：将 P10/TO02/TI02 用于定时器输入时
将端口模式寄存器 1 的 PM10 位设为 1。
将端口寄存器 1 的 P10 位设为 0 或 1。

PM0[※]、PM1 和 PM5 可由 1 位或 8 位存储器操作指令设置。
产生复位信号将这些寄存器设为 FFH。

注 仅限于 78K0R/KD3-L 和 78K0R/KE3-L

图 6-23. 端口模式寄存器 0、1、5（PM0、PM1、PM5）的格式

地址: FFF20H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PM0	1	1	1	1	1	1	PM01	PM00

地址: FFF21H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PM1	PM17	PM16	PM15	PM14	PM13	PM12	PM11	PM10

地址: FFF25H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PM5	1	1	1	1	PM53	PM52	PM51	PM50

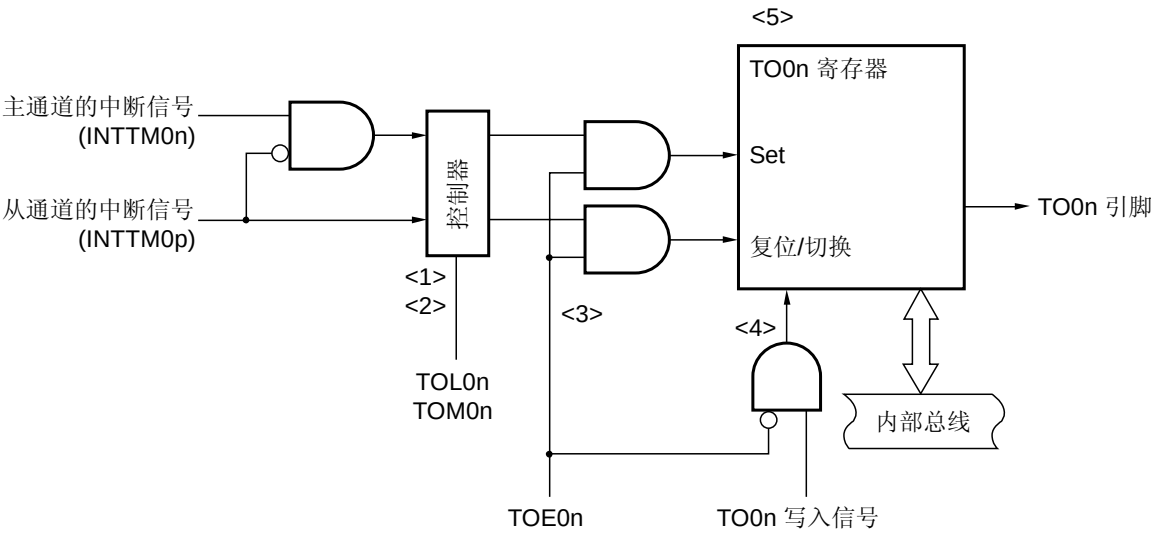
PMmn	Pmn 引脚 I/O 模式选择 (m = 0、1、5; n = 0 至 7)
0	输出模式 (输出缓存器开启)
1	输入模式 (输出缓存器关闭)

备注 上图显示了 78K0R/KE3-L 产品的端口模式寄存器 0、1 和 5 的格式。有关其它产品的端口模式寄存器的格式，请参见 4.3 控制端口功能的寄存器 中的 (1) 端口模式寄存器（PMxx）。

6.4 通道输出（TO0n 引脚）控制

6.4.1 TO0n 引脚输出电路配置

图 6-24. 输出电路配置



以下说明 TO0n 引脚的输出电路。

- <1> 当 TOM0n = 0（切换模式）时，忽略 TOL0n 寄存器的设置值并仅将 INTTM0p（从通道定时器中断）发送至 TO0n 寄存器。
- <2> 当 TOM0n = 1（组合操作模式）时，将 INTTM0n（主通道定时器中断）和 INTTM0p（从通道定时器中断）都发送至 TO0n 寄存器。
此时，TOL0n 寄存器变为有效且按照以下控制信号。

当 TOL0n = 0 时：正向操作（INTTM0 → 置位，INTTM0p → 复位）
当 TOL0n = 1 时：反向操作（INTTM0 → 复位，INTTM0p → 置位）

当同时产生 INTTM0n 和 INTTM0p 时，（PWM 的 0% 输出），INTTM0p（复位信号）优先，并屏蔽 INTTM0n（置位信号）。

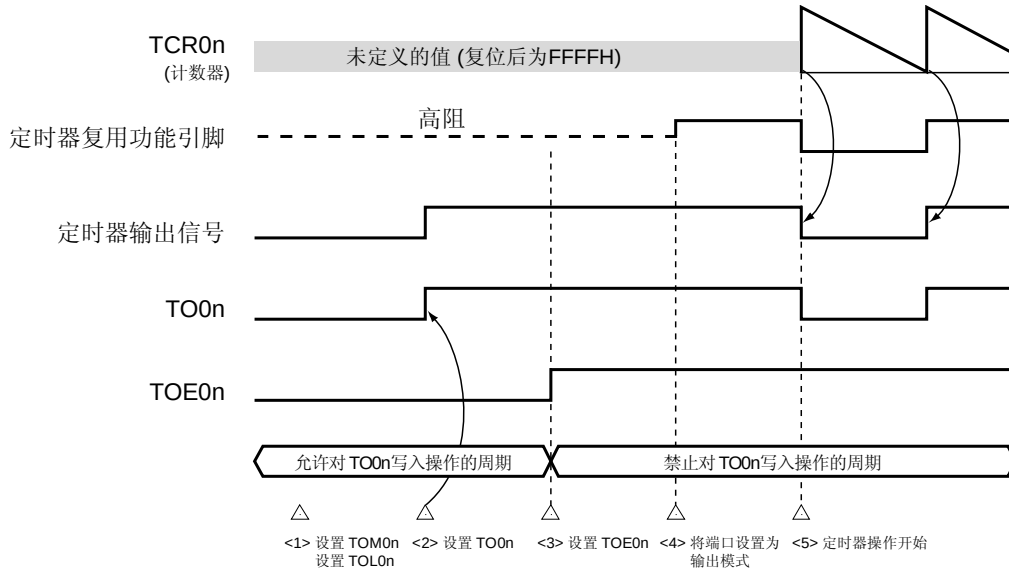
- <2> 当 TOE0n = 1 时，将 INTTM0n（主通道定时器中断）和 INTTM0p（从通道定时器中断）发送至 TO0n 寄存器。写入 TO0n 寄存器（TO0n 写入信号）变为无效。
当 TOE0n = 1 时，TO0n 引脚输出决不随除了中断信号之外的信号改变。
要初始化 TO0n 引脚的输出电平，有必要设置 TOE0n = 0 并向 TO0n 写入一个值。
- <4> TOE0n = 0 时，写入目标通道的 TO0n 位（TO0n 写入信号）变为有效。当 TOE0n = 0 时，INTTM0n（主通道定时器中断）和 INTTM0p（从通道定时器中断）都不发送至 TO0n 寄存器。
- <5> 总是可以读取 TO0n 寄存器，并可以检查 TO0n 引脚的输出电平。

备注 1. n = 0 至 7（对于主通道，n = 0、2、4 或 6）
2. p = n + 1、n + 2、n + 3 ...（p ≤ 7）

6.4.2 TO0n 引脚输出设置

下图显示了 TO0n 输出引脚从初始设置到定时器操作开始的步骤和状态转换。

图 6-25. 从定时器输出设置到操作开始的状态转换



<1> 设置定时器输出的操作模式。

- TOM0n 位 (0: 切换模式, 1: 组合操作模式)
- TOL0n 位 (0: 正向输出, 1: 反向输出)

<2> 通过设置 TO0n 将定时器输出信号设置至初始状态。

<3> 通过将 1 写入 TOE0n 使能定时器输出操作 (禁止写入 TO0n)。

<4> 将端口 I/O 设置设置至输出 (参见 6.3 (15) 端口模式寄存器 0、1、5)。

<5> 使能定时器操作 (TS0n = 1)。

备注 n = 0 至 7

6.4.3 通道输出操作时的注意事项

(1) 定时器操作期间改变寄存器 TO0、TOE0、TOL0 和 TOM0 中的值

因为定时器操作 (TCR0n 和 TDR0n 的操作) 不依赖于 TO0n 输出电路, 且改变在 TO0、TOE0、TOL0 和 TOM0 中设置的值不影响定时器操作, 所以可在定时器操作期间改变值。但是, 要通过定时器操作从 TO0n 引脚输出一个预期的波形, 则需将 TO0、TOE0、TOL0 和 TOM0 设为在各个操作的设置示例的寄存器中规定的值。

当将 TOE0、TOL0 和 TOM0 (除 TO0) 中设置的值变为接近于定时器中断 (INTTM0n) 时, 根据值在定时器中断 (INTTM0n) 信号产生时刻之前的瞬间还是产生后的瞬间改变, 输出至 TO0n 引脚的波形可能会有所不同。

备注 n = 0 至 7

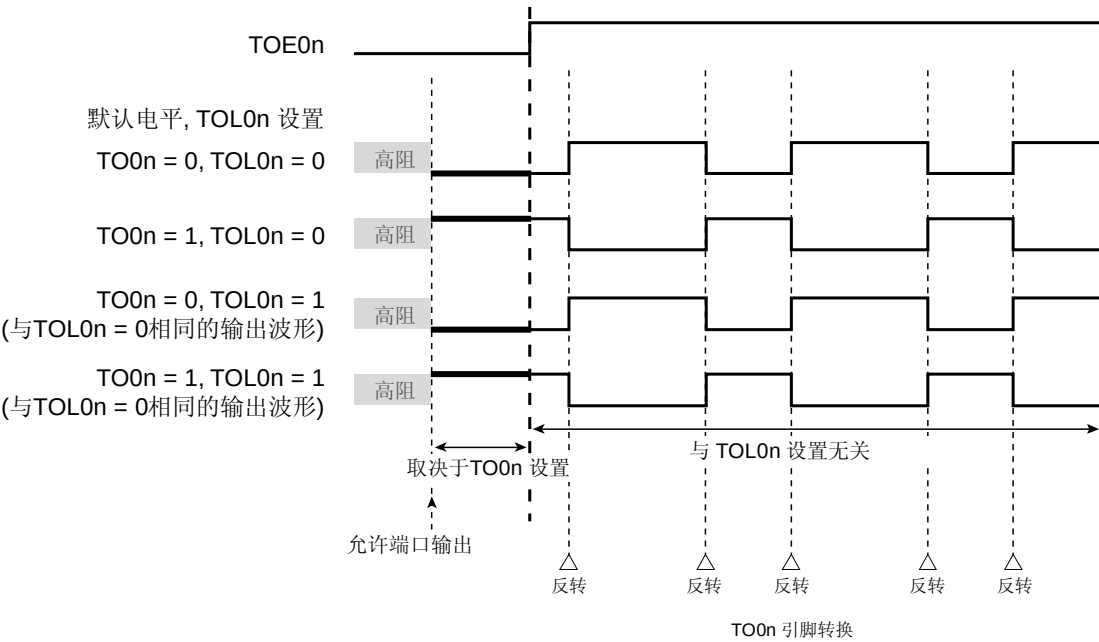
(2) TO0n 引脚的默认电平和定时器操作开始后的输出电平

下图显示了当在使能端口前 $TOE0n = 0$ 且改变默认电平后设置 $TOE0n = 1$ 状态下已经做完成写入时 TO0n 引脚的输出电平转换。

(a) 通过 $TOM0n = 0$ 设置开始操作时（切换输出）

$TOM0n = 0$ 时， $TOL0n$ 的设置无效。当设置默认电平之后开始定时器操作时，产生切换信号且 TO0n 引脚的输出电平反转。

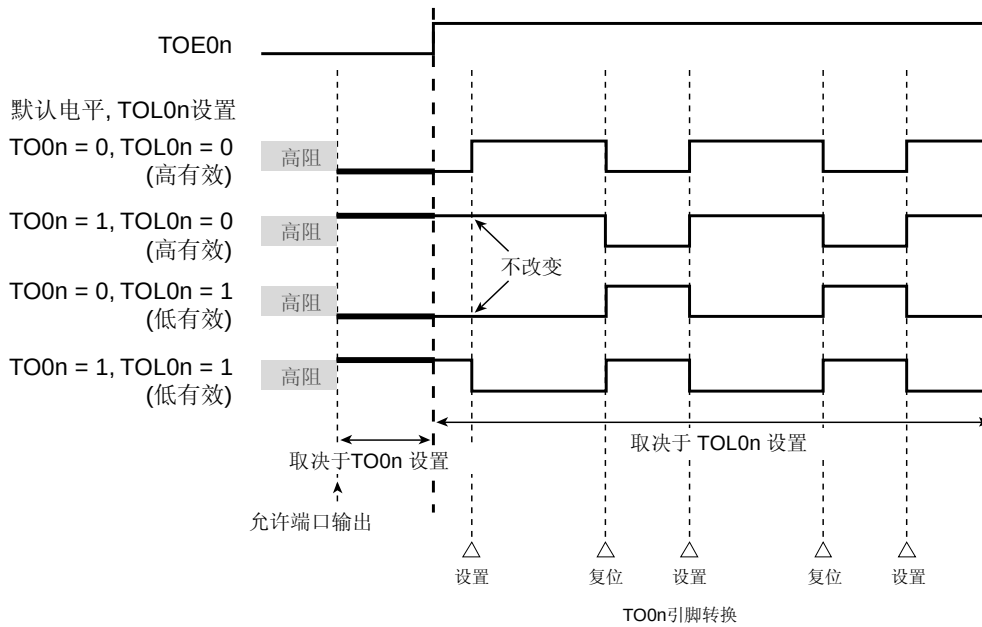
图 6-26. TO0n 引脚在切换输出（ $TOM0n = 0$ ）时的输出状态



备注 1. 反转：反转 TO0n 引脚输出状态
2. n = 0 至 7

(b) 通过 $TOM0n = 1$ 设置开始操作时（组合操作模式（PWM 输出））

当 $TOM0n = 1$ 时，有效电平取决于 $TOL0n$ 设置。

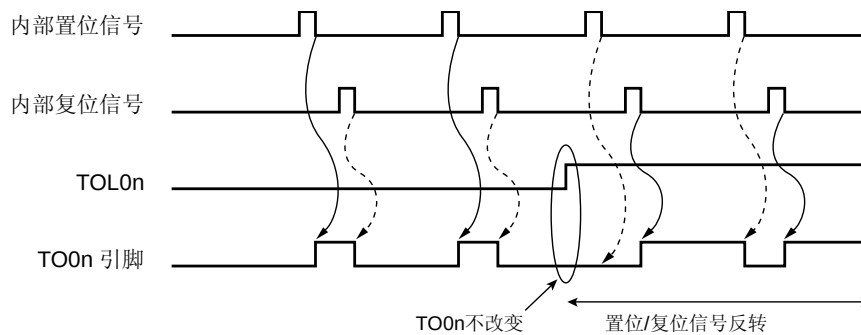
图 6-27. $TO0n$ 引脚在 PWM 输出（ $TOM0n = 1$ ）时的输出状态

- 备注
1. 置位: $TO0n$ 引脚的输出信号由无效电平变为有效电平。
复位: $TO0n$ 引脚的输出信号由有效电平变为无效电平。
 2. $n = 0$ to 7

(3) 组合操作模式下 $TO0n$ 引脚的操作（ $TOM0n = 1$ ）(a) 当定时器操作期间已改变了 $TOL0n$ 设置时

当定时器操作期间已改变了 $TOL0n$ 设置时，设置在 $TO0n$ 改变条件产生时变为有效。重写 $TOL0$ 不会改变 $TO0n$ 的输出电平。

下图显示了定时器操作（ $TOM0n = 1$ ）期间改变了 $TOL0n$ 的值时的操作。

图 6-28. 定时器操作期间改变了 $TOL0n$ 时的操作

- 备注
1. 置位: $TO0n$ 引脚的输出信号由无效电平变为有效电平。
复位: $TO0n$ 引脚的输出信号由有效电平变为无效电平。
 2. $n = 0$ 至 7

(b) 置位/复位时序

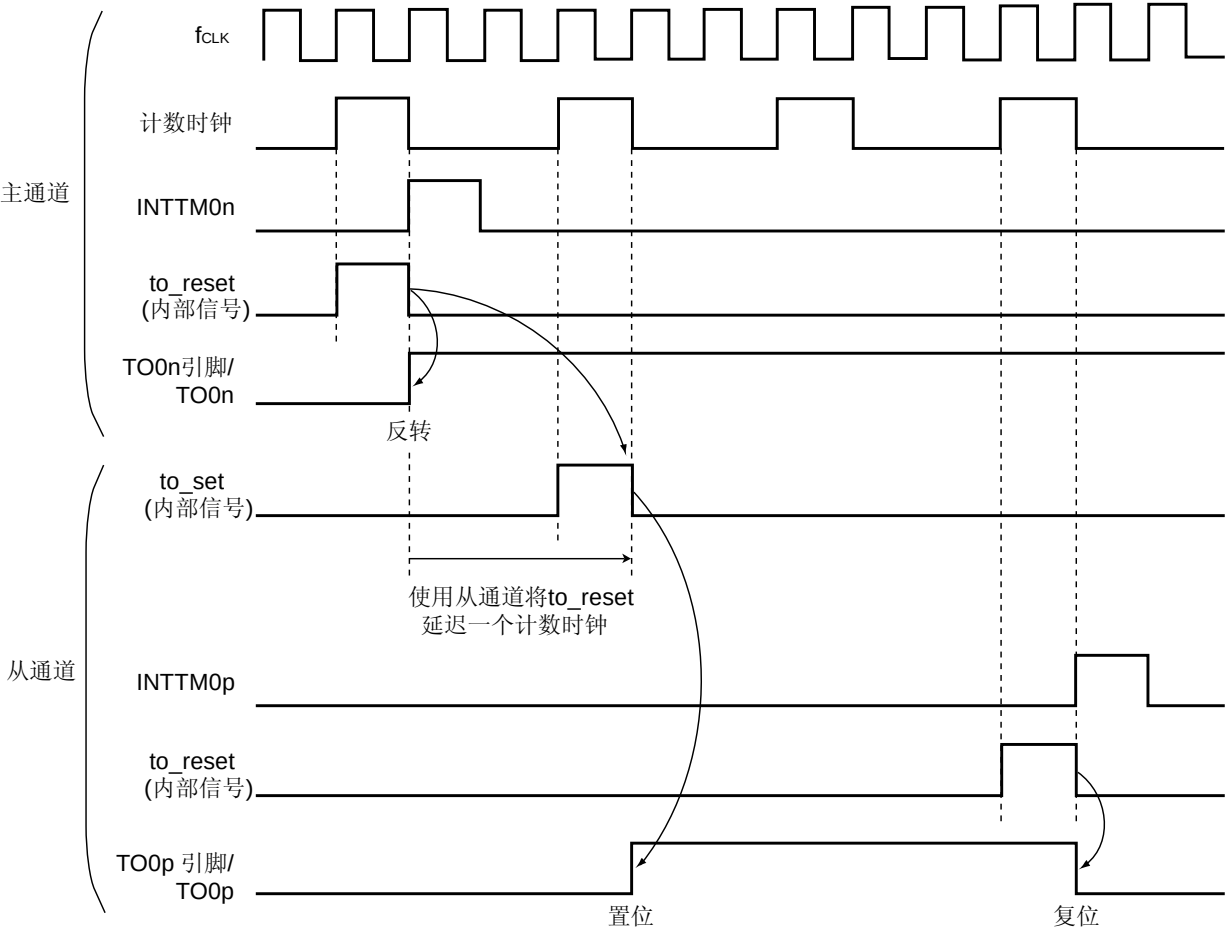
为了在 PWM 输出时实现 0%/100%输出，需通过从通道将产生主通道定时器中断（INTTM0n）处的 TO0n 引脚/TO0n 设置时序延时一个计数时钟。

如果同时产生置位条件和复位条件，则赋予后者较高的优先级。

图 6-29 显示了主通道/从通道按照以下设置时的置位/复位操作状态。

主通道： TOE0n = 1、TOM0n = 0、TOL0n = 0
从通道： TOE0p = 1、TOM0p = 1、TOL0p = 0

图 6-29. 置位/复位时刻的操作状态

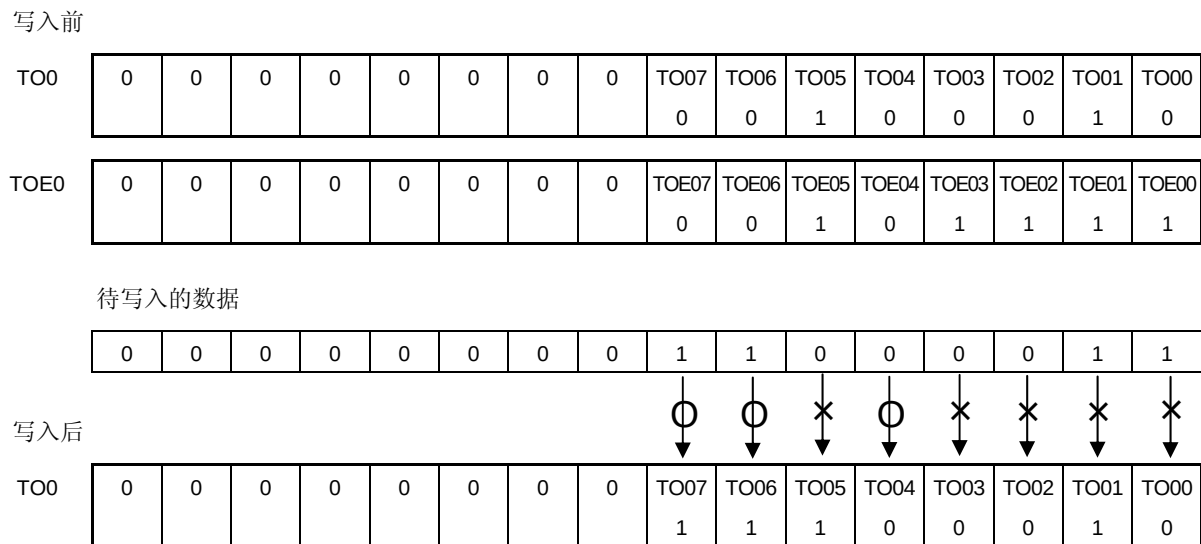


- 备注
- 1. to_reset: TO0n 引脚复位/切换信号
to_set: TO0n 引脚置位信号
 - 2. n = 0 至 7 （对于主通道，n = 0、2、4 或 6）
 - 3. p = n+1、n+2、n+3 ... (p ≤ 7)

6.4.4 TO0n 位的集合操作

在 TO0 寄存器中，可按照与 TS0 寄存器（通道开始触发器）相同的方式把所有通道的设置位设置在一个寄存器中。因此，可以共同地操作所有通道的 TO0n。仅指定的位还可以通过将相应的 TOE0n = 0 设置至目标 TO0n（通道输出）进行操作。

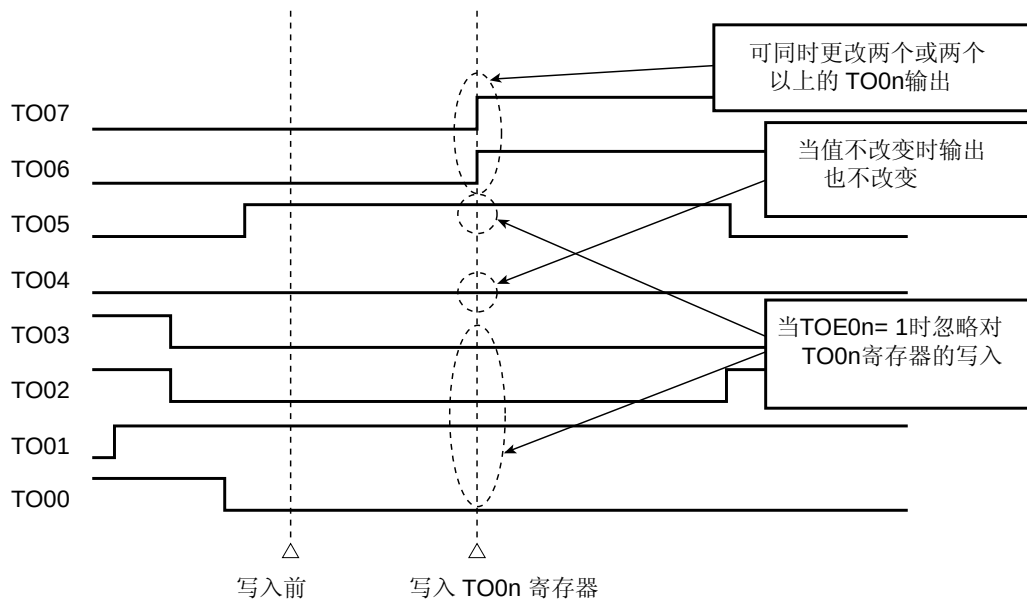
图 6-30. TO0n 位集合操作的示例



TOE0n = 0 时仅完成对 TO0n 位的写入，并忽略 TOE0n = 1 时对 TO0n 位的写入。

设置了 TOE0n = 1 的 TO0n（通道输出）不受写入操作的影响。即使对 TO0n 进行写入操作，写入操作也会被忽略并正常进行通过定时器操作的输出变化。

图 6-31. 通过 TO0n 位的集合操作 TO0n 的引脚状态



（注意事项 和 备注在下页给出）。

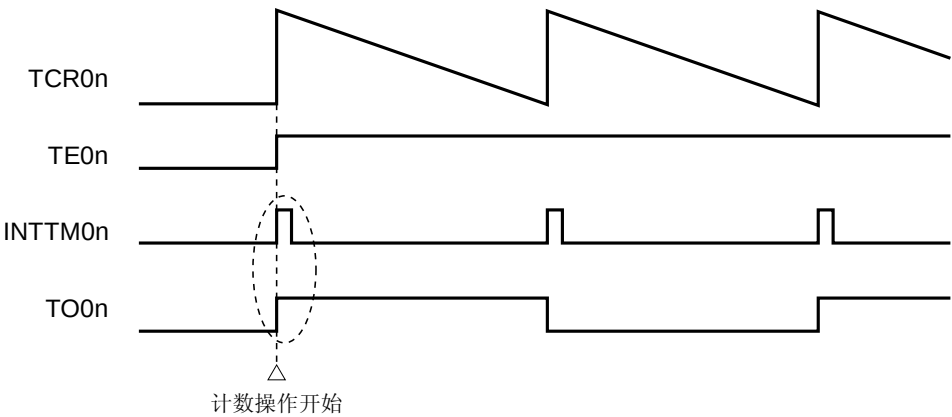
注意事项 当 $TOE0n = 1$ 时，即使各个定时器的定时器中断（ $INTTM0n$ ）的输出都争取写入 $TO0n$ ，仍然向 $TO0n$ 引脚做出输出。

备注 $n = 0$ 至 7

6.4.5 操作开始时的定时器中断和 $TO0n$ 引脚输出

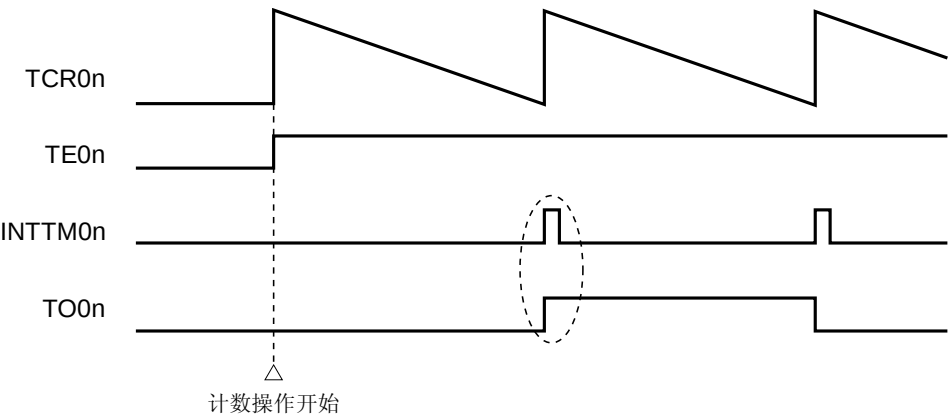
在间隔定时器模式或捕获模式下， $TMR0n$ 寄存器中的 $MD0n0$ 位设置是否在计数开始时产生定时器中断。
当将 $MD0n0$ 设为 1 时，可通过产生定时器中断（ $INTTM0n$ ）的得知计数操作开始的时刻。
在其它模式下，计数操作开始处的定时器中断和 $TO0n$ 输出都不受控制。
图 6-32 和 6-33 显示了设为间隔定时器模式（ $TOE0n = 1$ ， $TOM0n = 0$ ）时的操作示例。

图 6-32. 当将 $MD0n0$ 设为 1 时



当 $MD0n0$ 设为 1 时，在计数操作开始处输出一个定时器中断（ $INTTM0n$ ），且 $TO0n$ 执行切换操作。

图 6-33. 当将 $MD0n0$ 设为 0 时



当 $MD0n0$ 设为 0 时，在计数操作开始处不输出定时器中断（ $INTTM0n$ ）， $TO0n$ 也不发生改变。计数一个周期后，输出 $INTTM0n$ 且 $TO0n$ 进行切换操作。

备注 $n = 0$ 至 7

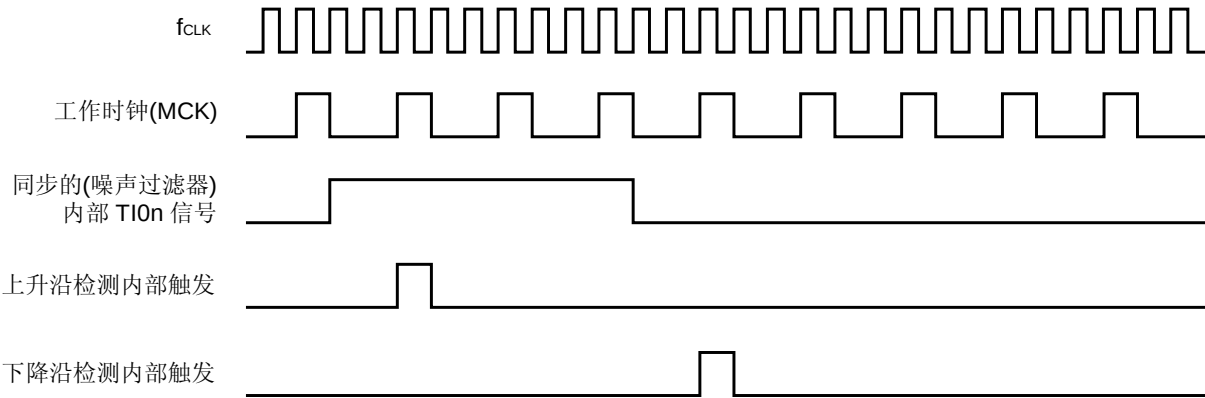
6.5 通道输入（TI0n 引脚）控制

6.5.1 TI0n 边沿检测电路

(1) 边沿检测基本操作时序

边沿检测电路依照工作时钟（MCK）进行采样。

图 6-34. 边沿检测基本操作时序



备注 n = 0 至 7

6.6 定时器阵列部件 TAUS 的基本功能

6.6.1 单一操作功能和组合操作功能的概述

定时器阵列部件 TAUS 由若干通道组成，具有单一操作功能和组合操作功能。其中单一操作功能允许各个通道独立操作，而组合操作功能使用两个或多个通道一起操作。

单一操作功能可用于任何通道，而与其它通道的操作模式无关。

组合操作功能通过联合主通道（主要计数周期的参考定时器）和从通道（按照主通道运行的定时器）实现，且使用该功能时必须遵循一些规则。

6.6.2 组合操作功能的基本规则

使用组合操作功能的基本规则如下。

- (1) 仅偶通道（通道 0、2、4 等）可被设为主通道。
- (2) 除了通道 0 之外的任何通道都可设为从通道。
- (3) 从通道必须低于主通道。

示例：如果通道 2 被设为主通道，则通道 3 或之后的通道（通道 3、4、5 等）可被设为从通道。

- (4) 可为一个主通道设置两个或多个从通道。
- (5) 要使用两个或多个主通道时，则它们当中的主通道之一的从通道可能没有设置。

示例：如果通道 0 和 4 被设为主通道，则通道 1 至 3 可被设为主通道 0 的从通道。通道 5 至 7 不能设为主通道 0 的从通道。

- (6) 与主通道一起组合操作的从通道的操作时钟必须与主通道的操作时钟相同。与主通道一起组合操作的从通道的 CKS 位（TMR0n 寄存器的位 15）的值必须与主通道的 CKS 位的值相同。
- (7) 主通道能够发送 INTTM0n（中断）、开始软件触发并计数较低通道的时钟。
- (8) 从通道能够使用 INTTM0n（中断）、开始软件触发并计数主通道的时钟，但是它不能发送自身的 INTTM0n（中断）、开始软件触发且不能计数较低通道的时钟。
- (9) 主通道不能发送 INTTM0n（中断）、开始软件触发且不能计数来自较高的主通道的时钟。
- (10) 要同时启动组合操作的通道，必须在同时设置组合通道的 TS0n 位。
- (11) 要同时停止组合的通道，必须在同时设置组合通道的 TT0n 位。

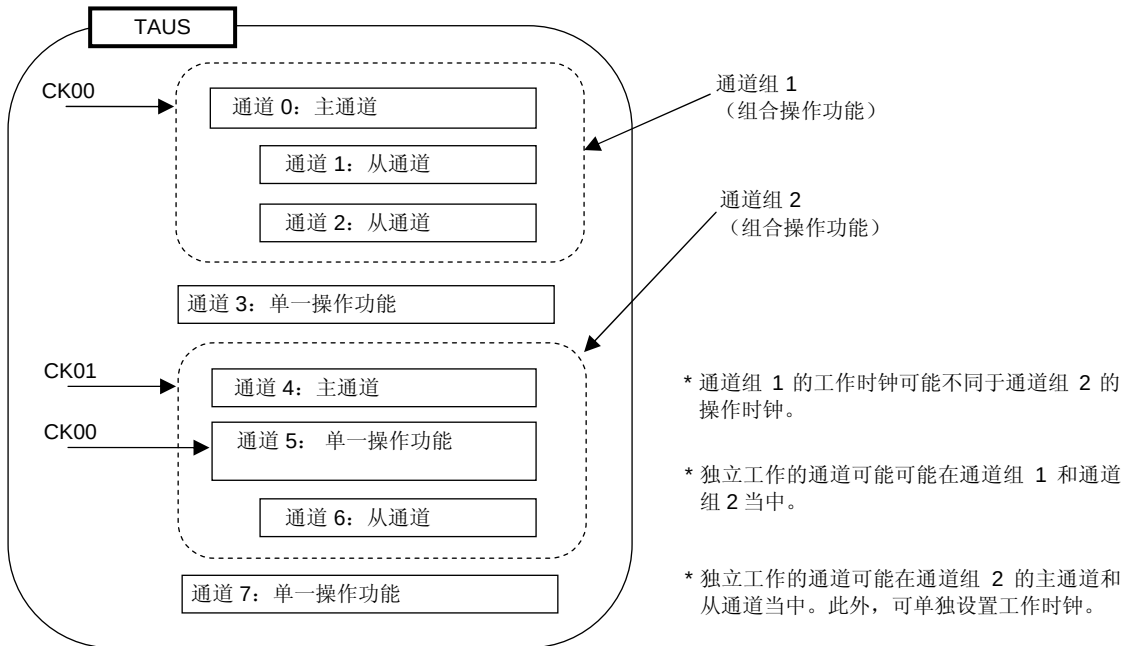
备注 n = 0 至 7

6.6.3 组合操作功能基本规则的可用范围

组合功能的规则适用于一个通道组（组成组合操作功能的一个主通道和从通道）。

如果指定了两个或多个不组合操作的通道组，则 **6.6.2 组合操作功能的基本规则** 中的组合操作功能的基本规则不适用于这些通道组。

示例



6.7 定时器阵列部件 TAUS 作为独立通道的操作

6.7.1 作为间隔定时器/方波输出的操作

(1) 间隔定时器

定时器阵列部件可用作按固定间隔生成中断 INTTM0n（定时器中断）的参考定时器。
中断产生周期可用下式计算。

$$\text{INTTM0n (定时器中断) 的生成周期} = \text{计数时钟的周期} \times (\text{TDR0n 的设置值} + 1)$$

<R> 除了 CK00 和 CK01 之外，还可以选择被 4 分频的副系统时钟（fSUB/4）作为计数时钟。 因此，间隔定时器可以使用固定为 fSUB/4 的计数时钟工作，而不考虑 fCLK 频率（主系统时钟、副系统时钟）。 然而，改变选为 fCLK 的时钟（改变系统时钟控制寄存器（CKC）的值）时，应首先停止定时器阵列部件 TAUS（TT0 = 00FFH）。

(2) 作为方波输出的操作

一旦产生了 INTTM0n，TO0n 会立即进行切换操作并输出占空比为 50% 的方波。
从 TO0n 输出方波的周期和频率可用下式计算。

- 从 TO0n 输出的方波的周期 = 计数时钟的周期 × (TDR0n 的设置值 + 1) × 2
- 从 TO0n 输出的方波频率 = 计数时钟的频率 / (TDR0n 的设置值 + 1) × 2

TCR0n 在间隔定时器模式下作为向下计数器工作。
通道开始触发位（TS0n）设为 1 之后，TCR0n 载入 TDR0n 在第一个计数时钟处的值。 如果 TMR0n 的 MD0n0 = 1，则输出 INTTM0n 并切换 TO0n。
之后，TCR0n 与计数时钟同步向下计数。
当 TCR0n = 0000H 时，输出 INTTM0n 并在下一个计数时钟处切换 TO0n。 同时，TCR0n 再次载入 TDR0n 的值。 之后，重复相同的操作。
TDR0n 可在任何时间重写。 TDR0n 的新值从下一个周期变为有效。

图 6-35. 作为间隔定时器/方波输出操作的框图

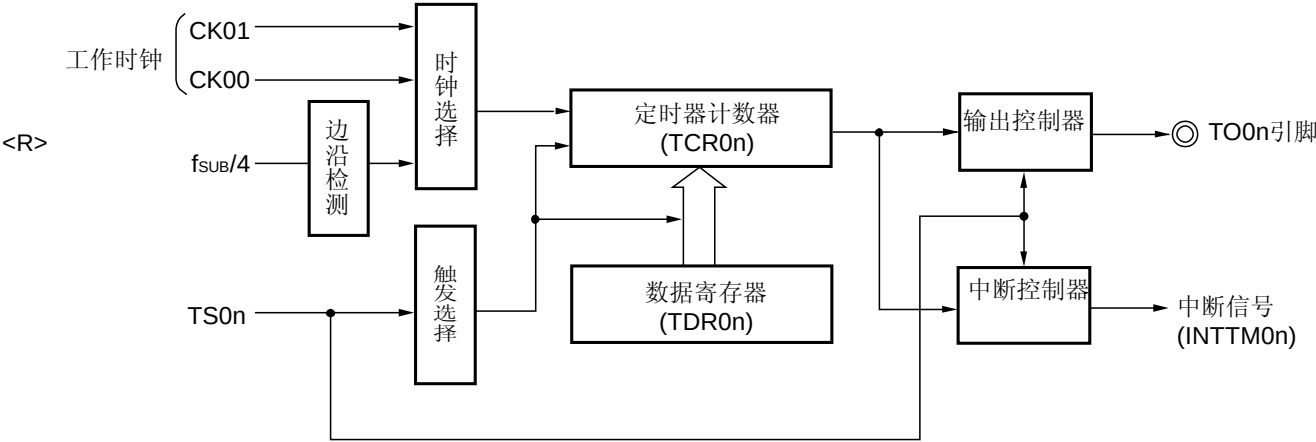
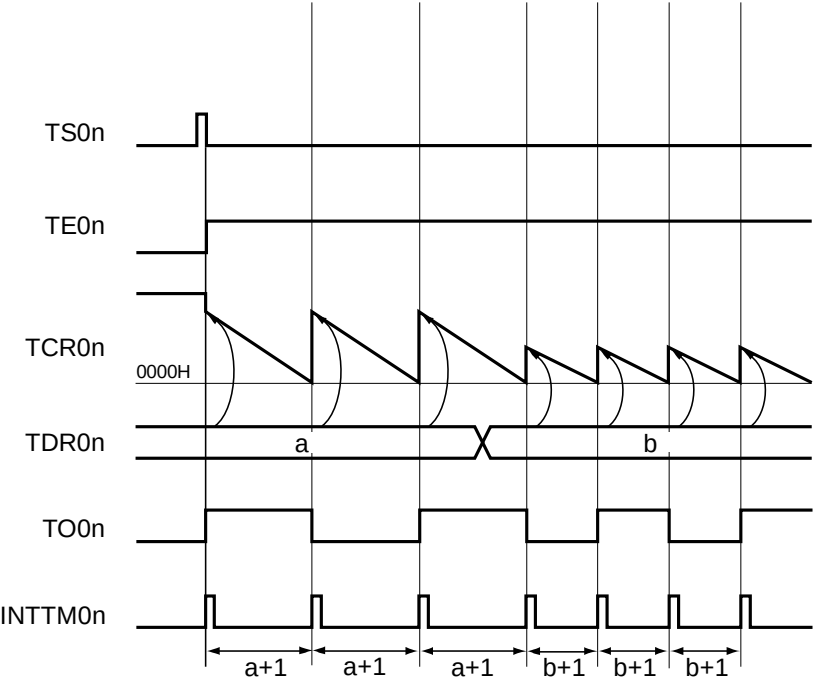


图 6-36. 作为间隔定时器/方波输出操作的基本时序的示例（MD0n0 = 1）



备注 n = 0 至 7

图 6-37. 作为间隔定时器/方波输出操作期间设置寄存器内容的示例（1/3）

(1) 选择 CK00 或 CK01 作为计数时钟时

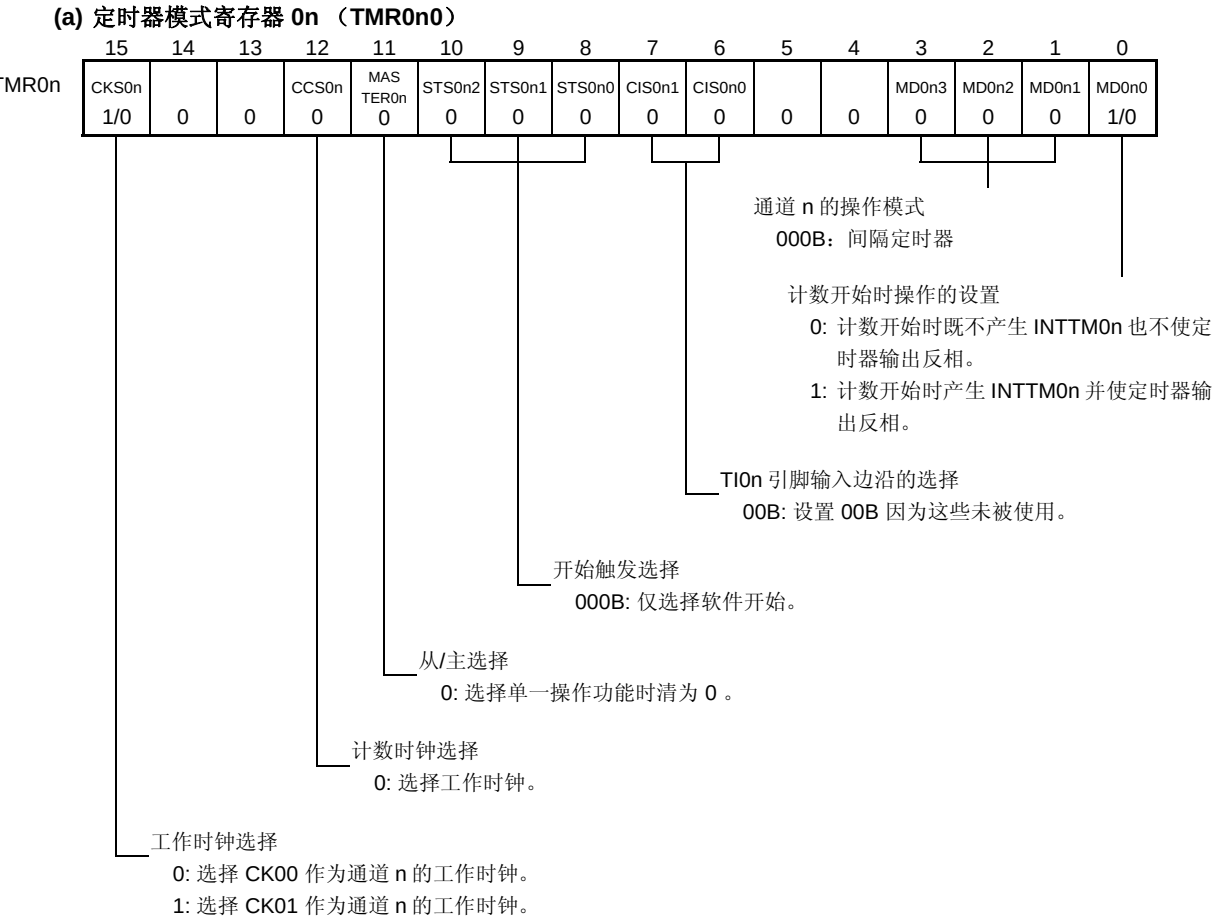
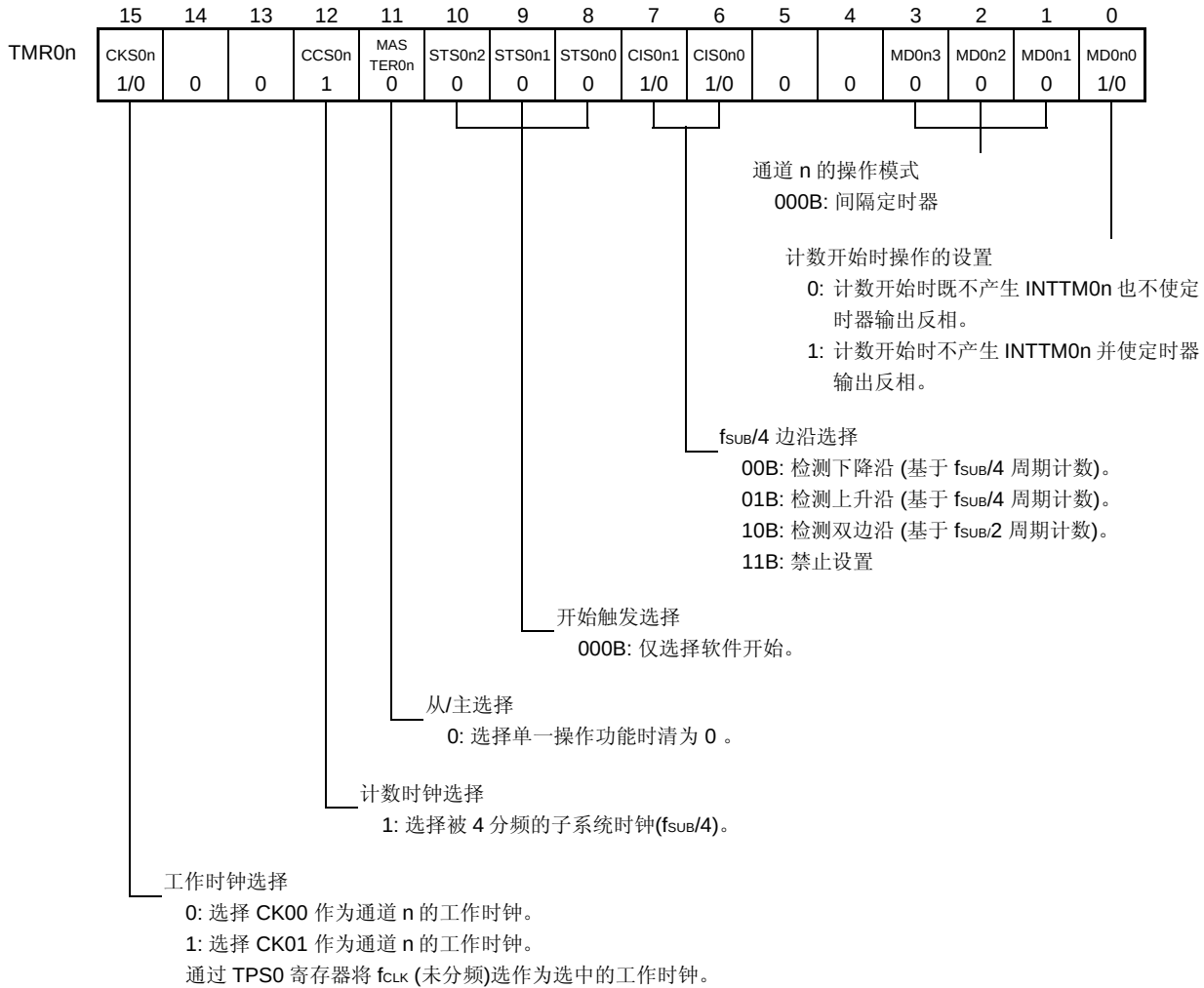


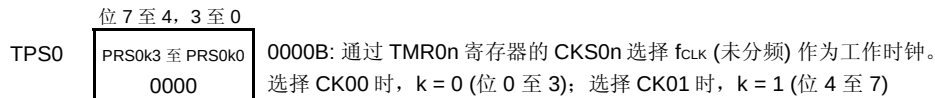
图 6-37. 作为间隔定时器/方波输出操作期间设置寄存器内容的示例 (2/3)

<R> (2) 选择 $f_{SUB}/4$ 作为计数时钟时

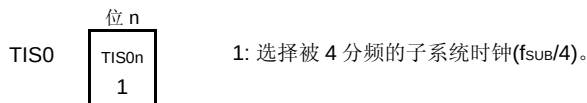
(a) 定时器模式寄存器 0n (TMDR0n)



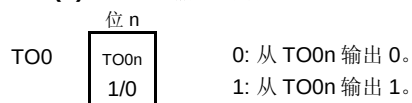
(b) 定时器时钟选择寄存器 0 (TPS0)



(c) 定时器输入选择寄存器 0 (TIS0)



(d) 定时器输出寄存器 0 (TO0)



备注

1. $n = 0$ 至 7, $k = 0、1$
2. f_{SUB} : 子系统时钟振荡频率

图 6-37. 作为间隔定时器/方波输出操作期间设置寄存器内容的示例（3/3）

<R> (2) 选择 fsub/4 作为计数时钟时（连续的）

(e) 定时器输出使能寄存器 0（TOE0）

TOE0	位 n	
	TOE0n	
	1/0	0: 通过计数操作停止 TO0n 输出。 1: 通过计数操作允许 TO0n 输出。

(f) 定时器输出电平寄存器 0（TOL0）

TOL0	位 n	
	TOL0n	
	0	0: 当 TOM0n = 0（切换模式）时清除为 0。

(g) 定时器输出模式寄存器 0（TOM0）

TOM0	位 n	
	TOM0n	
	0	0: 设置切换模式。

备注 n = 0 至 7

图 6-38. 间隔定时器/方波输出功能的操作步骤

		软件操作	硬件状态
<R>	TAUS 默认设置		电源关闭状态 (停止提供时钟并禁止写入每个寄存器。)
		将PER2寄存器的TAU0EN位设为1。	电源打开状态 各通道停止工作。 (开始提供时钟并允许写入每个寄存器。)
		设置TPS0寄存器。 确定CK00 和 CK01的时钟频率	
	通道默认设置	设置 TMR0n 寄存器 (确定通道的操作模式)。 当选择 $f_{sub}/4$ 作为计数时钟时将 TIS0n 位设为1 ($f_{sub}/4$) 。 将间隔 (周期)值设置至 TDR0n寄存器。	通道停止操作。 (提供时钟并消耗一定的功率。)
		要使用TO0n输出 将TOM0寄存器的TOM0n位清除为 0 (切换模式)。 将TOL0n 位清除为0。 设置TO0n 位并确定TO0n输出的默认电平。	TO0n引脚进入高阻输出状态。 当端口模式寄存器处于输出模式且端口寄存器为0时, 输出TO0n默认设置电平。
		将 TOE0n 设为 1 并使能TO0n的操作。 将端口寄存器和端口模式寄存器清除为0。	因为通道停止工作, 所以TO0n不改变。 TO0n引脚输出TO0n设置电平。
	操作开始	将 TOE0n 设为1 (仅当操作恢复时)。 将 TS0n 位设为 1。 因为 TS0n 位为触发位所以它自动返回 0 。	TE0n = 1, 计数操作开始。 在计数时钟输入处载入TDR0n的值。 如果TMR0n寄存器的MD0n0位为1则产生INTTM0n, 且TO0n执行切换操作 。
	操作期间	TMR0n寄存器的设置值、TOM0n和TOL0n位不能够改变。 TDR0n寄存器的设置值可被改变。 总是可以读取TCR0n寄存器。 未使用TSR0n寄存器。 TO0 和 TOE0寄存器的设置值可被改变。	计数器(TCR0n)向下计数。 当计数值达到0000H时, 再次将TDR0n的值载入TCR0n并继续计数操作。通过检测TCR0n = 0000H, 产生INTTM0n且TO0n执行切换操作。之后, 重复上述操作。
	操作停止	TT0n位被设为 1。 因为 TT0n 位为触发位, 所以它自动返回 0。	TE0n = 0, 且计数操作停止。 TCR0n保持计数值并停止。 TO0n输出不被初始化而保持当前状态。
		将TOE0n清为并将值设置至TO0n位。	TO0n引脚输出TO0n设置电平。
TAUS 停止	要保持 TO0n 引脚的输出电平 将要被保持的值设置至端口寄存器之后将TO0n位清为 0。 当没必要保持 TO0n 引脚的输出电平时 将端口模式寄存器转换为输入模式。	TO0n引脚输出电平通过端口功能保持。 TO0n 引脚输出电平进入高阻输出状态。	
	PER2寄存器的TAU0EN位被清为 0。	电源关闭状态 初始化所有的电路以及各个通道的SFR。 (TO0n位被清为 0且TO0n引脚被设为端口模式。)	

操作恢复。

备注 n = 0 至 7

6.7.2 作为外部事件计数器的操作

定时器阵列部件可用作外部事件计数器，对 TI0n 引脚检测的有效输入边沿（外部事件）的次数进行计数。当达到指定的计数值时，事件计数器产生一个中断。计数的指定数目可用下式计算。

计数的指定数目 = TDR0n 的设置值 + 1

TCR0n 在事件计数器模式下作为向下计数器工作。

当通道开始触发位（TS0n）被设为 1 时，TCR0 载入 TDR0n 的值。

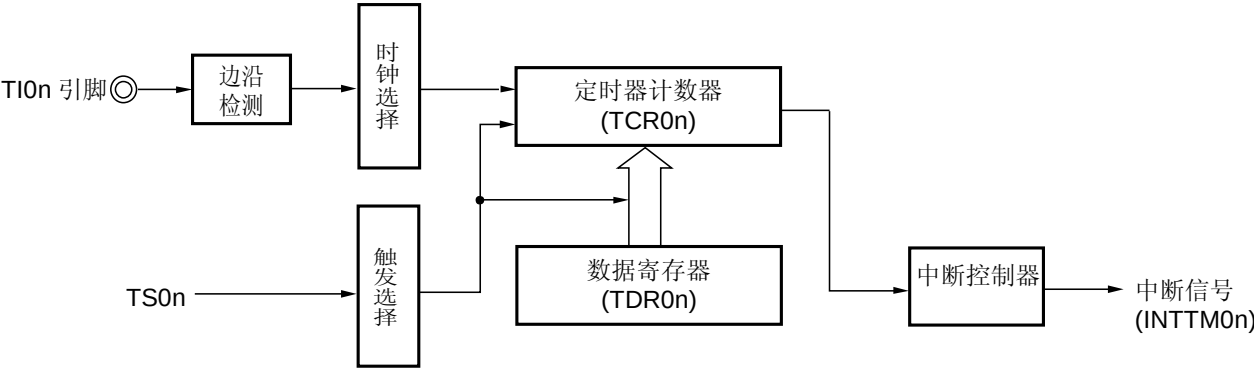
每次检测到 TI0n 引脚的有效输入边沿时，TCR0n 向下计数。当 TCR0n = 0000H 时，TCR0n 再次载入 TDR0n 的值并输出 INTTM0n。

之后，重复以上的操作。

因为 TO0n 取决于外部事件且不规律，所以禁止使用 TO0n。

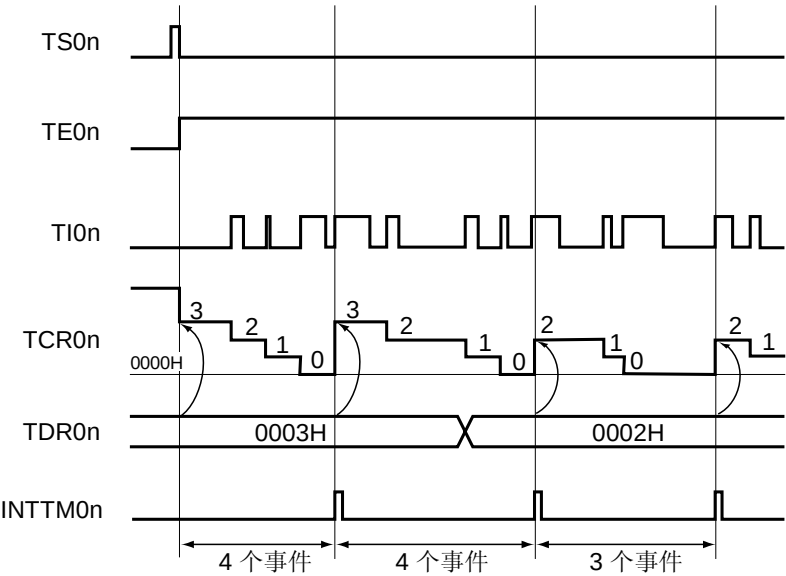
TDR0n 可在任何时间重写。TDR0n 的新值在下一个周期内变为有效。

图 6-39. 外部事件计数的操作框图



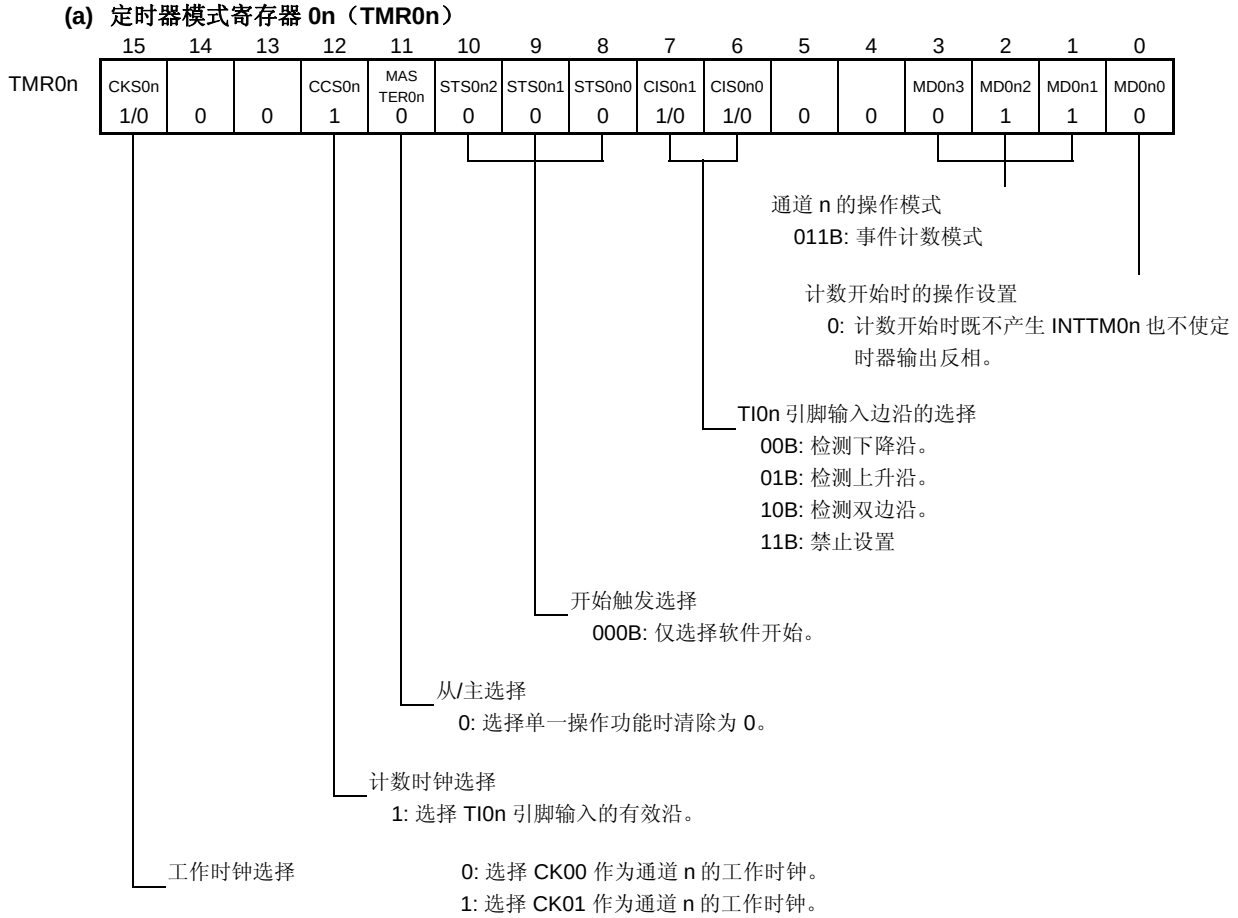
备注 n = 0 至 7

图 6-40. 外部事件计数器操作基本时序的示例

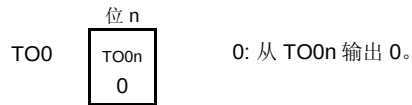


备注 n = 0 至 7

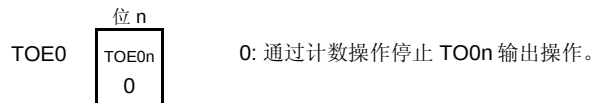
图 6-41. 外部事件计数器模式下寄存器的设置内容的示例



(b) 定时器输出寄存器 0 (TO0)



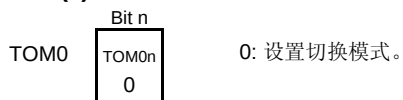
(c) 定时器输出使能寄存器 0 (TOE0)



(d) 定时器输出电平寄存器 0 (TOL0)



(e) 定时器输出模式寄存器 0 (TOM0)



备注 n = 0 至 7

图 6-42. 使用外部时间计数器功能时的操作步骤



备注 n = 0 至 7

6.7.3 作为分频器（仅限于 78K0R/KD3-L 和 78K0R/KE3-L 通道 0）的操作

定时器阵列部件可用作分频器，将输入至 TI00 引脚的时钟分频并从 TO00 输出结果。

从 TO00 输出的分频时钟频率可由下式计算。

- 当检测到上升沿/下降沿时：
分频后的时钟频率 = 输入时钟频率 / { (的设置值 + 1) × 2 }
- 检测到双边沿时：
分频后的时钟频率 ≅ 输入时钟频率 / (TDR00 的设置值 + 1)

TCR00 在间隔定时器模式下作为向下计数器工作。

通道开始触发位 (TS00) 设为 1 之后，当检测到 TI00 有效沿时，TCR00 载入 TDR00 的值。此时，如果 TMR00 时 MD000 = 0，则不输出 INTTM00 且不切换 TO00。如果 TMR00 的 MD000 = 1，则输出 INTTM00 并切换 TO00。

之后，TCR00 在 TI00 的有效沿向下计数。当 TCR00 = 0000H 时，它使 TO00 反转。同时，TCR00 再次载入 TDR00 的值并继续计数。

如果选择检测 TI00 的双边沿，则输入时钟的占空比误差会影响 TO00 输出的分配时钟周期。

TO00 输出时钟的周期包括工作时钟的一个周期的采样误差。

TO00 输出的时钟周期 = 理想 TO00 输出时钟周期 ± 工作时钟周期 (误差)

TDR00 可在任何时间重写。TDR0n 的新值在下一个周期内变为有效。

图 6-43. 用作分频器操作的框图

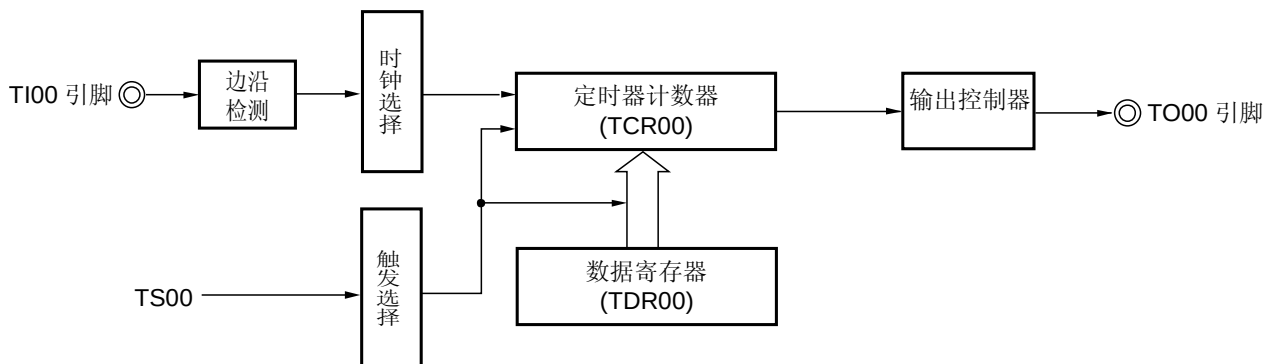


图 6-44. 作为分频器操作的基本时序的示例 (MD000 = 1)

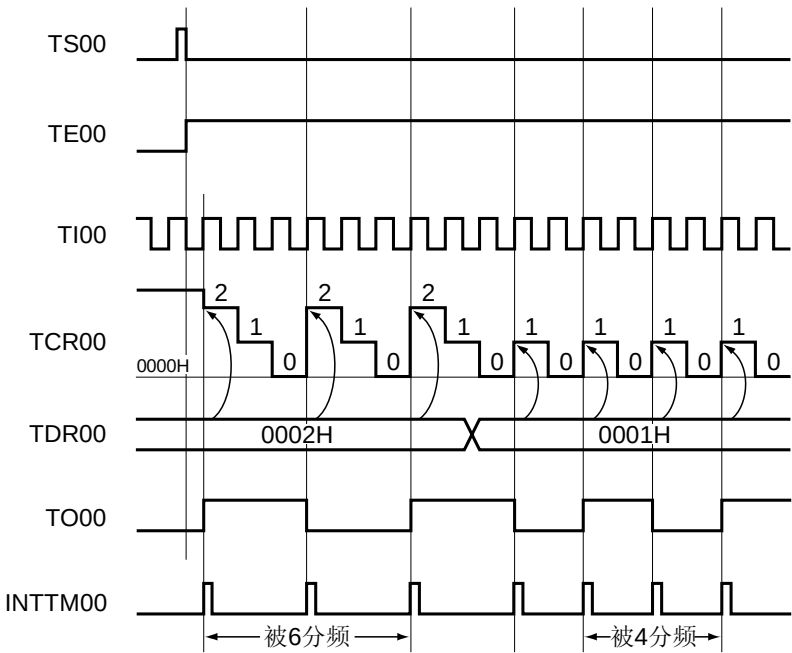
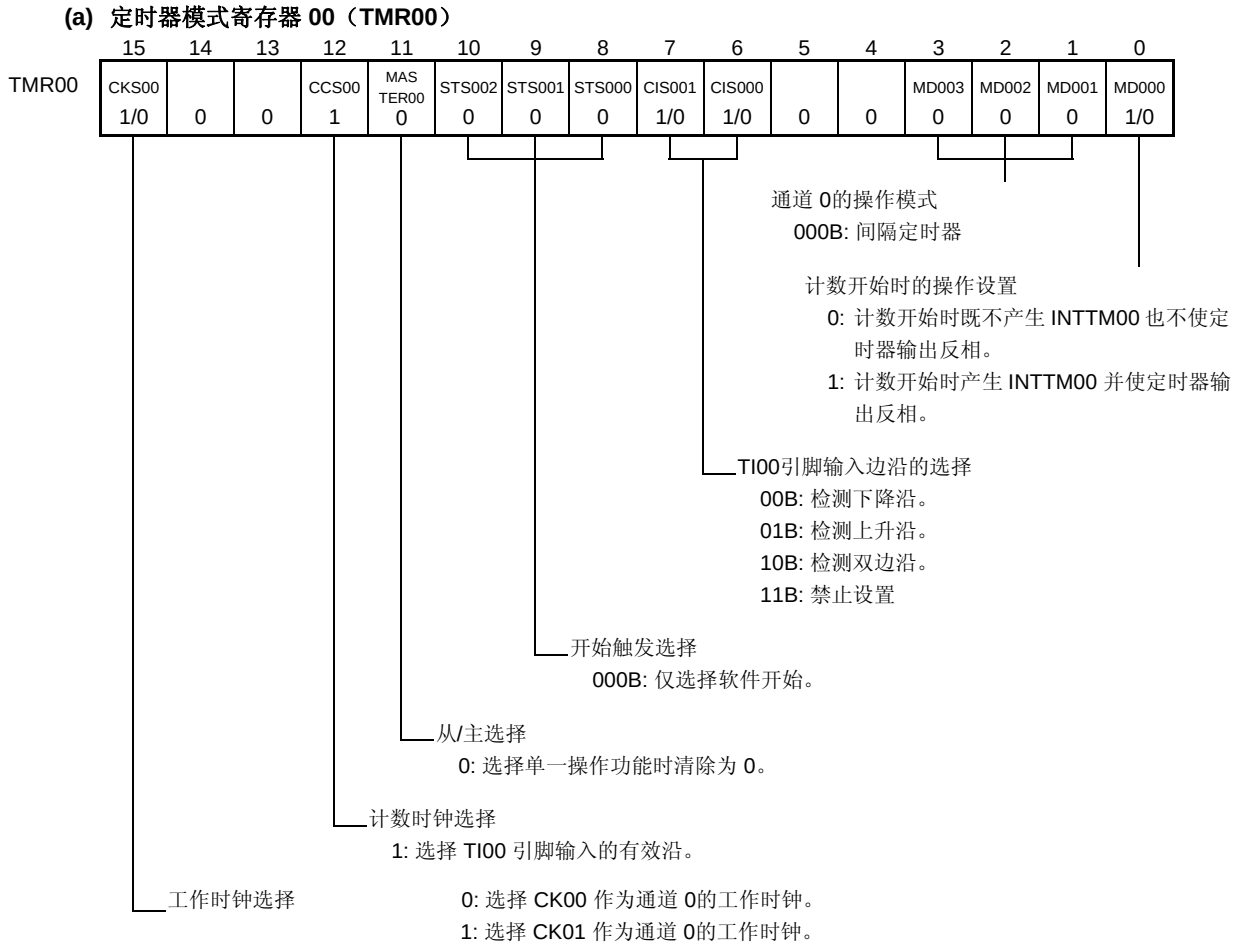


图 6-45. 作为分频器操作期间寄存器的设置内容的示例



(b) 定时器输出寄存器 0 (TO0)

位 0	
TO0	TO00
	1/0
	0: 从 TO00 输出 0。 1: 从 TO00 输出 1。

(c) 定时器输出使能寄存器 0 (TOE0)

位 0	
TOE0	TOE00
	1/0
	0: 通过计数操作停止 TO00 输出操作。 1: 通过计数操作使能 TO00 输出操作。

(d) 定时器输出电平寄存器 0 (TOL0)

位 0	
TOL0	TOL00
	0
	0: 当 TOM00 = 0(切换模式)时清除为 0。

(e) 定时器输出模式寄存器 0 (TOM0)

位 0	
TOM0	TOM00
	0
	0: 设置切换模式。

图 6-46. 使用分频器功能时的操作步骤

	软件操作	硬件状态
TAUS 默认设置		电源关闭状态 (停止提供时钟并禁止写入每个寄存器。)
	将PER2寄存器的TAU0EN位设为 1。	电源打开状态 各通道停止工作。 (开始提供时钟并允许写入每个寄存器。)
	设置TPS0寄存器。 确定CK00 和 CK01的时钟频率。	
通道默认设置	设置TMR00寄存器 (确定通道的操作模式)。 设置间隔 (周期) 值至TDR00 寄存器。	通道停止工作。 (提供时钟并消耗一定的功率。)
	将TOM0寄存器的TOM00位清为0 (切换模式)。 将TOL00 位清为 0。 设置 TO00位并确定TO00输出的默认电平。	TO00引脚进入高阻输出状态。 当端口模式寄存器处于输出模式且端口寄存器为0时，输出TO00默认设置电平。
	将 TOE00设为 1并使能TO00的操作。	TO00由于通道停止工作而不改变。
	将端口寄存器和端口模式寄存器清为 0。	TO00引脚输出TO00设置电平。
操作开始	将TOE00设为1 (仅当恢复操作时)。 将TS00位设为 1。 因为TS00位为触发位，所以它自动返回 0。	TE00 = 1，且计数操作开始。 在计数时钟输入时将TDR00的值载入TCR00。 如果TMR00寄存器的MD000位为1，则产生INTTM00且TO00执行切换操作。
操作期间	TDR00 寄存器的设置值可被改变。 总是可以读取 TCR00 寄存器。 未使用 TSR00 寄存器。 TO0 和 TOE0 寄存器的设置值可被改变。 TMR00 寄存器的设置值、TOM00 和 TOL00 位不可改变。	计数器 (TCR00) 向下计数。 当计数值达到0000H时，TDR00的值再次被载入TCR0，并继续计数操作。 通过检测TCR00 = 0000H，产生INTTM00且TO00执行切换操作。之后，重复以上的操作。
操作停止	TT00位被设为1。 因为TT00位为触发位，所以它自动返回为0。	TE00 = 0，且计数操作停止。 TCR00保持计数值并停止。 TO00输出不被初始化而保持当前状态。
	TOE00被清为 0且值被设置至TO00位。	TO00引脚输出TO00 设置电平。
TAUS 停止	要保持TO00引脚的输出电平 将要被保持的值设置至端口寄存器之后将TO00位清为 0。 当没必要保持 TO00 引脚的输出电平时	TO00引脚输出电平通过端口功能保持。
	将端口模式寄存器转换为输入模式。	TO00引脚输出电平进入高阻输出状态。
	将PER2寄存器的TAU0EN位清除为0。	电源关闭状态 初始化所有的电路和各通道的SFR。 (TO00位被清为 0且TO00引脚 被设为端口模式)。

操作恢复。

6.7.4 用作输入脉冲测量的操作

计数值可在 TI0n 有效沿处捕获，并可测量输入至 TI0n 的脉冲的间隔。
脉冲间隔可用下式计算。

$$\text{TI0n 输入脉冲间隔} = \text{计数时钟的周期} \times ((10000\text{H} \times \text{TSR0n: OVF}) + (\text{TDR0n 的捕获值} + 1))$$

注意事项 因为使用 TMR0n 寄存器的 CKS0n 位所选中的工作时钟对 TI0n 引脚输入进行采样，所以会产生大小等于一个工作时钟的误差。

TCR0n 在捕获模式下作为向上计数器工作。

当通过开始触发（TS0n）被设为 1 时，TCR0n 从 0000H 开始与计数时钟同步向上计数。

当检测到 TI0n 引脚输入有效沿时，将计数值发送（捕获）至 TDR0n，且同时将计数器（TCR0n）清除为 0000H 并输出 INTTM0n。如果此时计数器溢出，则将 TSR0n 寄存器的 OVF 位设为 1。如果计数器没有溢出，则清除 OVF 位。之后，重复以上的操作。

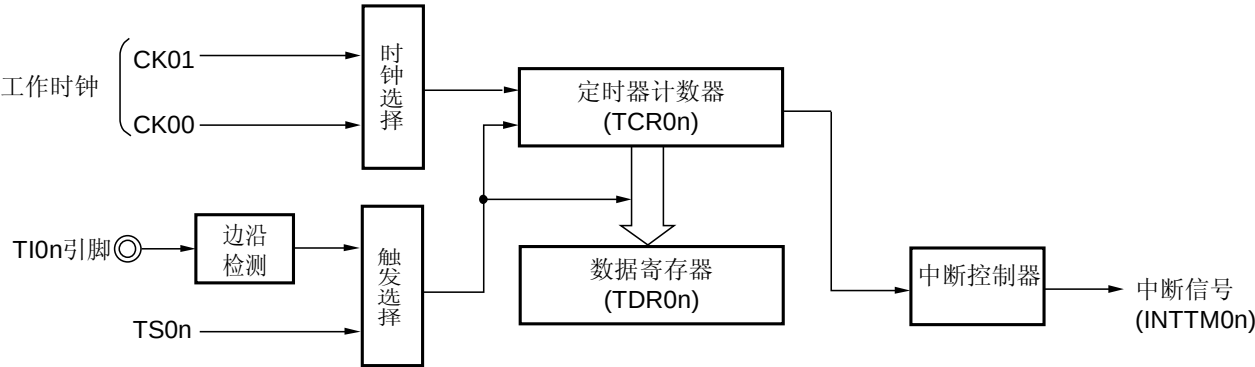
一旦将计数值捕获至 TDR0n 寄存器，则 TSR0n 寄存器的 OVF 立即根据计数器在测量期间溢出与否更新。所以，可检查捕获值的溢出状态。

如果计数器达到两个或两个以上周期的饱和计数，则认定它发生溢出，并将 TSR0n 寄存器的 OVF 位设为 1。但是，如果发生了两个或两个以上的溢出，则不能检测 OVF 位的标准间隔值。

将 TMR0n 寄存器的 STS0n2 至 STS0n0 设为 001B，以使用 TI0n 的有效沿作为开始触发和捕获触发。

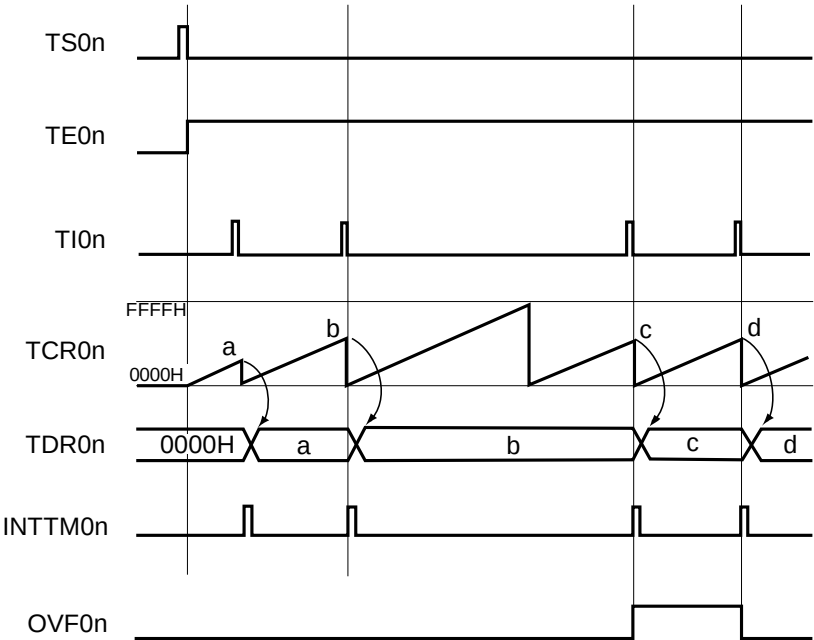
当 TE0n = 1 时，可使用软件操作（TS0n = 1）代替 TI0n 引脚输入作为捕获触发。

图 6-47. 作为输入脉冲间隔测量操作的框图



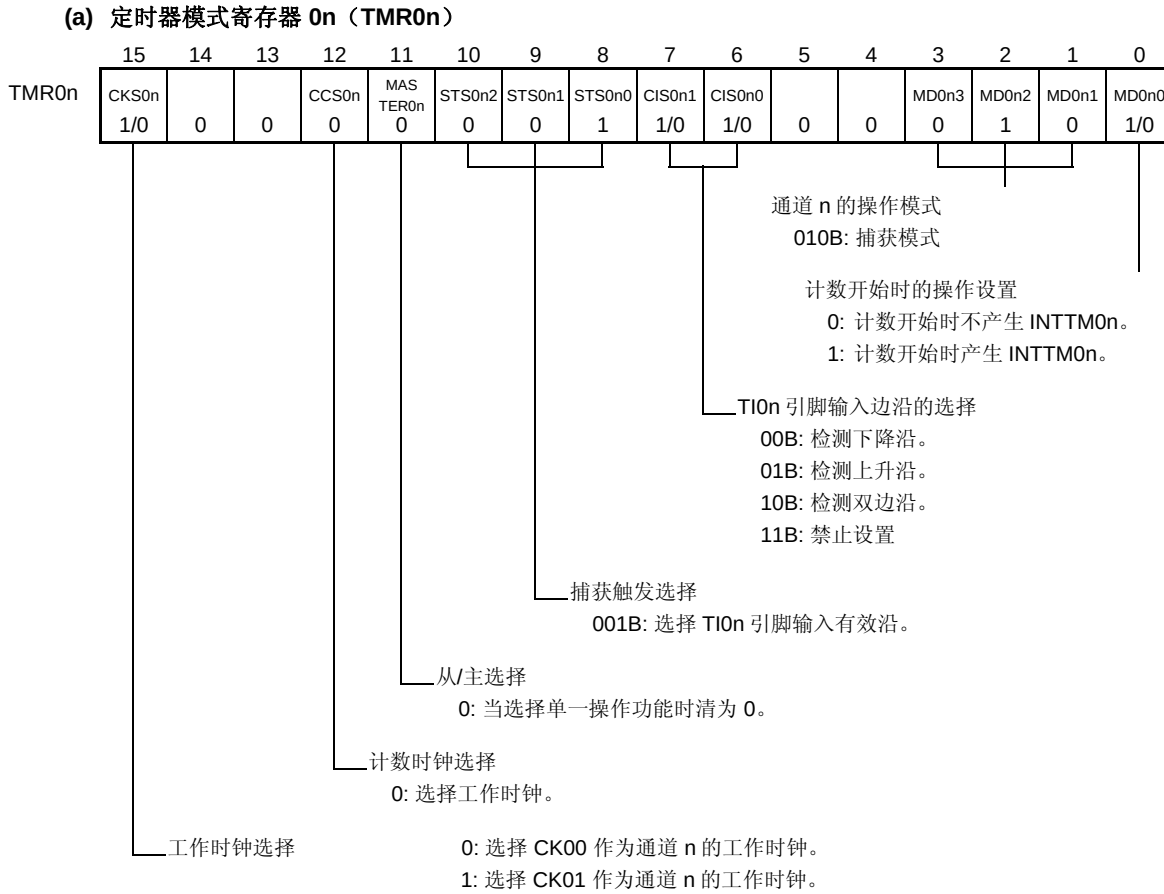
备注 n = 0 至 7

图 6-48. 作为输入脉冲间隔测量操作的基本时序的示例（MD0n0 = 0）

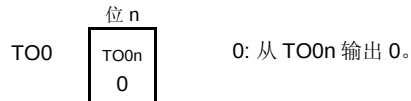


备注 n = 0 至 7

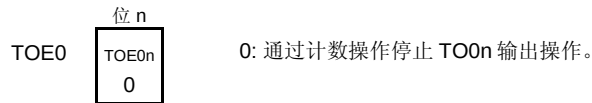
图 6-49. 测量输入脉冲间隔的寄存器设置内容的示例



(b) 定时器输出寄存器 0 (TO0)



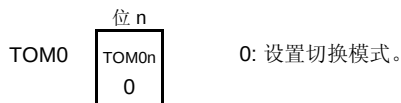
(c) 定时器输出使能寄存器 0 (TOE0)



(d) 定时器输出电平寄存器 0 (TOL0)



(e) 定时器输出模式寄存器 0 (TOM0)



备注 n = 0 至 7

图 6-50. 使用输入脉冲间隔测量功能时的操作步骤

	软件操作	硬件状态
TAUS 默认设置		电源关闭状态 (停止提供时钟并禁止写入每个寄存器。)
	将PER2寄存器的TAU0EN位设为1。	电源打开状态. 各通道停止工作。 (开始提供时钟并运行写入每个寄存器。)
	设置TPS0寄存器。 确定 CK00 和 CK01的时钟频率。	
通道默认设置	设置 TMR0n 寄存器 (确定通道的操作模式)。	通道停止工作。 (提供时钟并消耗一定的功率。)
操作开始	将TS0n 位设为1。 因为TS0n 位为触发位，所以它自动返回为 0。	TE0n = 1，且计数操作开始。 在计数时钟输入处将TCR0n清为 0000H。 当TMR0n寄存器的MD0n0 位为1时，产生 INTTM0n。
操作期间	仅 TMR0n 寄存器 CIS0n1 和 CIS0n0 位的设置值可被改变。 总是可以读取 TDR0n 寄存器。 总是可以读取 TCR0n 寄存器。 总是可以读取 TSR0n 寄存器。 TOM0n 寄存器的设置值、TOL0n、TO0n 和 TOE0n 位不可改变。	计数器(TCR0n)从 0000H 开始向上计数。 当检测到 TI0n 引脚输入的有效沿时，将计数值发送（捕获）至 TDR0n。 同时，将 TCR0n 清为 0000H 并产生 INTTM0n 信号。 如果此时发生溢出，则设置 TSR0n 寄存器的 OVF 位；如果没有发生溢出，则清除 OVF 位。 之后，重复以上的操作。
操作停止	TT0n位被设为1。 因为TT0n位为触发位，所以它自动返回为0。	TE0n = 0，且计数操作停止。 TCR0n保持计数值并停止。 TSR0n寄存器的OVF位也被保持。
TAUS 停止	将PER2寄存器的TAU0EN位清除为0。	电源关闭状态 初始化所有电路和各个通道的SFR。

<R> 操作恢复。

备注 n = 0 至 7

6.7.5 用作输入信号高/低电平宽度测量的操作

可通过在 $TI0n$ 的一个边沿开始计数，在另一个边沿捕获计数个数来测量 $TI0n$ 的信号宽度（高电平宽度/低电平宽度）。 $TI0n$ 的信号宽度可用下式计算。

$$TI0n \text{ 输入的信号宽度} = \text{计数时钟的周期} \times ((10000H \times \text{TSR0n: OVF}) + (\text{TDR0n 的捕获值} + 1))$$

注意事项 因为使用 $TMR0n$ 寄存器的 $CKS0n$ 位所选中的工作时钟对 $TI0n$ 引脚输入进行采样，所以会产生大小等于一个工作时钟的误差。

$TCR0n$ 在捕获&单次计数模式下作为向上计数器工作。

当通道开始触发 ($TS0n$) 被设为 1 时, $TE0n$ 被设为 1 并设置 $TI0n$ 引脚开始边沿检测等待状态。

当检测到 $TI0n$ 开始有效沿 (要测量高电平宽度时 $TI0n$ 的上升沿) 时, 计数器与计数时钟同步向上计数。当稍后检测到有效触发沿 (要测量高电平宽度时 $TI0n$ 的下降沿) 时, 将计数值发送至 $TDR0n$, 并同时输出 $INTTM0n$ 。如果此时计数器溢出, 则将 $TSR0n$ 寄存器的 OVF 位设为 1。如果计数器没有溢出, 则清除 OVF 位。 $TCR0n$ 在值“发送至 $TDR0n$ 的值 + 1”处停止, 并设置 $TI0n$ 引脚开始边沿检测等待状态。之后, 重复以上的操作。

一旦将计数值捕获至 $TDR0n$ 寄存器, 则 $TSR0n$ 寄存器的 OVF 立即根据计数器在测量期间溢出与否更新。所以, 可检查捕获值的溢出状态。

如果计数器达到两个或两个以上周期的饱和计数, 则认定它发生溢出, 并将 $TSR0n$ 寄存器的 OVF 位设为 1。但是, 如果发生了两个或两个以上的溢出, 则不能检测 OVF 位的标准间隔值。

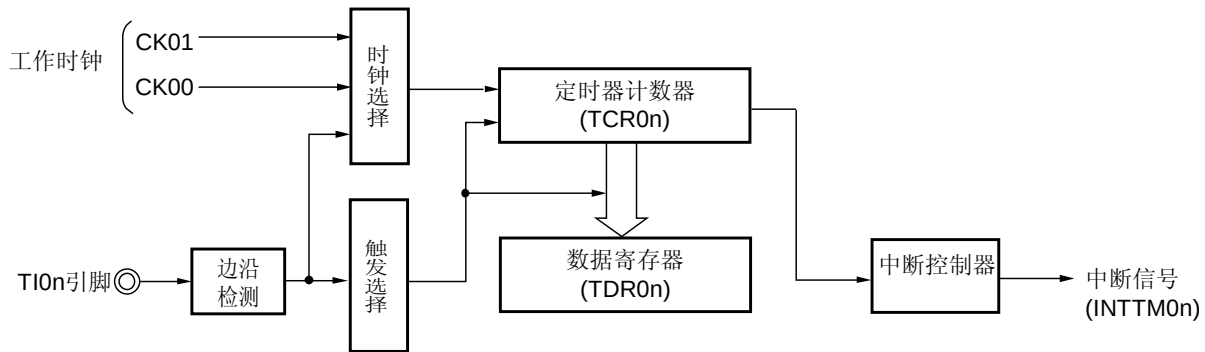
可使用 $TMR0n$ 寄存器的 $CIS0n1$ 和 $CIS0n0$ 位选择测量 $TI0n$ 引脚的高电平宽度还是测量 $TI0n$ 引脚的低电平宽度。

因为该功能用于测量 $TI0n$ 引脚输入的信号宽度, 所以在 $TE0n$ 为 1 期间不能将 $TS0n$ 设为 1。

$TMR0n$ 的 $CIS0n1$ 、 $CIS0n0 = 10B$: 测量低电平宽度。

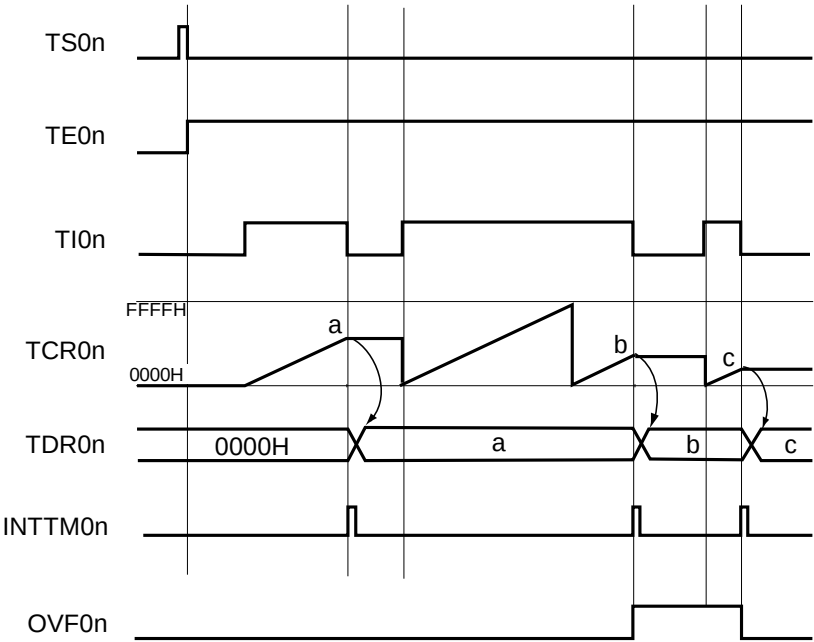
$TMR0n$ 的 $CIS0n1$ 、 $CIS0n0 = 11B$: 测量高电平宽度。

图 6-51. 用作输入信号高/低电平宽度测量操作的框图



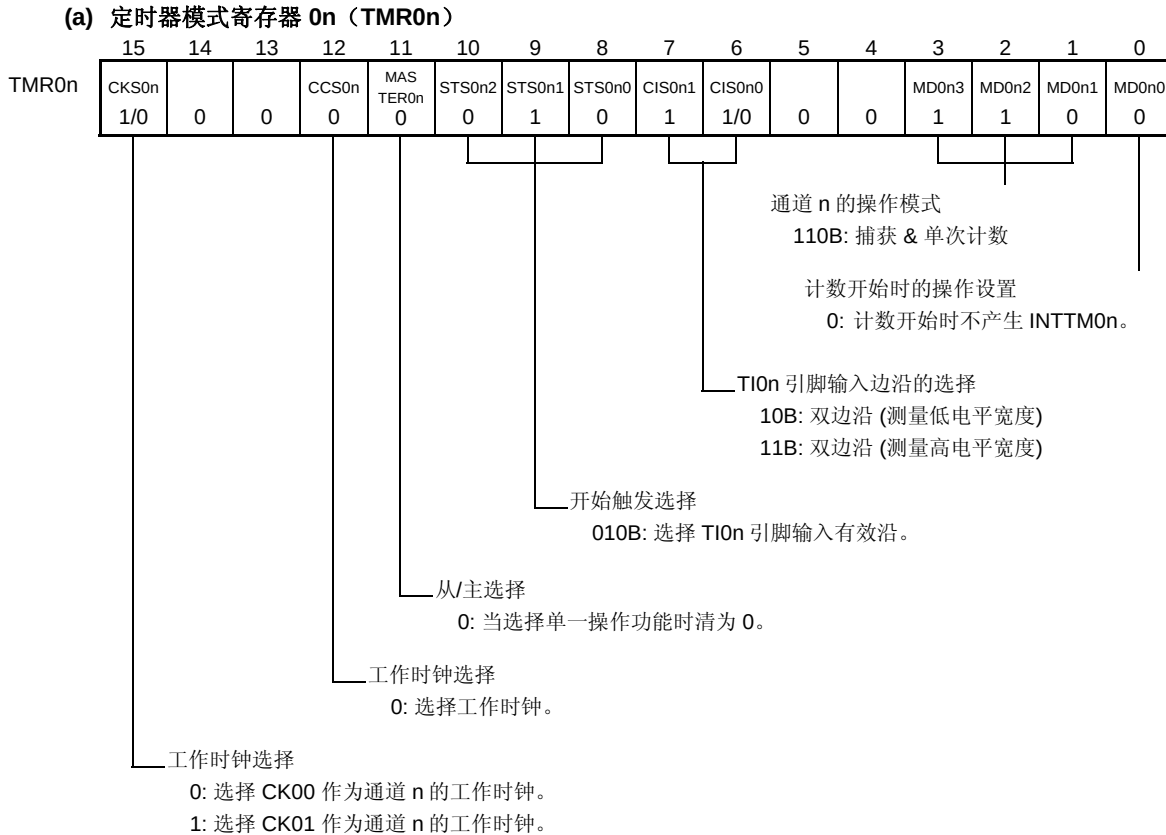
备注 n = 0 至 7

图 6-52. 用作输入信号高/低电平宽度测量操作基本时序的示例

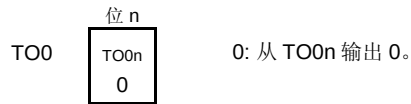


备注 n = 0 至 7

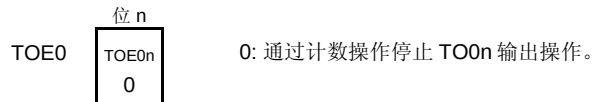
图 6-53. 测量输入信号高/低电平宽度的寄存器设置内容的示例



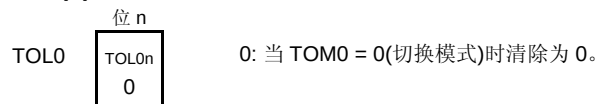
(b) 定时器输出寄存器 0 (TO0)



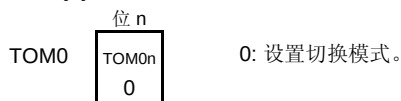
(c) 定时器输出使能寄存器 0 (TOE0)



(d) 定时器输出电平寄存器 0 (TOL0)



(e) 定时器输出模式寄存器 0 (TOM0)



备注 n = 0 至 7

图 6-54. 使用输入信号高/低电平宽度测量功能时的操作步骤

	软件操作	硬件状态
TAUS 默认设置		电源关闭状态 (停止提供时钟并禁止写入每个寄存器。)
	将PER2寄存器的TAU0EN位设为1。	电源打开状态. 各通道停止工作。 (开始提供时钟并允许写入每个寄存器。)
	设置TPS0寄存器。 确定 CK00 和 CK01的时钟频率。	
通道默认设置	设置TMR0n寄存器 (确定通道的操作模式)。 将TOE0n清为 0并停止TO0n的操作。	通道停止工作。 (提供时钟并消耗一定的功率。)
操作开始	将 TS0n 位设为1。 因为TS0n位为触发位，所以它自动返回为0。	TE0n = 1, 并设置TI0n引脚开始边沿检测等待状态 。
	检测 TI0n 引脚输入计数开始有效沿。	将TCR0n 清为0000H并开始向上计数。
操作期间	TDR0n寄存器的设置值可被改变。 总是可以读取TCR0n寄存器。 未使用 TSR0n寄存器。 TMR0n寄存器的设置值、TOL0n、TO0n和TOE0n位不可改变。	当检测到 TI0n 引脚开始边沿时，计时器(TCR0n)从 0000H 开始向上计数。 如果检测到 TI0n 引脚的捕获边沿，则将计数值发送至 TDR0n 并产生 INTTM0n。 如果此时发生溢出，则设置 TSR0n 寄存器的 OVF 位；如果没有发生溢出，则清除 OVF 位。 TCR0n 停止计数，直到检测到下一个 TI0n 引脚开始边沿为止。
操作停止	TT0n位被设为1。 因为TT0n位为触发位，所有它自动返回为0。	TE0n = 0, 且计数操作停止。 TCR0n保持计数值并停止。 TSR0n寄存器的OVF位也被保持。
TAUS 停止	将PER2寄存器的TAU0EN位清除为0。	电源关闭状态 初始化所有电路和各通道的SFR。

<R>
操作恢复。

备注 n = 0 至 7

6.8 定时器阵列部件 TAUS 复通道的操作

6.8.1 用作 PWM 功能的操作

两个通道可用作一个通道组生成任何周期和占空比的脉冲。
输出脉冲的周期和占空比可由下式计算。

脉冲周期 = {TDR0n 的设置值（主） + 1} × 计数时钟周期

占空比 [%] = {TDR0m 的设置值（从）}/{TDR0n 的设置值（主） + 1} × 100

0% 输出: TDR0m 的设置值（从） = 0000H

100%输出: TDR0m 的设置值（从） ≥ {TDR0n 的设置值（主） + 1}

备注 如果 TDR0m 的设置值（从） > （TDR0n 的设置值（主） + 1），占空比超过 100%，则将其概括为 100%输出。

主通道在间隔定时器模式下工作并计数周期。当通道开始触发（TS0n）被设为 1 时，输出 INTTM0n。TCR0n 从 TDR0n 的载入值开始与计数时钟同步开始向下计数。当 TCR0n = 0000H 时，输出 INTTM0n。TCR0n 再次载入 TDR0n 的值。之后，它重复相似的操作。

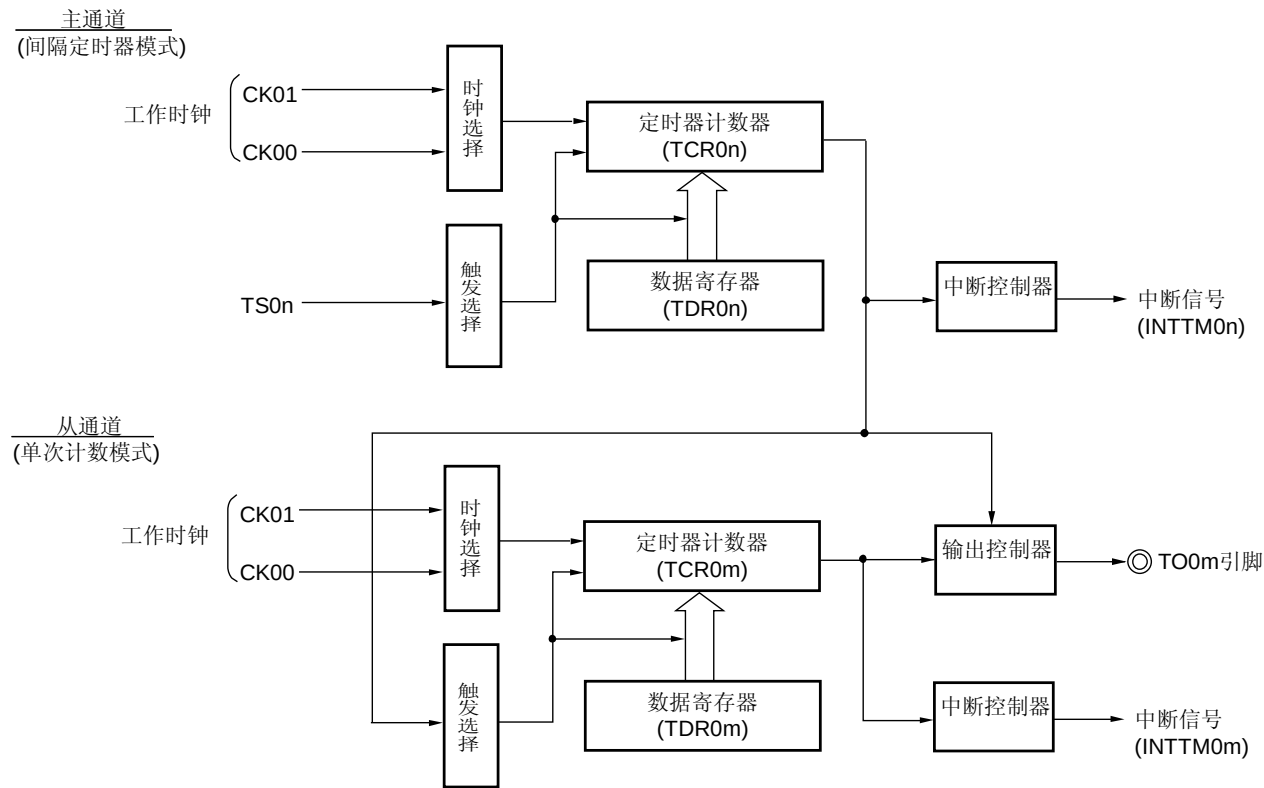
从通道的 TCR0m 在单次计数模式下工作，计数占空比并从 TO0m 引脚输出 PWM 波形。从通道的 TCR0m 通过使用主通道的 INTTM0n 作为开始触发载入 TDR0m 的值，并停止计数直到输入下一个开始触发（主通道的 INTTM0n）为止。

TO0m 的输出电平在主通道产生 INTTM0n 之后的一个计数时钟后变为有效，当 TCR0m = 0000H 时变为无效。

注意事项 要重写主通道的 TDR0 和从通道的 TDR0m，需进行两次写入访问。TDR0n 和 TDR0m 的值被载入 TCR0n 和 TRC0m 的时序依赖于主通道的 INTTM0n 的产生。因此，当在主通道的 INTTM0n 产生之前和产生之后分开进行重写时，TO0m 引脚不能输出预期的波形。所以，要重写主通道的 TDR0n 和从通道的 TDR0m，务必在主通道产生 INTTM0n 之后的瞬间重写这两个寄存器。

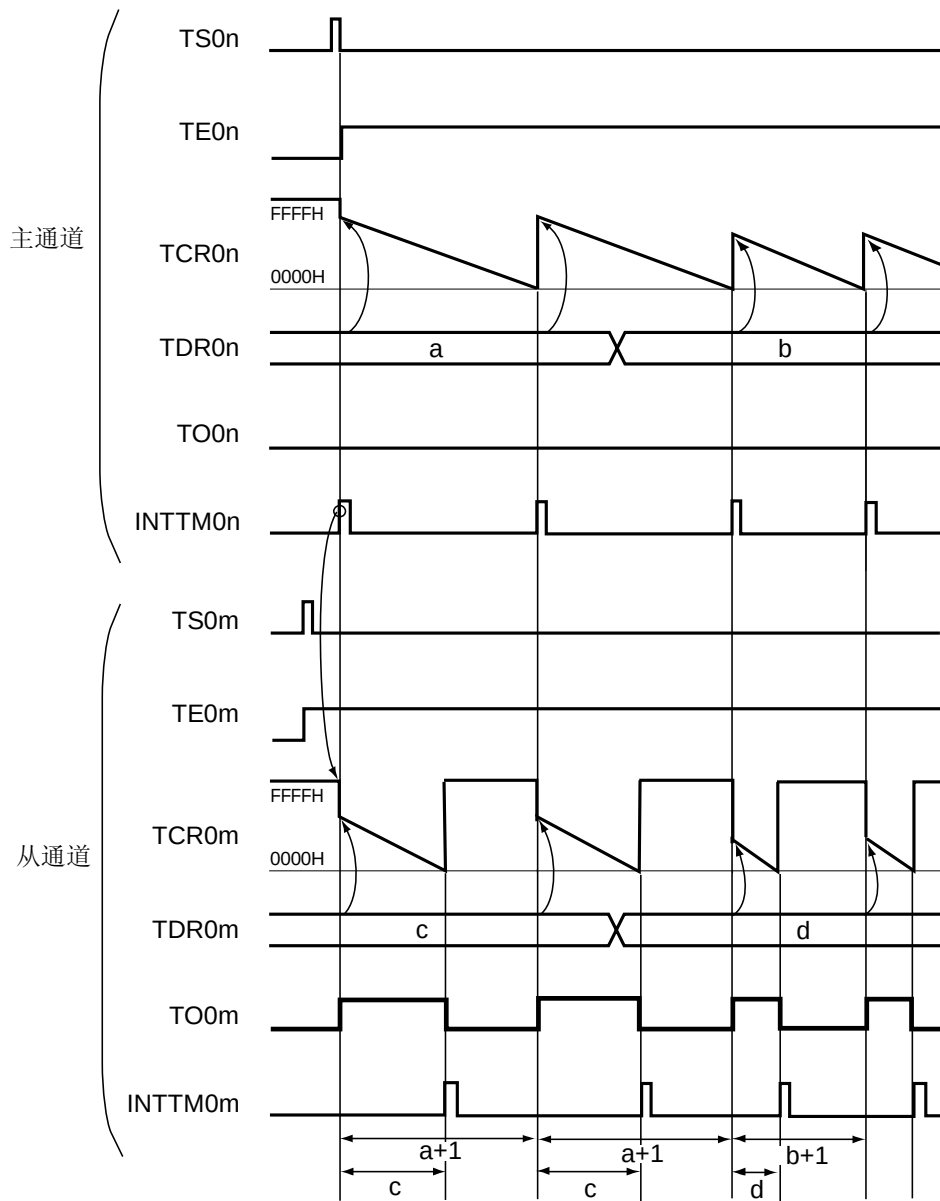
备注 n = 0、2、4、6
 m = n + 1

图 6-55. 用作 PWM 功能的框图



备注 n = 0、2、4、6
 m = n + 1

图 6-56. 用作 PWM 功能的基本时序的示例



备注 $n = 0, 2, 4, 6$
 $m = n + 1$

图 6-57. 使用 PWM 功能（主通道）时寄存器的设置内容的示例

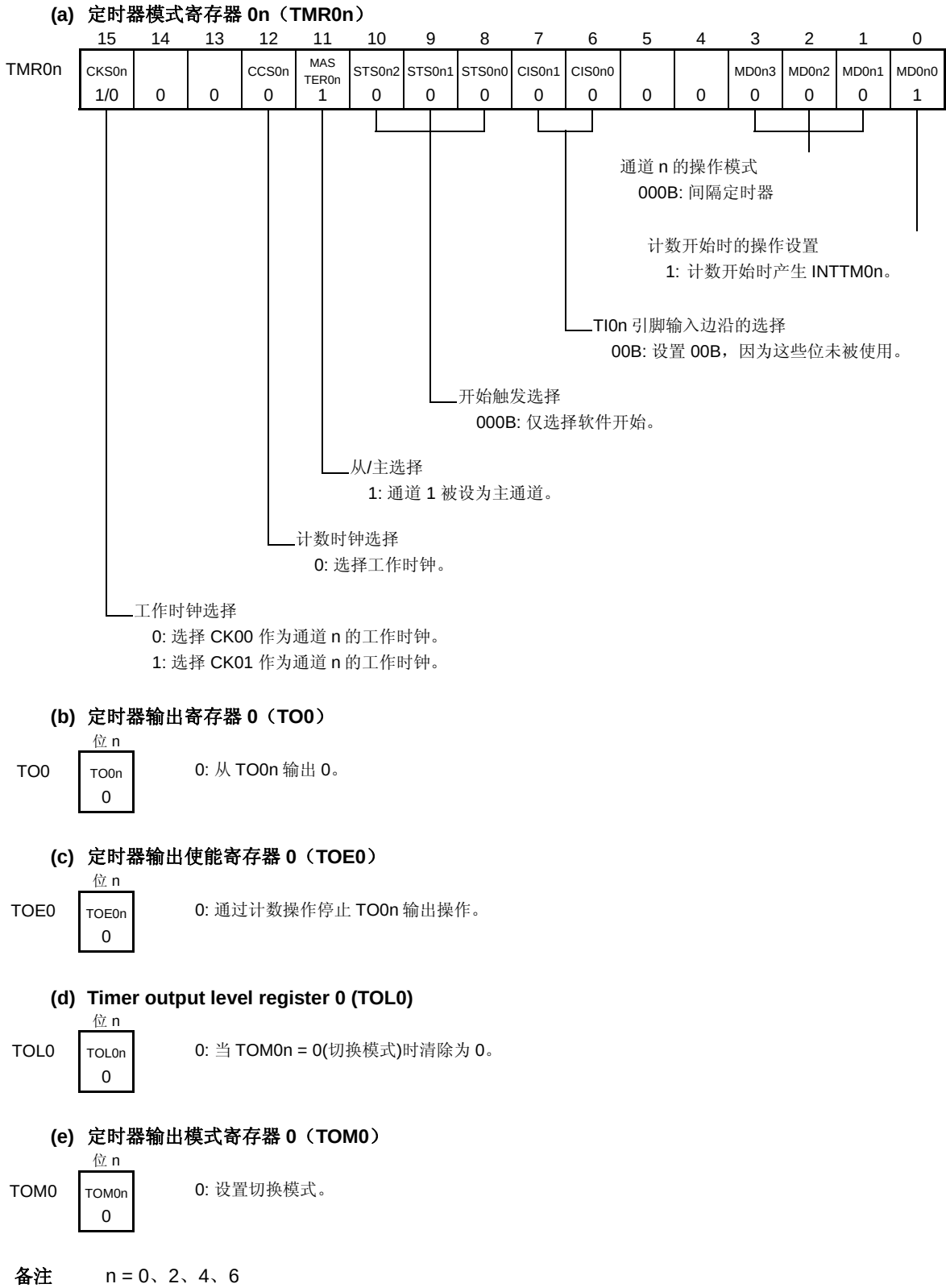
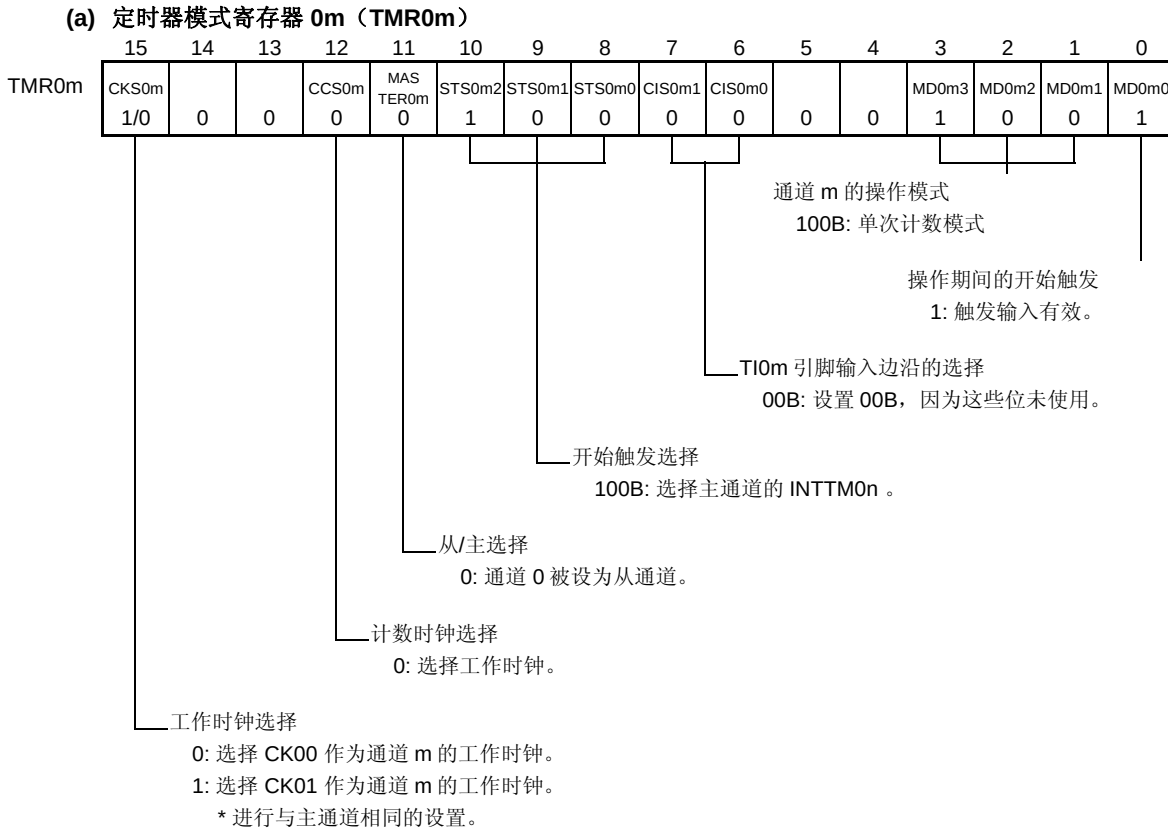


图 6-58. 使用 PWM 功能（从通道）时寄存器的设置内容的示例



(b) 定时器输出寄存器 0 (TO0)

位 m	
TO0m	0: 从 TO0m 输出 0。
1/0	1: 从 TO0m 输出 1。

(c) 定时器输出使能寄存器 0 (TOE0)

位 m	
TOE0m	0: 通过计数操作停止 TO0m 输出操作。
1/0	1: 通过计数操作使能 TO0m 输出操作。

(d) 定时器输出电平寄存器 0 (TOL0)

位 m	
TOL0m	0: 正逻辑输出 (高有效)
1/0	1: 反相输出 (低有效)

(e) 定时器输出模式寄存器 0 (TOM0)

位 m	
TOM0m	1: 设置组合操作功能模式。
1	

备注 n = 0、2、4、6
 m = n + 1

图 6-59. 使用 PWM 功能时的操作步骤 (1/2)

	软件操作	硬件状态
TAUS 默认设置		电源关闭状态 (停止提供时钟并禁止写入每个寄存器。)
	将PER2寄存器的TAU0EN位设为1。	电源打开状态. 各通道停止工作。 (开始提供时钟并允许写入每个寄存器。)
	设置TPS0寄存器。 确定CK00和CK01的时钟频率。	
通道默认设置	设置将被使用的两个通道的TMR0n和TMR0m寄存器（确定通道的操作模式）。 将间隔（周期）值设置至主通道的TDR0n寄存器；将占空比设置至从通道的TDR0m寄存器。	通道停止工作。 (提供时钟并消耗一定的功率。)
	设置从通道。 TOM0寄存器的TOM0m位被设为1（组合操作模式） 设置TOL0m 位。 设置TO0m位并确定 TO0m输出的默认电平。	TO00m引脚进入高阻输出状态。
	将TOE0m设为1并使能TO0m的操作。 将端口寄存器和端口模式寄存器清为0。	当端口模式寄存器处于输出模式且端口寄存器为0时，输出TO0m默认电平。 TO0m不改变，因为通道停止工作。 TO0m引脚输出TO0m设置电平。

备注 n = 0、2、4、6
 m = n + 1

图 6-59. 使用 PWM 功能时的操作步骤 (2/2)

	软件操作	硬件状态
<R> 操作恢复。	操作开始 将TOE0m (从)设为1(仅当操作被恢复时)。 TS0寄存器的TS0n (主)和TS0m (从)位同时被设为1。 因为TS0n 和 TS0m 位为触发位, 所以它们可自动返回为0。	TE0n = 1, TE0m = 1 当主通道开始计数时, 产生INTTM0n。 通过该中断进行触发, 从通道也开始计数。
	操作期间 TMR0n和TMR0m寄存器的设置值、TOM0n、TOM0m、TOL0n和TOL0m位不可改变。 TDR0n和TDR0m寄存器的设置值可在产生主通道的INTTM0n之后进行更改。 总是可以读取TCR0n和TCR0m寄存器。 TSR0n和TSR0m寄存器未被使用。 TO0 和 TOE0寄存器的设置值可被改变。	主通道的计数器将TDR0n值载入至TCR0n, 并向下计数。当计数值达到TCR0n = 0000H, 产生INTTM0n输出。 同时, TDR0n寄存器的值被载入TCR0n, 且计数器重新开始向下计数。 在从通道, TDR0m的值被载入TCR0m, 通过主通道的INTTM0n触发, 且计数器开始向下计数。 从主通道产生INTTM0n输出之后的一个时钟之后, TO0m的输出电平变为有效。 当TCR0m = 0000H时它变为有效, 并停止计数操作。之后, 重复以上的操作。
	操作停止 TT0n (主)和TT0m (从)位同时被设为1。 因为TT0n和TT0m位为触发位, 所有它们可自动返回为0。	TE0n, TE0m = 0, 且计数操作停止。 TCR0n和TCR0m保持计数值并停止。 TO0m输出不被初始化而保持当前状态。
	从通道的TOE0m被清为0且值被设置至TO0m位。	TO0m引脚输出TO0m设置电平。
TAUS 停止	要保持TO0m引脚输出电平 将要被保持的值设置至端口寄存器后, 清除 TO0m位。 当不必要保持TO0m引脚输出电平时	TO0m引脚输出电平通过端口功能保持。
	将端口模式寄存器转换为输入模式。 将PER2寄存器的TAU0EN位清除为0。	TO0m引脚输出电平进入高阻输出状态。 电源关闭状态 初始化所有电路和各个通道的SFR。 (TO0m位被清为0且TO0m引脚被设为端口模式。)

备注 n = 0、2、4、6
 m = n + 1

6.8.2 作为单次脉冲输出功能的操作

通过使用两个通道作为一组，可从输入至 TI0n 引脚的信号产生具有任何延迟脉冲宽度的单次脉冲。
延迟时间和脉冲宽度可由下式计算。

$$\begin{aligned} \text{延迟时间} &= \{\text{TDR0n 的设置值 (主)} + 2\} \times \text{计数时钟周期} \\ \text{脉冲宽度} &= \{\text{TDR0m 的设置值 (从)}\} \times \text{计数时钟周期} \end{aligned}$$

主通道在单次计数模式下工作并计数延迟。主通道的 TCR0n 基于开始触发检测开始操作，且 TCR0n 载入 TDR0n 的值。TCR0n 从其已载入的 TDR0n 值开始与计数时钟同步开始向下计数。当 TCR0n = 0000H 时，它输出 INTTM0n 并停止计数，直到检测到下一个开始触发为止。

从通道在单次计数模式下工作并计数脉冲宽度。从通道的 TCR0m 使用主通道的 INTTM0 作为开始触发开始工作，并载入 TDR0m 值。TCR0m 从其载入的 TDR0m 的值开始与计数值同步开始向下计数。当 TCR0m = 0000H 时，它输出 INTTM0m 并停止计数，直到检测到下一个开始触发（主通道的 INTTM0n）为止。TO0m 的输出电平在主通道产生 INTTM0n 之后的一个计数时钟后变为有效，当 TCR0m = 0000H 时变为无效。

除了使用 TI0n 引脚输入，还可以使用软件操作（TS0n = 1）作为开始触发输出单次脉冲。

注意事项 主通道的 TDR0n 的载入时序不同于从通道的 TDR0m 的载入时序。因此，如果在操作期间重写 TDR0n 和 TDR0m，会输出非法波形。务必在产生了将被重写的通道的 INTTM0n 之后再重写 TDR0n 和 TDR0m。

备注 n = 0、2、4、6
 m = n + 1

图 6-60. 用作单次脉冲输出功能操作的框图

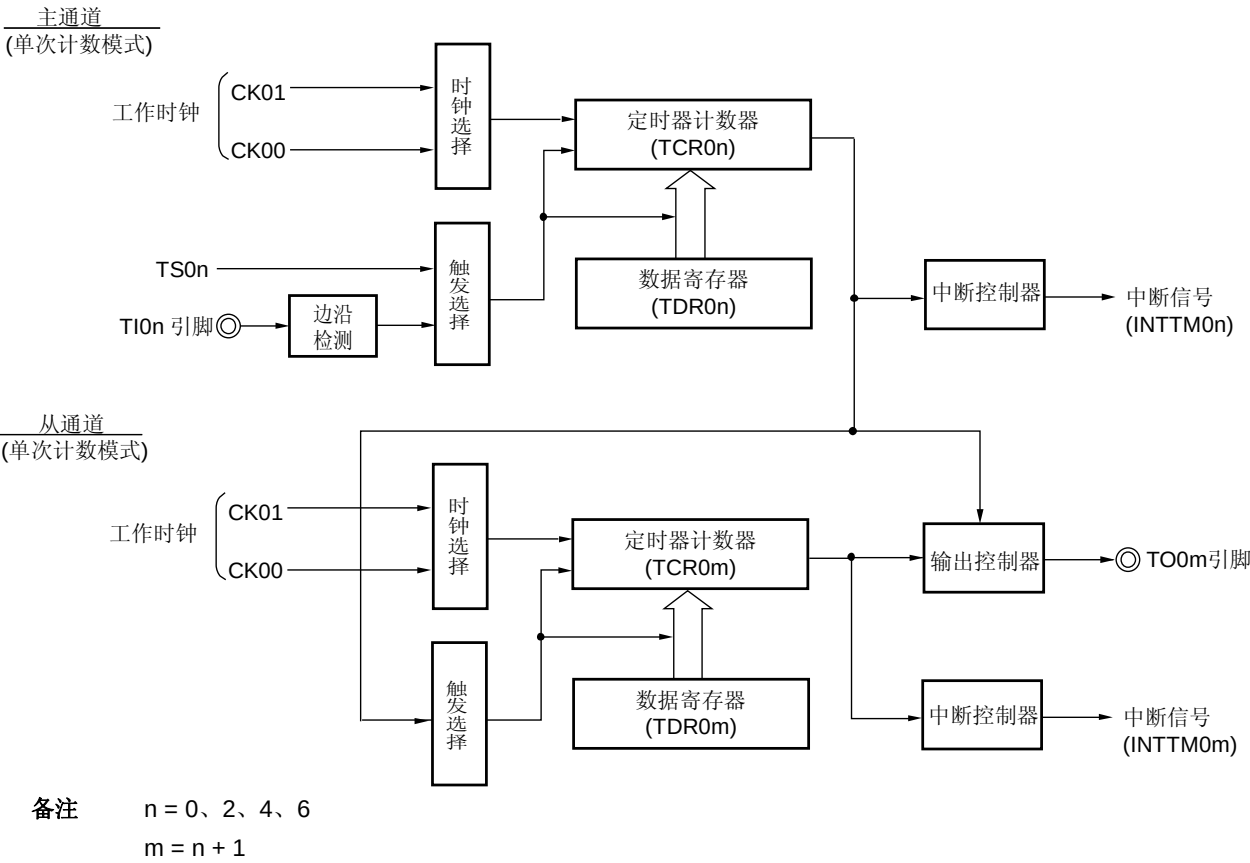
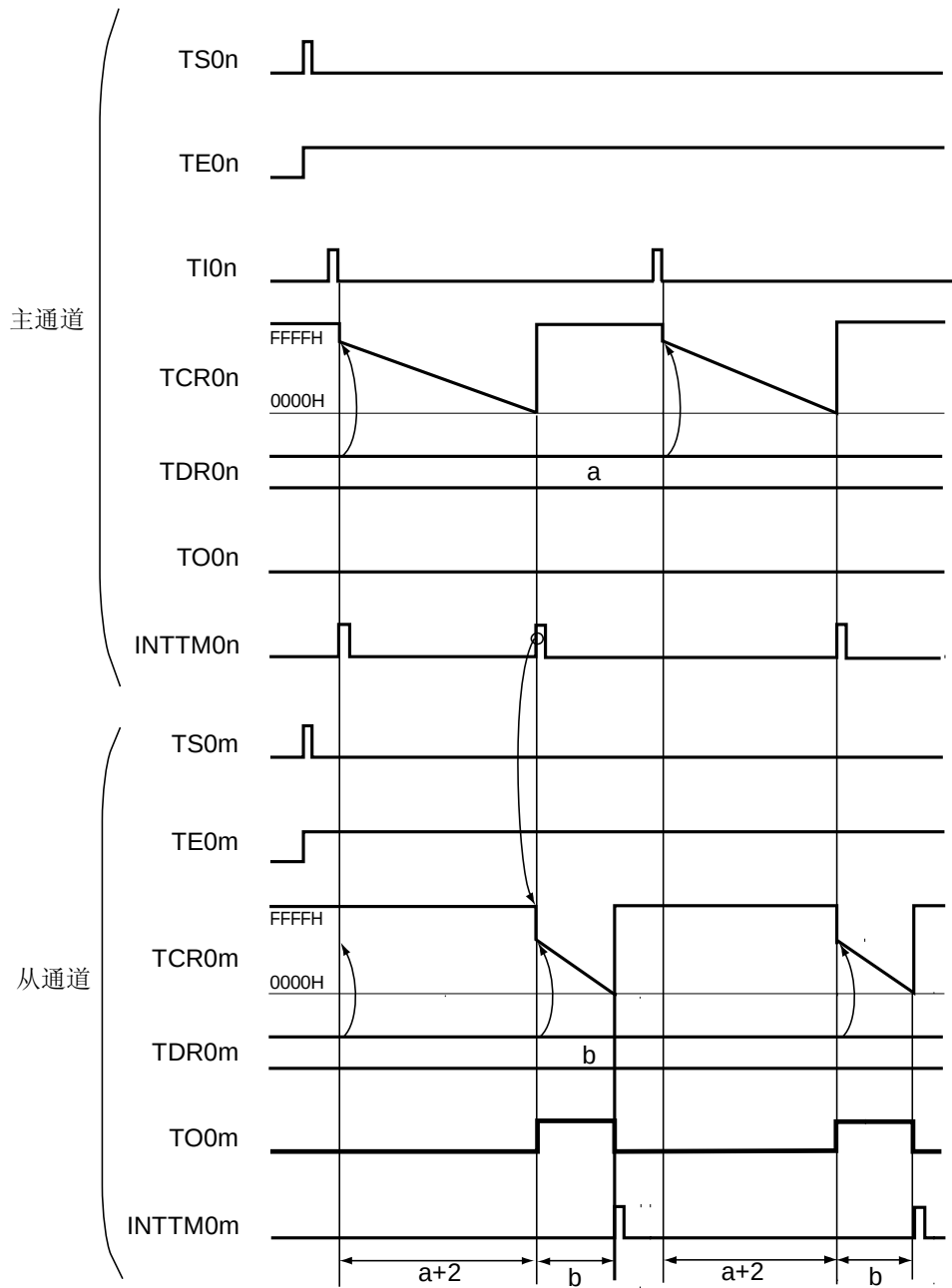
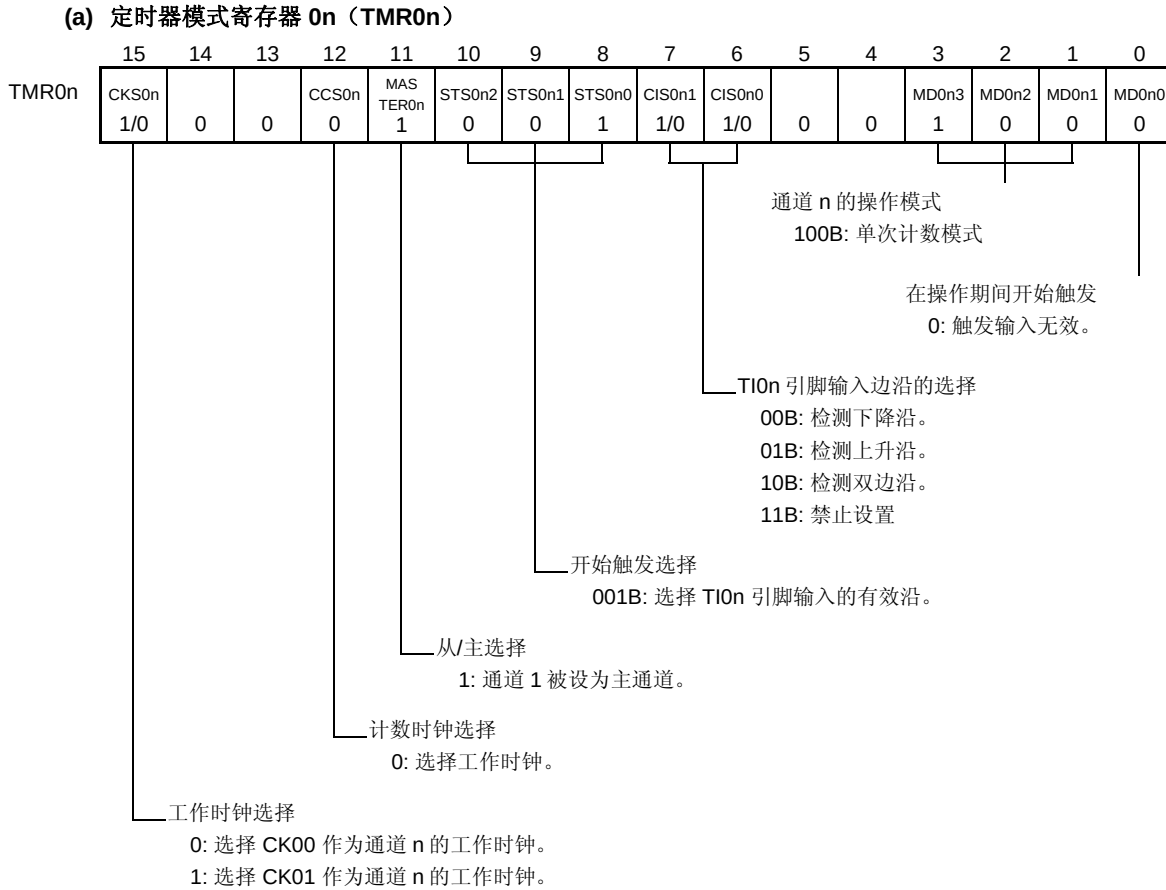


图 6-61. 作为单次脉冲输出功能操作的基本时序的示例

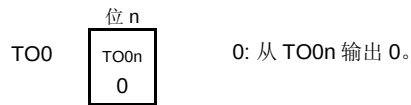


备注 $n = 0、2、4、6$
 $m = n + 1$

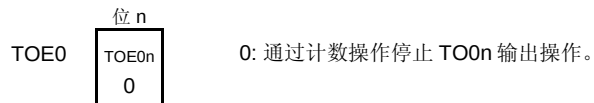
图 6-62. 寄存器设置内容的示例
当使用单次脉冲输出功能时（主通道）



(b) 定时器输出寄存器 0 (TO0)



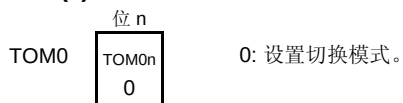
(c) 定时器输出使能寄存器 0 (TOE0)



(d) 定时器输出电平能寄存器 0 (TOL0)

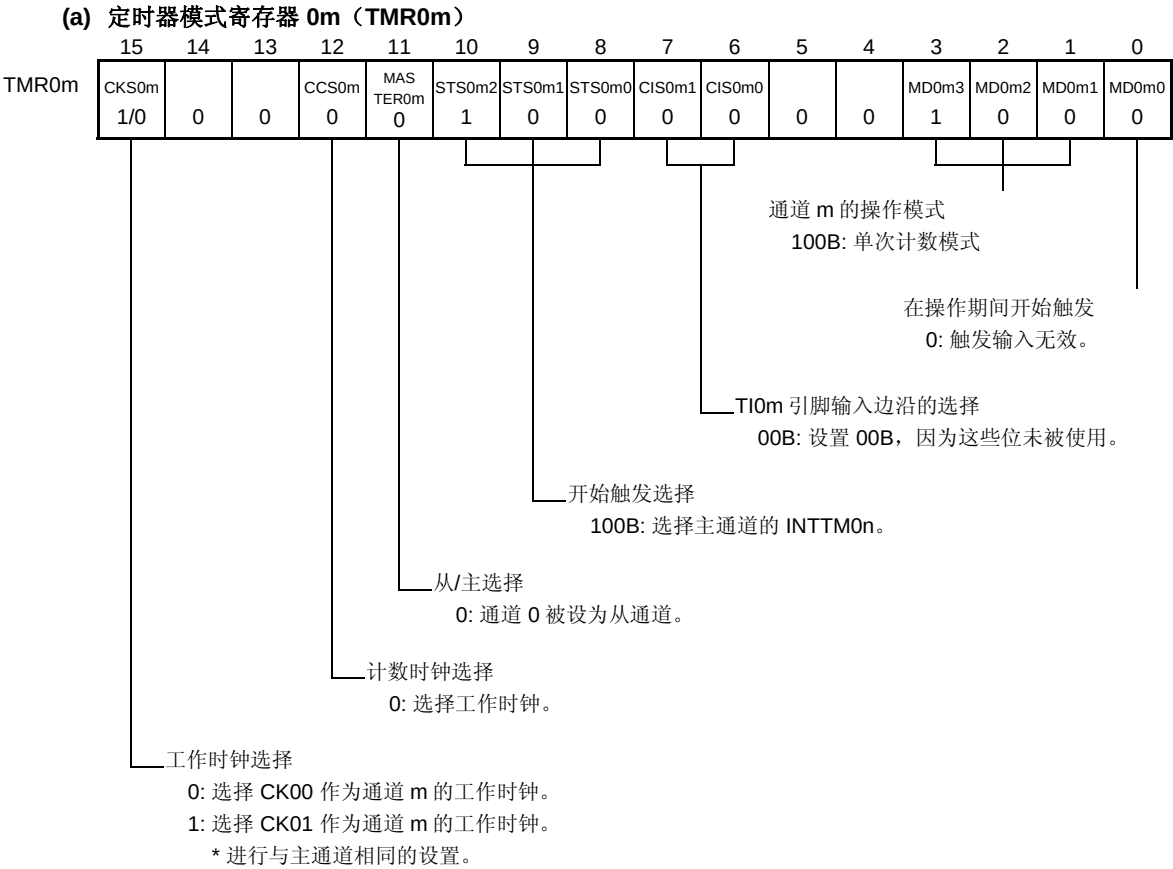


(e) 定时器输出模式寄存器 0 (TOM0)



备注 n = 0、2、4、6

图 6-63. 寄存器设置内容的示例
当使用单次脉冲输出功能时（从通道）



(b) 定时器输出寄存器 0 (TO0)

位 t m	
TO0	TO0m
	1/0
0: 从 TO0m 输出 0。 1: 从 TO0m 输出 1。	

(c) 定时器输出使能寄存器 0 (TOE0)

位 m	
TOE0	TOE0m
	1/0
0: 通过计数操作停止 TO0m 的输出操作。 1: 通过计数操作使能 TO0m 的输出操作。	

(d) 定时器输出电平寄存器 0 (TOL0)

位 m	
TOL0	TOL0m
	1/0
0: 正逻辑输出 (高有效) 1: 反相输出 (低有效)	

(e) 定时器输出模式寄存器 0 (TOM0)

位 m	
TOM0	TOM0m
	1
1: 设置组合操作模式。	

备注 n = 0、2、4、6
 m = n + 1

图 6-64. 单次脉冲输出功能的操作步骤 (1/2)

	软件操作	硬件状态
TAUS 默认设置		电源关闭状态 (停止提供时钟并禁止写入每个寄存器。)
	将PER2寄存器的TAU0EN位设为1。	电源打开状态. 各通道停止工作。 (开始提供时钟并允许写入每个寄存器。)
	设置TPS0寄存器。 确定CK00和CK01的时钟频率。	
通道默认设置	设置将被使用的两个通道的 TMR0n 和 TMR0m 寄存器 (确定通道的操作模式)。 将输出延迟设置至主通道的 TDR0n 寄存器; 将脉冲宽度 设置至从通道的 TDR0m 寄存器。	通道停止工作。 (提供时钟并消耗一定的功率。)
	设置从通道。 TOM0寄存器的TOM0m位被设为1 (组合操作模式) 设置TOL0m位。	TO0m0引脚进入高阻输出状态。
	设置TO0m位并确定TO0m输出的 默认电平。	当端口模式寄存器处于输出模式且端口寄存器为0时, 输出 TO0m默认电平。
	将TOE0m设为1并使能 TO0m的操作。 将端口寄存器和端口模式寄存器清为0。	TO0m不改变, 因为通道停止工作。 TO0m引脚输出TO0m的设置电平。

备注 $n = 0、2、4、6$
 $m = n + 1$

图 6-64. 单次脉冲输出功能的操作步骤 (2/2)



备注 n = 0、2、4、6
 m = n + 1

6.8.3 用作复用 PWM 输出功能的操作

通过扩展 PWM 功能并使用两个或两个以上的从通道，可以产生多个 PWM 输出信号。

例如，当使用两个从通道时，输出脉冲的周期和占空比可由下式计算。

$$\begin{aligned}\text{脉冲周期} &= \{\text{TDR0n 的设置值 (主)} + 1\} \times \text{计数时钟周期} \\ \text{占空比 1 [\%]} &= \{\text{TDR0p 的设置值 (从 1)}\} / \{\text{TDR0n 的设置值 (主)} + 1\} \times 100 \\ \text{占空比 2 [\%]} &= \{\text{TDR0q 的设置值 (从 2)}\} / \{\text{TDR0n 的设置值 (主)} + 1\} \times 100\end{aligned}$$

备注 如果 TDR0p 的设置值 (从 1) > {TDR0n 的设置值 (主) + 1} 或 {TDR0q 的设置值 (从 2)} > {TDR0n (主) 的设置值 + 1}，尽管占空比超过了 100%，也将其概括为 100% 输出。

主通道的 TCR0n 在间隔定时器模式下工作并计数周期。

从通道 1 的 TCR0p 在单次计数模式下工作，计数占空比并从 TO0p 引脚输出 PWM 波形。TCR0p 使用主通道的 INTTM0n 作为开始触发将 TDR0p 的值载入 TCR0p，并开始向下计数。当 TCR0p = 0000H 时，TCR0p 输出 INTTM0p 并停止计数，直到输入了下一个开始触发（主通道的 INTTM0n）为止。TO0p 的输出电平在主通道产生 INTTM0n 之后的一个计数时钟后变为有效，当 TCR0p = 0000H 时变为无效。

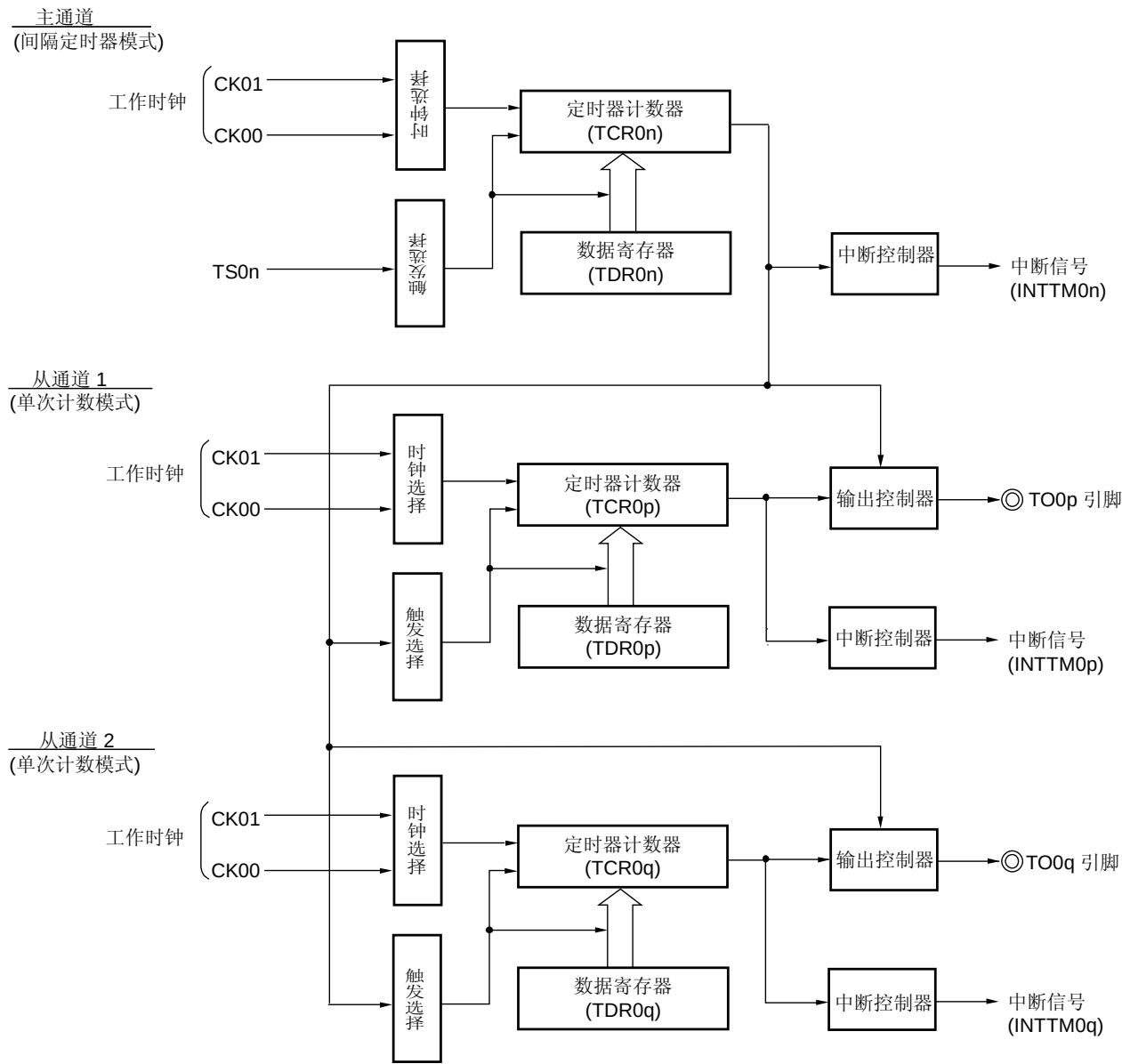
与从通道 1 的 TCR0p 相同，通道 2 的 TCR0q 在单次计数模式下工作，计数占空比并从 TO0q 引脚输出 PWM 波形。TCR0q 使用主通道的 INTTM0n 作为开始触发将 TDR0q 的值载入 TCR0q，并开始向下计数。当 TCR0q = 0000H 时，TCR0q 输出 INTTM0p 并停止计数，直到输入了下一个开始触发（主通道的 INTTM0n）为止。TO0q 的输出电平在主通道产生 INTTM0n 之后的一个计数时钟后变为有效，当 TCR0q = 0000H 时变为无效。

当使用通道 0 作为以上的主通道时，最多可同时输出七种类型的 PWM 信号。

备注 要重写主通道的 TDR0 和从通道 1 的 TDR0m，至少需进行两次写入访问。因为从主通道产生 INTTM0n 之后，TDR0n 和 TDR0p 的值被载入 TCR0n 和 TCR0p，所以如果在主通道产生 INTTM0n 前后单独进行重写，TO0p 引脚不能输出预期的波形。所以，要重写从通道的 TDR0n 和从通道的 TDR0p，务必在主通道产生 INTTM0n 之后的瞬间重写这两个寄存器（这同样适用于从通道 2）。

备注 $n = 0, 2, 4$
 $n < p < q \leq 7$
 p 和 q 为 n 之后的连续整数 ($p = n + 1$ 、 $q = n + 2$)

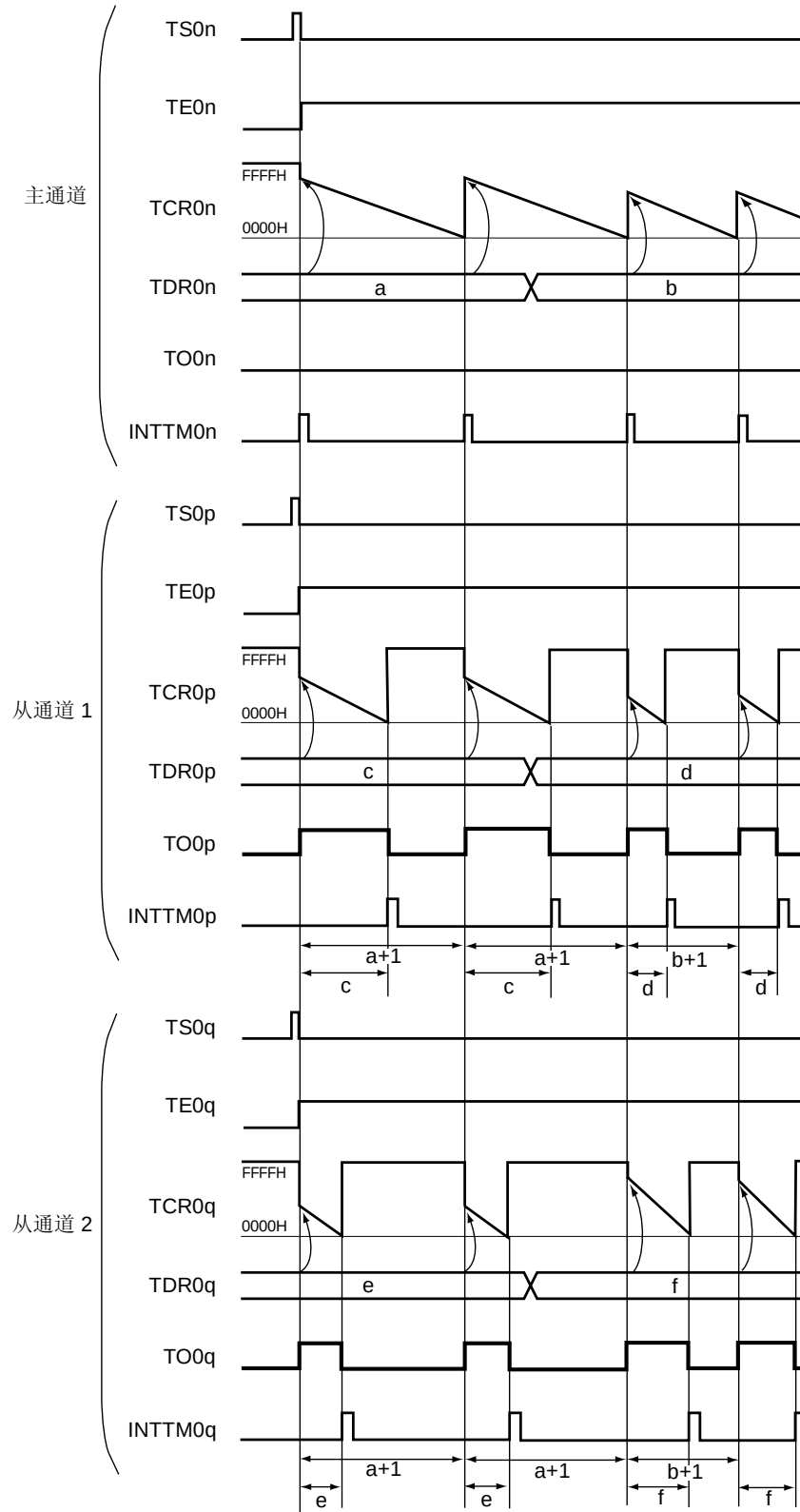
图 6-65. 作为 PWM 输出功能操作的框图（输出两种类型的 PWM）



备注

- 1. $n = 0、2、4$
- 2. $p = n + 1$
 $q = n + 2$

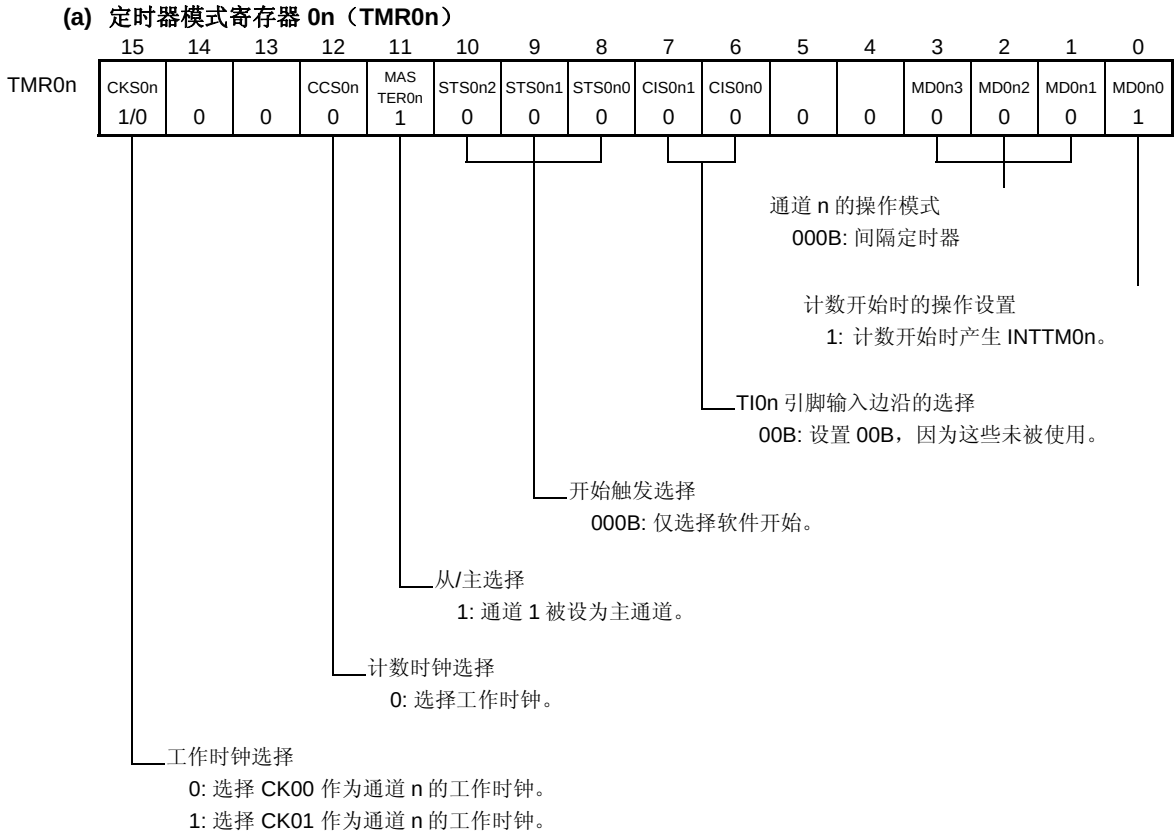
图 6-66. 作为 PWM 输出功能操作基本时序的示例（输出两种类型的 PWM）



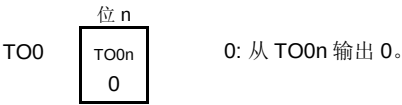
备注

1. $n = 0, 2, 4$
2. $p = n + 1$
 $q = n + 2$

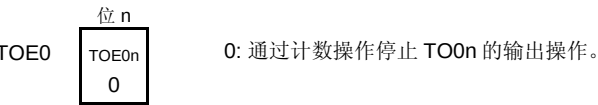
图 6-67. 寄存器设置内容的示例
当使用复用 PWM 输出功能（主通道）时



(b) 定时器输出寄存器 0 (TO0)



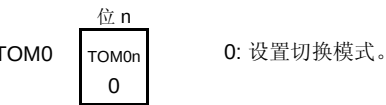
(c) 定时器输出使能寄存器 0 (TOE0)



(d) 定时器输出电平寄存器 0 (TOL0)



(e) 定时器输出模式寄存器 0 (TOM0)

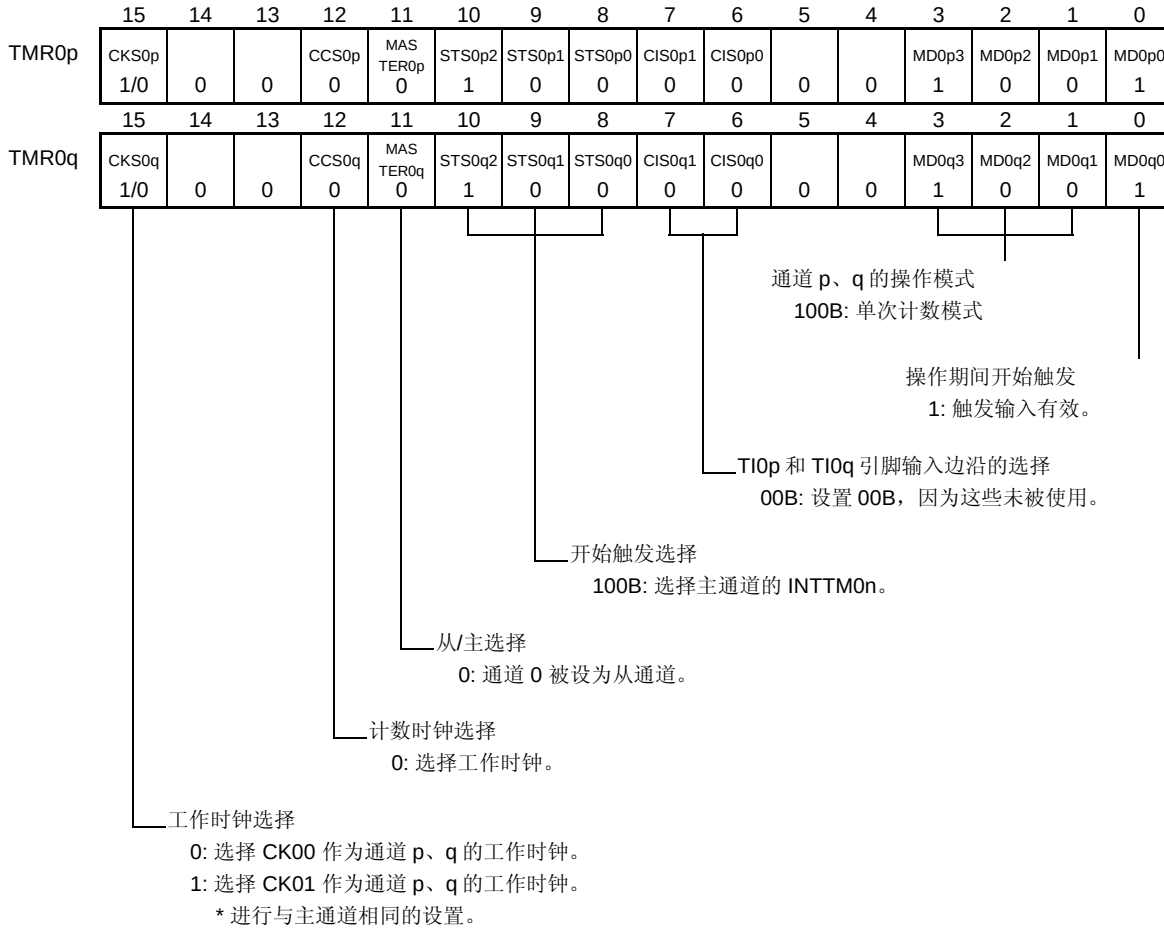


备注 n = 0、2、4

图 6-68. 寄存器设置内容的示例

当使用复用 PWM 输出功能（从通道）时（输出两种类型的 PWM）

(a) 定时器模式寄存器 0p、0q (TMR0p、TMR0q)



(b) 定时器输出寄存器 0 (TO0)

	位 q	位 p	
TO0	TO0q 1/0	TO0p 1/0	0: 从 TO0p 或 TO0q 输出 0。 1: 从 TO0p 或 TO0q 输出 1。

(c) 定时器输出使能寄存器 0 (TOE0)

	位 q	位 p	
TOE0	TOE0q 1/0	TOE0p 1/0	0: 通过计数操作停止 TO0p 或 TO0q 的输出操作。 1: 通过计数操作使能 TO0p 或 TO0q 的输出操作。

(d) 定时器输出电平寄存器 0 (TOL0)

	位 q	位 p	
TOL0	TOL0q 1/0	TOL0p 1/0	0: 正逻辑输出 (低有效) 1: 反相输出 (低有效)

(e) 定时器输出模式寄存器 0 (TOM0)

	位 q	位 p	
TOM0	TOM0q 1	TOM0p 1	1: 设置组合操作模式。

备注 n = 0、2、4; p = n+1; q = n+2

图 6-69. 使用复用 PWM 输出功能时的操作步骤 (1/2)

	软件操作	硬件状态
TAUS 默认设置		电源关闭状态 (停止提供时钟并禁止写入每个寄存器。)
	将PER2寄存器的TAU0EN位设为1。	电源打开状态. 各通道停止工作 (开始提供时钟并允许写入每个寄存器。)
	设置TPS0寄存器。 确定 CK00 和 CK01的时钟频率。	
通道默认设置	设置将被使用的各通道的 TMR0n、TMR0p和TMR0q寄存器（确定通道的操作模式）。 将间隔（周期）值设置至主通道的TDR0n寄存器；将占空比设置至从通道的TDR0p 和 TDR0q寄存器。	通道停止工作。 (提供时钟并消耗一定的功率。)
	设置从通道。 TOM0寄存器的TOM0p和TOM0q位被设为1（组合操作模式） 将 TOL0p 和 TOL0q位清为0。 设置TO0p 和 TO0q 位并确定TO0p和TO0q 的默认电平。	TO0p 和 TO0q 引脚进入高阻输出状态。
	将TOE0p 或 TOE0q 设为1 并使能TO0p 和 TO0q的操作。 将端口寄存器和端口模式寄存器清为0。	当端口模式寄存器处于输出模式且端口寄存器为 0 时，输出 TO0p 和 TO0q 默认电平。 TO0p 或 TO0q 不改变，因为通道停止工作。 TO0p 和 TO0q 引脚输出 TO0p 和 TO0q 的设置电平。

备注

1. n = 0、2、4

2. p = n + 1; q = n + 2

图 6-69. 使用复用 PWM 输出功能时的操作步骤 (2/2)

	软件操作	硬件状态
操作开始	将TOE0p 和 TOE0q (从)设为1(仅当操作被恢复时)。 TS0寄存器的TS0n位 (主)和TS0p 和 TS0q (从) 位同时被设为1。 因为TS0n、TS0p和TS0q位为触发位, 所以它们可自动返回为0。	TE0n = 1, TE0p、TE0q = 1 当主通道开始计数时, 产生INTTM0n。 通过该中断进行触发, 从通道也开始计数。
操作期间	TMR0n、TMR0p、TMR0q寄存器的设置值以及TOM0n, TOM0p、TOM0q、TOL0n、TOL0p和TOL0q位不可改变。 TDR0n、TDR0p和TDR0q寄存器的设置值可在产生主通道的INTTM0n之后进行更改。 总是可以读取TCR0n、TCR0p和TCR0q寄存器。 TSR0n、TSR0p和TSR0q寄存器未被使用。 TO0 和 TOE0寄存器的设置值可被改变。	主通道的计数器将TDR0n值载入至TCR0n, 并向下计数。当计数值达到TCR0n = 0000H, 产生INTTM0n输出。 同时, TDR0n寄存器的值被载入TCR0n, 且计数器重新开始向下计数。 在从通道1, TDR0p的值被发送至TCR0p, 通过主通道的INTTM0n触发, 且计数器开始向下计数。 从主通道产生INTTM0n输出之后的一个时钟之后, TO0p的输出电平变为有效。 当TCR0p = 0000H时它变为有效, 并停止计数操作。 在从通道2, TDR0q的值被发送至TCR0q, 通过主通道的INTTM0n触发, 计数器开始向下计数。 从主通道产生INTTM0n输出之后的一个时钟之后, TO0q的输出电平变为有效。 当TCR0q = 0000H时它变为有效, 并停止计数操作。之后, 重复以上的操作。
操作停止	TT0n (主)、TT0p和TT0q (从)位同时被设为1。 因为TT0n、TT0p和TT0q位为触发位, 所以它们可自动返回为0。 将从通道的TOE0p或TOE0q清为0 并将值设置至TO0p和TO0q 位。	TE0n、TE0p、TE0q = 0, 且计数操作停止。 TCR0n、TCR0p和TCR0q保持计数值并停止。 TO0p 和TO0q 输出不被初始化而是保持当前状态。 TO0p 和 TO0q 引脚输出TO0p 和 TO0q的设置电平。
TAUS 停止	要保持TO0p 和 TO0q 引脚输出电平 将要被保持的值设置至端口寄存器之后将 TO0p和TO0q 位清为0。 当没必要保持 TO0p 和 TO0q 引脚输出电平时 将端口寄存器转换为输入模式。 将PER2寄存器的TAU0EN位清除为0。	TO0p 和 TO0q引脚输出电平通过端口功能保持。 TO0p 和 TO0q引脚输出电平进入高阻输出状态。 电源关闭状态 初始化所有电路和各个通道的SFR。 (将TO0p 和 TO0q 位清为0 并将TO0p 和 TO0q 引脚设为端口模式。)

备注

1. n = 0、2、4
2. p = n + 1; q = n + 2

操作恢复。

第七章 实时计数器

7.1 实时计数器的功能

实时计数器具有以下特点：

- 具有年、月、星期、日、时、分和秒的计数器，最多能计数至 99 年。
- 稳定时期中断功能（时期：1 个月到 5 秒）。
- 报警中断功能（报警：星期、时、分）
- 间隔中断功能
- 引脚输出 1 Hz 的功能。
- 引脚输出 512 Hz 或 16.384 kHz 或 32.768 kHz 的功能。

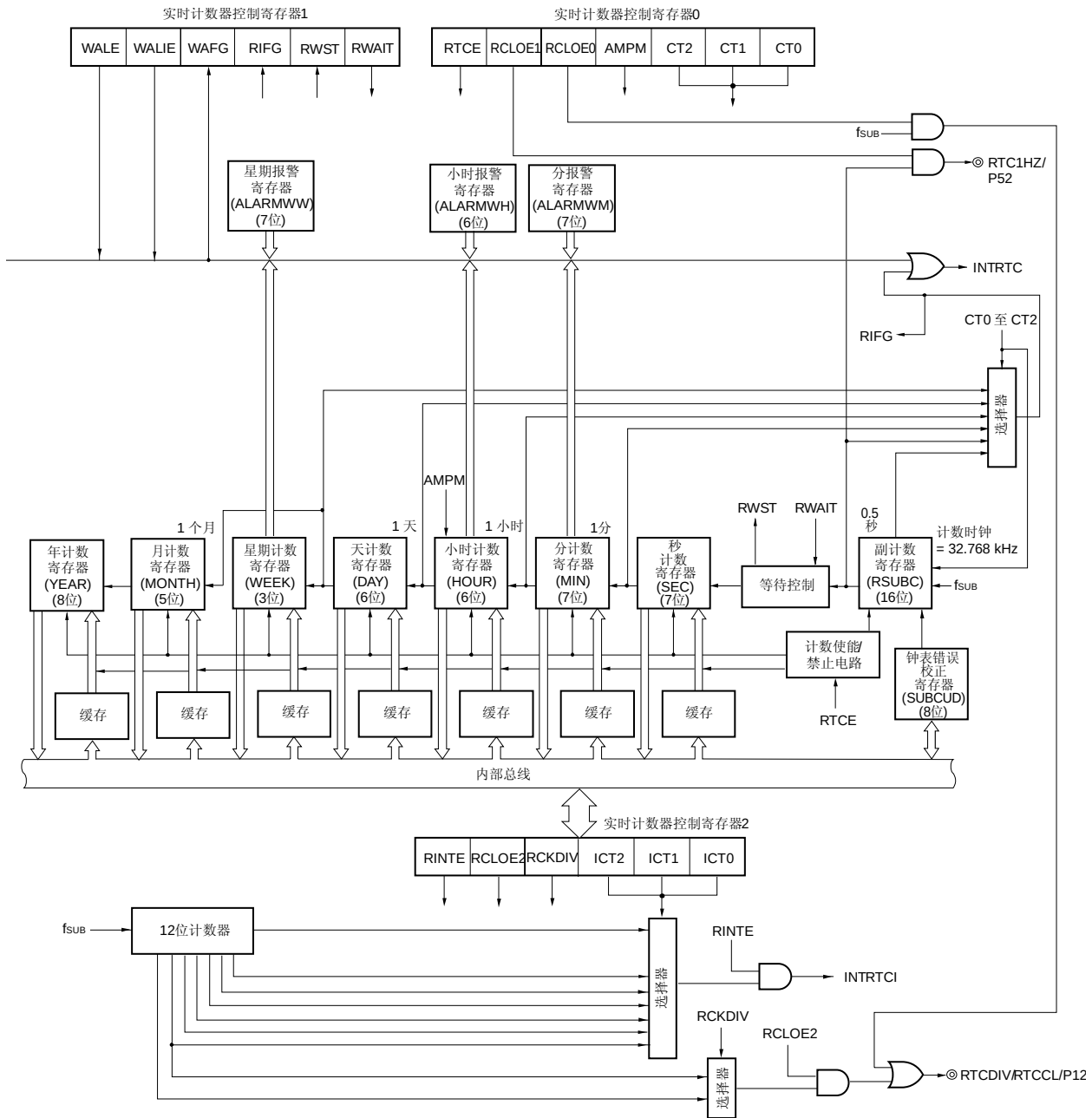
7.2 实时计数器的配置

实时计数器包含以下硬件：

表 7-1. 实时计数器的配置

项目	配置
控制寄存器	外围使能寄存器 0（PER0）
	实时计数器控制寄存器 0（RTCC0）
	实时计数器控制寄存器 1（RTCC1）
	实时计数器控制寄存器 2（RTCC2）
	副-计数寄存器（RSUBC）
	秒计数寄存器（SEC）
	分计数寄存器（MIN）
	小时计数寄存器（HOUR）
	天计数寄存器（DAY）
	星期计数寄存器（WEEK）
	月计数寄存器（MONTH）
	年计数寄存器（YEAR）
	钟表错误校正寄存器（SUBCUD）
	分报警寄存器（ALARMWM）
	小时报警寄存器（ALARMWH）
	星期报警寄存器（ALARMWW）

图 7-1. 实时计数器的框图



7.3 控制实时计数器的寄存器

实时计数器由以下寄存器控制：

- 外围使能寄存器 0 (PER0)
- 实时计数器控制寄存器 0 (RTCC0)
- 实时计数器控制寄存器 1 (RTCC1)
- 实时计数器控制寄存器 2 (RTCC2)
- 副-计数寄存器 (RSUBC)
- 秒计数寄存器 (SEC)
- 分计数寄存器 (MIN)
- 小时计数寄存器 (HOUR)
- 天计数寄存器 (DAY)
- 星期计数寄存器 (WEEK)
- 月计数寄存器 (MONTH)
- 年计数寄存器 (YEAR)
- 钟表错误校正寄存器 (SUBCUD)
- 分报警寄存器 (ALARMWM)
- 小时报警寄存器 (ALARMWH)
- 星期报警寄存器 (ALARMWW)

(1) 外围使能寄存器 0 (PER0)

PER0 寄存器用于启动或禁止各外围硬件宏功能的使用。若未使用硬件宏功能，则为其供应的时钟会停止工作，以便于减少功耗和噪声。

当使用实时计数器时，确保将该寄存器的位 7 (RTCEN) 置为 1。

可通过 1 位或 8 位的存储操作指令设置 PER0。

产生的复位信号将该寄存器清除为 00H。

图 7-2. 外围使能寄存器 (PER0) 的格式

地址: F00F0H 复位后: 00H R/W

符号	<7>	6	<5>	<4>	3	<2>	1	0
PER0	RTCEN	0	ADCEN	IICAEN	0	SAU0EN	0	0

RTCEN	实时计数器 (RTC) 输入时钟供应的控制 ^注
0	停止输入时钟的供应 - 实时计数器 (RTC) 使用的 SFR 不能被写入 - 实时计数器 (RTC) 处于复位状态。
1	供应输入时钟 实时计数器 (RTC) 使用的 SFR 可以被写入

注 当实时计数器 (RTC) 使用的寄存器被 CPU 访问时，使用由 RTCEN 控制的输入时钟。

RTCEN 不能控制输入至 RTC 的操作时钟 (f_{SUB}) 的供应

- 注意事项 1.使用实时计数器时，在副系统时钟 (f_{SUB}) 稳定时首先将 RTCEN 设置为 1。如果 RTCEN = 0，则忽略对实时计数器的控制寄存器的写入，若此时读取寄存器，则只能读到默认值。
2. 当使用副系统时钟时，通过设置工作速度模式控制寄存器 (OSMC) 的 RTCLPC 为 1，供应至除实时计数器外的其它外围的时钟可在 HALT 模式下被停止。
- 这种情况下，将 RTCEN 设置为 1 并将且 PER0 的位 0 至 6 设置为 0。此外，也将 PER1 和 PER2 寄存器的位 0 至位 7 设置为 0。
3. 务必要将 PER0 寄存器的位 0、1、3 以及 6 (44 引脚产品：位 0、1、3、4 和 6) 清为 0。

(2) 实时计数器控制寄存器 0 (RTCC0)

RTCC0 寄存器是 8 位寄存器，用于启动或停止实时计数器操作，并控制 RTCCL 和 RTC1HZ 引脚，以及设置 12-或 24-小时制系统和固定周期中断功能。

可通过 1 位或 8 位的存储操作指令设置 RTCC0。

产生的复位信号将该寄存器清除为 00H。

图 7-3. 实时计数器控制寄存器 0（RTCC0）的格式

地址：FFF9DH 复位后：00H R/W

符号

6

<5>

<4>

3

2

1

0

RTCC0

RTCE	0	RCLOE1	RCLOE0	AMPM	CT2	CT1	CT0

RTCE	实时计数器操作控制						
0	停止计数器操作。						
1	启动计数器操作。						

RCLOE1	RTC1HZ 引脚输出控制						
0	禁止 RTC1HZ 引脚输出（1 Hz）。						
1	允许 RTC1HZ 引脚输出（1 Hz）。						

RCLOE0 ^注	RTCCL 引脚输出控制						
0	禁止 RTCCL 引脚输出（32.768 kHz）。						
1	允许 RTCCL 引脚输出（32.768 kHz）。						

AMPM	选择12-/24-小时系统						
0	12-小时系统（显示上午和下午）						
1	24-小时系统						
• 若要更改 AMPM 的值，将 RWAIT（RTCC1 的位 0）设置为 1，并复位小时计数寄存器（HOUR）。 • 表 7-2 展示了所显示时间的位数。							

CT2	CT1	CT0	稳定时期中断（INTRTC）的选择
0	0	0	不使用稳定时期中断功能。
0	0	1	每 0.5 s 一次（与秒计数同步）
0	1	0	每 1 s 一次（与秒计数一致）
0	1	1	每 1 m1 次（每分钟的 00 秒）
1	0	0	每 1 小时一次（每小时的 00 分 00 秒）
1	0	1	每 1 天一次（每天的 00 小时 00 分 00 秒）
1	1	×	每 1 个月一次（每月的第 1 天上午 00 小时 00 分 00 秒）

若要在计数器工作（RTCE = 1）时改变 CT2 至 CT0 的值，需用中断屏蔽标志寄存器禁止中断服务 INTRTC 后，重写 CT2 至 CT0 的值。此外，在重写 CT2 至 CT0 的值后，清除 RIFG 和 RTCIF 标志使能中断服务。

注 RCLOE0 和 RCLOE2 不可同时使能。

注意事项 当 RTCE = 1 时如果改变了 RCLOE0 和 RCLOE1 的值，32.768 kHz 和 1 Hz 的输出信号的最后波形有可能变短。

备注 ×：无需理会

(3) 实时计数器控制寄存器 1 (RTCC1)

RTCC1 寄存器是 8 位寄存器，用于控制报警中断功能和计数器的等待时间。
可通过 1 位或 8 位的存储操作指令设置 RTCC1。
产生的复位信号将该寄存器清除为 00H。

图 7-4. 实时计数器控制寄存器 1 (RTCC1) 的格式 (1/2)

地址: FFF9EH 复位后: 00H R/W

符号	<7>	<6>	5	<4>	<3>	2	<1>	<0>
RTCC1	WALE	WALIE	0	WAFG	RIFG	0	RWST	RWAIT

<R>

WALE	报警操作控制
0	匹配操作无效。
1	匹配操作有效。
若要当计数器操作 (RTCE = 1) 且 WALIE = 1 时给 WALE 位设置一个值，需用中断屏蔽标志寄存器禁止中断服务 INTRTC 后重写 WALE 位。此外，重写 WALE 位后清除 WAFG 和 RTCIF 标志。若要设置各报警寄存器 (RTCC1 的 WALIE 标志、ALARMWM 寄存器、ALARMWH 寄存器以及 ALARMWW 寄存器)，需设置 WALE 位为匹配操作无效 ("0")。	

WALIE	报警中断 (INTRTC) 功能操作的控制
0	不生成与报警匹配的中断。
1	生成与报警匹配的中断。

WAFG	报警检测状态标志
0	报警不匹配
1	检测到与报警匹配
该状态标志表示与报警匹配的检测。该标志仅在 WALE = 1 时有效，而在检测到报警匹配后设置为“1” (1 个时钟 32.768 kHz)。向其写入“0”时该标志被清零。写入“1”无效。	

图 7-4. 实时计数器控制寄存器 1（RTCC1）的格式（2/2）

RIFG	稳定时期中断状态标志
0	不产生稳定时期中断。
1	产生稳定时期中断。
该标志表示稳定时期中断产生的状态。当稳定时期中断产生时，其被置为“1”。向其写入“0”时该标志被清零。写入“1”无效。	

RWST	实时计数器的等待状态
0	计数器正在工作。
1	读取计数器数值或向计数器写入数值的模式。
该状态标志表示 RWAIT 的设置是否有效。 读取或写入计数器数值前，确保该标志的值为 1。	

RWAIT	实时计数器的等待控制
0	设置计数器操作。
1	停止 SEC 至 YEAR 计数器。读取计数器数值或向计数器写入数值的模式。
该位控制计数器的操作。 确保向其写入“1”以读取或写入计数器数值。 因为 RSUBC 连续工作，所以在 1 秒内完成对其的读取或写入，且将该位清除为 0。 当 RWAIT = 1 时，直到计数器设置可被读取或写入占用的时间为 1 个时钟（32.768 kHz）。 当 RWAIT = 1 时如果 RSUBC 溢出，则在 RWAIT = 0 后其会向上计数。但是，如果写入秒计数寄存器，因为 RSUBC 被清除所以其不会向上计数。	

- 注意事项

如果用 1 位操作指令执行对 RTCC1 寄存器写入，则 RIFG 和 WAFG 标志有可能被清除。因此，若要执行对 RTCC1 寄存器的写入，务必使用 8 位操作指令。为避免在写入期间 RIFG 和 WAFG 标志被清除，禁止将相应位设置为 1 的写入。如果不使用 RIFG 和 WAFG 标志，而值可以被改变，则可使用 1 位操作指令向 RTCC1 寄存器写入。
- 备注

固定周期中断和报警匹配中断使用相同中断源（INTRTC）。当同时使用这两种中断时，可基于发生的 INTRTC 通过检查固定周期中断状态标志（RIFG）和报警检测状态标志（WAFG）判定发生的是哪种中断。

(4) 实时计数器控制寄存器 2 (RTCC2)

RTCC2 寄存器是 8 位寄存器，用于控制间隔中断功能和 RTCDIV 引脚。

可通过 1 位或 8 位的存储操作指令设置 RTCC2。

产生的复位信号将该寄存器清除为 00H。

图 7-5. 实时计数器控制寄存器 2 (RTCC2) 的格式

地址: FFF9FH 复位后: 00H R/W

符号	<7>	<6>	<5>	4	3	2	1	0
RTCC2	RINTE	RCLOE2	RCKDIV	0	0	ICT2	ICT1	ICT0

RINTE	ICT2	ICT1	ICT0	间隔中断 (INTRTCI) 的选择
0	x	x	x	不产生间隔中断。
1	0	0	0	$2^6/f_{XT}$ (1.953125 ms)
1	0	0	1	$2^7/f_{XT}$ (3.90625 ms)
1	0	1	0	$2^8/f_{XT}$ (7.8125 ms)
1	0	1	1	$2^9/f_{XT}$ (15.625 ms)
1	1	0	0	$2^{10}/f_{XT}$ (31.25 ms)
1	1	0	1	$2^{11}/f_{XT}$ (62.5 ms)
1	1	1	x	$2^{12}/f_{XT}$ (125 ms)

RCLOE2 ^注	RTCDIV 引脚输出控制
0	禁止 RTCDIV 引脚输出
1	允许 RTCDIV 引脚输出

RCKDIV	选择 RTCDIV 引脚的输出频率
0	RTCDIV 引脚输出 512 Hz。 (1.95 ms)
1	RTCDIV 引脚输出 16.384 kHz。 (0.061 ms)

注 RCLOE0 和 RCLOE2 不可同时使能。

注意事项 1. 在 RINTE = 0 时改变 ICT2, ICT1 和 ICT0 的值。

2. 当停止从 RTCDIV 引脚输出时，输出持续两个 f_{XT} 时钟长的最大值，而后进入低电平。当正在输出 512 Hz 的频率时，若在进入高电平后立即停止输出，则有可能产生一个至少有一个 f_{XT} 时钟宽的脉冲。

3. 在实时计数器启动工作后，RTCDIV 引脚的输出宽度可能比在第一个间隔期间设置的宽度短。

<R>

(5) 副计数寄存器 (RSUBC)

RSUBC 寄存器是 16 位寄存器，用于计数实时计数器的 1 秒的参考时间。通常情况下，计数值的范围为 0000H 至 7FFFH，用 32.768 kHz 时钟计数一秒。

RSUBC 可用 16 位存储操作指令设置。

产生的复位信号将该寄存器清除为 0000H。

- 注意事项
- 1. 当用 SUBCUD 寄存器进行校正时，该值可能变成 8000H 或更大的数值。
 - 2. 该寄存器也可通过写入秒计数寄存器进行有效复位而清零。
 - 3. 如果操作期间读取该寄存器，则不能保证读取值，因为读取到的是正在变化的数值。

图 7-6. 副计数寄存器 (RSUBC) 的格式

地址: FFF90H 复位后: 0000H R

符号	7	6	5	4	3	2	1	0
RSUBC	SUBC7	SUBC6	SUBC5	SUBC4	SUBC3	SUBC2	SUBC1	SUBC0

地址: FFF91H 复位后: 0000H R

符号	7	6	5	4	3	2	1	0
RSUBC	SUBC15	SUBC14	SUBC13	SUBC12	SUBC11	SUBC10	SUBC9	SUBC8

(6) 秒计数寄存器 (SEC)

SEC 寄存器是 8 位寄存器，从 0 到 59（十进制）中取值，表示秒的计数值。

当副计数器溢出时其向上计数。

当数据写入该寄存器时，其先被写入缓存，最多两个时钟（32.768 kHz）过后被写入计数器。给该寄存器以 BCD 码的格式设置一个 00 到 59 的十进制数。

SEC 可用 8 位存储操作指令设置。

产生的复位信号将该寄存器清除为 00H。

图 7-7. 秒计数寄存器 (SEC) 的格式

地址: FFF92H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
SEC	0	SEC40	SEC20	SEC10	SEC8	SEC4	SEC2	SEC1

(7) 分计数寄存器 (MIN)

MIN 寄存器是 8 位寄存器，从 0 到 59（十进制）中取值，表示分的计数值。

当秒计数器溢出时其向上计数。

当数据写入该寄存器时，其先被写入缓存，最多两个时钟（32.768 kHz）过后被写入计数器。当该寄存器被写入时如果秒计数寄存器溢出，则该寄存器忽略溢出而设置为写入的数值。给该寄存器以 BCD 码的格式设置一个 00 到 59 的十进制数。

MIN 可用 8 位存储操作指令设置。

产生的复位信号将该寄存器清除为 00H。

图 7-8. 分计数寄存器 (MIN) 的格式

地址：FFF93H 复位后：00H R/W

符号	7	6	5	4	3	2	1	0
MIN	0	MIN40	MIN20	MIN10	MIN8	MIN4	MIN2	MIN1

(8) 小时计数寄存器 (HOUR)

HOUR 寄存器是 8 位寄存器，从 00 到 23 或 01 到 12 或 21 到 32（十进制）中取值，表示小时的计数值。

当分计数器溢出时其向上计数。

当数据写入该寄存器时，其先被写入缓存，最多两个时钟（32.768 kHz）过后被写入计数器。当该寄存器被写入时如果分计数寄存器溢出，则该寄存器忽略溢出而设置为写入的数值。以 BCD 码的格式给该寄存器设置 00 至 23、01 至 12 或 21 至 32 中的十进制数值。

HOUR 可用 8 位存储操作指令设置。

产生的复位信号将该寄存器清除为 12H。

复位后如果 AMPM 位（RTCC0 寄存器的位 3）设置为 1 则该寄存器的值为 00H。

图 7-9. 小时计数寄存器 (HOUR) 的格式

地址：FFF94H 复位后：12H R/W

符号	7	6	5	4	3	2	1	0
HOUR	0	0	HOUR20	HOUR10	HOUR8	HOUR4	HOUR2	HOUR1

注意事项 如果 AMPM = 0（如果选择 12 小时的系统）HOUR 寄存器的位 5（HOUR20）表示 AM（0）/PM（1）。

表 7-2 展示了 AMPM 位的设定值、HOUR 寄存器的值以及时间之间的关系。

表 7-2. 显示的时间位数

24 小时显示 (AMPM 位 = 1)		12 小时显示 (AMPM 位 = 1)	
时间	HOUR 寄存器	时间	HOUR 寄存器
0	00H	0 a.m.	12H
1	01H	1 a.m.	01H
2	02H	2 a.m.	02H
3	03H	3 a.m.	03H
4	04H	4 a.m.	04H
5	05H	5 a.m.	05H
6	06H	6 a.m.	06H
7	07H	7 a.m.	07H
8	08H	8 a.m.	08H
9	09H	9 a.m.	09H
10	10H	10 a.m.	10H
11	11H	11 a.m.	11H
12	12H	0 p.m.	32H
13	13H	1 p.m.	21H
14	14H	2 p.m.	22H
15	15H	3 p.m.	23H
16	16H	4 p.m.	24H
17	17H	5 p.m.	25H
18	18H	6 p.m.	26H
19	19H	7 p.m.	27H
20	20H	8 p.m.	28H
21	21H	9 p.m.	29H
22	22H	10 p.m.	30H
23	23H	11 p.m.	31H

当 AMPM 位为 “0” 时 HOUR 寄存器的值设置为按 12 小时显示，当 AMPM 位为 “1” 时 HOUR 寄存器的值设置为按 24 小时显示。

按 12 小时显示时，HOUR 寄存器的第五位显示 0 表示 AM，显示 1 表示 PM。

(9) 天计数寄存器 (DAY)

DAY 寄存器是 8 位寄存器，从 1 到 31（十进制）中取值，表示天的计数值。
当小时计数器溢出时其向上计数。
该计数器进行以下计数：

- 01 到 31（一月、三月、五月、七月、八月、十月、十二月）
- 01 到 30（四月、六月、九月、十一月）
- 01 到 29（闰年中的二月）
- 01 到 28（正常年中的二月）

<R>

当数据写入该寄存器时，其先被写入缓存，最多两个时钟（32.768 kHz）过后被写入计数器。当该寄存器被写入时如果小时计数器寄存器溢出，则该寄存器忽略溢出而设置为写入的数值。给该寄存器以 BCD 码的格式设置一个 01 到 31 的十进制数。
DAY 可用 8 位存储操作指令设置。
产生的复位信号将该寄存器清除为 01H。

图 7-10. 天计数寄存器 (DAY) 的格式

地址: FFF96H	复位后: 01H	R/W						
符号	7	6	5	4	3	2	1	0
DAY	0	0	DAY20	DAY10	DAY8	DAY4	DAY2	DAY1

(10) 星期计数寄存器 (WEEK)

WEEK 寄存器是 8 位寄存器，从 0 到 6（十进制）中取值，表示星期几的计数值。
它与天计数器同步向上计数。
当数据写入该寄存器时，其先被写入缓存，最多两个时钟（32.768 kHz）过后被写入计数器。给该寄存器以 BCD 码的格式设置一个 00 到 06 的十进制数。
WEEK 可用 8 位存储操作指令设置。
产生的复位信号将该寄存器清除为 00H。

图 7-11. 星期计数寄存器 (WEEK) 的格式

地址: FFF95H	复位后: 00H	R/W						
符号	7	6	5	4	3	2	1	0
WEEK	0	0	0	0	0	WEEK4	WEEK2	WEEK1

注意事项 与月计数寄存器或天计数寄存器对应的值不自动存入星期计数寄存器。复位解除后按如下设置星期计数寄存器。

天	星期
星期天	00H
星期一	01H
星期二	02H
星期三	03H
星期四	04H
星期五	05H
星期六	06H

(11) 月计数寄存器 (MONTH)

MONTH 寄存器是 8 位寄存器，从 1 到 12（十进制）中取值，表示月的计数值。

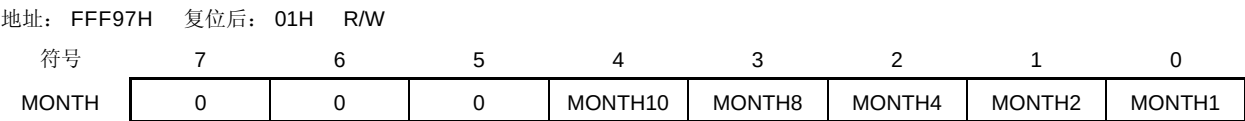
当天计数器溢出时其向上计数。

当数据写入该寄存器时，其先被写入缓存，最多两个时钟（32.768 kHz）过后被写入计数器。当该寄存器被写入时如果按天计数寄存器溢出，则该寄存器忽略溢出而设置为写入的数值。给该寄存器以 BCD 码的格式设置一个 01 到 12 的十进制数。

MONTH 可用 8 位存储操作指令设置。

产生的复位信号将该寄存器清除为 01H。

图 7-12. 月计数寄存器 (MONTH) 的格式



(12) 年计数寄存器 (YEAR)

YEAR 寄存器是 8 位寄存器，从 0 到 99（十进制）中取值，表示年的计数值。

当月计数器溢出时其向上计数。

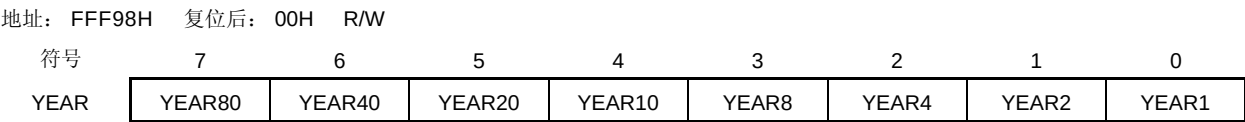
00, 04, 08, ..., 92 和 96 值表示闰年。

当数据写入该寄存器时，其先被写入缓存，最多两个时钟（32.768 kHz）过后被写入计数器。当该寄存器被写入时如果月计数寄存器溢出，则该寄存器忽略溢出而设置为写入的数值。给该寄存器以 BCD 码的格式设置一个 00 到 99 的十进制数。

YEAR 可用 8 位存储操作指令设置。

产生的复位信号将该寄存器清除为 00H。

图 7-13. 年计数寄存器 (YEAR) 的格式



产生的复位信号将该寄存器清除为 00H.

