To our customers,

Old Company Name in Catalogs and Other Documents

On April 1st, 2010, NEC Electronics Corporation merged with Renesas Technology Corporation, and Renesas Electronics Corporation took over all the business of both companies. Therefore, although the old company name remains in this document, it is a valid Renesas Electronics document. We appreciate your understanding.

Renesas Electronics website: http://www.renesas.com

April 1st, 2010
Renesas Electronics Corporation

Issued by: Renesas Electronics Corporation (http://www.renesas.com)

Send any inquiries to http://www.renesas.com/inquiry.

RENESAS

用户手册

V850E/IF3, V850E/IG3

32 位单片微控制器

硬件

V850E/IF3:

μPD70F3451

μPD70F3452

V850E/IG3:

μPD70F3453

μPD70F3454

文档编号 U18279CA2V0UD00（第 2 版）
发布日期 2009 年 9 月

[备忘录]

关于CMOS器件的注意事项

① **输入引脚处的电压波形**

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

② **未使用的输入引脚的处理**

CMOS器件的输入端保持开路可能导致故障。如果一个输入引脚未被连接，则由于噪声等原因可能会产生内部输入电平，从而导致故障。CMOS器件的操作特性与双极性或NMOS器件不同。CMOS器件的输入电平必须借助上拉或下拉电路固定于高电平或低电平。每一个未使用引脚都应该连接到V_{DD}或GND，如果有可能作为输出引脚时，需要通过附加电阻连接到V_{DD}或GND。对未使用引脚的处理因器件而异，必须遵循与器件相关的规定和说明。

③ **ESD防护措施**

如果MOS器件周围有强电场，将会击穿氧化栅极，从而影响器件的工作。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。对于环境必须进行适当的控制。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。半导体器件的存放和运输必须使用抗静电容器、静电屏蔽袋或导电材料包装。所有包括工作台和工作面的测试和测量工具必须良好接地。操作员应当佩戴静电消除腕带以保证良好接地。不能用手直接接触半导体器件。对于装配有半导体器件的PW板也应采取类似的静电防范措施。

④ **初始化之前的状态**

在上电时MOS器件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器件并没有被初始化。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。器件在收到复位信号后才进行初始化。具有复位功能的器件在上电后必须立即进行复位操作。

⑤ **电源开关顺序**

一个器件内部工作和外部接口使用不同电源的情况下，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，可能会导致器件的内部组件过电压，产生异常电流，从而引起内部组件的误操作和性能的劣化。

对每个器件电源的正确开关顺序，必须依据器件的规格说明分别进行判断。

⑥ **电源关闭状态下的输入信号**

不要向没有加电的器件输入信号或提供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，从而引起器件的误操作，并产生异常电流，从而使内部组件劣化。

每个器件电源关闭时的信号输入必须依据器件的规格说明分别进行判断。

注意事项： 该产品使用的 SuperFlash® 科技授权来自 Silicon Storage Technology, Inc。

EEPROM 是 NEC Electronics Corporation 的商标。
MINICUBE是 NEC Electronics Corporation在日本和德国的注册商标，以及在美国的商标。
SuperFlash 是 Silicon Storage Technology, Inc.在其它国家，包括美国和日本，注册的商标。

引言

读者

本手册适用于那些希望了解 V850E/IF3(μPD70F3451，70F3452)和 V850E/IG3(μPD70F3453，70F3454)的功能，并使用 V850E/IF3 和 V850E/IG3 设计应用系统的用户。

目的

本手册适用于帮助用户了解下述编排组织中所述的 V850E/IF3 和 V850E/IG3 的硬件功能。

编排组织

本手册分为两个部分： 硬件（本手册）和 架构（V850E1 架构用户手册）。

硬件	架构
• 引脚功能	• 数据类型
• CPU 功能	• 寄存器组
• 片上外围功能	• 指令格式和指令集
• Flash 存储器编程	• 中断和异常处理
• 电气规范	• 流水线操作

如何阅读本手册

假定本手册的读者具备电气工程、逻辑电路以及微控制器领域的常识。

要了解 V850E/IF3 和 V850E/IG3 的全部功能。
→按照**目录**阅读本使用手册。

要查找已知名称的寄存器的详细情况
→请阅读**附录 B 寄存器索引**

寄存器格式说明。
→位（其编号在各寄存器格式图的尖括（<>）内）名称在设备文件中定义为保留字。

要了解指令功能的详细情况
→参见 V850E1 架构用户手册。

要了解 V850E/IF3 和 V850E/IG3 的电气规范。
→请参阅**第二十八章 电气特性（目标）**

在本手册中，将“xxx 寄存器的 yyy 位“描述为”xxx.yyy 位”。请小心注意，如果在程序中描述“xxx.yyy”，编译程序/汇编程序则不能正确识别它。

标记 <R> 表示主要的修订内容。拷贝 PDF 文件中的“<R>”，并在“查找”区域内指定“<R>”，就可以很方便的搜索修订内容。

惯例

数据有效位：	高位在左，低位在右 低电平有效表示：
有效低电平表示：	$\overline{xxx}$（在引脚或信号名称上有上划线）
存储器映射地址：	高位地址在顶部，低位地址在底部
注：	正文中使用'**注**'标记的脚注
注意事项：	需特别注意的信息
备注：	补充信息
数字表示法：	二进制 ... xxxx 或 xxxxB
	十进制 ... xxxx
	十六进制 ... xxxxH
前缀表示 2 的幂（地址空间、存储容量 ）：	
	K （千）： $2^{10} = 1,024$
	M （兆）： $2^{20} = 1,024^2$
	G （千兆）： $2^{30} = 1,024^3$
数据类型：	字 … 32 位
	半字 … 16 位
	字节 … 8 位

相关文档　　本手册中的相关文档可能包括初稿版本。然而，初稿版本并未标注如此。

V850E/IF3 和 V850E/IG3 相关文档

文档名称	文档编号
V850E1 架构用户手册	U14559E
V850E/IF3, V850E/IG3 硬件用户手册	本手册
V850E/IF3, V850E/IG3 串行通信(UARTA)的样例程序应用笔记	U18723E
V850E/IF3, V850E/IG3 串行通信(UARTB)的样例程序应用笔记	U18724E
V850E/IF3, V850E/IG3 串行通信(CSIB)的样例程序应用笔记	U18725E
V850E/IF3, V850E/IG3 串行通信(I^2C) 的样例程序应用笔记	U18726E
V850E/IF3, V850E/IG3 DMA 功能的样例程序应用笔记	U18727E
V850E/IF3, V850E/IG3 定时器 M 的样例程序应用笔记	U18728E
V850E/IF3, V850E/IG3 看门狗定时器的样例程序应用笔记	U18729E
V850E/IF3, V850E/IG3 定时器 AA 的样例程序应用笔记	U18730E
V850E/IF3, V850E/IG3 定时器 AB 的样例程序应用笔记	U18731E
V850E/IF3, V850E/IG3 定时器 T 的样例程序应用笔记	U18732E
V850E/IF3, V850E/IG3 端口功能的样例程序应用笔记	U18733E
V850E/IF3, V850E/IG3 时钟发生器的样例程序应用笔记	U18734E
V850E/IF3, V850E/IG3 待机功能的样例程序应用笔记	U18735E
V850E/IF3, V850E/IG3 中断功能的样例程序应用笔记	U18736E
V850E/IF3, V850E/IG3 A/D 转换器 0 和 1 的样例程序应用笔记	U18737E
V850E/IF3, V850E/IG3 A/D 转换器 2 的样例程序应用笔记	U18738E
V850E/IF3, V850E/IG3 低电压检测器(LVI)功能的样例程序应用笔记	U18739E
V850E/IF3, V850E/IG3 由定时器 AB，定时器 Q 选项，定时器 AA，A/D 转换器 0、1 控制的 6 相 PWM 输出应用笔记	U18717E

开发工具的相关文档（用户手册）

文档名称		文档编号
QB-V850EIX3 在线仿真器		U18651E
QB-V850MINI 片上调试仿真器		U17638E
QB-MINI2 具有编程功能的片上调试仿真器		U18371E
CA850 Ver. 3.20 C 编译程序包	U18512E	U17293E
	U18513E	U17291E
	U18514E	U17292E
	U18515E	U17294E
PM+ Ver. 6.30 项目管理器		U18416E
ID850QB Ver. 3.40 集成调试器	U18604E	U17964E
TW850 Ver. 2.00 性能分析调谐工具		U17241E
SM+ 系统仿真器	U18601E	U16218E
	U18212E	U14873E
RX850 Ver. 3.20 实时操作系统	U13430E	U13430E
	U17419E	U17419E
	U13431E	U13431E
	U17420E	U17420E
RX850 Pro Ver. 3.21 实时操作系统	U18165E	U13773E
	U17421E	U17421E
	U18164E	U13772E
	U17422E	U17422E
AZ850 Ver. 3.30 系统性能分析器		U17423E
PG-FP4 Flash 存储器编程器		U15260E
PG-FP5 Flash 存储器编程器		U18865E

目 录

第一章　导言

V850E/IF3 和 V850E/IG3 是 NEC 电子 V850 单片微控制器系列产品，本章给出 V850E/IF3 和 V850E/IG3 的概况。

1.1　概述

V850E/IF3 和 V850E/IG3 是 32 位单片微控制器，它使用 V850E1 CPU 内核，并内置 ROM/RAM 和各种相关外围功能，例如 DMA 控制器、定时器/计数器、看门狗定时器、串行接口、A/D 转换器以及片上调试功能。

除了高速实时响应特性和单时钟基本指令之外，V850E/IF3 和 V850E/IG3 还支持乘法指令、饱和运算指令、位操作指令等为数字伺服控制应用而优化的指令，这些指令由硬件乘法器来实现。此外，作为一种实时控制系统，V850E/IF3 和 V850E/IG3 还能够在各种应用中实现卓越的性价比，例如电机变频控制。

表 1-1 列出了 V850E/IF3 和 V850E/IG3 系列产品。

表 1-1.　V850E/IF3 和 V850E/IG3 产品列表

功能 部件编号		封装	ROM		RAM 大小	工作频率（MAX.）	可屏蔽中断		不可屏蔽中断
			类型	大小			外部	内部	
V850E/IF3	μPD70F3451	80GC	Flash 存储器	128 KB	8 KB	64 MHz	15	73	1
	μPD70F3452	80GC		256 KB	12 KB				
V850E/IG3	μPD70F3453	100GC		128 KB	8 KB		21	74	
	μPD70F3453	100GF		128 KB	8 KB				
	μPD70F3454	100GC		256 KB	12 KB				
	μPD70F3454	100GF		256 KB	12 KB				
<R>	μPD70F3454	161F1		256 KB	12 KB				

备注　**1.** 80GC（V850E/IF3）：　80-针塑封 LQFP（14 × 14）

100GC（V850E/IG3）：　100-针塑封 LQFP（密间距）（14 × 14）

100GF（V850E/IG3）：　100-针塑封 LQFP（14 × 20）

<R>　161F1（V850E/IG3）：　161-针塑封 FBGA（10 × 10）

2. 本手册中的 V850E/IG3 的部件编号列举如下。

- GF 版本

  μPD70F3453GF-GAS-AX，70F3454GF-GAS-AX

表 1-2 显示了 V850E/IF3 和 V850E/IG3 之间的功能区别。

表 1-2. 850E/IF3 和 V850E/IG3 之间的功能区别

项目		V850E/IF3	V850E/IG3
端口功能	I/O	44	56
	输入	4	8
	片内上拉电阻	44	56
中断源		外部中断： 15 内部中断： 74	外部中断： 21 内部中断： 75
<R> 外部总线功能		无	支持（仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A）
定时器 AA0 至 AA4		定时器 AA0（无 I/O） 定时器 AA1（无 I/O） 定时器 AA2 定时器 AA3（无 I/O） 定时器 AA4	定时器 AA0（无 I/O） 定时器 AA1（无 I/O） 定时器 AA2 定时器 AA3 定时器 AA4
定时器 T0, T1		定时器 T0（无 I/O） 定时器 T1	定时器 T0 定时器 T1
电机控制功能	高阻抗输出控制引脚	TOA2OFF TOB0OFF TOB1OFF	TOA2OFF TOA3OFF TOB0OFF TOB1OFF
A/D 转换器 2	模拟输入	4 通道	8 通道
片上调试功能	片上调试仿真器	NIMICUBE2	NIMICUBE NIMICUBE2
用于外部引脚的供电电源		EV_{DD0}, EV_{DD1}	EV_{DD0} 至 EV_{DD2}
封装		80-针塑封 LQFP（14 × 14）	100-针塑封 LQFP（14 × 14） 100-针塑封 LQFP（14 × 20） <R> 161-针塑封 FBGA（10 × 10）

1.2 V850E/IF3

1.2.1 特性（V850E/IF3）

○ 最短指令执行周期：
15.6 ns（内部运行于 64 MHz）

○ 通用寄存器：32 位 × 32 个

○ CPU 特性：带符号乘法（16 位 × 16 位 → 32 位或 32 位 × 32 位 → 64 位）： 1 至 2 个时钟。
饱和操作指令（包括上溢/下溢出检测功能）。
32 位移位指令： 1 个时钟
位操作指令。
长/短两种格式的数据装载/存储指令。
带符号装载指令

○ 内部存储器：

部件编号	内部 ROM	内部 RAM
μPD70F3451	128 KB（flash 存储器）	8 KB
μPD70F3452	256 KB（flash 存储器）	12 KB

○ 片上调试功能：支持 **MINICUBE®2**

○ 中断和异常处理：
不可屏蔽中断：1 个中断源（外部：无，内部：1）
可屏蔽中断：88 个中断源（外部：15，内部：73）
软件异常：32 个中断源
异常陷阱：2 个中断源

○ DMA 控制器：4 通道
传输单元：8 位/16 位
最大传输计数：65,536（2^{16}）
传输类型：2-周期
传输模式：单次/单步/块
传输目标：片上外围 I/O ↔ 内部 RAM
片上外围 I/O ↔ 片上外围 I/O
传输请求：片上外围 I/O/软件
下一地址设置功能

○ I/O 线：总共：48（输入端口： 4， I/O 端口： 44）

○ 定时器功能：	16 位间隔定时器 M（TMM）：	4 通道
	16 位定时器/事件计数器 AA（TAA）：	5 通道
	16 位定时器/事件计数器 AB（TAB）：	2 通道
	16 位定时器/事件计数器 T（TMT）：	2 通道
	电机控制功能（使用定时器TAB：2 通道（TAB0，TAB1）， TAA：2 通道（TAA0，TAA1））	
	带死区时间的 16 位精确 6-相 PWM 功能：2 通道	
	高阻抗输出控制功能	
	定时器调谐操作产生 A/D 触发的功能	
	任意的周期设置功能	
	任意的死区时间设置功能	
	看门狗定时器：	1 通道

○ 串行接口：

异步串行接口 A（UARTA）
异步串行接口 B（UARTB）
3 线长度可变串行接口 B（CSIB）
I^2C总线接口（I^2C）

UARTA0/CSIB0：	1 通道
UARTA1/I2C：	1 通道
UARTA2/CSIB1：	1 通道
UARTB/CSIB2：	1 通道

○ A/D 转换器：

12 位分辨率 A/D 转换器（A/D 转换器 0 和 1）： 5 通道 + 5 通道（2 单元）

A/D 转换器 0 的一个通道和 A/D 转换器 1 的三个通道提供一个运算放大器用于输入电平放大，提供一个比较器用于过电压检测。

10 位分辨率 A/D 转换器（A/D 转换器 2）： 4 通道

○ 时钟发生器：

可以连接 4 至 8 MHz 谐振器（禁止外部时钟输入）
通过PLL 时钟合成器的倍频功能（固定为 8 倍频，f_{XX} = 32 至 64 MHz）
CPU时钟分频功能（f_{XX}，$f_{XX}/2$，$f_{XX}/4$，$f_{XX}/8$）

○ 省电功能： HALT/IDLE/STOP 模式

○ 上电清零功能：

○ 低电压检测功能：

○ 封装： 80 针塑封 LQFP（14 × 14）

○ 工作供电电压：

$V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = AV_{DD0} = AV_{DD1} = AV_{DD2}$ = 4.0 至 5.5 V
（当 A/D 转换器 0, 1 或 2 工作时）
$V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = AV_{DD0} = AV_{DD1} = AV_{DD2}$ = 3.5 至 5.5 V
（当 A/D 转换器 0 至 2 都不工作时）

1.2.2 应用领域（V850E/IF3）

- 消费设备（比如变频空调、洗衣机、干燥机、冰箱等）
- 工业设置（比如电机控制、通用变频器等）

1.2.3 订购信息（V850E/IF3）

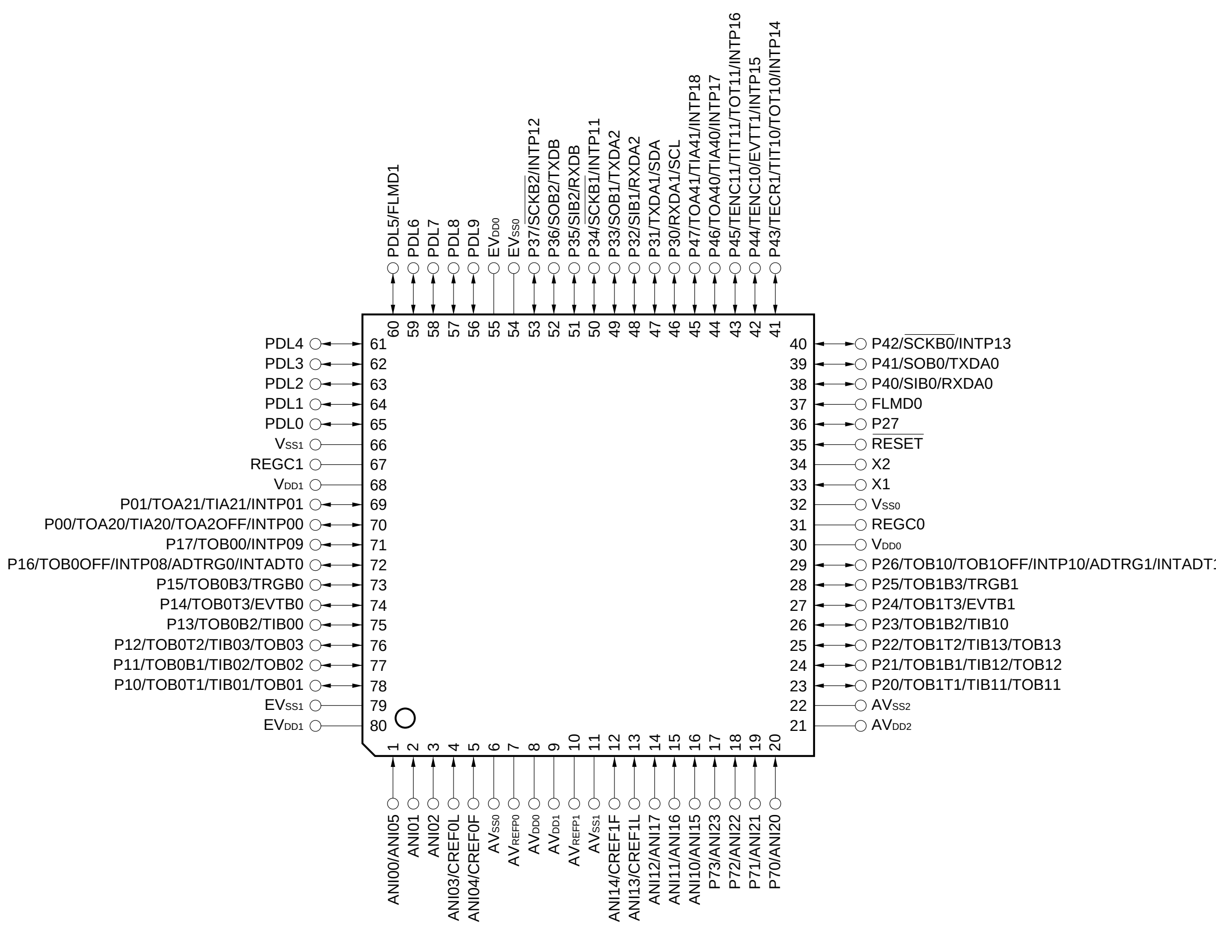
部件编号	封装	内部 ROM
μPD70F3451GC-UBT-A	80-针塑封 LQFP（14 × 14）	Flash 存储器（128 KB）
μPD70F3452GC-UBT-A	80-针塑封 LQFP（14 × 14）	Flash 存储器（256 KB）

备注 V850E/IF3 微控制器是无铅产品。

1.2.4　引脚配置（V850E/IF3）

- 80 针塑封 LQFP（14 × 14）
  μPD70F3451GC-UBT-A
  μPD70F3452GC-UBT-A

俯视图

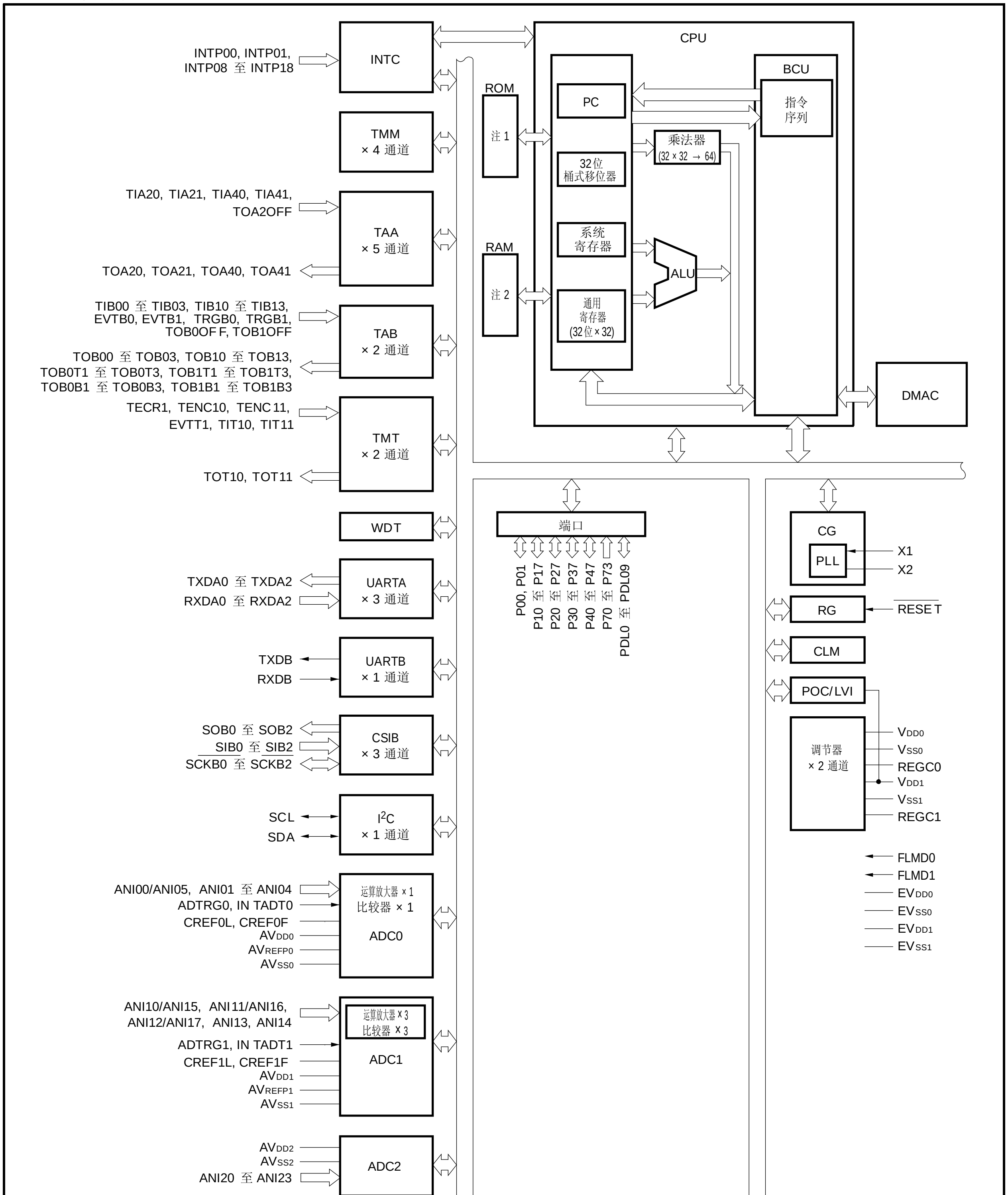


引脚名称（V850E/IF3）

ADTRG0, ADTRG1：	A/D 触发输入
ANI00 至 ANI05,	
ANI10 至 ANI17,	
ANI20 至 ANI23：	模拟输入
AV_{DD0} 至 AV_{DD2}：	模拟供电电源
AV_{REFP0}, AV_{REFP1}：	模拟参考电压
AV_{SS0} 至 AV_{SS2}：	模拟地
CREF0F, CREF1F,	
CREF0L, CREF1L：	比较器参考电压
EV_{DD0}, EV_{DD1}：	端口的供电电源
EV_{SS0}, EV_{SS1}：	端口的地
EVTB0, EVTB1,	
EVTT1：	定时器事件计数输入
FLMD0, FLMD1：	Flash 编程模式
INTADT0, INTADT1,	
INTP00, INTP01,	
INTP08 至 INTP18：	外部中断输入
P00, P01：	端口 0
P10 至 P17：	端口 1
P20 至 P27：	端口 2
P30 至 P37：	端口 3
P40 至 P47：	端口 4
P70 至 P73：	端口 7
PDL0 至 PDL9：	端口 DL
REGC0, REGC1：	稳压器控制
$\overline{RESET}$：	复位
RXDA0 至 RXDA2,	
RXDB：	接收数据
$\overline{SCKB0}$ 至 $\overline{SCKB2}$：	串行时钟
SCL：	串行时钟
SDA：	串行数据
SIB0 至 SIB2：	串行输入
SOB0 至 SOB2：	串行输出
TECR1：	定时器编码器清除输入
TENC10, TENC11：	定时器编码器输入
TIA20, TIA21,	
TIA40, TIA41,	
TIB00 至 TIB03,	
TIB10 至 TIB13,	
TIT10, TIT11：	定时器触发输入
TOA20, TOA21,	
TOA40, TOA41,	
TOB00 至 TOB03,	
TOB0B1 至 TOB0B3,	
TOB0T1 至 TOB0T3,	
TOB10 至 TOB13,	
TOB1B1 至 TOB1B3,	
TOB1T1 至 TOB1T3,	
TOT10, TOT11：	定时器输出
TOA2OFF,	
TOB0OFF, TOB1OFF：	定时器输出关闭
TRGB0, TRGB1：	定时器触发输入
TXDA0 至 TXDA2,	
TXDB：	发送数据
V_{DD0}, V_{DD1}：	供电电源
V_{SS0}, V_{SS1}：	地
X1, X2：	时钟振荡器引脚

1.2.5　功能框图（V850E/IF3）

（1）内部框图



注　**1.**　μPD70F3451：　128 KB（flash 存储器）
　　　　μPD70F3452：　256 KB（flash 存储器）
　　2.　μPD70F3451：　8 KB
　　　　μPD70F3452：　12 KB

（2）内部单元

（a）CPU

CPU 使用五级流水线控制，单时钟周期内可以完成地址运算、算术逻辑运算、数据传输以及几乎所有其它指令的处理。

其它片内专有硬件，诸如乘法器（32 位×32 位→64 位）和桶式移位器（32 位），加速处理复杂操作。

（b）总线控制单元（BCU）

BCU 控制内部总线。控制单元根据从 CPU 获取的物理地址，启动所需的外部总线周期。当从外部存储空间取到指令且 CPU 不要求启动总线周期时，BCU 将产生预取地址和预取指令代码。预取指令代码存储在指令队列中。

（i） DMA 控制器（DMAC）

该控制器控制数据传输，传输发生在片上外围 I/O 和内部 RAM、或片上外围 I/O 和片上外围 I/O 之间，而和 CPU 无关。

传输类型是两周期传输，在传输模式下使用单次传输、单步传输、和块传输。

（c）ROM

这是一个 flash 存储器，其映射地址从 00000000H 开始。

取指令期间，CPU 可在一个时钟周期内完成对 ROM/flash 存储器的访问。内部 ROM 容量和区域会根据产品而不同。

部件编号	内部 ROM 容量	内部 ROM 区域
μPD70F3451	128 KB（flash 存储器）	x0000000H 至 x001FFFFH
μPD70F3452	256 KB（flash 存储器）	x0000000H 至 x003FFFFH

（d）RAM

内部 RAM 容量和区域根据产品各有不同。

在指令取指或数据访问期间，CPU 可在一个时钟周期内完成数据的访问。

部件编号	内部 RAM 容量	内部 RAM 区域
μPD70F3451	8 KB	xFFFC000H 至 xFFFDFFFH
μPD70F3452	12 KB	xFFFC000H 至 xFFFEFFFH

（e）中断控制器（INTC）

该控制器用来处理片上外围和外部硬件产生的硬件中断请求（INTP00, INTP01, INTP08 至 INTP18, INTADT0, INTADT1）。可以为这些中断请求指定 8 个级别的中断优先级，同时也能执行多重服务控制。

（f） 时钟发生器（CG）

时钟发生器包括两种基本工作模式：PLL 模式（固定为 8 倍频）和时钟直通模式。它可以产生四种类型的时钟（f_{XX}，$f_{XX}/2$， $f_{XX}/4$，$f_{XX}/8$），并将其中之一提供给 CPU 作为工作时钟（f_{CPU}）。

（g）定时器/计数器

V850E/IF3 提供了 4 个通道的 16 位定时器/事件计数器 M（TMM），5 个通道的 16 位定时器/事件计数器 AA（TAA），2 个通道的 16 位定时器/事件计数器 AB（TAB）和 2 个通道的 16 位间隔定时器 T（TMT），可以测量脉冲间隔宽度或频率，使能一个用于电机控制的变频功能，并输出一个可编程脉冲。

（h）看门狗定时器

集成的看门狗定时器用于检测程序死循环、系统异常等。

溢出发生后，看门狗定时器产生一个不可屏蔽中断请求信号（INTWDT）或内部复位信号（WDTRES）。

（i）串行接口

V850E/IF3 包含 8 路串行接口通道：3 路异步串行接口A（UARTA）通道，1 路异步串行接口B（UARTB）通道，3 路时钟串行接口B（CSIB）通道和 1 路I^2C总线接口（I^2C）通道。其中，UARTA0 和CSIB0 共用一个引脚，UARTA1 和I^2C共用一个引脚，UARTA2 和CSIB1 共用一个引脚，UARTB 和CSIB2 共用一个引脚。

用作 UARTA 时，数据经由 TXDAn 引脚和 RXDAn 引脚进行传输（n = 0 至 2）。

用作 UARTB 时，数据经由 TXDB 引脚和 RXDB 引脚进行传输。

用作 CSIB 时，数据经由 SOBn 引脚、SIBn 引脚和 $\overline{\text{SCKBn}}$ 引脚进行传输（n = 0 至 2）。

用作I^2C时，数据经由SCL和SDA引脚进行传输。

（j）A/D 转换器（ADC）

10 位 A/D 转换器包含 16 个模拟输入引脚。使用逐次逼近的方式进行转换。

为每个高速高分辨率的 12 位 A/D 转换器（ADC0, ADC1）提供一路通道（总共两通道），每个通道各有 5 个模拟输入引脚。为 10 位 A/D 转换器（ADC2）提供一路通道，有 4 个模拟输入引脚。

1 路 ADC0 通道和 3 路 ADC1 通道都包含有一个运算放大器和一个比较器，这些 A/D 转换器可以放大模拟输入电压，并检测过电压输入。

（k）片上调试功能

可以使用支持 MINICUBE2 的片上调试功能，因此，可以搭建一个简单低成本的调试环境。

（l）端口

通用寄存器端口功能和控制引脚功能列举如下：

端口	I/O	复用功能
P0	2 位 I/O	定时器/计数器 I/O、外部中断输入
P1	8 位 I/O	定时器/计数器 I/O、A/D 转换器 0 的外部触发输入、外部中断输入
P2	8 位 I/O	定时器/计数器 I/O、A/D 转换器 1 的外部触发输入、外部中断输入
P3	8 位 I/O	串行接口 I/O、外部中断输入
P4	8 位 I/O	串行接口 I/O、定时器/计数器 I/O、外部中断输入
P7	4 位 I/O	A/D 转换器 2 输入
PDL	10 位 I/O	

1.3 V850E/IG3

1.3.1 特性（V850E/IG3）

○ 最短指令执行周期：
15.6 ns（内部运行于 64 MHz）

○ 通用寄存器： 32 位 × 32 个

○ CPU 特性： 带符号乘法（16 位 × 16 位 → 32 位或 32 位 × 32 位 → 64 位）： 1 至 2 个时钟。
饱和操作指令（包括上溢/下溢出检测功能）。
32 位移位指令： 1 个时钟
位操作指令。
长/短两种格式的数据装载/存储指令。
带符号装载指令

<R> ○ 存储空间（仅限 μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A）：
256 MB 线性地址空间（程序/数据共用）
片选输出功能：2 片空间
存储器块划分功能：2 MB/块

- 外部总线接口： 分离总线模式： 16 位地址总线
8 位/16 位数据总线
复用总线模式： 8 位地址总线
8 位/16 位数据总线
8 位/16 位数据总线宽度功能
外部总线频率切换功能：32/16 MHz
等待功能
• 可编程等待功能。
• 外部等待功能
空闲状态功能
总线保持功能

○ 内部存储器：

部件编号	内部 ROM	内部 RAM
μPD70F3453	128 KB（flash 存储器）	8 KB
μPD70F3454	256 KB（flash 存储器）	12 KB

○ 片上调试功能： 支持 **MINICUBE®2**

○ 中断和异常处理： 不可屏蔽中断： 1 个中断源（外部：无，内部：1）
可屏蔽中断： 95 个中断源（外部：21，内部：74）
软件异常： 32 个中断源
异常陷阱： 2 个中断源

○ DMA 控制器：　4 通道

传输单元：	8 位/16 位
最大传输计数：	65,536（2^{16}）
传输类型：	2-周期
传输模式：	单次/单步/块
传输目标：	片上外围 I/O ↔ 内部 RAM
	片上外围 I/O ↔ 片上外围 I/O
传输请求：	片上外围 I/O/软件

下一地址设置功能

○ I/O 线：　总共：64（输入端口： 8， I/O 端口： 56）

○ 定时器功能：

16 位间隔定时器 M（TMM）：	4 通道
16 位定时器/事件计数器 AA（TAA）：	5 通道
16 位定时器/事件计数器 AB（TAB）：	2 通道
16 位定时器/事件计数器 T（TMT）：	2 通道

电机控制功能（使用定时器TAB：2 通道（TAB0 和 TAB1），
TAA：2 通道（TAA0 和 TAA1））

带死区时间的 16 位精确 6-相 PWM 功能：2 通道
高阻抗输出控制功能
定时器调谐操作产生 A/D 触发的功能
任意的周期设置功能
任意的死区时间设置功能

看门狗定时器：　1 通道

○ 串行接口：

异步串行接口 A（UARTA）
异步串行接口 B（UARTB）
3 线长度可变串行接口 B（CSIB）
I^2C总线接口（I^2C）

UARTA0/CSIB0：	1 通道
UARTA1/I2C：	1 通道
UARTA2/CSIB1：	1 通道
UARTB/CSIB2：	1 通道

○ A/D 转换器：

12 位分辨率 A/D 转换器（A/D 转换器 0 和 1）： 5 通道 + 5 通道
（2 单元）

A/D 转换器 0 的一个通道和 A/D 转换器 1 的三个通道提供一个运算放大器用于输入电平放大，提供一个比较器用于过电压检测。

10 位分辨率 A/D 转换器（A/D 转换器 2）： 8 通道

○ 时钟发生器：

可以连接 4 至 8 MHz 谐振器（禁止外部时钟输入）
通过PLL 时钟合成器的倍频功能（固定为 8 倍频，f_{XX} = 32 至 64 MHz）
CPU时钟分频功能（f_{XX}，$f_{XX}/2$，$f_{XX}/4$，$f_{XX}/8$）

○ 省电功能： HALT/IDLE/STOP 模式

○ 上电清零功能：

○ 低电压检测功能：

○ 封装： 100-针塑封 LQFP（密间距）（14 × 14）
100-针塑封 LQFP（14 × 20）
<R> 161-针塑封 FBGA（10 × 10）

○ 工作供电电压： $V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = EV_{DD2} = AV_{DD0} = AV_{DD1} = AV_{DD2}$ = 4.0 至 5.5 V
（当 A/D 转换器 0, 1 或 2 工作时）
$V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = EV_{DD2} = AV_{DD0} = AV_{DD1} = AV_{DD2}$ = 3.5 至 5.5 V
（当 A/D 转换器 0 至 2 都不工作时）

1.3.2 应用领域（V850E/IG3）

- 消费设备（比如变频空调、洗衣机、干燥机、冰箱等）
- 工业设置（比如电机控制、通用变频器等）

1.2.3 订购信息（V850E/IG3）

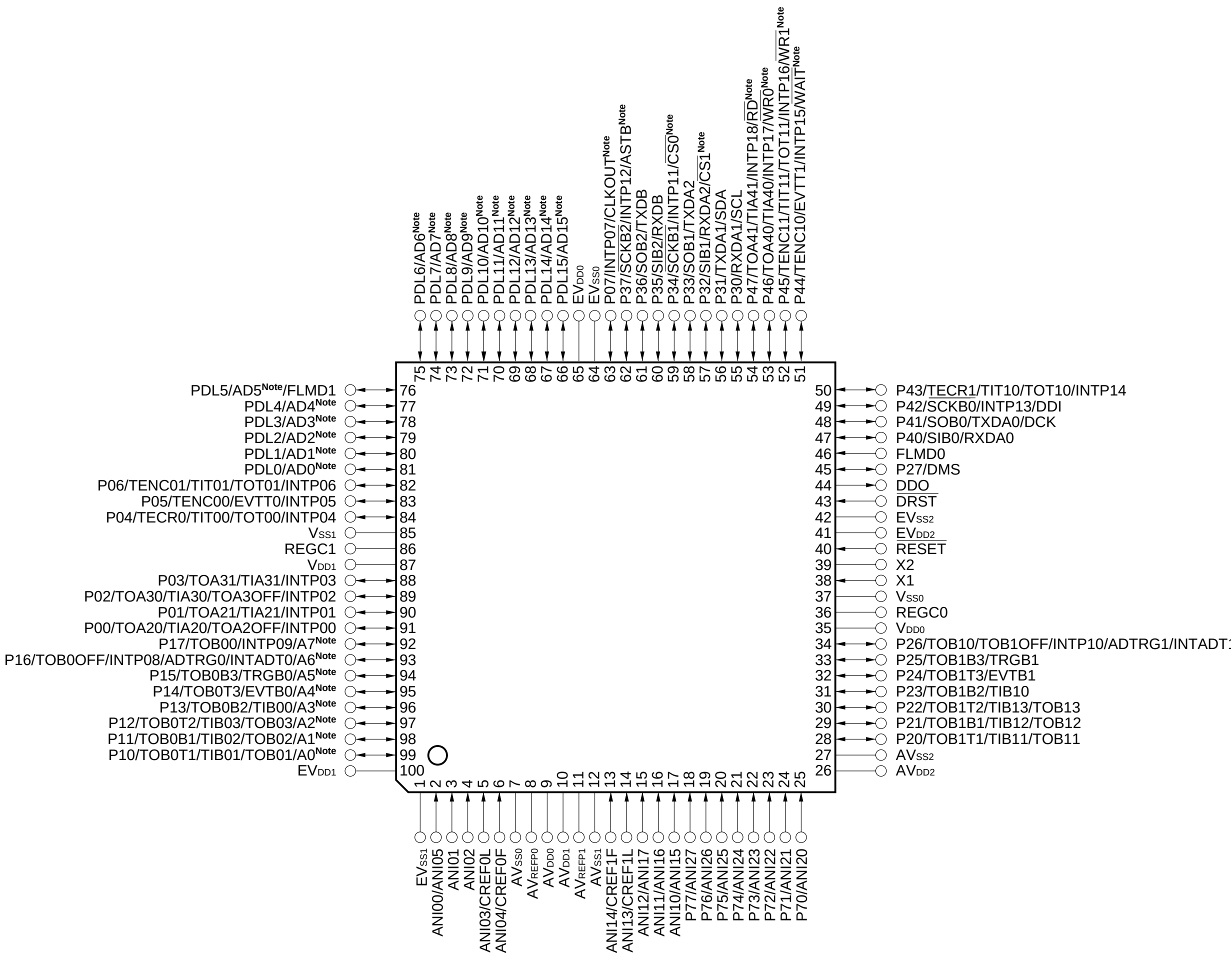
	部件编号	封装	内部 ROM
	μPD70F3453GC-8EA-A	100-针塑封 LQFP（密间距）（14 × 14）	Flash 存储器（128 KB）
	μPD70F3453GF-GAS-AX	100-针塑封 LQFP（14 × 20）	Flash 存储器（128 KB）
	μPD70F3454GC-8EA-A	100-针塑封 LQFP（密间距）（14 × 14）	Flash 存储器（256 KB）
	μPD70F3454GF-GAS-AX	100-针塑封 LQFP（14 × 20）	Flash 存储器（256 KB）
<R>	μPD70F3454F1-DA9-A	161-针塑封 FBGA（10 × 10）	Flash 存储器（256 KB）

备注 V850E/IG3 微控制器是无铅产品。

1.3.4　引脚配置（V850E/IG3）

- 100 针塑封 LQFP（14 × 14）
  μPD70F3453GC-8EA-A
  μPD70F3454GC-8EA-A

俯视图

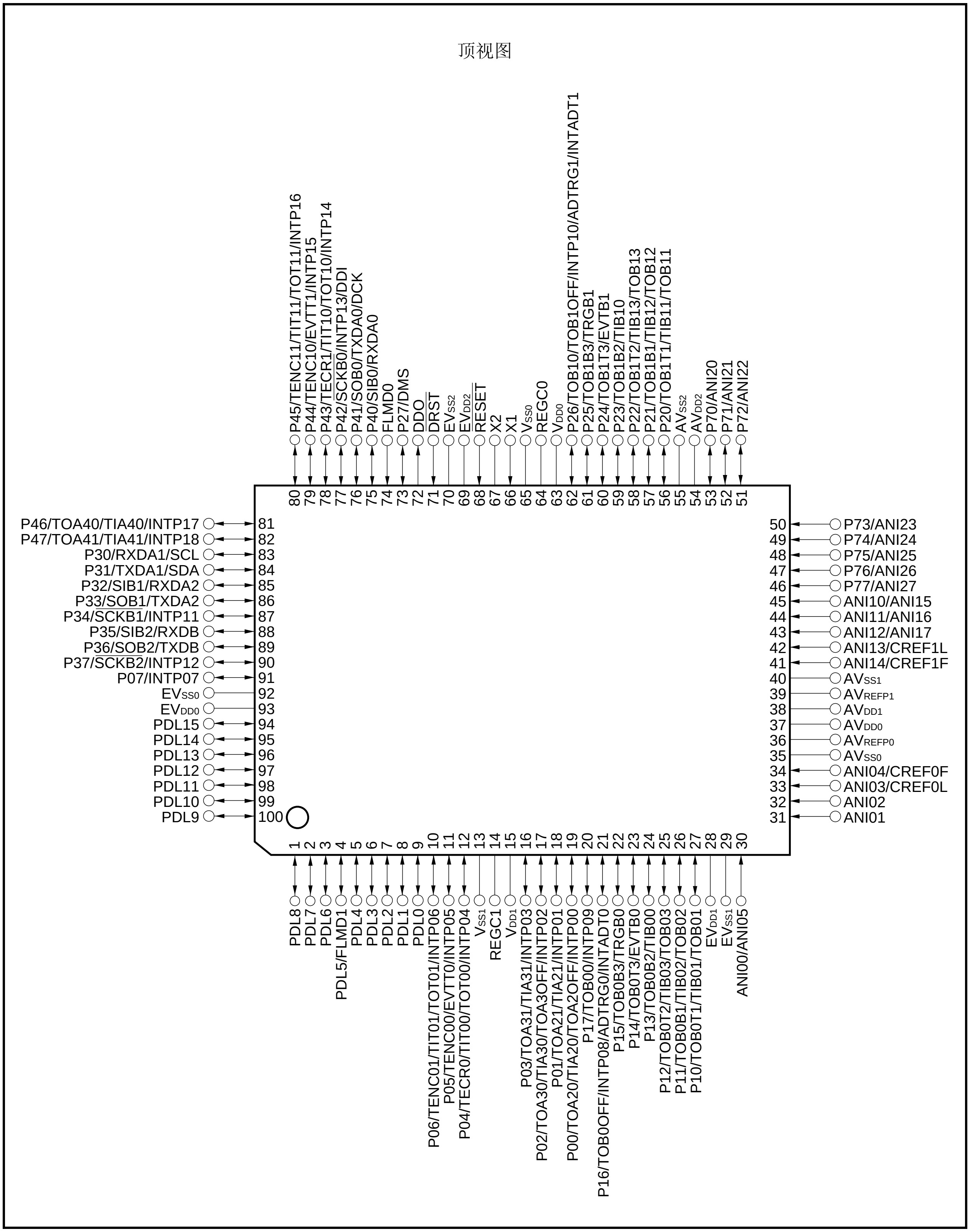


<R>　注　仅在μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A 中可用。

- 100-针塑封 LQFP（14 × 20）

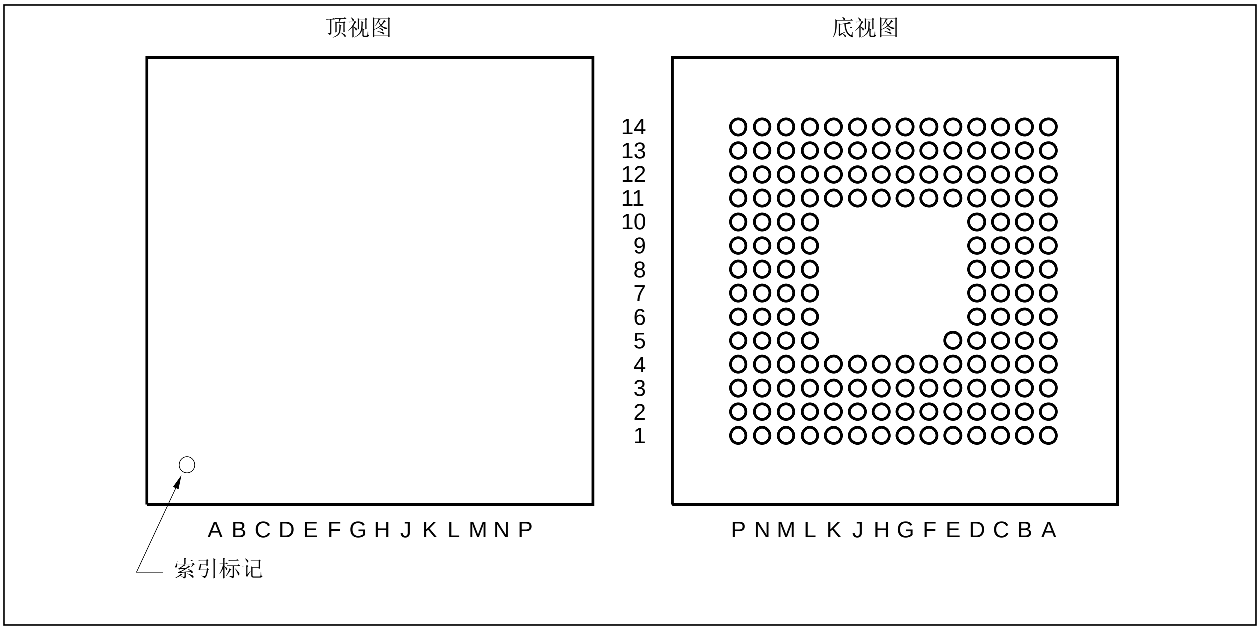
  μPD70F3453GF-GAS-AX

  μPD70F3454GF-GAS-AX



• 161-针塑封 FBGA（10 × 10）

<R>　　μPD70F3454F1-DA9-A



（1/2）

引脚编号	名称	引脚编号	名称	引脚编号	名称
A1	EV_{SS1}	B10	PDL1/AD1	D5	IC1
A2	IC1	B11	PDL3/AD3	D6	IC1
A3	EV_{DD1}	B12	PDL6/AD6	D7	IC1
A4	P12/TOB0T2/TIB03/TOB03/A2	B13	IC1	D8	IC1
A5	P15/TOB0B3/TRGB0/A5	B14	IC1	D9	IC1
A6	P00/TOA20/TIA20/TOA2OFF/INTP00	C1	EV_{SS1}	D10	IC1
A7	V_{DD1}	C2	EV_{SS1}	D11	IC1
A8	REGC1	C3	P11/TOB0B1/TIB02/TOB02/A1	D12	PDL11/AD11
A9	V_{SS1}	C4	P14/TOB0T3/EVTB0/A4	D13	PDL10/AD10
A10	PDL0/AD0	C5	P17/TOB00/INTP09/A7	D14	PDL9/AD9
A11	PDL2/AD2	C6	P02/TOA30/TIA30/TOA3OFF/INTP02	E1	AV_{SS0}
A12	PDL5/AD5/FLMD1	C7	P03/TOA31/TIA31/INTP03	E2	AV_{SS0}
A13	IC1	C8	P04/TECR0/TIT00/TOT00/INTP04	E3	ANI02
A14	EV_{SS0}	C9	P05/TENC00/EVTT0/INTP05	E4	EV_{SS1}
B1	EV_{SS1}	C10	P06/TENC01/TIT01/TOT01/INTP06	E5	IC0
B2	EV_{DD1}	C11	PDL4/AD4	E11	IC1
B3	P10/TOB0T1/TIB01/TOB01/A0	C12	IC1	E12	PDL14/AD14
B4	P13/TOB0B2/TIB00/A3	C13	PDL8/AD8	E13	PDL13/AD13
B5	P16/TOB0OFF/INTP08/ADTRG0/INTADT0/A6	C14	PDL7/AD7	E14	PDL12/AD12
B6	P01/TOA21/TIA21/INTP01	D1	ANI00/ANI05	F1	AV_{DD0}
B7	V_{DD1}	D2	ANI01	F2	AV_{REFP0}
B8	REGC1	D3	EV_{SS1}	F3	ANI03/CREF0L
B9	V_{SS1}	D4	IC1	F4	ANI04/CREF0F

备注　1. 将 IC0 引脚连接到 EV_{SS0}, EV_{SS1} 或 EV_{SS2}

2. 保持 IC1 引脚开路

（2/2）

引脚编号	名称	引脚编号	名称	引脚编号	名称
F11	IC1	L6	IC1	P5	V_{DD0}
F12	PDL15/AD15	L7	IC1	P6	REGC0
F13	EV_{DD0}	L8	IC1	P7	V_{SS0}
F14	EV_{DD0}	L9	IC1	P8	X1
G1	AV_{DD1}	L10	IC1	P9	$\overline{\text{RESET}}$
G2	AV_{REFP1}	L11	IC1	P10	EV_{DD2}
G3	ANI13/CREF1L	L12	P46/TOA40/TIA40/INTP17/$\overline{\text{WR0}}$	P11	EV_{SS2}
G4	ANI14/CREF1F	L13	P45/TENC11/TIT11/TOT11/INTP16/$\overline{\text{WR1}}$	P12	P40/$\overline{\text{SIB0}}$/RXDA0
G11	IC1	L14	P44/TENC10/EVTT1/INTP15/$\overline{\text{WAIT}}$	P13	IC1
G12	P07/INTP07/CLKOUT	M1	AV_{DD2}	P14	EV_{SS2}
G13	EV_{SS0}	M2	AV_{DD2}		
G14	EV_{SS0}	M3	AV_{SS2}		
H1	AV_{SS1}	M4	P24/TOB1T3/EVTB1		
H2	AV_{SS1}	M5	P25/TOB1B3/TRGB1		
H3	ANI12/ANI17	M6	P26/TOB10/TOB1OFF/INTP10/ADTRG1/INTADT1		
H4	IC2	M7	IC1		
H11	IC1	M8	IC1		
H12	P37/$\overline{\text{SCKB2}}$/INTP12/ASTB	M9	$\overline{\text{DRST}}$		
H13	P36/SOB2/TXDB	M10	DDO		
H14	P35/SIB2/RXDB	M11	P27/DMS		
J1	ANI10/ANI15	M12	IC1		
J2	ANI11/ANI16	M13	P43/TECR1/TIT10/TOT10/INTP14		
J3	IC2	M14	P42/$\overline{\text{SCKB0}}$/INTP13/DDI		
J4	P77/ANI27	N1	IC1		
J11	IC1	N2	AV_{SS2}		
J12	P34/$\overline{\text{SCKB1}}$/INTP11/$\overline{\text{CS0}}$	N3	P21/TOB1B1/TIB12/TOB12		
J13	P33/SOB1/TXDA2	N4	P23/TOB1B2/TIB10		
J14	P32/SIB1/RXDA2/$\overline{\text{CS1}}$	N5	V_{DD0}		
K1	P73/ANI23	N6	REGC0		
K2	P74/ANI24	N7	V_{SS0}		
K3	P75/ANI25	N8	X2		
K4	P76/ANI26	N9	FLMD0		
K11	IC1	N10	EV_{DD2}		
K12	P31/TXDA1/SDA	N11	EV_{SS2}		
K13	P30/RXDA1/SCL	N12	P41/SOB0/TXDA0/DCK		
K14	P47/TOA41/TIA41/INTP18/$\overline{\text{RD}}$	N13	IC1		
L1	P70/ANI20	N14	IC1		
L2	P71/ANI21	P1	AV_{SS2}		
L3	P72/ANI22	P2	IC1		
L4	AV_{SS2}	P3	P20/TOB1T1/TIB11/TOB11		
L5	IC1	P4	P22/TOB1T2/TIB13/TOB13		

备注 1. 保持 IC1 引脚开路

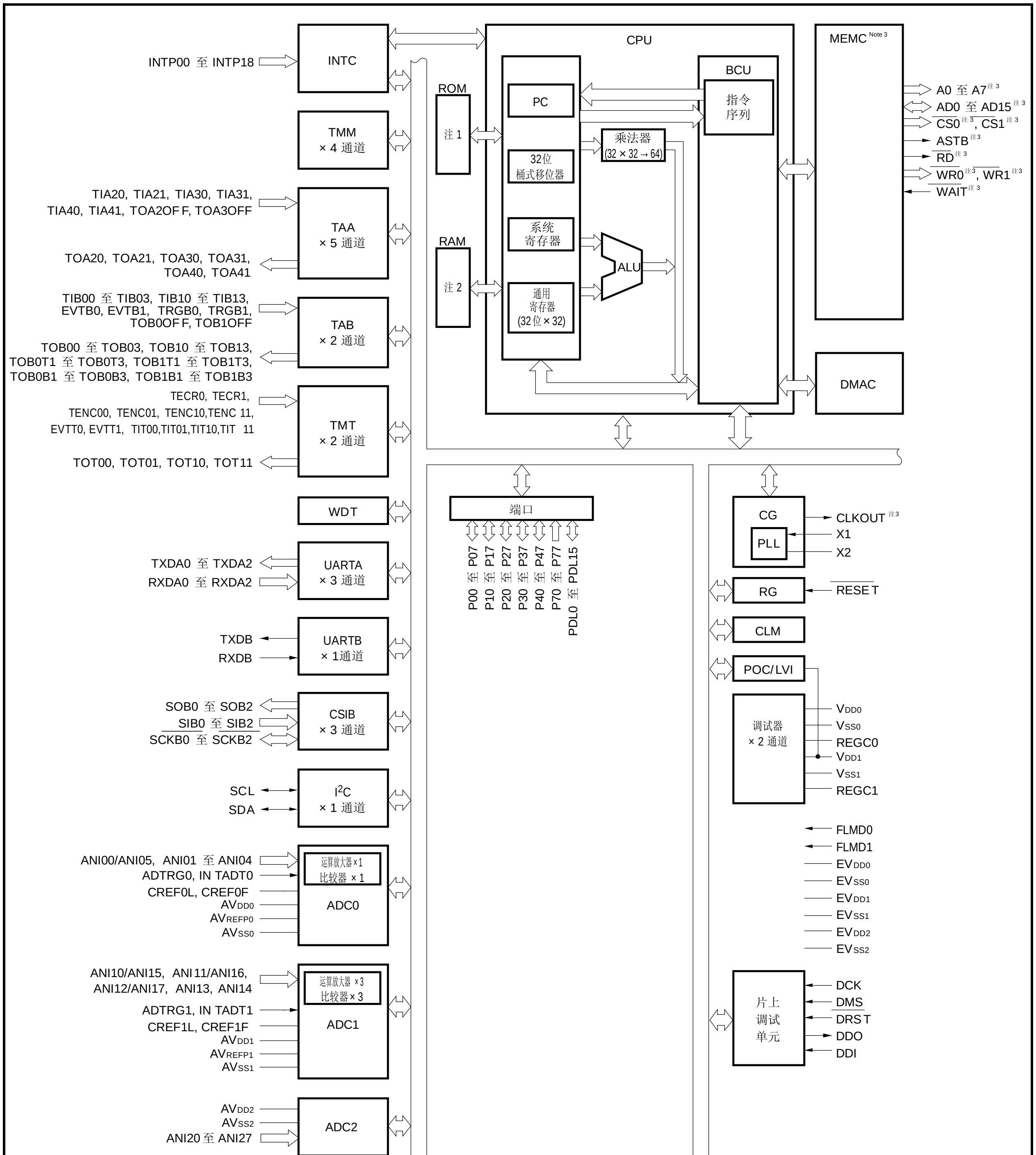
2. 通过一个电阻将 IC2 引脚单独连接到 AV_{SS2}

引脚名称（**V850E/IG3**）

A0 至 A7：	地址总线
AD0 至 AD15：	地址/数据总线
ADTRG0, ADTRG1：	A/D 触发输入
ANI00 至 ANI05,	
ANI10 至 ANI17,	
ANI20 至 ANI27：	模拟 输入
ASTB：	地址选通
AV_{DD0} 至 AV_{DD2}：	模拟供电电源
AV_{REFP0}, AV_{REFP1}：	模拟参考电压
AV_{SS0} 至 AV_{SS2}：	模拟地
CLKOUT：	时钟 输出
CREF0F, CREF1F,	
CREF0L, CREF1L：	比较器参考电压
$\overline{\text{CS0}}$, $\overline{\text{CS1}}$：	片选
DCK：	调试 时钟
DDI：	调试数据输入
DDO：	调试数据输出
DMS：	调试模式选择
$\overline{\text{DRST}}$：	调试复位
EV_{DD0} 至 EV_{DD2}：	端口的供电电源
EV_{SS0} 至 EV_{SS2}：	端口的地
EVTB0, EVTB1,	
EVTT0, EVTT1：	定时器事件计数输入
FLMD0, FLMD1：	Flash 编程模式
<R> IC0 至 IC2：	内部连接
INTADT0, INTADT1,	
INTP00 至 INTP18：	外部中断输入
P00 至 P07：	端口 0
P10 至 P17：	端口 1
P20 至 P27：	端口 2
P30 至 P37：	端口 3
P40 至 P47：	端口 4
P70 至 P77：	端口 7
PDL0 至 PDL15：	端口 DL
$\overline{\text{RD}}$：	读取选通
REGC0, REGC1：	稳压器控制
$\overline{\text{RESET}}$：	复位
RXDA0 至 RXDA2,	
RXDB：	接收数据
$\overline{\text{SCKB0}}$ 至 $\overline{\text{SCKB2}}$：	串行时钟
SCL：	串行时钟
SDA：	串行数据
SIB0 至 SIB2：	串行输入
SOB0 至 SOB2：	串行输出
TECR0, TECR1：	定时器编码器清除输入
TENC00, TENC01,	
TENC10, TENC11：	定时器编码器输入
TIA20, TIA21,	
TIA30, TIA31,	
TIA40, TIA41,	
TIB00 至 TIB03,	
TIB10 至 TIB13,	
TIT00, TIT01,	
TIT10, TIT11：	定时器触发输入
TOA20, TOA21,	
TOA30, TOA31,	
TOA40, TOA41,	
TOB00 至 TOB03,	
TOB0B1 至 TOB0B3,	
TOB0T1 至 TOB0T3,	
TOB10 至 TOB13,	
TOB1B1 至 TOB1B3,	
TOB1T1 至 TOB1T3,	
TOT00, TOT01,	
TOT10, TOT11：	定时器输出
TOA2OFF, TOA3OFF,	
TOB0OFF, TOB1OFF：	定时器输出关闭
TRGB0, TRGB1：	定时器触发输入
TXDA0 至 TXDA2,	
TXDB：	发送数据
V_{DD0}, V_{DD1}：	供电电源
V_{SS0}, V_{SS1}：	地
$\overline{\text{WAIT}}$：	等待
$\overline{\text{WR0}}$, $\overline{\text{WR1}}$：	写入选通
X1, X2：	时钟振荡器引脚

1.3.5 功能框图（V850E/IG3）

（1） 内部框图



注 **1.** μPD70F3453： 128 KB（flash 存储器）

μPD70F3454： 256 KB（flash 存储器）

2. μPD70F3453： 8 KB

μPD70F3454： 12 KB

<R> **3.** 仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

（2） 内部单元

（a） **CPU**

CPU 使用五级流水线控制，单时钟周期内可以完成地址运算、算术逻辑运算、数据传输以及几乎所有其它指令的处理。

其它片内专有硬件，诸如乘法器（32 位×32 位→64 位）和桶式移位器（32 位），加速处理复杂操作。

（b） **总线控制单元（BCU）**

<R> BCU 根据 CPU 获取的物理地址，启动所需的外部总线周期。当从外部存储区域（仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A）取到指令且 CPU 不要求启动总线周期时，BCU 将产生预取地址和预取指令代码。预取指令代码存储在指令队列中。

<R> BCU 控制存储器控制器（MEMC），并执行外部存储器访问（仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A）。

<R> （i） **存储器控制器（MEMC）（仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A）**

控制对 SRAM、外部 ROM 和外部 I/O 的访问。

（ii） **DMA 控制器（DMAC）**

该控制器控制数据传输，传输发生在片上外围 I/O 和内部 RAM、或片上外围 I/O 和片上外围 I/O 之间，而和 CPU 无关。

传输类型是两周期传输，在传输模式下使用单次传输、单步传输、和块传输。

（c） **ROM**

这是一个 flash 存储器，其映射地址从 00000000H 开始。

取指令期间，CPU 可在一个时钟周期内完成对 ROM/flash 存储器的访问。内部 ROM 容量和区域会根据产品而各有不同。

部件编号	内部 ROM 容量	内部 ROM 区域
μPD70F3453	128 KB（flash 存储器）	x0000000H 至 x001FFFFH
μPD70F3454	256 KB（flash 存储器）	x0000000H 至 x003FFFFH

（d） **RAM**

内部 RAM 容量和区域会根据产品而各有不同。

在指令取指或数据访问期间，CPU 可在一个时钟周期内完成数据的访问。

部件编号	内部 RAM 容量	内部 RAM 区域
μPD70F3453	8 KB	xFFFC000H 至 xFFFDFFFH
μPD70F3454	12 KB	xFFFC000H 至 xFFFEFFFH

（e） 中断控制器（INTC）

该控制器用来处理片上外设和外部硬件产生的硬件中断请求（INTP00 至 INTP18, INTADT0, INTADT1）。可以为这些中断请求指定 8 个级别的中断优先级，同时也能执行多重服务控制。

（f） 时钟发生器（CG）

时钟发生器包括两种基本工作模式：PLL 模式（固定为 8 倍频）和时钟直通模式。它可以产生四种类型的时钟（f_{XX}，$f_{XX}/2$， $f_{XX}/4$，$f_{XX}/8$），并将其中之一提供给 CPU 作为工作时钟（f_{CPU}）。

（g） 定时器/计数器

V850E/IG3 提供了 4 个通道的 16 位定时器/事件计数器 M（TMM），5 个通道的 16 位定时器/事件计数器 AA（TAA），2 个通道的 16 位定时器/事件计数器 AB（TAB）和 2 个通道的 16 位间隔定时器 T（TMT），可以测量脉冲间隔宽度或频率，使能一个用于电机控制的变频功能，并输出一个可编程脉冲。

（h） 看门狗定时器（WDT）

集成的看门狗定时器用于检测程序死循环、系统异常等。

溢出发生后，看门狗定时器产生一个不可屏蔽中断请求信号（INTWDT）或内部复位信号（WDTRES）。

（i） 串行接口

V850E/IG3 包含 8 路串行接口通道：3 路异步串行接口A（UARTA）通道，1 路异步串行接口B（UARTB）通道，3 路时钟串行接口B（CSIB）通道和 1 路I^2C总线接口（I^2C）通道。其中，UARTA0 和CSIB0 共用一个引脚，UARTA1 和I^2C共用一个引脚，UARTA2 和CSIB1 共用一个引脚，UARTB 和CSIB2 共用一个引脚。

用作 UARTA 时，数据经由 TXDAn 引脚和 RXDAn 引脚进行传输（n = 0 至 2）。

用作 UARTB 时，数据经由 TXDB 引脚和 RXDB 引脚进行传输。

用作 CSIB 时，数据经由 SOBn 引脚、SIBn 引脚和 $\overline{\text{SCKBn}}$ 引脚进行传输（n = 0 至 2）。

用作I^2C时，数据经由SCL和SDA引脚进行传输。

（j） A/D 转换器（ADC）

10 位 A/D 转换器包含 16 个模拟输入引脚。使用逐次逼近的方式进行转换。

为每个高速高分辨率的 12 位 A/D 转换器（ADC0, ADC1）提供一路通道（总共两通道），每个通道各有 5 个模拟输入引脚。为 10 位 A/D 转换器（ADC2）提供一路通道，有 4 个模拟输入引脚。

1 路 ADC0 通道和 3 路 ADC1 通道都包含有一个运算放大器和一个比较器，这些 A/D 转换器可以放大模拟输入电压，并检测过电压输入。

（k） 片上调试功能

可以使用支持 MINICUBE 和 MINICUBE2 的片上调试功能，因此，可以搭建一个简单便宜的调试环境。

（l）　端口

通用寄存器端口功能和控制引脚功能列举如下：

端口	I/O	复用功能
P0	8 位 I/O	定时器/计数器 I/O、外部中断输入、外部总线接口控制信号输出
P1	8 位 I/O	定时器/计数器 I/O、外部总线接口控制信号输出、A/D 转换器 0 的外部触发输入、外部中断输入
P2	8 位 I/O	定时器/计数器 I/O、A/D 转换器 1 的外部触发输入、外部中断输入、调试输入
P3	8 位 I/O	串行接口 I/O、外部总线接口控制信号输出、外部中断输入
P4	8 位 I/O	串行接口 I/O、定时器/计数器 I/O、调试输入、外部中断输入、外部总线接口控制信号 I/O
P7	8 位 I/O	A/D 转换器 2 输入
PDL	16 位 I/O	外部总线接口控制信号 I/O

第二章　引脚功能

2.1　引脚功能列表

V850E/IF3 和 V850E/IG3 的引脚名称和功能列举如下。根据他们的功能，这些引脚可以被分为端口引脚和非端口引脚。

2.1　引脚功能的列表

有两种类型的引脚 I/O 缓冲器电源：AV_{DD2} 和 EV_{DD0}, EV_{DD1}, EV_{DD2}（仅限 V850E/IG3）。这些电源和引脚的关系如下：

表 2-1. 各个引脚的 I/O 缓冲器电源

（a） V850E/IF3

电源	对应引脚
AV_{DD2}	P70 至 P73
EV_{DD0}, EV_{DD1}	P00, P01, P10 至 P17, P20 至 P27, P30 至 P37, P40 至 P47, PDL0 至 PDL9, $\overline{\text{RESET}}$

（b） V850E/IG3

电源	对应引脚
AV_{DD2}	P70 至 P77
EV_{DD0}, EV_{DD1}, EV_{DD2}	P00 至 P07, P10 至 P17, P20 至 P27, P30 至 P37, P40 至 P47, PDL0 至 PDL15, $\overline{\text{RESET}}$, DCK, DDI, DDO, DMS, $\overline{\text{DRST}}$

<R>　**（1）**　端口引脚

（1/3）

引脚名称	引脚编号				I/O	功能	复用功能
	IF3	IG3					
	GC	GC	GF	F1			
P00	70	91	19	A6	I/O	端口 0 V850E/IF3：2 位 I/O 端口 V850E/IG3：8 位 I/O 端口 可以按位使能输入数据读取/输出数据写入。 可按位指定片内上拉电阻。（当引脚是端口模式且为输入模式时，当引脚用作复用功能的输入引脚时，当 TOA21 和 TOA31（仅限 V850E/IG3） 引脚（复用功能的输出引脚） 变为高阻状态时，可以连接片内上拉电阻。）	TOA20/TIA20/TOA2OFF/INTP00
P01	69	90	18	B6			TOA21/TIA21/INTP01
P02[注1]	–	89	17	C6			TOA30[注1]/TIA30[注1]/TOA3OFF[注1]/INTP02[注1]
P03[注1]	–	88	16	C7			TOA31[注1]/TIA31[注1]/INTP03[注1]
P04[注1]	–	84	12	C8			TECR0[注1]/TIT00[注1]/TOT00[注1]/INTP04[注1]
P05[注1]	–	83	11	C9			TENC00[注1]/EVTT0[注1]/INTP05[注1]
P06[注1]	–	82	10	C10			TENC01[注1]/TIT01[注1]/TOT01[注1]/INTP06[注1]
P07[注1]	–	63	91	G12			INTP07[注1]/CLKOUT[注2]
P10	78	99	27	B3	I/O	端口 1 8 位 I/O 端口 可以按位使能输入数据读取/输出数据写入。 可按位指定片内上拉电阻。（当引脚是端口模式且为输入模式时，当引脚用作复用功能的输入引脚时，当 TOB0B1 至 TOB0B3 和 TOB0T1 至 TOB0T3 引脚（复用功能的输出引脚） 变为高阻状态时，可以连接片内上拉电阻。）	TOB0T1/TIB01/TOB01/A0[注2]
P11	77	98	26	C3			TOB0B1/TIB02/TOB02/A1[注2]
P12	76	97	25	A4			TOB0T2/TIB03/TOB03/A2[注2]
P13	75	96	24	B4			TOB0B2/TIB00/A3[注2]
P14	74	95	23	C4			TOB0T3/EVTB0/A4[注2]
P15	73	94	22	A5			TOB0B3/TRGB0/A5[注2]
P16	72	93	21	B5			TOB0OFF/INTP08/ADTRG0/INTADT0/A6[注2]
P17	71	92	20	C5			TOB00/INTP09/A7[注2]

注　**1.** 仅限 V850E/IG3
　　2. 仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

备注　IF3：V850E/IF3
　　IG3：V850E/IG3
　　GC（V850E/IF3）:　80 针塑封 LQFP（14 × 14）
　　GC（V850E/IG3）:　100 针塑封 LQFP（密间距）（14 × 14）
　　GF（V850E/IG3）:　100 针塑封 LQFP（14 × 20）
　　F1（V850E/IG3）:　161 针塑封 FBGA（10 × 10）

（2/3）

引脚名称	引脚编号				I/O	功能	复用功能
	IF3	IG3					
	GC	GC	GF	F1			
P20	23	28	56	P3	I/O	端口 2 8 位 I/O 端口 可以按位使能输入数据读取/输出数据写入。 可按位指定片内上拉电阻。（当引脚是端口模式且为输入模式时，当引脚用作复用功能的输入引脚时，当 TOB1B1 至 TOB1B3 和 TOB1T1 至 TOB1T3 引脚（复用功能的输出引脚） 变为高阻状态时，可以连接片内上拉电阻。）	TOB1T1/TIB11/TOB11
P21	24	29	57	N3			TOB1B1/TIB12/TOB12
P22	25	30	58	P4			TOB1T2/TIB13/TOB13
P23	26	31	59	N4			TOB1B2/TIB10
P24	27	32	60	M4			TOB1T3/EVTB1
P25	28	33	61	M5			TOB1B3/TRGB1
P26	29	34	62	M6			TOB10/TOB1OFF/INTP10/ADTRG1/INTADT1
P27	36	45	73	M11			DMS[注1]
P30	46	55	83	K13	I/O	端口 3 8 位 I/O 端口 可以按位使能输入数据读取/输出数据写入。 可按位指定片内上拉电阻。（当引脚是端口模式且为输入模式时，当引脚用作复用功能（包括从模式下的 SCKB1 和 SCKB2 引脚）的输入引脚时，可以连接片内上拉电阻。） 当使用复用功能时，如果选择 SCL 或 SDA 引脚，可以指定 N 沟道开漏输出。	RXDA1/SCL
P31	47	56	84	K12			TXDA1/SDA
P32	48	57	85	J14			SIB1/RXDA2/$\overline{\text{CS1}}$[注2]
P33	49	58	86	J13			SOB1/TXDA2
P34	50	59	87	J12			$\overline{\text{SCKB1}}$/INTP11/$\overline{\text{CS0}}$[注2]
P35	51	60	88	H14			SIB2/RXDB
P36	52	61	89	H13			SOB2/TXDB
P37	53	62	90	H12			$\overline{\text{SCKB2}}$/INTP12/ASTB[注2]
P40	38	47	75	P12	I/O	端口 4 8 位 I/O 端口 可以按位使能输入数据读取/输出数据写入。 可按位指定片内上拉电阻。（当引脚是端口模式且为输入模式时，当引脚用作复用功能（包括从模式下的 $\overline{\text{SCKB0}}$ 引脚）的输入引脚时，可以连接片内上拉电阻。）	SIB0/RXDA0
P41	39	48	76	N12			SOB0/TXDA0/DCK[注1]
P42	40	49	77	M14			$\overline{\text{SCKB0}}$/INTP13/DDI[注1]
P43	41	50	78	M13			TECR1/TIT10/TOT10/INTP14
P44	42	51	79	L14			TENC10/EVTT1/INTP15/$\overline{\text{WAIT}}$[注2]
P45	43	52	80	L13			TENC11/TIT11/TOT11/INTP16/$\overline{\text{WR1}}$[注2]
P46	44	53	81	L12			TOA40/TIA40/INTP17/$\overline{\text{WR0}}$[注2]
P47	45	54	82	K14			TOA41/TIA41/INTP18/$\overline{\text{RD}}$[注2]

注　**1.** 仅限 V850E/IG3

　　2. 仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

备注　IF3：V850E/IF3

IG3：V850E/IG3

GC（V850E/IF3）：　80 针塑封 LQFP（14 × 14）

GC（V850E/IG3）：　100 针塑封 LQFP（密间距）（14 × 14）

GF（V850E/IG3）：　100 针塑封 LQFP（14 × 20）

F1（V850E/IG3）：　161 针塑封 FBGA（10 × 10）

（3/3）

引脚名称	引脚编号				I/O	功能	复用功能
	IF3	IG3					
	GC	GC	GF	F1			
P70	20	25	53	L1	输入	端口 7 V850E/IF3：4 位 I/O 端口 V850E/IG3：8 位 I/O 端口	ANI20
P71	19	24	52	L2			ANI21
P72	18	23	51	L3			ANI22
P73	17	22	50	K1			ANI23
P74[注 1]	–	21	49	K2			ANI24[注 1]
P75[注 1]	–	20	48	K3			ANI25[注 1]
P76[注 1]	–	19	47	K4			ANI26[注 1]
P77[注 1]	–	18	46	J4			ANI27[注 1]
PDL0	65	81	9	A10	I/O	端口 DL V850E/IF3：10 位 I/O 端口 V850E/IG3：16 位 I/O 端口 可以按位使能输入数据读取/输出数据写入。 可按位指定片内上拉电阻。（当引脚是端口模式且为输入模式时，可以连接片内上拉电阻。）	AD0[注 2]
PDL1	64	80	8	B10			AD1[注 2]
PDL2	63	79	7	A11			AD2[注 2]
PDL3	62	78	6	B11			AD3[注 2]
PDL4	61	77	5	C11			AD4[注 2]
PDL5	60	76	4	A12			AD5[注 2]/FLMD1
PDL6	59	75	3	B12			AD6[注 2]
PDL7	58	74	2	C14			AD7[注 2]
PDL8	57	73	1	C13			AD8[注 2]
PDL9	56	72	100	D14			AD9[注 2]
PDL10[注 1]	–	71	99	D13			AD10[注 2]
PDL11[注 1]	–	70	98	D12			AD11[注 2]
PDL12[注 1]	–	69	97	E14			AD12[注 2]
PDL13[注 1]	–	68	96	E13			AD13[注 2]
PDL14[注 1]	–	67	95	E12			AD14[注 2]
PDL15[注 1]	–	66	94	F12			AD15[注 2]

注　**1.** 仅限 V850E/IG3
　　2. 仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

备注　IF3：V850E/IF3
IG3：V850E/IG3
GC（V850E/IF3）:　80 针塑封 LQFP（14 × 14）
GC（V850E/IG3）:　100 针塑封 LQFP（密间距）（14 × 14）
GF（V850E/IG3）:　100 针塑封 LQFP（14 × 20）
F1（V850E/IG3）:　161 针塑封 FBGA（10 × 10）

<R> （2） 非端口引脚

（1/8）

引脚名称	引脚编号				I/O	功能	复用功能
	IF3	IG3					
	GC	GC	GF	F1			
A0[注1]	–	99	–	B3	输出	外部存储器的 8 位地址总线	P10/TOB0T1/TIB01/TOB01
A1[注1]	–	98	–	C3			P11/TOB0B1/TIB02/TOB02
A2[注1]	–	97	–	A4			P12/TOB0T2/TIB03/TOB03
A3[注1]	–	96	–	B4			P13/TOB0B2/TIB00
A4[注1]	–	95	–	C4			P14/TOB0T3/EVTB0
A5[注1]	–	94	–	A5			P15/TOB0B3/TRGB0
A6[注1]	–	93	–	B5			P16/TOB0OFF/INTP08/ADTRG0/INTADT0
A7[注1]	–	92	–	C5			P17/TOB00/INTP09
AD0[注1]	–	81	–	A10	I/O	外部存储器的 16 位地址/数据总线	PDL0
AD1[注1]	–	80	–	B10			PDL1
AD2[注1]	–	79	–	A11			PDL2
AD3[注1]	–	78	–	B11			PDL3
AD4[注1]	–	77	–	C11			PDL4
AD5[注1]	–	76	–	A12			FLMD1/PDL5
AD6[注1]	–	75	–	B12			PDL6
AD7[注1]	–	74	–	C14			PDL7
AD8[注1]	–	73	–	C13			PDL8
AD9[注1]	–	72	–	D14			PDL9
AD10[注1]	–	71	–	D13			PDL10[注2]
AD11[注1]	–	70	–	D12			PDL11[注2]
AD12[注1]	–	69	–	E14			PDL12[注2]
AD13[注1]	–	68	–	E13			PDL13[注2]
AD14[注1]	–	67	–	E12			PDL14[注2]
AD15[注1]	–	66	–	F12			PDL15[注2]
ADTRG0	72	93	21	B5	输入	A/D 转换器 0 的外部触发输入	P16/TOB0OFF/INTP08/INTADT0/A6[注1]
ADTRG1	29	34	62	M6	输入	A/D 转换器 1 的外部触发输入	P26/TOB10/TOB1OFF/INTP10/INTADT1

注 **1.** 仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

2. 仅限 V850E/IG3

备注 IF3：V850E/IF3

IG3：V850E/IG3

GC（V850E/IF3）： 80 针塑封 LQFP（14 × 14）

GC（V850E/IG3）： 100 针塑封 LQFP（密间距）（14 × 14）

GF（V850E/IG3）： 100 针塑封 LQFP（14 × 20）

F1（V850E/IG3）： 161 针塑封 FBGA（10 × 10）

（2/8）

引脚名称	引脚编号				I/O	功能	复用功能
	IF3	IG3					
	GC	GC	GF	F1			
ANI00	1	2	30	D1	输入	A/D 转换器 0 的模拟输入	ANI05
ANI01	2	3	31	D2			–
ANI02	3	4	32	E3			–
ANI03	4	5	33	F3			CREF0L
ANI04	5	6	34	F4			CREF0F
ANI05	1	2	30	D1			ANI00
ANI10	16	17	45	J1	输入	A/D 转换器 1 的模拟输入	ANI15
ANI11	15	16	44	J2			ANI16
ANI12	14	15	43	H3			ANI17
ANI13	13	14	42	G3			CREF1L
ANI14	12	13	41	G4			CREF1F
ANI15	16	17	45	J1			ANI10
ANI16	15	16	44	J2			ANI11
ANI17	14	15	43	H3			ANI12
ANI20	20	25	53	L1	输入	A/D 转换器 2 的模拟输入	P70
ANI21	19	24	52	L2			P71
ANI22	18	23	51	L3			P72
ANI23	17	22	50	K1			P73
ANI24[注 1]	–	21	49	K2			P74[注 1]
ANI25[注 1]	–	20	48	K3			P75[注 1]
ANI26[注 1]	–	19	47	K4			P76[注 1]
ANI27[注 1]	–	18	46	J4			P77[注 1]
ASTB[注 2]	–	62	–	H12	输出	外部数据总线的地址选通输出	P37/$\overline{\text{SCKB2}}$/INTP12
AV_{DD0}	8	9	37	F1	–	用于 A/D 转换器 0 的正极电源供应	–
AV_{DD1}	9	10	38	G1	–	用于 A/D 转换器 1 的正极电源供应	–
AV_{DD2}	21	26	54	M1, M2	–	用于 A/D 转换器 2 的正极电源供应	–
AV_{REFP0}	7	8	36	F2	–	用于 A/D 转换器 0 的参考电压输入	–
AV_{REFP1}	10	11	39	G2	–	用于 A/D 转换器 1 的参考电压输入	–

注　**1.** 仅限 V850E/IG3
　　2. 仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

备注　IF3：V850E/IF3
IG3：V850E/IG3
GC（V850E/IF3）:　80 针塑封 LQFP（14 × 14）
GC（V850E/IG3）:　100 针塑封 LQFP（密间距）（14 × 14）
GF（V850E/IG3）:　100 针塑封 LQFP（14 × 20）
F1（V850E/IG3）:　161 针塑封 FBGA（10 × 10）

（3/8）

引脚名称	引脚编号				I/O	功能	复用功能
	IF3	IG3					
	GC	GC	GF	F1			
AV_{SS0}	6	7	35	E1, E2	–	用于 A/D 转换器 0 的地电位	–
AV_{SS1}	11	12	40	H1, H2	–	用于 A/D 转换器 1 的地电位	–
AV_{SS2}	22	27	55	**注 1**	–	用于 A/D 转换器 2 的地电位	–
CLKOUT[注2]	–	63	–	G12	输出	外部总线时钟输出	P07[注3]INTP07[注3]
CREF0L	4	5	33	F3	–	用于 A/D 转换器 0 的小范围比较器参考电压	ANI03
CREF1L	13	14	42	G3	–	用于 A/D 转换器 1 的小范围比较器参考电压	ANI13
CREF0F	5	6	34	F4	–	用于 A/D 转换器 0 的全范围比较器参考电压	ANI04
CREF1F	12	13	41	G4	–	用于 A/D 转换器 1 的全范围比较器参考电压	ANI14
$\overline{\text{CS0}}$[注2]	–	59	–	J12	输出	片选输出	P34/$\overline{\text{SCKB1}}$/INTP11
$\overline{\text{CS1}}$[注2]	–	57	–	J14			P32/SIB1/RXDA2
DCK[注3]	–	48	76	N12	输入	片上调试仿真器的调试时钟	P41/SOB0/TXDA0
DDI[注3]	–	49	77	M14	输入	片上调试仿真器的调试数据输入	P42/$\overline{\text{SCKB0}}$/INTP13
DDO[注3]	–	44	72	M10	输出	片上调试仿真器的调试数据输出	–
DMS[注3]	–	45	73	M11	输入	片上调试仿真器的调试模式选择	P27
$\overline{\text{DRST}}$[注3]	–	43	71	M9	输入	片上调试仿真器的调试复位输入	–
EV_{DD0}	55	65	93	F13, F14	–	用于外部引脚的正极电源供应	–
EV_{DD1}	80	100	28	A3, B2			–
EV_{DD2}[注3]	–	41	69	N10, P10			–
EV_{SS0}	54	64	92	注 4	–	用于外部引脚的地电位	–
EV_{SS1}	79	1	29	注 5			–
EV_{SS2}[注3]	–	42	70	注 6			–
EVTB0	74	95	23	C4	输入	TAB0, TAB1 的外部事件计数输入	A4[注1]/P14/TOB0T3
EVTB1	27	32	60	M4			P24/TOB1T3

注　**1.** L4, M3, N2, P1

2. 仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

3. 仅限 V850E/IG3

4. A14, G13, G14

5. A1, B1, C1, C2, D3, E4

6. N11, P11, P14

备注　IF3：V850E/IF3

IG3：V850E/IG3

GC（V850E/IF3）:　80 针塑封 LQFP（14 × 14）

GC（V850E/IG3）:　100 针塑封 LQFP（密间距）（14 × 14）

GF（V850E/IG3）:　100 针塑封 LQFP（14 × 20）

F1（V850E/IG3）:　161 针塑封 FBGA（10 × 10）

（4/8）

引脚名称	引脚编号				I/O	功能	复用功能
	IF3	IG3					
	GC	GC	GF	F1			
EVTT0[注1]	–	83	11	C9	输入	TMT0, TMT1 的外部事件计数输入/外部触发输入	P05[注1]/TENC00[注1]/INTP05[注1]
EVTT1	42	51	79	L14			P44/TENC10/INTP15/$\overline{WAIT}$[注2]
FLMD0	37	46	74	N9	输入	用于设置 flash 存储器编程模式的引脚	–
FLMD1	60	76	4	A12			PDL5/AD5[注2]
IC0[注3]	–	–	–	E5	–	内部连接的引脚	–
IC1[注3]	–	–	–	注 4	–		–
IC1[注3]	–	–	–	注 5	–		–
INTADT0	72	93	21	B5	输入	外部可屏蔽中断请求输入	P16/TOB0OFF/INTP08/ADTRG0/A6[注2]
INTADT1	29	34	62	M6			P26/TOB10/TOB1OFF/INTP10/ADTRG1
INTP00	70	91	19	A6			P00/TOA20/TIA20/TOA2OFF
INTP01	69	90	18	B6			P01/TOA21/TIA21
INTP02[注1]	–	89	17	C6			P02[注1]/TOA30[注1]/TIA30[注1]/TOA3OFF[注1]
INTP03[注1]	–	88	16	C7			P03[注1]/TOA31[注1]/TIA31[注1]
INTP04[注1]	–	84	12	C8			P04[注1]/TECR0[注1]/TIT00[注1]/TOT00[注1]
INTP05[注1]	–	83	11	C9			P05[注1]/TENC00[注1]/EVTT0[注1]
INTP06[注1]	–	82	10	C10			P06[注1]/TENC01[注1]/TIT01[注1]/TOT01[注1]
INTP07[注1]	–	63	91	G12			P07[注1]/CLKOUT[注2]
INTP08	72	93	21	B5			P16/TOB0OFF/ADTRG0/INTADT0/A6[注2]
INTP09	71	92	20	C5			P17/TOB00/A7[注2]
INTP10	29	34	62	M6			P26/TOB10/TOB1OFF/ADTRG1/INTADT1
INTP11	50	59	87	J12			P34/$\overline{SCKB1}$/$\overline{CS0}$[注2]
INTP12	53	62	90	H12			P37/$\overline{SCKB2}$/ASTB[注2]
INTP13	40	49	77	M14			P42/$\overline{SCKB0}$/DDI[注1]
INTP14	41	50	78	M13			P43/TECR1/TIT10/TOT10
INTP15	42	51	79	L14			P44/TENC10/EVTT1/$\overline{WAIT}$[注2]
INTP16	43	52	80	L13			P45/TENC11/TIT11/TOT11/$\overline{WR1}$[注2]
INTP17	44	53	81	L12			P46/TOA40/TIA40/$\overline{WR0}$[注2]
INTP18	45	54	82	K14			P47/TOA41/TIA41/$\overline{RD}$[注2]

注 **1.** 仅限 V850E/IG3
2. 仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A
3. 仅限μPD70F3454F1-DA9-A
4. A2, A13, B13, B14, C12, D4 至 D11, E11, F11, G11, H11, J11, K11, L5 注 L11, M7, M8, M12, N1, N13, N14, P2, P13
5. H4, J3

备注 IF3：V850E/IF3
IG3：V850E/IG3
GC（V850E/IF3）: 80 针塑封 LQFP（14 × 14）
GC（V850E/IG3）: 100 针塑封 LQFP（密间距）（14 × 14）
GF（V850E/IG3）: 100 针塑封 LQFP（14 × 20）
F1（V850E/IG3）: 161 针塑封 FBGA（10 × 10）

（5/8）

引脚名称	引脚编号				I/O	功能	复用功能
	IF3	IG3					
	GC	GC	GF	F1			
$\overline{\text{RD}}$[注2]	–	54	–	K14	输出	外部数据总线的读取选通输出	P47/TOA41/TIA41/INTP18
REGC0	31	36	64	N6, P6	–	稳压器输出稳定电容连接	–
REGC1	67	86	14	A8, B8			–
$\overline{\text{RESET}}$	35	40	68	P9	输入	系统复位输入	–
RXDA0	38	47	75	P12	输入	UARTA0 至 UARTA2 的串行接收数据输入	P40/SIB0
RXDA1	46	55	83	K13			P30/SCL
RXDA2	48	57	85	J14			P32/SIB1/$\overline{\text{CS1}}$[注1]
RXDB	51	60	88	H14	输入	UARTB0 的串行接收数据输入	P35/SIB2
$\overline{\text{SCKB0}}$	40	49	77	M14	I/O	CSIB0 至 CSIB2 的串行时钟 I/O	P42/INTP13/DDI[注2]
$\overline{\text{SCKB1}}$	50	59	87	J12			P34/INTP11/$\overline{\text{CS0}}$[注1]
$\overline{\text{SCKB2}}$	53	62	90	H12			P37/INTP12/ASTB[注1]
SCL	46	55	83	K13	I/O	串行时钟 I/O	P30/RXDA1
SDA	47	56	84	K12	I/O	串行发送/接收数据 I/O	P31/TXDA1
SIB0	38	47	75	P12	输入	CSIB0 至 CSIB2 的串行接收数据输入	P40/RXDA0
SIB1	48	57	85	J14			P32/RXDA2/$\overline{\text{CS1}}$[注1]
SIB2	51	60	88	H14			P35/RXDB
SOB0	39	48	76	N12	输出	CSIB0 至 CSIB2 的串行发送数据输出	P41/TXDA0/DCK[注2]
SOB1	49	58	86	J13			P33/TXDA2
SOB2	52	61	89	H13			P36/TXDB
TECR0[注2]	–	84	12	C8	输入	TMT0, TMT1 的编码器清除输入	P04[注2]/TIT00[注2]/TOT00[注2]/INTP04[注2]
TECR1	41	50	78	M13			P43/TIT10/TOT10/INTP14
TENC00[注2]	–	83	11	C9	输入	TMT0, TMT1 的编码器输入	P05[注2]/EVTT0[注2]/INTP05[注2]
TENC01[注2]	–	82	10	C10			P06[注2]/TIT01[注2]/TOT01[注2]/INTP06[注2]
TENC10	42	51	79	L14			P44/EVTT1/INTP15/$\overline{\text{WAIT}}$[注1]
TENC11	43	52	80	L13			P45/TIT11/TOT11/INTP16/$\overline{\text{WR1}}$[注1]
TIA20	70	91	19	A6	输入	TAA2 的外部事件计数输入/外部触发输入/捕获触发输入	P00/TOA20/TOA2OFF/INTP00

注　**1.** 仅限 μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A
　　2. 仅限 V850E/IG3

备注　IF3：V850E/IF3
IG3：V850E/IG3
GC（V850E/IF3）：　80 针塑封 LQFP（14 × 14）
GC（V850E/IG3）：　100 针塑封 LQFP（密间距）（14 × 14）
GF（V850E/IG3）：　100 针塑封 LQFP（14 × 20）
F1（V850E/IG3）：　161 针塑封 FBGA（10 × 10）

（6/8）

引脚名称	引脚编号				I/O	功能	复用功能
	IF3	IG3					
	GC	GC	GF	F1			
TIA21	69	90	18	B6	输入	TAA2 的捕获触发输入	P01/TOA21/INTP01
TIA30[注2]	–	89	17	C6	输入	TAA3 的外部事件计数输入/外部触发输入/捕获触发输入	P02[注2]/TOA30[注2]/TOA3OFF[注2]/INTP02[注2]
TIA31[注2]	–	88	16	C7		TAA3 的捕获触发输入	P03[注2]/TOA31[注2]/INTP03[注2]
TIA40	44	53	81	L12	输入	TAA4 的外部事件计数输入/外部触发输入/捕获触发输入	P46/TOA40/INTP17/$\overline{\text{WR0}}$[注1]
TIA41	45	54	82	K14	输入	TAA4 的捕获触发输入	P47/TOA41/INTP18/$\overline{\text{RD}}$[注1]
TIB00	75	96	24	B4	输入	TAB0, TAB1 的捕获触发输入	P13/TOB0B2/A3[注1]
TIB01	78	99	27	B3			P10/TOB0T1/TOB01/A0[注1]
TIB02	77	98	26	C3			P11/TOB0B1/TOB02/A1[注1]
TIB03	76	97	25	A4			P12/TOB0T2/TOB03/A2[注1]
TIB10	26	31	59	N4			P23/TOB1B2
TIB11	23	28	56	P3			P20/TOB1T1/TOB11
TIB12	24	29	57	N3			P21/TOB1B1/TOB12
TIB13	25	30	58	P4			P22/TOB1T2/TOB13
TIT00[注2]	–	84	12	C8	输入	TIT0 的捕获触发输入	P04[注2]/TECR0[注2]/TOT00[注2]/INTP04[注2]
TIT01[注2]	–	82	10	C10			P06[注2]/TENC01[注2]/TOT01[注2]/INTP06[注2]
TIT10	41	50	78	M13	输入	TIT1 的捕获触发输入	P43/TECR1/TOT10/INTP14
TIT11	43	52	80	L13			P45/TENC11/TOT11/INTP16/$\overline{\text{WR1}}$[注1]
TOA20	70	91	19	A6	输出	TAA2 的定时器输出	P00/TIA20/TOA2OFF/INTP00
TOA21	69	90	18	B6			P01/TIA21/INTP01
TOA2OFF	70	91	19	A6	输入	高阻抗输出控制信号输入	P00/TOA20/TIA20/INTP00
TOA30[注2]	–	89	17	C6	输出	TAA3 的定时器输出	P02[注2]/TIA30[注2]/TOA3OFF[注2]/INTP02[注2]
TOA31[注2]	–	88	16	C7			P03[注2]/TIA31[注2]/INTP03[注2]
TOA3OFF[注2]	–	89	17	C6	输入	高阻抗输出控制信号输入	P02[注2]/TOA30[注2]/TIA30[注2]/INTP02[注2]
TOA40	44	53	81	L12	输出	TAA4 的定时器输出	P46/TIA40/INTP17/$\overline{\text{WR0}}$[注1]
TOA41	45	54	82	K14			P47/TIA41/INTP18/$\overline{\text{RD}}$[注1]

注 **1.** 仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

2. 仅限 V850E/IG3

备注 IF3：V850E/IF3

IG3：V850E/IG3

GC（V850E/IF3）: 80 针塑封 LQFP（14 × 14）

GC（V850E/IG3）: 100 针塑封 LQFP（密间距）（14 × 14）

GF（V850E/IG3）: 100 针塑封 LQFP（14 × 20）

F1（V850E/IG3）: 161 针塑封 FBGA（10 × 10）

（7/8）

引脚名称	引脚编号				I/O	功能	复用功能
	IF3	IG3					
	GC	GC	GF	F1			
TOB00	71	92	20	C5	输出	TAB0 的定时器输出	P17/INTP09/A7注 1
TOB01	78	99	27	B3			P10/TOB0T1/TIB01/A0注 1
TOB02	77	98	26	C3			P11/TOB0B1/TIB02/A1注 1
TOB03	76	97	25	A4			P12/TOB0T2/TIB03/A2注 1
TOB0B1	77	98	26	C3	输出	TAB0 的 6 相 PWM 下桥臂脉冲信号输出	P11/TIB02/TOB02/A1注 1
TOB0B2	75	96	24	B4			P13/TIB00/A3注 1
TOB0B3	73	94	22	A5			P15/TRGB0/A5注 1
TOB0OFF	72	93	21	B5	输入	TAB0 的 6 相 PWM 高阻输出控制信号输入	P16/INTP08/ADTRG0/INTADT0/A6注 1
TOB0T1	78	99	27	B3	输出	TAB0 的 6 相 PWM 上桥臂脉冲信号输出	P10/TIB01/TOB01/A0注 1
TOB0T2	76	97	25	A4			P12/TIB03/TOB03/A2注 1
TOB0T3	74	95	23	C4			P14/EVTB0/A4注 1
TOB10	29	34	62	M6	输出	TAB1 的定时器输出	P26/TOB1OFF/INTP10/ADTRG1/INTADT1
TOB11	23	28	56	P3			P20/TOB1T1/TIB11
TOB12	24	29	57	N3			P21/TOB1B1/TIB12
TOB13	25	30	58	P4			P22/TOB1T2/TIB13
TOB1B1	24	29	57	N3	输出	TAB1 的 6 相 PWM 下桥臂脉冲信号输出	P21/TIB12/TOB12
TOB1B2	26	31	59	N4			P23/TIB10
TOB1B3	28	33	61	M5			P25/TRGB1
TOB1OFF	29	34	62	M6	输入	TAB1 的 6 相 PWM 高阻输出控制信号输入	P26/TOB10/INTP10/ADTRG1/INTADT1
TOB1T1	23	28	56	P3	输出	TAB1 的 6 相 PWM 上桥臂脉冲信号输出	P20/TIB11/TOB11
TOB1T2	25	30	58	P4			P22/TIB13/TOB13
TOB1T3	27	32	60	M4			P24/EVTB1
TOT00注 2	–	84	12	C8	输出	TMT0, TMT1 的定时器输出	P04注 2/TECR0注 2/TIT00注 2/INTP04注 2
TOT01注 2	–	82	10	C10			P06注 2/TENC01注 2/TIT01注 2/INTP06注 2
TOT10	41	50	78	M13			P43/TECR1/TIT10/INTP14
TOT11	43	52	80	L13			P45/TENC11/TIT11/INTP16/$\overline{\text{WR1}}$注 1

注 **1.** 仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A
2. 仅限 V850E/IG3

备注 IF3：V850E/IF3
IG3：V850E/IG3
GC（V850E/IF3）: 80 针塑封 LQFP（14 × 14）
GC（V850E/IG3）: 100 针塑封 LQFP（密间距）（14 × 14）
GF（V850E/IG3）: 100 针塑封 LQFP（14 × 20）
F1（V850E/IG3）: 161 针塑封 FBGA（10 × 10）

（8/8）

引脚名称	引脚编号				I/O	功能	复用功能
	IF3	IG3					
	GC	GC	GF	F1			
TRGB0	73	94	22	A5	输入	TAB0, TAB1 的外部触发输入	P15/TOB0B3/A5注 1
TRGB1	28	33	61	M5			P25/TOB1B3
TXDA0	39	48	76	N12	输出	UARTA0 至 UARTA2 的串行发送数据输出	P41/SOB0/DCK注 2
TXDA1	47	56	84	K12			P31/SDA
TXDA2	49	58	86	J13			P33/SOB1
TXDB	52	61	89	H13	输出	UARTB0 的串行发送数据输出	P36/SOB2
V_{DD0}	30	35	63	N5, P5	–	用于内部单元的正极电源供应	–
V_{DD1}	68	87	15	A7, B7			–
V_{SS0}	32	37	65	N7, P7	–	用于内部单元的地电位	–
V_{SS1}	66	85	13	A9, B9			–
$\overline{\text{WAIT}}$注	–	51	–	L14	输入	外部等待请求输入	P44/TENC10/EVTT1/INTP15
$\overline{\text{WR0}}$注	–	53	–	L12	输出	外部数据总线的写入选通输出	P46/TOA40/TIA40/INTP17
$\overline{\text{WR1}}$注	–	52	–	L13			P45/TENC11/TIT11/TOT11/INTP16
X1	33	38	66	P8	输入	用于系统时钟的谐振器连接引脚	–
X2	34	39	67	N8	–		–

注　仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

备注　IF3：V850E/IF3

IG3：V850E/IG3

GC（V850E/IF3）:　80 针塑封 LQFP（14 × 14）

GC（V850E/IG3）:　100 针塑封 LQFP（密间距）（14 × 14）

GF（V850E/IG3）:　100 针塑封 LQFP（14 × 20）

F1（V850E/IG3）:　161 针塑封 FBGA（10 × 10）

2.2　引脚状态

各种模式下的引脚工作状态描述如下：

表 2-2. 各种工作模式下的引脚工作状态

工作状态 引脚	复位	HALT 模式/ DMA 传输期间	IDLE 模式/ STOP 模式	空闲状态
AD0 至 AD15[注1]（PDL0 至 PDL15）	Hi-Z[注2]	工作中	Hi-Z	保持
A0 至 A7[注1]（P10 至 P17）	Hi-Z[注2]	工作中	Hi-Z	保持
$\overline{\text{CS0}}$[注1], $\overline{\text{CS1}}$[注1]（P34, P32）	Hi-Z[注2]	工作中	H	保持
$\overline{\text{WR0}}$[注1], $\overline{\text{WR1}}$[注1]（P46, P45）	Hi-Z[注2]	工作中	H	H
$\overline{\text{RD}}$[注1]（P47）	Hi-Z[注2]	工作中	H	H
ASTB[注1]（P37）	Hi-Z[注2]	工作中	H	H
$\overline{\text{WAIT}}$[注1]（P44）	Hi-Z[注2]	工作中	–	–
CLKOUT[注1]（P07）	Hi-Z[注2]	工作中	保持	工作中

<R>　注　**1.** 仅限 μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

2. 由于总线控制引脚也可以用作端口引脚，在复位后它被初始化为输入模式（端口模式）。

备注　Hi-Z：高阻

保持：保持上一个外部总线周期的状态。

H：　高电平输出

–：　无采样输入（不响应）

<R>

2.3 引脚 I/O 电路及未使用引脚的推荐连接

当通过电阻连接到 AV_{SS2}，EV_{DD0}，EV_{DD1}，EV_{DD2}（仅限 V850E/IG3） 或 EV_{SS0}，EV_{SS1}，EV_{SS2}（仅限 V850E/IG3）时，推荐使用 1 至 10 kΩ 电阻。

（1/4）

引脚	复用功能	引脚编号				I/O 电路类型	推荐的连接方式
		IF3	IG3				
		GC	GC	GF	F1		
P00	TOA20/TIA20/TOA2OFF/INTP00	70	91	19	A6	5-AH	输入： 通过电阻单独连接至 EV_{DD0}, EV_{DD1}, EV_{DD2}[注1] 或 EV_{SS0}, EV_{SS1}, EV_{SS2}[注1] 输出： 保持开路。
P01	TOA21/TIA21/INTP01	69	90	18	B6		
P02[注1]	TOA30[注1]/TIA30[注1]/TOA3OFF[注1]/ INTP02[注1]	–	89	17	C6		
P03[注1]	TOA31[注1]/TIA31[注1]/INTP03[注1]	–	88	16	C7		
P04[注1]	TECR0[注1]/TIT00[注1]/TOT00[注1]/INTP04[注1]	–	84	12	C8		
P05[注1]	TENC00[注1]/EVTT0[注1]/INTP05[注1]	–	83	11	C9		
P06[注1]	TENC01[注1]/TIT01[注1]/TOT01[注1]/INTP06[注1]	–	82	10	C10		
P07[注1]	INTP07[注1]/CLKOUT[注2]	–	63	91	G12		
P10	TOB0T1/TIB01/TOB01/A0[注2]	78	99	27	B3		
P11	TOB0B1/TIB02/TOB02/A1[注2]	77	98	26	C3		
P12	TOB0T2/TIB03/TOB03/A2[注2]	76	97	25	A4		
P13	TOB0B2/TIB00/A3[注2]	75	96	24	B4		
P14	TOB0T3/EVTB0/A4[注2]	74	95	23	C4		
P15	TOB0B3/TRGB0/A5[注2]	73	94	22	A5		
P16	TOB0OFF/INTP08/ADTRG0/INTADT0/A6[注2]	72	93	21	B5		
P17	TOB00/INTP09/A7[注2]	71	92	20	C5		
P20	TOB1T1/TIB11/TOB11	23	28	56	P3		
P21	TOB1B1/TIB12/TOB12	24	29	57	N3		
P22	TOB1T2/TIB13/TOB13	25	30	58	P4		
P23	TOB1B2/TIB10	26	31	59	N4		
P24	TOB1T3/EVTB1	27	32	60	M4		
P25	TOB1B3/TRGB1	28	33	61	M5		
P26	TOB10/TOB1OFF/INTP10/ADTRG1/INTADT1	29	34	62	M6		
P27	DMS[注1]	36	45	73	M11		

注 **1.** 仅限 V850E/IG3

2. 仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

备注 IF3：V850E/IF3

IG3：V850E/IG3

GC（V850E/IF3）: 80 针塑封 LQFP（14 × 14）

GC（V850E/IG3）: 100 针塑封 LQFP（密间距）（14 × 14）

GF（V850E/IG3）: 100 针塑封 LQFP（14 × 20）

F1（V850E/IG3）: 161 针塑封 FBGA（10 × 10）

（2/4）

引脚	复用功能	引脚编号				I/O 电路类型	推荐的连接方式
		IF3	IG3				
		GC	GC	GF	F1		
P30	RXDA1/SCL	46	55	83	K13	5-AH	输入： 通过电阻单独连接至 EV_{DD0}, EV_{DD1}, EV_{DD2}[注1] 或 EV_{SS0}, EV_{SS1}, EV_{SS2}[注1] 输出： 保持开路。
P31	TXDA1/SDA	47	56	84	K12		
P32	SIB1/RXDA2/$\overline{\text{CS1}}$[注1]	48	57	85	J14		
P33	SOB1/TXDA2	49	58	86	J13	5-AG	
P34	$\overline{\text{SCKB1}}$/INTP11/$\overline{\text{CS0}}$[注1]	50	59	87	J12	5-AH	
P35	SIB2/RXDB	51	60	88	H14		
P36	SOB2/TXDB	52	61	89	H13	5-AG	
P37	$\overline{\text{SCKB2}}$/INTP12/ASTB[注1]	53	62	90	H12	5-AH	
P40	SIB0/RXDA0	38	47	75	P12		
P41	SOB0/TXDA0/DCK[注2]	39	48	76	N12	5-AG	
P42	$\overline{\text{SCKB0}}$/INTP13/DDI[注2]	40	49	77	M14	5-AH	
P43	TECR1/TIT10/TOT10/INTP14	41	50	78	M13		
P44	TENC10/EVTT1/INTP15/$\overline{\text{WAIT}}$[注1]	42	51	79	L14		
P45	TENC11/TIT11/TOT11/INTP16/$\overline{\text{WR1}}$[注1]	43	52	80	L13		
P46	TOA40/TIA40/INTP17/$\overline{\text{WR0}}$[注1]	44	53	81	L12		
P47	TOA41/TIA41/INTP18/$\overline{\text{RD}}$[注1]	45	54	82	K14		
P70	ANI20	20	25	53	L1	11-G	通过电阻单独连接至 AV_{SS2}
P71	ANI21	19	24	52	L2		
P72	ANI22	18	23	51	L3		
P73	ANI23	17	22	50	K1		
P74[注2]	ANI24[注2]	–	21	49	K2		
P75[注2]	ANI25[注2]	–	20	48	K3		
P76[注2]	ANI26[注2]	–	19	47	K4		
P77[注2]	ANI27[注2]	–	18	46	J4		
PDL0	AD0[注1]	65	81	9	A10	5-AG	输入： 通过电阻单独连接至 EV_{DD0}, EV_{DD1}, EV_{DD2}[注1] 或 EV_{SS0}, EV_{SS1}, EV_{SS2}[注1] 输出： 保持开路。
PDL1	AD1[注1]	64	80	8	B10		
PDL2	AD2[注1]	63	79	7	A11		
PDL3	AD3[注1]	62	78	6	B11		

注 **1.** 仅限 μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

2. 仅限 V850E/IG3

备注 IF3：V850E/IF3

IG3：V850E/IG3

GC（V850E/IF3）： 80 针塑封 LQFP（14 × 14）

GC（V850E/IG3）： 100 针塑封 LQFP（密间距）（14 × 14）

GF（V850E/IG3）： 100 针塑封 LQFP（14 × 20）

F1（V850E/IG3）： 161 针塑封 FBGA（10 × 10）

（3/4）

引脚	复用功能	引脚编号				I/O 电路类型	推荐的连接方式
		IF3	IG3				
		GC	GC	GF	F1		
PDL4	AD4[注1]	61	77	5	C11	5-AG	输入： 通过电阻单独连接至 EV_{DD0}, EV_{DD1}, EV_{DD2}[注2] 或 EV_{SS0}, EV_{SS1}, EV_{SS2}[注2] 输出： 保持开路。
PDL5	AD5[注1]/FLMD1	60	76	4	A12		
PDL6	AD6[注1]	59	75	3	B12		
PDL7	AD7[注1]	58	74	2	C14		
PDL8	AD8[注1]	57	73	1	C13		
PDL9	AD9[注1]	56	72	100	D14		
PDL10[注2]	AD10[注1]	–	71	99	D13		
PDL11[注2]	AD11[注1]	–	70	98	D12		
PDL12[注2]	AD12[注1]	–	69	97	E14		
PDL13[注2]	AD13[注1]	–	68	96	E13		
PDL14[注2]	AD14[注1]	–	67	95	E12		
PDL15[注2]	AD15[注1]	–	66	94	F12		
ANI00	ANI05	1	2	30	D1	7-C	连接到 AV_{SS0} 或 AV_{SS1}.
ANI01	–	2	3	31	D2		
ANI02	–	3	4	32	E3		
ANI03	CREF0L	4	5	33	F3		
ANI04	CREF0F	5	6	34	F4		
ANI10	ANI15	16	17	45	J1		
ANI11	ANI16	15	16	44	J2		
ANI12	ANI17	14	15	43	H3		
ANI13	CREF1L	13	14	42	G3		
ANI14	CREF1F	12	13	41	G4		

注 **1.** 仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A
2. 仅限 V850E/IG3

备注 IF3：V850E/IF3
IG3：V850E/IG3
GC（V850E/IF3）: 80 针塑封 LQFP（14 × 14）
GC（V850E/IG3）: 100 针塑封 LQFP（密间距）（14 × 14）
GF（V850E/IG3）: 100 针塑封 LQFP（14 × 20）
F1（V850E/IG3）: 161 针塑封 FBGA（10 × 10）

（4/4）

引脚	复用功能	引脚编号				I/O 电路类型	推荐的连接方式
		IF3	IG3				
		GC	GC	GF	F1		
DDO[注 1]	–	–	44	72	M10	3-C	保持开路（当 DRST 为高电平时输出）.
$\overline{\text{DRST}}$[注 1]	–	–	43	71	M9	2-M	保持开路（片内下拉电阻）
FLMD0	–	37	46	74	N9	2	–
IC0[注 2]	–	–	–	–	E5	–	连接到 EV_{SS0}, EV_{SS1} 或 EV_{SS2}.
IC1[注 2]	–	–	–	–	**注 3**		保持开路
IC2[注 2]	–	–	–	–	**注 4**		通过电阻单独连接至 AV_{SS2}
$\overline{\text{RESET}}$	–	35	40	68	P9	2	当使用上电清零（POC）时，将该引脚上拉

注　**1.** 仅限 V850E/IG3

2. 仅限μPD70F3454F1-DA9-A

3. A2, A13, B13, B14, C12, D4 至 D11, E11, F11, G11, H11, J11, K11, L5 至 L11, M7, M8, M12, N1, N13, N14, P2, P13

4. H4, J3

备注　IF3：V850E/IF3

IG3：V850E/IG3

GC（V850E/IF3）:　80 针塑封 LQFP（14 × 14）

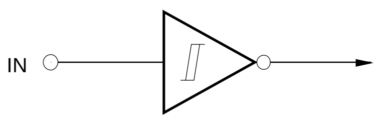
GC（V850E/IG3）:　100 针塑封 LQFP（密间距）（14 × 14）

GF（V850E/IG3）:　100 针塑封 LQFP（14 × 20）

F1（V850E/IG3）:　161 针塑封 FBGA（10 × 10）

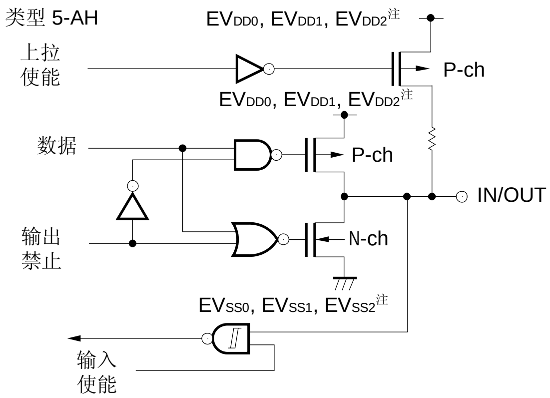
2.4 引脚 I/O 电路

类型 2

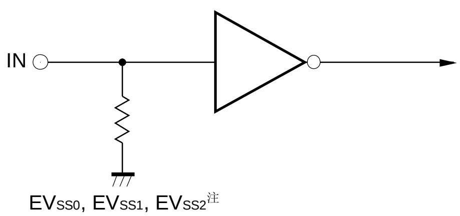


带有滞后特性的施密特触发输入

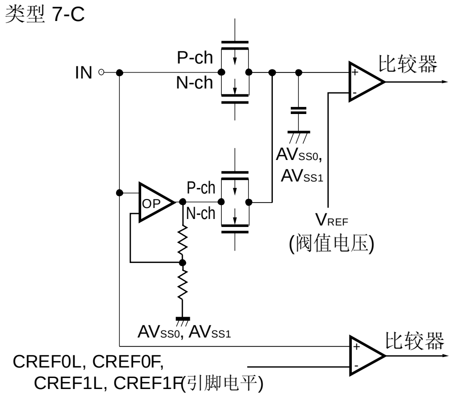
类型 5-AH



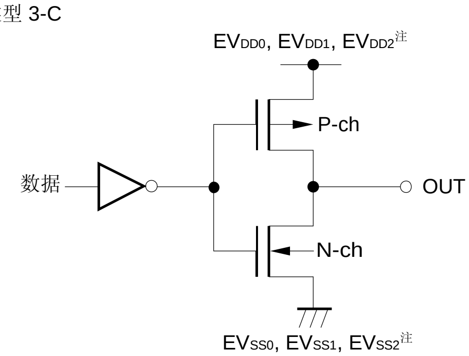
类型 2-M



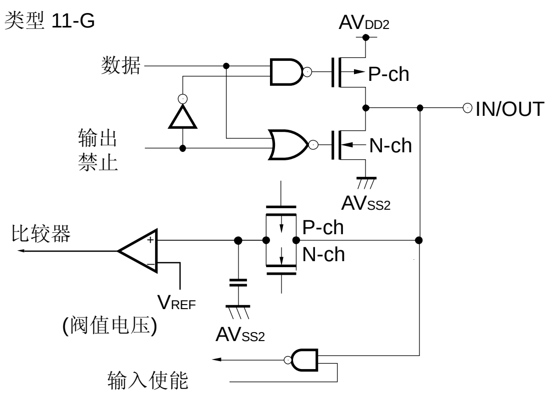
类型 7-C



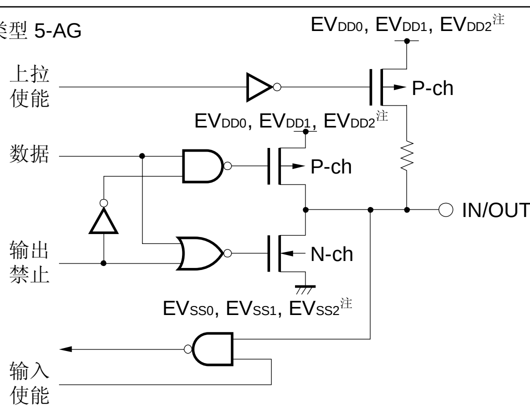
类型 3-C



类型 11-G



类型 5-AG



注　　仅限 V850E/IG3。

第三章 CPU 架构

V850E/IF3 和 V850E/IG3 单片机的 CPU 基于 RISC 架构，且使用 5 级流水线，几乎所有指令的执行都可在一个时钟周期内完成。

3.1 特性

- 最短指令执行周期：　15.6 ns（内部运行于 64 MHz）。
- 通用寄存器：32 位×32 个寄存器
- 内部 32 位架构
- 5 级流水线控制
- 乘法/除法指令
- 饱和运算指令
- 单时钟 32 位移位指令
- 长/短指令格式的数据装载/存储指令
- 4 种类型位操作指令：
  - SET1
  - CLR1
  - NOT1
  - TST1

3.2 CPU 寄存器集

V850E/IF3 和 V850E/IG3 的寄存器可分为两种类型：通用程序寄存器和专用系统寄存器。所有寄存器都是 32 位宽度。

详细信息，请参见 **V850E1 架构用户手册**。

图 3-1. CPU 寄存器集

（1）程序寄存器组

31 … 0

r0	(零寄存器)
r1	(汇编程序保留寄存器)
r2	
r3	(堆栈指针 (SP))
r4	(全局指针 (GP))
r5	(文本指针(TP))
r6	
r7	
r8	
r9	
r10	
r11	
r12	
r13	
r14	
r15	
r16	
r17	
r18	
r19	
r20	
r21	
r22	
r23	
r24	
r25	
r26	
r27	
r28	
r29	
r30	(元素指针 (EP))
r31	(连接指针(LP))

31 … 0

PC	(程序计数器)

（2） 系统寄存器组

31 … 0

EIPC	(中断状态保存寄存器)
EIPSW	(中断状态保存寄存器)

FEPC	(NMI 状态保存寄存器)
FEPSW	(NMI状态保存寄存器)

ECR	(中断源寄存器)

PSW	(程序状态字)

CTPC	(CALLT 执行状态保存寄存器)
CTPSW	(CALLT 执行状态保存寄存器)

DBPC	(异常/调试陷阱状态保存寄存器)
DBPSW	(异常/调试陷阱状态保存寄存器)

CTBP	(CALLT基址指针)

3.2.1 程序寄存器集

程序寄存器集包括通用寄存器和程序计数器。

（1） 通用寄存器（r0 至 r31）

共有 32 个通用寄存器可供使用，r0 至 r31。这些寄存器中的任何一个都可用作数据变量或地址变量。

但是，执行某些指令会隐式使用 r0 和 r30 寄存器，当使用这些寄存器时，一定要加以注意。寄存器 r0 的内容总是保持为 0，用于使用数据 0 的操作或 0 偏移量的寻址。当使用 SLD 和 SST 指令时，使用 r30 寄存器作为基址指针访问存储器。汇编编译器和 C 编译器也隐式使用 r1、r3 至 r5、和 r31。使用这些寄存器之前，必须保存寄存器中的内容以防止丢失，并且在使用这些寄存器之后，将保存的内容恢复到寄存器中。r2 可能会被实时操作系统使用。如果实时操作系统没有使用 r2，那么它可以用作变量寄存器。

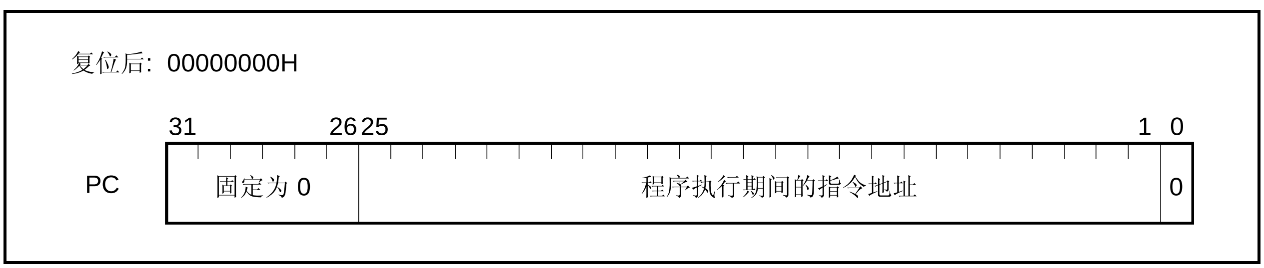
表 3-1. 通用寄存器

名称	用途	操作
r0	零寄存器	始终保持为零。
r1	汇编程序保留寄存器	用作生成 32 位立即数的工作寄存器
r2	地址/数据的变量寄存器（如果实时操作系统没有使用寄存器 r2）	
r3	堆栈指针	用于在函数调用时生成堆栈帧
r4	全局指针	用于访问数据区中的全局变量
r5	文本指针	用作指示文本区域起始地址的寄存器（该区域为程序代码存放区域）
r6 至 r29	地址/数据变量的寄存器	
r30	元素指针	用作访问存储器的基址指针
r31	链接指针	编译器调用函数时使用

（2） 程序计数器 （PC）

该寄存器用来保存程序执行期间的指令地址。该寄存器的低 26 位有效。其中第 31 位至第 26 位固定为 0，如果第 25 位向第 26 位发生进位，也将忽略不计。

第 0 位固定为 0，这意味着不能执行跳转到奇地址处的操作。



3.2.2 系统寄存器集

系统寄存器控制 CPU 的状态，并保存中断信息。

通过使用系统寄存器载入/存储指令（LDSR，STSR 指令），指定下表所列的某个系统寄存器，来实现对系统寄存器的读取/写入操作。

表 3-2. 系统寄存器编号

系统寄存器编号	系统寄存器名称	操作数说明	
		LDSR 指令	STSR 指令
0	中断状态保存寄存器（EIPC）[注 1]	√	√
1	中断状态保存寄存器（EIPSW）[注 1]	√	√
2	NMI状态保存寄存器（FEPC）[注 1]	√	√
3	NMI状态保存寄存器（FEPSW）[注 1]	√	√
4	中断源寄存器（ECR）	×	√
5	程序状态字（PSW）	√	√
6 至 15	将来功能扩展预留寄存器（访问这些寄存器编号的操作无法保证）	×	×
16	CALLT 指令执行状态保存寄存器（CTPC）	√	√
17	CALLT 指令执行状态保存寄存器（CTPSW）	√	√
18	异常/调试陷阱状态保存寄存器（DBPC）	√[注 2]	√[注 2]
19	异常/调试陷阱状态保存寄存器（DBPSW）	√[注 2]	√[注 2]
20	CALLT 指令基址指针（CTBP）	√	√
21 至 31	将来功能扩展预留寄存器（访问这些寄存器编号的操作无法保证）	×	×

注 **1.** 由于只有一组这样中断状态保存寄存器，所以，如果允许多重中断，这些寄存器的内容必须通过编程加以保存。

2. 只有在 DBTRAP 指令或非法操作码的执行和 DBRET 指令的执行间隔期间，才能读取/写入这些寄存器。

注意事项 即使 EIPC 或 FEPC、或 CTPC 的第 0 位由 LDSR 指令设置为 1，中断服务之后，由 RETI 指令返回主程序时也会忽略第 0 位（因为 PC 的第 0 位固定为 0）。设置 EIPC、FEPC 和 CTPC 的值时，使用偶数值（第 0 位 = 0）。

备注 √：允许访问
×：禁止访问

（1） 中断状态保存寄存器（EIPC 和 EIPSW）

有两个中断状态保存寄存器，EIPC 和 EIPSW。

在产生软件异常或者可屏蔽中断时，程序计数器（PC）中的内容被保存到 EIPC，程序状态字（PSW）中的内容保存到 EIPSW（在产生不可屏蔽中断（NMI）时，这些相应寄存器内容被保存到 NMI 状态保存寄存器（FEPC 及 FEPSW）中。

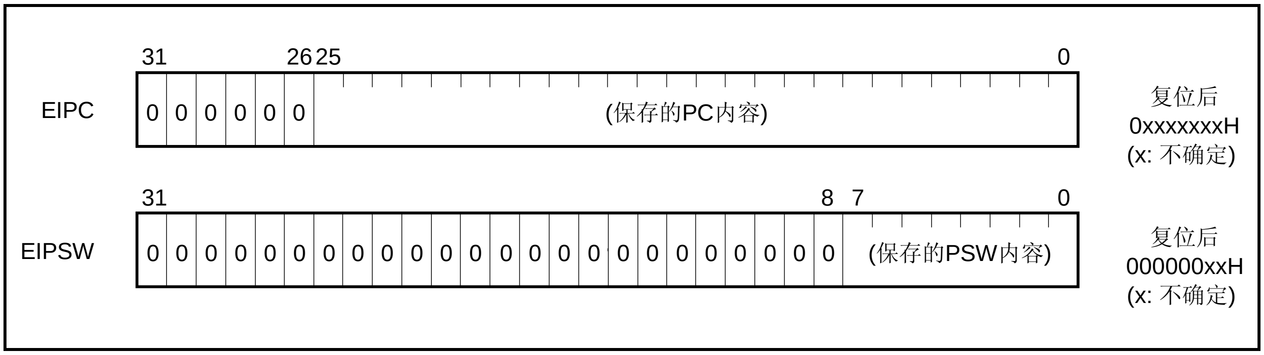
当有软件异常或者是可屏蔽中断产生时，就将当前执行指令的下一条指令地址保存到 EIPC 中，某些指令除外（参见 **20.9 CPU 不响应中断期间**）。

将 PSW 的当前内容被保存到 EIPSW 寄存器中。

由于只有一组中断状态保存寄存器可用，所以，当允许多重中断服务时，这些寄存器的内容必须通过编程加以保存。

EIPC 的第 31 位至第 26 位以及 EIPSW 的第 31 位至第 8 位是保留位，用于将来功能扩展（这些位始终固定为 0）。

执行 RETI 指令时，将寄存器 EIPC 的值和寄存器 EIPSW 的值分别恢复到 PC 和 PSW 中。



（2） NMI 状态保存寄存器（FEPC 和 FEPSW）

有两个 NMI 状态保存寄存器，FEPC 和 FEPSW。

NMI 产生时，程序计数器（PC）中的内容被保存到 FEPC，程序状态字（PSW）中的内容被保存到 FEPSW。

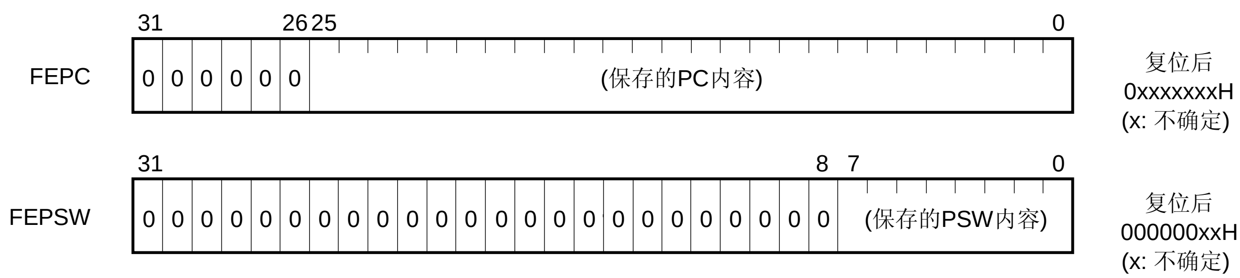
当有 NMI 产生时，就将当前执行指令的下一条指令地址保存到寄存器 FEPC 中，某些指令除外。

<R> 将 PSW 当前内容保存到 FEPSW 寄存器中。

由于只有一组 NMI 状态保存寄存器可用，所以，当允许多重中断服务时，这些寄存器的内容必须通过编程加以保存。

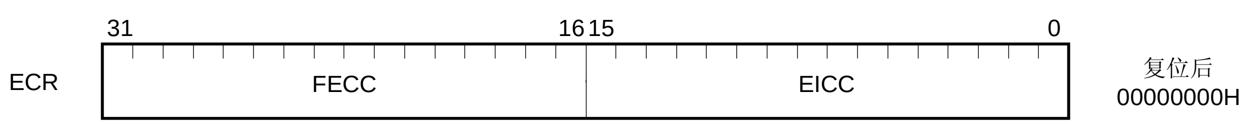
FEPC 的第 31 位至第 26 位以及 FEPSW 的第 31 位至第 8 位是保留位，用于将来功能扩展（固定为 0）。

执行 RETI 指令时，将寄存器 FEPC 的值和寄存器 FEPSW 的值分别恢复到 PC 和 PSW 中。



（3） 中断源寄存器（ECR）

产生中断或异常时，中断源寄存器（ECR）保存异常或中断的源信息。该寄存器保存各中断源的异常代码。该寄存器是只读寄存器，所以，不能通过 LDSR 指令将数据写入该寄存器。



位的位置	位名称	描述
31 至 16	FECC	不可屏蔽中断（NMI）的异常代码
15 至 0	EICC	异常或可屏蔽中断的异常代码

（4） 程序状态字（PSW）

程序状态字（PSW）是一个标志位的集合，指示程序状态（指令执行结果）和 CPU 状态。

如果因为使用 LDSR 指令而更改了该寄存器某位的内容，在 LDSR 指令完成后，新内容就会立即生效。但是，如果通过 LDSR 指令正在执行对 PSW 的写入时，中断请求响应会被挂起。

该寄存器的第 31 位至第 8 位是保留位，用于将来功能扩展（固定为 0）。

（1/2）

	31 … 8	7	6	5	4	3	2	1	0	复位后
PSW	RFU	NP	EP	ID	SAT	CY	OV	S	Z	00000020H

位的位置	标志名称	描述
31 至 8	RFU	保留区域。固定为“0”。
7	NP	表示正在进行不可屏蔽中断（NMI）服务。当响应 NMI 中断请求时，该标志位置 1，同时禁止多重中断。 0：　NMI 服务没有进行。 1：　NMI 服务正在进行。
6	EP	表示正在进行异常处理。产生异常时，该标志位置 1。而且即使该位被置位，也可以响应中断请求。 0：　没有处理异常。 1：　正在处理异常。
5	ID	表示是否允许响应可屏蔽中断请求。 0：　允许中断（EI） 1：　禁止中断（DI）
4	SAT[注]	表示执行饱和运算的结果已经溢出且计算结果已经饱和。由于该位是一个累积标志位，所以，当饱和运算指令结果饱和时，该位置 1，即使是下一个运算结果没有饱和，该位也不会清 0。使用 LDSR 指令将该位清零。执行算术运算指令时，该位既不置 1，也不清 0。 0：　没有饱和 1：　饱和
3	CY	表示运算结果是否有进位或借位产生。 0：　无进位或借位产生 1：　产生进位或借位
2	OV[注]	表示在运算过程中有无溢出。 0：　没有发生溢出。 1：　已经发生溢出
1	S[注]	表示运算结果是否为负数 0：　运算结果为正数或零。 1：　结果为负数。
0	Z	表示运算结果是否为零。 0：　结果不为 0. 1：　结果为 0。

备注　　请阅读下页的**“注”**。

（2/2）

注 在饱和运算期间，已经饱和的运算结果由 OV 标志和 S 标志的内容来决定。在饱和运算期间，只有当 OV 标志位置位（为 1）时，SAT 标志位才置位（为 1）。

运算结果的状态	标志位状态			已经饱和的操作结果
	SAT	OV	S	
超过正值上界	1	1	0	7FFFFFFFH
超过负值下界	1	1	1	80000000H
正值（未超过最大值）	保持运算前的值	0	0	实际的操作结果
负值（未超过最大值）			1	

（5） CALLT 执行状态保存寄存器（CTPC 和 CTPSW）

有两个 CALLT 执行状态保存寄存器，CTPC 和 CTPSW。

执行 CALLT 指令时，程序计数器（PC）的内容被保存到 CTPC，程序状态字（PSW）的内容被保存到 CTPSW。

寄存器 CTPC 中保存的内容是 CALLT 指令之后下一条指令的地址。

将 PSW 当前内容保存到 CTPSW 寄存器中。

CTPC 的第 31 位至第 26 位以及 CTPSW 的第 31 位至第 8 位是保留位，用于将来功能扩展（固定为 0）。

	31					26	25 … 0	
CTPC	0	0	0	0	0	0	(保存的PC内容)	复位后 0xxxxxxxH (x: 不确定)

	31 … 8	7 … 0	
CTPSW	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	(保存的PSW内容)	复位后 000000xxH (x: 不确定)

（6） 异常/调试陷阱状态保存寄存器（DBPC 和 DBPSW）

有两个异常/调试陷阱状态保存寄存器，DBPC 和 DBPSW。

产生异常陷阱或调试陷阱时，程序计数器（PC）的内容保存到 DBPC，程序状态字（PSW）的内容保存到 DBPSW。

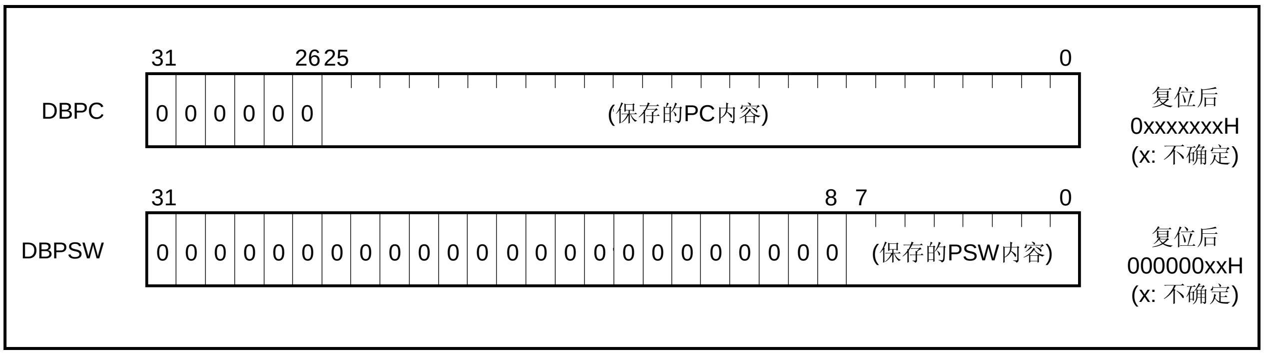
在异常陷阱或者调试陷阱产生时，保存到 DBPC 寄存器中的内容是当前执行指令的下一条指令的地址。

将 PSW 的当前内容保存到 DBPSW 寄存器中。

只有在 DBTRAP 指令或非法操作码的执行和 DBRET 指令的执行间隔期间，才能读取或写入这些寄存器。

DBPC 的第 31 位至第 26 位以及 DBPSW 的第 31 位至第 8 位是保留位，用于将来功能扩展（固定为 0）。

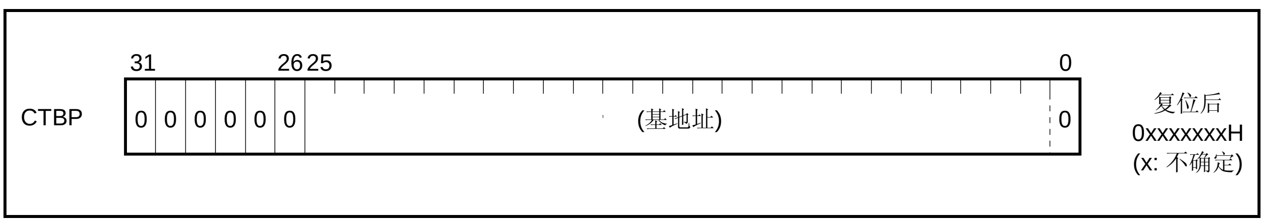
在执行 DBRET 指令之后，将寄存器 DBPC 的值和寄存器 DBPSW 的值分别恢复到 PC 和 PSW 中。



（7） CALLT 指令基址指针（CTBP）

CALLT 基址指针（CTBP）用于指定表地址并生成目标地址（第 0 位固定为 0）。

该寄存器的第 31 位至第 26 位是保留位，用于将来功能扩展（固定为 0）。



3.3　操作模式

3.3.1　操作模式

V850E/IF3 和 V850E/IG3 具有下列几种操作模式。通过 FLMD0 和 FLMD1 引脚进行模式的指定。

（1） 正常工作模式

该模式下，系统复位解除后，程序跳转到内部 ROM 中的复位入口地址处，然后开始执行指令。

（2） Flash 存储器编程模式

如果指定该模式，可以使用 flash 编程器将某段程序写入内部 flash 存储器。

3.3.1　操作模式的规范

根据 FLMD0 引脚和 FLMD1 引脚的状态（输入电平）来指定操作模式。

在正常操作模式下，复位解除后，向 FLMD0 引脚输入低电平。

连接 flash 存储器编程器后，在 flash 存储器编程模式下，flash 编程器会向 FLMD0 引脚输入高电平。但是，在自编程模式下，必须由外部电路输入高电平。

非自编程模式时，在应用系统中固定这些引脚值，在操作期间不要改变。

FLMD1	FLMD0	操作模式	注
×	L	正常工作模式	内部 ROM 区域从地址 000000H 开始分配
L	H	Flash 存储器编程模式	–
H	H	禁止设置	

备注　L：　低电平输入
H：　高电平输入
×：　忽略

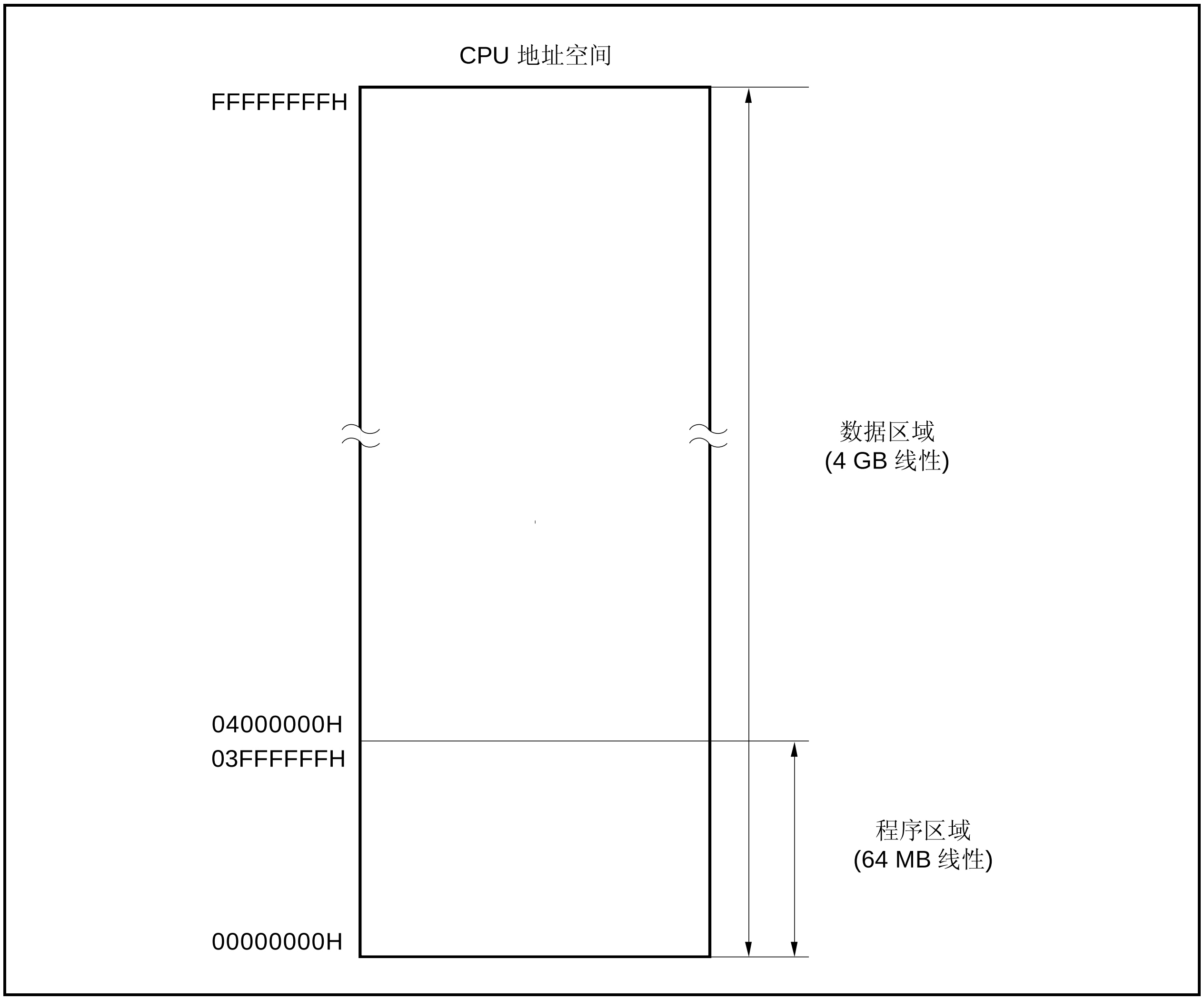
3.4 地址空间

3.4.1 CPU 地址空间

V850E/IF3 和 V850E/IG3 的 CPU 为 32 位架构，操作数寻址（数据访问）支持高达 4 GB 线性地址空间（数据空间）。同样，在指令地址寻址时，最大支持 64 MB 线性地址空间（程序空间）。

图 3-2 显示 CPU 地址空间。

图 3-2. CPU 地址空间

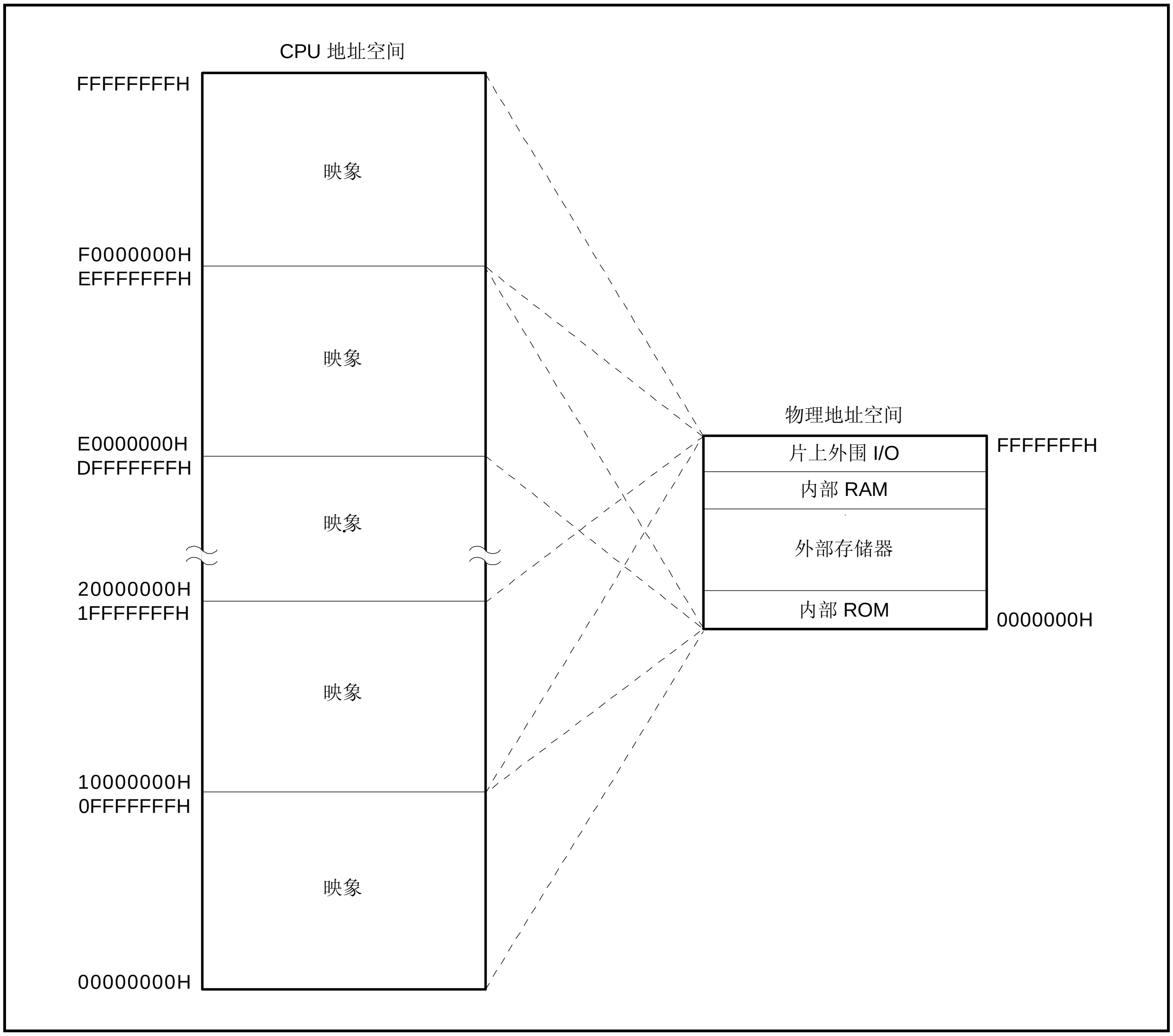


3.4.2 映象

一个 256 MB 物理地址空间被看作 4 GB CPU 地址空间中的 16 个映象。实际上，不论 CPU 地址的第 31 位至第 28 位的值如何，访问的都是相同的 256 MB 物理地址空间。图 3-3 显示虚拟地址空间的映象。

物理地址 x0000000H 可以被看作 CPU 地址 00000000H，并且，还可以被看作地址 10000000H，地址 20000000H, … , 地址 E0000000H 或地址 F0000000H。

图 3-3. 地址空间的映象



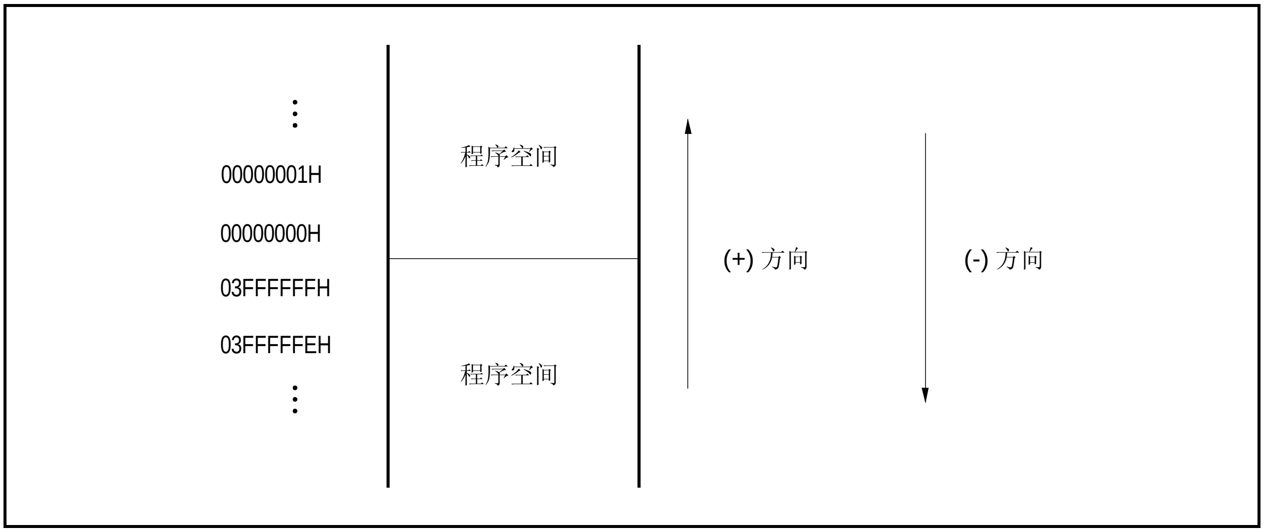
3.4.3　CPU 地址空间的绕回

（1） 程序空间

对于 32 位的 PC（程序计数器），其高 6 位固定为 0，且只有低 26 位是有效的。在计算跳转地址时，即使发生第 25 位至第 26 位的进位或借位，高 6 位会忽略进位或借位。

因此，程序空间的地址上限 03FFFFFFH 和地址下限 00000000H 是相邻的连续地址。像这种程序空间的最高地址和最低地址为相邻的连续地址的情况，称为地址的绕回。

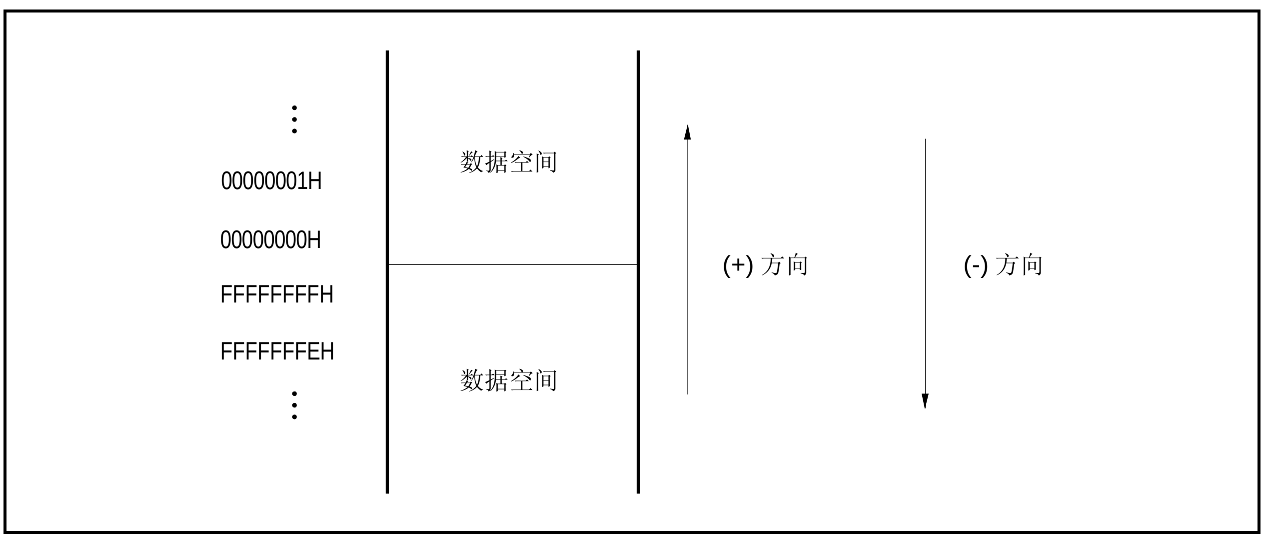
注意事项　由于 03FFF000H 至 03FFFFFFH 的 4 KB 区域可以被看作 0FFFF000H 至 0FFFFFFFH 的一个映象。该区域禁止访问。因此，如果跳转地址的计算结果落入该区域，则不要执行此操作。



（2） 数据空间

操作数地址计算的结果超出 32 位时，该地址将被忽略。

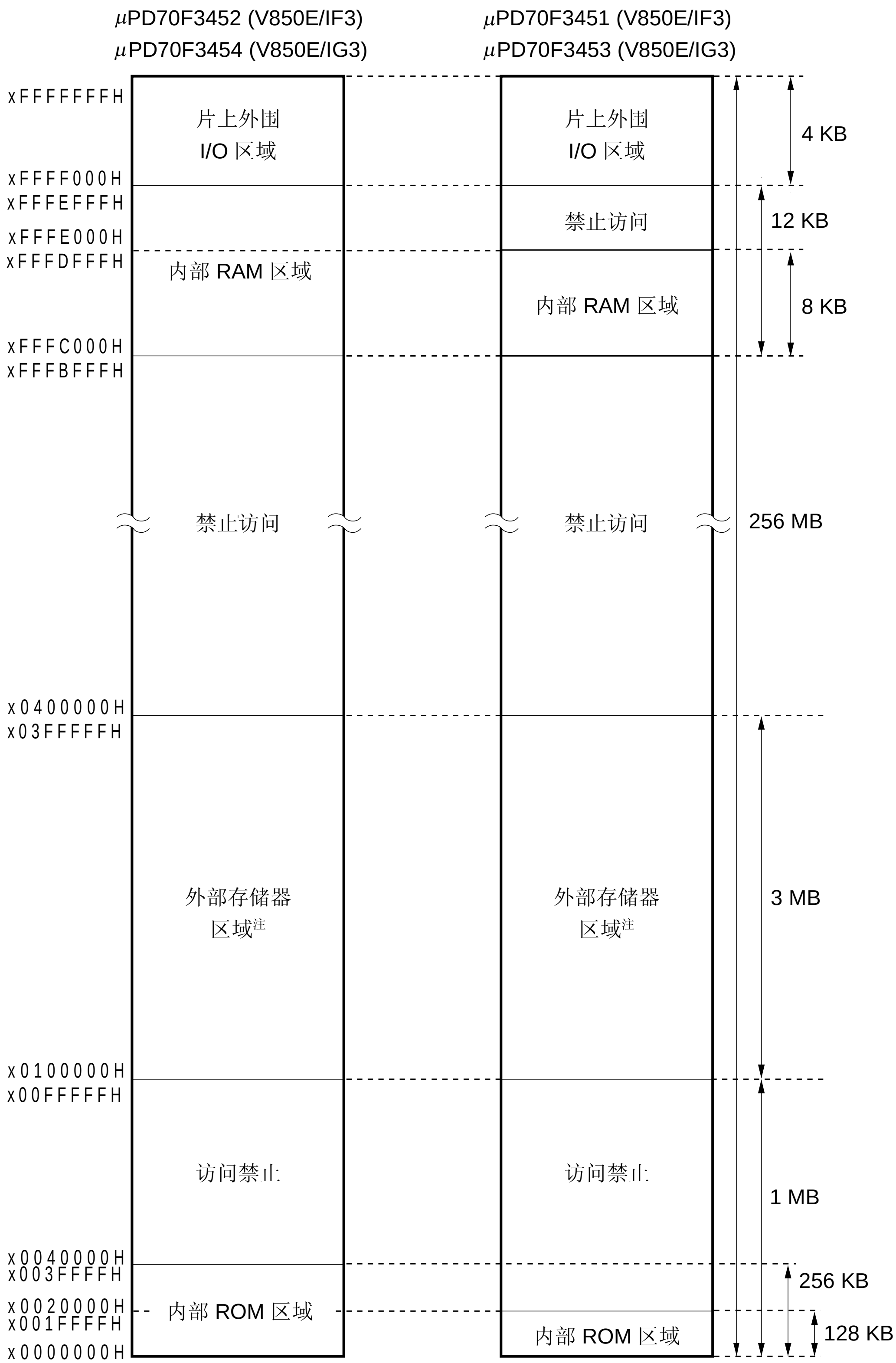
因此，数据空间的上限地址 FFFFFFFFH 和下限地址 00000000H 是相邻的连续地址，在这些地址的边界会发生绕回。



3.4.4　存储器映射

V850E/IF3 和 V850E/IG3 的保留区域如图 3-4 所示。

图 3-4. 数据存储空间映射图（物理地址）



<R>　注　μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A：　外部存储器区域
其它：　禁止访问区域

3.4.5 区域

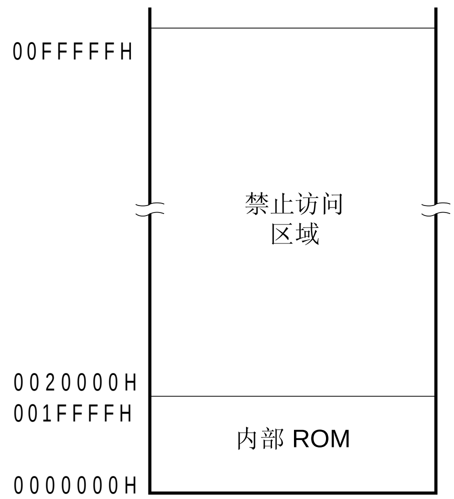
（1） 内部 ROM 区域

地址 00000H 至 FFFFFH 的 1 MB 区域被保留作为内部 ROM 区域。

（a） μPD70F3451 (V850E/IF3), μPD70F3453 (V850E/IG3)

地址 000000H 至 01FFFFH 的 128 KB 区域作为物理内部 ROM。

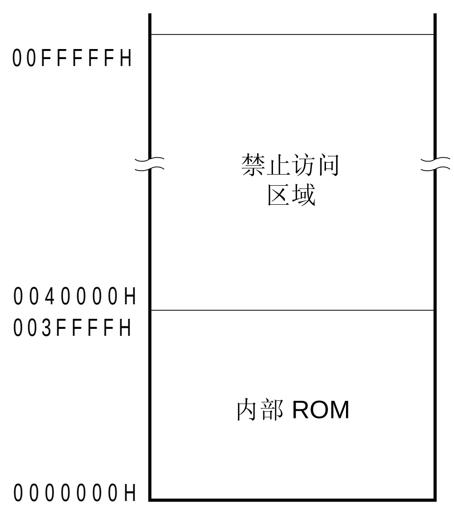
图 3-5. 内部 ROM 区域（128 KB）



（b） PD70F3452 (V850E/IF3), PD70F3454 (V850E/IG3)

地址 000000H 至 03FFFFH 的 256 KB 区域作为物理内部 ROM。

图 3-6. 内部 ROM 区域（256 KB）



（2） 内部 RAM 区域

地址 FFFC000H 至 FFFEFFFH 的 12 KB 区域被保留作为内部 RAM 区域。

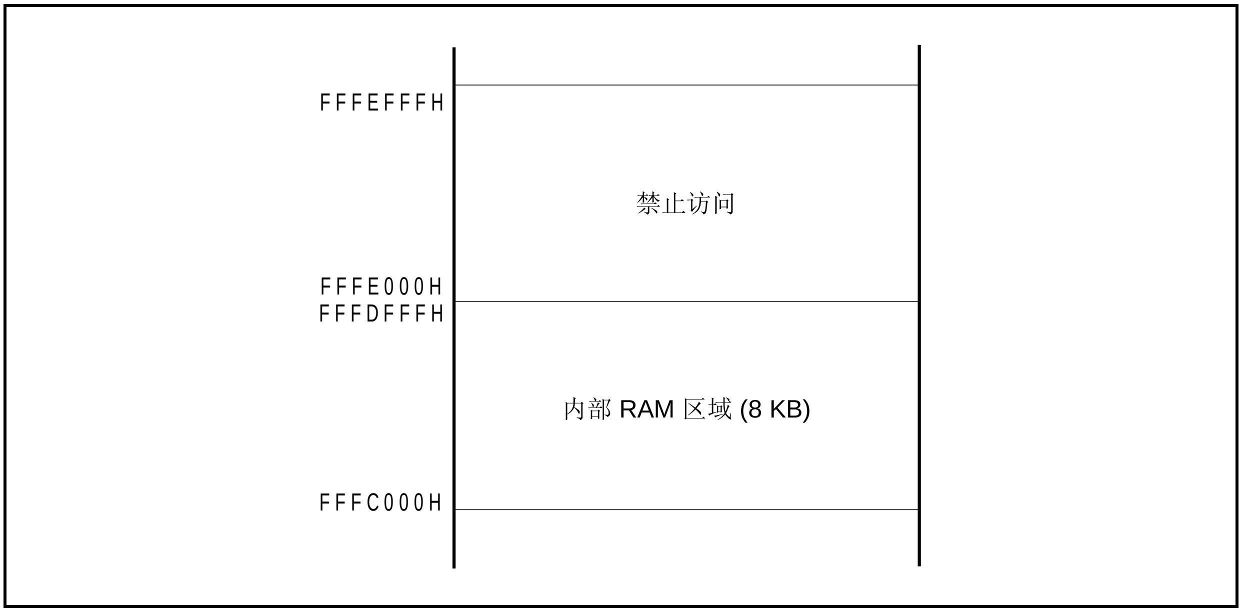
（a） PD70F3451 (V850E/IF3), PD70F3453 (V850E/IG3)

地址 FFFC000H 至 FFFDFFFH 的 8 KB 区域作为物理内部 RAM。

注意事项 下列区域禁止访问。

地址 FFFE000H 至 FFFEFFFH

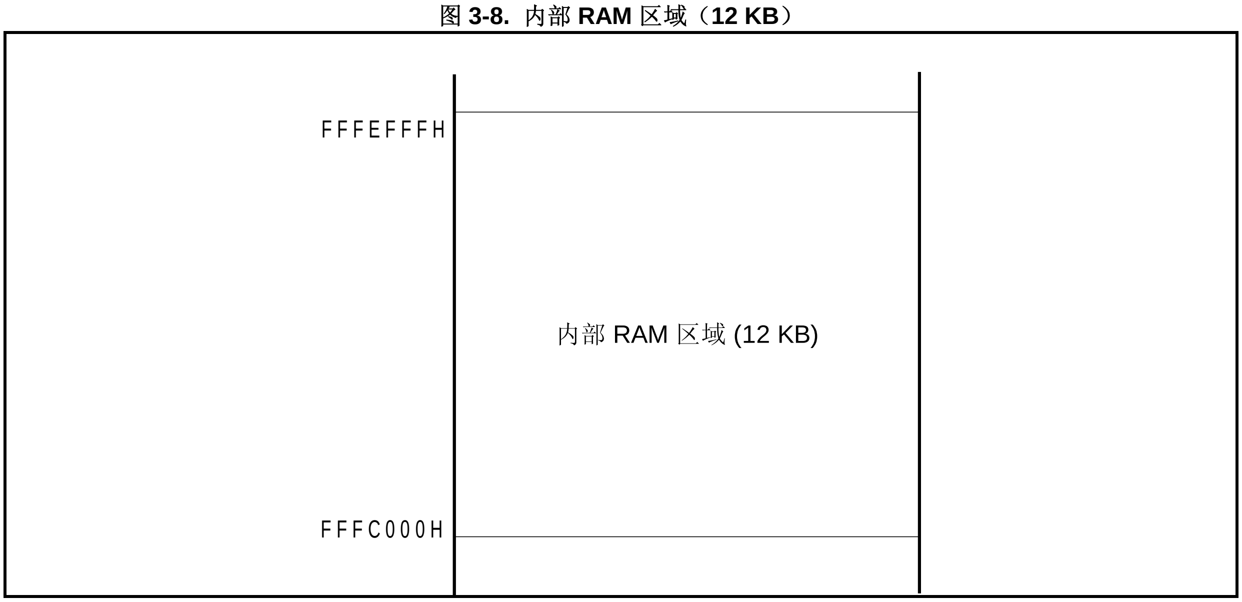
图 3-7. 内部 RAM 区域（8 KB）



（b） PD70F3452 (V850E/IF3), PD70F3454 (V850E/IG3)

地址 FFFC000H 至 FFFEFFFH 的 12 KB 区域作为物理内部 RAM。

图 3-8. 内部 RAM 区域（12 KB）



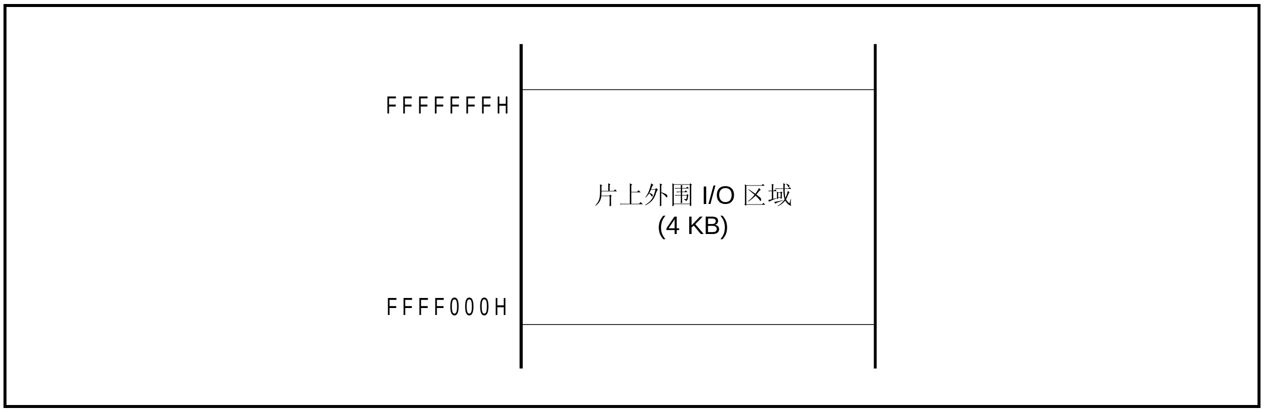
（**3**） 片上外围 **I/O** 区域

地址 FFFF000H 至 FFFFFFFH 的 4 KB 区域被保留作为片上外围 I/O 区域。

地址 3FFF000H至 3FFFFFFH[注]可被看作地址FFFF000H 至 FFFFFFFH的一个映象。

注 **地址** 3FFF000H 至 3FFFFFFH 禁止访问。要访问片上外围 I/O 区域，指定地址 FFFF000H 至 FFFFFFFH。

图 **3-9.** 片上外围 **I/O** 区域



某些外围 I/O 寄存器可以被存储器映射到片内外围 I/O 区域，这些寄存器与操作模式规范和用于片上外围 I/O 的状态监控相关。不能从该区域执行程序获取。

注意事项 **1.** 在 **V850E/IF3** 和 **V850E/IG3** 中，如果对寄存器进行字访问，会按照先低位后高位的顺序，对该字区域执行两次半字访问，其中地址的低 **2** 位被忽略。

2. 支持字节访问的寄存器，如果执行半字访问，在读取操作期间，那么该寄存器的高 **8** 位将不确定，在写入操作期间，数据的低 **8** 位被写入该寄存器。

3. 没有定义为寄存器的地址区域被保留为将来扩展。如果对这些地址进行访问，其操作不确定而且操作的安全性不保证。

地址 **3FFF000H** 至 **3FFFFFFH** 不能被指定作为 **DMA** 传输的源**/**目的地址。请确保使用地址 **FFFF000H** 至 **FFFFFFFH** 作为 **DMA** 传输的源**/**目的地址。

<R> **(4)** 外部存储器区域 **(**仅限 **μPD70F3454GC-8EA-A** 和 **70F3454F1-DA9-A)**

有 3 MB (0100000H 至 03FFFFFH)可以作为外部存储区域。详情参见**第十八章 总线控制功能**。

3.4.6 地址空间的使用建议

V850E/IF3 和 V850E/IG3 架构要求访问数据空间的操作数时，必须有一个寄存器用作指针以产生地址。该指针±32 KB 范围内的地址可以通过操作数指令直接访问。由于能够用作指针的通用寄存器数量是有限的，可以最小程度防止当改变指针值时降低地址计算性能，应该尽可能多的指定通用寄存器来保存变量，这样也可以减少程序大小。

（1）程序空间

对于 32 位的 PC（程序计数器），其高 6 位固定为 0，而只有低 26 位是有效的。因此，从地址 00000000H 开始的 64 MB 连续地址，无条件对应程序空间的存储器映射。

（2）数据空间

对于 V850E/IF3 和 V850E/IG3 来说，256 MB 的物理地址空间可以被看作 4GB CPU 地址空间的 16 个映象。因此，26 位地址的最高有效位（第 25 位）作为地址符号扩展到 32 位。

（a）使用地址绕回的应用示例

如果 LD/ST disp16[R]指令指定 R = r0 （零寄存器）时，符号扩展可以引用 00000000H ± 32KB 地址范围。零寄存器（r0）是由硬件将其值固定为 0 的，这样就不需要专门用作指针的额外寄存器。

示例：μPD70F3454 (V850E/IG3)

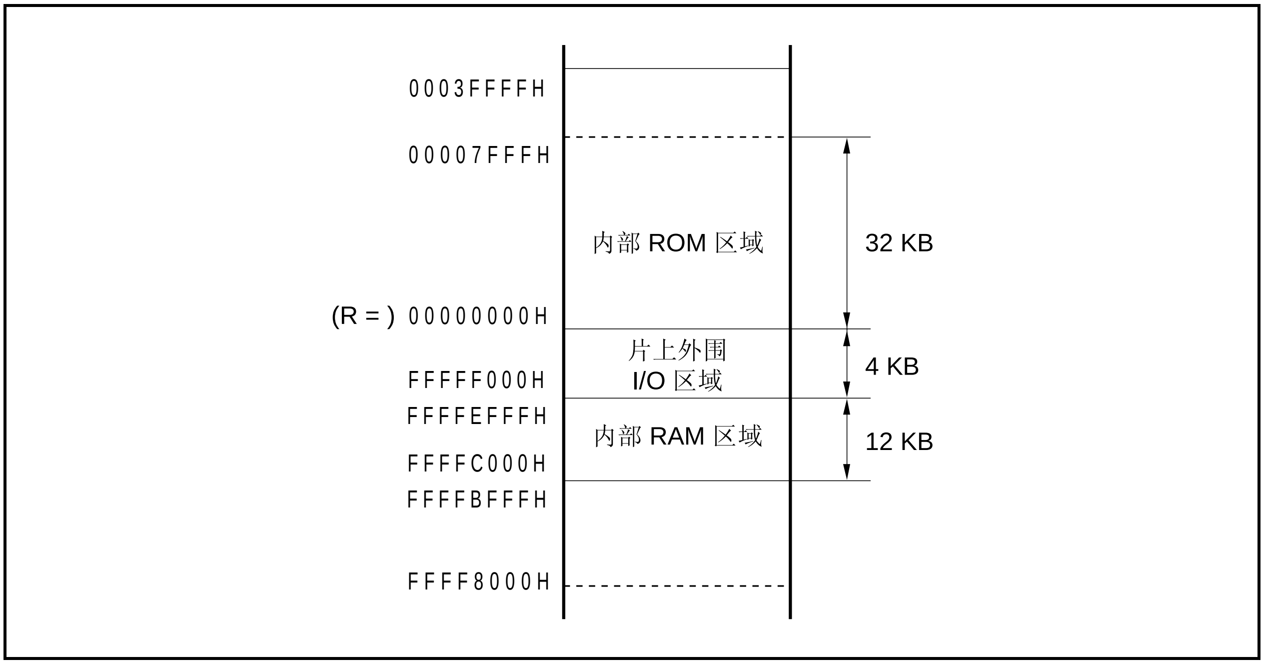
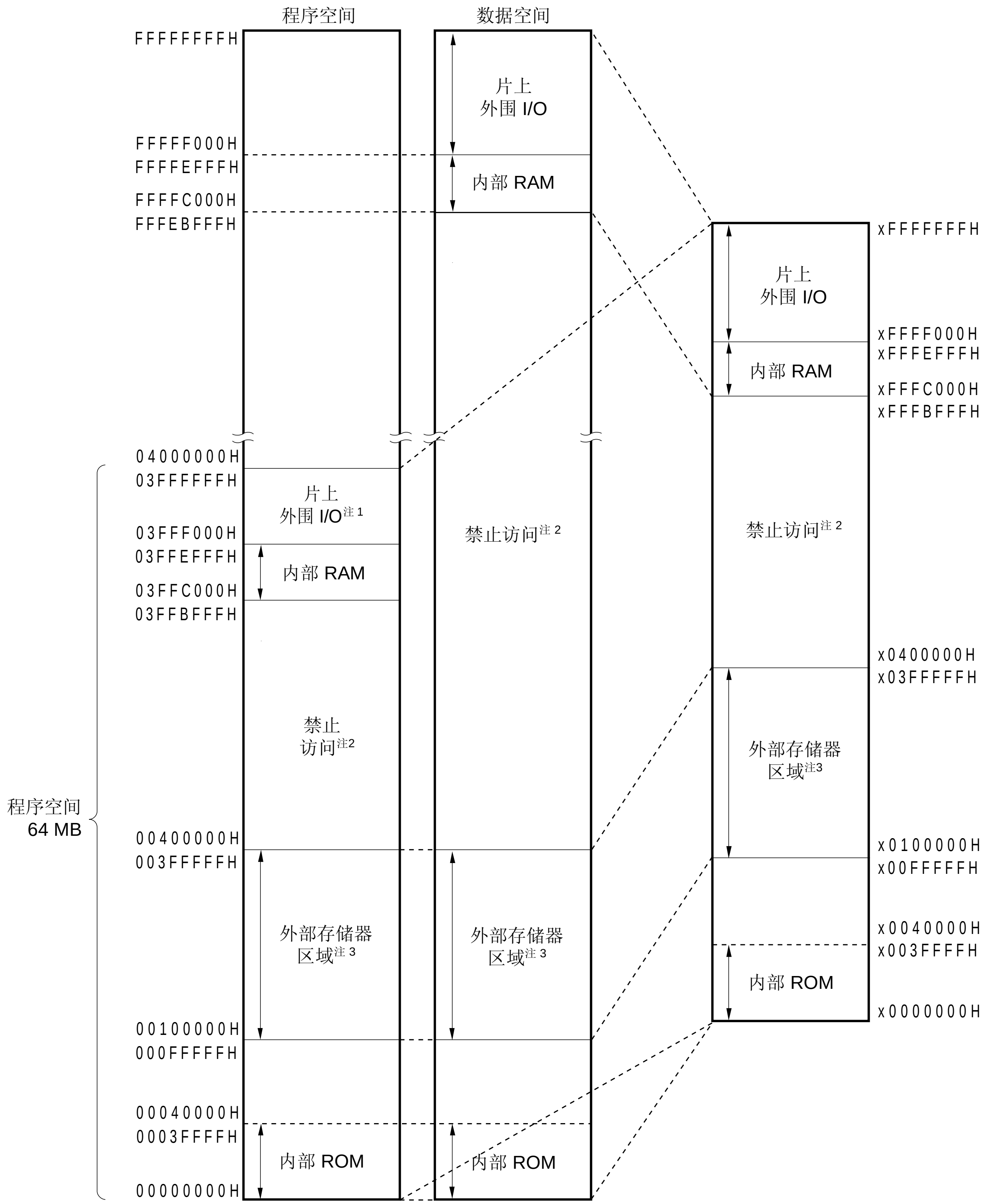


图 3-10. 建议的存储器映射图



注 1. 该区域禁止访问。要访问片上外围 I/O，指定地址 FFFF000H 至 FFFFFFFH。
2. 如果访问了禁止访问区域，操作无法保证。

<R> 3. μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A： 外部存储器区域
其它： 禁止访问区域注 2

备注 1. 箭头表示推荐区域。
2. 这是用于 PD70F3454 (V850E/IG3)的推荐存储器映射。

3.4.7 外围 I/O 寄存器

(1/14)

地址	功能寄存器名称	符号	R/W	操作位数			复位后
				1	8	16	
FFFFF004H	DL 端口寄存器	PDL	R/W			√	不确定
FFFFF004H	DLL 端口寄存器	PDLL		√	√		不确定
FFFFF005H	DLH 端口寄存器	PDLH		√	√		不确定
FFFFF024H	DL 端口模式寄存器	PMDL				√	FFFFH
FFFFF024H	DL 端口模式寄存器 L	PMDLL		√	√		FFH
FFFFF025H	DL 端口模式寄存器 H	PMDLH		√	√		FFH
FFFFF044H	DL 端口模式控制寄存器	PMCDL				√	0000H
FFFFF044H	DL 端口模式控制寄存器 L	PMCDLL		√	√		00H
FFFFF045H	DL 端口模式控制寄存器 H	PMCDLH		√	√		00H
FFFFF066H	总线宽度配置寄存器	BSC				√	5555H
FFFFF06EH	系统等待控制寄存器	VSWC			√		77H
FFFFF080H	DMA 源地址寄存器 0L	DSA0L				√	不确定
FFFFF082H	DMA 源地址寄存器 0H	DSA0H				√	不确定
FFFFF084H	DMA 目的地址寄存器 0L	DDA0L				√	不确定
FFFFF086H	DMA 目的地址寄存器 0H	DDA0H				√	不确定
FFFFF088H	DMA 源地址寄存器 1L	DSA1L				√	不确定
FFFFF08AH	DMA 源地址寄存器 1H	DSA1H				√	不确定
FFFFF08CH	DMA 目的地址寄存器 1L	DDA1L				√	不确定
FFFFF08EH	DMA 目的地址寄存器 1H	DDA1H				√	不确定
FFFFF090H	DMA 源地址寄存器 2L	DSA2L				√	不确定
FFFFF092H	DMA 源地址寄存器 2H	DSA2H				√	不确定
FFFFF094H	DMA 目的地址寄存器 2L	DDA2L				√	不确定
FFFFF096H	DMA 目的地址寄存器 2H	DDA2H				√	不确定
FFFFF098H	DMA 源地址寄存器 3L	DSA3L				√	不确定
FFFFF09AH	DMA 源地址寄存器 3H	DSA3H				√	不确定
FFFFF09CH	DMA 目的地址寄存器 3L	DDA3L				√	不确定
FFFFF09EH	DMA 目的地址寄存器 3H	DDA3H				√	不确定
FFFFF0C0H	DMA 传输计数寄存器 0	DBC0				√	不确定
FFFFF0C2H	DMA 传输计数寄存器 1	DBC1				√	不确定
FFFFF0C4H	DMA 传输计数寄存器 2	DBC2				√	不确定
FFFFF0C6H	DMA 传输计数寄存器 3	DBC3				√	不确定
FFFFF0D0H	DMA 寻址控制寄存器 0	DADC0				√	0000H
FFFFF0D2H	DMA 寻址控制寄存器 1	DADC1				√	0000H
FFFFF0D4H	DMA 寻址控制寄存器 2	DADC2				√	0000H
FFFFF0D6H	DMA 寻址控制寄存器 3	DADC3				√	0000H
FFFFF0E0H	DMA 通道控制寄存器 0	DCHC0		√	√		00H
FFFFF0E2H	DMA 通道控制寄存器 1	DCHC1		√	√		00H
FFFFF0E4H	DMA 通道控制寄存器 2	DCHC2		√	√		00H
FFFFF0E6H	DMA 通道控制寄存器 3	DCHC3		√	√		00H

(2/14)

地址	功能寄存器名称	符号	R/W	操作位数			复位后
				1	8	16	
FFFFF100H	中断屏蔽寄存器 0	IMR0	R/W			√	FFFFH
FFFFF100H	中断屏蔽寄存器 0L	IMR0L		√	√		FFH
FFFFF101H	中断屏蔽寄存器 0H	IMR0H		√	√		FFH
FFFFF102H	中断屏蔽寄存器 1	IMR1				√	FFFFH
FFFFF102H	中断屏蔽寄存器 1L	IMR1L		√	√		FFH
FFFFF103H	中断屏蔽寄存器 1H	IMR1H		√	√		FFH
FFFFF104H	中断屏蔽寄存器 2	IMR2				√	FFFFH
FFFFF104H	中断屏蔽寄存器 2L	IMR2L		√	√		FFH
FFFFF105H	中断屏蔽寄存器 2H	IMR2H		√	√		FFH
FFFFF106H	中断屏蔽寄存器 3	IMR3				√	FFFFH
FFFFF106H	中断屏蔽寄存器 3L	IMR3L		√	√		FFH
FFFFF107H	中断屏蔽寄存器 3H	IMR3H		√	√		FFH
FFFFF108H	中断屏蔽寄存器 4	IMR4				√	FFFFH
FFFFF108H	中断屏蔽寄存器 4L	IMR4L		√	√		FFH
FFFFF109H	中断屏蔽寄存器 4H	IMR4H		√	√		FFH
FFFFF10AH	中断屏蔽寄存器 5	IMR5				√	FFFFH
FFFFF10AH	中断屏蔽寄存器 5L	IMR5L		√	√		FFH
FFFFF10BH	中断屏蔽寄存器 5H	IMR5H		√	√		FFH
FFFFF110H	中断控制寄存器	LVILIC		√	√		47H
FFFFF112H	中断控制寄存器	LVIHIC		√	√		47H
FFFFF114H	中断控制寄存器	PIC00		√	√		47H
FFFFF116H	中断控制寄存器	PIC01		√	√		47H
FFFFF118H	中断控制寄存器	PIC02[注]		√	√		47H
FFFFF11AH	中断控制寄存器	PIC03[注]		√	√		47H
FFFFF11CH	中断控制寄存器	PIC04[注]		√	√		47H
FFFFF11EH	中断控制寄存器	PIC05[注]		√	√		47H
FFFFF120H	中断控制寄存器	PIC06[注]		√	√		47H
FFFFF122H	中断控制寄存器	PIC07[注]		√	√		47H
FFFFF124H	中断控制寄存器	PIC08		√	√		47H
FFFFF126H	中断控制寄存器	PIC09		√	√		47H
FFFFF128H	中断控制寄存器	PIC10		√	√		47H
FFFFF12AH	中断控制寄存器	PIC11		√	√		47H
FFFFF12CH	中断控制寄存器	PIC12		√	√		47H
FFFFF12EH	中断控制寄存器	PIC13		√	√		47H
FFFFF130H	中断控制寄存器	PIC14		√	√		47H
FFFFF132H	中断控制寄存器	PIC15		√	√		47H
FFFFF134H	中断控制寄存器	PIC16		√	√		47H
FFFFF136H	中断控制寄存器	PIC17		√	√		47H
FFFFF138H	中断控制寄存器	PIC18		√	√		47H
FFFFF13AH	中断控制寄存器	CMPIC0L		√	√		47H

注　　仅限 V850E/IG3

(3/14)

地址	功能寄存器名称	符号	R/W	操作位数			复位后
				1	8	16	
FFFFF13CH	中断控制寄存器	CMPIC0F	R/W	√	√		47H
FFFFF13EH	中断控制寄存器	CMPIC1L		√	√		47H
FFFFF140H	中断控制寄存器	CMPIC1F		√	√		47H
FFFFF142H	中断控制寄存器	TB0OVIC		√	√		47H
FFFFF144H	中断控制寄存器	TB0CCIC0		√	√		47H
FFFFF146H	中断控制寄存器	TB0CCIC1		√	√		47H
FFFFF148H	中断控制寄存器	TB0CCIC2		√	√		47H
FFFFF14AH	中断控制寄存器	TB0CCIC3		√	√		47H
FFFFF14CH	中断控制寄存器	TB1OVIC		√	√		47H
FFFFF14EH	中断控制寄存器	TB1CCIC0		√	√		47H
FFFFF150H	中断控制寄存器	TB1CCIC1		√	√		47H
FFFFF152H	中断控制寄存器	TB1CCIC2		√	√		47H
FFFFF154H	中断控制寄存器	TB1CCIC3		√	√		47H
FFFFF156H	中断控制寄存器	TT0OVIC		√	√		47H
FFFFF158H	中断控制寄存器	TT0CCIC0		√	√		47H
FFFFF15AH	中断控制寄存器	TT0CCIC1		√	√		47H
FFFFF15CH	中断控制寄存器	TT0IECIC[注]		√	√		47H
FFFFF15EH	中断控制寄存器	TT1OVIC		√	√		47H
FFFFF160H	中断控制寄存器	TT1CCIC0		√	√		47H
FFFFF162H	中断控制寄存器	TT1CCIC1		√	√		47H
FFFFF164H	中断控制寄存器	TT1IECIC		√	√		47H
FFFFF166H	中断控制寄存器	TA0OVIC		√	√		47H
FFFFF168H	中断控制寄存器	TA0CCIC0		√	√		47H
FFFFF16AH	中断控制寄存器	TA0CCIC1		√	√		47H
FFFFF16CH	中断控制寄存器	TA1OVIC		√	√		47H
FFFFF16EH	中断控制寄存器	TA1CCIC0		√	√		47H
FFFFF170H	中断控制寄存器	TA1CCIC1		√	√		47H
FFFFF172H	中断控制寄存器	TA2OVIC		√	√		47H
FFFFF174H	中断控制寄存器	TA2CCIC0		√	√		47H
FFFFF176H	中断控制寄存器	TA2CCIC1		√	√		47H
FFFFF178H	中断控制寄存器	TA3OVIC		√	√		47H
FFFFF17AH	中断控制寄存器	TA3CCIC0		√	√		47H
FFFFF17CH	中断控制寄存器	TA3CCIC1		√	√		47H
FFFFF17EH	中断控制寄存器	TA4OVIC		√	√		47H
FFFFF180H	中断控制寄存器	TA4CCIC0		√	√		47H
FFFFF182H	中断控制寄存器	TA4CCIC1		√	√		47H
FFFFF184H	中断控制寄存器	DMAIC0		√	√		47H
FFFFF186H	中断控制寄存器	DMAIC1		√	√		47H
FFFFF188H	中断控制寄存器	DMAIC2		√	√		47H
FFFFF18AH	中断控制寄存器	DMAIC3		√	√		47H

注　　仅限 V850E/IG3

(4/14)

地址	功能寄存器名称	符号	R/W	操作位数			复位后
				1	8	16	
FFFFF18CH	中断控制寄存器	UREIC	R/W	√	√		47H
FFFFF18EH	中断控制寄存器	URIC		√	√		47H
FFFFF190H	中断控制寄存器	UTIC		√	√		47H
FFFFF192H	中断控制寄存器	UIFIC		√	√		47H
FFFFF194H	中断控制寄存器	UTOIC		√	√		47H
FFFFF196H	中断控制寄存器	UA0REIC		√	√		47H
FFFFF198H	中断控制寄存器	UA0RIC		√	√		47H
FFFFF19AH	中断控制寄存器	UA0TIC		√	√		47H
FFFFF19CH	中断控制寄存器	CB0REIC		√	√		47H
FFFFF19EH	中断控制寄存器	CB0RIC		√	√		47H
FFFFF1A0H	中断控制寄存器	CB0TIC		√	√		47H
FFFFF1A2H	中断控制寄存器	UA1REIC		√	√		47H
FFFFF1A4H	中断控制寄存器	UA1RIC		√	√		47H
FFFFF1A6H	中断控制寄存器	UA1TIC		√	√		47H
FFFFF1A8H	中断控制寄存器	CB1REIC		√	√		47H
FFFFF1AAH	中断控制寄存器	CB1RIC		√	√		47H
FFFFF1ACH	中断控制寄存器	CB1TIC		√	√		47H
FFFFF1AEH	中断控制寄存器	UA2REIC		√	√		47H
FFFFF1B0H	中断控制寄存器	UA2RIC		√	√		47H
FFFFF1B2H	中断控制寄存器	UA2TIC		√	√		47H
FFFFF1B4H	中断控制寄存器	CB2REIC		√	√		47H
FFFFF1B6H	中断控制寄存器	CB2RIC		√	√		47H
FFFFF1B8H	中断控制寄存器	CB2TIC		√	√		47H
FFFFF1BAH	中断控制寄存器	IICIC		√	√		47H
FFFFF1BCH	中断控制寄存器	AD0IC		√	√		47H
FFFFF1BEH	中断控制寄存器	AD1IC		√	√		47H
FFFFF1C0H	中断控制寄存器	AD2IC		√	√		47H
FFFFF1C2H	中断控制寄存器	TM0EQIC0		√	√		47H
FFFFF1C4H	中断控制寄存器	TM1EQIC0		√	√		47H
FFFFF1C6H	中断控制寄存器	TM2EQIC0		√	√		47H
FFFFF1C8H	中断控制寄存器	TM3EQIC0		√	√		47H
FFFFF1CAH	中断控制寄存器	ADT0IC		√	√		47H
FFFFF1CCH	中断控制寄存器	ADT1IC		√	√		47H
FFFFF1FAH	正在服务的中断优先级寄存器	ISPR	R	√	√		00H
FFFFF1FCH	命令寄存器	PRCMD	W		√		不确定
FFFFF1FEH	省电控制寄存器	PSC	R/W	√	√		00H
FFFFF200H	A/D0 转换结果寄存器 0	AD0CR0	R			√	0000H
FFFFF201H	A/D0 转换结果寄存器 0H	AD0CR0H			√		00H
FFFFF202H	A/D0 转换结果寄存器 1	AD0CR1				√	0000H
FFFFF203H	A/D0 转换结果寄存器 1H	AD0CR1H			√		00H

(5/14)

地址	功能寄存器名称	符号	R/W	操作位数			复位后
				1	8	16	
FFFFF204H	A/D0 转换结果寄存器 2	AD0CR2	R			√	0000H
FFFFF205H	A/D0 转换结果寄存器 2H	AD0CR2H			√		00H
FFFFF206H	A/D0 转换结果寄存器 3	AD0CR3				√	0000H
FFFFF207H	A/D0 转换结果寄存器 3H	AD0CR3H			√		00H
FFFFF208H	A/D0 转换结果寄存器 4	AD0CR4				√	0000H
FFFFF209H	A/D0 转换结果寄存器 4H	AD0CR4H			√		00H
FFFFF20AH	A/D0 转换结果寄存器 5	AD0CR5				√	0000H
FFFFF20BH	A/D0 转换结果寄存器 5H	AD0CR5H			√		00H
FFFFF20CH	A/D0 转换结果寄存器 6	AD0CR6				√	0000H
FFFFF20DH	A/D0 转换结果寄存器 6H	AD0CR6H			√		00H
FFFFF20EH	A/D0 转换结果寄存器 7	AD0CR7				√	0000H
FFFFF20FH	A/D0 转换结果寄存器 7H	AD0CR7H			√		00H
FFFFF210H	A/D0 转换结果寄存器 8	AD0CR8				√	0000H
FFFFF211H	A/D0 转换结果寄存器 8H	AD0CR8H			√		00H
FFFFF212H	A/D0 转换结果寄存器 9	AD0CR9				√	0000H
FFFFF213H	A/D0 转换结果寄存器 9H	AD0CR9H			√		00H
FFFFF214H	A/D0 转换结果寄存器 10	AD0CR10				√	0000H
FFFFF215H	A/D0 转换结果寄存器 10H	AD0CR10H			√		00H
FFFFF216H	A/D0 转换结果寄存器 11	AD0CR11				√	0000H
FFFFF217H	A/D0 转换结果寄存器 11H	AD0CR11H			√		00H
FFFFF218H	A/D0 转换结果寄存器 12	AD0CR12				√	0000H
FFFFF219H	A/D0 转换结果寄存器 12H	AD0CR12H			√		00H
FFFFF21AH	A/D0 转换结果寄存器 13	AD0CR13				√	0000H
FFFFF21BH	A/D0 转换结果寄存器 13H	AD0CR13H			√		00H
FFFFF21CH	A/D0 转换结果寄存器 14	AD0CR14				√	0000H
FFFFF21DH	A/D0 转换结果寄存器 14H	AD0CR14H			√		00H
FFFFF21EH	A/D0 转换结果寄存器 15	AD0CR15				√	0000H
FFFFF21FH	A/D0 转换结果寄存器 15H	AD0CR15H			√		00H
FFFFF220H	A/D 转换器 0 扫描模式寄存器	AD0SCM	R/W			√	0000H
FFFFF220H	A/D 转换器 0 扫描模式寄存器 L	AD0SCML		√	√		00H
FFFFF221H	A/D 转换器 0 扫描模式寄存器 H	AD0SCMH		√	√		00H
FFFFF222H	A/D 转换器 0 转换时间控制寄存器	AD0CTC		√	√		00H
FFFFF224H	A/D 转换器 0 转换通道指定寄存器	AD0CHEN				√	0000H
FFFFF224H	A/D 转换器 0 转换通道指定寄存器 L	AD0CHENL		√	√		00H
FFFFF225H	A/D 转换器 0 转换通道指定寄存器 H	AD0CHENH		√	√		00H
FFFFF230H	A/D 转换器 0 控制寄存器	AD0CTL0		√	√		00H
FFFFF231H	A/D 转换器 0 触发选择寄存器	AD0TSEL		√	√		10H
FFFFF232H	A/D 转换器 0 通道指定寄存器 1	AD0CH1		√	√		00H

(6/14)

地址	功能寄存器名称	符号	R/W	操作位数			复位后
				1	8	16	
FFFFF233H	A/D 转换器 0 通道指定寄存器 2	AD0CH2	R/W	√	√		00H
FFFFF240H	A/D0 转换结果扩展寄存器 0	AD0ECR0	R			√	0000H
FFFFF241H	A/D0 转换结果扩展寄存器 0H	AD0ECR0H			√		00H
FFFFF242H	A/D0 转换结果扩展寄存器 1	AD0ECR1				√	0000H
FFFFF243H	A/D0 转换结果扩展寄存器 1H	AD0ECR1H			√		00H
FFFFF244H	A/D0 转换结果扩展寄存器 2	AD0ECR2				√	0000H
FFFFF245H	A/D0 转换结果扩展寄存器 2H	AD0ECR2H			√		00H
FFFFF246H	A/D0 转换结果扩展寄存器 3	AD0ECR3				√	0000H
FFFFF247H	A/D0 转换结果扩展寄存器 3H	AD0ECR3H			√		00H
FFFFF248H	A/D0 转换结果扩展寄存器 4	AD0ECR4				√	0000H
FFFFF249H	A/D0 转换结果扩展寄存器 4H	AD0ECR4H			√		00H
FFFFF254H	A/D 转换器 0 标志寄存器	AD0FLG			√		00H
FFFFF255H	A/D 转换器 0 标志缓冲寄存器	AD0FLGB			√		00H
FFFFF260H	运算放大器 0 控制寄存器 0	OP0CTL0	R/W		√		00H
FFFFF261H	比较器 0 控制寄存器 0	CMP0CTL0			√		00H
FFFFF262H	比较器 0 控制寄存器 1	CMP0CTL1	R		√		00H
FFFFF263H	比较器 0 控制寄存器 2	CMP0CTL2	R/W		√		00H
FFFFF264H	比较器 0 控制寄存器 3	CMP0CTL3			√		00H
FFFFF270H	A/D 转换器 0 时钟选择寄存器	AD0OCKS			√		00H
FFFFF274H	A/D 转换器 1 时钟选择寄存器	AD1OCKS			√		00H
FFFFF278H	比较器输出数字噪声消除寄存器 0L	CMPNFC0L			√		00H
FFFFF27AH	比较器输出数字噪声消除寄存器 0F	CMPNFC0F			√		00H
FFFFF27CH	比较器输出数字噪声消除寄存器 1L	CMPNFC1L			√		00H
FFFFF27EH	比较器输出数字噪声消除寄存器 1F	CMPNFC1F			√		00H
FFFFF280H	A/D1 转换结果寄存器 0	AD1CR0	R			√	0000H
FFFFF281H	A/D1 转换结果寄存器 0H	AD1CR0H			√		00H
FFFFF282H	A/D1 转换结果寄存器 1	AD1CR1				√	0000H
FFFFF283H	A/D1 转换结果寄存器 1H	AD1CR1H			√		00H
FFFFF284H	A/D1 转换结果寄存器 2	AD1CR2				√	0000H
FFFFF285H	A/D1 转换结果寄存器 2H	AD1CR2H			√		00H
FFFFF286H	A/D1 转换结果寄存器 3	AD1CR3				√	0000H
FFFFF287H	A/D1 转换结果寄存器 3H	AD1CR3H			√		00H
FFFFF288H	A/D1 转换结果寄存器 4	AD1CR4				√	0000H
FFFFF289H	A/D1 转换结果寄存器 4H	AD1CR4H			√		00H
FFFFF28AH	A/D1 转换结果寄存器 5	AD1CR5				√	0000H
FFFFF28BH	A/D1 转换结果寄存器 5H	AD1CR5H			√		00H
FFFFF28CH	A/D1 转换结果寄存器 6	AD1CR6				√	0000H
FFFFF28DH	A/D1 转换结果寄存器 6H	AD1CR6H			√		00H
FFFFF28EH	A/D1 转换结果寄存器 7	AD1CR7				√	0000H
FFFFF28FH	A/D1 转换结果寄存器 7H	AD1CR7H			√		00H
FFFFF290H	A/D1 转换结果寄存器 8	AD1CR8				√	0000H
FFFFF291H	A/D1 转换结果寄存器 8H	AD1CR8H			√		00H

(7/14)

地址	功能寄存器名称	符号	R/W	操作位数			复位后
				1	8	16	
FFFFF292H	A/D1 转换结果寄存器 9	AD1CR9	R			√	0000H
FFFFF293H	A/D1 转换结果寄存器 9H	AD1CR9H			√		00H
FFFFF294H	A/D1 转换结果寄存器 10	AD1CR10				√	0000H
FFFFF295H	A/D1 转换结果寄存器 10H	AD1CR10H			√		00H
FFFFF296H	A/D1 转换结果寄存器 11	AD1CR11				√	0000H
FFFFF297H	A/D1 转换结果寄存器 11H	AD1CR11H			√		00H
FFFFF298H	A/D1 转换结果寄存器 12	AD1CR12				√	0000H
FFFFF299H	A/D1 转换结果寄存器 12H	AD1CR12H			√		00H
FFFFF29AH	A/D1 转换结果寄存器 13	AD1CR13				√	0000H
FFFFF29BH	A/D1 转换结果寄存器 13H	AD1CR13H			√		00H
FFFFF29CH	A/D1 转换结果寄存器 14	AD1CR14				√	0000H
FFFFF29DH	A/D1 转换结果寄存器 14H	AD1CR14H			√		00H
FFFFF29EH	A/D1 转换结果寄存器 15	AD1CR15				√	0000H
FFFFF29FH	A/D1 转换结果寄存器 15H	AD1CR15H			√		00H
FFFFF2A0H	A/D 转换器 1 扫描模式寄存器	AD1SCM	R/W			√	0000H
FFFFF2A0H	A/D 转换器 1 扫描模式寄存器 L	AD1SCML		√	√		00H
FFFFF2A1H	A/D 转换器 1 扫描模式寄存器 H	AD1SCMH		√	√		00H
FFFFF2A2H	A/D 转换器 1 conversion time 控制寄存器	AD1CTC		√	√		00H
FFFFF2A4H	A/D 转换器 1 转换通道指定寄存器	AD1CHEN				√	0000H
FFFFF2A4H	A/D 转换器 1 转换通道指定寄存器 L	AD1CHENL		√	√		00H
FFFFF2A5H	A/D 转换器 1 转换通道指定寄存器 H	AD1CHENH		√	√		00H
FFFFF2B0H	A/D 转换器 1 控制寄存器	AD1CTL0		√	√		00H
FFFFF2B1H	A/D 转换器 1 触发选择寄存器	AD1TSEL		√	√		10H
FFFFF2B2H	A/D 转换器 1 通道指定寄存器 1	AD1CH1		√	√		00H
FFFFF2B3H	A/D 转换器 1 通道指定寄存器 2	AD1CH2		√	√		00H
FFFFF2C0H	A/D1 转换结果扩展寄存器 0	AD1ECR0	R			√	0000H
FFFFF2C1H	A/D1 转换结果扩展寄存器 0H	AD1ECR0H			√		00H
FFFFF2C2H	A/D1 转换结果扩展寄存器 1	AD1ECR1				√	0000H
FFFFF2C3H	A/D1 转换结果扩展寄存器 1H	AD1ECR1H			√		00H
FFFFF2C4H	A/D1 转换结果扩展寄存器 2	AD1ECR2				√	0000H
FFFFF2C5H	A/D1 转换结果扩展寄存器 2H	AD1ECR2H			√		00H
FFFFF2C6H	A/D1 转换结果扩展寄存器 3	AD1ECR3				√	0000H
FFFFF2C7H	A/D1 转换结果扩展寄存器 3H	AD1ECR3H			√		00H
FFFFF2C8H	A/D1 转换结果扩展寄存器 4	AD1ECR4				√	0000H
FFFFF2C9H	A/D1 转换结果扩展寄存器 4H	AD1ECR4H			√		00H
FFFFF2D4H	A/D 转换器 1 标志寄存器	AD1FLG			√		00H
FFFFF2D5H	A/D 转换器 1 标志缓冲寄存器	AD1FLGB			√		00H
FFFFF2E0H	运算放大器 1 控制寄存器 0	OP1CTL0	R/W		√		00H
FFFFF2E1H	比较器 1 控制寄存器 0	CMP1CTL0			√		00H
FFFFF2E2H	比较器 1 控制寄存器 1	CMP1CTL1	R		√		00H

(8/14)

地址	功能寄存器名称	符号	R/W	操作位数			复位后
				1	8	16	
FFFFF2E3H	比较器 1 控制寄存器 2	CMP1CTL2	R/W		√		00H
FFFFF2E4H	比较器 1 控制寄存器 3	CMP1CTL3			√		00H
FFFFF2F0H	A/D 触发下降沿指定寄存器	ADTF		√	√		00H
FFFFF2F2H	A/D 触发上升沿指定寄存器	ADTR		√	√		00H
FFFFF2F4H	比较器输出中断下降沿指定寄存器	CMPOF		√	√		00H
FFFFF2F6H	比较器输出中断上升沿指定寄存器	CMPOR		√	√		00H
FFFFF2F8H	A/DLDTRG1 输入选择寄存器	ADLTS1			√		00H
FFFFF2FAH	A/DLDTRG2 输入选择寄存器	ADLTS2			√		00H
FFFFF310H	数字噪声消除 0 控制寄存器 14	INTNFC14			√		00H
FFFFF312H	数字噪声消除 0 控制寄存器 15	INTNFC15			√		00H
FFFFF314H	数字噪声消除 0 控制寄存器 16	INTNFC16			√		00H
FFFFF400H	端口 0 寄存器	P0		√	√		不确定
FFFFF402H	端口 1 寄存器	P1		√	√		不确定
FFFFF404H	端口 2 寄存器	P2		√	√		不确定
FFFFF406H	端口 3 寄存器	P3		√	√		不确定
FFFFF408H	端口 4 寄存器	P4		√	√		不确定
FFFFF420H	端口 0 模式寄存器	PM0		√	√		FFH
FFFFF422H	端口 1 模式寄存器	PM1		√	√		FFH
FFFFF424H	端口 2 模式寄存器	PM2		√	√		FFH
FFFFF426H	端口 3 模式寄存器	PM3		√	√		FFH
FFFFF428H	端口 4 模式寄存器	PM4		√	√		FFH
FFFFF440H	端口 0 模式控制寄存器	PMC0		√	√		00H
FFFFF442H	端口 1 模式控制寄存器	PMC1		√	√		00H
FFFFF444H	端口 2 模式控制寄存器	PMC2		√	√		00H
FFFFF446H	端口 3 模式控制寄存器	PMC3		√	√		00H
FFFFF448H	端口 4 模式控制寄存器	PMC4		√	√		00H
FFFFF460H	端口 0 功能控制寄存器	PFC0		√	√		00H
FFFFF462H	端口 1 功能控制寄存器	PFC1		√	√		00H
FFFFF464H	端口 2 功能控制寄存器	PFC2		√	√		00H
FFFFF466H	端口 3 功能控制寄存器	PFC3		√	√		00H
FFFFF468H	端口 4 功能控制寄存器	PFC4		√	√		00H
FFFFF480H	总线周期类型配置寄存器 0	BCT0				√	CCCCH
FFFFF484H	数据等待控制寄存器 0	DWC0				√	7777H
FFFFF488H	地址等待控制寄存器	AWC				√	FFFFH
FFFFF48AH	总线周期控制寄存器	BCC				√	AAAAH
FFFFF48EH	总线时钟分频控制寄存器	DVC			√		81H
FFFFF540H	TMM0 控制寄存器 0	TM0CTL0		√	√		00H
FFFFF544H	TMM0 比较寄存器 0	TM0CMP0				√	0000H
FFFFF550H	TMM1 控制寄存器 0	TM1CTL0		√	√		00H

(9/14)

地址	功能寄存器名称	符号	R/W	操作位数			复位后
				1	8	16	
FFFFF554H	TMM1 比较寄存器 0	TM1CMP0	R/W			√	0000H
FFFFF560H	TMM2 控制寄存器 0	TM2CTL0		√	√		00H
FFFFF564H	TMM2 比较寄存器 0	TM2CMP0				√	0000H
FFFFF570H	TMM3 控制寄存器 0	TM3CTL0		√	√		00H
FFFFF574H	TMM3 比较寄存器 0	TM3CMP0				√	0000H
FFFFF580H	TMT0 控制寄存器 0	TT0CTL0		√	√		00H
FFFFF581H	TMT0 控制寄存器 1	TT0CTL1		√	√		00H
FFFFF582H	TMT0 控制寄存器 2	TT0CTL2[注]		√	√		00H
FFFFF583H	TMT0 I/O 控制寄存器 0	TT0IOC0[注]		√	√		00H
FFFFF584H	TMT0 I/O 控制寄存器 1	TT0IOC1[注]		√	√		00H
FFFFF585H	TMT0 I/O 控制寄存器 2	TT0IOC2[注]		√	√		00H
FFFFF586H	TMT0 I/O 控制寄存器 3	TT0IOC3[注]		√	√		00H
FFFFF587H	TMT0 选项寄存器 0	TT0OPT0		√	√		00H
FFFFF588H	TMT0 选项寄存器 1	TT0OPT1[注]		√	√		00H
FFFFF58AH	TMT0 捕获/比较寄存器 0	TT0CCR0				√	0000H
FFFFF58CH	TMT0 捕获/比较寄存器 1	TT0CCR1				√	0000H
FFFFF58EH	TMT0 计数器读取缓冲寄存器	TT0CNT	R			√	0000H
FFFFF590H	TMT0 计数器写入寄存器	TT0TCW[注]	R/W			√	0000H
FFFFF5A0H	数字噪声消除 2 控制寄存器 0	TTNFC0[注]			√		00H
FFFFF5A2H	数字噪声消除 2 控制寄存器 1	TTNFC1			√		00H
FFFFF5A4H	TMT0 捕获输入选择寄存器	TTISL0[注]			√		不确定
FFFFF5A6H	TMT1 捕获输入选择寄存器	TTISL1			√		不确定
FFFFF5C0H	TMT1 控制寄存器 0	TT1CTL0		√	√		00H
FFFFF5C1H	TMT1 控制寄存器 1	TT1CTL1		√	√		00H
FFFFF5C2H	TMT1 控制寄存器 2	TT1CTL2		√	√		00H
FFFFF5C3H	TMT1 I/O 控制寄存器 0	TT1IOC0		√	√		00H
FFFFF5C4H	TMT1 I/O 控制寄存器 1	TT1IOC1		√	√		00H
FFFFF5C5H	TMT1 I/O 控制寄存器 2	TT1IOC2		√	√		00H
FFFFF5C6H	TMT1 I/O 控制寄存器 3	TT1IOC3		√	√		00H
FFFFF5C7H	TMT1 选项寄存器 0	TT1OPT0		√	√		00H
FFFFF5C8H	TMT1 选项寄存器 1	TT1OPT1		√	√		00H
FFFFF5CAH	TMT1 捕获/比较寄存器 0	TT1CCR0				√	0000H
FFFFF5CCH	TMT1 捕获/比较寄存器 1	TT1CCR1				√	0000H
FFFFF5CEH	TMT1 计数器读取缓冲寄存器	TT1CNT	R			√	0000H
FFFFF5D0H	TMT1 计数器写入寄存器	TT1TCW	R/W			√	0000H
FFFFF5E0H	TAB0 控制寄存器 0	TAB0CTL0		√	√		00H
FFFFF5E1H	TAB0 控制寄存器 1	TAB0CTL1		√	√		00H
FFFFF5E2H	TAB0 I/O 控制寄存器 0	TAB0IOC0		√	√		00H
FFFFF5E3H	TAB0 I/O 控制寄存器 1	TAB0IOC1		√	√		00H
FFFFF5E4H	TAB0 I/O 控制寄存器 2	TAB0IOC2		√	√		00H

注　仅限 V850E/IG3

(10/14)

地址	功能寄存器名称	符号	R/W	操作位数			复位后
				1	8	16	
FFFFF5E5H	TAB0 选项寄存器 0	TAB0OPT0	R/W	√	√		00H
FFFFF5E6H	TAB0 捕获/比较寄存器 0	TAB0CCR0				√	0000H
FFFFF5E8H	TAB0 捕获/比较寄存器 1	TAB0CCR1				√	0000H
FFFFF5EAH	TAB0 捕获/比较寄存器 2	TAB0CCR2				√	0000H
FFFFF5ECH	TAB0 捕获/比较寄存器 3	TAB0CCR3				√	0000H
FFFFF5EEH	TAB0 计数器读取缓冲寄存器	TAB0CNT	R			√	0000H
FFFFF600H	TAB0 选项寄存器 1	TAB0OPT1	R/W	√	√		00H
FFFFF601H	TAB0 选项寄存器 2	TAB0OPT2		√	√		00H
FFFFF602H	TAB0 I/O 控制寄存器 3	TAB0IOC3		√	√		A8H
FFFFF603H	TAB0 选项寄存器 3	TAB0OPT3		√	√		00H
FFFFF604H	TAB0 死区时间比较寄存器	TAB0DTC				√	0000H
FFFFF610H	高阻抗输出控制寄存器 00	HZA0CTL0		√	√		00H
FFFFF611H	高阻抗输出控制寄存器 01	HZA0CTL1		√	√		00H
FFFFF618H	高阻抗输出控制寄存器 10	HZA1CTL0		√	√		00H
FFFFF619H	高阻抗输出控制寄存器 11	HZA1CTL1[注]		√	√		00H
FFFFF620H	TAB1 控制寄存器 0	TAB1CTL0		√	√		00H
FFFFF621H	TAB1 控制寄存器 1	TAB1CTL1		√	√		00H
FFFFF622H	TAB1 I/O 控制寄存器 0	TAB1IOC0		√	√		00H
FFFFF623H	TAB1 I/O 控制寄存器 1	TAB1IOC1		√	√		00H
FFFFF624H	TAB1 I/O 控制寄存器 2	TAB1IOC2		√	√		00H
FFFFF625H	TAB1 选项寄存器 0	TAB1OPT0		√	√		00H
FFFFF626H	TAB1 捕获/比较寄存器 0	TAB1CCR0				√	0000H
FFFFF628H	TAB1 捕获/比较寄存器 1	TAB1CCR1				√	0000H
FFFFF62AH	TAB1 捕获/比较寄存器 2	TAB1CCR2				√	0000H
FFFFF62CH	TAB1 捕获/比较寄存器 3	TAB1CCR3				√	0000H
FFFFF62EH	TAB1 计数器读取缓冲寄存器	TAB1CNT	R			√	0000H
FFFFF640H	TAB1 选项寄存器 1	TAB1OPT1	R/W	√	√		00H
FFFFF641H	TAB1 选项寄存器 2	TAB1OPT2		√	√		00H
FFFFF642H	TAB1 I/O 控制寄存器 3	TAB1IOC3		√	√		A8H
FFFFF643H	TAB1 选项寄存器 3	TAB1OPT3		√	√		00H
FFFFF644H	TAB1 死区时间比较寄存器	TAB1DTC				√	0000H
FFFFF650H	高阻抗输出控制寄存器 20	HZA2CTL0		√	√		00H
FFFFF651H	高阻抗输出控制寄存器 21	HZA2CTL1		√	√		00H
FFFFF658H	高阻抗输出控制寄存器 30	HZA3CTL0		√	√		00H
FFFFF659H	高阻抗输出控制寄存器 31	HZA3CTL1		√	√		00H
FFFFF660H	TAA0 控制寄存器 0	TAA0CTL0		√	√		00H
FFFFF661H	TAA0 控制寄存器 1	TAA0CTL1		√	√		00H
FFFFF665H	TAA0 选项寄存器 0	TAA0OPT0		√	√		00H
FFFFF666H	TAA0 捕获/比较寄存器 0	TAA0CCR0				√	0000H
FFFFF668H	TAA0 捕获/比较寄存器 1	TAA0CCR1				√	0000H
FFFFF66AH	TAA0 计数器读取缓冲寄存器	TAA0CNT	R			√	0000H

注　　仅限 V850E/IG3

(11/14)

地址	功能寄存器名称	符号	R/W	操作位数			复位后
				1	8	16	
FFFFF680H	TAA1 控制寄存器 0	TAA1CTL0	R/W	√	√		00H
FFFFF681H	TAA1 控制寄存器 1	TAA1CTL1		√	√		00H
FFFFF685H	TAA1 选项寄存器 0	TAA1OPT0		√	√		00H
FFFFF686H	TAA1 捕获/比较寄存器 0	TAA1CCR0				√	0000H
FFFFF688H	TAA1 捕获/比较寄存器 1	TAA1CCR1				√	0000H
FFFFF68AH	TAA1 计数器读取缓冲寄存器	TAA1CNT	R			√	0000H
FFFFF6A0H	TAA2 控制寄存器 0	TAA2CTL0	R/W	√	√		00H
FFFFF6A1H	TAA2 控制寄存器 1	TAA2CTL1		√	√		00H
FFFFF6A2H	TAA2 I/O 控制寄存器 0	TAA2IOC0		√	√		00H
FFFFF6A3H	TAA2 I/O 控制寄存器 1	TAA2IOC1		√	√		00H
FFFFF6A4H	TAA2 I/O 控制寄存器 2	TAA2IOC2		√	√		00H
FFFFF6A5H	TAA2 选项寄存器 0	TAA2OPT0		√	√		00H
FFFFF6A6H	TAA2 捕获/比较寄存器 0	TAA2CCR0				√	0000H
FFFFF6A8H	TAA2 捕获/比较寄存器 1	TAA2CCR1				√	0000H
FFFFF6AAH	TAA2 计数器读取缓冲寄存器	TAA2CNT	R			√	0000H
FFFFF6C0H	振荡稳定时间选择寄存器	OSTS	R/W		√		04H
FFFFF6D0H	看门狗定时器模式寄存器	WDTM			√		67H
FFFFF6D1H	看门狗定时器使能寄存器	WDTE			√		1AH
FFFFF700H	端口 0 功能控制扩展寄存器	PFCE0		√	√		00H
FFFFF702H	端口 1 功能控制扩展寄存器	PFCE1		√	√		00H
FFFFF704H	端口 2 功能控制扩展寄存器	PFCE2		√	√		00H
FFFFF706H	端口 3 功能控制扩展寄存器	PFCE3		√	√		00H
FFFFF708H	端口 4 功能控制扩展寄存器	PFCE4		√	√		00H
FFFFF802H	系统状态寄存器	SYS		√	√		00H
FFFFF810H	DMA 触发因素寄存器 0	DTFR0		√	√		00H
FFFFF812H	DMA 触发因素寄存器 1	DTFR1		√	√		00H
FFFFF814H	DMA 触发因素寄存器 2	DTFR2		√	√		00H
FFFFF816H	DMA 触发因素寄存器 3	DTFR3		√	√		00H
FFFFF820H	省电模式寄存器	PSMR		√	√		00H
FFFFF828H	处理器时钟控制寄存器	PCC		√	√		03H
FFFFF82CH	PLL 控制寄存器	PLLCTL		√	√		01H
FFFFF870H	时钟监视器模式寄存器	CLM		√	√		00H
FFFFF888H	复位源标志寄存器	RESF		√	√		00H/10H/01H
FFFFF890H	低电压检测寄存器	LVIM		√	√		00H
FFFFF891H	低电压检测电平选择寄存器	LVIS			√		00H
FFFFFA00H	UARTA0 控制寄存器 0	UA0CTL0		√	√		10H
FFFFFA01H	UARTA0 控制寄存器 1	UA0CTL1			√		00H
FFFFFA02H	UARTA0 控制寄存器 2	UA0CTL2			√		FFH
FFFFFA03H	UARTA0 选项控制寄存器 0	UA0OPT0		√	√		14H
FFFFFA04H	UARTA0 状态寄存器	UA0STR		√	√		00H
FFFFFA06H	UARTA0 接收数据寄存器	UA0RX	R		√		FFH

(12/14)

地址	功能寄存器名称	符号	R/W	操作位数			复位后
				1	8	16	
FFFFFA07H	UARTA0 发送数据寄存器	UA0TX	R/W		√		FFH
FFFFFA10H	UARTA1 控制寄存器 0	UA1CTL0		√	√		10H
FFFFFA11H	UARTA1 控制寄存器 1	UA1CTL1			√		00H
FFFFFA12H	UARTA1 控制寄存器 2	UA1CTL2			√		FFH
FFFFFA13H	UARTA1 选项控制寄存器 0	UA1OPT0		√	√		14H
FFFFFA14H	UARTA1 状态寄存器	UA1STR		√	√		00H
FFFFFA16H	UARTA1 接收数据寄存器	UA1RX	R		√		FFH
FFFFFA17H	UARTA1 发送数据寄存器	UA1TX	R/W		√		FFH
FFFFFA20H	UARTA2 控制寄存器 0	UA2CTL0		√	√		10H
FFFFFA21H	UARTA2 控制寄存器 1	UA2CTL1			√		00H
FFFFFA22H	UARTA2 控制寄存器 2	UA2CTL2			√		FFH
FFFFFA23H	UARTA2 选项控制寄存器 0	UA2OPT0		√	√		14H
FFFFFA24H	UARTA2 状态寄存器	UA2STR	R/W	√	√		00H
FFFFFA26H	UARTA2 接收数据寄存器	UA2RX	R		√		FFH
FFFFFA27H	UARTA2 发送数据寄存器	UA2TX	R/W		√		FFH
FFFFFA40H	UARTB 控制寄存器 0	UBCTL0		√	√		10H
FFFFFA42H	UARTB 控制寄存器 2	UBCTL2				√	FFFFH
FFFFFA44H	UARTB 状态寄存器	UBSTR		√	√		00H
FFFFFA46H	UARTB 接收数据寄存器 AP	UBRXAP	R			√	00FFH
FFFFFA46H	UARTB 接收数据寄存器	UBRX			√		FFH
FFFFFA48H	UARTB 发送数据寄存器	UBTX	W		√		FFH
FFFFFA4AH	UARTB FIFO 控制寄存器 0	UBFIC0	R/W	√	√		00H
FFFFFA4BH	UARTB FIFO 控制寄存器 1	UBFIC1		√	√		00H
FFFFFA4CH	UARTB FIFO 控制寄存器 2	UBFIC2				√	0000H
FFFFFA4CH	UARTB FIFO 控制寄存器 2L	UBFIC2L			√		00H
FFFFFA4DH	UARTB FIFO 控制寄存器 2H	UBFIC2H			√		00H
FFFFFA4EH	UARTB FIFO 状态寄存器 0	UBFIS0	R		√		00H
FFFFFA4FH	UARTB FIFO 状态寄存器 1	UBFIS1			√		10H
FFFFFB00H	TAA3 控制寄存器 0	TAA3CTL0	R/W	√	√		00H
FFFFFB01H	TAA3 控制寄存器 1	TAA3CTL1		√	√		00H
FFFFFB02H	TAA3 I/O 控制寄存器 0	TAA3IOC0[注]		√	√		00H
FFFFFB03H	TAA3 I/O 控制寄存器 1	TAA3IOC1[注]		√	√		00H
FFFFFB04H	TAA3 I/O 控制寄存器 2	TAA3IOC2[注]		√	√		00H
FFFFFB05H	TAA3 选项寄存器 0	TAA3OPT0		√	√		00H
FFFFFB06H	TAA3 捕获/比较寄存器 0	TAA3CCR0				√	0000H
FFFFFB08H	TAA3 捕获/比较寄存器 1	TAA3CCR1				√	0000H
FFFFFB0AH	TAA3 计数器读取缓冲寄存器	TAA3CNT	R			√	0000H
FFFFFB20H	TAA4 控制寄存器 0	TAA4CTL0	R/W	√	√		00H
FFFFFB21H	TAA4 控制寄存器 1	TAA4CTL1		√	√		00H
FFFFFB22H	TAA4 I/O 控制寄存器 0	TAA4IOC0		√	√		00H
FFFFFB23H	TAA4 I/O 控制寄存器 1	TAA4IOC1		√	√		00H

注　　仅限 V850E/IG3

(13/14)

地址	功能寄存器名称	符号	R/W	操作位数			复位后
				1	8	16	
FFFFFB24H	TAA4 I/O 控制寄存器 2	TAA4IOC2	R/W	√	√		00H
FFFFFB25H	TAA4 选项寄存器 0	TAA4OPT0		√	√		00H
FFFFFB26H	TAA4 捕获/比较寄存器 0	TAA4CCR0				√	0000H
FFFFFB28H	TAA4 捕获/比较寄存器 1	TAA4CCR1				√	0000H
FFFFFB2AH	TAA4 计数器读取缓冲寄存器	TAA4CNT	R			√	0000H
FFFFFB40H	数字噪声消除 1 控制寄存器 2	TANFC2	R/W		√		00H
FFFFFB42H	数字噪声消除 1 控制寄存器 3	TANFC3[注]			√		00H
FFFFFB44H	数字噪声消除 1 控制寄存器 4	TANFC4			√		00H
FFFFFB80H	A/D 转换器 2 模式寄存器 0	AD2M0		√	√		00H
FFFFFB81H	A/D 转换器 2 模式寄存器 1	AD2M1		√	√		00H
FFFFFB82H	A/D 转换器 2 通道指定寄存器	AD2S		√	√		00H
FFFFFB90H	A/D2 转换结果寄存器 0	AD2CR0	R			√	0000H
FFFFFB91H	A/D2 转换结果寄存器 0H	AD2CR0H			√		00H
FFFFFB92H	A/D2 转换结果寄存器 1	AD2CR1				√	0000H
FFFFFB93H	A/D2 转换结果寄存器 1H	AD2CR1H			√		00H
FFFFFB94H	A/D2 转换结果寄存器 2	AD2CR2				√	0000H
FFFFFB95H	A/D2 转换结果寄存器 2H	AD2CR2H			√		00H
FFFFFB96H	A/D2 转换结果寄存器 3	AD2CR3				√	0000H
FFFFFB97H	A/D2 转换结果寄存器 3H	AD2CR3H			√		00H
FFFFFB98H	A/D2 转换结果寄存器 4	AD2CR4[注]				√	0000H
FFFFFB99H	A/D2 转换结果寄存器 4H	AD2CR4H[注]			√		00H
FFFFFB9AH	A/D2 转换结果寄存器 5	AD2CR5[注]				√	0000H
FFFFFB9BH	A/D2 转换结果寄存器 5H	AD2CR5H[注]			√		00H
FFFFFB9CH	A/D2 转换结果寄存器 6	AD2CR6[注]				√	0000H
FFFFFB9DH	A/D2 转换结果寄存器 6H	AD2CR6H[注]			√		00H
FFFFFB9EH	A/D2 转换结果寄存器 7	AD2CR7[注]				√	0000H
FFFFFB9FH	A/D2 转换结果寄存器 7H	AD2CR7H[注]			√		00H
FFFFFBB0H	端口 7 寄存器	P7	R	√	√		不确定
FFFFFBB8H	端口 7 模式控制寄存器	PMC7	R/W	√	√		00H
FFFFFC00H	外部中断下降沿指定寄存器 0	INTF0		√	√		00H
FFFFFC02H	外部中断下降沿指定寄存器 1	INTF1		√	√		00H
FFFFFC04H	外部中断下降沿指定寄存器 2	INTF2		√	√		00H
FFFFFC20H	外部中断上升沿指定寄存器 0	INTR0		√	√		00H
FFFFFC22H	外部中断上升沿指定寄存器 1	INTR1		√	√		00H
FFFFFC24H	外部中断上升沿指定寄存器 2	INTR2		√	√		00H
FFFFFC40H	上拉电阻选项寄存器 0	PU0		√	√		00H
FFFFFC42H	上拉电阻选项寄存器 1	PU1		√	√		00H
FFFFFC44H	上拉电阻选项寄存器 2	PU2		√	√		00H
FFFFFC46H	上拉电阻选项寄存器 3	PU3		√	√		00H
FFFFFC48H	上拉电阻选项寄存器 4	PU4		√	√		00H

注　　仅限 V850E/IG3

(14/14)

地址	功能寄存器名称	符号	R/W	操作位数			复位后
				1	8	16	
FFFFFC66H	端口 3 功能寄存器	PF3	R/W	√	√		00H
FFFFFD00H	CSIB0 控制寄存器 0	CB0CTL0		√	√		01H
FFFFFD01H	CSIB0 控制寄存器 1	CB0CTL1		√	√		00H
FFFFFD02H	CSIB0 控制寄存器 2	CB0CTL2			√		00H
FFFFFD03H	CSIB0 状态寄存器	CB0STR		√	√		00H
FFFFFD04H	CSIB0 接收数据寄存器	CB0RX	R			√	0000H
FFFFFD04H	CSIB0 接收数据寄存器 L	CB0RXL			√		00H
FFFFFD06H	CSIB0 发送数据寄存器	CB0TX	R/W			√	0000H
FFFFFD06H	CSIB0 发送数据寄存器 L	CB0TXL			√		00H
FFFFFD10H	CSIB1 控制寄存器 0	CB1CTL0		√	√		01H
FFFFFD11H	CSIB1 控制寄存器 1	CB1CTL1		√	√		00H
FFFFFD12H	CSIB1 控制寄存器 2	CB1CTL2			√		00H
FFFFFD13H	CSIB1 状态寄存器	CB1STR		√	√		00H
FFFFFD14H	CSIB1 接收数据寄存器	CB1RX	R			√	0000H
FFFFFD14H	CSIB1 接收数据寄存器 L	CB1RXL			√		00H
FFFFFD16H	CSIB1 发送数据寄存器	CB1TX	R/W			√	0000H
FFFFFD16H	CSIB1 发送数据寄存器 L	CB1TXL			√		00H
FFFFFD20H	CSIB2 控制寄存器 0	CB2CTL0		√	√		01H
FFFFFD21H	CSIB2 控制寄存器 1	CB2CTL1		√	√		00H
FFFFFD22H	CSIB2 控制寄存器 2	CB2CTL2			√		00H
FFFFFD23H	CSIB2 状态寄存器	CB2STR		√	√		00H
FFFFFD24H	CSIB2 接收数据寄存器	CB2RX	R			√	0000H
FFFFFD24H	CSIB2 接收数据寄存器 L	CB2RXL			√		00H
FFFFFD26H	CSIB2 发送数据寄存器	CB2TX	R/W			√	0000H
FFFFFD26H	CSIB2 发送数据寄存器 L	CB2TXL			√		00H
FFFFFD80H	IIC 移位寄存器 0	IIC0			√		00H
FFFFFD82H	IIC 控制寄存器 0	IICC0		√	√		00H
FFFFFD83H	从地址寄存器 0	SVA0			√		00H
FFFFFD84H	IIC 时钟选择寄存器 0	IICCL0		√	√		00H
FFFFFD85H	IIC 功能扩展寄存器 0	IICX0		√	√		00H
FFFFFD86H	IIC 状态寄存器 0	IICS0	R	√	√		00H
FFFFFD8AH	IIC 标志寄存器 0	IICF0	R/W	√	√		00H
FFFFFD90H	IICOPS 时钟选择寄存器	IICOCKS			√		00H
FFFFFF44H	上拉电阻选项寄存器 DL	PUDL				√	0000H
FFFFFF44H	上拉电阻选项寄存器 DLL	PUDLL		√	√		00H
FFFFFF45H	上拉电阻选项寄存器 DLH	PUDLH		√	√		00H

3.4.8　专用寄存器

专用寄存器是可防止程序挂起引起的数据非法写入的寄存器。V850E/IF3 和 V850E/IG3 具有以下五个专用寄存器。

- 节电控制寄存器（PSC）
- 处理器时钟控制寄存器 （PCC）
- 复位源标志寄存器（RESF）
- 时钟监控模式寄存器（CLM）
- 低电压检测寄存器（LVIM）

此外，还提供一个命令寄存器（PRCDM）来防止对专用寄存器的非法写操作，这样应用系统就不会因为程序死循环而意外停止工作。对专用寄存器的写操作需要特定流程实现，非法保存操作将报告给系统状态寄存器（SYS）。

（1） 向专用寄存器设置数据

按照以下流程向专用寄存器设置数据。

<1> 需要写入专用寄存器的数据放入通用寄存器中。
<2> 将步骤<1>准备好的数据写入命令寄存器中。
<3> 将设置数据写入专用寄存器中（使用如下指令完成）。
- 存储指令（ST/SST 指令）
- 位操作指令 （SET1/CLR1/NOT1 指令）。

（<4> 至 <8> 插入NOP指令（5 条指令））[注]
<9> 如果必要，则使能 DMA 操作。

[实例] 使用 PSC 寄存器（设置待机模式）

```
     ST.B r11， PSMR[r0]       ；设置 PSMR 寄存器（设置 IDLE1、IDLE2 和 STOP 模式）。
<1>MOV0x02， r10
<2>ST.B r10， PRCMD[r0]       ；写入 PRCMD 寄存器。
<3>ST.B r10， PSC[r0]         ；设置 PSC 寄存器。
<4>NOP注                       ；空操作伪指令
<5>NOP注                       ；空操作伪指令
<6>NOP注                       ；空操作伪指令
<7>NOP注                       ；空操作伪指令
<8>NOP注                       ；空操作伪指令
（下一条指令）
```

读取专用寄存器时没有特定流程。

注 设置 IDLE 模式或 STOP 模式（通过将 PSC.STB 位置 1）后，要立即插入 5 条或更多的 NOP 指令。

注意事项 1. **执行存储指令来向命令寄存器存储数据时，不响应中断。原因是假设上述步骤<2>和<3>是通过连续的存储指令执行。若在步骤<2>和<3>之间插入了其他指令，且该指令又引起了中断响应，则将打乱上述的顺序，导致错误操作。**

2. **虽然写入命令寄存器的是伪数据，仍然使用在设置专用寄存器（例中的步骤<3>）时所用的通用寄存器将数据，通过存储指令将数据写入至命令寄存器（例中的步骤<2>）。当通用寄存器用于寻址时，同样使用这种用法。**

下面是使用位操作指令向专用寄存器 (示例中为<3>) 设置数据的示例。

```
CLR1 4, RESF[r0]
```

3. **在执行该处理之前，终结所有的 DMA 传输操作。**

（2） 命令寄存器（PRCMD）

PRCMD 是 8 位寄存器，用于保护那些会严重影响应用系统的寄存器免受非法写入，从而使系统不会因为程序死循环而意外停止工作。预先准备的数据写入 PRCMD 寄存器后，对专用寄存器的第一个写操作有效。这样，专用寄存器的值仅能按照特定的流程进行写入，这样就可以防止对专用寄存器的非法写操作。

通过 SYS.PRERR 位可以检查到对专用寄存器的非法写入操作。

PRCMD 寄存器仅能按字节进行写入操作（若对其进行读取，则读到的数据不确定）。

复位使得该寄存器不确定。

复位后：不确定　　W　　地址：FFFFF1FCH

	7	6	5	4	3	2	1	0
PRCMD	REG7	REG6	REG5	REG4	REG3	REG2	REG1	REG0

（3） 系统状态寄存器（SYS）

该寄存器包含了整个系统的操作状态的状态标志。

如果该寄存器没有按照正确序列写入，包括对 PRCMD 寄存器的访问，数据不会被写入目标寄存器，产生一个保护错误，PRERR 标志被置位。从 CPU 通过指令向该寄存器写入“0”，则该寄存器被清零。

该寄存器可进行字节读写或者按位读写。

系统复位后，该寄存器的值被置为 00H。

复位后：00H R/W 地址：FFFFF802H

	7	6	5	4	3	2	1	<0>
SYS	0	0	0	0	0	0	0	PRERR

PRERR	检测到保护错误
0	未发生保护错误
1	已发生保护错误

PRERR 标志的操作按下列条件进行的。

（a） 设置条件（PRERR 标志=1）

（i） 当数据写入专用寄存器时，却没有任何内容写入 PRCMD 寄存器，（没有执行 **3.4.8（1） 设置数据至专用寄存器**的步骤 **<2>**， 就执行步骤 **<3>**）

（ii） 对 PRCMD 寄存器写入数据后，数据写入了片上周边 I/O 寄存器而没有写入专用寄存器（包括执行位操作指令）（如果 **3.4.8（1） 设置数据至专用寄存器**中的步骤<3>不是对专用寄存器的设置）

备注 即使在 PRCMD 寄存器的写入访问与专用寄存器的写入访问（比如访问内部 RAM）操作之间，读取片上外围 I/O 寄存器（执行位操作指令除外），PRERR 标志不会被置位，且数据可以写入专用寄存器。

（b） 清零条件（PRERR 标志=0）

（i） 当将 0 写入 SYS.PRERR 标志位时。

（ii） 系统被复位时。

注意事项 1. 如果在对 PRCMD 寄存器进行写访问之后，立即将 0 写入 SYS.PRERR 位（并非专用寄存器），那么 PRERR 位将被清除为零（写访问优先）。

2. 如果在对 PRCMD 寄存器进行写访问之后，立即将数据写入 PRCMD 寄存器（该寄存器并不是专用寄存器），那么 PRERR 位被设为 1。

3.4.9 系统等待控制寄存器（VSWC）

VSWC 寄存器用于控制访问片上外围 I/O 寄存器时的总线等待时间。
访问 V850E1 CPU 核的片上外围 I/O 寄存器，基本需要三个时钟。但是，在 V850E/IF3 和 V850E/IG3 中，在这 3 个时钟之外，还需要一个 VSWC 寄存器设置的等待。将 VSWC 设置为 13H（设置 4 个时钟的等待）。
VSWC 寄存器按字节进行读写（地址： FFFFF06EH，初始值：77H）。

CPU 时钟频率（f_{CPU}）	VSWC 的设置值
500 kHz ≤ f_{CPU} < 64 MHz	13 H

注意事项 **使用 V850E/IF3 和 V850E/IG3 时，必须首先设置 VSWC 寄存器。**
在设置 VSWC 寄存器之后，按需设置其他寄存器。

备注　当访问一个包括表示片上外围功能状态 (比如 UAnSTR)的状态标志的寄存器，或表示定时器的计数值 (比如 TAAnCNT) 的寄存器时，如果标志和计数值改变的时序和寄存器访问的时序重叠，会发生寄存器访问重试操作。因此，访问片上外围 I/O 寄存器可能会需要较长时间。

4.1　特性

4.1.1　V850E/IF3

○ 只能输入的端口：　4
　I/O 端口：　44
○ 可按位指定输入数据读取/输出数据写入
○ 可按位连接片内上拉电阻(仅限端口 0 至 4 和 DL)
　但是，当引脚是端口模式且为输入模式时或当引脚用作复用功能的输入引脚时，才可以连接片内上拉电阻。并且，片内上拉电阻可以连接到 TOB0T1 至 TOB0T3、TOB0B1 至 TOB0B3 和 TOA21 引脚，这些是复用功能模式下的输出引脚，当这些引脚由于 TOB0OFF、TOA2OFF 或 TOB1OFF 引脚或软件处理变为高阻状态时，可以连接片内上拉电阻。

4.1.2　V850E/IG3

○ 只能输入的端口：　8
　I/O 端口：　56
○ 可按位指定输入数据读取/输出数据写入
○ 可按位连接片内上拉电阻(仅限端口 0 至 4 和 DL)
　但是，当引脚是端口模式且为输入模式时或当引脚用作复用功能的输入引脚时，才可以连接片内上拉电阻。并且，片内上拉电阻可以连接到 TOB0T1 至 TOB0T3、TOB0B1 至 TOB0B3、TOA21、TOB1T1 至 TOB1T3、TOB1B1 至 TOB1B3 和 TOA31 引脚，这些是复用功能模式下的输出引脚，当这些引脚由于 TOB0OFF、TOB1OFF、TOA2OFF 或 TOA3OFF 引脚或软件处理变为高阻状态时，可以连接片内上拉电阻。

4.2　端口配置

4.2.1　V850E/IF3

V850E/IF3 总共内置 48 个输入/输出端口 (包括 4 个只能作为输入的端口) 端口名为 0 至 4，7 和 DL。端口配置如图 4-1 所示。

引脚的 I/O 缓冲有两种供电电源系统：AV_{DD2} 和 EV_{DD0}，EV_{DD1}。供电电源和引脚之间的关系如表 4-1 所示。

图 4-1. 端口配置 (V850E/IF3)

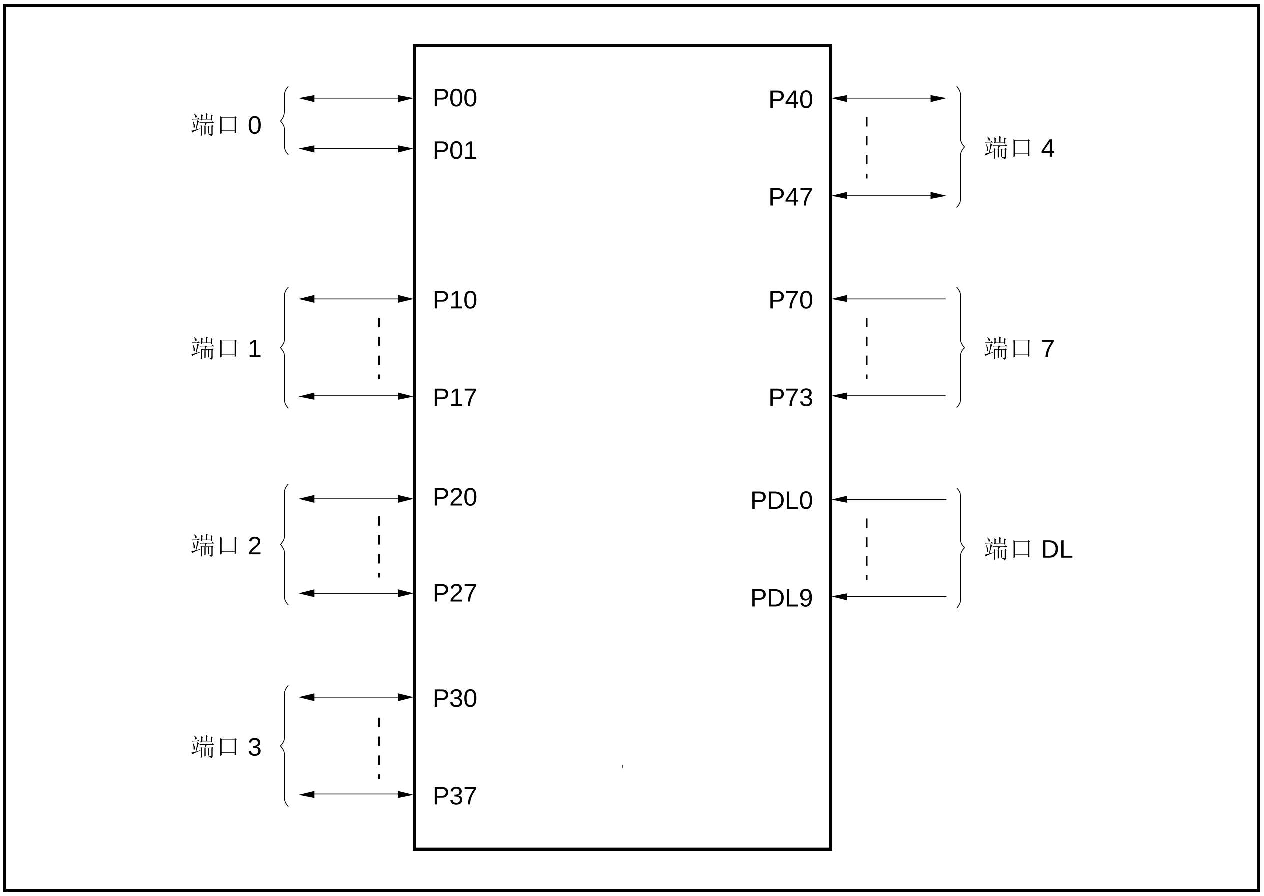


表 4-1. 各个引脚 I/O 缓冲器的供电电源 (V850E/IF3)

供电电源	对应引脚
AV_{DD2}	P70 至 P73
EV_{DD0}, EV_{DD1}	P00, P01, P10 至 P17, P20 至 P27, P30 至 P37, P40 至 P47, PDL0 至 PDL9, $\overline{RESET}$

4.2.2 V850E/IG3

V850E/IG3 总共内置 64 个输入/输出端口 (包括 8 个只能作为输入的端口) 端口名为 0 至 4，7 和 DL。端口配置如图 4-2 所示。

引脚的 I/O 缓冲有两种供电电源系统：AV_{DD2} 和 EV_{DD0}，EV_{DD1}，EV_{DD2}。供电电源和引脚之间的关系如表 4-2 所示。

图 4-2. 端口配置 (V850E/IG3)

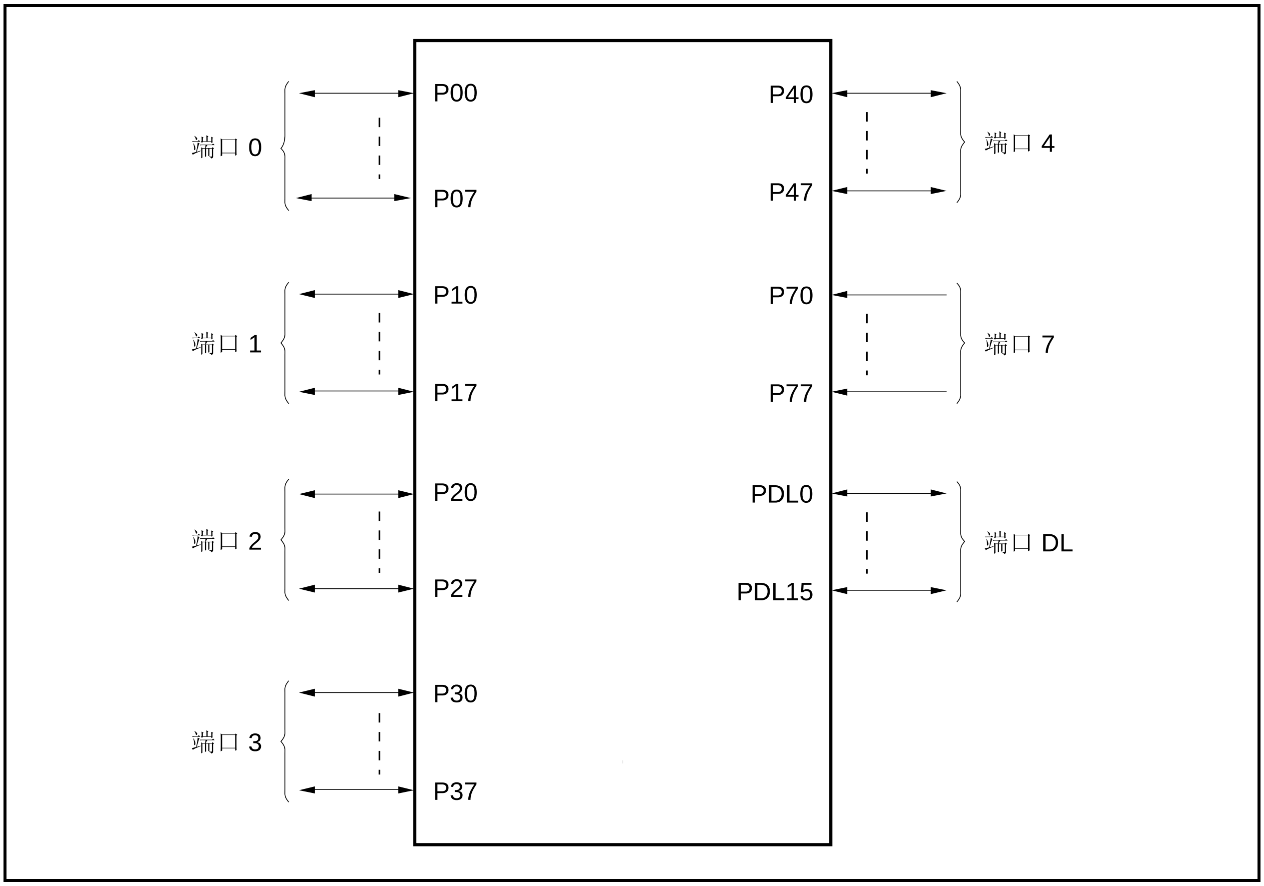


表 4-2. 各个引脚 I/O 缓冲器的供电电源 (V850E/IG3)

供电电源	对应引脚
AV_{DD2}	P70 至 P77
EV_{DD0}, EV_{DD1}, EV_{DD2}	P00 至 P07, P10 至 P17, P20 至 P27, P30 至 P37, P40 至 P47, PDL0 至 PDL15, $\overline{\text{RESET}}$, DCK, DDI, DDO, DMS, $\overline{\text{DRST}}$

4.3　端口配置

表 4-3. 端口配置 (V850E/IF3)

项目	配置
控制寄存器	端口 n 寄存器 (Pn：n = 0 至 4, 7, DL) 端口 n 模式寄存器 (PMn：n = 0 至 4, DL) 端口 n 模式控制寄存器 (PMCn：n = 0 至 4, 7, DL) 端口 n 功能控制寄存器 (PFCn：n = 0 至 4) 端口 n 功能控制扩展寄存器 (PFCEn：n = 0 至 4) 上拉电阻选项寄存器 (PUn：n = 0 至 4, DL) 端口 3 功能寄存器 (PF3)
端口	仅输入：4, I/O：44
上拉电阻	软件控制：44

表 4-4. 端口配置 (V850E/IG3)

项目	配置
控制寄存器	端口 n 寄存器 (Pn：n = 0 至 4, 7, DL) 端口 n 模式寄存器 (PMn：n = 0 至 4, DL) 端口 n 模式控制寄存器 (PMCn：n = 0 至 4, 7, DL) 端口 n 功能控制寄存器 (PFCn：n = 0 至 4) 端口 n 功能控制扩展寄存器 (PFCEn：n = 0 至 4) 上拉电阻选项寄存器 (PUn：n = 0 至 4, DL) 端口 3 功能寄存器 (PF3)
端口	仅输入：8，I/O：56
上拉电阻	软件控制：56

（1）端口 n 寄存器 （Pn）

通过写入或读取 Pn 寄存器，从外部器件输入数据，或输出数据到外部器件。

Pn 寄存器由保存输出数据的端口锁存器和读取引脚状态的电路组成。

Pn 寄存器的每一位对应端口的一个引脚，而且可以按位读出或写入。

复位后：不确定　　R/W

	7	6	5	7	3	2	1	0
Pn	Pn7	Pn6	Pn5	Pn4	Pn3	Pn2	Pn1	Pn0

Pnm	输出数据的控制(在输出模式下)
0	输出0
1	输出1

无论 PMCn 寄存器如何设置，对 Pn 寄存器的数据写入或读取如下所示：

表 4-5. Pn 寄存器的写入/读出

寄存器 PMn 的设置	写入 Pn 寄存器	读取 Pn 寄存器
输出模式 （PMnm 位 = 0）	数据写入输出锁存器[注] 端口模式下（PMCn = 0），输出锁存器的内容从相应引脚输出	读取输出锁存器的值
输入模式 （PMnm = 1）	数据写入输出锁存器 引脚状态不受影响[注]	读取引脚状态

注　**1.** 写入到输出锁存器的值会一直保留，直到有新的数值写入输出锁存器。

2. 同样，设置复用功能时，当 PMn 寄存器处于输出模式，读取的是 Pn 寄存器的值。

3. 设置复用功能时，如果 PMn 寄存器处于输入模式，不论复用功能是输入或输出功能，读取的是引脚此时的状态。

（2）端口 n 模式寄存器 （PMn）

PMn 寄存器用于指定对应端口引脚的输入或输出模式。

该寄存器的每一位对应端口 n 的一个引脚，而且可以按位设定为输入或输出模式。

复位后: FFH　　R/W

	7	6	5	4	3	2	1	0
PMn	PMn7	PMn6	PMn5	PMn4	PMn3	PMn2	PMn1	PMn0

PMnm	输入/输出模式控制
0	输出模式
1	输入模式

（3）端口 n 模式控制寄存器（PMCn）

PMCn 寄存器指定端口模式或复用功能模式。

该寄存器的每一位对应端口 n 的一个引脚，而且可以按位设定端口的工作模式。

复位后: 00H　　R/W

	7	6	5	4	3	2	1	0
PMCn	PMCn7	PMCn6	PMCn5	PMCn4	PMCn3	PMCn2	PMCn1	PMCn0

PMCnm	操作模式说明
0	端口模式
1	复用功能模式

（4）端口 n 功能 控制寄存器（PFCn）

如果引脚有两个复用功能，PFCn 寄存器用来指定需要使用的引脚复用功能。

该寄存器的每一位对应端口 n 的一个引脚，而且可以按位设定端口引脚的复用功能。

复位后: 00H　　R/W

	7	6	5	4	3	2	1	0
PFCn	PFCn7	PFCn6	PFCn5	PFCn4	PFCn3	PFCn2	PFCn1	PFCn0

PFCnm	复用功能说明
0	复用功能 1
1	复用功能 2

（**5**）端口 **n** 功能控制扩展寄存器 （**PFCEn**）

如果引脚有三个或更多的复用功能，PFCEn 寄存器用来指定需要使用的引脚复用功能。

该寄存器的每一位对应端口 n 的一个引脚，而且可以按位设定端口引脚的复用功能。

复位后: 00H　　R/W

	7	6	5	4	3	2	1	0
PFCEn	PFCEn7	PFCEn6	PFCEn5	PFCEn4	PFCEn3	PFCEn2	PFCEn1	PFCEn0

	7	6	5	4	3	2	1	0
PFCn	PFCn7	PFCn6	PFCn5	PFCn4	PFCn3	PFCn2	PFCn1	PFCn0

PFCEnm	PFCnm	复用功能说明
0	0	复用功能 1
0	1	复用功能 2
1	0	复用功能 3
1	1	复用功能 4

（**6**）上拉电阻选项寄存器（**PUn**）

PUn 寄存器指定片内上拉电阻的连接。

上拉电阻选项寄存器的每一位对应端口 n 的一个引脚，而且可以按位设定。

复位后： 00H　　R/W

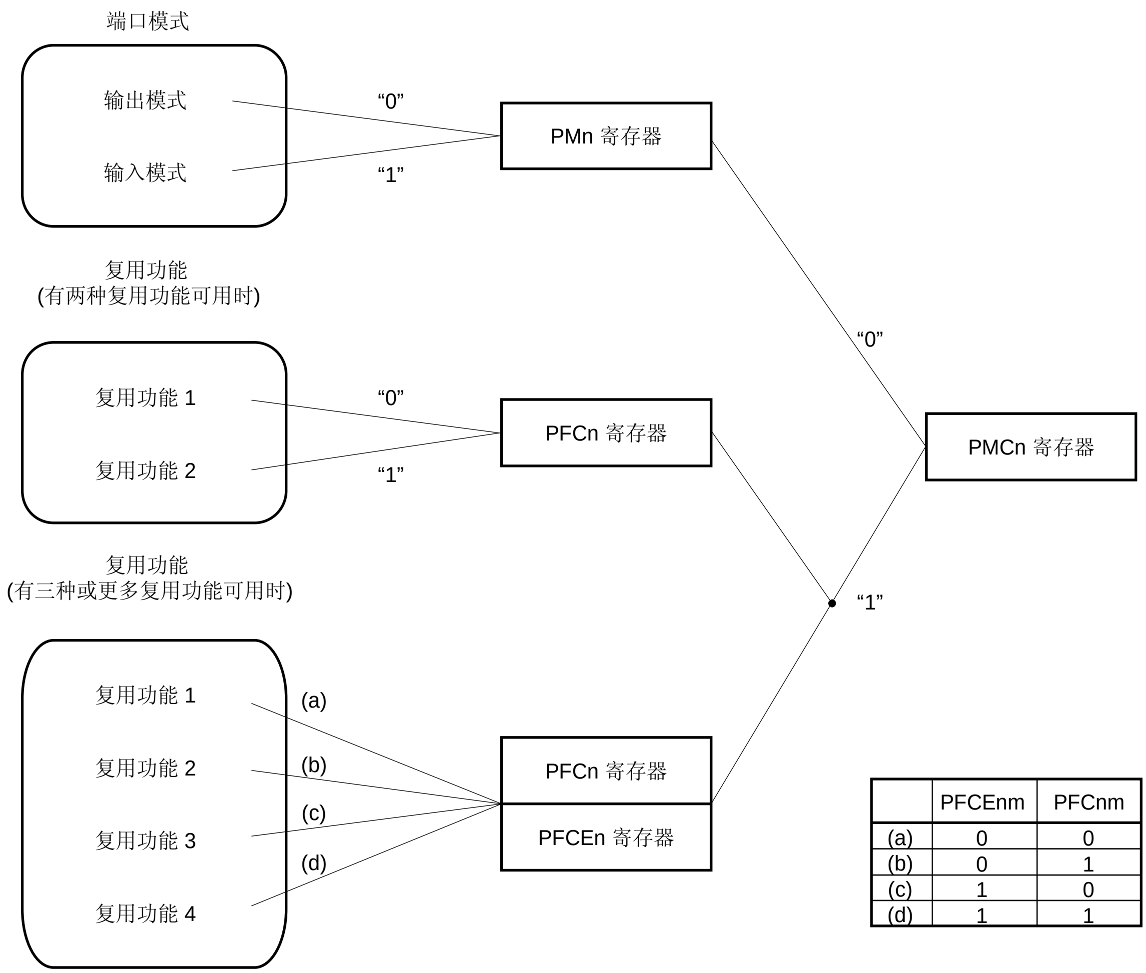
	7	6	5	4	3	2	1	0
PUn	PUn7	PUn6	PUn5	PUn4	PUn3	PUn2	PUn1	PUn0

PUnm	片内上拉电阻连接的控制
0	未连接
1	已连接

（7）端口设置

端口设置如下所示：

图 4-3. 寄存器的设置和引脚功能



	PFCEnm	PFCnm
(a)	0	0
(b)	0	1
(c)	1	0
(d)	1	1

注意事项　要从端口模式切换到外部中断输入**(INTPn) (**通过将 **PMCa.PMCam** 位从 **0** 改变为 **1)**，如果检测到一个错误有效边沿，可能会输入一个外部中断。因此，请确保通过 **INTRk**，**INTFk**，**ADTR** 或 **ADTF** 寄存器设置为**“**无边沿检测**”**，选择外部中断输入**(INTPn)**，然后指定有效边沿**(V850E/IF3**：**n = 00, 01, 08** 至 **18, ADT0, ADT1, a = 0** 至 **4, m = 0** 至 **7, k = 0** 至 **2, V850E/IG3**：**n = 00** 至 **18, ADT0, ADT1, a = 0** 至 **4, m = 0** 至 **7, k = 0** 至 **2)).**

从外部中断输入**(INTPn)**切换到端口模式时**(**通过将 **PMCam** 位从 **1** 改变为 **0)**，可能会检测到一个边沿。因此，请确保通过 **INTRk**，**INTFk**，**ADTR** 或 **ADTF** 寄存器设置为**“**无边沿检测**”**，然后设置端口模式。

备注　按下列步骤切换到复用功能（关于 n，参加表 **4-3** 和表 **4-4**）。

<1> 设置 PFCn 和 PFCEn 寄存器。

<2> 设置 PFCn 寄存器。

<3> 设置 INTRk，INTFk，ADTR 和 ADTF 寄存器（当设置外部中断引脚时）。

如果在设置 PFCn 和 PFCEn 寄存器之前，先设置 PMCn 寄存器，则在设置 PFCn 和 PFCEn 寄存器时可以会选择预期之外的外围功能。

4.3.1 端口 0

端口 0 可以按位设置为输入或输出模式。

各个产品的 I/O 引脚数量会有不同。

名称	I/O 端口的数量
V850E/IF3	2 位 I/O 端口
V850E/IG3	8 位 I/O 端口

端口 0 包括以下复用功能引脚：

<R>

表 4-6. 端口 0 复用功能引脚

引脚名称	引脚编号				复用功能引脚名称	I/O	上拉[注1]
	IF3	IG3					
	GC	GC	GF	F1			
P00	70	91	19	A6	TOA20/TIA20/TOA2OFF/INTP00	I/O	提供
P01	69	90	18	B6	TOA21/TIA21/INTP01	I/O	
P02[注2]	–	89	17	C6	TOA30[注2]/TIA30[注2]/TOA3OFF[注2]/INTP02[注2]	I/O	
P03[注2]	–	88	16	C7	TOA31[注2]/TIA31[注2]/INTP03[注2]	I/O	
P04[注2]	–	84	12	C8	TECR0[注2]/TIT00[注2]/TOT00[注2]/INTP04[注2]	I/O	
P05[注2]	–	83	11	C9	TENC00[注2]/EVTT0[注2]/INTP05[注2]	输入	
P06[注2]	–	82	10	C10	TENC01[注2]/TIT01[注2]/TOT01[注2]/INTP06[注2]	I/O	
P07[注2]	–	63	91	C12	INTP07[注2]/CLKOUT[注3]	I/O	

备注 IF3：V850E/IF3
IG3：V850E/IG3
GC（V850E/IF3）: 80 针塑封 LQFP（14 × 14）
GC（V850E/IG3）: 100 针塑封 LQFP（密间距）（14 × 14）
<R> GF（V850E/IG3）: 100 针塑封 LQFP（14 × 20）
F1（V850E/IG3）: 161 针塑封 FBGA（10 × 10）

注 **1.** 软件上拉功能
2. 仅限 V850E/IG3
<R> **3.** 仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

注意事项 **1.** 要控制用于电机控制的定时器的高阻抗输出。请确保设置 **PMC0.PMC0n** 位为 **1**，然后指定要检测的边沿，并使能高阻抗输出控制器的操作，因为如果检测到错误有效边沿，电机控制定时器的输出可能会变为高阻抗状态。**(V850E/IF3：n = 0, V850E/IG3：n = 0, 2).**

2. 当 **P01** 和 **P03(**仅限 **V850E/IG3)**被用作 **TOA21** 和 **TOA31 (**仅限 **V850E/IG3)**时，通过输入下列有效信号，它们变为高阻抗状态。

- 从高阻抗输出控制器输出的高阻抗设置信号
- 从时钟监视器输出的时钟停止检测信号

注意事项 **3.** 要从端口模式切换到外部中断输入**(INTP0n)** (通过将 **PMC0.PMC0n** 位从 **0** 改变为 **1)**，如果检测到一个错误有效边沿，可能会输入一个外部中断。因此，请确保禁止边沿检测（**INTF0.INTF0n** 位 = **0** 和 **INTR0.INTR0n** 位 = **0**），选择外部中断输入**(INTP0n)**，然后指定有效边沿**(V850E/IF3：n = 0, 1，V850E/IG3：n = 0** 至 **7)**。

从外部中断输入**(INTP0n)**切换到端口模式时**(**通过将 **PMC0n** 位从 **1** 改变为 **0)**，可能会检测到一个边沿。因此，请确保禁止边沿检测**(INTF0n** 位 **= 0, INTR0n** 位 **= 0)**，然后选择端口模式。

4. 要控制外部中断功能和电机输出控制功能的高阻抗输出，将 **PMC0n** 位设置为 **1 (V850E/IF3：n = 0, 1，V850E/IG3：n = 0** 至 **7)**。

(1) 寄存器

(a) 端口 0 寄存器（P0）

复位后： 不确定　　R/W　　地址： FFFFF400H

	7	6	5	4	3	2	1	0
P0	P07注	P06注	P05注	P04注	P03注	P02注	P01	P00

P0n	输出数据控制 (输出模式下)
0	输出 0.
1	输出 1.

注　仅对 V850E/IG3 有效
使用 V850E/IF3，该寄存器的读取值不确定。

备注　V850E/IF3：n = 0, 1
V850E/IG3：n = 0 至 7

(b) 端口 0 模式寄存器 （PM0）

复位后： FFH　　R/W　　地址： FFFFF420H

	7	6	5	4	3	2	1	0
PM0	PM07注	PM06注	PM05注	PM04注	PM03注	PM02注	PM01	PM00

PM0n	输入/输出模式控制 (端口模式下)
0	输出模式
1	输入模式

注　仅对 V850E/IG3 有效
使用 V850E/IF3，请确保设置该位为 1。

备注　V850E/IF3：n = 0, 1
V850E/IG3：n = 0 至 7

(c) 端口 0 模式控制寄存器（PMC0）

复位后： 00H R/W 地址： FFFFF440H

	7	6	5	4	3	2	1	0
PMC0	PMC07注 1	PMC06注 1	PMC05注 1	PMC04注 1	PMC03注 1	PMC02注 1	PMC01	PMC00

PMC07注1	P07 引脚操作模式说明
0	I/O 端口
1	INTP07 输入注2/CLKOUT 输出注3

PMC06注 1	P06 引脚操作模式说明
0	I/O 端口
1	TENC01 输入注 2/TIT01输入注 2/TOT01 输出注 2/INTP06 输入注 1

PMC05注 1	P05 引脚操作模式说明
0	I/O 端口
1	TENC00 输入注 2/EVTT0 输入注2/INTP05 输入注 2

PMC04注 1	P04 引脚操作模式说明
0	I/O 端口
1	TECR0 输入注 2/TIT00 输入注 2/TOT00 输出注 2/INTP04 输入注 2

PMC03注 1	P03 引脚操作模式说明
0	I/O 端口
1	TOA31 输出注 2/TIA31 输入注 2/INTP03 输入注 2

PMC02注1	P02 引脚操作模式说明
0	I/O 端口
1	TOA30 输出注 2/TIA30 输入注 2/TOA3OFF 输入注 2/INTP02 输入注2

PMC01	P01 引脚操作模式说明
0	I/O 端口
1	TOA21 输出/TIA21 输入/INTP01 输入

PMC00	P00 引脚操作模式说明
0	I/O 端口
1	TOA20 输出/TIA20 输入/TOA2OFF输入/INTP00 输入

注 1. 仅对 V850E/IG3 有效
使用 V850E/IF3，请确保设置该位为 0。
2. 仅限 V850E/IG3
<R> 3. 仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

(d) 端口 0 功能控制寄存器（PFC0）

复位后： 00H　R/W　地址： FFFFF460H

	7	6	5	4	3	2	1	0
PFC0	PFC07注	PFC06注	PFC05注	PFC04注	PFC03注	PFC02注	PFC01	PFC00

注　仅对 V850E/IG3 有效
使用 V850E/IF3，请确保设置该位为 0。

备注　复用功能的规范，参见 **4.3.1 (1) (f) 端口 0 复用功能的设置。**

(e) 端口 0 功能扩展寄存器 (PFCE0)

复位后： 00H　R/W　地址：FFFFF700H

	7	6	5	4	3	2	1	0
PFCE0	0	PFCE06注	PFCE05注	PFCE04注	PFCE03注	PFCE02注	PFCE01	PFCE00

注　仅对 V850E/IG3 有效
使用 V850E/IF3，请确保设置该位为 0。

备注　复用功能的规范，参见 **4.3.1 (1) (f) 端口 0 复用功能的设置。**

(f)　端口 0 复用功能的设置

PFC07[注1]	P07[注1] 引脚的复用功能说明
0	INTP07 输入[注1]
1	CLKOUT 输出[注2]

PFCE06[注1]	PFC06[注1]	P06[注1] 引脚的复用功能说明
0	0	TENC01 输入[注1]/TIT01 输入[注1] (两个功能交替使用)
0	1	TOT01 输出[注1]
1	0	INTP06 输入[注1]
1	1	禁止设置

PFCE05[注1]	PFC05[注1]	P05[注1] 引脚的复用功能说明
0	0	TENC00 输入[注1]
0	1	EVTT0 输入[注1]
1	0	INTP05 输入[注1]
1	1	禁止设置

PFCE04[注1]	PFC04[注1]	P04[注1] 引脚的复用功能说明
0	0	TECR0 输入[注1]/TIT00 输入[注1] (两个功能交替使用)
0	1	TOT00 输出[注1]
1	0	INTP04 输入[注1]
1	1	禁止设置

PFCE03[注1]	PFC03[注1]	P03[注1] 引脚的复用功能说明
0	0	TOA31 输出[注1]
0	1	TIA31 输入[注1]
1	0	INTP03 输入[注1]
1	1	禁止设置

PFCE02[注1]	PFC02[注1]	P02[注1] 引脚的复用功能说明
0	0	TOA30 输出[注1]
0	1	TIA30 输入[注1]
1	0	TOA3OFF 输入[注1]/INTP02 输入[注1] (两个功能交替使用)
1	1	禁止设置

注 1.　仅限 V850E/IG3

<R>　2.　仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

PFCE01	PFC01	P01 引脚的复用功能说明
0	0	TOA21 输出
0	1	TIA21 输入
1	0	INTP01 输入
1	1	禁止设置

PFCE00	PFC00	P00 引脚的复用功能说明
0	0	TOA20 输出
0	1	TIA20 输入
1	0	TOA2OFF 输入/INTP00 输入
1	1	禁止设置

(g) 上拉电阻选项寄存器 0 (PU0)

复位后： 00H　　R/W　　地址： FFFFFC40H

	7	6	5	4	3	2	1	0
PU0	PU07注 1	PU06注1	PU05注 1	PU04注 1	PU03注1	PU02注 1	PU01	PU00

PU0n	片内上拉电阻连接的控制
0	未连接
1	已连接注 2

注　**1.** 仅对 V850E/IG3 有效
使用 V850E/IF3，请确保设置该位为 0。

2. 当引脚是端口模式且为输入模式时或当引脚用作复用功能的输入引脚时，才可以连接片内上拉电阻。并且，片内上拉电阻可以连接到 TOA21 和 TOA31(仅限 V850E/IG3)引脚，这些是复用功能模式下的输出引脚，当这些引脚由于 TOA2OFF 或 TOA3OFF(仅限 V850E/IG3)引脚或软件处理变为高阻状态时，可以连接片内上拉电阻。当引脚处于输出模式时，片内上拉电阻不能连接。

备注　V850E/IF3：n = 0, 1
V850E/IG3：n = 0 至 7

4.3.2　端口 1

端口 1 可以按位设置为输入或输出模式。

端口 1 包括以下复用功能引脚：

<R>　表 4-7. 端口 1 的复用功能引脚

引脚名称	引脚编号				复用功能引脚名称	I/O	上拉[注 1]
	IF3	IG3					
	GC	GC	GF	F1			
P10	78	99	27	B3	TOB0T1/TIB01/TOB01/A0[注 2]	I/O	提供
P11	77	98	26	C3	TOB0B1/TIB02/TOB02/A1[注 2]	I/O	
P12	76	97	25	A4	TOB0T2/TIB03/TOB03/A2[注 2]	I/O	
P13	75	96	24	B4	TOB0B2/TIB00/A3[注 2]	I/O	
P14	74	95	23	C4	TOB0T3/EVTB0/A4[注 2]	I/O	
P15	73	94	22	A5	TOB0B3/TRGB0/A5[注 2]	I/O	
P16	72	93	21	B5	TOB0OFF/INTP08/ADTRG0/INTADT0/A6[注 2]	I/O	
P17	71	92	20	C5	TOB00/INTP09/A7[注 2]	I/O	

注　**1.** 软件上拉功能

<R>　**2.** 仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

注意事项　当 **P10** 至 **P15** 被用作 **TOB0T1** 至 **TOB0T3** 和 **TOB0B1** 至 **TOB0B3** 时，通过输入下列有效信号，它们变为高阻抗状态。

- 从高阻抗输出控制器输出的高阻抗设置信号
- 从时钟监视器输出的时钟停止检测信号

备注　**IF3**：**V850E/IF3**

IG3：**V850E/IG3**

GC（**V850E/IF3**）：　**80** 针塑封 **LQFP**（**14** × **14**）

GC（**V850E/IG3**）：　**100** 针塑封 **LQFP**（密间距）（**14** × **14**）

<R>　**GF**（**V850E/IG3**）：　**100** 针塑封 **LQFP**（**14** × **20**）

F1（**V850E/IG3**）：　**161** 针塑封 **FBGA**（**10** × **10**）

(1) 寄存器

(a) 端口 1 寄存器 (P1)

复位后： 不确定　R/W　地址： FFFFF402H

	7	6	5	4	3	2	1	0
P1	P17	P16	P15	P14	P13	P12	P11	P10

P1n	输出数据控制 (输出模式下)
0	输出 0.
1	输出 1.

备注　n = 0 至 7

（b） 端口 1 模式寄存器 （PM1）

复位后： FFH　R/W　地址： FFFFF422H

	7	6	5	4	3	2	1	0
PM1	PM17	PM16	PM15	PM14	PM13	PM12	PM11	PM10

PM1n	输入/输出模式控制 (端口模式下)
0	输出模式
1	输入模式

备注　n = 0 至 7

(c) 端口 1 模式控制寄存器 (PMC1)

复位后： 00H R/W 地址： FFFFF442H

	7	6	5	4	3	2	1	0
PMC1	PMC17	PMC16	PMC15	PMC14	PMC13	PMC12	PMC11	PMC10

PMC17	P17 引脚操作模式说明
0	I/O 端口
1	TOB00 输出/INTP09 输入/A7 输出[注]

PMC16	P16 引脚操作模式说明
0	I/O 端口
1	TOB0OFF 输入/INTP08 输入/ADTRG0 输入/INTADT0 输入/A6 输出[注]

PMC15	P15 引脚操作模式说明
0	I/O 端口
1	TOB0B3 输出/TRGB0 输入/A5 输出[注]

PMC14	P14 引脚操作模式说明
0	I/O 端口
1	TOB0T3 输出/EVTB0 输入/A4 输出[注]

PMC13	P13 引脚操作模式说明
0	I/O 端口
1	TOB0B2 输出/TIB00 输入/A3 输出[注]

PMC12	P12 引脚操作模式说明
0	I/O 端口
1	TOB0T2 输出/TIB03 输入/TOB03 输出/A2 输出[注]

PMC11	P11 引脚操作模式说明
0	I/O 端口
1	TOB0B1 输出/TIB02 输入/TOB02 输出/A1 输出[注]

PMC10	P10 引脚操作模式说明
0	I/O 端口
1	TOB0T1 输出/TIB01 输入/TOB01输出/A0 输出[注]

<R> 注 仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

(d) 端口 **1** 功能控制寄存器 **(PFC1)**

复位后：　00H　　R/W　　地址：　FFFFF462H

	7	6	5	4	3	2	1	0
PFC1	PFC17	PFC16	PFC15	PFC14	PFC13	PFC12	PFC11	PFC10

备注　　复用功能的规范，参见 **4.3.2 (1) (f)** 端口 **1** 复用功能的设置。

(e) 端口 **1** 功能控制扩展寄存器 **(PFCE1)**

复位后：　00H　　R/W　　地址：　FFFFF702H

	7	6	5	4	3	2	1	0
PFCE1	PFCE17	PFCE16	PFCE15	PFCE14	PFCE13	PFCE12	PFCE11	PFCE10

备注　　复用功能的规范，参见 **4.3.2 (1) (f)** 端口 **1** 复用功能的设置。

(f)　端口 1 复用功能的设置

PFCE17	PFC17	P17 引脚的复用功能说明
0	0	TOB00 输出
0	1	INTP09 输入
1	0	A7 输出[注]
1	1	禁止设置

PFCE16	PFC16	P16 引脚的复用功能说明
0	0	TOB0OFF 输入/INTP08 输入 (两个功能交替使用)
0	1	ADTRG0 输入/INTADT0 输入 (两个功能交替使用)
1	0	A6 输出[注]
1	1	禁止设置

PFCE15	PFC15	P15 引脚的复用功能说明
0	0	TOB0B3 输出
0	1	TRGB0 输入
1	0	A5 输出[注]
1	1	禁止设置

PFCE14	PFC14	P14 引脚的复用功能说明
0	0	TOB0T3 输出
0	1	EVTB0 输入
1	0	A4 输出[注]
1	1	禁止设置

PFCE13	PFC13	P13 引脚的复用功能说明
0	0	TOB0B2 输出
0	1	TIB00 输入
1	0	A3 输出[注]
1	1	禁止设置

PFCE12	PFC12	P12 引脚的复用功能说明
0	0	TOB0T2 输出
0	1	TIB03 输入
1	0	TOB03 输出
1	1	A2 输出[注]

<R>　注　仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

PFCE11	PFC11	P11 引脚的复用功能说明
0	0	TOB0B1 输出
0	1	TIB02 输入
1	0	TOB02 输出
1	1	A1 输出[注]

PFCE10	PFC10	P10 引脚的复用功能说明
0	0	TOB0T1 输出
0	1	TIB01 输入
1	0	TOB01 输出
1	1	A0 输出[注]

<R>　注　仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

(g) 上拉电阻选项寄存器 1 (PU1)

复位后：　00H　R/W　地址：　FFFFFC42H

	7	6	5	4	3	2	1	0
PU1	PU17	PU16	PU15	PU14	PU13	PU12	PU11	PU10

PU1n	片内上拉电阻连接的控制
0	未连接
1	已连接[注]

注　当引脚是端口模式且为输入模式时或当引脚用作复用功能的输入引脚时，才可以连接片内上拉电阻。并且，片内上拉电阻可以连接到 TOB0T1 至 TOB0T3 和 TOB0B1 至 TOB0B3 引脚，这些是复用功能模式下的输出引脚，当这些引脚由于 TOB0OFF 引脚或软件处理变为高阻状态时，可以连接片内上拉电阻。当引脚处于输出模式时，片内上拉电阻不能连接。

备注　n = 0 至 7

4.3.3　端口 2

端口 2 可以按位设置为输入或输出模式。

端口 2 包括以下复用功能引脚。

<R>　**表 4-8. 端口 2 的复用功能引脚**

引脚名称	引脚编号				复用功能引脚名称	I/O	上拉注 1
	IF3	IG3					
	GC	GC	GF	F1			
P20	23	28	56	P3	TOB1T1/TIB11/TOB11	I/O	提供
P21	24	29	57	N3	TOB1B1/TIB12/TOB12	I/O	
P22	25	30	58	P4	TOB1T2/TIB13/TOB13	I/O	
P23	26	31	59	N4	TOB1B2/TIB10	I/O	
P24	27	32	60	M4	TOB1T3/EVTB1	I/O	
P25	28	33	61	M5	TOB1B3/TRGB1	I/O	
P26	29	34	62	M6	TOB10/TOB1OFF/INTP10/ADTRG1/INTADT1	I/O	
P27	36	45	73	M11	DMS注 2, 3	输入	

注　**1.** 软件上拉功能

2. 仅限 V850E/IG3

3. P27 引脚也可以作为片上调试引脚。通过 $\overline{\text{DRST}}$ 引脚的电平可以选择片上调试功能或端口功能(包括复用功能)，如下表所示。

端口 2 功能	
低电平输入到 $\overline{\text{DRST}}$ 引脚	高电平输入到 $\overline{\text{DRST}}$ 引脚
P27	DMS

注意事项　当 P20 至 P25 被用作 TOB1T1 至 TOB1T3 和 TOB1B1 至 TOB1B3 时，通过输入下列有效信号，它们变为高阻抗状态。

- **从高阻抗输出控制器输出的高阻抗设置信号**
- **从时钟监视器输出的时钟停止检测信号**

备注　**IF3：V850E/IF3**

IG3：V850E/IG3

GC（V850E/IF3）：　80 针塑封 LQFP（14 × 14）

GC（V850E/IG3）：　100 针塑封 LQFP（密间距）（14 × 14）

<R>　**GF（V850E/IG3）：　100 针塑封 LQFP（14 × 20）**

F1（V850E/IG3）：　161 针塑封 FBGA（10 × 10）

(1) 寄存器

(a) 端口 2 寄存器 (P2)

复位后：不确定　R/W　地址：　FFFFF404H

	7	6	5	4	3	2	1	0
P2	P27	P26	P25	P24	P23	P22	P21	P20

P2n	输出数据控制(输出模式下)
0	输出 0.
1	输出 1.

备注　n = 0 至 7

(b) 端口 2 模式寄存器 (PM2)

复位后： FFH　R/W　地址：　FFFFF424H

	7	6	5	4	3	2	1	0
PM2	PM27	PM26	PM25	PM24	PM23	PM22	PM21	PM20

PM2n	输入/输出模式控制 (端口模式下)
0	输出模式
1	输入模式

备注　n = 0 至 7

(c) 端口 2 模式控制寄存器 (PMC2)

复位后： 00H　　R/W　　地址： FFFFF444H

	7	6	5	4	3	2	1	0
PMC2	0	PMC26	PMC25	PMC24	PMC23	PMC22	PMC21	PMC20

PMC26	P26 引脚操作模式说明
0	I/O 端口
1	TOB10 输出/TOB1OFF 输入/INTP10 输入/ADTRG1 输入/INTADT1 输入

PMC25	P25 引脚操作模式说明
0	I/O 端口
1	TOB1B3 输出/TRGB1 输入

PMC24	P24 引脚操作模式说明
0	I/O 端口
1	TOB1T3 输出/EVTB1 输入

PMC23	P23 引脚操作模式说明
0	I/O 端口
1	TOB1B2 输出/TIB10 输入

PMC22	P22 引脚操作模式说明
0	I/O 端口
1	TOB1T2 输出/TIB13 输入/TOB13 输出

PMC21	P21 引脚操作模式说明
0	I/O 端口
1	TOB1B1 输出/TIB12 输入/TOB12 输出

PMC20	P20 引脚操作模式说明
0	I/O 端口
1	TOB1T1 输出/TIB11 输入/TOB11 输出

(d) 端口 **2** 功能控制寄存器 **(PFC2)**

复位后： 00H　　R/W　　地址： FFFFF464H

	7	6	5	4	3	2	1	0
PFC2	0	PFC26	PFC25	PFC24	PFC23	PFC22	PFC21	PFC20

备注　　复用功能的规范，参见 **4.3.3 (1) (f)** 端口 **2** 复用功能的设置。

(e) 端口 **2** 功能控制扩展寄存器 **(PFCE2)**

复位后： 00H　　R/W　　地址： FFFFF704H

	7	6	5	4	3	2	1	0
PFCE2	0	PFCE26	0	0	0	PFCE22	PFCE21	PFCE20

备注　　复用功能的规范，参见 **4.3.3 (1) (f)** 端口 **2** 复用功能的设置。

(f)　端口 2 复用功能的设置

PFCE26	PFC26	P26 引脚的复用功能说明
0	0	TOB10 输出
0	1	TOB1OFF 输入/INTP10 输入 (两个功能交替使用)
1	0	ADTRG1 输入/INTADT1 输入 (两个功能交替使用)
1	1	禁止设置

PFC25	P25 引脚的复用功能说明
0	TOB1B3 输出
1	TRGB1 输入

PFC24	P24 引脚的复用功能说明
0	TOB1T3 输出
1	EVTB1 输入

PFC23	P23 引脚的复用功能说明
0	TOB1B2 输出
1	TIB10 输入

PFCE22	PFC22	P22 引脚的复用功能说明
0	0	TOB1T2 输出
0	1	TIB13 输入
1	0	TOB13 输出
1	1	禁止设置

PFCE21	PFC21	P21 引脚的复用功能说明
0	0	TOB1B1 输出
0	1	TIB12 输入
1	0	TOB12 输出
1	1	禁止设置

PFCE20	PFC20	P20 引脚的复用功能说明
0	0	TOB1T1 输出
0	1	TIB11 输入
1	0	TOB11 输出
1	1	禁止设置

(g) 上拉电阻选项寄存器 2 (PU2)

复位后： 00H R/W 地址： FFFFFC44H

	7	6	5	4	3	2	1	0
PU2	PU27	PU26	PU25	PU24	PU23	PU22	PU21	PU20

PU2n	片内上拉电阻连接的控制
0	未连接
1	已连接[注]

注 当引脚是端口模式且为输入模式时或当引脚用作复用功能的输入引脚时，才可以连接片内上拉电阻。并且，片内上拉电阻可以连接到 TOB1T1 至 TOB1T3 和 TOB1B1 至 TOB1B3 引脚，这些是复用功能模式下的输出引脚，当这些引脚由于 TOB1OFF 引脚或软件处理变为高阻状态时，可以连接片内上拉电阻。当引脚处于输出模式时，片内上拉电阻不能连接。

备注 n = 0 至 7

4.3.4　端口 3

端口 3 可以按位设置为输入或输出模式。

端口 3 包括以下复用功能引脚。

<R>

表 4-9. 端口 3 的复用功能引脚

引脚名称	引脚编号				复用功能引脚名称	I/O	上拉[注 1]
	IF3	IG3					
	GC	GC	GF	F1			
P30	46	55	83	K13	RXDA1/SCL	I/O	提供
P31	47	56	84	K12	TXDA1/SDA	I/O	
P32	48	57	85	J14	SIB1/RXDA2/$\overline{\text{CS1}}$[注 2]	I/O	
P33	49	58	86	J13	SOB1/TXDA2	输出	
P34	50	59	87	J12	$\overline{\text{SCKB1}}$/INTP11/$\overline{\text{CS0}}$[注 2]	I/O	
P35	51	60	88	H14	SIB2/RXDB	输入	
P36	52	61	89	H13	SOB2/TXDB	输出	
P37	53	62	90	H12	$\overline{\text{SCKB2}}$/INTP12/ASTB[注 2]	I/O	

注　**1.** 软件上拉功能

<R>　**2.** 仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

备注　IF3：V850E/IF3

IG3：V850E/IG3

GC (V850E/IF3)：80 针塑封 LQFP (14 × 14)

GC (V850E/IG3)：100 针塑封 LQFP (密间距) (14 × 14)

GF (V850E/IG3)：100 针塑封 LQFP (14 × 20)

<R>　F1 (V850E/IG3)：161 针塑封 FBGA (10 × 10)

(1) 寄存器

(a) 端口 3 寄存器 (P3)

复位后：　不确定　　R/W　　地址：FFFFF406H

	7	6	5	4	3	2	1	0
P3	P37	P36	P35	P34	P33	P32	P31	P30

P3n	输出数据控制 (输出模式下)
0	输出 0.
1	输出 1.

备注　　**n = 0 至 7**

(b) 端口 3 模式寄存器 (PM3)

复位后：FFH　　R/W　　地址：FFFFF426H

	7	6	5	4	3	2	1	0
PM3	PM37	PM36	PM35	PM34	PM33	PM32	PM31	PM30

PM3n	输入/输出模式控制 (端口模式下)
0	输出模式
1	输入模式

备注　　**n = 0 至 7**

(c) 端口 3 模式控制寄存器 (PMC3)

复位后： 00H　　R/W　　地址： FFFFF446H

	7	6	5	4	3	2	1	0
PMC3	PMC37	PMC36	PMC35	PMC34	PMC33	PMC32	PMC31	PMC30

PMC37	P37 引脚操作模式说明
0	I/O 端口
1	$\overline{\text{SCKB2}}$ I/O/NTP12 输入/ASTB 输出[注]

PMC36	P36 引脚操作模式说明
0	I/O 端口
1	SOB2 输出/TXDB 输出

PMC35	P35 引脚操作模式说明
0	I/O 端口
1	SIB2 输入/RXDB 输入

PMC34	P34 引脚操作模式说明
0	I/O 端口
1	$\overline{\text{SCKB1}}$ I/O/INTP11输入/$\overline{\text{CS0}}$ 输出[注]

PMC33	P33 引脚操作模式说明
0	I/O 端口
1	SOB1 输出/TXDA2 输出

PMC32	P32 引脚操作模式说明
0	I/O 端口
1	SIB1 输入/RXDA2 输入/$\overline{\text{CS1}}$ 输出[注]

PMC31	P31 引脚操作模式说明
0	I/O 端口
1	TXDA1 输出/SDA I/O

PMC30	P30 引脚操作模式说明
0	I/O 端口
1	RXDA1 输入/SCL I/O

<R>　注　仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

(d) 端口 3 功能控制寄存器 (PFC3)

复位后： 00H　　R/W　　地址： FFFFF466H

	7	6	5	4	3	2	1	0
PFC3	PFC37	PFC36	PFC35	PFC34	PFC33	PFC32	PFC31	PFC30

备注　复用功能的规范，参见 **4.3.4 (1) (f) 端口 3 复用功能的设置**。

(e) 端口 3 功能控制扩展寄存器 (PFCE3)

复位后： 00H　　R/W　　地址： FFFFF706H

	7	6	5	4	3	2	1	0
PFCE3	PFCE37	PFCE36	PFCE35	PFCE34	0	PFCE32	PFCE31	PFCE30

备注　复用功能的规范，参见 **4.3.4 (1) (f) 端口 3 复用功能的设置**。

(f)　端口 3 复用功能的设置

PFCE37	PFC37	P37 引脚的复用功能说明
0	0	$\overline{\text{SCKB2}}$ 输入/输出
0	1	INTP12 输入
1	0	ASTB输出[注]
1	1	禁止设置

PFCE36	PFC36	P36 引脚的复用功能说明
0	0	SOB2 输出
0	1	TXDB 输出
1	0	禁止设置
1	1	禁止设置

PFCE35	PFC35	P35 引脚的复用功能说明
0	0	SIB2 输入
0	1	RXDB 输入
1	0	禁止设置
1	1	禁止设置

PFCE34	PFC34	P34 引脚的复用功能说明
0	0	$\overline{\text{SCKB1}}$ 输入/输出
0	1	INTP11 输入
1	0	$\overline{\text{CS0}}$ 输出[注]
1	1	禁止设置

PFC33	P33 引脚的复用功能说明
0	SOB1 输出
1	TXDA2 输出

PFCE32	PFC32	P32 引脚的复用功能说明
0	0	SIB1 输入
0	1	RXDA2 输入
1	0	$\overline{\text{CS1}}$ 输出[注]
1	1	禁止设置

<R>　注　仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

PFCE31	PFC31	P31 引脚的复用功能说明
0	0	TXDA1 输出
0	1	SDA 输入/输出
1	0	禁止设置
1	1	禁止设置

PFCE30	PFC30	P30 引脚的复用功能说明
0	0	RXDA1 输入
0	1	SCL 输入/输出
1	0	禁止设置
1	1	禁止设置

(g) 上拉电阻选项寄存器 3 (PU3)

复位后： 00H　　R/W　　地址：　FFFFFC46H

	7	6	5	4	3	2	1	0
PU3	PU37	PU36	PU35	PU34	PU33	PU32	PU31	PU30

PU3n	片内上拉电阻连接的控制
0	未连接
1	已连接[注]

注　当引脚是端口模式且为输入模式时或当引脚用作复用功能(包括从模式下的 $\overline{\text{SCKB1}}$ 和 $\overline{\text{SCKB2}}$ 引脚)的输入引脚时，才可以连接片内上拉电阻。当引脚处于输出模式时，片内上拉电阻不能连接。

备注　**n = 0 至 7**

(h) 端口 3 功能寄存器 (PF3)

复位后： 00H　　R/W　　地址：　FFFFFC66H

	7	6	5	4	3	2	1	0
PF3	0	0	0	0	0	0	PF31	PF30

PF3n	正常输出或 N沟道漏极开路输出控制 (n = 0, 1)
0	正常输出 (CMOS 输出)
1	N沟道漏极开路输出[注]

注　当使用I^2C时，设置为N沟道开漏输出。

4.3.5　端口 4

端口 4 可以按位设置为输入或输出模式。

端口 4 包括以下复用功能引脚。

<R>　**表 4-10. 端口 4 的复用功能引脚**

引脚名称	引脚编号				复用功能引脚名称	I/O	上拉[注 1]
	IF3	IG3					
	GC	GC	GF	F1			
P40	38	47	75	P12	SIB0/RXDA0	输入	提供
P41	39	48	76	N12	SOB0/TXDA0/DCK[注 2, 3]	I/O	
P42	40	49	77	M14	$\overline{\text{SCKB0}}$/INTP13/DDI[注 2, 3]	I/O	
P43	41	50	78	M13	TECR1/TIT10/TOT10/INTP14	I/O	
P44	42	51	79	L14	TENC10/EVTT1/INTP15/$\overline{\text{WAIT}}$[注 4]	输入	
P45	43	52	80	L13	TENC11/TIT11/TOT11/INTP16/$\overline{\text{WR1}}$[注 4]	I/O	
P46	44	53	81	L12	TOA40/TIA40/INTP17/$\overline{\text{WR0}}$[注 4]	I/O	
P47	45	54	82	K14	TOA41/TIA41/INTP18/$\overline{\text{RD}}$[注 4]	I/O	

注　**1.** 软件上拉功能

2. 仅限 V850E/IG3

3. P41 和 P42 引脚也可以作为片上调试引脚。通过 $\overline{\text{DRST}}$ 引脚的电平可以选择片上调试功能或端口功能(包括复用功能)，如下表所示。

端口 4 功能	
低电平输入到 $\overline{\text{DRST}}$ 引脚	高电平输入到 $\overline{\text{DRST}}$ 引脚
P41/SOB0/TXDA0	DCK
P42/$\overline{\text{SCKB0}}$/INTP13	DDI

<R>　**4.** 仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

备注　IF3：V850E/IF3

IG3：V850E/IG3

GC (V850E/IF3)：80 针塑封 LQFP (14 × 14)

GC (V850E/IG3)：100 针塑封 LQFP (密间距) (14 × 14)

GF (V850E/IG3)：100 针塑封 LQFP (14 × 20)

<R>　F1 (V850E/IG3)：161 针塑封 FBGA (10 × 10)

(1) 寄存器

(a) 端口 4 寄存器 (P4)

复位后： 不确定 R/W 地址： FFFFF408H

	7	6	5	4	3	2	1	0
P4	P47	P46	P45	P44	P43	P42	P41	P40

P4n	输出数据控制 (输出模式下)
0	输出 0.
1	输出 1.

备注 **n = 0 至 7**

(b) 端口 4 模式寄存器 (PM4)

复位后： FFH R/W 地址： FFFFF428H

	7	6	5	4	3	2	1	0
PM4	PM47	PM46	PM45	PM44	PM43	PM42	PM41	PM40

PM4n	输入/输出模式控制 (端口模式下)
0	输出模式
1	输入模式

备注 **n = 0 至 7**

(c) 端口 4 模式控制寄存器 (PMC4)

复位后： 00H　　R/W　　地址： FFFFF448H

	7	6	5	4	3	2	1	0
PMC4	PMC47	PMC46	PMC45	PMC44	PMC43	PMC42	PMC41	PMC40

PMC47	P47 引脚操作模式说明
0	I/O 端口
1	TOA41 输出/TIA41 输入/INTP18 输入/$\overline{\text{RD}}$ 输出[注 1]

PMC46	P46 引脚操作模式说明
0	I/O 端口
1	TOA40 输出/TIA40 输入/INTP17 输入/$\overline{\text{WR0}}$ 输出[注 1]

PMC45	P45 引脚操作模式说明
0	I/O 端口
1	TENC11 输入/TIT11 输入/TOT11 输出/INTP16 输入/$\overline{\text{WR1}}$ 输出[注 1]

PMC44	P44 引脚操作模式说明
0	I/O 端口
1	TENC10 输入/EVTT1 输入/INTP15 输入/$\overline{\text{WAIT}}$ 输入[注 1]

PMC43	P43 引脚操作模式说明
0	I/O 端口
1	TECR1 输入/TIT10 输入/TOT10 输出/INTP14 输入

PMC42	P42 引脚操作模式说明
0	I/O 端口
1	$\overline{\text{SCKB0}}$ 输入/输出/INTP13输入/DDI 输入[注2]

PMC41	P41 引脚操作模式说明
0	I/O 端口
1	SOB0 输出/TXDA0 输出/DCK 输入[注2]

PMC40	P40 引脚操作模式说明
0	I/O 端口
1	SIB0 输入RXDA0 输入

<R>　注　**1.** 仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

2. 仅限 V850E/IG3

(d) 端口 4 功能控制寄存器 (PFC4)

复位后： 00H　　R/W　　地址： FFFFF468H

	7	6	5	4	3	2	1	0
PFC4	PFC47	PFC46	PFC45	PFC44	PFC43	PFC42	PFC41	PFC40

备注　　复用功能的规范，参见 **4.3.5 (1) (f) 端口 4 复用功能的设置**。

(e) 端口 4 功能控制扩展寄存器 (PFCE4)

复位后： 00H　　R/W　　地址： FFFFF708H

	7	6	5	4	3	2	1	0
PFCE4	PFCE47	PFCE46	PFCE45	PFCE44	PFCE43	PFCE42	PFCE41	0

备注　　复用功能的规范，参见 **4.3.5 (1) (f) 端口 4 复用功能的设置**。

(f) 端口 4 复用功能的设置

PFCE47	PFC47	P47 引脚的复用功能说明
0	0	TOA41 输出
0	1	TIA41 输入
1	0	INTP18 输入
1	1	$\overline{\text{RD}}$输出[注]

PFCE46	PFC46	P46 引脚的复用功能说明
0	0	TOA40 输出
0	1	TIA40 输入
1	0	INTP17 输入
1	1	$\overline{\text{WR0}}$ 输出[注]

PFCE45	PFC45	P45 引脚的复用功能说明
0	0	TENC11 输入/TIT11 输入 (两个功能交替使用)
0	1	TOT11 输出
1	0	INTP16 输入
1	1	$\overline{\text{WR1}}$ 输出[注]

PFCE44	PFC44	P44 引脚的复用功能说明
0	0	TENC10 输入
0	1	EVTT1 输入
1	0	INTP15 输入
1	1	$\overline{\text{WAIT}}$输入[注]

PFCE43	PFC43	P43 引脚的复用功能说明
0	0	TECR1 输入/TIT10 输入 (两个功能交替使用)
0	1	TOT10 输出
1	0	INTP14 输入
1	1	禁止设置

PFCE42	PFC42	P42 引脚的复用功能说明
0	0	$\overline{\text{SCKB0}}$ I/O
0	1	INTP13 输入
1	0	禁止设置
1	1	禁止设置

<R> 注　仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

PFCE41	PFC41	P41 引脚的复用功能说明
0	0	SOB0 输出
0	1	TXDA0 输出
1	0	禁止设置
1	1	禁止设置

PFC40	P40 引脚的复用功能说明
0	SIB0 输入
1	RXDA0 输入

(g) 上拉电阻选项寄存器 4 (PU4)

复位后： 00H　　R/W　　地址：　FFFFFC48H

	7	6	5	4	3	2	1	0
PU4	PU47	PU46	PU45	PU44	PU43	PU42	PU41	PU40

PU4n	片内上拉电阻连接的控制
0	未连接
1	已连接[注]

注　当引脚是端口模式且为输入模式时或当引脚用作复用功能(包括从模式下的 $\overline{\text{SCKB0}}$ 引脚)的输入引脚时，才可以连接片内上拉电阻。当引脚处于输出模式时，片内上拉电阻不能连接。

备注　**n = 0 至 7**

4.3.6　端口 7

端口 7 是输入端口，它所有的引脚固定为输入模式。
各个产品的 I/O 引脚数量会有不同。

类属名	I/O 端口的数量
V850E/IF3	4 位只能输入的端口
V850E/IG3	8 位只能输入的端口

端口 7 包括以下复用功能引脚。

<R> **表 4-11. 端口 7 的复用功能引脚**

引脚名称	引脚编号				复用功能引脚名称	I/O	上拉[注1]
	IF3	IG3					
	GC	GC	GF	F1			
P70	20	25	53	L1	ANI20	输入	无
P71	19	24	52	L2	ANI21	输入	
P72	18	23	51	L3	ANI22	输入	
P73	17	22	50	K1	ANI23	输入	
P74[注2]	–	21	49	K2	ANI24[注2]	输入	
P75[注2]	–	20	48	K3	ANI25[注2]	输入	
P76[注2]	–	19	47	K4	ANI26[注2]	输入	
P77[注2]	–	18	46	J4	ANI27[注2]	输入	

注　**1.** 软件上拉功能
　　2. 仅限 V850E/IG3

备注　IF3：V850E/IF3
IG3：V850E/IG3
GC (V850E/IF3)：80 针塑封 LQFP (14 × 14)
GC (V850E/IG3)：100 针塑封 LQFP (密间距) (14 × 14)
GF (V850E/IG3)：100 针塑封 LQFP (14 × 20)
<R> F1 (V850E/IG3)：161 针塑封 FBGA (10 × 10)

(1) 寄存器

(a) 端口 7 寄存器 (P7)

复位后：　不确定　　R　　地址：　FFFFFBB0H

	7	6	5	4	3	2	1	0
P7	P77[注]	P76[注]	P75[注]	P74[注]	P73	P72	P71	P70

P7n	输入数据控制
0	输入低电平.
1	输入高电平.

注　　仅对 V850E/IG3 有效
　　使用 V850E/IF3，该寄存器的读取值不确定。

注意事项　当一起使用端口输入引脚和模拟输入引脚(ANI2n)时，用作 ANI2n 的引脚在 PMC7 寄存器中对应的位，请确保将其置位 (1)。

备注　V850E/IF3：n = 0 至 3
　　V850E/IG3：n = 0 至 7

(b) 端口 7 模式控制寄存器 (PMC7)

复位后：　00H　　R/W　　地址：　FFFFFBB8H

	7	6	5	4	3	2	1	0
PMC7	PMC77[注]	PMC76[注]	PMC75[注]	PMC74[注]	PMC73	PMC72	PMC71	PMC70

PMC7n	P7n 引脚操作模式说明
0	输入端口 (使能读取 P7n 。 当该位被读取时，输入缓冲器打开)
1	ANI2n 输入 (禁止读取 P7n 。 当该位被读取时，输入缓冲器关闭)

注　　仅对 V850E/IG3 有效
　　使用 V850E/IF3，请确保设置这些位为 0。

注意事项
1. **在 A/D 转换期间，不要使用 A/D 转换器 2 改变为端口模式。**
2. **PMC7 寄存器使能或禁止 P7 寄存器的读取。当 PMC7n 位 = 1 时，即使读取 P7 寄存器，输入缓冲也不会打开。这种情况下，P7n 位的读取值固定为低电平(V850E/IF3：n = 0 至 3, V850E/IG3：n = 0 至 7)。当读取 ANI2n 输入 (中间电平)时，这样可以防止可能流经的直通电流。**

备注　V850E/IF3：n = 0 至 3
　　V850E/IG3：n = 0 至 7

4.3.7　端口 DL

端口 DL 可以按位设置为输入或输出模式。

端口 DL 包括以下复用功能引脚。

<R>　**表 4-12. 端口 DL 的复用功能引脚**

引脚名称	引脚编号				复用功能引脚名称	I/O	上拉注 1
	IF3	IG3					
	GC	GC	GF	F1			
PDL0	65	81	9	A10	AD0注 3	I/O	提供
PDL1	64	80	8	B10	AD1注 3	I/O	
PDL2	63	79	7	A11	AD2注 3	I/O	
PDL3	62	78	6	B11	AD3注 3	I/O	
PDL4	61	77	5	C11	AD4注 3	I/O	
PDL5	60	76	4	A12	AD5注 3/FLMD1注 4	I/O	
PDL6	59	75	3	B12	AD6注 3	I/O	
PDL7	58	74	2	C14	AD7注 3	I/O	
PDL8	57	73	1	C13	AD8注 3	I/O	
PDL9	56	72	100	D14	AD9注 3	I/O	
PDL10注 2	–	71	99	D13	AD10注 3	I/O	
PDL11注 2	–	70	98	D12	AD11注 3	I/O	
PDL12注 2	–	69	97	E14	AD12注 3	I/O	
PDL13注 2	–	68	96	E13	AD13注 3	I/O	
PDL14注 2	–	67	95	E12	AD14注 3	I/O	
PDL15注 2	–	66	94	F12	AD15注 3	I/O	

注　**1.** 软件上拉功能

2. 仅限 V850E/IG3

<R>　**3.** 仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

4. 该引脚在 flash 编程模式下使用，不能通过端口控制寄存器来操作。详情参见**第二十七章　FLASH 存储器**。

备注　IF3：V850E/IF3

IG3：V850E/IG3

GC (V850E/IF3)：80 针塑封 LQFP (14 × 14)

GC (V850E/IG3)：100 针塑封 LQFP (密间距) (14 × 14)

GF (V850E/IG3)：100 针塑封 LQFP (14 × 20)

<R>　F1 (V850E/IG3)：161 针塑封 FBGA (10 × 10)

(1)　寄存器

(a)　端口 DL 寄存器 (PDL)

复位后：不确定　　R/W　　地址：　　PDL　FFFFF004H
PDLL　FFFFF004H, PDLH　FFFFF005H

	15	14	13	12	11	10	9	8
PDL (PDLH[注1])	PDL15[注2]	PDL14[注2]	PDL13[注2]	PDL12[注2]	PDL11[注2]	PDL10[注2]	PDL9	PDL8

	7	6	5	4	3	2	1	0
(PDLL)	PDL7	PDL6	PDL5	PDL4	PDL3	PDL2	PDL1	PDL0

PDLn	输出数据的控制 (输出模式)
0	输出 0
1	输出 1

注　**1.** 要按字节或按位读取/写入 PDL 寄存器的第 8 位至第 15 位，指定它们作为 PDLH 寄存器的第 0 位至第 7 位。

2. 仅对 V850E/IG3 有效
使用 V850E/IF3，该寄存器的读取值不确定。

备注　**1.** PDL 寄存器可以按照 16 位单元来读取或写入。
当 PDL 寄存器的高 8 位被用作 PDLH 寄存器，低 8 位用作 PDLL 寄存器时，这些寄存器可以按照字节或按位进行读取或写入。

2. V850E/IF3：n = 0 至 9
V850E/IG3：n = 0 至 15

(b) 端口 DL 模式寄存器 (PMDL)

复位后： FFFFH　R/W　地址：　PMDL　FFFFF024H
PMDLL　FFFFF024H, PMDLH　FFFFF025H

	15	14	13	12	11	10	9	8
PMDL (PMDLH注 1)	PMDL15注2	PMDL14注 2	PMDL13注 2	PMDL12注2	PDAL11注 2	PDAL10注2	PMDL9	PMDL8

	7	6	5	4	3	2	1	0
(PMDLL)	PMDL7	PMDL6	PMDL5	PMDL4	PMDL3	PMDL2	PMDL1	PMDL0

PMDLn	输入/输出模式控制 (端口模式下)
0	输出模式
1	输入模式

注　**1.** 要按字节或按位读取/写入 PMDL 寄存器的第 8 位至第 15 位，指定它们作为 PMDLH 寄存器的第 0 位至第 7 位。

2. 仅对 V850E/IG3 有效
使用 V850E/IF3，请确保将这些位设置为 1。

备注　**1.** PMDL 寄存器可以按照 16 位单元来读取或写入。
当 PMDL 寄存器的高 8 位被用作 PMDLH 寄存器，低 8 位用作 PMDLL 寄存器时，这些寄存器可以按照字节或按位进行读取或写入。

2. V850E/IF3：n = 0 至 9
V850E/IG3：n = 0 至 15

(c) 端口 DL 模式控制寄存器 (PMCDL)

复位后： 0000H　　R/W　　地址：　PMCDL FFFFF044H
PMCDLL FFFFF044H, PMCDLH FFFFF045H

	15	14	13	12	11	10	9	8
PMCDL (PMCDLH注 1)	PMCDL15注 2	PMCDL14注 2	PMCDL13注 2	PMCDL12注 2	PMCDL11注 2	PMCDL10注 2	PMCDL9	PMCDL8

	7	6	5	4	3	2	1	0
(PMCDLL)	PMCDL7	PMCDL6	PMCDL5	PMCDL4	PMCDL3	PMCDL2	PMCDL1	PMCDL0

PMCDLn	PMCDLn 引脚操作模式说明
0	I/O 端口
1	ADn 输入/输出

注　**1.** 要按字节或按位读取/写入 PMCDL 寄存器的第 8 位至第 15 位，指定它们作为 PMCDLH 寄存器的第 0 位至第 7 位。

2. 仅对 V850E/IG3 有效
使用 V850E/IF3，请确保将这些位设置为 0。

备注　**1.** PMCDL 寄存器可以按照 16 位单元来读取或写入。
当 PMCDL 寄存器的高 8 位被用作 PMCDLH 寄存器，低 8 位用作 PMCDLL 寄存器时，这些寄存器可以按照字节或按位进行读取或写入。

2. V850E/IF3：n = 0 至 9
V850E/IG3：n = 0 至 15

(d) 上拉电阻选项寄存器 **DL (PUDL)**

复位后： 0000H R/W 地址： PUDL FFFFFF44H
PUDLL FFFFFF44H, PUDLH FFFFFF45H

	15	14	13	12	11	10	9	8
PUDL (PUDLH注 1)	PUDL15注 2	PUDL14注 2	PUDL13注 2	PUDL12注 2	PUDL11注 2	PUDL10注 2	PUDL9	PUDL8

	7	6	5	4	3	2	1	0
(PUDLL)	PUDL7	PUDL6	PUDL5	PUDL4	PUDL3	PUDL2	PUDL1	PUDL0

PUDLn	片内上拉电阻连接的控制
0	未连接
1	已连接注 3

注 **1.** 要按字节或按位读取/写入 PUDL 寄存器的第 8 位至第 15 位，指定它们作为 PUDLH 寄存器的第 0 位至第 7 位。

2. 仅对 V850E/IG3 有效
使用 V850E/IF3，请确保将这些位设置为 0。

3. 当引脚是端口模式且为输入模式时，才可以连接片内上拉电阻。当引脚处于输出模式时，片内上拉电阻不能连接。

备注 **1.** PUDL 寄存器可以按照 16 位单元来读取或写入。
当 PUDL 寄存器的高 8 位被用作 PUDLH 寄存器，低 8 位用作 PUDLL 寄存器时，这些寄存器可以按照字节或按位进行读取或写入。

2. V850E/IF3：n = 0 至 9
V850E/IG3：n = 0 至 15

4.4　各种设置的输出数据和端口读取值

表 4-14 显示的值，用于选择各个引脚复用功能、各种设置的输出数据和端口读取值。在表 4-14 所列的设置之外，还需要各个外围功能控制寄存器的设置。

表 4-13. 各种设置的输出数据和端口读取值 (1/12)

端口名称	功能	PMCmn	PFCEmn	PFCmn	PMmn	输出数据	Pmn 读取值	备注
P00, P01, P02[注], P03[注]	输出端口	0	×	×	0	端口锁存	端口锁存	
	输入端口				1	–	引脚电平	
	TOA20, TOA21, TOA30[注], TOA31[注]	1	0	0	0	复用输出(定时器输出)	端口锁存	
					1		引脚电平	
	TIA20, TIA21, TIA30[注], TIA31[注]	1	0	1	0	–	端口锁存	复用输入(定时器输入)
					1		引脚电平	
	TOA2OFF, INTP00, INTP01, TOA3OFF[注], INTP02[注], INTP03[注]	1	1	0	0	–	端口锁存	复用输入(定时器输入，外部中断输入 (必需指定有效边沿))
					1		引脚电平	
P04[注], P06[注]	输出端口	0	×	×	0	端口锁存	端口锁存	
	输入端口				1	–	引脚电平	
	TECR0[注], TIT00[注], TENC01[注], TIT01[注]	1	0	0	0	–	端口锁存	复用输入(定时器输入)
					1		引脚电平	
	TOT00[注], TOT01[注]	1	0	1	0	复用输出(定时器输出)	端口锁存	
					1		引脚电平	
	INTP04[注], INTP06[注]	1	1	0	0	–	端口锁存	复用输入(外部中断输入 (必需指定有效边沿))
					1		引脚电平	

注　　仅限 V850E/IG3

备注　　×：无须理会

<R> **表 4-13. 各种设置的输出数据和端口读取值 (2/12)**

端口名称	功能	PMCmn	PFCEmn	PFCmn	PMmn	输出数据	Pmn 读取值	备注
P05[注 1]	输出端口	0	×	×	0	端口锁存	端口锁存	
P05[注 1]	输入端口	0	×	×	1	–	引脚电平	
P05[注 1]	TENC00[注 1]	1	0	0	0	–	端口锁存	复用输入(定时器输入)
P05[注 1]	TENC00[注 1]	1	0	0	1	–	引脚电平	复用输入(定时器输入)
P05[注 1]	EVTT0[注 1]	1	0	1	0	–	端口锁存	复用输入(定时器输入)
P05[注 1]	EVTT0[注 1]	1	0	1	1	–	引脚电平	复用输入(定时器输入)
P05[注 1]	INTP05[注 1]	1	1	0	0	–	端口锁存	复用输入(外部中断输入 (必需指定有效边沿))
P05[注 1]	INTP05[注 1]	1	1	0	1	–	引脚电平	复用输入(外部中断输入 (必需指定有效边沿))
P07[注 1]	输出端口	0	无	×	0	端口锁存	端口锁存	
P07[注 1]	输入端口	0	无	×	1	–	引脚电平	
P07[注 1]	INTP07[注 1]	1	无	0	0	–	端口锁存	复用输入(外部中断输入 (必需指定有效边沿))
P07[注 1]	INTP07[注 1]	1	无	0	1	–	引脚电平	复用输入(外部中断输入 (必需指定有效边沿))
P07[注 1]	CLKOUT[注 2]	1	无	1	0	复用输出(总线输出)	端口锁存	
P07[注 1]	CLKOUT[注 2]	1	无	1	1	复用输出(总线输出)	引脚电平	

注 **1.** 仅限 V850E/IG3

<R> **2.** 仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

备注 ×：无须理会

<R> 表 4-13. 各种设置的输出数据和端口读取值 (3/12)

端口名称	功能	PMCmn	PFCEmn	PFCmn	PMmn	输出数据	Pmn 读取值	备注
P10 至 P12	输出端口	0	×	×	0	端口锁存	端口锁存	
	输入端口				1	–	引脚电平	
	TOB0T1, TOB0B1, TOB0T2	1	0	0	0	复用输出 1 (定时器输出)	端口锁存	
					1		引脚电平	
	TIB01 至 TIB03	1	0	1	0	–	端口锁存	复用输入(定时器输入)
					1		引脚电平	
	TOB01 至 TOB03	1	1	0	0	复用输出 2 (定时器输出)	端口锁存	
					1		引脚电平	
	A0[注], A1[注], A2[注]	1	1	1	0	复用输出 3 (总线输出)	端口锁存	
					1		引脚电平	
P13 至 P15	输出端口	0	×	×	0	端口锁存	端口锁存	
	输入端口				1	–	引脚电平	
	TOB0B2, TOB0T3, TOB0B3	1	0	0	0	复用输出 1 (定时器输出)	端口锁存	
					1		引脚电平	
	TIB00, EVTB0, TRGB0	1	0	1	0	–	端口锁存	复用输入(定时器输入)
					1		引脚电平	
	A3[注], A4[注], A5[注]	1	1	0	0	复用输出 2 (总线输出)	端口锁存	
					1		引脚电平	

注 仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

备注 ×：无须理会

<R> **表 4-13. 各种设置的输出数据和端口读取值 (4/12)**

端口名称	功能	PMCmn	PFCEmn	PFCmn	PMmn	输出数据	Pmn 读取值	备注
P16	输出端口	0	×	×	0	端口锁存	端口锁存	
	输入端口				1	–	引脚电平	
	TOB0OFF/INTP08	1	0	0	0	复用输出 1 (定时器输出)	端口锁存	
					1		引脚电平	
	ADTRG0/INTADT0	1	0	1	0	–	端口锁存	复用输入(A/D 输入, 外部中断输入 (必需指定有效边沿))
					1		引脚电平	
	A6[注]	1	1	0	0	复用输出 2 (总线输出)	端口锁存	
					1		引脚电平	
P17	输出端口	0	×	×	0	端口锁存	端口锁存	
	输入端口				1	–	引脚电平	
	TOB00	1	0	0	0	复用输出 1 (定时器输出)	端口锁存	
					1		引脚电平	
	INTP09	1	0	1	0	–	端口锁存	复用输入(外部中断输入 (必需指定有效边沿))
					1		引脚电平	
	A7[注]	1	1	0	0	复用输出 2 (总线输出)	端口锁存	
					1		引脚电平	
P20 至 P22	输出端口	0	×	×	0	端口锁存	端口锁存	
	输入端口				1	–	引脚电平	
	TOB1T1, TOB1B1, TOB1T2	1	0	0	0	复用输出 1 (定时器输出)	端口锁存	
					1		引脚电平	
	TIB11 至 TIB13	1	0	1	0	–	端口锁存	复用输入(定时器输入)
					1		引脚电平	
	TOB11 至 TOB13	1	1	0	0	复用输出 2 (定时器输出)	端口锁存	
					1		引脚电平	

注 仅限 μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

备注 ×：无须理会

表 4-13. 各种设置的输出数据和端口读取值 (5/12)

端口名称	功能	PMCmn	PFCEmn	PFCmn	PMmn	输出数据	Pmn 读取值	备注
P23 至 P25	输出端口	0	无	×	0	端口锁存	端口锁存	
	输入端口				1	–	引脚电平	
	TOB1B2, TOB1T3, TOB1B3	1	无	0	0	复用输出(定时器输出)	端口锁存	
					1		引脚电平	
	TIB10, EVTB1, TRGB1	1	无	1	0	–	端口锁存	复用输入(定时器输入)
					1		引脚电平	
P26	输出端口	0	×	×	0	端口锁存	端口锁存	
	输入端口				1	–	引脚电平	
	TOB10	1	0	0	0	复用输出(定时器输出)	端口锁存	
					1		引脚电平	
	TOB1OFF/INTP10	1	0	1	0	–	端口锁存	复用输入(定时器输入, 外部中断输入 (必需指定有效边沿))
					1		引脚电平	
	ADTRG1/INTADT1	1	1	0	0	–	端口锁存	复用输入(A/D 输入, 外部中断输入 (必需指定有效边沿))
					1		引脚电平	
P27[注]	输出端口	无	无	无	0	端口锁存	端口锁存	
	输入端口				1	–	引脚电平	

注　P27 引脚也可以作为片上调试引脚(仅限 V850E/IG3)。通过 $\overline{\text{DRST}}$ 引脚的电平可以选择片上调试功能或端口功能(包括复用功能)，设置方法如下表所示。

端口 2 功能	
低电平输入到 $\overline{\text{DRST}}$ 引脚	高电平输入到 $\overline{\text{DRST}}$ 引脚
P27	DMS

备注　×：无须理会

<R> **表 4-13. 各种设置的输出数据和端口读取值 (6/12)**

端口名称	功能	PMCmn	PFCEmn	PFCmn	PMmn	输出数据	Pmn 读取值	备注
P30	输出端口	0	×	×	0	端口锁存	端口锁存	
	输入端口				1	–	引脚电平	
	RXDA1	1	0	0	0	–	端口锁存	复用输入(串行输入)
					1		引脚电平	
	SCL	1	0	1	0	复用 I/O (串行 I/O)	端口锁存	在主模式下的输出 在从模式下的输入
					1		引脚电平	
P31	输出端口	0	×	×	0	端口锁存	端口锁存	
	输入端口				1	–	引脚电平	
	TXDA1	1	0	0	0	复用输出(串行输出)	端口锁存	
					1		引脚电平	
	SDA	1	0	1	0	复用 I/O (串行 I/O)	端口锁存	在主模式下的输出 在从模式下的输入
					1		引脚电平	
P32	输出端口	0	×	×	0	端口锁存	端口锁存	
	输入端口				1	–	引脚电平	
	SIB1	1	0	0	0	–	端口锁存	复用输入(串行输入)
					1		引脚电平	
	RXDA2	1	0	1	0	–	端口锁存	复用输入(串行输入)
					1		引脚电平	
	$\overline{\text{CS1}}$[注]	1	1	0	0	复用输出(总线输出)	端口锁存	
					1		引脚电平	

注 仅限 μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

备注 ×：无须理会

<R> **表 4-13. 各种设置的输出数据和端口读取值 (7/12)**

端口名称	功能	PMCmn	PFCEmn	PFCmn	PMmn	输出数据	Pmn 读取值	备注
P33	输出端口	0	无	×	0	端口锁存	端口锁存	
	输入端口				1	–	引脚电平	
	SOB1	1	无	0	0	复用输出 1 (串行输出)	端口锁存	
					1		引脚电平	
	TXDA2	1	无	1	0	复用输出 2 (串行输出)	端口锁存	
					1		引脚电平	
P34	输出端口	0	×	×	0	端口锁存	端口锁存	
	输入端口				1	–	引脚电平	
	$\overline{\text{SCKB1}}$	1	0	0	0	复用 I/O (串行 I/O)	端口锁存	在主模式下的输出 在从模式下的输入
					1		引脚电平	
	INTP11	1	0	1	0	–	端口锁存	复用输入(外部中断输入 (必需指定有效边沿))
					1		引脚电平	
	$\overline{\text{CS0}}$[注]	1	1	0	0	复用输出(总线输出)	端口锁存	
					1		引脚电平	
P35	输出端口	0	×	×	0	端口锁存	端口锁存	
	输入端口				1	–	引脚电平	
	SIB2	1	0	0	0	–	端口锁存	复用输入(串行输入)
					1		引脚电平	
	RXDB	1	0	1	0	–	端口锁存	复用输入(串行输入)
					1		引脚电平	

注 仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

备注 ×：无须理会

<R> **表 4-13. 各种设置的输出数据和端口读取值 (8/12)**

端口名称	功能	PMCmn	PFCEmn	PFCmn	PMmn	输出数据	Pmn 读取值	备注
P36	输出端口	0	×	×	0	端口锁存	端口锁存	
	输入端口				1	–	引脚电平	
	SOB2	1	0	0	0	复用输出 1 (串行输出)	端口锁存	
					1		引脚电平	
	TXDB	1	0	1	0	复用输出 2 (串行输出)	端口锁存	
					1		引脚电平	
P37	输出端口	0	×	×	0	端口锁存	端口锁存	
	输入端口				1	–	引脚电平	
	$\overline{\text{SCKB2}}$	1	0	0	0	复用 I/O (串行 I/O)	端口锁存	在主模式下的输出 在从模式下的输入
					1		引脚电平	
	INTP12	1	0	1	0	–	端口锁存	复用输入(外部中断输入 (必需指定有效边沿))
					1		引脚电平	
	ASTB[注]	1	1	0	0	复用输出(总线输出)	端口锁存	
					1		引脚电平	
P40	输出端口	0	无	×	0	端口锁存	端口锁存	
	输入端口				1	–	引脚电平	
	SIB0	1	无	0	0	–	端口锁存	复用输入(串行输入)
					1		引脚电平	
	RXDA0	1	无	1	0	–	端口锁存	复用输入(串行输入)
					1		引脚电平	

注　仅限 μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

备注　×：无须理会

表 4-13. 各种设置的输出数据和端口读取值 (9/12)

端口名称	功能	PMCmn	PFCEmn	PFCmn	PMmn	输出数据	Pmn 读取值	备注
P41[注]	输出端口	0	×	×	0	端口锁存	端口锁存	
	输入端口				1	–	引脚电平	
	SOB0	1	0	0	0	复用输出 1 (串行输出)	端口锁存	
					1		引脚电平	
	TXDA0	1	0	1	0	复用输出 2 (串行输出)	端口锁存	
					1		引脚电平	
P42[注]	输出端口	0	×	×	0	端口锁存	端口锁存	
	输入端口				1	–	引脚电平	
	$\overline{\text{SCKB0}}$	1	0	0	0	复用 I/O (串行 I/O)	端口锁存	在主模式下的输出 在从模式下的输入
					1		引脚电平	
	INTP13	1	0	1	0	–	端口锁存	复用输入(外部中断输入 (必需指定有效边沿))
					1		引脚电平	

注 P41 和 P42 引脚也可以作为片上调试引脚 (仅限 V850E/IG3).。通过 $\overline{\text{DRST}}$ 引脚的电平可以选择片上调试功能或端口功能(包括复用功能)，设置方法如下表所示。

端口 4 功能	
低电平输入到 $\overline{\text{DRST}}$ 引脚	高电平输入到 $\overline{\text{DRST}}$ 引脚
P41/SOB0/TXDA0	DCK
P42/$\overline{\text{SCKB0}}$/INTP13	DDI

备注 ×：无须理会

<R> **表 4-13. 各种设置的输出数据和端口读取值 (10/12)**

端口名称	功能	PMCmn	PFCEmn	PFCmn	PMmn	输出数据	Pmn 读取值	备注
P43	输出端口	0	×	×	0	端口锁存	端口锁存	
	输入端口				1	–	引脚电平	
	TECR1/TIT10	1	0	0	0	–	端口锁存	复用输入(定时器输入)
					1		引脚电平	
	TOT10	1	0	1	0	复用输出(定时器输出)	端口锁存	
					1		引脚电平	
	INTP14	1	1	0	0	–	端口锁存	复用输入(外部中断输入 (必需指定有效边沿))
					1		引脚电平	
P44	输出端口	0	×	×	0	端口锁存	端口锁存	
	输入端口				1	–	引脚电平	
	TENC10	1	0	0	0	–	端口锁存	复用输入(定时器输入)
					1		引脚电平	
	EVTT1	1	0	1	0	复用输出(定时器输出)	端口锁存	
					1		引脚电平	
	INTP15	1	1	0	0	–	端口锁存	复用输入(外部中断输入 (必需指定有效边沿))
					1		引脚电平	
	$\overline{\text{WAIT}}$[注]	1	1	1	0	–	端口锁存	复用输入(总线输入)
					1		引脚电平	

注 仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

备注 ×：无须理会

<R> **表 4-13. 各种设置的输出数据和端口读取值 (11/12)**

端口名称	功能	PMCmn	PFCEmn	PFCmn	PMmn	输出数据	Pmn 读取值	备注
P45	输出端口	0	×	×	0	端口锁存	端口锁存	
	输入端口				1	–	引脚电平	
	TENC11/TIT11	1	0	0	0	–	端口锁存	复用输入(定时器输入)
					1		引脚电平	
	TOT11	1	0	1	0	复用输出 1 (定时器输出)	端口锁存	
					1		引脚电平	
	INTP16	1	1	0	0	–	端口锁存	复用输入(外部中断输入 (必需指定有效边沿))
					1		引脚电平	
	$\overline{\text{WR1}}$[注]	1	1	1	0	复用输出 2 (总线输出)	端口锁存	
					1		引脚电平	
P46, P47	输出端口	0	×	×	0	端口锁存	端口锁存	
	输入端口				1	–	引脚电平	
	TOA40，TOA41	1	0	0	0	复用输出 1 (定时器输出)	端口锁存	
					1		引脚电平	
	TIA40，TIA41	1	0	1	0	–	端口锁存	复用输入(定时器输入)
					1		引脚电平	
	INTP17，INTP18	1	1	0	0	–	端口锁存	复用输入(外部中断输入 (必需指定有效边沿))
					1		引脚电平	
	$\overline{\text{WR0}}$[注], $\overline{\text{RD}}$[注]	1	1	1	0	复用输出 2 (总线输出)	端口锁存	
					1		引脚电平	

注 仅限 μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

备注 ×：无须理会

<R> **表 4-13. 各种设置的输出数据和端口读取值 (12/12)**

端口名称	功能	PMCmn	PFCEmn	PFCmn	PMmn	输出数据	Pmn 读取值	备注
P70 至P73, P74 至P77[注 1]	输入端口	0	无	无	无	–	引脚电平	只做输入的端口
	ANI20 至ANI23, ANI24 至ANI27[注 1]	1				–	低电平	
PDL0 至PDL9, PDL10 至PDL15[注 1]	输出端口	0	无	无	0	端口锁存	端口锁存	
	输入端口				1	–	引脚电平	
	AD0 至AD15[注 2]	1	无	无	0	复用 I/O (总线 I/O)	端口锁存	
					1		引脚电平	

注 **1.** 仅限 V850E/IG3

2. 仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

3. PDL5 引脚在 flash 编程模式下使用，不能通过端口控制寄存器来操作。详情参见**第二十七章 FLASH 存储器。**

备注 ×：无须理会

4.5 使用复用功能时的端口寄存器设置

下面列出了各个端口用作复用功能时的端口寄存器设置。当端口引脚用作复用功能引脚时，请参考各个引脚的详细描述。

表 4-14. 端口引脚用作复用功能引脚 (1/8)

引脚名称	复用引脚		Pn 寄存器的 Pnx 位	PMn 寄存器的 PMnx 位	PMCn 寄存器的 PMCnx 位	PFCEn 寄存器的 PFCEnx 位	PFCn 寄存器的 PFCnx 位	其它位 (寄存器)
	名称	I/O						
P00	TOA20	输出	P00 = 无需设置	PM00 = 无需设置	PMC00 = 1	PFCE00 = 0	PFC00 = 0	
	TIA20	输入	P00 = 无需设置	PM00 = 无需设置	PMC00 = 1	PFCE00 = 0	PFC00 = 1	
	TOA2OFF	输入	P00 = 无需设置	PM00 = 无需设置	PMC00 = 1	PFCE00 = 1	PFC00 = 0	
	INTP00	输入	P00 = 无需设置	PM00 = 无需设置	PMC00 = 1	PFCE00 = 1	PFC00 = 0	INTF00 (INTF0), INTR00 (INTR0)
P01	TOA21	输出	P01 = 无需设置	PM01 = 无需设置	PMC01 = 1	PFCE01 = 0	PFC01 = 0	
	TIA21	输入	P01 = 无需设置	PM01 = 无需设置	PMC01 = 1	PFCE01 = 0	PFC01 = 1	
	INTP01	输入	P01 = 无需设置	PM01 = 无需设置	PMC01 = 1	PFCE01 = 1	PFC01 = 0	INTF01 (INTF0), INTR01 (INTR0)
P02[注]	TOA30[注]	输出	P02 = 无需设置	PM02 = 无需设置	PMC02 = 1	PFCE02 = 0	PFC02 = 0	
	TIA30[注]	输入	P02 = 无需设置	PM02 = 无需设置	PMC02 = 1	PFCE02 = 0	PFC02 = 1	
	TOA3OFF[注]	输入	P02 = 无需设置	PM02 = 无需设置	PMC02 = 1	PFCE02 = 1	PFC02 = 0	
	INTP02[注]	输入	P02 = 无需设置	PM02 = 无需设置	PMC02 = 1	PFCE02 = 1	PFC02 = 0	INTF02 (INTF0), INTR02 (INTR0)
P03[注]	TOA31[注]	输出	P03 = 无需设置	PM03 = 无需设置	PMC03 = 1	PFCE03 = 0	PFC03 = 0	
	TIA31[注]	输入	P03 = 无需设置	PM03 = 无需设置	PMC03 = 1	PFCE03 = 0	PFC03 = 1	
	INTP03[注]	输入	P03 = 无需设置	PM03 = 无需设置	PMC03 = 1	PFCE03 = 1	PFC03 = 0	INTF03 (INTF0), INTR03 (INTR0)
P04[注]	TECR0[注]	输入	P04 = 无需设置	PM04 = 无需设置	PMC04 = 1	PFCE04 = 0	PFC04 = 0	
	TIT00[注]	输入	P04 = 无需设置	PM04 = 无需设置	PMC04 = 1	PFCE04 = 0	PFC04 = 0	
	TOT00[注]	输出	P04 = 无需设置	PM04 = 无需设置	PMC04 = 1	PFCE04 = 0	PFC04 = 1	
	INTP04[注]	输入	P04 = 无需设置	PM04 = 无需设置	PMC04 = 1	PFCE04 = 1	PFC04 = 0	INTF04 (INTF0), INTR04 (INTR0)
P05[注]	TENC00[注]	输入	P05 = 无需设置	PM05 = 无需设置	PMC05 = 1	PFCE05 = 0	PFC05 = 0	
	EVTT0[注]	输入	P05 = 无需设置	PM05 = 无需设置	PMC05 = 1	PFCE05 = 0	PFC05 = 1	
	INTP05[注]	输入	P05 = 无需设置	PM05 = 无需设置	PMC05 = 1	PFCE05 = 1	PFC05 = 0	INTF05 (INTF0), INTR05 (INTR0)

注　　仅限 V850E/IG3

<R> **表 4-14. 端口引脚用作复用功能引脚 (2/8)**

引脚名称	复用引脚		Pn 寄存器的 Pnx 位	PMn 寄存器的 PMnx 位	PMCn 寄存器的 PMCnx 位	PFCEn 寄存器的 PFCEnx 位	PFCn 寄存器的 PFCnx 位	其它位 (寄存器)
	名称	I/O						
P06[注1]	TENC01[注1]	输入	P06 = 无需设置	PM06 = 无需设置	PMC06 = 1	PFCE06 = 0	PFC06 = 0	
	TIT01[注1]	输入	P06 = 无需设置	PM06 = 无需设置	PMC06 = 1	PFCE06 = 0	PFC06 = 0	
	TOT01[注1]	输出	P06 = 无需设置	PM06 = 无需设置	PMC06 = 1	PFCE06 = 0	PFC06 = 1	
	INTP06[注1]	输入	P06 = 无需设置	PM06 = 无需设置	PMC06 = 1	PFCE06 = 1	PFC06 = 0	INTF06 (INTF0), INTR06 (INTR0)
P07[注1]	INTP07[注1]	输入	P07 = 无需设置	PM07 = 无需设置	PMC07 = 1	–	PFC07 = 0	INTF07 (INTF0), INTR07 (INTR0)
	CLKOUT[注2]	输出	P07 = 无需设置	PM07 = 无需设置	PMC07 = 1	–	PFC07 = 1	
P10	TOB0T1	输出	P10 = 无需设置	PM10 = 无需设置	PMC10 = 1	PFCE10 = 0	PFC10 = 0	
	TIB01	输入	P10 = 无需设置	PM10 = 无需设置	PMC10 = 1	PFCE10 = 0	PFC10 = 1	
	TOB01	输出	P10 = 无需设置	PM10 = 无需设置	PMC10 = 1	PFCE10 = 1	PFC10 = 0	
	A0[注2]	输出	P10 = 无需设置	PM10 = 无需设置	PMC10 = 1	PFCE10 = 1	PFC10 = 1	
P11	TOB0B1	输出	P11 = 无需设置	PM11 = 无需设置	PMC11 = 1	PFCE11 = 0	PFC11 = 0	
	TIB02	输入	P11 = 无需设置	PM11 = 无需设置	PMC11 = 1	PFCE11 = 0	PFC11 = 1	
	TOB02	输出	P11 = 无需设置	PM11 = 无需设置	PMC11 = 1	PFCE11 = 1	PFC11 = 0	
	A1[注2]	输出	P11 = 无需设置	PM11 = 无需设置	PMC11 = 1	PFCE11 = 1	PFC11 = 1	
P12	TOB0T2	输出	P12 = 无需设置	PM12 = 无需设置	PMC12 = 1	PFCE12 = 0	PFC12 = 0	
	TIB03	输入	P12 = 无需设置	PM12 = 无需设置	PMC12 = 1	PFCE12 = 0	PFC12 = 1	
	TOB03	输出	P12 = 无需设置	PM12 = 无需设置	PMC12 = 1	PFCE12 = 1	PFC12 = 0	
	A2[注2]	输出	P12 = 无需设置	PM12 = 无需设置	PMC12 = 1	PFCE12 = 1	PFC12 = 1	
P13	TOB0B2	输出	P13 = 无需设置	PM13 = 无需设置	PMC13 = 1	PFCE13 = 0	PFC13 = 0	
	TIB00	输入	P13 = 无需设置	PM13 = 无需设置	PMC13 = 1	PFCE13 = 0	PFC13 = 1	
	A3[注2]	输出	P13 = 无需设置	PM13 = 无需设置	PMC13 = 1	PFCE13 = 1	PFC13 = 0	

注 **1.** 仅限 V850E/IG3

2. 仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

<R> **表 4-14. 端口引脚用作复用功能引脚 (3/8)**

引脚名称	复用引脚 名称	复用引脚 I/O	Pn 寄存器的 Pnx 位	PMn 寄存器的 PMnx 位	PMCn 寄存器的 PMCnx 位	PFCEn 寄存器的 PFCEnx 位	PFCn 寄存器的 PFCnx 位	其它位 (寄存器)
P14	TOB0T3	输出	P14 = 无需设置	PM14 = 无需设置	PMC14 = 1	PFCE14 = 0	PFC14 = 0	
	EVTB0	输入	P14 = 无需设置	PM14 = 无需设置	PMC14 = 1	PFCE14 = 0	PFC14 = 1	
	A4[注]	输出	P14 = 无需设置	PM14 = 无需设置	PMC14 = 1	PFCE14 = 1	PFC14 = 0	
P15	TOB0B3	输出	P15 = 无需设置	PM15 = 无需设置	PMC15 = 1	PFCE15 = 0	PFC15 = 0	
	TRGB0	输入	P15 = 无需设置	PM15 = 无需设置	PMC15 = 1	PFCE15 = 0	PFC15 = 1	
	A5[注]	输出	P15 = 无需设置	PM15 = 无需设置	PMC15 = 1	PFCE15 = 1	PFC15 = 0	
P16	TOB0OFF	输入	P16 = 无需设置	PM16 = 无需设置	PMC16 = 1	PFCE16 = 0	PFC16 = 0	
	INTP08	输入	P16 = 无需设置	PM16 = 无需设置	PMC16 = 1	PFCE16 = 0	PFC16 = 0	INTF08 (INTF1), INTR08 (INTR1)
	ADTRG0	输入	P16 = 无需设置	PM16 = 无需设置	PMC16 = 1	PFCE16 = 0	PFC16 = 1	
	INTADT0	输入	P16 = 无需设置	PM16 = 无需设置	PMC16 = 1	PFCE16 = 0	PFC16 = 1	ADTF0 (ADTF), ADTR0 (ADTR)
	A6[注]	输出	P16 = 无需设置	PM16 = 无需设置	PMC16 = 1	PFCE16 = 1	PFC16 = 0	
P17	TOB00	输出	P17 = 无需设置	PM17 = 无需设置	PMC17 = 1	PFCE17 = 0	PFC17 = 0	
	INTP09	输入	P17 = 无需设置	PM17 = 无需设置	PMC17 = 1	PFCE17 = 0	PFC17 = 1	INTF09 (INTF1), INTR09 (INTR1)
	A7[注]	输出	P17 = 无需设置	PM17 = 无需设置	PMC17 = 1	PFCE17 = 1	PFC17 = 0	
P20	TOB1T1	输出	P20 = 无需设置	PM20 = 无需设置	PMC20 = 1	PFCE20 = 0	PFC20 = 0	
	TIB11	输入	P20 = 无需设置	PM20 = 无需设置	PMC20 = 1	PFCE20 = 0	PFC20 = 1	
	TOB11	输出	P20 = 无需设置	PM20 = 无需设置	PMC20 = 1	PFCE20 = 1	PFC20 = 0	
P21	TOB1B1	输出	P21 = 无需设置	PM21 = 无需设置	PMC21 = 1	PFCE21 = 0	PFC21 = 0	
	TIB12	输入	P21 = 无需设置	PM21 = 无需设置	PMC21 = 1	PFCE21 = 0	PFC21 = 1	
	TOB12	输出	P21 = 无需设置	PM21 = 无需设置	PMC21 = 1	PFCE21 = 1	PFC21 = 0	
P22	TOB1T2	输出	P22 = 无需设置	PM22 = 无需设置	PMC22 = 1	PFCE22 = 0	PFC22 = 0	
	TIB13	输入	P22 = 无需设置	PM22 = 无需设置	PMC22 = 1	PFCE22 = 0	PFC22 = 1	
	TOB13	输出	P22 = 无需设置	PM22 = 无需设置	PMC22 = 1	PFCE22 = 1	PFC22 = 0	

注　仅限 μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

表 4-14. 端口引脚用作复用功能引脚 (4/8)

引脚名称	复用引脚		Pn 寄存器的 Pnx 位	PMn 寄存器的 PMnx 位	PMCn 寄存器的 PMCnx 位	PFCEn 寄存器的 PFCEnx 位	PFCn 寄存器的 PFCnx 位	其它位 (寄存器)
	名称	I/O						
P23	TOB1B2	输出	P23 = 无需设置	PM23 = 无需设置	PMC23 = 1	–	PFC23 = 0	
	TIB10	输入	P23 = 无需设置	PM23 = 无需设置	PMC23 = 1	–	PFC23 = 1	
P24	TOB1T3	输出	P24 = 无需设置	PM24 = 无需设置	PMC24 = 1	–	PFC24 = 0	
	EVTB1	输入	P24 = 无需设置	PM24 = 无需设置	PMC24 = 1	–	PFC24 = 1	
P25	TOB1B3	输出	P25 = 无需设置	PM25 = 无需设置	PMC25 = 1	–	PFC25 = 0	
	TRGB1	输入	P25 = 无需设置	PM25 = 无需设置	PMC25 = 1	–	PFC25 = 1	
P26	TOB10	输出	P26 = 无需设置	PM26 = 无需设置	PMC26 = 1	PFCE26 = 0	PFC26 = 0	
	TOB1OFF	输入	P26 = 无需设置	PM26 = 无需设置	PMC26 = 1	PFCE26 = 0	PFC26 = 1	
	INTP10	输入	P26 = 无需设置	PM26 = 无需设置	PMC26 = 1	PFCE26 = 0	PFC26 = 1	INTF10 (INTF1), INTR10 (INTR1)
	ADTRG1	输入	P26 = 无需设置	PM26 = 无需设置	PMC26 = 1	PFCE26 = 1	PFC26 = 0	
	INTADT1	输入	P26 = 无需设置	PM26 = 无需设置	PMC26 = 1	PFCE26 = 1	PFC26 = 0	ADTF1 (ADTF), ADTR1 (ADTR)
P27	DMS[注 1, 2]	输入	P27 = 无需设置	PM27 = 无需设置	–	–	–	
P30	RXDA1	输入	P30 = 无需设置	PM30 = 无需设置	PMC30 = 1	PFCE30 = 0	PFC30 = 0	
	SCL	I/O	P30 = 无需设置	PM30 = 无需设置	PMC30 = 1	PFCE30 = 0	PFC30 = 1	PF30 (PF3) = 1
P31	TXDA1	输出	P31 = 无需设置	PM31 = 无需设置	PMC31 = 1	PFCE31 = 0	PFC31 = 0	
	SDA	I/O	P31 = 无需设置	PM31 = 无需设置	PMC31 = 1	PFCE31 = 0	PFC31 = 1	PF31 (PF3) = 1

注 **1.** 仅限 V850E/IG3

2. P27 引脚也可以作为片上调试引脚。通过 $\overline{\text{DRST}}$ 引脚的电平可以选择片上调试功能或端口功能(包括复用功能)，设置方法如下表所示。

端口 2 功能	
低电平输入到 $\overline{\text{DRST}}$ 引脚	高电平输入到 $\overline{\text{DRST}}$ 引脚
P27	DMS

<R> **表 4-14. 端口引脚用作复用功能引脚 (5/8)**

引脚名称	复用引脚		Pn 寄存器的 Pnx 位	PMn 寄存器的 PMnx 位	PMCn 寄存器的 PMCnx 位	PFCEn 寄存器的 PFCEnx 位	PFCn 寄存器的 PFCnx 位	其它位 (寄存器)
	名称	I/O						
P32	SIB1	输入	P32 = 无需设置	PM32 = 无需设置	PMC32 = 1	PFCE32 = 0	PFC32 = 0	
	RXDA2	输入	P32 = 无需设置	PM32 = 无需设置	PMC32 = 1	PFCE32 = 0	PFC32 = 1	
	$\overline{\text{CS1}}$[注]	输出	P32 = 无需设置	PM32 = 无需设置	PMC32 = 1	PFCE32 = 1	PFC32 = 0	
P33	SOB1	输出	P33 = 无需设置	PM33 = 无需设置	PMC33 = 1	–	PFC33 = 0	
	TXDA2	输出	P33 = 无需设置	PM33 = 无需设置	PMC33 = 1	–	PFC33 = 1	
P34	$\overline{\text{SCKB1}}$	I/O	P34 = 无需设置	PM34 = 无需设置	PMC34 = 1	PFCE34 = 0	PFC34 = 0	
	INTP11	输入	P34 = 无需设置	PM34 = 无需设置	PMC34 = 1	PFCE34 = 0	PFC34 = 1	INTF11 (INTF1), INTR11 (INTR1)
	$\overline{\text{CS0}}$[注]	输出	P34 = 无需设置	PM34 = 无需设置	PMC34 = 1	PFCE34 = 1	PFC34 = 0	
P35	SIB2	输入	P35 = 无需设置	PM35 = 无需设置	PMC35 = 1	PFCE35 = 0	PFC35 = 0	
	RXDB	输入	P35 = 无需设置	PM35 = 无需设置	PMC35 = 1	PFCE35 = 0	PFC35 = 1	
P36	SOB2	输出	P36 = 无需设置	PM36 = 无需设置	PMC36 = 1	PFCE36 = 0	PFC36 = 0	
	TXDB	输出	P36 = 无需设置	PM36 = 无需设置	PMC36 = 1	PFCE36 = 0	PFC36 = 1	
P37	$\overline{\text{SCKB2}}$	I/O	P30 = 无需设置	PM37 = 无需设置	PMC37 = 1	PFCE37 = 0	PFC37 = 0	
	INTP12	输入	P30 = 无需设置	PM37 = 无需设置	PMC37 = 1	PFCE37 = 0	PFC37 = 1	INTF12 (INTF1), INTR12 (INTR1)
	ASTB[注]	输出	P30 = 无需设置	PM37 = 无需设置	PMC37 = 1	PFCE37 = 1	PFC37 = 0	
P40	SIB0	输入	P40 = 无需设置	PM40 = 无需设置	PMC40 = 1	–	PFC40 = 0	
	RXDA0	输入	P40 = 无需设置	PM40 = 无需设置	PMC40 = 1	–	PFC40 = 1	

注 仅限 μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

<R> 表 4-14. 端口引脚用作复用功能引脚 (6/8)

引脚名称	复用引脚		Pn 寄存器的 Pnx 位	PMn 寄存器的 PMnx 位	PMCn 寄存器的 PMCnx 位	PFCEn 寄存器的 PFCEnx 位	PFCn 寄存器的 PFCnx 位	其它位 (寄存器)
	名称	I/O						
P41	SOB0	输出	P41 = 无需设置	PM41 = 无需设置	PMC41 = 1	PFCE41 = 0	PFC41 = 0	
	TXDA0	输出	P41 = 无需设置	PM41 = 无需设置	PMC41 = 1	PFCE41 = 0	PFC41 = 1	
	DCK注 1, 2	输入	P41 = 无需设置	PM41 = 无需设置	PMC41 = 无需设置	PFCE41 = 无需设置	PFC41 = 无需设置	
P42	$\overline{\text{SCKB0}}$	I/O	P42 = 无需设置	PM42 = 无需设置	PMC42 = 1	PFCE42 = 0	PFC42 = 0	
	INTP13	输入	P42 = 无需设置	PM42 = 无需设置	PMC42 = 1	PFCE42 = 0	PFC42 = 1	INTF13 (INTF1), INTR13 (INTR1)
	DDI注 1, 2	输入	P42 = 无需设置	PM42 = 无需设置	PMC42 = 无需设置	PFCE42 = 无需设置	PFC42 = 无需设置	
P43	TECR1	输入	P43 = 无需设置	PM43 = 无需设置	PMC43 = 1	PFCE43 = 0	PFC43 = 0	
	TIT10	输入	P43 = 无需设置	PM43 = 无需设置	PMC43 = 1	PFCE43 = 0	PFC43 = 0	
	TOT10	输出	P43 = 无需设置	PM43 = 无需设置	PMC43 = 1	PFCE43 = 0	PFC43 = 1	
	INTP14	输入	P43 = 无需设置	PM43 = 无需设置	PMC43 = 1	PFCE43 = 1	PFC43 = 0	INTF14 (INTF2), INTR14 (INTR2)
P44	TENC10	输入	P44 = 无需设置	PM44 = 无需设置	PMC44 = 1	PFCE44 = 0	PFC44 = 0	
	EVTT1	输入	P44 = 无需设置	PM44 = 无需设置	PMC44 = 1	PFCE44 = 0	PFC44 = 1	
	INTP15	输入	P44 = 无需设置	PM44 = 无需设置	PMC44 = 1	PFCE44 = 1	PFC44 = 0	INTF15 (INTF2), INTR15 (INTR2)
	$\overline{\text{WAIT}}$注 3	输入	P44 = 无需设置	PM44 = 无需设置	PMC44 = 1	PFCE44 = 1	PFC44 = 1	

注 **1.** 仅限 V850E/IG3

2. P41 和 P42 引脚也可以作为片上调试引脚。通过 $\overline{\text{DRST}}$ 引脚的电平可以选择片上调试功能或端口功能(包括复用功能)，如下表所示。

端口 4 功能	
低电平输入到 $\overline{\text{DRST}}$ 引脚	高电平输入到 $\overline{\text{DRST}}$ 引脚
P41/SOB0/TXDA0	DCK
P42/$\overline{\text{SCKB0}}$/INTP13	DDI

3. 仅限 μPD70F3454GC-8EA-A, 70F3454F1-DA9-A

<R> **表 4-14. 端口引脚用作复用功能引脚 (7/8)**

引脚名称	复用引脚 名称	复用引脚 I/O	Pn 寄存器的 Pnx 位	PMn 寄存器的 PMnx 位	PMCn 寄存器的 PMCnx 位	PFCEn 寄存器的 PFCEnx 位	PFCn 寄存器的 PFCnx 位	其它位 (寄存器)
P45	TENC11	输入	P45 = 无需设置	PM45 = 无需设置	PMC45 = 1	PFCE45 = 0	PFC45 = 0	
	TIT11	输入	P45 = 无需设置	PM45 = 无需设置	PMC45 = 1	PFCE45 = 0	PFC45 = 0	
	TOT11	输出	P45 = 无需设置	PM45 = 无需设置	PMC45 = 1	PFCE45 = 0	PFC45 = 1	
	INTP16	输入	P45 = 无需设置	PM45 = 无需设置	PMC45 = 1	PFCE45 = 1	PFC45 = 0	INTF16 (INTF2), INTR16 (INTR2)
	$\overline{\text{WR1}}$[注 1]	输出	P45 = 无需设置	PM45 = 无需设置	PMC45 = 1	PFCE45 = 1	PFC45 = 1	
P46	TOA40	输出	P46 = 无需设置	PM46 = 无需设置	PMC46 = 1	PFCE46 = 0	PFC46 = 0	
	TIA40	输入	P46 = 无需设置	PM46 = 无需设置	PMC46 = 1	PFCE46 = 0	PFC46 = 1	
	INTP17	输入	P46 = 无需设置	PM46 = 无需设置	PMC46 = 1	PFCE46 = 1	PFC46 = 0	INTF17 (INTF1), INTR17 (INTR1)
	$\overline{\text{WR0}}$[注 1]	输出	P46 = 无需设置	PM46 = 无需设置	PMC46 = 1	PFCE46 = 1	PFC46 = 1	
P47	TOA41	输出	P47 = 无需设置	PM47 = 无需设置	PMC47 = 1	PFCE47 = 0	PFC47 = 0	
	TIA41	输入	P47 = 无需设置	PM47 = 无需设置	PMC47 = 1	PFCE47 = 0	PFC47 = 1	
	INTP18	输入	P47 = 无需设置	PM47 = 无需设置	PMC47 = 1	PFCE47 = 1	PFC47 = 0	INTF18 (INTF1), INTR18 (INTR1)
	$\overline{\text{RD}}$[注 1]	输出	P47 = 无需设置	PM47 = 无需设置	PMC47 = 1	PFCE47 = 1	PFC47 = 1	
P70	ANI20	输入	P70 = 无需设置	–	PMC70 = 1	–	–	
P71	ANI21	输入	P71 = 无需设置	–	PMC71 = 1	–	–	
P72	ANI22	输入	P72 = 无需设置	–	PMC72 = 1	–	–	
P73	ANI23	输入	P73 = 无需设置	–	PMC73 = 1	–	–	
P74[注 2]	ANI24[注 2]	输入	P74 = 无需设置	–	PMC74 = 1	–	–	
P75[注 2]	ANI25[注 2]	输入	P75 = 无需设置	–	PMC75 = 1	–	–	
P76[注 2]	ANI26[注 2]	输入	P76 = 无需设置	–	PMC76 = 1	–	–	
P77[注 2]	ANI27[注 2]	输入	P77 = 无需设置	–	PMC77 = 1	–	–	

注 **1.** 仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

2. 仅限 V850E/IG3

<R> **表 4-14. 端口引脚用作复用功能引脚 (8/8)**

引脚名称	复用功能		Pn 寄存器的 Pnx 位	PMn 寄存器的 PMnx 位	PMCn 寄存器的 PMCnx 位	PFCEn 寄存器的 PFCEnx 位	PFCn 寄存器的 PFCnx 位	其它位 (寄存器)
	名称	I/O						
PDL0	AD0[注1]	I/O	PDL0 = 无需设置	PMDL0 = 无需设置	PMCDL0 = 1	–	–	
PDL1	AD1[注1]	I/O	PDL1 = 无需设置	PMDL1 = 无需设置	PMCDL1 = 1	–	–	
PDL2	AD2[注1]	I/O	PDL2 = 无需设置	PMDL2 = 无需设置	PMCDL2 = 1	–	–	
PDL3	AD3[注1]	I/O	PDL3 = 无需设置	PMDL3 = 无需设置	PMCDL3 = 1	–	–	
PDL4	AD4[注1]	I/O	PDL4 = 无需设置	PMDL4 = 无需设置	PMCDL4 = 1	–	–	
PDL5	AD5[注1]	I/O	PDL5 = 无需设置	PMDL5 = 无需设置	PMCDL5 = 1	–	–	
	FLMD1[注2]	输入	PDL5 = 无需设置	PMDL5 = 无需设置	PMCDL5 = 无需设置	–	–	
PDL6	AD6[注1]	I/O	PDL6 = 无需设置	PMDL6 = 无需设置	PMCDL6 = 1	–	–	
PDL7	AD7[注1]	I/O	PDL7 = 无需设置	PMDL7 = 无需设置	PMCDL7 = 1	–	–	
PDL8	AD8[注1]	I/O	PDL8 = 无需设置	PMDL8 = 无需设置	PMCDL8 = 1	–	–	
PDL9	AD9[注1]	I/O	PDL9 = 无需设置	PMDL9 = 无需设置	PMCDL9 = 1	–	–	
PDL10[注3]	AD10[注1]	I/O	PDL10 = 无需设置	PMDL10 = 无需设置	PMCDL10 = 1	–	–	
PDL11[注3]	AD11[注1]	I/O	PDL11 = 无需设置	PMDL11 = 无需设置	PMCDL11 = 1	–	–	
PDL12[注3]	AD12[注1]	I/O	PDL12 = 无需设置	PMDL12 = 无需设置	PMCDL12 = 1	–	–	
PDL13[注3]	AD13[注1]	I/O	PDL13 = 无需设置	PMDL13 = 无需设置	PMCDL13 = 1	–	–	
PDL14[注3]	AD14[注1]	I/O	PDL14 = 无需设置	PMDL14 = 无需设置	PMCDL14 = 1	–	–	
PDL15[注3]	AD15[注1]	I/O	PDL15 = 无需设置	PMDL15 = 无需设置	PMCDL15 = 1	–	–	

注 **1.** 仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

2. PDL5 引脚也可以用作设置 flash 编程模式的引脚(FLMD1)。该引脚不需要通过端口控制寄存器操作。详情参见**第二十七章 FLASH 存储器。**

3. 仅限 V850E/IG3

4.6　噪声消除器

为下列引脚提供一个时序控制器，用于保障噪声消除时间。在噪声消除时间内发生改变的输入信号不会被内部响应。

表 **4-15.** 噪声消除器 **(1/2)**

目标引脚		滤波器类型	噪声消除宽度	采样时钟
$\overline{\text{RESET}}$		模拟滤波器	数个 10 ns	–
$\overline{\text{DRST}}$[注1]				
FLMD0				
P00/TOA20/TIA20/TOA2OFF/INTP00	TIA20	数字滤波器	2, 3 个时钟	f_{XX}, f_{XX}/4 可选
	TOA2OFF	模拟滤波器	数个 10 ns	–
	INTP00			
P01/TOA21/TIA21/INTP01	TIA21	数字滤波器	2, 3 个时钟	f_{XX}, f_{XX}/4 可选
	INTP01	模拟滤波器	数个 10 ns	–
P02[注1]/TOA30[注1]/TIA30[注1]/TOA3OFF[注1]/INTP02[注1]	TIA30[注1]	数字滤波器	2, 3 个时钟	f_{XX}, f_{XX}/4 可选
	TOA3OFF[注1]	模拟滤波器	数个 10 ns	–
	INTP02[注1]			
P03[注1]/TOA31[注1]/TIA31[注1]/INTP03[注1]	TIA31[注1]	数字滤波器	2, 3 个时钟	f_{XX}, f_{XX}/4 可选
	INTP03[注1]	模拟滤波器	数个 10 ns	–
P04[注1]/TECR0[注1]/TIT00[注1]/TOT00[注1]/INTP04[注1]	TECR0[注1]/TIT00[注1]	数字滤波器	2, 3 个时钟	f_{XX}, f_{XX}/4, f_{XX}/8, f_{XX}/16, f_{XX}/32, f_{XX}/64 可选
	INTP04[注1]	模拟滤波器	数个 10 ns	–
P05[注1]/TENC00[注1]/EVTT0[注1]/INTP05[注1]	TENC00[注1]	数字滤波器	2, 3 个时钟	f_{XX}, f_{XX}/4, f_{XX}/8, f_{XX}/16, f_{XX}/32, f_{XX}/64 可选
	EVTT0[注1]			
	INTP05[注1]	模拟滤波器	数个 10 ns	–
P06[注1]/TENC01[注1]/TIT01[注1]/TOT01[注1]/INTP06[注1]	TENC01[注1]/TIT01[注1]	数字滤波器	2, 3 个时钟	f_{XX}, f_{XX}/4, f_{XX}/8, f_{XX}/16, f_{XX}/32, f_{XX}/64 可选
	INTP06[注1]	模拟滤波器	数个 10 ns	–
P07[注1]/INTP07[注1]/CLKOUT[注2]	INTP07[注1]			
P10/TOB0T1/TIB01/TOB01/A0[注2]	TIB01	数字滤波器	2, 3 个时钟	f_{XX}/4
P11/TOB0B1/TIB02/TOB02/A1[注2]	TIB02			
P12/TOB0T2/TIB03/TOB03/A2[注2]	TIB03			
P13/TOB0B2/TIB00/A3[注2]	TIB00			
P14/TOB0T3/EVTB0/A4[注2]	EVTB0			
P15/TOB0B3/TRGB0/A5[注2]	TRGB0			

注　**1.**　仅限 V850E/IG3

<R>　**2.**　仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

<R>

<R>

注意事项 **1.** 可屏蔽中断引脚可以用来解除待机模式。详情参见第二十一章　待机功能。

2. 数字滤波器使用时钟来采样，因此当外围时钟(**f_{XX}**)停止(**STOP** 或 **IDLE** 模式)时，就不能响应输入信号。

3. 只有在复用功能模式下，噪声消除器有效。

表 4-15. 噪声消除器 (2/2)

目标引脚		滤波器类型	噪声消除宽度	采样时钟
P16/TOB0OFF/INTP08/ADTRG0/INTADT0/A6[注1]	TOB0OFF	模拟滤波器	数个 10 ns	–
	INTP08			
	ADTRG0			
	INTADT0			
P17/TOB00/INTP09/A7[注1]	INTP09			
P20/TOB1T1/TIB11/TOB11	TIB11	数字滤波器	2, 3 个时钟	f_{XX}/4
P21/TOB1B1/TIB12/TOB12	TIB12			
P22/TOB1T2/TIB13/TOB13	TIB13			
P23/TOB1B2/TIB10	TIB10			
P24/TOB1T3/EVTB1	EVTB1			
P25/TOB1B3/TRGB1	TRGB1			
P26/TOB10/TOB1OFF/INTP10/ADTRG1/INTADT1	TOB1OFF	模拟滤波器	数个 10 ns	–
	INTP10			
	ADTRG1			
	INTADT1			
P34/$\overline{\text{SCKB1}}$/INTP11/$\overline{\text{CS0}}$[注1]	INTP11			
P37/$\overline{\text{SCKB2}}$/INTP12/ASTB[注1]	INTP12			
P42/$\overline{\text{SCKB0}}$/INTP13/DDI[注2]	INTP13			
P43/TECR1/TIT10/TOT10/INTP14	TECR1/TIT10	数字滤波器	2, 3 个时钟	f_{XX}, f_{XX}/4, f_{XX}/8, f_{XX}/16, f_{XX}/32, f_{XX}/64 可选
	INTP14			f_{XX}/4, f_{XX}/16, f_{XX}/64, f_{XX}/128, f_{XX}/256, f_{XX}/512 可选
P44/TENC10/EVTT1/INTP15/$\overline{\text{WAIT}}$[注1]	TENC10			f_{XX}, f_{XX}/4, f_{XX}/8, f_{XX}/16, f_{XX}/32, f_{XX}/64 可选
	EVTT1			
	INTP15			f_{XX}/4, f_{XX}/16, f_{XX}/64, f_{XX}/128, f_{XX}/256, f_{XX}/512 可选
P45/TENC11/TIT11/TOT11/INTP16/$\overline{\text{WR1}}$[注1]	TENC11/TIT11			f_{XX}/, f_{XX}/4, f_{XX}/8, f_{XX}/16, f_{XX}/32, f_{XX}/64 可选
	INTP16			f_{XX}/4, f_{XX}/16, f_{XX}/64, f_{XX}/128, f_{XX}/256, f_{XX}/512 可选
P46/TOA40/TIA40/INTP17/$\overline{\text{WR0}}$[注1]	TIA40			f_{XX}, f_{XX}/4 可选
	INTP17	模拟滤波器	数个 10 ns	–
P47/TOA41/TIA41/INTP18/$\overline{\text{RD}}$[注1]	TIA41	数字滤波器	2, 3 个时钟	f_{XX}, f_{XX}/4 可选
	INTP18	模拟滤波器	数个 10 ns	–

<R> 注 **1.** 仅限 μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

2. 仅限 V850E/IG3

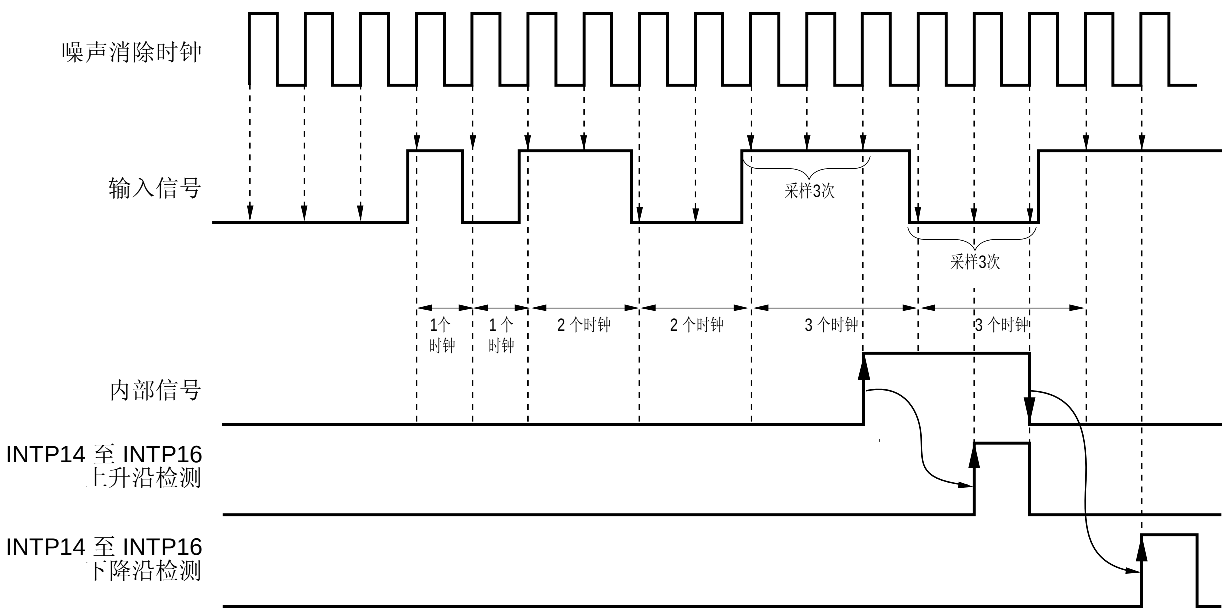
<R> 注意事项 **1.** 可屏蔽中断引脚可以用来解除待机模式。详情参见第二十一章 待机功能。

<R> **2.** 数字滤波器使用时钟来采样，因此当外围时钟(**f_{XX}**)停止(**STOP** 或 **IDLE** 模式)时，就不能响应输入信号。

3. 只有在复用功能模式下，噪声消除器有效。

下面是通过数字滤波器对 INTP14 至 INTP16、定时器 AA 输入引脚和定时器 T 输入引脚进行噪声消除的时序示例。

图 4-4. 噪声消除时序示例



注意事项　当 INTP14 至 INTP16 输入信号为高电平(或低电平)时，如果有两个或更少的噪声消除时钟，输入信号作为噪声被消除。如果采样三次或更多，检测到的边沿作为一个有效输入。

(1) 数字噪声消除 0 控制寄存器 n (INTNFCn)

INTNFCn 寄存器用于选择采样时钟，该时钟用来消除 INTPn 引脚上的数字噪声。如果使用 INTNFCn 寄存器选择的时钟，没有在该引脚上连续三次检测到相同的电平，该信号被认为是噪声而消除。
该寄存器可以按字节进行读取或写入。
系统复位后，该寄存器被设置为 00H。

注意事项 1. 如果输入信号持续了 2 个或 3 个时钟的长度，无法定义该信号检测作为有效边沿或作为噪声被消除。于是，该信号要被实际检测作为一个有效边沿，相同的信号电平输入必须持续 3 个时钟或更长时间。
2. 如果噪声的产生和采样时钟同步，通过给输入引脚加上一个滤波器来消除噪声。
3. 如果该引脚被用作正常输入端口引脚，噪声不能消除。

复位后： 00H　R/W　地址： INTNFC14 FFFFF310H, INTNFC15 FFFFF312H, INTNFC16 FFFFF314H

	7	6	5	4	3	2	1	0
INTNFCn (n = 14 至 16)	INTNFENn	0	0	0	0	INTNFCn2	INTNFCn1	INTNFCn0

<R>

INTNFENn	数字噪声消除的设置
0	注
1	数字噪声消除使能

INTNFCn2	INTNFCn1	INTNFCn0	采样时钟选择
0	0	0	$f_{XX}/4$
0	0	1	$f_{XX}/16$
0	1	0	$f_{XX}/64$
0	1	1	$f_{XX}/128$
1	0	0	$f_{XX}/256$
1	0	1	$f_{XX}/512$
其它情况			禁止设置

<R> 注　当使用 INTPn 引脚时，请确保设置 INTNFENn 位为 1 (数字噪声消除使能)。
同时，必须选择采样时钟。
当使用 INTPn 引脚的复用功能时，设置 INTNFENn 位为 0。

(2)　数字噪声消除 1 控制寄存器 n (TANFCn)

TANFCn 寄存器用于选择采样时钟，该时钟用来消除 TIAn0 或 TIAn1 引脚上的数字噪声。如果使用 TANFCn 寄存器选择的时钟，没有在该引脚上连续三次检测到相同的电平，该信号被认为是噪声而清除。
该寄存器可以按字节进行读取或写入。
系统复位后，该寄存器被设置为 00H。

注意事项 **1.** 如果输入信号持续了 **2** 个或 **3** 个时钟的长度，无法定义该信号被检测作为有效边沿或作为噪声被消除。于是，该信号要被实际检测作为一个有效边沿，相同的信号电平输入必须持续 **3** 个时钟或更长时间。
2. 如果噪声的产生和采样时钟同步，通过给输入引脚加上一个滤波器来消除噪声。
3. 如果该引脚被用作正常输入端口引脚，噪声不能消除。
4. 当 **TAAnCTL0.TAAnCE** 位被置为 **1 (**使能计数操作**)**时，噪声消除功能开始工作。

复位后：00H　　R/W　　地址：　TANFC2 FFFFFB40H, TANFC3 FFFFFB42H注,
TANFC4 FFFFFB44H

TANFCn (V850E/IF3 n = 2, 4; V850E/IG3 n = 2 至 4)

	7	6	5	4	3	2	1	0
TANFCn	TANFENn	0	0	0	0	0	0	TANFCn0

TANFENn	数字噪声消除设置
0	禁止数字噪声消除
1	使能数字噪声消除

TANFCn0	采样时钟选择
0	f_{XX}
1	$f_{XX}/4$

注　仅限 V850E/IG3

(3) 数字噪声消除 2 控制寄存器 n (TTNFCn)

TTNFCn 寄存器用于选择采样时钟，该时钟用来消除 TITn0, TITn1, EVTTn, TENCn0, TENCn1 或 TECRn 引脚上的数字噪声。如果使用 TTNFCn 寄存器选择的时钟，没有在该引脚上连续三次检测到相同的电平，该信号被认为是噪声而清除。

该寄存器可以按字节进行读取或写入。

系统复位后，该寄存器被设置为 00H。

注意事项 1. 如果输入信号持续了 **2** 个或 **3** 个时钟的长度，无法定义该信号被检测作为有效边沿或作为噪声被消除。于是，该信号要被实际检测作为一个有效边沿，相同的信号电平输入必须持续 **3** 个时钟或更长时间。

2. 如果噪声的产生和采样时钟同步，通过给输入引脚加上一个滤波器来消除噪声。

3. 如果该引脚被用作正常输入端口引脚，噪声不能消除。

4. 当 **TTnCTL0.TTnCE** 位被置为 **1 (**使能计数操作**)**时，噪声消除功能开始工作。

复位后：00H　R/W　地址：　TTNFC0 FFFFF5A0H[注], TTNFC1 FFFFF5A2H

	7	6	5	4	3	2	1	0
TTNFCn	TTNFENn	0	0	0	0	TTNFCn2	TTNFCn1	TTNFCn0

(V850E/IF3 n = 1)

(V850E/IG3 n = 0, 1)

TTNFENn	数字噪声消除设置
0	禁止数字噪声消除
1	使能数字噪声消除

TTNFCn2	TTNFCn1	TTNFCn0	采样时钟选择
0	0	0	f_{XX}
0	0	1	$f_{XX}/4$
0	1	0	$f_{XX}/8$
0	1	1	$f_{XX}/16$
1	0	0	$f_{XX}/32$
1	0	1	$f_{XX}/64$
其它情况			禁止设置

注　仅限 V850E/IG3

4.7 注意事项

4.7.1 设置端口引脚的注意事项

(1) 按照下列顺序设置端口的寄存器。

<1> 设置 PFCn 和 PFCEn 寄存器
<2> 设置 PMCn 寄存器
<3> 设置 INTFn and INTRn 寄存器

如果在设置 PFCn 和 PFCEn 寄存器之前，先设置 PMCn 寄存器，在设置 PFCn 和 PFCEn 寄存器时可能会选择预料之外的外围功能。

(2) 当引脚是端口模式且为输入模式时，或当引脚用作复用功能模式的输入引脚时，才可以连接片内上拉电阻。
并且，对于 V850E/IF3 来说，片内上拉电阻可以连接到 TOB0T1 至 TOB0T3、TOB0B1 至 TOB0B3 和 TOA21 引脚，这些是复用功能模式下的输出引脚，当这些引脚由于 TOB0OFF、TOA2OFF 或 TOB1OFF 引脚或软件处理变为高阻状态时，可以连接片内上拉电阻。
对于 V850E/IG3 来说，片内上拉电阻可以连接到 TOB0T1 至 TOB0T3, TOB0B1 至 TOB0B3, TOA21, TOB1T1 至 TOB1T3, TOB1B1 至 TOB1B3 和 TOA31 引脚，这些是复用功能模式下的输出引脚，当这些引脚由于 TOB0OFF, TOA2OFF, TOB1OFF 和 TOA3OFF 引脚或软件处理变为高阻状态时，可以连接片内上拉电阻。
按照下列顺序设置片内上拉电阻寄存器。

<1> 设置 PMCn 寄存器
<2> 设置 PMn 寄存器
<3> 设置 PU 寄存器

(3) 按照下列顺序设置 N 沟道开漏
• 用作端口模式
<1> 设置 PMCn 寄存器
<2> 设置 PFn 寄存器

• 用作I^2C复用功能模式下的输出引脚
<1> 设置 PFCn 和 PFCEn 寄存器
<2> 设置 PFn 寄存器
<3> 设置 PMCn 寄存器

4.7.2　端口 n 寄存器 （Pn）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位操作指令时，除了正常写入目标位外，还可能将该值写入其它输入端口位的输出锁存器。

所以，当一个端口由输入模式转变为输出模式时，建议重写输出锁存器。

<举例>　当 P20 为一个输出端口，同时，P21 到 P27 作为输入端口（所有引脚状态为高电平），端口锁存器的值为 00H。如果 P20 引脚的输出由于位操作指令的执行由低电平变为高电平，则端口锁存器的值为 FFH。

解释说明：向/从一个端口的 Pn 寄存器写入/读取的目标位是不同的，PMnm 位为 1 的写入的是输出锁存状态，读取目标是引脚状态

V850E/IF3 和 V850E/IG3 单片机按以下顺序执行位处理指令：

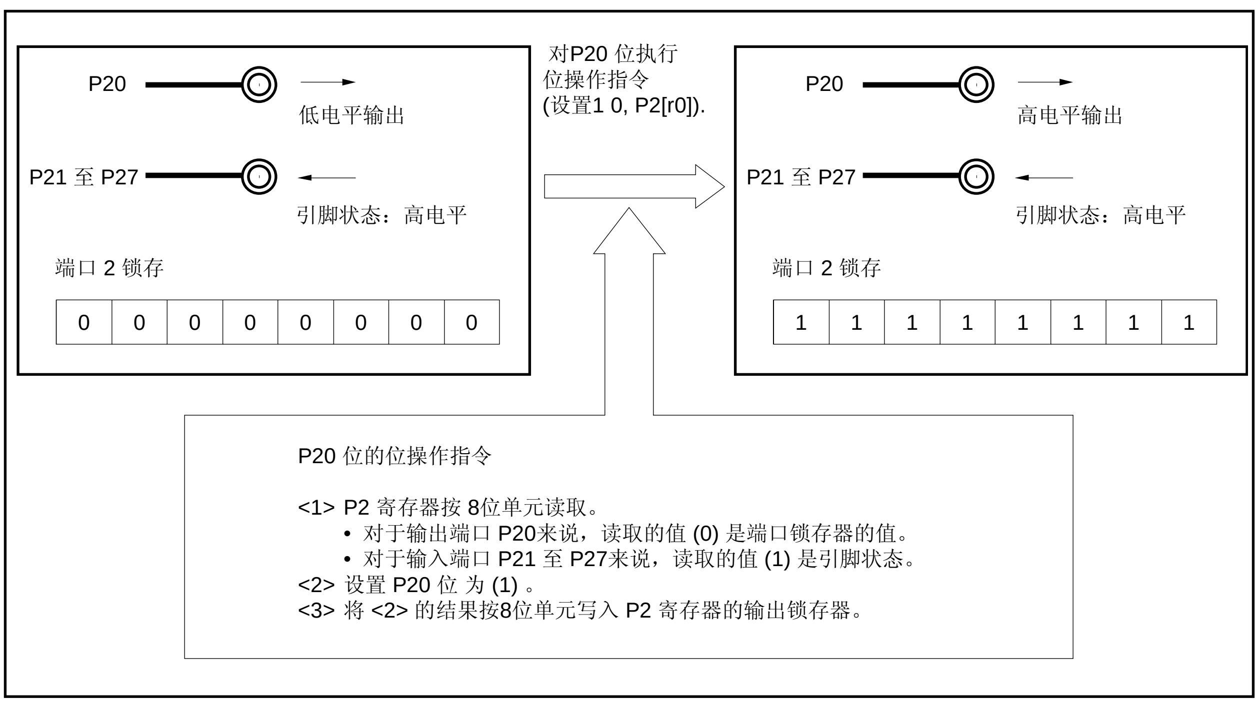
<1> 寄存器 Pn 按 8 位读取。
<2> 操作目标位。
<3> 按 8 位写入寄存器 Pn。

在步骤<1>中，作为输出端口的 P20 引脚的输出锁存器（0）的值被读取时，也读到了作为输入端口的 P21 至 P27 引脚状态。如果此时 P21 至 P27 引脚的状态为高电平，则读取的值为 FEH。

在步骤 <2>时，位操作过程使该值变为 FFH。

在步骤 <3>时，位操作将 FFH 写入输出锁存器。

图 4-5. 位处理指令 （P20 引脚）



第五章 时钟发生功能

5.1 概述

时钟发生器的特征如下：

○ 振荡器
- PLL 模式：f_X = 4 至 8 MHz（f_{XX} = 32 至 64 MHz）
- 时钟直通模式：f_X = 4 至 8 MHz （f_{XX} = 4 至 8 MHz）

○ PLL（锁相环）倍频（固定 × 8）功能
- 时钟直通模式/PLL 模式可选

○ 内部系统时钟的产生
- 4 个级别（f_{XX}，f_{XX}/2，f_{XX}/4，f_{XX}/8）

○ 外围时钟产生功能

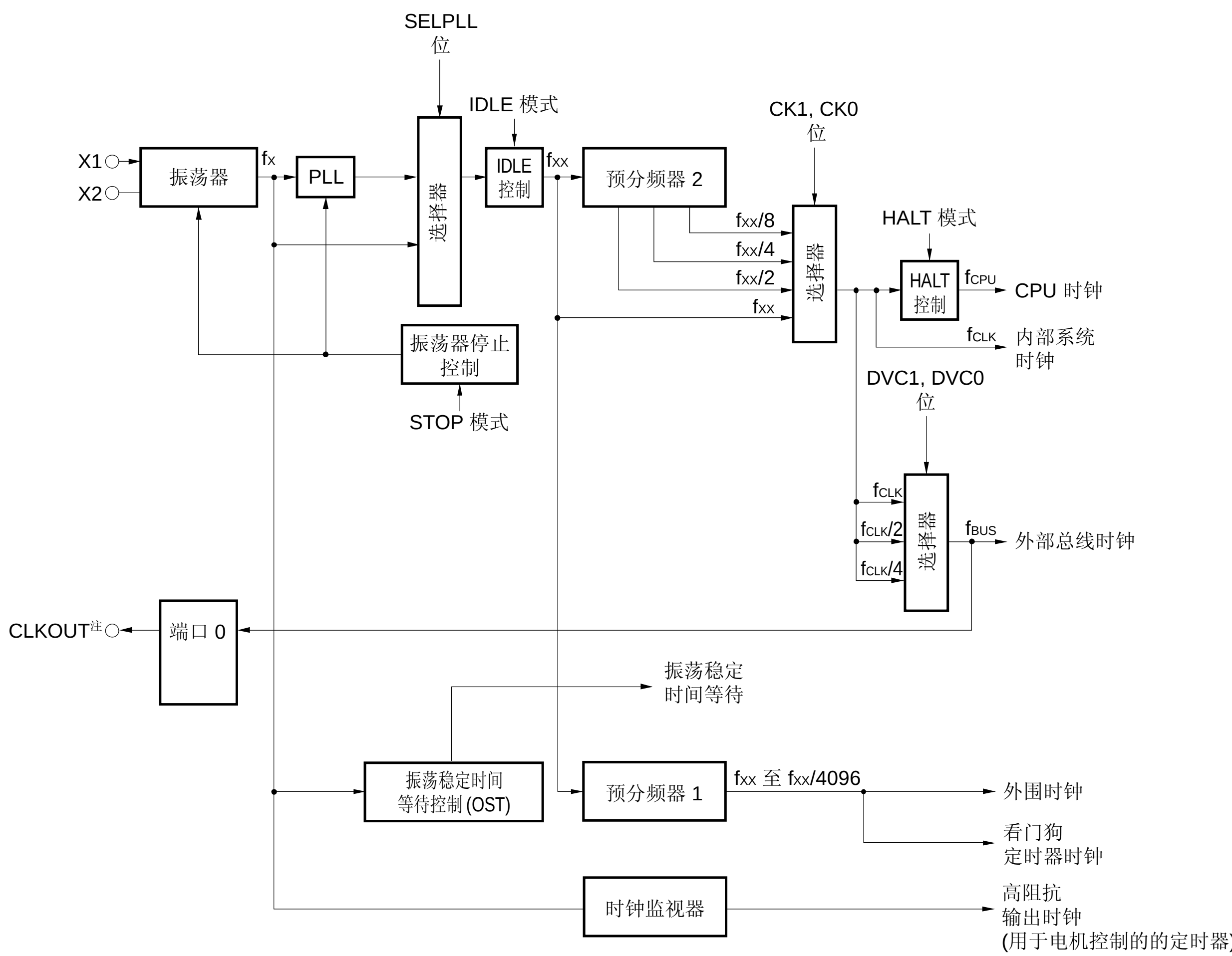
○ 振荡稳定时间选择

注意事项 振荡保证范围是 4 至 8 MHz。

备注 f_X： 振荡频率
f_{XX}： 主系统时钟频率

5.2　配置

图 **5-1.** 时钟发生器



<R>　注　仅限 μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

注意事项　**因为在刚复位后 f_{CPU} 和 f_{CLK} 不会经过 PLL，且通过预分频器 2 选择 $f_{XX}/8$。如果 f_X = 8 MHz，f_{CPU} 和 f_{CLK} 是 1 MHz。**

备注　f_X：　振荡频率
f_{XX}：　系统时钟频率
f_{CPU}：　CPU 时钟频率
f_{CLK}：　内部系统时钟频率
f_{BUS}：　外部总线时钟频率

表 **5-1.** 各个功能模块的操作时钟

功能模块		操作时钟
CPU		f_{CPU} (通过 PCC 寄存器从 f_{XX} 至 $f_{XX}/8$ 中选择)
DMA，中断控制器		f_{CLK} (通过 PCC 寄存器从 f_{XX} 至 $f_{XX}/8$ 中选择)
TAA	TAA0，TAA1	f_{XX}
	TAA2 至 TAA4	$f_{XX}/2$
TAB		f_{XX}
TMT		$f_{XX}/2$
TMM		$f_{XX}/2$
看门狗定时器		$f_{XX}/1024$
UARTA		f_{UCLK} (通过 UAnCTL1 寄存器从 $f_{XX}/2$ 至 $f_{XX}/4096$ 中选择)
UARTB		f_{XX}
CSIB		f_{CCLK} (通过 CBnCTL1 寄存器从 $f_{XX}/4$ 至 $f_{XX}/256$ 中选择)
I^2C		$f_{XX}/2$
Bus 控制功能		f_{BUS} (通过 DVC 寄存器从 $f_{CLK}/1$、$f_{CLK}/2$、$f_{CLK}/4$ 中选择)
A/D 转换器 0，1		f_{AD01} (通过 ADnOCKS 寄存器从 $f_{XX}/2$ 至 $f_{XX}/4$ 中选择)
A/D 转换器 2		$f_{AD2} = f_{XX}/2$

备注

1. f_{CPU}：CPU 时钟频率
f_{XX}:　外围时钟频率
f_{CLK}：内部系统时钟频率
f_{UCLK}: UARTA0 至 UARTA2 的基准时钟频率
f_{CCLK}: CSIB0 至 CSIB2 的基准时钟频率
f_{AD01}: A/D 转换器 0 和 1 的基准时钟频率
f_{AD2}:　A/D 转换器 2 的工作时钟频率
f_{BUS}：外部总线时钟频率

2. n = 0，1

（1）振荡器

主振荡器可以振荡产生以下频率（f_X）。

- PLL 模式(固定 × 8)： f_X = 4 至 8 MHz（f_{XX} = 32 至 64 MHz）
- 时钟直通模式： f_X = 4 至 8 MHz （f_{XX} = 4 至 8 MHz）

(2) IDLE 控制

除了振荡器、PLL、时钟监视操作、从模式下的 CSIB、低电压检测器(LVI)和上电清零电路(POC)之外，其它所有功能都停止。

(3) HALT 控制

只有 CPU 时钟 (f_{CPU})停止。

(4) PLL

该电路对振荡器(f_X)产生的时钟进行 8 倍频。

有两种操作模式：通过设置 PLL 控制寄存器 (PLLCTL)的 SELPLL 位选择操作模式，时钟直通模式下 f_X 直接输出，PLL 模式下输出倍频后的时钟。

（5）预分频器 1

该预分频器产生（f_{XX} 至 f_{XX}/4096）的时钟提供给片上外围功能。

（6）预分频器 2

该电路对系统时钟（f_{XX}）进行分频。

产生时钟（f_{XX} 至 f_{XX}/8）提供给 CPU（f_{CPU}）和内部系统（f_{CLK}）。

(7) 振荡稳定时间等待控制(OST)

该单元测量从振荡器产生的时钟开始输入直到振荡稳定的时间。也可以对 PLL 入锁时间计数。

计数时钟可以从 $2^{14}/f_X$ 至 $2^{18}/f_X$中选择。

(8) 时钟监视器

时钟监视器使用内部振荡时钟对振荡器产生的时钟(f_X)进行采样。当它检测到振荡停止时，用于电机控制的定时器的输出变为高阻抗状态(详情参见**第十章 电机控制功能**)。

5.3　控制寄存器

通过下列六个寄存器来控制时钟发生器。

- PLL 控制寄存器 (PLLCTL)
- 处理器时钟控制寄存器 (PCC)
- 省电控制寄存器(PSC)
- 省电模式寄存器 (PSMR)
- 振荡稳定时间选择寄存器 (OSTS)
- 时钟监视器模式寄存器 (CLM)

（1）　PLL 控制寄存器(PLLCTL)

PLLCTL 寄存器选择 CPU 操作时钟。
该寄存器可以按字节或按位进行读取或写入。
系统复位后，该寄存器被置为 01H。

复位后：　01H　　R/W　　地址：　FFFFF82CH

	7	6	5	4	3	2	<1>	0
PLLCTL	0	0	0	0	0	0	SELPLL	1

SELPLL	CPU 操作时钟选择
0	时钟直通模式
1	PLL 模式

注意事项　1.　请确保设置第 7 位至第 2 位为“0”，设置第 0 位为“1”。

2.　仅当 PLL 时钟频率稳定后，才可以设置 SELPLL 位为 1。如果在 PLL 时钟频率尚未稳定（未锁定期间）时重写 SELPLL 位，该位被写入 0。因此，要确保已经设置为 PLL 模式，可用下列程序作为参考。

```
_loop:   set1    1, PLLCTL
         tst1    1, PLLCTL
         bz      _loop
         (下一条指令)
```

（**2**） **处理器时钟控制寄存器 （PCC）**

处理器时钟控制寄存器 PCC 是一个专用寄存器，只有通过特定的流程才能将数据写入该寄存器（见 3.4.8 专用寄存器）。

该寄存器可以按字节或按位进行读取或写入。

系统复位后，该寄存器被置为 03H。

复位后： 03H R/W 地址： FFFFF828H

	7	6	5	4	3	2	1	0
PCC	0	0	0	0	0	0	CK1	CK0

CK1	CK0	时钟选择 (f_{CLK}/f_{CPU})
0	0	f_{XX}
0	1	$f_{XX}/2$
1	0	$f_{XX}/4$
1	1	$f_{XX}/8$

注意事项 **1. 请确保设置第 2 位至第 7 位为“0”。**

2. 在选择 PLL 模式(PLLCTL.SELPLL 位 = 1)之后，设置 PCC 寄存器为 00H。

（3）　节电控制寄存器（PSC）

PSC 寄存器是 8 位寄存器，用于控制待机功能，该寄存器的 STB 位用于指定待机模式。PSC 寄存器是专用寄存器，只能通过特定的流程才能将数据写入该寄存器（参见 **3.4.8 专用寄存器**）。

该寄存器可以按字节或按位进行读取或写入。

系统复位后，该寄存器被设置为 00H。

复位后：　00H　　R/W　　地址：　FFFFF1FEH

	7	6	5	<4>	3	2	<1>	0
PSC	0	0	0	INTM	0	0	STB	0

INTM	通过可屏蔽中断请求(INTxx注 1)控制注2 待机模式
0	使能通过 INTxx 请求解除待机模式
1	禁止通过 INTxx 请求解除待机模式

STB	设置操作模式
0	正常模式
1	待机模式

注　**1.** 详情参见**表 20-1 中断源列表**。

2. 只有在 IDLE 模式和 STOP 模式下，设置才有效。

注意事项　**1. 请确保设置第 0, 2, 3 位和第 5 位至第 7 位为“0”。**

2. 在通过 STB 位置为 1 设置待机模式之前，请确保设置 PCC 寄存器为 03H 然后设置 STB 位为 1。否则，待机模式可能无法设置或被解除。在解除待机模式之后，改变 PCC 寄存器的值为想要的值。

3. 要设置 IDLE 模式或 STOP 模式，按顺序设置 PCC 寄存器为 03H，和 PSMR.PSM0 位，然后设置 STB 位为 1。

（**4**）　节电模式寄存器（**PSMR**）

PSMR 寄存器是 8 位寄存器，用于控制软件待机模式下的操作。

该寄存器可按字节或按位进行读取或写入。

复位输入将该寄存器设置为 00H。

复位后： 00H　　R/W　　地址： FFFFF820H

	7	6	5	4	3	2	1	<0>
PSMR	0	0	0	0	0	0	0	PSM0

PSM0	指定软件待机模式下的操作
0	IDLE 模式
1	STOP 模式

注意事项　**1.** **请确保设置第 1 位至第 7 位为“0”。**

2. **仅当 PSC.STB 位为 1 时，PSM0 位有效。**

（5）　振荡稳定时间选择寄存器（OSTS）

OSTS 寄存器选择中断解除 STOP 模式之后，振荡器开始振荡直到振荡稳定的时间。

该寄存器可按字节或按位进行读取或写入。

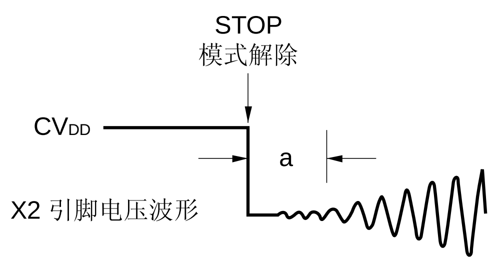
复位输入将该寄存器设置为 04H。

复位后：　04H　　R/W　　地址：　FFFFF6C0H

	7	6	5	4	3	2	1	0
OSTS	0	0	0	0	OSTS3	OSTS2	OSTS1	OSTS0

OSTS3	OSTS2	OSTS1	OSTS0	振荡稳定时间的 选择 (f_X = 8 MHz)
0	1	0	0	$2^{14}/f_X$ (2.05 ms)
0	1	0	1	$2^{15}/f_X$ (4.10 ms)
0	1	1	0	$2^{16}/f_X$ (8.19 ms)
0	1	1	1	$2^{17}/f_X$ (16.4 ms)
1	0	0	0	$2^{18}/f_X$ (32.8 ms)
其它情况				禁止设置

注意事项　**1.** 等待时间不包括解除 **STOP** 模式之后到开始振荡的时间（下图**“a”**表示的部分）。



2. 复位后 **OSTS** 寄存器的模式值是 **04H**。因此，如果使用 **8 MHz** 振荡器，振荡稳定时间大约为 **2 ms**。稳定等待时间的一半用来等待 **PLL** 入锁，因此振荡器的实际稳定时间约为 **1 ms**。当复位释放后，在复位信号的有效期间，请保证振荡能够稳定。通过中断输入而不是其它复位信号（$\overline{\textbf{RESET}}$ 引脚输入，低电压检测器**(LVI)**产生的复位信号**(LVIRES)**，上电清零电路**(POC)**产生的复位信号**(POCRES))**解除 **STOP** 模式，振荡的稳定时间由 **OSTS** 寄存器的设置值决定。因此，**OSTS** 寄存器的设置值应该是振荡器稳定所需时间的两倍 **(**因为一半时间用于 **PLL** 的稳定）。

3. 请确保设置第 **4** 位至第 **7** 位为**“0”**。

备注　　f_X：振荡频率

（6） 时钟监视器模式寄存器（CLM）

CLM 寄存器用于设置时钟监控器的操作模式。CLM 寄存器是专用寄存器，只能通过特定的流程才能将数据写入该寄存器（见 3.4.8 专用寄存器）。

该寄存器可按字节或按位进行读取或写入。

系统复位后，设置该寄存器为 00H。

复位后： 00H R/W 地址： FFFFF870H

	7	6	5	4	3	2	1	0
CLM	0	0	0	0	0	0	0	CLME

CLME	时钟监视器操作控制
0	禁止时钟监视器操作
1	使能时钟监视器操作

注意事项 **1.** 只有在复位后，才可以将 **CLME** 位清除为 **0**。

2. 当 **CLME** 位**= 1** 时，如果满足下列条件，时钟监视器功能被强制停止。

- 在 **STOP** 模式解除之后的振荡稳定期间
- 在断点期间 **(**片上调试仿真器**)**

3. 当 **CLME** 位**= 1** 时，如果检测到振荡**(**f_X**)**停止，用于电机控制的定时器的输出变为高阻抗状态。参见图 **10-4** （目标定时器输出）。

5.4　PLL 功能

5.4.1　概述

CPU 和外围模块的操作时钟可以在振荡频率 8 倍频输出和时钟直通模式之间切换。

使用 PLL 功能：　　输入时钟 (f_X) = 4 至 8 MHz，输出时钟 (f_{XX}) = 32 至 64 MHz
时钟直通模式：　　输入时钟 (f_X) = 4 至 8 MHz，输出时钟 (f_{XX}) = 4 至 8 MHz

5.4.2　PLL 模式

在 PLL 模式下，使用 PLL 将振荡频率(f_X) 8 倍频产生一个系统时钟(f_{XX})。

在 PLL 模式下，来自振荡器的时钟输入 PLL。在经过入锁时间(频率稳定时间)之后，必须有一个稳定的频率提供给内部电路。在此期间，相位被锁定在一个特定频率，并且振荡已稳定。在 V850E/IF3 和 V850E/IG3 中，自动保留复位解除之后的入锁时间。

注意事项　当使用 f_X = 8 MHz 的振荡器时，如果振荡器的振荡稳定时间必须为 3 ms (最大值)，复位输入($\overline{\text{RESET}}$ 有效)宽度必须是 2 ms (最小值)。

5.4.3　时钟直通模式

时钟直通模式下，系统时钟(f_{XX}) 和产生的振荡频率(f_X)相同。

5.5　操作

5.5.1　各个时钟的操作

下表显示各个时钟的操作状态。

表 5-2. 各个时钟的操作状态

省电模式	振荡器(fx)	PLL	内部系统时钟(f_{CLK})	外围时钟(f_{XX} 至 f_{XX}/4096)	外部总线时钟 (f_{BUS})注 1	CPU 时钟(f_{CPU})	看门狗定时器时钟注 2
正常操作	√	√	√	√	√	√	√
HALT 模式	√	√	√	√	√	×	√
IDLE 模式	√	√	×	×	×	×	×
在 STOP 模式下，和 STOP 模式解除之后的振荡稳定计数期间	×注 3	×注 3	×	×	×	×	×
在$\overline{\text{RESET}}$引脚输入注 4 期间，和之后的振荡稳定计数期间	√	×→√	√	×注 5	×注 6	√	×

<R>　注　**1.** 仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

2. 外围时钟(f_{XX}/1024) 被用作看门狗定时器时钟。

3. 在片上调试期间操作继续。

4. $\overline{\text{RESET}}$ 引脚输入，看门狗定时器产生的复位信号(WDTRES)，低电压检测器(LVI)产生的复位信号(LVIRES)，上电清零电路(POC)产生的复位信号(POCRES)。

5. 不执行从预分频器 (PRS)的输出。

6. 时钟不是从 CLKOUT 引脚输出。

备注　√：可运行
×：已停止

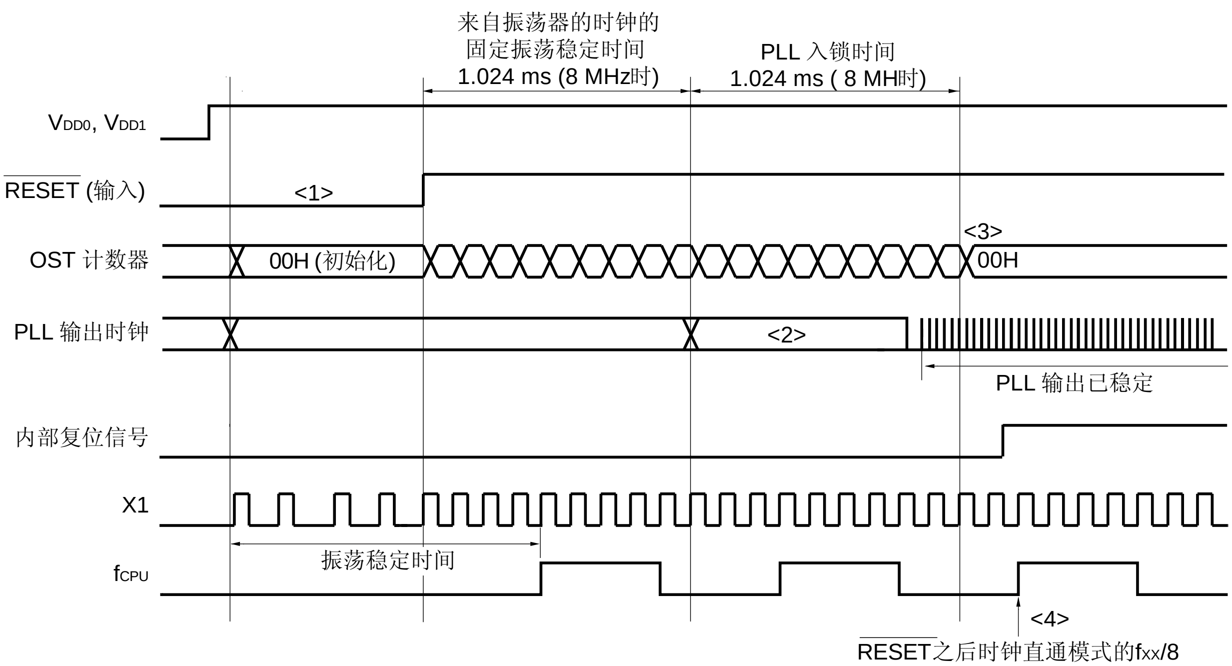
5.5.2　时钟输出功能

时钟输出功能用于从 CLKOUT 引脚输出外部总线时钟（f_{BUS}），仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A
<R>　可以支持。

当表 5-2 中的内部系统时钟 (f_{CLK}) 处于可操作状态(√)时，CLKOUT 引脚可以输出时钟。当它处于停止状态(×)时，该引脚不可以输出时钟。

5.5.3 操作时序

(1) 上电 (上电复位)



<1> 在电源上电后的 $\overline{\text{RESET}}$ 期间，振荡器被激活。
请确保 $\overline{\text{RESET}}$ 信号的低电平宽度是"谐振器的振荡稳定时间 – 固定的振荡稳定时间"或更长，要考虑振荡稳定时间。
在复位期间和固定的振荡稳定时间内，PLL 停止。

<2> 在 $\overline{\text{RESET}}$ 信号解除之后，经过固定的振荡稳定时间，PLL 停止被解除，开始计数入锁时间。

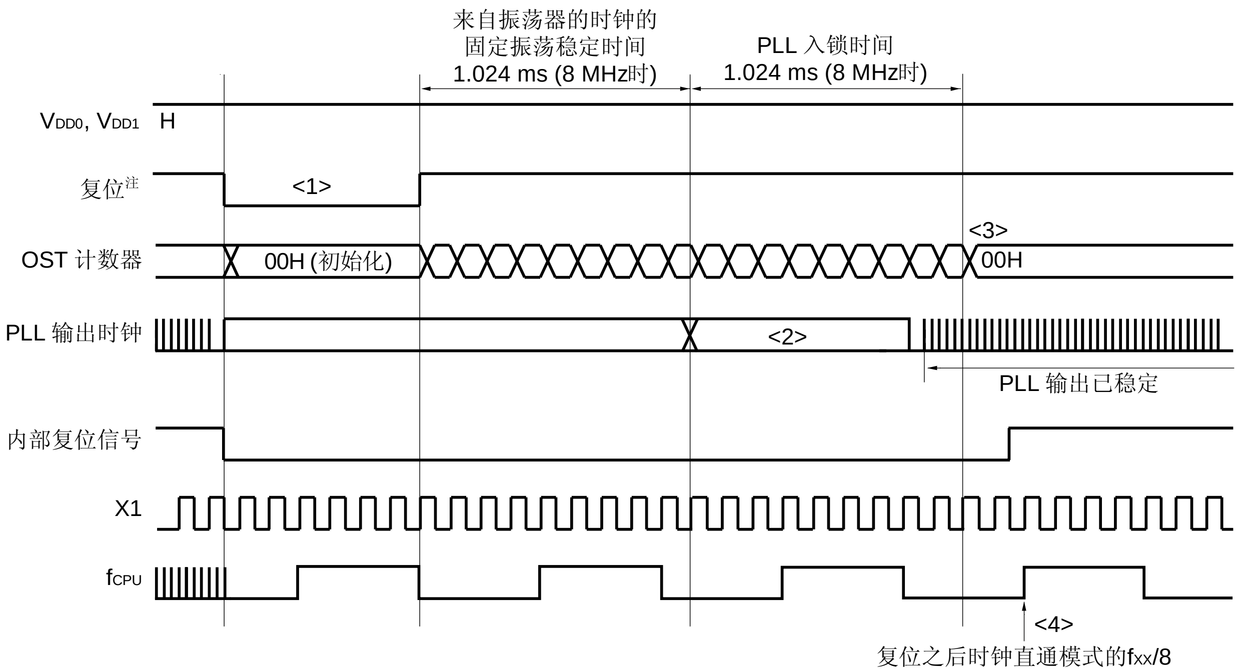
<3> 当入锁时间的计数结束时，PLL 已锁定。OST 计数器被初始化为 00H。

<4> 经过入锁时间，CPU 解除复位信号且运行于时钟直通模式(f_X)。CPU 操作时钟(f_{CPU}) 为 $f_{XX}/8$。PLL 模式可以通过软件来设置。

注意事项 **1. 在 $\overline{\text{RESET}}$ 期间，振荡器产生的时钟开始振荡。**
在 $\overline{\text{RESET}}$ 信号被解除之后，经过一个特定的等待时间(固定的振荡稳定时间)。

2. 要防止因为噪声的误操作，在设置 PLL 模式之前，不要通过 PCC 寄存器改变 CPU 操作时钟 (f_{CPU})的分频比。在改变分频比之前，请确保选择 PLL 模式。

(2) 上电时的复位



<1>　在复位期间，振荡器继续工作。
　　　在复位期间和固定的振荡稳定时间内，PLL 停止。

<2>　在复位信号解除之后，经过固定的振荡稳定时间，PLL 停止被解除，开始计数入锁时间。

<3>　当入锁时间的计数结束时，PLL 已锁定。OST 计数器被初始化为 00H。

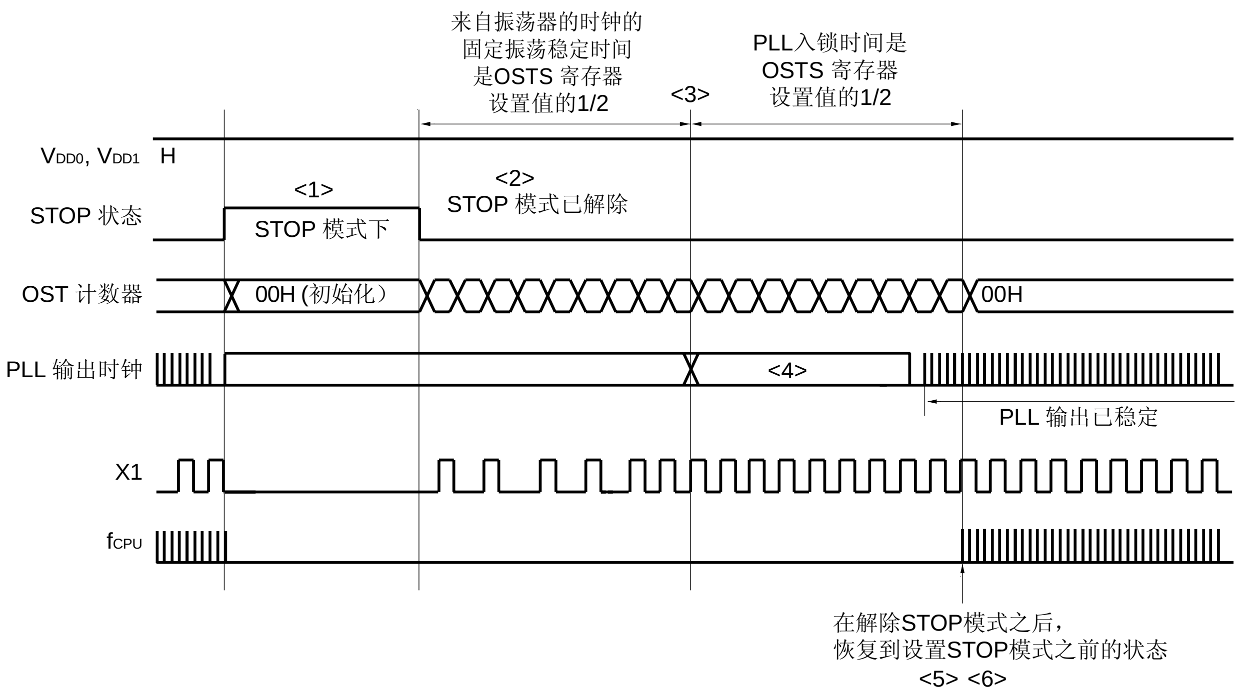
<4>　经过入锁时间，CPU 解除复位信号且运行于时钟直通模式(f_X)。CPU 操作时钟(f_{CPU}) 为 f_{XX}/8。PLL 模式可以通过软件来设置。

注　$\overline{\text{RESET}}$ 引脚输入，看门狗定时器产生的复位信号(WDTRES)，低电压检测器(LVI)产生的复位信号(LVIRES)，上电清零电路(POC)产生的复位信号(POCRES)

注意事项　1.　在复位期间，振荡器产生的时钟继续操作。
在复位信号被解除后，经过一个特定的等待时间(固定的振荡稳定时间)。

2.　要防止因为噪声的误操作，在设置 PLL 模式之前，不要通过 PCC 寄存器改变 CPU 操作时钟(f_{CPU})的分频比。在改变分频比之前，请确保选择 PLL 模式。

(3)　通过中断请求解除 STOP 模式



<1>　当设置 STOP 模式时，振荡器和 PLL 都停止。
此时，在 STOP 模式下 PLL 被停止。OST 计数器被初始化。

<2>　当 STOP 模式被解除时，振荡器被激活，且 OST 计数器开始计数振荡稳定时间。此时 PLL 保持停止。

<3>　当设置到 OSTS 的振荡稳定时间经过一半时，PLL 开始工作。在 PLL 开始工作之前，振荡器产生的时钟必须稳定。实际振荡稳定时间是“1/2 振荡稳定时间”。设置 OSTS 寄存器的值时，也要考虑这个情况。

<4>　当振荡稳定时间经过一半时，入锁等待时间启动。OST 计数器的剩余计数时间是入锁等待时间。

<5>　当 PLL 的入锁时间结束后，提供给内部电路的时钟开始。此时，设置 STOP 模式之前的状态被恢复。

<6>　当通过复位信号（$\overline{\text{RESET}}$ 引脚输入，低电压检测器(LVI)产生的复位信号(LVIRES)，上电清零电路(POC)产生的复位信号(POCRES)）解除 STOP 模式时，要执行的操作与**(1)　上电 (上电复位)** 和 **(2)　上电时的复位**中的操作相同。

5.6 时钟监视器

(1) 时钟监视器功能

时钟监视器使用内部振荡时钟对振荡器产生的时钟进行采样。当它检测到振荡停止时，用于电机控制的定时器的输出变为高阻抗状态(详情参见**第十章 电机控制功能**)。时钟监视器产生的高阻抗状态通过复位信号（$\overline{\text{RESET}}$ 引脚输入，上电清零电路(POC)产生的复位信号(POCRES)）解除，并且该引脚进入复位后的状态。

第六章 16 位定时器/事件计数器 AA（TAA）

定时器 AA（TAA）是 16 位定时器/事件计数器。
V850E/IF3 和 V850E/IG3 内有 TAA0 至 TAA4。

6.1 概述

各个通道的 TAAn 概述如下（n = 0 至 4）。

表 6-1. TAAn 概述

项目	TAA0	TAA1	TAA2	TAA3	TAA4
时钟选择	8 路	8 路	8 路	8 路	8 路
捕获触发输入引脚	无	无	2	**注 1**	2
外部事件计数输入引脚	无	无	1	**注 2**	1
外部触发输入引脚	无	无	1	**注 2**	1
定时器计数器	1	1	1	1	1
捕获/比较寄存器	2[注 3]	2[注 3]	2	2[注 4]	2
捕获/比较一致中断请求信号	2[注 3]	2[注 3]	2	2[注 4]	2
溢出中断请求信号	1	1	1	1	1
定时器输出引脚	无	无	2	**注 1**	2

注 **1.** V850E/IF3：无
V850E/IG3：2
2. V850E/IF3：无
V850E/IG3：1
3. 仅限比较功能
4. 在 V850E/IF3 内，仅限比较功能。

6.2 功能

各个通道的 TAAn 可以实现的功能各有不同，如下表所示（n = 0 至 4）。

表 6-2. TAAn 功能

功能	TAA0	TAA1	TAA2	TAA3	TAA4
间隔定时器	√	√	√	√	√
外部事件计数器	×	×	√	**注 1**	√
外部触发脉冲输出	×	×	√	**注 1**	√
单次触发脉冲输出	×	×	√	**注 1**	√
PWM 输出	×	×	√	**注 1**	√
自由运行定时器	√**注 2**	√**注 2**	√	√**注 3**	√
脉冲宽度测量	×	×	√	**注 1**	√
定时器调谐操作	√（TAB0）	√（TAB1）	×	×	×

注 **1.** V850E/IF3：×
V850E/IG3：√
2. 仅限比较功能
3. 在 V850E/IF3 内，仅限比较功能

6.3 配置

TAAn 包括下列硬件（n = 0 至 4）。

表 6-3. TAAn 的配置

项目	配置
定时器寄存器	16 位计数器 × 1
寄存器	TAAn 捕获/比较寄存器 0，1（TAAnCCR0，TAAnCCR1） TAAn 计数器读取缓冲寄存器（TAAnCNT） CCR0 和 CCR1 缓冲寄存器
定时器输入	总共 6 路（TIA20，TIA21，TIA30[注 1]，TIA31[注 1]，TIA40，TIA41 引脚）[注 2，3]
定时器输出	总共 6 路（TOA20，TOA21，TOA30[注 1]，TOA31[注 1]，TOA40，TOA41 引脚）[注 2，3]
控制寄存器	TAAn 控制寄存器 0，1（TAAnCTL0，TAAnCTL1） TAAm I/O 控制寄存器 0 至 2（TAAmIOC0 至 TAAmIOC2） TAAn 选项寄存器 0，1（TAAnOPT0，TAAnOPT1）

注 **1.** 仅限 V850E/IG3

2. TMP0 和 TMP1 不提供

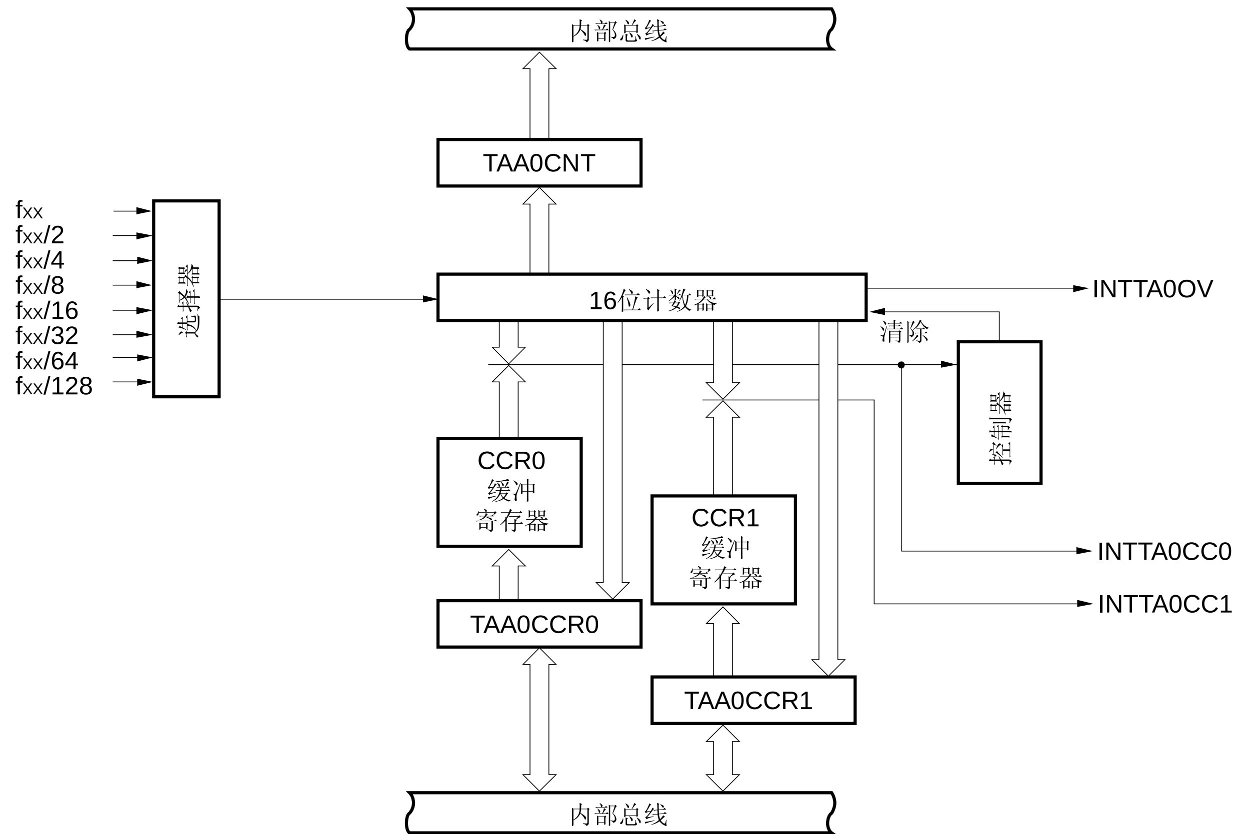
3. TIA20，TIA30 和 TIA40 引脚功能作为捕获触发输入信号，外部事件计数输入信号和外部触发输入信号，功能复用作为（复用功能）定时器输出引脚（TOA20，TOA30，TOA40）.

TIA21，TIA31 和 TIA41 引脚的功能复用作为捕获触发输入信号和定时器输出引脚（TOA21，TOA31，TOA41）.

备注 V850E/IF3：n = 0 至 4，m = 2，4

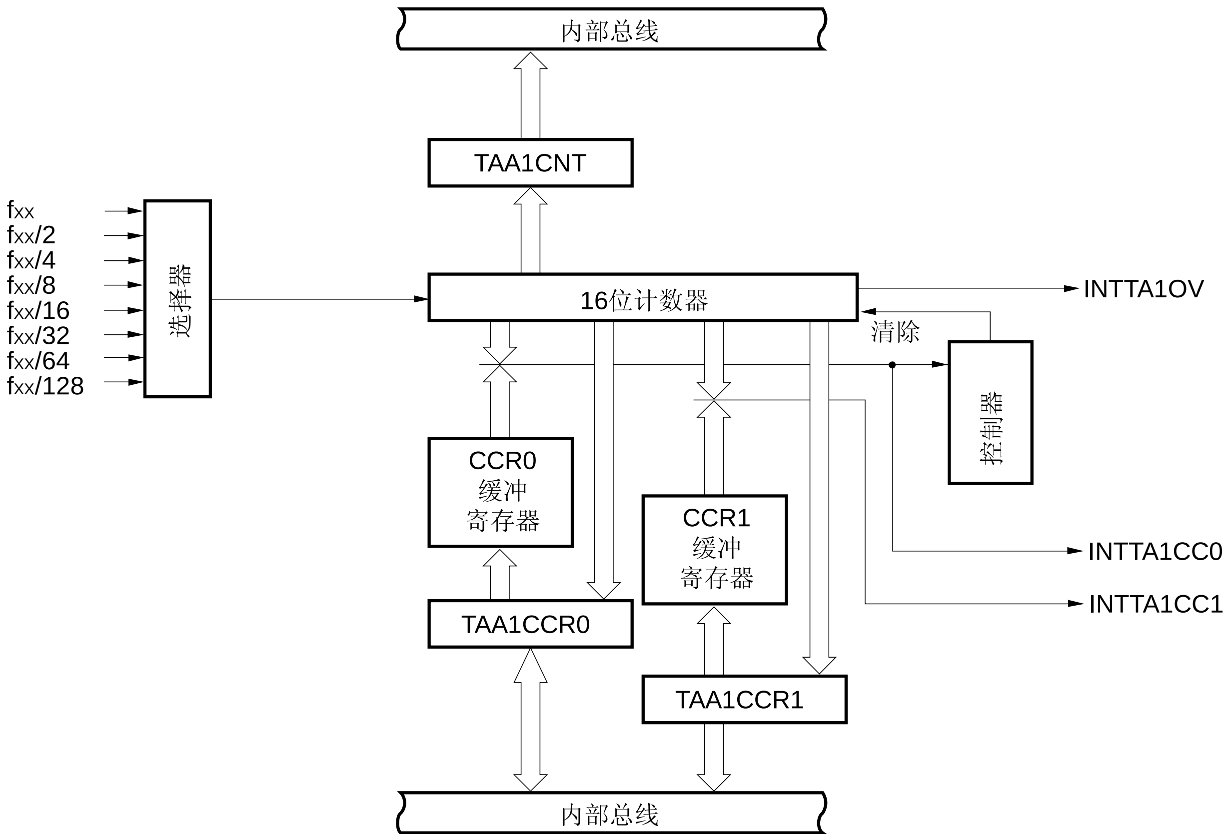
V850E/IG3：n = 0 至 4，m = 2 至 4

图 6-1. TAA0 框图



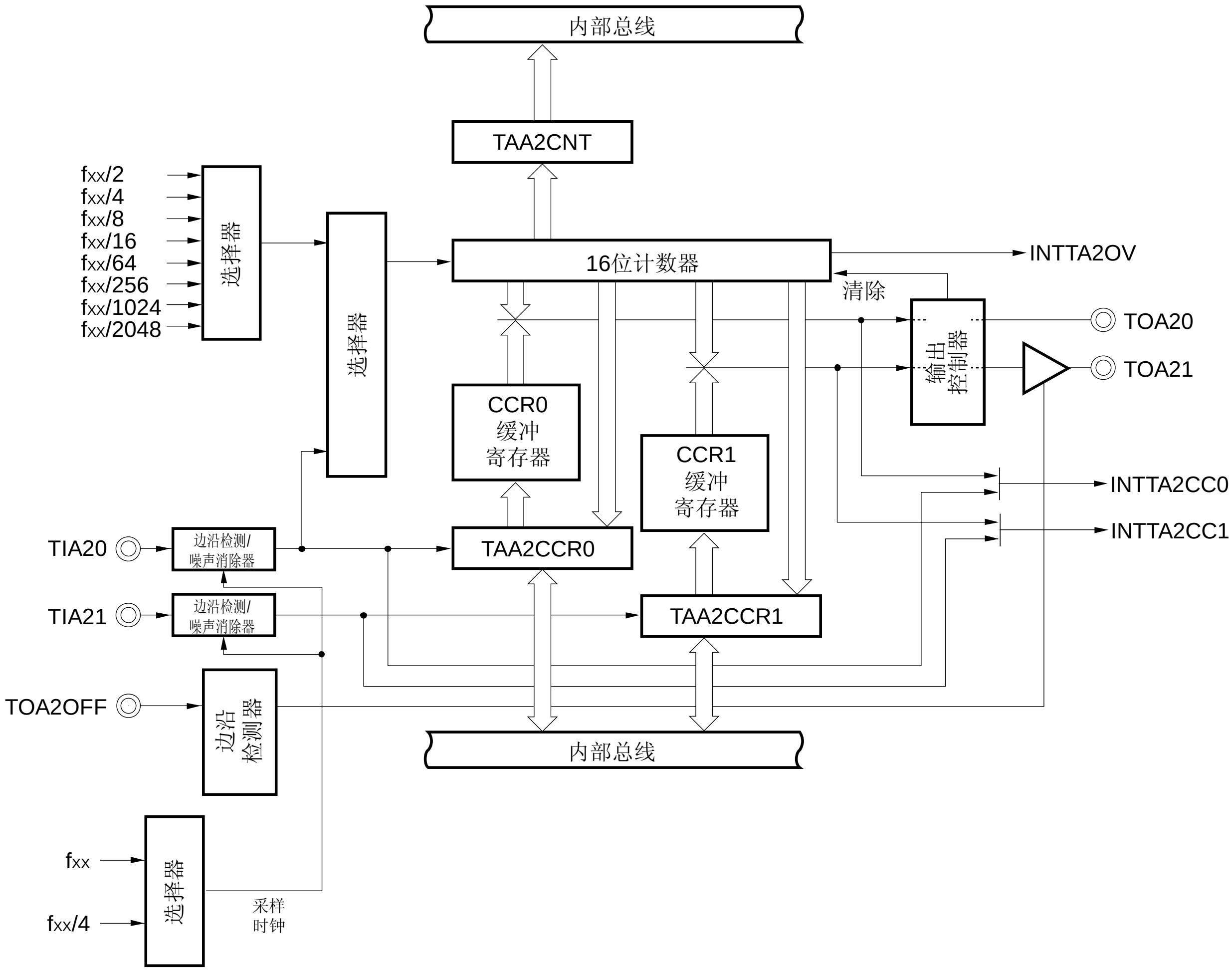
备注 fxx：外围时钟

图 6-2. TAA1 框图



备注 fxx：外围时钟

图 6-3. TAA2 框图

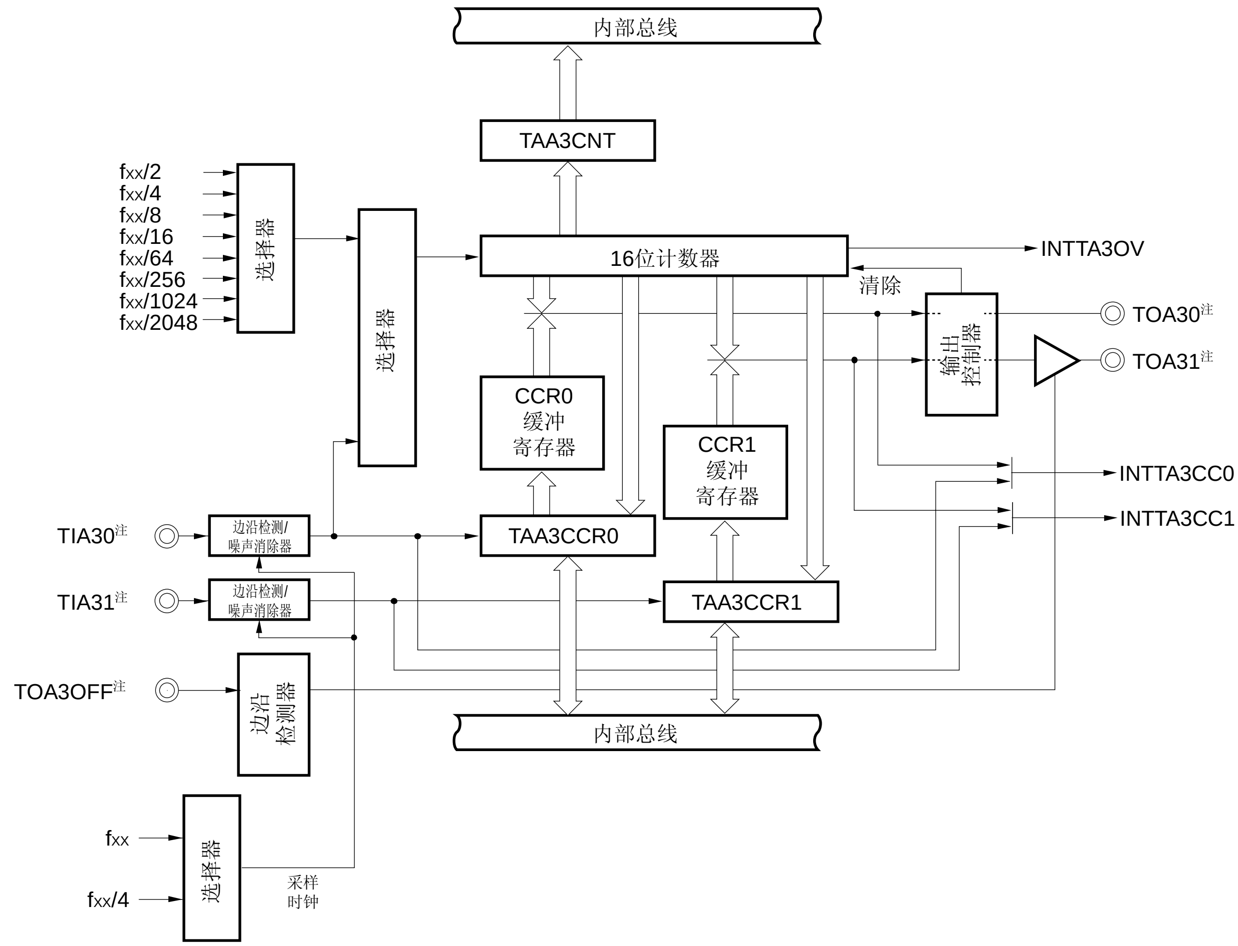


备注 1. fxx：外围时钟

2. 关于 TOA3OFF 引脚，参见 **10.3**（**6**）高阻抗输出控制寄存器 **00**，**01**，**10**，**11**，**20**，**21**，**30**，**31**（**HZAyCTL0**，**HZAyCTL1**）。

3. 关于噪声消除器，参见 **4.6** 噪声消除器。

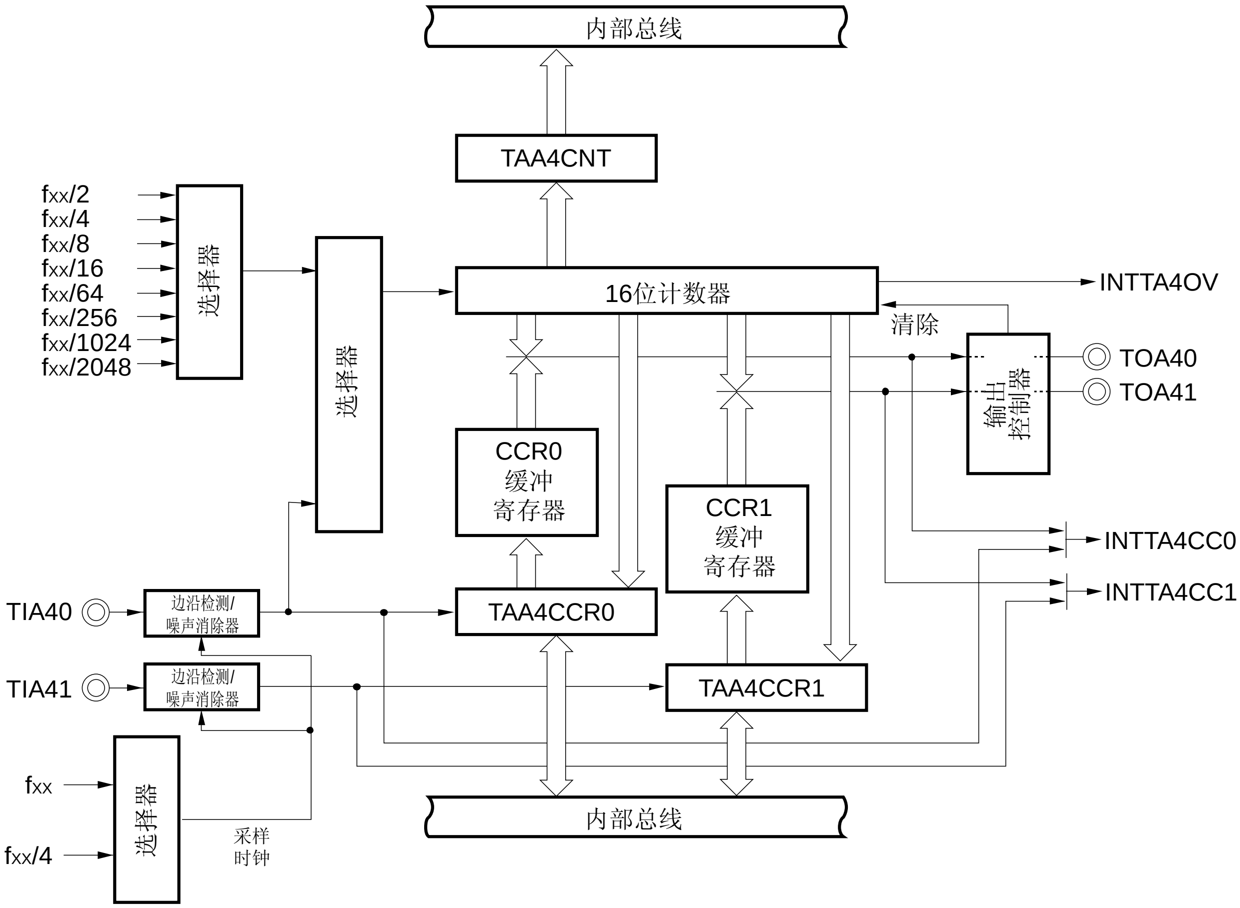
图 6-4. TAA3 框图



注　仅限 V850E/IG3

备注
1. fxx：外围时钟
2. 关于 TOA3OFF 引脚，参见 **10.3（6）高阻抗输出控制寄存器 00，01，10，11，20，21，30，31（HZAyCTL0，HZAyCTL1）**。
3. 关于噪声消除器，参见 **4.6 噪声消除器**。

图 6-5. TAA4 框图



备注 **1.** fxx：外围时钟

2. 关于噪声消除器，参见 **4.6 噪声消除器**。

（1）16 位计数器

此 16 位计数器可以计数内部时钟或外部事件。

该计数器的计数值可以通过读取 TAAnCNT 寄存器得到。

当 TAAnCTL0.TAAnCE 位= 0 时，16 位计数器的值为 FFFFH。如果此时读取 TAAnCNT 寄存器，则读回的值为 0000H 。

系统复位将 TAAnCE 位清 0。

（2）CCR0 缓冲寄存器

CCR0 缓冲寄存器是一个 16 位比较寄存器，用于比较 16 位计数器的计数值。

当 TAAnCCR0 寄存器用作比较寄存器时，写入 TAAnCCR0 寄存器的值会传输到 CCR0 缓冲寄存器。当 16 位计数器的计数值与 CCR0 缓冲寄存器的值相一致时，就会产生一个比较一致中断请求信号（INTTAnCC0）。

CCR0 缓冲寄存器不能直接读写。

复位后，TAAnCCR0 缓冲寄存器被清除为 0000H，CCR0 缓冲寄存器被清除为 0000H。

（3）CCR1 缓冲寄存器

这是一个 16 位比较寄存器，用于比较 16 位计数器的计数值。

当 TAAnCCR1 寄存器用作比较寄存器时，写入 TAAnCCR1 寄存器的值会传输到 CCR1 缓冲寄存器。当 16 位计数器的计数值和 CCR1 缓冲寄存器的值一致时，也会产生一个比较一致中断请求信号（INTTAnCC1）。

CCR1 缓冲寄存器不能直接读写。

复位后，TAAnCCR1 缓冲寄存器被清除为 0000H，CCR1 缓冲寄存器被清除为 0000H。

（4）边沿检测器

该电路检测输入到 TIA20，TIA21，TIA30（仅限 V850E/IG3），TIA31（仅限 V850E/IG3），TIA40 和 TIA41 引脚信号的有效边沿。通过设置 TAAmIOC1 和 TAAmIOC2 寄存器，可以选择无边沿、上升沿、下降沿、上升/下降双边沿作为有效沿。

（5）输出控制器

该电路控制 TOA20，TOA21，TOA30（仅限 V850E/IG3），TOA31（仅限 V850E/IG3），TOA40 和 TOA41 引脚的输出。TOAAm0 和 TOAAm1 引脚的输出由 TAAmIOC0 寄存器控制。

（6）选择器

此选择器选择 16 位计数器的计数时钟。可选择 8 种内部时钟或一个外部事件作为计数时钟。

6.4 寄存器

（1）TAAn 控制寄存器 0（TAAnCTL0）

TAAnCTL0 寄存器是 8 位寄存器，用于控制定时器 TAAn 的操作。

该寄存器可以按字节或按位进行读取和写入。

复位输入将该寄存器设为 00H。

随时可以用软件将相同的值写入 TAAnCTL0 寄存器。

复位后： 00H　R/W　地址：　TAA0CTL0 FFFFF660H, TAA1CTL0 FFFFF680H,
TAA2CTL0 FFFFF6A0H, TAA3CTL0 FFFFFB00H,
TAA4CTL0 FFFFFB20H

	<7>	6	5	4	3	2	1	0
TAAnCTL0 (n = 0 至 4)	TAAnCE	0	0	0	0	TAAnCKS2	TAAnCKS1	TAAnCKS0

TAAnCE	TAAn 操作控制
0	TAAn 操作禁止 (TAAn 异步复位[注])
1	TAAn 操作使能， TAAn 操作开始

TAAnCKS2	TAAnCKS1	TAAnCKS0	内部计数时钟选择	
			TAA0, TAA1	TAA2 至 TAA4
0	0	0	f_{XX}	$f_{XX}/2$
0	0	1	$f_{XX}/2$	$f_{XX}/4$
0	1	0	$f_{XX}/4$	$f_{XX}/8$
0	1	1	$f_{XX}/8$	$f_{XX}/16$
1	0	0	$f_{XX}/16$	$f_{XX}/64$
1	0	1	$f_{XX}/32$	$f_{XX}/256$
1	1	0	$f_{XX}/64$	$f_{XX}/1024$
1	1	1	$f_{XX}/128$	$f_{XX}/2048$

注　TAAnOPT0.TAAnOVF 位和 16 位计数器同时复位。而且，当 16 位计数器复位的同时，定时器输出（TOA20，TOA21，TOA30（仅限 V850E/IG3），TOA31（仅限 V850E/IG3），TOA40，TOA41 引脚）被复位为 TAAmIOC0 寄存器的设置值。（V850E/IF3：m = 2，4，V850E/IG3：m = 2 至 4）.

注意事项 **1. 当 TAAnCE 位 = 0 时，设置 TAAnCKS2 至 TAAnCKS0 位。**
当 TAAnCE 位的值从 0 改变为 1 时，可以同时设定 TAAnCKS2 至 TAAnCKS0 位。
2. 请确保将第 3 位至第 6 位清除为“0”。

备注　f_{XX}：外围时钟

（2）TAAn 控制寄存器 1（TAAnCTL1）

TAAnCTL1 寄存器是 8 位寄存器，用于控制定时器 TAAn 的操作。

该寄存器可以按字节或按位进行读取和写入。

复位输入将该寄存器设为 00H。

（1/2）

复位后：00H　　R/W　　地址：　　TAA0CTL1 FFFFF661H, TAA1CTL1 FFFFF681H,
TAA2CTL1 FFFFF6A1H, TAA3CTL1 FFFFFB01H,
TAA4CTL1 FFFFFB21H

	7	6	5	4	3	2	1	0
TAAnCTL1	TAAaSYE注1	TAAmEST注 2	TAAmEEE注2	0	0	TAAnMD2	TAAnMD1	TAAnMD0

V850E/IF3
n = 0 至 4
m = 2, 4
a = 0, 1

V850E/IG3
n = 0 至 4
m = 2 至 4
a = 0, 1

TAAaSYE注1	操作模式选择
0	TAAa 单独模式
1	调谐操作模式(参见10.4.5)
在调谐操作期间，TAAa 只可以被用作A/D 转换器 0和1的A/D 转换触发因素。 在调谐操作模式下，该位总是与TABa同步操作。	

TAAmEST注2	软件触发控制
0	-
1	位外部触发输入产生一个有效信号。 • 在单次脉冲输出模式下： 向TAAmEST位写入1作为触发器，输出一个单次脉冲。 • 在外部触发脉冲输出模式下： 向TAAmEST位写入1作为触发器，输出一个PWM波形。
TAAmEST位的读取值总是为0。	

注　**1.** 仅在 TAA0 和 TAA1 时，可以设置该位。请确保将 TAA2 至 TAA4 的第 7 位清除为“0”。
调谐操作模式的详情，参见**第十章 电机控制功能**.

2. 在 V850E/IF3 内，仅在 TAA2 和 TAA4 时，可以设置该位。请确保将 TAA0，TAA1 和 TAA3 的第 5 位及第 6 位清除为“0”。
在 V850E/IG3 内，仅在 TAA2 和 TAA4 时，可以设置该位。请确保将 TAA0 和 TAA1 的第 5 位及第 6 位清除为“0”。

（2/2）

TAAmEEE注1	计数时钟选择
0	禁止外部事件计数输入的操作(TIAm0 引脚) (执行对计数时钟的计数，该时钟由 TAAmCTL0.TAAmCKS0 至 TAAmCTL0.TAAmCKS2位选择)
1	使能外部事件计数输入的操作注2 (对外部事件计数输入信号的每个有效边沿执行计数(TIAm0引脚).)
通过TAAmEEE位选择使用内部计数时钟计数， 还是使用外部事件计数输入的有效边沿来计数。	

TAAnMD2	TAAnMD1	TAAnMD0	定时器模式选择
0	0	0	间隔定时器模式
0	0	1	外部事件计数模式
0	1	0	外部触发脉冲输出模式
0	1	1	单次触发脉冲输出模式
1	0	0	PWM 输出模式
1	0	1	自由运行定时器模式
1	1	0	脉宽测量模式
1	1	1	禁止设置

注 **1.** 使用 V850E/IF3 时，仅在 TAA2 和 TAA4 时，可以设置该位。请确保将 TAA0，TAA1 和 TAA3 的第 5 位及第 6 位清除为“0”。
使用 V850E/IG3 时，仅在 TAA2 和 TAA4 时，可以设置该位。请确保将 TAA0 和 TAA1 的第 5 位及第 6 位清除为“0”。

2. 将捕获触发输入（TIAm0 引脚）和外部触发输入（TIAm0 引脚）的有效边沿检测设置为“无边沿检测”。

注意事项 **1. 只有在外部触发脉冲输出模式或单次触发脉冲输出模式下，TAAmEST 位有效。在其他任何模式下，忽略向该位写入 1 的操作。**

2. 在外部事件计数模式下，不论 TAAmEEE 位的值如何，都选择为外部事件计数输入。

3. 当 TAAnCTL0.TAAnCE 位 = 0 时，设置 TAAaSYE，TAAmEEE 和 TAAnMD2 至 TAAnMD0 位（当 TAAnCE 位 = 1 时，可以写入相同的值）。在 TAAnCE 位 = 1 时执行写入，操作无法保证。如果重写误执行，将 TAAnCE 位清除为 0，然后再次设置该位。

4. 请确保将第 3 位和第 4 位清除为“0”。

（3）TAAm I/O 控制寄存器 0（TAAmIOC0）

TAAmIOC0 寄存器是 8 位寄存器，用于控制定时器输出（TOAm0，TOAm1 引脚）。
该寄存器可以按字节或按位进行读取和写入。
复位输入将该寄存器设为 00H。

（1/2）

复位后： 00H R/W 地址： TAA2IOC0 FFFFF6A2H, TAA3IOC0 FFFFFB02H注 1
TAA4IOC0 FFFFFB22H

	7	6	5	4	3	<2>	1	<0>
TAAmIOC0	0	0	0	0	TAAmOL1	TAAmOE1	TAAmOL0	TAAmOE0

V850E/IF3 m = 2, 4
V850E/IG3 m = 2 至 4

TAAmOL1	TOAm1 引脚输出电平设置注 2
0	TOAm1 引脚输出从高电平开始
1	TOAm1 引脚输出从低电平开始

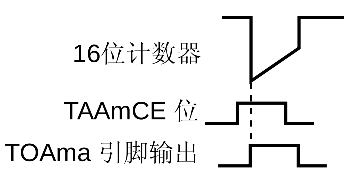
TAAmOE1	TOAm1 引脚输出设置
0	定时器输出禁止 • 当TAAmOL1 位 = 0：从TOAm1引脚输出低电平 • 当TAAmOL1 位 = 1：从TOAm1引脚输出高电平
1	定时器输出使能 (从TOAm1 引脚输出一个脉冲.)

TAAmOL0	TOAm0 引脚输出电平设置注 2
0	TOAm0 引脚输出从高电平开始
1	TOAm0 引脚输出从低电平开始

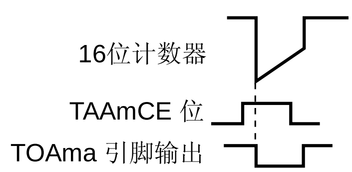
TAAmOE0	TOAm0 引脚输出设置
0	定时器输出禁止 • 当TAAmOL0位 = 0：从TOAm0引脚输出低电平 • 当TAAmOL0位 = 1：从TOAm0引脚输出高电平
1	定时器输出使能 (从TOAm0 引脚输出一个脉冲)

注 **1.** 仅限 V850E/IG3
2. 定时器输出引脚（TOAm0 和 TOAm1）的输出电平由 TAAmOLa 位指定如下（a = 0，1）。

• 当 TAAmOLa 位 = 0



• 当 TAAmOLa 位= 1



（2/2）

注意事项 **1.** 当 **TOAm0** 和 **TOAm1** 设置为输出模式时，如果 **TAAmIOC0** 寄存器的设置改变，引脚的输出改变。设置端口为输入模式，且使端口进入高阻抗状态，引脚状态不改变。

2. 当 **TAAmCTL0.TAAmCE** 位 **= 0** 时，重写 **TAAmOL1**，**TAAmOE1**，**TAAmOL0** 和 **TAAmOE0** 位（当 **TAAmCE** 位 **= 1** 时，可以写入相同的值）。如果重写误执行，将 **TAAmCE** 位清除为 **0**，然后再次设置该位。

3. 即使当操作 **TAAmOL0** 或 **TAAmOL1** 位时 **TAAmCE**，**TAAmOE0** 和 **TAAmOE1** 位是 **0**，**TOAm0** 和 **TOAm1** 引脚的输出电平也会改变。

（4）TAAm I/O 控制寄存器 1（TAAmIOC1）

TAAmIOC1 寄存器是 8 位寄存器，用于控制捕获触发输入信号（TIAm0，TIAm1 引脚）的有效边沿。

该寄存器可以按字节或按位进行读取和写入。

复位输入将该寄存器设为 00H。

复位后： 00H　R/W　地址：　TAA2IOC1 FFFFF6A3H, TAA3IOC1 FFFFFB03H[注],

TAA4IOC1 FFFFFB23H

	7	6	5	4	3	2	1	0
TAAmIOC1	0	0	0	0	TAAmIS3	TAAmIS2	TAAmIS1	TAAmIS0

（V850E/IF3 m = 2, 4）

（V850E/IG3 m = 2 至 4）

TAAmIS3	TAAmIS2	捕获触发输入信号 (TIAm1 引脚) 有效边沿设置
0	0	无边沿检测 (捕获操作无效)
0	1	检测上升沿
1	0	检测下降沿
1	1	检测双沿

TAAmIS1	TAAmIS0	捕获触发输入信号 (TIAm0 引脚) 有效边沿设置
0	0	无边沿检测 (捕获操作无效)
0	1	检测上升沿
1	0	检测下降沿
1	1	检测双沿

注　仅限 V850E/IG3

注意事项 1. 当 **TAAmCTL0.TAAmCE** 位 **= 0** 时，重写 **TAAmIS3** 至 **TAAmIS0** 位。（当 **TAAmCE** 位 **= 1** 时，可以写入相同的值）。如果重写误执行，将 **TAAmCE** 位清除为 **0**，然后再次设置该位。

2. 只有在自由运行定时器模式（仅当 **TAAmOPT0.TAAmCCS1** 和 **TAAmOPT0.TAAmCCS0** 位 **= 11**）和脉冲宽度测量模式下，**TAAmIS3** 至 **TAAmIS0** 位有效。在其它所有模式下，不可能进行捕获操作。

（5）TAAm I/O 控制寄存器 2（TAAmIOC2）

TAAmIOC2 寄存器是 8 位寄存器，用于控制外部事件计数输入信号（TIAm0 引脚）和外部触发输入信号（TIAm0 引脚）的有效边沿。

该寄存器可以按字节或按位进行读取和写入。

复位输入将该寄存器设为 00H。

复位后： 00H R/W 地址： TAA2IOC2 FFFFF6A4H, TAA3IOC2 FFFFFB04H[注],
TAA4IOC2 FFFFFB24H

	7	6	5	4	3	2	1	0
TAAmIOC2	0	0	0	0	TAAmEES1	TAAmEES0	TAAmETS1	TAAmETS0

（V850E/IF3 m = 2, 4）

（V850E/IG3 m = 2 至 4）

TAAmEES1	TAAmEES0	外部事件计数输入信号 (TIAm0 引脚) 有效边沿设置
0	0	无边沿检测 (外部事件计数无效)
0	1	检测上升沿
1	0	检测下降沿
1	1	检测双沿

TAAmETS1	TAAmETS0	外部触发输入信号(TIAm0 引脚) 有效边沿设置
0	0	无边沿检测 (外部触发无效)
0	1	检测上升沿
1	0	检测下降沿
1	1	检测双沿

注 仅限 V850E/IG3

注意事项 1. 当 **TAAmCTL0.TAAmCE** 位 **= 0** 时，重写 **TAAmEES1**，**TAAmEES0**，**TAAmETS1** 和 **TAAmETS0** 位。（当 **TAAmCE** 位 **= 1** 时，可以写入相同的值）。如果重写误执行，将 **TAAmCE** 位清除为 **0**，然后再次设置该位。

2. 只有当 **TAAmCTL1.TAAmEEE** 位 **= 1** 或设置为外部事件计数模式（**TAAmCTL1.TAAmMD2** 至 **TAAmCTL1.TAAmMD0** 位 **= 001**）时，**TAAmEES1** 和 **TAAmEES0** 位有效。

3. 只有在外部触发脉冲模式或单次触发脉冲输出模式下，**TAAmETS1** 和 **TAAmETS0** 位有效。

（6）TAAn 选项寄存器 0（TAAnOPT0）

TAAnOPT0 寄存器是 8 位寄存器，用于设置捕获/比较操作并检测溢出。

该寄存器可以按字节或按位进行读取和写入。

复位输入将该寄存器设为 00H。

复位后 00H R/W 地址： TAA0OPT0 FFFFF665H, TAA1OPT0 FFFFF685H, TAA2OPT0 FFFFF6A5H, TAA3OPT0 FFFFFB05H, TAA4OPT0 FFFFFB25H

	7	6	5	4	3	2	1	<0>
TAAnOPT0	0	0	TAAmCCS1[注]	TAAmCCS0[注]	0	0	0	TAAnOVF

V850E/IF3
n = 0 至 4
m = 2, 4

V850E/IG3
n = 0 至 4
m = 2 至 4

TAAmCCS1[注]	TAAmCCR1 寄存器捕获/比较选择
0	选择比较寄存器
1	选择捕获寄存器 (由TAAmCTL0.TAAmCE 位 = 0清除)
只有在自由运行定时器模式下，TAAmCCS1位的设置有效	

TAAmCCS0[注]	TAAmCCR0 寄存器捕获/比较选择
0	比较寄存器选择
1	选择捕获寄存器 (由TAAmCTL0.TAAmCE 位 = 0清除)
只有在自由运行定时器模式下，TAAmCCS0位的设置有效	

TAAnOVF	TAAn 溢出检测标志
置位 (1)	发生溢出
复位 (0)	TAAnOVF 位写入0 ，或TAAnCTL0.TAAnCE 位 = 0

- 在自由运行定时器模式下，当16 位计数器的计数值从FFFFH 溢出到 0000H时，TAAnOVF位被置为 1 。
- 在TAAnOVF位被置为 1 的同时，产生一个中断请求信号 (INTTAnOV) 。在自由运行定时器模式和脉宽测量模式以外的模式时，不产生INTTAnOV信号。
- 当TAAnOVF 位 = 1时，即使TAAnOVF位或TAAnOPT0寄存器被读取，TAAnOVF 位也不会被清除。
- 在 TAAnOVF 位清除为 0之前， INTTAnOV 信号产生之后，请确认 (通过读取) TAAnOVF 位被设置为 1.
- TAAnOVF位可读可写，但是不能通过软件将TAAnOVF位位置1 。在TAAn工作期间写入1 没有影响。

注　在 V850E/IF3 内，仅在 TAA2 和 TAA4 时，可以设置该位。请确保将 TAA0，TAA1 和 TAA3 的第 4 位及第 5 位清除为“0”。

在 V850E/IG3 内，仅在 TAA2 和 TAA4 时，可以设置该位。请确保将 TAA0 和 TAA1 的第 4 位及第 5 位清除为“0”。

注意事项 1.　当 TAAmCE 位 ＝ 0 时，重写 TAAmCCS1 和 TAAmCCS0 位。（当 TAAmCE 位 ＝ 1 时，可以写入相同的值）。如果重写误执行，将 TAAmCE 位清除为 0，然后再次设置该位。

2.　请确保将第 1 位至第 3 位、第 6 位和第 7 位清除为“0”。

（7）TAAn 捕获/比较寄存器 0（TAAnCCR0）

TAAmCCR0 寄存器是 16 位寄存器，根据不同工作模式，该寄存器可以用作捕获寄存器或比较寄存器。

TAAkCCR0 寄存器是只能用作比较寄存器的 16 位寄存器。

在自由计时器模式下，根据 TAAmOPT0.TAAmCCS0 位的设定，TAAnCCR0 寄存器可用作捕获寄存器或比较寄存器。在脉宽测量模式中，TAAmCCR0 寄存器只能用作捕获寄存器。而在其它模式时，该寄存器只能用作比较寄存器。

TAAnCCR0 寄存器可以在操作中进行读取和写入。

此寄存器可以按 16 位宽度进行读取或写入。

复位输入将该寄存器清为 0000H。

备注　V850E/IF3：n = 0 至 4，m = 2，4，k = 0，1，3
　　　V850E/IG3：n = 0 至 4，m = 2 至 4，k = 0，1

复位后：0000H　R/W　地址：TAA0CCR0 FFFFF666H, TAA1CCR0 FFFFF686H,
TAA2CCR0 FFFFF6A6H, TAA3CCR0 FFFFFB06H,
TAA4CCR0 FFFFFB26H

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TAAnCCR0 (n = 0 至 4)																

（a）作为比较寄存器时的功能

即便在 TAAnCTL0.TAAnCE 位=1 时，TAAnCCR0 寄存器仍可以重新写入。

TAAnCCR0 寄存器的设定值传输给 CCR0 缓冲寄存器。当 16 位计数器的值与 CCR0 缓冲寄存器的值相一致时，就会生成一个比较一致中断请求信号（INTTAnCC0）。如果此时 TOAm0 引脚输出允许，则 TOAm0 引脚的输出会反相。

在间隔定时器模式下当 TAAnCCR0 寄存器用作周期寄存器时，或在外部事件计数器模式、外部触发脉冲输出模式、单次触发脉冲输出模式或 PWM 输出模式下，TAAmCCR0 寄存器用作周期寄存器时，如果 16 位计数器的计数值与 CCR0 缓冲寄存器数值一致，则计数器清除为（0000H）。

将 TAAnCTL0.TAAnCE 位设置为 0，不会清除比较寄存器。

（b）用作捕获寄存器时的功能

在自由运行定时器模式下，当 TAAmCCR0 寄存器用作捕获寄存器时，如果检测到捕获触发输入引脚（TIAm0 引脚）的有效边沿，则 16 位计数器的当前计数值存储于 TAAmCCR0 寄存器中。在脉宽测量模式下，若检测到捕获触发输入引脚（TIAm0 引脚）的有效边沿，16 位计数器的当前计数值就储存在 TAAmCCR0 寄存器中，且 16 位计数器清零（0000H）。

即使捕获操作和读取 TAAmCCR0 寄存器的操作发生冲突，还是可以正确读取 TAAmCCR0 寄存器的值。

将 TAAmCTL0.TAAmCE 位设置为 0，不会清除捕获寄存器。

备注　V850E/IF3：n = 0 至 4，m = 2，4
　　　V850E/IG3：n = 0 至 4，m = 2 至 4

下表显示了捕获/比较寄存器在各工作模式下的功能，以及如何向比较寄存器写入数据。

表 6-4. 捕获/比较寄存器在各模式下的功能以及如何写入比较寄存器

操作模式	捕获/比较寄存器	如何写入比较寄存器
间隔定时器	比较寄存器	随时写入
外部事件计数器注 1	比较寄存器	随时写入
外部触发脉冲输出注 1	比较寄存器	批量写入注 2
单次触发脉冲输出注 1	比较寄存器	随时写入
PWM输出注 1	比较寄存器	批量写入注 2
自由运行定时器	捕获/比较寄存器	随时写入
脉冲宽度测量注 1	捕获寄存器	无

注　**1.** 使用 V850E/IF3 时，只有 TAA2 和 TAA4 可用此模式。使用 V850E/IG3 时，只有 TAA2 至 TAA4 可用此模式。

　　2. 向 TAAmCCR1 寄存器写入是触发器。

备注　关于随时写入和批量写入，参见 **6.6（2）随时写入和批量写入**。

（8）TAAn 捕获/比较寄存器 1（TAAnCCR1）

TAAmCCR1 寄存器是 16 位寄存器，根据不同工作模式，该寄存器可以用作捕获寄存器或比较寄存器。
TAAkCCR1 寄存器是只能用作比较寄存器的 16 位寄存器。

在自由计时器模式下，根据 TAAmOPT0.TAAmCCS1 位的设定，TAAmCCR1 寄存器可用作捕获寄存器或比较寄存器。在脉宽测量模式中，TAAmCCR1 寄存器只能用作捕获寄存器。而在其它模式时，该寄存器只能用作比较寄存器。

TAAnCCR1 寄存器可以在操作中进行读取和写入。

该寄存器可以按 16 位宽度进行读取或写入。

复位输入将该寄存器清为 0000H。

备注　V850E/IF3：n = 0 至 4，m = 2，4，k = 0，1，3
　　　V850E/IG3：n = 0 至 4，m = 2 至 4，k = 0，1

复位后：0000H　　R/W　　地址：　TAA0CCR1 FFFFF668H, TAA1CCR1 FFFFF688H,
TAA2CCR1 FFFFF6A8H, TAA3CCR1 FFFFFB08H,
TAA4CCR1 FFFFFB28H

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TAAnCCR1 (n = 0 至 4)																

（a）用作比较寄存器时的功能

即使 TAAnCTL0.TAAnCE 位=1 时，TAAnCCR1 寄存器仍可以重新写入。

TAAnCCR1 寄存器的设定值传输给 CCR1 缓冲寄存器。当 16 位计数器的值与 CCR1 缓冲寄存器的值相一致时，就会生成一个比较一致中断请求信号（INTTAnCC1）。如果此时 TOAm1 引脚输出允许，则 TOAm1 引脚的输出会反相。

将 TAAnCTL0.TAAnCE 位设置为 0，不会清除比较寄存器。

（b）用作捕获寄存器时的功能

在自由运行定时器模式下，当 TAAmCCR1 寄存器用作捕获寄存器时，若检测到捕获触发输入引脚（TIAm1 引脚）的有效边沿，则 16 位计数器的当前计数值存储于 TAAmCCR1 寄存器中。在脉宽测量模式下，若检测到捕获触发输入引脚（TIAm1 引脚）的有效边沿，16 位计数器的当前计数值就储存在 TAAmCCR1 寄存器中，且 16 位计数器清零（0000H）。

即使捕获操作和读取 TAAmCCR1 寄存器的操作发生冲突，还是可以正确读取 TAAmCCR1 寄存器的值。

将 TAAmCTL0.TAAmCE 位设置为 0，不会清除捕获寄存器。

备注 V850E/IF3：n = 0 至 4，m = 2，4
V850E/IG3：n = 0 至 4，m = 2 至 4

下表显示了捕获/比较寄存器在各工作模式下的功能，以及如何向比较寄存器写入数据。

表 6-5. 捕获/比较寄存器在各模式下的功能以及如何写入比较寄存器

操作模式	捕获/比较寄存器	如何写入比较寄存器
间隔定时器	比较寄存器	随时写入
外部事件计数器[注 1]	比较寄存器	随时写入
外部触发脉冲输出[注 1]	比较寄存器	批量写入[注 2]
单次触发脉冲输出[注 1]	比较寄存器	随时写入
PWM输出[注 1]	比较寄存器	批量写入[注 2]
自由运行定时器	捕获/比较寄存器	随时写入
脉冲宽度测量[注 1]	捕获寄存器	无

注 **1.** 使用 V850E/IF3 时，只有 TAA2 和 TAA4 可用此模式。使用 V850E/IG3 时，只有 TAA2 至 TAA4 可用此模式。

2. 向 TAAmCCR1 寄存器写入是触发器。

备注 关于随时写入和批量写入，参见 **6.6（2）随时写入和批量写入**。

（9）TAAn 计数器读取缓冲寄存器（TAAnCNT）

TAAnCNT 寄存器是读取缓冲寄存器，可以读取 16 位计数器的计数值。

如果在 TAAnCTL0.TAAnCE 位＝1 时读取该寄存器，可以读出 16 位定时器的计数值。

该寄存器为只读寄存器，按 16 位宽度读取。

当 TAAnCE 位=0 时，TAAnCNT 寄存器的值被清除为 0000H。如果此时读取 TAAnCNT 寄存器，则不会读到 16 位计数器的值（FFFFH），读到的值为 0000H。

复位后，TAAnCNT 寄存器被清除为 0000H，且 TAAnCE 位被清除为 0。

复位后：0000H　　R　　地址：　TAA0CNT FFFFF66AH, TAA1CNT FFFFF68AH,
TAA2CNT FFFFF6AAH, TAA3CNT FFFFFB0AH,
TAA4CNT FFFFFB2AH

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TAAnCNT (n = 0 至 4)																

6.5 定时器输出操作

下表展示了 TOAm0 和 TOAm1 引脚的操作和输出电平。

表 6-6. 各个模式下的定时器输出控制

操作模式	TOAm1 引脚	TOAm0 引脚
间隔定时器模式	PWM 输出	
外部事件计数模式	无	
外部触发脉冲输出模式	外部触发脉冲输出	PWM 输出
单次触发脉冲输出模式	单次触发脉冲输出	
PWM 输出模式	PWM 输出	
自由运行定时器模式	PWM 输出（仅当使用比较功能时）	
脉冲宽度测量模式	无	

备注 V850E/IF3：m = 2，4
V850E/IG3：m = 2 至 4

表 6-7. 定时器输出控制位控制下的 TOAm0 和 TOAm1 引脚的真值表

TAAmIOC0.TAAmOLa 位	TAAmIOC0.TAAmOEa 位	TAAmCTL0.TAAmCE 位	TOAma 引脚的电平
0	0	×	低电平输出
	1	0	低电平输出
		1	计数之前的瞬间低电平，计数开始之后高电平
1	0	×	高电平输出
	1	0	高电平输出
		1	计数之前的瞬间高电平，计数开始之后低电平

备注 V850E/IF3：m = 2，4，a = 0，1
V850E/IG3：m = 2 至 4，a = 0，1

6.6 操作

各个通道的 TAAn 可以实现的功能不同，各个通道的功能展示如下。

表 6-8. 各个模式下 TAA0 和 TAA1 说明

操作	软件触发位	外部触发输入	捕获/比较寄存器设置	比较寄存器写入方法
间隔定时器模式	无效	无效	比较专用	随时写入
外部事件计数模式	无			
外部触发脉冲输出模式	无			
单次触发脉冲输出模式	无			
PWM 输出模式	无			
自由运行定时器模式	无效	无效	比较专用	随时写入
脉冲宽度测量模式	无			

备注
1. TAAa 不具有定时器输入引脚（TIAa0，TIAa1）和定时器输出引脚（TOAa0，TOAa1）。当 16 位计数器的值与 TAAaCCR0 和 TAAaCCR1 寄存器的值一致时，它有中断请求信号（INTTAaCC0，INTTAaCC1）。
2. TAAa 具有和 TABa 执行调谐的功能。详情参见**第十章 电机控制功能**.
3. a = 0，1

表 6-9. 各个模式下 TAA2 至 TAA4 说明

操作	TAAmCTL1.TAAmEST 位 （软件触发位）	TIAm0 引脚 （外部触发输入）	捕获/比较寄存器设置	比较寄存器写入方法
间隔定时器模式	无效	无效	比较专用	随时写入
外部事件计数模式注 1	无效	无效	比较专用	随时写入
外部触发脉冲输出模式注 2	有效	有效	比较专用	批量写入
单次触发脉冲输出模式注 2	有效	有效	比较专用	随时写入
PWM 输出模式	无效	无效	比较专用	批量写入
自由运行定时器模式	无效	无效	可切换注 3	随时写入
脉冲宽度测量模式注 2	无效	无效	仅作捕获	不适用

注
1. 当使用外部事件计数模式时，将 TIAm0 引脚捕获触发输入有效边沿设置为“无边沿检测”（将 TAAmIOC1.TAAmIS1 和 TAAmIOC1.TAAmIS0 位清除为 00）。
2. 当使用外部触发脉冲输出模式，单次触发脉冲输出模式和脉冲宽度测量模式时，选择内部时钟作为计数时钟（将 TAAmCTL1.TAAmEEE 位清除为 0）。
3. 在 V850E/IF3 内，该设置是比较专用。

备注 V850E/IF3：m = 2，4
V850E/IG3：m = 2 至 4

（1）计数器基本操作

本节介绍 16 位计数器的基本操作。详情参见各个模式下的操作描述。

备注　V850E/IF3：n = 0 至 4，m = 2，4
　　　V850E/IG3：n = 0 至 4，m = 2 至 4

（a）　计数器开始操作

<R> • 在外部事件计数模式下
  当 TAAmCTL0.TAAmCE 位从 0 设置为 1 时，16 位计数器被置为 0000H。
  此后，每次检测到外部事件计数输入（TIAm0）的有效边沿时，向上计数到 0001H，0002H，0003H，…
- 在其它模式下
  从默认值 FFFFH 开始计数。
  从 FFFFH 向上计数到 0000H，0001H，0002H，0003H，如此继续。

（b）　清除操作

当 16 位计数器的值或捕获寄存器和比较寄存器的值一致时，其值被清除为 0000H。在计数器开始计数之后立即发生从 FFFFH 到 0000H 的计数操作，或当计数器溢出不是清除操作时。因此，不产生 INTTAnCC0 和 INTTAnCC1 中断信号。

（c）　溢出操作

在自由运行模式或脉冲宽度测量模式下，当计数器从 FFFFH 向上计数到 0000H 时，16 位计数器溢出。如果计数器溢出，TAAnOPT0.TAAnOVF 位被设置为 1，且产生一个中断请求信号（INTTAnOV）。请注意在下列条件时不产生 INTTAnOV 信号。

- 在计数操作刚刚开始之后
- 如果计数器值和比较值 FFFFH 一致并清除
- 在脉冲宽度测量模式下，当捕获到 FFFFH 并清除时，且计数器从 FFFFH 向上计数到 0000H

注意事项　**在产生溢出中断请求信号（INTTAnOV）之后，请确保检查溢出标志（TAAnOVF 位）被设置为 1。**

（d）　在计数操作期间的计数器读取操作

在计数操作期间，TAAn 的 16 位计数器的值可以通过 TAAnCNT 寄存器来读取。当 TAAnCTL0.TAAnCE 位 = 1 时，16 位计数器的值可以通过 TAAnCNT 寄存器来读取。当 TAAnCTL0.TAAnCE 位 = 0 时，16 位计数器是 FFFFH 且 TAAnCNT 寄存器是 0000H。

（e） 中断操作

TAAn 产生下列三种类型的中断请求信号。

- INTTAnCC0 中断： 该信号用作 CCR0 缓冲寄存器的一致中断请求信号和 TAAnCCR0 寄存器的捕获中断请求信号。
- INTTAnCC1 中断： 该信号用作 CCR1 缓冲寄存器的一致中断请求信号和 TAAnCCR1 寄存器的捕获中断请求信号。
- INTTAnOV 中断： 该信号用作溢出中断请求信号。

（2）随时写入和批量写入

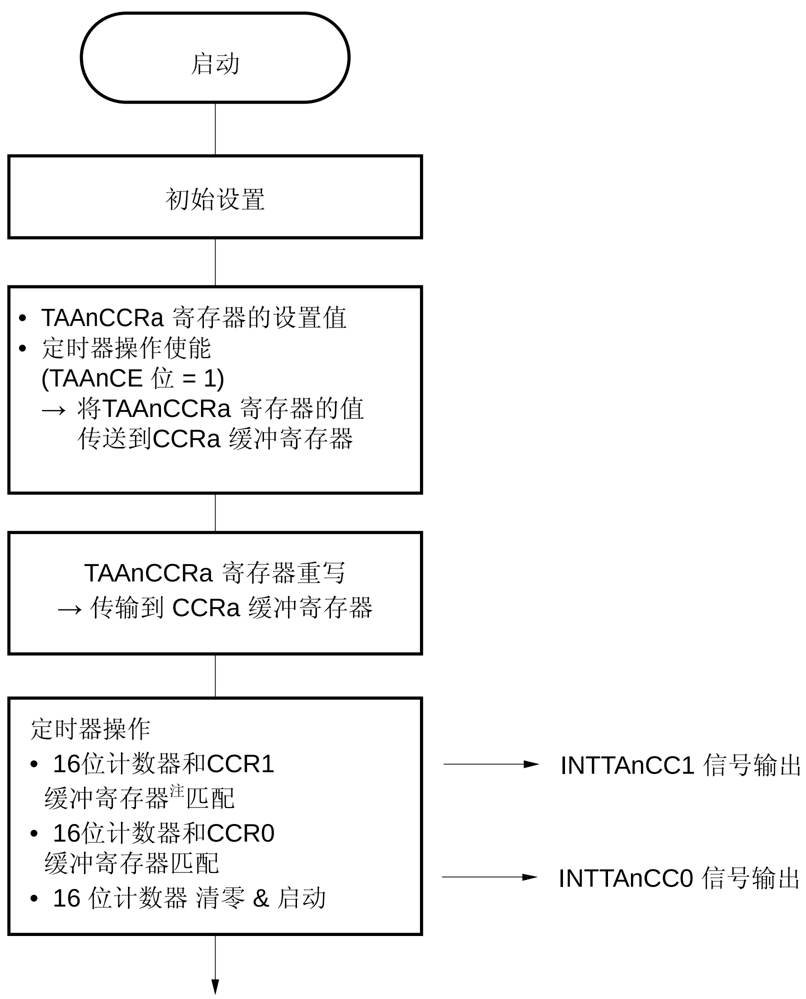
在定时器操作期间（TAAnCTL0.TAAnCE 位 = 1），TAAn 的 TAAnCCR0 和 TAAnCCR1 寄存器可以重写，但是 CCR0 和 CCR1 缓冲寄存器的写入方法（随时写入，批量写入）取决于不同的模式。

（a） 随时写入

该模式下，在定时器操作期间，数据随时可以从 TAAnCCR0 和 TAAnCCR1 寄存器传输到 CCR0 和 CCR1 缓冲寄存器。

备注 n = 0 至 4

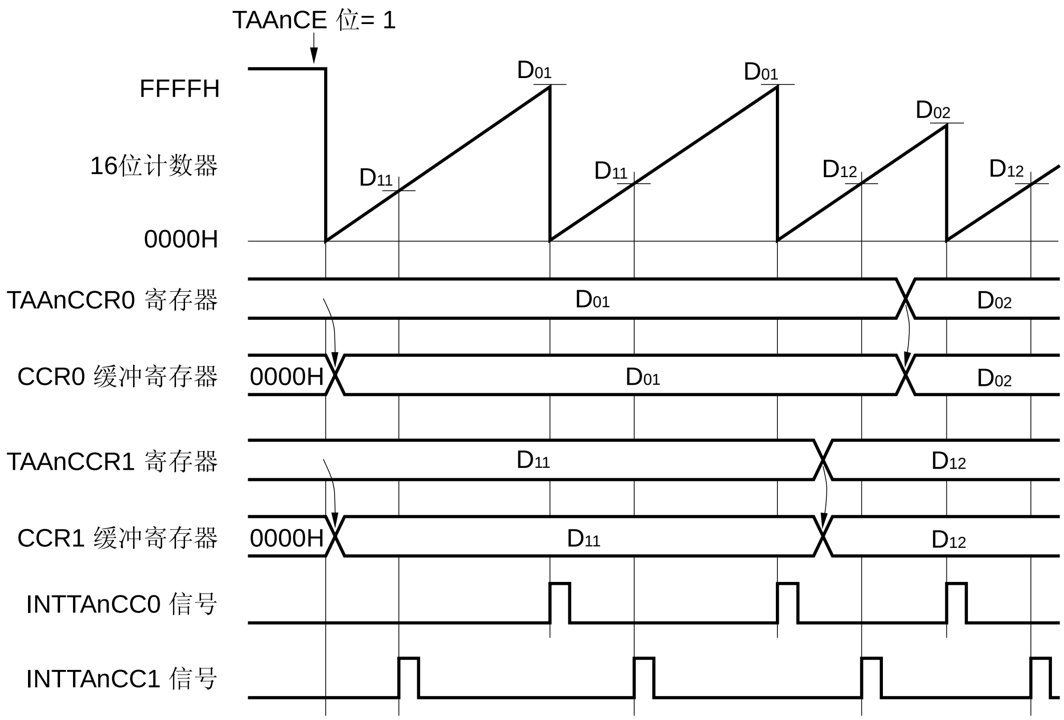
图 6-6. 随时写入基本操作的流程图



注 在 16 位计数器的值和 CCR1 缓冲寄存器的值一致时，不清除 16 位计数器。在 16 位计数器的值和 CCR0 缓冲寄存器的值一致时，才清除 16 位计数器。

备注 **1.** 上述流程图展示了一个间隔定时器模式下的操作示例。
2. n = 0 至 4
a = 0，1

图 **6-7.** 随时写入的时序



备注 **1.** D_{01}，D_{02}：TAAnCCR0 寄存器的设置值

D_{11}，D_{12}：TAAnCCR1 寄存器的设置值

2. 上述流程图展示了一个间隔定时器模式下的操作示例。

3. n = 0 至 4

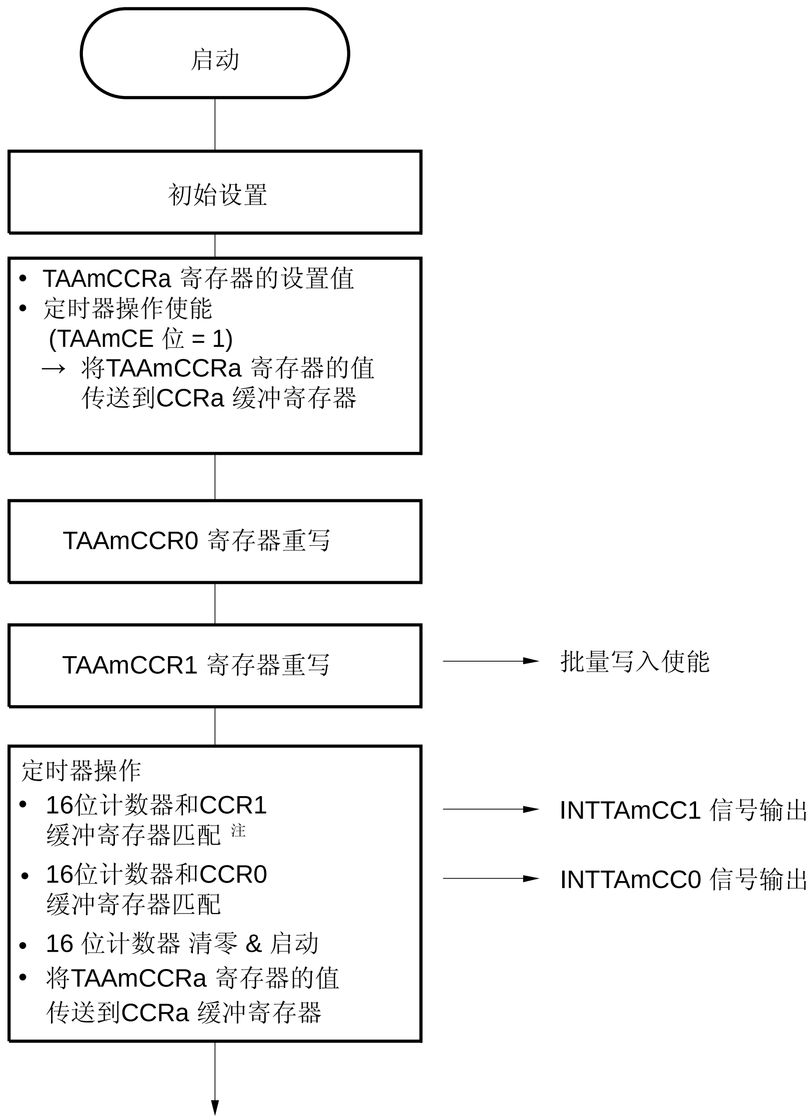
（b） 批量写入

该模式下，在定时器操作期间，数据一次性从 TAAmCCR0 和 TAAmCCR1 寄存器传输到 CCR0 和 CCR1 缓冲寄存器。当 16 位计数器的值和 CCR0 缓冲寄存器的值一致时，传输该数据。传输许可通过写入 TAAmCCR1 寄存器完成。是否写入 TAAmCCR1 寄存器可以控制下一次传输时序有效或无效。

为了让重写 TAAmCCR0 和 TAAmCCR1 寄存器的设置值成为 16 位计数器比较值（换句话说，为了该值被传输到 CCR0 和 CCR1 缓冲寄存器），必需重写 TAAmCCR0 寄存器，然后在 16 位计数器的值和 CCR0 缓冲寄存器的值一致之前写入 TAAmCCR1 寄存器。因此，在 16 位计数器的值和 CCR0 缓冲寄存器的值一致时，TAAmCCR0 和 TAAmCCR1 寄存器的值传输到 CCR0 和 CCR1 缓冲寄存器。所以，即使只想重写 TAAmCCR0 寄存器的值，也需要将相同的值（和 TAAmCCR1 寄存器的当前值相同）写入 TAAmCCR1 寄存器。

备注 V850E/IF3：m = 2，4
V850E/IG3：m = 2 至 4

图 6-8. 批量写入基本操作的流程图



注　在 16 位计数器的值和 CCR1 缓冲寄存器的值一致时，不清除 16 位计数器。在 16 位计数器的值和 CCR0 缓冲寄存器的值一致时，才清除 16 位计数器。

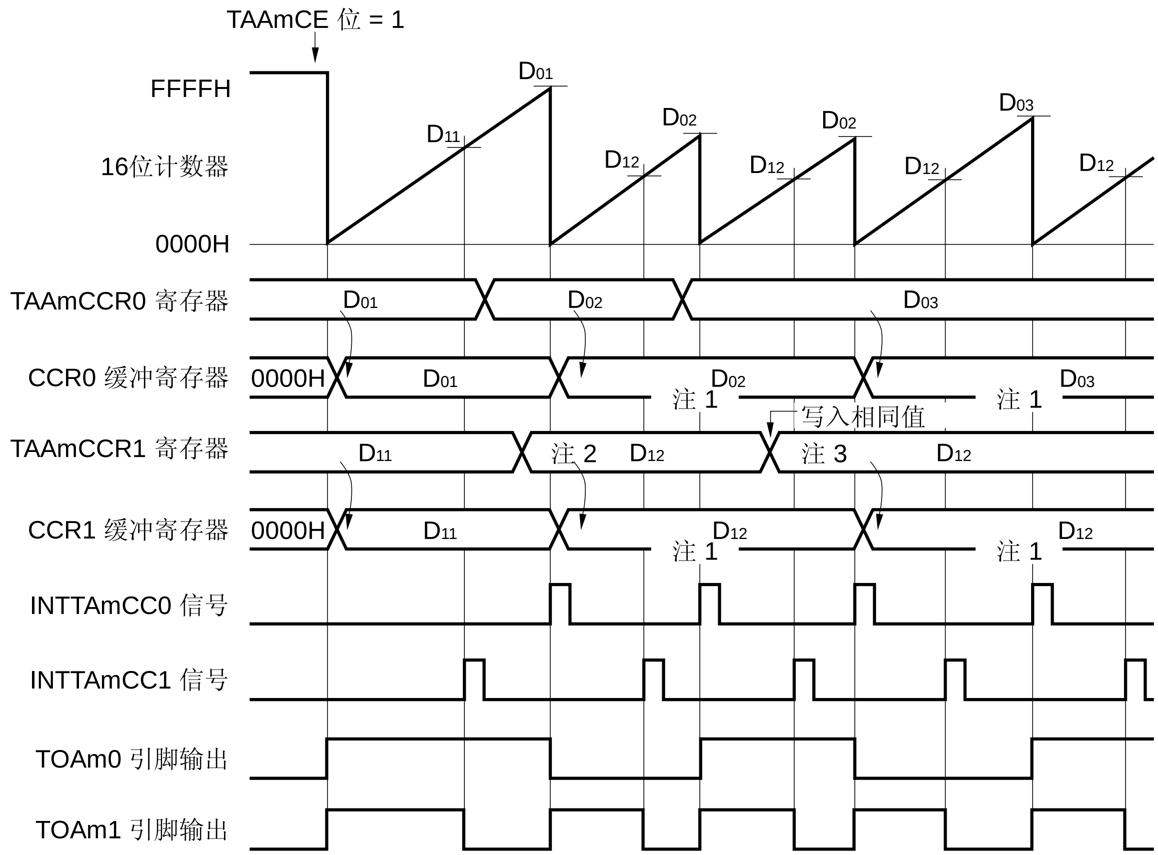
注意事项　TAAmCCR1 寄存器写入包括允许批量写入。于是，在重写 TAAmCCR0 寄存器之后，重写 TAAmCCR1 寄存器。