地址: FFF99H 复位后: 00H R/W

F6	钟表错误校正值的设置
0	增加{ (F5, F4, F3, F2, F1, F0) - 1} × 2。
1	减少{ (F5, F4, F3, F2, F1, F0) + 1} × 2。

当 (F6, F5, F4, F3, F2, F1, F0) = (*, 0, 0, 0, 0, 0, *) 时, 不可校正钟表错误。*表示 0 或 1。

F5 至 F0 是相应位的反向值 (当为 111100 时即 000011)。

校正值的范围:

(当 F6 = 0 时) 2, 4, 6, 8, ... , 120, 122, 124

(当 F6 = 1 时) -2, -4, -6, -8, ... , -120, -122, -124

使用钟表错误校正寄存器 (SUBCUD) 可校正的值的范围如下所示:

	DEV = 0 （每20 秒校正一次）	DEV = 1 （每 60 秒校正一次）
可校正范围	−189.2 ppm 至 189.2 ppm	−63.1 ppm 至 63.1 ppm
最大 量化误差	. 1.53 ppm	. 0.51 ppm
最小分辨率	. 3.05 ppm	. 1.02 ppm

备注 如果可校正范围为-63.1 ppm 或更低值以及 63.1 ppm 或更高值，将 DEV 设置为 0。

(14) 分报警寄存器 (ALARMWM)

该寄存器用于设置报警分钟。

ALARMWM 可用 8 位存储操作指令设置。

产生的复位信号将该寄存器清除为 00H。

注意事项 给该寄存器以 BCD 码的格式设置一个 00 到 59 的十进制数。如果设置值超出范围，则检测不到报警。

图 7-15. 分报警寄存器 (ALARMWM) 的格式

地址: FFF9AH 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
ALARMWM	0	WM40	WM20	WM10	WM8	WM4	WM2	WM1

(15) 小时报警寄存器 (ALARMWH)

该寄存器用于设置报警小时。

ALARMWH 可用 8 位存储操作指令设置。

产生的复位信号将该寄存器清除为 12H。

复位后如果 AMPM 位 (RTCC0 寄存器的位 3) 设置为 1 则该寄存器的值为 00H。

注意事项 以 BCD 码的格式给该寄存器设置 00 至 23、01 至 12 或 21 至 32 中的十进制数值。如果设置值超出范围，则检测不到报警。

图 7-16. 小时报警寄存器 (ALARMWH) 的格式

地址: FFF9BH 复位后: 12H R/W

符号	7	6	5	4	3	2	1	0
ALARMWH	0	0	WH20	WH10	WH8	WH4	WH2	WH1

注意事项 如果 AMPM = 0 (如果选择 12 小时的系统) ALARMWH 寄存器的位 5 (WH20) 表示 AM (0) /PM (1)。

(16) 星期报警寄存器 (ALARMWW)

该寄存器用于设置报警日期。

ALARMWW 可用 8 位存储操作指令设置。

产生的复位信号将该寄存器清除为 00H。

图 7-17. 星期报警寄存器 (ALARMWW) 的格式

地址: FFF9CH 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
ALARMWW	0	WW6	WW5	WW4	WW3	WW2	WW1	WW0

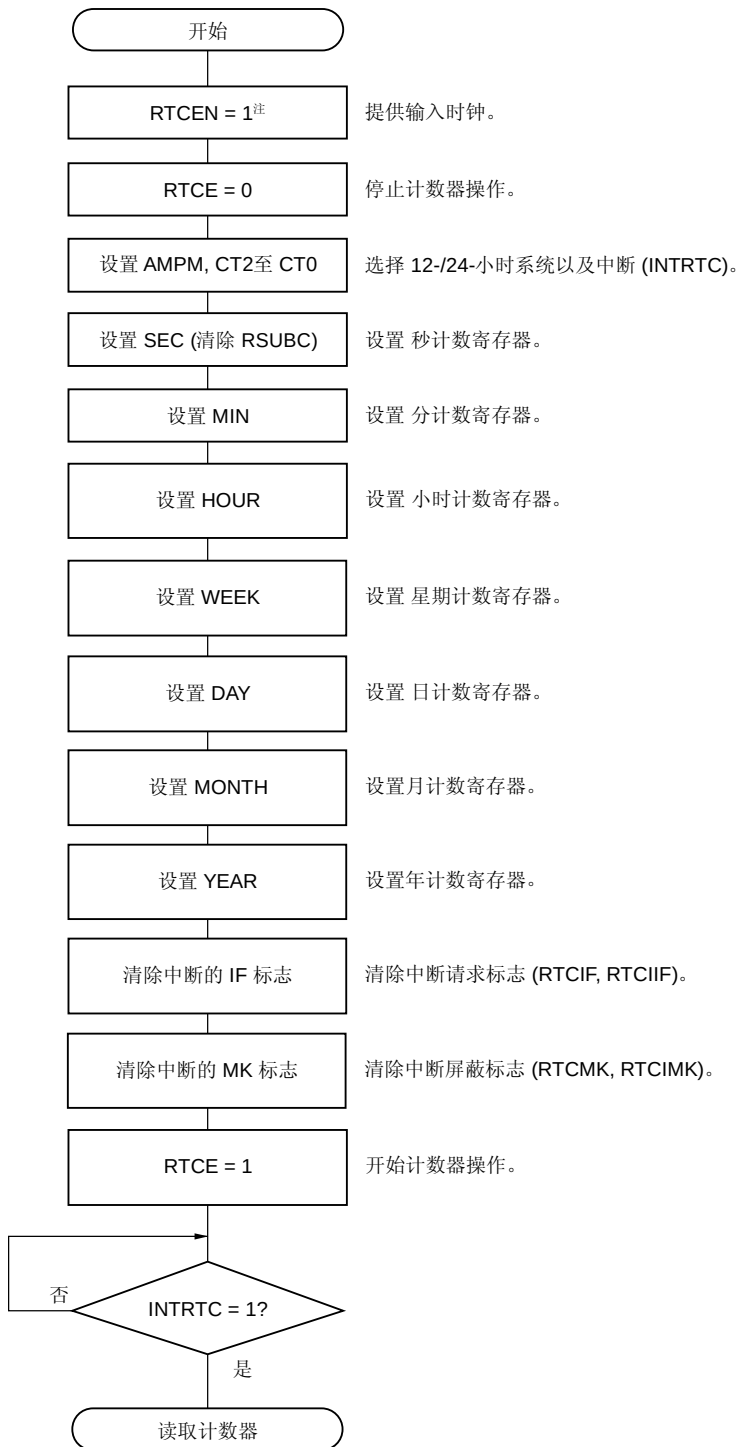
下面设置报警的示例。

报警的时间	天							12 小时显示				24 小时显示			
	星期天	星期一	星期二	星期三	星期四	星期五	星期六	小时	小时	分	分	小时	小时	分	分
	W	W	W	W	W	W	W	10	1	10	1	10	1	10	1
	W	W	W	W	W	W	W								
每天, 0: 00 a.m.	0	1	2	3	4	5	6								
每天, 1: 30 a.m.	1	1	1	1	1	1	1	1	2	0	0	0	0	0	0
每天, 11: 59 a.m.	1	1	1	1	1	1	1	0	1	3	0	0	1	3	0
每天, 11: 59 a.m.	1	1	1	1	1	1	1	1	1	5	9	1	1	5	9
星期一至星期五, 0: 00 p.m.	0	1	1	1	1	1	0	3	2	0	0	1	2	0	0
星期天, 1: 30 p.m.	1	0	0	0	0	0	0	2	1	3	0	1	3	3	0
星期一、星期三、星期五, 11: 59 p.m.	0	1	0	1	0	1	0	3	1	5	9	2	3	5	9

7.4 实时计数器操作

7.4.1 实时计数器的启动操作

图 7-18. 实时计数器的启动操作步骤



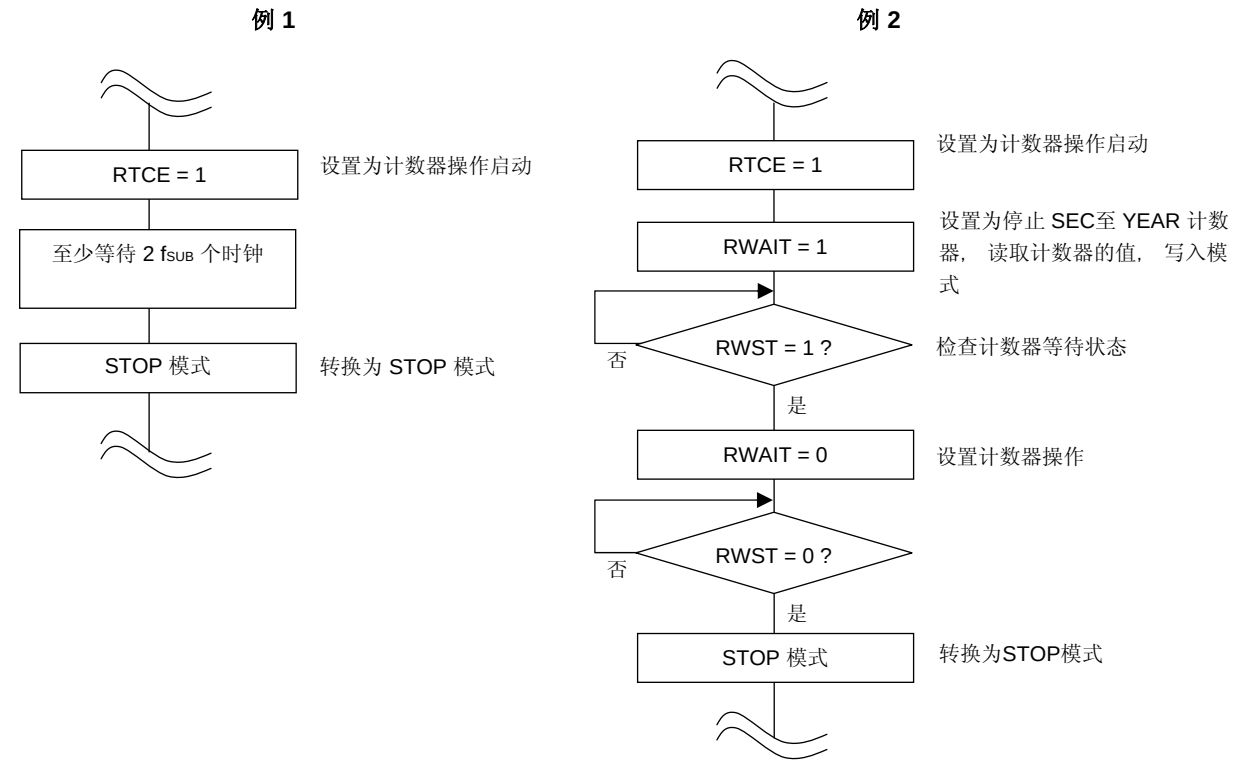
注 当副系统时钟 (f_{SUB}) 振荡稳定时先设置 RTCEN 为 1。

7.4.2 启动操作后转换至 STOP 模式

当设置 RTCE 为 1 后立即转换为 STOP 模式时，需要执行以下处理中的一种。
但是，将 RTCE 设置为 1 后，在第一个 INTRTC 中断发生后转换到 STOP 模式后不需要进行该处理。

- 在设置 RTCE 为 1 后（见图 7-19，例 1）至少经过两个副系统时钟（ f_{SUB} ）（大约 62 μ s）后，转换到 STOP 模式。
- 在设置 RTCE 为 1 后，轮询检测到 RWST 变成 1，而后设置 RWAIT 为 1。然后，轮询检测到 RWST 变成 0 后设置 RWAIT 为 0，转换到 STOP 模式。

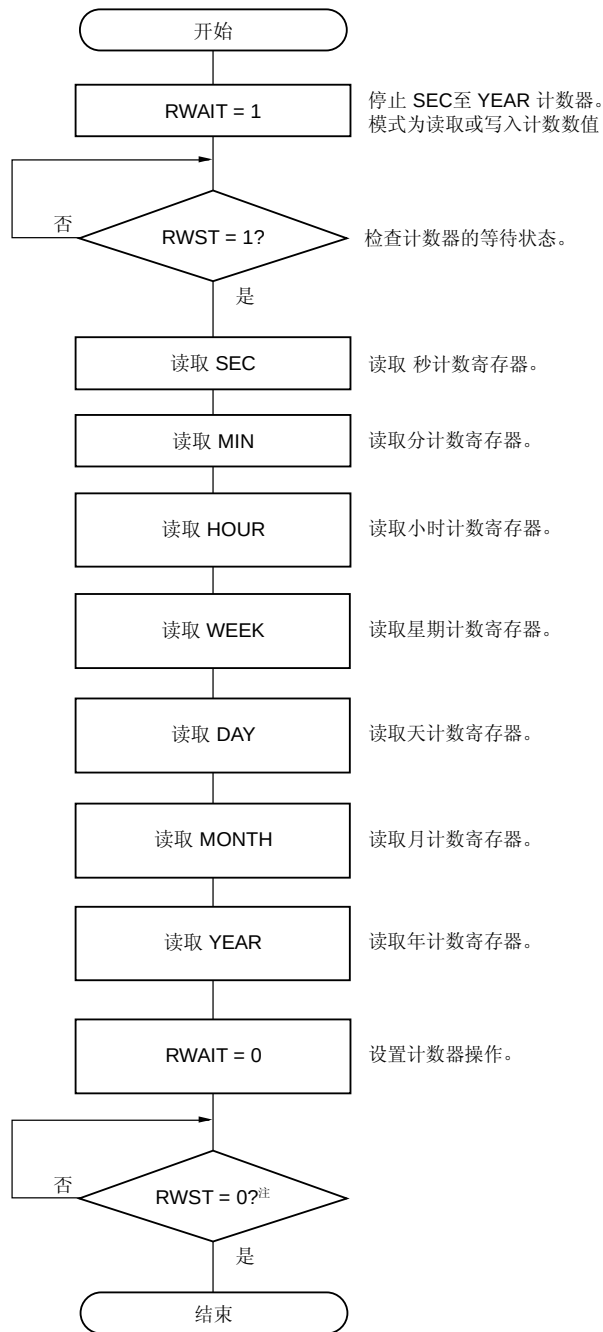
图 7-19. 设置 RTCE 为 1 后转换至 STOP 模式的步骤



7.4.3 读取/写入实时计数器

在先将 1 设置到 RWAIT 后读取或写入计数器。

图 7-20. 读取实时计数器的步骤

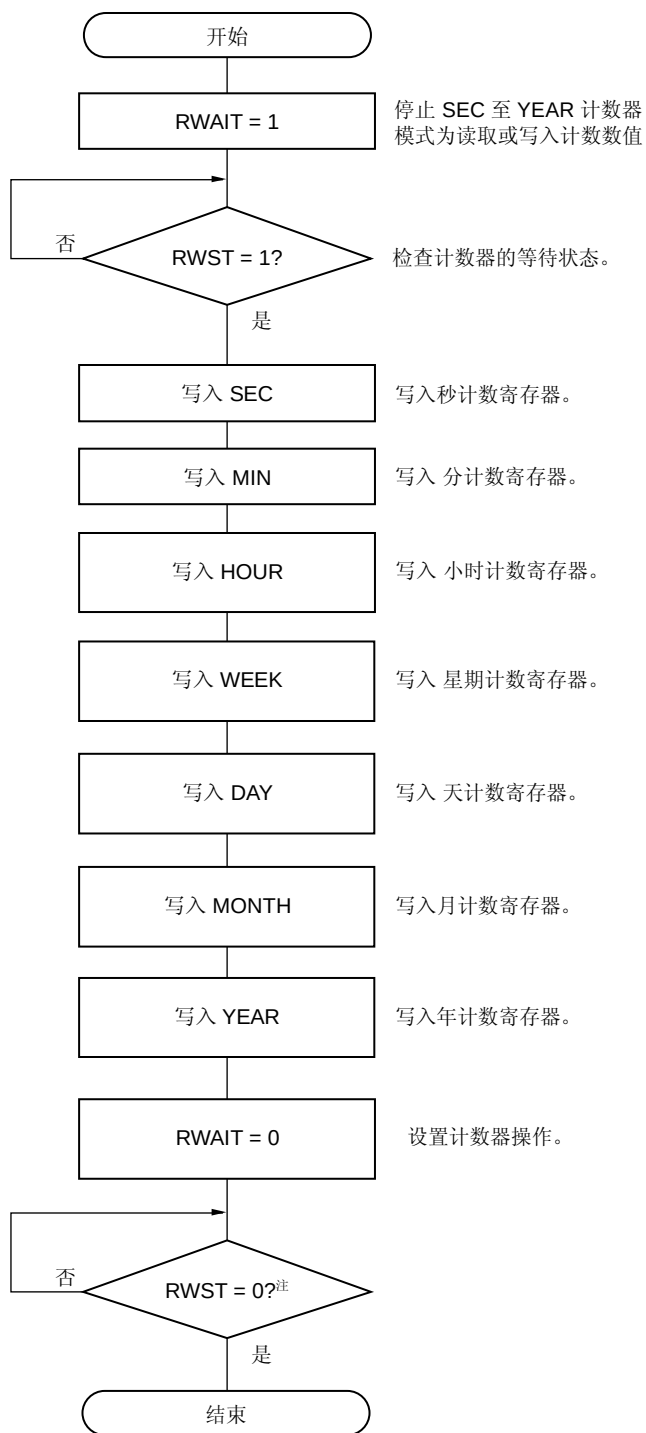


注 设置 STOP 模式前确保 RWST = 0。

注意事项 在 1 秒内完成将 RWAIT 设置为 1 以清除 RWAIT 为 0 的系列操作。

备注 SEC, MIN, HOUR, WEEK, DAY, MONTH 和 YEAR 可按任意顺序读取。
没必要设置所有的寄存器，只有其中一部分可被读取。

图 7-21. 写入实时计数器的步骤



注 设置 STOP 模式前确保 RWST = 0。

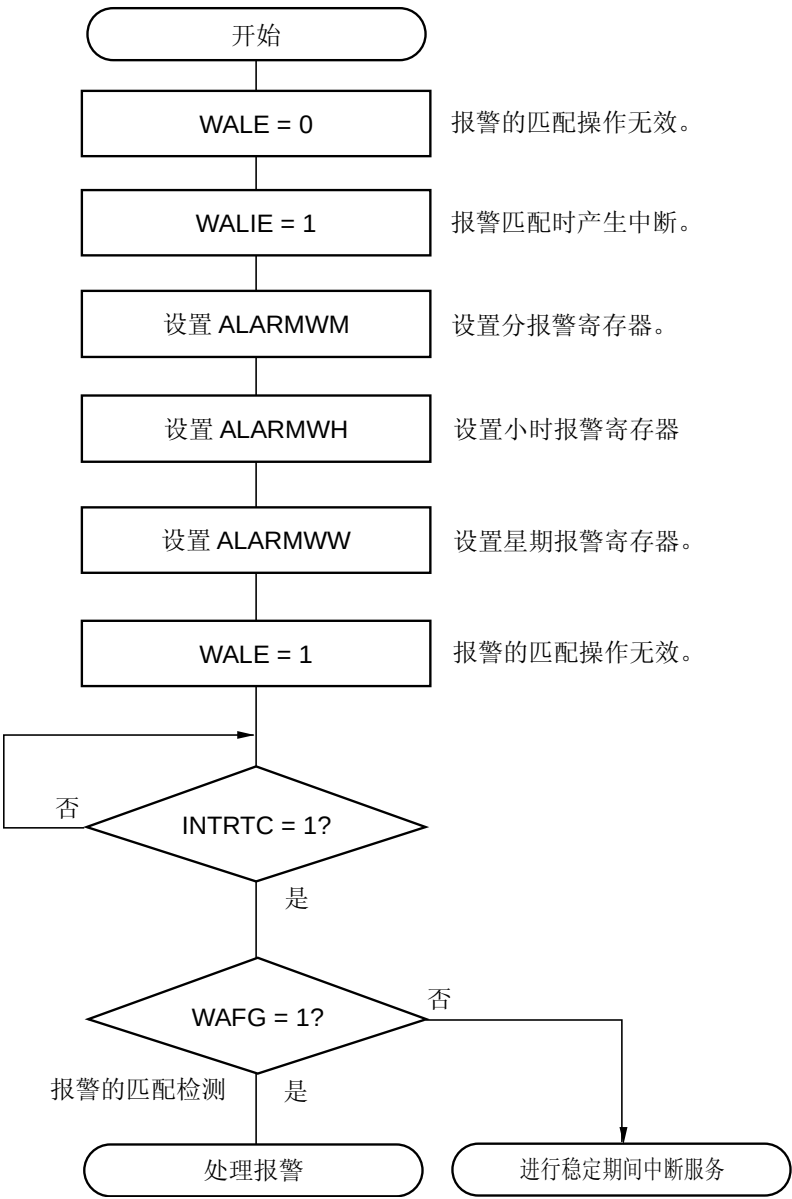
注意事项 在 1 秒内完成将 RWAIT 设置为 1 以清除 RWAIT 为 0 的系列操作。

备注 SEC, MIN, HOUR, WEEK, DAY, MONTH 和 YEAR 可按任意顺序写入。
没必要设置所有的寄存器，只有其中一部分可被写入。

7.4.4 实时计数器的报警设置

在先将 0 设置到 WALE 后再设置报警的时间。

图 7-22.报警设置步骤

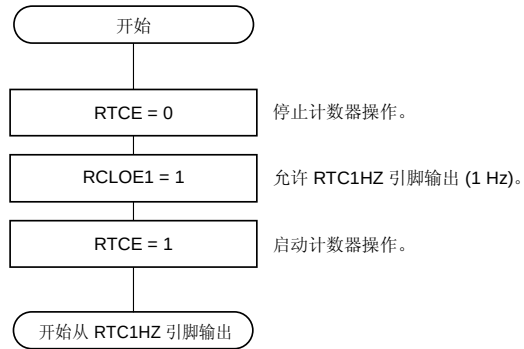


备注

1. ALARMWMM，ALARMWH 和 ALARMWW 可按任意顺序写入。
2. 固定周期中断和报警匹配中断使用相同中断源（INTRTC）。当同时使用这两种中断时，可基于发生的 INTRTC 通过检查固定周期中断状态标志（RIFG）和报警检测状态标志（WAFG）判定发生的是哪种中断。

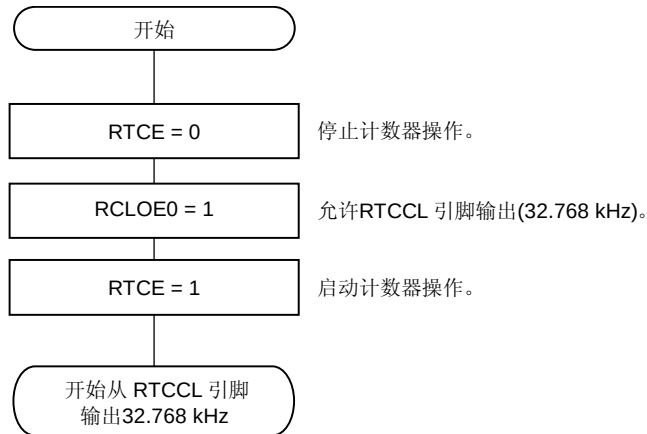
7.4.5 实时计数器的 1 Hz 输出设置

图 7-23.1 Hz 输出设置步骤



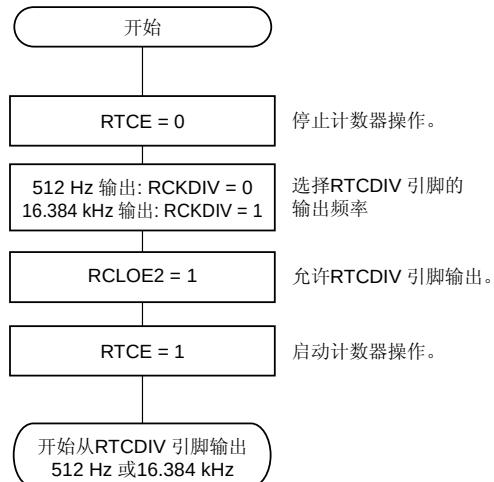
7.4.6 实时计数器的 32.768 kHz 输出设置

图 7-24 32.768kHz 输出设置步骤



7.4.7 实时计数器的 512 Hz, 16.384 kHz 输出设置

图 7-7-25. 512 Hz, 16.384 kHz 输出设置步骤



7.4.8 实时计数器钟表错误校正的举例

当钟表变慢或变快时，通过给钟表错误校正寄存器设置一个数值，即可对钟表进行高精度的校正。

计算校正值的举例

用以下表达式计算校正值，而该校正值在校正副计数寄存器（RSUBC）的计数值时使用。
当校正范围为-63.1 ppm 或更低值，以及 63.1 ppm 或更高值时，将 DEV 设置为 0。

（当 DEV = 0 时）

校正值^注 = 1 分钟内校正计数的数目 ÷ 3 = （振荡频率 ÷ 目标频率 - 1） × 32768 × 60 ÷ 3

（当 DEV = 1 时）

校正值^注 = 1 分钟内校正计数的数目 = （振荡频率 ÷ 目标频率 - 1） × 32768 × 60

注 校正值是通过使用钟表错误校正寄存器（SUBCUD）的位 6 至 0 计算得到的钟表错误校正值。

（当 F6 = 0 时） 校正值 = {（F5, F4, F3, F2, F1, F0） - 1} × 2
（当 F6 = 1 时） 校正值 = -{（/F5, /F4, /F3, /F2, /F1, /F0） + 1} × 2

当（F6, F5, F4, F3, F2, F1, F0）为（*, 0, 0, 0, 0, 0, *）时，不执行钟表错误校正。
“*”表示 0 或 1。
/F5 至 /F0 是对应位的反向值（当为 111100 时即 000011）。

- 备注**
- 1. 校正值是 2, 4, 6, 8, ... 120, 122, 124 或 -2, -4, -6, -8, ... -120, -122, -124。
 - 2. 振荡频率是副系统时钟（f_{SUB}）。
当钟表错误校正寄存器设置为初始值（00H）时，可通过 RTCCL 引脚的 32 kHz 输出频率或 RTC1HZ 引脚的输出频率 × 32768 计算出振荡频率。
 - 3. 目标频率是使用钟表错误校正寄存器执行校正后的结果频率。

校正举例 <1>

从 32772.3 Hz 校正到 32768 Hz 的举例（32772.3 Hz – 131.2 ppm）

[测量振荡频率]

当钟表错误校正寄存器设置为初始值（00H）时，可通过 RTCCL 引脚输出的大约 32 kHz 或 RTC1HZ 引脚输出的大约 1 Hz 的频率测量各产品的振荡频率^註。

关于 RTC1HZ 引脚输出大约 1 Hz 频率的设置步骤参见 7.4.5 实时计数器的 1 Hz 输出设置，而关于 RTCCL 引脚输出大约 32 kHz 频率的设置步骤参见 7.4.6 实时计数器的 32.768kHz 输出设置。

[计算校正值]

（当 RTCCL 引脚输出频率为 32772.3 Hz 时）

如果目标频率假设为 32768 Hz（32772.3 Hz – 131.2 ppm），–131.2 ppm 的校正范围小于(63.1 ppm，所以假设 DEV 为 0

当 DEV 为 0 时应用以下表达式计算校正值：

$$\begin{aligned}\text{校正值} &= 1 \text{ 分钟内校正计数的数目} \div 3 \\ &= (\text{振荡频率} \div \text{目标频率} - 1) \times 32768 \times 60 \div 3 \\ &= (32772.3 \div 32768 - 1) \times 32768 \times 60 \div 3 \\ &= 86\end{aligned}$$

[计算设置到（F6 至 F0）的值]

（当校正值是 86 时）

如果校正值为 0 或更大值（延迟时的值），假设 F6 为 0。

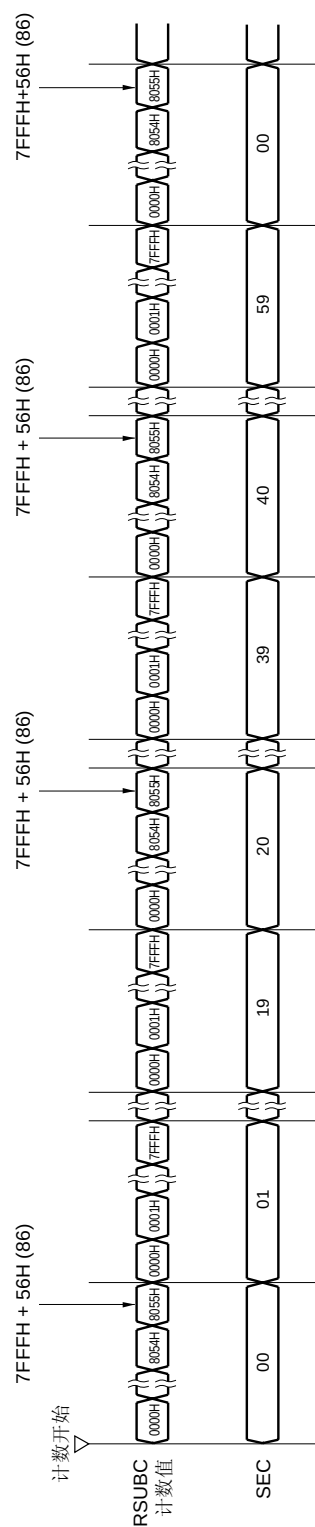
从校正值计算（F5，F4，F3，F2，F1，F0）。

$$\begin{aligned}\{ (F5, F4, F3, F2, F1, F0) - 1 \} \times 2 &= 86 \\ (F5, F4, F3, F2, F1, F0) &= 44 \\ (F5, F4, F3, F2, F1, F0) &= (1, 0, 1, 1, 0, 0)\end{aligned}$$

所以，当从 32772.3 Hz 校正到 32768 Hz（32772.3 Hz – 131.2 ppm）时，设置校正寄存器使得 DEV 为 0，而校正值为 86（SUBCUD 的位 6 至 0：0101100）最终使输出 32768 Hz（0 ppm）。

图 7-26 显示当（DEV，F6，F5，F4，F3，F2，F1，F0）是（0，0，1，0，1，1，0，0）时的操作。

图 7-26 当 (DEV, F6, F5, F4, F3, F2, F1, F0) = (0, 0, 1, 0, 1, 1, 0, 0) 时的操作



校正举例 <2>

从 32767.4 Hz 校正到 32768 Hz 的举例 (32767.4 Hz + 18.3 ppm)

[测量振荡频率]

当钟表错误校正寄存器设置为初始值 (00H) 时, 可通过 RTCCL 引脚输出的大约 32 kHz 或 RTC1HZ 引脚输出的大约 1 Hz 的频率测量各产品的振荡频率^註。

关于 RTC1HZ 引脚输出大约 1 Hz 频率的设置步骤参见 7.4.5 实时计数器的 1 Hz 输出设置, 而关于 RTCCL 引脚输出大约 32 kHz 频率的设置步骤参见 7.4.6 实时计数器的 32.768kHz 输出设置。

[计算校正值]

(当 RTCCL 引脚输出频率为 0.9999817 Hz 时)

振荡频率 = $32768 \times 0.9999817 \approx 32767.4 \text{ Hz}$

假设目标频率为 32768 Hz (32767.4 Hz + 18.3 ppm) 而 DEV 为 1。

当 DEV 为 1 时应用以下表达式计算校正值:

$$\begin{aligned} \text{校正值} &= 1 \text{ 分钟内校正计数的数目} \\ &= (\text{振荡频率} \div \text{目标频率} - 1) \times 32768 \times 60 \\ &= (32767.4 \div 32768 - 1) \times 32768 \times 60 \\ &= -36 \end{aligned}$$

[计算设置到 (F6 至 F0) 的值]

(当校正值是 -36 时)

如果校正值为 0 或更小值 (加快时的值), 假设 F6 为 1。

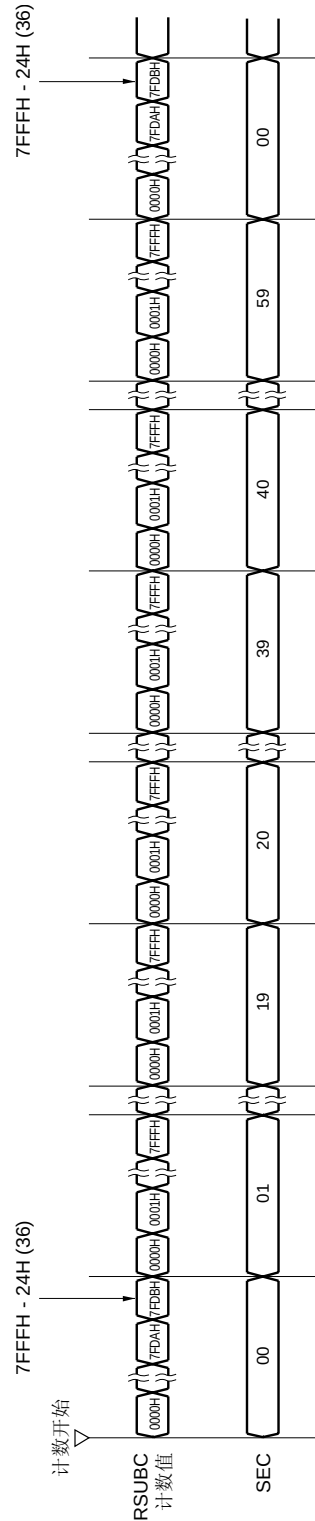
从校正值计算 (F5, F4, F3, F2, F1, F0)。

$$\begin{aligned} -\{ (/F5, /F4, /F3, /F2, /F1, /F0) - 1 \} \times 2 &= -36 \\ (/F5, /F4, /F3, /F2, /F1, /F0) &= 17 \\ (/F5, /F4, /F3, /F2, /F1, /F0) &= (0, 1, 0, 0, 0, 1) \\ (F5, F4, F3, F2, F1, F0) &= (1, 0, 1, 1, 1, 0) \end{aligned}$$

所以, 当从 32767.4 Hz 校正到 32768 Hz (32767.4 Hz + 18.3 ppm) 时, 设置校正寄存器使得 DEV 为 1, 而校正值为 -36 (SUBCUD 的位 6 至 0: 1101110) 最终使输出 32768 Hz (0 ppm)。

图 7-27 显示当 (DEV, F6, F5, F4, F3, F2, F1, F0) 是 (1, 1, 1, 0, 1, 1, 1, 0) 时的操作。

图 7-27 当 (DEV, F6, F5, F4, F3, F2, F1, F0) = (1, 1, 1, 0, 1, 1, 1, 0) 时的操作



8.1 比较器和可编程增益放大器的特性

可编程增益放大器和比较器的特性如下所示：

○ 比较器

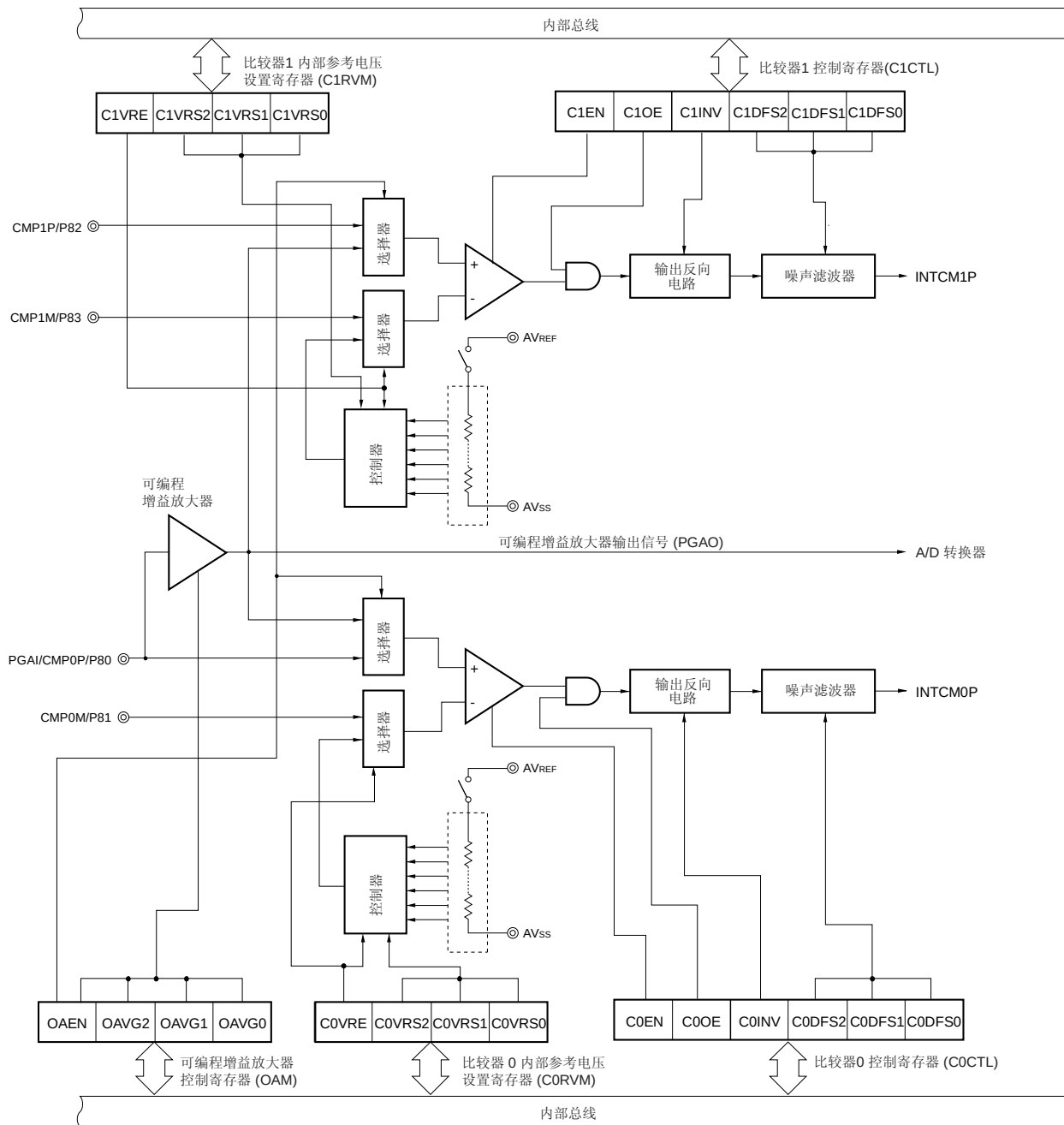
- 一个比较器配备有两个通道（CMP0、CMP1）。
- 可以连接负输入引脚（CMP0M、CMP1M）和正输入引脚（CMP0P、CMP1P）。
- 可编程增益放大器的输出信号可用作比较器的正输入信号^注。
- CMP0M 和 CMP1M 引脚输入以及内部生成参考电压（每个比较器的 6 个组合）可选作为参考电压。
- 可以选择噪声抑制数字滤波器的抑制宽度。
- 当超出参考电压（INTCMP0，INTCMP1）时产生中断请求。

○ 可编程增益放大器

- 可编程增益放大器放大并输出所输入的模拟电压。可以选择五个放大系数中的任一个。
- 可编程增益放大器的输出信号可用作比较器的正输入信号^注。
- 可编程增益放大器的输出信号可选作 A/D 转换器的模拟输入。

注 将可编程增益放大器的输出信号用作比较器的正输入信号时，该输出信号同时输入到比较器的通道 0 和通道 1。

图 8-1. 比较器/可编程增益放大器的框图



8.2 比较器和可编程增益放大器的配置

比较器和可编程增益放大器包括以下硬件。

表 8-1. 比较器和可编程增益放大器的配置

项目	配置
控制寄存器	外围使能寄存器1（PER1） 可编程增益放大器控制寄存器（OAM） 比较器 0 和 1 控制寄存器（C0CTL， C1CTL） 比较器 0 和 1 内部参考电压设置寄存器（C0RVM， C1RVM） 端口输入模式寄存器 8（PIM8） 端口模式寄存器 8 （PM8）

8.3 控制比较器和可编程增益放大器的寄存器

比较器和可编程增益放大器使用以下八种寄存器。

- 外围使能寄存器 1（PER1）
- 可编程增益放大器控制寄存器（OAM）
- 比较器 0 和 1 控制寄存器（C0CTL， C1CTL）
- 比较器 0 和 1 内部参考电压设置寄存器（C0RVM， C1RVM）
- 端口输入模式寄存器 8（PIM8）
- 端口模式寄存器 8 （PM8）

(1) 外围使能寄存器 1 (PER1)

该寄存器用于设置能否使用各个外围硬件宏功能。通过停止对未使用硬件的时钟供应，可减少功耗和噪声。
确保将位 3 (OACMPEN) 置为 1，以使用比较器或可编程增益放大器。

可通过 1 位或 8 位的存储操作指令设置 PER1。

产生的复位信号将该寄存器清除为 00H。

- 注意事项**
- 1. 设置比较器或可编程增益放大器时，确保先将 OACMPEN 设置为 1。当 OACMPEN 设置为 0 时，忽略对比较器或可编程增益放大器的控制寄存器的写入，并且读取值被初始化。
 - 2. 确保将 PER1 寄存器的位 0 至 2 以及位 4 至 7 设置为“0”。

图 8-2. 外围使能寄存器 1 (PER1) 的格式

地址：F00F1H 复位后：00H R/W

符号	7	6	5	4	<3>	2	1	0
PER1	0	0	0	0	OACMPEN	0	0	0

OACMPEN	比较器和可编程增益放大器输入时钟供应的控制
0	停止输入时钟的供应 - 比较器和可编程增益放大器使用的 SFR 不能被写入。 - 比较器和可编程增益放大器处于复位状态。
1	供应输入时钟 比较器和可编程增益放大器使用的 SFR 可以被写入。

(2) 可编程增益放大器控制寄存器 (OAM)

该寄存器用于启动或禁止可编程增益放大器的操作，并设置放大系数。
可通过 1 位或 8 位的存储操作指令设置 OAM。
产生的复位信号将该寄存器清除为 00H。

图 8-3. 可编程增益放大器控制寄存器 (OAM) 的格式

地址：F0240H 复位后：00H R/W

符号	<7>	6	5	4	3	2	1	0
OAM	OAEN	0	0	0	0	OAVG2	OAVG1	OAVG0

OAEN	可编程增益放大器操作控制
0	停止工作
1	允许工作 允许从可编程增益放大器输入引脚 (PGAI) 进行外部输入 将可编程增益放大器输出信号作为比较器0和1的正端输入电压输入

OAVG2	OAVG1	OAVG0	输入电压放大系数设置
0	0	1	×4
0	1	0	×6
0	1	1	×8
1	0	0	×10
1	0	1	×12
其它			禁止设置

- 注意事项
1. 在使能 (OAEN = 1) 可编程增益放大器的操作前设置放大系数。禁止在使能状态 (OAEN = 1) 改变放大系数的设置。
 2. 在设置 OAM 寄存器之后设置 CnCTL 寄存器。
 3. 若要将可编程增益放大器输出信号 (PGAO) 选作 A/D 转换器的模拟输入，则需设置 OAEN = 1，通过软件等待 3 μs，然后再启动 A/D 转换器 (ADCS = 1)。

备注 n = 0, 1

<R>

(3) 比较器 n 控制寄存器 (CnCTL)

该寄存器用于控制比较器 n 的操作、允许或禁止比较器输出、反向输出和设置噪声抑制宽度。
可通过 1 位或 8 位的存储操作指令设置 CnCTL。
产生的复位信号将该寄存器清除为 00H。

图 8-4. 比较器 n 控制寄存器 (CnCTL) 的格式

地址: F0241H (C0CTL), F0242H (C1CTL) 复位后: 00H R/W

符号	<7>	6	5	4	3	2	1	0
CnCTL	CnEN	0	0	CnOE	CnINV	CnDFS2	CnDFS1	CnDFS0

CnEN	比较器操作控制
0	停止工作
1	允许工作 允许输入到比较器n [±] 正端或负端的外部引脚

CnOE	允许或禁止比较器输出
0	禁止输出 (输出信号 = 固定为低电平)
1	允许输出

CnINV	输出反向设置
0	正向
1	反向

CnDFS2	CnDFS1	CnDFS0	噪声抑制宽度设置 (f _{CLK} = 20 MHz)
0	0	0	不使用噪声滤波器
0	0	1	250 ns
0	1	0	500 ns
0	1	1	1 μs
1	0	0	2 μs
其它			禁止设置

注 如果 OAEN = 1 (OAM 寄存器) 且 CnEN 设置为 1, 则可编程增益放大器输出信号将输入到比较器 n 的正端输入。

- 注意事项
1. 在将比较器输出设置为禁止状态 (CnOE = 0) 后才可重写 CnINV 和 CnDFS2 至 CnDFS0。
 2. 关于噪声抑制宽度, 一个额外的 CPU 时钟 (f_{CLK}) 有可能由设定值抑制。
(举例: 当 f_{CLK} = 20 MHz, CnDFS2 至 CnDFS0 = 001, 噪声抑制宽度 = 250 至 300 ns 时)
 3. 若要将比较器与可编程增益放大器联合工作, 则需在设置可编程增益放大器工作之后再设置比较器工作 (参见图 8-10 和图 8-11)
 4. 当 CnRVM 寄存器的 CnVRE 设置为 (1) 时, 无论允许或禁止比较器操作 (CnEN) 的值是多少, 比较器的负端外部引脚输入都被切断。
 5. 设置 CnEN = 1 然后通过软件等待 1 μs 之后, 使能中断信号。

- 备注
1. f_{CLK}: CPU 或外围硬件时钟频率
 2. n = 0, 1

(4) 比较器 n 内部参考电压选择寄存器 (CnRVM)

该寄存器用于设置比较器 n 的内部参考电压。内部参考电压可以从使用 AVREF 的六个电压中选择。
可通过 1 位或 8 位的存储操作指令设置 CnRVM。
产生的复位信号将该寄存器清除为 00H。

图 8-5. 比较器 n 内部参考电压选择寄存器 (CnRVM) 的格式

地址：F0243H (C0RVM)，F0244H (C1RVM) 复位后：00H R/W

符号	<7>	6	5	4	3	2	1	0
CnRVM	CnVRE	0	0	0	0	CnVRS2	CnVRS1	CnVRS0

CnVRE	内部参考电压操作控制
0	停止工作
1	允许工作 将内部参考电压连接到比较器n的负端输入

CnVRS2	CnVRS1	CnVRS0	参考电压设置	
			比较器 0 (n = 0) 可设置的参考电压	比较器 1 (n = 1) 可设置的参考电压
0	0	0	禁止设置	
0	0	1	2AVREF/16	3AVREF/16
0	1	0	4AVREF/16	5AVREF/16
0	1	1	6AVREF/16	7AVREF/16
1	0	0	8AVREF/16	9AVREF/16
1	0	1	10AVREF/16	11AVREF/16
1	1	0	12AVREF/16	13AVREF/16
1	1	1	禁止设置	

- 注意事项
1. 当内部参考电压的工作停止 (CnVRE = 0) 时，比较器的操作由 CnEN 控制。
 2. 当 CnVRE 设置为 (1) 时，无论允许或禁止比较器操作 (CnEN) 的值是多少，比较器的负端外部引脚输入都被切断。
 3. 在允许内部参考电压工作 (CnVRE = 1) 之前设置参考电压。禁止在使能状态 (CnVRE = 1) 改变参考电压的设置。
 4. 确保在 CnEN = 0 (比较器操作停止) 时改变 CnRVM 寄存器的值。

<R>

备注 n = 0, 1

(5) 端口输入模式寄存器 8 (PIM8)

该寄存器用于允许或禁止端口 8 以 1 位为单元进行数字输入。

设置为禁止数字输入（用作模拟输入）以使用比较器或可编程增益放大器。因为禁止数字输入（用作模拟输入）为默认设置，所以设置为允许数字输入以使用端口功能或外部中断功能。

可通过 1 位或 8 位的存储操作指令设置 PIM8。

产生的复位信号将该寄存器清除为 00H。

图 8-6. 端口输入模式寄存器 8 (PIM8) 的格式

地址：F0048H 复位后：00H R/W

符号	7	6	5	4	3	2	1	0
PIM8	0	0	0	0	PIM83	PIM82	PIM81	PIM80

PIM8n	允许或禁止P8n引脚数字输入（n = 0 至 3）的选择
0	禁止数字输入（用作模拟输入）
1	允许数字输入

(6) 端口模式寄存器 8 (PM8)

该寄存器用于设置端口 8 以 1 位为单元进行输入或输出。

设置 PM80 至 PM83 位为 1，以便将 P80/CMP0P/INTP3/PGAI，P81/CMP0M，P82/CMP1P/INTP7 或 P83/CMP1M 引脚用作比较器或可编程增益放大器输入功能的正端或负端的输入功能。

此时 P80 至 P83 的输出锁存可能为 0 或者为 1。

可通过 1 位或 8 位的存储操作指令设置 PM80 至 PM83。

产生的复位信号将此寄存器设置为 FFH。

图 8-7. 端口模式寄存器 8 (PM8) 的格式

地址：FFF28H 复位后：FFH R/W

符号	7	6	5	4	3	2	1	0
PM8	1	1	1	1	PM83	PM82	PM81	PM80

PM8n	P8n 引脚 I/O 模式选择（n = 0 至 3）
0	输出模式（输出缓存开启）
1	输入模式（输出缓存关闭）

注意事项 当 CMP0P，CMP1 引脚选作比较器的正端输入，且在负端使用内部参考电压时，复用为 CMP0M，CMP1M 引脚的端口功能可在输入模式下使用。但禁止使用输出模式。此外禁止访问端口寄存器 8 (P8)。

8.4 比较器和可编程增益放大器的操作

8.4.1 启动比较器和可编程增益放大器的操作

关于启动比较器和可编程增益放大器的步骤，各使用方法描述如下：

- 仅使用比较器
 - 用外部引脚输入作为比较器参考电压（图 8-8）
 - 用内部参考电压作为比较器参考电压（图 8-9）
- 使用比较器和可编程增益放大器（用可编程增益放大器的输出电压作为比较器比较电压的输入）
 - 用外部引脚输入作为比较器参考电压（图 8-10）
 - 用内部参考电压作为比较器参考电压（图 8-11）
- 用可编程增益放大器输出电压作为 A/D 转换器模拟输入（图 8-12）

图 8-8. 用外部引脚输入作为比较器参考电压（仅使用比较器）

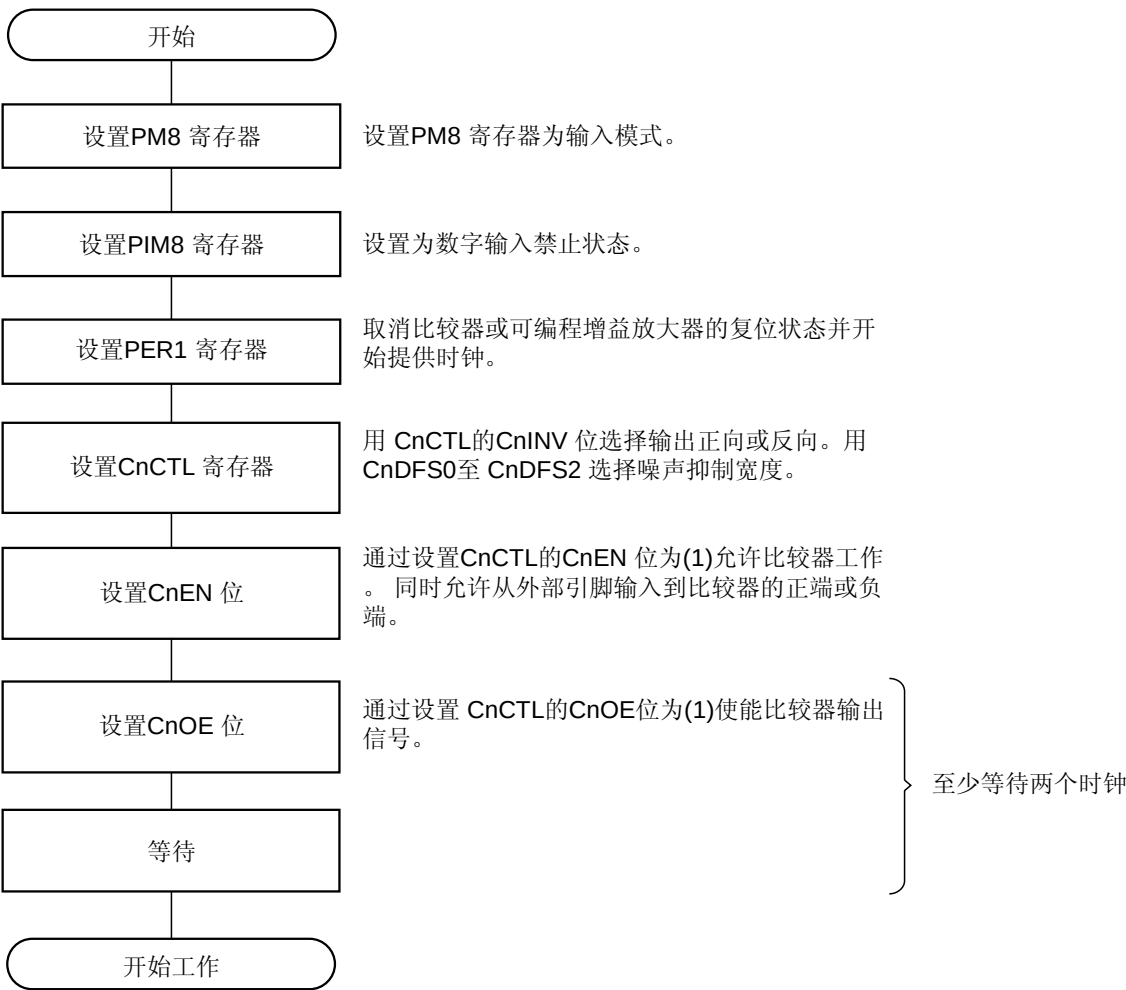


图 8-9. 用内部参考电压作为比较器参考电压
(仅使用比较器)

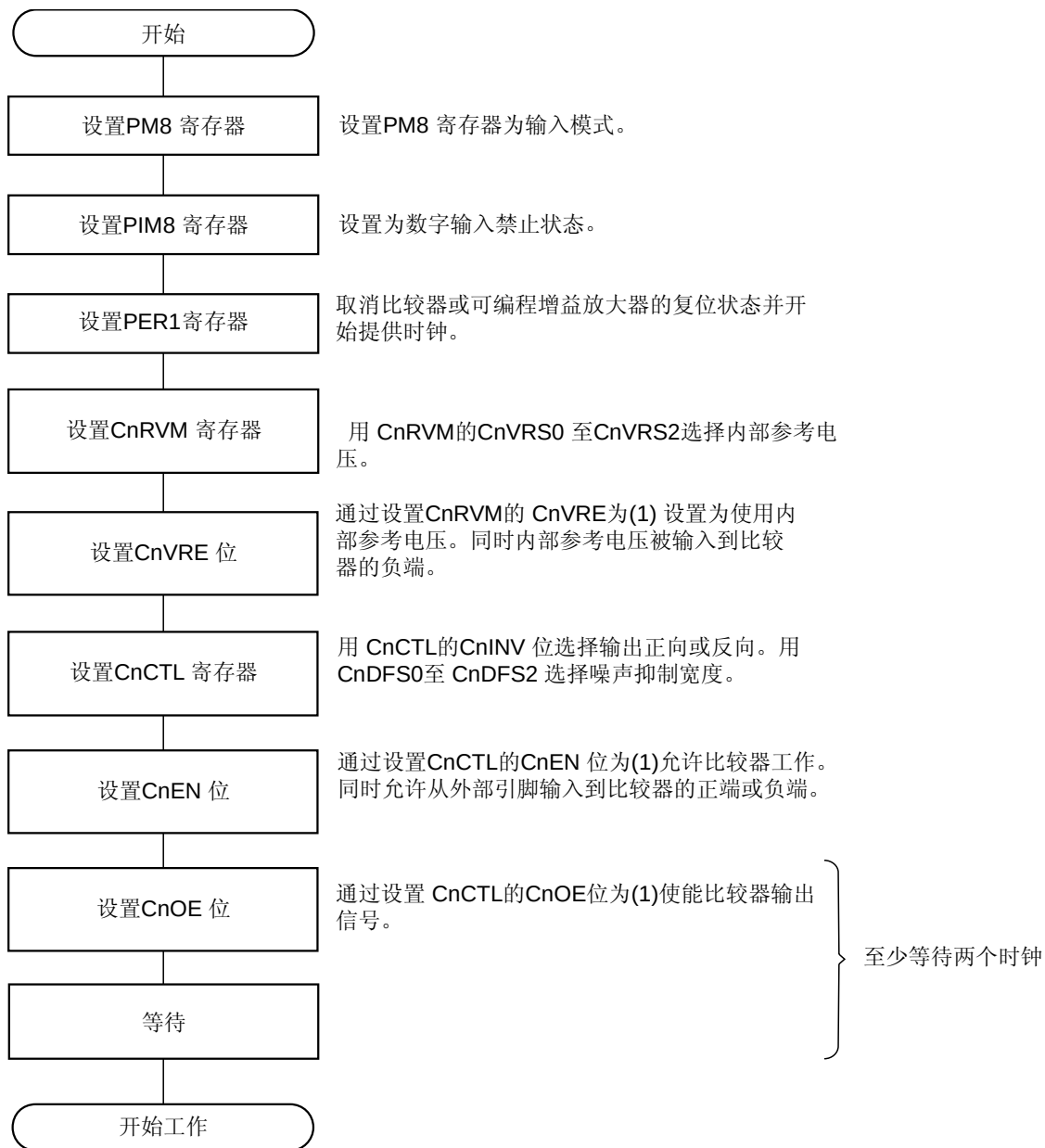
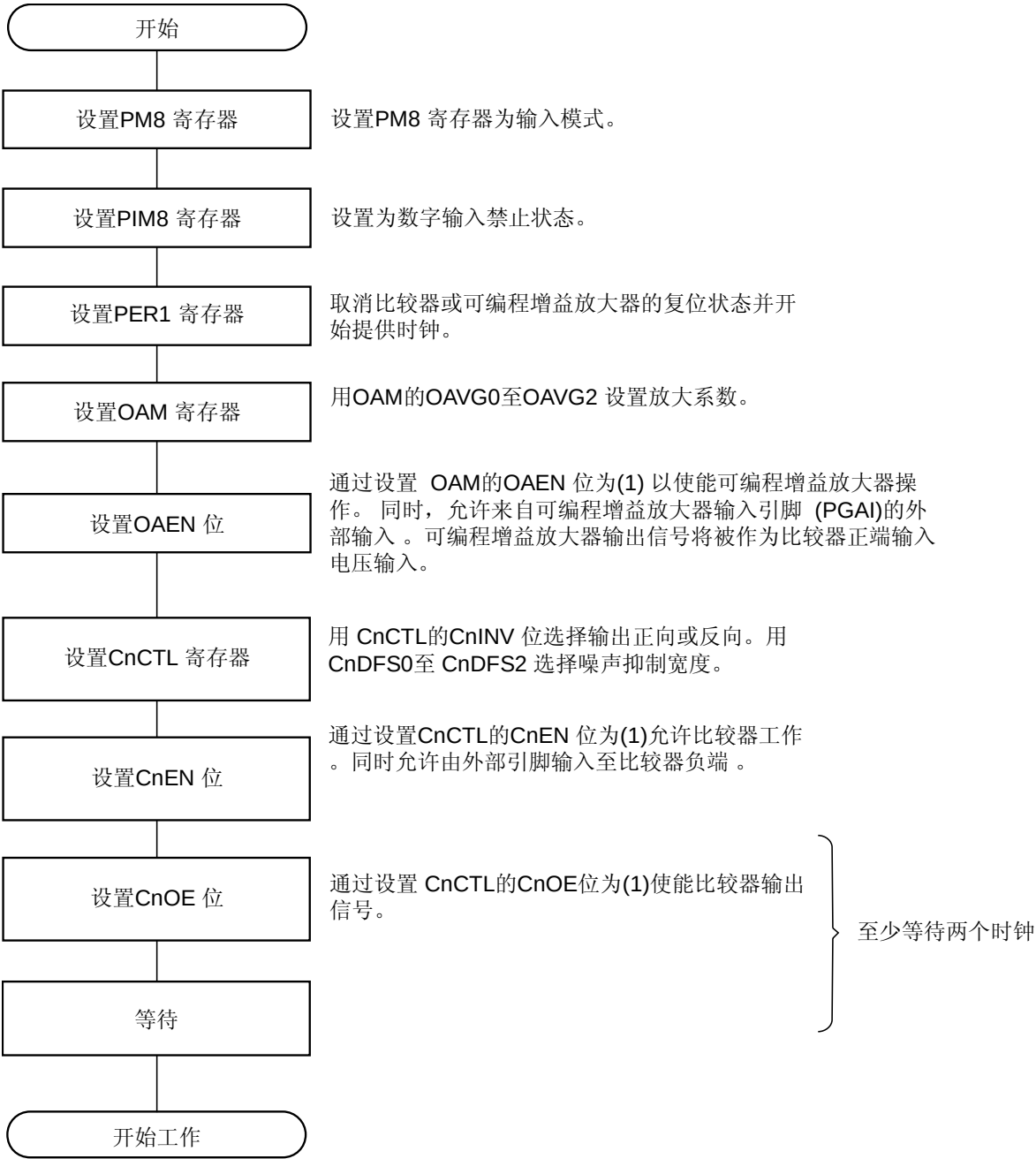


图 8-10. 用外部引脚输入作为比较器参考电压（使用比较器和可编程增益放大器）



备注 n = 0, 1

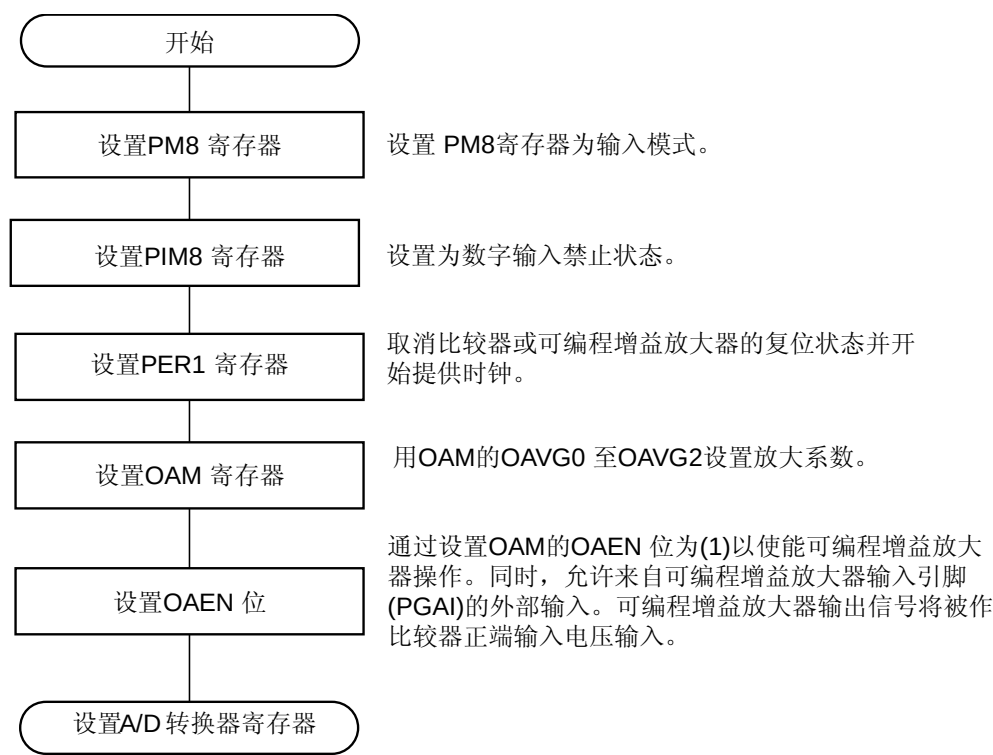
图 8-11. 用内部参考电压作为比较器参考电压（使用比较器和可编程增益放大器）



备注 n = 0, 1

用 A/D 转换器的模拟输入通道指定寄存器（ADS）选择可编程增益放大器输出信号为模拟输入之前，进行以下设置。

图 8-12. 用可编程增益放大器输出电压作为 A/D 转换器模拟输入



备注 n = 0, 1

8.4.2 停止比较器和可编程增益放大器的操作

关于停止比较器和可编程增益放大器的步骤，各使用方法描述如下：

- 仅使用比较器（图 8-13）
- 用可编程增益放大器输出电压作为比较器比较电压输入（图 8-14）
- 用可编程增益放大器输出电压作为 A/D 转换器模拟输入（图 8-15）

图 8-13. 仅使用比较器

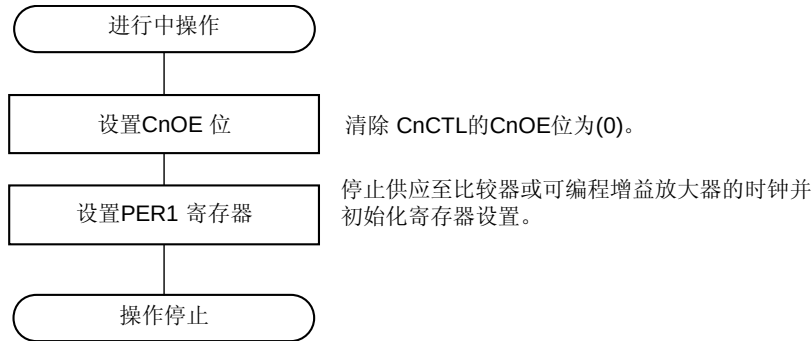


图 8-14. 用可编程增益放大器输出电压作为比较器比较电压输入

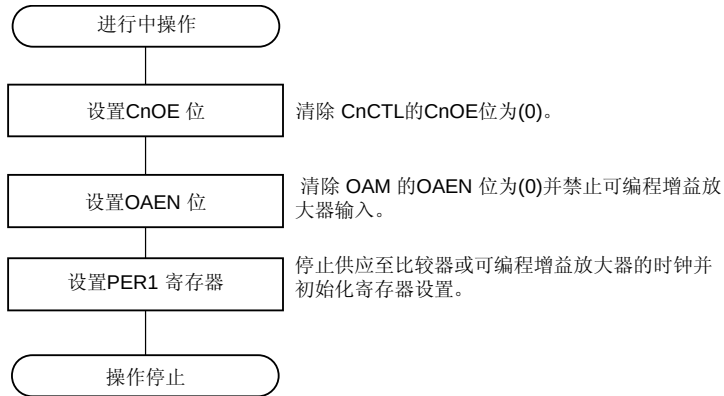
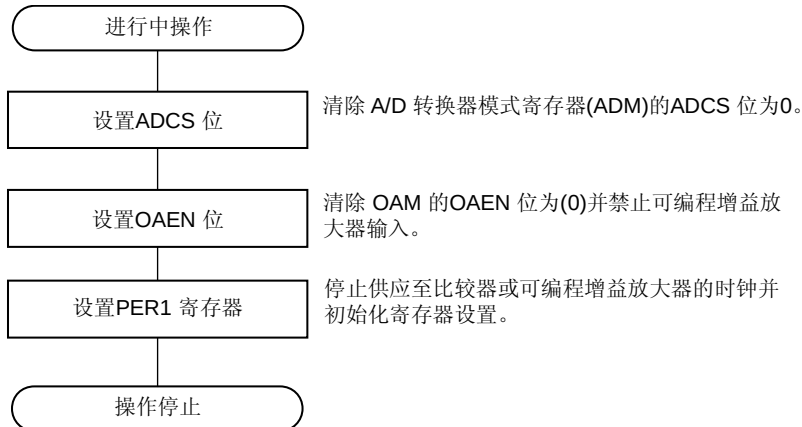


图 8-15. 用可编程增益放大器输出电压作为 A/D 转换器模拟输入



备注 n = 0, 1

第九章 时钟输出/蜂鸣器输出控制器

根据产品不同，时钟输出和蜂鸣器输出控制器的输出引脚数量不同。此外，78K0R/KC3-L 的 44-引脚产品未配备时钟输出和蜂鸣器输出控制器。

输出引脚	KC3-L（44-引脚）	KC3-L（48-引脚）	KD3-L	KE3-L
PCLBUZ0	—	√	√	√
PCLBUZ1	—	—	—	√

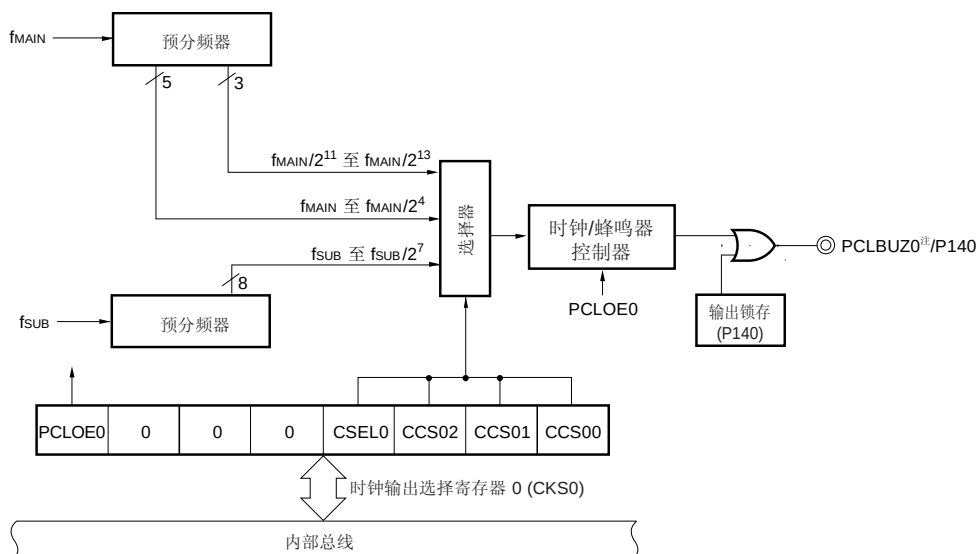
9.1时钟输出/蜂鸣器输出控制器的功能

在远程控制传输和时钟输出供应外围 IC 期间，时钟输出控制器供载波输出使用。
蜂鸣器输出的功能是输出蜂鸣器频率的方波。
其中一个引脚可以用于输出时钟或蜂鸣器声。
两个输出引脚（PCLBUZ0 和 PCLBUZ1）可用。
PCLBUZn 输出由时钟输出选择寄存器 n（CKSn）选择的时钟。
图 9-1 显示了时钟输出/蜂鸣器输出控制器的框图。

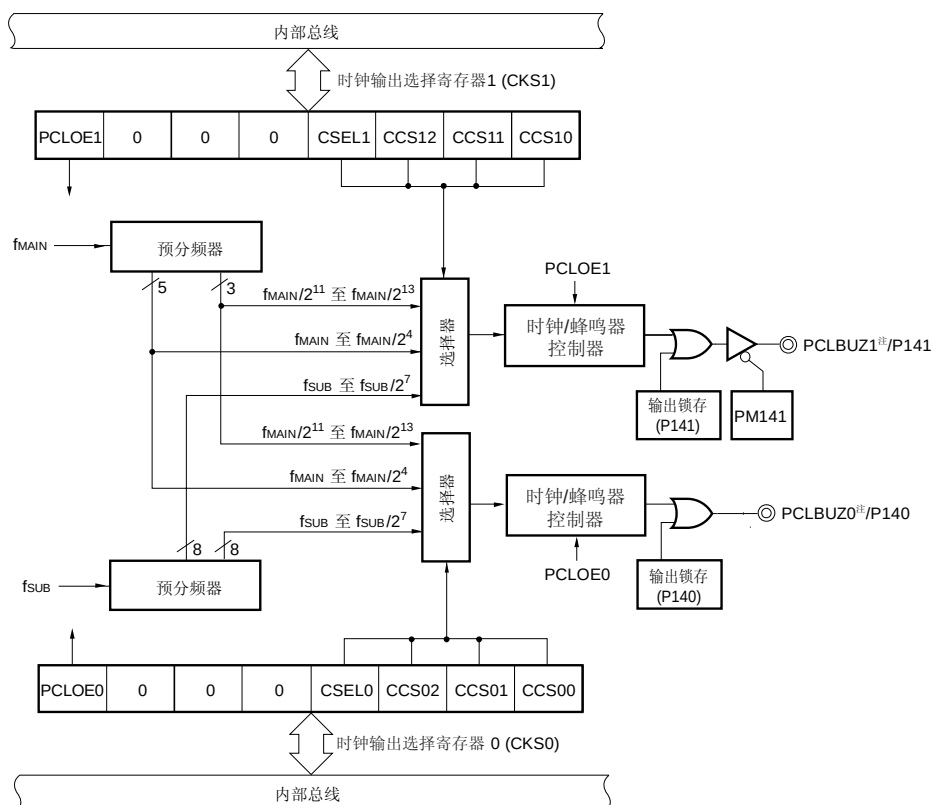
备注 n = 0: 78K0R/KC3-L（48-引脚），78K0R/KD3-L
 n = 0, 1: 78K0R/KE3-L

图 9-1. 时钟输出/蜂鸣器输出控制器的框图

- 78K0R/KC3-L (48-引脚), 78K0R/KD3-L



- 78K0R/KE3-L



注 PCLBUZ0 和 PCLBUZ1 引脚可以在 $2.7\text{ V} \leq V_{DD}$ 时输出一个高达 10 MHz 的时钟。在 $V_{DD} < 2.7\text{ V}$ 时禁止设置超出 5 MHz 的时钟。

9.2 时钟输出/蜂鸣器输出控制器的配置

时钟输出/蜂鸣器输出控制器包括以下硬件。

表 9-1. 时钟输出/蜂鸣器输出控制器的配置

项目	配置
控制寄存器	时钟输出选择寄存器 n（CKSn） 端口模式寄存器 14（PM14）（仅限于 78K0R/KE3-L） 端口寄存器 14（P14）

备注 n = 0: 78K0R/KC3-L（48-引脚），78K0R/KD3-L
 n = 0, 1: 78K0R/KE3-L

9.3 控制时钟输出/蜂鸣器输出控制器的寄存器

以下两种寄存器用于控制时钟输出/蜂鸣器输出控制器。

- 时钟输出选择寄存器 n（CKSn）
- 端口模式寄存器 14（PM14）（仅限于 78K0R/KE3-L）

（1）时钟输出选择寄存器 n（CKSn）

这些寄存器为时钟输出或为蜂鸣器频率输出引脚（PCLBUZn）设置输出允许/禁止，并设置输出时钟。
通过使用 CKSn 从 PCLBUZn 中选择时钟作为输出。
可通过 1 位或 8 位的存储操作指令设置 CKSn。
产生的复位信号将这些寄存器清除为 00H。

备注 n = 0: 78K0R/KC3-L（48-引脚），78K0R/KD3-L
 n = 0, 1: 78K0R/KE3-L

图 9-2. 时钟输出选择寄存器 n (CKSn) 的格式

地址: FFFA5H 复位后: 00H R/W

符号

<7> 6 5 4 3 2 1 0

CKSn

PCLOEn	0	0	0	CSELn	CCSn2	CCSn1	CCSn0

PCLOEn	PCLBUZn 输出使能/禁止说明
0	禁止输出 (默认)
1	允许输出

CSELn	CCSn2	CCSn1	CCSn0		PCLBUZn 输出时钟选择		
					f _{MAIN} = 5 MHz	f _{MAIN} = 10 MHz	f _{MAIN} = 20 MHz
0	0	0	0	f _{MAIN}	5 MHz	10 MHz [‡]	Setting prohibited [‡]
0	0	0	1	f _{MAIN} /2	2.5 MHz	5 MHz	10 MHz [‡]
0	0	1	0	f _{MAIN} /2 ²	1.25 MHz	2.5 MHz	5 MHz
0	0	1	1	f _{MAIN} /2 ³	625 kHz	1.25 MHz	2.5 MHz
0	1	0	0	f _{MAIN} /2 ⁴	312.5 kHz	625 kHz	1.25 MHz
0	1	0	1	f _{MAIN} /2 ¹¹	2.44 kHz	4.88 kHz	9.76 kHz
0	1	1	0	f _{MAIN} /2 ¹²	1.22 kHz	2.44 kHz	4.88 kHz
0	1	1	1	f _{MAIN} /2 ¹³	610 Hz	1.22 kHz	2.44 kHz
1	0	0	0	f _{SUB}	32.768 kHz		
1	0	0	1	f _{SUB} /2	16.384 kHz		
1	0	1	0	f _{SUB} /2 ²	8.192 kHz		
1	0	1	1	f _{SUB} /2 ³	4.096 kHz		
1	1	0	0	f _{SUB} /2 ⁴	2.048 kHz		
1	1	0	1	f _{SUB} /2 ⁵	1.024 kHz		
1	1	1	0	f _{SUB} /2 ⁶	512 Hz		
1	1	1	1	f _{SUB} /2 ⁷	256 Hz		

注 在 10 MHz 范围内使用输出时钟。此外，若要在 V_{DD} < 2.7 V 使用输出时钟，则需在 5 MHz 范围内使用。

注意事项 1. 在禁止时钟输出 (PCLOEn = 0) 之后改变输出时钟。

<R> 2. 当选用主系统时钟 (CSELn = 0) 时，若要转换为 STOP 模式，则需在执行 STOP 指令之前设置 PCLOEn = 0。当选用副系统时钟 (CSELn = 1) 时，因为时钟可在 STOP 模式下输出，所以可设置 PCLOEn = 1。

备注 1. n = 0: 78K0R/KC3-L (48-引脚), 78K0R/KD3-L
n = 0, 1: 78K0R/KE3-L
2. f_{MAIN}: 主系统时钟频率
3. f_{SUB}: 副系统时钟频率

(2) 端口模式寄存器 14 (PM14) (仅限于 78K0R/KE3-L)

该寄存器以 1 位为单元设置端口 14 的 P141 输入/输出。

当用 P140/PCLBUZ0 和 P141/PCLBUZ1 引脚作为时钟输出/蜂鸣器输出时，需将 PM141 以及 P140 和 P141 的输出锁存清除为 0。

通过 1 位或 8 位的存储操作指令设置 PM14。

产生的复位信号将此寄存器设置为 FFH。

图 9-3. 端口模式寄存器 14 (PM14) 的格式

[illegible]

9.4 时钟输出/蜂鸣器输出控制器的操作

其中一个引脚可以用于输出时钟或蜂鸣器声。

PCLBUZ0 输出由时钟输出选择寄存器 0 (CKS0) 选择的时钟/蜂鸣器。

PCLBUZ1 输出由时钟输出选择寄存器 1 (CKS1) 选择的时钟/蜂鸣器。

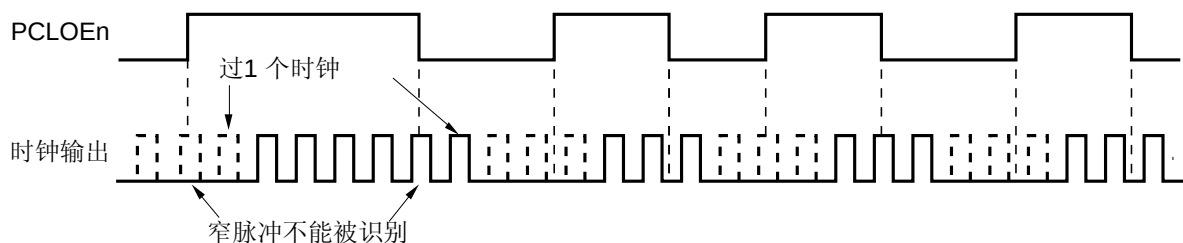
9.4.1 作为输出引脚的操作

PCLBUZn 输出步骤如下:

- <1> 用时钟输出选择寄存器 (CKSn) 的位 0 至 3 (CCSn0 至 CCSn2, CSELn) 选择 PCLBUZn 引脚 (在禁止状态下输出) 的输出频率。
- <2> 设置 CKSn 的为 7 (PCLOEn) 为 1 以使能时钟/蜂鸣器输出。

<R>	备注	1. 在转换允许或禁止时钟输出（PCLOEn）后，用作输出时钟的控制器需开始或停止 1 个时钟长的时间再输出时钟。此时，不输出宽度窄的脉冲。图 9-4 显示了用 PCLOEn 使能或停止输出以及输出时钟的时序。 2. n = 0: 78K0R/KC3-L（48-引脚），78K0R/KD3-L n = 0, 1: 78K0R/KE3-L

图 9-4. 远程控制输出应用举例



第十章 看门狗定时器

10.1 看门狗定时器的功能

看门狗定时器在内部低速振荡时钟下工作。

看门狗定时器用于检测无意中形成的程序循环。如果检测到程序循环，则产生内部复位信号。

可在以下情况中检测程序循环。

- 如果看门狗定时器计数器溢出
- 如果在看门狗定时器使能寄存器（WDTE）上执行 1 位处理指令
- 如果将除“ACH”之外的数据写入 WDTE
- 如果在窗口关闭期间将数据写入 WDTE

当因看门狗定时器而发生复位时，将复位控制标志寄存器（RESF）的位 4（WDRF）置为 1。有关 RESF 的细节，参见**第十九章 复位功能**。

当达到溢出时间的 75%时，可以产生间隔中断。

10.2 看门狗定时器的配置

看门狗定时器包含以下硬件。

表 10-1. 看门狗定时器的配置

项目	配置
控制寄存器	看门狗定时器使能寄存器(WDTE)

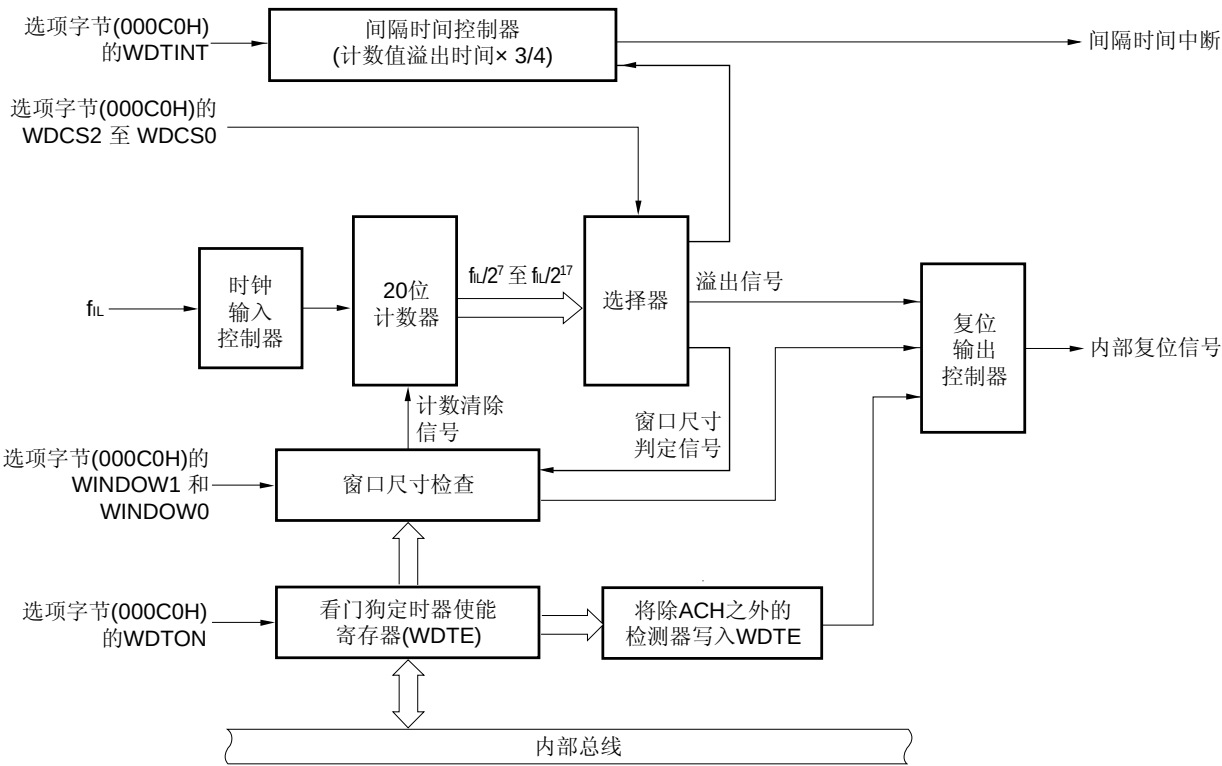
如何控制计数器操作、上溢出时间、窗口开启周期和间隔中断都可以由选项字节进行设置。

表 10-2. 选项字节和看门狗定时器的设置

看门狗定时器的设置	选项字节 (000C0H)
看门狗定时器间隔中断	位 7 (WDTINT)
窗口开启周期	位6 和 5 (WINDOW1、WINDOW0)
看门狗定时器的控制计数器操作	位4 (WDTON)
看门狗定时器的上溢出时间	位3 至 1 (WDCS2 至 WDCS0)
看门狗定时器的控制计数器操作 (在HALT/STOP模式中)	位0 (WDSTBYON)

备注 有关选项字节的细节，参见第二十三章 选项字节。

图 10-1. 看门狗定时器的框图



10.3 控制看门狗定时器的寄存器

看门狗定时器由看门狗定时器使能寄存器（WDTE）控制。

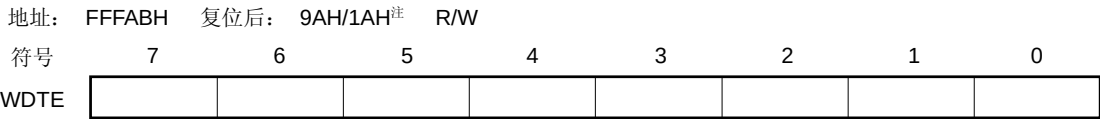
(1) 看门狗定时器使能寄存器（WDTE）

将“ACH”写入 WDTE 以清除看门狗定时器计数器，并重新开始计数。

该寄存器可以由 8 位存储器处理指令设置。

产生的复位信号将该寄存器置为 9AH或 1AH^注。

图 10-2. 看门狗定时器使能寄存器的格式（WDTE）



注 WDTE 复位值依据选项字节（000C0H）的 WDTON 设置值不同而不同。要运行看门狗定时器，需将 WDTON 置为 1。

WDTON 设置值	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. 如果将除“ACH”之外的值写入 WDTE，则产生内部复位信号。
 - 2. 如果执行 WDTE 的 1 位存储器处理指令，则产生内部复位信号。
 - 3. 从 WDTE 中读取的值为 9AH/1AH（这不同于写入值（ACH））。

10.4 看门狗定时器的操作

10.4.1 看门狗定时器的控制操作

1. 当使用看门狗定时器时，它的操作由选项字节(000C0H)指定。
 - 通过将选项字节（000C0H）的位 4（WDTON）置为 1，可以启动看门狗定时器的计数操作（复位解除后计数器开始工作）（有关细节，参见第二十三章）。

WDTON	看门狗定时器计数器
0	禁止计数器操作(复位后停止计数)
1	允许计数器操作(复位后开始计数)

- 使用选项字节（000C0H）的位 3 至位 1（WDCS2 至 WDSC0）设置上溢出时间（有关细节，参见 10.4.2 和第二十三章）。
 - 使用选项字节（000C0H）的位 6 和位 5（WINDOW1 和 WINDOW0）设置窗口开启周期（有关细节，参见 10.4.3 和第二十三章）。
2. 复位解除后，看门狗定时器开始计数。
 3. 在看门狗定时器开始计数之后和选项字节设置溢出时间之前，通过将“ACH”写入 WDTE，看门狗定时器被清零并重新开始计数。
 4. 在这之后，或者在窗口开启周期中复位解除之后，再次写 WDTE。如果在窗口开启周期内写 WDTE，则产生内部复位信号。
 5. 如果在未将“ACH”写入 WDTE 时溢出时间期满，则产生内部复位信号。
在下列情况中产生内部复位信号。
 - 如果在看门狗定时器使能寄存器（WDTE）上执行 1 位处理指令
 - 如果将除“ACH”之外的数据写入 WDTE

- 注意事项**
1. 复位解除后首次将数据写入 WDTE 时，只要在溢出时间之前写入寄存器，则不管窗口开启时间如何，看门狗定时器可在任何时序被清零，且看门狗定时器重新开始计数。
 2. 如果通过将“ACH”写入 WDTE 来清除看门狗定时器，则实际的溢出时间可能与由选项字节设置的溢出时间有所不同，最多相差 $2/f_{IL}$ 秒。
 3. 可以在计数值溢出之前清除看门狗定时器。

注意事项 4. 根据选项字节（000C0H）位 0（WDSTBYON）的设置值，看门狗定时器在 HALT 和 STOP 模式下的操作有如下不同。

	WDSTBYON = 0	WDSTBYON = 1
在HALT模式中	看门狗定时器操作停止。	看门狗定时器操作继续。
在STOP模式中		

如果 WDSTBYON = 0，则在 HALT 或 STOP 模式解除后，看门狗定时器恢复计数。此时，将计数器清除为 0 并开始计数。

当在解除 STOP 模式后使用 X1 振荡时钟工作时，在振荡稳定时间逝去之后 CPU 开始工作。

因此，如果 STOP 模式解除和看门狗定时器溢出之间的时间很短，则在振荡稳定时间期间会发生溢出，从而引起复位。

所以，当使用 X1 振荡时钟工作且在 STOP 模式解除后由间隔中断清除看门狗定时器时，要设置考虑到振荡稳定时间的溢出时间。

5. 在flash存储器自编程和EEPROM[™]仿真期间，看门狗定时器继续工作。处理期间，中断响应时间会延迟。设置溢出时间和窗口尺寸，要将延迟考虑进去。

10.4.2 看门狗定时器的溢出时间设置

使用选项字节（000C0H）的位 3 至位 1（WDCS2 至 WDCS0）设置看门狗定时器的溢出时间。

如果发生溢出，则产生一个内部复位信号。溢出时间之前的窗口开启周期内，通过将“ACH”写入 WDTE，可以清零当前计数，看门狗定时器重新开始计数。

可以设置以下溢出时间。

表 10-3. 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 (f _{IL} = 33 kHz (MAX.))
0	0	0	2 ⁷ /f _{IL} (3.88 ms)
0	0	1	2 ⁸ /f _{IL} (7.76 ms)
0	1	0	2 ⁹ /f _{IL} (15.52 ms)
0	1	1	2 ¹⁰ /f _{IL} (31.03 ms)
1	0	0	2 ¹² /f _{IL} (124.12 ms)
1	0	1	2 ¹⁴ /f _{IL} (496.48 ms)
1	1	0	2 ¹⁵ /f _{IL} (992.97 ms)
1	1	1	2 ¹⁷ /f _{IL} (3971.88 ms)

注意事项 在 flash 存储器自编程和 EEPROM 仿真期间，看门狗定时器继续工作。处理期间，中断响应时间会延迟。设置溢出时间和窗口尺寸，要将延迟考虑进去。

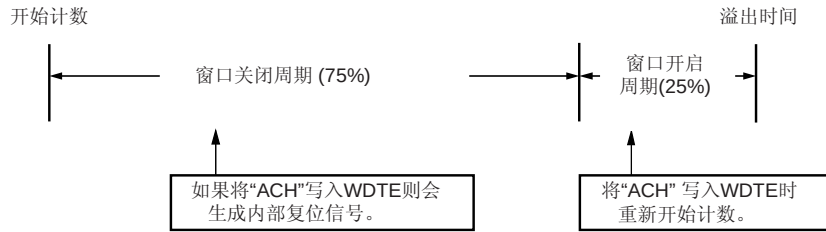
备注 f_{IL}: 内部低速振荡时钟频率

10.4.3 看门狗定时器窗口开启周期的设置

使用选项字节（000C0H）的位 6 和位 5（WINDOW1、WINDOW0）设置看门狗定时器的窗口开启周期。窗口概况如下所示。

- 如果在窗口开启周期内将“ACH”写入 WDTE，则清除看门狗定时器并重新开始计数。
- 即使在窗口关闭期间将“ACH”写入 WDTE，也会检测到异常并产生内部复位信号。

示例：若窗口开启周期为 25%



注意事项 复位解除后首次将数据写入 WDTE 时，只要在溢出时间之前写入寄存器，则不管窗口开启时间如何，看门狗定时器可在任何时序被清零，且看门狗定时器重新开始计数。

可以如下设置窗口开启周期。

表 10-4. 看门狗定时器窗口开启周期的设置

WINDOW1	WINDOW0	看门狗定时器窗口开启周期
0	0	25%
0	1	50%
1	0	75%
1	1	100%

- 注意事项**
1. 在 flash 存储器自编程和 EEPROM 仿真期间，看门狗定时器继续工作。处理期间，中断响应时间会延迟。设置溢出时间和窗口尺寸，要将延迟考虑进去。
 2. 当选项字节（000C0H）的位 0（WDSTBYON）= 0 时，无论 WINDOW1 和 WINDOW0 为何值，窗口开启周期都为 100%。
 3. 如果看门狗定时器符合下列条件的任意一个，就不能将窗口开启周期设置为 25%。
 - 使用 STOP 模式或软件停止所有主系统时钟（内部高速振荡时钟、X1 时钟和外部主系统时钟）时。

备注 如果将溢出时间置为 $2^{10}/f_{IL}$ ，则窗口关闭时间和开启时间如下。

	窗口开启周期的设置			
	25%	50%	75%	100%
窗口关闭时间	0 至 28.44 ms	0 至 18.96 ms	0 至 9.48 ms	无
窗口开启时间	28.44 至 31.03 ms	18.96 至 31.03 ms	9.48 至 31.03 ms	0 至 31.03 ms

- <窗口开启周期为 25%时>
- 溢出时间：
 $2^{10}/f_{IL} \text{ (MAX.)} = 2^{10}/33 \text{ kHz (MAX.)} = 31.03 \text{ ms}$
 - 窗口关闭时间：
 $0 \text{ 至 } 2^{10}/f_{IL} \text{ (MIN.)} \times (1 - 0.25) = 0 \text{ 至 } 2^{10}/27 \text{ kHz (MIN.)} \times 0.75 = 0 \text{ 至 } 28.44 \text{ ms}$
 - 窗口开启时间：
 $2^{10}/f_{IL} \text{ (MIN.)} \times (1 - 0.25) \text{ 至 } 2^{10}/f_{IL} \text{ (MAX.)} = 2^{10}/27 \text{ kHz (MIN.)} \times 0.75 \text{ 至 } 2^{10}/33 \text{ kHz (MAX.)}$
 $= 28.44 \text{ 至 } 31.03 \text{ ms}$

10.4.4 看门狗定时器间隔中断的设置

依据选项字节（000C0H）位 7（WDTINT）的设置，达到溢出时间的 75%时可以产生间隔中断（INTWDTI）。

表 10-5. 看门狗定时器间隔中断的设置

WDTINT	看门狗定时器间隔中断的使用
0	使用间隔中断。
1	达到溢出时间的 75%时产生间隔中断。

注意事项 当在解除 STOP 模式后使用 X1 振荡时钟工作时，在振荡稳定时间逝去之后 CPU 开始工作。因此，如果 STOP 模式解除和看门狗定时器溢出之间的时间很短，则在振荡稳定时间期间会发生溢出，从而引起复位。所以，当使用 X1 振荡时钟工作且在 STOP 模式解除后由间隔中断清除看门狗定时器时，要设置考虑到振荡稳定时间的溢出时间。

备注 即使在产生 INTWDTI 之后，看门狗定时器仍继续计数（直到 ACH 写入 WDTE 寄存器）。如果在溢出时间之前未将 ACH 写入 WDTE 寄存器，则会产生内部复位信号。

第十一章 A/D 转换器

A/D 转化器模拟输入通道的数量依据产品不同而不同。

	78K0R/KC3-L (44-引脚)	78K0R/KC3-L (48-引脚)	78K0R/KD3-L	78K0R/KE3-L
模拟输入通道	10 ch (ANI0 至 ANI9)	11 ch (ANI0 至 ANI10)	11 ch (ANI0 至 ANI10)	12 ch (ANI0 至 ANI11)

11.1 A/D 转换器的功能

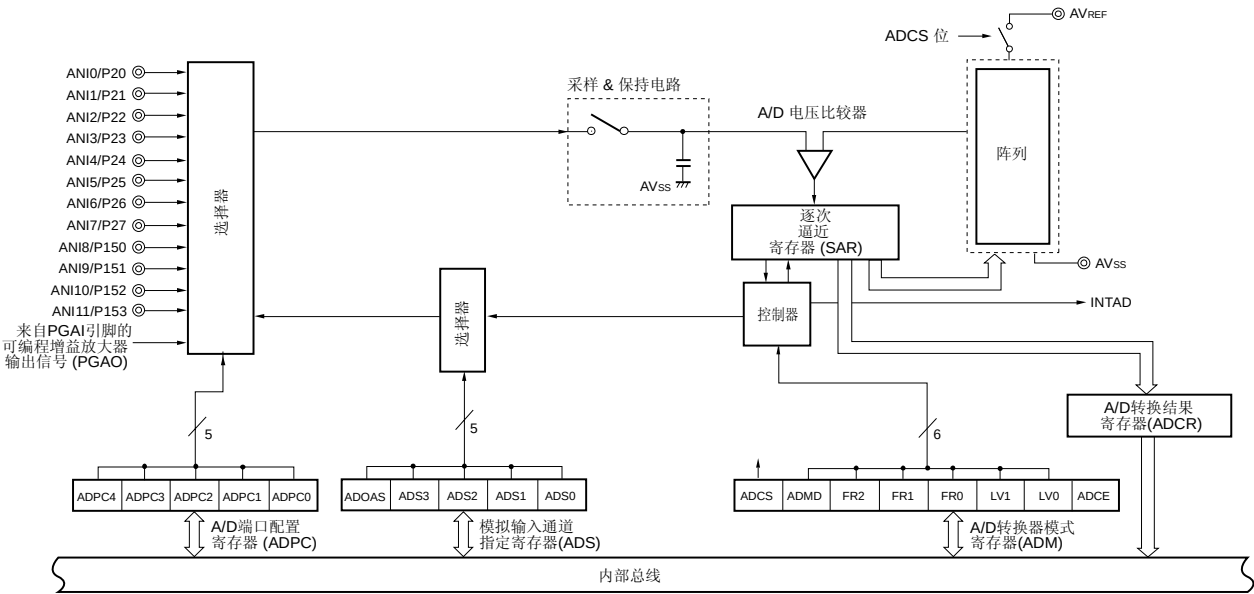
A/D 转换器是一个 10 位分辨率的转换器，将模拟输入信号转换成数字值，用来控制模拟输入三十个通道的总数，包括高达十二个的 A/D 转换器模拟输入（ANI0 至 ANI11）和可编程增益放大器输出（PGA0）的通道。

A/D 转换器具有以下功能。

• 10 位分辨率的 A/D 转换

10 位分辨率的 A/D 转换由从 ANI0 至 ANI11 中选择一个模拟输入通道重复执行。每次 A/D 转换结束后，都会产生一个中断请求信号（INTAD）。

图 11-1. A/D 转换器的框图



备注 ANI0 至 ANI9: 78K0R/KC3-L (44-引脚)
ANI0 至 ANI10: 78K0R/KC3-L (48-引脚) 和 78K0R/KD3-L
ANI0 至 ANI11: 78K0R/KE3-L

11.2 A/D 转换器的配置

A/D 转换器包括以下硬件。

(1) ANI0 至 ANI11 引脚

这些引脚为十二通道 A/D 转换器的模拟输入引脚。它们输入的是待转换成数字信号的模拟信号。除了选作为模拟输入引脚的一个外，其它引脚都能用作 I/O 端口引脚。

备注	ANI0 至 ANI9:	78K0R/KC3-L (44-引脚)
	ANI0 至 ANI10:	78K0R/KC3-L (48-引脚) 和 78K0R/KD3-L
	ANI0 至 ANI11:	78K0R/KE3-L

(2) PGAO

这是 PGAI 引脚的可编程增益放大器输出信号。通过将可编程增益放大器的输出信号选作为模拟输入，A/D 转换器可以执行 A/D 转换。

(3) 采样&保持电路

采样保持电路采集每个从输入电路连续发出的模拟输入电压，并发送采样数据到 A/D 电压比较器。A/D 转换期间，该电路保持所采样模拟输入的电压。

(4) A/D 电压比较器

该 A/D 电压比较器将从数组的电压档位产生的电压与模拟输入电压进行比较。如果发现比较的结果是模拟输入电压大于参考电压（ $1/2 AV_{REF}$ ），则将会设置逐次逼近寄存器（SAR）的最高有效位（MSB）。如果模拟输入电压小于参考电压（ $1/2 AV_{REF}$ ），那么逐次逼近寄存器（SAR）的最高有效位（MSB）复位。接着，自动设置 SAR 寄存器的位 10，并启动下一次比较。由位 11 的值来选择数组的电压档位，其结果已经设定。

位 11 = 0:	$(1/4 AV_{REF})$
位 11 = 1:	$(3/4 AV_{REF})$

将数组的档位电压和模拟输入电压进行比较，而 SAR 寄存器的位 10 则根据比较结果来执行操作。

模拟输入电压 $\geq$ 数组档位电压:	位 10 = 1
模拟输入电压 $\leq$ 数组档位电压:	位 10 = 0

这样的比较会持续至 SAR 寄存器的位 0。

(5) 数组

数组产生从模拟输入引脚输入的比较电压。

(6) 逐次逼近寄存器 (SAR)

SAR 寄存器为 12 位寄存器，用于设置档位电压数据，自数组的档位电压值与模拟输入引脚的电压值作比较，从最高有效位 (MSB) 开始，每次移 1 位。
如果数据在 SAR 中始终设置成最低有效位 (LSB) (A/D 转换结束)，那么 SAR 寄存器 (转换结果) 的内容将会被保存到 A/D 转换结果寄存器 (ADCR) 中。当所有指定的 A/D 转换操作都结束时，将产生 A/D 转换结束中断请求信号 (INTAD)。

(7) 10 位 A/D 转换结果寄存器 (ADCR)

每次 A/D 转换完成，A/D 转换结果将从逐次逼近寄存器加载到该寄存器，ADCR 寄存器将 A/D 转换结果保存在其高 10 位 (低 6 位固定为 0)。

(8) 8 位 A/D 转换结果寄存器 (ADCRH)

每次 A/D 转换完成，A/D 转换结果将从逐次逼近寄存器加载到该寄存器，且 ADCRH 寄存器存储 A/D 转换结果的高 8 位。

(9) 控制器

该电路控制待转换为数字信号的输入模拟信号的转换时间，还控制转换操作的开始和停止。当 A/D 转换已经完成时，该控制器产生 INTAD。

(10) AV_{REF} 引脚

该引脚输入 A/D 转换器、可编程增益放大器、比较器的供电电源引脚和 A/D 转换、以及比较器的参考电压。当端口 2、15 和 8 的所有引脚都用作模拟端口引脚时，确保 AV_{REF} 的电位在 $1.8\text{ V} \leq \text{AV}_{\text{REF}} \leq \text{V}_{\text{DD}}$ 内。当端口 2、15 和 8 的一个或多个引脚用作数字端口引脚时，确保 AV_{REF} 与 V_{DD} 电位相同。
根据其跨越 AV_{REF} 和 AV_{SS} 的供电电压，输入到 ANI0 至 ANI11 的模拟信号被转化为数字信号。

(11) AV_{SS} 引脚

该引脚为 A/D 转换器的接地电位引脚。即使不使用 A/D 转换器，也要保证该引脚的电位总是与 V_{SS} 引脚的电位相同。

备注	ANI0 至 ANI9:	78K0R/KC3-L (44-引脚)
	ANI0 至 ANI10:	78K0R/KC3-L (48-引脚) 和 78K0R/KD3-L
	ANI0 至 ANI11:	78K0R/KE3-L

11.3 A/D 转换器中使用的寄存器

A/D 转换器使用以下七种寄存器。

- 外围使能寄存器 0 (PER0)
- A/D 转换器模式寄存器 (ADM)
- 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、15、8 (PM2、PM15、PM8)
- 10 位 A/D 转换结果寄存器 (ADCR)
- 8 位 A/D 转换结果寄存器 (ADCRH)

(1) 外围使能寄存器 0 (PER0)

PER0 用于启动或禁止各外围硬件宏功能的使用。供应未使用的硬件宏功能的时钟会停止工作，以便于减少功耗和噪声。

当使用 A/D 转换器时，确保将该寄存器的位 5 (ADCEN) 置为 1。

可以由 1 位或 8 位存储器处理指令设置 PER0。

产生的复位信号将该寄存器清除为 00H。

图 11-2. 外围使能寄存器 0 (PER0) 的格式

地址: F00F0H 复位后: 00H R/W							
符号	<7>	6	<5>	<4>	3	<2>	1 0
PER0	RTCEN	0	ADCEN	IICAEN	0	SAU0EN	0 0

ADCEN	A/D转换器输入时钟供应的控制
0	停止输入时钟供应。 • A/D转换器使用的SFR不能写入。 • A/D转换器处于复位状态。
1	供应输入时钟。 • A/D转换器使用的SFR不能读/写。

- 注意事项
1. 设置 A/D 转换器时，确保首先将 ADCEN 置为 1。如果 ADCEN = 0，则忽略对 A/D 转换器的控制寄存器的写入，即使读取寄存器，也只能读取默认值。
 2. 确保将 PER0 寄存器的位 0、1、3 和 6 (78K0R/KC3-L 的 44-引脚产品的位 0、1、3、4 和 6) 清除为 0。

(2) A/D 转换器模式寄存器 (ADM)

该寄存器设置待进行 A/D 转换的模拟输入的转换时间，并控制转换的开始/停止。

可以由 1 位或 8 位存储器处理指令设置 ADM。

产生的复位信号将该寄存器清除为 00H。

图 11-3. A/D 转换器模式寄存器 (ADM) 的格式

地址: FFF30H 复位后: 00H R/W

符号 <7> 6 5 4 3 2 1 <0>

ADM	ADCS	ADMD	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

ADMD	A/D 转换操作模式指定
0	选择模式
1	扫描模式

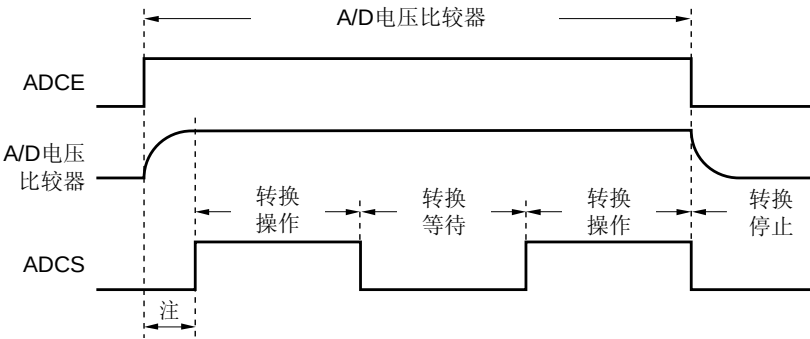
ADCE	A/D 电压比较操作控制 ^{注 2}
0	停止 A/D 电压比较器操作
1	允许 A/D 电压比较器操作 (A/D 电压比较器)

- 注
1. 有关 FR2 至 FR0、LV1、LV0 和 A/D 转换的细节，参见表 11-2 A/D 转换时间选择。
 2. A/D 电压比较器的操作由 ADCS 和 ADCE 控制，且从操作开始到操作稳定需要 1 μ S。因此，从 ADCE 设置为 1 的时刻开始 1 μ S 或更长时间之后，将 ADCS 置为 1 时，此时的转换结果优先与第一次的转换结果。否则，忽略第一次转换的数据。

表 11-1. ADCS 和 ADCE 的设置

ADCS	ADCE	A/D 转换操作
0	0	停止状态(不存在直流功耗路径)
0	1	转换等待模式(A/D 电压比较器: 仅比较器消耗功率)
1	0	禁止设置
1	1	转换模式(A/D 电压比较器)

图 11-4. 使用 A/D 电压比较器时的时序图



注 要使内部电路稳定，从 ADCE 位上升到 ADCS 位下降的时间必须为 1 μ s 或者更长。

注意事项 在将位 FR0 至 FR2、LV1 和 LV0 重写为同一数据之外的数值之前，必须停止 A/D 转换。

表 11-2. A/D 转换时间选择(1/3)

(1) $4.0\text{ V} \leq \text{AV}_{\text{REF}} \leq 5.5\text{ V}$

A/D转换器模式寄存器(ADM)					模式	转换时间选择				转换时钟 (f _{AD})
FR2	FR1	FR0	LV1	LV0		f _{CLK} = 2 MHz	f _{CLK} = 5 MHz	f _{CLK} = 10 MHz	f _{CLK} = 20 MHz	
0	0	0	0	0	标准	禁止设置	禁止设置	34.2 μs	17.1 μs	f _{CLK} /20
0	0	1					34.4 μs	17.2 μs	8.6 μs	f _{CLK} /10
0	1	0					27.6 μs	13.8 μs	6.9 μs	f _{CLK} /8
0	1	1				52.0 μs	20.8 μs	10.4 μs	5.2 μs	f _{CLK} /6
1	0	0				35.0 μs	14.0 μs	7.0 μs	禁止设置	f _{CLK} /4
1	0	1				26.5 μs	10.6 μs	5.3 μs		f _{CLK} /3
1	1	0				18.0 μs	7.2 μs	禁止设置		f _{CLK} /2
1	1	1				9.5 μs	禁止设置		f _{CLK}	
×	×	×	0	1	增强电压	禁止设置				—
0	0	0	1	0	高速1	禁止设置	64.4 μs	32.2 μs	16.1 μs	f _{CLK} /20
0	0	1					32.4 μs	16.2 μs	8.1 μs	f _{CLK} /10
0	1	0					65.0 μs	26.0 μs	13.0 μs	6.5 μs
0	1	1				49.0 μs	19.6 μs	9.8 μs	4.9 μs	f _{CLK} /6
1	0	0				33.0 μs	13.2 μs	6.6 μs	3.3 μs	f _{CLK} /4
1	0	1				25.0 μs	10.0 μs	5.0 μs	2.5 μs	f _{CLK} /3
1	1	0				17.0 μs	6.8 μs	3.4 μs	禁止设置	f _{CLK} /2
1	1	1				9.0 μs	3.6 μs	禁止设置		f _{CLK}
0	0	0	1	1	高速2	禁止设置	禁止设置	34.2 μs	17.1 μs	f _{CLK} /20
0	0	1					34.4 μs	17.2 μs	8.6 μs	f _{CLK} /10
0	1	0					27.6 μs	13.8 μs	6.9 μs	f _{CLK} /8
0	1	1				52.0 μs	20.8 μs	10.4 μs	5.2 μs	f _{CLK} /6
1	0	0				35.0 μs	14.0 μs	7.0 μs	3.5 μs	f _{CLK} /4
1	0	1				26.5 μs	10.6 μs	5.3 μs	禁止设置	f _{CLK} /3
1	1	0				18.0 μs	7.2 μs	3.6 μs		f _{CLK} /2
1	1	1				9.5 μs	3.8 μs	禁止设置		f _{CLK}

注意事项 1. 将 FR2 至 FR0、LV1 和 LV0 重写为除相同数据之外的值时，预先停止一次 A/D 转换（ADCS = 0）。
 2. 以上的转换时间不包括时钟频率误差。选择转换时间时要考虑到时钟频率误差。

备注 f_{CLK}: CPU/外围硬件时钟频率

表 11-2. A/D 转换时间选择(2/3)

(2) $2.7\text{ V} \leq \text{AV}_{\text{REF}} \leq 5.5\text{ V}$

A/D转换器模式寄存器(ADM)					模式	转换时间选择				转换时钟 (f _{AD})
FR2	FR1	FR0	LV1	LV0		f _{CLK} = 2 MHz	f _{CLK} = 5 MHz	f _{CLK} = 10 MHz	f _{CLK} = 20 MHz	
0	0	0	0	0	标准	禁止设置	禁止设置	34.2 μs	17.1 μs	f _{CLK} /20
0	0	1					34.4 μs	17.2 μs	8.6 μs	f _{CLK} /10
0	1	0					27.6 μs	13.8 μs	禁止设置	f _{CLK} /8
0	1	1					52.0 μs	20.8 μs		10.4 μs
1	0	0				35.0 μs	14.0 μs	禁止设置		f _{CLK} /4
1	0	1				26.5 μs	10.6 μs			f _{CLK} /3
1	1	0				18.0 μs	禁止设置		f _{CLK} /2	
1	1	1				9.5 μs			f _{CLK}	
×	×	×	0	1	增强电压	禁止设置				—
×	×	×	1	0	高速1	禁止设置				—
0	0	0	1	1	高速2	禁止设置	禁止设置	34.2 μs	17.1 μs	f _{CLK} /20
0	0	1					34.4 μs	17.2 μs	8.6 μs	f _{CLK} /10
0	1	0					27.6 μs	13.8 μs	6.9 μs	f _{CLK} /8
0	1	1				52.0 μs	20.8 μs	10.4 μs	5.2 μs	f _{CLK} /6
1	0	0				35.0 μs	14.0 μs	7.0 μs	3.5 μs	f _{CLK} /4
1	0	1				26.5 μs	10.6 μs	5.3 μs	禁止设置	f _{CLK} /3
1	1	0				18.0 μs	7.2 μs	3.6 μs		f _{CLK} /2
1	1	1				9.5 μs	3.8 μs	禁止设置		f _{CLK}

注意事项 1. 将 FR2 至 FR0、LV1 和 LV0 重写为除相同数据之外的值时，预先停止一次 A/D 转换（ADCS = 0）。
 2. 以上的转换时间不包括时钟频率误差。选择转换时间时要考虑到时钟频率误差。

备注 f_{CLK}: CPU/外围硬件时钟频率

表 11-2. A/D 转换时间选择(3/3)

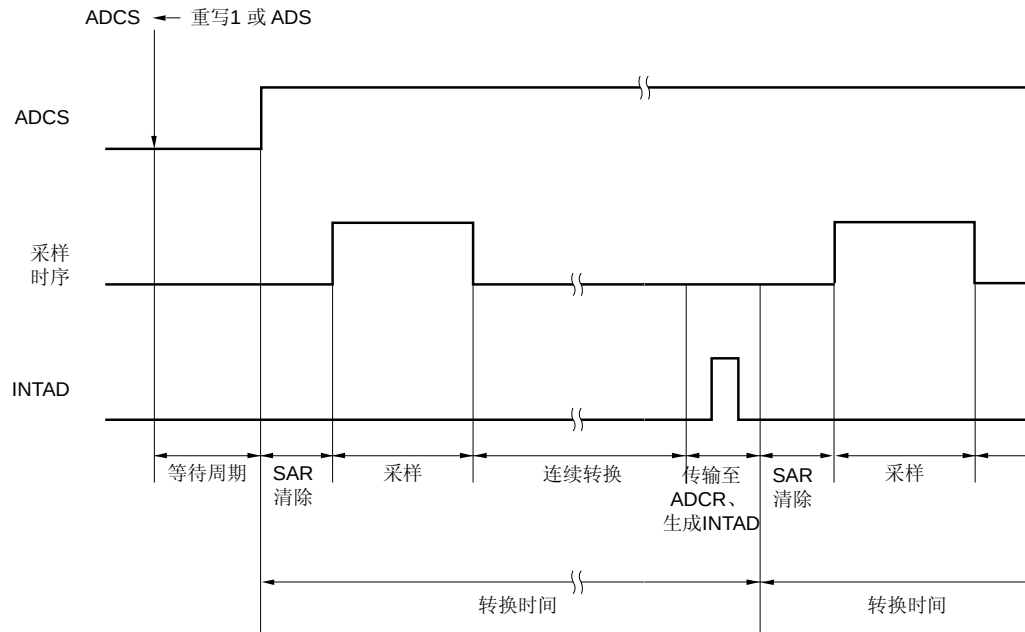
(3) $1.8\text{ V} \leq \text{AV}_{\text{REF}} \leq 4.0\text{ V}$

A/D转换器模式寄存器(ADM)					模式	转换时间选择				转换时钟 (f _{AD})
FR2	FR1	FR0	LV1	LV0		f _{CLK} = 2 MHz	f _{CLK} = 5 MHz	f _{CLK} = 10 MHz	f _{CLK} = 20 MHz	
×	×	×	0	0	标准	禁止设置				
0	0	0	0	1	增强电压	禁止设置	禁止设置	48.2 μs	24.1 μs	f _{CLK} /20
0	0	1					48.4 μs	24.2 μs	禁止设置	f _{CLK} /10
0	1	0					38.8 μs	禁止设置		f _{CLK} /8
0	1	1					29.2 μs			f _{CLK} /6
1	0	0				49.0 μs	禁止设置	f _{CLK} /4		
1	0	1				37.0 μs		f _{CLK} /3		
1	1	0				25.0 μs		f _{CLK} /2		
1	1	1				禁止设置		f _{CLK}		
×	×	×	1	0	高速1	禁止设置				—
×	×	×	1	1	高速2	禁止设置				—
其它					禁止设置					

注意事项 1. 将 FR2 至 FR0、LV1 和 LV0 重写为除相同数据之外的值时，预先停止一次 A/D 转换（ADCS = 0）。
 2. 以上的转换时间不包括时钟频率误差。选择转换时间时要考虑到时钟频率误差。

备注 f_{CLK} : CPU/外围硬件时钟频率

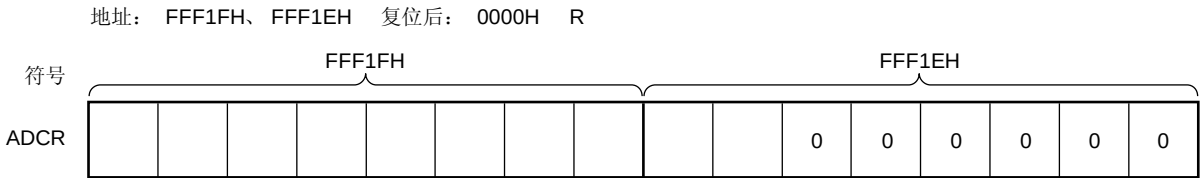
图 11-5. A/D 转换器采样和 A/D 转换时序



(3) 10 位 A/D 转换结果寄存器 (ADCR)

该寄存器是一个在选择模式存储 A/D 转换结果的 16 位寄存器。低 6 位固定为 0。每次 A/D 转换结束，将从逐次逼近寄存器中加载转换结果。转换结果的高 8 位存储在 FFF1FH 中，低 2 位存储在 FFF1EH 的高 2 位中。
可以由 16 位存储器处理指令读取 ADCR。
产生的复位信号将该寄存器清除为 0000H。

图 11-6. 10 位 A/D 转换结果寄存器(ADCR)的格式

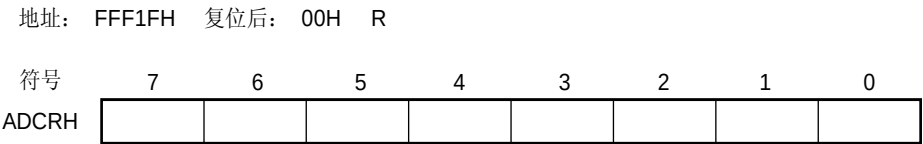


注意事项 写入 A/D 转换器模式寄存器 (ADM)、模拟输入通道指定寄存器 (ADS) 和 A/D 端口配置寄存器 (ADPC) 时，ADCR 的内容可能会不确定。在写入 ADM、ADS 和 ADPC 之前，读取转换完成后的转换结果。使用除以上时序外的时序可能会导致读取错误的转换结果。

(4) 8 位 A/D 转换结果寄存器 (ADCRH)

该寄存器是一个存储 A/D 转换结果的 8 位寄存器。存储 10 位分辨率的高 8 位。
可以由 8 位存储器处理指令设置 ADCRH。
产生的复位信号将该寄存器清除为 00H。

图 11-7. 8 位 A/D 转换结果寄存器(ADCRH)的格式



注意事项 写入 A/D 转换器模式寄存器 (ADM)、模拟输入通道指定寄存器 (ADS) 和 A/D 端口配置寄存器 (ADPC) 时，ADCRH 的内容可能会不确定。在写入 ADM、ADS 和 ADPC 之前，读取转换完成后的转换结果。使用除以上时序外的时序可能会导致读取错误的转换结果。

(5) 模拟输入通道指定寄存器 (ADS)

该寄存器指定要进行 A/D 转换的模拟电压的输入通道。

可以由 1 位或 8 位存储器处理指令设置 ADS。

产生的复位信号将该寄存器清除为 00H。

图 11-8. 模拟输入通道指定寄存器(ADS)的格式 (1/2)

地址: : FFF31H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
ADS	0	ADOAS	0	0	ADS3	ADS2	ADS1	ADS0

○ 选择模式 (ADMD = 0)

ADOAS	ADS3	ADS2	ADS1	ADS0	模拟输入通道	输入源
0	0	0	0	0	ANI0	P20/ANI0 引脚
0	0	0	0	1	ANI1	P21/ANI1 引脚
0	0	0	1	0	ANI2	P22/ANI2 引脚
0	0	0	1	1	ANI3	P23/ANI3 引脚
0	0	1	0	0	ANI4	P24/ANI4 引脚
0	0	1	0	1	ANI5	P25/ANI5 引脚
0	0	1	1	0	ANI6	P26/ANI6 引脚
0	0	1	1	1	ANI7	P27/ANI7 引脚
0	1	0	0	0	ANI8	P150/ANI8 引脚
0	1	0	0	1	ANI9	P151/ANI9 引脚
0	1	0	1	0	ANI10	P152/ANI10 引脚
0	1	0	1	1	ANI11	P153/ANI11 引脚
1	×	×	×	×	PGAO	可编程增益放大器输出信号
其它					禁止设置	

注 1. 允许设置
注 2. 禁止设置

注意事项 1. 务必将位 4、5 和 7 清除为“0”。

2. 在输入模式中, 使用端口模式寄存器 2、15 和 8 (PM2、PM15、PM8) 设置要用作 A/D 转换的通道。

3. 通过 ADS, 不要将由 ADPC 设置的引脚置为数字 I/O。

4. 在设置了可编程增益放大器的操作之后, 将自 PGAI 的可编程增益放大器的输出信号 (PGAO) 选作为模拟输入 (参考 11.4.1 A/D 转换器的基本操作)。

备注 1. ×: 忽略

2. P20/ANI0 至 P27/ANI9、P150/ANI8、P151/ANI9: 78K0R/KC3-L (44-引脚)

P20/ANI0 至 P27/ANI9、P150/ANI8 至 P152/ANI10: 78K0R/KC3-L (48-引脚) 和 78K0R/KD3-L

P20/ANI0 至 P27/ANI9、P150/ANI8 至 P153/ANI11: 78K0R/KE3-L

图 11-8. 模拟输入通道指定寄存器(ADS)的格式 (2/2)

地址: FFF31H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
ADS	0	ADOAS	0	0	ADS3	ADS2	ADS1	ADS0

○ 扫描模式 (ADMD = 1)											
		ADOAS	ADS3	ADS2	ADS1	ADS0	模拟输入通道				
							扫描 0	扫描 1	扫描 2	扫描 3	
↑注1↑注1↑注1 ↑注1↑注2↑注1	KC3-L (44-引脚)		0	0	0	0	0	ANI0	ANI1	ANI2	ANI3
	KC3-L (48-引脚)		0	0	0	0	1	ANI1	ANI2	ANI3	ANI4
	KD3-L		0	0	0	1	0	ANI2	ANI3	ANI4	ANI5
	KE3-L		0	0	0	1	1	ANI3	ANI4	ANI5	ANI6
			0	0	1	0	0	ANI4	ANI5	ANI6	ANI7
			0	0	1	0	1	ANI5	ANI6	ANI7	ANI8
			0	0	1	1	0	ANI6	ANI7	ANI8	ANI9
			0	0	1	1	1	ANI7	ANI8	ANI9	ANI10
			1	0	0	0	0	PGAO	ANI0	ANI1	ANI2
			1	0	0	0	1	PGAO	ANI1	ANI2	ANI3
			1	0	0	1	0	PGAO	ANI2	ANI3	ANI4
			1	0	0	1	1	PGAO	ANI3	ANI4	ANI5
			1	0	1	0	0	PGAO	ANI4	ANI5	ANI6
			1	0	1	0	1	PGAO	ANI5	ANI6	ANI7
			1	0	1	1	0	PGAO	ANI6	ANI7	ANI8
			1	0	1	1	1	PGAO	ANI7	ANI8	ANI9
			其它					禁止设置			

- 注 1. 允许设置
2. 禁止设置

- 注意事项 1. 务必将位 4、5 和 7 清除为“0”。
2. 在输入模式中, 使用端口模式寄存器 2、15 和 8 (PM2、PM15、PM8) 设置要用作 A/D 转换的通道。
3. 通过 ADS, 不要将由 ADPC 设置的引脚置为数字 I/O。
4. 在设置了可编程增益放大器的操作之后, 将自 PGAI 的可编程增益放大器的输出信号 (PGAO) 选作为模拟输入 (参考 11.4.1 A/D 转换器的基本操作)。

- 备注 1. ×: 忽略
2. P20/ANI0 至 P27/ANI9、P150/ANI8、P151/ANI9: 78K0R/KC3-L (44-引脚)
P20/ANI0 至 P27/ANI9、P150/ANI8 至 P152/ANI10: 78K0R/KC3-L (48-引脚) 和 78K0R/KD3-L
P20/ANI0 至 P27/ANI9、P150/ANI8 至 P153/ANI11: 78K0R/KE3-L

(6) A/D 端口配置寄存器 (ADPC)

该寄存器将 ANI0/P20 至 ANI7/P27 和 ANI8/P150 至 ANI11/P153 引脚转换为 A/D 转换器的模拟输入或者端口的数字 I/O。

可以由 8 位存储器处理指令设置 ADPC。

产生的复位信号将该寄存器设置为 10H。

图 11-9. A/D 端口配置寄存器(ADPC)的格式

地址: F0017H 复位后: 10H R/W

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

ADP C4	ADP C3	ADP C2	ADP C1	ADP C0	模拟输入 (A)/数字 I/O (D) 交换											
					端口 15				端口 2							
					ANI11 /P153	ANI10 /P152	ANI9 /P151	ANI8 /P150	ANI7 /P27	ANI6 /P26	ANI5 /P25	ANI4 /P24	ANI3 /P23	ANI2 /P22	ANI1 /P21	ANI0 /P20
0	0	0	0	0	A	A	A	A	A	A	A	A	A	A	A	A
0	0	0	0	1	A	A	A	A	A	A	A	A	A	A	A	D
0	0	0	1	0	A	A	A	A	A	A	A	A	A	A	D	D
0	0	0	1	1	A	A	A	A	A	A	A	A	A	D	D	D
0	0	1	0	0	A	A	A	A	A	A	A	A	D	D	D	D
0	0	1	0	1	A	A	A	A	A	A	A	D	D	D	D	D
0	0	1	1	0	A	A	A	A	A	A	D	D	D	D	D	D
0	0	1	1	1	A	A	A	A	A	D	D	D	D	D	D	D
0	1	0	0	0	A	A	A	A	D	D	D	D	D	D	D	D
0	1	0	0	1	A	A	A	D	D	D	D	D	D	D	D	D
0	1	0	1	0	A	A	D	D	D	D	D	D	D	D	D	D
0	1	0	1	1	A	D	D	D	D	D	D	D	D	D	D	D
1	0	0	0	0	D	D	D	D	D	D	D	D	D	D	D	D
其它					禁止设置											

- 注意事项**
1. 在输入模式中，使用端口模式寄存器 2 和 15 (PM2、PM15) 设置要用作 A/D 转换的通道。
 2. 通过模拟输入通道指定寄存器 (ADS)，不要将由 ADPC 设置的引脚置为数字 I/O。

备注

P20/ANI0 至 P27/ANI9、P150/ANI8、P151/ANI9: 78K0R/KC3-L (44-引脚)

P20/ANI0 至 P27/ANI9、P150/ANI8 至 P152/ANI10: 78K0R/KC3-L (48-引脚) 和 78K0R/KD3-L

P20/ANI0 至 P27/ANI9、P150/ANI8 至 P153/ANI11: 78K0R/KE3-L

(8) 端口模式寄存器 2、15 和 8 (PM2、PM15、PM8)

使用 ANI0/P20 至 ANI7/P27、ANI8/P150 至 ANI11/P153 和 PGAI/P80 引脚作为模拟输入端口，将 PM20 至 PM27、PM150 至 PM153 和 PM80 置为 1。此时，P20 至 P27、P150 至 P153 和 P80 的输出锁存可能为 0 或 1。如果将 PM20 至 PM27、PM150 至 PM153 和 PM80 置为 0，则它们不能用作模拟输入端口引脚。

可以由 1 位或 8 位存储器处理指令设置 PM2、PM15 和 PM8。

产生的复位信号将这些寄存器置为 FFH。

注意事项 如果将一个引脚设置为模拟输入端口，则读取到总是“0”而不是该引脚的电平。

备注 P20/ANI0 至 P27/ANI9、P150/ANI8、P151/ANI9: 78K0R/KC3-L (44-引脚)
P20/ANI0 至 P27/ANI9、P150/ANI8 至 P152/ANI10: 78K0R/KC3-L (48-引脚) 和 78K0R/KD3-L
P20/ANI0 至 P27/ANI9、P150/ANI8 至 P153/ANI11: 78K0R/KE3-L

图 11-11. 端口模式寄存器 2、15 和 8 (PM2、PM15、PM8) 的格式

地址: FFF22H 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

地址: FFF28H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PM8	1	1	1	1	PM83	PM82	PM81	PM80

地址: FFF2FH 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PM15	1	1	1	1	PM153	PM152	PM151	PM150

PMmn	Pmn 引脚 I/O 模式选择(mn = 20 至 27、150 至 153、80 至 83)
0	输出模式 (输出缓冲器开启)
1	输入模式(输出缓冲器关闭)

备注 如上图所示，显示了 78K0R/KE3-产品端口模式寄存器 2、8 和 15 的格式。有关其它产品端口模式寄存器的格式，参见 4.3 控制端口功能的寄存器中的 (1) 端口模式寄存器 (PMxx)。

根据 ADPC、ADS、PM2、PM15 和 PM8 的设置，ANI0/P20 至 ANI7/P27、ANI8/P150 至 ANI11/P153 和 PGAI/P80 引脚如下所示。

表 11-3. ANI0/P20 至 ANI7/P27、ANI8/P150 至 ANI11/P153 和 PGAI/P80 引脚的设置功能

ADPC	PM2、PM15和PM8	ADS	ANI0/P20至ANI7/P27、ANI8/P150至ANI11/P153和PGAI/P80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	选择 ANI.	模拟输入(待转换的)
		不选择 ANI.	模拟输入(不是待转换的)
	输出模式	选择ANI.	禁止设置
		不选择ANI.	

备注 P20/ANI0 至 P27/ANI9、P150/ANI8、P151/ANI9: 78K0R/KC3-L (44-引脚)
P20/ANI0 至 P27/ANI9、P150/ANI8 至 P152/ANI10: 78K0R/KC3-L (48-引脚) 和 78K0R/KD3-L
P20/ANI0 至 P27/ANI9、P150/ANI8 至 P153/ANI11: 78K0R/KE3-L

11.4 A/D 转换器操作

11.4.1 A/D 转换器的基本操作

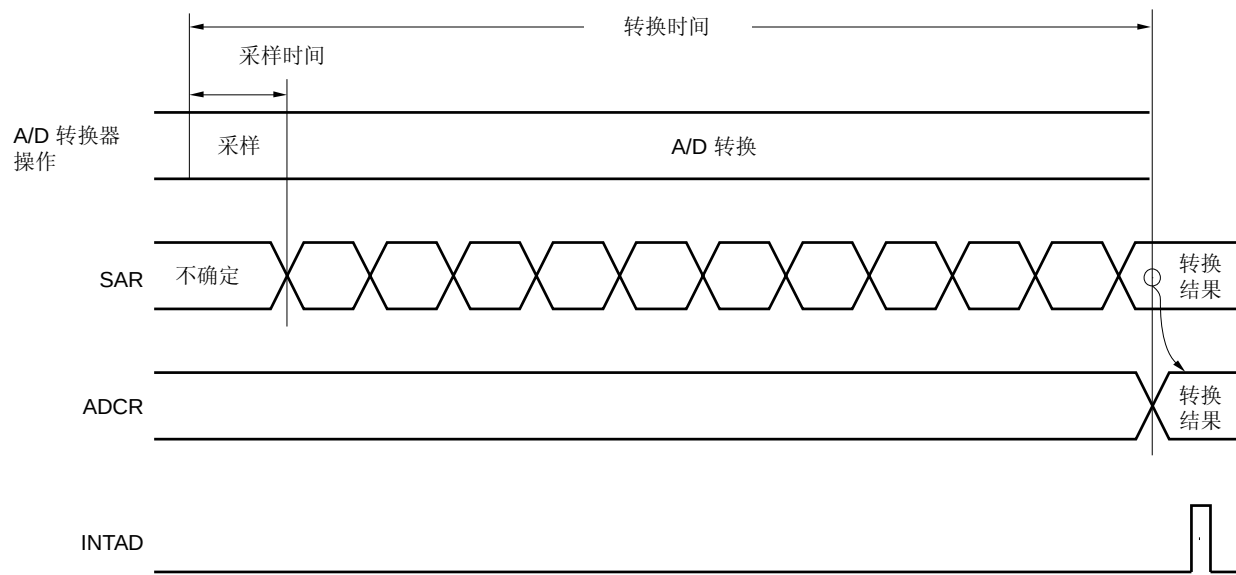
- <1> 将外围使能寄存器 0 (PER0) 的位 5 (ADCEN) 置为 1, 以启动 A/D 转换器的输入时钟供应。
- <2> 使用 ADM 的位 5 至位 1 (FR2 至 FR0、LV1 和 LV0) 设置 A/D 转换时间, 使用 ADM 的位 6 (ADMD) 设置操作模式。
- <3> 将 A/D 转换器模式寄存器 (ADM) 的位 0 (ADCE) 置为 1, 以启动 A/D 电压比较器的操作。
- <4> 使用 A/D 端口配置寄存器 (ADPC) 将 A/D 转换的通道设置为模拟输入, 使用端口模式寄存器 (PM2、PM15 和 PM8) 将其设置为输入模式。
- <5> 设置可编程增益放大器的操作, 以便为模拟输入通道设置可编程增益放大器输出 (PGAI 引脚) (参考 8.4.1 启动比较器和可编程增益放大器的操作)。
- <6> 使用模拟输入通道指定寄存器 (ADS) 为 A/D 转换选择一个通道。
- <7> 通过将 ADM 的位 7 (ADCS) 置为 1 启动转换操作。
(<8>至<14>是由硬件执行的操作。)
- <8> 由采样&保持电路对输入到所选模拟输入通道的电压进行采样。
- <9> 采样已完成一定时间时, 在保持状态中放置采样&保持电路并保持所采样的电压直到 A/D 转换操作结束。
- <10> 设置逐次逼近寄存器 (SAR) 的位 9。由档位选择器将串联电阻的档位电压设置为 $(1/2) AV_{REF}$ 。
- <11> 由电压比较器比较串联电阻的电压和所采样的电压之间的电压差。如果模拟输入大于 $(1/2) AV_{REF}$, SAR 的 MSB 仍然为 1。如果模拟输入小于 $(1/2) AV_{REF}$, MSB 复位为 0。
- <12> 接下来, SAR 的位 8 自动设置为 1, 且操作进入到下一次比较。如下所示, 根据位 9 的预置值, 选择串联电阻的档位电压。
 - 位 9 = 1: $(3/4) AV_{REF}$
 - 位 9 = 0: $(1/4) AV_{REF}$
 比较档位电压和采样电压, 并如下操作 SAR 的位 8。
 - 采样电压 $\geq$ 档位电压: 位 8 = 1
 - 采样电压 < 档位电压: 位 8 = 0
- <13> 用这种方式继续比较直到 SAR 的位 0。
- <14> 10 位比较完成之际, 有效的数字结果值保留在 SAR 中, 并将结果值传输到 A/D 转换结果寄存器 (ADCR、ADCRH), 然后锁存。
同时, 也会产生 A/D 转换结束中断请求 (INTAD)。
- <15> 重复步骤<8>至<14>, 直到 ADCS 位清除为 0。
要停止 A/D 转换器, 需将 ADCS 清除为 0。
要从 ADCE = 1 的状态重写开始 A/D 转换, 需从<7>开始。ADCE = 0 时, 要重新开始 A/D 转换, 需将 ADCE 置为 1, 等待 $1 \mu s$ 或者更长时间, 然后开始<7>。要更改 A/D 转换的通道, 从<6>开始。

注意事项 确保<3>至<7>的周期至少为 $1 \mu s$ 。

备注 A/D 转换结果寄存器的两种类型可用。

- ADCR (16 位): 存储 10 位 A/D 转换值
- ADCRH (8 位): 存储 8 位 A/D 转换值

图 11-12. A/D 转换器的基本操作



继续执行 A/D 转换操作直到 A/D 转换器模式寄存器（ADM）的位 7（ADCS）通过软件复位（0）。

如果在 A/D 转换操作期间对模拟输入通道指定寄存器（ADS）执行写操作，则初始化转换操作，如果设置 ADCS 位（1），则转换再次从头开始。

产生复位信号可以将 A/D 转换结果寄存器（ADCR、ADCRH）清除为 0000H 或 00H。

11.4.2 输入电压和转换结果

下列表达式给出了输入至模拟输入引脚（ANI0 至 ANI11、PGAI）的模拟输入电压和理论上的 A/D 转换结果之间的关系。

$$\text{SAR} = \text{INT} \left(\frac{V_{\text{AIN}}}{AV_{\text{REF}}} \times 1024 + 0.5 \right)$$
$$\text{ADCR} = \text{SAR} \times 64$$

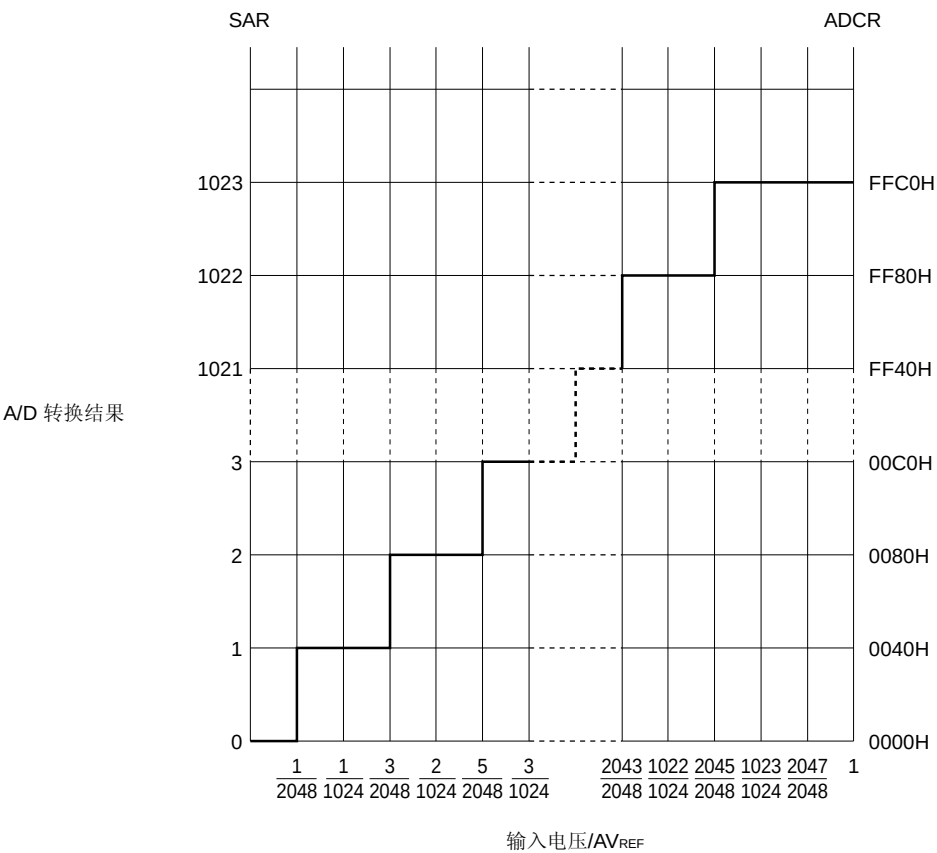
或

$$\left(\frac{\text{ADCR}}{64} - 0.5 \right) \times \frac{AV_{\text{REF}}}{1024} \leq V_{\text{AIN}} < \left(\frac{\text{ADCR}}{64} + 0.5 \right) \times \frac{AV_{\text{REF}}}{1024}$$

其中，INT(): 返回圆括号内值的整数部分
V_AIN: 模拟输入电压
AV_{REF}: AV_{REF} 引脚电压
ADCR: A/D 转换结果寄存器（ADCR）的值
SAR: 逐次逼近寄存器

图 11-13 显示了模拟输入电压和 A/D 转换结果之间的关系。

图 11-13. 模拟输入电压和 A/D 转换结果之间的关系



11.4.3 A/D 转换器操作模式

A/D 转换器操作模式中提供了选择模式和扫描模式。

(1) 选择模式

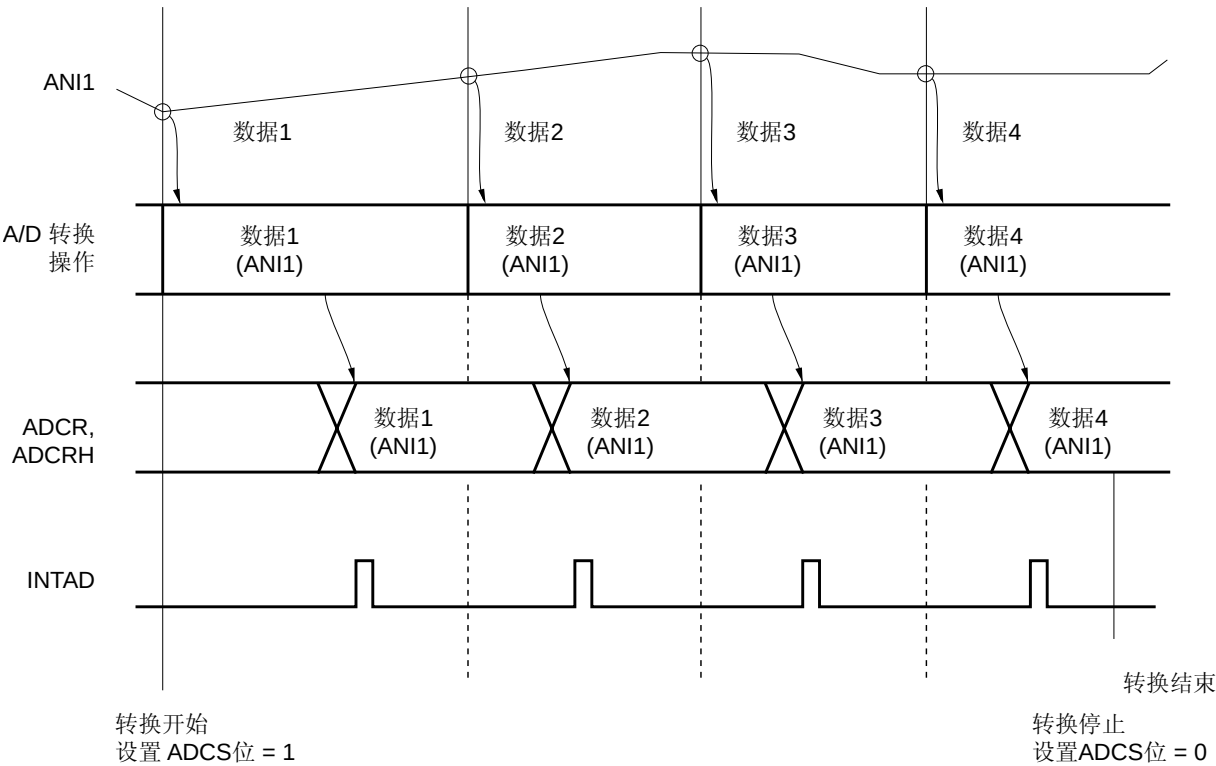
当 A/D 转换器模式寄存器（ADM）的 ADMD 位为 0 时，模拟输入通道指定寄存器（ADS）指定的一个模拟输入进行 A/D 转换。

A/D 转换完成时，转换结果存储在 A/D 转换结果寄存器（ADCR）中，并产生 ADCR 转换结束中断请求信号（INTAD）。

A/D 转换已经完成后，连续重复 A/D 转换，除非将 ADCS 位置为 0。

如果在转换期间将什么东西写入 ADM 或 ADS，则终止 A/D 转换。这种情况下，再次从头开始 A/D 转换。

图 11-14. 选择模式操作时序示例



(2) 扫描模式

当 A/D 转换器模式寄存器 (ADM) 的 ADMD 位为 1 时, 由模拟输入通道指定寄存器 (ADS) 指定的扫描 0 至 3 的四个模拟输入通道连续进行 A/D 转换。按次序执行 A/D 转换, 从扫描 0 指定的模拟输入通道开始。

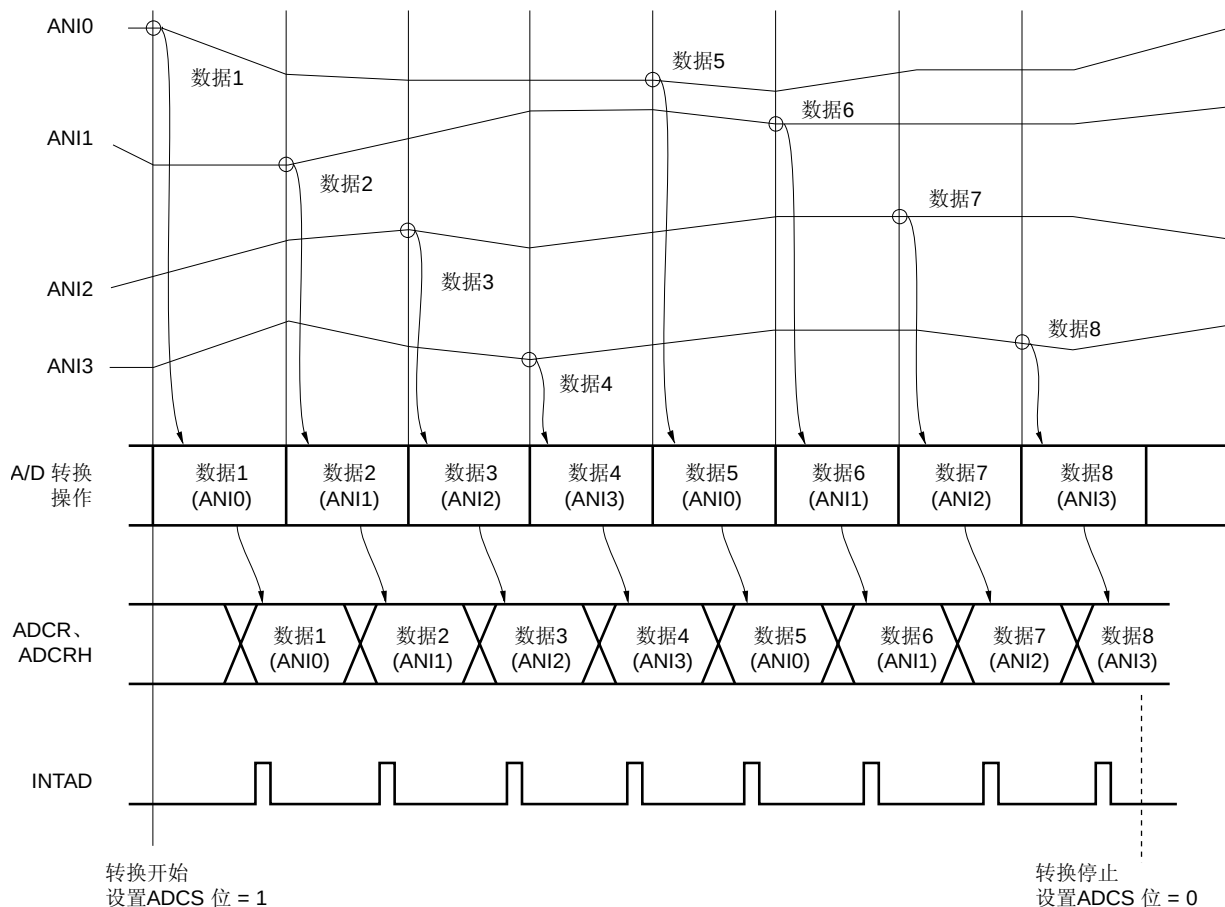
一个模拟输入的 A/D 转换完成时, 转换结果存储在 A/D 转换结果寄存器 (ADCR) 中, 并产生 ADCR 转换结束中断请求信号 (INTAD)。

所有模拟输入通道的 A/D 转换结果都存储在 ADCR 中。因此, 建议在每路模拟输入通道转换完成后, 立即把 ADCR 中的内容保存到 RAM 中。

A/D 转换已经完成后, 连续重复 A/D 转换, 除非将 ADCS 位置为 0。

如果在转换期间将什么东西写入 ADM 或 ADS, 则终止 A/D 转换。这种情况下, 从扫描 0 的模拟输入通道重新开始 A/D 转换。

图 11-15. 扫描模式操作时序示例



设置方法如下所述。

- <1> 将外围使能寄存器 0 (PER0) 的位 5 (ADCEN) 置为 1。
- <2> 使用 ADM 的位 5 至位 1 (FR2 至 FR0、LV1 和 LV0) 选择转换时间, 使用 ADM 的位 6 (ADMD) 选择操作模式。
- <3> 将 A/D 转换器模式寄存器 (ADCE) 的位 0 置为 1。
- <4> 使用 A/D 端口配置寄存器 (ADPC) 的位 4 至位 0 (ADPC4 至 ADPC0)、端口模式寄存器 2 (PM2) 的位 7 至位 0 (PM27 至 PM20)、端口模式寄存器 15 (PM15) 的位 3 至位 0 (PM153 至 PM150)、端口模式寄存器 8 (PM8) 的位 0 (PM80) 设置将用于模拟输入模式中的通道。
- <5> 设置可编程增益放大器的操作, 以便为模拟输入通道设置可编程增益放大器输出 (PGAI 引脚) (参考 8.4.1 启动比较器和可编程增益放大器的操作)。
- <6> 使用模拟输入通道指定寄存器 (ADS) 的位 6 和位 3 至位 0 (ADOAS、ADS3 至 ADS0) 选择要使用的通道。
- <7> 将 ADM 的位 7 (ADCS) 置为 1, 以开始 A/D 转换。
- <8> A/D 转换已完成时, 都会产生一个中断请求信号 (INTAD)。
- <9> 将 A/D 转换数据传输到 A/D 转换结果寄存器 (ADCR、ADCRH)。
- <更改通道>
 - <10> 使用 ADS 的位 6 和位 3 至位 0 (ADOAS、ADS3 至 ADS0) 更改通道以开始 A/D 转换。
 - <11> A/D 转换已完成时, 都会产生一个中断请求信号 (INTAD)。
 - <12> 将 A/D 转换数据传输到 A/D 转换结果寄存器 (ADCR、ADCRH)。
- <完成 A/D 转换>
 - <13> 将 ADCS 清除为 0。
 - <14> 将 ADCE 清除为 0。
 - <15> 将外围使能寄存器 0 (PER0) 的位 5 (ADCEN) 清除为 0。

注意事项 1. 确保<3>至<7>的周期至少为 1 μ s。

2. <3>可能在<4>和<6>之间进行。

3. 可以省略<3>。但是, 在这种情况下要忽略<7>后第一次转换的数据。

4. <8>到<11>的周期与使用 ADM 的位 5 至位 1 (FR2 至 FR0、LV1、LV0) 设置的转换时间不同。<10>至<11>的周期就是使用 FR2 至 FR0、LV1 和 LV0 设置的转换时间。

11.5 如何读取 A/D 转换器参数表

在这里，解释了 A/D 转换器的专用术语。

(1) 分辨率

这是可识别的最小模拟输入电压。也就是说，数字输出中每一位的模拟输入电压的百分比称为 **1LSB**（最低有效位）。考虑进最大范围的 **1LSB** 的百分比是由 **%FSR**（最大范围）表示的。

当分辨率为 10 位时，**1LSB** 表示如下。

$$\begin{aligned} 1\text{LSB} &= 1/2^{10} = 1/1024 \\ &= 0.098\%\text{FSR} \end{aligned}$$

精确度和分辨率无关，而是由总体的误差决定的。

(2) 总误差

此项显示了实际测量值和理论值的最大误差值。

零标称误差、满量程误差、积分线性误差以及组成这些的微分线性误差构成总体误差。

注意在参数表的总体误差里没有包含量化误差。

(3) 量化误差

当模拟值转换为数字值时，会自然产生 $\pm 1/2\text{LSB}$ 的误差。A/D 转换器中，在 $\pm 1/2\text{LSB}$ 范围内的模拟输入电压转换成相同的数字编码，所以不可能避免量化误差。

注意在参数表中的整体误差、零标称误差、满量程误差、积分线性误差和微分线性误差之中不包括量化误差。

图 11-16. 整体误差

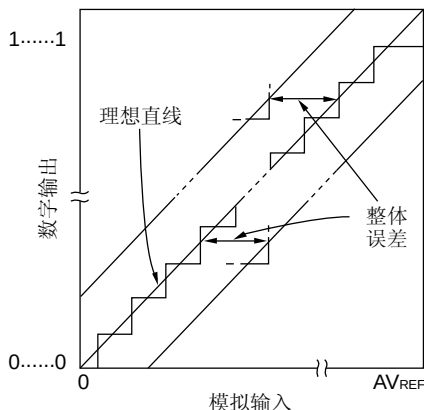
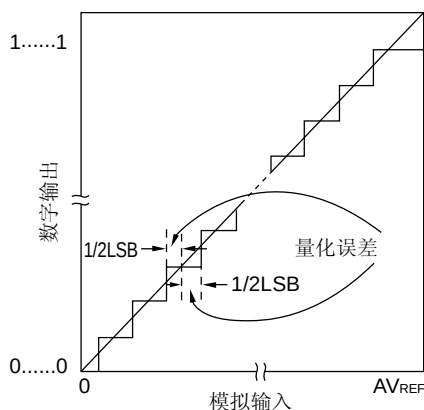


图 11-17. 量化误差



(4) 零标称误差

零度误差是指数字输出从 0.....000 变为 0.....001 时，模拟输入电压的实际测量值与理论值 ($1/2\text{LSB}$) 之差。

当数字输出从 0.....001 变为 0.....010 时，如果实际测量值大于理论值，则显示模拟输入电压的实际测量值和理论值 ($3/2\text{LSB}$) 的差。

(5) 满量程误差

当数字输出从 1.....110 变为 1.....111 时，显示模拟输入电压的实际测量值和理论值（满量程- 3/2LSB）的差。

(6) 积分线性误差

积分线性误差说明了转换特征偏离理想线性关系的程度。它表示当零标称误差和满量程误差均为 0 时，实际测量值和理想直线之差的最大值。

(7) 微分线性误差

当代码输出的理想宽度为 1LSB 时，微分线性误差表示实际测量值与理想值之差。

图 11-18. 零标称误差

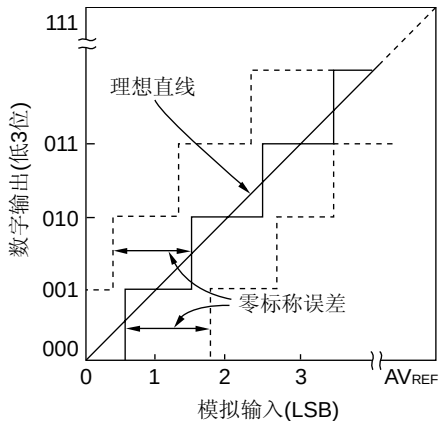


图 11-19. 满量程误差

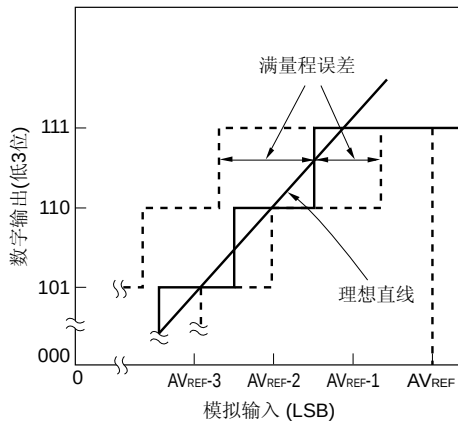


图 11-20. 积分线性误差

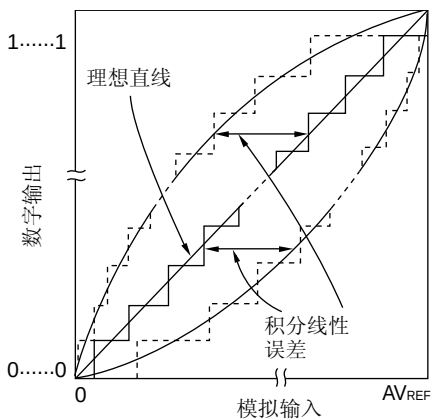
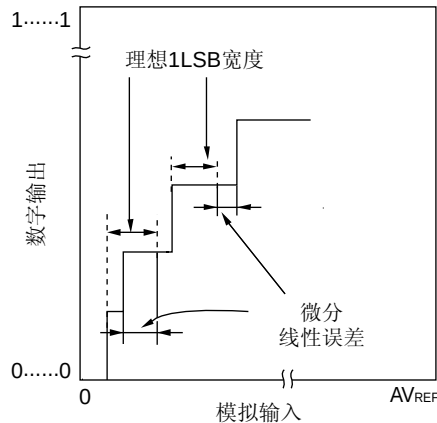


图 11-21. 微分线性误差



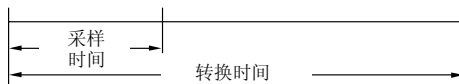
(8) 转换时间

转换时间表示从采样开始到获取到数字输出的时间。

在特性表中，采样时间包括在转换时间内。

(9) 采样时间

采样时间即：为将由采样&保持电路采样的模拟电压开启模拟开关的时间。



11.6 A/D 转换器的注意事项

(1) STOP 模式中的操作电流

<R>

在停止 A/D 转换器（将 A/D 转换器模式寄存器（ADM）的位 7（ADCS）置为 0）后，转换到 STOP 模式。同时，通过将 A/D 转换器模式寄存器（ADM）的位 0（ADCE）置为 0 可以减少操作电流。
要从待机状态重新开始，需将中断请求标志寄存器 1L（IF1L）的位 0（ADIF）清除为 0，然后开始操作。

(2) ANI0 至 ANI11 的输入范围

观察 ANI0 至 ANI11 输入电压的额定范围。如果将等于或高于 AV_{REF} 和等于或低于 AV_{SS} 的电压（即使在绝对最大额定值的范围内）输入到一个模拟输入通道，则此通道的转换值变得不确定。此外，其它通道的转换值也可能受到影响。

(3) 操作冲突

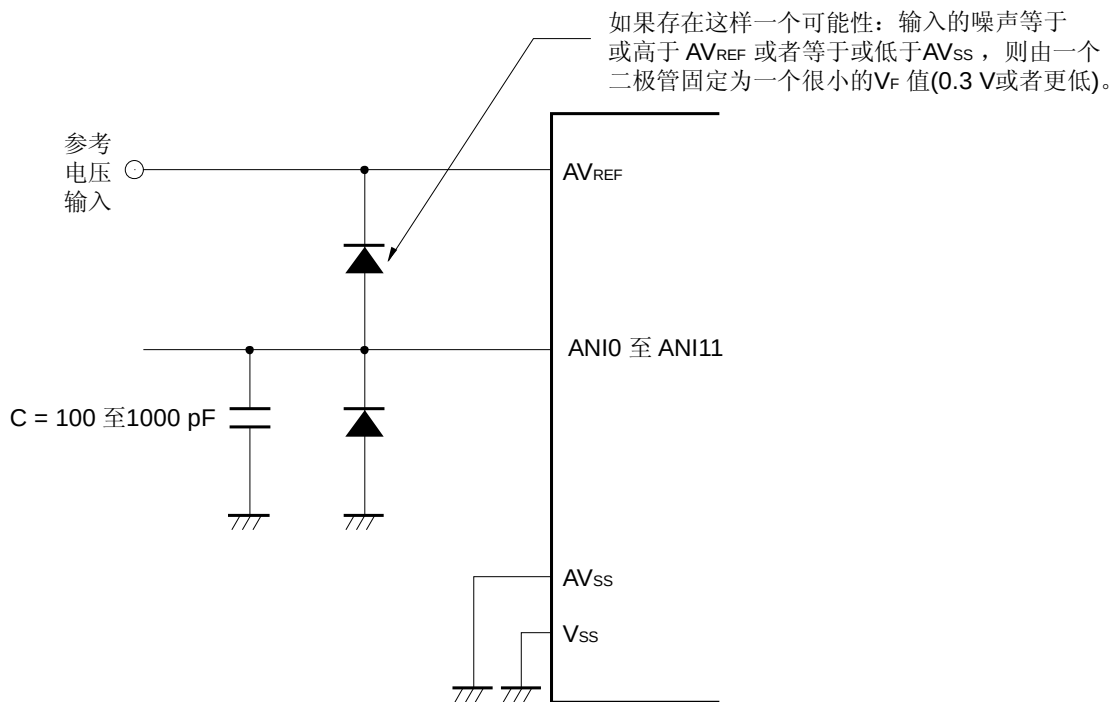
- <1> 转换结束之际，由指令进行的 A/D 转换结果寄存器（ADCR、ADCRH）的写操作和 ADCR 或 ADCRH 的读操作冲突。
ADCR 或 ADCRH 读操作优先。读操作之后，新的转换结果被写入 ADCR 或 ADCRH。
- <2> 转换结束之际，ADCR 或 ADCRH 的写操作与 A/D 转换器模式寄存器（ADM）的写操作、模拟输入通道指定寄存器（ADS）或 A/D 端口配置寄存器（ADPC）的写操作冲突。
ADM、ADS 或 ADPC 的写操作优先。不执行 ADCR 或 ADCRH 的写操作，也不产生转换结束中断信号（INTAD）。

(4) 应对噪声的措施

- 要保持 10 位分辨率，必须注意输入至 AV_{REF} 引脚和引脚 ANI0 至 ANI11 的噪声。
- <1> 通过一个低等效电阻和一个响应供电电源的适当频率连接一个电容。
 - <2> 模拟输入源的输出阻抗越高，影响越大。为了减小噪声，按如图 11-22 中所建议的方式连接一个外部电容器。
 - <3> 在转换期间不要在这些引脚和其它引脚之间进行转换。
 - <4> 如果在转换开始之后立即设置 HALT 模式，则会提高精确度。

备注 ANI0 至 ANI9: 78K0R/KC3-L (44-引脚)
ANI0 至 ANI10: 78K0R/KC3-L (48-引脚) 和 78K0R/KD3-L
ANI0 至 ANI11: 78K0R/KE3-L

图 11-22. 模拟输入引脚连线

**(5) ANI0/P20 至 ANI7/P27 和 ANI8/P150 至 ANI11/P153**

- <1> 模拟输入引脚（ANI0 至 ANI11）也会用作输入端口引脚（P20 至 P27、P150 至 P153）。选定 ANI0 至 ANI11 中的任意一个执行 A/D 转换时，当处理转换时不要访问 P20 至 P27 和 P150 至 P153；否则可能会降低转换分辨率。建议选择这样的引脚，它们从离 AV_{REF} 最远的 ANI0/P20 开始作为 P20 至 P27 和 P150 至 P153 使用。
- <2> 如果将数字脉冲应用到当前用于 A/D 转换引脚的邻近引脚上，由于耦合噪声，可能不会获得 A/D 转换的预期值。因此，不要把脉冲加到正在进行 A/D 转换引脚的邻近引脚上。

(6) ANI0 至 ANI11 引脚的输入阻抗

A/D 转换器在采样时间中为采样加入一个采样电容器。

因此，当未进行采样时，仅有漏电流流过，在采样期间会流经电容器的电流。因此，在其它状态下，根据是否在进行采样，输入阻抗会有所波动，

然而，要确保采样有效，建议保持模拟输入源的输出阻抗在 $1\text{ k}\Omega$ 以下，并将一个大约 100 pF 的电容连接到 ANI0 至 ANI11 引脚（参见图 11-22）。

(7) AV_{REF} 引脚的输入阻抗

将几十 $\text{k}\Omega$ 的串联电阻连接在 AV_{REF} 和 AV_{SS} 引脚之间。

因此，如果参考电压源的输出阻抗很高，这将导致其与 AV_{REF} 和 AV_{SS} 引脚之间的串联电阻串联，从而导致较大的参考电压误差。

备注	P20/ANI0 至 P27/ANI9、P150/ANI8、P151/ANI9: 78K0R/KC3-L (44-引脚)
	P20/ANI0 至 P27/ANI9、P150/ANI8 至 P152/ANI10: 78K0R/KC3-L (48-引脚) 和 78K0R/KD3-L
	P20/ANI0 至 P27/ANI9、P150/ANI8 至 P153/ANI11: 78K0R/KE3-L

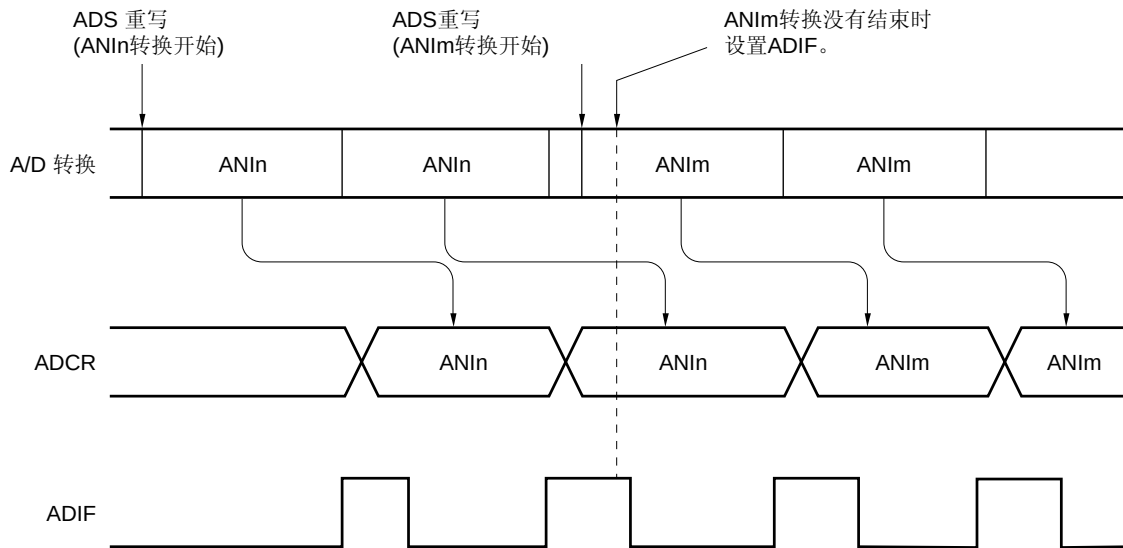
(8) 中断请求标志 (ADIF)

即使改变模拟输入通道指定寄存器 (ADS)，也不会清除中断请求标志 (ADIF)。

因此，如果在 A/D 转换期间更改模拟输入引脚，则在重写 ADS 之前可设置 A/D 转换结果和更改之前模拟输入的 ADIF。因此，同时需注意的是：当在重写 ADS 之后立即读取 ADIF 时，纵使更改后模拟输入的实际 A/D 转换还没有结束，也会设置 ADIF。

停止 A/D 转换然后恢复时，要在 A/D 转换操作重新开始之前清除 ADIF。

图 11-23. A/D 转换结束中断请求产生的时序



- 备注**
1. 78K0R/KC3-L (44-引脚): $n = 0$ 至 9
78K0R/KC3-L (48-引脚)、78K0R/KD3-L: $n = 0$ 至 10
78K0R/KE3-L: $n = 0$ 至 11
 2. 78K0R/KC3-L (44-引脚): $m = 0$ 至 9
78K0R/KC3-L (48-引脚)、78K0R/KD3-L: $m = 0$ 至 10
78K0R/KE3-L: $m = 0$ 至 11

(9) 刚刚启动 A/D 转换之后的转换结果

如果在 ADCE 位被置为 1 后的 $1\mu\text{s}$ 内将 ADCS 位设置为 1，或者如果在 ADCE bit = 0 时将 ADCS 位置为 1，则在 A/D 转换开始后瞬间的第一次 A/D 转换值可能不会在额定范围内下降。对此采取措施，例如循环检测 A/D 转换结束中断请求 (INTAD) 和删除第一次转换的结果。

(10) A/D 转换结果寄存器 (ADCR、ADCRH) 读操作

对 A/D 转换器模式寄存器 (ADM)、模拟输入通道指定寄存器 (ADS) 和 A/D 端口配置寄存器 (ADPC) 执行读操作时，ADCR 和 ADCRH 的内容可能会不确定。在写入 ADM、ADS 或 ADPC 之前，读取转换完成后的转换结果。使用除以上时序外的时序可能会导致读取错误的转换结果。

(11) 内部等效电路

模拟输入块的等效电路如下所示。

图 11-24. ANIn 引脚的内部等效电路

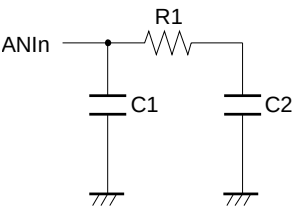


表 11-4. 等效电路的电阻和电容值（参考值）

AVREF	模式	R1	C1	C2
4.0 V ≤ VDD ≤ 5.5 V	标准	5.2 kΩ	8 pF	6.26 pF
	高速1	5.2 kΩ		
	高速2	7.8 kΩ		
2.7 V ≤ VDD < 4.0 V	标准	18.6 kΩ		
	高速2	7.8 kΩ		
1.8 V ≤ VDD < 4.0 V	增强电压	169.8 kΩ		

- 备注
- 1. 在表 11-4 中显示的电阻和电容值得不到保证。
 - 2. 78K0R/KC3-L (44-引脚): n = 0 至 9
78K0R/KC3-L (48-引脚): n = 0 至 10
78K0R/KD3-L: n = 0 至 10
78K0R/KE3-L: n = 0 至 11

第十二章 串行阵列部件

串行阵列部件每个单元具有四个串行通道，可以组合使用两个或多个不同的串行接口（3-线串行(CSI)、UART和简
易的I²C）。

由 78K0R/Kx3-L 支持的各个通道的功能分配如下表所示。

通道	用作 CSI	用作 UART	用作简易的I ² C
0	CSI00	UART0 (支持 LIN 总线)	—
1	CSI01		—
2	CSI10	UART1	IIC10
3	—		—

（组合示例） 当“UART0”用于通道 0 和 1 时，不能使用 CSI00 和 CSI01，但是可以使用 CSI10、
UART1 或 IIC10。

12.1 串行阵列部件的功能

由 78K0R/Kx3-L 支持的各串行接口具有以下特性。

12.1.1 3-线串行 I/O（CSI00、CSI01、CSI10）

这是使用三线的时钟通信功能，即：串行时钟（SCK）线和串行数据（SI 和 SO）线。

[数据发送/接收]

- 7 或 8 位的数据长度
- 发送/接收数据的相位控制
- 可选择的 MSB/LSB 先行
- 发送/接收数据的电平设置

[时钟控制]

- 主机/从机选择
- I/O 时钟的相位控制
- 经过各通道的预分频器和内部计数器的传输周期的设置
- 最大传输率

主机通信期间： 最大值 f_{CLK}/4，从机通信期间： 最大值 f_{MCK}/6[※]

[中断功能]

- 传输结束中断/缓冲器空中断

[错误检测标志]

- 超限错误

注 在满足 SCK 周期（t_{KCY}）特性的范围内使用时钟（参见第二十八章 电气规范（目标））。

12.1.2 UART (UART0、UART1)

这是指使用两条信号线的开始-停止同步功能，即：串行数据发送（TxD）和串行数据接收（RxD）线。它与通信方同步发送或接收数据（使用一个内部波特率）。通过使用两个通道可以实现全双工 UART 通信，一个专用于发送（偶数通道），另一个用于接收（奇数通道）。

[数据发送/接收]

- 5、7 或 8 位的数据长度
- 选择 MSB/LSB 先行
- 发送/接收数据的电平设置和反向选择
- 附加奇偶位和奇偶校验功能
- 附加终止位

[中断功能]

- 传输结束中断/缓冲器空闲中断
- 帧错误、奇偶错误或超限错误发生时的错误中断

[错误检测标志]

- 帧错误、奇偶校验错误或超限错误

LIN 总线在 UART0（0 或 1 通道）中普遍使用。

[LIN 总线功能]

- 唤醒信号检测
- 同步中断域（SBF）检测
- 测量同步域、计算波特率

} 使用外部中断（INTP0）或定时器阵列部件 TAUS。

12.1.3 简易的I²C (IIC10)

这是用来与两个或多个设备进行通信的时钟通信功能，使用两条信号线：串行时钟（SCL）和串行数据（SDA）。简易的I²C是为使用EEPROM、闪存或A/D转换器之类的设备进行单向通信设计的，它仅作为主机使用，并不具有检测等待状态的功能。

通过使用软件和操作控制寄存器，确保遵循开始和停止条件的 AC 指定。

[数据发送/接收]

- 主机发送、主机接收（仅使用一个单主机的主机功能）
- ACK输出功能^注和ACK检测功能
- 8 位的数据长度（当发送地址时，由高 7 位指定地址，最低有效位用于 R/W 控制。）
- 开始条件和停止条件的手动生成

[中断功能]

- 传输结束中断

[错误检测标志]

- 奇偶校验错误（ACK 错误）

* [未受简易I²C支持的功能]

- 从机发送、从机接收
- 仲裁失败检测功能
- 等待检测功能

注 当接收最后的数据时，如果将 0 写入 SOE02（SOE0 寄存器）位，将不会输出 ACK，且串行通信数据输出会停止。有关细节参见 12.7.3（2）处理流程。

备注 要使用I²C总线的全部功能，参见第十三章 串行接口IICA。

12.2 串行阵列部件的配置

串行阵列部件包括以下硬件。

表 12-1. 串行阵列部件的配置

项目	配置
移位寄存器	8 位
缓冲器寄存器	串行数据寄存器 0n (SDR0n)*的低 8 位
串行时钟 I/O	SCK00、SCK01、SCK10 引脚 (适用于 3-线串行 I/O)、SCL10 引脚(适用于简易的I ² C)
串行数据输入	SI00、SI01、SI10 引脚 (适用于 3-线串行 I/O)、RxD0 引脚(适用于支持 LIN 总线的 UART)、RxD1 引脚(适用于 UART)
串行数据输出	SO00、SO01、SO10 pins (适用于 3-线串行 I/O)、TxD0 引脚 (适用于支持 LIN 总线的 UART)、TxD1 引脚(适用于 UART)、输出控制器
串行数据 I/O	SDA10 引脚(适用于简易的I ² C)
控制寄存器	<部件设置块的寄存器> • 外围使能寄存器 0 (PER0) • 串行时钟选择寄存器 0 (SPS0) • 串行通道使能状态寄存器 0 (SE0) • 串行通道开始寄存器 0 (SS0) • 串行通道停止寄存器 0 (ST0) • 串行输出使能寄存器 0 (SOE0) • 串行输出寄存器 0 (SO0) • 串行输出电平寄存器 0 (SOL0) • 输入开关控制寄存器 (ISC) • 噪声滤波器使能寄存器 0 (NFEN0)
	<各通道的寄存器> • 串行数据寄存器 0n (SDR0n) • 串行模式寄存器 0n (SMR0n) • 串行通信操作设置寄存器 0n (SCR0n) • 串行状态寄存器 0n (SSR0n) • 串行标志清除触发器寄存器 0n (SIR0n) • 端口输入模式寄存器 3、7 (PIM3、PIM7) • 端口输出模式寄存器 3、7 (POM3、POM7) • 端口模式寄存器 3、7 (PM3、PM7) • 端口寄存器 3、7 (P3、P7)

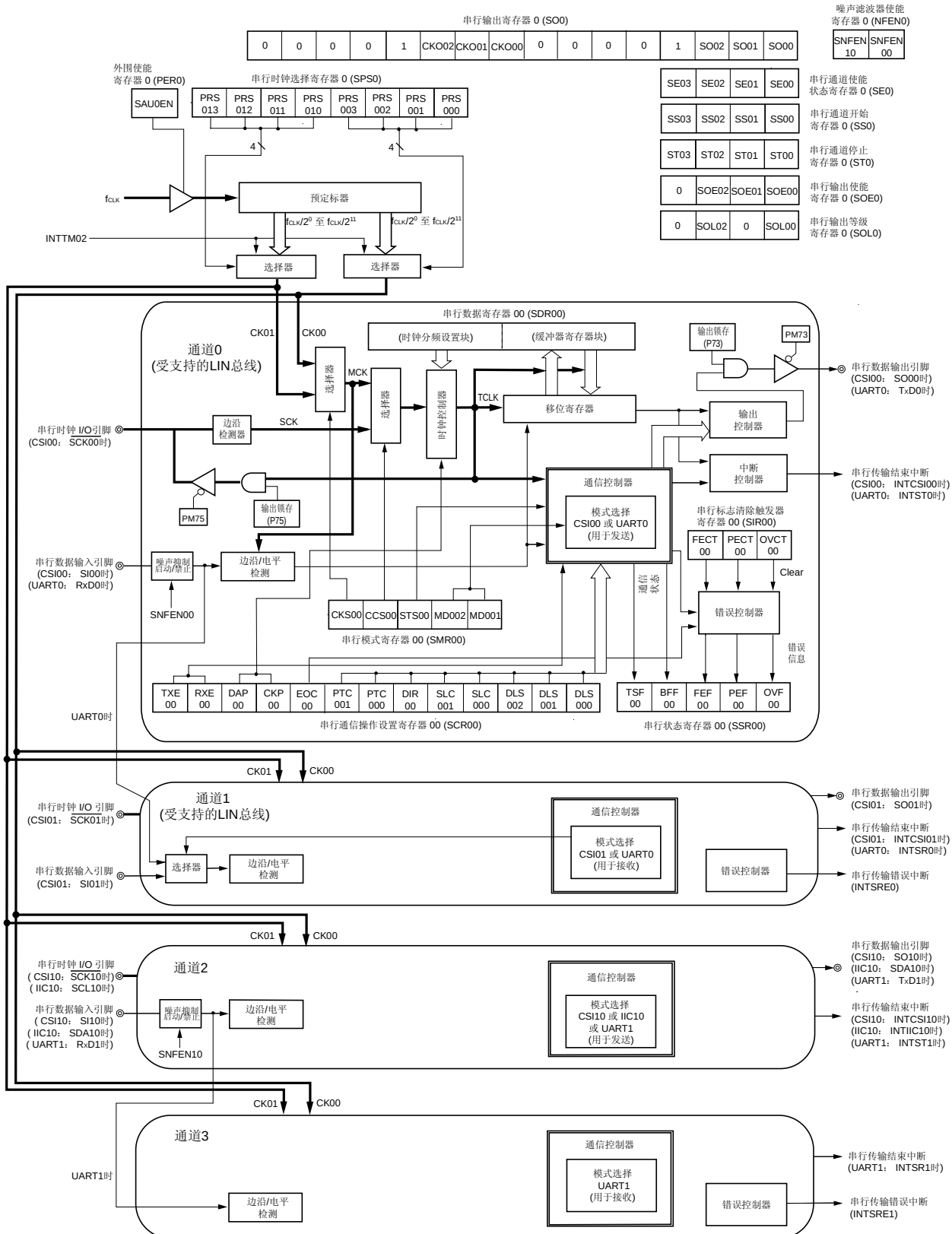
注 根据通信模式的不同，可以将串行数据寄存器 0n (SDR0n) 的低 8 位读作或写作以下 SFR。

- CSIp 通信 ... SIOp (CSIp 数据寄存器)
- UARTq 接收 ... RXDq (UARTq 接收数据寄存器)
- UARTq 发送 ... TXDq (UARTq 发送数据寄存器)
- IIC10 通信 ... SIO10 (IIC10 数据寄存器)

备注 n: 通道编号 (n = 0 至 3),
P: CSI 编号 (p = 00、01、10),
Q: UART 编号 (q = 0、1)

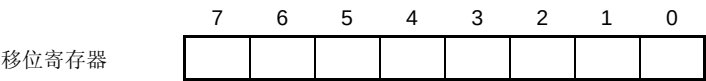
图 12-1.给出了串行阵列部件的框图。

图 12-1. 串行阵列部件的框图



(1) 移位寄存器

这是一款将并行数据转换为串行数据的 8 位寄存器，反之亦然。
接收期间，该寄存器将输入到串行引脚的数据转换为并行数据。
发送数据时，设置给该寄存器的值将从串行输出引脚以串行数据的形式输出。
不能由程序直接操作该移位寄存器。
要读取或写入移位寄存器，需使用串行数据寄存器 0n（SDR0n）的低 8 位。



(2) 串行数据寄存器 0n（SDR0n）的低 8 位

SDR0n 是通道 n 的发送/接收数据寄存器（16 位）。位 7 至位 0 作为发送/接收缓冲寄存器使用，位 15 至位 9 作为设置操作时钟（MCK）分频比的寄存器使用。
接收数据时，移位寄存器转换的并行数据存储在低 8 位。要发送数据时，将要传输到移位寄存器的发送数据设置给低 8 位。
根据 SCR0n 寄存器位 0 至位 2（DLS0n0 至 DLS0n2）的设置，不管数据的输出次序如何，存储在该寄存器低 8 位的数据如下。

- 5 位数据长度 (存储在 SDR0n 寄存器的位 0 至位 4) (仅能在 UART 模式中设置)
- 7 位数据长度(存储在 SDR0n 寄存器的位 0 至位 6)
- 8 位数据长度(存储在 SDR0n 寄存器的位 0 至位 7)

SDR0n 可按 16 位为单位进行读写。
根据通信模式的不同，可以将SDR0n的低 8 位读作或写作^注以下SFR。

- CSIp 通信 ... SIOp (CSIp 数据寄存器)
 - UARTq 接收 ... RXDq (UARTq 接收数据寄存器)
 - UARTq 发送 ... TXDq (UARTq 发送数据寄存器)
 - IIC10 通信 ... SIO10 (IIC10 数据寄存器)

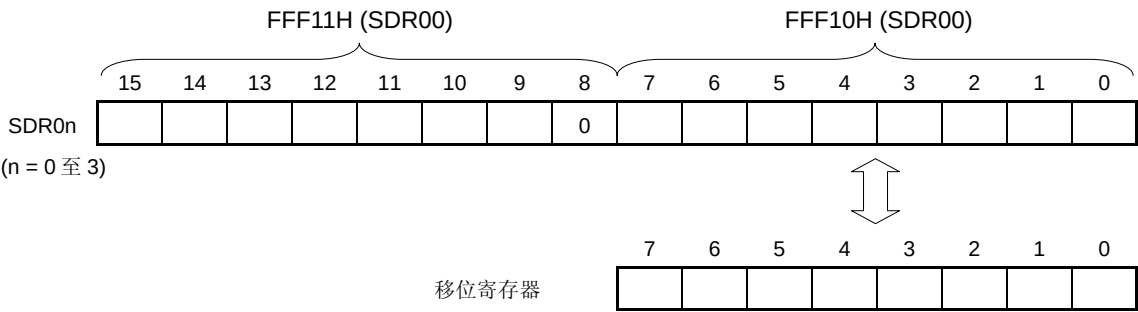
产生的复位信号将该寄存器清除为 0000H。
- 注

操作停止（SE0n = 0）时，禁止按 8 位为单位的写入。

- 备注
1. 接收到数据后，将“0”存储超出数据长度的位的位 0 至位 7。
 2. n: 通道编号 (n = 0 至 3),
p: CSI 编号 (p = 00、01、10),
q: UART 编号 (q = 0、1)

图 12-2. 串行数据寄存器 0n（SDR0n）的格式

地址： FFF10H、FFF11H (SDR00)、FFF12H、FFF13H (SDR01)、 复位后： 0000H R/W
 FFF44H、FFF45H (SDR02)、FFF46H、FFF47H (SDR03)



注意事项 务必将位 8 清除“0”。

- 备注
- 1. 有关 SDR0n 高 7 位的功能，参见 12.3 控制串行阵列部件的寄存器。
 - 2. n: 通道编号 (n = 0 至 3),

12.3 控制串行阵列部件的寄存器

串行阵列部件由下列寄存器控制。

- 外围使能寄存器 0 (PER0)
- 串行时钟选择寄存器 0 (SPS0)
- 串行模式寄存器 0n (SMR0n)
- 串行通信操作设置寄存器 0n (SCR0n)
- 串行数据寄存器 0n (SDR0n)
- 串行状态寄存器 0n (SSR0n)
- 串行标志清除触发器寄存器 0n (SIR0n)
- 串行通道使能状态寄存器 0 (SE0)
- 串行通道开始寄存器 0 (SS0)
- 串行通道停止寄存器 0 (ST0)
- 串行输出使能寄存器 0 (SOE0)
- 串行输出等级寄存器 0 (SOL0)
- 串行输出寄存器 0 (SO0)
- 输入开关控制寄存器 (ISC)
- 噪声滤波器使能寄存器 0 (NFEN0)
- 端口输入模式寄存器 3、7 (PIM3、PIM7)
- 端口输出模式寄存器 3、7 (POM3、POM7)
- 端口模式寄存器 3、7 (PM3、PM7)
- 端口寄存器 3、7 (P3、P7)

备注 n: 通道编号 (n = 0 至 3)

(1) 外围使能寄存器 0 (PER0)

PER0 用于启动或禁止各外围硬件宏功能的使用。供应未使用的硬件宏功能的时钟会停止工作，以便于减少功耗和噪声。

当使用串行阵列部件时，要确保将该寄存器的位 2 (SAU0EN) 设置为 1。

可以由 1 位或 8 位存储器处理指令设置 PER0。

产生的复位信号将该寄存器清除为 00H。

图 12-3. 外围使能寄存器 0 (PER0) 的格式

地址: F00F0H 复位后: 00H R/W

符号	<7>	6	<5>	<4>	3	<2>	1	0
PER0	RTCEN	0	ADCEN	IICAEN	0	SAU0EN	0	0

SAU0EN	串行阵列部件输入时钟的供应控制
0	停止输入时钟供应。 - 不能写入由串行阵列部件使用的SFR。 - 串行阵列部件处于复位状态。
1	供应输入时钟。 不能读取/写入由串行阵列部件使用的SFR。

- 注意事项
1. 设置串行阵列部件时，确保首先将 SAU0EN 置为 1。如果 SAU0EN = 0，则忽略对串行阵列部件的控制寄存器的写操作，即使读取寄存器，仅能读取默认值（输入开关控制寄存器（ISC）、噪声滤波器使能寄存器（NFEN0）、端口输入模式寄存器（PIM3、PIM7）、端口输出模式寄存器（POM3、POM7）、端口模式寄存器（PM3、PM7）和端口寄存器（P3、P7）除外）。
 2. 将 PER0 寄存器置为 1 后，经过 4 个或更多个时钟后务必设置 SPS0 寄存器。
 3. 务必将 PER0 寄存器的位 0、1、3 和 6（78K0R/KC3-L 的 44-引脚产品的位 0、1、3、4 和 6）置为 0。

(2) 串行时钟选择寄存器 0 (SPS0)

SPS0 是一款 16 位寄存器，用于选择通常供应各个通道的工作时钟（CK00、CK01）的两种类型。CK01 由 SPS0 的位 7 至位 4 选择，CK00 由位 3 至位 0 选择。

当寄存器处于工作中（SE0n = 1 时）时禁止重写 SPS0。

可以由 16 位存储器处理指令设置 SPS0。

可以使用 SPS0L 的 8 位存储器处理指令设置 SPS0 的低 8 位。

产生的复位信号将该寄存器清除为 0000H。

图 12-4. 串行时钟选择寄存器 0 (SPS0) 的格式

地址: F0126H、F0127H (SPS0)、F0166H、F0167H (SPS1) 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SPS0	0	0	0	0	0	0	0	0	PRS 013	PRS 012	PRS 011	PRS 010	PRS 003	PRS 002	PRS 001	PRS 000

PRS Op3	PRS Op2	PRS Op1	PRS Op0		操作时钟的选择 (CKOp) ^{#1}			
					f _{CLK} = 2 MHz	f _{CLK} = 5 MHz	f _{CLK} = 10 MHz	f _{CLK} = 20 MHz
0	0	0	0	f _{CLK}	2 MHz	5 MHz	10 MHz	20 MHz
0	0	0	1	f _{CLK} /2	1 MHz	2.5 MHz	5 MHz	10 MHz
0	0	1	0	f _{CLK} /2 ²	500 kHz	1.25 MHz	2.5 MHz	5 MHz
0	0	1	1	f _{CLK} /2 ³	250 kHz	625 kHz	1.25 MHz	2.5 MHz
0	1	0	0	f _{CLK} /2 ⁴	125 kHz	313 kHz	625 kHz	1.25 MHz
0	1	0	1	f _{CLK} /2 ⁵	62.5 kHz	156 kHz	313 kHz	625 kHz
0	1	1	0	f _{CLK} /2 ⁶	31.3 kHz	78.1 kHz	156 kHz	313 kHz
0	1	1	1	f _{CLK} /2 ⁷	15.6 kHz	39.1 kHz	78.1 kHz	156 kHz
1	0	0	0	f _{CLK} /2 ⁸	7.81 kHz	19.5 kHz	39.1 kHz	78.1 kHz
1	0	0	1	f _{CLK} /2 ⁹	3.91 kHz	9.77 kHz	19.5 kHz	39.1 kHz
1	0	1	0	f _{CLK} /2 ¹⁰	1.95 kHz	4.88 kHz	9.77 kHz	19.5 kHz
1	0	1	1	f _{CLK} /2 ¹¹	977 Hz	2.44 kHz	4.88 kHz	9.77 kHz
1	1	1	1	INTTM02 ^{#2}				
其它				禁止设置				

- 注 1.** 当改变为 f_{CLK} 所选的时钟（通过改变系统时钟控制寄存器（CKC）的值）时，在停止（ST0 = 000FH）串行阵列部件（SAU）的操作之后进行。为操作时钟选择了 INTTM02 时，也会停止定时器阵列部件（TT0 = 00FFH）。
- 2.** 不管 f_{CLK} 频率（主系统时钟、副系统时钟）如何，通过将 TAUS 的 TISO 寄存器的 TISO2 位设置为 1，并为输入时钟选择 f_{SUB}/4 和使用 SPS0 寄存器选择 INTTM02，在副系统时钟的固定分频比处操作 SAU。但是，更改 f_{CLK} 时，必须停止在上面注 1 中描述的 SAU 和 TAU。

- 注意事项 1.** 务必将位 15 至位 8 清除为“0”。
- 2.** 将 PER0 寄存器置为 1 后，经过 4 个或更多个时钟后务必设置 SPS0 寄存器。

- 备注 1.** f_{CLK}: CPU/外围硬件时钟频率
f_{SUB}: 副系统时钟频率
- 2.** p = 0、1

(3) 串行模式寄存器 0n (SMR0n)

SMR0n是设置通道n操作模式的寄存器。它还用于选择操作时钟（MCK）、指定是否输入串行时钟（SCK）、设置开始触发器、操作模式（CSI、UART或I²C）和中断源。仅在UART模式中，该寄存器还能使接收数据的电平反相。

当寄存器处于工作中（SE0n = 1 时）时禁止重写 SMR0n。然而，在操作期间，可以重写 MD0n0 位。

可以由 16 位存储器处理指令设置 SMR0n。

产生的复位信号将该寄存器设置为 0020H。

图 12-5. 串行模式寄存器 0n (SMR0n) 的格式 (1/2)

地址: F0110H、F0111H (SMR00) 至 F0116H、F0117H (SMR03) 复位后: 0020H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SMR0n	CKS 0n	CCS 0n	0	0	0	0	0	STS 0n	0	SIS 0n0	1	0	0	MD 0n2	MD 0n1	MD 0n0

CKS 0n	通道n的操作时钟(MCK)的选择
0	由PRS寄存器设置的预分频器输出时钟CK00
1	由PRS寄存器设置的预分频器输出时钟CK01
由边沿检测器使用操作时钟MCK。 此外，根据SDR0n寄存器CCS0n位和高7位的设置，会产生传输时钟(TCLK)。	

CCS 0n	通道n的传输时钟(TCLK)的选择
0	由CKS0n位指定的分频操作时钟MCK
1	自SCK引脚的时钟输入(CSI模式中的从机传输)
传输时钟TCLK用于移位寄存器、通信控制器、输出控制器、中断控制器和错误控制器。 当 CCS0n = 0时，由 SDR0n寄存器的高7位的设置MCK的分频比。	

STS 0n	开始触发源的选择
0	仅软件触发有效(选用于CSI、UART通信和简易的I ² C)。
1	RxD引脚的有效沿(选用于UART接收)
在将1设置给SS0寄存器后，满足上述触发源时，开始传输。	

注意事项 务必要将位 13 至 9、7、4 和 3 清除“0”。务必将位 5 设置为“1”。

备注 n: 通道编号 (n = 0 至 3)

图 12-5. 串行模式寄存器 0n (SMR0n) 的格式 (2/2)

地址: F0110H、F0111H (SMR00) 至 F0116H、F0117H (SMR03) 复位后: 0020H R/W

符号

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SMR0n	CKS 0n	CCS 0n	0	0	0	0	0	STS 0n	0	SIS 0n0	1	0	0	MD 0n2	MD 0n1	MD 0n0

SIS 0n0	UART模式中通道n接收数据电平的反相控制															
0	将检测到的下降沿作为起始位。 捕获输入通信数据的原值。															
1	将检测到的上升沿作为起始位。 使输入通信数据反相并捕获其值。															

MD 0n2	MD 0n1	通道n操作模式的选择														
0	0	CSI 模式														
0	1	UART 模式														
1	0	简易的 I ² C模式														
1	1	禁止设置														

MD 0n0	通道n中断源的选择															
0	传输结束中断															
1	缓冲器空闲中断															
对于连续发送，当SDR0n数据完成时通过将MD0n0置为1写入下一次发送的数据。																

备注 n: 通道编号 (n = 0 至 3)

(4) 串行通信操作设置寄存器 0n (SCR0n)

SCR0n 是通道 n 的通信操作设置寄存器。它用于设置数据发送/接收模式、数据和时钟的相位、是否屏蔽错误信号、奇偶位、开始位和数据长度。

当寄存器处于工作中 (SE0n = 1 时) 时禁止重写 SCR0n。

可以由 16 位存储器处理指令设置 SCR0n。

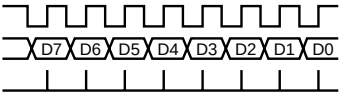
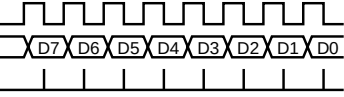
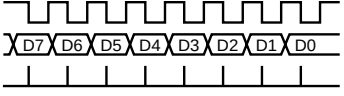
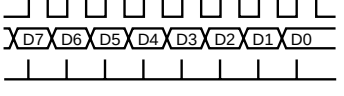
产生的复位信号将该寄存器置为 0087H。

图 12-6. 串行通信操作设置寄存器 0n (SCR0n) 的格式 (1/3)

地址: F0118H、F0119H (SCR00) 至 F011EH、F011FH (SCR03) 复位后: 0087H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SCR0n	TXE 0n	RXE 0n	DAP 0n	CKP 0n	0	EOC 0n	PTC 0n1	PTC 0n0	DIR 0n	0	SLC 0n1	SLC 0n0	0	DLS 0n2	DLS 0n1	DLS 0n0

TXE 0n	RXE 0n	通道n的操作模式设置
0	0	通信不开始.
0	1	仅接收
1	0	仅发送
1	1	发送/接收

DAP 0n	CKP 0n	CSI模式中数据和时钟相位的选择
0	0	SCKp SOp Slp 输入时序 
0	1	SCKp SOp Slp 输入时序 
1	0	SCKp SOp Slp 输入时序 
1	1	SCKp SOp Slp 输入时序 
在UART模式和简易的I²C模式中, 务必设置DAP0n、CKP0n = 0、0。		

注意事项 务必要将位 3、6 和 11 清除为“0”。务必将位 2 设置为“1”。

备注 n: 通道编号 (n = 0 至 3),
p: CSI 编号 (p = 00、01、10)

图 12-6. 串行通信操作设置寄存器 0n (SCR0n) 的格式 (2/3)

地址: F0118H、F0119H (SCR00) 至 F011EH、F011FH (SCR03) 复位后: 0087H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SCR0n	TXE 0n	RXE 0n	DAP 0n	CKP 0n	0	EOC 0n	PTC 0n1	PTC 0n0	DIR 0n	0	SLC 0n1	SLC 0n0	0	DLS 0n2	DLS 0n1	DLS 0n0

EOC0n	错误中断信号(INTSREx (x = 0、1))的屏蔽选择
0	屏蔽错误中断INTSREx (不屏蔽INTSRx)。
1	允许产生错误中断INTSREx (如果发生错位，屏蔽INTSRx)。
在CSI模式、简易的I ² C模式和UART发送期间，设置EOC0n = 0。 在UART发送期间，设置EOC0n = 1。	

PTC 0n1	PTC 0n0	UART模式中的校验位设置	
		发送	接收
0	0	不输出校验位。	无校验接收
0	1	输出0校验。	无校验位判断
1	0	输出偶校验。	判断为偶校验。
1	1	输出奇校验。	判断为奇校验。
在CSI模式和简易的I ² C模式中，务必设置PTC0n1、PTC0n0 = 0、0。			

DIR0n	CSI和UART模式中数据传输次序的选择
0	使用MSB先行输入/输出数据。
1	使用LSB先行输入/输出数据。
在简易的 I ² C模式中, 务必清除DIR0n, 即DIR0n = 0。	

SLC 0n1	SLC 0n0	UART模式中的终止位设置
0	0	无终止位
0	1	终止位长度 = 1 位
1	0	终止位长度= 2位
1	1	禁止设置
当选中传输结束中断时，在所有终止位全部完成传输时产生中断。 UART接收期间和在简化的I²C模式中，设置1位（SLC0n1、SLC0n0 = 0、1）。 在CSI模式中设置无终止位（SLC0n1、SLC0n0 = 0、0）。		

注意事项 务必要将位 3、6 和 11 清除为“0”。务必将位 2 设置为“1”。**备注** n: 通道编号 (n = 0 至 3)

图 12-6. 串行通信操作设置寄存器 0n (SCR0n) 的格式 (3/3)

地址: F0118H、F0119H (SCR00) 至 F011EH、F011FH (SCR03) 复位后: 0087H R/W

符号

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SCR0n	TXE 0n	RXE 0n	DAP 0n	CKP 0n	0	EOC 0n	PTC 0n1	PTC 0n0	DIR 0n	0	SLC 0n1	SLC 0n0	0	DLS 0n2	DLS 0n1	DLS 0n0

DLS 0n2	DLS 0n1	DLS 0n0	CSI和UART模式中的数据长度设置													
1	0	0	5位数据长度(存储在SDR0n寄存器的位0至位4) (仅能在UART模式中设置)													
1	1	0	7位数据长度(存储在SDR0n寄存器的位0至位6)													
1	1	1	8位数据长度(存储在SDR0n寄存器的位0至位7)													
其它			禁止设置													
在简易的I ² C模式中，务必设置DLS0n0 = 1。																

注意事项 务必要将位 3、6 和 11 清除为“0”。务必将位 2 设置为“1”。

备注 n: 通道编号 (n = 0 至 3)

(5) 串行数据寄存器 0n (SDR0n) 的高 7 位

SDR0n 是通道 n 的发送/接收数据寄存器（16 位）。位 7 至位 0 作为发送/接收缓冲寄存器使用，位 15 至位 9 作为设置操作时钟（MCK）分频比的寄存器使用。

如果将串行模式寄存器 0n（SMR0n）的 CCS0n 位清除位 0，通过将操作时钟分频，再由 SDR0n 的高 7 位设置的时钟用作传输时钟。

有关 SDR0n 低 8 位的功能，参见 12.2 串行阵列部件的配置。

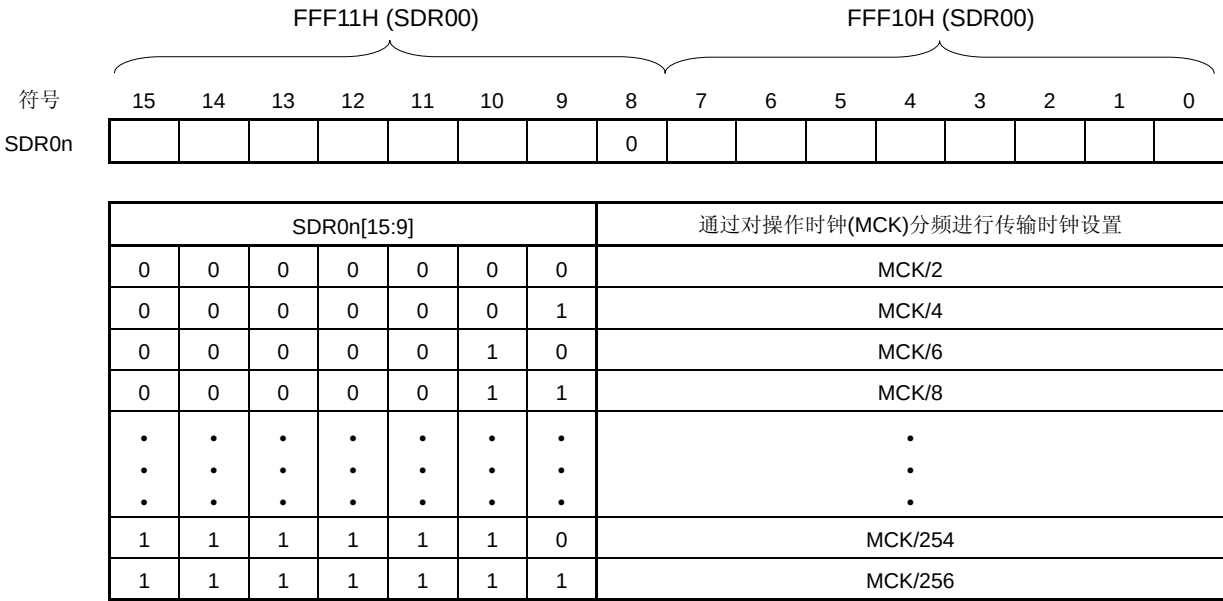
SDR0n 可按 16 位为单位进行读写。

但是，当操作停止（SE0n = 0）时仅能对高 7 位进行读或写操作。操作期间（SE0n = 1），仅将值写入 SDR0n 的低 8 位。操作期间读取 SDR0n 时，总是读取 0。

产生的复位信号将该寄存器清除为 0000H。

图 12-7. 串行数据寄存器 0n (SDR0n) 的格式

地址: FFF10H、FFF11H (SDR00)、FFF12H、FFF13H (SDR01)、 复位后: 0000H R/W
 FFF44H、FFF45H (SDR02)、FFF46H、FFF47H (SDR03)



- 注意事项
1. 务必将位 8 清除为“0”。

2. 使用 UART 时，禁止设置 SDR0n[15:9] = (0000000B、0000001B)。

- 备注
1. 有关 SDR0n 低 8 位的功能，参见 12.2 串行阵列部件的配置。

2. n: 通道编号 (n = 0 至 3)

(6) 串行状态寄存器 0n (SSR0n)

SSR0n 是指通道 n 的通信状态和错误发生状态的寄存器。由寄存器指示的错误是指：帧错误、奇偶校验错误和超限错误。

可以由 16 位存储器处理指令读取 SSR0n。

可以使用 SSR0nL 的 8 位存储器处理指令设置 SSR0n 的低 8 位。

产生的复位信号将该寄存器清除为 0000H。

图 12-8. 串行状态寄存器 0n (SSR0n) 的格式 (1/2)

地址： F0100H、F0101H (SSR00) 至 F0106H、F0107H (SSR03) 复位后： 0000H R

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSR0n	0	0	0	0	0	0	0	0	0	TSF 0n	BFF 0n	0	0	FEF 0n	PEF 0n	OVF 0n

TSF 0n	通道n的通信状态指示标志
0	未执行通信。
1	正在执行通信。
由于该标志为更新标志，所以当通信操作完成时它会自动清除。当将ST0n/SS0n位设置为1时也会清除该标志。	

BFF 0n	通道n的缓冲器寄存器状态指示标志
0	有效数据不存储在SDR0n寄存器中。
1	有效数据存储在SDR0n寄存器中。
这是一个更新标志。 当从SDR0n寄存器到移位寄存器的传输完成时它会自动清除。 接收期间，当已经从SDR0n寄存器读取数据时它会自动清除。 当将ST0n/SS0n 位设置为1时也会清除该标志。 当SCR0n寄存器的TXE0n位= 1（各通信模式中的发送或接收模式）时，如果将发送数据写入SDR0n寄存器，则会自动设置该标志。 当SCR0n寄存器的RXE0n位= 1（各通信模式中的发送或接收模式）时，如果将接收数据存储在SDR0n寄存器，则会自动设置该标志。 在接收错误的情况下也会对其进行设置。 如果在BFF0n = 1时将数据写入SDR0n寄存器，则丢弃存储在寄存器中的发送/接收数据并检测到超限错误（OVF0n = 1）。	

备注 n: 通道编号 (n = 0 至 3)

图 12-8. 串行状态寄存器 0n (SSR0n) 的格式 (2/2)

地址: F0100H、F0101H (SSR00) 至 F0106H、F0107H (SSR03) 复位后: 0000H R

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSR0n	0	0	0	0	0	0	0	0	0	TSF 0n	BFF 0n	0	0	FEF 0n	PEF 0n	OVF 0n

FEF 0n	通道n的帧错误检测标志
0	未发生错误。
1	UART接收期间发生帧错误。 <帧错误原因> 在UART接收完成时，如果没有检测到终止位，会发生帧错误。
这是一个累积标志，直到将1写入SIR0n寄存器的FECT0n位，才能将其清除。	

PEF 0n	通道n的奇偶校验错误检测
0	未发生错误。
1	在UART接收期间或者在I ² C 传输期间未检测到ACK时发生奇偶校验错误。 <奇偶校验错误原因> - 如果发送数据的奇偶校验位与UART接收完成时的奇偶位不相符，则会发生奇偶校验错误。 - 如果在I²C传输期间ACK信号没有从ACK接收时序中的从机中返回，则不会检测到ACK。
这是一个累积标志，直到将1写入SIR0n寄存器的PECT0n位，才能将其清除。	

OVF 0n	通道n的超限错误检测标志
0	未检测到错误。
1	发生超限错误。 <超限错误的原因> - 未读取存储在SDR0n寄存器中的接收数据，而是写入发送数据或将下一个接收数据写入。 - 在CSI模式中发送数据没有为从机发送或接收准备就绪。
这是一个累积标志，直到将1写入SIR0n寄存器的OVCT0n位，才能将其清除。	

备注 n: 通道编号 (n = 0 至 3)

(7) 串行标志清除触发器寄存器 0n (SIR0n)

SIR0n 是一款触发寄存器，用于清除通道 n 的各个错误标志。
当将该寄存器的各位 (FECT0n、PECT0n、OVCT0n) 置为 1 时，串行状态寄存器 0n 的对应位 (FEF0n、PEF0n、OVF0n) 清除为 0。由于 SIR0n 是触发寄存器，因此当清零 SSR0n 的对应位时会立即将其清除。
可以由 16 位存储器处理指令设置 SIR0n。
可以由 SIR0nL 的 8 位存储器处理指令设置 SIR0n 的低 8 位。
产生的复位信号将该寄存器清除为 0000H。

图 12-9. 串行标志清除触发器寄存器 0n (SIR0n) 的格式

地址: F0108H、F0109H (SIR00) 至 F010EH、F010FH (SIR03) 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SIR0n	0	0	0	0	0	0	0	0	0	0	0	0	0	FEC T0n	PEC T0n	OVC T0n

FEC T0n	通道n的帧错误标志清除触发器															
0	没有触发器操作															
1	将SSR0n寄存器的FEF0n位清除为0。															

PEC T0n	通道n的奇偶校验错位标志清除触发器															
0	没有触发器操作															
1	将SSR0n寄存器的PEF0n位清除为0。															

OVC T0n	通道n的超限错误标志清除触发器															
0	没有触发器操作															
1	将SSR0n寄存器的OVF0n位清除为0。															

注意事项 务必将位 15 至位 3 清除为“0”。

- 备注
1. n: 通道编号 (n = 0 至 3)
 2. 读取 SIR0n 寄存器时，总是读取到 0000H。

(8) 串行通道使能状态寄存器 0 (SE0)

SE0 指示各个通道的数据发送/接收操作是允许还是禁止。

当将 1 写入串行通道开始寄存器 0 (SS0) 中的一位时，该寄存器的对应位被置为 1。当将 1 写入串行通道停止寄存器 0 (ST0) 中的一位时，对应位清除为 0。

允许运行的通道 n 不能通过软件重写下文描述的串行输出寄存器 0 (SO0) 的 CKO0n 值，由通信操作反映的值会从串行时钟引脚输出。

停止运行的通道 n 可以通过软件设置 SO0 寄存器的 CKO0n 值，并从串行时钟引脚输出其值。用这种方式，诸如开始条件/停止条件之类的任何波形都可以由软件创建。

可以由 16 位存储器处理指令读取 SE0。

可以由 SE0L 的 1 位或 8 位存储器处理指令设置 SE0 的低 8 位。

产生的复位信号将该寄存器清除为 0000H。

图 12-10. 串行通道使能状态寄存器 0 (SE0) 的格式

地址: F0120H、F0121H 复位后: 0000H R

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SE0	0	0	0	0	0	0	0	0	0	0	0	0	SE0 3	SE0 2	SE0 1	SE0 0

SE0 n	通道n的启动/停止状态的操作指示
0	操作停止(因控制寄存器和移位寄存器的值，串行时钟I/O引脚、串行数据输出引脚的状态和保留的 [✱] PEF、PEF及OVF的错误标志而停止)。
1	启动操作。

注 SSR0n 寄存器的位 6 和位 5 (TSF0n、BFF0n) 被清除。

备注 n: 通道编号 (n = 0 至 3)

(9) 串行通道开始寄存器 0 (SS0)

SS0 是一个触发器寄存器，用于启动各通道控制的开始通信/开始计数操作。

当将 1 写入该寄存器 (SS0n) 中的一位时，串行通道使能状态寄存器 0 (SE0) 的对应位 (SE0n) 设置为 1。

由于 SS0n 是触发位，因此当 SE0n = 1 时会被立即清零。

可以由 16 位存储器处理指令设置 SS0。

可以由 SS0L 的 1 位或 8 位存储器处理指令设置 SS0 的低 8 位。

产生的复位信号将该寄存器清除为 0000H。

图 12-11. 串行同代开始寄存器 0 (SS0) 的格式

地址: F0122H、F0123H 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SS0	0	0	0	0	0	0	0	0	0	0	0	0	SS03	SS02	SS01	SS00

SS0n	通道n的开始触发器操作
0	没有触发器操作
1	将SE0n置为1，并进入通信等待状态(如果一个通信操作已在执行中，则停止操作并等候开始条件)。

注意事项 务必将位 15 至位 4 清除为“0”。

- 备注
1. n: 通道编号 (n = 0 至 3)

2. 读取 SS0 寄存器时，总是读取到 0000H。

(10) 串行通道停止寄存器 0 (ST0)

ST0 是一个触发器寄存器，用于启动各通道控制的停止通信/停止计数操作。

当将 1 写入该寄存器 (ST0n) 中的一位时，串行通道使能状态寄存器 0 (SE0) 的对应位 (SE0n) 清除为 0。

由于 ST0n 是触发位，因此当 SE0n = 0 时会被立即清零。

由 16 位存储器处理指令写入可以设置 ST0。

可以由 ST0L 的 1 位或 8 位存储器处理指令设置 ST0 的低 8 位。

产生的复位信号将该寄存器清除为 0000H。

图 12-12. 串行通道停止寄存器 0 (ST0) 的格式

地址: F0124H、F0125H 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ST0	0	0	0	0	0	0	0	0	0	0	0	0	ST0 3	ST0 2	ST0 1	ST0 0

ST0n	通道n的停止触发器的操作
0	没有触发器操作
1	将SE0n清除为0，并停止通信操作。 (因控制寄存器和移位寄存器的值，串行时钟I/O引脚、串行数据输出引脚的状态，和保留的 ^注 FEF、PEF及OVF的错误标志)。

注 SSR0n 寄存器的位 6 和位 5 (TSF0n、BFF0n) 被清除。

注意事项 务必将位 15 至位 4 清除为“0”。

- 备注 1. n: 通道编号 (n = 0 至 3)
2. 读取 ST0 寄存器时，总是读取到 0000H。

(11) 串行输出使能寄存器 0（SOE0）

SOE0 是一款用于启动或停止各通道的串行通信操作输出的寄存器。

启动串行输出的通道 n 不能通过软件重写下文中描述的串行输出寄存器 0（SO0）的 SO0n 值，而由通信操作反映的值会从串行数据输出引脚输出。

对于停止串行输出的通道 n，可以由软件设置 SO0 寄存器的 SO0n 值，且可从串行数据输出引脚输出该值。用这种方式，开始条件/停止条件的任何波形都可以由软件创建。

可以由 16 位存储器处理指令设置 SOE0。

可以由 SOE0L 的 1 位或 8 位存储器处理指令设置 SOE0 的低 8 位。

产生的复位信号将该寄存器清除为 0000H。

图 12-13. 串行输出使能寄存器 0（SOE0）的格式

地址： F012AH、F012BH 复位后： 0000H R/W																
符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOE0	0	0	0	0	0	0	0	0	0	0	0	0	0	SOE02	SOE01	SOE00

SOE0n	通道n的串行输出允许/禁止														
0	由串行通信操作停止输出。														
1	由串行通信操作允许输出。														

注意事项 务必将位 15 至位 3 清除为“0”。

备注 n： 通道编号 (n = 0 至 2)

(12) 串行输出寄存器 0 (SO0)

SO0 是各个通道串行输出的缓冲器寄存器。

该寄存器位 n 的值从通道 n 的串行数据输出引脚输出。

该寄存器位 (n + 8) 的值从通道 n 的串行时钟输出引脚输出。

仅当禁止串行输出 (SOE0n = 0) 时，该寄存器的 SO0n 才可以由软件重写。当允许串行输出 (SOE0n = 1) 时，忽略由软件进行的重写，且仅能由串行通信操作对寄存器的值进行更改。

仅当通道操作停止 (SE0n = 0) 时，该寄存器的 CKO0n 可以由软件进行重写。当允许通道操作 (SE0n = 1) 时，忽略由软件进行的重写，且仅能由串行通信操作对 CKO0n 的值进行更改。

当使用 P30/SO10/TxD1、P31/SI10/RxD1/SDA10/INTP1、P32/ $\overline{\text{SCK10}}$ /SCL10/INTP2、P70/KR0/SO01/INTP4、P72/KR2/ $\overline{\text{SCK01}}$ /INTP6、P73/KR3/SO00/TxD0 或 P75/KR5/ $\overline{\text{SCK00}}$ 引脚作为端口功能引脚时，将对应的 CKO0n 和 SO0n 位设置为“1”。

可以由 16 位存储器处理指令设置 SO0。

产生的复位信号将该寄存器清除为 0F0FH。

图 12-14. 串行输出寄存器 0 (SO0) 的格式

地址: F0128H、F0129H 复位后: 0F0FH R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SO0	0	0	0	0	1	CKO02	CKO01	CKO00	0	0	0	0	1	SO02	SO01	SO00

CKO0n	通道n的串行时钟输出
0	串行时钟输出值为“0”。
1	串行时钟输出值为“1”。

SO0n	通道n的串行数据输出
0	串行数据输出值为“0”。
1	串行数据输出值为“1”。

注意事项 务必将位 11 和位 3 设置“1”。

务必要将位 15 至 12 和位 7 至 4 清除为“0”。

备注 n: 通道编号 (n = 0 至 2)

(13) 串行输出电平寄存器 0 (SOL0)

SOL0 是一款用于设置各通道数据输出电平反相的寄存器。

仅能在UART模式中设置该寄存器。务必在CSI模式和简易的I²C模式中设置为 0000H。

仅当允许串行输出 (SOE0n = 1) 时，使用该寄存器进行反相的通道 n 反映在引脚输出处。当禁止串行输出 (SOE0n = 0) 时，SO0n 位输出其本身值。

当寄存器处于工作中 (SE0n = 1 时) 时禁止重写 SOL0。

可以由 16 位存储器处理指令设置 SOL0。

可以由 SOL0L 的 8 位存储器处理指令设置 SOL0 的低 8 位。

产生的复位信号将该寄存器清除为 0000H。

图 12-15. 串行输出电平寄存器 0 (SOL0) 的格式

地址: F0134H, F0135H 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOL0	0	0	0	0	0	0	0	0	0	0	0	0	0	SOL02	0	SOL00

SOL0n	在UART模式中选择通道n发送数据电平的反相
0	通信数据输出其本身。
1	通信数据经反相并输出。

注意事项 务必将位 15 至位 3 和位 1 清除为“0”。

备注 n: 通道编号 (n = 0、2)

(14) 输入开关控制寄存器 (ISC)

ISC 用于实现 UART0 的 LIN 总线通信操作与外部中断和定时器阵列部件 TAUS 的协调。

当将位 0 置为 1 时，串行数据输入 (RxD0) 引脚的输入信号选作为可用于检测唤醒信号的外部中断 (INTP0)。

当将位 1 置为 1 时，串行数据输入 (RxD0) 引脚的输入信号选作为定时器输入，以便于由定时器测量同步中断域和同步域的脉冲宽度。

可以由 1 位或 8 位存储器处理指令设置 ISC。

产生的复位信号将该寄存器清除为 00H。

图 12-16. 输入开关控制寄存器 (ISC) 的格式

地址: FFF3CH 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
ISC	0	0	0	0	0	ISC2	ISC1	ISC0

ISC1	定时器阵列部件TAUS的通道7的输入转换
0	将TI07引脚的 输入信号用作定时器输入(正常模式)。
1	将RxD0引脚的输入信号用作定时器输入(唤醒信号检测)。

ISC0	外部中断(INTP0)的输入转换
0	将INTP0引脚的输入信号用作外部中断(正常模式)。
1	将RxD0引脚的输入信号用作外部中断 (要测量同步中断域和同步域的脉冲宽度)。

注意事项 务必将位 7 至位 3 清除为“0”。

(15) 噪声滤波器使能寄存器 0 (NFEN0)

NFEN0 用于设置噪声滤波器能否用作从串行数据输入引脚至各通道的输入信号。
通过将该寄存器的对应位清除为 0，使用作CSI或简易的I²C通信引脚的噪声滤波器失效。
通过将该寄存器的对应位设置为 1，使用作 UART 通信引脚的噪声滤波器有效。
当启动噪声滤波器时，CPU/外围硬件时钟（f_{CLK}）与 2-时钟匹配检测同步。
可以由 1 位或 8 位存储器处理指令设置 NFEN0。
产生的复位信号将该寄存器清除为 00H。

图 12-17. 噪声滤波器使能寄存器 0 (NFEN0) 的格式

地址: F0060H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
NFEN0	0	0	0	0	0	SNFEN10	0	SNFEN00

SNFEN10	RxD1/SDA10/SI10/INTP1/P31引脚的噪声滤波器的使用情况
0	噪声滤波器关闭
1	噪声滤波器打开
将SNFEN10置为1以使用RxD1引脚。 将SNFEN10清除为0以使用SDA10、SI10、INTP1和P31引脚。	

SNFEN00	RxD0/SI00/KR4/P74引脚的噪声滤波器的使用情况
0	噪声滤波器关闭
1	噪声滤波器打开
将SNFEN00置为1以使用RxD0引脚。 将SNFEN00清除为0以使用SI00、KR4和P74引脚。	

注意事项 务必将位 7 至位 3 和位 1 清除为“0”。

(16) 端口输入模式寄存器 3、7（PIM3、PIM7）

这些寄存器按位为单位设置端口 3 和 7 的输入缓冲器。

可以由 1 位或 8 位存储器处理指令设置 PIM3 和 PIM7。

产生的复位信号将这些寄存器清除为 00H。

图 12-18. 端口输入模式寄存器 3 和 7（PIM3 和 PIM7）的格式

地址: F0043H	复位后: 00H	R/W								
符号	7	6	5	4	3	2	1	0		
PIM3	0	0	0	0	0	PIM32	PIM31	0		

地址: F0047H	复位后: 00H	R/W								
符号	7	6	5	4	3	2	1	0		
PIM7	0	0	PIM75	PIM74	0	PIM72	PIM71	0		

PIMmn	Pmn引脚输入缓冲器选择(m = 3、7; n = 1、2、4、5)									
0	正常输入缓冲器									
1	TTL输入缓冲器									

(17) 端口输出模式寄存器 3、7（POM3、POM7）

这些寄存器按位为单位设置端口 3 和 7 的输出模式。

可以由 1 位或 8 位存储器处理指令设置 POM3 和 POM7。

产生的复位信号将这些寄存器清除为 00H。

图 12-19. 端口输出模式寄存器 3 和 7（POM3 和 POM7）的格式

地址: F0053H	复位后: 00H	R/W								
符号	7	6	5	4	3	2	1	0		
POM3	0	0	0	0	0	POM32	POM31	POM30		

地址: F0057H	复位后: 00H	R/W								
符号	7	6	5	4	3	2	1	0		
POM7	0	0	POM75	0	POM73	POM72	0	POM70		

POMmn	Pmn引脚输出缓冲器选择 (m = 3、7; n = 0 至 3、5)									
0	正常输出模式									
1	N沟道漏极开路输出(耐压V _{DD})模式									

(18) 端口模式寄存器 3、7 (PM3、PM7)

这些寄存器按位为单位设置端口 3 和 7 的输入/输出。

当在串行数据输出或串行时钟输出中使用 P30/SO10/TxD1、P31/SI10/RxD1/SDA10/INTP1、P32/SCK10/SCL10/INTP2、P70/KR0/SO01/INTP4、P72/KR2/SCK01/INTP6、P73/KR3/SO00/TxD0 和 P75/KR5/SCK00 引脚时，将 PM30 至 PM32、PM70、PM72、PM73 和 PM75 位清除为 0，并将 P30 至 P32、P70、P72、P73 和 P75 的输出锁存置为 1。

当在串行数据输入或串行时钟输入中使用 P31/SI10/RxD1/SDA10/INTP1、P32/SCK10/SCL10/INTP2、P71/KR1/SI01/INTP5、P72/KR2/SCK01/INTP6、P74/KR4/SI00/RxD0 和 P75/KR5/SCK00 引脚时，将位 PM31、PM32、PM71、PM72、PM74 和 PM75 置为 1。此时，P31、P32、P71、P72、P74 和 P75 的输出锁存值可能为 0 或者 1。

可以由 1 位或 8 位存储器处理指令设置 PM3 和 PM7。

产生的复位信号将这些寄存器置为 FFH。

图 12-20. 端口模式寄存器 3 和 7 (PM3 和 PM7) 的格式

地址: FFF23H 复位后: FFH R/W							
符号	7	6	5	4	3	2	1 0
PM3	1	1	1	1	1	PM32	PM31 PM30

地址: FFF27H 复位后: FFH R/W							
符号	7	6	5	4	3	2	1 0
PM7	1	1	PM75	PM74	PM73	PM72	PM71 PM70

PMmn	Pmn引脚I/O模式选择 (m = 3、7; n = 0 至 5)
0	输出模式(输出缓冲器打开)
1	输入模式(输出缓冲器关闭)

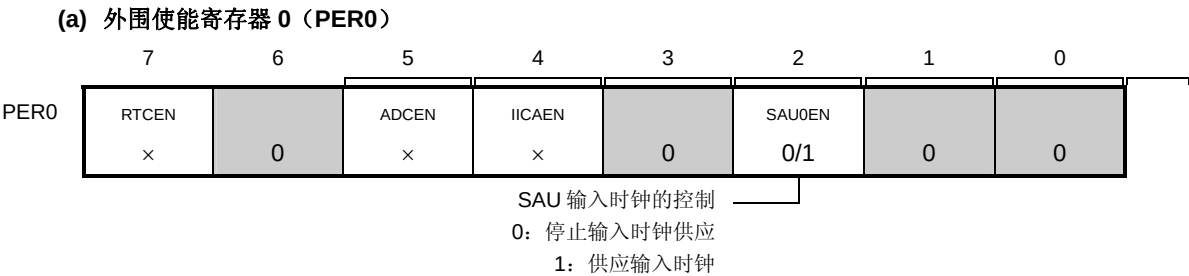
12.4 操作停止模式

串行阵列部件的各串行接口都具有操作停止模式。
在该模式中，不能执行串行通信，从而减少功耗。
此外，在该模式中，P30/SO10/TxD1、P31/SI10/RxD1/SDA10/INTP1、P32/SCK10/SCL10/INTP2、P70/KR0/SO01/INTP4、P71/KR1/SI01/INTP5、P72/KR2/SCK01/INTP6、P73/KR3/SO00/TxD0、P74/KR4/SI00/RxD0和P75/KR5/SCK00引脚可以用作端口功能引脚。

12.4.1 由部件停止操作

使用外围使能寄存器 0 (PER0) 设置由部件停止的操作。
PER0 用于启动或禁止各外围硬件宏功能的使用。供应未使用的硬件宏功能的时钟会停止工作，以便于减少功耗和噪声。
要停止串行阵列部件的操作，需将位 2 (SAU0EN) 置为 0。

图 12-21. 由部件停止操作时外围使能寄存器 0 (PER0) 的设置



- 注意事项
- 1. 如果 SAU0EN = 0，则忽略对串行阵列部件的控制寄存器的写操作，即使读取寄存器，仅能读取默认值（输入开关控制寄存器（ISC）、噪声滤波器使能寄存器（NFEN0）、端口输入模式寄存器（PIM3、PIM7）、端口输出模式寄存器（POM3、POM7）、端口模式寄存器（PM3、PM7）和端口寄存器（P3、P7）除外）。
 - 2. 务必要将位 6、3、1 和 0 清除为“0”。

备注

- ：禁止设置(由硬件固定)
- ×：未由串行真累部件使用的位(取决于其它外围功能的设置)
- 0/1：置为 0 或 1 取决于用户的用途

12.4.2 由通道停止操作

使用以下各寄存器设置由通道停止的操作。

图 12-22. 当由通道停止操作时的各寄存器设置（1/2）

(a) 串行通道使能状态寄存器 0 (SE0) ... 该寄存器指示各通道的数据发送/接收操作是启动还是停止。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
												SE03	SE02	SE01	SE00
												0/1	0/1	0/1	0/1

0: 操作停止

* SE0 寄存器是只读状态寄存器，使用 ST0 寄存器可停止其操作。
对于操作停止了通道，可以由软件设置 SO0 寄存器的 CKO0n 值。

(b) 串行通道停止寄存器 0 (ST0) ... 该寄存器是一个触发器寄存器，由各通道启动停止通信/停止计数。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
												ST03	ST02	ST01	ST00
												0/1	0/1	0/1	0/1

1: 将 SE0n 置为 0，并停止通信操作

* 由于 ST0n 是触发位，当 SE0n = 0 时立即将其清除。

(c) 串行输出使能寄存器 0 (SOE0) ... 该寄存器是用于启动或停止各通道串行通信操作输出的寄存器。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
												SOE02	SOE01	SOE00	
												0/1	0/1	0/1	

0: 由串行通信操作停止输出

* 对于串行输出停止了通道 n，可以由软件设置 SO0 寄存器的 SO0n 值。

(d) 串行输出寄存器 0 (SO0) ... 该寄存器是用于各通道串行输出的缓冲器寄存器。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
				CKO02	CKO01	CKO00					SO02	SO01	SO00		
				0/1	0/1	0/1					0/1	0/1	0/1		

1: 串行时钟输出值为“1” 1: 串行数据输出值为“1”

* 当将对各通道的引脚用作端口功能引脚时，将对应的 CKO0n 和 SO0n 位设置为“1”。

备注 n: 通道编号 (n = 0 至 3)
 ■: 禁止设置(由硬件固定), 0/1: 置为 0 或 1 取决于用户的用途

12.5 3-线串行 I/O（CSI00、CSI01、CSI10）通信的操作

这是使用三线的时钟通信功能，即：串行时钟（SCK）线和串行数据（SI 和 SO）线。

[数据发送/接收]

- 7 或 8 位的数据长度
- 发送/接收数据的相位控制
- 可选择的 MSB/LSB 先行
- 发送/接收数据的电平设置

[时钟控制]

- 主机/从机选择
- I/O 时钟的相位控制
- 经过各通道的预分频器和内部计数器的传输周期的设置
- 最大传输速率

主机通信期间：最大值为 $f_{CLK}/4$ ，从机通信期间：最大值为 $f_{MCK}/6$ [※]

[中断功能]

- 传输结束中断/缓冲器空闲中断

[错误检测标志]

- 超限错误

注 在满足 $\overline{SCK}$ 周期（ t_{KCY} ）特性的范围内使用时钟（参见第二十八章 电气规范（目标））。

支持 3-线串行 I/O（CSI00、CSI01、CSI10）的通道是指 SAU 的通道 0 至 2。

通道	用作CSI	用作UART	用作简易的I ² C
0	CSI00	UART0 (支持LIN总线)	—
1	CSI01		—
2	CSI10	UART1	IIC10
3	—		—

3-线串行 I/O（CSI00、CSI01、CIS10）执行以下六种通信操作。

- 主机发送 (参见 12.5.1.)
- 主机接收 (参见 12.5.2.)
- 主机发送/接收 (参见 12.5.3.)
- 从机发送 (参见 12.5.4.)
- 从机接收 (参见 12.5.5.)
- 从机发送/接收 (参见 12.5.6.)

12.5.1 主机发送

主机发送是指 78K0R/Kx3-L 输出一个传输时钟并将数据发送至其它设备。

3-线串行I/O	CSI00	CSI01	CSI10
目标通道	SAU的通道0	SAU的通道1	SAU的通道2
所用引脚	SCK00、SO00	SCK01、SO01	SCK10、SO10
中断	INTCSI00	INTCSI01	INTCSI10
	可以选择传输结束中断(在单次传输模式中)或缓冲器空闲中断(在连续传输模式中)。		
错误检测标志	无		
发送数据长度	7或8位		
传输速率	最大值 $f_{CLK}/4$ [MHz], 最小值 $f_{CLK}/(2 \times 2^{11} \times 128)$ [MHz] ^註 f_{CLK} : 系统时钟频率		
数据相位	可由DAP0n位选择 • DAP0n = 0: 始于串行时钟操作开始的数据输出。 • DAP0n = 1: 始于串行时钟操作开始前半个时钟的数据输出。		
时钟相位	可由CKP0n位选择 • CKP0n = 0: 正向 • CKP0n = 1: 反向		
数据方向	MSB或LSB先行		

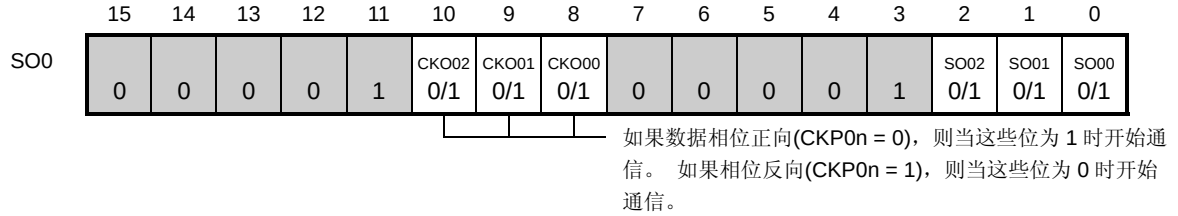
注 在满足以上条件和电气规范中的 AC 特性的范围内使用该操作（参见第二十八章 电气规范（目标））。

备注 n: 通道编号 (n = 0 至 2)

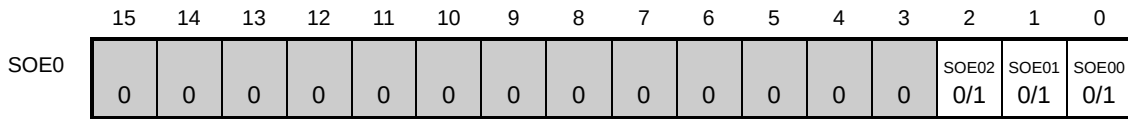
(1) 寄存器设置

图 12-23. 有关 3-线串行 I/O 主机发送寄存器的内容的示例
(CSI00、CSI01、CSI10)

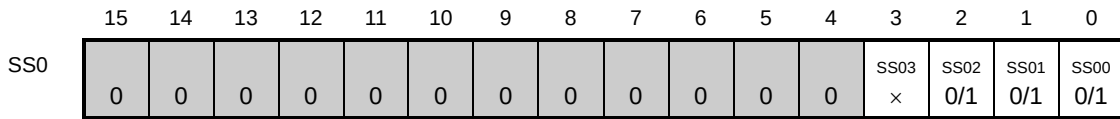
(a) 串行输出寄存器 0 (SO0) ... 仅设置目标通道的位。



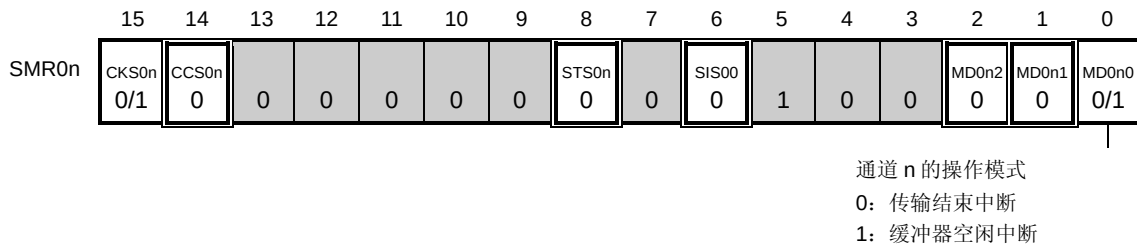
(b) 串行输出使能寄存器 0 (SOE0) ... 仅将目标通道的位设置为 1。



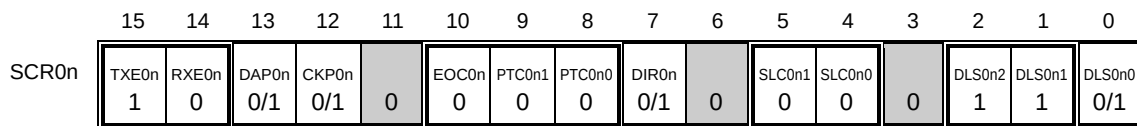
(c) 串行通道开始寄存器 0 (SS0) ... 仅将目标通道的位设置为 1。



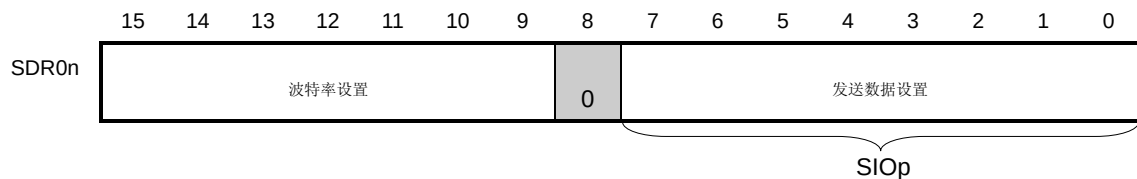
(d) 串行模式寄存器 0n (SMR0n)



(e) 串行通信操作设置寄存器 0n (SCR0n)



(f) 串行数据寄存器 0n (SDR0n) (低 8 位: SIOp)



备注

n: 通道编号 (n = 0 至 2)

p: CSI 编号 (p = 00、01、10)

□: 在 CSI 主机发送中设置固定,

■ 禁止设置 (设为初始值)

×: 在该模式中不能使用的位 (在任何模式中均不使用时设为初始值)

0/1: 置为 0 或 1 取决于用户的用途

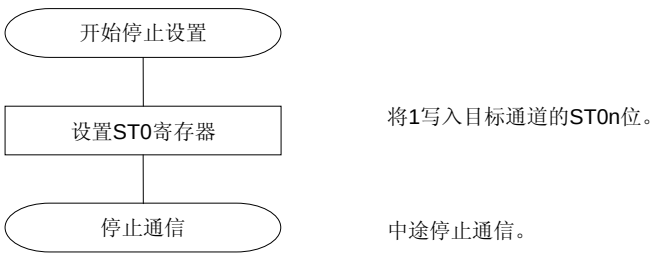
(2) 操作步骤

图 12-24. 主机发送的初始设置步骤



注意事项 将 PER0 寄存器置为 1 后，经过 4 个或更多个时钟后务必设置 SPS0 寄存器。

图 12-25. 停止主机发送的步骤



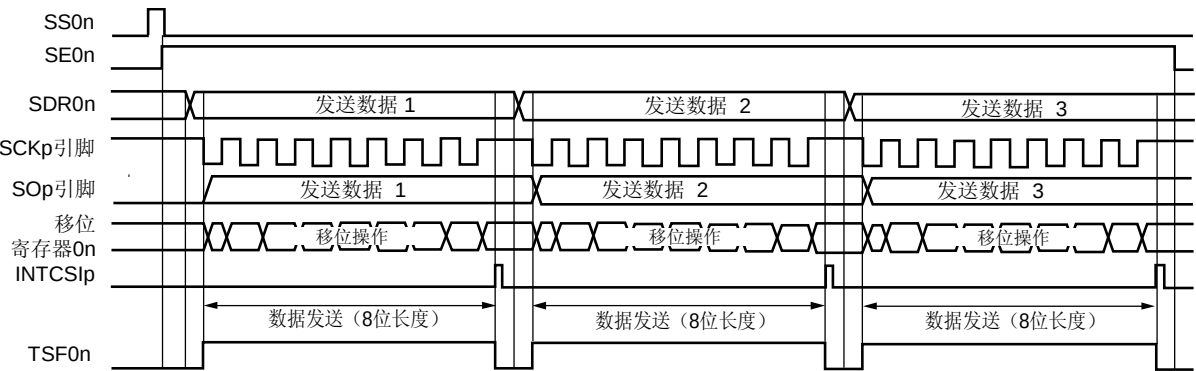
备注 即使在通信停止之后，仍保留引脚的电平。要恢复操作，需重新设置 SO0 寄存器（参见图 12-26 恢复主机通信的步骤）。

图 12-26. 恢复主机发送的步骤



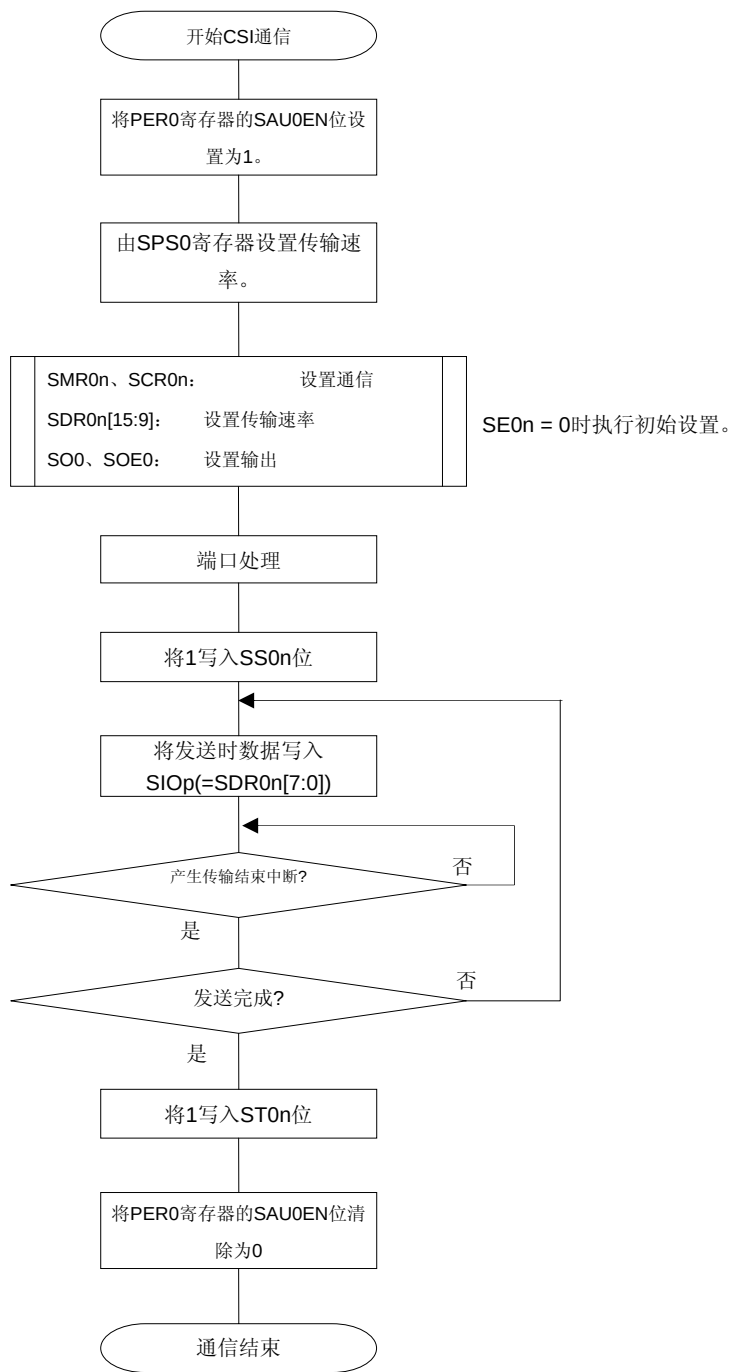
(3) 处理流程（单次发送模式中）

图 12-27. 主机发送的时序图（单次发送模式中）



备注 n: 通道编号 (n = 0 至 2)
p: CSI 编号 (p = 00、01、10)

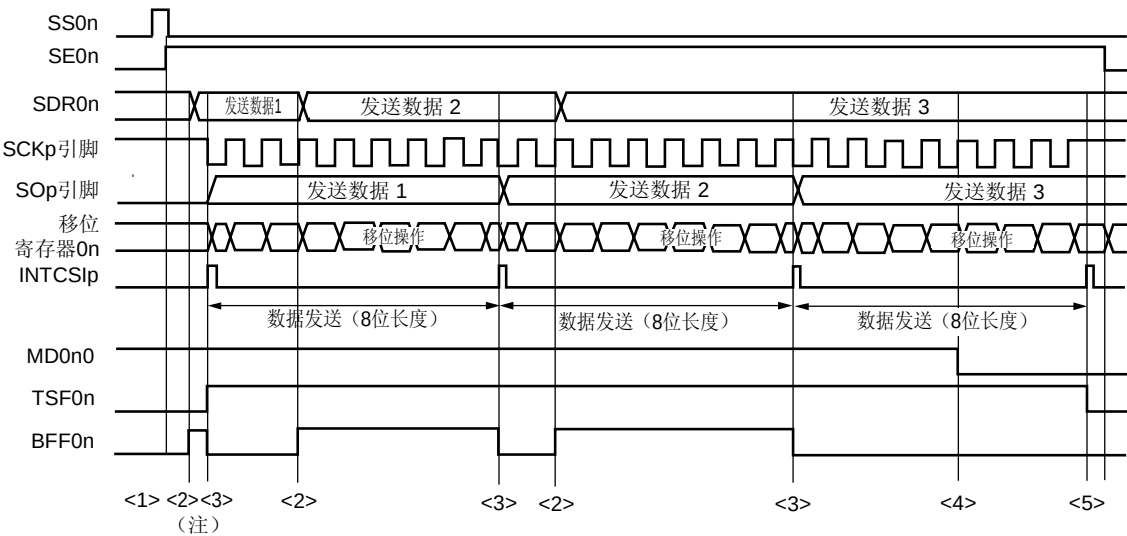
图 12-28. 主机发送流程图（单次发送模式中）



注意事项 将 PER0 寄存器置为 1 后，经过 4 个或更多个时钟后务必设置 SPS0 寄存器。

(4) 处理流程（连续发送模式中）

图 12-29. 主机发送的时序图（连续发送模式中）

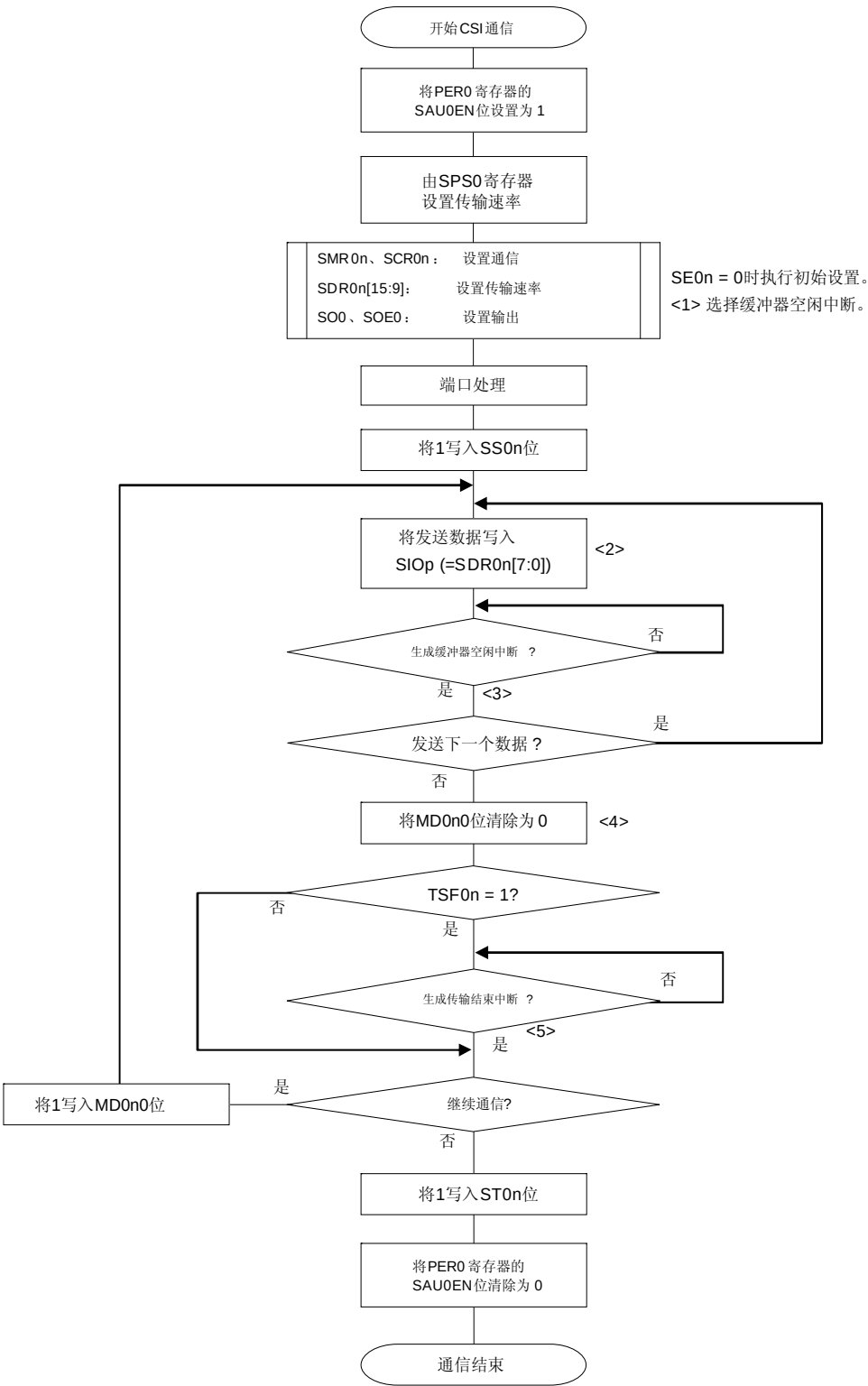


注 当在 BFF0n = 1 将发送数据写入 SDR0n 寄存器时，覆盖重写发送数据。

注意事项 即使在操作期间，仍可重写 MD0n0 位。
但是，在最后位的传输开始前对其进行重写，这样在最后发送数据的传输结束中断之前就能对其进行重写。

备注 n: 通道编号 (n = 0 至 2)
p: CSI 编号 (p = 00、01、10)

图 12-30. 主机发送的流程图（连续发送模式中）



注意事项 将 PER0 寄存器置为 1 后，经过 4 个或更多个时钟后务必设置 SPS0 寄存器。

备注 图中的<1>至<5>对应着图 12-29 主机发送时序图（连续发送模式中）中的<1>至<5>。

12.5.2 主机接收

主机接收是指 78K0R/Kx3-L 输出一个传输时钟并接收来自其它设备的数据。

3-线串行I/O	CSI00	CSI01	CSI10
目标通道	SAU 的通道0	SAU 的通道1	SAU 的通道2
所用引脚	SCK00、SI00	SCK01、SI01	SCK10、SI10
中断	INTCSI00	INTCSI01	INTCSI10
	仅传输结束中断（禁止设置缓冲器空闲中断。）		
错误检测标志	仅超限错误检测标志(OVF0n)		
传输数据长度	7或8位		
传输速率	最大值 $f_{CLK}/4$ [MHz]，最小值 $f_{CLK}/(2 \times 2^{11} \times 128)$ [MHz] ^注 f_{CLK} : 系统时钟频率		
数据相位	可由DAP0n位选择 • DAP0n = 0: 始于串行时钟操作开始的数据输入。 • DAP0n = 1: 始于串行时钟操作开始之前半个时钟的数据输入。		
时钟相位	可由CKP0n位选择 • CKP0n = 0: 正向 • CKP0n = 1: 反向		
数据方向	MSB或LSB先行		

注 在满足以上条件和电气规范中的 AC 特性的范围内使用该操作（参见第二十八章 电气规范（目标））。

备注 n: 通道编号 (n = 0 至 2)

(1) 寄存器设置

图 12-31. 有关 3-线串行 I/O 主机接收寄存器的内容的示例
(CSI00、CSI01、CSI10)

(a) 串行输出寄存器 0 (SO0) ... 仅设置目标通道的位。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SO0	0	0	0	0	1	CKO02 0/1	CKO01 0/1	CKO00 0/1	0	0	0	0	1	SO02 ×	SO01 ×	SO00 ×

如果数据相位正向 (CKP0n = 0)，则当这些位为 1 时开始通信。如果相位反向 (CKP0n = 1)，当这些位为 0 时开始通信。

(b) 串行输出使能寄存器 0 (SOE0) ... 仅将目标通道的位设置为 0。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOE0	0	0	0	0	0	0	0	0	0	0	0	0	0	SOE02 0/1	SOE01 0/1	SOE00 0/1

(c) 串行通道开始寄存器 0 (SS0) ... 仅将目标通道的位设置为 1。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SS0	0	0	0	0	0	0	0	0	0	0	0	0	SS03 ×	SS02 0/1	SS01 0/1	SS00 0/1

(d) 串行模式寄存器 0n (SMR0n)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SMR0n	CKS0n 0/1	CCS0n 0	0	0	0	0	0	STS0n 0	0	SIS0n0 0	1	0	0	MD0n2 0	MD0n1 0	MD0n0 0

通道 n 的操作模式
0: 传输结束中断

(e) 串行通信操作设置寄存器 0n (SCR0n)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SCR0n	TXE0n 0	RXE0n 1	DAP0n 0/1	CKP0n 0/1	0	EOC0n 0	PTC0n1 0	PTC0n0 0	DIR0n 0/1	0	SLC0n1 0	SLC0n0 0	0	DLS0n2 1	DLS0n1 1	DLS0n0 0/1

(f) 串行数据寄存器 0n (SDR0n) (低 8 位: SI0p)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SDR0n	波特率设置							0	接收数据寄存器 (将 FFH 作为伪数据写入。)							

SI0p

备注 n: 通道编号 (n = 0 至 2)

p: CSI 编号 (p = 00、01、10)

□: 在 CSI 主机发送模式中设置固定,

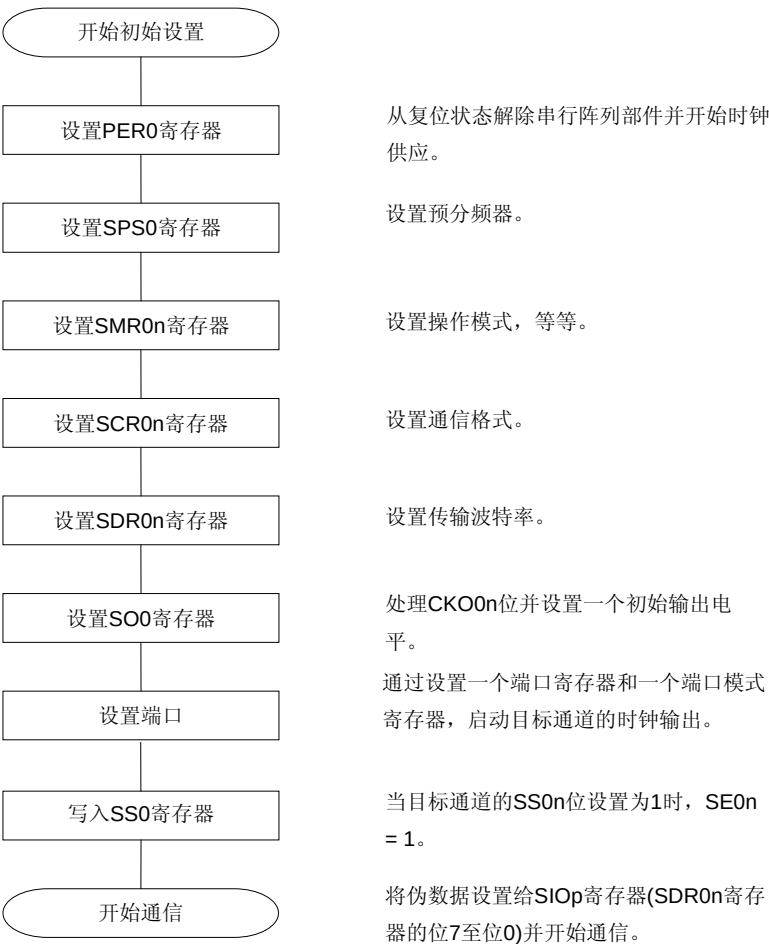
■: 禁止设置 (设为初始值)

×: 在该模式中不能使用的位(在任何模式均不使用时设为初始值)

0/1: 置为 0 或 1 取决于用户的用途

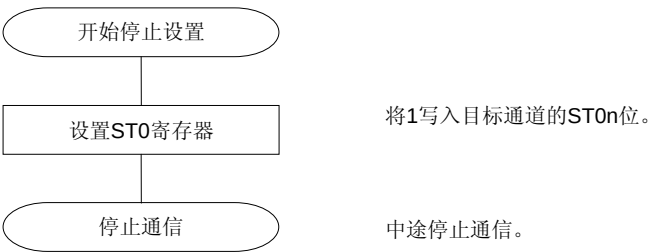
(2) 操作步骤

图 12-32. 主机接收的初始设置步骤



注意事项 将 PER0 寄存器置为 1 后，经过 4 个或更多个时钟后务必设置 SPS0 寄存器。

图 12-33. 停止主机接收的步骤



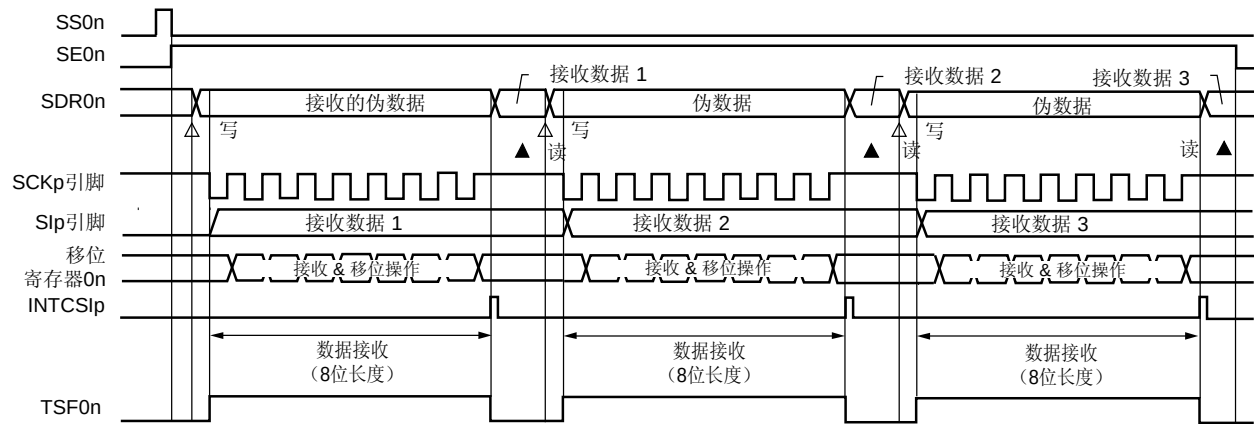
备注 即使在通信停止之后，仍保留引脚的电平。要恢复操作，需重新设置 SO0 寄存器（参见图 12-34 恢复主机接收的步骤）。

图 12-34. 恢复主机接收的步骤



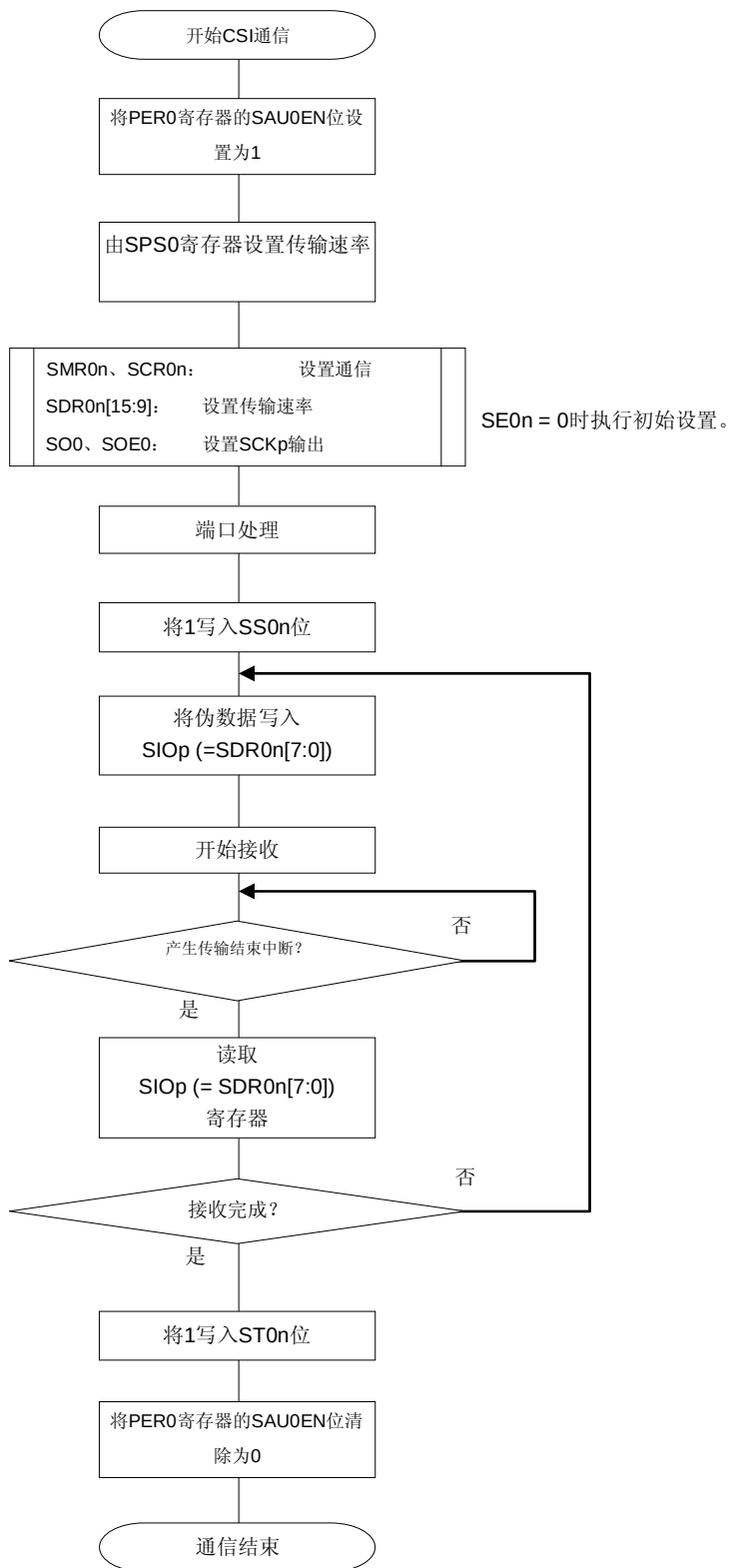
(3) 处理流程（单次接收模式中）

图 12-35. 主机接收的时序图（单次接收模式中）



备注 n: 通道编号 (n = 0 至 2)
 p: CSI 编号 (p = 00、01、10)

图 12-36. 主机接收的流程图（单次接收模式中）



注意事项 将 PER0 寄存器置为 1 后，经过 4 个或更多个时钟后务必设置 SPS0 寄存器。

12.5.3 主机发送/接收

主机发送/接收是指 78K0R/Kx3-L 输出一个传输时钟并将数据发送至其它设备或接收来自其它设备的数据。

3-线串行I/O	CSI00	CSI01	CSI10
目标通道	SAU的通道0	SAU的通道1	SAU的通道2
所用引脚	SCK00、SI00、SO00	SCK01、SI01、SO01	SCK10、SI10、SO10
中断	INTCSI00	INTCSI01	INTCSI10
	可以选择传输结束中断(单次传输模式中)或缓冲器空闲中断(连续传输模式中)。		
错误检测标志	仅超限错误检测标志(OVF0n)		
传输数据长度	7或8位		
传输速率	最大值 $f_{CLK}/4$ [MHz], 最小值 $f_{CLK}/(2 \times 2^{11} \times 128)$ [MHz] ^注 f_{CLK} : 系统时钟频率		
数据相位	可由DAP0n位选择 • DAP0n = 0: 始于串行时钟操作开始的数据I/O。 • DAP0n = 1: 始于串行时钟操作开始前半个时钟的数据I/O。		
时钟相位	可由CKP0n位选择 • CKP0n = 0: 正向 • CKP0n = 1: 反向		
数据方向	MSB或LSB先行		

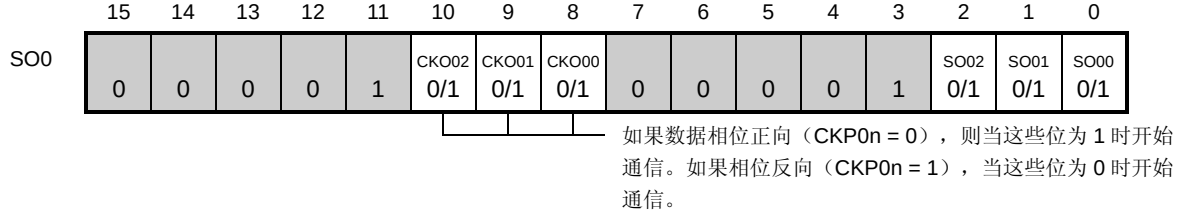
注 在满足以上条件和电气规范中的 AC 特性的范围内使用该操作（参见第二十八章 电气规范（目标））。

备注 n: 通道编号 (n = 0 至 2)

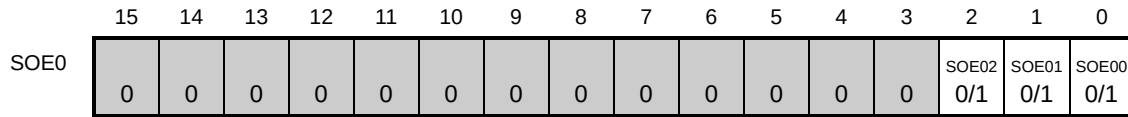
(1) 寄存器设置

图 12-37. 有关 3-线串行 I/O 主机发送/接收的寄存器的内容的示例
(CSI00、CSI01、CSI10)

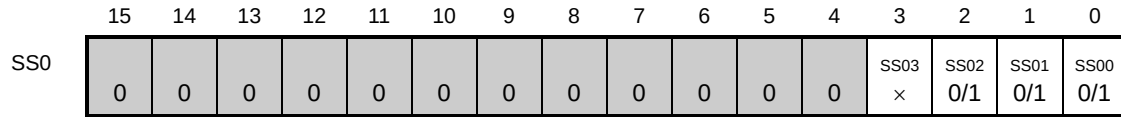
(a) 串行输出寄存器 0 (SO0) ... 仅设置目标通道的位。



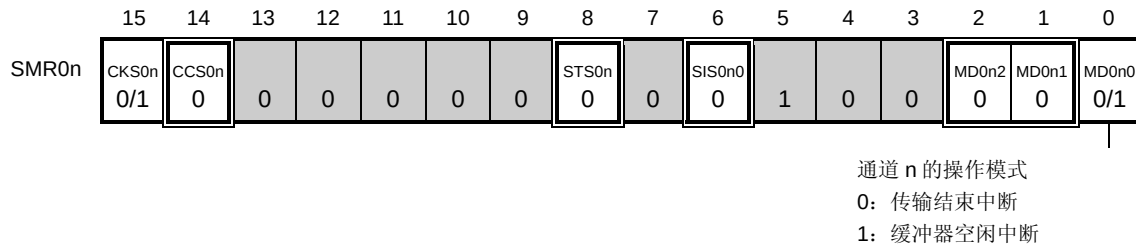
(b) 串行输出使能寄存器 0 (SOE0) ... 仅将目标通道的位设置为 1。



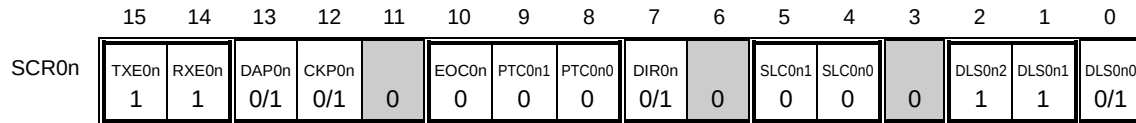
(c) 串行通道开始寄存器 0 (SS0) ... 仅将目标通道的位设置为 1。



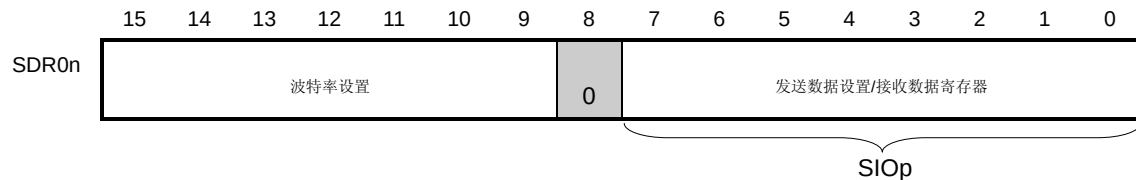
(d) 串行模式寄存器 0n (SMR0n)



(e) 串行通信操作设置寄存器 0n (SCR0n)



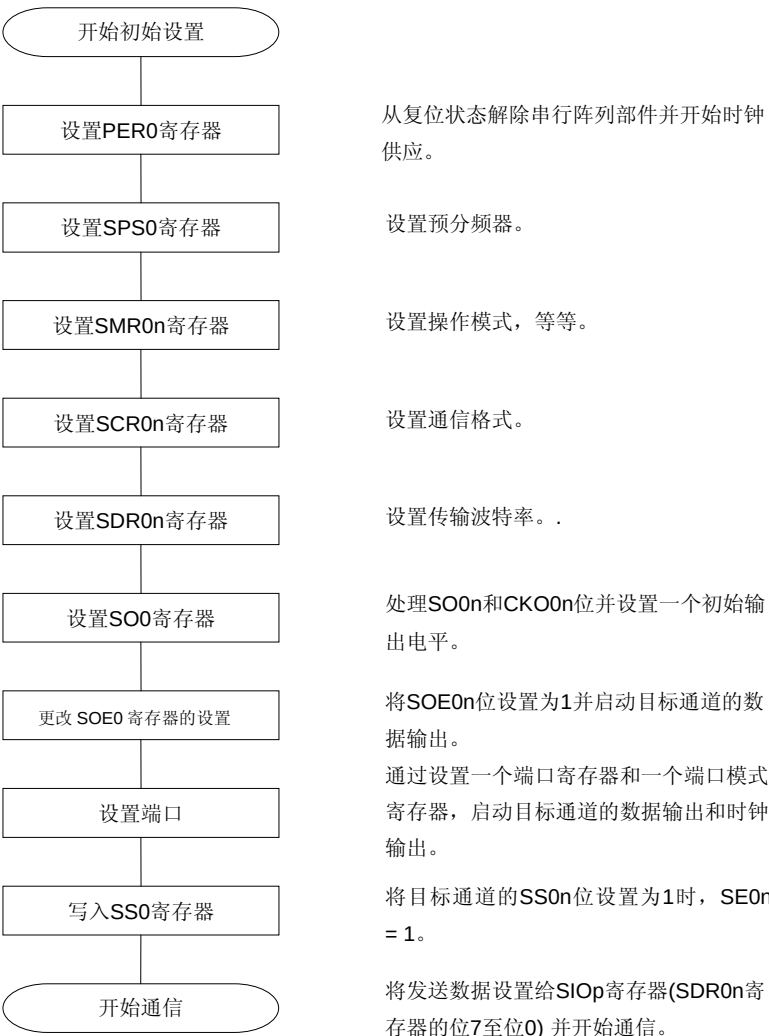
(f) 串行数据寄存器 0n (SDR0n) (低 8 位: SI0p)



备注 n: 通道编号(n = 0 至 2)
 p: CSI 编号 (p = 00、01、10)
 □ : 在 CSI 主机发送模式中设置固定, ■ : 禁止设置(设为初始值)
 ×: 在该模式中不能使用的位(在任何模式中均不使用时设为初始值)
 0/1: 置为 0 或 1 取决于用户的用途

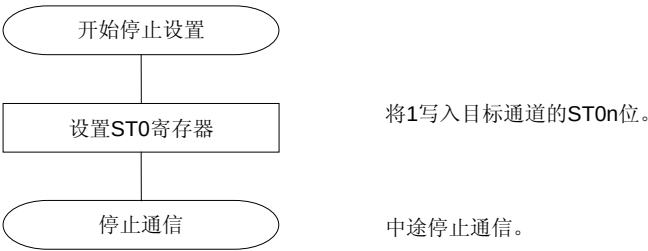
(2) 操作步骤

图 12-38. 主机发送/接收的初始设置步骤



注意事项 将 PER0 寄存器置为 1 后，经过 4 个或更多个时钟后务必设置 SPS0 寄存器。

图 12-39. 停止主机发送/接收的步骤



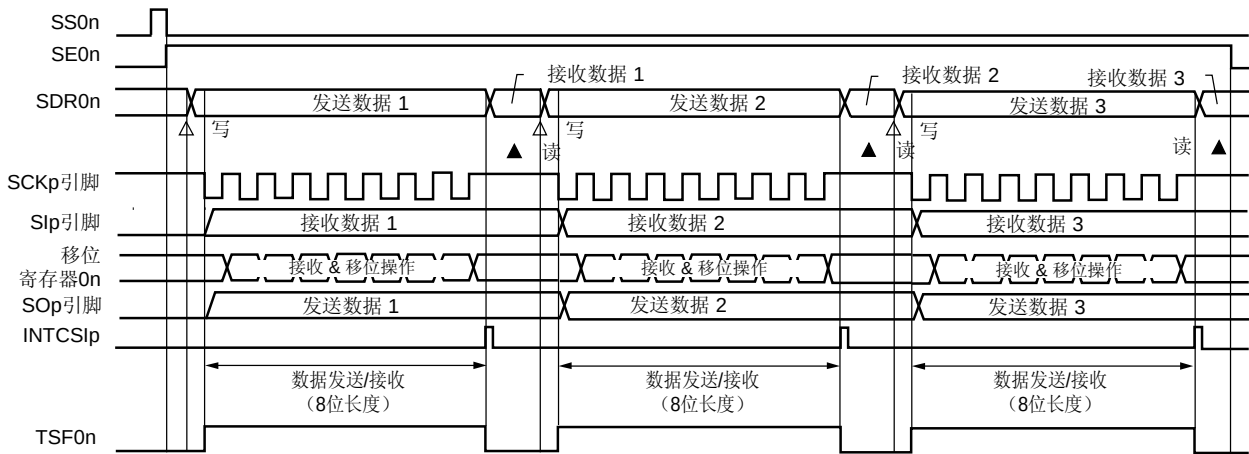
备注 即使在通信停止之后，仍保留引脚的电平。要恢复操作，需重新设置 SO0 寄存器（参见图 12-40 恢复主机发送/接收的步骤）。

图 12-40. 恢复主机发送/接收的步骤



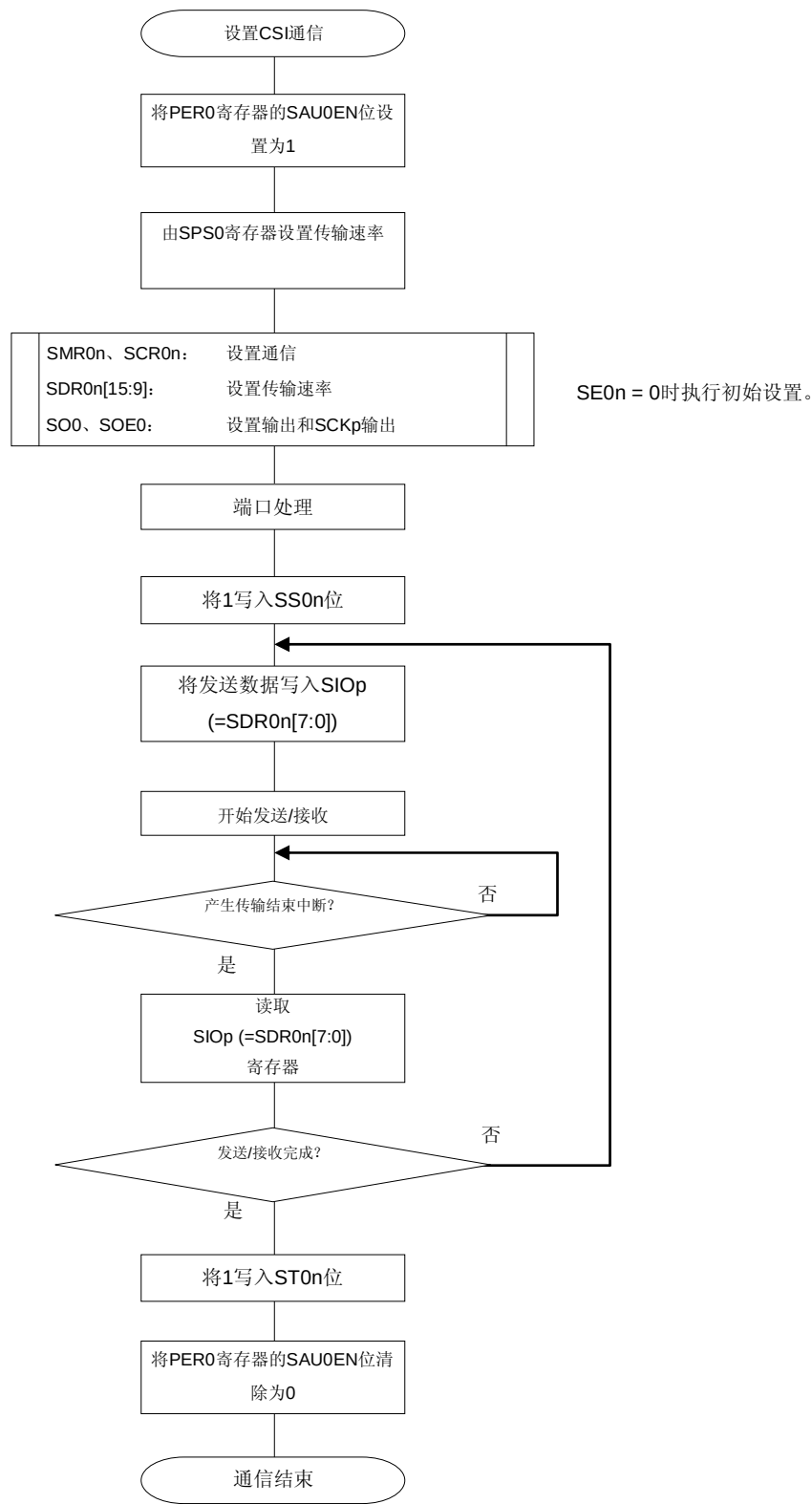
(3) 处理流程（单次发送/接收模式中）

图 12-41. 主机发送/接收的时序图（单次发送/接收模式中）



备注 n: 通道编号 (n = 0 至 2)
p: CSI 编号 (p = 00、01、10)

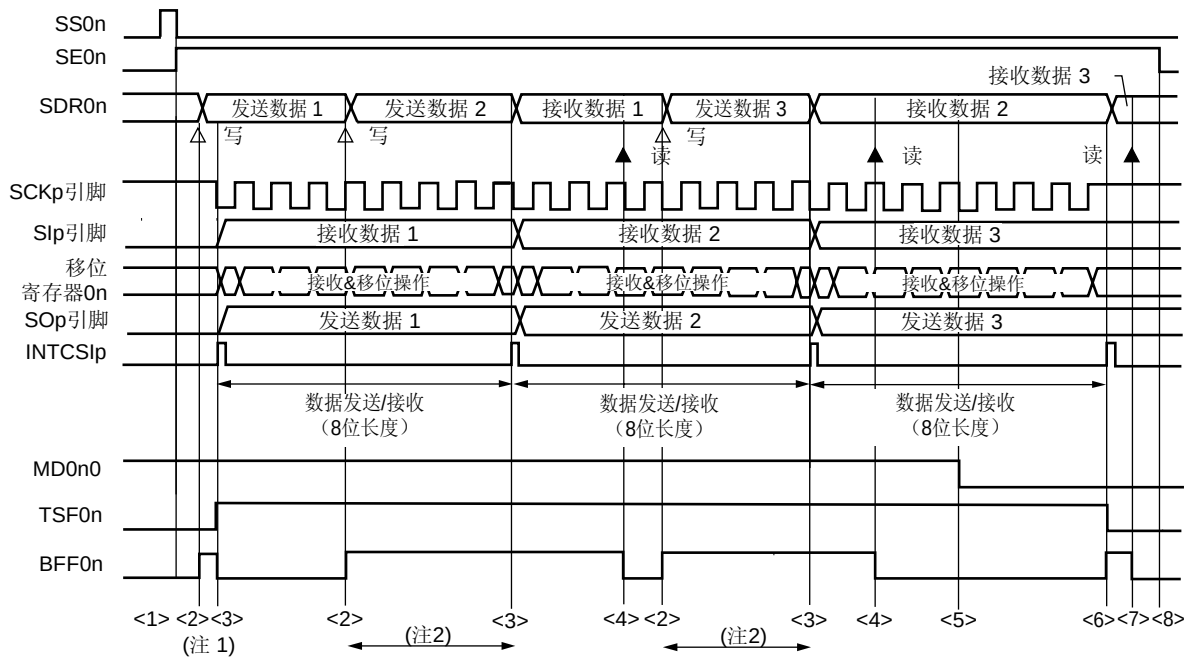
图 12-42. 主机发送/接收的流程图（单次发送/接收模式中）



注意事项 将 PER0 寄存器置为 1 后，经过 4 个或更多个时钟后务必设置 SPS0 寄存器。

(4) 处理流程（连续发送/接收模式中）

图 12-43. 主机发送/接收的时序图（连续发送/接收模式中）



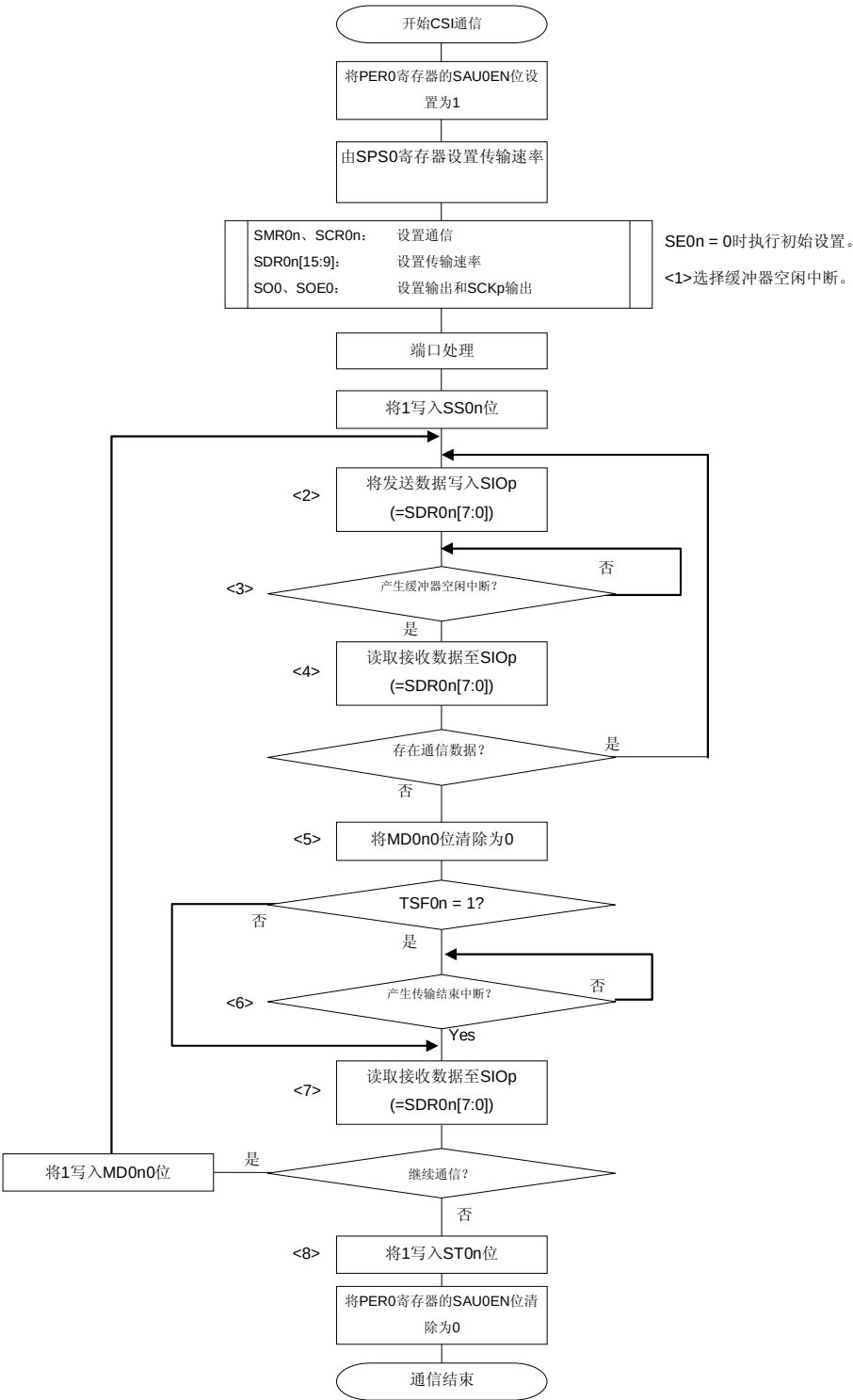
- 注
- 1. 当在 BFF0n = 1 将发送数据写入 SDR0n 寄存器时，覆盖重写发送数据。
 - 2. 在这个时期内通过读取 SDR0n 寄存器可以读取发送数据。此时，不影响传输操作。

注意事项 即使在操作期间，仍可重写 MD0n0 位。

但是，要在最后位的传输开始前对其进行重写，这样在最后发送数据的传输结束中断之前就已经对其进行了重写。

- 备注
- 1. 图中的<1>至<8>对应着图 12-44 主机发送/接收的流程图（连续发送/接收模式中）中的<1>至<8>。
 - 2. n: 通道编号 (n = 0 至 2)
p: CSI 编号 (p = 00、01、10)

图 12-44. 主机发送/接收的流程图（连续发送/接收模式中）



注意事项 将 PER0 寄存器置为 1 后，经过 4 个或更多个时钟后务必设置 SPS0 寄存器。

备注 图中的<1>至<8>对应着图 12-43 主机发送/接收的时序图（连续发送/接收模式中）中的<1>至<8>。

12.5.4 从机发送

从机发送是指 78K0R/Kx3-L 在一个正从其它设备被输入的传输时钟的状态中将数据发送至其它设备。

3-线串行I/O	CSI00	CSI01	CSI10
目标通道	SAU的通道0	SAU的通道1	SAU的通道2
所用引脚	SCK00、SO00	SCK01、SO01	SCK10、SO10
中断	INTCSI00	INTCSI01	INTCSI10
	可以选择传输结束中断(单次传输模式中)或缓冲器空闲中断(连续传输模式中)。		
错误检测标志	仅超限错误检测标志(OVF0n)		
传输数据长度	7或8位		
传输速率	$f_{CLK}/6$ [MHz]和 $f_{MCK}/2$ [MHz]中较小的一个就是最大传输速率 ^{注1,2} 。		
数据相位	可由DAP0n位选择 • DAP0n = 0: 始于串行时钟操作开始的数据输出。 • DAP0n = 1: 始于串行时钟操作开始前半个时钟的数据输出。		
时钟相位	可由CKP0n位选择 • CKP0n = 0: 正向 • CKP0n = 1: 反向		
数据方向	MSB或LSB先行		

注 1. 由于输入至引脚 SCK00、SCK01 和 SCK10 的外部串行时钟在内部被采样并使用，因此最大传输速率就是 $f_{CLK}/6$ [MHz]和 $f_{MCK}/2$ [MHz]两者中的较小者。

2. 在满足以上条件和电气规范中的 AC 特性的范围内使用该操作（参见第二十八章 电气规范（目标））。

备注 1. f_{MCK} : 目标通道的操作时钟(MCK)频率

f_{CLK} : 系统时钟频率

2. n: 通道编号(n = 0 至 2)

(1) 寄存器设置

图 12-45. 有关 3-线串行 I/O 从机发送寄存器的内容的示例
(CSI00、CSI01、CSI10)

(a) 串行输出寄存器 0 (SO0) ... 仅设置目标通道的位。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SO0	0	0	0	0	1	CKO02 ×	CKO01 ×	CKO00 ×	0	0	0	0	1	SO02 0/1	SO01 0/1	SO00 0/1

(b) 串行输出使能寄存器 0 (SOE0) ... 仅将目标通道的位设置为 1。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOE0	0	0	0	0	0	0	0	0	0	0	0	0	0	SOE02 0/1	SOE01 0/1	SOE00 0/1

(c) 串行通道开始寄存器 0 (SS0) ... 仅将目标通道的位设置为 1。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SS0	0	0	0	0	0	0	0	0	0	0	0	0	SS03 ×	SS02 0/1	SS01 0/1	SS00 0/1

(d) 串行模式寄存器 0n (SMR0n)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SMR0n	CKS0n 0/1	CCS0n 1	0	0	0	0	0	STS0n 0	0	SIS0n0 0	1	0	0	MD0n2 0	MD0n1 0	MD0n0 0/1

通道 n 的操作模式
0: 传输结束中断
1: 缓冲器空闲中断

(e) 串行通信操作设置寄存器 0n (SCR0n)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SCR0n	TXE0n 1	RXE0n 0	DAP0n 0/1	CKP0n 0/1	0	EOC0n 0	PTC0n1 0	PTC0n0 0	DIR0n 0/1	0	SLC0n1 0	SLC0n0 0	0	DLS0n2 1	DLS0n1 1	DLS0n0 0/1

(f) 串行数据寄存器 0n (SDR0n) (低 8 位: SI0p)

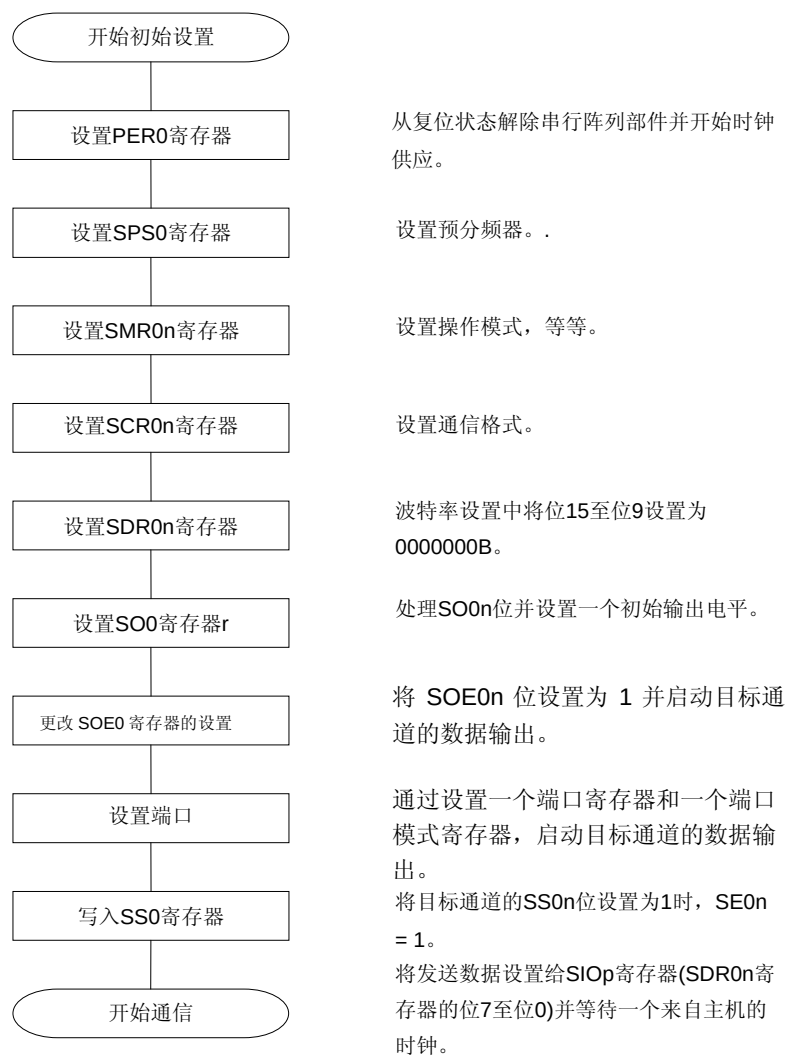
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SDR0n	波特率设置							0	发送数据设置							

SI0p

备注 n: 通道编号 (n = 0 至 2)
p: CSI 编号 (p = 00、01、10)
□: Setting is fixed 在 CS 主机发送模式中设置固定, ■: 禁止设置(设为初始值)
×: 在该模式中不能使用的位(在任何模式均不能使用时设为初始值)
0/1: 置为 0 或 1 取决于用户的用途

(2) 操作步骤

图 12-46. 从机发送的初始设置步骤



注意事项 将 PER0 寄存器置为 1 后，经过 4 个或更多个时钟后务必设置 SPS0 寄存器。

图 12-47. 停止从机发送的步骤



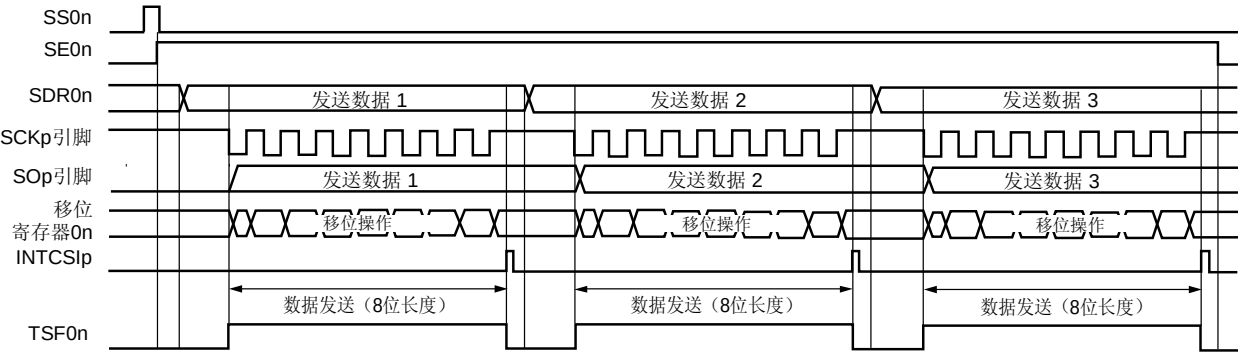
备注 即使在通信停止之后，仍保留引脚的电平。要恢复操作，需重新设置 SO0 寄存器（参见图 12-48 恢复从机发送的步骤）。

图 12-48. 恢复从机发送的步骤



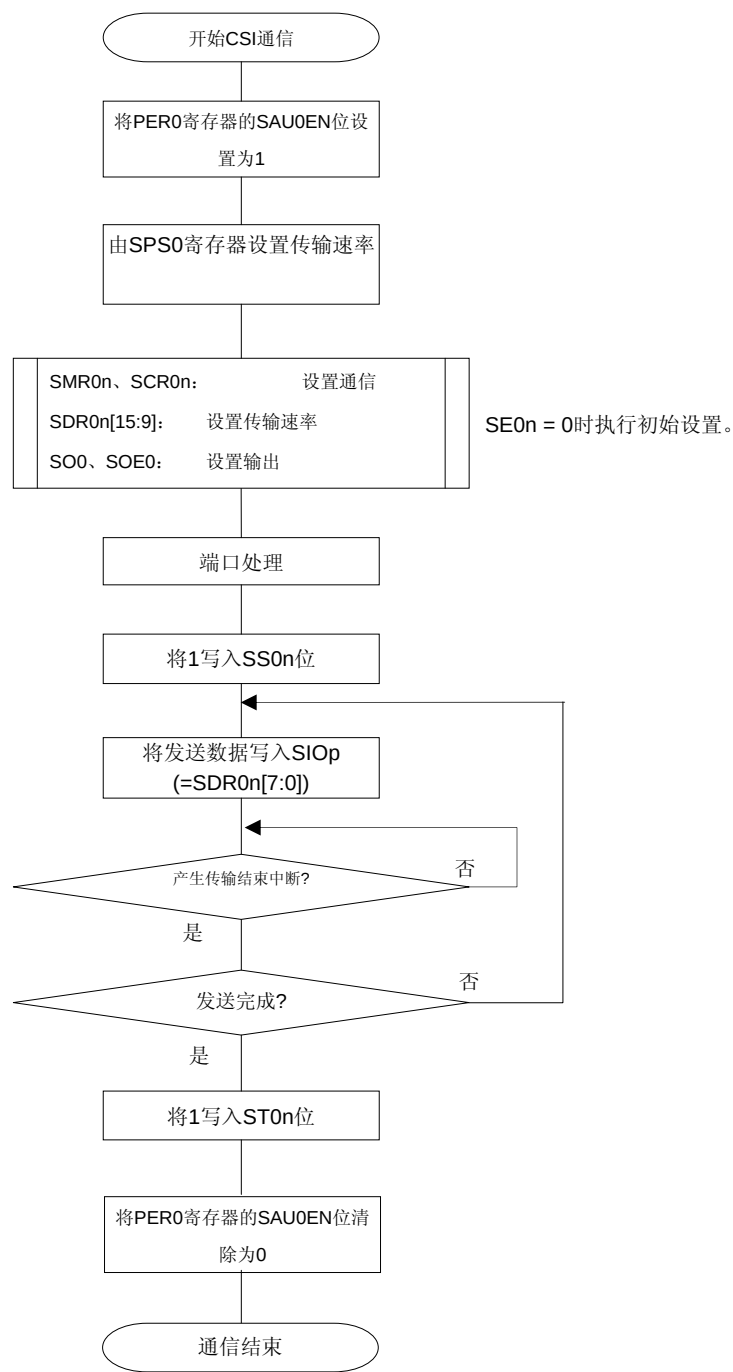
(3) 处理流程（单次发送模式中）

图 12-49. 从机发送的时序图（单次发送模式中）



备注 n: 通道编号 (n = 0 至 2)
p: CSI 编号 (p = 00、01、10)

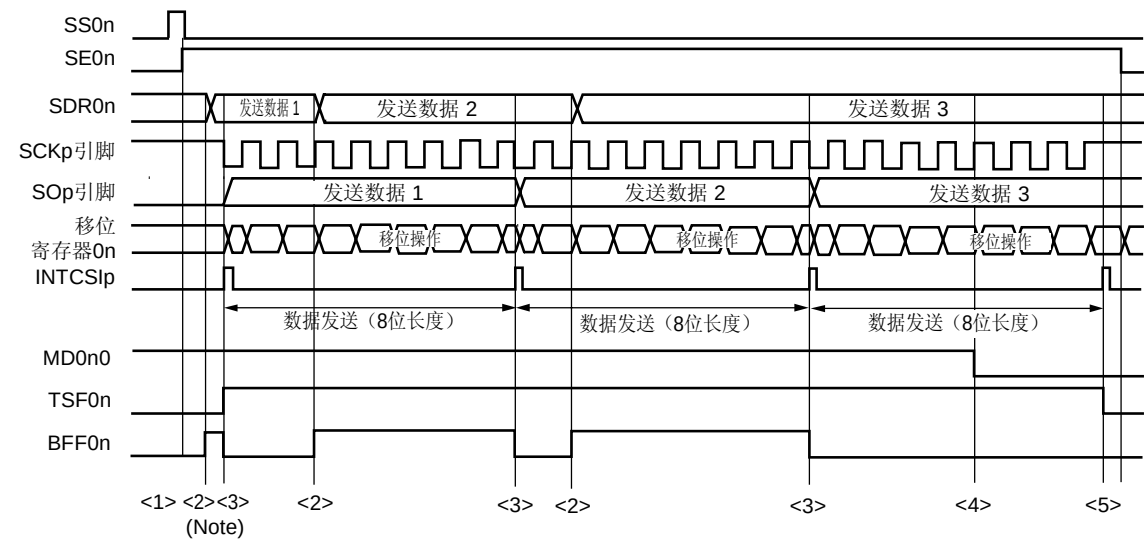
图 12-50. 从机发送的流程图（单次发送模式中）



注意事项 将 PER0 寄存器置为 1 后，经过 4 个或更多个时钟后务必设置 SPS0 寄存器。

(4) 处理流程（连续发送模式中）

图 12-51. 从机发送的时序图（连续发送模式中）

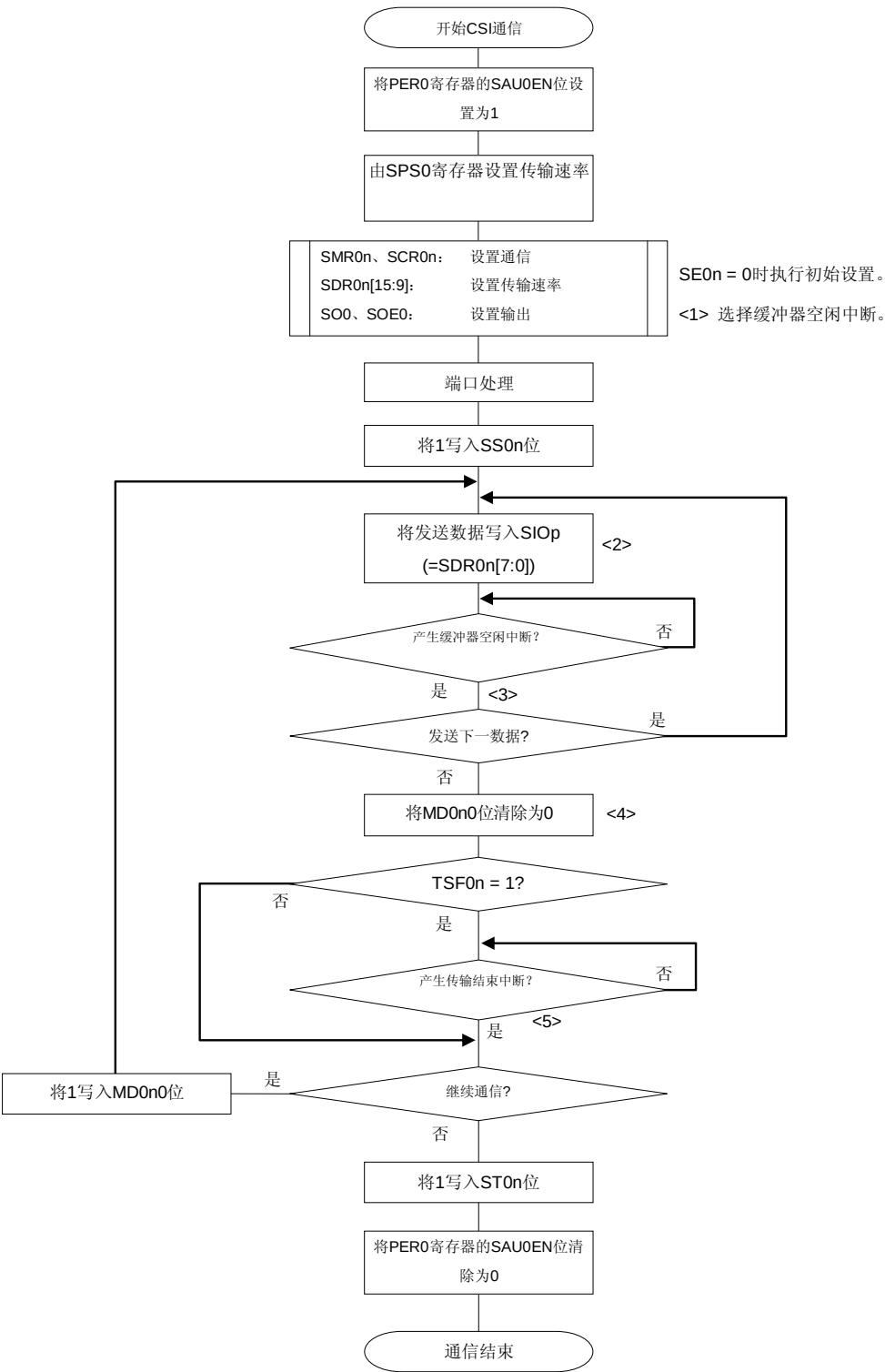


注 当在 BFF0n = 1 将发送数据写入 SDR0n 寄存器时，覆盖重写发送数据。

注意事项 即使在操作期间，仍可重写 MD0n0 位。但是，需在最后位的传输开始之前对其进行重写。

备注 n: 通道编号 (n = 0 至 2)
p: CSI 编号 (p = 00、01、10)

图 12-52. 从机发送的流程图（连续发送模式中）



注意事项 将 PER0 寄存器置为 1 后，经过 4 个或更多个时钟后务必设置 SPS0 寄存器。

备注 图中的<1>至<5>对应着图 12-51 从机发送的时序图（连续发送模式中）中的<1>至<5>。

12.5.5 从机接收

从机接收是指 78K0R/Kx3-L 在一个正从其它设备被输入的传输时钟的状态中接收来自其它设备的数据。

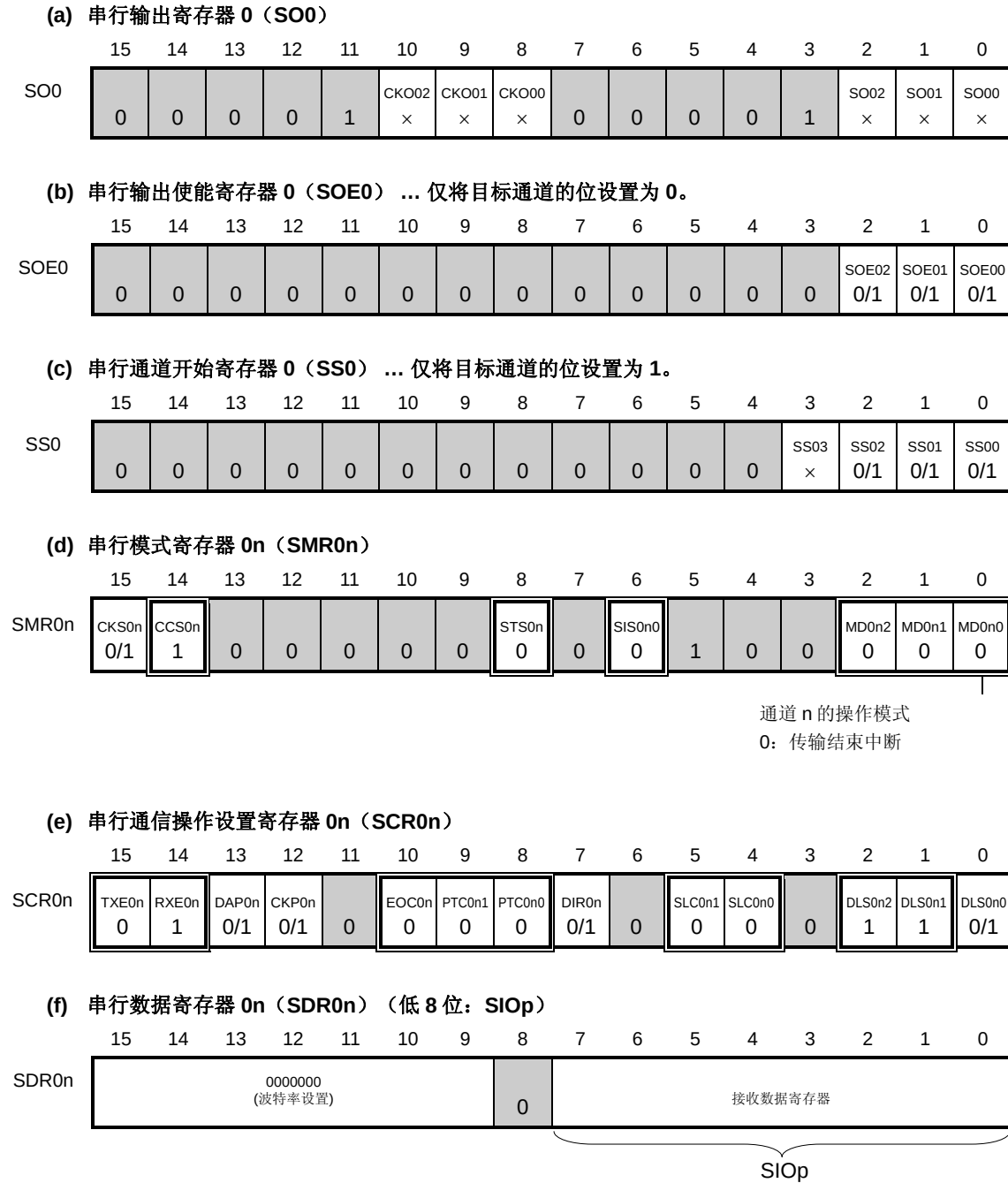
3-线串行I/O	CSI00	CSI01	CSI10
目标通道	SAU的通道0	SAU的通道1	SAU的通道2
所用引脚	SCK00、SI00	SCK01、SI01	SCK10、SI10
中断	INTCSI00	INTCSI01	INTCSI10
	仅传输结束中断(禁止设置缓冲器空闲中断。)		
错误检测标志	仅超限错误标志 (OVF0n)		
传输数据长度	7或8位		
传输速率	$f_{CLK}/6$ [MHz]和 $f_{MCK}/2$ [MHz]中的较小者就是最大传输速率 ^{注1, 2} 。		
数据相位	可由 DAP0n 位选择 • DAP0n = 0: 始于串行时钟操作开始的数据输入。 • DAP0n = 1: 始于串行时钟操作开始前半个时钟的数据输入。		
时钟相位	可由CKP0n位选择 • CKP0n = 0: 正向 • CKP0n = 1: 反向		
数据方向	MSB或LSB先行		

- 注 1. 由于输入至引脚 SCK00、SCK01 和 SCK10 的外部串行时钟在内部被采样并使用，因此最大传输速率就是 $f_{CLK}/6$ [MHz]和 $f_{MCK}/2$ [MHz]两者中的较小者。
2. 在满足以上条件和电气规范中的 AC 特性的范围内使用该操作（参见第二十八章 电气规范（目标））。

- 备注 1. f_{MCK} : 目标通道的操作时钟（MCK）频率
 f_{CLK} : 系统时钟频率
2. n: 通道编号 (n = 0 至 2)

(1) 寄存器设置

图 12-53. 有关 3-线串行 I/O 从机接收寄存器的内容的示例
(CSI00、CSI01、CSI10)



备注

n: 通道编号 (n = 0 至 2)

p: CSI 编号 (p = 00、01、10)

□: 在 CSI 主机发送模式中设置固定,

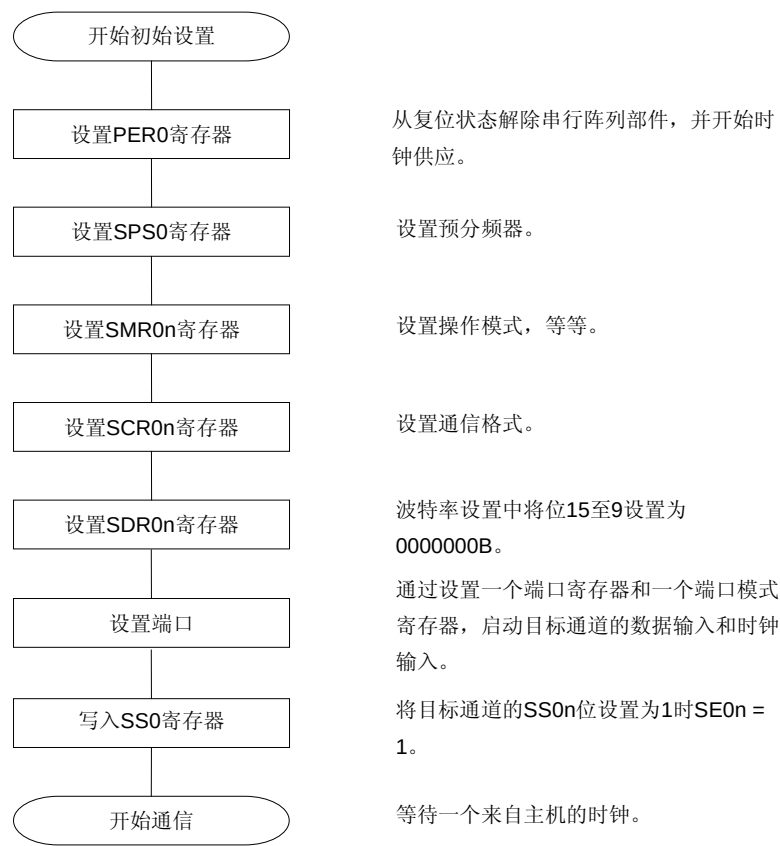
■: 禁止设置(设为初始值)

×: 该模式中不能使用的位(在任何模式中均不能使用时设为初始值)

0/1: 置为 0 或 1 取决于用户的用途

(2) 操作步骤

图 12-54. 从机接收的初始设置步骤



注意事项 将 PER0 寄存器置为 1 后，经过 4 个或更多个时钟后务必设置 SPS0 寄存器。

图 12-55. 停止从机接收的步骤

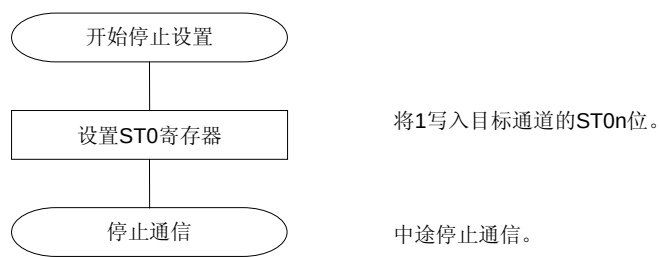
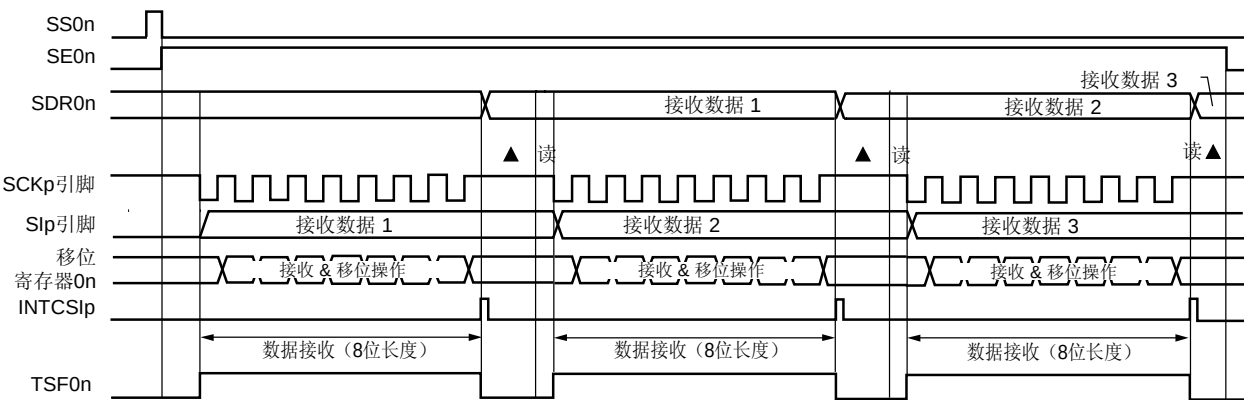


图 12-56. 恢复从机接收的步骤



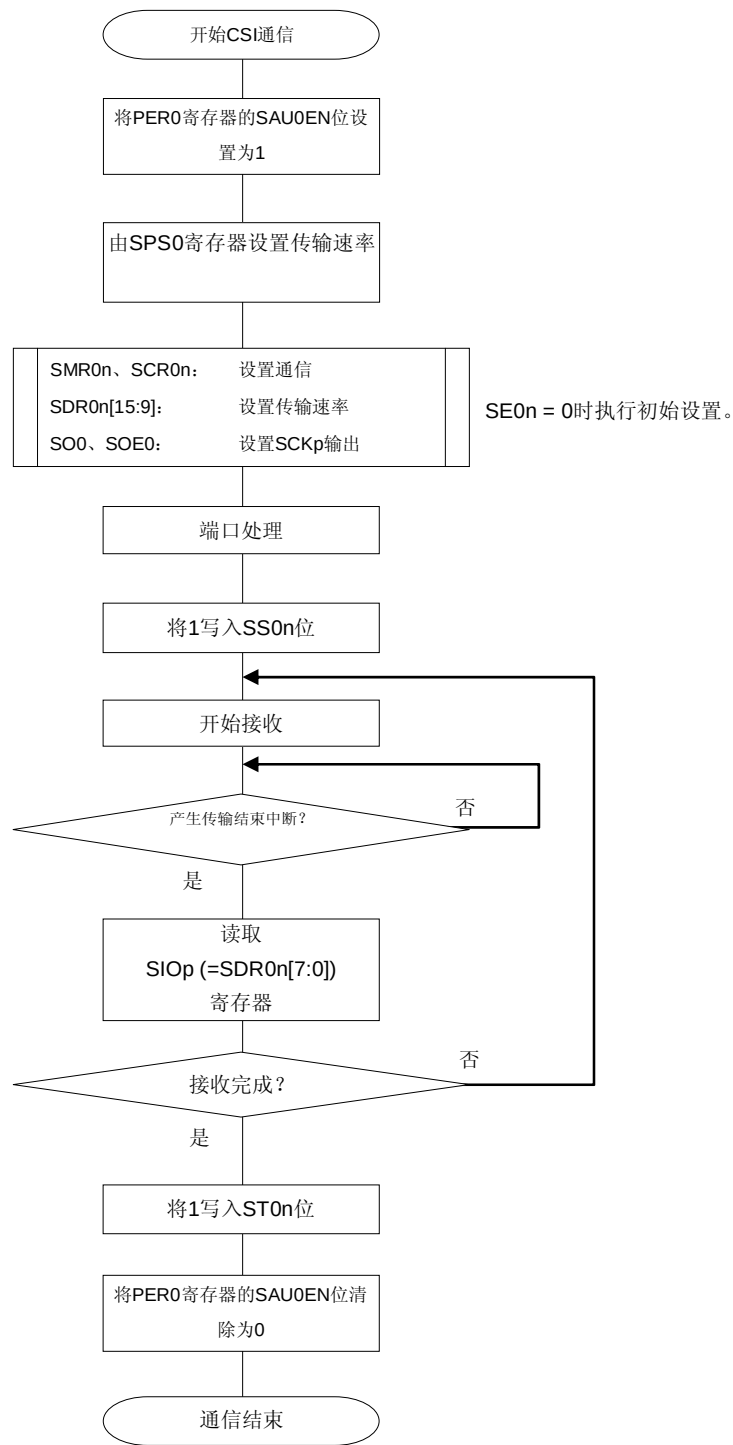
(3) 处理流程（单次接收模式中）

图 12-57. 从机接收的时序图（单次接收模式中）



备注 n: 通道编号 (n = 0 至 2)
 p: CSI 编号 (p = 00、01、10)

图 12-58. 从机接收的流程图（单次接收模式中）



注意事项 将 PER0 寄存器置为 1 后，经过 4 个或更多个时钟后务必设置 SPS0 寄存器。

12.5.6 从机发送/接收

从机发送/接收是指 78K0R/Kx3-L 在一个正从其它设备被输入的传输时钟的状态中将数据发送至其它设备或者接收来自其它设备的数据。

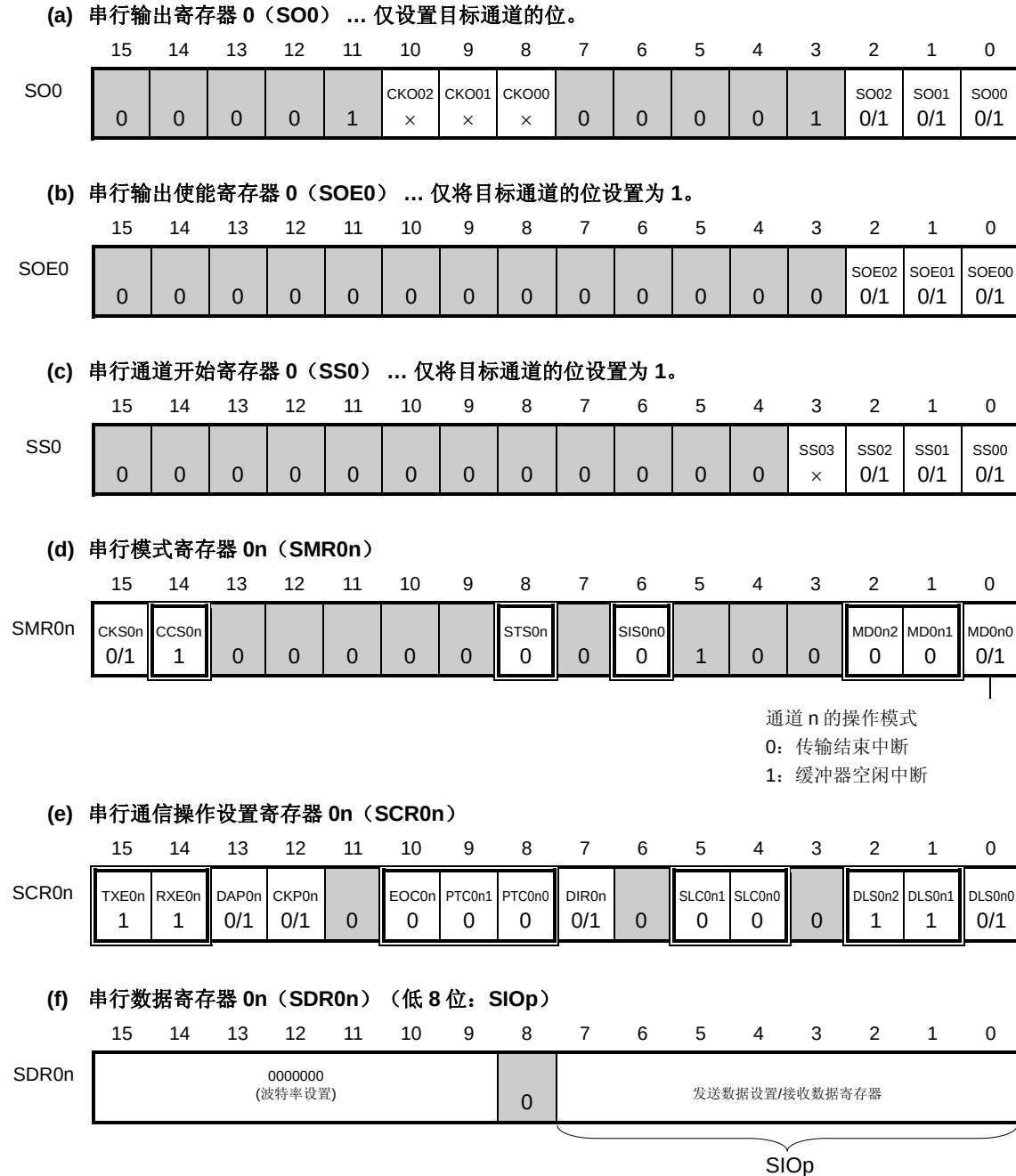
3-线串行I/O	CSI00	CSI01	CSI10
目标通道	SAU的通道0	SAU的通道1	SAU的通道2
所用引脚	SCK00、SI00、SO00	SCK01、SI01、SO01	SCK10、SI10、SO10
中断	INTCSI00	INTCSI01	INTCSI10
	可以选择传输结束中断(单次传输模式中)或者缓冲器空闲中断(连续传输模式中)。		
错误检测标志	仅超限错误检测标志(OVF0n)		
传输数据长度	7或8位		
传输速率	$f_{CLK}/6$ [MHz]和 $f_{MCK}/2$ [MHz]中的较小者就是最大传输速率 ^{注 1, 2} 。		
数据相位	可由DAP0n位选择 • DAP0n = 0: 始于串行时钟操作开始的数据I/O。 • DAP0n = 1: 始于串行时钟操作开始之前半个时钟的数据I/O。		
时钟相位	可由CKP0n位选择 • CKP0n = 0: 正向 • CKP0n = 1: 反向		
数据方向	MSB或LSB先行		

- 注 1.** 由于输入至引脚 SCK00、SCK01 和 SCK10 的外部串行时钟在内部被采样并使用，因此最大传输速率就是 $f_{CLK}/6$ [MHz]和 $f_{MCK}/2$ [MHz]两者中的较小者。
- 2.** 在满足以上条件和电气规范中的 AC 特性的范围内使用该操作（参见第二十八章 电气规范（目标））。

- 备注 1.** f_{MCK} : 目标通道的操作时钟（MCK）频率
 f_{CLK} : 系统时钟频率
- 2.** n: 通道编号 (n = 0 至 2)

(1) 寄存器设置

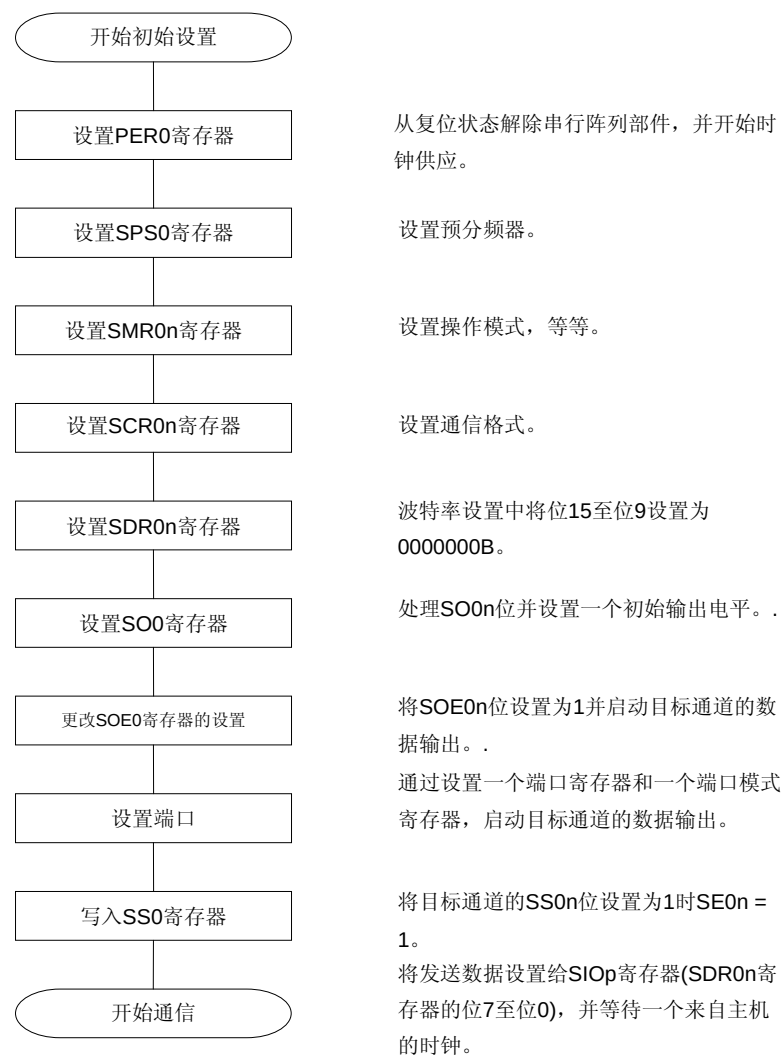
图 12-59. 有关 3-线串行 I/O 从机发送/接收寄存器的内容的示例
(CSI00、CSI01、CSI10)



备注 n: 通道编号 (n = 0 至 2)
p: CSI 编号 (p = 00、01、10)
□: 在 CSI 主机发送模式中设置固定, ■: 禁止设置(设为初始值)
×: 该模式中不能使用的位(在任何模式中均不能使用时设为初始值)
0/1: 置为 0 或 1 取决于用户的用途

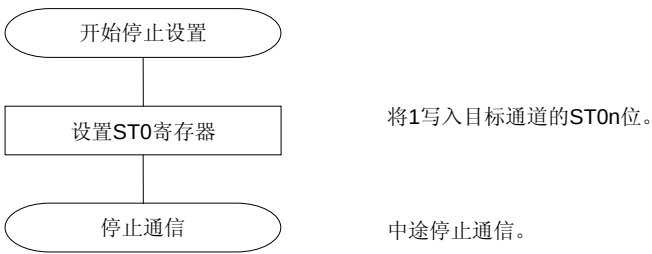
(2) 操作步骤

图 12-60. 从机发送/接收的初始设置步骤



注意事项 将 PER0 寄存器置为 1 后，经过 4 个或更多个时钟后务必设置 SPS0 寄存器。

图 12-61. 停止从机发送/接收的步骤



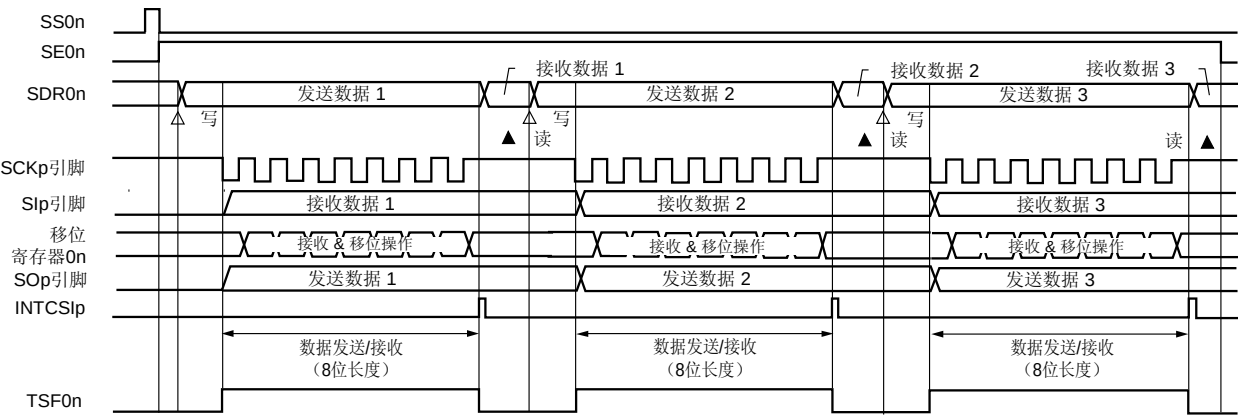
备注 即使在通信停止之后，仍保留引脚的电平。要恢复操作，需重新设置 SO0 寄存器（参见图 12-62 恢复从机发送/接收的步骤）。

图 12-62. 恢复从机发送/接收的步骤



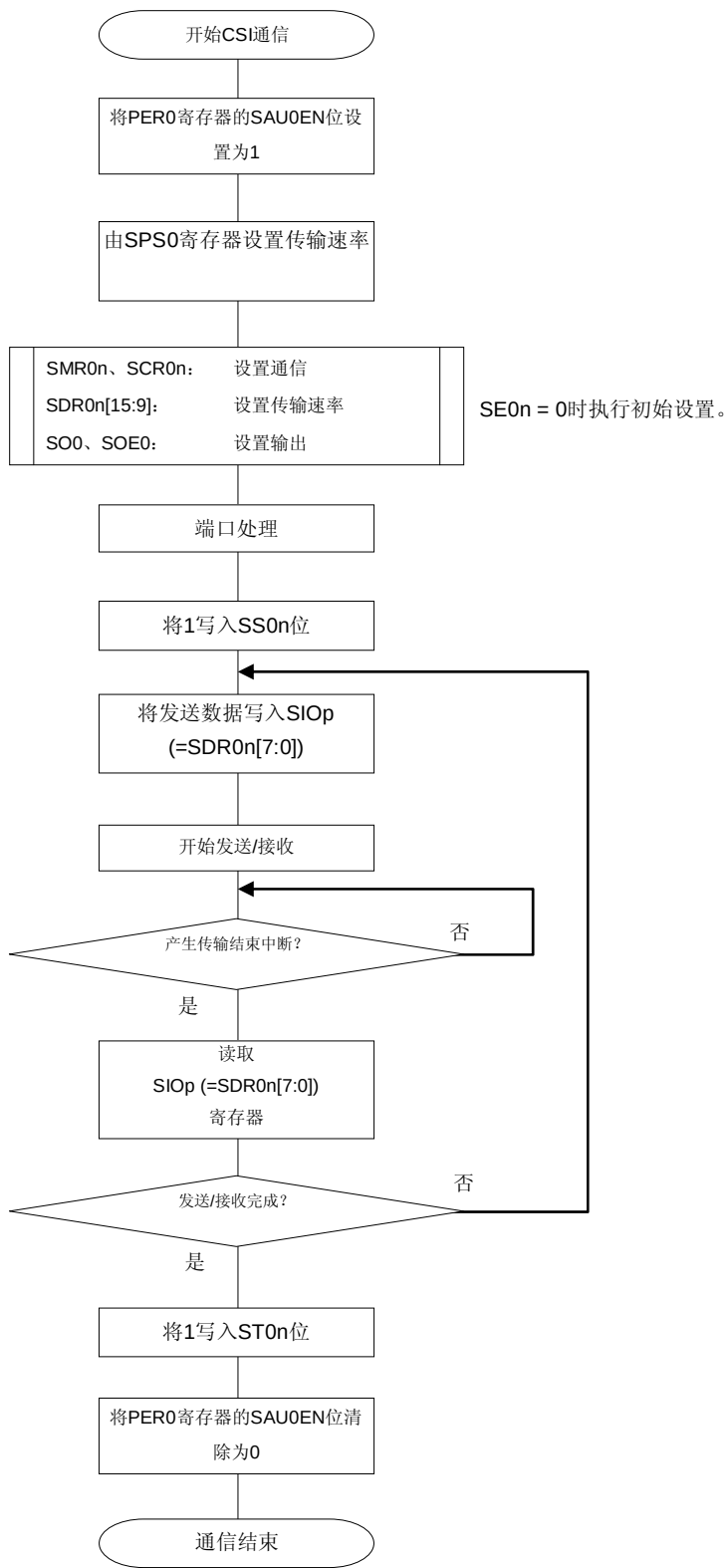
(3) 处理流程（单次发送/接收模式中）

图 12-63. 从机发送/接收的时序图（单次发送/接收模式中）



备注 n: 通道编号 (n = 0 至 2)
 p: CSI 编号 (p = 00、01、10)

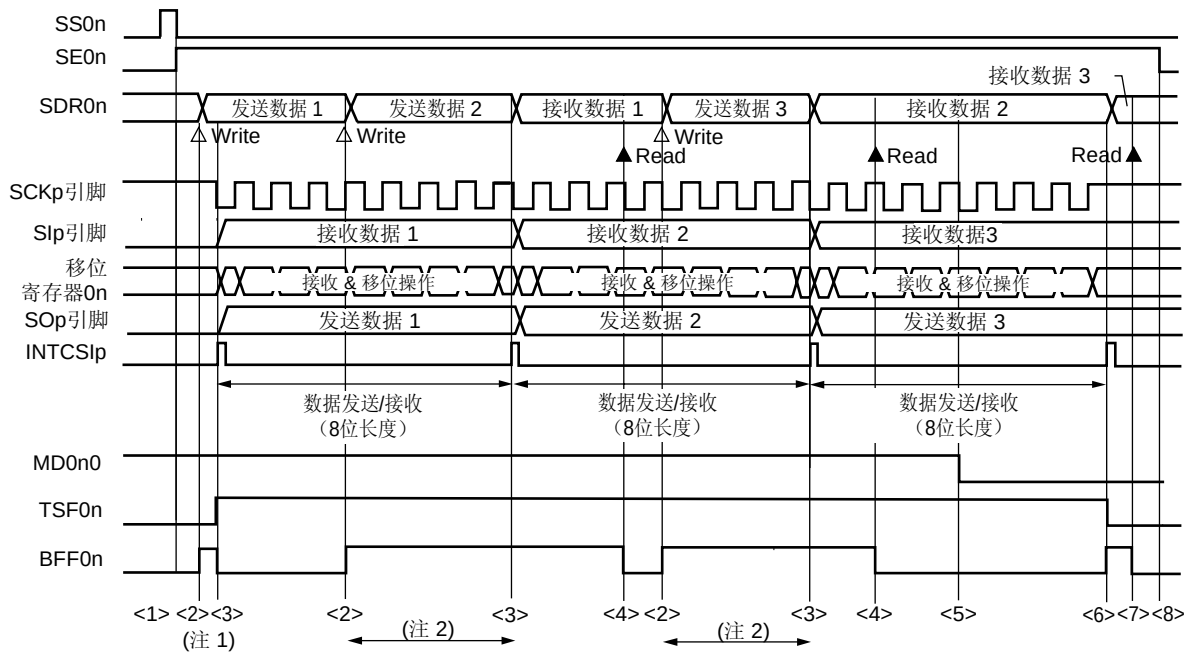
图 12-64. 从机发送/接收的流程图（单次发送/接收模式中）



注意事项 将 PER0 寄存器置为 1 后，经过 4 个或更多个时钟后务必设置 SPS0 寄存器。

(4) 处理流程（连续发送/接收模式中）

图 12-65. 从机发送/接收的时序图（连续发送/接收模式中）

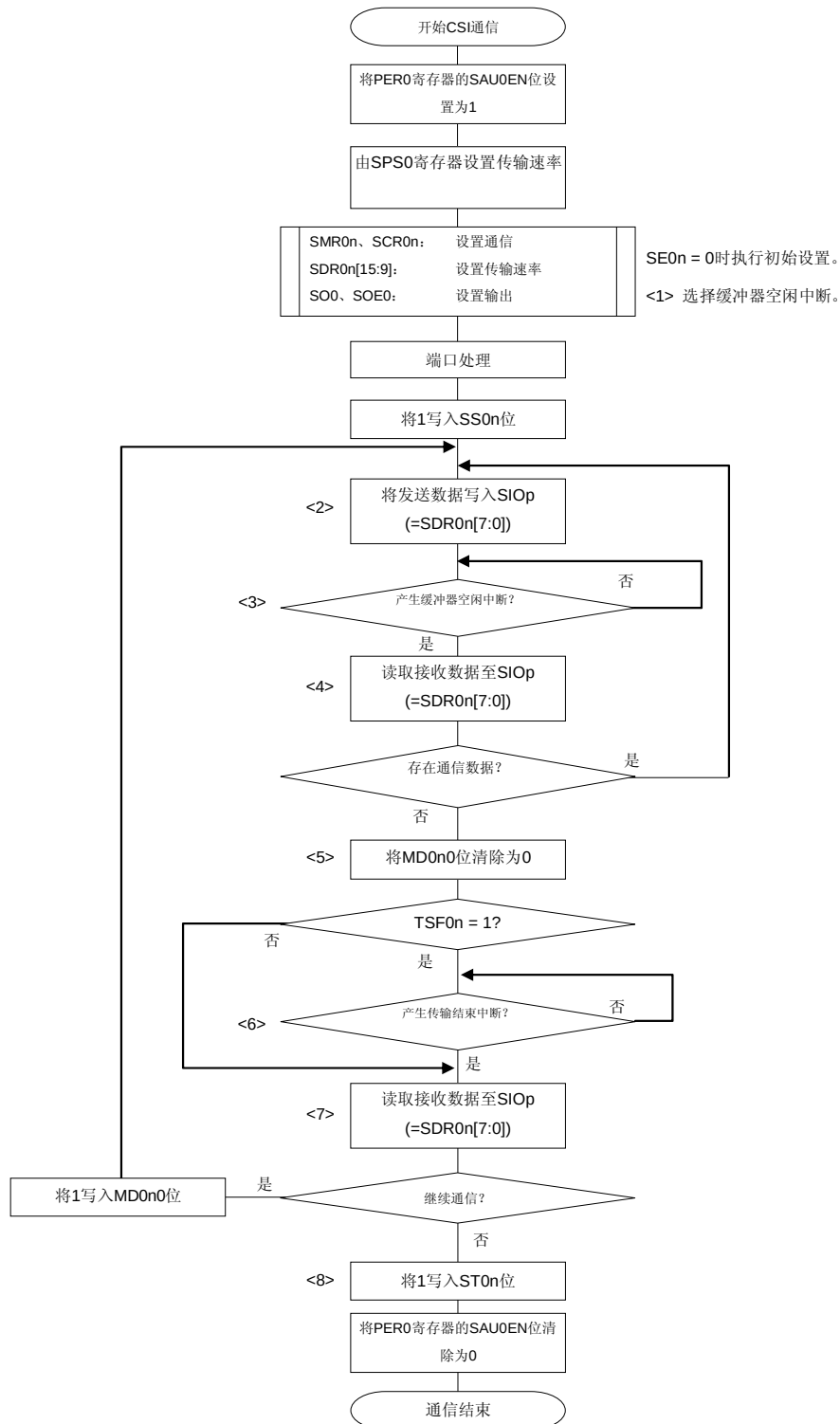


- 注 1. 当在 BFF0n = 1 将发送数据写入 SDR0n 寄存器时，覆盖重写发送数据。
2. 在这个时期内通过读取 SDR0n 寄存器可以读取发送数据。此时，不影响传输操作。

注意事项 即使在操作期间，仍可重写 MD0n0 位。
但是，要在最后位的传输开始前对其进行重写，这样在最后发送数据的传输结束中断之前就能对其进行重写。

- 备注 1. 图中的<1>至<8>对应着图 12-66 从机发送/接收的流程图（连续发送/接收模式中）中的<1>至<8>。
2. n: 通道编号 (n = 0 至 2)
p: CSI 编号 (p = 00、01、10)

图 12-66. 从机发送/接收的流程图（连续发送/接收模式中）



注意事项 将 PER0 寄存器置为 1 后，经过 4 个或更多个时钟后务必设置 SPS0 寄存器。

备注 图中的<1>至<8>对应着图 12-65 从机发送/接收的时序图（连续发送/接收模式中）中的<1>至<8>。

12.5.7 计算传输时钟频率

3-线串行 I/O（CSI00、CSI01、CSI10）通信的传输时钟频率可以通过下列表达式计算得到。

(1) 主机

(传输时钟频率) = {目标通道的操作时钟（MCK）频率} ÷ (SDR0n[15:9] + 1) ÷ 2 [Hz]

(2) 从机

(传输时钟频率) = {主机提供的串行时钟(SCK)频率}^注 [Hz]

注 可允许的最大频率为 f_{CLK}/6 [MHz] 和 f_{MCK}/2 [MHz]中的较小者。

备注 SDR0n[15:9]的值是指 SDR0n 寄存器位 15 至位 9 的值（0000000B 至 1111111B），也就是 0 至 127。

操作时钟（MCK）由串行时钟选择寄存器 0（SPS0）和串行模式寄存器 0n（SMR0n）的位 15（CKS0n）确定。

表 12-2. 选择操作时钟

SMR0n 寄存器	SPS0 寄存器								操作时钟 (MCK) ^{注1}	
CKS0n	PRS 013	PRS 012	PRS 011	PRS 010	PRS 003	PRS 002	PRS 001	PRS 000		f _{CLK} = 20 MHz
0	X	X	X	X	0	0	0	0	f _{CLK}	20 MHz
	X	X	X	X	0	0	0	1	f _{CLK} /2	10 MHz
	X	X	X	X	0	0	1	0	f _{CLK} /2 ²	5 MHz
	X	X	X	X	0	0	1	1	f _{CLK} /2 ³	2.5 MHz
	X	X	X	X	0	1	0	0	f _{CLK} /2 ⁴	1.25 MHz
	X	X	X	X	0	1	0	1	f _{CLK} /2 ⁵	625 kHz
	X	X	X	X	0	1	1	0	f _{CLK} /2 ⁶	313 kHz
	X	X	X	X	0	1	1	1	f _{CLK} /2 ⁷	156 kHz
	X	X	X	X	1	0	0	0	f _{CLK} /2 ⁸	78.1 kHz
	X	X	X	X	1	0	0	1	f _{CLK} /2 ⁹	39.1 kHz
	X	X	X	X	1	0	1	0	f _{CLK} /2 ¹⁰	19.5 kHz
	X	X	X	X	1	0	1	1	f _{CLK} /2 ¹¹	9.77 kHz
	X	X	X	X	1	1	1	1	INTTM02 ^{注2}	
1	0	0	0	0	X	X	X	X	f _{CLK}	20 MHz
	0	0	0	1	X	X	X	X	f _{CLK} /2	10 MHz
	0	0	1	0	X	X	X	X	f _{CLK} /2 ²	5 MHz
	0	0	1	1	X	X	X	X	f _{CLK} /2 ³	2.5 MHz
	0	1	0	0	X	X	X	X	f _{CLK} /2 ⁴	1.25 MHz
	0	1	0	1	X	X	X	X	f _{CLK} /2 ⁵	625 kHz
	0	1	1	0	X	X	X	X	f _{CLK} /2 ⁶	313 kHz
	0	1	1	1	X	X	X	X	f _{CLK} /2 ⁷	156 kHz
	1	0	0	0	X	X	X	X	f _{CLK} /2 ⁸	78.1 kHz
	1	0	0	1	X	X	X	X	f _{CLK} /2 ⁹	39.1 kHz
	1	0	1	0	X	X	X	X	f _{CLK} /2 ¹⁰	19.5 kHz
	1	0	1	1	X	X	X	X	f _{CLK} /2 ¹¹	9.77 kHz
	1	1	1	1	X	X	X	X	INTTM02 ^{注2}	
其它									禁止设置	

注 1. 当改变为 f_{CLK} 所选的时钟（通过改变系统时钟控制寄存器（CKC）的值）时，在停止（ST0 = 000FH）串行阵列部件（SAU）的操作之后进行。为操作时钟选择了 INTTM02 时，也会停止定时器阵列部件（TT0 = 00FFH）。

2. 不管 f_{CLK} 频率（主系统时钟、副系统时钟）如何，通过将 TAUS 的 TIS0 寄存器的 TIS02 位设置为 1，并为输入时钟选择 f_{SUB}/4 和使用 SPS0 寄存器选择 INTTM02，在副系统时钟的固定分频比处操作 SAU。但是，改变 f_{CLK} 时，必须按上述注 1 中所述停止 SAU 和 TAUS。

备注 1. X: 可忽略

2. n: 通道编号 (n = 0 至 2)

12.6 UART（UART0、UART1）通信的操作

这是开始-停止同步功能，使用两条信号线：串行数据发送（TxD）线和串行数据接收（RxD）线。它与通信方同步发送或接收数据（使用一个内部波特率）。通过使用两个通道可以实现全双工 UART 通信，一个专用于发送（偶数通道），另一个用于接收（奇数通道）。

[数据发送/接收]

- 5、7 或 8 位数据长度
- 选择 MSB/LSB 先行
- 发送/接收数据的电平设置和反相选择
- 附加奇偶校验位和奇偶校验功能
- 附加停止位

[中断功能]

- 传输结束中断/缓冲器空闲中断
- 帧错误、奇偶校验错误或超限错误发生时的错误中断

[错误检测标志]

- 帧错误、奇偶校验错误或超限错误

在 UART0 中支持 LIN 总线（部件的 0、1 通道）

[LIN 总线功能]

- 唤醒信号检测
- 同步中断域(SBF)检测
- 测量同步域、计算波特率

} 使用外部中断（INTP0）或者定时器阵列部件 TAUS。

UART0 使用 SAU 的通道 0 和 1。

UART1 使用 SAU 的通道 2 和 3。

通道	用作CSI	用作 UART	用作简易的I ² C
0	CSI00	UART0 (支持 LIN总线)	—
1	CSI01		—
2	CSI10	UART1	IIC10
3	—		—

UART 执行以下四种通信操作。

- UART 发送 (参见 12.6.1.)
- UART 接收 (参见 12.6.2.)
- LIN 发送（仅 UART0） (参见 12.6.3.)
- LIN 接收（仅 UART0） (参见 12.6.4.)