备注　**1.** 上述流程图展示了一个 PWM 输出模式下的操作示例。

2. V850E/IF3：m = 2，4，a = 0，1

V850E/IG3：m = 2 至 4，a = 0，1

图 **6-9.** 批量写入的时序



注　**1.** 因为 TAAmCCR1 寄存器没有重写，D_{03} 不传输。

2. 因为 TAAmCCR1 寄存器已经写入（D_{12}），当 16 位计数器的值和 TAAmCCR0 寄存器的值（D_{01}）一致时，数据传输到 CCR1 缓冲寄存器。

3. 因为 TAAmCCR1 寄存器已经写入（D_{12}），当 16 位计数器的值和 TAAmCCR0 寄存器的值（D_{02}）一致时，数据传输到 CCR1 缓冲寄存器。

备注　**1.** D_{01}，D_{02}，D_{03}：　TAAmCCR0 寄存器的设置值

D_{11}，D_{12}：　TAAmCCR1 寄存器的设置值

2. 上述流程图展示了一个 PWM 输出模式下的操作示例。

3. V850E/IF3：m = 2，4

V850E/IG3：m= 2 至 4

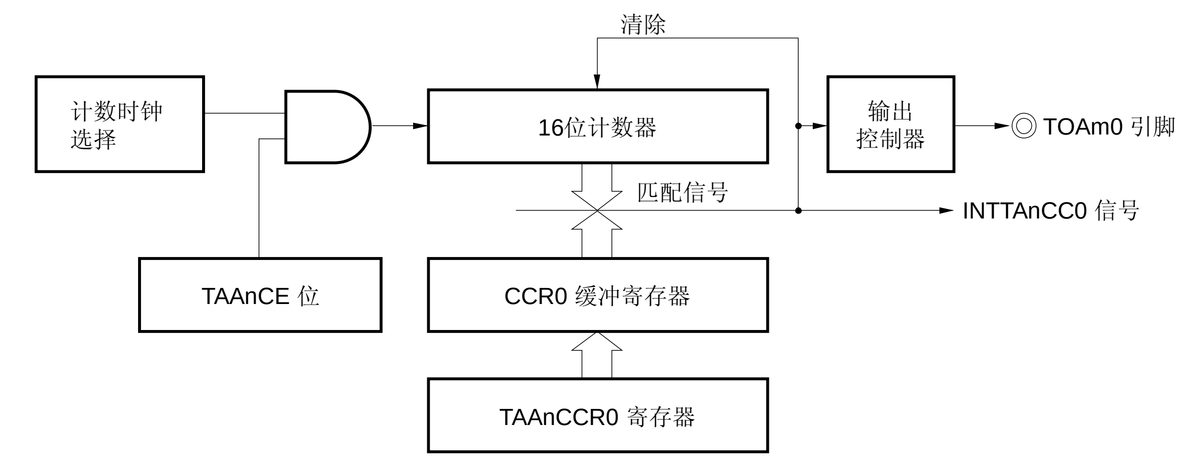
6.6.1 间隔定时器模式（TAAnMD2 至 TAAnMD0 位 = 000）

在间隔定时器模式下，如果 TAAnCTL0.TAAnCE 位设置为 1，按照 TAAnCCR0 寄存器设置的间隔，产生中断请求信号（INTTAnCC0）。可以从 TOAm0 引脚输出一个占空比 50%的 PWM 波形，其半周期等于该间隔。

间隔定时器模式不使用 TAAnCCR1 寄存器。但是，当 TAAnCCR1 寄存器的设置值传输到 CCR1 缓冲寄存器，16 位计数器的计数值和 CCR1 缓冲寄存器的值一致时，产生一个一致中断请求信号（INTTAnCC1）。另外，可以从 TOAm1 引脚输出一个占空比 50%的 PWM 波形，产生 INTTAmCC1 信号时，其波形反转。

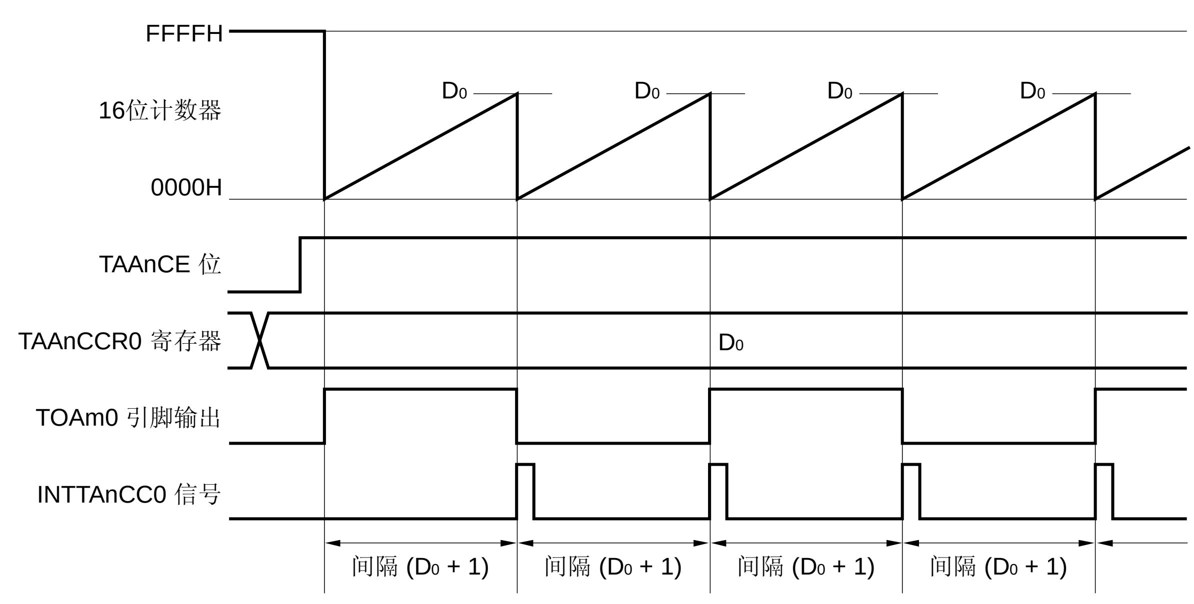
即使在定时器工作时，TAAnCCR0 和 TAAnCCR1 寄存器的值可以重写。

图 6-10. 间隔定时器的配置



备注　V850E/IF3：n = 0 至 4，m = 2，4
　　　V850E/IG3：n = 0 至 4，m = 2 至 4

图 6-11. 间隔定时器模式下的基本时序



备注　V850E/IF3：n = 0 至 4，m = 2，4
　　　V850E/IG3：n = 0 至 4，m = 2 至 4

当 TAAnCE 位置为 1 时，16 位计数器的值与计数时钟同步从 FFFFH 清除为 0000H，计数器开始计数。此时，TOAm0 引脚的输出反相。此外，TAAnCCR0 寄存器的设定值传输给 CCR0 缓冲寄存器。

当 16 位计数器的计数值和 CCR0 缓冲寄存器的值相一致时，则 16 位计数器清为 0000H，TOAm0 引脚输出反相，并生成一个一致的中断请求信号（INTTAnCC0）。

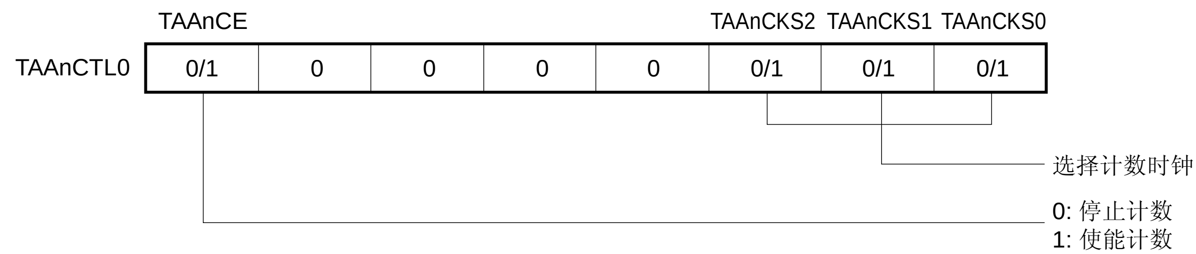
时间间隔可用下述表达式计算。

间隔 =（TAAnCCR0 寄存器的设置值 + 1）× 计数时钟周期

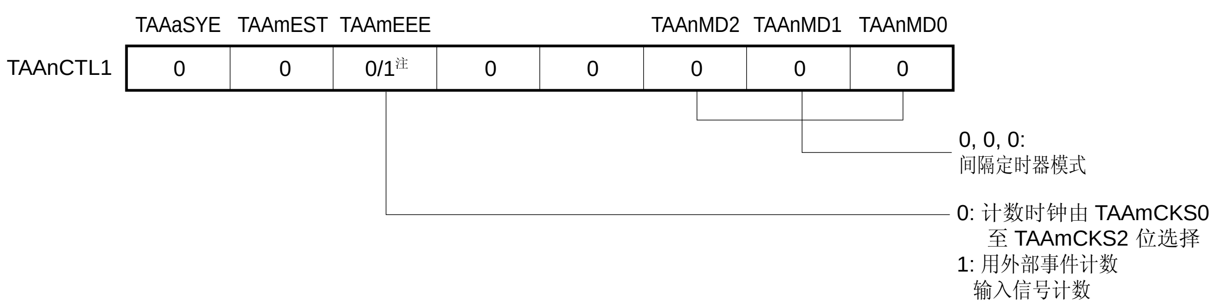
备注　V850E/IF3：n = 0 至 4，m = 2，4
　　　V850E/IG3：n = 0 至 4，m = 2 至 4

图 6-12. 间隔定时器模式操作的寄存器设置（1/3）

（a）　TAAn 控制寄存器 0（TAAnCTL0）



（b）　TAAn 控制寄存器 1（TAAnCTL1）



注　仅当使用定时器输出（TOAm1）时，TAAmEEE 位可以设置为 1。但是，设置 TAAmCCR0 和 TAAmCCR1 寄存器为相同值。

图 6-12. 间隔定时器模式操作的寄存器设置（2/3）

（c） TAAm I/O 控制寄存器 0（TAAmIOC0）

					TAAmOL1	TAAmOE1	TAAmOL0	TAAmOE0
TAAmIOC0	0	0	0	0	0/1	0/1	0/1	0/1

TAAmOE0：
0: 禁止 TOAm0 引脚输出
1: 使能 TOAm0 引脚输出

TAAmOL0：
在计数操作前，
设置TOAm0 引脚输出电平
0: 低电平
1: 高电平

TAAmOE1：
0: 禁止 TOAm1 引脚输出
1: 使能 TOAm1 引脚输出

TAAmOL1：
在计数操作前，
设置TOAm1 引脚输出电平
0: 低电平
1: 高电平

（d） TAAm I/O 控制寄存器 2（TAAmIOC2）

					TAAmEES1	TAAmEES0	TAAmETS1	TAAmETS0
TAAmIOC2	0	0	0	0	0/1[注]	0/1[注]	0	0

TAAmEES1、TAAmEES0：
选择外部事件计数输入
(TIAm0 引脚)的有效边沿

注 仅当使用定时器输出（TOAm1）时，可以设置 TAAmEES1 和 TAAmEES0 位。但是，设置 TAAmCCR0 和 TAAmCCR1 寄存器为相同值。

（e） TAAn 计数器读取缓冲寄存器（TAAnCNT）

通过读取 TAAnCNT 寄存器，可以读得 16 位计数器的计数值。

（f） TAAn 捕获/比较寄存器 0（TAAnCCR0）

如果 TAAnCCR0 寄存器被设置为 D_0，间隔如下。

间隔 =（D_0 + 1）× 计数时钟周期

图 6-12. 间隔定时器模式操作的寄存器设置（3/3）

（g） TAAn 捕获/比较寄存器 1（TAAnCCR1）

间隔定时器模式不使用 TAAnCCR1 寄存器。但是，TAAnCCR1 寄存器的设置值传输到 CCR1 缓冲寄存器。当 16 位计数器的计数值和 CCR1 缓冲寄存器的值一致时，TOAm1 引脚输出被反转并产生一个一致的中断请求信号（INTTAnCC1）。

通过设置该寄存器的值和 TAAnCCR0 寄存器的设置值相等，可以从 TOAm1 引脚输出一个占空比 50%的 PWM 波形。

不使用 TAAnCCR1 寄存器时，推荐将其值设置为 FFFFH。并且通过中断屏蔽标志（TAAnCCIC1.TAAnCCMK1）屏蔽该寄存器。

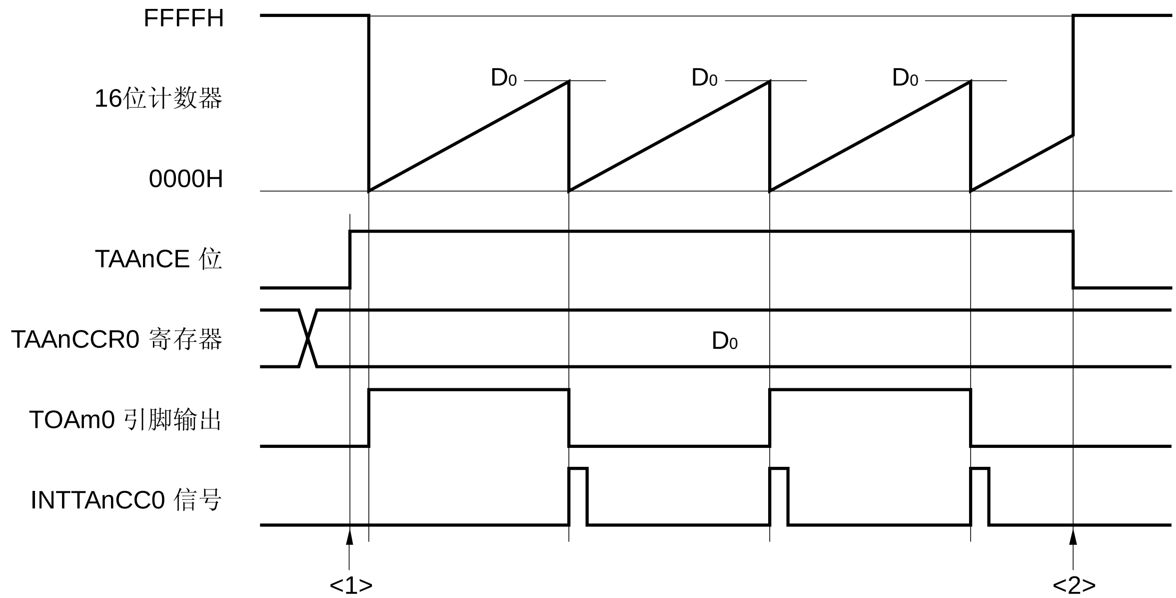
备注 **1.** 间隔定时器模式不使用 TAAm I/O 控制寄存器 1（TAAmIOC1）和 TAAn 选项寄存器 0（TAAnOPT0）。

2. V850E/IF3：n = 0 至 4，m = 2，4，a = 0，1

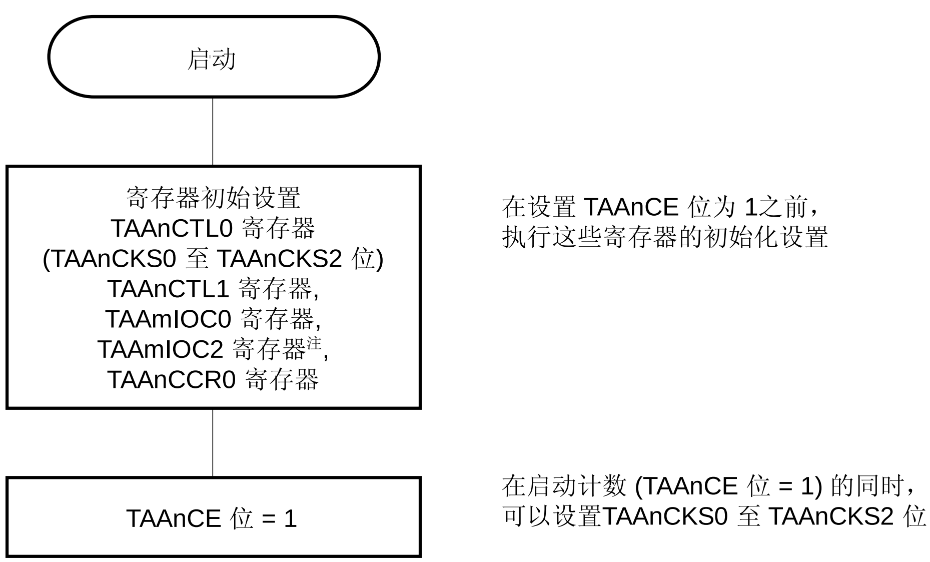
V850E/IG3：n = 0 至 4，m = 2 至 4，a = 0，1

（1） 间隔定时器模式操作流程

图 6-13. 间隔定时器模式下的软件处理流程（1/2）



<1> 计数操作启动流程

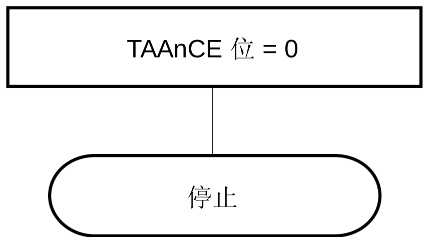


注 仅当使用定时器输出（TOAm1）时，可以设置 TAAmEES1 和 TAAmEES0 位。但是，设置 TAAmCCR0 和 TAAmCCR1 寄存器为相同值。

备注 V850E/IF3：n = 0 至 4，m = 2，4
V850E/IG3：n = 0 至 4，m = 2 至 4

图 **6-13.** 间隔定时器模式下的软件处理流程（**2/2**）

<2> 计数操作停止流程



计数器被初始化，并且
将 TAAnCE 位清除为 0，
停止计数
TOAm0 引脚的输出电平
由 TAAmIOC0 寄存器指定

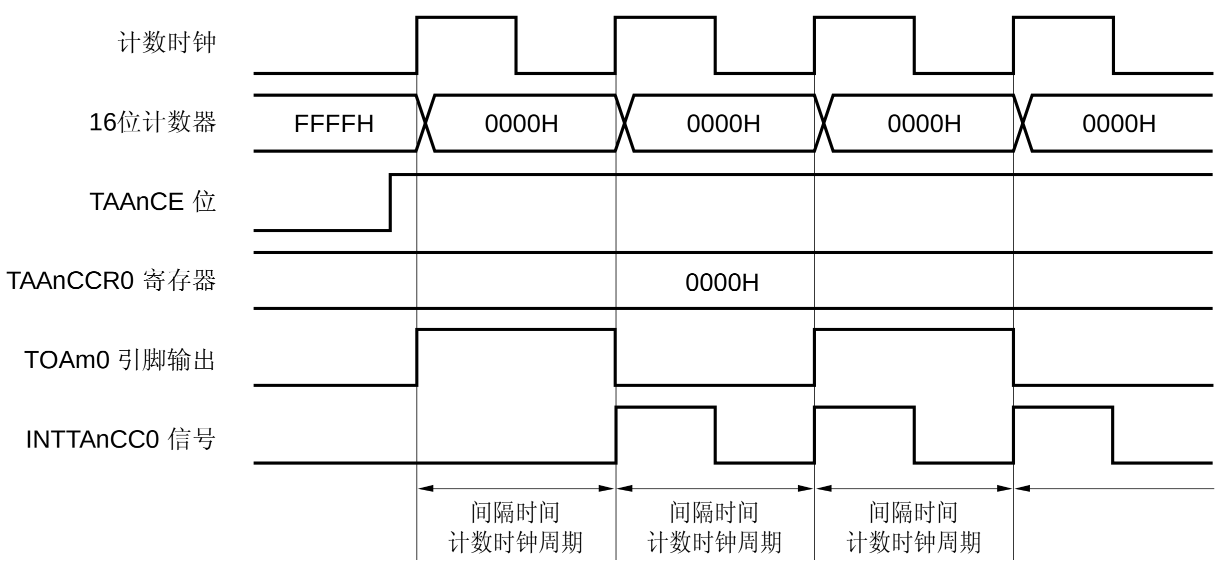
备注 V850E/IF3：n = 0 至 4，m = 2，4
V850E/IG3：n = 0 至 4，m = 2 至 4

（**2**） 间隔定时器模式操作时序

（**a**） 如果 **TAAnCCR0** 寄存器被设置为 **0000H** 时的操作

如果 TAAnCCR0 寄存器被清为 0000H，每个计数时钟都会产生 INTTAnCC0 信号，而且，TOAm0 引脚的输出被反转。

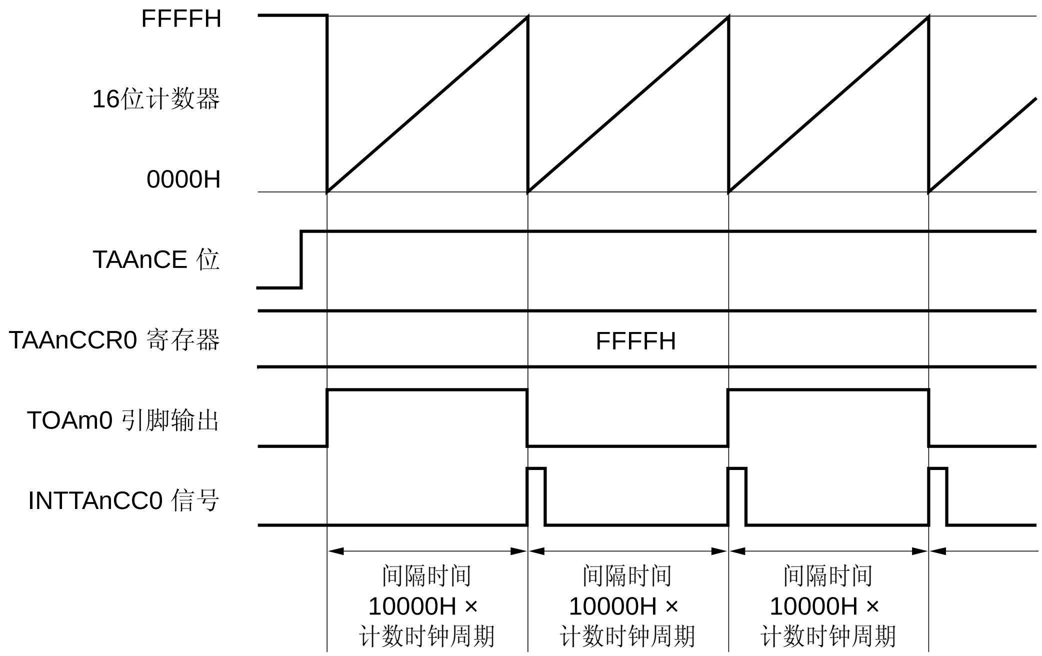
16 位计数器的值总是 0000H。



备注 V850E/IF3：n = 0 至 4，m = 2，4
V850E/IG3：n = 0 至 4，m = 2 至 4

（b） 如果 TAAnCCR0 寄存器被设置为 FFFFH 时的操作

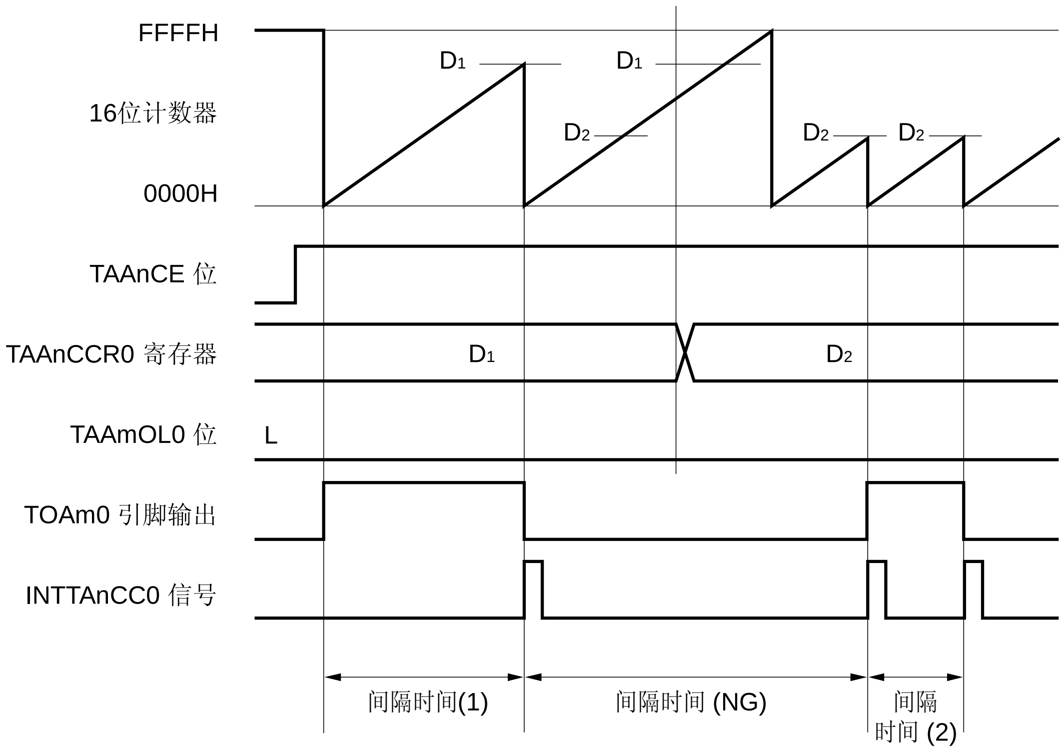
如果 TAAnCCR0 寄存器置为 FFFFH，那么 16 位计数器向上计数到 FFFFH。该计数器在下一个向上计数时刻同步清零，产生 INTTAnCC0 信号，并且 TOAm0 引脚输出被反转。此时，不产生溢出中断请求信号（INTTAnOV），而且，溢出标志（TAAnOPT0.TAAnOVF 位）也不会置 1。



备注 V850E/IF3：n = 0 至 4，m = 2，4
V850E/IG3：n = 0 至 4，m = 2 至 4

（c） 重写 TAAnCCR0 寄存器的注意事项

如果在计数器工作期间，将 TAAnCCR0 寄存器的值重写为比当前值较小的值，16 位计数器可能会溢出。如果发生溢出，停止计数，然后改变设置值。



备注 **1.** 间隔时间（1）： （D_1 + 1）× 计数时钟周期

间隔时间（NG）： （10000H + D_2 + 1）× 计数时钟周期

间隔时间（2）： （D_2 + 1）× 计数时钟周期

2. V850E/IF3：n = 0 至 4，m = 2，4

V850E/IG3：n = 0 至 4，m = 2 至 4

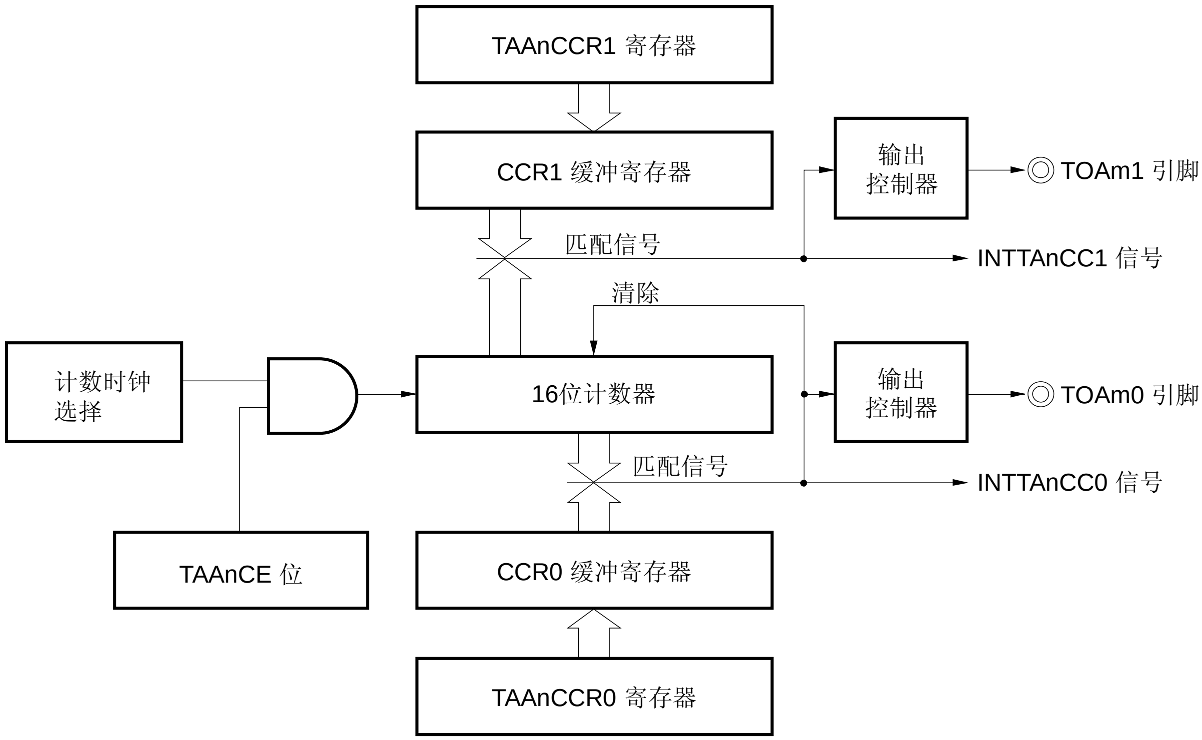
当计数值大于 D_2 而小于 D_1 时，如果将 TAAnCCR0 寄存器的值从 D_1 变为 D_2，在重写 TAAnCCR0 寄存器后，计数值会立即传输给 CCR0 缓冲寄存器。结果，用来和 16 位计数器进行比较的的值是 D_2。

但是，由于计数值已经超过了 D_2，所以 16 位计数器就会向上累积计数到 FFFFH，溢出，然后从 0000H 开始重新向上计数。当计数值和 D_2 一致时，产生 INTTAnCC0 信号，且 TOAm0 引脚的输出被反转。

因此，可能不会在原先预计的间隔时间“（D_1＋1）× 计数时钟周期” 或 “（D_2＋1）× 计数时钟周期”时产生 INTTAnCC0 信号，而会以“（10000H＋D_2＋1）× 计数时钟周期” 的间隔产生中断请求信号。

（d） **TAAnCCR1 寄存器的操作**

图 6-14. TAAnCCR1 寄存器的配置



备注　V850E/IF3：n = 0 至 4，m = 2，4
　　　V850E/IG3：n = 0 至 4，m = 2 至 4

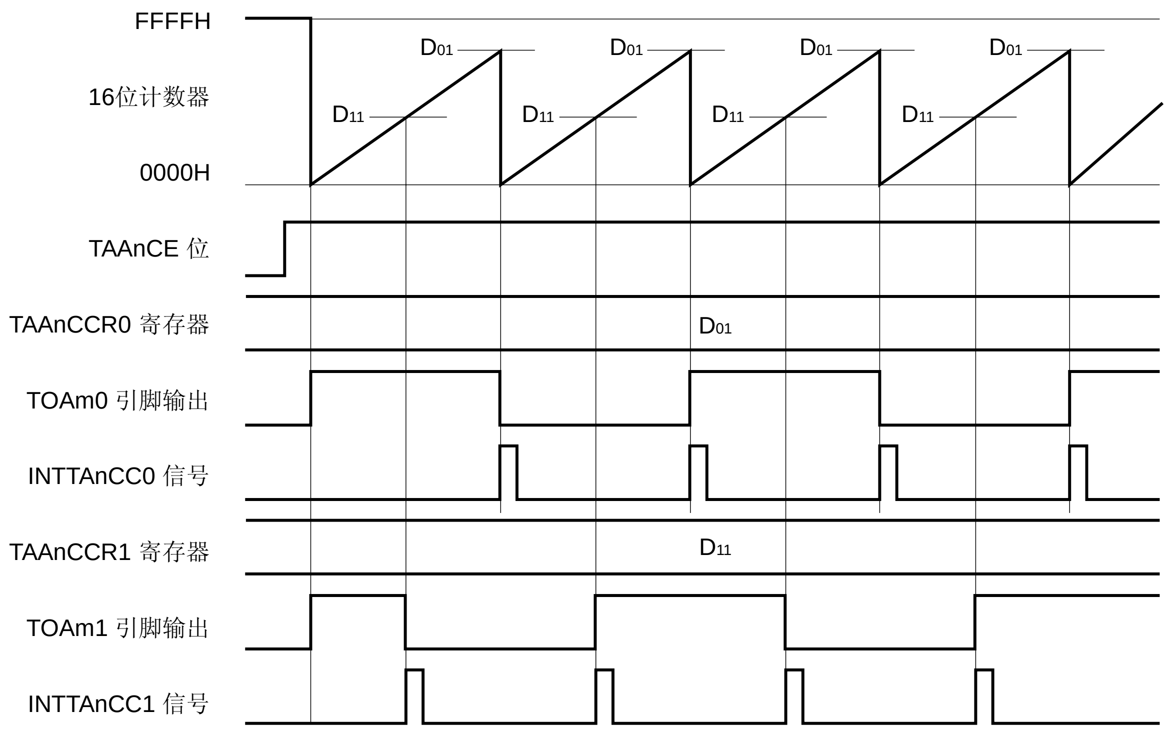
当 TAAnCCR1 寄存器的设置值和 TAAnCCR0 寄存器的值相等时，INTTAnCC0 信号的产生时序和 INTTAnCC1 信号相同，且 TOAm1 引脚输出被反转。也就是说，可以从 TOAm1 引脚输出一个占空比 50%的 PWM 波形。

下面展示了当 TAAnCCR1 寄存器的设置值和 TAAnCCR0 寄存器的值不同时的操作。

如果 TAAnCCR1 寄存器的设定值小于 TAAnCCR0 寄存器的设定值，则每个计数周期产生一次 INTTAnCC1 中断请求信号。同时，TOAm1 引脚的输出被反转。

TOAm1 引脚输出一个窄脉冲之后，输出一个占空比 50%的 PWM 波形。

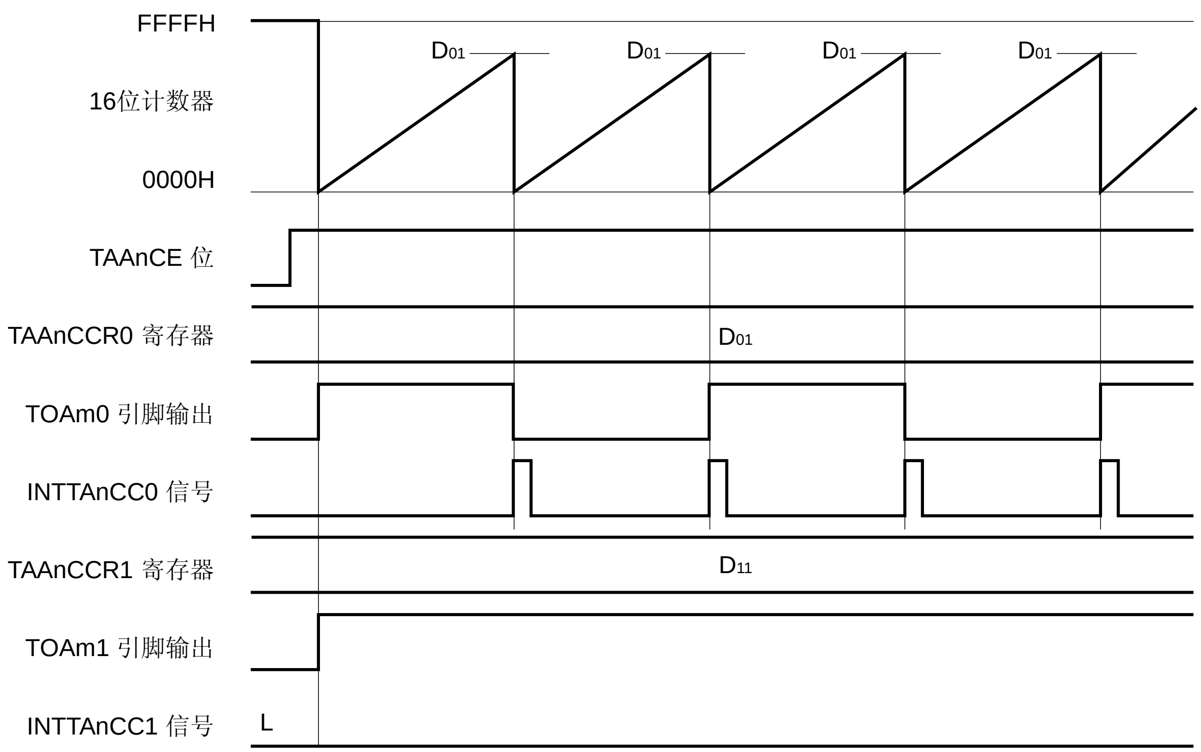
图 6-15. $D_{01} \geq D_{11}$ 的时序图



备注　V850E/IF3：n = 0 至 4，m = 2，4
　　　V850E/IG3：n = 0 至 4，m = 2 至 4

如果 TAAnCCR1 寄存器的设置值大于 TAAnCCR0 寄存器的值，16 位计数器的计数值不会和 TAAnCCR1 寄存器的值发生一致。于是，也就不会产生 INTTAnCC1 信号，TOAm1 引脚的输出也不会改变。
不使用 TAAnCCR1 寄存器时，推荐将其值设置为 FFFFH。

图 6-16. $D_{01} < D_{11}$ 的时序图



备注 V850E/IF3：n = 0 至 4，m = 2，4
V850E/IG3：n = 0 至 4，m = 2 至 4

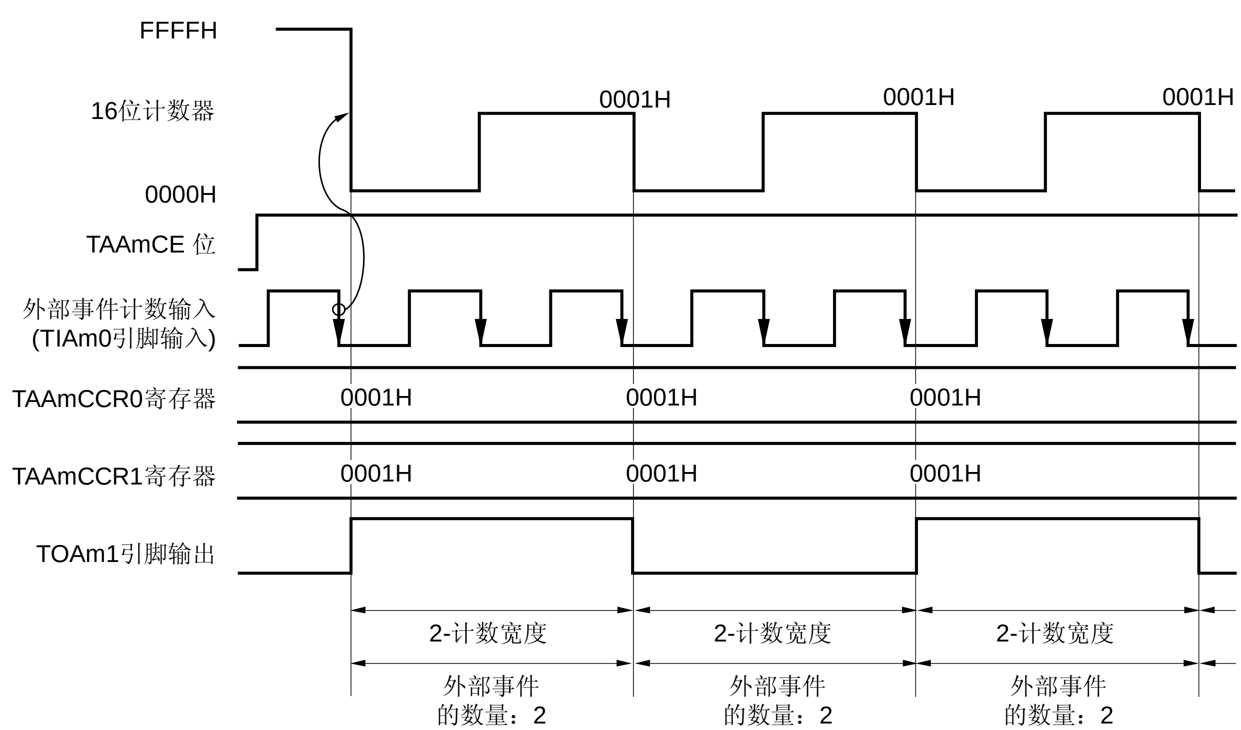
（3）外部事件计数输入（TIAm0）的操作

（a） 操作

在间隔定时器模式下，要在外部事件计数输入（TIAm0）的有效边沿计数 16 位计数器。在 TAAmCE 位从 0 设置为 1 之后，通过外部事件计数的有效边沿，16 位计数器从 FFFFH 清除为 0000H。

当 0001H 设置到 TAAmCCR0 和 TAAmCCR1 寄存器时，16 位计数器每计数两次，TOAm1 引脚输出反转。

只有当定时器输出（TOAm1）和外部事件计数输入一起使用时，TAAmCTL1.TAAmEEE 位可以在间隔定时器模式下设置为 1。



备注 V850E/IF3：m = 2，4
V850E/IG3：m = 2 至 4

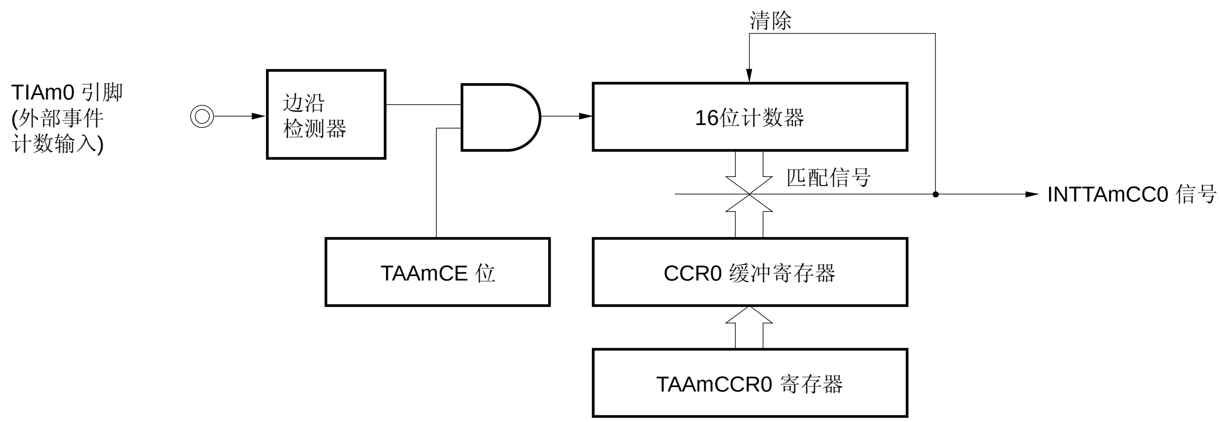
6.6.2 外部事件计数模式（TAAmMD2 至 TAAmMD0 位 = 001）

只有在 TAA2，TAA3（仅限 V850E/IG3）和 TAA4 时，该模式有效。

在外部事件计数模式下，当 TAAmCTL0.TAAmCE 位设置为 1 时，外部事件计数输入（TIAm0）的有效边沿计数。每次计数达到 TAAmCCR0 寄存器设置的边沿数量时，产生一个中断请求信号（INTTAmCC0）。TOAm0 和 TOAm1 引脚不能使用。当 TOAm1 引脚用于配合外部事件计数输入时，在间隔定时器模式下将 TAAmCTL1.TAAmEEE 位设置为 1（参见 **6.6.1（3）外部事件计数输入（TIAm0）的操作**）。

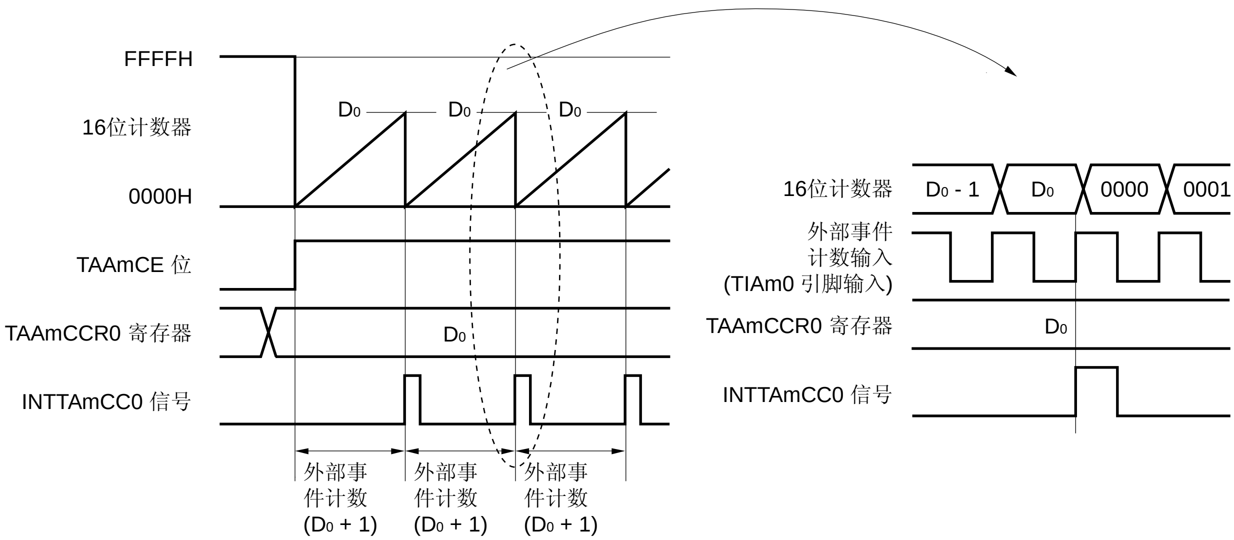
外部事件计数模式不使用 TAAmCCR1 寄存器。

图 6-17. 外部事件计数模式的配置



备注　V850E/IF3：m = 2，4
　　　V850E/IG3：m = 2 至 4

图 6-18. 外部事件计数模式的基本时序



备注　**1.** 该图显示当外部事件计数输入的有效边沿被指定为上升沿时的基本时序。

2. V850E/IF3：m = 2，4

V850E/IG3：m = 2 至 4

当 TAAmCE 位置为 1 时，16 位计数器的值从 FFFFH 清为 0000H。每次检测到外部事件计数输入的有效边沿时，计数器计数。此外，TAAmCCR0 寄存器的设定值传输到 CCR0 缓冲寄存器。

当 16 位计数器的计数值和 CCR0 缓冲寄存器的数值相一致时，16 位计数器清除为 0000H，同时生成一个一致的中断请求信号（INTTAmCC0）。

每当外部事件计数输入的有效边沿的检测数量达到"TAAmCCR0 寄存器设定值＋1"时，就会产生 INTTAmCC0 信号。

图 6-19. 外部事件计数模式操作的寄存器设置（1/2）

（a） TAAm 控制寄存器 0（TAAmCTL0）

	TAAmCE					TAAmCKS2	TAAmCKS1	TAAmCKS0
TAAmCTL0	0/1	0	0	0	0	0	0	0

TAAmCE：
0: 停止计数
1: 使能计数

（b） TAAm 控制寄存器 1（TAAmCTL1）

	TAAaSYE	TAAmEST	TAAmEEE			TAAmMD2	TAAmMD1	TAAmMD0
TAAmCTL1	0	0	0	0	0	0	0	1

TAAmMD2 至 TAAmMD0：
0, 0, 1:
外部事件计数模式

（c） TAAm I/O 控制寄存器 2（TAAmIOC2）

					TAAmEES1	TAAmEES0	TAAmETS1	TAAmETS0
TAAmIOC2	0	0	0	0	0/1	0/1	0	0

TAAmEES1、TAAmEES0：
选择外部事件计数输入
(TIAm0 引脚)的有效边沿

（d） TAAm 计数器读取缓冲寄存器（TAAmCNT）

通过读取 TAAmCNT 寄存器，可以读得 16 位计数器的计数值。

（e） TAAm 捕获/比较寄存器 0（TAAmCCR0）

如果 TAAmCCR0 寄存器设置为 D_0，当外部事件的数量达到（D_0 + 1）时，清除计数，并产生一致的中断请求信号（INTTAmCC0）。

图 6-19. 外部事件计数模式操作的寄存器设置（2/2）

（f） TAAm 捕获/比较寄存器 1（TAAmCCR1）

外部事件计数模式不使用 TAAmCCR1 寄存器。但是，TAAmCCR1 寄存器的设置值传输到 CCR1 缓冲寄存器。当 16 位计数器的计数值和 CCR1 缓冲寄存器的值一致时，产生一个一致中断请求信号（INTTAAmCC1）。

不使用 TAAmCCR1 寄存器时，推荐将其值设置为 FFFFH。并且通过中断屏蔽标志（TAAmCCIC1.TAAmCCMK1）屏蔽该寄存器。

注意事项 **1. 将 TAAmIOC0 寄存器设置为 00H。**

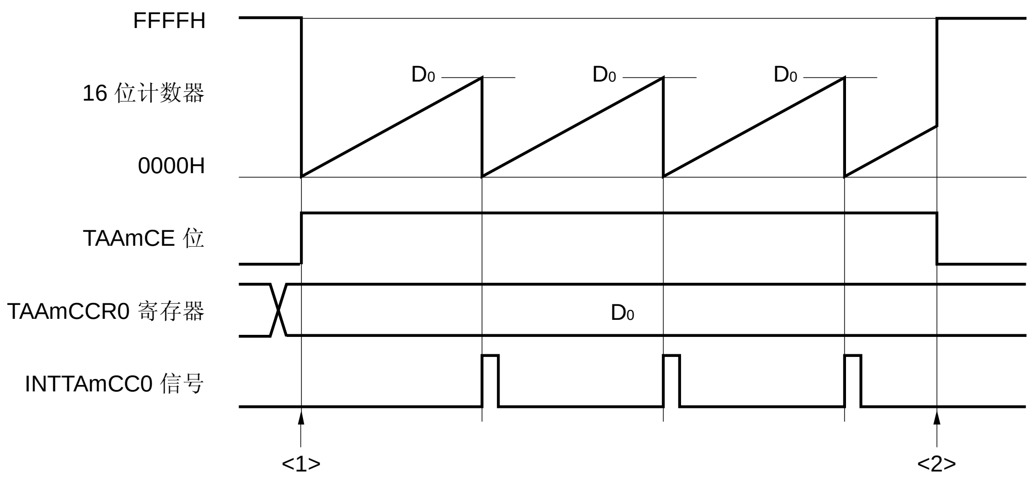
<R> **2. 当外部时钟用作计数时钟时，外部时钟只能从 TIAm0 引脚输入。此时，将 TAAmIOC1.TAAmIS1 和 TAAmIOC1.TAAmIS0 位设置为 00（捕获触发输入（TIAm0 引脚）：无边沿检测）。**

备注 **1.** 外部事件计数模式不使用 TAAm I/O 控制寄存器 1（TAAmIOC1）和 TAAm 选项寄存器 0（TAAmOPT0）。

2. V850E/IF3：m = 2，4，a = 0，1
V850E/IG3：m = 2 至 4，a = 0，1

（1） 外部事件计数模式操作流程

图 6-20. 外部事件计数模式下的软件处理流程



<1> 计数操作启动流程

启动

寄存器初始设置
TAAmCTL1 寄存器,
TAAmIOC2 寄存器,
TAAmCCR0,TAAmCCR1 寄存器

在设置TAAmCE位为1之前，执行这些寄存器的初始化设置。

TAAmCE 位 = 1

<2> 计数操作停止流程

TAAmCE 位 = 0

计数器被初始化，并且将TAAmCE 位清除为0，停止计数。

停止

备注 V850E/IF3：m = 2，4
V850E/IG3：m = 2 至 4

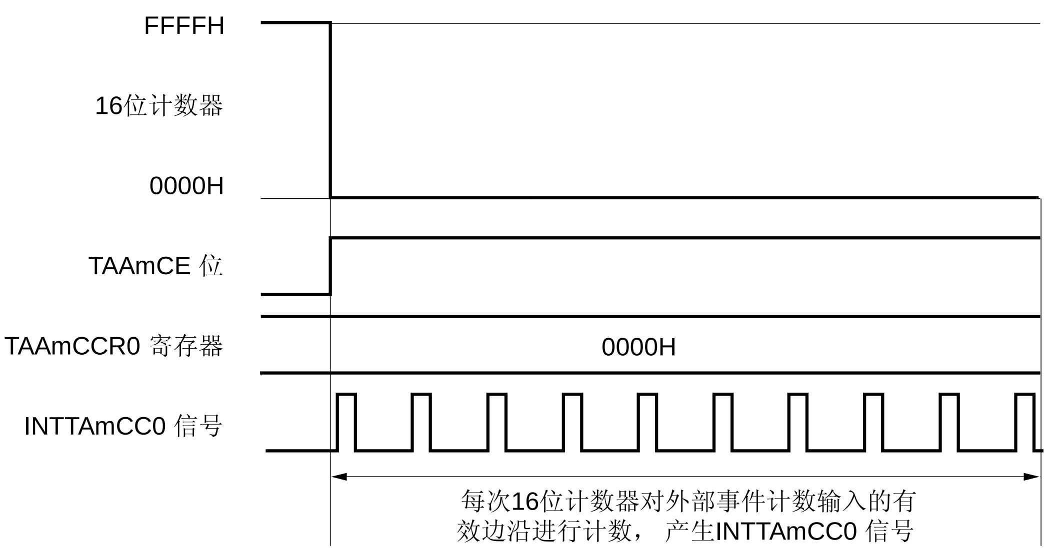
（2） 外部事件计数模式下的操作时序

注意事项 在外部事件计数模式下，禁止使用定时器输出（**TOAm0，TOAm1**）。如果定时器输出（**TOAm1**）和外部事件计数输入（**TIAm0**）一起使用，设置间隔定时器模式，并通过外部事件计数输入为计数时钟选择操作使能（**TAAmCTL1.TAAmEEE** 位 **= 1**）（参见 **6.6.1**（**3**）外部事件计数输入（**TIAm0**）的操作）。

（a） 如果 **TAAmCCR0** 寄存器被设置为 **0000H** 的操作

如果 TAAmCCR0 寄存器清为 0000H，每当外部事件计数信号的有效边沿的检测数量和 CCR0 缓冲寄存器的值一致时，16 位计数器重复清除为 0000H 并产生 INTTAmCC0 信号。

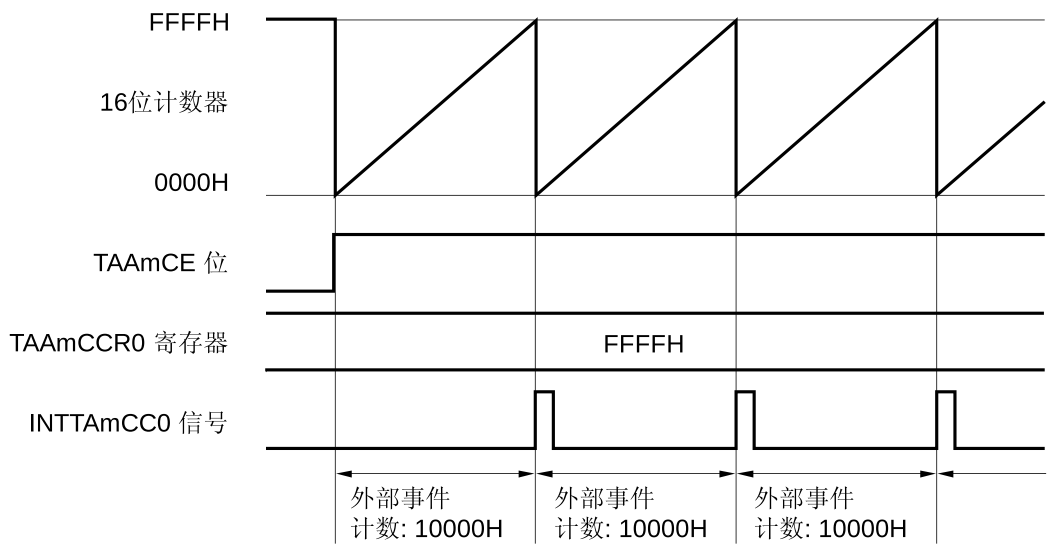
16 位计数器的值总是 0000H。



备注 V850E/IF3：m = 2，4
V850E/IG3：m = 2 至 4

（b） 如果 TAAmCCR0 寄存器被设置为 FFFFH 的操作

如果 TAAmCCR0 寄存器置为 FFFFH，每次检测到外部事件计数信号的有效边沿时，16 位计数器向上计数到 FFFFH。16 位计数器在下一个向上计数时刻同步清零，产生 INTTAmCC0 信号。此时，TAAmOPT0.TAAmOVF 位也不会置位。

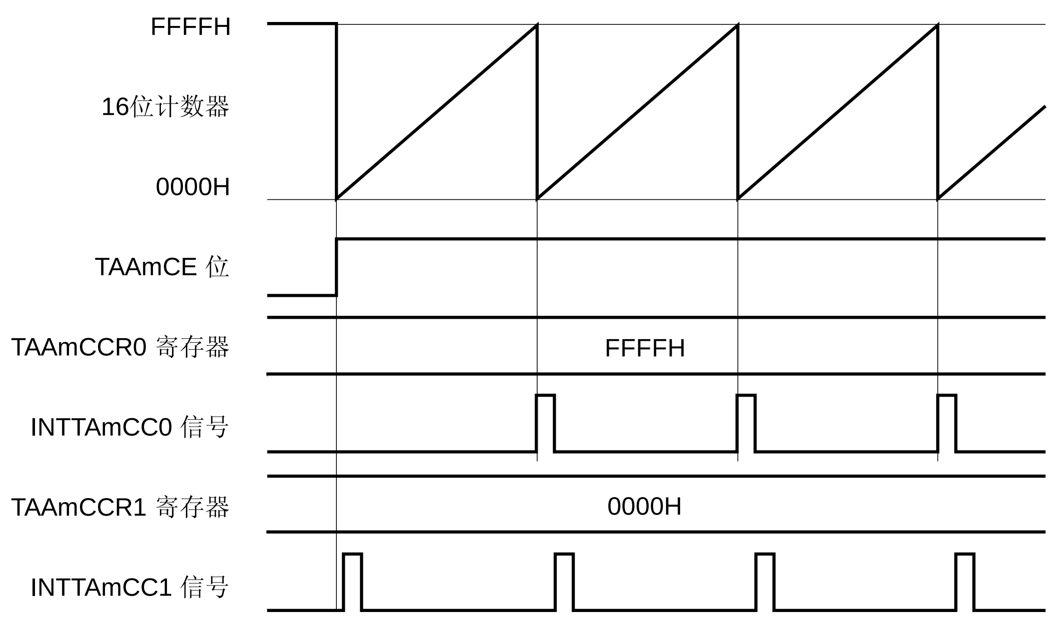


备注 V850E/IF3：m = 2，4
V850E/IG3：m = 2 至 4

（c） **TAAmCCR0 寄存器被设置为 FFFFH 且 TAAmCCR1 寄存器被设置为 0000H 的操作**

当 TAAmCCR0 寄存器设置为 FFFFH 时，每次检测到外部事件计数信号的有效边沿时，16 位计数器向上计数到 FFFFH。16 位计数器在下一个向上计数时刻同步清零，产生 INTTAmCC0 信号。此时，TAAmOPT0.TAAmOVF 位也不会置位。

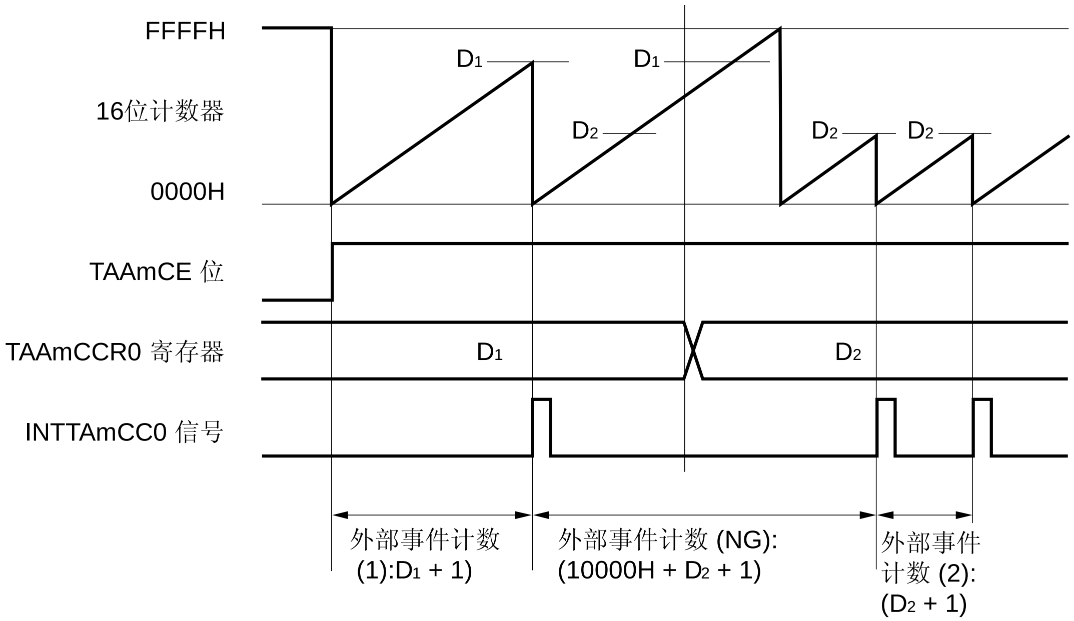
如果 TAAmCCR1 寄存器设置为 0000H，当 16 位计数器清除为 0000H 时，产生 INTTAmCC1 信号。



备注 V850E/IF3：m = 2，4
V850E/IG3：m = 2 至 4

（d） 重写 TAAmCCR0 寄存器的注意事项

如果在计数器工作期间，将 TAAmCCR0 寄存器的值重写为比当前值较小的值，16 位计数器可能会溢出。如果发生溢出时，停止计数，然后改变设置值。



备注 V850E/IF3：m = 2，4
V850E/IG3：m = 2 至 4

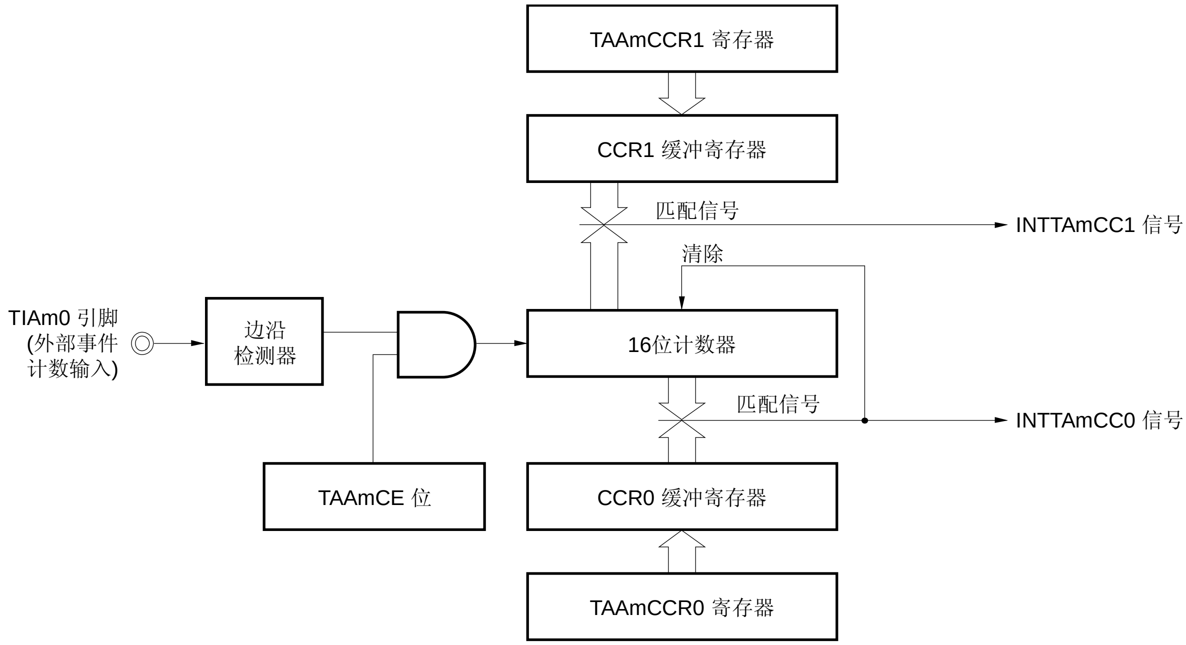
当计数值大于 D_2 而小于 D_1 时，如果将 TAAmCCR0 寄存器的值从 D_1 变为 D_2，在重写 TAAmCCR0 寄存器后，计数值会立即传输给 CCR0 缓冲寄存器。结果，用来和 16 位计数器进行比较的的值是 D_2。

但是，由于计数值已经超过了 D_2，所以 16 位计数器就会向上累积计数到 FFFFH，溢出，然后从 0000H 开始重新向上计数。当计数值和 D_2 一致时，产生 INTTAmCC0 信号。

因此，可能不会在原先预计的间隔时间“（D_1＋1）× 计数时钟周期” 或 “（D_2＋1）× 计数时钟周期”时产生 INTTAmCC0 信号，而会以“（10000H＋D_2＋1）× 计数时钟周期” 的间隔产生中断请求信号。

（e） **TAAmCCR1 寄存器的操作**

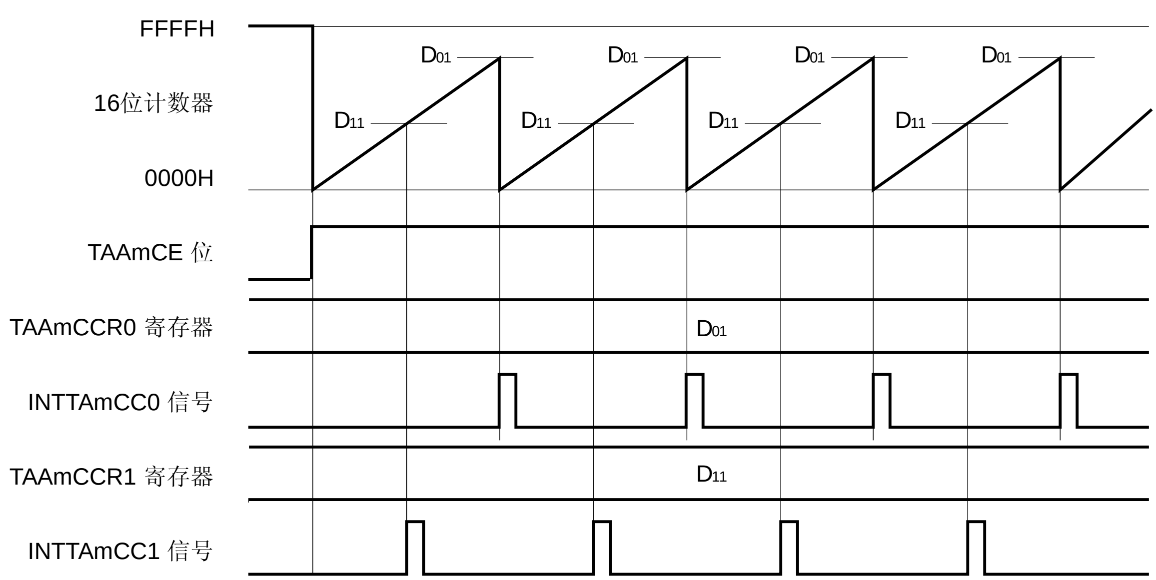
图 6-21. TAAmCCR1 寄存器的配置



备注 V850E/IF3：m = 2，4
V850E/IG3：m = 2 至 4

如果 TAAmCCR1 寄存器的设置值小于 TAAmCCR0 寄存器的设置值，每个周期产生一次 INTTAmCC1 信号。

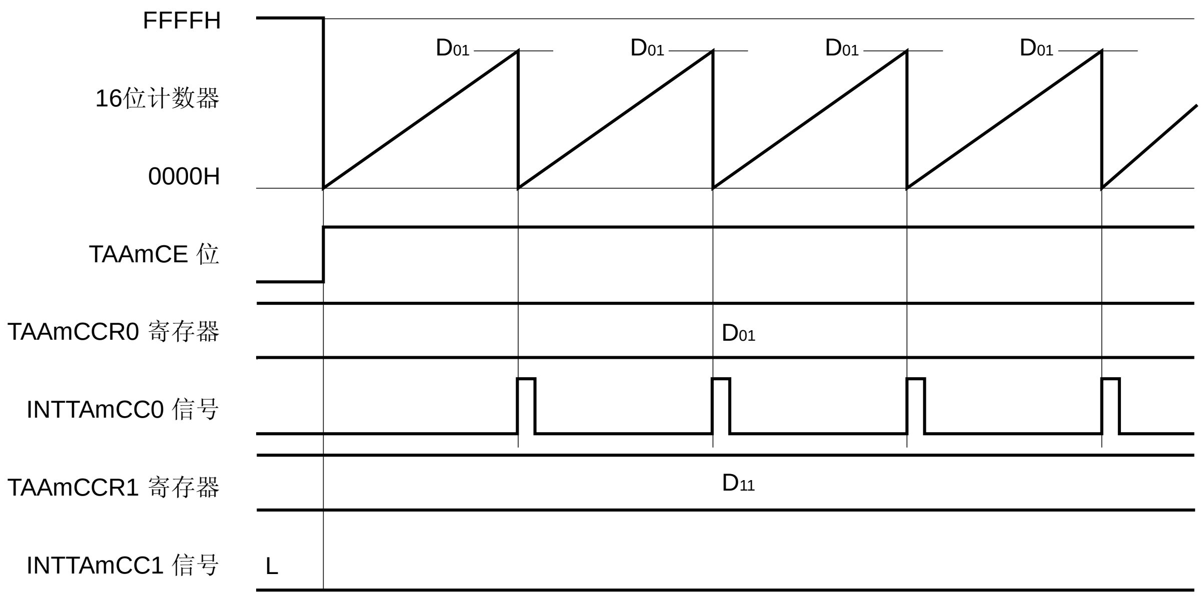
图 6-22. $D_{01} \geq D_{11}$ 的时序图



备注 V850E/IF3：m = 2，4
V850E/IG3：m = 2 至 4

如果 TAAmCCR1 寄存器的设置值大于 TAAmCCR0 寄存器的设置值，不产生 INTTAmCC1 信号，因为 16 位计数器的计数值和 TAAmCCR1 寄存器的值不一致。
不使用 TAAmCCR1 寄存器时，推荐将其值设置为 FFFFH。

图 6-23. D_{01} < D_{11} 的时序图



备注　V850E/IF3：m = 2，4
　　　V850E/IG3：m = 2 至 4

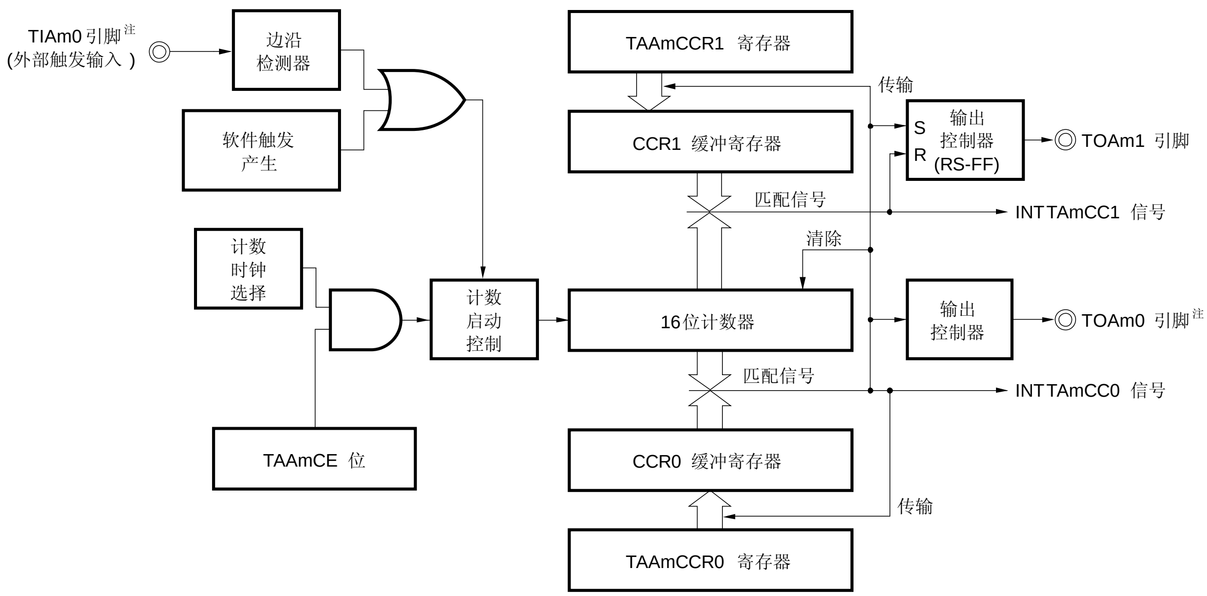
6.6.3 外部触发脉冲输出模式（TAAmMD2 至 TAAmMD0 位 = 010）

只有在 TAA2，TAA3（仅限 V850E/IG3）和 TAA4 时，该模式有效。

在外部触发脉冲输出模式下，当 TAAmCTL0.TAAmCE 位设置为 1 时，16 位定时器/事件计数器 AA 等待触发。检测到外部触发输入（TIAm0）的有效边沿时，16 位定时器/事件计数器 AA 开始计数，并从 TOAm1 引脚输出一个 PWM 波形。

通过产生一个软件触发，而不是使用外部触发，也可以输出脉冲。当使用软件触发时，可以从 TOAm0 引脚输出一个占空比 50%的 PWM 波形，其半周期等于 TAAmCCR0 寄存器的设置值 + 1。

<R> 图 6-24. 外部触发脉冲输出模式的配置

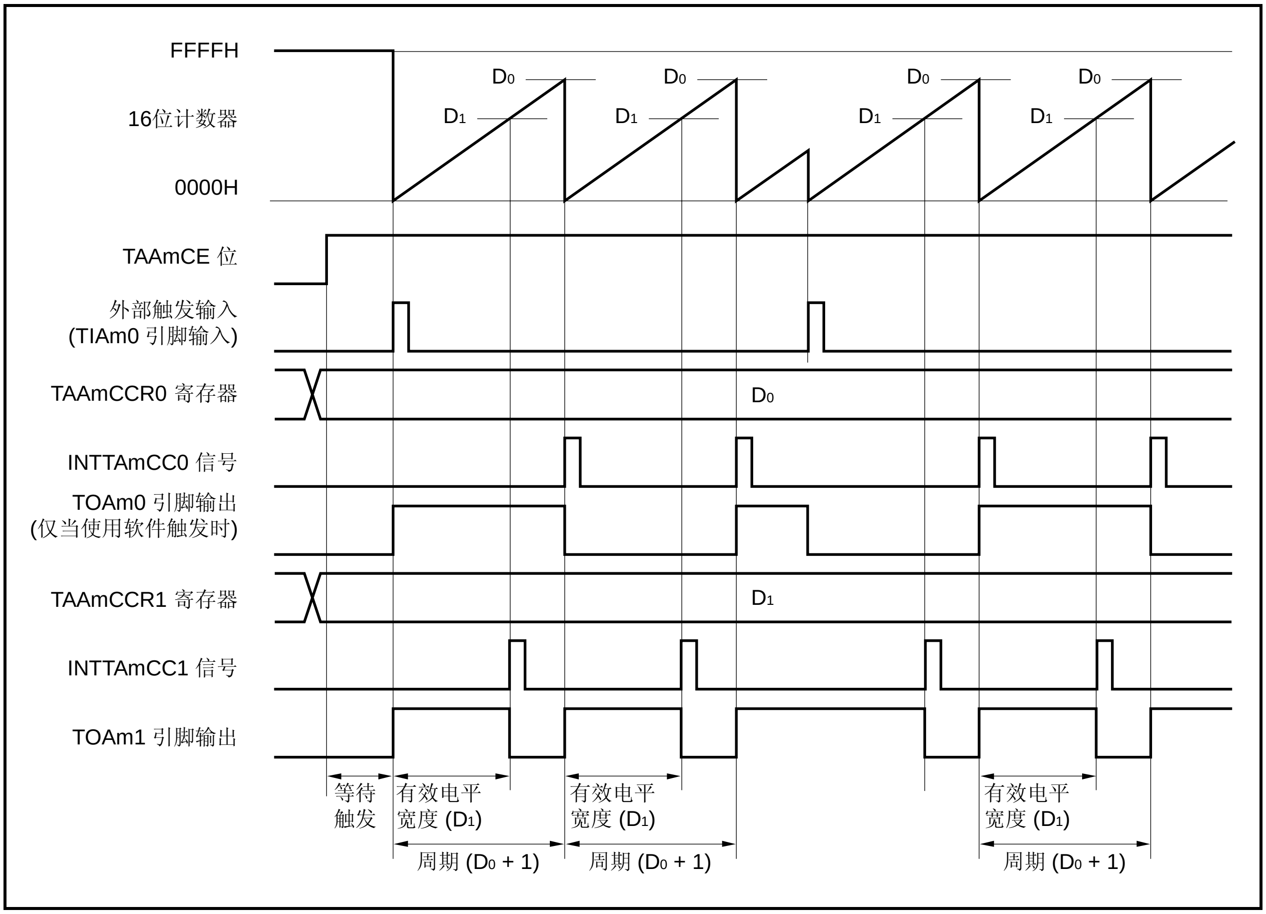


注　因为外部触发输入引脚（TIAm0）和定时器输出引脚（TOAm0）共用相同的复用功能引脚，这两个功能不能同时使用。

注意事项　在外部触发脉冲输出模式下，选择内部时钟作为计数时钟（将 TAAmCTL1.TAAmEEE 位清除为 0）。

备注　V850E/IF3：m = 2，4
V850E/IG3：m = 2 至 4

图 6-25. 外部触发脉冲输出模式的基本时序



当 TAAmCE 位设为 1 时，16 位定时器/事件计数器 AA 等待触发。当触发产生时，16 位计数器从 FFFFH 清除为 0000H，同时开始计数，并从 TOAm1 引脚输出一个 PWM 波形。如果在计数运行中再次产生触发，则计数器清为 0000H 并重新开始计数。（TOAm0 引脚输出被反转。在发生触发时，不管当前状态如何（高/低），TOAm1 引脚都会输出一个高电平。

PWM 波形的有效电平宽度、周期、占空比的计算如下。

有效电平宽度 =（TAAmCCR1 寄存器的设置值）× 计数时钟周期

周期 =（TAAmCCR0 寄存器的设置值＋1）× 计数时钟周期

占空比系数 =（TAAmCCR1 寄存器的设置值）/（TAAmCCR0 寄存器的设置值＋1）

当 16 位计数器在其计数值与 CCR0 缓冲寄存器的值一致之后，进行下一次计数时就会同步产生比较一致中断请求信号 INTTAmCC0，且 16 位计数器同时清除为 0000H。当 16 位计数器的计数值与 CCR1 缓冲寄存器的值一致时，会生成一致中断请求信号 INTTAmCC1。

当 16 位计数器的计数值与 CCRa 缓冲寄存器的值一致，且 16 位计数器清除为 0000H 时，TAAmCCRa 寄存器的设定值就传输到 CCRa 缓冲寄存器。

外部触发输入信号的有效边沿，或将软件触发（TAAmCTL1.TAAmEST 位）设置为 1，都可以用作触发信号。

备注　V850E/IF3：m = 2，4，a = 0，1，V850E/IG3：m = 2 至 4，a = 0，1

图 **6-26.** 外部触发脉冲输出模式的寄存器设置（**1/2**）

（**a**） **TAAm** 控制寄存器 **0**（**TAAmCTL0**）

<R>

	TAAmCE					TAAmCKS2	TAAmCKS1	TAAmCKS0
TAAmCTL0	0/1	0	0	0	0	0/1	0/1	0/1

TAAmCKS2 至 TAAmCKS0：选择计数时钟

TAAmCE：
0: 停止计数
1: 使能计数

（**b**） **TAAm** 控制寄存器 **1**（**TAAmCTL1**）

<R>

	TAAaSYE	TAAmEST	TAAmEEE			TAAmMD2	TAAmMD1	TAAmMD0
TAAmCTL1	0	0/1	0	0	0	0	1	0

TAAmMD2 至 TAAmMD0：
0, 1, 0:
外部触发脉冲输出模式

TAAmEEE：
0: 计数时钟由 TAAmCKS0
至 TAAmCKS2 位选择

TAAmEST：当写入 1 时，产生软件触发

（**c**） **TAAm I/O** 控制寄存器 **0**（**TAAmIOC0**）

					TAAmOL1	TAAmOE1	TAAmOL0	TAAmOE0
TAAmIOC0	0	0	0	0	0/1	0/1	0/1	0/1注

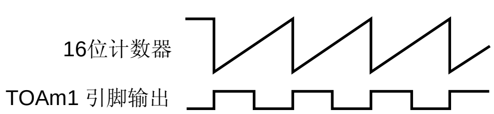
TAAmOE0：
0: 禁止 TOAm0 引脚输出
1: 使能 TOAm0 引脚输出

TAAmOL0：
在等待外部触发时，
TOAm0 引脚输出电平的设置
0: 低电平
1: 高电平

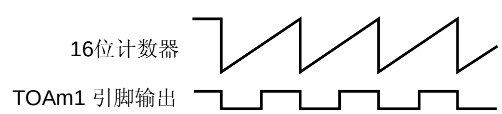
TAAmOE1：
0: 禁止 TOAm1 引脚输出
1: 使能 TOAm1 引脚输出

TAAmOL1：
在等待外部触发时，
TOAm1 引脚输出电平的设置
0: 低电平
1: 高电平

• 当 TAAmOL1 位 = 0



• 当 TAAmOL1 位 = 1



注　外部触发脉冲输出模式不使用 TOAm0 引脚时，该位清除为 0。

图 6-26. 外部触发脉冲输出模式的寄存器设置（2/2）

（d） TAAm I/O 控制寄存器 2（TAAmIOC2）

<R>

							TAAmEES1	TAAmEES0	TAAmETS1	TAAmETS0
TAAmIOC2	0	0	0	0	0	0	0/1	0/1		

选择外部触发输入
(TIAm0 引脚)的有效边沿

（e） TAAm 计数器读取缓冲寄存器（TAAmCNT）

通过读取 TAAmCNT 寄存器，可以读出 16 位计数器的值。

（f） TAAm 捕获/比较寄存器 0 和 1（TAAmCCR0 和 TAAmCCR1）

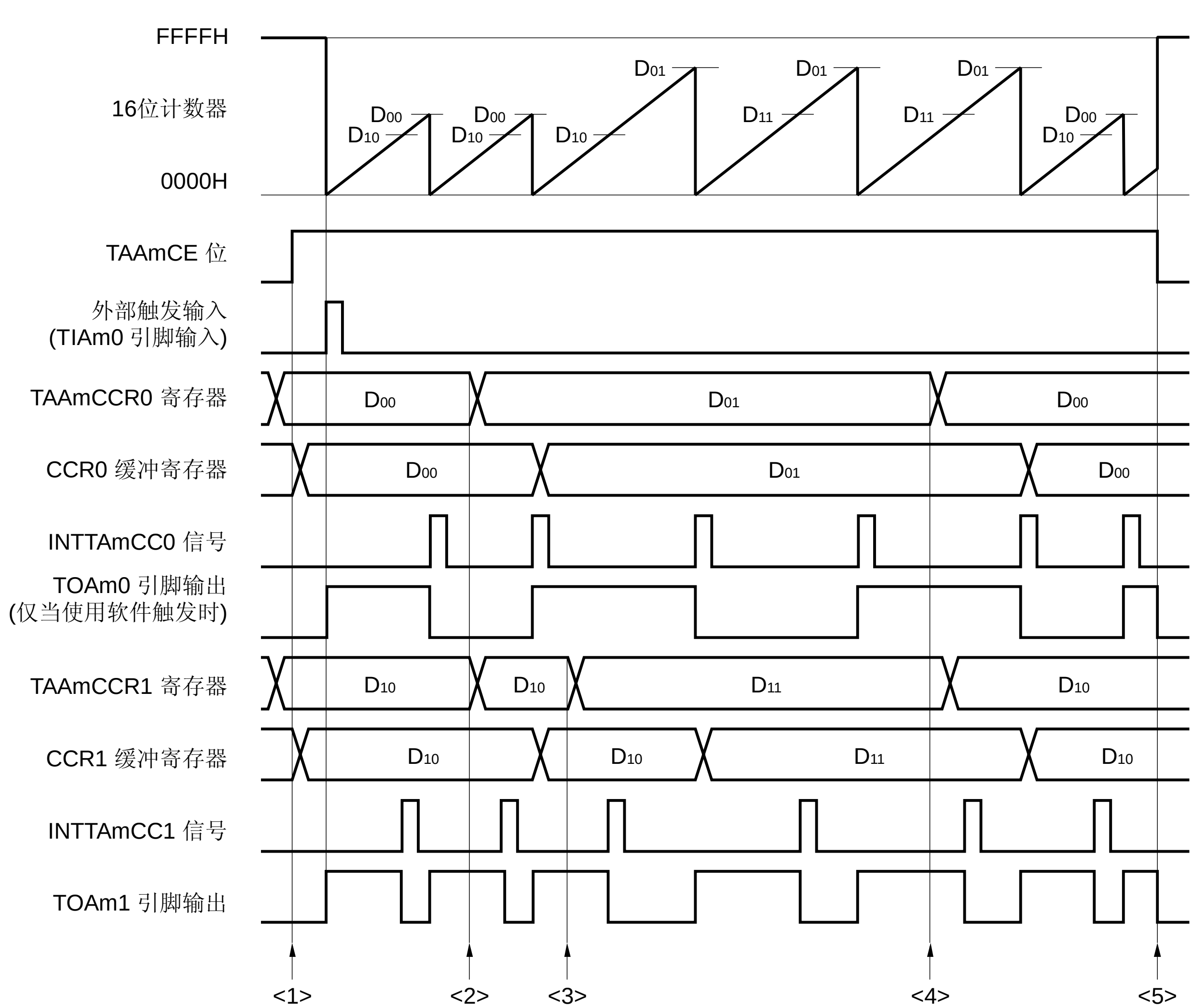
如果将 TAAmCCR0 寄存器的值设置为 D_0，将 TAAmCCR1 寄存器的值设置为 D_1，则 PWM 波形的周期和有效电平的计算如下：

周期 =（D_0＋1）× 计数时钟周期
有效电平宽度 = D_1 × 计数时钟周期

备注 **1.** 外部触发脉冲输出模式不使用 TAAm I/O 控制寄存器 1（TAAmIOC1）和 TAAm 选项寄存器 0（TAAmOPT0）。
2. V850E/IF3：m = 2，4
V850E/IG3：m = 2 至 4

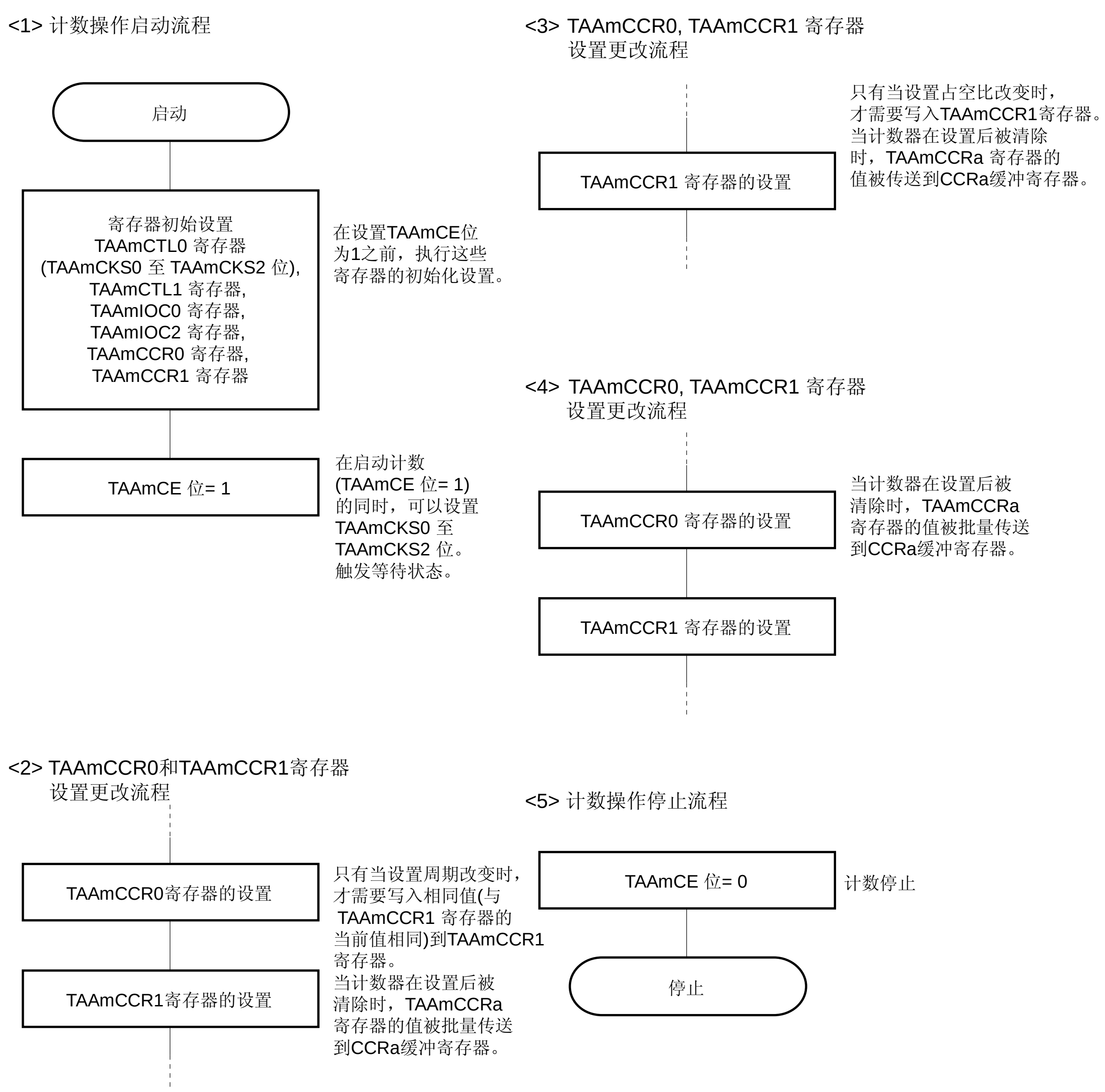
（1）外部触发脉冲输出模式的操作流程

图 6-27. 外部触发脉冲输出模式的软件处理流程（1/2）



备注　V850E/IF3：m = 2，4
V850E/IG3：m = 2 至 4

图 6-27. 外部触发脉冲输出模式的软件处理流程（2/2）



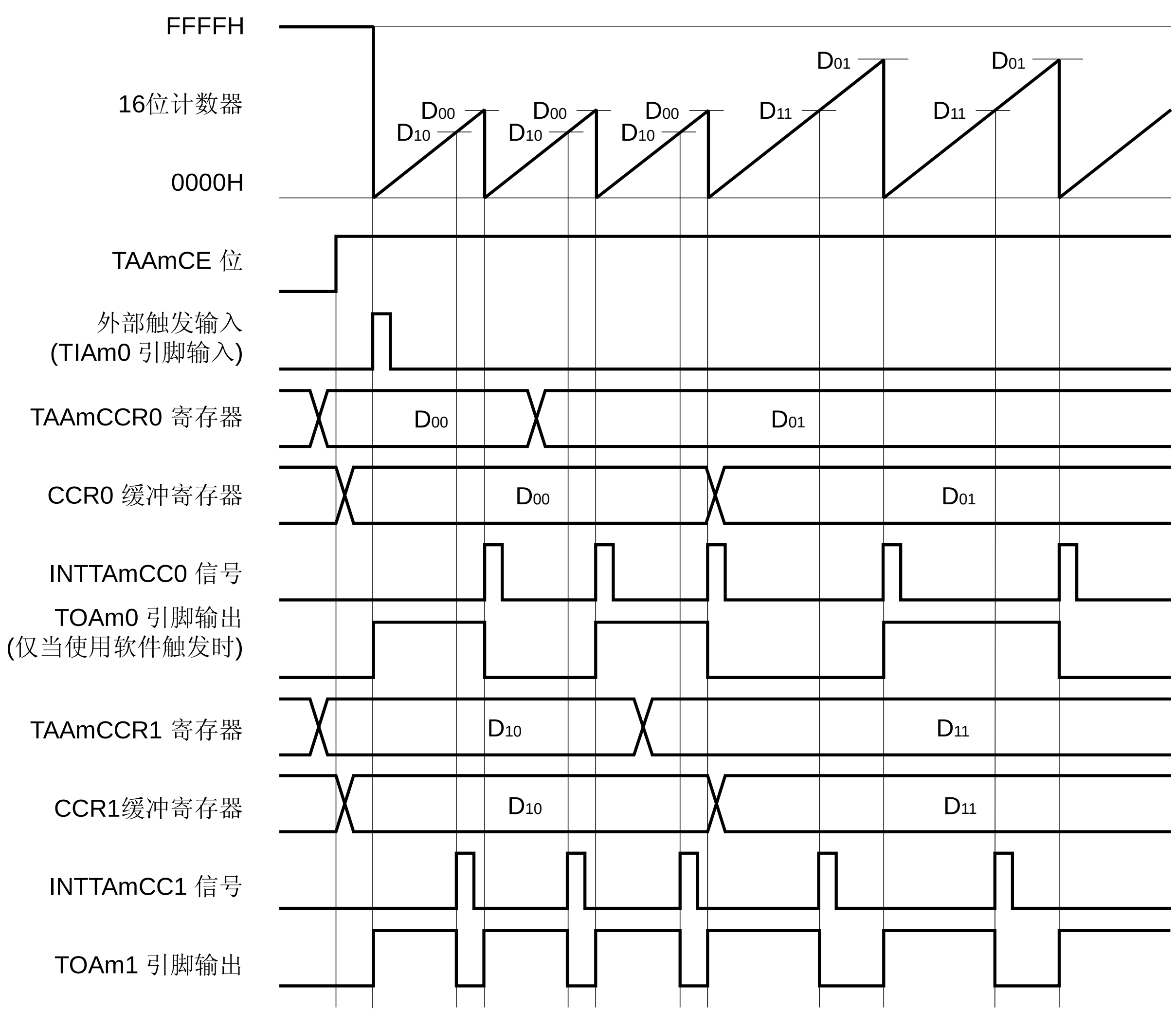
备注 V850E/IF3：m = 2，4，a = 0，1
V850E/IG3：m = 2 至 4，a = 0，1

（2）外部触发脉冲输出模式操作时序

（a） 操作期间改变脉冲宽度的注意事项

计数器计数过程中，如果要改变 PWM 波形，应最后写入 TAAmCCR1 寄存器。

当检测到 INTTAmCC0 信号以后，在 TAAmCCR1 寄存器写入操作之后重写 TAAmCCRa 寄存器。



备注　V850E/IF3：m = 2，4

V850E/IG3：m = 2 至 4

为了将数据从 TAAmCCRa 寄存器传输到 CCRa 缓冲寄存器，必须写入 TAAmCCR1 寄存器。

此时如果 PWM 波形的周期和有效电平宽度都要改变，应先将周期设定到 TAAmCCR0 寄存器，然后将有效电平宽度设定到 TAAmCCR1 寄存器。

如果只改变 PWM 波形的周期，应先把周期设定给 TAAmCCR0 寄存器，然后将相同值（和 TAAmCCR1 寄存器的当前值相等）写入 TAAmCCR1 寄存器。

只改变 PWM 波形的有效电平宽度（占空比）时，仅需要设置 TAAmCCR1 寄存器。

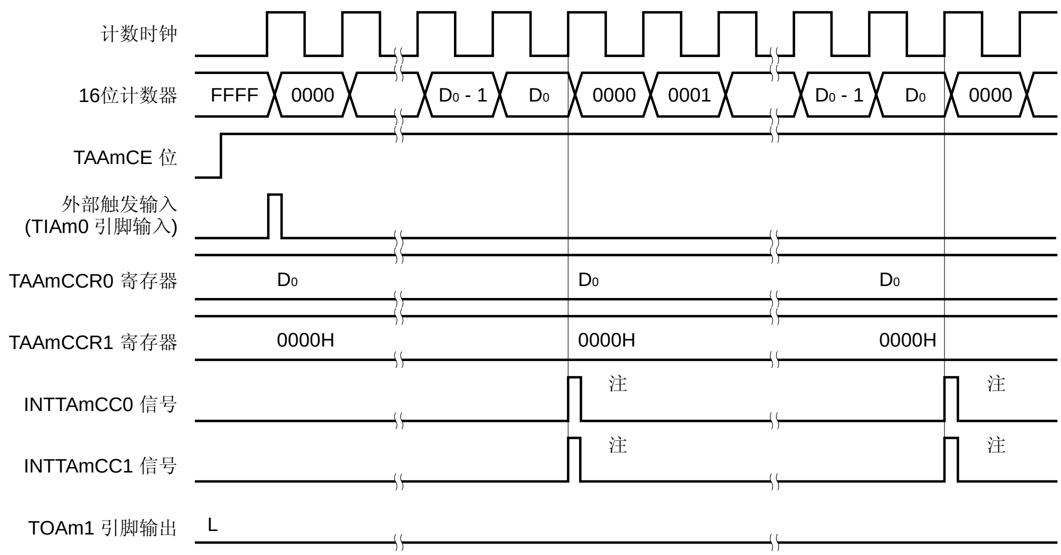
当数据写入 TAAmCCR1 寄存器之后，在清除 16 位计数器的同时，写入到 TAAmCCRa 寄存器的值传输到 CCRm 缓冲寄存器，并作为与 16 位计数器进行比较的值。

在写入 TAAmCCR1 寄存器一次之后，要再次写 TAAmCCR0 或 TAAmCCR1 寄存器，应该在 INTTAmCC0 信号产生之后进行。否则，CCRa 缓冲寄存器的值可能不确定，因为把数据从 TAAmCCRa 寄存器传输到 CCRa 缓冲寄存器的时序会和写入 TAAmCCRa 寄存器的时序发生冲突。

备注 V850E/IF3：m = 2，4，a = 0，1
V850E/IG3：m = 2 至 4，a = 0，1

（b） PWM 波形的 0%/100%输出

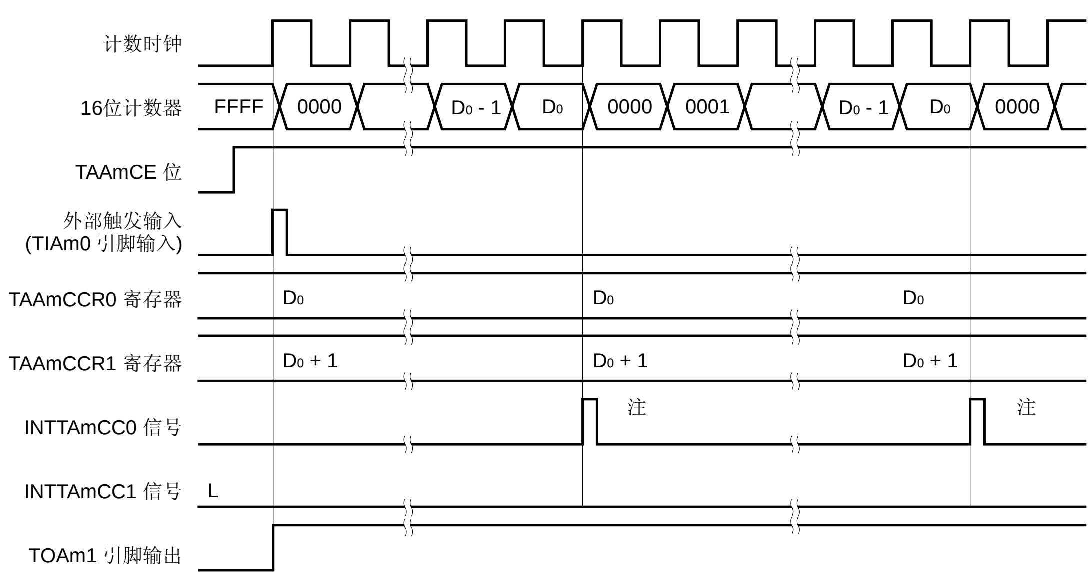
要输出一个 0% 波形，将 TAAmCCR1 寄存器设置为 0000H。在 16 位计数器的计数值和 CCR0 缓冲寄存器的值一致之后，16 位计数器清除为 0000H，且在下一个时序产生 INTTAmCC0 和 INTTAmCC1 信号。



注 时序实际延迟一个操作时钟（fxx）

备注 V850E/IF3：m = 2，4
V850E/IG3：m = 2 至 4

要输出 100%波形，将 TAAmCCR1 寄存器的值设置为（TAAmCCR0 寄存器的设置值 + 1）。如果 TAAmCCR0 寄存器的设置值是 FFFFH，不能生成 100%输出。

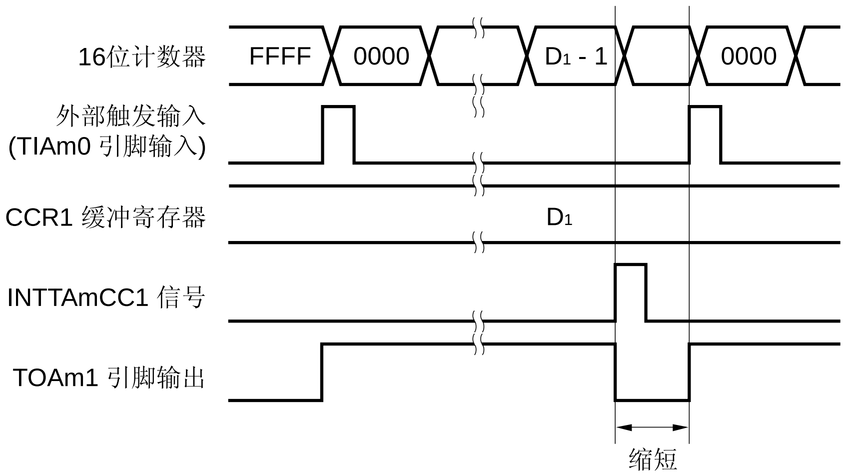


注 时序实际延迟一个操作时钟（fxx）

备注 V850E/IF3：m = 2，4
V850E/IG3：m = 2 至 4

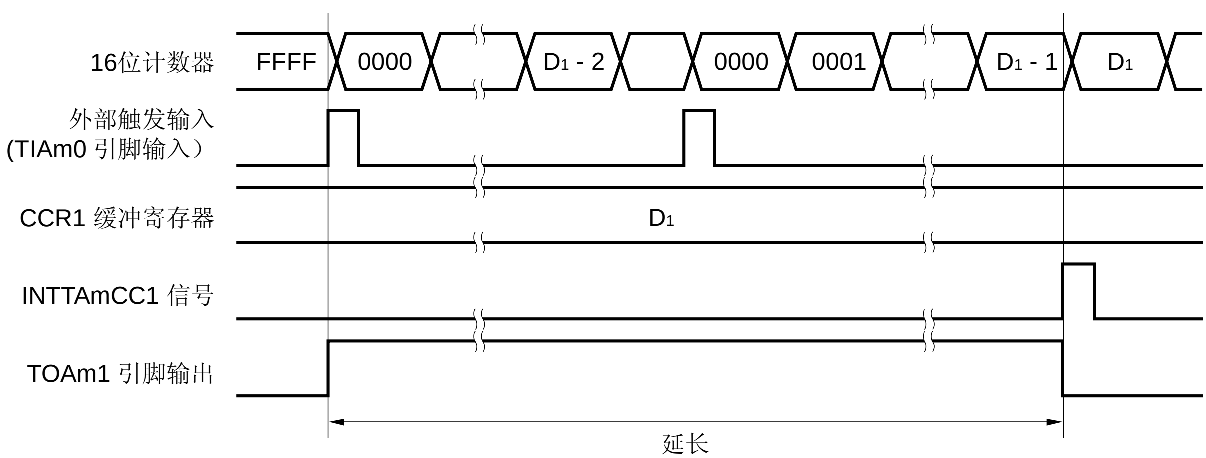
（c） 触发检测与 CCR1 缓冲寄存器一致之间的冲突

如果在 INTTAmCC1 信号产生之后，立刻检测到触发信号，则 16 位计数器马上清除为 0000H，TOAm1 引脚的输出信号生效，计数器继续计数。结果，PWM 波形的无效时段被缩短。



备注 V850E/IF3：m = 2，4
V850E/IG3：m = 2 至 4

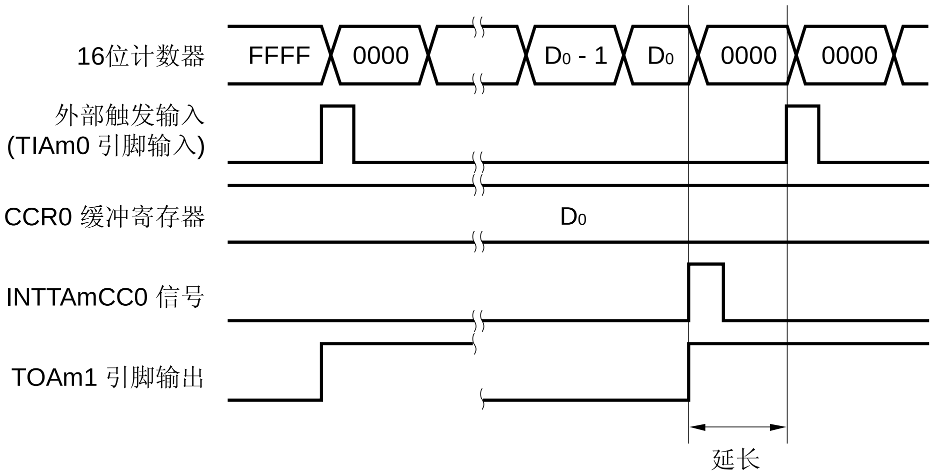
如果紧邻在 INTTAmCC1 信号产生之前检测到触发信号，则不会产生 INTTAmCC1 信号，而 16 位计数器清除为 0000H，计数继续进行。TOAm1 引脚的输出信号仍然保持有效。结果，PWM 波形的有效时段被延长。



备注 V850E/IF3：m = 2，4
V850E/IG3：m = 2 至 4

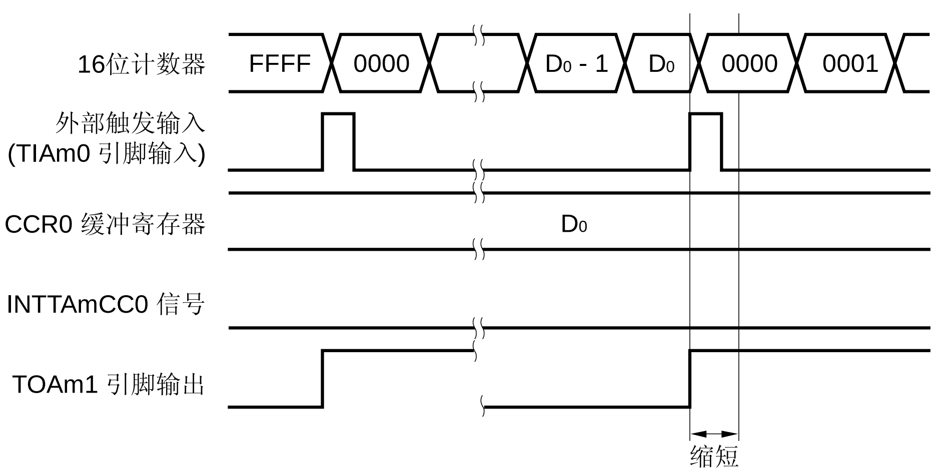
（d） 触发检测与 CCR0 缓冲寄存器一致之间的冲突

如果在 INTTAmCC0 信号产生后，立刻检测到触发信号，则 16 位计数器再次清除为 0000H，并继续向上计数。因此，TOAm1 引脚的有效时段被延长，延长长度为 INTTAmCC0 信号产生时刻至检测到触发信号时刻的时间间隔。



备注 V850E/IF3：m = 2，4
V850E/IG3：m = 2 至 4

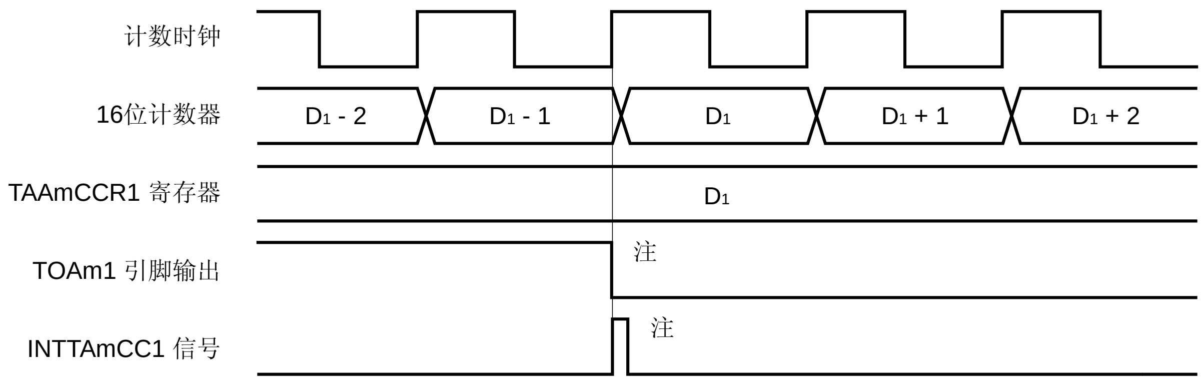
如果紧邻在 INTTAmCC0 信号产生之前检测到触发信号，则不会产生 INTTAmCC0 信号。16 位计数器清除为 0000H，TOAm1 引脚输出有效，计数器继续计数。结果，PWM 波形的无效时段被缩短。



备注 V850E/IF3：m = 2，4
V850E/IG3：m = 2 至 4

（e） 一致中断请求信号（INTTAmCC1）的产生时序

在外部触发脉冲输出模式下，INTTAmCC1 信号的产生时序与其它情况下 INTTAmCC1 信号的产生时序不同；当 16 位计数器的计数值与 TAAmCCR1 寄存器的值一致时，产生 INTTAmCC1 中断请求信号。



注 时序实际延迟一个操作时钟（fxx）

备注 V850E/IF3：m = 2，4
V850E/IG3：m = 2 至 4

通常在 16 位计数器的计数值与 TAAmCCR1 寄存器的值一致之后，在下一次计数时同步发生 INTTAmCC1 信号。
但是，在外部触发脉冲输出模式下，INTTAmCC1 信号会提早一个计数时钟周期产生。该时序的变化是为了一致 TOAm1 引脚输出信号变化的时序。

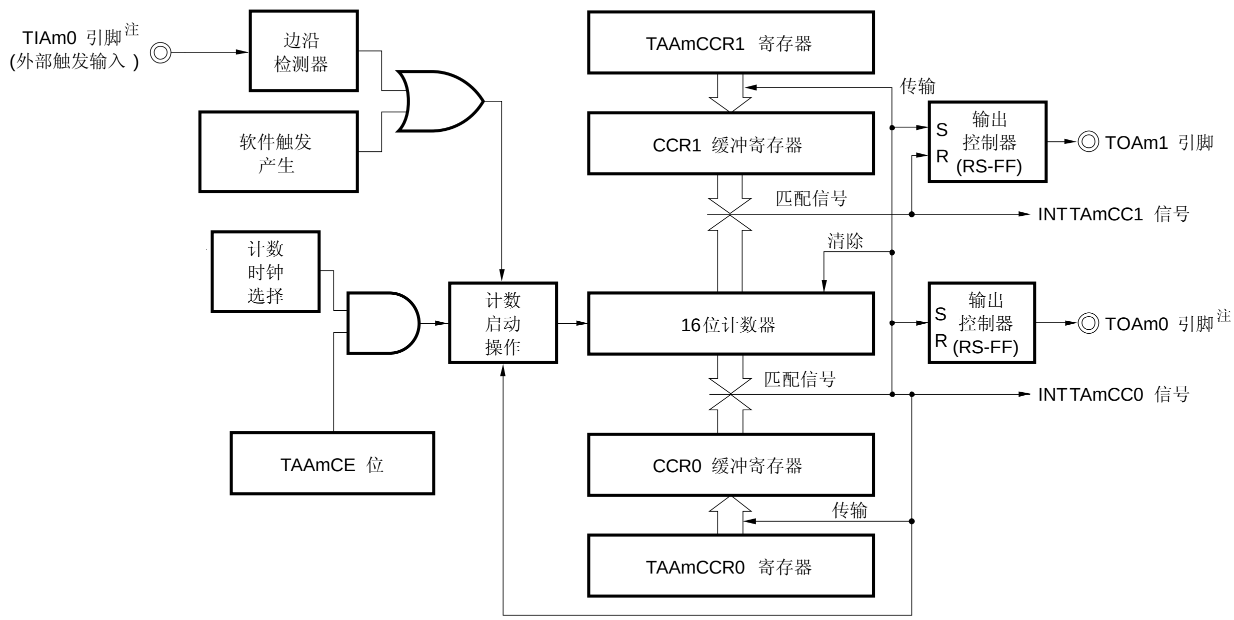
6.6.4 单次触发脉冲输出模式（TAAmMD2 至 TAAmMD0 位 = 011）

只有在 TAA2, TAA3（仅限 V850E/IG3）和 TAA4 时，该模式有效。

在单次触发脉冲输出模式下，当 TAAmCTL0.TAAmCE 位置为 1 时，16 位定时器/事件计数器 AA 等待触发。在检测到外部触发输入的有效边沿时，16 位定时器/事件计数器 AA 开始计数，同时从 TOAm1 引脚输出一个单次触发脉冲。

除外部触发输入（TIAm0）外，也可以产生软件触发信号来输出该脉冲。在使用软件触发时，TOAm0 引脚在 16 位计数器计数时输出有效电平，在计数器停止（等待触发）时输出无效电平。

<R> **图 6-28. 单次触发脉冲输出模式的配置**

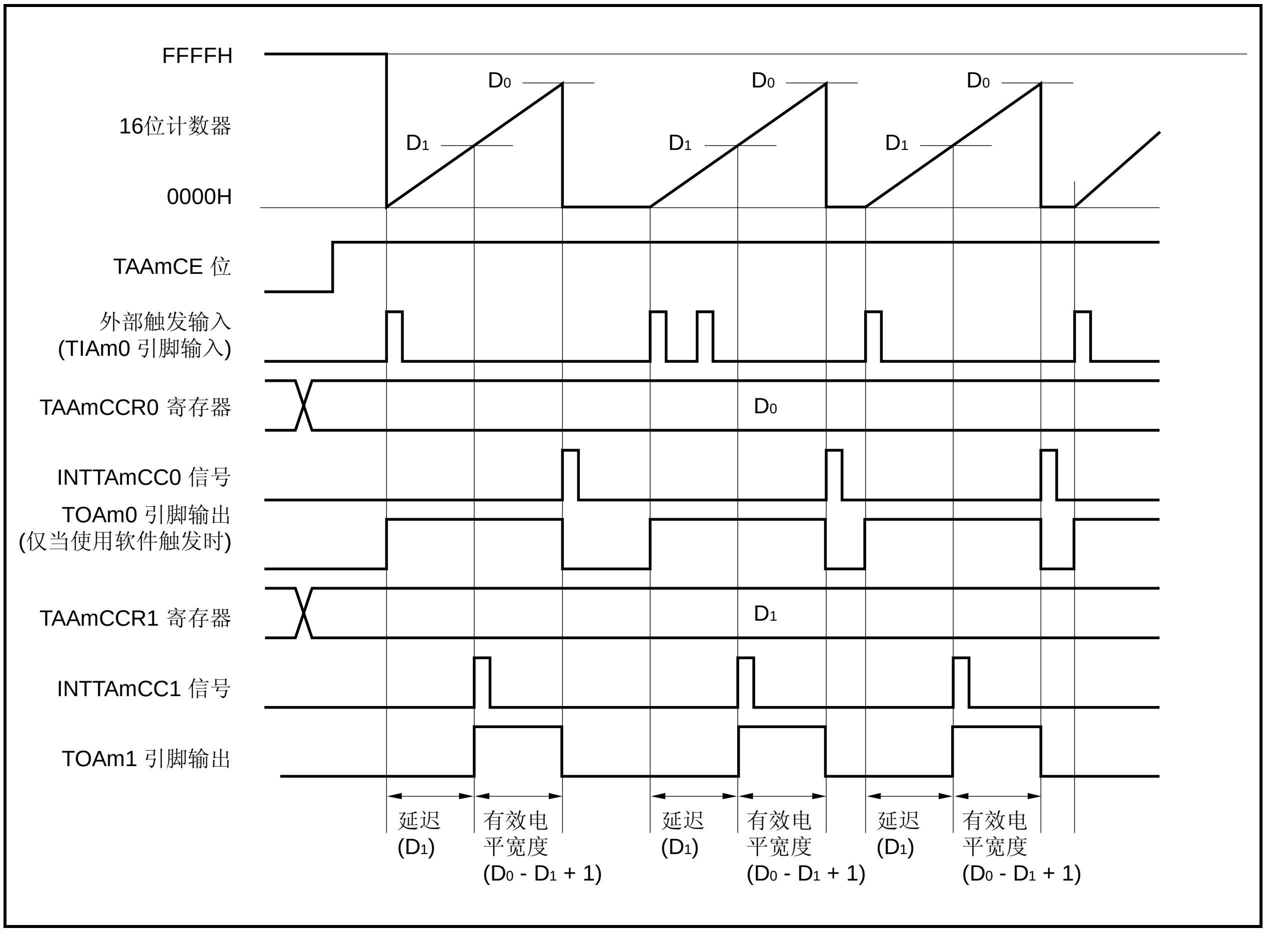


注　因为外部触发输入引脚（TIAm0）和定时器输出引脚（TOAm0）共用相同的复用功能，这两个功能不能同时使用。

注意事项　**在单次触发脉冲输出模式下，选择内部时钟作为计数时钟（将 TAAmCTL1.TAAmEEE 位清除为 0）。**

备注　V850E/IF3：m = 2, 4
V850E/IG3：m = 2 至 4

图 6-29. 单次触发脉冲输出模式的基本时序



当 TAAmCE 位置为 1 时，16 位计时器/事件计数器 AA 等待触发。在触发信号产生时，16 位计数器从 FFFFH 清除为 0000H，开始进行计数，并从 TOAm1 引脚输出一个单次触发脉冲。在输出单次触发脉冲后，16 位计数器设置为 FFFFH，计数停止，并重新等待触发。如果在单次触发脉冲输出期间再次产生触发信号，则忽略该触发信号。

单次触发脉冲的输出延迟时间和有效电平宽度可计算如下：

输出延迟时间＝（TAAmCCR1 寄存器的设置值）× 计数时钟周期

有效电平宽度＝（TAAmCCR0 寄存器的设置值－TAAmCCR1 寄存器的设置值＋1）× 计数时钟周期

当 16 位计数器的计数值与 CCR0 缓冲寄存器的值一致之后，产生一致中断请求信号（INTTAmCC0）。当 16 位计数器的计数值与 CCR1 缓冲寄存器的值一致时，会产生一致中断请求信号（INTTAmCC1）。

外部触发输入（TIAm0 引脚）的有效边沿或将软件触发（TAAmCTL1.TAAmEST）信号设置为 1，都可用作触发信号。

备注 V850E/IF3：m = 2, 4

V850E/IG3：m = 2 至 4

图 6-30. 单次触发脉冲输出模式的寄存器设置（1/2）

（a） TAAm 控制寄存器 0（TAAmCTL0）

<R>

	TAAmCE					TAAmCKS2	TAAmCKS1	TAAmCKS0
TAAmCTL0	0/1	0	0	0	0	0/1	0/1	0/1

TAAmCKS2～TAAmCKS0: 选择计数时钟

TAAmCE:
0: 停止计数
1: 使能计数

<R>

（b） TAAm 控制寄存器 1（TAAmCTL1）

	TAAaSYE	TAAmEST	TAAmEEE			TAAmMD2	TAAmMD1	TAAmMD0
TAAmCTL1	0	0/1	0	0	0	0	1	1

TAAmMD2～TAAmMD0:
0, 1, 1:
单次触发脉冲输出模式

TAAmEEE:
0: 计数时钟由 TAAmCKS0 至 TAAmCKS2 位选择

TAAmEST:
当写入 1 时，
产生软件触发

（c） TAAm I/O 控制寄存器 0（TAAmIOC0）

					TAAmOL1	TAAmOE1	TAAmOL0	TAAmOE0
TAAmIOC0	0	0	0	0	0/1	0/1	0/1	0/1注

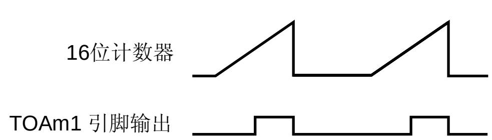
TAAmOE0:
0: 禁止 TOAm0 引脚输出
1: 使能 TOAm0 引脚输出

TAAmOL0:
在等待外部触发时，
TOAm0 引脚输出电平的设置
0: 低电平
1: 高电平

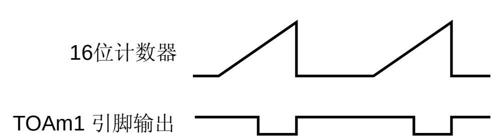
TAAmOE1:
0: 禁止 TOAm1 引脚输出
1: 使能 TOAm1 引脚输出

TAAmOL1:
在等待外部触发时，
TOAm1 引脚输出电平的设置
0: 低电平
1: 高电平

• 当 TAAmOL1 位 = 0



• 当 TAAmOL1 位= 1



注 在单次触发脉冲输出模式下不使用 TOAm0 引脚时，该位清除为 0。

图 6-30. 单次触发脉冲输出模式的寄存器设置（2/2）

（d） TAAm I/O 控制寄存器 2（TAAmIOC2）

							TAAmEES1	TAAmEES0	TAAmETS1	TAAmETS0
TAAmIOC2	0	0	0	0	0	0	0/1	0/1		

选择外部触发输入 (TIAm0 引脚)的有效边沿

（e） TAAm 计数器读取缓冲寄存器（TAAmCNT）

通过读取 TAAmCNT 寄存器，可以读出 16 位计数器的值。

（f）TAAm 捕获/比较寄存器 0 和 1（TAAmCCR0 和 TAAmCCR1）

若把 TAAmCCR0 寄存器的值设置为 D_0，TAAmCCR1 寄存器的值设置为 D_1，则单次触发脉冲的有效电平宽度和输出延迟时间的计算如下。

有效电平宽度＝（$D_0 - D_1$＋1）× 计数时钟周期
输出延迟时间＝ D_1 × 计数时钟周期

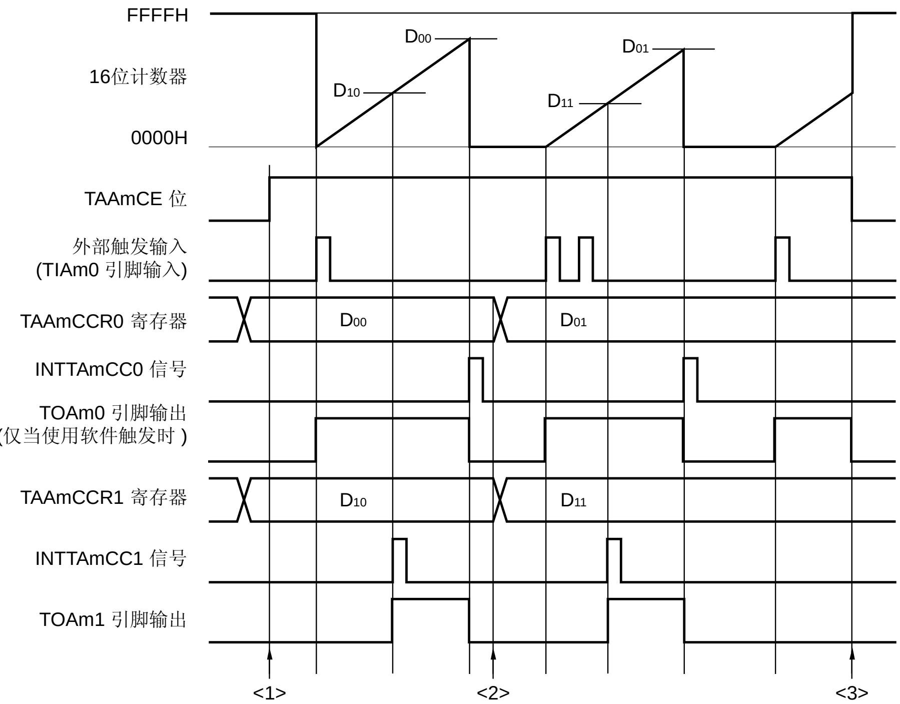
<R> 注意事项 **如果 TAAmCCR1 寄存器的设置值大于 TAAmCCR0 寄存器的值，即使在单次触发脉冲输出模式下，也不会输出单次触发脉冲。**

备注 **1.** 单次触发脉冲输出模式不使用 TAAm I/O 控制寄存器 1（TAAmIOC1）和 TAAm 选项寄存器 0（TAAmOPT0）。
2. V850E/IF3：m = 2, 4, a = 0, 1
V850E/IG3：m = 2 至 4, a = 0, 1

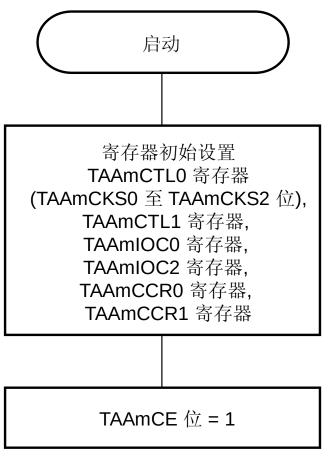
（1）单次触发脉冲输出模式下的操作流程

图 6-31. 单次触发脉冲输出模式下的软件处理流程

<R>



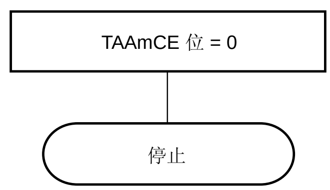
<1> 计数操作启动流程



在设置TAAmCE位为1之前，执行这些寄存器的初始化设置。

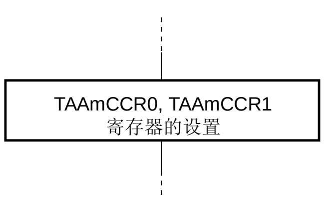
在启动计数(TAAmCE 位= 1)的同时，可以设置TAAmCKS0 至TAAmCKS2 位。触发等待状态。

<3> 计数操作停止流程



计数操作停止

<2> TAAmCCR0, TAAmCCR1 寄存器设置更改流程



在INTTAmCC0信号产生后，推荐立即重写TAAmCCRa寄存器，因为重写的值会立即送往CCRa缓冲寄存器。

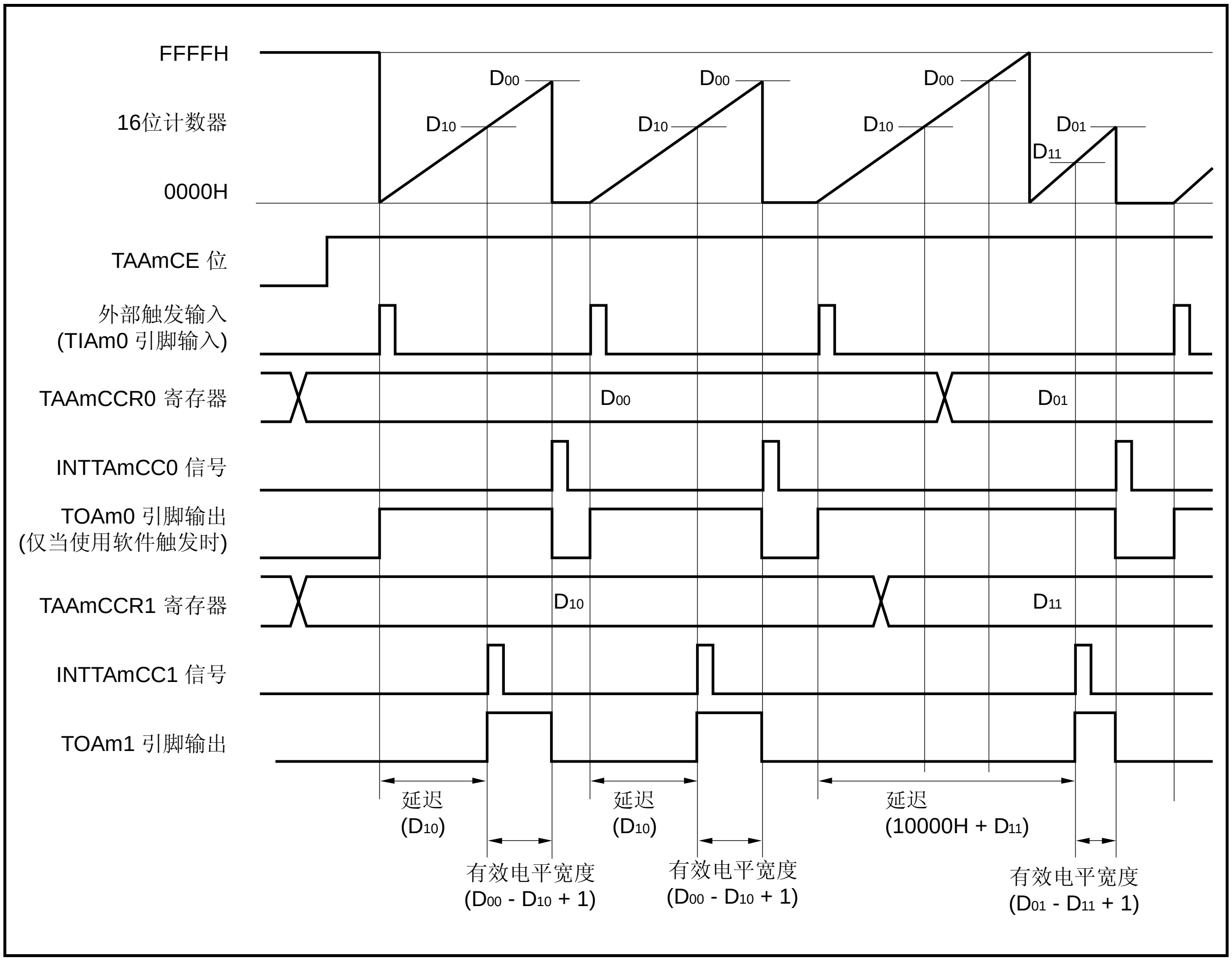
备注 V850E/IF3：m = 2, 4, a = 0, 1

V850E/IG3：m = 2 至 4, a = 0, 1

（2）单次触发脉冲输出模式的操作时序

（a） 重写 TAAmCCRa 寄存器的注意事项

如果在计数器工作期间，将 TAAmCCRa 寄存器的值重写为比当前值较小的值，16 位计数器可能会溢出。如果发生溢出时，停止计数，然后再改变设置值。



假设 D_{00}> D_{01} 而且 D_{10}> D_{11}，TAAmCCR0 寄存器由 D_{00} 重写为 D_{01}，TAAmCCR1 寄存器由 D_{10} 重写为 D_{11}，如果在 16 位计数器的计数值大于 D_{11} 且小于 D_{10} 时重写 TAAmCCR1 寄存器，并且在计数值大于 D_{01} 且小于 D_{00} 时重写 TAAmCCR0 寄存器，则一旦重写寄存器，就会立即在各设置值上反映出来，并与计数值进行比较。计数器向上计数到 FFFFH 后，从 0000H 开始重新向上计数。当计数值与 D_{11} 一致时，计数器会产生 INTTAmCC1 信号并使 TOAm1 引脚有效。当计数值与 D_{01} 一致时，计数器产生 INTTAmCC0 信号，使 TOAm1 引脚失效，并停止计数。

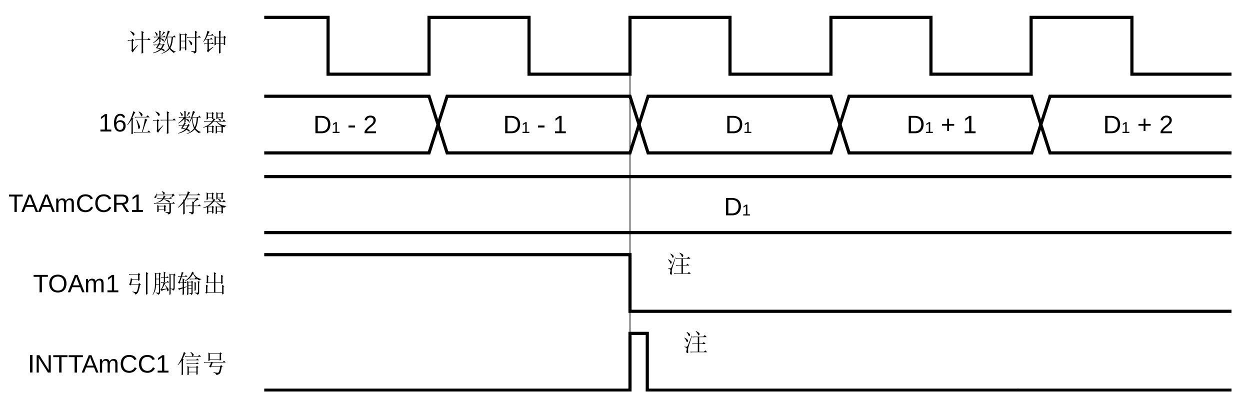
因此，计数器输出脉冲的延迟时段或有效时段可能会与原先预计的单次触发脉冲的输出有所不同。

备注 V850E/IF3：m = 2, 4, a = 0, 1

V850E/IG3：m = 2 至 4, a = 0, 1

（b） 一致中断请求信号（INTTAmCC1）的产生时序

在单次触发脉冲输出模式下，一致中断请求信号（INTTAmCC1）的发生时序和其它情况下的 INTTAmCC1 信号发生时序不同，当 16 位计数器的计数值与 TAAmCCR1 寄存器的值一致时，产生 INTTAmCC1 信号。



注　时序实际延迟一个工作时钟（fxx）.

备注　V850E/IF3：m = 2, 4
V850E/IG3：m = 2 至 4

通常在 16 位计数器的当前计数值与 TAAmCCR1 寄存器的值一致后，在下一次向上计数时同步发生 INTTAmCC1 信号。

但是，在单次触发脉冲输出模式下，INTTAmCC1 信号会提早一个计数时钟周期产生。该时序的变化是为了与 TOAm1 引脚输出信号变化的时序一致。

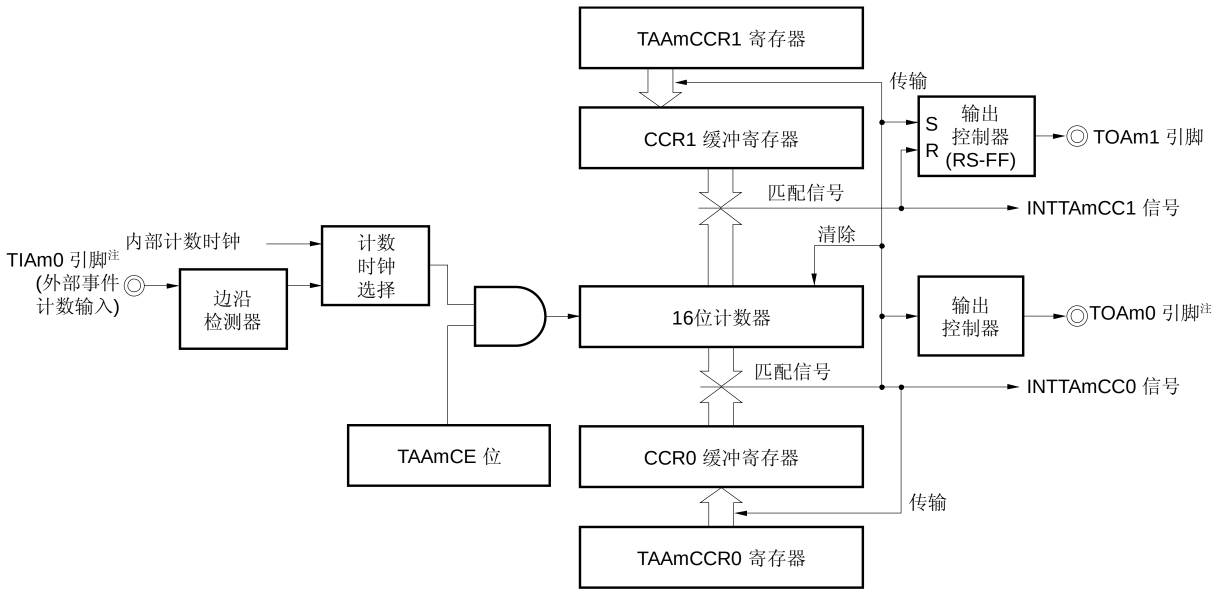
6.6.5 PWM 输出模式（TAAmMD2 至 TAAmMD0 位 = 100）

只有在 TAA2, TAA3（仅限 V850E/IG3）和 TAA4 时，该模式有效。

在 PWM 输出模式下，当 TAAmCTL0.TAAmCE 位设置为 1 时，从 TOAm1 引脚输出 PWM 波形。

而且，由 TOAm0 引脚输出的占空比 50%的 PWM 波形，其半周期等于 TAAmCCR0 寄存器的设置值 + 1。

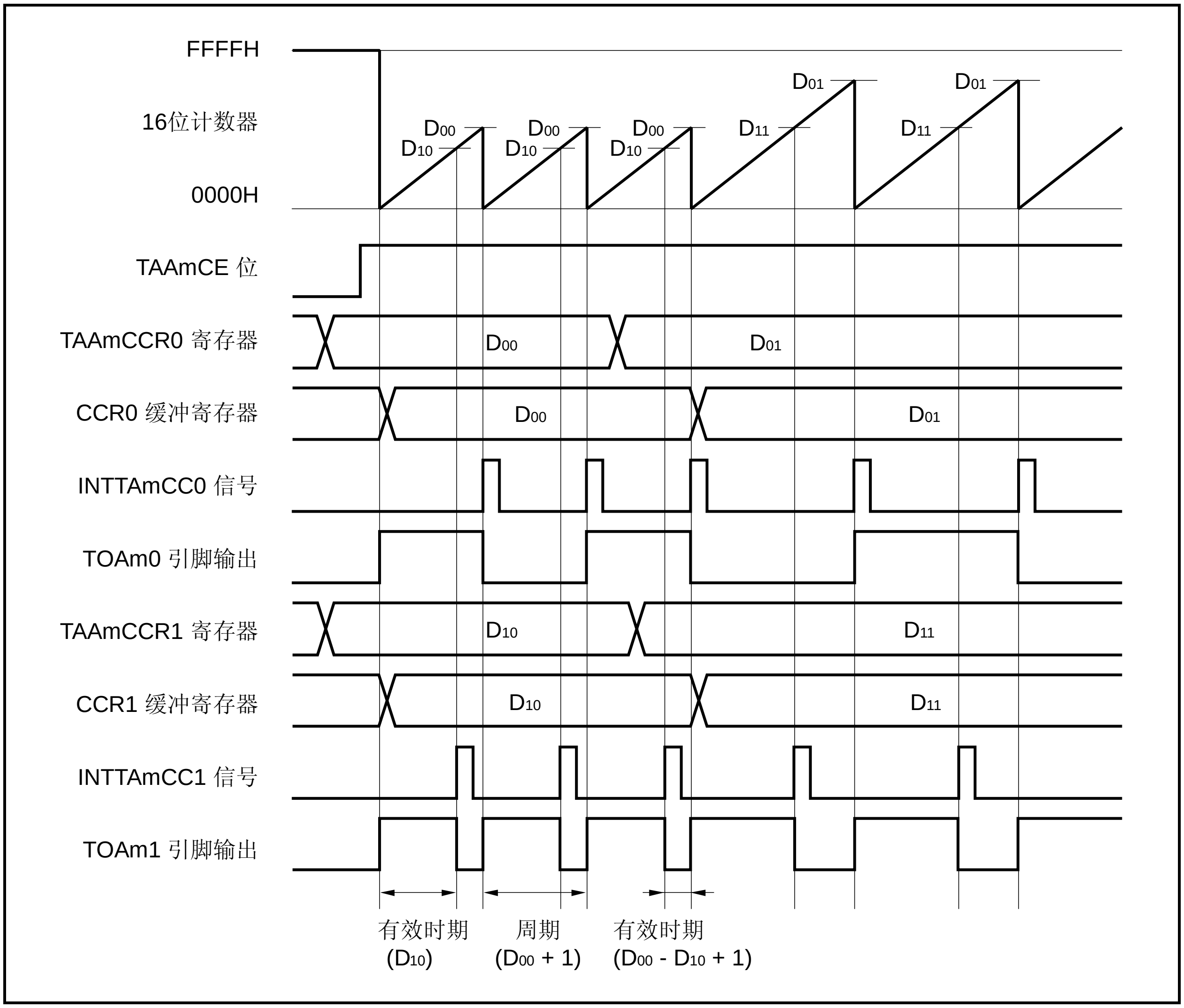
图 6-32. PWM 输出模式的配置



注 因为外部事件计数输入引脚（TIAm0）和定时器输出引脚（TOAm0）共用相同的复用功能引脚，这两个功能不能同时使用。

备注 V850E/IF3：m = 2, 4
V850E/IG3：m = 2 至 4

图 6-33. PWM 输出模式的基本时序



当 TAAmCE 位置为 1 时，16 位计数器从 FFFFH 清除为 0000H，计数开始，同时从 TOAm1 引脚输出 PWM 波形。PWM 波形的有效电平宽度、周期、占空比系数的计算如下。

有效电平宽度＝（TAAmCCR1 寄存器的设置值 ）× 计数时钟周期
周期＝（TAAmCCR0 寄存器的设置值 ＋ 1）× 计数时钟周期
占空比系数＝（TAAmCCR1 寄存器的设置值）/（TAAmCCR0 寄存器的设置值 ＋ 1）

在计数操作时，可通过重写 TAAmCCRa 寄存器来改变 PWM 波形。当 16 位计数器的计数值与 CCR0 缓冲寄存器的值一致时，16 位计数器清除为 0000H 时，新写入的值会发生作用。

当 16 位计数器的计数值与 CCR0 缓冲寄存器的值一致之后，进行下一次计数时，会同步产生一致中断请求信号 INTTAmCC0，且 16 位计数器清除为 0000H。当 16 位计数器的计数值与 CCR1 缓冲寄存器的值一致时，会生成一致中断请求信号 INTTAmCC1。

当 16 位计数器的计数值与 CCRa 缓冲寄存器的值一致时，TAAmCCRa 寄存器的设定值传输到 CCRa 缓冲寄存器，且 16 位计数器清除为 0000H。

备注 V850E/IF3：m = 2, 4, a = 0, 1
V850E/IG3：m = 2 至 4, a = 0, 1

图 6-34. PWM 输出模式的寄存器设置（1/2）

（a） TAAm 控制寄存器 0（TAAmCTL0）

	TAAmCE					TAAmCKS2	TAAmCKS1	TAAmCKS0
TAAmCTL0	0/1	0	0	0	0	0/1	0/1	0/1

- TAAmCKS2～TAAmCKS0：选择计数时钟注
- TAAmCE：
  0: 停止计数
  1: 使能计数

注　当 TAAmCTL1.TAAmEEE 位 = 1 时，该设置无效。

（b） TAAm 控制寄存器 1（TAAmCTL1）

	TAAaSYE	TAAmEST	TAAmEEE			TAAmMD2	TAAmMD1	TAAmMD0
TAAmCTL1	0	0	0/1	0	0	1	0	0

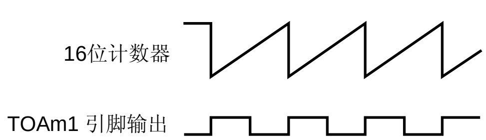
- TAAmMD2～TAAmMD0：
  1, 0, 0:
  PWM 输出模式
- TAAmEEE：
  0: 计数时钟由 TAAmCKS0 至 AAmCKS2 位选择
  1: 用外部事件计数输入信号计数

（c） TAAm I/O 控制寄存器 0（TAAmIOC0）

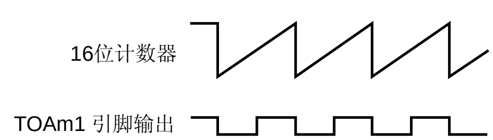
					TAAmOL1	TAAmOE1	TAAmOL0	TAAmOE0
TAAmIOC0	0	0	0	0	0/1	0/1	0/1	0/1注

- TAAmOE0：
  0: 禁止 TOAm0 引脚输出
  1: 使能 TOAm0 引脚输出
- TAAmOL0：在计数操作前，TOAm0引脚输出电平的设置
  0: 低电平
  1: 高电平
- TAAmOE1：
  0: 禁止 TOAm1 引脚输出
  1: 使能 TOAm1 引脚输出
- TAAmOL1：在计数操作前，TOAm1 引脚输出电平的设置
  0: 低电平
  1: 高电平

• 当 TAAmOL1 位= 0



• 当 TAAmOL1 位 = 1



注　PWM 输出模式不使用 TOAm0 引脚时，该位清除为 0。

图 **6-34. PWM** 输出模式的寄存器设置（**2/2**）

（d） TAAm I/O 控制寄存器 2（TAAmIOC2）

					TAAmEES1	TAAmEES0	TAAmETS1	TAAmETS0
TAAmIOC2	0	0	0	0	0/1	0/1	0	0

TAAmEES1, TAAmEES0：选择外部事件计数输入(TIAm0引脚)的有效边沿

（e） TAAm 计数器读取缓冲寄存器（TAAmCNT）

通过读取 TAAmCNT 寄存器，可以读出 16 位计数器的值。

（f）TAAm 捕获/比较寄存器 0 和 1（TAAmCCR0 和 TAAmCCR1）

如果将 TAAmCCR0 寄存器的值设置为 D_0，将 TAAmCCR1 寄存器的值设置为 D_1，则 PWM 波形的周期和有效电平的计算如下：

周期＝（D_0＋1）× 计数时钟周期

有效电平宽度＝ D_1 × 计数时钟周期

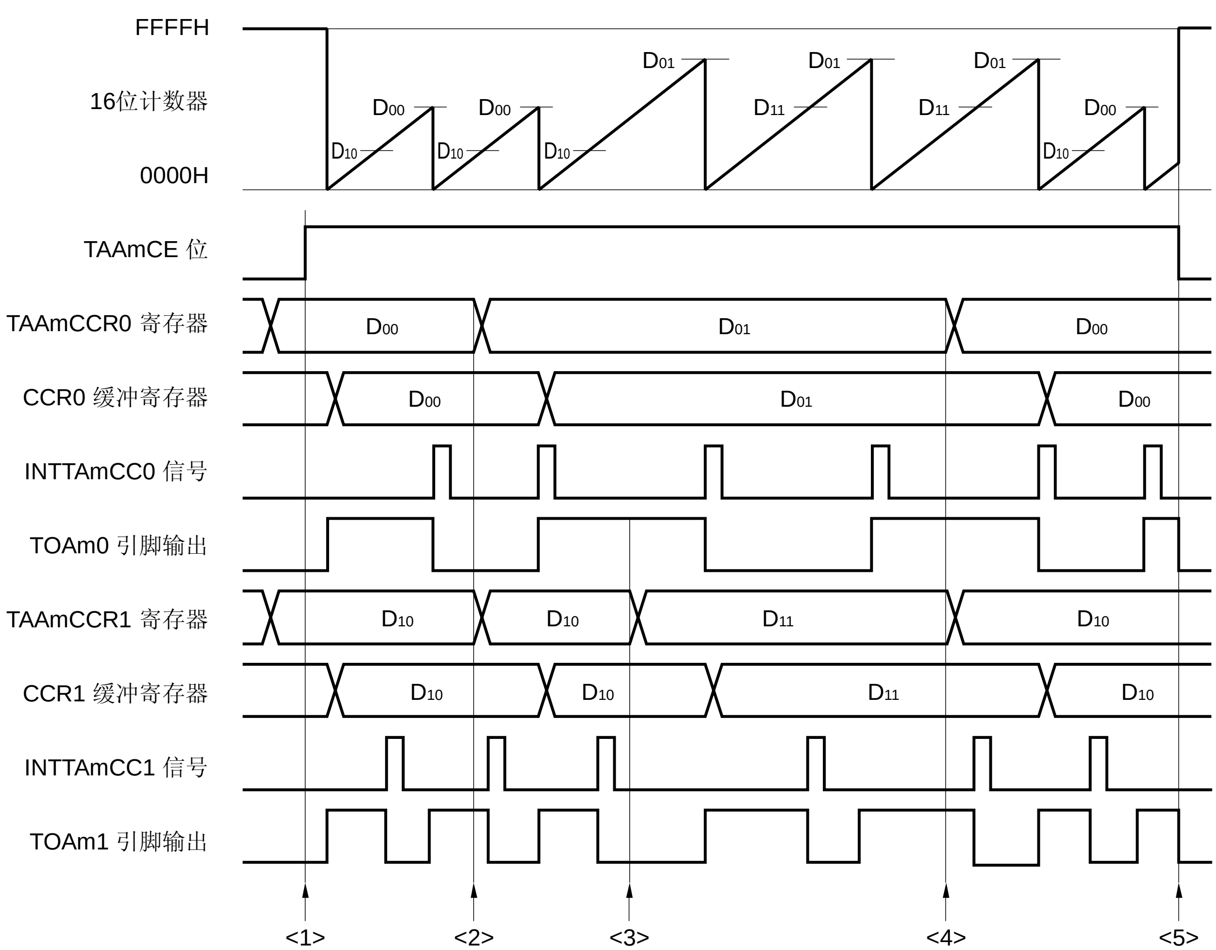
备注 **1.** PWM 输出模式不使用 TAAm I/O 控制寄存器 1（TAAmIOC1）和 TAAm 选项寄存器 0（TAAmOPT0）。

2. V850E/IF3：m = 2, 4, a = 0, 1

V850E/IG3：m = 2 至 4, a = 0, 1

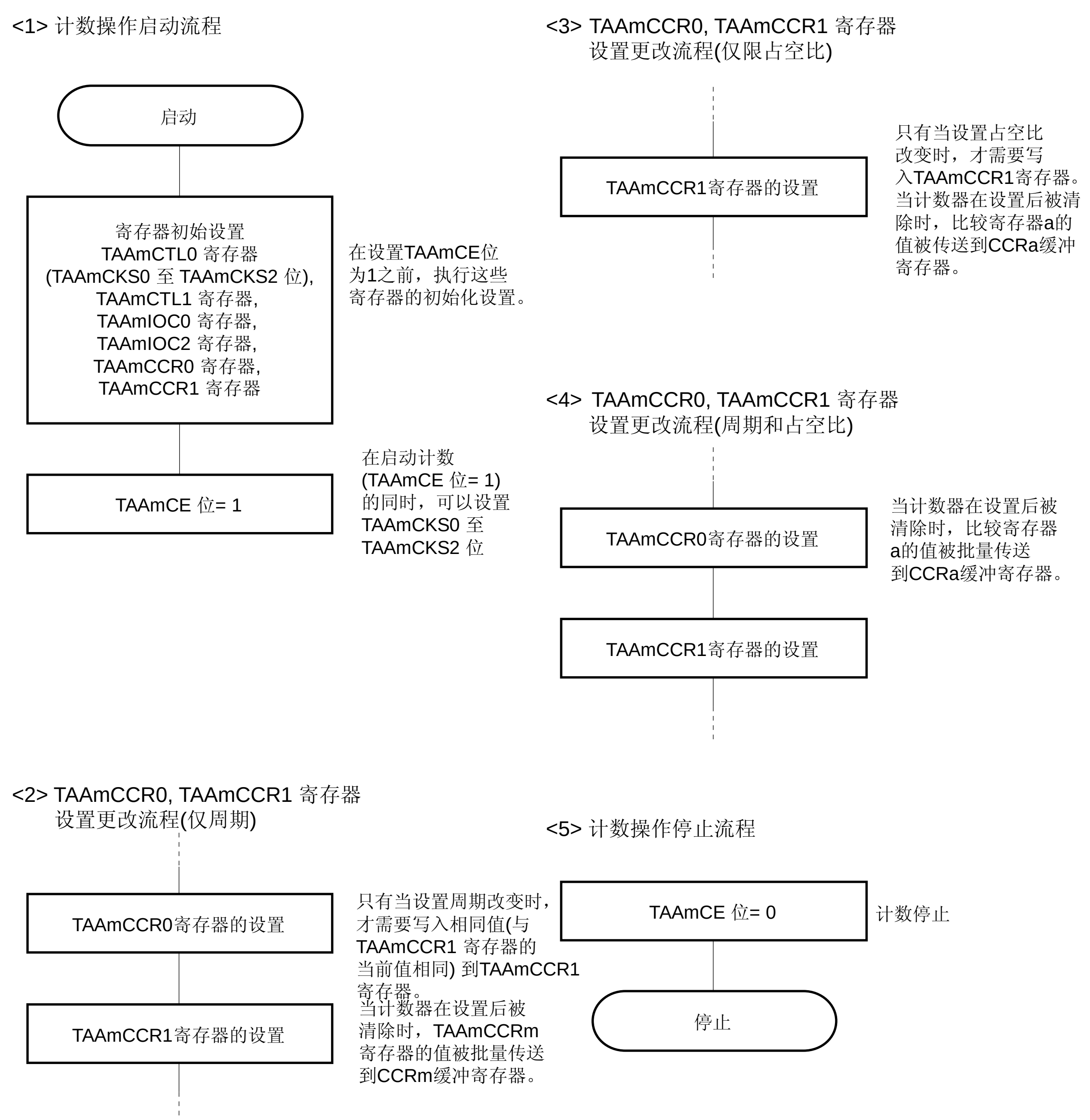
（1）**PWM** 输出模式的操作流程

图 **6-35. PWM** 输出模式的软件处理流程（**1/2**）



备注　V850E/IF3：m = 2, 4
　　　V850E/IG3：m = 2 至 4

图 6-35. PWM 输出模式的软件处理流程（2/2）



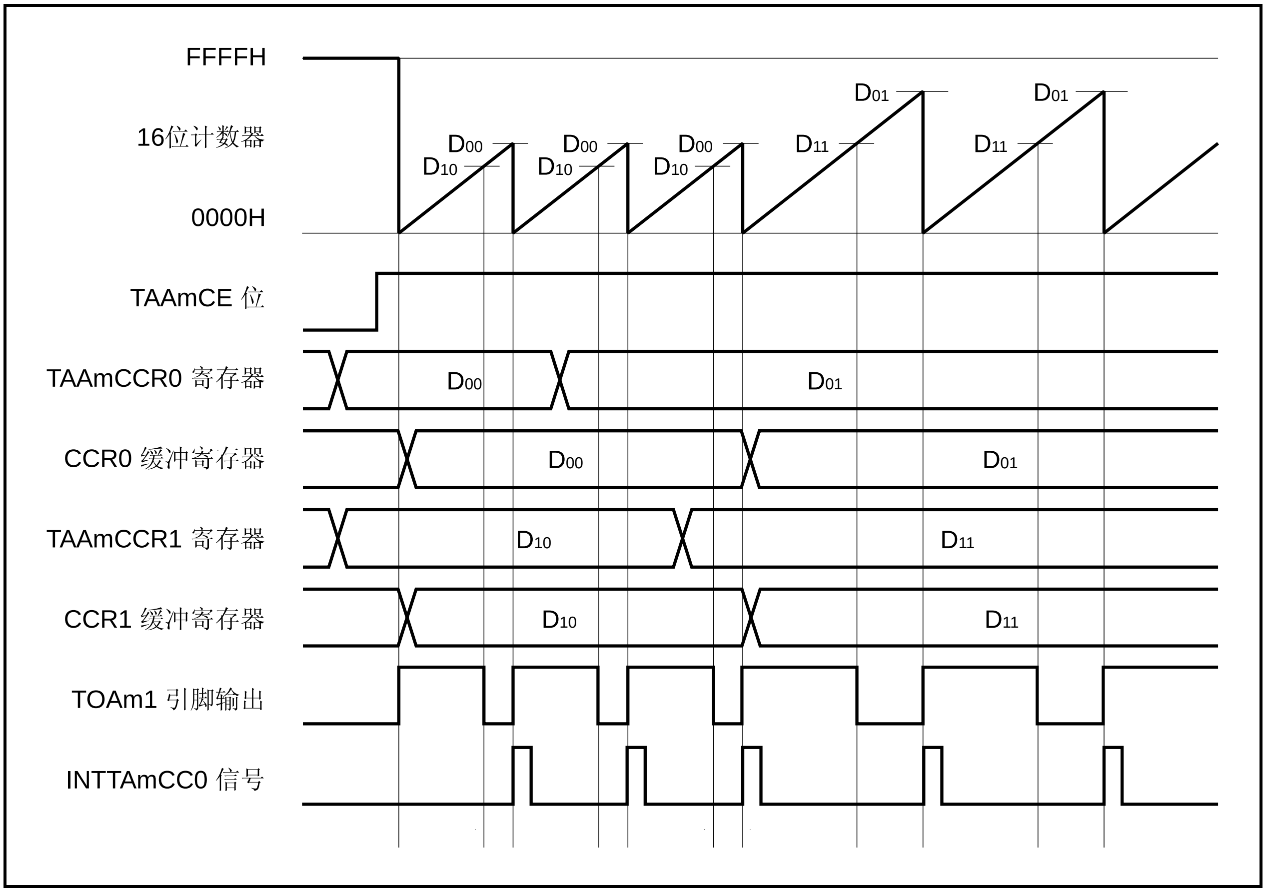
备注　V850E/IF3：m = 2, 4, a = 0, 1
　　　V850E/IG3：m = 2 至 4, a = 0, 1

（2）**PWM 输出模式操作时序**

（a） **操作期间改变脉冲宽度**

计数器操作过程中，若要改变 PWM 波形，应最后写入 TAAmCCR1 寄存器。

当检测到 INTTAmCC1 信号以后，在写入 TAAmCCR1 寄存器之后重写 TAAmCCRa 寄存器。



要把数据从 TAAmCCRa 寄存器传输到 CCRa 缓冲寄存器，必须对 TAAmCCR1 寄存器进行写入操作。

此时如果 PWM 波形的周期和有效电平都要改变，应先将周期设定到 TAAmCCR0 寄存器，然后再将有效电平设定到 TAAmCCR1 寄存器。

如果只改变 PWM 波形的周期，应首先把周期设定给 TAAmCCR0 寄存器，然后将相同值（和 TAAmCCR1 寄存器的当前值相等）写入 TAAmCCR1 寄存器。

如果只改变 PWM 波形的有效电平宽度（占空比），仅需要设置 TAAmCCR1 寄存器。

当数据写入 TAAmCCR1 寄存器之后，在对 16 位计数器清零的同时，写入 TAAmCCRa 寄存器的值传输到 CCRa 缓冲寄存器，并作为与 16 位计数器进行比较的值。

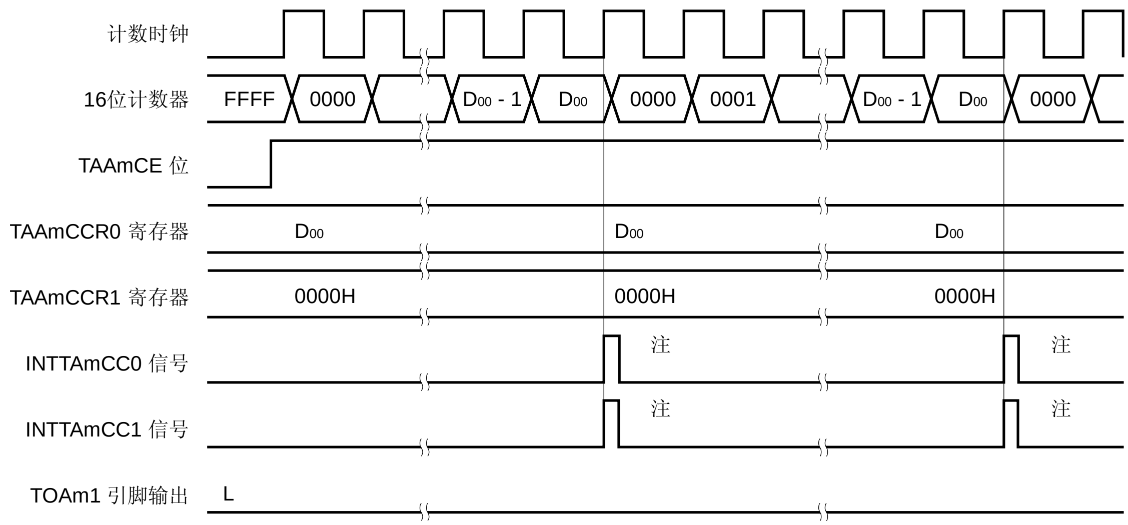
在写入 TAAmCCR1 寄存器一次之后，要再次写入 TAAmCCR0 或 TAAmCCR1 寄存器，应该在 INTTAmCC0 信号产生之后进行。否则，CCRa 缓冲寄存器的值可能不确定，因为把数据从 TAAmCCRa 寄存器传输到 CCRa 缓冲寄存器的时序会和写入 TAAmCCRa 寄存器时序发生冲突。

备注 V850E/IF3：m = 2, 4, a = 0, 1

V850E/IG3：m = 2 至 4, a = 0, 1

（b） 0%/100% PWM 波形的输出

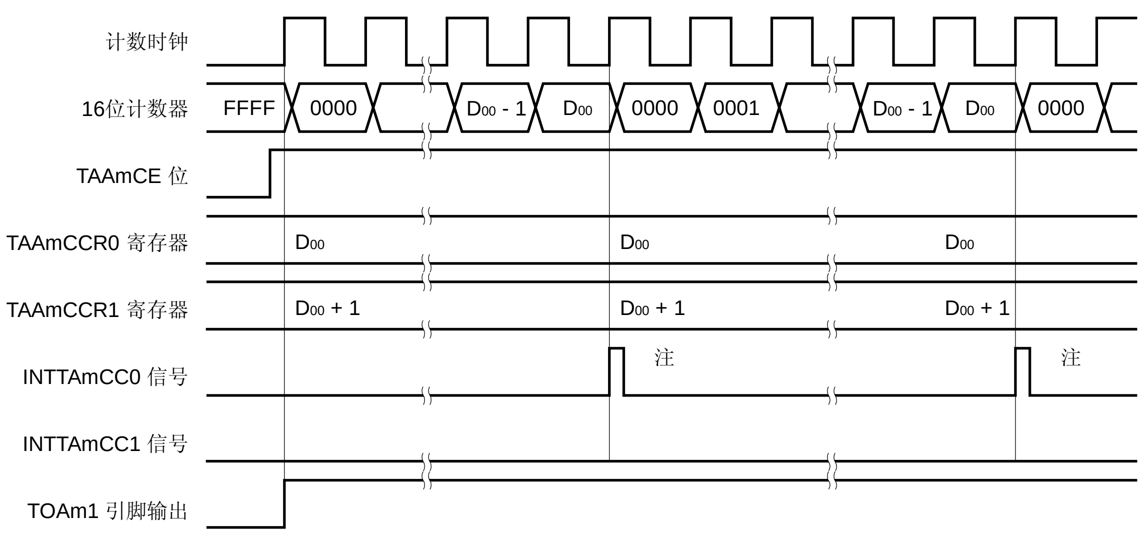
要输出一个 0% 波形，将 TAAmCCR1 寄存器设置为 0000H。在 16 位计数器的计数值和 CCR0 缓冲寄存器的值发生一致之后，16 位计数器清除为 0000H，且在下一个时序产生 INTTAmCC0 和 INTTAmCC1 信号。



注 时序实际延迟一个工作时钟（fxx）.

备注 V850E/IF3：m = 2, 4
V850E/IG3：m = 2 至 4

要输出 100%波形，将 TAAmCCR1 寄存器的值设置为（TAAmCCR0 寄存器的设置值 + 1）。如果 TAAmCCR0 寄存器的设置值是 FFFFH，不能生成 100%输出。

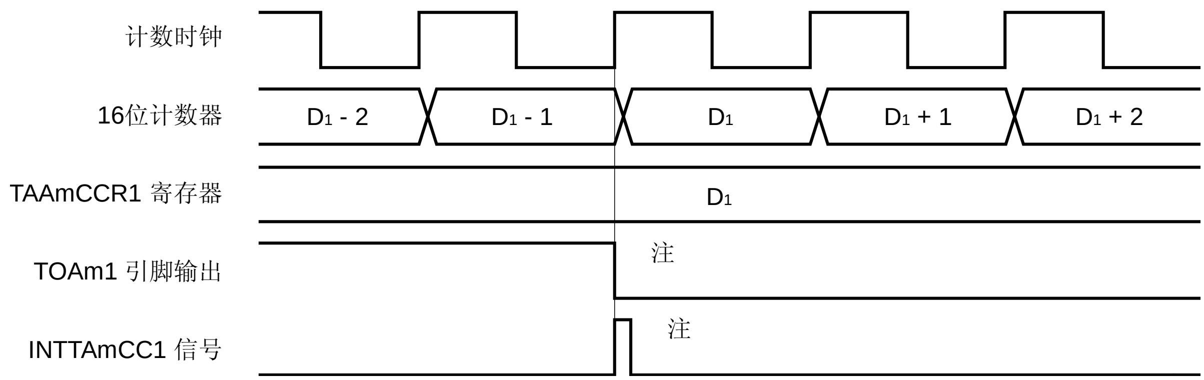


注 时序实际延迟一个工作时钟（fxx）.

备注 V850E/IF3：m = 2, 4
V850E/IG3：m = 2 至 4

（c） 一致中断请求信号（INTTAmCC1）的产生时序

在 PWM 输出模式下，INTTAmCC1 信号的产生时序与其它情况下 INTTAmCC1 信号的产生时序不同；当 16 位计数器的计数值与 TAAmCCR1 寄存器的值一致时，产生 INTTAmCC1 中断请求信号。



注 时序实际延迟一个工作时钟（fxx）.

备注 V850E/IF3：m = 2, 4

V850E/IG3：m = 2 至 4

通常在 16 位计数器的计数值与 TAAmCCR1 寄存器的值一致之后，在下一次计数时同步发生 INTTAmCC1 信号。

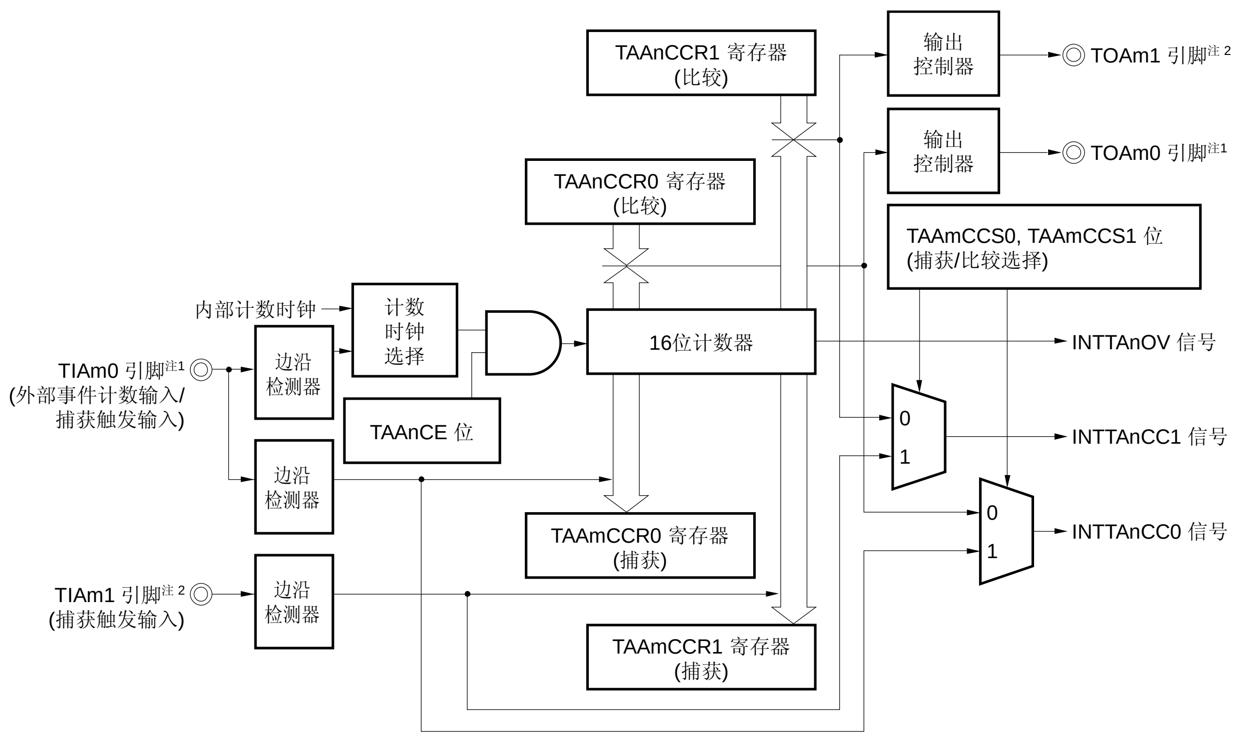
但是，在 PWM 输出模式下，INTTAmCC1 信号会提早一个计数时钟周期产生。该时序的变化是为了与 TOAm1 引脚输出信号变化的时序一致。

6.6.6 自由运行定时器模式（TAAnMD2 至 TAAnMD0 位 = 101）

在 TAA2 至 TAA4 时，比较功能都有效。只有在 TAA2, TAA3（仅限 V850E/IG3）和 TAA4 时，捕获功能有效。

在自由运行定时器模式下，当 TAAnCTL0.TAAnCE 位被设置为 1 时，16 位定时器/事件计数器 AA 开始计数。此时，根据 TAAmOPT0.TAAmCCS0 和 TAAmOPT0.TAAmCCS1 位的设置，TAAmCCR0 和 TAAmCCR1 寄存器可以用作比较寄存器或捕获寄存器。

图 6-36. 自由运行定时器模式的配置



注 **1.** 因为外部事件计数输入引脚（TIAm0）、捕获 trigger 输入引脚（TIAm0）和定时器输出引脚（TOAm0）共用相同的复用功能引脚，这两个或更多的功能不能同时使用。

2. 因为捕获触发输入引脚（TIAm1）和定时器输出引脚（TOAm1）共用相同的复用功能引脚，这两个功能不能同时使用。

备注 V850E/IF3：n = 0 至 4, m = 2, 4

V850E/IG3：n = 0 至 4, m = 2 至 4

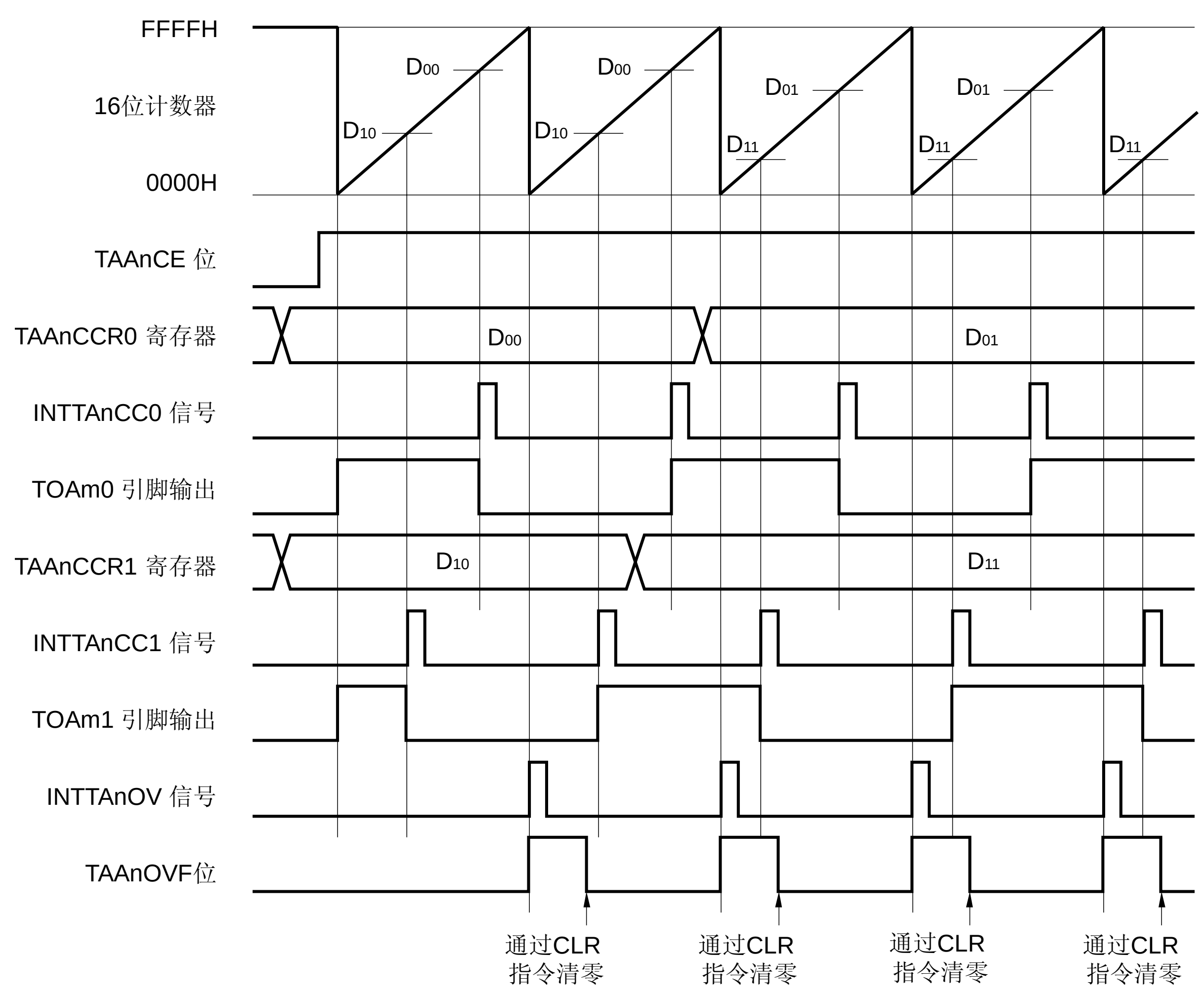
• 比较操作

当 TAAnCE 位置为 1 时，16 位定时器/事件计数器 AA 开始计数，TOAma 引脚的输出信号反转。此后，当 16 位计数器的计数值与 TAAnCCRa 寄存器的设定值一致时，产生一个一致中断请求信号（INTTAnCCa），而且 TOAma 引脚的输出信号反转。

16 位计数器继续随着计数时钟同步计数。当它向上计数到 FFFFH 时，会在下一个计数时钟时产生一个溢出中断请求信号（INTTAnOV），同时计数器清除为 0000H，然后继续进行计数。此时，溢出标志（TAAnOPT0.TAAnOVF 位）也置为 1，确认溢出标志位被置为 1 然后通过软件运行 CLR 指令将其清除为 0。

当计数器工作时，可以重写 TAAnCCRa 寄存器。如果进行重写，则新写入的值会立刻反映出来，并与计数值进行比较。

图 6-37. 自由运行定时器模式的基本时序（比较功能）



备注　V850E/IF3：n = 0 至 4, m = 2, 4, a = 0, 1

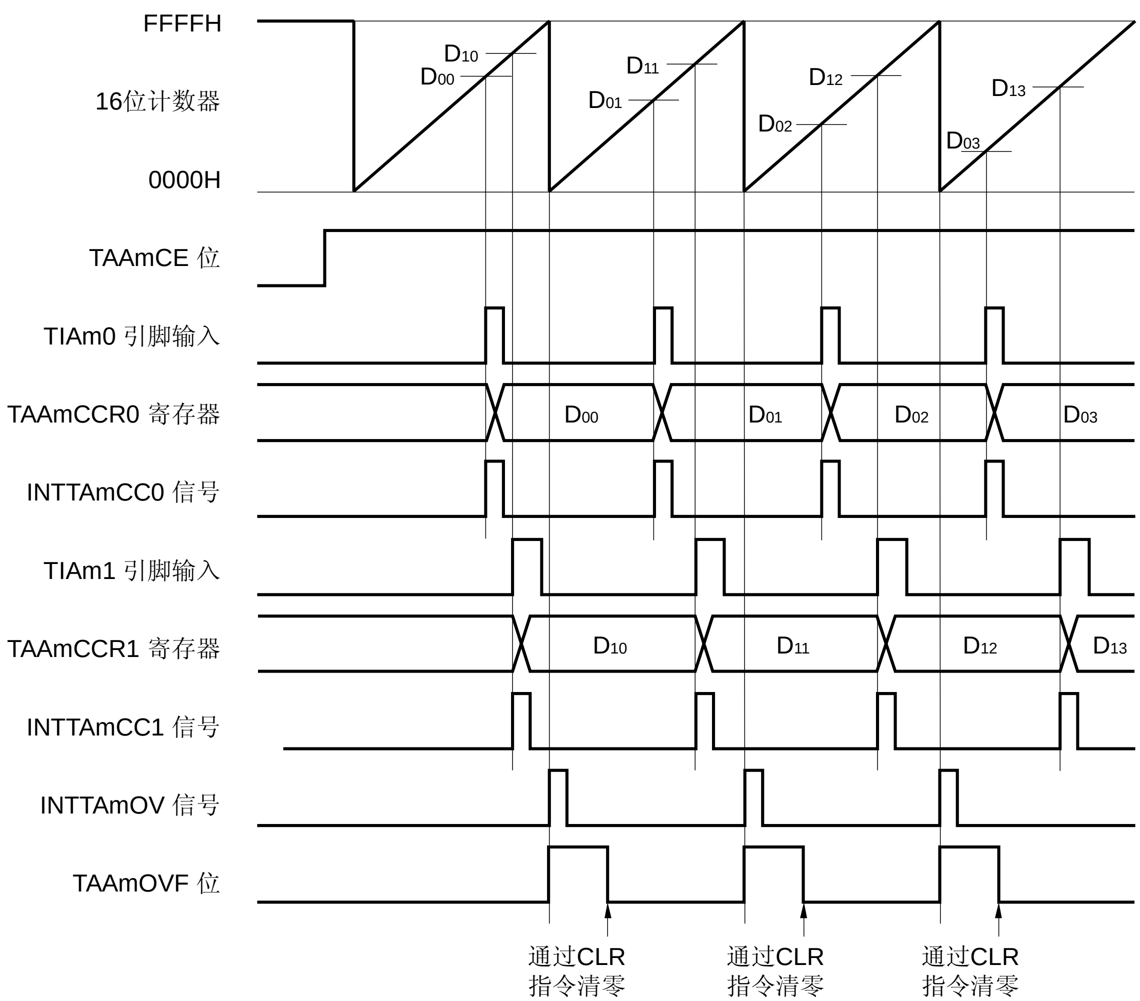
　　　V850E/IG3：n = 0 至 4, m = 2 至 4, a = 0, 1

- 捕获操作

当 TAAmCE 位置为 1 时，16 位计数器开始计数。当检测到信号的有效边沿输入 TIAma 引脚时，16 位计数器的计数值存储在 TAAmCCRa 寄存器中，并产生一个捕获中断请求信号（INTTAnCCa）。

16 位计数器继续随着计数时钟同步计数。当它计数到 FFFFH 时，会在下一个计数时钟时产生一个溢出中断请求信号（INTTAmOV），同时计数器清除为 0000H，然后继续进行计数。此时，溢出标志（TAAmOPT0.TAAmOVF 位）置为 1，确认溢出标志位置为 1 然后通过软件运行 CLR 指令将其清除为 0。

图 6-38. 自由运行定时器模式的基本时序（捕获功能）



备注　V850E/IF3：m = 2, 4, a = 0, 1
　　　V850E/IG3：m = 2 至 4, a = 0, 1

图 6-39. 自由运行定时器模式的寄存器设置（1/2）

（a） TAAn 控制寄存器 0（TAAnCTL0）

	TAAnCE					TAAnCKS2	TAAnCKS1	TAAnCKS0
TAAnCTL0	0/1	0	0	0	0	0/1	0/1	0/1

TAAnCKS2 至 TAAnCKS0：选择计数时钟[注]

TAAnCE：
0: 停止计数
1: 使能计数

注 当 TAAmCTL1.TAAmEEE 位 = 1 时，该设置无效。

（b） TAAn 控制寄存器 1（TAAnCTL1）

	TAAaSYE	TAAmEST	TAAmEEE			TAAnMD2	TAAnMD1	TAAnMD0
TAAnCTL1	0	0	0/1	0	0	1	0	1

TAAnMD2 至 TAAnMD0：
1, 0, 1:
自由运行定时器模式

TAAmEEE：
0: 计数时钟由TAAmCKS0 至 TAAmCKS2 位选择
1: 用外部事件计数
输入信号计数

（c） TAAm I/O 控制寄存器 0（TAAmIOC0）

					TAAmOL1	TAAmOE1	TAAmOL0	TAAmOE0
TAAmIOC0	0	0	0	0	0/1	0/1	0/1	0/1

TAAmOE0：
0: 禁止 TOAm0 引脚输出
1: 使能 TOAm0 引脚输出

TAAmOL0：
在计数操作前
TOAm0引脚输出电平的设置
0: 低电平
1: 高电平

TAAmOE1：
0: 禁止 TOAm1 引脚输出
1: 使能 TOAm1 引脚输出

TAAmOL1：
在计数操作前
TOAm1 引脚输出电平的设置
0: 低电平
1: 高电平

（d） TAAm I/O 控制寄存器 1（TAAmIOC1）

					TAAmIS3	TAAmIS2	TAAmIS1	TAAmIS0
TAAmIOC1	0	0	0	0	0/1	0/1	0/1	0/1

TAAmIS1, TAAmIS0：选择TIAm0 引脚输入[注]的有效边沿

TAAmIS3, TAAmIS2：选择TIAm1 引脚输入[注]的有效边沿

注 将未使用复用外部输入信号的有效边沿设置为“无边沿检测”。

图 **6-39.** 自由运行定时器模式的寄存器设置（**2/2**）

（e） TAAm I/O 控制寄存器 2（TAAmIOC2）

					TAAmEES1	TAAmEES0	TAAmETS1	TAAmETS0
TAAmIOC2	0	0	0	0	0/1	0/1	0	0

TAAmEES1, TAAmEES0：选择外部事件计数输入(TIAm0 引脚)[注]的有效边沿

注　将未使用复用外部输入信号的有效边沿设置为“无边沿检测”。

（f） TAAn 选项寄存器 0（TAAnOPT0）

			TAAmCCS1	TAAmCCS0				TAAnOVF
TAAnOPT0	0	0	0/1	0/1	0	0	0	0/1

TAAnOVF：溢出标志

TAAmCCS0：指定 TAAmCCR0 寄存器功能作为捕获寄存器或比较寄存器
0: 比较寄存器
1: 捕获寄存器

TAAmCCS1：指定 TAAmCCR1 寄存器功能作为捕获寄存器或比较寄存器
0: 比较寄存器
1: 捕获寄存器

（g） TAAn 计数器读取缓冲寄存器（TAAnCNT）

通过读取 TAAnCNT 寄存器，可以读出 16 位计数器的值。

（h） TAAn 捕获/比较寄存器 0 和 1（TAAnCCR0 和 TAAnCCR1）

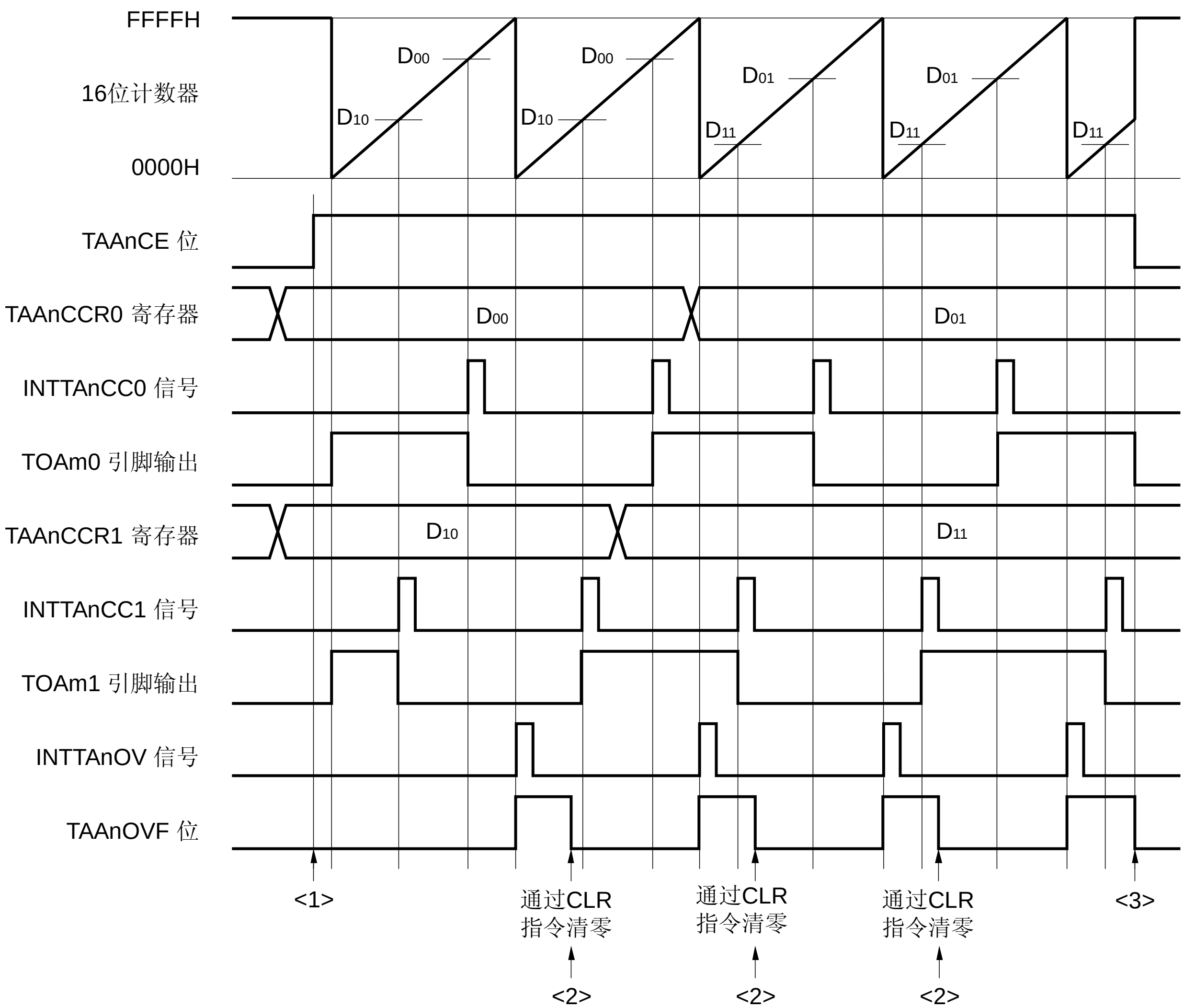
根据 TAAmOPT0.TAAmCCSa 位的设置，这些寄存器可以用作捕获寄存器或比较寄存器。
当该寄存器用作捕获寄存器时，在检测到信号的有效边沿输入 TIAma 引脚时，存储 16 位计数器的当前计数值。
当该寄存器用作比较寄存器时，而且 TAAnCCRa 寄存器的值设置为 D_a 时，则当计数器达到（D_a ＋1）时会产生 INTTAnCCa 信号，同时，TOAm0 和 TOAm1 引脚的输出信号反转。

备注　V850E/IF3：n = 0 至 4, m = 2, 4, a = 0, 1
　　　V850E/IG3：n = 0 至 4, m = 2 至 4, a = 0, 1

（1）自由运行定时器模式的操作流程

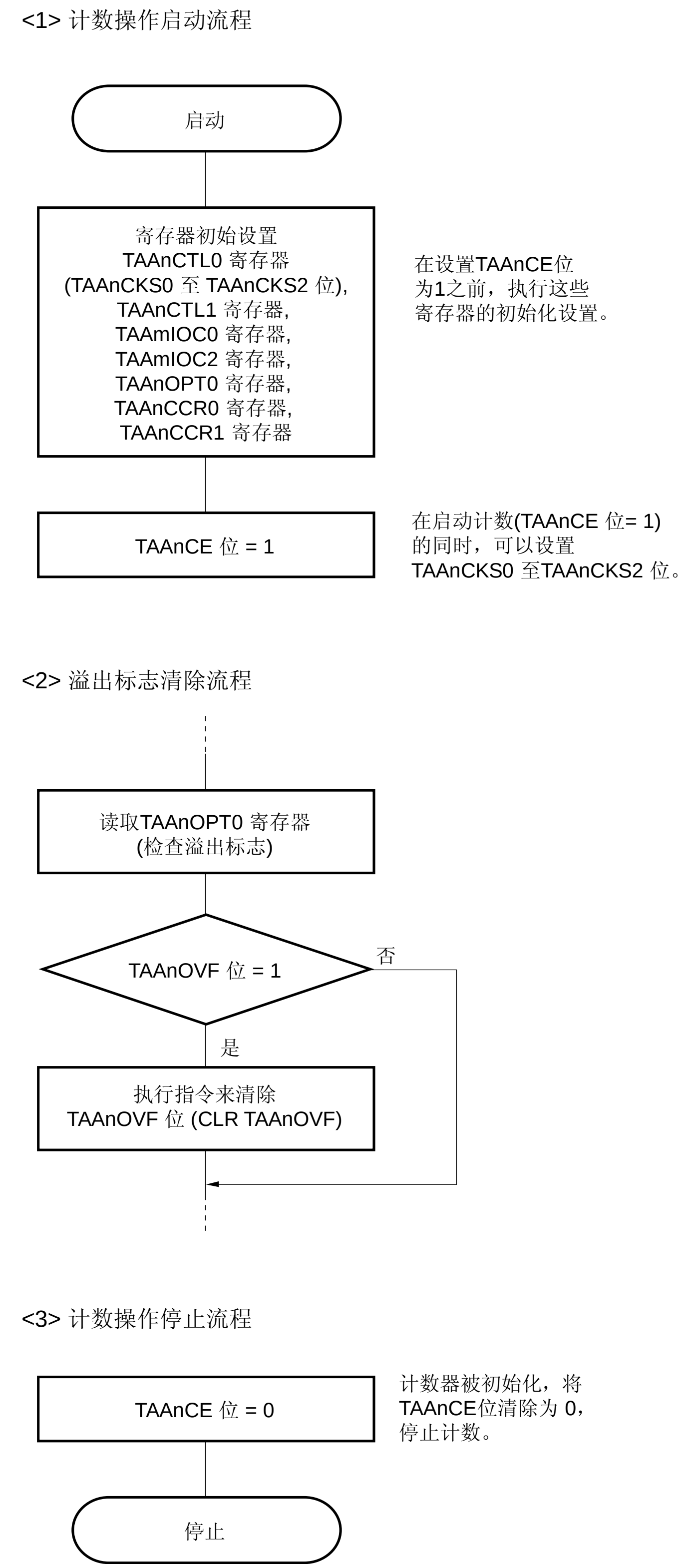
（a） 当捕获/比较寄存器用作比较寄存器时

图 6-40. 自由运行定时器模式的软件处理流程（比较功能）（1/2）



备注 n = 0 至 4

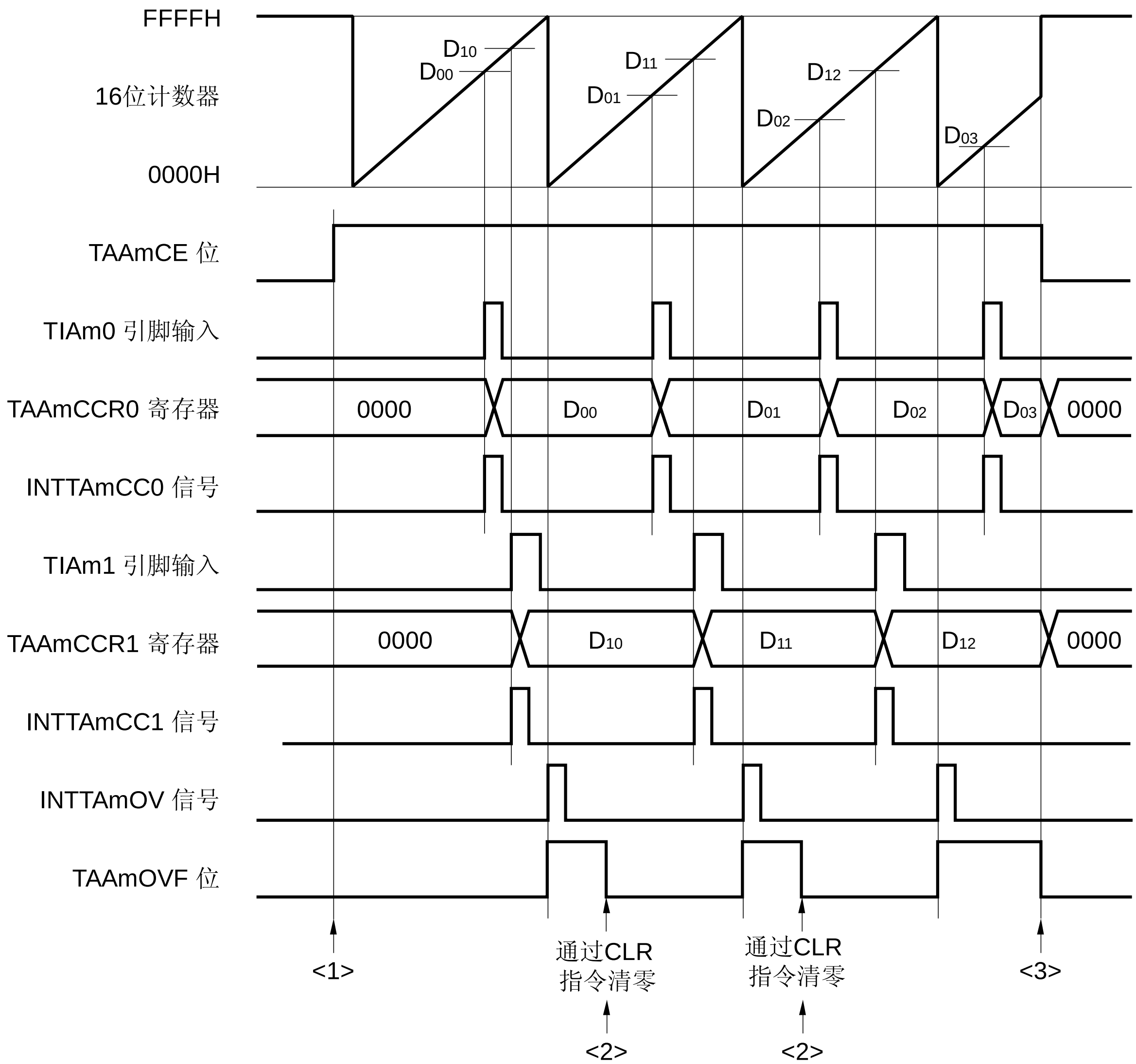
图 6-40. 自由运行定时器模式的软件处理流程（比较功能）（2/2）



备注 V850E/IF3：n = 0 至 4, m = 2, 4
V850E/IG3：n = 0 至 4, m = 2 至 4

（b） 当捕获/比较寄存器用作捕获寄存器时

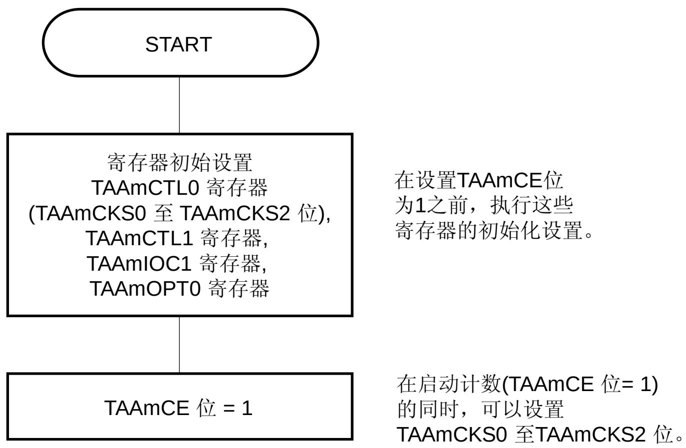
图 6-41. 自由运行定时器模式的软件处理流程（捕获功能）（1/2）



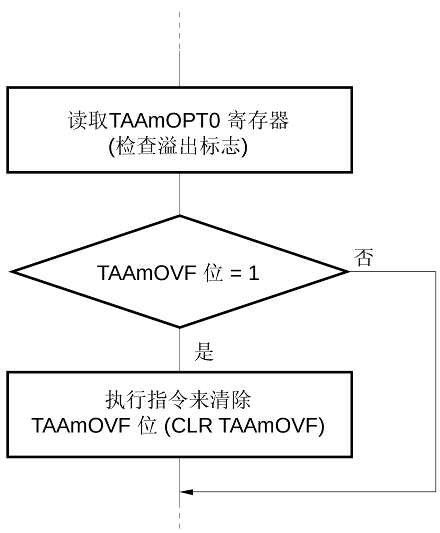
备注 V850E/IF3：m = 2, 4
V850E/IG3：m = 2 至 4

图 **6-41.** 自由运行定时器模式的软件处理流程（捕获功能）（**2/2**）

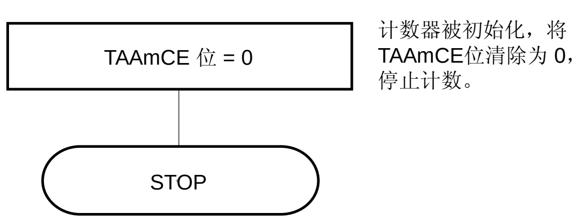
<1> 计数操作启动流程



<2> 溢出标志清除流程



<3> 计数操作停止流程

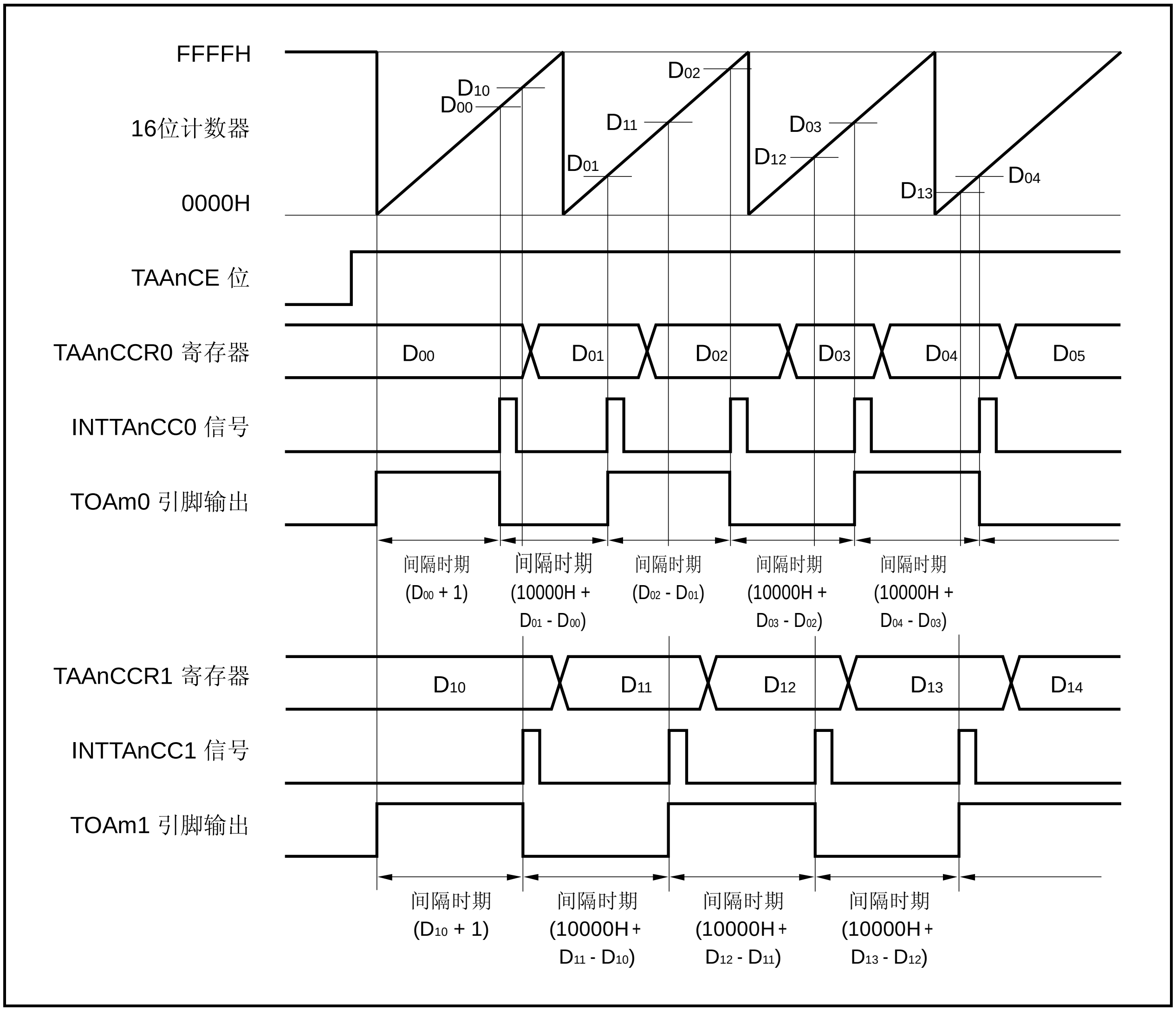


备注　V850E/IF3：m = 2, 4
　　　V850E/IG3：m = 2 至 4

（2）自由运行定时器模式的操作时序

（a） 比较寄存器时的间隔操作

当 16 位定时器/事件计数器 AA 用作间隔定时器，TAAnCCRa 寄存器用作比较寄存器时，在每次检测到 INTTAnCCa 信号后，都需要通过软件处理来设置比较值以产生下一个中断请求信号。



在自由运行定时器模式下进行时间间隔操作时，一个通道可以设置两个时间间隔。

要进行时间间隔操作，当检测到 INTTAnCCa 信号后，必须在中断服务程序中重置对应的 TAAnCCRa 寄存器的值。

重置 TAAnCCRa 寄存器的设置值可由下式计算，其中“D_a”是间隔时间：

比较寄存器默认值： $D_a - 1$

比较寄存器的第二次及其以后的设置值：上次设置值 ＋ D_a

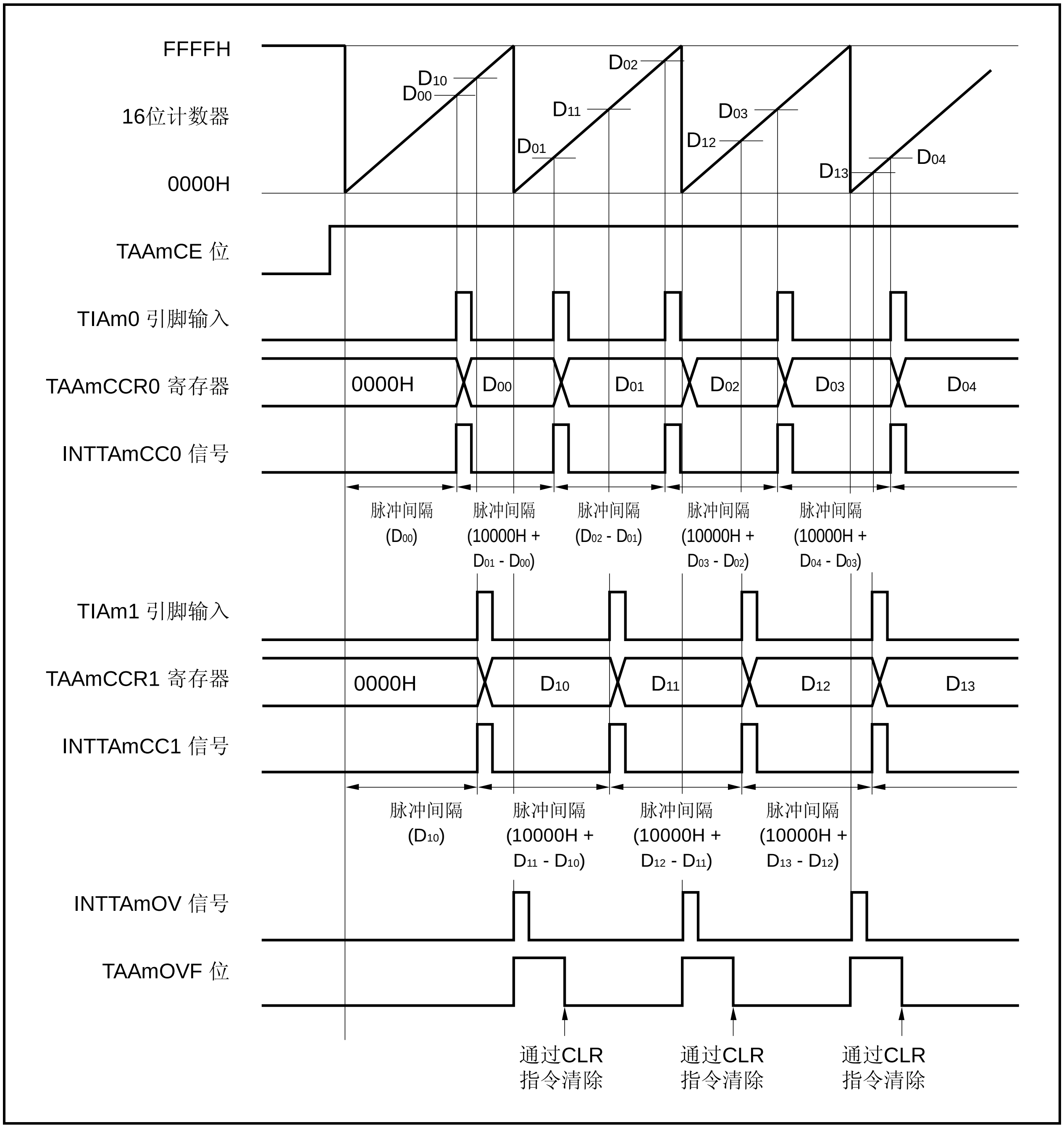
（如果计算结果大于 FFFFH，则从结果中减去 10000H，并将此值设置给寄存器。）

备注 V850E/IF3：n = 0 至 4, m = 2, 4, a = 0, 1

V850E/IG3：n = 0 至 4, m = 2 至 4, a = 0, 1

（b） 用捕获寄存器测量脉冲宽度

在 TAAmCCRa 寄存器用作捕获寄存器进行脉宽测量时，每次检测到 INTTAmCCa 信号时，需要通过软件处理来读取捕获寄存器并计算间隔时间。



在自由运行定时器模式下执行脉宽测量时，可以用一个通道测量两个脉冲宽度。

要测量脉冲宽度，可以与 INTTAmCCa 信号同步读取 TAAmCCRa 寄存器的值，并计算读取值与先前读取值之间的差，得到脉冲宽度。

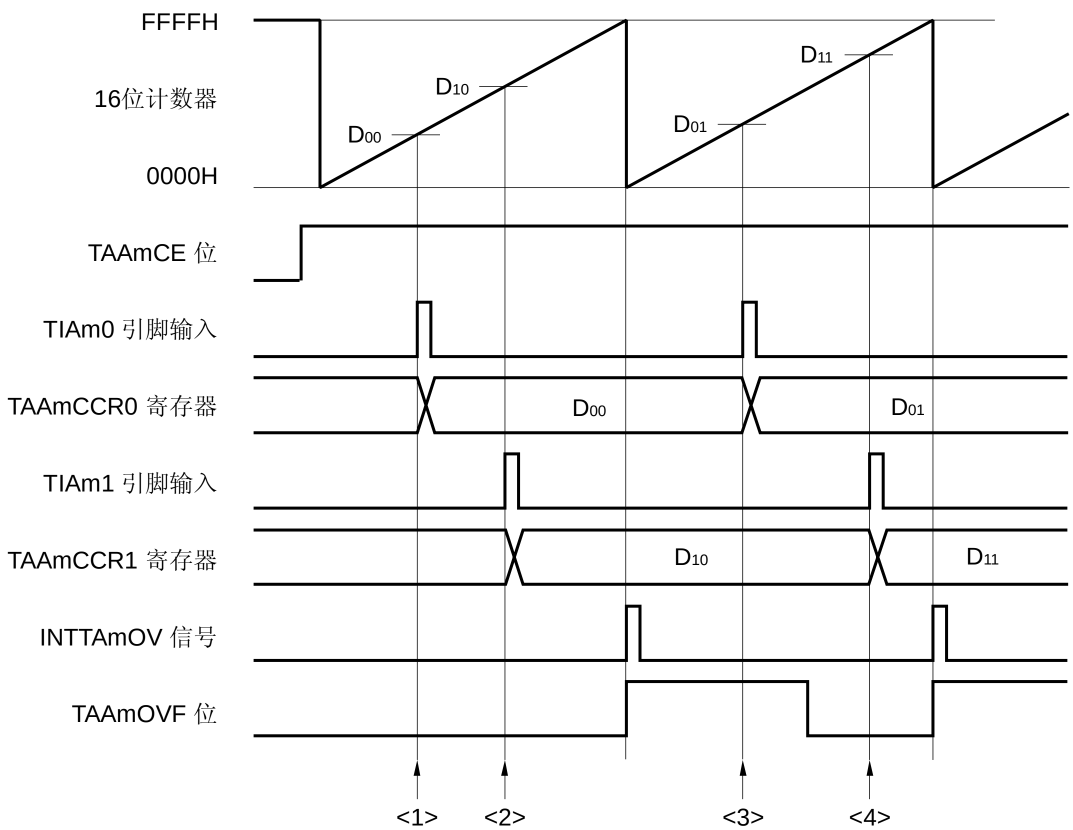
备注 V850E/IF3：m = 2, 4, a = 0, 1

V850E/IG3：m = 2 至 4, a = 0, 1

（c） 当使用两个捕获寄存器时对溢出的处理

当使用两个捕获寄存器时，处理溢出标志必须要格外小心。首先是一个不正确处理的示例，如下所示。

使用两个捕获寄存器时不正确处理的示例



在自由运行定时器模式下测量两个脉冲宽度时，可能出现下列问题。

<1> 读取 TAAmCCR0 寄存器（TIAm0 引脚输入的默认值设置）。
<2> 读取 TAAmCCR1 寄存器（TIAm1 引脚输入的默认值设置）。
<3> 读取 TAAmCCR0 寄存器。
读溢出标志。若溢出标志为 1，将其清为 0。
因为溢出标志为 1，所以可以用式（10000H＋D_{01} – D_{00}）计算脉冲宽度。
<4> 读取 TAAmCCR1 寄存器。
读溢出标志。因为此标志在<3>中被清零，所以，读取到 0。
因为溢出标志为 0，所以可以用（D_{11} – D_{10}）计算脉冲宽度（不正确）。

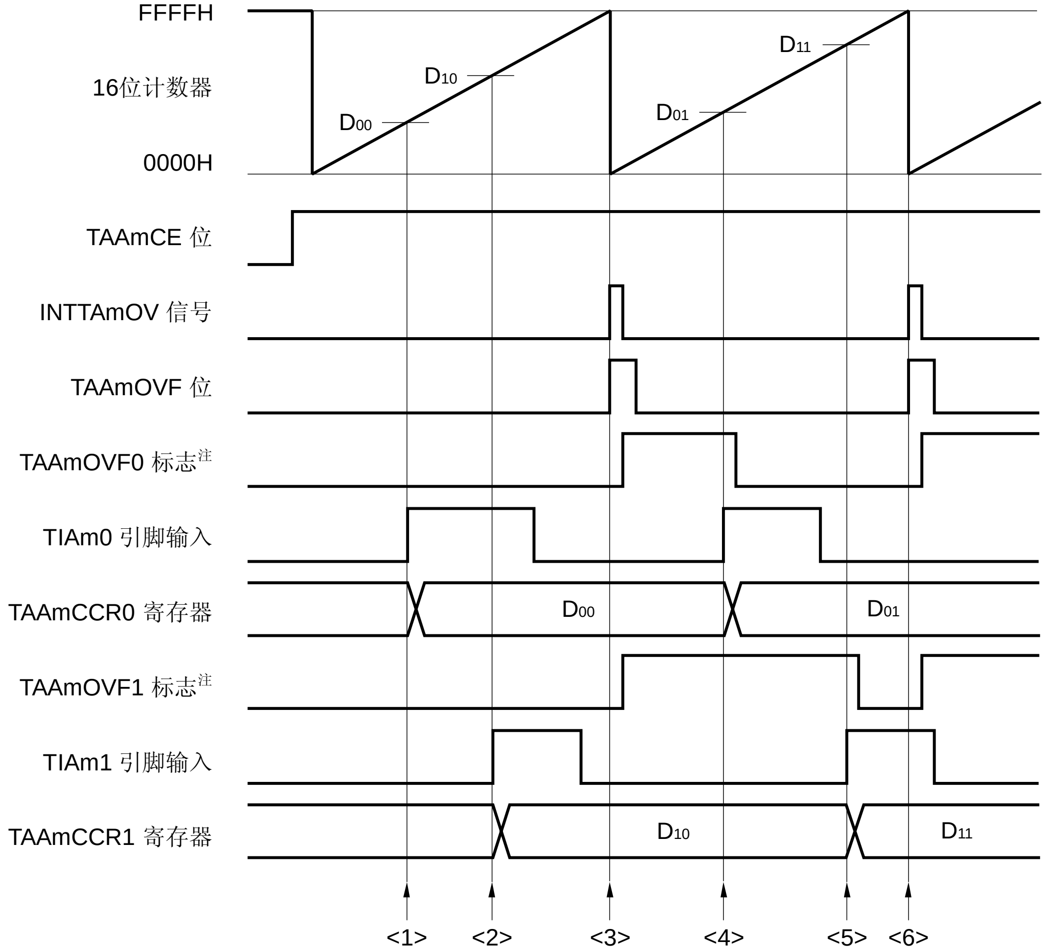
备注 V850E/IF3：m = 2, 4
V850E/IG3：m = 2 至 4

在使用两个捕获寄存器时，如果溢出标志由一个捕获寄存器清为 0，则另一个捕获寄存器可能得不到正确的脉冲宽度。

在使用两个捕获寄存器时，使用软件解决。关于如何使用软件的示例如下。

（1/2）

使用两个捕获寄存器时（使用溢出中断）的示例



注　由软件对内部 RAM 中的 TAAmOVF0 和 TAAmOVF1 标志进行设置。

<1> 读取 TAAnmCCR0 寄存器（TIAm0 引脚输入的默认值设置）。

<2> 读取 TAAmCCR1 寄存器（TIAm1 引脚输入的默认值设置）。

<3> 发生溢出，在溢出中断服务程序中将 TAAmOVF0 和 TAAmOVF1 标志设置为 1，并将溢出标志清为 0。

<4> 读取 TAAmCCR0 寄存器。
读取 TAAmOVF0 标志。如果 TAAmOVF0 标志为 1，则将其清为 0。
因为 TAAmOVF0 标志为 1，所以可以用式（$10000H + D_{01} – D_{00}$）计算脉冲宽度。

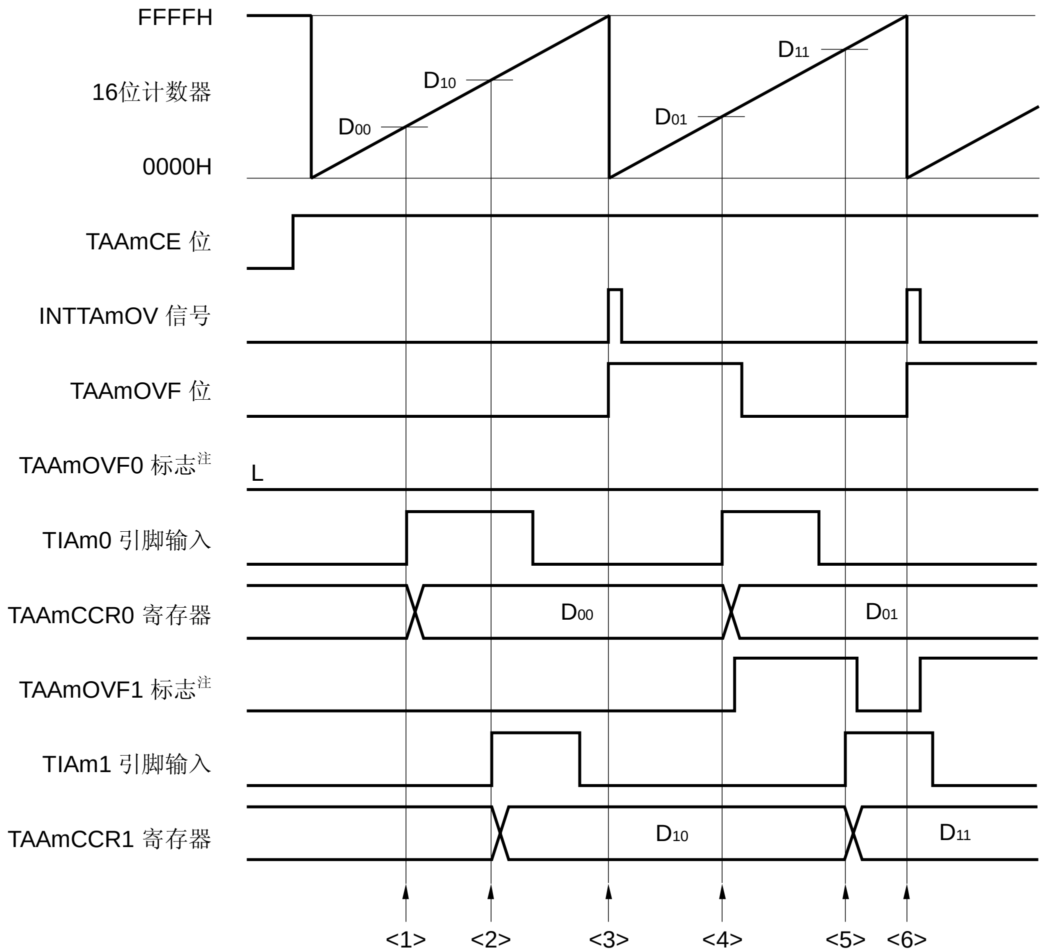
<5> 读取 TAAmCCR1 寄存器。
读取 TAAmOVF1 标志。如果 TAAmOVF1 标志为 1，则将其清为 0（TAAmOVF0 标志在<4>中已清零，而 TAAmOVF1 标志仍然保持为 1）。
因为 TAAmOVF1 标志为 1，所以可以用式（$10000H + D_{11} – D_{10}$）计算脉冲宽度（正确）。

<6> 重复 <3> ……

备注　V850E/IF3：m = 2, 4
V850E/IG3：m = 2 至 4

（2/2）

使用两个捕获寄存器时（不使用溢出中断）的示例



注　由软件对内部 RAM 中的 TAAmOVF0 和 TAAmOVF1 标志进行设置。

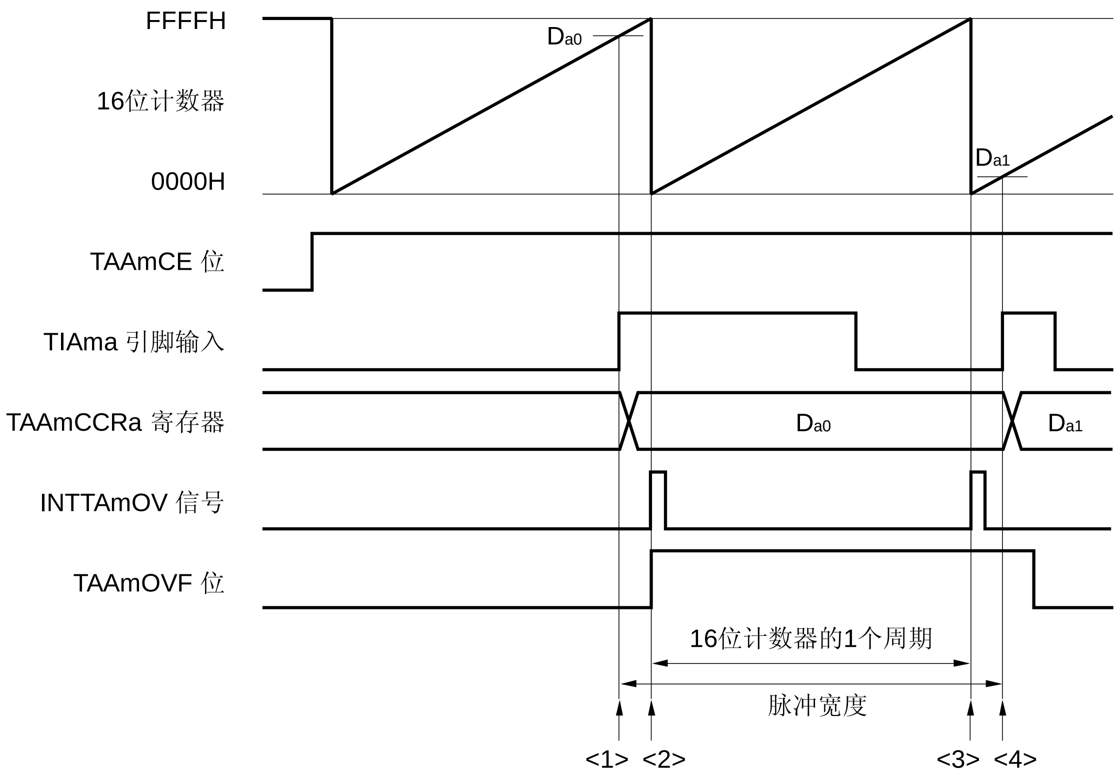
<1> 读取 TAAmCCR0 寄存器（TIAm0 引脚输入的默认值设置）。
<2> 读取 TAAmCCR1 寄存器（TIAm1 引脚输入的默认值设置）。
<3> 发生溢出，软件不作任何处理。
<4> 读取 TAAmCCR0 寄存器。
读溢出标志。若溢出标志为 1，则仅将 TAAmOVF1 标志置为 1，并把溢出标志清为 0。
因为溢出标志为 1，所以可以用式（10000H＋D_{01} – D_{00}）计算脉冲宽度。
<5> 读取 TAAmCCR1 寄存器。
读溢出标志。因为溢出标志在<4>中清零，所以读取到 0。
读取 TAAmOVF1 标志。如果 TAAmOVF1 标志为 1，则将其清为 0。
因为 TAAmOVF1 标志为 1，所以可以用式（10000H＋D_{11} – D_{10}）计算脉冲宽度（正确）。
<6> 重复 <3> ……

备注　V850E/IF3：m = 2, 4
V850E/IG3：m = 2 至 4

（**d**） 当捕获触发间隔较长时的溢出处理

如果脉冲宽度大于 16 位计数器的一个计数循环周期，则必须特别小心，因为从第一个捕获触发信号到下一个触发信号之间可能发生多次溢出。首先是一个不正确处理的示例，如下所示。

捕获触发间隔较长时的不正确处理示例



在自由运行定时器模式下测量长脉冲的宽度时，可能出现下列问题。

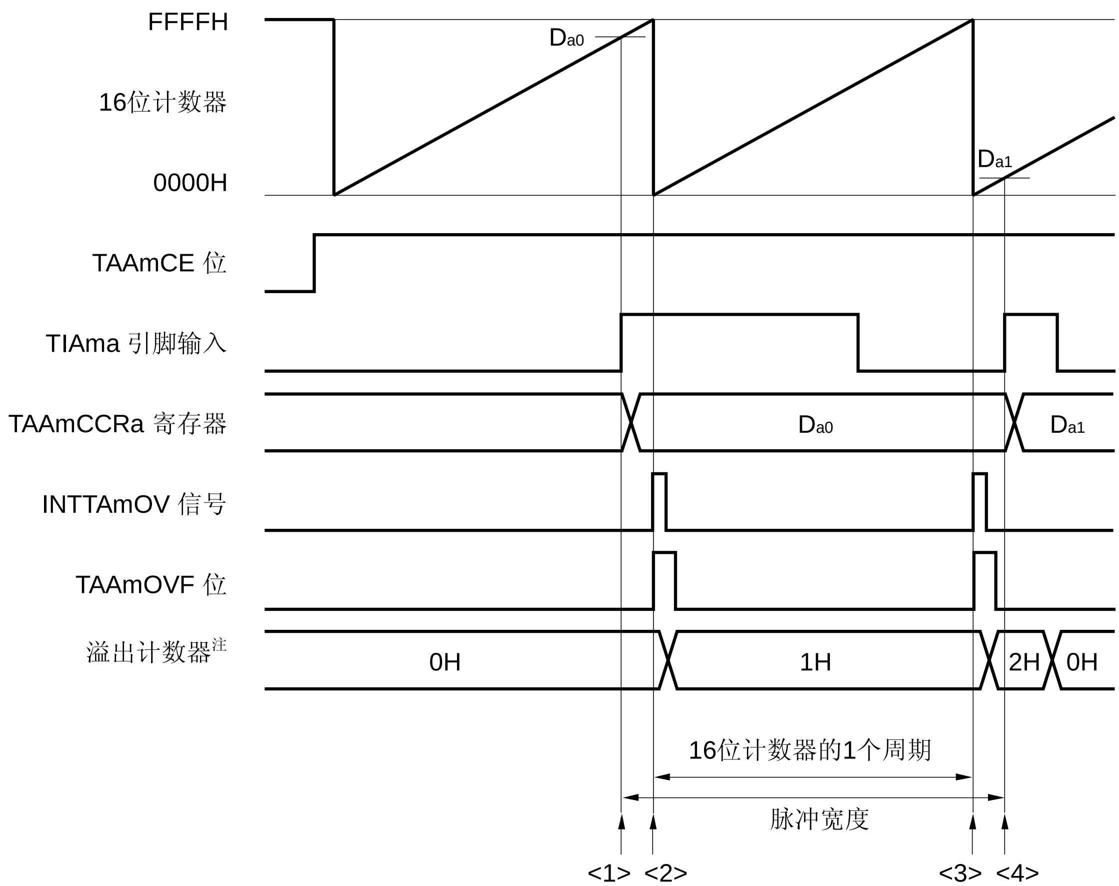
<1> 读取 TAAmCCRa 寄存器（TIAma 引脚输入的默认值设置）。
<2> 发生溢出，软件不作任何处理。
<3> 第二次发生溢出。软件不作任何处理。
<4> 读取 TAAmCCRa 寄存器。
读溢出标志。若溢出标志为 1，将其清为 0。
因为溢出标志为 1，所以可以用式（10000H + D_{a1} – D_{a0}）计算脉冲宽度（不正确）。
事实上，脉冲宽度应为（20000H + D_{a1} – D_{a0}），因为实际已经发生两次溢出。

备注 V850E/IF3：m = 2, 4, a = 0, 1
V850E/IG3：m = 2 至 4, a = 0, 1

如果在捕获触发间隔较长时，发生两次或更多次的溢出，则可能无法获得正确的脉冲宽度。

如果捕获触发间隔较长时，应该减慢计数时钟以延长 16 位计数器的一个循环周期，或者使用软件来解决。关于如何使用软件，示例如下。

捕获触发间隔较长时的示例



注　内部 RAM 中的溢出计数器由软件任意设置。

<1> 读取 TAAmCCRa 寄存器（TIAma 引脚输入的默认值设置）。
<2> 发生溢出，在溢出中断服务程序中增加溢出计数器的值，并将溢出标志清为 0。
<3> 再次发生溢出。在溢出中断服务程序中递增（＋1）溢出计数器的值，并把溢出标志清为 0。
<4> 读取 TAAmCCRa 寄存器。
读取溢出计数器。
→ 当溢出计数器是“N”时，可用式（$N \times 10000H + D_{a1} - D_{a0}$）计算脉冲宽度。
在本例中，脉冲宽度是（$20000H + D_{a1} - D_{a0}$），因为溢出实际发生了两次。
溢出计数器清零（0H）。

备注　V850E/IF3：m = 2, 4, a = 0, 1
V850E/IG3：m = 2 至 4, a = 0, 1

（e）　**清除溢出标志**

当 TAAmOVF 位为 1 时读取该位之后，通过 CLR 指令将 TAAmOVF 位清为 0，且当 TAAmOVF 位为 1 时读取该位之后，将 8 位数据（第 0 位是 0）写入 TAAmOPT0 寄存器，可以将溢出标志清 0。

6.6.7 脉冲宽度测量模式（TAAmMD2 至 TAAmMD0 位 = 110）

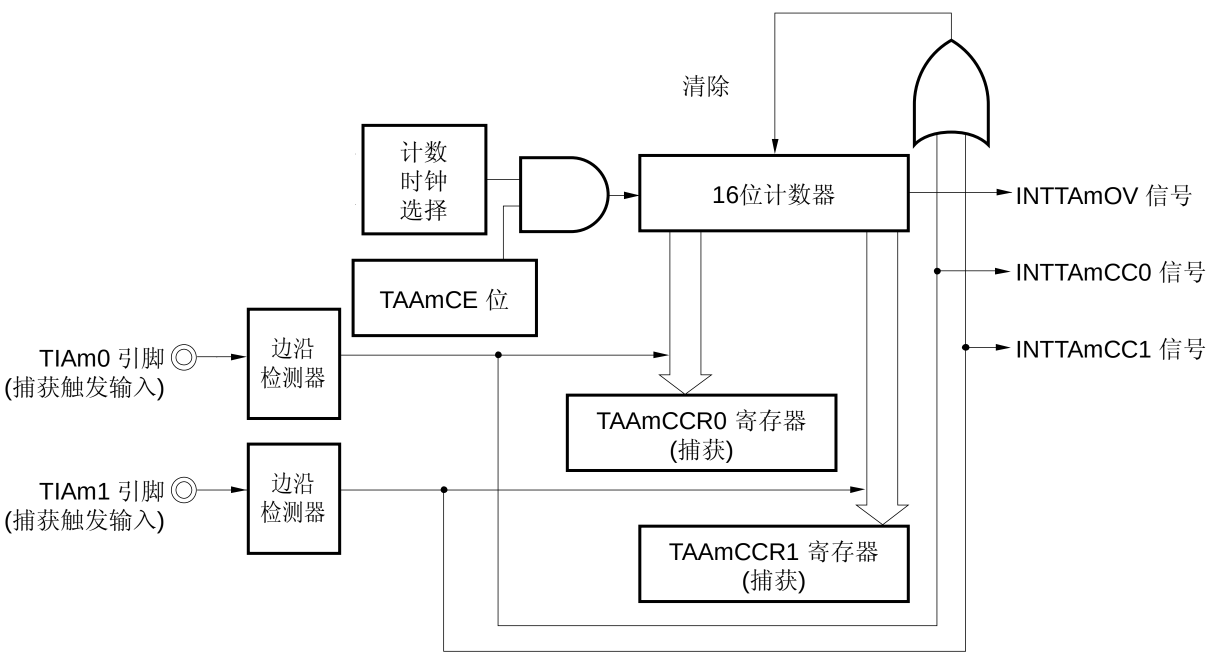
只有在 TAA2, TAA3（仅限 V850E/IG3）和 TAA4 时，该模式有效。

在脉冲宽度测量模式下，当 TAAmCTL0.TAAmCE 位被设置为 1 时，16 位定时器/事件计数器 AA 开始计数。每当检测到有效边沿输入 TIAma 引脚时，16 位计数器的计数值被存储在 TAAmCCRa 寄存器，且 16 位计数器清除为 0000H。

在发生捕获中断请求信号（INTTAmCCa）之后，通过读取 TAAmCCRa 寄存器可以测量有效边沿的间隔。

如图 6-43 所示，选择 TIAm0 或 TIAm1 引脚作为捕获触发输入引脚，并通过 TAAmIOC1 寄存器设置未使用引脚为“无边沿检测”。

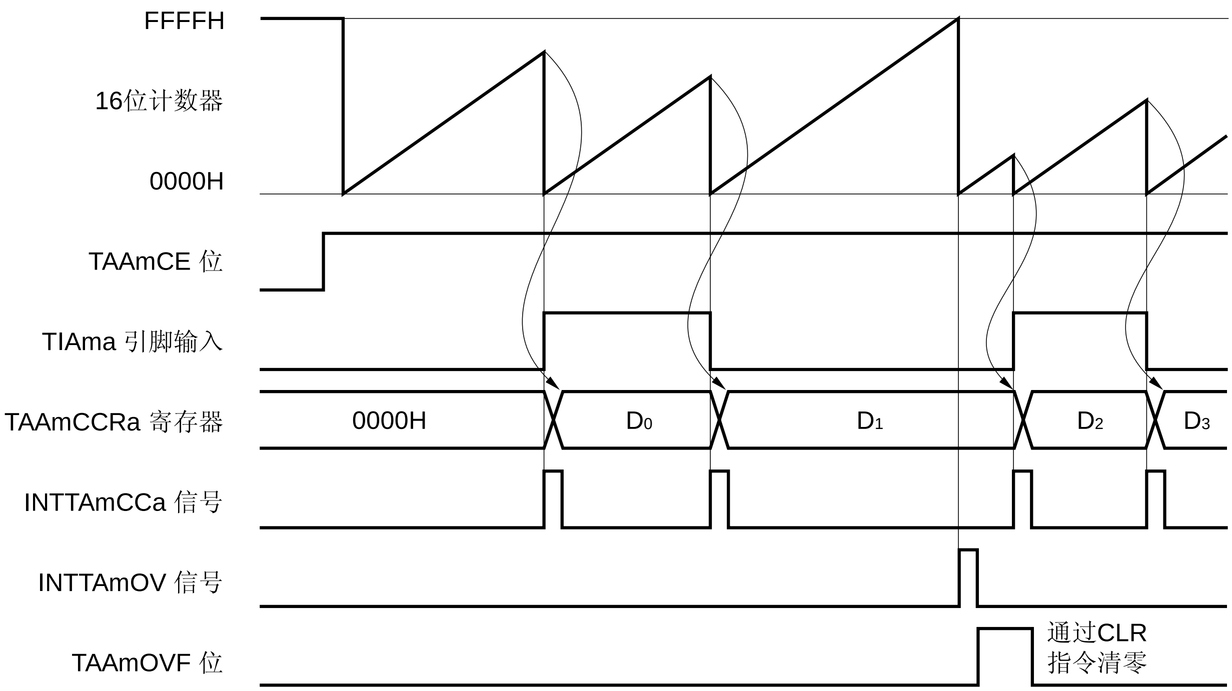
<R> **图 6-42. 脉冲宽度测量模式的配置**



注意事项 **在脉冲宽度测量模式下，选择内部时钟作为计数时钟（将 TAAmCTL1.TAAmEEE 位清除为 0）。**

备注 V850E/IF3：m = 2, 4, a = 0, 1
V850E/IG3：m = 2 至 4, a = 0, 1

图 6-43. 脉冲宽度测量模式的基本时序



备注 V850E/IF3：m = 2, 4, a = 0, 1
V850E/IG3：m = 2 至 4, a = 0, 1

当 TAAmCE 位置为 1 时，16 位计数器开始计数。当检测到有效边沿输入 TIAma 引脚时，16 位计数器的计数值存储到 TAAmCCRa 寄存器中，16 位计数器清除为 0000H，并产生一个捕获中断请求信号（INTTAmCCa）。

脉冲宽度的计算如下。

脉冲宽度＝捕获值 × 计数时钟周期

当 16 位计数器计数到 FFFFH 时，即使没有有效边沿输入 TIAma 引脚，在下一个计数时钟时，仍然会产生溢出中断请求信号（INTTAmOV），且计数器清除为 0000H，然后计数器继续计数。此时，溢出标志（TAAmOPT0.TAAmOVF 位）也置为 1。确认溢出标志位被置为 1 然后通过软件运行 CLR 指令将其清除为 0。

若溢出标志设置为 1，则脉冲宽度的计算如下。

脉冲宽度＝（10000H × TAAmOVF 位置位（1）的次数＋捕获值）× 计数时钟周期

备注 V850E/IF3：m = 2, 4, a = 0, 1
V850E/IG3：m = 2 至 4, a = 0, 1

图 6-44. 脉冲宽度测量模式的寄存器设置（1/2）

（a） **TAAm 控制寄存器 0（TAAmCTL0）**

<R>

	TAAmCE					TAAmCKS2	TAAmCKS1	TAAmCKS0
TAAmCTL0	0/1	0	0	0	0	0/1	0/1	0/1

TAAmCKS2 to TAAmCKS0: 选择计数时钟

TAAmCE:
0: 停止计数
1: 使能计数

（b） **TAAm 控制寄存器 1（TAAmCTL1）**

<R>

	TAAaSYE	TAAmEST	TAAmEEE			TAAmMD2	TAAmMD1	TAAmMD0
TAAmCTL1	0	0	0	0	0	1	1	0

TAAmMD2 to TAAmMD0:
1, 1, 0:
脉冲宽度测量模式

TAAmEEE:
0: 计数时钟由TAAmCKS0
至 TAAmCKS2 位选择

（c） **TAAm I/O 控制寄存器 1（TAAmIOC1）**

<R>

					TAAmIS3	TAAmIS2	TAAmIS1	TAAmIS0
TAAmIOC1	0	0	0	0	0/1	0/1	0/1	0/1

TAAmIS1, TAAmIS0: 选择TIAm0 引脚输入
的有效边沿

TAAmIS3, TAAmIS2: 选择TIAm1 引脚输入
的有效边沿

图 6-44. 脉冲宽度测量模式的寄存器设置（2/2）

（d） TAAm 选项寄存器 0（TAAmOPT0）

			TAAmCCS1	TAAmCCS0				TAAmOVF
TAAmOPT0	0	0	0	0	0	0	0	0/1

TAAmOVF：溢出标志

（e） TAAm 计数器读取缓冲寄存器（TAAmCNT）

通过读取 TAAmCNT 寄存器，可以读出 16 位计数器的值。

（f） TAAm 捕获/比较寄存器 0 和 1（TAAmCCR0 和 TAAmCCR1）

当检测到有效边沿输入 TIAm0 和 TIAm1 引脚时，这些寄存器保存 16 位计数器的计数值。

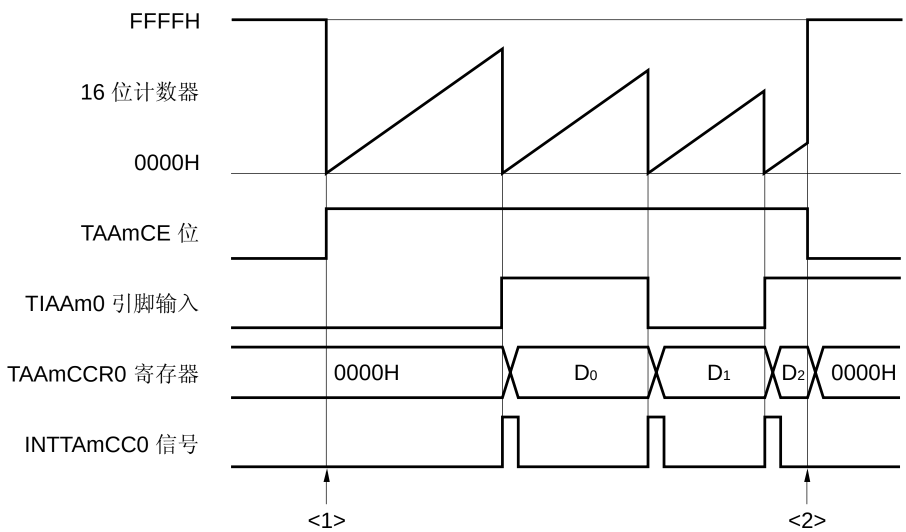
<R> **备注** **1.** 脉冲宽度测量模式不使用 TAAm I/O 控制寄存器 0（TAAmIOC0）和 TAAm I/O 控制寄存器 2（TAAmIOC2）。

2. V850E/IF3：m = 2, 4

V850E/IG3：m = 2 至 4

（1） 脉冲宽度测量模式的操作流程

图 6-45. 脉冲宽度测量模式的软件处理流程



<R>

<1> 计数操作启动流程

启动

寄存器初始设置
TAAmCTL0 寄存器
(TAAmCKS0 至 TAAmCKS2 位),
TAAmCTL1 寄存器,
TAAmIOC1 寄存器,
TAAmOPT0 寄存器

在设置TAAmCE位为1之前，执行这些寄存器的初始化设置。

TAAmCE 位 = 1

在启动计数(TAAmCE 位= 1)的同时，可以设置TAAmCKS0 至TAAmCKS2 位。

<2> 计数操作停止流程

TAAmCE 位 = 0

计数器被初始化，将TAAmCE位清除为 0，停止计数。

停止

备注 V850E/IF3：m = 2, 4
V850E/IG3：m = 2 至 4

（2） 脉冲宽度测量模式的操作时序

（a） 清除溢出标志

当 TAAmOVF 位为 1 时读取该位之后，通过 CLR 指令将 TAAmOVF 位清为 0，且当 TAAmOVF 位为 1 时读取该位之后，将 8 位数据（第 0 位是 0）写入 TAAmOPT0 寄存器，可以将溢出标志清 0。

第七章　16 位定时器/事件计数器 AB（TAB）

定时器 AB（TAB）是 16 位定时器/事件计数器。
V850E/IF3 和 V850E/IG3 单片机内置 TAB0 和 TAB1。

7.1　概述

定时器 TABn 的概况如下所示(n = 0, 1)。

• 时钟选择：	8 路
• 捕获/触发输入引脚：	4
• 外部事件计数输入引脚：	1
• 外部触发输入引脚：	1
• 定时器/计数器：	1
• 捕获/比较寄存器：	4
• 捕获/比较一致中断请求信号：	4
• 溢出中断请求信号：	1
• 定时器输出引脚[注]：	4

注　这是 TABn 的输出引脚的数量；不包括 TMQOPn 的输出引脚。TMQOPn 输出引脚的详情，参见 **第十章　电机控制功能。**

7.2　功能

定时器 TABn 具有下列功能(n = 0, 1)。

- 6 相PWM输出[注]
- 间隔定时器
- 外部事件计数器
- 外部触发脉冲输出
- 单次触发脉冲输出
- PWM 输出
- 自由运行定时器
- 脉冲宽度测量

注　连接到 TMQOPn。详情参见 **第十章　电机控制功能。**

7.3　配置

TABn 包括下列硬件：
(n = 0, 1)

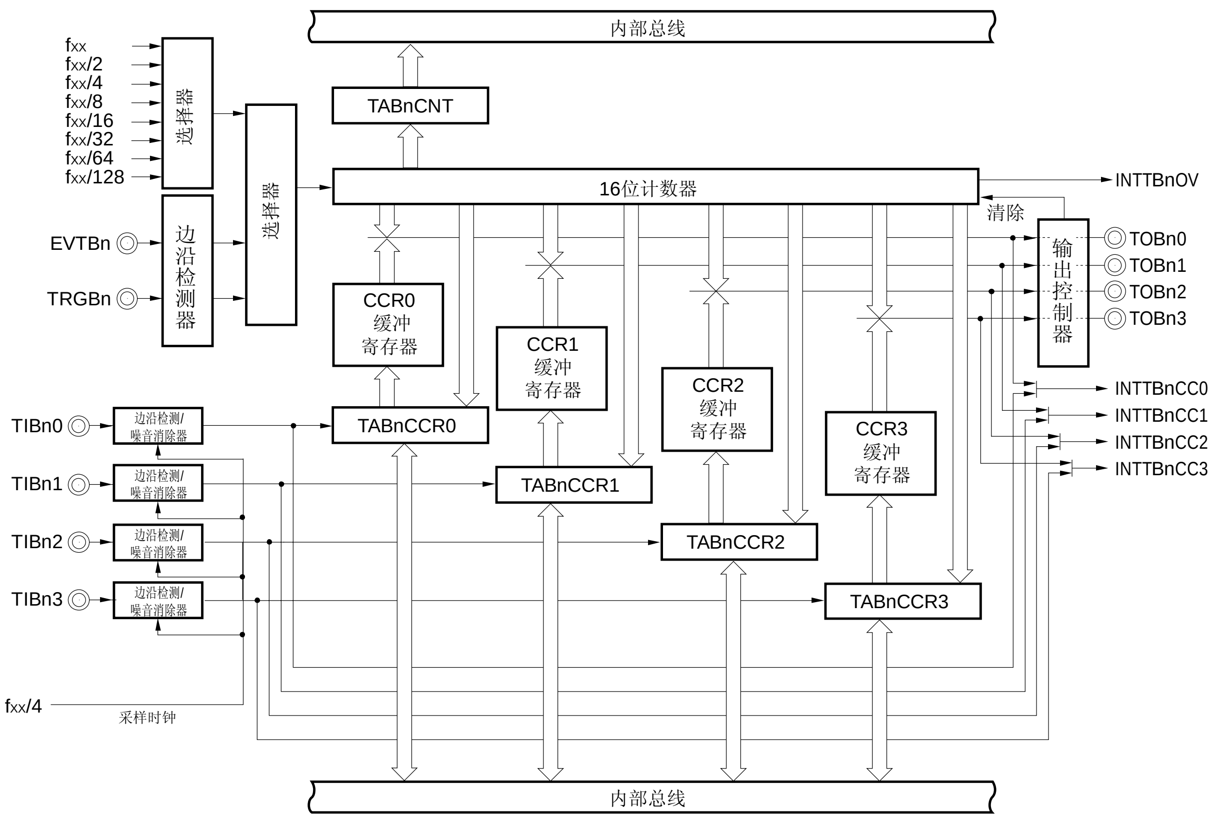
表 7-1. TABn 的配置

项目	配置
定时器寄存器	16 位计数器 × 1
寄存器	TABn 计数器读取缓冲寄存器(TABnCNT)：总共 2 个 TABn 捕获/比较寄存器 0 至 3 (TABnCCR0 至 TABnCCR3)：总共 8 个 CCR0 至 CCR3 缓冲寄存器：总共 8 个
定时器输入	总共 12 个(TIB00 至TIB03，TIB10 至TIB13, EVTB0, EVTB1, TRGB0, TRGB1 引脚)[注]
定时器输出	总共 8 个 (TOB00 至TOB03, TOB10 至TOB13 引脚)[注]
控制寄存器	TABn 控制寄存器 0, 1 (TABnCTL0, TABnCTL1) TABn I/O 控制寄存器 0 至 2 (TABnIOC0 至 TABnIOC2) TABn 选项寄存器 0 (TABnOPT0)

注　TIBn1 至 TIBn3 引脚功能复用作为定时器输出引脚 (TOBn1 至 TOBn3)。

备注　n = 0, 1

图 7-1. TABn 框图



备注　**1.** fxx：外围时钟
2. 关于噪声消除器，参见 **4.6 噪声消除器**。
3. n = 0, 1

（1）　16 位计数器

此 16 位计数器可以计数内部时钟或外部事件。

该计数器的计数值可以通过读取 TABnCNT 寄存器得到。

当 TABnCTL0.TABnCE 位＝ 0 时，16 位计数器的值为 FFFFH。如果此时读取 TABnCNT 寄存器，则读回的值为 0000H 。

系统复位将 TABnCE 位清除为 0。

（2）　CCR0 缓冲寄存器

CCR0 缓冲寄存器是一个 16 位比较寄存器，用于比较 16 位计数器的计数值。

当 TABnCCR0 寄存器用作比较寄存器时，写入 TABnCCR0 寄存器的值会传输到 CCR0 缓冲寄存器。当 16 位计数器的计数值与 CCR0 缓冲寄存器的值相一致时，就会产生一个比较一致中断请求信号（INTTBnCC0）。

CCR0 缓冲寄存器不能直接读写。

CCR0 缓冲寄存器在复位后清零为 0000H，TABnCCR0 寄存器清零为 0000H。

（3）　CCR1 缓冲寄存器

CCR1 缓冲寄存器是一个 16 位比较寄存器，用于比较 16 位计数器的计数值。

当 TABnCCR1 寄存器用作比较寄存器时，写入 TABnCCR1 寄存器的值会传输到 CCR1 缓冲寄存器。当 16 位计数器的计数值和 CCR1 缓冲寄存器的值一致时，就会产生一个比较一致中断请求信号（INTTBnCC1）。

CCR1 缓冲寄存器不能直接读写。

CCR1 缓冲寄存器在复位后清零为 0000H，TABnCCR1 寄存器清零为 0000H。

（4）　CCR2 缓冲寄存器

CCR2 缓冲寄存器是一个 16 位比较寄存器，用于比较 16 位计数器的计数值。

当 TABnCCR2 寄存器用作比较寄存器时，写入 TABnCCR2 寄存器的值会传输到 CCR2 缓冲寄存器。当 16 位计数器的计数值和 CCR2 缓冲寄存器的值一致时，就会产生一个比较一致中断请求信号（INTTBnCC2）。

CCR2 缓冲寄存器不能直接读写。

CCR2 缓冲寄存器在复位后清零为 0000H，TABnCCR2 寄存器清零为 0000H。

（5）　CCR3 缓冲寄存器

CCR3 缓冲寄存器是一个 16 位比较寄存器，用于比较 16 位计数器的计数值。

当 TABnCCR3 寄存器用作比较寄存器时，写入 TABnCCR3 寄存器的值会传输到 CCR3 缓冲寄存器。当 16 位计数器的计数值与 CCR3 缓冲寄存器的值相一致时，就会产生一个比较一致中断请求信号（INTTBnCC3）。

CCR3 缓冲寄存器不能直接读写。

CCR3 缓冲寄存器在复位后清零为 0000H，TABnCCR3 寄存器清零为 0000H。

（6）　边沿检测器

该电路检测输入到 TIBn0 至 TIBn3、EVTBn 和 TRGBn 引脚信号的有效边沿。通过设置 TABnIOC1 和 TABnIOC2 寄存器，可以选择各种有效沿：无边沿、上升沿、下降沿、上升/下降双边沿作为有效沿。

（7）　输出控制器

该电路控制 TOBn0 至 TOBn3 引脚的输出。输出控制器由 TABnIOC0 寄存器控制。

（8）　选择器

此选择器选择 16 位计数器的计数时钟。可选择 8 种内部时钟或一个外部事件作为计数时钟。

7.4　寄存器

（1）　TABn 控制寄存器 0（TABnCTL0）

TABnCTL0 寄存器是 8 位寄存器，用于控制定时器 TABn 的操作。

该寄存器可以按字节读写或按位读写。

复位输入将该寄存器设为 00H。

随时可以通过软件将相同的值写入 TABnCTL0 寄存器。

复位后：00H　R/W　地址：　TAB0CTL0 FFFFF5E0H, TAB1CTL0 FFFFF620H

	<7>	6	5	4	3	2	1	0
TABnCTL0 (n = 0, 1)	TABnCE	0	0	0	0	TABnCKS2	TABnCKS1	TABnCKS0

TABnCE	TABn 操作控制
0	TABn 操作禁止 (TABn 异步复位[注])
1	TABn 操作使能

TABnCKS2	TABnCKS1	TABnCKS0	内部计数时钟选择
0	0	0	fxx
0	0	1	fxx/2
0	1	0	fxx/4
0	1	1	fxx/8
1	0	0	fxx/16
1	0	1	fxx/32
1	1	0	fxx/64
1	1	1	fxx/128

注　TABnOPT0.TABnOVF 位和 16 位计数器被同时复位。并且，16 位计数器被复位的同时，定时器输出(TOBn0 至 TOBn3 引脚)被复位到 TABnIOC0 寄存器设置的状态。

注意事项　1. 当 TABnCE 位＝0 时，设置 TABnCKS2 至 TABnCKS0 位为“1”。
当 TABnCE 位的值从 0 变为 1 时，可以同时设定 TABnCKS2 至 TABnCKS0 位。
2. 请确保将第 3 位至第 6 位清为“0”。

备注　fxx：外围时钟

（2） TABn 控制寄存器 1（TABnCTL1）

TABnCTL1 寄存器是 8 位寄存器，用于控制定时器 TABn 的操作。

该寄存器可以按字节或按位进行读取和写入。

复位输入将该寄存器设为 00H。

复位后： 00H R/W 地址： TAB0CTL1 FFFFF5E1H, TAB1CTL1 FFFFF621H

	7	6	5	4	3	2	1	0
TABnCTL1 (n = 0, 1)	0	TABnEST	TABnEEE	0	0	TABnMD2	TABnMD1	TABnMD0

TABnEST	软件触发控制
0	-
1	为外部触发输入产生一个有效信号 • 在单次触发脉冲输出模式下： 向 TABnEST 位写入1作为触发信号，输出一个单次触发脉冲。 • 在外部触发脉冲输出模式下： 向 TABnEST 位写入1作为触发信号，输出一个PWM波形。
TABnEST 位的读取值始终为 0.	