12.6.1 UART 发送

UART 发送是指将数据从 78K0R/Kx3-L 不同步地发送至其它设备的操作（开始-停止同步）。
有两条通道用于 UART，其中的偶通道用于 UART 发送。

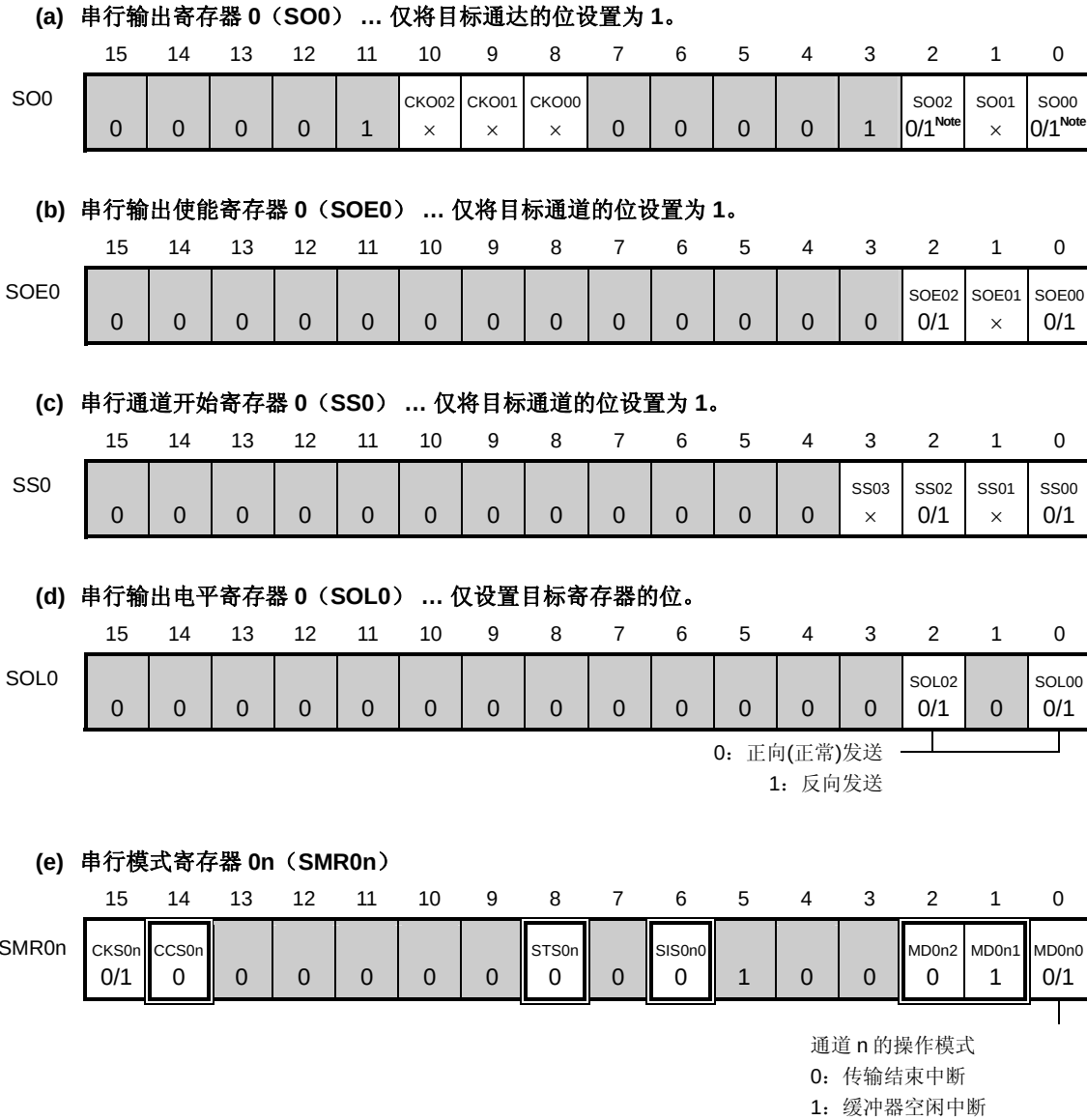
UART	UART0	UART1
目标通道	SAU的通道0	SAU的通道2
所用引脚	TxD0	TxD1
中断	INTST0	INTST1
	可以选择传输结束中断（单次传输模式中）或者缓冲器空闲中断（连续传输模式中）。	
错误检测标志	无	
传输数据长度	5、7或8位	
传输速率	最大值 $f_{MCK}/6$ [bps] (SDR0n [15:9] = 2 或以上)，最小值 $f_{CLK}/(2 \times 2^{11} \times 128)$ [bps] [‡]	
数据相位	正向输出(默认：高电平) 反向输出(默认：低电平)	
奇偶校验位	有以下选择 • 无校验位 • 附加0校验 • 附件偶校验 • 附加奇校验	
停止位	有以下选择 • 附加1位 • 附加2位	
数据方向	MSB或LSB先行	

注 在满足以上条件和电气规范中的 AC 特性的范围内使用该操作（参见第二十八章 电气规范（目标））。

备注 1. f_{MCK} : 目标通道的操作时钟（MCK）频率
 f_{CLK} : 系统时钟频率
 2. n: 通道编号 (n = 0、2)

(1) 寄存器设置

图 12-67. UART 的 UART 发送寄存器的内容的示例
(UART0、UART1) (1/2)



注 发送开始前，当目标通道的 SOL0n 位设为 0 时务必将其置为 1，当目标通道的 SOL0n 位设为 1 时务必将其置为 0。根据通信操作期间通信数据的不同，值也会不同。

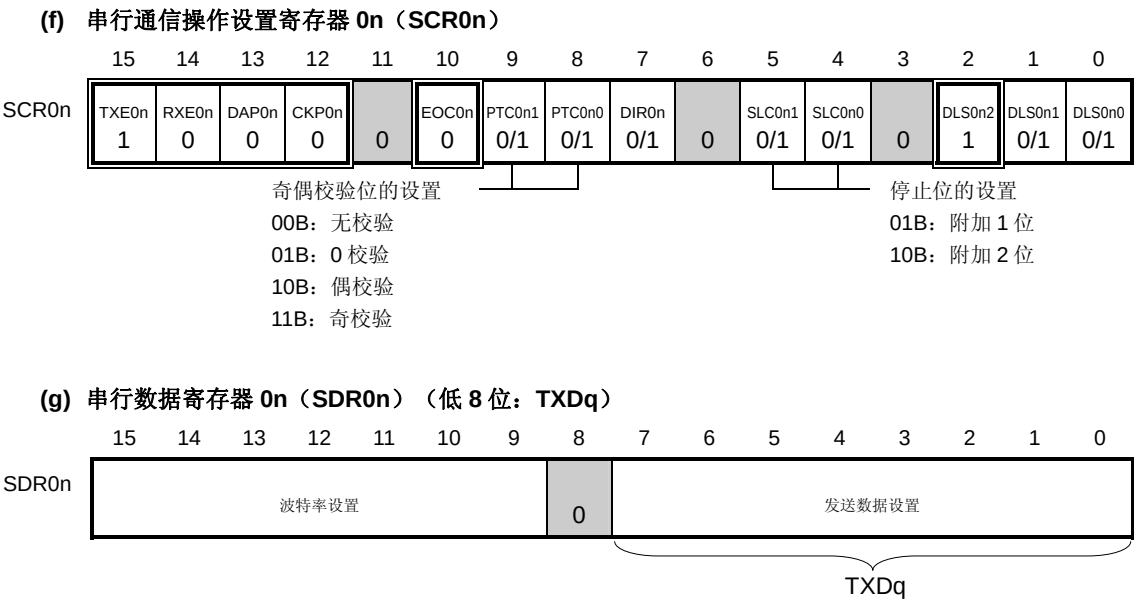
备注 n: 通道编号 (n = 0、2)

□: 在 UART 发送模式中设置固定, ■: 禁止设置 (由硬件固定)

×: 该模式中不能使用的位 (在任何模式中均不能使用时设为初始值)

0/1: 置为 0 或 1 取决于用户的用途

图 12-67. UART 的 UART 发送寄存器的内容的示例
(UART0、UART1) (2/2)



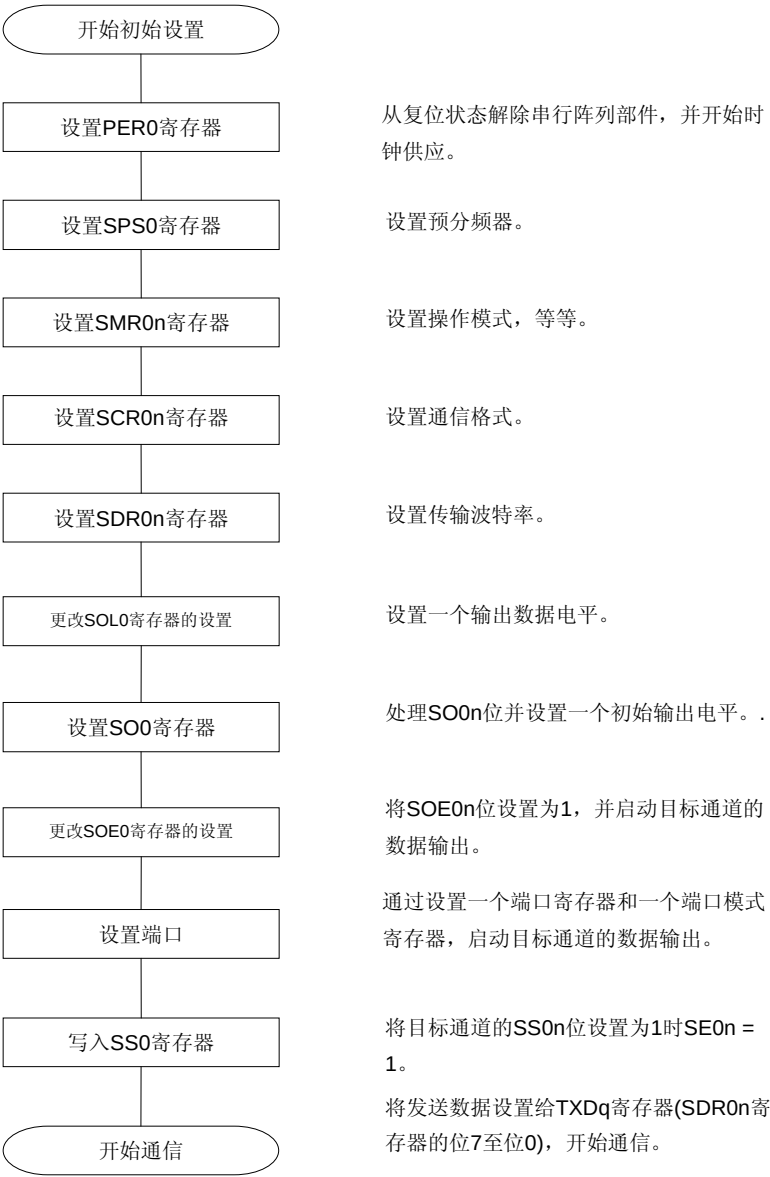
备注 n: 通道编号 (n = 0、2), q: UART 编号(q = 0、1)

☐: 在 UART 发送模式中设置固定, ☐: 禁止设置 (设为初始值)

0/1: 置为 0 或 1 取决于用户的用途

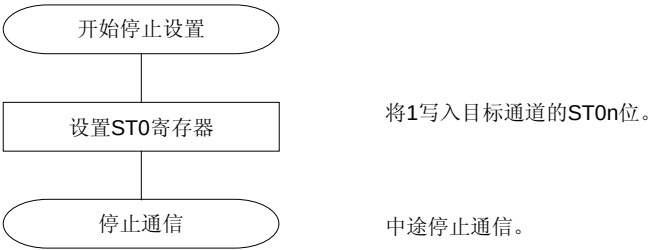
(2) 操作步骤

图 12-68. UART 发送的初始设置步骤



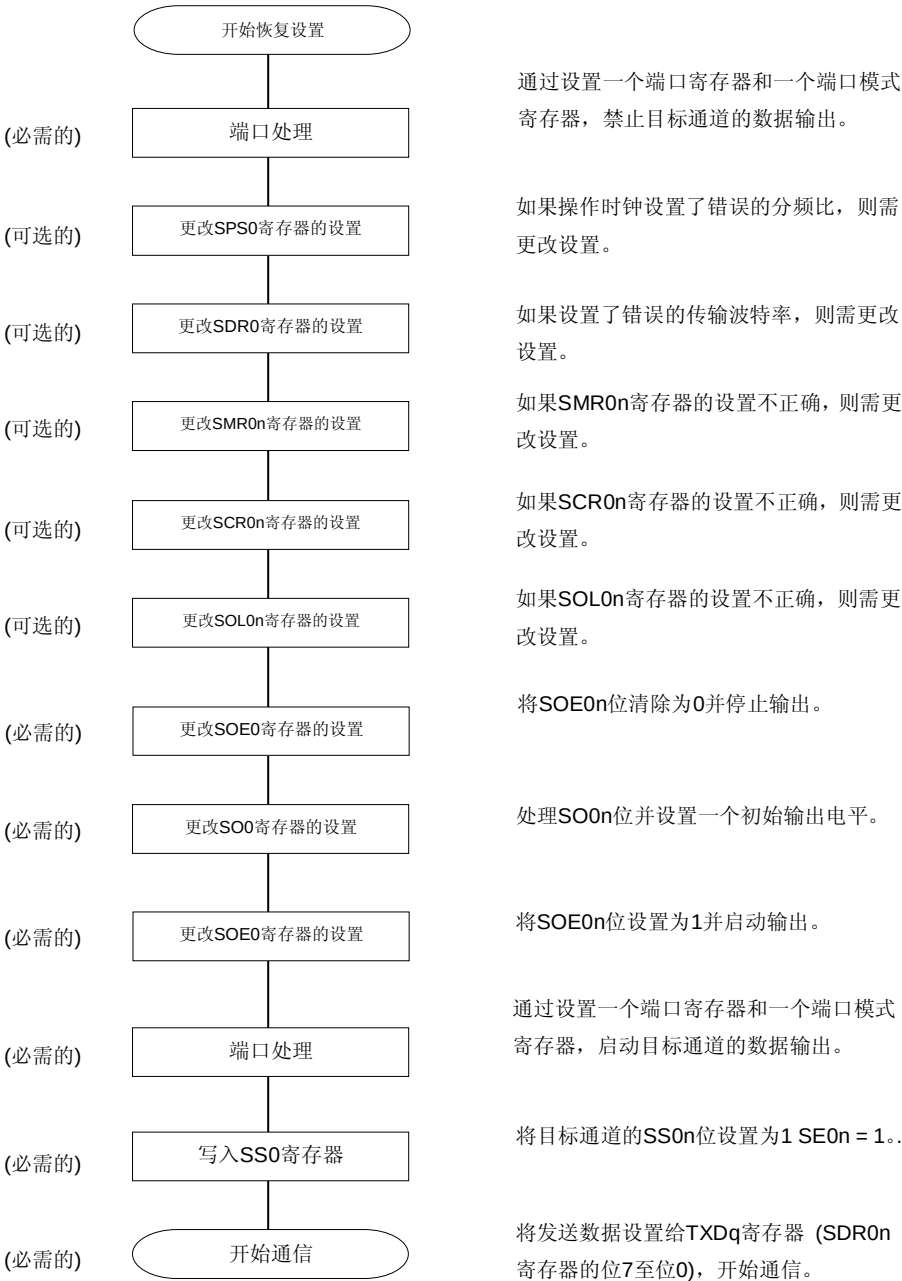
注意事项 将 PER0 寄存器置为 1 后，经过 4 个或更多个时钟后务必设置 SPS0 寄存器。

图 12-69. 停止 UART 发送的步骤



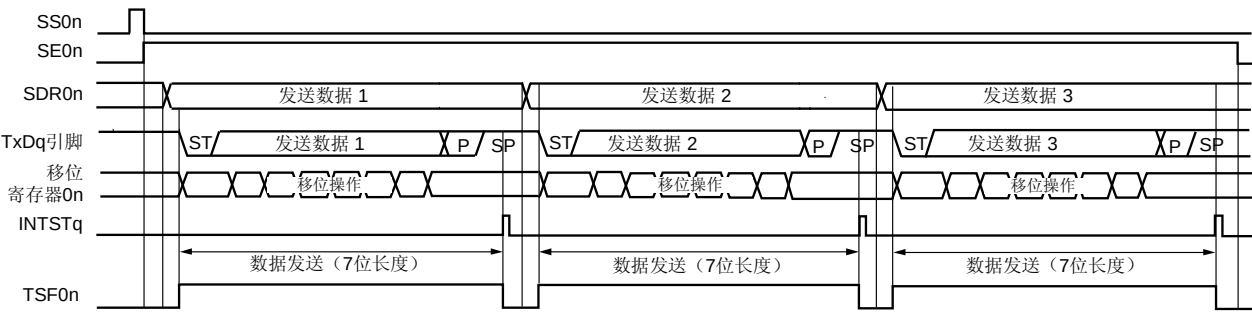
备注 即使在通信停止之后，仍保留引脚的电平。要恢复操作，需重新设置 SO0 寄存器（参见图 12-70 恢复 UART 发送的步骤）。

图 12-70. 恢复 UART 发送的步骤



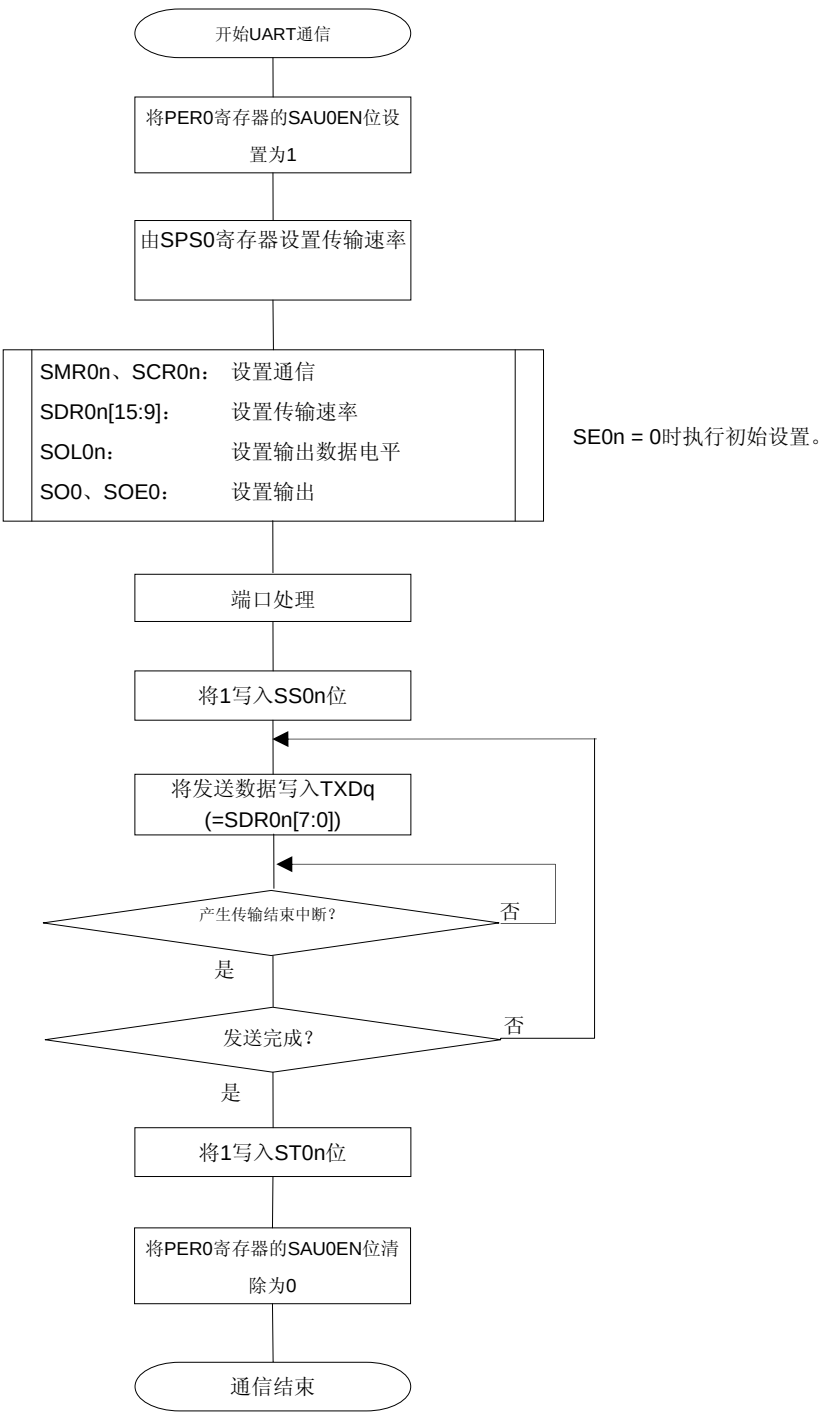
(3) 处理流程（单次发送模式中）

图 12-71. UART 发送的时序图（单次发送模式中）



备注 n: 通道编号 (n = 0、2), q: UART 编号(q = 0、1)

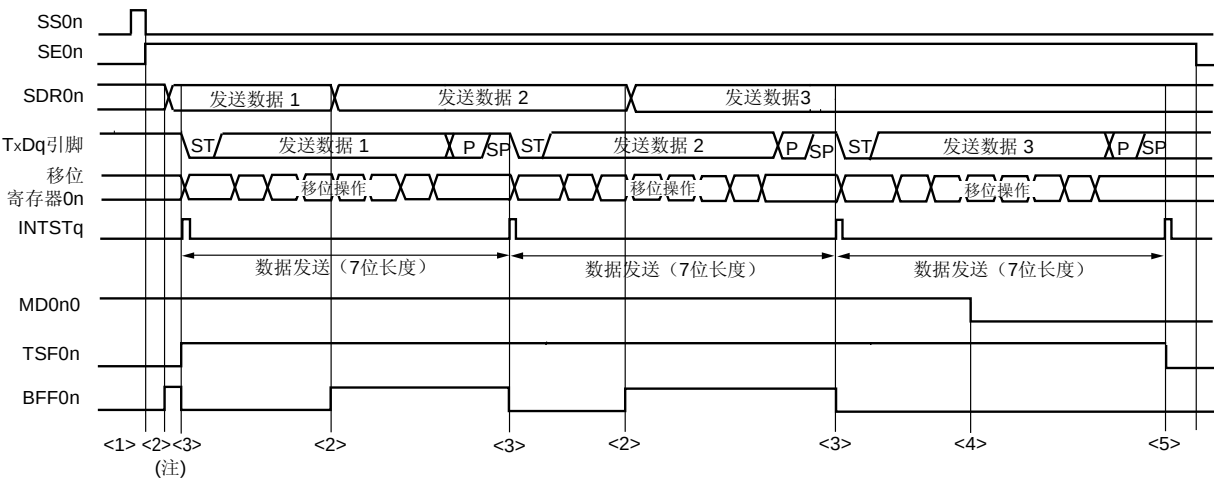
图 12-72. UART 发送的流程图（单次发送模式中）



注意事项 将 PER0 寄存器置为 1 后，经过 4 个或更多个时钟后务必设置 SPS0 寄存器。

(4) 处理流程（连续发送模式中）

图 12-73. UART 发送的时序图（连续发送模式中）

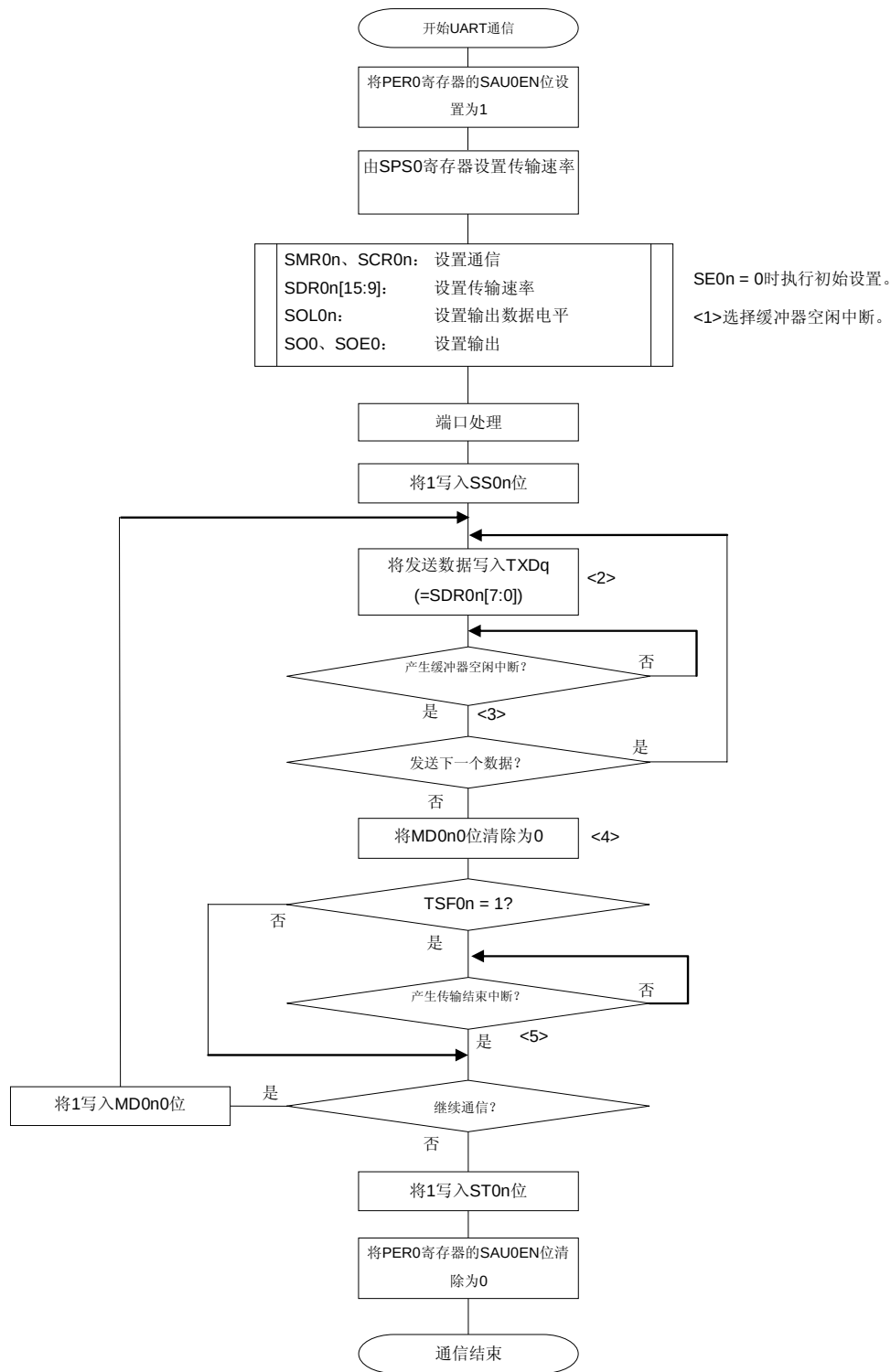


注 当在 BFF0n = 1 将发送数据写入 SDR0n 寄存器时，覆盖重写发送数据。

注意事项 即使在操作期间，仍可重写 MD0n0 位。
但是，要在最后位的传输开始前对其进行重写，这样在最后发送数据的传输结束中断之前就已经对其进行了重写。

备注 n: 通道编号 (n = 0、2)，q: UART 编号(q = 0、1)

图 12-74. UART 发送的流程图（连续发送模式中）



注意事项 将 PER0 寄存器置为 1 后，经过 4 个或更多个时钟后务必设置 SPS0 寄存器。

备注 图中的<1>至<5>对应着图 12-73 UART 发送的时序图（连续发送模式中）中的<1>至<5>。

12.6.2 UART 接收

UART 接收是指在 78K0R/Kx3-L 中不同步地接收来自其它设备的数据的操作（开始-停止同步）。

UART 所用的两个通道中的奇通道用于 UART 接收。

UART	UART0	UART1
目标通道	SAU的通道1	SAU的通道3
所用引脚	RxD0	RxD1
中断	INTSR0	INTSR1
	仅传输结束中断(禁止设置缓冲器空闲中断。)	
错误中断	INTSRE0	INTSRE1
错误检测标志	• 帧错误检测标志(FEF0n) • 奇偶校验错误检测标志(PEF0n) • 超限错误检测标志(OVF0n)	
传输数据长度	5、7或8位	
传输速率	最大值 $f_{MCK}/6$ [bps] ($SDR0n$ [15:9] = 2 或以上), 最小值 $f_{CLK}/(2 \times 2^{11} \times 128)$ [bps] [※]	
数据相位	正向输出(默认: 高电平) 反向输出(默认: 低电平)	
奇偶校验位	有以下选择 • 无校验位(无奇偶校验) • 附加0校验 (无奇偶校验) • 附加偶校验 • 附加奇校验	
停止位	附加1位	
数据方向	MSB或LSB先行	

注 在满足以上条件和电气规范中的 AC 特性的范围内使用该操作（参见第二十八章 电气规范（目标））。

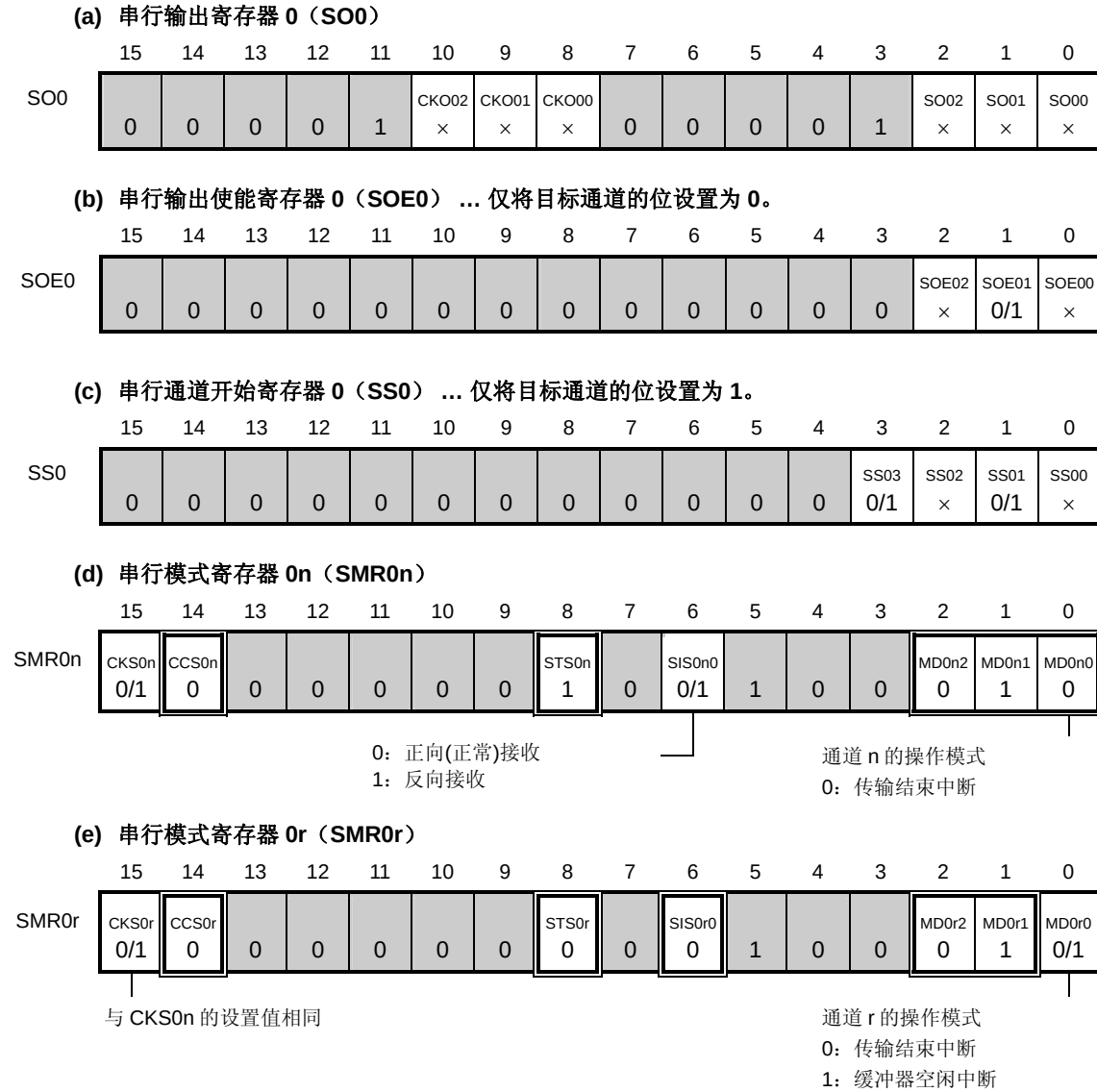
备注 1. f_{MCK} : 目标通道的操作时钟（MCK）频率

f_{CLK} : 系统时钟频率

2. n : 通道编号 ($n = 1、3$)

(1) 寄存器设置

图 12-75. UART 的 UART 接收寄存器的内容的示例
(UART0、UART1) (1/2)



注意事项 对于 UART 接收，务必要设置通道 r 的 SMR0r，其中通道 r 与通道 n 成对使用。

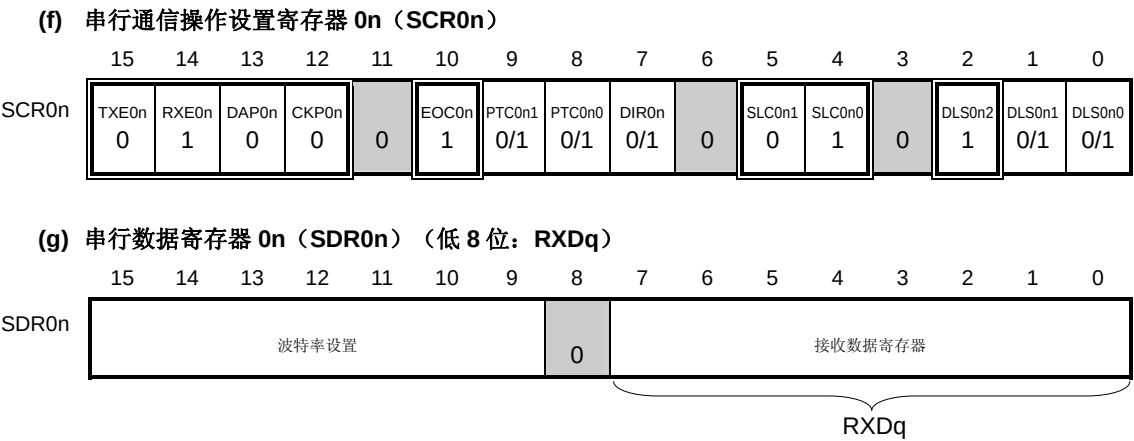
备注 n: 通道编号 (n = 1、3), r: 通道编号 (r = n - 1)

□: 在 UART 接收模式中设置固定, ■: 禁止设置(设为初始值)

×: 该模式中不能使用的位 (在任何模式中均不能使用时设为初始值)

0/1: 置为 0 或 1 取决于用户的用途

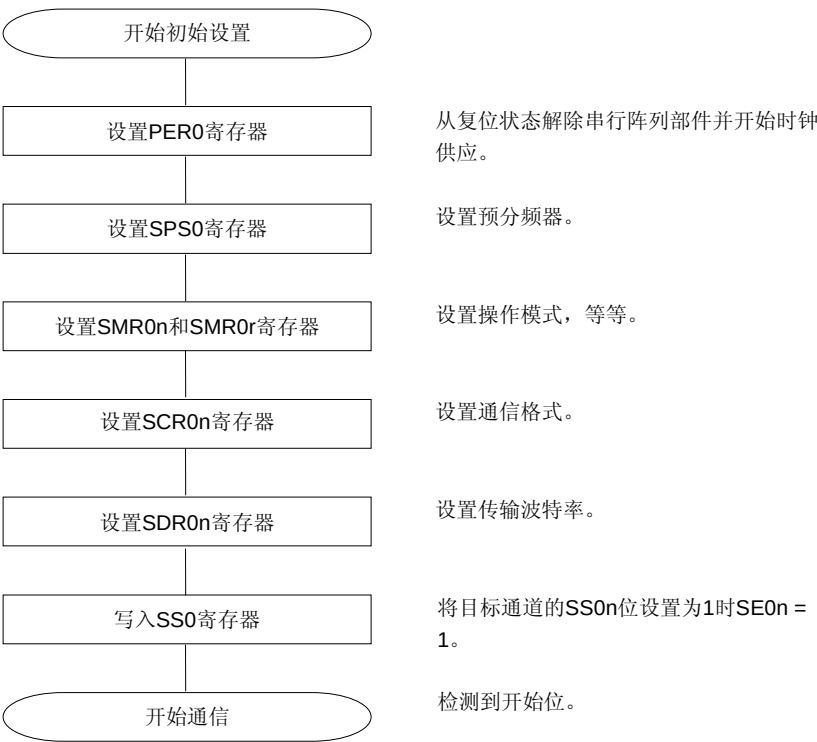
图 12-75. UART 的 UART 接收寄存器的内容的示例
(UART0、UART1) (2/2)



备注 n: 通道编号 (n = 1、3), q: UART 编号(q = 0、1)
□: 在 UART 接收模式中设置固定, ■: 禁止设置(设为初始值)
0/1: 置为 0 或 1 取决于用户的用途

(2) 操作步骤

图 12-76. UART 接收的初始设置步骤



注意事项 将 PER0 寄存器置为 1 后，经过 4 个或更多个时钟后务必设置 SPS0 寄存器。

图 12-77. 停止 UART 接收的步骤

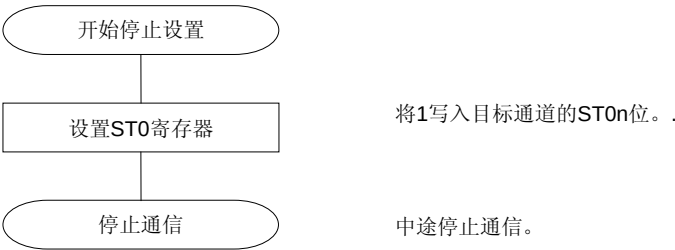
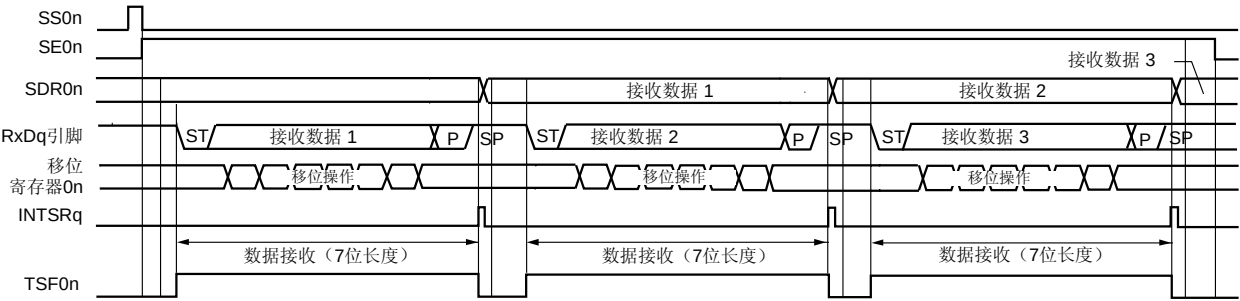


图 12-78. 恢复 UART 接收的步骤



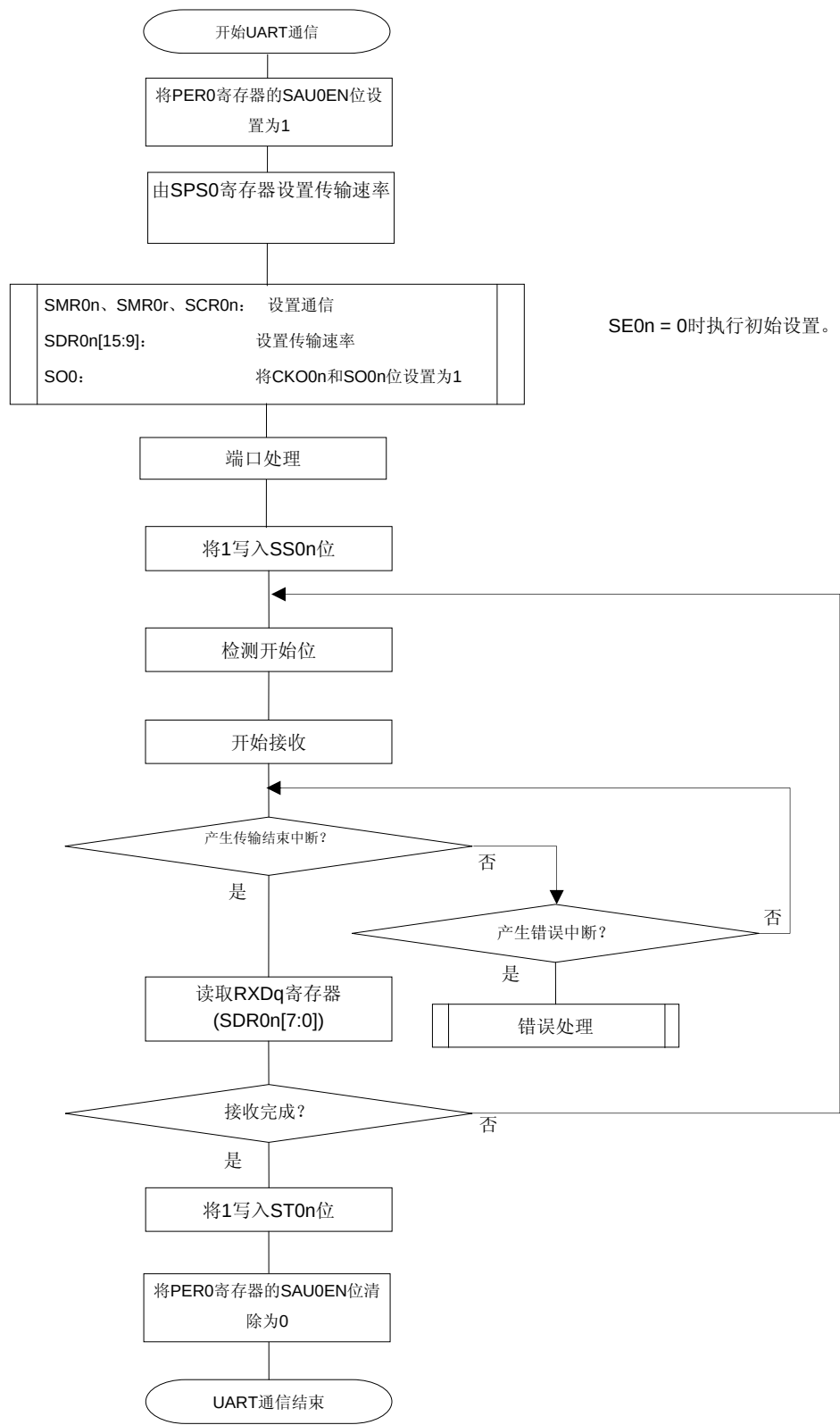
(3) 处理流程

图 12-79. UART 接收的时序图



备注 n: 通道编号 (n = 1、3), q: UART 编号(q = 0、1)

图 12-80. UART 接收流程图



注意事项 将 PER0 寄存器置为 1 后，经过 4 个或更多个时钟后务必设置 SPS0 寄存器。

12.6.3 LIN 发送

UART 发送中，UART0 支持 LIN 通信。
LIN 发送使用部件（SAU）的通道 0。

UART	UART0	UART1
LIN通信的支持情况	受支持	不受支持
目标通道	SAU的通道0	—
所用引脚	TxD0	—
中断	INTST0	—
	可以选择传输结束中断（单次传输模式中）或者缓冲器空闲中断（连续传输模式中）。	
错误检测标志	无	
传输数据长度	8位	
传输速率	最大值 $f_{MCK}/6$ [bps] (SDR00 [15:9] = 2或以上)，最小值 $f_{CLK}/(2 \times 2^{11} \times 128)$ [bps] ^注	
数据相位	正向输出 (默认：高电平) 反向输出 (默认：低电平)	
奇偶校验位	有以下选择 • 无校验位 • 附加0校验 • 附加偶校验 • 附加奇校验	
停止位	有以下选择 • 附加1位 • 附加2位	
数据方向	MSB或LSB先行	

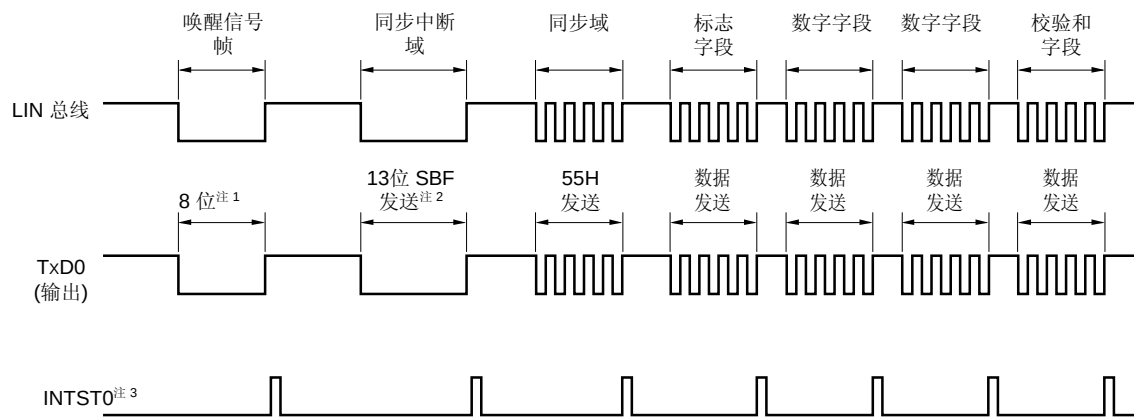
注 在满足以上条件和电气规范中的 AC 特性的范围内使用该操作（参见第二十八章 电气规范（目标））。

备注 f_{MCK} : 目标通信的操作时钟（MCK）频率
 f_{CLK} : 系统时钟频率

LIN 表示局部互联网，是一种低速（1 至 20 kbps）串行通信协议，是为减少汽车网络的成本而设计的。
LIN 通信为单主机通信，一台主机最多可连接 15 台从机。
从机用于控制开关、传动装置和传感器，这些从机通过 LIN 网与主机连接。
通常，主机接入诸如 CAN（区域控制网）之类的网络。
一个 LIN 总线是单线总线，节点遵从 ISO9141 通过收发器接入。
根据 LIN 协议，主机发送附带有波特率信息的一帧。从机接收该帧，并校正来自主机的波特率误差。如果一个从机的波特率误差在±15%以内，则可以确定通信。

图 12-81 给出了 LIN 的发送操作的简图。

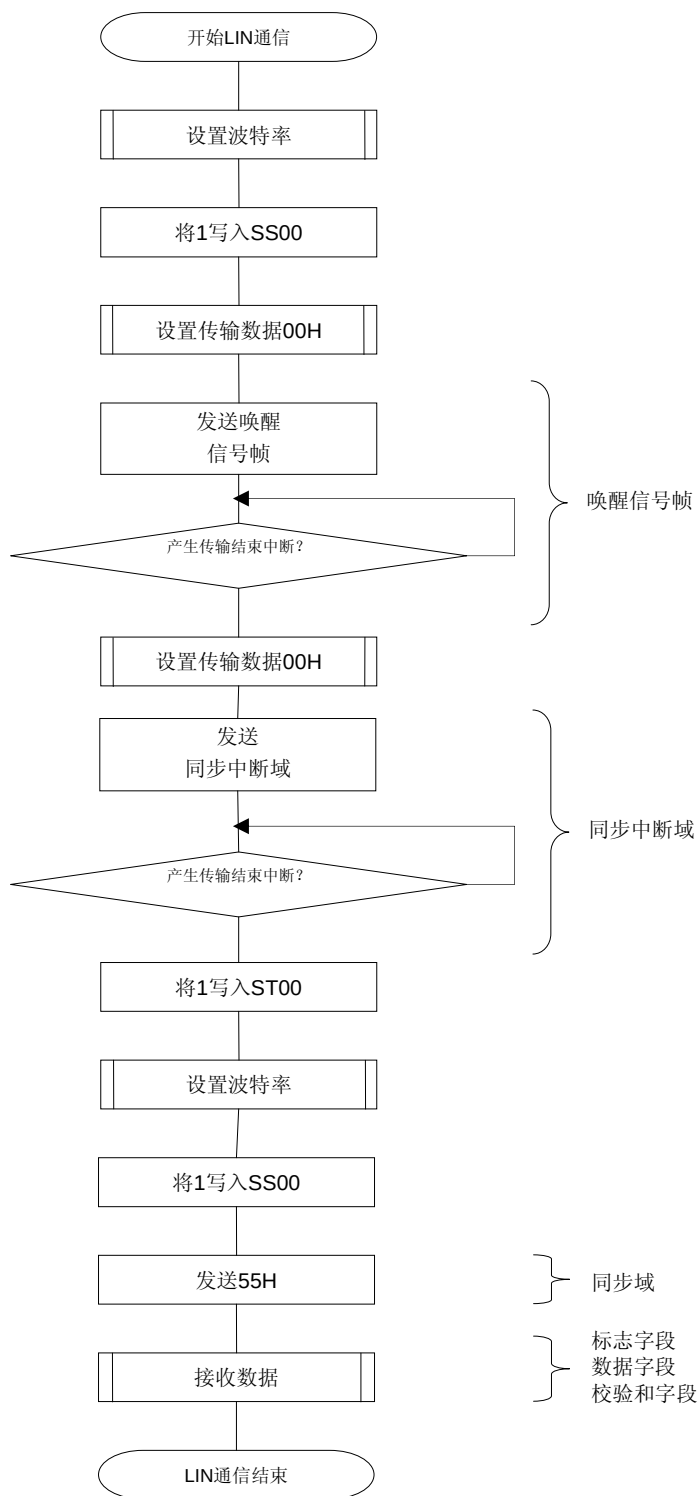
图 12-81. LIN 的发送操作



- 注**
1. 设置波特率，以便于满足唤醒信号的标准，并发送 00H 数据。
 2. 同步中断域的宽度为 13 位，能输出一个低电平。因此，主机的波特率为 N [bps]时，同步中断域的波特率计算如下。
(同步中断域的波特率) = $9/13 \times N$
以这个波特率发送 00H 数据，就会产生同步中断域。
 3. 发送完成时输出 INTST0。执行 SBF 发送时也会输出 INTST0。

备注 域与域之间的间隔由软件控制。

图 12-82. LIN 发送的流程图



12.6.4 LIN 接收

UART 接收中，UART0 支持 LIN 通信。
LIN 接收使用部件（SAU）的通道 1。

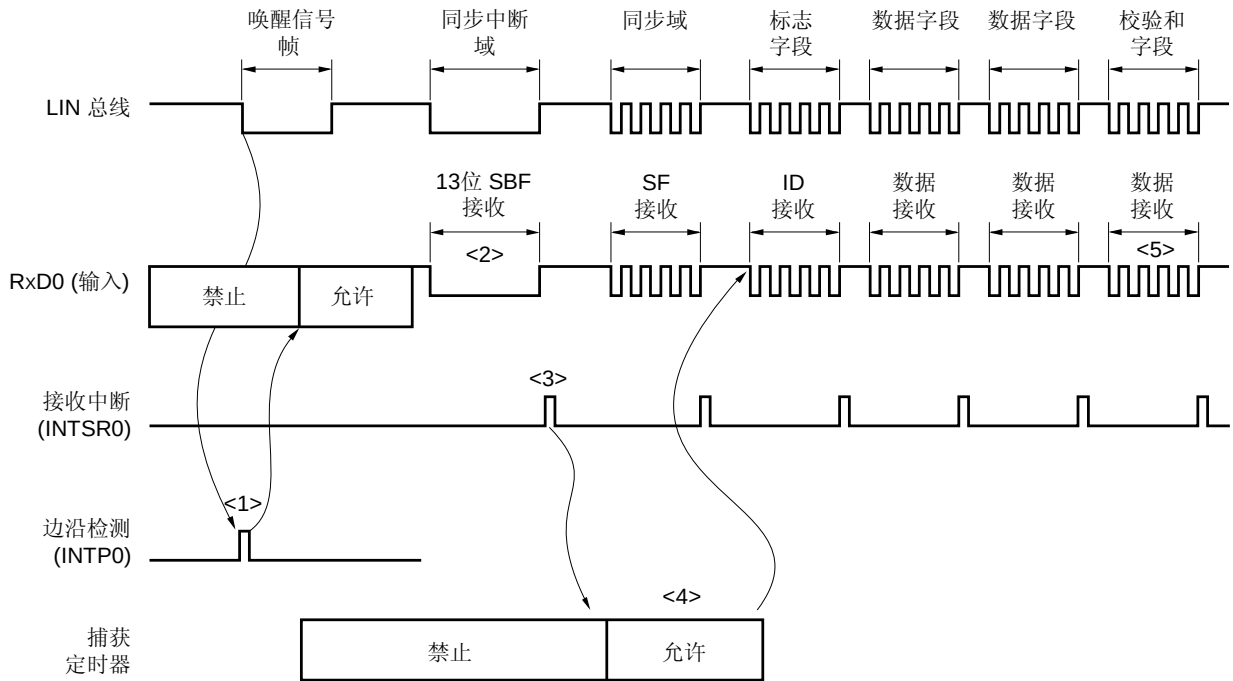
UART	UART0	UART1
LIN通信的支持情况	受支持	不受支持
目标通道	SAU的通道1	—
所用引脚	RxD0	—
中断	INTSR0	—
	仅传输结束中断(禁止设置缓冲器空闲中断。)	
错误中断	INTSRE0	—
错误检测标志	• 帧错误检测标志(FEF01) • 奇偶校验错误检测标志(PEF01) • 超限错误检测标志(OVF01)	
传输数据长度	8位	
传输速率	最大值 $f_{MCK}/6$ [bps] (SDR01 [15:9] = 2或以上), 最小值 $f_{CLK}/(2 \times 2^{11} \times 128)$ [bps] ^注	
数据相位	正向输出 (默认: 高电平) 反向输出 (默认: 低电平)	
奇偶校验位	有以下选择 • 无校验位 • 附加0校验 • 附加偶校验 • 附加奇校验	
Stop bit	有以下选择 • 附加1位 • 附加2位	
数据方向	MSB或LSB先行	

注 在满足以上条件和电气规范中的 AC 特性的范围内使用该操作（参见第二十八章 电气规范（目标））。

备注 f_{MCK} : 目标通道的操作时钟（MCK）频率
 f_{CLK} : 系统时钟频率

图 12-83 给出了 LIN 接收操作的简图。

图 12-83. LIN 的接收操作



以下是信号处理的流程。

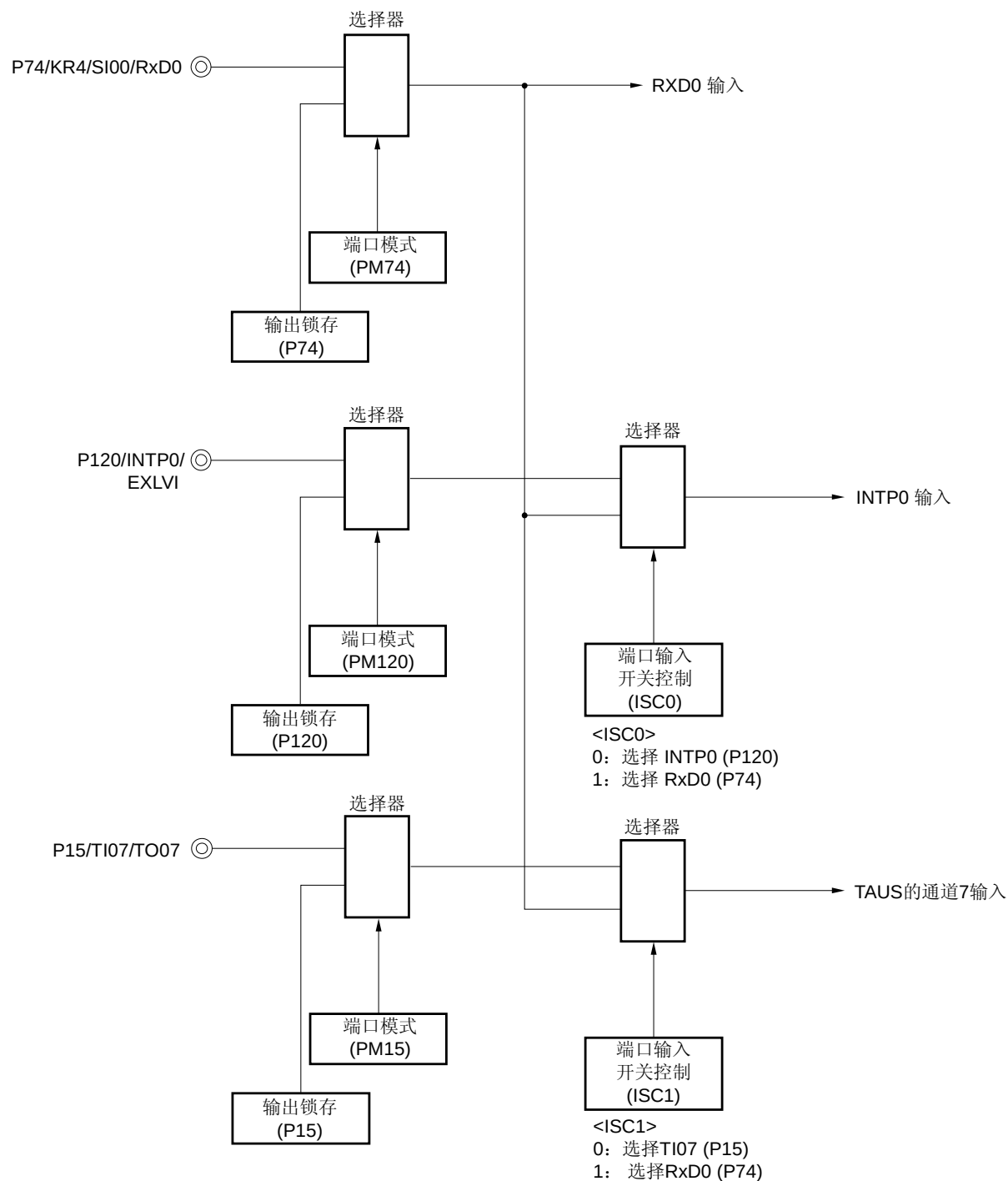
- <1> 通过检测引脚的中断边沿 (INTPO) 来检测唤醒信号。检测到唤醒信号时，允许 UART0 (RXE01 = 1) 的接收，并等待 SBF 接收。
- <2> 检测到 SBF 的开始位时，开始接收，继而按设置的波特率将串行数据存储在 RxD0 寄存器 (=串行数据寄存器 01 (SDR01) 的位 7 至位 0) 中。检测到停止位时，产生接收结束中断请求 (INTSR0)。当检测到 11 位或更多位的低等级数据作为 SBF 时，则判断 SBF 接收已正确完成。如果检测到少于 11 位的低等级数据作为 SBF，则判断 SBF 接收已发生错误，且系统返回 SBF 接收等待状态。
- <3> 当 SBF 接收已正确完成时，开始定时器阵列部件 TAUS 的通道 7，并测量同步域的位间隔 (脉冲宽度) (参见 6.7.5 测量输入信号高-/低电平宽度的操作)。
- <4> 计算来自同步域 (SF) 位间隔的波特率误差。停止 UART0 一次，调整 (重新设置) 波特率。
- <5> 校验和字段应由软件识别。此外，初始化接收到校验和字段之后的 UART0 和等待 SBF 接收的处理也应该由软件执行。

图 12-84 显示了处理 LIN 接收的端口的配置。

通过检测外部中断（INTP0）的边沿，接收来自 LIN 主机发送的唤醒信号。可以使用定时器阵列部件 TAUS 的外部事件捕获操作来测量从主机发送的同步域的长度，以计算波特率误差。

通过控制端口输入（ISC0/ISC1）的转换，可以将接收端口输入（RxD0）的输入源输入至外部中断引脚（INTP0）和定时器阵列部件 TAUS。

图 12-84. 处理 LIN 接收的端口配置



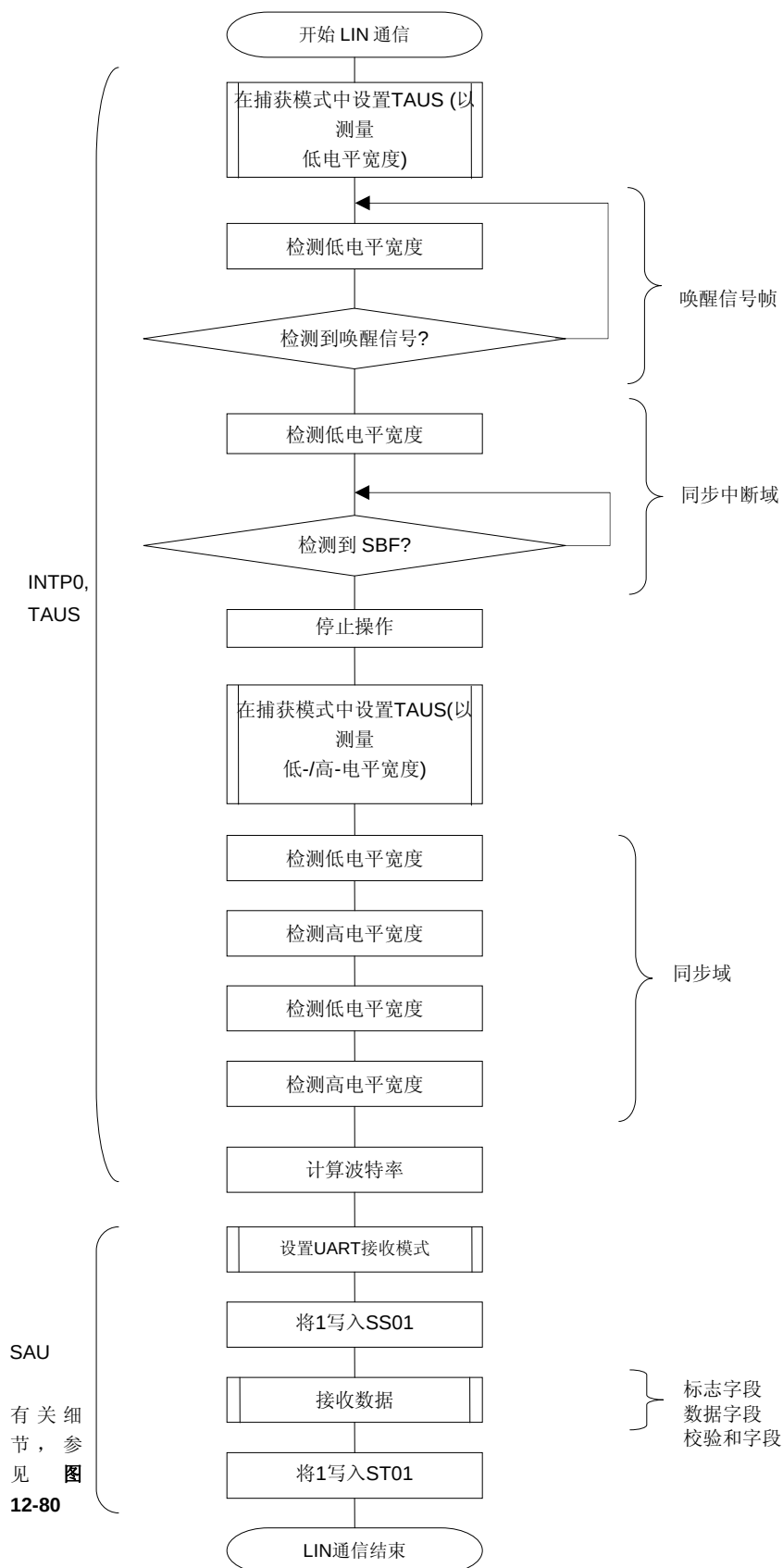
备注 INSC0、ISC1: 输入开关控制寄存器（ISC）的位 0 和位 1（参见图 12-16。）

用于 LIN 通信操作的外设功能如下所示。

<所用的外设功能>

- 外部中断（INTP0）；唤醒信号检测
用途： 检测唤醒信号的边沿和通信的开始
- 定时器阵列部件 TAUS 的通道 7；波特率误差检测
用途： 检测同步域（SF）的长度并将由位数其划分，一边检测误差（在捕获模式中测量输入至 RxD0 的边沿间隔。）
- 串行阵列部件（SAU）的通道 0 和 1（UART0）

图 12-85. LIN 接收的流程图



12.6.5 计算波特率

(1) 波特率计算表达式

UART (UART0、UART1) 通信的波特率可以由下列表达式计算得出。

$$(\text{波特率}) = \{ \text{目标通道的操作时钟 (MCK) 频率} \} \div (\text{SDR0n}[15:9] + 1) \div 2 \text{ [bps]}$$

注意事项 禁止设置 $\text{SDR0n}[15:9] = (0000000\text{B}、0000001\text{B})$ 。

- 备注**
1. 使用 UART 时, $\text{SDR0n}[15:9]$ 的值是指 SDR0n 寄存器位 15 至位 9 的值 (0000010B 至 1111111B), 也就是 2 至 127。
 2. n: 通道编号 (n = 0 至 3)

操作时钟 (MCK) 由串行时钟选择寄存器 0 (SPS0) 和串行模式寄存器 0n (SMR0n) 的位 15 (CKS0n) 控制。

表 12-3. 操作时钟的选择

SMR0n 寄存器	SPS0 寄存器								操作时钟(MCK) ^{注1}	
CKS0n	PRS 013	PRS 012	PRS 011	PRS 010	PRS 003	PRS 002	PRS 001	PRS 000		f _{CLK} = 20 MHz
0	X	X	X	X	0	0	0	0	f _{CLK}	20 MHz
	X	X	X	X	0	0	0	1	f _{CLK} /2	10 MHz
	X	X	X	X	0	0	1	0	f _{CLK} /2 ²	5 MHz
	X	X	X	X	0	0	1	1	f _{CLK} /2 ³	2.5 MHz
	X	X	X	X	0	1	0	0	f _{CLK} /2 ⁴	1.25 MHz
	X	X	X	X	0	1	0	1	f _{CLK} /2 ⁵	625 kHz
	X	X	X	X	0	1	1	0	f _{CLK} /2 ⁶	313 kHz
	X	X	X	X	0	1	1	1	f _{CLK} /2 ⁷	156 kHz
	X	X	X	X	1	0	0	0	f _{CLK} /2 ⁸	78.1 kHz
	X	X	X	X	1	0	0	1	f _{CLK} /2 ⁹	39.1 kHz
	X	X	X	X	1	0	1	0	f _{CLK} /2 ¹⁰	19.5 kHz
	X	X	X	X	1	0	1	1	f _{CLK} /2 ¹¹	9.77 kHz
	X	X	X	X	1	1	1	1	INTTM02 ^{注2}	
1	0	0	0	0	X	X	X	X	f _{CLK}	20 MHz
	0	0	0	1	X	X	X	X	f _{CLK} /2	10 MHz
	0	0	1	0	X	X	X	X	f _{CLK} /2 ²	5 MHz
	0	0	1	1	X	X	X	X	f _{CLK} /2 ³	2.5 MHz
	0	1	0	0	X	X	X	X	f _{CLK} /2 ⁴	1.25 MHz
	0	1	0	1	X	X	X	X	f _{CLK} /2 ⁵	625 kHz
	0	1	1	0	X	X	X	X	f _{CLK} /2 ⁶	313 kHz
	0	1	1	1	X	X	X	X	f _{CLK} /2 ⁷	156 kHz
	1	0	0	0	X	X	X	X	f _{CLK} /2 ⁸	78.1 kHz
	1	0	0	1	X	X	X	X	f _{CLK} /2 ⁹	39.1 kHz
	1	0	1	0	X	X	X	X	f _{CLK} /2 ¹⁰	19.5 kHz
	1	0	1	1	X	X	X	X	f _{CLK} /2 ¹¹	9.77 kHz
	1	1	1	1	X	X	X	X	INTTM02 ^{注2}	
其它									禁止设置	

- 注 1. 当改变为 f_{CLK} 所选的时钟（通过改变系统时钟控制寄存器（CKC）的值）时，在停止（ST0 = 000FH）串行阵列部件（SAU）的操作之后进行。为操作时钟选择了 INTTM02 时，也会停止定时器阵列部件（TT0 = 00FFH）。
2. 不管 f_{CLK} 频率（主系统时钟、副系统时钟）如何，通过将 TAUS 的 TIS0 寄存器的 TIS02 位设置为 1，并为输入时钟选择 f_{SUB}/4 和使用 SPS0 寄存器选择 INTTM02，在副系统时钟的固定分频比处操作 SAU。但是，改变 f_{CLK} 时，必须按上述注 1 中所述停止 SAU 和 TAUS。

备注 1. X: 可忽略

2. n: 通道编号 (n = 0 至 3)

(2) 发送期间的波特率误差

发送期间 UART（UART0、UART1）通信的波特率误差可以由下列表达式计算得出。确保发送端的波特率在接收端波特率的允许范围内。

$$(\text{波特率误差}) = (\text{计算出的波特率值}) \div (\text{目标波特率}) \times 100 - 100 [\%]$$

这是在 $f_{\text{CLK}} = 20 \text{ MHz}$ 处设置 UART 波特率的示例。

UART波特率 (目标波特率)	$f_{\text{CLK}} = 20 \text{ MHz}$			
	操作时钟 (MCK)	SDR0n[15:9]	计算出的波特率	来自目标波特率的误差
300 bps	$f_{\text{CLK}}/2^9$	64	300.48 bps	+0.16 %
600 bps	$f_{\text{CLK}}/2^8$	64	600.96 bps	+0.16 %
1200 bps	$f_{\text{CLK}}/2^7$	64	1201.92 bps	+0.16 %
2400 bps	$f_{\text{CLK}}/2^6$	64	2403.85 bps	+0.16 %
4800 bps	$f_{\text{CLK}}/2^5$	64	4807.69 bps	+0.16 %
9600 bps	$f_{\text{CLK}}/2^4$	64	9615.38 bps	+0.16 %
19200 bps	$f_{\text{CLK}}/2^3$	64	19230.8 bps	+0.16 %
31250 bps	$f_{\text{CLK}}/2^3$	39	31250.0 bps	$\pm 0.0 \%$
38400 bps	$f_{\text{CLK}}/2^2$	64	38461.5 bps	+0.16 %
76800 bps	$f_{\text{CLK}}/2$	64	76923.1 bps	+0.16 %
153600 bps	f_{CLK}	64	153846 bps	+0.16 %
312500 bps	f_{CLK}	31	312500 bps	$\pm 0.0 \%$

备注 n: 通道编号 (n = 0、2)

(3) 接收端可允许的波特率范围

UART（UART0、UART1）通信期间接收端可允许的波特率范围可以由下列表达式计算得出。确保发送端的波特率在接收端波特率的允许范围内。

(可接收的最大波特率) =
$$\frac{2 \times k \times \text{Nfr}}{2 \times k \times \text{Nfr} - k + 2} \times \text{Brate}$$

(可接收的最小波特率) =
$$\frac{2 \times k \times (\text{Nfr} - 1)}{2 \times k \times \text{Nfr} - k - 2} \times \text{Brate}$$

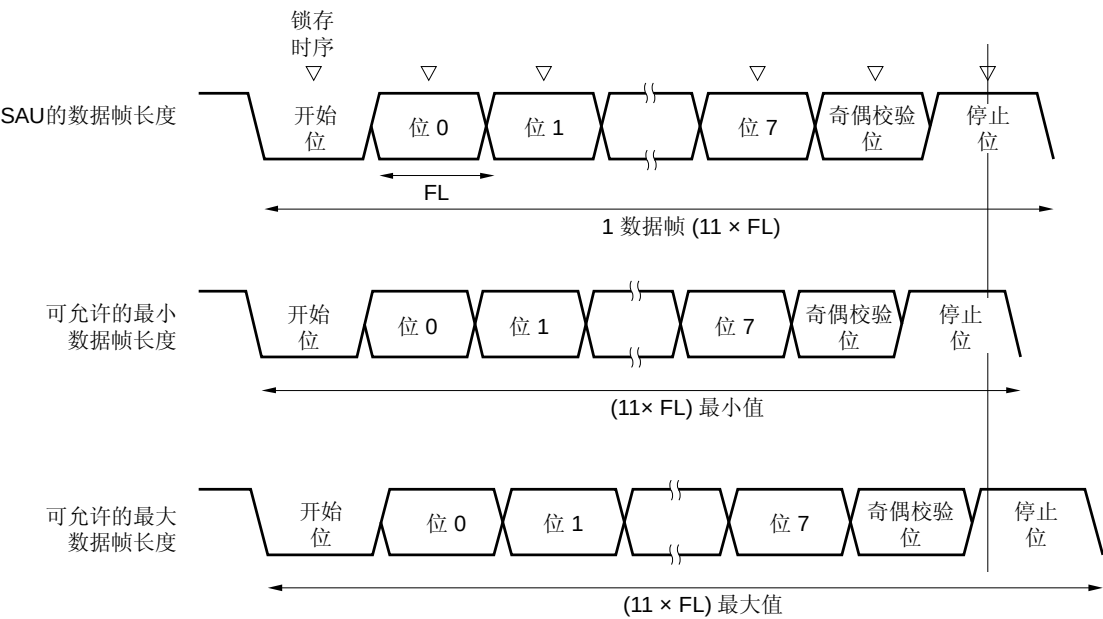
Brate: 在接收端计算出的波特率值（参见 12.6.5 (1) 波特率计算表达式。）

k: SDR0n[15:9] + 1

Nfr: 1 数据帧长度 [位]
= (开始位) + (数据长度) + (奇偶校验位) + (停止位)

备注 n: 通道编号 (n = 1、3)

图 12-86. 接收端可允许的波特率范围（1 数据帧长度 = 11 位）



如图 12-86 所示，在检测到开始位后，锁存接收数据的时序由分频比确定，该分频比由串行数据寄存器 0n（SDR0n）的位 15 至位 9 设置。如果在该锁存时序之前接收到最后的数据（停止位），则数据可以被正确接收。

12.7 简易的 I²C (IIC10) 通信的操作

这是用来与两个或多个设备进行通信的时钟通信功能，使用两条信号线：串行时钟（SCL）和串行数据（SDA）。该通信功能是为执行与诸如 EEPROM、flash 存储器和 A/D 转换器之类的设备进行单通信而设计的，因此，仅能由主机使用，而不具有等待检测功能。通过使用软件和操作控制寄存器，确保遵循开始和停止条件的 AC 指定。

[数据发送/接收]

- 主机发送、主机接收（仅使用单主机的主机功能）
- ACK 输出功能[※]和 ACK 检测功能
- 8 位的数据长度
（当发送地址时，由高 7 位指定地址，最低有效位用于 R/W 控制。）
- 开始条件和停止条件的手动产生

[中断功能]

- 传输结束中断

[错误检测标志]

- 奇偶校验错误（ACK 错误）

* [不受简易的 I²C 支持的功能]

- 从机发送、从机接收
- 仲裁失败检测功能
- 等待检测功能

注 当接收到最后的数据时，如果将 0 写入 SOE02（SOE0 寄存器）位，则将不会输出 ACK，且停止串行通信数据输出。更多细节参见 12.7.3 (2) 中的处理流程。

备注 要使用 I²C 总线的全部功能，参见第 13 章 串行接口 IICA。

支持简易的 I²C (IIC10) 的通道为 SAU 的通道 2。

通道	用作CSI	用作UART	用作简易的I ² C
0	CSI00	UART0 (支持LIN总线)	—
1	CSI01		—
2	CSI10	UART1	IIC10
3	—		—

简易的 I²C (IIC10) 执行以下四种通信操作。

- 地址域发送 (参见 12.7.1.)
- 数据发送 (参见 12.7.2.)
- 数据接收 (参见 12.7.3.)
- 产生停止条件 (参见 12.7.4.)

12.7.1 地址域发送

地址域发送是指在 I²C 通信中最初执行的以识别传输（从机）目标的发送操作。产生开始条件后，在一帧内发送地址（7 位）和传输方向（1 位）。

简易的I ² C	IIC10
目标通道	SAU的通道2
所用引脚	SCL10、SDA10
中断	INTIIC10
	仅传输结束中断(禁止设置缓冲器空闲中断。)
错误检测标志	奇偶校验错误检测标志(PEF02)
传输数据长度	8位(发送时，高7位指定为地址，最低有效位指定为R/W控制)
传输速率	最大值 f _{CLK} /4 MHz f _{CLK} ：系统时钟频率 但是，在I ² C的各模式中必须满足以下条件。 • 最大值 400 kHz (最初模式) • 最大值100 kHz (标准模式)
数据电平	正向输出 (默认：高电平)
奇偶校验位	无校验位
停止位	附加1位(适用于ACK接收时序)
数据方向	MSB先行

(1) 寄存器设置

图 12-87. 简易的 I²C (IIC10) 的地址域发送寄存器的内容的示例

(a) 串行输出寄存器 0 (SO0) ... 仅设置目标通道的位。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SO0	0	0	0	0	1	CKO02 0/1	CKO01 ×	CKO00 ×	0	0	0	0	1	SO02 0/1	SO01 ×	SO00 ×

通过处理 SO02 位产生开始条件。

(b) 串行输出使能寄存器 0 (SOE0) ... 仅设置目标通道的位。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOE0	0	0	0	0	0	0	0	0	0	0	0	0	0	SOE02 0/1	SOE01 ×	SOE00 ×

在产生开始条件之前 SOE02 = 0，在产生之后 SOE02 = 1。

(c) 串行通道开始寄存器 0 (SS0) ... 仅将目标通道的位设置为 1。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SS0	0	0	0	0	0	0	0	0	0	0	0	0	SS03 ×	SS02 0/1	SS01 ×	SS00 ×

(d) 串行模式寄存器 02 (SMR02)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SMR02	CKS02 0/1	CCS02 0						STS02 0		SIS020 0	1	0	0	MD022 1	MD021 0	MD020 0

通道 2 的操作模式

0: 传输结束中断

(e) 串行通信操作设置寄存器 02 (SCR02)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SCR02	TXE02 1	RXE02 0	DAP02 0	CKP02 0		EOC02 0	PTC021 0	PTC020 0	DIR02 0		SLC021 0	SLC020 1		DLS022 1	DLS021 1	DLS020 1

奇偶校验位的设置

00B: 无校验

停止位的设置

01B: 附加 1 位(ACK)

(f) 串行数据寄存器 02 (SDR02) (低 8 位: SIO10)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SDR02	波特率设置							0	发送数据设置(地址 + RW)							

SIO10

备注

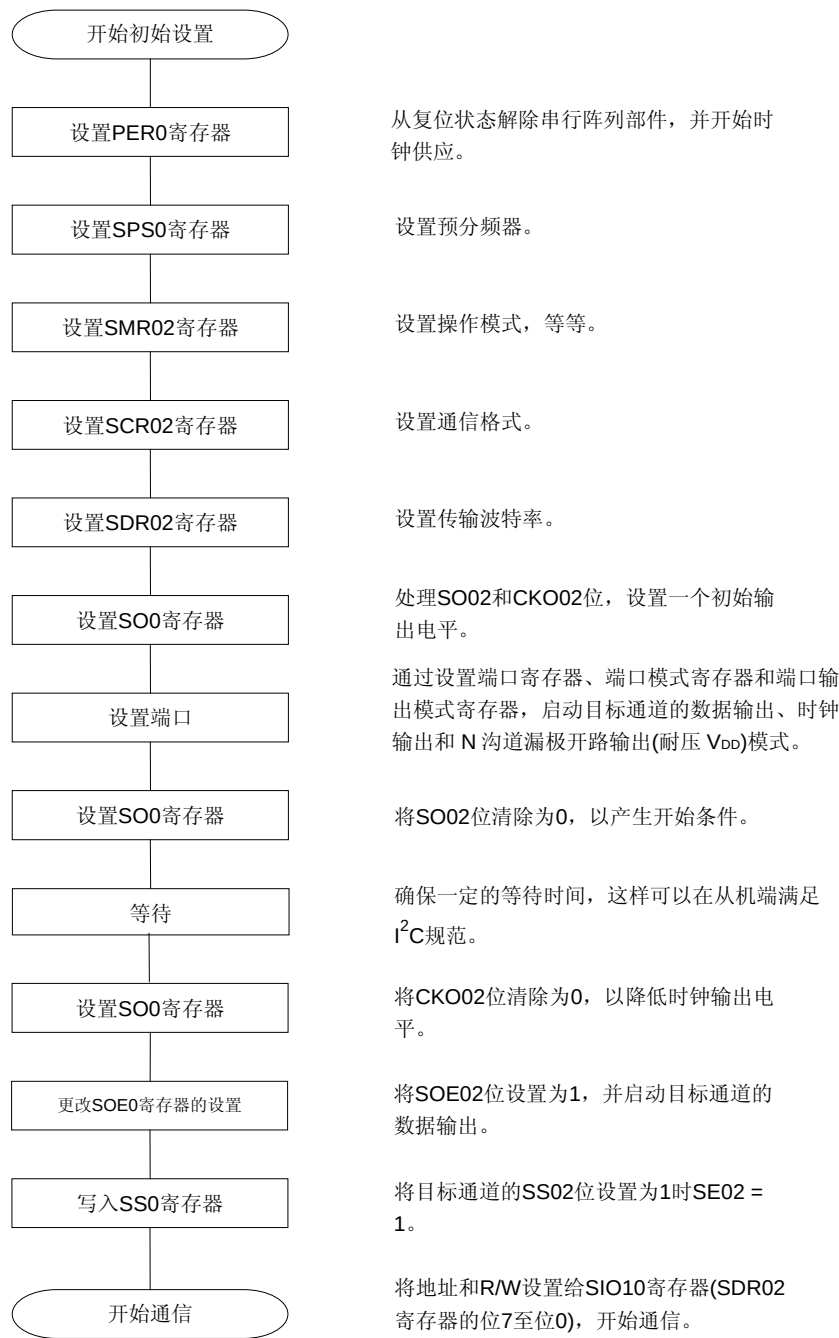
□: 在 IIC 模式中设置固定, ■: 禁止设置 (设为初始值)

×: 该模式中不能使用的位 (在任何模式中均不能使用时设为初始值)

0/1: 置为 0 或 1 取决于用户的用途

(2) 操作步骤

图 12-88. 地址域发送的初始设置步骤



注意事项 将 PER0 寄存器置为 1 后，经过 4 个或更多个时钟后务必设置 SPS0 寄存器。

(3) 处理流程

图 12-89. 地址域发送的时序图

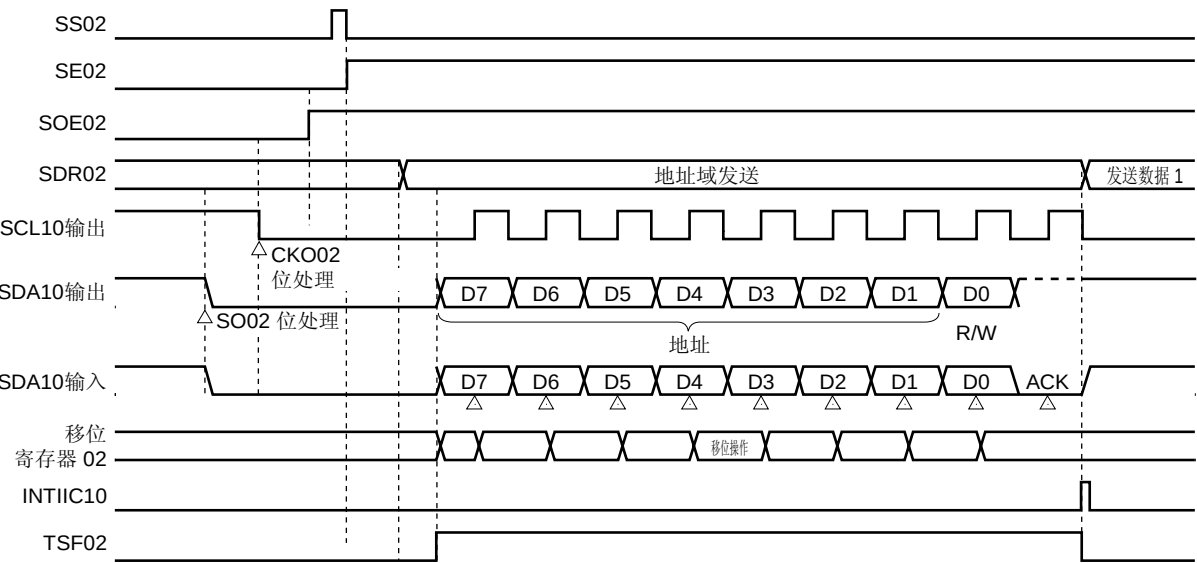
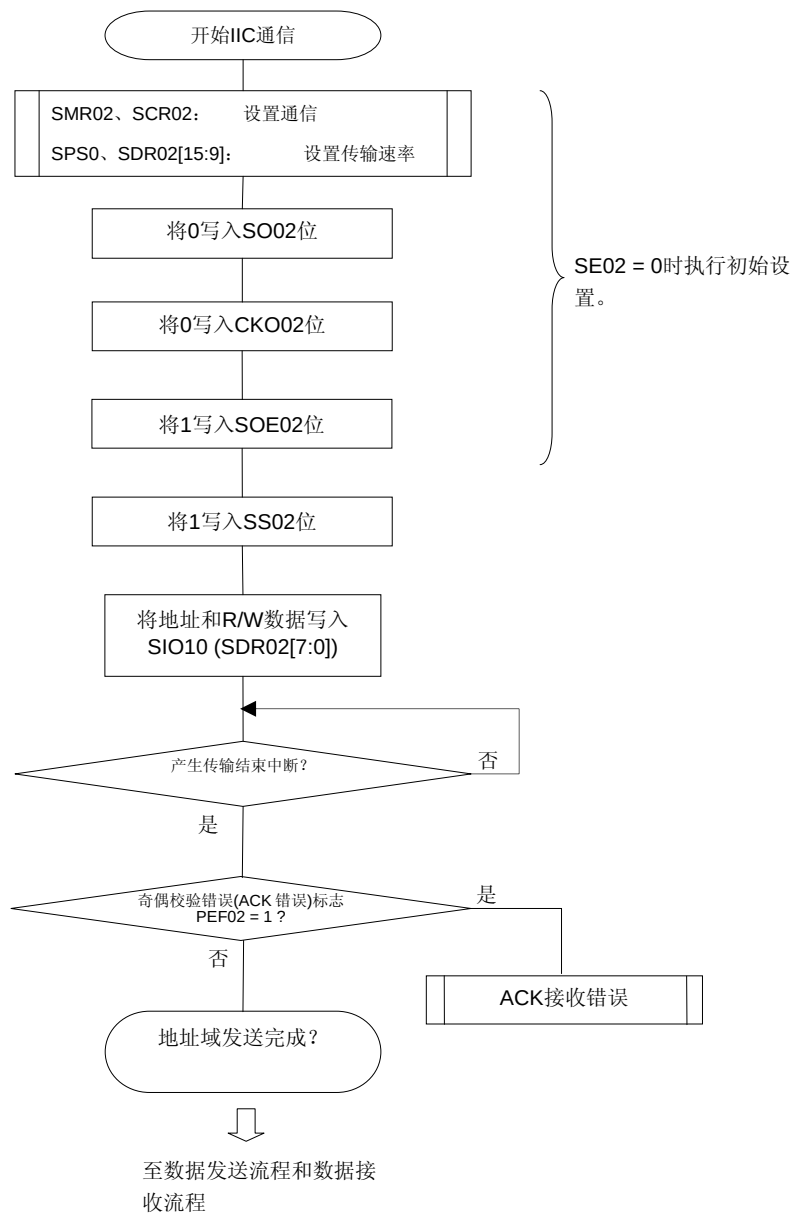


图 12-90. 地址域发送的流程图



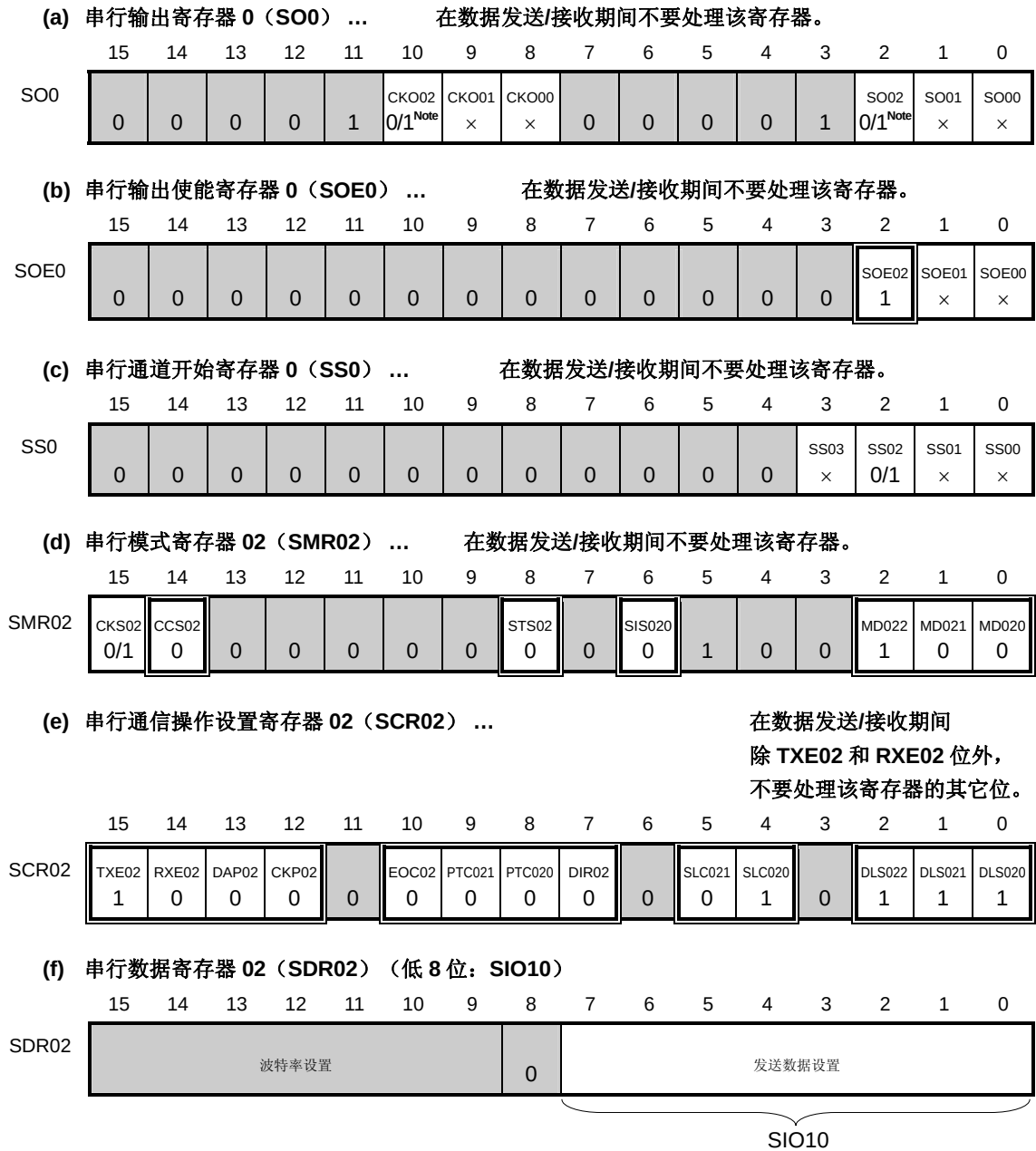
12.7.2 数据发送

数据发送是指在发送地址域之后将数据发送至传输（从机）目标的操作。在将所有数据发送至从机后，产生停止条件并解除总线。

简易的 I ² C	IIC10
目标通道	SAU 的通道2
所用引脚	SCL10、SDA10
中断	INTIIC10
	仅传输结束中断(禁止设置缓冲器空闲中断。)
错误检测标志	奇偶校验错误检测标志(PEF02)
传输数据长度	8位
传输速率	最大值 f _{CLK} /4 MHz f _{CLK} : 系统时钟频率 但是，在I ² C的各模式中必须满足以下条件。 • 最大值 400 kHz (最初模式) • 最大值100 kHz (标准模式)
数据电平	正向输出(默认: 高电平)
奇偶校验位	无校验位
停止位	附加1位(适用于ACK接收时序)
数据方向	MSB先行

(1) 寄存器设置

图 12-91. 简易的 I²C (IIC10) 的数据发送寄存器的内容的示例



注 根据通信操作期间通信数据的不同，值也会不同。

备注 ☐：在 IIC 模式中设置固定， ☐：禁止设置（设为初始值）
×：该模式中不能使用的位（在任何模式中均不能使用时设为初始值）
0/1：置为 0 或 1 取决于用户的用途

(2) 处理流程

图 12-92. 数据发送的时序图

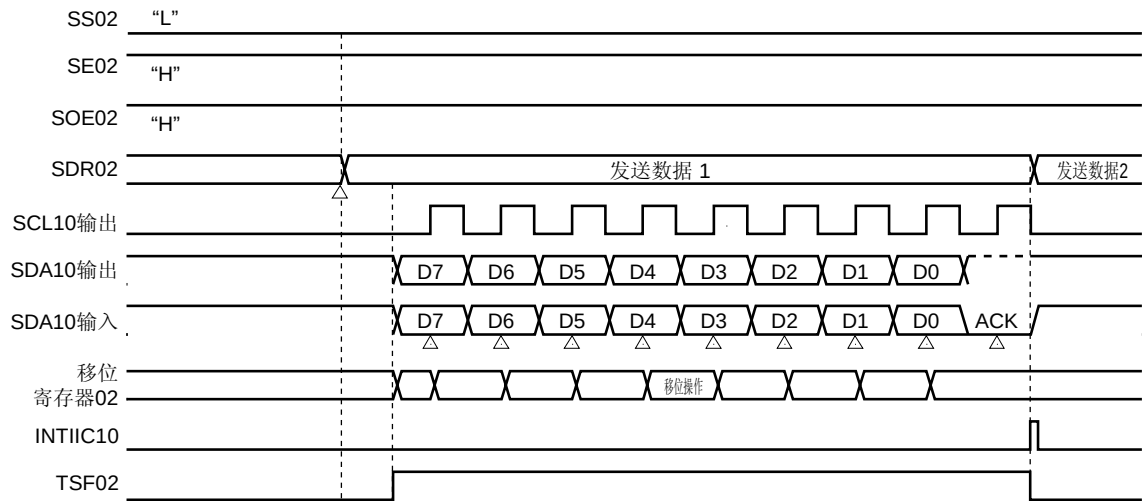
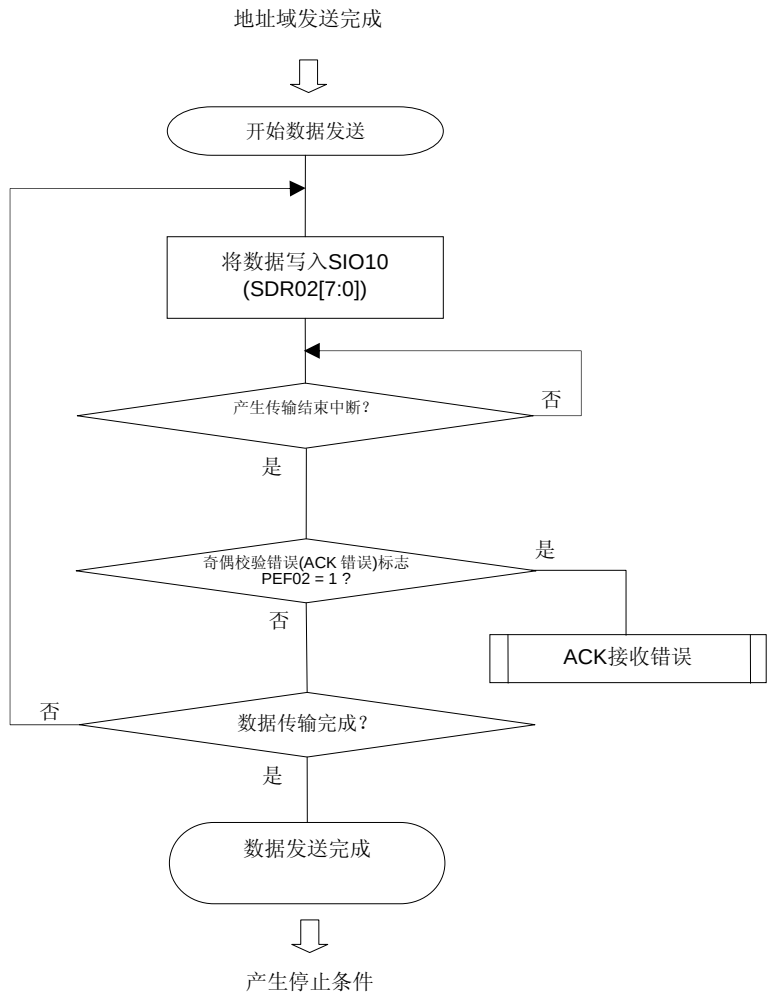


图 12-93. 数据发送的流程图



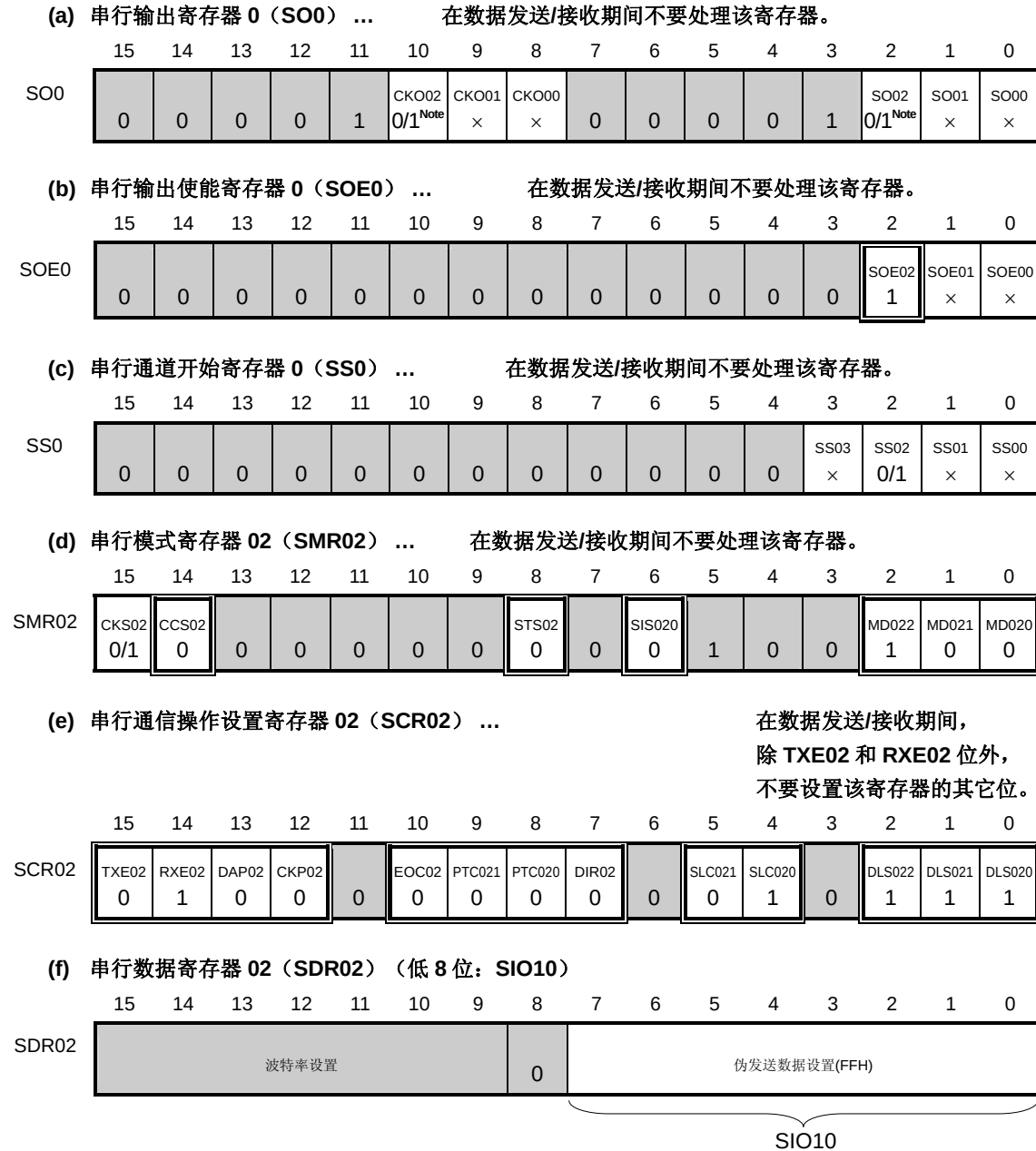
12.7.3 数据接收

数据接收是指在发送地址域之后将数据接收至传输（从机）目标的操作。在将所有数据接收至从机后，产生停止条件并解除总线。

简易的 I ² C	IIC10
目标通道	SAU的通道2
所用引脚	SCL10、SDA10
中断	INTIIC10
	仅传输结束中断(禁止设置缓冲器空闲中断。)
错误检测标志	无
传输数据长度	8位
传输速率	最大值, f _{CLK} /4 MHz f _{CLK} : 系统时钟频率 但是，在I ² C的各模式中必须满足以下条件。 • 最大值 400 kHz (最初模式) • 最大值100 kHz (标准模式)
数据电平	正向输出(默认: 高电平)
奇偶校验位	无校验位
停止位	附加1位(ACK发送)
数据方向	MSB先行

(1) 寄存器设置

图 12-94. 简易的 I²C (IIC10) 的数据接收寄存器的内容的示例



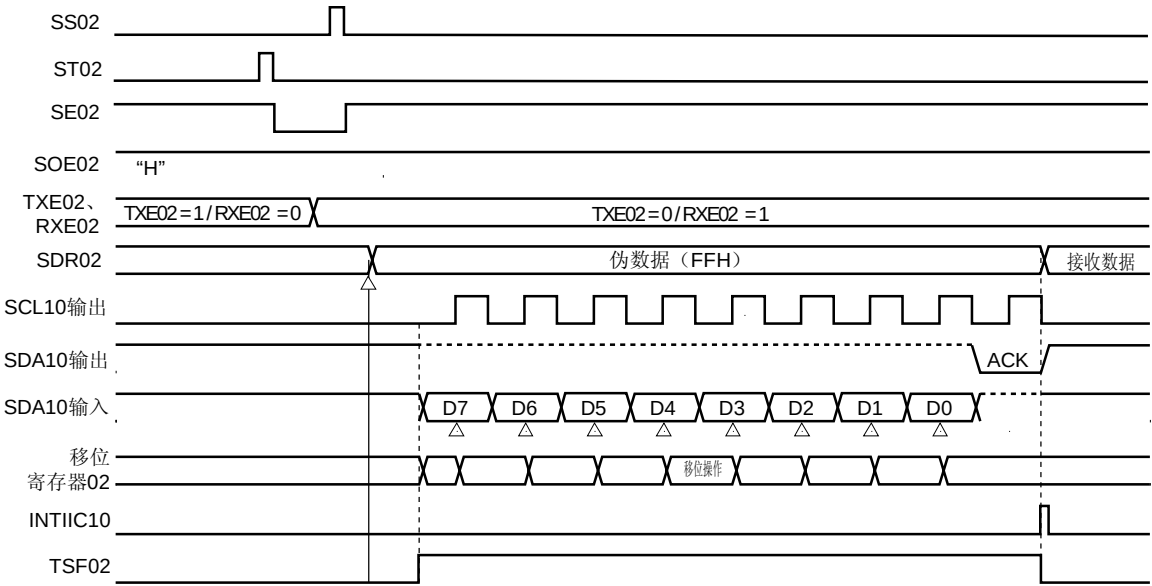
注 根据通信操作期间通信数据的不同，值也会不同。

备注 □: 在 IIC 模式中设置固定， ■: 禁止设置 (设为初始值)
 ×: 该模式中不能使用的位 (在任何模式中均不能使用时设为初始值)
 0/1: 置为 0 或 1 取决于用户的用途

(2) 处理流程

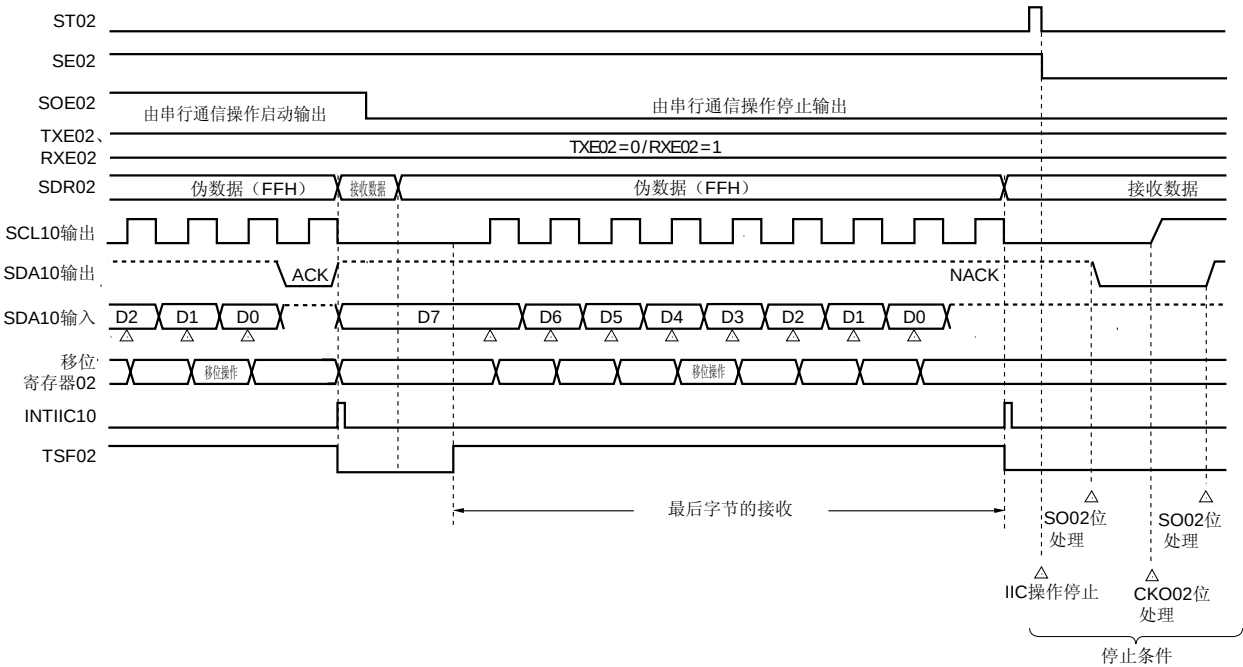
图 12-95. 数据接收的时序图

(a) 开始数据接收时



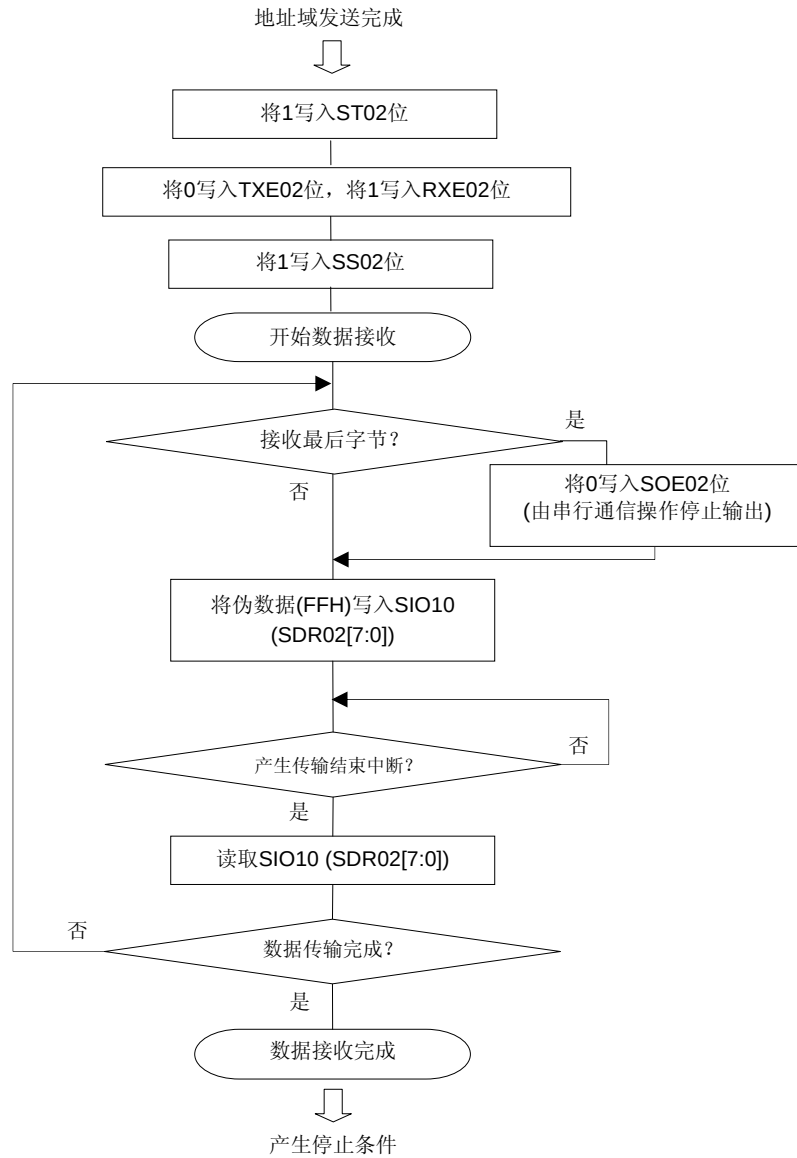
<R>

(b) 接收最后的数据时



备注 m: 部件编号 (m = 0、1), n: 通道编号 (n = 0、2), r: IIC 编号 (r = 10、20)

图 12-96. 数据接收的流程图



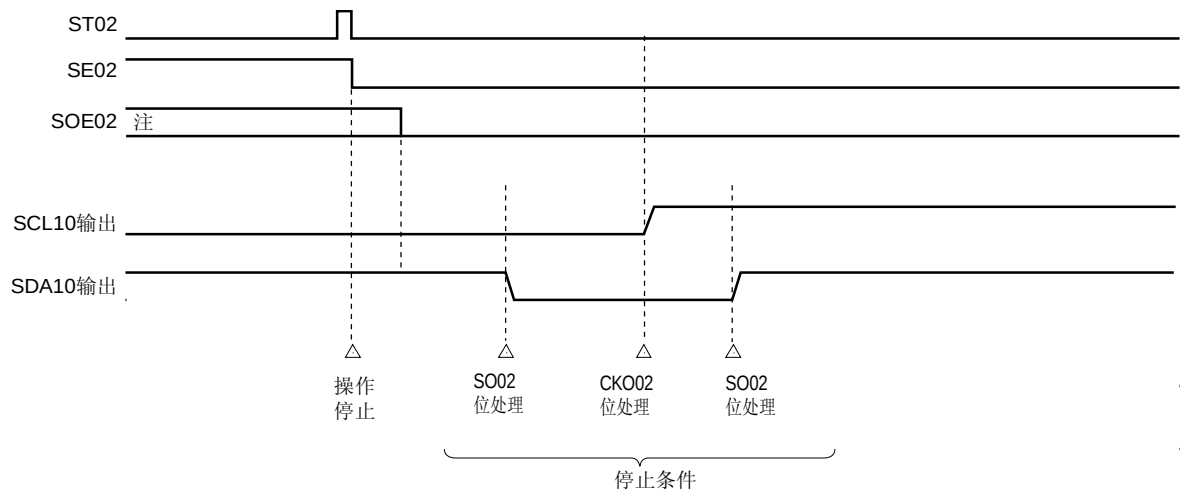
注意事项 当接收最后的数据（NACK）时，不输出 ACK。通过将“1”设置给 ST02 位以停止操作，从而完成通信，并产生一个停止条件。

12.7.4 产生停止条件

在将所有数据发送至目标从机或者接收来自目标从机的所有数据后，会产生停止条件并解除总线。

(1) 处理流程

图 12-97. 停止条件产生的时序图



注 接收期间，在接收最后的数据之前将 SOE02 位清除为 0。

图 12-98. 停止条件产生的流程图



12.7.5 计算传输速率

简易的 I²C (IIC10) 通信的传输速率可以由以下表达式计算得出。

$$(\text{传输速率}) = \{\text{目标通道的操作时钟 (MCK) 频率}\} \div (\text{SDR02}[15:9] + 1) \div 2$$

备注 SDR02[15:9]的值是指 SDR02 寄存器位 15 至位 9 的值 (0000000B 至 1111111B)，也就是 0 至 127。

操作时钟 (MCK) 由串行时钟选择寄存器 0 (SPS0) 和串行模式寄存器 02 (SMR02) 的位 15 (CKS02) 控制。

表 12-4. 操作时钟的选择

SMR02 寄存器	SPS0 寄存器								操作时钟(MCK) ^{‡1}	
CKS02	PRS013	PRS012	PRS011	PRS010	PRS003	PRS002	PRS001	PRS000		f _{CLK} = 20 MHz
0	X	X	X	X	0	0	0	0	f _{CLK}	20 MHz
	X	X	X	X	0	0	0	1	f _{CLK} /2	10 MHz
	X	X	X	X	0	0	1	0	f _{CLK} /2 ²	5 MHz
	X	X	X	X	0	0	1	1	f _{CLK} /2 ³	2.5 MHz
	X	X	X	X	0	1	0	0	f _{CLK} /2 ⁴	1.25 MHz
	X	X	X	X	0	1	0	1	f _{CLK} /2 ⁵	625 kHz
	X	X	X	X	0	1	1	0	f _{CLK} /2 ⁶	313 kHz
	X	X	X	X	0	1	1	1	f _{CLK} /2 ⁷	156 kHz
	X	X	X	X	1	0	0	0	f _{CLK} /2 ⁸	78.1 kHz
	X	X	X	X	1	0	0	1	f _{CLK} /2 ⁹	39.1 kHz
	X	X	X	X	1	0	1	0	f _{CLK} /2 ¹⁰	19.5 kHz
	X	X	X	X	1	0	1	1	f _{CLK} /2 ¹¹	9.77 kHz
	X	X	X	X	1	1	1	1	INTTM0 ^{‡2}	
1	0	0	0	0	X	X	X	X	f _{CLK}	20 MHz
	0	0	0	1	X	X	X	X	f _{CLK} /2	10 MHz
	0	0	1	0	X	X	X	X	f _{CLK} /2 ²	5 MHz
	0	0	1	1	X	X	X	X	f _{CLK} /2 ³	2.5 MHz
	0	1	0	0	X	X	X	X	f _{CLK} /2 ⁴	1.25 MHz
	0	1	0	1	X	X	X	X	f _{CLK} /2 ⁵	625 kHz
	0	1	1	0	X	X	X	X	f _{CLK} /2 ⁶	313 kHz
	0	1	1	1	X	X	X	X	f _{CLK} /2 ⁷	156 kHz
	1	0	0	0	X	X	X	X	f _{CLK} /2 ⁸	78.1 kHz
	1	0	0	1	X	X	X	X	f _{CLK} /2 ⁹	39.1 kHz
	1	0	1	0	X	X	X	X	f _{CLK} /2 ¹⁰	19.5 kHz
	1	0	1	1	X	X	X	X	f _{CLK} /2 ¹¹	9.77 kHz
	1	1	1	1	X	X	X	X	INTTM02 ^{‡2}	
其它									禁止设置	

- 注 1.** 当改变为 f_{CLK} 所选的时钟（通过改变系统时钟控制寄存器（CKC）的值）时，在停止（ST0 = 000FH）串行阵列部件（SAU）的操作之后进行。为操作时钟选择了 INTTM02 时，也会停止定时器阵列部件（TT0 = 00FFH）。
- 2.** 不管 f_{CLK} 频率（主系统时钟、副系统时钟）如何，通过将 TAUS 的 TIS0 寄存器的 TIS02 位设置为 1，并为输入时钟选择 f_{SUB}/4 和使用 SPS0 寄存器选择 INTTM02，在副系统时钟的固定分频比处操作 SAU。但是，改变 f_{CLK} 时，必须按上面注 1 中所述的停止 SAU 和 TAUS。

备注 X: 可忽略

这是 $MCK = f_{CLK} = 20\text{ MHz}$ 处 IIC 传输速率的设置示例。

IIC传输模式 (预期的传输速率)	$f_{CLK} = 20\text{ MHz}$			
	操作时钟(MCK)	SDR02[15:9]	计算出的传输速率	来自预期的传输速率的误差
100 kHz	f_{CLK}	99	100 kHz	0.0%
400 kHz	f_{CLK}	24	400 kHz	0.0%

12.8 发生错误时的处理步骤

如果发生各类型的错误，则要遵循的处理步骤如图 12-99 至图 12-101 所述。

图 12-99. 发生奇偶校验错误或超限错误时的处理步骤

软件处理	硬件状态	备注
读取SDR0n寄存器。	➔ BFF = 0，启动通道n以接收数据。	如果在错误处理期间完成下一次接收，此操作用以防止超限错误。
读取SSR0n寄存器。		识别错误类型，读取到的值用于清除错误标志。
写SIR0n寄存器。	➔ 清除错误标志。	通过将从SSR0n寄存器读到的值照旧写入SIR0n寄存器，则仅在读取期间可以清除错误。

图 12-100. 发生帧错误时的处理步骤

软件处理	硬件状态	备注
读取SDR0n寄存器。	➔ BFF = 0，启动通道n以接收数据。	如果在错误处理期间完成下一次接收，此操作用以防止超限错误。
读取SSR0n寄存器。		识别错误类型，读取到的值用于清除错误标志。
写SIR0n寄存器。	➔ 清除错误标志。	通过将从SSR0n寄存器读到的值照旧写入SIR0n寄存器，则仅在读取期间可以清除错误。
将 ST0n位设置为1。	➔ SE0n = 0，通道n停止操作。	
与其它通信方同步		重新确认与其它通信方同步，并恢复通信，这是因为考虑到移动开始位已经导致帧错误发生。
将SS0n位设置为1。	➔ SE0n = 1，启动通道n，使之运行。	

备注 n: 通道编号 (n = 0 至 3)

图 12-101. 在简易的 I²C 模式中发生奇偶校验错误（ACK 错误）时的处理步骤

软件处理	硬件状态	备注
读取SDR02寄存器。 —————▶	BFF = 0，启动通道2以接收数据。	如果在错误处理期间完成下一次接收，此操作用以防止超限错误。
读取SSR02寄存器。		识别错误类型，读取到的值用于清除错误标志。
写SIR02寄存器。 —————▶	清除错误标志。	通过将从SSR02寄存器读到的值照旧写入SIR02寄存器，则仅在读取期间可以清除错误。
将ST02位设置为1。 —————▶	SE02 = 0，通道2停止操作。	由于没有返回ACK，所以从机没有为接收准备就绪。因此，创建停止位，解除总线，并从开始条件重新开始通信。或者，产生重新开始条件，且从地址发送开始重新发送。
创建停止条件。		
创建开始条件。		
将SS02位设置为1。 —————▶	SE02 = 1，启动通道2，使之运行。	

12.9 寄存器设置和引脚之间的关系

表 12-5 至 12-8 给出了串行阵列部件各通道的寄存器设置和引脚之间的关系。

表 12-5. 寄存器设置和引脚之间的关系（通道 0：CSI00、UART0 发送）

SE 00 注 1	MD 002	MD0 01	SOE 00	SO0 0	CKO 00	TXE 00	RXE 00	PM 75	P75	PM 74 注 2	P74 注 2	PM 73	P73	操作模式	引脚功能		
															SCK00/ KR5/P75	SI00/ RxD0/KR4/ P74 注 2	SO00/ TxD0/KR3/ P73
0	0	0	0	1	1	0	0	× 注 3	× 注 3	× 注 3	× 注 3	× 注 3	× 注 3	工作停止模式	KR5/P75	KR4/P74	KR3/P73
	0	1														KR4/P74/ RxD0	
1	0	0	0	1	1	0	1	1	×	1	×	×	×	从机 CSI00 接收	SCK00 (输入)	SI00	KR3/P73
			1	0/1 注 4	1	1	0	1	×	×	×	0	1	从机 CSI00 发送	SCK00 (输入)	KR4/P74	SO00
			1	0/1 注 4	1	1	1	1	×	1	×	0	1	从机 CSI00 发送 /接收	SCK00 (输入)	SI00	SO00
			0	1	0/1 注 4	0	1	0	1	1	×	×	×	主机 CSI00 接收	SCK00 (输出)	SI00	KR3/P73
			1	0/1 注 4	0/1 注 4	1	0	0	1	×	×	0	1	主机 CSI00 发送	SCK00 (输出)	KR4/P74	SO00
			1	0/1 注 4	0/1 注 4	1	1	0	1	1	×	0	1	主机 CSI00 发送 /接收	SCK00 (输出)	SI00	SO00
	0	1	1	0/1 注 4	1	1	0	×	×	×	×	0	1	UART0 发送 注 5	KR5/P75	KR4/P74/ RxD0	TxD0

注 1. SE0 寄存器是只读状态寄存器，使用 SS0 和 ST0 寄存器进行设置。

2. 当将通道 1 设置给 UART0 接收时，该引脚变为 RxD0 功能引脚（参考表 12-6）。这种情况下，必须为通道 0 选择工作停止模式或 UART0 发送。

3. 该引脚可设置成端口功能引脚。

4. 这里的 0 或 1 取决于通信操作。更多细节，参考 12.3 (12) 串行输出寄存器 0 (SO0)。

5. 成对使用 UART0 发送和接收时，将通道 1 设置给 UART0 接收（参考表 12-6）。

备注 X: 可忽略

表 12-6. 寄存器设置和引脚之间的关系（通道 1: CSI01、UART0 接收）

SE 01 注 1	MD 012	MD0 11	SOE 01	SO 01	CKO 01	TXE 01	RXE 01	PM 72	P72	PM 71	P71	PM 70	P70	PM 74 注 2	P74 注 2	操作模式	引脚功能			
																	SCK01/ KR2/ INTP6/ P72	SI01/ KR1/ INTP5/ P71	SO01/ KR0/ INTP4/ P70	SI00/RxD0/ KR4/P74 注 2
0	0	0	0	1	1	0	0	× 注 3	× 注 3	× 注 3	× 注 3	× 注 3	× 注 3	× 注 3	× 注 3	工作停止模 式	KR2/ INTP6/ P72	KR1/ INTP5/ P71	KR0/ INTP4/ P70	KR4/P74
	0	1																		
1	0	0	0	1	1	0	1	1	×	1	×	×	×	×	×	从机 CSI01 接收	SCK01 (输入)	SI01	KR0/ INTP4/ P70	SI00/KR4/ P74
			1	0/1 注 4	1	1	0	1	×	×	×	0	1	×	×	Slave CSI01 发送	SCK01 (输入)	KR1/ INTP5/ P71	SO01	SI00/KR4/ P74
			1	0/1 注 4	1	1	1	1	×	1	×	0	1	×	×	从机 CSI01 发送/接收	SCK01 (输入)	SI01	SO01	SI00/KR4/ P74
			0	1	0/1 注 4	0	1	0	1	1	×	×	×	×	×	主机 CSI01 接收	SCK01 (输出)	SI01	KR0/ INTP4/ P70	SI00/KR4/ P74
			1	0/1 注 4	0/1 注 4	1	0	0	1	×	×	0	1	×	×	主机 CSI01 发送	SCK01 (输出)	KR1/ INTP5/ P71	SO01	SI00/KR4/ P74
			1	0/1 注 4	0/1 注 4	1	1	0	1	1	×	0	1	×	×	主机 CSI01 发送/接收	SCK01 (输出)	SI01	SO01	SI00/KR4/ P74
	0	1	0	1	1	0	1	×	×	×	×	×	×	1	×	UART0 接收 注 5, 6	KR2/ INTP6/ P72	KR1/ INTP5/ P71	KR0/ INTP4/ P70	RxD0

注 1. SE0 寄存器是只读状态寄存器，使用 SS0 和 ST0 寄存器进行设置。

2. 当将通道 1 设置给 UART0 接收时，该引脚变为 RxD0 功能引脚。在这种情况下，将通道 0 设置给工作停止模式或 UART0 发送（参考表 12-5）。

当将通道 0 置为 CSI00 时，该引脚不能用作 RxD0 功能引脚。在这种情况下，将通道 1 设置成工作停止模式或 CSI01。

3. 该引脚可设置成端口功能引脚。

4. 这里的 0 或 1 取决于通信操作。更多细节，参考 12.3 (12) 串行输出寄存器 0 (SO0)。

5. 成对使用 UART0 发送和接收时，将通道 0 设置给 UART0 发送（参考表 12-5）。

6. 在 UART0 接收期间还必须设置通道 0 的 SMR00 寄存器。关于细节，参见 12.5.2 (1) 寄存器设置。

备注 X: 可忽略

表 12-7. 寄存器设置和引脚之间的关系 (通道 2: CSI10、UART1 发送、IIC10)

SE 02 注1	MD 022	MD 021	SOE 02	SO 02	CKO 02	TXE 02	RXE 02	PM3 2	P32	PM 31 注2	P31 注2	PM 30	P30	操作模式	引脚功能		
															$\overline{\text{SCK10}}$ / SCL10/ INTP2/P32	SI10/SDA10/ RxD1/INTP1/ P31注2	SO10/ TxD1/P30
0	0	0	0	1	1	0	0	× 注3	× 注3	× 注3	× 注3	× 注3	× 注3	工作停止模式	INTP2/P32	INTP1/P31	P30
	0	1														RxD1/INTP1/ P31	
	1	0														INTP1/P31	
1	0	0	0	1	1	0	1	1	×	1	×	×	×	从机 CSI10 接收	$\overline{\text{SCK10}}$ (输入)	SI10	P30
			1	0/1 注4	1	1	0	1	×	×	×	0	1	从机 CSI10 发送	$\overline{\text{SCK10}}$ (输入)	INTP1/P31	SO10
			1	0/1 注4	1	1	1	1	×	1	×	0	1	从机 CSI10 发送/接收	$\overline{\text{SCK10}}$ (输入)	SI10	SO10
			0	1	0/1 注4	0	1	0	1	1	×	×	×	主机 CSI10 接收	$\overline{\text{SCK10}}$ (输出)	SI10	P30
			1	0/1 注4	0/1 注4	1	0	0	1	×	×	0	1	主机 CSI10 发送	$\overline{\text{SCK10}}$ (输出)	INTP1/P31	SO10
			1	0/1 注4	0/1 注4	1	1	0	1	1	×	0	1	主机 CSI10 发送/接收	$\overline{\text{SCK10}}$ (输出)	SI10	SO10
	0	1	1	0/1 注4	1	1	0	×	×	×	×	0	1	UART1 发送注5	INTP2/P32	RxD1/INTP1/ P31	TxD1
	0	1	0	0/1 注6	0/1 注6	0	0	0	1	0	1	×	×	IIC10 开始条件	SCL10	SDA10	P30
1	1	1	1	0/1 注4	0/1 注4	1	0	0	1	0	1	×	×	IIC10 地址域发送	SCL10	SDA10	P30
														IIC10 数据发送	SCL10	SDA10	P30
														IIC10 数据接收	SCL10	SDA10	P30
														IIC10 停止条件	SCL10	SDA10	P30
0	0	0	0	0/1 注7	0/1 注7	0	0	0	1	0	1	×	×	IIC10 停止条件	SCL10	SDA10	P30
						1	0										
						0	1										

注 1. SE0 寄存器是只读状态寄存器，使用 SS0 和 ST0 寄存器进行设置。

- 当将通道 3 设置给 UART1 接收时，该引脚变为 RxD1 功能引脚（参考表 12-8）。在这种情况下，必须为通道 2 选择 UART1 发送的工作停止模式。
- 该引脚可设置成端口功能引脚。
- 这里是 0 或 1 取决与通信操作。更多细节，参考 12.3 (12) 串行输出寄存器 0 (SO0)。
- 成对使用 UART1 发送和接收时，将通道 3 设置给 UART1 接收（参考表 12-8）。
- 在产生开始条件之前将 CKO02 位设置为 1。产生开始条件时，将 SO02 位从 1 清除为 0。
- 产生停止条件之前将 CKO02 位设置为 1。产生停止条件时，将 SO02 位从 0 清除为 1。

备注 X: 可忽略

表 12-8. 寄存器设置和引脚之间的关系（通道 3: UART1 接收）

SE03 ^{注1}	MD032	MD031	TXE03	RXE03	PM31 ^{注2}	P31 ^{注2}	操作模式	引脚功能
								SI10/SDA10/RxD1/INTP1/P31 ^{注2}
0	0	1	0	0	x ^{注3}	x ^{注3}	工作停止模式	SI10/SDA10/INTP1/P31 ^{注2}
1	0	1	0	1	1	x	UART1 接收 ^{注4, 5}	RxD1

- 注 1. SE0 寄存器是只读状态寄存器，使用 SS0 和 ST0 寄存器进行设置。
2. 当将通道 3 设置成 UART1 接收时，该引脚变为 RxD1 功能引脚。在这种情况下，将通道 2 设置工作停止模式或 UART1 发送（参考表 12-7）。
- 将通道 2 设置成 CSI10 或 IIC10 时，该引脚不能用作 RxD1 功能引脚。在这种情况下，将通道 3 设置成工作停止模式。
3. 该引脚可设置为端口功能引脚。
4. 成对使用 UART1 发送和接收时，将通道 2 设置给 UART1 发送（参考表 12-7）。
5. 在 UART1 接收期间还必须设置通道 2 的 SMR02 寄存器。有关细节，参考 12.5.2 (1) 寄存器设置。

备注 X: 可忽略

第十三章 串行接口 IICA

备注 78K0R/KC3-L 的 44-引脚产品未配备串行接口 IICA。

13.1 串行接口 IICA 的功能

串行接口 IICA 具有以下三种模式。

(1) 工作停止模式

不执行串行传输时使用该模式。因此，可以用来减少功耗。

(2) I²C 总线模式（支持多主机通信）

该模式用于在多个设备间进行 8 位数据传输，使用两条信号线：串行时钟（SCL0）线和串行数据总线（SDA0）线。

该模式遵从 I²C 总线格式，主机设备可以通过串行数据总线，可以产生至从设备的“开始条件”、“地址”、“传输方向指定”、“数据”和“停止条件”数据。从设备自动检测由软件接收到的状态和数据。这种功能可以使控制 I²C 总线的部分应用程序得到简化。

由于 SCL0 和 SDA0 引脚用作漏极开路输出，因此，IICA 的串行时钟信号线和串行数据总线信号线都需要上拉电阻。

(3) 唤醒模式

在 STOP 模式中，当已接收到来自主机设备的扩展码或本机地址时，通过产生中断请求信号（INTIICA）可以解除 STOP 模式。使用 IICA 控制寄存器 1（IICCTL1）的 WUP 位可以对其进行设置。

图 13-1 给出了串行接口 IICA 的框图。

图 13-1. 串行接口 IICA 的框图

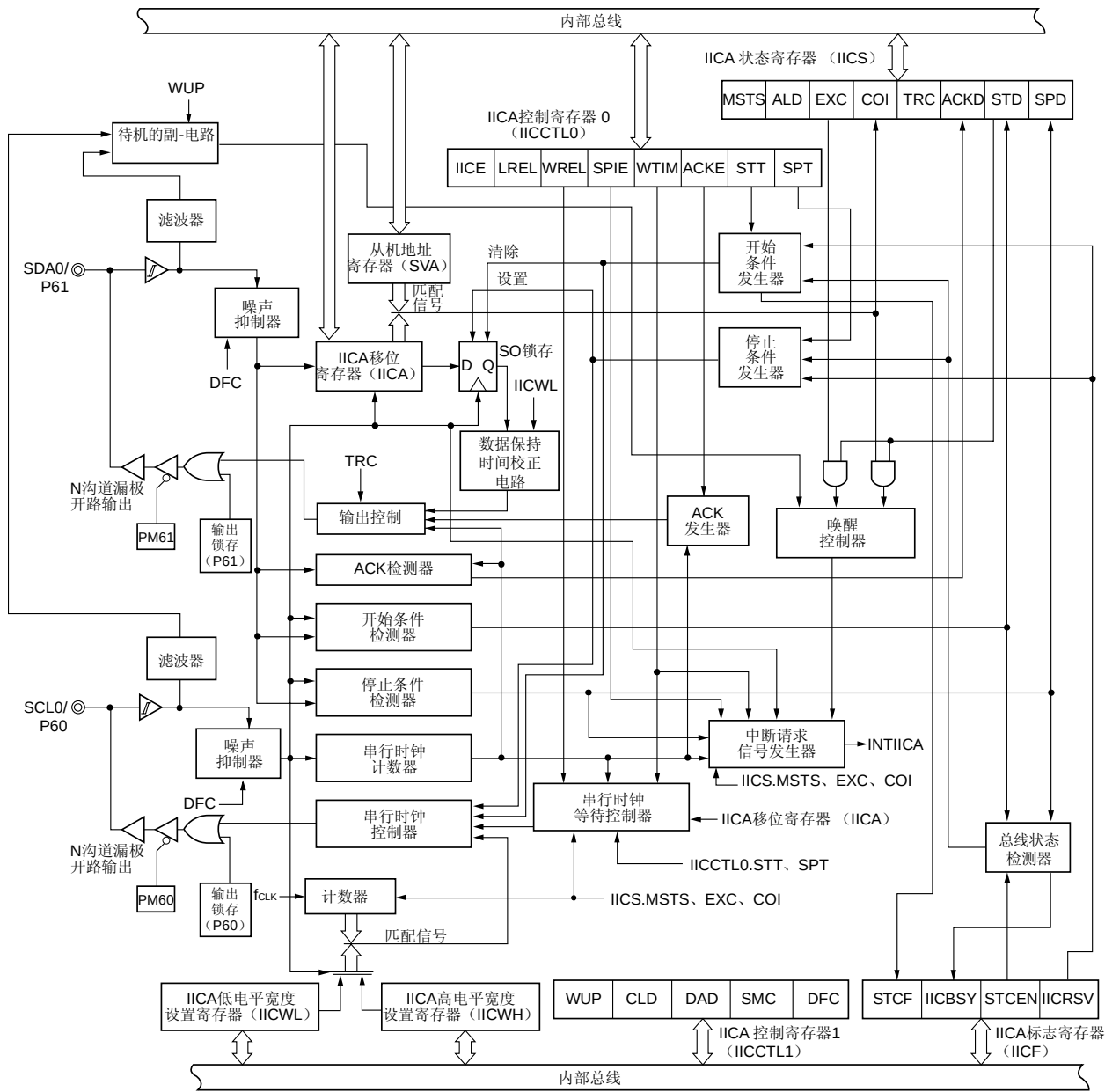
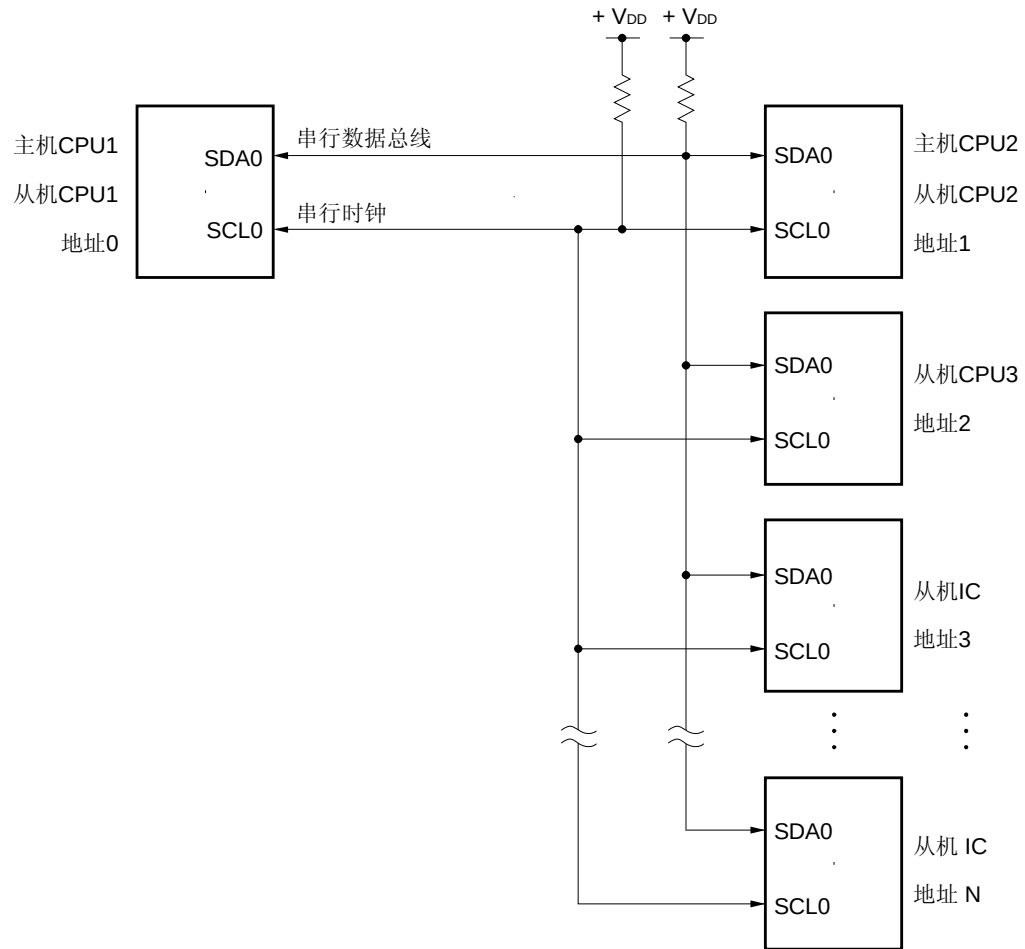


图 13-2 所示为串行总线结构的示例。

图 13-2. 使用I²C 总线的串行总线结构示例



13.2 串行接口 IICA 的配置

串行接口 IICA 包括以下硬件。

表 13-1. 串行接口 IICA 的配置

项目	配置
寄存器	IICA 移位寄存器 (IICA) 从地址寄存器 (SVA)
控制寄存器	外围使能寄存器 0 (PER0) IICA 控制寄存器 0 (IICCTL0) IICA 状态寄存器 (IICS) IICA 标志寄存器 (IICF) IICA 控制寄存器 1 (IICCTL1) IICA 低电平宽度设置寄存器 (IICWL) IICA 高电平宽度设置寄存器 (IICWH) 端口模式寄存器 6 (PM6) 端口寄存器 6 (P6)

(1) IICA 移位寄存器 (IICA)

IICA 用于将 8 位串行数据与串行时钟同步地转换为 8 位并行数据，反之亦然。IICA 既能用于发送又能用于接收。

目前的发送和接收操作由对 IICA 的写和读操作控制。

在等待期间通过将数据写入 IICA 来取消等待状态和开始数据传输。

可以由 8 位存储器处理指令设置 IICA。

产生的复位信号将 IICA 清除为 00H。

图 13-3. IICA 移位寄存器 (IICA) 的格式

地址: FFF50H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
IICA								

- 注意事项
1. 在数据传输期间不要将数据写入 IICA 中。
 2. 仅在等待期间，对 IICA 进行写或读操作。在等待期间以外的其它通信状态时，禁止访问 IICA。
然而，当设备作为主机服务时，在将通信触发位 (STT) 设置为 1 后，仅能写入 IICA 一次。
 3. 保留通信时，在检测到由停止条件触发引起的中断之后，将数据写入 IICA。

<R>

(2) 从地址寄存器 (SVA)

在从机模式中，该寄存器存储七位{A6、A5、A4、A3、A2、A1、A0}本机地址。

可以由 8 位存储器处理指令设置 SVA。

但是，STD = 1（检测到开始条件时）时，禁止对该寄存器进行重写。

产生的复位信号将 SVA 清除为 00H。

图 13-4. 从地址寄存器（SVA）的格式

地址: F0234H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
SVA	A6	A5	A4	A3	A2	A1	A0	0 ^注

注 位 0 固定为 0。

(3) SO 锁存器

SO 锁存器用来保持 SDA0 引脚的输出电平。

(4) 唤醒控制器

当该寄存器接收到的地址与设置给从地址寄存器（SVA）的值匹配时，或当接收到扩展码时，该电路就会产生一个中断请求信号（INTIICA）。

(5) 串行时钟计数器

该计数器对在发送/接收操作期间输出或输入的串行时钟进行计数，并用于检验 8 位数据的发送或接收。

(6) 中断请求信号发生器

该电路控制中断请求信号（INTIICA）的产生。

由以下两种触发都可产生一个 I²C 中断请求信号。

- 串行时钟的第 8 个或第 9 个时钟下降沿（由 WTIM 位设定）
- 当检测到停止条件时产生中断请求（由 SPIE 位设定）

备注 WTIM 位: IICA 控制寄存器 0（IICCTL0）的位 3
SPIE 位: IICA 控制寄存器 0（IICCTL0）的位 4

(7) 串行时钟控制器

主机模式中，该电路从采样时钟中产生经 SCL0 引脚输出的时钟。

(8) 串行时钟等待控制器

该电路控制等待时间。

(9) ACK 发生器、停止条件检测器、开始条件检测器以及 ACK 检测器

这些电路产生并检测各个状态。

(10) 数据保持时间校正电路

该电路根据相应的串行时钟下降沿，为数据产生保持时间。

(11) 开始条件发生器

当将 STT 位设置为 1 时，该电路产生一个开始条件。

然而，在通信保留禁止状态（IICRSV 位 = 1）中，未解除总线（IICBSY 位 = 1）时，忽略开始条件请求，并将 STCF 位设置为 1。

(12) 停止条件发生器

当将 SPT 位设置为 1 时，该电路产生一个停止条件。

(13) 总线状态检测器

该电路通过检测开始条件和停止条件，从而检测总线是否被释放。
然而，由于操作后不能立刻检测到总线状态，因此由 STCEN 位设定其初始状态。

- 备注**
- STT 位： IICA 控制寄存器 0（IICCTL0）的位 1
 - SPT 位： IICA 控制寄存器 0（IICCTL0）的位 0
 - IICRSV 位： IICA 标志寄存器（IICF）的位 0
 - IICBSY 位： IICA 标志寄存器（IICF）的位 6
 - STCF 位： IICA 标志寄存器（IICF）的位 7
 - STCEN 位： IICA 标志寄存器（IICF）的位 1

13.3 控制串行接口 IICA 的寄存器

串行接口 IICA 由以下八个寄存器控制。

- 外围使能寄存器 0 (PER0)
- IICA 控制寄存器 0 (IICCTL0)
- IICA 标志寄存器 (IICF)
- IICA 状态寄存器 (IICS)
- IICA 控制寄存器 1 (IICCTL1)
- IICA 低电平宽度设置寄存器 (IICWL)
- IICA 高电平宽度设置寄存器 (IICWH)
- 端口模式寄存器 6 (PM6)
- 端口寄存器 6 (P6)

(1) 外围使能寄存器 0 (PER0)

PER0 用于启动或禁止各外围硬件宏功能的使用。供应未使用的硬件宏功能的时钟会停止工作，以便于减少功耗和噪声。

当使用串行接口 IICA 时，务必将该寄存器的位 4 (IICAEN) 置为 1。

可以由 1 位或 8 位存储器处理指令设置 PER0。

产生的复位信号将该寄存器清除为 00H。

图 13-5. 外围使能寄存器 0 (PER0) 的格式

地址: F00F0H 复位后: 00H R/W

符号	<7>	6	<5>	<4>	3	<2>	1	0
PER0	RTCEN	0	ADCEN	IICAEN	0	SAU0EN	0	0

IICAEN	串行接口 IICA 输入时钟供应控制
0	停止输入时钟供应。 • 不能写入由串行接口 IICA 使用的 SFR。 • 串行接口 IICA 处于复位状态中。
1	供应输入时钟。 • 不能读取或写入由串行接口使用的 IICA。

- 注意事项**
1. 设置串行接口 IICA 时，务必首先将 IICAEN 置为 1。如果 IICAEN = 0，则忽略对串行接口 IICA 的控制寄存器的写入，即使读取寄存器，也只能读取默认值。
 2. 务必要将 PER0 寄存器的位 0、1、3 和 6 清除为 0。

(2) IICA 控制寄存器 0 (IICCTL0)

该寄存器用于允许/停止 I²C 操作、设置等待时间以及设置其它 I²C 操作。

可以由 1 位或 8 位存储器处理指令设置 IICCTL0。然而，当 IICE 位 = 0 时或在等待期间时，设置 SPIE、WTIM 和 ACKE 位。当 IICE 位从“0”置为“1”时，可以同时设置这些位。

产生的复位信号将该寄存器清除为 00H。

图 13-6. IICA 控制寄存器 0（IICCTL0）的格式（1/4）

地址： F0230H 复位后： 00H R/W

符号	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
IICCTL0	IICE	LREL	WREL	SPIE	WTIM	ACKE	STT	SPT

IICE	I ² C操作使能
0	停止操作。 复位IICA状态寄存器(IICS) ^{※1} 。 停止内部操作。
1	操作使能。
当 SCL0 和 SDA0 线为高电平时务必设置该位(1)。	
清除条件 (IICE = 0)	
• 由指令清除 • 复位	
设置条件 (IICE = 1)	
• 由指令设置	
LREL ^{※2,3}	从通信退出
0	正常操作
1	从当前通信退出并设置为待机模式，本设置在被执行之后自动清除为 0。 包括在已接收到本地无关扩展码的情况下使用。 SCL0 和 SDA0 信号线被设为高阻状态。 IICA 控制寄存器 0（IICS）和 IICA 状态寄存器（IICS）的下列标志被清除为 0。 • STT • SPT • MSTs • EXC • COI • TRC • ACKD • STD
从通信中退出后的待机状态一直保持有效，直到遇到下面的通信入口条件为止。 • 检测到停止条件后，在主机模式下重新开始。 • 在开始条件之后，发生地址匹配或扩展码接收。	
清除条件 (LREL = 0)	
• 执行后自动被清除 • 复位	
设置条件 (LREL = 1)	
• 由指令设置	
WREL ^{※2,3}	取消等待
0	不取消等待。
1	取消等待。 取消等待后自动清除该设置。
发送状态下，在等待期间的第九个时钟脉冲处设置 WREL（取消等待）时，SDA0 信号线进入高阻抗状态（TRC = 0）。	
清除条件 (WREL = 0)	
• 执行后自动被清除 • 复位	
设置条件 (WREL = 1)	
• 由指令设置	

注 1. 复位 IICS 寄存器、IICF 寄存器的 STCF 和 IICBSY 位、以及 IICCTL1 寄存器的 CLD 和 DAD 位。

 2. IICE0 为 0 时该位信号无效。

 3. 读取 LREL 和 WREL 位时，总是读出 0。

<R>

<R>

注意事项 当SCL0 信号线为高电平，SDA0 信号线为低电平，且IICCTL1 寄存器的DFC为 1 时，如果允许（IICE = 1）I²C操作，则将立刻检测到开始条件。I²C开始操作（IICE = 1）后的瞬间，使用 1 位存储器处理指令设置LREL（1）。

图 13-6. IICA 控制寄存器 0 (IICCTL0) 的格式 (2/4)

SPIE ^{注 1}	检测到停止条件时，允许/禁止中断请求的产生
0	禁止
1	允许
如果 IICCTL1 寄存器的 WUP 为 1，即使 SPIE = 1 也将不会产生停止条件中断。	
清除条件 (SPIE = 0)	设置条件 (SPIE = 1)
- 由指令清除 - 复位	由指令设置

WTIM ^{注 1}	等待和中断请求产生控制
0	在第 8 个时钟的下降沿产生中断请求。 主机模式： 在输出 8 个时钟以后，时钟输出被置为低电平，并设置等待。 从机模式： 在输入 8 个时钟以后，时钟被置为低电平，并设置主设备等待。
1	在第 9 个时钟信号的下降沿产生中断请求。 主机模式： 输出 9 个时钟以后，时钟输出被置为低电平，并且设置等待。 从机模式： 输入 9 个时钟以后，时钟被置为低电平，并设置主设备等待。
在地址传输期间，在第九个时钟的下降沿产生中断，而与该位的设置无关。地址传输结束后，该位的设置生效。主机模式时，在地址传输期间的第九个时钟下降沿处，插入一个等待状态。在从设备接收到本机地址时，发出响应 (ACK) 信号之后，在第九个时钟的下降沿处插入一个等待状态。然而，当从设备收到扩展码时，在第八个时钟的下降沿处插入一个等待状态。	
清除条件 (WTIM = 0)	设置条件 (WTIM = 1)
- 由指令清除 - 复位	由指令设置

ACKE ^{注 1, 2}	响应控制
0	禁止响应。
1	允许响应。 在第九个时钟期间，SDA0 信号线置为低电平。
清除条件 (ACKE = 0)	设置条件 (ACKE = 1)
- 由指令清除 - 复位	由指令设置

- 注
1. IICE0 为 0 时该位信号无效。在那期间设置该位。
 2. 地址传输期间，如果代码并非扩展码，则设置值无效。
当设备作为从机服务且地址间匹配，则不管设置值如何，都会产生响应。

图 13-6. IICA 控制寄存器 0 (IICCTL0) 的格式 (3/4)

<R>

STT ^注	开始条件触发器
0	不产生开始条件。
1	解除总线（待机状态下，IICBSY = 0 时）时： 产生开始条件（主机开始）。当 SCL0 信号线为高电平，SDA0 信号线从高电平变为低电平，然后产生开始条件。接下来，按比例过了一定的时间之后，SCL0 信号线变为低电平（等待状态）。 当第三方通信时： • 允许通信保留功能（IICRSV = 0）时 其功能如同开始条件保留标志一样。当设置为 1 时，总线解除后，自动产生一个开始条件。 • 禁止通信保留功能（IICRSV = 1）时 STCF 置为 1，并清除将 STT 置位（1）的信息。不产生开始条件。 处于等待状态（使用主设备时）： 等待释放后，产生一个重新开始条件。
有关设置时序的注意事项 • 主机接收：传输期间不能设置为 1。只能在等待期间 ACKE 已经清除为 0，且从设备已经被告知最末接收时，才可以置为 1。 • 主机发送：在响应期间，通常不会产生开始条件。在输出第 9 个时钟之后的等待期间置为 1。 • 不能和 SPT 同时置为 1。 • 将 STT 置为 1，那么，禁止在将它清除为 0 之前再次对其进行设置。	
清除条件 (STT = 0)	设置条件 (STT = 1)
• 在禁止通信保留时通过将 STT 置为 1 进行清除 • 由仲裁失败清除 • 主机产生开始条件后清除 • 由 LREL = 1（从通信退出）清除 • IICE = 0（操作停止）时 • 复位	• 由指令设置

注 IICE0 为 0 时该位信号无效。

备注 1. 在数据设置后读取位 1 时，位 1（STT）变为 0。
2. IICRSV: IIC 标志寄存器（IICF）的位 0
STCF: IIC 标志寄存器（IICF）的位 7

图 13-6. IICA 控制寄存器 0 (IICCTL0) 的格式 (4/4)

SPT	停止条件触发器
0	不产生停止条件。
1	产生停止条件（终止主机设备传输）。 SDA0 信号线变为低电平以后，或者将 SCL0 信号线设置为高电平，或者一直等到它变为高电平。接下来，按比例经过一段时间后，SDA0 信号线由低电平变为高电平，产生停止条件。
有关设置时序的注意事项 • 主机接收： 传输期间不能置为 1。 只能在等待期间 ACKE 已经清除为 0，且从设备已经被告知最末接收时，才可以置为 1。 • 主机发送： 在响应期间，通常不会产生停止条件。因此，在输出第 9 个时钟之后的等待期间置为 1。 • 不能和 STT 同时置为 1。 • 仅在主机模式中 SPT 可以置为 1。 • 在 WTIM 已经清除为 0 时，如果在输出 8 个时钟后的等待期间将 SPT 置为 1，需要注意，停止条件将在第 9 个时钟的高电平期间产生。在输出 8 个时钟后的等待期间，WTIM 应该从 0 变为 1，并且，在输出第 9 个时钟的后的等待期间，SPT 应该置为 1。 • 将 SPT 置为 1，那么，禁止在将它清除为 0 之前再次对其进行设置。	
清除条件 (SPT = 0)	设置条件 (SPT = 1)
• 由仲裁失败清除 • 检测到停止条件后自动清除 • 由 LREL = 1（从通信退出）清除 • IICE = 0（操作停止）时 • 复位	• 由指令设置

注意事项 当 IICA 状态寄存器 (IICS) 的位 3 (TRC) 置为 1 时，清除 TRC 且 SDA0 信号线置为高阻抗之后，在第九个时钟期间将 WREL 置为 1，并取消等待。

备注 在数据设置后读取位 0 时，位 0 (SPT) 变为 0。

<R>

(3) IICA 状态寄存器 (IICS)

该寄存器表示I²C的状态。

仅当 STT = 1 和等待期间，由 1 位或 8 位存储器处理指令读取 IICS。

产生的复位信号将该寄存器清除为 00H。

注意事项 当 IICA 控制寄存器 1 (IICCTL1) 的 WUP 置为 1 时，禁止读取 IICA 状态寄存器 (IICS)。当 WUP 从 0 变为 1 时，直到检测到下一个开始条件或停止条件时才会反映状态上的变化，与 INTIICA 中断请求无关。因此，要使用唤醒模式，通过检测停止条件允许 (SPIE = 1) 中断产生，在已检测到中断后读取 IICS 寄存器。

图 13-7. IICA 状态寄存器 (IICS) 的格式 (1/3)

地址: FFF51H 复位后: 00H R

符号	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
IICS	MSTS	ALD	EXC	COI	TRC	ACKD	STD	SPD

MSTS	主机状态检查标志
0	从设备状态或通信待机状态
1	主机设置通信状态
清除条件 (MSTS = 0)	
- 检测到停止条件时 - ALD = 1 (仲裁失败) 时 - 由 LREL = 1 (从通信退出) 清除 - IICE 从 1 变为 0 (操作停止) 时 - 复位	
设置条件 (MSTS = 1)	
产生开始条件时	

ALD	仲裁失败检测
0	此状态意味着不存在仲裁或仲裁结果为“赢”。
1	此状态指示仲裁结果为“失败”。 MSTS is cleared.
清除条件 (ALD = 0)	
- 读取^注IICS后自动清除 - IICE 从 1 变为 0(操作停止)时 - 复位	
设置条件 (ALD = 1)	
仲裁结果为“失败”时。	

EXC	扩展码接收检测
0	未收到扩展码。
1	收到扩展码。
清除条件 (EXC = 0)	
- 检测到开始条件时 - 检测到停止条件时 - 由 LREL = 1 (从通信退出) 清除 - IICE 从 1 变为 0 (操作停止) 时 - 复位	
设置条件 (EXC = 1)	
当接收到的地址数据的高 4 位为“0000”或“1111” (在第 8 个时钟的上升沿设置) 时。	

注 当对除了 IICS 之外的其它位执行 1 位存储器处理指令时，该寄存器也被清除。因此，使用 ALD 位时，在其它位的数据之前读取该位的数据。

备注 LREL: IICA 控制寄存器 0 (IICCTL0) 的位 6
 IICE: IICA 控制寄存器 0 (IICCTL0) 的位 7

图 13-7. IICA 状态寄存器 (IICS) 的格式 (2/3)

COI	匹配地址检测
0	地址不匹配。
1	地址匹配。
清除条件 (COI = 0)	
- 检测到开始条件时 - 检测到停止条件时 - 由 LREL = 1 (从通信退出) 清除 - IICE 从 1 变为 0 (操作停止) 时 - 复位	
设置条件 (COI = 1)	
当接收到的地址和本机地址 (从地址寄存器 (SVA)) 匹配时 (在第 8 个时钟的上升沿设置)。	

TRC	发送/接收状态检测
0	接收状态 (不同于发送状态)。SDA0 信号线设置为高阻态。
1	发送状态。SO0 锁存器内的值允许输出到 SDA0 信号线 (在首字节的第 9 个时钟的下降沿开始有效)。
清除条件 (TRC = 0)	
<主机和从机> - 检测到停止条件时 - 由 LREL = 1 (从通信退出) 清除 - IICE 从 1 变为 0 (操作停止) 时 - 由 WREL = 1[※] (取消等待) 清除 - ALD 从 0 变为 1 (仲裁失败) 时 - 复位 <主机> - 将“1”输出至首字节的 LSB (传输方向指示位) 时 <从机> - 检测到开始条件时 - 将“0”输出至首字节的 LSB (传输方向指示位) 时 <不用于通信时>	
设置条件 (TRC = 1)	
<主机> - 产生开始条件时 - 将“0”输出至首字节的 LSB (传输方向指示位) 时 <从机> - 将“1”输出至首字节的 LSB (传输方向指示位) 时	

注 当 IICA 状态寄存器 (IICS) 的位 3 (TRC) 为 1 时, 如果在第九个时钟处通过将 IICA 控制寄存器 0 (IICCTL0) 的位 5 (WREL) 置为 1 而取消等待状态, 则清除 TRC, 且 SDA0 信号线进入高阻抗状态。

备注 LREL: IICA 控制寄存器 0 (IICCTL0) 的位 6
IICE: IICA 控制寄存器 0 (IICCTL0) 的位 7

图 13-7. IICA 状态寄存器 (IICS) 的格式 (3/3)

ACKD	响应(ACK)检测	
0	未检测到响应。	
1	检测到响应。	
清除条件 (ACKD = 0)		设置条件 (ACKD = 1)
• 检测到停止条件时 • 处于下一字节的第一个时钟的上升沿 • 由 LREL = 1 (从通信退出) 清除 • IICE 从 1 变为 0 (操作停止) 时 • 复位		• 在 SCL0 的第 9 个时钟的上升沿处 SDA0 信号线设置为低电平后

STD	开始条件检测	
0	未检测到开始条件。	
1	这表明地址传输期间有效。	
清除条件 (STD = 0)		设置条件 (STD = 1)
• 检测到停止条件时 • 在地址传输后的下一个字节的第一个时钟的上升沿 • 由 LREL = 1 (从通信退出) 清除 • IICE 从 1 变为 0 (操作停止) 时 • 复位		• 检测到开始条件时

SPD	停止条件检测	
0	未检测到停止条件。	
1	终止主机设备的通信，并解除总线。	
清除条件 (SPD = 0)		设置条件 (SPD = 1)
• 在该位的设置和开始条件的检测之后，在地址传输字节的首个时钟的上升沿 • IICE 从 1 变为 0 (操作停止) 时 • 复位		• 检测到停止条件时

备注 LREL: IICA 控制寄存器 0 (IICCTL0) 的位 6
 IICE: IICA 控制寄存器 0 (IICCTL0) 的位 7

(4) IICA 标志寄存器 (IICF)

该寄存器设置I²C的操作模式，并指示I²C总线的状态。
可以由 1 位或 8 位存储器处理指令设置 IICF。但是，STCF 和 IICBSY 位为只读位。
IICRSV 位可用于允许/禁止通信保留功能。
STCEN 可用于设置 IICBSY 位的初始值。
仅当禁止I²C的操作 (IICA控制寄存器 0 (IICCTL0) 的位 7 (IICE) = 0) 时，IICRSV和STCEN才可被写入。当操作使能时，可以读取IICF寄存器。
产生的复位信号将该寄存器清除为 00H。

图 13-8. IICA 标志寄存器 (IICF) 的格式

地址: FFF52H 复位后: 00H R/W^注

符号	<7>	<6>	5	4	3	2	<1>	<0>
IICF	STCF	IICBSY	0	0	0	0	STCEN	IICRSV

STCF	STT 清除标志
0	产生开始条件
1	产生开始条件失败: 清除STT标志
清除条件 (STCF = 0)	
· 由STT = 1清除 · IICE = 0 (操作停止)时 · 复位	
设置条件 (STCF = 1)	
· 当禁止通信保留 (IICRSV = 1) 时 产生开始条件失败且将STT清除为0。	

IICBSY	I ² C 总线状态标志
0	总线解除状态 (STCEN = 1时通信初始状态)
1	总线通信状态 (STCEN = 0时通信初始状态)
清除条件 (IICBSY = 0)	
· 停止条件的检测 · IICE = 0 (操作停止)时 · 复位	
设置条件 (IICBSY = 1)	
· 开始条件的检测 · STCEN = 0时IICE的设置	

STCEN	初始化开始使能触发器
0	操作使能 (IICE = 1) 后, 在停止条件检测时允许开始条件的产生。
1	操作使能 (IICE = 1) 后, 在没有检测到停止条件时允许开始条件的产生。
清除条件 (STCEN = 0)	
· 由指令清除清除清除 · 开始条件的检测 · 复位	
设置条件 (STCEN = 1)	
· 由指令设置	

IICRSV	通信保留功能禁止位
0	允许通信保留
1	禁止通信保留
清除条件 (IICRSV = 0)	
· 由指令清除清除清除 · 复位	
设置条件 (IICRSV = 1)	
· 由指令设置	

注 位 6 和位 7 为只读位。**注意事项 1.** 仅当停止操作 (IICE = 0) 时才写入 STCEN。

2. 当 STCEN = 1 时, 不管总线实际状态如何, 总是确认总线解除状态 (IICBSY = 0), 所以, 当产生首个开始条件 (STT = 1) 时, 有必要证实当前没有进行第三方通信, 以免破坏其通信。

3. 仅当停止操作 (IICE = 0) 时才写入 IICRSV。**备注** STT: IICA 控制寄存器 0 (IICCTL0) 的位 1

IICE: IICA 控制寄存器 0 (IICCTL0) 的位 7

(5) IICA 控制寄存器 1 (IICCTL1)

该寄存器用于设置I²C的操作模式和检测SCL0 和SDA0 引脚的状态。
可以由 1 位或 8 位存储器处理指令设置 IICCTL1。但是，CLD 和 DAD 位为只读位。
当 IICA 控制寄存器 0 (IICCTL0) 的位 7 (IICE) 为 0 时，设置 IICCTL1，WUP 位除外。
产生的复位信号将该寄存器清除为 00H。

图 13-9. IICA 控制寄存器 1 (IICCTL1) 的格式 (1/2)

地址: F0231H 复位后: 00H R/W^注

符号	7	6	<5>	<4>	<3>	<2>	1	0
IICCTL1	WUP	0	CLD	DAD	SMC	DFC	0	0

WUP	地址匹配唤醒控制
0	在 STOP 模式中停止地址匹配唤醒功能的操作。
1	在 STOP 模式中允许地址匹配唤醒功能的操作。
WUP = 1 时要切换至 STOP 模式，需在置位 (1) WUP 后至少三个时钟时执行 STOP 指令。 在地址匹配或接收到扩展码后清除 (0) WUP。通过清除 (0) WUP，才能进入后续通信。(在已清除 (0) WUP 之后，必须解除等待状态，且必须写入发送数据。) WUP = 1 时，地址匹配或已接收到扩展码时的中断时序与 WUP = 0 时的中断时序相同。将会发生时钟采样差延迟。此外，在 WUP = 1 时，即使将 SPIE 位设置为 1，也不会产生停止条件中断。 当由来自串行接口 IICA 的中断之外的中断源设置 WUP = 0 时，直到检测到后续开始条件或停止条件时才会执行诸如主机设备之类的操作。在没有等到对后续开始条件或停止条件的检测时，通过置位 (1) STT 位不会输出开始条件。	
清除条件 (WUP = 0)	设置条件 (WUP = 1)
• 由指令清除 (地址匹配或接收扩展码之后)	• 由指令设置 (当MSTS、EXC和COI为“0”，且STD也为“0”(未进入通信时) ^注

注 在下面显示的期间中，必须检查 IICS 的状态且必须设置 WUP。

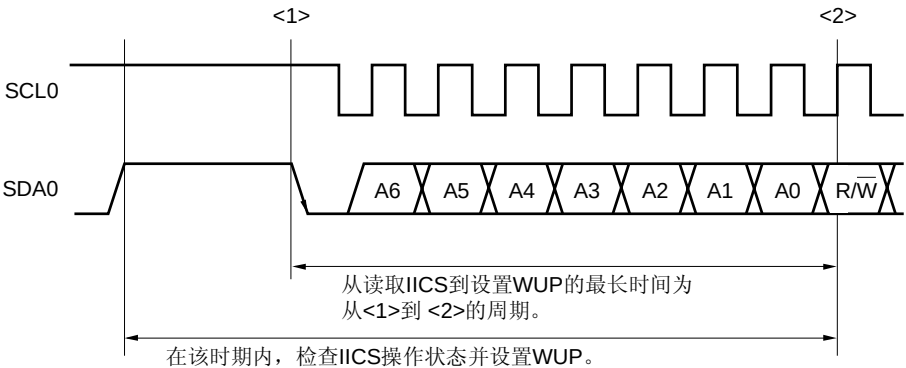


图 13-9. IICA 控制寄存器 1 (IICCTL1) 的格式 (2/2)

CLD	SCL0 引脚电平的检测(仅当 IICE = 1 时有效)	
0	在低电平时检测 SCL0 引脚。	
1	在高电平时检测 SCL0 引脚。	
清除条件 (CLD = 0)		设置条件 (CLD = 1)
• SCL0 引脚为低电平时 • IICE = 0(操作停止)时 • 复位		• SCL0 引脚为高电平时

DAD	SDA0 引脚电平的检测 (仅当 IICE = 1 时有效)	
0	在低电平时检测 SDA0 引脚。	
1	在高电平时检测 SDA0 引脚。	
清除条件 (DAD = 0)		设置条件 (DAD = 1)
• SDA0 引脚为低电平时 • IICE = 0 (操作停止)时 • 复位		• SDA0 引脚为高电平时

SMC	操作模式转换	
0	在标准模式下操作。	
1	在快速模式下操作。	

DFC	数字滤波器操作控制
0	数字滤波器关闭。
1	数字滤波器打开。
数字滤波器只能在快速模式下使用。 在快速模式中，传输时钟没有变化，而与 DFC 位置位（1）或清除（0）无关。 在快速模式下数字滤波器用于噪声抑制。	

注 位 4 和位 5 为只读位。

备注 IICE: IICA 控制寄存器 0 (IICCTL0) 的位 7

(6) IICA 低电平宽度设置寄存器 (IICWL)

该寄存器用于设置 SCL0 引脚信号的低电平宽度，该信号由处于主机模式中的串行接口 IICA 输出。

可以由 8 位存储器处理指令设置 IICWL。

当 IICA 控制寄存器 0 (IICCTL0) 的位 7 (IICE) 为 0 时，设置 IICWL。

产生的复位信号将该寄存器设置为 FFH。

图 13-10. IICA 低电平宽度设置寄存器 (IICWL) 的格式

地址:	F0232H	复位后:	FFH	R/W				
符号	7	6	5	4	3	2	1	0
IICWL								

(7) IICA 高电平宽度设置寄存器 (IICWH)

该寄存器用于设置 SCL0 引脚信号的高电平宽度，该信号由处于主机模式中的串行接口 IICA 输出。

可以由 8 位存储器处理指令设置 IICWH。

当 IICA 控制寄存器 0 (IICCTL0) 的位 7 (IICE) 为 0 时，设置 IICWH。

产生的复位信号将该寄存器设置为 FFH。

图 13-11. IICA 高电平宽度设置寄存器 (IICWH) 的格式

地址:	F0233H	复位后:	FFH	R/W				
Symbol	7	6	5	4	3	2	1	0
IICWH								

(8) 端口模式寄存器 6 (PM6)

该寄存器按位为单位设置端口 6 的输入/输出。

使用 P60/SCL0 引脚和 P61/SDA0 引脚作为串行数据 I/O 时，清除 PM60 和 PM61，并输出 P60 和 P61 的锁存为 0。

在设置输出模式之前将 IICE (IICA 控制寄存器 0 (IICCTL0) 的位 7) 置为 1，这是因为当 IICE 为 0 时，P60/SCL0 和 P61/SDA0 引脚输出低电平 (固定)。

可以由 1 位或 8 位存储器处理指令设置 PM6。

产生的复位信号将该寄存器设置为 FFH。

图 13-12. 端口模式寄存器 6 (PM6) 的格式

地址:	FFF26H	复位后:	FFH	R/W								
符号	7	6	5	4	3	2	1	0				
PM6	1	1	1	1	1	1	PM61	PM60				

PM6n	P6n 引脚 I/O 模式选择 (n = 0、1)
0	输出模式 (输出缓冲器打开)
1	输入模式 (输出缓冲器关闭)

13.4 I²C总线模式功能

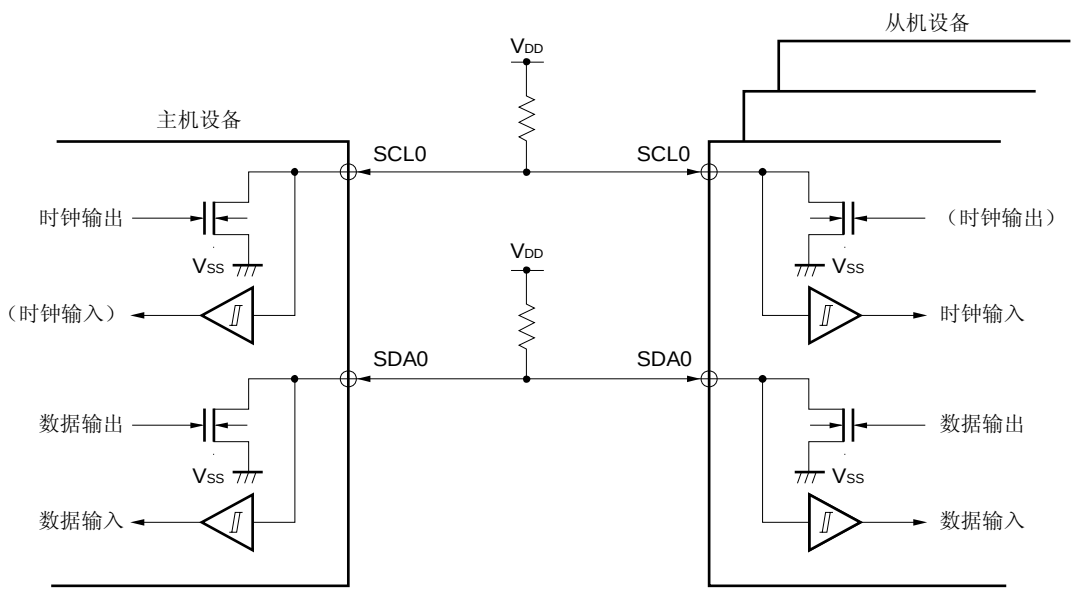
13.4.1 引脚配置

串行时钟引脚（SCL0）和串行数据总线引脚（SDA0）配置如下。

- (1) SCL0..... 该引脚用于串行时钟输入和输出。
对于主机设备和从机设备，该引脚都是 N 沟道漏极开路输出。输入为施密特输入。
- (2) SDA0 该引脚用于串行数据输入和输出。
对于主机设备和从机设备，该引脚都是 N 沟道漏极开路输出。输入为施密特输入。

由于从串行时钟信号线和串行数据总线信号线输出的都是 N 沟道漏极开路输出，所以都需要外接上拉电阻。

图 13-13. 引脚配置图



<R>

13.4.2 使用 IICWL 和 IICWH 寄存器设置传输时钟

(1) 在主机端设置传输时钟

$$\text{传输时钟} = \frac{\text{IICWL} + \text{IICWH} + f_{\text{CLK}} (t_{\text{R}} + t_{\text{F}})}{f_{\text{CLK}}}$$

此时，IICWL 和 IICWH 的最佳设定值如下所示。

（所有设定值的小数部分上舍入。）

- 快速模式时

$$\begin{aligned} \text{IICWL} &= \frac{0.52}{\text{传输时钟}} \times f_{\text{CLK}} \\ \text{IICWH} &= \left(\frac{0.48}{\text{传输时钟}} - t_{\text{R}} - t_{\text{F}} \right) \times f_{\text{CLK}} \end{aligned}$$

- 正常模式时

$$\begin{aligned} \text{IICWL} &= \frac{0.48}{\text{传输时钟}} \times f_{\text{CLK}} \\ \text{IICWH} &= \left(\frac{0.53}{\text{传输时钟}} - t_{\text{R}} - t_{\text{F}} \right) \times f_{\text{CLK}} \end{aligned}$$

(2) 在从机端设置 IICWL 和 IICWH

（所有设定值的小数部分上舍入。）

- 快速模式时

$$\begin{aligned} \text{IICWL} &= 1.3 \mu\text{S} \times f_{\text{CLK}} \\ \text{IICWH} &= (1.2 \mu\text{S} - t_{\text{R}} - t_{\text{F}}) \times f_{\text{CLK}} \end{aligned}$$

- 正常模式时

$$\begin{aligned} \text{IICWL} &= 4.7 \mu\text{S} \times f_{\text{CLK}} \\ \text{IICWH} &= (5.3 \mu\text{S} - t_{\text{R}} - t_{\text{F}}) \times f_{\text{CLK}} \end{aligned}$$

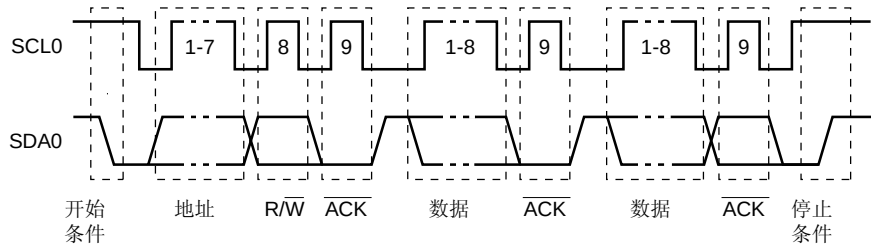
备注

IICWL: IICA 低电平宽度设置寄存器
 IICWH: IICA 高电平宽度设置寄存器
 t_F: SDA0 和 SCL0 信号下降时间
 t_R: SDA0 和 SCL0 信号上升时间
 f_{CLK}: CPU/外围硬件时钟频率

13.5 I²C总线的定义和控制方法

以下部分描述了I²C总线的串行数据通信格式和I²C总线所用信号。图 13-14 显示了经I²C总线的串行数据总线输出的“开始条件”、“地址”、“数据”和“停止条件”的传输时序。

图 13-14. I²C总线串行数据的传输时序



主机设备产生开始条件、从地址和停止条件。

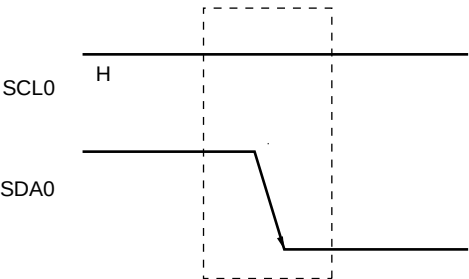
响应信号（ $\overline{\text{ACK}}$ ）既可由主机产生，也可由从机产生（通常情况下，它由接收 8 位数据的设备输出）。

串行时钟（SCL0）由主机设备持续输出。然而，在从机设备中，SCL0 信号的低电平时段可被延长，并插入一个等待状态。

13.5.1 开始条件

当 SCL0 引脚为高电平且 SDA0 引脚由高电平变为低电平时，形成开始条件。SCL0 引脚和 SDA0 引脚的开始条件是开始串行传输时主机设备产生至从机设备的信号。当设备用作从机设备时可以检测开始条件。

图 13-15. 开始条件



已检测到停止条件（SPD：IICA 状态寄存器（IICS）的位 0 = 1）之后，当 IICA 控制寄存器 0（IICCTL0）的位 1（STT）被置位（1）时，输出开始条件。检测到开始条件时，IICS 的位 1（STD）被置位（1）。

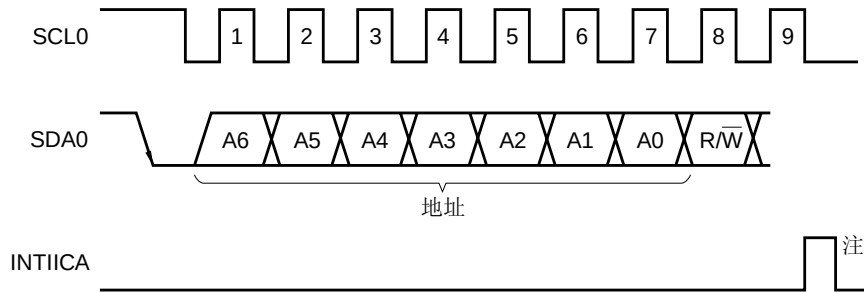
13.5.2 地址

由开始条件之后的 7 位数据定义地址。

地址为 7 位数据字段，输出地址从而选择通过总线连接到主机设备的从机设备。所以，每一个经总线连接的从机设备都必须具有唯一的地址。

从机设备包含检测开始条件的硬件单元，并检查 7 位地址数据是否和存储在从地址寄存器（SVA）内的数据值匹配。如果地址数据和 SVA 寄存器内储存的数据值相匹配，则选定从设备并且和主机进行通信，直到主机设备产生一个开始条件或结束条件为止。

图 13-16. 地址



注 如果在从机设备操作期间接收了除本机地址或扩展码之外的数据，则不会发出 INTIICA。

当组成从地址和 13.5.3 传输方向指定中描述的传输方向的 8 位总数据写入 IICA 移位寄存器（IICA）时，输出地址。将接收到的地址写入 IICA。

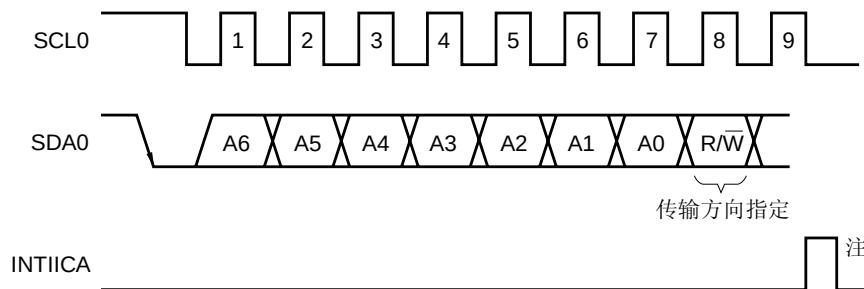
从地址分配至 IICA 寄存器的高 7 位。

13.5.3 传输方向指定

除了 7 位地址数据之外，主机设备还发送指示传输方向的 1 位数据。

当此传输方向指示位为“0”时，表示主机设备正向从机设备发送数据。当此传输方向指示位为“1”时，表示主机设备正接收来自从机设备的数据。

图 13-17. 传输方向指定



注 如果在从机设备操作期间接收了除本机地址或扩展码之外的数据，则不会发出 INTIICA。

13.5.4 响应信号 ($\overline{\text{ACK}}$)

$\overline{\text{ACK}}$ 用于检查发送端和接收端传输数据的状态。

每次接收到 8 位数据，接收端都会返回 $\overline{\text{ACK}}$ 。

通常，在发送 8 位数据之后发送端接收 $\overline{\text{ACK}}$ 。当 $\overline{\text{ACK}}$ 从接收端返回时，这表明接收已经正确执行且会继续处理。使用 IICA 状态寄存器 (IICS) 的位 2 (ACKD) 可以检验是否已检测到 $\overline{\text{ACK}}$ 。

当主机接收最后数据项时，不会返回 $\overline{\text{ACK}}$ ，取而代之的是产生一个停止条件。如果在接收数据后从机没有返回 $\overline{\text{ACK}}$ ，则主机输出一个停止条件或者重新开始条件，并停止发送。如果没有返回 $\overline{\text{ACK}}$ ，则可能的原因如下。

- <1> 未正常执行接收。
- <2> 接收到最后数据项。
- <3> 由地址指定的接收端不存在。

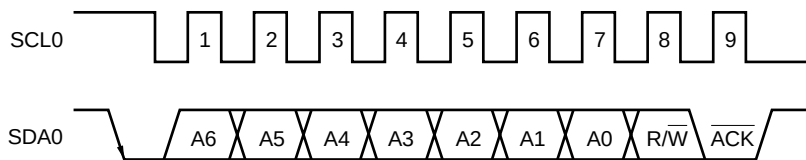
要产生 $\overline{\text{ACK}}$ ，接收端需使 SDA0 信号线在第九个时钟处为低电平（指示正常接收）。

通过将 IICA 控制寄存器 0 (IICCTL0) 的位 2 (ACKE) 置为 1，允许 $\overline{\text{ACK}}$ 的自动产生。由 7 位地址信息之后的八位数据设置 IICS 寄存器的位 3 (TRC)。通常，在接收 (TRC = 0) 中将 ACKE 置为 1。

如果在接收 (TRC = 0) 期间从机不再接收数据或者不需要下一个数据项，通过将 ACKE 清除为 0，从机必须告知主机，然后就不再接收数据。

当在接收 (TRC = 0) 期间主机不需要下一个数据项时，必须将 ACKE 清除为 0，这样就不会产生 $\overline{\text{ACK}}$ 。以这种方式，主机在发送端通知从机不再需要任何数据的信息（发送将会停止）。

图 13-18. $\overline{\text{ACK}}$



当接收到本机地址时，不管 ACKE 的设定值如何， $\overline{\text{ACK}}$ 信号都自动产生。当接收到本机地址以外的地址时，不产生 $\overline{\text{ACK}}$ 信号 (NACK)。

当接收到扩展码时，如果预先将 ACKE 置为 1，则会产生 $\overline{\text{ACK}}$ 。

根据等待时序的设置，接收数据时产生 $\overline{\text{ACK}}$ 的方式有如下差别。

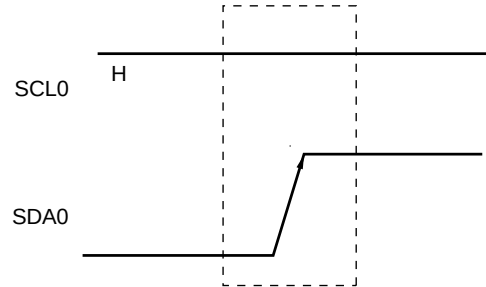
- 选择 8 个时钟的等待状态时 (IICCTL0 寄存器的位 3 (WTIM) = 0) :
在解除等待状态之前将 ACKE 置为 1，在 SCL0 引脚的第八个时钟的下降沿处产生 $\overline{\text{ACK}}$ 。
- 选择 9 个时钟的等待状态时 (IICCTL0 寄存器的位 3 (WTIM) = 1) :
通过预先将 ACKE 置为 1 产生 $\overline{\text{ACK}}$ 。

13.5.5 停止条件

当 SCL0 引脚为高电平时，将 SDA0 引脚由低电平变为高电平即产生一个停止条件。

停止条件是串行传输已经完成时主机设备产生至从机设备的信号。当设备用作从机设备时可以检测停止条件。

图 13-19. 停止条件



当将 IICA 控制寄存器 0 (IICCTL0) 的位 0 (SPT) 置为 1 时，产生停止条件。当检测到停止条件时，将 IICA 状态寄存器 (IICS) 的位 0 (SPD) 置为 1，并在 IICCTL0 的位 4 (SPIE) 置为 1 时产生 INTIICA。

13.5.6 等待状态

等待状态用于通知通信伙伴设备（主机或从机）已经准备就绪（即：处于等待状态），可以进行收发数据。

将 SCL0 引脚设为低电平从而通知通信伙伴本机处于等待状态。当主机设备和从机设备的等待状态都被取消时，就可以开始下一次传输。

图 13-20. 等待状态（1/2）

(1) 主机设备具有九个时钟等待状态，从机设备具有八个时钟等待状态时
（主机发送，从机接收，且 ACKE = 1）

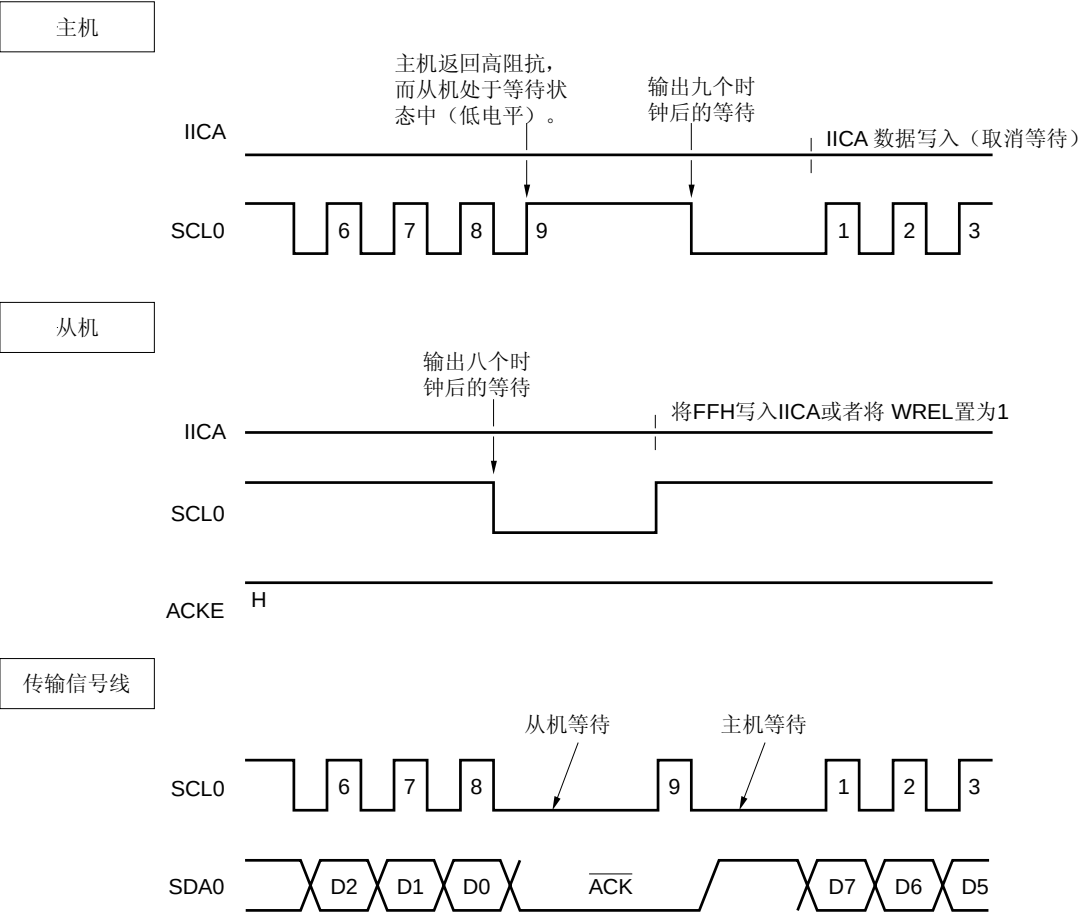
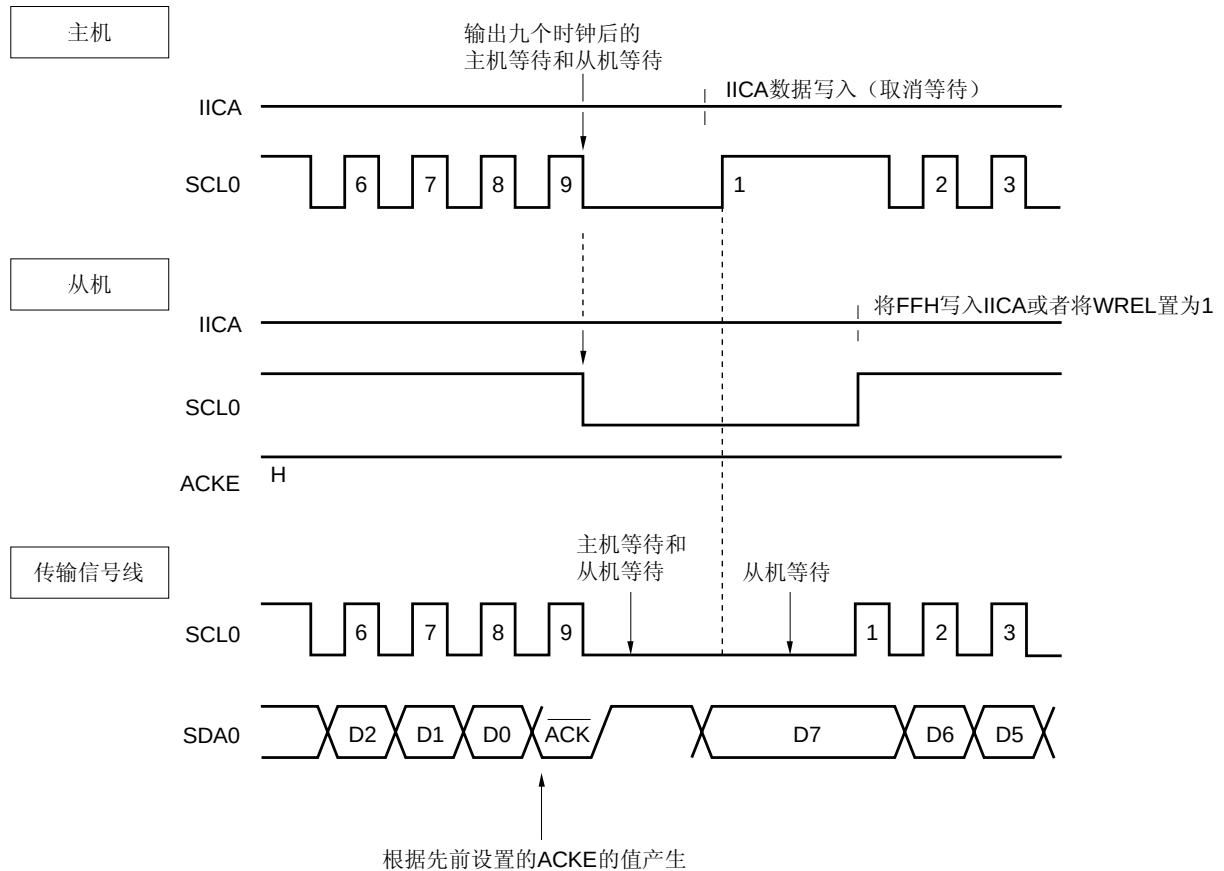


图 13-20. 等待状态 (2/2)

(2) 主机设备和从机设备都具有九个时钟等待状态时
(主机发送, 从机接收, 且 $ACKE = 1$)



备注 ACKE: IICA 控制寄存器 0 (IICCTL0) 的位 2
WREL: IICA 控制寄存器 0 (IICCTL0) 的位 5

根据 IICA 控制寄存器 0 (IICCTL0) 的位 3 (WTIM) 的设置, 等待状态可能会自动产生。

通常, 接收端会在 IICCTL0 的位 5 (WREL) 置为 1 或将 FFH 写入 IICA 移位寄存器 (IICA) 时取消等待状态, 而发送端在将数据写入 IICA 寄存器时取消等待状态。

主机设备还可以通过以下任一方法取消等待状态。

- 通过将 IICCTL0 的位 1 (STT) 置为 1
- 通过将 IICCTL0 的位 0 (SPT) 置为 1

13.5.7 取消等待

I²C通常由以下处理取消等待状态。

- 将数据写入 IICA 移位寄存器 (IICA)
- 置位 IICA 控制寄存器 0 (IICCTL0) 的位 5 (WREL) (取消等待)
- 置为 IICCTL0 寄存器的位 1 (STT) (产生开始条件)^注
- 置位 IICCTL0 寄存器的位 0 (SPT) (产生停止条件)^注

注 仅限于主机设备

当执行以上等待取消的处理时，I²C取消等待状态，并恢复通信。

要取消等待状态并发送数据（包括地址），需将数据写入 IICA。

要在取消等待状态之后接收数据，或者完成数据发送，需将 IICA 控制寄存器 0 (IICCTL0) 的位 5 (WREL) 置为 1。

要在取消等待状态后产生一个重新开始条件，需将 IICCTL0 的位 1 置为 1。

要在等待状态取消后产生一个停止条件，需将 IICCTL0 的位 0 置为 1。

对于一次等待状态，仅执行一次取消处理。

例如，如果在取消等待状态之后，通过将 WREL 置为 1 而将数据写入 IICA，则可能向 SDA0 中输出错误值，这是因为更改 SDA0 信号线的时序与写入 IICA 的时序冲突。

除了上述的以外，如果在通信中止时将 IICE 清除为 0，则会停止通信，这样就能取消等待状态。

如果 I²C 总线因噪声而发生死锁，通过置位 IICCTL0 的位 6 (LREL) 而从通信中将处理救出来，这样就能取消等待状态。

注意事项 如果在 WUP = 1 时执行取消等待状态的处理，则不会取消等待状态。

13.5.8 中断请求（INTIICA）产生时序和等待控制

IICA 控制寄存器 0（IICCTL0）的位 3（WTIM）的设置确定了 INTIICA 产生的时序和相应的等待控制，如表 13-2 所示。

表 13-2. INTIICA 产生时序和等待控制

WTIM	从机设备操作期间			主机设备操作期间		
	地址	数据接收	数据发送	地址	数据接收	数据发送
0	9 ^{#1,2}	8 ^{#2}	8 ^{#2}	9	8	8
1	9 ^{#1,2}	9 ^{#2}	9 ^{#2}	9	9	9

- 注 1.** 仅当与设置给从地址寄存器（SVA）的地址匹配时，从设备的 INTIICA 信号与等待周期才会在第 9 个时钟的下降沿处发生。
- 此刻，不论设置给 IICCTL0 的位 2（ACKE）的值如何，都会产生 $\overline{ACK}$ 信号。对于已经接收到扩展码的从设备，INTIICA 信号在第 8 个时钟的下降沿处发生。
- 但是，如果在重新开始后地址不匹配，则 INTIICA 在第 9 个时钟的下降沿处产生，而并不发生等待。
- 2.** 如果接收到的地址和从地址寄存器（SVA）的内容不匹配，且不能接收到扩展码，则既不产生 INTIICA 信号，也不产生等待。

备注 表中的数目表示串行时钟的时钟信号数。中断请求和等待控制都和这些时钟信号的下降沿同步。

(1) 在地址发送/接收期间

- 从机设备操作：不论 WTIM 位如何设置，中断请求和等待时序都根据上述注 1 和注 2 中说明的条件确定。
- 主机设备操作：不论 WTIM 位如何设置，中断请求和等待时序都在第 9 个时钟的下降沿发生。

(2) 在数据接收期间

- 主机/从机设备操作：中断请求和等待时序根据 WTIM 位决定。

(3) 在数据发送期间

- 主机/从机设备操作：中断请求和等待时序根据 WTIM 位确定。

(4) 等待状态取消方法

四种等待取消方法如下所示。

- 将数据写入 IICA 移位寄存器（IICA）
- 置位 IICA 控制寄存器 0（IICCTL0）的位 5（WREL）（取消等待）
- 置位 IICCTL0 寄存器的位 1（STT）（产生开始条件）[#]
- 置位 IICCTL0 寄存器的位 0（SPT）（产生停止条件）[#]

注 仅限于主机设备。

当已选择 8 时钟等待时（WTIM = 0），取消等待状态之前必须确定是否产生了 $\overline{ACK}$ 信号。

(5) 停止条件检测

当检测待停止条件时产生 INTIICA（仅当 SPIE = 1 时）。

13.5.9 地址匹配检测方法

在I²C总线模式中，主机设备可以通过发送相应的从机地址来选择特定的从机设备。

可以由硬件自动检测地址匹配。当已将本机地址设置给从地址寄存器（SVA）且设置给 SVA 的地址与由主机设备发送的从地址匹配时，或当接收到扩展码时，都会发生中断请求（INTIICA）。

13.5.10 错误检测

在I²C总线模式中，数据发送期间的串行数据总线（SDA0）的状态由发送设备的IICA移位寄存器捕获，这样，发送以前IICA内的数据就可以和所发送的IICA数据进行比较，从而能够检测发送错误。当比较数据值不匹配时，判断为错误发生。

13.5.11 扩展码

- (1) 当接收地址的高 4 位为“0000”或“1111”时，扩展码接收标志（EXC）被置为 1 指示扩展码接收，并在第 8 个时钟的下降沿产生一个中断请求（INTIICA）。存储在从地址寄存器（SVA）中的本机地址不受影响。
- (2) 如果由 10 位传输地址将“11110××0”设置给 SVA，且从主机设备传输“11110××0”，则结果如下。注意，INTIICA 信号在第 8 个时钟的下降沿产生。

- 数据的高 4 位匹配：EXC = 1
- 7 位数据匹配：COI = 1

备注 EXC: IICA 状态寄存器（IICS）的位 5
COI: IICA 状态寄存器（IICS）的位 4

- (3) 由于根据紧随扩展码之后的数据不同，则中断请求信号发生后的处理也各不相同，这样的处理由软件执行。如果在从机设备操作时接收扩展码，即使地址不匹配，从机设备也是多方通信。例如，在接收到扩展码后，如果你不想将目标设备作为从机设备操作，则将 IICA 控制寄存器 0（IICCTL0）的位 6（LREL）置为 1，以将下一次通信操作设置位待机模式。

表 13-3. 主要扩展码的位定义

从地址	R/W 位	描述
0 0 0 0 0 0 0	0	通用呼叫地址
1 1 1 1 0 × ×	0	10-位从地址指定(地址确认期间)
1 1 1 1 0 × ×	1	10-位从地址指定(地址匹配后，产生读取命令时)

备注 除上述内容外，扩展码的其它细节参见由NXP半导体产生的I²C总线指定。

表 13-4. 仲裁期间的状态和中断请求产生时序

仲裁期间的状态	中断请求产生时序
地址发送期间	紧随字节传输 ^{註1} 之后的第 8 或第 9 个时钟的下降沿
地址发送后读取/写入数据	
扩展码发送期间	
扩展码发送后读取/写入数据	
数据发送期间	
数据发送后 $\overline{\text{ACK}}$ 传输期间	
数据传输期间检测到重新开始条件时	
数据传输期间检测到停止条件时	产生停止条件时(SPIE = 1 时) ^{註2}
试图产生重新开始条件且数据处于低电平时	紧随字节传输 ^{註1} 之后的第 8 或第 9 个时钟的下降沿
试图产生重新开始条件且检测到停止条件时	产生停止条件时(SPIE = 1 时) ^{註2}
试图产生停止开始条件且数据处于低电平时	紧随字节传输 ^{註1} 之后的第 8 或第 9 个时钟的下降沿
试图产生重新开始条件且 SCL0 处于低电平时	

- 注
- 1. WTIM (IICA 控制寄存器 0 (IICCTL0) 的位 3) = 1 时, 中断请求在第 9 个时钟的下降沿发生。当 WTIM = 0 并且接收到扩展码的从地址时, 中断请求信号在第 8 个时钟的下降沿发生。
 - 2. 当有可能发生仲裁时, 在主机设备操作下设置 SPIE = 1。

备注 SPIE: IICA 控制寄存器 0 (IICCTL0) 的位 4

13.5.13 唤醒功能

当收到本机地址和扩展码后，I²C总线从机功能是指产生一个中断请求信号（INTIICA）的功能。

当地址不匹配时，该功能就会阻止不必要的 INTIICn 信号产生，从而使处理更有效率。

当检测到开始条件时，设置唤醒待机模式。仲裁失败可能将主机设备（已经产生了开始条件）变为从机设备，在这种可能情况下发送地址时，唤醒待机模式有效。

然而，检测到停止条件时，不管唤醒功能如何，都要设置 IICA 控制寄存器 0（IICCTL0）的位 4（SPIE），并且确定是允许还是禁止中断请求。

要在 STOP 模式中使用唤醒功能，需将 WUP 置为 1。不管操作时钟如何，都能接收地址。当已接收到本机地址和扩展码时，也会产生中断请求信号（INTIICA）。在已产生中断后使用一条指令清除（0）WUP 位，则操作返回正常模式。

地址匹配时设置 WUP 位和清除 WUP（0）位的流程如下所示。

图 13-22. 设定 WUP = 1 时的流程

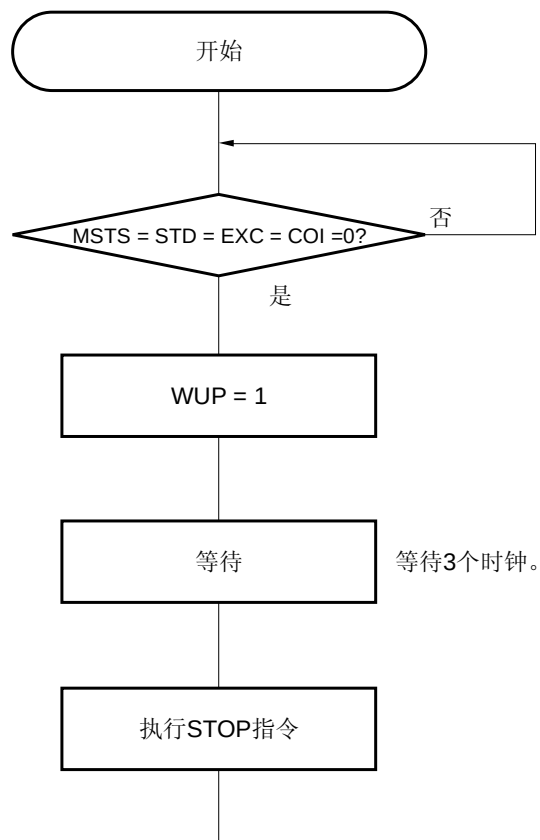
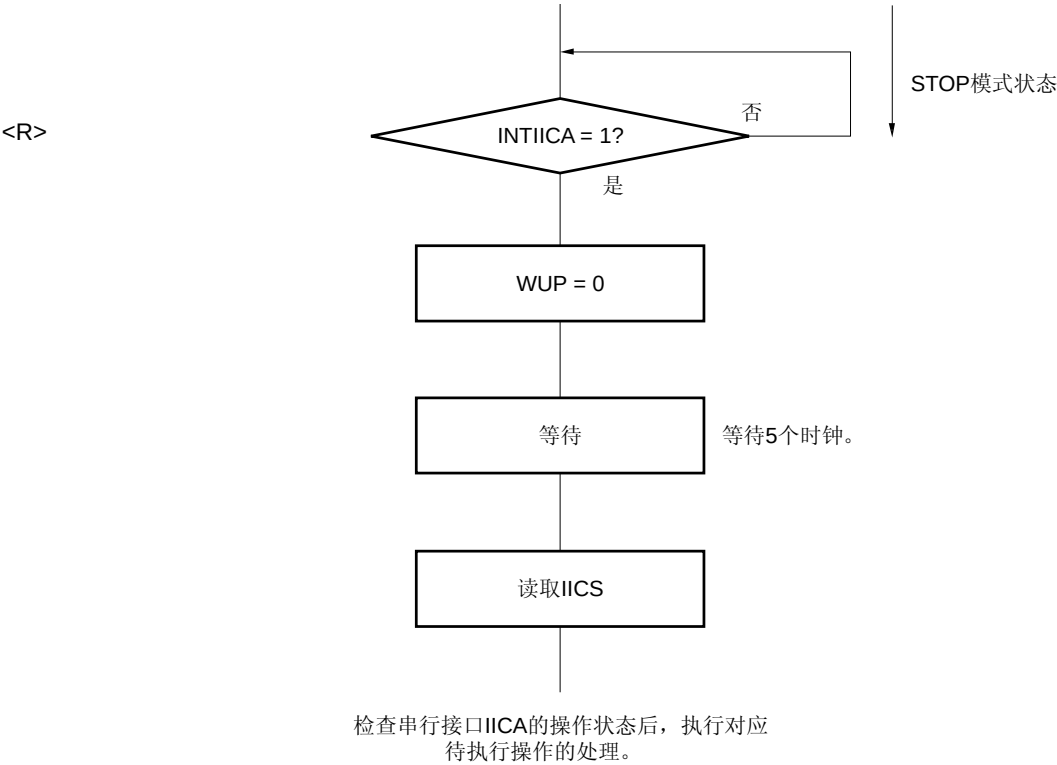


图 13-23. 地址匹配时设定 WUP = 0 时的流程（包括扩展码接收）



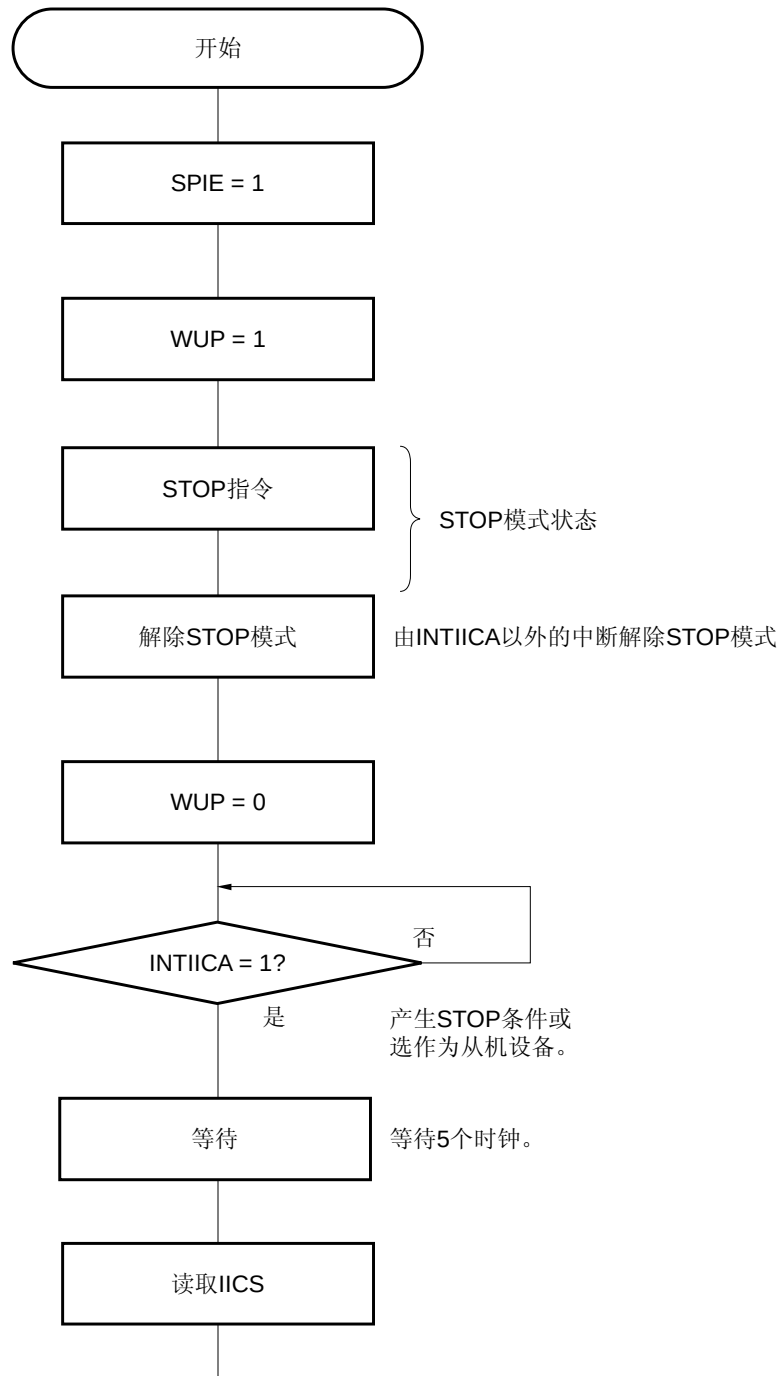
除了由从串行接口 IICA 产生的中断请求（INTIICA）外，使用以下流程执行处理，以解除 STOP 模式。

- 主机设备操作： 流程如图 13-24 所示
- 从机设备操作： 与图 13-23 的流程相同

<R>

<R>

图 13-24. 由 INTIICA 以外的中断解除 STOP 模式后主机设备操作时



检查串行接口IICA的操作状态后，执行对应待执行操作的处理。

13.5.14 通信保留

(1) 允许通信保留功能时 (IICA 标志寄存器 (IICF) 的位 0 (IICRSV) = 0)

在目前不能使用总线的情况下开始主机设备通信时，通信保留功能在总线解除时再允许开始条件的发送。有两种模式下不能使用总线。

- 仲裁结果既不是主机操作也不是从机操作时
- 接收扩展码并禁止从机操作时（未返回 $\overline{ACK}$ ，通过将 IICA 控制寄存器 0 (IICCTL0) 的位 6 (LREL) 置为 1 并保存通信从而解除总线）。

当未使用总线（检测到停止条件后）时，如果将 IICCTL0 的位 1 (STT) 置为 1，开始条件自动产生并设置等待状态。

如果在将 IICCTL0 的位 4 置为 1 后将地址写入 IICA 移位寄存器 (IICA)，则通过产生中断请求信号 (INTIICA) 检测到总线被解除（停止条件检测），然后设备作为主机自动开始通信。在检测到停止条件之前向 IICn 寄存器写入的数据无效。

当 STT 被设置为 1 时，根据总线状态确定工作模式（开始条件或通信保留状态）。

- 如果总线已被释放..... 产生开始条件
- 如果总线未被释放（待机模式） 通信保留

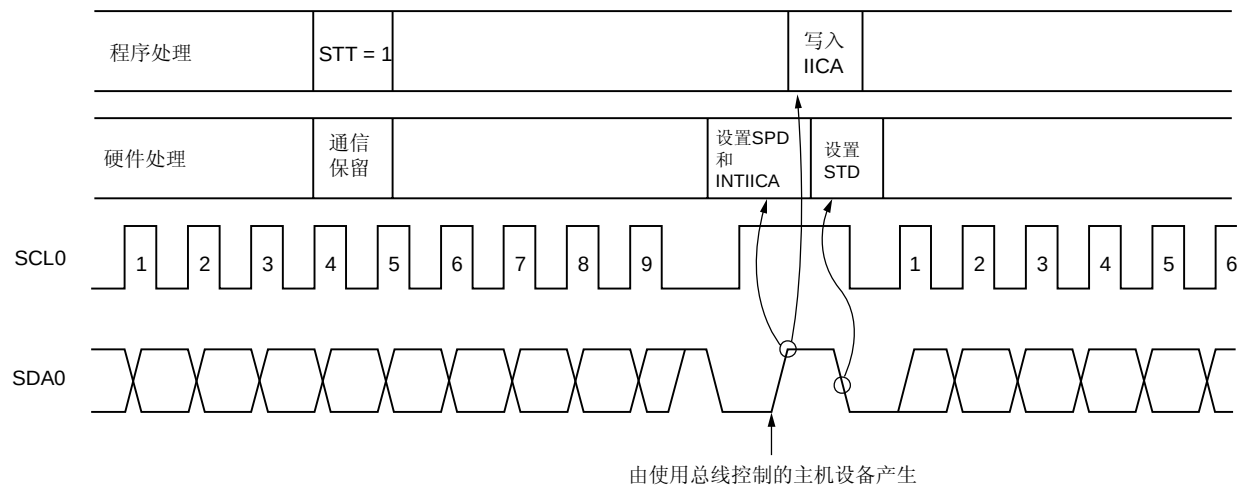
在将 STT 置为 1 且经过等待时间后，使用 MSTS (IICA 状态寄存器 (IICS) 的位 7) 检查通信保留是否运行。使用软件保证由以下表达式计算得到的等待时间。

从设置 STT = 1 到检查 MSTS 标志的等待时间：
 $(IICWL \text{ 设定值} + IICWH \text{ 设定值} + 4 \text{ (时钟)}) \times f_{CLK} + t_F \times 2$

备注 IICWL: IICA 低电平宽度设置寄存器
 IICWH: IICA 高电平宽度设置寄存器
 t_F : SDA0 和 SCL0 信号下降时间（参见第 28 章 电气规范（目标））
 f_{CLK} : CPU/外围硬件时钟频率

图 13-25 给出了通信保留时序。

图 13-25. 通信保留时序



备注

- IICA: IICA 移位寄存器
- STT: IICA 控制寄存器 0 (IICCTL0) 的位 1
- STD: Bit 1 of IICA 状态寄存器 (IICS) 的位 1
- SPD: IICA 状态寄存器 (IICS) 的位 0

通过如图 13-26 所示的时序，受理通信保留。将 IICA 状态寄存器 (IICS) 的位 1 置为 1 后，在检测到停止条件前通过将控制寄存器 0 (IICCTL0) 的位 1 (STT) 置为 1 可以产生通信保留。

图 13-26. 受理通信保留的时序

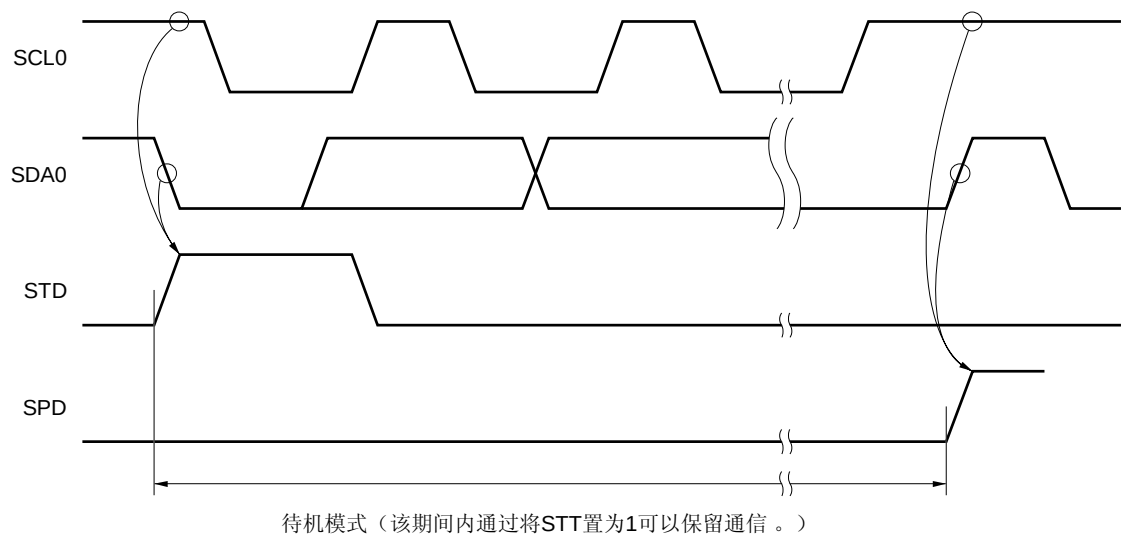
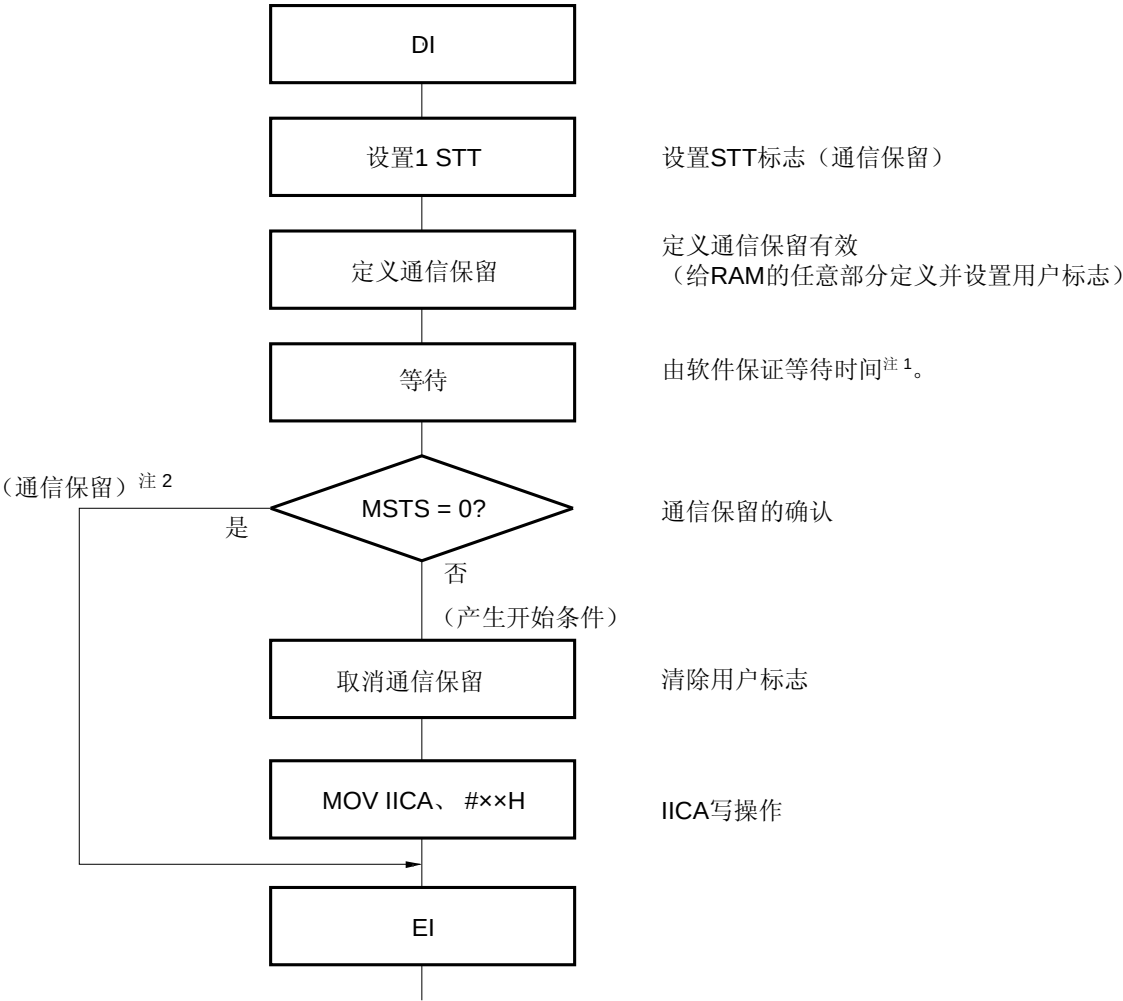


图 13-27 给出了通信保留协议。

图 13-27. 通信保留协议



- 注
1. 等待时间计算如下。
$$(\text{IICWL 设定值} + \text{IICWH 设定值} + 4 \text{ (时钟)}) \times f_{\text{CLK}} + t_{\text{F}} \times 2$$
 2. 当发生停止条件中断请求时，通信保留操作执行对 IICA 移位寄存器（IICA）的写入。

备注

STT: IICA 控制寄存器 0（IICCTL0）的位 1

MSTS: IICA 状态寄存器（IICS）的位 7

IICA: IICA 移位寄存器

IICWL: IICA 低电平宽度设置寄存器

IICWH: IICA 高电平宽度设置寄存器

t_F: SDA0 和 SCL0 信号下降时间（参见第 28 章 电气规范（目标））

f_{CLK}: CPU/外围硬件时钟频率

(2) 禁止通信保留功能时 (IICA 标志寄存器 (IICF) 的位 0 (IICRSV) = 1)

在总线通信期间通信中不使用总线时，在将 IICA 控制寄存器 0 (IICCTL0) 的位 1 (STT) 置为 1 时，请求被拒绝，且不会产生开始条件。不能使用总线的情况包括以下两种状态。

- 仲裁结果既不是主机操作也不是从机操作时
- 接收扩展码并禁止从机操作时 (未返回 $\overline{\text{ACK}}$ ，通过将 IICCTL0 的位 6 (LREL) 置为 1 并保存通信从而解除总线)。

为了确认是产生了开始条件还是请求遭到拒绝，需检查 STCF (IICF 的位 7)。设定 STT = 1 后，将 STCF 置为 1 前会花费多达 5 个时钟。因而，需由软件保证该时间。

13.5.15 注意事项

(1) STCEN (IICA 标志寄存器 (IICF) 的位 1) = 0 时

紧随 I²C 操作使能之后 (IICE = 1)，不管总线的实际状态如何，总线通信状态 (IICBSY (IICF 的位 6) = 1) 总是得到认可。当从没有检测到停止条件的模式变为主机设备通信模式时，首先产生停止条件从而解除总线，然后执行主机设备通信。

使用多个主机时，总线没有解除时 (未检测到停止条件时) 不能执行主机设备通信。

使用以下次序产生停止条件。

- <1> 设置 IICA 控制寄存器 1 (IICCTL1)。
- <2> 将 IICA 控制寄存器 0 (IICCTL0) 的位 7 (IICE) 置为 1。
- <3> 将 IICCTL0 的位 0 (SPT) 置为 1。

(2) STCEN = 1 时

紧随 I²C 操作使能之后 (IICE = 1)，不管总线的实际状态如何，总线释放状态 (IICBSY = 0) 总是得到认可。要产生首个开始条件 (STT (IICA 控制寄存器 0 (IICCTL0) 的位 1) = 1)，有必要确认总线是否已经解除，以便不干扰其它通信。

(3) 如果其它 I²C 通信已在处理中

当 SDA0 引脚为低电平而 SCL0 引脚为高电平时，如果 I²C 操作使能且通信中的设备已在处理中，则 I²C 的宏认为 SDA0 引脚已经变为低电平 (检测开始条件)。如果此时总线的值被认作扩展码，则返回 $\overline{\text{ACK}}$ ，但是这会干扰其它 I²C 通信。要避免这些，需按以下次序开始 I²C。

- <1> 将 IICCTL0 的位 4 (SPIE) 清除为 0，从而在检测到停止条件时禁止中断请求信号 (INTIICA) 的产生。
- <2> 将 IICCTL0 的位 7 (IICE) 置为 1，以允许 I²C 的操作。
- <3> 等待开始条件检测。
- <4> 返回 $\overline{\text{ACK}}$ (将 IICE 置为 1 后的 4 至 8 个时钟) 之前将 IICCTL0 的位 6 (LREL) 置为 1，以强制禁止检测。

(4) 在设置了 STT 和 SPT (IICCTL0 的位 1 和位 0) 之后和将它们清除为 0 之前，禁止再次对其进行设置。

(5) 当保留发送时，将 SPIE (IICCTL0 的位 4) 置为 1，以便于在检测到停止条件时产生中断请求。当在产生中断请求后将通信数据写入 IICA 时，开始传输。当检测到停止条件时，如果不产生中断，在等待状态中设备就会停止，这是因为在开始通信时不产生中断请求。但是，没有必要在软件检测到 MSTs (IICS 的位 7) 时将 SPIE 置为 1。

13.5.16 通信操作

以下展示三种操作流程。

(1) 单主机系统中的主机操作

当使用 78K0R/Kx3-L 单片机作为单主机系统中的主机设备时，操作流程图如下所示。

该操作流程可以宽泛的分为初始设置和通信处理两部分。启动时执行初始设置。如果需要和从机进行通信，则进行通信准备后执行通信处理。

(2) 多主机系统中的主机操作

在 I²C 总线多主机系统中，当总线参与到通信中时，不能由 I²C 总线规范判断总线是解除还是被使用。这里，当数据线和时钟都在一定的时间（1 帧）里保持高电平时，78K0R/Kx3-L 就与解除状态下的总线一起参与通信。

该操作流程图可以宽泛的分为初始设置、通信等待和通信处理。当 78K0R/Kx3-L 仲裁失败并被认定为从机设备时可以在这省略处理，而只是显示作为主机的处理操作。启动时执行初始设置以便参与通信。之后，作为主机设备等待通信请求或作为从机设备等待指定。在通信处理操作中进行实际通信，并且支持与从机间的发送/接收操作和与其它主机间的仲裁处理。

(3) 从机操作

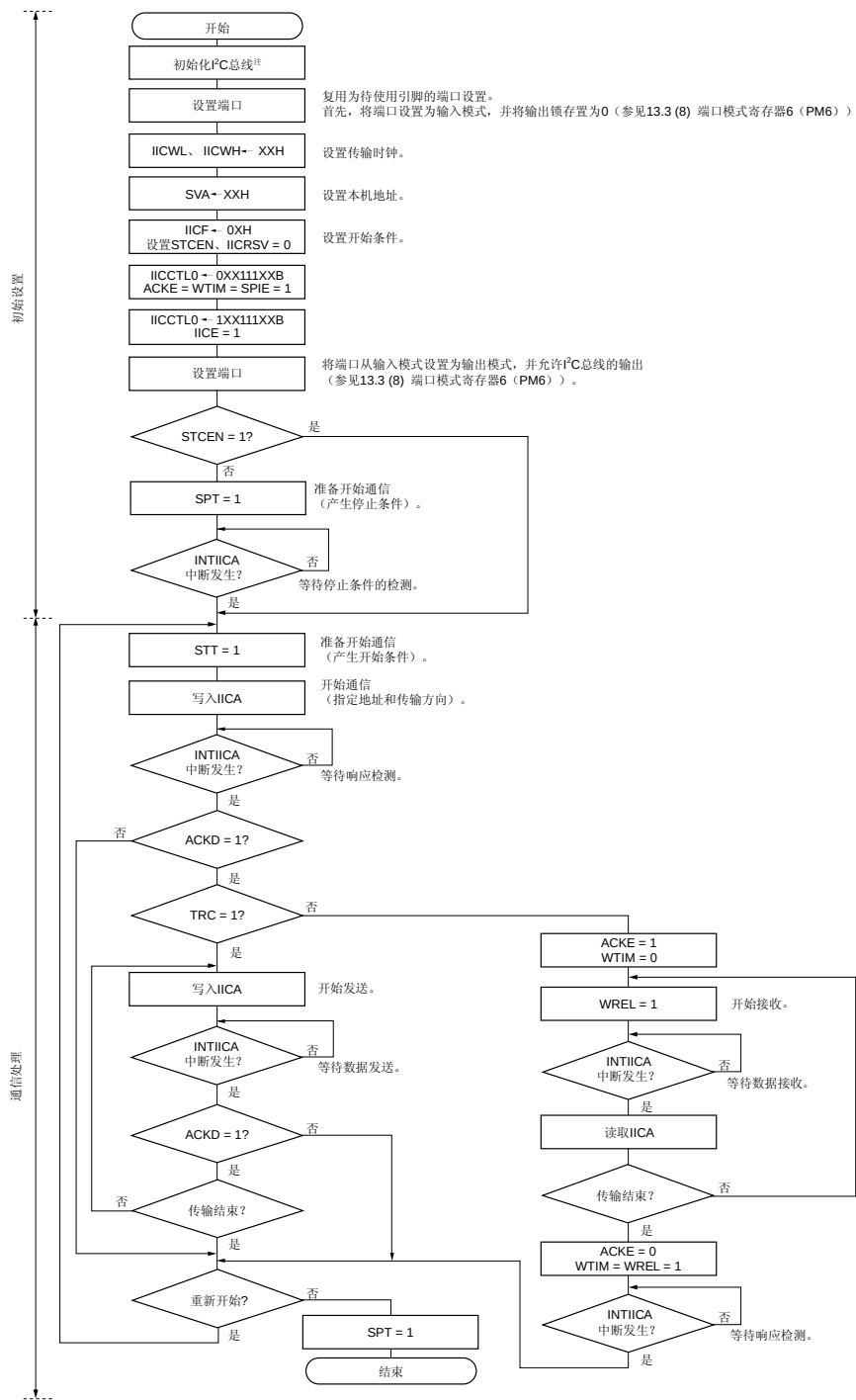
78K0R/Kx3-L 用作 I²C 总线从机设备时的例子如下所示。

当用作从机时，由一个中断启动操作。启动时执行初始设置，之后，等待 INTIICA 中断发生（通信等待）。当 INTIICA 中断发生时，判断通信状态，并且其判决结果作为一个标志传送给主处理程序。

通过检查该标志，执行必要的通信处理。

(1) 单主机系统中的主机操作

图 13-28. 单主机系统中的主机操作

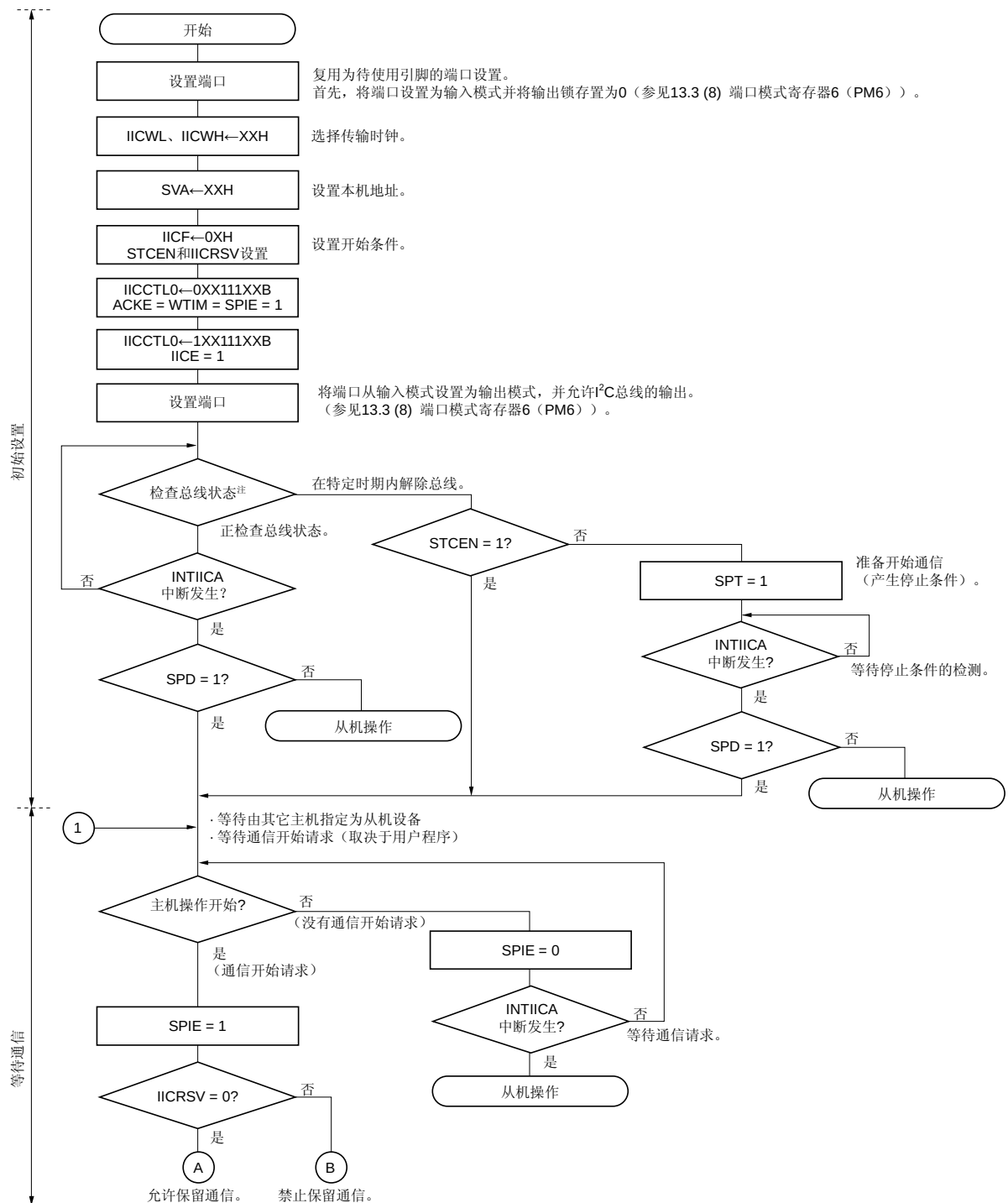


注 依照正通信的产品的规范解除（SCL0 和 SDA0 引脚 = 高电平）I²C总线。例如，如果EEPROM正向SDA0 引脚输出一低电平，在输出端口模式设置SCL0 引脚，并从输出端口输出一个时钟脉冲，直到SDA0 引脚固定为高电平。

备注 至于发送和接收的格式，要遵循正进行通信的产品的规范。

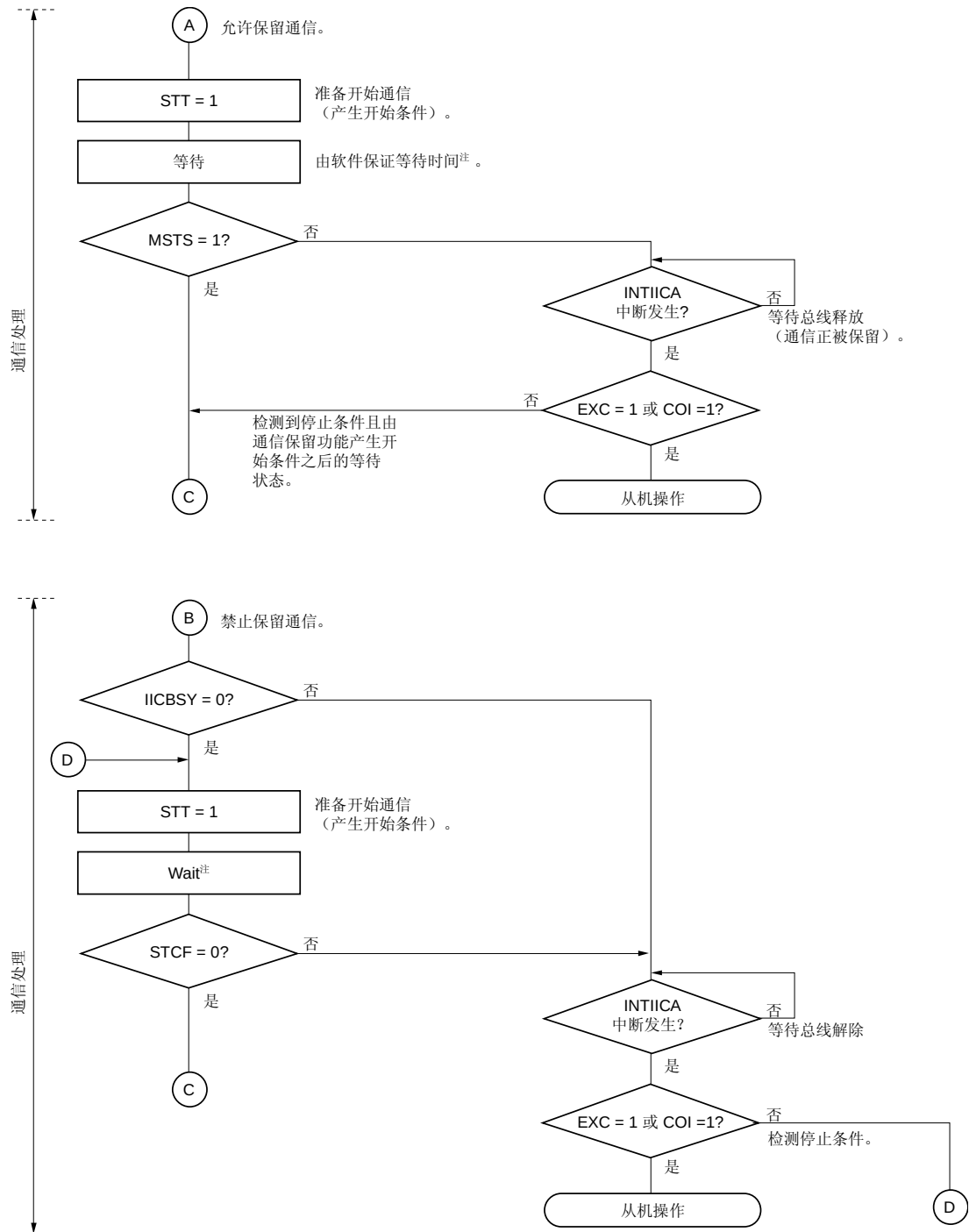
(2) 多主机系统中的主机操作

图 13-29. 多主机系统中的主机操作 (1/3)



注 确定特定时期内总线解除（CLD位 = 1，DAD位 = 1）（例如，一帧时间。如果SDA0引脚固定为低电平，依照正在进行通信的产品的规范，决定是否解除I²C总线（SCL0和SDA0引脚 = 高电平）。

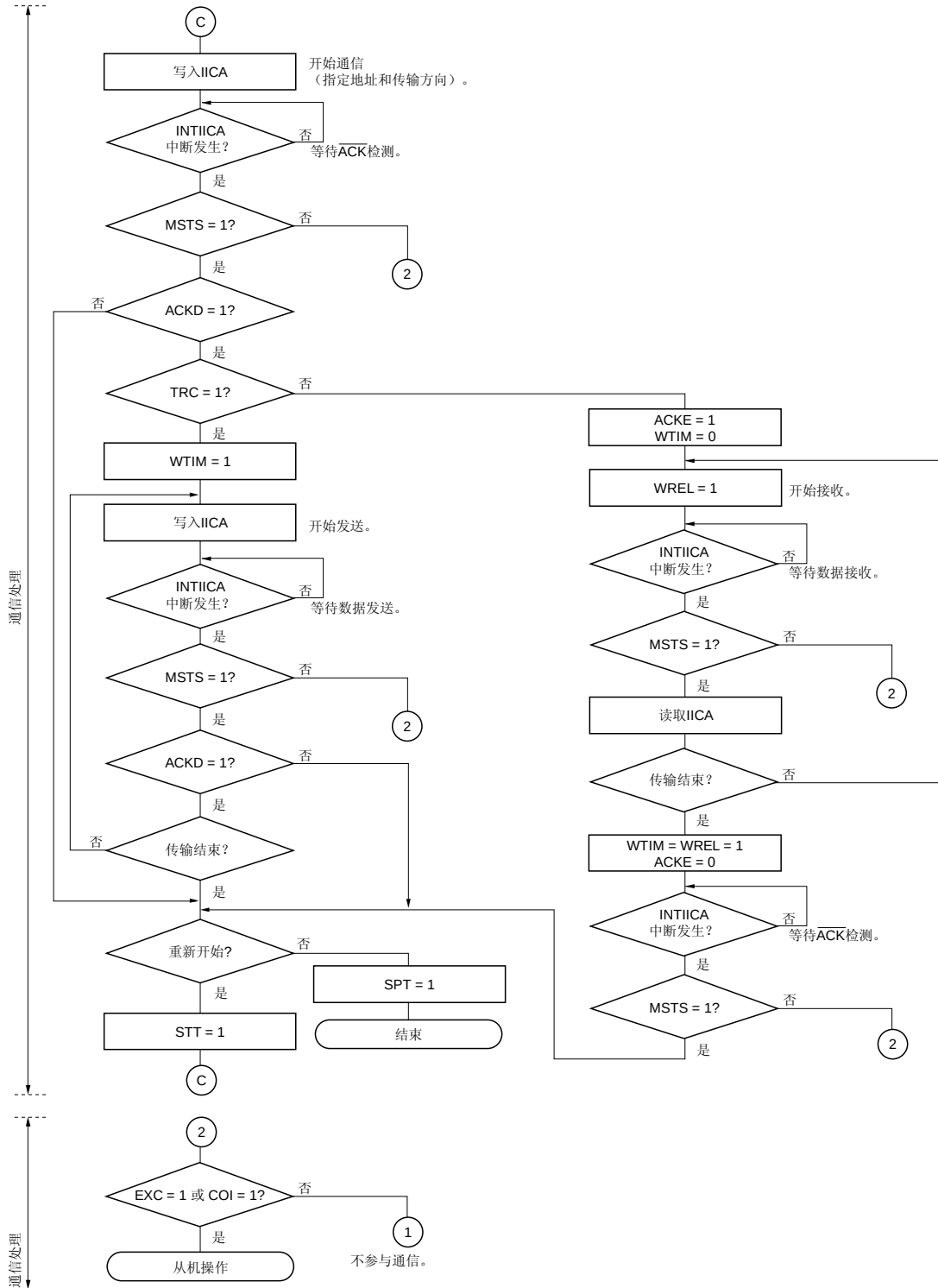
图 13-29. 多主机系统中的主机操作（2/3）



注 等待时间计算如下。
(IICWL 设定值 + IICWH 设定值 + 4 (时钟)) × fCLK + tF × 2

备注 IICWL: IICA 低电平宽度设置寄存器
IICWH: IICA 高电平宽度设置寄存器
tF: SDA0 和 SCL0 信号下降时间 (参见第 28 章 电气规范 (目标))
fCLK: CPU/外围硬件时钟频率

图 13-29. 多主机系统中的主机操作 (3/3)

**备注**

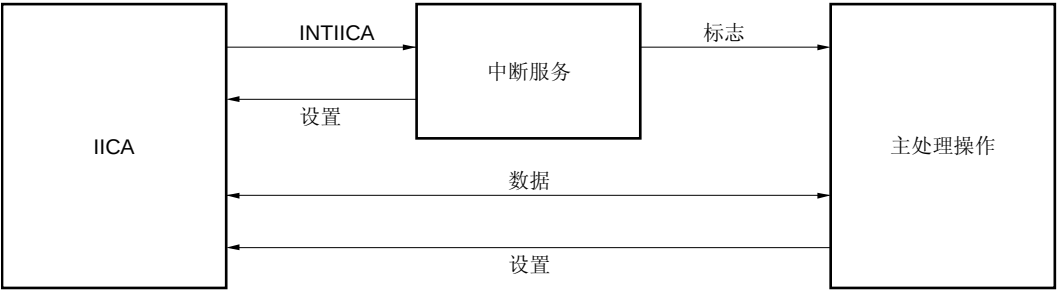
1. 至于发送和接收的格式，要遵循正进行通信的产品的规范。
2. 要将设备用作多主机系统中的主机设备，需在每次发生中断 INTIICA 时读取 MSTS 位，以检查仲裁结果。
3. 要将设备用作多主机系统中的从机设备，需在每次发生中断 INTIICA 时使用 IICS 和 IICF 寄存器检查状态，并确定下一步要执行的处理操作。

(3) 从机操作

从机操作的处理步骤如下。

基本而言，从机操作是由事件驱动的。因此，需要由 INTIICA 中断进行处理（需要彻底改变操作状态的处理，比如通信过程中停止条件的检测）。

在下述说明中，假定扩展码不支持数据通信。还假设 INTIICA 中断服务仅执行转变处理，且实际数据通信由主程序执行。



因此，通过准备以下三种标志并将其代替 INTIICA 传送至主处理程序，从而执行数据通信处理。

<1> 通信模式标志

该标志指示以下两种通信状态。

- 清除模式：未执行数据通信的状态
- 通信模式：执行数据通信的状态（从有效地址到停止条件检测，没有来自主机的 $\overline{\text{ACK}}$ 检测，地址不匹配）

<2> 就绪标志

该标志指示允许数据通信。它的功能与普通数据通信的 INTIICA 中断相同。由中断服务设置该标志，由主处理操作清除该标志。开始通信时由中断服务清除该标志。但是，发送第一个数据时不能由中断服务设置就绪标志。因此，在不清除标志的情况下，发送第一个数据（地址匹配解释为下一个数据的请求信号）。

<3> 通信方向标志

该标志指示通信的方向。它的值与 TRC 的相同。

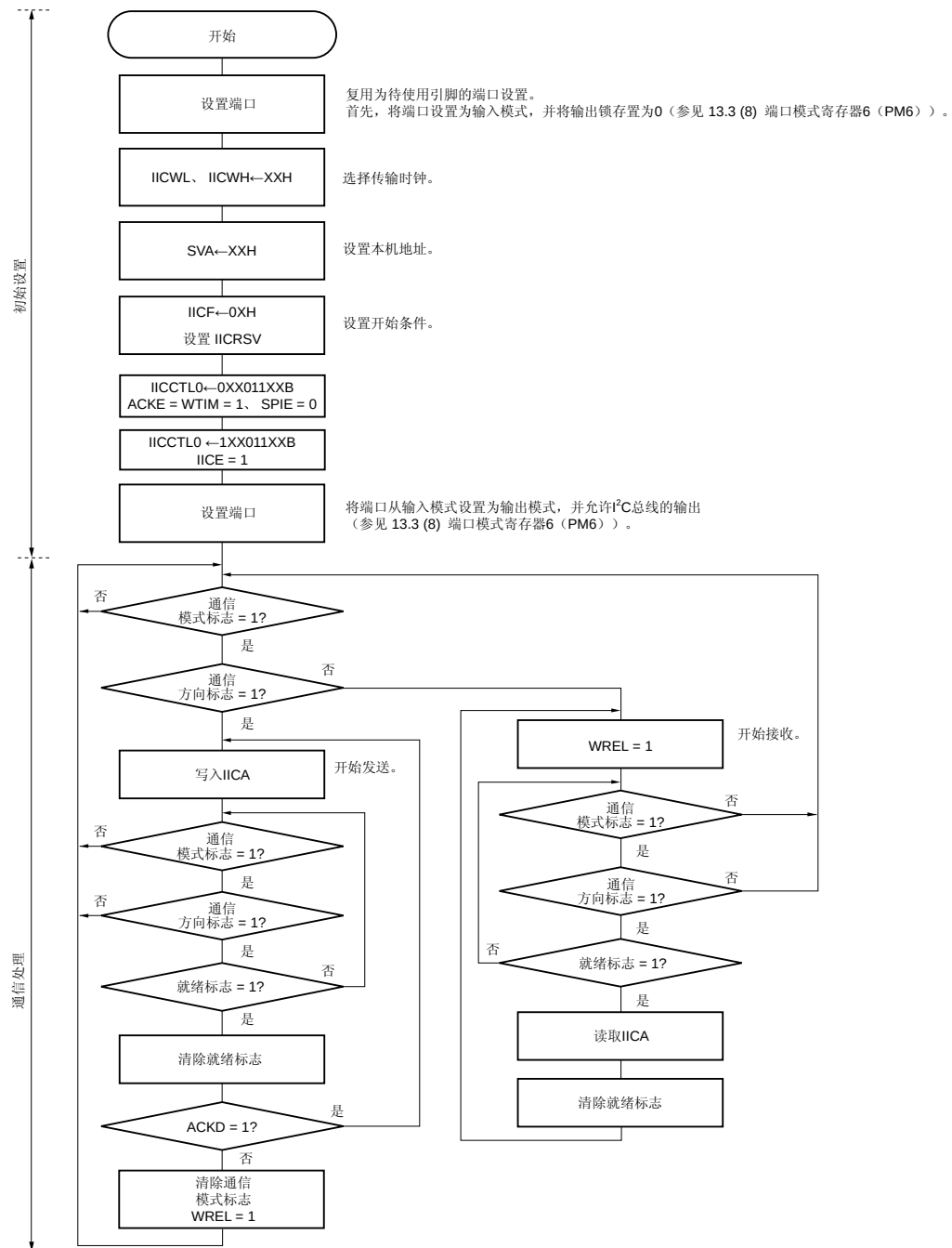
从机操作的主处理说明如下。

允许开始串行接口 IICA 和等待通信。通信使能后时，使用通信模式标志和就绪标志执行通信（停止条件和开始条件的处理由中断处理执行。这里，使用标志检查状态）。

主机不再返回 $\overline{\text{ACK}}$ 之前一直重复发送操作。如果 $\overline{\text{ACK}}$ 没有从主机返回，则通信结束。

对于接收，要接收必需量的数据。通信完成时， $\overline{\text{ACK}}$ 不作为下一数据返回。此后，主机产生停止条件或重新开始条件。中途从通信状态中退出。

图 13-30. 从机操作流程图（1）

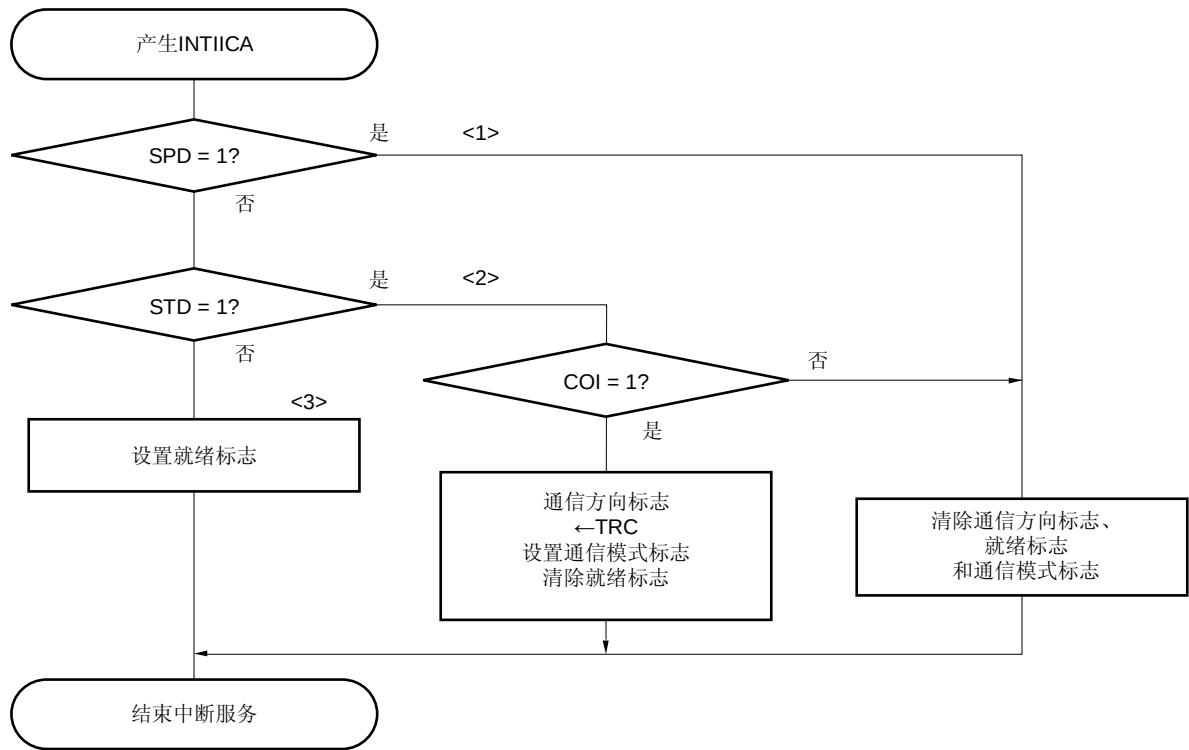


使用 INTIICA 中断的从机处理步骤的示例说明如下（假定未使用扩展码时执行处理）。INTIICA 中断检查状态，并执行以下操作。

- <1> 如果产生停止条件，停止通信。
- <2> 如果产生开始条件，地址不匹配时检查地址并结束通信。如果地址匹配，设置通信模式并取消等待状态，处理程序从中断中返回（清除就绪标志）。
- <3> 对于数据发送/接收，仅设置就绪标志。在等待状态中使用剩余的 I²C 总线使处理从中断中返回。

备注 上面的<1>至<3>对应着图 13-31 从机操作流程图（2）中的<1>至<3>。

图 13-31. 从机操作流程图（2）



13.5.17 I²C中断请求（INTIICA）发生的时序

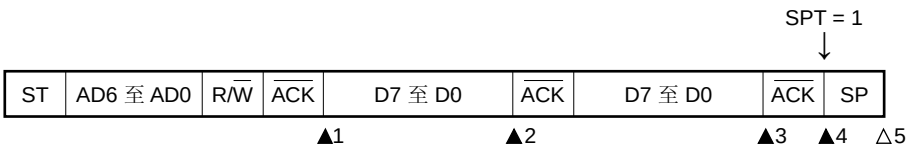
发送或接收数据的时序和中断请求信号 INTIICA 的产生时序，以及产生 INTIICA 信号时 IICS 寄存器的值如下所示。

备注	ST:	开始条件
	AD6 至 AD0:	地址
	R/ $\overline{W}$:	传输方向指定
	$\overline{ACK}$:	响应
	D7 至 D0:	数据
	SP:	停止条件

(1) 主机设备操作

(a) 开始 ~ 地址 ~ 数据 ~ 数据 ~ 停止（发送/接收）

(i) WTIM = 0 时



- ▲1: IICS = 1000×110B
▲2: IICS = 1000×000B
▲3: IICS = 1000×000B (Sets WTIM to 1)[※]
▲4: IICS = 1000××00B (Sets SPT to 1)[※]
△5: IICS = 00000001B

注 要产生停止条件，需将 WTIM 置为 1，并改变产生 INTIICA 中断请求信号的时序。

备注 ▲: 总是产生
△: 仅当 SPIE = 1 时产生
×: 可忽略

(ii) WTIM = 1 时

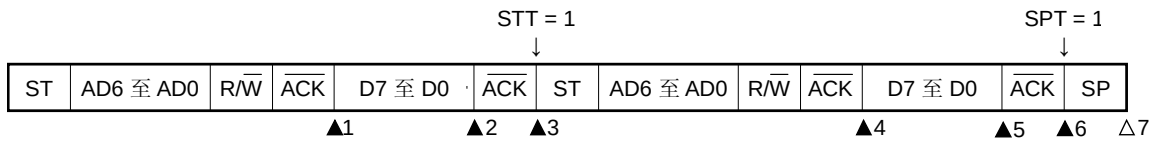


- ▲1: IICS = 1000×110B
▲2: IICS = 1000×100B
▲3: IICS = 1000××00B (将 SPT 置为 1)
△4: IICS = 00000001B

备注 ▲: 总是产生
△: 仅当 SPIE = 1 时产生
×: 可忽略

(b) 开始 ~ 地址 ~ 数据 ~ 开始 ~ 地址 ~ 数据 ~ 停止 (重新开始)

(i) WTIM = 0 时

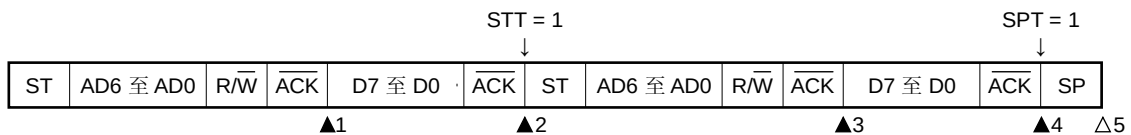


- ▲1: IICS = 1000×110B
 ▲2: IICS = 1000×000B (将WTIM置为 1)^{注1}
 ▲3: IICS = 1000××00B (将WTIM清除为 0^{注2}, 将STT置为 1)
 ▲4: IICS = 1000×110B
 ▲5: IICS = 1000×000B (将WTIM置为 1)^{注3}
 ▲6: IICS = 1000××00B (将 SPT 置为 1)
 △7: IICS = 00000001B

注 1. 要产生开始条件, 需将 WTIM 置为 1, 并改变产生 INTIICA 中断请求信号的时序。
 2. 将 WTIM 清除为 0, 以恢复初始设置。
 3. 要产生停止条件, 需将 WTIM 置为 1, 并改变产生 INTIICA 中断请求信号的时序。

备注 ▲: 总是产生
 △: 仅当 SPIE = 1 时产生
 ×: 可忽略

(ii) WTIM = 1 时

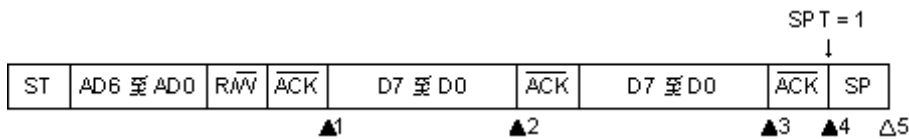


- ▲1: IICS = 1000×110B
 ▲2: IICS = 1000××00B (将 STT 置为 1)
 ▲3: IICS = 1000×110B
 ▲4: IICS = 1000××00B (将 SPT 置为 1)
 △5: IICS = 00000001B

备注 ▲: 总是产生
 △: 仅当 SPIE = 1 时产生
 ×: 可忽略

(c) 开始~扩展码~ 数据~ 数据~ 停止（发送扩展码）

(i) WTIM = 0 时



- ▲1: IICS = 1010×110B
▲2: IICS = 1010×000B
▲3: IICS = 1010×000B (将WTIM置为 1)[#]
▲4: IICS = 1010××00B (将 SPT 置为 1)
△5: IICS = 00000001B

注 要产生停止条件，需将 WTIM 置为 1，并改变产生 INTIICA 中断请求信号的时序。

备注 ▲: 总是产生
△: 仅当 SPIE = 1 时产生
×: 可忽略

(ii) WTIM = 1 时



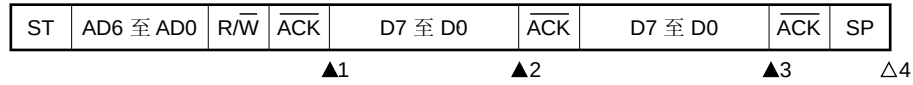
- ▲1: IICS = 1010×110B
▲2: IICS = 1010×100B
▲3: IICS = 1010××00B (将 SPT 置为 1)
△4: IICS = 00001001B

备注 ▲: 总是产生
△: 仅当 SPIE = 1 时产生
×: 可忽略

(2) 从机设备操作（从地址数据接收）

(a) 开始 ~ 地址 ~ 数据 ~ 数据 ~ 停止

(i) WTIM = 0 时



▲1: IICS = 0001×110B

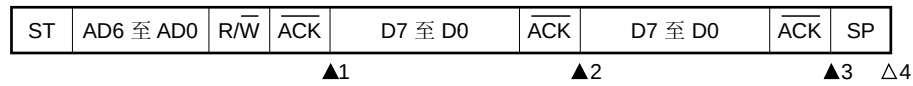
▲2: IICS = 0001×000B

▲3: IICS = 0001×000B

△4: IICS = 00000001B

备注 ▲: 总是产生
 △: 仅当 SPIE = 1 时产生
 ×: 可忽略

(ii) WTIM = 1 时



▲1: IICS = 0001×110B

▲2: IICS = 0001×100B

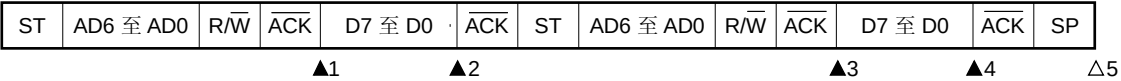
▲3: IICS = 0001××00B

△4: IICS = 00000001B

备注 ▲: 总是产生
 △: 仅当 SPIE = 1 时产生
 ×: 可忽略

(b) 开始 ~ 地址 ~ 数据 ~ 开始 ~ 地址 ~ 数据 ~ 停止

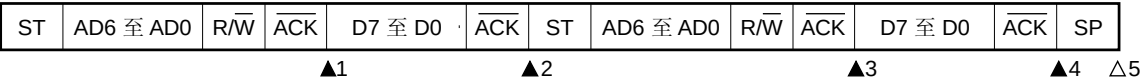
(i) **WTIM = 0** 时（重新开始后，与 **SVA** 匹配）



- ▲1: IICS = 0001×110B
▲2: IICS = 0001×000B
▲3: IICS = 0001×110B
▲4: IICS = 0001×000B
△5: IICS = 00000001B

备注 ▲: 总是产生
 △: 仅当 SPIE = 1 时产生
 ×: 可忽略

(ii) **TIM = 1** 时（重新开始后，与 **SVA** 匹配）

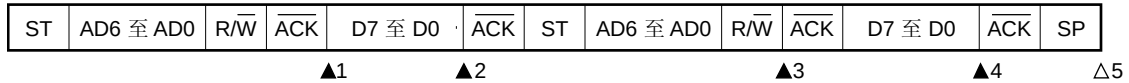


- ▲1: IICS = 0001×110B
▲2: IICS = 0001××00B
▲3: IICS = 0001×110B
▲4: IICS = 0001××00B
△5: IICS = 00000001B

备注 ▲: 总是产生
 △: 仅当 SPIE = 1 时产生
 ×: 可忽略

(c) 开始 ~ 地址 ~ 数据 ~ 开始 ~ 扩展码 ~ 数据 ~ 停止

(i) $WTIM = 0$ 时 (重新开始后, 地址不匹配 (=扩展码))



▲1: IICS = 0001×110B

▲2: IICS = 0001×000B

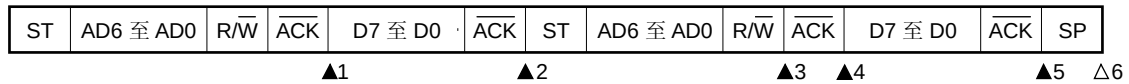
▲3: IICS = 0010×010B

▲4: IICS = 0010×000B

△5: IICS = 00000001B

备注 ▲: 总是产生
 △: 仅当 $SPIE = 1$ 时产生
 ×: 可忽略

(ii) $WTIM = 1$ 时 (重新开始后, 地址不匹配 (=扩展码))



▲1: IICS = 0001×110B

▲2: IICS = 0001××00B

▲3: IICS = 0010×010B

▲4: IICS = 0010×110B

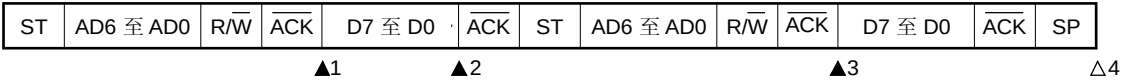
▲5: IICS = 0010××00B

△6: IICS = 00000001B

备注 ▲: 总是产生
 △: 仅当 $SPIE = 1$ 时产生
 ×: 可忽略

(d) 开始 ~ 地址 ~ 数据 ~ 开始 ~ 地址 ~ 数据 ~ 停止

(i) $WTIM = 0$ 时（重新开始后，地址不匹配（=非扩展码））



▲1: IICS = 0001×110B

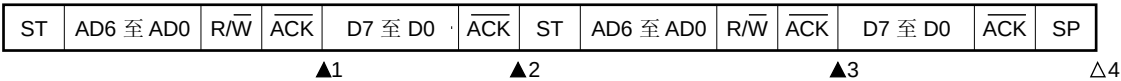
▲2: IICS = 0001×000B

▲3: IICS = 00000110B

△4: IICS = 00000001B

备注 ▲: 总是产生
 △: 仅当 $SPIE = 1$ 时产生
 ×: 可忽略

(ii) $WTIM = 1$ 时（重新开始后，地址不匹配（=非扩展码））



▲1: IICS = 0001×110B

▲2: IICS = 0001××00B

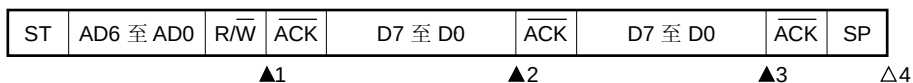
▲3: IICS = 00000110B

△4: IICS = 00000001B

备注 ▲: 总是产生
 △: 仅当 $SPIE = 1$ 时产生
 ×: 可忽略

(3) 从机设备操作（接收扩展码时）

设备接收扩展码时总是参与在通信中。

(a) 开始 ~ 扩展码 ~ 数据 ~ 数据 ~ 停止**(i) WTIM = 0 时**

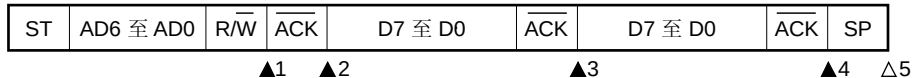
▲1: IICS = 0010×010B

▲2: IICS = 0010×000B

▲3: IICS = 0010×000B

△4: IICS = 00000001B

备注 ▲: 总是产生
 △: 仅当 SPIE = 1 时产生
 ×: 可忽略

(ii) WTIM = 1 时

▲1: IICS = 0010×010B

▲2: IICS = 0010×110B

▲3: IICS = 0010×100B

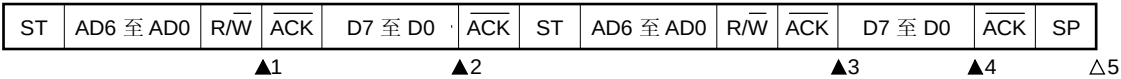
▲4: IICS = 0010××00B

△5: IICS = 00000001B

备注 ▲: 总是产生
 △: 仅当 SPIE = 1 时产生
 ×: 可忽略

(b) 开始 ~ 扩展码 ~ 数据 ~ 开始 ~ 地址 ~ 数据 ~ 停止

(i) **WTIM = 0** 时（重新开始后，与 **SVA** 匹配）



- ▲1: IICS = 0010×010B
▲2: IICS = 0010×000B
▲3: IICS = 0001×110B
▲4: IICS = 0001×000B
△5: IICS = 00000001B

备注 ▲: 总是产生
 △: 仅当 SPIE = 1 时产生
 ×: 可忽略

(ii) **WTIM = 1** 时（重新开始后，与 **SVA** 匹配）

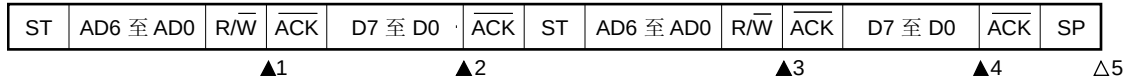


- ▲1: IICS = 0010×010B
▲2: IICS = 0010×110B
▲3: IICS = 0010××00B
▲4: IICS = 0001×110B
▲5: IICS = 0001××00B
△6: IICS = 00000001B

备注 ▲: 总是产生
 △: 仅当 SPIE = 1 时产生
 ×: 可忽略

(c) 开始 ~ 扩展码 ~ 数据 ~ 开始 ~ 扩展码 ~ 数据 ~ 停止

(i) $WTIM = 0$ 时 (重新开始后, 接收扩展码)



▲1: IICS = 0010×010B

▲2: IICS = 0010×000B

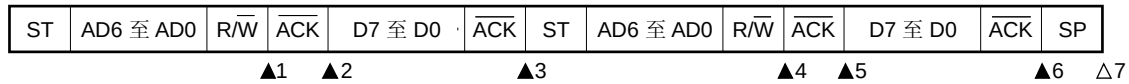
▲3: IICS = 0010×010B

▲4: IICS = 0010×000B

△5: IICS = 00000001B

备注 ▲: 总是产生
 △: 仅当 $SPIE = 1$ 时产生
 ×: 可忽略

(ii) $WTIM = 1$ 时 (重新开始后, 接收扩展码)



▲1: IICS = 0010×010B

▲2: IICS = 0010×110B

▲3: IICS = 0010××00B

▲4: IICS = 0010×010B

▲5: IICS = 0010×110B

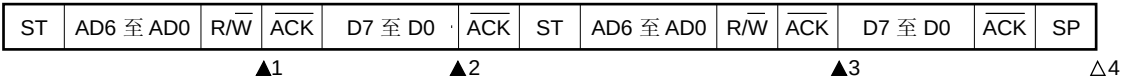
▲6: IICS = 0010××00B

△7: IICS = 00000001B

备注 ▲: 总是产生
 △: 仅当 $SPIE = 1$ 时产生
 ×: 可忽略

(d) 开始 ~ 扩展码 ~ 数据 ~ 开始 ~ 地址 ~ 数据 ~ 停止

(i) **WTIM = 0** 时（重新开始后，地址不匹配（=非扩展码））



▲1: IICS = 00100010B

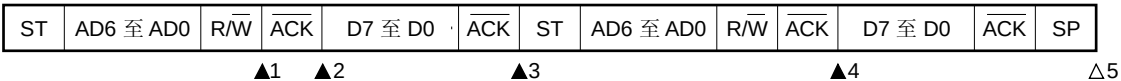
▲2: IICS = 00100000B

▲3: IICS = 00000110B

△4: IICS = 00000001B

备注 ▲: 总是产生
 △: 仅当 SPIE = 1 时产生
 ×: 可忽略

(ii) **WTIM = 1** 时（重新开始后，地址不匹配（=非扩展码））



▲1: IICS = 00100010B

▲2: IICS = 00100110B

▲3: IICS = 00100×00B

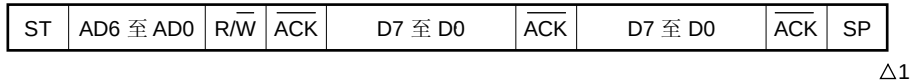
▲4: IICS = 00000110B

△5: IICS = 00000001B

备注 ▲: 总是产生
 △: 仅当 SPIE = 1 时产生
 ×: 可忽略

(4) 不通信时的操作

(a) 开始 ~ 扩展码 ~ 数据 ~ 数据 ~ 停止



△1: IICS = 00000001B

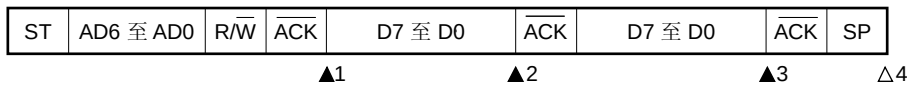
备注 △: 仅当 SPIE = 1 时产生

(5) 仲裁失败操作（仲裁失败后作为从机设备的操作）

当设备用作多主机系统中的主机设备时，需在每次发生中断请求信号 INTIICA 时读取 MSTs 位，以检查仲裁结果。

(a) 在从地址数据发送期间发生仲裁失败时

(i) WTIM = 0 时



▲1: IICS = 0101×110B

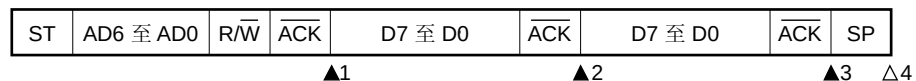
▲2: IICS = 0001×000B

▲3: IICS = 0001×000B

△4: IICS = 00000001B

备注 ▲: 总是产生
 △: 仅当 SPIE = 1 时产生
 ×: 可忽略

(ii) **WTIM = 1 时**

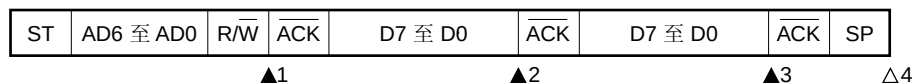


- ▲1: IICS = 0101×110B
 ▲2: IICS = 0001×100B
 ▲3: IICS = 0001××00B
 △4: IICS = 00000001B

备注 ▲: 总是产生
△: 仅当 SPIE = 1 时产生
×: 可忽略

(b) 在扩展码发送期间发生仲裁失败时

(i) $WTIM = 0$ 时



- ▲1: IICS = 0110×010B
 ▲2: IICS = 0010×000B
 ▲3: IICS = 0010×000B
 △4: IICS = 00000001B

备注 ▲: 总是产生
△: 仅当 SPIE = 1 时产生
x: 可忽略

(ii) WTIM = 1 时

ST	AD6 至 AD0	R/W	ACK	D7 至 D0	ACK	D7 至 D0	ACK	SP
			▲1 ▲2		▲3		▲4	△5

▲1: IICS = 0110×010B

▲2: IICS = 0010×110B

▲3: IICS = 0010×100B

▲4: IICS = 0010××00B

△5: IICS = 00000001B

备注 ▲: 总是产生
 △: 仅当 SPIE = 1 时产生
 ×: 可忽略

(6) 仲裁失败发生时的操作（在仲裁失败后不进行通信）

当设备用作多主机系统中的主机设备时，需在每次发生中断请求信号 INTIICA 时读取 MST5 位，以检查仲裁结果。

(a) 在从地址数据发送期间发生仲裁失败时（WTIM = 1 时）

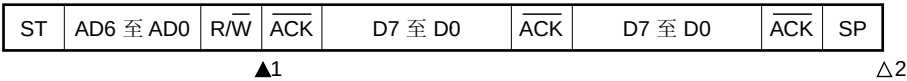
ST	AD6 至 AD0	R/W	ACK	D7 至 D0	ACK	D7 至 D0	ACK	SP
			▲1					△2

▲1: IICS = 01000110B

△2: IICS = 00000001B

备注 ▲: 总是产生
 △: 仅当 SPIE = 1 时产生

(b) 在扩展码发送期间发生仲裁失败时

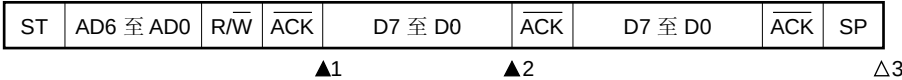


▲1: IICS = 0110×010B
由软件设置 LREL = 1
△2: IICS = 00000001B

备注 ▲: 总是产生
 △: 仅当 SPIE = 1 时产生
 ×: 可忽略

(c) 在数据发送期间发生仲裁失败时

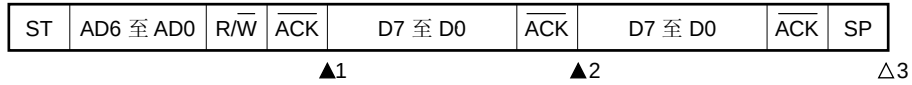
(i) WTIM = 0 时



▲1: IICS = 10001110B
▲2: IICS = 01000000B
△3: IICS = 00000001B

备注 ▲: 总是产生
 △: 仅当 SPIE = 1 时产生

(ii) WTIM = 1 时



▲1: IICS = 10001110B

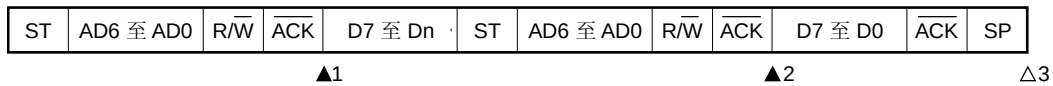
▲2: IICS = 01000100B

△3: IICS = 00000001B

备注 ▲: 总是产生
 △: 仅当 SPIE = 1 时产生

(d) 在数据传输期间，由于重新开始条件导致发生仲裁失败时

(i) 非扩展码（例如：与 SVA 不匹配）



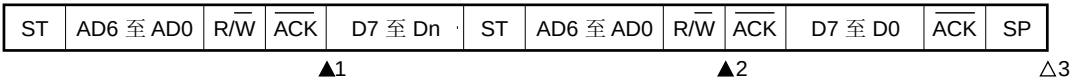
▲1: IICS = 1000×110B

▲2: IICS = 01000110B

△3: IICS = 00000001B

备注 ▲: 总是产生
 △: 仅当 SPIE = 1 时产生
 ×: 可忽略
 n = 6 至 0

(ii) 扩展码



▲1: IICS = 1000×110B

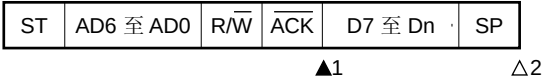
▲2: IICS = 01100010B

由软件设置 LREL = 1

△3: IICS = 00000001B

备注 ▲: 总是产生
 △: 仅当 SPIE = 1 时产生
 ×: 可忽略
 n = 6 至 0

(e) 在数据传输期间，由于停止条件导致发生仲裁失败时



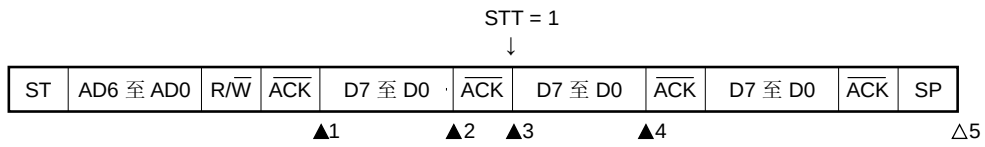
▲1: IICS = 10000110B

△2: IICS = 01000001B

备注 ▲: 总是产生
 △: 仅当 SPIE = 1 时产生
 ×: 可忽略
 n = 6 至 0

(f) 当试图产生重新开始条件时，由于低电平数据而导致发生仲裁失败时

(i) $WTIM = 0$ 时



▲1: IICS = 1000×110B

▲2: IICS = 1000×000B (将 $WTIM$ 置为 1)

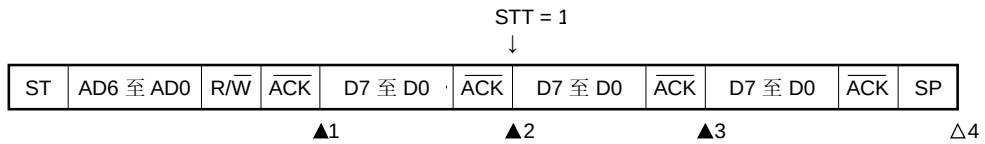
▲3: IICS = 1000×100B (将 $WTIM$ 清除为 0)

▲4: IICS = 01000000B

△5: IICS = 00000001B

备注 ▲: 总是产生
 △: 仅当 $SPIE = 1$ 时产生
 ×: 可忽略

(ii) $WTIM = 1$ 时



▲1: IICS = 1000×110B

▲2: IICS = 1000×100B (将 STT 置为 1)

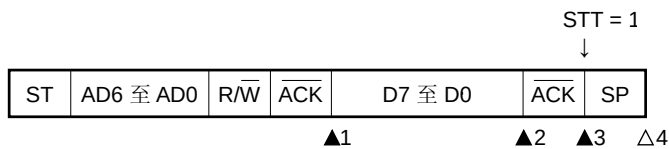
▲3: IICS = 01000100B

△4: IICS = 00000001B

备注 ▲: 总是产生
 △: 仅当 $SPIE = 1$ 时产生
 ×: 可忽略

(g) 当试图产生重新开始条件时，由于停止条件导致发生仲裁失败时

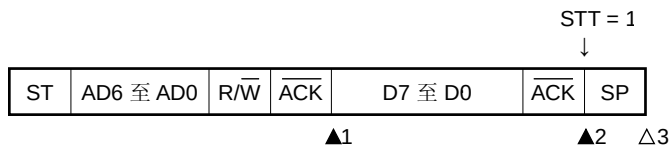
(i) $WTIM = 0$ 时



- ▲1: IICS = 1000×110B
▲2: IICS = 1000×000B (将 WTIM 置为 1)
▲3: IICS = 1000××00B (将 STT 置为 1)
△4: IICS = 01000001B

备注 ▲: 总是产生
 △: 仅当 SPIE = 1 时产生
 ×: 可忽略

(ii) $WTIM = 1$ 时

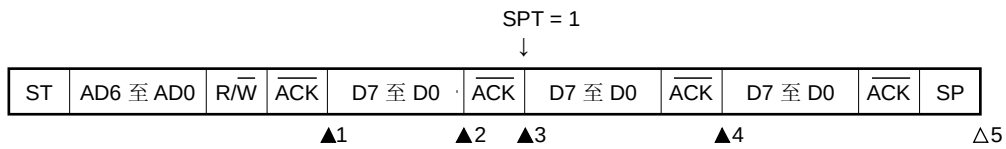


- ▲1: IICS = 1000×110B
▲2: IICS = 1000××00B (将 STT 置为 1)
△3: IICS = 01000001B

备注 ▲: 总是产生
 △: 仅当 SPIE = 1 时产生
 ×: 可忽略

(h) 当试图产生停止条件时，由于低电平数据而导致发生仲裁失败时

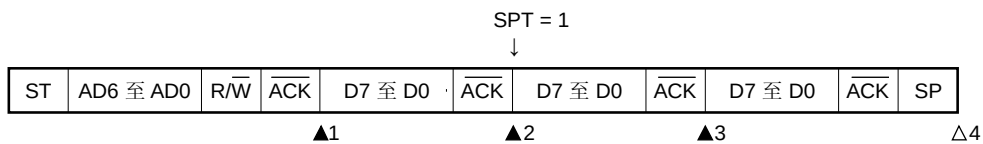
(i) $WTIM = 0$ 时



- ▲1: IICS = 1000×110B
 ▲2: IICS = 1000×000B (将 WTIM 置为 1)
 ▲3: IICS = 1000×100B (将 WTIM 清除为 0)
 ▲4: IICS = 01000100B
 △5: IICS = 00000001B

备注 ▲: 总是产生
 △: 仅当 $SPIE = 1$ 时产生
 ×: 可忽略

(ii) $WTIM = 1$ 时



- ▲1: IICS = 1000×110B
 ▲2: IICS = 1000×100B (将 SPT 置为 1)
 ▲3: IICS = 01000100B
 △4: IICS = 00000001B

备注 ▲: 总是产生
 △: 仅当 $SPIE = 1$ 时产生
 ×: 可忽略

13.6 时序图

当使用 I²C 总线模式时，主机设备通过串行总线输出一个地址，以选择多个从机设备之一作为自己的通信伙伴。

输出从地址以后，主机设备发送 TRC 位（IICA 状态寄存器（IICS）的位 3）指明数据传输方向，之后，开始与从机设备的串行通信。

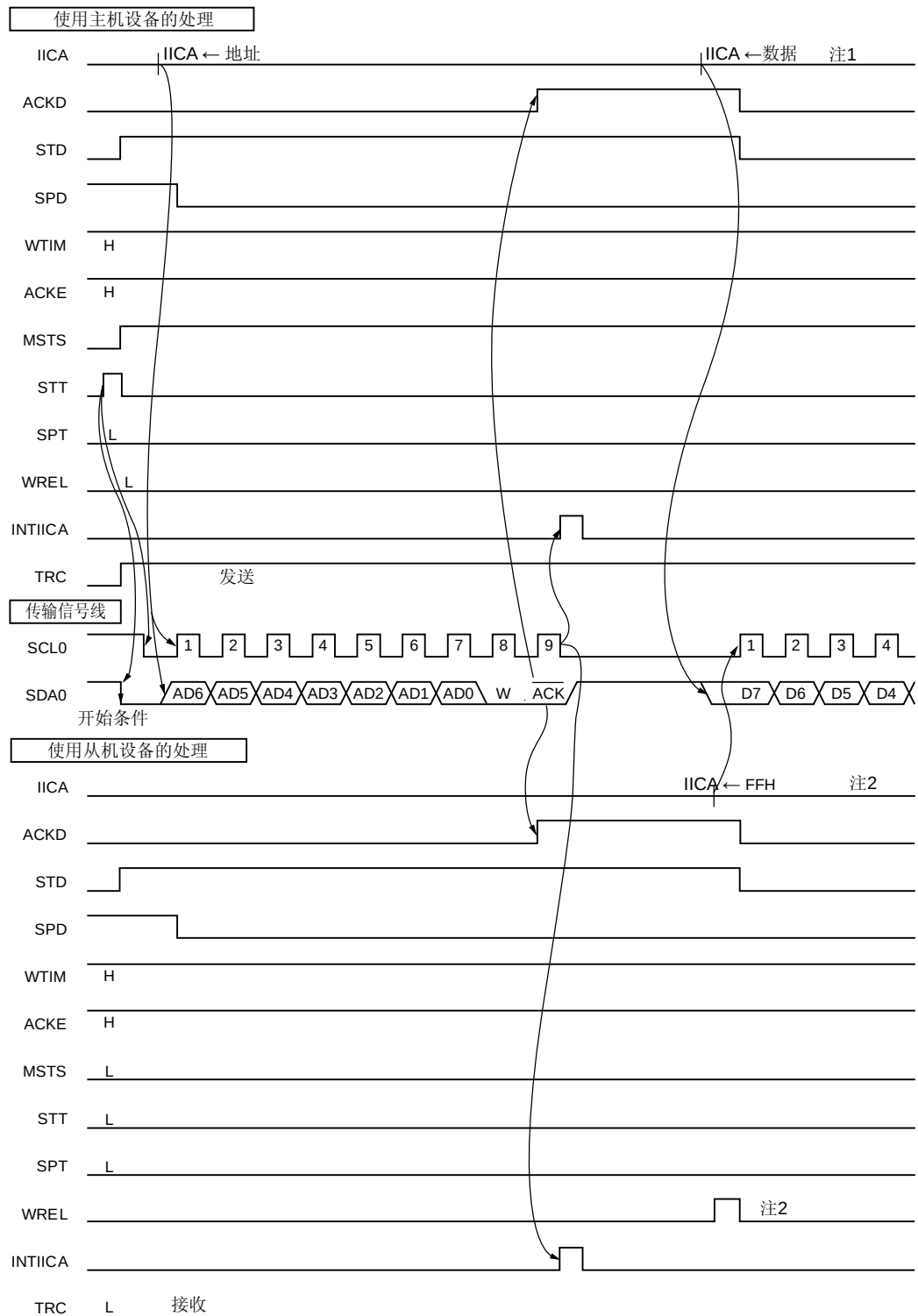
图 13-33 和图 13-34 显示了数据通信的时序图。

IICA 移位寄存器（IICA）的移位操作与串行时钟（SCL0）的下降沿同步。发送数据被传送到 SO 锁存器并经由 SDA0 引脚输出（MSB 先行）。

经 SDA0 引脚输入的数据在 SCL0 的上升沿被捕获到 IICA 中。

图 13-32. 主机向从机通信示例
(主机设备和从机设备都选择 9 时钟等待时) (1/3)

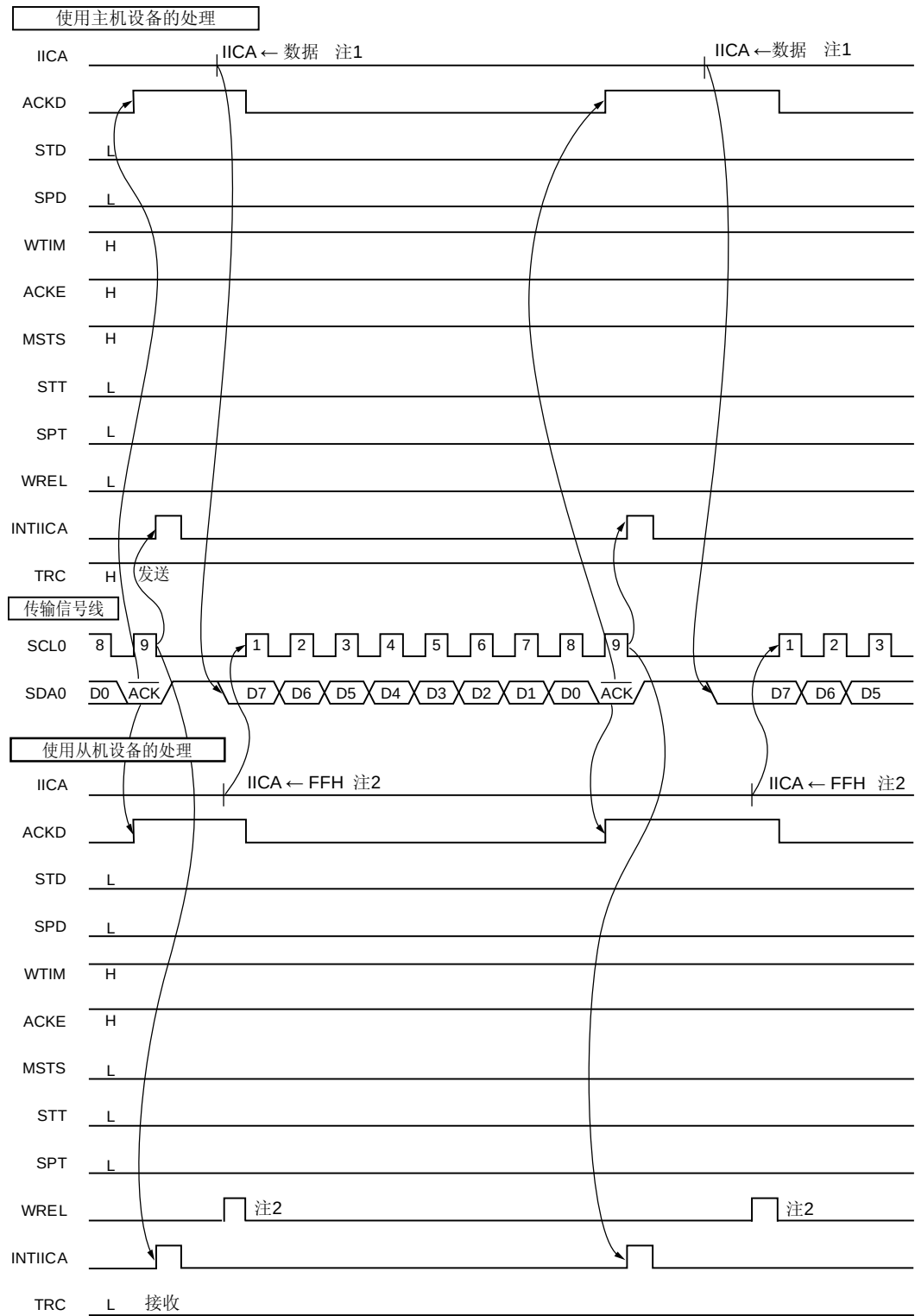
(1) 开始条件 ~ 地址



- 注 1. 为了取消主机发送期间的等待状态，将数据写入 IICA，而不设置 WREL。
2. 要取消从机时的等待状态，需将“FFH”写入 IICA，或者设置 WREL。

图 13-32. 主机向从机通信示例
(主机设备和从机设备都选择 9 时钟等待时) (2/3)

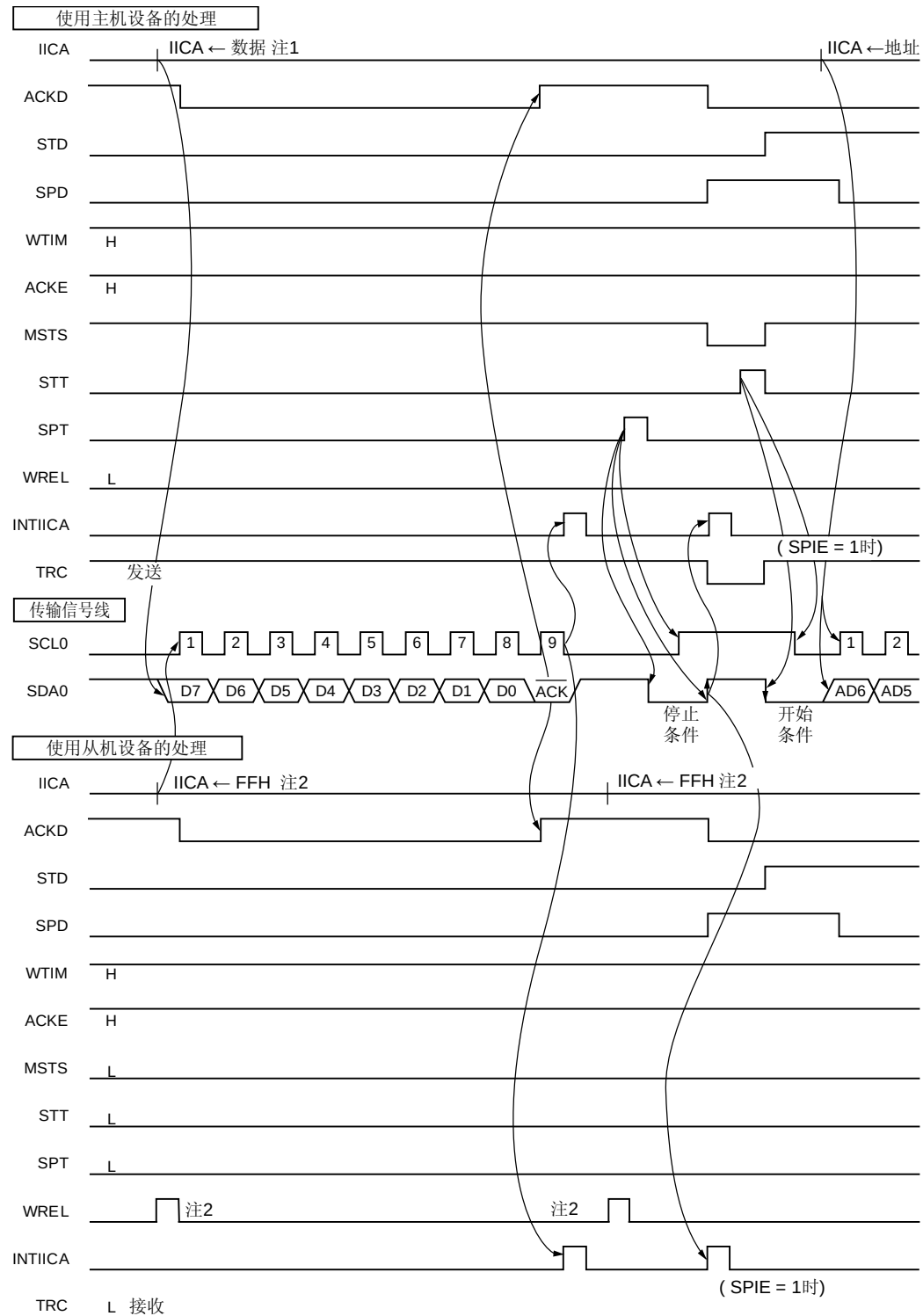
(2) 数据



- 注 1. 为了取消主机发送期间的等待状态，将数据写入 IICA，而不设置 WREL。
2. 要取消从机时的等待状态，将“FFH”写入 IICA，或者设置 WREL。

图 13-32. 主机向从机通信示例
(主机设备和从机设备都选择 9 时钟等待时) (3/3)

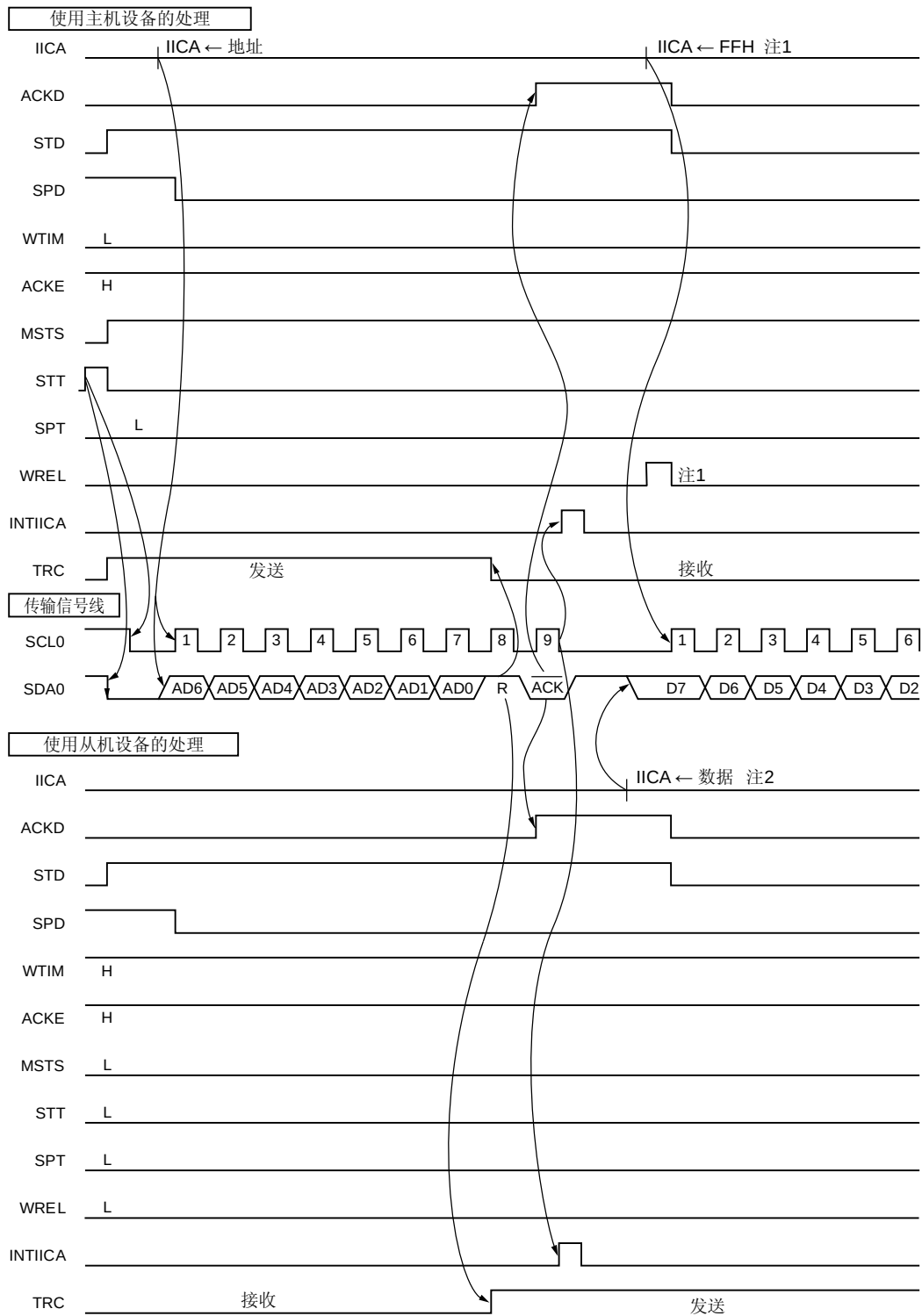
(3) 停止条件



- 注
1. 为了取消主机发送期间的等待状态，将数据写入 IICA，而不设置 WREL。
 2. 要取消从机时的等待状态，将“FFH”写入 IICA，或者设置 WREL。

图 13-33. 从机向主机通信示例
(主机选择 8 时钟等待而从机选择 9 时钟等待时) (1/3)

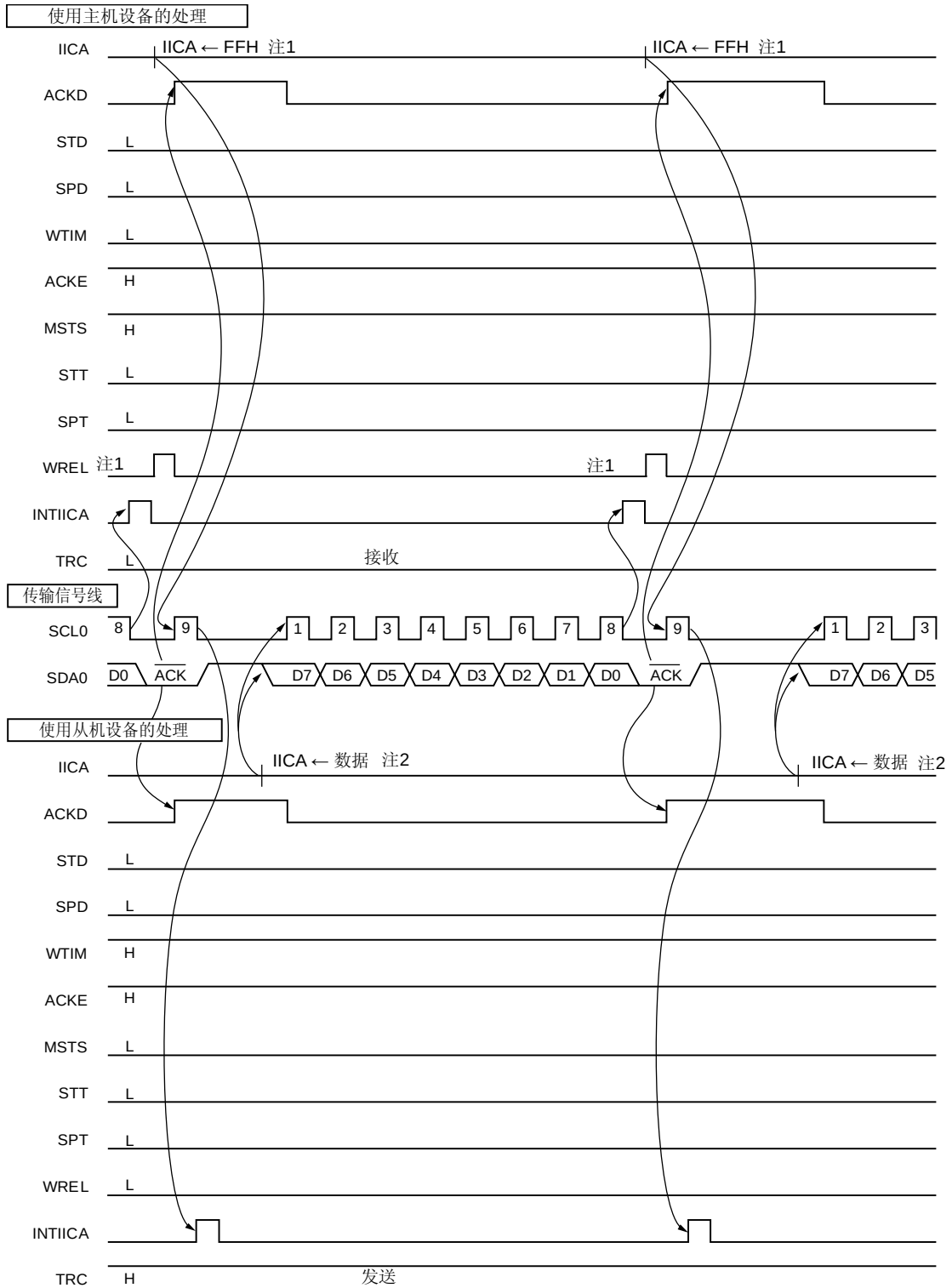
(1) 开始条件 ~ 地址



注 1. 要取消主机时的等待状态, 将"FFH"写入 IICA, 或者设置 WREL。
注 2. 为了取消从机发送期间的等待状态, 将数据写入 IICA, 而不设置 WREL。

图 13-33. 从机向主机通信示例
(主机选择 8 时钟等待而从机选择 9 时钟等待时) (2/3)

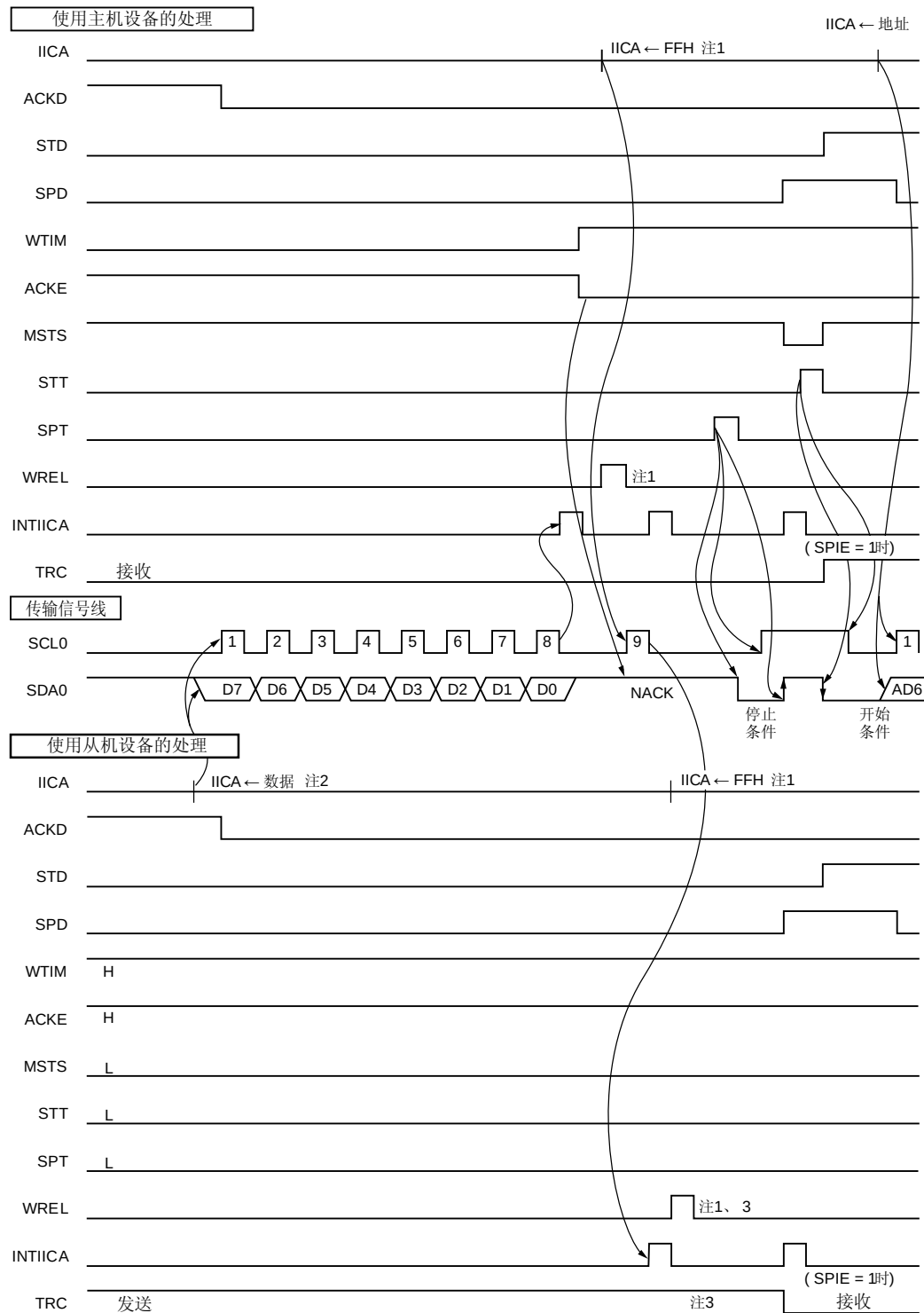
(2) 数据



- 注 1. 要取消主机时的等待状态，将“FFH”写入 IICA，或者设置 WREL。
2. 为了取消从机发送期间的等待状态，将数据写入 IICA，而不设置 WREL。

图 13-33. 从机向主机通信示例
(主机选择 8 时钟和 9 时钟等待而从机选择 9 时钟等待时) (3/3)

(3) 停止条件



- 注
1. 要取消等待状态，需将“FFH”写入 IICA，或者设置 WREL。
 2. 为了取消从机发送期间的等待状态，将数据写入 IICA，而不设置 WREL。
 3. 如果通过设置 WREL 取消从机发送期间的等待状态，则 TRC 将被清除。

14.1 乘法器/除法器的功能

乘法器/除法器具有以下功能。

- 16 位 × 16 位= 32 位(乘法)
- 32 位÷ 32 位= 32 位, 32 位余数 (除法)

14.2 乘法器/除法器的配置

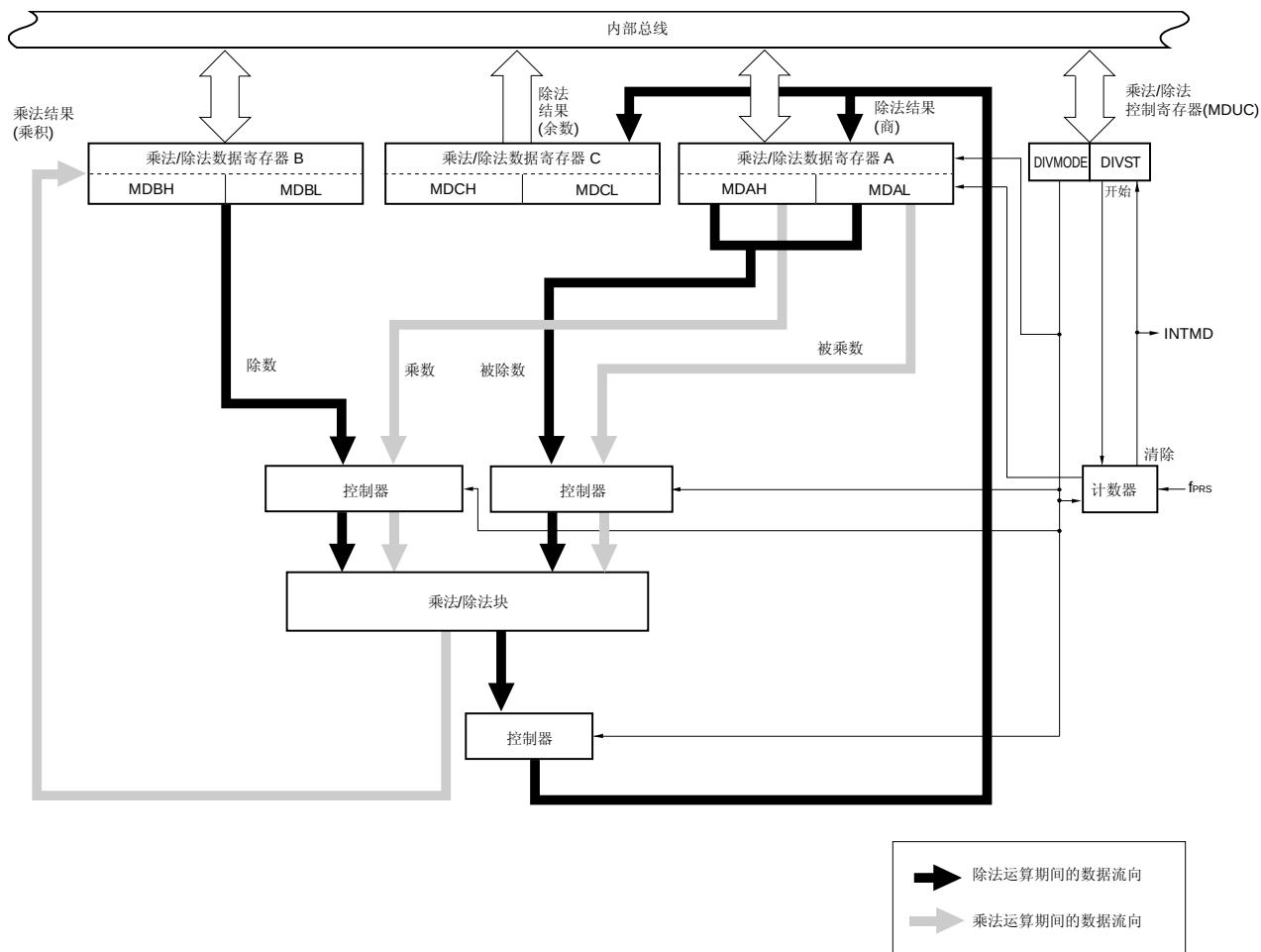
乘法器/除法器由以下硬件组成。

表 14-1. 乘法器/除法器的配置

项	配置
寄存器	乘法/除法数据寄存器 A (L) (MDAL) 乘法/除法数据寄存器 A (H) (MDAH) 乘法/除法数据寄存器 B (L) (MDBL) 乘法/除法数据寄存器 B (H) (MDBH) 乘法/除法数据寄存器 C (L) (MDCL) 乘法/除法数据寄存器 C (H) (MDCH)
控制寄存器	乘法/除法控制寄存器(MDUC)

图 14-1 显示了乘法器/除法器的框图。

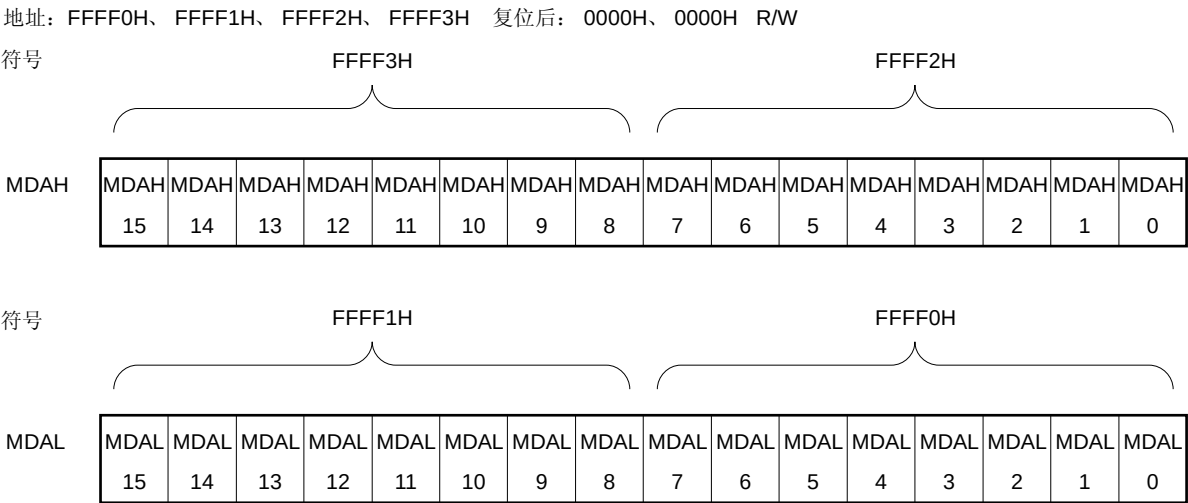
图 14-1. 乘法器/除法器的框图



(1) 乘法/除法数据寄存器 A（MDAH、MDAL）

MDAH 和 MDAL 寄存器设置用于乘法或除法操作中的数值，且存储操作结果。在乘法模式中设置乘数和被乘数，在除法模式中设置被除数。此外，在除法模式中，操作结果（商）存储在 MDAH 和 MDAL 寄存器中。
可以由 16 位的处理指令设置 MDAH 和 MDAL。
产生的复位信号可以将这些寄存器清除为 0000H。

图 14-2. 乘法/除法数据寄存器 A 的格式（MDAH、MDAL）



- 注意事项
- 1. 在除法操作进行期间，不要重写 MDAH 和 MDAL 的值（当乘法/除法控制寄存器（MDUC）为 81H 时）。在这种情况下将会执行该操作，但是操作结果将会是一个不确定值。
 - 2. 在除法操作进行（MDUC 为 81H 时）期间读取的 MDAH 和 MDAL 的值将不能得到保证。

下表给出了操作执行期间 MDAH 和 MDAL 的功能。

表 14-2. 操作执行期间 MDAH 和 MDAL 的功能

DIVMODE	操作模式	设置	操作结果
0	乘法模式	MDAH: 乘数 MDAL: 被乘数	—
1	除法模式	MDAH: 除数 (高 16 位) MDAL: 被除数 (低 16 位)	MDAH: 除法结果(商) 高 16 位 MDAL: 除法结果(商) 低 16 位

备注 DIVMODE: 乘法/除法控制寄存器（MDUC）的位 7

(2) 乘法/除法数据寄存器 B（MDBL、MDBH）

MDBH 和 MDBL 寄存器设置用于乘法或除法操作中的数值，且存储操作结果。这些寄存器存储乘法模式中的操作结果（乘积），并设置除法模式中的除数。

可以由 16 位处理指令设置 MDBH 和 MDBL。

产生的复位信号可以将这些寄存器清除为 0000H。

图 14-3. 乘法/除法数据寄存器 B 的格式（MDBH、MDBL）

地址：FFFF4H、FFFF5H、FFFF6H、FFFF7H 复位后：0000H、0000H R/W



- 注意事项
- 1. 在除法操作进行期间，不要重写 MDBH 和 MDBL 的值（当乘法/除法控制寄存器（MDUC）为 81H 时）。操作结果将会是一个不确定值。
 - 2. 在除法模式中不要将 MDBH 和 MDBL 设置为 0000H。如果设置了它们，则操作结果将会是一个不确定值。

下表给出了操作执行期间 MDBH 和 MDBL 的功能。

表 14-3. 操作执行期间 MDBH 和 MDBL 的功能

DIVMODE	操作模式	设置	操作结果
0	乘法模式	—	MDBH: 乘法结果(乘积) 高 16 位 MDBL: 乘法结果(乘积) 低 16 位
1	除法模式	MDBH: 除数(高 16 位) MDBL: 被除数(低 16 位)	—

备注 DIVMODE: 乘法/除法控制寄存器（MDUC）的位 7

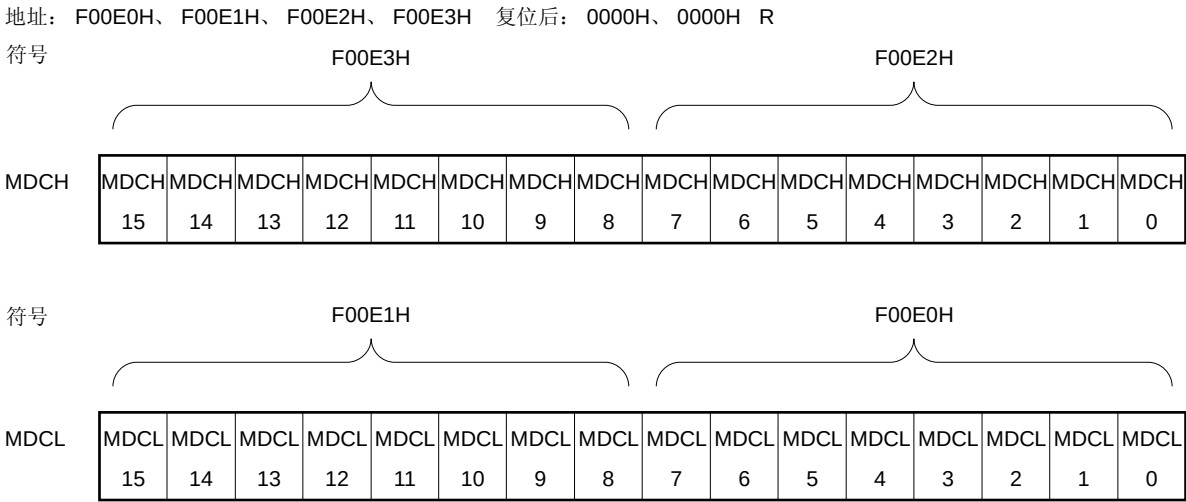
(3) 乘法/除法数据寄存器 C (MDCL、MDCH)

MDCH 和 MDCL 寄存器存储除法模式中操作结果的余数。在乘法模式中不使用它们。

MDCH 和 MDCL 可以由 16 位处理指令读取。

产生的复位信号可以将这些寄存器清除为 0000H。

图 14-4. 乘法/除法数据寄存器 C 的格式 (MDCH、MDCL)



注意事项 在除法操作进行（乘法/除法控制寄存器（MDUC）为 81H 时）期间读取的 MDCH 和 MDCL 的值将不能得到保证。

表 14-4. 操作执行期间 MDCH 和 MDCL 的功能

DIVMODE	操作模式	设置	操作结果
0	乘法模式	—	—
1	除法模式	—	MDCH: 余数(高 16 位) MDCL: 余数(低 16 位)

备注 DIVMODE: 乘法/除法控制寄存器（MDUC）的位 7

执行乘法和执行除法时寄存器配置的区别如下。

- 乘法期间的寄存器配置

$$\text{MDAL (位 15 至 0)} \times \text{MDAH (位 15 至 0)} = [\text{MDBH (位 15 至 0)、MDBL (位 15 至 0)}]$$
- 除法期间的寄存器配置

$$[\text{MDAH (位 15 至 0)、MDAL (位 15 至 0)}] \div [\text{MDBH (位 15 至 0)、MDBL (位 15 至 0)}] =$$

<商>
<余数>

$$[\text{MDAH (位 15 至 0)、MDAL (位 15 至 0)}] \dots [\text{MDCH (位 15 至 0)、MDCL (位 15 至 0)}]$$

14.3 控制乘法器/除法器的寄存器

使用乘法/除法控制寄存器（MDUC）控制乘法器/除法器。

(1) 乘法/除法控制寄存器（MDUC）

MDUC 是一款 8 位寄存器，控制乘法器/除法器的操作。
可以由 1 位或 8 位存储器处理指令设置 MDUC。
产生的复位信号将该寄存器清除为 00H。

图 14-5. 乘法/除法控制寄存器的格式（MDUC）

地址： F00E8H 复位后： 00H R/W

符号	<7>	6	5	4	3	2	1	<0>
MDUC	DIVMODE	0	0	0	0	0	0	DIVST

DIVMODE	操作模式(乘法/除法)选择
0	乘法模式
1	除法模式

DIVST ^注	开始/停止除法操作
0	完成除法操作处理
1	开始除法操作/进行除法操作处理

注 仅能在除法模式中设置 DIVST（1）。在除法模式中，通过设置(1) DIVST 开始除法操作。操作结束时 DIVST 自动清除（0）。在乘法模式中，通过将乘数和被乘数分别设置给 MDAH 和 MDAL 自动开始操作。

- 注意事项
- 1. 处理操作（DIVST 为 1 时）期间不能重写 DIVMODE。如果对其进行重写，则操作结果将会是一个不确定值。
 - 2. 在除法操作处理（DIVST 为 1 时）期间使用软件不能清除（0）DIVST。

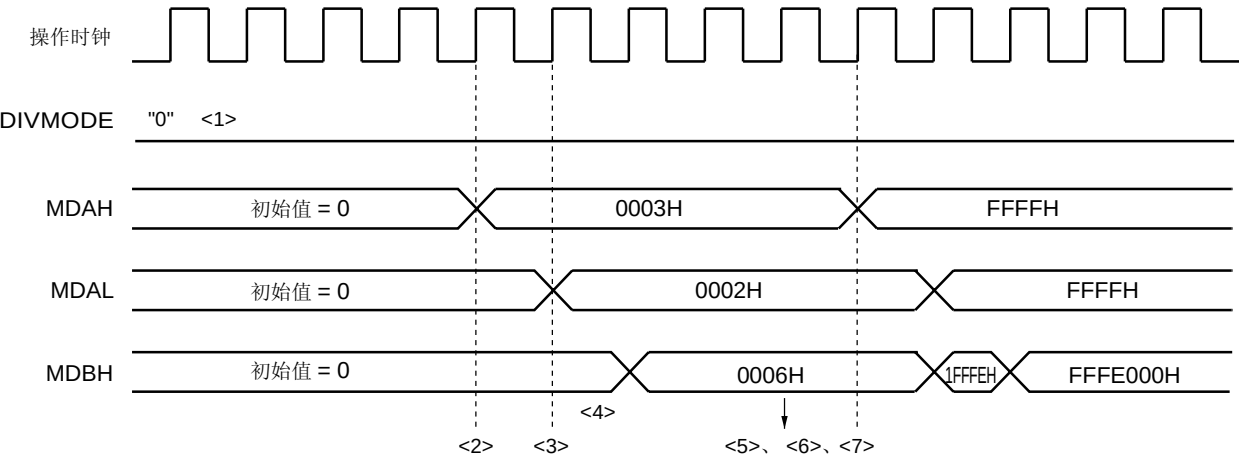
14.4 乘法器/除法器的操作

14.4.1 乘法操作

- 初始设置
 - <1> 将乘法/除法控制寄存器（MDUC）的位 7（DIVMODE）置为 0。
 - <2> 将被乘数设置给乘法/除法数据寄存器 A（L）（MDAL）。
 - <3> 将乘数设置给乘法/除法数据寄存器 A（H）（MDAH）。
 - （步骤<2>和<3>的执行次序不分先后。将乘数和被乘数分别设置给 MDAH 和 MDAL 时，自动开始乘法操作。）
- 操作处理期间
 - <4> 等待至少一个时钟。产生一个时钟时操作将结束。
- 操作结束
 - <5> 从乘法/除法数据寄存器 B（L）（MDBL）中读取乘积（低 16 位）。
 - <6> 从乘法/除法数据寄存器 B（H）（MDBH）中读取乘积（高 16 位）。
 - （步骤<5>和<6>的执行次序不分先后。）
- 下一次操作
 - <7> 要执行下一次乘法操作，需从乘法操作“初始设置”开始。
 - <8> 要执行下一次除法操作，需从 14.4.2 除法操作中的“初始设置”开始。

备注 步骤<1>至<7>对应着图 14-6 中的<1>至<7>。

图 14-6. 乘法操作（0003H × 0002H）的时序图

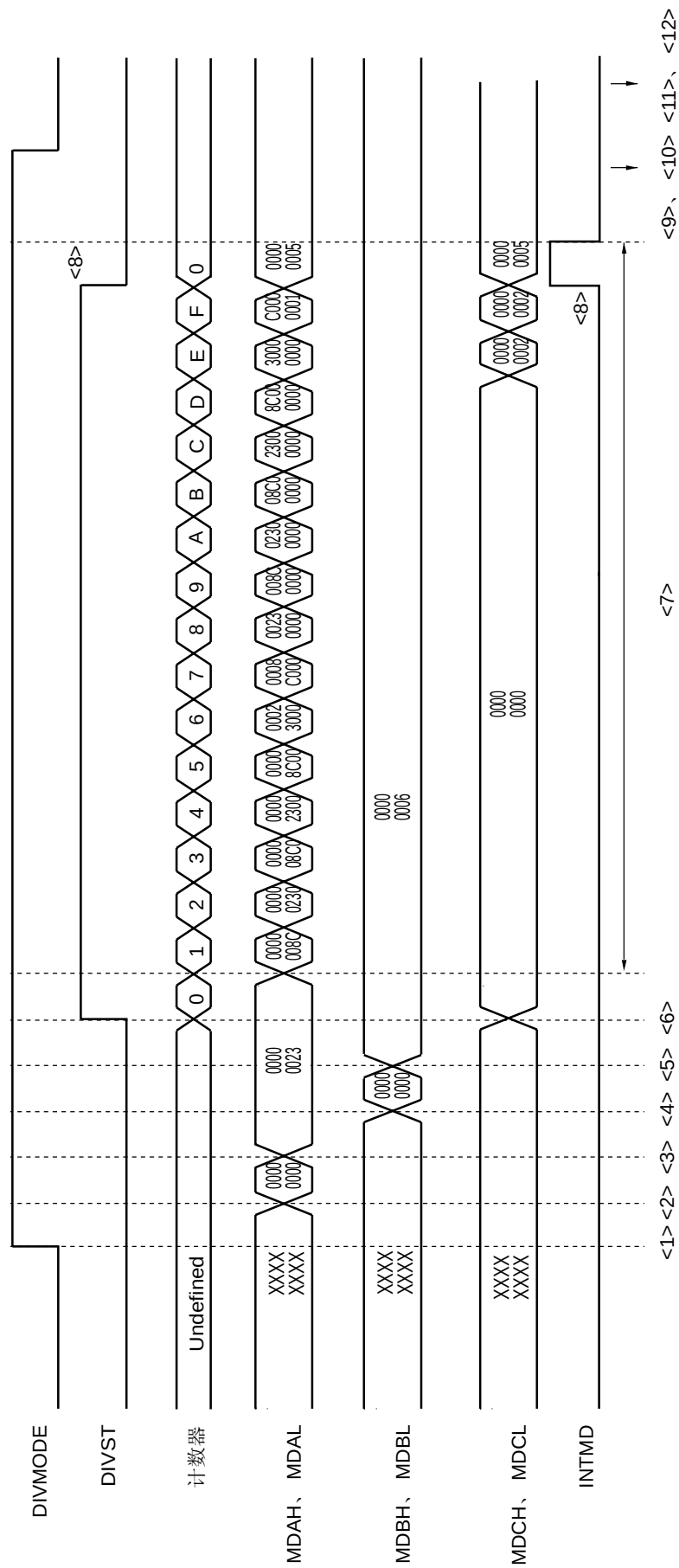


14.4.2 除法操作

- 初始设置
 - <1> 将乘法/除法控制寄存器 (MDUC) 的位 7 (DIVMODE) 置为 1。
 - <2> 将被除数 (高 16 位) 设置给乘法/除法数据寄存器 A (H) (MDAH)。
 - <3> 将被除数 (低 16 位) 设置给乘法/除法数据寄存器 A (L) (MDAL)。
 - <4> 将除数 (高 16 位) 设置给乘法/除法数据寄存器 B (H) (MDBH)。
 - <5> 将除数 (低 16 位) 设置给乘法/除法数据寄存器 B (L) (MDBL)。
 - <6> 将 MDUC 的位 0 (DIVST) 置为 1。
(步骤<2>至<5>的执行次序不分先后。)
- 操作处理期间
 - <7> 当下列处理中的任一个完成时, 操作将会结束。
 - 至少 16 个时钟的等待 (当产生 16 个时钟时操作会结束。)
 - DIVST 是否已被清除的检验
 - 除法完成中断 (INTMD) 的产生
(操作处理期间, 读取的 MDBL、MDBH、MDCH 和 MDCL 的值不能保证。)
- 操作结束
 - <8> 将 DIVST 清除 (0), 并产生中断请求信号 (INTMD) (操作结束)。
 - <9> 从 MDAL 中读取商值 (低 16 位)。
 - <10> 从 MDAH 中读取商值 (高 16)。
 - <11> 从乘法/除法数据寄存器 C (L) (MDCL) 中读取余数 (低 16 位)。
 - <12> 从乘法/除法数据寄存器 C (H) (MDCH) 中读取余数 (高 16 位)。
(步骤<9>至<12>的执行次序不分先后。)
- 下一次操作
 - <13> 要执行下一次乘法操作, 需从 14.4.1 乘法操作中的“初始设置”开始。
 - <14> 要执行下一次除法操作, 需从除法操作的“初始设置”开始。

备注 步骤<1>至<12>对应着图 14-7 中的<1>至<12>。

图 14-7. 除法操作（示例：35 ÷ 6 = 5、余数 5）的时序图



第十五章 DMA 控制器

78K0R/Kx3-L 配有一个内部 DMA（直接存储器访问）控制器。

数据可不通过 CPU 在支持 DMA、SFR 的外围硬件和内部 RAM 之间自动传输。

结果，可以用 SFR 和内部 RAM 之间的传输并行执行 CPU 的正常内部操作和数据传输，因此，可以处理很大容量的数据。此外，还可以实现使用通信的实时控制、定时器和 A/D。

15.1 DMA 控制器的功能

- DMA 通道的数目：2
- 传输单位：8 或 16 位
- 最大传输单位：1024 次
- 传输类型：2-周期传输（在 2 个时钟内处理一个传输且在处理期间 CPU 不工作。）
- 传输模式：单次传输模式
- 传输请求：可从下列外围硬件中断中选择
 - A/D 转换器
 - 串行接口（CSI00，CSI01，CSI10，UART0，UART1 或 IIC10）
 - 定时器（通道 0，1，4 或 5）
- 传输目标：SFR 和内部 RAM 之间

以下是使用 DMA 功能的实例。

- 串行接口的连续传输
- 模拟数据的批量传输
- 在固定间隔处捕获 A/D 转换结果
- 在固定间隔处捕获端口值

15.2 DMA 控制器的配置

DMA 控制器包含以下硬件：

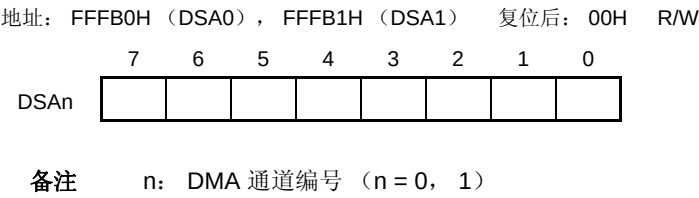
表 15-1. DMA 控制器的配置

项目	配置
地址寄存器	• DMA SFR 地址寄存器 0, 1 (DSA0, DSA1) • DMA RAM 地址寄存器 0, 1 (DRA0, DRA1)
计数寄存器	• DMA 字节计数寄存器 0, 1 (DBC0, DBC1)
控制寄存器	• DMA 模式控制寄存器 0, 1 (DMC0, DMC1) • DMA 操作控制寄存器 0, 1 (DRC0, DRC1)

(1) DMA SFR 地址寄存器 n (DSAn)

该寄存器是 8 位寄存器，用于设置 SFR 地址，此地址即为 DMA 通道 n 的传输源或目的地。
设置 SFR 地址的低 8 位为 FFF00H 至 FFFFFH。
该寄存器不能自动增加而是固定为一个指定值。
在 16 位传输模式中，最低有效位被忽略并按偶地址处理。
DSAn 可以按 8 位进行读写。但在 DMA 传输期间不能被写入。
产生的复位信号将该寄存器清除为 00H。

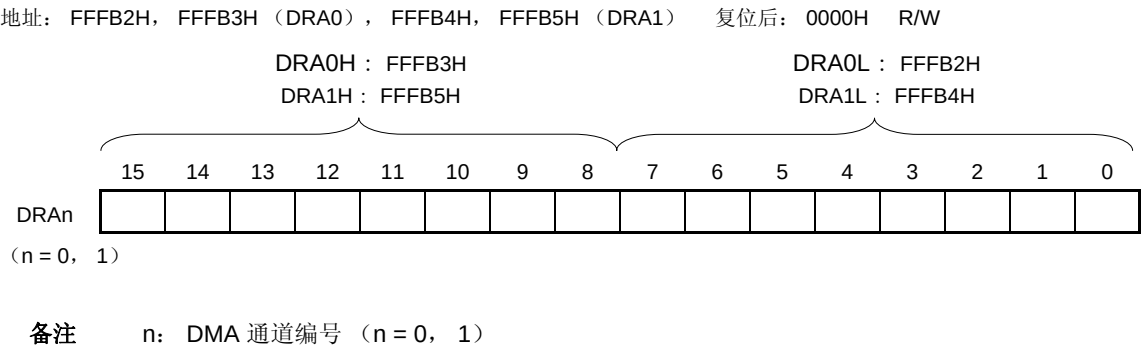
图 15-1. DMA SFR 地址寄存器 n (DSAn) 的格式



(2) DMA RAM 地址寄存器 n (DRAn)

该寄存器是 16 位寄存器，用于设置 RAM 地址，此地址即为 DMA 通道 n 的传输源或目的地。
内部 RAM 区除通用寄存器（以 μ PD78F1001， 78F1004 和 78F1007 为例地址为 FF900H 至 FFEDFH）的地址均可设置给该寄存器。
设置 RAM 地址的低 16 位。
当启动 DMA 传输后该寄存器可自动增加。在 8 位传输模式中本寄存器增加+1，在 16 位传输模式下增加+2。
DMA 传输从设置给 DRAn 寄存器的地址开始。当最末地址的数据已经被传输，在 8 位传输模式下 DRAn 以最末地址+1 的数值停止，而在 16 位传输模式下 DRAn 以最末地址+2 的数值停止。
在 16 位传输模式中，最低有效位被忽略并按偶地址处理。
DRAn 能以 8 位或 16 位读写。但在 DMA 传输期间不能被写入。
产生的复位信号将该寄存器清除为 0000H。

图 15-2. DMA RAM 地址寄存器 n (DRAn) 的格式



(3) DMA 字节计数寄存器 n (DBCn)

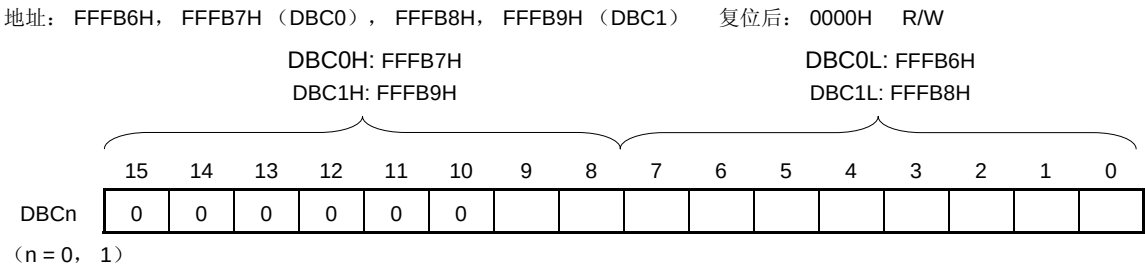
该寄存器是 10 位寄存器，用于设置 DMA 通道 n 执行传输次数的数目。在执行 DMA 传输前确保给 DBCn 寄存器设置传输次数的数目。

每执行一次 DMA 传输，该寄存器自动减少计数。在 DMA 传输期间通过读取 DBCn 寄存器，可得出剩余传输次数的数目。

DBCn 能以 8 位或 16 位读写。但在 DMA 传输期间不能被写入。

产生的复位信号将该寄存器清除为 0000H。

图 15-3. DMA 字节计数寄存器 n (DBCn) 的格式



DBCn[9:0]	传输次数的数目 (当写入 DBCn 时)	传输次数的剩余数目 (当读取 DBCn 时)
000H	1024	传输完成或等待 DMA 传输 1024 次
001H	1	等待 DMA 剩余一次的传输
002H	2	等待 DMA 剩余两次的传输
003H	3	等待 DMA 剩余三次的传输
⋮	⋮	⋮
3FEH	1022	等待 DMA 剩余 1022 次的传输
3FFH	1023	等待 DMA 剩余 1023 次的传输

- 注意事项
1.

务必将位 15 至位 10 清为“0”。
2.

如果通用寄存器已被指定或已超出内部 RAM 的空间，若继续传输，则通用寄存器或 SFR 空间会被写入或读取，导致这些空间的数据丢失。务必设置传输次数的数目，而该数目不在内部 RAM 空间内。

备注

n: DMA 通道编号 (n = 0, 1)

15.3 控制 DMA 控制器的寄存器

DMA 控制器由以下寄存器控制:

- DMA 模式控制寄存器 n (DMCn)
- DMA 操作控制寄存器 n (DRCn)

备注 n: DMA 通道编号 (n = 0, 1)

(1) DMA 模式控制寄存器 n (DMCn)

DMCn 是用于设置 DMA 通道 n 的传输模式的寄存器。其用于选择传输方向，数据容量，挂起设置以及启动源。位 7 (STGn) 启动 DMA 的软件触发。

操作期间 (DSTn = 1 时) 禁止重写 DMCn 的位 6, 5 以及 3 至 0。

可通过 1 位或 8 位的存储操作指令设置 DMCn。

产生的复位信号将该寄存器清除为 00H。

图 15-4. DMA 模式控制寄存器 n (DMCn) 的格式 (1/2)

地址: FFFBAH (DMC0), FFFBBH (DMC1) 复位后: 00H R/W

符号

	<7>	<6>	<5>	<4>	3	2	1	0
DMCn	STGn	DRSn	DSn	DWAITn	IFCn3	IFCn2	IFCn1	IFCn0

STGn [※]	DMA传输启动软件触发
0	没有触发操作
1	当DMA操作使能（DENn = 1）时启动DMA传输。

当DMA操作使能（DENn = 1）时通过向STGn写入1启动DMA传输。

读取该位时，恒为00H。

DRSn	DMA传输方向的选择
0	SFR至内部RAM
1	内部 RAM至SFR

DSn	DMA传输数据容量的说明
0	8 位
1	16 位

DWAITn	DMA传输的挂起
0	基于DMA启动请求执行DMA传输（未保持挂起）。
1	如果有则保持 DMA启动请求挂起。

通过将DWAITn的值清除为0，保持挂起的DMA传输被启动。

当DWAITn设置为1时，实际保持DMA传输挂起的时间为2个时钟。

注 不论 IFCn0 至 IFCn3 的值为多少，都可使用软件触发 (STGn)。

备注 n: DMA 通道编号 (n = 0, 1)

图 15-4. DMA 模式控制寄存器 n（DMCn）的格式（2/2）

地址： FFFBAH（DMC0）， FFFBBH（DMC1） 复位后： 00H R/W

符号	<7>		<6>		<5>		<4>		3	2	1	0
DMCn	STGn		DRSn		DSn		DWAITn		IFCn3	IFCn2	IFCn1	IFCn0

IFCn 3	IFCn 2	IFCn 1	IFCn 0	DMA 启动源*的选择	
				触发信号	触发内容
0	0	0	0	—	禁止用中断触发 DMA 传输。 (只允许软件触发。)
0	0	1	0	INTTM00	定时器通道 0 中断
0	0	1	1	INTTM01	定时器通道 1 中断
0	1	0	0	INTTM04	定时器通道 4 中断
0	1	0	1	INTTM05	定时器通道 5 中断
0	1	1	0	INTST0/INTCSI00	UART0 发送结束中断 /CSI00 传输结束中 断
0	1	1	1	INTSR0/INTCSI01	UART0 接收结束中断 /CSI01 传输结束中 断
1	0	0	0	INTST1/INTCSI10/INTIIC10	UART1 发送结束中断/ CSI10 传输结束中断/ IIC10 传输结束中断
1	0	0	1	INTSR1	UART1 接收结束中断
1	1	0	0	INTAD	A/D 转换结束中断
其它				禁止设置	

注 不论 IFCn0 至 IFCn3 的值为多少，都可使用软件触发（STGn）。

备注 n: DMA 通道编号（n = 0, 1）

(2) DMA 操作控制寄存器 n (DRCn)

DRCn 是用于允许或禁止 DMA 通道 n 传输的寄存器。
操作期间 (DSTn = 1 时) 禁止重写该寄存器的位 7 (DENn)。
可通过 1 位或 8 位的存储操作指令设置 DRCn。
产生的复位信号将该寄存器清除为 00H。

图 15-5. DMA 操作控制寄存器 n (DRCn) 的格式

地址: FFFBCH (DRC0), FFFBDH (DRC1) 复位后: 00H R/W

符号	<7>	6	5	4	3	2	1	<0>
DRCn	DENn	0	0	0	0	0	0	DSTn

DENn	DMA 操作使能标志
0	禁止DMA通道n操作（停止 DMA的操作时钟）。
1	允许DMA通道n操作。
在DMA操作使能（DENn = 1）后当DSTn = 1时DMAC等待DMA触发。	

DSTn	DMA传输模式标志
0	完成DMA通道n的DMA传输。
1	未完成DMA通道n的DMA传输（仍在执行）。
在DMA操作使能（DENn = 1）后当DSTn = 1时DMAC等待DMA触发。 当输入由IFCn3至 IFCn0设置的软件触发（STGn）或启动源触发时，开始DMA传输。 此后当DMA传输完成时，该位自动清除为0。 在执行中若向该位写入0则强制终止DMA传输。	

- 注意事项
1. 当 DMA 传输完成时，DSTn 标志自动清除为 0。仅当 DSTn = 0 时才允许写入 DENn 标志。因此当在没有等到 DMA 的中断 (INTDMA_n) 产生而 DMA 传输终止时，先设置 DSTn 为 0 然后设置 DENn 为 0（详情参见 15.5.5 用软件强制终止）。
 2. 当 OSMC 寄存器的 FSEL 已经被置 1 时，在设置后至少三个时钟不允许 (DENn = 1) DMA 操作。

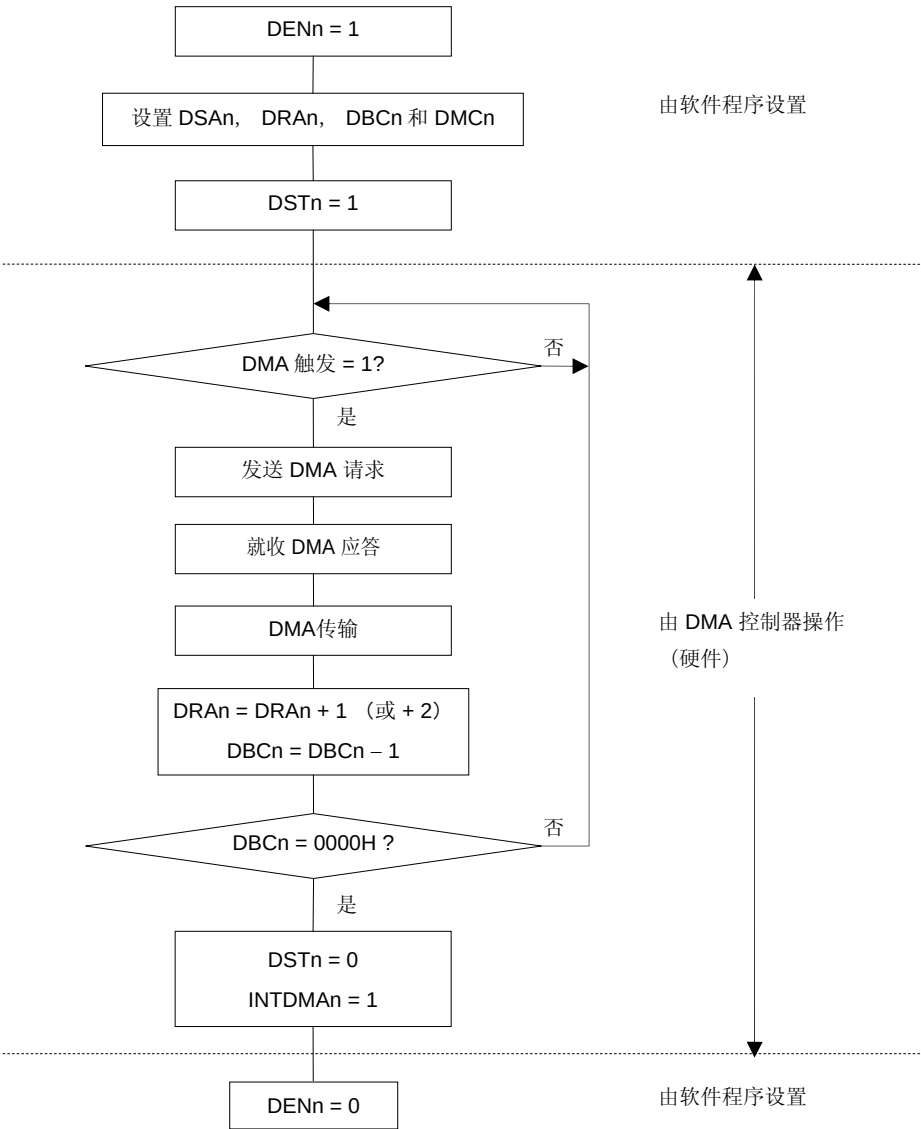
备注 n: DMA 通道编号 (n = 0, 1)

15.4 DMA 控制器的操作

15.4.1 操作步骤

- <1> 当 DENn = 1 时，使能 DMA 控制器操作。写入其它寄存器前，确保将 DENn 置 1。用 8 位操作指令写入 80H。
- <2> 分别将 SFR 地址、RAM 地址、传输次数的数目以及 DMA 传输的模式设置到 DSA_n、DRA_n、CBC_n 以及 DMC_n 寄存器。
- <3> 当 DST_n = 1 时，DMA 控制器等待 DMA 触发。用 8 位操作指令写入 81H。
- <4> 当输入由 IFC_n3 至 IFC_n0 指定的软件触发（STG_n）或启动源触发时，开始 DMA 传输。
- <5> 当 DBC_n 寄存器设置的传输次数的数目达到 0 时，传输完成，由于产生中断（INTDMA_n），传输自动终止。
- <6> 当不使用 DMA 控制器时，通过清除 DEN_n 为 0 以停止 DMA 控制器的操作。

图 15-6. 操作步骤



备注 n: DMA 通道编号 (n = 0, 1)

15.4.2 传输模式

使用 DMCn 寄存器的位 6 和位 5 (DRSn 和 DS_n)，可为 DMA 传输选择以下四种模式之一。

DRSn	DS _n	DMA 传输模式
0	0	从1字节数据的SFR（固定地址）传输到 RAM（地址+1）
0	1	从2字节数据的SFR（固定地址）传输到 RAM（地址 +2）
1	0	从1字节数据的RAM（地址 +1）传输到SFR（固定地址）
1	1	从2字节数据的RAM（地址 +2）传输到SFR（固定地址）

通过使用这些传输模式，用串行接口最多可连续传输 1024 字节数据，A/D 转换结果的数据可被连续传输，而端口数据可用定时器以固定时间间隔进行扫描。

15.4.3 DMA 传输终止

当 DBCn = 00H 和 DMA 传输完成时，DSTn 位自动清除为 0。产生一个中断请求 (INTDMAn) 从而终止传输。

当 DSTn 位被清除为 0 以强制终止 DMA 传输时，DBCn 和 DRAn 寄存器保持传输被终止时的值。

如果传输被强制终止则不产生中断请求 (INTDMAn)。

备注 n: DMA 通道编号 (n = 0, 1)

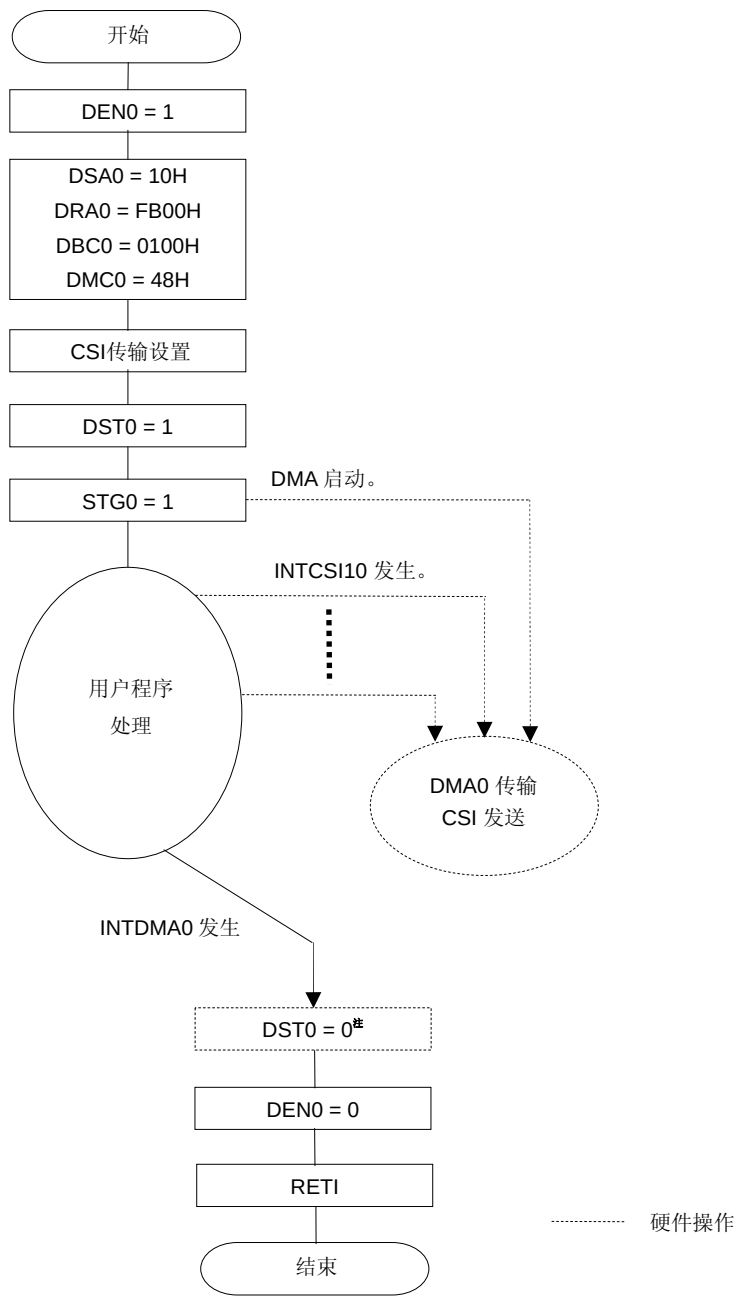
15.5 DMA 控制器设置举例

15.5.1 CSI 连续发送

一个 CSI 连续发送的设置举例流程图如下所示。

- CSI10 连续发送
- DMA 通道 0 用于 DMA 传输。
- DMA 启动源：INTCSI10 （软件触发（STG0）仅用于第一次启动源）
- 由 IFC03 至 IFC00 （DMC0 寄存器的位 3 至 0）= 1000B 指定 CSI10 的中断。
- 将 RAM 的 FFB00H 至 FFBFFH （256 个字节）传输至 CSI 发送缓存（SIO10）的 FFF10H。

图 15-7. CSI 连续发送的设置举例



注 当 DMA 传输完成时，DST0 标志自动清除为 0。
仅当 DST0 = 0 时才允许写入 DEN0 标志。若要不等到 DMA0 的中断（INTDMA0）发生而终止 DMA 传输，
则需设置 DST0 为 0 然后再设置 DEN0 为 0。（详情参见 15.5.5 用软件强制终止）

连续发送的第一个触发不能用 CSI 的中断启动。需用软件触发。

第二次及以后的 CSI 发送自动执行。

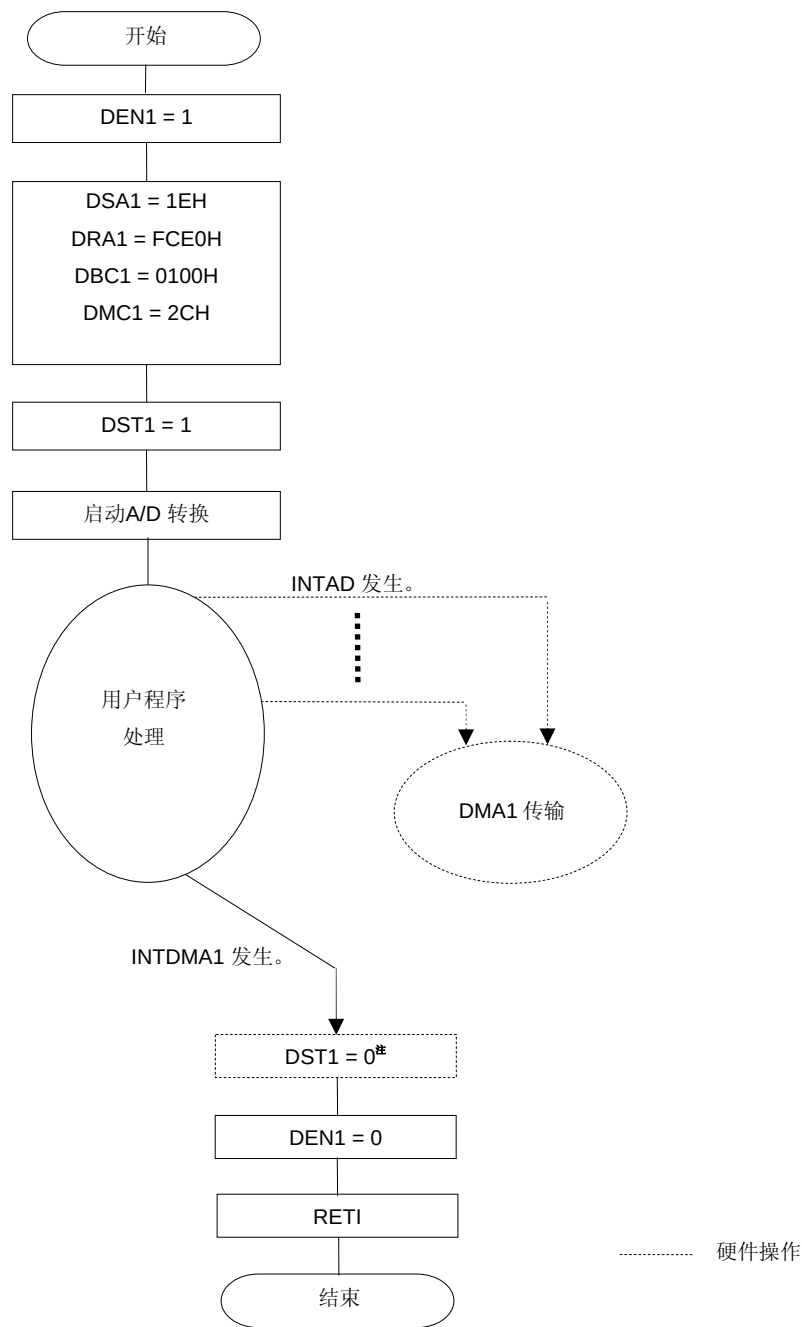
一旦最末数据被写入发送缓存就会产生 DMA 中断（INTDMA0）。这时，正在发送 CSI 的最末数据。因此若要再次启动 DMA 传输，需等待 CSI 传输完成。

15.5.2 连续捕获 A/D 转换结果

一个连续捕获 A/D 转换结果的设置举例流程图如下所示。

- 连续捕获 A/D 转换结果。
- DMA 通道 1 用于 DMA 传输。
- DMA 启动源：INTAD
- 由 IFC13 至 IFC10（DMC1 寄存器的位 3 至 0）= 1100B 指定 A/D 的中断。
- 将 10 位 A/D 转换结果寄存器的 FFF1EH 和 FFF1FH（2 个字节）传输至 RAM 中 FFCE0H 至 FFEDFH 的 512 字节。

图 15-8. 连续捕获 A/D 转换结果的设置举例



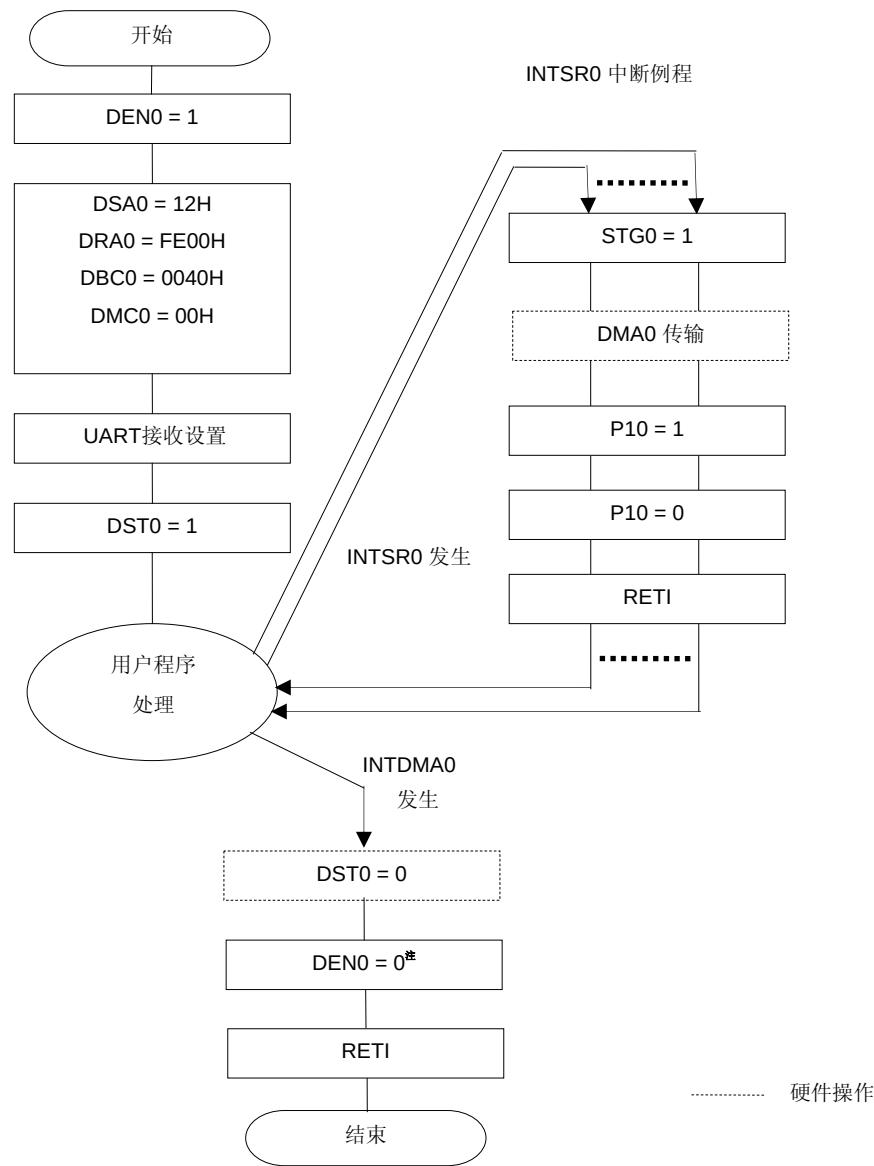
注 当 DMA 传输完成时，DST1 标志自动清除为 0。
仅当 DST1 = 0 时才允许写入 DEN1 标志。若要不等到 DMA1 的中断（INTDMA1）发生而终止 DMA 传输，
则需设置 DST1 为 0 然后再设置 DEN1 为 0。（详情参见 15.5.5 用软件强制终止）

15.5.3 UART 连续接收+ ACK 发送

一个 UART 连续接收+ ACK 发送的设置举例流程图如下所示。

- 从 UART0 连续接收数据并在接收完成时将 ACK 输出至 P10。
- DMA 通道 0 用于 DMA 传输。
- DMA 启动源：软件触发（中断发生时禁止 DMA 传输）。
- 将 UART 接收数据寄存器 0（RXD0）的 FFF12H 传输至 RAM 中 FFE00H 至 FFE3FH 的 64 字节。

图 15-9. UART 连续接收+ ACK 发送的设置举例



注 当 DMA 传输完成时，DST0 标志自动清除为 0。
仅当 DST0 = 0 时才允许写入 DEN0 标志。若要不等到 DMA0 的中断（INTDMA0）发生而终止 DMA 传输，
则需设置 DST0 为 0 然后再设置 DEN0 为 0。（详情参见 15.5.5 用软件强制终止）

备注 这是一个用软件左 DMA 启动源的举例。
如果不发送 ACK 且仅是从 UART 连续接收数据，则 UART 接收结束中断（INTSR0）可用作数据接收的
DMA 启动源。

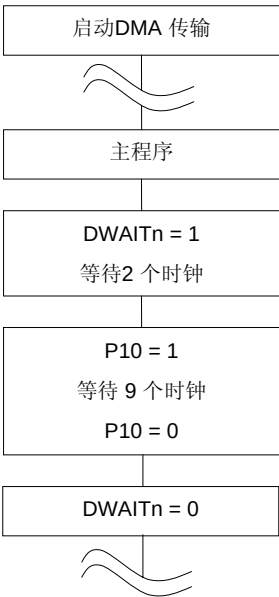
15.5.4 由 DWAITn 保持 DMA 传输挂起

当 DMA 传输启动时，在指令被执行后进行传输。此时，CPU 的操作被停止并延迟 2 个时钟，若这将导致设置系统的操作出现问题，则可通过设置 DWAITn 为 1 使 DMA 传输被挂起。

例如，若要从 P10 引脚输出 10 个操作频率时钟宽度的脉冲，若在中途启动 DMA 传输，则时钟宽度增值 12 个时钟。此时，可设置 DWAITn 为 1 使 DMA 传输挂起。

在设置 DWAITn 为 1 后，直到 DMA 传输被挂起需要 2 个时钟。

图 15-10. 由 DWAITn 保持 DMA 传输挂起的设置举例



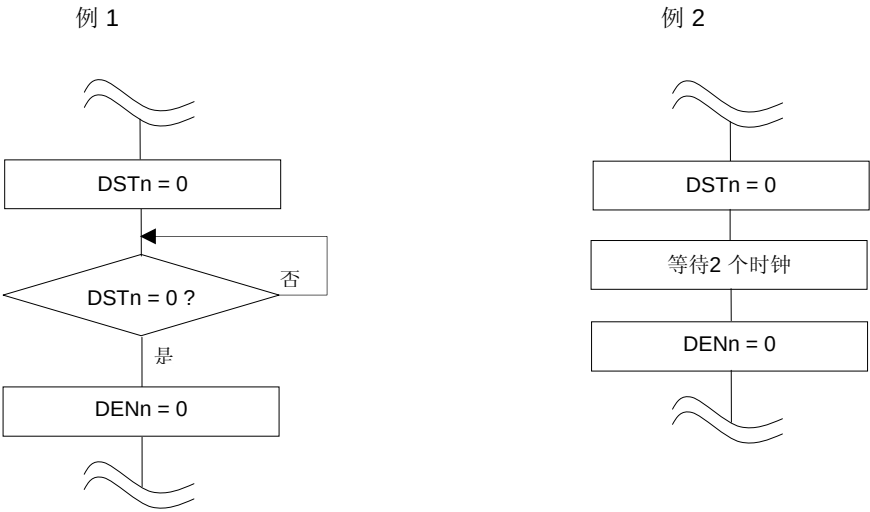
- 备注
- 1. n: DMA 通道编号 (n = 0, 1)
 - 2. 1 个时钟: $1/f_{CLK}$ (f_{CLK} : CPU 时钟)

15.5.5 用软件强制终止

用软件将 DSTn 设置为 0 后，直到 DMA 传输实际停止且 DSTn 设置为 0 需要 2 个时钟的时间。因此，若要不等到 DMA 的中断（INTDMA_n）发生而用软件强制终止 DMA 传输，需要执行以下处理中的一种。

- 用软件将 DSTn 设置为 0（用 DRCn = 80H 按 8 位操作指令写入），通过轮询确认 DSTn 已经被清除为 0，然后将 DENn 设置为 0（用 DRCn = 00H 按 8 位操作指令写入）。
- 用软件将 DSTn 设置为 0（用 DRCn = 80H 按 8 位操作指令写入），两个或更多个时钟后，将 DENn 设置为 0（用 DRCn = 00H 按 8 位操作指令写入）。

图 15-11. DMA 传输的强制终止



- 备注
1. n: DMA 通道编号 (n = 0, 1)
 2. 1 个时钟: $1/f_{CLK}$ (f_{CLK} : CPU 时钟)

15.6 使用 DMA 控制器的注意事项

(1) DMA 的优先级

DMA传输期间，其它DMA通道产生的请求被挂起。完成正在进行的DMA传输后，开始进行挂起的DMA传输。
当在短期间^注内，从一个DMA通道中依次产生多个请求，则其将被依次传输，完成后，再执行其它DMA通道的请求。这种情况下，在第一个DMA传输和下一个DMA传输之间可执行一个或两个指令。
但是，如果同时传输两个 DMA 请求，则 DMA 通道 0 比通道 1 优先。
如果同时产生 DMA 请求和中断请求，则 DMA 传输优先，然后在执行中断服务。

注 短期间指 8 个或更短的 CPU 时钟期间。时钟期间的长度和 DMA 操作的关系如下所示。

- 1 个时钟期间： 不接收禁止设置的 DMA 请求。
- 2 至 4 个时钟期间： 执行依次产生请求的通道的 DMA 传输。
- 5 至 8 个时钟期间： 执行依次产生请求的通道的 DMA 传输还是执行其它通道的 DMA 请求，取决于所执行的 CPU 指令次数的数目。

(2) DMA 响应时间

DMA 传输的响应时间如下：

表 15-2. DMA 传输的响应时间

	最短时间	最长时间
响应时间	2 个时钟 ^注	10 个时钟 ^注

注 不包括 DMA 传输的两个时钟

备注 1 个时钟： 1/f_{CLK}（f_{CLK}： CPU 时钟）

但在以下情况中，DMA 传输可能被延迟更长时间。DMA 传输延迟时钟的数目根据条件变化。

- 由 RAM 进行指令执行
- DMA 挂起指令的执行

<R>

(3) 待机模式中的操作

待机模式中 DMA 控制器操作如下：

表 15-3. 待机模式中 DMA 的操作

状态	DMA 操作
HALT 模式	正常操作
STOP 模式	停止操作。 如果DMA传输和执行STOP指令冲突，则有可能破坏DMA传输。因此在执行STOP指令前停止DMA传输。

(4) DMA 挂起指令

即使产生 DMA 请求，下述指令执行后 DMA 传输也会被立即挂起。

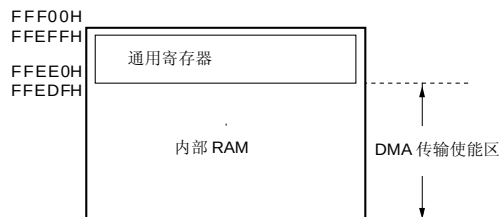
- CALL !addr16
- CALL &!addr16
- CALL !!addr20
- CALL rp
- CALLT [addr5]
- BRK
- IF0L, IF0H, IF1L, IF1H, IF2L, IF2H, MK0L, MK0H, MK1L, MK1H, MK2L, MK2H, PR00L, PR00H, PR01L, PR01H, PR02L, PR02H, PR10L, PR10H, PR11L, PR11H, PR12L, PR12H 和 PSW 寄存器各自的位操作指令，以及包括 ES 寄存器的带操作数的 8 位操作指令

(5) 指定通用寄存器区的地址或除内部 RAM 区外的地址情况下的操作

由 DRA0n 指定的地址在 DMA 传输期间递增。如果地址增加到通用寄存器区的地址或超出内部 RAM 区的地址，则需执行以下操作。

- 在从 SFR 至 RAM 的传输模式中
该地址的数据被丢失。
- 在从 RAM 至 SFR 的传输模式中
不明确数据被传输到 SFR。

在任何一种情况下，都有可能产生故障或对系统有所破坏。因此，确保地址在内部 RAM 区而不是在通用寄存器区。



第十六章 中断功能

根据产品的不同，中断源的数量不同。

		78K0R/KC3-L (44-引脚)	78K0R/KC3-L (48-引脚)	78K0R/KD3-L	78K0R/KE3-L
可屏蔽 中断	内部	9	9	9	9
	外部	24	25	25	25

16.1 中断功能类型

使用以下两种类型的中断。

(1) 可屏蔽中断

这些中断可进行屏蔽控制。通过设置优先级指定标志寄存器（PR00L， PR00H， PR01L， PR01H， PR02L， PR10L， PR10H， PR11L， PR11H， PR12L），可屏蔽中断可分为 4 个优先级组。
当产生较高优先级中断时，多重中断服务被应用到较低级中断。如果同时产生具有相同优先级的两个或更多的中断请求，则根据向量中断服务的优先级进行处理。优先级顺序参见表 16-1。
产生待机解除信号和解除 STOP 和 HALT 模式。
外部中断请求和内部中断请求作为可屏蔽中断被提供。