TABnEEE	计数时钟选择
0	禁止外部事件计数输入(EVTBn 引脚)的操作. (执行对计数时钟的计数，该时钟由 TABnCTL0.TABnCKS0 至 TABnCKS2 位选择.)
1	使能外部事件计数输入(EVTBn 引脚)的操作 (对外部事件计数输入信号(EVTBn 引脚)的每个有效边沿执行计数)
通过 TABnEEE 位选择使用内部计数时钟计数， 还是使用外部事件计数输入的有效边沿来计数。	

TABnMD2	TABnMD1	TABnMD0	定时器模式选择
0	0	0	间隔定时器模式
0	0	1	外部事件计数模式
0	1	0	外部触发脉冲输出模式
0	1	1	单次触发脉冲输出模式
1	0	0	PWM 输出模式
1	0	1	自由运行定时器模式
1	1	0	脉宽测量模式
1	1	1	禁止设置

注 仅使用 TABn 时，不能使用 6 相 PWM 输出模式。详情参见 **第十章 电机控制功能**。

注意事项 1. 只有在外触发脉冲输出模式或单次触发脉冲输出模式下，TABnEST 位有效。在其它模式下，向该位写入 1 的操作都被忽略。

2. 不论 TABnEEE 位的取值如何，在外部事件计数模式下，都会选择外部事件输入。

3. TABnCTL0.TABnCE 位＝ 0 时，设置 TABnEEE 和 TABnMD2 至 TABnMD0 位（TABnCE 位＝ 1 时，可以写入相同的值）。在 TABnCE 位＝ 1 时，重写操作是无法保证的，若重写发生错误，应把 TABnCE 位清除为 0，再重新对这些位进行设定。

4. 请确保将第 3 位、第 4 位和第 7 位清除为“0”。

（3）　TABn 输入/输出控制寄存器 0（TABnIOC0）

TABnIOC0 寄存器是 8 位寄存器，用于控制定时器的输出（TOBn0 至 TOBn3，TOBnT1 至 TOBnT3 引脚）。
该寄存器可以按字节或按位进行读取和写入。
复位输入将该寄存器设为 00H。

复位后：00H　　R/W　　地址：　　TAB0IOC0 FFFFF5E2H, TAB1IOC0 FFFFF622H

	7	<6>	5	<4>	3	<2>	1	<0>
TABnIOC0	TABnOL3	TABnOE3	TABnOL2	TABnOE2	TABnOL1	TABnOE1	TABnOL0	TABnOE0

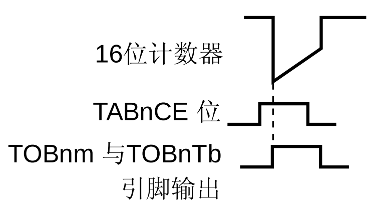
（n = 0, 1
b = 1 至 3）

TABnOLm	TOBnm, TOBnTb 引脚输出电平设置[注] (m = 0 至 3)
0	TOBnm 与 TOBnTb 引脚输出从高电平开始
1	TOBnm 与 TOBnTb 引脚输出从低电平开始

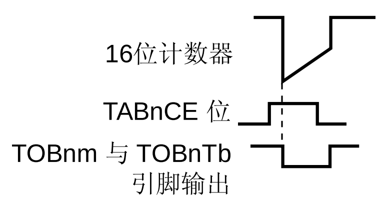
TABnOEm	TOBnm, TOBnTb 引脚输出设置 (m = 0 至 3)
0 1	定时器输出禁止 • 当 TABnOLm 位 = 0: 从TOBnm 与 TOBnTb 引脚输出低电平 • 当 TABnOLm 位 = 1: 从 TOBnm 与TOBnTb 引脚输出高电平
	定时器输出使能 (从TOBnm 与 TOBnTb 引脚输出一个单次脉冲).

注　　由 **TABnOLm** 位指定的定时器输出引脚（TOBnm 和 TOBnTb）的输出电平如下所示。

• 当 TABnOLm 位 = 0



• 当 TABnOLm 位= 1



注意事项 **1.** 当 **TOBnm** 和 **TOBnTb** 被设置为输出模式时，如果 **TABnIOC0** 寄存器的设置被改变，引脚的输出改变。设置端口为输入模式并使端口变为高阻抗状态，引脚状态不改变。

2. 当 **TABnCTL0.TABnCE** 位＝**0** 时，重新写入 **TABnOLm** 和 **TABnOEm** 位（当 **TABnCE** 位＝**1** 时，可以写入相同值）。如果重写被误执行，将 **TABnCE** 位清为 **0**，然后重新设定该位。

3. 即使在 **TABnCE** 和 **TABnOEm** 位均为 **0** 时对 **TABnOLm** 位进行操作，**TOBnm** 和 **TOBnTb** 引脚的输出电平也会改变。

4. 在 **6** 相 **PWM** 输出模式下，要产生 **TOBnTb** 引脚输出和 **A/D** 转换器 **0** 和 **1** 的 **A/D** 转换器启动触发信号，请确保通过 **TABnIOC0** 寄存器设置 **TOBnTb** 引脚输出。此时，请确保将 **TABnOL0** 位清除为 **0** 并将 **TABnOE0** 位设置为 **1(b = 1** 至 **3)**。

备注　　**m = 0** 至 **3**

（4） TABn 输入/输出控制寄存器 1（TABnIOC1）

TABnIOC1 寄存器是 8 位寄存器，用于控制捕获触发输入信号（TIBn0 至 TIBn3 引脚）的有效边沿。
该寄存器可以按字节或按位进行读取和写入。
复位输入将该寄存器设为 00H。

复位后： 00H R/W 地址： TAB0IOC1 FFFFF5E3H, TAB1IOC1 FFFFF623H

	7	6	5	4	3	2	1	0
TABnIOC1 (n = 0, 1)	TABnIS7	TABnIS6	TABnIS5	TABnIS4	TABnIS3	TABnIS2	TABnIS1	TABnIS0

TABnIS7	TABnIS6	捕获触发输入信号 (TIBn3 引脚) 有效边沿设置
0	0	无边沿检测 (捕获操作无效)
0	1	检测上升沿
1	0	检测下降沿
1	1	检测双沿

TABnIS5	TABnIS4	捕获触发输入信号 (TIBn2 引脚) 有效边沿设置
0	0	无边沿检测 (捕获操作无效)
0	1	检测上升沿
1	0	检测下降沿
1	1	检测双沿

TABnIS3	TABnIS2	捕获触发输入信号 (TIBn1 引脚) 有效边沿设置
0	0	无边沿检测 (捕获操作无效)
0	1	检测上升沿
1	0	检测下降沿
1	1	检测双沿

TABnIS1	TABnIS0	捕获触发输入信号 (TIBn0 引脚) 有效边沿设置
0	0	无边沿检测 (捕获操作无效)
0	1	检测上升沿
1	0	检测下降沿
1	1	检测双沿

注意事项 **1.** 当 **TABnCTL0.TABnCE** 位＝ **0** 时，重写 **TABnIS7** 至 **TABnIS0** 位（当 **TABnCE** 位＝**1** 时，可以写入相同值）。如果重写不能正确执行，将 **TABnCE** 位清为 **0**，然后重新进行位的设定。

2. 只有在自由运行定时器模式(仅当 **TABnOPT0.TABnCCSm** 位 **= 1)**和脉冲宽度测量模式(**m = 0** 至 **3)**下，**TABnIS7** 至 **TABnIS0** 位才有效。在所有其它模式下，无法进行捕获操作。

（5）　TABn 输入/输出控制寄存器 2（TABnIOC2）

TABnIOC2 寄存器是 8 位寄存器，用于控制外部事件计数器输入信号（EVTBn 引脚）和外部触发输入信号（TRGBn）的有效边沿。

该寄存器可以按字节或按位进行读取和写入。

复位输入将该寄存器设为 00H。

复位后：00H　R/W　地址：　TAB0IOC2 FFFFF5E4H, TAB1IOC2 FFFFF624H

	7	6	5	4	3	2	1	0
TABnIOC2 (n = 0 , 1)	0	0	0	0	TABnEES1	TABnEES0	TABnETS1	TABnETS0

TABnEES1	TABnEES0	外部事件计数输入信号 (EVTBn 引脚) 有效边沿设置
0	0	无边沿检测 (外部事件计数无效)
0	1	检测上升沿
1	0	检测下降沿
1	1	检测双沿

TABnETS1	TABnETS0	外部触发输入信号 (TRGBn 引脚) 有效边沿设置
0	0	无边沿检测 (外部触发无效)
0	1	检测上升沿
1	0	检测下降沿
1	1	检测双沿

注意事项 **1.** 当 **TABnCTL0.TABnCE** 位＝ **0** 时，重新写入 **TABnEES1**，**TABnEES0**，**TABnETS1** 和 **TABnETS0** 位（当 **TABnCE** 位＝ **1**，可以写入相同的值）。如果重新写入不能正确执行，将 **TABnCE** 位清为 **0**，然后重新进行位的设定。

2. 只有当 **TABnCTL1.TABnEEE** 位＝ **1** 或设置为外部事件计数模式（**TABnCTL1.TABnMD2** 至 **TABnCTL1.TABnMD0**＝**001**）时，**TABnEES1** 和 **TABnEES0** 位有效。

3. 只有在外部触发脉冲输出模式或在单次触发脉冲输出模式下，**TABnETS1** 和 **TABnETS0** 位有效。

（6） TABn 选项寄存器 0（TABnOPT0）

TABnOPT0 寄存器为 8 位寄存器，用来设置捕获/比较操作和检测计数器溢出。

该寄存器可以按字节或按位进行读取和写入。

复位输入将该寄存器设为 00H。

复位后：00H R/W 地址： TAB0OPT0 FFFFF5E5H, TAB1OPT0 FFFFF625H

	<7>	<6>	<5>	<4>	3	<2>	<1>	<0>
TABnOPT0	TABnCCS3	TABnCCS2	TABnCCS1	TABnCCS0	0	TABnCMS[注]	TABnCUF[注]	TABnOVF

(n = 0, 1)

TABnCCSm	TABnCCRm 寄存器捕获/比较选择 (m = 0 至 3)
0	选择比较寄存器
1	选择捕获寄存器 (通过TABnCTL0.TABnCE 位 = 0清除)
只有在自由运行定时器模式下，TABnCCSm位的设置有效。	

TABnOVF	TABn 溢出标志
置位 (1)	发生溢出
重置 (0)	TABnOVF 位写入 0或TABnCTL0.TABnCE 位 = 0

- 在自由运行定时器模式或脉宽测量模式下，当16 位计数器的计数值从FFFFH 溢出到 0000H时， TABnOVF位被置为 1 。
- TABnOVF 位被置 1 的同时，产生一个中断请求信号 (INTTBnOV) 。在自由运行定时器模式和脉宽测量模式之外的模式下，不产生INTTBnOV 信号。
- 当TABnOVF位 = 1时，即使TABnOVF位或TABnOPT0 寄存器被读取，TABnOVF位也不会被清除为0。
- 产生INTTBnOV信号之后，在将TABnOVF位清除为0之前，请确认 (通过读取) TABnOVF位被设置为1。
- TABnOVF位可读可写，但是不能通过软件将TABnOVF位置1。在TABn工作期间写入1 没有影响。

注 关于 **TABnCMS** 和 **TABnCUF** 位的详情，参见第十章 电机控制功能。

注意事项 1. 当 TABnCE 位= 0 时，重新写入 TABnCCS3 至 TABnCCS0 位（当 TABnCE 位= 1 时，可以写入相同的值）。如果重新写入不能正确执行，将 TABnCE 位清为 0 ，然后重新进行位的设定。

2. 请确保将第 3 位清为“0”。

（7）　TABn 捕获/比较寄存器 0（TABnCCR0）

TABnCCR0 寄存器是个 16 位寄存器，根据不同工作模式，可以用作捕获寄存器或比较寄存器 。

在自由计时器模式下，根据 TABnOPT0.TABnCCS0 位的设定，此寄存器可用作捕获寄存器或比较寄存器。在脉宽测量模式中，TABnCCR0 寄存器只可以被用作捕获寄存器。而在其它模式时，该寄存器只可以用作比较寄存器。

TABnCCR0 寄存器可以在运行中进行读取和写入。

此寄存器可以按 16 位宽度进行读取或写入。

复位输入将该寄存器清为 0000H。

复位后：0000H　　R/W　　地址：　　TAB0CCR0 FFFFF5E6H, TAB1CCR0 FFFFF626H

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TABnCCR0 (n = 0, 1)																

（**a**） 用作比较寄存器时的功能

即便在 TABnCTL0.TABnCE 位＝1 时，TABnCCR0 寄存器仍可以重新写入。

TABnCCR0 寄存器的设置值传输给 CCR0 缓冲寄存器。当 16 位计数器的值与 CCR0 缓冲寄存器的值相一致时，就会生成一个比较一致中断请求信号（INTTBnCC0）。如果此时 TOBn0 引脚输出允许，则 TOBn0 引脚的输出会反相。

在间隔定时器模式、外部事件计数器模式、外部触发脉冲输出模式、单次触发脉冲输出模式或 PWM 输出模式时，当 TABnCCR0 寄存器作为周期寄存器时，如果 16 位计数器的计数值与 CCR0 缓冲寄存器数值一致，则计数器清零为（0000H）。

将 TABnCTL0.TABnCE 位设置为 0，比较寄存器不清除。

（**b**） 用作捕获寄存器时的功能

在自由运行定时器模式下，当 TABnCCR0 寄存器用作捕获寄存器时，若检测到捕获触发输入引脚（TIBn0 引脚）的有效边沿，则 16 位计数器的当前计数值存储于 TABnCCR0 寄存器中。在脉宽测量模式下，若检测到捕获触发输入引脚（TIBn0 引脚）的有效边沿，16 位计数器的当前计数值就储存在 TABnCCR0 寄存器中，且 16 位计数器清零（0000H）。

即使捕获操作和读取 TABnCCR0 寄存器的操作发生冲突，还是可以正确读取 TABnCCR0 寄存器的值。

将 TABnCTL0.TABnCE 位设置为 0，捕获寄存器被清除。

下表显示了捕获/比较寄存器在各工作模式下的功能，以及如何向比较寄存器写入数据。

表 7-2. 捕获/比较寄存器在各模式下的功能以及如何写入比较寄存器

操作模式	捕获/比较寄存器	如何写入比较寄存器
间隔定时器	比较寄存器	随时写入
外部事件计数器	比较寄存器	随时写入
外部触发脉冲输出	比较寄存器	批量写入[注]
单次触发脉冲输出	比较寄存器	随时写入
PWM（脉宽调制信号）输出	比较寄存器	批量写入[注]
自由运行定时器	捕获/比较寄存器	随时写入
脉冲宽度测量	捕获寄存器	无

注　　写入到 TABnCCR1 寄存器是触发信号。

备注　　关于随时写入和批量写入，参见 **7.6 (2) 随时写入和批量写入**。

（8）　TABn 捕获/比较寄存器 1（TABnCCR1）

TABnCCR1 寄存器是个 16 位寄存器，根据不同工作模式，可以用作捕获寄存器或比较寄存器 。

在自由计时器模式下，根据 TABnOPT0.TABnCCS1 位的设定，此寄存器可用作捕获寄存器或比较寄存器。在脉宽测量模式中，TABnCCR1 寄存器只可以被用作捕获寄存器。而在其它模式时，该寄存器只可以用作比较寄存器。

TABnCCR1 寄存器可以在运行中进行读取和写入。

此寄存器可以按 16 位宽度进行读取或写入。

复位输入将该寄存器清为 0000H。

复位后：0000H　　R/W　　地址：　　TAB0CCR1 FFFFF5E8H, TAB1CCR1 FFFFF628H

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TABnCCR1																

(n = 0, 1)

（**a**）用作比较寄存器时的功能

即便在 TABnCTL0.TABnCE 位= 1 时，TABnCCR1 寄存器仍可以重新写入。

TABnCCR1 寄存器的设置值传输给 CCR1 缓冲寄存器。当 16 位计数器的值与 CCR1 缓冲寄存器的值相一致时，就会生成一个比较一致中断请求信号（INTTBnCC1）。如果此时 TOBn1 引脚输出允许，则 TOBn1 引脚的输出会反相。

将 TABnCTL0.TABnCE 位设置为 0，比较寄存器不清除。

（**b**）用作捕获寄存器时的功能

在自由运行定时器模式下，当 TABnCCR1 寄存器用作捕获寄存器时，若检测到捕获触发输入引脚（TIBn1 引脚）的有效边沿，则 16 位计数器的当前计数值存储于 TABnCCR1 寄存器中。在脉宽测量模式下，若检测到捕获触发输入引脚（TIBn1 引脚）的有效边沿，16 位计数器的当前计数值就储存在 TABnCCR1 寄存器中，且 16 位计数器清零（0000H）。

即使捕获操作和读取 TABnCCR1 寄存器的操作发生冲突，还是可以正确读取 TABnCCR1 寄存器的值。

将 TABnCTL0.TABnCE 位设置为 0，捕获寄存器被清除。

下表显示了捕获/比较寄存器在各工作模式下的功能，以及如何向比较寄存器写入数据。

表 7-3. 捕获/比较寄存器在各模式下的功能以及如何写入比较寄存器

操作模式	捕获/比较寄存器	如何写入比较寄存器
间隔定时器	比较寄存器	随时写入
外部事件计数器	比较寄存器	随时写入
外部触发脉冲输出	比较寄存器	批量写入[注]
单次触发脉冲输出	比较寄存器	随时写入
PWM（脉宽调制信号）输出	比较寄存器	批量写入[注]
自由运行定时器	捕获/比较寄存器	随时写入
脉冲宽度测量	捕获寄存器	无

注　　写入到 TABnCCR1 寄存器是触发信号。

备注　　关于随时写入和批量写入，参见 **7.6 (2) 随时写入和批量写入**。

（9）　TABn 捕获/比较寄存器 2（TABnCCR2）

TABnCCR2 寄存器是个 16 位寄存器，根据不同工作模式，可以用作捕获寄存器或比较寄存器 。

在自由计时器模式下，根据 TABnOPT0.TABnCCS2 位的设定，此寄存器可用作捕获寄存器或比较寄存器。在脉宽测量模式中，TABnCCR2 寄存器只可以被用作捕获寄存器。而在其它模式时，该寄存器只可以用作比较寄存器。

TABnCCR2 寄存器可以在运行中进行读取和写入。

此寄存器可以按 16 位宽度进行读取或写入。

复位输入将该寄存器清为 0000H。

复位后：0000H　　R/W　　地址：TAB0CCR2 FFFFF5EAH, TAB1CCR2 FFFFF62AH

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TABnCCR2 (n = 0, 1)																

（a）用作比较寄存器时的功能

即便在 TABnCTL0.TABnCE 位= 1 时，TABnCCR2 寄存器仍可以重新写入。

TABnCCR2 寄存器的设置值传输给 CCR2 缓冲寄存器。当 16 位计数器的计数值和 CCR2 缓冲寄存器的值一致时，也会产生一个比较一致中断请求信号（INTTBnCC2）。如果此时 TOBn2 引脚输出允许，则 TOBn2 引脚的输出会反相。

将 TABnCTL0.TABnCE 位设置为 0，比较寄存器不清除。

（b）用作捕获寄存器时的功能

在自由运行定时器模式下，当 TABnCCR2 寄存器用作捕获寄存器时，若检测到捕获触发输入引脚（TIBn2 引脚）的有效边沿，则 16 位计数器的当前计数值存储于 TABnCCR2 寄存器中。在脉宽测量模式下，若检测到捕获触发输入引脚（TIBn2 引脚）的有效边沿，16 位计数器的当前计数值就储存在 TABnCCR2 寄存器中，且 16 位计数器清零（0000H）。

即使捕获操作和读取 TABnCCR2 寄存器的操作发生冲突，还是可以正确读取 TABnCCR2 寄存器的值。

将 TABnCTL0.TABnCE 位设置为 0，捕获寄存器被清除。

下表显示了捕获/比较寄存器在各工作模式下的功能，以及如何向比较寄存器写入数据。

表 7-4. 捕获/比较寄存器在各模式下的功能以及如何写入比较寄存器

操作模式	捕获/比较寄存器	如何写入比较寄存器
间隔定时器	比较寄存器	随时写入
外部事件计数器	比较寄存器	随时写入
外部触发脉冲输出	比较寄存器	批量写入[注]
单次触发脉冲输出	比较寄存器	随时写入
PWM（脉宽调制信号）输出	比较寄存器	批量写入[注]
自由运行定时器	捕获/比较寄存器	随时写入
脉冲宽度测量	捕获寄存器	无

注　写入到 TABnCCR1 寄存器是触发信号。

备注　关于随时写入和批量写入，参见 **7.6 (2) 随时写入和批量写入**。

（10） TABn 捕获/比较寄存器 3（TABnCCR3）

TABnCCR3 寄存器是个 16 位寄存器，根据不同工作模式，可以用作捕获寄存器或比较寄存器 。

在自由计时器模式下，根据 TABnOPT0.TABnCCS3 位的设定，此寄存器可用作捕获寄存器或比较寄存器。在脉宽测量模式中，TABnCCR3 寄存器只可以被用作捕获寄存器。而在其它模式时，该寄存器只可以用作比较寄存器。

TABnCCR3 寄存器可以在运行中进行读取和写入。

此寄存器可以按 16 位宽度进行读取或写入。

复位输入将该寄存器清为 0000H。

复位后： 0000H　　R/W　　地址： TAB0CCR3 FFFFF5ECH, TAB1CCR3 FFFFF62CH

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TABnCCR3 (n = 0, 1)																

（a）用作比较寄存器时的功能

即便在 TABnCTL0.TABnCE 位= 1 时，TABnCCR3 寄存器仍可以重新写入。

TABnCCR3 寄存器的设置值传输给 CCR3 缓冲寄存器。当 16 位计数器的计数值与 CCR3 缓冲寄存器的值相一致时，就会生成一个比较一致中断请求信号（INTTBnCC3）。如果此时 TOBn3 引脚输出允许，则 TOBn3 引脚的输出会反相。

将 TABnCTL0.TABnCE 位设置为 0，比较寄存器不清除。

（b）用作捕获寄存器时的功能

在自由运行定时器模式下，当 TABnCCR3 寄存器用作捕获寄存器时，若检测到捕获触发输入引脚（TIBn3 引脚）的有效边沿，则 16 位计数器的当前计数值存储于 TABnCCR3 寄存器中。在脉宽测量模式下，若检测到捕获触发输入引脚（TIBn3 引脚）的有效边沿，16 位计数器的当前计数值就储存在 TABnCCR3 寄存器中，且 16 位计数器清零（0000H）。

即使捕获操作和读取 TABnCCR3 寄存器的操作发生冲突，还是可以正确读取 TABnCCR3 寄存器的值。

将 TABnCTL0.TABnCE 位设置为 0，捕获寄存器被清除。

下表显示了捕获/比较寄存器在各工作模式下的功能，以及如何向比较寄存器写入数据。

表 7-5. 捕获/比较寄存器在各模式下的功能以及如何写入比较寄存器

操作模式	捕获/比较寄存器	如何写入比较寄存器
间隔定时器	比较寄存器	随时写入
外部事件计数器	比较寄存器	随时写入
外部触发脉冲输出	比较寄存器	批量写入[注]
单次触发脉冲输出	比较寄存器	随时写入
PWM（脉宽调制信号）输出	比较寄存器	批量写入[注]
自由运行定时器	捕获/比较寄存器	随时写入
脉冲宽度测量	捕获寄存器	无

注　写入到 TABnCCR1 寄存器是触发信号。

备注　关于随时写入和批量写入，参见 **7.6 (2) 随时写入和批量写入**。

（11） TABn 计数器读取缓冲寄存器（TABnCNT）

TABnCNT 寄存器是读出缓冲寄存器，可以读取 16 位计数器的计数值。

如果在 TABnCTL0.TABnCE 位＝1 时读取该寄存器，可以读出 16 位定时器的计数值。

该寄存器为只读寄存器，按 16 位宽度读取。

当 TABnCE 位＝0 时，TABnCNT 寄存器的值被清除为 0000H。如果此时读取 TABnCNT 寄存器，则不会读出 16 位计数器的值（FFFFH），读到的值为 0000H 。

在复位后 TABnCNT 寄存器的值被清为 0000H，因为 TPnCE 位被清除为 0。

复位后： 0000H　　R　　地址： TAB0CNT FFFFF5EEH, TAB1CNT FFFFF62EH

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

(n = 0, 1)

7.5 定时器输出操作

下表列出了 TOBn0 至 TOBn3 引脚的操作和输出电平。

表 7-6. 各个模式下的输出操作

操作模式	TOBn0 引脚	TOBn1 引脚	TOBn2 引脚	TOBn3 引脚
间隔定时器模式	PWM 输出			
外部事件计数模式	无			
外部触发脉冲输出模式	PWM 输出	外部触发脉冲输出	外部触发脉冲输出	外部触发脉冲输出
单次触发脉冲输出模式		单次触发脉冲输出	单次触发脉冲输出	单次触发脉冲输出
PWM 输出模式		PWM 输出	PWM 输出	PWM 输出
自由运行定时器模式	PWM 输出 (仅当使用比较功能时)			
脉冲宽度测量模式	无			

备注　n = 0, 1

表 7-7. 定时器输出控制位控制下的 TOBn0 至 TOBn3 引脚真值表

TABnIOC0.TABnOLa 位	TABnIOC0.TABnOEa 位	TABnCTL0.TABnCE 位	TOBna 引脚的电平
0	0	×	低电平输出
	1	0	低电平输出
		1	计数之前的瞬间低电平，计数开始之后高电平
1	0	×	高电平输出
	1	0	高电平输出
		1	计数之前的瞬间高电平，计数开始之后低电平

备注　n = 0, 1
　　　a = 0 至 3

7.6 操作

TABn 可以执行下列操作。

表 7-8. 各个模式下的 TABn 规范

操作	TABnCTL1.TABnEST 位（软件触发位）	TRGBn 引脚（外部触发输入）	捕获/比较寄存器设置	比较寄存器写入
间隔计时器模式	无效	无效	仅作比较寄存器	随时写入
外部事件计数器模式	无效	无效	仅作比较寄存器	随时写入
外部触发脉冲输出模式	有效	有效	仅作比较寄存器	批量写入
单次触发脉冲输出模式	有效	有效	仅作比较寄存器	随时写入
PWM 输出模式	无效	无效	仅作比较寄存器	批量写入
自由定时器模式	无效	无效	可以切换	随时写入
脉冲宽度测量模式	无效	无效	仅作捕获寄存器	无效

备注 **1.** TABn 具有和 TAAn 执行调谐的功能。详情参见**第十章 电机控制功能**。

2. n = 0, 1

（1）　计数器基本操作

该节介绍 16 位计数器的基本操作。详情参见各个模式下的操作描述。

备注　n = 0, 1
a = 0 至 3

（a）　计数器开始操作

<R>
- 在外部事件计数模式下
  当 TABnCTL0.TABnCE 位从 0 设置为 1 时，16 位计数器被置为 0000H。
  此后，每次检测到外部事件计数输入（EVTBn）的有效边沿时，向上计数到 0001H，0002H，0003H，…
- 在其它模式下
  在所有模式下，从默认值 FFFFH 开始计数。
  从 FFFFH 向上计数到 0000H，0001H，0002H，0003H，如此继续。

（b）　清除操作

当 16 位计数器的值和比较寄存器的一致并清除时，当它的值被捕获时，它被清除为 0000H。在计数器开始计数之后或当计数器溢出不是清除操作时，立即发生从 FFFFH 到 0000H 的计数操作。因此，不产生 INTTBnCCa 中断信号。

（c）　溢出操作

在自由运行模式或脉冲宽度测量模式下，当计数器从 FFFFH 向上计数到 0000H 时，16 位计数器溢出。如果计数器溢出，TABnOPT0.TABnOVF 位被设置为 1，且产生一个中断请求信号（INTTBnOV）。请注意在下列条件时不产生 INTTBnOV 信号。

- 在计数操作刚刚开始之后
- 如果计数器值和比较值 FFFFH 一致并被清除
- 在脉冲宽度测量模式下，当捕获到 FFFFH 并清除时，且计数器从 FFFFH 向上计数到 0000H

注意事项　在产生溢出中断请求信号（INTTBnOV）之后，请确保检查溢出标志（TABnOVF 位）被设置为 1。

（d）　在计数操作期间的计数器读取操作

在计数操作期间，TABn 的 16 位计数器的值可以通过 TABnCNT 寄存器来读取。
当 TABnCTL0.TABnCE 位 = 1 时，16 位计数器的值可以通过 TABnCNT 寄存器来读取。当 TABnCE 位 = 0 时，16 位计数器是 FFFFH 且 TABnCNT 寄存器是 0000H。

（e）　中断操作

TABn 产生下列五种的中断请求信号。

- INTTBnCC0 中断：　该信号用作 CCR0 缓冲寄存器的一个一致中断请求信号，也用作对 TABnCCR0 寄存器的一个捕获中断请求信号。
- INTTBnCC1 中断：　该信号用作 CCR1 缓冲寄存器的一个一致中断请求信号，也用作对 TABnCCR1 寄存器的一个捕获中断请求信号。

- INTTBnCC2 中断：　该信号用作 CCR2 缓冲寄存器的一个一致中断请求信号，也用作对 TABnCCR2 寄存器的一个捕获中断请求信号。
- INTTBnCC3 中断：　该信号用作 CCR3 缓冲寄存器的一个一致中断请求信号，也用作对 TABnCCR3 寄存器的一个捕获中断请求信号。
- INTTBnOV 中断：　该信号用作一个溢出中断请求信号。

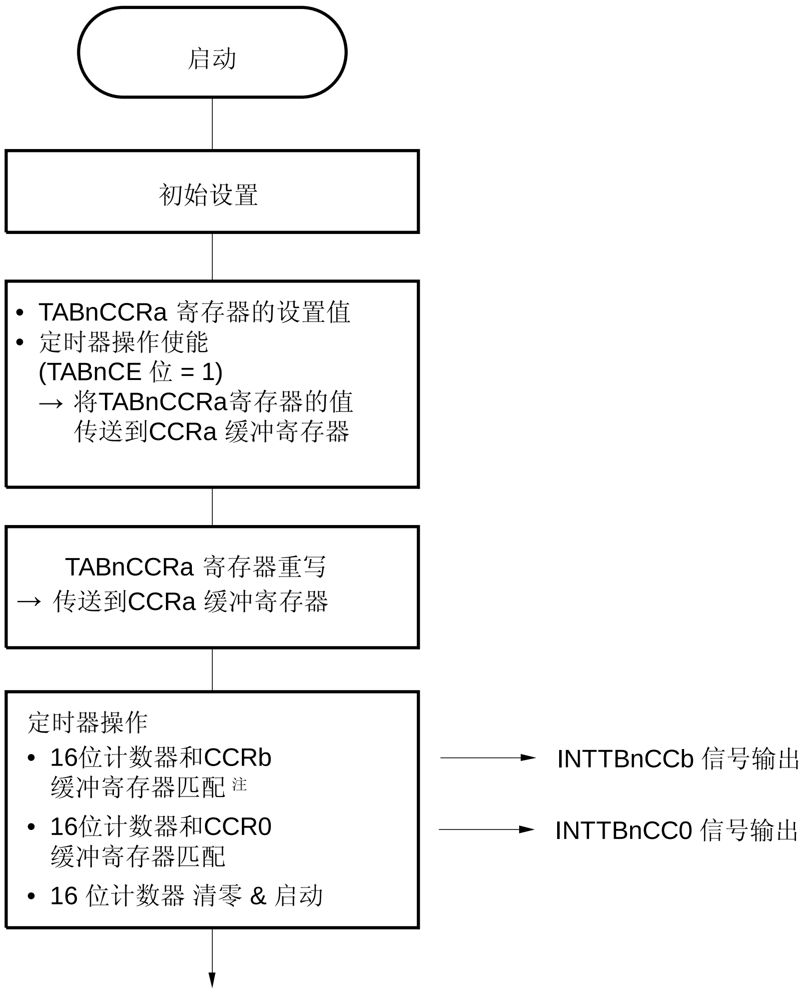
（2）随时写入和批量写入

在定时器操作期间（TABnCTL0.TABnCE 位 = 1），TABn 的 TABnCCR0 至 TABnCCR3 寄存器可以重写，但是 CCR0 至 CCR3 缓冲寄存器的写入方法（随时写入，批量写入）取决于不同的模式。

（a）　随时写入

该模式下，在定时器操作期间，数据随时可以从 TABnCCR0 至 TABnCCR3 寄存器传输到 CCR0 至 CCR3 缓冲寄存器。

图 7-2. 随时写入基本操作的流程图

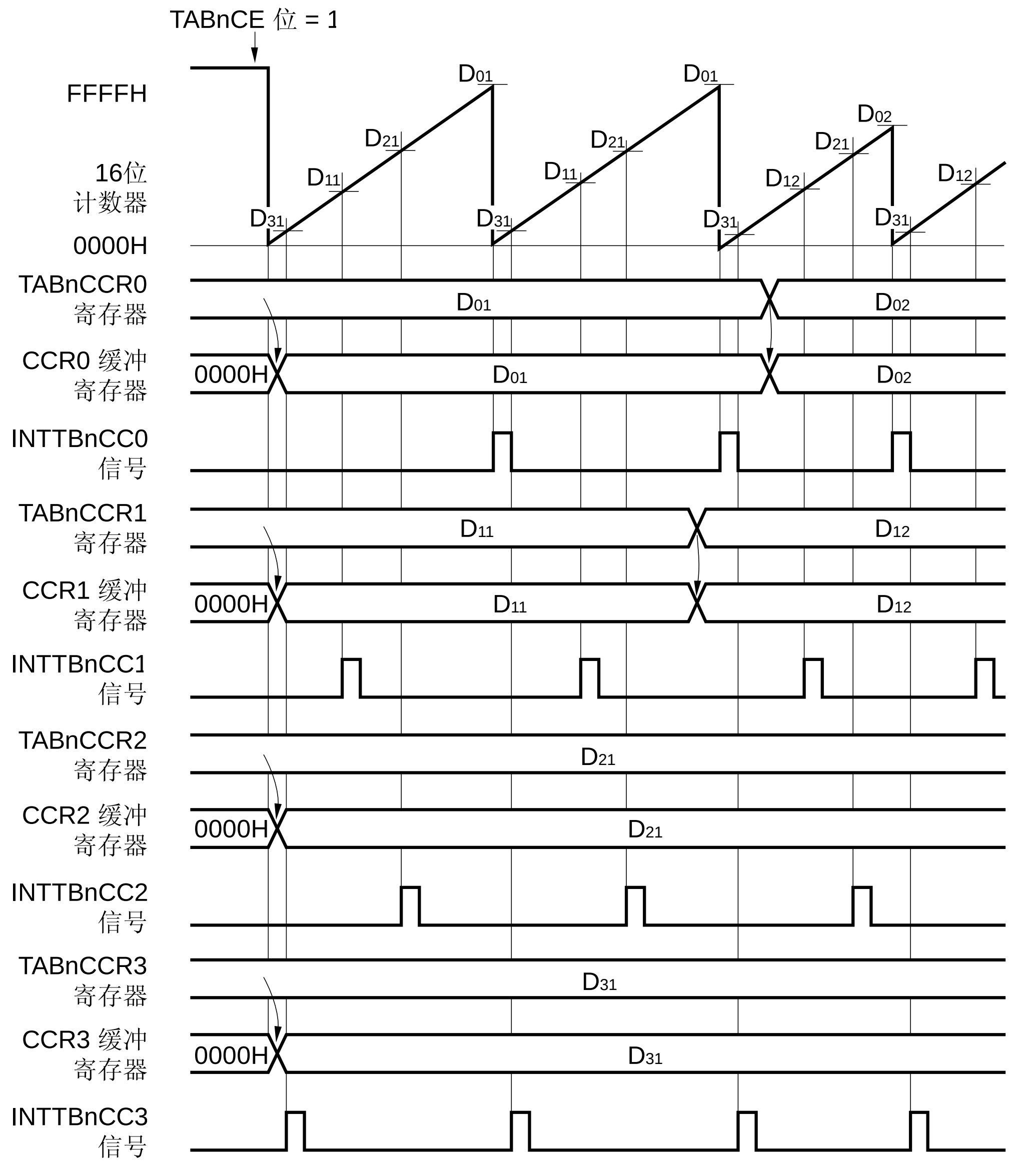


注　　在 16 位计数器的值和 CCRb 缓冲寄存器的值一致时，16 位计数器不清除。在 16 位计数器的值和 CCR0 缓冲寄存器的值一致时，才清除 16 位计数器。

备注　　**1.** 上述流程图展示了一个间隔定时器模式下的操作示例。

2. n = 0, 1
a = 0 至 3
b = 1 至 3

图 7-3. 随时写入的时序



备注 1. D_{01}，D_{02}： TABnCCR0 寄存器的设置值

D_{11}，D_{12}： TABnCCR1 寄存器的设置值

D_{21}： TABnCCR2 寄存器的设置值

D_{31}： TABnCCR3 寄存器的设置值

2. 上述流程图展示了一个间隔定时器模式下的操作示例。

3. n = 0，1

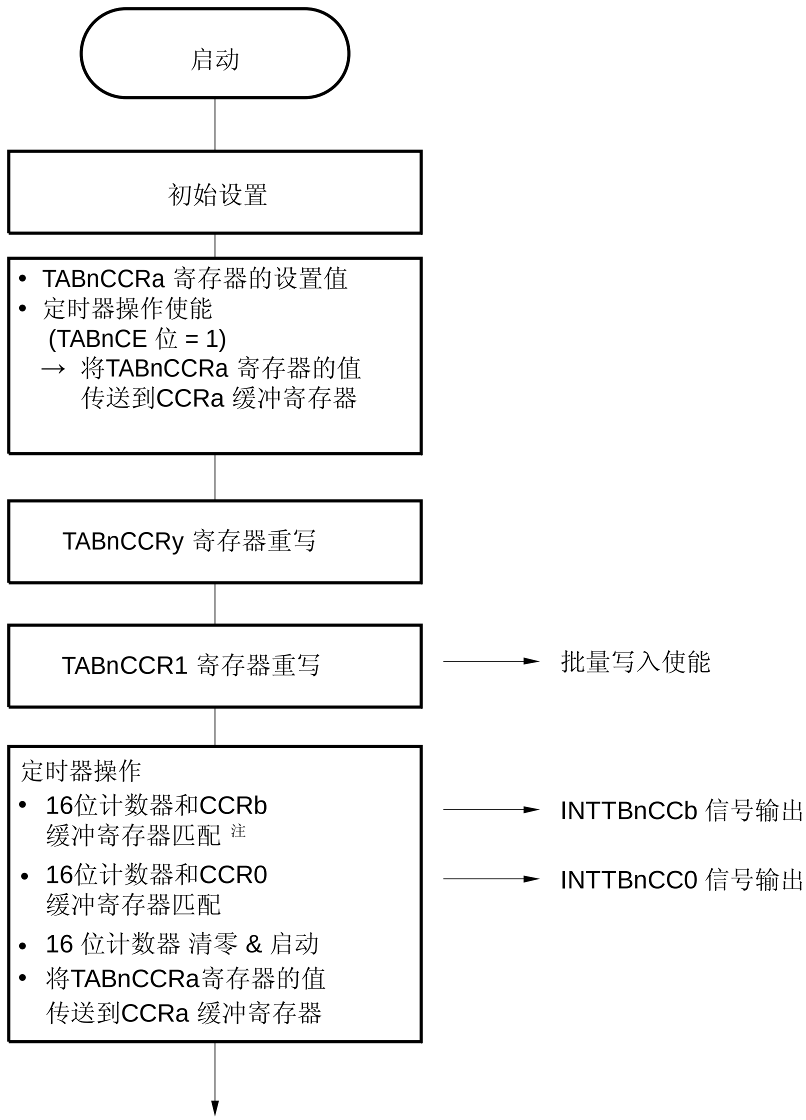
（b） 批量写入

该模式下，在定时器操作期间，数据一次性从 TABnCCR0 至 TABnCCR3 寄存器传输到 CCR0 至 CCR3 缓冲寄存器。在 16 位计数器的值和 CCR0 缓冲寄存器的值一致时，传输该数据。写入 TABnCCR1 寄存器就可以使能传输。

是否写入 TABnCCR1 寄存器可以控制使能或禁止下一次传输时序。

为了让重写 TABnCCR0 至 TABnCCR3 寄存器的设置值成为 16 位计数器比较值（换句话说，为了该值被传输到 CCR0 至 CCR3 缓冲寄存器），必需重写 TABnCCR0 寄存器，然后在 16 位计数器的值和 CCR0 缓冲寄存器的值一致之前写入 TABnCCR1 寄存器。因此，在 16 位计数器的值和 CCR0 缓冲寄存器的值一致时，TABnCCR0 至 TABnCCR3 寄存器的值被传输到 CCR0 至 CCR3 缓冲寄存器。所以，即使只想重写 TABnCCR0，TABnCCR2 或 TABnCCR3 寄存器的值，也需要将相同的值（和 TABnCCR1 寄存器的当前值相同）写入 TABnCCR1 寄存器。

图 7-4. 批量写入基本操作的流程图



注　在 16 位计数器的值和 CCRb 缓冲寄存器的值一致时，16 位计数器不清除。在 16 位计数器的值和 CCR0 缓冲寄存器的值一致时，才清除 16 位计数器。

注意事项　写入 TABnCCR1 寄存器包括批量写入的使能。于是，在重写 TABnCCR0，TABnCCR2 和 TABnCCR3 寄存器之后，重写 TABnCCR1 寄存器。

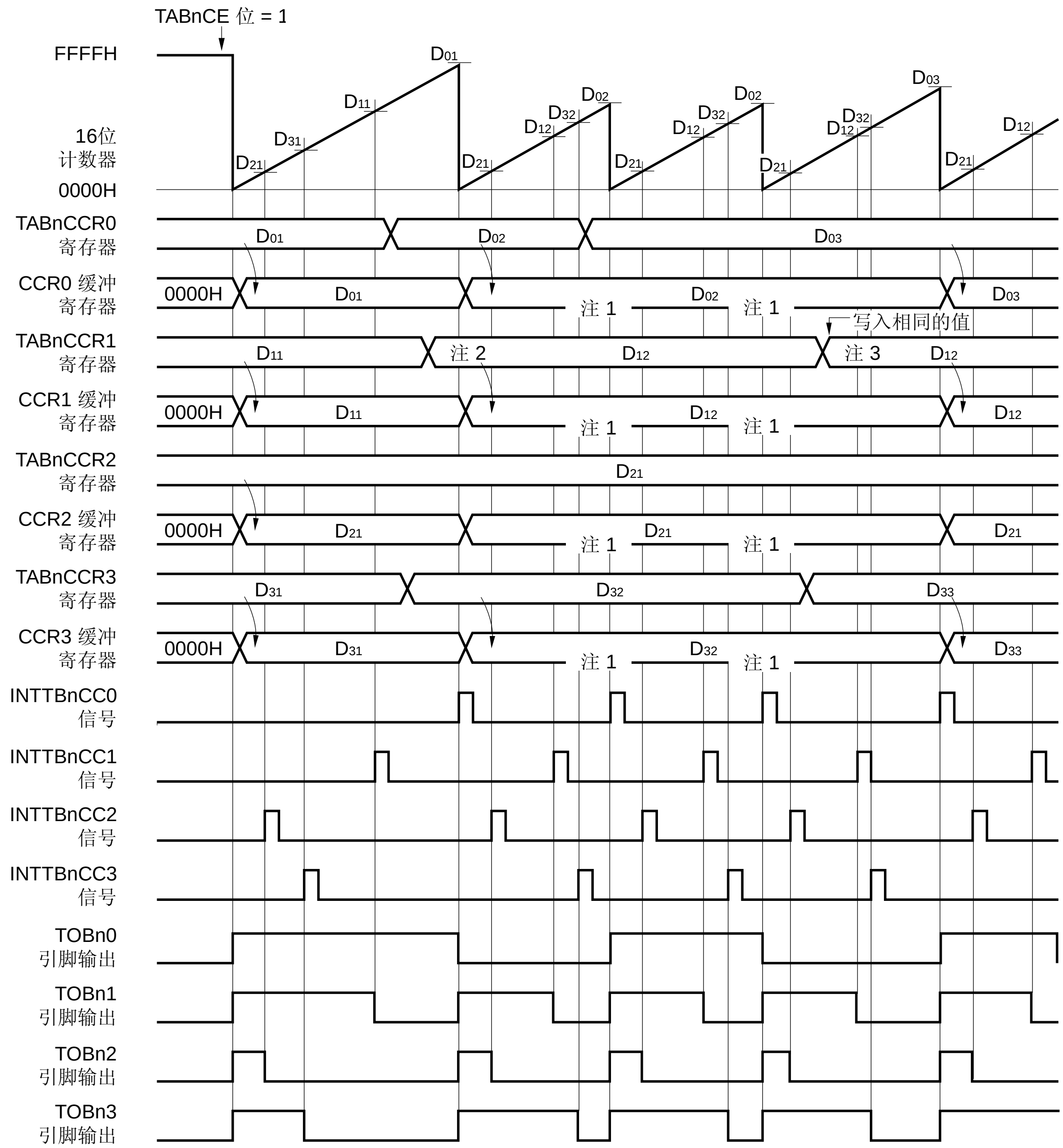
备注　**1.** 上述流程图展示了一个 PWM 输出模式下的操作示例。

2. a = 0 至 3

b = 1 至 3

y = 0, 2, 3

图 7-5. 批量写入的时序



注　**1.** 因为 TABnCCR1 寄存器没有被重写，D_{02} 不传输。

2. 因为 TABnCCR1 寄存器已经被写入（D_{12}），当 16 位计数器的值和 TABnCCR0 寄存器的值（D_{01}）一致时，数据被传输到 CCR1 缓冲寄存器。

3. 因为 TABnCCR1 寄存器已经被写入（D_{12}），当 16 位计数器的值和 TABnCCR0 寄存器的值（D_{02}）一致时，数据被传输到 CCR1 缓冲寄存器。

备注　**1.** D_{01}，D_{02}，D_{03}：　TABnCCR0 寄存器的设置值

D_{11}，D_{12}：　TABnCCR1 寄存器的设置值

D_{21}：　TABnCCR2 寄存器的设置值

D_{31}, D_{32}, D_{33}：　TABnCCR3 寄存器的设置值

2. 上述流程图展示了一个 PWM 输出模式下的操作示例。

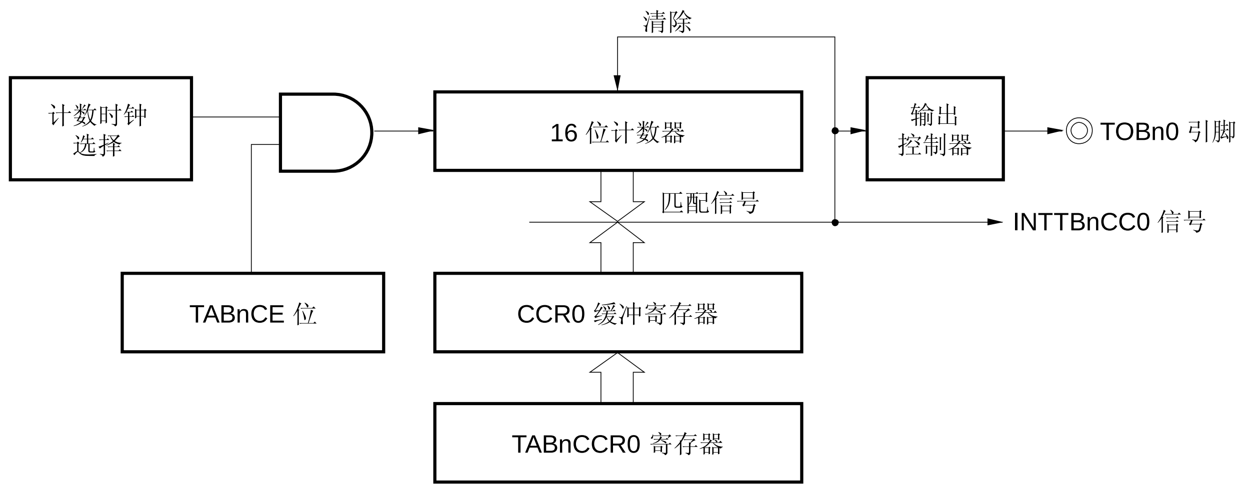
7.6.1　间隔定时器模式（TABnMD2 至 TABnMD0 位＝000）

在间隔定时器模式下，如果 TABnCTL0.TABnCE 位被置为 1，则会以 TABnCCR0 寄存器指定的时间间隔生成中断请求信号（INTTBnCC0），从 TOBn0 引脚输出占空比 50%的 PWM 波形，其半周期等于该指定的时间间隔。

间隔定时器模式不使用 TABnCCR1 至 TABnCCR3 寄存器。但是，TABnCCR1 至 TABnCCR3 寄存器的设置值被传输到 CCR1 至 CCR3 缓冲寄存器，且当 16 位计数器的计数值与 CCR1 至 CCR3 缓冲寄存器一致时，产生比较一致中断请求信号(INTTBnCC1 至 INTTBnCC3)。而且，可以从 TOBn1 至 TOBn3 引脚输出一个占空比 50%的 PWM 波形，当产生 INTTBnCC1 至 INTTBnCC3 信号时，该波形被反转。

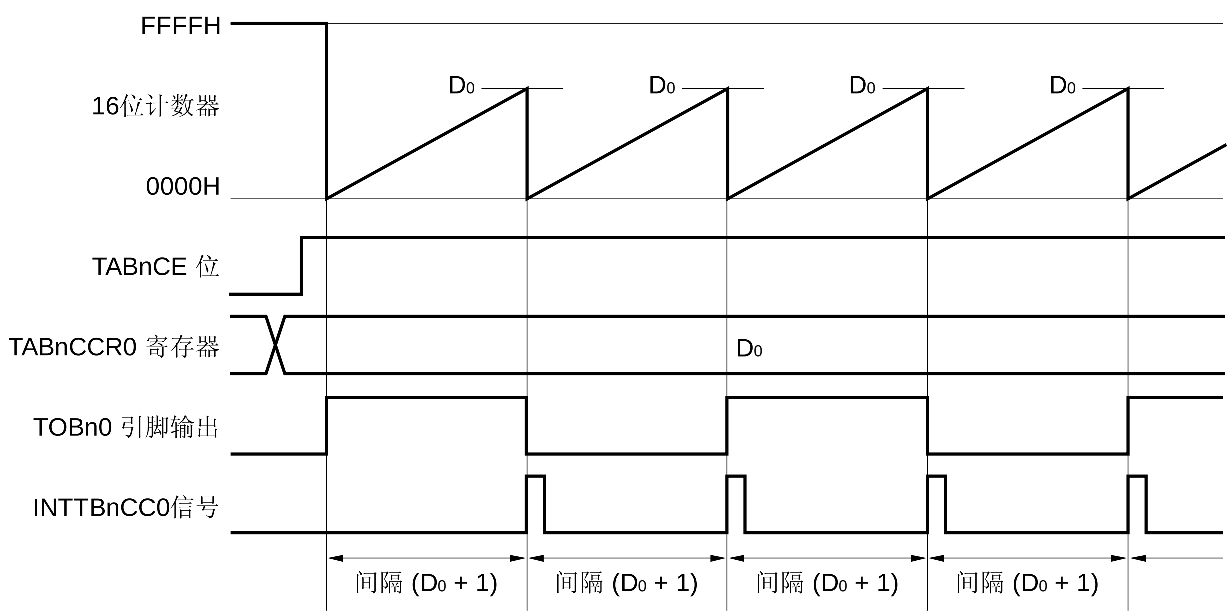
即使在定时器工作时，也可以重写 TABnCCR1 至 TABnCCR3 寄存器的值。

图 7-6. 间隔定时器的配置



备注　n = 0, 1

图 7-7. 间隔定时器模式的基本时序



备注　n = 0, 1

当 TABnCE 位被置为 1 时，16 位计数器的值由计数时钟同步从 FFFFH 清零为 0000H，计数器开始计数。此时，TOBn0 引脚的输出被反转。此外，TABnCCR0 寄存器的设置值传输给 CCR0 缓冲寄存器。

当 16 位计数器的计数值和 CCR0 缓冲寄存器的值相一致时，则 16 位计数器清除为 0000H，TOBn0 引脚输出反相，并生成一个比较一致中断请求信号（INTTBnCC0）。

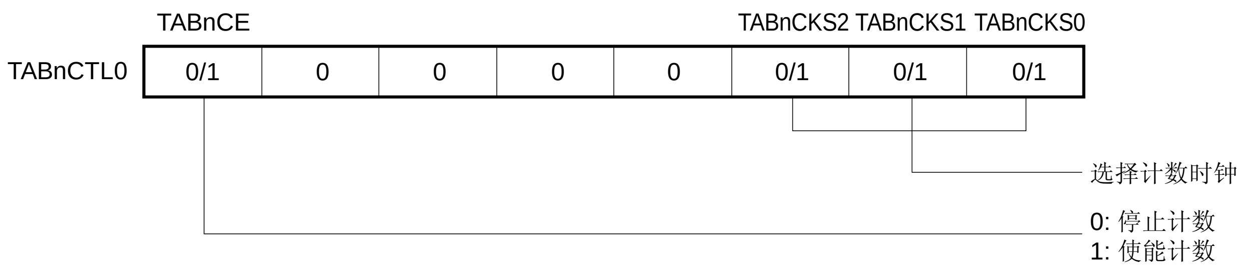
时间间隔可用下述表达式计算。

时间间隔 =（TABnCCR0 寄存器的设置值＋1） × 计数时钟周期

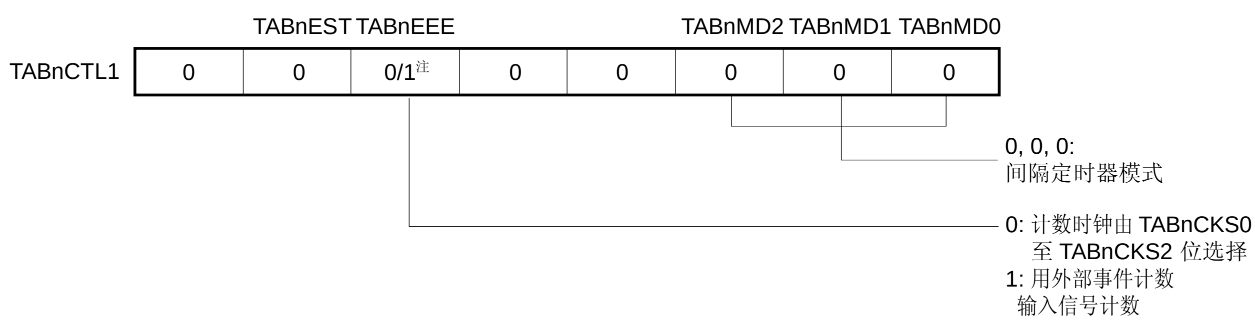
备注　n = 0, 1

图 7-8. 间隔定时器模式操作时的寄存器设置（1/3）

（a） TABn 控制寄存器 0（TABnCTL0）



（b） TABn 控制寄存器 1（TABnCTL1）



注　仅当使用定时器输出(TOBn0 至 TOBn3)时，TABnEEE 位才可以被设置为 1。但是，TABnCCR0 至 TABnCCR3 寄存器设置为相同值。

图 7-8. 间隔定时器模式操作时的寄存器设置（2/3）

（c） TABn 输入/输出控制寄存器 0（TABnIOC0）

	TABnOL3	TABnOE3	TABnOL2	TABnOE2	TABnOL1	TABnOE1	TABnOL0	TABnOE0
TABnIOC0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TABnOE0:
0: 禁止 TOBn0 引脚输出
1: 使能 TOBn0 引脚输出

TABnOL0: 在计数操作前，设置 TOBn0 引脚输出电平
0: 低电平
1: 高电平

TABnOE1:
0: 禁止 TOBn1 引脚输出
1: 使能 TOBn1 引脚输出

TABnOL1: 在计数操作前，设置 TOBn1 引脚输出电平
0: 低电平
1: 高电平

TABnOE2:
0: 禁止 TOBn2 引脚输出
1: 使能 TOBn2 引脚输出

TABnOL2: 在计数操作前，设置TOBn2 引脚输出电平
0: 低电平
1: 高电平

TABnOE3:
0: 禁止 TOBn3 引脚输出
1: 使能 TOBn3 引脚输出

TABnOL3: 在计数操作前，设置TOBn3 引脚输出电平
0: 低电平
1: 高电平

(d) TABn I/O 控制寄存器 2 (TABnIOC2)

					TABnEES1	TABnEES0	TABnETS1	TABnETS0
TABnIOC2	0	0	0	0	0/1注	0/1注	0	0

TABnEES1, TABnEES0: 选择外部有效边沿事件计数输入 (EVTBn 引脚).

注　仅当使用定时器输出(TOBn0 至 TOBn3)时，TABnEES1 和 TABnEES0 位才可以被置位。但是，TABnCCR0 至 TABnCCR3 寄存器设置为相同值。

（e） TABn 计数器读取缓冲寄存器（TABnCNT）

通过读取 TABnCNT 寄存器，可以读取 16 计数器的当前值。

（f） TABn 捕获/比较寄存器 0（TABnCCR0）

如果 TABnCCR0 寄存器的值被设置为 D_0，则时间间隔的计算如下：

时间间隔＝（D_0＋1）× 计数时钟周期

图 7-8. 间隔定时器模式操作时的寄存器设置（3/3）

（g） TABn 捕获/比较寄存器 1 至 3（TABnCCR1 至 TABnCCR3）

间隔定时器模式不使用 TABnCCR1 至 TABnCCR3 寄存器。但是，TABnCCR1 至 TABnCCR3 寄存器的设置值仍然会被传输到 CCR1 至 CCR3 缓冲寄存器。当 16 位计数器的计数值与 CCR1 至 CCR3 缓冲寄存器的值一致时，会产生比较一致中断请求信号（INTTBnCC1 至 INTTBnCC3）。

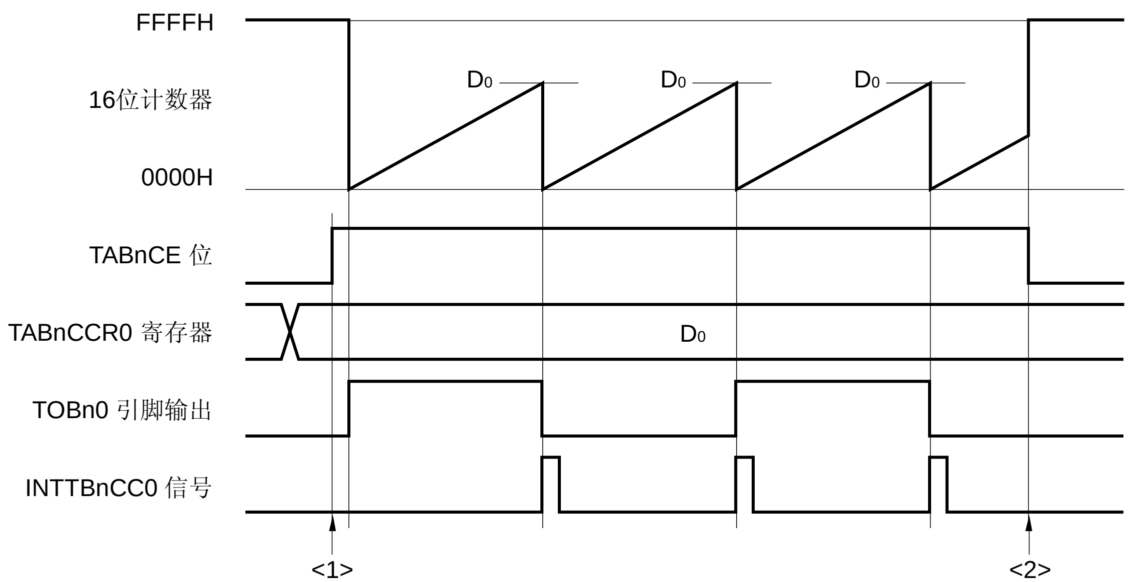
不使用 TABnCCR1 至 TABnCCR3 寄存器时，推荐将它们的值设置为 FFFFH。并且通过中断屏蔽标志(TABnCCIC1.TABnCCMK1 至 TABnCCIC3.TABnCCMK3)屏蔽这些寄存器。

备注 **1.** 间隔定时器模式不使用 TABn I/O 控制寄存器 1 (TABnIOC1)和 TABn 选项寄存器 0 (TABnOPT0)。

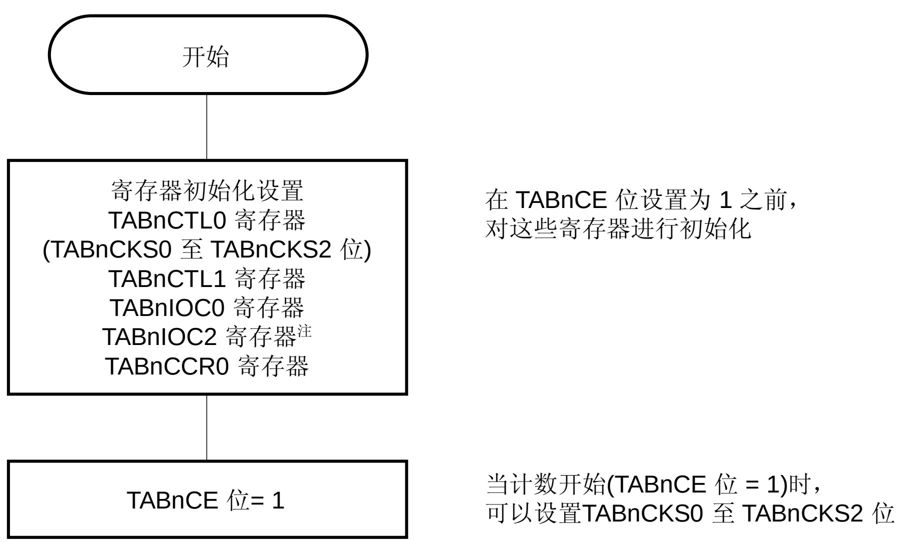
2. n = 0, 1

（1）　间隔定时器模式操作流程

图 7-9. 间隔定时器模式下的软件处理流程（1/2）



<1> 计数操作开始流程

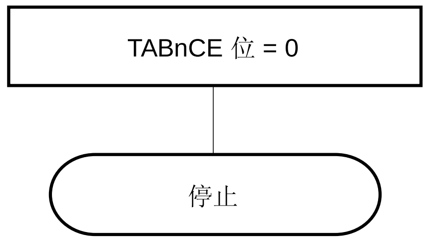


注　仅当使用定时器输出(TOBn0 至 TOBn3)时，TABnEES1 和 TABnEES0 位才可以被置位。但是，TABnCCR0 至 TABnCCR3 寄存器设置为相同值。

备注　n = 0, 1

图 **7-9.** 间隔定时器模式下的软件处理流程（**2/2**）

<2> 计数器操作停止流程



对计数器初始化，并且将 TABnCE 位清为0，停止计数
TOBn0 引脚的输出电平由 TABnIOC0 寄存器指定

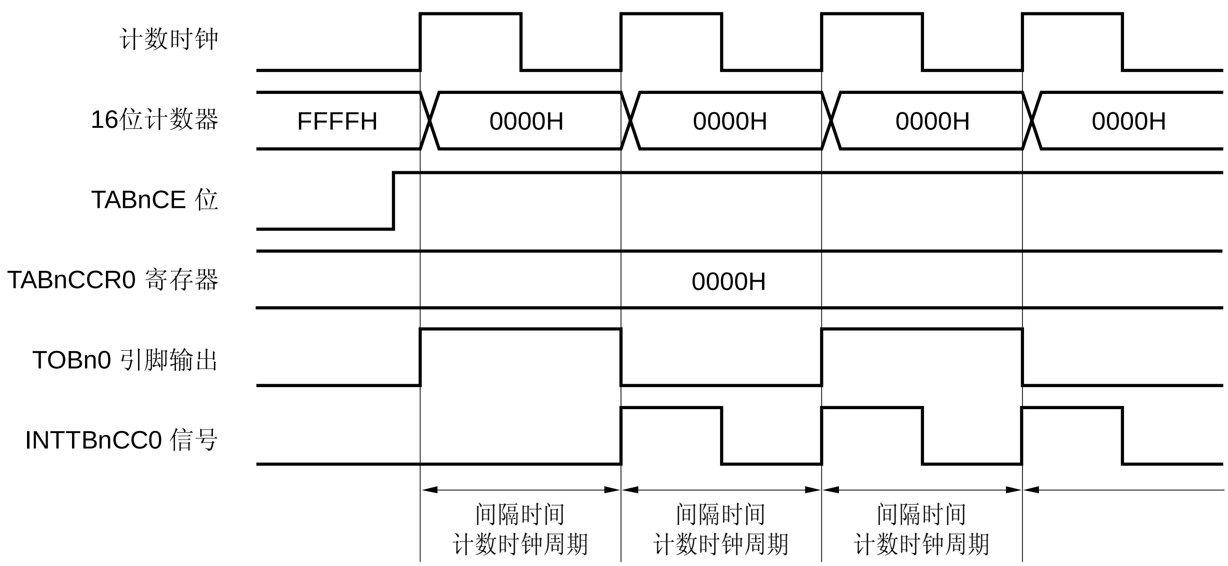
备注 n = 0, 1

（2）　间隔定时器模式操作时序

（a） TABnCCR0 寄存器清为 0000H 时的操作

如果 TABnCCR0 寄存器被清除为 0000H，每个时钟计数时都会产生 INTTBnCC0 信号，而且，TOBn0 引脚的输出被反转。

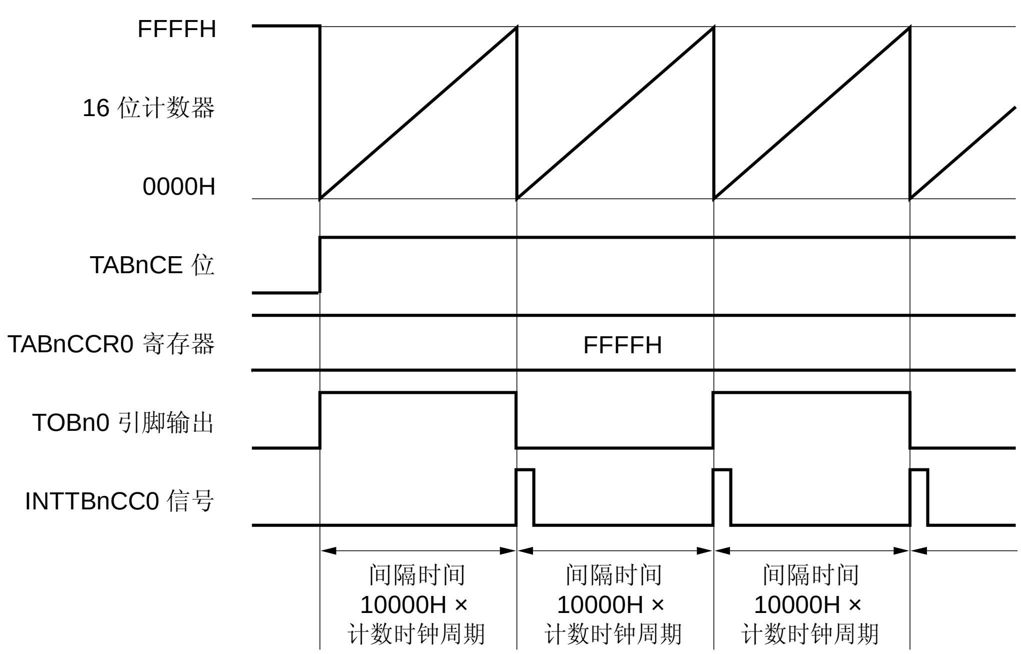
16 位计数器的值总是 0000H。



备注　n = 0, 1

（b） TABnCCR0 寄存器被设置为 FFFFH 时的操作

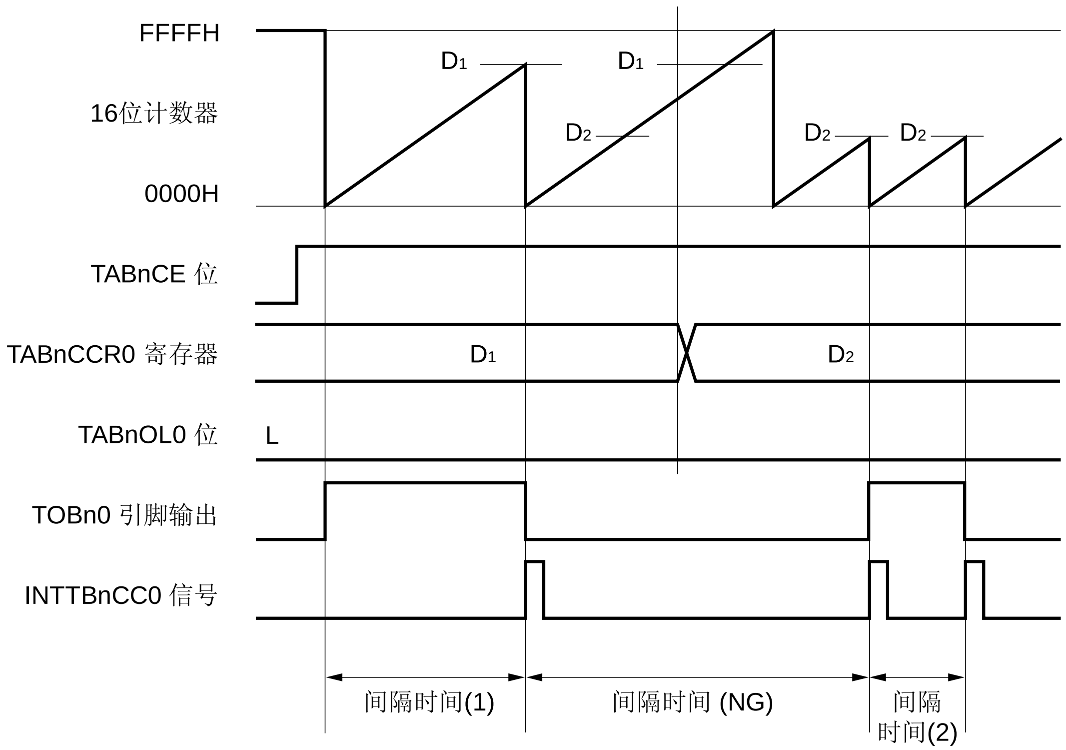
如果 TABnCCR0 寄存器被置为 FFFFH，那么 16 位计数器可以计数到 FFFFH。该计数器在下一个计数时刻被同步清零。产生 INTTBnCC0 信号，并且 TOBn0 引脚输出反相。此时，不产生溢出中断请求信号（INTTBnOV），而且，溢出标志（TABnOPT0.TABnOVF 位）也不会置 1。



备注　n = 0, 1

（c）　重写 TABnCCR0 寄存器的注意事项

如果在计数期间，将寄存器 TABnCCR0 的值重写为比当前值小的值，16 位计数器可能会溢出。当可能发生溢出时，停止计数，然后改变其设置值。



备注　**1.**　间隔时间（1）：　（D_1＋1）× 计数时钟周期

间隔时间（NG）：（10000H＋D_2＋1）× 计数时钟周期

间隔时间（2）：　（D_2＋1）× 计数时钟周期

2. n = 0, 1

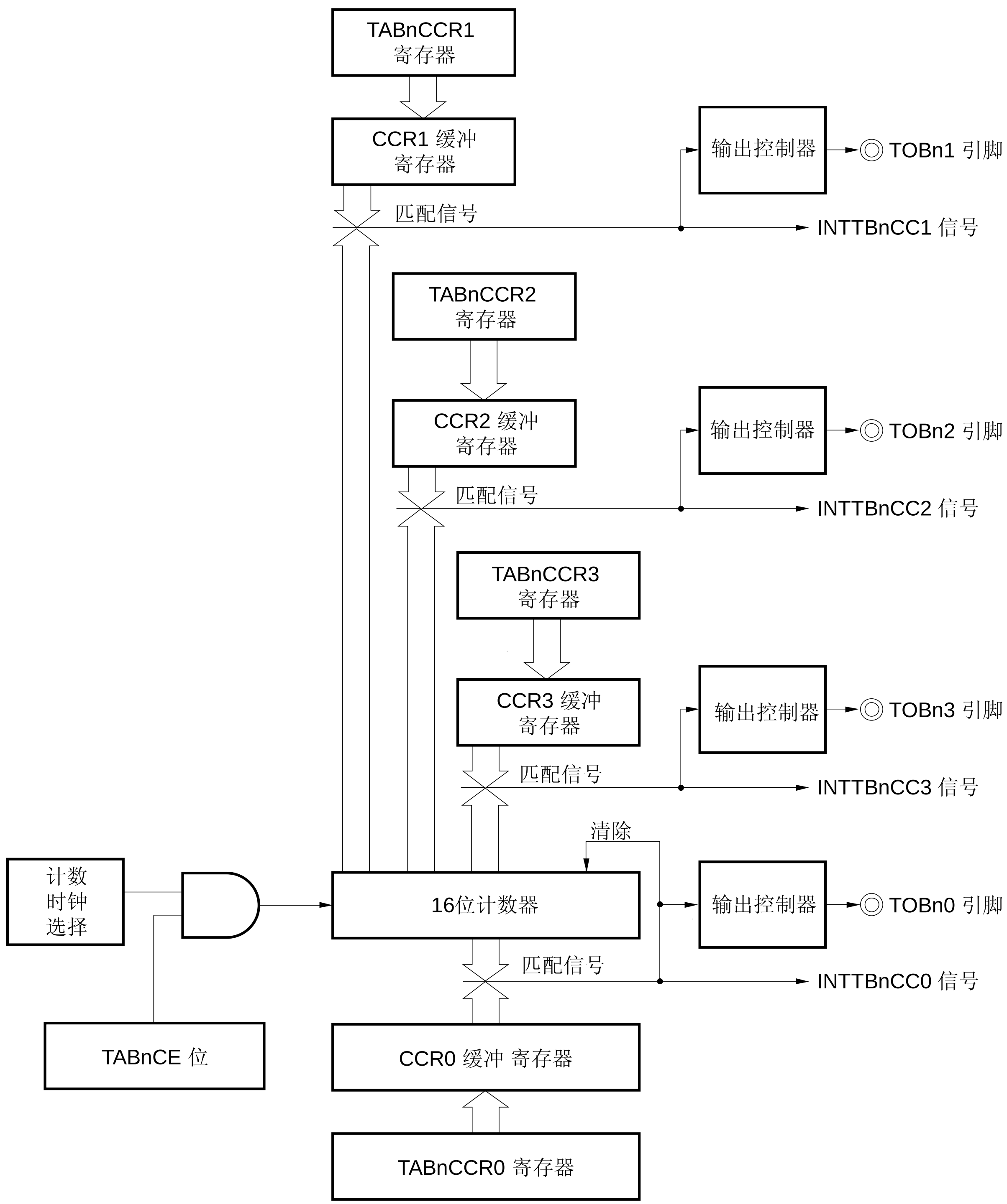
当 16 位计数器计数值大于 D2 而小于 D1 时，如果将 TABnCCR0 寄存器的值从 D_1 变为 D_2，则在重写 TABnCCR0 寄存器后，计数值会立即传输给 CCR0 缓冲寄存器。结果，用来比较的 16 位计数器的值是 D_2。

但是，由于计数值已经超过了 D_2，所以 16 位计数器就会一直向上累积计数到 FFFFH，溢出，然后从 0000H 开始重新计数。当计数值和 D_2 相符时才产生 INTTBnCC0 信号，且 TOBn0 引脚的输出反相。

因此，可能不会在原先预计的间隔时间"（D_1＋1）× 计数时钟周期"或"（D_2＋1）× 计数时钟周期"时产生 INTTBnCC0 信号，而会以"（10000H＋D_2＋1）× 计数时钟周期"的间隔产生中断请求信号。

（d） TABnCCR1 至 TABnCCR3 寄存器的操作

图 7-10. TABnCCR1 至 TABnCCR3 寄存器的配置



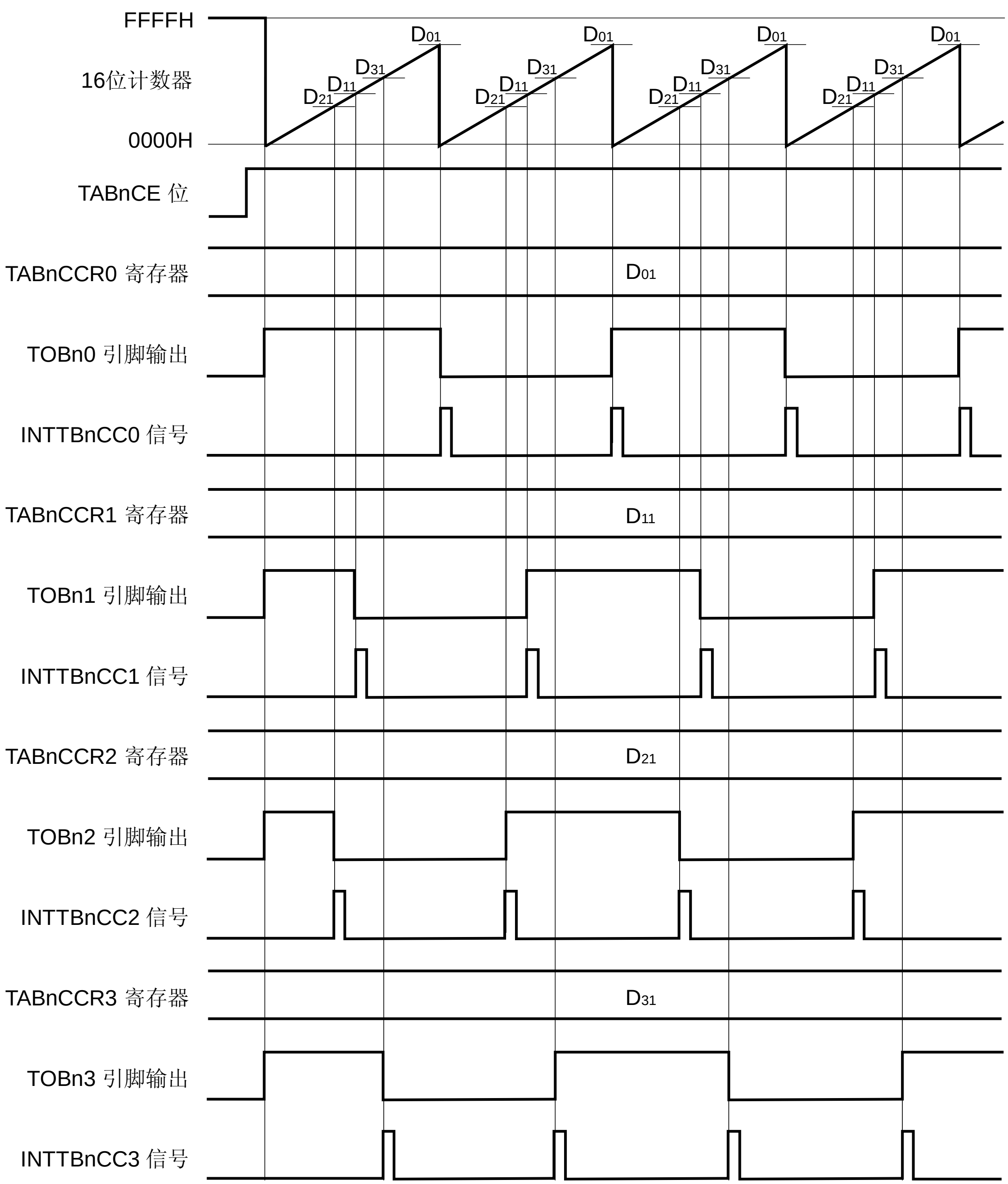
备注　n = 0, 1

如果 TABnCCRb 寄存器的设置值和 TABnCCR0 寄存器的值相等，则产生 INTTBnCCb 信号的时序和 INTTBnCC0 信号的时序相同，且 TOBnb 引脚的输出被反转。也就是说，可以从 TOBnb 引脚输出一个占空比 50%的 PWM 波形。

下面展示了 TABnCCRb 寄存器的设置值和 TABnCCR0 寄存器的值不等时的操作。

如果 TABnCCRb 寄存器的设置值小于 TABnCCR0 寄存器的值，每个周期产生 INTTBnCCb 信号。同时，TOBnb 引脚的输出被反转。在输出一个窄脉冲之后，从 TOBnb 引脚输出一个占空比 50%的 PWM 波形。

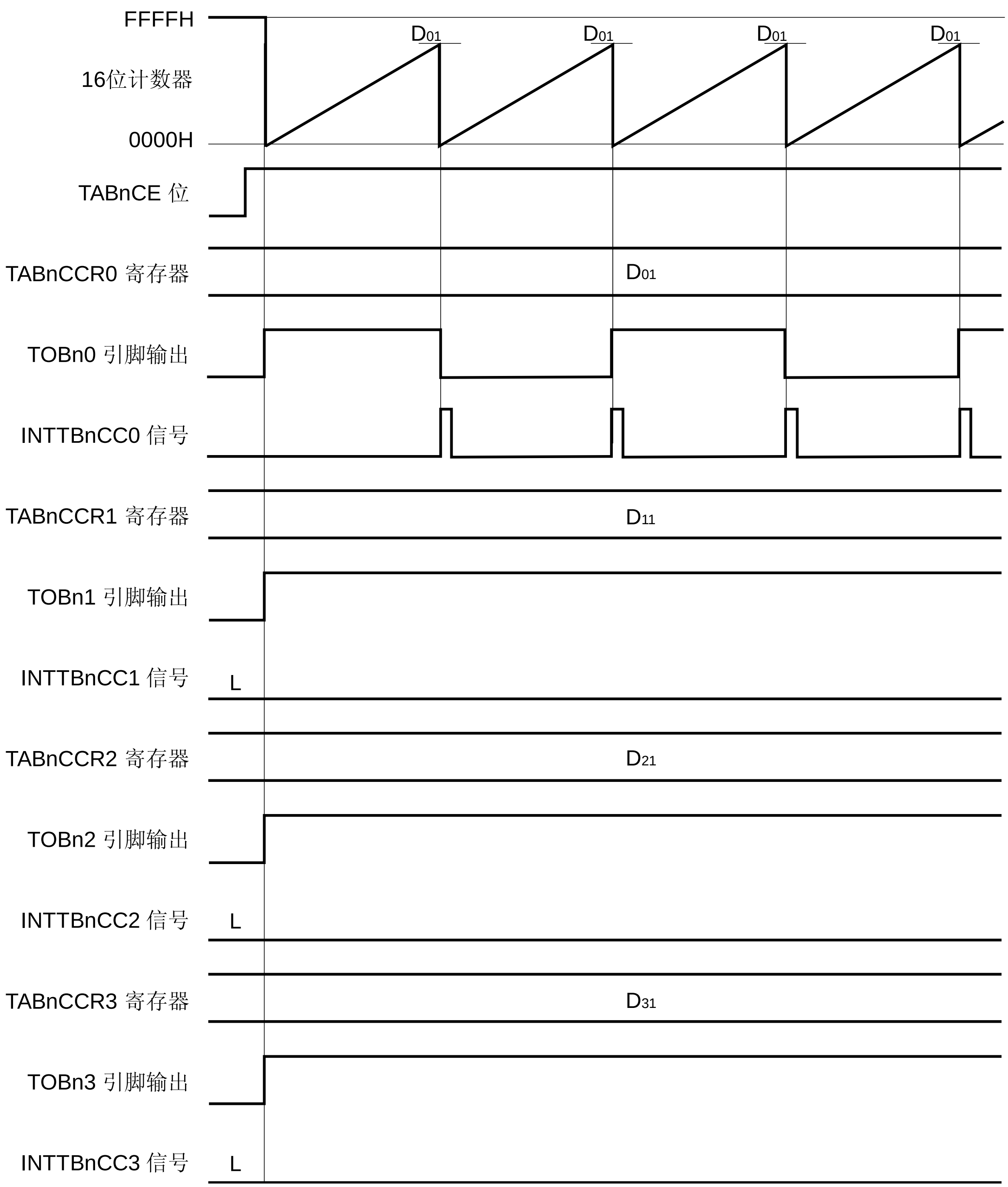
图 7-11.　当 $D_{01} \geq D_{b1}$ 时的时序图



备注　n = 0, 1
　　　b = 1 至 3

如果 TABnCCRb 寄存器的设置值大于 TABnCCR0 寄存器的值，则 16 位计数器的计数值不会和 TABnCCRb 寄存器的值发生一致。因此也就不会产生 INTTBnCCb 信号，TOBnb 引脚的输出也不会改变。不使用 TABnCCRb 寄存器时，推荐将其值设置为 FFFFH。

图 7-12. 当 D_{01} < D_{b1} 时的时序图



备注 n = 0, 1
b = 1 至 3

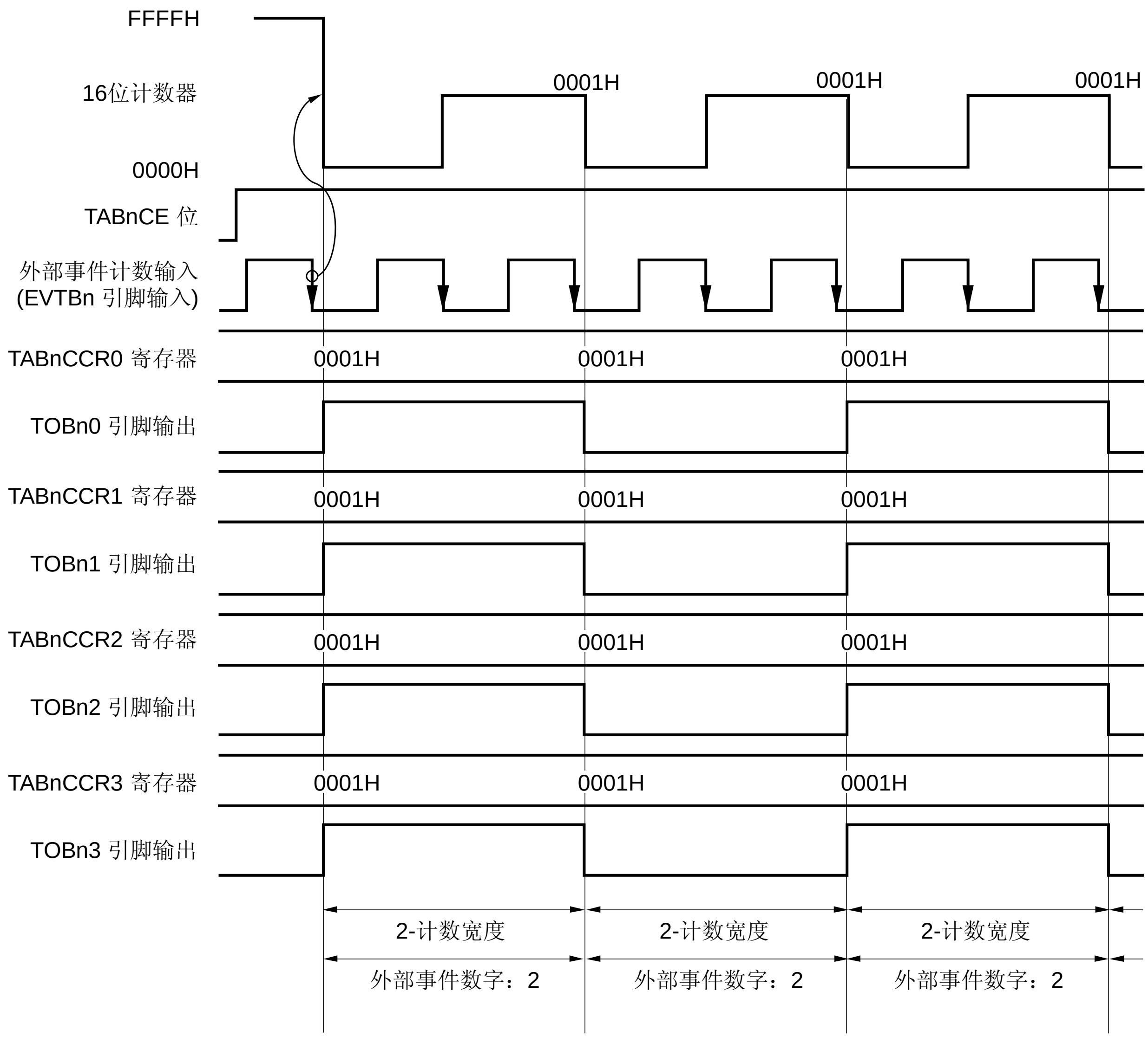
(3) 外部事件计数输入 (EVTBn)的操作

(a) 操作

在间隔定时器模式下，要在外部事件计数输入 (EVTBn)的有效边沿处计数 16 位计数器，在 TABnCE 位从 0 设置为 1 之后，通过外部事件计数输入的有效边沿将 16 位计数器从 FFFFH 清除为 0000H。

当 TABnCCR0 和 TABnCCRb 寄存器的设置值都是 0001H 时，每当 16 位计数器计数两次，TOBn0 和 TOBnb 引脚的输出被反转(b = 1 至 3)。

仅当定时器输出(TOBn0, TOBnb) 和外部事件计数输入一起使用时，TABnCTL0.TABnEEE 位可以在间隔定时器模式下被设置为 1。



备注　n = 0, 1
　　　b = 1 至 3

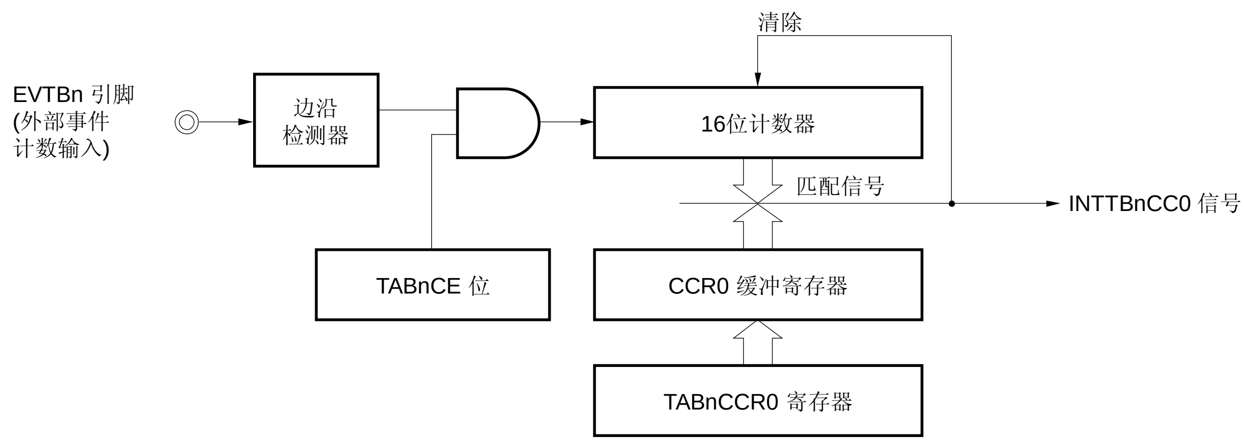
7.6.2　外部事件计数模式（TABnMD2 至 TABnMD0 位＝001）

在外部事件计数器模式下，当 TABnCTL0.TABnCE 位被置为 1 时，对外部事件计数输入(EVTBn)的有效边沿进行计数，每次有效边沿的计数数量达到 TABnCCR0 寄存器指定的数量时，产生一个中断请求信号（INTTBnCC0）。不能使用 TOBn0 至 TOBn3 引脚。

当 TOBn0 至 TOBn3 引脚用于配合外部事件计数输入时，在间隔定时器模式下将 TABnCTL1.TABnEEE 位设置为 1（参见 **7.6.1（3） 外部事件计数输入(EVTBn)的操作**）。

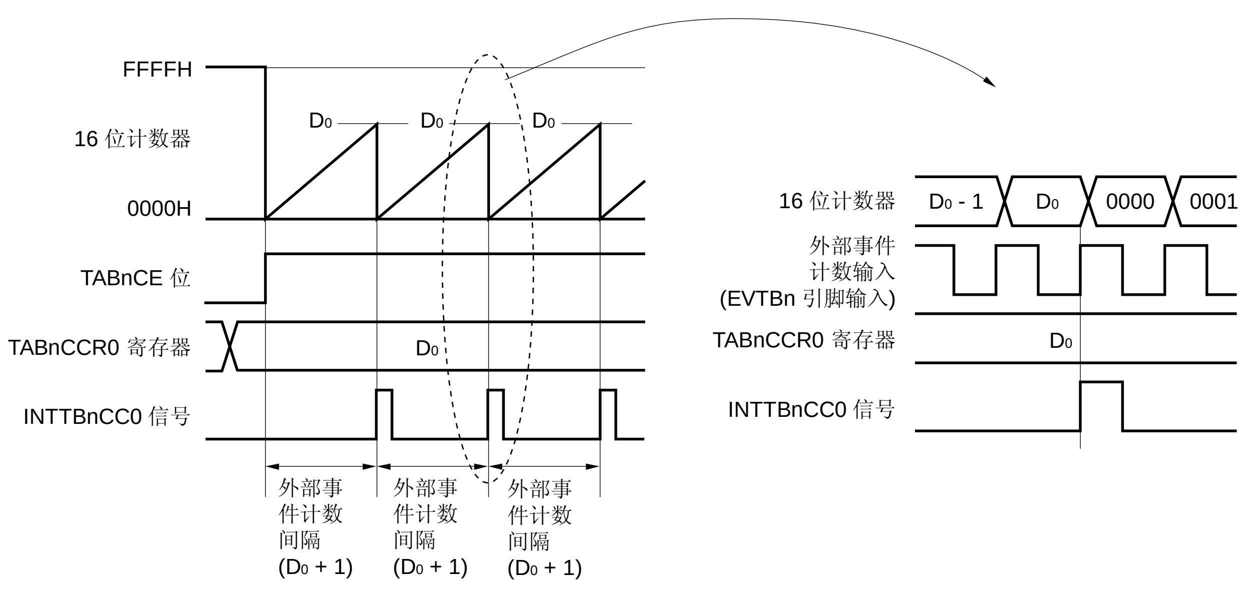
外部事件计数模式不使用 TABnCCR1 至 TABnCCR3 寄存器。

图 7-13. 外部事件计数模式的配置



备注　n = 0, 1

图 7-14. 外部事件计数模式的基本时序



备注　**1.** 本图显示了当上升沿被指定作为外部事件计数输入有效边沿时的基本时序。
2. n = 0, 1

当 TABnCE 位被置为 1 时，16 位计数器的值从 FFFFH 被清除为 0000H。计数器对检测到的每一次外部事件计数输入的有效边沿进行计数。此外，TABnCCR0 寄存器的设置值被传输给 CCR0 缓冲寄存器。

当 16 位计数器的计数值和 CCR0 缓冲寄存器的数值相一致时，16 位计数器清零为 0000H，同时生成一个比较一致中断请求信号（INTTBnCC0）。

每当外部事件计数的有效边沿数量检测达到（TABnCCR0 寄存器的设置值＋1）时，就会生成 INTTBnCC0（比较一致中断请求）信号。

图 7-15. 外部事件计数模式下的寄存器操作设置（1/2）

（a） TABn 控制寄存器 0（TABnCTL0）

	TABnCE					TABnCKS2	TABnCKS1	TABnCKS0
TABnCTL0	0/1	0	0	0	0	0	0	0

TABnCE：
0: 停止计数
1: 使能计数

（b） TABn 控制寄存器 1（TABnCTL1）

		TABnEST	TABnEEE			TABnMD2	TABnMD1	TABnMD0
TABnCTL1	0	0	0	0	0	0	0	1

TABnMD2, TABnMD1, TABnMD0：
0, 0, 1:
外部事件计数模式

（c） TABn I/O 控制寄存器 2（TABnIOC2）

					TABnEES1	TABnEES0	TABnETS1	TABnETS0
TABnIOC2	0	0	0	0	0/1	0/1	0	0

TABnEES1, TABnEES0：
选择外部事件计数
输入的有效边沿 (EVTBn 引脚)

（d） TABn 计数器读取缓冲寄存器（TABnCNT）

可以通过读取 TABnCNT 寄存器，读取 16 位计数器的计数值。

（e） TABn 捕获/比较寄存器 0（TABnCCR0）

如果 TABnCCR0 寄存器的值被设定为 D_0，当外部事件计数值达到（D_0＋1）时，产生比较一致中断请求信号（INTTBnCC0）。

图 7-15. 外部事件计数模式下的寄存器操作设置（2/2）

（f） TABn 捕获/比较寄存器 1 至 3（TABnCCR1 至 TABnCCR3）

通常在外部事件计数模式中不使用 TABnCCR1 至 TABnCCR3 寄存器。但是，TABnCCR1 至 TABnCCR3 寄存器的设置值仍然被传输给 CCR1 至 CCR3 缓冲寄存器。当 16 位计数器的计数值和 CCR1 至 CCR3 缓冲寄存器的值一致时，会产生一个比较一致中断请求信号（INTTBnCC1 至 INTTBnCC3）。

不使用 TABnCCR1 至 TABnCCR3 寄存器时，推荐将它们的值设置为 FFFFH。并且通过中断屏蔽标志(TABnCCIC1.TABnCCMK1 至 TABnCCIC3.TABnCCMK3)屏蔽这些寄存器。

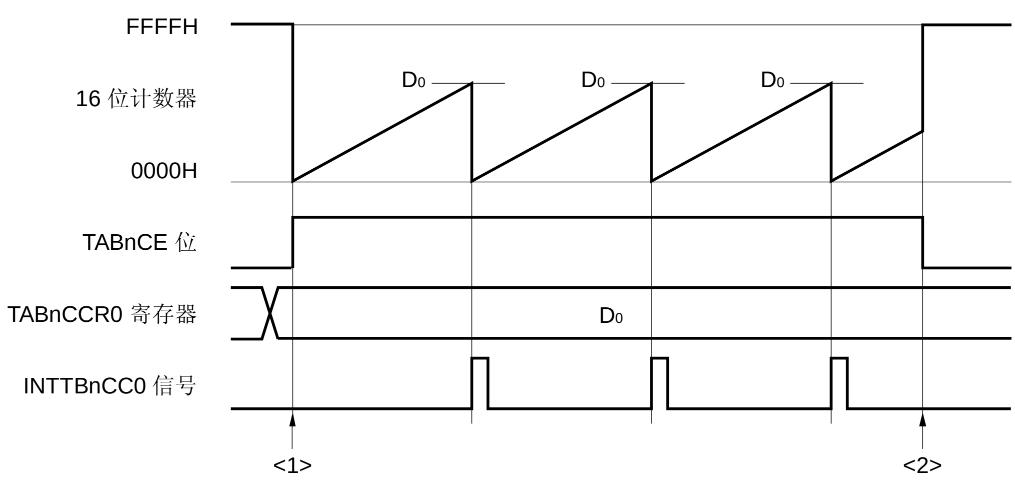
注意事项　设置 TABnIOC0 寄存器为 00H。

备注　**1.** 在外部事件计数模式下，不使用 TABn 输入/输出控制寄存器 1（TABnIOC1）和 TABn 选项寄存器 0（TABnOPT0）。

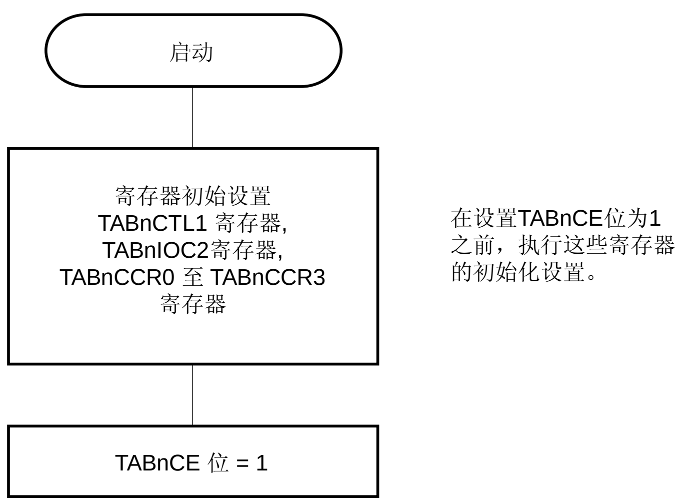
2. n = 0, 1

（1）　外部事件计数模式的操作流程

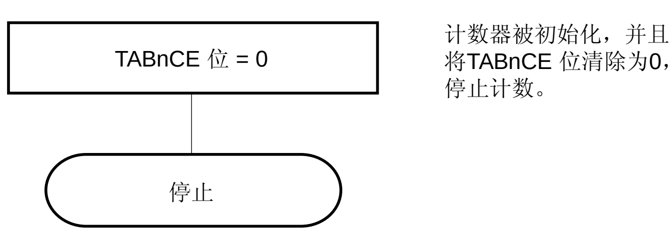
图 7-16. 外部事件计数模式下的软件处理流程



<1> 计数操作启动流程



<2> 计数操作停止流程



备注　n = 0, 1

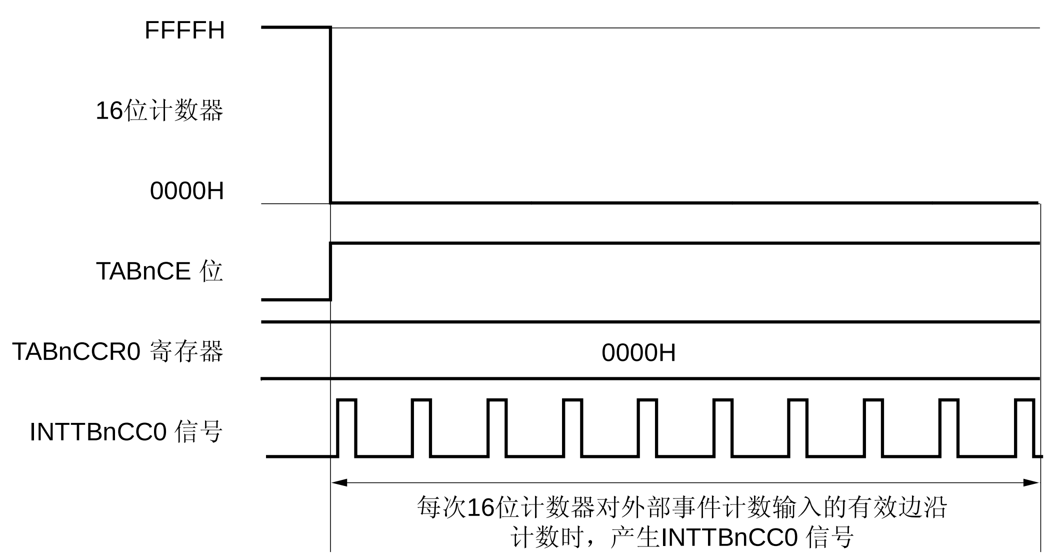
（**2**）　外部事件计数模式的操作时序

注意事项　　在外部事件计数器模式下，禁止使用定时器输出（**TOBn0** 至 **TOBn3**）。如果 **TOBn0** 至 **TOBn3** 引脚用于配合外部事件计数输入(EVTBn)，设置间隔定时器模式，并为计数时钟选择外部事件计数输入操作使能（**TABnCTL1.TABnEEE 位= 1**）（参见 7.6.1（3） 外部事件计数输入(EVTBn)的操作）。

(a) 如果 **TABnCCR0** 寄存器被设置为 **FFFFH** 时的操作

当 TABnCCR0 寄存器被设置为 0000H 时，每当检测到外部事件计数信号的有效边沿数量与 CCR0 缓冲寄存器的值一致时，16 位计数器被清除为 0000H。

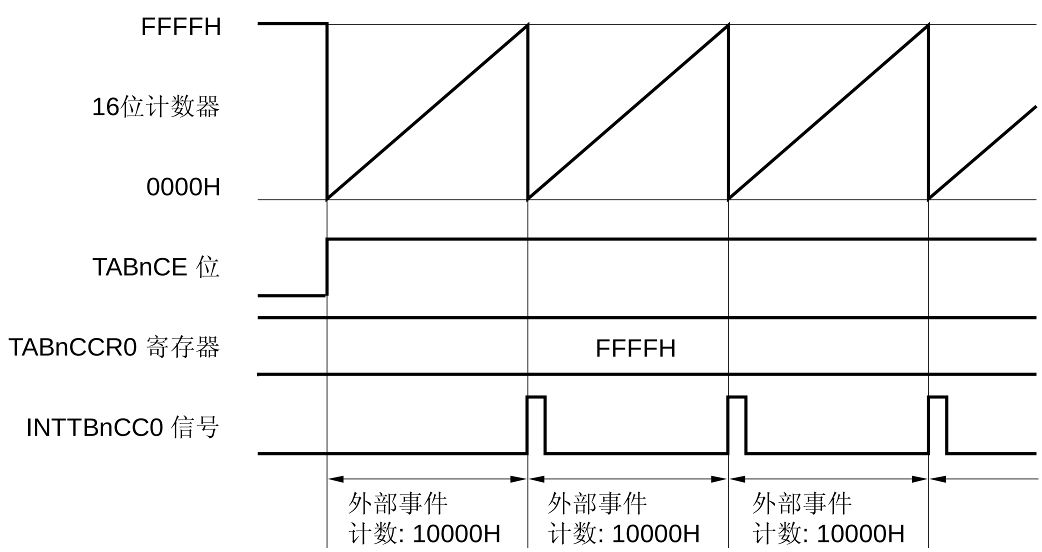
16 位计数器的值总是 0000H。



备注　　n = 0, 1

(b) 如果 TABnCCR0 寄存器被设置为 FFFFH 时的操作

如果 TABnCCR0 寄存器被设置为 FFFFH，16 位计数器每次检测到的外部事件输入信号有效边沿时，都计数到 FFFFH。当下一个计数到来时，16 位计数器被同步清除为 0000H，并且产生 INTTBnCC0 信号。此时，不设置 TABnOPT0.TABnOVF 位。

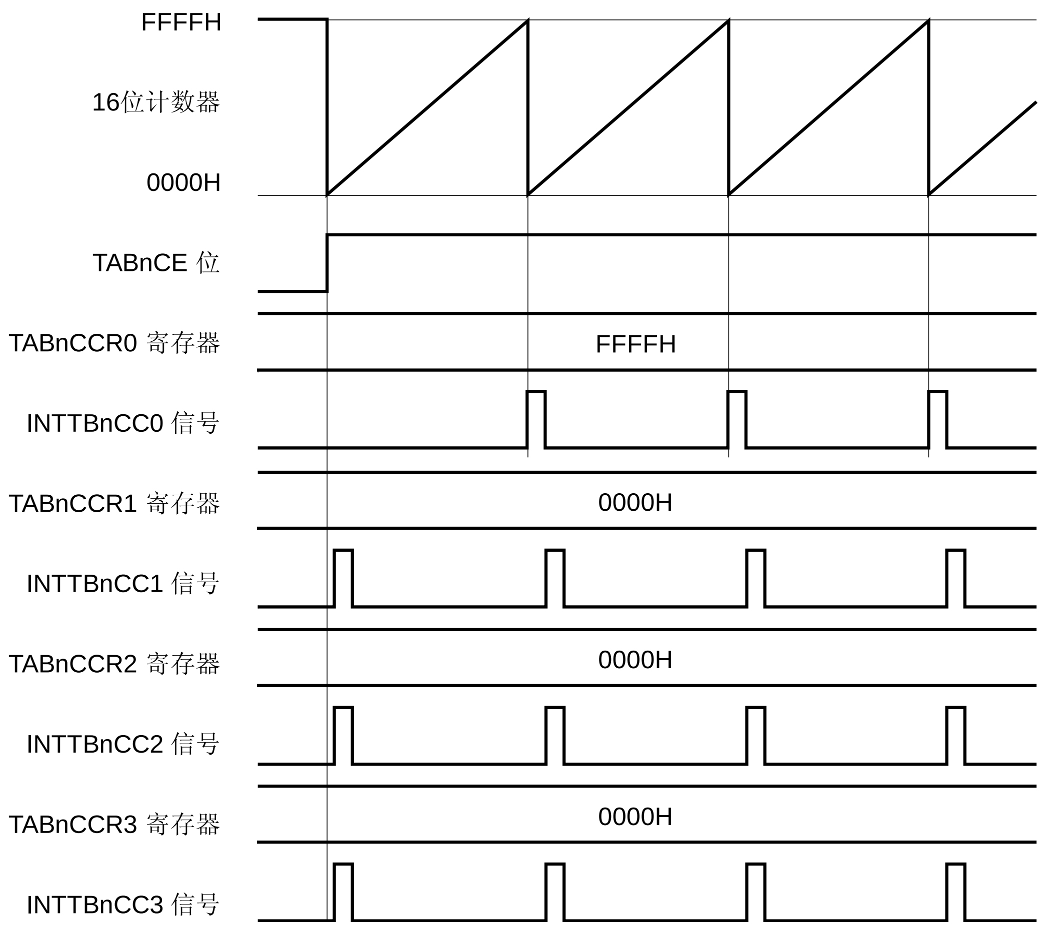


备注　n = 0, 1

(c)　TABnCCR0 寄存器被设置为 FFFFH 且 TABnCCRb 寄存器被设置为 0000H 的操作

当 TABnCCR0 寄存器被设置为 FFFFH 时，每次检测到外部事件计数信号的有效边沿时，16 位计数器向上计数到 FFFFH。16 位计数器在下一个向上计数时刻被同步清零，产生 INTTBnCC0 信号。此时，TABnOPT0.TABnOVF 位也不会置位。

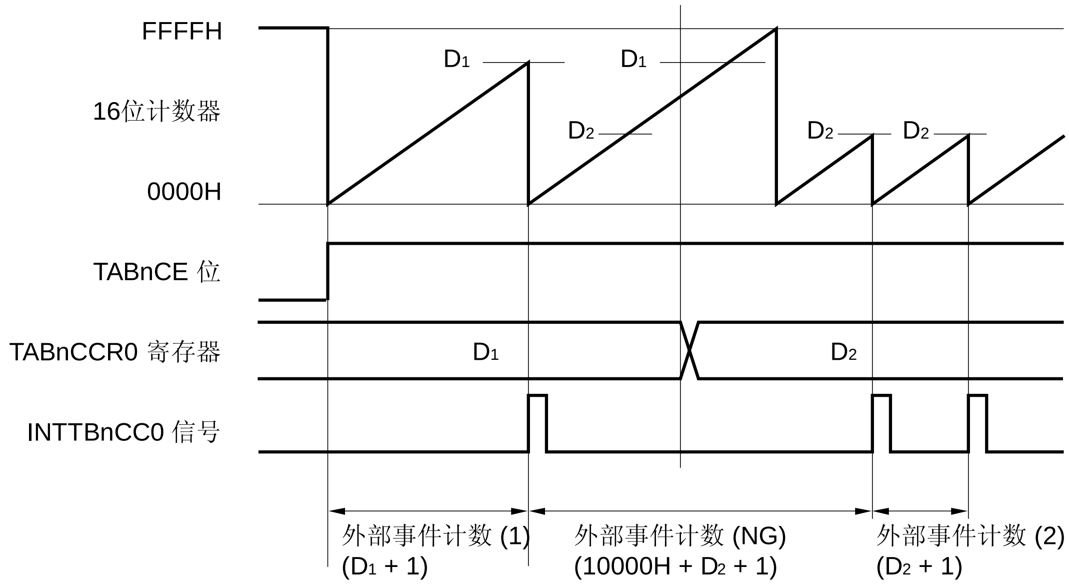
如果 TABnCCRb 寄存器被设置为 0000H，当 16 位计数器被清除为 0000H 时，产生 INTTBnCCb 信号。



备注　n = 0, 1
　　　b = 1 至 3

(d) 重写 TABnCCR0 寄存器的注意事项

如果在计数器工作期间，将 TABnCCR0 寄存器的值重写为比当前值较小的值，16 位计数器可能会溢出。当可能发生溢出时，停止计数，然后改变设置值。



备注　n = 0, 1

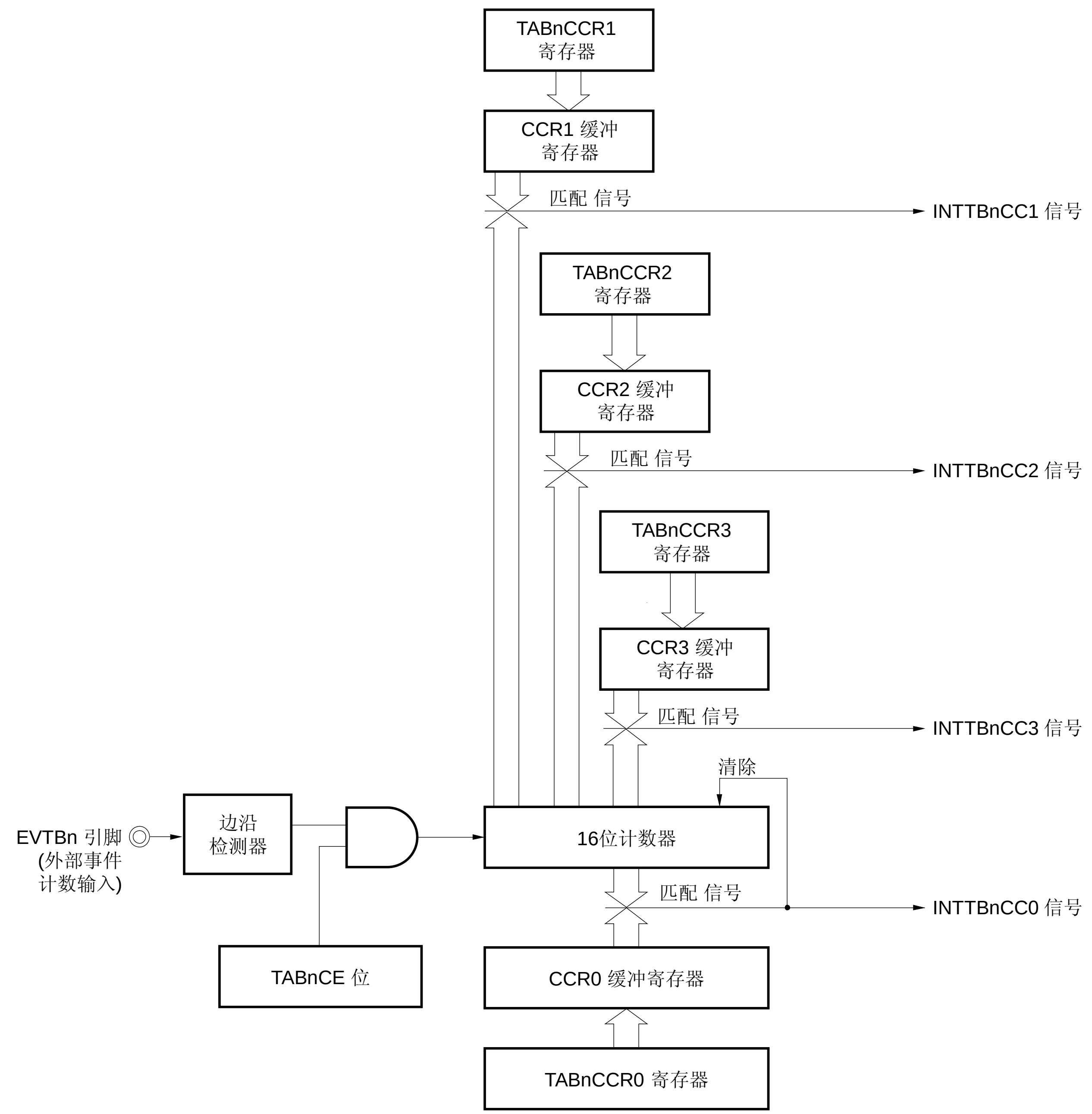
当计数值大于 D_2 而小于 D_1 时，如果将 TABnCCR0 寄存器的值从 D_1 改变为 D_2，则在重写 TABnCCR0 寄存器后，计数值会立即传输给 CCR0 缓冲寄存器。因而，与 16 位计数器相比较的值会是 D_2。

但是，由于计数值已经超过了 D_2，所以 16 位计数器就会向上累积计数到 FFFFH，溢出，然后从 0000H 开始重新计数。当计数值与 D_2 一致时，会生成 INTTBnCC0 信号。

因此，可能不会在原先预计的"（D_1＋1）次"或"（D_2＋1）次"有效边沿生成 INTTBnCC0 中断请求信号，而可能在"（10000H＋D_2＋1）次"有效边沿生成中断请求信号。

(e) TABnCCR1 至 TABnCCR3 寄存器的操作

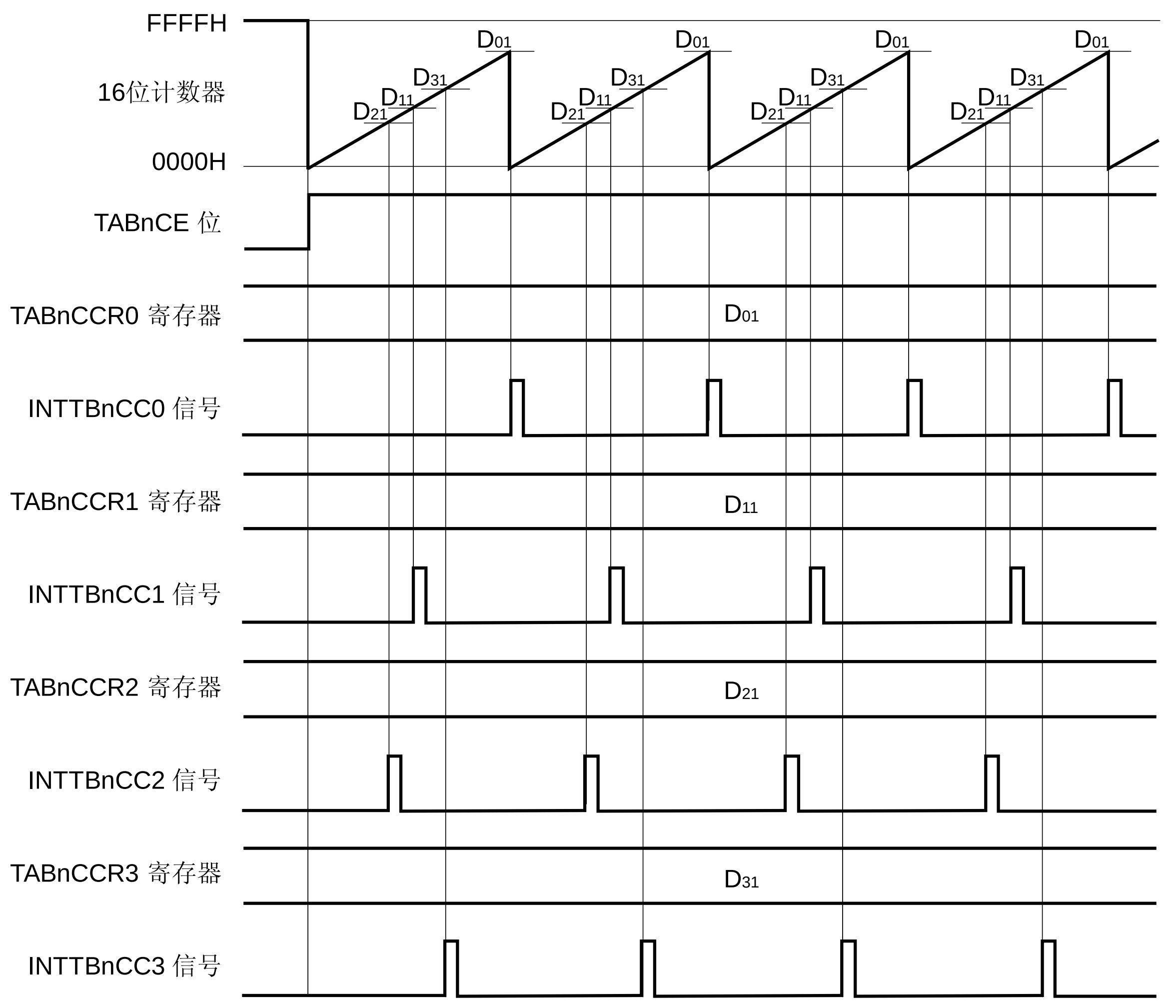
图 7-17. TABnCCR1 至 TABnCCR3 寄存器的配置



备注 n = 0, 1

如果 TABnCCRb 寄存器的设置值小于 TABnCCR0 寄存器的设置值，则每个计数周期都产生一次 INTTBnCCb 中断请求信号。

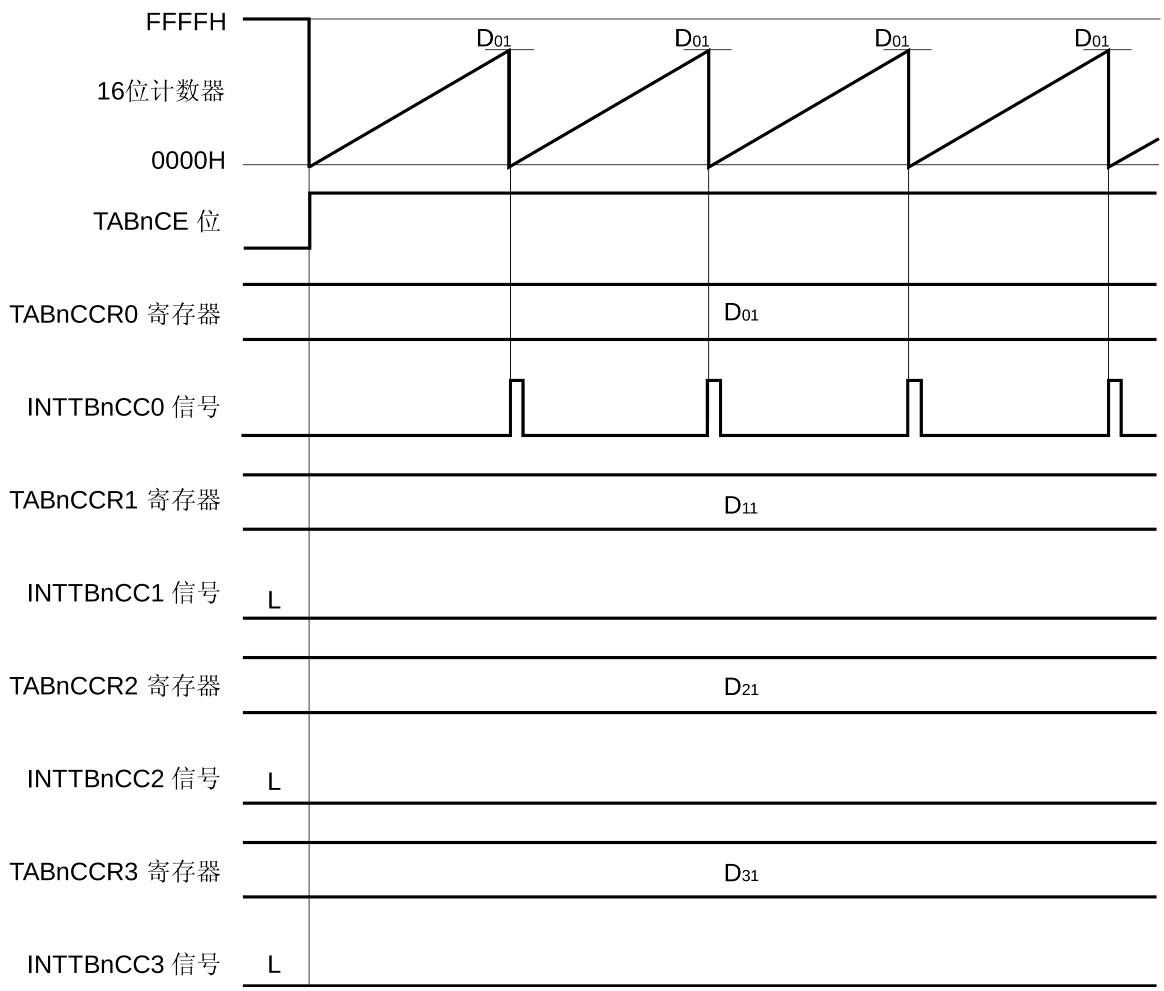
图 7-18. 当 $D_{01} \geq D_{b1}$ 时的时序图



备注 n = 0, 1
b = 1 至 3

如果 TABnCCRb 寄存器的设置值大于 TABnCCR0 寄存器的设置值，16 位计数器的计数值不会和 TABnCCRb 寄存器的值发生一致。因此也就不会产生 INTTABnCCb 中断请求信号。

图 7-19. 当 D_{01} < D_{b1} 时的时序图



备注　n = 0, 1
　　　b = 1 至 3

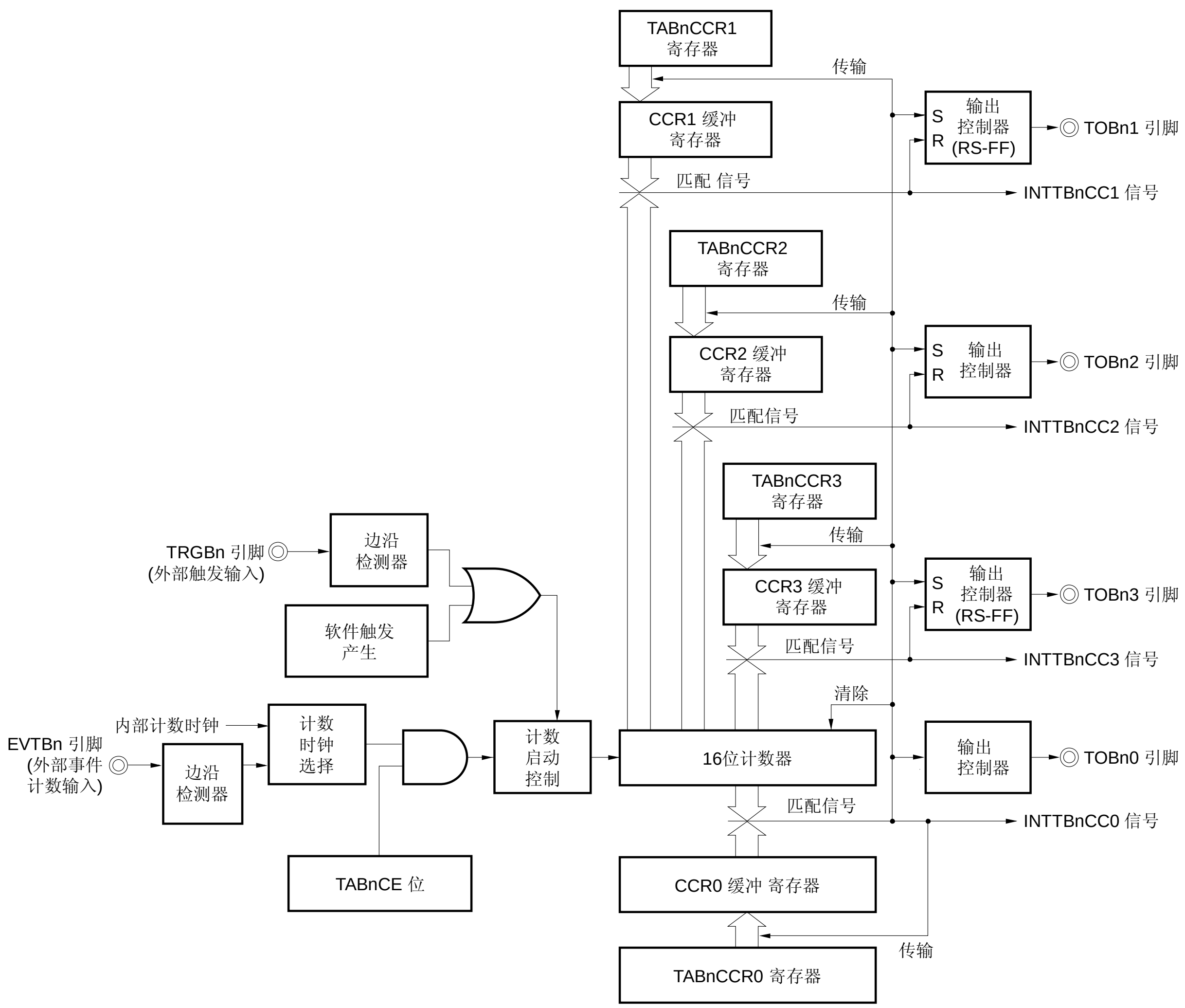
7.6.3　外部触发脉冲输出模式（TABnMD2 至 TABnMD0 位＝010）

在外部触发脉冲输出模式下，当 TABnCTL0.TABnCE 位被置为 1 时，16 位定时器/事件计数器 AB 等待触发。检测到外部触发输入信号(TRGBn)的有效边沿时，16 位定时器/事件计数器 AB 开始计数，并从 TOBn1 至 TOBn3 引脚输出 PWM 波形（最多 3 相）。

可以从 TOBn0 引脚输出一个占空比 50%的 PWM 波形，其半周期等于 TABnCCR0 寄存器的设置值 + 1。

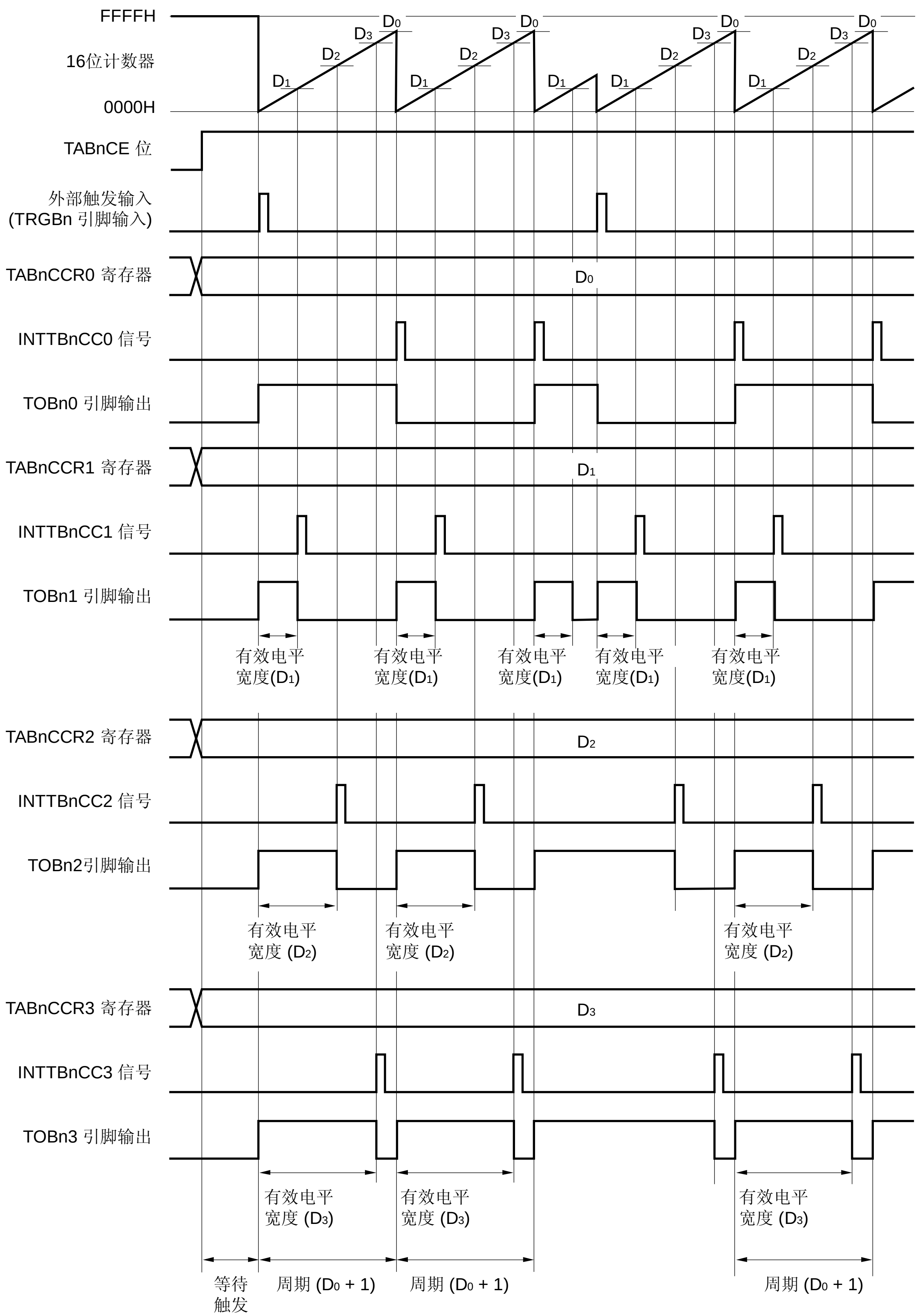
通过产生一个软件触发，而不是使用外部触发，也可以输出脉冲。

图 7-20. 外部触发脉冲输出模式的配置



备注　n = 0, 1

图 7-21. 外部触发脉冲输出模式的基本时序



备注　n = 0, 1

当 TABnCE 位设置为 1 时，16 位定时器/事件计数器 AB 等待触发。当触发产生时，16 位计数器从 FFFFH 清零为 0000H，同时开始计数，并从 TOBnb 引脚输出一个 PWM 波形。如果在计数运行中再次产生触发，则计数器清除为 0000H 并重新开始计数。（TOBn0 引脚输出反相。在产生触发时，不管当前状态如何（高/低），TOBnb 引脚都会输出一个高电平。

PWM 波形的有效电平宽度、周期、占空比系数的计算如下。

有效电平宽度＝（TABnCCRb 寄存器的设置值）× 计数时钟周期
周期＝（TABnCCR0 寄存器的设置值＋1）× 计数时钟周期
占空比系数＝（TABnCCRb 寄存器的设置值）/（TABnCCR0 寄存器的设置值＋1）

当 16 位计数器在其计数值与 CCR0 缓冲寄存器的值一致之后进行下一次计数时就会同步产生比较一致（中断）请求信号 INTTBnCC0，且 16 位计数器清零为 0000H。当 16 位计数器的计数值与 CCRb 缓冲寄存器的值一致时，会生成比较一致中断请求信号 INTTBnCCb。

当 16 位计数器的计数值与 CCR0 缓冲寄存器的值一致，而且 16 位计数器清零为 0000H 时，TABnCCRa 寄存器的设置值就被传输到 CCRa 缓冲寄存器。

外部触发输入信号的有效边沿(TRGBn)，或将软件触发（TABnCTL1.TABnEST 位）设为 1，都可以用作触发信号。

备注 n = 0, 1
a = 0 至 3
b = 1 至 3

图 7-22. 外部触发脉冲输出模式的寄存器设置（1/3）

（a） TABn 控制寄存器 0（TABnCTL0）

	TABnCE					TABnCKS2	TABnCKS1	TABnCKS0
TABnCTL0	0/1	0	0	0	0	0/1	0/1	0/1

TABnCKS2–TABnCKS0：选择计数时钟[注]

TABnCE：0: 停止计数
1: 使能计数

注 当 TABnCTL1.TABnEEE 位 = 1 时，该设置无效。

图 7-22. 外部触发脉冲输出模式的寄存器设置（2/3）

（b） TABn 控制寄存器 1（TABnCTL1）

		TABnEST	TABnEEE			TABnMD2	TABnMD1	TABnMD0
TABnCTL1	0	0/1	0/1	0	0	0	1	0

TABnMD2, TABnMD1, TABnMD0:
0, 1, 0:
外部触发脉冲
输出模式

TABnEEE:
0: 计数时钟由 TABnCKS0 至 TABnCKS2 位选择
1: 由外部事件计数输入信号计数

TABnEST:
当写入 1 时，
产生软件触发信号

（c） TABn 输入/输出控制寄存器 0（TABnIOC0）

	TABnOL3	TABnOE3	TABnOL2	TABnOE2	TABnOL1	TABnOE1	TABnOL0	TABnOE0
TABnIOC0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TABnOE0:
0: 禁止 TOBn0 引脚输出
1: 使能 TOBn0 引脚输出

TABnOL0:
当等待外部触发时
设置 TOBn0 引脚输出电平
0: 低电平
1: 高电平

TABnOE1:
0: 禁止 TOBn1 引脚输出
1: 使能 TOBn1 引脚输出

TABnOL1:
当等待外部触发时
设置 TOBn1 引脚输出电平
0: 低电平
1: 高电平

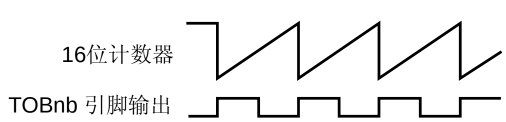
TABnOE2:
0: 禁止 TOBn2 引脚输出
1: 使能 TOBn2 引脚输出

TABnOL2:
当等待外部触发时
设置 TOBn2 引脚输出电平
0: 低电平
1: 高电平

TABnOE3:
0: 禁止 TOBn3 引脚输出
1: 使能 TOBn3 引脚输出

TABnOL3:
当等待外部触发时
设置 TOBn3引脚输出电平
0: 低电平
1: 高电平

- 当 TABnOLb 位 = 0



- 当 TABnOLb 位 = 1

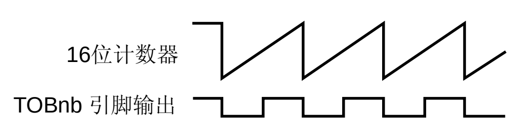


图 **7-22.** 外部触发脉冲输出模式的寄存器设置（**3/3**）

（d） TABn 输入/输出控制寄存器 2（TABnIOC2）

					TABnEES1	TABnEES0	TABnETS1	TABnETS0
TABnIOC2	0	0	0	0	0/1	0/1	0/1	0/1

TABnETS1、TABnETS0：选择外部触发输入的有效边沿(TRGBn 引脚)

TABnEES1、TABnEES0：选择外部事件计数输入的有效边沿(EVTBn 引脚)

（e） TABn 计数器读取缓冲寄存器（TABnCNT）

通过读取 TABnCNT 寄存器，可以读出 16 位计数器的值。

（f） TABn 捕获/比较寄存器 0 至 3（TABnCCR0 至 TABnCCR3）

如果将 TABnCCR0 寄存器的值设置为 D_0，将 TABnCCR1 寄存器的值设置为 D_1，将 TABnCCR2 寄存器的值设置为 D_2，将 TABnCCR3 寄存器的值设置为 D_3，则 PWM 波形的周期和有效电平的计算如下：

周期＝（D_0＋1）× 计数时钟周期
TOBn1 引脚 PWM 波形有效电平宽度＝D_1 × 计数时钟周期
TOBn2 引脚 PWM 波形有效电平宽度＝D_2 × 计数时钟周期
TOBn3 引脚 PWM 波形有效电平宽度＝D_3 × 计数时钟周期

备注 **1.** 在外部触发脉冲输出模式下，不使用 TABn 输入/输出控制寄存器 1（TABnIOC1）和 TABn 选项寄存器 0（TABnOPT0）。

2. n = 0, 1
b = 1 至 3

（1）外部触发脉冲输出模式的操作流程

图 7-23. 外部触发脉冲输出模式的软件处理流程（1/2）

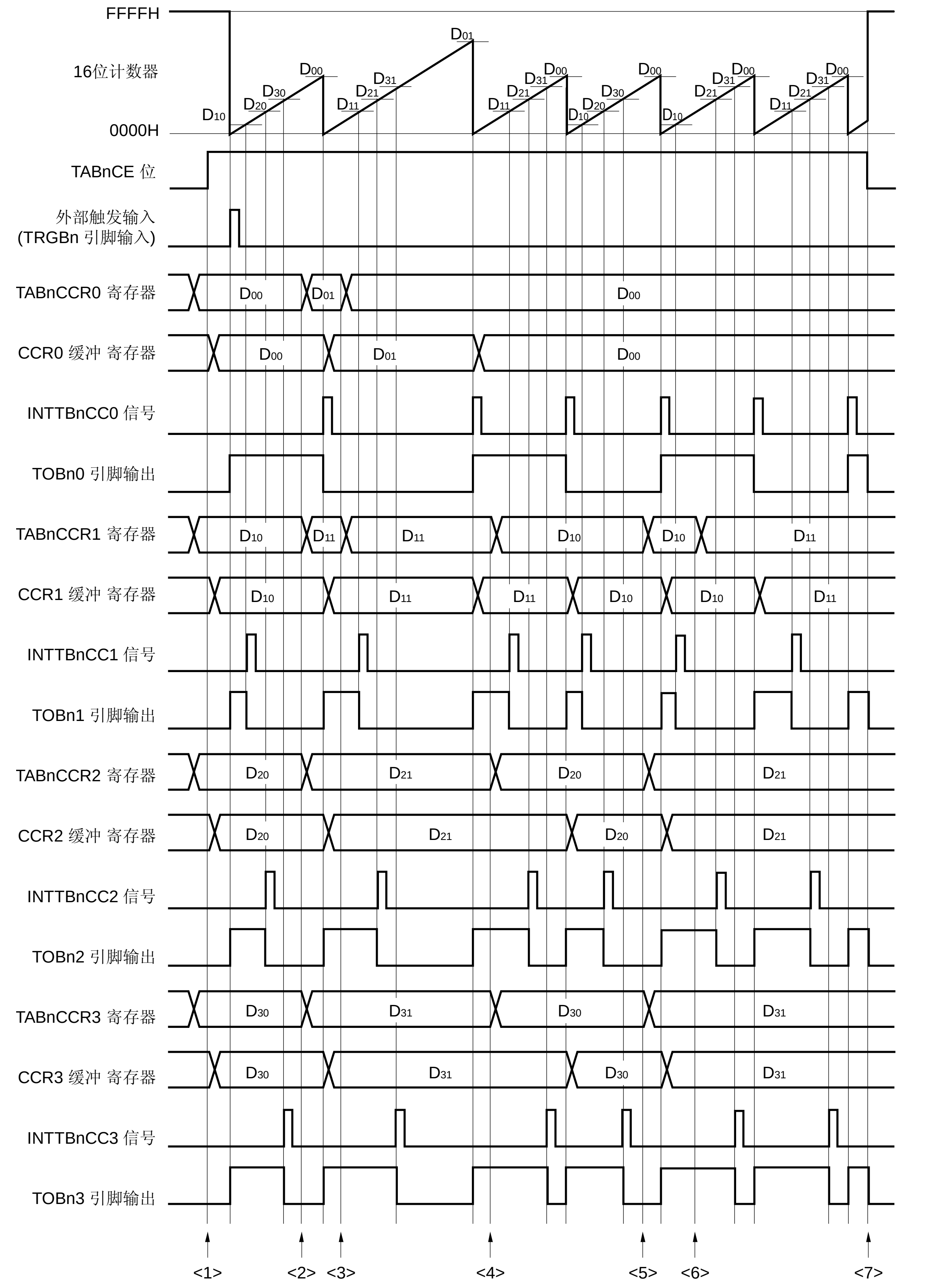
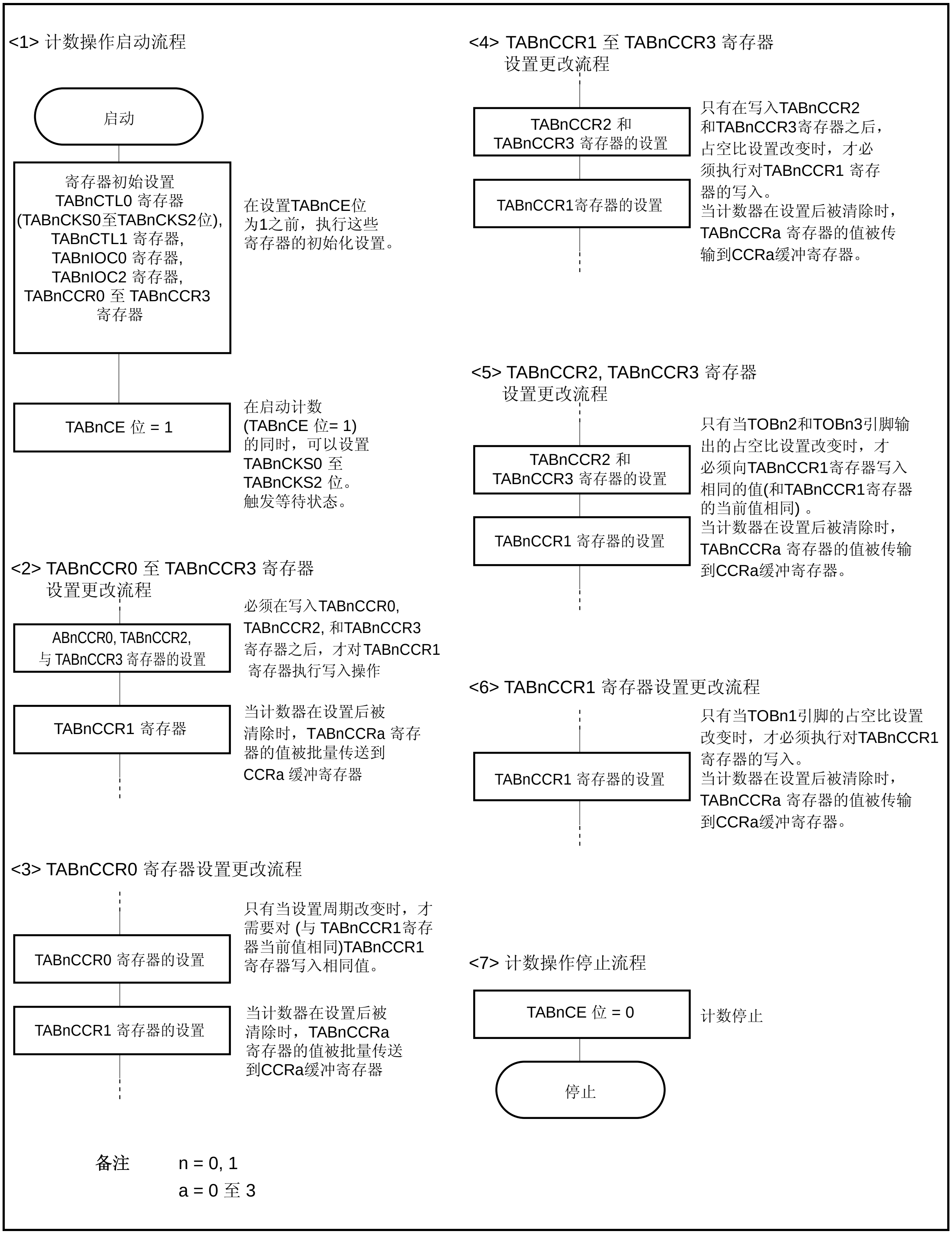


图 **7-23.** 外部触发脉冲输出模式的软件处理流程（**2/2**）



（2）外部触发脉冲输出模式的操作时序

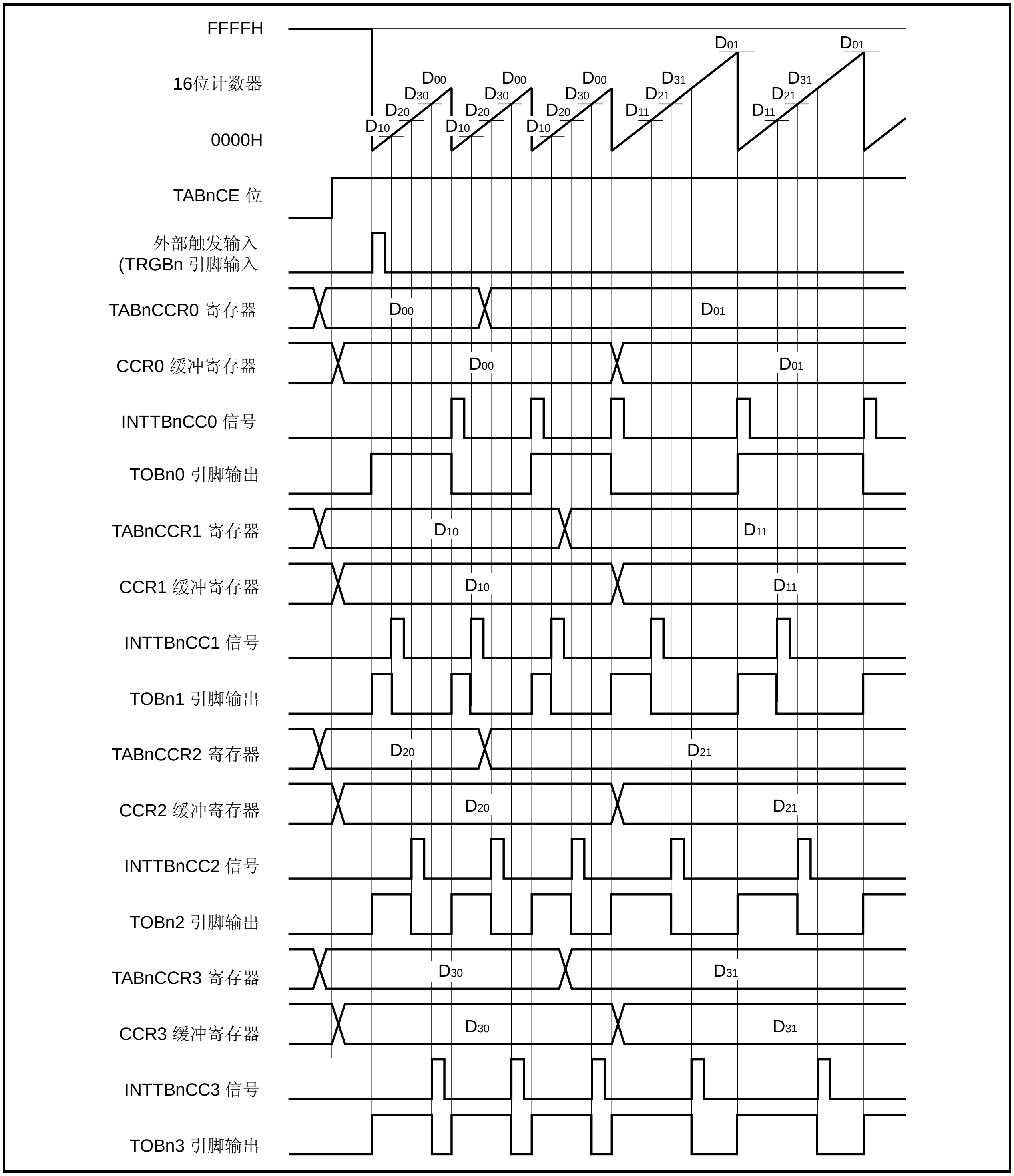
（a） 有关操作中改变脉冲宽度的注意事项

计数器计数过程中，若要改变 PWM 波形，应重新写入 TABnCCR1 寄存器。

当检测到 INTTBnCC0 信号以后，在 TABnCCR1 寄存器写入操作之后重写 TABnCCRb 寄存器。

备注　　n = 0, 1

　　　　b = 1 至 3



为了将数据从 TABnCCRa 寄存器传输到 CCRa 缓冲寄存器，必须写入 TABnCCR1 寄存器。

如果 PWM 波形的周期和有效电平宽度都要改变，先将周期设定到 TABnCCR0 寄存器，将有效电平宽度设定到 TABnCCR2 和 TABnCCR3 寄存器，然后将有效电平设定到 TABnCCR1 寄存器。

如果只改变 PWM 波形的周期，应先把周期设置到 TABnCCR0 寄存器，然后再将相同的值（与 TABnCCR1 寄存器的当前值相同）写入 TABnCCR1 寄存器。

如果只改变 PWM 波形的有效电平宽度（占空比系数），需要先向 TABnCCR2 和 TABnCCR3 寄存器设置有效电平，然后再向 TABnCCR1 寄存器设置有效电平。

如果只改变 TOBn1 引脚输出的 PWM 波形的有效电平宽度（占空比系数），仅需要设置 TABnCCR1 寄存器。

如果只改变 TOBn2 和 TOBn3 引脚输出的 PWM 波形的有效电平宽度（占空比系数），要先向 TABnCCR2 和 TABnCCR3 寄存器设置有效电平宽度，然后再将相同的值（与 TABnCCR1 寄存器的当前值相同）写入 TABnCCR1 寄存器。

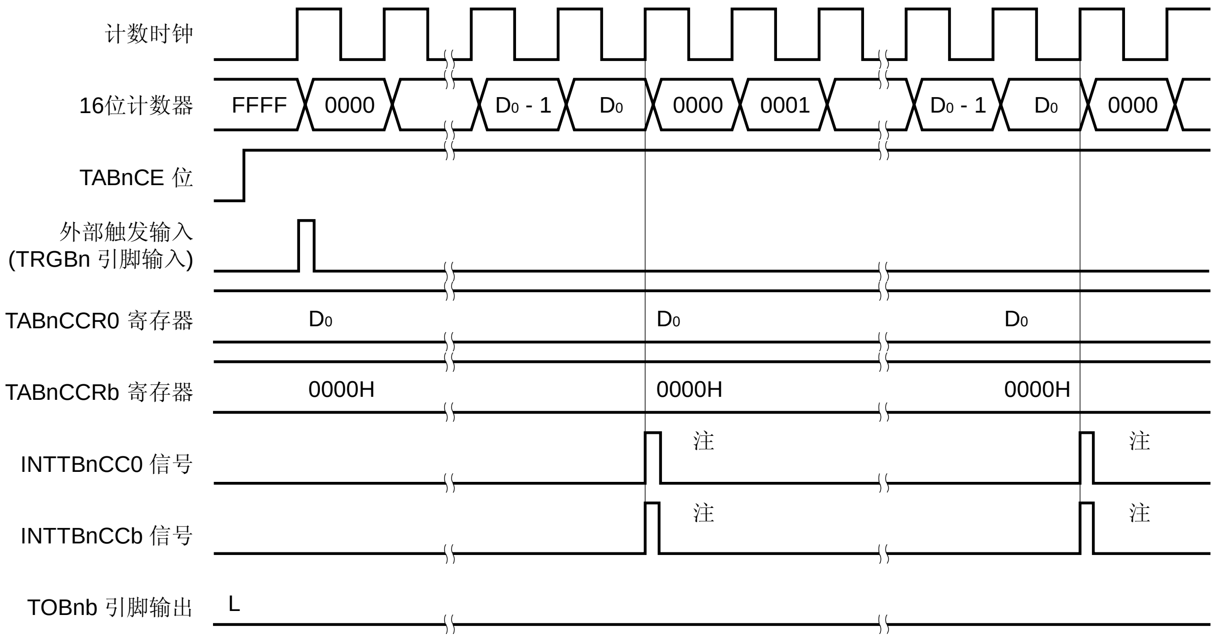
在数据写入 TABnCCR1 寄存器后，在对 16 位计数器清零的同时，写入到 TABnCCRa 寄存器的值会被传输到 CCRa 缓冲寄存器，并作为与 16 位计数器进行比较的值。

在写入 TABnCCR1 寄存器一次之后，要再次写入 TABnCCR0 至 TABnCCR3 寄存器，应该在 INTTBnCC0 信号产生之后进行。否则，CCRa 缓冲寄存器的值可能不确定，因为把数据从 TABnCCRa 寄存器传输到 CCRa 缓冲寄存器的时序会和写入 TABnCCRa 寄存器的时序发生冲突。

备注 n = 0, 1
a = 0 至 3

（b） **PWM 波形的 0%和 100% 输出**

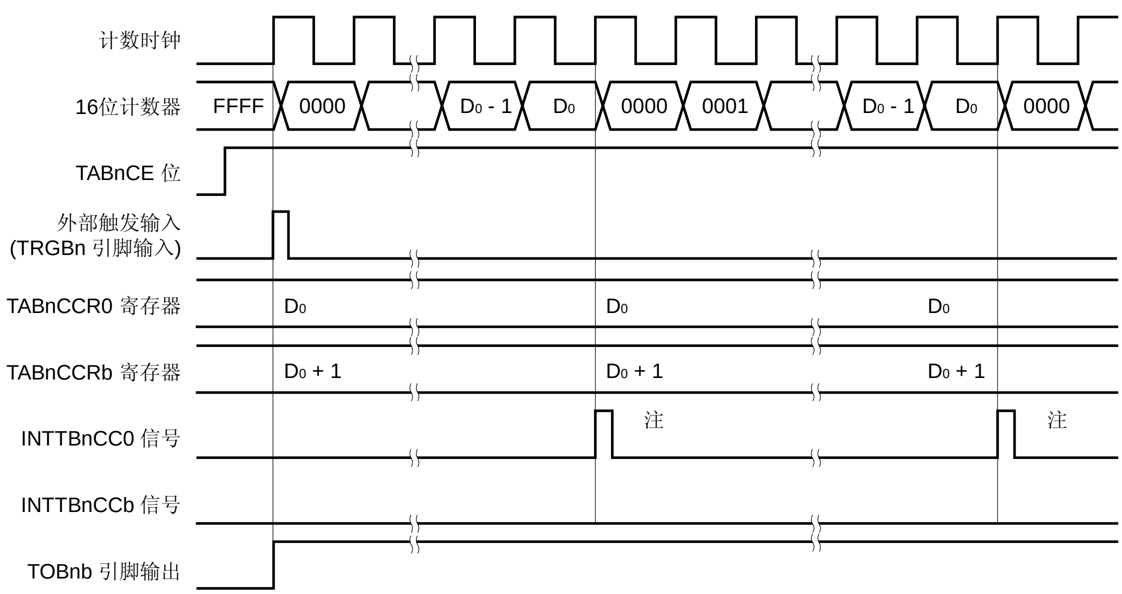
要输出一个 0% 波形，将 TABnCCRb 寄存器设置为 0000H。在 16 位计数器的计数值和 CCR0 缓冲寄存器的值发生一致之后，16 位计数器被清除为 0000H，且在下一个时序产生 INTTBnCC0 和 INTTBnCCb 信号。



注　　时序实际被延迟一个操作时钟(fxx)

备注　　n = 0, 1, b = 1 至 3

为了输出 100%的波形，需要将 TABnCCRb 寄存器的值设为（TABnCCR0 寄存器的设置值＋1）。如果 TABnCCR0 寄存器的设置值为 FFFFH，则不会产生 100%波形输出。

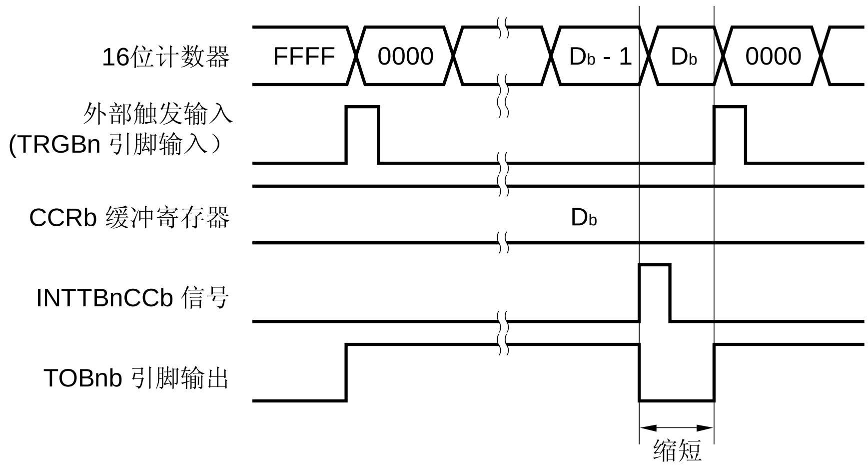


注　　时序实际被延迟一个操作时钟(fxx)

备注　　n = 0, 1, b = 1 至 3

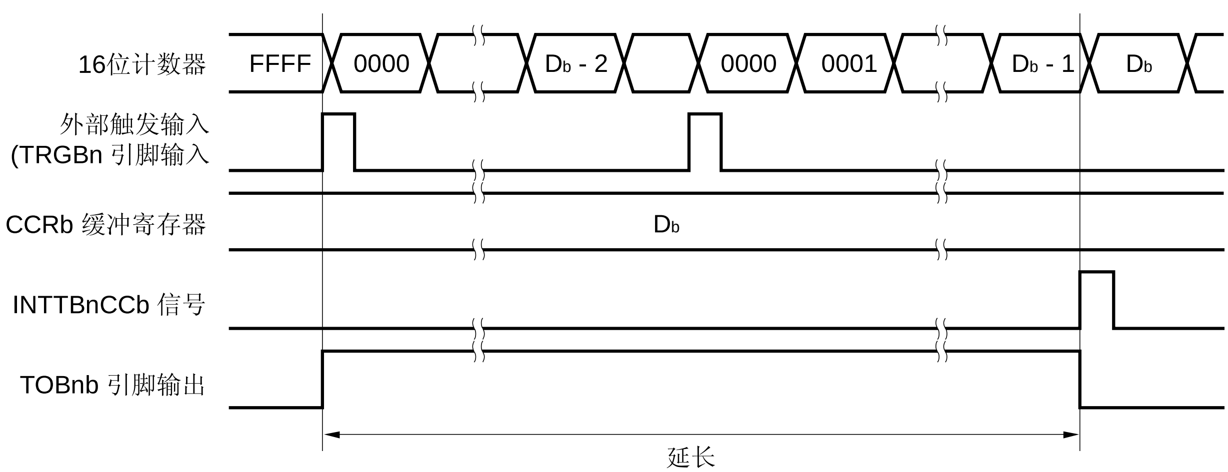
（c） 触发检测和 CCRb 缓冲寄存器一致之间的冲突

如果在 INTTBnCCb 信号产生后，立刻检测到触发信号，则 16 位计数器立刻清零为 0000H，TOBnb 引脚的输出信号生效，计数器继续计数。结果，PWM 波形的无效时段被缩短。



备注　n = 0, 1
b = 1 至 3

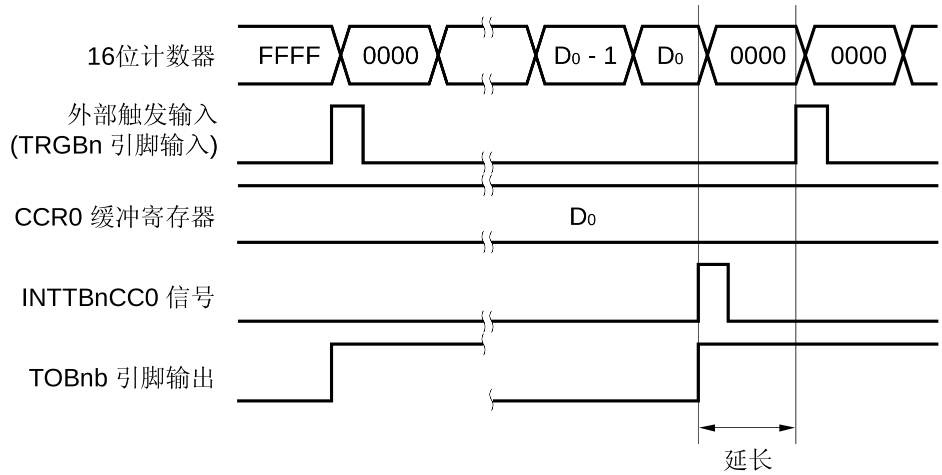
如果紧邻在 INTTBnCCb 信号产生之前检测到触发信号，则不会产生 INTTBnCCb 信号，而 16 位计数器清零为 0000H，计数继续进行。TOBnb 引脚的输出信号仍然保持有效。结果，PWM 波形的有效时段被延长。



备注　n = 0, 1
b = 1 至 3

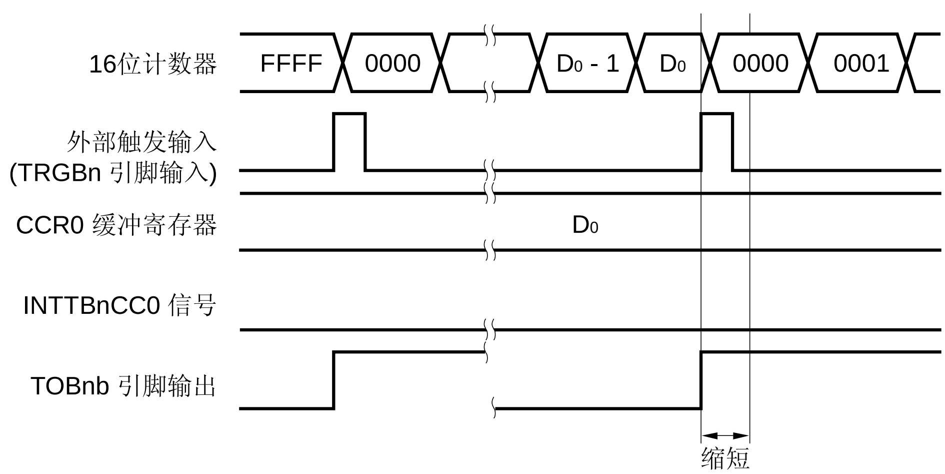
（d） 触发检测和 CCR0 缓冲寄存器一致之间的冲突

如果在 INTTBnCC0 信号产生后，立刻检测到触发信号，则 16 位计数器立刻清零为 0000H，并继续计数。因此，TOBnb 引脚的有效时段被延长，延长长度为 INTTBnCC0 信号产生时刻至检测到触发信号时刻的时间间隔。



备注 n = 0, 1
b = 1 至 3

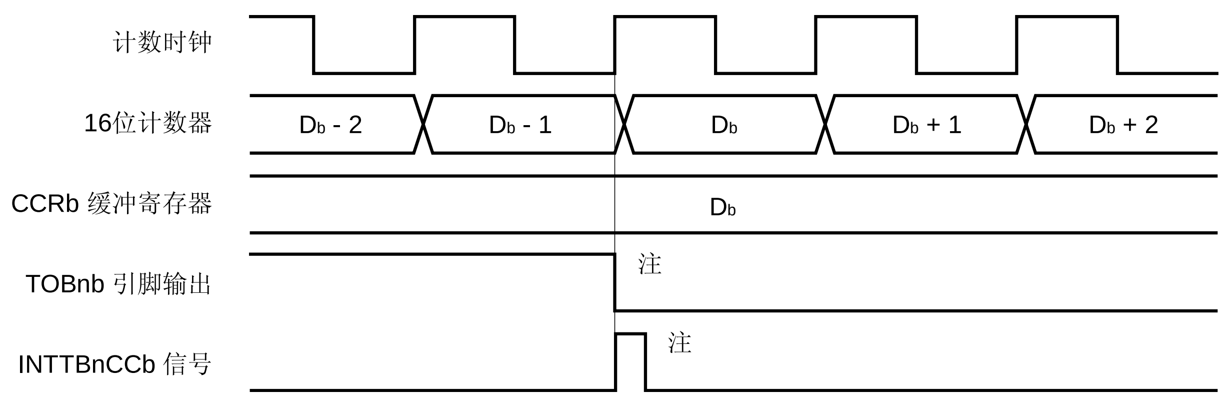
如果紧邻在 INTTBnCC0 信号产生之前检测到触发信号，则不会产生 INTTBnCC0 信号。16 位计数器清零为 0000H，TOBnb 引脚输出有效，计数器继续计数。结果，PWM 波形的无效时段被缩短。



备注 n = 0, 1
b = 1 至 3

（e） 比较一致中断请求信号（INTTBnCCb）的发生时序

在外部触发脉冲输出模式下，INTTBnCCb 信号的发生时序与其它情况下 INTTBnCCb 信号的发生时序不同；当 16 位计数器的计数值与 CCRb 缓冲寄存器的值一致时，产生 INTTBnCCb 中断请求信号。



注　时序实际被延迟一个操作时钟(f_{XX})

备注　n = 0, 1
　　　b = 1 至 3

通常在 16 位计数器的当前计数值与 CCRb 缓冲寄存器的值一致之后，在下一次计数时同步发生 INTTBnCCb 信号。

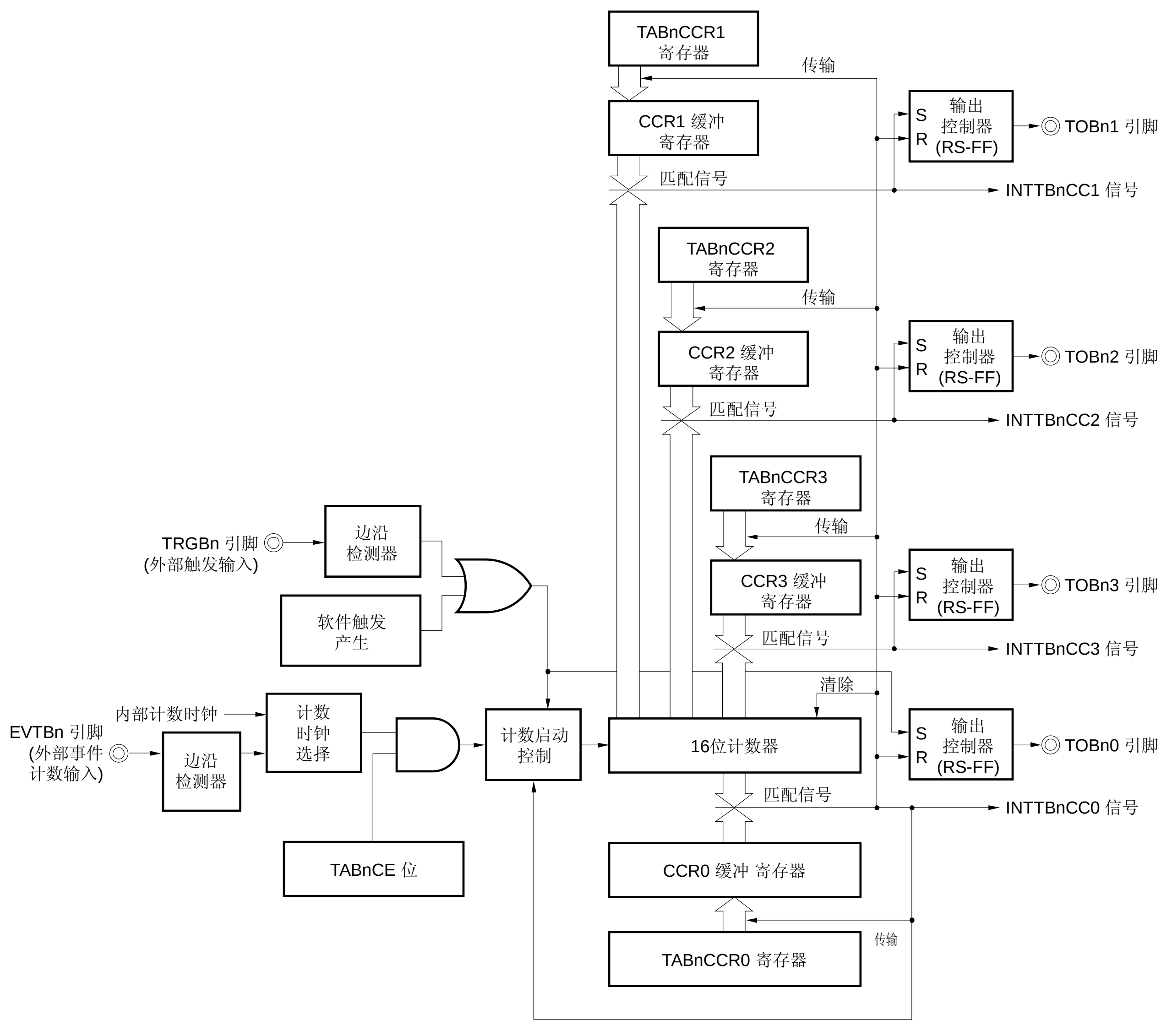
但是，在外部触发脉冲输出模式下，会提早一个计数时钟周期产生。该时序的变化是为了一致 TOBnb 引脚输出信号变化的时序。

7.6.4　单次触发脉冲输出模式（TABnMD2 至 TABnMD0 位 ＝ 011）

在单次触发脉冲输出模式下，当 TABnCTL0.TABnCE 位被置为 1 时，16 位定时器/事件计数器 AB 等待触发。在检测到外部触发输入（TRGBn）的有效边沿时，16 位定时器/事件计数器 AB 开始计数，同时从 TOBn1 至 TOBn3 引脚输出一个单次触发脉冲。在 16 位计数器计数时，TOBn0 引脚输出有效电平，在计数器停止（等待触发）时输出无效电平。

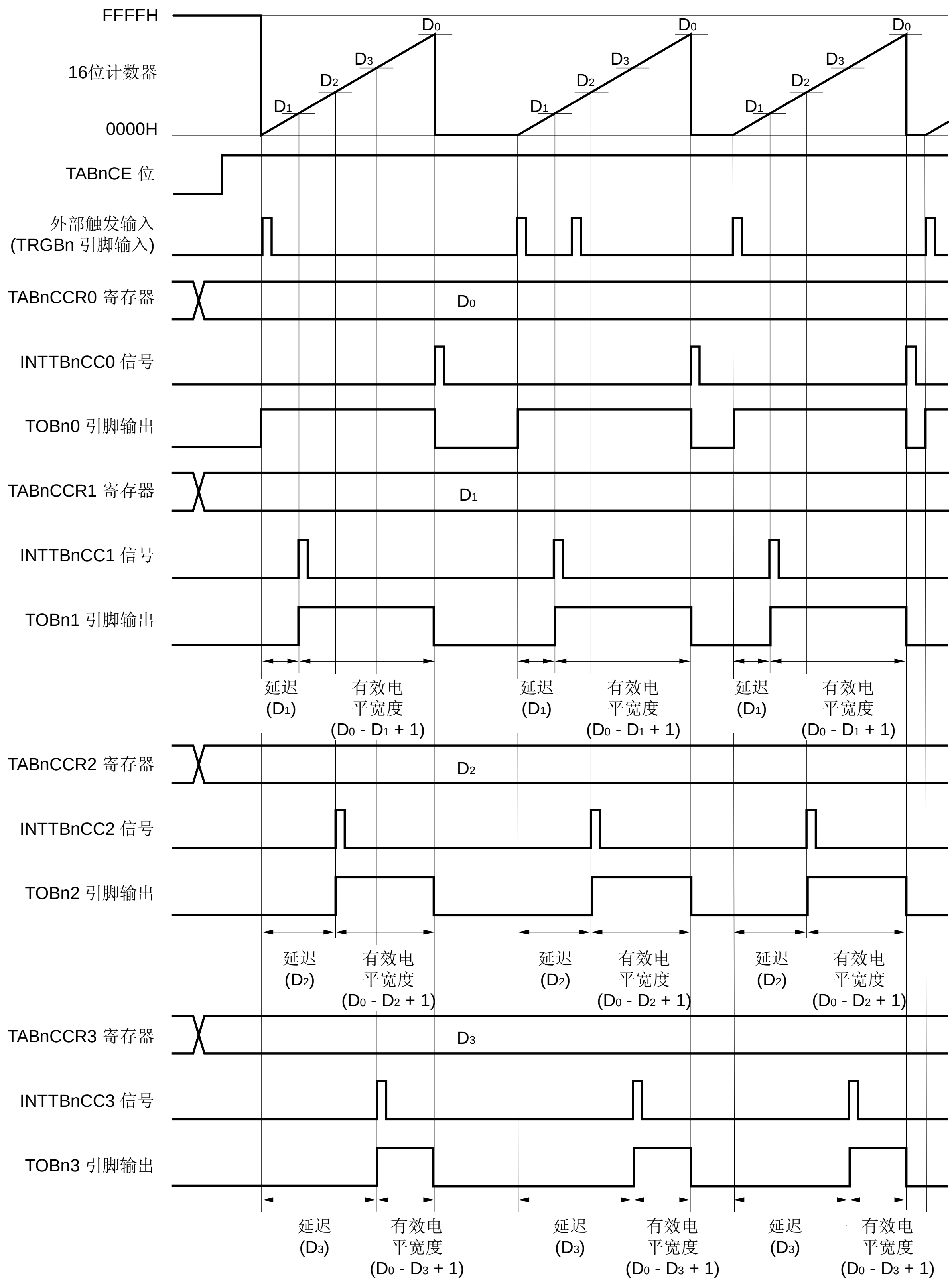
除外部触发输入外，也可以生成软件触发信号来输出该脉冲。

图 7-24. 单次触发脉冲输出模式的配置



备注　　n = 0, 1

图 7-25. 单次触发脉冲输出模式的基本时序



备注　　n = 0, 1

当 TABnCE 位置为 1 时，16 位定时器/事件计数器 AB 等待触发。在触发信号发生时，16 位计数器从 FFFFH 清零为 0000H，开始进行计数，并从 TOBnb 引脚输出一个单次触发脉冲。在输出单次触发脉冲后，16 位计数器被设置为 0000H，计数停止，并重新等待触发。如果再次产生触发信号，16 位计数器从 0000H 开始计数。如果在单次触发脉冲输出期间再次产生触发信号，则该触发信号被忽略。

单次触发脉冲的输出延迟时间和有效电平宽度可计算如下：

输出延迟时间＝（TABnCCRb 寄存器的设定值）× 计数时钟周期

有效电平宽度＝（TABnCCR0 寄存器的设定值 – TABnCCRb 寄存器的设定值 ＋1）× 计数时钟周期

当 16 位计数器的计数值与 CCR0 缓冲寄存器的值一致之后，产生比较一致中断请求信号 INTTBnCC0。当 16 位计数器的计数值与 CCRb 缓冲寄存器的值一致时，会生成比较一致中断请求信号 INTTBnCCb。

外部触发输入（TRGBn）的有效边沿或将软件触发（TABnCTL1.TABnEST 位）信号设置为 1，都可被用作触发信号。

备注　n = 0, 1，k＝1 至 3

图 7-26. 单次触发脉冲输出模式时的寄存器设置（1/3）

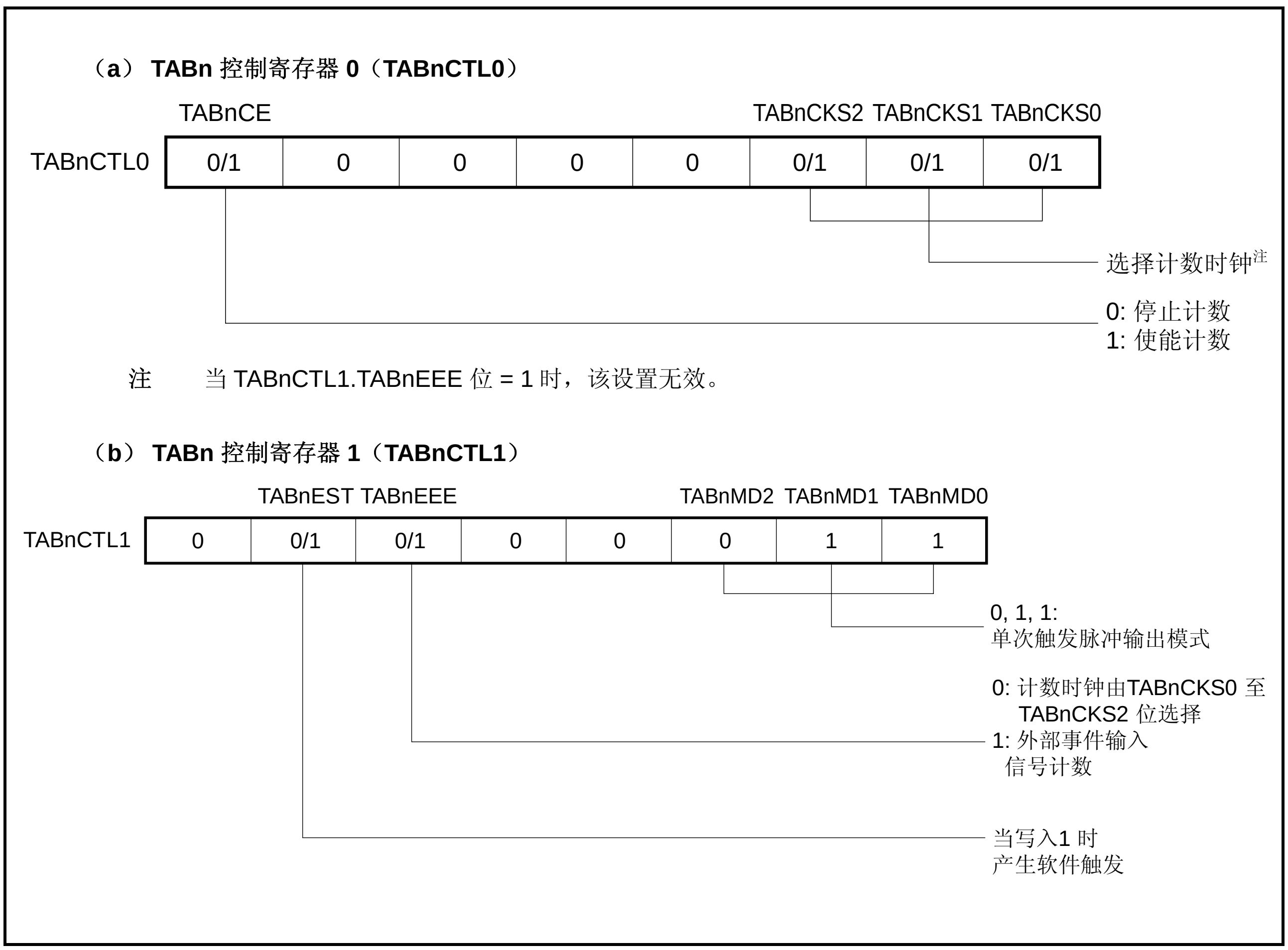


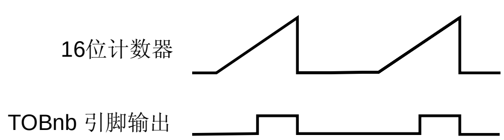
图 **7-26.** 单次触发脉冲输出模式时的寄存器设置（**2/3**）

（c） TABn I/O 控制寄存器 0（TABnIOC0）

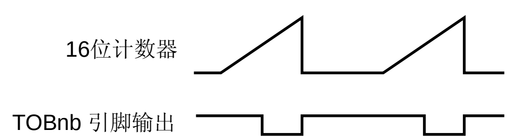
	TABnOL3	TABnOE3	TABnOL2	TABnOE2	TABnOL1	TABnOE1	TABnOL0	TABnOE0
TABnIOC0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

- TABnOE0：
  0: 禁止 TOBn0 引脚输出
  1: 使能 TOBn0 引脚输出
- TABnOL0：当等待外部触发时，TOBn0 引脚输出电平的设置
  0: 低电平
  1: 高电平
- TABnOE1：
  0: 禁止 TOBn1 引脚输出
  1: 使能 TOBn1 引脚输出
- TABnOL1：当等待外部触发时，TOBn1 引脚输出电平的设置
  0: 低电平
  1: 高电平
- TABnOE2：
  0: 禁止 TOBn2 引脚输出
  1: 使能 TOBn2 引脚输出
- TABnOL2：当等待外部触发时，TOBn2 引脚输出电平的设置
  0: 低电平
  1: 高电平
- TABnOE3：
  0: 禁止 TOBn3 引脚输出
  1: 使能 TOBn3 引脚输出
- TABnOL3：当等待外部触发时，TOBn3 引脚输出电平的设置
  0: 低电平
  1: 高电平

• 当 TABnOLb 位 = 0



• 当 TABnOLb 位 = 1



（d） TABn I/O 控制寄存器 2（TABnIOC2）

					TABnEES1	TABnEES0	TABnETS1	TABnETS0
TABnIOC2	0	0	0	0	0/1	0/1	0/1	0/1

- TABnETS1、TABnETS0：选择外部触发输入的有效边沿 (TRGBn 引脚)
- TABnEES1、TABnEES0：选择外部事件计数输入的有效边沿 (EVTBn 引脚)

（e） TABn 计数器读取缓冲寄存器（TABnCNT）

通过读取 TABnCNT 寄存器，可以读出 16 位计数器的值。

图 **7-26.** 单次触发脉冲输出模式时的寄存器设置（**3/3**）

（f） TABn 捕获/比较寄存器 0 至 3（TABnCCR0 至 TABnCCR3）

若把 TABnCCR0 寄存器的值设置为 D_0，TABnCCRb 寄存器的值设置为 D_b，则单次触发脉冲的有效电平宽度和输出延迟时间的计算如下。

有效电平宽度＝（$D_b - D_0$ ＋1）× 计数时钟周期

输出延迟时间＝（D_b） × 计数时钟周期

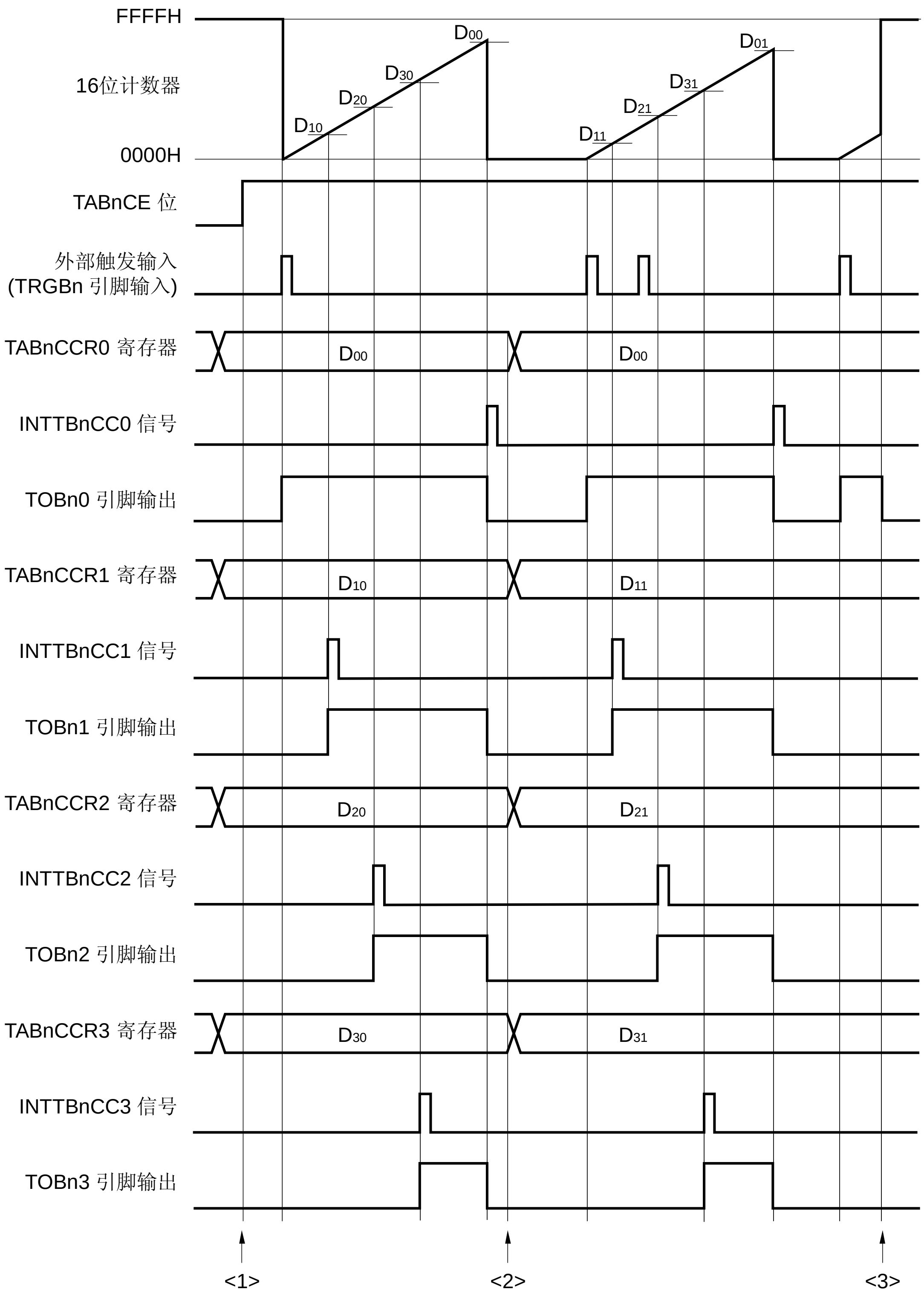
注意事项　如果 TABnCCRb 寄存器中设置的值大于 TABnCCR0 寄存器中设置的值，那么，即使在单次触发脉冲输出模式下也不会输出单次触发脉冲。

备注　**1.** 在单次触发脉冲输出模式下，不使用 TABn I/O 控制寄存器 1（TABnIOC1）和 TABn 选项寄存器 0（TABnOPT0）。

2. n = 0, 1
b = 1 至 3

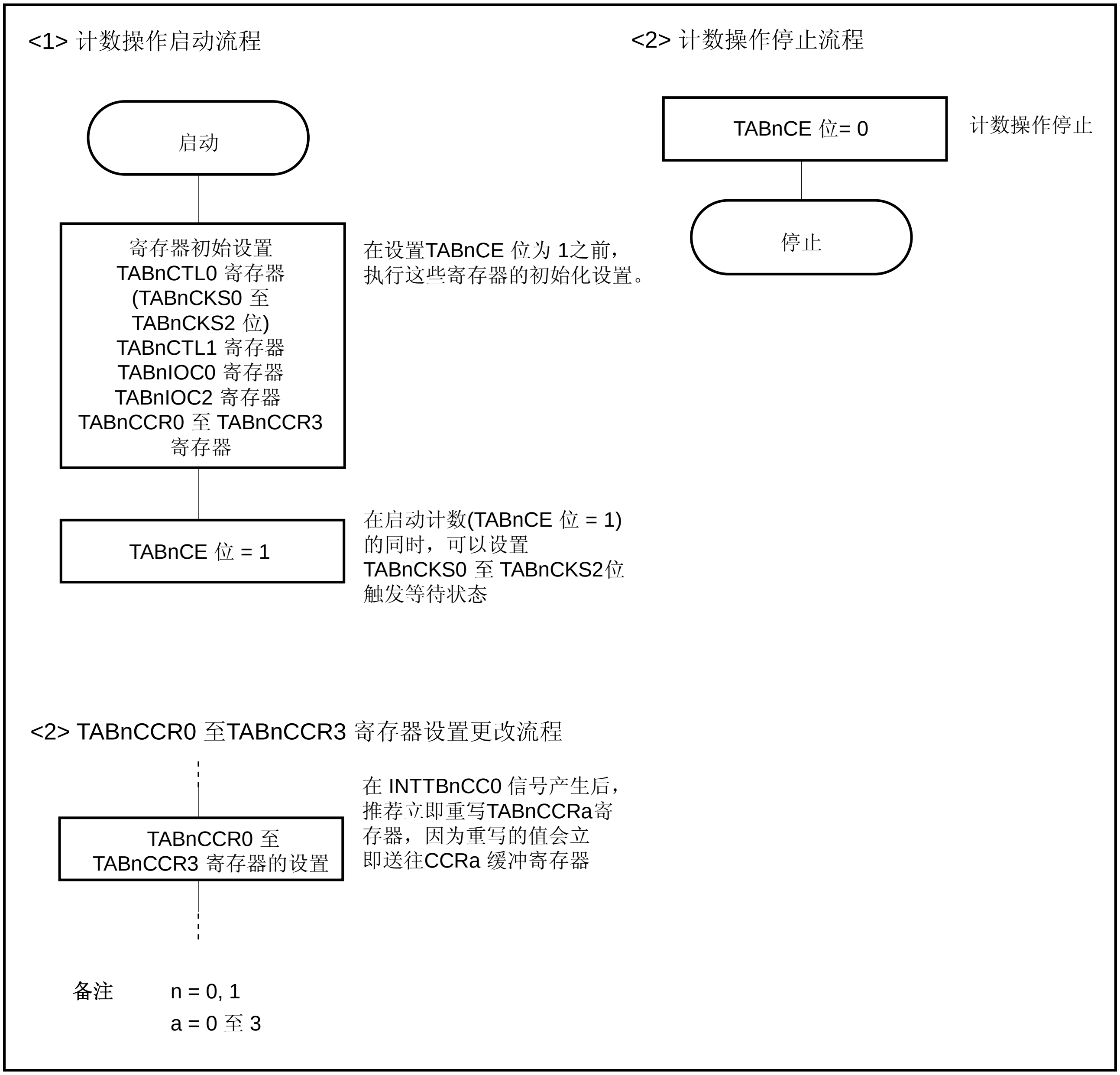
（1）单次触发脉冲输出模式下的操作流程

图 7-27. 单次触发脉冲输出模式的软件处理流程（1/2）



备注 n = 0, 1

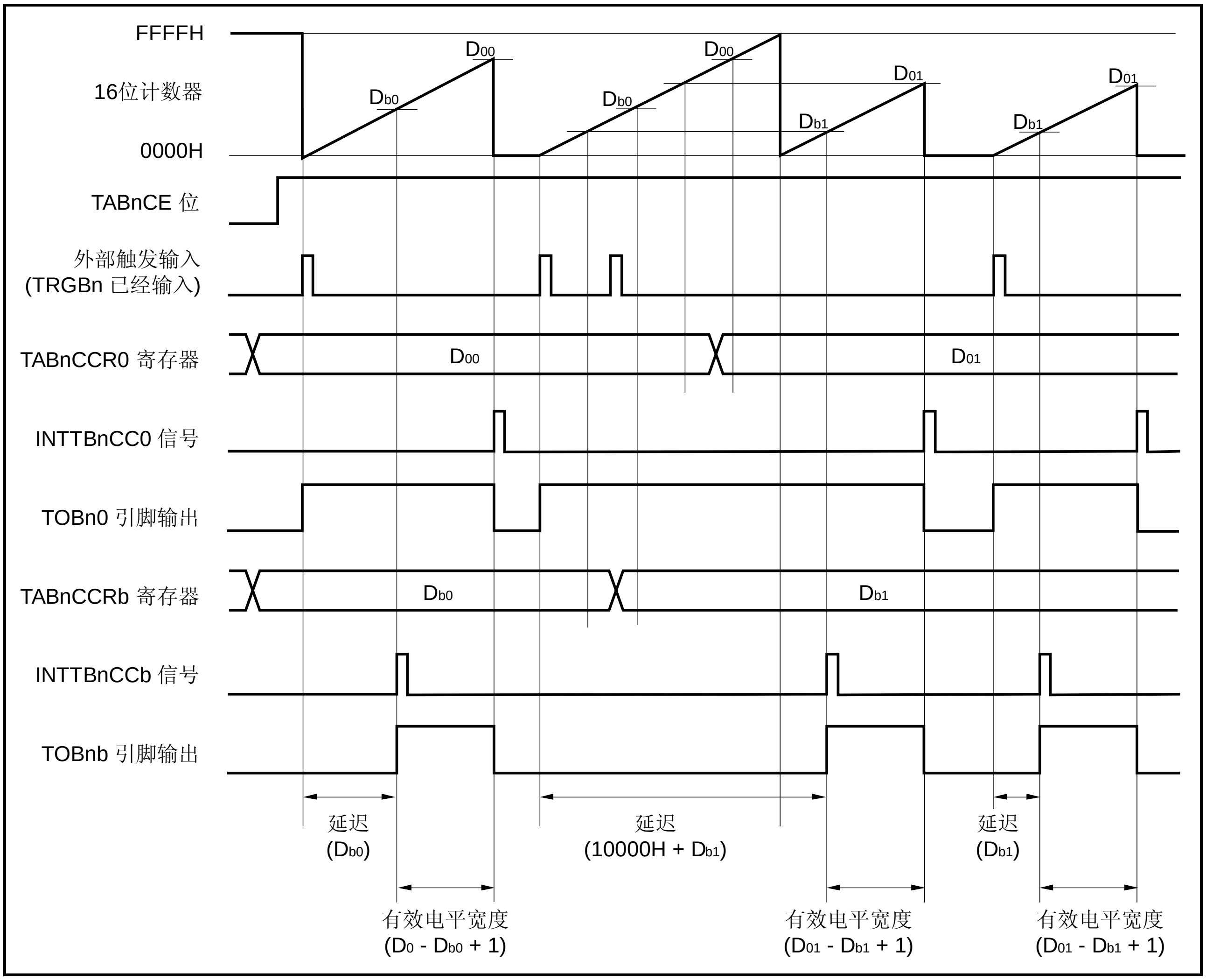
图 7-27. 单次触发脉冲输出模式的软件处理流程（2/2）



（2）单次触发脉冲输出模式时的操作时序

（a）重写 TABnCCRa 寄存器的注意事项

要将 TABnCCRa 寄存器的值改为比当前值较小的值，应停止计数，然后改变其设定值。当可能发生溢出时，停止计数，然后再改变设置值。



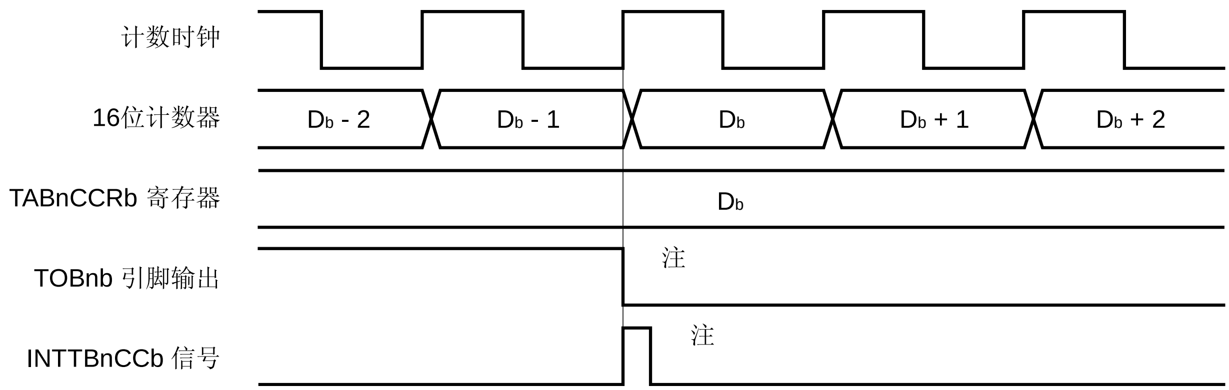
假设 D_{00} > D_{01} 而且 D_{b0} > D_{b1}，当 TABnCCR0 寄存器由 D_{00} 重写为 D_{01}，TABnCCRb 寄存器由 D_{b0} 重写为 D_{b1} 时，如果在 16 位计数器的计数值大于 D_{b1} 且小于 D_{b0} 时重写 TABnCCRb 寄存器，并且在计数值大于 D_{01} 且小于 D_{00} 时重写 TABnCCR0 寄存器，则一旦重写寄存器，就会在各设置值上反映出来，并立即与计数值进行比较。计数器向上计数到 FFFFH 后，从 0000H 开始重新向上计数。当计数值与 D_{b1} 一致时，计数器会产生 INTTBnCCb 信号并使 TOBnb 引脚有效。当计数值与 D_{01} 一致时，计数器产生 INTTBnCC0 信号，使 TOBnb 引脚失效，并停止计数。

因此，计数器输出脉冲的延迟时段或有效时段可能会与原先预计的单次触发脉冲有所不同。

备注 n = 0, 1, a = 0 至 3, b = 1 至 3

（**b**） 比较一致中断请求信号（**INTTBnCCb**）的发生时序

单次触发脉冲输出模式下，INTTBnCCb 的产生时序和其它情况下的 INTTBnCCb 信号发生时序不同，当 16 位计数器的计数值与 TABnCCRb 寄存器的值一致时，产生 INTTBnCCb 中断请求信号。



注　　时序实际被延迟一个操作时钟(fxx)

备注　　n = 0, 1
b = 1 至 3

通常在 16 位计数器的当前计数值与 TABnCCRb 寄存器的值一致之后，在下一次计数时同步发生 INTTBnCCb 信号。

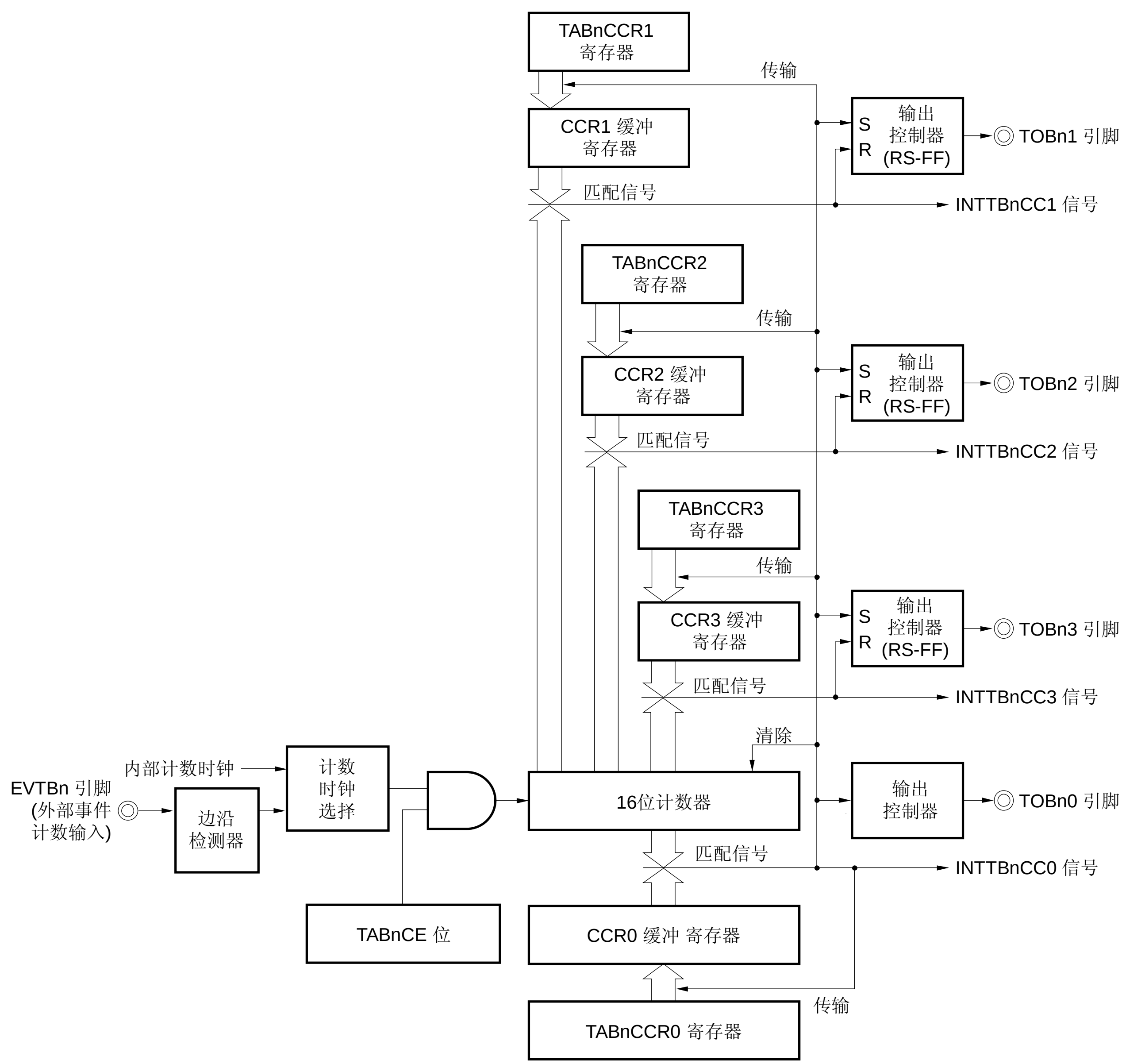
但是，在单次触发脉冲输出模式下，INTTP0CC1 信号会提早一个计数时钟周期产生。该时序的变化是为了一致 TOBnb 引脚输出信号变化的时序。

7.6.5　PWM 输出模式（TABnMD2 至 TABnMD0 位＝100）

在 PWM 输出模式下，当 TABnCTL0.TABnCE 位被置为 1 时，PWM 波形从 TOBn1 至 TOBn3 引脚输出。

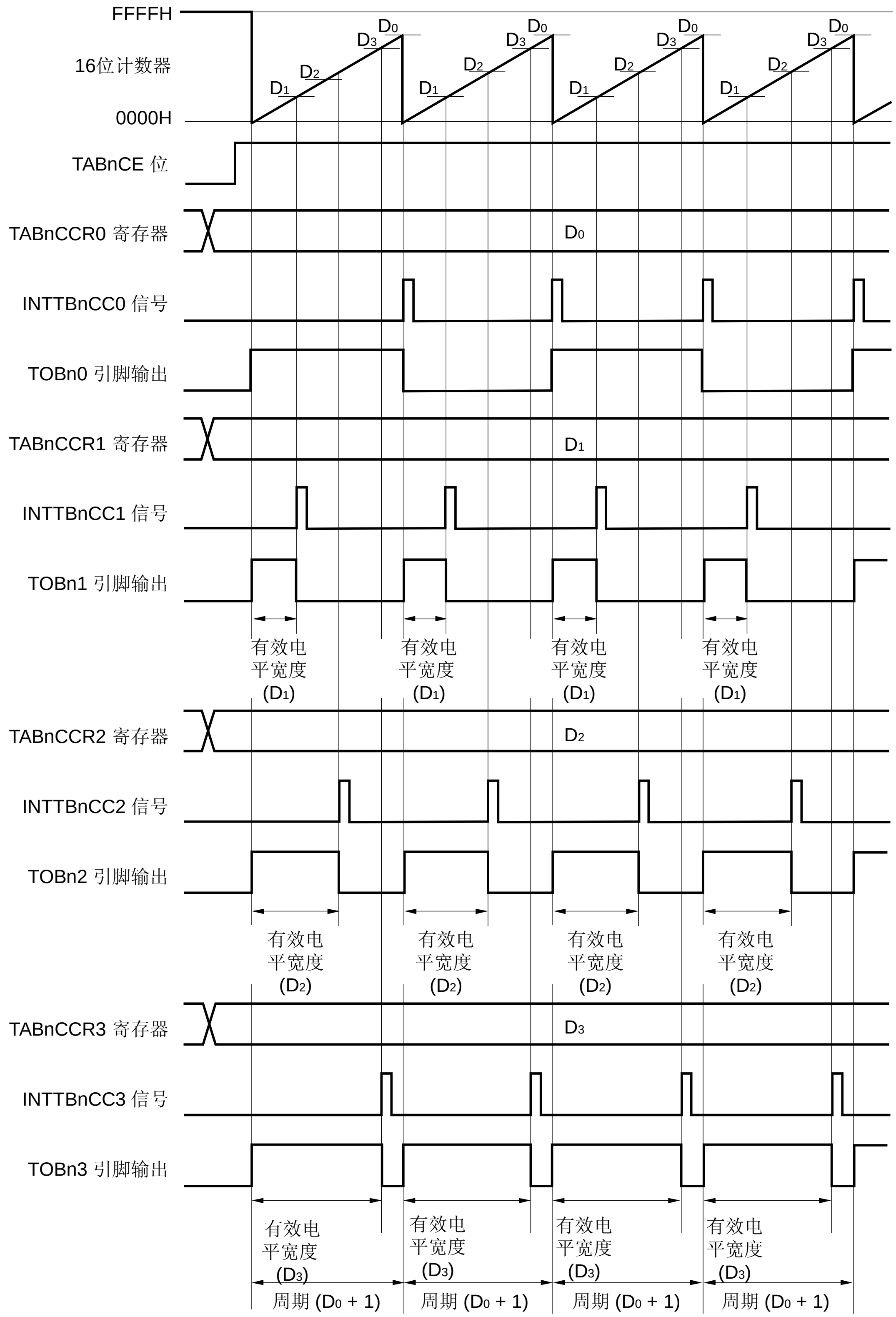
而且，从 TOBn0 引脚输出一个占空比 50%的 PWM 波形，其半周期等于 TABnCCR0 寄存器的设置值 + 1。

图 7-28. PWM 输出模式的配置



备注　　n = 0, 1

图 7-29. PWM 输出模式的基本时序



备注　　n = 0, 1

当 TABnCE 位置为 1 时，16 位计数器从 FFFFH 清零为 0000H，计数开始，同时从 TOBnb 引脚输出 PWM 波形。PWM 波形的有效电平宽度、周期、占空比系数的计算如下。

有效电平宽度＝（TABnCCRb 寄存器的设定值）× 计数时钟周期
周期＝（TABnCCR0 寄存器的设定值 ＋1）× 计数时钟周期
占空比系数＝（TABnCCRb 寄存器的设定值）/（TABnCCR0 寄存器的设定值 ＋1）

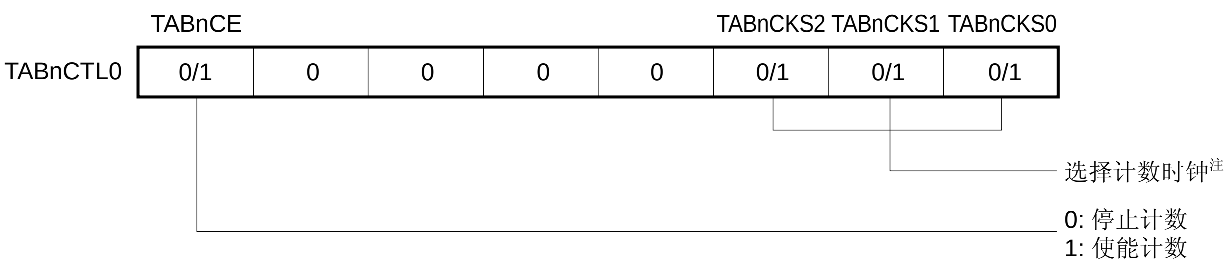
在计数操作时，可通过重写 TABnCCRa 寄存器来改变 PWM 波形。当 16 位计数器的计数值与 CCR0 缓冲寄存器的值一致时，16 位计数器清零为 0000H 时，新写入的值会发生作用。

当 16 位计数器的计数值与 CCR0 缓冲寄存器的值一致之后，进行下一次计数时，会同步产生比较一致中断请求信号 INTTBnCC0，且 16 位计数器清零为 0000H。当 16 位计数器的计数值与 CCRb 缓冲寄存器的值一致时，会生成比较一致中断请求信号 INTTBnCCb。

备注 n = 0, 1
a = 0 至 3
b = 1 至 3

图 7-30. PWM 输出模式时的寄存器设置（1/3）

（a） TABn 控制寄存器 0（TABnCTL0）



注 当 TABnCTL1.TABnEEE 位＝1 时，设置无效。

（b） TABn 控制寄存器 1（TABnCTL1）

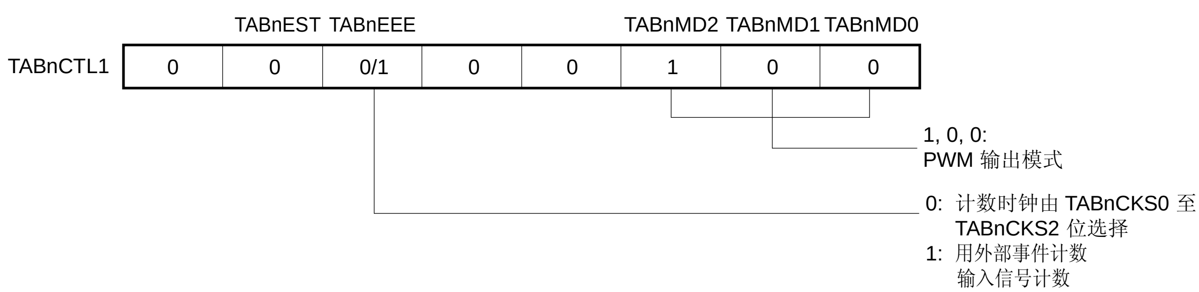


图 7-30. PWM 输出模式时的寄存器设置（2/3）

（c） TABn I/O 控制寄存器 0（TABnIOC0）

	TABnOL3	TABnOE3	TABnOL2	TABnOE2	TABnOL1	TABnOE1	TABnOL0	TABnOE0
TABnIOC0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TABnOE0:
0: 禁止 TOBn0 引脚 输出
1: 使能 TOBn0 引脚 输出

TABnOL0: 计数操作前，TOBn0 引脚输出电平的设置
0: 低电平
1: 高电平

TABnOE1:
0: 禁止 TOBn1 引脚 输出
1: 使能 TOBn1 引脚 输出

TABnOL1: 计数操作前，TOBn1 引脚输出电平的设置
0: 低电平
1: 高电平

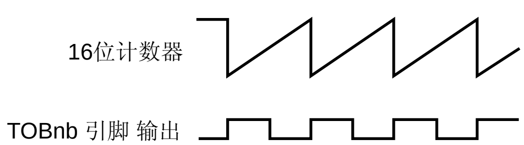
TABnOE2:
0: 禁止 TOBn2 引脚 输出
1: 使能 TOBn2 引脚 输出

TABnOL2: 计数操作前，TOBn2 引脚输出电平的设置
0: 低电平
1: 高电平

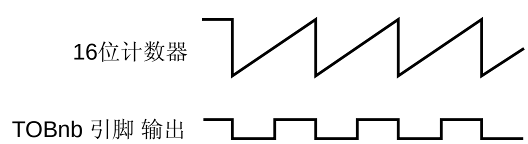
TABnOE3:
0: 禁止 TOBn3 引脚 输出
1: 使能 TOBn3 引脚 输出

TABnOL3: 计数操作前，TOBn3 引脚输出电平的设置
0: 低电平
1: 高电平

• 当 TABnOLb 位 = 0



• 当 TABnOLb 位 = 1



（d） TABn I/O 控制寄存器 2（TABnIOC2）

					TABnEES1	TABnEES0	TABnETS1	TABnETS0
TABnIOC2	0	0	0	0	0/1	0/1	0	0

TABnEES1, TABnEES0: 选择外部事件计数输入的有效边沿 (EVTBn 引脚).