(2) 软件中断

这是一个由执行 BRK 指令产生的向量中断。即使中断被禁止它也能被响应。软件中断不受中断优先级控制。

16.2 中断源和配置

中断源包括可屏蔽中断和软件中断。此外，还有五个复位源（参见表 16-1）。

表 16-1. 中断源列表 (1/2)

中断类型	默认优先级 ^{注1}	中断源		内部/外部	向量表地址	基本配置类型 ^{注2}
		名称	触发			
可屏蔽	0	INTWDTI	看门狗定时器间隔 ^{注3} (溢出时间的 75%)	内部	0004H	(A)
	1	INTLVI	低压检测 ^{注4}		0006H	
	2	INTP0	引脚输入边沿检测	外部	0008H	(B)
	3	INTP1			000AH	
	4	INTP2			000CH	
	5	INTP3			000EH	
	6	INTP4			0010H	
	7	INTP5			0012H	
	8	INTCMP0	CMP0 检测	内部	0016H	(A)
	9	INTCMP1	CMP1 检测		0018H	
	10	INTDMA0	DMA0 传输结束		001AH	
	11	INTDMA1	DMA1 传输结束		001CH	
	12	INTST0/ INTCSI00	UART0 发送结束/ CSI00 通信结束		001EH	
	13	INTSR0/ INTCSI01	UART0 接收结束/ CSI01 通信结束		0020H	
	14	INTSRE0	UART0 通信错误发生		0022H	
	15	INTST1 /INTCSI10 /INTIIC10	UART1 发送结束/ CSI10 通信结束/ IIC10 通信结束		0024H	
	16	INTSR1	UART1 接收结束		0026H	
	17	INTSRE1	UART1 通信错误发生		0028H	
	18	INTIICA ^{注5}	IICA 通信结束		002AH	
	19	INTTM00	定时器通道 0 计数或捕获结束		002CH	
	20	INTTM01	定时器通道 1 计数或捕获结束		002EH	
	21	INTTM02	定时器通道 2 计数或捕获结束		0030H	
	22	INTTM03	定时器通道 3 计数或捕获结束		0032H	
	23	INTAD	A/D 转换结束		0034H	

- 注
1. 如果两个或多个可屏蔽中断同时发生时，默认优先级决定中断顺序。0 指示最高优先级，33 指示最低优先级。
 2. 基本配置类型 (A) 至 (C) 对应着图 16-1 中的 (A) 至 (C)。
 3. 选项字节 (000C0H) 的位 7 (WDTINT) 设置为 1 时。
 4. 低-电压检测寄存器 (LVIM) 的位 1 (LVIMD) 清除为 0 时。
 5. 78K0R/KC3-L 的 44-引脚产品不配备。

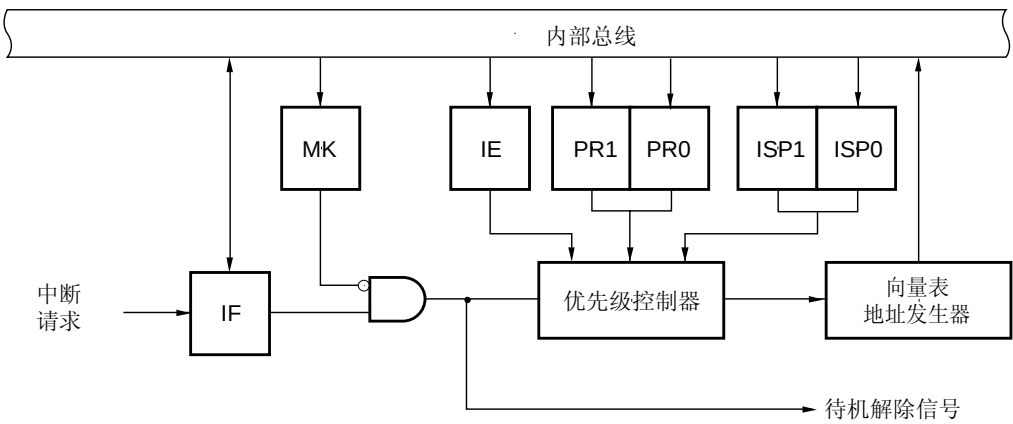
表 16-1. 中断源列表 (2/2)

中断类型	默认优先级 ^{注1}	中断源		内部/外部	向量表地址	基本配置类型 ^{注2}
		名称	触发			
<R>	24	INTRTC	实时计数器/报警匹配检测的固定周期信号	内部	0036H	(A)
	25	INTRTCI	实时计数器的间隔信号检测		0038H	
	26	INTKR	按键返回信号检测	外部	003AH	(C)
	27	INTMD	除法操作结束	内部	0040H	(A)
	28	INTTM04	定时器通道 4 计数或捕获结束		0042H	
	29	INTTM05	定时器通道 5 计数或捕获结束		0044H	
	30	INTTM06	定时器通道 6 计数或捕获结束		0046H	
	31	INTTM07	定时器通道 7 计数或捕获结束		0048H	
	32	INTP6	引脚输出边沿检测	外部	004AH	(B)
	33	INTP7			004CH	
<R>	软件	BRK	BRK 指令执行	—	007EH	(D)
<R>	复位	RESET	RESET 引脚输入	—	0000H	—
		POC	上电清除			
		LVI	电压检测 ^{注3}			
		WDT	看门狗定时器溢出			
		TRAP	执行非法指令 ^{注4}			

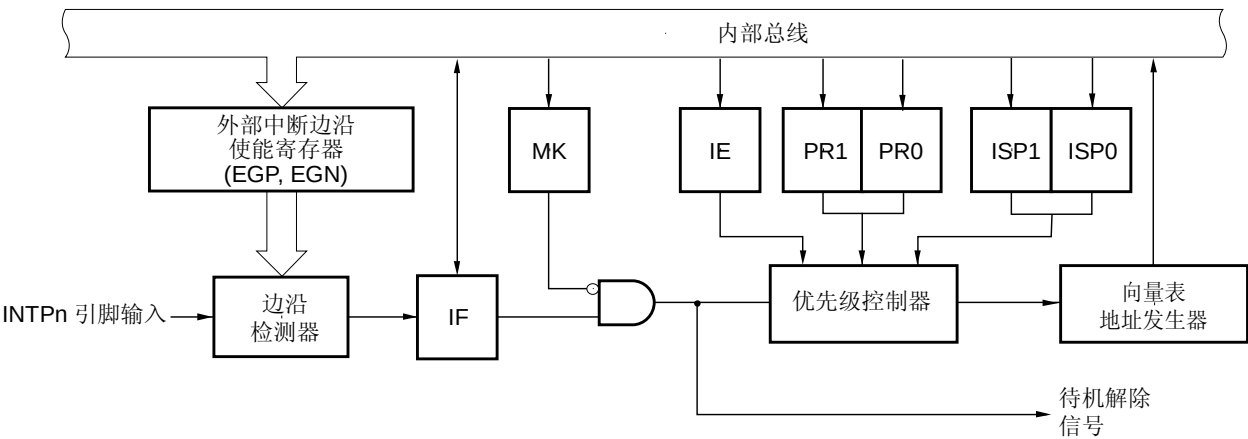
- 注
1. 如果两个或多个可屏蔽中断同时发生时，默认优先级决定中断顺序。0 指示最高优先级，33 指示最低优先级。
 2. 基本配置类型 (A) 至 (C) 对应着图 16-1 中的 (A) 至 (C)。
 3. 低-电压检测寄存器 (LVIM) 的位 1 (LVIMD) 设置为 1 时。
 4. 执行 FFH 中的指令代码时。
在线仿真器或片上调试仿真器不能产生通过执行非法指令的复位。

图 16-1. 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)

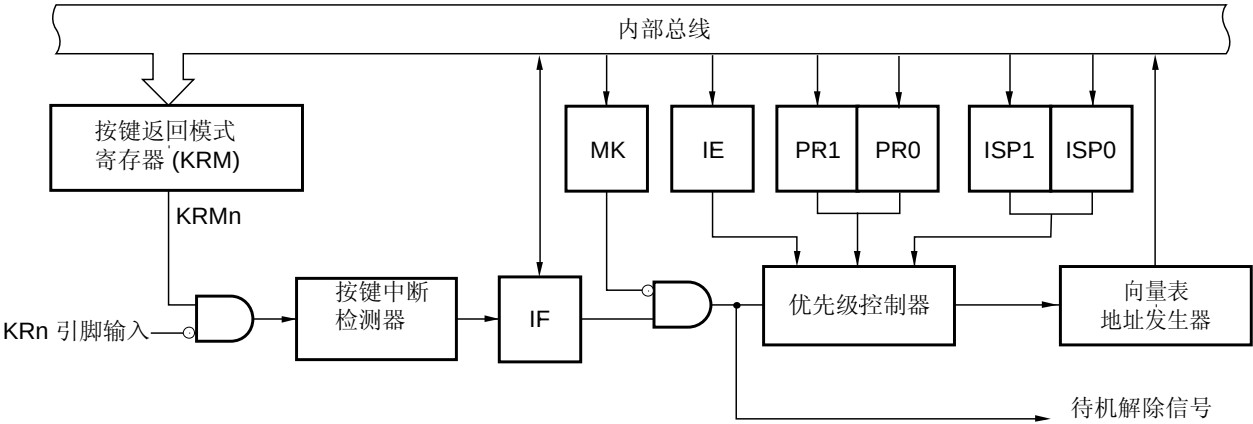


- IF: 中断请求标志
- IE: 中断使能标志
- ISP0: 服务中优先级标志 0
- ISP1: 服务中优先级标志 1
- MK: 中断屏蔽标志
- PR0: 优先级指定标志 0
- PR1: 优先级指定标志 1

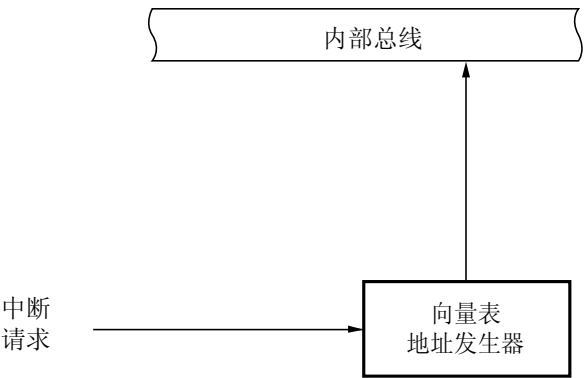
备注 n = 0 至 7

图 16-1. 中断功能的基本配置 (2/2)

<R> (C) 外部可屏蔽中断 (INTKR)



(D) 软件中断



- IF: 中断请求标志
IE: 中断使能标志
ISP0: 服务中优先级标志 0
ISP1: 服务中优先级标志 1
MK: 中断屏蔽标志
PR0: 优先级指定标志 0
PR1: 优先级指定标志 1

备注 n = 0 至 7

16.3 控制中断功能的寄存器

以下 6 中类型的寄存器可用于控制中断功能。

- 中断请求标志寄存器 (IF0L, IF0H, IF1L, IF1H, IF2L)
- 中断屏蔽标志寄存器 (MK0L, MK0H, MK1L, MK1H, MK2L)
- 优先级指定标志寄存器 (PR00L, PR00H, PR01L, PR01H, PR02L, PR10L, PR10H, PR11L, PR11H, PR12L)
- 外部中断上升沿使能寄存器 (EGP0)
- 外部中断下降沿使能寄存器 (EGN0)
- 程序状态字 (PSW)

表 16-2 为与中断请求源对应的中断请求标志、中断屏蔽标志以及优先级指定标志的列表。

表 16-2. 与中断请求源对应的标志 (1/2)

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
INTWDTI	WDTIIF	IF0L	WDTIMK	MK0L	WDTIPR0, WDTIPR1	PR00L, PR10L
INTLVI	LVIIIF		LVIMK		LVIPR0, LVIPR1	
INTP0	PIF0		PMK0		PPR00, PPR10	
INTP1	PIF1		PMK1		PPR01, PPR11	
INTP2	PIF2		PMK2		PPR02, PPR12	
INTP3	PIF3		PMK3		PPR03, PPR13	
INTP4	PIF4		PMK4		PPR04, PPR14	
INTP5	PIF5		PMK5		PPR05, PPR15	
INTCMP0	CMPIF0	IF0H	CMPMK0	MK0H	CMPPR00, CMPPR10	PR00H, PR10H
INTCMP1	CMPIF1		CMPMK1		CMPPR01, CMPPR11	
INTDMA0	DMAIF0		DMAMK0		DMAPR00, DMAPR10	
INTDMA1	DMAIF1		DMAMK1		DMAPR01, DMAPR11	
INTST0 ^{‡1}	STIF0 ^{‡1}		STMK0 ^{‡1}		STPR00, STPR10 ^{‡1}	
INTCSI00 ^{‡1}	CSIIIF00 ^{‡1}		CSIMK00 ^{‡1}		CSIPR000, CSIPR100 ^{‡1}	
INTSR0 ^{‡2}	SRIF0 ^{‡2}		SRMK0 ^{‡2}		SRPR00, SRPR10 ^{‡2}	
INTCSI01 ^{‡2}	CSIIIF01 ^{‡2}		CSIMK01 ^{‡2}		CSIPR001, CSIPR101 ^{‡2}	
INTSRE0	SREIF0		SREMK0		SREPR00, SREPR10	

- 注
1. 禁止同时使用 UART0 和 CSI00，因为它们共享中断请求源标志。如果发生中断源 INST0 和 INTCSI00 其中的一个，则 IF0H 的位 5 被置 1。MK0H，PR00H 和 PR10H 的位 5 支持这两种中断源。
 2. 禁止同时使用 UART0 和 CSI01，因为它们共享中断请求源标志。如果发生中断源 INSR0 和 INTCSI01 其中的一个，则 IF0H 的位 6 被置 1。MK0H，PR00H 和 PR10H 的位 6 支持这两种中断源。

表 16-2. 与中断请求源对应的标志 (2/2)

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
INTST1 ^{注1}	STIF1 ^{注1}	IF1L	STMK1 ^{注1}	MK1L	STPR01, STPR11 ^{注1}	PR01L , PR11L
INTCSI10 ^{注1}	CSIIIF10 ^{注1}		CSIMK10 ^{注1}		CSIPR010, CSIPR110 ^{注1}	
INTIIC10 ^{注1}	IICIF10 ^{注1}		IICMK10 ^{注1}		IICPR010, IICPR110 ^{注1}	
INTSR1	SRIF1		SRMK1		SRPR01, SRPR11	
INTSRE1	SREIF1		SREMK1		SREPR01, SREPR11	
INTIICA ^{注2}	IICAIF ^{注2}		IICAMK ^{注2}		IICAPR0, IICAPR1 ^{注2}	
INTTM00	TMIF00		TMMK00		TMPR000, TMPR100	
INTTM01	TMIF01		TMMK01		TMPR001, TMPR101	
INTTM02	TMIF02		TMMK02		TMPR002, TMPR102	
INTTM03	TMIF03		TMMK03		TMPR003, TMPR103	
INTAD	ADIF	IF1H	ADMK	MK1H	ADPR0, ADPR1	PR01H , PR11H
INTRTC	RTCIF		RTCMK		RTCPR0, RTCPR1	
INTRTCI	RTCIIIF		RTCIMK		RTCIPR0, RTCIPR1	
INTKR	KRIF		KRMK		KRPR0, KRPR1	
INTMD	MDIF		MDMK		MDPR0, MDPR1	
INTTM04	TMIF04		TMMK04		TMPR004, TMPR104	
INTTM05	TMIF05	IF2L	TMMK05	MK2L	TMPR005, TMPR105	PR02L , PR12L
INTTM06	TMIF06		TMMK06		TMPR006, TMPR106	
INTTM07	TMIF07		TMMK07		TMPR007, TMPR107	
INTP6	PIF6		PMK6		PPR06, PPR16	
INTP7	PIF7		PMK7		PPR07, PPR17	

- 注 1. 禁止同时使用 UART1, CSI10 和 IIC10, 因为它们共享中断请求源标志。如果发生中断源 INTST1, INTCSI10 和 INTIIC10 其中的一个, 则 IF1L 的位 0 被置 1。MK1L, PR01L 和 PR11L 的位 0 支持这三种中断源。
2. 78K0R/KC3-L 的 44-引脚产品不配备。

(1) 中断请求标志寄存器 (IF0L, IF0H, IF1L, IF1H, IF2L)

当传输相应的中断请求或执行指令时，中断请求标志被置 1。基于中断请求的响应或复位信号的产生而执行指令时，这些寄存器被清除为 0。

当中断被响应时，中断请求标志自动清除然后进入中断例程。

可通过 1 位或 8 位的存储操作指令设置 IF0L, IF0H, IF1L, IF1H 和 IF2L。当 IF0L 和 IF0H 与 IF1L 和 IF1H 组合形成 16 位寄存器 IF0 与 IF1 时，它们可用 16 位存储操作指令进行设置。用作 IF2 的 IF2L 也可用 16 位存储操作指令进行设置。

产生的复位信号将这些寄存器清除为 00H。

备注 如果执行向该寄存器写入数据的指令，则指令执行的时钟数目需增加 2 个时钟。

图 16-2. 中断请求标志寄存器 (IF0L, IF0H, IF1L, IF1H, IF2L) 的格式 (1/2)

地址: FFFE0H 复位后: 00H R/W

符号	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
IF0L	PIF5	PIF4	PIF3	PIF2	PIF1	PIF0	LVIF	WDTIF

地址: FFFE1H 复位后: 00H R/W

符号	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
IF0H	SREIF0	SRIF0 CSIF01	STIF0 CSIF00	DMAIF1	DMAIF0	CMPIF1	CMPIF0	0

地址: FFFE2H 复位后: 00H R/W

符号	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
IF1L	TMIF03	TMIF02	TMIF01	TMIF00	IICAF [※]	SREIF1	SRIF1	STIF1 CSIF10 IICIF10

地址: FFFE3H 复位后: 00H R/W

符号	<7>	<6>	5	4	<3>	<2>	<1>	<0>
IF1H	TMIF04	MDIF	0	0	KRIF	RTCIIF	RTCIF	ADIF

地址: FFFD0H 复位后: 00H R/W

符号	7	6	5	<4>	<3>	<2>	<1>	<0>
IF2L	0	0	0	PIF7	PIF6	TMIF07	TMIF06	TMIF05

注 78K0R/KC3-L 的 44-引脚产品不配备。

图 16-2. 中断请求标志寄存器（IF0L, IF0H, IF1L, IF1H, IF2L）的格式（2/2）

XXIFX	中断请求标志
0	不产生中断请求信号
1	产生中断请求，中断请求状态

- 注意事项**
1. 确保将 IF0H 的位 0、IF1H 的位 4 和位 5 以及 IF2L 的位 5 至 7 清除为 0。
 2. 在待机解除后若要操作定时器、串行接口或 A/D 转换器，需将中断请求标志清除后再操作。中断请求标志有可能被噪声设置。
 3. 当处理一个中断请求标志寄存器的标志时，需使用 1 位存储操作指令（CLR1）。因为被编译的汇编程序必须是 1 位存储操作指令（CLR1），所以用 C 语言描述时，需使用如“IF0L.0 = 0;”或“_asm (“clr1 IF0L, 0”);”之类的位操作指令。

如果用 8 位存储操作指令如“IF0L &= 0xfe;”用 C 语言描述一个程序并被编译，则其变成三种指令的汇编程序。

```
mov a, IF0L
and a, #0FEH
mov IF0L, a
```

这种情况下，即使同一中断请求标志寄存器（IF0L）的另一位的请求标志在执行“mov a, IF0L”和“mov IF0L, a”中间的时序时被设置为 1，该标志在执行“mov IF0L, a”时也会被清除为 0。因此，在 C 语言中使用 8 位存储操作指令必须格外谨慎。

(2) 中断屏蔽标志寄存器 (MK0L, MK0H, MK1L, MK1H, MK2L)

中断屏蔽标志用于允许/禁止相应的可屏蔽中断服务。
可通过 1 位或 8 位的存储操作指令设置 MK0L, MK0H, MK1L, MK1H 和 MK2L。当 MK0L 和 MK0H 与 MK1L 和 MK1H 组合形成 16 位寄存器 MK0 和 MK1 时, 它们可用 16 位存储操作指令进行设置。用作 MK2 的 MK2L 也可用 16 位存储操作指令进行设置。
产生的复位信号将这些寄存器设置为 FFH。

备注 如果执行向该寄存器写入数据的指令, 则指令执行的时钟数目需增加 2 个时钟。

图 16-3. 中断屏蔽标志寄存器 (MK0L, MK0H, MK1L, MK1H, MK2L) 的格式

地址： FFFE4H 复位后： FFH R/W

符号	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
MK0L	PMK5	PMK4	PMK3	PMK2	PMK1	PMK0	LVIMK	WDTIMK

地址： FFFE5H 复位后： FFH R/W

符号	<7>	<6>	<5>	<4>	<3>	<2>	<1>	0
MK0H	SREMK0	SRMK0 CSIMK01	STMK0 CSIMK00	DMAMK1	DMAMK0	CMPMK1	CMPMK0	1

地址： FFFE6H 复位后： FFH R/W

符号	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
MK1L	TMMK03	TMMK02	TMMK01	TMMK00	IICAMK [‡]	SREMK1	SRMK1	STMK1 CSIMK10 IICMK10

地址： FFFE7H 复位后： FFH R/W

符号	<7>	<6>	5	4	<3>	<2>	<1>	<0>
MK1H	TMMK04	MDMK	1	1	KRMK	RTCIMK	RTCMK	ADMK

地址： FFFD4H 复位后： FFH R/W

符号	7	6	5	<4>	<3>	<2>	<1>	<0>
MK2L	1	1	1	PMK7	PMK6	TMMK07	TMMK06	TMMK05

XXMKX	中断服务控制						
0	使能中断服务						
1	禁止中断服务						

注 78K0R/KC3-L 的 44-引脚产品不配备。

注意事项 确保将 MK0H 的位 0、MK1H 的位 4 和位 5 以及 MK2L 的位 5 至 7 设置为 1。

(3) 优先级指定标志寄存器 (PR00L, PR00H, PR01L, PR01H, PR02L, PR10L, PR10H, PR11L, PR11H, PR12L)

优先级指定标志寄存器用于设置相应的可屏蔽中断的优先等级。

联合使用 PR0xy 和 PR1xy (xy = 0L, 0H, 1L, 1H 或 2L) 设置优先等级。

可用 1 位或 8 位存储操作指令设置 PR00L, PR00H, PR01L, PR01H, PR02L, PR10L, PR10H, PR11L, PR11H 和 PR12L。如果 PR00L 和 PR00H、PR01L 和 PR01H、PR10L 和 PR10H 与 PR11L 和 PR11H 组合成 16 位寄存器 PR00, PR01, PR10 与 PR11, 则它们可用 16 位存储操作指令设置。用作 PR02 的 PR02L 和用作 PR12 的 PR12L 也可用 16 位存储操作指令进行设置。

产生的复位信号将这些寄存器设置为 FFH。

备注 如果执行向该寄存器写入数据的指令, 则指令执行的时钟数目需增加 2 个时钟。

图 16-4. 优先级指定标志寄存器

(PR00L, PR00H, PR01L, PR01H, PR02L, PR10L, PR10H, PR11L, PR11H, PR12L) 的格式 (1/2)

地址: FFFE8H 复位后: FFH R/W

符号	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
PR00L	PPR05	PPR04	PPR03	PPR02	PPR01	PPR00	LVIPR0	WDTIPR0

地址: FFFECH 复位后: FFH R/W

符号	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
PR10L	PPR15	PPR14	PPR13	PPR12	PPR11	PPR10	LVIPR1	WDTIPR1

地址: FFFE9H 复位后: FFH R/W

符号	<7>	<6>	<5>	<4>	<3>	<2>	<1>	0
PR00H	SREPR00	SRPR00 CSIPR001	STPR00 CSIPR000	DMAPR01	DMAPR00	CMPPR01	CMPPR00	1

地址: FFFEDH 复位后: FFH R/W

符号	<7>	<6>	<5>	<4>	<3>	<2>	<1>	0
PR10H	SREPR10	SRPR10 CSIPR101	STPR10 CSIPR100	DMAPR11	DMAPR10	CMPPR11	CMPPR10	1

图 16-4. 优先级指定标志寄存器

(PR00L, PR00H, PR01L, PR01H, PR02L, PR10L, PR10H, PR11L, PR11H, PR12L) 的格式 (2/2)

地址: FFFEAH 复位后: FFH R/W

符号	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
PR01L	TMPR003	TMPR002	TMPR001	TMPR000	IICAPR0 ^注	SREPR01	SRPR01	STPR01 CSIPR010 IICPR010

地址: FFFEEH 复位后: FFH R/W

符号	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
PR11L	TMPR103	TMPR102	TMPR101	TMPR100	IICAPR1 ^注	SREPR11	SRPR11	STPR11 CSIPR110 IICPR110

地址: FFFEBH 复位后: FFH R/W

符号	<7>	<6>	5	4	<3>	<2>	<1>	<0>
PR01H	TMPR004	MDPR0	1	1	KRPR0	RTCIPR0	RTCPR0	ADPR0

地址: FFFEFH 复位后: FFH R/W

符号	<7>	<6>	5	4	<3>	<2>	<1>	<0>
PR11H	TMPR104	MDPR1	1	1	KRPR1	RTCIPR1	RTCPR1	ADPR1

地址: FFFD8H 复位后: FFH R/W

符号	7	6	5	<4>	<3>	<2>	<1>	<0>
PR02L	1	1	1	PPR07	PPR06	TMPR007	TMPR006	TMPR005

地址: FFFDCH 复位后: FFH R/W

符号	7	6	5	<4>	<3>	<2>	<1>	<0>
PR12L	1	1	1	PPR17	PPR16	TMPR107	TMPR106	TMPR105

XXPR1X	XXPR0X	优先级选择
0	0	指定 0 等级 (高优先级)
0	1	指定 1 等级
1	0	指定 2 等级
1	1	指定 3 等级 (低优先级)

注 78K0R/KC3-L 的 44-引脚产品不配备。

注意事项 确保将 PR00H 和 PR10H 的位 0, PR01H 和 PR11H 的位 4 和 5 以及 PR02L 和 PR12L 的位 5 至 7 设置为 1。

(4) 外部中断上升沿使能寄存器 (EGP0)，外部中断下降沿使能寄存器 (EGN0)

这些寄存器为 INTP0 至 INTP7 指定有效沿。

可通过 1 位或 8 位的存储操作指令设置 EGP0 和 EGN0。

产生的复位信号将这些寄存器清除为 00H。

**图 16-5. 外部中断上升沿使能寄存器 (EGP0)
和外部中断下降沿使能寄存器 (EGN0) 的格式**

地址: FFF38H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
EGP0	EGP7	EGP6	EGP5	EGP4	EGP3	EGP2	EGP1	EGP0

地址: FFF39H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
EGN0	EGN7	EGN6	EGN5	EGN4	EGN3	EGN2	EGN1	EGN0

EGPn	EGNn	INTPn 引脚有效沿选择 (n = 0 至 7)
0	0	禁止边沿检测
0	1	下降沿
1	0	上升沿
1	1	双边沿

表 16-3 显示了与 EGPn 和 EGNn 对应的端口。

表 16-3 与 EGPn 和 EGNn 对应的端口

检测使能寄存器		边沿检测端口	中断请求信号
EGP0	EGN0	P120	INTP0
EGP1	EGN1	P31	INTP1
EGP2	EGN2	P32	INTP2
EGP3	EGN3	P80	INTP3
EGP4	EGN4	P70	INTP4
EGP5	EGN5	P71	INTP5
EGP6	EGN6	P72	INTP6
EGP7	EGN7	P82	INTP7

注意事项 因为当外部中断功能转换为端口功能时，可能会检测到边沿，所以通过清除 EGPn 和 EGNn 为 0 选择端口模式。

备注 n = 0 至 7

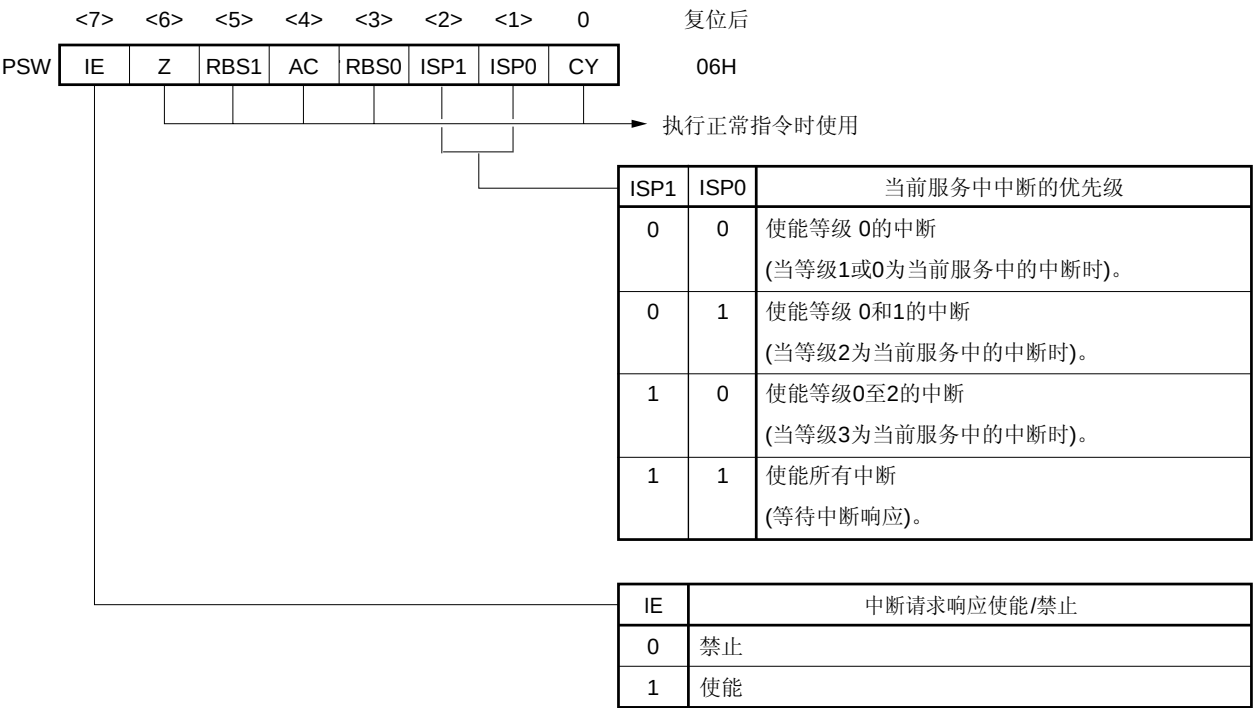
(5) 程序状态字 (PSW)

程序状态字是用于保持指令执行结果和中断请求当前状态的寄存器。设置可屏蔽中断使能/禁止的 IE 标志以及控制多重中断服务的 ISP0 和 ISP1 标志被映射到 PSW。

除可进行 8 位读取/写入外，该寄存器还可利用位操作指令和专用指令 (EI 和 DI) 执行操作。当向量中断请求被响应时，通过指令 BRK 指令，则 PSW 的内容自动存入堆栈且 IE 标志复位为 0。如果可屏蔽中断请求被响应，则被响应中断的优先级指定标志的内容被传送到 ISP0 和 ISP1 标志。PSW 内容也可用 PUSH PSW 指令存入堆栈。用 RETI，RETB 和 POP PSW 指令将其内容从堆栈恢复。

产生的复位信号将 PSW 设置为 06H。

图 16-6. 程序状态字的配置



16.4 中断服务操作

16.4.1 可屏蔽中断请求响应

当中断请求标志设置为 1 且相应于该中断请求的屏蔽（MK）标志被清除为 0 时，可屏蔽中断请求被响应。如果中断处于中断使能状态（当 IE 标志被设置为 1 时），则向量中断请求会被响应。但是，在较高优先级中断请求服务期间，较低优先级的中断请求不被响应。

从可屏蔽中断请求的产生到向量中断服务被执行所需的时间列于以下表 16-4 中。

关于中断请求响应时序，参见图 16-8 和 16-9。

表 16-4. 从可屏蔽中断传输到服务中断的时间

	最短时间	最长时间 ^注
服务时间	9 个时钟	14 个时钟

注 如果中断请求刚好在 RET 指令前产生，则等待时间会变长。

备注 1 个时钟：1/fCLK（fCLK： CPU 时钟）

如果同时产生两个或更多的可屏蔽中断请求，则先响应在优先级指定标志中指定具有较高优先等级的请求。如果两个或更多的中断请求具有相同优先等级，则先响应最高默认优先级的请求。

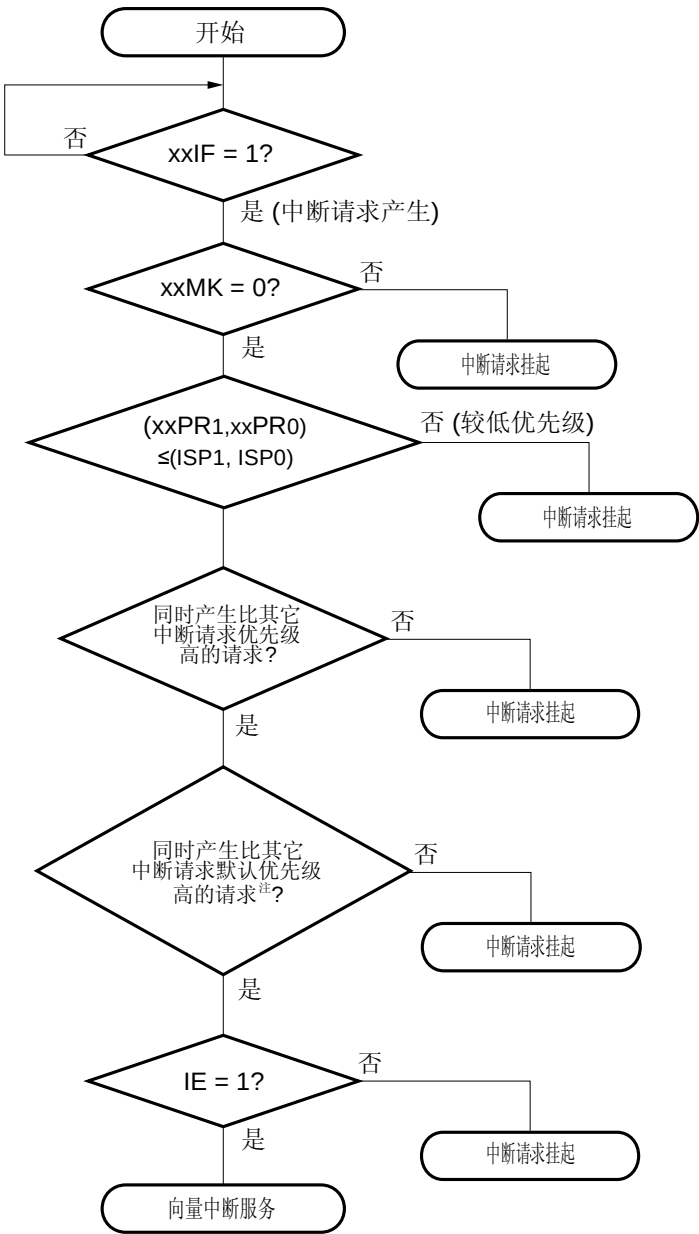
当被挂起的中断请求被识别时，该请求被响应。

图 16-7 显示了中断请求响应规则。

如果可屏蔽中断请求被响应，则其内容按 PSW 的顺序存入堆栈，然后 PC，IE 标志复位为（0），且与响应的中断对应的优先级指定标志的内容被传送到 ISP1 和 ISP0 标志。由各中断请求确定的向量表数据被载入 PC 并转移。

可以用 RETI 指令从中断中恢复。

图 16-7. 中断请求响应处理规则



xxIF: 中断请求标志
xxMK: 中断屏蔽标志
xxPR0: 优先级指定标志 0
xxPR1: 优先级指定标志 1
IE: 控制可屏蔽中断请求响应的标志（1 = 允许，0 = 禁止）
ISP0, ISP1: 指示当前服务的中断的优先级级别的标志（参见图 16-6）

注 关于默认优先级，参见表 16-1 中断源列表。

图 16-8. 中断请求响应时序（最短时间）

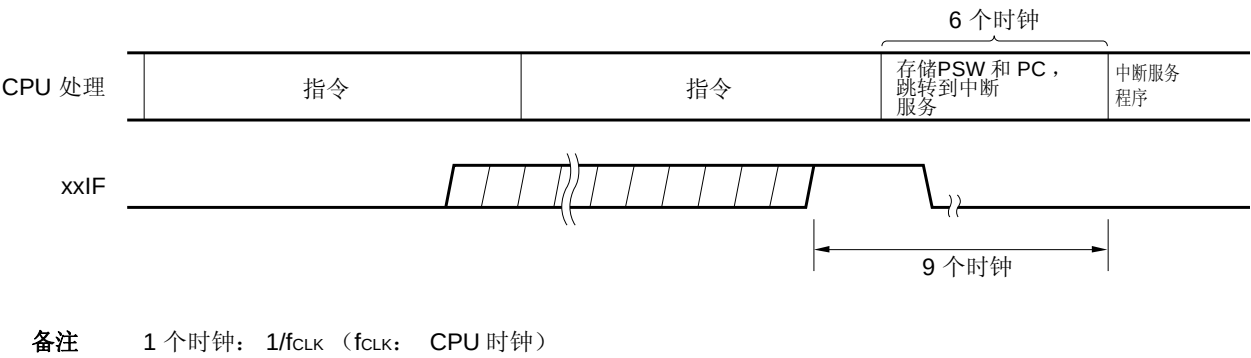
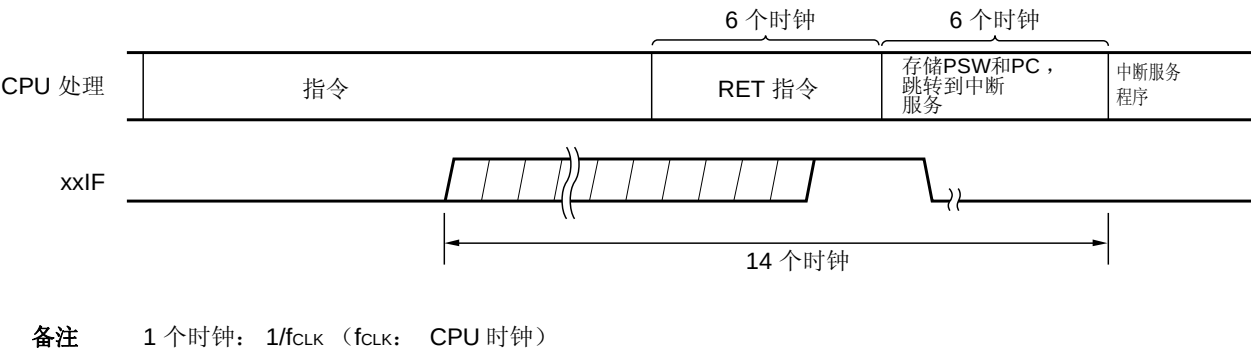


图 16-9. 中断请求响应时序（最长时间）



16.4.2 软件中断请求响应

软件中断请求通过执行 **BRK** 指令响应。软件中断不能被禁止。

如果软件中断请求被响应，则该内容按程序状态字（**PSW**）的顺序存入堆栈，然后程序计数器（**PC**），**IE** 标志复位为（0），且向量表（0007EH，0007FH）的内容被载入 **PC** 并转移。

可以用 **RETB** 指令从软件中断中恢复。

注意事项 从软件中断中恢复禁止使用 **RETI** 指令。

16.4.3 多重中断服务

在一个中断执行期间另一中断请求被响应时，发生多重中断服务。

多重中断服务不会发生除非选择了中断请求响应使能状态（**IE = 1**）。当一个中断请求被响应时，中断请求响应标志变成禁止（**IE = 0**）。因此，若要允许多重中断服务，需要用 **EI** 指令在中断服务期间将 **IE** 标志设置为（1），以允许中断响应。

此外，即使允许中断，多重中断服务也可能不被使能，这要受中断优先级的控制。两种类型的优先级控制可用：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。

中断使能状态下，如果产生一个中断请求，而该中断请求的优先级等于或高于当前服务的中断时，该请求作为多重中断服务被响应。如果中断服务期间产生一个中断，而该中断的优先级低于当前服务的中断时，该请求不作为多重中断服务被响应。因为处于中断禁止状态或因为其具有较低优先级而不被允许的中断请求被挂起。当前中断服务结束时，至少执行一个主处理指令后，挂起的中断请求被响应。

表 16-5 显示了允许作为多重中断服务的中断请求之间的关系，图 16-10 显示了多重中断服务的举例。

**表 16-5. 中断服务期间允许作为多重中断服务的
的中断请求之间的关系**

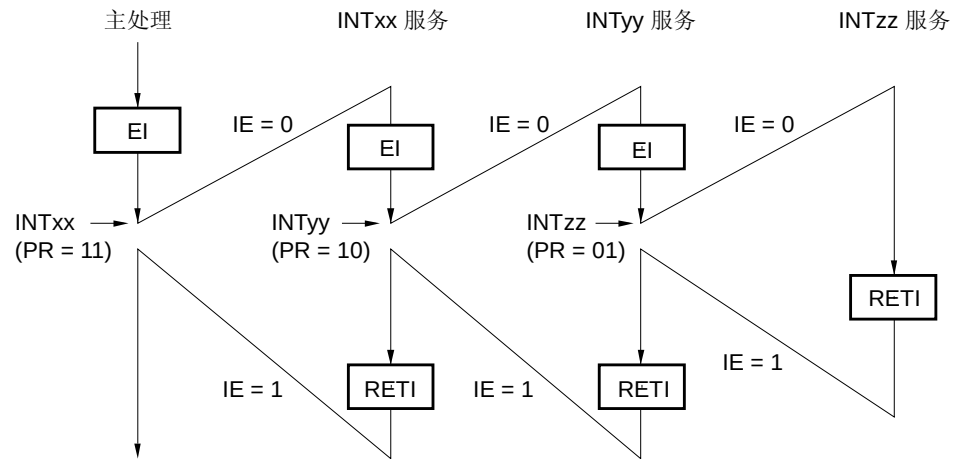
多重中断请求 正在服务的中断		可屏蔽中断请求								软件中断 请求
		优先等级 0 (PR = 00)		优先等级 1 (PR = 01)		优先等级 2 (PR = 10)		优先等级 3 (PR = 11)		
		IE = 1	IE = 0	IE = 1	IE = 0	IE = 1	IE = 0	IE = 1	IE = 0	
可屏蔽中断	ISP1 = 0 ISP0 = 0	○	×	×	×	×	×	×	×	○
	ISP1 = 0 ISP0 = 1	○	×	○	×	×	×	×	×	○
	ISP1 = 1 ISP0 = 0	○	×	○	×	○	×	×	×	○
	ISP1 = 1 ISP0 = 1	○	×	○	×	○	×	○	×	○
软件中断		○	×	○	×	○	×	○	×	○

备注

- : 多重中断服务使能
- ×: 多重中断服务禁止
- ISP0, ISP1 和 IE 是 PSW 中包含的标志。
 - ISP1 = 0, ISP0 = 0: 等级 1 或等级 0 的中断被服务。
 - ISP1 = 0, ISP0 = 1: 等级 2 的中断被服务。
 - ISP1 = 1, ISP0 = 0: 等级 3 的中断被服务。
 - ISP1 = 1, ISP0 = 1: 等待中断响应。
 - IE = 0: 中断请求响应被禁止。
 - IE = 1: 中断请求响应被使能。
- PR 是 PR00L, PR00H, PR01L, PR01H, PR02L, PR10L, PR10H, PR11L, PR11H 和 PR12L 中包含的标志。
 - PR = 00: 用 $\times\times PR1\times = 0$, $\times\times PR0\times = 0$ 指定等级 0 (较高优先等级)
 - PR = 01: 用 $\times\times PR1\times = 0$, $\times\times PR0\times = 1$ 指定等级 1
 - PR = 10: 用 $\times\times PR1\times = 1$, $\times\times PR0\times = 0$ 指定等级 2
 - PR = 11: 用 $\times\times PR1\times = 1$, $\times\times PR0\times = 1$ 指定等级 3 (较低优先等级)

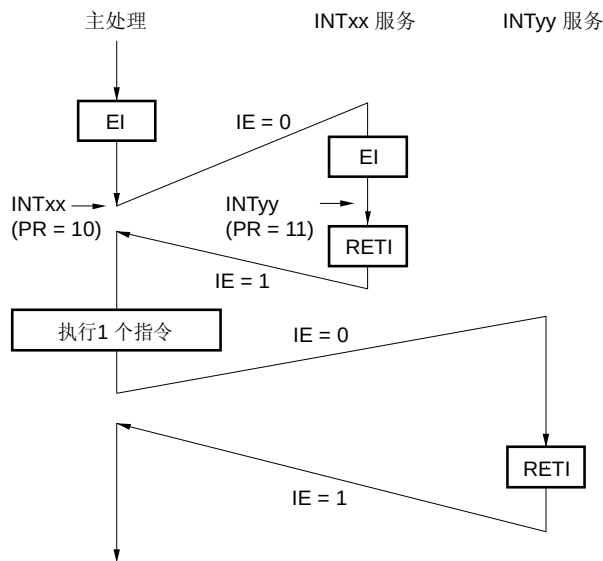
图 16-10. 多重中断服务的举例 (1/2)

例 1. 多重中断服务发生两次



中断 INTxx 服务期间，INTyy 和 INTzz 两个中断请求被响应，因而发生多重中断服务。在每个中断请求响应前，总是发出 EI 指令以允许中断请求响应。

例 2. 由于优先级控制不发生多重中断服务

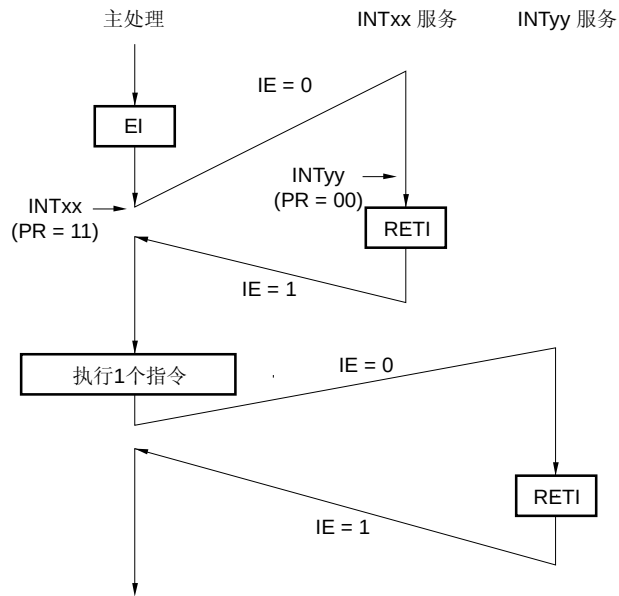


因为中断请求 INTyy 的优先级低于 INTxx，所以在中断 INTxx 的服务期间发出的中断请求 INTyy 不被响应，因而不发生多重中断服务。INTyy 中断请求被挂起，在执行一个主处理指令后该请求被响应。

- PR = 00: 用 $\times\times PR1\times = 0$, $\times\times PR0\times = 0$ 指定等级 0 (较高优先级)
- PR = 01: 用 $\times\times PR1\times = 0$, $\times\times PR0\times = 1$ 指定等级 1
- PR = 10: 用 $\times\times PR1\times = 1$, $\times\times PR0\times = 0$ 指定等级 2
- PR = 11: 用 $\times\times PR1\times = 1$, $\times\times PR0\times = 1$ 指定等级 3 (较低优先级)
- IE = 0: 中断请求响应被禁止。
- IE = 1: 中断请求响应被使能。

图 16-10. 多重中断服务的举例 (2/2)

例 3. 由于不允许中断而不发生多重中断服务



中断 INT_{xx} 服务期间不允许中断（不发出 EI 指令），因此中断请求 INT_{yy} 不被响应从而不发生多重中断服务。 INT_{yy} 中断请求被挂起，在执行一个主处理指令后该请求被响应。

- $PR = 00$: 用 $xxPR1x = 0$, $xxPR0x = 0$ 指定等级 0（较高优先级）
 $PR = 01$: 用 $xxPR1x = 0$, $xxPR0x = 1$ 指定等级 1
 $PR = 10$: 用 $xxPR1x = 1$, $xxPR0x = 0$ 指定等级 2
 $PR = 11$: 用 $xxPR1x = 1$, $xxPR0x = 1$ 指定等级 3（较低优先级）
 $IE = 0$: 中断请求响应被禁止。
 $IE = 1$: 中断请求响应被使能。

16.4.4 中断请求保持

有一些指令，即使向它们发出中断请求而另一指令正在执行时，中断响应会保持挂起，直到下一个指令执行结束。这些指令（中断请求保持指令）列出如下：

- MOV PSW, #byte
- MOV PSW, A
- MOV1 PSW. bit, CY
- SET1 PSW. bit
- CLR1 PSW. bit
- RETB
- RETI
- POP PSW
- BTCLR PSW. bit, \$addr8
- EI
- DI
- SKC
- SKNC
- SKZ
- SKNZ
- 对 the IF0L, IF0H, IF1L, IF1H, IF2L, MK0L, MK0H, MK1L, MK1H, MK2L, PR00L, PR00H, PR01L, PR01H, PR02L, PR10L, PR10H, PR11L, PR11H 和 PR12L 寄存器的操作指令

注意事项 BRK 指令不是以上列出的中断请求保持指令之一。但是，因为执行 BRK 指令使 IE 标志被清除，所以软件中断有效。因此即使在 BRK 指令执行期间产生可屏蔽中断请求，该中断请求也不被响应。

图 16-11 显示了中断请求被保持挂起的时序。

图 16-11. 中断请求保持



第十七章 按键中断功能

根据产品的不同，按键中断输入通道的数量不同。

	78K0R/KC3-L	78K0R/KD3-L	78K0R/KE3-L
按键中断输入通道	6 通道	8 通道	

17.1 按键中断功能

通过设置按键返回模式寄存器（KRM）和将下降沿输入到按键中断输入引脚（KR0 至 KR7），可以生成按键中断（INTKR）。

表 17-1. 按键中断检测引脚的分配

标志	描述说明
KRM0	用 1 位单元控制 KR0 信号。
KRM1	用 1 位单元控制 KR1 信号。
KRM2	用 1 位单元控制 KR2 信号。
KRM3	用 1 位单元控制 KR3 信号。
KRM4	用 1 位单元控制 KR4 信号。
KRM5	用 1 位单元控制 KR5 信号。
KRM6	用 1 位单元控制 KR6 信号。
KRM7	用 1 位单元控制 KR7 信号。

17.2 按键中断的配置

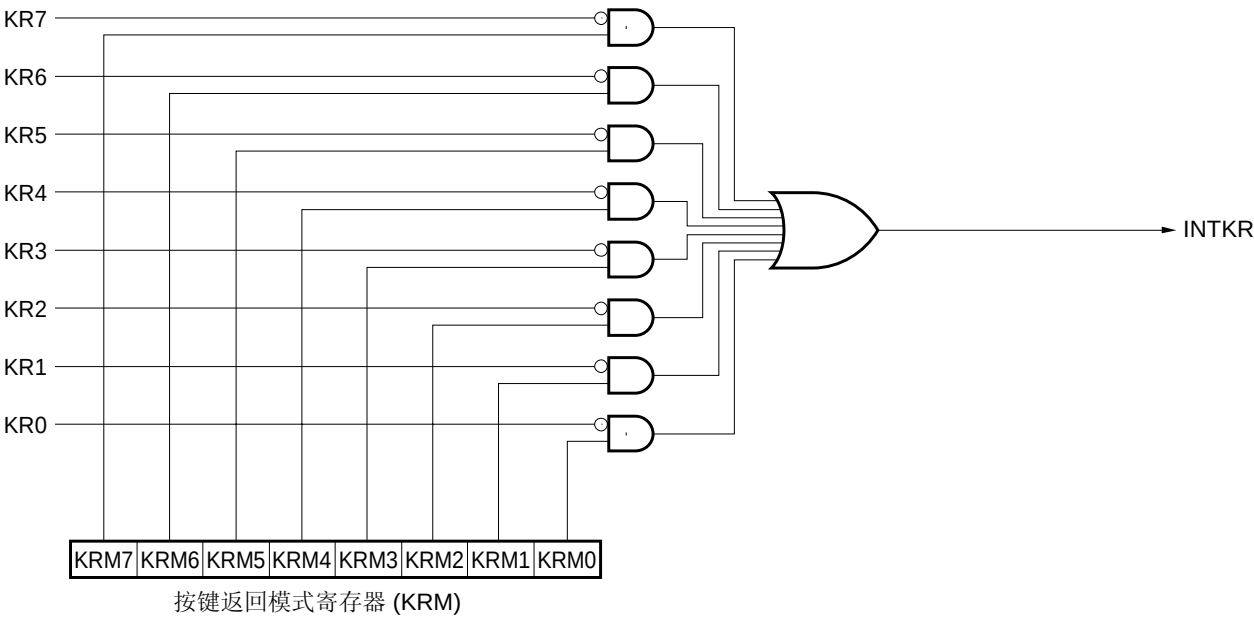
按键中断包含以下硬件：

表 17-2. 按键中断的配置

项目	配置
控制寄存器	按键返回模式寄存器（KRM）

备注 KR0 至 KR5：78K0R/KC3-L
 KR0 至 KR7：78K0R/KD3-L 和 78K0R/KE3-L

图 17-1. 按键中断的框图



备注 KR0 至 KR5: 78K0R/KC3-L
 KR0 至 KR7: 78K0R/KD3-L 和 78K0R/KE3-L

17.3 控制按键中断的寄存器

(1) 按键返回模式寄存器 (KRM)

该寄存器用 KR0 至 KR7 信号分别控制 KRM0 至 KRM7 位。
可通过 1 位或 8 位的存储操作指令设置 KRM。
产生的复位信号将该寄存器清除为 00H。

图 17-2. 按键返回模式寄存器 (KRM) 的格式

地址： FFF37H 复位后： 00H R/W

符号	7	6	5	4	3	2	1	0
KRM	KRM7	KRM6	KRM5	KRM4	KRM3	KRM2	KRM1	KRM0

KRMn	按键中断模式控制
0	不检测按键中断信号
1	检测按键中断信号

- 注意事项
- <R>
1. 如果所使用的 KRM0 至 KRM7 位中的任意一位被设置为 1，则需将相应的上拉电阻寄存器 7 (PU7) 的位 0 至 7 (PU70 至 PU77) 设置为 1。
2. 当低电平正输入到按键中断输入引脚时，如果此时 KRM 寄存器的目标位被设置，则会产生中断。若要忽略该中断，则需用中断屏蔽标志禁止中断服务后，再设置 KRM 寄存器。然后，清除中断请求标志，等待按键中断输入低电平 (250 ns 或更长) 后允许中断服务。
3. 按键中断模式中未使用的位可作正常端口使用。
- 备注
1. n = 0 至 7
2. KR0 至 KR5: 78K0R/KC3-L
- KR0 至 KR7: 78K0R/KD3-L 和 78K0R/KE3-L

18.1 待机功能和配置

18.1.1 待机功能

待机功能减小系统的操作电流，以下两种模式可用。

(1) HALT 模式

执行 HALT 指令设置 HALT 模式。在 HALT 模式中，CPU 操作时钟停止工作。如果在设置 HALT 模式之前，高速系统时钟振荡器、内部高速振荡器、20 MHz 内部高速振荡器或副系统时钟振荡器处于工作状态，则持续各个时钟振荡。在该模式中，操作电流没有减少到 STOP 模式中的值，而 HALT 模式对基于中断请求产生而立即重新开始的操作和实现间歇操作频率有效。

(2) STOP 模式

执行 STOP 指令设置 STOP 模式。在 STOP 模式中，高速系统时钟振荡器和内部高速振荡器停止工作，停止了整个系统，从而相当大的减少了 CPU 的工作电流。

由于该模式可由中断请求清除，因此它允许待执行间歇操作。但是，当选定 X1 时钟时，由于在 STOP 模式解除后到确保振荡稳定时间需要一段等待时间，所以若要在中断请求产生时立即开始处理则需选择 HALT 模式。

在两种模式的任意一种，都保持设置待机模式前的寄存器的所有内容、标志和数据内存。保持 I/O 端口输出锁存和输出缓冲器状态。

- 注意事项**
1. 仅当 CPU 工作于主系统时钟时，STOP 模式可用。当 CPU 工作于副系统时钟时，不能设置 STOP 模式。当 CPU 工作于主系统时钟或工作于副系统时钟时，STOP 模式都可用。
 2. 当转换到 STOP 模式时，在执行 STOP 指令前确保停止用主系统时钟工作的外设硬件操作。
 3. 当使用待机功能时，对于 A/D 转换器的工作电流衰减建议按以下顺序：首先将 A/D 转换器模式寄存器（ADM）的位 7（ADCS）和位 0（ADCE）清除为 0 以停止 A/D 转换操作，然后执行 STOP 指令。
 4. 当使用待机功能时，对于比较器的工作电流衰减建议按以下顺序：首先将比较器 n 控制寄存器（CnCTL）的位 7（CnEN）和比较器 n 内部参考电压选择寄存器的位 7（CnVRE）清除为 0，然后执行 STOP 指令。
 5. 当使用待机功能时，对于可编程增益放大器的工作电流衰减建议按以下顺序：首先将可编程增益放大器控制寄存器（OAM）的位 7（OAEN）清除为 0 以停止可编程增益放大器的操作，然后执行 STOP 指令。

备注 n = 0, 1

注意事项 6. 可以由选项字节选择在 **HALT** 或 **STOP** 模式中是继续还是停止内部低速振荡器振荡。详情参见第二十三章 选项字节。

7. 当 **CPU** 工作于 **20 MHz** 内部高速振荡时钟时，不能执行 **STOP** 指令。确保在转换为内部高速振荡时钟操作后执行 **STOP** 指令。

18.1.2 控制待机功能的寄存器

待机功能由以下两种寄存器来控制：

- 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**）

备注 关于这些启动、停止或选择时钟的寄存器，参见第五章 时钟发生器。

(1) 振荡稳定时间计数器状态寄存器 (OSTC)

这是指示 X1 时钟振荡稳定时间计数器的计数状态的寄存器。

可以在下列情况中检测 X1 时钟振荡时间。

- 当内部高速振荡时钟或副系统时钟正用作 CPU 时钟时，若 X1 时钟开始振荡。
- 若进入 STOP 模式，然后解除，而内部高速振荡时钟正用作 CPU 时钟且随 X1 时钟振荡时。

可通过 1 位或 8 位的存储操作指令读取 OSTC。

复位解除时（通过 RESET 输入，POC，LVI，WDT 和执行非法指令而复位），STOP 指令和 MSTOP（CSC 寄存器的位 7）= 1 将该寄存器清除为 00H。

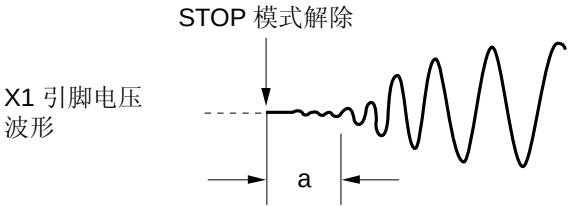
图 18-1. 振荡稳定时间计数器状态寄存器 (OSTC) 的格式

地址：FFFA2H 复位后：00H R

符号	7	6	5	4	3	2	1	0
OSTC	MOST8	MOST9	MOST10	MOST11	MOST13	MOST15	MOST17	MOST18

MOST8	MOST9	MOST10	MOST11	MOST13	MOST15	MOST17	MOST18	振荡稳定时间状态		
									$f_x = 10 \text{ MHz}$	$f_x = 20 \text{ MHz}$
0	0	0	0	0	0	0	0	$2^9/f_x \text{ max.}$	25.6 μs max.	12.8 μs max.
1	0	0	0	0	0	0	0	$2^9/f_x \text{ min.}$	25.6 μs min.	12.8 μs min.
1	1	0	0	0	0	0	0	$2^9/f_x \text{ min.}$	51.2 μs min.	25.6 μs min.
1	1	1	0	0	0	0	0	$2^{10}/f_x \text{ min.}$	102.4 μs min.	51.2 μs min.
1	1	1	1	0	0	0	0	$2^{11}/f_x \text{ min.}$	204.8 μs min.	102.4 μs min.
1	1	1	1	1	0	0	0	$2^{13}/f_x \text{ min.}$	819.2 μs min.	409.6 μs min.
1	1	1	1	1	1	0	0	$2^{15}/f_x \text{ min.}$	3.27 ms min.	1.64 ms min.
1	1	1	1	1	1	1	0	$2^{17}/f_x \text{ min.}$	13.11 ms min.	6.55 ms min.
1	1	1	1	1	1	1	1	$2^{18}/f_x \text{ min.}$	26.21 ms min.	13.11 ms min.

- 注意事项
1. 以上时间过后，这些位从 MOST8 开始依次设置为 1 并保持为 1。
 2. 振荡稳定时间计数器向上计数至 OSTS 所设置的振荡稳定时间。若进入 STOP 模式，然后解除，而内部高速振荡时钟正用作 CPU 时钟时，按以下设置振荡稳定时间。
 - 期望的 OSTC 振荡稳定时间 $\leq$ OSTS 设置的振荡稳定时间因此，需注意的是，STOP 模式解除后，只有达到 OSTS 设置的振荡稳定时间的状态可设置到 OSTC。
 3. X1 时钟振荡稳定等待时间不包括直到时钟振荡开始的时间（下图中的“a”）。



备注 f_x : X1 时钟振荡频率

(2) 振荡稳定时间选择寄存器 (OSTS)

该寄存器用于选择解除 STOP 模式时的 X1 时钟振荡稳定等待时间。

X1 时钟选作为 CPU 时钟时，在 STOP 模式解除后，操作会等待一段时间，该时间由 OSTS 设置。

内部高速振荡时钟选作为 CPU 时钟时，由 OSTC 确认在 STOP 模式解除后所需的振荡稳定时间已流逝。直到用 OSTC 设定的时间都可以检测到振荡稳定时间。

用 8 位存储操作指令设置 n。

产生的复位信号将寄存器设置为 07H。

图 18-2. 振荡稳定时间选择寄存器 (OSTS) 的格式

地址: FFFA3H 复位后: 07H 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		
				$f_x = 10\text{ MHz}$	$f_x = 20\text{ MHz}$
0	0	0	$2^8/f_x$	25.6 μs	禁止设置
0	0	1	$2^9/f_x$	51.2 μs	25.6 μs
0	1	0	$2^{10}/f_x$	102.4 μs	51.2 μs
0	1	1	$2^{11}/f_x$	204.8 μs	102.4 μs
1	0	0	$2^{13}/f_x$	819.2 μs	409.6 μs
1	0	1	$2^{15}/f_x$	3.27 ms	1.64 ms
1	1	0	$2^{17}/f_x$	13.11 ms	6.55 ms
1	1	1	$2^{18}/f_x$	26.21 ms	13.11 ms

- 注意事项
1. 当 X1 时钟用作 CPU 时钟时若要设置 STOP 模式，则需在执行 STOP 指令前设置 OSTS。

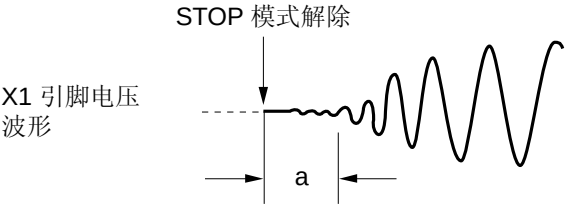
2. 禁止将振荡稳定时间设置为 20 μs 或更短的时间。

3. 在改变 OSTS 寄存器设置前，确认 OSTS 寄存器的计数操作已完成。

4. 在 X1 时钟振荡稳定时间期间禁止改变 OSTS 寄存器的值。

5. 振荡稳定时间计数器向上计数至 OSTS 所设置的振荡稳定时间。若进入 STOP 模式，然后解除，而内部高速振荡时钟正用作 CPU 时钟时，按以下设置振荡稳定时间。
 - 期望的 OSTC 振荡稳定时间 $\leq$ OSTS 设置的振荡稳定时间因此，需注意的是，STOP 模式解除后，只有达到 OSTS 设置的振荡稳定时间的状态可设置到 OSTC。

6. X1 时钟振荡稳定等待时间不包括直到时钟振荡开始的时间（下图中的“a”）。



备注 f_x : X1 时钟振荡频率

18.2 待机功能操作

18.2.1 HALT 模式

(1) HALT 模式

通过执行 HALT 指令设置 HALT 模式。无论 CPU 之前的设置是高速系统时钟、内部高速振荡时钟、20MHz 内部高速振荡时钟或是副系统时钟，都可设置 HALT 模式。

HALT 模式中的操作状态如下所示：

表 18-1. HALT 模式下的操作状态 (1/2)

HALT 模式设置 项目			当执行 HALT 指令而 CPU 工作于主系统时钟时		
			CPU 工作于内部高速振荡时钟（f _{IH} ）或 20 MHz 的内部高速振荡时钟（f _{IH20} ）时	CPU 工作于 X1 时钟（f _x ）时	CPU 工作于外部主系统时钟（f _{EX} ）时
系统时钟			供应至 CPU 的时钟停止		
主系统时钟	f _{IH} , f _{IH20}	继续工作（不能被停止）	HALT 模式设置前的状态被保持。		
			HALT 模式设置前的状态被保持。	继续工作（不能被停止）	不能工作
				不能工作	继续工作（不能被停止）
	副系统时钟	f _{XT}	HALT 模式设置前的状态被保持。	HALT 模式设置前的状态被保持。	
f _{IL}		由选项字节（000C0H）的位 0（WDSTBYON）和 4（WTON）设置 ● WTON = 0： 停止 ● WTON = 1 和 WDSTBYON = 1： 振荡 ● WTON = 1 和 WDSTBYON = 0： 停止			
CPU			操作停止		
Flash 存储器			操作停止		
RAM			该值被保持。		
端口（锁存）			HALT 模式设置前的状态被保持。		
定时器阵列部件 TAUS			可运行		
实时计数器（RTC）					
看门狗定时器			由选项字节（000C0H）的位 0（WDSTBYON）和 4（WTON）设置 ● WTON = 0： 停止 ● WTON = 1 和 WDSTBYON = 1： 操作 ● WTON = 1 和 WDSTBYON = 0： 停止		
时钟输出/蜂鸣器输出 ^注			可运行		
A/D 转换器					
可编程增益放大器					
比较器					
串行阵列部件（SAU）					
串行接口（IICA） ^注					
乘法器/除法器					
DMA 控制器					
上电清零功能					
低电压检测功能					
外部中断					
按键中断功能					

注 78K0R/KC3-L 的 44-引脚产品未配置此项。

备注 f_{IH} : 内部高速振荡时钟
 f_{IH20} : 20 MHz 内部高速振荡时钟
 f_x : X1 时钟
 f_{EX} : 外部主系统时钟
 f_{XT} : XT1 时钟
 f_{IL} : 内部低速振荡时钟

表 18-1. HALT 模式下的操作状态 (2/2)

HALT 模式设置		当执行 HALT 指令而 CPU 工作于副系统时钟时
项目		CPU 工作于 XT1 时钟（f _{XT} ）时
系统时钟		供应至 CPU 的时钟停止
主系统时钟	f _{IH} ,	HALT 模式设置前的状态被保持。
	f _{IH20}	
	f _X	
	f _{EX}	
副系统时钟	f _{XT}	继续工作（不能被停止）
f _{IL}		由选项字节（000C0H）的位 0（WDSTBYON）和 4（WTON）设置 • WTON = 0: 停止 • WTON = 1 和 WDSTBYON = 1: 振荡 • WTON = 1 和 WDSTBYON = 0: 停止
CPU		操作停止
Flash 存储器		工作停止（处于低功耗模式中的等待状态）
RAM		该值被保持。
端口（锁存）		HALT 模式设置前的状态被保持。
定时器阵列部件 TAUS		可运行
实时计数器（RTC）		
看门狗定时器		由选项字节（000C0H）的位 0（WDSTBYON）和 4（WTON）设置 • WTON = 0: 停止 • WTON = 1 和 WDSTBYON = 1: 工作 • WTON = 1 和 WDSTBYON = 0: 停止
时钟输出/蜂鸣器输出 [※]		可运行
A/D 转换器		不能工作
可编程增益放大器		可运行
比较器		
串行阵列部件（SAU）		
串行阵列部件（IICA） ^註		不能工作
乘法器/除法器		可运行
DMA 控制器		
上电清零功能		
低电压检测功能		
外部中断		
按键中断功能		

注 78K0R/KC3-L 的 44-引脚产品未配置此项。

备注 f_{IH} : 内部高速振荡时钟
 f_{IH20} : 20 MHz 内部高速振荡时钟
 f_X : X1 时钟
 f_{EX} : 外部主系统时钟
 f_{XT} : XT1 时钟
 f_{IL} : 内部低速振荡时钟

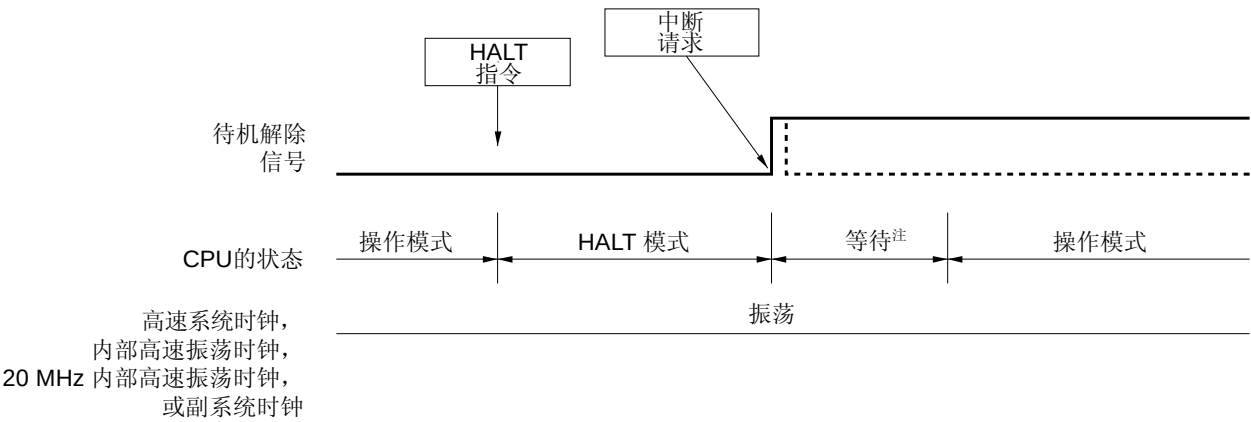
(2) HALT 模式解除

可由以下两种原因解除 HALT 模式。

(a) 由不可屏蔽中断请求解除

当产生不可屏蔽中断请求时，HALT 模式被解除。如果允许中断响应，则执行向量中断服务。如果禁止中断响应，则执行下一个地址指令。

图 18-3. 由中断请求产生解除 HALT 模式



- <R> 注 等待时间如下：
- 当执行向量中断服务时：
 - 使用主系统时钟时：10 至 12 个时钟
 - 当使用副系统时钟时：8 至 10 个时钟
 - 当不执行向量中断服务时：
 - 使用主系统时钟时：5 或 6 个时钟
 - 当使用副系统时钟时：3 或 4 个时钟

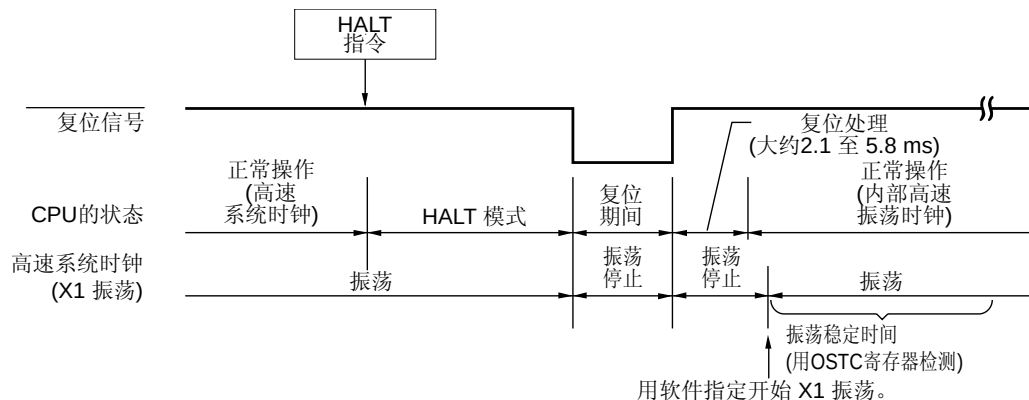
备注 虚线表示已经解除待机模式的中断请求被响应的情况。

(b) 由复位信号产生解除

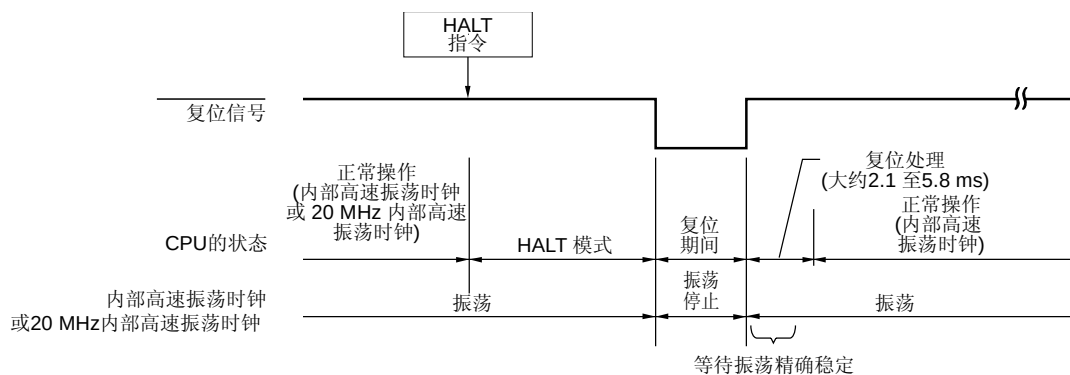
当复位信号产生时，HALT 模式被解除，以正常复位操作为例，在转移到复位向量地址后程序被执行。

图 18-4. 由复位解除 HALT 模式

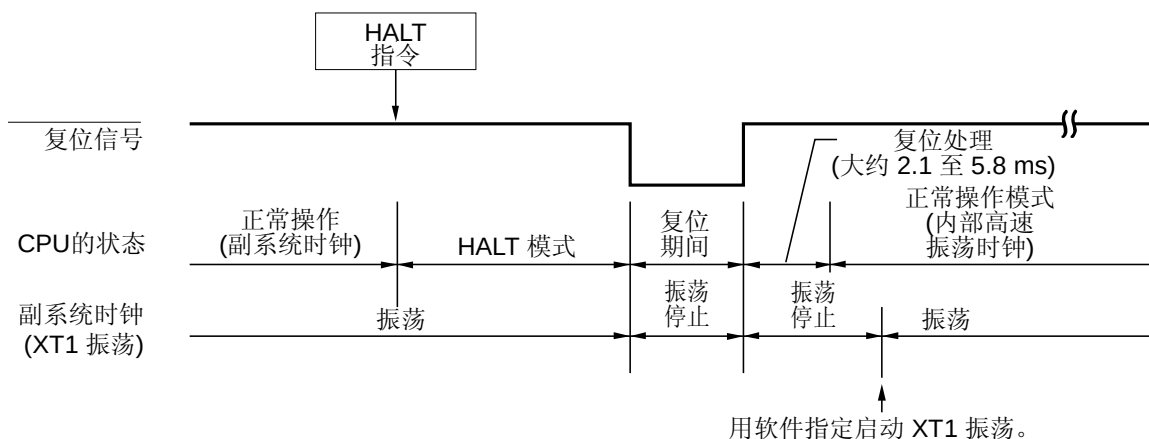
(1) 当高速系统时钟用作 CPU 时钟时



(2) 当内部高速振荡时钟或 20 MHz 的内部高速振荡时钟用作 CPU 时钟时



(3) 当副系统时钟用作 CPU 时钟时



备注 fx: X1 时钟振荡频率

18.2.2 STOP 模式

(1) STOP 模式设置和操作状态

通过执行 STOP 指令设置 STOP 模式，只有设置前的 CPU 时钟为内部高速振荡时钟、X1 时钟或外部主系统时钟时才可进行设置。

- 注意事项**
1. 因为中断请求信号用于清除待机模式，所以若存在带有一个中断请求标志设置和中断屏蔽标志复位的中断源，如果对其进行设置则待机模式被立即清除。因此，执行 STOP 指令后 STOP 模式立即复位至 HALT 模式，一旦振荡稳定时间选择寄存器（OSTS）设置的等待时间过后，系统立即返回到操作模式。
 2. 当 CPU 工作于 20 MHz 内部高速振荡时钟时，不能执行 STOP 指令。确保在转换为内部高速振荡时钟操作后执行 STOP 指令。

STOP 模式中的操作状态如下所示：

表 18-2. STOP 模式下的操作状态

STOP 模式设置 项目			当执行 STOP 指令而 CPU 工作于主系统时钟时		
			当 CPU 工作于内部高速振荡 时钟 (f _{IH}) 时	CPU 工作于 X1 时钟 (f _X) 时	CPU 工作于外部主系统时钟 (f _{EX}) 时
系统时钟			供应至 CPU 的时钟停止		
	主系统时钟	f _{IH}	停止		
		f _X			
		f _{EX}			
	副系统时钟	f _{XT}	STOP 模式设置前的状态被保持。		
	f _{IL}		由选项字节 (000C0H) 的位 0 (WDSTBYON) 和 4 (WTON) 设置 ● WTON = 0: 停止 ● WTON = 1 和 WDSTBYON = 1: 振荡 ● WTON = 1 和 WDSTBYON = 0: 停止		
CPU			操作停止		
Flash 存储器			操作停止		
RAM			该值被保持		
端口 (锁存)			STOP 模式设置前的状态被保持。		
定时器阵列部件 TAUS			禁止操作		
实时计数器 (RTC)			可运行		
看门狗定时器			由选项字节 (000C0H) 的位 0 (WDSTBYON) 和 4 (WTON) 设置 ● WTON = 0: 停止 ● WTON = 1 和 WDSTBYON = 1: 操作 ● WTON = 1 和 WDSTBYON = 0: 停止		
时钟输出/蜂鸣器输出 ^註			仅当副系统时钟选作计数时钟时可操作		
A/D 转换器			禁止操作		
可编程增益放大器					
比较器					
串行阵列部件 (SAU)					
串行接口 (IICA) ^註			可被地址匹配唤醒		
乘法器/除法器			禁止操作		
DMA 控制器					
上电清除功能			可运行		
低电压检测功能					
外部中断					
按键中断功能					

注 78K0R/KC3-L 的 44-引脚产品未配置此项。

备注 f_{IH} : 内部高速振荡时钟
 f_X : X1 时钟
 f_{EX} : 外部主系统时钟
 f_{XT} : XT1 时钟
 f_{IL} : 内部低速振荡时钟

- 注意事项
1. 若要使用在 STOP 模式中停止操作的外设硬件，即时钟在 STOP 模式下停止振荡的硬件，则需在 STOP 模式解除后，重新启动外设硬件。
 2. 若要在 STOP 模式下停止内部低速振荡时钟，则需用选项字节停止在 HALT/STOP 模式下的看门狗定时器操作（000C0H 的位 0（WDSTBYON）= 0），而后执行 STOP 指令。
 3. 当 CPU 工作于高速系统时钟（X1 振荡）时，若要缩短 STOP 模式解除后的振荡稳定时间，需要在下一个 STOP 指令执行前暂时将 CPU 时钟转换成内部高速振荡时钟。STOP 模式解除后，在将 CPU 时钟从内部高速振荡时钟改变为高速系统时钟（X1 振荡）前，用振荡稳定时间计数器状态寄存器（OSTC）检查振荡稳定时间。
 4. 当 CPU 工作于 20 MHz 内部高速振荡时钟时，不能执行 STOP 指令。确保在转换为内部高速振荡时钟操作后执行 STOP 指令。

（2）STOP 模式解除

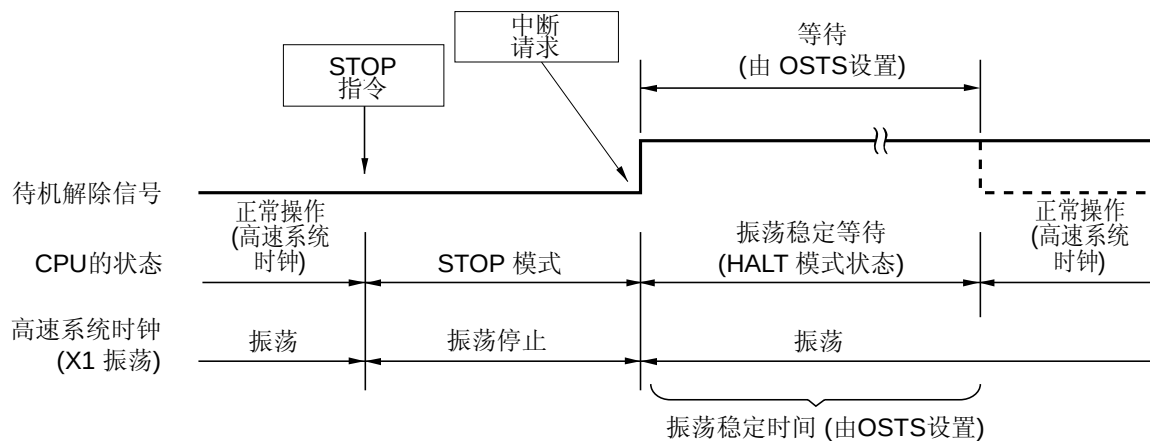
可由以下两种原因解除 STOP 模式。

（a）由不可屏蔽中断请求解除

当产生不可屏蔽中断请求时，STOP 模式被解除。振荡稳定时间过后，如果允许中断响应，则执行向量中断服务。如果禁止中断响应，则执行下一个地址指令。

图 18-5. 由中断请求产生解除 STOP 模式（1/2）

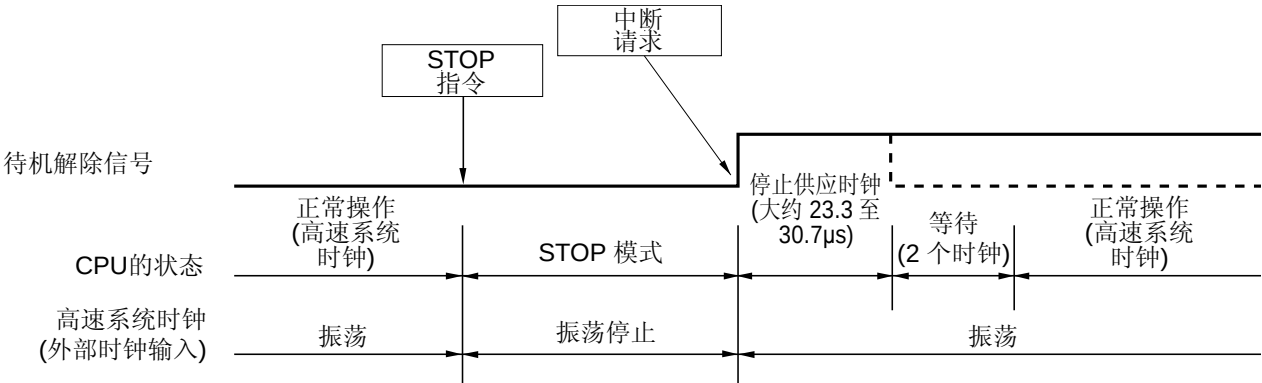
（1）当高速系统时钟（X1 振荡）用作 CPU 时钟时



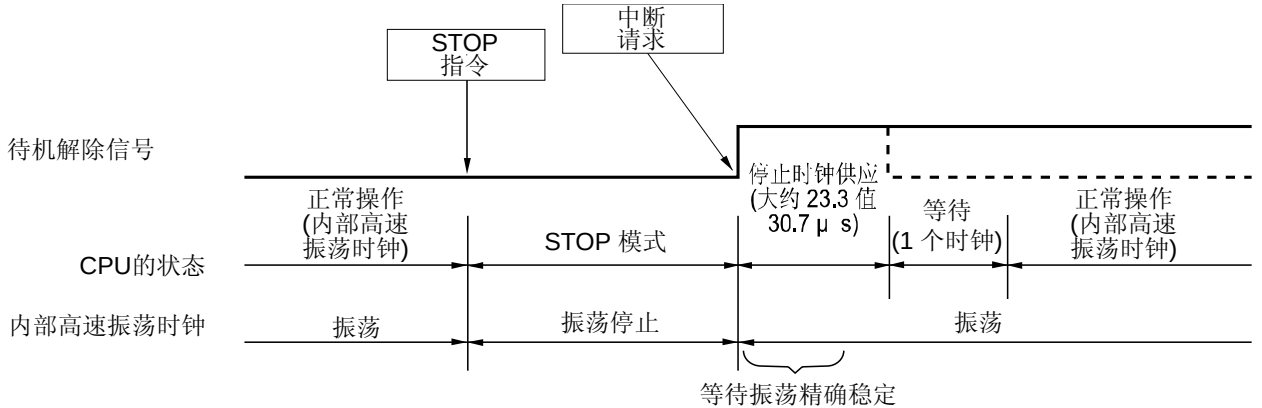
备注 虚线表示已经解除待机模式的中断请求被响应的情况。

图 18-5. 由中断请求产生解除 STOP 模式 (2/2)

(2) 当高速系统时钟（外部时钟输入）用作 CPU 时钟时



(3) 当高速振荡时钟用作 CPU 时钟时



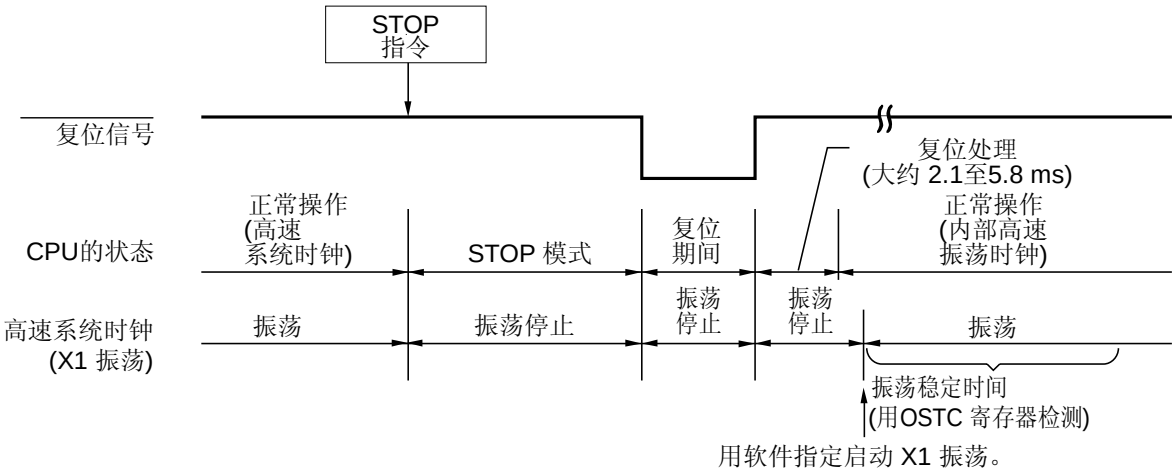
备注 虚线表示已经解除待机模式的中断请求被响应的情况。

(b) 由复位信号产生解除

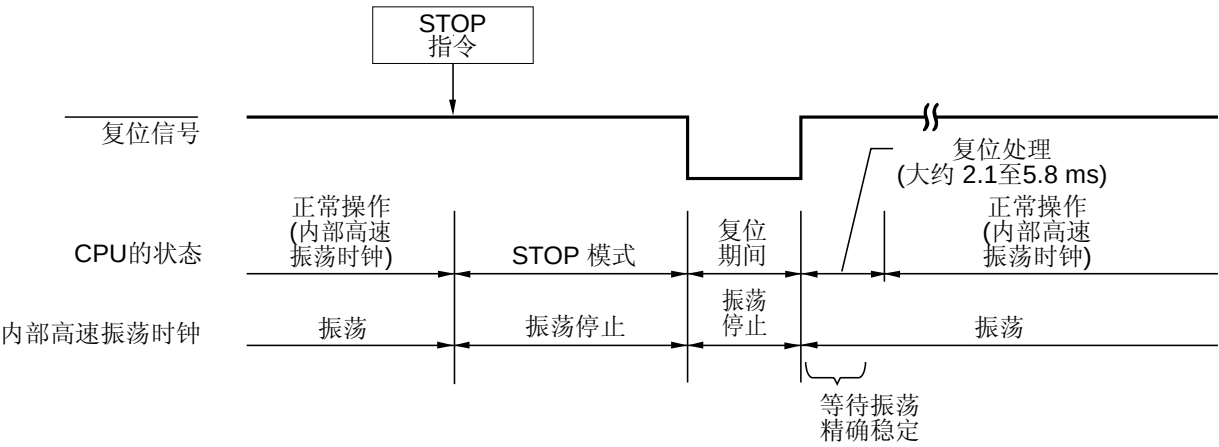
当复位信号产生时，STOP 模式被解除，以正常复位操作为例，在转移到复位向量地址后程序被执行。

图 18-6. 由复位解除 STOP 模式

(1) 当高速系统时钟用作 CPU 时钟时



(2) 当高速振荡时钟用作 CPU 时钟时



备注 fx: X1 时钟振荡频率

第十九章 复位功能

以下五种操作可用来产生复位信号。

- (1) 通过 $\overline{\text{RESET}}$ 引脚输入外部复位
- (2) 通过看门狗定时器程序循环检测进行内部复位
- (3) 通过比较供应电压与上电清除 (POC) 电路的检测电压进行内部复位
- (4) 通过比较电压检测器 (LVI) 或从外部输入的输入电压 (EXLVI) 与检测电压进行内部复位
- (5) 通过执行非法指令^注进行内部复位

产生复位信号时，外部复位起始程序和内部复位起始程序从地址 0000H 和 0001H 开始执行。

当低电平输入到 $\overline{\text{RESET}}$ 引脚时，或看门狗定时器溢出时，或通过 POC 和 LVI 电路电压检测或执行非法指令^注，复位有效，而硬件的各项设置为表 19-1 和 19-2 所显示的状态。在复位信号产生期间或刚好在复位解除后的振荡稳定时间期间，除低电平输出的 P140 引脚外，其它各引脚为高阻状态。

当低电平输入到 $\overline{\text{RESET}}$ 引脚时，该器件复位。当高电平输入到 $\overline{\text{RESET}}$ 引脚时，器件从复位状态解除，在复位完成后，开始以内部高速振荡时钟执行程序。通过看门狗定时器复位可自动解除，在复位完成后，开始以内部高速振荡时钟（参见图 19-2 至 19-4）执行程序复位后当 $V_{DD} \geq V_{POR}$ 或 $V_{DD} \geq V_{LVI}$ 时，通过 POC 和 LVI 电路供电电源检测，复位可自动解除，在复位完成后，开始以内部高速振荡时钟（参见第二十章 上电清除电路 和 第二十一章 电压检测器）执行程序。

注 该非法指令在执行指令代码 FFH 时产生。

在线仿真器或片上调试仿真器不能产生通过执行非法指令的复位。

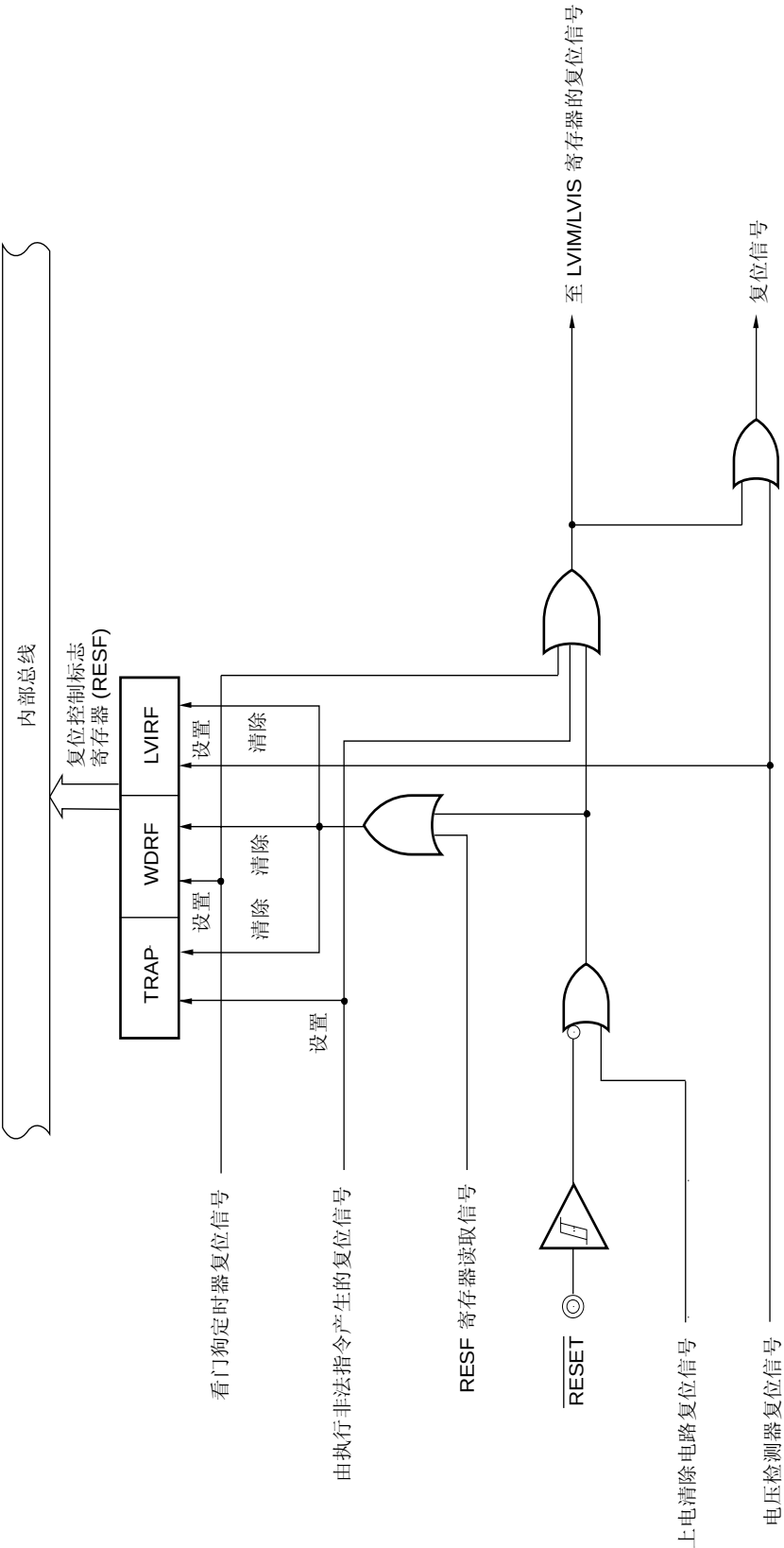
注意事项 1. 对于外部复位，可将一个 10 μs 或更长时间的的低电平输入到 $\overline{\text{RESET}}$ 引脚。

（若要执行基于上电的外部复位，供应电压在操作范围 ($V_{DD} \geq 1.8 \text{ V}$) 期间，必须持续输入一个至少 10 μs 的低电平。）

2. 复位输入期间，X1 时钟、XT1 时钟、内部高速振荡时钟以及内部低速振荡时钟停止振荡。外部主系统时钟输入变得无效。
3. 当通过复位解除 STOP 模式时，在复位输入期间，保持 STOP 模式中 RAM 的内容。
4. 当复位有效时，因为各 SFR 和 2nd SFR 被初始化，所以端口引脚 P140 被设置为低电平输出而其它端口引脚变成高阻状态。

备注 V_{POR} : POC 供应电源上升检测电压

图 19-1. 复位功能的框图



注意事项 一个 LVI 电路的内部复位不能使得 LVI 电路（自身）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 19-2. RESET 输入引起的复位时序

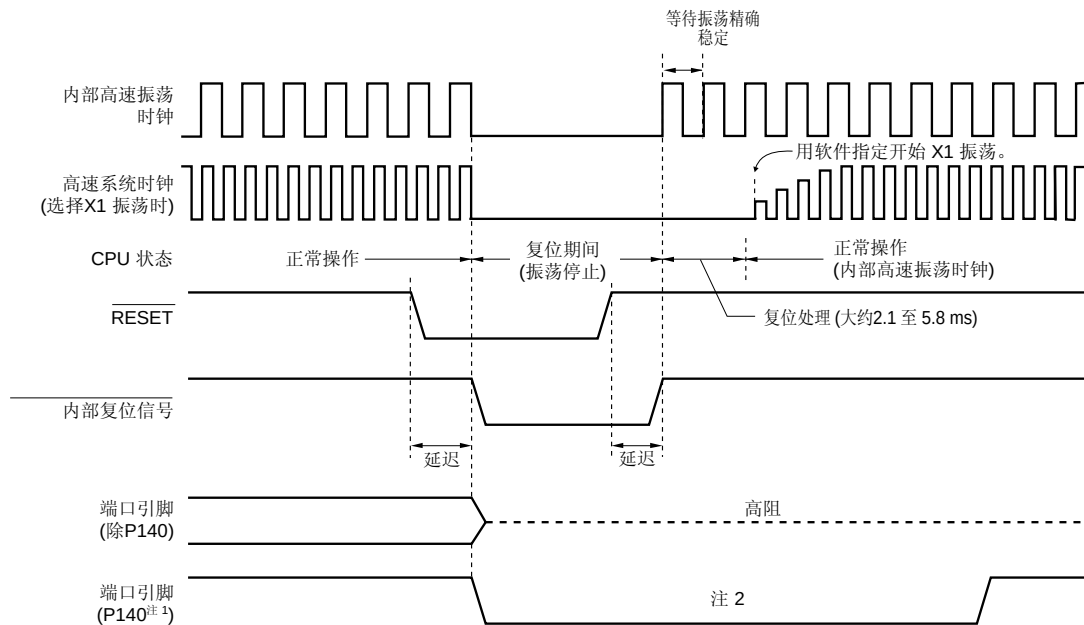
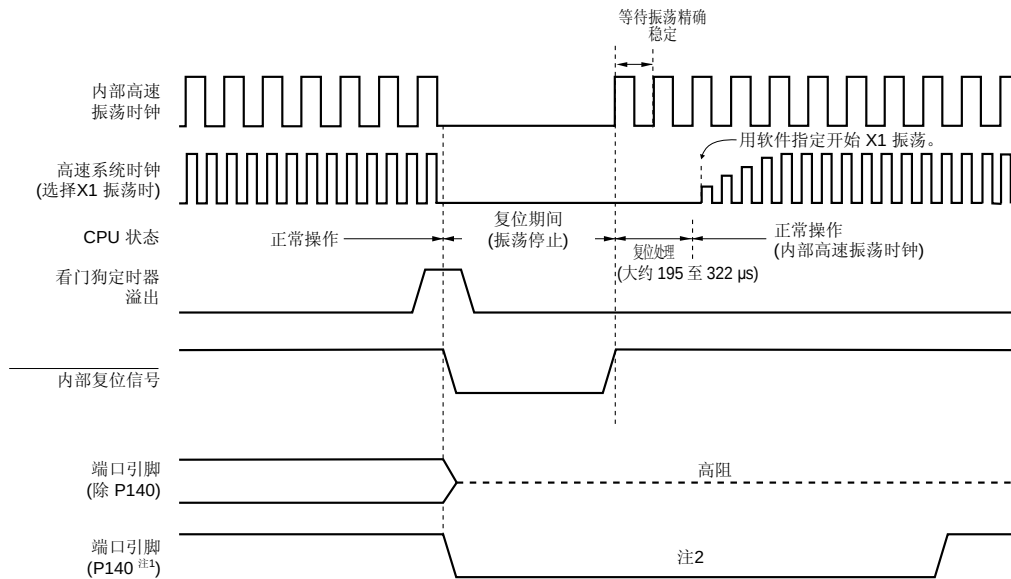


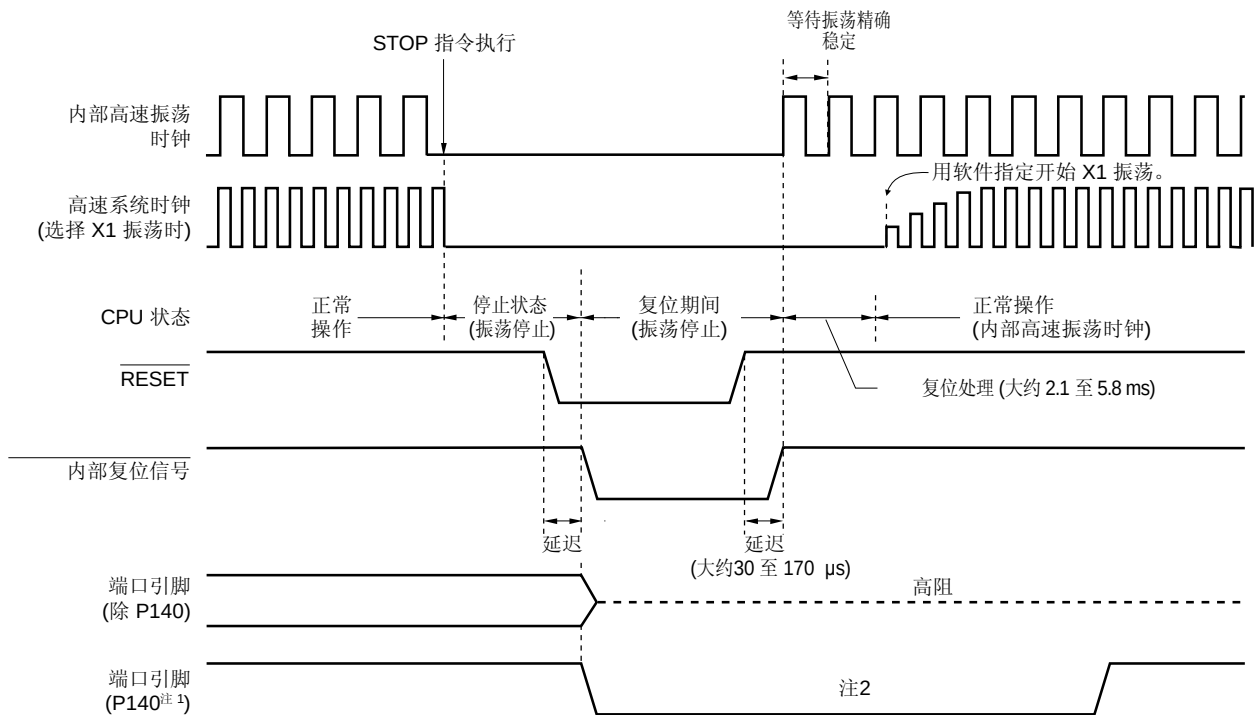
图 19-3. 由看门狗定时器溢出引起的复位时序



- 注
1. 78K0R/KC3-L 的 44-引脚产品未配置 P140。
 2. 在复位有效前当 P140 设置为高电平输出时，P140 的输出信号作为一个输入到外部器件的复位信号进行伪输出，这是因为复位有效时 P140 作为低电平输出。若要解除外部器件的复位信号，需用软件将 P140 设置为高电平。

注意事项 看门狗定时器内部复位不能将看门狗定时器本身复位。

图 19-4. STOP 模式中由 RESETS 输入引起的复位时序



- 注
1. 78K0R/KC3-L 的 44-引脚产品未配置 P140。
 2. 用软件将 P140 设置为高电平输出。在复位有效前当 P140 设置为高电平输出时，P140 的输出信号作为一个输入到外部器件的复位信号进行伪输出，这是因为复位有效时 P140 作为低电平输出。若要解除外部器件的复位信号，需用软件将 P140 设置为高电平。

备注 关于上电清除电路和电压寄存器的复位时序，参见第二十章 上电清除电路 和 第二十一章 电压检测器。

表 19-1. 复位期间的操作状态

项目			复位期间
系统时钟			供应至 CPU 的时钟停止。
	主系统时钟	f _{IH}	操作停止
		f _X	操作停止（X1 和 X2 引脚为输入端口模式）
		f _{EX}	时钟输入无效（引脚为输入端口模式）
	副系统时钟	f _{XT}	操作停止（XT1 和 XT2 引脚为输入端口模式）
	f _{IL}		操作停止
CPU			
Flash 存储器			
RAM			
端口（锁存）			
定时器阵列部件 TAUS			
实时计数器（RTC）			
看门狗定时器			
时钟输出/蜂鸣器输出 ^註			
A/D 转换器			
可编程增益放大器			
比较器			
串行阵列部件（SAU）			
串行接口（IICA） ^註			
乘法器/除法器			
DMA 控制器			
上电清除功能			
低电压检测功能			
外部中断			
按键中断功能			

注 78K0R/KC3-L 的 44-引脚产品未配置此项。

备注

f_{IH}: 内部高速振荡时钟

f_X: X1 振荡时钟

f_{EX}: 外部主系统时钟

f_{XT}: XT1 振荡时钟

f_{IL}: 内部低速振荡时钟

表 19-2. 应答复位后的硬件状态 (1/4)

硬件		应答复位后 ^{注1}
程序计数器 (PC)		复位向量表的内容 (0000H, 0001H) 被设置
栈指针 (SP)		不确定
程序状态字 (PSW)		06H
RAM	数据内存	不确定 ^{注2}
	通用寄存器	不确定 ^{注2}
处理器模式控制寄存器 (PMC)		00H
端口寄存器 (P0 至 P8, P12, P14, P15) (输出锁存)		00H
端口模式寄存器	PM0 至 PM8, PM12, PM15	FFH
	PM14	FEH
端口输入模式寄存器 3, 7, 8 (PIM3, PIM7, PIM8)		00H
端口输出模式寄存器 3, 7 (POM3, POM7)		00H
上拉电阻选项寄存器 (PU0, PU1, PU3 至 PU5, PU7, PU12, PU14)		00H
时钟操作模式控制寄存器 (CMC)		00H
时钟操作状态控制寄存器 (CSC)		C0H
系统时钟控制寄存器 (CKC)		09H
20 MHz 内部高速振荡控制寄存器 (DSCCTL)		00H
振荡稳定时间计数器状态寄存器 (OSTC)		00H
振荡稳定时间选择寄存器 (OSTS)		07H
噪声过滤器使能寄存器 0, 1, 2 (NFEN0, NFEN1, NFEN2)		00H
外围使能寄存器 0, 1, 2 (PER0, PER1, PER2)		00H
操作速度模式控制寄存器 (OSMC)		00H
定时器阵列部件 (TAUS)	定时器数据寄存器 00, 01, 02, 03, 04, 05, 06, 07 (TDR00, TDR01, TDR02, TDR03, TDR04, TDR05, TDR06, TDR07)	0000H
	定时器模式寄存器 00, 01, 02, 03, 04, 05, 06, 07 (TMR00, TMR01, TMR02, TMR03, TMR04, TMR05, TMR06, TMR07)	0000H
	定时器状态寄存器 00, 01, 02, 03, 04, 05, 06, 07 (TSR00, TSR01, TSR02, TSR03, TSR04, TSR05, TSR06, TSR07)	0000H
	定时器输入选择寄存器 0 (TIS0)	00H
	定时器计数器寄存器 00, 01, 02, 03, 04, 05, 06, 07 (TCR00, TCR01, TCR02, TCR03, TCR04, TCR05, TCR06, TCR07)	FFFFH
	定时器通道使能状态寄存器 0 (TE0)	0000H
	定时器通道启动触发器寄存器 0 (TS0)	0000H
	定时器通道停止触发器寄存器 0 (TT0)	0000H
	定时器时钟选择寄存器 0 (TPS0)	0000H
	定时器通道输出寄存器 0 (TO0)	0000H
	定时器通道输出使能寄存器 0 (TOE0)	0000H
	定时器通道输出电平寄存器 0 (TOL0)	0000H
	定时器通道输出模式寄存器 0 (TOM0)	0000H

- 注 1. 复位信号产生或振荡稳定时间等待期间, 硬件状态只有中 PC 的内容变得不确定。复位后其它所有硬件状态保持不变。
2. 在待机模式中执行复位时, 即使复位后还会保持预复位状态。

备注 根据产品安装特殊功能寄存器 (SFR)。参见 3.2.4 特殊功能寄存器 (SFR) 和 3.2.5 扩展的特殊功能寄存器 (第二 SFR: 第二 特殊功能寄存器)

表 19-2. 应答复位后的硬件状态 (2/4)

硬件		应答复位后的状态 ^{※1}
实时计数器	副-计数寄存器 (RSUBC)	0000H
	秒计数寄存器 (SEC)	00H
	分计数寄存器 (MIN)	00H
	小时计数寄存器 (HOUR)	12H
	星期计数寄存器 (WEEK)	00H
	天计数寄存器 (DAY)	01H
	月计数寄存器 (MONTH)	01H
	年计数寄存器 (YEAR)	00H
	钟表错误校正寄存器 (SUBCUD)	00H
	分报警寄存器 (ALARMWM)	00H
	小时报警寄存器 (ALARMWH)	12H
	星期报警寄存器 (ALARMWW)	00H
	控制寄存器 0 (RTCC0)	00H
	控制寄存器 1 (RTCC1)	00H
	控制寄存器 2 (RTCC2)	00H
时钟输出/蜂鸣器 输出控制器	时钟输出选择寄存器 0、1 (CKS0、CKS1)	00H
看门狗定时器	使能寄存器 (WDTE)	1AH/9AH ^{※2}
A/D 转换器	10 位 A/D 转换结果寄存器 (ADCR)	0000H
	8 位 A/D 转换结果寄存器 (ADCR)	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)	10H
串行阵列部件 (SAU)	串行数据寄存器 00, 01, 02, 03 (SDR00, SDR01, SDR02, SDR03)	0000H
	串行状态寄存器 00, 01, 02, 03 (SSR00, SSR01, SSR02, SSR03)	0000H
	串行标志清除触发寄存器 00, 01, 02, 03 (SIR00, SIR01, SIR02, SIR03)	0000H
	串行模式寄存器 00, 01, 02, 03 (SMR00, SMR01, SMR02, SMR03)	0020H
	串行通信操作设置寄存器 00, 01, 02, 03 (SCR00, SCR01, SCR02, SCR03)	0087H
	串行通道使能状态寄存器 0 (SE0)	0000H
	串行通道启动触发寄存器 0 (SS0)	0000H
	串行通道停止触发寄存器 0 (ST0)	0000H
	串行时钟选择寄存器 0 (SPS0)	0000H
	串行输出寄存器 0 (SO0)	0F0FH
	串行输出使能寄存器 0 (SOE0)	0000H
	串行输出电平寄存器 0 (SOL0)	0000H
	输入转换控制寄存器 (ISC)	00H

注 1. 复位信号产生或振荡稳定时间等待期间, 硬件状态只有中 PC 的内容变得不确定。复位后其它所有硬件状态保持不变。

2. WDTE 的复位值取决于选项字节的设置。

备注 根据产品安装特殊功能寄存器 (SFR)。参见 3.2.4 特殊功能寄存器 (SFR) 和 3.2.5 扩展的特殊功能寄存器 (第二 SFR: 第二特殊功能寄存器)

表 19-2. 应答复位后的硬件状态 (3/4)

硬件		应答复位后的状态 ^{注1}
串行接口 IICA	IICA 移位寄存器 (IICA)	00H
	IICA 状态寄存器 (IICS)	00H
	IICA 标志寄存器 (IICF)	00H
	IICA 控制寄存器 0 (IICCTL0)	00H
	IICA 控制寄存器 1 (IICCTL1)	00H
	IICA 低电平宽度设置寄存器 (IICWL)	FFH
	IICA 高电平宽度设置寄存器 (IICWH)	FFH
	从地址寄存器 (SVA)	00H
乘法器/除法器	乘法/除法数据寄存器 A (L) (MDAL)	0000H
	乘法/除法数据寄存器 A (H) (MDAH)	0000H
	乘法/除法数据寄存器 B (L) (MDBL)	0000H
	乘法/除法数据寄存器 B (H) (MDBH)	0000H
	乘法/除法数据寄存器 C (L) (MDCL)	0000H
	乘法/除法数据寄存器 C (H) (MDCH)	0000H
	乘法/除法控制寄存器 (MDUC)	00H
按键中断	按键返回模式寄存器 (KRM)	00H
复位功能	复位控制标志寄存器 (RESF)	不确定 ^{注2}
低电压检测器	低电压检测寄存器 (LVIM)。	00H ^{注3}
	低电压检测电平选择寄存器 (LVIS)。	0EH ^{注2}
电源稳压器	稳压器模式控制寄存器 (RMC)	00H
DMA 控制器	SFR 地址寄存器 0, 1 (DSA0, DSA1)	00H
	RAM 地址寄存器 0L, 0H, 1L, 1H (DRA0L, DRA0H, DRA1L, DRA1H)	00H
	字节计数寄存器 0L, 0H, 1L, 1H (DBC0L, DBC0H, DBC1L, DBC1H)	00H
	模式控制寄存器 0, 1 (DMC0, DMC1)	00H
	操作控制寄存器 0, 1 (DRC0, DRC1)	00H

- 注 1. 复位信号产生或振荡稳定时间等待期间, 硬件状态只有中 PC 的内容变得不确定。复位后其它所有硬件状态保持不变。
2. 根据复位源不同, 这些值有所不同。

复位源 寄存器		RESET 输入	由于 POC 复位	由于执行非法指令复位	由于 WDT 复位	由于 LVI 复位
RESF	TRAP 位	清零	清零	设置 (1)	保持	保持
	WDRF 位			保持	设置 (1)	保持
	LVIRF 位			保持	保持	设置 (1)
LVIS		清除 (0EH)	清除 (0EH)	清除 (0EH)	清除 (0EH)	保持

3. 该值根据复位源和选项字节的的不同而不同。

备注 根据产品安装特殊功能寄存器 (SFR)。参见 3.2.4 特殊功能寄存器 (SFR) 和 3.2.5 扩展的特殊功能寄存器 (第二 SFR: 第二 特殊功能寄存器)

表 19-2. 应答复位后的硬件状态 (4/4)

硬件		应答复位后的状态 ^注
中断	请求标志寄存器 0L, 0H, 1L, 1H, 2L (IF0L, IF0H, IF1L, IF1H, IF2L)	00H
	Mask flag 寄存器 0L, 0H, 1L, 1H, 2L (MK0L, MK0H, MK1L, MK1H, MK2L)	FFH
	优先级指定标志寄存器 00L, 00H, 01L, 01H, 02L, 10L, 10H, 11L, 11H, 12L (PR00L, PR00H, PR01L, PR01H, PR10L, PR10H, PR11L, PR11H, PR02L, PR12L)	FFH
	外部中断上升沿使能 寄存器 0 (EGP0)	00H
	外部中断下降沿使能 寄存器 0 (EGN0)	00H
可编程增益放大器	可编程增益放大器控制 寄存器 (OAM)	00H
比较器	比较器 0 控制寄存器 (C0CTL)	00H
	比较器 0 内部参考电压设置寄存器 (C0RVM)	00H
	比较器 1 控制 寄存器 (C1CTL)	00H
	比较器 1 内部参考电压设置寄存器 (C1RVM)	00H
BCD 校正电路	BCD 校正结果寄存器 (BCDAJ)	不确定

注 复位信号产生或振荡稳定时间等待期间，硬件状态只有中 PC 的内容变得不确定。复位后其它所有硬件状态保持不变。

备注 根据产品安装特殊功能寄存器 (SFR)。参见 3.2.4 特殊功能寄存器 (SFR) 和 3.2.5 扩展的特殊功能寄存器 (第二 SFR: 第二 特殊功能寄存器)

19.1 确认复位源的寄存器

78K0R/Kx3-L 中存在多种内部复位产生源。复位控制标志寄存器（RESF）由于存储已经产生复位请求的产生源。RESF 可用 8 位存储操作指令读取。

通过上电清除（POC）电路和读取 RESF 以清除 TRAP，WDRF 和 LVIRF，使得 $\overline{\text{RESET}}$ 输入复位。

图 19-5. 复位控制标志寄存器（RESF）的格式

地址：FFFA8H 复位后：不确定 R

符号	7	6	5	4	3	2	1	0
RESF	TRAP ^{注1}	不确定	不确定	WDRF ^{注1}	不确定	不确定	不确定	LVIRF ^{注1}
TRAP	由执行非法指令产生内部复位请求 ^{注2}							
0	不产生内部复位请求或 RESF 被清除。							
1	产生内部复位请求。							
WDRF	由看门狗定时器（WDT）产生内部复位请求							
0	不产生内部复位请求或 RESF 被清除。							
1	产生内部复位请求。							
LVIRF	由电压寄存器（LVI）产生内部复位请求							
0	不产生内部复位请求或 RESF 被清除。							
1	产生内部复位请求。							

- 注
1. 复位后的值根据复位源的不同而不同。
 2. 该非法指令在执行指令代码 FFH 时产生。
在线仿真器或片上调试仿真器不能产生通过执行非法指令的复位。

- 注意事项
1. 禁止用 1 位存储操作指令读取数据。
 2. 禁止只根据 RESF 寄存器 8 位数据的读取值进行判断，因为除 TRAP，WDRF 和 LVIRF 外的位变得不确定。
 3. 当使用 LVI 默认启动功能（000C1H 的位 0（LVIOFF）= 0）时，根据上电波形，LVIRF 标志有可能从开始就变成 1。

当产生复位请求时 RESF 的状态如表 19-3 所示。

表 19-3. 当产生复位请求时 RESF 的状态

复位源 标志	$\overline{\text{RESET}}$ 输入	由于 POC 复位	由于执行非法指令复位	由于 WDT 复位	由于 LVI 复位
TRAP	清零	清零	设置（1）	保持	保持
WDRF			保持	设置（1）	保持
LVIRF			保持	保持	设置（1）

第二十章 上电清零电路

20.1 上电清零电路的功能

上电清零电路（POC）具有以下功能。

- 接通电源时生成内部复位信号。
当供电电压（ V_{DD} ）超过 $1.61\text{ V} \pm 0.09\text{ V}^{\text{#}}$ 时解除复位信号。

注意事项 如果低电压检测器（LVI）由默认的选项字节设置为ON，则复位信号直到供电电压（ V_{DD} ）超过 $2.07\text{ V} \pm 0.2\text{ V}^{\text{#}}$ 时才会被解除。

- 比较供电电压（ V_{DD} ）和检测电压（ $V_{PDR} = 1.59\text{ V} \pm 0.09\text{ V}^{\text{#}}$ ），在 $V_{DD} < V_{PDR}$ 时生成内部复位信号。

注 这些是初始值，易更改。

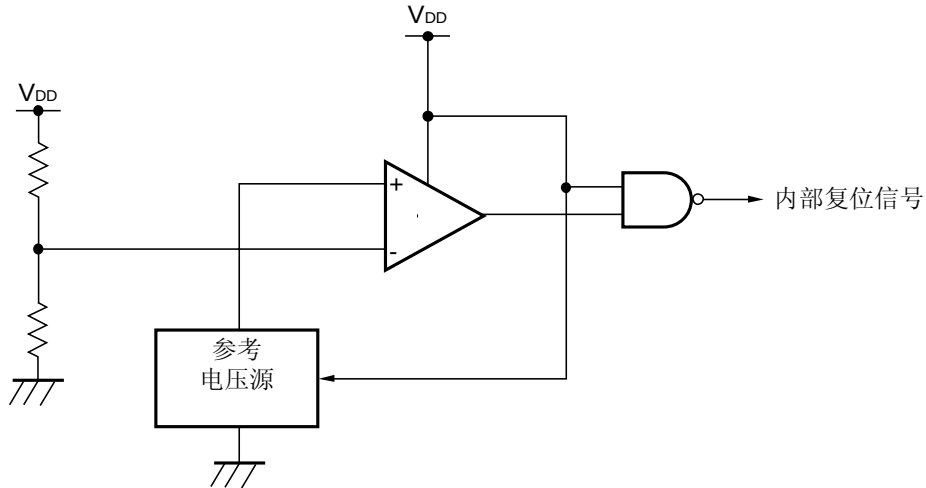
注意事项 如果在 POC 电路中生成内部复位信号，则将复位控制标志寄存器（RESF）的 TRAP，WDRF 和 LVIRF 被清零。

备注 本产品结合了多个生成内部复位信号的硬件功能。当由看门狗定时器（WDT）、低电压检测器（LVI）或执行非法指令生成内部复位信号时，表示复位源的标志在复位控制标志寄存器（RESF）中被设置。当由 WDT 或 LVI 生成内部复位信号时，不会将 RESF 清除为 00H，而将标志置为 1。
关于 RESF 细节参见第十九章 复位功能。

20.2 上电清零电路的配置

上电清零电路的框图如图 20-1。

图 20-1. 上电清零电路的框图



20.3 上电清零电路的操作

- 上电时产生内部复位信号。供应电压（ V_{DD} ）超出检测电压（ $V_{PDR} = 1.61\text{ V} \pm 0.09\text{ V}^{\#}$ ）时，复位状态解除。

注意事项 如果低电压检测器（LVI）由默认的选项字节设置为ON，则复位信号直到供电电压（ V_{DD} ）超过 $2.07\text{ V} \pm 0.2\text{ V}^{\#}$ 时才会被解除。

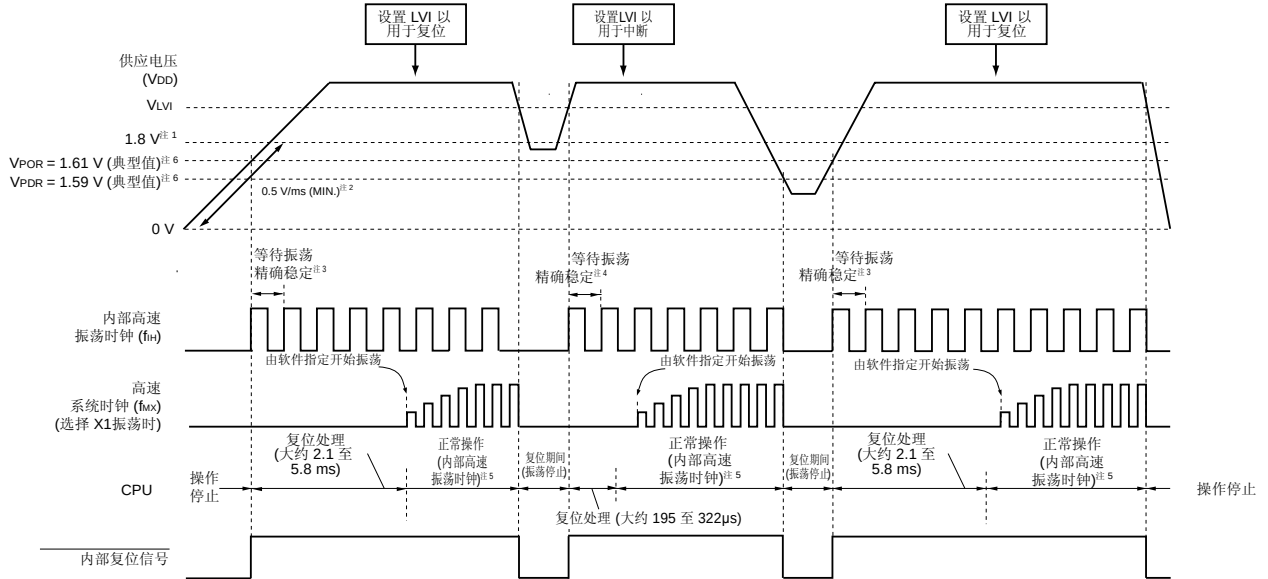
- 比较供应电压（ V_{DD} ）和检测电压（ $V_{PDR} = 1.59\text{ V} \pm 0.09\text{ V}^{\#}$ ）。当 $V_{DD} < V_{PDR}$ 时，产生内部复位信号。

注 这些是初始值，可更改。

由上电清零电路和电压检测器产生的内部复位信号的时序如下所示。

图 20-2. 由上电清零电路和电压检测器产生内部复位信号的时序 (1/2)

(1) 上电时 LVI 为 OFF (选项字节: LVIOFF = 1)



- 注**
1. 保证操作的范围时 $1.8\text{ V} \leq V_{DD} \leq 5.5\text{ V}$ 。当供应电压下降时若要使状态处于低于 1.8 V 复位状态时，需使用低压检测器的复位功能或者向 **RESET** 引脚输入低电平。
 2. 在上电申请慢于 0.5 V/ms (MIN.) 后，如果电压比率上升至 1.8 V ，则在电压达到 1.8 V 之前需向 **RESET** 引脚输入低电平，或者用选项字节默认设置 LVI 为 ON (选项字节: LVIOFF = 0)。
 3. 复位处理时间如等待内部电压稳定，包括内部高速振荡时钟的振荡精确稳定时间。
 4. 内部复位处理时间包括内部高速振荡时钟的振荡精确稳定时间。
 5. 内部高速振荡时钟和高速系统时钟或副系统时钟可选作 CPU 时钟。若要使用 X1 时钟，需用 OSTC 寄存器以确认振荡稳定时间是否已过。若要使用 XT1 时钟，需用定时器功能以确认稳定时间是否已过。
 6. 此为初始值，易于更改。

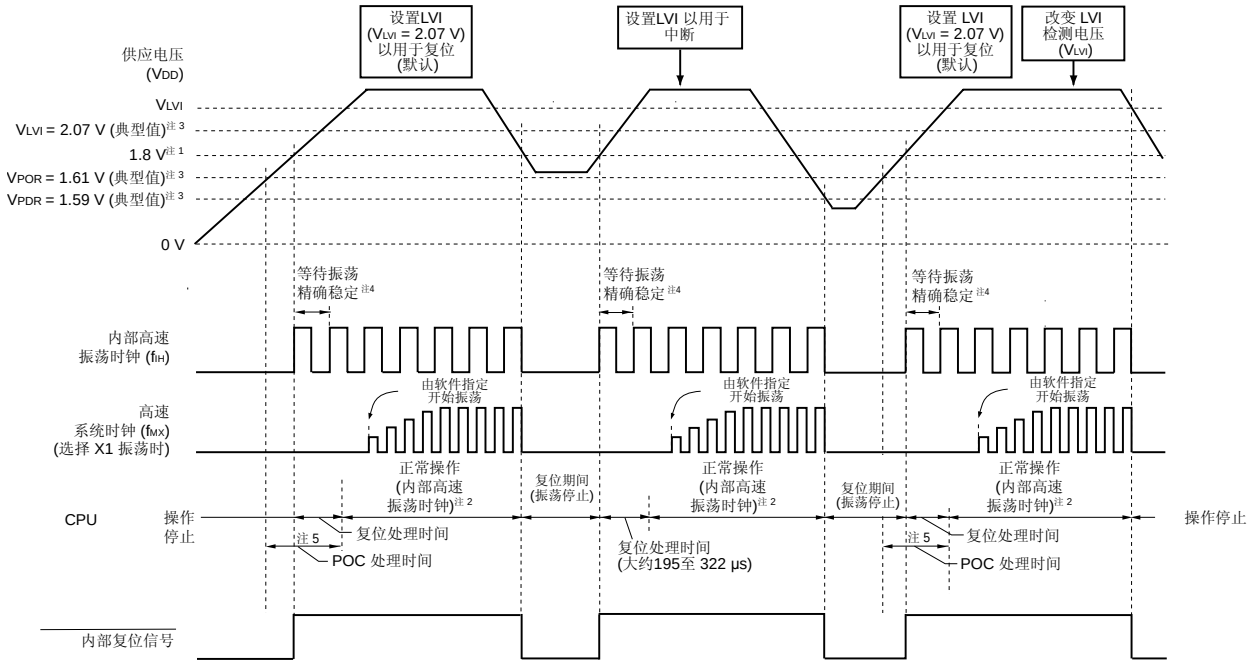
注意事项 复位状态解除后用软件设置电压检测器 (参见第二十一章 低压检测器)。

备注

- VLVI: LVI 检测电压
- VPOR: POC 供应电源上升检测电压
- VPDR: POC 供应电源下降检测电压

图 20-2. 由上电清零电路和电压检测器产生内部复位信号的时序 (2/2)

(2) 上电时 LVI 为 ON (选项字节: LVIOFF = 0)



- 注
1. 保证操作的范围时 $1.8\text{ V} \leq V_{DD} \leq 5.5\text{ V}$ 。当供应电压下降时若要使状态处于低于 1.8 V 复位状态时，需使用低压检测器的复位功能或者向 **RESET** 引脚输入低电平。
 2. 内部高速振荡时钟和高速系统时钟或副系统时钟可选作 CPU 时钟。若要使用 X1 时钟，需用 OSTC 寄存器以确认振荡稳定时间是否已过。若要使用 XT1 时钟，需用定时器功能以确认稳定时间是否已过。
 3. 这些是初始值，易更改。
 4. 内部复位处理时间包括内部高速振荡时钟的振荡精确稳定时间。
 5. 达到 POC 检测电压 (1.61 V (典型值)) 与启动正常操作之间需要以下时间。
 - 当从 1.61 V (典型值) 达到 2.07 V (典型值) 的时间少于 5.8 ms 时：
 - 达到 1.61 V (典型值) 与启动正常操作之间需要大约 2.1 至 6.2 ms 的 POC 处理时间。
 - 当从 1.61 V (典型值) 达到 2.07 V (典型值) 的时间大于 5.8 ms 时：
 - 达到 2.07 V (典型值) 与启动正常操作之间需要大约 195 至 $322\text{ }\mu\text{s}$ 的复位处理时间。

注意事项 复位状态解除后用软件设置电压检测器 (参见第二十一章 低压检测器)。

备注

V_{LVI} : LVI 检测电压

V_{POR} : POC 供应电源上升检测电压

V_{PDR} : POC 供应电源下降检测电压

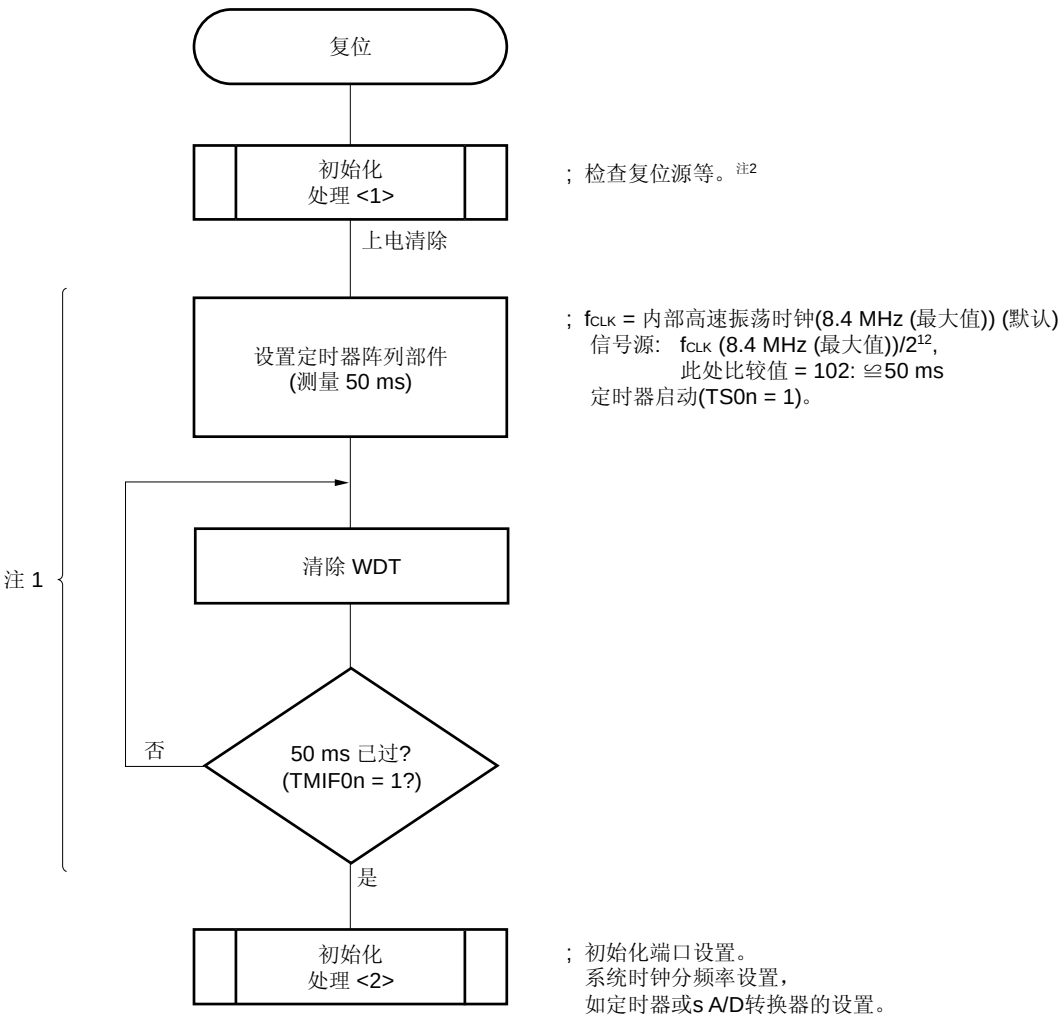
20.4 上电清零电路的注意事项

如果系统的供应电压（V_{DD}）某个时间段内在 POC 检测电压（V_{POR}， V_{PDR}）的附近值波动，则该系统可能重复复位和从复位状态解除。这种情况下，通过执行以下操作，可任意设置从复位解除到启动微控制器操作的时间。

<操作>
解除复位信号后，通过用定时器的软件计数器，等待各系统供应电压的波动时间段，然后初始化端口。

图 20-3. 复位解除后软件处理举例（1/2）

- 如果供应电压在 POC 检测电压附近波动时间为 50 ms 或更少

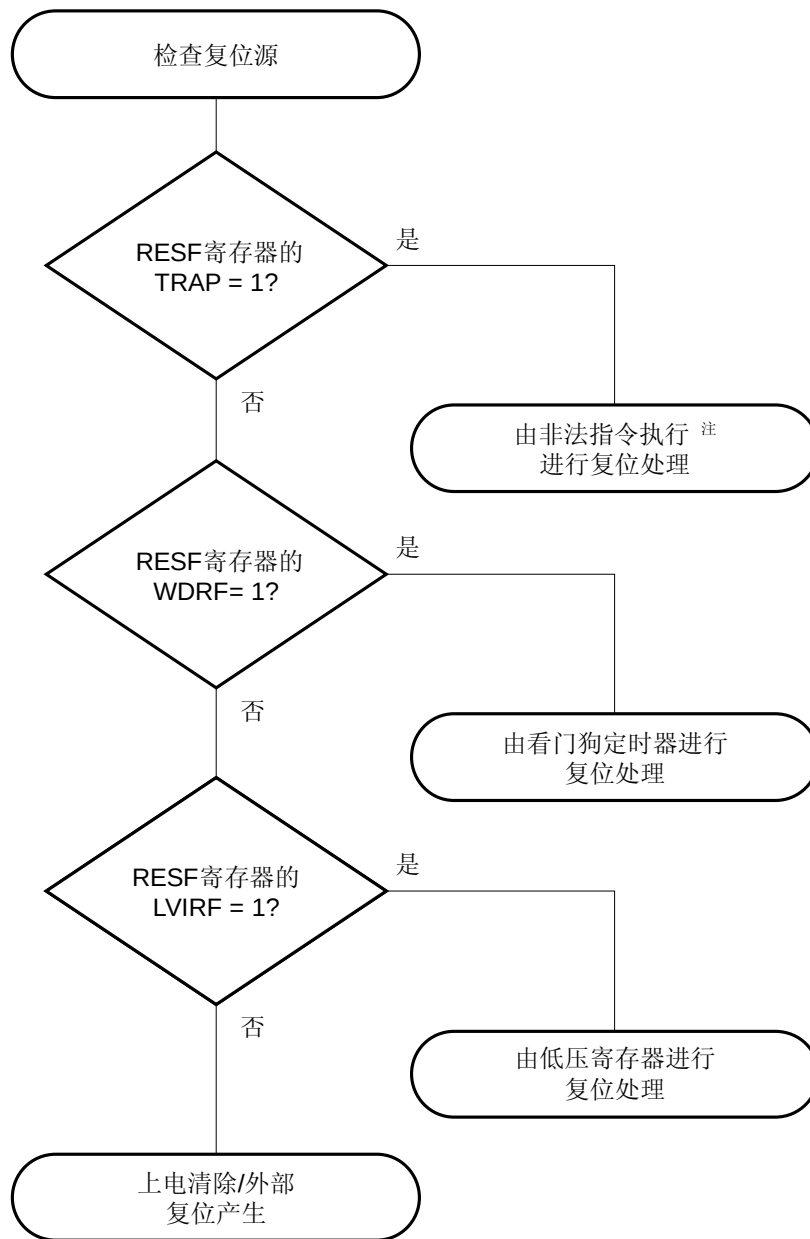


注 1. 如果在该期间再次产生复位，则不开始初始化处理步骤<2>。
2. 流程图在下页。

备注 n = 0 至 7

图 20-3. 复位解除后软件处理举例（2/2）

• 检查复位源



注 该非法指令在执行指令代码 FFH 时产生。
在线仿真器或片上调试仿真器不能产生通过执行非法指令的复位。

第二十一章 低电压检测器

21.1 低电压检测器的功能

低电压监测器（LVI）具有以下功能：

- LVI电路将供电电压（V_{DD}）与检测电压，或与来自检测电压（V_{EXLVI} = 1.21 V ±0.1 V[※]）的外部输入引脚（EXLVI）的输入电压进行比较，从而生成内部复位信号或内部中断信号。
- 低电压检测器（LVI）可以由默认的选项字节设置为ON。如果将其置为ON以从POC检测电压（V_{POR} = 1.61 V（典型值））或更低的电压提高供电电源，则当供电电压（V_{DD}）< 检测电压（V_{LVI} = 2.07 V ±0.2 V[※]）时生成内部复位信号。在这之后，当供电电压（V_{DD}）< 检测电压（V_{LVI} = 2.07 V ±0.1 V[※]）时生成内部复位信号。
- 供电电压（V_{DD}）或来自外部输入引脚（EXLVI）的输入电压可以由软件选择检测。
- 检测后，由软件选择生成复位信号或中断信号。
- 可以由软件更改供电电压的检测等级（V_{LVI}，16级）。
- 在 STOP 模式下操作。

注 此为初始值，可更改。

根据软件的选择，可以生成如下的复位信号和中断信号。

供电电压（V _{DD} ）等级检测的选择（V _{DD} ） （LVISEL = 0）		从外部输入引脚（EXLVI）输入电压等级检测的选择 （LVISEL = 1）	
选择复位（LVIMD = 1）.	选择中断（LVIMD = 0）.	选择复位（LVIMD = 1）.	选择中断（LVIMD = 0）.
当 V _{DD} < V _{LVI} 时产生内部复位信号，当 V _{DD} ≥ V _{LVI} 时解除复位信号。	当 V _{DD} 下降到低于 V _{LVI} （V _{DD} < V _{LVI} ）或当 V _{DD} 变成 V _{LVI} 或更高电压（V _{DD} ≥ V _{LVI} ）时，产生内部中断信号。	当 EXLVI < V _{EXLVI} 时产生内部复位信号，当 EXLVI ≥ V _{EXLVI} 时解除信号。	当 EXLVI 下降到低于 V _{EXLVI} （EXLVI < V _{EXLVI} ）或当 EXLVI 变成 V _{EXLVI} 或更高电压（EXLVI ≥ V _{EXLVI} ）.时，产生内部中断信号。

备注 LVISEL: 低电压检测寄存器（LVIM）的位 2。
LVIMD: LVIM 的位 1。

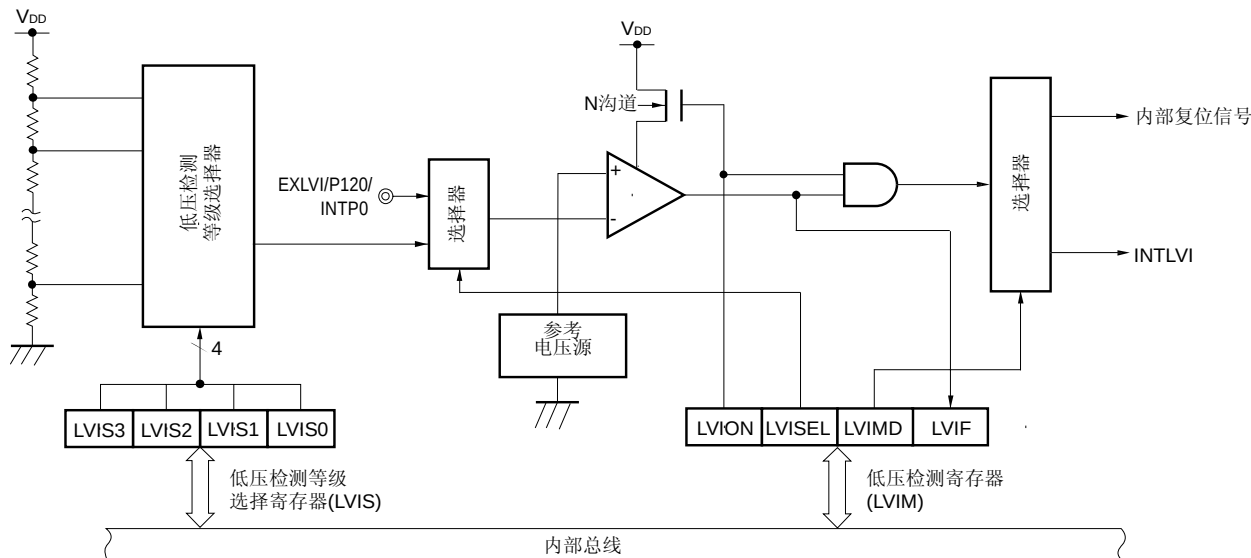
当低电压检测器工作时，通过读取低电压检测标志（LVIF: LVIM 的位 0）可以检测供电电压或来自外部输入引脚的输入电压是高于还是低于检测等级。

当低电压检测器用于复位时，若发生复位则复位控制标志寄存器（RESF）的位 0（LVIRF）置为 1。RESF 细节参见第十九章 RESET 功能。

21.2 电压检测器的配置

电压寄存器的框图如图 21-1。

图 21-1. 低电压检测器的框图



21.3 控制低电压检测器的寄存器

低电压检测器由以下寄存器控制：

- 低电压检测寄存器 (LVIM)。
- 低电压检测等级选择寄存器 (LVIS)。
- 端口模式寄存器 12 (PM12)。

(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。

可通过 1 位或 8 位的存储操作指令设置此寄存器。

产生的复位信号将该寄存器清除为 00H。

图 21-2. 低电压检测寄存器（LVIM）的格式

地址： FFFA9H 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	LVISEL	LVIMD	LVIF

LVION ^{注 3, 4}	使能低电压检测操作
0	禁止操作
1	允许操作

LVISEL ^{注 3}	电压检测选择
0	供应电压（V _{DD} ）的检测等级
1	从外部输入引脚（EXLVI）输入电压的检测等级

LVIMD	低电压检测操作模式（中断/复位）的选择
0	• LVISEL = 0: 当供应电压（V_{DD}）下降到低于检测电压（V_{LVI}）（V_{DD} < V_{LVI}）或当 V_{DD} 变成 V_{LVI} 或 更高电压（V_{DD} ≥ V_{LVI}）时，产生内部中断信号。 • LVISEL = 1: 当从外部输入引脚（EXLVI）的输入电压下降到低于检测电压（V_{EXLVI}）（EXLVI < V_{EXLVI}）或当 EXLVI 变成 V_{EXLVI} 或 更高电压（EXLVI ≥ V_{EXLVI}）时，产生内部中断信号。
1	• LVISEL = 0: 当 供应电压（V_{DD}）< 检测电压（V_{LVI}）时产生内部复位信号，当 V_{DD} ≥ V_{LVI} 时解除复位信号。 • LVISEL = 1: 当从外部输入引脚的输入电压（EXLVI）< 检测电压（V_{EXLVI}）时产生内部复位信号，当 EXLVI ≥ V_{EXLVI} 时解除复位信号。

LVIF	低电压检测标志
0	• LVISEL = 0: 供应电压（V_{DD}）≥ 检测电压（V_{LVI}），或当禁止 LVI 操作时 • LVISEL = 1: 从外部输入引脚（EXLVI）的输入电压 ≥ 检测电压（V_{EXLVI}），或当禁止 LVI 操作时
1	• LVISEL = 0: 供应电压（V_{DD}）< 检测电压（V_{LVI}） • LVISEL = 1: 从外部输入引脚（EXLVI）的输入电压（EXLVI）< 检测电压（V_{EXLVI}）

- 注
1. 复位值根据复位源和选项字节设置的不同而不同。
该寄存器不能由 LVI 复位清除（00H）。
当应用除 LVI 外的复位信号时，如果选项字节 LVIOFF = 0 则该寄存器被设置为“82H”，如果选项字节 LVIOFF = 1，则该寄存器被设置为“00H”。
 2. 位 0 为只读位。
 3. 在除 LVI 复位的其它复位情况下，LVION， LVIMD 和 LVISEL 被清除为 0。在 LVI 复位情况下这些不能被清除为 0。

<R> 注 4. 当 LVION 被设置为 1 时, LVI 电路的比较器开始工作。在 LVION 被置 1 与电压用 LVIF 确认之间, 需使用软件等待以下时间段。

- 操作稳定时间 (10 μ s (MAX.))
- 最小脉冲宽度 (200 μ s (MIN.))

这段时间中, LVIF 的值不论电压等级为多少都可能被设置/清除, 因此, 不能使用该值。同样, 这段时间中 LVIIF 中断请求标志可能被置 1。

<R> 注意事项 1. 若要停止 LVI, 确保用 1 位存储操作指令将 LVION 清除为 (0)。
2. 从外部输入引脚输入的电压 (EXLVI) 必须为 $EXLVI < V_{DD}$ 。
3. 当 LVI 用于中断模式 (LVIMD = 0) 而 LVISEL 被置 0 时, 当供应电压 (V_{DD}) 低于或等于检测电压 (V_{LVI}) (如果 LVISEL = 1, 则外部输入引脚的输入电压 (EXLVI) 低于或等于检测电压 (V_{EXLVI})) 时, 产生中断请求信号 (INTLVI), 而 LVIIF 有可能被置 1。

(2) 低电压检测等级选择寄存器 (LVIS)

该寄存器选择低电压检测等级。
可通过 1 位或 8 位的存储操作指令设置此寄存器。
输入产生的复位信号将该寄存器设置为 0EH。

图 21-3. 低电压检测等级选择寄存器 (LVIS) 的格式

地址: FFFAAH 复位后: 0EH^{注 1} 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	检测等级
0	0	0	0	V _{LV10} (4.22 ±0.1 V) ^{注 2}
0	0	0	1	V _{LV11} (4.07 ±0.1 V) ^{注 2}
0	0	1	0	V _{LV12} (3.92 ±0.1 V) ^{注 2}
0	0	1	1	V _{LV13} (3.76 ±0.1 V) ^{注 2}
0	1	0	0	V _{LV14} (3.61 ±0.1 V) ^{注 2}
0	1	0	1	V _{LV15} (3.45 ±0.1 V) ^{注 2}
0	1	1	0	V _{LV16} (3.30 ±0.1 V) ^{注 2}
0	1	1	1	V _{LV17} (3.15 ±0.1 V) ^{注 2}
1	0	0	0	V _{LV18} (2.99 ±0.1 V) ^{注 2}
1	0	0	1	V _{LV19} (2.84 ±0.1 V) ^{注 2}
1	0	1	0	V _{LV110} (2.68 ±0.1 V) ^{注 2}
1	0	1	1	V _{LV111} (2.53 ±0.1 V) ^{注 2}
1	1	0	0	V _{LV112} (2.38 ±0.1 V) ^{注 2}
1	1	0	1	V _{LV113} (2.22 ±0.1 V) ^{注 2}
1	1	1	0	V _{LV114} (2.07 ±0.1 V) ^{注 2}
1	1	1	1	V _{LV115} (1.91 ±0.1 V) ^{注 2}

- 注 1. 复位值根据复位源的不同而不同。
如果用 LVI 对 LVIS 寄存器进行复位，则该寄存器不被复位而是保持当前值。如果用 LVI 外的其它复位源进行的复位有效，则该寄存器的值被复位至“0EH”。
2. 这些是初始值，易更改。

注意事项 1. 务必将位 4 至位 7 清为“0”。

注意事项 2. 用以下方法之一更改 LVIS 的值。

- 停止 LVI 后更改 LVIS 的值时
 - <1> 停止 LVI (LVION = 0)。
 - <2> 更改 LVIS 寄存器。
 - <3> 设置为用作中断的模式 (LVIMD = 0)。
 - <4> 屏蔽 LVI 中断 (LVIMK = 1)。
 - <5> 使能 LVI 操作 (LVION = 1)。
 - <6> 取消 LVI 中断屏蔽 (LVIMK = 0) 前, 因为在使能 LVI 操作时 LVIIF 标志可能被设置, 所以用软件将其清除。
- 设置为用作中断 (LVIMD = 0) 的模式后更改 LVIS 的值时
 - <1> 屏蔽 LVI 中断 (LVIMK = 1)。
 - <2> 设置为用作中断的模式 (LVIMD = 0)。
 - <3> 更改 LVIS 寄存器。
 - <4> 取消 LVI 中断屏蔽 (LVIMK = 0) 前, 因为在 LVIS 寄存器被更改时 LVIIF 标志可能被设置, 所以用软件将其清除。

3. 检测到从外部输入引脚的输入电压 (EXLVI) 时, 检测电压 (V_{EXLVI}) 被固定。因此不需要设置 LVIS。

(3) 端口模式寄存器 12 (PM12)

当用 P120/EXLVI/INTP0 引脚作为外部低电压检测电压输入时, 将 PM120 设置为 1。此时 P120 的输出锁存可能为 0 也可能为 1。

可通过 1 位或 8 位的存储操作指令设置 PM12。

产生的复位信号将该寄存器设置为 FFH。

图 21-4. 端口模式寄存器 12 (PM12) 的格式

地址: FFF2CH 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PM12	1	1	1	1	1	1	1	PM120

PM120	P120 引脚 I/O 模式选择
0	输出模式 (输出缓存打开)
1	输入模式 (输出缓存关闭)

21.4 低电压检测器的操作

低电压检测器可以使用以下两种模式。

(1) 用作复位 (LVIMD = 1)

- 如果 LVISEL = 0，将供应电压 (V_{DD}) 和检测电压 (V_{LVI}) 进行比较，当 $V_{DD} < V_{LVI}$ 时，产生内部复位信号；当 $V_{DD} \geq V_{LVI}$ 时，解除内部复位。
- 如果 LVISEL = 1，将从外部输入引脚的输入电压 ($EXLVI$) 和检测电压 (V_{EXLVI}) 进行比较，当 $EXLVI < V_{EXLVI}$ 时，产生内部复位信号；当 $EXLVI \geq V_{EXLVI}$ 时，解除内部复位。

备注 低电压检测器 (LVI) 可以由默认的选项字节设置为 ON。如果将其置为 ON 以从 POC 检测电压 ($V_{POR} = 1.61\text{ V}$ (典型值)) 或更低的电压提高供电电源，则当供电电压 (V_{DD}) < 检测电压 ($V_{LVI} = 2.07\text{ V} \pm 0.2\text{ V}^{\#}$) 时生成内部复位信号。在这之后，当供电电压 (V_{DD}) < 检测电压 ($V_{LVI} = 2.07\text{ V} \pm 0.1\text{ V}^{\#}$) 时生成内部复位信号。

(2) 用作中断 (LVIMD = 0)

- 如果 LVISEL = 0，对供应电压 (V_{DD}) 和检测电压 (V_{LVI}) 进行比较。当 V_{DD} 下降到低于 V_{LVI} ($V_{DD} < V_{LVI}$) 或当 V_{DD} 变成 V_{LVI} 或更高电压 ($V_{DD} \geq V_{LVI}$) 时，产生中断信号 (INTLVI)。
- 如果 LVISEL = 1，对从外部输入引脚的输入电压 ($EXLVI$) 和检测电压 ($V_{EXLVI} = 1.21\text{ V} \pm 0.1\text{ V}^{\#}$) 进行比较。当 $EXLVI$ 下降到低于 V_{EXLVI} ($EXLVI < V_{EXLVI}$) 或当 $EXLVI$ 变成 V_{EXLVI} 或更高电压 ($EXLVI \geq V_{EXLVI}$) 时，产生中断信号 (INTLVI)。

注 此为初始值，易于更改。

当低电压检测器工作时，通过读取低电压检测标志 (LVIF: LVIM 的位 0) 可以检测供电电压或来自外部输入引脚的输入电压是高于还是低于检测等级。

备注 LVIMD: 低电压检测寄存器 (LVIM) 的位 1。
LVISEL: LVIM 的位 2。

21.4.1 用作复位时

(1) 当检测供应电压 (V_{DD}) 的等级时(a) 当设置为 LVI 默认启动功能停止时 ($LVIOFF = 1$)

• 启动操作时

- <1> 屏蔽 LVI 中断 ($LVIMK = 1$)。
- <2> 清除低电压检测寄存器 (LVIM) 的位 2 ($LVISEL$) 为 0 (检测供应电压 (V_{DD}) 的等级) (默认值)。
- <3> 用低电压检测等级选择寄存器 (LVIS) 的位 3 至位 0 ($LVIS3$ 至 $LVIS0$) 设置检测电压。
- <4> 将 LVIM 的位 7 ($LVION$) 设置为 1 (使能 LVI 操作)。
- <5> 用软件等待以下时间区间 (总计 $210\mu s$)。
 - 操作稳定时间 ($10\mu s$ (MAX.))
 - 最小脉冲宽度 ($200\mu s$ (MIN.))
- <6> 等待直到用 LVIM 的位 0 ($LVIF$) 检测到供应电压 (V_{DD}) $\geq$ 检测电压 (V_{LVI}) 为止。
- <7> 将 LVIM 的位 1 ($LVIMD$) 设置为 1 (检测到该电平时产生复位)。

<R>

图 21-5 显示了由低电压检测器产生内部复位信号的时序。时序流程中的数字对应以上的步骤<1>至<7>。

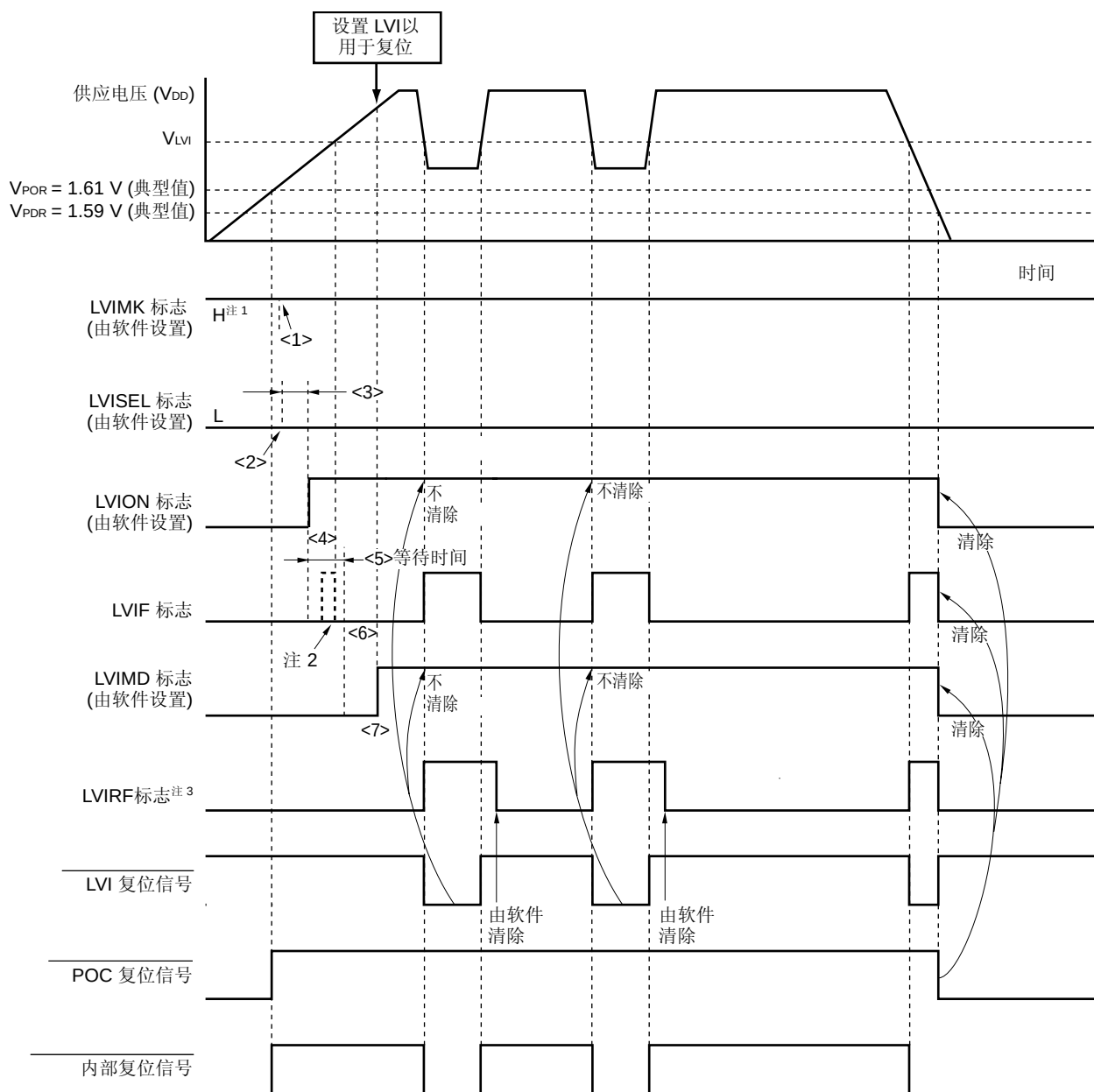
- 注意事项**
1. 务必执行步骤<1>。当 $LVIMK = 0$ 时，在步骤<4>的处理后有可能立即发生中断。
 2. 当 $LVIMD$ 设置为 1 时，如果供应电压 (V_{DD}) $\geq$ 检测电压 (V_{LVI})，则不产生内部复位信号。

• 当停止操作时

确保用 1 位存储操作指令将 $LVIMD$ 清零然后将 $LVION$ 清零。

<R>

图 21-5. 由低电压检测器产生内部复位信号的时序
(位: LVISEL = 0, 选项字节: LVIOFF = 1)



- 注
1. LVIMK 标志被产生的复位信号设置为“1”。
 2. 中断请求标志寄存器的 LVIF 标志和 LVIF 标志可能被设置为 (1)。
 3. LVIRF 是复位控制标志寄存器 (RESF) 的位 0。RESF 细节参见第十九章 RESET 功能。

- 备注
1. 以上图 21-5 中的<1> 至 <7>对应于 21.4.1 (1) (a) 当设置为 LVI 默认启动功能停止时 (LVIOFF = 1) 中“当启动操作时”中描述的<1> 至<7>。
 2. V_{POR}: POC 供应电源上升检测电压
V_{PDR}: POC 供应电源下降检测电压

(b) 当设置为 LVI 默认启动功能使能时 (LVIOFF = 0)

- 启动操作时

在以下初始设置状态中启动。

- 将 LVIM 的位 7 (LVION) 设置为 1 (使能 LVI 操作)。
- 清除低电压检测寄存器 (LVIM) 的位 2 (LVISEL) 为 0 (检测供应电压 (V_{DD}) 的等级)。
- 设置低电压检测等级寄存器 (LVIS) 为 0EH (默认值: V_{LVI} = 2.07 V ± 0.1 V)。
- 将 LVIM 的位 1 (LVIMD) 设置为 1 (检测到该电平时产生复位)。
- 设置 LVIM 的位 0 (LVIF) 为 0 (“供应电压 (V_{DD}) ≥ 检测电压 (V_{LVI})”)。

图 21-6 显示了由低电压检测器产生内部复位信号的时序。

- 当停止操作时

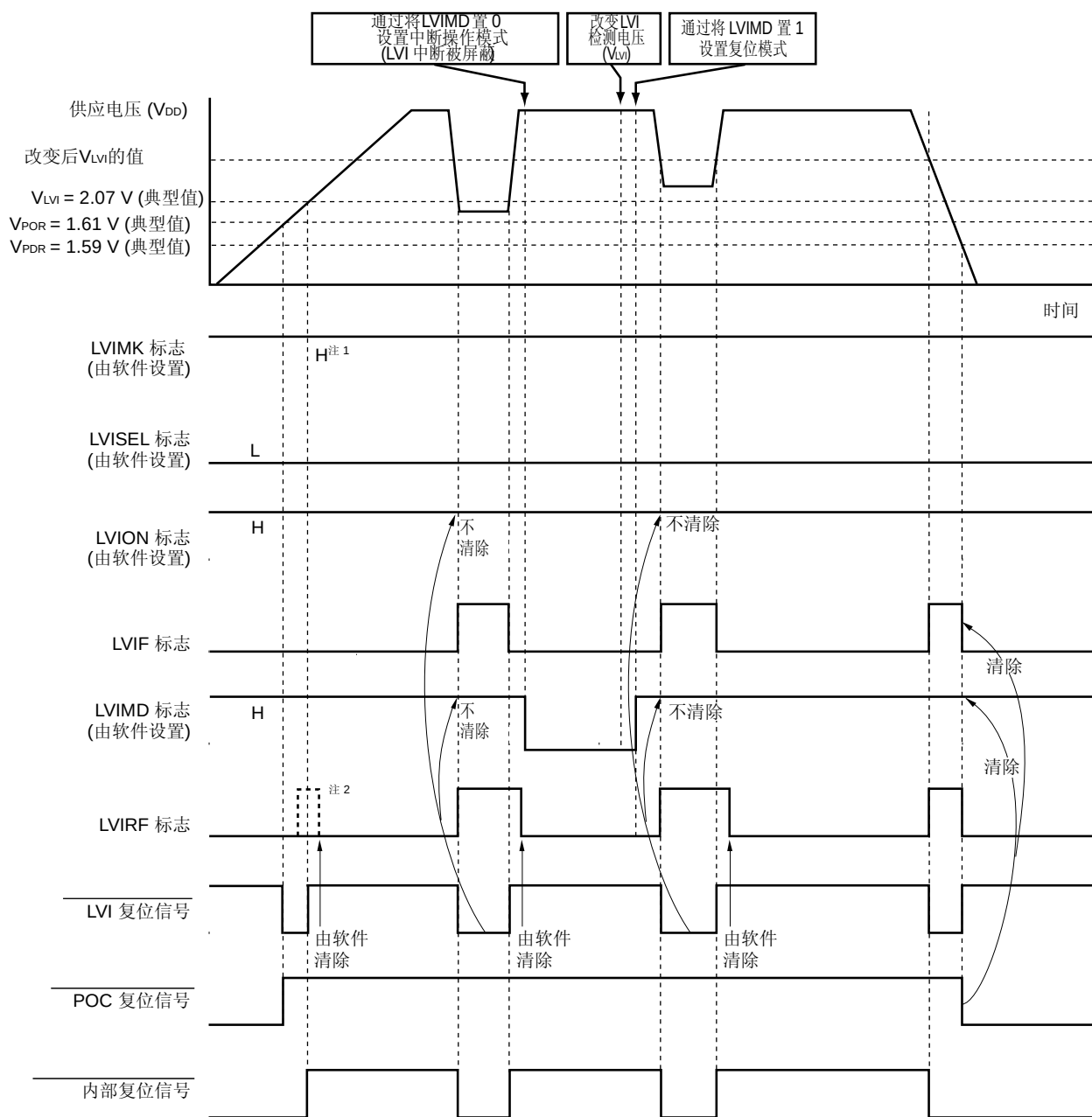
<R>

确保用 1 位存储操作指令将 LVIMD 清零然后将 LVION 清零。

注意事项 当使用 LVI 默认启动功能时，若用软件将其设置为 LVI 操作禁止时，其操作如下：

- LVION = 0 期间不执行低电压检测。
- LVION = 0 若产生复位，则在复位解除后当 CPU 启动时需将 LVION 重新设置为 1。但是，由于 WDT 和执行非法指令而发生复位时，有一时间段不能正常执行低电压检测。这是因为由 LVI 检测的脉冲宽度必需为 200 μs max.，基于复位发生 LVION = 1 被设置，而在未等待 LVI 稳定时间的情况下 CPU 开始工作。

图 21-6. 由低电压检测器产生内部复位信号的时序
(位: LVISEL = 0, 选项字节: LVIOFF = 0)



注 1. LVIMK 标志被产生的复位信号设置为“1”。

2. LVIRF 是复位控制标志寄存器 (RESF) 的位 0。

当使用 LVI 默认启动功能（000C1H 的位 0（LVIOFF）= 0）时，根据上电波形，LVIRF 标志有可能从开始就变成 1。

RESF 细节参见第十九章 RESET 功能。

备注 V_{POR} : POC 供应电源上升检测电压

V_{PDR} : POC 供应电源下降检测电压

(2) 当检测从外部输入引脚的输入电压 (EXLVI) 的等级时

• 启动操作时

- <1> 屏蔽 LVI 中断 ($LVIMK = 1$)。
- <2> 设置低电压检测寄存器 (LVIM) 的位 2 (LVISEL) 为 1 (检测从外部输入引脚的输入电压 (EXLVI) 的等级)。
- <3> 将 LVIM 的位 7 (LVION) 设置为 1 (使能 LVI 操作)。
- <4> 用软件等待以下时间区间 (总计 $210\mu s$)。
- 操作稳定时间 ($10\mu s$ (MAX.))
 - 最小脉冲宽度 ($200\mu s$ (MIN.))
- <5> 等待直到用 LVIM 的位 0 (LVIF) 检测到从外部输入引脚的输入电压 (EXLVI) $\geq$ 检测电压 ($V_{EXLVI} = 1.21 V$ (典型值)) 为止。
- <6> 将 LVIM 的位 1 (LVIMD) 设置为 1 (检测到该等级时产生复位信号)。

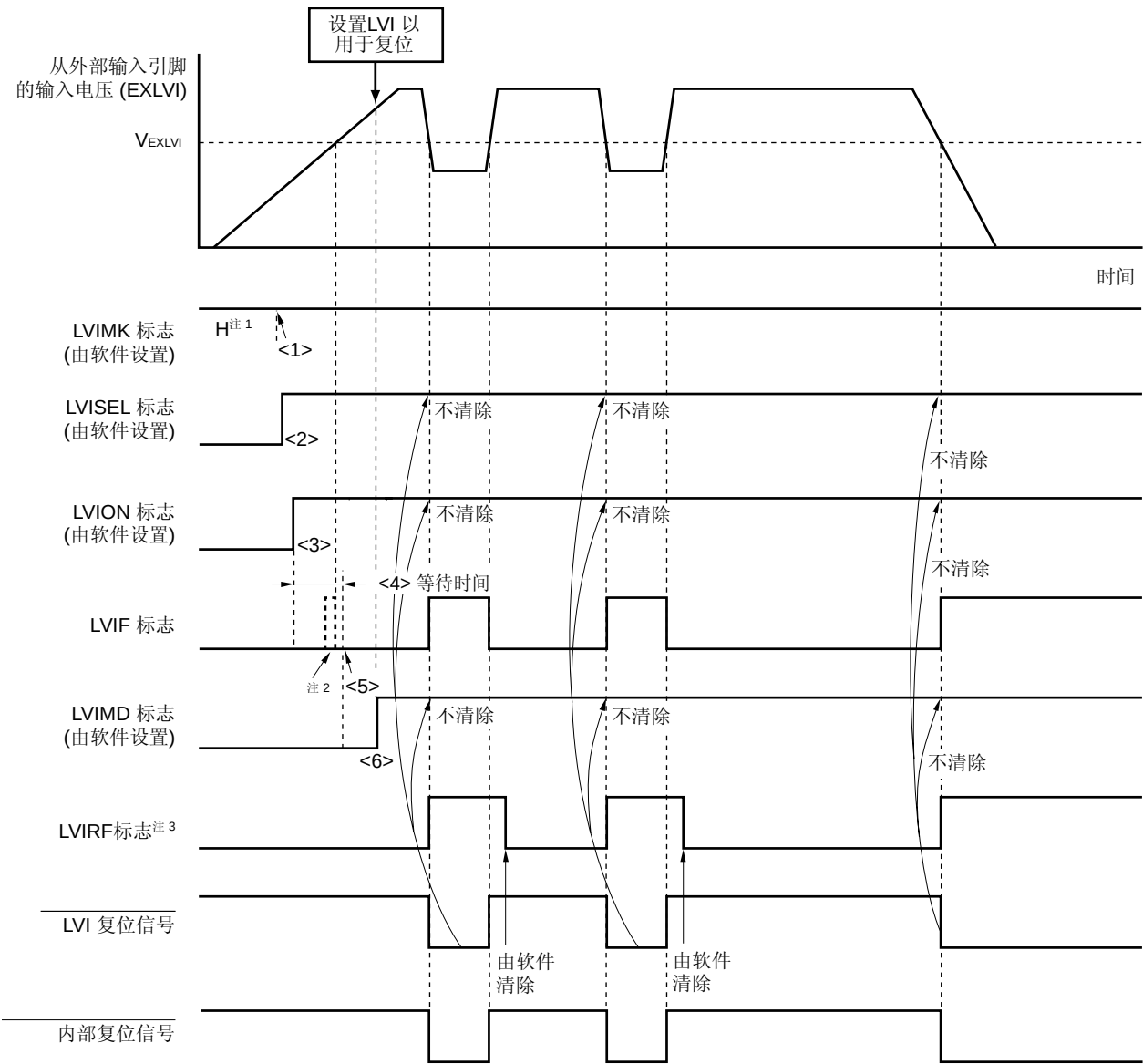
图 21-7 显示了由低电压检测器产生内部复位信号的时序。时序流程中的数字对应以上的步骤<1>至<6>。

- 注意事项**
1. 务必执行步骤<1>。当 $LVIMK = 0$ 时, 在步骤<3>的处理后有可能立即发生中断。
 2. 当 LVIMD 设置为 1 时, 如果从外部输入引脚的输入电压 (EXLVI) $\geq$ 检测电压 ($V_{EXLVI} = 1.21 V$ (典型值)), 则不产生内部复位信号。
 3. 从外部输入引脚输入的电压 (EXLVI) 必须为 $EXLVI < V_{DD}$ 。

• 当停止操作时

确保用 1 位存储操作指令将 LVIMD 清零然后将 LVION 清零。

图 21-7. 由低电压检测器产生内部复位信号的时序
(位: LVISEL =1)



- 注
1. LVIMK 标志被产生的复位信号设置为“1”。
 2. 中断请求标志寄存器的 LVIF 标志和 LVIF 标志可能被设置为 (1) 。
 3. LVIRF 是复位控制标志寄存器 (RESF) 的位 0。RESF 细节参见第十九章 RESET 功能。

备注 以上图 21-7 中的<1> 至 <6>对应于 21.4.1 (2) 当检测从外部输入引脚的输入电压 (EXLVI) 等级时中“当启动操作时”中描述的<1> 至<6> 。

21.4.2 用作中断时

(1) 当检测供应电压 (V_{DD}) 的等级时(a) 当设置为 LVI 默认启动功能停止时 ($LVIOFF = 1$)

• 启动操作时

- <1> 屏蔽 LVI 中断 ($LVIMK = 1$)。
- <2> 清除低电压检测寄存器 (LVIM) 的位 2 (LVISEL) 为 0 (检测供应电压 (V_{DD}) 的等级) (默认值)。
清除 LVIM 的位 1 (LVIMD) 设置为 0 (检测到该等级时产生中断信号) (默认值)。
- <3> 用低电压检测等级选择寄存器 (LVIS) 的位 3 至位 0 (LVIS3 至 LVIS0) 设置检测电压。
- <4> 将 LVIM 的位 7 (LVION) 设置为 1 (使能 LVI 操作)。
- <5> 用软件等待以下时间区间 (总计 210 μs)。
 - 操作稳定时间 (10 μs (MAX.))
 - 最小脉冲宽度 (200 μs (MIN.))
- <6> 在 LVIM 的位 0 (LVIF) 检测到 V_{DD} 的下降沿时确认“供应电压 (V_{DD}) $\geq$ 检测电压 (V_{LVI})”，或者在 LVIM 的位 0 (LVIF) 检测到 V_{DD} 的上升沿时确认“供应电压 (V_{DD}) $<$ 检测电压 (V_{LVI})”。
- <7> 清除 LVI 的中断请求标志 (LVIIIF) 为 0。
- <8> 解除 LVI 的中断屏蔽标志 (LVIMK)。
- <9> 执行 EI 指令 (当使用向量中断时)。

<R>

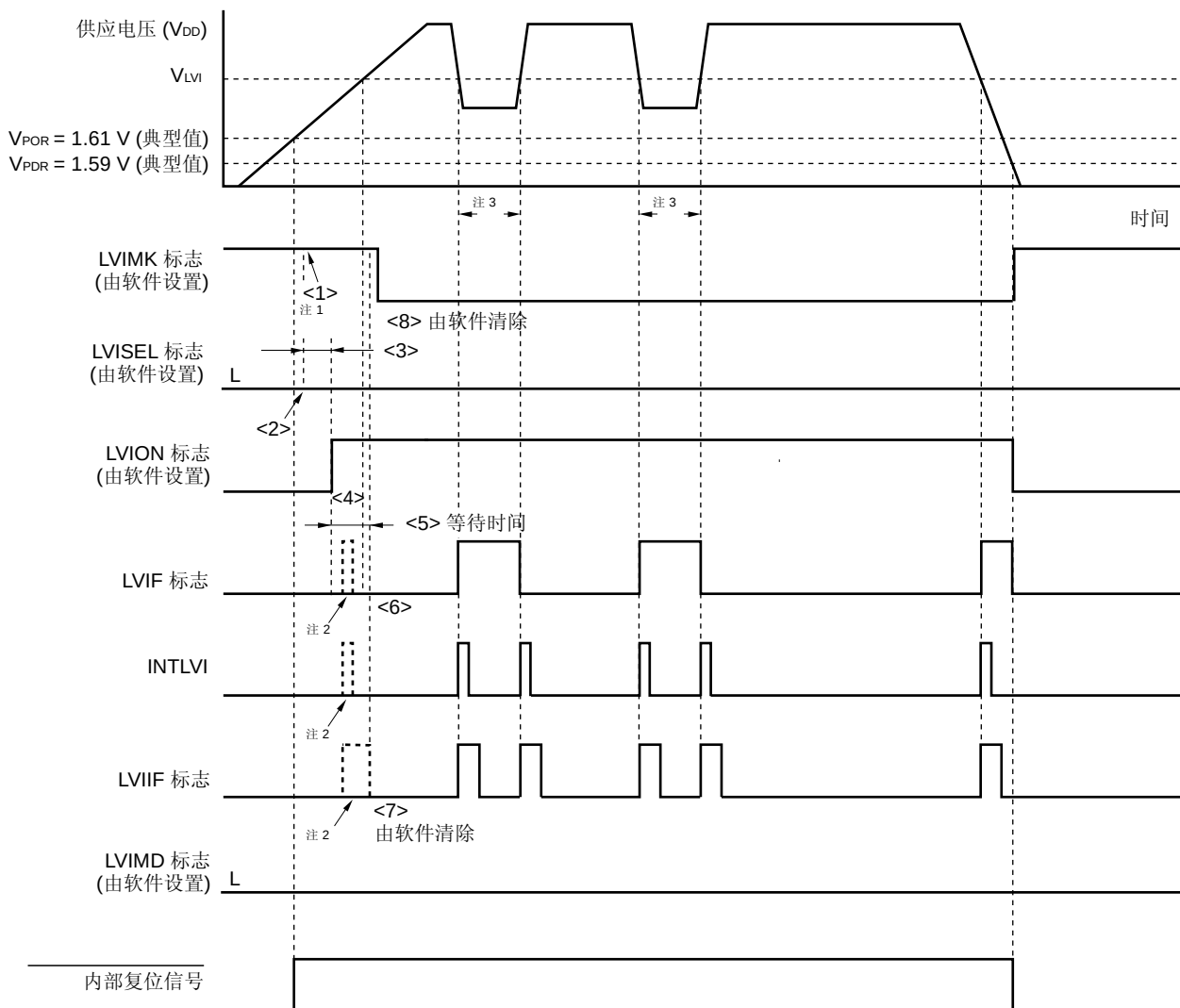
图 21-8 显示了由低电压检测器产生中断信号的时序。时序流程中的数字对应以上的步骤<1>至<8>。

• 当停止操作时

确保用 1 位存储操作指令将 LVION 清零。

<R>

图 21-8. 由低电压检测器产生中断信号的时序
(位: LVISEL = 0, 选项字节: LVIOFF = 1)



- 注
1. LVIMK 标志被产生的复位信号设置为“1”。
 2. 产生中断请求信号 (INTLVI) 而 LVIF 和 LVIIF 标志可能被值 (1)。
 3. 当供应电压 (V_{DD}) 低于或等于检测电压 (V_{LVI}) 时, 如果禁止 LVI 操作, 则产生中断请求信号 (INTLVI) 而 LVIIF 可能被置 1。

- 备注
1. 以上图 21-8 中的<1> 至 <8>对应于 21.4.2 (1) (a) 当设置为 LVI 默认启动功能停止时 (LVIOFF = 1) 中 “当启动操作时”中描述的<1> 至<8>。
 2. V_{POR} : POC 供应电源上升检测电压
 V_{PDR} : POC 供应电源下降检测电压

(b) 当设置为 LVI 默认启动功能使能时 (LVIOFF = 0)

- 启动操作时

<1> 在以下初始设置状态中启动。

- 将 LVIM 的位 7 (LVION) 设置为 1 (使能 LVI 操作)。
- 清除低电压检测寄存器 (LVIM) 的位 2 (LVISEL) 为 0 (检测供应电压 (V_{DD}) 的等级)。
- 设置低电压检测等级寄存器 (LVIS) 为 0EH (默认值: $V_{LVI} = 2.07\text{ V} \pm 0.1\text{ V}$)。
- 将 LVIM 的位 1 (LVIMD) 设置为 1 (检测到该等级时产生复位)。
- 设置 LVIM 的位 0 (LVIF) 为 0 (检测到下降沿“供应电压 (V_{DD}) $\geq$ 检测电压 (V_{LVI})”)。

<2> 清除 LVIM 的位 1 (LVIMD) 设置为 0 (检测到该等级时产生中断信号) (默认值)。

<3> 解除 LVI 的中断屏蔽标志 (LVIMK)。

<4> 执行 EI 指令 (当使用向量中断时)。

图 21-9 显示了由低电压检测器产生中断信号的时序。时序流程中的数字对应以上的步骤<1>至<3>。

- 当停止操作时

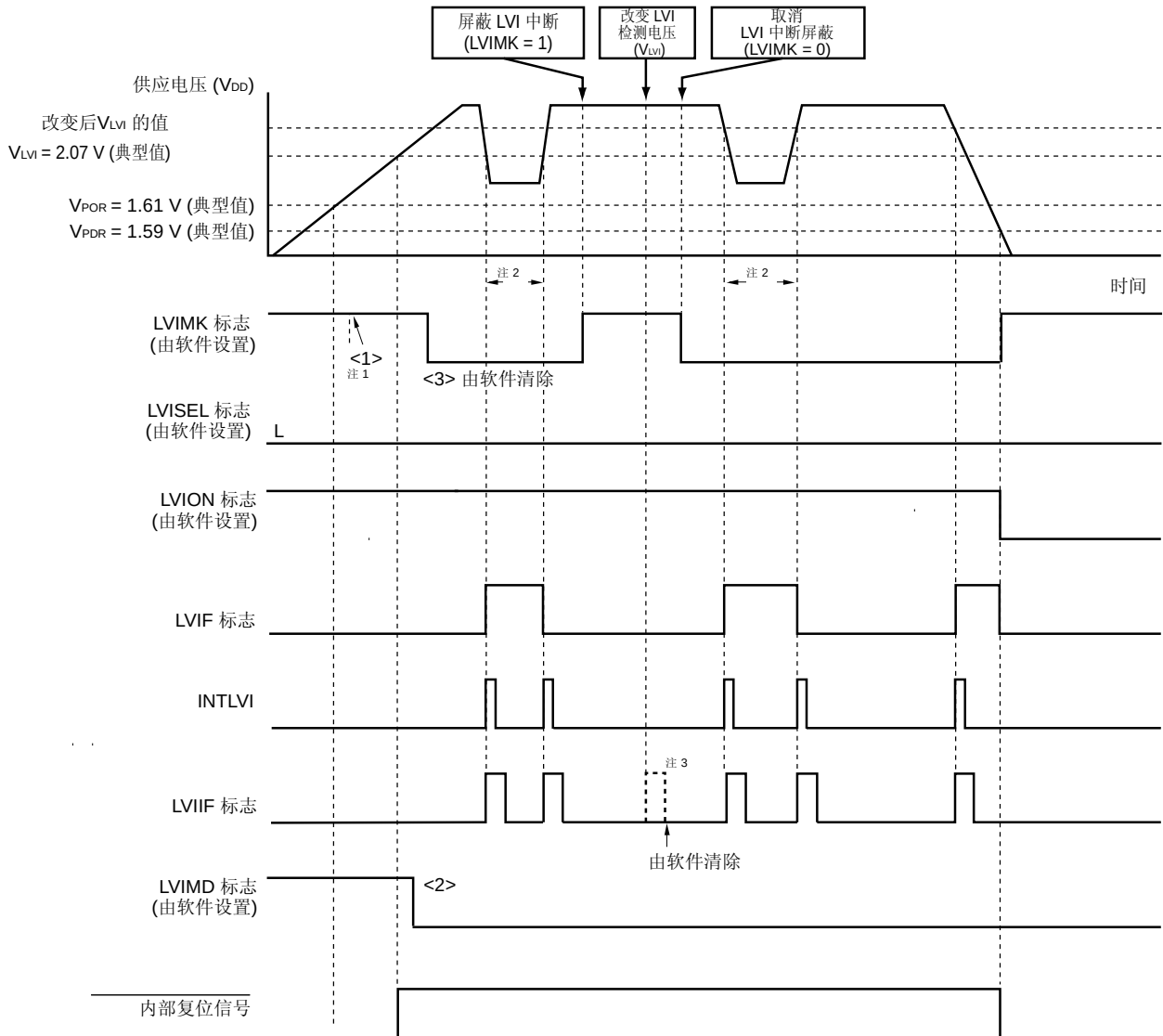
<R>

确保用 1 位存储操作指令将 LVION 清零。

注意事项 1. 当使用 LVI 默认启动功能时, 若用软件将其设置为 LVI 操作禁止时, 其操作如下:

- LVION = 0 期间不执行低电压检测。
 - LVION = 0 若产生复位, 则在复位解除后当 CPU 启动时需将 LVION 重新设置为 1。但是, 由于 WDT 和执行非法指令而发生复位时, 有一时间段不能正常执行低电压检测。
这是因为由 LVI 检测的脉冲宽度必需为 $200\text{ }\mu\text{s max.}$, 基于复位发生 LVION = 1 被设置, 而在未等待 LVI 稳定时间的情况下 CPU 开始工作。
2. 当使用 LVI 默认启动功能 (000C1H 的位 0 (LVIOFF) = 0) 时, 根据上电波形, LVIRF 标志有可能从开始就变成 1。
RESF 细节参见第十九章 RESET 功能。

图 21-9. 由低电压检测器产生中断信号的时序
(位: LVISEL = 0, 选项字节: LVIOFF = 0)



- 注
1. LVIMK 标志被产生的复位信号设置为“1”。
 2. 当供应电压 (V_{DD}) 低于或等于检测电压 (V_{LVI}) 时, 如果禁止 LVI 操作, 则产生中断请求信号 (INTLVI) 而 LVIIF 可能被置 1。
 3. 当 LVI 检测电压被改变时 LVIIF 标志有可能被设置。

- 备注
1. 以上图 21-9 中的<1> 至 <3>对应于 21.4.2 (1) (b) 当设置为 LVI 默认启动功能使能时 (LVIOFF = 0) 中“当启动操作时”中描述的<1> 至<3>。
 2. V_{POR} : POC 供应电源上升检测电压
 V_{PDR} : POC 供应电源下降检测电压

(2) 当检测从外部输入引脚的输入电压 (EXLVI) 的等级时

• 启动操作时

- <1> 屏蔽 LVI 中断 (LVIMK = 1)。
- <2> 设置低电压检测寄存器 (LVIM) 的位 2 (LVISEL) 为 1 (检测从外部输入引脚的输入电压 (EXLVI) 的等级)。
清除 LVIM 的位 1 (LVIMD) 设置为 0 (检测到该等级时产生中断信号) (默认值)。
- <3> 将 LVIM 的位 7 (LVION) 设置为 1 (使能 LVI 操作)。
- <4> 用软件等待以下时间区间 (总计 210 μ s) 。
 - 操作稳定时间 (10 μ s (MAX.))
 - 最小脉冲宽度 (200 μ s (MIN.))
- <5> 在 LVIM 的位 0 (LVIF) 检测到 EXLVI 的下降沿时, 确认“从外部输入引脚的输入电压 (EXLVI) $\geq$ 检测电压 ($V_{EXLVI} = 1.21\text{ V}$ (典型值))”; 或者在 LVIM 的位 0 (LVIF) 检测到 EXLVI 的上升沿时, 确认“从外部输入引脚的输入电压 (EXLVI) $<$ 检测电压 ($V_{EXLVI} = 1.21\text{ V}$ (典型值))”。
- <6> 清除 LVI 的中断请求标志 (LVIF) 为 0。
- <7> 解除 LVI 的中断屏蔽标志 (LVIMK) 。
- <8> 执行 EI 指令 (当使用向量中断时)。

<R>

图 21-10 显示了由低电压检测器产生中断信号的时序。时序流程中的数字对应以上的步骤<1>至<7>。

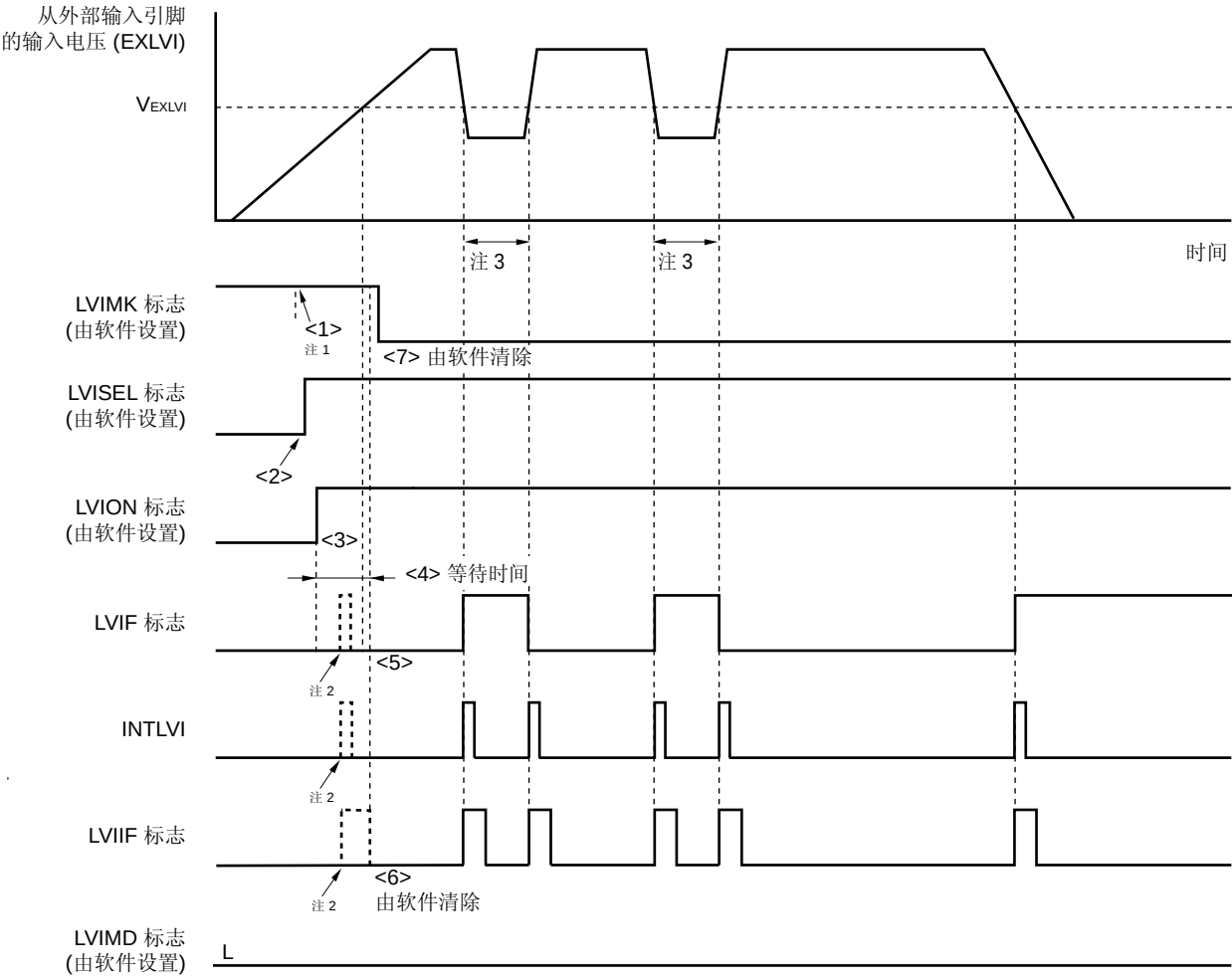
注意事项 从外部输入引脚输入的电压 (EXLVI) 必须为 $EXLVI < V_{DD}$ 。

• 当停止操作时

确保用 1 位存储操作指令将 LVION 清零。

<R>

图 21-10. 由低电压检测器产生中断信号的时序
(位: LVISEL = 1)



注 1. LVIMK 标志被产生的复位信号设置为“1”。
2. 产生中断请求信号 (INTLVI) 而 LVIF 和 LVIIF 标志可能被值 (1)。
3. 当外部输入引脚的输入电压 (EXLVI) 低于或等于检测电压 (V_{EXLVI}) 时, 如果禁止 LVI 操作, 则产生中断请求信号 (INTLVI) 而 LVIIF 可能被置 1。

备注 以上图 21-10 中的<1> 至 <7>对应于 21.4.2 (2) 当检测从外部输入引脚的输入电压 (EXLVI) 等级时中 “当启动操作时”中描述的<1> 至<7> 。

21.5 有关低电压检测器的注意事项

(1) 当供应电压 (V_{DD}) 频率在 LVI 检测电压 (V_{LVI}) 附近波动的测量方法

当系统的供应电压 (V_{DD}) 某段时间内在 LVI 检测电压 (V_{LVI}) 附近波动时, 根据低电压检测器如何使用, 进行以下操作。

操作举例 1: 用作复位时

该系统可能被重复复位和从复位状态中解除。

从复位解除到微控制器操作启动的时间可用以下操作任意设置。

<操作>

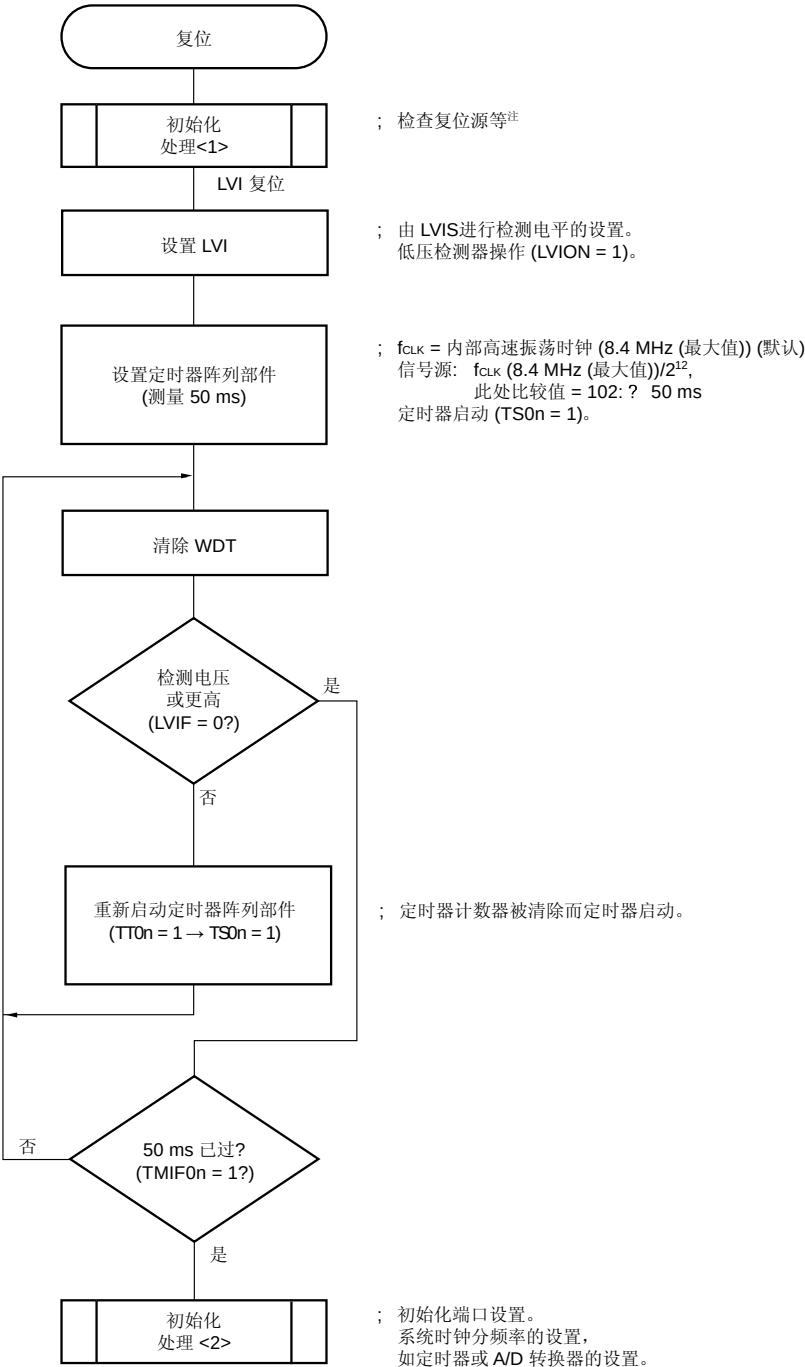
解除复位信号后, 通过用定时器的软件计数器, 等待各系统供应电压的波动时间段, 然后初始化端口 (参见图 21-11)。

备注 如果低电压检测寄存器 (LVIM) 的位 2 (LVISEL) 设置为“1”, 则以上描述需进行如下更改。

- 供应电压 (V_{DD}) → 从外部输入引脚的输入电压 (V_{EXLVI})
- 检测电压 (V_{LVI}) → 检测电压 ($V_{EXLVI} = 1.21\text{ V}$)

图 21-11. 复位解除后软件处理举例（1/2）

- 如果供应电压在 LVI 检测电压附近波动时间为 50 ms 或更少

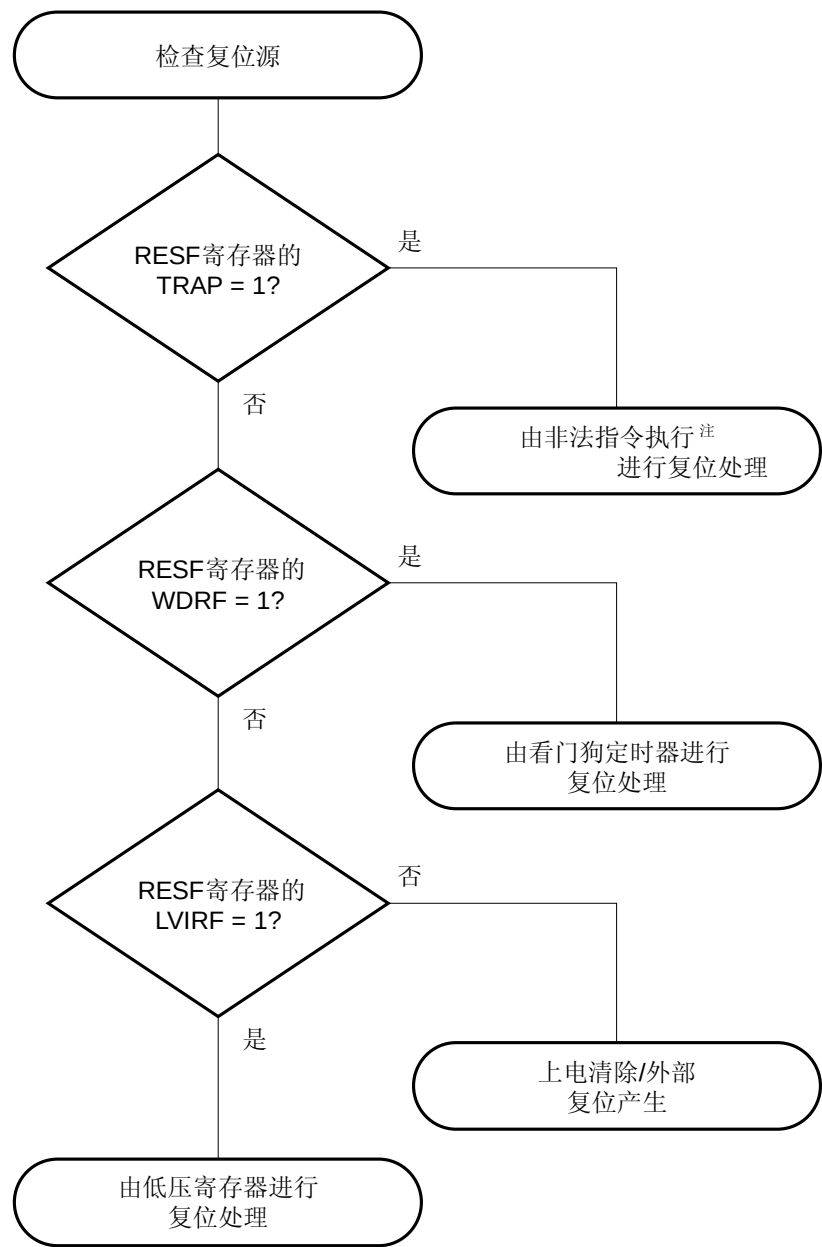


注 流程图在下页。

- 备注
1. 如果低电压检测寄存器 (LVIM) 的位 2 (LVISEL) 设置为“1”，则以上描述需进行如下更改。
 - 供应电压 (V_{DD}) → 从外部输入引脚的输入电压 (EXLVI)
 - 检测电压 (V_{LVI}) → 检测电压 (V_{EXLVI} = 1.21 V)
 2. n = 0 至 7

图 21-11. 复位解除后软件处理举例（2/2）

• 检查复位源



注 执行指令代码 FFH 时。
在线仿真器或片上调试仿真器不能产生通过执行非法指令的复位。

备注 如果低电压检测寄存器（LVIM）的位 2（LVISEL）设置为“1”，则以上描述需进行如下更改。

- 供应电压（V_{DD}） → 从外部输入引脚的输入电压（EXLVI）
- 检测电压（V_{LVI}） → 检测电压（V_{EXLVI} = 1.21 V）

操作举例 2：用作中断时

可能频繁产生中断请求。
进行以下操作。

<操作>

用低电压检测寄存器 (LVIM) 的位 0 (LVIF) 在 LVI 中断的服务例程中，当检测到 V_{DD} 的下降沿时确认“供应电压 (V_{DD}) $\geq$ 检测电压 (V_{LVI})”；或者当检测到 V_{DD} 的上升沿时确认“供应电压 (V_{DD}) $<$ 检测电压 (V_{LVI})”。清除中断请求标志寄存器 0L (IF0L) 的位 1 (LVIIF) 为 0。

对于供应电压有长时间在 LVI 检测电压附近波动的系统，在等待供应电压波动时间后，执行以上操作。

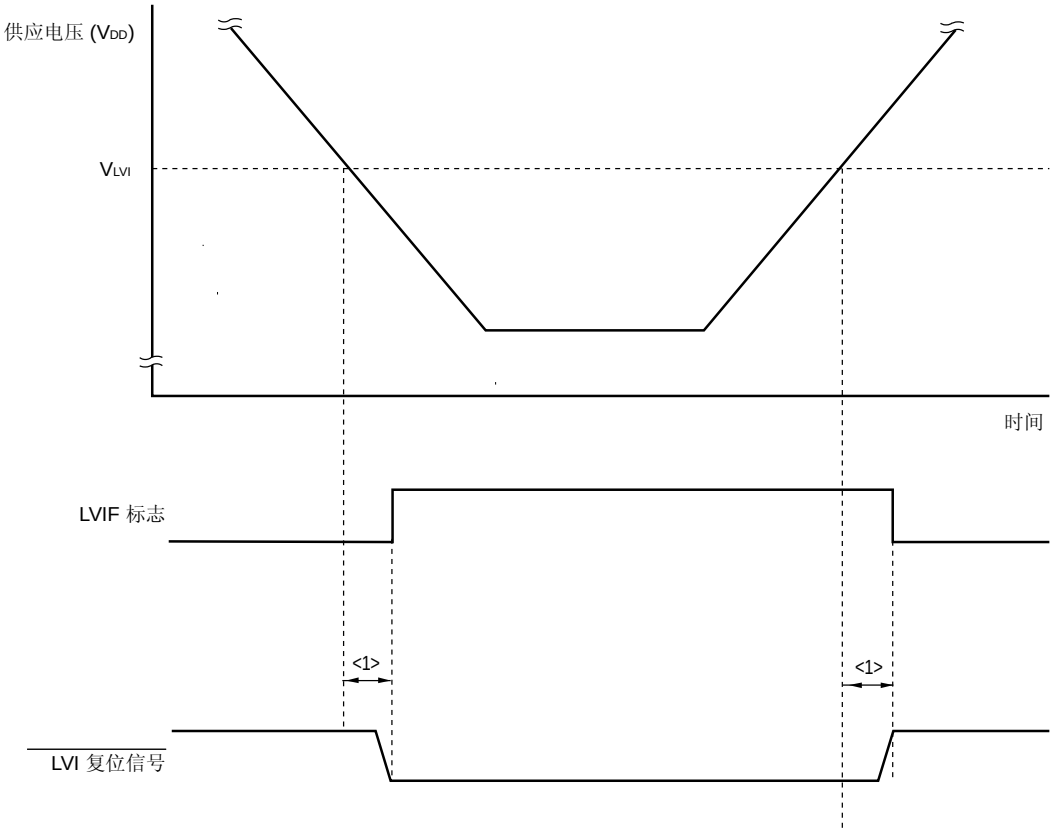
备注 如果低电压检测寄存器 (LVIM) 的位 2 (LVISEL) 设置为“1”，则以上描述需进行如下更改。

- 供应电压 (V_{DD}) $\rightarrow$ 从外部输入引脚的输入电压 ($EXLVI$)
- 检测电压 (V_{LVI}) $\rightarrow$ 检测电压 ($V_{EXLVI} = 1.21\text{ V}$)

(2) 从 LVI 复位源产生时刻到 LVI 复位产生或解除时刻的延迟

从供应电压 (V_{DD}) $<$ LVI 检测电压 (V_{LVI}) 时刻到 LVI 复位产生的时刻有一段延迟。
同样，从 LVI 检测电压 (V_{LVI}) $\leq$ 供应电压 (V_{DD}) 到 LVI 复位解除的时刻也由一段延迟（参见图 21-12）。

图 21-12. 从 LVI 复位源产生时刻到 LVI 复位产生或解除时刻的延迟



<1>: 最小脉冲宽度 (200 μ s (MIN.))

<R>

第二十二章 稳压器

22.1 稳压器概述

78K0R/Kx3-L 包括一个使器件用恒定电压工作的电路。此时，为稳定稳压器输出电压，将 REGC 引脚通过一个电容（0.47 至 1 μF ：目标）连接到 Vss。但是，当使用已进入的 STOP 模式时，由于内部高速振荡时钟和外部主系统时钟的操作，建议用 0.47 μF 电容。要使用具有良好特性的电容，因为其将用于稳定内部电压。

稳压器输出电压正常为 2.4 V（典型值），在低功耗模式下为 1.8 V（典型值）。

22.2 控制稳压器的寄存器

(1) 稳压器模式控制寄存器 (RMC)

该寄存器设置稳压器的输出电压。

用 8 位存储操作指令设置 RMC。

复位输入将该寄存器设置为 00H。

图 22-1. 稳压器模式控制寄存器 (RMC) 的格式

地址: F00F4H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
RMC								

RMC[7: 0]	稳压器输出电压的控制
5AH	固定为低耗模式（1.8 V）
00H	根据条件转换正常电流模式（2.4 V）和低耗模式（1.8 V）（参见 表 22-1）
其它	禁止设置

- 注意事项**
1. 仅在低耗模式下可重写 **RMC** 寄存器（参见表 22-1）。也就是说，在 **CPU** 工作于副系统时钟（**f_{XT}**），而高速系统时钟（**f_{MX}**）、高速内部振荡时钟（**f_{IH}**）和 **20 MHz** 内部高速振荡时钟（**f_{IH20}**）都停止期间，重写该寄存器。
 2. 当使用固定为低耗模式的设置时，可在以下情况中使用 **RMC** 寄存器。
 <当 **X1** 时钟选作 **CPU** 时钟时> **f_X ≤ 5 MHz** 和 **f_{CLK} ≤ 1 MHz**
 <当高速内部振荡时钟、外部输入时钟或副系统时钟选作 **CPU** 时钟> **f_{CLK} ≤ 1 MHz**
 3. 在低耗模式下禁止自编程功能。

(注意事项在下页给出。)

注意事项 4. 更改 RMC 寄存器后到更改操作速度模式控制寄存器（OSMC）需要等待一段时间。设置为低耗模式时需用软件等待 2 ms，而设置为正常电流模式时需要等待 10μs，正如以下步骤所描述：

- 当设置为低功耗模式时
 - <1> f_{CLK} 选择为 1 MHz 的频率。
 - <2> 设置 RMC 为 5AH（设置稳压器为低功耗模式）。
 - <3> 等待 2 ms。
 - <4> 分别将 OSMC 的 FLPC 和 FSEL 设置为 1 和 0。
- 当设置为正常电流模式时
 - <1> 设置 RMC 为 00H（设置稳压器为正常电流模式）。
 - <2> 等待 10μs。
 - <3> 改变 OSMC 的 FLPC 和 FSEL。
 - <4> 改变 f_{CLK} 频率。

表 22-1. 稳压器输出电压条件

模式	输出电压	条件
低功耗模式	1.8 V	RESET 引脚复位期间
		STOP 模式下（除 OCD 模式期间）
		在 CPU 工作于副系统时钟（f _{XT} ）期间，当高速系统时钟（f _{MX} ）、高速内部振荡时钟（f _{IH} ）和 20 MHz 内部高速振荡时钟（f _{IH20} ）都停止时
		在 HALT 模式下，当已经设置 CPU 工作于副系统时钟（f _{XT} ）时，高速系统时钟（f _{MX} ）、高速内部振荡时钟（f _{IH} ）和 20 MHz 内部高速振荡时钟（f _{IH20} ）都停止
正常电流模式	2.4 V	其它

23.1 选项字节的功能

78K0R/Kx3-L 的 flash 存储器的地址 000C0H 至 000C3H 组成一个选项字节区域。

选项字节由用户选项字节（000C0H 至 000C2H）和片上调试选项字节（000C3H）组成。

加电或复位并启动时，会自动引用一个选项字节并设置指定的功能。使用产品时，务必使用选项字节设置以下的功能。

自编程期间使用引导区交换操作时，000C0H 至 000C3H 会被 010C0H 至 010C3H 代替。因此，要将与 000C0H 至 000C3H 相同的值设置至 010C0H 至 010C3H。

注意事项 确保将 FFH 设置到 000C2H（使用引导区交换操作时 000C2H/010C2H）。

23.1.1 用户选项字节（000C0H 至 000C2H/010C0H 至 010C2H）

（1）000C0H/010C0H

- 看门狗定时器的操作
 - 操作在 HALT 或 STOP 模式下停止或使能。
- 看门狗定时器间隔时间的设置
- 看门狗定时器的操作
 - 操作停止或使能。
- 看门狗定时器窗口开启周期的设置
- 看门狗定时器间隔中断的设置
 - 用或者不用

注意事项 因为使用引导区交换操作时 000C0H 被 010C0H 代替，所以需将与 000C0H 相同的值设置至 010C0H。

（2）000C1H/010C1H

- 基于复位解除 LVI 的设置（基于加电）
 - 基于复位解除（由 $\overline{\text{RESET}}$ 引脚复位，而不包括由 LVI，POC，WDT 或非法指令引起的复位），LVI 默认为 ON 或 OFF。
- 内部高速振荡器频率的设置
 - 从 1 MHz，8 MHz 或 20 MHz 中选择

注意事项 因为使用引导区交换操作时 000C1H 被 010C1H 代替，所以需将与 000C1H 相同的值设置至 010C1H。

（3）000C2H/010C2H

- 务必设置 FFH，因为这些地址为保留区域。

注意事项 因为在使用引导区交换操作时 000C2H 被 010C2H 所替换，所以将 FFH 设置至 010C2H。

23.1.2 片上调试选项字节（000C3H/ 010C3H）

- 片上调试操作的控制
 - 禁止或使能片上调试功能。
- 片上调试安全 ID 认证失败情况下 flash 存储器中数据的处理
 - 片上调试安全 ID 认证失败情况下，flash 存储器中的数据被擦除或不被擦除。

注意事项 因为使用引导区交换操作时 000C3H 被 010C3H 代替，所以需将与 000C3H 相同的值设置至 010C3H。

23.2 用户选项字节的格式

用户选项字节的格式如下。

图 23-1. 用户选项字节的格式（000C0H/010C0H）（1/2）

地址： 000C0H/010C0H^{※1}

7	6	5	4	3	2	1	0
WDTINIT	WINDOW1	WINDOW0	WDTON	WDCS2	WDCS1	WDCS0	WDSTBYON
WDTINIT	看门狗定时器间隔中断的使用						
0	不使用间隔中断。						
1	当达到溢出时间的 75%时产生间隔中断。						
WINDOW1	WINDOW0	看门狗定时器窗口打开的周期 ^{※2}					
0	0	25%					
0	1	50%					
1	0	75%					
1	1	100%					
WDTON	看门狗定时器计数器的操作控制						
0	禁止计数器操作（复位后计数停止）						
1	允许计数器操作（复位后计数开始）						
WDCS2	WDCS1	WDCS0	看门狗定时器溢出时间 (f _{IL} = 33 kHz （MAX.）)				
0	0	0	2 ¹⁰ /f _{IL} （31.03 ms）				
0	0	1	2 ¹¹ /f _{IL} （62.06 ms）				
0	1	0	2 ¹² /f _{IL} （124.1 ms）				
0	1	1	2 ¹³ /f _{IL} （248.2 ms）				
1	0	0	2 ¹⁵ /f _{IL} （992.9 ms）				
1	0	1	2 ¹⁷ /f _{IL} （3.971 s）				
1	1	0	2 ¹⁸ /f _{IL} （7.943 s）				
1	1	1	2 ²⁰ /f _{IL} （31.17 s）				

图 23-1. 用户选项字节的格式（000C0H/010C0H）（2/2）

地址：000C0H/010C0H^{※1}

7	6	5	4	3	2	1	0
WDTINIT	WINDOW1	WINDOW0	WDTON	WDSCS2	WDSCS1	WDSCS0	WDSTBYON
WDSTBYON	看门狗定时器计数器的操作控制（HALT/STOP 模式）						
0	在HALT/STOP模式下 ^{※2} 计数器操作停止						
1	在 HALT/STOP 模式下计数器操作使能						

- 注
1. 因为使用引导区交换操作时 000C0H 被 010C0H 代替，所以需将与 000C0H 相同的值设置至 010C0H。
 2. 当 WDSTBYON = 0 时窗口打开周期是 100%，而不考虑 WINDOW1 和 WINDOW0 的值。

注意事项 在 flash 存储器和 EEPROM 仿真的自编程期间，看门狗定时器继续操作。处理期间，中断响应时间被延迟。设置溢出时间和窗口大小时需考虑延迟。

备注 f_{IL}：内部低速振荡时钟频率

图 23-2. 用户选项字节的格式（000C1H/010C1H）

地址：000C1H/010C1H^{※1}

7	6	5	4	3	2	1	0
1	1	1	1	1	FRQSEL2	FRQSEL1	LVIOFF
FRQSEL2	FRQSEL1	内部高速振荡器频率					
0	1	8 MHz/20 MHz ^{※2}					
1	0	1 MHz ^{※3}					
其它		禁止设置					
LVIOFF	加电时 LVI 的设置						
0	基于复位解除（基于加电）默认 LVI 为 ON（LVI 默认启动功能使能）						
1	基于复位解除（基于加电）默认 LVI 为 OFF（LVI 默认启动功能停止）						

- 注
1. 因为使用引导区交换操作时 000C1H 被 010C1H 代替，所以需将与 000C1H 相同的值设置至 010C1H。
 2. 当选择 8 MHz 或 20 MHz 时，在复位解除后，8 MHz 内部高速振荡器自动开始振荡。若要使用 20 MHz 内部高速振荡器以操作微控制器，则需通过设置 20 MHz 内部高速振荡控制寄存器（DSCCTL）的位 0（DSCON）为 1，使振荡在 V_{DD} ≥ 2.7 时启动。微控制器工作时该电路不能改变为的 1 MHz 内部高速振荡器。
 3. 当选择 1 MHz 时，在复位解除后微控制器工作于 1 MHz 内部高速振荡器。微控制器工作时该电路不能改变为的 8 MHz 内部高速振荡器。

（注意事项在下页给出）

- 注意事项 1. 务必将 7 至 3 位设置为 1。
2. 当使用 LVI 默认启动功能时，若用软件将其设置为 LVI 操作禁止时，其操作如下：
- LVION = 0 期间不执行低压检测。
 - LVION = 0 若产生复位，则在复位解除后当 CPU 启动时需将 LVION 重新设置为 1。但是，由于 WDT 和执行非法指令而发生复位时，有一时间段不能正常执行低压检测。
- 这是因为由 LVI 检测的脉冲宽度必需为 200 μ s max.，基于复位发生 LVION = 1 被设置，而在未等待 LVI 稳定时间的情况下 CPU 开始工作。

图 23-3. 用户选项字节的格式（000C2H/010C2H）

地址： 000C2H/010C2H*

7	6	5	4	3	2	1	0
1	1	1	1	1	1	1	1

注 务必将 FFH 设置到 000C2H，因为这些地址为保留区域。因为在使用引导区交换操作时 000C2H 被 010C2H 所替换，所以将 FFH 设置至 010C2H。

23.3 片上调试选项字节的格式

片上调试选项字节的格式如下。

图 23-4. 片上调试选项字节的格式（000C3H/010C3H）

地址： 000C3H/010C3H*

7	6	5	4	3	2	1	0
OCDENSET	0	0	0	0	1	0	OCDERSD

OCDENSET	OCDERSD	片上调试操作的控制
0	0	禁止片上调试操作。
0	1	禁止设置
1	0	由于使能片上调试和认证片上调试安全 ID 失败，擦除 flash 存储器的数据。
1	1	由于使能片上调试和认证片上调试安全 ID 失败，不擦除 flash 存储器的数据。

注 因为使用引导区交换操作时 000C3H 被 010C3H 代替，所以需将与 000C3H 相同的值设置至 010C3H 。

注意事项 位 7 和 0（OCDENSET 和 OCDERSD）只能指定一个值。

务必将位 6 至 1 位设置为 000010B。

备注 当使用片上调试功能时，位 3 至 1 的值被额外写入，所以在设置后该值变得不稳定。

但是，确保将默认值（0， 1 和 0）设置到位 3 至 0。

23.4 选项字节的设置

除在源程序中描述外，用户选项字节和片上调试选项字节可使用 RA78K0R 或 PM+链接器选项进行设置。这样做时，由链接器选项设置的内容优先，即使在源程序中存在描述，与以下涉及内容相同。

关于如何设置链接器选项参见 RA78K0R 汇编器程序包用户手册。

以下为选项字节设置的软件描述举例。

OPT	CSEG	OPT_BYTE	
	DB	10H	； 不使用看门狗定时器的间隔中断 ； 使能看门狗定时器操作 ； 看门狗定时器的窗口打开周期为 25%， ； 看门狗定时器的溢出时间为 $2^{10}/f_{IL}$ ， ； 在 HALT/STOP 模式中停止看门狗定时器操作
	DB	0FBH	； 为内部高速振荡器选择 8 MHz 或 20 MHz ； 停止 LVI 默认启动功能
	DB	0FFH	； 保留区域
	DB	85H	； 使能片上调试操作，不擦除 flash 内存 ； 安全 ID 认证失败时的数据

在自编程期间使用引导区交换功能时，000C0H 至 000C3H 被交换为 010C0H 至 010C3H。因此，要将与 000C0H 至 000C3H 相同的值描述至 010C0H 至 010C3H 如下。

OPT2	CSEG	AT	010C0H
	DB		10H ； 不使用看门狗定时器间隔中断 ； 使能看门狗定时器操作， ； 看门狗定时器的窗口打开周期为 25%， ； 看门狗定时器的溢出时间为 $2^{10}/f_{IL}$ ， ； HALT/STOP 模式下停止看门狗定时器操作
	DB		0FBH ； 为内部高速振荡器选择 8 MHz 或 20 MHz ； 停止 LVI 默认启动功能
	DB		0FFH ； 保留区域
	DB		85H ； 使能片上调试操作，不擦除 flash 内存 ； 安全 ID 认证失败时的数据

注意事项 若要用汇编语言指定选项字节，将 OPT_BYTE 用作 CSEG 伪指令的重置属性名称。若要在 010C0H 至 010C3H 指定选项字节以使用引导区交换功能，用重置属性 AT 指定绝对地址。

第二十四章 FLASH 存储器

78K0R/Kx3-L 包括 flash 存储器，将其安装在板上后程序可被写入、擦除和重写。

24.1 用 Flash 存储器编程器写入

通过使用专用的存储器编程器，数据可被写入到在板或脱板的 flash 存储器。

(1) 在板编程

在将 78K0R/Kx3-L 安装到目标系统后，flash 存储器的内容可被重写。连接专用 flash 存储器编程器的连接器必须安装在目标系统上。

(2) 脱板编程

在将 78K0R/Kx3-L 安装到目标系统上前，通过专用的编程适配器（FA 系列），数据可被写入 flash 存储器。

备注 FA 系列是 Naito Densai Machida Mfg. Co., Ltd. 的产品。

表 24-1. 78K0R/Kx3-L 和专用 Flash 存储器编程器之间的接线

专用 Flash 存储器编程器的引脚配置			引脚名称	引脚编号				
				KC3-L 44 引脚	KC3-L 48 引脚	KD3-L	KE3-L	
信号名称	I/O	引脚功能		LQFP (10x10)	TQFP (7x7)	LQFP (10x10)	LQFP (12x12), LQFP (10x10), TQFP (7x7)	FBGA (5x5)
SI/RxD	输入	接收信号	TOOL0/P40	2	39	4	5	D6
SO/TxD	输出	发送信号						
SCK	输出	传输时钟	—	—	—	—	—	—
CLK	输出	时钟输出	—	—	—	—	—	—
/RESET	输出	复位信号	$\overline{\text{RESET}}$	3	40	5	6	E7
FLMD0	输出	模式信号	FLMD0	6	43	8	9	E8
V _{DD}	I/O	V _{DD} 电压产生/ 电源监控	V _{DD}	11	48	13	15	B7
			EV _{DD}	—	—	—	16	A8
			AV _{REF}	32	23	38	47	G1
GND	—	接地	V _{SS}	10	47	12	13	C7
			EV _{SS}	—	—	—	14	B8
			AV _{SS}	33	24	39	48	H1

注 开发中

用适配器进行 flash 存储器写入时，建议连接举例如下。

图 24-1. 进行 Flash 存储器写入的适配器连线举例（78K0R/KC3-L44 引脚产品）

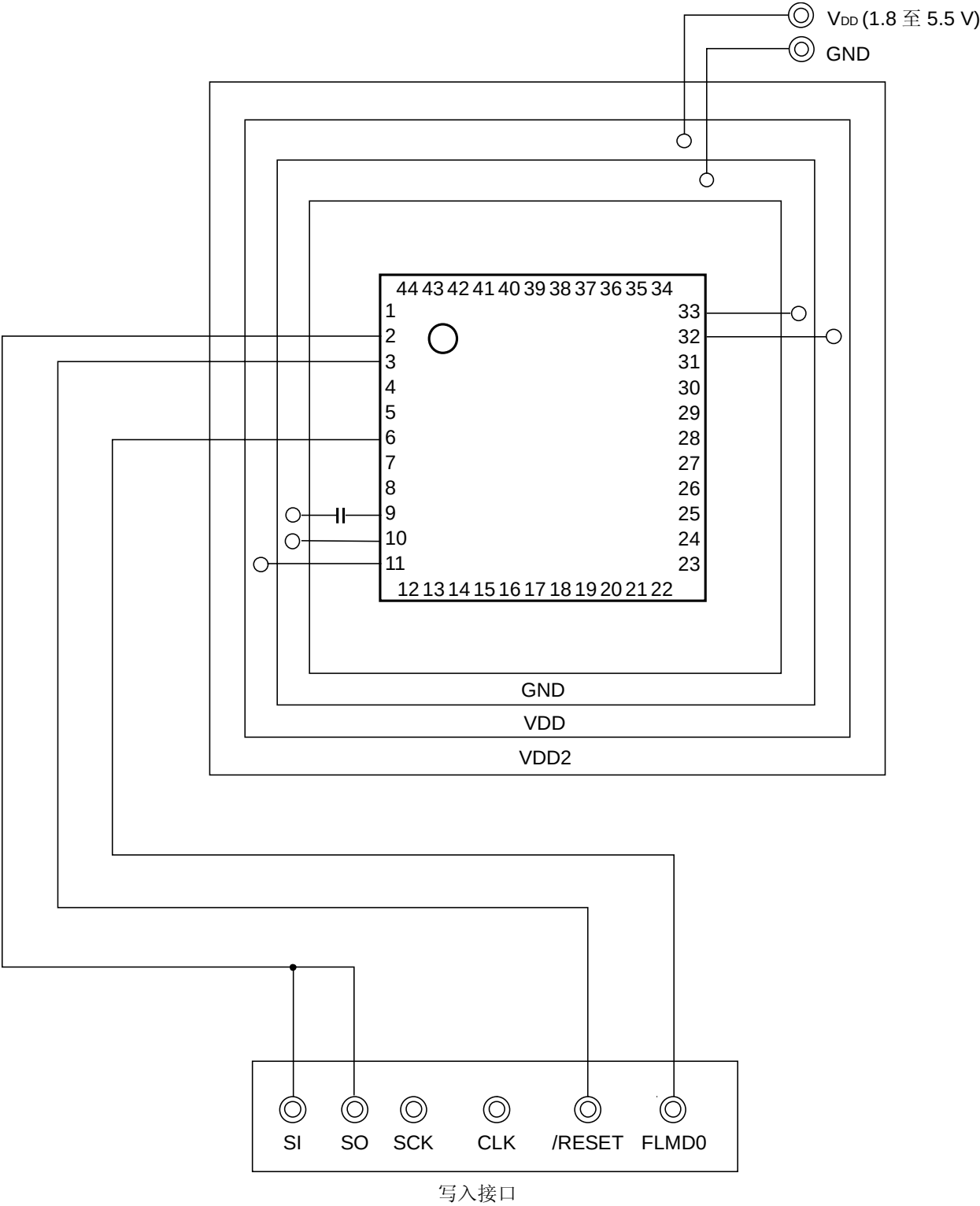


图 24-2. 进行 Flash 存储器写入的适配器连线举例（78K0R/KC3-L48 引脚产品）

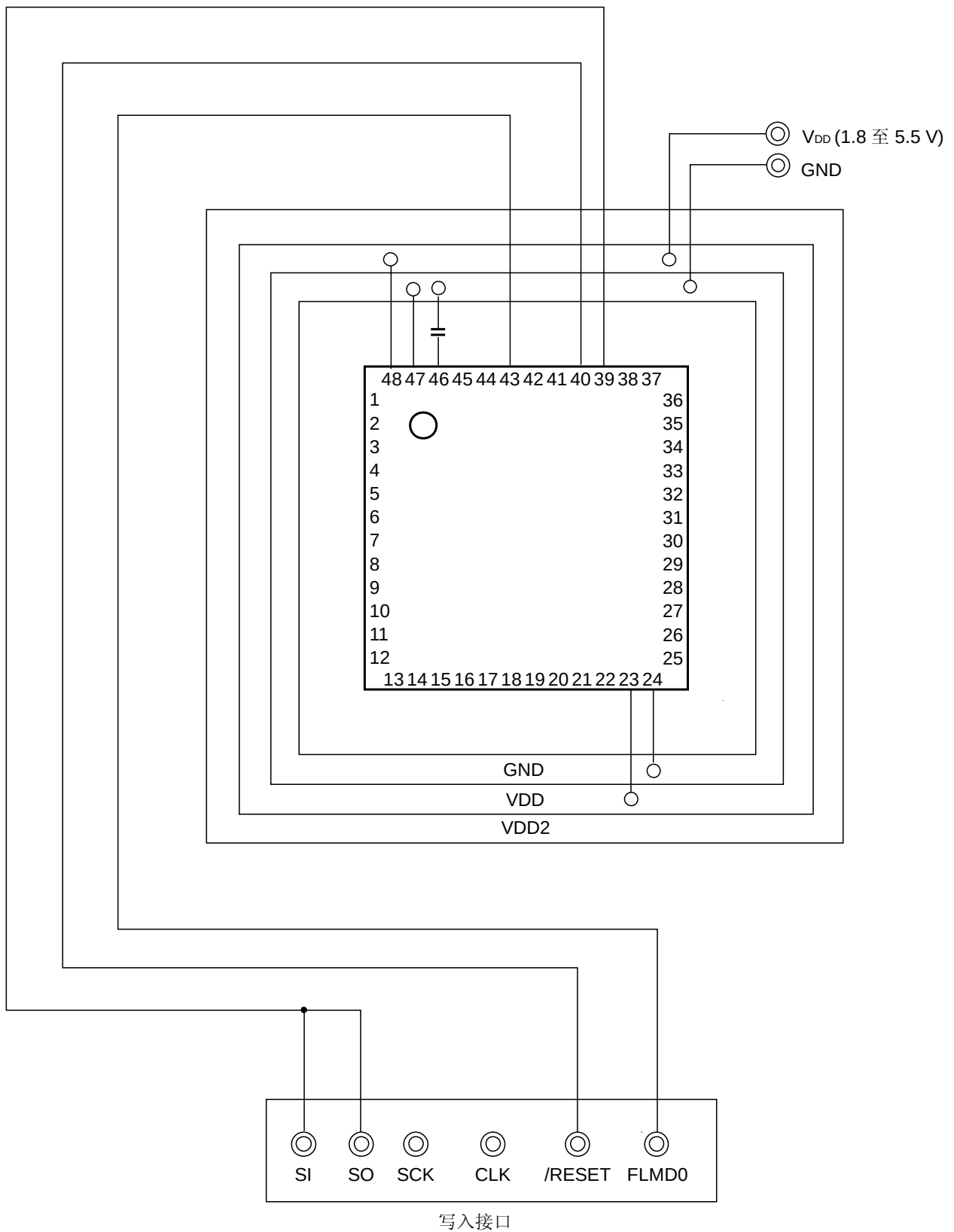


图 24-3. 进行 Flash 存储器写入的适配器连线举例（78K0R/KD3-L）

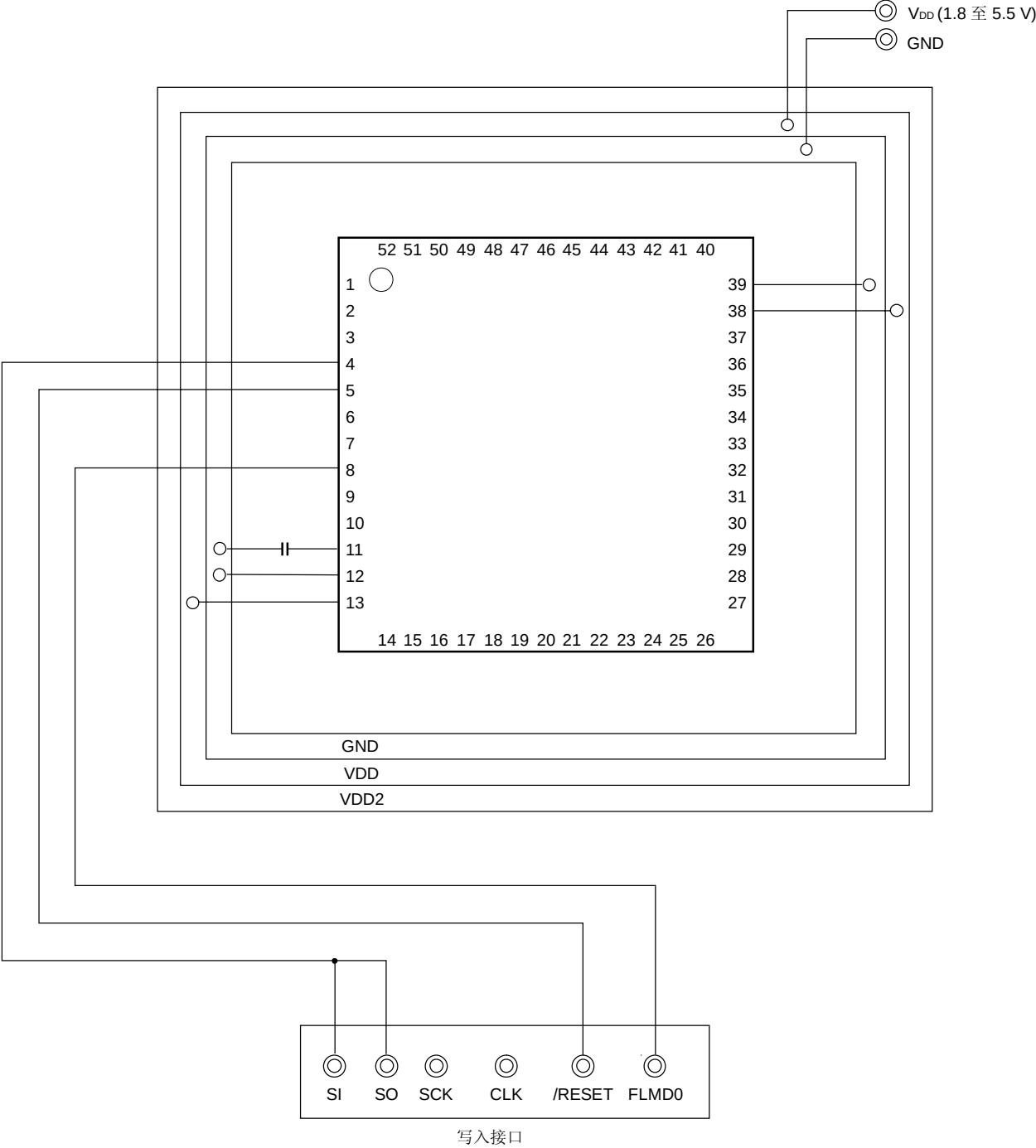
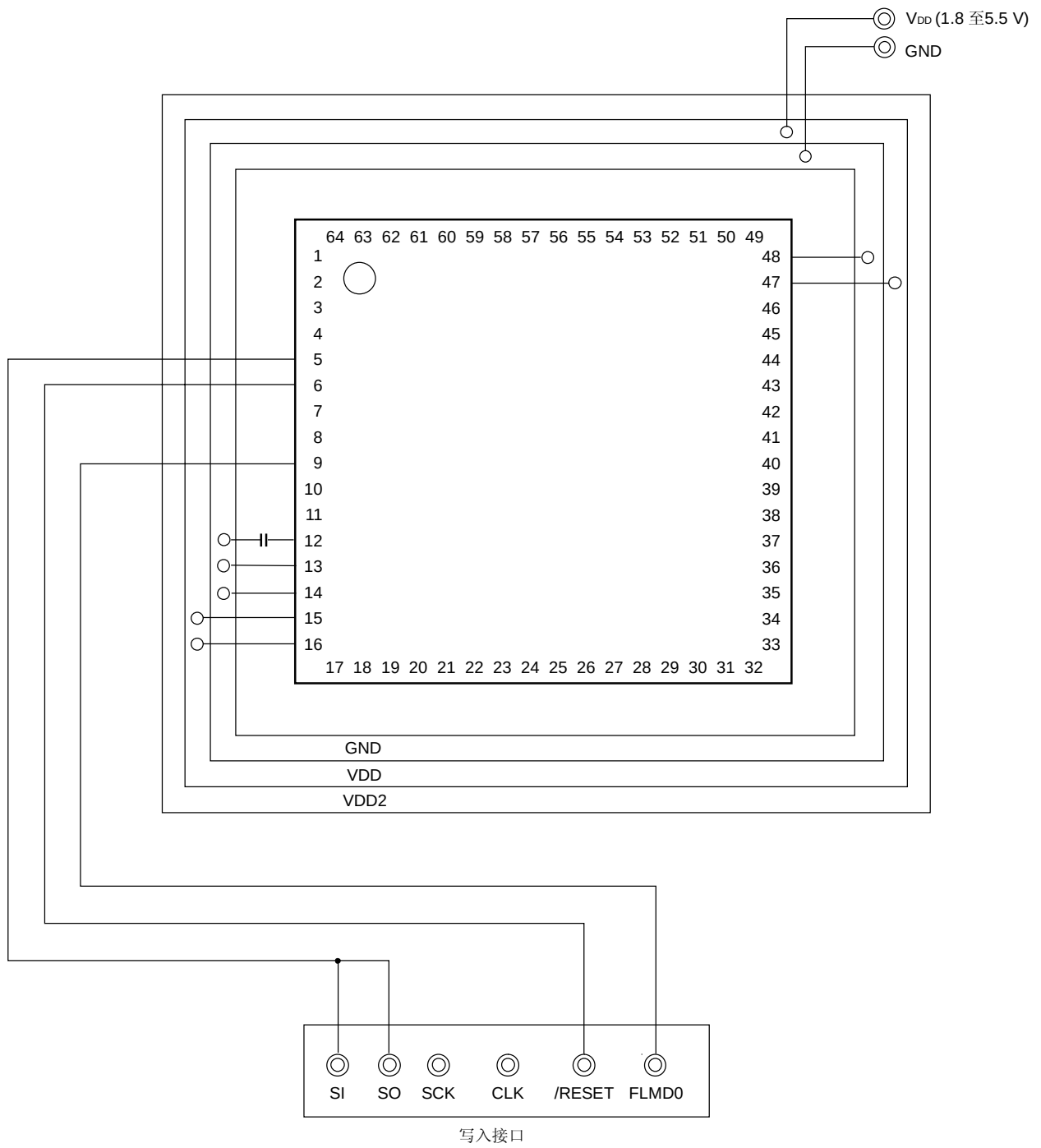


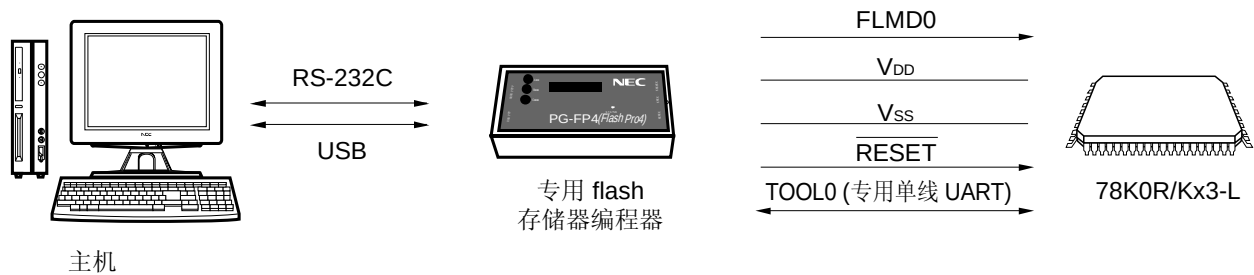
图 24-4. 进行 Flash 存储器写入的适配器连线举例（78K0R/KE3-L）



24.2 编程环境

将程序写入 78K0R/Kx3-L 的 flash 存储器所需要的环境如下所示。

图 24-5. 将程序写入 Flash 存储器的环境



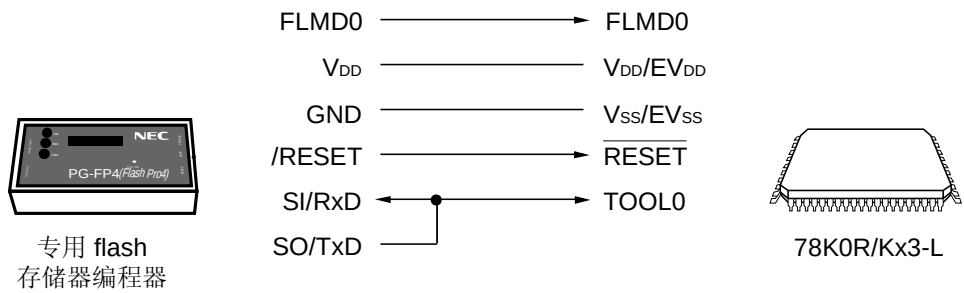
需要一台控制专用 flash 存储器编程器的主机。
若要连接专用 flash 存储器编程器和 78K0R/Kx3-L，需通过专用单线 UART，TOOL0 用于写入或擦除之类的处理。
若要写入脱板 flash 存储器，需要专用编程适配器（FA 系列）。

24.3 通信模式

用 TOOL0 引脚通过 78K0R/Kx3-L 的专用单线 UART 的串行通信，建立专用 flash 存储器编程器和 78K0R/Kx3-L 之间的通信。

传输速率： 115, 200 bps 至 1, 000, 000 bps

图 24-6. 与专用 Flash 存储器编程器的通信



当用 FlashPro4 作为专用 flash 存储器编程器时，FlashPro4，FlashPro5 向 78K0R/Kx3-L 产生以下信号。详情参见 FlashPro4，FlashPro5 的用户手册。

表 24-2. 引脚连接

FlashPro4, FlashPro5			78K0R/Kx3-L	连接
信号名称	I/O	引脚功能	引脚名称	
FLMD0	输出	模式信号	FLMD0	⊙
V _{DD}	I/O	V _{DD} 电压产生/电源监控	V _{DD} , EV _{DD} , AV _{REF}	⊙
GND	—	接地	V _{SS} , EV _{SS} , AV _{SS}	⊙
CLK	输出	时钟输出	—	×
/RESET	输出	复位信号	RESET	⊙
SI/RxD	输入	接收信号	TOOL0	⊙
SO/TxD	输出	发送信号		
SCK	输出	传输时钟	—	×

备注 ⊙: 必须连接的引脚。

 ×: 不必连接的引脚。

24.4 板上引脚的连接

若要写入板上 flash 存储器，则连接专用 flash 存储器编程器的连接器必须在目标系统上提供。首先提供一种在板功能，即选择正常操作模式或 flash 存储器编程模式。

当 flash 存储器编程模式被设置时，所有那些不用于编程 flash 存储器的引脚状态与复位后的状态相同。因此，如果外部设备不能在复位后立即识别状态，这些引脚必须进行如下处理。

24.4.1 FLMD0 引脚

(1) Flash 存储器编程模式中

当数据被 flash 存储器编程器写入时，将该引脚直接连接到 flash 存储器编程器。这将向 FLMD0 引脚提供一个 V_{DD} 电平的写入电压。

因为 FLMD0 引脚被复位进行内部下拉，所以不必对其进行外部下拉。若要将其进行外部下拉，使用一个 1 kΩ 至 200 kΩ 的电阻。

(2) 正常操作模式中

正常操作中建议将该引脚保持开路。

FLMD0 引脚在复位解除前必须总保持在 V_{SS} 电平，但因为其被复位进行内部下拉，所以不必对其进行外部下拉。但是，用后台事件控制寄存器 (BECTL) 的位 7 (FLMDPUP)，必须保持选择将其下拉（例如 FLMDPUP = “0”，默认值）（参见 24.5 (1) 后台事件控制寄存器）。若要将其进行外部下拉，使用一个 200kΩ 或更小的电阻。

通过将该引脚直接连接到 V_{SS} 引脚，硬件可禁止用编程器的 flash 存储器的自编程和重写。

(3) 自编程模式下

使用自编程功能时建议将该引脚保持开路。若要将其进行外部下拉，使用一个 100 k Ω 至 200 k Ω 的电阻。
自编程模式下，在自编程库中该设置转换为上拉。

图 24-7. FLMD0 引脚连接举例



24.4.2 TOOL0 引脚

在 flash 存储器编程模式下，直接将该引脚连接到专用 flash 存储器编程器，或通过外部电阻将其连接到 EV_{DD} 而对其进行上拉。

在正常操作模式下允许片上调试时，将该引脚经过一外部电阻连接到 EV_{DD} 而对其进行上拉，在复位解除前，务必保持 V_{DD} 电平输入到 TOOL0 引脚（禁止下拉该引脚）。

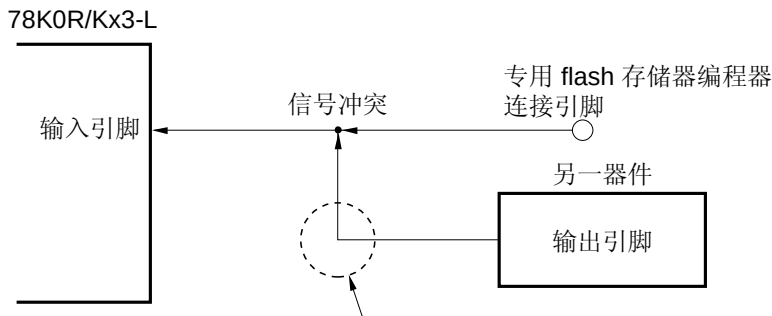
备注 因为使用单线 UART，所以 SAU 和 IICA 引脚不用于 78K0R/Kx3-L 和专用 flash 存储器编程器之间的通信。

24.4.3 RESET 引脚

如果专用 flash 存储器编程器的复位信号连接到 RESET 引脚，而 RESET 引脚被连接到板上复位信号发生器，则将发生信号冲突。若要阻止该冲突，则需隔离与复位信号发生器的连接。

如果复位信号从用户系统输入，而 flash 存储器编程模式被设置，则 flash 存储器将不能被正确编程。除了来自于专用 flash 存储器编程器的复位信号以外，不要输入其它信号。

图 24-8. 信号冲突 (RESET 引脚)



在 flash 存储器编程模式下，由另一器件输出的信号将与专用 flash 存储器编程器输出的信号冲突。因此，需隔离另一器件。

24.4.4 端口引脚

当flash 存储器编程模式被设置时，所有那些不用于flash 存储器存储器编程的引脚状态与复位后的状态相同。如果连接到端口的外部设备不能在复位后立即识别端口状态，则端口引脚必须经过电阻连接到EVDD^注或 EVSS^注。

注 没有 EVSS 引脚的产品，将其连接到 Vss。没有 EVDD 引脚的产品，将其连接到 VDD。

24.4.5 REGC 引脚

正常操作期间，用相同方法通过电容将 REGC 引脚连接到 GND（0.47 至 1 μ F：目标）。但是，当使用已进入的 STOP 模式时，由于内部高速振荡时钟和外部主系统时钟的操作，建议用 0.47 μ F 电容。要使用具有良好特性的电容，因为其将用于稳定内部电压。

24.4.6 X1 和 X2 引脚

按与正常操作模式相同的状态连接 X1 和 X2。

备注 在 flash 存储器编程模式下，使用内部高速振荡时钟（f_{IH}）。

24.4.7 电源供应

若要使用 flash 存储器编程器的供应电压输出，则需将 VDD 引脚连接到 flash 存储器编程器的 VDD，将 VSS 引脚连接到 flash 存储器编程器的 GND。

若要使用板上供应电压，按照正常操作模式中的连接。

但是，当使用板上供应电压时，务必将 VDD 和 VSS 引脚连接到 flash 存储器编程器的 VDD 和 GND，以使用 flash 存储器编程器的电源监控器功能。

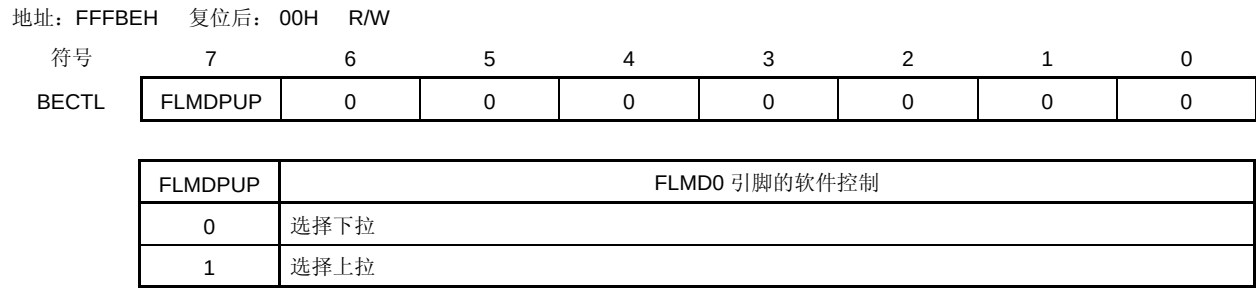
提供与正常操作模式下相同的其它电源供应（EVDD，EVSS，AVREF 和 AVSS）。

24.5 控制 Flash 存储器的寄存器

(1) 后台事件控制寄存器（BECTL）

即使 FLMD0 引脚不被外部控制，但是它也可用 BECTL 寄存器通过软件控制，将其设置为自编程模式。但是，根据 FLMD0 引脚的处理，其有可能不被软件设置为自编程模式。当使用 BECTL 时，建议将 FLMD0 引脚保持开路。当将其进行外部下拉时，用 100 k 或更大的电阻。此外，正常模式下，用 BECTL 进行下选择。自编程模式下，在自编程库中该设置转换为上拉。
可通过 1 位或 8 位的存储操作指令设置 BECTL 寄存器。
复位输入将该寄存器设置为 00H。

图 24-9. 后台事件控制寄存器（BECTL）的格式

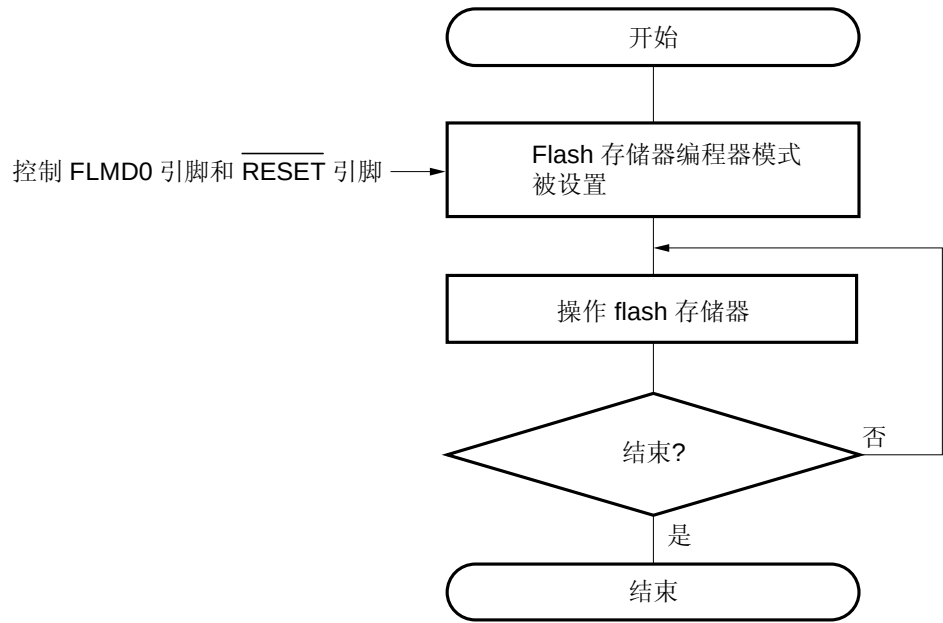


24.6 编程方法

24.6.1 控制 flash 存储器

下图显示了操作 Flash 存储器的流程:

图 24-10. Flash 存储器操作步骤



26.6.2 Flash 存储器编程模式

若要用专用 flash 存储器编程器重写 flash 存储器的内容，则需在 flash 存储器编程模式下设置 78K0R/Kx3-L。要设置该模式，需设置 FLMD0 引脚和 TOOL0 引脚为 V_{DD} 并清除复位引脚。
当进行板上 flash 存储器写入时，用跳线改变模式。

图 24-11. Flash 存储器编程模式

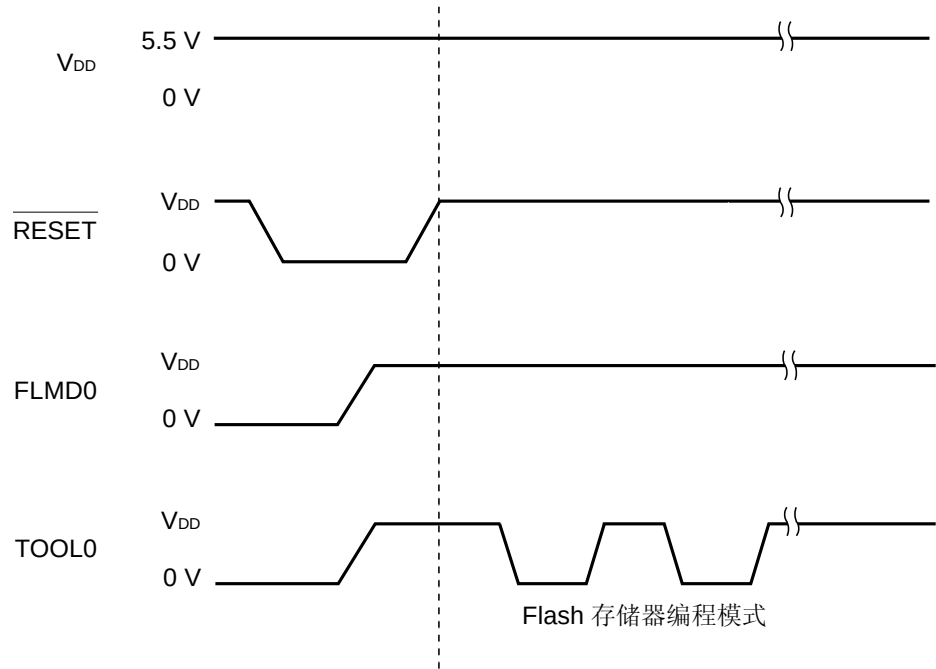


表 24-3. 复位解除后 FLMD0 引脚和操作模式之间的关系

FLMD0	操作模式
0	正常操作模式
V _{DD}	Flash 存储器编程模式

24.6.3 选择通信模式

78K0R/Kx3-L 的通信模式如下。

表 24-4. 通信模式

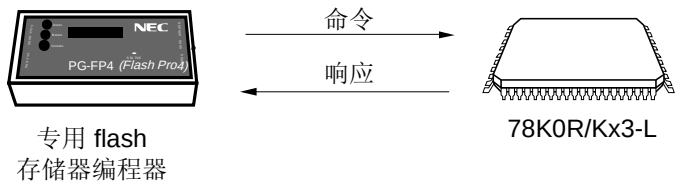
通信模式	标准设置 ¹				使用引脚
	端口	速度	频率	倍频系数	
1 线模式 (专用单线 UART)	UART-ch0	1 Mbps ²	—	—	TOOL0

注 1. 在 flash 存储器编程器的 GUI 上为标准设置选择项目。
2. 因为除波特率误差外其它因素如信号波形突变，同样会影响 UART 通信，所以与波特率误差一样需要对突变进行完全评估。

24.6.4 通信命令

78K0R/Kx3-L 与专用 flash 存储器编程器通过命令进行通信。从 flash 存储器编程器发出的信号称为命令，从 78K0R/Kx3-L 发出的信号称为响应。

图 24-12. 通信命令



78K0R/Kx3-L 的 flash 存储器控制命令列于下表。所有的命令都从编程器发出，78K0R/Kx3-L 执行相应于各命令的处理。

表 24-5. Flash 存储器控制命令

类别	命令名称	功能
验证	验证	将 flash 存储器指定区域中的内容与由编程器发送的数据相比较。
擦除	片擦除	擦除整个 flash 存储区域。
	块擦除	擦除 flash 存储器指定区域。
空白检查	块空白检查	检查 flash 存储器中指定块是否被正确擦除。
写入	编程	向 flash 存储器指定区域中写入数据。
获取信息	硅签字	获取 78K0R/Kx3-L 信息（例如部件编号和 flash 存储器配置）。
	获取版本信息	获取 78K0R/Kx3-L 固件版本。
	校验和	获取指定区域的校验和数据。
安全性	安全设置	设置安全信息。
其它	复位	用于检测通信的同步状态。
	波特率设置	当 UART 通信模式被选择时，设置波特率。

对于专用 flash 存储器被称为发出的命令，78K0R/Kx3-L 返回一个响应。从 78K0R/Kx3-L 发出的响应名称列表如下。

表 24-6. 响应名称

响应名称	功能
ACK	响应命令/数据
NAK	响应非法命令/数据

24.7 安全设置

78K0R/Kx3-L 支持保护功能，该保护功能禁止重写被写入内部 Flash 存储器中的用户程序，这样，程序就不会被未经授权的人改变。

用安全设置命令可执行以下操作。当编程模式被下一次设置时，安全设置有效。

- 禁止批量擦除（片擦除）

在板/脱板编程中，该设置禁止对 flash 存储器中的整块执行块擦除和批量擦除（片擦除）命令。一旦禁止批量擦除（片擦除）命令的执行，则所有的禁止设置（包括批量擦除（片擦除）的禁止设置）不能再被取消。

注意事项 对批量擦除的安全设置被设置后，不能对设备的执行删除。此外，即使执行一个写入命令，与已经写入 flash 存储器的数据不同的数据不能被写入，这是因为禁止执行擦除命令。

- 禁止块擦除

在板/脱板编程中，禁止执行对 flash 存储器中指定块的块擦除命令。但是，可通过自编程方法对块进行擦除。

- 禁止写入

在板/脱板编程中，禁止执行对 flash 存储器中整块的写入和块擦除命令。但是，可通过自编程方法对块进行写入。

- 禁止重写引导区簇 0

该设置禁止执行批量擦除（片擦除）命令，块擦除命令以及对 flash 存储器中引导区簇 0（00000H 至 00FFFH）的写入命令。

当 flash 存储器出厂时，默认设置为允许批量擦除（片擦除），块擦除，写入命令和重写引导区簇 0。由在板/脱板编程和自编程进行安全设置。每种安全设置都可联合使用。

执行批量擦除（片擦除）命令可清除所有的安全设置。

表 24-7 显示了当使能 78K0R/Kx3-L 安全功能时，擦除和写入命令的关系。

备注 若要在自编程期间禁止写入和擦除，则需使用 flash 屏蔽窗口功能（详情参见 24.8.2）。

表 24-7. 使能安全功能和命令的关系

(1) 在板/脱板编程期间

有效安全设置	执行的命令		
	批量擦除（片擦除）	块擦除	写入
禁止批量擦除（片擦除）	不能被批量擦除	不能擦除块	不能执行 ^注 。
禁止块擦除	能被批量擦除		能执行
禁止写入			不能执行
禁止重写引导区簇 0	不能被批量擦除	不能擦除引导区簇 0。	不能写入引导区簇 0。

注 确认没有数据被写入到写入区。因为禁止批量擦除（片擦除）后，数据不能被擦除，所以如果数据不能被擦除则不能写入数据。

(2) 自编程期间

有效安全设置	执行的命令	
	块擦除	写入
禁止批量擦除（片擦除）	能擦除块	能执行
禁止块擦除		
禁止写入		
禁止重写引导区簇 0	不能擦除引导区簇 0。	不能写入引导区簇 0。

备注 若要在自编程期间禁止写入和擦除，则需使用 flash 屏蔽窗口功能（详情参见 24.8.2）。

表 24-8. 各编程模式下的安全设置

(1) 在板/脱板编程

安全性	安全设置	如何禁止安全设置
禁止批量擦除（片擦除）	由专用 flash 存储器编程器的 GUI 等等设置。	设置后不能被禁止。
禁止块擦除		执行批量擦除（片擦除）命令。
禁止写入		
禁止重写引导区簇 0		设置后不能被禁止。

(2) 自编程

安全性	安全设置	如何禁止安全设置
禁止批量擦除（片擦除）	用信息库设置	设置后不能被禁止。
禁止块擦除		在板/脱板编程期间执行批量擦除（片擦除）命令（自编程期间不能被禁止）。
禁止写入		
禁止重写引导区簇 0		

24.8 用自编程进行 Flash 存储器编程

78K0R/Kx3-L 支持自编程功能，该功能可通过用户程序重写 flash 存储器。因为该功能允许用户程序用 78K0R/Kx3-L 自编程库重写 flash 存储器，所以该功能可在该区域更新程序。

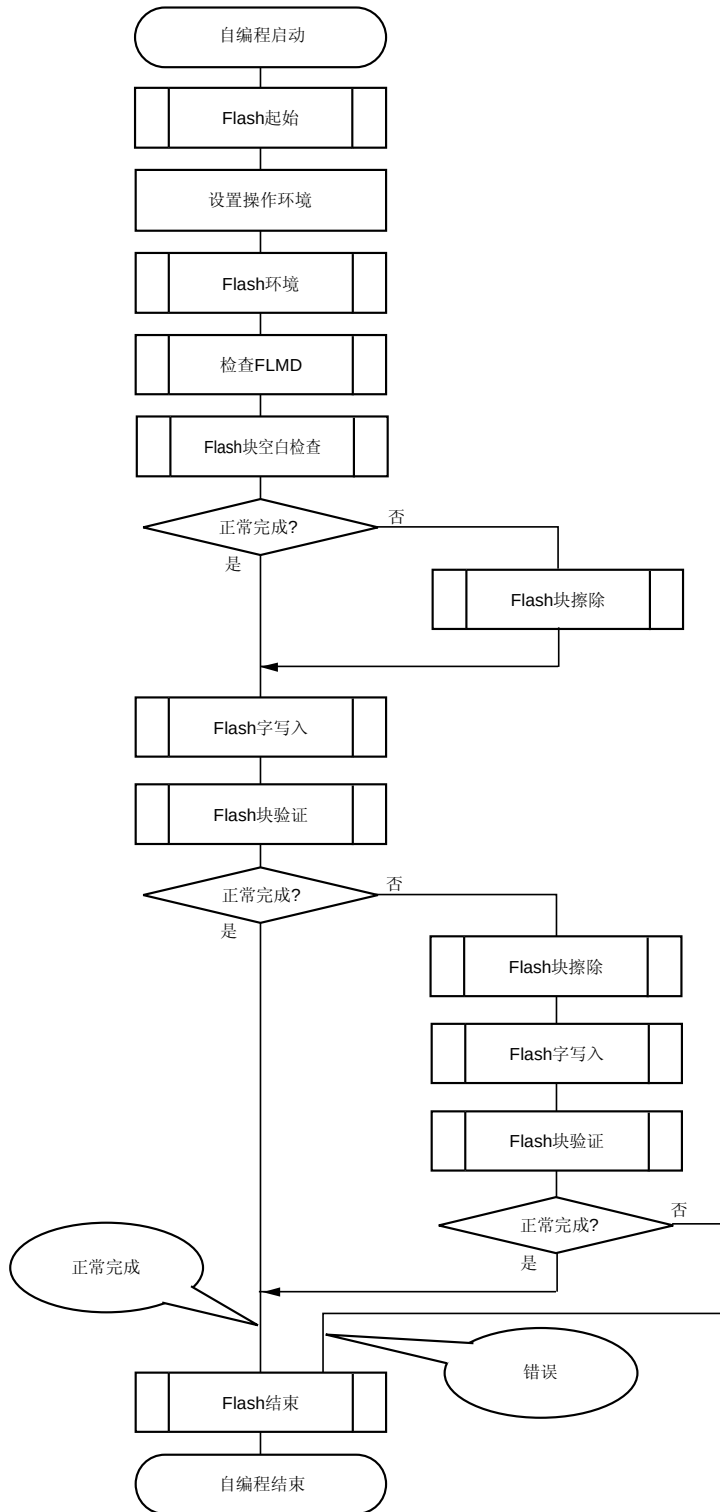
如果自编程期间发生中断，则自编程暂时停止，而执行服务中断。如果在 EI 状态产生不可屏蔽中断请求，则该请求直接从自编程库转移到中断例程。之后自编程模式被恢复后，自编程会继续执行。但是，中断响应时间与正常操作模式下的时间不同。

- 注意事项**
1. 当 CPU 工作于副系统时钟时，不能使用自编程功能。
 2. 自编程模式下，调用自编程起始库（FlashStart）。
 3. 若要在自编程期间禁止中断，与正常操作模式下方法相同，在 IE 标志被 DI 指令清除为（0）的状态下，执行自编程库。若要允许中断，将中断屏蔽标志清零（0），以接收 IE 标志被 EI 指令设置为（1）的状态，然后执行自编程库。
 4. 在低功耗模式下禁止自编程功能。关于低功耗模式的详情，参见第二十二章 稳压器。
 5. 在自编程库功能的执行期间，禁止 DMA 操作（DENn = 0）。

<R>

以下流程图显示了用自编程库进行重写 flash 存储器的流程。

图 24-13. 自编程流程（重写 Flash 存储器）



24.8.1 引导区交换功能

如果由于临时电源失败或其它原因使重写引导区失败，则会因为引导区数据破坏而禁止用复位或重写进行重新启动程序。

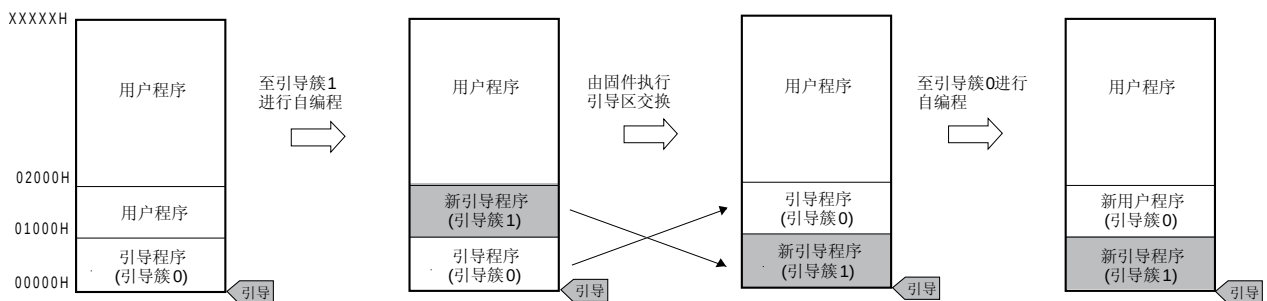
引导区交换功能用于避免这种问题。

擦除引导簇 0[※]即引导程序区之前，通过自编程预先将一个新的引导程序写入引导簇 1 中。当程序被正确写入引导簇 1 时，用 78K0R/Kx3-L 固件的设置功能交换引导簇 1 和引导簇 0，所以引导簇 1 用作引导区。此后，擦除或写入最初的引导程序区即引导簇 0。

结果，即使在写入引导编程区期间发生电源失败，也会正确执行程序，因为当程序被复位并下一次开始时，由被交换的引导簇 1 引导。

注 一个引导簇为 4 KB 区域，引导簇 0 和 1 被引导交换功能进行交换。

图 24-14. 引导交换功能



上图举例中，注释如下。

引导簇 0: 引导交换前的引导程序

引导簇 1: 引导交换后的引导程序

块编号

引导簇 1

引导簇 0

擦除块 4

擦除块 5

擦除块 6

擦除块 7

由引导簇 0 引导

写入块 4 至 7

引导区交换

擦除块 4

擦除块 5

由引导簇 1 引导

擦除块 6

擦除块 7

写入块 4 至 7

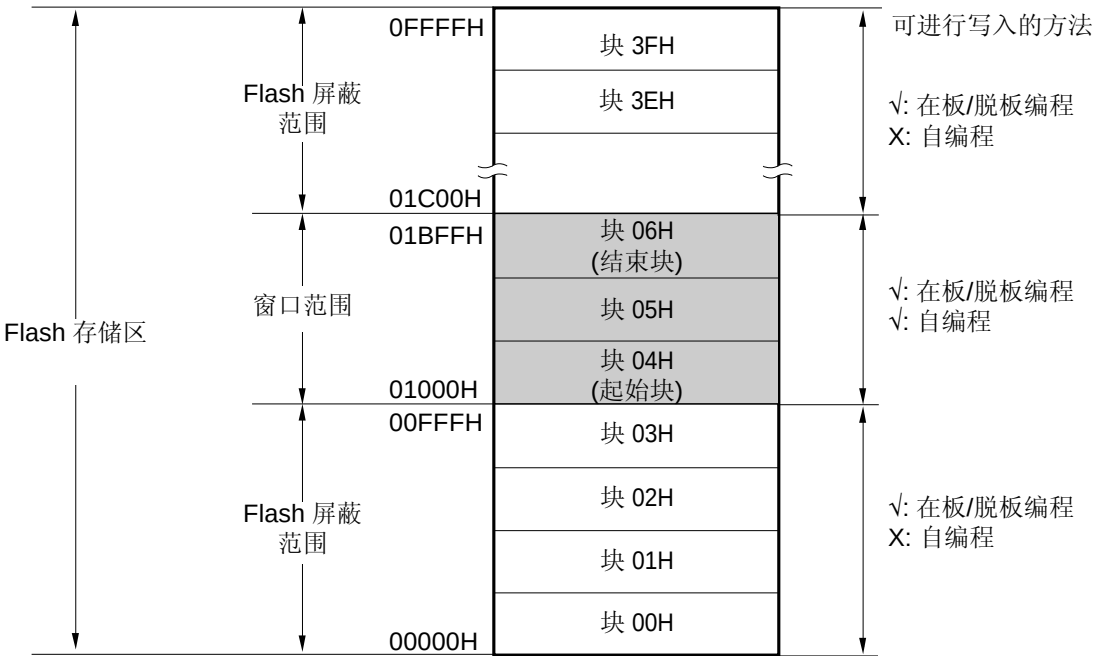
<R> 24.8.2 Flash 屏蔽窗口功能

Flash 屏蔽窗口功能作为一项自编程安全功能提供。该功能禁止写入和擦除指定仅作为自编程期间窗口的范围外的区域。

通过指定起始块和结束块设置窗口范围。在板/脱板编程和自编程期间都可设置或改变窗口范围。

禁止在自编程期间写入和擦除窗口范围外的区域。但是，在板/脱板编程期间，指定为窗口范围外的区域可被写入和擦除。

图 24-16. Flash 屏蔽窗口设置举例
(目标器件: μ PD78F1003, 起始块: 04H, 结束块: 06H)



注意事项 如果引导簇 0 的禁止重写区与 flash 屏蔽窗口范围重叠，则禁止重写引导簇 0 优先。

表 24-9. Flash屏蔽窗口功能设置/改变方法和命令的关系

编程条件	窗口范围设置/改变方法	执行命令	
		块擦除	写入
自编程	如果设置信息库指定起始和结束块。	只有在窗口范围内允许块擦除。	只有在窗口范围内允许写入。
在板/脱板编程	用专用 flash 存储器编程器的 GUI 等等指定起始和结束块。	窗口范围外也允许块擦除。	窗口范围外也允许写入。

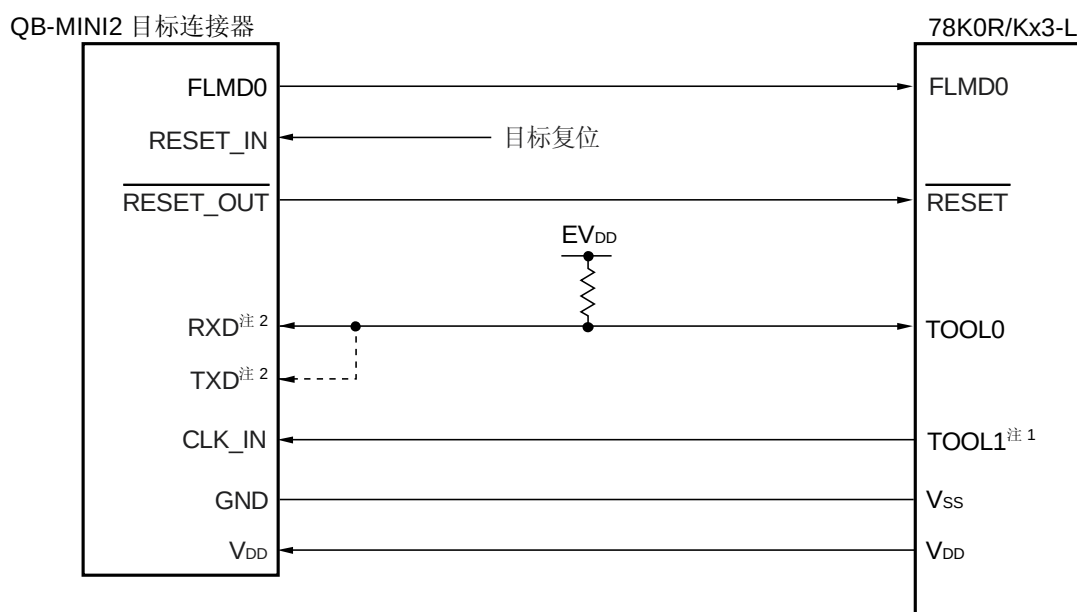
备注 对于禁止在板/脱板编程期间写入/擦除，参见 24.7 安全设置。

25.1 将 QB-MINI2 连接到 78K0R/Kx3-L

78K0R/Kx3-L 用 V_{DD} , FLMD0, $\overline{RESET}$, TOOL0, TOOL1^{注 1} 以及 V_{SS} 引脚通过片上调试仿真器 (QB-MINI2) 与主机通信。

注意事项 78K0R/Kx3-L 具有片上调试功能, 用于开发和评估。请不要在量产时使用该产品的片上调试功能, 因为使用该功能时, 可能会超出 flash 内存能保证的可重写次数, 因此产品当可靠性得不到保证。NEC 电子对使用片上调试功能时产生的问题概不负责。

图 25-1. QB-MINI2 和 78K0R/Kx3-L 的连接举例



- 注**
1. 在 1 线模式中通信不需要连接, 但在 2 线模式中通信需要。此时, 因为当不连接 QB-MINI2 时也不连接 TOOL1, 所以需按照表 2-2 未使用引脚的连接进行必要的连接。
 2. 因为 QB-MINI2 中 RXD 和 TXD 被短路, 所以不需要进行图中虚线连接。当使用另一 flash 存储器编程器时, 该编程器中 RXD 和 TXD 可能不被短路。这种情况下, 它们必须在目标系统中短路。

备注 在片上调试中自编程时建议 FLMD0 引脚开路。若要将其进行外部下拉, 使用一个 100kΩ 或更大的电阻。

使用 TOOL0 引脚的 1 线模式 (单线 UART) 或使用 TOOL0 和 TOOL1 引脚的 2 线模式, 用于串行通信。对于 flash 存储器编程, 使用 1 线模式。1 线模式或 2 线模式用于片上调试。表 25-1 列出了 1 线模式和 2 线模式的区别。

表 25-1. 1 线模式和 2 线模式的区别列表

通信模式	Flash 存储器编程功能	调试功能
1 线模式	可用	- 不支持伪实时 RAM 监控器（RRM）功能。 - 不支持 DMM 功能（在 RUN 中重写存储器）。 - 调试器速度比 2 线模式慢 2 至 4 倍。
2 线模式	无	- 支持伪实时 RAM 监控器（RRM）功能。 - 支持 DMM 功能（在 RUN 中重写存储器）。

备注 但是，2 线模式不用于 flash 编程，即使 TOOL1 引脚用 QB-MINI2 的 CLK_IN 连接，也会正常执行写入。

25.2 片上调试安全 ID

78K0R/Kx3-L 在 flash 存储器 000C3H（参见第二十三章 选项字节）处有片上调试操作控制位，在 000C4H 至 000CDH 位有片上调试安全 ID 设置区，以防止第三方读取存储器内容。

当使用引导区交换功能时，预先设置一个与 010C3H 和 010C4H 相同的值到 010CDH，因为 000C3H，000C4H 至 000CDH 与 010C3H，010C4H 至 010CDH 被交换。

关于片上调试安全 ID，参见带编程功能的 QB-MINI2 片上调试仿真器用户手册（U18371E）。

表 25-2. 片上调试安全 ID

地址	片上调试安全 ID
000C4H 至 000CDH	任何 10 字节 ID 码
010C4H 至 010CDH	

25.3 用户资源保护

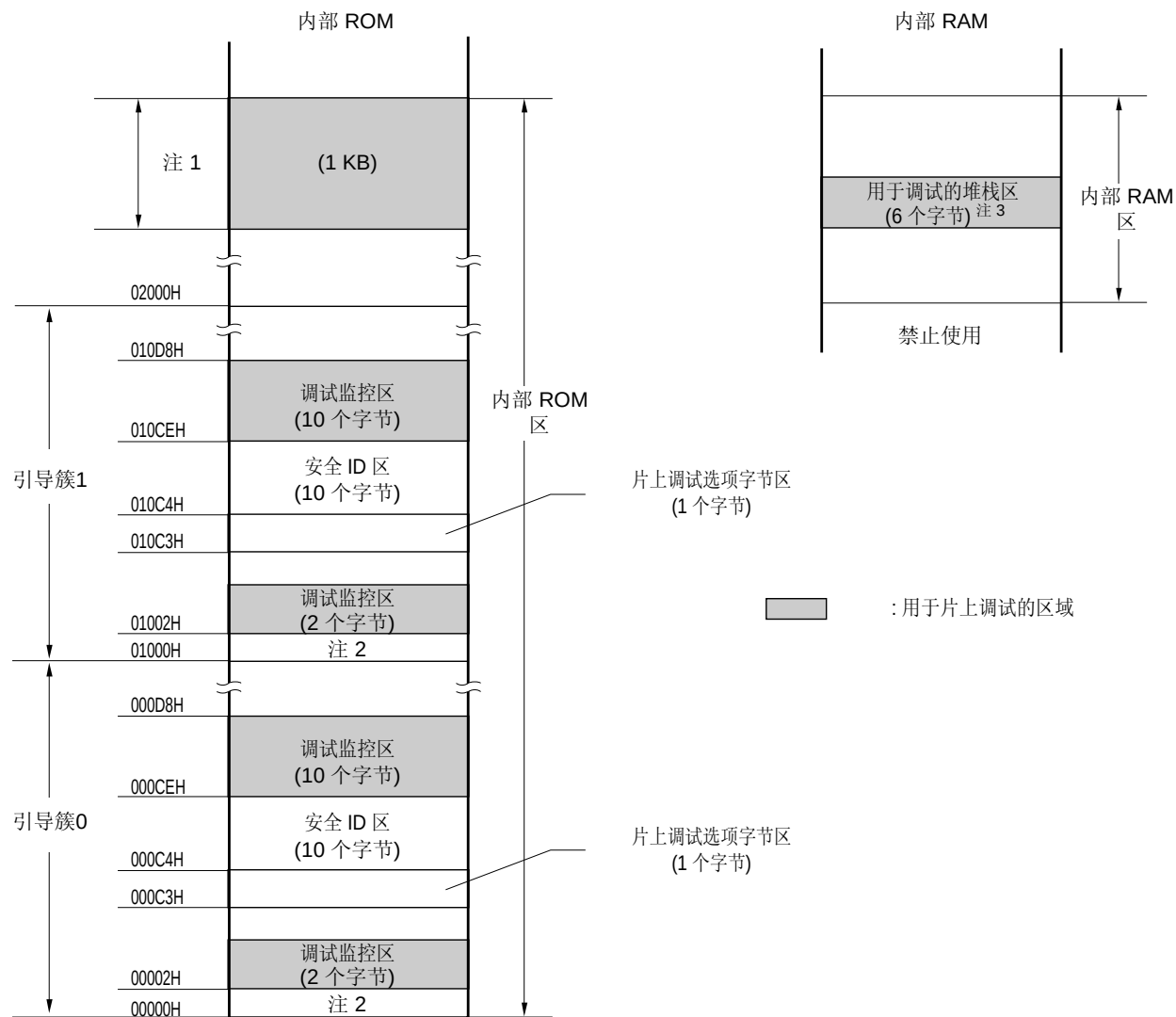
若要进行 78K0R/Kx3-L 和 QB-MINI2 通信，与各调试功能相同，必须预先设置内存空间的安全性。

如果使用 NEC 电子汇编器 RA78K0R 或编译器 CC78K0R，则可用链接器选项设置这些项。

（1）内存空间的安全性

图 25-2 中的阴影部分是用来放置调试监控程序的预留区，因此不能在这些空间分配用户程序或数据。当使用片上调试功能时，必须保护这些空间不被用户程序使用。此外，该区域禁止被用户程序重写。

图 25-2. 调试监控程序的内存空间分配



注 1. 根据产品不同地址有以下区别。

产品 ()：内部 ROM	注 1 的地址
μPD78F1000 (16 KB)	03C00H 至 03FFFH
μPD78F1001, 78F1004, 78F1007 (32 KB)	07C00H 至 07FFFH
μPD78F1002, 78F1005, 78F1008 (48 KB)	0BC00H 至 0BFFFH
μPD78F1003, 78F1006, 78F1009 (64 KB)	0FC00H 至 0FFFFH

- 2. 调试中，复位向量被重写到分配给监控程序的地址。
- 3. 因为该区被分配在紧接堆栈区前，所以该区地址根据堆栈的增加和减少而有所不同。因此堆栈区使用 6 个附加字节。

关于内存空间的保护方法详情，参见 带编程功能的 QB-MINI2 片上调试仿真器用户手册 (U18371E)。

26.1 BCD 校正电路功能

BCD（二进制编码的十进制数）码和 BCD 码相加/相减的结果通过该电路可获得 BCD 码。
对作为操作数的 A 寄存器和 BCDADJ 寄存器执行相加/相减，可获得十进制校正操作结果。

26.2 用于 BCD 校正电路的寄存器

BCD 校正电路使用以下寄存器。

- BCD 校正结果寄存器（BCDADJ）

（1）BCD 校正结果寄存器（BCDADJ）

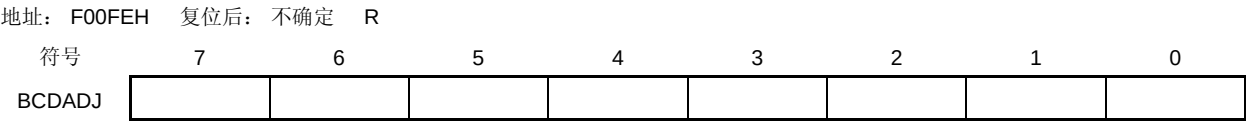
BCDADJ 寄存器存储校正值，该校正值是用作为操作数的 A 寄存器通过加/减指令获得以 BCD 编码的相加/相减结果。

根据读取时 A 寄存器的值以及 CY 和 AC 标志的值，从 BCDADJ 寄存器读取的数值有所不同。

用 8 位存储操作指令读取 BCDADJ。

复位输入后该寄存器不确定。

图 26-1. BCD 校正结果寄存器（BCDADJ）的格式



26.3 BCD 校正电路操作

BCD 校正电路的基本操作如下。

(1) 相加：用 BCD 编码值计算一个 BCD 码值和另一个 BCD 码值相加的结果

- <1> 将进行相加的 BCD 码存入 A 寄存器。
- <2> 将 A 寄存器的值和第二个操作数（用于相加的另一个 BCD 码）按二进制相加，二进制操作结果存入 A 寄存器，而校正值存入 BCDADJ 寄存器。
- <3> 通过按二进制相加 A 寄存器的值（二进制的相加结果）和 BCDADJ 寄存器（校正值），执行十进制校正，校正结果存入 A 寄存器和 CY 寄存器。

注意事项 根据读取时 A 寄存器的值以及 CY 和 AC 标志的值，从 BCDADJ 寄存器读取的数值有所不同。因此指令<2>或执行指令<3>，而不是执行其它指令。若要在中断使能状态中执行 BCD 校正，则需在中断功能中保存和恢复 A 寄存器。由 RETI 指令恢复 PSW（CY 标志和 AC 标志）。

举例如下。

例 1: $99 + 89 = 188$

指令	A 寄存器	CY 寄存器	AC 标志	BCDADJ 寄存器
MOV A, #99H ; <1>	99H	—	—	—
ADD A, #89H ; <2>	22H	1	1	66H
ADD A, !BCDADJ ; <3>	88H	1	0	—

例 2: $85 + 15 = 100$

指令	A 寄存器	CY 寄存器	AC 标志	BCDADJ 寄存器
MOV A, #85H ; <1>	85H	—	—	—
ADD A, #15H ; <2>	9AH	0	0	06H
ADD A, !BCDADJ ; <3>	00H	1	1	—

例 3: $80 + 80 = 160$

指令	A 寄存器	CY 寄存器	AC 标志	BCDADJ 寄存器
MOV A, #80H ; <1>	80H	—	—	—
ADD A, #80H ; <2>	00H	1	0	60H
ADD A, !BCDADJ ; <3>	60H	1	0	—

(2) 相减：用 BCD 编码值计算从另一个 BCD 码值减去一个 BCD 码值的结果

- <1> 将进行被减的 BCD 码存入 A 寄存器。
- <2> 将 A 寄存器的值和第二个操作数（用于相减的另一个 BCD 码）按二进制相减，二进制操作结果存入 A 寄存器，而校正值存入 BCDADJ 寄存器。
- <3> 通过按二进制相减 A 寄存器的值（二进制的相减结果）和 BCDADJ 寄存器（校正值），执行十进制校正，校正结果存入 A 寄存器和 CY 寄存器。

注意事项 根据读取时 A 寄存器的值以及 CY 和 AC 标志的值，从 BCDADJ 寄存器读取的数值有所不同。因此指令<2>或执行指令<3>，而不是执行其它指令。若要在中断使能状态中执行 BCD 校正，则需在中断功能中保存和恢复 A 寄存器。由 RETI 指令恢复 PSW（CY 标志和 AC 标志）。

举例如下。

例： $91 - 52 = 39$

指令	A 寄存器	CY 寄存器	AC 标志	BCDADJ 寄存器
MOV A, #91H ; <1>	91H	—	—	—
SUB A, #52H ; <2>	3FH	0	1	06H
SUB A, !BCDADJ ; <3>	39H	0	0	—

第二十七章 指令集

本章为 78K0R 微控制器指令集列表。关于各操作及操作模式参见单独的文档 **78K0R 微控制器指令用户手册 (U17792E)**。

备注 **表 27-5 操作列表**中阴影部分表示 78K0R 微控制器中新添加的操作或指令格式。

27.1 操作列表中的使用惯例

27.1.1 操作数标识符和指定方法

根据指令操作数标识符（详情参见汇编器说明）的描述方法，操作数在各指令的“操作数”列描述。有两种或更多中描述方法时，选择其中一种。大写的字母数字和符号，**#**，**!**，**!!**，**\$**，**\$!**，**[]**以及 **ES:** 是关键字，按其原形描述。各符号的意义如下。

- **#**: 立即数指定
- **!**: 16 位绝对地址指定
- **!!**: 20 位绝对地址指定
- **\$**: 8 位相对地址指定
- **\$!**: 16 位相对地址指定
- **[]**: 间接地址指定
- **ES:** : 扩展地址指定

以立即数为例，描述一个适当的数字值或标号。当使用标号时，确保描述 **#**，**!**，**!!**，**\$**，**\$!**，**[]**， and **ES:** 符号。

关于操作数寄存器标识符 **r** 和 **rp**，功能名称（**X**，**A**，**C** 等）或绝对名称（下表中括号中的名称，**R0**，**R1**，**R2** 等）都可用于描述。

表27-1. 操作数标识符和指定方法

标识符	描述方法
r rp sfr sfrp	X (R0)， A (R1)， C (R2)， B (R3)， E (R4)， D (R5)， L (R6)， H (R7) AX (RP0)， BC (RP1)， DE (RP2)， HL (RP3) 特殊功能寄存器符号 (SFR 符号) 特殊功能寄存器符号 (16 位可操作 SFR 符号。仅限于偶地址 [※])
saddr saddrp	FFE20H 至 FFF1FH 立即数或标号 FFE20H 至 FF1FH 立即数或标号 (仅限于偶地址 [※])
addr20 addr16 addr5	00000H 至 FFFFFH 立即数或标号 0000H 至 FFFFH 立即数或标号 (仅限于使用 16 位数据传输指令的偶地址 [※]) 0080H 至 00BFH 立即数或标号 (仅限于偶地址 [※])
word byte bit	16 位立即数或标号 8 位立即数或标号 3 位立即数或标号
RBn	RB0 至 RB3

注 当指定奇地址时位 0 = 0。

备注 关于特殊功能寄存器符号，参见**表 3-5 SFR 列表**和**表 3-6 扩展的 SFR (2nd SFR) 列表**。

27.1.2 操作栏的描述

当执行指令时，操作在“操作”栏中用以下符号显示。

表27-2. “操作”栏的符号

符号	功能
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
ES	ES 寄存器
CS	CS 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	进位标志
AC	辅助进位标志
Z	零标志
RBS	寄存器组选择标志
IE	中断请求使能标志
()	括号中由地址或寄存器内容指定的存储器内容
X _H , X _L	16 位累加器 寄存器: X _H = 高 8 位, X _L = 低 8 位
X _S , X _H , X _L	20 位累加器 寄存器: X _S = (位 19 至 16), X _H = (位 15 至 8), X _L = (位 7 至 0)
^	逻辑乘 (AND)
∨	逻辑和 (OR)
≠	非逻辑和 (非 OR)
—	反向数据
addr5	16 位立即数 (仅限于 0080H 至 00BF 的偶地址)
addr16	16 位立即数
addr20	20 位立即数
jdsp8	带符号的 8 位数据 (位移值)
jdsp16	带符号的 16 位数据 (位移值)

27.1.3 标志操作栏的描述

当执行指令时，标志值的变化在“标志” 栏中用以下符号显示。

表27-3. “标志” 栏的符号

符号	标志值的变化
(空 白)	不变
0	清除为 0
1	设置为 1
×	根据结果设置/清除
R	恢复前一个保存的值

27.1.4 PREFIX 指令

带 “ES: ”的指令有 PREFIX 操作码作为前缀，通过将 ES 寄存器的值从 F0000H 至 FFFFFH 增加到 64 KB 空间，使得访问数据区扩展到 1 MB 空间（00000H 至 FFFFFH）。当 PREFIX 操作码附加到目标指令作为前缀时，紧接 PREFIX 操作码后，只能执行一个作为 ES 寄存器值添加地址的指令。

表27-4. Use PREFIX 操作码的使用举例

指令	操作码				
	1	2	3	4	5
MOV !addr16, #byte	CFH	!addr16		#byte	–
MOV ES: !addr16, #byte	11H	CFH	!addr16		#byte
MOV A, [HL]	8BH	–	–	–	–
MOV A, ES: [HL]	11H	8BH	–	–	–

注意事项 在执行 PREFIX 指令前，用 MOV ES, A 等设置 ES 寄存器。

27.2 操作列表

表27-5. 操作列表 (1/17)

指令组	助记符	操作数	字节	时钟		操作	标志		
				注 1	注 2		Z	AC	CY
8 位数据传输	MOV	r, #byte	2	1	–	$r \leftarrow \text{字节}$			
		saddr, #byte	3	1	–	$(saddr) \leftarrow \text{字节}$			
		sfr, #byte	3	1	–	$sfr \leftarrow \text{字节}$			
		!addr16, #byte	4	1	–	$(addr16) \leftarrow \text{字节}$			
		A, r ^{注 3}	1	1	–	$A \leftarrow r$			
		r, A ^{注 3}	1	1	–	$r \leftarrow A$			
		A, saddr	2	1	–	$A \leftarrow (saddr)$			
		saddr, A	2	1	–	$(saddr) \leftarrow A$			
		A, sfr	2	1	–	$A \leftarrow sfr$			
		sfr, A	2	1	–	$sfr \leftarrow A$			
		A, !addr16	3	1	4	$A \leftarrow (addr16)$			
		!addr16, A	3	1	–	$(addr16) \leftarrow A$			
		PSW, #byte	3	3	–	$PSW \leftarrow \text{字节}$	x	x	x
		A, PSW	2	1	–	$A \leftarrow PSW$			
		PSW, A	2	3	–	$PSW \leftarrow A$	x	x	x
		ES, #byte	2	1	–	$ES \leftarrow \text{字节}$			
		ES, saddr	3	1	–	$ES \leftarrow (saddr)$			
		A, ES	2	1	–	$A \leftarrow ES$			
		ES, A	2	1	–	$ES \leftarrow A$			
		CS, #byte	3	1	–	$CS \leftarrow \text{字节}$			
		A, CS	2	1	–	$A \leftarrow CS$			
		CS, A	2	1	–	$CS \leftarrow A$			
		A, [DE]	1	1	4	$A \leftarrow (DE)$			
		[DE], A	1	1	–	$(DE) \leftarrow A$			
		[DE + byte], #byte	3	1	–	$(DE + \text{字节}) \leftarrow \text{字节}$			
		A, [DE + byte]	2	1	4	$A \leftarrow (DE + \text{字节})$			
		[DE + byte], A	2	1	–	$(DE + \text{字节}) \leftarrow A$			
		A, [HL]	1	1	4	$A \leftarrow (HL)$			
		[HL], A	1	1	–	$(HL) \leftarrow A$			
		[HL + byte], #byte	3	1	–	$(HL + \text{字节}) \leftarrow \text{字节}$			

- 注
1. 当访问内部 RAM 区或 SFR 区时，或者对于不进行数据访问的指令。
 2. 访问程序内存区时。
 3. 除 $r = A$

- 备注
1. 一个指令时钟周期是一个 CPU 时钟 (f_{CPU}) 周期，而 CPU 时钟由系统时钟控制寄存器 (CKC) 选择。
 2. 时钟编号用于当程序在内部 ROM (flash 存储器) 区时。

表 27-5. 操作列表 (2/17)

指令组	助记符	操作数	字节	时钟		操作	标志		
				注 1	注 2		Z	AC	CY
8 位数据传输	MOV	A, [HL + byte]	2	1	4	$A \leftarrow (HL + \text{字节})$			
		[HL + byte], A	2	1	–	$(HL + \text{字节}) \leftarrow A$			
		A, [HL + B]	2	1	4	$A \leftarrow (HL + B)$			
		[HL + B], A	2	1	–	$(HL + B) \leftarrow A$			
		A, [HL + C]	2	1	4	$A \leftarrow (HL + C)$			
		[HL + C], A	2	1	–	$(HL + C) \leftarrow A$			
		word[B], #byte	4	1	–	$(B + \text{字}) \leftarrow \text{字节}$			
		A, word[B]	3	1	4	$A \leftarrow (B + \text{字})$			
		word[B], A	3	1	–	$(B + \text{字}) \leftarrow A$			
		word[C], #byte	4	1	–	$(C + \text{字}) \leftarrow \text{字节}$			
		A, word[C]	3	1	4	$A \leftarrow (C + \text{字})$			
		word[C], A	3	1	–	$(C + \text{字}) \leftarrow A$			
		word[BC], #byte	4	1	–	$(BC + \text{字}) \leftarrow \text{字节}$			
		A, word[BC]	3	1	4	$A \leftarrow (BC + \text{字})$			
		word[BC], A	3	1	–	$(BC + \text{字}) \leftarrow A$			
		[SP + byte], #byte	3	1	–	$(SP + \text{字节}) \leftarrow \text{字节}$			
		A, [SP + byte]	2	1	–	$A \leftarrow (SP + \text{字节})$			
		[SP + byte], A	2	1	–	$(SP + \text{字节}) \leftarrow A$			
		B, saddr	2	1	–	$B \leftarrow (\text{saddr})$			
		B, !addr16	3	1	4	$B \leftarrow (\text{addr16})$			
		C, saddr	2	1	–	$C \leftarrow (\text{saddr})$			
		C, !addr16	3	1	4	$C \leftarrow (\text{addr16})$			
		X, saddr	2	1	–	$X \leftarrow (\text{saddr})$			
		X, !addr16	3	1	4	$X \leftarrow (\text{addr16})$			
		ES : !addr16 , #byte	5	2	–	$(ES, \text{addr16}) \leftarrow \text{字节}$			
		A, ES: !addr16	4	2	5	$A \leftarrow (ES, \text{addr16})$			
		ES: !addr16, A	4	2	–	$(ES, \text{addr16}) \leftarrow A$			
		A, ES: [DE]	2	2	5	$A \leftarrow (ES, DE)$			
		ES: [DE], A	2	2	–	$(ES, DE) \leftarrow A$			
		ES: [DE + byte], #byte	4	2	–	$((ES, DE) + \text{字节}) \leftarrow \text{字节}$			
		A, ES : [DE + byte]	3	2	5	$A \leftarrow ((ES, DE) + \text{字节})$			
		ES: [DE + byte], A	3	2	–	$((ES, DE) + \text{字节}) \leftarrow A$			

- 注 1. 当访问内部 RAM 区或 SFR 区时，或者对于不进行数据访问的指令。
 2. 访问程序内存区时。

- 备注 1. 一个指令时钟周期是一个 CPU 时钟 (f_{CPU}) 周期，而 CPU 时钟由系统时钟控制寄存器 (CKC) 选择。
 2. 时钟编号用于当程序在内部 ROM (flash 存储器) 区时。

表 27-5. 操作列表 (3/17)

指令组	助记符	操作数	字节	时钟		操作	标志		
				注 1	注 2		Z	AC	CY
8 位数据传输	MOV	A, ES: [HL]	2	2	5	$A \leftarrow (ES, HL)$			
		ES: [HL], A	2	2	—	$(ES, HL) \leftarrow A$			
		ES: [HL + byte], #byte	4	2	—	$((ES, HL) + \text{字节}) \leftarrow \text{字节}$			
		A, ES: [HL + byte]	3	2	5	$A \leftarrow ((ES, HL) + \text{字节})$			
		ES: [HL + byte], A	3	2	—	$((ES, HL) + \text{字节}) \leftarrow A$			
		A, ES: [HL + B]	3	2	5	$A \leftarrow ((ES, HL) + B)$			
		ES: [HL + B], A	3	2	—	$((ES, HL) + B) \leftarrow A$			
		A, ES: [HL + C]	3	2	5	$A \leftarrow ((ES, HL) + C)$			
		ES: [HL + C], A	3	2	—	$((ES, HL) + C) \leftarrow A$			
		ES: word[B], #byte	5	2	—	$((ES, B) + \text{字}) \leftarrow \text{字节}$			
		A, ES: word[B]	4	2	5	$A \leftarrow ((ES, B) + \text{字})$			
		ES: word[B], A	4	2	—	$((ES, B) + \text{字}) \leftarrow A$			
		ES: word[C], #byte	5	2	—	$((ES, C) + \text{字}) \leftarrow \text{字节}$			
		A, ES: word[C]	4	2	5	$A \leftarrow ((ES, C) + \text{字})$			
		ES: word[C], A	4	2	—	$((ES, C) + \text{字}) \leftarrow A$			
		ES: word[BC], #byte	5	2	—	$((ES, BC) + \text{字}) \leftarrow \text{字节}$			
		A, ES: word[BC]	4	2	5	$A \leftarrow ((ES, BC) + \text{字})$			
		ES: word[BC], A	4	2	—	$((ES, BC) + \text{字}) \leftarrow A$			
		B, ES: !addr16	4	2	5	$B \leftarrow (ES, \text{addr16})$			
		C, ES: !addr16	4	2	5	$C \leftarrow (ES, \text{addr16})$			
		X, ES: !addr16	4	2	5	$X \leftarrow (ES, \text{addr16})$			
	XCH	A, r ^{≠3}	1 (r=X) 2 (除 r=X)	1	—	$A \leftrightarrow r$			
		A, saddr	3	2	—	$A \leftrightarrow (\text{saddr})$			
		A, sfr	3	2	—	$A \leftrightarrow \text{sfr}$			
		A, !addr16	4	2	—	$A \leftrightarrow (\text{addr16})$			
		A, [DE]	2	2	—	$A \leftrightarrow (DE)$			
		A, [DE + byte]	3	2	—	$A \leftrightarrow (DE + \text{字节})$			
		A, [HL]	2	2	—	$A \leftrightarrow (HL)$			
		A, [HL + byte]	3	2	—	$A \leftrightarrow (HL + \text{字节})$			
		A, [HL + B]	2	2	—	$A \leftrightarrow (HL + B)$			
		A, [HL + C]	2	2	—	$A \leftrightarrow (HL + C)$			

- 注
1. 当访问内部 RAM 区或 SFR 区时，或者对于不进行数据访问的指令。
 2. 访问程序内存区时。
 3. 除 $r = A$

- 备注
1. 一个指令时钟周期是一个 CPU 时钟 (f_{CPU}) 周期，而 CPU 时钟由系统时钟控制寄存器 (CKC) 选择。
 2. 时钟编号用于当程序在内部 ROM (flash 存储器) 区时。

表 27-5. 操作列表 (4/17)

指令组	助记符	操作数	字节	时钟		操作	标志		
				注 1	注 2		Z	AC	CY
8 位数据传输	XCH	A, ES: !addr16	5	3	–	$A \leftrightarrow (ES, \text{addr16})$			
		A, ES: [DE]	3	3	–	$A \leftrightarrow (ES, DE)$			
		A, ES: [DE + byte]	4	3	–	$A \leftrightarrow ((ES, DE) + \text{字节})$			
		A, ES: [HL]	3	3	–	$A \leftrightarrow (ES, HL)$			
		A, ES: [HL + byte]	4	3	–	$A \leftrightarrow ((ES, HL) + \text{字节})$			
		A, ES: [HL + B]	3	3	–	$A \leftrightarrow ((ES, HL) + B)$			
		A, ES: [HL + C]	3	3	–	$A \leftrightarrow ((ES, HL) + C)$			
	ONEB	A	1	1	–	$A \leftarrow 01H$			
		X	1	1	–	$X \leftarrow 01H$			
		B	1	1	–	$B \leftarrow 01H$			
		C	1	1	–	$C \leftarrow 01H$			
		saddr	2	1	–	$(saddr) \leftarrow 01H$			
		!addr16	3	1	–	$(\text{addr16}) \leftarrow 01H$			
		ES: !addr16	4	2	–	$(ES, \text{addr16}) \leftarrow 01H$			
	CLRB	A	1	1	–	$A \leftarrow 00H$			
		X	1	1	–	$X \leftarrow 00H$			
		B	1	1	–	$B \leftarrow 00H$			
		C	1	1	–	$C \leftarrow 00H$			
		saddr	2	1	–	$(saddr) \leftarrow 00H$			
		!addr16	3	1	–	$(\text{addr16}) \leftarrow 00H$			
		ES: !addr16	4	2	–	$(ES, \text{addr16}) \leftarrow 00H$			
	MOVS	[HL + byte], X	3	1	–	$(HL + \text{字节}) \leftarrow X$	×		×
		ES: [HL + byte], X	4	2	–	$(ES, HL + \text{字节}) \leftarrow X$	×		×
16 位数据传输	MOVW	rp, #word	3	1	–	$rp \leftarrow \text{字}$			
		saddrp, #word	4	1	–	$(saddrp) \leftarrow \text{字}$			
		sfrp, #word	4	1	–	$sfrp \leftarrow \text{字}$			
		AX, saddrp	2	1	–	$AX \leftarrow (saddrp)$			
		saddrp, AX	2	1	–	$(saddrp) \leftarrow AX$			
		AX, sfrp	2	1	–	$AX \leftarrow sfrp$			
		sfrp, AX	2	1	–	$sfrp \leftarrow AX$			
		AX, rp ^{注 3}	1	1	–	$AX \leftarrow rp$			
		rp, AX ^{注 3}	1	1	–	$rp \leftarrow AX$			

- 注
1. 当访问内部 RAM 区或 SFR 区时，或者对于不进行数据访问的指令。
 2. 访问程序内存区时。
 3. 除 $rp = AX$

- 备注
1. 一个指令时钟周期是一个 CPU 时钟 (f_{CPU}) 周期，而 CPU 时钟由系统时钟控制寄存器 (CKC) 选择。
 2. 时钟编号用于当程序在内部 ROM (flash 存储器) 区时。

表 27-5. 操作列表 (5/17)

指令组	助记符	操作数	字节	时钟		操作	标志		
				注 1	注 2		Z	AC	CY
16 位 数 数据传输	MOVW	AX, !addr16	3	1	4	AX ← (addr16)			
		!addr16, AX	3	1	–	(addr16) ← AX			
		AX, [DE]	1	1	4	AX ← (DE)			
		[DE], AX	1	1	–	(DE) ← AX			
		AX, [DE + byte]	2	1	4	AX ← (DE + 字节)			
		[DE + byte], AX	2	1	–	(DE + 字节) ← AX			
		AX, [HL]	1	1	4	AX ← (HL)			
		[HL], AX	1	1	–	(HL) ← AX			
		AX, [HL + byte]	2	1	4	AX ← (HL + 字节)			
		[HL + byte], AX	2	1	–	(HL + 字节) ← AX			
		AX, word[B]	3	1	4	AX ← (B + 字)			
		word[B], AX	3	1	–	(B + 字) ← AX			
		AX, word[C]	3	1	4	AX ← (C + 字)			
		word[C], AX	3	1	–	(C + 字) ← AX			
		AX, word[BC]	3	1	4	AX ← (BC + 字)			
		word[BC], AX	3	1	–	(BC + 字) ← AX			
		AX, [SP + byte]	2	1	–	AX ← (SP + 字节)			
		[SP + byte], AX	2	1	–	(SP + 字节) ← AX			
		BC, saddrp	2	1	–	BC ← (saddrp)			
		BC, !addr16	3	1	4	BC ← (addr16)			
		DE, saddrp	2	1	–	DE ← (saddrp)			
		DE, !addr16	3	1	4	DE ← (addr16)			
		HL, saddrp	2	1	–	HL ← (saddrp)			
		HL, !addr16	3	1	4	HL ← (addr16)			
		AX, ES: !addr16	4	2	5	AX ← (ES, addr16)			
		ES: !addr16, AX	4	2	–	(ES, addr16) ← AX			
		AX, ES: [DE]	2	2	5	AX ← (ES, DE)			
		ES: [DE], AX	2	2	–	(ES, DE) ← AX			
		AX, ES:[DE + byte]	3	2	5	AX ← ((ES, DE) + 字节)			
		ES:[DE + byte], AX	3	2	–	((ES, DE) + 字节) ← AX			
		AX, ES: [HL]	2	2	5	AX ← (ES, HL)			
		ES: [HL], AX	2	2	–	(ES, HL) ← AX			

- 注
1. 当访问内部 RAM 区或 SFR 区时，或者对于不进行数据访问的指令。
 2. 访问程序内存区时。

- 备注
1. 一个指令时钟周期是一个 CPU 时钟 (f_{CPU}) 周期，而 CPU 时钟由系统时钟控制寄存器 (CKC) 选择。
 2. 时钟编号用于当程序在内部 ROM (flash 存储器) 区时。

表 27-5. 操作列表 (6/17)

指令组	助记符	操作数	字节	时钟		操作	标志		
				注 1	注 2		Z	AC	CY
16 位 数 数据传输	MOVW	AX, ES:[HL + byte]	3	2	5	$AX \leftarrow ((ES, HL) + \text{字节})$			
		ES:[HL + byte], AX	3	2	—	$((ES, HL) + \text{字节}) \leftarrow AX$			
		AX, ES: word[B]	4	2	5	$AX \leftarrow ((ES, B) + \text{字})$			
		ES: word[B], AX	4	2	—	$((ES, B) + \text{字}) \leftarrow AX$			
		AX, ES: word[C]	4	2	5	$AX \leftarrow ((ES, C) + \text{字})$			
		ES: word[C], AX	4	2	—	$((ES, C) + \text{字}) \leftarrow AX$			
		AX, ES: word[BC]	4	2	5	$AX \leftarrow ((ES, BC) + \text{字})$			
		ES: word[BC], AX	4	2	—	$((ES, BC) + \text{字}) \leftarrow AX$			
		BC, ES: !addr16	4	2	5	$BC \leftarrow (ES, \text{addr16})$			
		DE, ES: !addr16	4	2	5	$DE \leftarrow (ES, \text{addr16})$			
		HL, ES: !addr16	4	2	5	$HL \leftarrow (ES, \text{addr16})$			
	XCHW	AX, rp ^{注 3}	1	1	—	$AX \leftrightarrow rp$			
	ONEW	AX	1	1	—	$AX \leftarrow 0001H$			
		BC	1	1	—	$BC \leftarrow 0001H$			
	CLRW	AX	1	1	—	$AX \leftarrow 0000H$			
		BC	1	1	—	$BC \leftarrow 0000H$			
8 位操作	ADD	A, #byte	2	1	—	$A, CY \leftarrow A + \text{字节}$	x	x	x
		saddr, #byte	3	2	—	$(saddr), CY \leftarrow (saddr) + \text{字节}$	x	x	x
		A, r ^{注 4}	2	1	—	$A, CY \leftarrow A + r$	x	x	x
		r, A	2	1	—	$r, CY \leftarrow r + A$	x	x	x
		A, saddr	2	1	—	$A, CY \leftarrow A + (saddr)$	x	x	x
		A, !addr16	3	1	4	$A, CY \leftarrow A + (\text{addr16})$	x	x	x
		A, [HL]	1	1	4	$A, CY \leftarrow A + (HL)$	x	x	x
		A, [HL + byte]	2	1	4	$A, CY \leftarrow A + (HL + \text{字节})$	x	x	x
		A, [HL + B]	2	1	4	$A, CY \leftarrow A + (HL + B)$	x	x	x
		A, [HL + C]	2	1	4	$A, CY \leftarrow A + (HL + C)$	x	x	x
		A, ES: !addr16	4	2	5	$A, CY \leftarrow A + (ES, \text{addr16})$	x	x	x
		A, ES: [HL]	2	2	5	$A, CY \leftarrow A + (ES, HL)$	x	x	x
		A, ES: [HL + byte]	3	2	5	$A, CY \leftarrow A + ((ES, HL) + \text{字节})$	x	x	x
		A, ES: [HL + B]	3	2	5	$A, CY \leftarrow A + ((ES, HL) + B)$	x	x	x
		A, ES: [HL + C]	3	2	5	$A, CY \leftarrow A + ((ES, HL) + C)$	x	x	x

- 注
1. 当访问内部 RAM 区或 SFR 区时，或者对于不进行数据访问的指令。
 2. 访问程序内存区时。
 3. 除 $rp = AX$
 4. 除 $tr = A$

- 备注
1. 一个指令时钟周期是一个 CPU 时钟 (f_{CPU}) 周期，而 CPU 时钟由系统时钟控制寄存器 (CKC) 选择。
 2. 时钟编号用于当程序在内部 ROM (flash 存储器) 区时。

表 27-5. 操作列表 (7/17)

指令组	助记符	操作数	字节	时钟		操作	标志		
				注 1	注 2		Z	AC	CY
8 位操作	ADDC	A, #byte	2	1	–	$A, CY \leftarrow A + \text{字节} + CY$	x	x	x
		saddr, #byte	3	2	–	$(saddr), CY \leftarrow (saddr) + \text{字节} + CY$	x	x	x
		A, $r^{\#3}$	2	1	–	$A, CY \leftarrow A + r + CY$	x	x	x
		r, A	2	1	–	$r, CY \leftarrow r + A + CY$	x	x	x
		A, saddr	2	1	–	$A, CY \leftarrow A + (saddr) + CY$	x	x	x
		A, !addr16	3	1	4	$A, CY \leftarrow A + (addr16) + CY$	x	x	x
		A, [HL]	1	1	4	$A, CY \leftarrow A + (HL) + CY$	x	x	x
		A, [HL + byte]	2	1	4	$A, CY \leftarrow A + (HL + \text{字节}) + CY$	x	x	x
		A, [HL + B]	2	1	4	$A, CY \leftarrow A + (HL + B) + CY$	x	x	x
		A, [HL + C]	2	1	4	$A, CY \leftarrow A + (HL + C) + CY$	x	x	x
		A, ES: !addr16	4	2	5	$A, CY \leftarrow A + (ES, addr16) + CY$	x	x	x
		A, ES: [HL]	2	2	5	$A, CY \leftarrow A + (ES, HL) + CY$	x	x	x
		A, ES: [HL + byte]	3	2	5	$A, CY \leftarrow A + ((ES, HL) + \text{字节}) + CY$	x	x	x
		A, ES: [HL + B]	3	2	5	$A, CY \leftarrow A + ((ES, HL) + B) + CY$	x	x	x
		A, ES: [HL + C]	3	2	5	$A, CY \leftarrow A + ((ES, HL) + C) + CY$	x	x	x
	SUB	A, #byte	2	1	–	$A, CY \leftarrow A - \text{字节}$	x	x	x
		saddr, #byte	3	2	–	$(saddr), CY \leftarrow (saddr) - \text{字节}$	x	x	x
		A, $r^{\#3}$	2	1	–	$A, CY \leftarrow A - r$	x	x	x
		r, A	2	1	–	$r, CY \leftarrow r - A$	x	x	x
		A, saddr	2	1	–	$A, CY \leftarrow A - (saddr)$	x	x	x
		A, !addr16	3	1	4	$A, CY \leftarrow A - (addr16)$	x	x	x
		A, [HL]	1	1	4	$A, CY \leftarrow A - (HL)$	x	x	x
		A, [HL + byte]	2	1	4	$A, CY \leftarrow A - (HL + \text{字节})$	x	x	x
		A, [HL + B]	2	1	4	$A, CY \leftarrow A - (HL + B)$	x	x	x
		A, [HL + C]	2	1	4	$A, CY \leftarrow A - (HL + C)$	x	x	x
		A, ES: !addr16	4	2	5	$A, CY \leftarrow A - (ES: addr16)$	x	x	x
		A, ES: [HL]	2	2	5	$A, CY \leftarrow A - (ES: HL)$	x	x	x
		A, ES: [HL + byte]	3	2	5	$A, CY \leftarrow A - ((ES: HL) + \text{字节})$	x	x	x
		A, ES: [HL + B]	3	2	5	$A, CY \leftarrow A - ((ES: HL) + B)$	x	x	x
		A, ES: [HL + C]	3	2	5	$A, CY \leftarrow A - ((ES: HL) + C)$	x	x	x

- 注
1. 当访问内部 RAM 区或 SFR 区时，或者对于不进行数据访问的指令。
 2. 访问程序内存区时。
 3. 除 $r = A$

- 备注
1. 一个指令时钟周期是一个 CPU 时钟 (f_{CPU}) 周期，而 CPU 时钟由系统时钟控制寄存器 (CKC) 选择。
 2. 时钟编号用于当程序在内部 ROM (flash 存储器) 区时。

表 27-5. 操作列表 (8/17)

指令组	助记符	操作数	字节	时钟		操作	标志		
				注 1	注 2		Z	AC	CY
8 位操作	SUBC	A, #byte	2	1	–	$A, CY \leftarrow A - \text{字节} - CY$	x	x	x
		saddr, #byte	3	2	–	$(saddr), CY \leftarrow (saddr) - \text{字节} - CY$	x	x	x
		A, $r^{\#3}$	2	1	–	$A, CY \leftarrow A - r - CY$	x	x	x
		r, A	2	1	–	$r, CY \leftarrow r - A - CY$	x	x	x
		A, saddr	2	1	–	$A, CY \leftarrow A - (saddr) - CY$	x	x	x
		A, !addr16	3	1	4	$A, CY \leftarrow A - (addr16) - CY$	x	x	x
		A, [HL]	1	1	4	$A, CY \leftarrow A - (HL) - CY$	x	x	x
		A, [HL + byte]	2	1	4	$A, CY \leftarrow A - (HL + \text{字节}) - CY$	x	x	x
		A, [HL + B]	2	1	4	$A, CY \leftarrow A - (HL + B) - CY$	x	x	x
		A, [HL + C]	2	1	4	$A, CY \leftarrow A - (HL + C) - CY$	x	x	x
		A, ES: !addr16	4	2	5	$A, CY \leftarrow A - (ES: addr16) - CY$	x	x	x
		A, ES: [HL]	2	2	5	$A, CY \leftarrow A - (ES: HL) - CY$	x	x	x
		A, ES: [HL + byte]	3	2	5	$A, CY \leftarrow A - ((ES: HL) + \text{字节}) - CY$	x	x	x
		A, ES: [HL + B]	3	2	5	$A, CY \leftarrow A - ((ES: HL) + B) - CY$	x	x	x
		A, ES: [HL + C]	3	2	5	$A, CY \leftarrow A - ((ES: HL) + C) - CY$	x	x	x
	AND	A, #byte	2	1	–	$A \leftarrow A \wedge \text{字节}$	x		
		saddr, #byte	3	2	–	$(saddr) \leftarrow (saddr) \wedge \text{字节}$	x		
		A, $r^{\#3}$	2	1	–	$A \leftarrow A \wedge r$	x		
		r, A	2	1	–	$r \leftarrow r \wedge A$	x		
		A, saddr	2	1	–	$A \leftarrow A \wedge (saddr)$	x		
		A, !addr16	3	1	4	$A \leftarrow A \wedge (addr16)$	x		
		A, [HL]	1	1	4	$A \leftarrow A \wedge (HL)$	x		
		A, [HL + byte]	2	1	4	$A \leftarrow A \wedge (HL + \text{字节})$	x		
		A, [HL + B]	2	1	4	$A \leftarrow A \wedge (HL + B)$	x		
		A, [HL + C]	2	1	4	$A \leftarrow A \wedge (HL + C)$	x		
		A, ES: !addr16	4	2	5	$A \leftarrow A \wedge (ES: addr16)$	x		
		A, ES: [HL]	2	2	5	$A \leftarrow A \wedge (ES: HL)$	x		
		A, ES: [HL + byte]	3	2	5	$A \leftarrow A \wedge ((ES: HL) + \text{字节})$	x		
		A, ES: [HL + B]	3	2	5	$A \leftarrow A \wedge ((ES: HL) + B)$	x		
		A, ES: [HL + C]	3	2	5	$A \leftarrow A \wedge ((ES: HL) + C)$	x		

- 注
1. 当访问内部 RAM 区或 SFR 区时，或者对于不进行数据访问的指令。
 2. 访问程序内存区时。
 3. 除 $r = A$

- 备注
1. 一个指令时钟周期是一个 CPU 时钟 (f_{CPU}) 周期，而 CPU 时钟由系统时钟控制寄存器 (CKC) 选择。
 2. 时钟编号用于当程序在内部 ROM (flash 存储器) 区时。

表 27-5. 操作列表 (9/17)

指令组	助记符	操作数	字节	时钟		操作	标志		
				注 1	注 2		Z	AC	CY
8 位操作	OR	A, #byte	2	1	–	$A \leftarrow A \vee \text{字节}$	x		
		saddr, #byte	3	2	–	$(\text{saddr}) \leftarrow (\text{saddr}) \vee \text{字节}$	x		
		A, r ^{#3}	2	1	–	$A \leftarrow A \vee r$	x		
		r, A	2	1	–	$r \leftarrow r \vee A$	x		
		A, saddr	2	1	–	$A \leftarrow A \vee (\text{saddr})$	x		
		A, !addr16	3	1	4	$A \leftarrow A \vee (\text{addr16})$	x		
		A, [HL]	1	1	4	$A \leftarrow A \vee (\text{HL})$	x		
		A, [HL + byte]	2	1	4	$A \leftarrow A \vee (\text{HL} + \text{字节})$	x		
		A, [HL + B]	2	1	4	$A \leftarrow A \vee (\text{HL} + B)$	x		
		A, [HL + C]	2	1	4	$A \leftarrow A \vee (\text{HL} + C)$	x		
		A, ES: !addr16	4	2	5	$A \leftarrow A \vee (\text{ES: addr16})$	x		
		A, ES: [HL]	2	2	5	$A \leftarrow A \vee (\text{ES: HL})$	x		
		A, ES: [HL + byte]	3	2	5	$A \leftarrow A \vee ((\text{ES: HL}) + \text{字节})$	x		
		A, ES: [HL + B]	3	2	5	$A \leftarrow A \vee ((\text{ES: HL}) + B)$	x		
		A, ES: [HL + C]	3	2	5	$A \leftarrow A \vee ((\text{ES: HL}) + C)$	x		
	XOR	A, #byte	2	1	–	$A \leftarrow A \nabla \text{字节}$	x		
		saddr, #byte	3	2	–	$(\text{saddr}) \leftarrow (\text{saddr}) \nabla \text{字节}$	x		
		A, r ^{#3}	2	1	–	$A \leftarrow A \nabla r$	x		
		r, A	2	1	–	$r \leftarrow r \nabla A$	x		
		A, saddr	2	1	–	$A \leftarrow A \nabla (\text{saddr})$	x		
		A, !addr16	3	1	4	$A \leftarrow A \nabla (\text{addr16})$	x		
		A, [HL]	1	1	4	$A \leftarrow A \nabla (\text{HL})$	x		
		A, [HL + byte]	2	1	4	$A \leftarrow A \nabla (\text{HL} + \text{字节})$	x		
		A, [HL + B]	2	1	4	$A \leftarrow A \nabla (\text{HL} + B)$	x		
		A, [HL + C]	2	1	4	$A \leftarrow A \nabla (\text{HL} + C)$	x		
		A, ES: !addr16	4	2	5	$A \leftarrow A \nabla (\text{ES: addr16})$	x		
		A, ES: [HL]	2	2	5	$A \leftarrow A \nabla (\text{ES: HL})$	x		
		A, ES: [HL + byte]	3	2	5	$A \leftarrow A \nabla ((\text{ES: HL}) + \text{字节})$	x		
		A, ES: [HL + B]	3	2	5	$A \leftarrow A \nabla ((\text{ES: HL}) + B)$	x		
		A, ES: [HL + C]	3	2	5	$A \leftarrow A \nabla ((\text{ES: HL}) + C)$	x		

- 注
1. 当访问内部 RAM 区或 SFR 区时，或者对于不进行数据访问的指令。
 2. 访问程序内存区时。
 3. 除 r = A

- 备注
1. 一个指令时钟周期是一个 CPU 时钟 (f_{CPU}) 周期，而 CPU 时钟由系统时钟控制寄存器 (CKC) 选择。
 2. 时钟编号用于当程序在内部 ROM (flash 存储器) 区时。

表 27-5. 操作列表 (10/17)

指令组	助记符	操作数	字节	时钟		操作	标志		
				注 1	注 2		Z	AC	CY
8 位操作	CMP	A, #byte	2	1	–	A – 字节	×	×	×
		saddr, #byte	3	1	–	(saddr) – 字节	×	×	×
		A, r ^{注 3}	2	1	–	A – r	×	×	×
		r, A	2	1	–	r – A	×	×	×
		A, saddr	2	1	–	A – (saddr)	×	×	×
		A, !addr16	3	1	4	A – (addr16)	×	×	×
		A, [HL]	1	1	4	A – (HL)	×	×	×
		A, [HL + byte]	2	1	4	A – (HL + 字节)	×	×	×
		A, [HL + B]	2	1	4	A – (HL + B)	×	×	×
		A, [HL + C]	2	1	4	A – (HL + C)	×	×	×
		!addr16, #byte	4	1	4	(addr16) – 字节	×	×	×
		A, ES: !addr16	4	2	5	A – (ES: addr16)	×	×	×
		A, ES: [HL]	2	2	5	A – (ES: HL)	×	×	×
		A, ES: [HL + byte]	3	2	5	A – ((ES: HL) + 字节)	×	×	×
		A, ES: [HL + B]	3	2	5	A – ((ES: HL) + B)	×	×	×
		A, ES: [HL + C]	3	2	5	A – ((ES: HL) + C)	×	×	×
		ES: !addr16, #byte	5	2	5	(ES: addr16) – 字节	×	×	×
	CMP0	A	1	1	–	A – 00H	×	×	×
		X	1	1	–	X – 00H	×	×	×
		B	1	1	–	B – 00H	×	×	×
		C	1	1	–	C – 00H	×	×	×
		saddr	2	1	–	(saddr) – 00H	×	×	×
		!addr16	3	1	4	(addr16) – 00H	×	×	×
		ES: !addr16	4	2	5	(ES: addr16) – 00H	×	×	×
	CMPS	X, [HL + byte]	3	1	4	X – (HL + 字节)	×	×	×
		X, ES: [HL + byte]	4	2	5	X – ((ES: HL) + 字节)	×	×	×

注 1. 当访问内部 RAM 区或 SFR 区时，或者对于不进行数据访问的指令。

2. 访问程序内存区时。

3. 除 r = A

备注 1. 一个指令时钟周期是一个 CPU 时钟 (f_{CPU}) 周期，而 CPU 时钟由系统时钟控制寄存器 (CKC) 选择。

2. 时钟编号用于当程序在内部 ROM (flash 存储器) 区时。

表 27-5. 操作列表 (11/17)

指令组	助记符	操作数	字节	时钟		操作	标志		
				注 1	注 2		Z	AC	CY
16 位操作	ADDW	AX, #word	3	1	–	AX, CY $\leftarrow$ AX + 字	x	x	x
		AX, AX	1	1	–	AX, CY $\leftarrow$ AX + AX	x	x	x
		AX, BC	1	1	–	AX, CY $\leftarrow$ AX + BC	x	x	x
		AX, DE	1	1	–	AX, CY $\leftarrow$ AX + DE	x	x	x
		AX, HL	1	1	–	AX, CY $\leftarrow$ AX + HL	x	x	x
		AX, saddrp	2	1	–	AX, CY $\leftarrow$ AX + (saddrp)	x	x	x
		AX, !addr16	3	1	4	AX, CY $\leftarrow$ AX + (addr16)	x	x	x
		AX, [HL+byte]	3	1	4	AX, CY $\leftarrow$ AX + (HL + 字节)	x	x	x
		AX, ES: !addr16	4	2	5	AX, CY $\leftarrow$ AX + (ES: addr16)	x	x	x
		AX, ES: [HL+byte]	4	2	5	AX, CY $\leftarrow$ AX + ((ES: HL) + 字节)	x	x	x
	SUBW	AX, #word	3	1	–	AX, CY $\leftarrow$ AX – 字	x	x	x
		AX, BC	1	1	–	AX, CY $\leftarrow$ AX – BC	x	x	x
		AX, DE	1	1	–	AX, CY $\leftarrow$ AX – DE	x	x	x
		AX, HL	1	1	–	AX, CY $\leftarrow$ AX – HL	x	x	x
		AX, saddrp	2	1	–	AX, CY $\leftarrow$ AX – (saddrp)	x	x	x
		AX, !addr16	3	1	4	AX, CY $\leftarrow$ AX – (addr16)	x	x	x
		AX, [HL+byte]	3	1	4	AX, CY $\leftarrow$ AX – (HL + 字节)	x	x	x
		AX, ES: !addr16	4	2	5	AX, CY $\leftarrow$ AX – (ES: addr16)	x	x	x
		AX, ES: [HL+byte]	4	2	5	AX, CY $\leftarrow$ AX – ((ES: HL) + 字节)	x	x	x
	CMPW	AX, #word	3	1	–	AX – 字	x	x	x
		AX, BC	1	1	–	AX – BC	x	x	x
		AX, DE	1	1	–	AX – DE	x	x	x
		AX, HL	1	1	–	AX – HL	x	x	x
		AX, saddrp	2	1	–	AX – (saddrp)	x	x	x
		AX, !addr16	3	1	4	AX – (addr16)	x	x	x
		AX, [HL+byte]	3	1	4	AX – (HL + 字节)	x	x	x
		AX, ES: !addr16	4	2	5	AX – (ES: addr16)	x	x	x
		AX, ES: [HL+byte]	4	2	5	AX – ((ES: HL) + 字节)	x	x	x
乘	MULU	X	1	1	–	AX $\leftarrow$ A $\times$ X			

- 注
1. 当访问内部 RAM 区或 SFR 区时，或者对于不进行数据访问的指令。
 2. 访问程序内存区时。

- 备注
1. 一个指令时钟周期是一个 CPU 时钟 (f_{CPU}) 周期，而 CPU 时钟由系统时钟控制寄存器 (CKC) 选择。
 2. 时钟编号用于当程序在内部 ROM (flash 存储器) 区时。

表 27-5. 操作列表 (12/17)

指令组	助记符	操作数	字节	时钟		操作	标志		
				注 1	注 2		Z	AC	CY
递增 / 递减	INC	r	1	1	—	$r \leftarrow r + 1$	×	×	
		saddr	2	2	—	$(saddr) \leftarrow (saddr) + 1$	×	×	
		laddr16	3	2	—	$(addr16) \leftarrow (addr16) + 1$	×	×	
		[HL+byte]	3	2	—	$(HL+字节) \leftarrow (HL+字节) + 1$	×	×	
		ES: laddr16	4	3	—	$(ES, addr16) \leftarrow (ES, addr16) + 1$	×	×	
		ES: [HL+byte]	4	3	—	$((ES:HL) + 字节) \leftarrow ((ES:HL) + 字节) + 1$	×	×	
	DEC	r	1	1	—	$r \leftarrow r - 1$	×	×	
		saddr	2	2	—	$(saddr) \leftarrow (saddr) - 1$	×	×	
		laddr16	3	2	—	$(addr16) \leftarrow (addr16) - 1$	×	×	
		[HL+byte]	3	2	—	$(HL+字节) \leftarrow (HL+字节) - 1$	×	×	
		ES: laddr16	4	3	—	$(ES, addr16) \leftarrow (ES, addr16) - 1$	×	×	
		ES: [HL+byte]	4	3	—	$((ES:HL) + 字节) \leftarrow ((ES:HL) + 字节) - 1$	×	×	
	INCW	rp	1	1	—	$rp \leftarrow rp + 1$			
		saddrp	2	2	—	$(saddrp) \leftarrow (saddrp) + 1$			
		laddr16	3	2	—	$(addr16) \leftarrow (addr16) + 1$			
		[HL+byte]	3	2	—	$(HL+字节) \leftarrow (HL+字节) + 1$			
		ES: laddr16	4	3	—	$(ES, addr16) \leftarrow (ES, addr16) + 1$			
		ES: [HL+byte]	4	3	—	$((ES:HL) + 字节) \leftarrow ((ES:HL) + 字节) + 1$			
	DECW	rp	1	1	—	$rp \leftarrow rp - 1$			
		saddrp	2	2	—	$(saddrp) \leftarrow (saddrp) - 1$			
		laddr16	3	2	—	$(addr16) \leftarrow (addr16) - 1$			
		[HL+byte]	3	2	—	$(HL+字节) \leftarrow (HL+字节) - 1$			
		ES: laddr16	4	3	—	$(ES, addr16) \leftarrow (ES, addr16) - 1$			
		ES: [HL+byte]	4	3	—	$((ES:HL) + 字节) \leftarrow ((ES:HL) + 字节) - 1$			
移位	SHR	A, cnt	2	1	—	$(CY \leftarrow A_0, A_{m-1} \leftarrow A_m, A_7 \leftarrow 0) \times cnt$			×
	SHRW	AX, cnt	2	1	—	$(CY \leftarrow AX_0, AX_{m-1} \leftarrow AX_m, AX_{15} \leftarrow 0) \times cnt$			×
	SHL	A, cnt	2	1	—	$(CY \leftarrow A_7, A_m \leftarrow A_{m-1}, A_0 \leftarrow 0) \times cnt$			×
		B, cnt	2	1	—	$(CY \leftarrow B_7, B_m \leftarrow B_{m-1}, B_0 \leftarrow 0) \times cnt$			×
		C, cnt	2	1	—	$(CY \leftarrow C_7, C_m \leftarrow C_{m-1}, C_0 \leftarrow 0) \times cnt$			×
	SHLW	AX, cnt	2	1	—	$(CY \leftarrow AX_{15}, AX_m \leftarrow AX_{m-1}, AX_0 \leftarrow 0) \times cnt$			×
		BC, cnt	2	1	—	$(CY \leftarrow BC_{15}, BC_m \leftarrow BC_{m-1}, BC_0 \leftarrow 0) \times cnt$			×
	SAR	A, cnt	2	1	—	$(CY \leftarrow A_0, A_{m-1} \leftarrow A_m, A_7 \leftarrow A_7) \times cnt$			×
	SARW	AX, cnt	2	1	—	$(CY \leftarrow AX_0, AX_{m-1} \leftarrow AX_m, AX_{15} \leftarrow AX_{15}) \times cnt$			×

- 注 1. 当访问内部 RAM 区或 SFR 区时，或者对于不进行数据访问的指令。
 2. 访问程序内存区时。

- 备注 1. 一个指令时钟周期是一个 CPU 时钟 (f_{cpu}) 周期，而 CPU 时钟由系统时钟控制寄存器 (CKC) 选择。
 2. 时钟编号用于当程序在内部 ROM (flash 存储器) 区时。
 3. cnt 表示位移计数。

表 27-5. 操作列表 (13/17)

指令组	助记符	操作数	字节	时钟		操作	标志		
				注 1	注 2		Z	AC	CY
循环	ROR	A, 1	2	1	–	$(CY, A_7 \leftarrow A_0, A_{m-1} \leftarrow A_m) \times 1$			×
	ROL	A, 1	2	1	–	$(CY, A_0 \leftarrow A_7, A_{m+1} \leftarrow A_m) \times 1$			×
	RORC	A, 1	2	1	–	$(CY \leftarrow A_0, A_7 \leftarrow CY, A_{m-1} \leftarrow A_m) \times 1$			×
	ROLC	A, 1	2	1	–	$(CY \leftarrow A_7, A_0 \leftarrow CY, A_{m+1} \leftarrow A_m) \times 1$			×
	ROLWC	AX, 1	2	1	–	$(CY \leftarrow AX_{15}, AX_0 \leftarrow CY, AX_{m+1} \leftarrow AX_m) \times 1$			×
		BC, 1	2	1	–	$(CY \leftarrow BC_{15}, BC_0 \leftarrow CY, BC_{m+1} \leftarrow BC_m) \times 1$			×
位操作	MOV1	CY, saddr.bit	3	1	–	$CY \leftarrow (saddr).位$			×
		CY, sfr.bit	3	1	–	$CY \leftarrow sfr.位$			×
		CY, A.bit	2	1	–	$CY \leftarrow A.位$			×
		CY, PSW.bit	3	1	–	$CY \leftarrow PSW.位$			×
		CY, [HL].bit	2	1	4	$CY \leftarrow (HL).位$			×
		saddr.bit, CY	3	2	–	$(saddr).位 \leftarrow CY$			
		sfr.bit, CY	3	2	–	$sfr.位 \leftarrow CY$			
		A.bit, CY	2	1	–	$A.位 \leftarrow CY$			
		PSW.bit, CY	3	4	–	$PSW.位 \leftarrow CY$	×	×	
		[HL].bit, CY	2	2	–	$(HL).位 \leftarrow CY$			
		CY, ES: [HL].bit	3	2	5	$CY \leftarrow (ES, HL).位$			×
		ES: [HL].bit, CY	3	3	–	$(ES, HL).位 \leftarrow CY$			
	AND1	CY, saddr.bit	3	1	–	$CY \leftarrow CY \wedge (saddr).位$			×
		CY, sfr.bit	3	1	–	$CY \leftarrow CY \wedge sfr.位$			×
		CY, A.bit	2	1	–	$CY \leftarrow CY \wedge A.位$			×
		CY, PSW.bit	3	1	–	$CY \leftarrow CY \wedge PSW.位$			×
		CY, [HL].bit	2	1	4	$CY \leftarrow CY \wedge (HL).位$			×
		CY, ES: [HL].bit	3	2	5	$CY \leftarrow CY \wedge (ES, HL).位$			×
	OR1	CY, saddr.bit	3	1	–	$CY \leftarrow CY \vee (saddr).位$			×
		CY, sfr.bit	3	1	–	$CY \leftarrow CY \vee sfr.位$			×
		CY, A.bit	2	1	–	$CY \leftarrow CY \vee A.位$			×
		CY, PSW.bit	3	1	–	$CY \leftarrow CY \vee PSW.位$			×
		CY, [HL].bit	2	1	4	$CY \leftarrow CY \vee (HL).位$			×
		CY, ES: [HL].bit	3	2	5	$CY \leftarrow CY \vee (ES, HL).位$			×

- 注
1. 当访问内部 RAM 区或 SFR 区时，或者对于不进行数据访问的指令。
 2. 访问程序内存区时。

- 备注
1. 一个指令时钟周期是一个 CPU 时钟 (f_{CPU}) 周期，而 CPU 时钟由系统时钟控制寄存器 (CKC) 选择。
 2. 时钟编号用于当程序在内部 ROM (flash 存储器) 区时。

表 27-5. 操作列表 (14/17)

指令组	助记符	操作数	字节	时钟		操作	标志		
				注 1	注 2		Z	AC	CY
位操作	XOR1	CY, saddr.bit	3	1	–	$CY \leftarrow CY \oplus (\text{saddr}).\text{位}$			×
		CY, sfr.bit	3	1	–	$CY \leftarrow CY \oplus \text{sfr.位}$			×
		CY, A.bit	2	1	–	$CY \leftarrow CY \oplus A.\text{位}$			×
		CY, PSW.bit	3	1	–	$CY \leftarrow CY \oplus \text{PSW.位}$			×
		CY, [HL].bit	2	1	4	$CY \leftarrow CY \oplus (\text{HL}).\text{位}$			×
		CY, ES: [HL].bit	3	2	5	$CY \leftarrow CY \oplus (\text{ES}, \text{HL}).\text{位}$			×
	SET1	saddr.bit	3	2	–	$(\text{saddr}).\text{位} \leftarrow 1$			
		sfr.bit	3	2	–	$\text{sfr.位} \leftarrow 1$			
		A.bit	2	1	–	$A.\text{位} \leftarrow 1$			
		!addr16.bit	4	2	–	$(\text{addr16}).\text{位} \leftarrow 1$			
		PSW.bit	3	4	–	$\text{PSW.位} \leftarrow 1$	×	×	×
		[HL].bit	2	2	–	$(\text{HL}).\text{位} \leftarrow 1$			
		ES: !addr16.bit	5	3	–	$(\text{ES}, \text{addr16}).\text{位} \leftarrow 1$			
		ES: [HL].bit	3	3	–	$(\text{ES}, \text{HL}).\text{位} \leftarrow 1$			
	CLR1	saddr.bit	3	2	–	$(\text{saddr.位}) \leftarrow 0$			
		sfr.bit	3	2	–	$\text{sfr.位} \leftarrow 0$			
		A.bit	2	1	–	$A.\text{位} \leftarrow 0$			
		!addr16.bit	4	2	–	$(\text{addr16}).\text{位} \leftarrow 0$			
		PSW.bit	3	4	–	$\text{PSW.位} \leftarrow 0$	×	×	×
		[HL].bit	2	2	–	$(\text{HL}).\text{位} \leftarrow 0$			
		ES: !addr16.bit	5	3	–	$(\text{ES}, \text{addr16}).\text{位} \leftarrow 0$			
		ES: [HL].bit	3	3	–	$(\text{ES}, \text{HL}).\text{位} \leftarrow 0$			
	SET1	CY	2	1	–	$CY \leftarrow 1$			1
	CLR1	CY	2	1	–	$CY \leftarrow 0$			0
	NOT1	CY	2	1	–	$CY \leftarrow \overline{CY}$			×

- 注
1. 当访问内部 RAM 区或 SFR 区时，或者对于不进行数据访问的指令。
 2. 访问程序内存区时。

- 备注
1. 一个指令时钟周期是一个 CPU 时钟 (f_{CPU}) 周期，而 CPU 时钟由系统时钟控制寄存器 (CKC) 选择。
 2. 时钟编号用于当程序在内部 ROM (flash 存储器) 区时。

表 27-5. 操作列表 (15/17)

指令组	助记符	操作数	字节	时钟		操作	标志		
				注 1	注 2		Z	AC	CY
调用 / 返回	CALL	rp	2	3	–	$(SP - 2) \leftarrow (PC + 2)_S, (SP - 3) \leftarrow (PC + 2)_H,$ $(SP - 4) \leftarrow (PC + 2)_L, PC \leftarrow CS, rp,$ $SP \leftarrow SP - 4$			
		\$!addr20	3	3	–	$(SP - 2) \leftarrow (PC + 3)_S, (SP - 3) \leftarrow (PC + 3)_H,$ $(SP - 4) \leftarrow (PC + 3)_L, PC \leftarrow PC + 3 +$ jdisp16, $SP \leftarrow SP - 4$			
		!addr16	3	3	–	$(SP - 2) \leftarrow (PC + 3)_S, (SP - 3) \leftarrow (PC + 3)_H,$ $(SP - 4) \leftarrow (PC + 3)_L, PC \leftarrow 0000, \text{addr16},$ $SP \leftarrow SP - 4$			
		!!addr20	4	3	–	$(SP - 2) \leftarrow (PC + 4)_S, (SP - 3) \leftarrow (PC + 4)_H,$ $(SP - 4) \leftarrow (PC + 4)_L, PC \leftarrow \text{addr20},$ $SP \leftarrow SP - 4$			
	CALLT	[addr5]	2	5	–	$(SP - 2) \leftarrow (PC + 2)_S, (SP - 3) \leftarrow (PC + 2)_H,$ $(SP - 4) \leftarrow (PC + 2)_L, PC_S \leftarrow 0000,$ $PC_H \leftarrow (0000, \text{addr5} + 1),$ $PC_L \leftarrow (0000, \text{addr5}),$ $SP \leftarrow SP - 4$			
	BRK	–	2	5	–	$(SP - 1) \leftarrow PSW, (SP - 2) \leftarrow (PC + 2)_S,$ $(SP - 3) \leftarrow (PC + 2)_H, (SP - 4) \leftarrow (PC + 2)_L,$ $PC_S \leftarrow 0000,$ $PC_H \leftarrow (0007FH), PC_L \leftarrow (0007EH),$ $SP \leftarrow SP - 4, IE \leftarrow 0$			
	RET	–	1	6	–	$PC_L \leftarrow (SP), PC_H \leftarrow (SP + 1),$ $PC_S \leftarrow (SP + 2), SP \leftarrow SP + 4$			
	RETI	–	2	6	–	$PC_L \leftarrow (SP), PC_H \leftarrow (SP + 1),$ $PC_S \leftarrow (SP + 2), PSW \leftarrow (SP + 3),$ $SP \leftarrow SP + 4$	R	R	R
	RETB	–	2	6	–	$PC_L \leftarrow (SP), PC_H \leftarrow (SP + 1),$ $PC_S \leftarrow (SP + 2), PSW \leftarrow (SP + 3),$ $SP \leftarrow SP + 4$	R	R	R

- 注
1. 当访问内部 RAM 区或 SFR 区时，或者对于不进行数据访问的指令。
 2. 访问程序内存区时。

- 备注
1. 一个指令时钟周期是一个 CPU 时钟（f_{CPU}）周期，而 CPU 时钟由系统时钟控制寄存器（CKC）选择。
 2. 时钟编号用于当程序在内部 ROM（flash 存储器）区时。

表 27-5. 操作列表 (16/17)

指令组	助记符	操作数	字节	时钟		操作	标志		
				注 1	注 2		Z	AC	CY
栈操作	PUSH	PSW	2	1	—	$(SP - 1) \leftarrow PSW, (SP - 2) \leftarrow 00H,$ $SP \leftarrow SP - 2$			
		rp	1	1	—	$(SP - 1) \leftarrow rp_H, (SP - 2) \leftarrow rp_L,$ $SP \leftarrow SP - 2$			
	POP	PSW	2	3	—	$PSW \leftarrow (SP + 1), SP \leftarrow SP + 2$	R	R	R
		rp	1	1	—	$rp_L \leftarrow (SP), rp_H \leftarrow (SP + 1), SP \leftarrow SP + 2$			
	MOVW	SP, #word	4	1	—	SP ← 字			
		SP, AX	2	1	—	SP ← AX			
		AX, SP	2	1	—	AX ← SP			
		HL, SP	3	1	—	HL ← SP			
		BC, SP	3	1	—	BC ← SP			
		DE, SP	3	1	—	DE ← SP			
	ADDW	SP, #byte	2	1	—	SP ← SP + 字节			
	SUBW	SP, #byte	2	1	—	SP ← SP - 字节			
无条件转移	BR	AX	2	3	—	PC ← CS, AX			
		\$addr20	2	3	—	PC ← PC + 2 + jdisp8			
		!addr20	3	3	—	PC ← PC + 3 + jdisp16			
		!addr16	3	3	—	PC ← 0000, addr16			
		!!addr20	4	3	—	PC ← addr20			
条件转移	BC	\$addr20	2	2/4 ^{注3}	—	如果 CY = 1, PC ← PC + 2 + jdisp8			
	BNC	\$addr20	2	2/4 ^{注3}	—	如果 CY = 0, PC ← PC + 2 + jdisp8			
	BZ	\$addr20	2	2/4 ^{注3}	—	如果 Z = 1, PC ← PC + 2 + jdisp8			
	BNZ	\$addr20	2	2/4 ^{注3}	—	如果 Z = 0, PC ← PC + 2 + jdisp8			
	BH	\$addr20	3	2/4 ^{注3}	—	如果 (Z ∨ CY)=0, PC ← PC+3+jdisp8			
	BNH	\$addr20	3	2/4 ^{注3}	—	如果 (Z ∨ CY)=1, PC ← PC+3+jdisp8			
	BT	saddr.bit, \$addr20	4	3/5 ^{注3}	—	如果 (saddr).位 = 1, PC ← PC + 4 + jdisp8			
		sfr.bit, \$addr20	4	3/5 ^{注3}	—	如果 sfr.位 = 1, PC ← PC + 4 + jdisp8			
		A.bit, \$addr20	3	3/5 ^{注3}	—	如果 A.位 = 1, PC ← PC + 3 + jdisp8			
		PSW.bit, \$addr20	4	3/5 ^{注3}	—	如果 PSW.位 = 1, PC ← PC + 4 + jdisp8			
		[HL].bit, \$addr20	3	3/5 ^{注3}	6/8	如果 (HL).位 = 1, PC ← PC + 3 + jdisp8			
		ES : [HL].bit, \$addr20	4	4/6 ^{注3}	7/9	如果 (ES, HL).位 = 1, PC ← PC + 4 + jdisp8			

- 注
1. 当访问内部 RAM 区或 SFR 区时，或者对于不进行数据访问的指令。
 2. 访问程序内存区时。
 3. 这表示“当条件不满足/当条件满足”时时钟的编号。

- 备注
1. 一个指令时钟周期是一个 CPU 时钟 (f_{CPU}) 周期，而 CPU 时钟由系统时钟控制寄存器 (CKC) 选择。
 2. 时钟编号用于当程序在内部 ROM (flash 存储器) 区时。

表 27-5. 操作列表 (17/17)

指令组	助记符	操作数	字节	时钟		操作	标志		
				注 1	注 2		Z	AC	CY
转移条件	BF	saddr.bit, \$addr20	4	3/5 ^{注3}	—	如果 (saddr).位 = 0, PC ← PC + 4 + jdisp8			
		sfr.bit, \$addr20	4	3/5 ^{注3}	—	如果 sfr.位 = 0, PC ← PC + 4 + jdisp8			
		A.bit, \$addr20	3	3/5 ^{注3}	—	如果 A.位 = 0, PC ← PC + 3 + jdisp8			
		PSW.bit, \$addr20	4	3/5 ^{注3}	—	如果 PSW.位 = 0, PC ← PC + 4 + jdisp8			
		[HL].bit, \$addr20	3	3/5 ^{注3}	6/8	如果 (HL).位 = 0, PC ← PC + 3 + jdisp8			
		ES : [HL].bit , \$addr20	4	4/6 ^{注3}	7/9	如果 (ES, HL).位 = 0, PC ← PC + 4 + jdisp8			
	BTCLR	saddr.bit, \$addr20	4	3/5 ^{注3}	—	如果 (saddr).位 = 1, PC ← PC + 4 + jdisp8, 然后 (saddr).位复位			
		sfr.bit, \$addr20	4	3/5 ^{注3}	—	如果 sfr.位 = 1, PC ← PC + 4 + jdisp8 然后 sfr.位复位			
		A.bit, \$addr20	3	3/5 ^{注3}	—	如果 A.位 = 1, PC ← PC + 3 + jdisp8 然后 A.位复位			
		PSW.bit, \$addr20	4	3/5 ^{注3}	—	如果 PSW.位 = 1, PC ← PC + 4 + jdisp8 然后 PSW.位复位	×	×	×
		[HL].bit, \$addr20	3	3/5 ^{注3}	—	如果 (HL).位 = 1, PC ← PC + 3 + jdisp8 然后 (HL).位复位			
		ES : [HL].bit , \$addr20	4	4/6 ^{注3}	—	如果 (ES, HL).位 = 1, PC ← PC + 4 + jdisp8, 然后 (ES, HL).位复位			
有条件的跳过	SKC	—	2	1	—	如果 CY = 1, 下一个指令跳过			
	SKNC	—	2	1	—	如果 CY = 0, 下一个指令跳过			
	SKZ	—	2	1	—	如果 Z = 1, 下一个指令跳过			
	SKNZ	—	2	1	—	如果 Z = 0, 下一个指令跳过			
	SKH	—	2	1	—	如果 (Z ∨ CY) = 0, 下一个指令跳过			
	SKNH	—	2	1	—	如果 (Z ∨ CY) = 1, 下一个指令跳过			
CPU 控制	SEL	RBn	2	1	—	RBS[1: 0] ← n			
	NOP	—	1	1	—	没有操作			
	EI	—	3	4	—	IE ← 1 (使能中断)			
	DI	—	3	4	—	IE ← 0 (禁止中断)			
	HALT	—	2	3	—	设置 HALT 模式			
	STOP	—	2	3	—	设置 STOP 模式			

- 注
1. 当访问内部 RAM 区或 SFR 区时, 或者对于不进行数据访问的指令。
 2. 访问程序内存区时。
 3. 这表示“当条件不满足/当条件满足”时时钟的编号。

- 备注
1. 一个指令时钟周期是一个 CPU 时钟 (f_{cpu}) 周期, 而 CPU 时钟由系统时钟控制寄存器 (CKC) 选择。
 2. 时钟编号用于当程序在内部 ROM (flash 存储器) 区时。
 3. n 表示寄存器组的编号 (n = 0 至 3)。

第二十八章 电气特性（目标）

注意事项 1. 78K0R/Kx3-L 的这些规范是目标值，在器件评估后可能会发生变化。

2. 78K0R/Kx3-L 具有片上调试功能，用于开发和评估。请不要在量产时使用该产品的片上调试功能，因为使用该功能时，可能会超出 flash 内存能保证的可重写次数，因此产品当可靠性得不到保证。NEC 电子对使用片上调试功能时产生的问题概不负责。

3. 安装的引脚根据产品不同如下所示：

（1）端口功能

端口	78K0R/KC3-L		78K0R/KD3-L	78K0R/KE3-L
	44 引脚	48 引脚	52 引脚	64 引脚
端口 0	—	—	P00, P01	
端口 1	P10 至 P13			P10 至 P17
端口 2	P20 至 P27			
端口 3	P30 至 P32			P30 至 P33
端口 4	P40, P41			P40 至 P43
端口 5	P50 至 P52			P50 至 P53
端口 6	—	P60, P61		
端口 7	P70 至 P75		P70 至 P77	
端口 8	P80 至 P83			
端口 12	P120 至 P124			
端口 14	—	P140		P140, P141
端口 15	P150, P151	P150 至 P152		P150 至 P153

（2）非端口功能（1/2）

功能名称	78K0R/KC3-L		78K0R/KD3-L	78K0R/KE3-L
	44 引脚	48 引脚	52 引脚	64 引脚
供电电源， 接地	V _{DD} , AV _{REF} , V _{SS} , AV _{SS}			V _{DD} , EV _{DD} , AV _{REF} , V _{SS} , EV _{SS} , AV _{SS}
稳压器	REGC			
复位	RESET			
时钟振荡	X1, X2, XT1, XT2, EXCLK			
写入 flash 存储器	FLMD0			
中断	INTP0-INTP7			
定时器	SLTI, SLTO, TI02 至 TI07, TO02-TO07		SLTI, SLTO, TI00, TI02 至 TI07, TO00, TO02 至 TO07	
实时计数器	RTCDIV, RTCCL, RTC1HZ			
比较器	CMP0M, CMP0P, CMP1M, CMP1P			
可编程增益放大器	PGAI			

(2) 非端口功能（2/2）

功能名称		78K0R/KC3-L		78K0R/KD3-L	78K0R/KE3-L
		44 引脚	48 引脚	52 引脚	64 引脚
串行接口	UART0	RxD0, TxD0			
	UART1	RxD1, TxD1			
	CSI00	SCK00, SI00, SO00			
	CSI01	SCK01, SI01, SO01			
	CSI10	SCK10, SI10, SO10			
	IIC10	SCL10, SDA10			
	IICA	—	SCL0, SDA0		
A/D 转换器		ANI0 至 ANI9	ANI0 至 ANI10		ANI0 至 ANI11
时钟输出/蜂鸣器输出		—	PCLBUZ0		PCLBUZ0, PCLBUZ1
按键中断		KR0 至 KR5		KR0 至 KR7	
低压检测器（LVI）		EXLVI			
片上调试功能		TOOL0, TOOL1			

注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

最大额定值（ $T_A = 25^\circ\text{C}$ ）（1/2）

参数	符号	条件	最大额定值范围	单位
供应电压	V_{DD}		-0.5 至 +6.5	V
	EV_{DD}		-0.5 至 +6.5	V
	V_{SS}		-0.5 至 +0.3	V
	EV_{SS}		-0.5 至 +0.3	V
	AV_{REF}		-0.5 至 $V_{DD} + 0.3^{\#1}$	V
	AV_{SS}		-0.5 至 +0.3	V
REGC 引脚输入电压	V_{IREGC}	REGC	-0.3 至 3.6 和 -0.3 至 $V_{DD} + 0.3^{\#2}$	V
输入电压	V_{I1}	P00, P01, P10 至 P17, P30 至 P33, P40 至 P43, P50 至 P53, P70 至 P77, P120 至 P124, P141, EXCLK, RESET, FLMD0	-0.3 至 $EV_{DD} + 0.3$ 和 -0.3 至 $V_{DD} + 0.3^{\#1}$	V
	V_{I2}	P60, P61（N 沟道漏极开路）	-0.3 至 +6.5	V
	V_{I3}	P20 至 P27, P80 至 P83, P150 至 P153	-0.3 至 $AV_{REF} + 0.3$ 和 -0.3 至 $V_{DD} + 0.3^{\#1}$	V
输出电压	V_{O1}	P00, P01, P10 至 P17, P30 至 P33, P40 至 P43, P50 至 P53, P60, P61, P70 至 P77, P120, P140, P141	-0.3 至 $EV_{DD} + 0.3^{\#1}$	V
	V_{O2}	P20 至 P27, P80 至 P83, P150 至 P153	-0.3 至 $AV_{REF} + 0.3$	V
模拟输入电压	V_{AN}	ANI0 至 ANI11, PGAI, CMP0M, CMP0P, CMP1M, CMP1P	-0.3 至 $AV_{REF} + 0.3^{\#1}$ 和 -0.3 至 $V_{DD} + 0.3^{\#1}$	V

- 注**
1. 必须为 6.5 V 或更低的电压。
 2. 通过电容（0.47 至 1 μF ：目标）将 REGC 引脚连接到 V_{SS} 。该值调整 REGC 引脚的最大额定范围。向其供应电压时禁止使用该引脚。

注意事项 任何参数，哪怕瞬间超出其最大额定值，都可能使产品性能受损。就是说，工作在最大额定值时产品处于物理损坏的边缘，所以，产品必须在不超出其最大额定值的条件下使用。

备注 除非另外说明，复用功能引脚的特性与端口引脚的相同。

注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

最大额定值（ $T_A = 25^\circ\text{C}$ ）（2/2）

参数	符号	条件		最大额定值范围	单位
输出电流，高	I _{OH1}	各引脚	P00, P01, P10 至 P17, P30 至 P33, P40 至 P43, P50 至 P53, P70 至 P77, P120, P140, P141	-10	mA
		全部引脚总和 -80 mA	P00, P01, P40 至 P43, P120, P140, P141	-25	mA
			P10 至 P17, P30 至 P33, P50 至 P53, P70 至 P77	-55	mA
	I _{OH2}	各引脚	P20 至 P27, P80 至 P83,	-0.5	mA
		全部引脚总和	P150 至 P153		-2
输出电流，低	I _{OL1}	各引脚	P00, P01, P10 至 P17, P30 至 P33, P40 至 P43, P50 至 P53, P60, P61, P70 至 P77, P120, P140, P141	30	mA
		全部引脚总和 200 mA	P00, P01, P40 至 P43, P120, P140, P141	60	mA
			P10 至 P17, P30 至 P33, P50 至 P53, P60, P61, P70 至 P77	140	mA
	I _{OL2}	各引脚	P20 至 P27, P80 至 P83,	1	mA
		全部引脚总和	P150 至 P153	5	mA
工作环境温度	T _A	正常操作模式下		-40 至 +85	°C
		Flash 存储器编程模式下			
储存温度	T _{stg}			-65 至 +150	°C

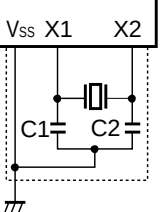
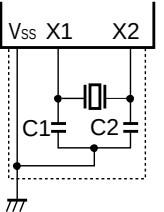
注意事项 任何参数，哪怕瞬间超出其最大额定值，都可能使产品性能受损。就是说，工作在最大额定值时产品处于物理损坏的边缘，所以，产品必须在不超出其最大额定值的条件下使用。

备注 除非另外说明，复用功能引脚的特性与端口引脚的相同。

注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

X1 振荡器特性

($T_A = -40$ 至 $+85^{\circ}\text{C}$, $1.8\text{ V} \leq V_{DD} = EV_{DD} \leq 5.5\text{ V}$, $V_{SS} = EV_{SS} = AV_{SS} = 0\text{ V}$)

振荡器	推荐电路	参数	条件	最小值	典型值	最大值	单位
陶瓷振荡器		X1 时钟振荡频率 (f_x) ^註	$2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}$	2.0		20.0	MHz
			$1.8\text{ V} \leq V_{DD} < 2.7\text{ V}$	2.0		5.0	
晶体振荡器		X1 时钟振荡频率 (f_x) ^註	$2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}$	2.0		20.0	MHz
			$1.8\text{ V} \leq V_{DD} < 2.7\text{ V}$	2.0		5.0	

注 仅指示振荡器特性关于指令执行时间参见 AC 特性。

注意事项 1. 当使用 X1 振荡器时，为了避免连接电容引起的负作用，应该如下连接上图虚线中的电路。

- 保证连线尽可能短
- 不要将该连线和其它信号线交叉
- 布线时不要将该连线靠近通过大波动电流的信号线
- 保证振荡器电容的接地点和 V_{SS} 相同的电位
- 不要将电容连接到一个大电流通过的接地点
- 不要从振荡器获取信号

2. 复位解除后，在 CPU 通过内部高速振荡时钟启动后，由用户使用振荡稳定时间计数器状态寄存器（OSTC）检查 X1 时钟振荡稳定时间。在用谐振器充分评估振荡稳定时间后，计算 OSTC 寄存器和振荡稳定时间选择寄存器（OSTS）的振荡稳定数据。

备注 有关谐振器的选择和振荡器常量，需要用户自行评估振荡特性或者向谐振器厂商申请谐振器评估。

注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

内部振荡器特性

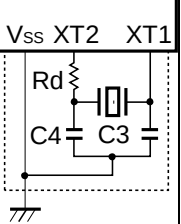
($T_A = -40$ 至 $+85^{\circ}\text{C}$, $1.8\text{ V} \leq V_{DD} = EV_{DD} \leq 5.5\text{ V}$, $V_{SS} = EV_{SS} = AV_{SS} = 0\text{ V}$)

振荡器	参数	条件	最小值	典型值	最大值	单位
内部高速振荡时钟频率 注	f_{IH1M}			1		MHz
	f_{IH8M}	$2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}$		8		MHz
		$1.8\text{ V} \leq V_{DD} < 2.7\text{ V}$		8		MHz
	f_{IH20M}	$2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}$		20		MHz
内部低速振荡时钟频率	f_{IL}			30		kHz

注 仅指示振荡器特性。关于指令执行时间参见 AC 特性。

XT1 振荡器特性

($T_A = -40$ 至 $+85^{\circ}\text{C}$, $1.8\text{ V} \leq V_{DD} = EV_{DD} \leq 5.5\text{ V}$, $V_{SS} = EV_{SS} = AV_{SS} = 0\text{ V}$)

振荡器	推荐电路	项	条件	最小值	典型值	最大值	单位
晶体振荡器		XT1 时钟振荡频率 (f_{XT}) 注		32	32.768	35	kHz

注 仅指示振荡器特性关于指令执行时间参见 AC 特性。

注意事项 1. 当使用 XT1 振荡器时，为了避免连接电容引起的负作用，应该如下连接上图虚线中的电路。

- 保证连线尽可能短
- 不要将该连线和其它信号线交叉
- 布线时不要将该连线靠近通过大波动电流的信号线
- 保证振荡器电容的接地点和 V_{SS} 相同的电位
- 不要将电容连接到一个大电流通过的接地点
- 不要从振荡器获取信号

2. 为了减少功耗，XT1 荡器被设计为低振幅电路，而比起 X1 振荡器，其更容易受由于噪声引起的干扰。所以，使用 XT1 时钟时需要特别注意布线的方式。

备注 有关振荡器的选择和振荡器常量，需要用户自行评估振荡特性或者向谐振器厂商申请谐振器评估。

注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

推荐的振荡器电路常量

XT1 振荡：晶体振荡器（T_A = -40 至 +85°C）

生产商	部件编号	SMD/ 铅	频率 （MHz）	加载电容 CL （pF）	XT1 振荡器振荡模式 ^註	推荐的电路常量		振荡电压范围	
						C1 （pF）	C2 （pF）	最小 值 （V）	最大 值 （V）
Seiko Instruments Inc.	SP-T2A	SMD	32.768	4.4	低耗振荡	7	7	1.8	5.5
					超低耗振荡	7	7		
	VT-200	铅		6.0	低耗振荡	5	5		
					超低耗振荡	5	5		
	SSP-T7	小 SMD		3.7	低耗振荡	4	3		
					超低耗振荡	4	3		

注 用时钟操作模式控制寄存器（CMC）的位 AMPHS1 和 AMPHS0 设置 XT1 振荡模式。

注意事项 以上显示的振荡器常量是基于振荡器厂商在特殊环境下进行评估的参考值。实际应用中如果需要优化振荡器特性，可项振荡器厂商申请对执行电路进行评估。振荡电压和振荡频率仅表示振荡器的特性。使用 78K0R/Kx3-L 使得内部操作条件在 DC 特性和 AC 特性范围内。

注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

DC 特性（1/11）

（ $T_A = -40$ 至 $+85^\circ\text{C}$ ， $1.8\text{ V} \leq V_{DD} = EV_{DD} \leq 5.5\text{ V}$ ， $1.8\text{ V} \leq AV_{REF} \leq V_{DD}$ ， $V_{SS} = EV_{SS} = AV_{SS} = 0\text{ V}$ ）