（e） TABn 计数器读取缓冲寄存器（TABnCNT）

通过读取 TABnCNT 寄存器，可以读出 16 位计数器的值。

图 7-30. PWM 输出模式时的寄存器设置（3/3）

（f） TABn 捕获/比较寄存器 0 至 3（TABnCCR0 至 TABnCCR3）

如果将 TABnCCR0 寄存器的值设置为 D_0，将 TABnCCRb 寄存器的值设置为 D_b，则 PWM 波形的周期和有效电平的计算如下：

PWM 波形周期＝（D_0 ＋1）× 计数时钟周期
PWM 波形有效电平宽度＝D_b × 计数时钟周期

备注　**1.** 在 PWM 输出模式下，不使用 TABn I/O 控制寄存器 1（TABnIOC1）和 TABn 选项寄存器 0（TABnOPT0）。

2. n = 0, 1
b = 1 至 3

（1）**PWM** 输出模式下的操作流程

图 **7-31. PWM** 输出模式的软件处理流程（**1/2**）

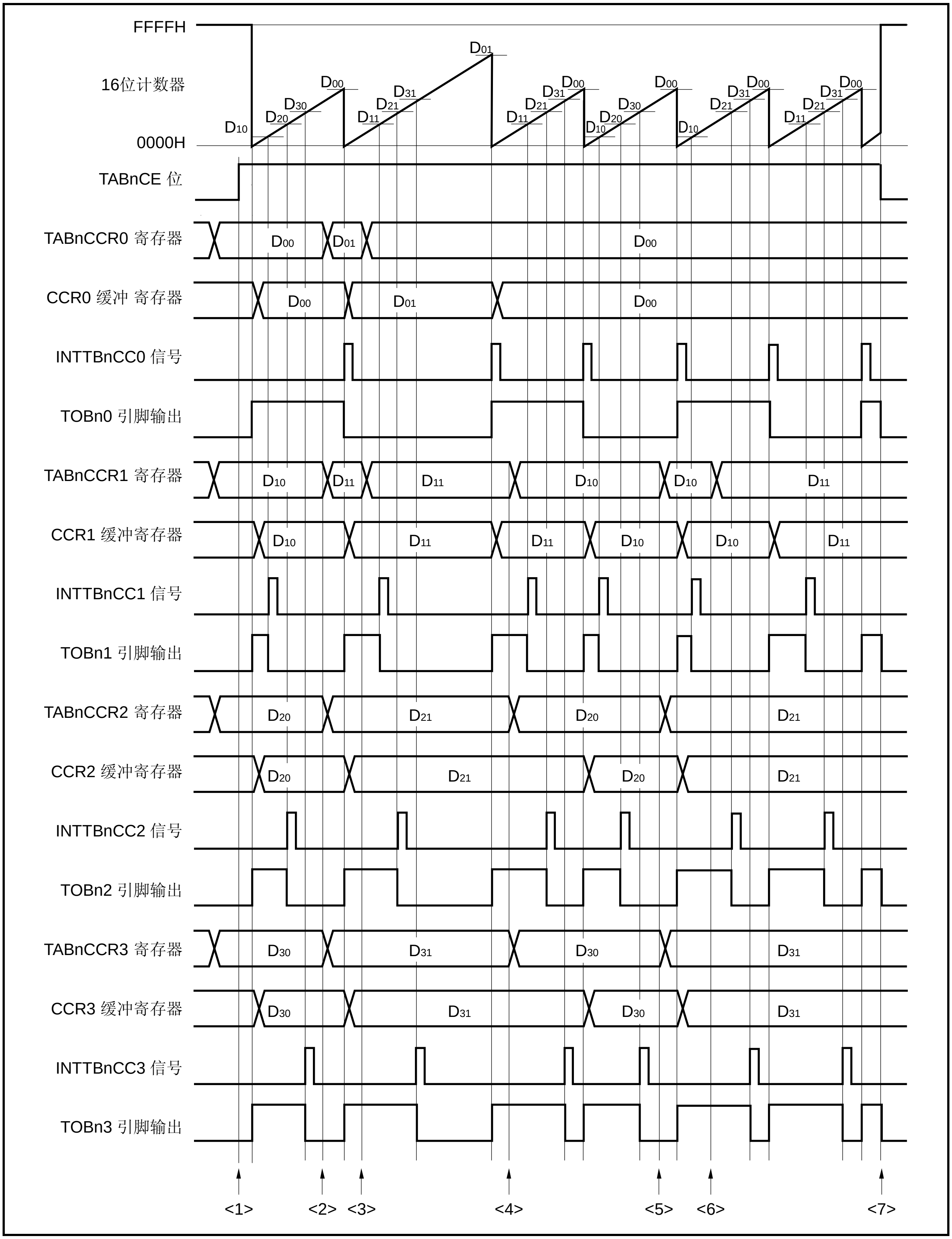
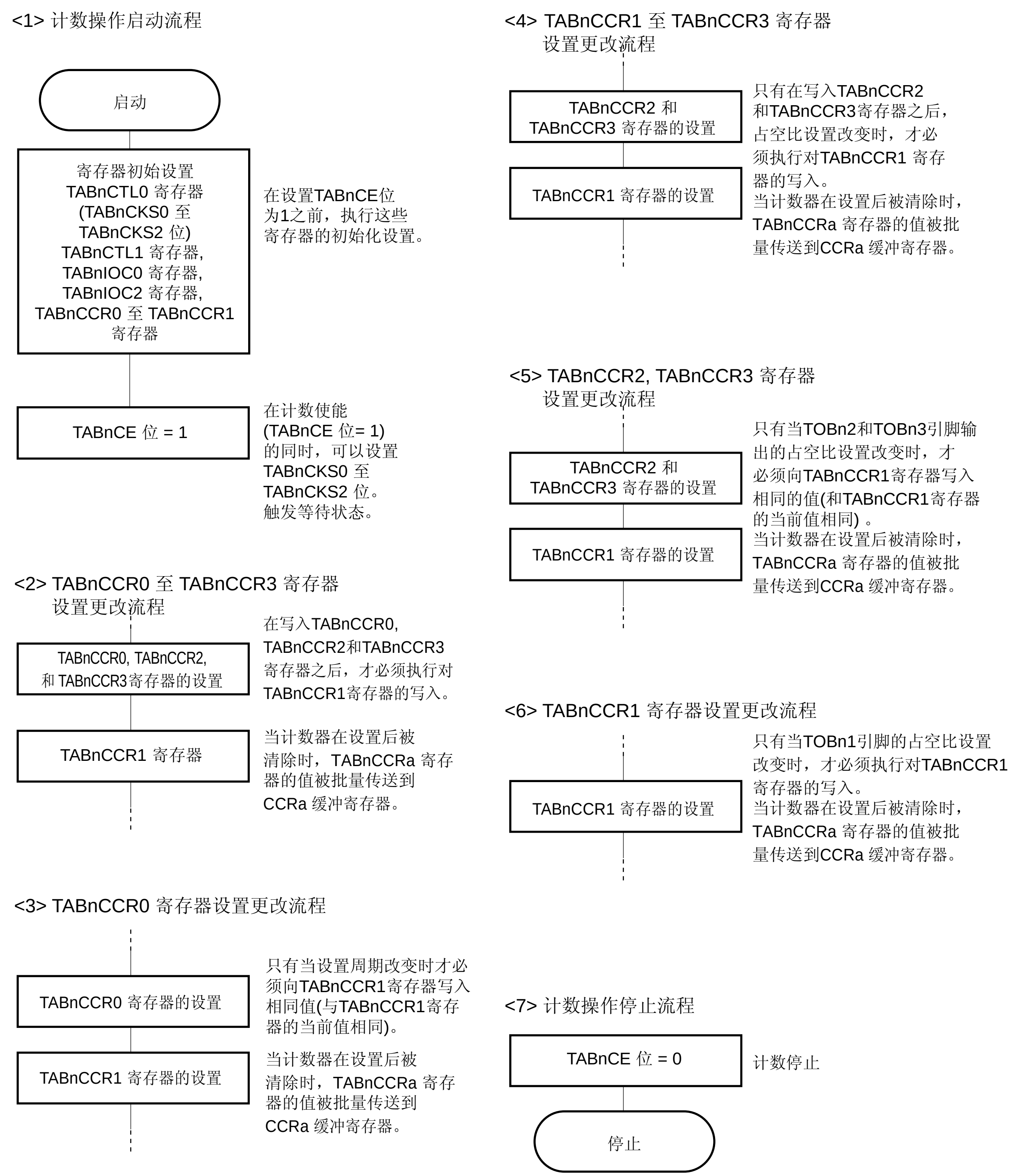


图 7-31. PWM 输出模式时的软件处理流程（2/2）



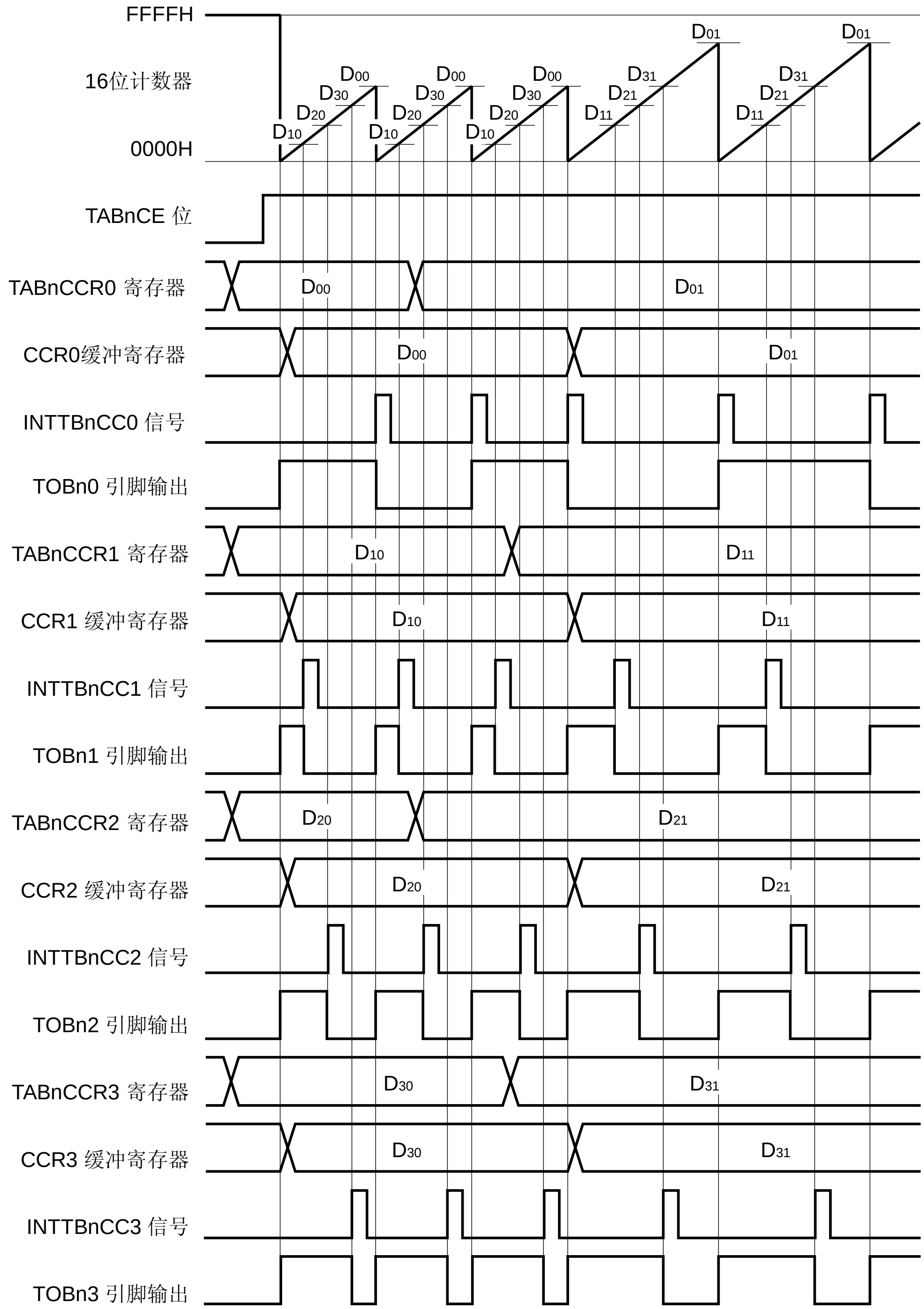
备注　n = 0, 1
　　　　a = 0 至 3

（2）**PWM 输出模式的操作时序**

（a） 操作过程中改变脉冲宽度

计数器操作过程中，若要改变 PWM 波形，应重新写入 TABnCCR1 寄存器。

当检测到 INTTBnCC0 信号以后，在写入 TABnCCR1 寄存器之后重写 TABnCCRa 寄存器。



备注　　n = 0, 1

为了将数据从 TABnCCRa 寄存器传输到 CCRa 缓冲寄存器，必须对 TABnCCR1 寄存器进行写入操作。

此时如果 PWM 波形的周期和有效电平都要改变，应先将周期设定到 TABnCCR0 寄存器，再将有效电平宽度设定到 TABnCCR2 和 TABnCCR3 寄存器，然后将有效电平设定到 TABnCCR1 寄存器。

如果只改变 PWM 波形的周期，应先把周期设置到 TABnCCR0 寄存器，然后将相同的值（与 TABnCCR1 寄存器的当前值相同）写入 TABnCCR1 寄存器。

如果只改变 PWM 波形的有效电平宽度（占空比），需要先向 TABnCCR2 和 TABnCCR3 寄存器设置有效电平，然后再向 TABnCCR1 寄存器设置有效电平。

如果只改变 TOBn1 引脚输出的 PWM 波形的有效电平宽度（占空比系数），仅需要设置 TABnCCR1 寄存器。

如果只改变 TOBn2 和 TOBn3 引脚输出的 PWM 波形的有效电平宽度（占空比），应先向 TABnCCR2 和 TABnCCR3 寄存器设置有效电平宽度，然后将相同的值（与 TABnCCR1 寄存器的当前值相同）写入 TABnCCR1 寄存器。

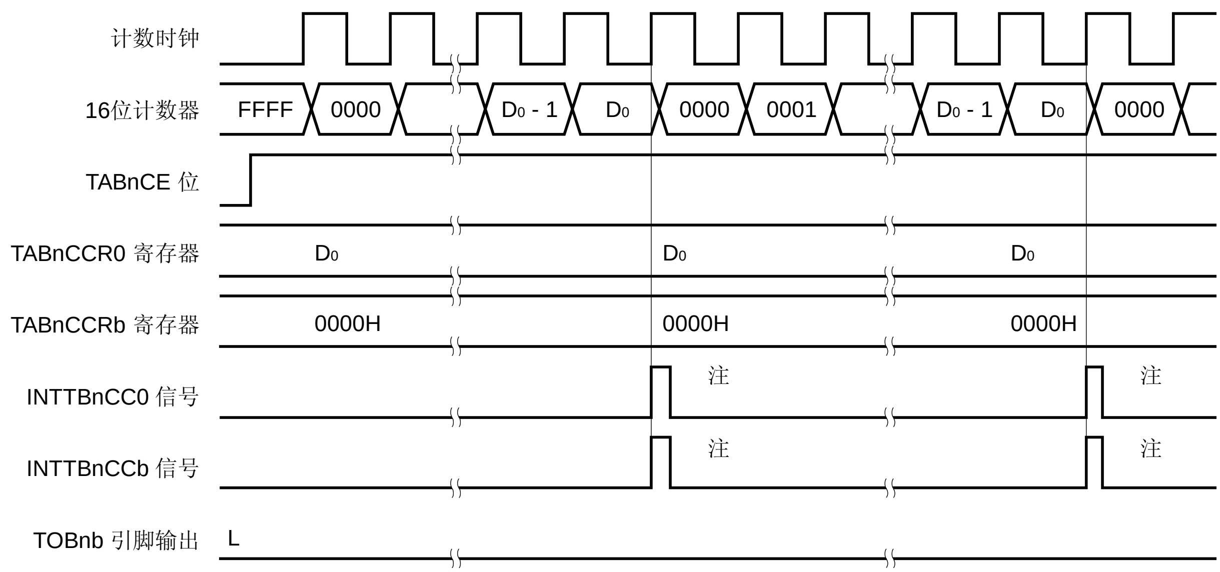
当数据写入 TABnCCR1 寄存器之后，在对 16 位计数器清零的同时，写入到 TABnCCRa 寄存器的值会被传输到 CCRa 缓冲寄存器，并作为与 16 位计数器进行比较的值。

在写入 TABnCCR1 寄存器一次之后，要再次写入 TABnCCR0 至 TABnCCR3 寄存器，应该在 INTTBnCC0 信号产生之后进行。否则，CCRa 缓冲寄存器的值可能不确定，因为把数据从 TABnCCRa 寄存器传输到 CCRa 缓冲寄存器的时序会和写入 TABnCCRa 寄存器的时序发生冲突。

备注　n = 0, 1
　　　a = 0 至 3,

（b） PWM 波形的 0%和 100% 输出

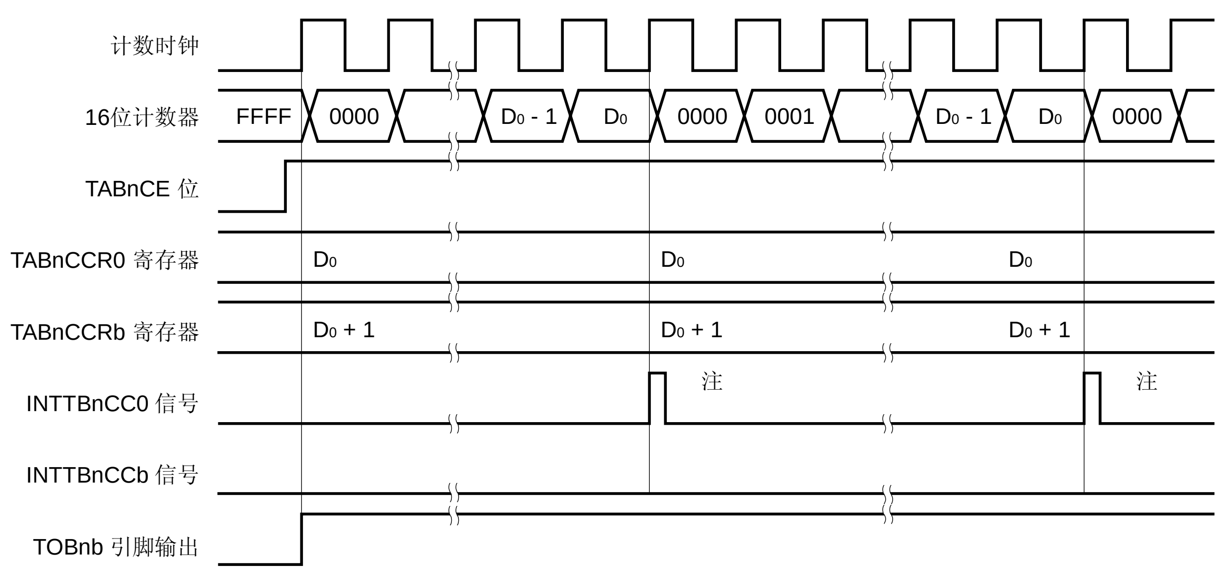
要输出一个 0% 波形，将 TABnCCRb 寄存器设置为 0000H。在 16 位计数器的计数值和 CCR0 缓冲寄存器的值发生一致之后，16 位计数器被清除为 0000H，且在下一个时序产生 INTTBnCC0 和 INTTBnCCb 信号。



注　　时序实际被延迟一个工作时钟（fxx）

备注　　n = 0, 1，b = 1 至 3

为了输出 100%的波形，需要将 TABnCCRb 寄存器的值设为（TABnCCR0 寄存器的设定值 ＋1）。如果 TABnCCR0 寄存器的设定值为 FFFFH，则不会产生 100%波形输出。

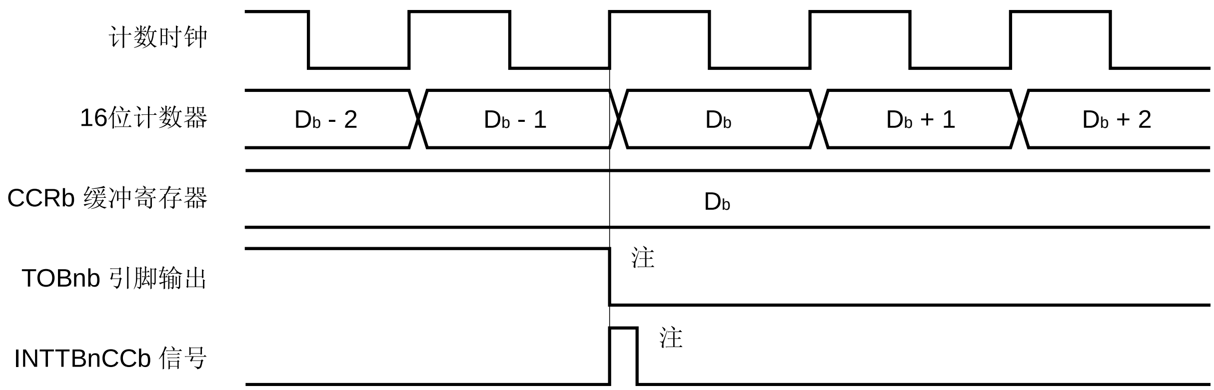


注　　时序实际被延迟一个工作时钟（fxx）

备注　　n = 0, 1，b = 1 至 3

（c） 比较一致中断请求信号（INTTBnCCb）的发生时序

在 PWM 输出模式下，INTTBnCCb 信号的产生时序与其它情况下 INTTBnCCb 信号的发生时序有所不同；当 16 位计数器的计数值与 TABnCCRb 寄存器的值一致时，产生 INTTBnCCb 信号。



注　　时序实际被延迟一个工作时钟（fxx）

备注　　n = 0, 1，b = 1 至 3

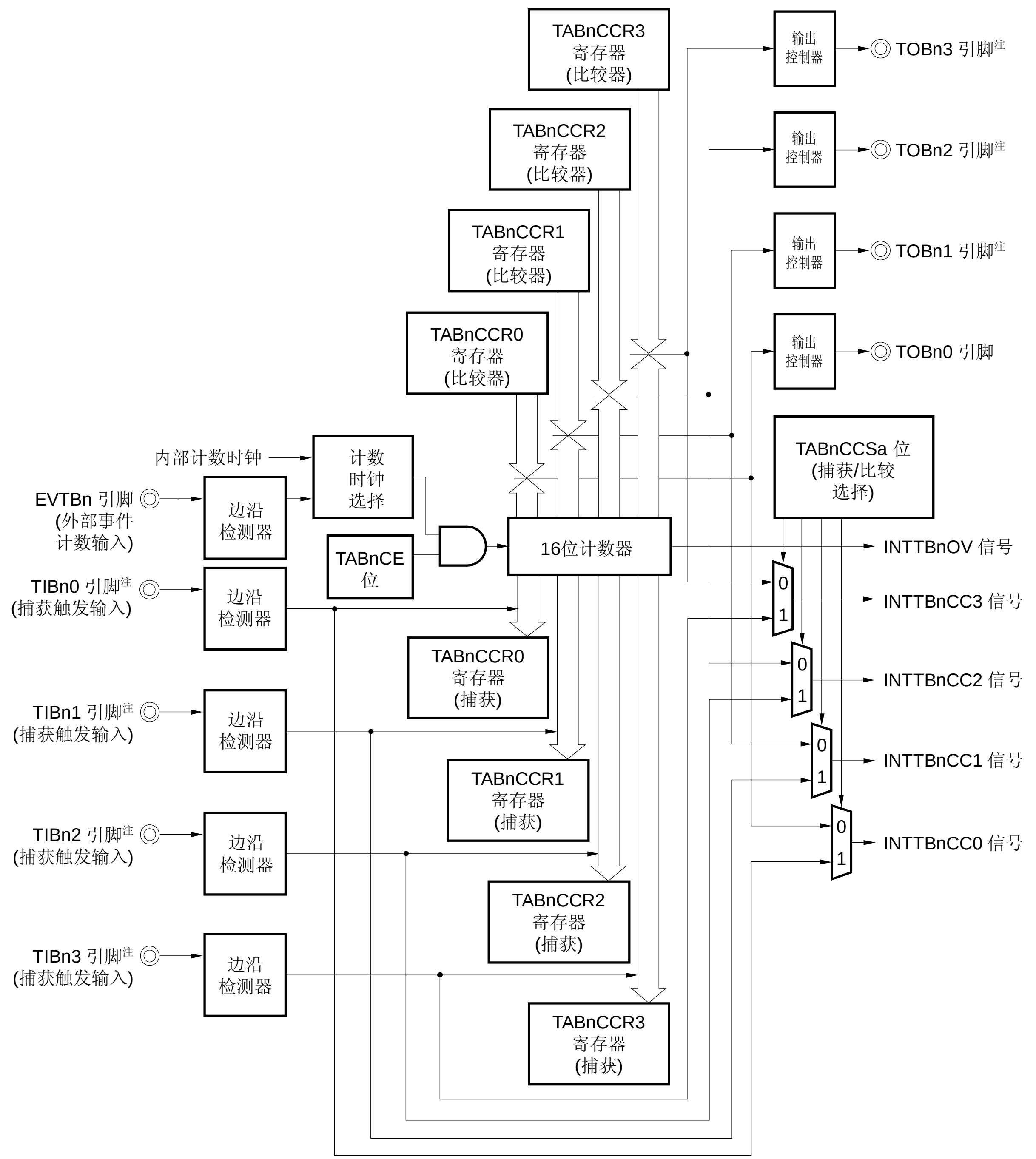
通常在 16 位计数器的计数值与 TABnCCRb 寄存器的值一致之后，在下一次计数时同步发生 INTTBnCCb 信号。

但是，在 PWM 输出模式下，会提早一个计数时钟周期产生。该时序的变化是为了一致 TOBnb 引脚输出信号变化的时序。

7.6.6　自由运行定时器模式（TABnMD2 至 TABnMD0 位＝101）

在自由运行定时器模式下，当 TABnCTL0.TABnCE 位置为 1 时，16 位定时器/事件计数器 AB 开始计数。此时，根据 TABnOPT0.TABnCCSa 位的设置，TABnCCRa 寄存器可以用作比较寄存器或捕获寄存器。

图 7-32. 自由运行定时器模式的配置



注　因为捕获触发输入引脚（TIBnb）和定时器输出引脚（TOBnb）共用相同的复用功能引脚，这两个功能不能同时使用。

备注　n = 0, 1，a = 0 至 3，b = 1 至 3

- 比较操作

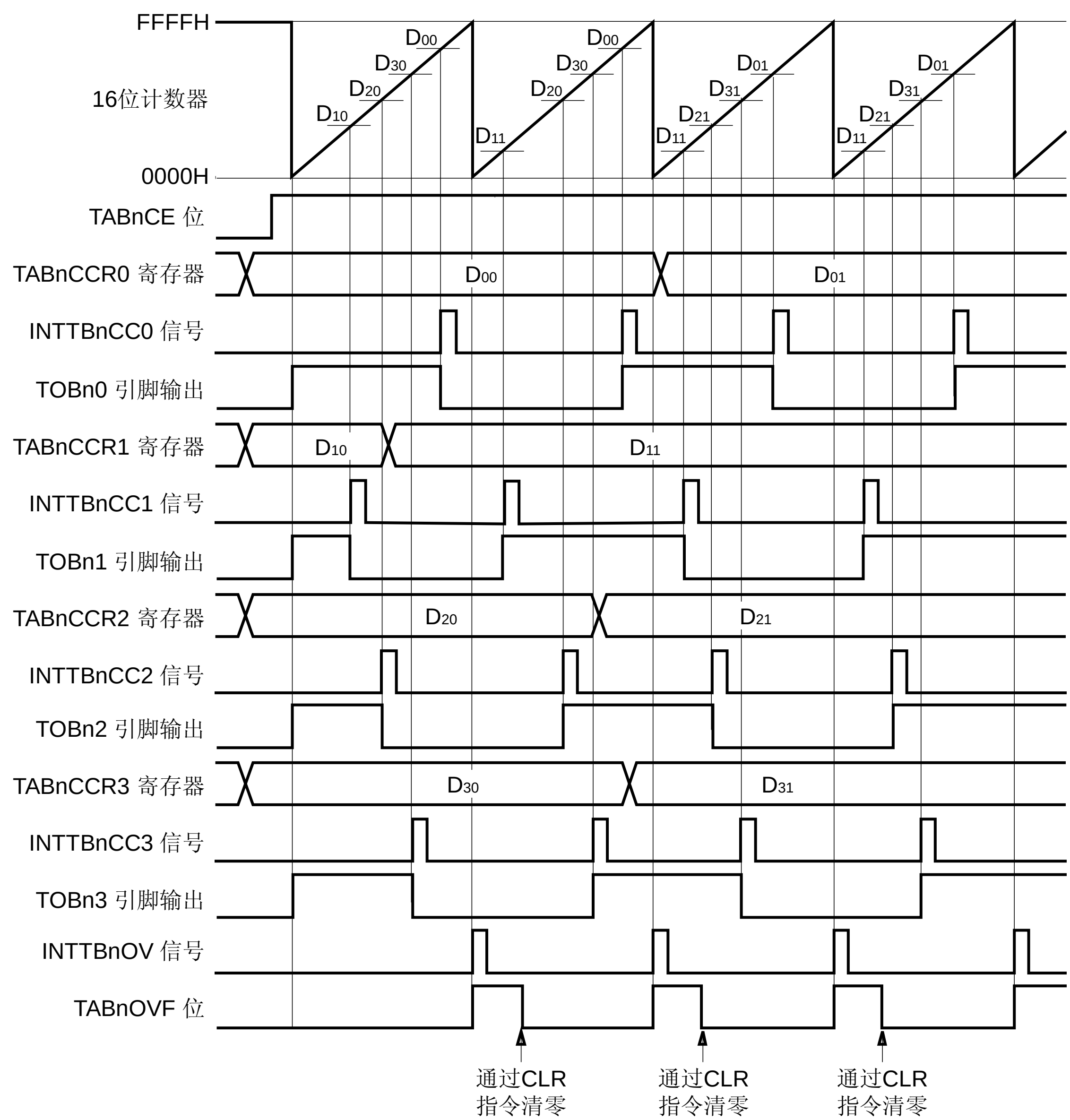
当 TABnCE 位被置为 1 时，16 位定时器/事件计数器 AB 开始计数，TOBn0 至 TOBn3 引脚的输出信号反相。此后当 16 位计数器的计数值与 TABnCCRa 寄存器的设定值一致时，产生一个比较一致中断请求信号（INTTBnCCa），而且 TOBn0 至 TOBn3 引脚的输出信号被反转。

16 位计数器继续对计数时钟同步计数。当它计数到 FFFFH 时，会在下一个计数时钟时产生一个溢出中断请求信号（INTTBnOV），同时计数器清零为 0000H，然后继续进行计数。此时，溢出标志（TABnOPT0.TABnOVF 位）被置为 1，确认溢出标志位被置为 1 然后通过软件运行 CLR 指令将其清除为 0。

当计数器工作时，可以重写 TABnCCRa 寄存器。如果进行重写，则新写入的值会立刻反映出来，并与计数值进行比较。

备注　n = 0, 1
　　　a = 0 至 3,

图 7-33. 自由运行定时器模式下的基本时序（比较功能）



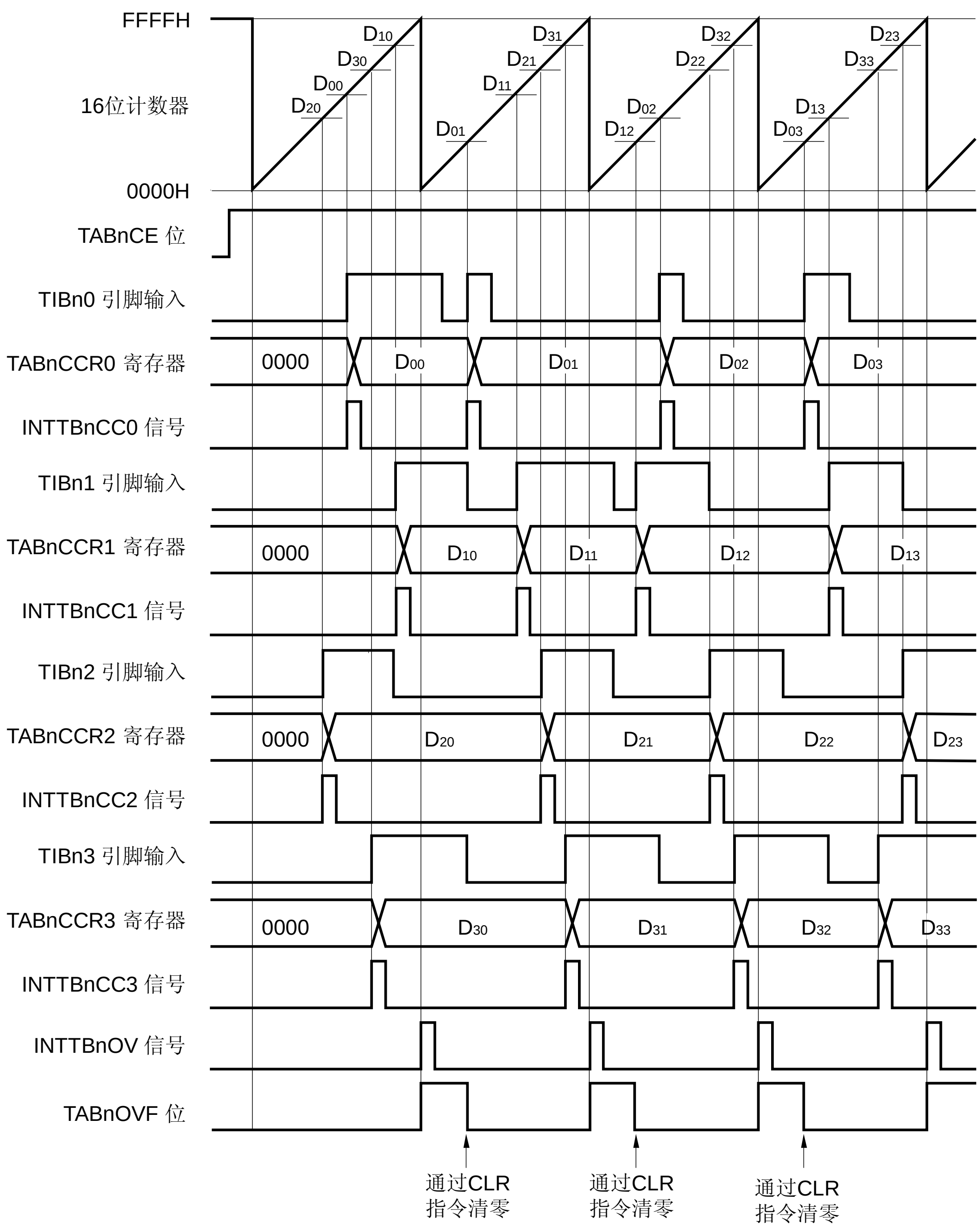
备注　n = 0, 1

- 捕获操作

当 TABnCE 位被置为 1 时，16 位计数器开始计数。当检测到 TIBna 引脚输入信号的有效边沿时，16 位计数器的计数值存储在 TABnCCRa 寄存器中，并产生一个捕获中断请求信号（INTTBnCCa）。

16 位计数器继续对计数时钟同步计数。当它计数到 FFFFH 时，会在下一个计数时钟时产生一个溢出中断请求信号（INTTBnOV），同时计数器清零为 0000H，然后继续进行计数。此时，溢出标志（TABnOPT0.TABnOVF 位）被置为 1，确认溢出标志位被置为 1 然后通过软件运行 CLR 指令将其清除为 0。

图 7-34. 自由运行定时器模式下的基本时序（捕获功能）



备注　n = 0，1

　　　a = 0 至 3

图 **7-35.** 自由运行定时器模式下的寄存器设置（**1/3**）

（**a**） **TABn** 控制寄存器 **0**（**TABnCTL0**）

	TABnCE					TABnCKS2	TABnCKS1	TABnCKS0
TABnCTL0	0/1	0	0	0	0	0/1	0/1	0/1

TABnCKS2～TABnCKS0：选择计数时钟注

TABnCE：
0: 停止计数
1: 使能计数

注　当 TABnCTL1.TABnEEE 位＝1 时，设置无效。

（**b**） **TABn** 控制寄存器 **1**（**TABnCTL1**）

		TABnEST	TABnEEE			TABnMD2	TABnMD1	TABnMD0
TABnCTL1	0	0	0/1	0	0	1	0	1

TABnMD2～TABnMD0：
1, 0, 1:
自由运行模式

TABnEEE：
0: 计数时钟由TABnCKS0 至 TABnCKS2 位选择
1: 用外部事件计数输入信号计数

图 7-35. 自由运行定时器模式下的寄存器设置（2/3）

（c） TABn I/O 控制寄存器 0（TABnIOC0）

	TABnOL3	TABnOE3	TABnOL2	TABnOE2	TABnOL1	TABnOE1	TABnOL0	TABnOE0
TABnIOC0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

0: 禁止 TOBn0 引脚输出
1: 使能 TOBn0 引脚输出

计数操作之前，TOBn0 引脚输出电平的设置
0: 低电平
1: 高电平

0: 禁止 TOBn1 引脚输出
1: 使能 TOBn1 引脚输出

计数操作之前，TOBn1 引脚输出电平的设置
0: 低电平
1: 高电平

0: 禁止 TOBn2 引脚输出
1: 使能 TOBn2 引脚输出

计数操作之前，TOBn2 引脚输出电平的设置
0: 低电平
1: 高电平

0: 禁止 TOBn3 引脚输出
1: 使能 TOBn3 引脚输出

计数操作之前，TOBn3 引脚输出电平的设置
0: 低电平
1: 高电平

（d） TABn I/O 控制寄存器 1（TABnIOC1）

	TABnIS7	TABnIS6	TABnIS5	TABnIS4	TABnIS3	TABnIS2	TABnIS1	TABnIS0
TABnIOC1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

选择TIBn0 引脚输入的有效边沿

选择TIBn1 引脚输入的有效边沿

选择TIBn2 引脚输入的有效边沿

选择TIBn3 引脚输入的有效边沿

图 **7-35.** 自由运行定时器模式下的寄存器设置（**3/3**）

（**e**） **TABn I/O** 控制寄存器 **2**（**TABnIOC2**）

					TABnEES1	TABnEES0	TABnETS1	TABnETS0
TABnIOC2	0	0	0	0	0/1	0/1	0	0

TABnEES1, TABnEES0：选择外部事件计数输入的有效边沿 (EVTBn 引脚)

（**f**） **TABn** 选项寄存器 **0**（**TABnOPT0**）

	TABnCCS3	TABnCCS2	TABnCCS1	TABnCCS0		TABnCMS	TABnCUF	TABnOVF
TABnOPT0	0/1	0/1	0/1	0/1	0	0	0	0/1

TABnOVF：溢出标志

TABnCCS0：指定 TABnCCR0 寄存器功能作为捕获寄存器或比较寄存器
0: 比较寄存器
1: 捕获寄存器

TABnCCS1：指定TABnCCR1 寄存器功能作为捕获寄存器或比较寄存器
0: 比较寄存器
1: 捕获寄存器

TABnCCS2：指定 TABnCCR2 寄存器功能作为捕获寄存器或比较寄存器
0: 比较寄存器
1: 捕获寄存器

TABnCCS3：指定 TABnCCR3 寄存器功能作为捕获寄存器或比较寄存器
0: 比较寄存器
1: 捕获寄存器

（**g**） **TABn** 计数器读取缓冲寄存器（**TABnCNT**）

通过读取 TABnCNT 寄存器，可以读出 16 位计数器的值。

（**h**） **TABn** 捕获**/**比较寄存器 **0** 至 **3**（**TABnCCR0** 至 **TABnCCR3**）

根据 TABnOPT0.TABnCCSa 位的设置，这些寄存器可以用作捕获寄存器或比较寄存器。

当用作捕获寄存器时，在检测到 TIBna 引脚输入信号的有效边沿时，存储 16 位计数器的当前计数值。

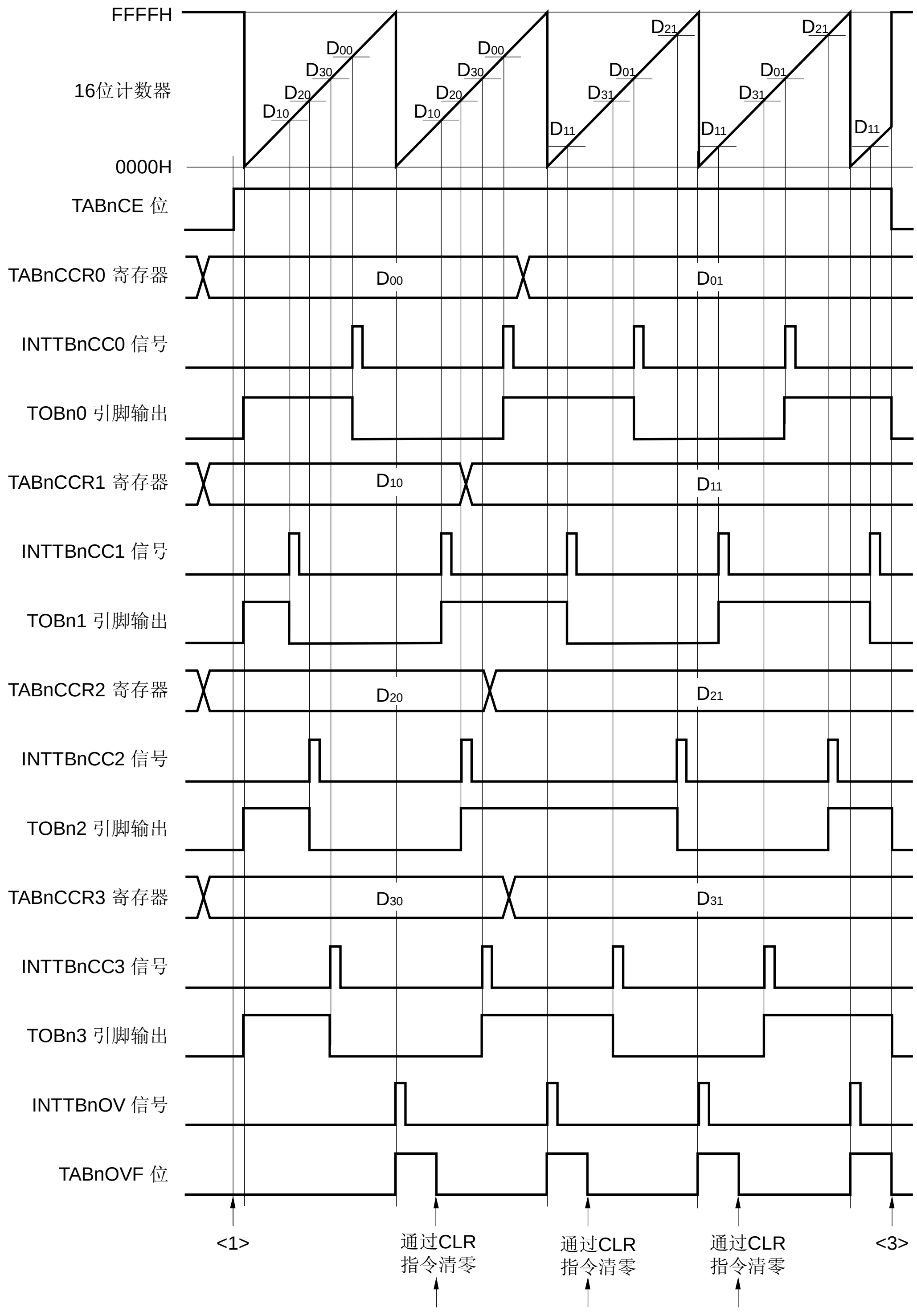
当用作比较寄存器时，而且 TABnCCRa 寄存器的值被设置为 D_a 时，则当计数器达到（D_a ＋1）时会产生 INTTBnCCa 中断信号，同时，TOBn0 至 TOBn3 引脚的输出信号被反转。

备注　n = 0, 1
a = 0 至 3,

（1）自由运行定时器模式时的操作流程

（a）当捕获/比较寄存器用作比较寄存器时

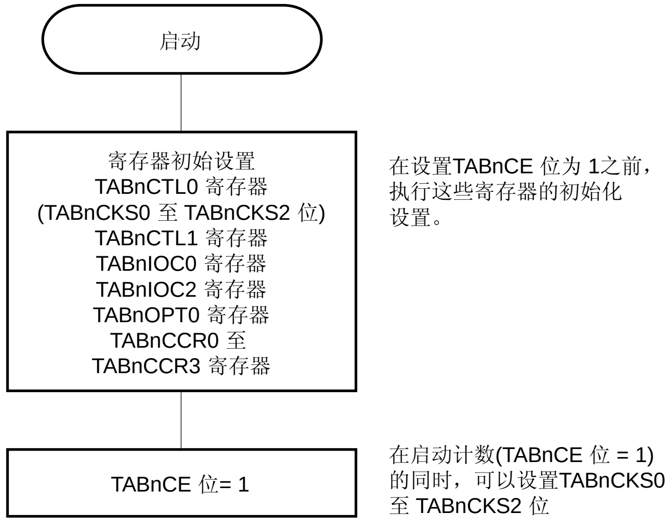
图 7-36. 自由运行定时器模式的软件处理流程（比较功能）（1/2）



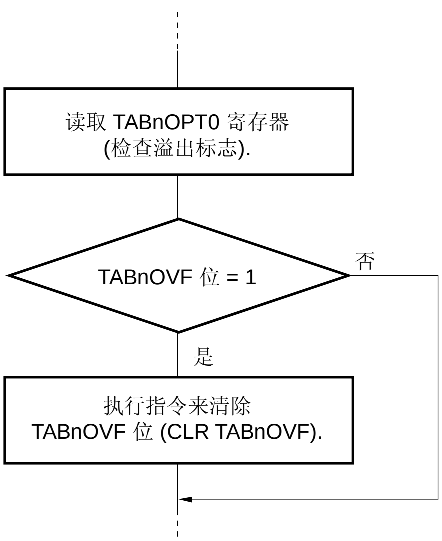
备注　　n = 0, 1

图 **7-36.** 自由运行定时器模式的软件处理流程（比较功能）（**2/2**）

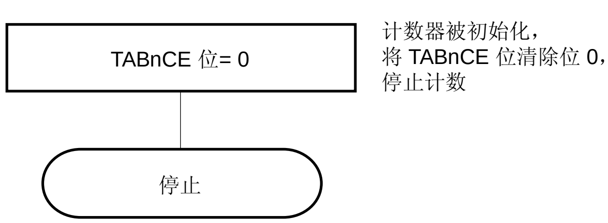
<1> 计数操作启动流程



<2> 溢出标志清除流程



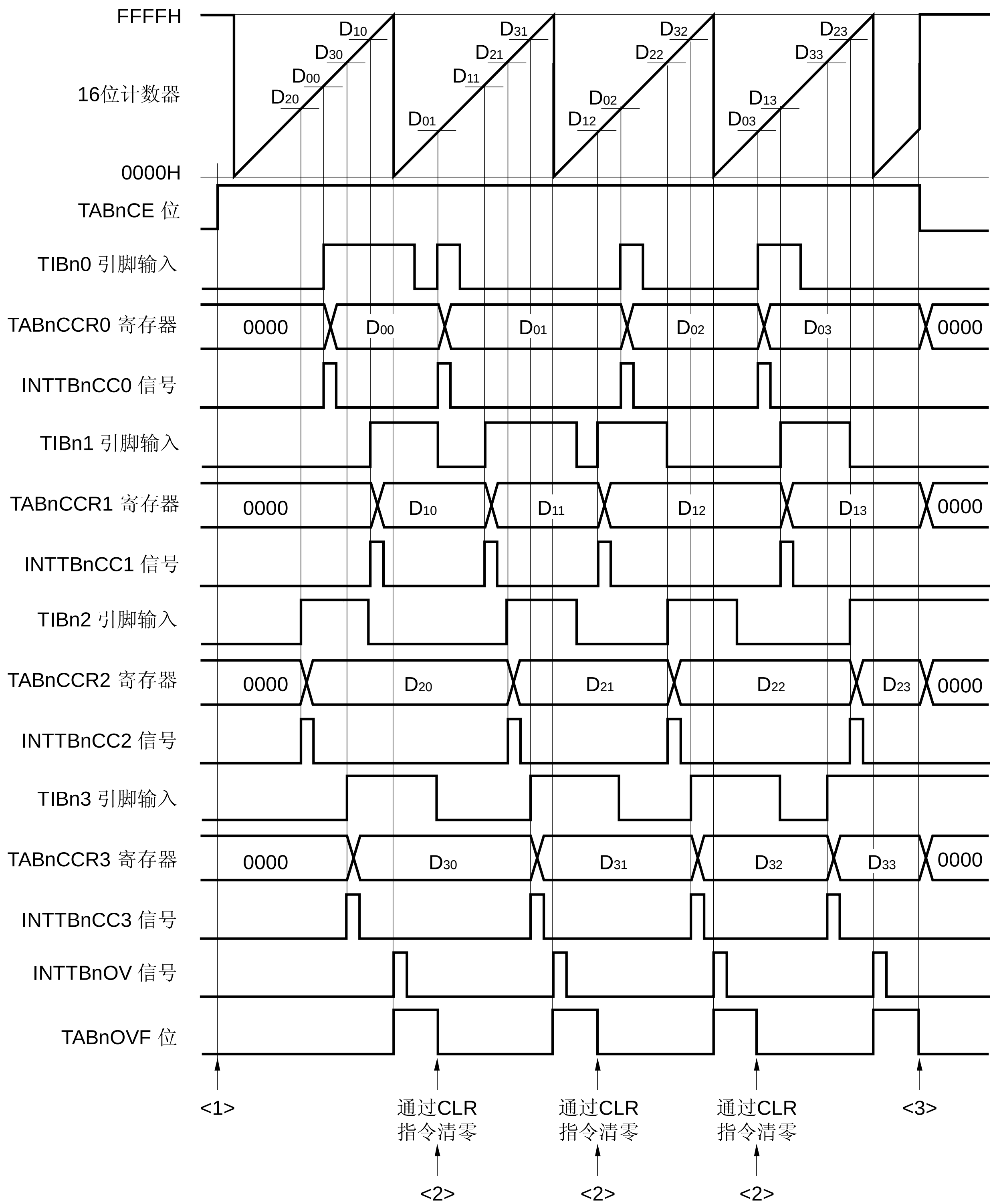
<3> 计数操作停止流程



备注　　n = 0, 1

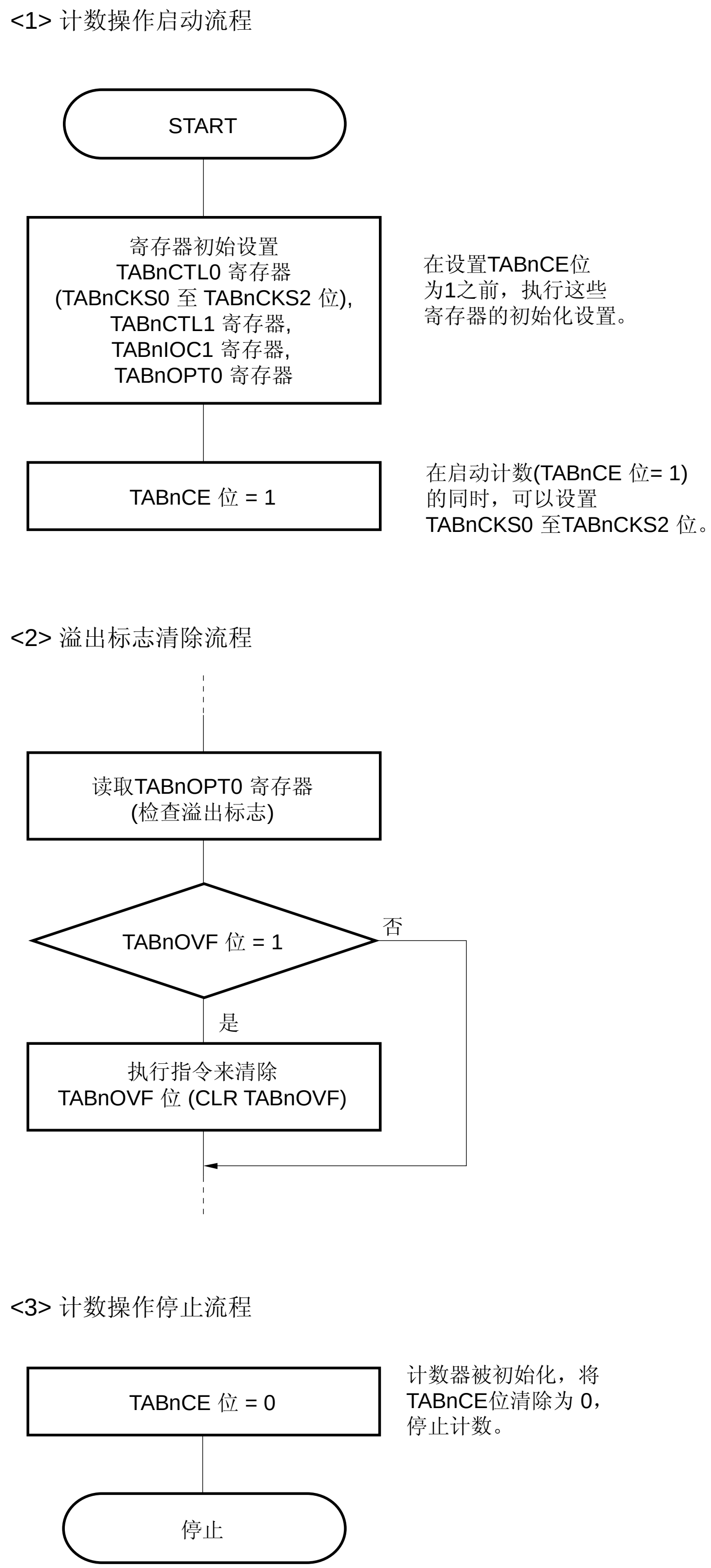
（b） 当捕获/比较寄存器用作捕获寄存器时

图 7-37. 自由运行定时器模式的软件处理流程（捕获功能）（1/2）



备注　n = 0, 1

图 **7-37.** 自由运行定时器模式时的软件处理流程（捕获功能）（**2/2**）

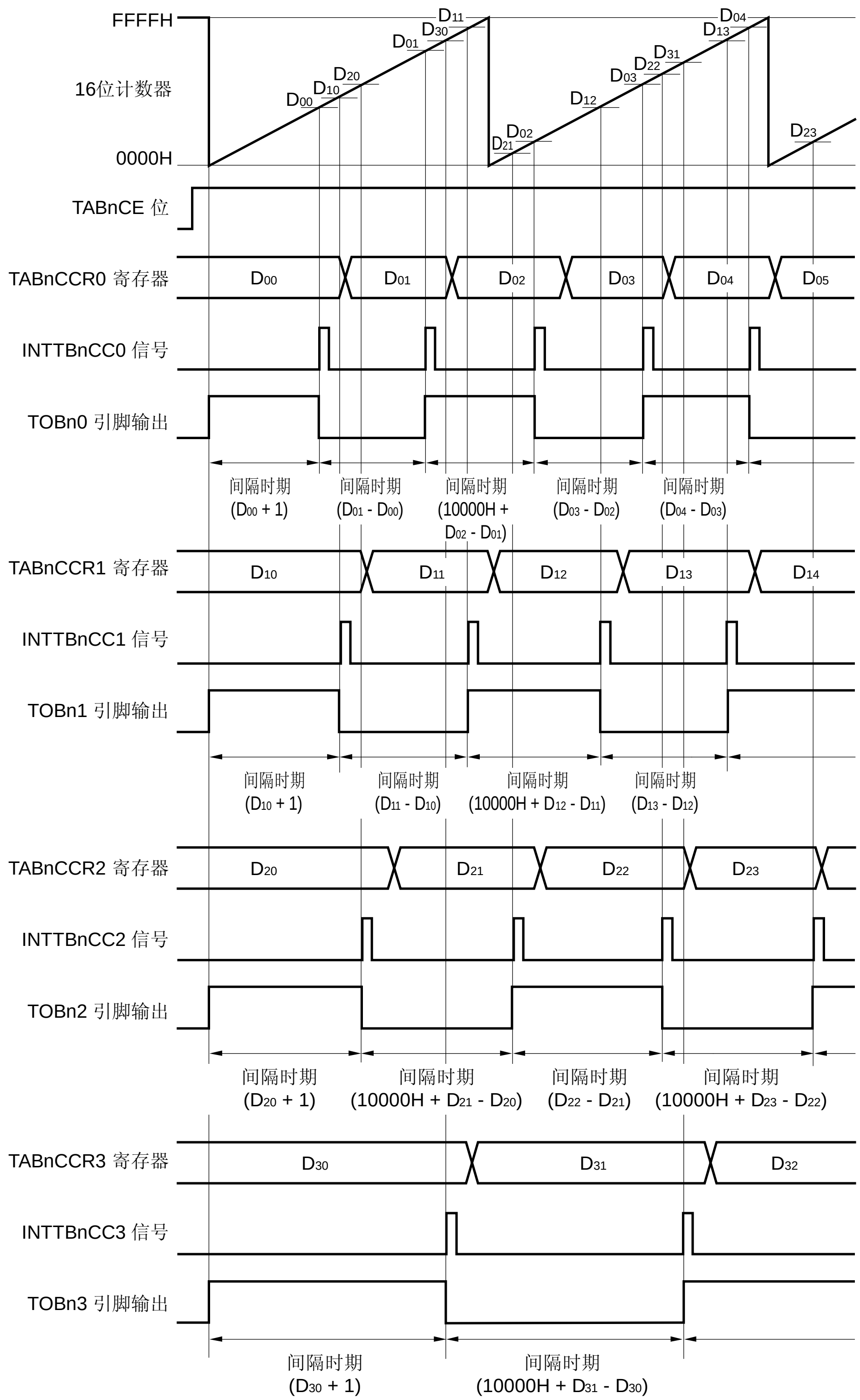


备注　　n = 0, 1

（2）自由运行定时器模式时的操作时序

（a）比较寄存器的间隔定时操作

当 16 位定时器/事件计数器 AB 用作间隔定时器，TABnCCRa 寄存器用作比较寄存器时，在每次检测到 INTTBnCCa 信号后，都需要通过软件设置一个比较值以产生下一个中断请求信号。



备注 n = 0, 1，a = 0 至 3

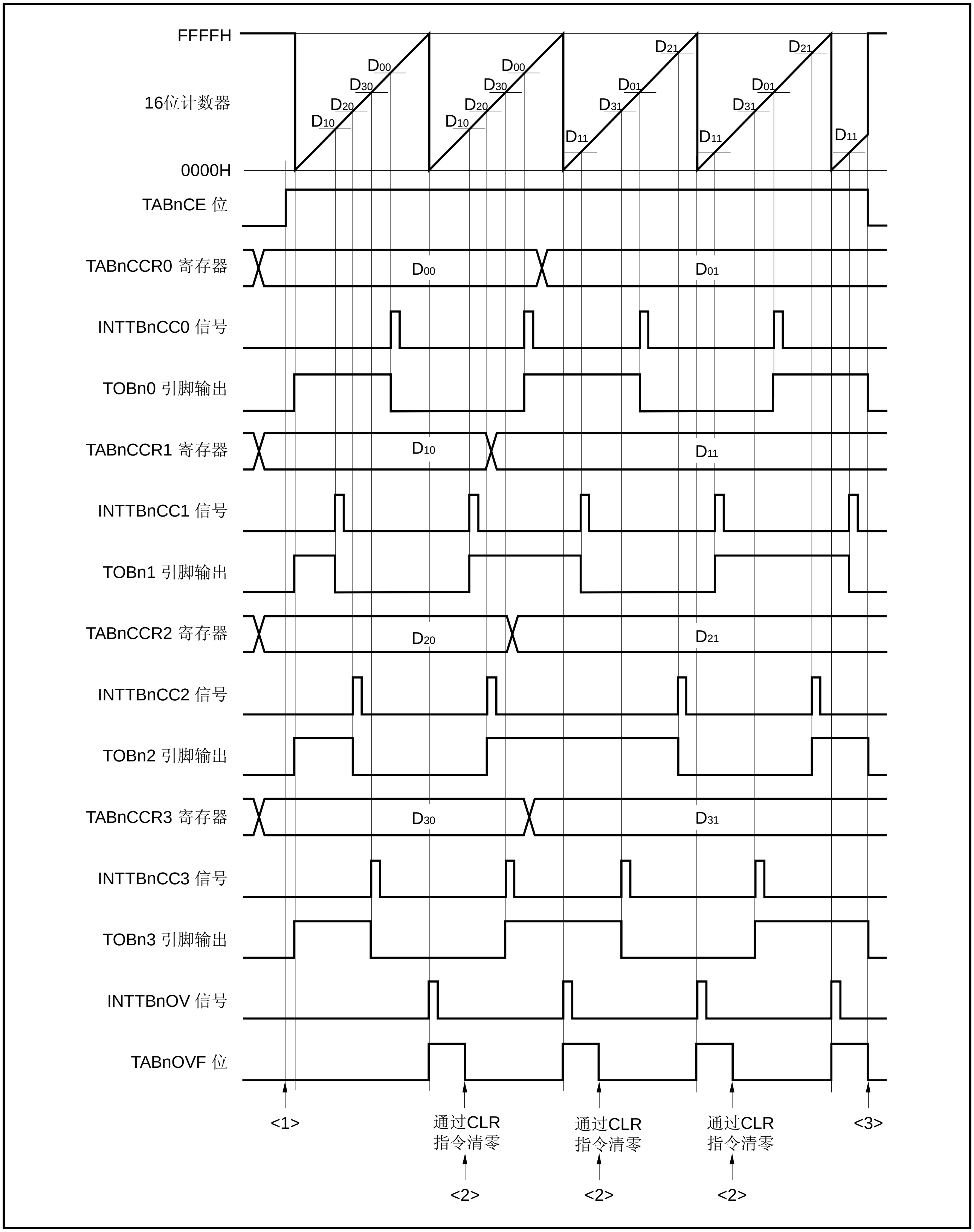
在自由运行定时器模式下进行时间间隔操作时，一个通道可以设置四个时间间隔。
要进行时间间隔操作，当检测到 INTTBnCCa 信号后，必须在中断服务程序中重置相应的 TABnCCRa 寄存器的值。
重置 TABnCCRa 寄存器的设置值可由下式计算，其中"D_a"是间隔时间：

比较寄存器默认值： $D_a - 1$
比较寄存器的第二次及其以后的设置值：上次设置值 ＋ D_a
（若计算结果大于 FFFFH，则从结果中减去 10000H，并将此值设置给寄存器。）

备注 n = 0, 1
a = 0 至 3,

（b） 使用捕获寄存器进行脉冲宽度测量

在用 TABnCCRa 寄存器作为捕获寄存器进行脉宽测量时，每次检测到 INTTBnCCa 信号时，都需要通过软件读取捕获寄存器并计算间隔时间。



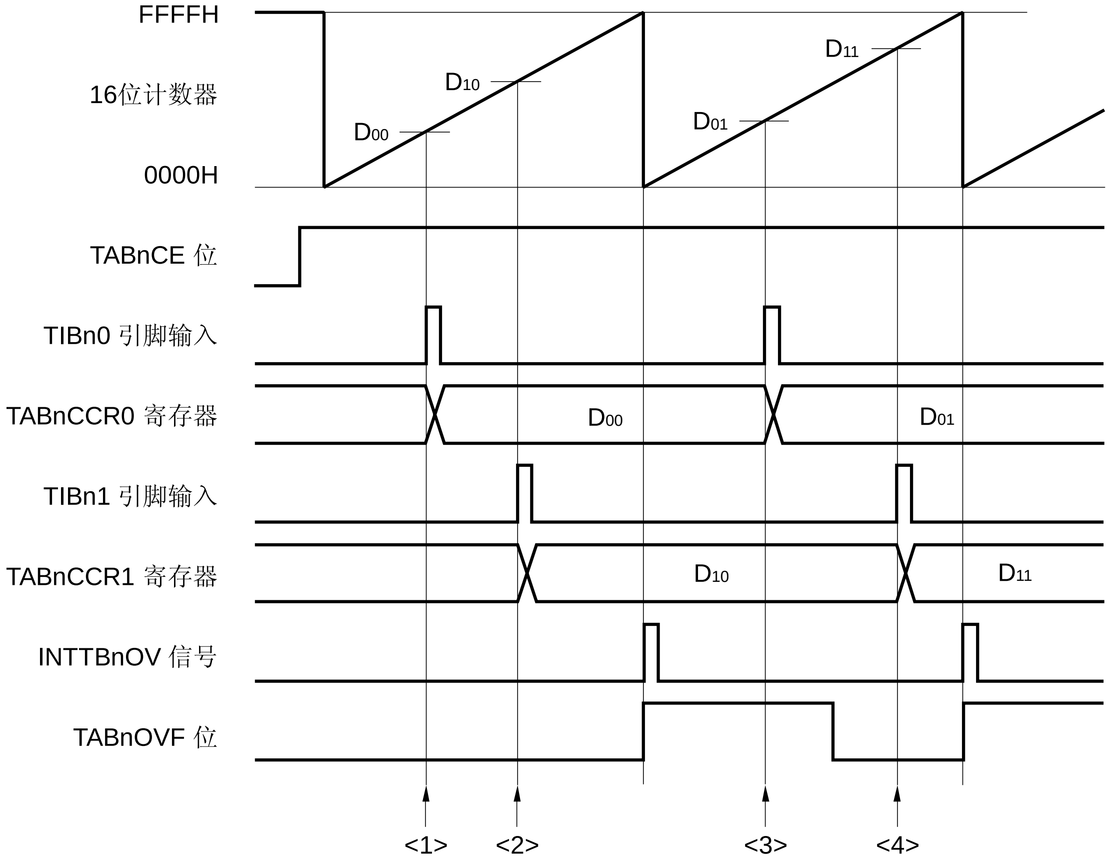
在自由运行定时器模式下执行脉宽测量时，可以用一个通道测量四个脉冲宽度。
要测量脉冲宽度，可以与 INTTBnCCa 信号同步读取 TABnCCRa 寄存器的值，并计算读取值与先前读取值之间的差，得到脉冲宽度。

备注　n = 0, 1
　　　a = 0 至 3,

（c） 使用两个捕获寄存器时对溢出的处理

当使用两个捕获寄存器时必须要格外小心。首先是一个不正确处理的示例，如下所示。

使用两个捕获寄存器时的不正确处理示例



在自由运行定时器模式下测量两个脉冲宽度时，可能出现下列问题。

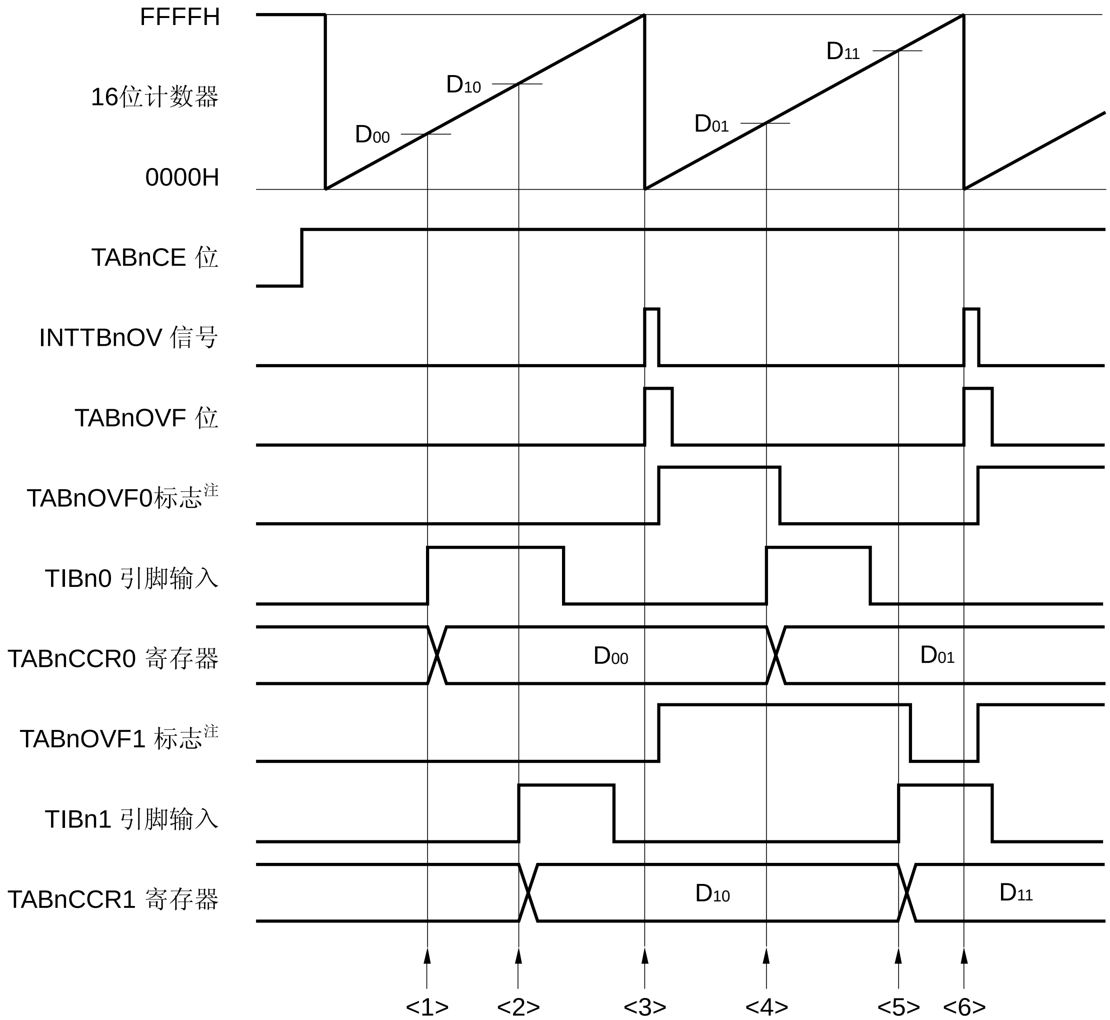
<1>　读取 TABnCCR0 寄存器（TIBn0 引脚输入的默认值设置）。
<2>　读取 TABnCCR1 寄存器（TIBn1 引脚输入的默认值设置）。
<3>　读取 TABnCCR0 寄存器。
　　读溢出标志。若溢出标志为 1，将其清为 0。
　　因为溢出标志为 1，所以脉冲宽度可以用式（10000H ＋$D_{01} - D_{00}$）计算。
<4>　读取 TABnCCR1 寄存器。
　　读溢出标志。因为此标志在<3>中被清零，所以，读取到 0。
　　因为溢出标志为 0，所以脉冲宽度可以用式（$D_{11} - D_{10}$）计算。（不正确）

备注　　n = 0, 1

在使用两个捕获寄存器时，如果溢出标志由一个捕获寄存器清为 0，则另一个捕获寄存器可能得不到正确的脉冲宽度。
在使用两个捕获寄存器时，使用软件解决。关于如何使用软件的示例如下。

（1/2）

使用两个捕获寄存器时的示例（使用溢出中断）



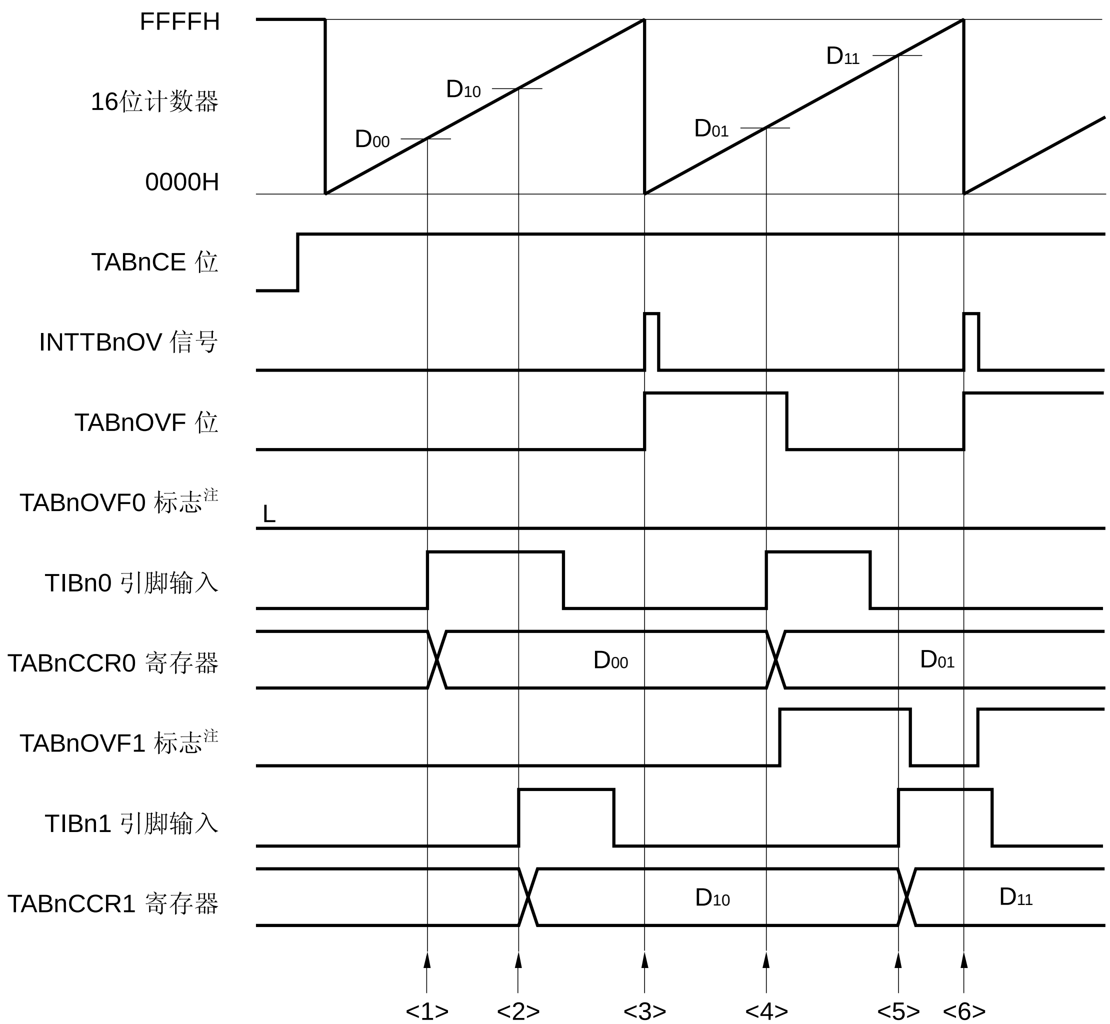
注　　由软件对内部 RAM 中的 TABnOVF0 和 TABnOVF1 标志进行设置。

<1> 读取 TABnCCR0 寄存器（TIBn0 引脚输入的默认值设置）。
<2> 读取 TABnCCR1 寄存器（TIBn1 引脚输入的默认值设置）。
<3> 发生溢出，在溢出中断服务程序中将 TABnOVF0 和 TABnOVF1 标志设置为 1，并将溢出标志清为 0。
<4> 读取 TABnCCR0 寄存器。
读取 TABnOVF0 标志。如果 TABnOVF0 标志为 1，则将其清为 0。
因为 TABnOVF0 标志为 1，所以脉冲宽度可以用式（10000H ＋D_{01} – D_{00}）计算。
<5> 读取 TABnCCR1 寄存器。
读取 TABnOVF1 标志。如果 TABnOVF1 标志为 1，则将其清为 0（TABnOVF0 标志在<4>中已清零，而 TABnOVF1 标志仍然保持为 1）。
因为 TABnOVF1 标志为 1，所以脉冲宽度可以用式（10000H ＋D_{11} – D_{10}）计算。（正确）
<6> 重复 <3> ……

备注　　n = 0, 1

（2/2）

使用两个捕获寄存器时的示例（不使用溢出中断）



注　由软件对内部 RAM 中的 TABnOVF0 和 TABnOVF1 标志进行设置。

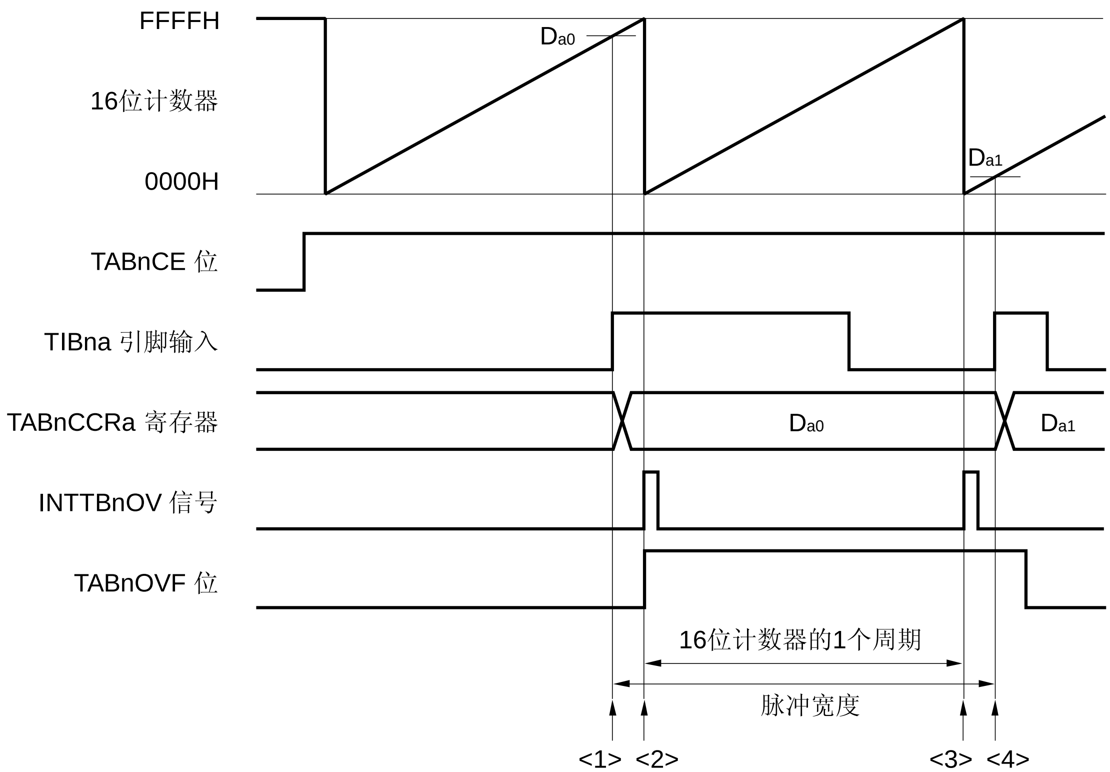
<1> 读取 TABnCCR0 寄存器（TIBn0 引脚输入的默认值设置）。
<2> 读取 TABnCCR1 寄存器（TIBn1 引脚输入的默认值设置）。
<3> 发生溢出，软件不作任何处理。
<4> 读取 TABnCCR0 寄存器。
读溢出标志。若溢出标志为 1，则只把 TABnOVF1 标志置为 1，并把溢出标志清为 0。
因为溢出标志为 1，所以脉冲宽度可以用式（10000H ＋D_{01} – D_{00}）计算。
<5> 读取 TABnCCR1 寄存器。
读溢出标志。因为溢出标志在<4>中被清零，所以读取 0。
读取 TABnOVF1 标志。如果 TABnOVF1 标志为 1，则将其清为 0。
因为 TABnOVF1 标志为 1，所以脉冲宽度可以用式（10000H ＋D_{11} – D_{10}）计算。（正确）
<6> 重复 <3> ……

备注　n = 0, 1

（d）当捕获触发间隔较长时的溢出处理

如果脉冲宽度大于 16 位计数器的一个计数循环周期，则必须加以注意，因为从第一个捕获触发信号到下一个触发信号之间可能发生多次溢出。首先是一个不正确处理的示例，如下所示，。

捕获触发间隔较长时的不正确处理示例



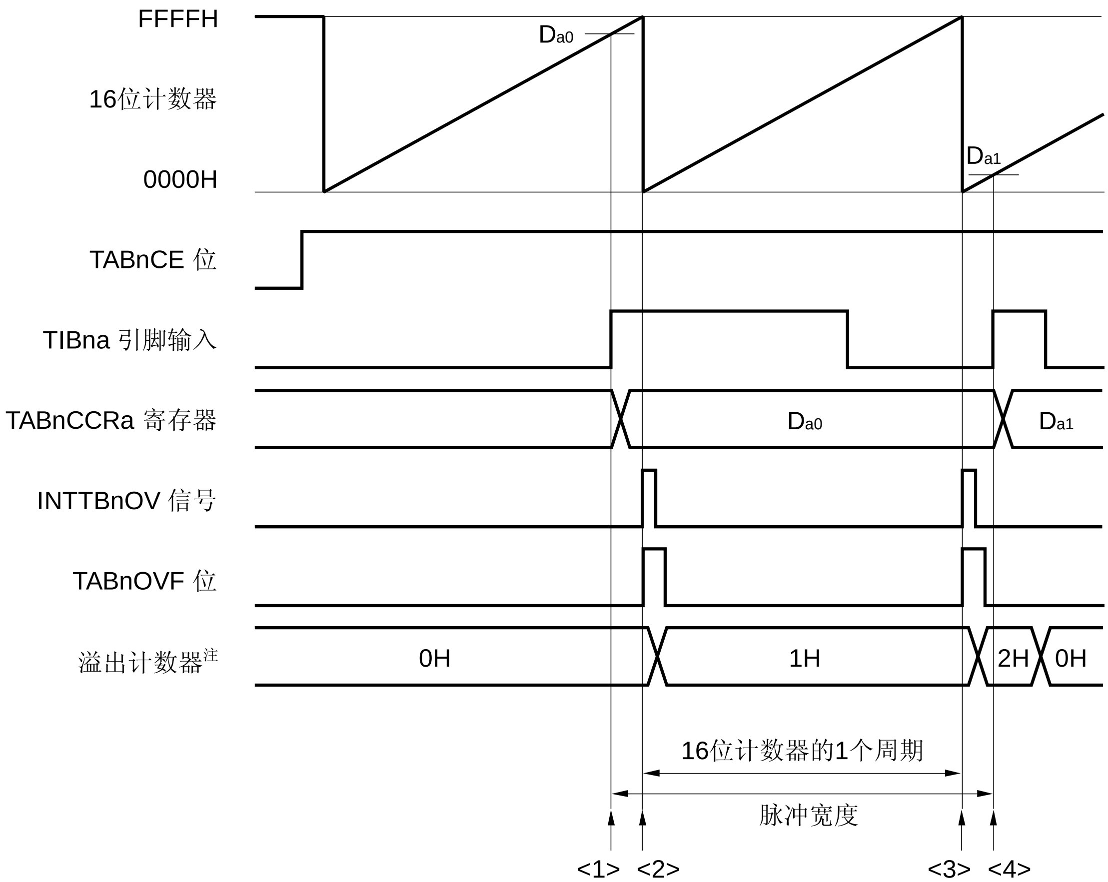
在自由运行定时器模式下测量长脉冲的宽度时，如果脉冲宽度较大，可能出现下列问题。

<1> 读取 TABnCCRa 寄存器（TIBna 引脚输入的默认值设置）。
<2> 发生溢出，软件不作任何处理。
<3> 再次发生溢出。软件不作任何处理。
<4> 读取 TABnCCRa 寄存器。
读溢出标志。若溢出标志为 1，将其清为 0。
因为溢出标志为 1，所以脉冲宽度可以用式（10000H ＋ D_{a1} – D_{a0}）计算。（不正确）
事实上，脉冲宽度应为（20000H ＋ D_{a1} – D_{a0}），因为实际已经发生两次溢出。

备注　n = 0, 1，a = 0 至 3,

如果在捕获触发间隔期间，发生两次或更多次的溢出，则无法获得正确的脉冲宽度。
如果捕获触发间隔时间较长时，应该减慢计数时钟以延长 16 位计数器的循环周期，或者使用软件来解决。
关于如何使用软件，示例如下。

捕获触发间隔较长时的示例



注　　内部 RAM 中的溢出计数器由软件任意设置。

<1>　读取 TABnCCRa 寄存器（TIBna 引脚输入的默认值设置）。
<2>　发生溢出，在溢出中断服务程序中增加溢出计数器的值，并将溢出标志清为 0。
<3>　再次发生溢出。在溢出中断服务程序中增加（ +1）溢出计数器的值，并把溢出标志清为 0。
<4>　读取 TABnCCRa 寄存器。
读取溢出计数器。
→ 当溢出计数器为“N”时，脉冲宽度可用式（N × 10000H ＋ D_{a1} – D_{a0}）计算。
在本例中，脉冲宽度为（20000H ＋ D_{a1} – D_{a0}），因为溢出实际发生了两次。
溢出计数器清零（0H）。

备注　　n = 0, 1
a = 0 至 3,

（e） 清除溢出标志

当 TABnOVF 位为 1 时读取该位之后，通过 CLR 指令将 TABnOVF 位清为 0，且当 TABnOVF 位为 1 时读取该位之后，将 8 位数据（第 0 位是 0）写入 TABnOPT0 寄存器，可以将溢出标志清 0。

7.6.7 脉冲宽度测量模式（TABnMD2 至 TABnMD0 位＝110）

在脉冲宽度测量模式下，当 TABnCTL0.TABnCE 位被置为 1 时，16 位定时器/事件计数器 AB 开始计数。每次检测到 TIBna 引脚输入信号的有效边沿时，16 位计数器的计数值就存储在 TABnCCRa 寄存器中，且 16 位计数器清零为 0000H。

在发生捕获中断请求信号（INTTBnCCa）之后，可以通过读取 TABnCCRa 寄存器来测量有效边沿的间隔。

选择 TIBn0 至 TIBn3 引脚的其中之一作为捕获触发信号输入引脚。使用 TABnIOC1 寄存器将未使用的引脚指定为“无边沿检测”。如图 7-39 所示。

图 7-38. 脉冲宽度测量模式的配置

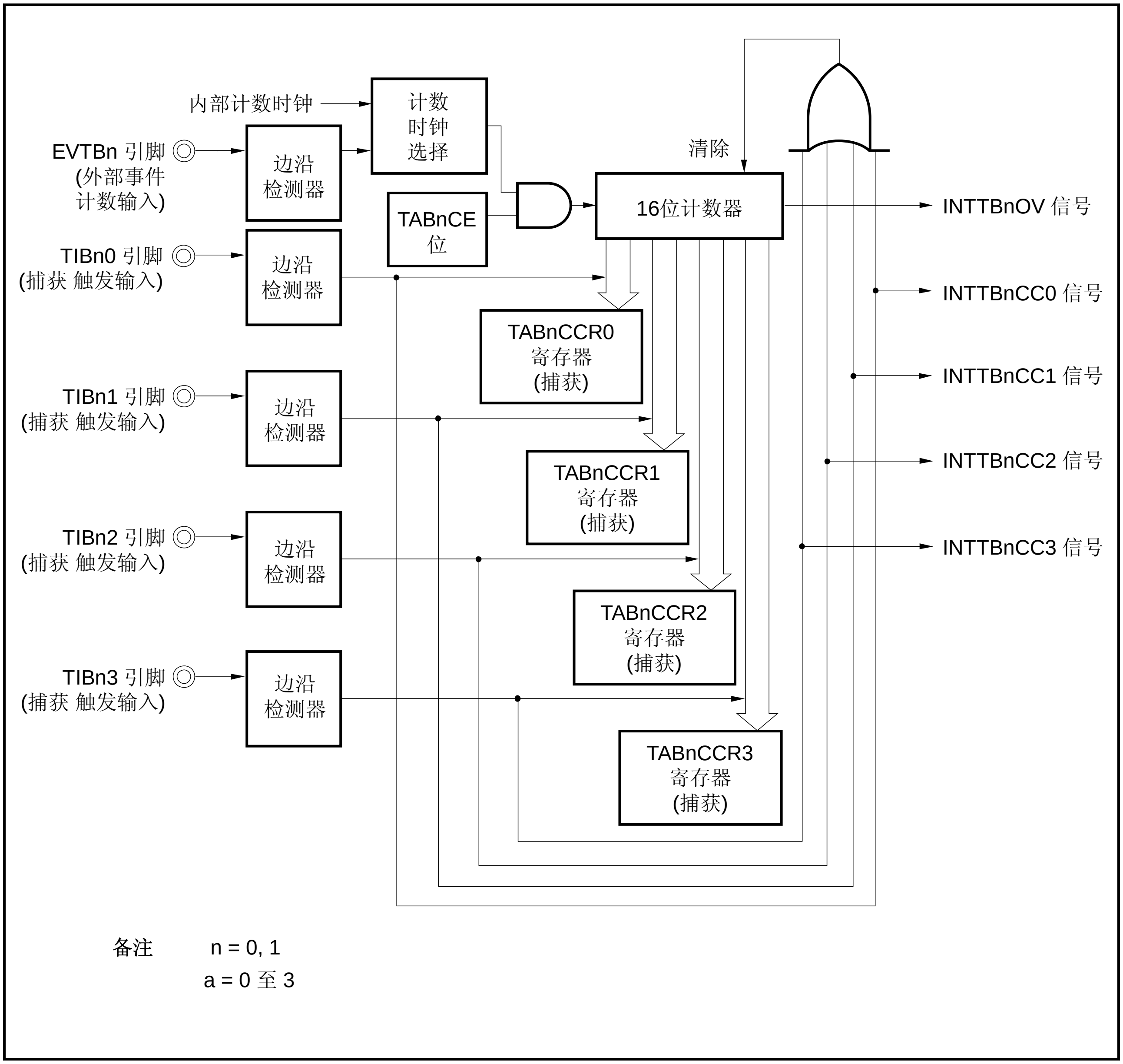
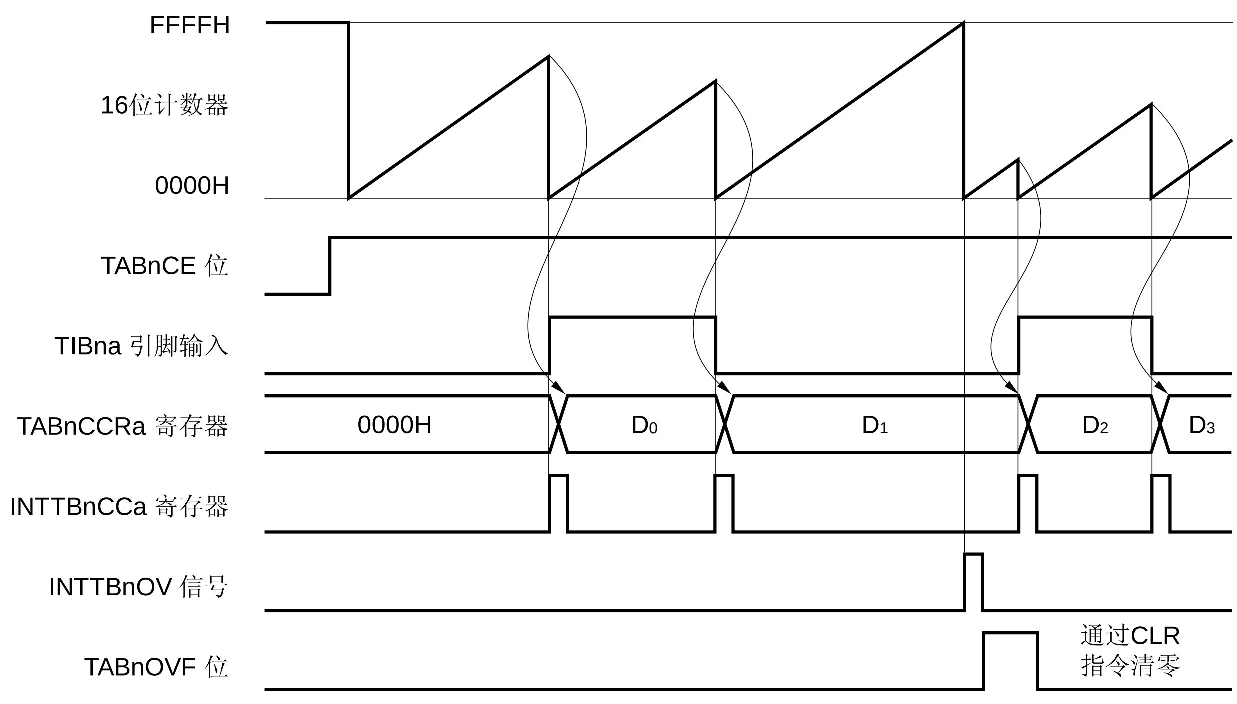


图 7-39. 脉冲宽度测量模式的基本时序



备注 n = 0, 1
a = 0 至 3

当 TABnCE 位被置为 1 时，16 位计数器开始计数。当检测到 TIBna 引脚输入信号的有效边沿时，16 位计数器的计数值被存储到 TABnCCRa 寄存器中，16 位计数器清零为 0000H，并产生一个捕获中断请求信号（INTTBnCCa）。

脉冲宽度的计算如下。

脉冲宽度＝捕获值 × 计数时钟周期

当 16 位计数器计数到 FFFFH 时，即使没有有效边沿输入，在下一个计数时钟时，仍然会产生溢出中断请求信号（INTTBnOV），且计数器清零为 0000H，然后计数器继续计数。此时，溢出标志（TABnOPT0.TABnOVF 位）也被置为 1。使用软件执行 CLR 指令将溢出标志清为 0。

若溢出标志被设置为 1，则脉冲宽度的计算如下。

脉冲宽度＝（10000H × TABnOVF 位置位（1）的次数 ＋捕获值）× 计数时钟周期

备注 n = 0, 1
a = 0 至 3,

图 7-40. 脉冲宽度测量模式时的寄存器设置（1/2）

（a） TABn 控制寄存器 0（TABnCTL0）

	TABnCE					TABnCKS2	TABnCKS1	TABnCKS0
TABnCTL0	0/1	0	0	0	0	0/1	0/1	0/1

TABnCKS2 至 TABnCKS0：选择计数时钟[注]

TABnCE：
0: 停止计数
1: 使能计数

注　　当 TABnCTL1.TABnEEE 位 = 1 时，该设置无效。

（b） TABn 控制寄存器 1（TABnCTL1）

		TABnEST	TABnEEE			TABnMD2	TABnMD1	TABnMD0
TABnCTL1	0	0	0/1	0	0	1	1	0

TABnMD2 至 TABnMD0：
1, 1, 0:
脉冲宽度测量模式

TABnEEE：
0: 由TABnCKS0 至
TABnCKS2 位选择计数时钟
1: 外部事件计数输入信号计数

（c） TABn I/O 控制寄存器 1（TABnIOC1）

	TABnIS7	TABnIS6	TABnIS5	TABnIS4	TABnIS3	TABnIS2	TABnIS1	TABnIS0
TABnIOC1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TABnIS1, TABnIS0：选择TIBn0 引脚输入的有效边沿

TABnIS3, TABnIS2：选择TIBn1 引脚输入的有效边沿

TABnIS5, TABnIS4：选择TIBn2 引脚输入的有效边沿

TABnIS7, TABnIS6：选择TIBn3 引脚输入的有效边沿

（d） TABn I/O 控制寄存器 2（TABnIOC2）

					TABnEES1	TABnEES0	TABnETS1	TABnETS0
TABnIOC2	0	0	0	0	0/1	0/1	0	0

TABnEES1, TABnEES0：选择外部事件计数输入的有效边沿
(EVTBn 引脚)

图 7-40. 脉冲宽度测量模式时的寄存器设置（2/2）

（e） TABn 选项寄存器 0（TABnOPT0）

	TABnCCS3	TABnCCS2	TABnCCS1	TABnCCS0		TABnCMS	TABnCUF	TABnOVF
TABnOPT0	0	0	0	0	0	0	0	0/1

TABnOVF：溢出标志

（f） TABn 计数器读取缓冲寄存器（TABnCNT）

通过读取 TABnCNT 寄存器，可以读出 16 位计数器的值。

（g） TABn 捕获/比较寄存器 0 至 3（TABnCCR0 至 TABnCCR3）

当检测到 TIBna 引脚输入的有效边沿时，这些寄存器保存 16 位计数器的当前计数值。

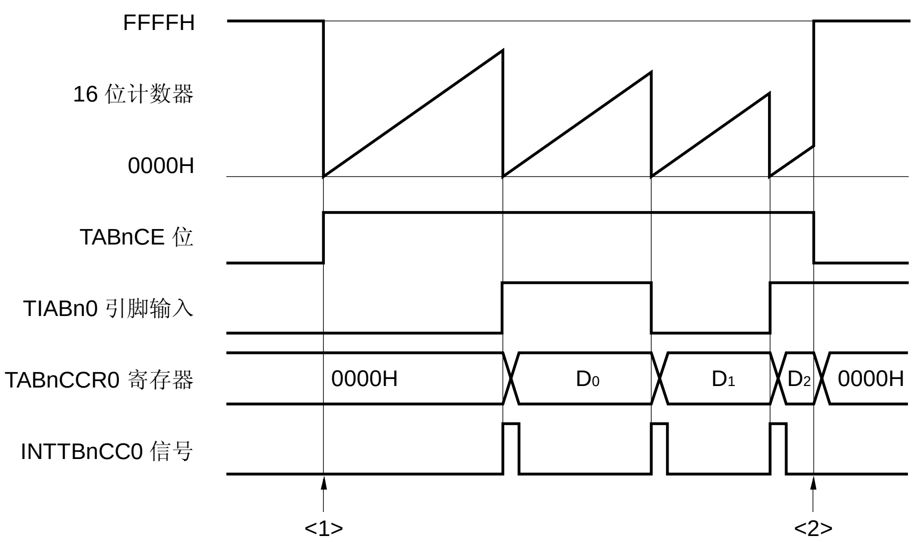
备注　**1.** 在脉冲宽度测量模式下，不使用 TABnI/O 控制寄存器 0（TABnIOC0）。

2. n = 0, 1

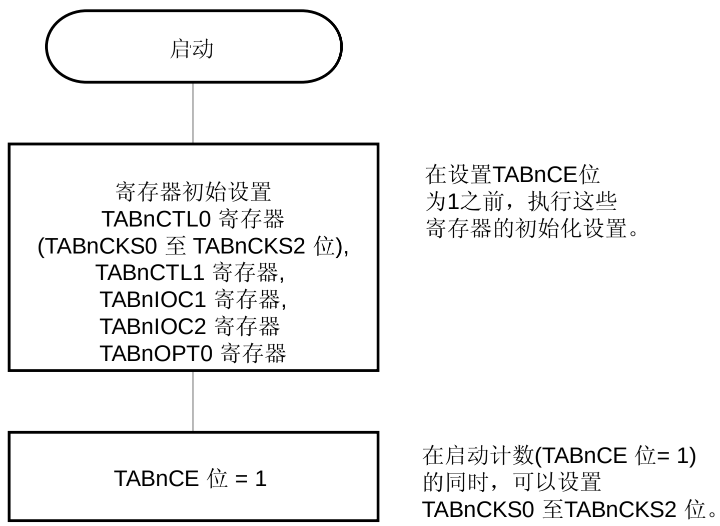
a＝0 至 3

（1）脉冲宽度测量模式时的操作流程

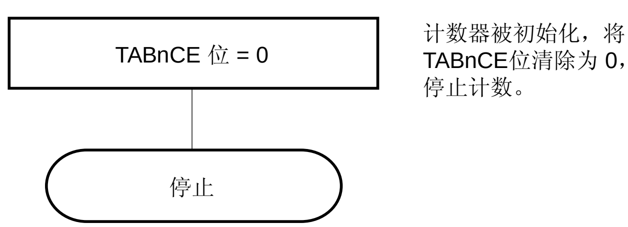
图 7-41. 脉冲宽度测量模式的软件处理流程



<1> 计数操作启动流程



<2> 计数操作停止流程



备注 n = 0, 1

（2）脉冲宽度测量模式时的操作时序

（a） 清除溢出标志

当 TABnOVF 位为 1 时读取该位之后，通过 CLR 指令将 TABnOVF 位清为 0，且当 TABnOVF 位为 1 时读取该位之后，将 8 位数据（第 0 位是 0）写入 TABnOPT0 寄存器，可以将溢出标志清 0。

第八章　16 位定时器/事件计数器 T（TMT）

定时器 T（TMT）是 16 位定时器/事件计数器。

在定时器 AA（TAA）基础上增加了编码器计数功能和其它功能。但是，当 TMT 运行于内部定时器模式时，它不具有通过外部事件计数输入进行操作的功能。

V850E/IF3 和 V850E/IG3 内置 TMT0 和 TMT1。

8.1　概述

TMTn 通道概述如下(n = 0, 1)。

表 **8-1. TMTn** 概述

项目	TMT0	TMT1
时钟选择	8 路	8 路
捕获触发输入引脚	**注 1**	2
外部事件计数输入引脚	**注 2**	1
外部触发输入引脚	**注 2**	1
编码器输入引脚	**注 1**	2
编码器清除输入引脚	**注 2**	1
定时器计数器	1	1
捕获/比较寄存器	2[注 3]	2
捕获/比较一致中断请求信号	2[注 3]	2
溢出中断请求信号	1	1
编码器清除中断请求信号	**注 1**	1
定时器输出引脚	**注 4**	2

注　**1.** V850E/IF3：无
V850E/IG3：2

2. V850E/IF3：无
V850E/IG3：1

3. 在 V850E/IF3 内，仅作比较功能。

8.2 功能

各个通道 TMTn 可以实现的功能，如下表所示(n = 0, 1)。

表 8-2. TMTn 功能

功能	TMT0	TMT1
间隔定时器	√	√
外部事件计数器	**注 1**	√
外部触发脉冲输出	**注 1**	√
单次触发脉冲输出	**注 1**	√
PWM 输出	**注 1**	√
自由运行定时器	√[注 2]	√
脉冲宽度测量	×	√
三角波 PWM 输出模式	**注 1**	√
编码器计数功能	**注 1**	√

注 **1.** V850E/IF3：×
V850E/IG3：√
2. 在 V850E/IF3 内，仅作比较功能

8.3　配置

TMTn 包括下列硬件(n = 0, 1)。

表 8-3. TMTn 的配置

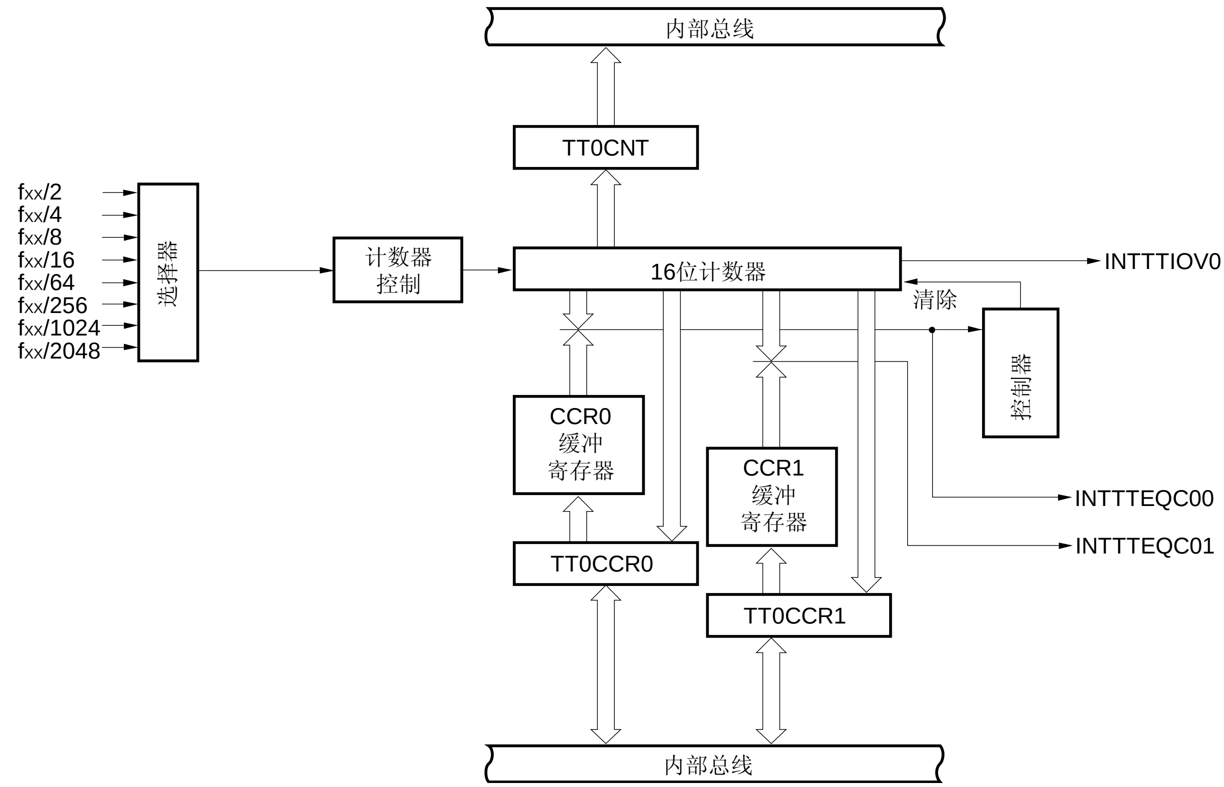
项目	配置
定时器寄存器	16 位计数器 × 1
寄存器	TMTn 捕获/比较寄存器 0, 1 (TTnCCR0, TTnCCR1) TMTn 计数器读取缓冲寄存器 (TTnCNT) TMTm 计数器写入寄存器 (TTmTCW) CCR0 和 CCR1 缓冲寄存器
定时器输入	总共 2 路 (TIT00[注 1], TIT01[注 1], TIT10, TIT11, EVTT0[注 1], EVTT1, TENC00[注 1], TENC01[注 1], TENC10, TENC11, TECR0[注 1], TECR1 引脚)[注 2]
定时器输出	总共 2 路 (TOT00[注 1], TOT01[注 1], TOT10, TOT11 引脚)[注 2]
控制寄存器	TMTn 控制寄存器 0, 1 (TTnCTL0, TTnCTL1) TMTm 控制寄存器 2 (TTmCTL2) TMTm I/O 控制寄存器 0 至 3 (TTmIOC0 至 TTmIOC3) TMTn 选项寄存器 0 (TTnOPT0) TMTm 选项寄存器 1 (TTmOPT1) TMTm 捕获输入选择寄存器 (TTISLm)

注　**1.** 仅限 V850E/IG3

2. TIT00/TECR0 和 TIT10/TECR1 引脚复用作为捕获触发输入引脚 (TIT00, TIT10)，编码器清除输入引脚 (TECR0, TECR1)和定时器输出引脚 (TOT00, TOT10)。
TENC00/EVTT0 和 TENC10/EVTT1 引脚复用作为编码器输入引脚 (TENC00, TENC10)，外部事件计数输入引脚 (EVTT0, EVTT1)和外部触发输入引脚 (EVTT0, EVTT1)。
TIT01/TENC01 和 TIT11/TENC11 引脚复用作为捕获触发输入引脚 (TIT01, TIT11)，编码器输入引脚 (TENC01, TENC11)和定时器输出引脚 (TOT01, TOT11)。

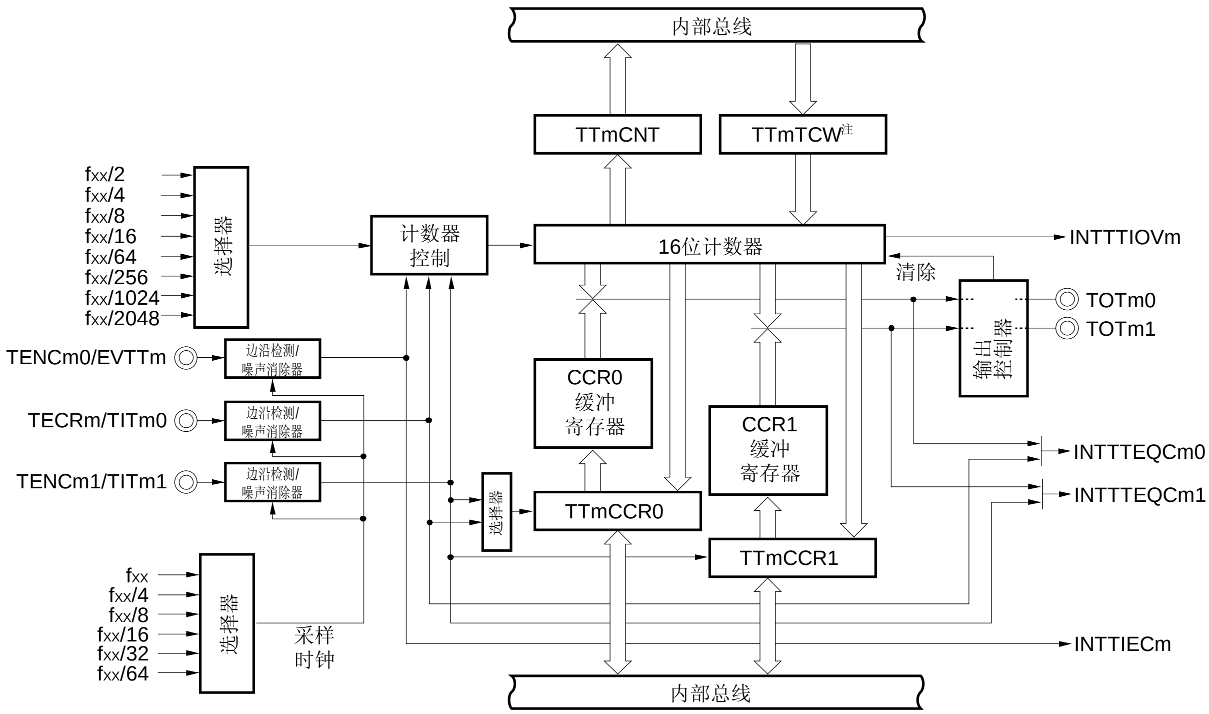
备注　V850E/IF3：n = 0, 1，m = 1
V850E/IG3：n = 0, 1，m = 0, 1

图 8-1. V850E/IF3 中 TMT0 的框图



备注　　fxx：外围时钟

图 8-2. V850E/IF3 中 TMT1 的框图与 V850E/IG3 中 TMT0 和 TMT1 的框图



注　只有在编码器比较模式下，从 TTmTCW 寄存器设置到 16 位计数器的初始值有效。
当 TTmCTL0.TTmCE 位 = 0 时，重写 TTmTCW 寄存器。
当 TTmCE 位 = 1 时，TTmTCW 寄存器的值被传输到 16 位计数器。

备注
1. fxx：外围时钟
2. 关于噪声消除器，参见 **4.6 噪声消除器**。
3. V850E/IF3：m = 1
V850E/IG3：m = 0, 1

(1) 16 位计数器

此 16 位计数器可以计数内部时钟或外部事件。

该计数器的计数值可以通过读取 TTnCNT 寄存器得到。

当 TTnCTL0.TTnCE 位= 0 时，16 位计数器的值为 FFFFH。如果此时读取 TTnCNT 寄存器，则读回的值为 0000H 。

系统复位将 TTnCE 位清 0。

(2) CCR0 缓冲寄存器

CCR0 缓冲寄存器是一个 16 位比较寄存器，用于比较 16 位计数器的计数值。

当 TTnCCR0 寄存器用作比较寄存器时，写入 TTnCCR0 寄存器的值会传输到 CCR0 缓冲寄存器。当 16 位计数器的计数值与 CCR0 缓冲寄存器的值相一致时，就会产生一个比较一致中断请求信号（INTTTEQCn0）。

CCR0 缓冲寄存器不能直接读写。

CCR0 缓冲寄存器在复位后清零为 0000H，TTnCCR0 寄存器清零为 0000H。

(3) CCR1 缓冲寄存器

CCR1 缓冲寄存器是一个 16 位比较寄存器，用于比较 16 位计数器的计数值。

当 TTnCCR1 寄存器用作比较寄存器时，写入 TTnCCR1 寄存器的值会传输到 CCR1 缓冲寄存器。当 16 位计数器的计数值和 CCR1 缓冲寄存器的值一致时，就会产生一个比较一致中断请求信号（INTTTEQCn1）。

CCR1 缓冲寄存器不能直接读写。

CCR1 缓冲寄存器在复位后清零为 0000H，TTnCCR1 寄存器清零为 0000H。

(4) 边沿检测器

该电路检测输入到 TIT00 (仅限 V850E/IG3), TIT01 (仅限 V850E/IG3), TIT10, TIT11, EVTT0 (仅限 V850E/IG3), EVTT1, TENC00 (仅限 V850E/IG3), TENC01 (仅限 V850E/IG3), TENC10, TENC11, TECR0 (仅限 V850E/IG3) 和 TECR1 引脚的有效边沿。通过设置 TTmIOC1, TTmIOC2 和 TTmIOC3 寄存器，可以选择各种有效沿：无边沿、上升沿、下降沿、上升/下降双边沿作为有效沿。

(5) 输出控制器

该电路控制 TOT00 (仅限 V850E/IG3), TOT01 (仅限 V850E/IG3), TOT10 和 TOT11 引脚的输出。输出控制器由 TTmIOC0 寄存器控制。

(6) 选择器

此选择器选择 16 位计数器的计数时钟。可选择 8 种类型内部时钟或一个外部事件作为计数时钟。

(7) 计数器控制

通过 TTnCTL1 寄存器选择的定时器模式来控制计数操作。

8.4 寄存器

(1) TMTn 控制寄存器 0 (TTnCTL0)

TTnCTL0 寄存器是 8 位寄存器，用于控制定时器 TMTn 的操作。
该寄存器可以按字节或按位进行读取和写入。
复位输入将该寄存器设为 00H。
随时可以用软件将相同的值写入 TTnCTL0 寄存器。

复位后： 00H　R/W　地址： TT0CTL0 FFFFF580H, TT1CTL0 FFFFF5C0H

	<7>	6	5	4	3	2	1	0
TTnCTL0 (n = 0, 1)	TTnCE	0	0	0	0	TTnCKS2	TTnCKS1	TTnCKS0

TTnCE	TMTn 操作控制
0	TMTn 操作禁止 (TMTn 异步复位[注])
1	TMTn 操作使能， TMTn 操作开始

TTnCKS2	TTnCKS1	TTnCKS0	内部计数时钟选择
0	0	0	fxx/2
0	0	1	fxx/4
0	1	0	fxx/8
0	1	1	fxx/16
1	0	0	fxx/64
1	0	1	fxx/256
1	1	0	fxx/1024
1	1	1	fxx/2048

注　TTnOPT0.TTnOVF 位和 16 位计数器被同时重置。而且，在 16 位计数器被重置的同时，定时器输出(TOT00 (仅限 V850E/IG3)，TOT01(仅限 V850E/IG3)，TOT10 和 TOT11 引脚)被重置为 TTmIOC0 寄存器设置的状态(V850E/IF3：m = 1, V850E/IG3：m = 0, 1)。

注意事项 **1. 当 TTnCE 位 = 0 时，设置 TTnCKS2 至 TTnCKS0 位。**
当 TTnCE 位的值从 0 改变为 1 时，可以同时设置 TTnCKS2 至 TTnCKS0 位。
2. 请确保将第 3 位至第 6 位清除为“0”。

备注　fxx：外围时钟

(2) TMTn 控制寄存器 1 (TTnCTL1)

TTnCTL1 寄存器是 8 位寄存器，用于控制定时器 TMTn 的操作。
该寄存器可以按字节或按位进行读取和写入。
复位输入将该寄存器设为 00H。

(1/2)

复位后：00H　R/W　地址：　TT0CTL1 FFFFF581H, TT1CTL1 FFFFF5C1H

	7	6	5	4	3	2	1	0
TTnCTL1	0	TTmEST[注]	TTmEEE[注]	0	TTnMD3	TTnMD2	TTnMD1	TTnMD0

V850E/IF3
n = 0, 1
m = 1

V850E/IG3
n = 0, 1
m = 0, 1

TTmEST[注]	软件触发控制
0	-
1	为外部触发输入产生一个有效信号 • 在单次触发脉冲输出模式下： 向TTmEST位写入1作为触发信号，输出一个单次触发脉冲。 • 在外部触发脉冲输出模式下： 向TTmEST位写入1作为触发信号，输出一个PWM波形。
TTmEST位的读取值总是为0.	

TTmEEE[注]	计数时钟选择
0	禁止外部事件计数输入的操作 (EVTTm 引脚). (执行对计数时钟的计数，该时钟由 TTmCTL0.TTmCKS0 至 TTmCTL0.TTmCKS2 位选择.)
1	使能外部事件计数输入的操作 (EVTTm 引脚). (对外部事件计数输入信号 (EVTTm 引脚)的有效边沿执行计数.)
通过TTmEEE位选择使用内部计数时钟计数， 还是使用外部事件计数输入的有效边沿来计数。	

TTnMD3	TTnMD2	TTnMD1	TTnMD0	定时器模式选择
0	0	0	0	间隔定时器模式
0	0	0	1	外部事件计数模式
0	0	1	0	外部触发脉冲输出模式
0	0	1	1	单次触发脉冲输出模式
0	1	0	0	PWM 输出模式
0	1	0	1	自由运行定时器模式
0	1	1	0	脉宽测量模式
0	1	1	1	三角波 PWM 输出模式
1	0	0	0	编码器比较模式
其它情况				禁止设置

注　在 V850E/IF3 内，只有 TMT1 的该位可以置位。请确保将 TMT0 的第 5 位和第 6 位清除为“0”。

(2/2)

注意事项 **1.** 只有在外触发脉冲输出模式或单次触发脉冲输出模式下，**TTmEST** 位有效。在其它模式下，向该位写入 **1** 的操作都被忽略。

2. 只有在间隔定时器模式、外部触发脉冲输出模式、单次触发脉冲输出模式、**PWM** 输出模式、自由运行定时器模式、脉宽测量模式或三角波 **PWM** 输出模式下，**TTmEEE** 位才有效。在其它模式下，向该位写入 **1** 的操作都被忽略。

3. 在外部事件计数模式或编码器比较模式下，无论 **TTmEEE** 位的值如何，总是选择外部事件计数输入 **(EVTTm)** 或编码器输入**(TENCm0，TENCm1)** 。

4. 当 **TTnCTL0.TTnCE** 位**= 0** 时，设置 **TTmEEE** 和 **TTnMD3** 至 **TTnMD0** 位（当 **TTnCE** 位**= 1** 时，可以写入相同的值）。在 **TTnCE** 位**= 1** 时执行重写操作，操作无法保证。若重写被误执行，应把 **TTnCE** 位清为 **0**，再重新对这些位进行设定。

5. 请确保将第 **4** 位和第 **7** 位清为**“0”**。

(3) TMTm 控制寄存器 2 (TTmCTL2)

TTmCTL2 寄存器是 8 位寄存器，用于控制编码器计数功能的操作。
只有在编码器比较模式下，TTmCTL2 寄存器有效。
该寄存器可以按字节或按位进行读取和写入。
复位输入将该寄存器设为 00H。

注意事项　TTmCTL2 寄存器各个位的详情，参见 8.6.9 (5) TTmCTL2 寄存器的控制位。

(1/2)

复位后： 00H　R/W　地址：　TT0CTL2 FFFFF582H 注, TT1CTL2 FFFFF5C2H

	7	6	5	4	3	2	1	0
TTmCTL2	TTmECC	0	0	TTmLDE	TTmECM1	TTmECM0	TTmUDS1	TTmUDS0

(V850E/IF3 m = 1)
(V850E/IG3 m = 0, 1)

TTmECC	编码器计数器控制
0	正常操作
1	当 TTmCTL0.TTmCE 位= 0 时，保持 16 位计数器的计数值．

TTmLDE	16 位计数器的传输设置
0	在下溢时，禁止将 TTmCCR0 的设置值传输给 16 位计数器
1	在下溢时，允许将 TTmCCR0 的设置值传输给 16 位计数器

TTmECM1	编码器清除操作 1 的控制
0	当计数值和 CCR1 寄存器的值一致时， 16 位计数器不被清除为 0000H 。
1	当计数值和 CCR1 寄存器的值一致时， 16 位计数器清除为 0000H 。

TTmECM0	编码器清除操作 0 的控制
0	当计数值和 CCR0 寄存器的值一致时， 16 位计数器不被清除为 0000H 。
1	在计数值和 CCR0 寄存器的值一致之后， 如果计数是向上计数，则 16 位计数器被清除为 0000H 。

注　　仅限 V850E/IG3

(2/2)

TTmUDS1	TTmUDS0	上/下计数选择
0	0	检测到TENCm0 输入的有效边沿时 当TENCm1 = 高电平，向下计数。 当TENCm1 = 低电平，向上计数。
0	1	检测到TENCm0 输入的有效边沿时，向上计数。 检测到TENCm1 输入的有效边沿时，向下计数。
1	0	检测到TENCm0 输入的上升沿时，向下计数。 检测到TENCm0 输入的下降沿时，向上计数。 但是，只有当TENCm1 = 低电平时，才执行计数操作。
1	1	TENCm0 和TENCm1的上升沿和下降沿都检测。 计数操作通过边沿检测和电平检测的组合来自动识别。

注意事项

1. 只有在编码器比较模式下，**TTmECC** 位有效。在其它模式下，向该位写入 **1** 的操作都被忽略。
   如果在 **TTmECC** 位=**1** 时，**TTmCTL0.TTmCE** 位被清除为 **0**，则定时器/计数器和捕获寄存器(**TTmCCR0** 和 **TTmCCR1**)的值，且 **TTmOPT1, TTmEUF, TTmEOF** 和 **TTmESF** 标志被保留。
   如果在 **TTmECC** 位=**1** 时，**TTmCE** 位从 **0** 置为 **1**，**TTmTCW** 寄存器的值不会被传输到 **16** 位计数器。
2. 只有当 **TTmECM1** 和 **TTmECM0** 位= **00, 01** 时，**TTmLDE** 位有效。当 **TTmECM1** 和 **TTmECM0** 位= **10, 11** 时，向该位写入“**1**”的操作都被忽略。
3. 由 **TTmIOC3.TTmEIS1** 和 **TTmIOC3.TTmEIS0** 位指定的 **TENC00** 和 **TENC01** 输入边沿检测无效，且当 **TTmUDS1** 和 **TTmUDS0** 位= **10, 11** 时，固定为双沿检测。
4. 当 **TTmCTL0.TTmCE** 位 = **0** 时，设置 **TTmLDE, TTmECM1, TTmECM0, TTmUDS1** 和 **TTmUDS0** 位（**TTmCE** 位= **1** 时，可以向这些位写入相同的值）。如果在 **TTmCE** 位 = **1** 时更改这些位的值，操作无法保证。若被误更改，应把 **TTmCE** 位清为 **0**，再重新对这些位设定正确值。
5. 请确保将第 **5** 位和第 **6** 位清为“**0**”。

(4) TMTm I/O 控制寄存器 0 (TTmIOC0)

TTmIOC0 寄存器是 8 位寄存器，用于控制定时器的输出（TOTm0，TOTm1 引脚）。

该寄存器可以按字节或按位进行读取和写入。

复位输入将该寄存器设为 00H。

复位后： 00H　　R/W　　地址：　　TT0IOC0 FFFFF583H[注1], TT1IOC0 FFFFF5C3H

TTmIOC0
(V850E/IF3 m = 1)
(V850E/IG3 m = 0, 1)

7	6	5	4	3	<2>	1	<0>
0	0	0	0	TTmOL1	TTmOE1	TTmOL0	TTmOE0

TTmOL1	TOTm1 引脚输出电平设置[注2]
0	TOTm1 引脚输出从高电平开始
1	TOTm1 引脚输出从低电平开始

TTmOE1	TOTm1 引脚输出设置
0	定时器输出禁止 • 当 TTmOL1 位 = 0时，从TOTm1 引脚输出低电平 • 当 TTmOL1 位= 1时，从TOTm1 引脚输出高电平
1	定时器输出使能 (从 TOTm1 引脚输出一个脉冲)

TTmOL0	TOTm0 引脚输出电平设置[注 2]
0	TOTm0 引脚输出从高电平开始
1	TOTm0 引脚输出从低电平开始

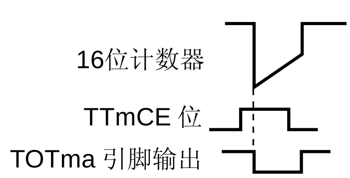
TTmOE0	TOTm0 引脚输出设置
0	定时器输出禁止 • 当 TTmOL0 位 = 0时，从TOTm0 引脚输出低电平 • 当 TTmOL0 位 = 0时，从TOTm0 引脚输出高电平
1	定时器输出使能 (从 TOTm0 引脚输出一个脉冲)

注　1. 仅限 V850E/IG3

2. 在三角波 PWM 输出模式之外的模式下，定时器输出引脚 (TOTm0, TOTm1)的输出电平如下，由 TTmOLa 位指定(a = 0, 1)。

• 当 TTmOLa 位 = 0

• 当 TTmOLa 位 = 1



关于三角波 PWM 输出模式下的输出电平，参见图 **8-44 三角波 PWM 输出模式的基本时序**。

注意事项 1. 当 TOTm0 和 TOTm1 输出被设置为端口模式时，如果 TTmIOC0 寄存器的设置更改，引脚的输出改变。设置该端口为输入模式且置为高阻抗状态，引脚状态不会有任何改变。

2. 当 TTmCTL0.TTmCE 位=0 时，重新写入 TTmOL1, TTmOE1, TTmOL0 和 TTmOE0（当 TTmCE 位=1 时，可以写入相同值）。如果重新被误执行，将 TTmCE 位清为 0，然后重新设定该位。

3. 即使在 TTmCE, TTmOE0 和 TTmOE1 位均为 0 时对 TTmOL0 或 TTmOL1 位进行操作，TOTm0 和 TOTm1 引脚的输出电平也会改变。

(5) TMTm I/O 控制寄存器 1 (TTmIOC1)

TTmIOC1 寄存器是 8 位寄存器，用于控制捕获触发输入信号（TITm0，TITm1 引脚）的有效边沿。
该寄存器可以按字节或按位进行读取和写入。
复位输入将该寄存器设为 00H。

复位后：00H　R/W　地址：TT0IOC1 FFFFF584H[注], TT1IOC1 FFFFF5C4H

	7	6	5	4	3	2	1	0
TTmIOC1	0	0	0	0	TTmIS3	TTmIS2	TTmIS1	TTmIS0

(V850E/IF3 m = 1)
(V850E/IG3 m = 0, 1)

TTmIS3	TTmIS2	捕获触发输入信号 (TITm1 引脚) 有效边沿设置
0	0	无边沿检测 (捕获操作无效)
0	1	检测上升沿
1	0	检测下降沿
1	1	检测双沿

TTmIS1	TTmIS0	捕获触发输入信号 (TITm0 引脚) 有效边沿设置
0	0	无边沿检测 (捕获操作无效)
0	1	检测上升沿
1	0	检测下降沿
1	1	检测双沿

注　仅限 V850E/IG3

注意事项 1. **当 TTmCTL0.TTmCE 位 = 0 时，重新写入 TTmIS3 至 TTmIS0 位（当 TTmCE 位 = 1 时，可以写入相同值）。如果重写被误执行，将 TTmCE 位清为 0，然后重新进行位的设定。**

2. 只有在自由运行定时器模式(仅当 TTmOPT0.TTmCCS1 位= 1)和脉冲宽度测量模式下，TTmIS3 和 TTmIS2 位才有效。在所有其它模式下，无法进行捕获操作。
只有在自由运行定时器模式(仅当 TTmOPT0.TTmCCS0 位= 1)和脉冲宽度测量模式下，TTmIS1 和 TTmIS0 位才有效。在所有其它模式下，无法进行捕获操作。

(6) TMTm I/O 控制寄存器 2 (TTmIOC2)

TTmIOC2 寄存器是 8 位寄存器，用于控制外部事件计数器输入信号（EVTTm 引脚）和外部触发输入信号（EVTTm 引脚）的有效边沿。

该寄存器可按字节读写或按位读写。

复位输入将该寄存器设为 00H。

复位后： 00H R/W 地址：TT0IOC2 FFFFF585H注, TT1IOC2 FFFFF5C5H

	7	6	5	4	3	2	1	0
TTmIOC2	0	0	0	0	TTmEES1	TTmEES0	TTmETS1	TTmETS0

(V850E/IF3 m = 1)
(V850E/IG3 m = 0, 1)

TTmEES1	TTmEES0	外部事件计数输入信号 (EVTTm 引脚) 有效边沿设置
0	0	无边沿检测 (外部事件计数无效)
0	1	检测上升沿
1	0	检测下降沿
1	1	检测双沿

TTmETS1	TTmETS0	外部触发输入信号 (EVTTm 引脚) 有效边沿设置
0	0	无边沿检测 (外部触发无效)
0	1	检测上升沿
1	0	检测下降沿
1	1	检测双沿

注 仅限 V850E/IG3

注意事项 **1.** 当 **TTmCTL0.TTmCE** 位**= 0** 时，重新写入 **TTmEES1, TTmEES0, TTmETS1** 和 **TTmETS0** 位（当 **TTmCE** 位**= 1**，可以写入相同的值）。如果重写被误执行，将 **TTmCE** 位清为 **0**，然后重新进行位的设定。

2. 只有当 **TTmCTL1.TTmEEE** 位**= 1** 或设置为外部事件计数模式（**TTmCTL1.TTmMD3** 至 **TTmCTL1.TTmMD0=0001**）时，**TTmEES1** 和 **TTmEES0** 位有效。

3. 只有在外部触发脉冲输出模式或在单次触发脉冲输出模式下，**TTmETS1** 和 **TTmETS0** 位有效。

(7) TMTm I/O 控制寄存器 3 (TTmIOC3)

TTmIOC3 寄存器是 8 位寄存器，用于控制编码器清除功能的操作。

只有在编码器比较模式下，TTmIOC3 寄存器有效。

该寄存器可按字节读写或按位读写。

复位输入将该寄存器设为 00H。

(1/2)

复位后： 00H R/W 地址： TT0IOC3 FFFFF586H注, TT1IOC3 FFFFF5C6H

	7	6	5	4	3	2	1	0
TTmIOC3	TTmSCE	TTmZCL	TTmBCL	TTmACL	TTmECS1	TTmECS0	TTmEIS1	TTmEIS0

(V850E/IF3 m = 1)

(V850E/IG3 m = 0, 1)

TTmSCE	编码器清除选择
0	检测到编码器清除信号的边沿时，清除 16位计数器。 (TECRm 引脚)
1	检测到TENCm0, TENCm1, 与 TECRm 引脚的清除电平条件时， 清除16位计数器。

- 当TTmSCE 位 = 0时，检测到TTmECS1 与 TTmECS0 位指定的 TECRm引脚有效边沿，则清除16位计数器为0000H。
- 当TTmSCE 位= 1时，TTmZCL，TTmBCL 和 TTmACL位的清除电平条件和TECRm，TENCm1和TENCm0引脚的输入电平匹配，则清除16位计数器为0000H。
- 当TTmSCE 位= 1时，TTmZCL，TTmBCL 和 TTmACL位的设置有效，且 TTmECS1 和 TTmECS0位的设置无效。
  不产生编码器清除中断请求信号 (INTTTIECn) 。
- 当TTmSCE 位= 0时，TTmZCL，TTmBCL 和 TTmACL位的设置无效，且 TTmECS1 和 TTmECS0 位的设置有效。
  当检测到TTmECS1 和 TTmECS0 位指定的有效信边沿时，产生 INTTIECn 信号。
- 当TTmSCE 位= 1时，请确保设置 TTmCTL2.TTmUDS1 与TTmCTL2.TTmUDS0 位为 10 或 11。
  如果TTmUDS1 与 TTmUDS0 位= 00 或 01且TTmSCE 位= 1，则操作无法保证。

TTmZCL	编码器清除信号的清除电平选择(TECRm 引脚)
0	TECRm 引脚低电平时清除
1	TECRm 引脚高电平时清除
仅 当 TTmSCE 位 = 1时，TTmZCL位的设置有效	

TTmBCL	编码器输入信号的清除电平选择 (TENCm1 引脚)
0	TENCm1 引脚低电平时清除
1	TENCm1 引脚高电平时清除
仅当 TTmSCE 位 = 1时，TTmBCL位的设置有效	

TTmACL	编码器输入信号的清除电平选择 (TENCm0 引脚)
0	TENCm0 引脚低电平时清除
1	TENCm0 引脚高电平时清除
仅当 TTmSCE 位 = 1时，TTmACL位的设置有效	

注 仅限 V850E/IG3

(2/2)

TTmECS1	TTmECS0	编码器清除信号 (TECRm 引脚)的有效边沿设置
0	0	无边沿检测 (清除编码器无效)
0	1	检测上升沿
1	0	检测下降沿
1	1	检测双沿

TTmEIS1	TTmEIS0	编码器输入信号 (TENCm0, TENCm1 引脚)的有效边沿设置
0	0	无边沿检测 (编码器输入无效)
0	1	检测上升沿
1	0	检测下降沿
1	1	检测双沿

注意事项 1. 当 **TTmCTL0.TTmCE** 位= **0** 时，重新写入 **TTmSCE, TTmZCL, TTmBCL, TTmACL, TTmECS1, TTmECS0, TTmEIS1** 和 **TTmEIS0** 位（当 **TTmCE** 位= **1**，可以写入相同的值）。如果重写被误执行，将 **TTmCE** 位清为 **0**，然后重新进行位的设定。

2. 只有当 **TTmSCE** 位 = **0** 且设置为编码器比较模式时，**TTmECS1** 和 **TTmECS0** 位有效。

3. 只有当 **TTmCTL2.TTmUDS1** 和 **TTmCTL2.TTmUDS0** 位 = **00** 或 **01** 时，**TTmEIS1** 和 **TTmEIS0** 位有效。

(8) TMTn 选项寄存器 0 (TTnOPT0)

TTnOPT0 寄存器为 8 位寄存器，用来设置捕获/比较操作和检测溢出。

该寄存器可以按字节或按位进行读取和写入。

复位输入将该寄存器设为 00H。

复位后： 00H R/W 地址：TT0OPT0 FFFFF587H, TT1OPT0 FFFFF5C7H

	7	6	5	4	3	2	1	<0>
TTnOPT0	0	0	TTmCCS1注	TTmCCS0注	0	0	0	TTnOVF

V850E/IF3 n = 0, 1 m = 1

V850E/IG3 n = 0, 1 m = 0, 1

TTmCCS1注	TTmCCR1 寄存器捕获/比较选择
0	选择比较寄存器
1	选择捕获寄存器 (通过 TTmCTL0.TTmCE 位 = 0来清除)
只有在自由运行定时器模式下，TTmCCS1 位的设置有效	

TTmCCS0注	TTmCCR0 寄存器捕获/比较选择
0	选择比较寄存器
1	选择捕获寄存器 (通过 TTmCTL0.TTmCE 位 = 0来清除)
只有在自由运行定时器模式下，TTmCCS0 位的设置有效	

TTnOVF	TMTn 溢出检测标志
设置 (1)	发生溢出
复位 (0)	TTnOVF 位写入0， 或者 TTnCTL0.TTnCE 位 = 0

- 在自由运行定时器模式或脉宽测量模式下，当16 位计数器的计数值从FFFFH 溢出到 0000H时，TTnOVF位被置为1。
- 当TTnOVF位被置为 1 时，产生一个溢出中断请求信号(INTTTIOVn) 。在自由运行定时器模式和脉宽测量模式之外的模式下，不产生 INTTTIOVn 信号。
- 当 TTnOVF 位 = 1时，即使TTnOVF 位或 TTnOPT0寄存器被读取，TTnOVF位也不会被清除
- 在产生INTTTIOVn信号之后，将TTnOVF 位清除为 0前，请确认 (通过读取) TTnOVF 位被置为 1。
- TTnOVF位可读可写，但是不能通过软件将 TTnOVF 位位置1 。在TMTn工作期间写入1 没有影响。

注 在 V850E/IF3 内，只有 TMT1 的该位可以被置位。请确保将 TMT0 的第 4 位和第 5 位清除为“0”。

注意事项 **1.** 当 **TTmCE** 位= **0** 时，重新写入 **TTmCCS1** 和 **TTmCCS0** 位（当 **TTmCE** 位= **1** 时，可以写入相同的值）。如果重写被误执行，将 **TTmCE** 位清为 **0** ，然后重新进行位的设定。

2. 请确保将第 **1** 位至第 **3** 位，第 **6** 位和第 **7** 位清为 **“0”**。

(9) TMTm 选项寄存器 1 (TTmOPT1)

TTmOPT1 寄存器为 8 位寄存器，用来检测溢出，下溢，编码器计数功能的向上/向下计数操作。
只有在编码器比较模式下，TTmOPT1 寄存器有效。
该寄存器可以按字节或按位进行读取和写入。
复位输入将该寄存器设为 00H。
即使当 TTmCTL0.TTmCE 位 = 1 时，也可以重写该寄存器。

(1/2)

复位后： 00H R/W 地址： TT0OPT1 FFFFF588H[注], TT1OPT1 FFFFF5C8H

	7	6	5	4	3	<2>	<1>	<0>
TTmOPT1	0	0	0	0	0	TTmEUF	TTmEOF	TTmESF

(V850E/IF3 m = 1)

(V850E/IG3 m = 0, 1)

TTmEUF	TMTm 下溢检测标志
设置 (1)	发生下溢
复位 (0)	写入TTmEUF 位来清除， 或当 TTmCTL0.TTmCE 位 = 0时清除

- 在编码器比较模式下，当16位计数器从0000H 下溢到FFFFH时，TTmEUF 位被置为 1。
- 当 TTmCTL2.TTmLDE 位 = 1，16位计数器的值从0000H 更改为TTmCCR0寄存器的设置值时，TTmEUF 位被置为 1。
- 当TTmEUF 位被置为 1的同时，产生一个溢出中断请求信号 (INTTTIOVm) 。
- 当TTmEUF 位 = 1时，即使TTmEUF 位或 TTmOPT1寄存器被读取，TTmEUF位也不会被清除。
- 当 TTmCTL2.TTmECC 位 = 1时，即使TTmCTL0.TTmCE位被清除为0，TTmEUF 位的状态被保留。
- 在产生INTTTIOVm信号之后，将 TTmEUF位清除为 0前，请确认 (通过读取) TTmEUF 位被置为 1。
- TTmEUF 位可读可写，但是不能通过软件将 该位置1 。在TMTm工作期间设置该位没有影响。

注　仅限 V850E/IG3

(2/2)

TTmEOF	TMTm 编码器功能的溢出检测标志
置位 (1)	发生溢出
重置 (0)	向 TTmEOF 位写入0来清除，或当TTmCTL0.TTmCE 位 = 0 时清除。

- 在编码器比较模式下，当16位计数器从FFFFH溢出为0000H时，TTmEOF 位被置为 1。
- 当TTmEOF 位被置为 1的同时，产生一个溢出中断请求信号(INTTTIOVm)，同时TTmOPT0.TTmOVF 位不被置 1。
- 当TTmEOF 位 = 1时，即使TTmEOF 位或 TTmOPT1寄存器被读取，TTmEOF位也不会被清除。
- 当TTmCTL2.TTmECC 位 = 1时，即使TTmCTL0.TTmCE位被清除为0，TTmEOF 位的状态被保留。
- 在产生INTTTIOVm信号之后，将TTmEOF位清除为 0前，请确认 (通过读取) TTmEOF位被置为 1。
- TTmEOF 位可读可写，但是不能通过软件将 该位置1 。在TMTm工作期间对该位写入 1 没有影响。

TTmESF	TMTm 向上/向下操作状态检测标志
0	TMTm 向上计数
1	TMTm 向下计数

- 当TTmCTL2.TTmECC 位 = 0时，如果TTmCTL0.TTmCE 位= 0，则该位被清除为 0。
- 当TTmECC 位 = 1时，即使TTmCE 位 = 0，TTmESF 位的状态也会被保留。

注意事项　请确保将第 3 位至第 7 位清为“0”。

(10) TMTm 捕获输入选择寄存器 (TTISLm)

当 TTmCCR0 寄存器用作捕获寄存器时，TTISLm 寄存器用于选择 TITm0 或 TITm1 引脚作为捕获触发输入信号的输入引脚。

该寄存器可以按字节或按位进行读取和写入。

复位输入将使该寄存器不确定。

复位后： 不确定　R/W　地址：TTISL0 FFFFF5A4H[注], TTISL1 FFFFF5A6H

	7	6	5	4	3	2	1	0
TTISLm	0	0	0	0	0	0	0	TTISLm

(V850E/IF3 m = 1)

(V850E/IG3 m = 0, 1)

TTISLm	TTmCCR0 寄存器的捕获触发输入信号选择
0	TITm0 输入
1	TITm1 输入

注　仅限 V850E/IG3

(11) TMTn 捕获/比较寄存器 0 (TTnCCR0)

TTmCCR0 寄存器是 16 位寄存器，根据不同模式，可以被用作捕获寄存器或比较寄存器。V850E/IF3 的 TT0CCR0 寄存器是 16 位寄存器，只能用作比较寄存器。

只有在自由计时器模式下，根据 TTmOPT0.TTmCCS0 位的设置，该寄存器可用作捕获寄存器或比较寄存器。在脉宽测量模式中，TTmCCR0 寄存器只可以被用作捕获寄存器。而在其它模式时，该寄存器只可以用作比较寄存器。

在运行期间，TTnCCR0 寄存器可以进行读取和写入。

该寄存器可以按 16 位宽度进行读取或写入。

复位输入将该寄存器清为 0000H。

备注 V850E/IF3：n = 0, 1, m = 1
V850E/IG3：n = 0, 1, m = 0, 1

复位后： 0000H R/W 地址： TT0CCR0 FFFFF58AH, TT1CCR0 FFFFF5CAH

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TTnCCR0 (n = 0, 1)																

(a) 用作比较寄存器

即便在 TTnCTL0.TTnCE 位=1 时，TTnCCR0 寄存器仍可以重新写入。

TTnCCR0 寄存器的设置值传输给 CCR0 缓冲寄存器。当 16 位计数器的值与 CCR0 缓冲寄存器的值相一致时，就会生成一个比较一致中断请求信号（INTTTEQCn0）。如果此时 TOTm0 引脚输出使能，则 TOTm0 引脚的输出会被反转。

当 TTnCCR0 寄存器在间隔定时器模式下作为周期寄存器时，或在外部事件计数器模式、外部触发脉冲输出模式、单次触发脉冲输出模式、PWM 输出模式三角波 PWM 输出模式或编码器比较模式下，当 TTnCCR0 寄存器作为周期寄存器时，如果 16 位计数器的计数值与 CCR0 缓冲寄存器数值一致，则计数值清零（0000H）。

将 TTnCTL0.TTnCE 位设置为 0，比较寄存器不被清除。

(b) 用作捕获寄存器

在自由运行定时器模式下，当 TTmCCR0 寄存器用作捕获寄存器时，若检测到捕获触发输入引脚（TITm0 引脚）的有效边沿，则 16 位计数器的当前计数值存储于 TTmCCR0 寄存器中。在脉宽测量模式下，若检测到捕获触发输入引脚（TITm0 引脚）的有效边沿，16 位计数器的当前计数值就储存在 TTmCCR0 寄存器中，且 16 位计数器清零（0000H）。

即使捕获操作和读取 TTmCCR0 寄存器的操作发生冲突，还是可以正确读取 TTmCCR0 寄存器的值。

设置 TTmCTL0.TTmCE 位为 0，捕获寄存器被清除。

备注　V850E/IF3：m = 1
　　　V850E/IG3：m = 0, 1

下表显示了捕获/比较寄存器在各工作模式下的功能，以及如何向比较寄存器写入数据。

表 8-4. 各模式下捕获/比较寄存器的功能以及如何写入比较寄存器

操作模式	TTnCCR0 寄存器	如何写入比较寄存器
间隔定时器	比较寄存器	随时写入
外部事件计数器注 1	比较寄存器	随时写入
外部触发脉冲输出注 1	比较寄存器	批量写入注 2
单次触发脉冲输出注 1	比较寄存器	随时写入
PWM输出注 1	比较寄存器	批量写入注 2
自由运行定时器	捕获/比较寄存器	随时写入
脉冲宽度测量注 1	捕获寄存器	无
三角波PWM输出注 1	比较寄存器	批量写入注 2
编码器 比较注 1	比较寄存器	随时写入

注　**1.** 在 V850E/IF3 内，只有 TMT1 可以设置该模式。
　　2. 写入 TTnCCR1 寄存器作为触发。

备注　随时写入和批量写入，参见 **8.6 (2) 随时写入和批量写入**。

(12) TMTn 捕获/比较寄存器 1 (TTnCCR1)

TTmCCR1 寄存器是 16 位寄存器，根据不同模式，可以被用作捕获寄存器或比较寄存器。V850E/IF3 的 TT0CCR1 寄存器是 16 位寄存器，只能用作比较寄存器。

只有在自由计时器模式下，根据 TTmOPT0.TTmCCS1 位的设置，该寄存器可用作捕获寄存器或比较寄存器。

在脉宽测量模式中，TTmCCR1 寄存器只能用作捕获寄存器。而在其它模式时，该寄存器只能用作比较寄存器。

在运行期间，TTnCCR1 寄存器可以进行读取和写入。

该寄存器可以按 16 位宽度进行读取或写入。

复位输入将该寄存器清为 0000H。

备注 V850E/IF3：n = 0, 1, m = 1
V850E/IG3：n = 0, 1, m = 0, 1

复位后：0000H R/W 地址：TT0CCR1 FFFFF58CH, TT1CCR1 FFFFF5CCH

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TTnCCR1 (n = 0, 1)																

(a) 用作比较寄存器

即便在 TTnCTL0.TTnCE 位=1 时，TTnCCR1 寄存器仍可以重新写入。

TTnCCR1 寄存器的设置值传输给 CCR1 缓冲寄存器。当 16 位计数器的值与 CCR1 缓冲寄存器的值相一致时，就会生成一个比较一致中断请求信号（INTTTEQCn1）。如果此时 TOTm1 引脚输出使能，则 TOTm1 引脚的输出会被反转。

将 TTnCTL0.TTnCE 位设置为 0，比较寄存器不被清除。

(b) 用作捕获寄存器

在自由运行定时器模式下，当 TTmCCR1 寄存器用作捕获寄存器时，若检测到捕获触发输入引脚（TITm1 引脚）的有效边沿，则 16 位计数器的当前计数值存储到 TTmCCR1 寄存器中。在脉宽测量模式下，若检测到捕获触发输入引脚（TITm1 引脚）的有效边沿，16 位计数器的计数值存储到 TTmCCR1 寄存器中，且 16 位计数器清零（0000H）。

即使捕获操作和读取 TTmCCR1 寄存器的操作发生冲突，还是可以正确读取 TTmCCR1 寄存器的值。

设置 TTmCTL0.TTmCE 位为 0，捕获寄存器被清除。

备注　　V850E/IF3：m = 1
　　　　V850E/IG3：m = 0, 1

下表显示了捕获/比较寄存器在各工作模式下的功能，以及如何向比较寄存器写入数据。

表 8-5. 各模式下捕获/比较寄存器的功能以及如何写入比较寄存器

操作模式	TTnCCR1 寄存器	如何写入比较寄存器
间隔定时器	比较寄存器	随时写入
外部事件计数器[注 1]	比较寄存器	随时写入
外部触发脉冲输出[注 1]	比较寄存器	批量写入[注 2]
单次触发脉冲输出[注 1]	比较寄存器	随时写入
PWM输出[注 1]	比较寄存器	批量写入[注 2]
自由运行定时器	捕获/比较寄存器	随时写入
脉冲宽度测量[注 1]	捕获寄存器	无
三角波PWM输出[注 1]	比较寄存器	批量写入[注 2]
编码器 比较[注 1]	比较寄存器	随时写入

注　**1.** 在 V850E/IF3 内，只有 TMT1 可以设置该模式。
　　2. 写入 TTnCCR1 寄存器作为触发。

备注　　随时写入和批量写入，参见 **8.6 (2) 随时写入和批量写入**。

(13) TMTm 计数器写入寄存器 (TTmTCW)

TTmTCW 寄存器用于设置 16 位计数器的初始值。
只有在编码器比较模式下，TTmTCW 寄存器有效。
该寄存器可以按 16 位宽度进行读取或写入。
当 TTmCTL0.TTmCE 位 = 0 时，可以重写 TTmTCW 寄存器。
当 TTmCE 位被置位（1）时，TTmTCW 寄存器的值被传输到 16 位计数器。
复位输入将该寄存器清为 0000H。

复位后： 0000H R/W 地址： TT0TCW FFFFF590H注, TT1TCW FFFFF5D0H

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

(V850E/IF3 m = 1)

(V850E/IG3 m = 0, 1)

注 仅限 V850E/IG3

(14) TMTn 计数器读取缓冲寄存器 (TTnCNT)

TTnCNT 寄存器是读出缓冲寄存器 ，可以读取 16 位计数器的计数值。
如果在 TTnCTL0.TTnCE 位 = 1 时读取该寄存器，可以读出 16 位定时器的计数值。
该寄存器为只读寄存器，按 16 位宽度读取。
当 TTmCTL2.TTmECC 和 TTmCE 位=0 时，TTmCNT 寄存器的值被清除为 0000H。如果此时读取 TTmCNT 寄存器，则不会读出 16 位计数器的值（FFFFH），读到的值为 0000H。当 TTmECC 位 = 1 且 TTmCE 位 = 0 时，TTmCNT 寄存器不是被置为 0000H，而是读出上一次的值。
在复位后，TTmECC 和 TTmCE 位被置为 0，TTmCNT 寄存器的值被清除为 0000H。

复位后： 0000H R 地址：TT0CNT FFFFF58EH, TT1CNT FFFFF5CEH

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

(n = 0, 1)

8.5 定时器输出操作

下表列出了 TOTm0 和 TOTm1 引脚的操作和输出电平。

表 8-6. 各个模式下的定时器输出控制

操作模式	TOTm1 引脚	TOTm0 引脚
间隔定时器模式	PWM 输出	
外部事件计数模式[注]	无	
外部触发脉冲输出模式[注]	外部触发脉冲输出	PWM 输出
单次触发脉冲输出模式[注]	单次触发脉冲输出	
PWM输出模式[注]	PWM 输出	
自由运行定时器模式	PWM 输出(仅当使用比较功能时)	
脉冲宽度测量模式[注]	无	
三角波PWM输出模式[注]	三角波 PWM 输出	
编码器 比较模式[注]	无	

注　　在 V850E/IF3 内，只有 TMT1 可以设置该模式。

备注　　V850E/IF3：m = 1，a = 0, 1
V850E/IG3：m = 0, 1，a = 0, 1

表 8-7. 定时器输出控制位控制下的 TOTm0 和 TOTm1 引脚的真值表

TTmIOC0.TTmOLa 位	TTmIOC0.TTmOEa 位	TTmCTL0.TTmCE 位	TOTma 引脚的电平
0	0	×	低电平输出
	1	0	低电平输出
		1	计数前为低电平，计数启动后为高电平
1	0	×	高电平输出
	1	0	高电平输出
		1	计数前为高电平，计数启动后为低电平

备注　　V850E/IF3：m = 1, a = 0, 1
V850E/IG3：m = 0, 1, a = 0, 1

8.6　操作

每个通道的 TMTn 可以实现的功能不同，各个通道的功能列举如下(n = 0, 1)。

表 8-8. 各个模式下的 TMTm 规范

操作模式	TTmCTL1.TTmEST 位 (软件触发位)	EVTTm 引脚 (外部触发输入)	捕获/比较寄存器设置	比较寄存器写入方法
间隔定时器模式	无效	无效	仅作比较	随时写入
外部事件计数模式	无效	无效	仅作比较	随时写入
外部触发脉冲输出模式	有效	有效	仅作比较	批量写入
单次触发脉冲输出模式	有效	有效	仅作比较	随时写入
PWM 输出模式	无效	无效	仅作比较	批量写入
自由运行定时器模式	无效	无效	可切换	随时写入
脉冲宽度测量模式	无效	无效	仅作捕获	不适用
三角波 PWM 输出模式	无效	无效	仅作比较	批量写入
编码器 比较模式	无效	无效	仅作比较	随时写入

备注　V850E/IF3：m = 1
V850E/IG3：m = 0, 1

表 8-9. 各个模式下 V850E/IF3 的 TMT0 规范

操作模式	软件触发位	外部触发输入	捕获/比较寄存器设置	比较寄存器写入方法
间隔定时器模式	无效	无效	仅作比较	随时写入
外部事件计数模式	无			
外部触发脉冲输出模式	无			
单次触发脉冲输出模式	无			
PWM 输出模式	无			
自由运行定时器模式	无效	无效	仅作比较	随时写入
脉冲宽度测量模式	无			
三角波 PWM 输出模式	无			
编码器 比较模式	无			

备注　V850E/IF3 的 TMT0 不具有定时器输入引脚 (TIT00, TIT01, TECR0, TENC00, TENC01, EVTT0)和定时器输出引脚 (TOT00, TOT01)。它有中断请求信号(INTTTEQC00, INTTTEQC01) 表示 16 位计数器的值与 TT0CCR0 和 TT0CCR1 寄存器的值之间发生一致。

(1) 基本计数器操作

本章节解释了 16 位计数器的基本操作。详情参见各种模式下的操作描述。

备注　V850E/IF3：n = 0, 1, m = 1
V850E/IG3：n = 0, 1, m = 0, 1

(a) 计数启动操作

<R>
- 外部事件计数模式
  当 TTmCTL0.TTmCE 位被从 0 设置为 1 时，16 位计数器被设置为 0000H。
  此后，每当检测到外部事件计数输入(EVTTm)的有效边沿时，它向上计数 0001H, 0002H, 0003H, …
- 编码器比较模式
  计数操作通过 TENCm0 和 TENCm1 相来控制。
  将 TTmTCW 寄存器的设置值传输给 16 位计数器并且计数操作启动时(当 TTmCTL2.TTmECC 位 = 0 时，在 TTmCTL0.TTmCE 位从 0 变为 1 的时刻，TTmTCW 寄存器设置值被传输到 16 位计数器)，执行 16 位计数器初始设置。
- 三角波 **PWM** 模式
  16 位计数器从初始值 FFFFH 开始计数。
  从 FFFFH 向上计数，0000H，0001H，0002H，0003H 以此类推。
  在向上操作之后，16 位计数器的值和 CCR0 缓冲寄存器一致时，计数器向下计数。
- 上述之外的模式
  16 位计数器从初始值 FFFFH 开始计数。
  从 FFFFH 向上计数，0000H，0001H，0002H，0003H 以此类推。

(b) 清除操作

当 16 位计数器的值和比较寄存器一致并被清除时，它被清除为 0000H。当 6 位计数器的值被捕获并清除时，当检测到编码器清除信号的边沿并清除时，当检测到 TENCm0, TENCm1 和 TECRm 引脚的清除电平条件并清除时，它被清除为 0000H。在计数器被启动后或当计数器溢出不是清除操作时，立即发生从 FFFFH 到 0000H 的计数操作。因此不产生 INTTTEQCn0 和 INTTTEQCn1 中断信号。

(c) 溢出操作

在自由运行模式、脉宽测量模式和编码器比较模式下，当 16 位计数器从 FFFFH 向上计数到 0000H 时，发生溢出。如果在自由运行模式、脉宽测量模式下发生计数器溢出，TTnOPT0.TTnOVF 位被置为 1，且产生一个中断请求信号(INTTTIOVn)。

如果在编码器比较模式下发生计数器溢出，TTnOPT1.TTnEOF 位被置为 1，且产生一个中断请求信号(INTTTIOVn)。

注意在下列条件时不产生 INTTTIOVn 信号。

- 在计数操作刚刚启动后
- 如果计数器的值和比较值 FFFFH 一致并被清除时
- 在脉宽测量模式下，当捕获到 FFFFH 并清除为 0000H 时

注意事项　**在溢出中断请求信号 (INTTTIOVn) 产生之后，请确保检查溢出标志(TTnOVF, TTmEOF 位) 被置为 1。**

(d) 计数值保持操作

在编码器比较模式下，由 TTmCTL2.TTmECC 位来控制 16 位计数器的值的保存。当 TTmECC 位 = 0 且 TTmCTL0.TTmCE 位 = 0 时，16 位计数器的值被重置为 FFFFH。当 TTmCE 位下次被设置为 1 时，TTmTCW 寄存器的设置值被传输到 16 位计数器，并且执行计数操作。

如果 TTmECC 位 = 1 且 TTmCE 位 = 0，16 位计数器的值被保存。当 TTmCE 位下次被设置为 1 时，计数器从保存值恢复计数操作。

(e) 在计数操作期间读取计数器的操作

在计数操作期间，可以通过 TTnCNT 寄存器来读取 TMTn 的 16 位计数器的值。

当 TTnCTL0.TTnCE 位= 1 时，可以通过 TTnCNT 寄存器来读取 16 位计数器的值。但是，如果在 TTmCTL2.TTmECC 位 = 0 且 TTmCE 位 = 0 时读取 TTmCNT 寄存器，它的值是 0000H。当 TTmECC 位 = 1 且 TTmCE 位 = 0 时读取 TTmCNT 寄存器，读到的值是保存值。

(f) 下溢操作

在编码器比较模式下，当 16 位计数器从 0000H 改变为 FFFFH 时，发生一个 16 位计数器的下溢。当下溢发生时，TTmOPT1.TTmEUF 位被置为 1，且产生一个中断请求信号 (INTTTIOVm) 。

(g) 中断操作

TMTn 产生下列四种类型的中断请求信号

- INTTTEQCn0 中断：　该信号可以作为 CCR0 缓冲寄存器的一致中断请求信号，也可以作为 TTnCCR0 寄存器的捕获中断请求信号。
- INTTTEQCn1 中断：　该信号可以作为 CCR1 缓冲寄存器的一致中断请求信号，也可以作为 TTnCCR1 寄存器的捕获中断请求信号。
- INTTTIOVn 中断：　该信号作为溢出中断请求信号。
- INTTIECm 中断：　该信号作为检测到编码器清除输入(TECRm 引脚)有效边沿的中断请求信号。

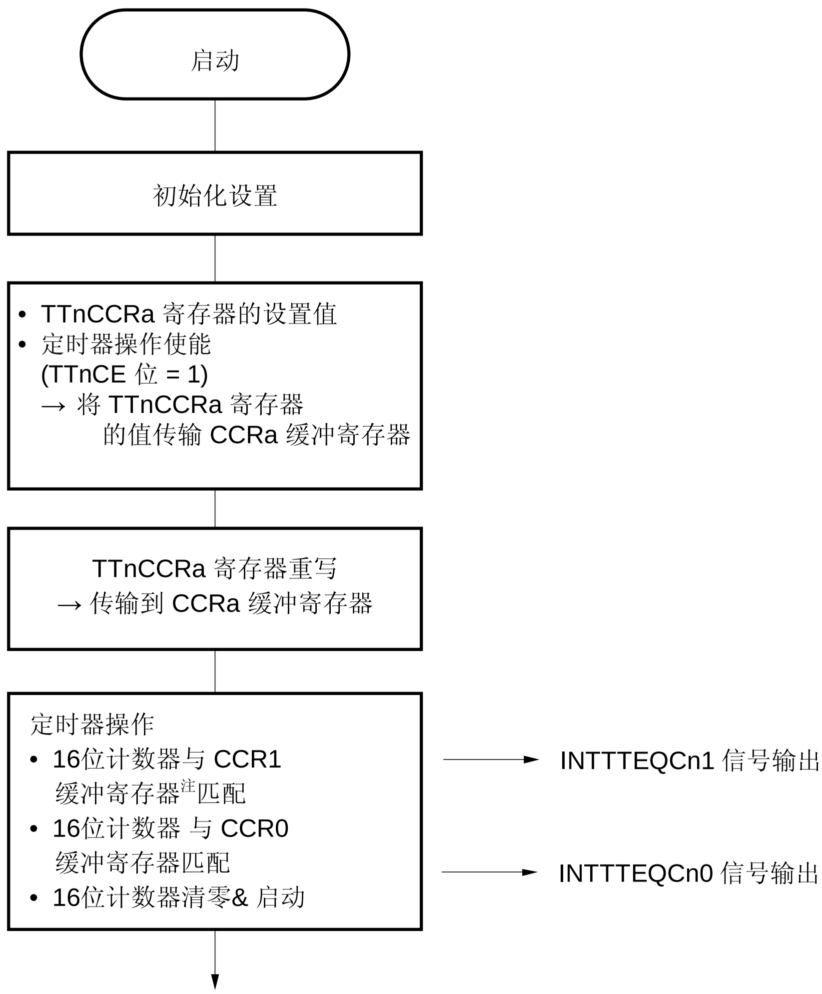
(2) 随时写入和批量写入

在定时器操作期间(TTnCTL0.TTnCE 位 = 1)，可以重写 TMTn 的 TTnCCR0 和 TTnCCR1 寄存器，但是 CCR0 和 CCR1 缓冲寄存器的写入方法（随时写入，批量写入）取决于不同的模式。

(a) 随时写入

该模式下，在定时器操作期间的任意时刻，数据可以从 TTnCCR0 和 TTnCCR1 寄存器传输到 CCR0 和 CCR1 缓冲寄存器 (n = 0, 1)。

图 **8-3.** 随时写入的基本操作流程图

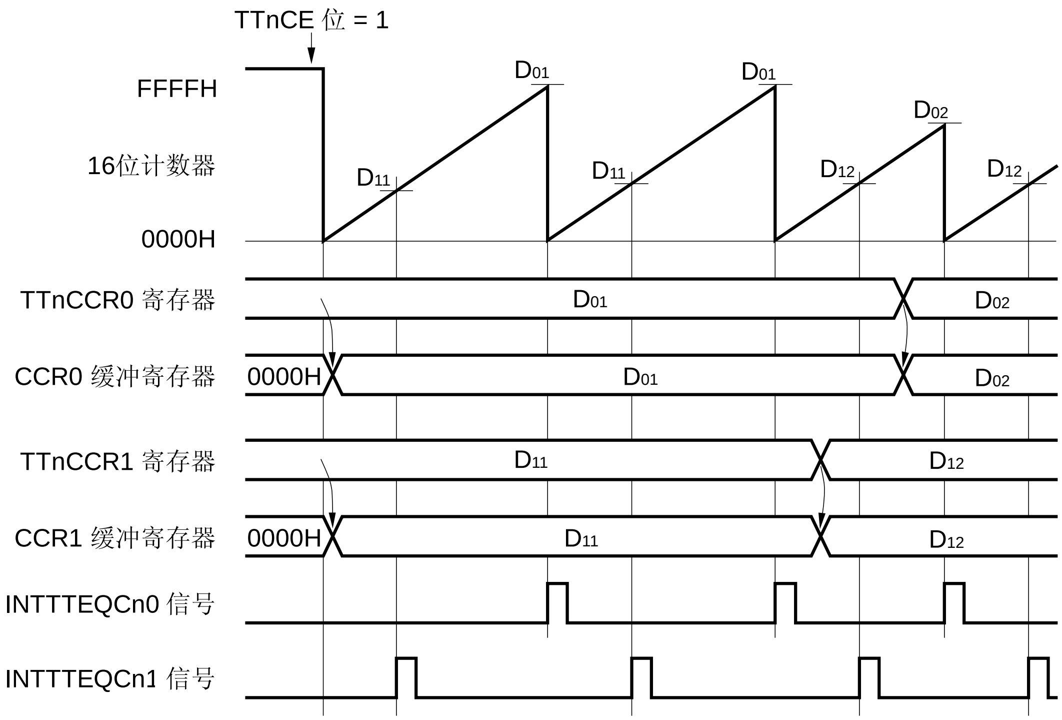


注　**在 16 位计数器的值和 CCR1 缓冲寄存器的值一致时，16 位计数器不清除。在 16 位计数器的值和 CCR0 缓冲寄存器的值一致时，才清除 16 位计数器。**

备注　**1.** 上述流程图展示的是间隔定时器模式下的操作示例。

2. n = 0, 1
a = 0, 1

图 **8-4.** 随时写入的时序示例



备注　1. D_{01}, D_{02}：TTnCCR0 寄存器的设置值
D_{11}, D_{12}：TTnCCR1 寄存器的设置值
2. 上述流程图展示的是间隔定时器模式下的操作示例。
3. n = 0, 1

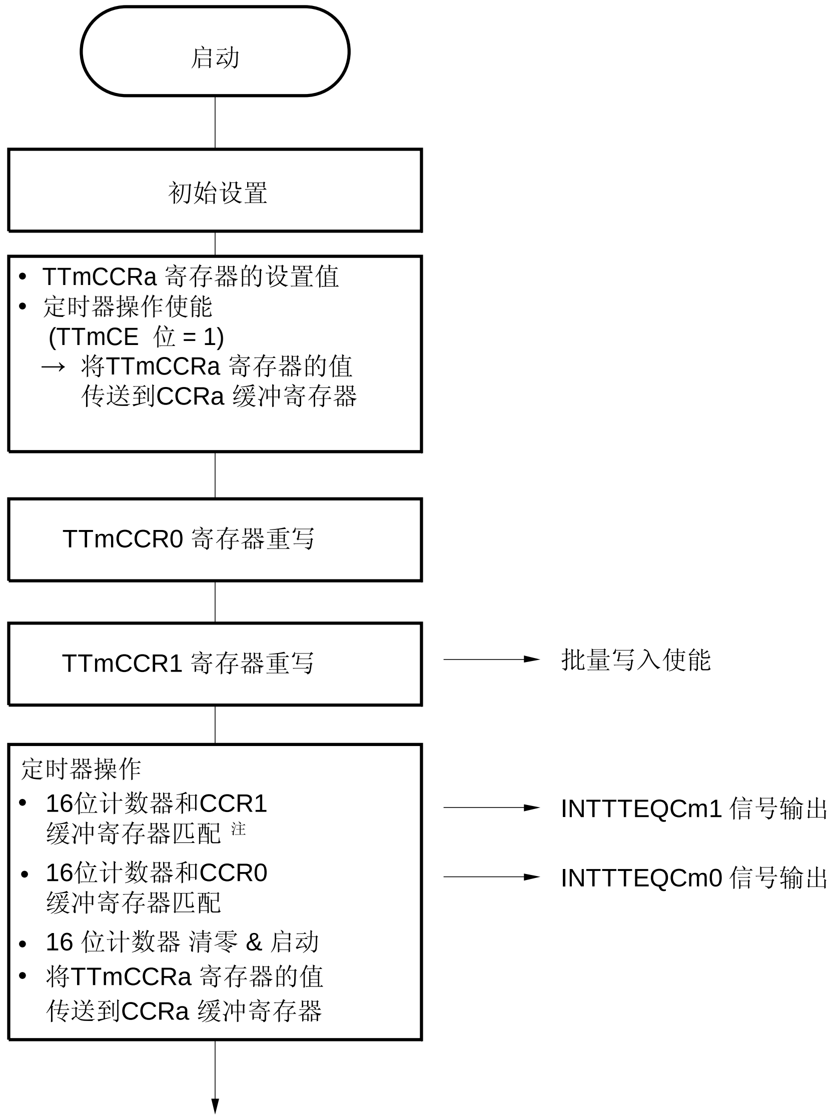
(b) 批量写入

该模式下，在定时器操作期间，数据一次性从 TTmCCR0 和 TTmCCR1 寄存器传输到 CCR0 和 CCR1 缓冲寄存器。在 16 位计数器的值和 CCR0 缓冲寄存器的值一致时，传输该数据。写入 TTmCCR1 寄存器就可以使能传输。是否写入 TTmCCR1 寄存器可以控制使能或禁止下一次传输时序。

为了让重写 TTmCCR0 和 TTmCCR1 寄存器时的设置值成为 16 位计数器比较值（换句话说，为了该值被传输到 CCR0 和 CCR1 缓冲寄存器），必需重写 TTmCCR0 寄存器，然后在 16 位计数器的值和 CCR0 缓冲寄存器的值一致之前写入 TTmCCR1 寄存器。因此，在 16 位计数器的值和 CCR0 缓冲寄存器的值一致时，TTmCCR0 和 TTmCCR1 寄存器的值被传输到 CCR0 和 CCR1 缓冲寄存器。所以，即使只想重写 TTmCCR0 寄存器的值，也需要将相同的值（和 TTmCCR1 寄存器的当前值相同）写入 TTmCCR1 寄存器。

备注　V850E/IF3：m = 1
　　　V850E/IG3：m = 0, 1

图 8-5. 批量写入的基本操作流程图示例

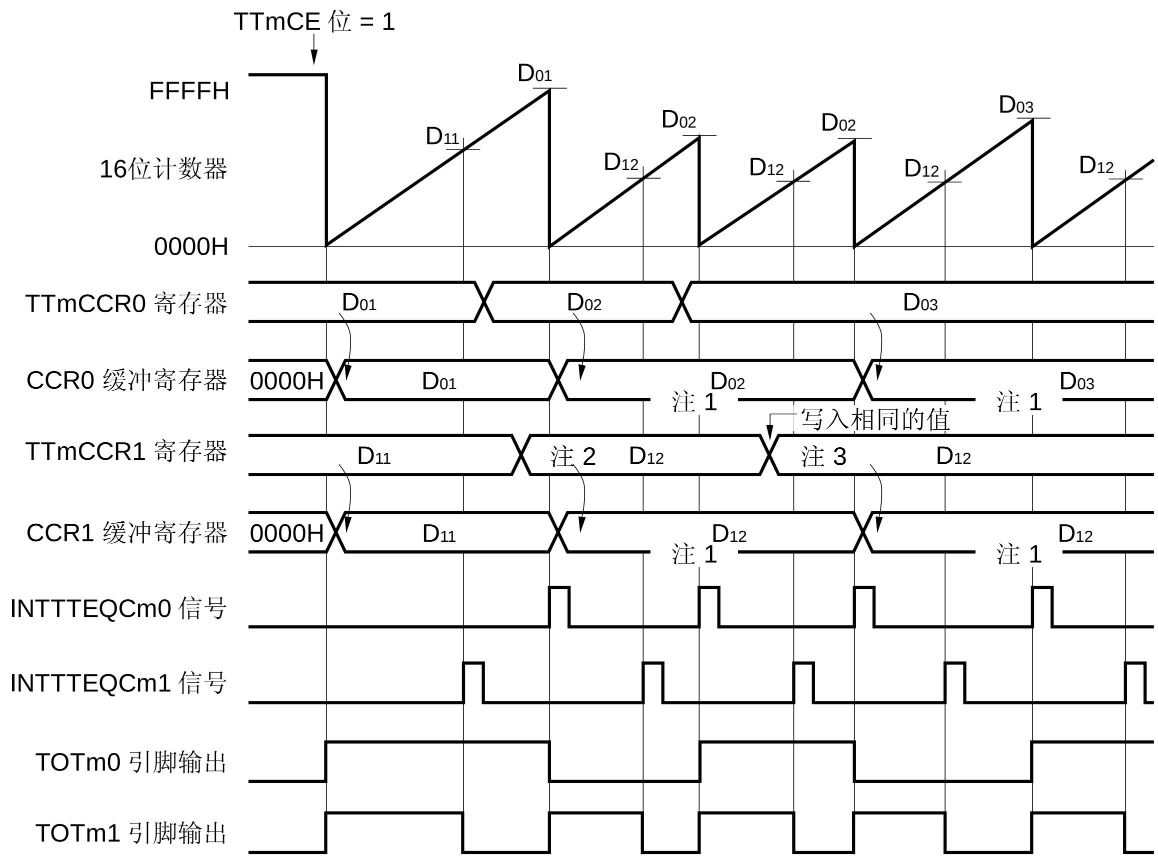


注　在 16 位计数器的值和 CCR1 缓冲寄存器的值一致时，16 位计数器不清除。在 16 位计数器的值和 CCR0 缓冲寄存器的值一致时，才清除 16 位计数器。

注意事项　写入 TTmCCR1 寄存器包括批量写入的使能。于是在重写 TTmCCR0 寄存器之后重写 TTmCCR1 寄存器。

备注　**1.** 上述流程图展示的是 PWM 输出模式下的操作示例。
2. V850E/IF3：m = 1, a = 0, 1
V850E/IG3：m = 0, 1, a = 0, 1

图 8-6. 批量写入的时序示例



注 **1.** 因为 TTmCCR1 寄存器没有重写，D_{03} 没有被传输。

2. 因为 TTmCCR1 寄存器已经被重写(D_{12})，在 16 位计数器的值和 TTmCCR0 寄存器的值(D_{01})一致时，数据被传输到 CCR1 缓冲寄存器 。

3. 因为 TTmCCR1 寄存器已经被重写(D_{12})，在 16 位计数器的值和 TTmCCR0 寄存器的值(D_{02})一致时，数据被传输到 CCR1 缓冲寄存器 。

备注 1. D_{01}, D_{02}, D_{03}：TTmCCR0 寄存器的设置值
D_{11}, D_{12}：TTmCCR1 寄存器的设置值

2. 上述流程图展示的是 PWM 输出模式下的操作示例。

3. V850E/IF3：m = 1
V850E/IG3：m = 0, 1

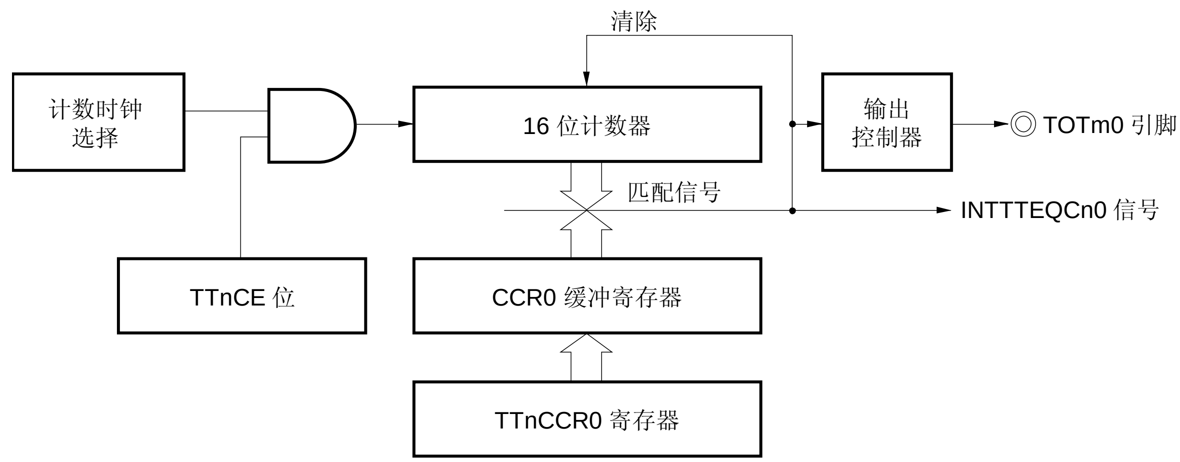
8.6.1 间隔定时器模式 (TTnMD3 至 TTnMD0 位 = 0000)

在间隔定时器模式下，如果 TTnCTL0.TTnCE 位被置为 1，则会以 TTnCCR0 寄存器指定的时间间隔生成中断请求信号（INTTTEQCn0），从 TOTm0 引脚输出占空比为 50%的 PWM 波形，其半周期等于指定的时间间隔。

间隔定时器模式不使用 TTnCCR1 寄存器。但是 TTnCCR1 寄存器的设置值被传输到 CCR1 缓冲寄存器，当 16 位计数器的计数值和 CCR1 缓冲寄存器一致时，产生一个比较一致中断请求信号 (INTTTEQCn1)。并且，从 TOTm1 引脚输出占空比 50%的 PWM 波形，在产生 INTTTEQCm1 信号时电平被反转。

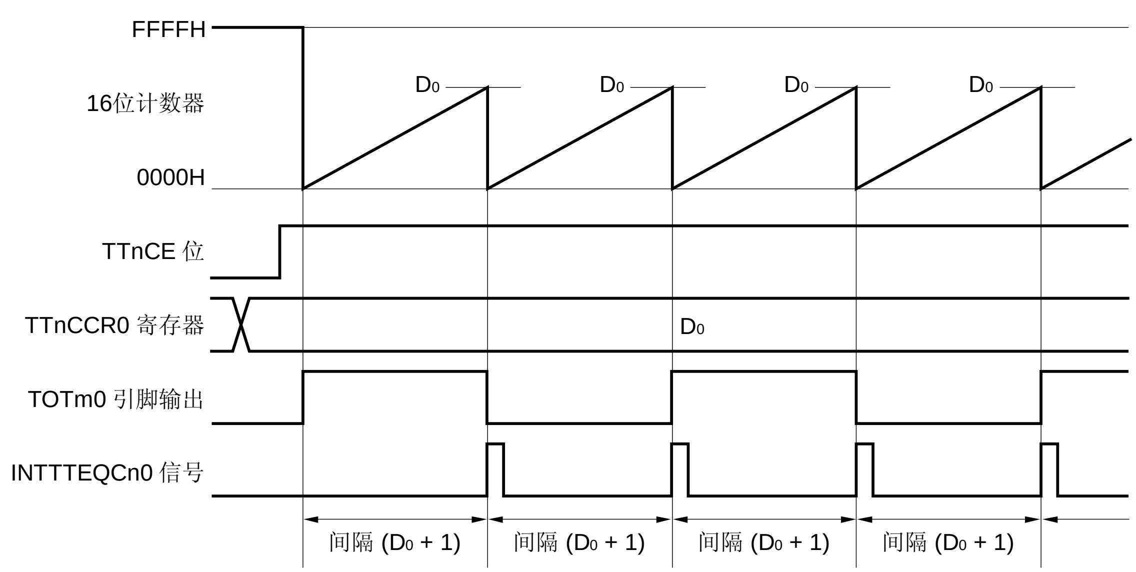
即使在定时器操作期间，也可以重写 TTnCCR0 和 TTnCCR1 寄存器的值。

图 8-7. 间隔定时器的配置



备注 V850E/IF3：m = 1, n = 0, 1
V850E/IG3：m = 0, 1, n = 0, 1

图 8-8. 间隔定时器模式操作的基本时序



备注 V850E/IF3：m = 1, n = 0, 1
V850E/IG3：m = 0, 1, n = 0, 1

当 TTnCE 位被置为 1 时，16 位计数器的值随着计数时钟从 FFFFH 被同步清零为 0000H，计数器开始计数。此时，TOTm0 引脚的输出被反转。此外，TTnCCR0 寄存器的设置值传输给 CCR0 缓冲寄存器。

当 16 位计数器的计数值和 CCR0 缓冲寄存器的值相一致时，则 16 位计数器被清除为 0000H，TOTm0 引脚输出反相，并产生一个比较一致中断请求信号（INTTTEQCn0）。

时间间隔可用下述表达式计算。

时间间隔 =（TTnCCR0 寄存器的设置值＋1） × 计数时钟周期

备注　V850E/IF3：n = 0, 1, m = 1
　　　V850E/IG3：n = 0, 1, m = 0, 1

图 8-9. 间隔定时器模式操作的寄存器设置 (1/2)

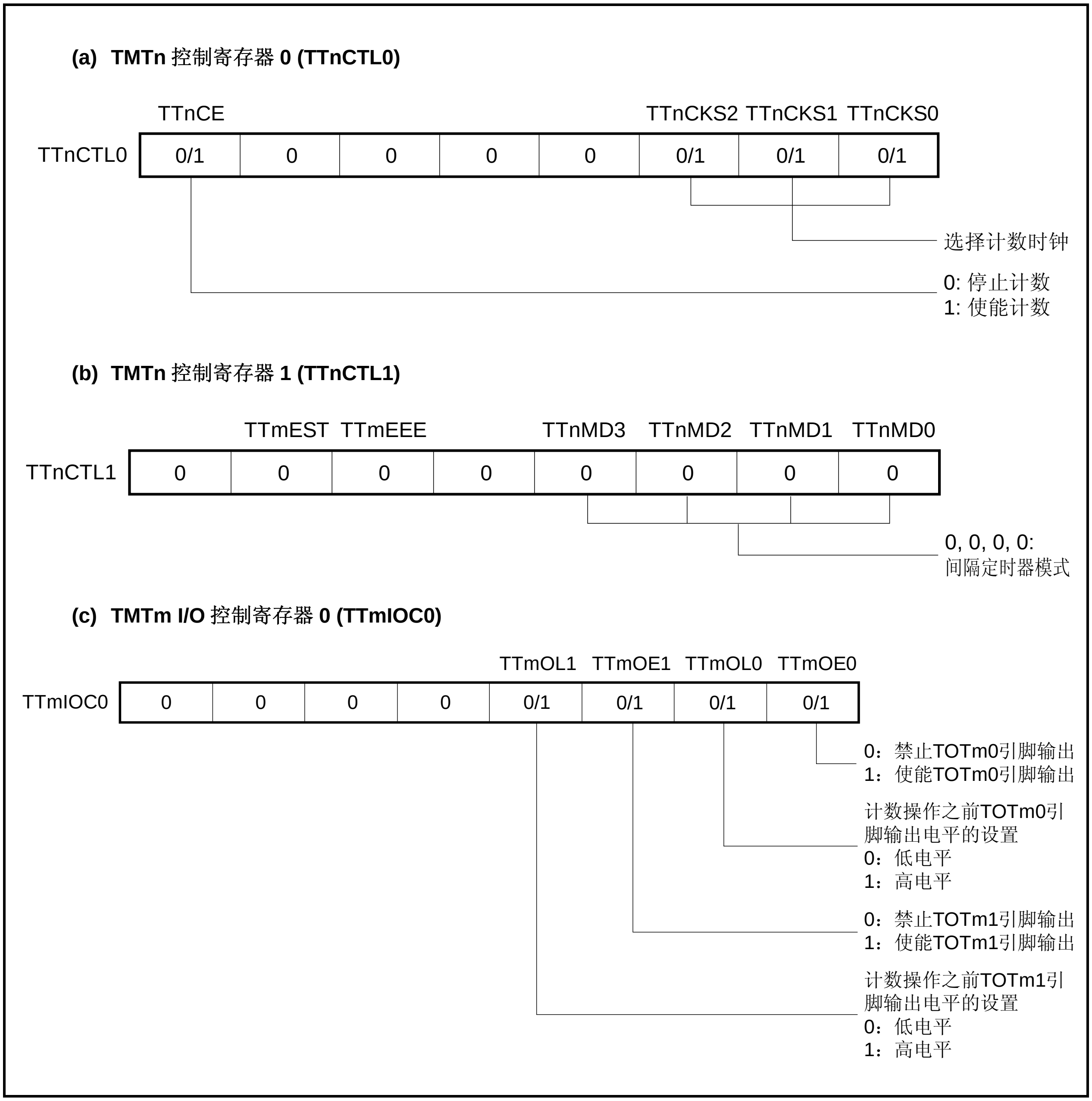


图 8-9. 间隔定时器模式操作的寄存器设置 (2/2)

(d) TMTn 计数器读取缓冲寄存器 (TTnCNT)

通过读取 TTnCNT 寄存器，可以读取 16 计数器的当前值。

（e） TMTn 捕获/比较寄存器 0（TTnCCR0）

如果 TTnCCR0 寄存器的值被设置为 D_0，则时间间隔的计算如下：

时间间隔=（D_0＋1）× 计数时钟周期

（f） TMTn 捕获/比较寄存器 1（TTnCCR1）

间隔定时器模式不使用 TTnCCR1 寄存器。但是，TTnCCR1 寄存器的设置值仍然会被传输给 CCR1 缓冲寄存器。当 16 位计数器的计数值与 CCR1 缓冲寄存器的值一致时，TOTm1 引脚输出被反转，且产生比较一致中断请求信号（INTTTEQCn1）。

将该寄存器的值设置为和 TTnCCR0 寄存器的设置值相等，可以从 TOTm1 引脚输出一个占空比 50%的 PWM 波形。

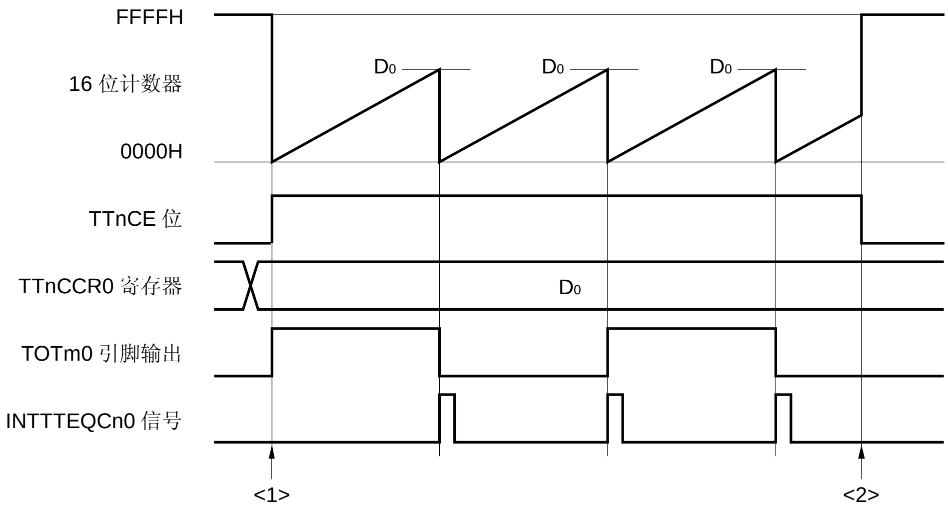
不使用 TTnCCR1 寄存器时，推荐将其值设置为 FFFFH。并且通过相应的中断屏蔽标志（TTnCCIC1.TTnCCMK1）屏蔽该寄存器。

备注 **1.** 间隔定时器模式不使用 TMTm 控制寄存器 2 (TTmCTL2), TMTm I/O 控制寄存器 1 (TTmIOC1), TMTm I/O 控制寄存器 2 (TTmIOC2), TMTm I/O 控制寄存器 3 (TTmIOC3), TMTn 选项寄存器 0 (TTnOPT0), TMTm 选项寄存器 1 (TTmOPT1), TMTm 捕获输入选择寄存器 (TTISLm)和 TMTm 计数器写入寄存器 (TTmTCW)。

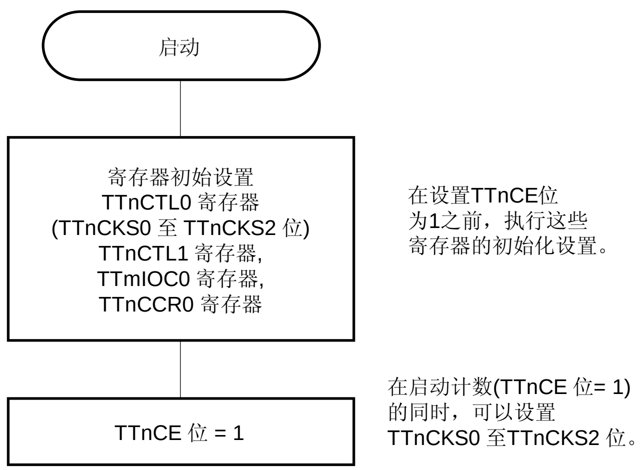
2. V850E/IF3：m = 1, n = 0, 1
V850E/IG3：m = 0, 1, n = 0, 1

(1) 间隔定时器模式操作流程

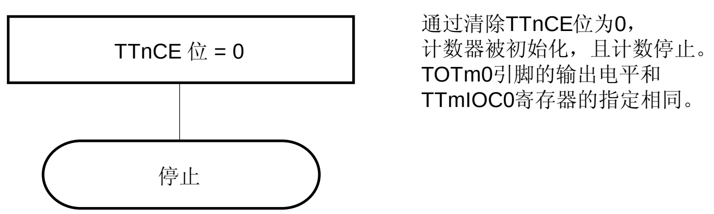
图 **8-10.** 间隔定时器模式的软件处理流程



<1> 计数操作启动流程



<2> 计数操作停止流程



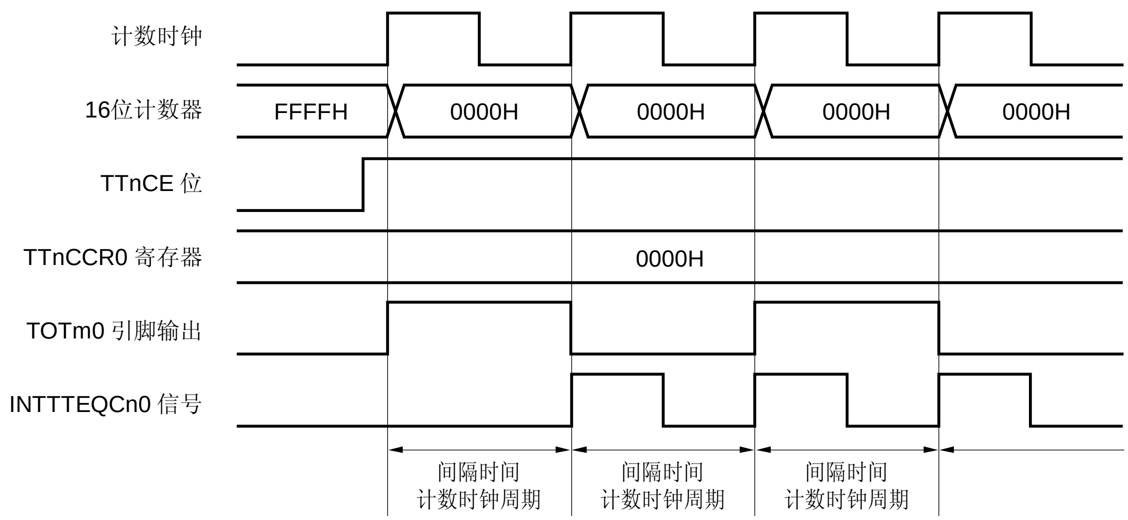
备注　V850E/IF3：n = 0, 1, m = 1
　　　V850E/IG3：n = 0, 1, m = 0, 1

(2) 间隔定时器模式操作时序

(a) TTnCCR0 寄存器被设置为 0000H 时的操作

如果 TTnCCR0 寄存器被清除为 0000H，在每个计数时钟都会产生 INTTTEQCn0 信号，而且，TOTm0 引脚的输出被反转。

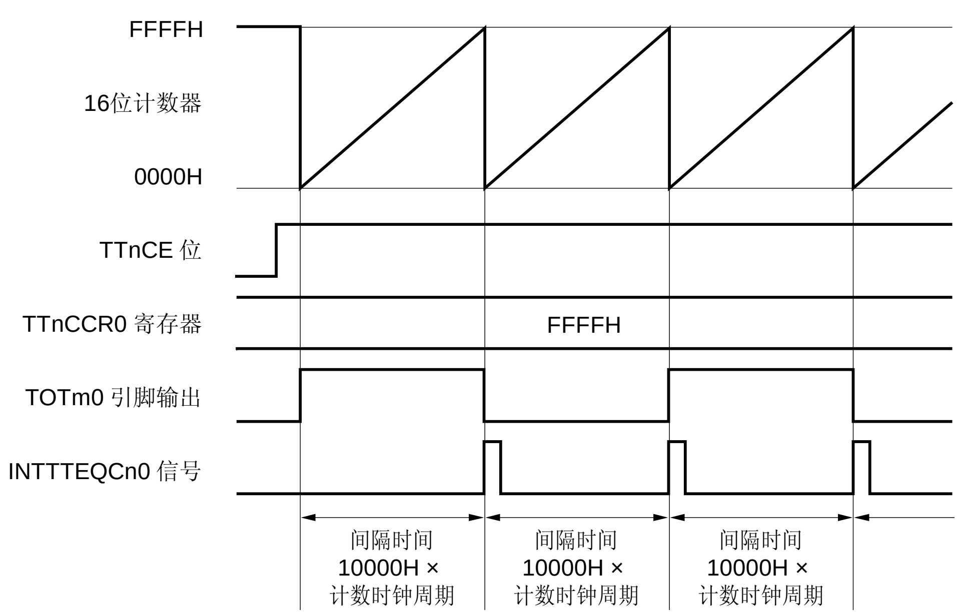
16 位计数器的值总是 0000H。



备注 V850E/IF3：n = 0, 1, m = 1
V850E/IG3：n = 0, 1, m = 0, 1

(b) TTnCCR0 寄存器被设置为 FFFFH 时的操作

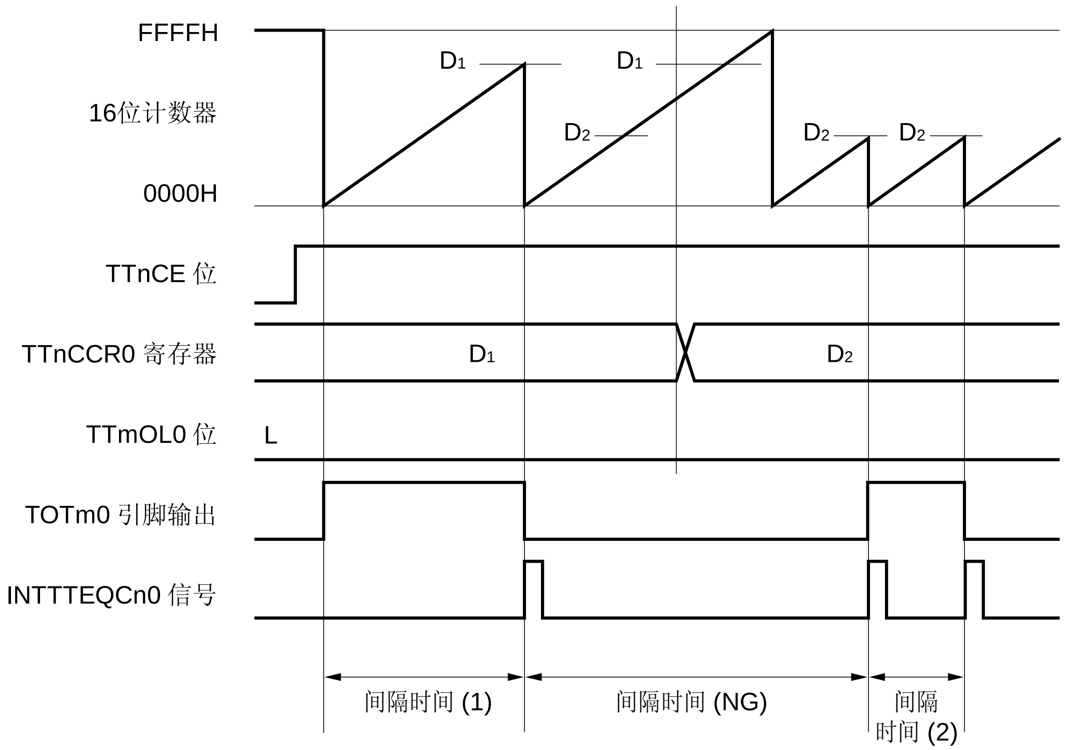
如果 TTnCCR0 寄存器被置为 FFFFH，16 位计数器向上计数到 FFFFH，并在下一个计数时刻被同步清零。产生 INTTTEQCn0 信号，并且 TOTm0 引脚的输出被反转。此时，不产生溢出中断请求信号（INTTTIOVn0OV），而且，溢出标志（TTnOPT0.TTnOVF 位）也不会置 1。



备注 V850E/IF3：n = 0, 1, m = 1
V850E/IG3：n = 0, 1, m = 0, 1

(c) **重新写入 TTnCCR0 寄存器的注意事项**

如果在计数器工作期间，将寄存器 TTnCCR0 的值重写为比当前值小的值，16 位计数器可能会溢出。当可能发生溢出时，应停止计数，然后改变其设置值。



备注　**1.** 间隔时间 (1)：　$(D_1 + 1) \times$ 计数时钟周期

间隔时间 (NG):　$(10000H + D_2 + 1) \times$ 计数时钟周期

间隔时间 (2):　$(D_2 + 1) \times$ 计数时钟周期

2. V850E/IF3：n = 0, 1, m = 1

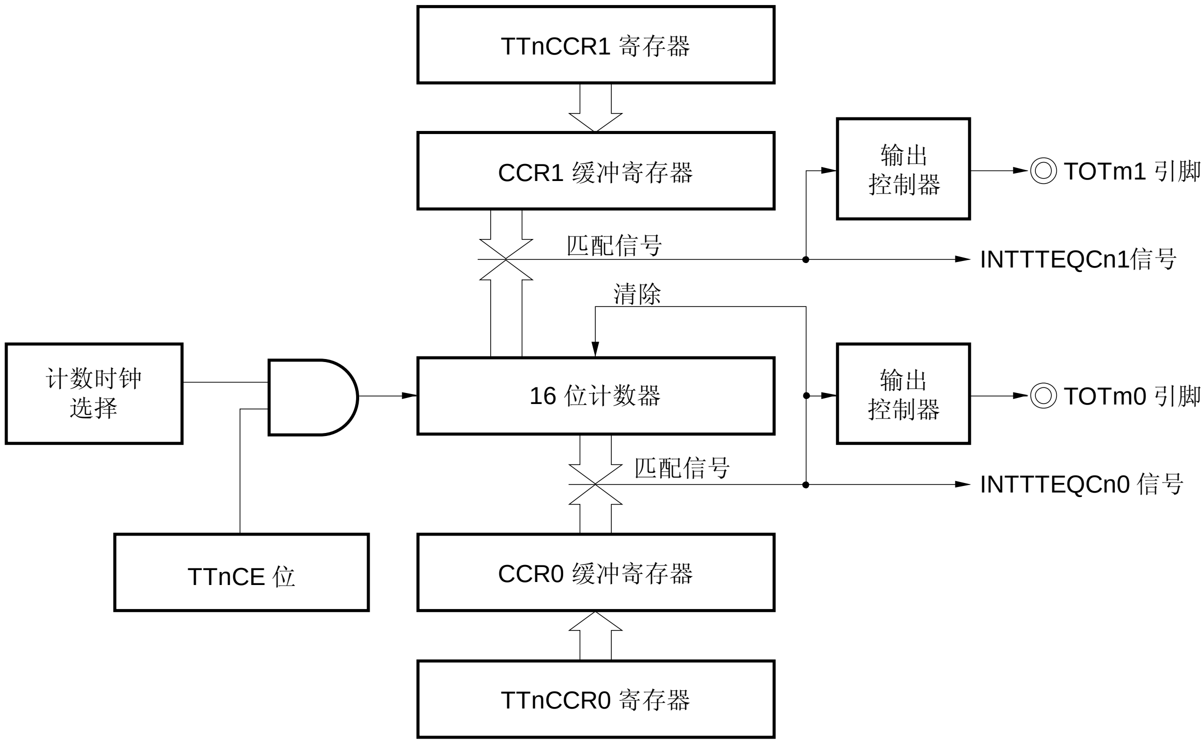
V850E/IG3：n = 0, 1, m = 0, 1

当 16 位计数器计数值大于 D_2 而小于 D_1 时，如果将 TTnCCR0 寄存器的值从 D_1 变为 D_2，则在重写 TTnCCR0 寄存器后，计数值会立即传输给 CCR0 缓冲寄存器。结果，用来比较的 16 位计数器的值是 D_2。但是，由于计数值已经超过了 D_2，所以 16 位计数器就会一直向上累积计数到 FFFFH，溢出，然后从 0000H 开始重新计数。当计数值和 D_2 相符时才产生 INTTTEQCn0 信号，且 TOTm0 引脚的输出被反转。

因此，可能不会在原先预计的间隔时间“（D_1＋1） × 计数时钟周期”或“（D_2＋1） × 计数时钟周期”时产生 INTTTEQCn0 信号，而会以“（10000H＋D_2＋1） × 计数时钟周期”的间隔产生中断请求信号。

(d) TTnCCR1 寄存器的操作

图 **8-11. TTnCCR1** 寄存器的配置



备注　V850E/IF3：n = 0, 1, m = 1
　　　V850E/IG3：n = 0, 1, m = 0, 1

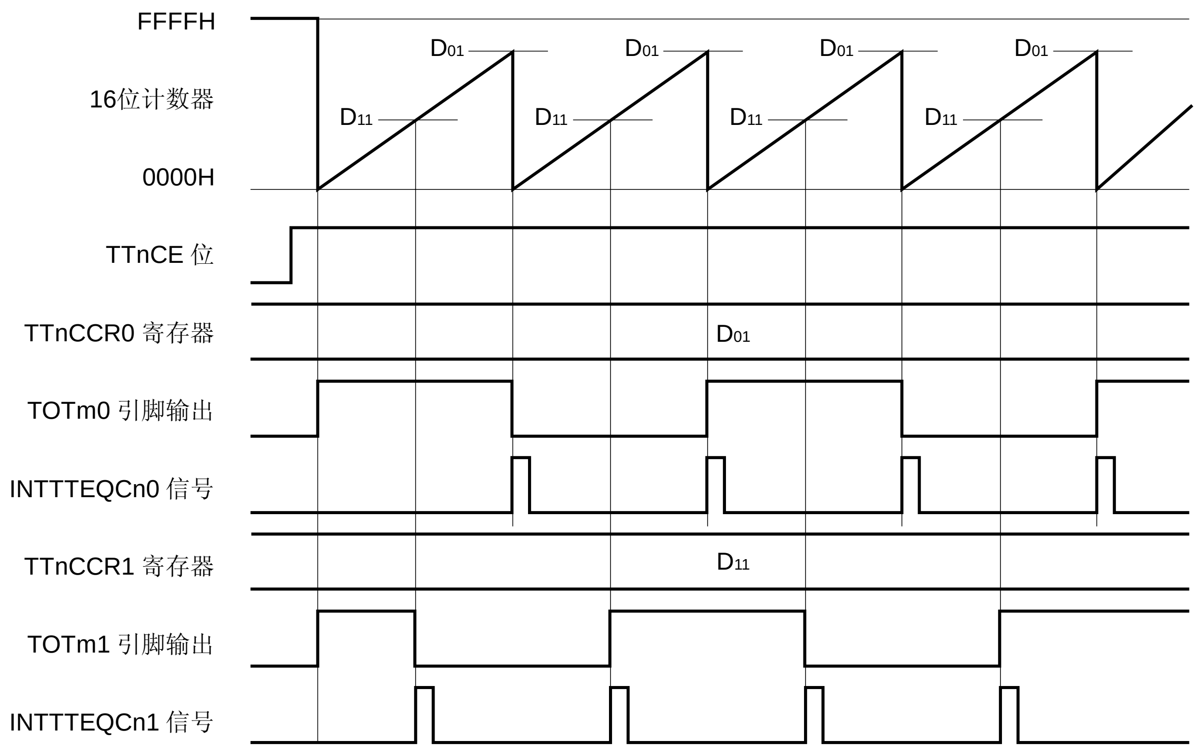
当 TTnCCR1 寄存器的设置值和 TTnCCR0 寄存器的值相同时，INTTTEQCn0 信号与 INTTTEQCn1 信号的产生时序相同，且 TOTm1 引脚输出被反转。换句话说，可以从 TOTm1 引脚输出一个占空比 50%的 PWM 波形。

下面展示了当 TTnCCR1 寄存器的设置值和 TTnCCR0 寄存器的值不等的操作。

如果 TTnCCR1 寄存器的设置值小于 TTnCCR0 寄存器的值，则每个计数周期产生一次 INTTTEQCn1 中断请求信号。同时，TOTm1 引脚的输出被反转。

在输出一个窄宽度脉冲后，TOTm1 引脚输出一个占空比 50%的 PWM 波形。

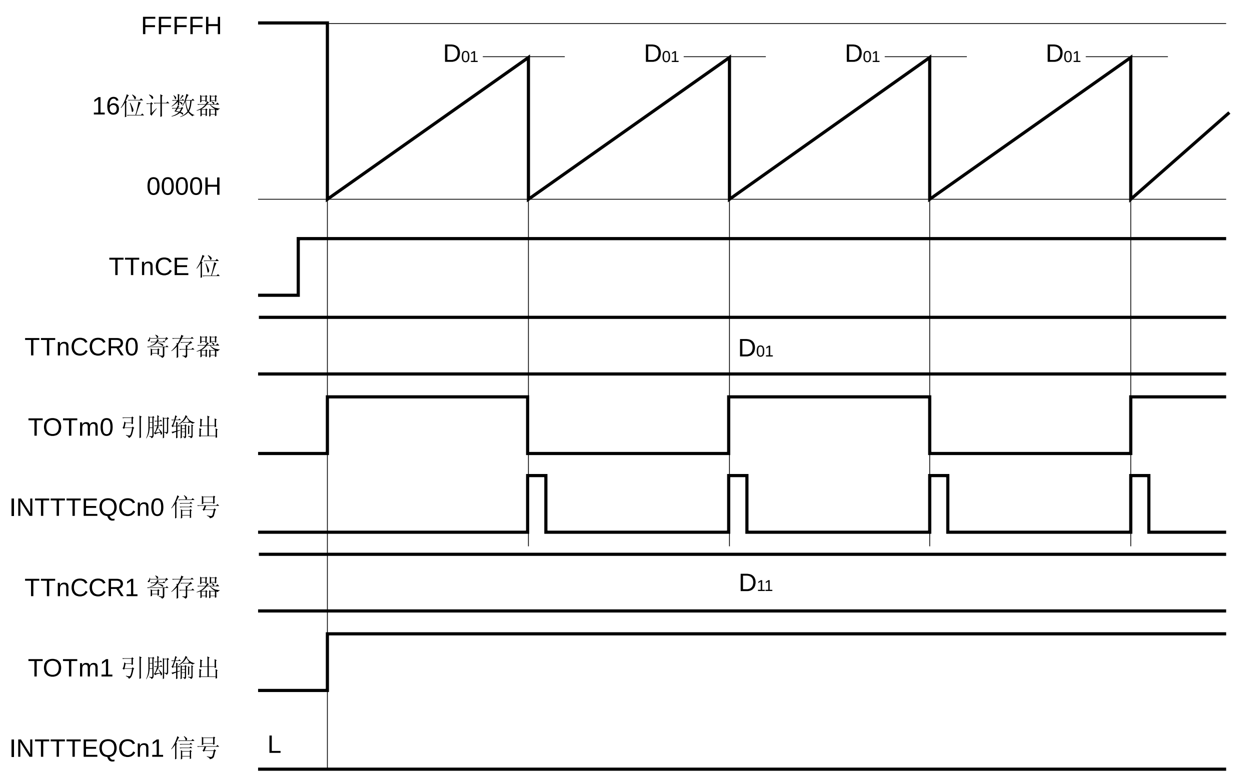
图 8-12. 当 $D_{01} \geq D_{11}$ 时的时序图



备注 V850E/IF3：n = 0, 1, m = 1
V850E/IG3：n = 0, 1, m = 0, 1

如果 TTnCCR1 寄存器的设置值大于 TTnCCR0 寄存器的值，则 16 位计数器的计数值不会和 TTnCCR1 寄存器的值发生一致。因此也就不会产生 INTTTEQCn1 信号，TOTm1 引脚的输出也不会改变。
不使用 TTnCCR1 寄存器时，推荐将其值设置为 FFFFH。

图 8-13. 当 D_{01} < D_{11} 时的时序图



备注　V850E/IF3：n = 0, 1, m = 1
　　　V850E/IG3：n = 0, 1, m = 0, 1

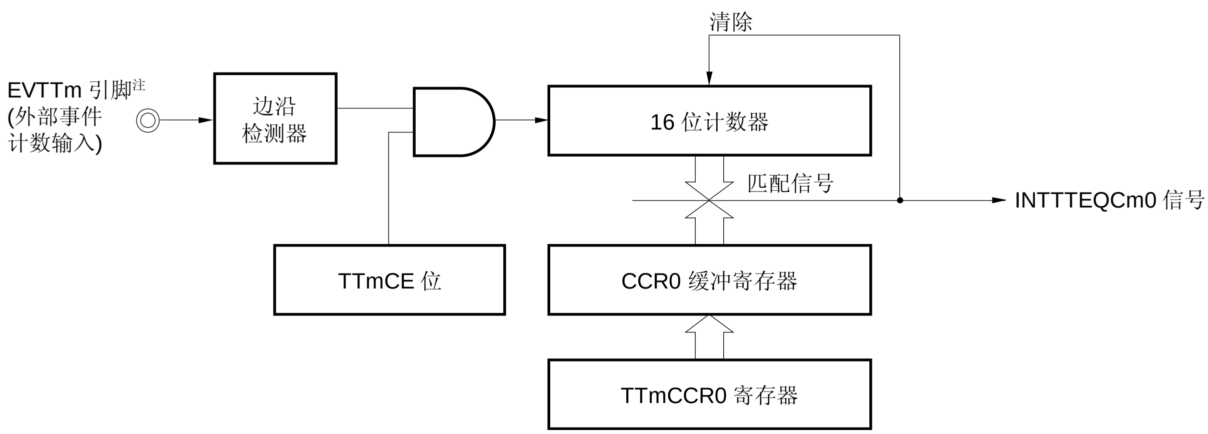
8.6.2 外部事件计数模式 (TTmMD3 至 TTmMD0 位 = 0001)

只有在 TMT0 (仅限 V850E/IG3)和 TMT1 中，该模式有效。

在外部事件计数器模式下，当 TTmCTL0.TTmCE 位被置为 1 时，对外部事件计数输入(EVTTm)的有效边沿进行计数，每次有效边沿的计数数量达到 TTmCCR0 寄存器指定的数量时，产生一个中断请求信号（INTTTEQCm0）。不能使用 TOTm0 和 TOTm1 引脚。

外部事件计数模式不使用 TTmCCR1 寄存器。

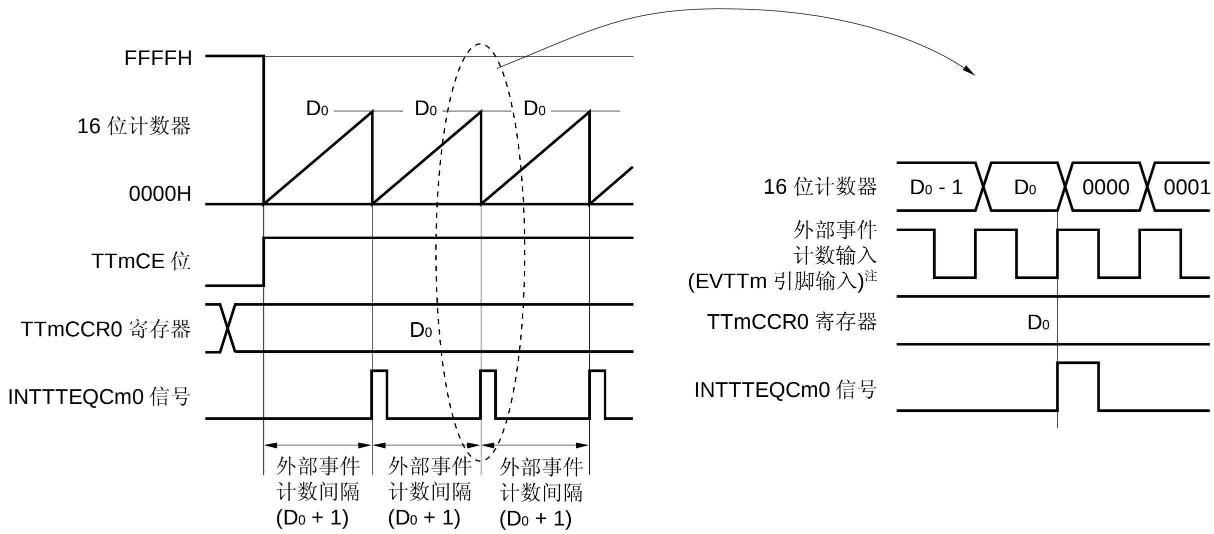
图 8-14. 外部事件计数模式的配置



注　通过 TTmIOC2.TTmEES1 和 TTmIOC2.TTmEES0 位设置

备注　V850E/IF3：m = 1
V850E/IG3：m = 0, 1

图 8-15. 外部事件计数模式的基本时序



备注　**1.** 本图显示了当上升沿被指定作为外部事件计数输入有效边沿时的基本时序。

2. V850E/IF3：m = 1

V850E/IG3：m = 0, 1

当 TTmCE 位被置为 1 时，16 位计数器的值从 FFFFH 被清除为 0000H。计数器对检测到的每一次外部事件计数输入的有效边沿进行计数。此外，TTmCCR0 寄存器的设置值被传输给 CCR0 缓冲寄存器。

当 16 位计数器的计数值和 CCR0 缓冲寄存器的数值相一致时，16 位计数器被清除为 0000H，并产生一个比较一致中断请求信号（INTTTEQCm0）。

每当检测到外部事件计数输入的有效边沿的数量达到（TTmCCR0 寄存器的设置值＋1）时，就会产生 INTTTEQCm0（比较一致中断请求）信号。

图 8-16. 外部事件计数模式操作的寄存器设置 (1/2)

(a) TMTm 控制寄存器 0 (TTmCTL0)

	TTmCE					TTmCKS2	TTmCKS1	TTmCKS0
TTmCTL0	0/1	0	0	0	0	0	0	0

TTmCE：
0: 停止计数
1: 使能计数

(b) TMTm 控制寄存器 1 (TTmCTL1)

		TTmEST	TTmEEE		TTmMD3	TTmMD2	TTmMD1	TTmMD0
TTmCTL1	0	0	0	0	0	0	0	1

TTmMD3～TTmMD0：
0, 0, 0, 1:
外部事件计数模式

(c) TMTm I/O 控制寄存器 2 (TTmIOC2)

					TTmEES1	TTmEES0	TTmETS1	TTmETS0
TTmIOC2	0	0	0	0	0/1	0/1	0	0

TTmEES1, TTmEES0：
选择外部事件计数
输入 (EVTTm 引脚)
的有效边沿

(d) TMTm 计数器读取缓冲寄存器 (TTmCNT)

通过读取 TTmCNT 寄存器，可以获得 16 位计数器的计数值。

(e) TMTm 捕获/比较寄存器 0 (TTmCCR0)

如果 TTmCCR0 寄存器的值被设定为 D_0，当外部事件计数值达到（D_0＋1）时，计数器被清除，并产生一个比较一致中断请求信号（INTTTEQCm0）。

图 8-16. 外部事件计数模式操作的寄存器设置 (2/2)

(f) TMTm 捕获/比较寄存器 1 (TTmCCR1)

外部事件计数模式不使用 TTmCCR1 寄存器。但是，TTmCCR1 寄存器的设置值仍然被传输给 CCR1 缓冲寄存器。当 16 位计数器的计数值和 CCR1 缓冲寄存器的值一致时，会产生一个比较一致中断请求信号（INTTTEQCm1）。

不使用 TTmCCR1 寄存器时，推荐将其值设置为 FFFFH。并且通过中断屏蔽标志（TTmCCIC1.TTmCCMK1）来屏蔽该寄存器。

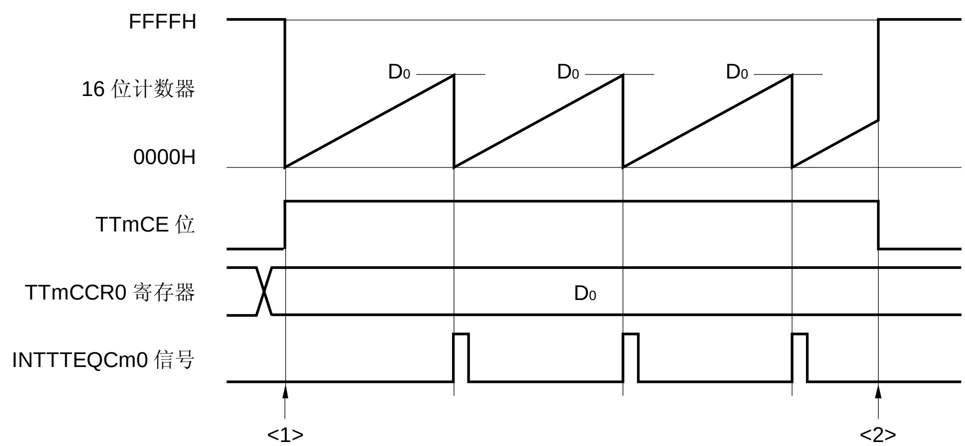
<R> **注意事项　请确保将 TTmIOC0 寄存器设置为 00H。**

备注　**1.** 外部事件计数模式不使用 TMTm 控制寄存器 2 (TTmCTL2), TMTm I/O 控制寄存器 1 (TTmIOC1), TMTm I/O 控制寄存器 3 (TTmIOC3), TMTm 选项寄存器 0 (TTmOPT0), TMTm 选项寄存器 1 (TTmOPT1), TMTm 捕获输入选择寄存器 (TTISLm)和 TMTm 计数器写入寄存器 (TTmTCW)。

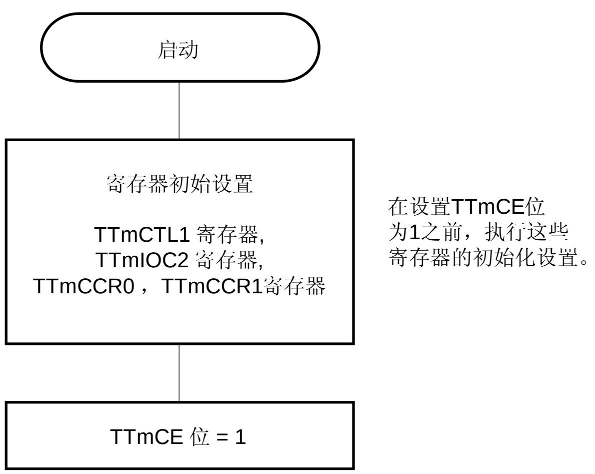
2. V850E/IF3：m = 1
V850E/IG3：m = 0, 1

(1) 外部事件计数模式操作流程

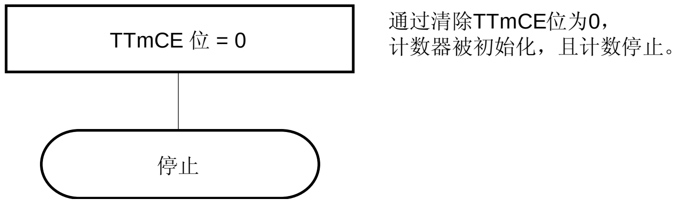
图 **8-17.** 外部事件计数模式的软件处理流程



<1> 计数操作启动流程



<2> 计数操作停止流程



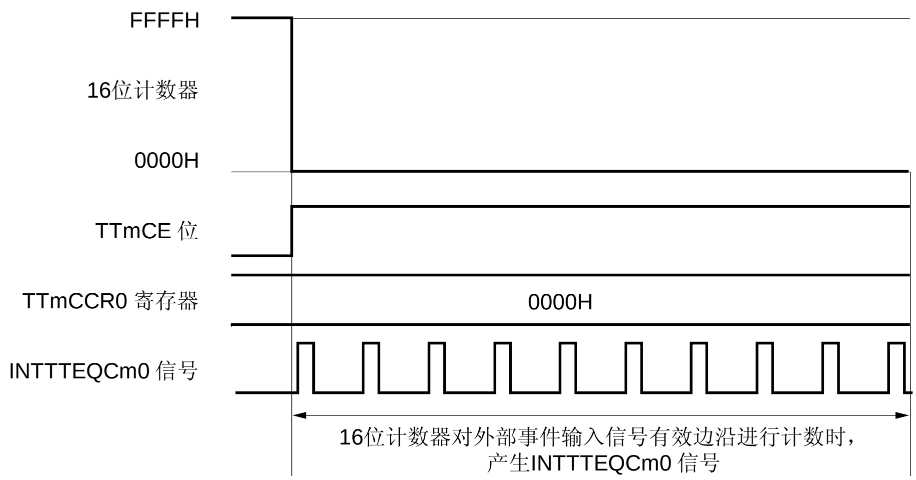
备注　V850E/IF3：m = 1
　　　V850E/IG3：m = 0, 1

(2) 外部事件计数模式的操作时序

<R> 注意事项　在外部事件计数模式下，禁止使用定时器输出**(TOTm0, TOTm1)**。

(a) TTmCCR0 寄存器被设置为 0000H 时的操作

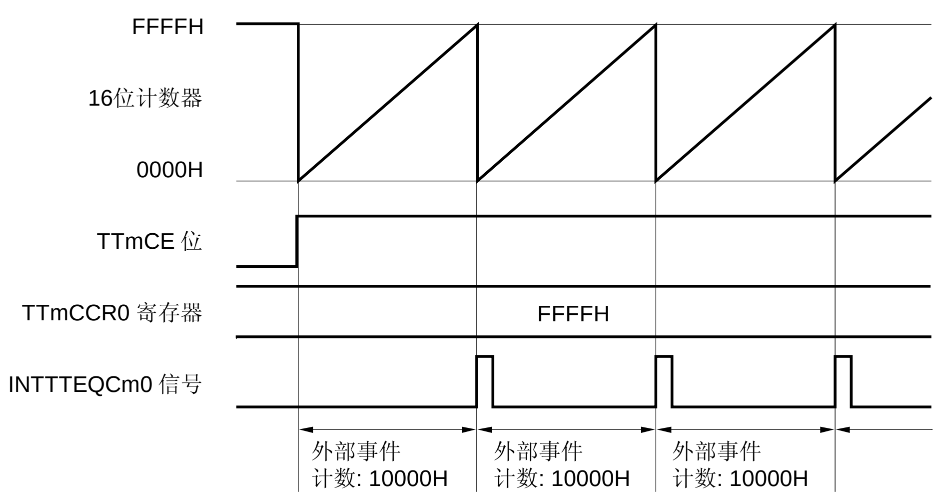
如果 TTmCCR0 寄存器被设置为 0000H，每当检测到的外部事件输入信号有效边沿时，或 16 位计数器的值和 CCR0 缓冲寄存器的值一致时，16 位计数器都会被清除为 0000H，并且产生 INTTTEQCm0 信号。
16 位计数器的值总是 0000H。



备注　V850E/IF3：m = 1
V850E/IG3：m = 0, 1

(b) TTmCCR0 寄存器被设置为 FFFFH 时的操作

如果 TTmCCR0 寄存器被设置为 FFFFH，每次检测到的外部事件输入信号有效边沿时，16 位计数器都计数到 FFFFH。该计数器在下一个向上计数的时刻被同步清为 0000H，并且产生 INTTTEQCm0 信号。此时，TTmOPT0.TTmOVF 位不置位。

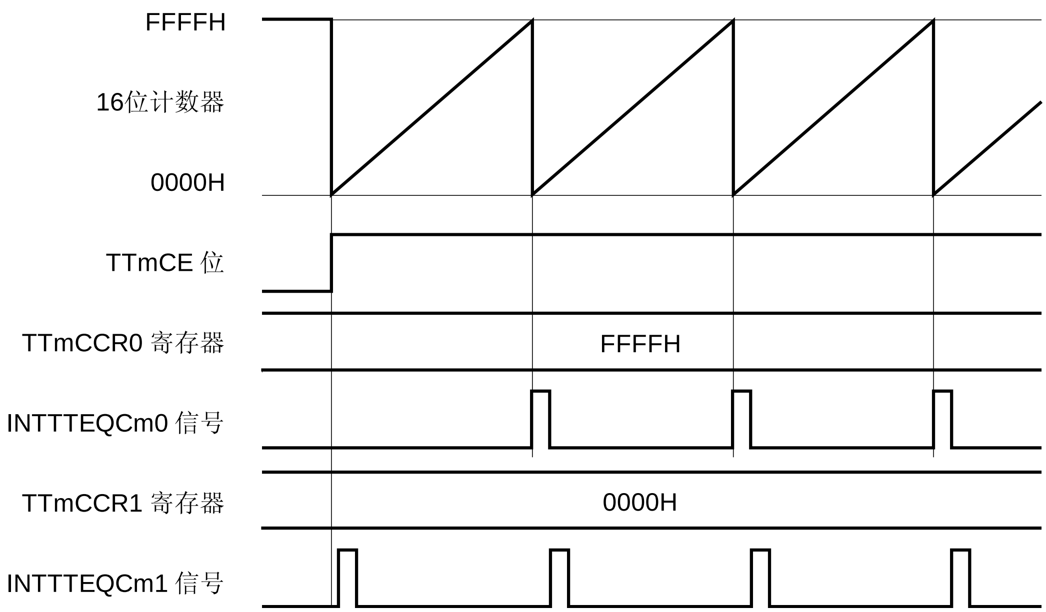


备注　V850E/IF3：m = 1
V850E/IG3：m = 0, 1

(c)　TTmCCR0 寄存器被设置为 FFFFH 且 TTmCCR1 寄存器被设置为 0000H 的操作

当 TTmCCR0 寄存器被设置为 FFFFH 时，每次检测到外部事件计数信号的有效边沿时，16 位计数器向上计数到 FFFFH。该计数器在下一个向上计数的时刻被同步清为 0000H，产生 INTTTEQCm0 信号。此时，TTmOPT0.TTmOVF 位也不会置位。

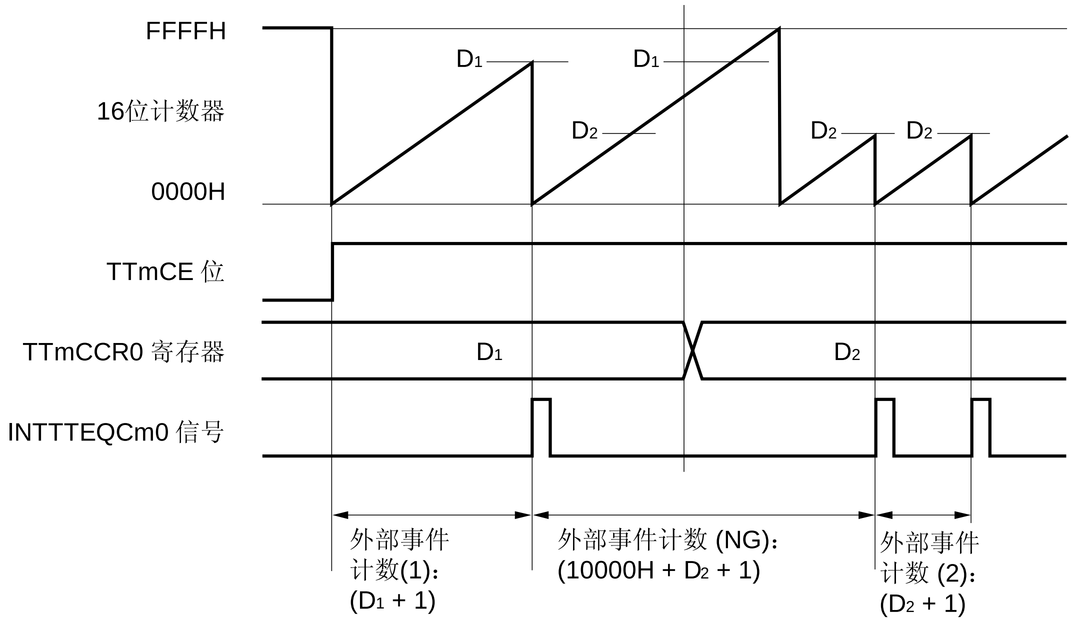
如果 TTmCCR1 寄存器被设置为 0000H，当 16 位计数器被清除为 0000H 时，产生 INTTTEQCm1 信号。



备注　V850E/IF3：m = 1
　　　V850E/IG3：m = 0, 1

(d) 重写 TTmCCR0 寄存器的注意事项

如果在计数器工作期间，将 TTmCCR0 寄存器的值重写为比当前值较小的值，16 位计数器可能会溢出。当可能发生溢出时，停止计数，然后改变设置值。



备注　　V850E/IF3：m = 1
　　　　V850E/IG3：m = 0, 1

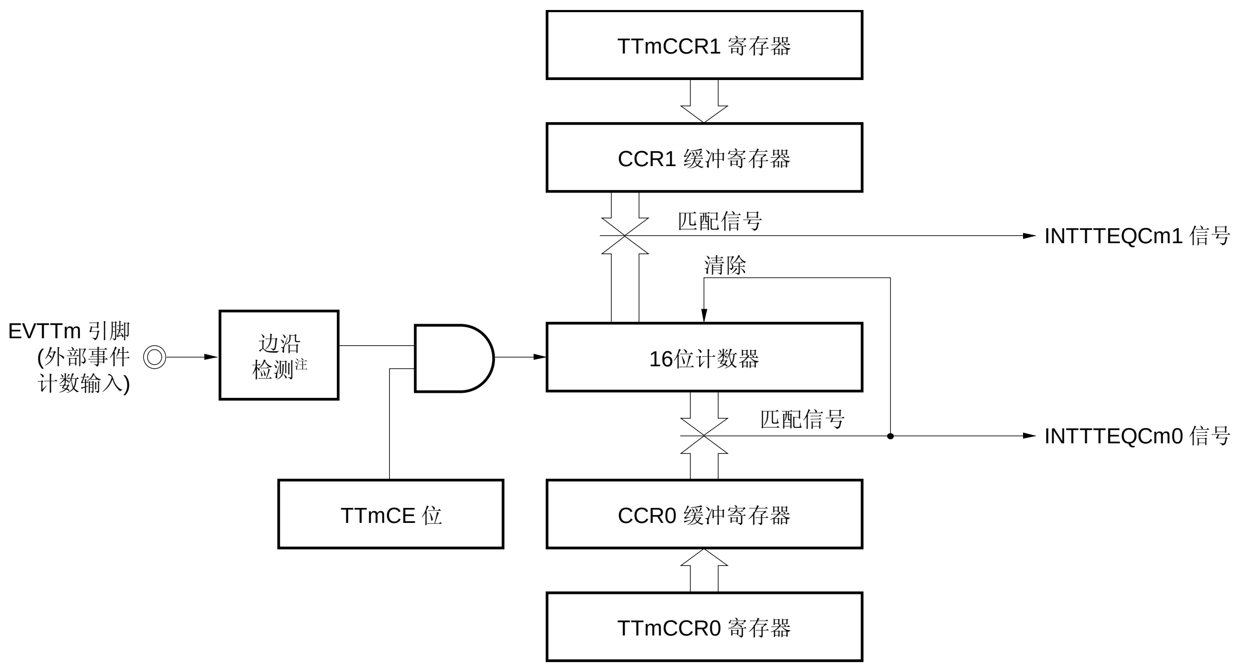
当计数值大于 D_2 而小于 D_1 时，如果将 TTmCCR0 寄存器的值从 D_1 变为 D_2，在重写 TTmCCR0 寄存器后，计数值会立即传输给 CCR0 缓冲寄存器。结果，用来和 16 位计数器进行比较的的值是 D_2。

但是，由于计数值已经超过了 D_2，所以 16 位计数器就会向上累积计数到 FFFFH，溢出，然后从 0000H 开始重新向上计数。当计数值和 D_2 一致时，产生 INTTTEQCm0 信号。

因此，可能不会按照原先预期在"（D_1＋1）次"的有效边沿计数或"（D_2＋1）次"的有效边沿计数时产生 INTTTEQCm0 信号，而会在"（10000H＋D_2＋1）次"的有效边沿计数时产生中断请求信号。

(e) TTmCCR1 寄存器的操作

图 8-18. TTmCCR1 寄存器的配置

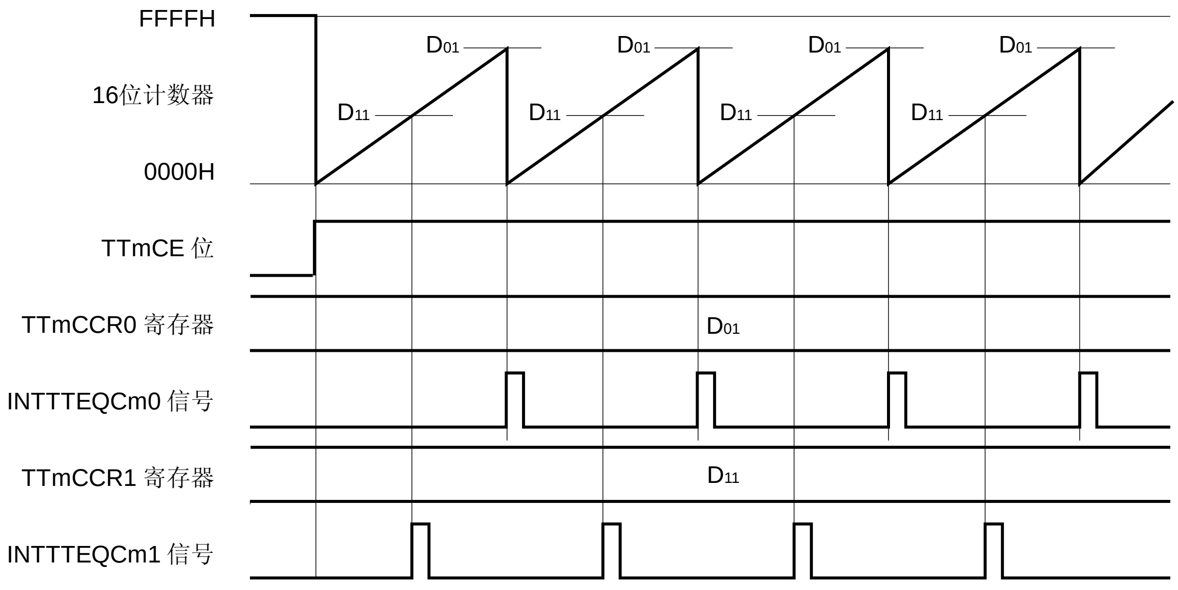


注　　通过 TTmIOC2.TTmEES1 和 TTmIOC2.TTmEES0 位设置。

备注　　V850E/IF3：m = 1
V850E/IG3：m = 0, 1

如果 TTmCCR1 寄存器的设置值小于 TTmCCR0 寄存器的值，则每个计数周期都产生 INTTTEQCm1 中断请求信号。

图 8-19. 当 $D_{01} \geq D_{11}$ 时的时序图

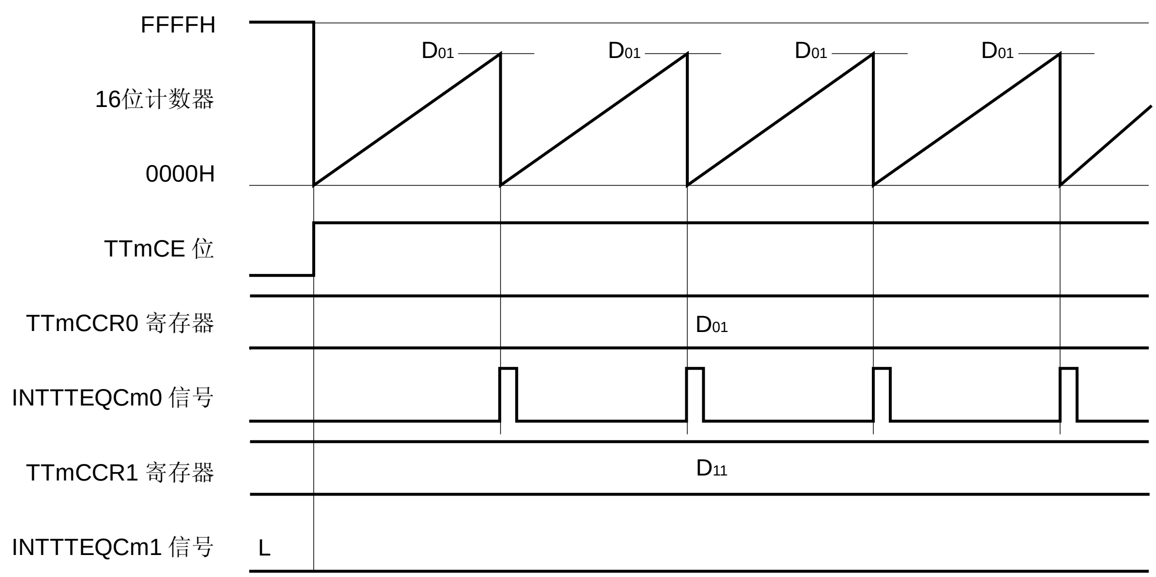


备注　　V850E/IF3：m = 1

　　　　V850E/IG3：m = 0, 1

如果 TTmCCR1 寄存器的设置值大于 TTmCCR0 寄存器的值，则不产生 INTTTEQCm1 中断请求信号，因为 16 位计数器和 TTmCCR1 寄存器的值不会一致。
不使用 TTmCCR1 寄存器时，推荐将其值设置为 FFFFH。

图 8-20. 当 D_{01} < D_{11} 时的时序图



备注　V850E/IF3：m = 1
　　　V850E/IG3：m = 0, 1

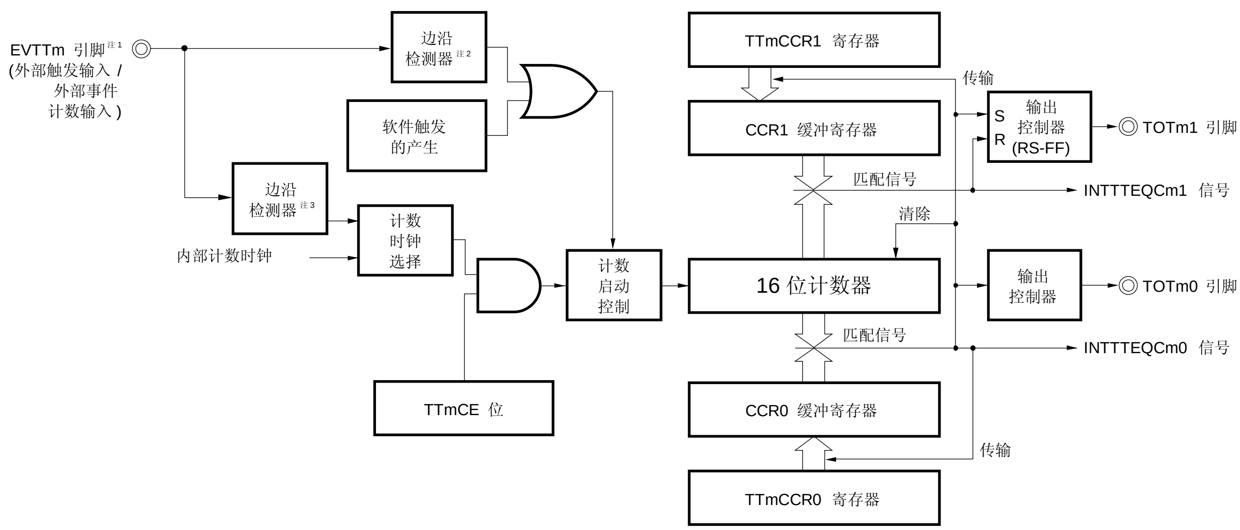
8.6.3　外部触发脉冲输出模式 (TTmMD3 至 TTmMD0 位 = 0010)

只有在 TMT0 (仅限 V850E/IG3)和 TMT1 中，该模式有效。

在外部触发脉冲输出模式下，当 TTmCTL0.TTmCE 位被置为 1 时，16 位定时器/事件计数器 T 等待触发信号，当检测到外部触发输入(EVTTm)的有效边沿后，16 位定时器/事件计数器 T 开始计数，并且从 TOTm1 引脚输出一个 PWM 波形。

不使用外部触发，而是使用软件触发，同样可以得到输出脉冲。当使用软件触发时，可以从 TOTm0 引脚输出一个占空比 50%的 PWM 波形，其半周期等于 TTmCCR0 寄存器的设置值 + 1。

图 8-21. 外部触发脉冲输出模式的配置



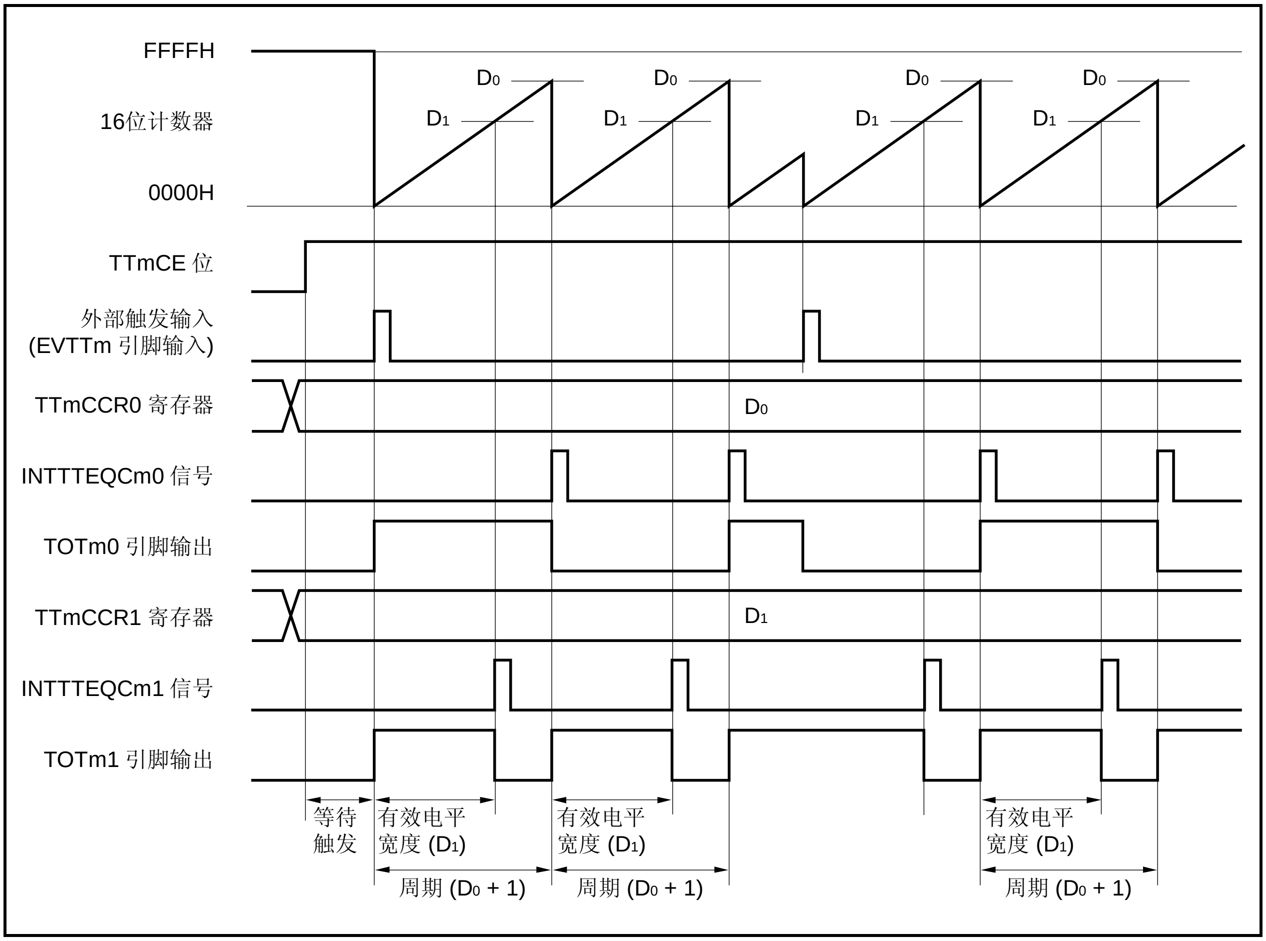
注　**1.** 因为外部触发输入引脚(EVTTm)和外部事件计数输入引脚(EVTTm)共用相同的复用功能引脚，这两个功能不能同时使用。

2. 外部触发输入的边沿检测器。
通过 TTmIOC2.TTmETS1 和 TTmIOC2.TTmETS0 位设置。

3. 外部事件计数输入的边沿检测器。
通过 TTmIOC2.TTmEES1 和 TTmIOC2.TTmEES0 位设置。

备注　V850E/IF3：m = 1
V850E/IG3：m = 0, 1

图 8-22. 外部触发脉冲输出模式的基本时序



当 TTmCE 位被设置为 1 时，16 位定时器/事件计数器 T 等待触发。当触发产生时，16 位计数器从 FFFFH 清零为 0000H，同时开始计数，并从 TOTm1 引脚输出一个 PWM 波形。如果在计数运行中再次产生触发，则计数器清为 0000H 并重新开始计数。（TOTm0 引脚的输出被反转。在产生触发时，不管当前状态如何（高/低），TOTm1 引脚都会输出一个高电平。）

PWM 波形的有效电平宽度、周期、占空比系数的计算如下。

有效电平宽度=（TTmCCR1 寄存器的设置值）× 计数时钟周期

周期=（TTmCCR0 寄存器的设置值＋1）× 计数时钟周期

占空比系数=（TTmCCR1 寄存器的设置值）/（TTmCCR0 寄存器的设置值＋1）

当 16 位计数器在其计数值与 CCR0 缓冲寄存器的值一致之后，进行下一次计数时就会产生比较一致中断请求信号(INTTTEQCm0)，且 16 位计数器清零为 0000H。当 16 位计数器的计数值与 CCR1 缓冲寄存器的值一致时，会生成比较一致中断请求信号(INTTTEQCm1)。

当 16 位计数器的计数值与 CCRa 缓冲寄存器的值一致时，TTmCCRa 寄存器的设置值就被传输到 CCRa 缓冲寄存器，且 16 位计数器清零为 0000H。

外部触发输入(EVTTm)的有效边沿，或将软件触发（TTmCTL1.TTmEST 位）设置为 1，都可以用作触发信号（n = 0，1）。

备注　V850E/IF3：m = 1, a = 0, 1
　　　V850E/IG3：m = 0, 1, a = 0, 1

图 8-23. 外部触发脉冲输出模式的寄存器设置(1/2)

(a) TMTm 控制寄存器 0 (TTmCTL0)

	TTmCE					TTmCKS2	TTmCKS1	TTmCKS0
TTmCTL0	0/1	0	0	0	0	0/1	0/1	0/1

TTmCKS2–TTmCKS0：选择计数时钟[注]

TTmCE：
0: 停止计数
1: 使能计数

注　当 TTmCTL1.TTmEEE 位 = 1 时，该设置无效。

(b) TMTm 控制寄存器 1 (TTmCTL1)

		TTmEST	TTmEEE		TTmMD3	TTmMD2	TTmMD1	TTmMD0
TTmCTL1	0	0/1	0/1	0	0	0	1	0

TTmMD3–TTmMD0：
0, 0, 1, 0:
外部触发脉冲
输出模式

TTmEEE：
0: 运行于由TTmCKS0 至
TTmCKS2 位选择的时钟
1: 由外部事件计数输入
信号计数

TTmEST：
当 写入1 时
产生软件触发

(c) TMTm I/O 控制寄存器 0 (TTmIOC0)

					TTmOL1	TTmOE1	TTmOL0	TTmOE0
TTmIOC0	0	0	0	0	0/1	0/1	0/1	0/1

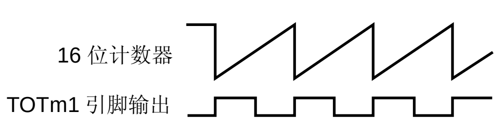
TTmOE0：
0：禁止TOTm0引脚输出
1：使能TOTm0引脚输出

TTmOL0：
在等待外部触发时，
TOTm0引脚输出电平的设置
0：低电平
1：高电平

TTmOE1：
0：禁止TOTm1引脚输出
1：使能TOTm1引脚输出

TTmOL1：
在等待外部触发时，
TOTm1引脚输出电平的设置
0：低电平
1：高电平

• 当TTmOL1 位= 0



• 当TTmOL1 位= 1

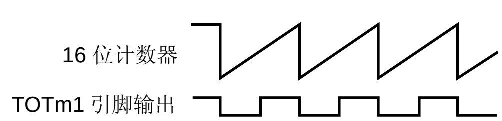


图 **8-23.** 外部触发脉冲输出模式的寄存器设置**(2/2)**

(d) TMTm I/O 控制寄存器 2 (TTmIOC2)

					TTmEES1	TTmEES0	TTmETS1	TTmETS0
TTmIOC2	0	0	0	0	0/1	0/1	0/1	0/1

TTmETS1, TTmETS0：选择外部触发输入 (EVTTm 引脚)[注]的有效边沿

TTmEES1, TTmEES0：选择外部事件计数输入 (EVTTm 引脚)[注]的有效边沿

注　设置未使用的复用外部输入信号的有效边沿选择为“无边沿检测”。

(e) TMTm 计数器读取缓冲寄存器 (TTmCNT)

通过读取 TTmCNT 寄存器，可以读出 16 位计数器的值。

(f) TMTm 捕获/比较寄存器 0 和 1 (TTmCCR0 和 TTmCCR1)

如果将 TTmCCR0 寄存器的值设置为 D_0，将 TTmCCR1 寄存器的值设置为 D_1，则 PWM 波形的周期和有效电平的计算如下：

周期=（D_0＋1）× 计数时钟周期

有效电平宽度= D_1 × 计数时钟周期

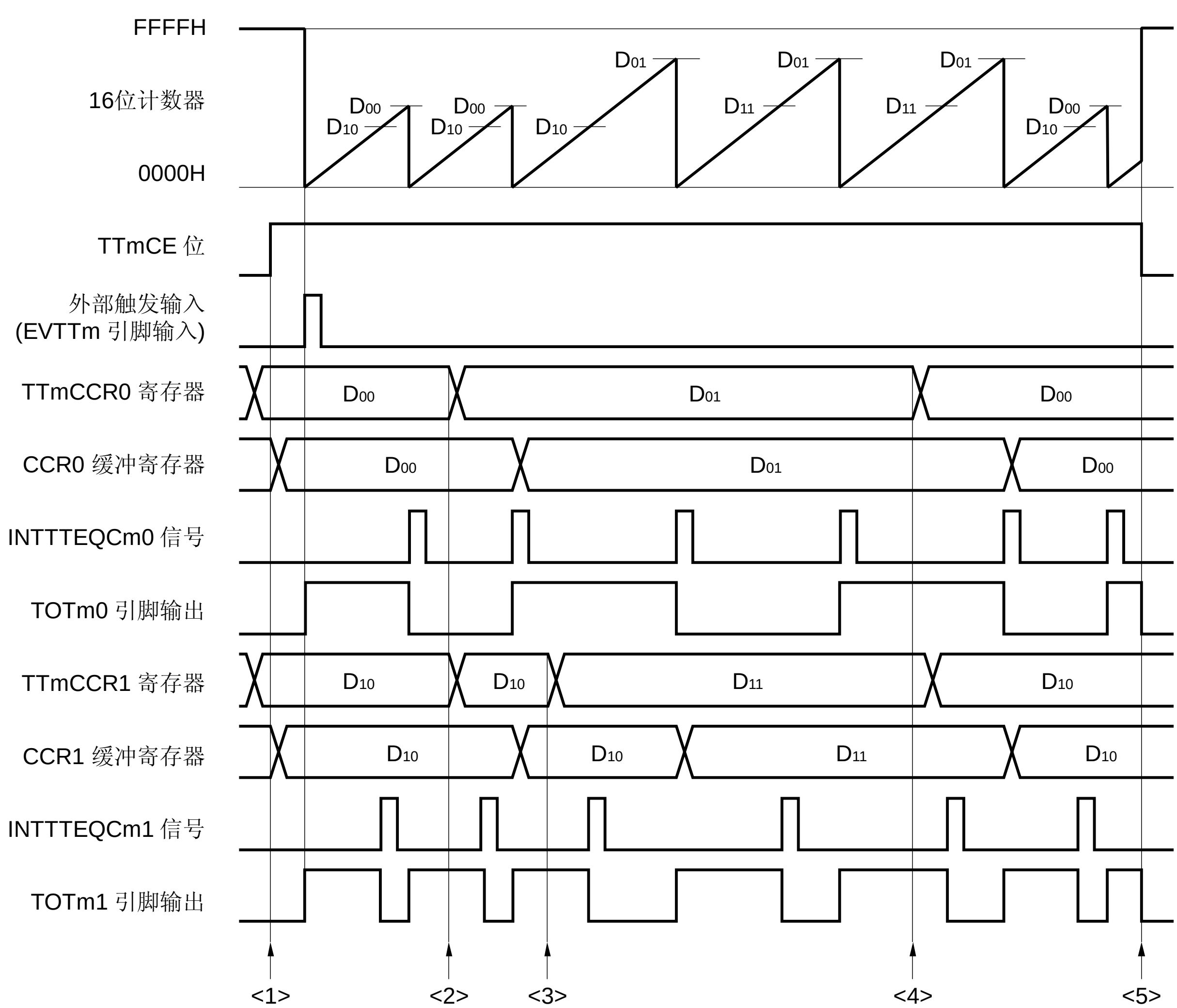
备注　**1.** 外部触发脉冲输出模式不使用 TMTm 控制寄存器 2 (TTmCTL2), TMTm I/O 控制寄存器 1 (TTmIOC1), TMTm I/O 控制寄存器 3 (TTmIOC3), TMTm 选项寄存器 0 (TTmOPT0), TMTm 选项寄存器 1 (TTmOPT1), TMTm 捕获输入选择寄存器 (TTISLm)和 TMTm 计数器写入寄存器。

2. V850E/IF3：m = 1

V850E/IG3：m = 0, 1

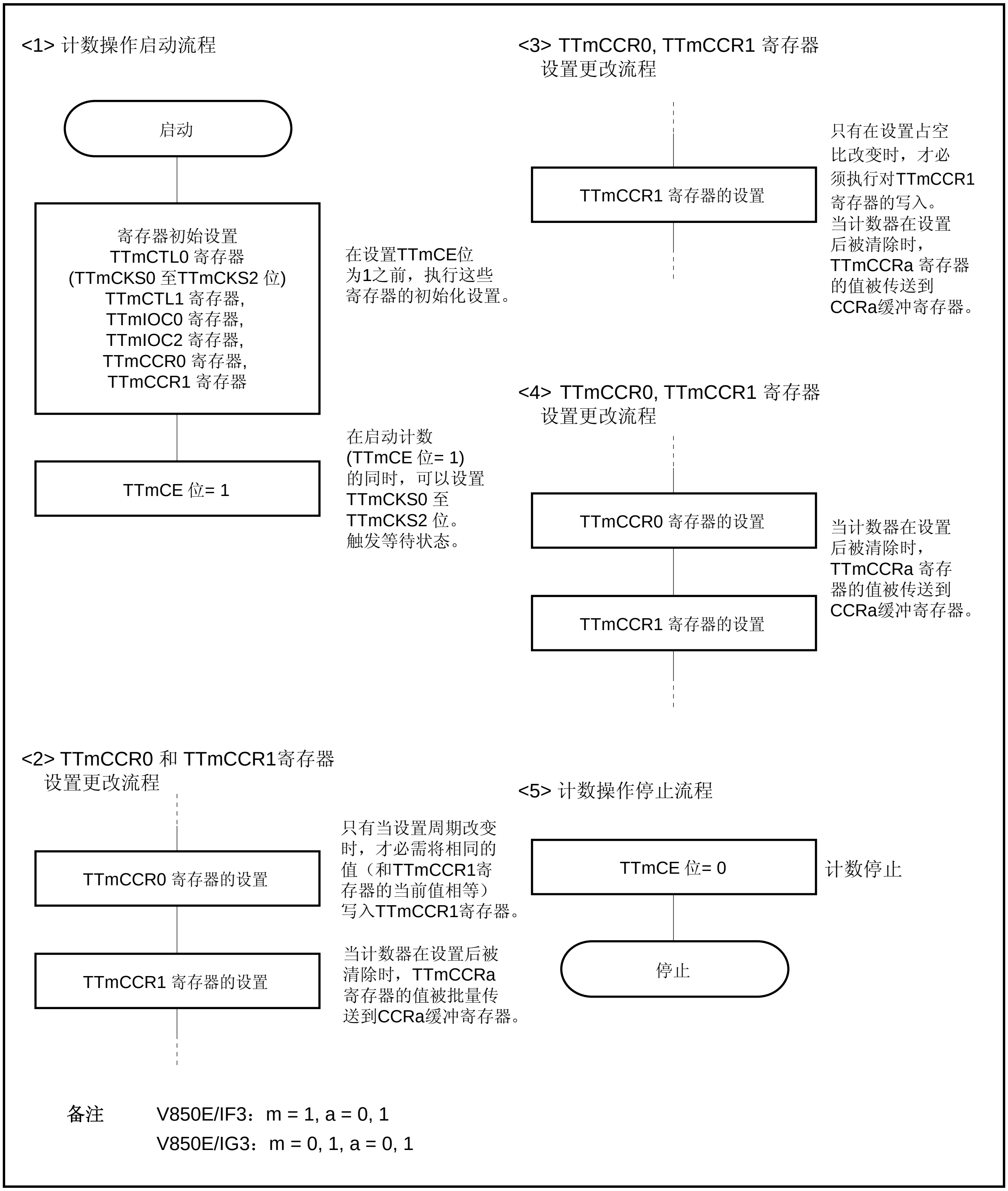
(1) 外部触发脉冲输出模式的操作流程

图 8-24.　外部触发脉冲输出模式的软件处理流程 (1/2)



备注　V850E/IF3：m = 1
　　　V850E/IG3：m = 0, 1

图 8-24.　外部触发脉冲输出模式的软件处理流程 (2/2)

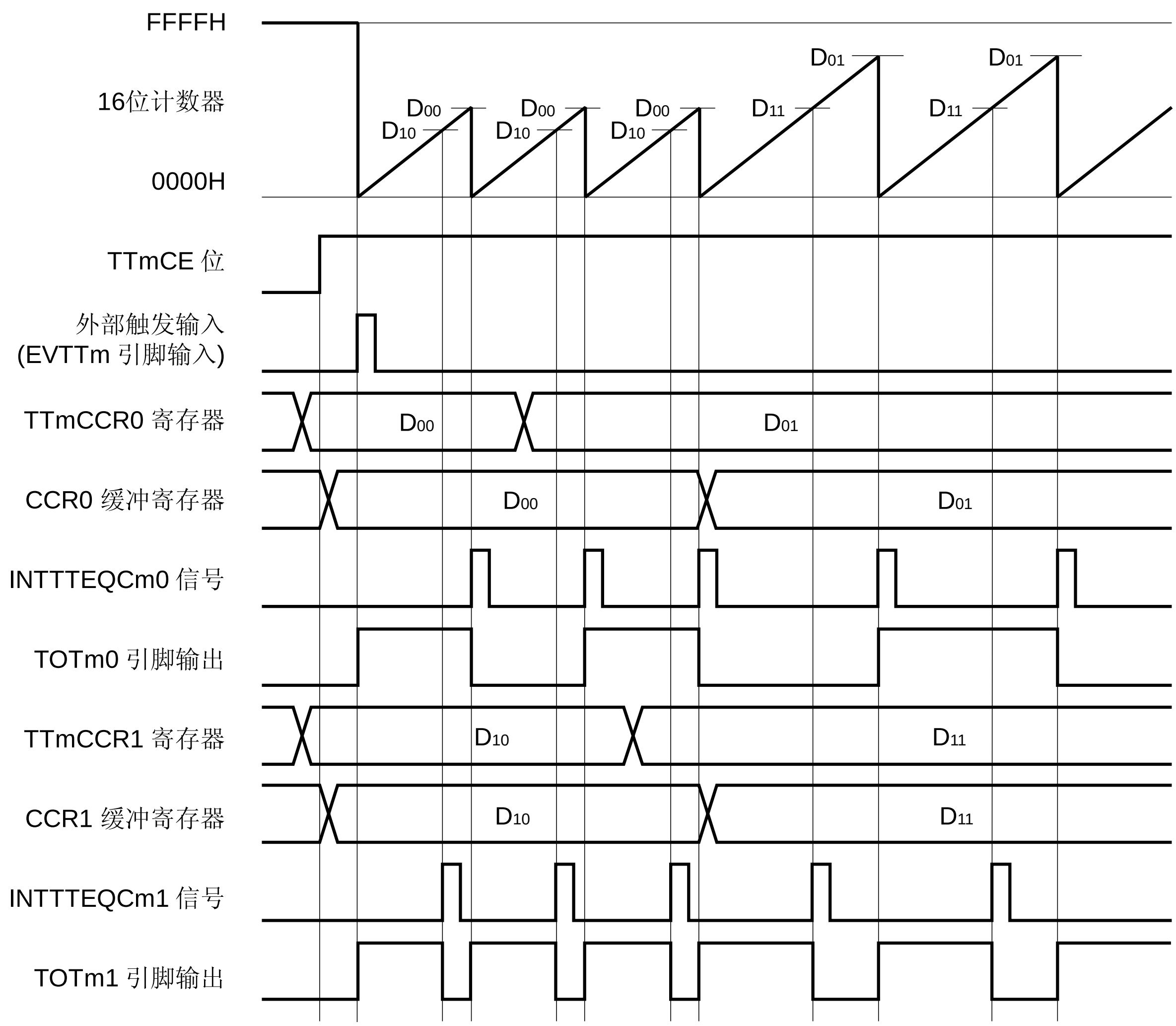


(2) 外部触发脉冲输出模式操作时序

(a) 有关操作中改变脉冲宽度的注意事项

计数器计数过程中，若要改变 PWM 波形，应最后写入 TTmCCR1 寄存器。

当检测到 INTTTEQCm0 信号以后，TTmCCR1 寄存器写入操作之后重写 TTmCCRa 寄存器。



备注　V850E/IF3：m = 1

V850E/IG3：m = 0, 1

为了将数据从 TTmCCRa 寄存器传输到 CCRa 缓冲寄存器，必须写入 TTmCCR1 寄存器。

如果此时 PWM 波形的周期和有效电平宽度都要改变，先将周期设定到 TTmCCR0 寄存器，然后将有效电平宽度设定到 TTmCCR1 寄存器。

如果只改变 PWM 波形的周期，应先把周期设置到 TTmCCR0 寄存器，然后再将相同的值（和 TTmCCR1 寄存器的当前值相等）写入 TTmCCR1 寄存器。

如果只改变 PWM 波形的有效电平宽度（占空比系数），仅需设置 TTmCCR1 寄存器。

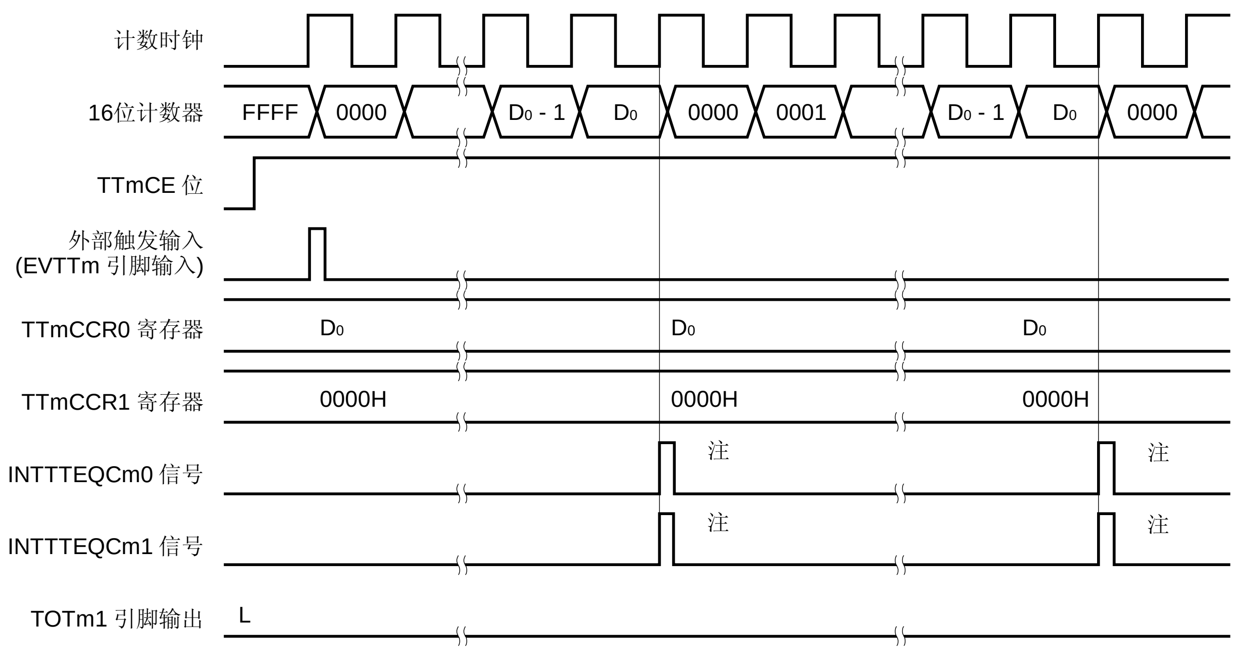
在数据写入 TTmCCR1 寄存器后，在对 16 位计数器清零的同时，写入到 TTmCCRa 寄存器的值会被传输到 CCRa 缓冲寄存器，并作为与 16 位计数器进行比较的值。

在写入 TTmCCR1 寄存器一次之后，要再次写入 TTmCCR0 或 TTmCCR1 寄存器，应该在 INTTTEQCm0 信号产生之后进行。否则，CCRa 缓冲寄存器的值可能不确定，因为把数据从 TTmCCRa 寄存器传输到 CCRa 缓冲寄存器的时序会和写入 TTmCCRa 寄存器的时序发生冲突。

备注　V850E/IF3：m = 1, a = 0, 1
　　　V850E/IG3：m = 0, 1, a = 0, 1

(b) PWM 波形的 0%/100%输出

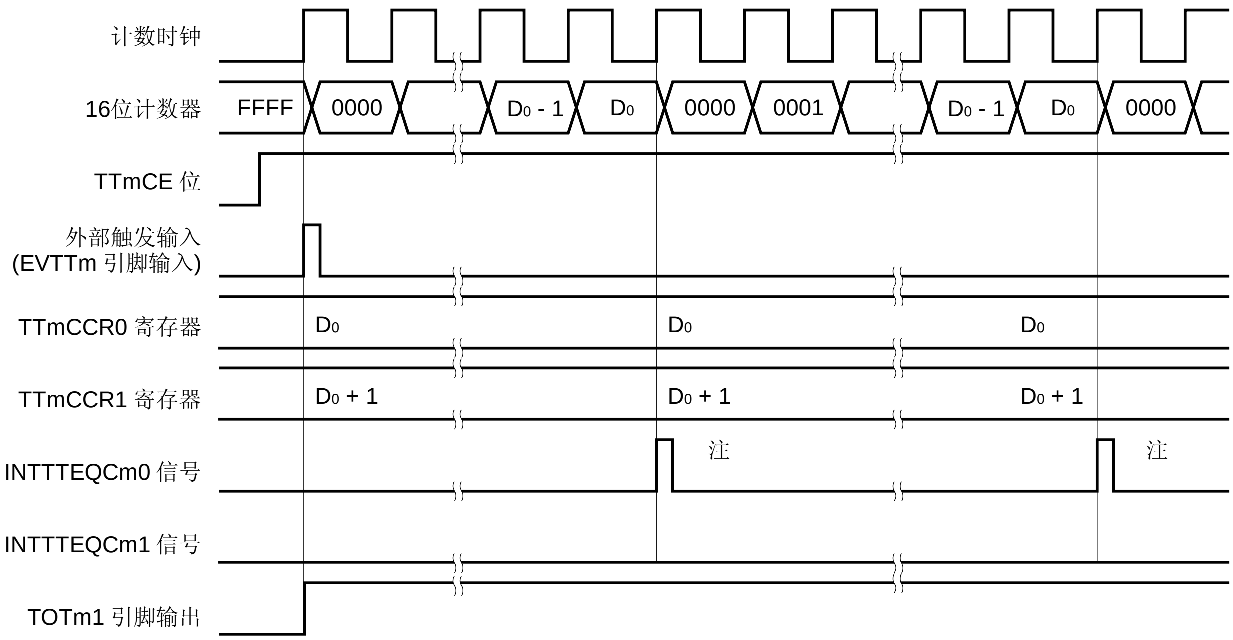
要输出一个 0% 波形，将 TTmCCR1 寄存器设置为 0000H。在 16 位计数器的计数值和 CCR0 缓冲寄存器的值发生一致之后，在下一个时序 16 位计数器被清除为 0000H，且产生 INTTTEQCm0 和 INTTTEQCm1 信号。



注　　时序实际被延迟一个操作时钟(fxx)

备注　　V850E/IF3：m = 1
　　　　V850E/IG3：m = 0, 1

要输出 100%波形，将 TTmCCR1 寄存器的值设置为（TTmCCR0 寄存器的设置值 + 1）。如果 TTmCCR0 寄存器的设置值是 FFFFH，不能生成 100%输出。

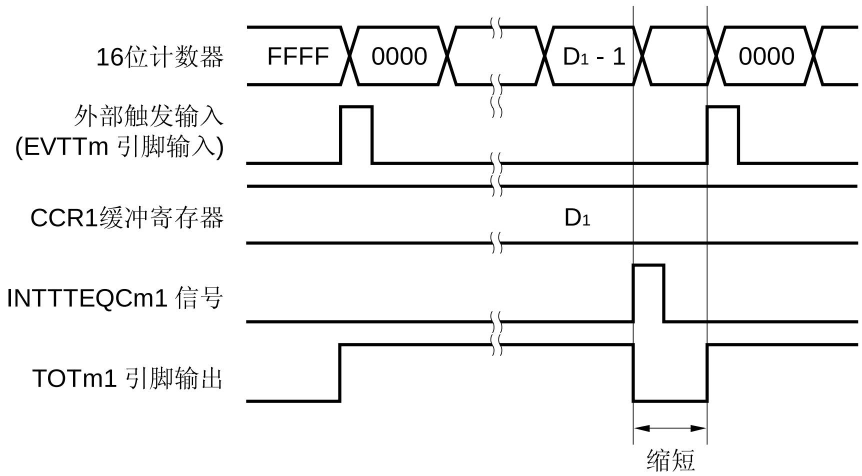


注　　时序实际被延迟一个操作时钟(fxx).

备注　　V850E/IF3：m = 1
V850E/IG3：m = 0, 1

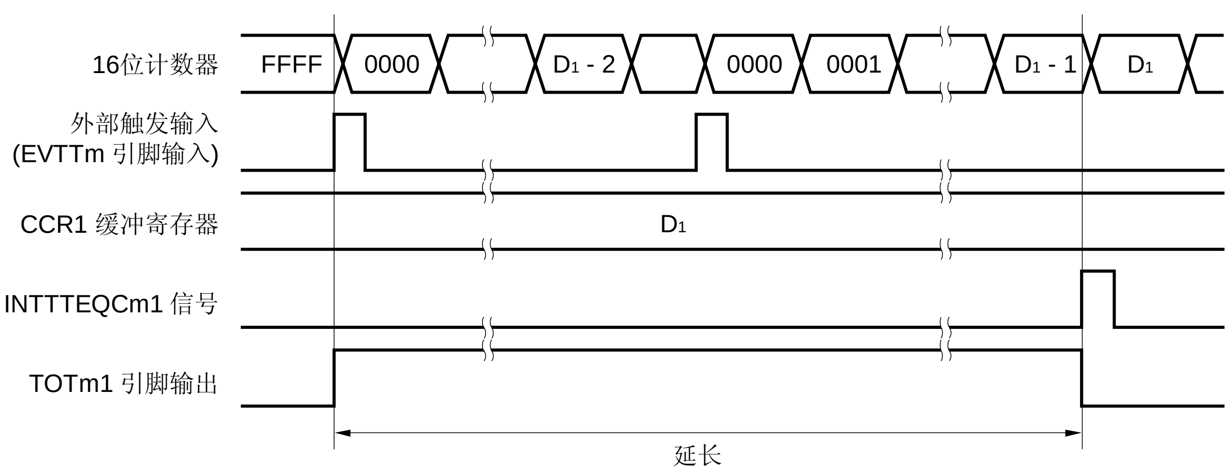
(c) 触发检测与 CCR1 缓冲寄存器一致之间的冲突

如果在 INTTTEQCm1 信号产生之后，立刻检测到触发信号，则 16 位计数器马上清除为 0000H，TOTm1 引脚的输出信号生效，计数器继续计数。结果，PWM 波形的无效时段被缩短。



备注　V850E/IF3：m = 1
　　　V850E/IG3：m = 0, 1

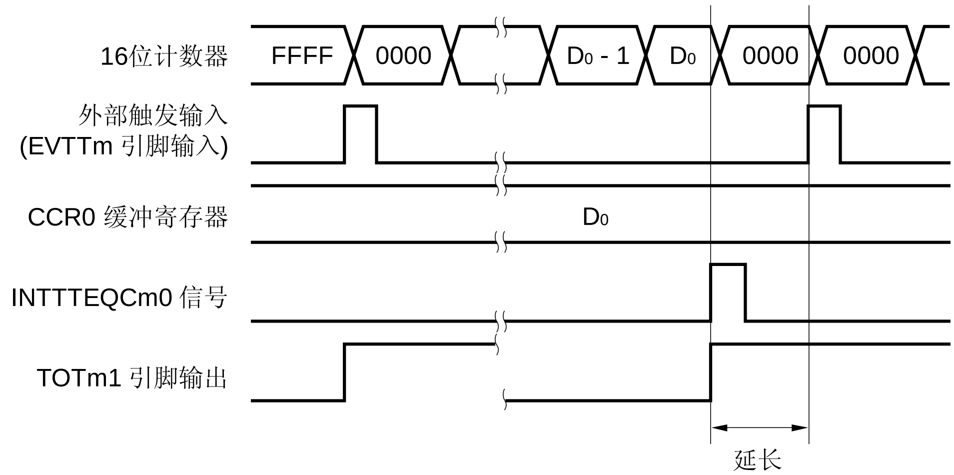
如果紧邻在 INTTTEQCm1 信号产生之前检测到触发信号，则不会产生 INTTTEQCm1 信号，而 16 位计数器被清除为 0000H，计数继续进行。TOTm1 引脚的输出信号仍然保持有效。结果，PWM 波形的有效时段被延长。



备注　V850E/IF3：m = 1
　　　V850E/IG3：m = 0, 1

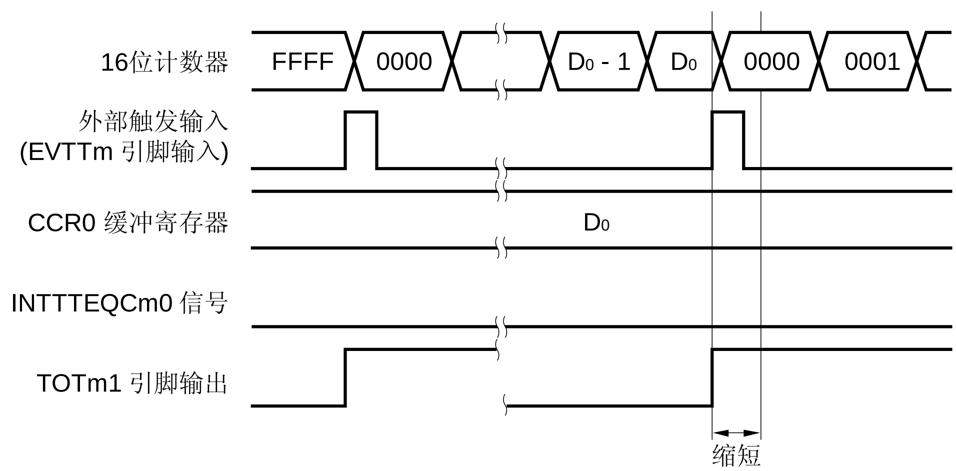
(d) 触发检测与 CCR0 缓冲寄存器一致之间的冲突

如果在 INTTTEQCm0 信号产生后，立刻检测到触发信号，则 16 位计数器再次被清除为 0000H，并继续向上计数。因此，TOTm1 引脚的有效时段被延长，延长长度为 INTTTEQCm0 信号的产生时刻至检测到触发信号时刻的时间间隔。



备注　V850E/IF3：m = 1
　　　V850E/IG3：m = 0, 1

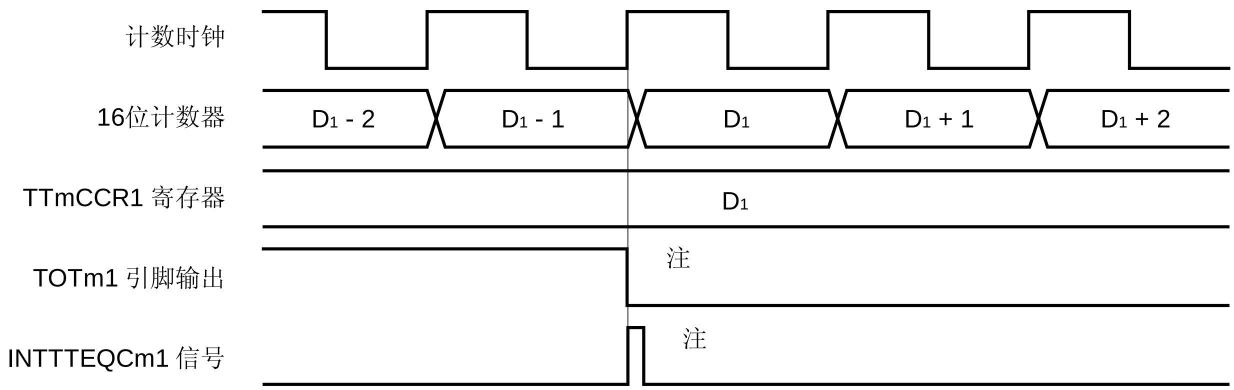
如果紧邻在 INTTTEQCm0 信号产生之前检测到触发信号，则不会产生 INTTTEQCm0 信号。16 位计数器被清除为 0000H，TOTm1 引脚输出有效，计数器继续计数。结果，PWM 波形的无效时段被缩短。



备注　V850E/IF3：m = 1
　　　V850E/IG3：m = 0, 1

(e)　比较一致中断请求信号（INTTTEQCm1）的产生时序

在外部触发脉冲输出模式下，INTTTEQCm1 信号的产生时序与其它情况下 INTTTEQCm1 信号的产生时序不同；当 16 位计数器的计数值与 TTmCCR1 寄存器的值一致时，产生 INTTTEQCm1 中断请求信号。



注　　时序实际被延迟一个操作时钟(f_{XX})

备注　　V850E/IF3：m = 1
V850E/IG3：m = 0, 1

通常在 16 位计数器的计数值与 TTmCCR1 寄存器的值一致之后，在下一次计数时同步发生 INTTTEQCm1 信号。

但是，在外部触发脉冲输出模式下，会提早一个计数时钟周期产生。该时序的变化是为了一致 TOTm1 引脚输出信号变化的时序。

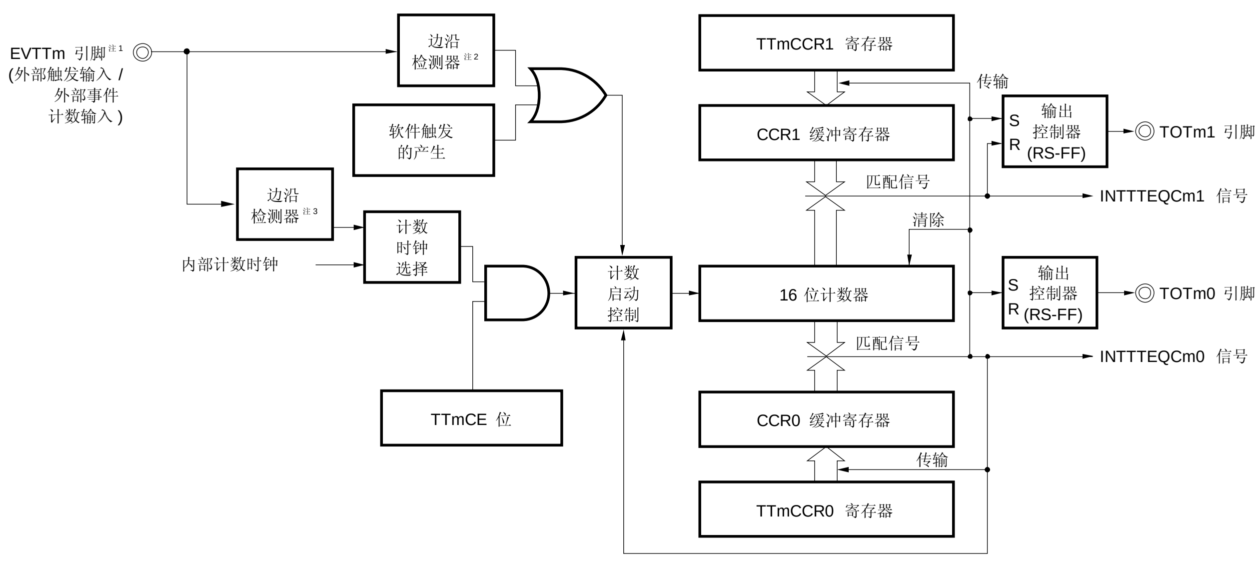
8.6.4　单次触发脉冲输出模式（TTmMD3 至 TTmMD0 位 = 0011）

只有在 TMT0(仅限 V850E/IG3)和 TMT1 中，该模式下有效。

在单次触发脉冲输出模式下，当 TTmCTL0.TTmCE 位被置为 1 时，16 位定时器/事件计数器 T 等待触发。在检测到外部触发输入（EVTTm）的有效边沿时，16 位定时器/事件计数器 T 开始计数，同时从 TOTm1 引脚输出一个单次触发脉冲。

除外部触发输入（EVTTm）外，也可以通过软件触发信号来输出该脉冲。在使用软件触发时，TOTm0 引脚在 16 位计数器计数时输出有效电平，在计数器停止（等待触发）时输出无效电平。

图 8-25. 单次触发脉冲输出模式的配置

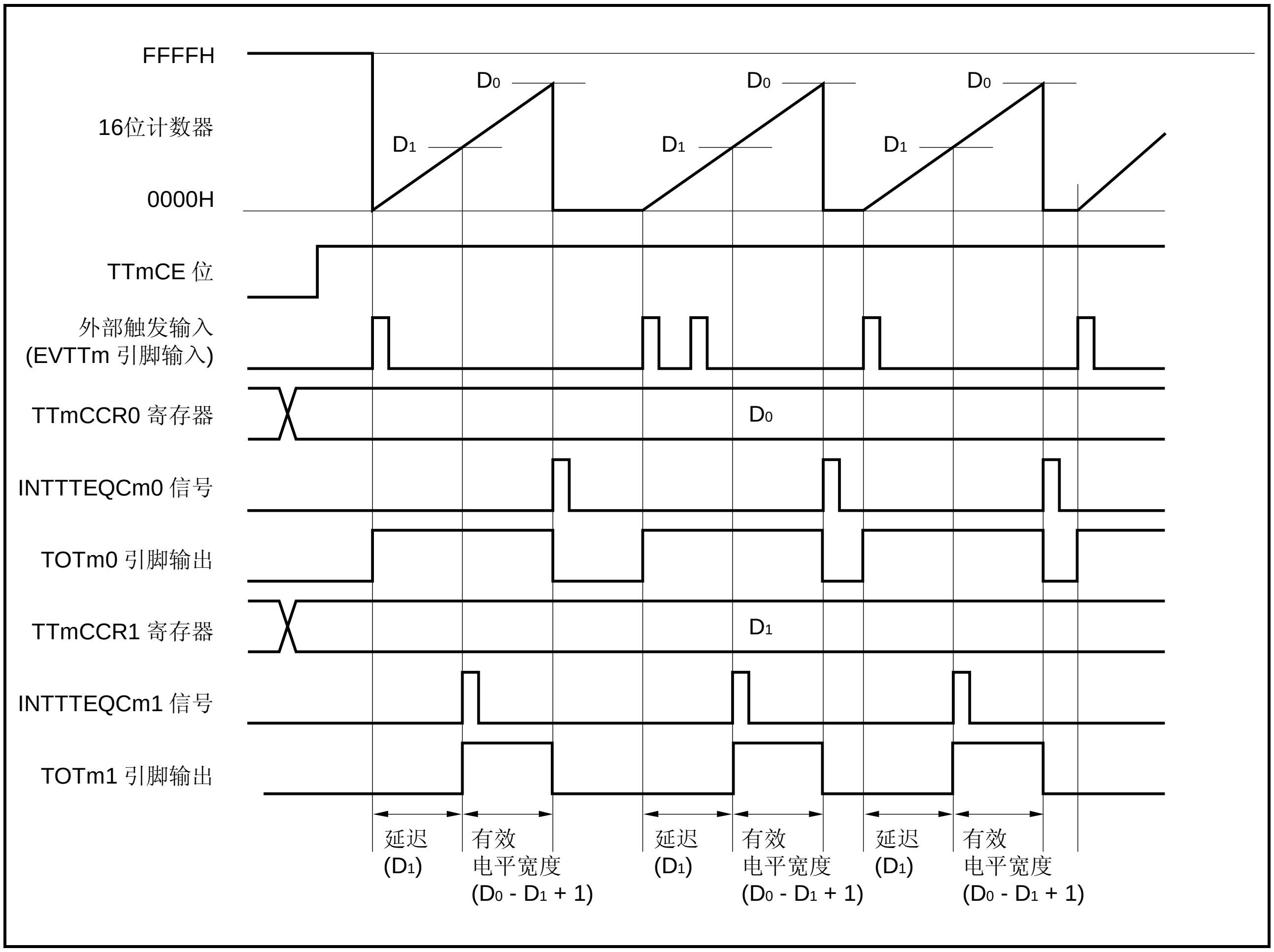


注　**1.** 由于外部触发输入引脚（EVTTm）和外部事件计数输入引脚（EVTTm）是相同的复用功能引脚，两个功能不能同时使用。

2. 外部触发输入的边沿检测器。
通过 TTmIOC2.TTmETS1 和 TTmIOC2.TTmETS0 位设置。

3. 外部事件计数输入的边沿检测器。
通过 TTmIOC2.TTmEES1 和 TTmIOC2.TTmEES0 位设置。

图 8-26. 单次触发脉冲输出模式的基本时序



当 TTmCE 位置为 1 时，16 位计时器/事件计数器 T 等待触发。在触发信号发生时，16 位计数器从 FFFFH 清零为 0000H，开始进行计数，并从 TOTm1 引脚输出一个单次触发脉冲。在输出单次触发脉冲后，16 位计数器被设置为 FFFFH，计数停止，并重新等待触发。当触发信号再次产生，16 位计数器从 0000H 开始计数。如果在单次触发脉冲输出期间再次产生触发信号，则该触发信号被忽略。

单次触发脉冲的输出延迟时间和有效电平宽度可计算如下：

输出延迟时间＝（TTmCCR1 寄存器的设置值）× 计数时钟周期

有效电平宽度＝（TTmCCR0 寄存器的设置值 – TTmCCR1 寄存器的设置值 ＋1）× 计数时钟周期

当 16 位计数器的计数值与 CCR0 缓冲寄存器的值一致之后，产生比较一致中断请求信号（INTTTEQCm0）。当 16 位计数器的计数值与 CCR1 缓冲寄存器的值一致时，会生成比较一致中断请求信号（INTTTEQCm1）。

外部触发输入（EVTTm 引脚）的有效边沿或将软件触发（TTmCTL1.TTmEST 位）信号设置为 1，都可被用作触发信号。

备注　V850E/IF3：m = 1

V850E/IG3：m = 0, 1

图 **8-27.** 单次触发脉冲输出模式时的寄存器设置 （**1/2**）

（a） TMTm 控制寄存器 0 （TTmCTL0）

	TTmCE					TTmCKS2	TTmCKS1	TTmCKS0
TTmCTL0	0/1	0	0	0	0	0/1	0/1	0/1

TTmCKS2 至 TTmCKS0: 选择计数时钟注

TTmCE:
0: 停止计数
1: 使能计数

注　　当 TTmCTL1.TTmEEE 位 = 1 时该设置无效。

（b） TMTm 控制寄存器 1 （TTmCTL1）

		TTmEST	TTmEEE		TTmMD3	TTmMD2	TTmMD1	TTmMD0
TTmCTL1	0	0/1	0/1	0	0	0	1	1

TTmMD3 至 TTmMD0:
0, 0, 1, 1:
单次触发脉冲输出模式

TTmEEE:
0: 计数时钟由 TTmCKS0 至
TTmCKS2 位选择
1: 用外部事件计数
输入信号计数

TTmEST:
当写入1时，
产生软件触发

（c） TMTm I/O 控制寄存器 0 （TTmIOC0）

					TTmOL1	TTmOE1	TTmOL0	TTmOE0
TTmIOC0	0	0	0	0	0/1	0/1	0/1	0/1

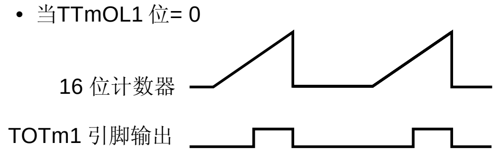
TTmOE0:
0：禁止TOTm0引脚输出
1：使能TOTm0引脚输出

TTmOL0:
在等待外部触发时，
TOTm0引脚输出电平的设置
0：低电平
1：高电平

TTmOE1:
0：禁止TOTm1引脚输出
1：使能TOTm1引脚输出

TTmOL1:
在等待外部触发时，
TOTm1引脚输出电平的设置
0：低电平
1：高电平

• 当TTmOL1 位= 0



• 当TTmOL1 位= 1

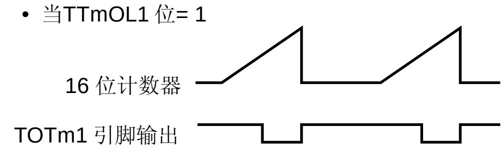


图 **8-27.** 单次触发脉冲输出模式时的寄存器设置 （**2/2**）

（d） TMTm I/O 控制寄存器 2 （TTmIOC2）

					TTmEES1	TTmEES0	TTmETS1	TTmETS0
TTmIOC2	0	0	0	0	0/1	0/1	0/1	0/1

TTmETS1, TTmETS0：选择外部触发输入 (EVTTm 引脚)[注] 的有效边沿

TTmEES1, TTmEES0：选择外部事件计数输入 (EVTTm 引脚)[注] 的有效边沿

注　　设置未使用的复用外部输入信号的有效边沿选择为“无边沿检测”。

（e） TMTm 计数器读取缓冲寄存器 （TTmCNT）

通过读取 TTmCNT 寄存器，可以获得 16 位计数器的值。

（f） TMTm 捕获/比较寄存器 0 和 1（TTmCCR0 和 TTmCCR1）

若把 TTmCCR0 寄存器的值设置为 D_0，TTmCCR1 寄存器的值设置为 D_1，则单次触发脉冲的有效电平宽度和输出延迟时间的计算如下。

有效电平宽度＝（$D_0 - D_1 + 1$）× 计数时钟周期

输出延迟时间＝D_1 × 计数时钟周期

<R> **注意事项　如果 TTmCCR1 寄存器的设置值大于 TTmCCR0 寄存器的值，即使在单次触发脉冲输出模式下，单次触发脉冲也不输出。**

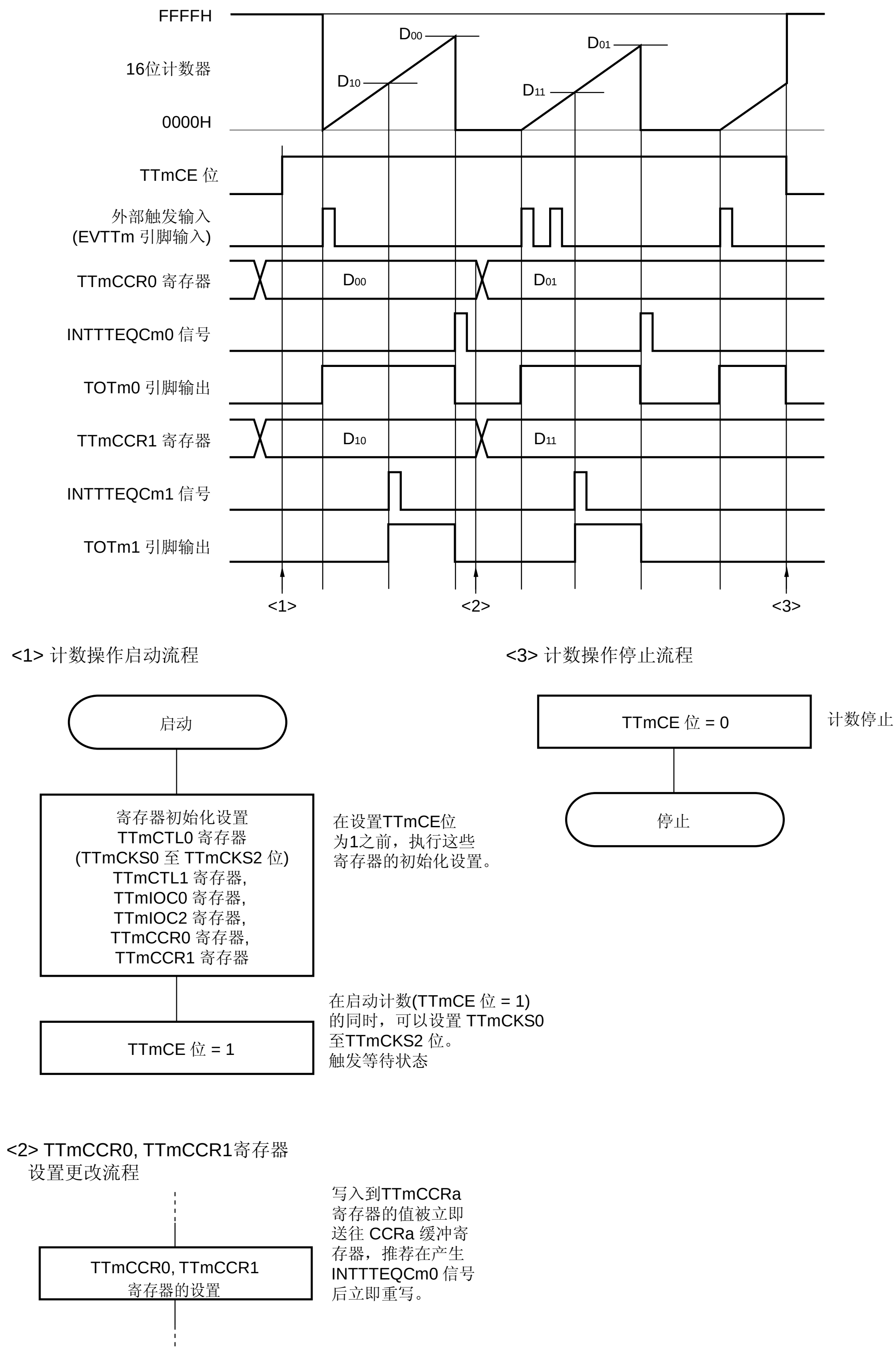
备注

1. 单次触发脉冲输出模式不使用 TMTm 控制寄存器 2（TTmCTL2）、TMTm I/O 控制寄存器 1（TTmIOC1）、TMTm I/O 控制寄存器 3（TTmIOC3）、TMTm 选项寄存器 0（TTmOPT0）、TMTm 选项寄存器 1（TTmOPT1）和 TMTm 计数器写入寄存器（TTmTCW）。

2. V850E/IF3：m = 1
V850E/IG3：m = 0, 1

（1）单次触发脉冲输出模式下的操作流程

图 8-28. 单次触发脉冲输出模式的软件处理流程



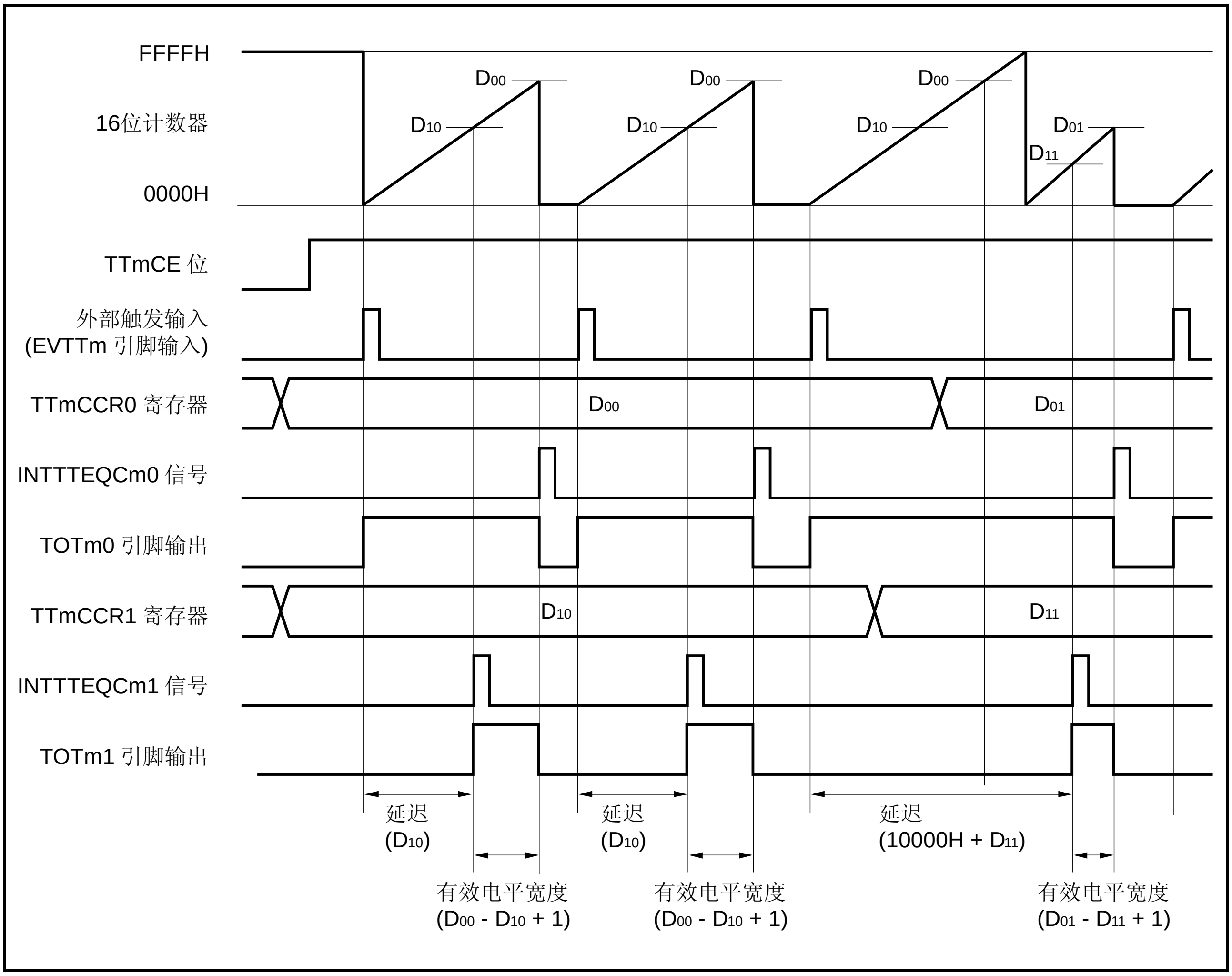
备注　　V850E/IF3：m = 1, a = 0, 1

　　　　V850E/IG3：m = 0, 1, a = 0, 1

（2）单次触发脉冲输出模式时的操作时序

（a）重写 TTmCCRa 寄存器的注意事项

如果在计数器工作期间，将 TTmCCRa 寄存器的值重写为比当前值较小的值，则 16 位计数器可能会溢出。如果可能发生溢出，应停止计数，然后改变其设置值。



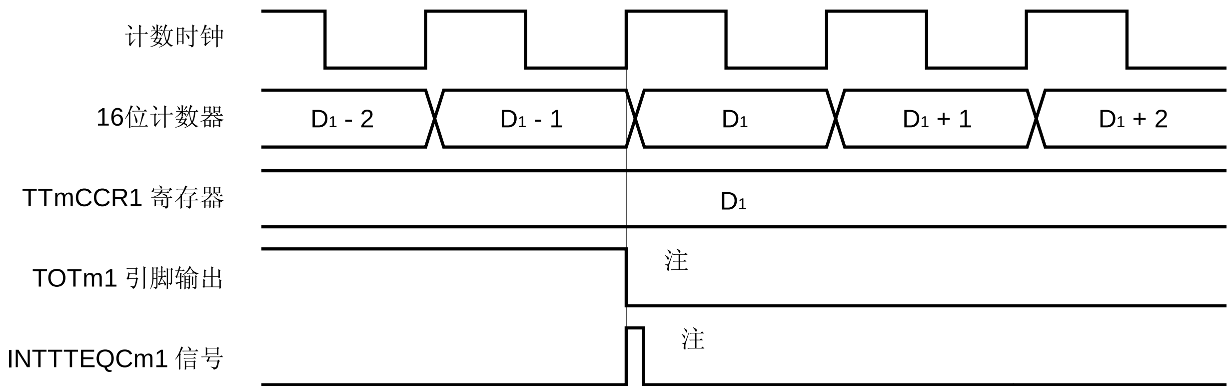
假设 $D_{00} > D_{01}$ 且 $D_{10} > D_{11}$，当 TTmCCR0 寄存器由 D_{00} 重写为 D_{01}，TTmCCR1 寄存器由 D_{10} 重写为 D_{11} 时，如果在 16 位计数器的计数值大于 D_{11} 且小于 D_{10} 时重写 TTmCCR1 寄存器，并且在计数值大于 D_{01} 且小于 D_{00} 时重写 TTmCCR0 寄存器，一旦重写寄存器，就会在各设置值上反映出来，并立即与计数值进行比较。计数器向上计数到 FFFFH 后，从 0000H 开始重新向上计数。当计数值与 D_{11} 一致时，计数器会产生 INTTTEQCm1 信号并使 TOTm1 引脚有效。当计数值与 D_{01} 一致时，计数器产生 INTTTEQCm0 信号，使 TOTm1 引脚失效，并停止计数。

因此，计数器输出脉冲的延迟时段或有效时段可能会与原先预计的单次触发脉冲有所不同。

备注　V850E/IF3：m = 1, a = 0, 1
　　　V850E/IG3：m = 0, 1, a = 0, 1

（**b**） 比较一致中断请求信号（**INTTTEQCm1**）的发生时序

单次触发脉冲输出模式下，比较一致中断请求信号（INTTTEQCm1）的发生时序和其它情况下的 INTTTEQCm1 信号发生时序不同；当 16 位计数器的计数值与 TTmCCR1 寄存器的值一致时，产生 INTTTEQCm1 中断请求信号。



注　时序实际被延迟一个操作时钟（fxx）。

备注　V850E/IF3：m = 1
V850E/IG3：m = 0, 1

通常在 16 位计数器的当前计数值与 TTmCCR1 寄存器的值一致之后，在下一次向上计数时发生 INTTTEQCm1 信号。

但是，在单次触发脉冲输出模式下，会提早一个计数时钟周期产生。该时序的变化是为了一致 TOTm1 引脚的改变时序。

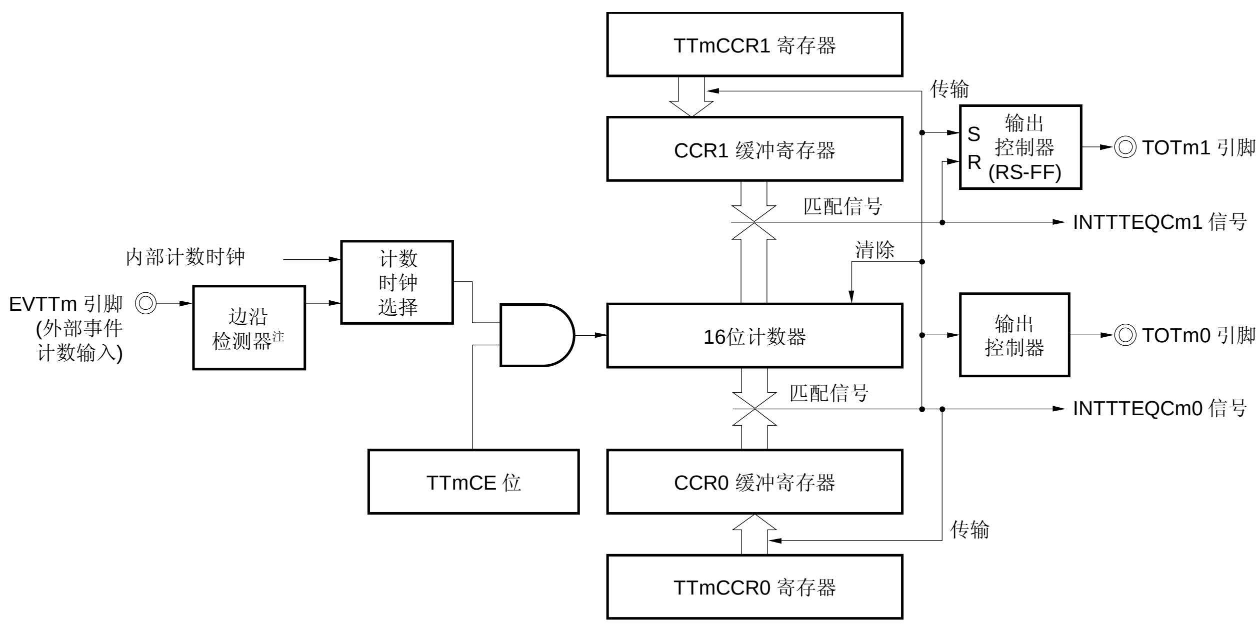
8.6.5　PWM 输出模式（TTmMD3 至 TTmMD0 位＝0100）

只有在 TMT0(仅限 V850E/IG3)和 TMT1 中，该模式下有效。

在 PWM 输出模式下，当 TTmCTL0.TTmCE 位被置为 1 时，PWM 波形从 TOTm1 引脚输出。

而且，从 TOTm0 引脚输出的占空比 50%的 PWM 波形，其半周期等于 TTmCCR0 寄存器的设置值 + 1。

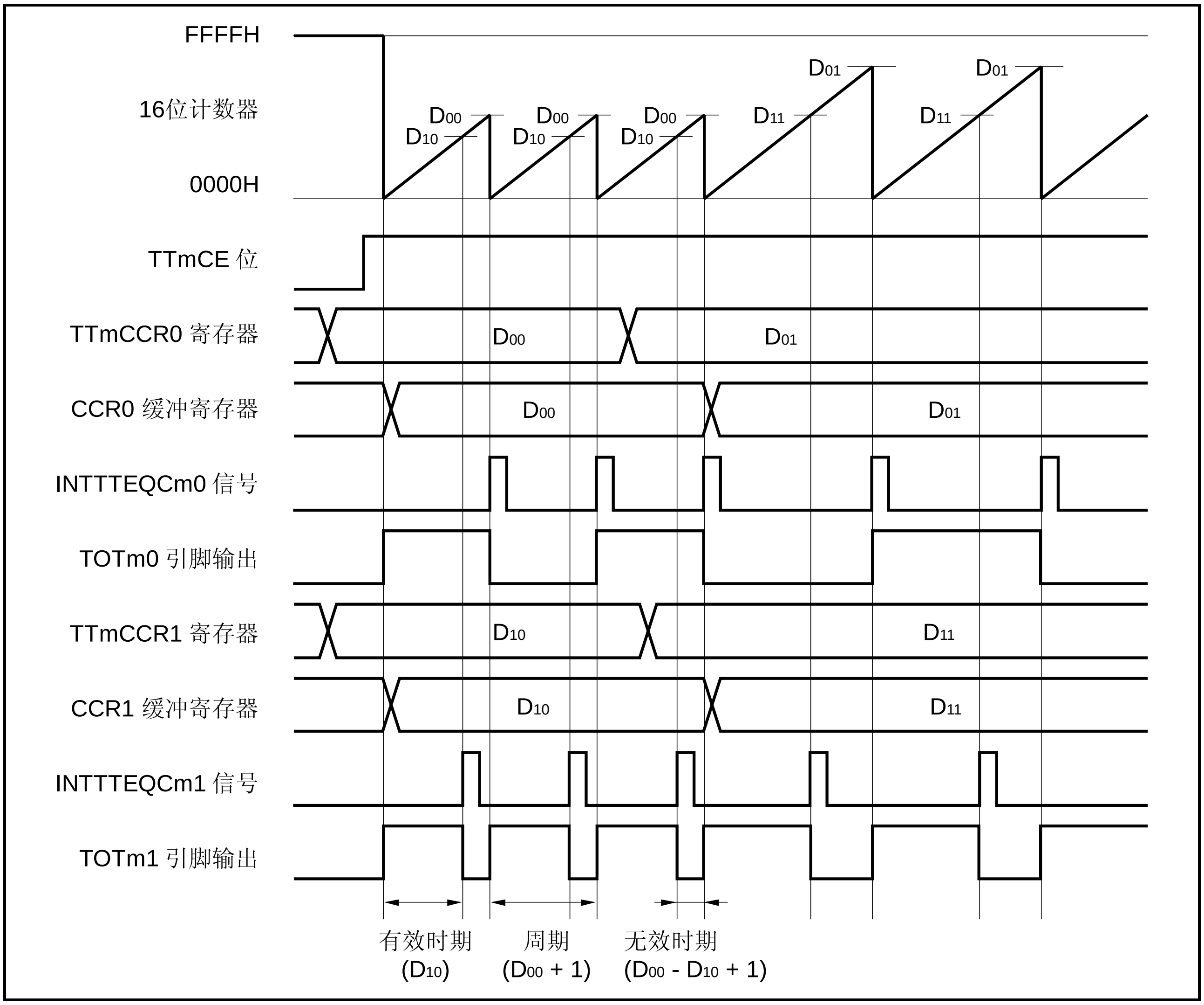
图 8-29. PWM 输出模式的配置



注　　通过 TTmIOC2.TTmEES1 和 TTmIOC2.TTmEES0 位设置。

备注　V850E/IF3：m = 1
V850E/IG3：m = 0, 1

图 8-30. PWM 输出模式的基本时序



当 TTmCE 位置为 1 时，16 位计数器从 FFFFH 清零为 0000H，计数开始，同时从 TOTm1 引脚输出 PWM 波形。PWM 波形的有效电平宽度、周期、占空比系数的计算如下。

有效电平宽度＝（TTmCCR1 寄存器的设置值）× 计数时钟周期
周期＝（TTmCCR0 寄存器的设置值 ＋1）× 计数时钟周期
占空比系数＝（TTmCCR1 寄存器的设置值）/（TTmCCR0 寄存器的设置值 ＋1）

在计数操作时，可通过重写 TTmCCRa 寄存器来改变 PWM 波形。当 16 位计数器的计数值与 CCR0 缓冲寄存器的值一致时，16 位计数器清零为 0000H 时，新写入的值会发生作用。

当 16 位计数器的计数值与 CCR0 缓冲寄存器的值一致之后，进行下一次计数时，会产生比较一致中断请求信号（INTTTEQCm0），且 16 位计数器清零为 0000H。当 16 位计数器的计数值与 CCR1 缓冲寄存器的值一致时，会生成比较一致中断请求信号（INTTTEQCm1）。

当 16 位计数器的计数值与 CCRa 缓冲寄存器的值一致时，TTmCCRa 寄存器的设置值被传输到 CCRa 缓冲寄存器，且 16 位计数器被清除为 0000H。

备注　V850E/IF3：m = 1, a = 0, 1, V850E/IG3：m = 0, 1, a = 0, 1

图 8-31. PWM 输出模式时的寄存器设置（1/2）

（a） TMTm 控制寄存器 0 （TTmCTL0）

	TTmCE					TTmCKS2	TTmCKS1	TTmCKS0
TTmCTL0	0/1	0	0	0	0	0/1	0/1	0/1

TTmCKS2～TTmCKS0：选择计数时钟[注]

TTmCE：
0：停止计数
1：使能计数

注　　当 TTmCTL1.TTmEEE 位＝1 时，设置无效。

（b） TMTm 控制寄存器 1 （TTmCTL1）

		TTmEST	TTmEEE		TTmMD3	TTmMD2	TTmMD1	TTmMD0
TTmCTL1	0	0	0/1	0	0	1	0	0

TTmMD3～TTmMD0：
0, 1, 0, 0:
PWM 输出模式

TTmEEE：
0: 计数时钟由 TTmCKS0 至 TTmCKS2 位选择
1: 用外部事件计数输入信号计数

（c） TMTm I/O 控制寄存器 0 （TTmIOC0）

					TTmOL1	TTmOE1	TTmOL0	TTmOE0
TTmIOC0	0	0	0	0	0/1	0/1	0/1	0/1

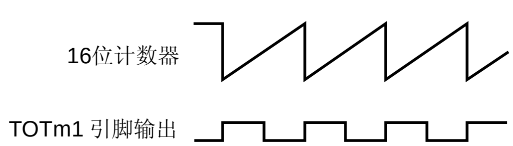
TTmOE0：
0: 禁止 TOTm0 引脚输出
1: 使能 TOTm0 引脚输出

TTmOL0：
在计数操作之前，TOTm0 引脚输出电平的设置
0: 低电平
1: 高电平

TTmOE1：
0: 禁止 TOTm1 引脚输出
1: 使能 TOTm1 引脚输出

TTmOL1：
在计数操作之前，TOTm1 引脚输出电平的设置
0: 低电平
1: 高电平

• 当 TTmOL1 位 = 0



• 当 TTmOL1 位 = 1

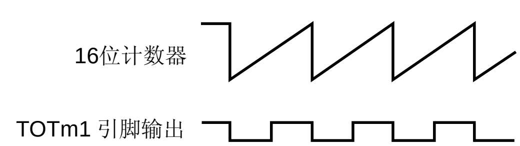


图 8-31. PWM 输出模式时的寄存器设置（2/2）

（d） **TMTm I/O 控制寄存器 2 （TTmIOC2）**

					TTmEES1	TTmEES0	TTmETS1	TTmETS0
TTmIOC2	0	0	0	0	0/1	0/1	0	0

选择外部事件计数输入
(EVTTm 引脚)的有效边沿.

（e） **TMTm 计数器读取缓冲寄存器 （TTmCNT）**

通过读取 TTmCNT 寄存器，可以获得 16 位计数器的值。

（f） **TMTm 捕获/比较寄存器 0 和 1（TTmCCR0 和 TTmCCR1）**

如果将 TTmCCR0 寄存器的值设置为 D_0，将 TTmCCR1 寄存器的值设置为 D_1，则 PWM 波形的周期和有效电平的计算如下：

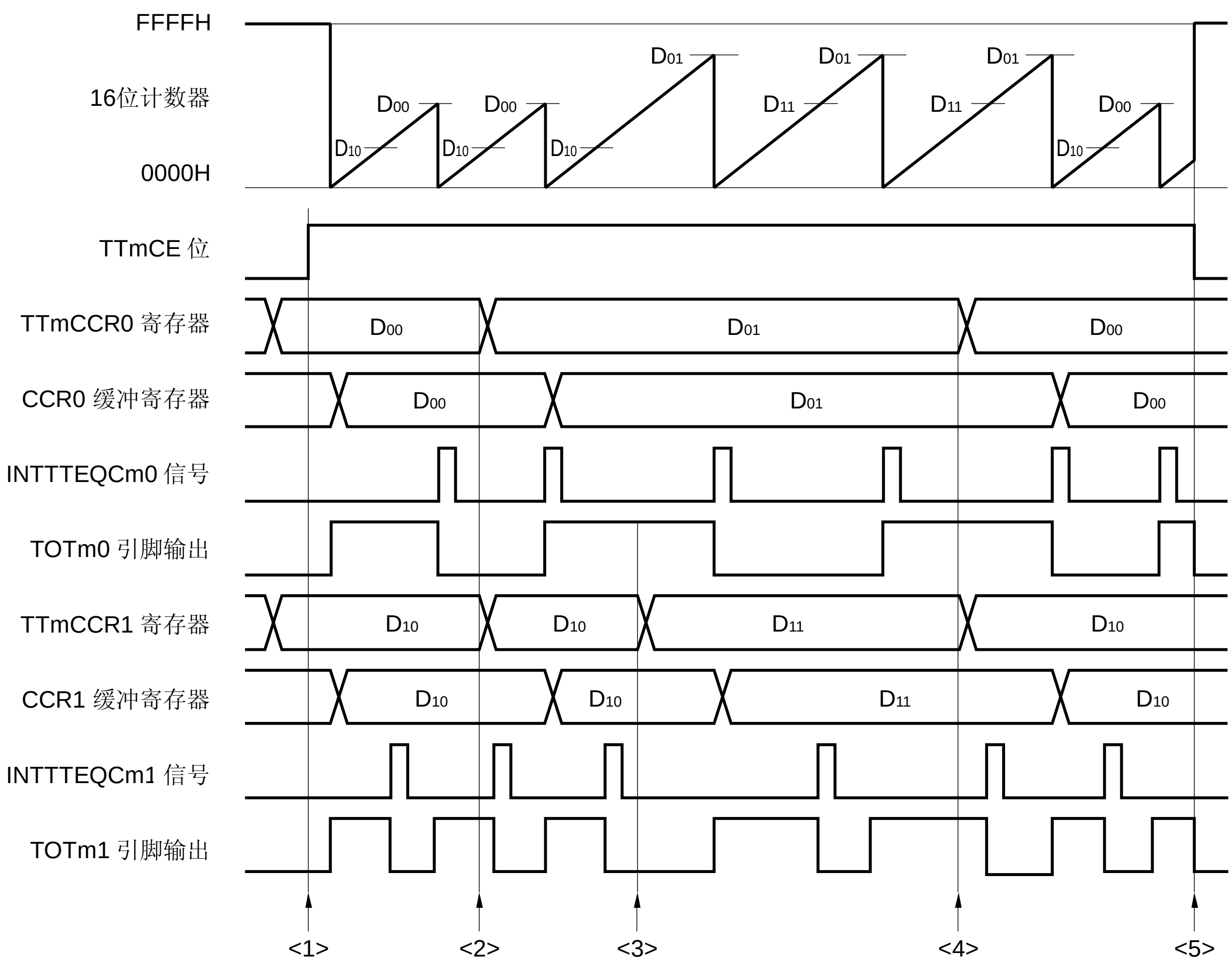
周期＝（D_0 ＋1）× 计数时钟周期
有效电平宽度＝D_1 × 计数时钟周期

备注　1. PWM 输出模式不使用 TMTm 控制寄存器 2（TTmCTL2）、TMTm I/O 控制寄存器 1（TTmIOC1）、TMTm I/O 控制寄存器 3（TTmIOC3）、TMTm 选项寄存器 0（TTmOPT0）、TMTm 选项寄存器 1（TTmOPT1）和 TMTm 计数器写入寄存器（TTmTCW）。

2. V850E/IF3：m = 1
V850E/IG3：m = 0, 1

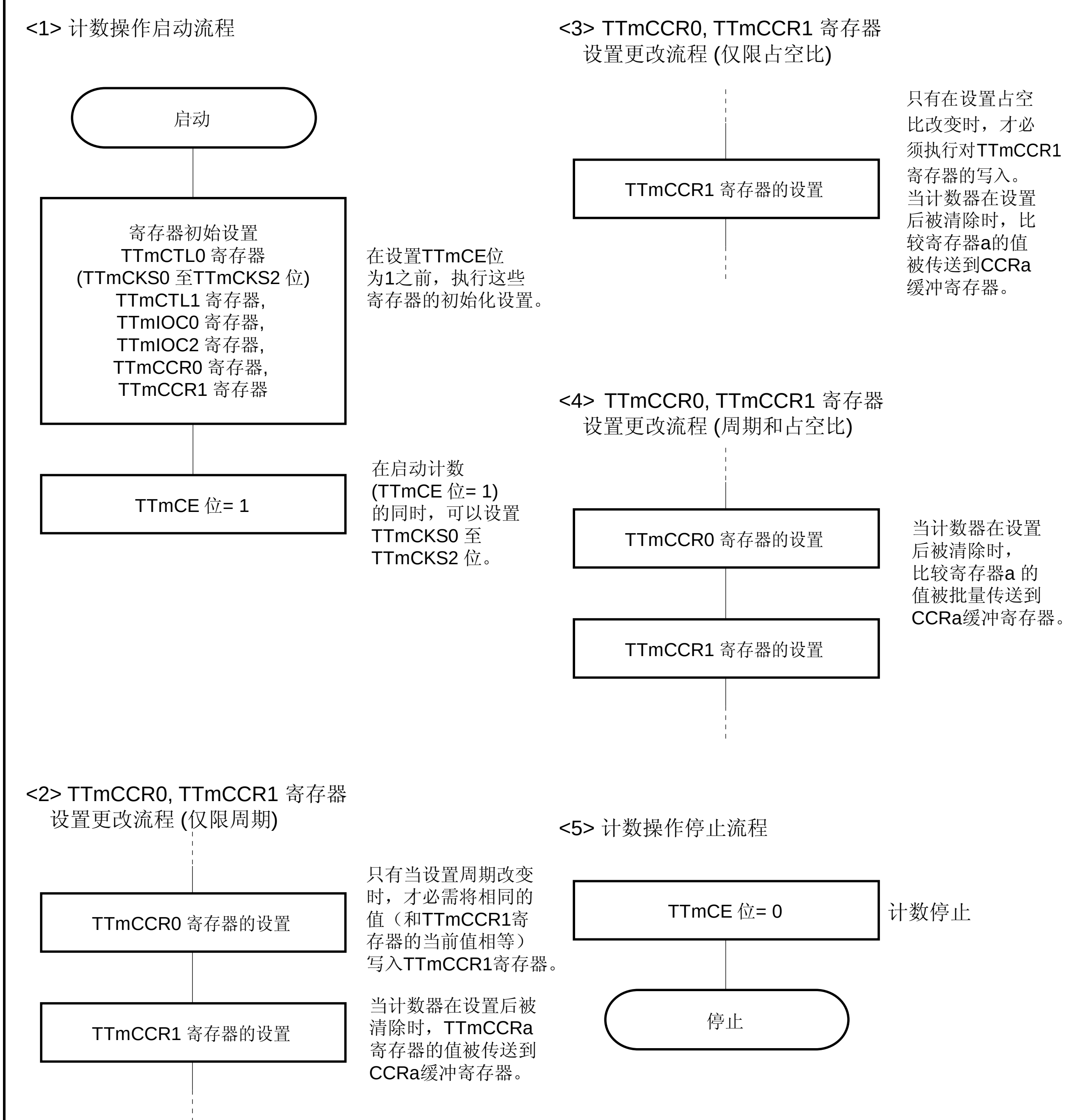
（1）**PWM** 输出模式下的操作流程

图 **8-32. PWM** 输出模式的软件处理流程（**1/2**）



备注　V850E/IF3：m = 1
　　　V850E/IG3：m = 0, 1

图 8-32. PWM 输出模式时的软件处理流程（2/2）



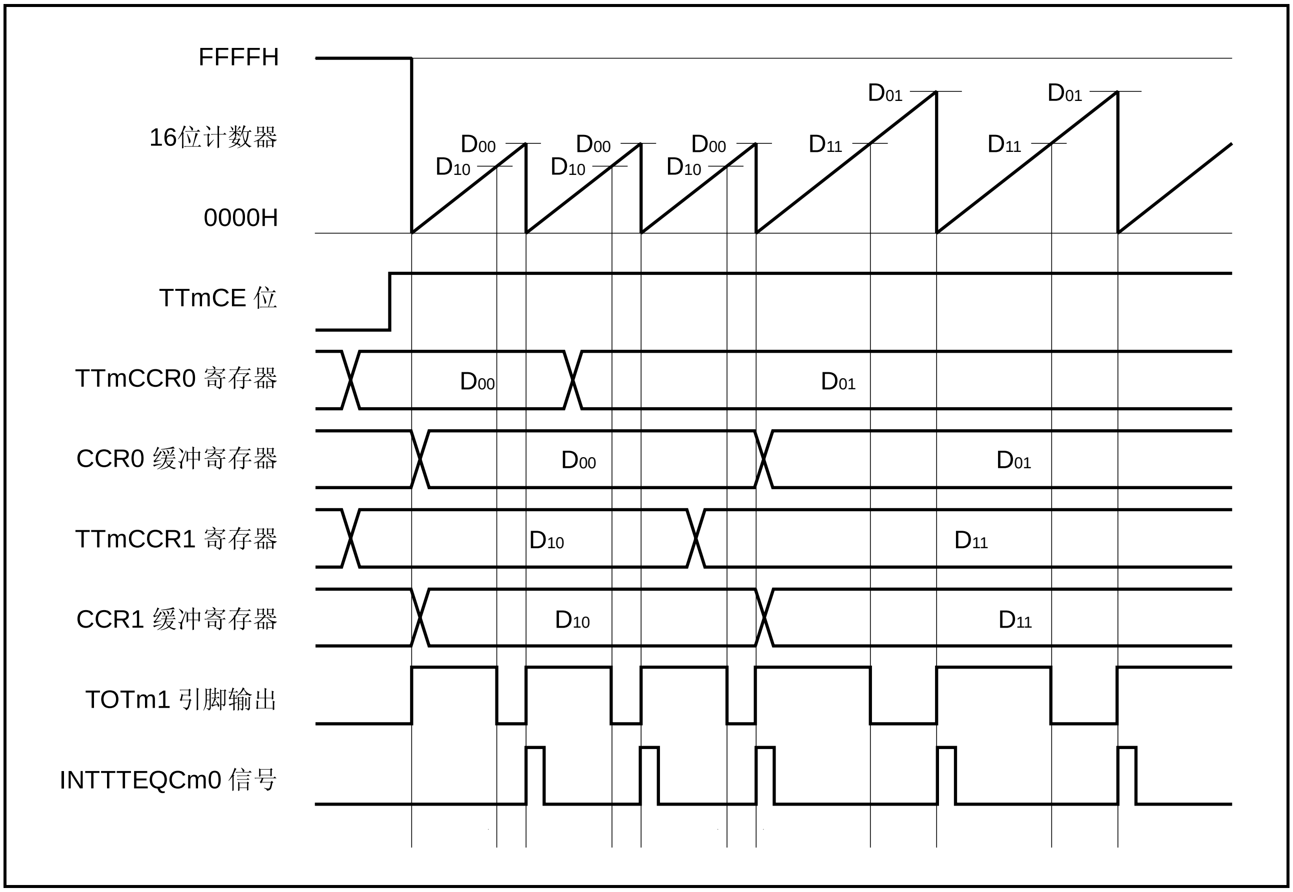
备注　V850E/IF3：m = 1, a = 0, 1
V850E/IG3：m = 0, 1, a = 0, 1

（2）PWM 输出模式的操作时序

（a）操作过程中改变脉冲宽度

计数器操作过程中，若要改变 PWM 波形，应最后写入 TTmCCR1 寄存器。

当检测到 INTTTEQCm0 信号以后，在写入 TTmCCR1 寄存器之后重写 TTmCCRa 寄存器。



为了将数据从 TTmCCRa 寄存器传输到 CCRa 缓冲寄存器，必须对 TTmCCR1 寄存器进行写入操作。

此时如果 PWM 波形的周期和有效电平都要改变，应先将周期设定到 TTmCCR0 寄存器，再将有效电平宽度设定到 TTmCCR1 寄存器。

如果只改变 PWM 波形的周期，应先将周期设置到 TTmCCR0 寄存器，然后将相同的值（和 TTmCCR1 寄存器当前值相等）写入 TTmCCR1 寄存器。

如果只改变 PWM 波形的有效电平宽度（占空比系数），仅需要设置 TTmCCR1 寄存器。

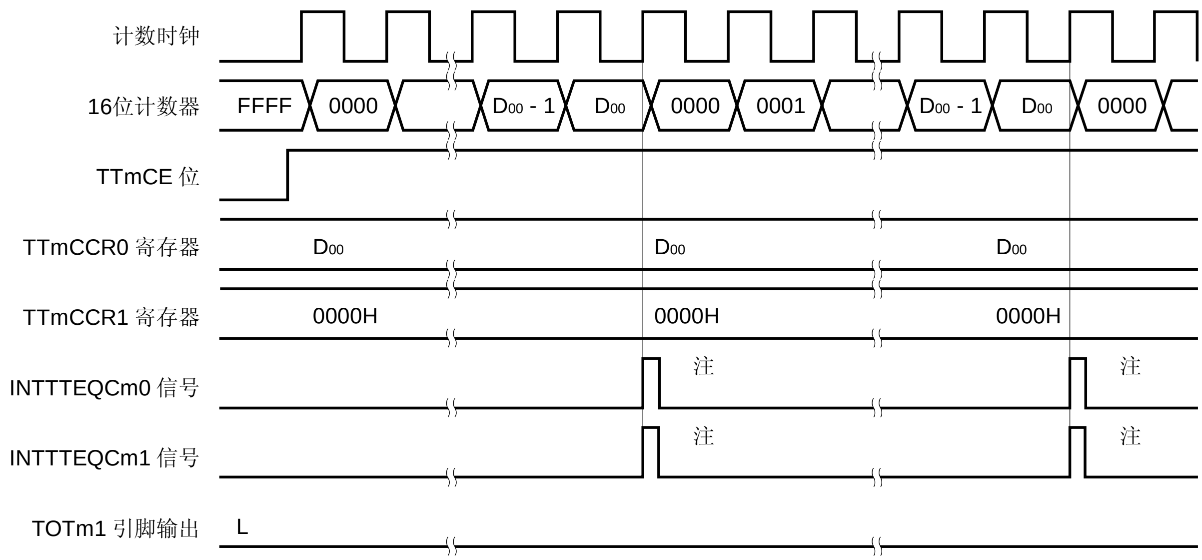
当数据写入 TTmCCR1 寄存器之后，在对 16 位计数器清零的同时，写入到 TTmCCRa 寄存器的值会被传输到 CCRa 缓冲寄存器，并作为与 16 位计数器进行比较的值。

在写入 TTmCCR1 寄存器一次之后，要再次写入 TTmCCR0 或 TTmCCR1 寄存器，应该在 INTTTEQCm0 信号产生之后进行。否则，CCRa 缓冲寄存器的值可能不确定，因为把数据从 TTmCCRa 寄存器传输到 CCRa 缓冲寄存器的时序会和写入 TTmCCRa 寄存器的时序发生冲突。

备注　V850E/IF3：m = 1, a = 0, 1, V850E/IG3：m = 0, 1, a = 0, 1

（b） PWM 波形的 0%和 100% 输出

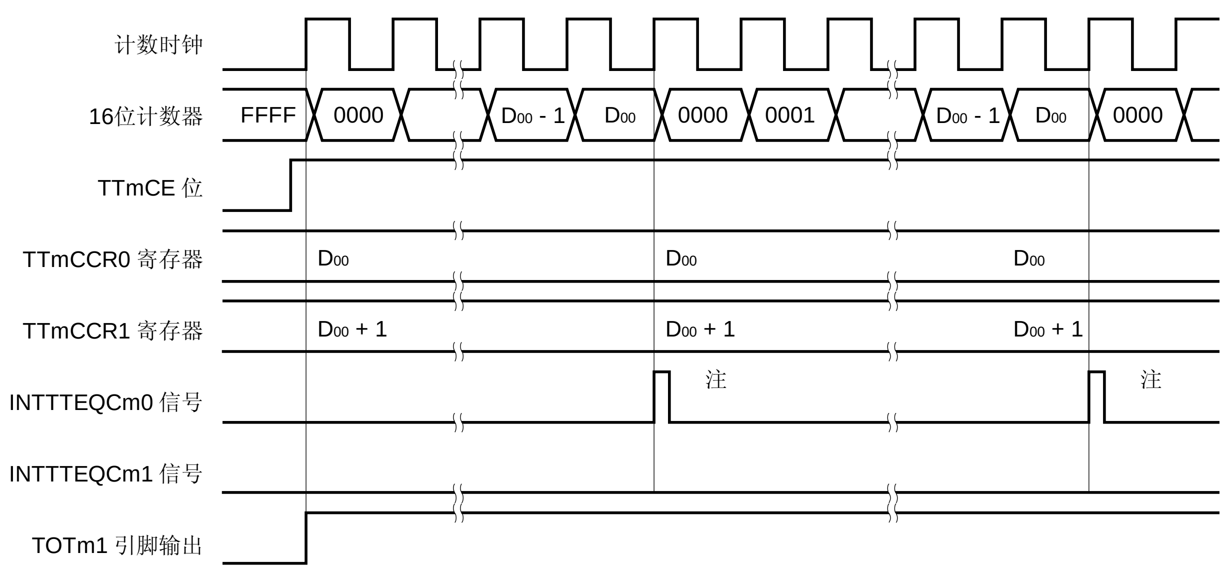
为了输出 0%的波形，需要将 TTmCCR1 寄存器设置为 0000H。16 位计数器的计数值和 CCR0 缓冲寄存器一致之后，在下一个时序处 16 位计数器被清除为 0000H，且产生 INTTTEQCm0 和 INTTTEQCm1 中断请求信号。



注　时序实际被延迟一个操作时钟 （fxx）。

备注　V850E/IF3：m = 1
V850E/IG3：m = 0, 1

为了输出 100%的波形，需要将 TTmCCR1 寄存器的值设为（TTmCCR0 寄存器的设置值 ＋1）。如果 TTmCCR0 寄存器的设置值为 FFFFH，则不会产生 100%波形输出。

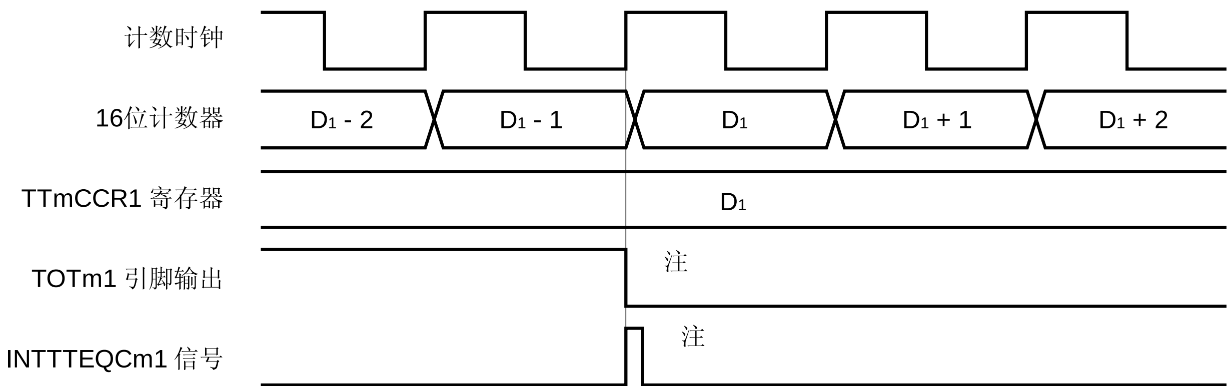


注　时序实际被延迟一个操作时钟 （fxx）。

备注　V850E/IF3：m = 1
V850E/IG3：m = 0, 1

（c） 比较一致中断请求信号（INTTTEQCm1）的发生时序

在 PWM 输出模式下，INTTTEQCm1 信号的发生时序与其它情况下 INTTTEQCm1 信号的发生时序有所不同；当 16 位计数器的计数值与 TTmCCR1 寄存器的值一致时，产生 INTTTEQCm1 信号。



注　　时序实际被延迟一个操作时钟（fxx）。

备注　　V850E/IF3：m = 1
V850E/IG3：m = 0, 1

通常在 16 位计数器的计数值与 TTmCCR1 寄存器的值一致之后，在下一次计数时同步发生 INTTTEQCm1 信号。

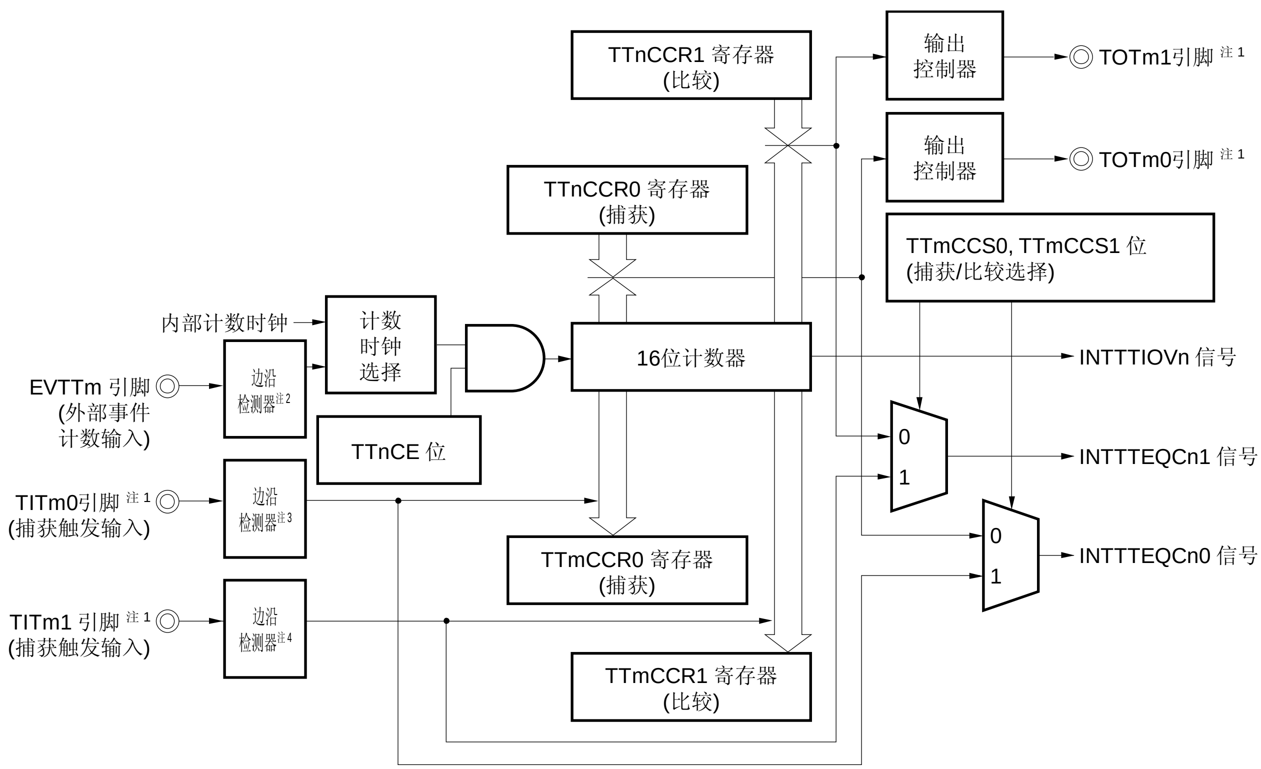
但是，在 PWM 输出模式下，会提早一个计数时钟周期产生。该时序的变化是为了一致 TOTm1 引脚输出信号变化的时序。

8.6.6　自由运行定时器模式（TTnMD3 至 TTnMD0 位＝0101）

在所有 TMT0 和 TMT1 中，比较功能都有效。只有在 TMT0(仅限 V850E/IG3)和 TMT1 中，捕获功能有效。

在自由运行定时器模式下，当 TTnCTL0.TTnCE 位置为 1 时，16 位定时器/事件计数器 T 开始计数。此时，根据 TTmOPT0.TTmCCS0 和 TTmOPT0.TTmCCS1 位的设置，TTmCCR0 和 TTmCCR1 寄存器可以用作比较寄存器或捕获寄存器。

图 8-33. 自由运行定时器模式的配置



注　**1.** 因为捕获触发输入引脚 （TITm0, TITm1） 和定时器输出引脚（TOTm0, TOTm1）共用相同的复用功能引脚，这两个功能不能同时使用。

2. 通过 TTmIOC2.TTmEES1 和 TTmIOC2.TTmEES0 位设置。

3. 通过 TTmIOC1.TTmIS1 和 TTmIOC1.TTmIS0 位设置。

4. 通过 TTmIOC1.TTmIS3 和 TTmIOC1.TTmIS2 位设置。

备注　V850E/IF3：m = 1

V850E/IG3：m = 0, 1

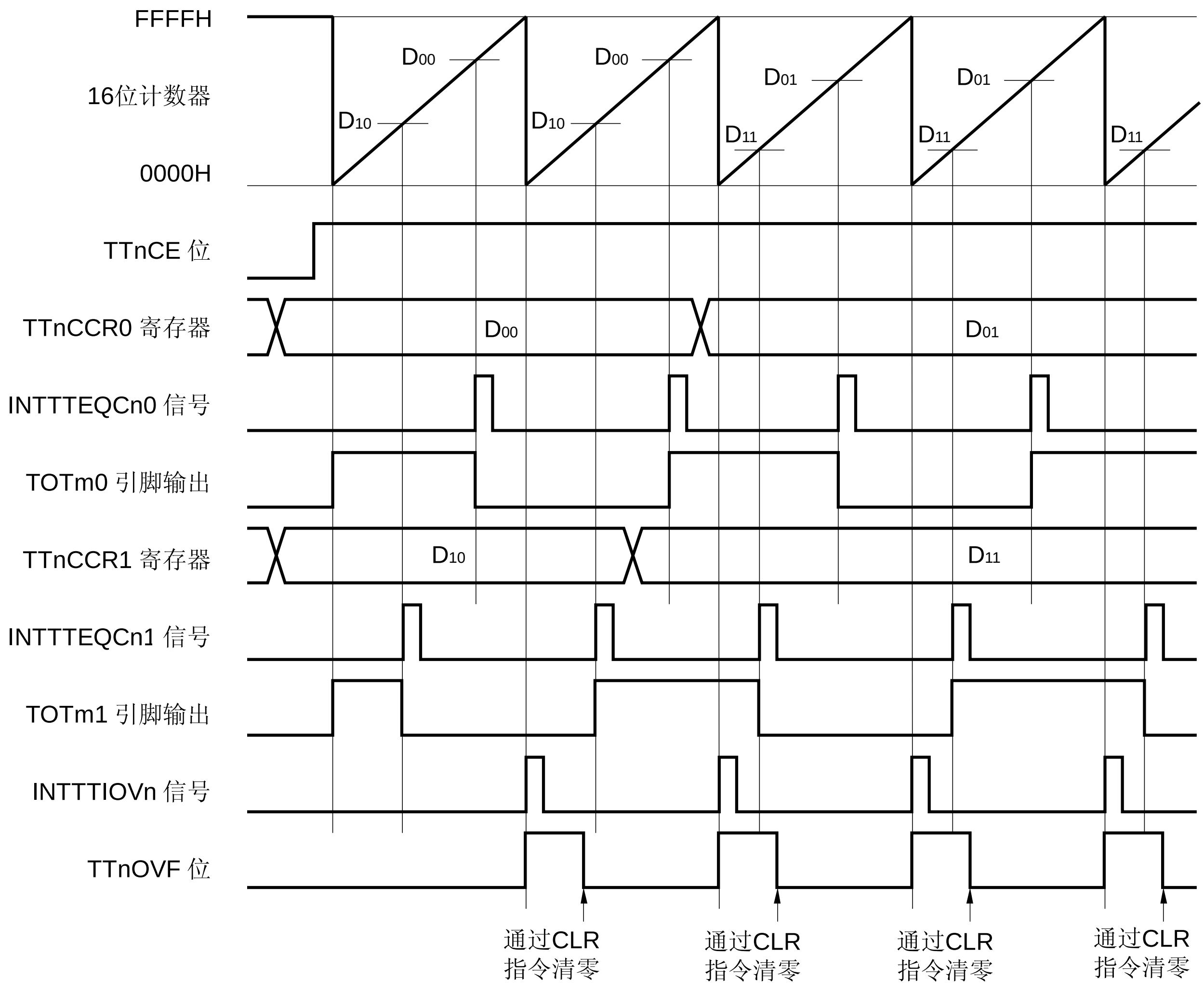
• 比较操作

当 TTnCE 位被置为 1 时，16 位定时器/事件计数器 T 开始计数，TOTma 引脚的输出信号被反转。此后当 16 位计数器的计数值与 TTnCCRa 寄存器的设置值一致时，产生一个比较一致中断请求信号（INTTTEQCna），而且 TOTna 引脚的输出信号被反转。

16 位计数器继续对计数时钟同步计数。当它计数到 FFFFH 时，会在下一个计数时钟时产生一个溢出中断请求信号（INTTTIOVn），同时计数器清零为 0000H，然后继续进行计数。此时，溢出标志（TTnOPT0.TTnOVF 位）被置为 1，通过软件确认溢出标志为 1 然后运行 CLR 指令清除为 0。

当计数器工作时，可以重写 TTnCCRa 寄存器。如果进行重写，则新写入的值会在“任意时刻写入”时反映出来，并与计数值进行比较。

图 8-34. 自由运行定时器模式下的基本时序（比较功能）



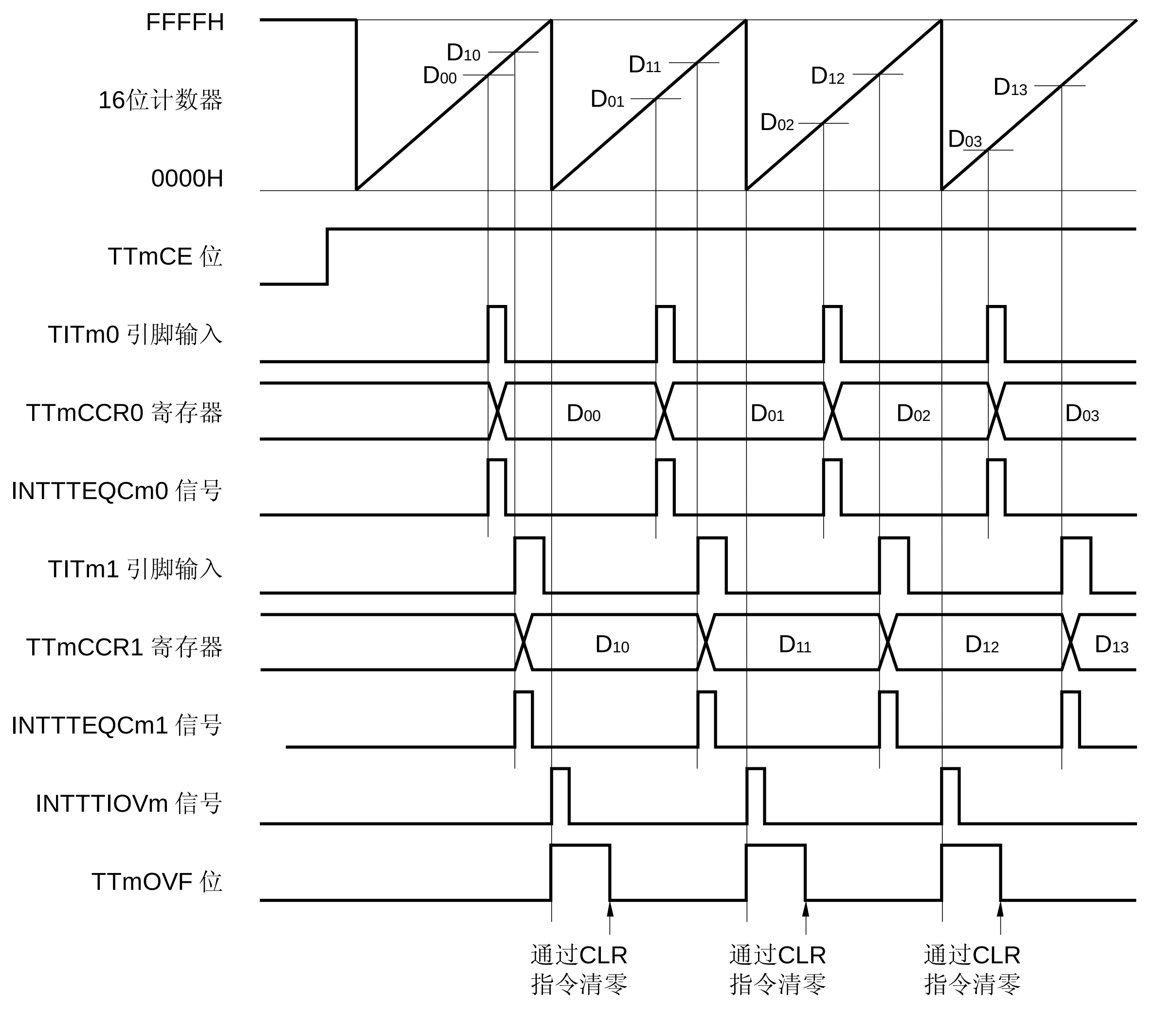
备注　V850E/IF3：n = 0, 1, m = 1, a = 0, 1
V850E/IG3：n = 0, 1, m = 0, 1, a = 0, 1

• 捕获操作

当 TTmCE 位被置为 1 时，16 位计数器开始计数。当检测到 TITma 引脚输入信号的有效边沿时，16 位计数器的计数值存储在 TTmCCRa 寄存器中，并产生一个捕获中断请求信号（INTTTEQCma）。

16 位计数器继续对计数时钟同步计数。当它计数到 FFFFH 时，会在下一个计数时钟时产生一个溢出中断请求信号（INTTTIOVm），同时计数器清零为 0000H，然后继续进行计数。此时，溢出标志（TTmOPT0.TTmOVF 位）被置为 1，通过软件确认溢出标志为 1 然后运行 CLR 指令溢出标志位清 0。

图 8-35. 自由运行定时器模式下的基本时序 （捕获功能）



备注　V850E/IF3：m = 1
　　　V850E/IG3：m = 0, 1

图 8-36. 自由运行定时器模式下的寄存器设置（1/2）

（a） **TMTn 控制寄存器 0 （TTnCTL0）**

	TTnCE					TTnCKS2	TTnCKS1	TTnCKS0
TTnCTL0	0/1	0	0	0	0	0/1	0/1	0/1

TTnCKS2～TTnCKS0：选择计数时钟[注]

TTnCE：
0: 停止计数
1: 使能计数

注 当 TTmCTL1.TTmEEE 位＝1 时，设置无效。

（b） **TMTn 控制寄存器 1 （TTnCTL1）**

		TTmEST	TTmEEE		TTnMD3	TTnMD2	TTnMD1	TTnMD0
TTnCTL1	0	0	0/1	0	0	1	0	1

TTnMD3～TTnMD0：
0, 1, 0, 1:
自由运行定时器模式

TTmEEE：
0: 计数时钟由TTmCKS0至TTmCKS2 位选择
1: 用外部事件计数输入信号计数

（c） **TMTm I/O 控制寄存器 0 （TTmIOC0）**

					TTmOL1	TTmOE1	TTmOL0	TTmOE0
TTmIOC0	0	0	0	0	0/1	0/1	0/1	0/1

TTmOE0：
0: 禁止 TOTm0 引脚输出
1: 使能 TOTm0 引脚输出

TTmOL0：
在计数操作之前，
TOTm0引脚输出电平的设置
0: 低电平
1: 高电平

TTmOE1：
0: 禁止 TOTm1 引脚输出
1: 使能 TOTm1 引脚输出

TTmOL1：
在计数操作之前，
TOTm1引脚输出电平的设置
0: 低电平
1: 高电平

（d） **TMTm I/O 控制寄存器 1 （TTmIOC1）**

					TTmIS3	TTmIS2	TTmIS1	TTmIS0
TTmIOC1	0	0	0	0	0/1	0/1	0/1	0/1

TTmIS1, TTmIS0：选择 TITm0 引脚输入的有效边沿

TTmIS3, TTmIS2：选择 TITm1 引脚输入的有效边沿

图 **8-36.** 自由运行定时器模式下的寄存器设置（**2/2**）

（**e**） **TMTm I/O** 控制寄存器 **2** （**TTmIOC2**）

					TTmEES1	TTmEES0	TTmETS1	TTmETS0
TTmIOC2	0	0	0	0	0/1	0/1	0	0

选择外部事件计数输入
(EVTTm 引脚)的有效边沿

（**f**） **TMTn** 选项寄存器 **0**（**TTnOPT0**）

			TTmCCS1	TTmCCS0				TTnOVF
TTnOPT0	0	0	0/1	0/1	0	0	0	0/1

溢出标志

指定TTmCCR0 寄存器
作为捕获寄存器
或比较寄存器
0：比较寄存器
1：捕获寄存器

指定TTmCCR1 寄存器
作为捕获寄存器
或比较寄存器
0：比较寄存器
1：捕获寄存器

（**g**） **TMTn** 计数器读取缓冲寄存器 （**TTnCNT**）

通过读取 TTnCNT 寄存器，可以获得 16 位计数器的值。

(h) TMTn 捕获/比较寄存器 **0** 和 **1 (TTnCCR0** 和 **TTnCCR1)**

根据 TTmOPT0.TTmCCSa 位的设置，这些寄存器可以用作捕获寄存器或比较寄存器。

当用作捕获寄存器时，在检测到 TITma 引脚输入信号的有效边沿时，存储 16 位计数器的当前计数值。

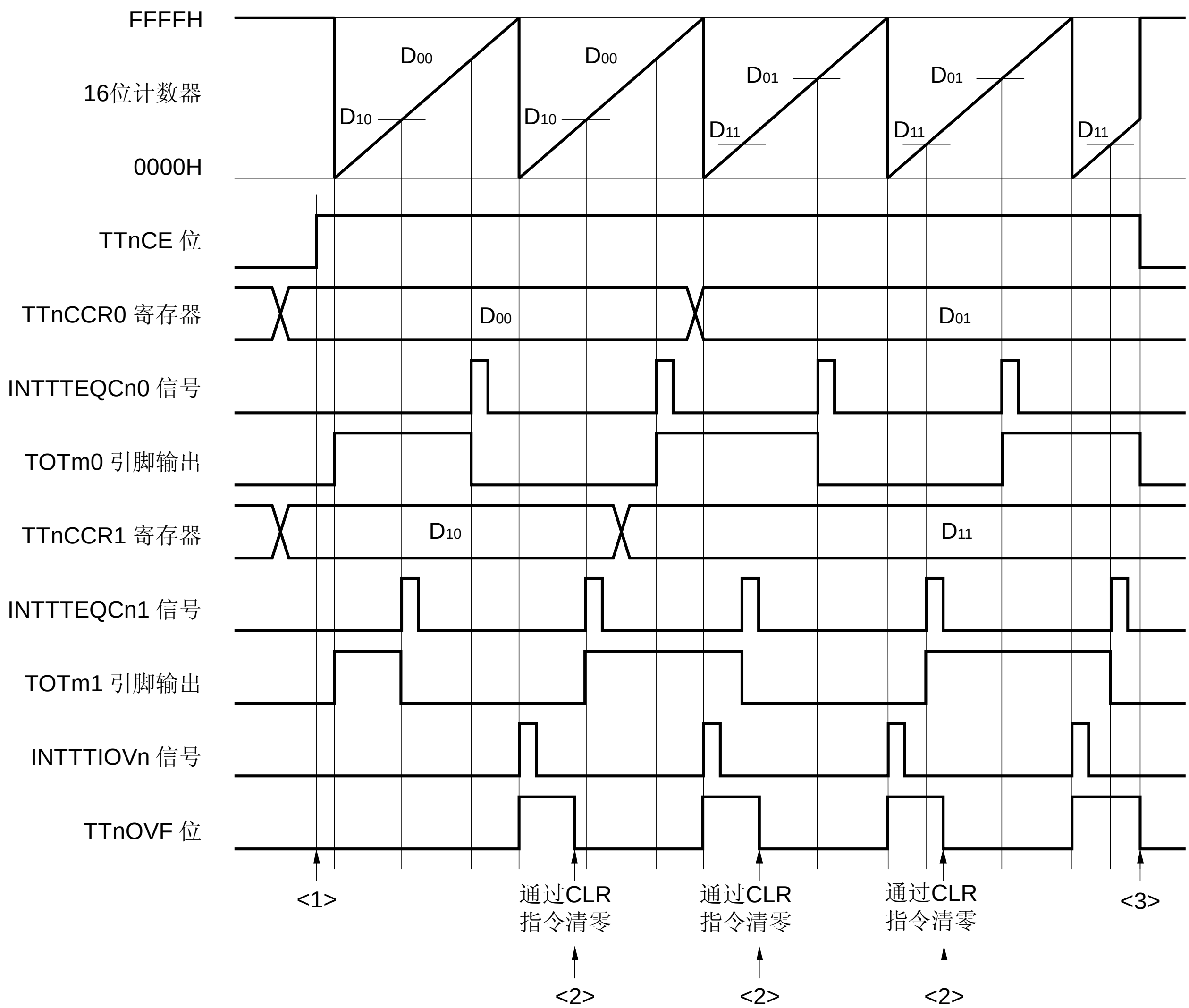
当用作比较寄存器时，而且 TTnCCRa 寄存器的值被设置为 D_a 时，则当计数器达到（D_a＋1）时会产生 INTTTEQCna 中断信号，同时，TOTm0 和 TOTm1 引脚的输出信号被反转。

备注　V850E/IF3：n = 0, 1, m = 1, a = 0, 1
　　　V850E/IG3：n = 0, 1, m = 0, 1, a = 0, 1

（1） 自由运行定时器模式时的操作流程

（a） 当捕获/比较寄存器用作比较寄存器时

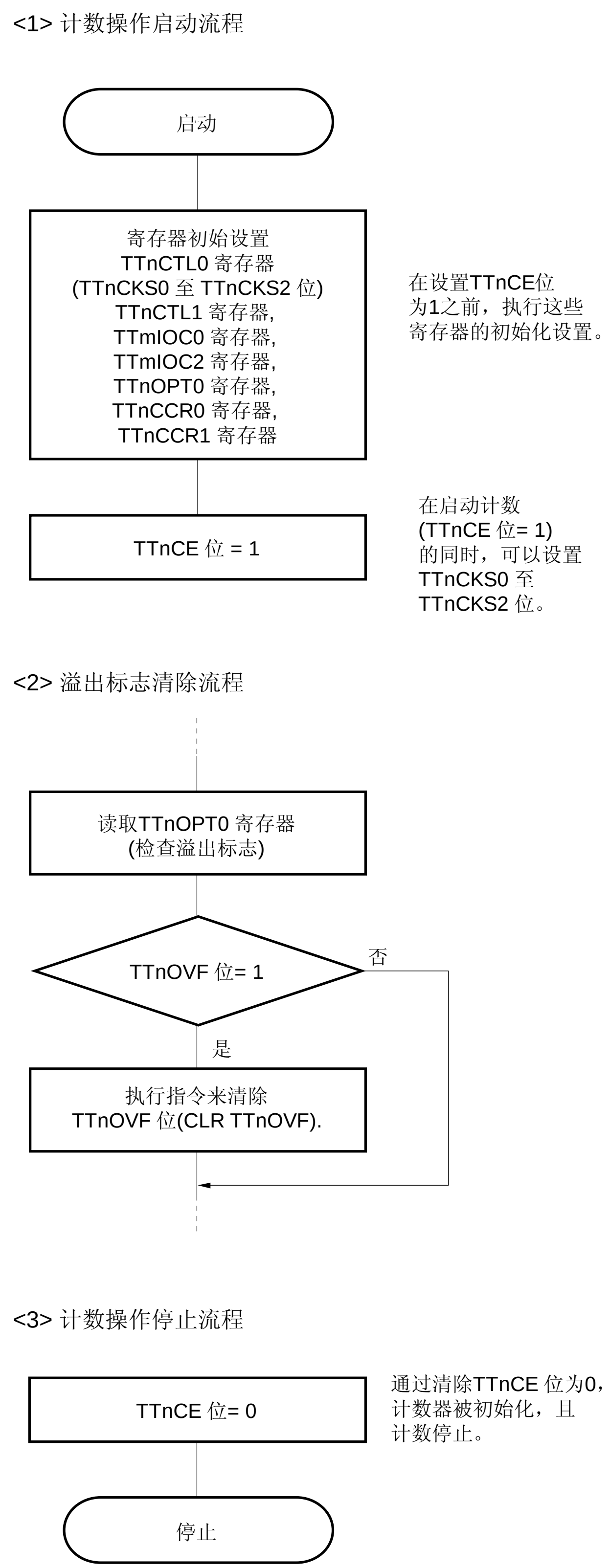
图 8-37. 自由运行定时器模式的软件处理流程（比较功能）（1/2）



备注　V850E/IF3：n = 0, 1, m = 1

V850E/IG3：n = 0, 1, m = 0, 1

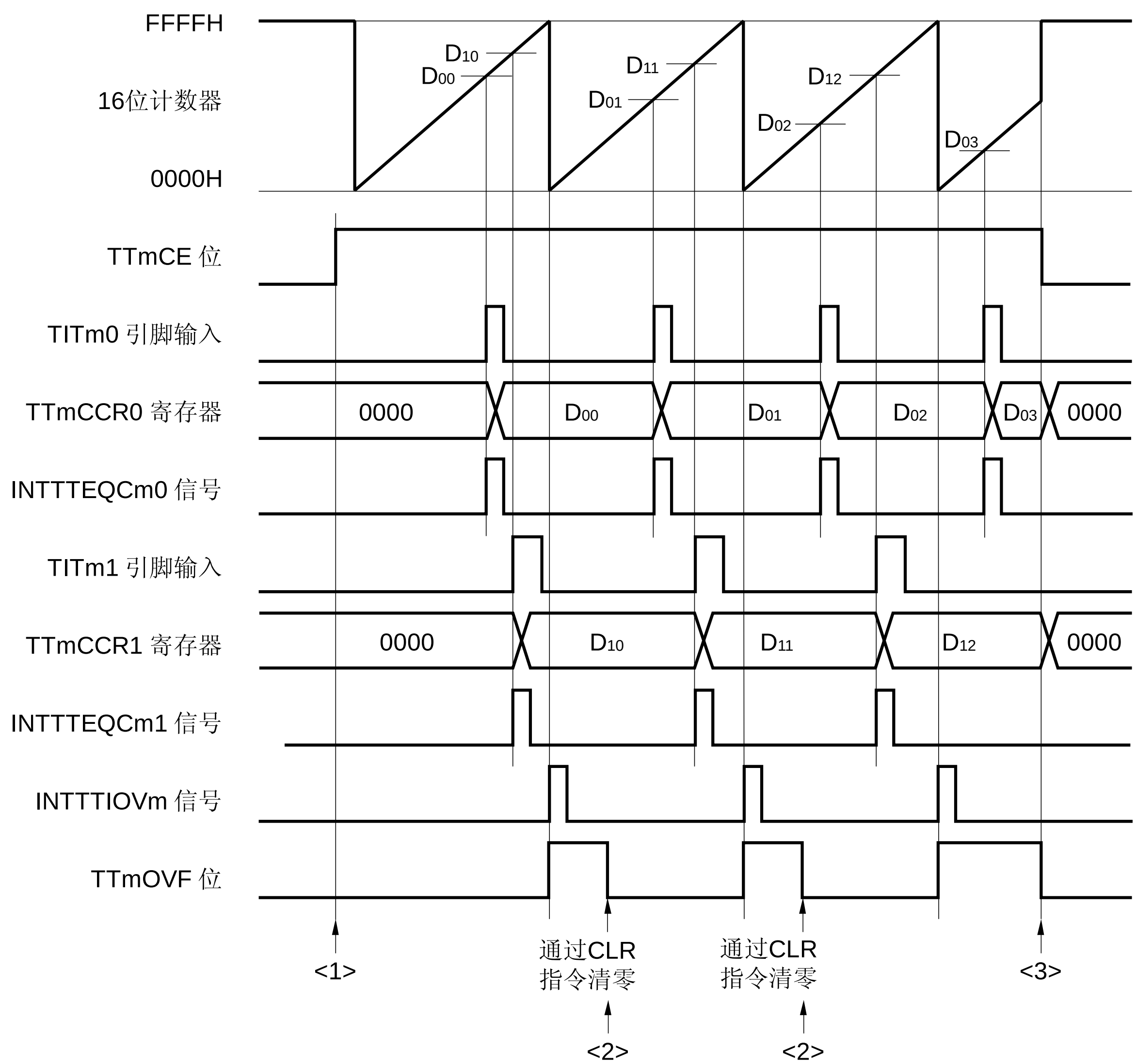
图 8-37. 自由运行定时器模式的软件处理流程（比较功能）（2/2）



备注　V850E/IF3：n = 0, 1, m = 1
　　　V850E/IG3：n = 0, 1, m = 0, 1

（b） 当捕获/比较寄存器用作捕获寄存器时

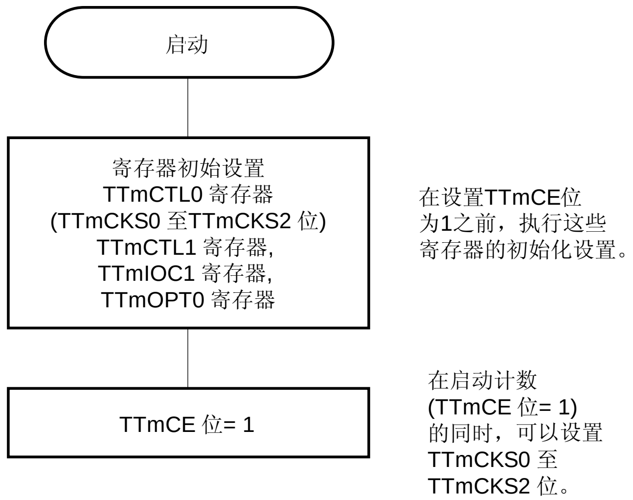
图 8-38. 自由运行定时器模式的软件处理流程（捕获功能） （1/2）



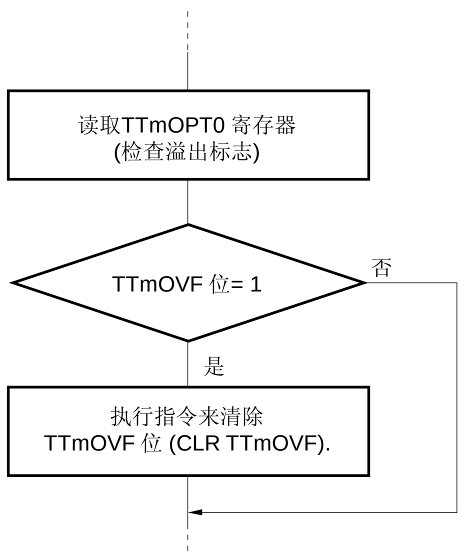
备注 V850E/IF3：m = 1
V850E/IG3：m = 0, 1

图 8-38. 自由运行定时器模式时的软件处理流程（捕获功能）（**2/2**）

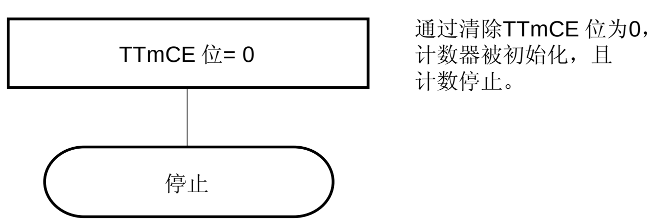
<1> 计数操作启动流程



<2> 溢出标志清除流程



<3> 计数操作停止流程

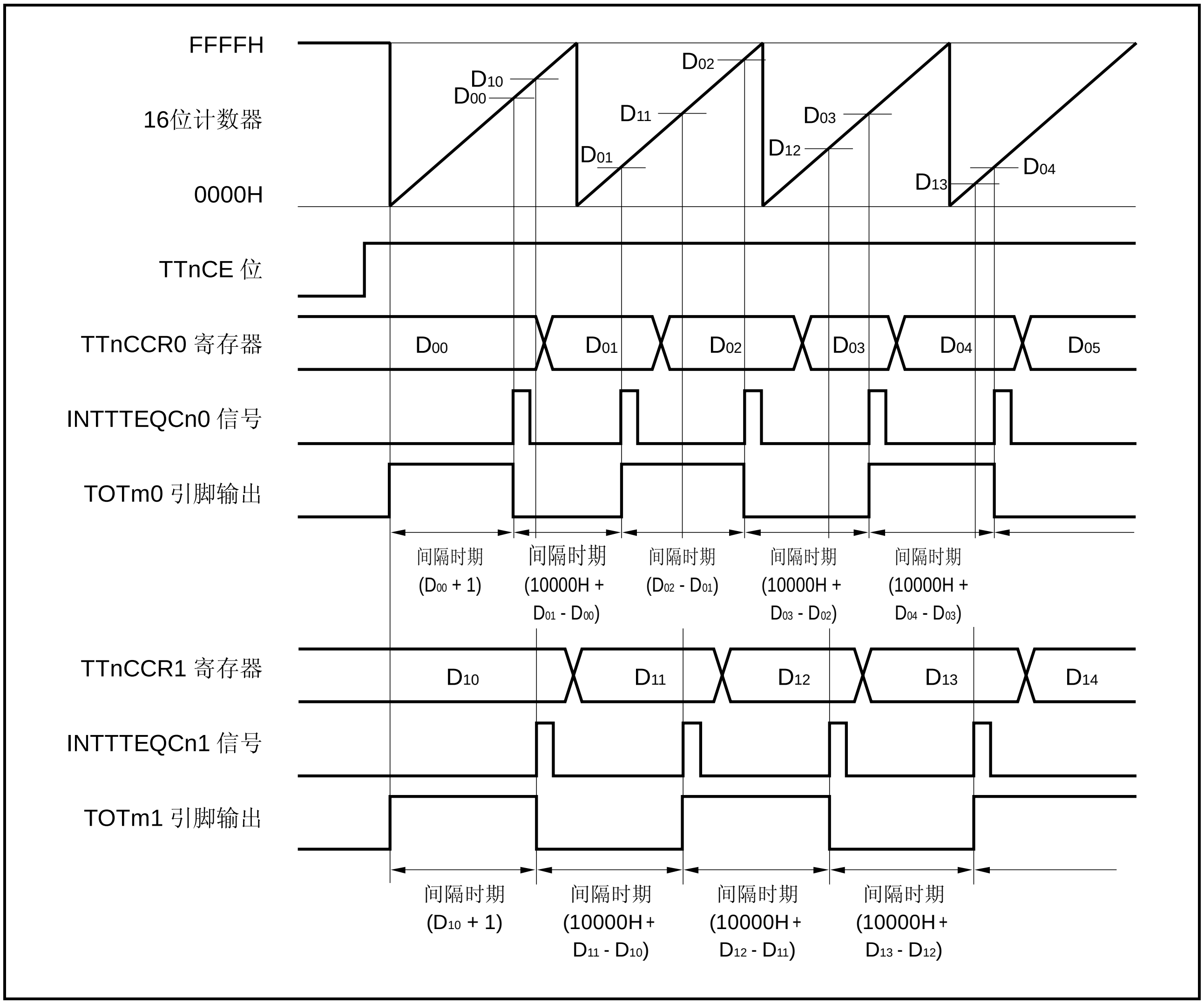


备注　　V850E/IF3：m = 1
　　　　V850E/IG3：m = 0, 1

（2）自由运行定时器模式时的操作时序

（a）比较寄存器的间隔定时操作

当 16 位定时器/事件计数器 T 用作间隔定时器，TTnCCRa 寄存器用作比较寄存器时，在每次检测到 INTTTEQCna 信号后，都需要通过软件设置一个比较值以产生下一个中断请求信号。



在自由运行定时器模式下进行时间间隔操作时，一个通道可以设置两个时间间隔。

要进行时间间隔操作，当检测到 INTTTEQCna 信号后，必须在中断服务程序中重置相应的 TTnCCRa 寄存器的值。

重置 TTnCCRa 寄存器的设置值可由下式计算，其中“D_a”是间隔时间：

比较寄存器默认值：　$D_a - 1$

比较寄存器的第二次及其以后的设置值：上次设置值 ＋ D_a

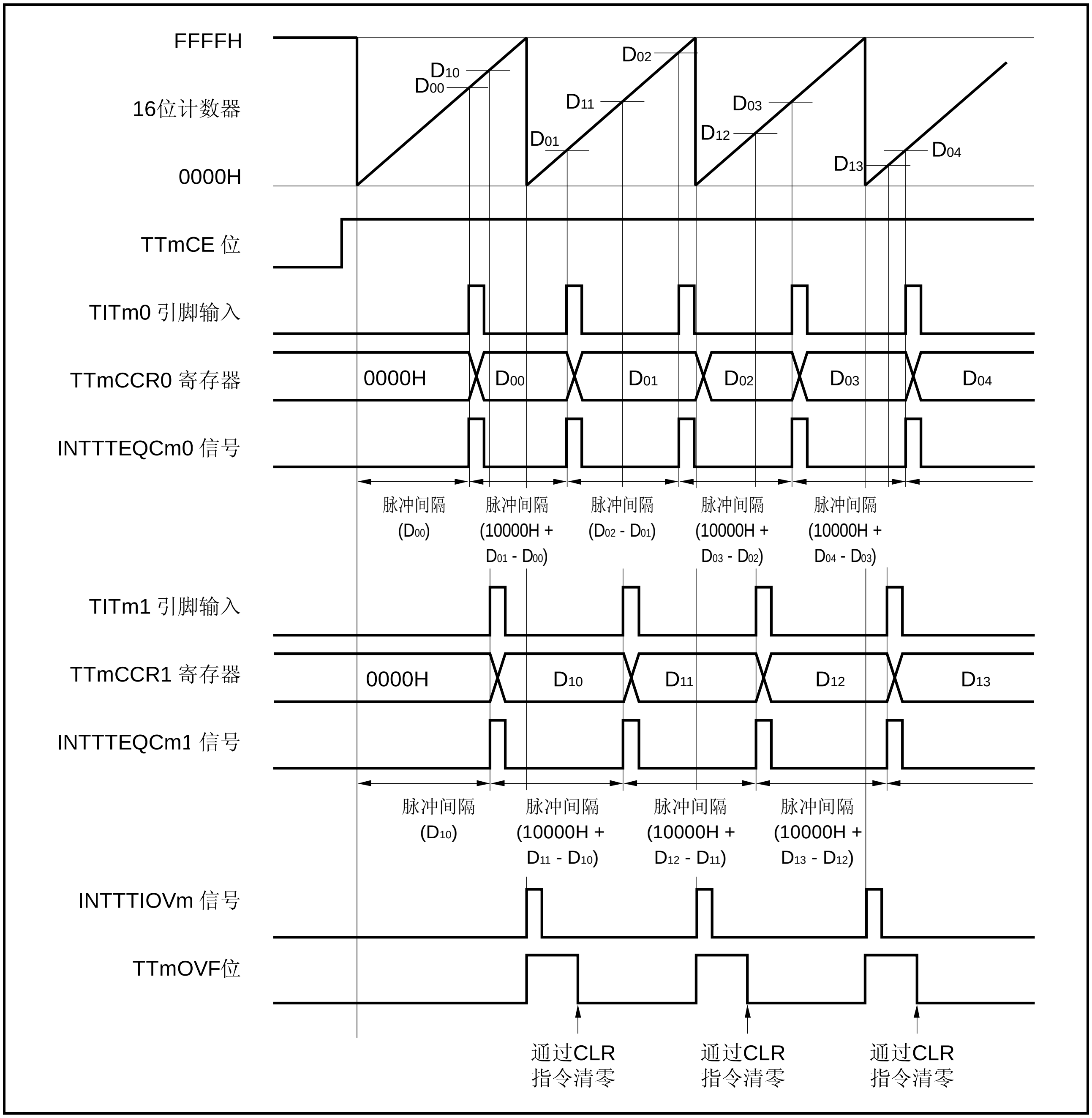
（若计算结果大于 FFFFH，则从结果中减去 10000H，并将此值设置给寄存器。）

备注　　V850E/IF3：n = 0, 1, m = 1, a = 0, 1

　　　　V850E/IG3：n = 0, 1, m = 0, 1, a = 0, 1

（b） 使用捕获寄存器进行脉冲宽度测量

在用 TTmCCRa 寄存器作为捕获寄存器进行脉宽测量时，每次检测到 INTTTEQCma 信号时，都需要通过软件读取捕获寄存器并计算间隔时间。



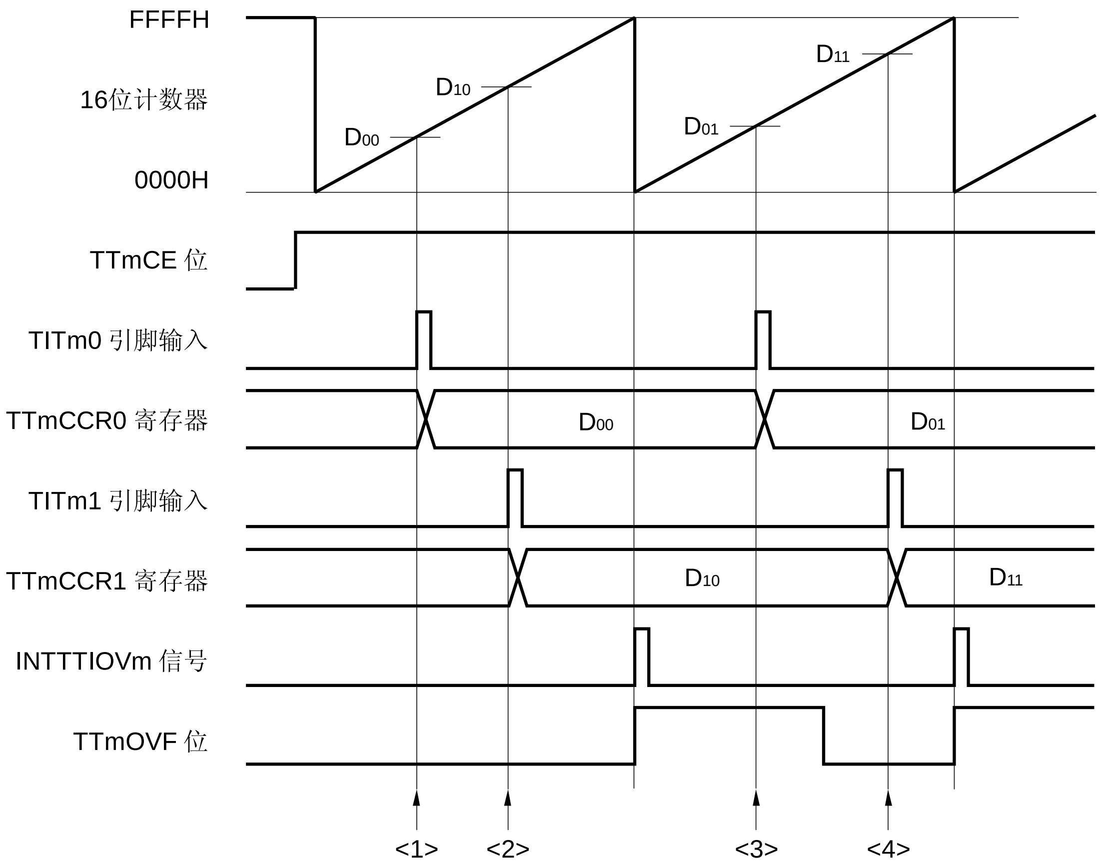
在自由运行定时器模式下执行脉宽测量时，可以用一个通道测量两个脉冲宽度。
要测量脉冲宽度，可以与 INTTTEQCma 信号同步读取 TTmCCRa 寄存器的值，并计算读取值与先前读取值之间的差，得到脉冲宽度。

备注 V850E/IF3：m = 1, a = 0, 1
V850E/IG3：m = 0, 1, a = 0, 1

（c） 当使用两个捕获寄存器时对溢出的处理

当使用两个捕获寄存器时必须要格外小心。首先是一个不正确处理的示例，如下所示。

使用两个捕获寄存器时的错误处理示例



在自由运行定时器模式下测量两个脉冲宽度时，可能出现下列问题。

<1> 读取 TTmCCR0 寄存器（TITm0 引脚输入的默认值设置）。

<2> 读取 TTmCCR1 寄存器（TITm1 引脚输入的默认值设置）。

<3> 读取 TTmCCR0 寄存器。
读溢出标志。若溢出标志为 1，将其清为 0。
因为溢出标志为 1，所以脉冲宽度可以用式（10000H ＋D_{01} – D_{00}）计算。

<4> 读取 TTmCCR1 寄存器。
读溢出标志。因为此标志在<3>中被清零，所以，读取到 0。
因为溢出标志为 0，所以脉冲宽度可以用式（D_{11} – D_{10}）计算。 （不正确）

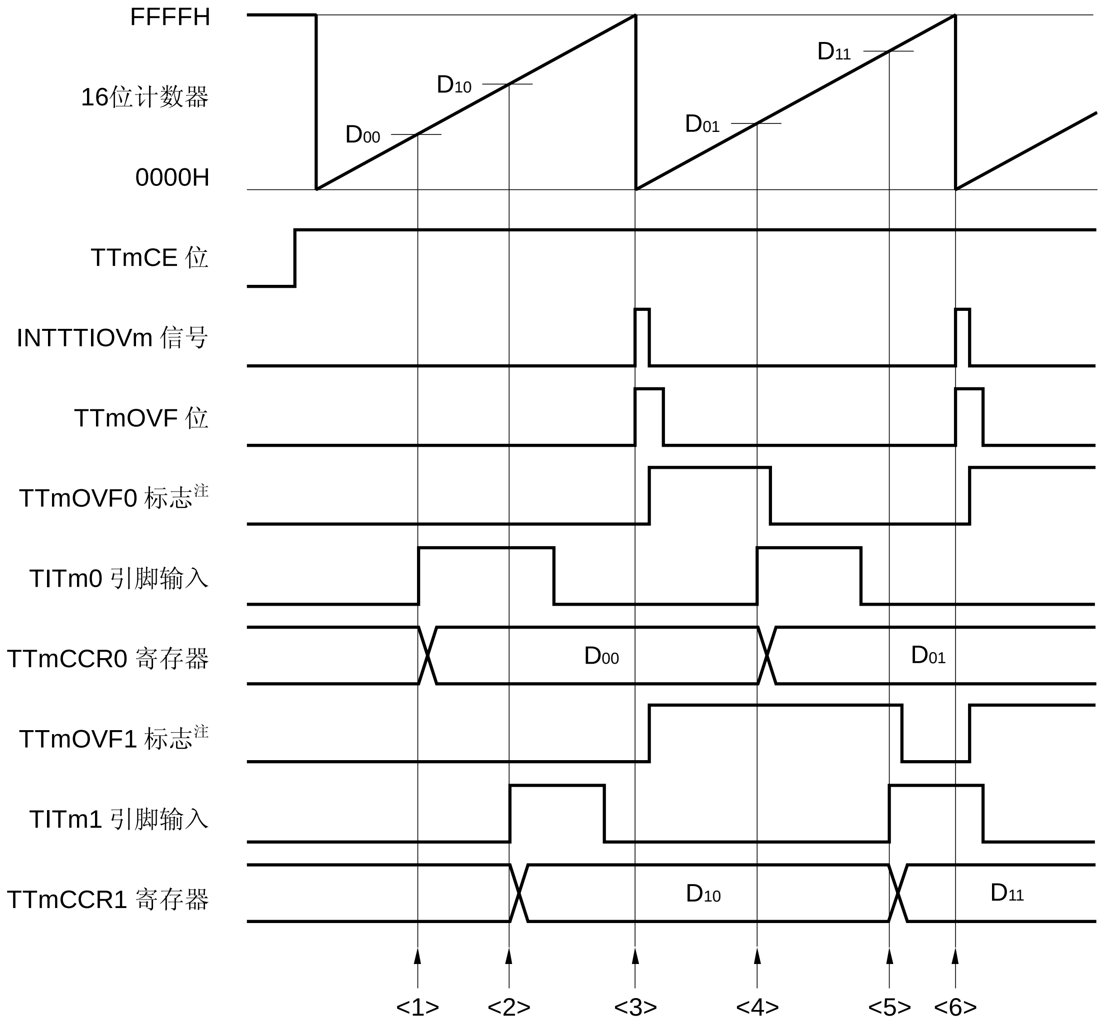
备注　V850E/IF3：m = 1
　　　V850E/IG3：m = 0, 1

在使用两个捕获寄存器时，如果溢出标志由一个捕获寄存器清为 0，则另一个捕获寄存器可能得不到正确的脉冲宽度。

在使用两个捕获寄存器时，使用软件解决。关于如何使用软件的示例如下。

（1/2）

使用两个捕获寄存器时的示例（使用溢出中断）



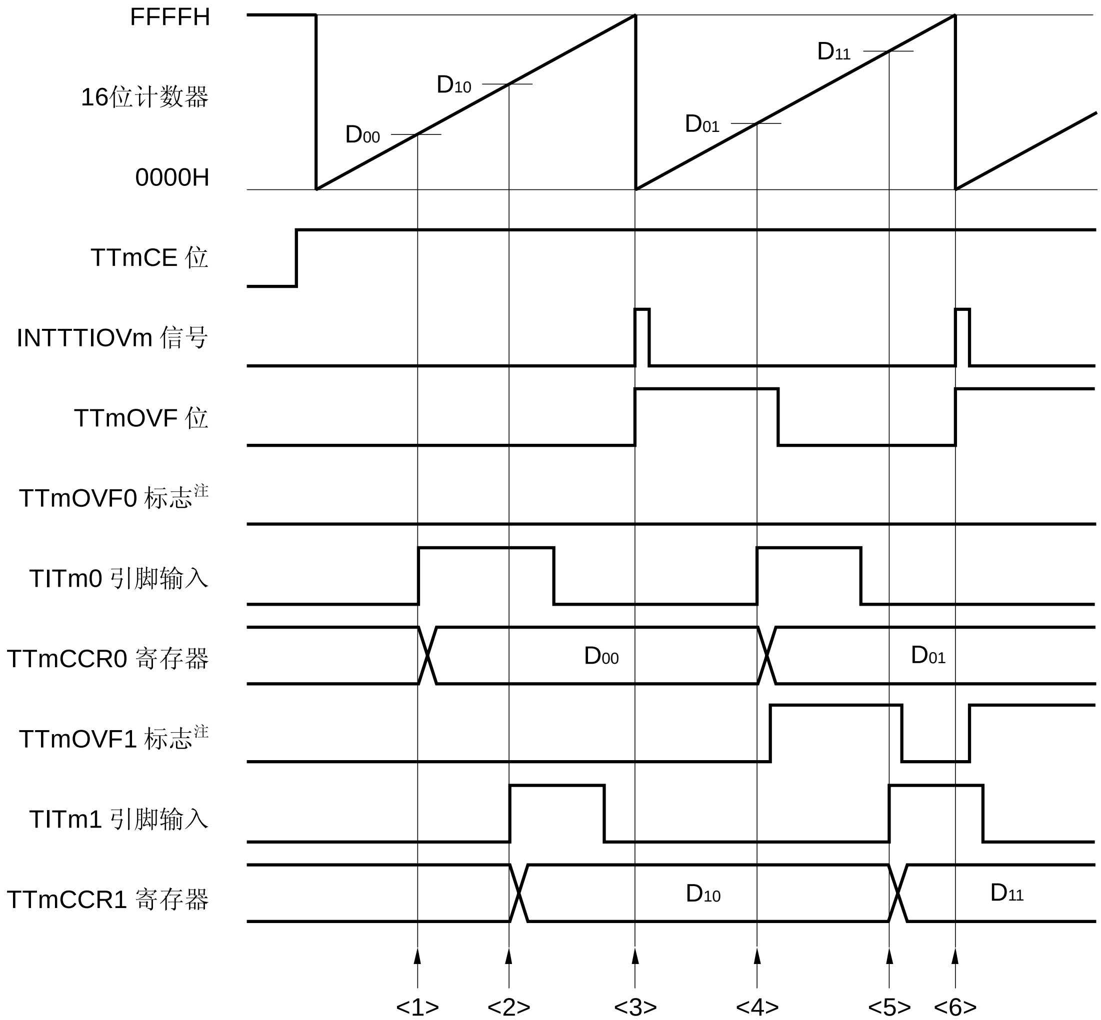
注　由软件对内部 RAM 中的 TTmOVF0 和 TTmOVF1 标志进行设置。

<1> 读取 TTmCCR0 寄存器（TITm0 引脚输入的默认值设置）。
<2> 读取 TTmCCR1 寄存器（TITm1 引脚输入的默认值设置）。
<3> 发生溢出，在溢出中断服务程序中将 TTmOVF0 和 TTmOVF1 标志设置为 1，并将溢出标志清为 0。
<4> 读取 TTmCCR0 寄存器。
读取 TTmOVF0 标志。如果 TTmOVF0 标志为 1，则将其清为 0。
因为 TTmOVF0 标志为 1，所以脉冲宽度可以用式（10000H ＋D_{01} − D_{00}）计算。
<5> 读取 TTmCCR1 寄存器。
读取 TTmOVF1 标志。如果 TTmOVF1 标志为 1，则将其清为 0（TTmOVF0 标志在<4>中已清零，而 TTmOVF1 标志仍然保持为 1）。
因为 TTmOVF1 标志为 1，所以脉冲宽度可以用式（10000H ＋D_{11} − D_{10}）计算（正确）。
<6> 重复 <3> ……

备注　V850E/IF3：m = 1
V850E/IG3：m = 0, 1

（2/2）

使用两个捕获寄存器时的示例（不使用溢出中断）



注　　由软件对内部 RAM 中的 TTmOVF0 和 TTmOVF1 标志进行设置。

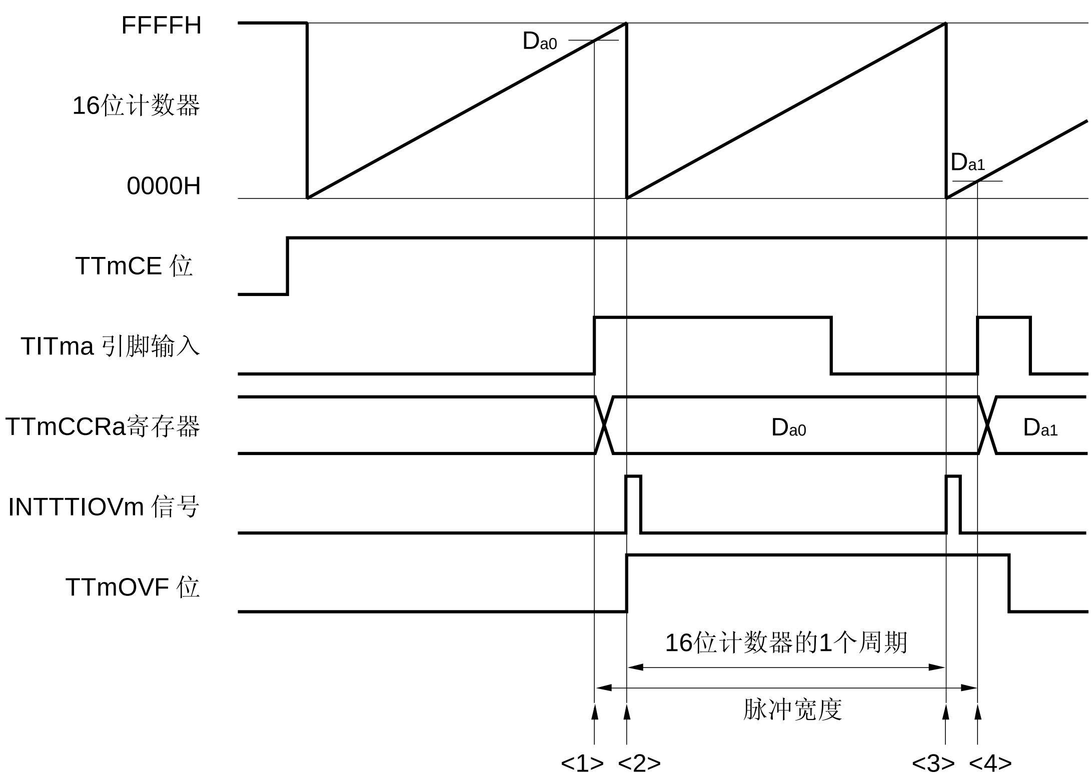
<1>　读取 TTmCCR0 寄存器（TITm0 引脚输入的默认值设置）。
<2>　读取 TTmCCR1 寄存器（TITm1 引脚输入的默认值设置）。
<3>　发生溢出，软件不作任何处理。
<4>　读取 TTmCCR0 寄存器。
　　读溢出标志。若溢出标志为 1，则只把 TTmOVF1 标志置为 1，并把溢出标志清为 0。
　　因为溢出标志为 1，所以脉冲宽度可以用式（10000H ＋D_{01} – D_{00}）计算。
<5>　读取 TTmCCR1 寄存器。
　　读溢出标志。因为溢出标志在<4>中被清零，所以读取 0。
　　读取 TTmOVF1 标志。如果 TTmOVF1 标志为 1，则将其清为 0。
　　因为 TTmOVF1 标志为 1，所以脉冲宽度可以用式（10000H ＋D_{11} – D_{10}）计算。（正确）
<6>　重复 <3> ……

备注　　V850E/IF3：m = 1
　　　　V850E/IG3：m = 0, 1

（d）当捕获触发间隔较长时的溢出处理

如果脉冲宽度大于 16 位计数器的一个计数循环周期，则必须加以注意，因为从第一个捕获触发信号到下一个触发信号之间可能发生多次溢出。首先是一个不正确处理的示例，如下所示，。

捕获触发间隔较长时的不正确处理示例



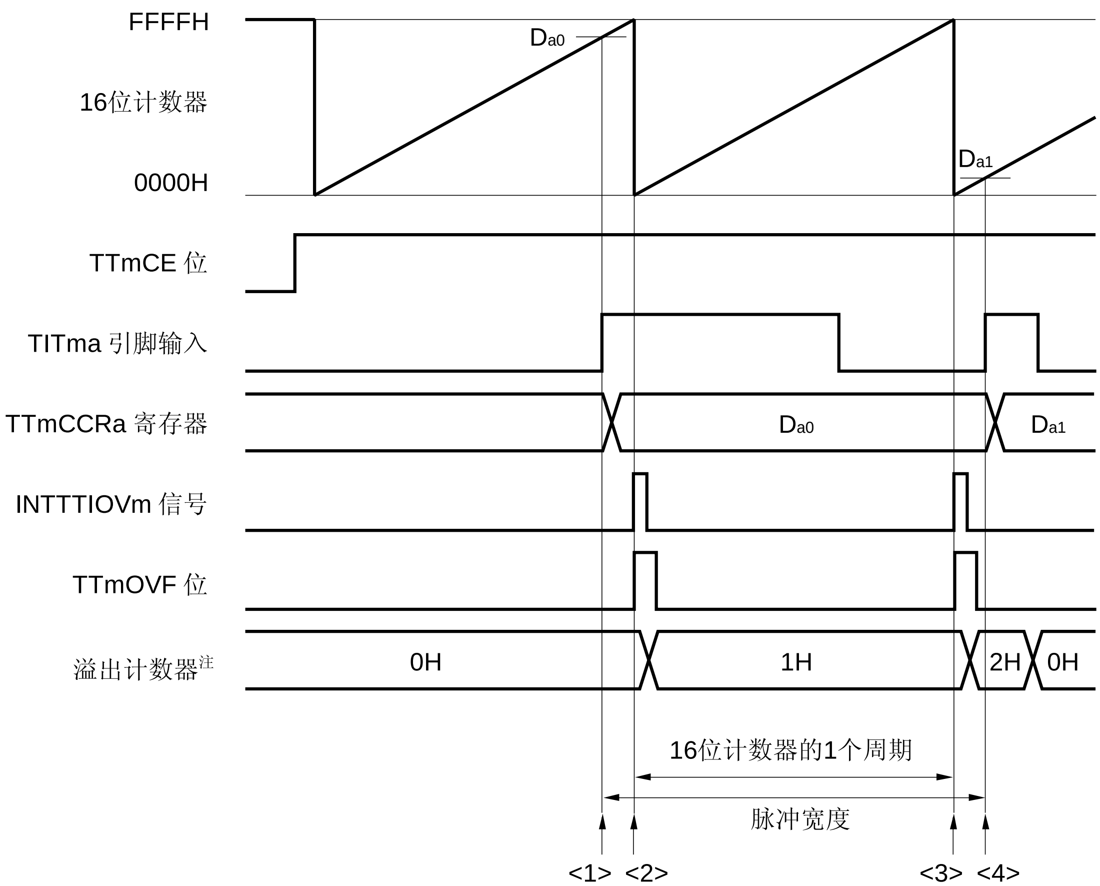
在自由运行定时器模式下测量长脉冲的宽度时，如果脉冲宽度较大，可能出现下列问题。

<1> 读取 TTmCCRa 寄存器（TITma 引脚输入的默认值设置）。
<2> 发生溢出，软件不作任何处理。
<3> 再次发生溢出。软件不作任何处理。
<4> 读取 TTmCCRa 寄存器。
读溢出标志。若溢出标志为 1，将其清为 0。
因为溢出标志为 1，所以脉冲宽度可以用式（10000H ＋D_{a1} – D_{a0}）计算。（不正确）
事实上，脉冲宽度应为（20000H ＋D_{a1} – D_{a0}），因为实际已经发生两次溢出。

备注　V850E/IF3：m = 1, a = 0, 1
V850E/IG3：m = 0, 1, a = 0, 1

如果在捕获触发间隔期间，发生两次或更多次的溢出，则无法获得正确的脉冲宽度。
如果捕获触发间隔时间较长时，应该减慢计数时钟以延长 16 位计数器的循环周期，或者使用软件来解决。
关于如何使用软件，示例如下。

捕获触发间隔较长时的示例



注　内部 RAM 中的溢出计数器由软件任意设置。

<1> 读取 TTmCCRa 寄存器（TITma 引脚输入的默认值设置）。
<2> 发生溢出，在溢出中断服务程序中增加溢出计数器的值，并将溢出标志清为 0。
<3> 再次发生溢出。在溢出中断服务程序中增加溢出计数器的值，并把溢出标志清为 0。
<4> 读取 TTmCCRa 寄存器。
读取溢出计数器。
→ 当溢出计数器为“N”时，脉冲宽度可用式（N × 10000H ＋D_{a1} – D_{a0}）计算。
在本例中，脉冲宽度为（20000H ＋D_{a1} – D_{a0}），因为溢出实际发生了两次。
溢出计数器清零（0H）。

备注　V850E/IF3：m = 1, a = 0, 1
V850E/IG3：m = 0, 1, a = 0, 1

（e） 清除溢出标志

当 TTmOVF 位为 1 时读取该位之后，通过 CLR 指令将 TTmOVF 位清为 0，且当 TTmOVF 位为 1 时读取该位之后，将 8 位数据（第 0 位是 0）写入 TTmOPT0 寄存器，可以将溢出标志清 0。

8.6.7 脉冲宽度测量模式 （TTmMD3 至 TTmMD0 位＝0110）

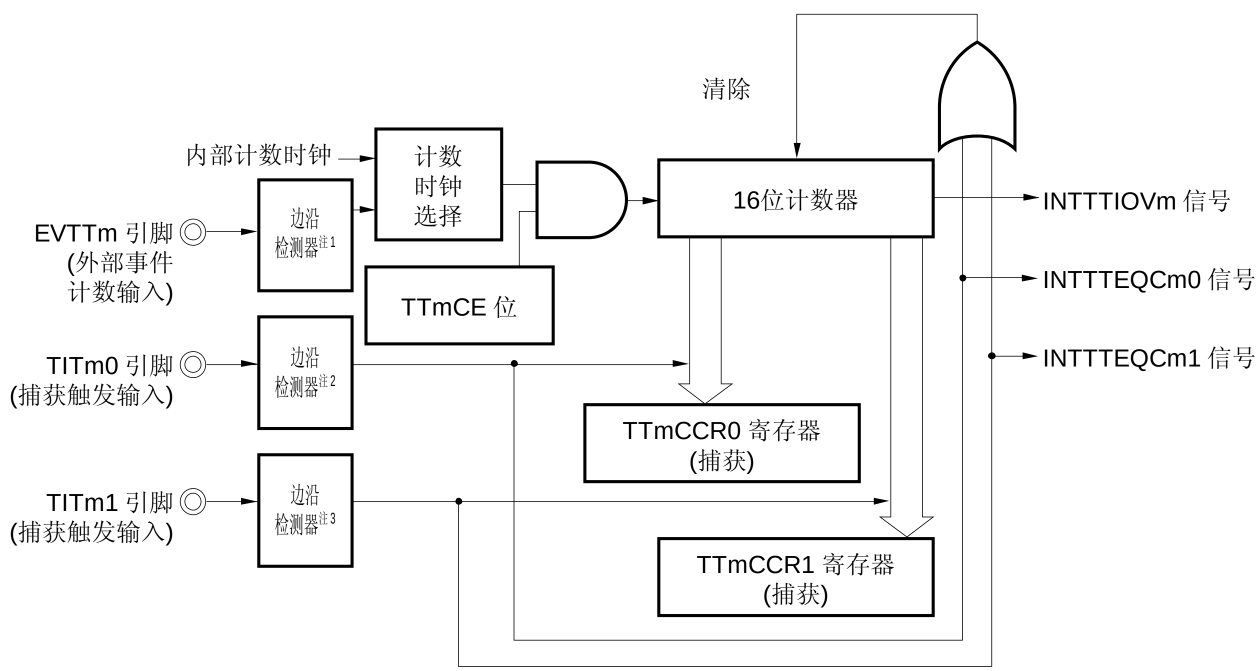
只有在 TMT0(仅限 V850E/IG3)和 TMT1 中，该模式下有效。

在脉冲宽度测量模式下，当 TTmCTL0.TTmCE 位被置为 1 时，16 位计时器/事件计数器 T 开始计数。每次检测到 TITma 引脚输入信号的有效边沿时，16 位计数器的计数值就保存到 TTmCCRa 寄存器中，且 16 位计数器清零为 0000H。

在发生捕获中断请求信号（INTTTEQCma）之后，可以通过读取 TTmCCRa 寄存器来测量有效边沿的间隔。

如图 8-40 所示，选择 TITm0 或 TITm1 引脚作为捕获触发信号输入引脚。使用 TTmIOC1 寄存器将未使用的引脚指定为“无边沿检测”。

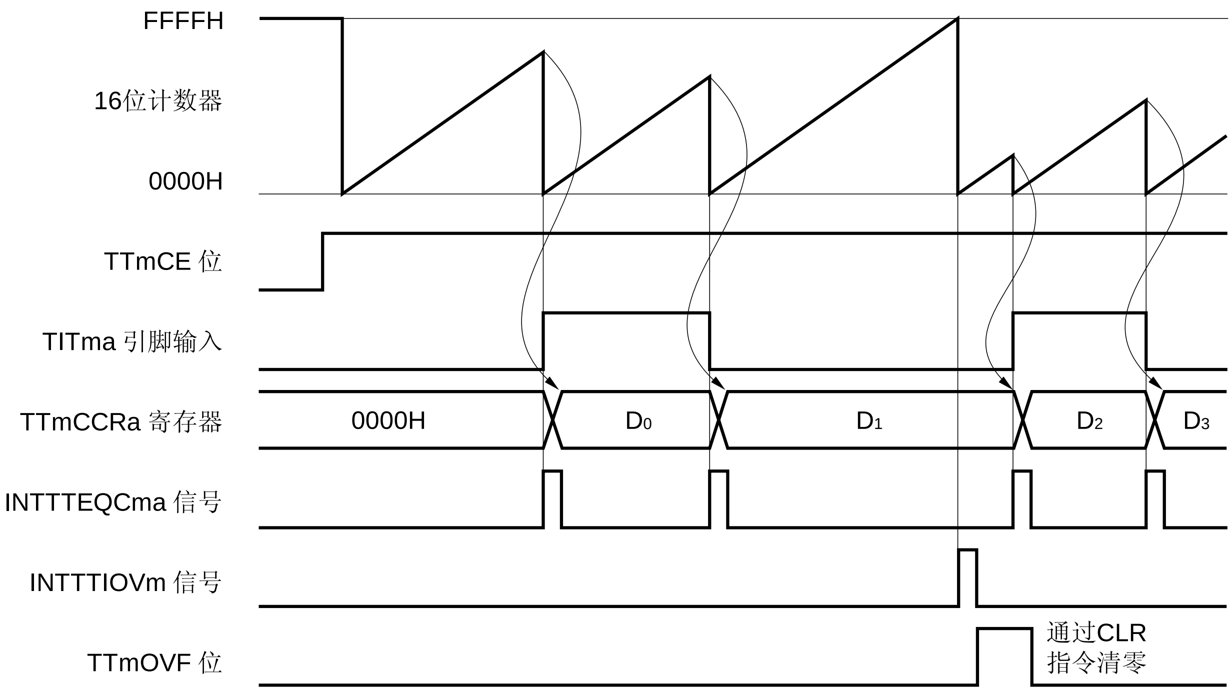
图 8-39. 脉冲宽度测量模式的配置



注 1. 通过 TTmIOC2.TTmEES1 和 TTmIOC2.TTmEES0 位设置。
2. 通过 TTmIOC1.TTmIS1 和 TTmIOC1.TTmIS0 位设置。
3. 通过 TTmIOC1.TTmIS3 和 TTmIOC1.TTmIS2 位设置。

备注 V850E/IF3：m = 1, a = 0, 1
V850E/IG3：m = 0, 1, a = 0, 1

图 8-40. 脉冲宽度测量模式的基本时序



备注 V850E/IF3：m = 1, a = 0, 1
V850E/IG3：m = 0, 1, a = 0, 1

当 TTmCE 位被置为 1 时，16 位计数器开始计数。当检测到 TITma 引脚输入信号的有效边沿时，16 位计数器的计数值被存储到 TTmCCRa 寄存器中，16 位计数器清零为 0000H，并产生一个捕获中断请求信号（INTTTEQCma）。

脉冲宽度的计算如下。

脉冲宽度＝捕获值 × 计数时钟周期

当 16 位计数器计数到 FFFFH 时，即使没有有效边沿输入 TITma 引脚，在下一个计数时钟时，仍然会产生溢出中断请求信号（INTTTIOVm），且计数器清零为 0000H，然后计数器继续计数。此时，溢出标志（TTmOPT0.TTmOVF 位）也被置为 1。使用软件执行 CLR 指令将溢出标志清为 0。

若溢出标志被设置为 1，则脉冲宽度的计算如下。

脉冲宽度＝（10000H × TTmOVF 位置位（1）的次数 ＋捕获值）× 计数时钟周期

备注 V850E/IF3：m = 1, a = 0, 1
V850E/IG3：m = 0, 1, a = 0, 1

图 8-41. 脉冲宽度测量模式时的寄存器设置（1/2）

（a） TMTm 控制寄存器 0 （TTmCTL0）

	TTmCE					TTmCKS2	TTmCKS1	TTmCKS0
TTmCTL0	0/1	0	0	0	0	0/1	0/1	0/1

TTmCKS2 至 TTmCKS0：选择计数时钟[注]

TTmCE：
0: 停止计数
1: 使能计数

注　　当 TTmCTL1.TTmEEE 位 = 1 时该设置无效。

（b） TMTm 控制寄存器 1（TTmCTL1）

		TTmEST	TTmEEE		TTmMD3	TTmMD2	TTmMD1	TTmMD0
TTmCTL1	0	0	0/1	0	0	1	1	0

TTmMD3 至 TTmMD0：
0, 1, 1, 0:
脉冲宽度测量模式

TTmEEE：
0: 计数时钟由TTmCKS0 至 TTmCKS2 位选择
1: 用外部事件计数输入信号计数

（c） TMTm I/O 控制寄存器 1 （TTmIOC1）

					TTmIS3	TTmIS2	TTmIS1	TTmIS0
TTmIOC1	0	0	0	0	0/1	0/1	0/1	0/1

TTmIS1、TTmIS0：选择 TITm0 引脚输入的有效边沿

TTmIS3、TTmIS2：选择 TITm1 引脚输入的有效边沿

（d） TMTm I/O 控制寄存器 2 （TTmIOC2）

					TTmEES1	TTmEES0	TTmETS1	TTmETS0
TTmIOC2	0	0	0	0	0/1	0/1	0	0

TTmEES1、TTmEES0：选择外部事件计数输入 (EVTTm 引脚)的有效边沿

图 8-41. 脉冲宽度测量模式时的寄存器设置（2/2）

（e） TMTm 选项寄存器 0（TTmOPT0）

			TTmCCS1	TTmCCS0				TTmOVF
TTmOPT0	0	0	0	0	0	0	0	0/1

TTmOVF：溢出标志

（f） TMTm 计数器读取缓冲寄存器 （TTmCNT）

通过读取 TTmCNT 寄存器，可以获得 16 位计数器的值。

（g） TMTm 捕获/比较寄存器 0 和 1（TTmCCR0 和 TTmCCR1）

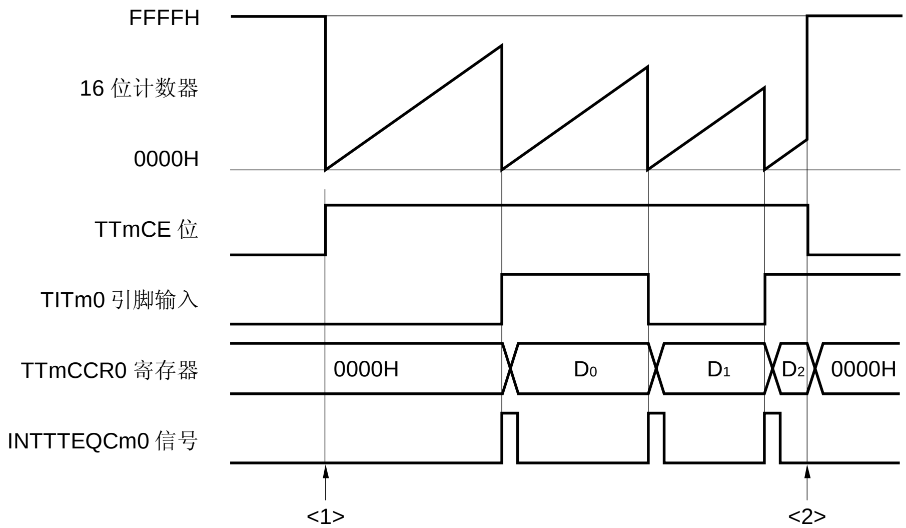
当检测到 TITm0 和 TITm1 引脚输入的有效边沿时，这些寄存器保存 16 位计数器的当前计数值。

备注　**1.** 脉冲宽度测量模式不使用 TMTm 控制寄存器 2（TTmCTL2）、TMTm I/O 控制寄存器 0（TTmIOC0）、TMTm I/O 控制寄存器 3（TTmIOC3）、TMTm 选项寄存器 1（TTmOPT1）、TMTm 捕获输入选择寄存器(TTISLm)和 TMTm 计数器写入寄存器（TTmTCW）。

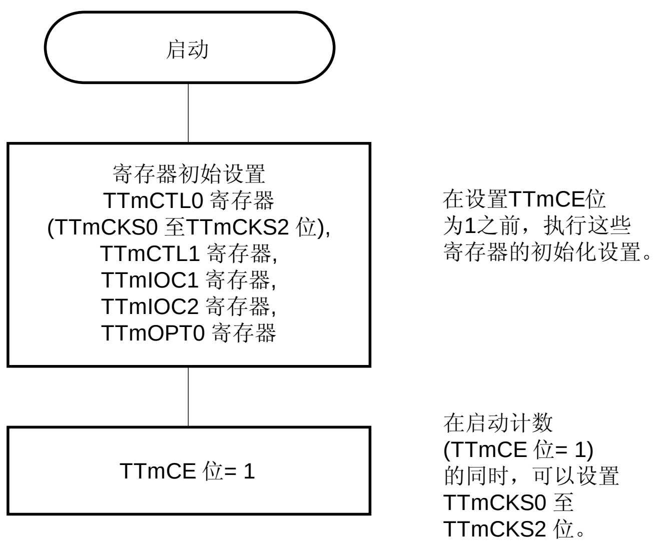
2. V850E/IF3：m = 1
V850E/IG3：m = 0, 1

（1）脉冲宽度测量模式时的操作流程

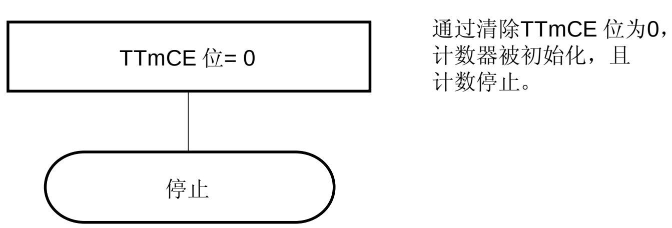
图 8-42. 脉冲宽度测量模式的软件处理流程



<1> 计数操作启动流程



<2> 计数操作停止流程



备注　V850E/IF3：m = 1

V850E/IG3：m = 0, 1

（2）脉冲宽度测量模式时的操作时序

（a） 清除溢出标志

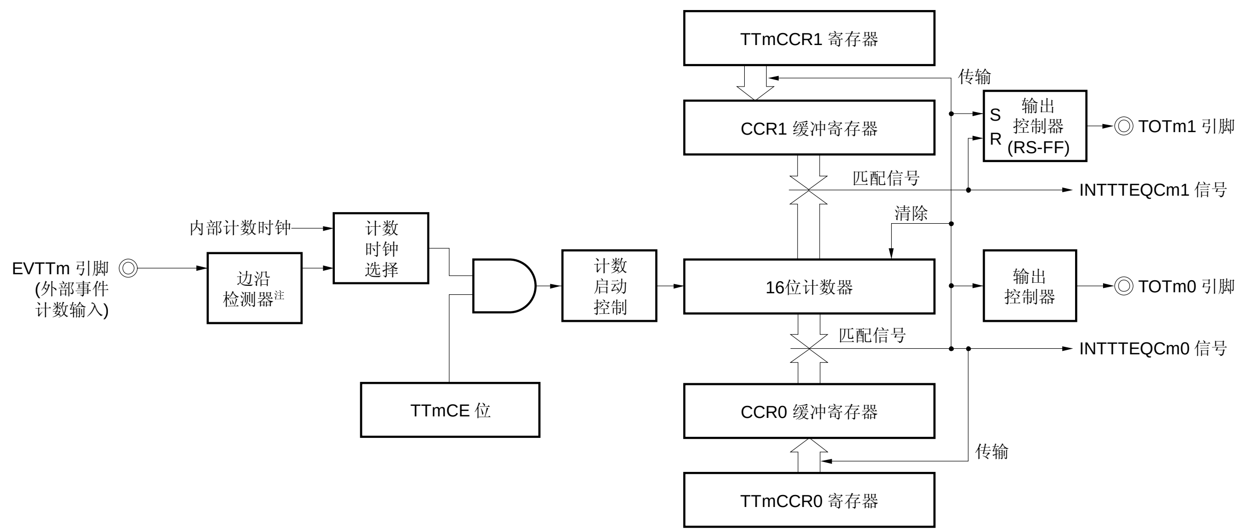
当 TTmOVF 位为 1 时读取该位之后，通过 CLR 指令将 TTmOVF 位清为 0，且当 TTmOVF 位为 1 时读取该位之后，将 8 位数据（第 0 位是 0）写入 TTmOPT0 寄存器，可以将溢出标志清 0。

8.6.8　三角波 PWM 输出模式（TTmMD3 至 TTmMD0 位 = 0111）

在三角波 PWM 输出模式下，当 TTmCTL0.TTmCE 位被置为 1 时，从 TOTm1 引脚输出一个三角波 PWM 波形。

当 16 位计数器的计数值和 CCR0 缓冲寄存器的值一致，且 16 位计数器被置为 0000H 时，从 TOTm1 引脚输出一个反相的 PWM 波形。

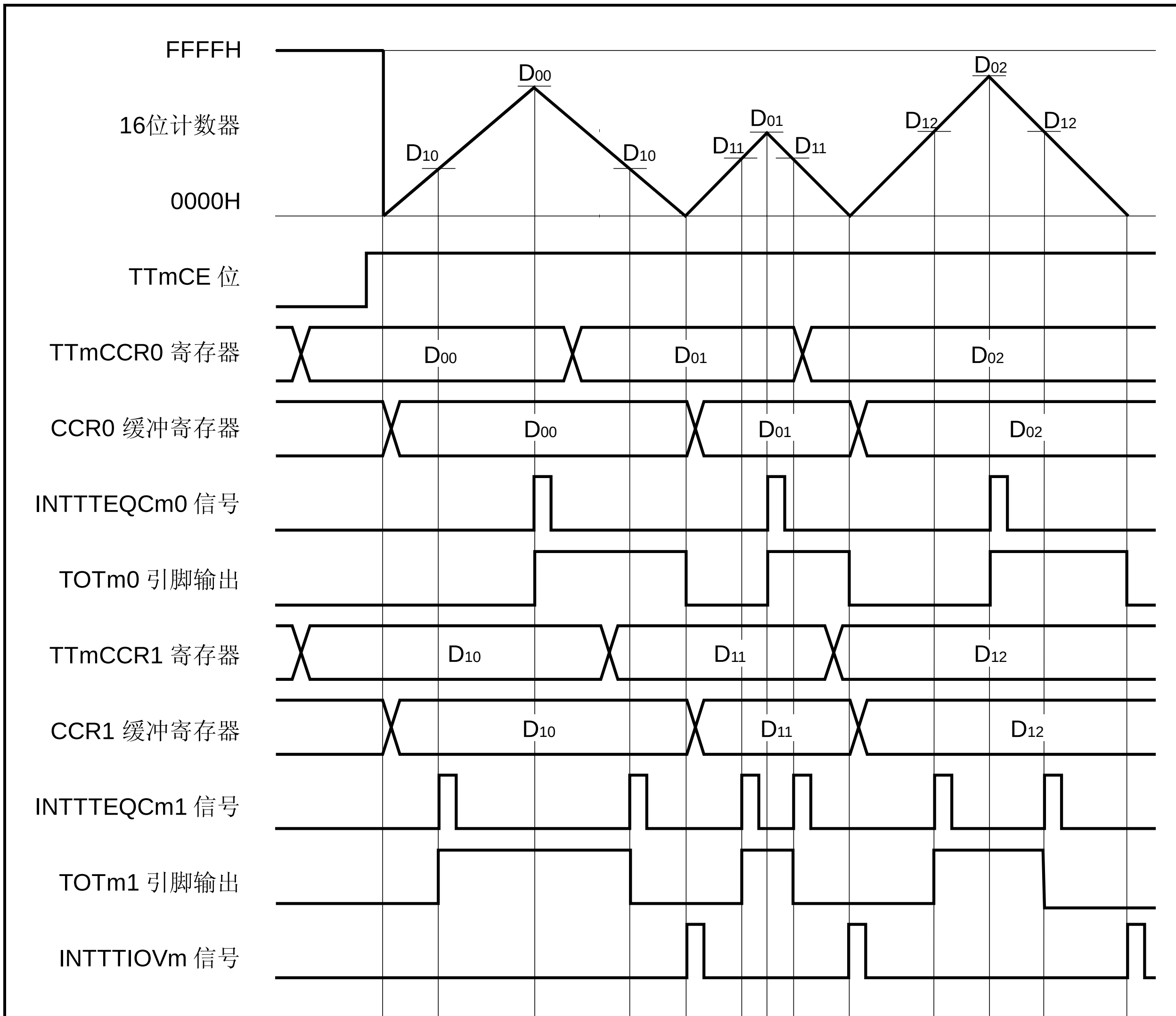
图 8-43. 三角波 PWM 输出模式的配置



注　　通过 TTmIOC2.TTmEES1 和 TTmIOC2.TTmEES0 位设置。

备注　　V850E/IF3：m = 1
　　　　V850E/IG3：m = 0, 1

图 8-44. 三角波 **PWM** 输出模式的基本时序



备注　**1.** V850E/IF3：m = 1

V850E/IG3：m = 0, 1

2. 当 TTmIOC0 寄存器 = 05H 时的时序图。

当 TTmCE 位被置为 1 时，16 位计数器被从 FFFFH 清除为 0000H，且启动计数。三角 PWM 波形从 TOTm1 引脚输出。

在三角波 PWM 输出模式下，计数器向上或向下计数。当向下计数时 16 位计数器达到 0000H，产生一个溢出中断请求信号（INTTTIOVm）。此时 TTmOPT0.TTmOVF 位不被置为 1。当计数器向上计数时，如果 16 位计数器的计数值和 CCR0 缓冲寄存器的值一致，产生一个比较一致中断请求信号（INTTTEQCm0）。

当 16 位计数器的值和 CCR0 缓冲寄存器的值一致时，计数方向从向上改为向下计数，且当计数器被清除为 0000H 时，计数方向从向下改为向上计数。

在操作期间，通过重写 TTmCCRa 寄存器，可以改变 PWM 波形。要在操作期间改变 PWM 波形，最后写入 TTmCCR1 寄存器。

三角 PWM 波形的周期由 TTmCCR0 寄存器设置，占空比由 TTmCCR1 寄存器设置。TTmCCR0 寄存器的设置值范围是 “0 ≤ TTmCCR0 ≤ FFFEH”。向下计数时，当 16 位计数器达到 0000H，重写值被反映出来。

即使只需要改变 PWM 波形的周期，应先将周期设置到 TTmCCR0 寄存器，然后将相同的值（和 TTmCCR1 寄存器的当前值相等）写入 TTmCCR1 寄存器。

要将数据从 TTmCCRa 寄存器传输到 CCRa 缓冲寄存器，必须将数据写入 TTmCCR1 寄存器（a = 0，1）。

<R> **(1) 0%/100%的 PWM 输出**

在三角波 PWM 输出模式下，PWM 输出可以有 0% 波形输出和 100%波形输出。

当 TTmCCR0 寄存器 = M 时，将 TTmCCR1 寄存器设置为“M + 1”，可以输出 0% 波形。

将 TTmCCR1 寄存器设置为“0000H”，可以输出 100% 波形。

TOTm0 和 TOTm1 的输出电平可以由 TTmIOC0 寄存器设置。

备注 V850E/IF3：m = 1

V850E/IG3：m = 0, 1

图 8-45. 0% PWM 输出波形(TTmIOC0 寄存器 = 05H)

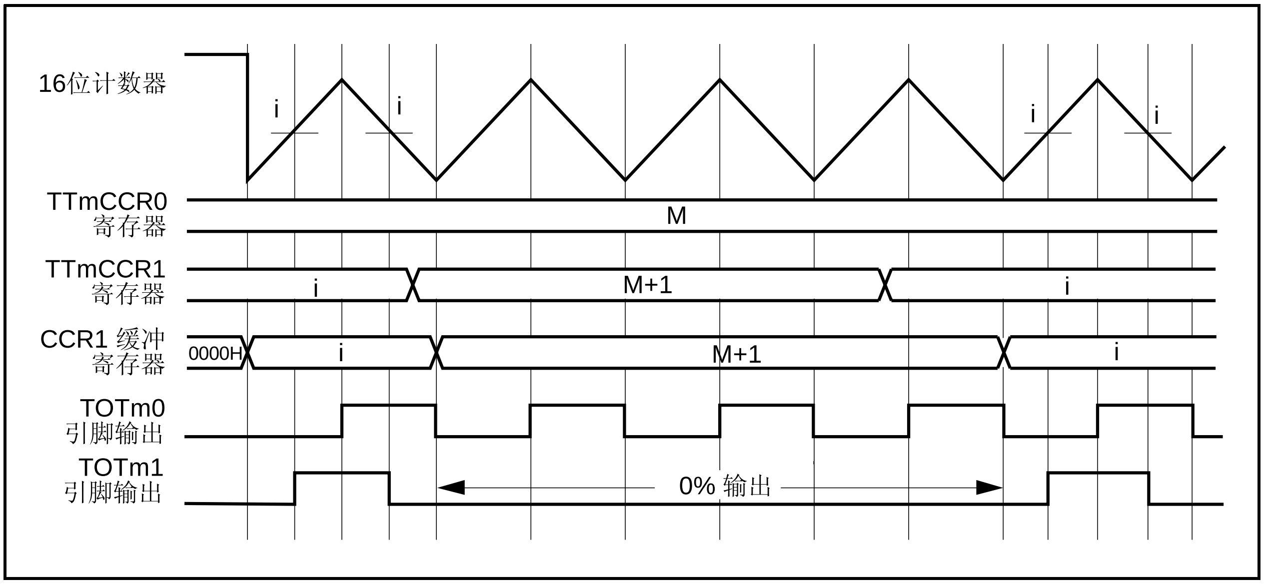
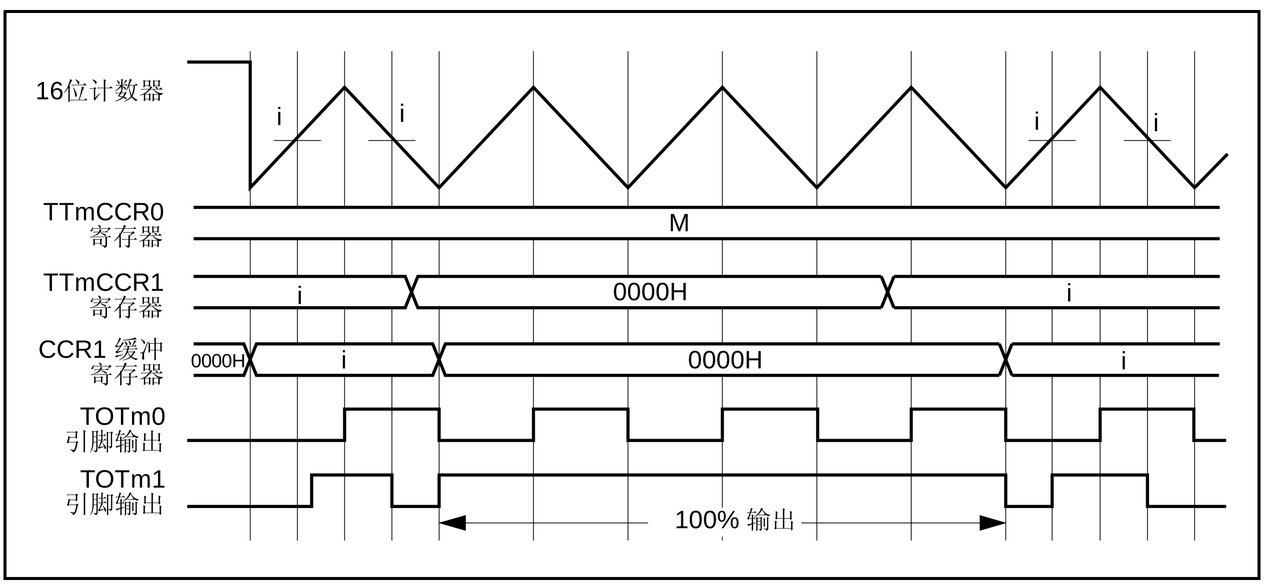


图 8-46. 100% PWM 输出波形(TTmIOC0 寄存器 = 05H)



8.6.9 编码器计数功能

编码器计数功能包括编码器比较模式（参见 **8.6.10 编码器比较模式 （TTmMD3 至 TTmMD0 位 = 1000）**）.

模式	TTmCCR0 寄存器	TTmCCR1 寄存器
编码器比较模式	仅作比较器	仅作比较器

（1）向上/向下计数控制

通过 16 位计数器向上或向下计数，由输入编码器信号的相（TENCm0 和 TENCm1）以及 TTmCTL2.TTmUDS1 和 TTmCTL2.TTmUDS0 位的设置来控制。

使用编码器计数功能时，不能使用内部计数时钟和外部事件计数输入(EVTTm)。设置 TTmCTL0.TTmCKS2 至 TTmCTL0.TTmCKS0 位为 000 且 TTmCTL1.TTmEEE 位为 0。

（2）16 位计数器的初始值设置

在计数器启动操作（TTmCTL0.TTmCE 位 = 0 → 1）之后，当 TTmCTL2.TTmECC 位 = 0 时，设置在 TTmTCW 寄存器的初始计数值立即被传输到 16 位计数器。在检测到编码器输入信号（TENCm0 或 TENCm1）的有效边沿后，计数器启动操作。

（3）基本操作

当 16 位计数器的计数值和 CCRa 缓冲寄存器的值一致时，TTmCCRa 寄存器产生一个比较一致中断请求信号（INTTTEQCma）。

（4）清除操作

在编码器比较模式下满足下列条件时，16 位计数器被清除。

- 当 16 位计数器的值和比较寄存器（TTmCTL2.TTmECM1 和 TTmCTL2.TTmECM0 位被置位）的值一致时。
- 当检测到编码器清除输入信号(TECRm)的边沿时(当 TTmIOC3.TTmSCE 位 = 0 时，TTmECS1 和 TTmECS0 位被置位)。
- 当检测到 TENCm0，TENCm1 和 TECRm 引脚的清除电平条件时(当 TTmSCE 位 = 1 时，TTmZCL, TTmBCL, 和 TTmACL 位被置位)。

备注　V850E/IF3：m = 1, a = 0, 1
　　　V850E/IG3：m = 0, 1, a = 0, 1

（5）TTmCTL2 寄存器的控制位

在编码器比较模式下，TTmCTL2 寄存器的设置如下。

表 8-10. TTmCTL2 寄存器的设置

模式	TTmUDS1, TTmUDS0 位（<1>）	TTmECM1 位（<2>）	TTmECM0 位（<2>）	TTmLDE 位（<3>）	计数器清除（目标比较寄存器）	传输到计数器
编码器比较模式	可以设置为 00, 01, 10, 或 11.	0	0	0	–	–
				1		可能
			1	0	TTmCCR0	–
				1		可能[注]
		1	0	无效	TTmCCR1	–
			1	无效	TTmCCR0, TTmCCR1	–

注　　计数器可以操作的范围是从 0000H 到 TTmCCR0 寄存器的设置值。

备注　　V850E/IF3：m = 1
　　　　V850E/IG3：m = 0, 1

（a） 各个位的概述

<1> TTmUDS1 和 TTmUDS0 位通过编码器输入引脚（TENCm0 或 TENCm1）的输入的相来识别 16 位计数器计数方向（向上或向下）。

<2> 当 16 位计数器的计数值和 CCR0 或 CCR1 缓冲寄存器的值一致时，TTmECM1 和 TTmECM0 位控制 16 位计数器的清除。

<3> 当计数器下溢时，TTmLDE 位控制将 TTmCCR0 寄存器的设置值传输给 16 位计数器的功能。只有当 TTmECM1 和 TTmECM0 位分别为 00 或 01 时，TTmLDE 位有效。当这些位被置为其他值，该位无效。

（b） 各个位的详细解释

<1> TTmUDS1 和 TTmUDS0 位：向上/向下计数选择

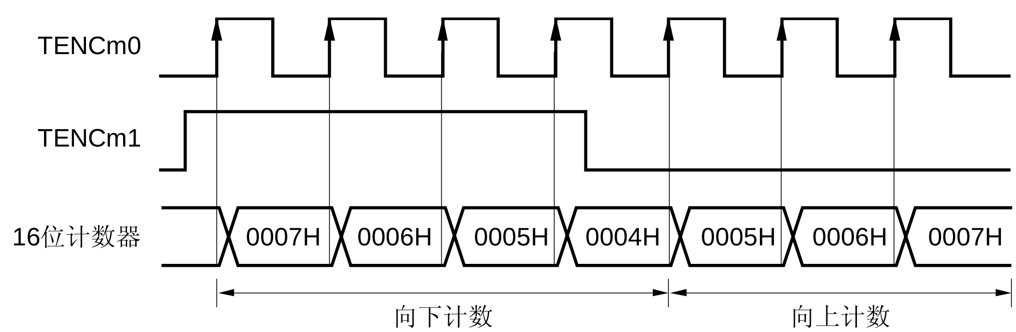
16 位计数器是向上计数或向下计数通过 TENCm0 或 TENCm1 引脚输入的相来识别，取决于 TTmUDS1 和 TTmUDS0 位的设置。只有在编码器比较模式下，这些位有效。

- 当 TTmUDS1 和 TTmUDS0 位 = 00

TENCm0 引脚	TENCm1 引脚	计数操作
上升沿	高电平	向下计数
下降沿		
双沿		
上升沿	低电平	向上计数
下降沿		
双沿		

备注　检测由 TTmIOC3.TTmEIS1 和 TTmEIS0 位指定的 TENCm0 引脚边沿。

图 8-47. 操作示例（当 **TENCm0** 引脚的有效边沿被指定为上升沿且 **TENCm1** 引脚的有效边沿被指定为无边沿检测）



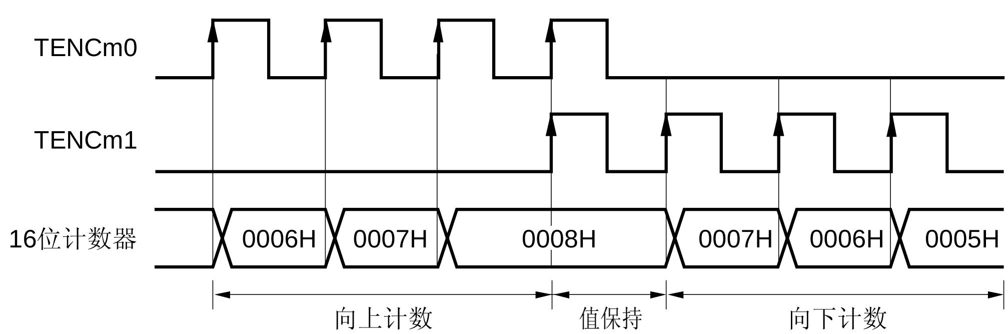
备注　V850E/IF3：m = 1
V850E/IG3：m = 0, 1

- 当 TTmUDS1 和 TTmUDS0 位 = 01

TENCm0 引脚	TENCm1 引脚	计数操作
低电平	上升沿	向下计数
	下降沿	
	双沿	
高电平	上升沿	
	下降沿	
	双沿	
上升沿	高电平	向上计数
下降沿		
双沿		
上升沿	低电平	
下降沿		
双沿		
同时输入到 TENCm0 和 TENCm1 引脚		计数器不执行计数操作，而是保持之前的瞬时值

备注　　检测由 TTmIOC3.TTmEIS1 和 TTmIOC3.TTmEIS0 位指定的 TENCm0 引脚的边沿。

图 8-48. 操作示例 （当 TENCm0 和 TENCm1 引脚的有效边沿被指定为上升沿时）



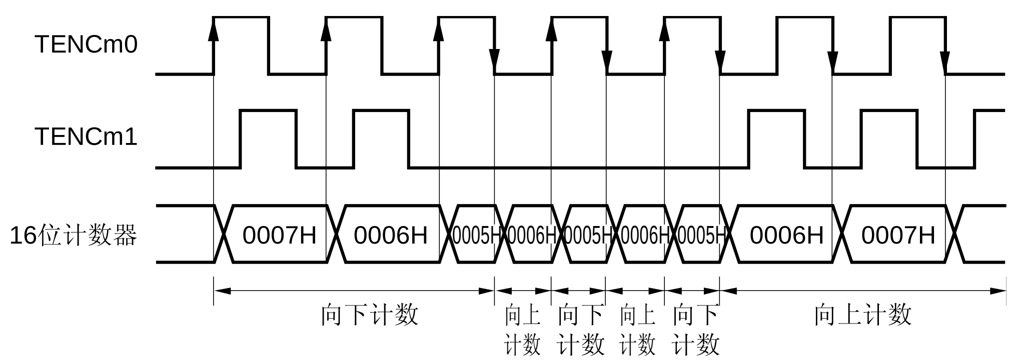
备注　　V850E/IF3：m = 1
V850E/IG3：m = 0, 1

- 当 TTmUDS1 和 TTmUDS0 位 = 10

<table>
<tr><th>TENCm0 引脚</th><th>TENCm1 引脚</th><th>计数操作</th></tr>
<tr><td>低电平</td><td>下降沿</td><td>计数器不执行计数操作，而是保持之前的瞬时值</td></tr>
<tr><td>上升沿</td><td>低电平</td><td>向下计数</td></tr>
<tr><td>高电平</td><td>上升沿</td><td rowspan="4">计数器不执行计数操作，而是保持之前的瞬时值</td></tr>
<tr><td>下降沿</td><td rowspan="2">高电平</td></tr>
<tr><td>上升沿</td></tr>
<tr><td>高电平</td><td>下降沿</td></tr>
<tr><td>下降沿</td><td>低电平</td><td>向上计数</td></tr>
<tr><td>低电平</td><td rowspan="3">上升沿</td><td rowspan="3">计数器不执行计数操作，而是保持之前的瞬时值</td></tr>
<tr><td>上升沿</td></tr>
<tr><td>下降沿</td></tr>
<tr><td>上升沿</td><td rowspan="2">下降沿</td><td>向下计数</td></tr>
<tr><td>下降沿</td><td>向上计数</td></tr>
</table>

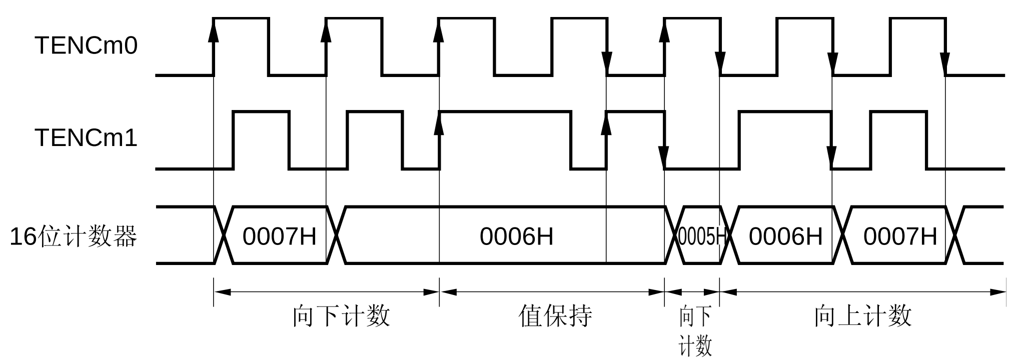
注意事项 TENCm0 和 **TENCm1** 引脚有效边沿的指定无效。

图 **8-49.** 操作示例 （当 **TENCm0** 和 **TENCm1** 引脚的有效边沿不相同时的计数操作）



备注 V850E/IF3：m = 1
V850E/IG3：m = 0, 1

图 **8-50.** 操作示例 （当 **TENCm0** 和 **TENCm1** 引脚的有效边沿相同时的计数操作）



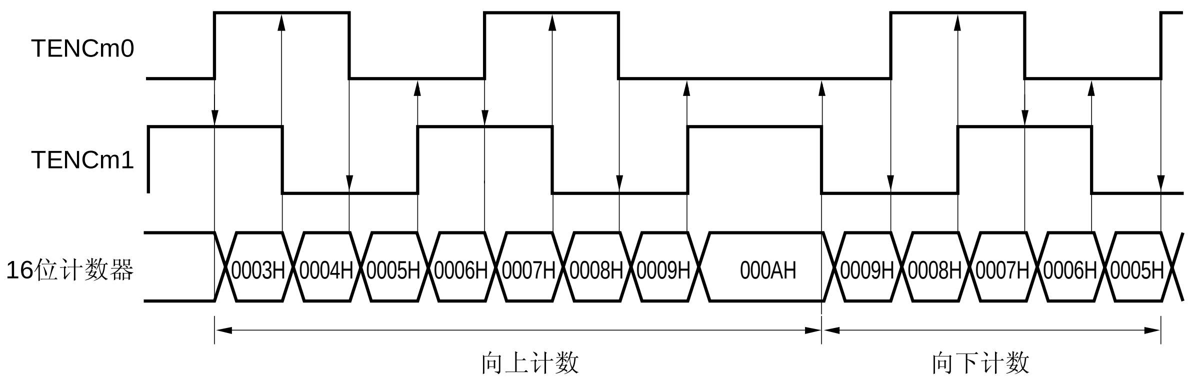
备注 V850E/IF3：m = 1
V850E/IG3：m = 0, 1

- 当 TTmUDS1 和 TTmUDS0 位 = 11

<table>
<tr><th>TENCm0 引脚</th><th>TENCm1 引脚</th><th>计数操作</th></tr>
<tr><td>低电平</td><td>下降沿</td><td rowspan="4">向下计数</td></tr>
<tr><td>上升沿</td><td>低电平</td></tr>
<tr><td>高电平</td><td>上升沿</td></tr>
<tr><td>下降沿</td><td rowspan="2">高电平</td></tr>
<tr><td>上升沿</td><td rowspan="4">向上计数</td></tr>
<tr><td>高电平</td><td>下降沿</td></tr>
<tr><td>下降沿</td><td>低电平</td></tr>
<tr><td>低电平</td><td>上升沿</td></tr>
<tr><td colspan="2">同时输入到 TENCm0 和 TENCm1 引脚</td><td>计数器不执行计数操作，而是保持之前的瞬时值</td></tr>
</table>

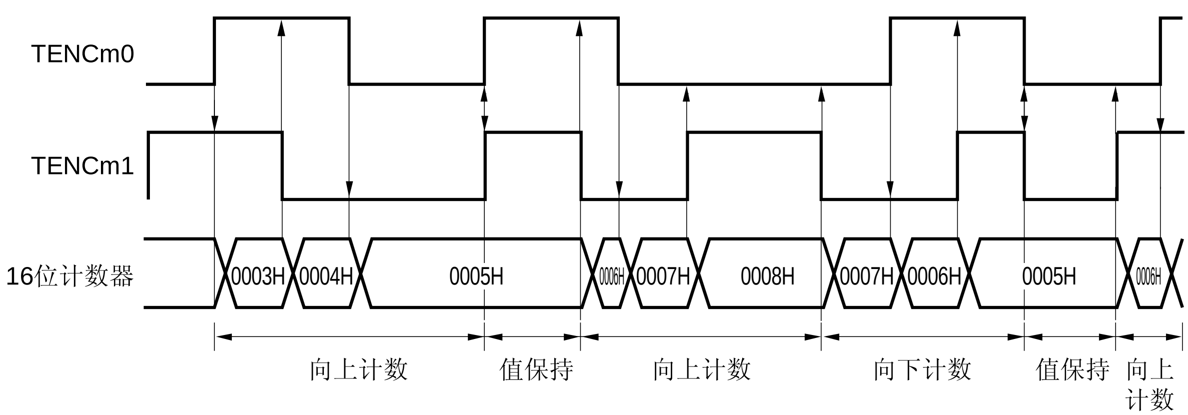
注意事项 **TENCm0** 和 **TENCm1** 引脚有效边沿的指定无效。

图 **8-51.** 操作示例 （当 **TENCm0** 和 **TENCm1** 引脚的有效边沿不相同时的计数操作）



备注 V850E/IF3：m = 1
V850E/IG3：m = 0, 1

图 **8-52.** 操作示例 （当 **TENCm0** 和 **TENCm1** 引脚的有效边沿相同时的计数操作）



备注 V850E/IF3：m = 1
V850E/IG3：m = 0, 1

<2> TTmECM1 和 TTmECM0 位： 比较寄存器一致时的定时器 /计数器清除功能

当计数器的计数值和 CCRa 缓冲寄存器的值一致时，16 位计数器根据 TTmECM1 和 TTmECM0 位的设置值执行计数操作。

- 当 TTmECM1 和 TTmECM0 位 = 00

  当 16-位计数器的计数值和 CCRa 缓冲寄存器的值一致时，不清除 16 位计数器。

- 当 TTmECM1 和 TTmECM0 位 = 01

  当 16-位计数器的计数值和 CCR0 缓冲寄存器的值一致时，16 位计数器在下列条件下执行一次计数操作。

下一次计数操作	描述
向上计数	16 位计数器被清除为 0000H
向下计数	16 位计数器的计数值向下计数

- 当 TTmECM1 和 TTmECM0 位 = 10

  当 16-位计数器的计数值和 CCR1 缓冲寄存器的值一致时，16 位计数器在下列条件下执行一次计数操作。

下一次计数操作	描述
向上计数	16 位计数器的计数值向上计数
向下计数	16 位计数器被清除为 0000H

- 当 TTmECM1 和 TTmECM0 位 = 11

  当 16-位计数器的计数值和 CCR0 缓冲寄存器的值一致时，16 位计数器在下列条件下执行一次计数操作。

下一次计数操作	描述
向上计数	16 位计数器被清除为 0000H
向下计数	16 位计数器的计数值向下计数

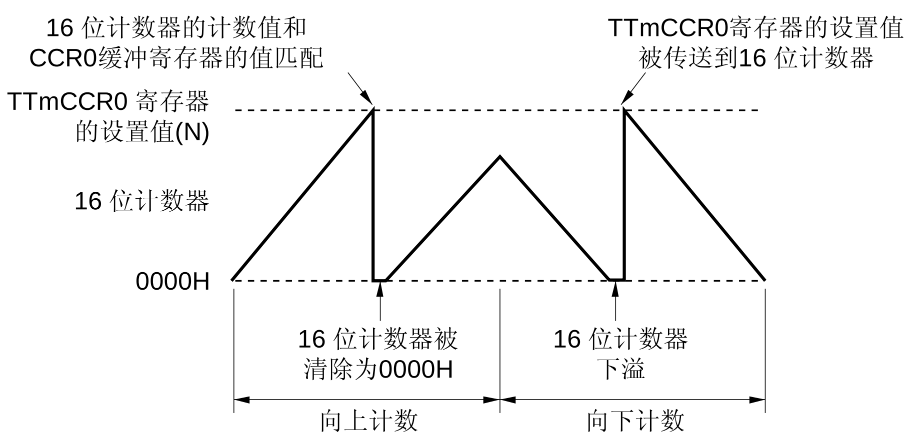
  当 16-位计数器的计数值和 CCR1 缓冲寄存器的值一致时，16 位计数器在下列条件下执行一次计数操作。

下一次 计数操作	描述
向上计数	16 位计数器的计数值向上计数
向下计数	16 位计数器被清除为 0000H

<3> TTmLDE 位： 当计数器下溢时，将 TTmCCR0 寄存器的设置值传输到 16 位计数器。
当 TTmLDE 位 = 1 时，发生计数器下溢，TTmCCR0 寄存器的设置值可以被传输到 16 位计数器。
只有在编码器比较模式下，TTmLDE 位才有效。

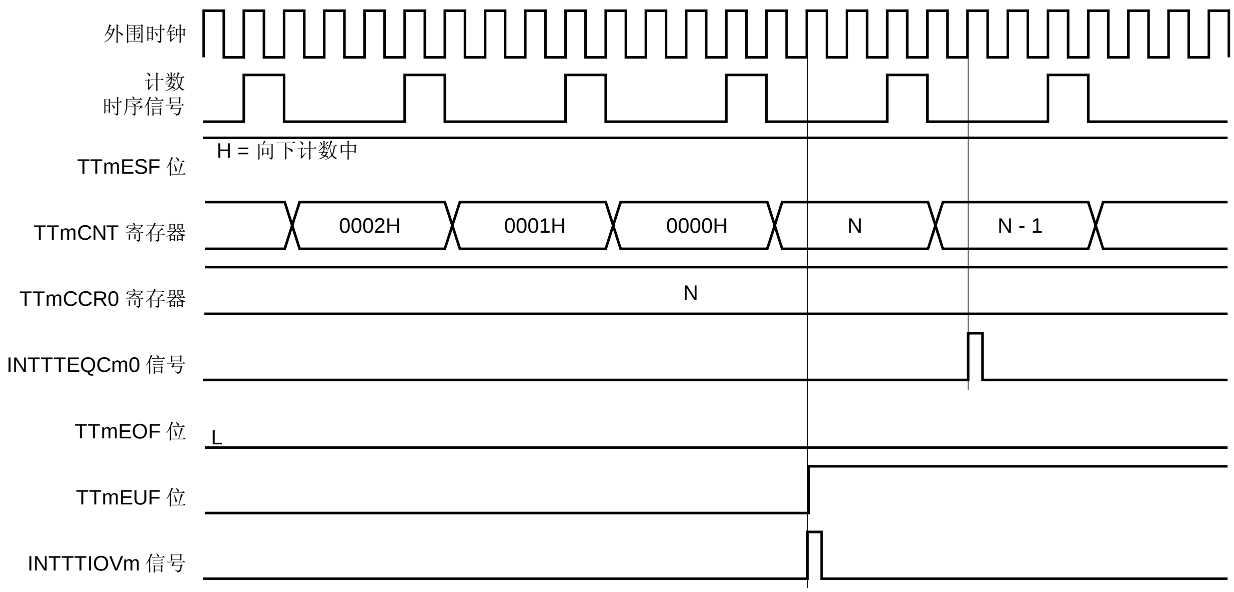
- 计数操作的范围是从 0000H 到 TTmCCR0 寄存器的设置值
  如果当 TTmLDE 位 = 1 且 TTmECM1 和 TTmECM0 位 = 01 时，16 位计数器执行了一次计数操作，或者当 TTmECM0 位 = 1 时，计数器的计数值和 CCR0 缓冲寄存器的值一致，如果下一次计数操作是向上计数，则 16 位计数器被清除为 0000H。
  如果当 TTmLDE 位 = 1 时，16 位计数器下溢，TTmCCR0 寄存器的设置值被传输到计数器。
  因此，计数器可以操作的范围是从 0000H 到 TTmCCR0 寄存器的设置值，其中计数值的上限的值是 TTmCCR0 寄存器的设置值，下限的值是 0000H。

图 8-53. 操作示例 （计数操作的范围是从 0000H 到 TTmCCR0 寄存器的设置值）



备注　V850E/IF3：m = 1
　　　V850E/IG3：m = 0, 1

图 8-54. 操作时序（计数操作的范围是从 **0000H** 到 **TTmCCR0** 寄存器的设置值）



备注　**1.** **TTmESF** 位：**TMTm** 选项寄存器 **1(TTmOPT1)**的第 **0** 位。
TTmEOF 位：**TMTm** 选项寄存器 **1(TTmOPT1)**的第 **1** 位。
TTmEUF 位：**TMTm** 选项寄存器 **1(TTmOPT1)**的第 **2** 位。
2. V850E/IF3：m = 1
V850E/IG3：m = 0, 1

（6）通过编码器清除信号（TECRm 引脚）将计数器清除为 0000H

通过 TECRm 引脚的输入信号，有两种方法可以将 16 位计数器清除为 0000H，清除方法由 TTmIOC3.TTmSCE 位来选择。根据 TTmSCE 位的设置，该位也可以控制 TTmIOC3.TTmZCL, TTmIOC3.TTmBCL, TTmIOC3.TTmACL, TTmIOC3.TTmESC1 和 TTmIOC3.TTmECS0 位。

只有在编码器比较模式下，才可以通过下面描述的方法来清除计数器。

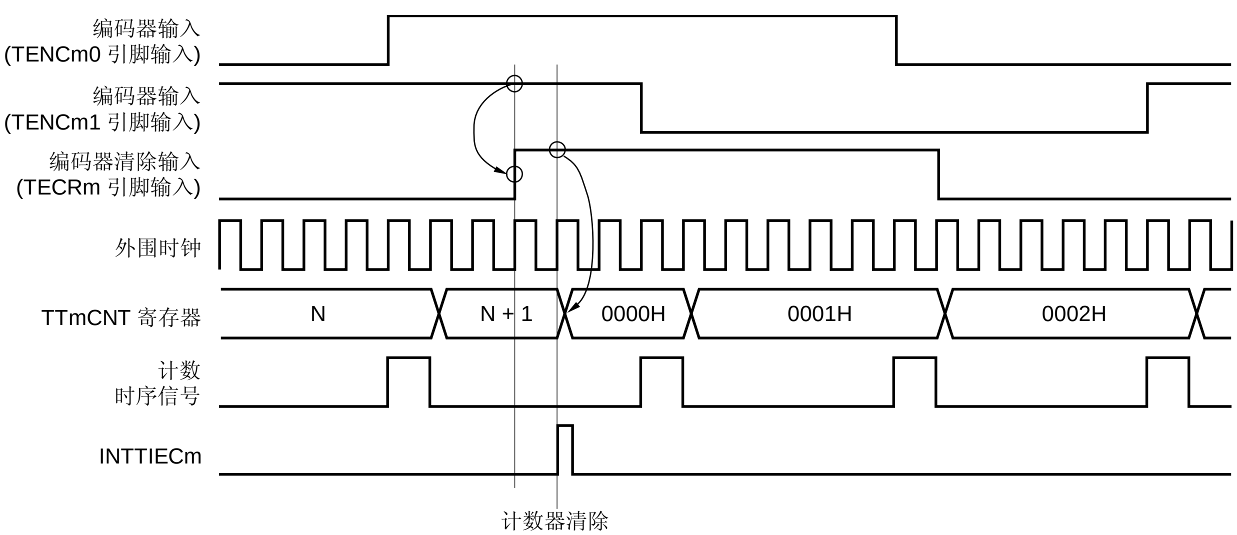
表 8-11. TTmSCE 位和 TTmZCL, TTmBCL, TTmACL, TTmECS1 和 TTmECS0 位之间的关系

清除方法	TTmSCE 位	TTmZCL 位	TTmBCL 位	TTmACL 位	TTmECS1, TTmECS0 位
<1>	0	无效	无效	无效	有效
<2>	1	有效	有效	有效	无效

（a） 清除方法 <1>：通过检测编码器清除信号（TECRm 引脚）的边沿（TTmSCE 位 = 0）

当 TTmSCE 位 = 0 时，如果检测到由 TTmECS1 和 TTmECS0 位指定的 TECRm 引脚有效边沿，16 位计数器被外部时钟同步清除为 0000H。此时，产生编码器清除中断请求信号（INTTTmEC）。当 TTmSCE 位 = 0 时，TTmZCL, TTmBCL 和 TTmACL 位的设置无效。

图 8-55. 操作示例 （当 TTmSCE 位 = 0, TTmECS1 和 TTmECS0 位 = 01 且 TTmUDS1 和 TTmUDS0 位 = 11）



备注　V850E/IF3：m = 1
　　　V850E/IG3：m = 0, 1

（b）清除方法 <2>：通过检测 TENCm0, TENCm1 和 TECRm 引脚的清除电平条件（TTmSCE 位 = 1）

当 TTmSCE 位 = 1 时，如果检测到由 TTmZCL, TTmBCL 和 TTmACL 位指定的 TECRm, TENCm0 或 TENCm1 引脚的清除电平条件，16 位计数器被清除为 0000H。此时不产生编码器清除中断请求信号（INTTTmEC）。当 TTmSCE 位 = 1 时，TTmECS1 和 TTmECS0 位的设置无效。

表 8-12. 当 TTmSCE 位 = 1 时的 16 位计数器清除条件

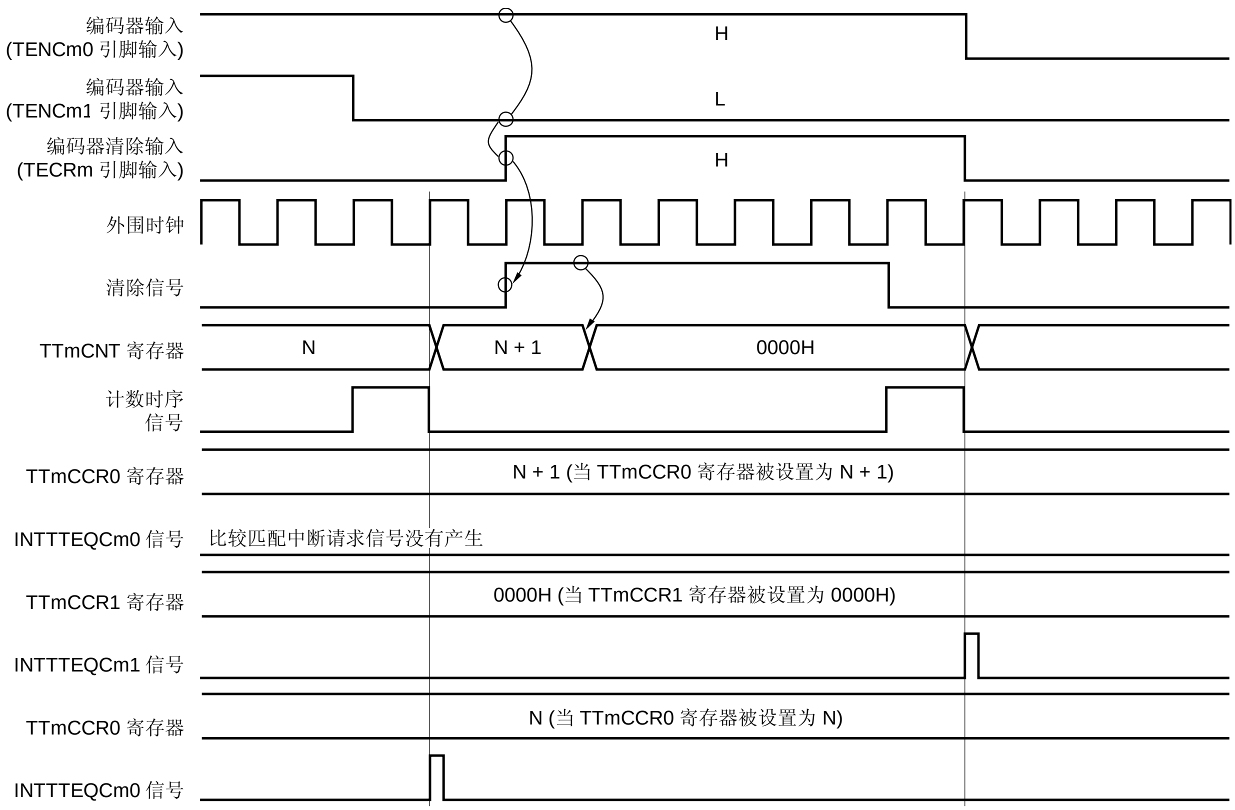
清除电平条件设置			编码器引脚的输入电平		
TTmZCL 位	TTmBCL 位	TTmACL 位	TECRm 引脚	TENCm1 引脚	TENCm0 引脚
0	0	0	L	L	L
0	0	1	L	L	H
0	1	0	L	H	L
0	1	1	L	H	H
1	0	0	H	L	L
1	0	1	H	L	H
1	1	0	H	H	L
1	1	1	H	H	H

注意事项　**当 TTmZCL, TTmBCL 和 TTmACL 位的清除电平条件和 TECRm, TENCm1 或 TENCm0 引脚的输入电平一致时，16 位计数器被清除为 0000H。**

备注　V850E/IF3：m = 1
V850E/IG3：m = 0, 1

图 8-56. 操作示例 （当 **TTmSCE** 位 **= 1, TTmZCL** 位 **= 1, TTmBCL** 位 **= 0, TTmACL** 位 **= 1, TTmUDS1** 和 **TTmUDS0** 位 **= 11, TECRm =** 高电平**, TENCm1 =** 低电平且 **TENCm0 =** 高电平）（**1/3**）

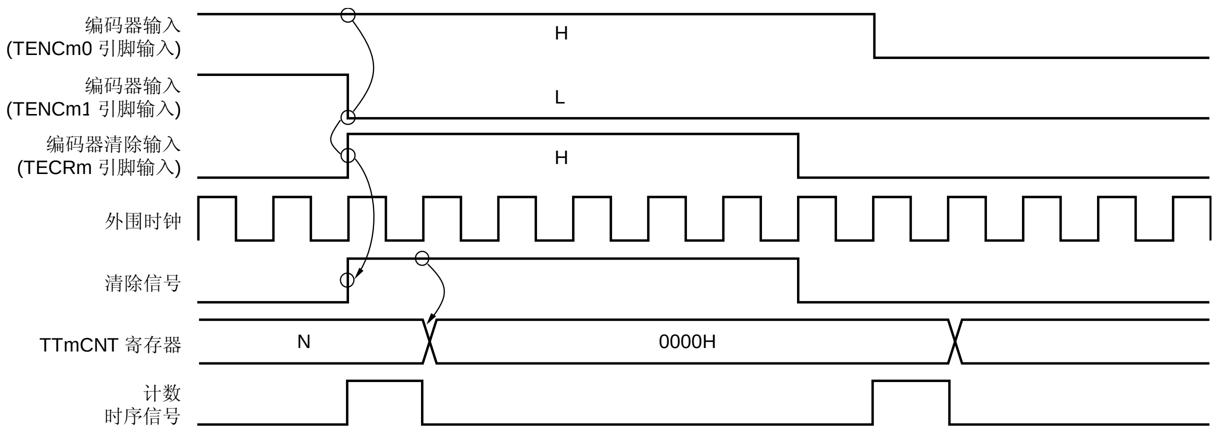
（**i**） 当计数器向上计数时，如果输入到 **TECRm** 引脚的高电平落后于输入到 **TENCm1** 引脚的低电平，在向上计数后，该计数器被清除。



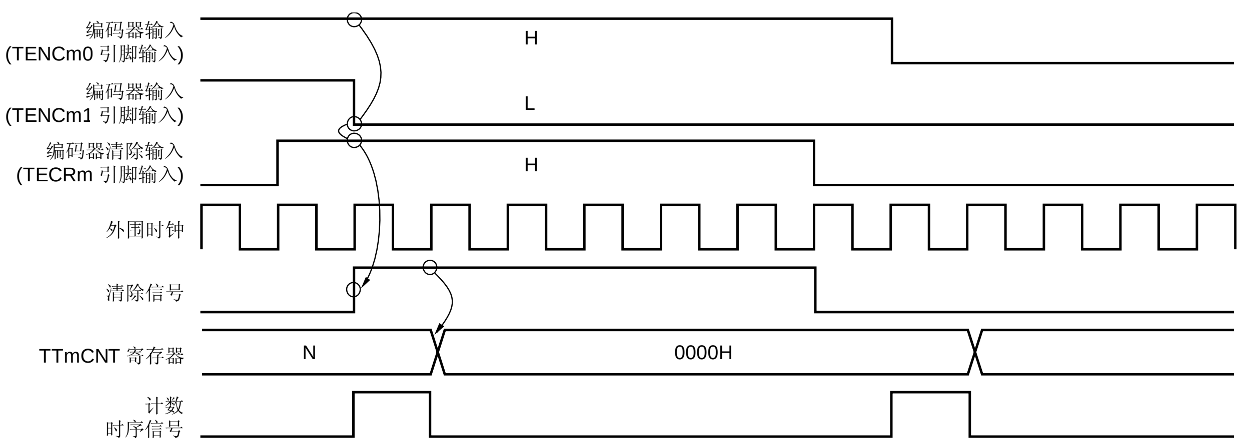
备注　　V850E/IF3：m = 1
　　　　V850E/IG3：m = 0, 1

图 **8-56.** 操作示例 （当 **TTmSCE** 位 **= 1, TTmZCL** 位 **= 1, TTmBCL** 位 **= 0, TTmACL** 位 **= 1, TTmUDS1** 和 **TTmUDS0** 位 **= 11, TECRm =** 高电平**, TENCm1 =** 低电平且 **TENCm0 =** 高电平）（**2/3**）

（**ii**）当计数器向上计数时，如果 **TECRm** 引脚的高电平和 **TENCm1** 引脚的低电平同时输入，该计数器不执行向上计数就被清除。



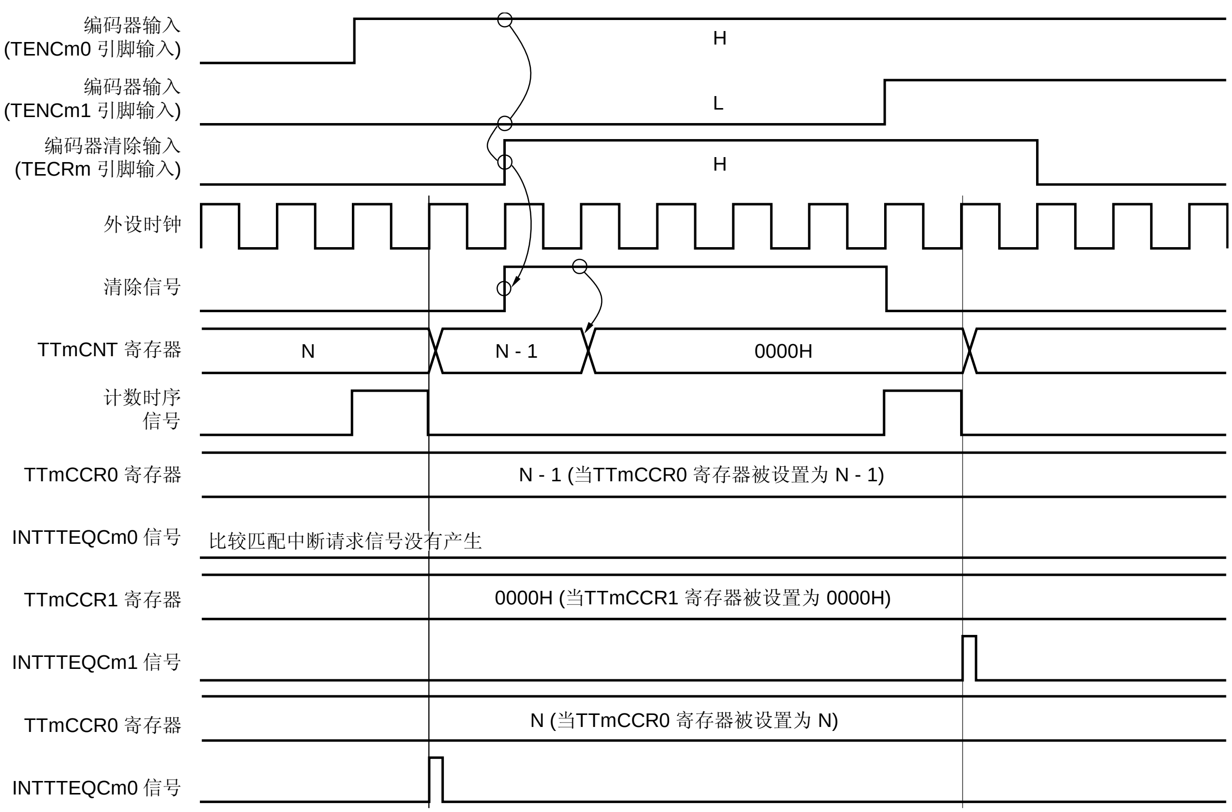
（**iii**）当计数器向上计数时，如果输入到 **TECRm** 引脚的高电平早于输入到 **TENCm1** 引脚的低电平，该计数器不执行向上计数就被清除。



备注　　V850E/IF3：m = 1
　　　　V850E/IG3：m = 0, 1

图 8-56. 操作示例（当 TTmSCE 位 = 1, TTmZCL 位 = 1, TTmBCL 位 = 0, TTmACL 位 = 1, TTmUDS1 和 TTmUDS0 位 = 11, TECRm = 高电平, TENCm1 = 低电平且 TENCm0 = 高电平）（3/3）

（**iv**） 当计数器向上计数时，如果输入到 **TECRm** 引脚的高电平落后于输入到 **TENCm1** 引脚的低电平，在向上计数后，该计数器被清除。



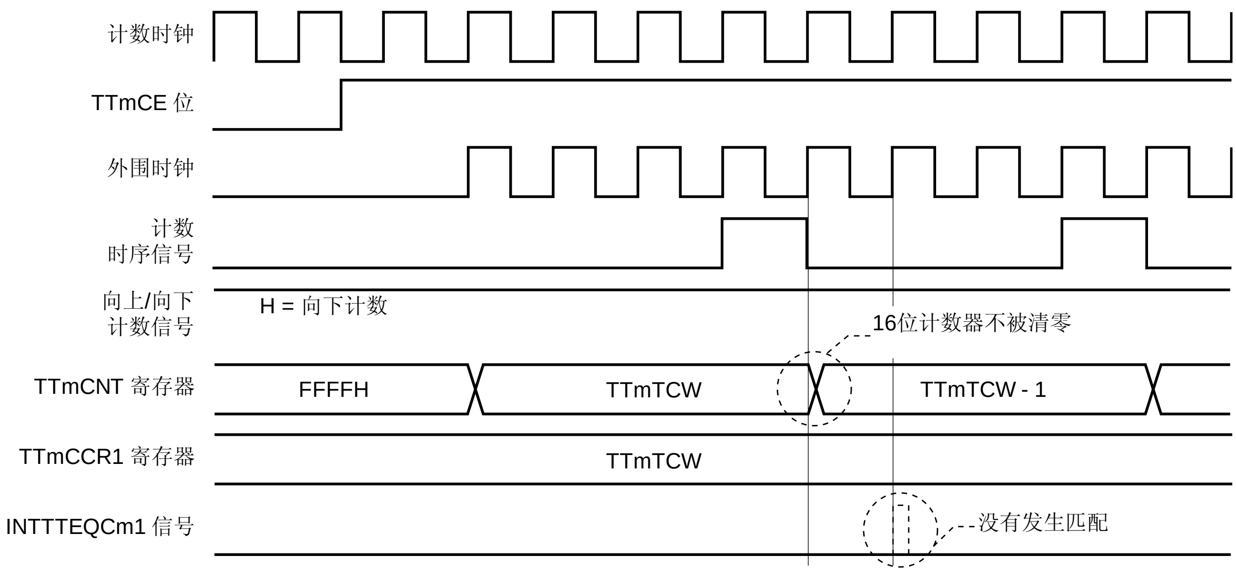
备注　V850E/IF3：m = 1
　　　V850E/IG3：m = 0, 1

如果计数器按照这种方法清除，即使输入到 TECRm 引脚的信号比较晚，也不会发生误计数，因为设置了 TECRm, TENCm1 和 TENCm0 引脚的清除电平条件，当检测到清除电平条件时，16 位计数器被清除为 0000H。

（7）使用编码器计数功能的注意事项

（a） 如果在操作启动后，没有立即产生比较一致中断

当 TTmCTL2.TTmECC 位 = 0 时，如果 TTmTCW 寄存器的值被设置到 TTmCCR0 或 TTmCCR1 寄存器，并启动计数器操作，如果在操作启动后，立即发生 16 位计数器的计数值（TTmTCW）和 CCRa 缓冲寄存器的值一致，一致被屏蔽且不产生比比较一致中断请求信号（INTTTEQCma）（a = 0, 1）。而且，设置 TTmCTL2.TTmECM1 和 TTmCTL2.TTmECM0 位并不会清除 16 位计数器为 0000H。

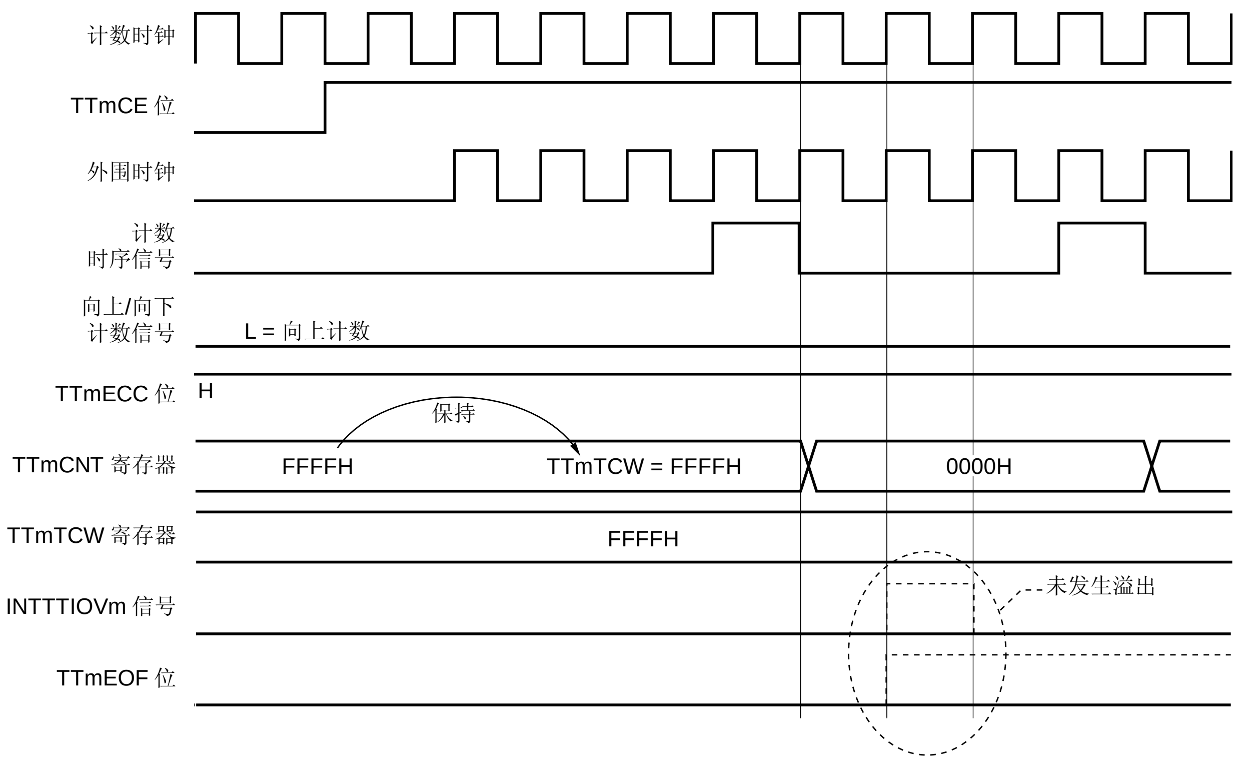


备注 V850E/IF3：m = 1
V850E/IG3：m = 0, 1

（**b**） 如果在操作启动后，没有立即发生溢出

如果在 TTmCTL2.TTmECC 位 = 1 时恢复计数操作，如果它保持的计数值是 FFFFH 且下一个计数操作是向上计数，16 位计数器不溢出。

在计数器启动操作之后，从一个计数值(TTmTCW 寄存器的值 = FFFFH)开始向上计数，计数器从 FFFFH 溢出到 0000H。但是，溢出的检测被屏蔽，溢出标志(TTmEOF)不被置位，不产生溢出中断请求信号（INTTTIOVm）。



备注 V850E/IF3：m = 1
V850E/IG3：m = 0, 1

8.6.10　编码器比较模式（TTmMD3 至 TTmMD0 位 = 1000）

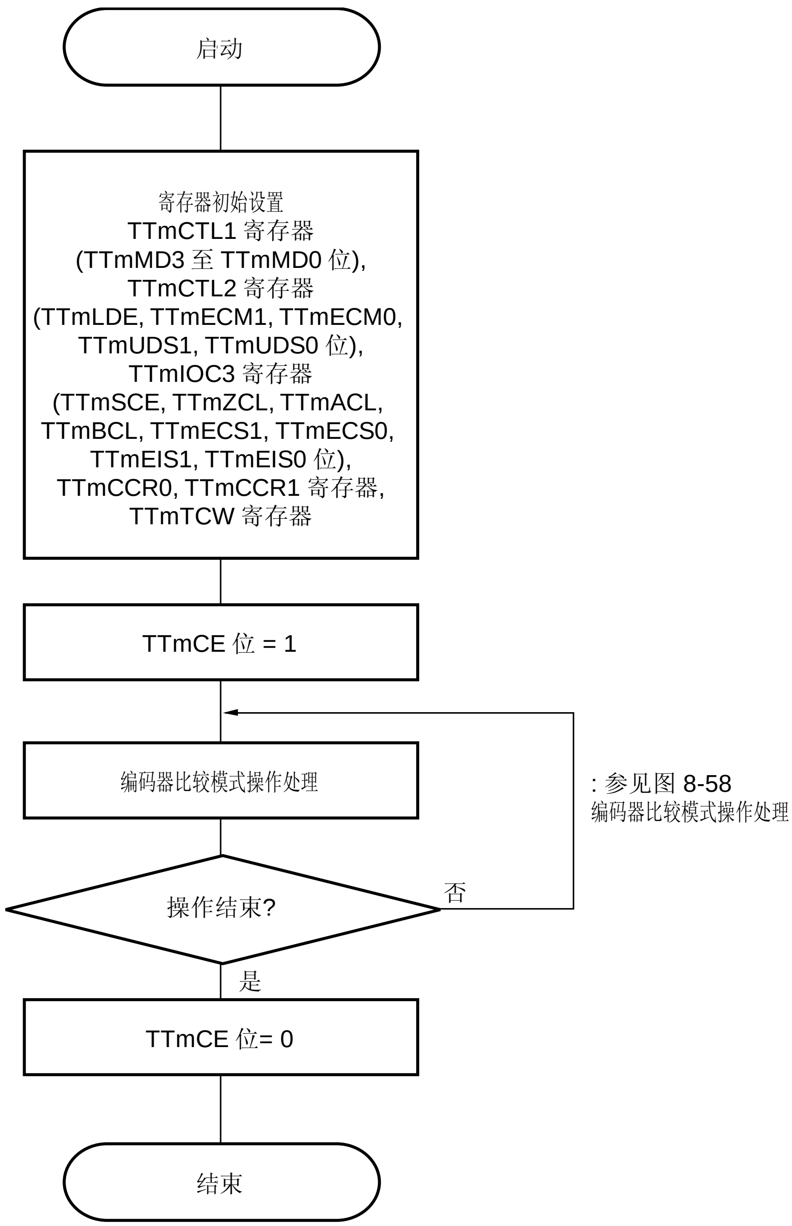
在编码器比较模式下，将 TTmCCR0 和 TTmCCR1 寄存器都用作比较寄存器，且输入引脚用作编码器计数功能（TENCm0, TENCm1 和 TECRm），可以控制编码器。

在该模式下，可以通过三种方法将 16 位计数器清除为 0000H：当计数器的计数值和 CCRa 缓冲寄存器的值一致时（产生比较一致中断请求信号（INTTTEQCma）），当检测到编码器清除输入（TECRm 引脚）的边沿时，当检测到 TENCm0, TENCm1 和 TECRm 引脚的清除电平条件时。