参数	符号	条件	最小值	典型值	最大值	单位
输出电路，高 ^{注1}	I_{OH1}	P00, P01, P10 至 P17, P30 至 P33, P40 至 P43, P50 至 P53, P70 至 P77, P120, P140, P141 的各引脚	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$		-3.0	mA
			$2.7\text{ V} \leq V_{DD} < 4.0\text{ V}$		-1.0	mA
			$1.8\text{ V} \leq V_{DD} < 2.7\text{ V}$		-1.0	mA
		P00, P01, P40 至 P43, P120, P140, P141 的总和 (占空比 = 70% ^{注2})	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$		-20.0	mA
			$2.7\text{ V} \leq V_{DD} < 4.0\text{ V}$		-10.0	mA
			$1.8\text{ V} \leq V_{DD} < 2.7\text{ V}$		-5.0	mA
		P10 至 P17, P30 至 P33, P50 至 P53, P70 至 P77 的总和 (占空比 = 70% ^{注2})	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$		-30.0	mA
			$2.7\text{ V} \leq V_{DD} < 4.0\text{ V}$		-19.0	mA
			$1.8\text{ V} \leq V_{DD} < 2.7\text{ V}$		-10.0	mA
		全部引脚总和 (占空比 = 60% ^{注2})	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$		-50.0	mA
			$2.7\text{ V} \leq V_{DD} < 4.0\text{ V}$		-29.0	mA
			$1.8\text{ V} \leq V_{DD} < 2.7\text{ V}$		-15.0	mA
	I_{OH2}	P20 至 P27, P80 至 P83, P150 至 P153 的各引脚	$AV_{REF} \leq V_{DD}$		-0.1	mA

注 1. 即使电流从 EV_{DD} 流向输出引脚，也可保证设备操作的电流值。

2. 占空比为 60% 或 70% 的条件下的特性。

可用下式计算改变占空比的输出电流值（当占空比从 70% 变成 $n\%$ 时）

• 引脚的输出电路总和 = $(I_{OH} \times 0.7) / (n \times 0.01)$

<举例> 在 $n = 50\%$ 而 $I_{OH} = 20.0\text{ mA}$ 处

引脚的输出电路总和 = $(-20.0 \times 0.7) / (50 \times 0.01) = -28.0\text{ mA}$

但是，允许流进引脚的电流并不根据占空比而发生变化。大于最大额定范围的电流禁止流入引脚。

注意事项 在 N 沟道漏极开路模式下，P30 至 P32，P70，P72，P73 以及 P75 不输出高电平。

备注 除非另外说明，复用功能引脚的特性与端口引脚的相同。

注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

DC 特性（2/11）

（ $T_A = -40$ 至 $+85^{\circ}\text{C}$ ， $1.8\text{ V} \leq V_{DD} = EV_{DD} \leq 5.5\text{ V}$ ， $1.8\text{ V} \leq AV_{REF} \leq V_{DD}$ ， $V_{SS} = EV_{SS} = AV_{SS} = 0\text{ V}$ ）

参数	符号	条件	最小值	典型值	最大值	单位
输出电流，低 ^{註1}	I _{OL1}	P00, P01, P10 至 P17, P30 至 P33, P40 至 P43, P50 至 P53, P70 至 P77, P120, P140, P141 的各引脚	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$		8.5	mA
			$2.7\text{ V} \leq V_{DD} < 4.0\text{ V}$		1.0	mA
			$1.8\text{ V} \leq V_{DD} < 2.7\text{ V}$		0.5	mA
		P60, P61 的各引脚	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$		15.0	mA
			$2.7\text{ V} \leq V_{DD} < 4.0\text{ V}$		3.0	mA
			$1.8\text{ V} \leq V_{DD} < 2.7\text{ V}$		2.0	mA
		P00, P01, P40 至 P43, P120, P140, P141 的总和 （占空比 = 70% ^{註2} ）	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$		20.0	mA
			$2.7\text{ V} \leq V_{DD} < 4.0\text{ V}$		15.0	mA
			$1.8\text{ V} \leq V_{DD} < 2.7\text{ V}$		9.0	mA
		P10 至 P17, P30 至 P33, P50 to P53, P60, P61, P70 至 P77 的总和 （占空比 = 70% ^{註2} ）	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$		45.0	mA
			$2.7\text{ V} \leq V_{DD} < 4.0\text{ V}$		35.0	mA
			$1.8\text{ V} \leq V_{DD} < 2.7\text{ V}$		20.0	mA
		全部引脚总和 （占空比 = 60% ^{註2} ）	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$		65.0	mA
			$2.7\text{ V} \leq V_{DD} < 4.0\text{ V}$		40.0	mA
			$1.8\text{ V} \leq V_{DD} < 2.7\text{ V}$		29.0	mA
	I _{OL2}	P20 至 P27, P80 至 P83, P150 至 P153 的各引脚	$AV_{REF} \leq V_{DD}$		0.4	mA

注 1. 即使电流从输出引脚流向 EV_{SS}，V_{SS} 以及 AV_{SS} 引脚，也可保证设备操作的电流值。

2. 占空比为 60% 或 70% 的条件下的特性。

可用下式计算改变占空比的输出电流值（当占空比从 70% 变成 n% 时）

• 引脚输出电流总和 = $(I_{OL} \times 0.7) / (n \times 0.01)$

<举例> 在 $n = 50\%$ 而 $I_{OL} = 20.0\text{ mA}$ 处

引脚输出电流总和 = $(20.0 \times 0.7) / (50 \times 0.01) = 28.0\text{ mA}$

但是，允许流进引脚的电流并不根据占空比而发生变化。大于最大额定范围的电流禁止流入引脚。

备注 除非另外说明，复用功能引脚的特性与端口引脚的相同。

注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

DC 特性（3/11）

（ $T_A = -40$ 至 $+85^{\circ}\text{C}$ ， $1.8\text{ V} \leq V_{DD} = EV_{DD} \leq 5.5\text{ V}$ ， $1.8\text{ V} \leq AV_{REF} \leq V_{DD}$ ， $V_{SS} = EV_{SS} = AV_{SS} = 0\text{ V}$ ）

参数	符号	条件		最小值	典型值	最大值	单位
输入电压，高	V_{IH1}	P01, P30, P33, P42, P43, P53, P123, P124, P141		$0.7V_{DD}$		V_{DD}	V
	V_{IH2}	P00, P10 至 P17, P31, P32, P40, P41, P50 至 P52, P70 至 P77, P120 至 P122, EXCLK, $\overline{\text{RESET}}$	正常输入缓存	$0.8V_{DD}$		V_{DD}	V
	V_{IH3}	P31, P32, P71, P72, P74, P75	TTL 输入缓存 $4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$	2.2		V_{DD}	V
			TTL 输入缓存 $2.7\text{ V} \leq V_{DD} < 4.0\text{ V}$	2.0		V_{DD}	V
			TTL 输入缓存 $1.8\text{ V} \leq V_{DD} < 2.7\text{ V}$	1.6		V_{DD}	V
	V_{IH4}	P20 至 P27, P81, P83, P150 至 P153	$2.7\text{ V} \leq AV_{REF} \leq V_{DD}$	$0.7AV_{REF}$		AV_{REF}	V
			$AV_{REF} = V_{DD} < 2.7\text{ V}$				
	V_{IH5}	P80, P82	$AV_{REF} = V_{DD}$	$0.8AV_{REF}$		AV_{REF}	V
	V_{IH6}	P60, P61		$0.7V_{DD}$		6.0	V
	V_{IH7}	FLMD0		$0.9V_{DD}$ 注		V_{DD}	V

注 当用于 flash 内存编程模式时必须为 $0.9V_{DD}$ 或更高的电压。

注意事项 即使在 N 沟道漏极开路模式下，P30 至 P32，P70，P72，P73 以及 P75 的 V_{IH} 的最大值为 V_{DD} 。

备注 除非另外说明，复用功能引脚的特性与端口引脚的相同。

注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

DC 特性（4/11）

（ $T_A = -40$ 至 $+85^{\circ}\text{C}$ ， $1.8\text{ V} \leq V_{DD} = EV_{DD} \leq 5.5\text{ V}$ ， $1.8\text{ V} \leq AV_{REF} \leq V_{DD}$ ， $V_{SS} = EV_{SS} = AV_{SS} = 0\text{ V}$ ）

参数	符号	条件	最小值	典型值	最大值	单位
输入电压，低	V_{IL1}	P01, P30, P33, P42, P43, P53, P123, P124, P141	0		$0.3V_{DD}$	V
	V_{IL2}	P00, P10 至 P17, P31, P32, P40, P41, P50 至 P52, P70 至 P77, P120 至 P122, EXCLK, $\overline{\text{RESET}}$	0		$0.2V_{DD}$	V
	V_{IL3}	P31, P32, P71, P72, P74, P75	TTL 输入缓存 $4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$	0	0.8	V
			TTL 输入缓存 $2.7\text{ V} \leq V_{DD} < 4.0\text{ V}$	0	0.5	V
			TTL 输入缓存 $1.8\text{ V} \leq V_{DD} < 2.7\text{ V}$	0	0.2	V
	V_{IL4}	P20 至 P27, P81, P83, P150 至 P153	$2.7\text{ V} \leq AV_{REF} \leq V_{DD}$	0	$0.3AV_{REF}$	V
			$AV_{REF} = V_{DD} < 2.7\text{ V}$			
	V_{IL5}	P80, P82	$AV_{REF} = V_{DD}$	0	$0.2AV_{REF}$	V
	V_{IL6}	P60, P61	0		$0.3V_{DD}$	V
	V_{IL7}	FLMD0 [※]	0		$0.1V_{DD}$	V

注 当 flash 内存禁止写入时，将 FLMD0 引脚直接连接到 V_{SS} ，并维持一个低于 $0.1V_{DD}$ 的电压。

备注 除非另外说明，复用功能引脚的特性与端口引脚的相同。

注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

DC 特性（5/11）

$(T_A = -40$ 至 $+85^{\circ}\text{C}$, $1.8\text{ V} \leq V_{DD} = EV_{DD} \leq 5.5\text{ V}$, $1.8\text{ V} \leq AV_{REF} \leq V_{DD}$, $V_{SS} = EV_{SS} = AV_{SS} = 0\text{ V}$)

参数	符号	条件	最小值	典型值	最大值	单位
输出电压，高	V _{OH1}	P00, P01, P10 至 P17, P30 至 P33, P40 至 P43, P50 至 P53, P70 至 P77, P120, P140, P141	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $I_{OH1} = -3.0\text{ mA}$	$V_{DD} - 0.7$		V
			$1.8\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $I_{OH1} = -1.0\text{ mA}$	$V_{DD} - 0.5$		V
	V _{OH2}	P20 至 P27, P80 至 P83, P150 至 P153	$AV_{REF} \leq V_{DD}$, $I_{OH2} = -0.1\text{ mA}$	$AV_{REF} - 0.5$		V
输出电压，低	V _{OL1}	P00, P01, P10 至 P17, P30 至 P33, P40 至 P43, P50 至 P53, P70 至 P77, P120, P140, P141	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $I_{OL1} = 8.5\text{ mA}$		0.7	V
			$2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $I_{OL1} = 1.0\text{ mA}$		0.5	V
			$1.8\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $I_{OL1} = 0.5\text{ mA}$		0.4	V
	V _{OL2}	P20 至 P27, P80 至 P83, P150 至 P153	$AV_{REF} = V_{DD}$, $I_{OL2} = 0.4\text{ mA}$		0.4	V
	V _{OL3}	P60, P61	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $I_{OL1} = 15.0\text{ mA}$		2.0	V
			$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $I_{OL1} = 5.0\text{ mA}$		0.4	V
			$2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $I_{OL1} = 3.0\text{ mA}$		0.4	V
			$1.8\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $I_{OL1} = 2.0\text{ mA}$		0.4	V

注意事项 在 N 沟道漏极开路模式下，P30 至 P32，P70，P72，P73 以及 P75 不输出高电平。

备注 除非另外说明，复用功能引脚的特性与端口引脚的相同。

注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

DC 特性（6/11）

（ $T_A = -40$ 至 $+85^{\circ}\text{C}$ ， $1.8\text{ V} \leq V_{DD} = EV_{DD} \leq 5.5\text{ V}$ ， $1.8\text{ V} \leq AV_{REF} \leq V_{DD}$ ， $V_{SS} = EV_{SS} = AV_{SS} = 0\text{ V}$ ）

参数	符号	条件	最小值	典型值	最大值	单位
输入漏电流，高	I _{LIH1}	P00, P01, P10 至 P17, P30 至 P33, P40 至 P43, P50 至 P53, P60, P61, P70 至 P77, P120, P141, FLMD0, RESET	$V_I = V_{DD}$		1	μA
	I _{LIH2}	P20 至 P27, P80 至 P83, P150 至 P153	$V_I = AV_{REF}$, $2.7\text{ V} \leq AV_{REF} \leq V_{DD}$		1	μA
			$V_I = AV_{REF}$, $AV_{REF} = V_{DD} < 2.7\text{ V}$			
	I _{LIH3}	P121 至 P124 (X1, X2, XT1, XT2)	$V_I = V_{DD}$	在输入端口	1	μA
输入漏电流，低	I _{LIL1}	P00, P01, P10 至 P17, P30 至 P33, P40 至 P43, P50 至 P53, P60, P61, P70 至 P77, P120, P141, FLMD0, RESET	$V_I = V_{SS}$		-1	μA
	I _{LIL2}	P20 至 P27, P80 至 P83, P150 至 P153	$V_I = V_{SS}$, $2.7\text{ V} \leq AV_{REF} \leq V_{DD}$		-1	μA
			$V_I = V_{SS}$, $AV_{REF} = V_{DD} < 2.7\text{ V}$			
	I _{LIL3}	P121 至 P124 (X1, X2, XT1, XT2)	$V_I = V_{SS}$	在输入端口	-1	μA
				在振荡器连接处	-10	μA

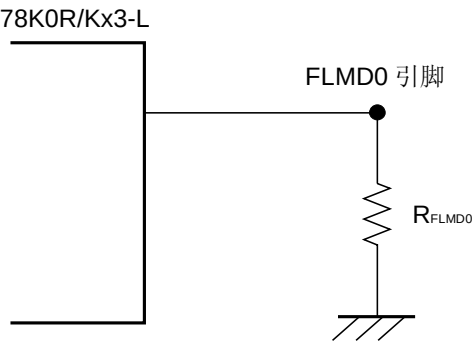
备注 除非另外说明，复用功能引脚的特性与端口引脚的相同。

注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

DC 特性（7/11）
（TA = -40 至 +85℃， 1.8 V ≤ VDD = EVDD ≤ 5.5 V， 1.8 V ≤ AVREF ≤ VDD， VSS = EVSS = AVSS = 0 V）

参数	符号	条件	最小值	典型值	最大值	单位
片上上拉电阻	Ru	P00, P01, P10 至 P17, P30 至 P33, P40 至 P43, P50 至 P53, P70 至 P77, P120, P141 Vi = Vss, 在输入端口	10	20	100	kΩ
FLMD0 引脚外部下拉电阻 ^注	RFLMD0	当允许用软件进行自编程模式设置时	100			kΩ

注 建议保持 FLMD0 引脚开路。如果需要将该引脚在外部下拉，将 RFLMD0 设置为 100 kΩ或更大。



备注 除非另外说明，复用功能引脚的特性与端口引脚的相同。

注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

DC 特性（8/11）

($T_A = -40$ 至 $+85^\circ\text{C}$, $1.8\text{ V} \leq V_{DD} = EV_{DD} \leq 5.5\text{ V}$, $1.8\text{ V} \leq AV_{REF} \leq V_{DD}$, $V_{SS} = EV_{SS} = AV_{SS} = 0\text{ V}$)

参数	符号	条件		最小值	典型值	最大值	单位	
供应电流	I _{DD1} ^{註 1}	操作模式	f _{MX} = 20 MHz ^{註 2} , V _{DD} = 5.0 V	方波输入		6.5	T.B.D	mA
				振荡器连接		6.8	T.B.D	mA
			f _{MX} = 20 MHz ^{註 2} , V _{DD} = 3.0 V	方波输入		6.5	T.B.D	mA
				振荡器连接		6.8	T.B.D	mA
			f _{MX} = 10 MHz ^{註 2, 3} , V _{DD} = 5.0 V	方波输入		3.5	T.B.D	mA
				振荡器连接		3.6	T.B.D	mA
			f _{MX} = 10 MHz ^{註 2, 3} , V _{DD} = 3.0 V	方波输入		3.5	T.B.D	mA
				振荡器连接		3.6	T.B.D	mA
			f _{MX} = 5 MHz ^{註 2, 3} , V _{DD} = 3.0 V	方波输入		1.9	T.B.D	mA
				振荡器连接		2.0	T.B.D	mA
			f _{MX} = 5 MHz ^{註 2, 3} , V _{DD} = 2.0 V	方波输入		1.5	T.B.D	mA
				振荡器连接		1.5	T.B.D	mA
			f _{IH20} = 20 MHz ^{註 4}	V _{DD} = 5.0 V		6.8	T.B.D	mA
				V _{DD} = 3.0 V		6.8	T.B.D	mA
			f _{IH} = 8 MHz ^{註 4}	V _{DD} = 5.0 V		2.8	T.B.D	mA
				V _{DD} = 3.0 V		2.8	T.B.D	mA
f _{IH} = 1 MHz ^{註 4}	V _{DD} = 3.0 V		200	T.B.D	μA			
f _{SUB} = 32.768 kHz ^{註 5} , T _A = -40 至 +70°C	V _{DD} = 5.0 V		5.0	T.B.D	μA			
	V _{DD} = 3.0 V		5.0	T.B.D	μA			
	V _{DD} = 2.0 V		5.0	T.B.D	μA			
f _{SUB} = 32.768 kHz ^{註 5} , T _A = -40 至 +85°C	V _{DD} = 5.0 V		5.0	T.B.D	μA			
	V _{DD} = 3.0 V		5.0	T.B.D	μA			
	V _{DD} = 2.0 V		5.0	T.B.D	μA			

- 注
1. 流入 V_{DD} , EV_{DD} 和 AV_{REF} 的全部电流, 包括当输入引脚固定为 V_{DD} 或 V_{SS} 时的输入漏电流。在最大值列下的值包括外设操作电流。但是不包括流入 A/D 转换器、可编程增益放大器、比较器、LVI 电路、I/O 端口以及片内上拉/下拉电阻的电流。
 2. 当内部高速振荡、20 MHz 内部高速振荡以及副系统时钟停止时。
 3. 当 AMPH（时钟操作模式控制寄存器（CMC）的位 0）= 0 时, FLPC 和 FSEL（操作速度模式控制寄存器（OSMC）的位 1 和 0）= 0。
 4. 当高速系统时钟和副系统时钟停止。
 5. 当内部高速振荡、20 MHz 内部高速振荡以及高速系统时钟停止时。当看门狗定时器停止时。

- 备注
1. f_{MX} : 高速系统时钟频率（X1 时钟振荡频率或外部主系统时钟频率）
 2. f_{IH20} : 20 MHz 内部高速振荡时钟频率
 3. f_{IH} : 内部高速振荡时钟频率
 4. f_{SUB} : 副系统时钟频率（XT1 时钟振荡频率）

注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

DC 特性（9/11）

（ $T_A = -40$ 至 $+85^{\circ}\text{C}$ ， $1.8\text{ V} \leq V_{DD} = EV_{DD} \leq 5.5\text{ V}$ ， $1.8\text{ V} \leq AV_{REF} \leq V_{DD}$ ， $V_{SS} = EV_{SS} = AV_{SS} = 0\text{ V}$ ）

参数	符号	条件		最小值	典型值	最大值	单位	
供应电流	I _{DD2} ^{註1}	HALT 模式	f _{MX} = 20 MHz ^{註2} , V _{DD} = 5.0 V	方波输入		1.1	T.B.D	mA
				振荡器连接		1.4	T.B.D	mA
			f _{MX} = 20 MHz ^{註2} , V _{DD} = 3.0 V	方波输入		1.1	T.B.D	mA
				振荡器连接		1.4	T.B.D	mA
			f _{MX} = 10 MHz ^{註2, 3} , V _{DD} = 5.0 V	方波输入		0.65	T.B.D	mA
				振荡器连接		0.75	T.B.D	mA
			f _{MX} = 10 MHz ^{註2, 3} , V _{DD} = 3.0 V	方波输入		0.65	T.B.D	mA
				振荡器连接		0.75	T.B.D	mA
			f _{MX} = 5 MHz ^{註2, 3} , V _{DD} = 3.0 V	方波输入		0.39	T.B.D	mA
				振荡器连接		0.44	T.B.D	mA
			f _{MX} = 5 MHz ^{註2, 3} , V _{DD} = 2.0 V	方波输入		0.3	T.B.D	mA
				振荡器连接		0.35	T.B.D	mA
			f _{IH20} = 20 MHz ^{註4}	V _{DD} = 5.0 V		1.4	T.B.D	mA
				V _{DD} = 3.0 V		1.4	T.B.D	mA
			f _{IH} = 8 MHz ^{註4}	V _{DD} = 5.0 V		0.45	T.B.D	mA
				V _{DD} = 3.0 V		0.45	T.B.D	mA
			f _{IH} = 1 MHz ^{註4}	V _{DD} = 5.0 V		65	T.B.D	μA

- 注
1. 流入 V_{DD} ， EV_{DD} 和 AV_{REF} 的全部电流，包括当输入引脚固定为 V_{DD} 或 V_{SS} 时的输入漏电流。最大值包括外设操作电流。但是不包括流入 A/D 转换器、可编程增益放大器、比较器、LVI 电路、I/O 端口以及片内上拉/下拉电阻的电流。在 flash 内存执行 HALT 指令期间。
 2. 当内部高速振荡、20 MHz 内部高速振荡以及副系统时钟停止时。
 3. 当 AMPH（时钟操作模式控制寄存器（CMC）的位 0）= 0 时，FLPC 和 FSEL（操作速度模式控制寄存器（OSMC）的位 1 和 0）= 0。
 4. 当高速系统时钟和副系统时钟停止。

- 备注
1. f_{MX} ：高速系统时钟频率（X1 时钟振荡频率或外部主系统时钟频率）
 2. f_{IH20} ：20 MHz 内部高速振荡时钟频率
 3. f_{IH} ：内部高速振荡时钟频率

注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

DC 特性（10/11）

($T_A = -40$ 至 $+85^{\circ}\text{C}$, $1.8\text{ V} \leq V_{DD} = EV_{DD} \leq 5.5\text{ V}$, $1.8\text{ V} \leq AV_{REF} \leq V_{DD}$, $V_{SS} = EV_{SS} = AV_{SS} = 0\text{ V}$)

参数	符号	条件			最小值	典型值	最大值	单位
供应电流	I_{DD2} ^{註1}	HALT 模式	$f_{SUB} = 32.768\text{ kHz}$ ^{註2} , $T_A = -40$ 至 $+70^{\circ}\text{C}$	$V_{DD} = 5.0\text{ V}$		1.0	T.B.D	μA
				$V_{DD} = 3.0\text{ V}$		1.0	T.B.D	μA
				$V_{DD} = 2.0\text{ V}$		1.0	T.B.D	μA
			$f_{SUB} = 32.768\text{ kHz}$ ^{註2} , $T_A = -40$ 至 $+85^{\circ}\text{C}$	$V_{DD} = 5.0\text{ V}$		1.0	T.B.D	μA
				$V_{DD} = 3.0\text{ V}$		1.0	T.B.D	μA
				$V_{DD} = 2.0\text{ V}$		1.0	T.B.D	μA
	I_{DD3} ^{註3}	STOP 模式	$T_A = -40$ 至 $+70^{\circ}\text{C}$			0.33	T.B.D	μA
			$T_A = -40$ 至 $+85^{\circ}\text{C}$			0.33	T.B.D	μA

- 注
1. 流入 V_{DD} , EV_{DD} 和 AV_{REF} 的全部电流, 包括当输入引脚固定为 V_{DD} 或 V_{SS} 时的输入漏电流。最大值包括外设操作电流。但是不包括流入 A/D 转换器、可编程增益放大器、比较器、LVI 电路、I/O 端口以及片内上拉/下拉电阻的电流。在 flash 内存执行 HALT 指令期间。
 2. 当内部高速振荡、20 MHz 内部高速振荡以及高速系统时钟停止时。当看门狗定时器停止时。当实时计数器操作时。
 3. 流入 V_{DD} , EV_{DD} 和 AV_{REF} 的全部电流, 包括当输入引脚固定为 V_{DD} 或 V_{SS} 时的输入漏电流。最大值包括外设操作电流和 STOP 漏电流。但是不包括流入 A/D 转换器、可编程增益放大器、比较器、LVI 电路、I/O 端口以及片内上拉/下拉电阻的电流。当副系统时钟停止时。当看门狗定时器停止时。

备注 f_{SUB} : 副系统时钟频率 (XT1 时钟振荡频率)

注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

DC 特性（11/11）

（ $T_A = -40$ 至 $+85^\circ\text{C}$ ， $1.8\text{ V} \leq V_{DD} = EV_{DD} \leq 5.5\text{ V}$ ， $1.8\text{ V} \leq AV_{REF} \leq V_{DD}$ ， $V_{SS} = EV_{SS} = AV_{SS} = 0\text{ V}$ ）

参数	符号	条件			最小值	典型值	最大值	单位
RTC 工作电流	I _{RTC} 註 1, 2	f _{SUB} = 32.768 kHz		V _{DD} = 3.0 V		0.2	1.0	μA
				V _{DD} = 2.0 V		0.2	1.0	μA
看门狗定时器工作电流	I _{WDT} 註 2, 3	f _{IL} = 30 kHz				0.31	T.B.D	μA
A/D 转换器工作电流	I _{ADC} 註 4	以最大速度转换期间	高速模式 1	AV _{REF} = V _{DD} = 5.0 V		1.72	T.B.D	mA
			高速模式 2	AV _{REF} = V _{DD} = 3.0 V		0.72	T.B.D	mA
			正常模式	AV _{REF} = V _{DD} = 5.0 V		0.86	T.B.D	mA
			电压推动模式	AV _{REF} = V _{DD} = 3.0 V		0.37	T.B.D	mA
可编程增益放大器工作电流	I _{AMP} 註 5					T.B.D	T.B.D	μA
比较器工作电流	I _{CMP} 註 6	不使用内部参考电压时的各通道		AV _{REF} = V _{DD} = 5.0 V		T.B.D	T.B.D	μA
				AV _{REF} = V _{DD} = 3.0 V		T.B.D	T.B.D	μA
		使用内部参考电压时的各通道		AV _{REF} = V _{DD} = 5.0 V		T.B.D	T.B.D	μA
				AV _{REF} = V _{DD} = 3.0 V		T.B.D	T.B.D	μA
LVI 工作电流	I _{LVI} 註 7					9	18	μA

- 注**
1. 仅流入实时计数器的电流（不包括 XT1 振荡器的操作电流）。当实时计数器在操作模式或 HALT 模式中操作时，78K0R/Kx3-L 电流值的典型值是 I_{DD1} 或 I_{DD2} 与 I_{RTC} 典型值的和。 I_{DD1} 和 I_{DD2} 的最大值包括实时计数器操作电流。当实时计数器在 $f_{CLK} = f_{SUB}/2$ 操作时， I_{DD2} 典型值包括实时计数器操作电流。
 2. 当内部高速振荡、20 MHz 内部高速振荡以及高速系统时钟停止时。
 3. 仅流入看门狗定时器的电流（包括 30 kHz 内部振荡器的操作电流）。看门狗定时器在 STOP 模式操作 $f_{CLK} = f_{SUB}/2$ 时，78K0R/Kx3-L 当电流值是 I_{DD1} 、 I_{DD2} 或 I_{DD3} 与 I_{WDT} 的和。
 4. 仅流入 A/D 转换器的电流（ AV_{REF} 引脚）。当 A/D 转换器在操作模式或 HALT 模式工作时，78K0R/Kx3-L 的电流值为 I_{DD1} 或 I_{DD2} 与 I_{ADC} 的和。
 5. 仅流入可编程增益放大器的电流（ AV_{REF} 引脚）。当可编程增益放大器在操作模式或 HALT 模式工作时，78K0R/Kx3-L 的电流值为 I_{DD1} 或 I_{DD2} 与 I_{AMP} 的和。
 6. 仅流入比较器的电流（ AV_{REF} 引脚）。当比较器在操作模式或 HALT 模式工作时，78K0R/Kx3-L 的电流值为 I_{DD1} 或 I_{DD2} 与 I_{CMP} 的和。
 7. 仅流入 LVI 电路的电流（ AV_{REF} 引脚）。当 LVI 电路在操作模式或 HALT 模式工作时，78K0R/Kx3-L 的电流值为 I_{DD1} 、 I_{DD2} 或 I_{DD3} 与 I_{LVI} 的和。

- 备注**
1. f_{IL} : 内部低速振荡时钟频率
 2. f_{SUB} : 副系统时钟频率（XT1 时钟振荡频率）
 3. f_{CLK} : CPU/外设硬件时钟频率

注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

AC 特性

（1）基本操作（1/6）

($T_A = -40$ 至 $+85^{\circ}\text{C}$, $1.8\text{ V} \leq V_{DD} = EV_{DD} \leq 5.5\text{ V}$, $1.8\text{ V} \leq AV_{REF} \leq V_{DD}$, $V_{SS} = EV_{SS} = AV_{SS} = 0\text{ V}$)

参数	符号	条件			最小值	典型值	最大值	单位	
指令周期（最短指令执行时间）	T _{CY}	主系统时钟（f _{MAIN} ）工作	正常电流模式	2.7V ≤ V _{DD} ≤ 5.5V	0.05		8	μs	
				1.8V ≤ V _{DD} < 2.7V	0.2		8	μs	
			低功耗模式			1		8	μs
		副系统时钟（f _{SUB} ）工作				57.2	61	62.5	μs
		自编程模式下	正常电流模式	2.7V ≤ V _{DD} ≤ 5.5V	0.05		0.5	μs	
外部主系统时钟频率	f _{EX}	2.7 V ≤ V _{DD} ≤ 5.5 V			2.0		20.0	MHz	
		1.8 V ≤ V _{DD} < 2.7 V			2.0		5.0	MHz	
外部主系统时钟输入高电平宽度，低电平宽度	t _{EXH} , t _{EXL}	2.7 V ≤ V _{DD} ≤ 5.5 V			24			ns	
		1.8 V ≤ V _{DD} < 2.7 V			96			ns	
TI00, TI02 至 TI07 输入高电平宽度，低电平宽度	t _{TIH} , t _{TIL}				1/f _{MCK} +10			ns	
TO00, TO02 至 TO07 输出频率	f _{TO}	2.7 V ≤ V _{DD} ≤ 5.5 V					10	MHz	
		1.8 V ≤ V _{DD} < 2.7 V					5	MHz	
PCLBUZ0, PCLBUZ1 输出频率	f _{PCL}	2.7 V ≤ V _{DD} ≤ 5.5 V					10	MHz	
		1.8 V ≤ V _{DD} < 2.7 V					5	MHz	
中断输入高电平宽度，低电平宽度	t _{INTH} , t _{INTL}				1			μs	
按键中断输入低电平宽度	t _{KR}				250			ns	
RESET 低电平宽度	t _{RSL}				10			μs	

备注 1. f_{MCK} : 定时器阵列部件的操作时钟频率。

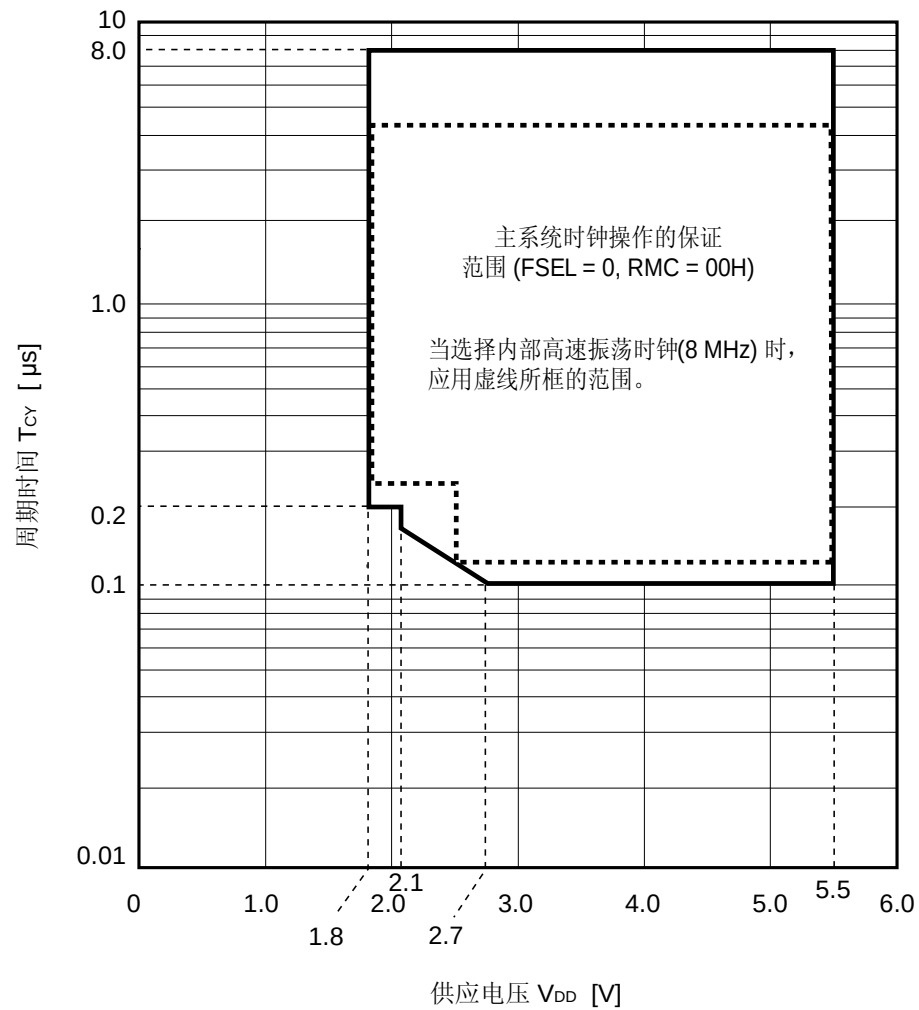
（由 $TMR0n$ 寄存器的 $CKS0n$ 位设置操作时钟。n: 通道编号（n = 0 至 7））

2. 关于根据稳压器输出电压而定的正常电流模式和低功耗模式，参见第二十二章 稳压器。

注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

（1）基本操作（2/6）

在主系统时钟操作期间最短的指令执行时间（FSEL = 0, RMC = 00H）

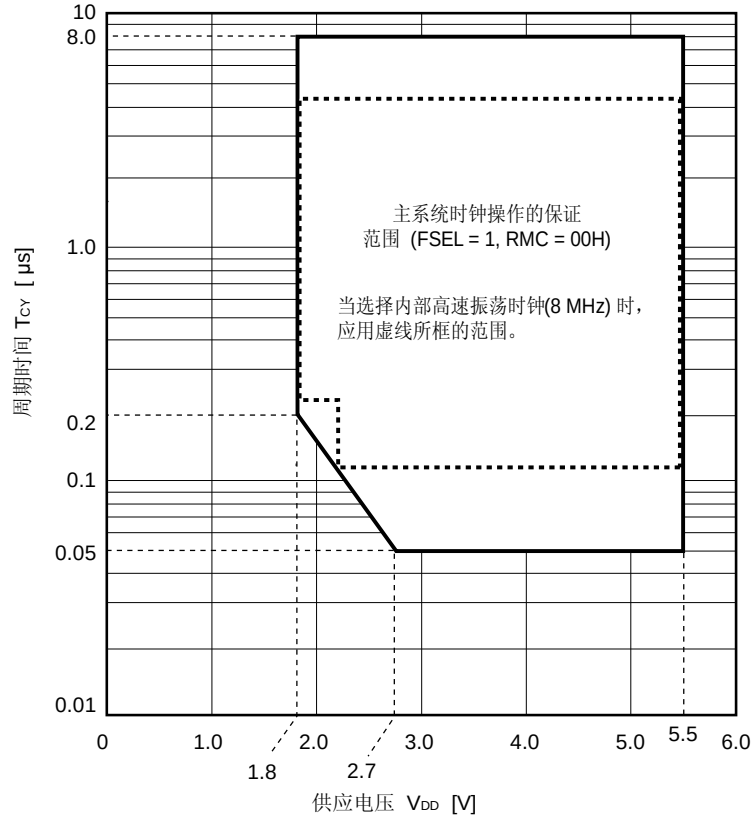


备注 FSEL: 操作速度模式控制寄存器的位 0 (OSMC)

注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

（1）基本操作（3/6）

在主系统时钟操作期间最短的指令执行时间（FSEL = 1, RMC = 00H）



注意事项 当 V_{DD} 低于 2.25 V 时禁止以下操作：

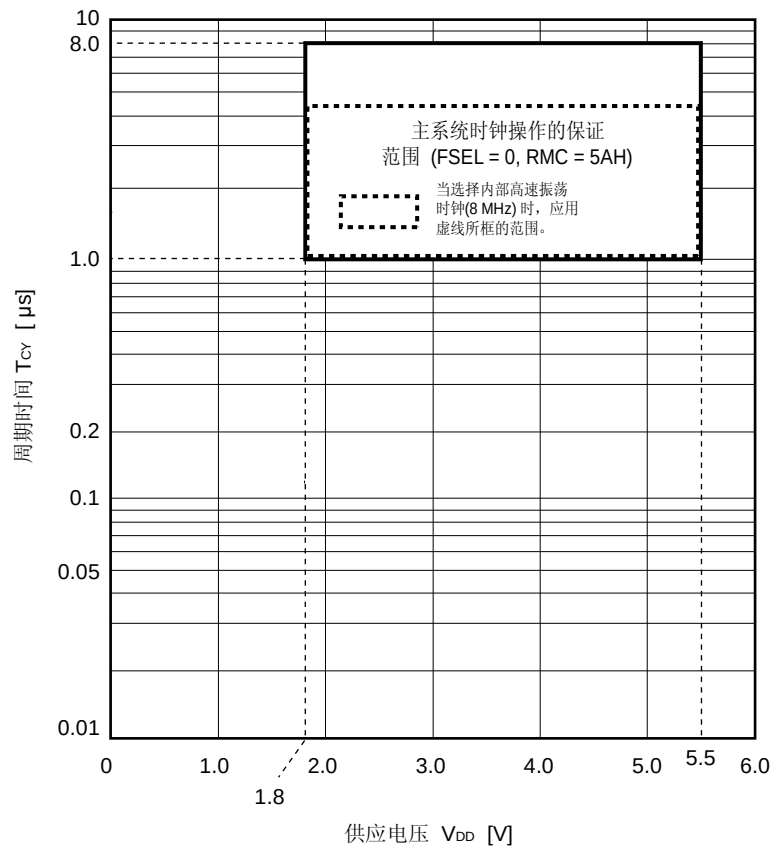
- 对 FSEL 进行从 0 到 1 的重写操作
- 当 FSEL 设置为 1 时，在 f_{EX} 操作和 f_{IH} 操作期间解除 STOP 模式
（即使频率被分频也必须禁止执行该操作。在 f_x 操作期间有可能解除 STOP 模式。）
- 当 FSEL = 1 时，从 f_{SUB} 至 f_{MAIN} 转换的操作
（即使频率被分频也必须禁止执行该操作。）

- 备注
1. FSEL: 操作速度模式控制寄存器的位 0 (OSMC)
 2. f_x : X1 时钟振荡频率
 f_{IH} : 内部高速振荡时钟频率
 f_{EX} : 外部主系统时钟频率
 f_{MAIN} : 主系统时钟频率
 f_{SUB} : 副系统时钟频率
 f_{CLK} : CPU/外设硬件时钟频率

注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

（1）基本操作（4/6）

在主系统时钟操作期间最短的指令执行时间（FSEL = 0, RMC = 5AH）

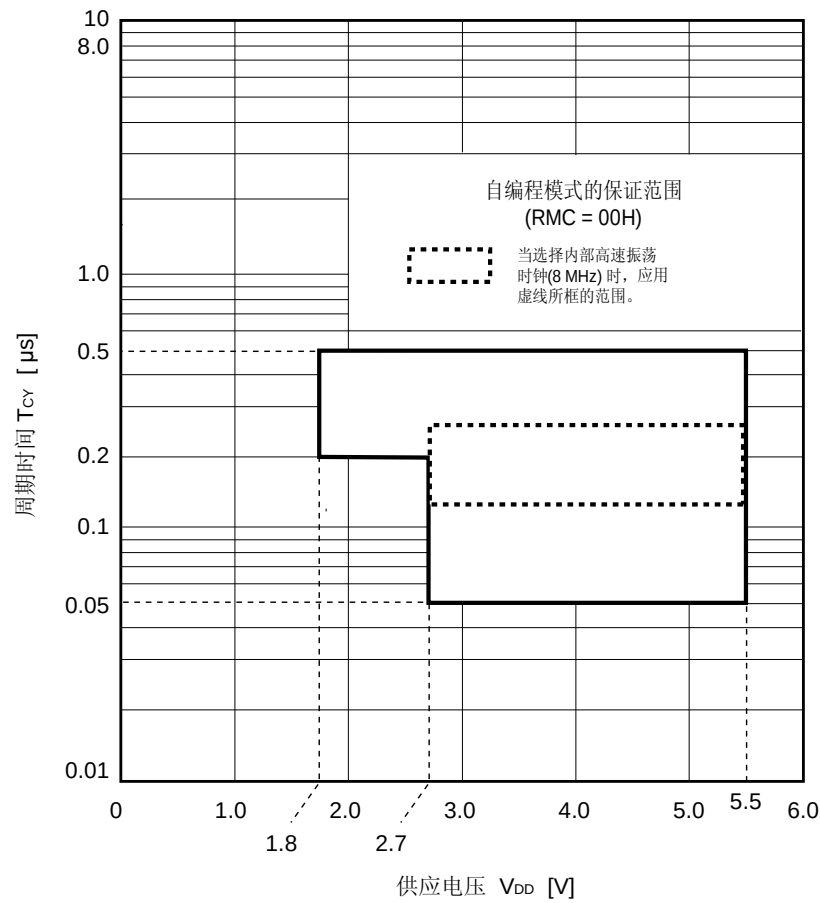


- 备注
1. FSEL: 操作速度模式控制寄存器的位 0 (OSMC)
 2. 当 RMC 设置为 5AH 时, 整个电压范围为 1 MHz (最大)。

注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

(1) 基本操作（5/6）

在自编程模式期间最短的指令执行时间（RMC = 00H）



- 备注
1. FSEL: 操作速度模式控制寄存器的位 0 (OSMC)
 2. 当 RMC 设置为 5AH 或 CPU 工作于副系统时钟时, 禁止使用自编程功能。

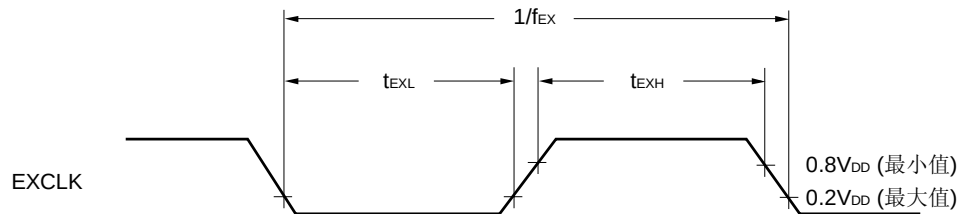
注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

（1）基本操作（6/6）

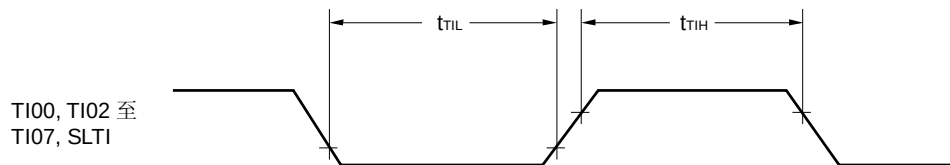
AC 时序检测点



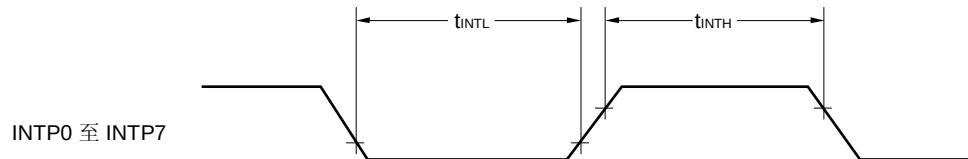
外部主系统时钟时序



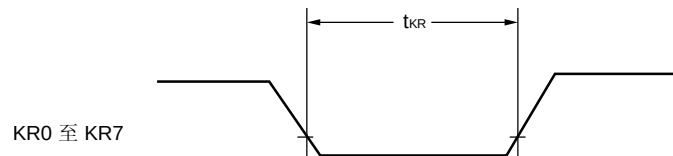
TI 时序



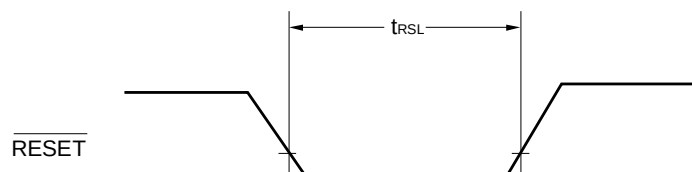
中断请求输入时序



按键中断输入时序



$\overline{RESET}$ 输入时序



注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

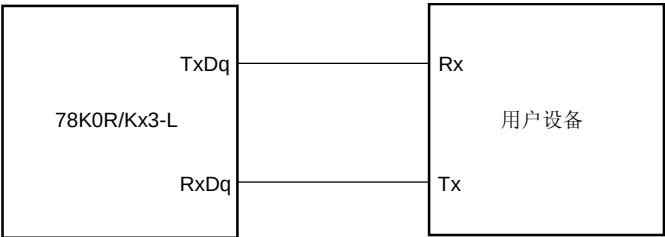
(2) 串行接口：串行阵列部件（1/17）

(TA = -40 至 +85°C， 1.8 V ≤ VDD = EVDD ≤ 5.5 V， VSS = EVSS = AVSS = 0 V)

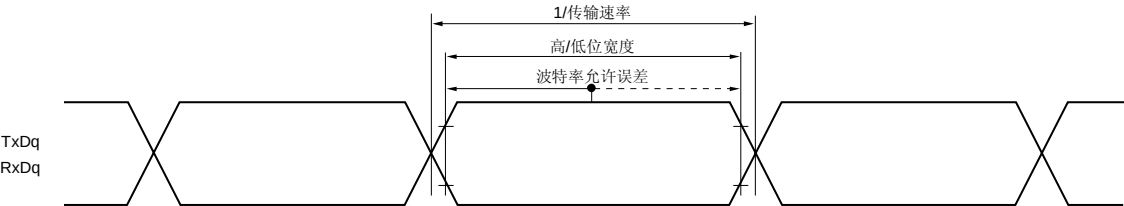
(a) 在相同电位通信期间（UART 模式）（专用波特率发生器输出）

参数	符号	条件	最小值	典型值	最大值	单位
传输速率					fMCK/6	bps
		fCLK = 20 MHz， fMCK = fCLK			3.3	Mbps

UART 模式连接图（在相同电位通信期间）



UART 模式的位宽（在相同电位通信期间）（参考）



注意事项 用 PIMg 和 POMg 寄存器为 RxDq 选择正常输入缓冲器，为 TxDq 选择正常输出模式。

- 备注
- 1. q: UART 的编号（q = 0， 1）， g: PIM 和 POM 的编号（g = 3， 7）
 - 2. fMCK: 串行阵列部件操作时钟频率
（由 SMR0n 寄存器的 CKS0n 位设置操作时钟. n: 通道编号（n = 0 至 3））

注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

(2) 串行接口：串行阵列部件（2/17）

($T_A = -40$ 至 $+85^{\circ}\text{C}$, $1.8\text{ V} \leq V_{DD} = EV_{DD} \leq 5.5\text{ V}$, $V_{SS} = EV_{SS} = AV_{SS} = 0\text{ V}$)

(b) 在相同电位通信期间（CSI 模式）（主机模式，SCKp...内部时钟输出）

参数	符号	条件	最小值	典型值	最大值	单位
$\overline{\text{SCKp}}$ 周期时间	t_{KCY1}	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$	200			ns
		$2.7\text{ V} \leq V_{DD} < 4.0\text{ V}$	400			ns
		$1.8\text{ V} \leq V_{DD} < 2.7\text{ V}$	800			ns
$\overline{\text{SCKp}}$ 高电平/低电平宽度	$t_{\text{KH1}},$ t_{KL1}	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$	$t_{\text{KCY1}}/2 - 20$			ns
		$2.7\text{ V} \leq V_{DD} < 4.0\text{ V}$	$t_{\text{KCY1}}/2 - 35$			ns
		$1.8\text{ V} \leq V_{DD} < 2.7\text{ V}$	$t_{\text{KCY1}}/2 - 80$			ns
Slp 建立时间（到 $\overline{\text{SCKp}}\uparrow$ ） ^{※1}	t_{SIK1}	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$	70			ns
		$2.7\text{ V} \leq V_{DD} < 4.0\text{ V}$	100			ns
		$1.8\text{ V} \leq V_{DD} < 2.7\text{ V}$	190			ns
Slp 保持时间（从 $\overline{\text{SCKp}}\uparrow$ 起） ^{※2}	t_{KSI1}		30			ns
从 $\overline{\text{SCKp}}\downarrow$ 到 Sop 输出间 ^{※3} 的延迟时间	t_{KSO1}	$C = 50\text{ pF}$ ^{※4}			40	ns

- 注
1. 当 $\text{DAP0n} = 0$ 和 $\text{CKP0n} = 0$ 时，或 $\text{DAP0n} = 1$ 和 $\text{CKP0n} = 1$ 时。当 $\text{DAP0n} = 0$ 和 $\text{CKP0n} = 1$ 时，或 $\text{DAP0n} = 1$ 和 $\text{CKP0n} = 0$ 时，Slp 建立时间变成“到 $\overline{\text{SCKp}}\downarrow$ ”。
 2. 当 $\text{DAP0n} = 0$ 和 $\text{CKP0n} = 0$ 时，或 $\text{DAP0n} = 1$ 和 $\text{CKP0n} = 1$ 时。当 $\text{DAP0n} = 0$ 和 $\text{CKP0n} = 1$ 时，或 $\text{DAP0n} = 1$ 和 $\text{CKP0n} = 0$ 时，Slp 保持时间变成“从 $\overline{\text{SCKp}}\downarrow$ 起”。
 3. 当 $\text{DAP0n} = 0$ 和 $\text{CKP0n} = 0$ 时，或 $\text{DAP0n} = 1$ 和 $\text{CKP0n} = 1$ 时。当 $\text{DAP0n} = 0$ 和 $\text{CKP0n} = 1$ 时，或 $\text{DAP0n} = 1$ 和 $\text{CKP0n} = 0$ 时，到 Sop 的延迟时间变成“从 $\overline{\text{SCKp}}\uparrow$ 起”。
 4. C 是 $\overline{\text{SCKp}}$ 和 Sop 输出线的负载电容。

注意事项 用 PIMg 和 POMg 寄存器为 Slp 选择正常输入缓冲器，为 Sop 和 $\overline{\text{SCKp}}$ 选择正常输出模式。

备注 p: CSI 的编号（p = 00, 01, 10），n: 通道的编号（n = 0 至 2），g: PIM 和 POM 的编号（g = 3, 7）

注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

(2) 串行接口：串行阵列部件（3/17）

($T_A = -40$ 至 $+85^\circ\text{C}$, $1.8\text{ V} \leq V_{DD} = EV_{DD} \leq 5.5\text{ V}$, $V_{SS} = EV_{SS} = AV_{SS} = 0\text{ V}$)

(c) 在相同电位通信期间（CSI 模式）（从机模式， $\overline{\text{SCKp}}$...外部时钟输出）

参数	符号	条件	最小值	典型值	最大值	单位
$\overline{\text{SCKp}}$ 周期时间	t_{KCY2}	$16\text{ MHz} < f_{\text{MCK}}$	$8/f_{\text{MCK}}$			ns
		$f_{\text{MCK}} \leq 16\text{ MHz}$	$6/f_{\text{MCK}}$			ns
$\overline{\text{SCKp}}$ 高电平/低电平宽度	t_{KH2} , t_{KL2}		$t_{\text{KCY2}}/2$			ns
Sip 建立时间 (至 $\overline{\text{SCKp}}\uparrow$) ^{※1}	t_{SIK2}		$1/f_{\text{MCK}}+80$			ns
Sip 保持时间 (从 $\overline{\text{SCKp}}\uparrow$ 起) ^{※2}	t_{SI2}		50			ns
从 $\overline{\text{SCKp}}\downarrow$ 至 Sop 输出 ^{※3} 的延迟时间	t_{KSO2}	$C = 50\text{ pF}$ ^{※4}	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$		$1/f_{\text{MCK}}+120$	ns
			$2.7\text{ V} \leq V_{DD} < 4.0\text{ V}$		$1/f_{\text{MCK}}+120$	ns
			$1.8\text{ V} \leq V_{DD} < 2.7\text{ V}$		$1/f_{\text{MCK}}+180$	ns

- 注
1. 当 $\text{DAP0n} = 0$ 和 $\text{CKP0n} = 0$ 时，或 $\text{DAP0n} = 1$ 和 $\text{CKP0n} = 1$ 时。当 $\text{DAP0n} = 0$ 和 $\text{CKP0n} = 1$ 时，或 $\text{DAP0n} = 1$ 和 $\text{CKP0n} = 0$ 时，Sip 建立时间变成“到 $\overline{\text{SCKp}}\downarrow$ ”。
 2. 当 $\text{DAP0n} = 0$ 和 $\text{CKP0n} = 0$ 时，或 $\text{DAP0n} = 1$ 和 $\text{CKP0n} = 1$ 时。当 $\text{DAP0n} = 0$ 和 $\text{CKP0n} = 1$ 时，或 $\text{DAP0n} = 1$ 和 $\text{CKP0n} = 0$ 时，Sip 保持时间变成“从 $\overline{\text{SCKp}}\downarrow$ 起”。
 3. 当 $\text{DAP0n} = 0$ 和 $\text{CKP0n} = 0$ 时，或 $\text{DAP0n} = 1$ 和 $\text{CKP0n} = 1$ 时。当 $\text{DAP0n} = 0$ 和 $\text{CKP0n} = 1$ 时，或 $\text{DAP0n} = 1$ 和 $\text{CKP0n} = 0$ 时，到 Sop 的延迟时间变成“从 $\overline{\text{SCKp}}\uparrow$ 起”。
 4. C 是 $\overline{\text{SCKp}}$ 和 Sop 输出线的负载电容。

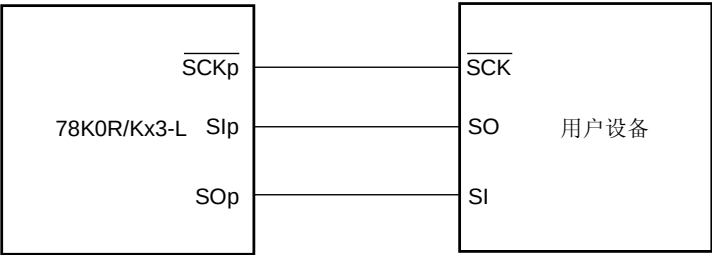
注意事项 用 PIMg 和 POMg 寄存器为 Sip 和 $\overline{\text{SCKp}}$ 选择正常输入缓冲器，为 Sop 选择正常输出模式。

- 备注
1. p: CSI 的编号 (p = 00, 01, 10), g: PIM 和 POM 的编号 (g = 3, 7)
 2. f_{MCK} : 串行阵列部件操作时钟频率
(由 SMR0n 寄存器的 CKS0n 位设置操作时钟. n: 通道编号 (n = 0 至 2))

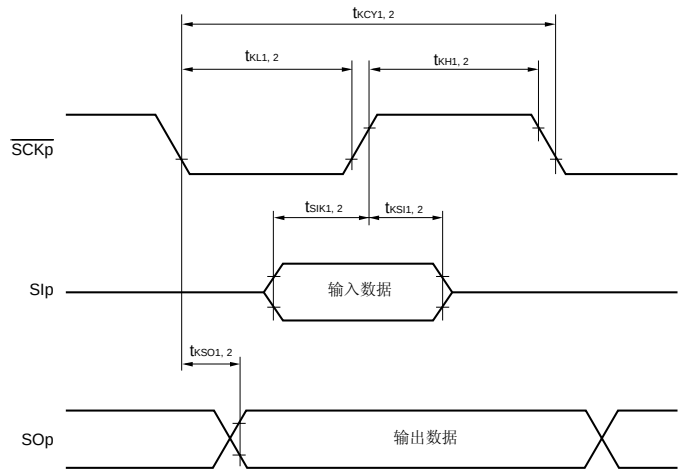
注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

(2) 串行接口：串行阵列部件（4/17）

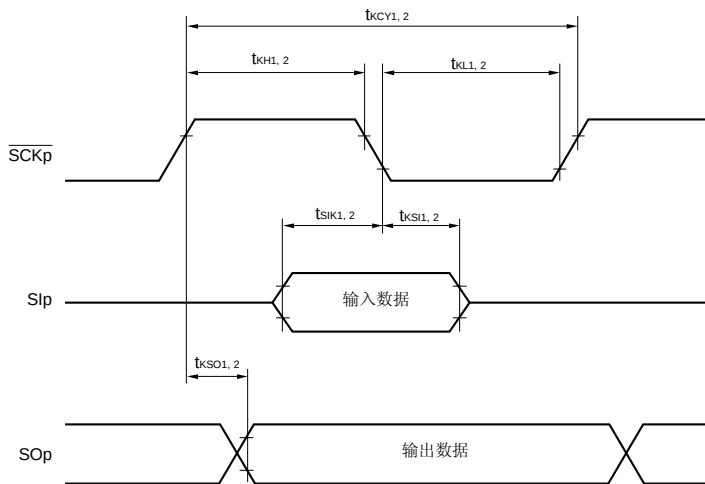
CSI 模式连接图（在相同电位通信期间）



CSI 模式串行传输时序（在相同电位通信期间）
（当 DAP0n = 0 和 CKP0n = 0 时，或 DAP0n = 1 和 CKP0n = 1 时）



CSI 模式串行传输时序（在相同电位通信期间）
（当 DAP0n = 0 和 CKP0n = 1 时，或 DAP0n = 1 和 CKP0n = 0 时）



- 备注
- 1. p: CSI 的编号（p = 00, 01, 10）
 - 2. n: 通道的编号（n = 0 至 2）

注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

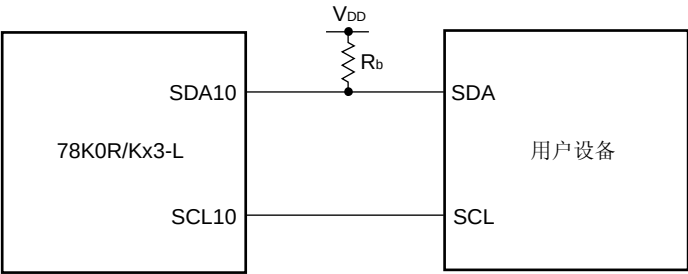
(2) 串行接口：串行阵列部件（5/17）

($T_A = -40$ 至 $+85^{\circ}\text{C}$, $2.7\text{ V} \leq V_{DD} = EV_{DD} \leq 5.5\text{ V}$, $V_{SS} = EV_{SS} = AV_{SS} = 0\text{ V}$)

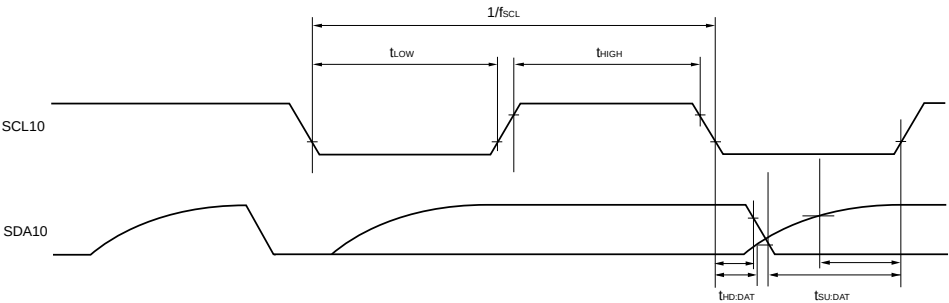
(d) 在相同电位通信期间（简化的 I^2C 模式）

参数	符号	条件	最小值	最大值	单位
SCL10 时钟频率	f_{SCL}	$2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $C_b = 100\text{ pF}$, $R_b = 3\text{ k}\Omega$		400	kHz
SCL10 = "L"时保持时间	t_{LOW}	$2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $C_b = 100\text{ pF}$, $R_b = 3\text{ k}\Omega$	995		ns
SCL10 = "H"时保持时间	t_{HIGH}	$2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $C_b = 100\text{ pF}$, $R_b = 3\text{ k}\Omega$	995		ns
数据建立时间（接收）	$t_{SU: DAT}$	$2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $C_b = 100\text{ pF}$, $R_b = 3\text{ k}\Omega$	$1/f_{MCK}+120$		ns
数据保持时间（发送）	$t_{HD: DAT}$	$2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $C_b = 100\text{ pF}$, $R_b = 3\text{ k}\Omega$	0	160	ns

简易 I^2C 模式连接图（在相同电位通信期间）



简易 I^2C 模式串行传输时序（在相同电位通信期间）



注意事项 用 PIM3 和 POM3 寄存器为 SDA10 选择正常输入缓冲器和 N 沟道漏极开路输出 (V_{DD} 耐压) 模式，为 SCL10 选择正常输出模式。

- 备注
- $R_b[\Omega]$: 通信线 (SDA10) 上拉电阻，
 $C_b[F]$: 通信线 (SCL10, SDA10) 负载电容
 - f_{MCK} : 串行阵列部件操作时钟频率
(由 SMR02 寄存器的 CKS02 位设置的操作时钟。)

注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

(2) 串行接口：串行阵列部件（6/17）

($T_A = -40$ 至 $+85^{\circ}\text{C}$, $2.7\text{ V} \leq V_{DD} = EV_{DD} \leq 5.5\text{ V}$, $V_{SS} = EV_{SS} = AV_{SS} = 0\text{ V}$)

(e) 在不同电位通信（2.5 V, 3 V）（UART 模式）（专用波特率发生器输出）（1/2）

参数	符号	条件		最小值	典型值	最大值	单位
传输速率		接收	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$,			$f_{MCK}/6$	bps
			$2.7\text{ V} \leq V_b \leq 4.0\text{ V}$	$f_{CLK} = 20\text{ MHz}$, $f_{MCK} = f_{CLK}$		3.3	Mbps
			$2.7\text{ V} \leq V_{DD} \leq 4.0\text{ V}$,			$f_{MCK}/6$	bps
			$2.3\text{ V} \leq V_b \leq 2.7\text{ V}$	$f_{CLK} = 20\text{ MHz}$, $f_{MCK} = f_{CLK}$		3.3	Mbps

注意事项 用 PIMg 和 POMg 寄存器为 RxDq 选择 TTL 输入缓冲器，为 TxDq 选择 N 沟道漏极开路输出（ V_{DD} 耐压）模式。

备注

1. q: UART 的编号（q = 0, 1），g: PIM 和 POM 的编号（g = 3, 7）
2. $V_b[V]$: 通信线电压
3. f_{MCK} : 串行阵列部件操作时钟频率
（由 SMR0n 寄存器的 CKS0n 位设置操作时钟. n: 通道编号（n = 0 至 3））
4. UART 模式中当在不同电位通信时，下列 V_{IH} 和 V_{IL} 是串行阵列部件 AC 特性的观测点。
 $4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$: $V_{IH} = 2.2\text{ V}$, $V_{IL} = 0.8\text{ V}$
 $2.7\text{ V} \leq V_{DD} \leq 4.0\text{ V}$, $2.3\text{ V} \leq V_b \leq 2.7\text{ V}$: $V_{IH} = 2.0\text{ V}$, $V_{IL} = 0.5\text{ V}$

注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

(2) 串行接口：串行阵列部件（7/17）

($T_A = -40$ 至 $+85^\circ\text{C}$, $2.7\text{ V} \leq V_{DD} = EV_{DD} \leq 5.5\text{ V}$, $V_{SS} = EV_{SS} = AV_{SS} = 0\text{ V}$)

(e) 在不同电位通信（2.5 V，3 V）（UART 模式）（专用波特率发生器输出）（2/2）

参数	符号	条件		最小值	典型值	最大值	单位
传输速率		发送	4.0 V $\leq$ V _{DD} $\leq$ 5.5 V,			注 1	
			2.7 V $\leq$ V _b $\leq$ 4.0 V	f _{CLK} = 16.8 MHz, f _{MCK} = f _{CLK} , C _b = 50 pF, R _b = 1.4 k Ω , V _b = 2.7 V		2.8 ^{注 2}	Mbps
			2.7 V $\leq$ V _{DD} $\leq$ 4.0 V,			注 3	
			2.3 V $\leq$ V _b $\leq$ 2.7 V	f _{CLK} = 19.2 MHz, f _{MCK} = f _{CLK} , C _b = 50 pF, R _b = 2.7 k Ω , V _b = 2.3 V		1.2 ^{注 4}	Mbps

注 1. 用 f_{MCK}/6 或下式得出的较小的最大传输速率是有效的最大传输速率。

当 4.0 V $\leq$ V_{DD} = EV_{DD} $\leq$ 5.5 V 和 2.7 V $\leq$ V_b $\leq$ 4.0 V 时，计算传输速率的表达式：

$$\text{最大传输速率} = \frac{1}{\left\{ -C_b \times R_b \times \ln \frac{2.2}{V_b} \right\} \times 3} \quad [\text{bps}]$$

$$\text{波特率误差（理论值）} = \frac{\frac{1}{\text{传输速率} \times 2} - \left\{ -C_b \times R_b \times \ln \left(1 - \frac{2.2}{V_b} \right) \right\}}{\left(\frac{1}{\text{传输速率}} \right) \times \text{传输位的编号}} \times 100 [\%]$$

该值为在发送和接收端之间相对差的理论值。

2. 当满足“条件”列的条件时，该值可按举例所示计算得到。在用户条件下参见以上注 1 计算最大传输速率。

3. 用 f_{MCK}/6 或下式得出的较小的最大传输速率是有效的最大传输速率。

当 2.7 V $\leq$ V_{DD} = EV_{DD} $\leq$ 4.0 V 和 2.3 V $\leq$ V_b $\leq$ 2.7 V 时，计算传输速率的表达式：

$$\text{最大传输速率} = \frac{1}{\left\{ -C_b \times R_b \times \ln \frac{2.0}{V_b} \right\} \times 3} \quad [\text{bps}]$$

$$\text{波特率误差（理论值）} = \frac{\frac{1}{\text{传输速率} \times 2} - \left\{ -C_b \times R_b \times \ln \left(1 - \frac{2.0}{V_b} \right) \right\}}{\left(\frac{1}{\text{传输速率}} \right) \times \text{传输位的编号}} \times 100 [\%]$$

该值为在发送和接收端之间相对差的理论值。

4. 当满足“条件”列的条件时，该值可按举例所示计算得到。在用户条件下参见以上注 3 计算最大传输速率。

注意事项 用 PIMg 和 POMg 寄存器为 RxDq 选择 TTL 输入缓冲器，为 TxDq 选择 N 沟道漏极开路输出（V_{DD} 耐压）模式。

（备注在下一页给出。）

注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

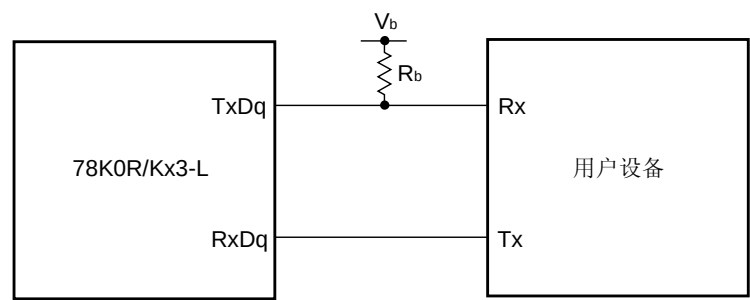
（2）串行接口：串行阵列部件（8/17）

- 备注**
1. $R_b[\Omega]$: 通信线（TxDq）上拉电阻， $C_b[F]$: 通信线（TxDq）负载电容 $V_b[V]$: 通信线电压
 2. q: UART 的编号（q = 0, 1），g: PIM 和 POM 的编号（g = 3, 7）
 3. f_{MCK} : 串行阵列部件操作时钟频率
（由 SMR0n 寄存器的 CKS0n 位设置操作时钟. n: 通道编号（n = 0 至 3））
 4. UART 模式中当在不同电位通信时，下列 V_{IH} 和 V_{IL} 是串行阵列部件 AC 特性的观测点。
 $4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$: $V_{IH} = 2.2\text{ V}$, $V_{IL} = 0.8\text{ V}$
 $2.7\text{ V} \leq V_{DD} \leq 4.0\text{ V}$, $2.3\text{ V} \leq V_b \leq 2.7\text{ V}$: $V_{IH} = 2.0\text{ V}$, $V_{IL} = 0.5\text{ V}$

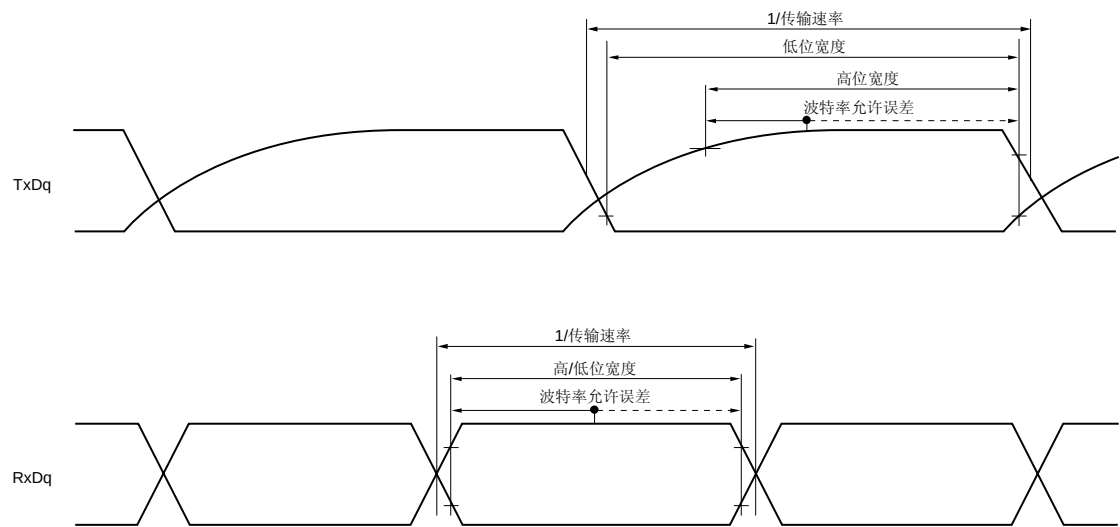
注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

(2) 串行接口：串行阵列部件（9/17）

UART 模式连接图（在不同电位通信期间）



UART 模式的位宽（在不同电位通信期间）（参考）



注意事项 用 PIMg 和 POMg 寄存器为 RxDq 选择 TTL 输入缓冲器，为 TxDq 选择 N 沟道漏极开路输出（V_{DD} 耐压）模式。

- 备注
- 1. R_b[Ω]: 通信线（TxDq）上拉电阻，V_b[V]: 通信线电压
 - 2. q: UART 的编号（q = 0, 1），g: PIM 和 POM 的编号（g = 3, 7）

注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

(2) 串行接口：串行阵列部件（10/17）

($T_A = -40$ 至 $+85^{\circ}\text{C}$, $2.7\text{ V} \leq V_{DD} = EV_{DD} \leq 5.5\text{ V}$, $V_{SS} = EV_{SS} = AV_{SS} = 0\text{ V}$)

(f) 在不同电位通信（2.5 V, 3 V）（CSI 模式）（主机模式， $\overline{\text{SCKp}}$...内部时钟输出）（1/2）

参数	符号	条件	最小值	典型值	最大值	单位
$\overline{\text{SCKp}}$ 周期时间	t_{KCY1}	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$, $C_b = 50\text{ pF}$, $R_b = 1.4\text{ k}\Omega$	500			ns
		$2.7\text{ V} \leq V_{DD} \leq 4.0\text{ V}$, $2.3\text{ V} \leq V_b < 2.7\text{ V}$, $C_b = 50\text{ pF}$, $R_b = 2.7\text{ k}\Omega$	1000			ns
$\overline{\text{SCKp}}$ 高电平宽度	t_{KH1}	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$, $C_b = 50\text{ pF}$, $R_b = 1.4\text{ k}\Omega$	$t_{\text{KCY1}}/2 - 120$			ns
		$2.7\text{ V} \leq V_{DD} \leq 4.0\text{ V}$, $2.3\text{ V} \leq V_b < 2.7\text{ V}$, $C_b = 50\text{ pF}$, $R_b = 2.7\text{ k}\Omega$	$t_{\text{KCY1}}/2 - 275$			ns
$\overline{\text{SCKp}}$ 低电平宽度	t_{KL1}	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$, $C_b = 50\text{ pF}$, $R_b = 1.4\text{ k}\Omega$	$t_{\text{KCY1}}/2 - 20$			ns
		$2.7\text{ V} \leq V_{DD} \leq 4.0\text{ V}$, $2.3\text{ V} \leq V_b < 2.7\text{ V}$, $C_b = 50\text{ pF}$, $R_b = 2.7\text{ k}\Omega$	$t_{\text{KCY1}}/2 - 35$			ns
Slp 建立时间 (至 $\overline{\text{SCKp}}\uparrow$) ^注	t_{SIK1}	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$, $C_b = 50\text{ pF}$, $R_b = 1.4\text{ k}\Omega$	195			ns
		$2.7\text{ V} \leq V_{DD} \leq 4.0\text{ V}$, $2.3\text{ V} \leq V_b < 2.7\text{ V}$, $C_b = 50\text{ pF}$, $R_b = 2.7\text{ k}\Omega$	380			ns
Slp 保持时间 (从 $\overline{\text{SCKp}}\uparrow$ 起) ^注	t_{KSI1}	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$, $C_b = 50\text{ pF}$, $R_b = 1.4\text{ k}\Omega$	30			ns
		$2.7\text{ V} \leq V_{DD} \leq 4.0\text{ V}$, $2.3\text{ V} \leq V_b < 2.7\text{ V}$, $C_b = 50\text{ pF}$, $R_b = 2.7\text{ k}\Omega$	30			ns
从 $\overline{\text{SCKp}}\downarrow$ 至 Sop 输出 ^注 的延迟时间	t_{KSO1}	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$, $C_b = 50\text{ pF}$, $R_b = 1.4\text{ k}\Omega$			165	Ns
		$2.7\text{ V} \leq V_{DD} \leq 4.0\text{ V}$, $2.3\text{ V} \leq V_b < 2.7\text{ V}$, $C_b = 50\text{ pF}$, $R_b = 2.7\text{ k}\Omega$			320	Ns

注 当 DAP0n = 0 和 CKP0n = 0 时，或 DAP0n = 1 和 CKP0n = 1 时。

注意事项 用 PIMg 和 POMg 寄存器为 Slp 选择 TTL 输入缓冲器，为 Sop 和 SCKp 选择 N 沟道漏极开路输出（ V_{DD} 耐压）模式。

备注 1. $R_b[\Omega]$ 通信线（ $\overline{\text{SCKp}}$, Sop）上拉电阻, $C_b[\text{F}]$: 通信线（Slp, Sop, $\overline{\text{SCKp}}$ ）负载电容
 $V_b[\text{V}]$: 通信线电压

2. p: CSI 的编号（p = 00, 01, 10）, n: 通道的编号（n = 0 至 2）, g: PIM 和 POM 的编号（g = 3, 7）

3. CSI 模式中当在不同电位通信时，下列 V_{IH} 和 V_{IL} 是串行阵列部件 AC 特性的观测点。

$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$: $V_{IH} = 2.2\text{ V}$, $V_{IL} = 0.8\text{ V}$

$2.7\text{ V} \leq V_{DD} \leq 4.0\text{ V}$, $2.3\text{ V} \leq V_b \leq 2.7\text{ V}$: $V_{IH} = 2.0\text{ V}$, $V_{IL} = 0.5\text{ V}$

注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

(2) 串行接口：串行阵列部件（11/17）

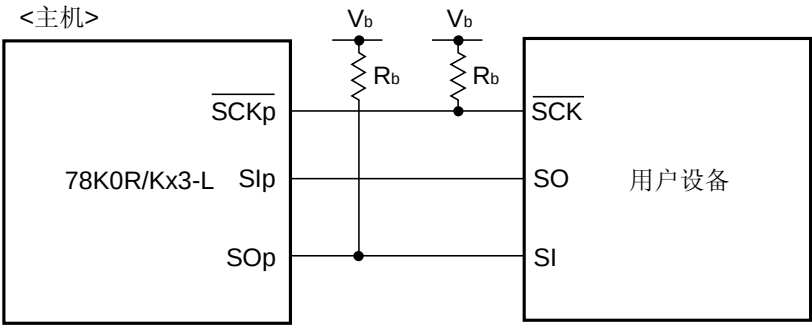
($T_A = -40$ 至 $+85^{\circ}\text{C}$, $2.7\text{ V} \leq V_{DD} = EV_{DD} \leq 5.5\text{ V}$, $V_{SS} = EV_{SS} = AV_{SS} = 0\text{ V}$)

(f) 在不同电位通信 (2.5 V, 3 V) (CSI 模式) (主机模式, $\overline{\text{SCKp}}$...内部时钟输出) (2/2)

参数	符号	条件	最小值	典型值	最大值	单位
Slp 建立时间 (至 $\overline{\text{SCKp}}$ ↓)※	t_{SIK1}	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}, 2.7\text{ V} \leq V_b \leq 4.0\text{ V}$, $C_b = 50\text{ pF}$, $R_b = 1.4\text{ k}\Omega$	70			ns
		$2.7\text{ V} \leq V_{DD} \leq 4.0\text{ V}, 2.3\text{ V} \leq V_b < 2.7\text{ V}$, $C_b = 50\text{ pF}$, $R_b = 2.7\text{ k}\Omega$	100			ns
Slp 保持时间 (从 $\overline{\text{SCKp}}$ ↓起)※	t_{KSI1}	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$, $C_b = 50\text{ pF}$, $R_b = 1.4\text{ k}\Omega$	30			ns
		$2.7\text{ V} \leq V_{DD} \leq 4.0\text{ V}, 2.3\text{ V} \leq V_b < 2.7\text{ V}$, $C_b = 50\text{ pF}$, $R_b = 2.7\text{ k}\Omega$	30			ns
从 $\overline{\text{SCKp}}$ ↑至 SOp 输出※的延迟时间	t_{KSO1}	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}, 2.7\text{ V} \leq V_b \leq 4.0\text{ V}$, $C_b = 50\text{ pF}$, $R_b = 1.4\text{ k}\Omega$			40	ns
		$2.7\text{ V} \leq V_{DD} \leq 4.0\text{ V}, 2.3\text{ V} \leq V_b < 2.7\text{ V}$, $C_b = 50\text{ pF}$, $R_b = 2.7\text{ k}\Omega$			40	ns

注 当 DAP0n = 0 和 CKP0n = 1 时, 或 DAP0n = 1 和 CKP0n = 0 时。

CSI 模式连接图 (在不同电位通信期间)



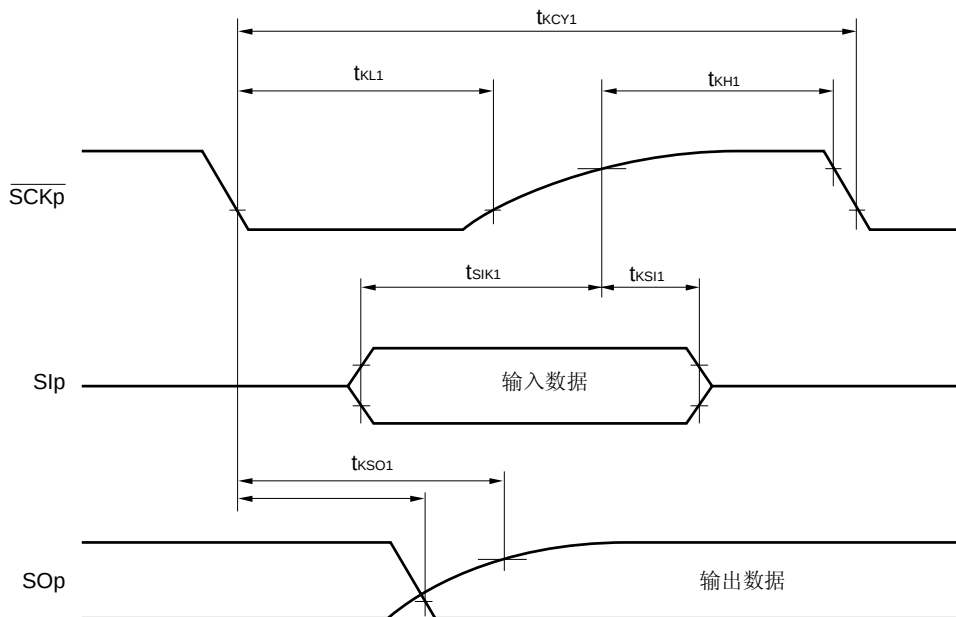
注意事项 用 PIMg 和 POMg 寄存器为 Slp 选择 TTL 输入缓冲器, 为 SOp 和 $\overline{\text{SCKp}}$ 选择 N 沟道漏极开路输出 (V_{DD} 耐压) 模式。

- 备注
- $R_b[\Omega]$ 通信线 ($\overline{\text{SCKp}}$, SOp) 上拉电阻, $C_b[\text{F}]$: 通信线 (Slp, SOp, $\overline{\text{SCKp}}$) 负载电容
 $V_b[\text{V}]$: 通信线电压
 - p: CSI 的编号 (p = 00, 01, 10), n: 通道的编号 (n = 0 至 2), g: PIM 和 POM 的编号 (g = 3, 7)
 - CSI 模式中当在不同电位通信时, 下列 V_{IH} 和 V_{IL} 是串行阵列部件 AC 特性的观测点。
 $4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$: $V_{IH} = 2.2\text{ V}$, $V_{IL} = 0.8\text{ V}$
 $2.7\text{ V} \leq V_{DD} \leq 4.0\text{ V}$, $2.3\text{ V} \leq V_b \leq 2.7\text{ V}$: $V_{IH} = 2.0\text{ V}$, $V_{IL} = 0.5\text{ V}$

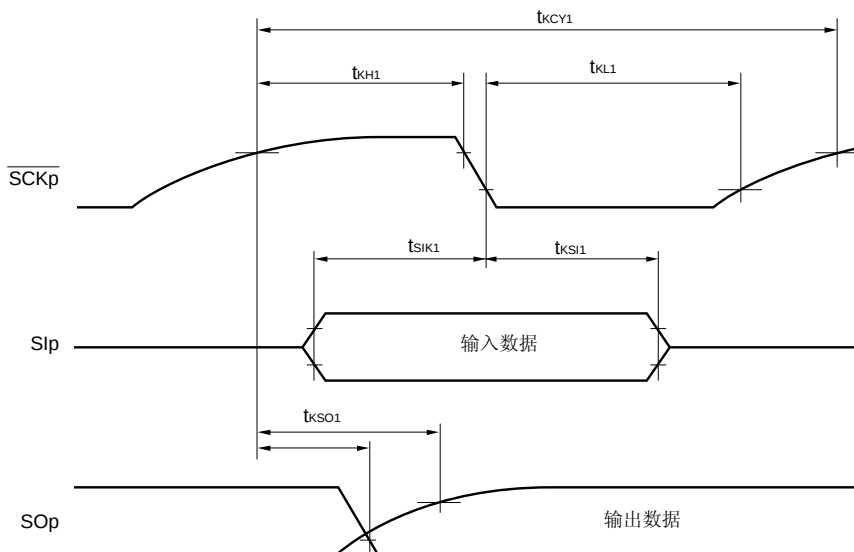
注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

(2) 串行接口：串行阵列部件（12/17）

CSI 模式串行传输时序：主机模式（在不同电位通信期间）
（当 $DAP0n = 0$ 和 $CKP0n = 0$ 时，或 $DAP0n = 1$ 和 $CKP0n = 1$ 时）



CSI 模式串行传输时序：主机模式（在不同电位通信期间）
（当 $DAP0n = 0$ 和 $CKP0n = 1$ 时，或 $DAP0n = 1$ 和 $CKP0n = 0$ 时）



注意事项 用 PIMg 和 POMg 寄存器为 Slp 选择 TTL 输入缓冲器，为 Sop 和 SCKp 选择 N 沟道漏极开路输出（ V_{DD} 耐压）模式。

备注 p: CSI 的编号（ $p = 00, 01, 10$ ），n: 通道的编号（ $n = 0$ 至 2），g: PIM 和 POM 的编号（ $g = 3, 7$ ）

注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

(2) 串行接口：串行阵列部件（13/17）

($T_A = -40$ 至 $+85^\circ\text{C}$, $2.7\text{ V} \leq V_{DD} = EV_{DD} \leq 5.5\text{ V}$, $V_{SS} = EV_{SS} = AV_{SS} = 0\text{ V}$)

(g) 在不同电位通信（2.5 V, 3 V）（CSI 模式）（从机模式， $\overline{\text{SCKp}}$...外部时钟输入）

参数	符号	条件	最小值	典型值	最大值	单位
$\overline{\text{SCKp}}$ 周期时间	t_{KCY2}	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$	$16.6\text{ MHz} < f_{MCK}$	$12/f_{MCK}$		ns
			$12.5\text{ MHz} < f_{MCK} \leq 16.6\text{ MHz}$	$10/f_{MCK}$		ns
			$8.3\text{ MHz} < f_{MCK} \leq 12.5\text{ MHz}$	$8/f_{MCK}$		ns
			$f_{MCK} \leq 8.3\text{ MHz}$	$6/f_{MCK}$		ns
		$2.7\text{ V} \leq V_{DD} < 4.0\text{ V}$, $2.3\text{ V} \leq V_b \leq 2.7\text{ V}$	$17.5\text{ MHz} < f_{MCK}$	$18/f_{MCK}$		ns
			$15\text{ MHz} < f_{MCK} \leq 17.5\text{ MHz}$	$16/f_{MCK}$		ns
			$12.5\text{ MHz} < f_{MCK} \leq 15\text{ MHz}$	$14/f_{MCK}$		ns
			$10\text{ MHz} < f_{MCK} \leq 12.5\text{ MHz}$	$12/f_{MCK}$		ns
			$7.5\text{ MHz} < f_{MCK} \leq 10\text{ MHz}$	$10/f_{MCK}$		ns
			$5\text{ MHz} < f_{MCK} \leq 7.5\text{ MHz}$	$8/f_{MCK}$		ns
			$f_{MCK} \leq 5\text{ MHz}$	$6/f_{MCK}$		ns
$\overline{\text{SCKp}}$ 高电平/低电平宽度	t_{KH2} , t_{KL2}	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$	$f_{KCY2}/2 - 20$			ns
		$2.7\text{ V} \leq V_{DD} < 4.0\text{ V}$, $2.3\text{ V} \leq V_b \leq 2.7\text{ V}$	$f_{KCY2}/2 - 35$			ns
Slp 建立时间 (至 $\overline{\text{SCKp}}\uparrow$) ^{註1}	t_{SIK2}		$1/f_{MCK} + 90$			ns
Slp 保持时间 (从 $\overline{\text{SCKp}}\uparrow$ 起) ^{註2}	t_{KSI2}		50			ns
从 $\overline{\text{SCKp}}\downarrow$ 至 SOp 输出 ^{註3} 的延迟时间	t_{KSO2}	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$, $C_b = 50\text{ pF}$, $R_b = 1.4\text{ k}\Omega$			$1/f_{MCK} + 245$	ns
		$2.7\text{ V} \leq V_{DD} < 4.0\text{ V}$, $2.3\text{ V} \leq V_b \leq 2.7\text{ V}$, $C_b = 50\text{ pF}$, $R_b = 2.7\text{ k}\Omega$			$1/f_{MCK} + 400$	ns

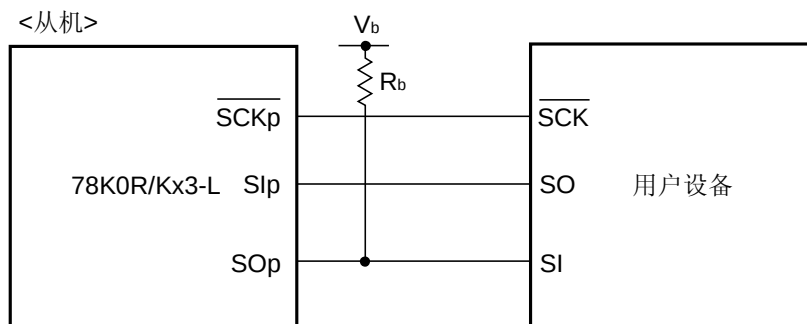
(注， 注意事项和备注在下页给出。)

注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

（2）串行接口：串行阵列部件（14/17）

- 注**
1. 当 $DAP0n = 0$ 和 $CKP0n = 0$ 时，或 $DAP0n = 1$ 和 $CKP0n = 1$ 时。当 $DAP0n = 0$ 和 $CKP0n = 1$ 时，或 $DAP0n = 1$ 和 $CKP0n = 0$ 时， $S\overline{lp}$ 建立时间变成“到 $\overline{SCKp}\downarrow$ ”。
 2. 当 $DAP0n = 0$ 和 $CKP0n = 0$ 时，或 $DAP0n = 1$ 和 $CKP0n = 1$ 时。当 $DAP0n = 0$ 和 $CKP0n = 1$ 时，或 $DAP0n = 1$ 和 $CKP0n = 0$ 时， $S\overline{lp}$ 保持时间变成“从 $\overline{SCKp}\downarrow$ 起”。
 3. 当 $DAP0n = 0$ 和 $CKP0n = 0$ 时，或 $DAP0n = 1$ 和 $CKP0n = 1$ 时。当 $DAP0n = 0$ 和 $CKP0n = 1$ 时，或 $DAP0n = 1$ 和 $CKP0n = 0$ 时，到 Sop 的延迟时间变成“从 $\overline{SCKp}\uparrow$ 起”。

CSI 模式连接图（在不同电位通信期间）



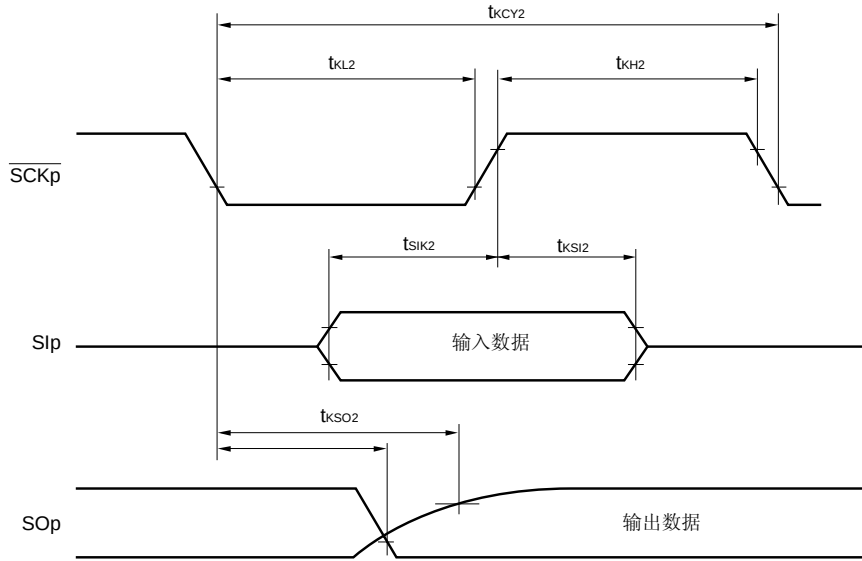
注意事项 用 PIMg 和 POMg 寄存器为 $S\overline{lp}$ 和 $\overline{SCKp}$ 选择 TTL 输入缓冲器，为 Sop 选择 N 沟道漏极开路输出（ V_{DD} 耐压）模式。

- 备注**
1. $R_b[\Omega]$: 通信线（ Sop ）上拉电阻， $C_b[F]$: 通信线（ Sop , $\overline{SCKp}$ ）负载电容
 $V_b[V]$: 通信线电压
 2. p: CSI 的编号（p = 00, 01, 10），g: PIM 和 POM 的编号（g = 3, 7）
 3. f_{MCK} : 串行阵列部件操作时钟频率
（由 $SMR0n$ 寄存器的 $CKS0n$ 位设置操作时钟. n: 通道编号（n = 0 至 2））
 4. CSI 模式中当在不同电位通信时，下列 V_{IH} 和 V_{IL} 是串行阵列部件 AC 特性的观测点。
 $4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$: $V_{IH} = 2.2\text{ V}$, $V_{IL} = 0.8\text{ V}$
 $2.7\text{ V} \leq V_{DD} \leq 4.0\text{ V}$, $2.3\text{ V} \leq V_b \leq 2.7\text{ V}$: $V_{IH} = 2.0\text{ V}$, $V_{IL} = 0.5\text{ V}$

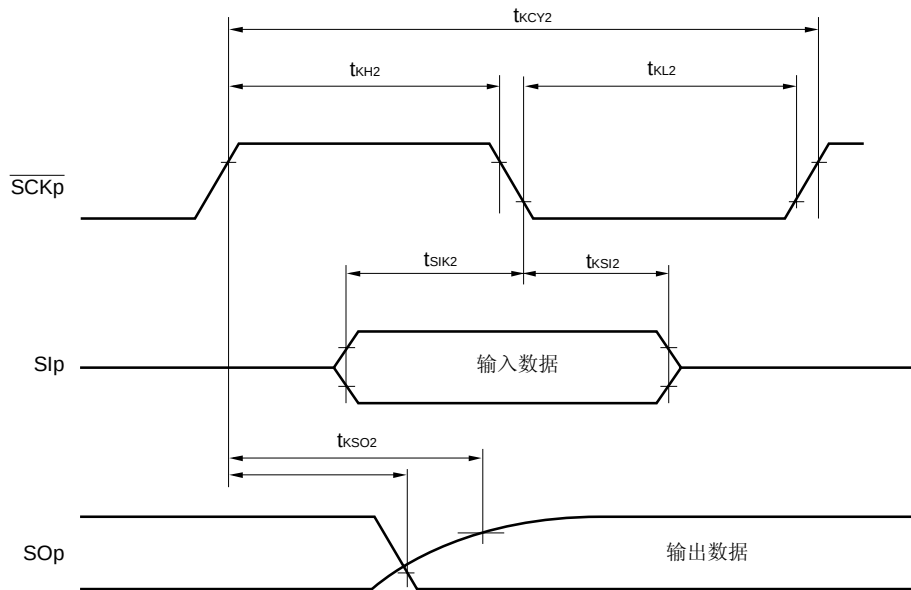
注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

(2) 串行接口：串行阵列部件（15/17）

CSI 模式串行传输时序：从机模式（在不同电位通信期间）
（当 $DAP0n = 0$ 和 $CKP0n = 0$ 时，或 $DAP0n = 1$ 和 $CKP0n = 1$ 时）



CSI 模式串行传输时序：从机模式（在不同电位通信期间）
（当 $DAP0n = 0$ 和 $CKP0n = 1$ 时，或 $DAP0n = 1$ 和 $CKP0n = 0$ 时）



注意事项 用 $PIMg$ 和 $POMg$ 寄存器为 Slp 和 $\overline{SCKp}$ 选择 TTL 输入缓冲器，为 SOp 选择 N 沟道漏极开路输出（ V_{DD} 耐压）模式。

备注 p ：CSI 的编号（ $p = 00, 01, 10$ ）， n ：通道的编号（ $n = 0$ 至 2）， g ：PIM 和 POM 的编号（ $g = 3, 7$ ）

注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

(2) 串行接口：串行阵列部件（16/17）

($T_A = -40$ 至 $+85^{\circ}\text{C}$, $2.7\text{ V} \leq V_{DD} = EV_{DD} \leq 5.5\text{ V}$, $V_{SS} = EV_{SS} = AV_{SS} = 0\text{ V}$)

(h) 在不同电位通信（2.5 V, 3 V）（简化的 I^2C 模式）

参数	符号	条件	最小值	最大值	单位
SCL10 时钟频率	f_{SCL}	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$, $C_b = 100\text{ pF}$, $R_b = 1.4\text{ k}\Omega$		400	kHz
		$2.7\text{ V} \leq V_{DD} \leq 4.0\text{ V}$, $2.3\text{ V} \leq V_b \leq 2.7\text{ V}$, $C_b = 100\text{ pF}$, $R_b = 2.7\text{ k}\Omega$		400	kHz
SCL10 = “L”时保持时间	t_{LOW}	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$, $C_b = 100\text{ pF}$, $R_b = 1.4\text{ k}\Omega$	1065		ns
		$2.7\text{ V} \leq V_{DD} \leq 4.0\text{ V}$, $2.3\text{ V} \leq V_b \leq 2.7\text{ V}$, $C_b = 100\text{ pF}$, $R_b = 2.7\text{ k}\Omega$	1065		ns
SCL10 = “H”时保持时间	t_{HIGH}	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$, $C_b = 100\text{ pF}$, $R_b = 1.4\text{ k}\Omega$	445		ns
		$2.7\text{ V} \leq V_{DD} \leq 4.0\text{ V}$, $2.3\text{ V} \leq V_b \leq 2.7\text{ V}$, $C_b = 100\text{ pF}$, $R_b = 2.7\text{ k}\Omega$	445		ns
数据建立时间（接收）	$t_{SU: DAT}$	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$, $C_b = 100\text{ pF}$, $R_b = 1.4\text{ k}\Omega$	$1/f_{MCK} + 190$		ns
		$2.7\text{ V} \leq V_{DD} \leq 4.0\text{ V}$, $2.3\text{ V} \leq V_b \leq 2.7\text{ V}$, $C_b = 100\text{ pF}$, $R_b = 2.7\text{ k}\Omega$	$1/f_{MCK} + 190$		ns
数据保持时间（发送）	$t_{HD: DAT}$	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$, $C_b = 100\text{ pF}$, $R_b = 1.4\text{ k}\Omega$	0	160	ns
		$2.7\text{ V} \leq V_{DD} \leq 4.0\text{ V}$, $2.3\text{ V} \leq V_b \leq 2.7\text{ V}$, $C_b = 100\text{ pF}$, $R_b = 2.7\text{ k}\Omega$	0	160	ns

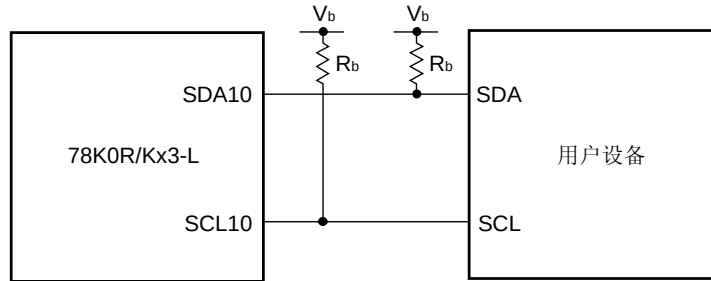
注意事项 用 PIM3 和 POM3 寄存器为 SDA10 选择 TTL 输入缓冲器和 N 沟道漏极开路输出（ V_{DD} 耐压）模式，为 SCL10 选择 N 沟道漏极开路输出（ V_{DD} 耐压）模式。

- 备注
- $R_b[\Omega]$: 通信线（SDA10, SCL10）上拉电阻， $C_b[\text{F}]$: 通信线（SDA10, SCL10）负载电容
 $V_b[\text{V}]$: 通信线电压
 - f_{MCK} : 串行阵列部件操作时钟频率
（由 SMR02 寄存器的 CKS02 位设置的操作时钟。）
 - 简化的 I^2C 模式中当在不同电位通信时，下列 V_{IH} 和 V_{IL} 是串行阵列部件 AC 特性的观测点。
 $4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$: $V_{IH} = 2.2\text{ V}$, $V_{IL} = 0.8\text{ V}$
 $2.7\text{ V} \leq V_{DD} \leq 4.0\text{ V}$, $2.3\text{ V} \leq V_b \leq 2.7\text{ V}$: $V_{IH} = 2.0\text{ V}$, $V_{IL} = 0.5\text{ V}$

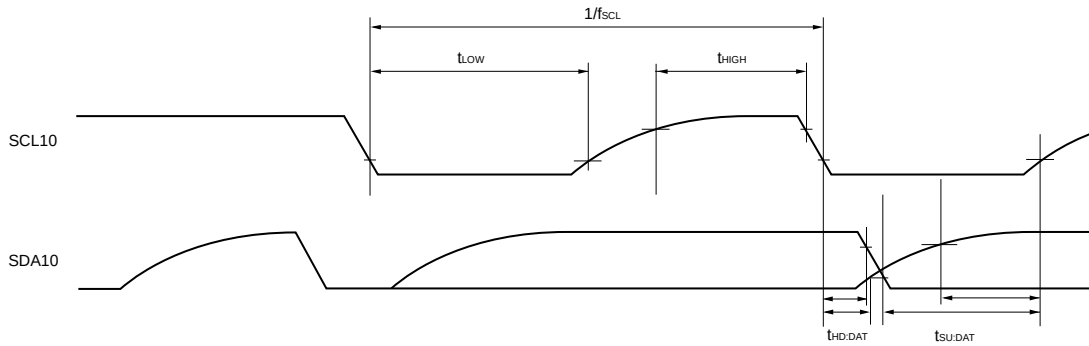
注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

（2）串行接口：串行阵列部件（17/17）

简化的 I^2C 模式连接图（在不同电位通信期间）



简化的 I^2C 模式串行传输时序（在不同电位通信期间）



注意事项 用 PIM3 和 POM3 寄存器为 SDA10 选择 TTL 输入缓冲器和 N 沟道漏极开路输出（V_{DD} 耐压）模式，为 SCL10 选择 N 沟道漏极开路输出（V_{DD} 耐压）模式。

备注 R_b[Ω]通信线（SDA10，SCL10）上拉电阻，V_b[V]：通信线电压

注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

（3）串行接口：IICA

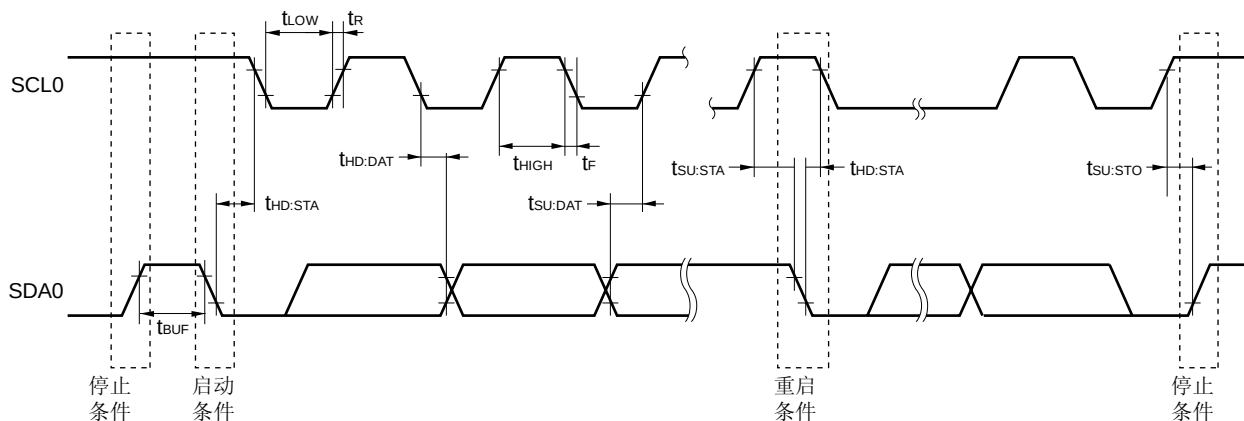
($T_A = -40$ 至 $+85^{\circ}\text{C}$, $1.8\text{ V} \leq V_{DD} = EV_{DD} \leq 5.5\text{ V}$, $V_{SS} = EV_{SS} = AV_{SS} = 0\text{ V}$)

（a）IICA

参数	符号	条件	标准模式		高速模式		单位
			最小值	最大值	最小值	最大值	
SCL0 时钟频率	f_{SCL}		0	100	0	400	kHz
重启条件的建立时间 ^{注 1}	$t_{\text{SU: STA}}$		4.7		0.6		μs
保持时间	$t_{\text{HD: STA}}$		4.0		0.6		μs
SCL0 = "L"时保持时间	t_{LOW}		4.7		1.3		μs
SCL0 = "H"时保持时间	t_{HIGH}		4.0		0.6		μs
数据建立时间（接收）	$t_{\text{SU: DAT}}$		250		100		ns
数据保持时间（发送） ^{注 2}	$t_{\text{HD: DAT}}$		0		0	0.9	μs
停止条件的建立时间	$t_{\text{SU: STO}}$		4.0		0.6		μs
总线空闲时间	t_{BUF}		4.7		1.3		μs
SDA0 和 SCL0 信号的上升时间	t_{R}			1000	$2.0+0.1 C_b$	300	ns
SDA0 和 SCL0 信号的下降时间	t_{F}			300	$2.0+0.1 C_b$	300	ns
各通信线（SCL0, SDA0）的加载电容	C_b			400		400	pF

- 注 1. 检测到启动/重启条件时，该时间段过后产生第一个时钟脉冲。
 2. $t_{\text{HD: DAT}}$ 的最大值（最大值）在正常传输期间，而在 ACK（应答）时序中插入一个等待状态。

IICA 串行传输时序



注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

(4) 串行接口：片上调试（UART）

($T_A = -40$ 至 $+85^{\circ}\text{C}$, $1.8\text{ V} \leq V_{DD} = E_{VDD} \leq 5.5\text{ V}$, $V_{SS} = E_{VSS} = A_{VSS} = 0\text{ V}$)

(a) 片上调试（UART）

参数	符号	条件	最小值	典型值	最大值	单位
传输速率			$f_{CLK}/2^{12}$		$f_{CLK}/6$	bps
		Flash 存储器编程模式			3.33	Mbps
TOOL1 输出频率	f_{TOOL1}	$2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}$			10	MHz
		$1.8\text{ V} \leq V_{DD} < 2.7\text{ V}$			2.5	MHz

注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

A/D 转换器特性

($T_A = -40$ 至 $+85^{\circ}\text{C}$, $1.8\text{ V} \leq V_{DD} = EV_{DD} \leq 5.5\text{ V}$, $1.8\text{ V} \leq AV_{REF} \leq V_{DD}$, $V_{SS} = EV_{SS} = AV_{SS} = 0\text{ V}$)

参数	符号	条件	最小值	典型值	最大值	单位
分辨率	RES				10	位
总误差 ^{注1, 2}	AINL	$4.0\text{ V} \leq AV_{REF} \leq 5.5\text{ V}$			± 0.4	%FSR
		$2.7\text{ V} \leq AV_{REF} < 4.0\text{ V}$			± 0.6	%FSR
		$1.8\text{ V} \leq AV_{REF} < 2.7\text{ V}$			± 1.2	%FSR
转换时间	t_{CONV}	$4.0\text{ V} \leq AV_{REF} \leq 5.5\text{ V}$	高速模式 1	2.5	66.6	μs
			正常模式	6.1	66.6	μs
		$2.7\text{ V} \leq AV_{REF} < 4.0\text{ V}$	高速模式 1	4.5	66.6	μs
			正常模式	12.2	66.6	μs
		$1.8\text{ V} \leq AV_{REF} < 2.7\text{ V}$	电压推动模式	27	66.6	μs
零标称误差 ^{注1, 2}	EVS	$4.0\text{ V} \leq AV_{REF} \leq 5.5\text{ V}$	正常模式		± 0.4	%FSR
		$2.7\text{ V} \leq AV_{REF} < 4.0\text{ V}$	正常模式		± 0.6	%FSR
满量程误差 ^{注1, 2}	EFS	$4.0\text{ V} \leq AV_{REF} \leq 5.5\text{ V}$	正常模式		± 0.4	%FSR
		$2.7\text{ V} \leq AV_{REF} < 4.0\text{ V}$	正常模式		± 0.6	%FSR
积分非线性误差 ^{注1}	ILE	$4.0\text{ V} \leq AV_{REF} \leq 5.5\text{ V}$			± 2.5	LSB
		$2.7\text{ V} \leq AV_{REF} < 4.0\text{ V}$			± 4.5	LSB
		$1.8\text{ V} \leq AV_{REF} < 2.7\text{ V}$			± 6.5	LSB
微分非线性误差 ^{注1}	DLE	$4.0\text{ V} \leq AV_{REF} \leq 5.5\text{ V}$			± 1.5	LSB
		$2.7\text{ V} \leq AV_{REF} < 4.0\text{ V}$			± 2.0	LSB
		$1.8\text{ V} \leq AV_{REF} < 2.7\text{ V}$			± 2.0	LSB
模拟输入电压	V_{AIN}	$1.8\text{ V} \leq AV_{REF} \leq 5.5\text{ V}$	AV_{SS}		AV_{REF}	V

- 注
1. 量化误差除外 ($\pm 1/2$ LSB)
 2. 该值表示满量程值的比率 (%FSR)。

注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

可编程增益放大器特性

($T_A = -40$ 至 $+85^{\circ}\text{C}$, $2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $2.7\text{ V} \leq AV_{REF} \leq V_{DD}$, $V_{SS} = AV_{SS} = 0\text{ V}$)

参数	符号	条件	最小值	典型值	最大值	单位
输入偏置电压	V_{IOAMP}			T.B.D		mV
输入电压范围	V_{IAMP}	×1 增益	$0.1AV_{REF}$		$0.45AV_{REF}$	V
		其它	$0.1AV_{REF}$ /增益		$0.9AV_{REF}$ /增益	V
最大输出电压	V_{OAMP}		$0.1AV_{REF}$		$0.9AV_{REF}$	V
转换率	SR_F	上升沿		T.B.D		V/ μs
	SR_R	下降沿		T.B.D		V/ μs
增益	RG		1 至 12			次
操作稳定等待时间	t_{AMP}				3	μs

备注

转换率：对于输出电压上升或下降的变化。

$V/\mu\text{s}$: 每 $1\ \mu\text{s}$ 电压中的变化

操作稳定等待时间：

在可编程增益放大器的操作使能后（OAM 寄存器的 $OAEN = 1$ ），直到进入满足可编程增益放大器的 DC 和 AC 特性的状态所需要的时间。

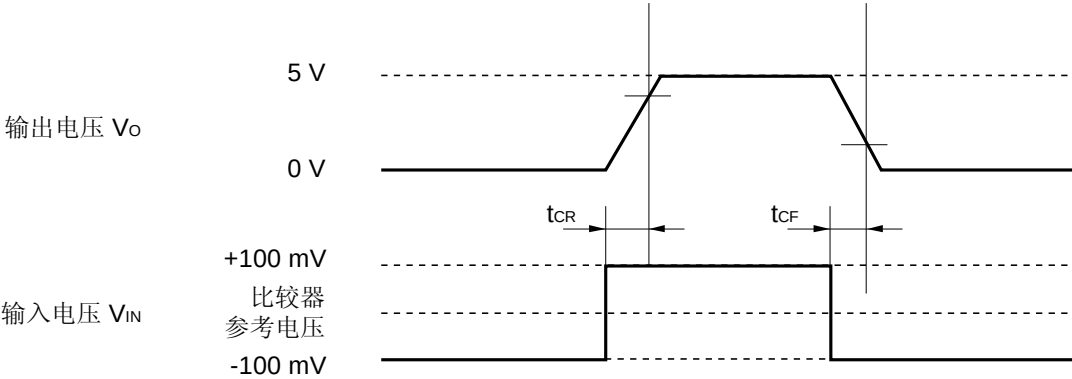
注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

比较器特性

($T_A = -40$ 至 $+85^{\circ}\text{C}$, $2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $2.7\text{ V} \leq AV_{REF} \leq V_{DD}$, $V_{SS} = AV_{SS} = 0\text{ V}$)

参数	符号	条件	最小值	典型值	最大值	单位
输入偏置电压	$V_{IO_{CMP}}$			T.B.D		mV
输入电压范围	$V_{I_{CMP}}$		$0.1AV_{REF}$		$0.9AV_{REF}$	V
内部参考电压偏差	V_{IREF}			T.B.D		%
响应时间	t_{CR}	输入振幅 = $\pm 100\text{ mV}$, 在上升沿处 ^{註 1}		T.B.D		ns
	t_{CF}	输入振幅 = $\pm 100\text{ mV}$, 在下降沿处 ^{註 2}		T.B.D		ns
操作稳定等待时间	t_{CMP}				1	μs
参考电压稳定等待时间	t_{VR}				1	μs

- 注
1. 当 CMP0P 输入或可编程增益放大器输出从比较器参考电压-100 mV 变化到比较器参考电压+100 mV 时，脉冲响应的特性。
 2. 当 CMP0P 输入或可编程增益放大器输出从比较器参考电压+100 mV 变化到比较器参考电压-100 mV 时，脉冲响应的特性。



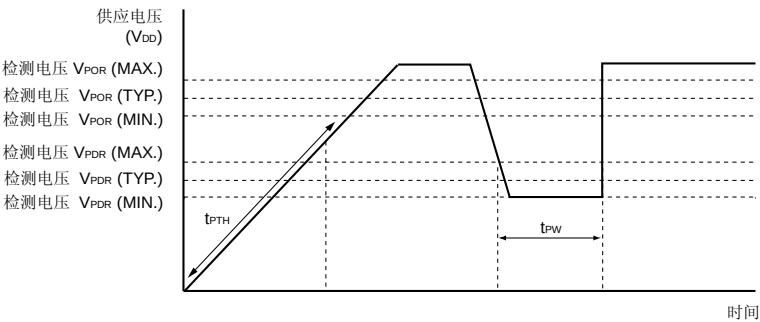
- 备注
- 操作稳定等待时间： 在比较器操作使能后（CnCTL 寄存器的 CnEN = 1）（n = 0, 1），直到进入满足比较器的 DC 和 AC 特性的状态所需要的时间。
- 参考电压稳定等待时间： 在内部参考电压使能（CnRVM 寄存器的 CnVRE = 1）（n = 0, 1）后，等到内部参考电压电路的电压电平达到理想值的 99% 时所需要的时间。

注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

POC 电路特性（TA = -40 至 +85℃，VSS = 0 V）

参数	符号	条件	最小值	典型值	最大值	单位
检测电压	V _{POR}	电源供应上升时间	1.52	1.61	1.70	V
	V _{PDR}	电源供应下降时间	1.5	1.59	1.68	V
电源供应电压上升斜坡	t _{PTH}	改变 V _{DD} 的斜坡：0 V → V _{POR}	0.5			V/ms
最小脉冲宽度	t _{PW}	当电压下降时	200			μs
检测延迟时间					200	μs

POC 电路时序



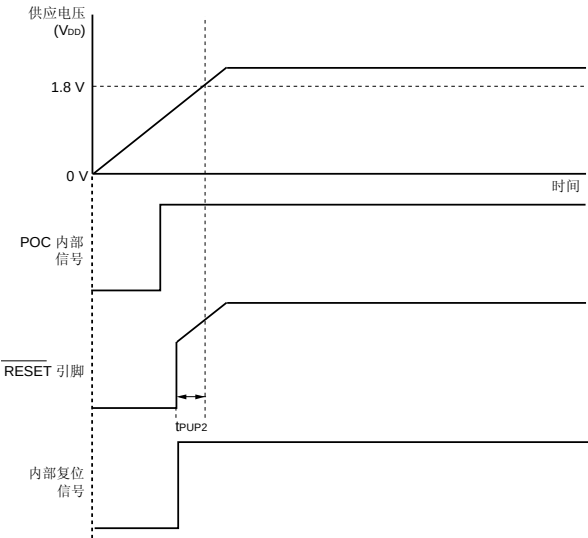
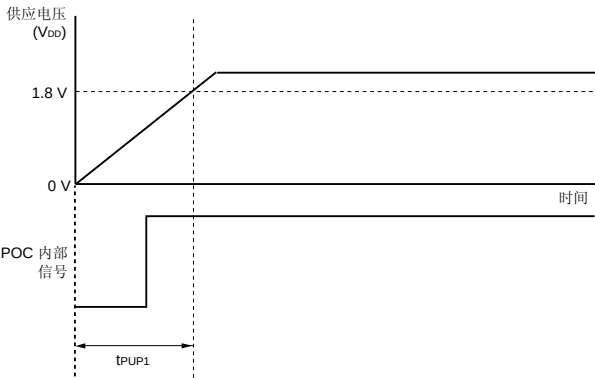
供应电压上升时间（TA = -40 至 +85℃，VSS = 0 V）

参数	符号	条件	最小值	典型值	最大值	单位
上升到 1.8 V 的最大时间（V _{DD} （最小值）） ^注 （V _{DD} ：0 V → 1.8 V）	t _{PUP1}	当不使用 $\overline{\text{RESET}}$ 输入时，LVI 默认启动功能停止被设置（LVIOFF（选项字节）= 1）			3.6	ms
上升到 1.8 V 的最大时间（V _{DD} （最小值）） ^注 （解除 $\overline{\text{RESET}}$ 输入 → V _{DD} ：1.8 V）	t _{PUP2}	当使用 $\overline{\text{RESET}}$ 输入时，LVI 默认启动功能停止被设置（LVIOFF（选项字节）= 1）			1.88	ms

注 确保在比这短的时间内上升供应电源。

供应电压上升时间时序

- 当不使用 $\overline{\text{RESET}}$ 引脚时
- 当使用 $\overline{\text{RESET}}$ 引脚时（在 POC 被解除后，当 $\overline{\text{RESET}}$ 引脚解除内部复位时）



注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

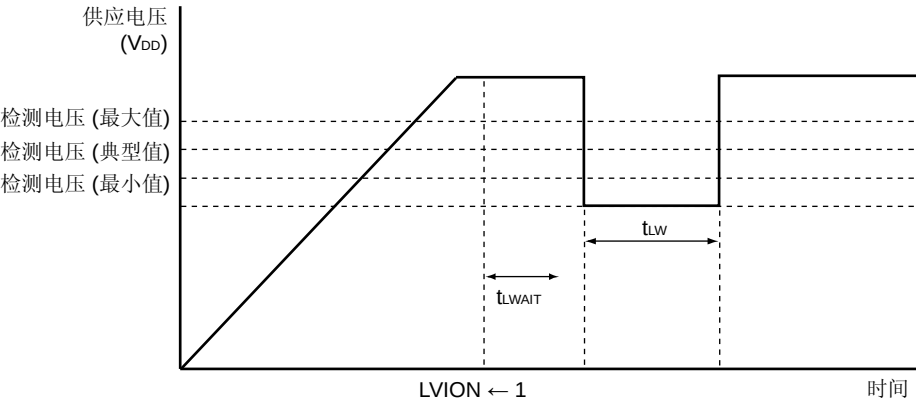
LVI 电路特性（ $T_A = -40$ 至 $+85^{\circ}\text{C}$ ， $V_{PDR} \leq V_{DD} = EV_{DD} \leq 5.5\text{ V}$ ， $V_{SS} = EV_{SS} = 0\text{ V}$ ）

参数		符号	条件	最小值	典型值	最大值	单位
检测电压	供应电压电平	V_{LVI0}		4.12	4.22	4.32	V
		V_{LVI1}		3.97	4.07	4.17	V
		V_{LVI2}		3.82	3.92	4.02	V
		V_{LVI3}		3.66	3.76	3.86	V
		V_{LVI4}		3.51	3.61	3.71	V
		V_{LVI5}		3.35	3.45	3.55	V
		V_{LVI6}		3.20	3.30	3.40	V
		V_{LVI7}		3.05	3.15	3.25	V
		V_{LVI8}		2.89	2.99	3.09	V
		V_{LVI9}		2.74	2.84	2.94	V
		V_{LVI10}		2.58	2.68	2.78	V
		V_{LVI11}		2.43	2.53	2.63	V
		V_{LVI12}		2.28	2.38	2.48	V
		V_{LVI13}		2.12	2.22	2.32	V
		V_{LVI14}		1.97	2.07	2.17	V
		V_{LVI15}		1.81	1.91	2.01	V
	外部输入引脚 ^{注 1}	V_{EXLVI}	$EXLVI < V_{DD}$ ， $1.8\text{ V} \leq V_{DD} \leq 5.5\text{ V}$	1.11	1.21	1.31	V
加电时电源供应电压		V_{PUPLVI}	当设置 LVI 默认启动功能使能时	1.87	2.07	2.27	V
最小脉冲宽度		t_{LW}		200			μs
检测延迟时间						200	μs
操作稳定等待时间 ^{注 2}		t_{LWAIT}				10	μs

注 1. 使用 EXLVI/P120/INTP0 引脚。
2. 从设置低压检测寄存器（LVIM）的位 7（LVION）为 1 到操作稳定所需要的时间

备注 $V_{LVI(n-1)} > V_{LVIn}$ ： $n = 1$ 至 15

LVI 电路时序

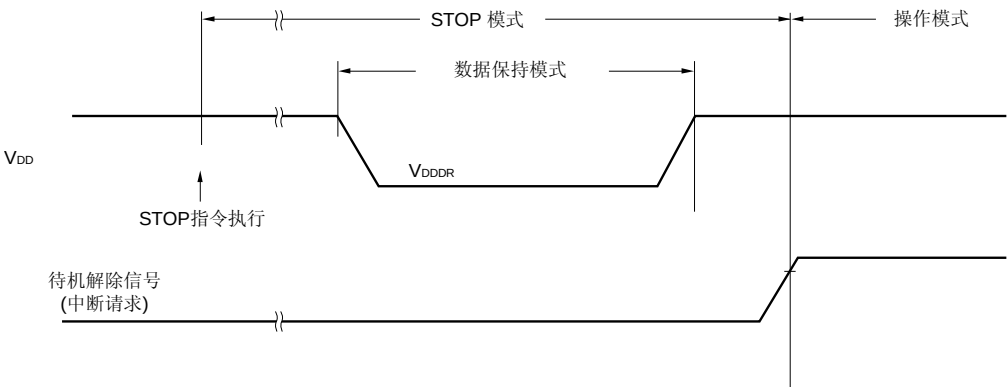


注意事项 根据产品安装这些引脚。参考本节开始处的注意事项 3。

数据存储 STOP 模式低供应电压数据保持特性（ $T_A = -40$ 至 $+85^{\circ}\text{C}$ ）

参数	符号	条件	最小值	典型值	最大值	单位
数据保持供应电压	V_{DDDR}		1.5 ^注		5.5	V

注 该值取决于 POC 检测电压。当电压下降时，数据保持直到 POC 复位有效为止，但当 POC 复位有效时数据不再被保持。



Flash 存储器编程特性

($T_A = -40$ 至 $+85^{\circ}\text{C}$, $1.8\text{ V} \leq V_{DD} = EV_{DD} \leq 5.5\text{ V}$, $V_{SS} = EV_{SS} = 0\text{ V}$)

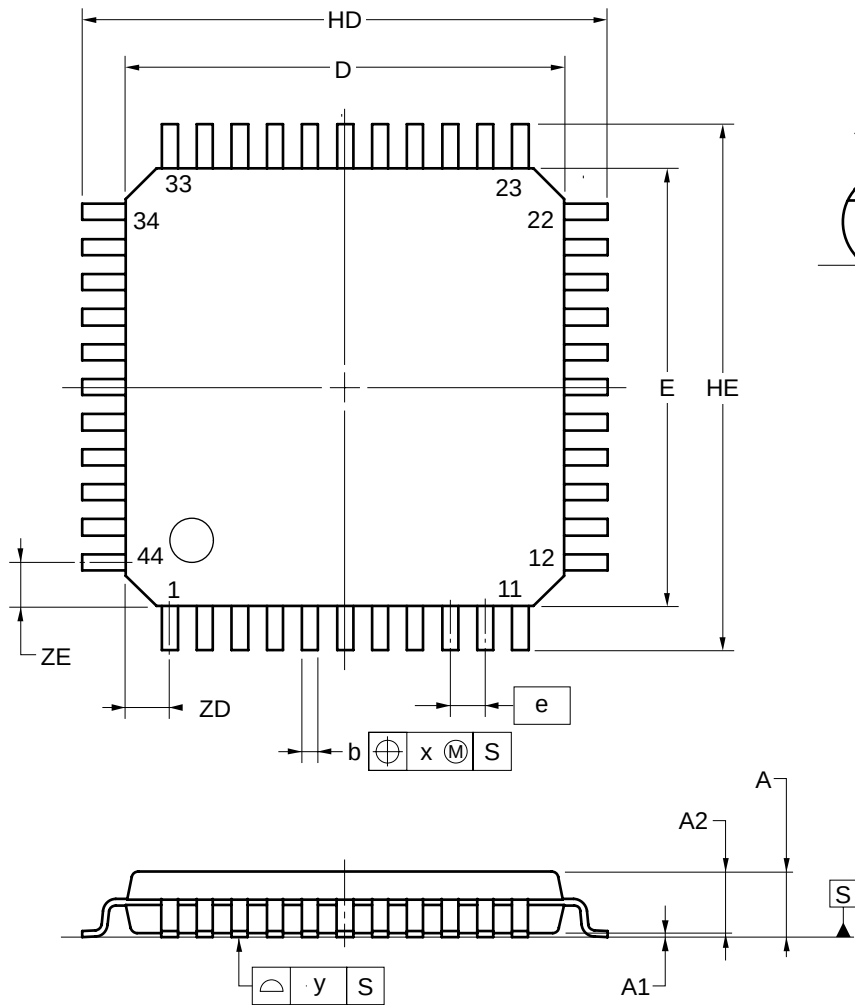
参数	符号	条件	最小值	典型值	最大值	单位
V_{DD} 供应电流	I_{DD}	Typ. = 10 MHz, Max. = 20 MHz		6	20	mA
CPU/外设硬件时钟频率	f_{CLK}	$2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}$	2		20	MHz
各芯片重写次数	C_{erwr}	保留: 15 年 1 次擦除 + 1 次擦除后写入 = 1 次重写 ^注	100			次

注 出厂后首次对产品进行写入时，“擦除→写入”和“单写入”都被当作是一次重写。

29.1 78K0R/KC3-L（44-引脚产品）

μ PD78F1000GB-GAF-AX, 78F1001GB-GAF-AX, 78F1002GB-GAF-AX, 78F1003GB-GAF-AX

44-引脚 LQFP 塑封 (10x10)



引线末端的细节

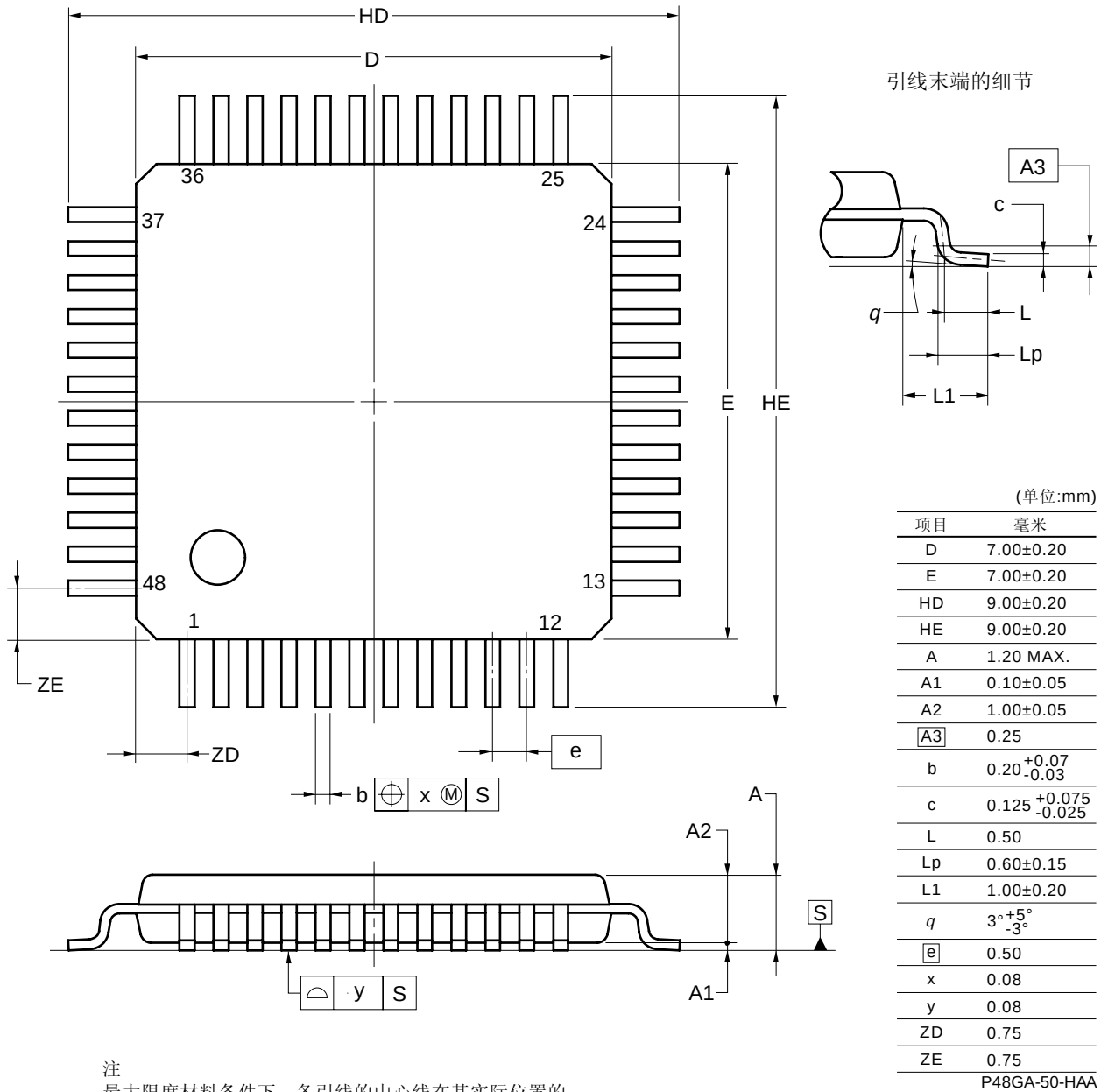
(单位:mm)	
项目	毫米
D	10.00±0.20
E	10.00±0.20
HD	12.00±0.20
HE	12.00±0.20
A	1.60 MAX.
A1	0.10±0.05
A2	1.40±0.05
A3	0.25
b	0.35 ^{+0.08} _{-0.04}
c	0.125 ^{+0.075} _{-0.025}
L	0.50
Lp	0.60±0.15
L1	1.00±0.20
q	3° ^{+5°} _{-3°}
e	0.80
x	0.20
y	0.10
ZD	1.00
ZE	1.00
P44GB-80-GAF	

注
最大限度材料条件下，各引线的中心线在其实际位置的
0.20 mm 内。

29.2 78K0R/KC3-L (48-引脚产品)

μ PD78F1001GA-HAA-AX, 78F1002GA-HAA-AX, 78F1003GA-HAA-AX

48-引脚 TQFP 塑封 (密间距) (7x7)



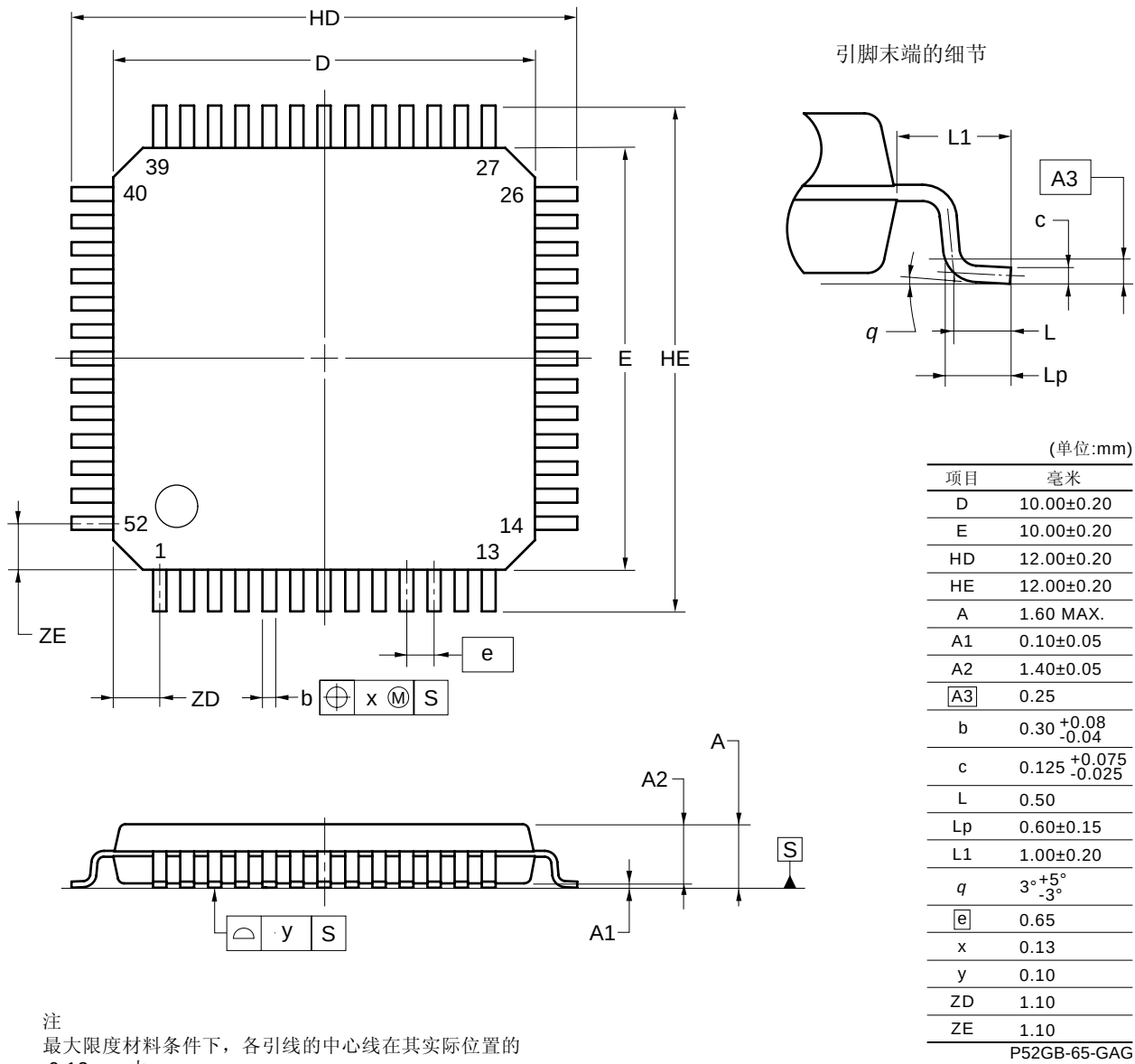
P48GA-50-HAA

© NEC Electronics Corporation 2005

29.3 78K0R/KD3-L

μ PD78F1004GB-GAG-AX, 78F1005GB-GAG-AX, 78F1006GB-GAG-AX

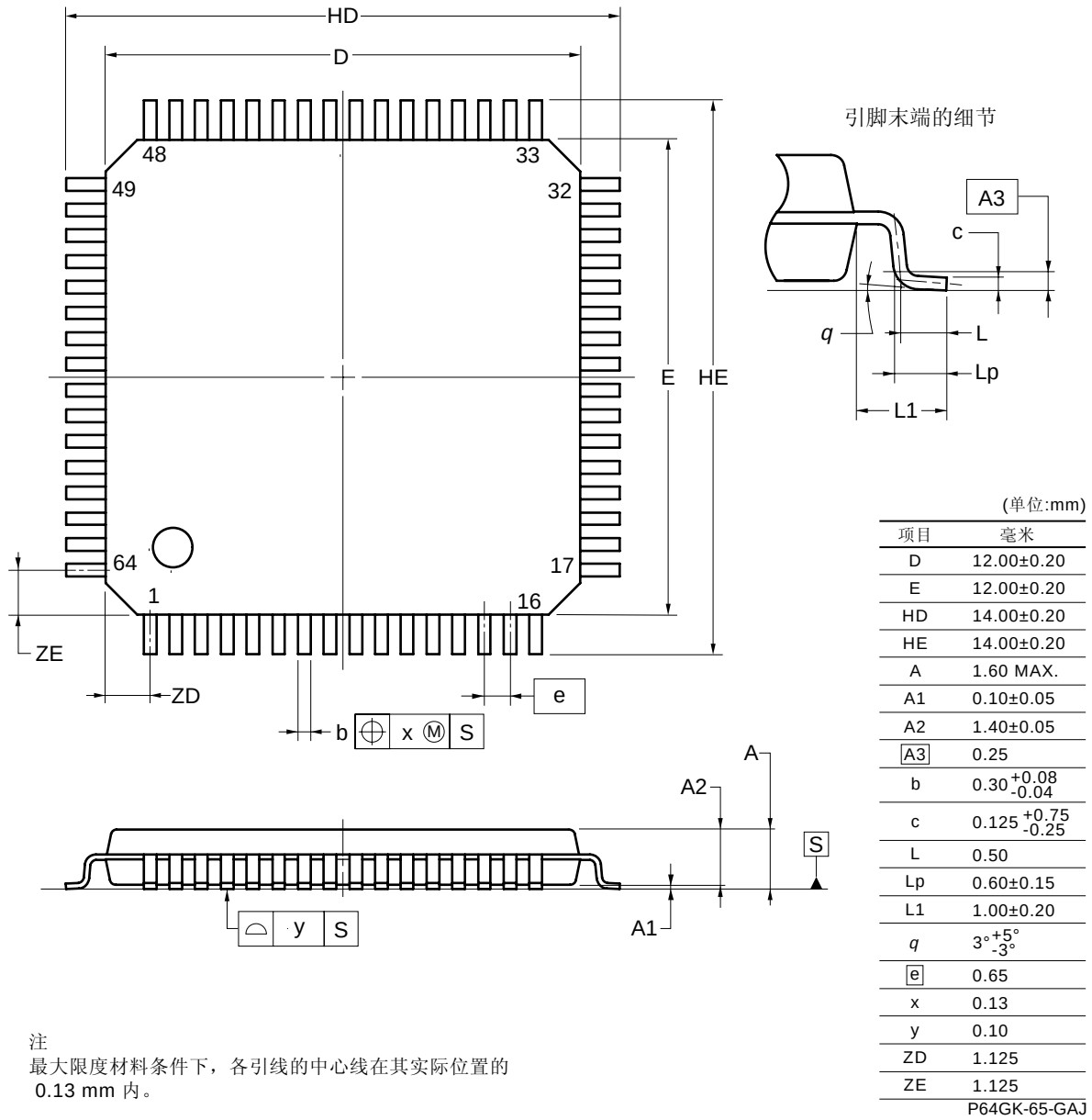
52-引脚 LQFP 塑封 (10x10)



29.4 78K0R/KE3-L

μ PD78F1007GK-GAJ-AX, 78F1008GK-GAJ-AX, 78F1009GK-GAJ-AX

64-引脚 LQFP 塑封 (12x12)

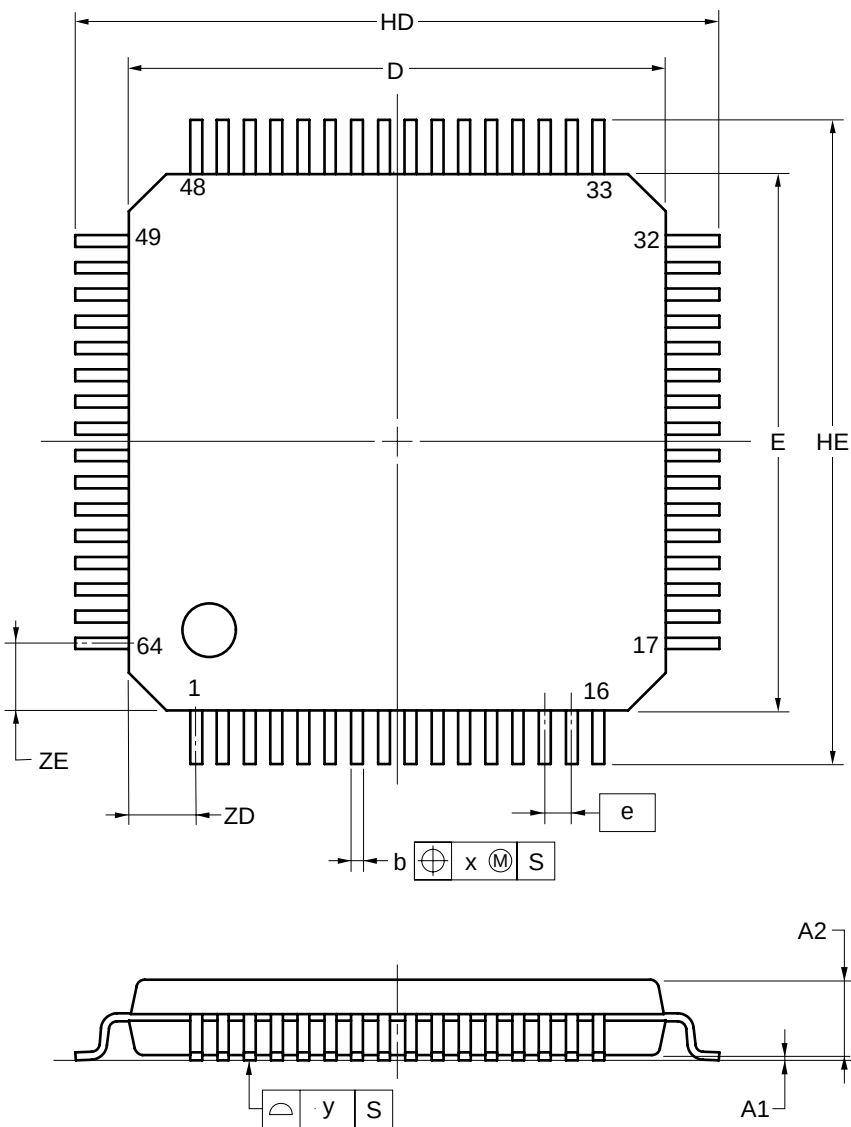


注
最大限度材料条件下, 各引线的中心线在其实际位置的
0.13 mm 内。

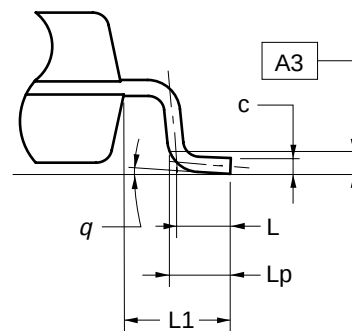
© NEC Electronics Corporation 2005

μ PD78F1007GB-GAH-AX, 78F1008GB-GAH-AX, 78F1009GB-GAH-AX

64-引脚 LQFP 塑封(密间距)(10x10)



引脚末端的细节



(单位:mm)

项目	毫米
D	10.00±0.20
E	10.00±0.20
HD	12.00±0.20
HE	12.00±0.20
A	1.60 MAX.
A1	0.10±0.05
A2	1.40±0.05
A3	0.25
b	0.20 ^{+0.07} _{-0.03}
c	0.125 ^{+0.075} _{-0.025}
L	0.50
Lp	0.60±0.15
L1	1.00±0.20
q	3°~5°
e	0.50
x	0.08
y	0.08
ZD	1.25
ZE	1.25

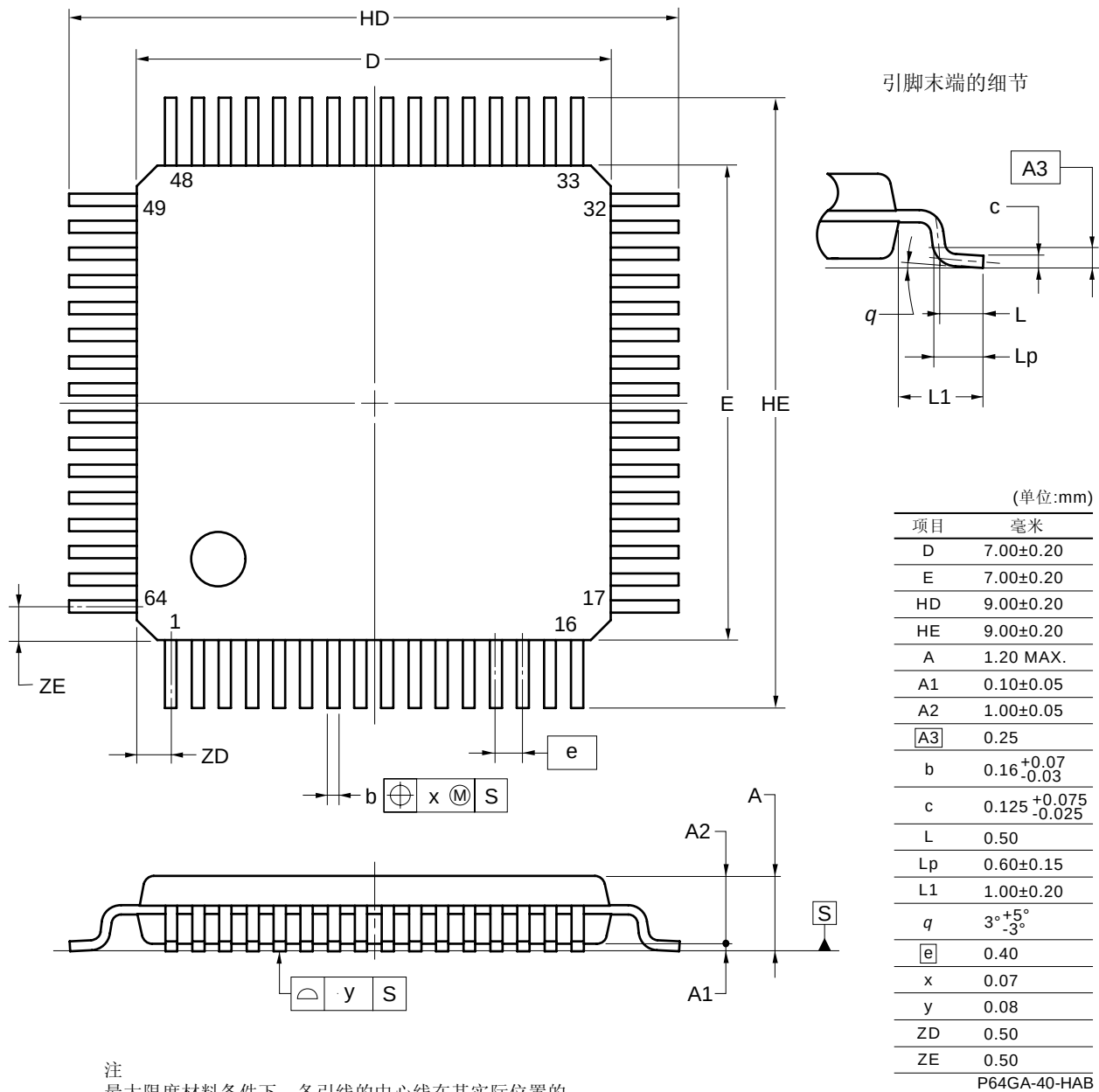
P64GB-50-GAH

注
最大限度材料条件下，各引线的中心线在其实际位置的
0.08 mm 内。

© NEC Electronics Corporation 2005

μPD78F1007GA-HAB-AX, 78F1008GA-HAB-AX, 78F1009GA-HAB-AX

64-引脚 TQFP 塑封 (密间距) (7x7)

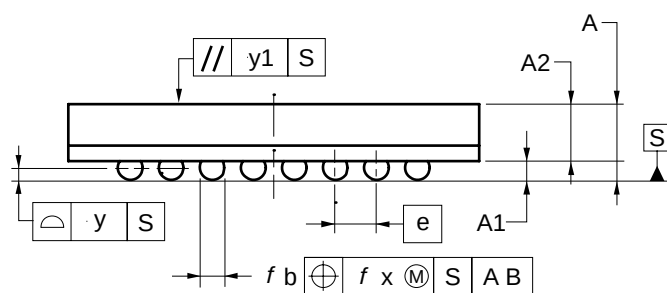
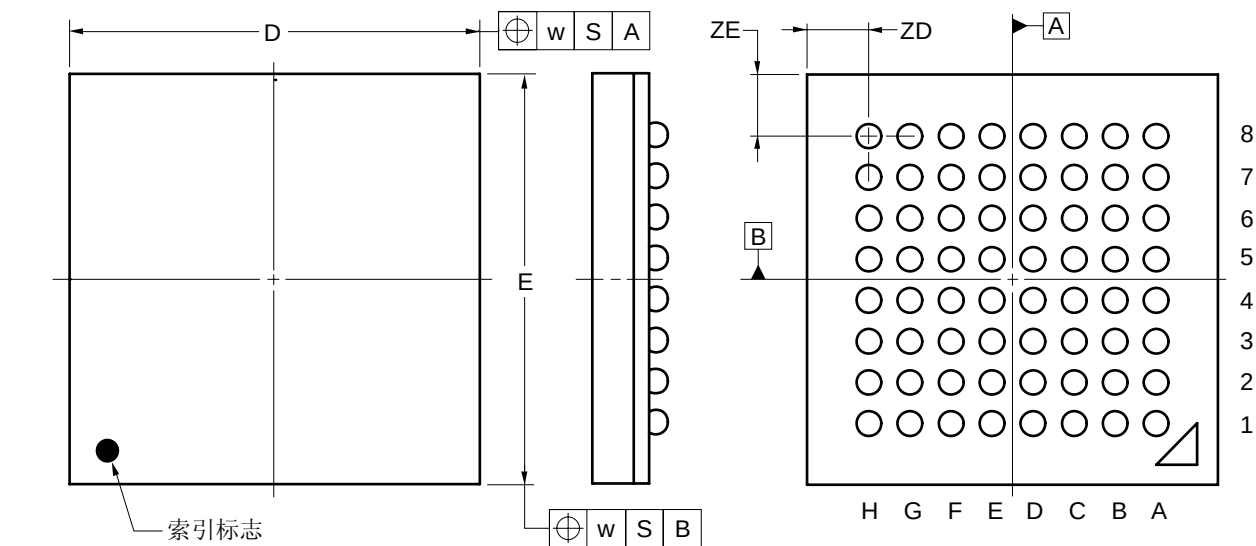


注
最大限度材料条件下, 各引线的中心线在其实位置的
0.07mm 内。

© NEC Electronics Corporation 2005

μ PD78F1007F1-AN1-A, 78F1008F1-AN1-A, 78F1009F1-AN1-A

64-引脚FBGA 塑封(5x5)



(单位 :mm)

项目	毫米
D	5.00±0.10
E	5.00±0.10
w	0.20
A	0.90±0.10
A1	0.21±0.05
A2	0.69
e	0.50
b	0.32±0.05
x	0.05
y	0.08
y1	0.20
ZD	0.75
ZE	0.75

P64F1-50-AN1

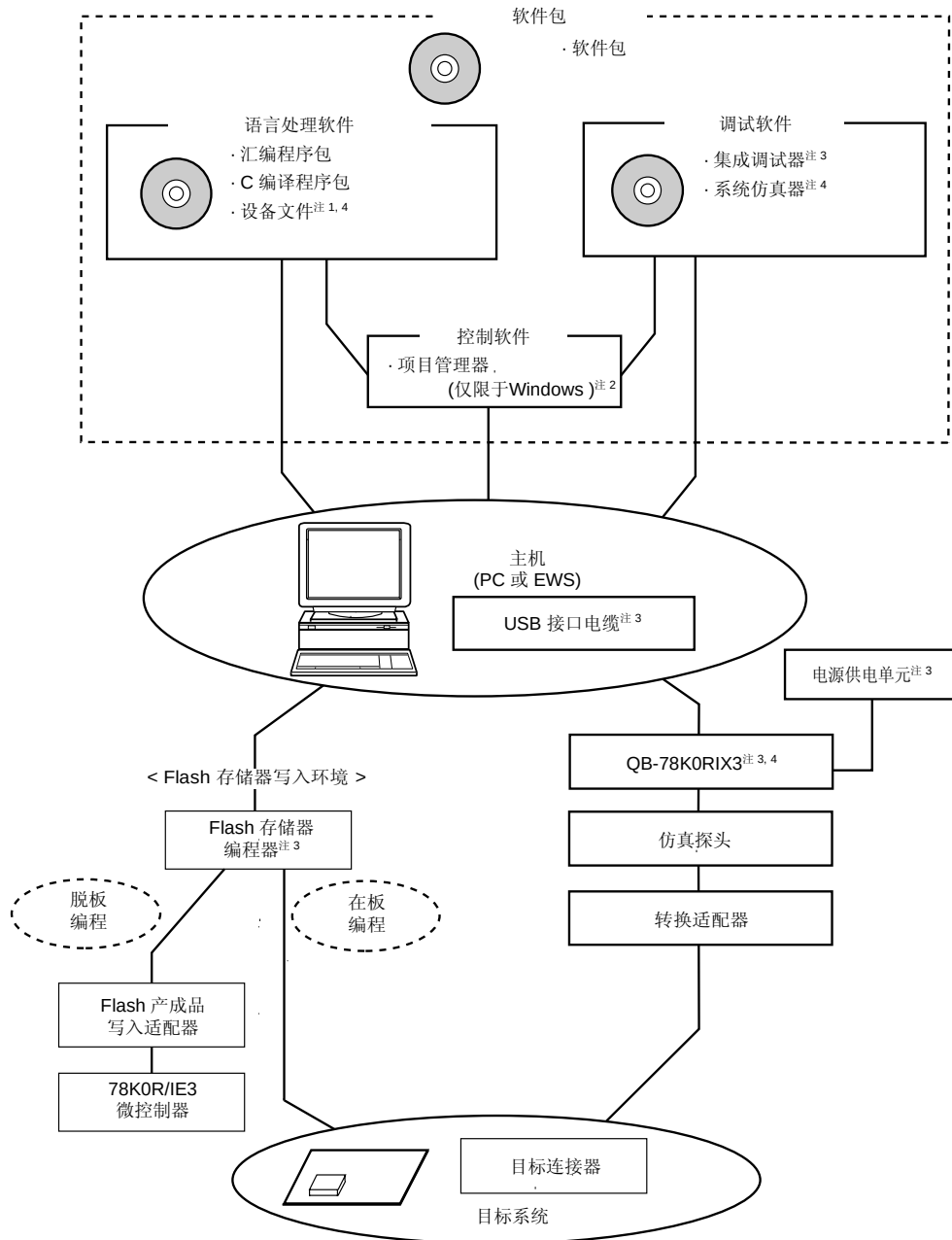
© NEC Electronics Corporation 2006

附录 A 开发工具

以下开发工具可用来开发使用 78K0R/Kx3-L 单片机的系统。

图 A-1 显示了开发工具的构成配置。

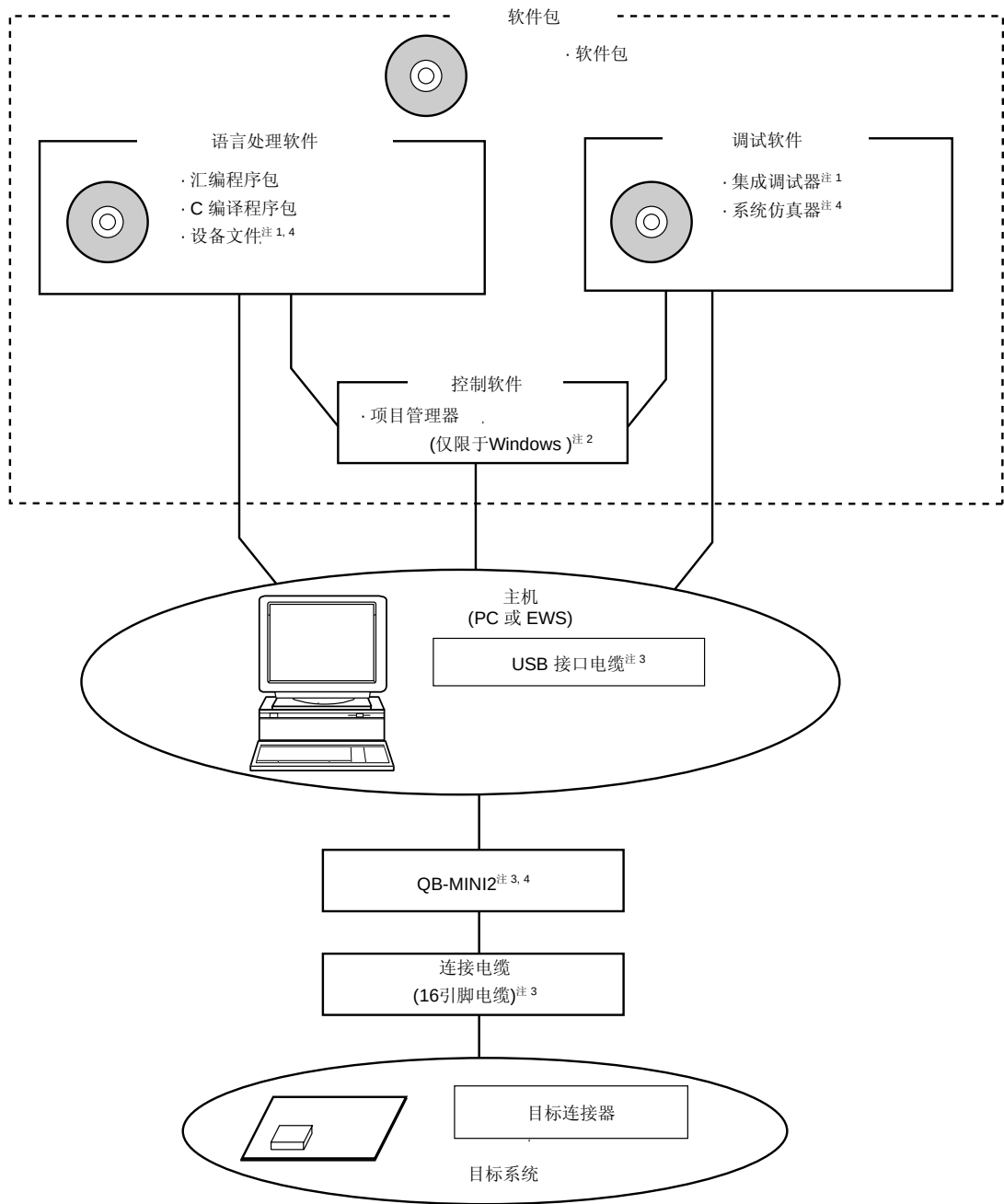
图 A-1. 开发工具配置 (1/2)

(1) 使用在线仿真器 QB-78K0RIX3^{注4}时

- 注
1. 从以下网站下载关于 78K0R/Kx3-L (DF781009) 的开发工具的设备文件：
(<http://www.necel.com/micro/ods/eng/index.html>)
 2. 项目管理器 PM+ 包含在汇编程序包内。
PM+ 只能在 Windows 下使用。
 3. 在线仿真器 QB-78K0RIX3 与集成调试器 ID78K0R-QB、带编程功能的片上调试仿真器 QB-MINI2、电源供应单元以及 USB 接口电缆一起提供。其它产品均单独出售。
 4. 正在开发之中

图 A-1. 开发工具配置 (2/2)

(2) 使用带编程功能的片上调试仿真器 QB-MINI2 时



- 注
1. 从以下网站下载关于 78K0R/Kx3-L (DF781009) 和集成调试器 (ID78K0R-QB) 的开发工具的设备文件: (<http://www.necel.com/micro/ods/eng/index.html>)
 2. 项目管理器 PM+ 包含在汇编程序包内。
PM+ 只能在 Windows 下使用。
 3. 片上调试仿真器 QB-MINI2 与 USB 接口电缆、连接电缆 (10 引脚电缆和 16 引脚电缆) 和 78K0-OCD 板一起提供。其它产品均单独出售。此外, 从以下网站下载关于 MINICUBE2 的操作 QB-MINI2 的软件: (<http://www.necel.com/micro/en/development/asia/minicube2/minicube2.html>)。
 4. 正在开发之中

A.1 软件包

SP78K0R 78K0R 串行软件包	本软件包包括用于 78K0R 微控制器的通用开发工具（软件）。

A.2 语言处理软件

RA78K0R 汇编程序包	该汇编器将用助记符写的程序转换为微控制器执行的目标代码。 该汇编器还具有自动创建符号表和转移指令优化的功能。 该汇编器应该配合设备文件（DF781009）使用。 <在 PC 环境中使用 RA78K0R 的注意事项> 汇编程序包基于 DOS 应用。其也可用于 Windows，但是，在 Windows 中需使用项目管理器（包括汇编程序包）。
CC78K0R 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。 该编译器应与汇编程序包和设备文件（均单独出售）一起使用。 <在 PC 环境中使用 CC78K0R 的注意事项> C 编译包基于 DOS 应用。其也可用于 Windows，但是，在 Windows 中需使用项目管理器（包括汇编程序包）。
DF781009 ^{#1, 2} 设备文件	本文件包含器件的特有信息。 设备文件应与工具（RA78K0R，CC78K0R，用于 78K0R 的 SM+以及 ID78K0R-QB）（均单出售）一起使用。 使用的工具不同，相应使用的操作系统和主机也不同。

- 注
1. DF781009 可与 RA78K0R，CC78K0R，用于 78K0R 的 SM+以及 ID78K0R-QB 通用。
 2. 正在开发之中

A.3 Flash 存储器编程工具

A.3.1 使用 flash 存储器编程器 PG-FP5, FL-PR5, PG-FP4 and FL-PR4 时

PG-FP5, FL-PR5, PG-FP4, FL-PR4 Flash 存储器编程器	专用于具备片上 Flash 存储的微控制器编程。
Flash 存储器编程适配器 ^注	Flash 存储器编程适配器用于连接到 flash 存储器编程器以供使用。

注 正在开发之中

- 备注
1. FL-PR4 和 FL-PR5 是 Naito Densei Machida Mfg. Co., Ltd. 的产品。
 2. 使用 flash 存储器编程适配器的最新版本。

A.3.2 使用带编程功能的片上调试仿真器 QB-MINI2 时

QB-MINI2 带编程功能的片上调试仿真器	专用于具备片上 Flash 存储的微控制器编程。其也可作为片上调试仿真器应用，当用 78K0R/Kx3-L 微控制器开发应用系统时，用于调试硬件和软件。当用作 flash 存储器编程器时，其应与连接电缆（16 引脚电缆）和用于连接主机的 USB 接口电缆一起使用。

- 备注
- 从以下网站下载关于 MINICUBE2 的操作 QB-MINI2 的软件：
[（http://www.necel.com/micro/en/development/asia/minicube2/minicube2.html）](http://www.necel.com/micro/en/development/asia/minicube2/minicube2.html)。

A.4 调试工具（硬件）

A.4.1 使用在线仿真器 QB-78K0RIX3 时

QB-78K0RIX3 ^{注1, 2} 在线仿真器	当开发使用 78K0R/Kx3-L 微控制器的应用系统时，利用在线仿真器调试其硬件和软件。其支持集成调试器（ID78K0R-QB）。该仿真器应与一个供电单元、仿真探头配合使用，USB 用于将其连接至主机。
QB-144-CA-01 检查引脚适配器	使用示波器等仪器观察波形时使用的检查引脚适配器。
QB-80-EP-01T 仿真探头	该仿真探头为柔性类型，用于连接在线仿真器和目标系统。
QB-xxxx-EA-xxx ^{注1, 2} 转换适配器	该转换适配器用于进行从在线仿真器到目标连接器的引脚转换。
QB-xxxx-YS-xxx ^{注1, 2} 空间适配器	空间适配器用于调整目标系统和在线仿真器间的高度。
QB-xxxx-YQ-xxx ^{注1, 2} YQ 连接器	YQ 连接器用于连接目标连接器和转换适配器。
QB-xxxx-HQ-xxx ^{注1, 2} 安装适配器	安装适配器用插座安装目标设备。
QB-xxxx-NQ-xxx ^{注1, 2} 目标连接器	目标连接器用于安装到目标系统。

注 1. 正在开发之中

2. 转换适配器、空间适配器、YQ 连接器、安装适配器和目标连接器的各部件编号，以及目标设备的封装描述如下：

	封装	转换适配器	空间适配器	YQ 连接器	安装适配器	目标连接器
78K0R/ KC3-L	44-引脚 LQFP 塑封 (GB-GAF 型)	QB-44GB- EA-04T	QB-44GB- YS-01T	QB-44GB- YQ-01T	QB-44GB- HQ-01T	QB-44GB- NQ-01T
	48-引脚 LQFP 塑封 (GA-HAA 型)	QB-48GA- EA-04T	QB-48GA- YS-01T	QB-48GA- YQ-01T	QB-48GA- HQ-01T	QB-48GA- NQ-01T
78K0R/ KD3-L	52-引脚 LQFP 塑封 (GB-GAG 型)	QB-52GB- EA-04T	QB-52GB- YS-01T	QB-52GB- YQ-01T	QB-52GB- HQ-01T	QB-52GB- NQ-01T
78K0R/ KE3-L	64-引脚 LQFP 塑封 (GB-GAH 型)	QB-64GB- EA-04T	QB-64GB- YS-01T	QB-64GB- YQ-01T	QB-64GB- HQ-01T	QB-64GB- NQ-01T
	64-引脚 LQFP 塑封 (GK-GAJ 型)	QB-64GK- EA-04T	QB-64GK- YS-01T	QB-64GK- YQ-01T	QB-64GK- HQ-01T	QB-64GK- NQ-01T
	64-引脚 TQFP 塑封 (GA-HAB 型)	QB-64GA- EA-01T	QB-64GA- YS-01T	QB-64GA- YQ-01T	QB-64GA- HQ-01T	QB-64GA- NQ-01T
	64-引脚 FBGA 塑封 (F1-AN1 型)	QB-64FC- EA-01T	无	无	无	QB-64FC- NQ-01T

（备注在下页给出。）

- 备注** 1. QB-78K0RIX3 与集成调试器 ID78K0R-QB、USB 接口电缆、电源供电单元、片上调试仿真器 QB-MINI2、连接电缆（10 引脚和 16 引脚电缆）以及 78K0-OCD 板一起提供。
- 使用 QB-MINI2 时，从以下网站下载开发工具的操作 QB-MINI2 的软件：
（<http://www.necel.com/micro/ods/eng/index.html>）
2. 根据部件编号不同，封装内容有以下不同：

封装内容 部件编号	在线仿真器	仿真探头	转换适配器	YQ连接器	目标连接器
QB-78K0RIX3-ZZZ	QB-78K0RIX3 ^注	无			
QB-78K0RIX3-T44GB		QB-80-EP-01T	QB-44GB-EA-04T	QB-44GB-YQ-01T	QB-44GB-NQ-01T
QB-78K0RIX3-T48GA			QB-48GA-EA-04T	QB-48GA-YQ-01T	QB-48GA-NQ-01T
QB-78K0RIX3-T52GB			QB-52GB-EA-04T	QB-52GB-YQ-01T	QB-52GB-NQ-01T
QB-78K0RIX3-T64GB			QB-64GB-EA-04T	QB-64GB-YQ-01T	QB-64GB-NQ-01T
QB-78K0RIX3-T64GK			QB-64GK-EA-04T	QB-64GK-YQ-01T	QB-64GK-NQ-01T
QB-78K0RIX3-T64GA			QB-64GA-EA-01T	QB-64GA-YQ-01T	QB-64GA-NQ-01T
QB-78K0RIX3-T64F1			QB-64FC-EA-01T	无	QB-64FC-NQ-01T

注 正在开发之中

A.4.2 使用带编程功能的片上调试仿真器 QB-MINI2 时

QB-MINI2 带编程功能的片上调试仿真器	当用 78K0R/Kx3-L 微控制器开发应用系统时，片上调试仿真器用于调试硬件和软件。其也可作为 flash 存储器编程器，专用于具有片上 flash 存储器的微控制器。当用作片上调试仿真器时，其应与连接电缆（16 引脚电缆）和用于连接主机的 USB 接口电缆一起使用。

- 备注** 从以下网站下载关于 MINICUBE2 的操作 QB-MINI2 的软件：
（<http://www.necel.com/micro/en/development/asia/minicube2/minicube2.html>）。

A.5 调试工具（软件）

用于 78K0R ^注 的 SM+ 系统仿真器	用于 78K0R 的 SM+ 是基于 Windows 的软件。 当在主机上模拟目标系统的操作时，用来实现在 C 源程序层面或汇编源程序层面的调试。 使用 78K0R 的 SM+ 允许在独立于硬件开发的基础上进行应用逻辑测试和性能测试，从而提供较高的开发效率和软件质量。 78K0R 的 SM+ 应该配合设备文件（DF781009）使用。
ID78K0R-QB 集成调试器	本调试器支持 78K0R 微控制器的在线仿真。ID78K0R-QB 是基于 Windows 的软件。 它具有改进的 C-兼容调试功能，并且，使用一个集成窗口显示源程序追踪执行结果，以及相关的源文件显示、反汇编显示、存储显示。本调试器应该配合设备文件使用（DF781009）。

注 正在开发之中

B.1 本版中主要修订内容

(1/3)

页码	说明	类别
第一章 概述		
p.24	修改 1.4.3 78K0R/KE3-L 中 64-引脚 FBGA (5×5) 塑封	(c)
第二章 引脚功能		
p.60	修改 2.2.14 RESET 中的描述	(c)
p.60	删除 2.2.15 REGC 中的描述	(c)
第三章 CPU 架构		
pp.75, 76	添加 3.1.2 镜像区的描述	(b)
pp.95, 96	修改 表 3-6. 扩展 SFR (2nd SFR) 列表 (3/4), (4/4)	(b)
第四章 端口功能		
p.138	修改 图 4-19. P80 的框图	(c)
p.139	修改 图 4-20. P81 和 P83 的框图	(c)
p.140	修改 图 4-21. P82 的框图	(c)
第五章 时钟发生器		
p.181	添加 图 5-7. 20 MHz 内部高速振荡控制寄存器 (DSCCTL) 的格式中注意事项 1 和 3	(c)
p.184	修改 图 5-9. 操作速度模式控制寄存器 (OSMC) 的格式中注意事项 4 和 5	(c)
p.206	修改 表 5-4 中 (3) 复位解除后 (ACPU 工作于副系统时钟 (D))	(c)
p.209	修改表 5-4 中 (9) CPU 时钟从高速系统时钟 (C) 改变为副系统时钟 (D)	(c)
p.210	修改 表 5-4 中 (11) CPU 时钟从副系统时钟 (D) 改变为高速系统时钟 (C)	(c)
p.210	修改表 5-4 中 (12) CPU 时钟从 20 MHz 内部高速振荡时钟 (J) 改变为内部高速振荡时钟 (B)	(c)
p.211	修改表 5-4 中 (14) • 当 CPU 工作于内部高速振荡时钟 (B) 时 STOP 模式 (H) 设置, • 当 CPU 工作于高速系统时钟 (C) 时 STOP 模式 (I) 设置	(c)
pp.212, 213	修改 表 5-5. 改变 CPU 时钟	(c)
第六章 定时器阵列部件		
p.228	修改 图 6-6 定时器模式寄存器 0n (TMR0n) 的格式中 CCS0n 位的描述	(b)
p.248	修改 6.4.3 (1) 定时器工作期间改变寄存器 TO0, TOE0, TOL0 和 TOM0 中的设定值中的描述	(c)
p.257	添加 6.7.1 (1) 间隔定时器的描述	(b)
p.258	修改 图 6-35 作为间隔定时器/方波输出操作的框图	(b)
pp.260, 261	添加 (2) 当 f _{sub} /4 选作计数时钟 至 图 6-37. 作为间隔定时器/方波输出操作期间寄存器设置内容举例	(b)
p.262	修改 图 6-38 间隔定时器/方波输出功能的操作步骤	(b)

备注 上表中分类修订“类别”如下:

- (a): 错误校正, (b): 添加/修改说明, (c): 添加/修改描述或注,
 (d): 添加/修改封装, 部件编号或管理区域 (e): 添加/修改相关文档

(2/3)

页码	说明	类别
第六章 定时器阵列部件 (续)		
p.265	修改 图 6-42 当使用外部事件计数器功能时操作步骤中操作期间的描述	(a)
p.269	修改 图 6-46 当使用频率分频器功能时操作步骤中操作期间的描述	(a)
p.273	修改 图 6-50 当使用输入脉冲间隔测量功能时操作步骤中操作期间的描述	(a)
p.277	修改 图 6-54 当使用输入信号高电平/低电平宽度测量功能时操作步骤中操作期间的描述	(a)
p.284	修改 图 6-59 当使用 PWM 功能时操作步骤中操作期间的描述	(c)
p.291	修改 图 6-64 单次脉冲输出功能操作步骤中操作期间的描述	(c)
p.298	修改 图 6-69 当使用 PWM 输出功能时操作步骤中操作期间的描述	(c)
第七章 实时计数器		
p.303	修改 图 7-3 实时计数器控制寄存器 0 (RTCC0) 的格式中 CT2 至 CT1 位的描述	(c)
p.304	修改 图 7-4 实时计数器控制寄存器 1 (RTCC1) 的格式 WALE 位的描述	(c)
p.306	添加 注意事项 3 至 图 7-5 实时计数器控制寄存器 2 (RTCC2) 的格式	(c)
pp. 308, 310, 312	修改 7.3 (7) 分计数寄存器 (MIN), (8) 小时计数寄存器 (HOUR), (9) 天计数寄存器 (DAY), (11) 月计数寄存器 (MONTH) 以及 (12) 年计数寄存器 (YEAR) 中的描述	(c)
p.313	添加 7.3 (13) 钟表错误校正寄存器 (SUBCUD)	(b)
第八章 比较器/可编程增益放大器		
p.331	添加 注意事项 3 至 图 8-3. 可编程增益放大器控制寄存器 (OAM) 的格式	(c)
p.332	添加 注意事项 5 至 图 8-4. 比较器 n 控制寄存器 (CnCTL) 的格式	(c)
p.333	添加 注意事项 4 至 图 8-5. 比较器 n 内部参考电压选择寄存器 (CnRVM) 的格式	(c)
第九章 时钟输出/蜂鸣器输出控制器		
p.344	修改 图 9-2. 时钟输出选择寄存器 n (CKSn) 的格式中注意事项 2	(c)
p.345	修改 9.4.1 作为输出引脚操作中备注 1	(b)
p.345	修改 图 9-4. 远程控制输出应用举例	(b)
第十一章 A/D 转换器		
p.377	修改 11.6 (1) STOP 模式下操作电流中的描述	(c)
第十二章 串行阵列部件		
p.502	修改 图 12-95. 数据接收时序图中 (b) 当接收最末数据时	(c)

备注 上表中分类修订“类别”如下:

- (a): 错误校正, (b): 添加/修改说明, (c): 添加/修改描述或注,
 (d): 添加/修改封装, 部件编号或管理区域, (e): 添加/修改相关文档

页码	说明	类别
第十三章 串行接口 IICA		
p.516	添加 注意事项 3 至 图 13-3. IICA 移位寄存器 (IICA) 的格式	(c)
pp.520 至 523	修改 图 13-6. IICA 控制寄存器 0 (IICCTL0) 的格式	(b) , (c)
p.528	添加 图 13-9. Format of IICA 控制寄存器 1 (IICCTL1) 的格式 中 WUP 位的描述	(c)
p.533	添加 13.4.2 用 IICWL 和 IICWH 寄存器设置传输时钟	(c)
p.546	删除 图 13-23. 基于地址匹配设置 WUP = 0 时的流程 (包括扩展码接收) 中的描述	(b)
p.546	删除 13.5.13 唤醒功能 中的描述	(b)
p.547	修改 图 13-24. 用非 INTIICA 解除 STOP 模式后作为主机设备操作时	(b)
p.547	删除 图 13-25. 用非 INTIICA 解除 STOP 模式后作为从机设备操作时 (当不需要作为主机设备操作时)	(b)
第十五章 DMA 控制器		
p.615	修改 表 15-2 DMA 传输的响应时间	(c)
第十六章 中断功能		
p.619	修改 表 16-1 中断源列表 (2/2)	(c)
p.621	添加 (C) 外部可屏蔽中断 (INTKR) 至 图 16-1. 中断功能的基本配置 (2/2)	(c)
第十七章 按键中断功能		
p.642	修改 17.3 (1) 按键返回模式寄存器 (KRM) 中的 注意事项 2	(c)
第十八章 待机功能		
p.650	添加 注 18-3 由中断请求产生解除 HALT 模式	(c)
第二十一章 低电压检测器		
p.676	修改 图 21-2. 低压检测寄存器 (LVIM) 的格式 中 注 4 和 注意事项 1	(b) , (c)
pp.680, 682	修改 21.4.1 (1) 当检测供应电压 (VDD) 的等级时 中的描述	(b) , (c)
p.684	修改 21.4.1 (2) 当检测从外部输入引脚输入电压 (EXLVI) 的等级时 中的描述	(b) , (c)
pp.686, 688	修改 21.4.2 (1) 当检测供应电压 (VDD) 的等级时 中的描述	(b) , (c)
p.690	修改 21.4.2 (2) 当检测从而不输入引脚输入电压 (EXLVI) 的等级时 中的描述	(b) , (c)
p.695	修改 图 21-12. 从 LVI 复位源产生时刻到 LVI 复位产生或解除时刻的延迟	(b)
第二十四章 FLASH 存储器		
p.717	添加 注意事项 5 至 24.8 用自编程进行 Flash 存储器编程	(c)
p.721	修改 24.8.2 Flash 屏蔽窗口功能 中的描述	(c)
附录 B 修订历史		
pp.811 至 813	添加本章	(c)

备注 上表中分类修订“类别”如下：

- (a)：错误校正，(b)：添加/修改说明，(c)：添加/修改描述或注，
(d)：添加/修改封装，部件编号或管理区域，(e)：添加/修改相关文档

详细信息请联系:

中国区

MCU 技术支持热线:

电话: +86-400-700-0606 (普通话)

服务时间: 9:00-12:00, 13:00-17:00 (不含法定节假日)

网址:

<http://www.cn.necel.com/> (中文)

<http://www.necel.com/> (英文)

[北京]

日电电子(中国)有限公司

中国北京市海淀区知春路 27 号

量子芯座 7, 8, 9, 15 层

电话: (+86) 10-8235-1155

传真: (+86) 10-8235-7679

[深圳]

日电电子(中国)有限公司深圳分公司

深圳市福田区益田路卓越时代广场大厦 39 楼

3901, 3902, 3909 室

电话: (+86) 755-8282-9800

传真: (+86) 755-8282-9899

[上海]

日电电子(中国)有限公司上海分公司

中国上海市浦东新区银城中路 200 号

中银大厦 2409-2412 和 2509-2510 室

电话: (+86) 21-5888-5400

传真: (+86) 21-5888-5230

[香港]

香港日电电子有限公司

香港九龙旺角太子道西 193 号新世纪广场

第 2 座 16 楼 1601-1613 室

电话: (+852) 2886-9318

传真: (+852) 2886-9022

2886-9044

上海恩益禧电子国际贸易有限公司

中国上海市浦东新区银城中路 200 号

中银大厦 2511-2512 室

电话: (+86) 21-5888-5400

传真: (+86) 21-5888-5230

[成都]

日电电子(中国)有限公司成都分公司

四川省成都市二环路南三段 15 号

天华大厦 608 室

电话: (+86)28-8512-5224

传真: (+86)28-8512-5334

[长春]

日电电子(中国)有限公司长春分公司

吉林省长春市朝阳区

西安大路 727 号中银大厦 A 座 1609 室

电话: (+86)431-8859-7533 / 8859-8533

传真: (+86)431-8680-2944

[大连]

日电电子(中国)有限公司长春分公司

大连市中山路 88 号天安国际大厦 2701 室

电话: (+86)411-8230-8815 / 8230-8825

传真: (+86)411-8230-8835