当 16 位计数器下溢时，TTmCCR0 寄存器的设置值被传输到计数器。

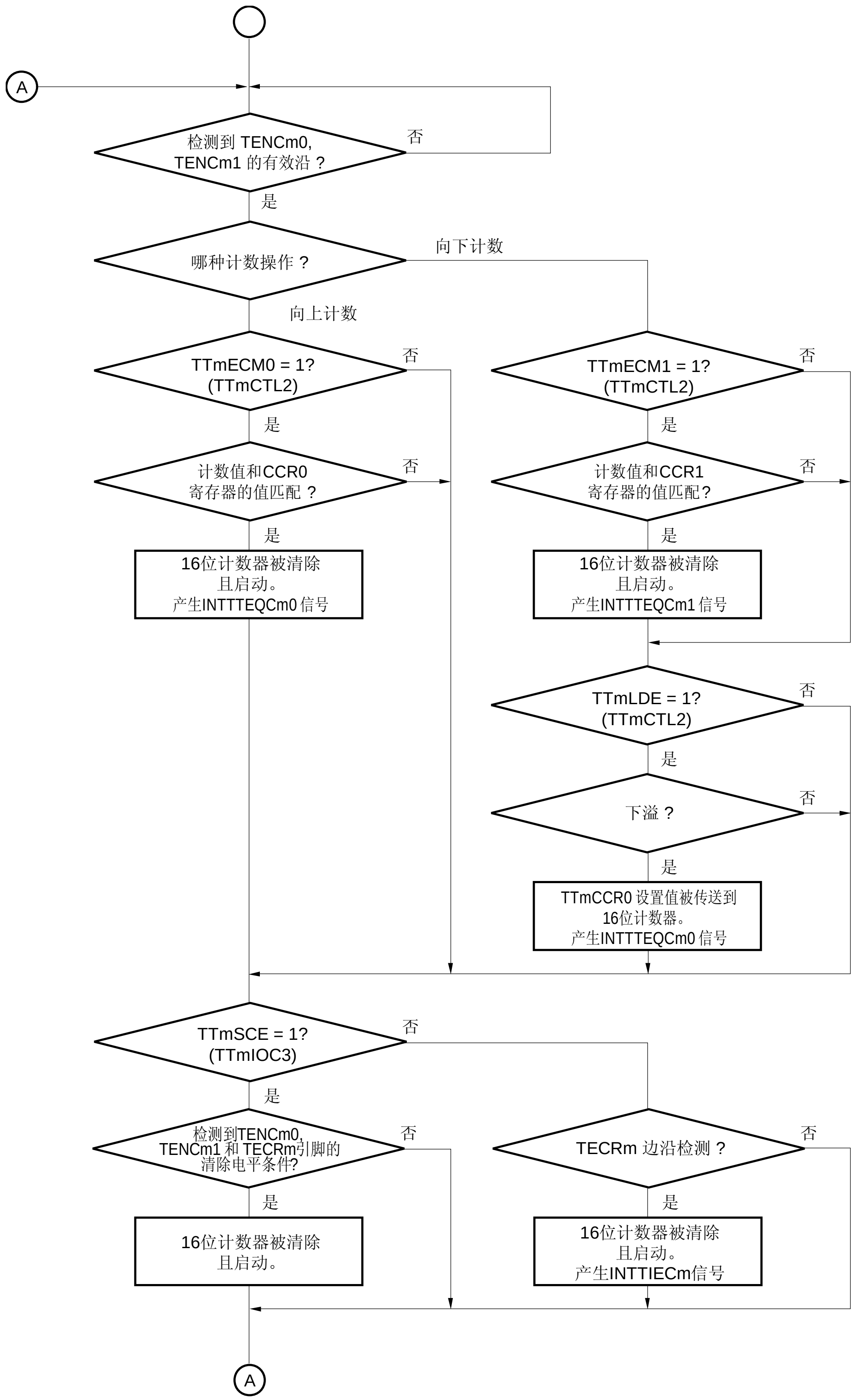
（1）编码器比较模式操作流程

图 8-57. 编码器比较模式操作流程



备注　V850E/IF3：m = 1
　　　V850E/IG3：m = 0, 1

图 8-58. 编码器比较模式操作处理



备注　V850E/IF3：m = 1

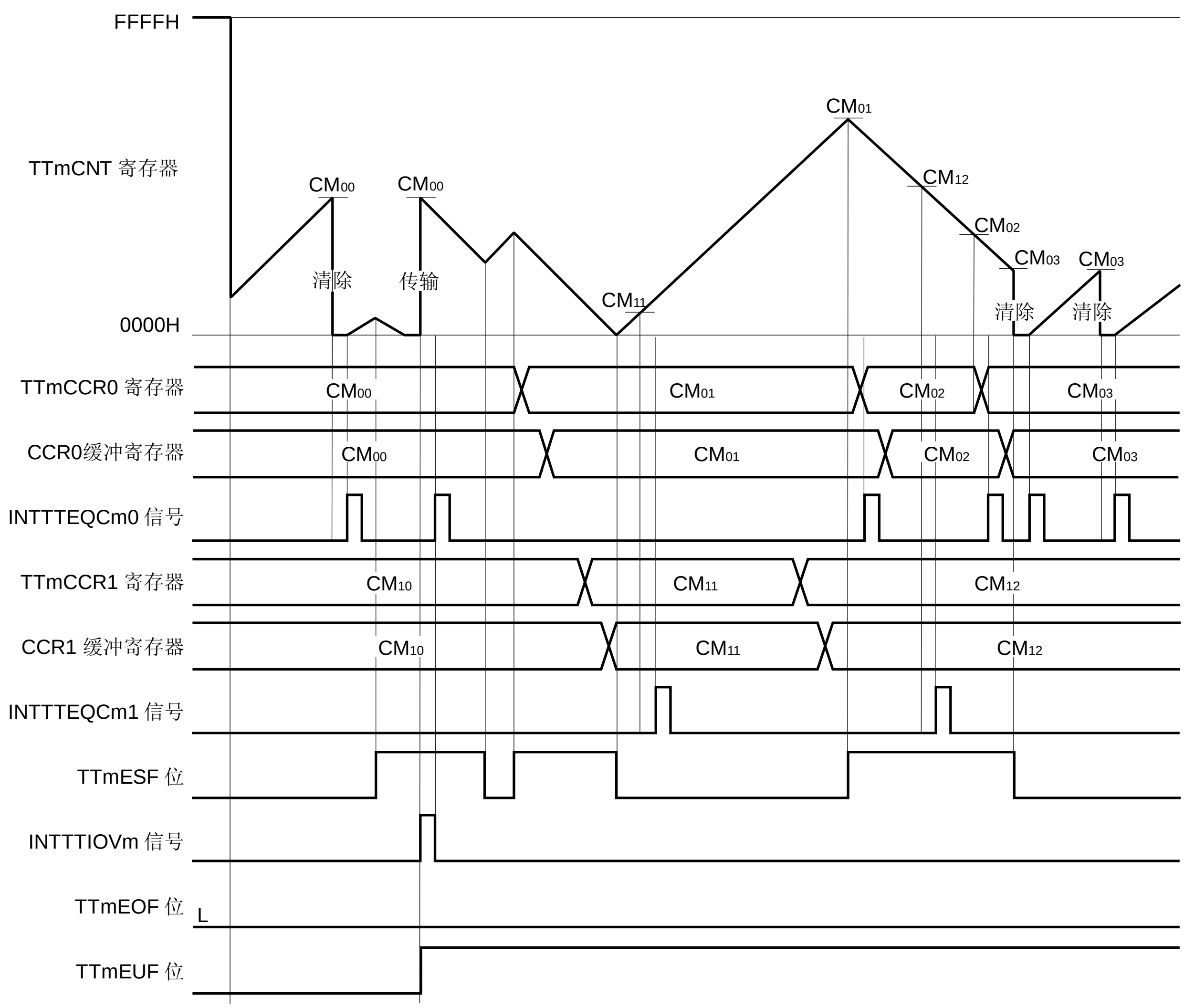
V850E/IG3：m = 0, 1

（2）编码器比较模式操作时序

（a） 基本时序 1

[寄存器设置条件]

- TTmCTL2.TTmECM1 和 TTmCTL2.TTmECM0 位 = 01
  当 16 位计数器的计数值和 CCR0 缓冲寄存器的值一致时，16 位计数器被清除为 0000H。
- TTmCTL2.TTmLDE 位 = 1
  当 16 位计数器下溢时，TTmCCR0 寄存器的设置值被传输到计数器。
- TTmIOC3.TTmSCE 位 = 0 且 TTmIOC3.TTmECS1 和 TTmIOC3.TTmECS0 位 = 00
  待检测和清除的编码器清除输入信号（TECRm 引脚）的边沿的说明(未指定边沿)



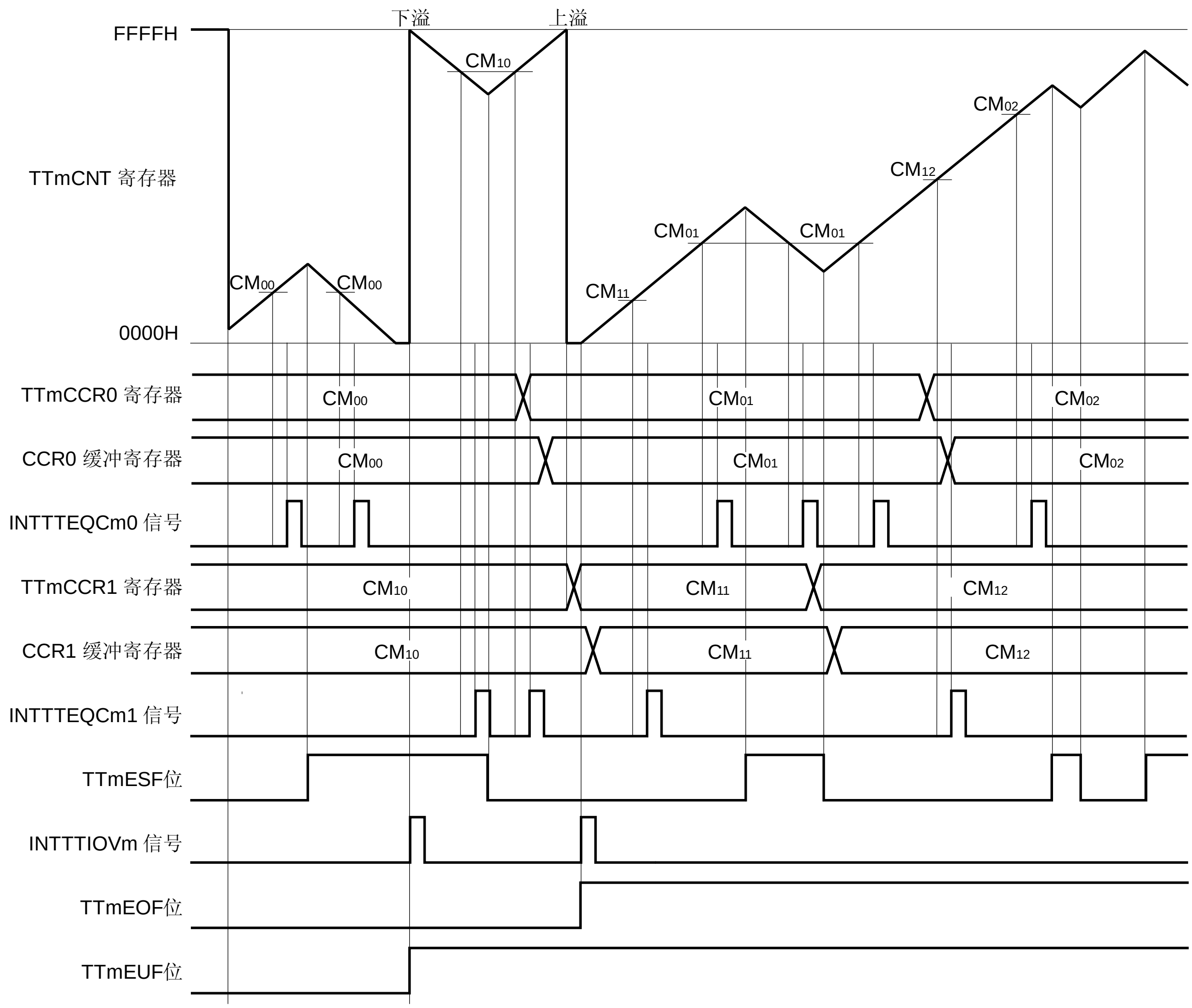
备注　V850E/IF3：m = 1
V850E/IG3：m = 0, 1

当 16 位计数器启动操作（TTmCE 位 = 0 → 1）时，TTmTCW 寄存器的设置值被传输到计数器，且 16 位计数器启动操作。
当计数器的计数值和 CCR0 缓冲寄存器的值一致时，产生比较一致中断请求信号(INTTTEQCm0)。因为 TTmECM0 位 = 1，如果下一个计数操作是向上计数，16 位计数器被清除为 0000H。
当计数器的计数值和 CCR1 缓冲寄存器的值一致时，产生比较一致中断请求信号(INTTTEQCm1)。因为 TTmECM1 位 = 0，当 16 位计数器的值和 CCR1 缓冲寄存器的值一致时，16 位计数器被清除为 0000H。
当 TTmLDE 位 = 1 且 TTmECM0 位 = 1 时，计数器的操作范围是从 0000H 到 TTmCCR0 寄存器的设置值。

（b） 基本时序 2

[寄存器设置条件]

- TTmCTL2.TTmECM1 和 TTmCTL2.TTmECM0 位 = 00
  即使 16 位计数器的计数值和 CCRa 缓冲寄存器的值一致时，16 位计数器也不会被清除。（n = 0, 1）
- TTmCTL2.TTmLDE 位 = 0
  在 16 位计数器下溢后，TTmCCR0 寄存器的设置值不被传输到计数器。
- TTmIOC3.TTmSCE 位 = 0 和 TTmIOC3.TTmECS1 和 TTmIOC3.TTmECS0 位 = 00
  待检测和清除的编码器清除输入信号（TECRm 引脚）的边沿的说明(未指定边沿)



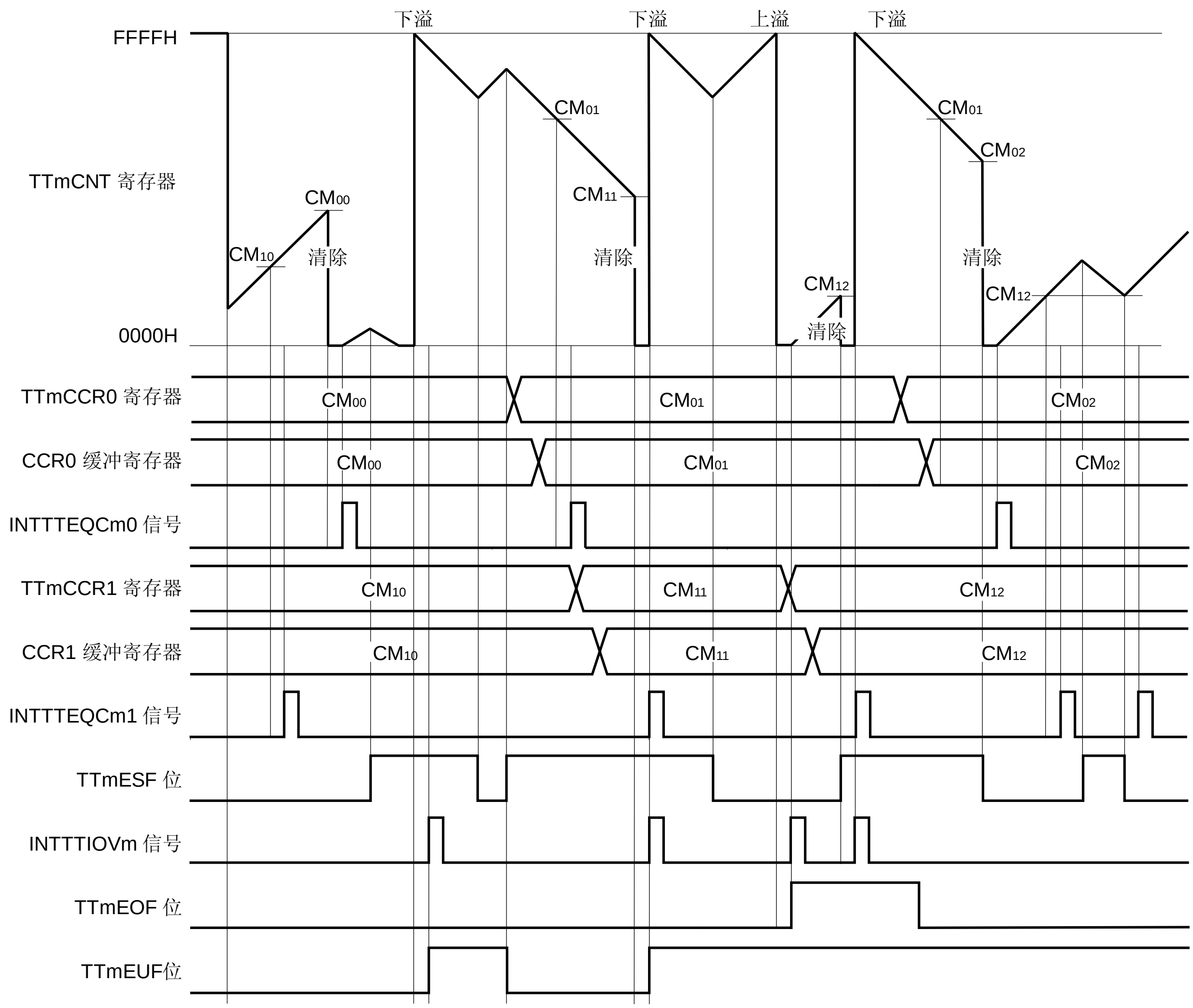
备注　V850E/IF3：m = 1
　　　V850E/IG3：m = 0, 1

当 16 位计数器启动操作（TTmCE 位 = 0 → 1）时，TTmTCW 寄存器的设置值被传输到计数器，且 16 位计数器启动操作。
当计数器的计数值和 CCR0 缓冲寄存器的值一致时，产生比较一致中断请求信号(INTTTEQCm0)。
当计数器的计数值和 CCR1 缓冲寄存器的值一致时，产生比较一致中断请求信号(INTTTEQCm1)。
即使 16 位计数器的计数值和 CCRa 缓冲寄存器的值一致时，16 位计数器也不会被清除为 0000H，因为 TTmECM1 和 TTmECM0 位 = 00。（a = 0, 1）

（c） 基本时序 3

[寄存器设置条件]

- TTmCTL2.TTmECM1 和 TTmCTL2.TTmECM0 位 = 11
  当 16 位计数器的计数值和 CCR0 缓冲寄存器的值一致时，16 位计数器被清除为 0000H。
  当 16 位计数器的计数值和 CCR1 缓冲寄存器的值一致时，16 位计数器被清除为 0000H。
- TTmCTL2.TTmLDE 位的设置无效。
- TTmIOC3.TTmSCE 位 = 0，且 TTmIOC3.TTmECS1 和 TTmIOC3.TTmECS0 位 = 00
  待检测和清除的编码器清除输入信号（TECRm 引脚）的边沿的说明(未指定边沿)



备注 V850E/IF3：m = 1
V850E/IG3：m = 0, 1

当 16 位计数器启动操作（TTmCE 位 = 0 → 1）时，TTmTCW 寄存器的设置值被传输到计数器，且 16 位计数器启动操作。
当计数器的计数值和 CCR0 缓冲寄存器的值一致时，产生比较一致中断请求信号(INTTTEQCm0)。此时，如果下一个计数操作是向上计数，则 16 位计数器被清除为 0000H。
当计数器的计数值和 CCR1 缓冲寄存器的值一致时，产生比较一致中断请求信号(INTTTEQCm1)。此时，如果下一个计数操作是向下计数，则 16 位计数器被清除为 0000H。

第九章 16 位间隔定时器 M (TMM)

定时器 M (TMM)是一个 16 位间隔定时器。
V850E/IF3 和 V850E/IG3 内置有 TMM0 至 TMM3。

9.1 概述

定时器 TMMn 概述如下（n = 0 至 3）。

- 间隔功能
- 8 路时钟可选
- 16 位计数器 × 1
  （定时器计数操作过程中，不能读取 16 位计数器。）
- 比较寄存器 × 1
  （定时器计数操作过程中，不能写入比较寄存器。）
- 比较一致中断 × 1

定时器 M 只支持清零并启动模式。不支持自由运行定时器模式。

9.2　配置

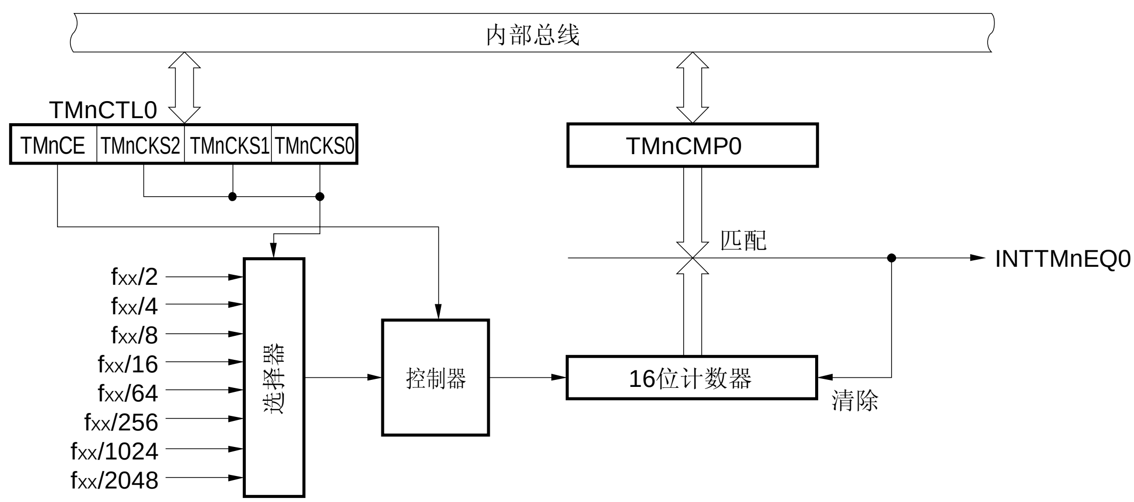
TMMn 包括下列硬件(n = 0 至 3)。

表 9-1. TMMn 的配置

项目	配置
定时器寄存器	16 位计数器 × 1
寄存器	TMMn 比较寄存器 0 （TMnCMP0）
控制寄存器	TMMn 控制寄存器 0 （TMnCTL0）

备注　　n = 0 至 3

图 9-1. TMMn 框图



备注　**1.** fxx：外围时钟频率
　　　2. n = 0 至 3

(1) 16 位计数器

该计数器是 16 位计数器，对内部时钟进行计数。
该计数器不可以读写。

(2) TMMn 比较寄存器 0（TMnCMP0）

TMnCMP0 寄存器是一个 16 位的比较寄存器。
该寄存器可以按 16 位宽度进行读取或写入。
复位输入将该寄存器清为 0000H。
总是可以通过软件将相同的值写入 TMnCMP0 寄存器。
在 TMMn 操作期间（TMnCTL0.TMnCE 位 = 1），禁止重写 TMnCMP0 寄存器。

复位后：0000H　　R/W　　地址：　　TM0CMP0 FFFFF544H, TM1CMP0 FFFFF554H,
TM2CMP0 FFFFF564H, TM3CMP0 FFFFF574H

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMnCMP0 (n = 0 至 3)																

9.3 控制寄存器

(1) TMMn 控制寄存器 0（TMnCTL0）

TMnCTL0 寄存器是 8 位寄存器，用于控制定时器 TMMn 的操作。
该寄存器可以按字节读写或按位读写。
复位输入将该寄存器设为 00H。
总是可以用软件将相同的值写入 TMnCTL0 寄存器。

复位后： 00H R/W 地址： TM0CTL0 FFFFF540H, TM1CTL0 FFFFF550H,
TM2CTL0 FFFFF560H, TM3CTL0 FFFFF570H

	<7>	6	5	4	3	2	1	0
TMnCTL0 (n = 0 至 3)	TMnCE	0	0	0	0	TMnCKS2	TMnCKS1	TMnCKS0

TMnCE	内部时钟操作使能/禁止 说明
0	TMMn 操作禁止 (16 位计数器异步复位)
1	TMMn 操作使能。开始供给操作时钟。 TMMn 操作启动。

使用TMnCE位对TMMn的内部时钟控制和内部电路重置是异步执行的。
当TMnCE位被清除为0时，TMMn 的内部时钟
被禁止 (固定为低电平)，且16位计数器异步复位。

TMnCKS2	TMnCKS1	TMnCKS0	计数时钟选择
0	0	0	$f_{XX}/2$
0	0	1	$f_{XX}/4$
0	1	0	$f_{XX}/8$
0	1	1	$f_{XX}/16$
1	0	0	$f_{XX}/64$
1	0	1	$f_{XX}/256$
1	1	0	$f_{XX}/1024$
1	1	1	$f_{XX}/2048$

注意事项 1. 当 TMnCE 位=0 时，对 TMnCKS2 至 TMnCKS0 置位。
但是，当 TMnCE 的值从 0 改变为 1 时，不可以同时设置 TMnCKS2 到 TMnCKS0 位的值。
2. 请确保将第 3 位至第 6 位清为“0”。

备注 f_{XX}： 外围时钟频率

9.4　操作

9.4.1　间隔计时器模式

在间隔定时器模式中，如果 TMnCTL0.TMnCE 位置为 1，则会以 TMnCMP0 设置的间隔生成中断请求信号（INTTMnEQ0）。

图 9-2. 间隔定时器的配置

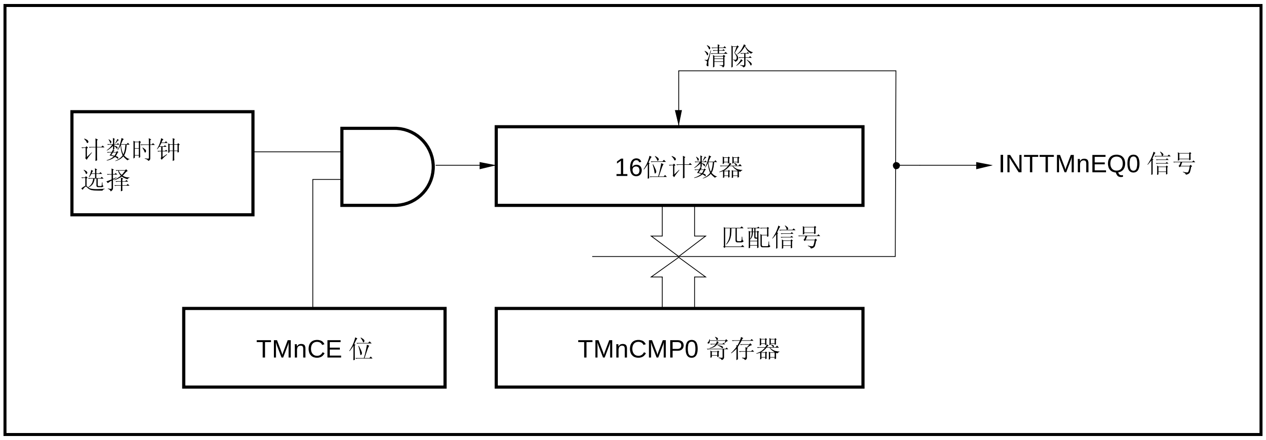
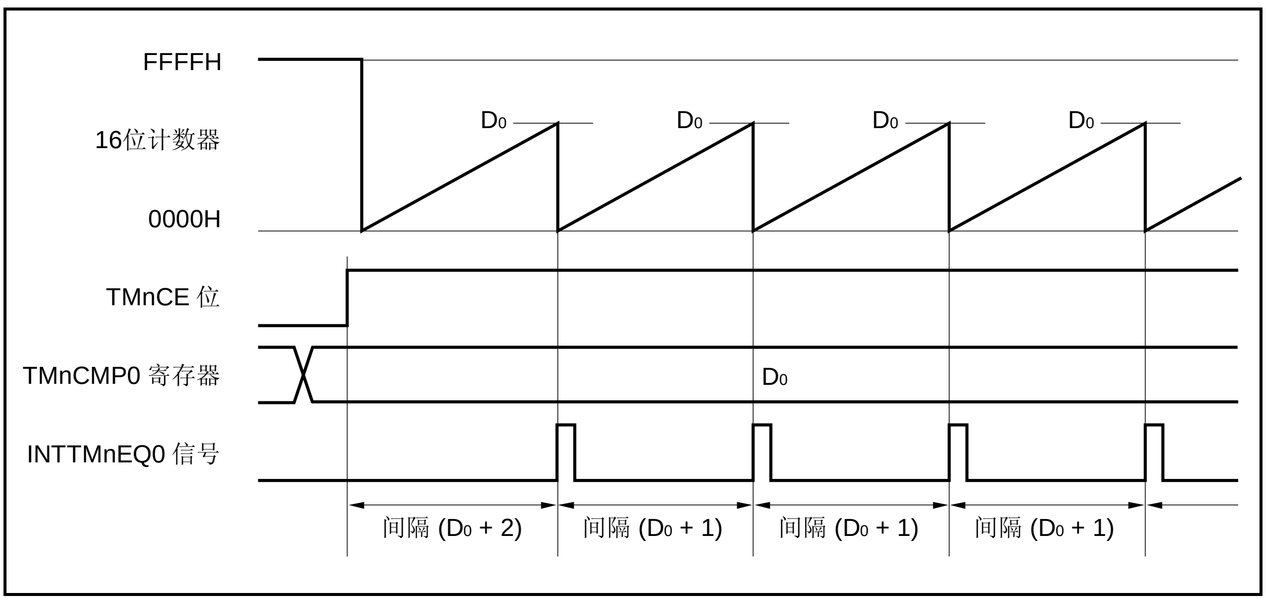


图 9-3. 间隔定时器工作的基本时序



当 TMnCE 位置为 1 时，16 位计数器的值由计数时钟同步从 FFFFH 清零为 0000H，计数器开始计数。

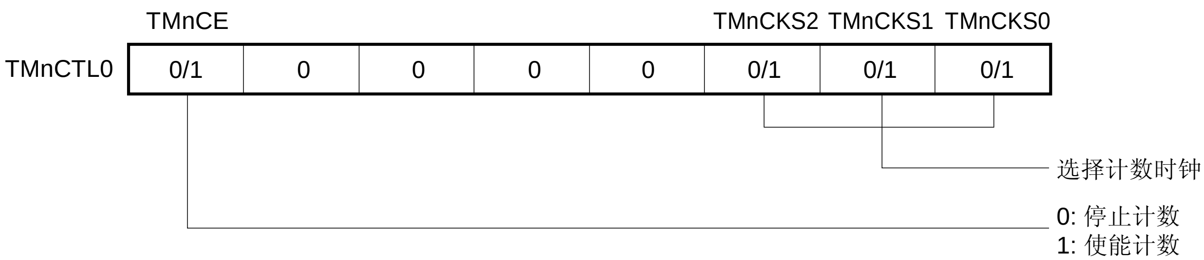
当 16 位计数器的计数值和 TMnCMP0 寄存器的数值相一致时，16 位计数器清零为 0000H，同时生成一个比较一致中断请求信号(INTTMnEQ0)。

时间间隔可用下述表达式计算。

时间间隔＝(TMnCMP0 寄存器的设定值+ 1) × 计数时钟周期

图 9-4. 间隔定时器模式操作时的寄存器设置

(a) TMMn 控制寄存器 0 （TMnCTL0）



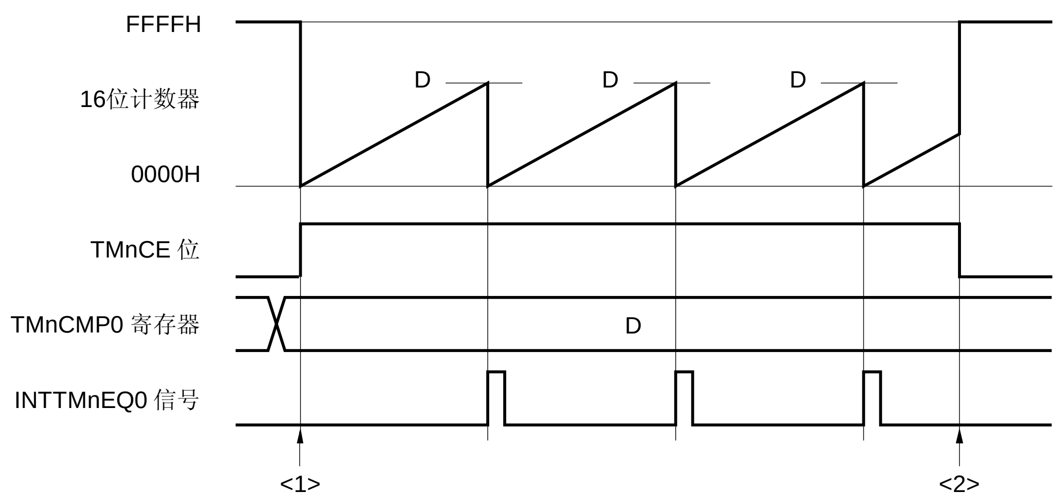
(b) TMMn 比较寄存器 0 （TMnCMP0）

如果TMnCMP0 寄存器被设为D_0，则间隔时间的计算如下：

间隔时间＝(D_0 + 1) × 计数时钟周期

(1) 间隔定时器模式操作流程

图 **9-5.** 间隔定时器模式下的软件处理流程



<1> 计数操作开始流程

开始

寄存器初始时钟
TMnCTL0 寄存器
(TMnCKS0 至 TMnCKS2 位)
TMnCMP0 寄存器

在设置TMnCE位为1之前，执行这些寄存器的初始化设置。

TMnCE 位 = 1

在启动计数(TMnCE 位= 1)的同时，可以设置TMnCKS0 至TMnCKS2 位。

<2> 计数操作停止流程

TMnCE 位 = 0

通过清除TMnCE 位为0，计数器被初始化，且计数停止。

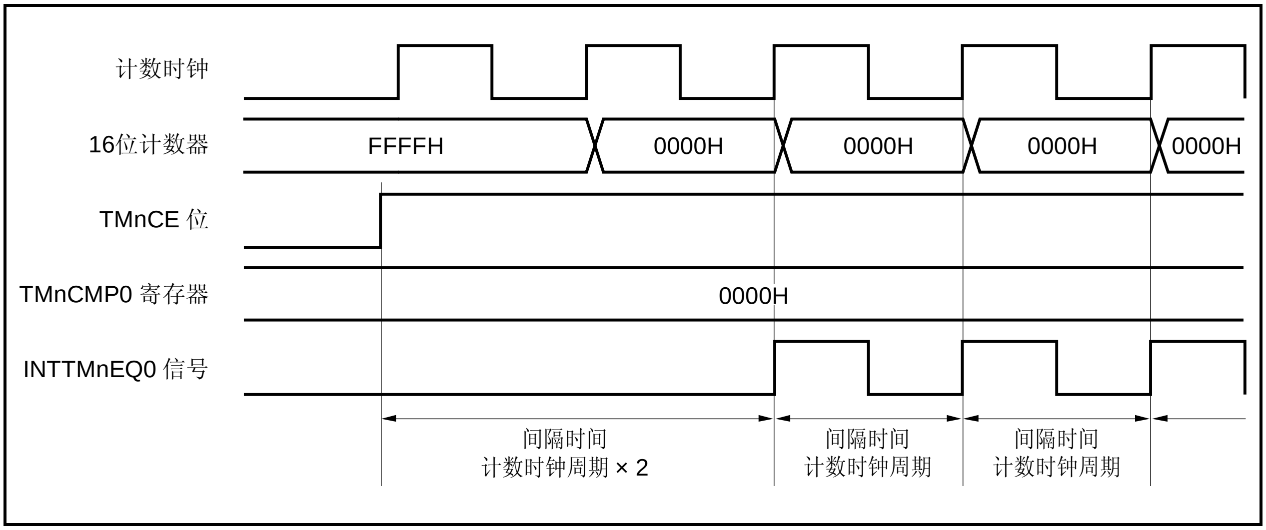
停止

(2) 间隔定时器模式操作时序

(a) TMnCMP0 寄存器设置为 0000H 时的操作

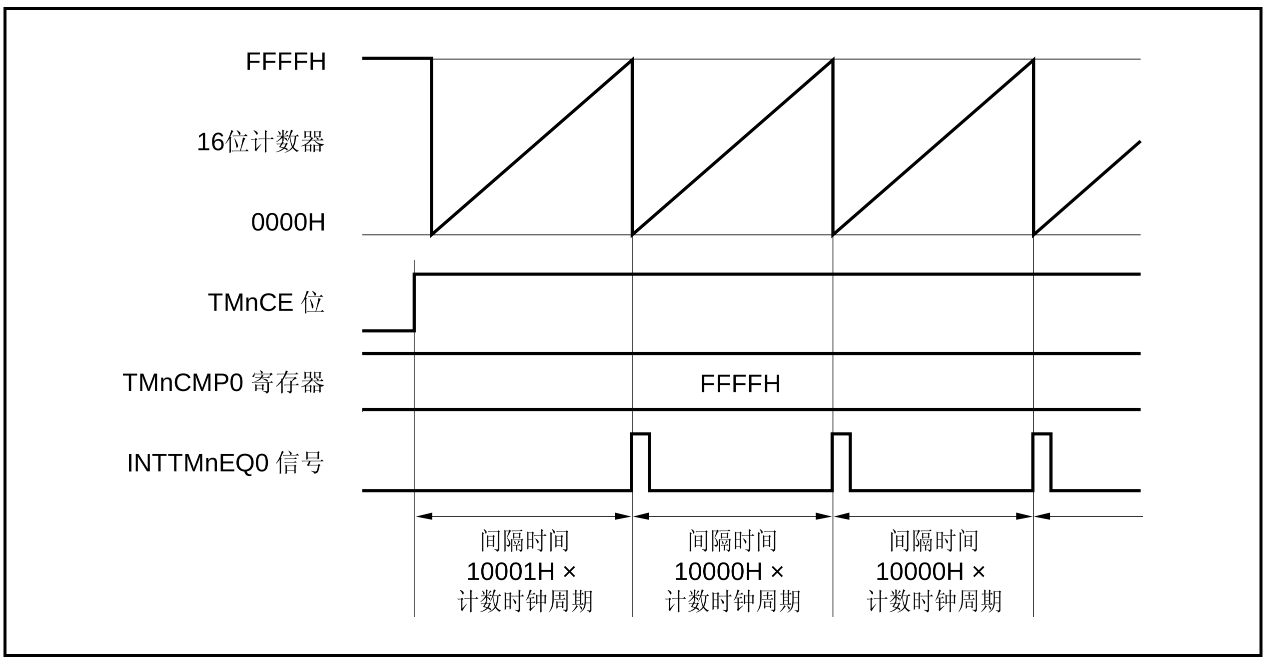
如果 TMnCMP0 寄存器被清为 0000H，那么每个计数时钟都会产生 INTTMnEQ0 信号。

16 位计数器的值总是 0000H。



(b) TMnCMP0 寄存器被设置为 FFFFH 时的操作

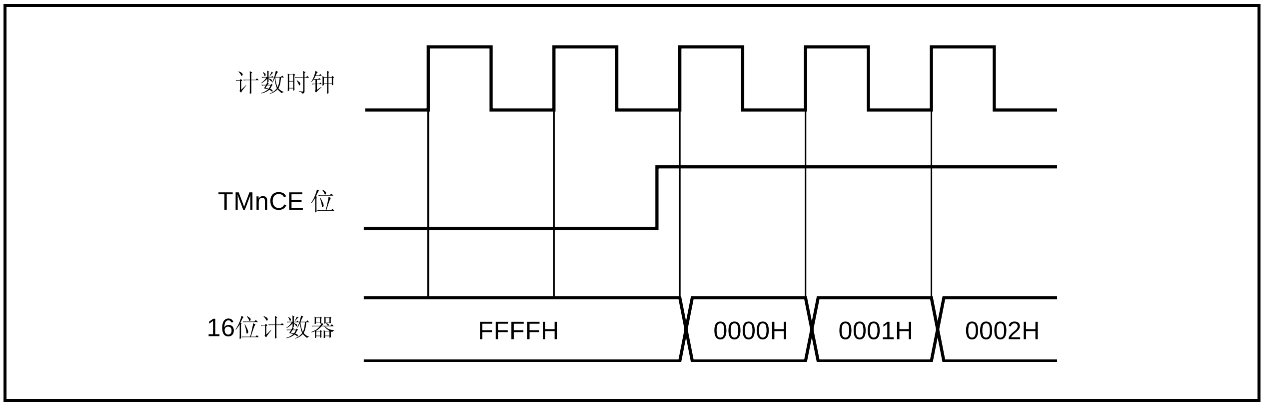
如果 TMnCMP0 寄存器设置为 FFFFH，则 16 位计数器最大计数至 FFFFH。计数器在下一个计数时刻被同步清为 0000H，并产生 INTTMnEQ0 信号。



9.5　注意事项

(1)　启动定时器的错误

在 TMnCTL0.TMnCE 位置为 1 且 TMMn 启动之后，需要一个时钟的时间产生第一个比较一致中断请求信号 (INTTMnEQ0)。这是因为当 TMnCE 位 = 0 时 16 位计数器的值是 FFFFH，且 TMMn 随着该计数时钟同步启动。



(2)　TMMn 工作时，禁止重写 TMnCMP0 和 TMnCTL0 寄存器

如果在 TMnCTL0.TMnCE 位 = 1 时对这两个寄存器进行重写，那么操作结果将无法保证。

如果这些寄存器被错误地改写，先将 TMnCE 位清除为 0，然后重置这些寄存器。

第十章 电机控制功能

10.1 功能概述

定时器 ABn （TABn） 和 TMQn 选项（TMQOPn）可以用作电机控制中的变频功能。它可以使用定时器 AAn（TAAn）来执行调谐功能，并且当 TABn 的值与 TAAn 的值一致时，可以启动 A/D 转换器 0 和 1 的 A/D 转换。以下操作可以用于实现电机控制功能。

- 16 位精度的 6 相 PWM 输出功能（带死区定时器，用于上桥臂和下上桥臂）
- 定时器调谐操作功能（使用 TAAn 时可调）
- 周期设置功能（在波峰或波谷中断期间可以改变周期）
- 比较寄存器重写： 随时重写，批量重写，或间隙重写（TABn 操作时可以选择）
- 中断及传输分选功能
- 死区时间设置功能
- A/D 转换器 0 和 1 的 A/D 触发功能（可以产生四种类型的时序）
- 0%输出和 100%输出
- 通过波峰中断和波谷中断可以选择 0%输出和 100%输出
- 强制输出停止功能
  - 当检测到外部引脚输入（TOABnOFF， TOAAnOFF）的有效边沿时
  - 当 A/D 转换器模块的比较器检测到过电压时
  - 当时钟监视功能检测到主时钟振荡停止时

备注　V850E/IF3：n = 0, 1，m = 2
　　　V850E/IG3：n = 0, 1，m = 2, 3

10.2 配置

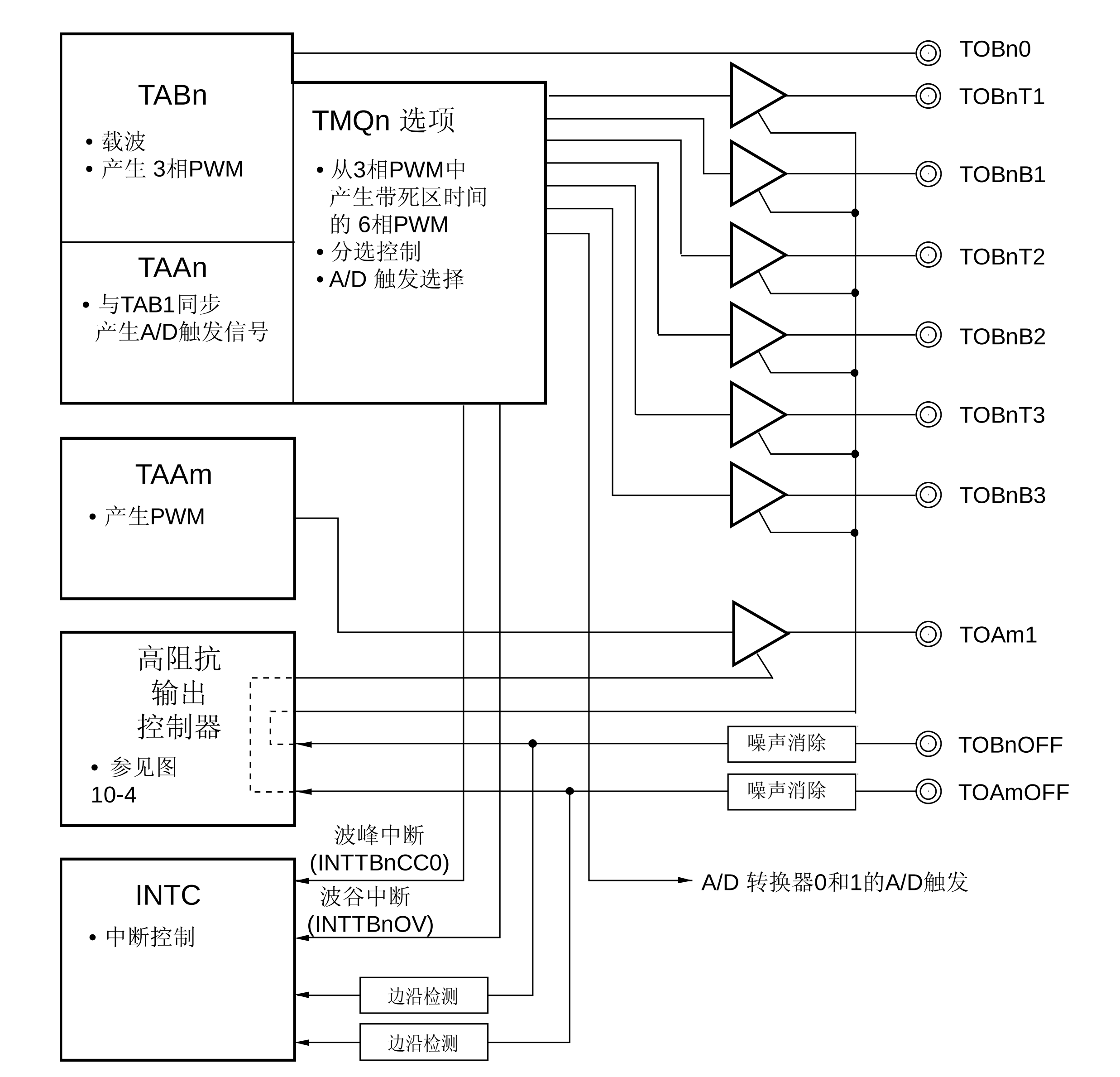
电机控制功能包含以下硬件。

项目	配置
定时器寄存器	死区时间计数器 m
比较寄存器	TABn 死区时间比较寄存器 （TABnDTC 寄存器）
控制寄存器	TABn 选项寄存器 0 （TABnOPT0） TABn 选项寄存器 1 （TABnOPT1） TABn 选项寄存器 2 （TABnOPT2） TABn 选项寄存器 3 （TABnOPT3） TABn 输入/输出控制寄存器 3 （TABnIOC3） 高阻抗输出控制寄存器 0，1（HZAyCTLa）

备注 V850E/IF3：m = 0 至 3，n = 0, 1，y = 0, 2, 3，当 y = 1 时 a = 0，a = 0, 1
V850E/IG3：m = 0 至 3，n = 0, 1。y = 0 至 3，a = 0, 1

- 使用 TABn（TOABn1，TOABn2，TOABn3）的输出，可以产生带死区时间的 6 相 PWM 输出。
- 6 相 PWM 输出的输出电平可以独立设置。
- 16 位定时器 / 计数器 TABn 向上 / 向下计数三角波。当定时器 / 计数器下溢且发生周期一致时，会产生一个中断。但是，中断产生次数被限制为最多 31 次。
- TAAn 可以和 TABn 同时执行计数（定时器调谐操作功能）。TAAn 可以设置为四种方式，它可以产生两种类型的 A/D 触发源（(INTTAnCC0 和 INTTAnCC1)）和两种类型的中断：TABn 的下溢中断（INTTBnOV）和周期一致中断（INTTBnCC0）。

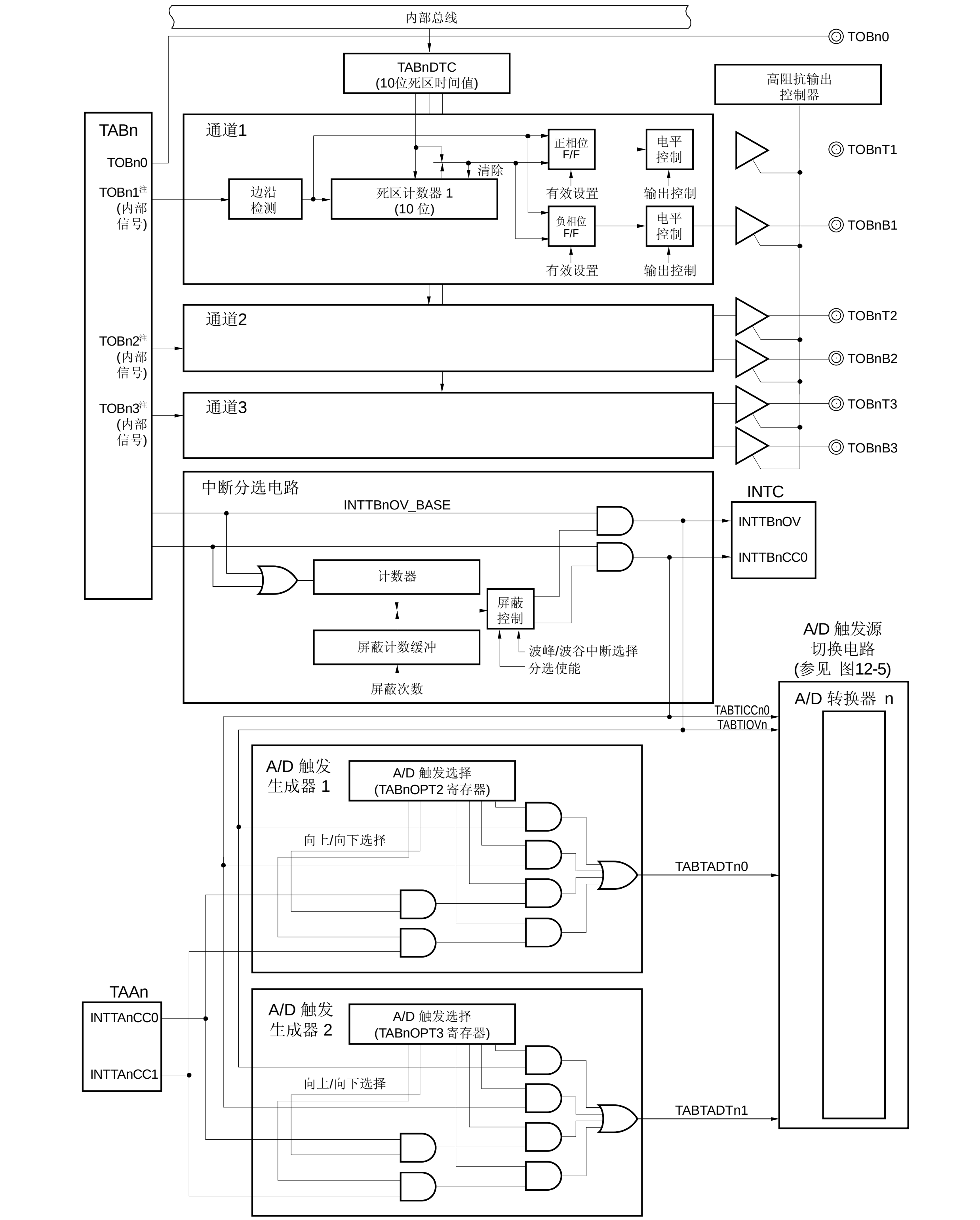
图 **10-1.** 电机控制的框图



备注 V850E/IF3：n = 0, 1，m = 2

V850E/IG3：n = 0, 1，m = 2, 3

图 10-2. TMQn 选项



注　　TOBn1，TOBn2 和 TOBn3 是输出引脚的复用功能。

备注　　V850E/IF3：n = 0, 1，m = 2

　　　　V850E/IG3：n = 0, 1，m = 2, 3

（1） TABn 死区时间比较寄存器 **（TABnDTC）**

TABnDTC 寄存器是 10 位比较寄存器，用于指定死区时间值。

当 TABnCTL0.TABnCE 位 = 1 时，禁止重写该寄存器。

该寄存器可以按 16 位宽度进行读取或写入。

系统复位后，该寄存器被置为 0000H 。

注意事项 **要产生死区时间周期，将 TABnDTC 寄存器设置为 1 或更大的值。**

当操作停止时（TABnCTL0.TABnCE 位 = 0），不产生死区时间，所以 TOABnT1 至 TOABnT3 及 TOABnB1 至 TOABnB3 引脚的输出电平处于初始状态。因此，为了保护系统，需要采取一些措施，比如在停止操作前，允许 TOABnT1 至 TOABnT3 及 TOABnB1 至 TOABnB3 引脚变为高阻抗状态，或者在设置这些引脚的输出电平时选择端口模式。

不需要死区时间时，将 TABnDTC 寄存器清除为 0。

复位后： 0000H R/W 地址： TAB0DTC FFFFF604H, TAB1DTC FFFFF644H

	15 … 10	9 … 0
TABnDTC (n = 0, 1)	000000	TABnDTC9 至 TABnDTC0

（2） 死区时间计数器 **1** 至 **3**

死区时间计数器是 10 位计数器，用于计数死区时间。

在 TOBnm 输出信号的上升沿或下降沿，这些定时器通过 TABn 被清零或向上计数，当它们的计数值与 TABnDTC 寄存器的值一致时，它们被清除或者停止计数。这些计数器的计数时钟相同，都由 TABn 的 TABnCTL0.TABnCKS2 至 TABnCTL0.TABnCKS0 位设置。

备注 **1.** 当 TABnOPT2.TABnDTM 位 = 1 时，操作不同。详情参见 **10.4.2 （4）自动死区时间宽度缩窄功能 （TABnOPT2.TABnDTM 位 = 1）**。

2. n = 0, 1，**m = 1** 至 **3**

10.3 控制寄存器

（1） TABn 选项寄存器 0（TABnOPT0）

TABnOPT0 寄存器是 8 位寄存器，用于控制由定时器 Qn 选项功能。

该寄存器可以按字节读写或按位读取或写入。但是，TABnCUF 位是只读的。

复位输入将该寄存器设为 00H。

注意事项 **只有在 6 相 PWM 输出模式下，才可以设置 TABnCMS 和 TABnCUF 位。请确保单独使用 TABn 时，将这些位清除为 0。**

复位后：00H R/W 地址：TAB0OPT0 FFFFF5E5H, TAB1OPT0 FFFFF625H

	<7>	<6>	<5>	<4>	3	<2>	<1>	<0>
TABnOPT0 (n = 0, 1)	TABnCCS3[注1]	TABnCCS2[注1]	TABnCCS1[注1]	TABnCCS0[注1]	0	TABnCMS	TABnCUF	TABnOVF[注2]

TABnCMS	比较寄存器重写模式选择
0	批量写入模式 (传输操作)
1	随时写入模式

- 只有当设置为6相PWM输出模式 (当TABnCTL1.TABnMD2 至 TABnCTL1.TABnMD0 位 = 111)时，TABnCMS 位有效。在其他任何模式下，将TABnCMS 位清除为 0。
- 定时器在工作时 (当TABnCTL0.TABnCE 位 = 1时)，可以重写TABnCMS位。
- 在批量写入模式下重写下列比较寄存器。
  TABnCCR0 至 TABnCCR3，TAnCCR0，TAnCCR1，TABnOPT1 和TABnDTC寄存器

TABnCUF	定时器ABn的向上/向下标志
0	定时器ABn向上计数
1	定时器ABn向下计数

只有当设置为6相PWM输出模式 (当TABnCTL1.TABnMD2 至 TABnCTL1.TABnMD0 位 = 111)时，TABnCUF 位有效。

注 **1.** **请确保**在 6 相 PWM 输出模式下，将 TABnCCS3 至 TABnCCS0 位清除为 0。

2. TABnOVF 位的详情，参见**第七章 16 位定时器/事件计数器 AB (TAB)。**

（2） **TABn 选项寄存器 1（TABnOPT1）**

TABnOPT1 寄存器是 8 位寄存器，用于控制由定时器 Qn 选项功能所产生的中断请求信号。

<R> TABnOPT1 寄存器产生的信号输出到中断分选电路、A/D 触发产生器 1 和 A/D 触发产生器 2，如图 10-2 所示。

当 TABnCTL0.TABnCE 位为 1 时，可以重写该寄存器。

可以选择两种重写模式(批量写入模式和随时写入模式)，模式由 TABnOPT0.TABnCMS 位的设置决定。

该寄存器可以按字节或按位读取或写入。

复位输入将该寄存器设为 00H。

复位后：00H　　R/W　　地址：　TAB0OPT1　FFFFF600H, TAB1OPT1　FFFFF640H

	<7>	<6>	5	4	3	2	1	0
TABnOPT1	TABnICE	TABnIOE	0	TABnID4	TABnID3	TABnID2	TABnID1	TABnID0

(n = 0, 1)

<R>

TABnICE	波峰中断 (INTTBnCC0 信号) 使能[注]
0	不使用 INTTBnCC0 信号 (不将其用作中断挑选的计数信号)
1	使用 INTTBnCC0 信号 (将其用作中断挑选的计数信号)

<R>

TABnIOE	波谷中断 (INTTBnOV 信号) 使能[注]
0	不使用 INTTBnOV 信号(不将其用作中断挑选的计数信号)
1	使用 INTTBnOV 信号(将其用作中断挑选的计数信号)

TABnID4	TABnID3	TABnID2	TABnID1	TABnID0	中断次数
0	0	0	0	0	不挑选（输出所有中断）
0	0	0	0	1	1 次屏蔽（每2次中断输出一次）
0	0	0	1	0	2 次屏蔽（每3次中断输出一次）
0	0	0	1	1	3 次屏蔽（每4次中断输出一次）
:	:	:	:	:	:
1	1	1	0	0	28 次屏蔽(每29次中断输出一次)
1	1	1	0	1	29 次屏蔽(每30次中断输出一次)
1	1	1	1	0	30 次屏蔽(每31次中断输出一次)
1	1	1	1	1	31 次屏蔽(每32次中断输出一次)

<R> 注　当使用波峰中断 (INTTBnCC0 信号)和波谷中断(INTTBnOV 信号)作为中断分选的计数信号，或作为 A/D 触发信号，设置要使用的信号为 1。
在分选中断时序产生一个 A/D 触发时序。

（3） TABn 选项寄存器 2（TABnOPT2）

TABnOPT2 寄存器是 8 位寄存器，用于控制定时器 Qn 选项功能。

当 TABnCTL0.TABnCE 位为 1 时，可以重写该寄存器。但是，当 TABnCE 位为 1 时，禁止重写 TABnDTM 位。可以重写相同的值。

该寄存器可以按字节或按位读取或写入。

复位输入将该寄存器设为 00H。

(1/2)

复位后：00H　　R/W　　地址：TAB0OPT2 FFFFF601H, TABnOPT2 FFFFF641H

	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
TABnOPT2	TABnRDE	TABnDTM	TABnATM3	TABnATM2	TABnAT3	TABnAT2	TABnAT1	TABnAT0

(n = 0, 1
m = 1 至 3)

TABnRDE	传输分选使能
0	不分选传输（每次波峰和波谷时都产生传输时序）
1	分选传输的间隔和 TABnOPT1 寄存器设置的中断分选相同。

TABnDTM	死区时间计数器操作模式选择
0	如果TABn的TOBnm输出是窄脉冲（TOBnm输出宽度<死区时间宽度），死区时间计数器在正常向上计数，则死区时间计数器清零并再次向上计数。
1	如果TABn的TOBnm输出是窄脉冲（TOBnm输出宽度<死区时间宽度），死区时间计数器在正常向上计数，则死区时间计数器向下计数，且死区控制宽度自动变窄。
在定时器运行期间禁止重写TABnDTM位。如果误写，将TABnCE位清零来停止定时器，重新设置TABnDTM位。	

注意事项 **1.** 当使用中断分选时（**TABnOPT1.TABnID4** 至 **TABnOPT1.TABnID0** 位设置为 **00000** 以外的值），请确保将 **TABnRDE** 位设置为 **1**。
因此，中断的产生和传输同时发生。不能单独设置中断和传输。如果中断和传输是分别设置的（**TABnRDE** 位 **= 0**），则传输无法正常执行。

2. 要产生死区时间周期，将 **TABnDTC** 设置为 **1** 或更大的值。
当操作停止时（**TABnCTL0.TABnCE** 位 **= 0**），不产生死区时间周期，**TOBnT1** 至 **TOBnT3** 和 **TOBnB1** 至 **TOBnB3** 引脚的输出电平处于默认状态。因此，为了保护系统，采取一些措施，比如在停止操作之前，允许 **TOBnT1** 至 **TOBnT3** 和 **TOBnB1** 至 **TOBnB3** 引脚变为高阻抗状态，或者在设置这些引脚的输出电平时选择端口模式。
当不需要死区时间时，将 **TABnDTC** 寄存器设置为 **0**。

(2/2)

TABnATM3	TABnATM3 模式选择
0	当死区时间计数器向上计数时，将INTTAnCC1中断输出作为A/D触发信号（TABTADTn0）
1	当死区时间计数器向下计数时，将INTTAnCC1中断输出作为A/D触发信号（TABTADTn0）

TABnATM2	TABnATM2 模式选择
0	当死区时间计数器向上计数时，将INTTAnCC0中断输出作为A/D触发信号（TABTADTn0）
1	当死区时间计数器向下计数时，将INTTAnCC0中断输出作为A/D触发信号（TABTADTn0）

TABnAT3[注]	A/D 触发输出控制 3
0	禁止为INTTAnCC1中断输出A/D触发信号(TABTADTn0)
1	使能为INTTAnCC1中断输出A/D触发信号(TABTADTn0)

TABnAT2[注]	A/D 触发输出控制 2
0	禁止为INTTAnCC0中断输出A/D触发信号(TABTADTn0)
1	使能为INTTAnCC0中断输出A/D触发信号(TABTADTn0)

TABnAT1[注]	A/D 触发输出控制 1
0	禁止为INTTBnCC0中断（波峰中断）输出A/D触发信号(TABTADTn0)
1	使能为INTTBnCC0中断（波峰中断）输出A/D触发信号(TABTADTn0)

TABnAT0[注]	A/D 触发输出控制 0
0	禁止为INTTBnOV中断（波谷中断）输出A/D触发信号(TABTADTn0)
1	使能为INTTBnOV中断（波谷中断）输出A/D触发信号(TABTADTn0)

注　TABnAT3 至 TABnAT0 位的设置详情，参见**第十二章 A/D 转换器 0 和 1**。

（4） TABn 选项寄存器 3（TABnOPT3）

TABnOPT3 寄存器是 8 位寄存器，用于控制定时器 Qn 选项功能。

当 TABnCTL0.TABnCE 位为 1 时，可以重写该寄存器

该寄存器可以按字节或按位读取或写入。

复位输入将该寄存器设为 00H。

复位后：00H　　R/W　　地址：TAB0OPT3 FFFFF603H, TAB1OPT3 FFFFF643H

	7	6	<5>	<4>	<3>	<2>	<1>	<0>
TABnOPT3 (n = 0, 1)	0	0	TABnATM7	TABnATM6	TABnAT7	TABnAT6	TABnAT5	TABnAT4

TABnATM7	TABnATM7 模式选择
0	当计数器向上计数时，将 INTAnCC1T中断输出作为A/D触发信号（TABTADTn1）
1	当计数器向下计数时，将 INTTAnCC1中断输出作为A/D触发信号（TABTADTn1）

TABnATM6	TABnATM6 模式选择
0	当计数器向上计数时，将 INTTAnCC0中断输出作为A/D触发信号（TABTADTn1）
1	当计数器向下计数时，将 INTTAnCC0中断输出作为A/D触发信号（TABTADTn1）

TABnAT7注	A/D 触发器输出控制3
0	禁止为INTTAnCC1中断输出A/D触发信号(TABTADTn1)
1	使能为INTTAnCC1中断输出A/D触发信号(TABTADTn1)

TABnAT6注	A/D 触发器输出控制2
0	禁止为INTTAnCC0中断输出A/D触发信号(TABTADTn1)
1	使能为INTTAnCC0中断输出A/D触发信号(TABTADTn1)

TABnAT5注	A/D 触发器输出控制1
0	禁止为INTTBnCC0中断（波峰中断）输出 A/D触发信号(TABTADTn1)
1	使能为INTTBnCC0中断（波峰中断）输出 A/D触发信号(TABTADTn1)

TABnAT4注	A/D 触发器输出控制0
0	禁止为INTTBnOV中断（波谷中断）输出A/D触发信号(TABTADTn1)
1	使能为INTTBnOV中断（波谷中断）输出A/D触发信号(TABTADTn1)

注　TABnAT7 至 TABnAT4 位的设置详情，参见**第十二章 A/D 转换器 0 和 1**。

（5） TABn I/O 控制寄存器 3 （TABnIOC3）

TABnIOC3 寄存器是 8 位寄存器，用于控制定时器 Qn 选项功能输出。

若要从 TOBnTm 引脚输出，应先将 TABnIOC0.TABnOEm 位设置为 1，然后设置 TABnIOC3 寄存器。

只有在 TABnCTL0.TABnCE 位为 0 时，才可以重写 TABnIOC3 寄存器。

当 TABnCTL0.TABnCE 位为 1 时，禁止重写 TABnIOC3 寄存器中每一位；但是，当 TABnCTL0.TABnCE 位为 1 时，TABnIOC3 寄存器的每一位都可以写入相同的值。

该寄存器可以按字节或按位读取或写入。

复位输入将该寄存器设为 A8H。

注意事项　当定时器用于 6 相 PWM 输出模式之外的模式，将 TABnIOC3 寄存器设置为默认值（A8H）。

备注　　通过 TABnIOC0 寄存器设置 TOBnTm 引脚的输出电平。

复位后：A8H　　R/W　　地址：TAB0IOC3　FFFFF602H, TAB1IOC3　FFFFF642H

	<7>	<6>	<5>	<4>	<3>	<2>	1	0
TABnIOC3	TABnOLB3	TABnOEB3	TABnOLB2	TABnOEB2	TABnOLB1	TABnOEB1	0	0

(n = 0, 1
m = 1 至 3)

TABnOLBm	TOBnBm 引脚输出电平的设置
0	禁止TOBnBm 引脚输出的反转
1	使能TOBnBm 引脚输出的反转

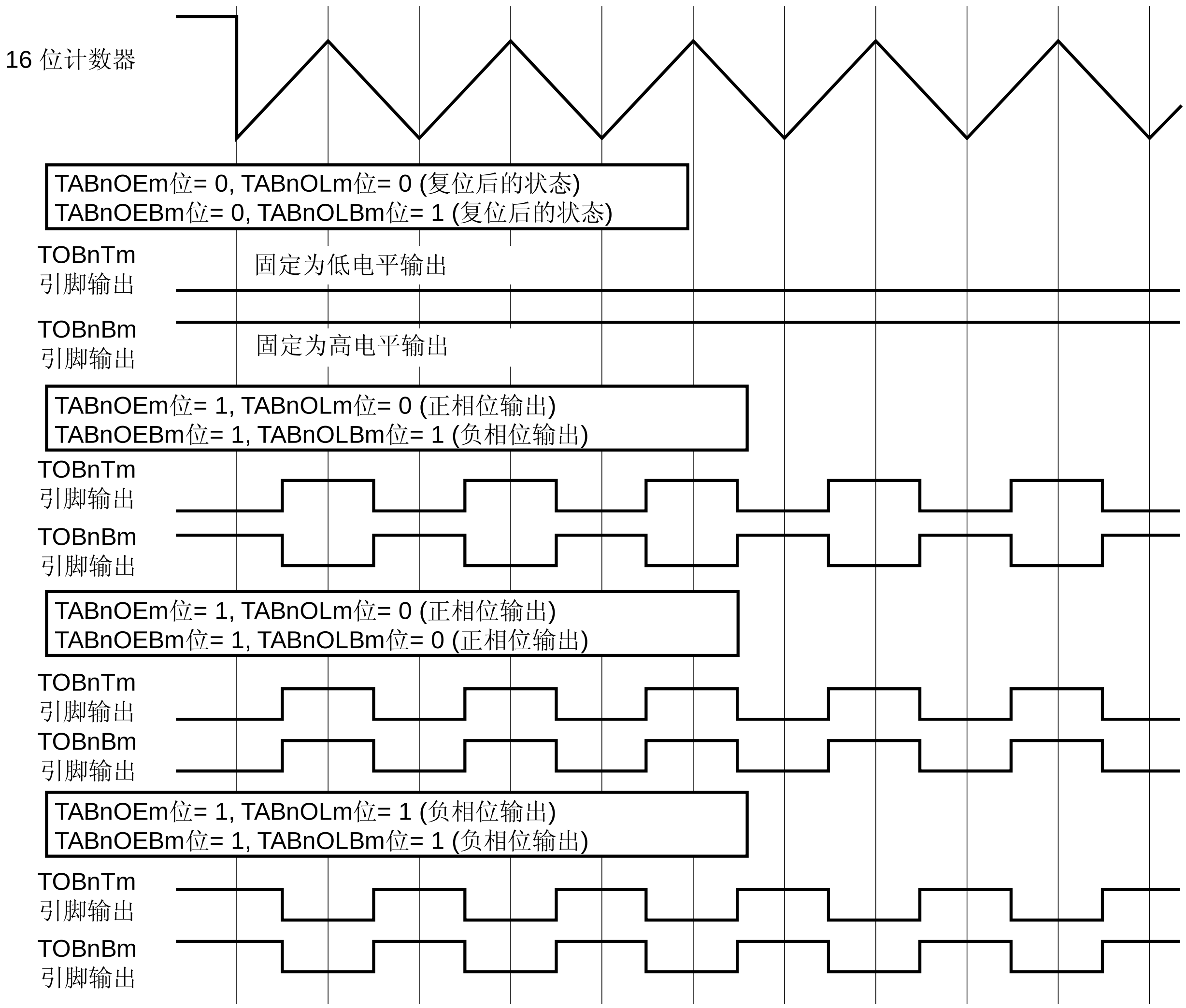
TABnOEBm	TOBnBm 引脚输出的设置
0	禁止TOBnBm 引脚输出 • 当TABnOLBm 位 = 0，从TOBnBm引脚输出低电平 • 当TABnOLBm 位 = 1，从TOBnBm引脚输出高电平
1	使能TOBnBm引脚输出

（a）从 **TOBnTm** 和 **TOBnBm** 引脚输出

TOBnTm 引脚输出由 TABnIOC0.TABnOLm 位和 TABnIOC0.TABnOEm 位控制。通过 TABnIOC3.TABnOLBm 位和 TABnIOC3.TABnOEBm 来控制 TOBnBm 的引脚输出。

以下显示了 6 相 PWM 输出模式中各种设置的定时器输出。

图 10-3. TOBnTm 和 TOBnBm 引脚的输出控制（不带死区时间）



备注　n = 0, 1
　　　m = 1 至 3

表 **10-1. TOBnTm** 引脚输出

TABnOLm 位	TABnOEm 位	TABnCE 位	TOBnTm 引脚输出
0	0	x	低电平输出
	1	0	低电平输出
		1	TOBnTm 正相输出
1	0	x	高电平输出
	1	0	高电平输出
		1	TOBnTm 负相输出

备注 n = 0, 1
m = 1 到 3

表 **10-2. TOBnBm** 引脚输出

TABnOLBm 位	TABnOEBm 位	TABnCE 位	TOBnBm 引脚输出
0	0	x	低电平输出
	1	0	低电平输出
		1	TOBnBm 正相输出
1	0	x	高电平输出
	1	0	高电平输出
		1	TOBnBm 负相输出

备注 n = 0, 1
m = 1 到 3

（6）高阻抗输出控制寄存器 00, 01, 10, 11, 20, 21, 30, 31 (HZAyCTL0, HZAyCTL1)

HZAyCTL0 和 HZAyCTL1 寄存器是 8 位寄存器，用于控制输出缓冲的高阻抗状态。

这些寄存器可以按字节读写或按位读取或写入。但是，HZAyDCFn 位为只读位，不能对其写入。

不能进行 16 位访问。

复位输入将这些寄存器置为 00H 。

总是可以通过软件向 HZAyCTLn 寄存器写入相同的数值。

以下显示了检测因素和控制寄存器之间的关系。

可以进行高阻抗控制的引脚	高阻抗控制因素		控制寄存器
	外部引脚	A/D 单元 (比较器)	
当 TOB0T1 至 TOB0T3 被输出时 当 TOB0B1 至 TOB0B3 被输出时	TOB0OFF	–	HZAyCTL0
	–	当超过 ANI00/ANI05 输入的小范围参考电压时(上升沿) 或没有达到时(下降沿)	HZA2CTL0
	–	当超过 ANI00/ANI05 输入的全范围参考电压时(上升沿) 或没有达到时(下降沿)	HZA2CTL1
当 TOA21 被输出时	TOA2OFF	–	HZAyCTL1
当 TOB1T1 至 TOB1T3 被输出时 当 TOB1B1 至 TOB1B3 被输出时	TOB1OFF	–	HZA1CTL0 [注]
	–	当超过 ANI10 至 ANI12 和 ANI15 至 ANI17 输入的小范围参考电压时(上升沿) 或没有达到时(下降沿)	HZA3CTL0
	–	当超过 ANI10 至 ANI12 和 ANI15 至 ANI17 输入的全范围参考电压时(上升沿) 或没有达到时(下降沿)	HZA3CTL1
当 TOA31 [注]被输出时	TOA3OFF [注]	–	HZA1CTL1 [注]

注　　仅限 V850E/IG3

注意事项　只有当端口引脚被设置为上表中所示的功能时，才可以执行高阻抗控制。

(1/3)

复位后：00H　R/W　地址：HZA0CTL0 FFFFF610H, HZA0CTL1 FFFFF611H,
HZA1CTL0 FFFFF618H, HZA1CTL1 FFFFF619H注1,
HZA2CTL0 FFFFF650H, HZA2CTL1 FFFFF651H,
HZA3CTL0 FFFFF658H, HZA3CTL1 FFFFF659H

	<7>	<6>	5	4	<3>	<2>	1	<0>
HZAyCTLn	HZAyDCEn	HZAyDCMn	HZAyDCNn	HZAyDCPn	HZAyDCTn	HZAyDCCn	0	HZAyDCFn

V850E/IF3
n = 0, 1
y = 0, 2, 3
当 y = 1时，n = 0

V850E/IG3
n = 0, 1
y = 0 至 3

HZAyDCEn	高阻抗输出控制
0	禁止高阻抗输出控制操作。引脚可以作为输出引脚。
1	使能高阻抗输出控制操作。

HZAyDCMn	通过HZAyDCCn位清除高阻抗状态的条件
0	不论外部引脚注2输入如何，HZAyDCCn位的设置有效。
1	当 外部引脚注2输入保持的电平被检测为异常时（有效电平），HZAyDCCn位的设置无效。
当HZAyDCEn 位 = 0时，重写HZAyDCMn位。	

注　**1.** 仅限 V850E/IG3

2. • V850E/IF3

HZAyCTL0：TOB0OFF 引脚，　HZAyCTL1：TOA2OFF 引脚，
HZA1CTL0：TOB1OFF 引脚，
HZA2CTL0：ANI00/ANI05 引脚，HZA2CTL1：ANI00/ANI05 引脚，
HZA3CTL0：ANI10 至 ANI12, ANI15 至 ANI17 引脚,
HZA3CTL1：ANI10 至 ANI12, ANI15 至 ANI17 引脚

• V850E/IG3

HZAyCTL0：TOB0OFF 引脚，　HZAyCTL1：TOA2OFF 引脚，
HZA1CTL0：TOB1OFF 引脚，　HZA1CTL1：TOA3OFF 引脚，
HZA2CTL0：ANI00/ANI05 引脚，HZA2CTL1：ANI00/ANI05 引脚，
HZA3CTL0：ANI10 至 ANI12, ANI15 至 ANI17 引脚,
HZA3CTL1：ANI10 至 ANI12, ANI15 至 ANI17 引脚

(2/3)

HZAyDCNn	HZAyDCPn	外部引脚[注1]输入边沿说明
0	0	无有效边沿(禁止通过外部引脚[注1]输入设置HZAyDCFn 位
0	1	外部引脚[注1]的上升沿输入有效（通过上升沿输入检测到异常）[注2]
1	0	外部引脚[注1]的下降沿输入有效（通过下降沿输入检测到异常）[注2]
1	1	禁止设置

- 当HZAyDCEn 位为0时，重写HZAyDCNn和HZAyDCPn位。
- INTP00, INTP02, INTP08和INTP10 引脚的边沿的说明，参见 20.4.2 (1) 外部中断上升沿说明寄存器 0（INTR0)，外部中断下降沿说明寄存器 0（INTF0）和(2) 外部中断上升沿说明寄存器 1（INTR1)，外部中断上升沿说明寄存器 1（INTF1)。
- 外部引脚的边沿必须从TOB0OFF，TOB1OFF，TOA2OFF和TOA3OFF[注3] 引脚开始指定。然后除TOB0OFF，TOB1OFF，TOA2OFF和TOA3OFF[注3] 引脚之外的外部引脚的边沿必须被指定。否则，当指定TOB0OFF，TOB1OFF，TOA2OFF和TOA3OFF[注3] 引脚的边沿时，可能会检测到不确定的边沿。
- 在操作使能后（设置HZAyDCEn位为1），当有效边沿输入时，执行高阻抗输出控制。如果操作使能时，外部引脚[注1]处于有效电平，因此，不执行高阻抗输出。

注 **1.** • V850E/IF3

HZAyCTL0：TOB0OFF 引脚， HZAyCTL1：TOA2OFF 引脚，
HZA1CTL0：TOB1OFF 引脚，
HZA2CTL0：ANI00/ANI05 引脚， HZA2CTL1：ANI00/ANI05 引脚，
HZA3CTL0：ANI10 至 ANI12, ANI15 至 ANI17 引脚,
HZA3CTL1：ANI10 至 ANI12, ANI15 至 ANI17 引脚

• V850E/IG3

HZAyCTL0：TOB0OFF 引脚， HZAyCTL1：TOA2OFF 引脚，
HZA1CTL0：TOB1OFF 引脚， HZA1CTL1：TOA3OFF 引脚，
HZA2CTL0：ANI00/ANI05 引脚， HZA2CTL1：ANI00/ANI05 引脚，
HZA3CTL0：ANI10 至 ANI12, ANI15 至 ANI17 引脚,
HZA3CTL1：ANI10 至 ANI12, ANI15 至 ANI17 引脚

2. 要检测比较器的电压超过参考电压，设置上升沿输入。要检测比较器的电压没有达到参考电压，设置下降沿输入。

3. 仅限 V850E/IG3

(3/3)

HZAyDCTn	高阻抗输出触发位
0	无操作
1	通过软件将引脚设置为高阻抗状态，且HZAyDCFn位被置为1。

- 当表示异常的边沿输入到外部引脚[注]时（根据HZAyDCNn 和 HZAyDCPn 位的设置来检测）时，即使该位被置1，HZAyDCTn 位也无效。
- 当读取HZAyDCTn位时，该位总是0，因为它是软件触发位。
- 当 HZAyDCEn 位 = 0时，即使该位被置1，HZAyDCTn 位也无效。
- 禁止同时将HZAyDCTn 和 HZAyDCCn 位设置为 1。

HZAyDCCn	高阻抗输出控制清除位
0	无操作
1	已经进入高阻抗状态的引脚可以通过软件设置为使能输出，且HZAyDCFn 位被清除为 0。

- 不论外部引脚[注]的状态如何，当HZAyDCM位=0时，引脚总是可用作输出引脚。
- 当表示异常的边沿输入到外部引脚[注]时（根据HZAyDCNn 和 HZAyDCPn 位的设置来检测）时，当 HZAyDCM 位 = 1时，即使该位被置1，HZAyDCCn 位也无效。
- 读取HZAyDCCn位时，它总是0。
- 当HZAyDCEn 位 = 1时，即使该位被置1，HZAyDCCn 位也无效。
- 禁止同时将HZAyDCTn 和 HZAyDCCn 位置 1。

HZAyDCFn	高阻抗输出状态标志
重置 (0)	表示引脚的输出已经使能 • 当HZAyDCEn位 = 0时，该位被清0。 • 当HZAyDCCn位 = 1时，该位被清0。
置位 (1)	表示引脚进入高阻抗状态 • 当HZAyDCTn位 = 1时，该位被设置为1。 • 当表示异常的边沿输入到外部引脚[注]时（根据HZAyDCNn 和 HZAyDCPn 位的设置来检测）时，该位被设置为1。

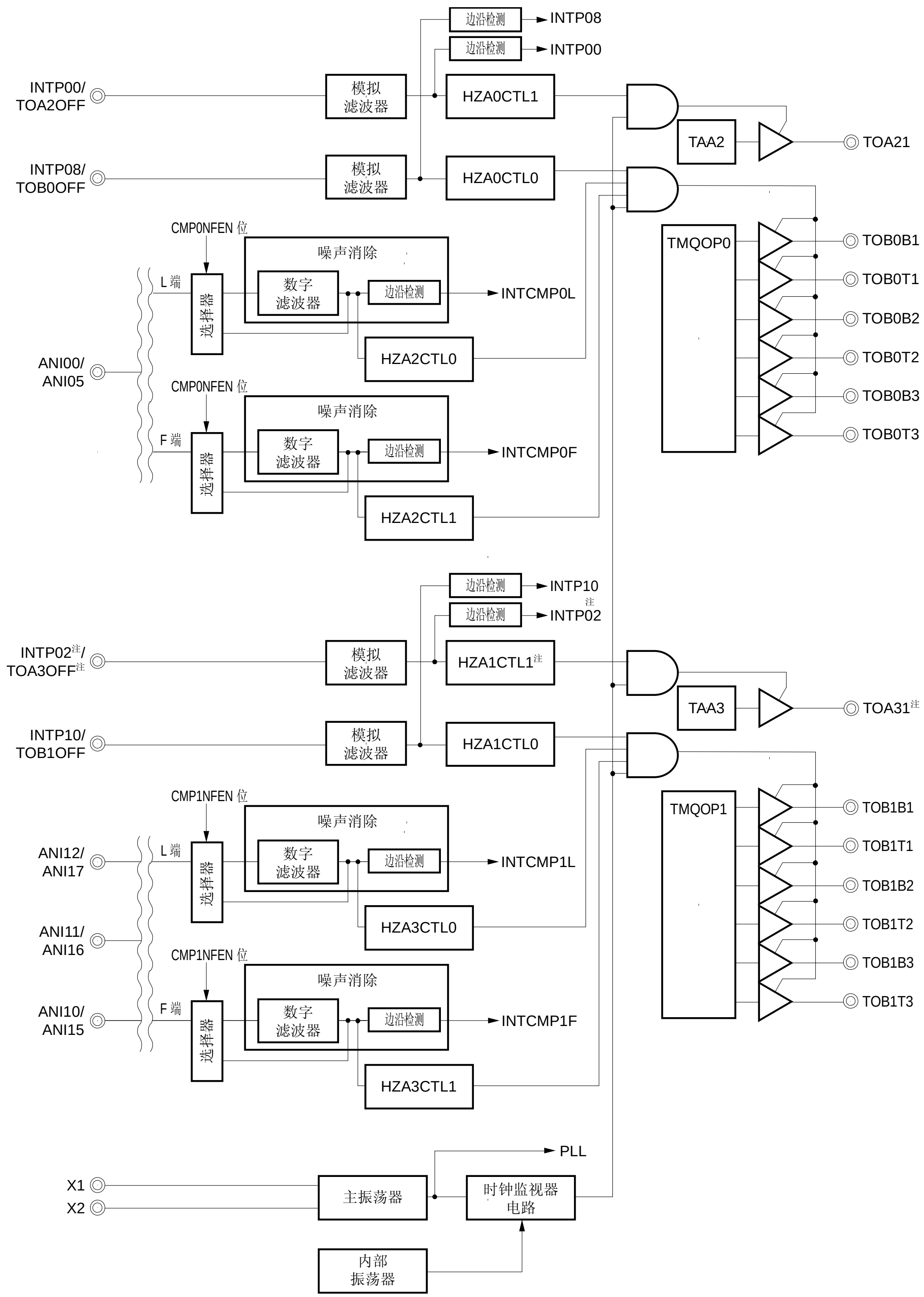
注 • V850E/IF3

HZAyCTL0：TOB0OFF 引脚， HZAyCTL1：TOA2OFF 引脚，
HZA1CTL0：TOB1OFF 引脚，
HZA2CTL0：ANI00/ANI05 引脚， HZA2CTL1：ANI00/ANI05 引脚，
HZA3CTL0：ANI10 至 ANI12, ANI15 至 ANI17 引脚,
HZA3CTL1：ANI10 至 ANI12, ANI15 至 ANI17 引脚

• V850E/IG3

HZAyCTL0：TOB0OFF 引脚， HZAyCTL1：TOA2OFF 引脚，
HZA1CTL0：TOB1OFF 引脚， HZA1CTL1：TOA3OFF 引脚，
HZA2CTL0：ANI00/ANI05 引脚， HZA2CTL1：ANI00/ANI05 引脚，
HZA3CTL0：ANI10 至 ANI12, ANI15 至 ANI17 引脚,
HZA3CTL1：ANI10 至 ANI12, ANI15 至 ANI17 引脚

图 10-4. 高阻抗输出控制器配置



注　　仅限 V850E/IG3

备注　　当参考 **图 10-4** 时，也参考图 **12-3** 和 **12-4**。

（a）设置步骤

（i） 高阻抗控制操作的设置

<1> 设置 HZAyDCMn，HZAyDCNn，以及 HZAyDCPn 位。

<2> 将 HZAyDCEn 位设置为 1（使能高阻抗控制）。

（ii） 在使能高阻抗控制操作后更改设置

<1> 将 HZAyDCEn 位清除为 0（停止高阻抗控制操作）。

<2> 更改 HZAyDCMn，HZAyDCNn，及 HZAyDCPn 位的设置。

<3> 将 HZAyDCEn 位设置为 1（再次使能高阻抗控制操作）。

（iii）引脚处于高阻抗状态时恢复输出

如果 HZAyDCMn 位为 1，则在检测到外部引脚[注]的有效边沿后，将 HZAyDCCn 位设置为 1 以清除高阻抗状态。但是，除非在外部引脚[注]的输入电平无效时设置该位，否则将不能清除高阻抗状态。

<1> 将 HZAyDCCn 位设置为 1（命令信号清除高阻抗状态）。

<2> 读取 HZAyDCFn 位并检测标志状态。

<3> 如果 HZAyDCFn 位为 1，则返回<1>。必须检查外部引脚[注]的输入电平。
如果 HZAyDCFn 位为 0，则引脚可以用作输出引脚。

（iv）通过软件使引脚进入高阻抗状态

当外部引脚[注]的输入电平无效时，必须通过软件将 HZAyDCTn 位设置为 1，使引脚进入高阻抗状态。示例过程如下，其中的设置并不取决于 HZAyDCMn 位的设置。

<1> 将 HZAyDCTn 位设置为 1（高阻抗输出命令）。

<2> 读取 HZAyDCFn 位，检测标志状态。

<3> 如果 HZAyDCFn 位为 0，则返回<1>。必须检查外部引脚[注]的输入电平。
如果 HZAyDCFn 位为 1，则引脚处于高阻抗状态。

但是，如果使用外部引脚[注]时 HZAyDCPn 位和 HZAyDCNn 未被清除为 0，那么在 HZAyDCTn 位被置为 1 时，引脚将进入高阻抗状态。

注
- V850E/IF3
  HZA0CTL0：TOB0OFF 引脚， HZA0CTL1：TOA2OFF 引脚，
  HZA1CTL0：TOB1OFF 引脚，
  HZA2CTL0：ANI00/ANI05 引脚， HZA2CTL1：ANI00/ANI05 引脚，
  HZA3CTL0：ANI10 至 ANI12, ANI15 至 ANI17 引脚，,
  HZA3CTL1：ANI10 至 ANI12, ANI15 至 ANI17 引脚，
- V850E/IG3
  HZA0CTL0：TOB0OFF 引脚， HZA0CTL1：TOA2OFF 引脚，
  HZA1CTL0：TOB1OFF 引脚， HZA1CTL1：TOA3OFF 引脚，
  HZA2CTL0：ANI00/ANI05 引脚， HZA2CTL1：ANI00/ANI05 引脚，
  HZA3CTL0：ANI10 至 ANI12, ANI15 至 ANI17 引脚，,
  HZA3CTL1：ANI10 至 ANI12, ANI15 至 ANI17 引脚，

10.4 操作

10.4.1 系统概述

（1） **6 相 PWM 输出概述**

通过联合使用定时器 ABn 和 TABn 选项，可以使用 6 相 PWM 输出模式生成 6 相 PWM 输出波形。

通过将 TABn 的 TABnCTL1.TABnMD2 至 TABnCTL1.TABnMD0 位设置为“111”，可以使能 6 相 PWM 输出模式。

TABn 的一个 16 位计数器和四个 16 位比较寄存器用于生成基本的 3 相波形。

比较寄存器的功能如下。

TAAn 可以和 TABn 执行调谐操作，来启动 A/D 转换器 0 和 1 的转换触发源。

备注 n = 0, 1

比较寄存器	功能	可设置范围
TABnCCR0 寄存器	设置周期	0002H ≤ m ≤ FFFEH
TABnCCR1 寄存器	指定 U 相的输出宽度	0000H ≤ i ≤ m + 1
TABnCCR2 寄存器	指定 V 相的输出宽度	0000H ≤ j ≤ m + 1
TABnCCR3 寄存器	指定 W 相的输出宽度	0000H ≤ k ≤ m + 1

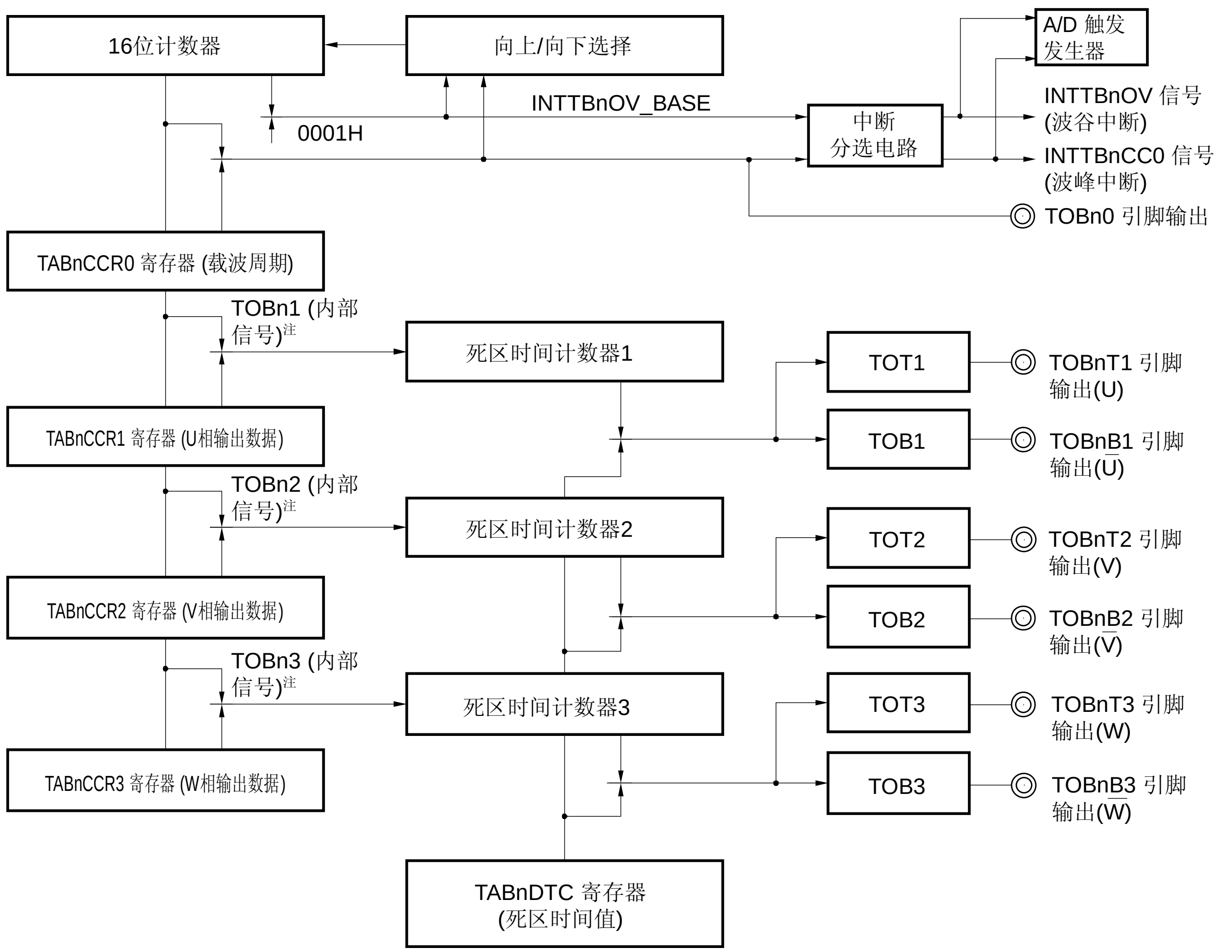
备注 m = TABnCCR0 寄存器的设置值
i = TABnCCR1 寄存器的设置值
j = TABnCCR2 寄存器的设置值
k = TABnCCR3 寄存器的设置值

死区时间间隔从基本 3 相波形中生成，该波形通过使用三个 10 位死区时间计数器和一个比较寄存器来生成，其中比较寄存器用于创建与基本 3 相波形相位相反的波形。然后生成 6 相 PWM 输出波形（U，$\overline{\text{U}}$，V，$\overline{\text{V}}$，W，以及 $\overline{\text{W}}$）。

用于生成基本 3 相波形的 16 位计数器向上计数或向下计数。操作开始后，该计数器向上计数。当它的计数值与 TABnCCR0 寄存器中设置的周期一致时，计数器开始向下计数。当计数值等于 0001H 时，计数器将再次开始向上计数。这意味着载波周期等于两倍的“TABnCCR0 寄存器设置值+ 1”的值。

生成死区时间间隔的 10 位死区时间计数器 1 至 3 向上计数。因此，TABn 死区时间比较寄存器（TABnDTC）中的设置值被用作死区时间值。由于使用了三个计数器，因此可以在 U，V，W 三相独立产生死区时间。但是，由于只有一个指定死区时间值（TABnDTC）的寄存器，因此在三个相中使用相同的死区时间值。

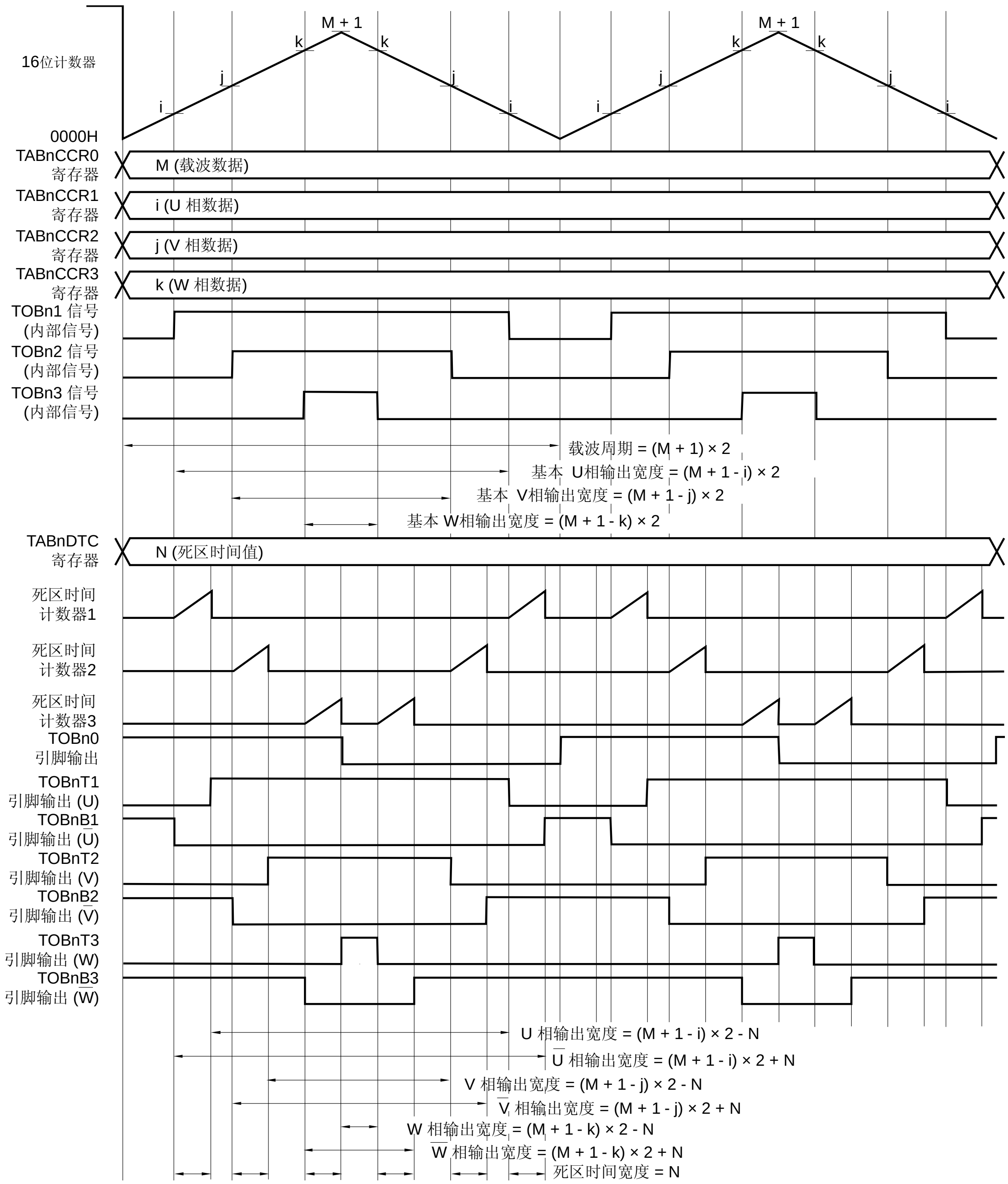
图 10-5. 6 相 PWM 输出模式概况



注　TOBn1、TOBn2 和 TOBn3 是输出引脚的复用功能。

备注　n = 0, 1

图 **10-6. 6** 相 **PWM** 输出模式时序图



注意事项 **1.** 在 **6** 相 **PWM** 输出模式下，**TABnCCR0** 寄存器的值**“M”**应该在 **0002H ≤ M ≤ FFFEH** 的范围内。

2. 只有小于**“M + 1”**的值可以被设置到 **TABnCCR1**，**TABnCCR2**，以及 **TABnCCR3** 寄存器。

3. 如果 **TABnCCR1**，**TABnCCR2** 和 **TABnCCR3** 寄存器被设置为**“0000H”**，输出是 **100%**。如果 **TABnCCR1**，**TABnCCR2** 和 **TABnCCR3** 寄存器被设置为**“M + 1”**，则输出 **0%**。

如果 **TABnCCR1**，**TABnCCR2** 和 **TABnCCR3** 寄存器被设置为**“M + 2”**或更大的值，在 **16** 位计数器的波峰（**M + 1**）处，输出（**50%**占空比）上升，在波谷（**0000H**）处下降。

4. 如果 **U**，**V**，**W** 相的输出宽度的等式（例如（**M + 1 – i**） × **2 – N**）结果为 **0** 或更低，它将会收敛为 **0** （**100%** 输出）。如果结果大于**“**（**M + 1**） × **2”**，它将会收敛为（**M + 1**） × **2** （**0%** 输出）。

备注 n = 0, 1

（2）中断请求

可用的中断请求类型有两种：INTTBnCC0 （波峰中断）信号和 INTTBnOV （波谷中断）信号。
INTTBnCC0 和 INTTBnOV 可以使用 TABnOPT1 寄存器来分选。
关于分选中断的详情参见 **10.4.3 中断分选功能**。

- INTTBnCC0（波峰中断）信号： 该中断信号表示向上计数的 16 位计数器的值与 TABnCCR0 寄存器的值一致
- INTTBnOV（波谷中断）信号： 该中断信号表示向下计数的 16 位计数器的值等于 0001H

（3）定时器操作期间重写寄存器

以下寄存器都有缓冲寄存器，可以用随时重写模式，批量重写模式或间隙批量重写模式进行重写。

相关单元	寄存器
定时器 AAn	TAAn 捕捉 / 比较寄存器 0 （TAAnCCR0） TAAn 捕捉 / 比较寄存器 1 （TAAnCCR1）
定时器 ABn	TABn 捕捉 / 比较寄存器 0 （TABnCCR0） TABn 捕捉 / 比较寄存器 1 （TABnCCR1） TABn 捕捉 / 比较寄存器 2 （TABnCCR2） TABn 捕捉 / 比较寄存器 3 （TABnCCR3）
定时器 Qn 选项	TABn 选项寄存器 1 （TABnOPT1）

备注 n = 0, 1

关于比较寄存器传输功能的详情参见 **10.4.4 使用传输功能重写寄存器的操作**

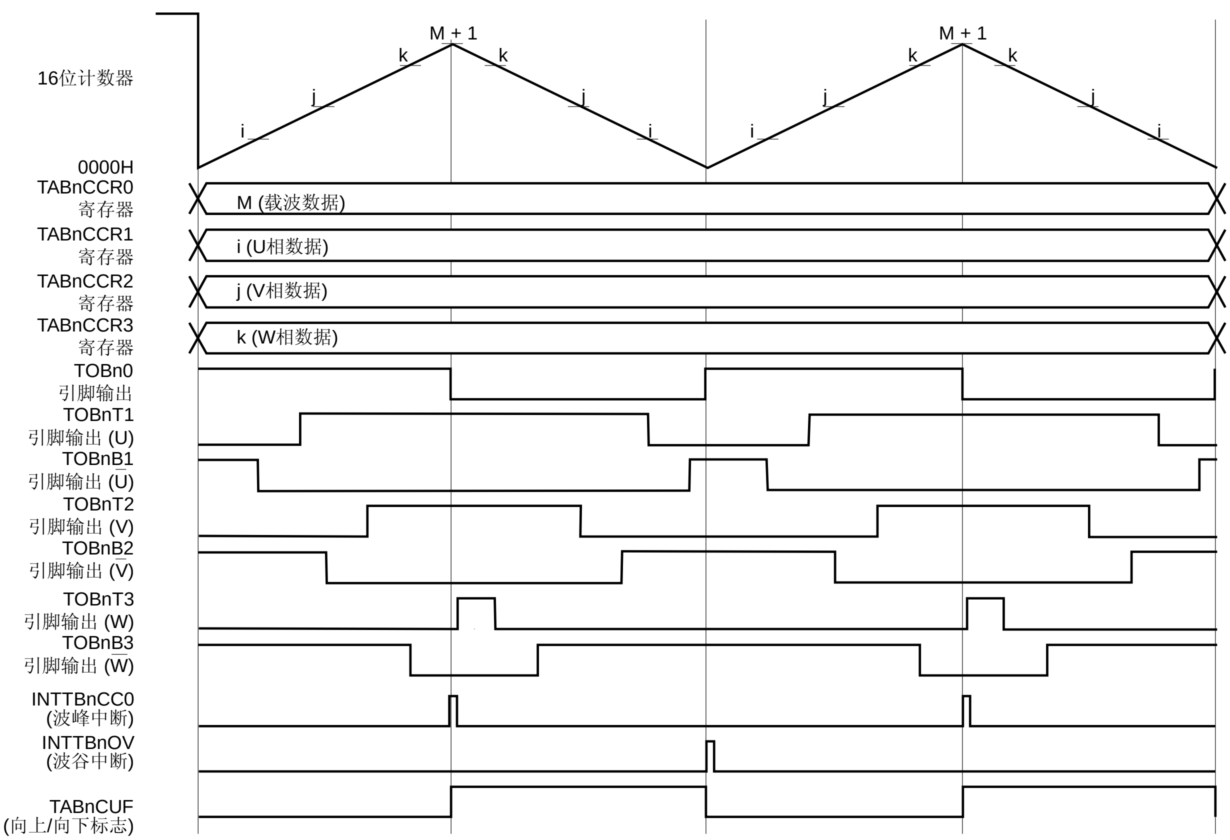
（4） 16 位计数器的向上计数 / 向下计数操作

通过使用 TABn 选项寄存器 0 （TABnOPT0）的 TABnCUF 位可以检查 16 位计数器的操作状态。

TABnCUF 位的状态	16 位计数器的状态	16 位计数器值的范围
TABnCUF 位 = 0	向上计数	0000H – m
TABnCUF 位 = 1	向下计数	（m + 1） – 0001H

备注 **1.** m = TABnCCR0 寄存器的设置值
2. n = 0, 1

图 **10-7.** 中断和向上 / 向下标志



备注 n = 0, 1

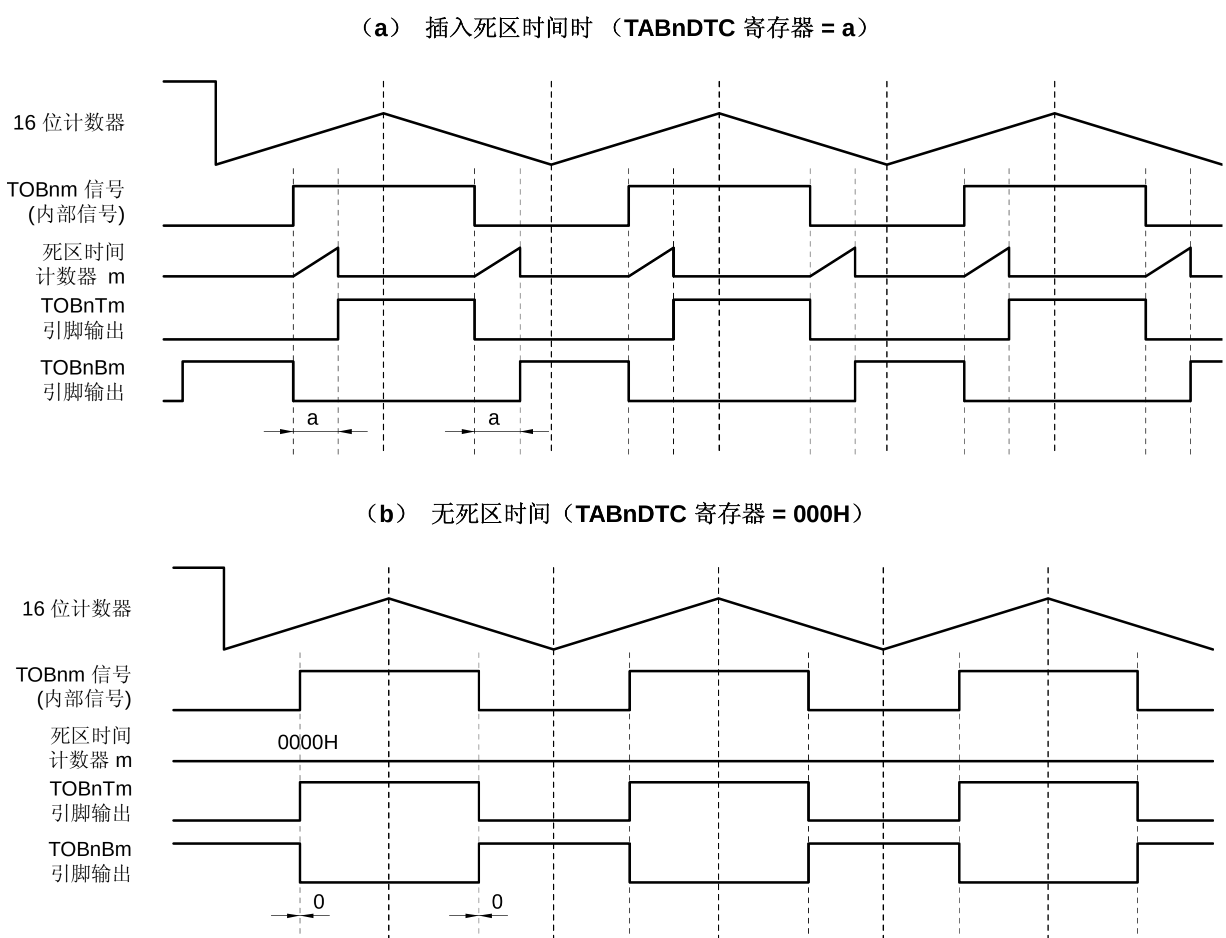
10.4.2 死区时间控制（产生负相位波形信号）

（1） 死区时间控制原理

在 6 相 PWM 输出模式中，比较寄存器 1 至 3（TABnCCR1， TABnCCR2 和 TABnCCR3）用于设置占空比，而比较寄存器 0（TABnCCR0）用于设置周期。通过设置这四个寄存器并启动 TABn 操作，可以生成三种占空比可变的 PWM 输出波形（基本 3 相波形）。这三种 PWM 输出波形被输入到定时器 Qn 选项单元（TMQOPn）中，生成带有死区时间的反转信号以组成三组（六个）PWM 波形。

TMQOPn 单元由三个 10 位计数器（死区时间计数器 1 至 3）组成，这三个计数器与 TABn 的计数时钟同步，通过 TABn 死区时间比较寄存器（TABnDTC）指定死区时间。如果 TABnDTC 寄存器的值被设置为“a”，那么死区时间的值为“a”，此间隔“a”将会出现在正相位波形和负相位波形之间。

图 10-8. 带有死区时间的 PWM 输出波形（1）



备注 n = 0, 1，m = 1 至 3

（2）0%/100%的 PWM 输出

V850E/IF3 和 V850E/IG3 可以实现 PWM 输出的 0%波形输出和 100%波形输出。

从 TOBnTm 引脚输出的连续低电平作为 0%波形输出。从 TOBnTm 引脚中输出的连续高电平作为 100%波形输出。

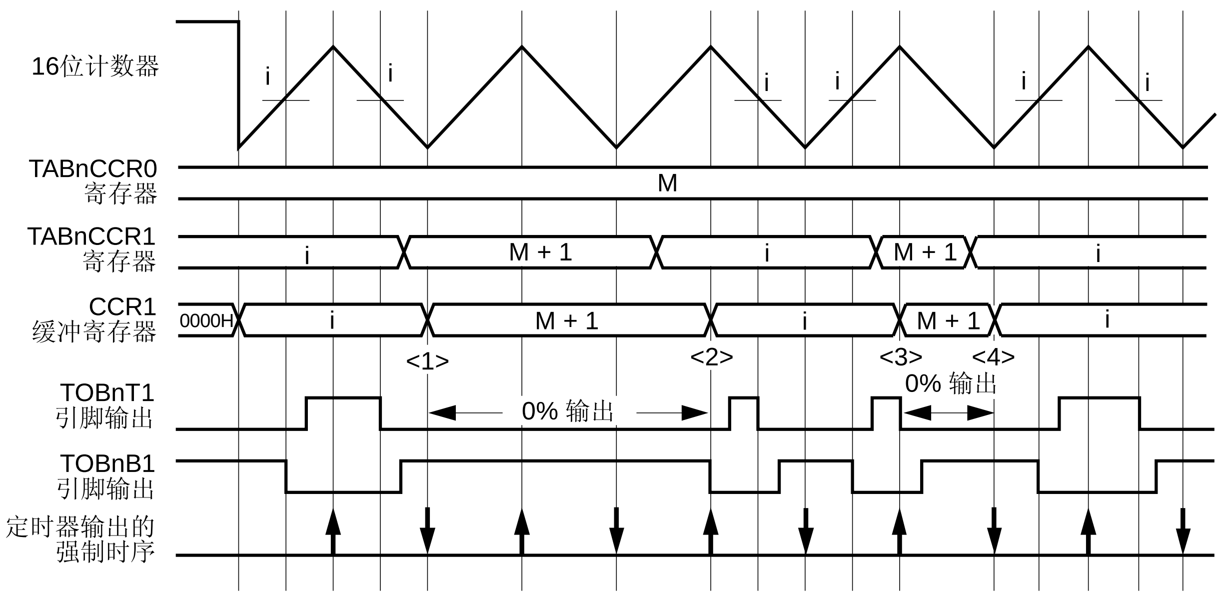
当 TABnCCR0 寄存器 = M 时，将 TABnCCRm 寄存器设为“M + 1”，可以输出 0%波形。

将 TABnCCRm 寄存器设为“0000H”，可以输出 100%波形。

定时器进行操作时，可以对 TABnCCRm 寄存器进行重写，可以在波峰中断（INTTBnCC0）和波谷中断（INTTBnOV）时选择 0%波形输出或 100%波形输出。

备注 n = 0, 1，m = 1 至 3

图 10-9. 0% PWM 输出波形 （带死区时间）



<1> 通过波谷中断来选择 0%输出（不与 16 位计数器一致）
波谷中断强行置低定时器输出。这样会产生 0%输出。

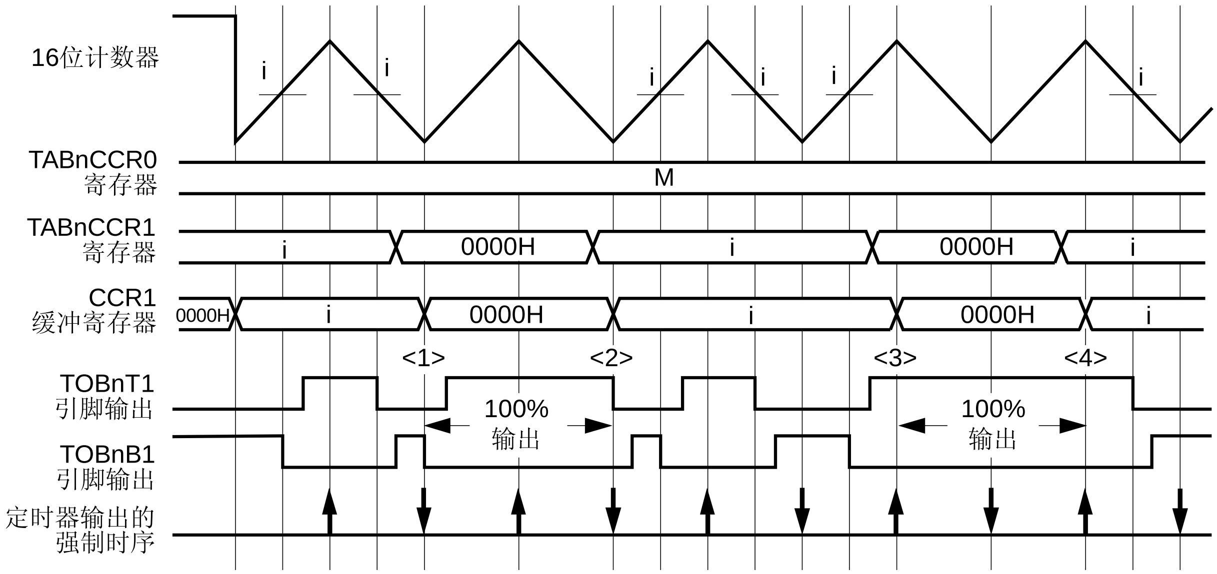
<2> 通过波峰中断来取消 0%输出（不与 16 位计数器一致）
波峰中断强行置高定时器输出。这样会取消 0%输出。

<3> 通过波峰中断来选择 0%输出（与 16 位计数器一致）
波峰中断强行置高定时器输出，但 TABnCCRm 寄存器的值与 16 位计数器的值一致时，会优先拉低定时器输出。结果将会输出 0%波形。

<4> 通过波谷中断来取消 0%输出（不与 16 位计数器一致）
波谷中断强行置低定时器输出。这样会取消 0%输出。

备注 ↑表示强行置高，而↓表示强行置低。

图 **10-10. 100% PWM** 输出波形 （带死区时间）



<1> 通过波谷中断来选择 100%输出（与 16 位计数器一致）
波谷中断强行置低定时器输出，但 TABnCCRm 寄存器的值与 16 位计数器的值一致时，优先置高定时器输出。结果将会产生 100%输出。

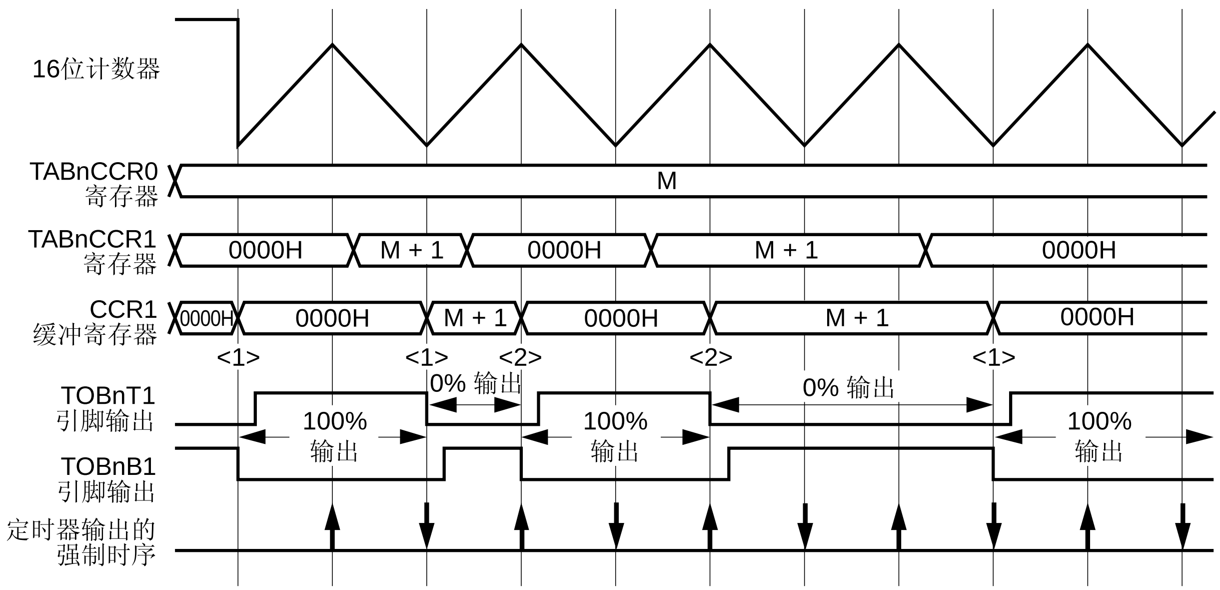
<2> 通过波谷中断来取消 100%输出（不与 16 位计数器一致）
波谷中断强行置低定时器输出。这样会取消 100%输出。

<3> 通过波峰中断来选择 100%输出（不与 16 位计数器一致）
波峰中断强行置高定时器输出。这样会产生 100%输出。

<4> 通过波峰中断来取消 100%输出（不与 16 位计数器一致）
波峰中断强行置高定时器输出。这样会取消 100%输出。

备注 ↑表示强行置高，而↓表示强行置低。

图 **10-11.** 从 **0%** 到 **100%**以及从 **100%** 到 **0%**的 **PWM** 输出波形（带死区时间）



<1> 波谷中断选择 100% ←→ 0% 或 0% ←→ 100%输出。

定时器启动后，立即可以选择 100% ←→ 0% 或 0% ←→ 100%输出。

<2> 波峰中断选择 100% ←→ 0%输出。

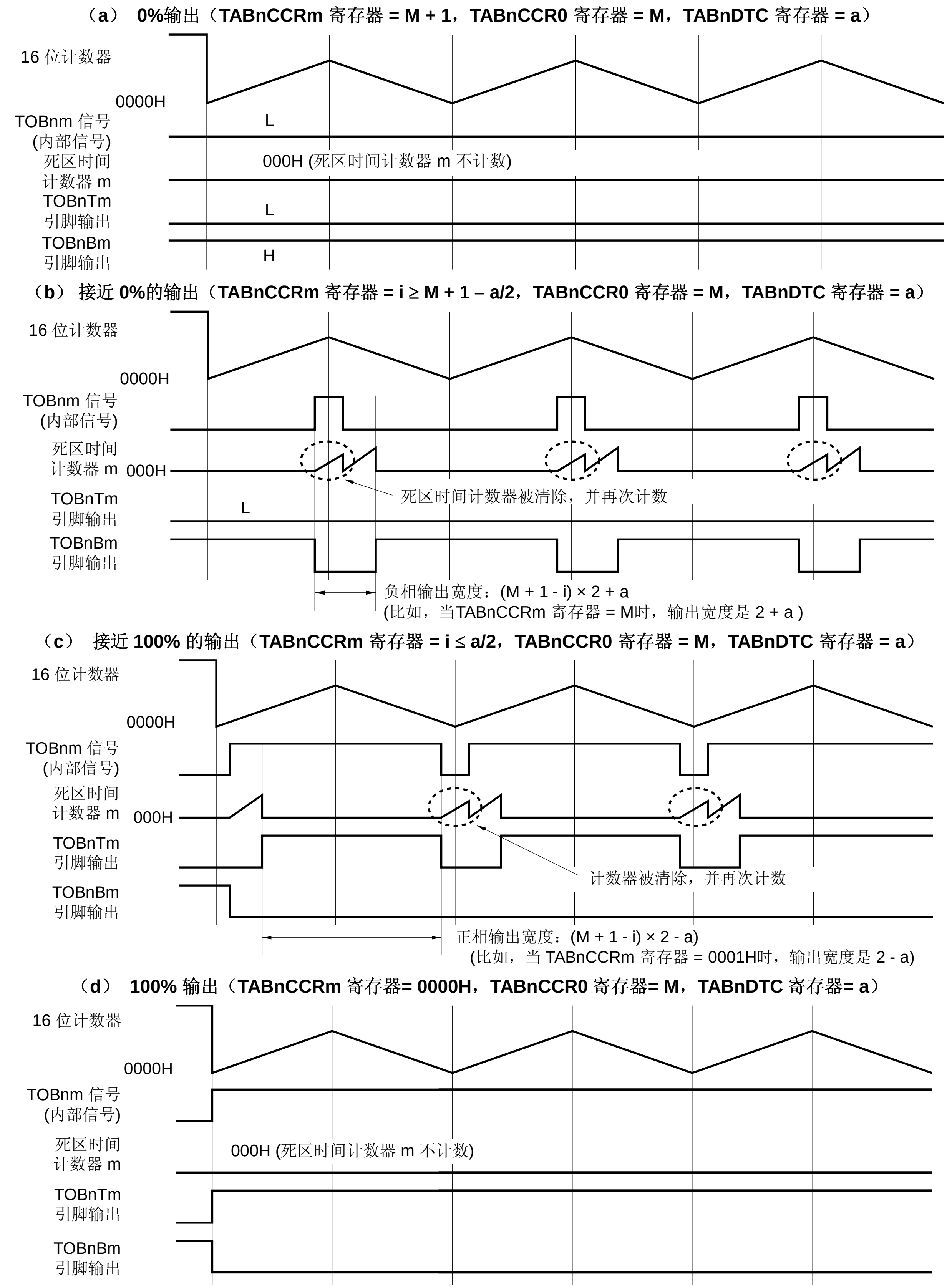
在 16 位计数器值与 TABnCCR0 寄存器值一致时，通过使用定时器输出强行置高功能，波峰中断可以选择 100% ←→ 0%输出。

（3） 接近 0% 和 100%输出的输出波形

当死区时间计数时，16 位计数器的值与比较寄存器的值一致产生中断，那么死区时间计数器将会被清除，并再次开始计数操作。

以下显示了接近 0% 和 100%输出的死区时间控制的输出波形。

图 10-12. 带死区时间的 PWM 输出波形（2）



备注 n = 0, 1，m = 1 至 3

（4） 自动缩小死区时间宽度功能（**TABnOPT2.TABnDTM** 位 = 1）

将 TABnOPT2.TABnDTM 位设置为 1，可以在接近 0%输出或 100%输出时自动缩小死区时间宽度。

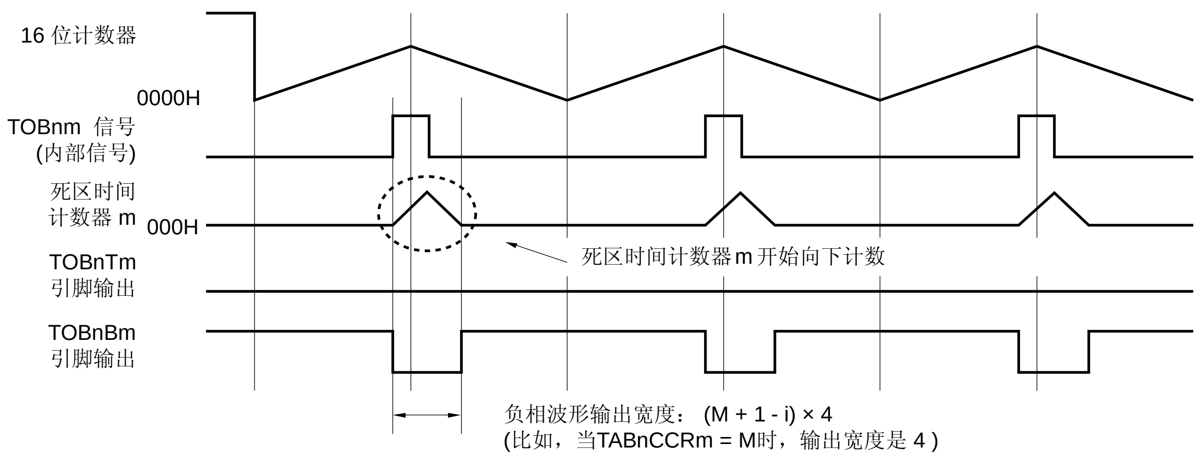
将 TABnDTM 位设置为 1，死区时间计数器不会被清除，但如果在死区时间计数期间改变定时器 AB 的 TOBnm（内部信号）输出，则将会开始向下计数。

以下时序图展示了当 TABnDTM 位设置为 1 时死区计数器的操作。

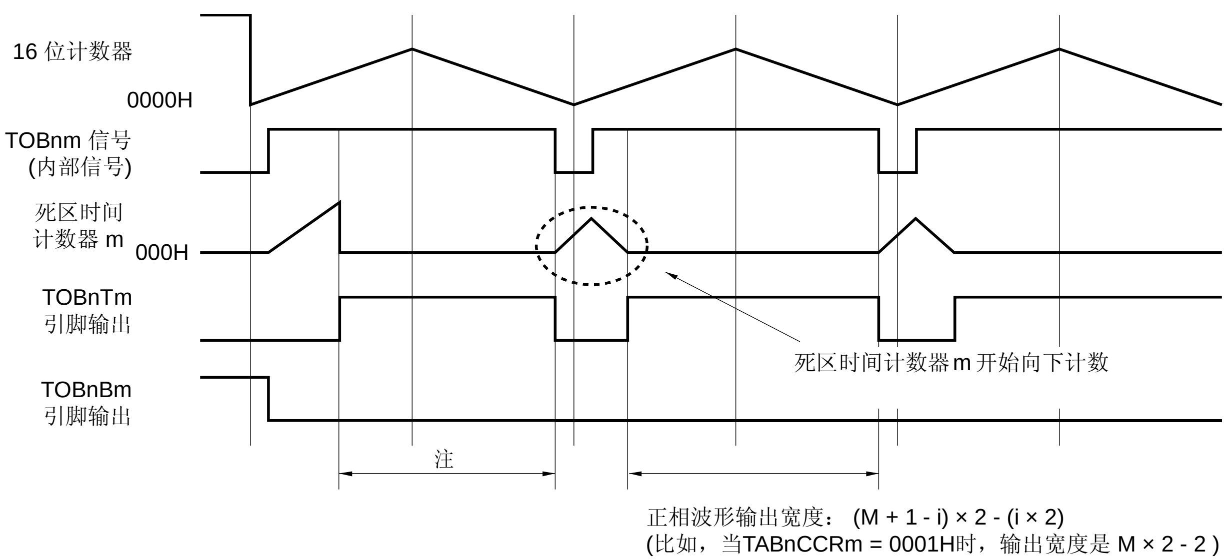
图 10-13. 死区时间计数器 m 的操作（1）

（a） 接近 0%的输出

（TABnCCRm 寄存器 = i ≥ M + 1 – a/2，TABnCCR0 寄存器= M，TABnDTC 寄存器= a）



（b） 接近 100%的输出（TABnCCRm 寄存器 = i ≤ a/2，TABnCCR0 寄存器 = M，TABnDTC 寄存器 = a）



注 设置 TABnCTL0.TABnCE 后第一个波形的输出宽度将会不同于第二个及之后波形的输出宽度。由于要对死区时间进行完整计数，所以第一个波形会短于第二个波形。

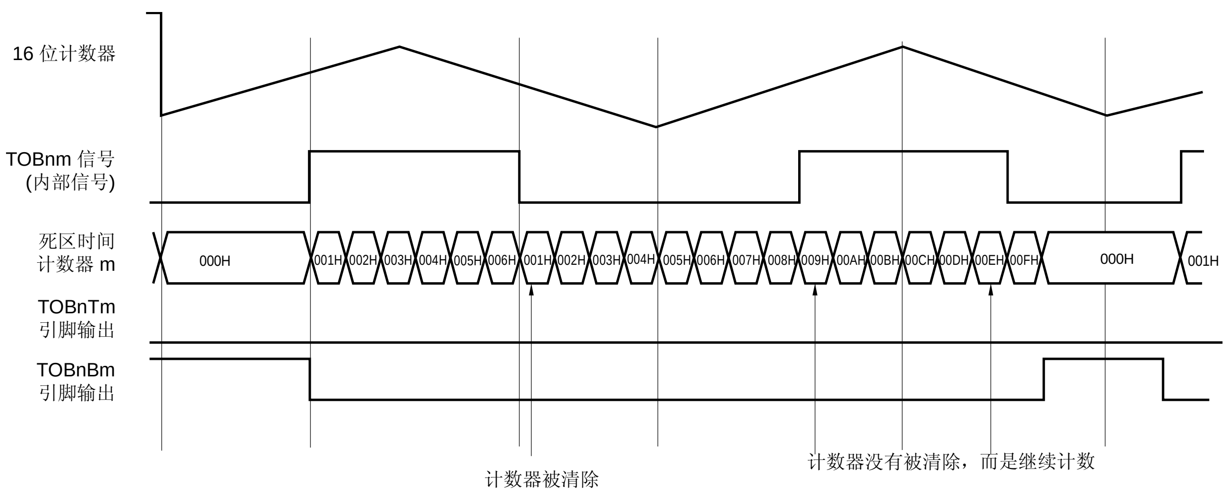
备注 n = 0, 1，m = 1 至 3

（5） 设置错误时的死区时间控制

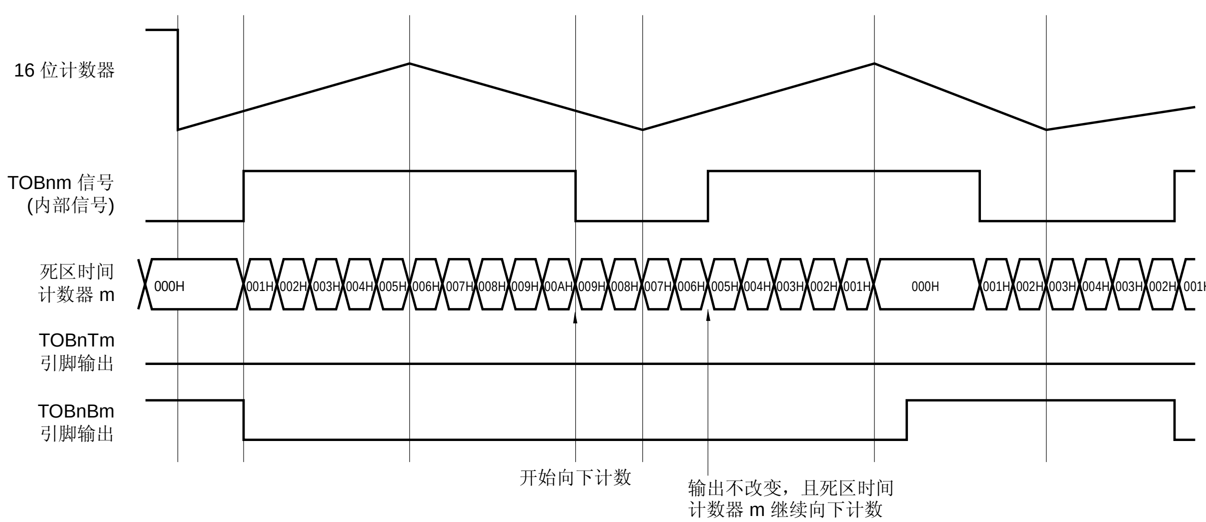
通常情况下，只有在死区时间计数期间接近 0% 和 100%的输出时，TABn 的 TOBnm（内部信号）输出变化一次。本节介绍 TABnCCR0 寄存器（载波周期）和 TABnDTC 寄存器（死区时间值）设置错误的示例。如果这些寄存器设置错误，那么在死区时间计数期间 TABn 的 TOBnm（内部信号）输出的变化多于一次。下面的流程图显示了在这种情况下的 6 相 PWM 输出波形。

图 10-14. 死区时间计数器 m 的操作（2）

（a） 当 TABnOPT2.TABnDTM 位 = 0，TABnCCR0 寄存器 = 0006H，TABnDTC 寄存器= 000FH，TABnCCRm 寄存器= 0004H



（b） 当 TABnOPT2.TABnDTM 位= 1，TABnCCR0 寄存器 = 0006H，TABnDTC 寄存器= 000FH，TABnCCRm 寄存器 = 0002H



备注 n = 0, 1，m = 1 至 3

10.4.3 中断分选功能

- 可以分选的中断有 INTTBnCC0（波峰中断）和 INTTBnOV（波谷中断）。
- TABnOPT1.TABnICE 位用于使能 INTTBnCC0 中断的输出，并且指定中断分选的计数信号。
- TABnOPT1.TABnIOE 位用于使能 INTTBnOV 中断的输出，并且指定中断分选的计数信号。
- TABnOPT1.TABnID4 至 TABnOPT1.TABnID0 位用于指定将要被分选的中断的次数，并且指定中断分选的计数信号。

  在指定的分选计数次数内，中断被屏蔽，在下一次中断时序发生中断。
- TABnOPT2.TABnRDE 位用于指定是否对传输进行分选。

  如果指定对传输进行分选，那么分选后，传输将会与中断输出同时执行。如果指定不对传输进行分选，那么在已经写入 TABnCCR1 寄存器之后的传输时序进行传输。
- TABnOPT0.TABnCMS 位用于指定带有传输功能的寄存器是执行批量重写还是随时重写。

  当 TABnCMS 位为 0 时，这些寄存器的值会在传输时被同步更新。当 TABnCMS 位为 1 时，则新值写入这些寄存器之后立即更新寄存器的值。

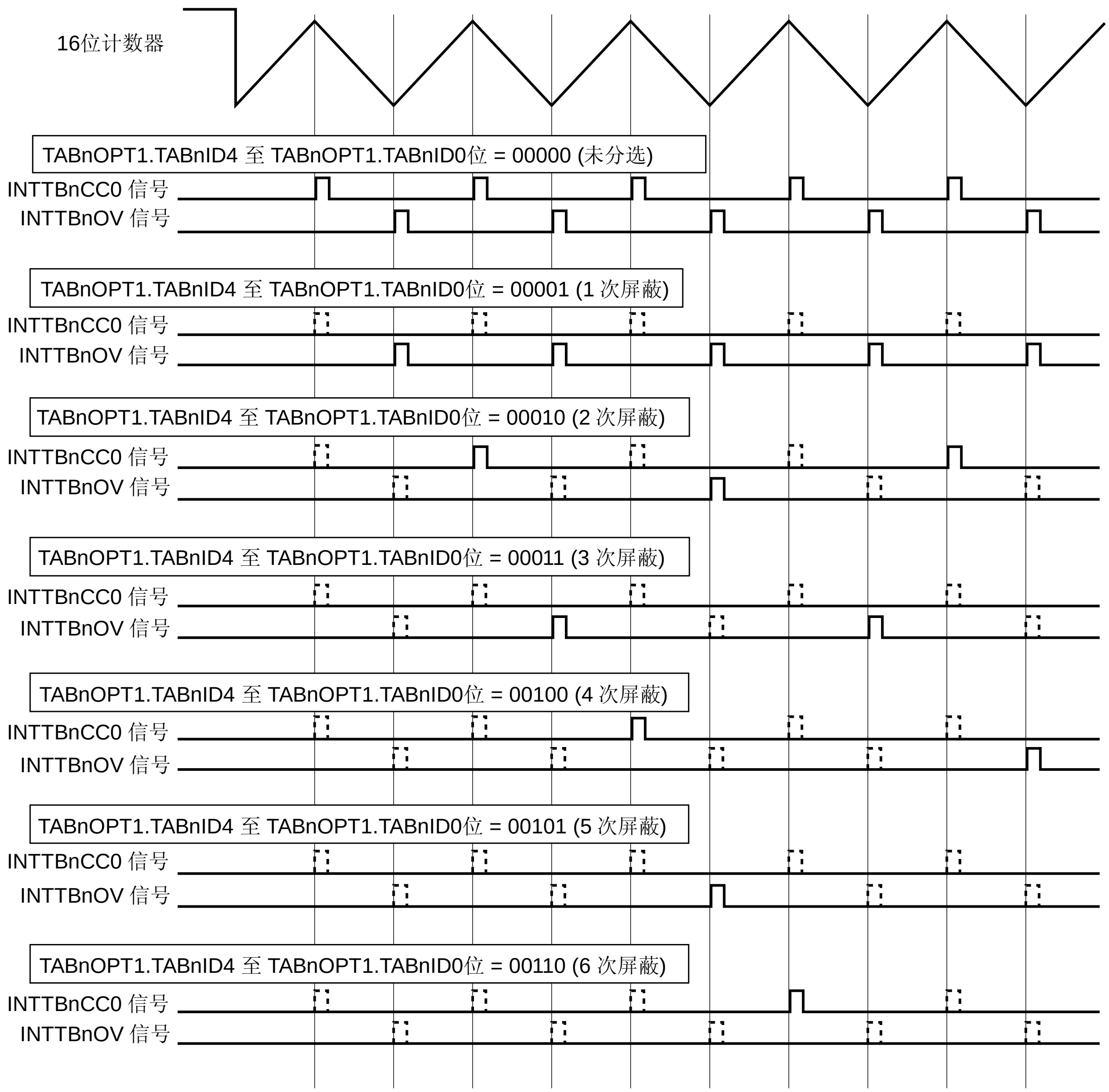
  从 TABnCCRm 寄存器至 CCRm 缓冲寄存器的传输操作与中断分选时序同步进行。

注意事项 1. 在批量重写模式（传输模式）下使用中断分选功能时，将会按间隙批量重写模式（传输分选模式）来执行操作。

2. 分选之后，在该时序产生一个中断。

（1）中断分选操作

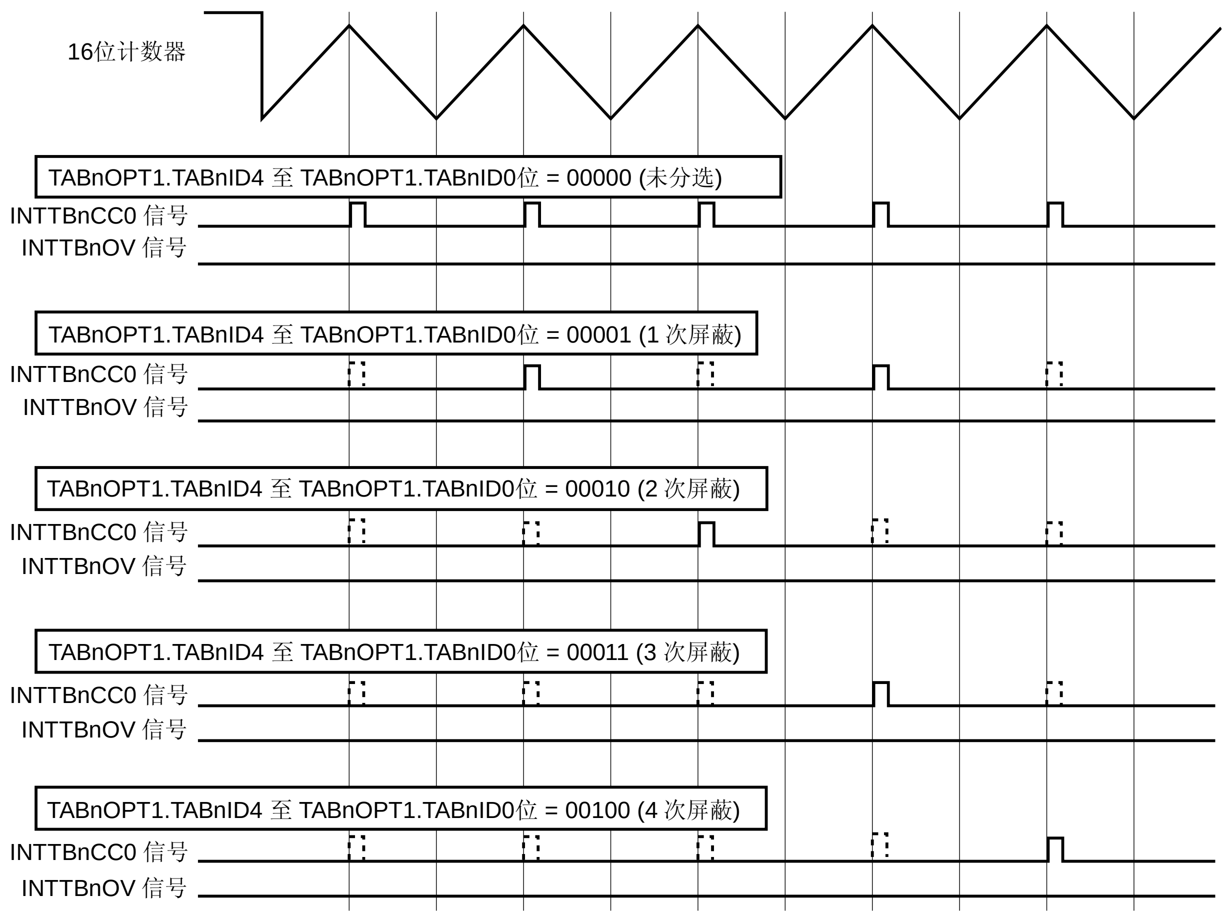
图 10-15. 当 TABnOPT1.TABnICE 位 = 1、TABnOPT1.TABnIOE 位 = 1、TABnOPT2.TABnRDE 位 = 1 （波峰/波谷中断输出）时的中断分选操作



备注 **1.** ⌐¦： 被分选的中断

2. n = 0, 1

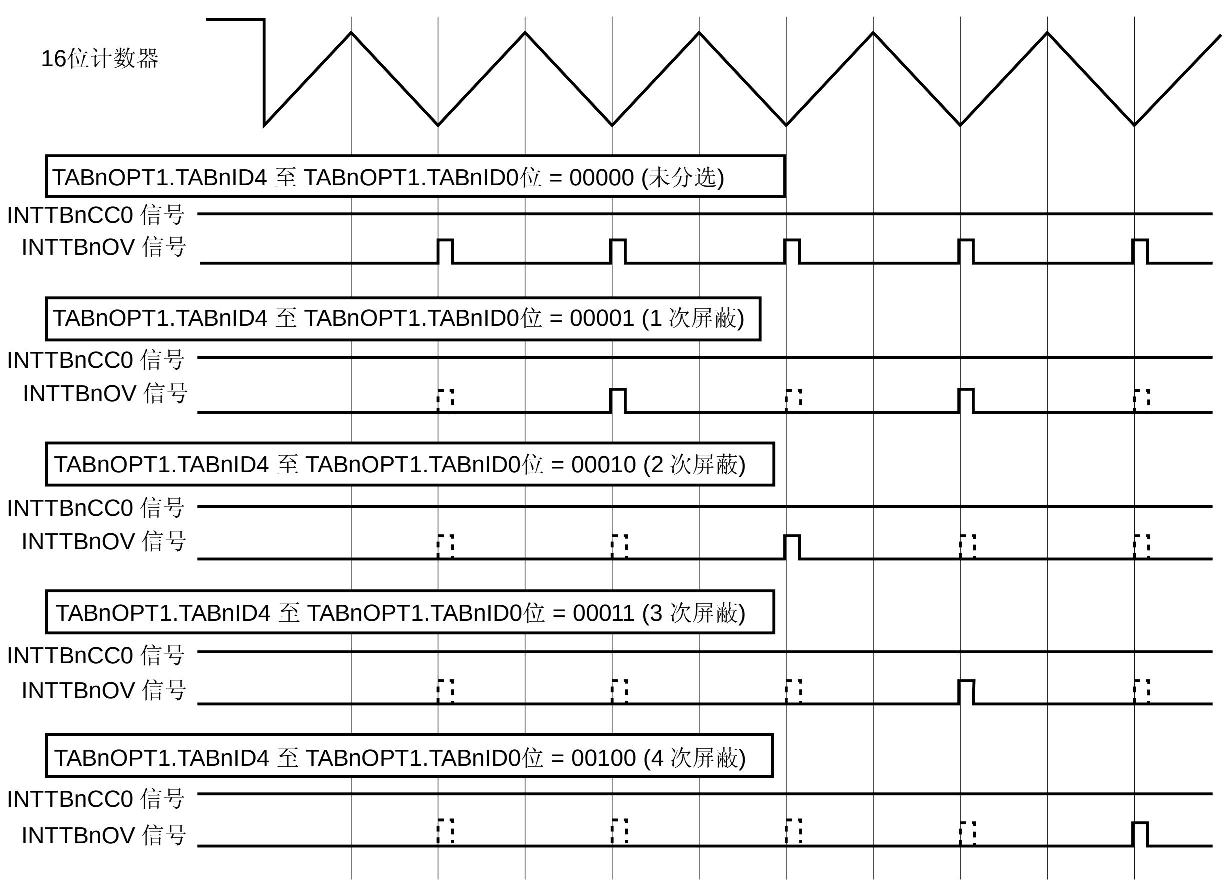
图 **10-16.** 当 **TABnOPT1.TABnICE** 位 **= 1**、**TABnOPT1.TABnIOE** 位 **= 0**、**TABnOPT2.TABnRDE** 位 **= 1** （波峰中断输出）时的中断分选操作



备注 **1.** : 被分选的中断

2. n = 0, 1

图 10-17. 当 TABnOPT1.TABnICE 位 = 0、TABnOPT1.TABnIOE 位 = 1、TABnOPT2.TABnRDE 位 = 1 （波谷中断输出）时的中断分选操作



备注 **1.** ⁚ 被分选的中断

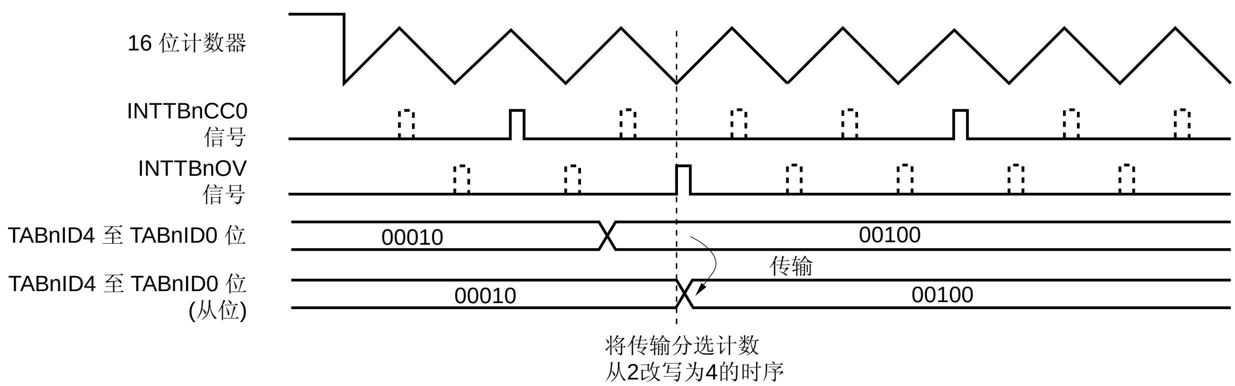
2. n = 0, 1

（**2**）交替输出波峰中断（**INTTBnCC0**）和波谷中断（**INTTBnOV**）

要交替输出波峰中断和波谷中断，将 TABnOPT1.TABnICE 和 TABnOPT1.TABnIOE 位都设置为 1。

图 10-18. 波峰/波谷中断输出

（**a**） **TABnOPT0.TABnCMS 位 = 0，TABnOPT2.TABnRDE 位 = 1** （使用传输分选控制）

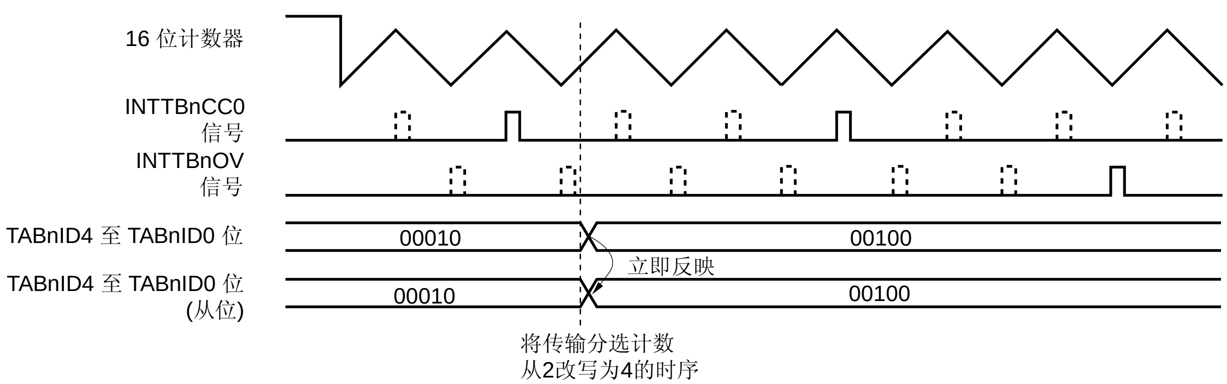


备注 **1.** 当被分选的中断输出时执行传输。忽略其它传输时序。

2. ¦‾¦：被分选的中断

3. n = 0, 1

（**b**） **TABnCMS 位 = 1，TABnRDE 位 = 0 或 1** （不使用传输控制）



备注 **1.** 重写被立即反映。忽略传输时序。

2. ¦‾¦：被分选的中断

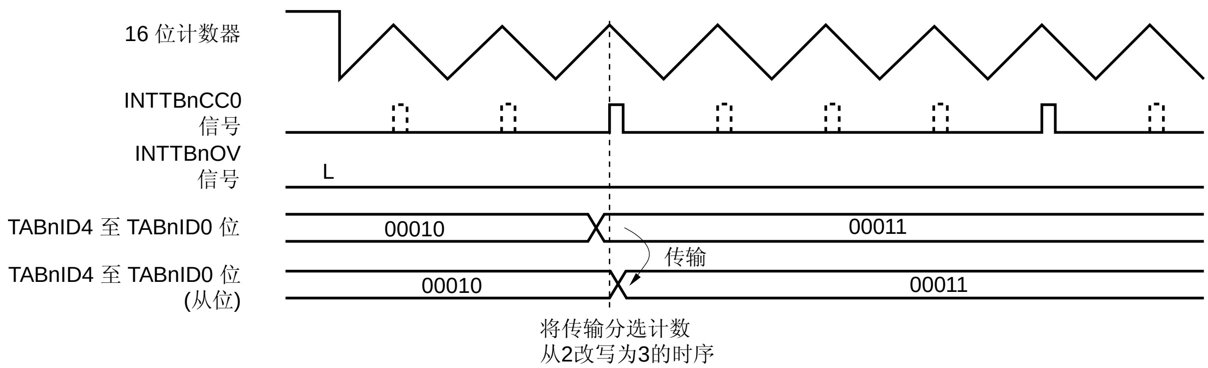
3. n = 0, 1

(3) 仅输出波峰中断(INTTBnCC0)

设置 TABnOPT1.TABnICE 位为 1，且将 TABnOPT1.TABnIOE 位清 0。

图 10-19. 波峰中断输出

(a) TABnOPT0.TABnCMS 位= 0，TABnOPT2.TABnRDE 位= 1(使用传输分选控制)

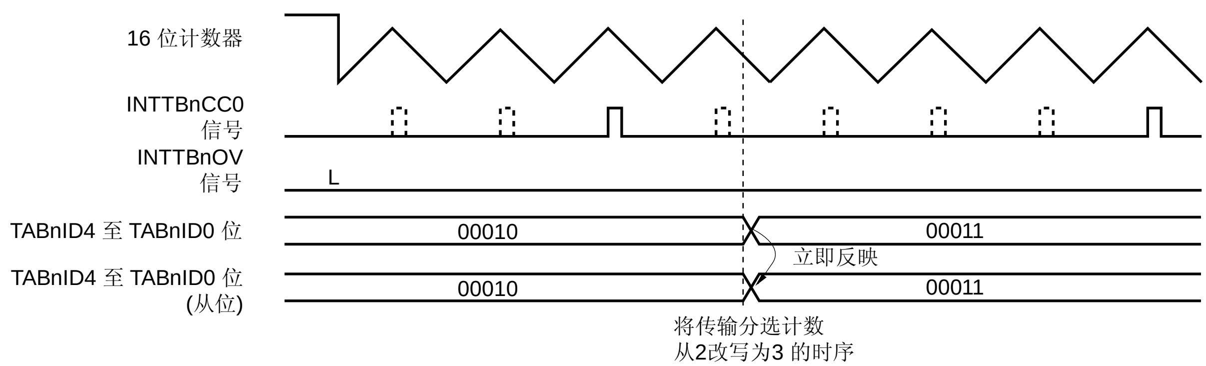


备注 1. 当被分选的中断输出时执行传输。忽略其它传输时序。

2. ¦¦: 被分选的中断

3. n = 0, 1

(b) TABnOPT0.TABnCMS 位 = 1，TABnOPT0.TABnRDE 位= 0 或 1 (不使用传输控制)



备注 1. 重写被立即反映。忽略传输时序。

2. ¦¦: 被分选的中断

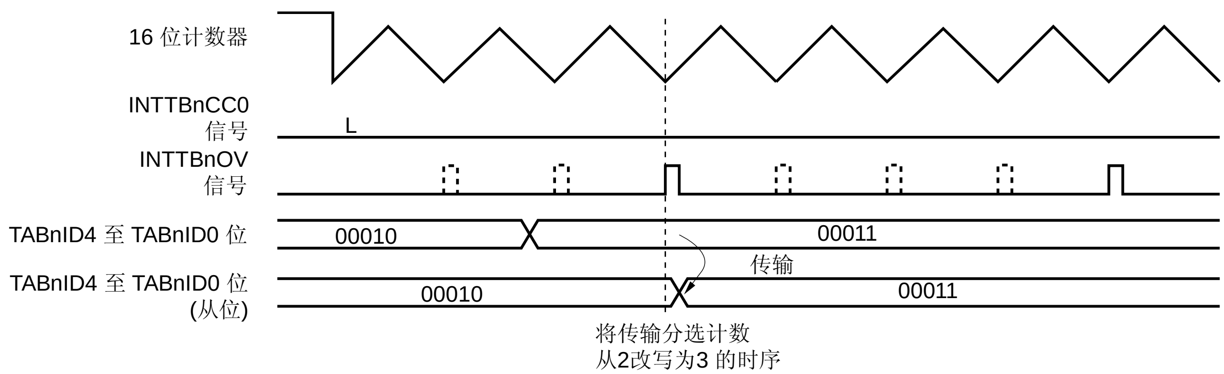
3. n = 0, 1

（4） 仅输出波谷中断（INTTBnOV）

将 TABnOPT1.TABnICE 位清 0 且将 TABnIOE 位置 1。

图 10-20. 波谷中断输出

（a） TABnOPT0.TABnCMS 位= 0，TABnOPT2.TABnRDE 位= 1（使用传输分选控制）

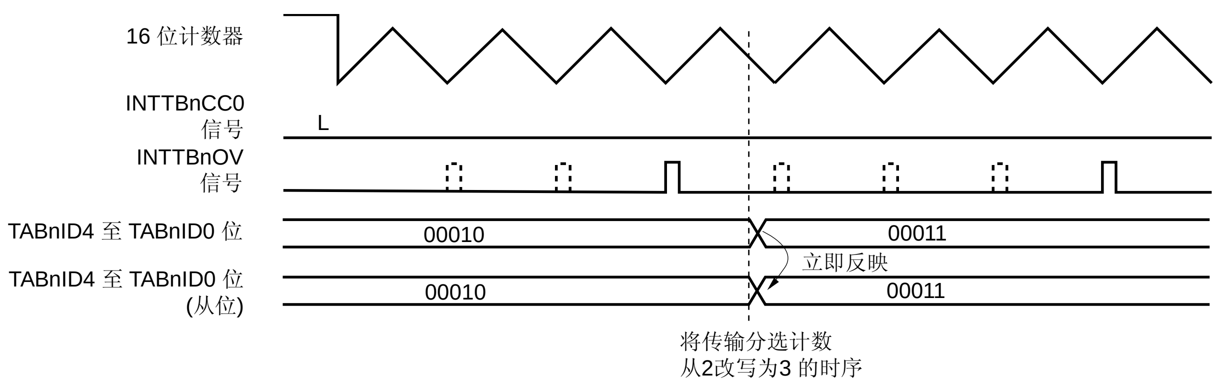


备注 **1.** 当被分选的中断输出时执行传输。忽略其它传输时序。

2. ┆┆：被分选的中断

3. n = 0, 1

（b） TABnOPT0.TABnCMS 位 = 1，TABnOPT0.TABnRDE = 0 或 1（不使用传输控制）



备注 **1.** 重写被立即反映。忽略传输时序。

2. ┆┆：被分选的中断

3. n = 0, 1

10.4.4 重写带有传输功能的寄存器的操作

以下七个寄存器提供有传输功能，且用于控制电机。每个寄存器都有一个缓冲寄存器。

- TABnCCR0：指定 16 位计数器（TAB）周期的寄存器
- TABnCCR1：指定 TOABnT1（U）和 TOABnB1（$\overline{\text{U}}$）占空比的寄存器
- TABnCCR2：指定 TOABnT2（V）和 TOABnB2（$\overline{\text{V}}$）占空比的寄存器
- TABnCCR3：指定 TOABnT3（W）和 TOABnB3（$\overline{\text{W}}$）占空比的寄存器
- TABnOPT1：指定中断分选的寄存器
- TAAnCCR0：指定 A/D 转换启动触发生成时序（调谐操作的 TAAn）的寄存器
- TAAnCCR1：指定 A/D 转换启动触发生成时序（调谐操作的 TAAn）的寄存器

为带有传输功能的寄存器提供以下三种重写模式。

- 随时重写模式
  通过将 TABnOPT0.TABnCMS 位置 1，即可指定该模式。TABnOPT2.TABnRDE 位的设置被忽略。
  在该模式下，每个比较寄存器被独立更新，而比较寄存器的值则会在写入新的值后立即更新。

- 批量重写模式（传输模式）
  通过将 TABnOPT0.TABnCMS 位清 0，将 TABnOPT1.TABnID4 至 TABnOPT1.TABnID0 位设置为 00000，并且将 TABnOPT2.TABnRDE 位清 0，可以指定该模式。
  向 TABnCCR1 寄存器写入数据时，这七个寄存器中的所有数据会在下一个传输时序立即传输到缓冲寄存器。除非 TABnCCR1 寄存器被重写，否则即使重写了其余六个寄存器，也不会执行传输操作。
  不考虑中断，传输时序是每个波峰（16 位计数器值与 TABnCCR0 寄存器的值一致）和波谷（16 位计数器值等于 0001H）的时序。

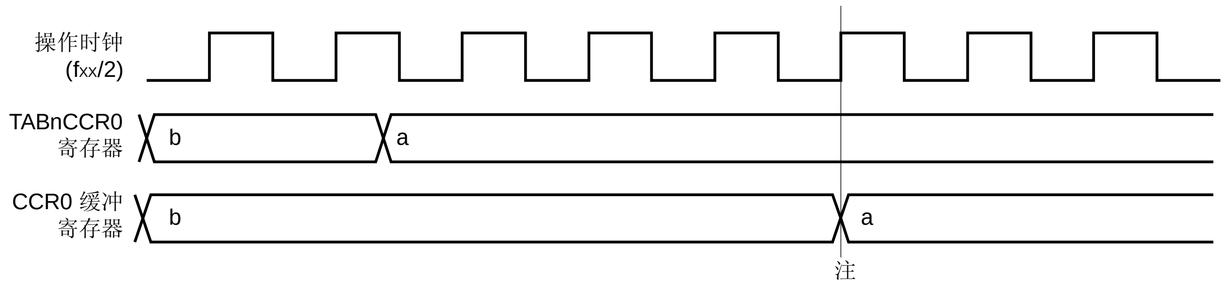
- 间隙批量重写模式（传输分选模式）
  通过将 TABnOPT0.TABnCMS 位清 0 及 TABnOPT2.TABnRDE 位设置 1，可以指定该模式。
  向 TABnCCR1 寄存器写入数据时，这七个寄存器中的所有数据会在下一个传输时序立即传输到缓冲寄存器。除非 TABnCCR1 寄存器被重写，否则即使重写了其余六个寄存器，也不会执行传输操作。
  如果通过 TABnOPT1 寄存器指定了中断分选，和分选中断一样，传输时序也被分选，并且这七个寄存器中的所有数据会在波峰中断（16 位计数器值与 TABnCCR0 寄存器值一致）或波谷中断（16 位计数器值与 0001H 一致）时被立即传输到缓冲寄存器。
  关于中断分选功能的详情参见 **10.4.3 中断分选功能**。

（1）随时重写模式

通过将 TABnOPT0.TABnCMS 位置 1，可以指定该模式。忽略 TABnOPT2.TABnRDE 位的设置。

在该模式下，写入带有传输功能的各个寄存器中的值被立即传输到内部缓冲寄存器中，并与计数值进行比较。如果在该模式下 16 位计数器的计数值与 TABnCCRm 寄存器的值一致之后，重写带有传输功能的寄存器，因为发生第一次一致后的下一次一致被忽略，因此重写值不会被反映。如果在向上计数过程中重写寄存器，那么在计数器开始向下计数后，新的寄存器值生效。

图 10-21. 反映重写值的时序

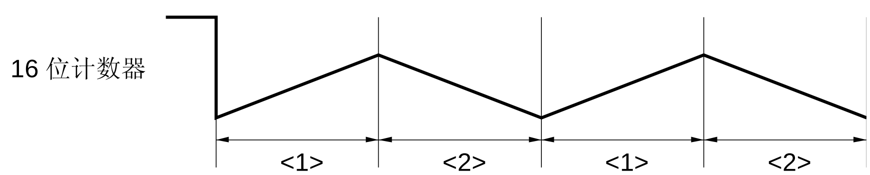


注　在重写寄存器（TABnCCR0, TABnCCR2, TABnCCR3, TABnOPT1, TAAnCCR0 或 TAAnCCR1）后，经过四个操作时钟后，重写值被传输到内部缓冲寄存器。但是，经过 5 个时钟后才会传递 TABnCCR1 寄存器的值。

（a） 重写 TABnCCR0 寄存器

即使在随时重写模式下重写 TABnCCR0 寄存器，在某些情况下，也可能无法反映新值。

图 10-22. 重写 TABnCCR0 寄存器的示例



在时期<1>中重写（向上计数期间重写）

如果新的重写值大于 16 位计数器的值，那么由于它将会与 16 位计数器的值相一致，因此不会有任何问题。如果新的重写值小于 16 位计数器的值，它将不会与计数器的值相一致。结果，16 位计数器将会溢出，并从 0000H 开始继续计数直到它与寄存器的值再次一致为止，无法输出正确的 PWM 波形。

在时期 <2>中重写 （向下计数期间重写）

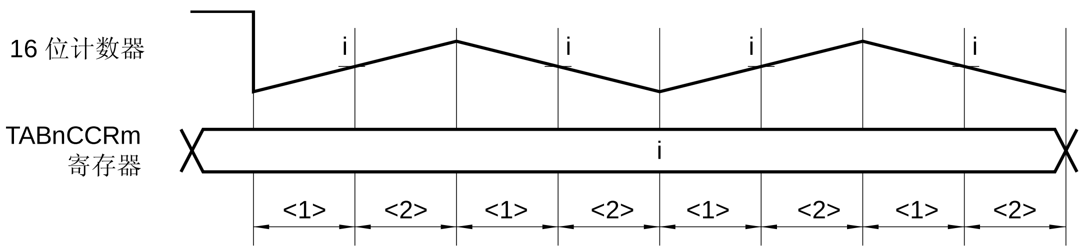
向下计数期间忽略与 16 位计数器的值的一致。因此，重写周期值将会在下个周期开始向上计数时反映为一致点。

（**b**）**重写 TABnCCRm** 寄存器

图 10-24 显示了 16 位计数器的值与 TABnCCRm 寄存器的值一致之前的重写时序（图 10-23 中的<a>），图 10-25 显示了 16 位计数器的值与 TABnCCRm 寄存器的值一致之后的重写时序（图 10-23 中的<b>）。

图 10-23. 16 位计数器与 TABnCCRm 寄存器的基本操作

（**a**） 基本图例



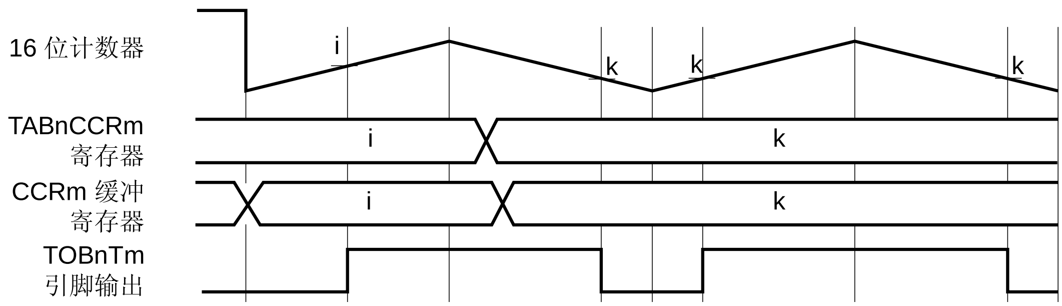
备注 **1.** i = TABnCCRm 寄存器的设置值

2. n = 0, 1，m = 1 至 3

图 10-24. 重写 TABnCCR1 至 TABnCCR3 寄存器的示例（一致发生之前的重写）

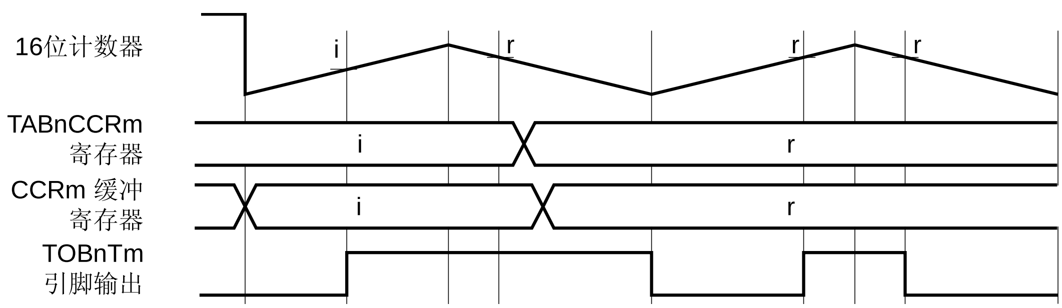
（a）

如果在 TABnCCRm 寄存器的值与 16 位计数器的值一致之前重写该寄存器，那么在寄存器被重写后，该寄存器的值将会与 16 位计数器的值相一致。因此，新的寄存器值将会立即被反映。



（b）

如果写入 TABnCCRm 寄存器的值小于 16 位计数器的值（如果计数器向下计数，则寄存器值大于 16 位计数器值时），那么由于寄存器的值与计数器的值不发生一致，输出波形如下。



如果寄存器的值与计数器的值不发生一致，那么 TOBnTm 引脚输出不会发生变化。即使 16 位计数器的值与 TABnCCRm 寄存器的值不发生一致，TOBnTm 引脚输出也会在发生波峰中断时变为高电平，在发生波谷中断时变为低电平。

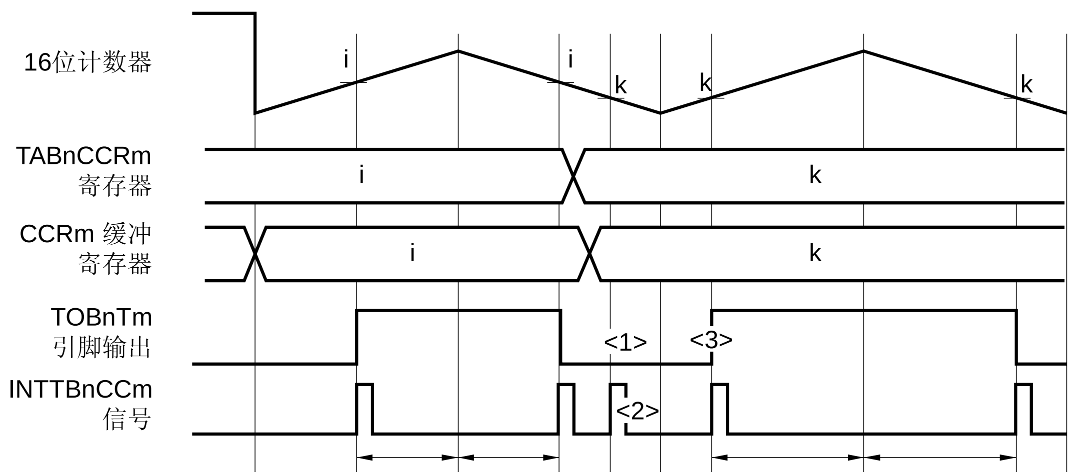
该功能可以提供 0% 输出和 100%输出。

详情参见 **10.4.2 （2） 0%/100%的 PWM** 输出。

备注 **1.** i， r， k = TABnCCRm 寄存器的设置值

2. n = 0, 1，m = 1 至 3

图 **10-25. 重写 TABnCCR1 至 TABnCCR3** 寄存器的示例 （一致发生之后的重写）



<1> 在产生一致信号后，重写寄存器导致 16 位计数器的计数值与 TABnCCRm 寄存器的值发生一致，忽略这个一致，且 PWM 输出不会发生变化。

<2> 即使 PWM 输出没有变化，也会输出由 16 位计数器值与 TABnCCRm 寄存器相一致而产生的中断（INTTBnCCm）。

<3> 在计数器的计数方向改为向上或向下后，16 位计数器与 TABnCCRm 寄存器的下一次一致将有效，且 PWM 输出会发生变化。

如果 TABnCCRm 寄存器与 16 位计数器的值一致之后，该寄存器的值被重写，那么将会忽略第一次一致后产生的一致，且重写值不会反映至 TOABnTm 引脚输出。如果在计数器向下计数时重写该寄存器，那么在计数器开始向下计数后产生的一致有效（如果在计数器向上计数时重写寄存器，那么在计数器开始向上计数后产生的一致有效）。

备注 **1.** i， r， k = TABnCCRm 寄存器的设置值

2. n = 0, 1，m = 1 至 3

（**c**） **重写 TABnOPT1 寄存器**

写入 TABnOPT1 寄存器时，中断分选计数器被清除。中断分选计数器被清除后，丢弃已经检测到的中断发生次数。结果，中断产生间隔会暂时延长。

为了避免该操作，应在间隙批量重写模式（传输分选模式）下重写 TABnOPT1 寄存器。

关于重写 TABnOPT1 寄存器的详情参见 **10.4.3 中断分选功能**。

（2）批量重写模式（传输模式）

通过将 TABnOPT0.TABnCMS 位清 0，将 TABnOPT1.TABnID4 至 TABnOPT1.TABnID0 位设为 00000，且将 TABnOPT2.TABnRDE 位清 0，可以指定该模式。

在该模式下，写入带有传输功能的各个寄存器中的值被立即传输到内部缓冲寄存器，并与计数器的值进行比较。

（a） 重写步骤

如果向 TABnCCR1 寄存器写入数据，那么在下个传输时序，设置在 TABnCCR0 至 TABnCCR3，TABnOPT1，TAAnCCR0 和 TAAnCCR1 寄存器中的值全部被立即传输到内部缓冲寄存器中。因此，应该最后写入 TABnCCR1 寄存器。产生传输时序之前，在写入 TABnCCR1 寄存器之后禁止对其他寄存器进行写操作（直到波峰（16 位计数器值与 TABnCCR0 寄存器值间的一致）或波谷（16 位计数器值等于 0001H））。操作步骤如下。

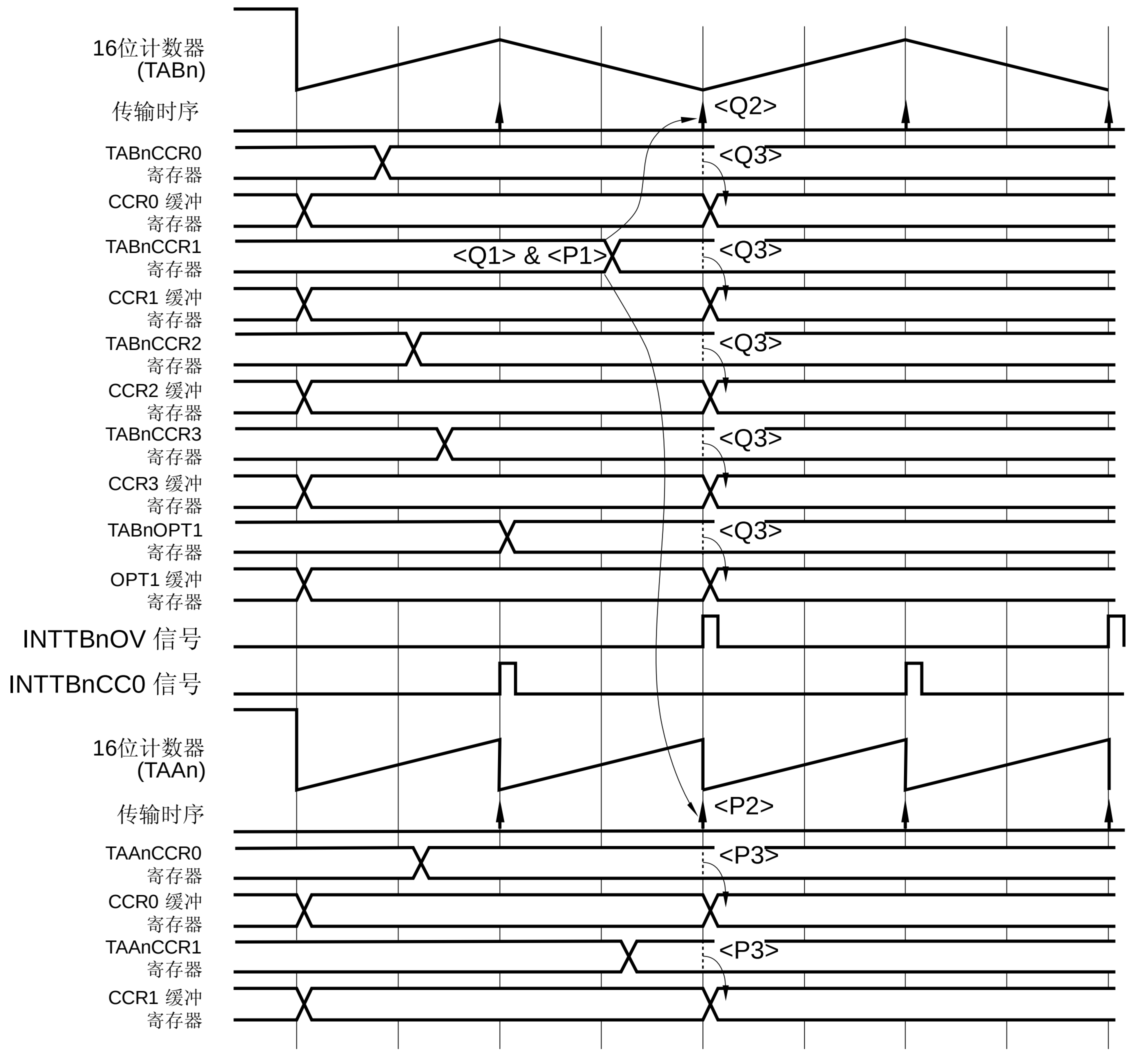
<1> 对 TABnCCR0, TABnCCR2, TABnCCR3, TABnOPT1, TAAnCCR0 和 TAAnCCR1 寄存器进行重写。
对于无需重写的寄存器，不要重写。

<2> 重写 TABnCCR1 寄存器。
即使无需重写 TABnCCR1 寄存器，也应该将相同的值重写到该寄存器。

<3> 保持下次重写挂起直至生成传输时序。
在产生 INTTBnOV 或 INTTBnCC0 中断后，对寄存器再次进行重写。

<4> 返回步骤 <1>。

图 10-26. 批量写入模式中的基本操作



[TABn 的操作]
<Q1> 写入 TABnCCR1 寄存器
<Q2> 在对 TABnCCR1 寄存器执行写操作后，目标时序是第一个传输时序。
<Q3> 在传输时序处立即传输数值。

[TAAn 的操作]
<P1> 写入 TABnCCR1 寄存器
<P2> 在对 TABnCCR1 寄存器执行写操作后，目标时序是第一个传输时序。
<P3> 在传输时序处立即传输数值。

（b） 重写 TABnCCR0 寄存器

在批量重写模式下对 TABnCCR0 寄存器进行重写时，根据传输是发生在波峰（16 位计数器值与 TABnCCR0 寄存器值间的一致）发生还是在波谷（16 位计数器值等于 0001H），输出波形会有所不同。通常情况下，推荐在 16 位计数器进行向下计数时重写 TABnCCR0 寄存器，并在波峰时序的传输时序处传输寄存器的值。

图 10-28 展示了在 16 位计数器向上计数时（图 10-27 中的时期<1>）重写 TABnCCR0 寄存器的示例。图 10-29 显示了在 16 位计数器向下计数时（图 10-27 中的时期 <2>）重写 TABnCCR0 寄存器的示例。

图 10-27. 16 位计数器的基本操作

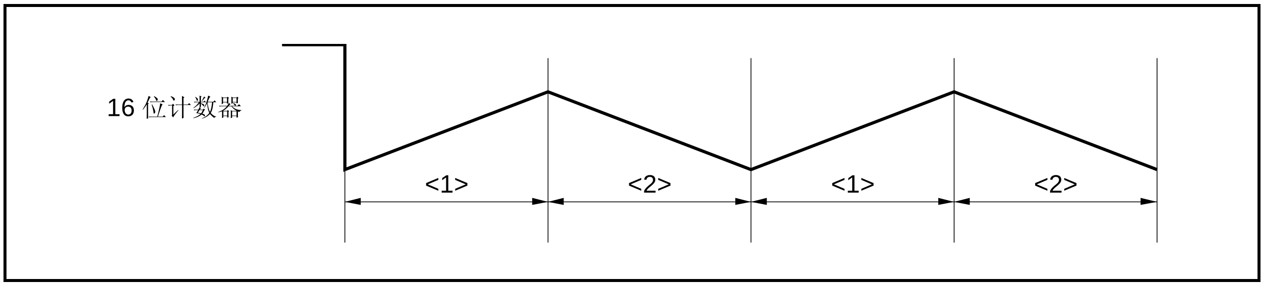
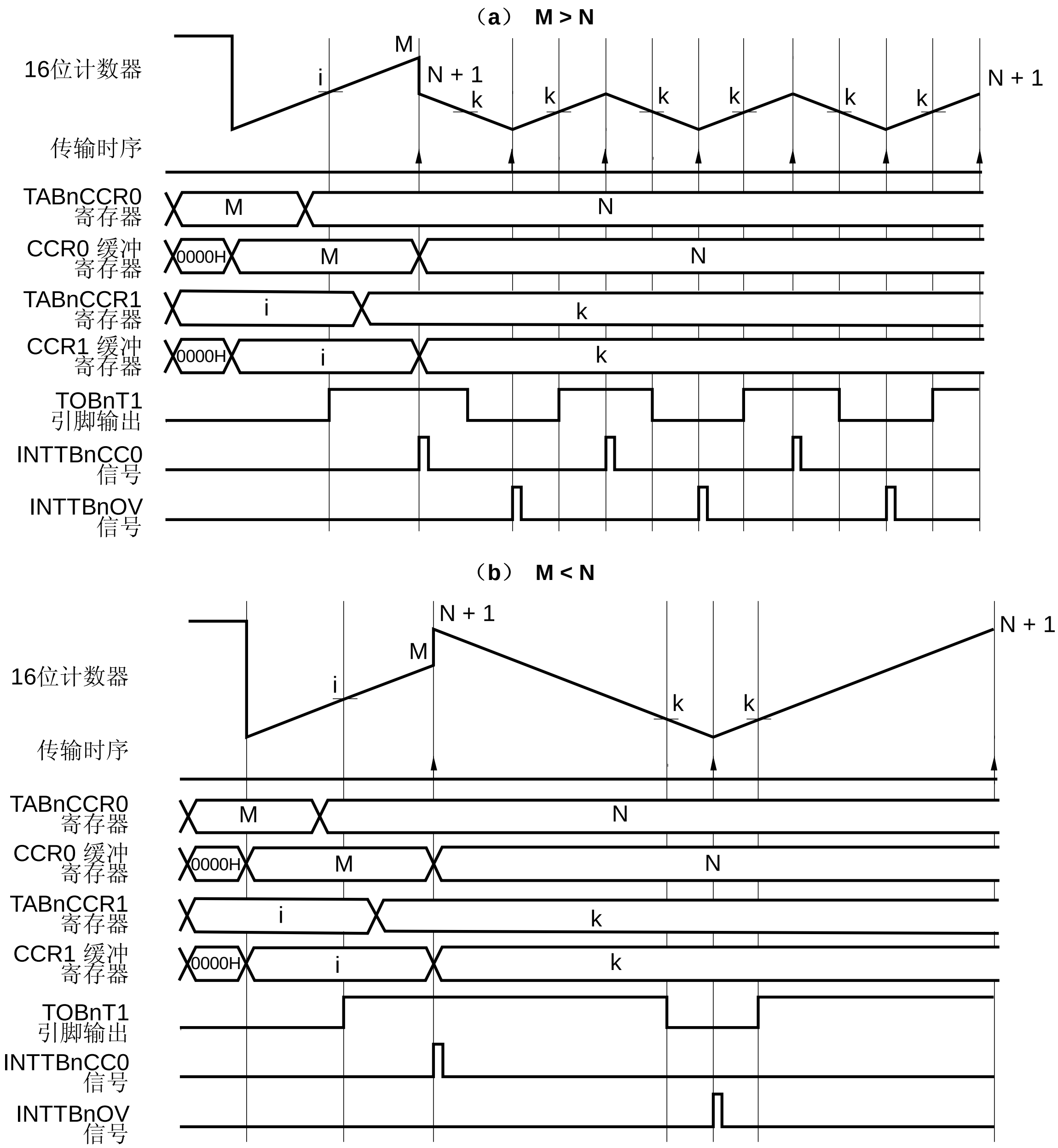


图 10-28 中的传输时序发生在波峰时序的位置。当 16 位计数器向下计数时，周期会发生变化并输出不对称的三角波。由于改变了周期，所以要重写占空比（电压数据值）。

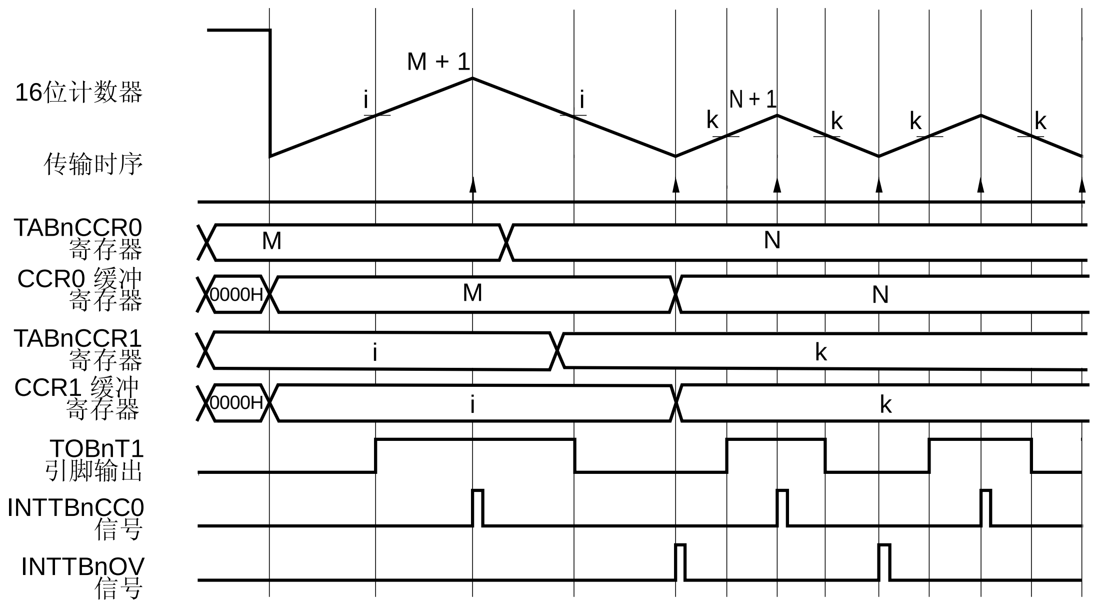
图 10-28. 重写 TABnCCR0 寄存器示例（向上计数期间）



备注 **1.** 如果在 6 相 PWM 输出模式下进行传输（16 位计数器的值与 CCR0 缓冲寄存器的值一致），那么 TABnCCR0 寄存器的值加 1 的结果会装载到 16 位计数器中。这种情况下，即使在波峰（16 位计数器的值与 TABnCCR0 寄存器的值一致）时序的传输时序处周期值发生变化，也能输出预期的波形。

2. M：重写之前 CCR0 缓冲寄存器的数值
N：重写之后 CCR0 缓冲寄存器的数值

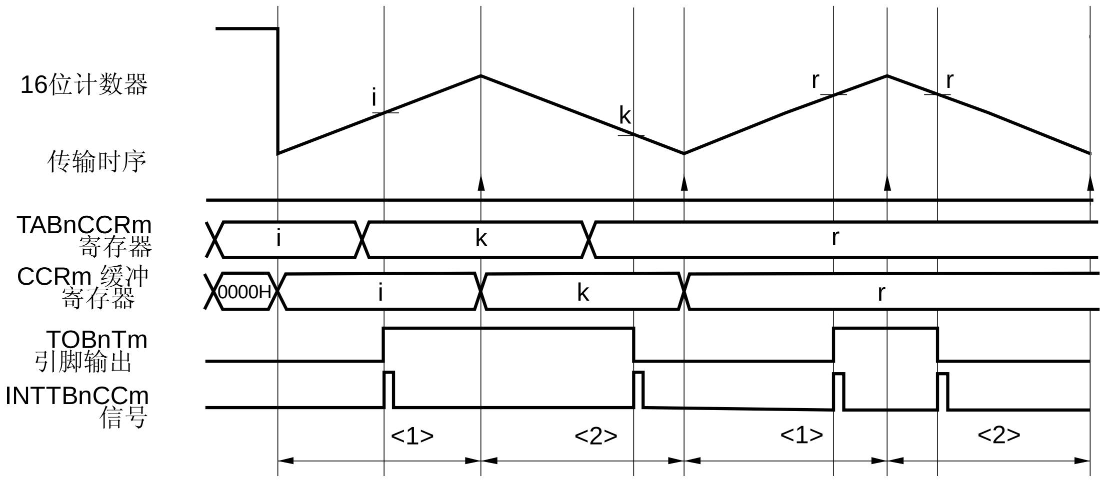
图 10-29. 重写 TABnCCR0 寄存器的示例（向下计数期间）



由于下个传输时序处于波谷（16 位计数器的值与 0001H 一致），因此周期值将从下个周期开始变化，且能够保持对称的三角波输出。由于周期发生变化，应该根据实际需要来重写占空比（电压数据值）。

（**c**）重写 **TABnCCRm** 寄存器

图 **10-30.** 重写 **TABnCCRm** 寄存器的示例



在期间 <1>重写 （向上计数期间重写）

由于传输 TABnCCRm 寄存器值发生在波峰（16 位计数器的值与 TABnCCR0 寄存器的值一致）的传输时序处，因此将输出不对称的三角波。

在期间 <2>重写 （向下计数期间重写）

由于传输 TABnCCRm 寄存器值发生在波谷（16 位计数器的值与 0001H 一致）的传输时序处，因此将输出对称的三角波。

备注 m = 1 至 3

（**d**）传输 **TABnOPT1** 寄存器的值

不要将 TABnOPT1.TABnID4 至 TABnOPT1.TABnID0 位设置为 00000 以外的值。当使用中断分选功能时，在间隙批量重写模式（传输分选模式）中重写 TABnOPT1 寄存器。

关于重写 TABnOPT1 寄存器的详情，参见 **10.4.3 中断分选功能**。

（3）间隙批量重写模式（传输分选模式）

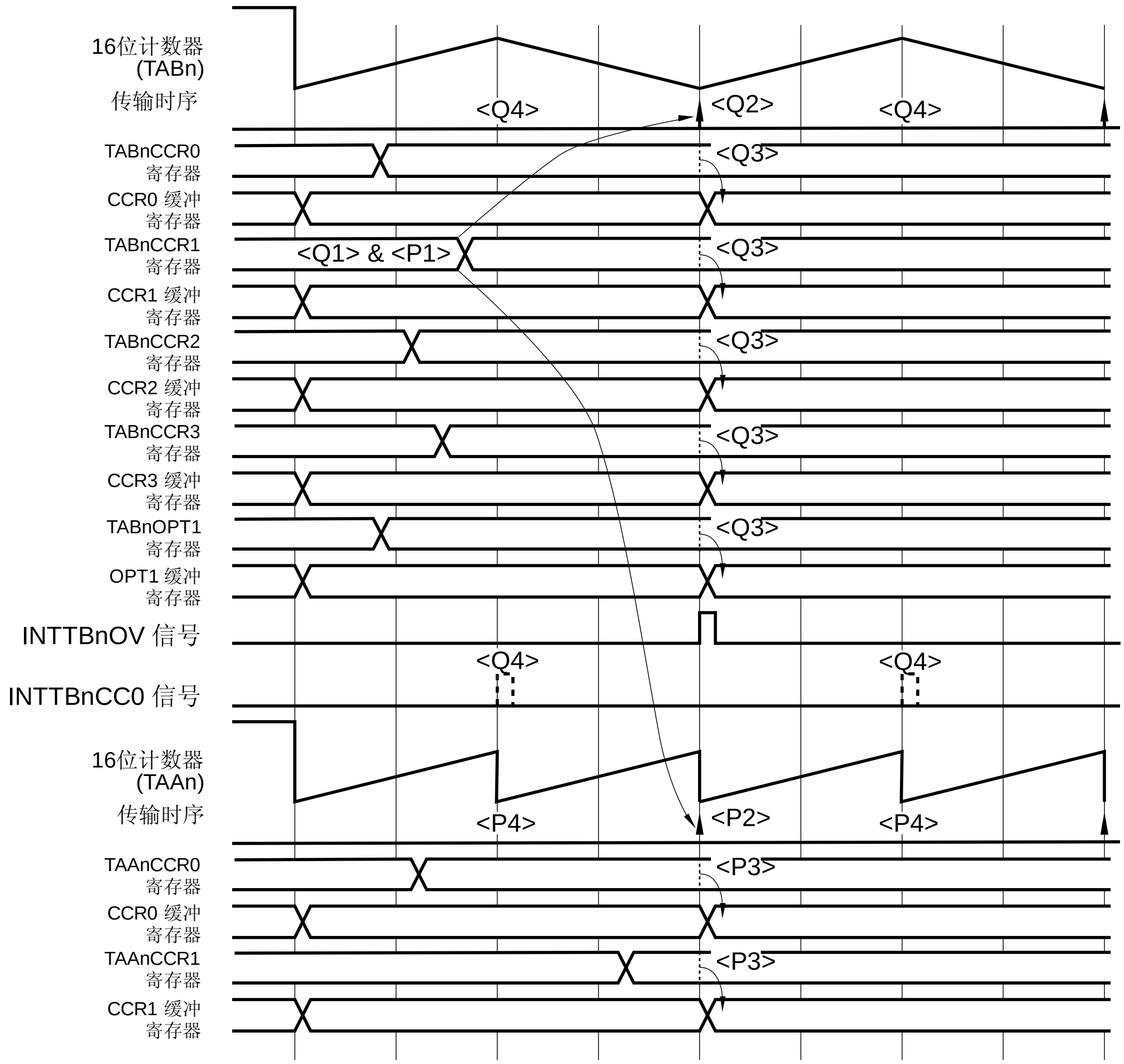
通过将 TABnOPT0.TABnCMS 位清 0，将 TABnOPT2.TABnRDE 位设置为 1，可以指定该模式。
在该模式中下，在分选的传输时序后，写入各个比较寄存器的值都会被立即传输到内部缓冲寄存器，并与计数器值进行比较。传输时序是通过中断分选产生中断（INTTBnCC0，INTTBnOV）的时序。
关于中断分选功能的详情参见 **10.4.3 中断分选功能**。

（a） 重写步骤

如果向 TABnCCR1 寄存器写入数据，那么在下个传输时序处，设置在 TABnCCR0 至 TABnCCR3，TABnOPT1，TAAnCCR0 及 TAAnCCR1 寄存器中的数值全部被立即传输到内部缓冲寄存器中。因此，TABnCCR1 寄存器应该最后写入。传输时序生成之前，在写入 TABnCCR1 寄存器后禁止对其他寄存器进行写操作（直到产生 INTTBnOV 或 INTTBnCC0 中断）。操作步骤如下。

<1> 重写 TABnCCR0，TABnCCR2，TABnCCR3，TABnOPT1，TAAnCCR0 和 TAAnCCR1 寄存器。
对于无需重写的寄存器，不要重写。
<2> 重写 TABnCCR1 寄存器。
即使无需重写 TABnCCR1 寄存器，也应该将相同的值重写到该寄存器中。
<3> 保持下次重写为未决状态直至生成传输时序。
产生 INTTBnOV 或 INTTBnCC0 中断之后，对寄存器再次进行重写。
<4> 返回步骤<1>。

图 **10-31.** 间隙批量重写模式的基本操作



[TABn 操作]

<Q1> 写入 TABnCCR1 寄存器。

<Q2> 在重写 TABnCCR1 寄存器后产生的传输时序处重写该寄存器。

<Q3> 在传输时序处这些寄存器全部被立即传输。

<Q4> 分选中断，则传输时序也被分选。

[TAAn 操作]

<P1> 写入 TABnCCR1 寄存器。

<P2> 在重写 TABnCCR1 寄存器后产生的传输时序处重写寄存器。

<P3> 在传输时序处这些寄存器全部被立即传输。

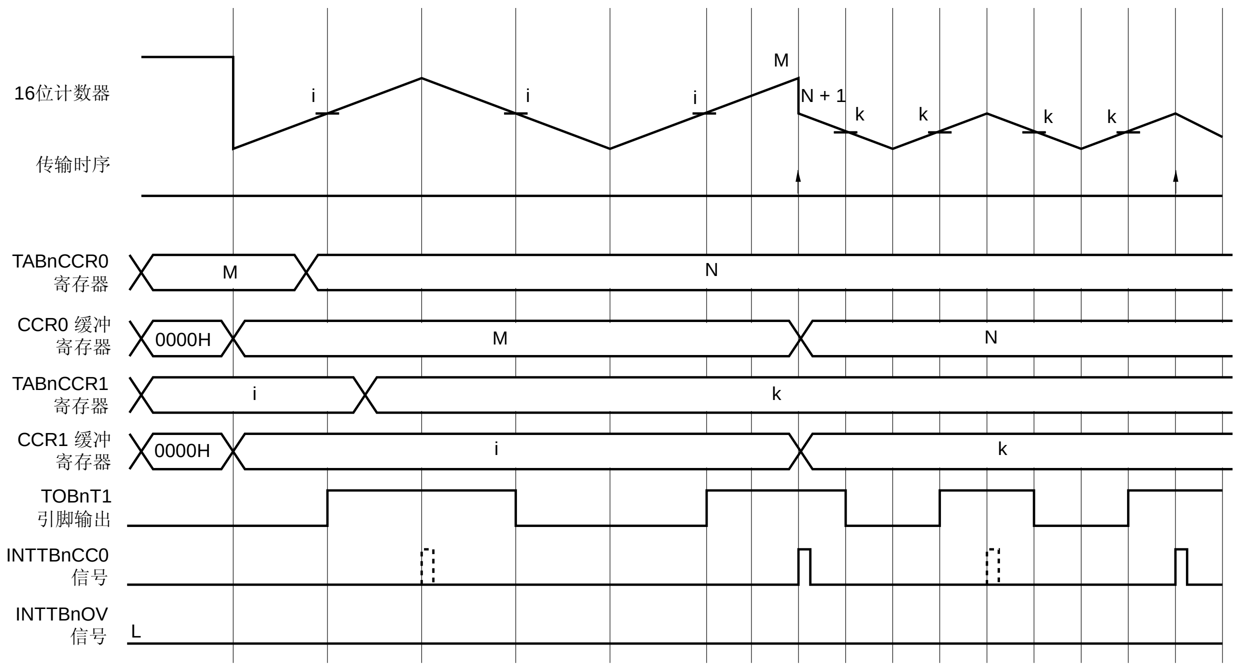
<P4> 分选中断，则传输时序也被分选。

备注 这是 TABnOPT1.TABnICE 位 = 1，TABnOPT1.TABnIOE 位 = 1，TABnOPT1.TABnID4 至 TABnOPT1.TABnID0 位= 00001 时的操作示例。

（b） 重写 TABnCCR0 寄存器

在间隙批量重写模式中重写 TABnCCR0 寄存器时，根据中断分选设置指定的波峰或波谷中断，输出波形会有所不同。下图说明了分选中断时输出波形的变化。

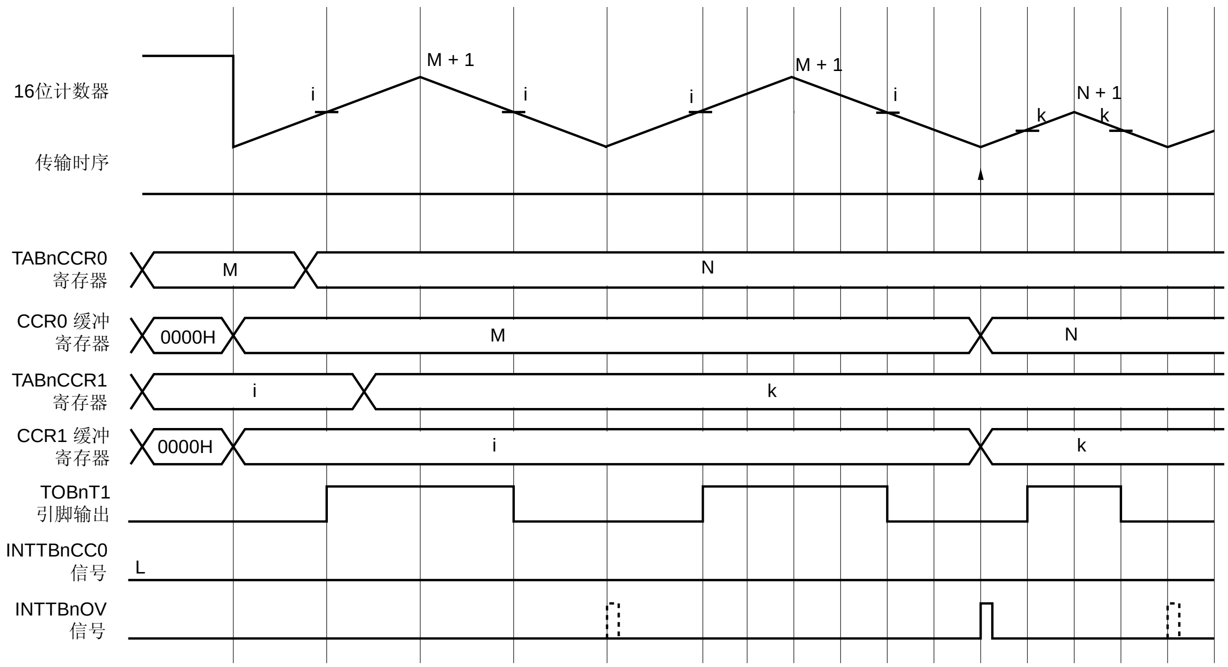
图 10-32. 重写 TABnCCR0 寄存器（设为波峰中断时）



发生波峰中断时会生成传输时序，向上和向下计数的周期会发生变化，且输出不对称的三角波。

备注
1. 这是 TABnOPT1.TABnICE 位 = 1， ABnOPT1.TABnIOE 位 = 0， TABnOPT1.TABnID4 至 TABnOPT1.TABnID0 位= 00001 的操作示例。
2. ⁞: 被分选的中断
3. n = 0, 1

图 **10-33.** 重写 **TABnCCR0** 寄存器（设为波谷中断时）



发生波谷中断时，生成传输时序，向上计数和向下计数的周期变为相等，且输出对称的三角波。

备注

1. 这是 TABnOPT1.TABnICE 位 = 0，TABnOPT1.TABnIOE 位 = 1，TABnOPT1.TABnID4 至 TABnOPT1.TABnID0 位= 00001 时的操作示例。

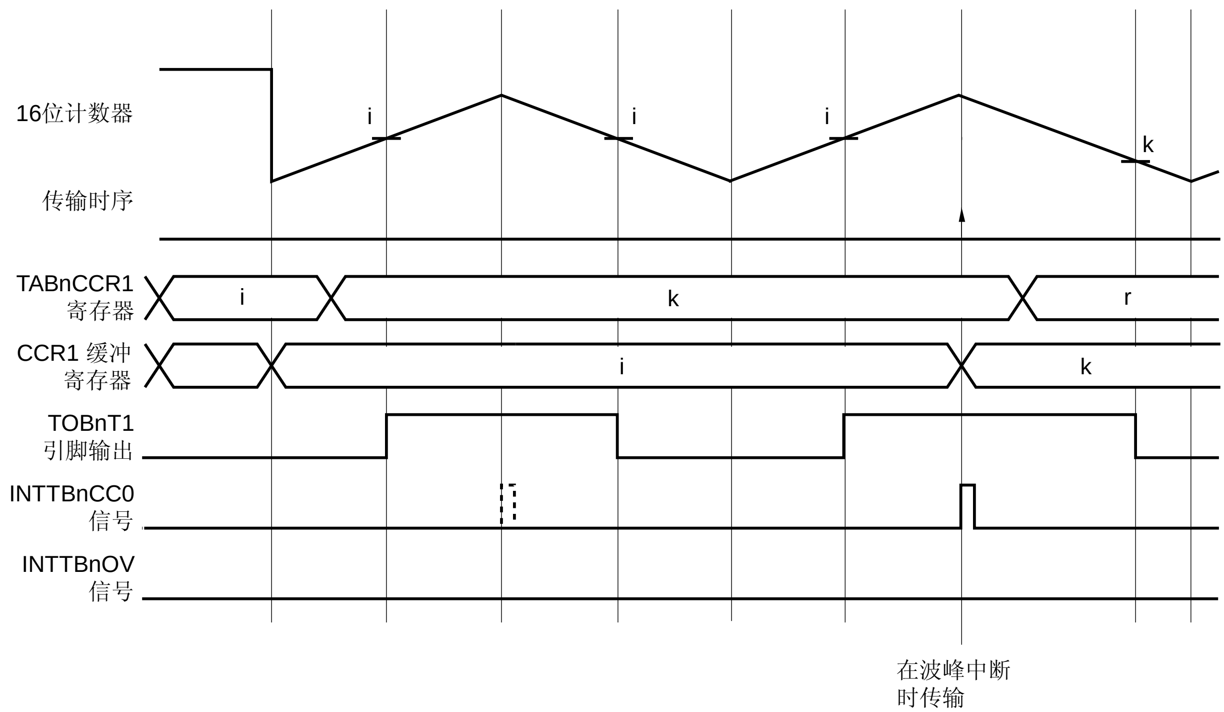
2. ┆┆：被分选的中断

3. n = 0, 1

（c）重写 **TABnCCR1** 至 **TABnCCR3** 寄存器

- 设为波峰中断时，在波峰传输
  由于在波峰中断的传输时序处传输这些寄存器，因此会输出不对称的三角波。

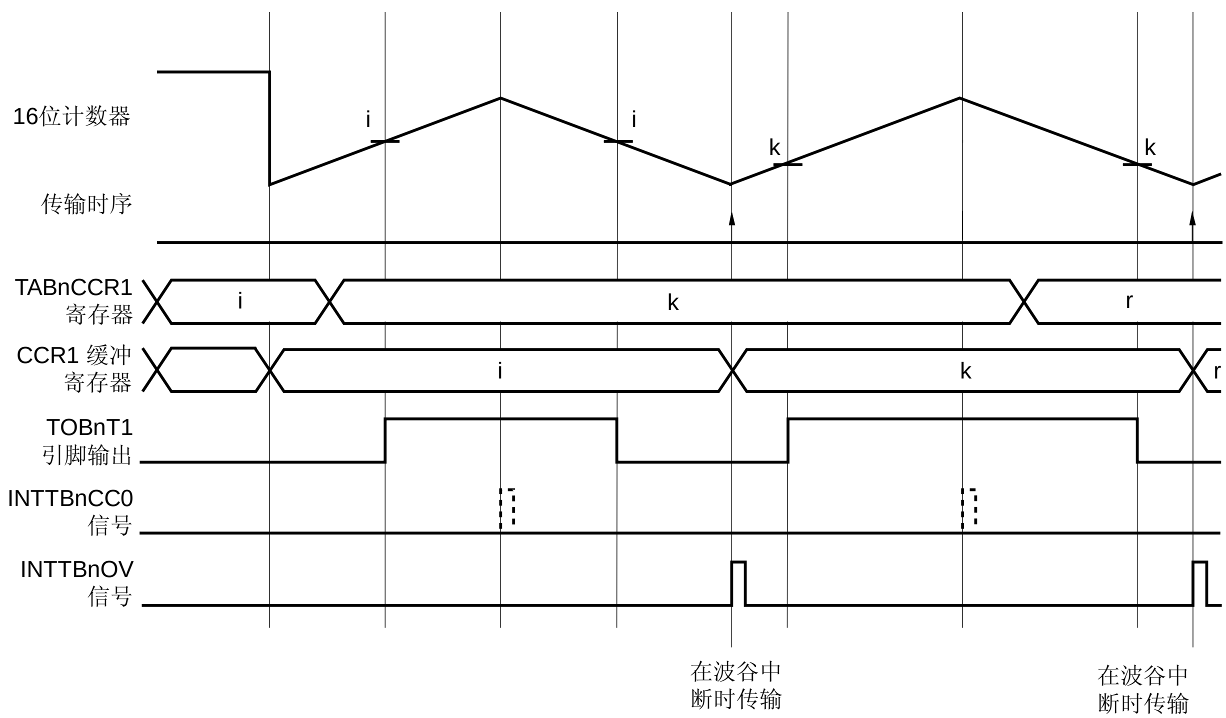
图 10-34. 重写 TABnCCR1 寄存器
（TABnOPT1.TABnICE 位= 1，TABnOPT1.TABnIOE 位= 0，TABnOPT1.TABnID4 至 TABnOPT1.TABnID0 位= 00001）



备注 **1.** ：被分选的中断
2. n = 0, 1

- 设为波谷中断时，在波谷传输
  由于在波谷中断的传输时序处传输这些寄存器，因此会输出对称的三角波。

图 10-35. 重写 TABnCCR1 寄存器
（TABnOPT1.TABnICE 位 = 1，TABnOPT1.TABnIOE 位= 1，TABnOPT1.TABnID4 至 TABnOPT1.TABnID0 位= 00001）



备注 **1.** ⁞: 被分选的中断
2. n = 0, 1

（**d**） **重写 TABnOPT1 寄存器**

当中断分选计数器的值与 16 位计数器的值一致时，会传输新的中断分选值，因此下次及后续的中断会在设定的间隔时间发生。

关于重写 TABnOPT1 寄存器的详情参见 **10.4.3 中断分选功能**。

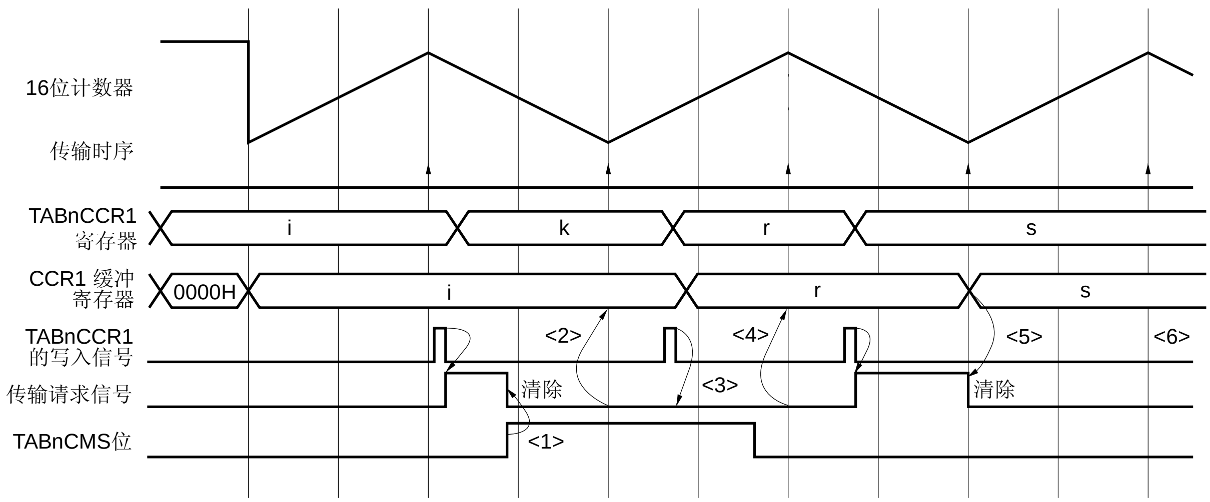
（4） 重写 TABnOPT0.TABnCMS 位

TABnCMS 位可以选择随时重写模式和批量重写模式。该位可以在定时器操作过程中（TABnCTL0.TABnCE 位 = 1 时）进行重写。但是，图 10-36 中说明的操作和注意事项是必需的。

如果在 TABnCMS 位被清除为 0 时，如果写入 TABnCCR1 寄存器，那么将会设定一个传输请求信号（内部信号）。

当设定了传输请求信号时，那么将会在下个传输时序处传输该寄存器，且清除传输请求信号。当 TABnCMS 位被设置为 1 时，该传输请求信号被清除。

图 10-36. 重写 TABnCMS 位



<1> 如果在 TABnCMS 位为 0 时重写 TABnCCR1 寄存器，会设定一个传输请求信号。
如果在该状态下设 TABnCMS 位为 1，该传输请求信号会被清除。

<2> 由于 TABnCMS 位被置为 1 且传输请求信号被清除，寄存器将不会被传输。

<3> 即使在 TABnCMS 位为 1 时写入 TABnCCR1 寄存器，也不会设定传输请求信号。

<4> 即使在 TABnCMS 位为 1 时写入 TABnCCR1 寄存器，也不会设定传输请求信号，因此即使 TABnCMS 位被清除为 0，也不会在后续的传输时序处发生传输。

<5> 如果当 TABnCMS 位为 0 时写入 TABnCCR1 寄存器，将会设定一个传输请求信号。
将会在后续的传输时序处执行传输，且传输请求信号被清除。

<6> 一旦执行传输，就会清除传输请求信号。因此，在下个传输时序处不会执行传输。

备注 n = 0, 1

10.4.5 用于A/D转换启动触发信号输出的TAAn调谐操作

本节介绍在 6 相 PWM 输出模式下 TAAn 和 TABn 的调谐功能。

在 6 相 PWM 输出模式下，TABn 为主，TAAn 为从，共同执行调谐操作。通过 TAAn 的 INTTAnCC0 和 INTTAnCC1 信号及 TABn 的 INTTBnOV 和 INTTBnCC0 信号，可以将 A/D 转换器 0 和 1 的转换启动触发信号设置为 A/D 转换启动触发源。

备注　　n = 0, 1

（1）调谐操作启动过程

应该按照以下步骤来设置 TAAn 和 TABn 寄存器，用来执行调谐操作。

（a）TAAn 寄存器的设置（停止 TABn 和 TAAn 的操作（将 TABnCTL0.TABnCE 位及 TAAnCTL0.TAAnCE 位清除为 0））。

- 设置 TAAnCTL1 寄存器为 85H （设置调谐操作从模式以及自由运行定时器模式）。
- 清除 TAAnOPT0 寄存器为 00H （选择比较寄存器）。
- 设置一个合适的值到 TAAnCCR0 和 TAAnCCR1 寄存器（为启动操作比较而设定的默认值）。

（b）TABn 寄存器的设置

- 设置 TABnCTL1 寄存器为 07H （设置主模式以及 6 相 PWM 输出模式）。
- 设置一个合适的值到 TABnIOC0 寄存器（设置 TOBnT1 至 TOBnT3 的输出模式）。
  但是，将 TABnOL0 位清 0 并将 TABnOE0 位置 1（使能正相输出）。除非已经完成该设置，否则不会产生波峰中断（INTTBnCC0）及波谷中断（INTTBnOV）。因而，不能正确生成 A/D 转换器的转换启动触发信号。
- 将 TABnIOC1 和 TABnIOC2 寄存器清除为 00H（不使用 TABn 的 TIBn0 至 TIBn3、EVTBn 及 TRGBn 引脚）。
- 将 TABnOPT0 寄存器清除为 00H （选择比较寄存器）。
- 设置一个合适的值到 TABnCCR0 至 TABnCCR3 寄存器（为启动操作比较而设定的默认值）。
- 设置 TABnCTL0 寄存器为 0xH （将 TABnCE 位清除为 0，且设置 TABn 的操作时钟）。
- 通过 TABnCTL0 寄存器设置的 TABn 操作时钟也会提供给 TAAn，并按相同时序执行计数操作。由 TAAnCTL0 寄存器设置的 TAAn 操作时钟被忽略。

（c）TMQOPn（TMQn 选项）寄存器的设置

- 设置一个合适的值到 TABnOPT1 及 TABnOPT2 寄存器。
- 设置一个合适的值到 TABnIOC3 寄存器（设置 TOBnB1 至 TOBnB3 为输出模式）。
- 设置一个合适的值到 TABnDTC 寄存器（为启动操作比较而设定的默认值）。

（d）复用功能的设置

- 通过设置端口为端口控制模式，选择端口的复用功能。

（e）设置 TAAnCE 位为 1 且立即设置 TABnCE 位为 1，以启动 6 相 PWM 输出操作

操作期间禁止重写 TABnCTL0，TABnCTL1，TABnIOC1，TABnIOC2，TAAnCTL0 和 TAAnCTL1 寄存器。如果在操作期间重写了这些寄存器中的任意一个，无法保证操作，也无法保证 PWM 输出波形。但是，允许对 TABnCTL0.TABnCE 位的重写，将其清零。直到 TAAnCTL0.TAAnCE 位被置 1 且随后 TABnCE 位被置 1，禁止操作（读取/写入）TABn、TAAn 及 TMQn 选项的其它寄存器。

（2） 调谐操作的清除过程

要清除调谐操作并退出 6 相 PWM 输出模式，应按照以下步骤来设置 TAAn 和 TABn 寄存器。

<1> 将 TABnCTL0.TABnCE 位清 0，且停止定时器操作。

<2> 将 TAAnCTL0.TAAnCE 位清 0 ，来分离 TAAn。

<3> 使用 TABnIOC0 寄存器停止定时器输出。

<4> 将 TAAnCTL1.TAAnSYE 位清 0，清除调谐操作 。

注意事项 禁止操作（读取/写入）TABn、TAAn 及 TMQn 选项的其它寄存器，直到 TABnCE 位被置 0 然后 TAAnCE 位被置 0。

（3）没有调谐 TAAn 时

当不需要将 TAAn 的一致中断信号作为启动 A/D 转换器 0 和 1 的转换触发源时，TAAn 可以用作一个不需要调谐的独立的定时器。在这种情况下，TAAn 的一致中断信号不能在 6 相 PWM 输出模式中被用作启动 A/D 转换的触发源。因此，应将 TABnOPT2.TABnAT0 至 TABnOPT2.TABnAT3 位和 TABnOPT3.TABnAT4 至 TABnOPT3.TABnAT7 位固定为 0。

其他控制位的使用和 TAAn 用于调谐时的方式相同。

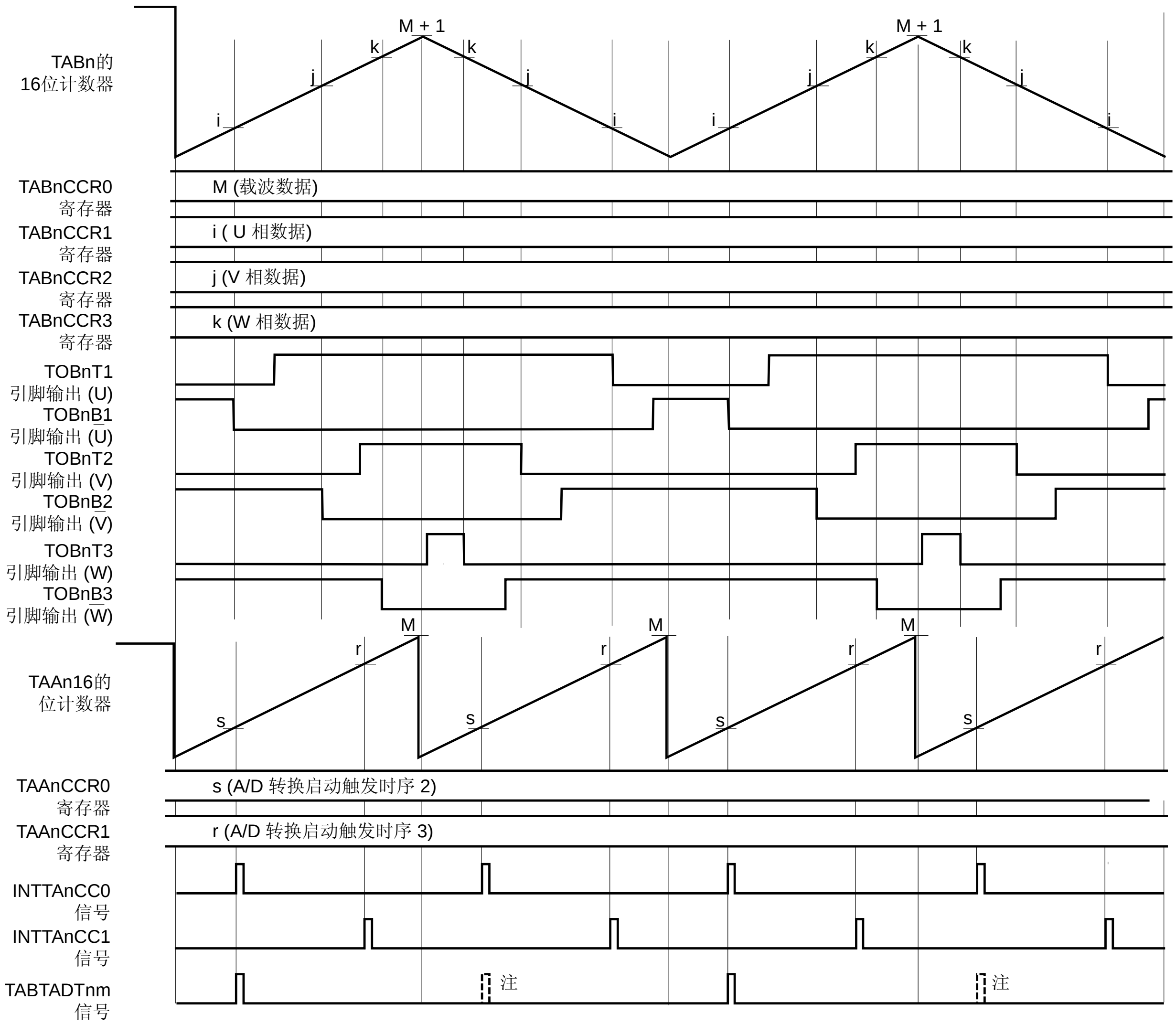
如果 TAAn 没有用于调谐，那么 TAAn 的比较寄存器（TAAnCCR0 及 TAAnCCR1） 将不会受到 TABnOPT0.TABnCMS 和 TABnOPT2.TABnRDE 位的影响。关于 TAAn 没有用于调谐时的初始化步骤，请参见 **10.4.5 （1） 调谐操作启动过程**一节的步骤（b）到（e）。不需要步骤（a），因为它是将 TAAn 设置为调谐操作的步骤。

（4）调谐操作期间 TAAn 的基本操作

TAAn 的 16 位计数器仅能向上计数。16 位计数器通过 TABnCCR0 寄存器的设定周期值来清除，并从 0000H 开始再次计数。当它向上计数时，该计数器的计数值与 TAAn 的 16 位计数器的值相同。但是，当 TABn 的 16 位计数器向下计数时，两个计数值不相同。

- 当 TABn 向上计数时（相同的值）
  TABn 的 16 位计数器： 0000H → M （向上计数）
  TAAn 的 16 位计数器： 0000H → M （向上计数）
- 当 TABn 向下计数时（不相同的值）
  TABn 的 16 位计数器： M + 1 → 0001H （向下计数）
  TAAn 的 16 位计数器： 0000H → M （向上计数）

图 **10-37. 调谐操作期间的 TAAn**



注 TABTADTn0 信号由 TABnOPT2.TABnATM2 和 TABnOPT2.TABnATM3 位来屏蔽。TABTADTn1 信号由 TABnOPT3.TABnATM6 和 TABnOPT3.TABnATM7 位来屏蔽。

备注 n = 0, 1，m = 0, 1

10.4.6 A/D转换启动触发输出功能

V850E/IF3 和 V850E/IG3 具有选择四个触发源（INTTBnOV，INTTBnCC0，INTTAnCC0，INTTAnCC1），可以生成 A/D 转换器 0 和 1 的 A/D 转换启动触发信号（TABTADTn0，TABTADTn1）。

触发源由 TABnOPT2.TABnAT0 至 TABnOPT2.TABnAT3 和 TABnOPT3.TABnAT4 至 TABnOPT3.TABnAT7 来指定。

- TABnAT0, TABnAT4 位= 1：
  当产生 INTTBnOV（计数器下溢）时，生成 A/D 转换启动触发信号。
- TABnAT1, TABnAT5 位 = 1：
  当产生 INTTBnCC0 （周期一致）时，生成 A/D 转换启动触发信号。
- TABnAT2, TABnAT6 位 = 1：
  当产生 INTTAnCC0（调谐操作期间和 TAAn 的 TAAnCCR0 寄存器一致）时，生成 A/D 转换启动触发信号。
- TABnAT3, TABnAT7 位 = 1：
  当产生 INTTAnCC1（调谐操作期间和 TAAn 的 TAAnCCR1 寄存器一致）时，生成 A/D 转换启动触发信号。

由 TABnAT0 至 TABnAT3 和 TABnAT4 至 TABnAT7 位选择的 A/D 转换启动触发信号，经过或操作后输出。因此，可以同时指定两个或更多个触发源。

由 TABnAT0, TABnAT1, TABnAT4 和 TABnAT5 位选择的 INTTBnOV 和 INTTBnCC0 信号是已分选中断信号。

因此，在中断被分选后输出这些信号，除非使能中断输出（TABnOPT1.TABnICE，TABnOPT1.TABnIOE 位），不会输出 A/D 转换启动触发信号。

根据 16 位计数器向上计数 / 向下计数的状态，来自 TAAn 的触发源（INTTAnCC0 和 INTTAnCC1）可以具有屏蔽 A/D 转换启动触发信号的功能，如果被屏蔽，TABnAT2, TABnAT3, TABnAT6 和 TABnAT7 位被置位。

- TABnATM2, TABnATM6 位：
  对应 TABnAT2 和 TABnAT6 位，且控制 TAAn 的 INTTAnCC0（一致中断信号）。
  - TABnATM2, TABnATM6 位 = 0：
    当 16 位计数器向上计数（TABnOPT0.TABnCUF 位 = 0）时，输出 A/D 转换启动触发信号，当 16 位计数器向下计数（TABnOPT0.TABnCUF 位 = 1）时，不输出 A/D 转换启动触发信号。
  - TABnATM2, TABnATM6 位 = 1：
    当 16 位计数器向下计数（TABnOPT0.TABnCUF 位 = 1）时，输出 A/D 转换启动触发信号，当 16 位计数器向上计数（TABnOPT0.TABnCUF 位 = 0）时，不输出 A/D 转换启动触发信号。

- TABnATM3, TABnATM7 位：
  对应 TABnAT3 和 TABnAT7 位，且控制 TAAn 的 INTTAnCC1（一致中断信号）。
  - TABnATM3, TABnATM7 位 = 0：
    当 16 位计数器向上计数（TABnCUF 位 = 0）时，输出 A/D 转换启动触发信号，当 16 位计数器向下计数（TABnCUF 位 = 1）时，不输出 A/D 转换启动触发信号。
  - TABnATM3, TABnATM7 位 = 1：
    当 16 位计数器向下计数（TABnCUF 位 = 1）时，输出 A/D 转换启动触发信号，当 16 位计数器向上计数（TABnCUF 位 = 0）时，不输出 A/D 转换启动触发信号。

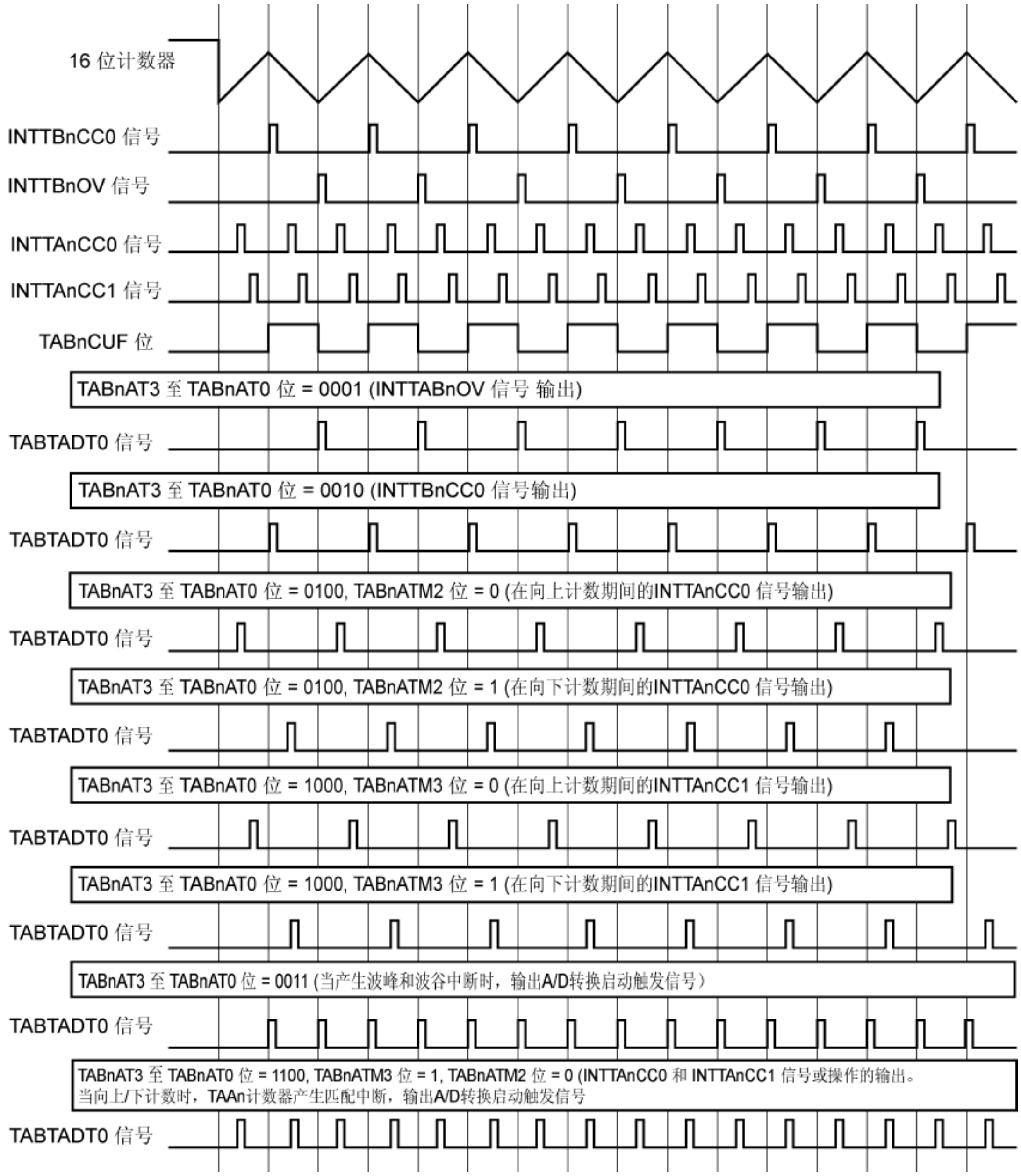
在定时器操作时，可以重写 TABnATM3, TABnATM2, TABnAT3 至 TABnAT0, TABnATM7, TABnATM6 和 TABnAT7 至 TABnAT4 位。如果在定时器操作时重写设置 A/D 转换启动触发信号的位，那么新的设置会立即反映在 A/D 转换启动触发信号的输出状态上。这些控制位不具有传输功能，并且只能在随时重写模式下使用。

注意事项 **1.** 只有在 **TAAn** 被用作调谐操作中的 **TABn** 的从定时器时，由 **TABnAT2, TABnAT3, TABnAT6** 和 **TABnAT7** 位设置的 **A/D** 转换启动触发信号才能输出。如果 **TABn** 和 **TAAn** 没有执行调谐操作，或者没有使用 **6** 相 **PWM** 输出模式，那么输出无法保证。

2. **TOBn0** 信号输出被内部用于识别 **16** 位计数器是向上计数还是向下计数。因此，通过将 **TABnIOC0.TABnOL0** 位清零且设置 **TABnIOC0.TABnOE0** 位为 **1**，可以使能 **TOBn0** 引脚输出。

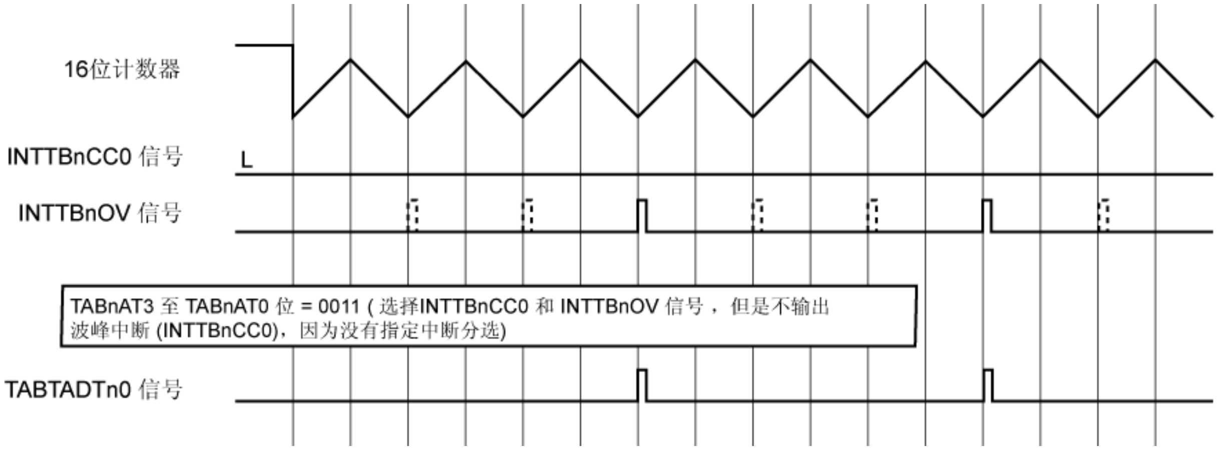
图 **10-38. A/D** 转换启动触发（**TABTADTn0**）信号输出的示例

（**TABnOPT1.TABnICE** 位 **= 1**，**TABnOPT1.TABnIOE** 位 **= 1**，**TABnOPT1.TABnID4** 至 **TABnOPT1.TABnID0** 位**= 00000**：没有中断分选）



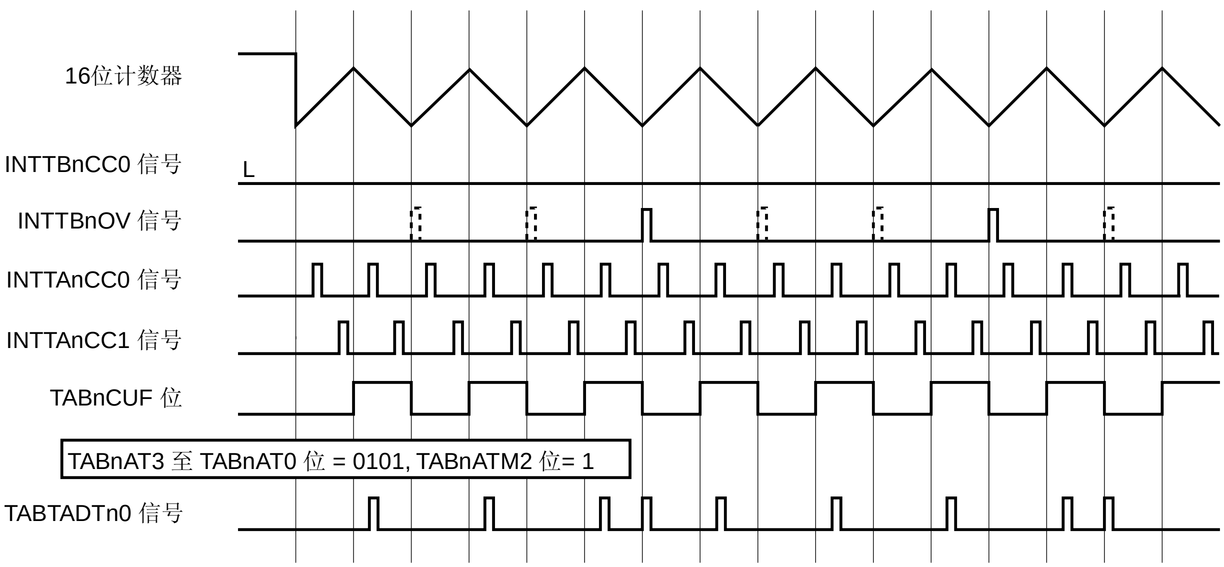
备注　　n = 0, 1

图 10-39. A/D 转换启动触发（TABTADTn0）信号输出的示例
（TABnOPT1.TABnICE 位 = 0，TABnOPT1.TABnIOE 位 = 1，TABnOPT1.TABnID4 至 TABnOPT1.TABnID0 位= 00010：带有中断分选）（1）



备注 1. ⁝: 被分选的中断
2. n = 0，1

图 10-40. A/D 转换启动触发（TABTADTn0）信号输出的示例
（TABnOPT1.TABnICE 位 = 0，TABnOPT1.TABnIOE 位 = 1，TABnOPT1.TABnID4 至 TABnOPT1.TABnID0 位= 00010：带有中断分选）（2）



注意事项 **INTTBnCC0** 信号被分选，但是 **INTTAnCC0** 信号没有被分选。

备注 1. ⁝: 被分选的中断
2. n = 0, 1

（1） 边界条件下的操作（**16** 位计数器与 **INTTAnCC0** 信号一致时的操作）

表 10-3. 当 TABnCCR0 寄存器= M，TABnAT2 位 = 1，TABnAT6 位 = 1，TABnATM2 位 = 0，TABnATM6 位 = 0 时的操作（选择向上计数期间）

TAAnCCR0 寄存器的值	TABn 的 16 位计数器的值	TAAn 的 16 位计数器的值	TABn 的 16 位计数器的状态	从 TABTADTnm 信号输出的 INTTAnCC0 信号
0000H	0000H	0000H	–	输出
0000H	M + 1	0000H	–	不输出
0001H	0001H	0001H	向上计数	输出
0001H	M	0001H	向下计数	不输出
M	M	M	向上计数	输出
M	0001H	M	向下计数	不输出

表 10-4. 当 TABnCCR0 寄存器 = M，TABnAT2 位 = 1，TABnAT6 位 = 1，TABnATM2 位 = 1，TABnATM6 位 = 1 的操作（选择向下计数期间）

TAAnCCR0 寄存器的值	TABn 的 16 位计数器的值	TAAn 的 16 位计数器的值	TABn 的 16 位计数器的状态	从 TABTADTnm 信号信号输出的 INTTAnCC0 信号
0000H	0000H	0000H	–	不输出
0000H	M + 1	0000H	–	输出
0001H	0001H	0001H	向上计数	不输出
0001H	M	0001H	向下计数	输出
M	M	M	向上计数	不输出
M	0001H	M	向下计数	输出

注意事项 当 **TABnCCR0** 寄存器 **= M** 时，**TAAnCCRm** 寄存器允许**“0”** 至**“M”**的数值设置。禁止设置 **“M + 1”**或更大的值。
如果设置的值等于或大于**“M + 1”**，**TAAn** 的 **16** 位计数器在**“M”**处清除。所以将不会输出 **TABTADTnm** 信号。

备注 n = 0, 1，m = 0，1

11.1 功能

看门狗定时器具有以下功能：

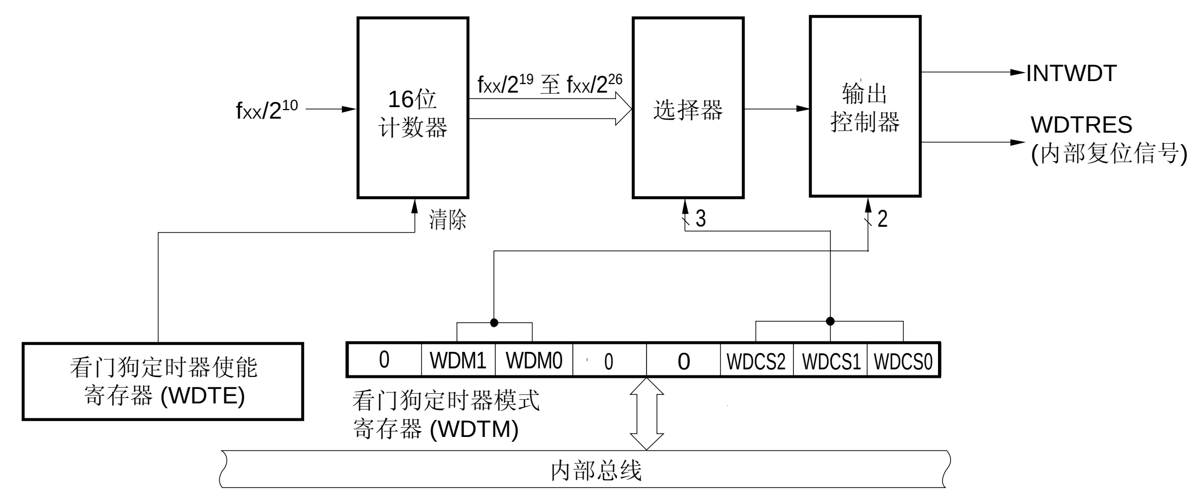
- 复位模式：看门狗定时器溢出时执行复位操作（产生 WDTRES 信号）
- 不可屏蔽中断请求模式：
  看门狗定时器溢出时执行不可屏蔽中断操作（产生 INTWDT 信号）

注意事项 在复位解除后，看门狗定时器停止。
当"ACH"写入WDTE寄存器时，启动操作。同样，即使不需要改变默认设置（复位模式，间隔时间：2^{26}/fxx），为了确保操作的正确性，也要对WDTM寄存器执行一次写入操作。

11.2 配置

看门狗定时器的功能框图如下所示：

图 11-1. 看门狗定时器的功能框图



备注 $f_{XX}/2^{10}$: 看门狗定时器时钟
fxx： 外围时钟
INTWDT： 看门狗定时器溢出时产生不可屏蔽中断请求信号
WDTRES： 看门狗定时器溢出时产生复位信号

看门狗定时器包括下列硬件：

表 11-1. 看门狗定时器的配置

项目	配置
控制寄存器	看门狗定时器模式寄存器 （WDTM） 看门狗定时器使能寄存器（WDTE）

11.3 控制寄存器

（1）看门狗定时器模式寄存器（WDTM）

WDTM 寄存器用于设置看门狗定时器的溢出时间和工作时钟。

该寄存器可按字节读取或写入。该寄存器读取次数不限，但是仅能在复位解除后写入一次。它不能被二次或多次写入。

复位输入将该寄存器设置为 67H。

复位后： 67H　　R/W　　地址：FFFFF6D0H

	7	6	5	4	3	2	1	0
WDTM	0	WDM1	WDM0	0	0	WDCS2	WDCS1	WDCS0

WDM1	WDM0	看门狗定时器操作模式的规范
0	0	停止操作
0	1	不可屏蔽中断请求模式 (产生INTWDT 信号)
1	×	复位模式 (产生WDTRES 信号)

注意事项　**1. 关于 WDCS2 至 WDCS0 位的细节，参见表 11-2 溢出时间。**

2. 请确保将第 3 位，第 4 位和第 7 位清除为“0”。

表 11-2. 溢出时间

WDCS2	WDCS1	WDCS0	溢出时间	f_{XX} = 64 MHz	f_{XX} = 32 MHz
0	0	0	$2^{19}/f_{XX}$	8.2 ms	16.4 ms
0	0	1	$2^{20}/f_{XX}$	16.4 ms	32.8 ms
0	1	0	$2^{21}/f_{XX}$	32.8 ms	65.5 ms
0	1	1	$2^{22}/f_{XX}$	65.5 ms	131.1 ms
1	0	0	$2^{23}/f_{XX}$	131.1 ms	262.1 ms
1	0	1	$2^{24}/f_{XX}$	262.1 ms	524.3 ms
1	1	0	$2^{25}/f_{XX}$	524.3 ms	1,048.5 ms
1	1	1	$2^{26}/f_{XX}$	1,048.5 ms	2,097.1 ms

（2）看门狗定时器使能寄存器（WDTE）

将“ACH”写入 WDTE 寄存器，可以清除看门狗定时器的计数值，并且重新开始计数。

WDTE 寄存器可以按字节读取或写入。

复位输入将该寄存器设为 1AH。

复位后： 1AH　R/W　地址： FFFFF6D1H

	7	6	5	4	3	2	1	0
WDTE								

注意事项 **1. 如果“ACH”写入 WDTE 寄存器来使能看门狗定时器操作，然后有“ACH”之外的数值写入 WDTE 寄存器，则在看门狗溢出时间，根据 WDTM.WDM1 和 WDTM.WDM0 位的指定，产生一个不可屏蔽中断请求信号（INTWDT）或一个复位信号（WDTRES）。**

2. 对 WDTE 寄存器执行 1 位读取或写入时，输出一个内部复位信号。

3. 在看门狗定时器工作之前，WDTE 寄存器的读取值是“1AH”，开始工作后，寄存器的读出值是“9AH”。该寄存器的读取值和写入值（ACH）不同。

11.4 操作

复位解除后，看门狗定时器停止。

复位解除后，WDTM 寄存器只能写入一次。如果在看门狗定时器启动之后，二次写入该寄存器，则在看门狗溢出时间，根据 WDTM.WDM1 和 WDTM.WDM0 位的指定，产生一个不可屏蔽中断请求信号（INTWDT）或一个复位信号（WDTRES）。如果对看门狗定时器写入相同的值，也会产生 INTWDT 或 WDTRES 信号。如果该寄存器写入次数操过三次，则操作无法保证。

要使用看门狗定时器，使用字节操作向 WDTM 寄存器写入操作模式和间隔时间。之后，不能停止看门狗定时器的操作。

如果不使用看门狗定时器，则向 WDTM 寄存器写入 00H。

11.5 注意事项

因为看门狗定时器溢出而产生的不可屏蔽中断请求信号（INTWDT）的周期可以计算获得，如果没有清除看门狗定时器而连续产生INTWDT，周期为“WDTM寄存器设置的间隔时间 + 2^7 外围时钟脉冲宽度”。

请注意，脉冲宽度不包括在看门狗定时器启动之后直到产生第一次中断请求信号的时间。

12.1 特征

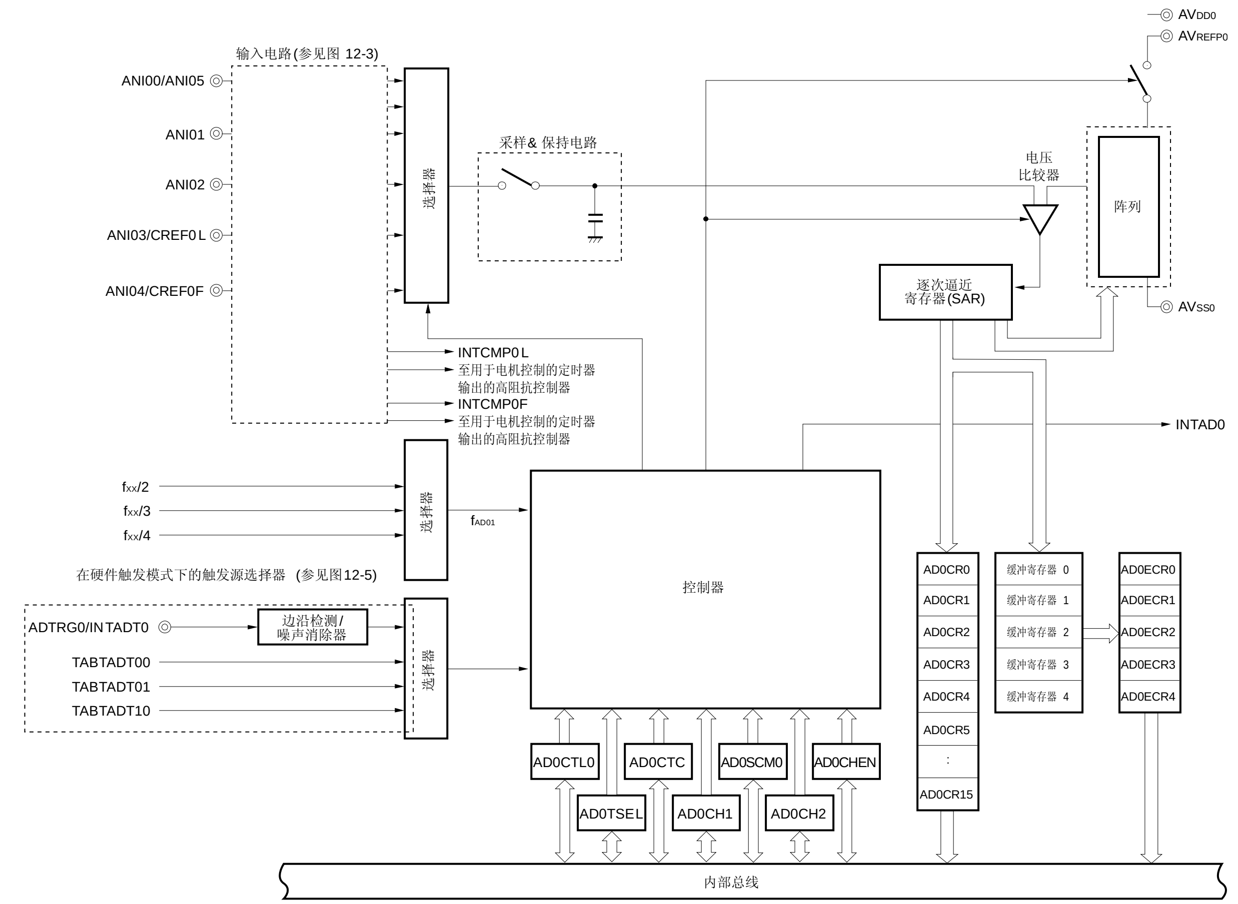
- 两个 12 位分辨率 A/D 转换器电路（A/D 转换器 0 和 1）
  两个电路可以同时采样
- 模拟输入
  - 未使用比较器时
    两个电路中总共有 10 路通道
    A/D 转换器 0：ANI00/ANI05，ANI01 至 ANI04（5 通道）
    A/D 转换器 1：ANI10/ANI15，ANI11/ANI16，ANI12/ANI17，ANI13，ANI14（5 通道）
  - 使用比较器时
    两个电路中总共有 6/8 路通道
    [6 通道（当使用比较器的小范围和全范围时）]
    A/D 转换器 0：ANI00/ANI05，ANI01，ANI02（3 通道）
    A/D 转换器 1：ANI10/ANI15，ANI11/ANI16，ANI12/ANI17（3 通道）
    [8 通道（当使用比较器的小范围或全范围时）]
    A/D 转换器 0：ANI00/ANI05，ANI01，ANI02，ANI03 或 ANI04（4 通道）
    A/D 转换器 1：ANI10/ANI15，ANI11/ANI16，ANI12/ANI17，ANI13 或 ANI14（4 通道）
- A/D 转换结果寄存器
  12 位 × 16 + 12 位 × 16
  A/D 转换器 0：AD0CR0 至 AD0CR15
  A/D 转换器 1：AD1CR0 至 AD1CR15
- A/D 转换结果扩展寄存器
  只能在扩展缓冲模式下使用
  12 位 × 5 + 12 位 × 5
  A/D 转换器 0：AD0ECR0 至 AD0ECR4
  A/D 转换器 1：AD1ECR0 至 AD1ECR4
- 操作模式
  - 正常操作模式
    A/D 触发模式
    A/D 触发轮询模式
    硬件触发模式
  - 扩展操作模式
    转换通道指定模式
    扩展缓冲模式
- 用于输入电平放大的运算放大器（×2.5 至×10）
  只有当使用用于输入电平放大的运算放大器，才可以使用这些通道。
  两个电路中总共有 4 个单元
  A/D 转换器 0：ANI05（1 个单元）
  A/D 转换器 1：ANI15 至 ANI17（3 个单元）

- 过电压检测比较器
  - 只有当使用过电压检测比较器，才可以使用这些通道。
  - 两个电路中总共有 4 个单元
    A/D 转换器 0：1 个单元
    A/D 转换器 1：3 个单元
  - 参考电压
    从 CREF0L，CREF1L 引脚（小范围端）的输入电压范围 = $0.02AV_{DD} + 0.1$ 至 $0.5AV_{DD} - 0.1$ V
    从 CREF0F，CREF1F 引脚（全范围端）的输入电压范围 = $0.02AV_{DD} + 0.1$ 至 $0.92AV_{DD} - 0.1$ V
  - 当检测到过电压时，发生中断。中断请求通过过电压检测信号引脚 ANI00/ANI05（全范围端和小范围端）输出，并作为三个通道 ANI10/ANI15，ANI11/ANI16 和 ANI12/ANI17 的过电压检测信号的逻辑和（OR）结果，或者作为两个输出信号（全范围端和小范围端）的逻辑积（AND）结果。
  - 当检测到过电压时，用于电机控制的定时器的输出可以设置为高阻抗状态。
- 逐次逼近方法
- 操作电压范围
  $EV_{DD0} = EV_{DD1} = EV_{DD2}$（仅限 V850E/IG3）$= AV_{DD0} = AV_{DD1} = AV_{REFP0} = AV_{REFP1} = 4.0$ 至 5.5 V

12.2 配置

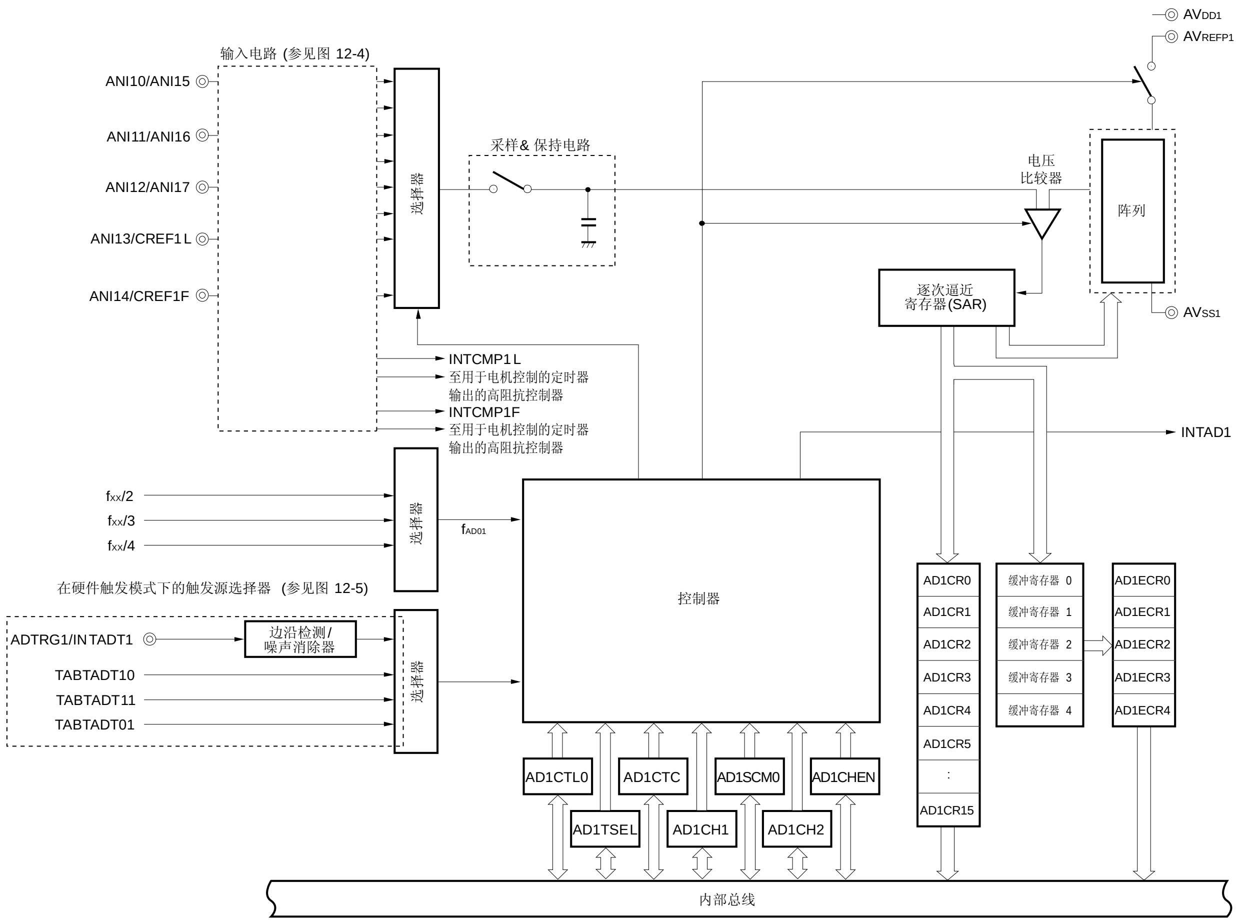
框图展示如下。

图 12-1. A/D 转换器 0 的框图



备注　f_{XX}：外围时钟
f_{AD01}：基本时钟
缓冲寄存器 0 至 4：A/D0 转换结果扩展缓冲寄存器 0 至 4

图 **12-2. A/D** 转换器 **1** 的框图



备注 f_{XX}：外围时钟

f_{AD01}：基本时钟

缓冲寄存器 0 至 4：A/D1 转换结果扩展缓冲寄存器 0 至 4

注意事项 **1.** 如果在模拟输入引脚（**ANI00** 至 **ANI05**，**ANI10** 至 **ANI17**）或在 **A/D** 转换器参考电压输入引脚（**AV**$_{\mathbf{REFP0}}$，**AV**$_{\mathbf{REFP1}}$）有噪声，噪声可能会产生非法的转换结果。

需要软件处理来防止非法转换结果对系统的负面影响。

下面是一个软件处理的示例。

- 对一定数量的 **A/D** 转换取平均值，并将该值用作 **A/D** 转换结果**.**
- 连续执行一定数量的 **A/D** 转换，并使用这些结果，忽略任何可能获得的意外结果。
- 如果获得的一个 **A/D** 转换结果判定已经产生系统误操作，请确保在执行误操作处理之前，重新检查系统误操作。

2. 不要将超出 **AV**$_{\mathbf{SSn}}$至 **AV**$_{\mathbf{REFPn}}$ 范围的电压输入到用作 **A/D** 转换器 **0** 和 **1**（**n = 0**，**1**）输入引脚的引脚。

图 12-3. A/D 转换器 0 中用于输入电平放大的运算放大器和过电压检测比较器的框图

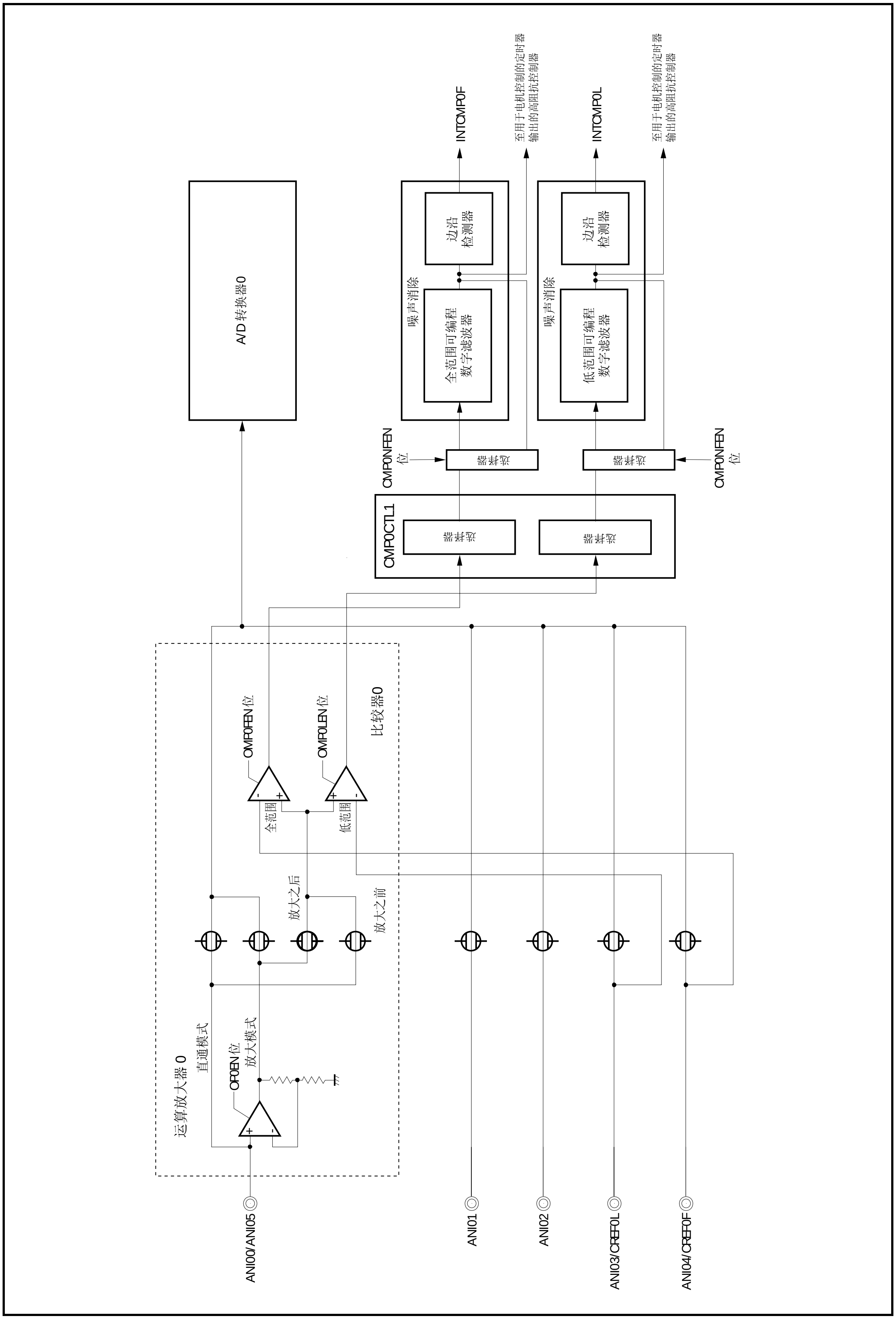


图 12-4. A/D 转换器 1 中用于输入电平放大的运算放大器和过电压检测比较器的框图

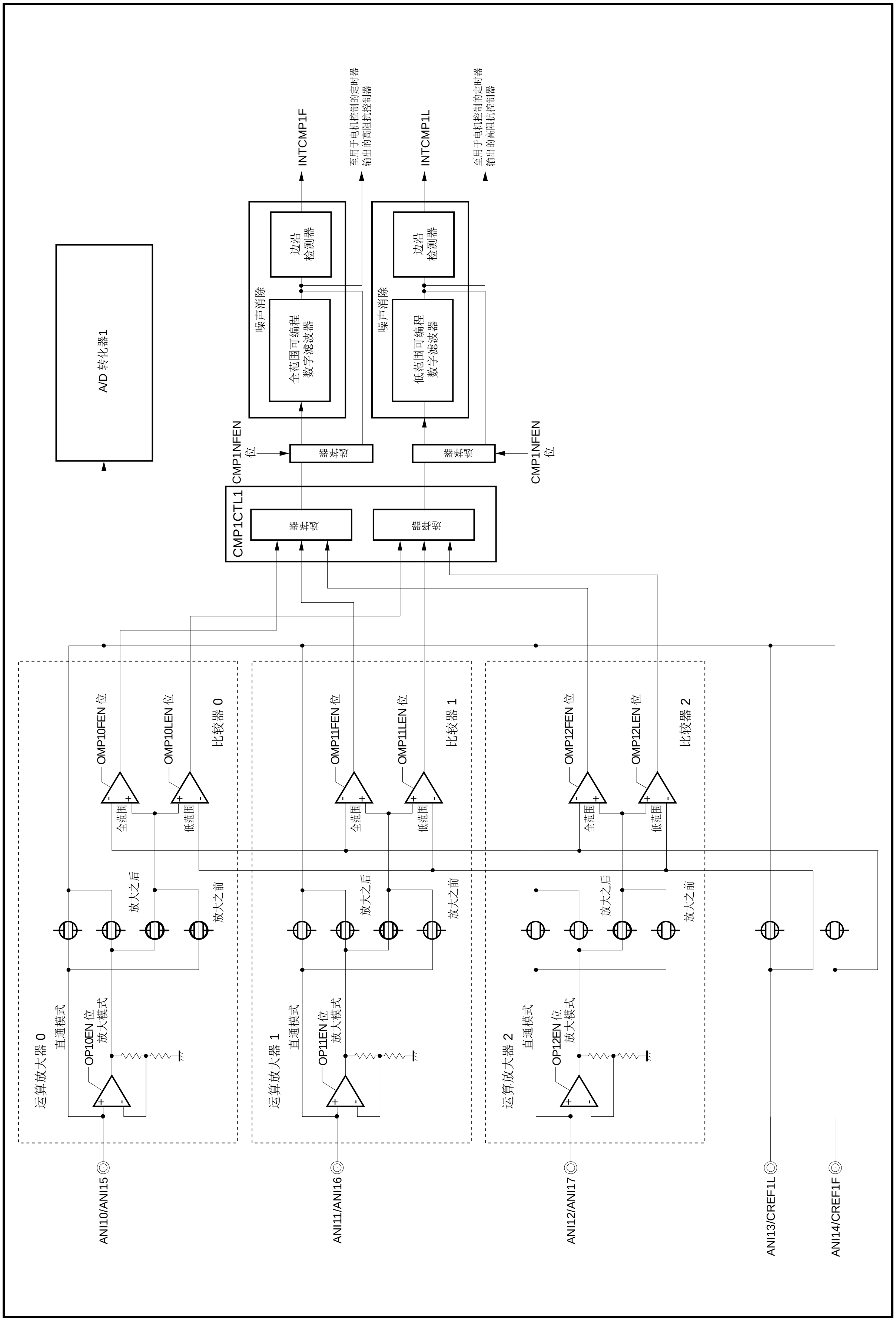
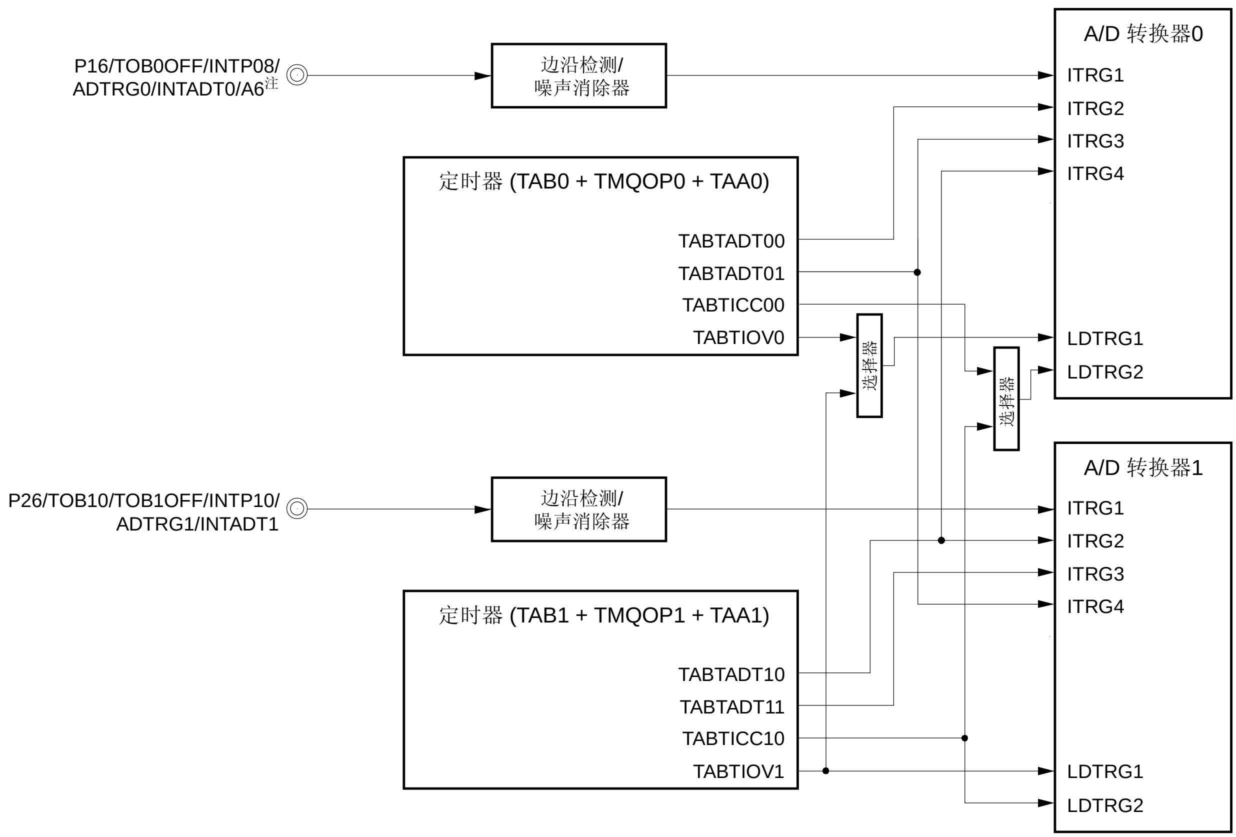


图 12-5. 硬件触发模式下触发源选择器的框图



<R> 注 仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

A/D 转换器 0 和 1 包括下列硬件。

表 12-1. A/D 转换器 0 和 1 的配置

项目	配置
模拟输入	未使用比较器： ANI00/ANI05，ANI01 至 ANI04，ANI10/ANI15 至 ANI12/ANI17，ANI13，ANI14（两个电路中总共有 10 路通道） 使用比较器（当使用比较器的小范围和全范围时）： ANI00/ANI05，ANI01，ANI02，ANI10/ANI15 至 ANI12/ANI17（两个电路中总共有 6 路通道） 使用比较器（当使用比较器的小范围或全范围时）： ANI00/ANI05，ANI01，ANI02，ANI03 或 ANI04，ANI10/ANI15 至 ANI12/ANI17，ANI13 或 ANI14（两个电路中总共有 8 路通道）
寄存器	逐次逼近寄存器（SAR） A/Dn 转换结果寄存器 0 至 15（ADnCR0 至 ADnCR15） A/Dn 转换结果寄存器 0H 至 15H（ADnCR0H 至 ADnCR15H） A/Dn 转换结果扩展寄存器 0 至 4（ADnECR0 至 ADnECR4） （仅在扩展操作模式（扩展缓冲模式）） A/Dn 转换结果扩展寄存器 0H 至 4H（ADnECR0H 至 ADnECR4H） （仅在扩展操作模式（扩展缓冲模式））
控制寄存器	A/D 转换器 n 扫描模式寄存器（ADnSCM） A/D 转换器 n 扫描模式寄存器 L（ADnSCML） A/D 转换器 n 扫描模式寄存器 H（ADnSCMH） A/D 转换器 n 转换时间控制寄存器（ADnCTC） A/D 转换器 n 转换通道指定寄存器（ADnCHEN） A/D 转换器 n 转换通道指定寄存器 L（ADnCHENL） A/D 转换器 n 转换通道指定寄存器 H（ADnCHENH） A/D 转换器 n 控制寄存器（ADnCTL0） A/D 转换器 n 触发选择寄存器（ADnTSEL） A/D 转换器 n 通道指定寄存器 1（ADnCH1） A/D 转换器 n 通道指定寄存器 2（ADnCH2） A/D 转换器 n 标志寄存器（ADnFLG） A/D 转换器 n 标志缓冲寄存器（ADnFLGB） A/DLDTRG1 输入选择寄存器（ADLTS1） A/DLDTRG2 输入选择寄存器（ADLTS2） 运算放大器 n 控制寄存器 0（OPnCTL0） 比较器 n 控制寄存器 0（CMPnCTL0） 比较器 n 控制寄存器 1（CMPnCTL1） 比较器 n 控制寄存器 2（CMPnCTL2） 比较器 n 控制寄存器 3（CMPnCTL3） A/D 转换器 n 时钟选择寄存器（ADnOCKS） A/D 触发下降沿指定寄存器（ADTF） A/D 触发上升沿指定寄存器（ADTR） 比较器输出中断上升沿指定寄存器（CMPOR） 比较器输出中断下降沿指定寄存器（CMPOF） 比较器输出数字噪声消除寄存器 nL（CMPNFCnL） 比较器输出数字噪声消除寄存器 nF（CMPNFCnF）

备注 n = 0，1

（1）选择器

选择器根据 ADnSCM，ADnCTC，ADnCHEN，ADnCTL0，ADnTSEL，ADnCH1，ADnCH2，ADLTS1，ADLTS2 和 ADnOCKS 寄存器设置的模式来选择模拟输入引脚，并将输入送入采样&保持电路（n = 0，1）。

ANI05，ANI15 至 ANI17 内置用于输入电平放大的运算放大器框图和过电压检测比较器。每个模拟输入引脚的运算放大器和比较器可以指定为开启或关闭。ANI05，ANI15 至 ANI17 的运算放大器放大倍数（增益）可以选择为 2.5 至 10 倍。

（2）采样&保持电路

采样&保持电路顺序从每个输入电路采样模拟输入电压，并送入电压比较器。当使用输入电压放大的运算放大器时，采样到的是由 OPnCTL0.OPnGA3 至 OPnCTL0.OPnGA0 位指定的增益 × 输入电压。在 A/D 转换期间，该电路还保持采样到的模拟输入电压。

（3）电压比较器

该比较器将阵列的电压档位器产生的电压和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_{REFPn}），逐次逼近寄存器（SAR）的最高有效位（MSB）被置位。如果模拟输入电压小于参考电压（1/2 AV_{REFPn}），SAR 的 MSB 重置。

接下来，逐次逼近寄存器（SAR）的第 10 位自动置位，进行下一次比较。阵列的电压档位器由第 11 位的值选择，上次的比较结果已经设置到第 11 位内。

第 11 位 = 0：（1/4 AV_{REFPn}）
第 11 位 = 1：（3/4 AV_{REFPn}）

阵列的电压档位器和模拟输入电压进行比较，根据比较结果，操作 SAR 的第 10 位。

模拟输入电压 ≥ 阵列的电压档位器：第 10 位 = 1
模拟输入电压 ≤ 阵列的电压档位器：第 10 位 = 0

如此继续比较，直到 SAR 的第 0 位。

（4）阵列

阵列为模拟输入引脚产生比较电压输入。

（5）逐次逼近寄存器（SAR）

SAR 是 12 位寄存器，用于存入电压档位器的数据，这个来自阵列的值和模拟输入引脚的电压值相一致，从最高有效位（MSB）开始一次设置一位。

如果持续向 SAR 内设置数据，直到最低有效位（LSB）（A/D 转换结束），SAR 的内容（转换结果）保持在 A/Dn 转换结果寄存器 0 至 15（ADnCR0 至 ADnCR15）（n = 0，1）。但是，在扩展缓冲模式下，转换结果被存储在 A/Dn 转换结果扩展缓冲寄存器 0 至 4，并且当产生选择装载触发 x 时，转移被存储在 ADnECR0 至 ADnECR4 寄存器（x = 1，2）。当所有指定的 A/D 转换操作结束后，，产生一个 A/Dn 转换结束中断请求信号（INTADn）。

（6）A/Dn 转换结果寄存器 0 至 15（ADnCR0 至 ADnCR15），A/Dn 转换结果寄存器 0H 至 15H（ADnCR0H 至 ADnCR15H）（n = 0，1）

ADnCR0 至 ADnCR15 和 ADnCR0H 至 ADnCR15H 寄存器是保持 A/D 转换结果的寄存器。每次 A/D 转换结束，转换结果从逐次逼近寄存器（SAR）装载，并存储在 ADnCR0 至 ADnCR15 寄存器的高 12 位。当读取时，这些寄存器的低 4 位总是 0。

A/D 转换结果的高 8 位从 ADnCR0H 至 ADnCR15H 寄存器读取。

要按照 16 位宽度读取 A/D 转换结果，指定 ADnCR0 至 ADnCR15 寄存器。要读取高 8 位，指定 ADnCR0H 至 ADnCR15H 寄存器。

（7）A/Dn 转换结果扩展寄存器 0 至 4（ADnECR0 至 ADnECR4），A/Dn 转换结果扩展寄存器 0H 至 4H（ADnECR0H 至 ADnECR4H）（n = 0，1）

ADnECR0 至 ADnECR4 和 ADnECR0H 至 ADnECR4H 寄存器是保存 A/D 转换结果的寄存器。只有在扩展缓冲模式下，才可以使用这些寄存器。当 A/D 转换完成后，A/D 转换结果存储在 A/Dn 转换结果扩展缓冲寄存器内。如果在此之后产生了选择装载触发 1，为了存储，A/D 转换结果从 A/Dn 转换结果扩展缓冲寄存器 0 至 2 移位到 ADnECR0 至 ADnECR2 寄存器的高 12 位。当读取时，第 1 位至第 3 位总是 0。如果产生选择装载触发 2，A/D 转换结果从 A/Dn 转换结果扩展缓冲寄存器 3 和 4 移位到 ADnECR3 和 ADnECR4 寄存器的高 12 位。当读取时，第 1 位至第 3 位总是 0。

A/D 转换结果的高 8 位从 ADnECR0H 至 ADnECR4H 寄存器读取。

要按照 16 位宽度读取 A/D 转换结果，指定 ADnECR0 至 ADnECR4 寄存器。要读取高 8 位，指定 ADnECR0H 至 ADnECR4H 寄存器。

（8）ANI00 至 ANI05，ANI10 至 ANI17 引脚（n = 0，1）

ANI00 至 ANI05 和 ANI10 至 ANI17 引脚是 A/D 转换器 0 和 1 的模拟输入引脚。由它们输入将要进行 A/D 转换的模拟信号。

注意事项 **请确保输入到 ANI00 至 ANI05 和 ANI10 至 ANI17 引脚的电压不能超过额定值。如果一个大于等于 AV_{REFPn} 的电压或低于等于 AV_{SSn}（即使在绝对最大额度范围内）的电压输入到某个通道，通道的转换值不确定，并且其它通道的转换值可能也会受到影响。**

（9）AV_{REFPn} 引脚（n = 0，1）

该引脚用于输入 A/D 转换器 0 和 1 的参考电压。基于 AV_{REFPn} 和 AV_{SSn}（n = 0，1）之间的电压，它将输入到模拟输入引脚的信号转换为数字信号。

总是使该引脚的电势与 EV_{DD0}，EV_{DD1} 和 EV_{DD2}（仅限 V850E/IG3）引脚的电势相同，即使未使用 A/D 转换器 0 和 1。

AV_{REFPn} 引脚的操作电压范围是 $EV_{DD0} = EV_{DD1} = EV_{DD2}$（仅限 V850E/IG3）$= AV_{DDn} = AV_{REFPn} = 4.0$ 至 5.5 V.

（10） AV_{SSn} 引脚（n = 0，1）

这是 A/D 转换器 0 和 1 的地引脚。总是使该引脚的电势与 EV_{SS0}，EV_{SS1} 和 EV_{SS2}（仅限 V850E/IG3）引脚的电势相同，即使未使用 A/D 转换器 0 和 1。

（11） AV_{DDn} 引脚（n = 0，1）

该引脚是 A/D 转换器 0 和 1 的模拟供电电源引脚。

向 AV_{DD0} 和 AV_{DD1} 引脚提供相同的电势。

总是使该引脚的电势和引脚与 EV_{DD0}，EV_{DD1} 和 EV_{DD2}（仅限 V850E/IG3）引脚的电势相同，即使未使用 A/D 转换器 0 和 1。

AV_{DDn} 引脚的操作电压范围是 $EV_{DD0} = EV_{DD1} = EV_{DD2}$（仅限 V850E/IG3）$= AV_{REFPn} = AV_{DDn} = 4.0$ 至 5.5 V.

（12） CREFnL，CREFnF 引脚（n = 0，1）

CREFnL 引脚为过电压检测提供比较器的参考电压的小范围，CREFnF 引脚提供比较器的参考电压的全范围（CREF0L 和 CREF1L 引脚的输入电压范围 = $0.02AV_{DD} + 0.1$ 至 $0.5AV_{DD} - 0.1$ V，CREF0F 和 CREF1F 引脚的输入范围 = $0.02AV_{DD} + 0.1$ 至 $0.92AV_{DD} - 0.1$ V）.

12.3 控制寄存器

A/D 转换器 0 和 1 由下列寄存器控制。

- A/D 转换器 n 扫描模式寄存器（ADnSCM）
- A/D 转换器 n 扫描模式寄存器 L（ADnSCML）
- A/D 转换器 n 扫描模式寄存器 H（ADnSCMH）
- A/D 转换器 n 转换时间控制寄存器（ADnCTC）
- A/D 转换器 n 转换通道指定寄存器（ADnCHEN）
- A/D 转换器 n 转换通道指定寄存器 L（ADnCHENL）
- A/D 转换器 n 转换通道指定寄存器 H（ADnCHENH）
- A/D 转换器 n 控制寄存器（ADnCTL0）
- A/D 转换器 n 触发选择寄存器（ADnTSEL）
- A/D 转换器 n 通道指定寄存器 1 和 2（ADnCH1，ADnCH2）
- A/D 转换器 n 标志寄存器（ADnFLG）
- A/D 转换器 n 标志缓冲寄存器（ADnFLGB）
- A/DLDTRG1 输入选择寄存器（ADLTS1）
- A/DLDTRG2 输入选择寄存器（ADLTS2）
- 运算放大器 n 控制寄存器 0（OPnCTL0）
- 比较器 n 控制寄存器 0 至 3（CMPnCTL0 至 CMPnCTL3）
- A/D 转换器 n 时钟选择寄存器（ADnOCKS）
- A/D 触发下降沿指定寄存器（ADTF）
- A/D 触发上升沿指定寄存器（ADTR）
- 比较器输出中断上升沿指定寄存器（CMPOR）
- 比较器输出中断下降沿指定寄存器（CMPOF）
- 比较器输出数字噪声消除寄存器 nL，nF（CMPNFCnL，CMPNFCnF）

也会使用下列寄存器。

- A/Dn 转换结果寄存器 0 至 15（ADnCR0 至 ADnCR15）
- A/Dn 转换结果寄存器 0H 至 15H（ADnCR0H 至 ADnCR15H）
- A/Dn 转换结果扩展寄存器 0 至 4（ADnECR0 至 ADnECR4）
- A/Dn 转换结果扩展寄存器 0H 至 4H（ADnECR0H 至 ADnECR4H）

（1）A/D 转换器 n 扫描模式寄存器（ADnSCM）

ADnSCM 寄存器用于指定正常操作模式并控制转换操作。

该寄存器可按照 16 位宽度进行读取或写入。

当 ADnSCM 寄存器的高 8 位用作 ADnSCMH 寄存器，低 8 位用作 ADnSCML 寄存器时，这些寄存器可以按位或按字节读取或写入。但是第 14 位是只读的。

系统复位后，该寄存器设为 0000H。

（1/2）

复位后： 0000H R/W 地址：AD0SCM FFFFF220H, AD1SCM FFFFF2A0H

	<15>	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADnSCM (n = 0, 1)	ADnCE	ADnCS	0	0	0	ADnPLM	ADnTRG1	ADnTRG0	ADnPS	0	0	0	0	0	0[注1]	0

注 1. 使用A/D转换器 1 和0时，请确保将第1位设置为"1"。
该设置可以和ADnSCM寄存器的其它位同时执行。

ADnCE	A/D 转换操作控制
0	停止转换操作
1	开始转换操作

ADnCS	A/D 转换器 n[注2]的状态
0	A/D 转换停止
1	A/D 转换操作中 (在连续转换期间，即使通道被改变，也保持“1” 。)

ADnPLM	ADnTRG1	ADnTRG0	正常操作模式规范
0	0	0	A/D 触发模式
0	0	1	硬件触发模式[注3]
1	0	0	A/D 触发轮询模式
其它情况			禁止设置

ADnPS	A/D 省电模式规范
0	A/D 省电模式
1	A/D 操作模式

注 **2.** 在 AdnCE 位设置为 1 后，从 A/D 转换启动开始的五个基本时钟（f_{AD01}），ADnCS 位置为 1。
当有触发信号时，启动 A/D 转换，比如，在硬件触发模式、转换通道指定模式，或扩展缓冲模式下输入一个来自定时器的触发信号。在 A/D 触发模式和 A/D 触发轮询模式下，当 ADnCE 位为 1 时，启动转换。

3. 在扩展操作模式（转换通道指定模式或扩展缓冲模式）下，请确保设置硬件触发模式。

（2/2）

注意事项 **1.** 在 **A/D** 触发模式或 **A/D** 触发轮询模式下，当 **ADnCE** 位写入 **1** 时，触发转换。
在硬件触发模式，转换通道指定模式，或扩展缓冲模式下，当 **ADnCE** 位写入 **1** 时，启动触发信号等待状态。
在所有模式下，即使 **A/Dn** 转换结束中断请求信号（**INTADn**）产生之后，**ADnCE** 位不能清除为 **0**。因此，要停止 **A/D** 转换操作，向 **ADnCE** 位写入 **0**。

2. 在 **A/D** 转换操作期间（**ADnCS** 位 **= 1**），如果写入 **ADnSCM0** 寄存器，在各个模式下执行的操作如下。在 **A/D** 转换操作期间，对应的转换结果寄存器不确定。
- 在 **A/D** 触发模式下，**A/D** 触发轮询模式
  A/D 转换停止，且再次重头执行。
- 在硬件触发模式，转换通道指定模式，或扩展缓冲模式下
  A/D 转换停止，且再次恢复触发待机状态。

3. 请确保当转换操作使能（**ADnCE** 位 **= 1**）时，在连续向 **ADnSCM0** 寄存器写入数据之前，至少经过五个基本时钟（f_{AD01}）。否则，寄存器可能无法正确设置。
当 **ADnCE** 位 **= 0** 时，**ADSCMn** 寄存器写入之后，如果 **ADnCE** 位置为 **1**，该寄存器可以连续写入。

4. 在连续转换期间，当更改转换通道时，**ADnCS** 位保持置位（**1**）。

5. 当不使用 **A/D** 转换器时，推荐设置为 **A/D** 省电模式（**ADnPS** 位 **= 0**）。

6. 当 **A/D** 转换操作启动后（在复位解除后，从 **A/D** 省电模式（**ADnPS** 位 **= 0**）恢复之后），设置步骤如下。
<1> 通过 **ADnOCKS** 寄存器选择一个输入时钟（f_{AD01}），设置 **ADnOCKSEN** 位为 **1**（使能向 **A/D** 转换器 **n** 供给操作时钟）。
<2> 设置 **ADnPS** 位为 **1**（**A/D** 操作模式）。
<3> 在**<2>**之后，等待 **1** μ**s** 或更长时间。
<4> 初始化 **A/D** 转换器 **0** 和 **1**。
<5> 设置 **ADnCE** 位为 **1**（使能转换操作）。

7. 当 **A/D** 转换操作停止时，设置步骤如下。
<1> 设置 **ADnCE** 位为 **0**（停止转换操作）。
<2> 设置 **ADnPS** 位为 **0**（**A/D** 省电模式）。
<3> 设置 **ADnOCKS.ADnOCKSEN** 位为 **0**（停止向 **A/D** 转换器 **n** 供给操作时钟）。

8. 在 **IDLE** 和 **STOP** 模式下，推荐设置为 **A/D** 省电模式。
通过复位信号解除 **IDLE** 或 **STOP** 模式时，请遵循上述注意事项 **6** 中的设置步骤。

9. 请确保设置第 **0** 位至第 **6** 位和第 **11** 位至第 **13** 位为**“0”**。

（2）A/D 转换器 n 转换时间控制寄存器（ADnCTC）

ADnCTC 寄存器用于指定 A/D 转换时钟的数量和 A/D 转换时间。

该寄存器可以按字节或按位进行读取或写入。

系统复位后，该寄存器设为 00H。

复位后：00H R/W 地址：AD0CTC FFFFF222H, AD1CTC FFFFF2A2H

	7	6	5	4	3	2	1	0
ADnCTC (n = 0, 1)	0	0	0	0	ADnFR3	ADnFR2	ADnFR1	ADnFR0

注意事项 **1.** 关于 **ADnFR3** 至 **ADnFR0** 位，参见表 **12-2 A/D** 转换时钟的数量和 **A/D** 转换时间。

2. 当 **ADnSCM.ADnCE** 位 **= 0**（转换操作停止）时，设置 **ADnFR3** 至 **ADnFR0** 位。

3. 请确保设置第 **4** 位至第 **7** 位为 **"0"**。

表 12-2. A/D 转换时钟的数量和 A/D 转换时间

ADnFR3	ADnFR2	ADnFR1	ADnFR0	A/D 转换时钟的数量注 1	A/D 转换时间（μs）注 2	
					f_{AD01} = 16 MHz（f_{XX} = 64 MHz）	f_{AD01} = 12 MHz（f_{XX} = 48 MHz）
0	0	0	0	89	5.56	7.42
0	0	0	1	88	5.50	7.33
0	0	1	0	57	3.56	4.75
0	0	1	1	56	3.50	4.67
0	1	0	0	41	2.56	3.42
0	1	0	1	40	2.50	3.33
0	1	1	0	35	2.19	2.92
0	1	1	1	34	2.13	2.83
1	0	0	0	34	2.13	2.83
1	0	0	1	33	2.06	2.75
1	0	1	0	33	2.06	2.75
1	0	1	1	32	2.00	2.67
1	1	0	0	32	2.00	2.67
1	1	0	1	31	禁止设置	2.58
1	1	1	0	31	禁止设置	2.58
1	1	1	1	30	禁止设置	2.50

注 **1.** 时钟的数量（f_{AD01}）从 A/D 转换的启动到结束。
在连续转换（1-通道转换（重复），多个通道转换，或者多个通道转换（重复））期间，每次转换的时钟的数量（f_{AD01}）相同。

2. 设置 A/D 转换时间在 2 至 7.42 μs 范围内。
A/D 转换时间= 1/f_{AD01} × A/D 转换时钟的数量

（3）A/D 转换器 n 转换通道指定寄存器（ADnCHEN）

ADnCHEN 寄存器用于指定模拟输入引脚，转换次数，和转换结果寄存器。

在 A/D 触发模式，A/D 触发轮询模式和硬件触发模式下，该寄存器用于指定模拟输入引脚。按照一对一的基础，ADnCRm 寄存器对应一路模拟输入引脚。使用对应 ANI00 至 ANI05 和 ANI10 至 ANI17 引脚（AD0CHEN00 至 AD0CHEN05 和 AD1CHEN00 至 AD1CHEN07）的位。如果指定了两路或更多模拟输入引脚，它们将依次选择，转换从最小数字开始（当 AD1CHEN 寄存器 = 004DH 时：ANI10 → ANI12 → ANI13 → ANI16）。在连续转换期间，如果没有指定某个模拟输入引脚，则跳过。

在转换通道指定模式下，指定转换次数和转换结果寄存器。通过 ADnCH1 寄存器来指定一路模拟输入引脚。设置到 ADnCHEN 寄存器低位的值，调整到最低位，是转换次数。按照一对一的基础，这些位对应 ADnCRm 和 ADnCHmH 寄存器。

因为 ADnCHEN 寄存器是主/从配置，在 A/D 转换操作期间，一路新的模拟输入引脚可以设置到主寄存器。在 A/D 转换（在产生 A/Dn 转换结束中断请求信号（INTADn）之后）完成之后，主寄存器的设置值传输到从寄存器。

该寄存器可按照 16 位宽度进行读取或写入。

当 ADnCHEN 寄存器的高 8 位用作 ADnCHENH 寄存器，低 8 位用作 ADnCHENL 寄存器时，这些寄存器可以按位或按字节读取或写入。

系统复位后，该寄存器设为 0000H。

复位后： 0000H R/W 地址：AD0CHEN FFFFF224H, AD1CHEN FFFFF2A4H

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADnCHEN (n = 0, 1)	ADn CHEN 15	ADn CHEN 14	ADn CHEN 13	ADn CHEN 12	ADn CHEN 11	ADn CHEN 10	ADn CHEN 9	ADn CHEN 8	ADn CHEN 7	ADn CHEN 6	ADn CHEN 5	ADn CHEN 4	ADn CHEN 3	ADn CHEN 2	ADn CHEN 1	ADn CHEN 0

备注 关于如何在 A/D 触发模式，A/D 触发轮询模式和硬件触发模式下指定模拟输入引脚，参见表 **12-3 在 A/D 触发模式，A/D 触发轮询模式和硬件触发模式下指定模拟输入引脚**。关于如何在转换通道指定模式下，指定转换次数和 A/D 转换结果寄存器，参见表 **12-4 转换通道指定模式下，ADnCHEN 寄存器的设置值与转换次数和 A/D 转换结果寄存器之间的对应关系**。

注意事项 **1.** 当 **ADnCHEN** 寄存器 **= 0000H** 时，禁止 **A/D** 转换操作。
如果 **ADnCHEN** 寄存器 **= 0000H**，操作和 **ADnCHEN** 寄存器 **= 0001H** 相同。

2. 当 **ADSCMn.ADnPS** 位 **= 0** 时，不要写入 **ADnCHEN** 寄存器。如果写入，则 **CPU** 死锁。

3. 在硬件触发模式下，当 **ADnSCM.ADnCE** 位 **= 1** 时，要改变 **ADnCHEN** 寄存器的设置，请确保设置 **ADnCE** 位为 **0**。

表 12-3. 在 A/D 触发模式，A/D 触发轮询模式和硬件触发模式下指定模拟输入引脚

ADnCHENm 位	模拟输入引脚的指定
0	禁止指定 ANInk 引脚
1	使能指定 ANInk 引脚

备注　A/D 转换器 0：n = 0，k = 0 至 5，m = 0 至 15
A/D 转换器 1：n = 1，k = 0 至 7，m = 0 至 15

表 12-4. 转换通道指定模式下，ADnCHEN 寄存器的设置值与转换次数和 A/D 转换结果寄存器之间的对应关系

ADnCHEN 寄存器值	转换次数	A/D 转换结果寄存器	
0001H	1	ADnCR0	ADnCR0H
0003H	2	ADnCR0，ADnCR1	ADnCR0H，ADnCR1H
0007H	3	ADnCR0 至 ADnCR2	ADnCR0H 至 ADnCR2H
000FH	4	ADnCR0 至 ADnCR3	ADnCR0H 至 ADnCR3H
001FH	5	ADnCR0 至 ADnCR4	ADnCR0H 至 ADnCR4H
003FH	6	ADnCR0 至 ADnCR5	ADnCR0H 至 ADnCR5H
007FH	7	ADnCR0 至 ADnCR6	ADnCR0H 至 ADnCR6H
00FFH	8	ADnCR0 至 ADnCR7	ADnCR0H 至 ADnCR7H
01FFH	9	ADnCR0 至 ADnCR8	ADnCR0H 至 ADnCR8H
03FFH	10	ADnCR0 至 ADnCR9	ADnCR0H 至 ADnCR9H
07FFH	11	ADnCR0 至 ADnCR10	ADnCR0H 至 ADnCR10H
0FFFH	12	ADnCR0 至 ADnCR11	ADnCR0H 至 ADnCR11H
1FFFH	13	ADnCR0 至 ADnCR12	ADnCR0H 至 ADnCR12H
3FFFH	14	ADnCR0 至 ADnCR13	ADnCR0H 至 ADnCR13H
7FFFH	15	ADnCR0 至 ADnCR14	ADnCR0H 至 ADnCR14H
FFFFH	16	ADnCR0 至 ADnCR15	ADnCR0H 至 ADnCR15H
其它情况	禁止设置		

注意事项　在转换通道指定模式下，通过 **ADnCH1** 寄存器指定一路模拟输入引脚。

备注　n = 0，1

（4）A/Dn 转换结果寄存器 0 至 15，0H 至 15H（ADnCR0 至 ADnCR15，ADnCR0H 至 ADnCR15H）

ADnCRm 和 ADnCRmH 寄存器用于在 A/D 触发模式、A/D 触发轮询模式、硬件触发模式或转换通道指定模式下保持 A/D 转换结果。每个电路提供十六个寄存器，有两个电路可用。每当 A/D 转换结束时，转换结果从逐次逼近寄存器（SAR）装载，并存储在 ADnCRm 寄存器的高 12 位。当读取这些寄存器时，低 4 位总是为 0。

A/D 转换结果高 8 位读取到 ADnCRmH 寄存器。

这些寄存器只能按照 16 位宽度或按字节读取。当按照 16 位宽度读取 A/D 转换结果时，指定 ADnCRm 寄存器，当读取高 8 位时，指定 ADnCRmH 寄存器。

系统复位后，该寄存器设为 0000H。

备注　当 A/D 转换的结果存储在 ADnCRm 寄存器中，对相同寄存器的读取访问挂起。在 A/D 转换结果的存储完成之后，执行被挂起的读取访问。与此类似，当对该寄存器进行读取时，将 A/D 转换结果存储到 ADnCRm 寄存器的操作挂起。当读取访问完成之后，执行挂起的 A/D 转换结果的存储处理。

复位后：0000H　R　地址：AD0CR0 FFFFF200H, AD0CR1 FFFFF202H,
AD0CR2 FFFFF204H, AD0CR3 FFFFF206H,
AD0CR4 FFFFF208H, AD0CR5 FFFFF20AH,
AD0CR6 FFFFF20CH, AD0CR7 FFFFF20EH,
AD0CR8 FFFFF210H, AD0CR9 FFFFF212H,
AD0CR10 FFFFF214H, AD0CR11 FFFFF216H,
AD0CR12 FFFFF218H, AD0CR13 FFFFF21AH,
AD0CR14 FFFFF21CH, AD0CR15 FFFFF21EH,
AD1CR0 FFFFF280H, AD1CR1 FFFFF282H,
AD1CR2 FFFFF284H, AD1CR3 FFFFF286H,
AD1CR4 FFFFF288H, AD1CR5 FFFFF28AH,
AD1CR6 FFFFF28CH, AD1CR7 FFFFF28EH,
AD1CR8 FFFFF290H, AD1CR9 FFFFF292H,
AD1CR10 FFFFF294H, AD1CR11 FFFFF296H,
AD1CR12 FFFFF298H, AD1CR13 FFFFF29AH,
AD1CR14 FFFFF29CH, AD1CR15 FFFFF29EH

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADnCRm (n = 0, 1) (m = 0 至 15)	ADnCRm11	ADnCRm10	ADnCRm9	ADnCRm8	ADnCRm7	ADnCRm6	ADnCRm5	ADnCRm4	ADnCRm3	ADnCRm2	ADnCRm1	ADnCRm0	0	0	0	0

复位后：0000H　R　地址：AD0CR0H FFFFF201H, AD0CR1H FFFFF203H,
AD0CR2H FFFFF205H, AD0CR3H FFFFF207H,
AD0CR4H FFFFF209H, AD0CR5H FFFFF20BH,
AD0CR6H FFFFF20DH, AD0CR7H FFFFF20FH,
AD0CR8H FFFFF211H, AD0CR9H FFFFF213H,
AD0CR10H FFFFF215H, AD0CR11H FFFFF217H,
AD0CR12H FFFFF219H, AD0CR13H FFFFF21BH,
AD0CR14H FFFFF21DH, AD0CR15H FFFFF21FH,
AD1CR0H FFFFF281H, AD1CR1H FFFFF283H,
AD1CR2H FFFFF285H, AD1CR3H FFFFF287H,
AD1CR4H FFFFF289H, AD1CR5H FFFFF28BH,
AD1CR6H FFFFF28DH, AD1CR7H FFFFF28FH,
AD1CR8H FFFFF291H, AD1CR9H FFFFF293H,
AD1CR10H FFFFF295H, AD1CR11H FFFFF297H,
AD1CR12H FFFFF299H, AD1CR13H FFFFF29BH,
AD1CR14H FFFFF29DH, AD1CR15H FFFFF29FH

	7	6	5	4	3	2	1	0
ADnCRmH (n = 0, 1) (m = 0 至 15)	ADnCRm11	ADnCRm10	ADnCRm9	ADnCRm8	ADnCRm7	ADnCRm6	ADnCRm5	ADnCRm4

在 A/D 触发模式、A/D 触发轮询模式、硬件触发模式下，模拟输入引脚和 A/D 转换结果寄存器之间的对应关系展示如下。

表 12-5. A/D 触发模式、A/D 触发轮询模式、硬件触发模式下，模拟输入引脚和 A/D 转换结果寄存器之间的对应关系

A/D 转换器	模拟输入引脚	A/D 转换结果寄存器
A/D 转换器 0	ANI00	AD0CR0，AD0CR0H
	ANI01	AD0CR1，AD0CR1H
	ANI02	AD0CR2，AD0CR2H
	ANI03	AD0CR3，AD0CR3H
	ANI04	AD0CR4，AD0CR4H
	ANI05	AD0CR5，AD0CR5H
A/D 转换器 1	ANI10	AD1CR0，AD1CR0H
	ANI11	AD1CR1，AD1CR1H
	ANI12	AD1CR2，AD1CR2H
	ANI13	AD1CR3，AD1CR3H
	ANI14	AD1CR4，AD1CR4H
	ANI15	AD1CR5，AD1CR5H
	ANI16	AD1CR6，AD1CR6H
	ANI17	AD1CR7，AD1CR7H

表 12-6. 转换通道指定模式下，模拟输入引脚和 A/D 转换结果寄存器之间的对应关系

ADnCHEN 寄存器设置值	模拟输入引脚	A/D 转换结果寄存器	
0001H	由 ADnCH1.ADnTRGCH12 至 ADnCH1.ADnTRGCH10 位设置	ADnCR0	ADnCR0H
0003H		ADnCR0，ADnCR1	ADnCR0H，ADnCR1H
0007H		ADnCR0 至 ADnCR2	ADnCR0H 至 ADnCR2H
000FH		ADnCR0 至 ADnCR3	ADnCR0H 至 ADnCR3H
001FH		ADnCR0 至 ADnCR4	ADnCR0H 至 ADnCR4H
003FH		ADnCR0 至 ADnCR5	ADnCR0H 至 ADnCR5H
007FH		ADnCR0 至 ADnCR6	ADnCR0H 至 ADnCR6H
00FFH		ADnCR0 至 ADnCR7	ADnCR0H 至 ADnCR7H
01FFH		ADnCR0 至 ADnCR8	ADnCR0H 至 ADnCR8H
03FFH		ADnCR0 至 ADnCR9	ADnCR0H 至 ADnCR9H
07FFH		ADnCR0 至 ADnCR10	ADnCR0H 至 ADnCR10H
0FFFH		ADnCR0 至 ADnCR11	ADnCR0H 至 ADnCR11H
1FFFH		ADnCR0 至 ADnCR12	ADnCR0H 至 ADnCR12H
3FFFH		ADnCR0 至 ADnCR13	ADnCR0H 至 ADnCR13H
7FFFH		ADnCR0 至 ADnCR14	ADnCR0H 至 ADnCR14H
FFFFH		ADnCR0 至 ADnCR15	ADnCR0H 至 ADnCR15H
0001H	由 ADnCH1.ADnTRGCH16 至 ADnCH1.ADnTRGCH14 位设置	ADnCR0	ADnCR0H
0003H		ADnCR0，ADnCR1	ADnCR0H，ADnCR1H
0007H		ADnCR0 至 ADnCR2	ADnCR0H 至 ADnCR2H
000FH		ADnCR0 至 ADnCR3	ADnCR0H 至 ADnCR3H
001FH		ADnCR0 至 ADnCR4	ADnCR0H 至 ADnCR4H
003FH		ADnCR0 至 ADnCR5	ADnCR0H 至 ADnCR5H
007FH		ADnCR0 至 ADnCR6	ADnCR0H 至 ADnCR6H
00FFH		ADnCR0 至 ADnCR7	ADnCR0H 至 ADnCR7H
01FFH		ADnCR0 至 ADnCR8	ADnCR0H 至 ADnCR8H
03FFH		ADnCR0 至 ADnCR9	ADnCR0H 至 ADnCR9H
07FFH		ADnCR0 至 ADnCR10	ADnCR0H 至 ADnCR10H
0FFFH		ADnCR0 至 ADnCR11	ADnCR0H 至 ADnCR11H
1FFFH		ADnCR0 至 ADnCR12	ADnCR0H 至 ADnCR12H
3FFFH		ADnCR0 至 ADnCR13	ADnCR0H 至 ADnCR13H
7FFFH		ADnCR0 至 ADnCR14	ADnCR0H 至 ADnCR14H
FFFFH		ADnCR0 至 ADnCR15	ADnCR0H 至 ADnCR15H
其它情况	禁止设置		

备注 n = 0，1

（5）A/D 转换器 n 控制寄存器（ADnCTL0）

ADnCTL0 寄存器用于指定操作模式。

该寄存器可以按字节或按位进行读取或写入。

系统复位后，该寄存器设为 00H。

复位后：00H　R/W　地址：AD0CTL0 FFFFF230H, AD1CTL0 FFFFF2B0H

	7	6	5	4	3	2	1	0
ADnCTL0 (n = 0, 1)	0	0	0	0	0	0	ADnMD1	ADnMD0

ADnMD1	ADnMD0	扩展操作模式规范
0	0	正常操作模式
0	1	禁止设置
1	0	转换通道指定模式
1	1	扩展缓冲模式

注意事项 **1.** 当 **ADnSCM.ADnCE** 位 **= 0**（转换操作停止）时，设置 **ADnMD1** 和 **ADnMD0** 位。（当 **ADnCE** 位 **= 1**（转换操作使能）时，相同的值可以写入这些位）。

2. 在转换通道指定模式和扩展缓冲模式下，和正常操作模式相比，**A/D** 转换的启动最多会延迟 **1.5** 基本时钟（f_{AD01}）。

3. 请确保在转换通道指定模式和扩展缓冲模式下设置硬件触发模式。

（6）A/D 转换器 n 触发选择寄存器（ADnTSEL）

ADnTSEL 寄存器用于指定硬件触发模式和转换通道指定模式下的触发器，以及扩展缓冲模式下的触发器（选择触发 1，选择触发 2，选择装载触发 1 和选择装载触发 2）。

该寄存器可以按字节或按位进行读取或写入。

系统复位后，该寄存器设为 10H。

复位后：10H R/W 地址：AD0TSEL FFFFF231H, AD1TSEL FFFFF2B1H

	7	6	5	4	3	2	1	0
ADnTSEL (n = 0, 1)	ADnLDTSEL2 注	0	ADnTRGSEL21 注	ADnTRGSEL20 注	ADnLDTSEL1 注	0	ADnTRGSEL11	ADnTRGSEL10

ADnLDTSEL2 注	ADnECR3, ADnECR4 寄存器的选择装载触发器2的规范
0	LDTRG1
1	LDTRG2

ADnTRGSEL21 注	ADnTRGSEL20 注	ADnECR3, ADnECR4 寄存器的选择装载触发器2的规范
0	0	ITRG1
0	1	ITRG2
1	0	ITRG3
1	1	ITRG4

ADnLDTSEL1 注	ADnECR0 至 ADnECR2寄存器的选择装载触发器1的规范
0	LDTRG1
1	LDTRG2

ADnTRGSEL11	ADnTRGSEL10	• 在硬件触发模式或转换通道指定模式下： 触发器规范 • 在扩展缓冲模式下： ADnECR0 至 ADnECR2寄存器的选择装载触发器1的规范
0	0	ITRG1
0	1	ITRG2
1	0	ITRG3
1	1	ITRG4

注 在硬件触发模式和转换通道指定模式下，请确保将第 3 位、第 5 位和第 7 位清除为“0”，将第 4 位设置为“1”。

注意事项 **当 ADnSCM.ADnCE 位 = 0（转换操作停止）时，设置 ADnTSEL 寄存器（当 ADnCE 位 = 1（转换操作使能）时，相同的值可以写入该寄存器）。**

（7）A/D 转换器 n 通道指定寄存器 1（ADnCH1）

ADnCH1 寄存器用于指定模拟输入引脚，该引脚在转换通道指定模式和扩展缓冲模式下用作选择触发 1。

该寄存器可以按字节或按位进行读取或写入。

系统复位后，该寄存器设为 00H。

复位后： 00H R/W 地址：AD0CH1 FFFFF232H, AD1CH1 FFFFF2B2H

	7	6	5	4	3	2	1	0
ADnCH1 (n = 0, 1)	0	ADn TRGCH16	ADn TRGCH15	ADn TRGCH14	0	ADn TRGCH12	ADn TRGCH11	ADn TRGCH10

ADnTRGCH16	ADnTRGCH15	ADnTRGCH14	用作选择触发器1的模拟输入引脚的规范
0	0	0	ANIn0
0	0	1	ANIn1
0	1	0	ANIn2
0	1	1	ANIn3
1	0	0	ANIn4
1	0	1	ANIn5
1	1	0	ANI16
1	1	1	ANI17

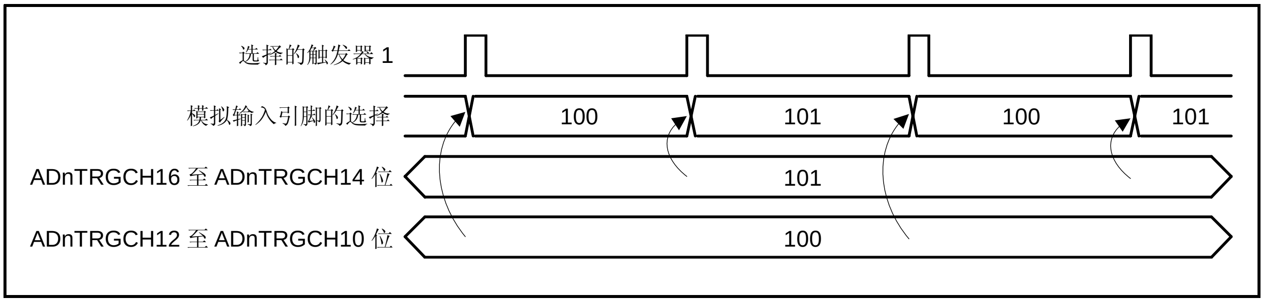
ADnTRGCH12	ADnTRGCH11	ADnTRGCH10	用作选择触发器1的模拟输入引脚的规范
0	0	0	ANIn0
0	0	1	ANIn1
0	1	0	ANIn2
0	1	1	ANIn3
1	0	0	ANIn4
1	0	1	ANIn5
1	1	0	ANI16
1	1	1	ANI17

注意事项 **1.** 当 **ADnSCM.ADnCE** 位 **= 0**（转换操作停止）时，设置 **ADnCH1** 寄存器（当 **ADnCE** 位 **= 1**（转换操作使能）时，相同的值可以写入该寄存器）。

2. 请确保将第 **3** 位至第 **7** 位清除为 **“0”**。

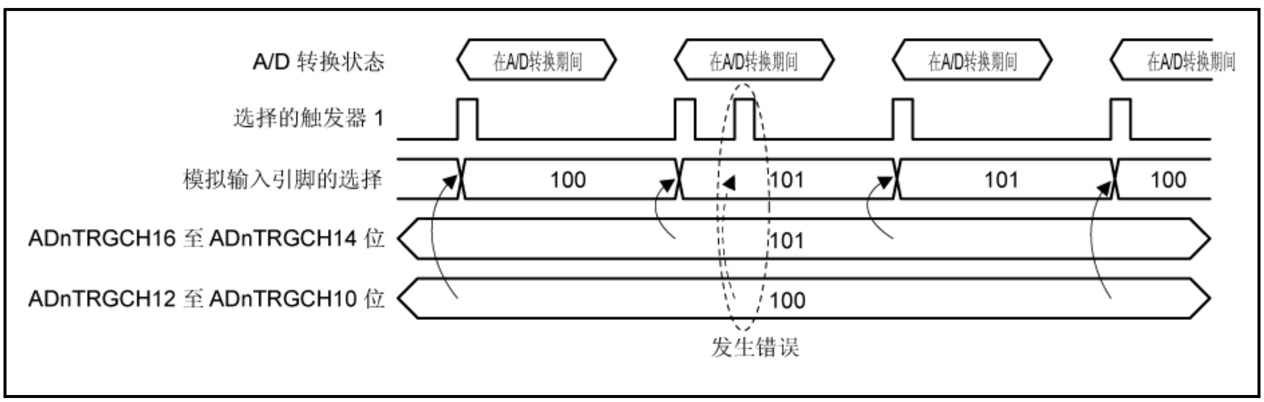
在转换通道指定模式或扩展缓冲模式下，当转换操作使能时（ADnSCM.ADnCE 位 = 1），可以设置 ADnCH1 寄存器。在转换操作使能（ADnCE 位 = 1）之后，第一个选择触发 1 产生时，通过选择 ADnTRGCH12 至 ADnTRGCH10 位指定的模拟输入引脚，并执行 A/D 转换。当下一个选择触发 1 产生时，通过选择 ADnTRGCH16 至 ADnTRGCH14 位指定的模拟输入引脚，并执行 A/D 转换。此后，每次产生选择触发 1 时，用于输出的模拟输入引脚交替选择。

图 12-6. ADnCH1 寄存器操作



如果发生错误（在 A/D 转换期间产生选择触发 1），通过 ADnTRGCH12 至 ADnTRGCH10 位指定的模拟输入引脚和通过 ADnTRGCH16 至 ADnTRGCH14 位指定的模拟输入引脚交替选择，但是所选中的模拟输入引脚不会改变，因为 A/D 转换正在进行中。

图 12-7. 发生错误时的 ADnCH1 寄存器操作



（8）A/D 转换器 n 通道指定寄存器 2（ADnCH2）

ADnCH2 寄存器用于在扩展缓冲模式下为选择触发 2 指定模拟输入引脚。

该寄存器可以按字节或按位进行读取或写入。

系统复位后，该寄存器设为 00H。

复位后：00H R/W 地址： AD0CH2 FFFFF233H, AD1CH2 FFFFF2B3H

	7	6	5	4	3	2	1	0
ADnCH2 (n = 0, 1)	0	ADn TRGCH26	ADn TRGCH25	ADn TRGCH24	0	ADn TRGCH22	ADn TRGCH21	ADn TRGCH20

ADnTRGCH26	ADnTRGCH25	ADnTRGCH24	用作选择触发器2的模拟输入引脚的规范
0	0	0	ANIn0
0	0	1	ANIn1
0	1	0	ANIn2
0	1	1	ANIn3
1	0	0	ANIn4
1	0	1	ANIn5
1	1	0	ANI16
1	1	1	ANI17

ADnTRGCH22	ADnTRGCH21	ADnTRGCH20	用作选择触发器2的模拟输入引脚的规范
0	0	0	ANIn0
0	0	1	ANIn1
0	1	0	ANIn2
0	1	1	ANIn3
1	0	0	ANIn4
1	0	1	ANIn5
1	1	0	ANI16
1	1	1	ANI17

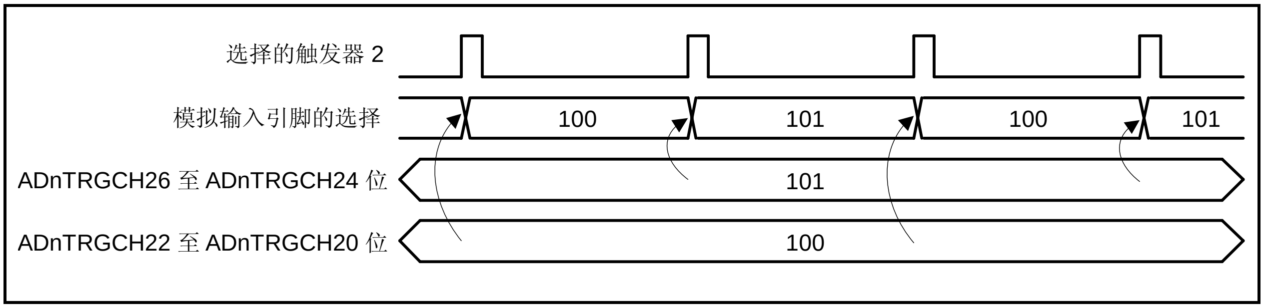
注意事项 **1.** 当 **ADnSCM.ADnCE** 位 **= 0**（转换操作停止）时，设置 **ADnCH2** 寄存器（当 **ADnCE** 位 **= 1**（转换操作使能）时，相同的值可以写入该寄存器）。

2. 只有在扩展缓冲模式下，**ADnCH2** 寄存器有效。在其它模式下无效。

3. 请确保将第 **3** 位至第 **7** 位清除为**“0”**。

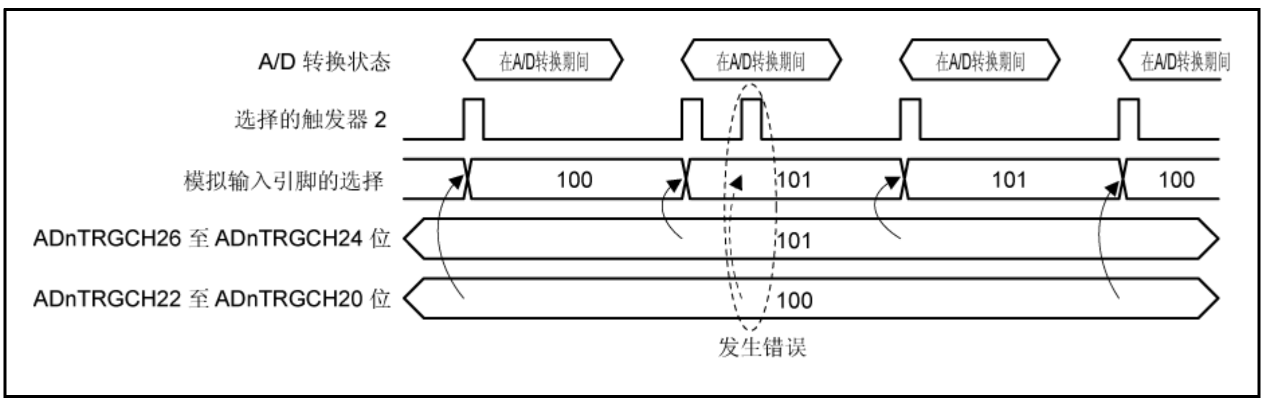
在扩展缓冲模式下，当转换操作使能时（ADnSCM.ADnCE 位 = 1），可以设置 ADnCH2 寄存器。在转换操作使能（ADnCE 位 = 1）之后，第一个选择触发 2 产生时，通过选择 ADnTRGCH22 至 ADnTRGCH20 位指定的模拟输入引脚，并执行 A/D 转换。当下一个选择触发 2 产生时，通过选择 ADnTRGCH26 至 ADnTRGCH24 位指定的模拟输入引脚，并执行 A/D 转换。此后，每次产生选择触发 2 时，用于输出的模拟输入引脚交替选择。

图 12-8. ADnCH2 寄存器操作



如果发生错误（在 A/D 转换期间产生选择触发 2），通过 ADnTRGCH22 至 ADnTRGCH20 位指定的模拟输入引脚和通过 ADnTRGCH26 至 ADnTRGCH24 位指定的模拟输入引脚交替选择，但是所选中的模拟输入引脚不会改变，因为 A/D 转换正在进行中。

图 12-9. 发生错误时的 ADnCH2 寄存器操作



（9）A/Dn 转换结果扩展寄存器 0 至 4，0H 至 4H（ADnECR0 至 ADnECR4，ADnECR0H 至 ADnECR4H）

ADnECRa 和 ADnECRaH 寄存器在它们的高 12 位中保持 A/D 转换结果，并在扩展缓冲模式下使用最低 1 位来指示 A/D 转换结果的状态（关于模拟输入引脚的 A/D 转换结果的信息，该通道通过 ADnCHx.ADnTRGCHx2 至 ADnTRGCHx0 位或 ADnTRGCHx6 至 ADnTRGCHx4 位指定）。每个电路提供五个寄存器，有两个电路可用。当 A/D 转换结束时，A/D 转换结果存储到 A/Dn 转换结果扩展缓冲寄存器 a。之后当选择装载触发 1 产生时，A/D 转换结果从 A/Dn 转换结果扩展缓冲寄存器 0 至 2 转移并存储到 ADnECR0 至 ADnECR2 寄存器的高 12 位。当读取这些寄存器时，第 1 位至第 3 位总是为 0。当选择装载触发 2 产生时，A/D 转换结果从 A/Dn 转换结果扩展缓冲寄存器 3 和 4 转移并存储到 ADnECR3 和 ADnECR4 寄存器高 12 位。当读取这些寄存器时，第 1 位至第 3 位总是为 0。

A/D 转换结果的高 8 位从 ADnECRaH 寄存器读取。

这些寄存器只能按照 16 位宽度或按字节读取。要按照 16 位宽度读取 A/D 转换结果，指定 ADnECRa 寄存器。指定 ADnECRaH 寄存器可以读取 A/D 转换结果的高 8 位。

系统复位后，该寄存器设为 0000H。

备注　当 A/D 转换的结果存储在 ADnECRa 寄存器中，对相同寄存器的读取访问挂起。在 A/D 转换结果的存储完成之后，执行被挂起的读取访问。与此类似，当对该寄存器进行读取时，将 A/D 转换结果存储到 ADnECRa 寄存器的操作挂起。当读取访问完成之后，执行挂起的 A/D 转换结果的存储处理。

复位后：0000H R 地址： AD0ECR0 FFFFF240H, AD0ECR1 FFFFF242H,
AD0ECR2 FFFFF244H, AD0ECR3 FFFFF246H,
AD0ECR4 FFFFF248H,
AD1ECR0 FFFFF2C0H, AD1ECR1 FFFFF2C2H,
AD1ECR2 FFFFF2C4H, AD1ECR3 FFFFF2C6H,
AD1ECR4 FFFFF2C8H

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADnECRa (n = 0, 1) (a = 0 至 4)	ADnECRa11	ADnECRa10	ADnECRa9	ADnECRa8	ADnECRa7	ADnECRa6	ADnECRa5	ADnECRa4	ADnECRa3	ADnECRa2	ADnECRa1	ADnECRa0	0	0	0	ADnCHFLGa

ADnCHFLGa	A/D 转换结果的状态 (x = 1, 2)
0	由ADnCHx.ADnTRGCHx2至ADnCHx.ADnTRGCHx0 位设置的模拟输入引脚的A/D转换结果。
1	由ADnCHx.ADnTRGCHx6至 ADnCHx.ADnTRGCHx4 位设置的模拟输入引脚的A/D转换结果。

复位后：00H R 地址： AD0ECR0H FFFFF241H, AD0ECR1H FFFFF243H,
AD0ECR2H FFFFF245H, AD0ECR3H FFFFF247H,
AD0ECR4H FFFFF249H,
AD1ECR0H FFFFF2C1H, AD1ECR1H FFFFF2C3H,
AD1ECR2H FFFFF2C5H, AD1ECR3H FFFFF2C7H,
AD1ECR4H FFFFF2C9H

	7	6	5	4	3	2	1	0
ADnECRaH (n = 0, 1) (a = 0 至 4)	ADnECRa11	ADnECRa10	ADnECRa9	ADnECRa8	ADnECRa7	ADnECRa6	ADnECRa5	ADnECRa4

注意事项 只有在扩展缓冲模式下，**ADnECRa** 和 **ADnECRaH** 寄存器有效。在其它模式下无效。

模拟输入引脚和 A/D 转换结果扩展寄存器之间的对应关系展示如下。

表 12-7. 模拟输入引脚和 A/D 转换结果扩展寄存器之间的对应关系

模拟输入引脚	A/Dn 转换结果寄存器
使用 ADnCH1 寄存器的 ADnTRGCH12 至 ADnTRGCH10，ADnTRGCH16 至 ADnTRGCH14 位设置	ADnECR0，ADnECR0H
	ADnECR1，ADnECR1H
	ADnECR2，ADnECR2H
使用 ADnCH2 寄存器的 ADnTRGCH22 至 ADnTRGCH20，ADnTRGCH26 至 ADnTRGCH24 位设置	ADnECR3，ADnECR3H
	ADnECR4，ADnECR4H

备注 n = 0，1

（10） A/D 转换器 n 标志寄存器（ADnFLG）

ADnFLG 寄存器指示在扩展缓冲模式下产生选择装载触发 x 时（x = 1 或 2），发生错误。只有当转换操作停止（ADnSCM.ADnCE 位 = 0）时，ADnTERR2 和 ADnTERR1 标志才可以读取和清除。

该寄存器只能按字节进行读取。

系统复位后，该寄存器设为 00H。

复位后：00H R 地址：AD0FLG FFFFF254H, AD1FLG FFFFF2D4H

	7	6	5	4	3	2	1	0
ADnFLG (n = 0, 1)	0	0	0	0	0	0	ADnTERR2注	ADnTERR1注

ADnTERR2注	选择装载触发器2的发生时序错误标志
0	没有发生选择装载触发器2的发生时序错误
1	已经发生选择装载触发器2的发生时序错误

ADnTERR1注	选择装载触发器1的发生时序错误标志
0	没有发生选择装载触发器1的发生时序错误
1	已经发生选择装载触发器1的发生时序错误

注 只有在扩展缓冲模式下，ADnTERR2 和 ADnTERR1 标志有效。在其它模式下，他们固定为 0。

（11） A/D 转换器 n 标志缓冲寄存器（ADnFLGB）

ADnFLGB 寄存器指示在扩展缓冲模式下产生选择装载触发 x 时（x = 1 或 2），发生错误。只有当转换操作停止（ADnSCM.ADnCE 位 = 0）时，ADnTERRB2 和 ADnTERRB1 标志才可以读取和清除。

该寄存器只能按字节进行读取。

系统复位后，该寄存器设为 00H。

复位后：00H R 地址：AD0FLGB FFFFF255H, AD1FLGB FFFFF2D5H

	7	6	5	4	3	2	1	0
ADnFLGB (n = 0, 1)	0	0	0	0	0	0	ADnTERRB2[注]	ADnTERRB1[注]

ADnTERRB2[注]	选择装载触发器2的发生时序错误标志
0	没有发生选择装载触发器2的发生时序错误
1	已经发生选择装载触发器2的发生时序错误

ADnTERRB1[注]	选择装载触发器1的发生时序错误标志
0	没有发生选择装载触发器1的发生时序错误
1	已经发生选择装载触发器1的发生时序错误

注 只有在扩展缓冲模式下，ADnTERRB2 和 ADnTERRB1 标志有效。在其它模式下，他们固定为 0。

（12） A/D LDTRG1 输入选择寄存器（ADLTS1）

ADLTS1 寄存器用于在扩展缓冲模式下为选择装载触发器（LDTRG1）指定输入信号。

该寄存器可以按字节进行读取或写入。

系统复位后，该寄存器设为 00H。

复位后： 00H R/W 地址： FFFFF2F8H

	7	6	5	4	3	2	1	0
ADLTS1	0	0	0	0	0	0	0	ADLTS10

ADLTS10	LDTRG1的输入信号规范
0	TABTIOV0 信号
1	TABTIOV1 信号

注　只有在扩展缓冲模式下，ADLTS1 寄存器有效。在其它模式下无效。

（13） A/D LDTRG2 输入选择寄存器（ADLTS2）

ADLTS2 寄存器用于在扩展缓冲模式下为选择装载触发器（LDTRG2）指定输入信号。

该寄存器可以按字节进行读取或写入。

系统复位后，该寄存器设为 00H。

复位后：00H R/W 地址： FFFFF2FAH

	7	6	5	4	3	2	1	0
ADLTS2	0	0	0	0	0	0	0	ADLTS20

ADLTS20	LDTRG2的输入信号规范
0	TABTICC00 信号
1	TABTICC10 信号

注　只有在扩展缓冲模式下，ADLTS2 寄存器有效。在其它模式下无效。

（14） 运算放大器 n 控制寄存器 0（OPnCTL0）

OPnCTL0 寄存器用于控制运算放大器的操作，可以放大输入电平，并指定增益。

该寄存器可以按字节进行读取或写入。

系统复位后，该寄存器设为 00H。

（1/2）

复位后：00H R/W 地址：FFFFF260H

	7	6	5	4	3	2	1	0
OP0CTL0	0	0	0	OP0EN	OP0GA3	OP0GA2	OP0GA1	OP0GA0

复位后：00H R/W 地址：FFFFF2E0H

	7	6	5	4	3	2	1	0
OP1CTL0	0	OP12EN	OP11EN	OP10EN	OP1GA3	OP1GA2	OP1GA1	OP1GA0

OP12EN	A/D 转换器 1的运算放大器2的操作控制
0	操作禁止 (未使用)
1	操作使能 (已使用)

OP11EN	A/D 转换器 1的运算放大器1的操作控制
0	操作禁止 (未使用)
1	操作使能 (已使用)

OP0EN	A/D 转换器 0的运算放大器0的操作控制
0	操作禁止 (未使用)
1	操作使能 (已使用)

OP10EN	A/D 转换器 1的运算放大器0的操作控制
0	操作禁止 (未使用)
1	操作使能 (已使用)

OPnGA3	OPnGA2	OPnGA1	OPnGA0	运算放大器的增益规范
0	0	0	0	×2.500
0	0	0	1	×2.667
0	0	1	0	×2.857
0	0	1	1	×3.077
0	1	0	0	×3.333
0	1	0	1	×3.636
0	1	1	0	×4.000
0	1	1	1	×4.444
1	0	0	0	×5.000
1	0	0	1	×5.714
1	0	1	0	×6.667
1	0	1	1	×8.000
1	1	0	0	×10.00
其它				禁止设置

（2/2）

注意事项	**1.**	请确保将 **OP0CTL0** 寄存器的第 **5** 位至第 **7** 位和 **OP1CTL0** 寄存器的第 **7** 位清除为 **"0"**。
	2.	在使能运算放大器的操作之后，需要 **10 μs** 的稳定时间。

（15） 比较器 n 控制寄存器 0（CMPnCTL0）

CMPnCTL0 寄存器用于控制过电压检测比较器的操作。

该寄存器可以按字节进行读取或写入。

系统复位后，该寄存器设为 00H。

（1/2）

复位后：00H R/W 地址：FFFFF261H

	7	6	5	4	3	2	1	0
CMP0CTL0	0	0	0	CMP0FEN	0	0	0	CMP0LEN

复位后：00H R/W 地址：FFFFF2E1H

	7	6	5	4	3	2	1	0
CMP1CTL0	0	CMP12FEN	CMP11FEN	CMP10FEN	0	CMP12LEN	CMP11LEN	CMP10LEN

CMP12FEN	A/D 转换器 1的比较器2(全范围)的操作控制
0	操作禁止 (未使用)
1	操作使能 (已使用)

CMP11FEN	A/D 转换器 1的比较器1(全范围)的操作控制
0	操作禁止 (未使用)
1	操作使能 (已使用)

CMP0FEN	A/D 转换器 0的比较器0(全范围)的操作控制
0	操作禁止 (未使用)
1	操作使能 (已使用)

CMP10FEN	A/D 转换器 1的比较器0(全范围)的操作控制
0	操作禁止 (未使用)
1	操作使能 (已使用)

CMP12LEN	A/D 转换器 1的比较器2(低范围)的操作控制
0	操作禁止 (未使用)
1	操作使能 (已使用)

CMP11LEN	A/D 转换器 1的比较器1(低范围)的操作控制
0	操作禁止 (未使用)
1	操作使能 (已使用)

CMP0LEN	A/D 转换器 0的比较器0(低范围)的操作控制
0	操作禁止 (未使用)
1	操作使能 (已使用)

CMP10LEN	A/D 转换器 1的比较器0(低范围)的操作控制
0	操作禁止 (未使用)
1	操作使能 (已使用)

（2/2）

注意事项 **1.** 请确保将 **CMP0CTL0** 寄存器的第 **1** 位至第 **3** 位、第 **5** 位至第 **7** 位和 **CMP1CTL0** 寄存器的第 **3** 位、第 **7** 位清除为**“0”**。

2. 在使能比较器的操作之后，需要 **TBD** μ**s** 的稳定时间。

3. 比较器的输入电压范围如下，不论是否已经通过运算放大器被放大。

CREF0L 和 **CREF1L** 引脚（小范围端）的输入电压范围：**0.02**AV_{DD} **+ 0.1** 至 **0.5**AV_{DD} **– 0.1 V**

CREF0F 和 **CREF1F** 引脚（全范围端）的输入电压范围：**0.02**AV_{DD} **+ 0.1** 至 **0.92**AV_{DD} **– 0.1V**

详情参见第二十八章 电气特性。

（16） 比较器 n 控制寄存器 1（CMPnCTL1）

CMPnCTL1 寄存器用于监视过电压检测比较器的输出。

该寄存器只能按字节进行读取。

系统复位后，该寄存器设为 00H。

（1/2）

复位后：00H R 地址： FFFFF262H

	7	6	5	4	3	2	1	0
CMP0CTL1	0	0	0	CMP0FOUT	0	0	0	CMP0LOUT

复位后：00H R 地址：FFFFF2E2H

	7	6	5	4	3	2	1	0
CMP1CTL1	0	CMP12FOUT	CMP11FOUT	CMP10FOUT	0	CMP12LOUT	CMP11LOUT	CMP10LOUT

CMP12FOUT	A/D 转换器 1的比较器2(全范围)的输出电平状态
0	比较器输出 = 0 (无过电压检测)
1	比较器输出 = 1 (过电压检测)

CMP11FOUT	A/D 转换器 1的比较器1(全范围)的输出电平状态
0	比较器输出 = 0 (无过电压检测)
1	比较器输出 = 1 (过电压检测)

CMP0FOUT	A/D 转换器 0的比较器0(全范围)的输出电平状态
0	比较器输出 = 0 (无过电压检测)
1	比较器输出 = 1 (过电压检测)

CMP10FOUT	A/D 转换器 1的比较器0(全范围)的输出电平状态
0	比较器输出 = 0 (无过电压检测)
1	比较器输出 = 1 (过电压检测)

CMP12LOUT	A/D 转换器 1的比较器2(低范围)的输出电平状态
0	比较器输出 = 0 (无过电压检测)
1	比较器输出 = 1 (过电压检测)

CMP11LOUT	A/D 转换器 1的比较器1(低范围)的输出电平状态
0	比较器输出 = 0 (无过电压检测)
1	比较器输出 = 1 (过电压检测)

CMP0LOUT	A/D 转换器 0的比较器0(低范围)的输出电平状态
0	比较器输出 = 0 (无过电压检测)
1	比较器输出 = 1 (过电压检测)

CMP10LOUT	A/D 转换器 1的比较器0(低范围)的输出电平状态
0	比较器输出 = 0 (无过电压检测)
1	比较器输出 = 1 (过电压检测)

（2/2）

注意事项	**1.**	当输入电压下降到不能检测的过电压的电平时，**CMP12FOUT**、**CMP11FOUT**、**CMP0FOUT**、**CMP10FOUT**、**CMP12LOUT**、**CMP11LOUT**、**CMP0LOUT** 和 **CMP10LOUT** 位设置为 **0**。
	2.	请确保将 **CMP0CTL1** 寄存器的第 **1** 位至第 **3** 位、第 **5** 位至第 **7** 位和 **CMP1CTL1** 寄存器的第 **3** 位、第 **7** 位清除为 **“0”**。

（17） 比较器 n 控制寄存器 2（CMPnCTL2）

CMPnCTL2 寄存器用于指定过电压检测比较器的比较信号。
该寄存器可以按字节进行读取或写入。
系统复位后，该寄存器被为 00H。

复位后：00H R/W 地址：FFFFF263H

	7	6	5	4	3	2	1	0
CMP0CTL2	0	0	0	0	0	0	0	CMP0SEL

复位后：00H R/W 地址：FFFFF2E3H

	7	6	5	4	3	2	1	0
CMP1CTL2	0	0	0	0	0	CMP12SEL	CMP11SEL	CMP10SEL

CMP12SEL	A/D 转换器 1的比较器2的比较信号规范
0	运算放大器2放大之前
1	运算放大器2放大之后

CMP11SEL	A/D 转换器 1的比较器1的比较信号规范
0	运算放大器1放大之前
1	运算放大器1放大之后

CMP0SEL	A/D 转换器 0的比较器0的比较信号规范
0	运算放大器0放大之前
1	运算放大器0放大之后

CMP10SEL	A/D 转换器 1的比较器0的比较信号规范
0	运算放大器0放大之前
1	运算放大器0放大之后

注意事项 **请确保将 CMP0CTL2 寄存器的第 1 位至第 7 位和 CMP1CTL2 寄存器的第 3 位至第 7 位清除为“0”。**

（18）　比较器 n 控制寄存器 3（CMPnCTL3）

CMPnCTL3 寄存器用于指定过电压检测比较器的检测方向，并选择边沿检测。

该寄存器可以按字节进行读取或写入。

系统复位后，该寄存器设为 00H。

（1/2）

复位后：00H　R/W　地址：FFFFF264H

	7	6	5	4	3	2	1	0
CMP0CTL3	CMP0FDS	0	0	CMP0FDE	CMP0LDS	0	0	CMP0LDE

复位后：00H　R/W　地址：FFFFF2E4H

	7	6	5	4	3	2	1	0
CMP1CTL3	CMP1FDS	CMP12FDE	CMP11FDE	CMP10FDE	CMP1LDS	CMP12LDE	CMP11LDE	CMP10LDE

CMPnFDS	A/D 转换器n的比较器(全范围)的检测方向规范 (n=0,1)
0	检测逻辑积(AND) (如果输入电压低于参考值，检测到)
1	检测逻辑和 (OR) (如果输入电压大于参考值，检测到)

CMP12FDE	A/D 转换器1的比较器2(全范围)的边沿检测选择
0	禁止边沿检测 (未使用比较器)
1	使能边沿检测 (已使用比较器)

CMP11FDE	A/D 转换器1的比较器1(全范围)的边沿检测选择
0	禁止边沿检测 (未使用比较器)
1	使能边沿检测 (已使用比较器)

CMP0FDE	A/D 转换器0的比较器0(全范围)的边沿检测选择
0	禁止边沿检测 (未使用比较器)
1	使能边沿检测 (已使用比较器)

CMP10FDE	A/D 转换器 1的比较器0(全范围)的边沿检测选择
0	禁止边沿检测 (未使用比较器)
1	使能边沿检测 (已使用比较器)

CMPnLDS	A/D 转换器n的比较器(全范围)的检测方向规范 (n=0,1)
0	检测逻辑积(AND) (如果输入电压低于参考值，检测到)
1	检测逻辑和 (OR) (如果输入电压大于参考值，检测到)

备注　参考值就是输入到 CREFnL 和 CREFnF 引脚（n = 0，1）的参考电压。

（2/2）

CMP12LDE	A/D 转换器1的比较器2(低范围)的边沿检测选择
0	禁止边沿检测 (未使用比较器)
1	使能边沿检测 (已使用比较器)

CMP11LDE	A/D 转换器1的比较器1(低范围)的边沿检测选择
0	禁止边沿检测 (未使用比较器)
1	使能边沿检测 (已使用比较器)

CMP0LDE	A/D 转换器0的比较器0(低范围)的边沿检测选择
0	禁止边沿检测 (未使用比较器)
1	使能边沿检测 (已使用比较器)

CMP10LDE	A/D 转换器1的比较器0(低范围)的边沿检测选择
0	禁止边沿检测 (未使用比较器)
1	使能边沿检测 (已使用比较器)

（19） A/D 转换器 n 时钟选择寄存器（ADnOCKS）

ADnOCKS 寄存器用于选择输入到 A/D 转换器 n 的时钟（f_{AD01}）。

该寄存器可以按字节进行读取或写入。

系统复位后，该寄存器设为 00H。

复位后：00H　R/W　地址：AD0OCKS FFFFF270H, AD1OCKS FFFFF274H

	7	6	5	4	3	2	1	0
ADnOCKS (n = 0, 1)	0	0	0	ADnOCKSEN	0	0	ADnOCKS1	ADnOCKS0

ADnOCKSEN	时钟操作控制
0	停止A/D转换器 n的操作时钟供给
1	使能A/D转换器 n的操作时钟供给

ADnOCKS1	ADnOCKS0	A/D 转换器n的输入时钟选择 (f_{AD01})
0	0	$f_{XX}/2$
0	1	$f_{XX}/3$
1	0	$f_{XX}/4$ (当 f_{XX} = 64 MHz)
1	1	禁止设置

注意事项 **1.** 设置 **f_{AD01} 为 16 MHz** 或更低。

2. 当使用 **A/D** 转换器 **n** 时，请确保设置 **ADnOCKS** 寄存器，并将 **ADnSCM.ADnPS** 位设置为 **1**，读取 **A/D** 转换结果寄存器也要一样设置。

3. 请确保将第 **2** 位、第 **3** 位和第 **5** 位至第 **7** 位清除为**“0”**。

（20） 比较器输出数字噪声消除寄存器 nL，nF（CMPNFCnL，CMPNFCnF）

CMPNFCnL 和 CMPNFCnF 寄存器用于控制过电压检测比较器输出的数字噪声消除。

该寄存器可以按字节进行读取或写入。

系统复位后，该寄存器设为 00H。

复位后：00H R/W 地址：CMPNFC0L FFFFF278H, CMPNFC1L FFFFF27CH

	7	6	5	4	3	2	1	0
CMPNFCnL (n = 0, 1)	CMPnNFEN	0	0	0	0	CMPnNFC2	CMPnNFC1	CMPnNFC0

复位后：00H R/W 地址：CMPNFC0F FFFFF27AH, CMPNFC1F FFFFF27EH

	7	6	5	4	3	2	1	0
CMPNFCnF (n = 0, 1)	CMPnNFEN	0	0	0	0	CMPnNFC2	CMPnNFC1	CMPnNFC0

<R>

CMPnNFEN	数字噪声消除的设置
0	注
1	执行数字噪声消除

CMPnNFC2	CMPnNFC1	CMPnNFC0	采样时钟选择
0	0	0	$f_{XX}/32$
0	0	1	$f_{XX}/64$
0	1	0	$f_{XX}/128$
0	1	1	$f_{XX}/256$
1	0	0	$f_{XX}/1024$
1	0	1	$f_{XX}/4096$
其它情况			禁止设置

<R> 注 使用比较器输出时，请确保将 CMPnNFEN 位设置为 1（数字噪声已消除）。同时，必须选择采样时钟。未使用比较器输出时，将 CMPnNFEN 位清除为 0。

注意事项 1. 数字噪声消除的基本操作和图 4-4 噪声消除时序的示例 相同。区别在于采样执行两次，且采样时钟不同。

2. 请确保将第 3 位至第 6 位清除为“0”。

（21） A/D 触发上升沿，下降沿指定寄存器（ADTR，ADTF）

ADTR 和 ADTF 寄存器用于指定 ADTRG0 和 ADTRG1 引脚的触发模式，并为各个引脚单独指定有效边沿（上升沿，下降沿，或双沿）。

该寄存器可以按字节或按位进行读取或写入。

系统复位后，该寄存器设为 00H。

注意事项 **从 A/D 转换器 n（复用功能）的外部触发输入功能切换为端口模式时，可能会检测到边沿。因此，请确保将 ADTFn 和 ADTRn 位设置为 00，然后设置端口模式。**

复位后：00H R/W 地址：FFFFF2F2H

	7	6	5	4	3	2	1	0
ADTR	0	0	0	0	0	0	ADTR1	ADTR0

复位后：00H R/W 地址：FFFFF2F0H

	7	6	5	4	3	2	1	0
ADTF	0	0	0	0	0	0	ADTF1	ADTF0

备注 关于有效边沿的指定说明，参见表 **12-8**。

表 12-8. ADTRG0 和 ADTRG1 引脚的有效边沿指定

ADTFn	ADTRn	有效边沿的指定说明
0	0	无边沿检测
0	1	上升沿
1	0	下降沿
1	1	双沿

注意事项 **这些引脚不用作 ADTRGn 引脚时，请确保将 ADTFn 和 ADTRn 位设置为 00。**

备注 n = 0，1

（22） 比较器输出中断上升沿，下降沿指定寄存器（CMPOR，CMPOF）

CMPOR 和 CMPOF 寄存器用于指定 INTCMP0L，INTCMP0F，INTCMP1L 和 INTCMP1F 信号的触发模式，并可以为各个中断请求信号（上升沿，下降沿，或双沿）单独指定有效边沿。

该寄存器可以按字节或按位进行读取或写入。

系统复位后，该寄存器设为 00H。

复位后：00H R/W 地址：FFFFF2F6H

	7	6	5	4	3	2	1	0
CMPOR	0	0	0	0	CMPOR1F	CMPOR1L	CMPOR0F	CMPOR0L

复位后：00H R/W 地址：FFFFF2F4H

	7	6	5	4	3	2	1	0
CMPOF	0	0	0	0	CMPOF1F	CMPOF1L	CMPOF0F	CMPOF0L

备注 关于有效边沿的指定说明，参见表 **12-9** 和 **12-10**。

表 12-9. INTCMP0F 和 INTCMP1F 信号的有效边沿指定

CMPOFnF	CMPORnF	有效边沿的指定说明
0	0	无边沿检测
0	1	上升沿
1	0	下降沿
1	1	双沿

备注 n = 0，1

表 12-10. INTCMP0L 和 INTCMP1L 信号的有效边沿指定

CMPOFnL	CMPORnL	有效边沿的指定说明
0	0	无边沿检测
0	1	上升沿
1	0	下降沿
1	1	双沿

备注 n = 0，1

12.4 操作

注意事项 1. A/D 转换器 0 和 1 支持两个电路同时采样。

2. 操作设置的详情，参见 12.3（1）A/D 转换器 n 扫描模式寄存器（ADnSCM）.

12.4.1 基本操作

按照下列过程执行 A/D 转换。

（1）通过 ADnOCKS 寄存器选择一个输入时钟（f_{AD01}），并将 ADnOCKSEN 位设置为 1（使能对 A/D 转换器 n 操作时钟的供给）。

（2）设置 ADnSCM.ADnPS 位 = 1.

（3）在 <2>之后等待 1 μs 或更久。

（4）通过ADnSCM[注]，ADnCTC，ADnCHEN，ADnCTL0，ADnTSEL，ADnCH1，ADnCH2，ADLTS1 和 ADLTS2 寄存器（n = 0，1）选择一个模拟输入引脚和操作模式。通过ADnCTC.ADnFR3 至ADnCTC.ADnFR0 位的设置来决定A/D 转换时钟的数量和A/D 转换时间。

注　请确保将 ADnSCM 寄存器的第 1 位设置为“1”。
该设置可以和 ADnSCM 寄存器的其它位同时执行。

（5）在 A/D 触发模式和 A/D 触发轮询模式下，将 ADnSCM.ADnCE 位设置为 1 可以启动 A/D 转换（n = 0，1）。如果在硬件触发模式，转换通道指定模式和扩展缓冲模式下，将 ADnCE 位设置为 1，A/D 转换器进入触发等待状态。

（6）当 A/D 转换启动之后，通过采样&保持电路对输入到选中模拟输入通道的电压进行采样。当输入电平放大使用运算放大器时，由 OPnCTL0.OPnGA3 至 OPnCTL0.OPnGA0 位指定的增益 × 被采样的输入电压。

（7）当采样已经执行了特定次数后，采样&保持电路进入保持状态，并保持输入模拟电压直到 A/D 转换结束。

（8）设置逐次逼近寄存器（SAR）的第 11 位。档位选择器将阵列的电压档位电平改变为参考电压（1/2AV_{REFPn}）。

（9）通过比较器，对阵列的电压档位产生的电压和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（1/2AV_{REFPn}），逐次逼近寄存器（SAR）的最高有效位（MSB）保持置位。如果模拟输入电压小于参考电压（1/2AV_{REFPn}），SAR 的 MSB 重置。

（10）接下来，逐次逼近寄存器（SAR）的第 10 位自动置位，进行下一次比较。阵列的电压档位器由第 11 位的值选择，上次的比较结果已经被设置到第 11 位内。

第 11 位 = 0：（1/4 AV_{REFPn}）
第 11 位 = 1：（3/4 AV_{REFPn}）

阵列的电压档位器和模拟输入电压进行比较，根据比较结果，操作 SAR 的第 10 位。

模拟输入电压 ≥ 阵列的电压档位器：第 10 位 = 1
模拟输入电压 ≤ 阵列的电压档位器：第 10 位 = 0

如此继续比较，直到 SAR 的第 0 位。

（11）当 12 位的比较完成后，有效数字值结果保留在逐次逼近寄存器（SAR）内。在 A/D 触发模式、A/D 触发轮询模式、硬件触发模式和转换通道指定模式下，该值传送到 A/Dn 转换结果寄存器 m（ADnCRm）且转换结果存储在该寄存器（n = 0, 1，m = 0 至 15）。在扩展缓冲模式下，有效数字值存储在 A/Dn 转换结果扩展缓冲寄存器 a，且当产生选择装载触发 x 时，转移到 A/Dn 转换结果扩展寄存器 a 并存储（x = 1，2，a = 0 至 4）。当 A/D 转换已经结束了指定的次数，会产生一个 A/Dn 转换结束中断请求信号（INTADn）。

12.4.2 输入电压和转换结果

输入到模拟输入引脚（ANInk）的模拟电压和 A/D 转换结果（A/Dn 转换结果寄存器 m（ADnCRm）或 A/Dn 转换结果扩展寄存器 a（ADnECRa）的转换结果）之间的关系如下：

$$\mathrm{ADCR} = \mathrm{INT}\left(\frac{V_{IN}}{AV_{REFP}} \times 4{,}096 + 0.5\right)$$

或，

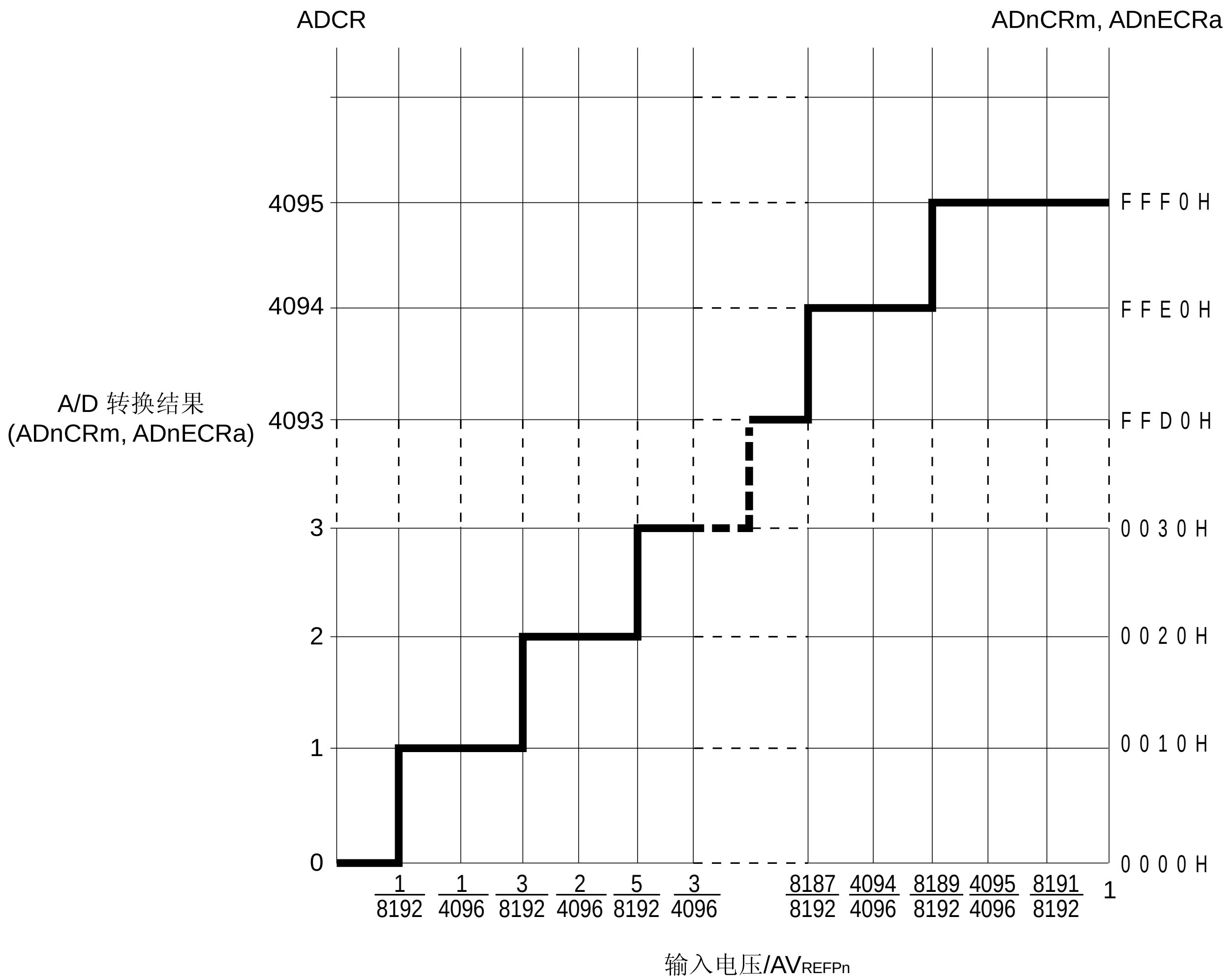
$$(\mathrm{ADCR} - 0.5) \times \frac{AV_{REFP}}{4{,}096} \leq V_{IN} < (\mathrm{ADCR} + 0.5) \times \frac{AV_{REFP}}{4{,}096}$$

INT（ ）： 返回（ ）中的数值的整数的功能
V_{IN}： 模拟输入电压
AV_{REFP}： AV_{REFPn} 引脚电压
ADCR： A/Dn 转换结果寄存器 m（ADnCRm）或 A/Dn 转换结果扩展寄存器 a（ADnECRa）的值

模拟输入电压和 A/D 转换结果之间的关系如下。

备注 A/D 转换器 0：n = 0，m = 0 至 15，k = 0 至 5，a = 0 至 4
A/D 转换器 1：n = 1，m = 0 至 15，k = 0 至 7，a = 0 至 4

图 12-10. 模拟输入电压和 A/D 转换结果之间的关系



备注　A/D 转换器 0：n = 0，m = 0 至 15，k = 0 至 5，a = 0 至 4

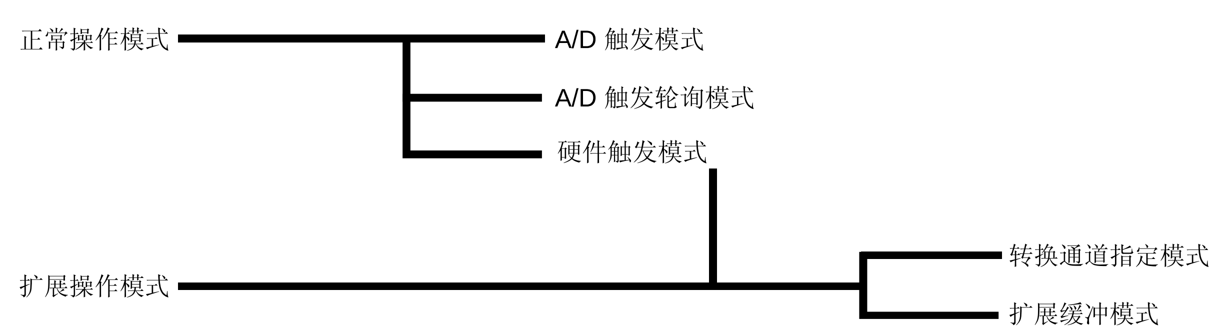
A/D 转换器 1：n = 1，m = 0 至 15，k = 0 至 7，a = 0 至 4

12.4.3 操作模式

通过指定操作模式，可以为 A/D 转换器 0 和 1 指定各种转换操作。操作模式由 ADnSCM，ADnCTC，ADnCHEN，ADnCTL0，ADnTSEL，ADnCH1，ADnCH2，ADLTS1，ADLTS2 和 ADnOCKS 寄存器设置。

下面是操作模式之间的关系。

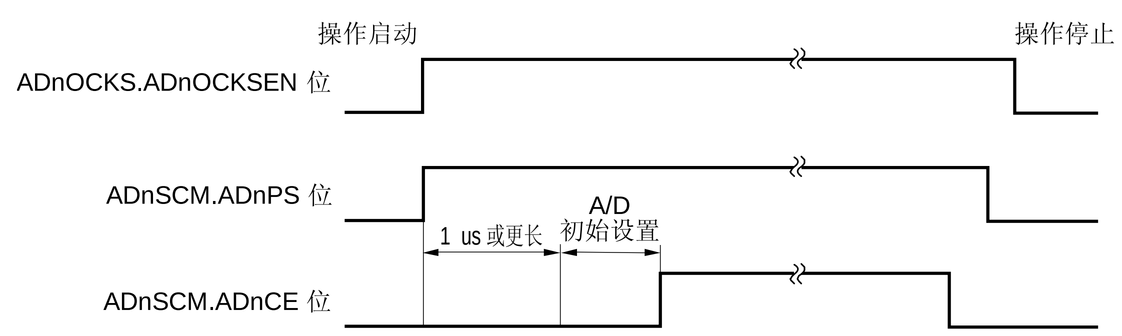
备注　　n = 0，1



注意事项　当使用转换通道指定模式或扩展缓冲模式时，请确保设置硬件触发模式。

12.4.4 操作设置

A/D 转换器 0 和 1 的启动或停止操作按照下列过程。



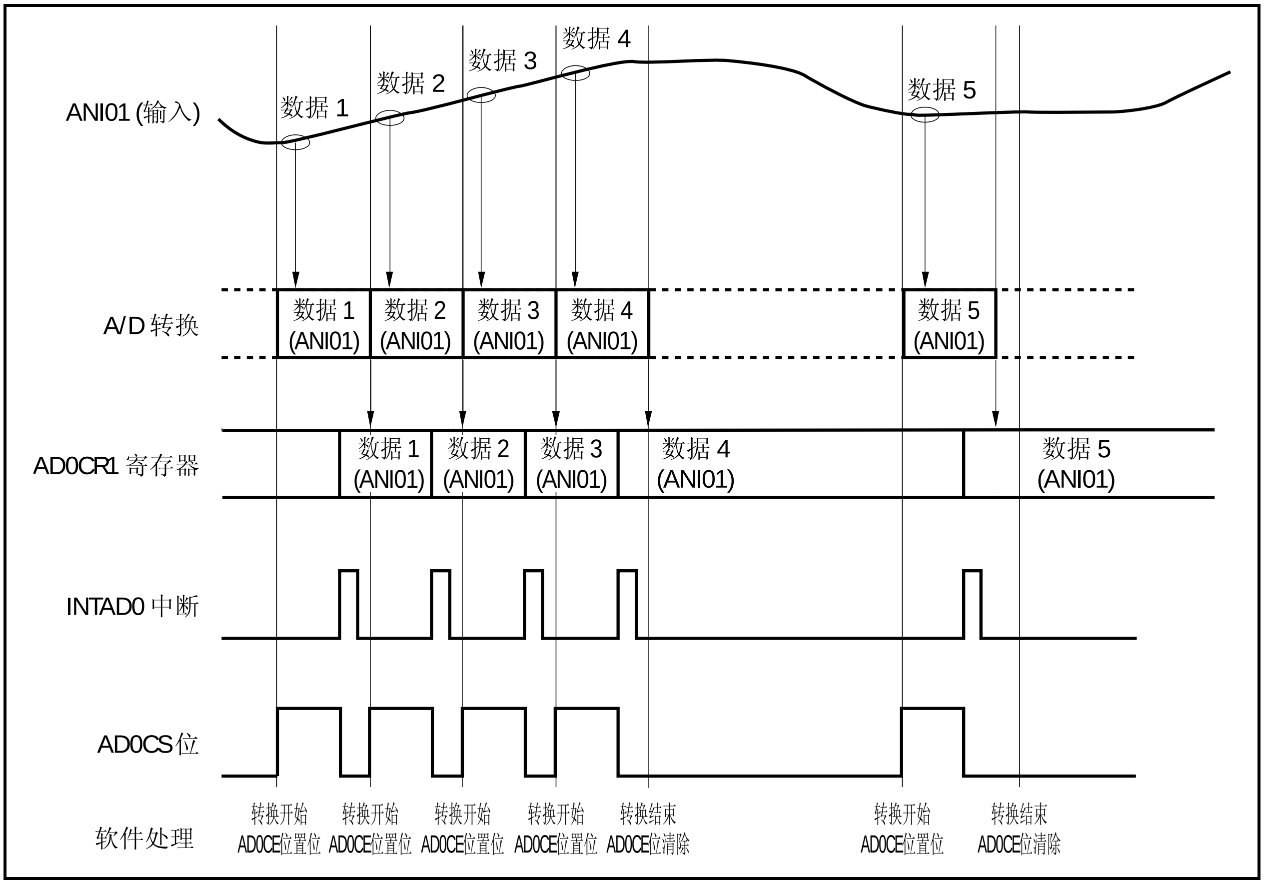
12.4.5 1-通道转换的操作

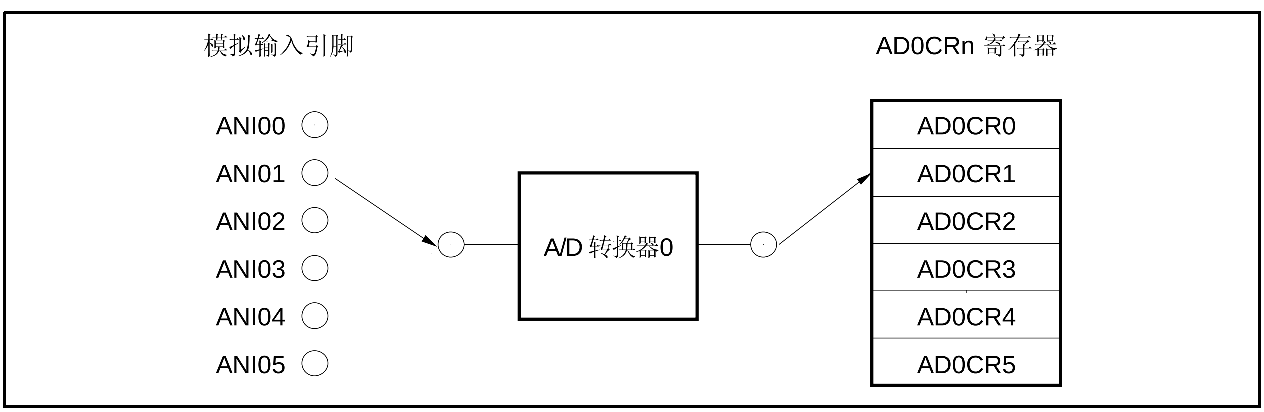
由 ADnCHEN 寄存器指定转换一个模拟输入引脚（ANInk）的信号。转换结果存储在 ANInk 引脚对应的 ADnCRk 寄存器。按照一对一的基础，ANInk 引脚和 ADnCRk 寄存器逐一对应，每次转换完成时，产生一个 A/Dn 转换结束中断请求信号（INTADn）。

在 A/D 转换完成之后，在 A/D 触发模式或 A/D 触发轮询模式下，转换操作停止。在硬件触发模式下，A/D 转换器等待触发。

备注　A/D 转换器 0：n = 0，k = 0 至 5
　　　A/D 转换器 1：n = 1，k = 0 至 7

图 12-11. 1-通道转换（A/D 触发模式）的操作：A/D 转换器 0





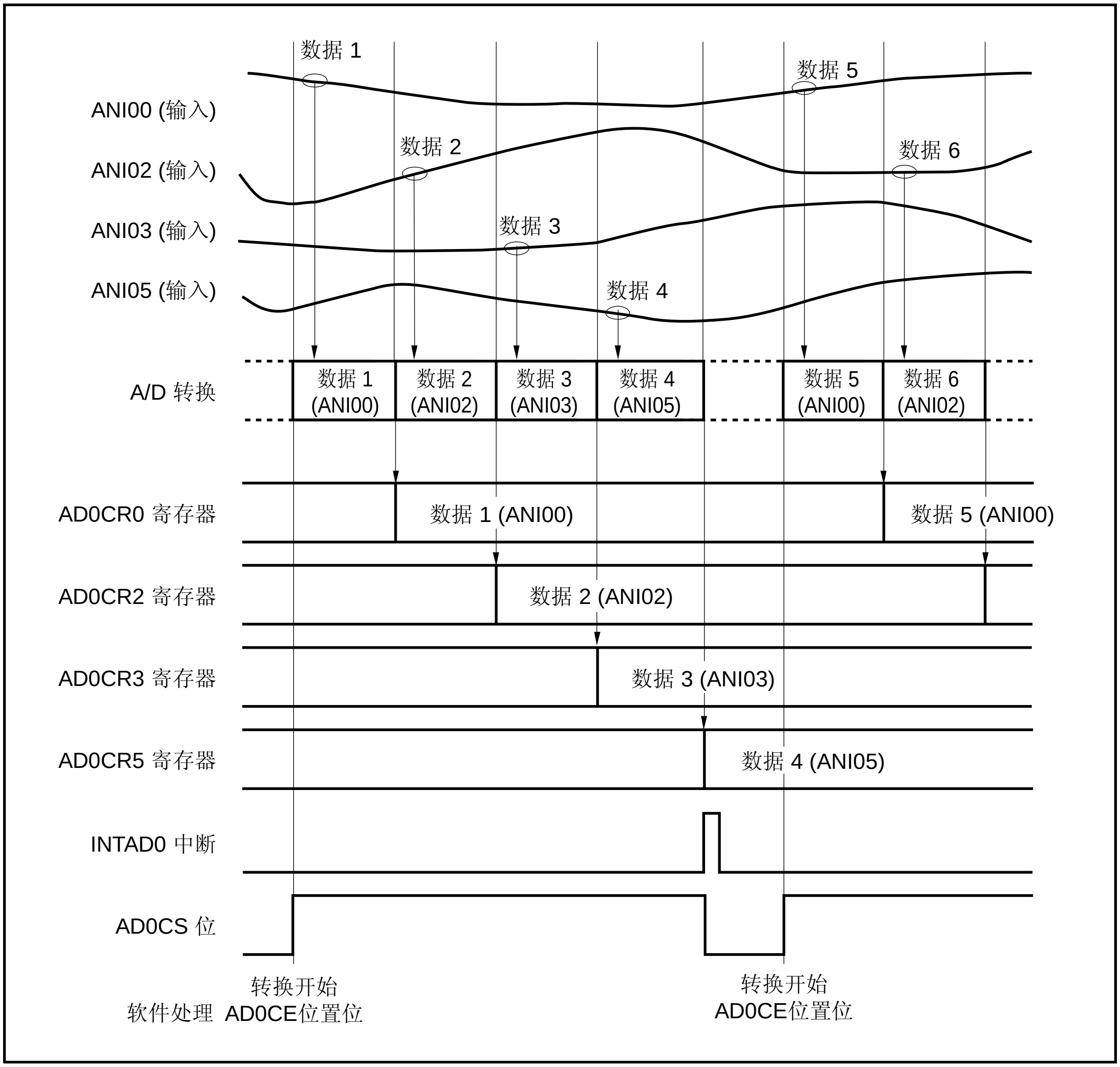
12.4.6 多个通道转换的操作

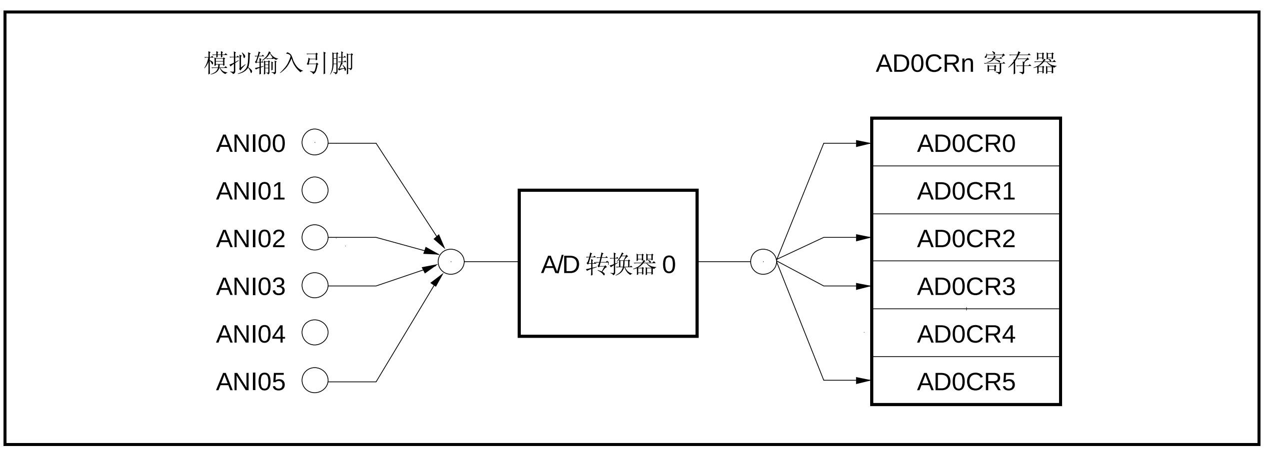
由 ADnCHEN 寄存器指定转换两个或多个模拟输入引脚（ANInk）的信号。从最小数字的引脚开始（在图 12-12 的示例中，ANI00 → ANI02 → ANI03 → ANI05）依次转换信号。如果没有指定某个模拟输入引脚，则跳过。转换结果存储在 ANInk 引脚对应的 ADnCRk 寄存器。按照一对一的基础，ANInk 引脚和 ADnCRk 寄存器逐一对应，当指定的模拟输入引脚的信号转换完成时，产生一个 A/Dn 转换结束中断请求信号（INTADn）。

在 A/D 转换完成之后，在 A/D 触发模式或 A/D 触发轮询模式下，转换操作停止。在硬件触发模式下，A/D 转换器等待触发。

备注　A/D 转换器 0：n = 0，k = 0 至 5
　　　　A/D 转换器 1：n = 1，k = 0 至 7

图 12-12. 多个通道转换（A/D 触发模式）的操作：A/D 转换器 0





12.4.7 A/D 触发模式（正常 操作模式）

当 ADnSCM.ADnCE 位设置为 1 时，A/D 转换启动。

当转换启动后，ADnSCM.ADnCS 位设置为 1（转换正在进行中）。

如果在 A/D 转换期间写入 ADnSCM 寄存器，转换停止，并再次重头开始。

（1）1-通道转换的操作

转换一次模拟输入引脚（ANInk）的信号，结果存储在 ADnCRk 寄存器内。按照一对一的基础，ANInk 引脚和 ADnCRk 寄存器逐一对应。

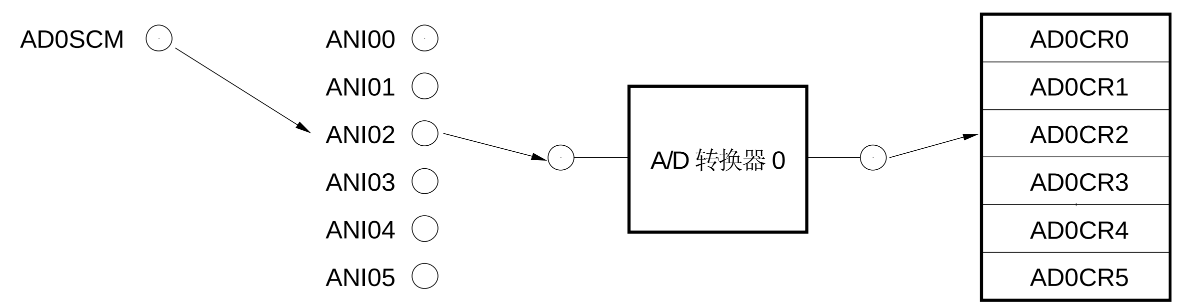
每次转换完成后，产生一个 A/Dn 转换结束中断请求信号（INTADn）。在 A/D 转换完成之后，A/D 转换器停止转换操作,且 ADnSCM.ADnCE 位保持设置为 1。通过将 ADnCE 位设置为 1，A/D 转换可以重新启动。

对于每次转换完成一次都应该读取 A/D 转换结果的应用，该操作是适合的。

模拟输入引脚	A/D 转换结果寄存器
ANInk	ADnCRk

备注　A/D 转换器 0：n = 0，k = 0 至 5
　　　A/D 转换器 1：n = 1，k = 0 至 7

图 12-13. 1-通道转换操作（A/D 触发模式）的示例：A/D 转换器 0



（1）　AD0CE 位 = 1（使能）
（2）　ANI02 引脚的信号被 A/D 转换
（3）　转换结果被存储在 AD0CR2 寄存器
（4）　AD0SCM.AD0CS 位 = 0
（5）　产生 INTAD0 中断请求信号

备注　这是个操作示例，其中 AD0SCM.AD0PLM，AD0TRG1 和 AD0TRG0 位 = 000，AD0CTL0.AD0MD1 和 AD0CTL0.AD0MD0 位 = 00，且 AD0CHEN 寄存器 = 0004H。

（2）多个通道转换的操作

由 ADnCHEN 寄存器指定的两个或多个模拟输入引脚的信号，从最小数字的引脚开始依次转换。转换结果存储在模拟输入引脚对应的 ADnCRk 寄存器。

当指定的模拟输入引脚的信号转换完成时，产生一个 A/Dn 转换结束中断请求信号（INTADn）。在 A/D 转换完成之后，A/D 转换器停止转换操作且 ADnSCM.ADnCE 位保持设置为 1。通过将 ADnCE 位设置为 1，A/D 转换可以重新启动。

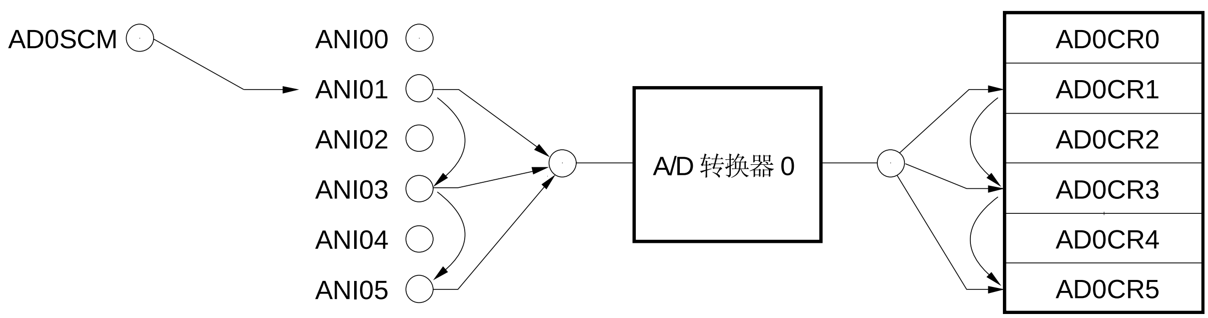
对于应该监视两个或多个模拟输入信号的应用，该操作是适合的。

模拟输入引脚	A/D 转换结果寄存器
ANInk[注]	ADnCRk
|	|
ANInk[注]	ADnCRk

注　可以通过 ADnCHEN 寄存器指定两个或多个。
但是，A/D 转换从最小数字的引脚开始依次执行。

备注　A/D 转换器 0：n = 0，k = 0 至 5
A/D 转换器 1：n = 1，k = 0 至 7

图 12-14. 多个通道转换操作（A/D 触发模式）的示例：A/D 转换器 0



（1）　AD0CE 位 = 1（使能）
（2）　A/D 转换 ANI01 引脚的信号
（3）　转换结果存储在 AD0CR1 寄存器
（4）　A/D 转换 ANI03 引脚的信号
（5）　转换结果存储在 AD0CR3 寄存器
（6）　A/D 转换 ANI05 引脚的信号
（7）　转换结果存储在 AD0CR5 寄存器
（8）　AD0SCM.AD0CS 位 = 0
（9）　产生 INTAD0 中断请求信号

备注　这是个操作示例，其中 AD0SCM.AD0PLM，AD0TRG1 和 AD0TRG0 位 = 000，AD0CTL0.AD0MD1 和 AD0CTL0.AD0MD0 位 = 00，且 AD0CHEN 寄存器 = 002AH。

12.4.8 A/D 触发轮询模式（正常 操作模式）

当 ADnSCM.ADnCE 位设置为 1 时，A/D 转换启动。

当转换启动后，ADnSCM.ADnCS 位设置为 1（转换正在进行中）。

如果在 A/D 转换期间写入 ADnSCM 寄存器，转换停止，并再次重头开始。

在 A/D 触发轮询模式下，不需要在 A/Dn 转换结束中断请求信号（INTADn）产生之后，将 1 写入到 ADnCE 位来启动 A/D 转换。

如果在 A/D 转换期间写入 ADnSCM 寄存器，转换停止，并再次重头开始。

（1）1-通道转换的操作

转换一次一个模拟输入引脚（ANInk）的信号，结果存储在 ADnCRk 寄存器内。按照一对一的基础，ANInk 引脚和 ADnCRk 寄存器逐一对应。

每次转换完成后，产生一个 A/Dn 转换结束中断请求信号（INTADn）。A/D 转换持续重复，直到 ADnSCM.ADnCE 位设置为 0。当 ADnCE 位清除为 0 时，转换操作停止。

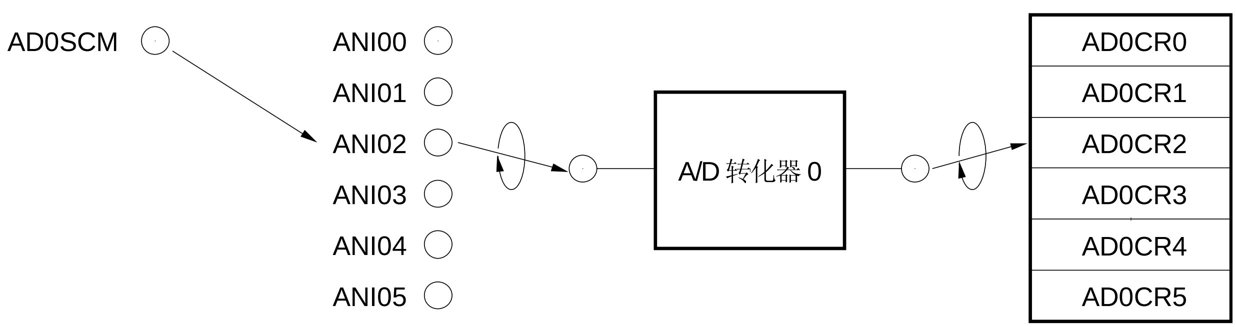
在 A/D 触发轮询模式下，不需要设置 ADnCE 位来重启转换操作。

该操作是适合经常读取 A/D 转换值的应用。

模拟输入引脚	A/D 转换结果寄存器
ANInk	ADnCRk

备注　A/D 转换器 0：n = 0，k = 0 至 5
A/D 转换器 1：n = 1，k = 0 至 7

图 12-15. 1-通道 转换操作（A/D 触发轮询模式）的示例：A/D 转换器 0



（1）　AD0CE 位 = 1（使能）
（2）　A/D 转换 ANI02 引脚的信号
（3）　转换结果存储在 AD0CR2 寄存器
（4）　AD0SCM.AD0CS 位 = 0
（5）　产生 INTAD0 中断请求信号
（6）　返回到（2）
（7）　设置 AD0CE 位为 0 来结束（停止）

备注　这是个操作示例，其中 AD0SCM.AD0PLM，AD0TRG1 和 AD0TRG0 位 = 100，AD0CTL0.AD0MD1 和 AD0CTL0.AD0MD0 位 = 00，且 AD0CHEN 寄存器 = 0004H。

（2）多个通道转换的操作

由 ADnCHEN 寄存器指定的两个或多个模拟输入引脚的信号，从最小数字的引脚开始依次转换。转换结果存储在模拟输入引脚对应的 ADnCRk 寄存器。

当指定的模拟输入引脚的信号转换完成时，产生一个 A/Dn 转换结束中断请求信号（INTADn）。A/D 转换持续重复，直到 ADnSCM.ADnCE 位设置为 0。当 ADnCE 位清除为 0 时，转换操作停止。

在 A/D 触发轮询模式下，不需要设置 ADnCE 位来重启转换操作。

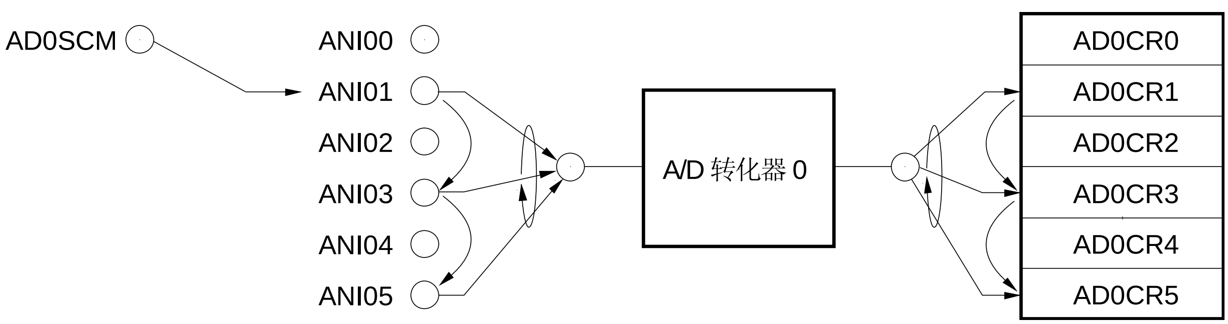
该操作是适合经常读取 A/D 转换值的应用。

模拟输入引脚	A/D 转换结果寄存器
ANInk[注]	ADnCRk
|	|
ANInk[注]	ADnCRk

注　可以通过 ADnCHEN 寄存器指定两个或多个。
但是，A/D 转换从最小数字的引脚开始依次执行。

备注　A/D 转换器 0：n = 0，x = 0 至 5
A/D 转换器 1：n = 1，x = 0 至 7

图 12-16. 多个通道转换操作（A/D 触发轮询模式）的示例：A/D 转换器 0



（1）　AD0CE 位 = 1（使能）
（2）　A/D 转换 ANI01 引脚的信号
（3）　转换结果存储在 AD0CR1 寄存器
（4）　A/D 转换 ANI03 引脚的信号
（5）　转换结果存储在 AD0CR3 寄存器
（6）　A/D 转换 ANI05 引脚的信号
（7）　转换结果存储在 AD0CR5 寄存器
（8）　AD0SCM.AD0CS 位 = 0
（9）　产生 INTAD0 中断请求信号
（10）　返回到（2）
（11）　将 AD0CE 位设置为 0 来结束（停止）

备注　这是个操作示例，其中 AD0SCM.AD0PLM，AD0TRG1 和 AD0TRG0 位 = 100，AD0CTL0.AD0MD1 和 AD0CTL0.AD0MD0 位 = 00，且 AD0CHEN 寄存器 = 002AH。

12.4.9 硬件触发模式（正常操作模式）

当 ADnSCM.ADnCE 位设置为 1 时，A/D 转换器等待触发，当由 ADnTSEL.ADnTRGSEL11 和 ADnTSEL.ADnTRGSEL10 位指定的触发产生时，A/D 转换启动。

当转换被启动后，ADnSCM.ADnCS 位被设置为 1（转换正在进行中）。

如果在 A/D 转换期间写入 ADnSCM 寄存器，转换停止，并再次重头开始。

（1） 1-通道转换的操作

转换一次一个模拟输入引脚（ANInk）的信号，由 ADnTSEL.ADnTRGSEL11 和 ADnTSEL.ADnTRGSEL10 位指定的信号用作触发器，转换结果存储在 ADnCRk 寄存器内。按照一对一的基础，ANInk 引脚和 ADnCRk 寄存器逐一对应。

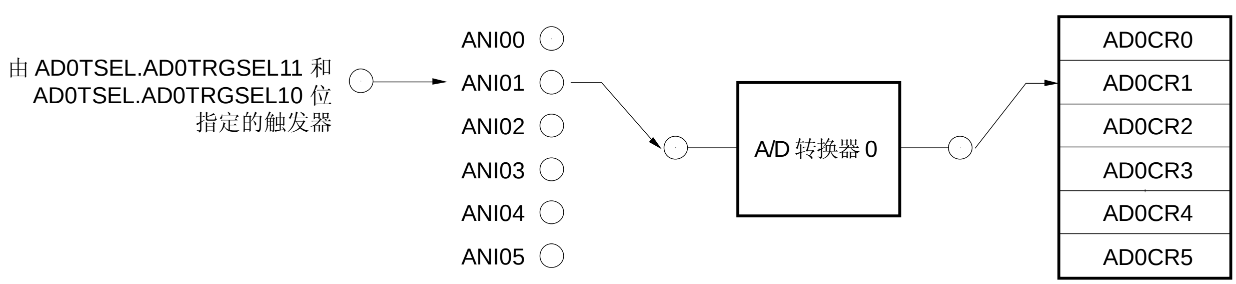
每次转换完成后，产生一个 A/Dn 转换结束中断请求信号（INTADn）。在 A/D 转换完成之后，转换器等待触发且 ADnSCM.ADnCE 位设置为 1。

对于每次由一个触发引起的转换完成后，都应该读取 A/D 转换结果的应用，该操作是适合的。

模拟输入引脚	A/D 转换结果寄存器
ANInk	ADnCRk

备注 A/D 转换器 0：n = 0，k = 0 至 5
A/D 转换器 1：n = 1，k = 0 至 7

图 12-17. 1-通道转换操作（硬件触发模式）的示例：A/D 转换器 0



（1） AD0CE 位 = 1（使能）
（2） 产生由 AD0TSEL.AD0TRGSEL11 位和 AD0TSEL.AD0TRGSEL10 位指定的触发
（3） A/D 转换 ANI01 引脚的信号
（4） 转换结果存储在 AD0CR1 寄存器
（5） AD0SCM.AD0CS 位 = 0
（6） 产生 INTAD0 中断请求信号
（7） 当输入下一个触发器时，返回到（3）
（8） 设置 AD0CE 位为 0 来结束（停止）

备注 这是个操作示例，其中 AD0SCM.AD0PLM，AD0TRG1 和 AD0TRG0 位 = 001，AD0CTL0.AD0MD1 和 AD0CTL0.AD0MD0 位 = 00，且 AD0CHEN 寄存器 = 0002H。

（2）多个通道转换的操作

由 ADnCHEN 寄存器指定的两个或多个模拟输入引脚的信号，从最小数字的引脚开始依次转换，由 ADnTSEL.ADnTRGSEL11 和 ADnTSEL.ADnTRGSEL10 位指定的信号用作触发器，转换结果存储在模拟输入引脚对应的 ADnCRk 寄存器。

当指定的模拟输入引脚的信号转换完成时，产生一个 A/Dn 转换结束中断请求信号（INTADn）。在 A/D 转换完成之后，转换器等待触发且 ADnSCM.ADnCE 位设置为 1。

在 A/D 触发轮询模式下，不需要设置 ADnCE 位来重启转换操作。

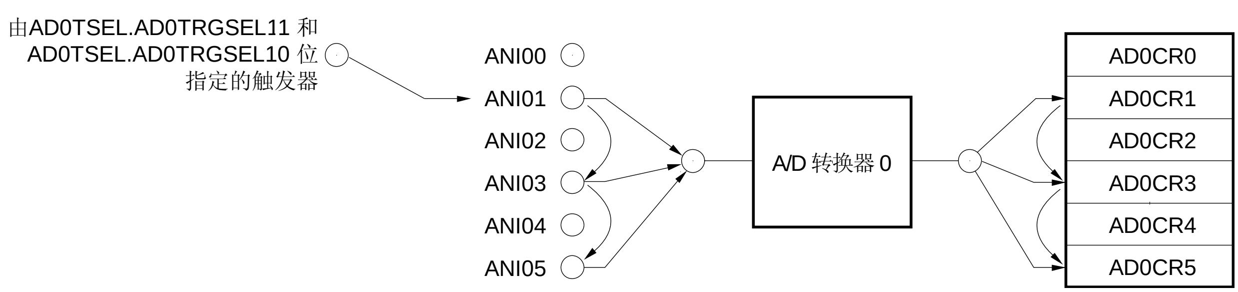
对于产生触发时需要监视两个或多个模拟输入信号的应用，该操作是适合的。

模拟输入引脚	A/D 转换结果寄存器
ANInk[注]	ADnCRk
|	|
ANInk[注]	ADnCRk

注　可以通过 ADnCHEN 寄存器指定两个或多个。
但是，A/D 转换从最小数字的引脚开始依次执行。

备注　A/D 转换器 0：n = 0，k = 0 至 5
A/D 转换器 1：n = 1，k = 0 至 7

图 12-18. 多个通道转换操作（硬件触发模式）的示例：A/D 转换器 0



（1）AD0CE 位 = 1（使能）
（2）产生由 AD0TSEL.AD0TRGSEL11 位和 AD0TSEL.AD0TRGSEL10 位指定的触发
（3）A/D 转换 ANI01 引脚的信号
（4）转换结果存储在 AD0CR1 寄存器
（5）A/D 转换 ANI03 引脚的信号
（6）转换结果存储在 AD0CR3 寄存器
（7）A/D 转换 ANI05 引脚的信号
（8）转换结果存储在 AD0CR5 寄存器
（9）AD0SCM.AD0CS 位 = 0
（10）产生 INTAD0 中断请求信号
（11）当输入下一个触发器时，返回到（3）
（12）设置 AD0CE 位为 0 来结束（停止）

备注　这是个操作示例，其中 AD0SCM.AD0PLM，AD0TRG1 和 AD0TRG0 位 = 001，AD0CTL0.AD0MD1 和 AD0CTL0.AD0MD0 位 = 00，且 AD0CHEN 寄存器 = 002AH。

12.4.10 转换通道指定模式（扩展操作模式）

当 ADnSCM.ADnCE 位设置为 1 时，A/D 转换器等待触发。由 ADnTSEL.ADnTRGSEL11 和 ADnTSEL.ADnTRGSEL10 位指定的选择触发 1 产生时，转换器启动 A/D 转换。

当转换启动后，ADnSCM.ADnCS 位设置为 1（转换正在进行中）。

如果在 A/D 转换期间写入 ADnSCM 寄存器，转换停止，转换器再次等待一个触发。

模拟输入引脚由 ADnCH1.ADnTRGCH12 至 ADnCH1.ADnTRGCH10 和 ADnCH1.ADnTRGCH16 至 ADnCH1.ADnTRGCH14 位指定。每次选择触发 1 产生时，由 ADnCH1.ADnTRGCH12 至 ADnCH1.ADnTRGCH10 和 ADnCH1.ADnTRGCH16 至 ADnCH1.ADnTRGCH14 位指定的模拟输入引脚依次选中。

某个指定模拟输入引脚的信号转换的次数，由 ADnCHEN 寄存器（最多 16 次）指定，使用选择触发 1 作为触发器结果存储在 ADnCHEN 寄存器指定的 ADnCRm 寄存器。转换结果从 ADnCR0 依次存储。

当指定模拟输入引脚的信号已经转换了 ADnCHEN 寄存器指定的次数（最多 16 次）时，产生一个 A/Dn 转换结束中断请求信号（INTADn）。在 A/D 转换完成之后，A/D 转换器等待触发且 ADnSCM.ADnCE 位保持设置为 1。

对于对需要监视两个或多个模拟输入信号的应用，该操作是适合的。

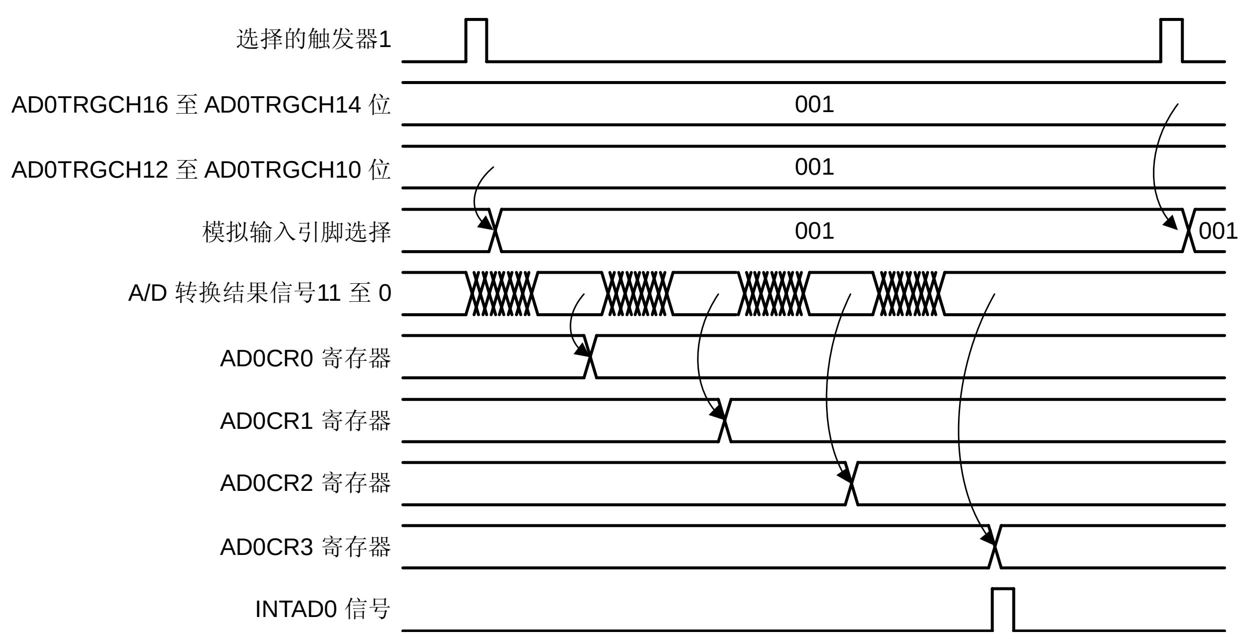
选择触发器	模拟输入引脚	A/D 转换结果扩展寄存器
选择触发 1	ANInx[注 1]	ADnCR0[注 3]
	ANInx[注 1]	|
	ANInx[注 1]	ADnCRm[注 3]
选择触发 2	ANIny[注 2]	ADnCR0[注 3]
	ANIny[注 2]	|
	ANIny[注 2]	ADnCRm[注 3]

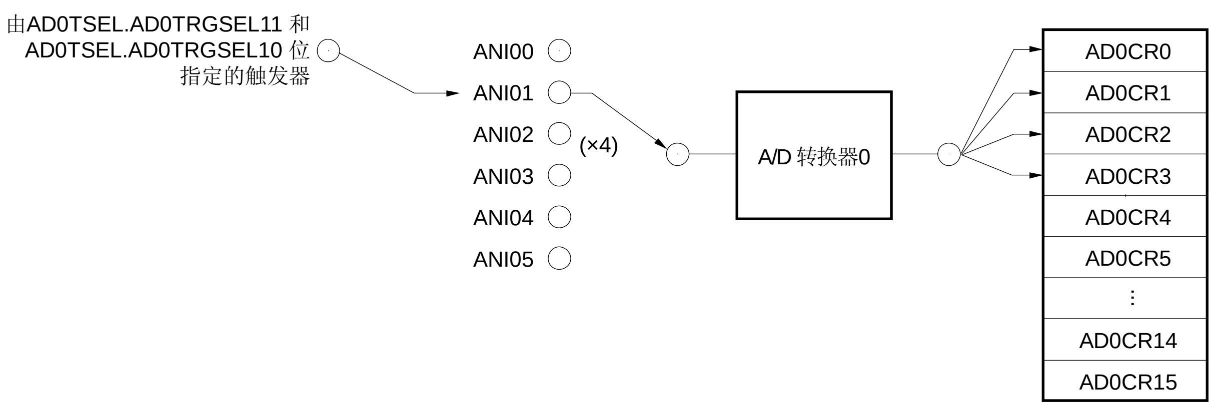
注 **1.** 由 ADnCH1.ADnTRGCH12 至 ADnCH1.ADnTRGCH10 位设置
2. 由 ADnCH1.ADnTRGCH16 至 ADnCH1.ADnTRGCH14 位设置
3. 可以由 ADnCHEN 寄存器设置为两次或多次

注意事项 1. 请确保设置硬件触发模式，也设置转换通道指定模式。
2. 请确保使用低位设置 ADnCHEN 寄存器，对齐在底部。禁止其它设置。
3. ADnCH2 寄存器的设置无效。
4. 不使用 ADnECRa，ADnECRaH，ADnFLG 和 ADnFLGB 寄存器。如果读取这些寄存器，会得到 0000H 和 00H。
5. 如果在 A/D 转换操作期间产生选择触发 1，则忽略。在 A/D 转换（在 INTADn 信号产生之后）完成之后，产生一个触发器时，可以接受下一个选择触发 1。

备注 A/D 转换器 0：n = 0，k = 0 至 5，m = 0 至 15
A/D 转换器 1：n = 1，k = 0 至 7，m = 0 至 15

图 12-19. 转换通道指定模式下的操作示例：A/D 转换器 0





（1）　AD0CE 位 = 1（使能）
（2）　产生由 AD0TSEL.AD0TRGSEL11 位和 AD0TSEL.AD0TRGSEL10 位指定的触发
（3）　A/D 转换 ANI01 引脚的信号
（4）　转换结果存储在 AD0CR0 寄存器
（5）　A/D 转换 ANI01 引脚的信号
（6）　转换结果存储在 AD0CR1 寄存器
（7）　A/D 转换 ANI01 引脚的信号
（8）　转换结果存储在 AD0CR2 寄存器
（9）　A/D 转换 ANI01 引脚的信号
（10）　转换结果存储在 AD0CR3 寄存器
（11）　AD0SCM.AD0CS 位 = 0
（12）　产生 INTAD0 中断请求信号
（13）　当下一个触发器被输入时，返回到（3）
（14）　设置 AD0CE 位为 0 来结束（停止）

备注　这是个操作示例，其中 AD0SCM.AD0PLM，AD0TRG1 和 AD0TRG0 位 = 001，AD0CTL0.AD0MD1 和 AD0CTL0.AD0MD0 位 = 10，AD0CHEN 寄存器 = 000FH，AD0CH1.AD0TRGCH12 至 AD0CH1.AD0TRGCH10 位 = 001，且 AD0CH1.AD0TRGCH16 至 AD0CH1.AD0TRGCH14 位 = 001。

12.4.11 扩展缓冲模式（扩展操作模式）

当 ADnSCM.ADnCE 位设置为 1 时，A/D 转换器等待一个触发。由 ADnTSEL.ADnTRGSEL11 和 ADnTSEL.ADnTRGSEL10 位指定的选择触发 1 产生时，或由 ADnTSEL.ADnTRGSEL21 和 ADnTSEL.ADnTRGSEL20 位指定的选择触发 2 产生时，转换器启动 A/D 转换。

当转换启动后，ADnSCM.ADnCS 位设置为 1（转换正在进行中）。

如果在 A/D 转换期间写入 ADnSCM 寄存器，转换停止，转换器再次等待一个触发。

用于选择触发 x 的模拟输入引脚由 ADnCHx.ADnTRGCHx2 至 ADnCHx.ADnTRGCHx0 和 ADnCHx.ADnTRGCHx6 至 ADnCHx.ADnTRGCHx4 位指定。每次选择触发 x 产生时，由 ADnCHx.ADnTRGCHx2 至 ADnCHx.ADnTRGCHx0 和 ADnCHx.ADnTRGCHx6 至 ADnCHx.ADnTRGCHx4 位 指定的模拟输入引脚依次选中。

当使用选择触发 1 时，当首次产生触发器时，由 ADnTRGCH12 至 ADnTRGCH10 位指定的模拟输入引脚的信号转换，结果存储在 A/Dn 转换结果扩展缓冲寄存器 0，并产生一个 A/Dn 转换结束中断请求信号（INTADn）。当第二次产生触发器时，由 ADnTRGCH16 至 ADnTRGCH14 位指定的模拟输入引脚的信号转换。结果存储在 A/Dn 转换结果扩展缓冲寄存器 0，同时，第一个存储在 A/Dn 转换结果扩展缓冲寄存器 0 中的值存储到 A/Dn 转换结果扩展缓冲寄存器 1，然后产生 INTADn 中断请求信号。使用选择触发 1 的 A/D 转换，最多有三个 A/Dn 转换结果扩展缓冲寄存器，0 至 2 可供使用。之后产生选择装载触发 1 时，A/Dn 转换结果扩展缓冲寄存器 0 至 2 的值传送到 ADnECR0 至 ADnECR2 寄存器。在 A/D 转换完成后，转换器等待触发且 ADnSCM.ADnCE 位保持设置为 1。

当使用选择触发 2 时，当首次产生触发器时，由 ADnTRGCH22 至 ADnTRGCH20 位指定的模拟输入引脚的信号转换，结果存储在 A/Dn 转换结果扩展缓冲寄存器 3，然后产生一个 A/Dn 转换结束中断请求信号（INTADn）。当第二次产生触发器时，由 ADnTRGCH26 至 ADnTRGCH24 位指定的模拟输入引脚的信号转换。结果存储在 A/Dn 转换结果扩展缓冲寄存器 4，同时，第一个存储在 A/Dn 转换结果扩展缓冲寄存器 3 中的值存储到 A/Dn 转换结果扩展缓冲寄存器 4，然后产生 INTADn 中断请求信号。使用选择触发 2 的 A/D 转换，最多有两个 A/Dn 转换结果扩展缓冲寄存器，3 和 4 可供使用。再次产生选择装载触发 2 时，A/Dn 转换结果扩展缓冲寄存器 3 和 4 的值被传送到 ADnECR3to ADnECR4 寄存器。在 A/D 转换完成后，转换器等待触发且 ADnSCM.ADnCE 位保持设置为 1。

因此，ADnECR0 至 ADnECR4 寄存器的内容可以一次全部保存到 RAM。

该操作适合对于只有很短时间可以保存转换结果，且产生触发时需要监视两个或多个模拟输入信号的应用。

选择触发器	模拟输入引脚	A/D 转换结果扩展寄存器
选择触发 1	ANInx[注 1]	ADnECR0 至 ADnECR2
选择触发 1	ANIny[注 2]	ADnECR0，ADnECR1
选择触发 1	ANInx[注 1]	ADnECR0
选择触发 2	ANIns[注 3]	ADnECR3，ADnECR4
选择触发 2	ANInt[注 4]	ADnECR3

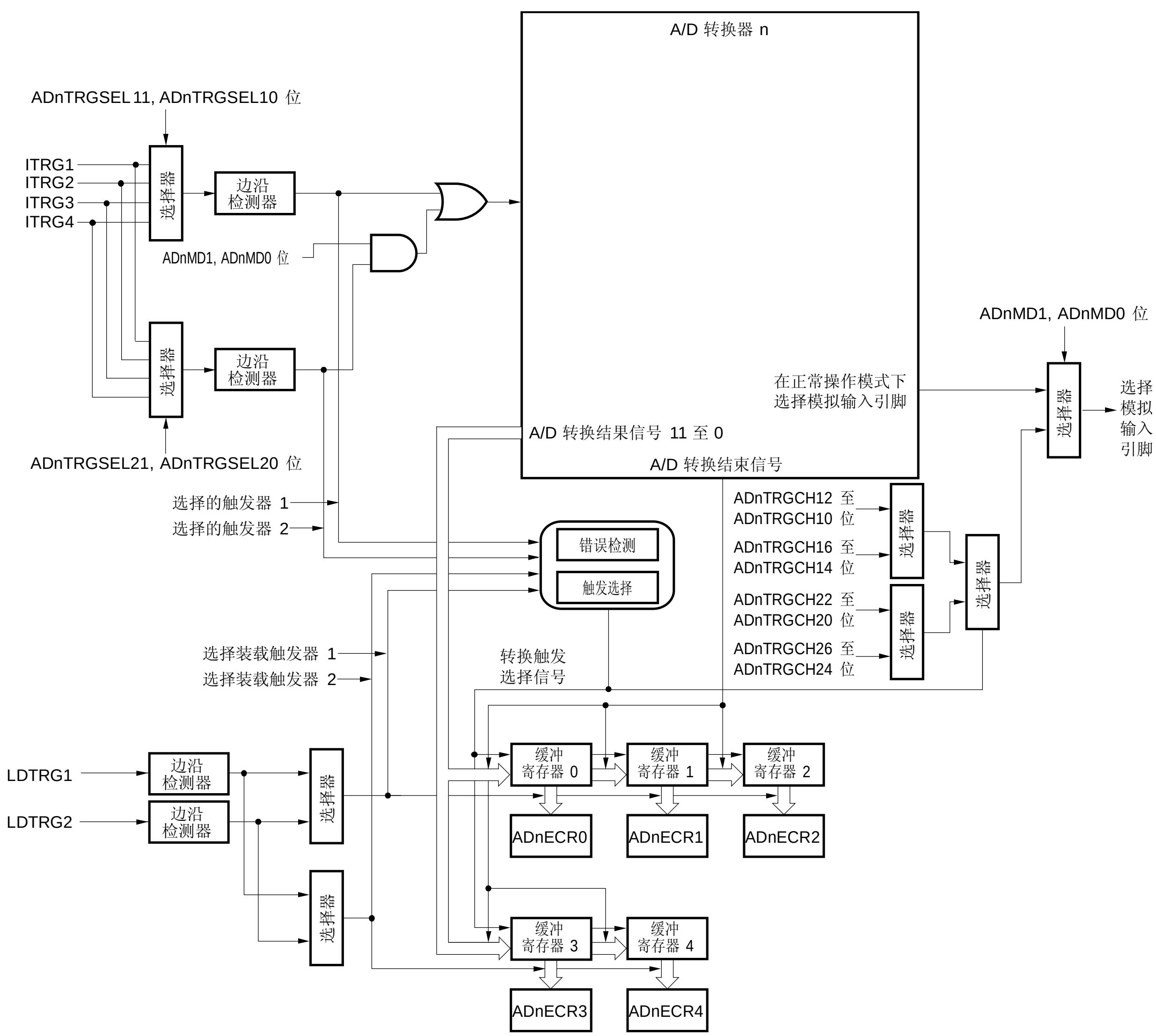
注
1. 由 ADnCH1.ADnTRGCH12 至 ADnCH1.ADnTRGCH10 位设置
2. 由 ADnCH1.ADnTRGCH16 至 ADnCH1.ADnTRGCH14 位设置
3. 由 ADnCH2.ADnTRGCH22 至 ADnCH2.ADnTRGCH20 位设置
4. 由 ADnCH2.ADnTRGCH26 至 ADnCH2.ADnTRGCH24 位设置

注意事项 **1.** 在扩展缓冲模式下，请确保设置硬件触发模式，并将 **ADnCHEN** 寄存器设置为 **0001H**。

2. 转换结果被存储在 **ADnECRa** 寄存器内。**ADnCRm** 寄存器的值不确定。

备注 n = 0，1

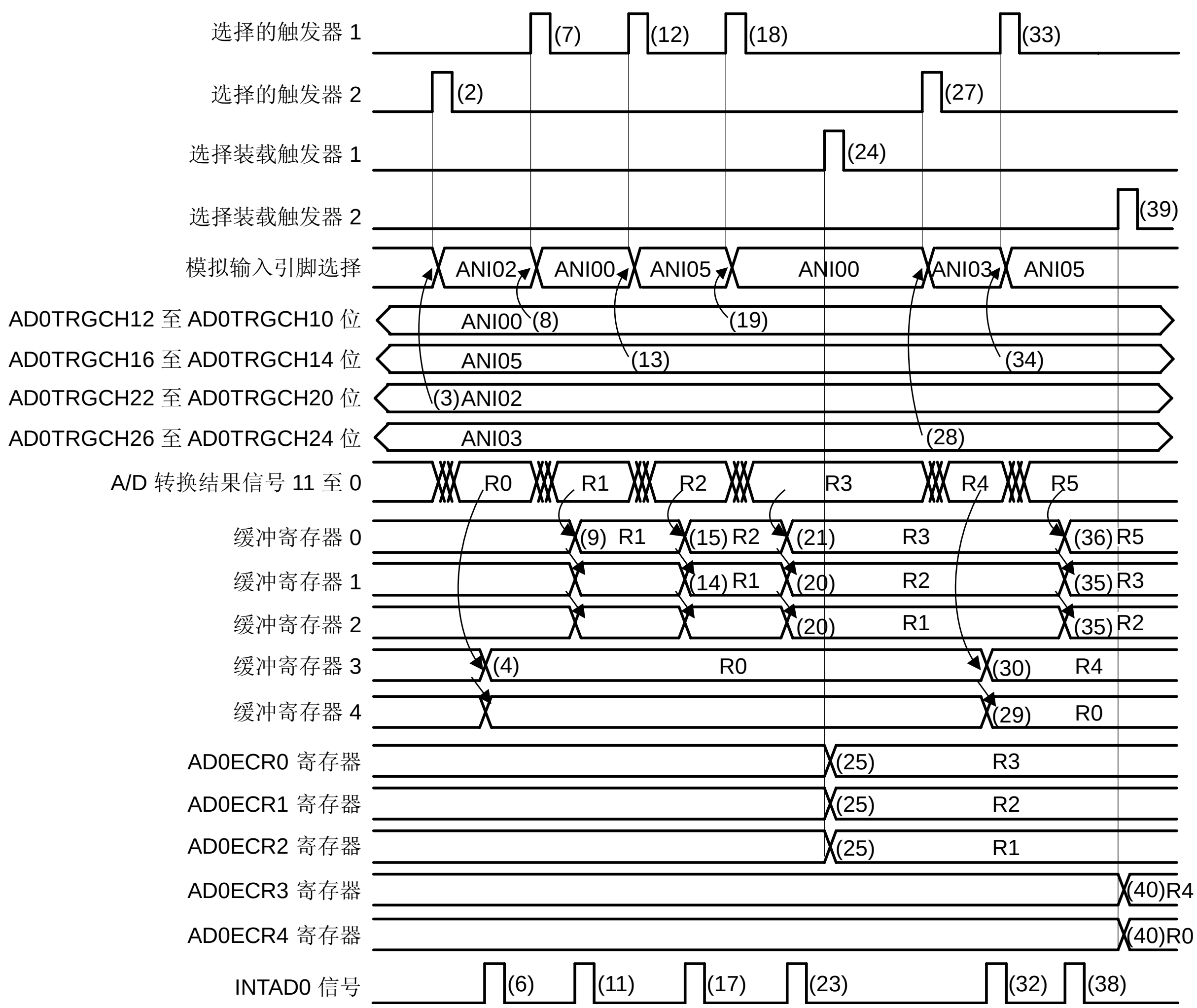
图 **12-20.** 扩展缓冲模式下的框图



备注 **1.** 缓冲寄存器 0 至 4：A/Dn 转换结果扩展缓冲寄存器 0 至 4

2. n = 0，1

图 12-21. 扩展缓冲模式下的操作示例：**A/D** 转换器 **0**（**1/2**）



备注　**1.** 缓冲寄存器 0 至 4：A/Dn 转换结果扩展缓冲寄存器 0 至 4

2. R0 至 R6：转换结果

3. n = 0，1

图 12-21. 扩展缓冲模式下的操作示例：A/D 转换器 0（2/2）

（1） AD0CE 位 = 1（使能）
（2） 选择触发 2 产生
（3） A/D 转换 ANI02 引脚的信号
（4） 转换结果被存储在缓冲寄存器 3
（5） AD0SCM.AD0CS 位 = 0
（6） 产生 INTAD0 中断请求信号
（7） 选择触发 1 产生
（8） A/D 转换 ANI00 引脚的信号
（9） 转换结果存储在缓冲寄存器 0
（10） AD0SCM.AD0CS 位 = 0
（11） 产生 INTAD0 中断请求信号
（12） 选择触发 1 产生
（13） A/D 转换 ANI05 引脚的信号
（14） 从缓冲寄存器 0 转移到缓冲寄存器 1
（15） 转换结果存储在缓冲寄存器 0
（16） AD0SCM.AD0CS 位 = 0
（17） 产生 INTAD0 中断请求信号
（18） 选择触发 1 产生
（19） A/D 转换 ANI00 引脚的信号
（20） 从缓冲寄存器 0 转移到缓冲寄存器 1 至缓冲寄存器 2
（21） 转换结果存储在缓冲寄存器 0
（22） AD0SCM.AD0CS 位 = 0
（23） 产生 INTAD0 中断请求信号
（24） 选择装载触发 1 产生
（25） 从缓冲寄存器 0 至 2 转移到 AD0ECR0 至 AD0ECR2
（26） AD0SCM.AD0CS 位 = 0
（27） 选择触发 2 产生
（28） A/D 转换 ANI03 引脚的信号
（29） 从缓冲寄存器 3 转移到缓冲寄存器 4
（30） 转换结果存储在缓冲寄存器 3
（31） AD0SCM.AD0CS 位 = 0
（32） 产生 INTAD0 中断请求信号
（33） 选择触发 1 产生
（34） A/D 转换 ANI05 引脚的信号
（35） 从缓冲寄存器 0 至缓冲寄存器 1 转移到缓冲寄存器 2
（36） 转换结果存储在缓冲寄存器 0
（37） AD0SCM.AD0CS 位 = 0
（38） INTAD0 中断请求信号产生
（39） 选择装载触发 2 产生
（40） 从缓冲寄存器 3 和 4 转移到 AD0ECR3 和 AD0ECR4 寄存器
（41） AD0SCM.AD0CS 位 = 0
（42） 当下一个触发输入时，按照那个触发来执行操作
（43） 将 ADnCE 位 设置为 0 来结束（停止）

（1）错误检测功能

扩展缓冲模式具有错误检测功能。如果在 A/D 转换期间产生一个触发器（选择触发 1，选择触发 2，选择装载触发 1 或选择装载触发 2），则发生错误。错误通过 ADnFLG.ADnTERR2 和 ADnFLG.ADnTERR1 标志，以及 ADnFLGB.ADnTERRB2 和 ADnFLGB.ADnTERRB1 标志来检测。

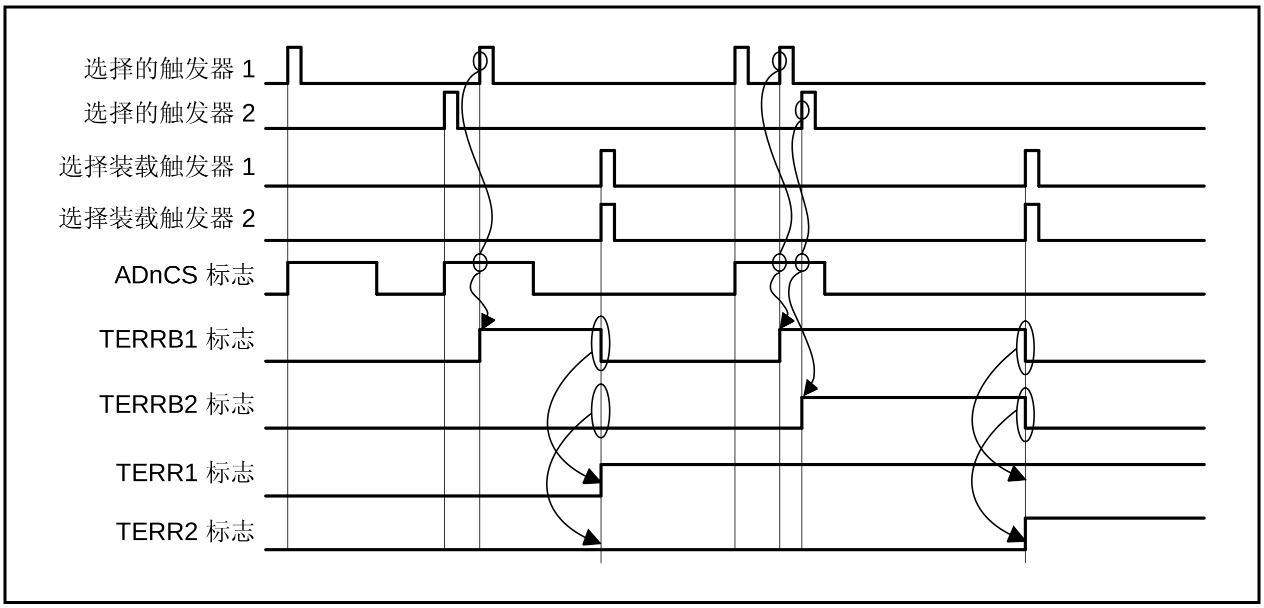
注意事项 **1. 当异步信号 ITRG1 至 ITRG4，LDTRG1 和 LDTRG2 信号同步时，产生选择触发 1，选择触发 2，选择装载触发 1 和选择装载触发 2。虽然输入这些触发器的时序看起来是同时的，它们的同时操作无法保证，因为异步信号同步。**

2. 触发器产生过一次之后（无错误发生），在最多 2.5 基本时钟（f_{AD01}）的时期内，即使再次产生选择触发 1 或 2，也会忽略。

（**a**） **在 A/D 转换期间，产生选择触发 1 或 2 的检测错误**

如果在 A/D 转换期间产生选择触发 1，ADnFLGB.ADnTERRB1 标志设置为 1，且忽略选择触发 1 引起的 A/D 转换。如果再次产生选择装载触发 1，ADnTERRB1 标志的值存储到 ADnFLG.ADnTERR1 标志。

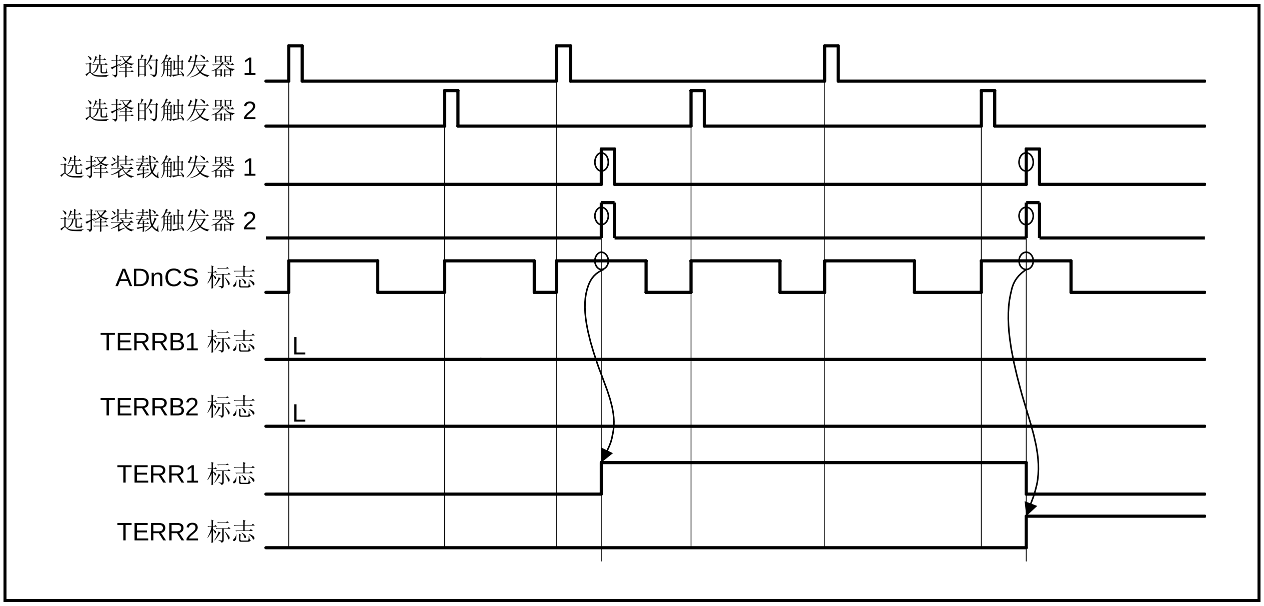
与此类似，如果在 A/D 转换期间产生选择触发 2，ADnFLGB.ADnTERRB2 标志设置为 1，且忽略选择触发 2 引起的 A/D 转换。如果再次产生选择装载触发 2，ADnTERRB2 标志的值存储到 ADnFLG.ADnTERR2 标志。



（**b**） **在 A/D 转换期间，产生选择装载触发 1 或 2 的检测错误**

如果在 A/D 转换期间产生选择装载触发 1，该 A/D 转换使用选择触发 1，ADnFLG.ADnTERR1 标志设置为 1。A/D 转换和装载操作正常执行。

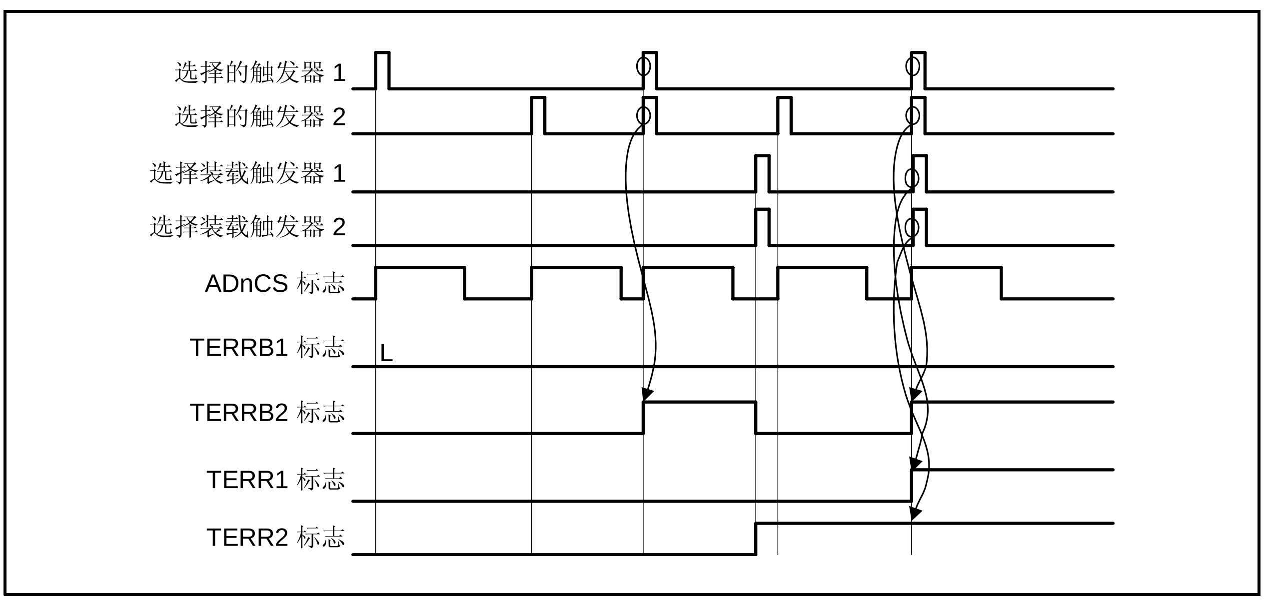
与此类似，如果在 A/D 转换期间产生选择装载触发 2，该 A/D 转换使用选择触发 2，ADnTERR2 标志设置为 1。A/D 转换和装载操作正常执行。



（**c**） **同时产生选择装载触发 1 和 2，或者同时产生选择触发 1 和 2 与选择装载触发 1 和 2 的检测错误**

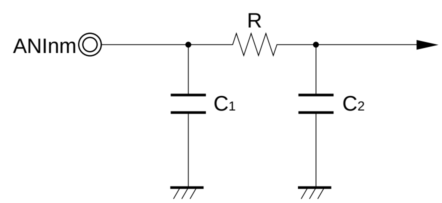
如果选择触发 1 和 2 同时产生，使用选择触发 1 的 A/D 转换启动，且忽略选择触发 2。因此 ADnFLGB.ADnTERRB2 标志设置为 1。

如果选择触发 1 和 2，与选择装载触发 1 和 2 同时产生，ADnFLGB.ADnTERRB2，ADnFLG.ADnTERR1 和 ADnFLG.ADnTERR2 标志设置为 1。选择触发 1 引起的 A/D 转换与选择装载触发 1 和 2 的装载操作正常执行。忽略选择触发 2。



12.5 内部等效电路

下图展示了模拟输入块的等效电路。



R	C_1	C_2
5.1 kΩ	15 pF	3.9 pF

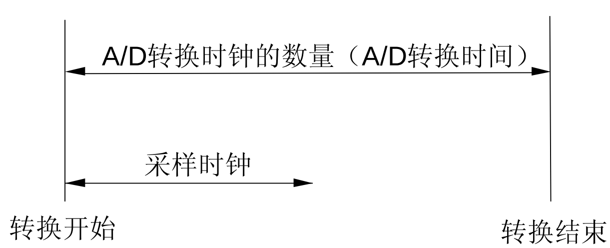
备注 **1.** 最大值如下（参考值）

2. 当 n = 0 时，m = 0 至 5

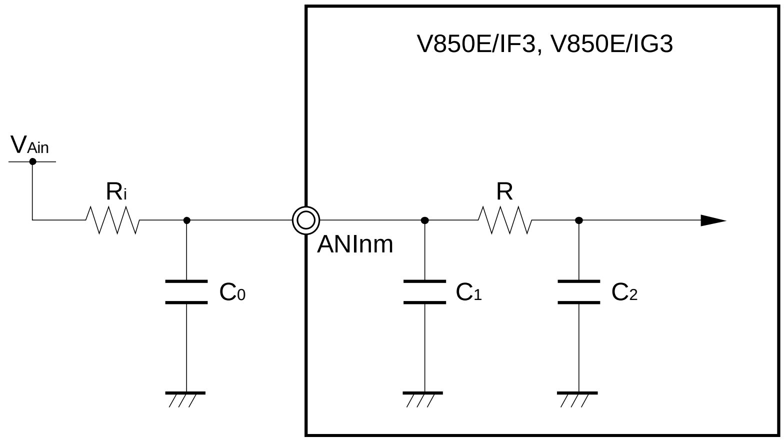
当 n = 1 时，m = 0 至 7

ADnCTC 寄存器				A/D 转换时钟的数量（f_{AD01}）	采样时钟的数量
ADnFR3 位	ADnFR2 位	ADnFR1 位	ADnFR0 位		
0	0	0	0	89	69.5
0	0	0	1	88	68.5
0	0	1	0	57	37.5
0	0	1	1	56	36.5
0	1	0	0	41	21.5
0	1	0	1	40	20.5
0	1	1	0	35	15.5
0	1	1	1	34	14.5
1	0	0	0	34	14.5
1	0	0	1	33	13.5
1	0	1	0	33	13.5
1	0	1	1	32	12.5
1	1	0	0	32	12.5
1	1	0	1	31	11.5
1	1	1	0	31	11.5
1	1	1	1	30	10.5

注意事项 采样时钟的数量包括在 A/D 转换时钟的数量内。



计算 A/D 转换器 0 和 1 的总误差的示例如下。



fXX（MHz）	A/D 转换时间（μs）	采样（μs）	R（kΩ）	C1（pF）	C2（pF）	C0（pF）	Ri（kΩ）	采样错误（LSB）[注]
64	2.00（31/fAD01）	0.78（12.5/fAD01）	5.1	15	3.9	100	1.0	364.8
						100	0.5	30.4
						100	0.25	0.1或更低
						100	0.125	0.1或更低
						50	1.0	62.4
						50	0.5	0.8
						50	0.25	0.1或更低
						50	0.125	0.1或更低

注　考虑信号源阻抗时，错误是"采样误差 + 总误差"。

备注
1. 这些值是通过计算得到的参考值，当 V_{Ain} 输入 0 V 至 5 V 的同时启动采样，模拟 R_i 和 C_0 对 C_2 电压的影响。
2. 当 n = 0 时，m = 0 至 5
当 n = 1 时，m = 0 至 7
3. f_{XX}：系统时钟频率
f_{AD01}：基本时钟频率

12.6 注意事项

12.6.1 停止转换操作

当 0 写入到 ADnSCM.ADnCE 位时，正在进行的转换操作停止。此时，A/Dn 转换结果寄存器 m（ADnCRm）和 A/Dn 转换结果扩展寄存器 a（ADnECRa）内的转换结果不确定。因此，在 A/D 转换完成之后（在发出 A/Dn 转换结束中断请求信号（INTADn）之后），读取 A/D 转换结果，然后按照需要将 0 写入到 ADnCE 位。

注意在所有模式下，即使产生了 INTADn 信号之后，ADnCE 位也不会清除为 0。

备注　n = 0，1
　　　m = 0 至 15

12.6.2 在硬件触发模式、转换通道指定模式和扩展缓冲模式下转换操作期间触发的间隔

在硬件触发模式、转换通道指定模式和扩展缓冲模式下，转换操作期间，忽略输入的触发。因此，在硬件触发模式、转换通道指定模式和扩展缓冲模式下，触发的间隔（输入时间）必须长于 ADnCTC.ADnFR3 至 ADnCTC.ADnFR0 位（参见 **表 12-2 A/D 转换时钟的数量和 A/D 转换时间**）指定的 A/D 转换时间。

备注　n = 0，1

12.6.3 写入 ADnSCM 寄存器

（1）重启 A/D 转换

要重启 A/D 转换，将相同的值写入 ADnSCM 寄存器。要改变 ADnPLM，ADnTRG1 和 ADnTRG0 位，请确保将 ADnCE 位设置为 0。

（2）A/D 转换结束和写入 ADnSCM 寄存器的冲突

在 A/D 转换操作期间，如果 A/D 转换的完成和 ADnSCM 寄存器的写入发生冲突，如果产生 A/Dn 转换结束中断请求信号（INTADn），转换结果正确存储到 ADnCRm 和 ADnECRa 寄存器。如果不产生 INTADn 信号，则放弃 A/D 转换操作。因此，ADnCRm 和 ADnECRa 寄存器保持上一次的转换结果。

（3）连续写入 ADnSCM 寄存器

当转换操作使能（ADnCE 位 = 1）时，要连续写入 ADnSCM0 寄存器，请确保等待的时间至少是 5 个基本时钟（f_{AD01}）。

当 ADnCE 位 = 0 时写入 ADSCMn 寄存器之后，如果 ADnCE 位设置为 1 时，可以连续写入 ADnSCM0 寄存器。

备注　n = 0，1

12.6.4 A/D 转换启动时序

在转换通道指定模式和扩展缓冲模式下，和正常操作模式对比，A/D 转换的启动会延迟最多 1.5 个基本时钟（f_{AD01}）。

12.6.5 待机模式下的操作

（1）HALT 模式

A/D 转换操作继续。如果通过一个没有屏蔽的可屏蔽中断请求信号解除 HALT 模式，则保持 ADnSCM，AdnCRm 和 ADnECRa 寄存器的值。

（2）IDLE 模式和 STOP 模式

不执行转换操作，因为供给 A/D 转换器 0 和 1 的时钟停止。

当设置 IDLE 或 STOP 模式时，请确保将 ADnSCM.ADnCE 位设置为 0。此时，推荐设置为 A/D 省电模式（ADnSCM.ADnPS 位 = 0）。

备注 n = 0，1
m = 0 至 15

12.6.6 在转换通道指定模式和扩展缓冲模式下接受触发的时序

在转换通道指定模式和扩展缓冲模式下，即使再次产生选择触发 1 或 2，也会忽略，直到由第一个产生的选择触发 1 或 2 触发的 A/D 转换启动之后，产生 A/Dn 转换结束中断请求信号（INTADn）。在扩展缓冲模式下，根据指定的错误条件，如果在此期间（但是 **12.4.11**（**1**）的**注意事项 2 错误检测功能**中的情况除外）产生选择触发 1 或 2，或产生选择装载触发 1 或 2，对应的错误标志被设置为 1。

备注 n = 0，1

12.6.7 A/D 转换结果的变化

A/D 转换的结果可能会因为供电电压的波动而变化，或者可能会受到噪声的影响。要降低变化，在程序中采取反向措施，比如取 A/D 转换结果的平均值。

12.6.8 A/D 转换结果迟滞特性

连续比较类型 A/D 转换器在内部采样&保持电容中保持模拟输入电压，然后执行 A/D 转换。在 A/D 转换完成之后，模拟输入电压保持在内部采样&保持电容中。结果，如果来自模拟输入源的输出阻抗太高，可能会发生下列现象。

- 当 A/D 转换使用相同的通道时，如果电压比上次 A/D 转换更高或更低，转换结果受上次值的影响的话，可能会出现迟滞特性。即使转换执行时是相同的电势，结果可能不同。
- 当切换模拟输入通道时，转换结果受上个通道的值的影响的话，可能会出现迟滞特性。这是因为 A/D 转换使用同一个 A/D 转换器。即使转换执行时是相同的电势，结果可能不同。

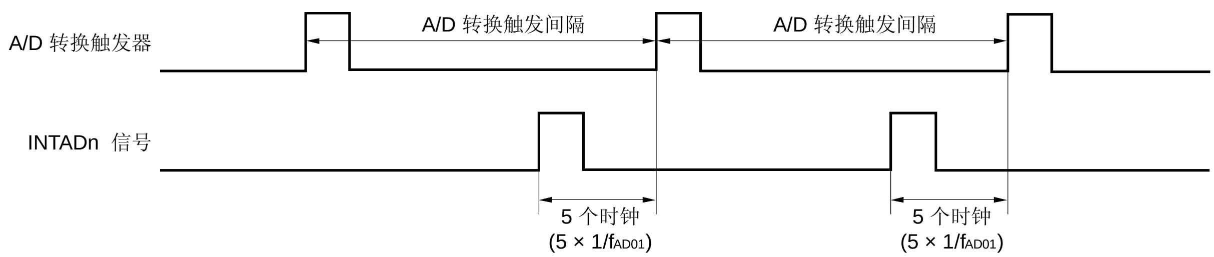
<R> 要获得更准确的转换结果，降低来自模拟输入源的输出阻抗或在同一通道上连续执行两次 A/D 转换，并丢弃第一次的转换结果。

12.6.9 连续转换的 A/D 转换触发间隔

连续转换的 A/D 转换触发间隔，在下一个触发输入之前，要保障至少经过下列的最小触发间隔时间。否则，触发将会无效（不保留）。

最小触发间隔时钟计数 = A/D 转换时钟计数 + 5 个时钟
最小触发间隔时间 = 最小触发间隔时钟计数 × 1/f_{AD01}

示例　f_{AD01} = 16 MHz，A/D 转换时间= 2 μs，A/D 转换时钟计数 = 32 个时钟
最小触发间隔时钟计数 = 32 + 5 = 37
最小触发间隔定时器 = 37 × 1/16 = 2.3125 [μs]



备注　n = 0，1

12.7 如何阅读 A/D 转换器特征表

本节解释仅适用于 A/D 转换器的专业术语。

（1） 分辨率

分辨率为能够识别的最低模拟输入电压，也就是说，每位数字输出所占模拟输入电压的百分比，称之为 1 LSB（最低有效位）。1 LSB 对满量程的比率表示为%FSR（满标度量程）。%FSR 是百分比表示的可转换模拟输入电压比率，不管分辨率如何，总是可以用下式表示

1%FSR =（可转换的模拟输入电压的最大值 – 可转换的模拟输入电压的最小值）/100
=（AV_{REFPn} – 0）/100
= AV_{REFPn}/100

当分辨率为 12 位时，1 LSB 表示如下。

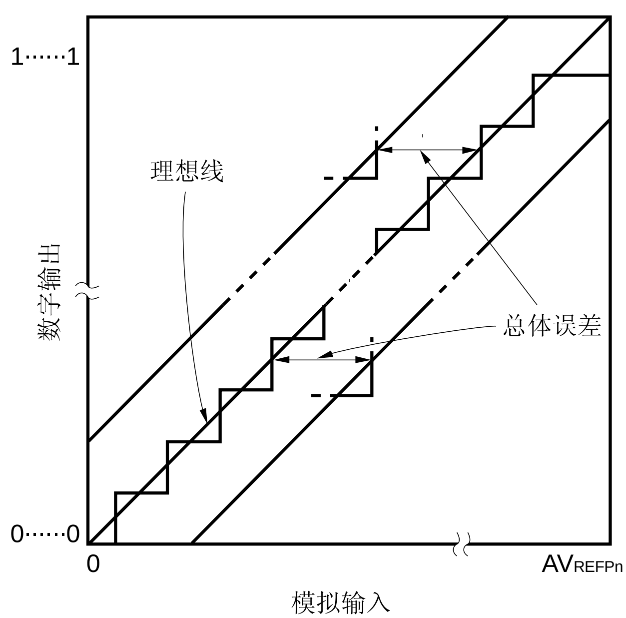
1 LSB = $1/2^{12}$ = 1/4,096
= 0.024%FSR

精确度是由总体误差决定，与分辨率无关。

（2） 总体误差

总体误差是实际测量值和理论值之间的最大误差值。
零标度误差、满量程误差、线性误差，这些误差的综合构成总体误差。
请注意，在特性表中的总体误差里没有包含量化误差。

图 12-22. 总体误差

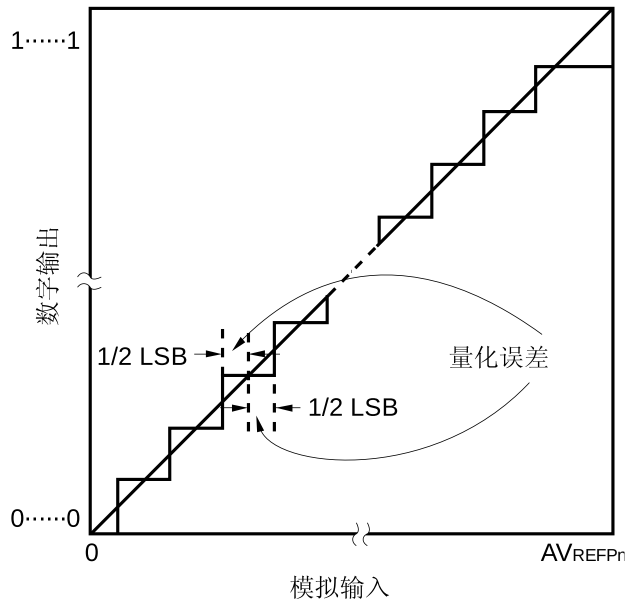


（3）量化误差

当模拟值转换为数字值时，会不可避免的产生±1/2 LSB 的误差。在 A/D 转换器内，±1/2 LSB 范围内的模拟输入电压转换为相同的数字值，所以量化误差不能避免。

请注意，量化误差不包括在特性表中的总体误差、零标度误差、满量程误差、积分线性误差或微分线性误差之中。

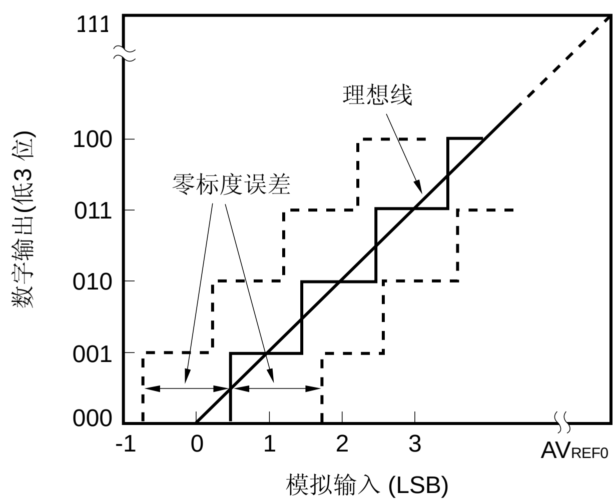
图 12-23. 量化误差



（4）零标度误差

当数字输出从 0……000 改变到 0……001 时，零标度误差就是模拟输入电压的实际测量值和理论值（1/2 LSB）之间的差别。

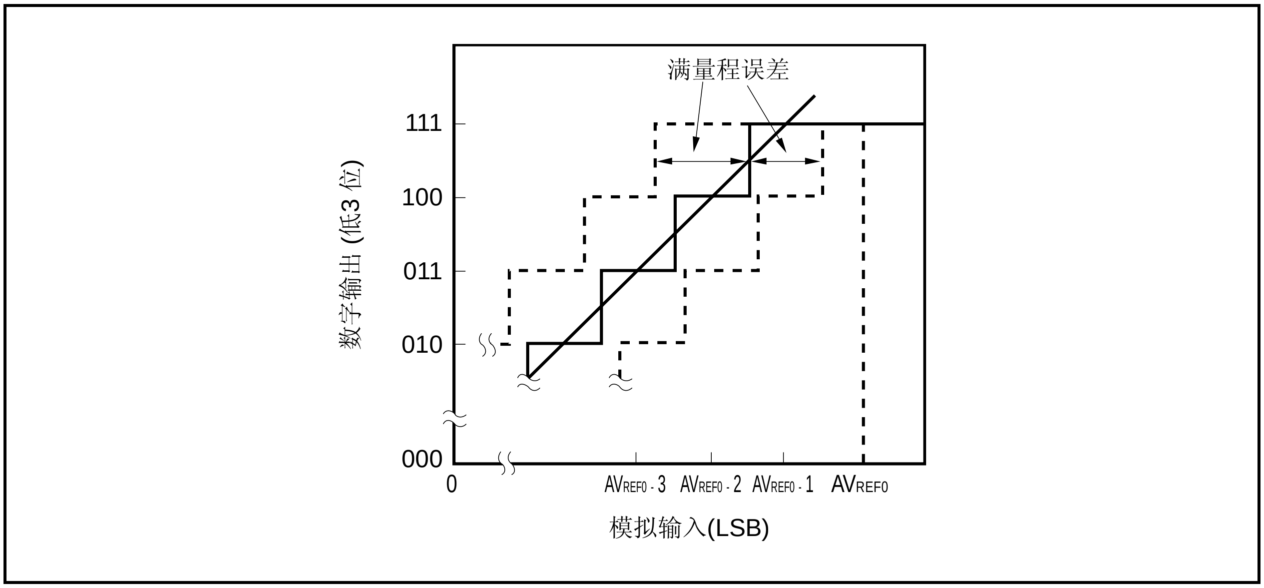
图 12-24. 零标度误差



（5）满量程误差

当数字输出从 1…110 改变到 1…111 时，满量程误差就是模拟输入电压的实际测量值与理论值（满量程 – 3/2 LSB）之间的差别。

图 12-25. 满量程误差

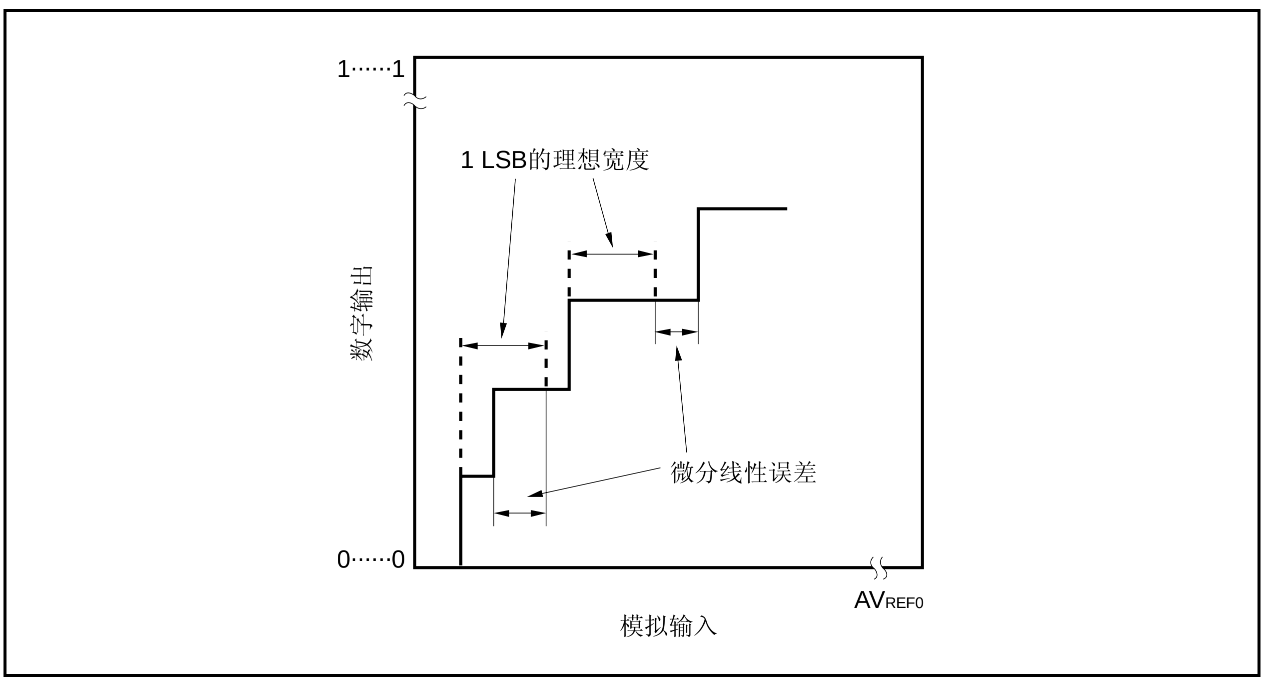


（6）微分线性误差

理想状态下，编码输出的宽度是 1 LSB。该误差显示了实际测量值和其理论值之间的差别。

表示输入到相同通道的模拟输入引脚的电压从AV_{SS}逐位增加到 AV_{REF0}时，A/D转换的基本特性。当输入电压增加或减小时，或使用两个或更多的通道时，参见 **12.7 （2）总体误差**。

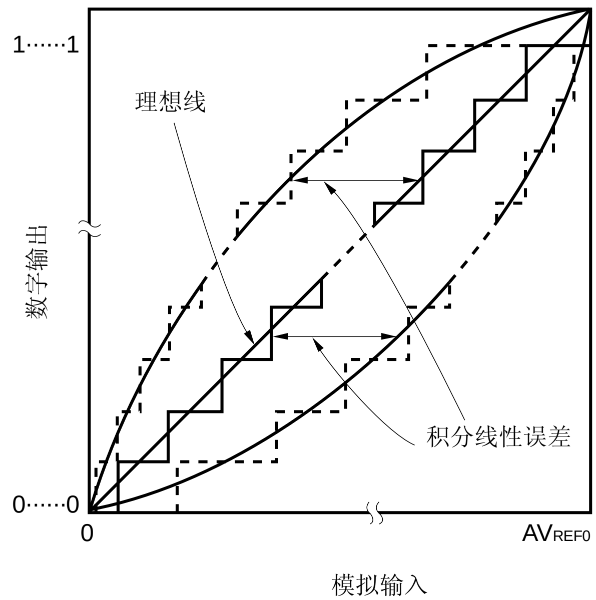
图 12-26. 微分线性误差



（7）积分线性误差

积分线性误差显示转换特性偏离理想线性关系的程度。它表示当零标度误差和满量程误差均为 0 时，实际测量值和其理论值之间的最大误差。

图 12-27. 积分线性误差



（8）转换时间

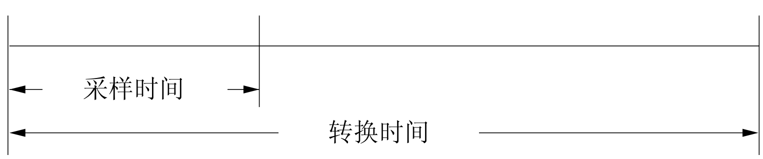
转换时间表示从触发信号产生到获得数字输出的时间。

特性表中，转换时间包含采样时间。

（9）采样时间

采样时间，即模拟开关打开，将待采样的模拟电压载入采样&保持电路的时间。

图 12-28. 采样时间



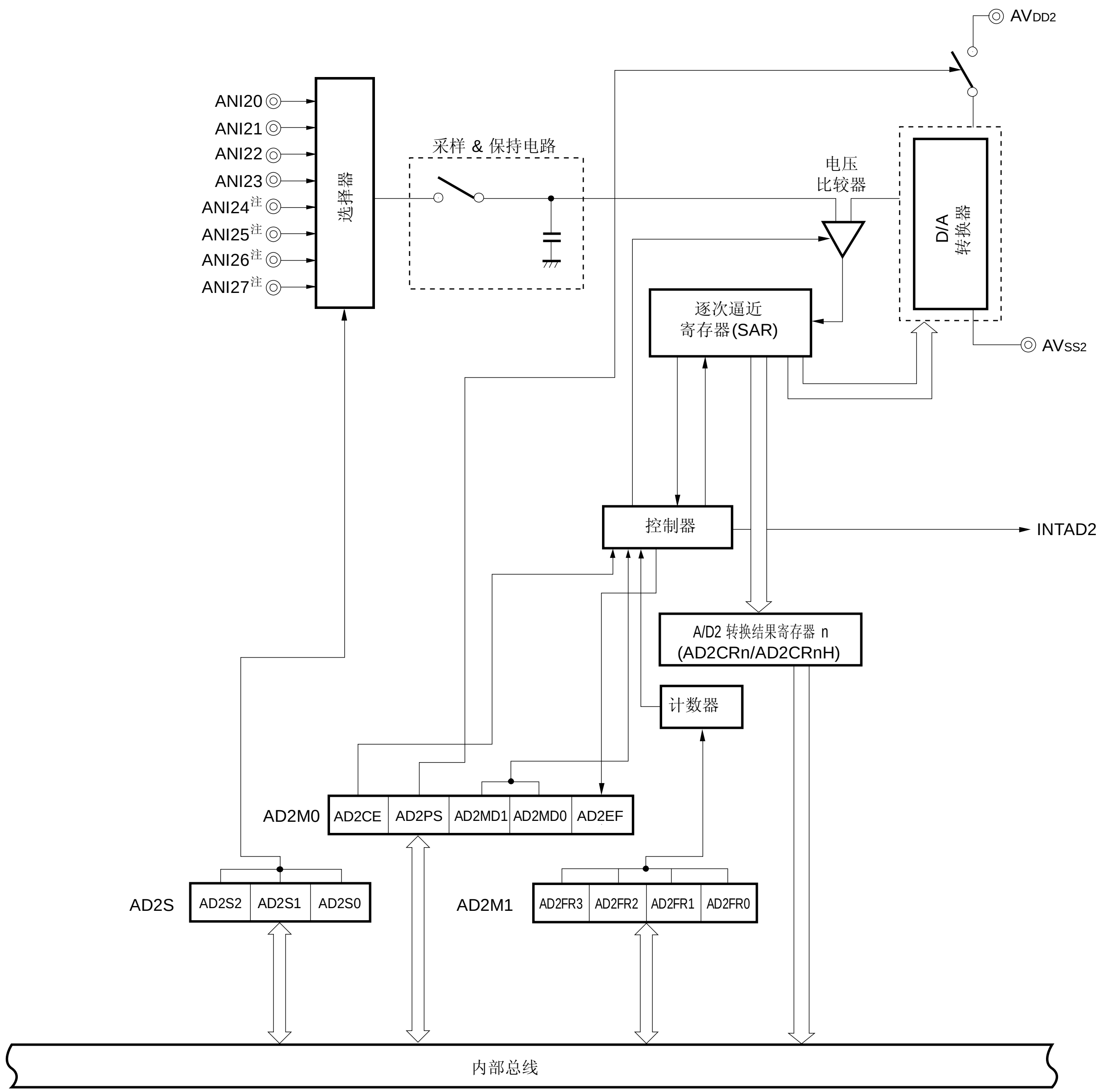
13.1 特征

- 片上 10 位分辨率 A/D 转换器
- 模拟输入
  V850E/IF3：ANI20 至 ANI23 (4 通道)
  V850E/IG3：ANI20 至 ANI27 (8 通道)
- A/D 转换结果寄存器
  V850E/IF3：AD2CR0 至 AD2CR3 (10 位 × 4)
  V850E/IG3：AD2CR0 至 AD2CR7 (10 位 × 8)
- A/D 转换触发模式
  软件触发模式
- A/D 转换操作模式
  连续选择模式
  连续扫描模式
  单次选择模式
  单次扫描模式
- 逐次逼近比较方法
- 操作电压：$EV_{DD0} = EV_{DD1} = EV_{DD2}$ (仅限 V850E/IG3) $= AV_{DD2} = 4.0$ 至 5.5 V

13.2 配置

框图如下所示。

图 13-1. A/D 转换器 2 的框图



备注　V850E/IF3：n = 0 至 3
V850E/IG3：n = 0 至 7

注　仅限 V850E/IG3

注意事项 **1.** 如果在模拟输入引脚（**ANI2n**）或在 **A/D** 转换器参考电压输入引脚（**AV$_{DD2}$**）有噪声，噪声可能会产生错误的转换结果。

需要软件处理来防止错误转换结果对系统的负面影响。

下面是一个软件处理的示例。

- 对一定数量的 **A/D** 转换取平均值，并将该值用作 **A/D** 转换结果**.**
- 连续执行一定数量的 **A/D** 转换，并使用这些结果，忽略任何可能获得的意外结果。
- 如果获得的一个 **A/D** 转换结果被判定已经产生系统误操作，请确保在执行误操作处理之前，重新检查该系统误操作。

2. 不要将超出 **AV$_{SS2}$** 至 **AV$_{DD2}$** 范围的电压输入到用作 **A/D** 转换器 **2** 的模拟输入引脚。

A/D 转换器 2 包括下列硬件。

表 13-1. A/D 转换器 2 的配置

项目	配置
模拟输入	V850E/IF3：ANI20 至 ANI23 (4 通道) V850E/IG3：ANI20 至 ANI27 (8 通道)
寄存器	逐次逼近寄存器 (SAR) V850E/IF3: A/D2 转换结果寄存器 0 至 3 (AD2CR0 至 AD2CR3) A/D2 转换结果寄存器 0H 至 3H (AD2CR0H 至 AD2CR3H)：只有高 8 位可以读取 V850E/IG3： A/D2 转换结果寄存器 0 至 7 (AD2CR0 至 AD2CR7) A/D2 转换结果寄存器 0H 至 7H (AD2CR0H 至 AD2CR7H)：只有高 8 位可以读取
控制寄存器	A/D 转换器 2 模式寄存器 0, 1 (AD2M0, AD2M1) A/D 转换器 2 通道指定寄存器 (AD2S)

(1) 逐次逼近寄存器 (SAR)

SAR 寄存器用于将模拟输入引脚的电压值和 D/A 转换器的电压档位器的值进行比较，并从最高有效位（MSB）开始保存结果。

如果 SAR 内的数据持续直到最低有效位（LSB）（A/D 转换结束），SAR 的内容（转换结果）被传输到 AD2CRn 寄存器。

当所有指定的 A/D 转换操作结束后，产生一个 A/D2 转换结束中断请求信号（INTAD2）。

(2) A/D 转换结果寄存器 n (AD2CRn)，A/D 转换结果寄存器 nH (AD2CRnH)

AD2CRn 寄存器用于保持 A/D 转换结果。转换结果存储在 AD2CRn 寄存器的高 10 位。当读取时，这些寄存器的低 6 位总为 0。

A/D 转换结果的高 8 位从 AD2CRn 寄存器读取。

要按照 16 位宽度读取 A/D 转换结果，指定 AD2CRn 寄存器。要读取高 8 位，指定 AD2CRnH 寄存器。

注意事项　**根据写入 AD2M0, AD2M1 和 AD2S 寄存器的操作，AD2CRn 寄存器的内容可能变为不确定。在转换之后且在写入 AD2M0, AD2M1 和 AD2S 寄存器之前，从 AD2CRn 寄存器读取转换结果。如果使用其它任何读取过程，不能从 AD2CRn 寄存器读取正确的转换结果。**

(3) 采样&保持电路

采样&保持电路对输入电路选择的模拟输入信号进行采样，并将采样到的数据送入电压比较器。在 A/D 转换期间，该电路还保持采样到的模拟输入电压。

(4) 电压比较器

电压比较器将采样到的值和 D/A 转换器的电压档位器产生的电压进行比较。

(5) D/A 转换器

D/A 转换器连接在 AV_{DD2} 和 AV_{SS2} 之间，并产生一个电压用于和输入模拟信号进行比较。

(6) ANI2n 引脚

ANI2n 引脚是 A/D 转换器 2 的模拟输入引脚。该引脚输入将要进行 A/D 转换的模拟信号。除 AD2S 寄存器选择作为模拟信号输入的引脚之外，其他管脚都可以用作输入端口使用。

注意事项 **请确保输入到 ANI2n 引脚的电压不能超过额定值。如果一个大于等于 AV_{DD2} 的电压或低于等于 AV_{SS2} 的电压被输入到某个通道，通道的转换值不确定，并且其它通道的转换值可能也会受到影响。**

注意事项 **1. 请确保输入到 ANI2n 引脚的电压不能超过额定值。如果一个大于或等于 AV_{DD2}，或者低于或等于 AV_{SS2} 的电压被输入到某个通道，此通道的转换值是不确定的，并且其它通道的转换值可能也会受到影响。**

2. 模拟输入引脚(ANI2n)复用作为输入端口引脚 (P7n)。当 ANI2n 的一个被选作 A/D 转换时，在转换期间，如果执行输入一个信号到端口 7，转换的分辨率会下降。

(7) AV_{DD2} 引脚

AV_{DD2} 引脚复用作为输入 A/D 转换器 2 的正极性供电电压和参考电压的引脚。基于 AV_{DD2} 和 AV_{SS2} 之间的电压，输入到 ANI2n 引脚转换为数字信号。

总是使该引脚的电势与 EV_{DD0}，EV_{DD1} 和 EV_{DD2}（仅限 V850E/IG3）引脚的电势相同，即使未使用 A/D 转换器 2。

AV_{DD2} 引脚的操作电压范围是 $EV_{DD0} = EV_{DD1} = EV_{DD2}$（仅限 V850E/IG3）$= AV_{DD2} = 4.0$ 至 5.5 V.

(8) AV_{SS2} 引脚

这是 A/D 转换器 2 的地引脚。总是使该引脚的电势与 EV_{SS0}，EV_{SS1} 和 EV_{SS2}（仅限 V850E/IG3）引脚的电势相同，即使未使用 A/D 转换器 2。

备注 V850E/IF3：n = 0 至 3
V850E/IG3：n = 0 至 7

13.3 控制寄存器

A/D 转换器 2 由下列寄存器来控制。

- A/D 转换器 2 模式寄存器 0, 1 (AD2M0 至 AD2M1)
- A/D 转换器 2 通道指定寄存器 (AD2S)

也用到下列寄存器。

- A/D2 转换结果寄存器 n (AD2CRn)
- A/D2 转换结果寄存器 nH (AD2CRnH)

(1) A/D 转换器 2 模式寄存器 0 (AD2M0)

AD2M0 寄存器用于指定操作模式并控制转换操作。

该寄存器可以按字节或按位进行读取或写入，但是，第 0 位是只读的。

系统复位后，该寄存器被设置为 00H。

复位后：00H R/W 地址：FFFFFB80H

	<7>	6	5	4	3	2	1	0
AD2M0	AD2CE	AD2PS	AD2MD1	AD2MD0	0	0	0	AD2EF

AD2CE	A/D转换操作的控制
0	转换操作停止
1	转换操作使能

AD2PS	A/D 转换控制
0	A/D 上电
1	A/D 下电

· 在AD2PS 位被设置为1 (A/D 电源打开)之后至少2 us，将AD2CE 位设置为1(当转换使能时)，A/D 转换器2转换的第一次结果有效。
如果在2 us时间之内，AD2CE 位被设置为1，转换操作被启动并在A/D转换时间之后结束，但是转换结果不确定。

· 未使用A/D 转换器2时，将AD2CE 位 (停止转换操作)和AD2PS 位 (关闭A/D电源) 设置为0，以降低功耗。

· 在A/D 转换操作期间 (AD2EF 位 = 1)，不要设置AD2PS2位。
未执行A/D 转换操作时，AD2CE和AD2PS 位可以被同时清除为0。

AD2MD1	AD2MD0	操作模式的规范
0	0	连续选择模式
0	1	连续扫描模式
1	0	单次选择模式
1	1	单次扫描模式

AD2EF	A/D 转换器2的状态 (状态)
0	A/D转换停止期间
1	A/D 转换工作期间

注意事项
1. 向第 0 位的写入被忽略。
2. A/D 转换启动之后，立即输入模拟信号的第一个引脚的转换分辨率可能会下降。详情参见 13.7 (6) 关于 AV_{DD2} 引脚。
3. 在 A/D 转换操作期间(AD2EF 位 = 1)，如果 AD2M0 和 AD2S 寄存器被写入，A/D 转换停止并重头再次开始。
4. 请确保将第 1 位至第 3 位清除为“0”。

(2) A/D 转换器 2 模式寄存器 1 (AD2M1)

AD2M1 寄存器用于指定 A/D 转换时钟的数量和 A/D 转换时间。

该寄存器可以按字节或按位进行读取或写入。

系统复位后，该寄存器被设为 00H。

复位后：00H　R/W　地址：FFFFFB81H

	7	6	5	4	3	2	1	0
AD2M1	0	0	0	0	AD2FR3	AD2FR2	AD2FR1	AD2FR0

注意事项 **1.** 关于 AD2FR3 至 AD2FR0 位，参见 表 **13-2** 转换模式期间的设置示例。

2. 在转换操作期间**(AD2CE** 位 **= 1)**，禁止改变 **AD2FR3** 至 **AD2FR0** 位。

3. 请确保将第 **4** 位至第 **7** 位清除为**"0"**。

表 **13-2.** 转换模式期间的设置示例

AD2FR3	AD2FR2	AD2FR1	AD2FR0	A/D转换的时钟数量[注]	A/D 转换时间	f_{AD2} = 32 MHz (f_{XX} = 64 MHz)	$f_{AD}2$ = 24 MHz (f_{XX} = 48 MHz)
0	0	0	1	62	62/f_{AD2}	禁止设置	禁止设置
0	0	1	0	93	93/f_{AD2}	禁止设置	3.86 μs
0	0	1	1	124	124/f_{AD2}	3.88 μs	5.17 μs
0	1	0	0	155	155/f_{AD2}	4.84 μs	6.46 μs
0	1	0	1	186	186/f_{AD2}	5.81 μs	7.75 μs
0	1	1	0	217	217/f_{AD2}	6.78 μs	9.04 μs
0	1	1	1	248	248/f_{AD2}	7.75 μs	禁止设置
1	0	0	0	279	279/f_{AD2}	8.72 μs	禁止设置
1	0	0	1	310	310/f_{AD2}	9.69 μs	禁止设置
其它情况					禁止设置		

注　时钟(f_{AD2})的数量是从 A/D 转换启动到结束。

<R>　注意事项　设置 **A/D** 转换时间范围从 **3.8** 至 **10 μs**。

备注　f_{AD2}：A/D 转换器 2 的操作时钟

(3) A/D 转换器 2 通道指定寄存器 (AD2S)

AD2S 寄存器用于指定待进行 A/D 转换的模拟输入引脚。

该寄存器可以按字节或按位进行读取或写入。

系统复位后，该寄存器被设为 00H。

复位后：00H R/W 地址：FFFFFB82H

	7	6	5	4	3	2	1	0
AD2S	0	0	0	0	0	AD2S2	AD2S1	AD2S0

AD2S2	AD2S1	AD2S0	选择模式	扫描模式
0	0	0	ANI20	ANI20
0	0	1	ANI21	ANI20, ANI21
0	1	0	ANI22	ANI20 至 ANI22
0	1	1	ANI23	ANI20 至 ANI23
1	0	0	ANI24[注]	ANI20 至 ANI24[注]
1	0	1	ANI25[注]	ANI20 至 ANI25[注]
1	1	0	ANI26[注]	ANI20 至 ANI26[注]
1	1	1	ANI27[注]	ANI20 至 ANI27[注]

注 仅限 V850E/IG3.
使用 V850E/IF3 时，该设置不可用，因为没有这些引脚。

注意事项 请确保将第 3 位至第 7 位清除为"0"。

(4) A/D2 转换结果寄存器 n, nH (AD2CRn, AD2CRnH)

AD2CRn 和 AD2CRnH 寄存器是保持 A/D 转换结果的寄存器。每次 A/D 转换结束，转换结果从逐次逼近寄存器（SAR）装载，并存储在 AD2CRn 寄存器的高 10 位。当读取时，这些寄存器的低 6 位总是 0。

A/D 转换结果的高 8 位从 AD2CRnH 寄存器读取。

这些寄存器只能按照 16 位或 8 位来读取。当按照 16 位读取 A/D 转换结果时，指定 AD2CRn 寄存器。要读取高 8 位时，指定 AD2CRnH 寄存器。

系统复位后，AD2CRn 寄存器被设为 0000H，AD2CRnH 寄存器被设为 00H。

注意事项　**如果向 AD2M0, AD2M1 和 AD2S 寄存器执行一个写入操作。AD2CRn 寄存器的内容可能变为不确定。在转换操作之后且在对 AD2M0, AD2M1 和 AD2S 寄存器执行写入操作之前，读取转换结果。如果不是上述时序，则可能无法读取到正确的转换结果。**

复位后：0000H　R　地址：AD2CR0 FFFFFB90H, AD2CR1 FFFFFB92H,
AD2CR2 FFFFFB94H, AD2CR3 FFFFFB96H,
AD2CR4 FFFFFB98H[注], AD2CR5 FFFFFB9AH[注],
AD2CR6 FFFFFB9CH[注], AD2CR7 FFFFFB9EH[注]

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AD2CRn (V850E/IF3 n = 0 至 3) (V850E/IG3 n = 0 至 7)	AD29	AD28	AD27	AD26	AD25	AD24	AD23	AD22	AD21	AD20	0	0	0	0	0	0

注　　仅限 V850E/IG3

复位后：00H　R　地址：AD2CR0H FFFFFB91H, AD2CR1H FFFFFB93H,
AD2CR2H FFFFFB95H, AD2CR3H FFFFFB97H,
AD2CR4H FFFFFB99H[注], AD2CR5H FFFFFB9BH[注],
AD2CR6H FFFFFB9DH[注], AD2CR7H FFFFFB9FH[注]

	7	6	5	4	3	2	1	0
AD2CRnH (V850E/IF3 n = 0 至 3) (V850E/IG3 n = 0 至 7)	AD29	AD28	AD27	AD26	AD25	AD24	AD23	AD22

注　　仅限 V850E/IG3

模拟输入引脚与 AD2CRn 和 AD2CRnH 寄存器的对应关系如下。

表 13-3. 模拟输入引脚与 AD2CRn 和 AD2CRnH 寄存器之间的对应

模拟输入引脚	A/D 转换结果寄存器
ANI20	AD2CR0, AD2CR0H
ANI21	AD2CR1, AD2CR1H
ANI22	AD2CR2, AD2CR2H
ANI23	AD2CR3, AD2CR3H
ANI24[注]	AD2CR4[注], AD2CR4H[注]
ANI25[注]	AD2CR5[注], AD2CR5H[注]
ANI26[注]	AD2CR6[注], AD2CR6H[注]
ANI27[注]	AD2CR7[注], AD2CR7H[注]

注　　仅限 V850E/IG3

输入到模拟输入引脚（ANI2n）的模拟电压和 A/D 转换结果（A/D2 转换结果寄存器 n（AD2CRn））之间的关系如下：

$$SAR = INT\left(\frac{V_{IN}}{AV_{DD2}} \times 1,024 + 0.5\right)$$

ADCR[注] = SAR × 64

或，

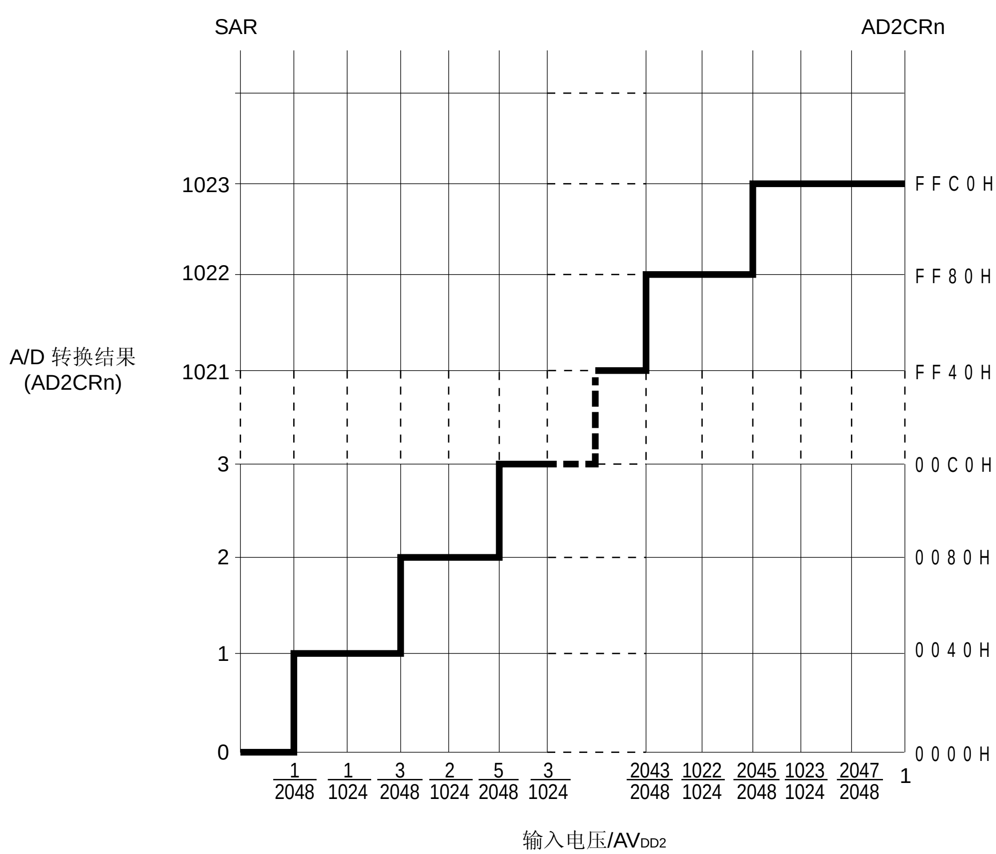
$$(SAR - 0.5) \times \frac{AV_{DD2}}{1,024} \leq V_{IN} < (SAR + 0.5) \times \frac{AV_{DD2}}{1,024}$$

INT（）： 返回（）中的数值的整数的功能
V_{IN}： 模拟输入电压
AV_{REFP}： AV_{DD2} 引脚电压
ADCR： A/D2 转换结果寄存器 n（AD2CRn）的值

注 AD2CRn 寄存器的低 6 位固定为 0。

模拟输入电压和 A/D 转换结果之间的关系如图 13-2 所示。

图 13-2. 模拟输入电压和 A/D 转换结果之间的关系



备注 V850E/IF3：n = 0 至 3
V850E/IG3：n = 0 至 7

13.4 操作

13.4.1 基本操作

<1> 当 AD2M0.AD2CE 位 = 0 时，设置 AD2M0.AD2PS 位为 1，打开 A/D 电源。
此时，除 AD2M0.AD2CE 位之外的位可以同时设置。

<2> 通过 AD2M0, AD2M1 和 AD2S 寄存器选择 A/D 转换的操作模式和 A/D 转换时间。

<3> 在打开 A/D 电源(AD2M0.AD2PS 位 = 0 → 1) 后至少 2 μs，设置 AD2M0.AD2CE 位为 1 (使能转换)来启动 A/D 转换。
如果在 2 μs 经过之前，设置 AD2CE 位，转换操作被启动，并在 A/D 转换时间之后结束，但是转换结果不确定。

<4> 当 A/D 转换被启动时，输入到选择的模拟输入通道的电压被采样&保持电路采样。

<5> 当采样已经执行了一段指定时间，采样&保持电路进入保持状态，并保持输入模拟电压直到 A/D 转换结束。

<6> 设置逐次逼近寄存器（SAR）的第 9 位，并改变 D/A 转换器的电压档位电平作为参考电压（1/2AV_{DD2}）。

<7> 通过一个比较器，对 D/A 转换器的电压档位产生的电压和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（1/2AV_{DD2}），SAR 寄存器的 MSB 保持置位。如果模拟输入电压小于（1/2AV_{DD2}），SAR 的 MSB 被重置。

<8> 接下来，SAR 寄存器的第 8 位被自动置位，启动下一次比较。D/A 转换器的电压档位根据第 9 位的值选择，上次的比较结果已经被设置如下。

第 9 位 = 1：(3/4 AV_{DD2})
第 9 位 = 0：(1/4 AV_{DD2})

D/A 转换器的电压档位器和模拟输入电压进行比较，根据比较结果，操作 SAR 寄存器的第 8 位。

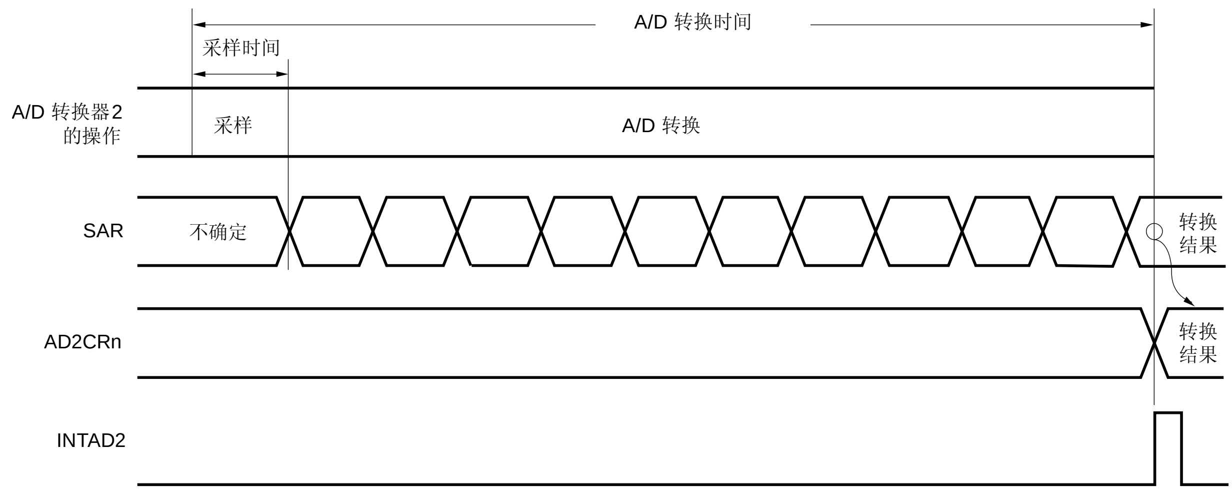
模拟输入电压 ≥ D/A 转换器的电压档位器：第 8 位 = 1
模拟输入电压 ≤ D/A 转换器的电压档位器：第 8 位 = 0

如此继续比较，直到 SAR 的第 0 位。

<9> 当 10 位的比较完成后，有效数字值结果保留在 SAR 寄存器内。该值被传送到 AD2CRn 寄存器且转换结果被存储在该寄存器(V850E/IF3：n = 0 至 3, V850E/IG3：n = 0 至 7)。在选择模式下的同时，或当扫描模式下所有指定的 A/D 转换操作都已经完成，会产生一个 A/D2 转换结束中断请求信号（INTAD2）。

<10> 在连续选择模式或连续扫描模式下，<4>至<9>步骤重复，直到 A/D 转换完成之后 AD2CE 位被设置为 0。
在单次选择模式或扫描模式下，完成之后，转换操作停止(此时，AD2M0.AD2CE 位保持为 1 且不被自动清除)。向 AD2CE 位写入 1 来再次启动转换操作。

图 13-3. A/D 转换器 2 的基本操作



备注 V850E/IF3：n = 0 至 3
V850E/IG3：n = 0 至 7

13.4.2 触发模式

用作 A/D 转换操作的启动时序的触发模式是软件触发模式。
该模式通过 AD2M0 寄存器来设置。

(1) 软件触发模式

在该模式下，通过将 AD2M0.AD2CE 位设置为 1，由 AD2S.AD2S2 至 AD2S.AD2S0 位指定的模拟输入引脚 (ANI2n)被用于 A/D 转换启动时序。

A/D 转换结束之后，转换结果存储在 A/D2 转换结果寄存器 n (AD2CRn)。在选择模式下的同时，或当扫描模式下所有指定的 A/D 转换操作都已经完成，会产生 A/D2 转换结束中断请求信号（INTAD2）。

如果由 AD2M0.AD2MD1 和 AD2M0.AD2MD0 位设置的操作模式是连续选择模式或连续扫描模式，转换操作重复，直到 AD2M0.AD2CE 位被设置为 0。在单次选择模式或扫描模式下，A/D 转换结束之后，转换操作停止。

当 A/D 转换被启动时，AD2M0.AD2EF 位被设置为 1(转换进行中)，当转换完成时被设置为 0 (转换停止)。

在 A/D 转换期间，如果 AD2M0 和 AD2S 寄存器被写入，转换被停止，并重头再次执行。

备注 V850E/IF3：n = 0 至 3
V850E/IG3：n = 0 至 7

13.4.3 操作模式

ANI2n 引脚可以设置四种操作模式：连续选择模式，连续扫描模式，单次选择模式和单次扫描模式。这些模式通过 AD2M0.AD2MD1 和 AD2M0.AD2MD0 寄存器来设置。

AD2M0, AD2M1 和 AD2S 寄存器与操作模式之间的关系如下。

触发模式	操作模式	设置值		
		AD2M0	AD2M1	AD2S
软件触发	连续选择	X100000XB	0000XXXXB	00000XXXB
	连续扫描	X101000XB	0000XXXXB	00000XXXB
	单次选择	X110000XB	0000XXXXB	00000XXXB
	单次扫描	X111000XB	0000XXXXB	00000XXXB

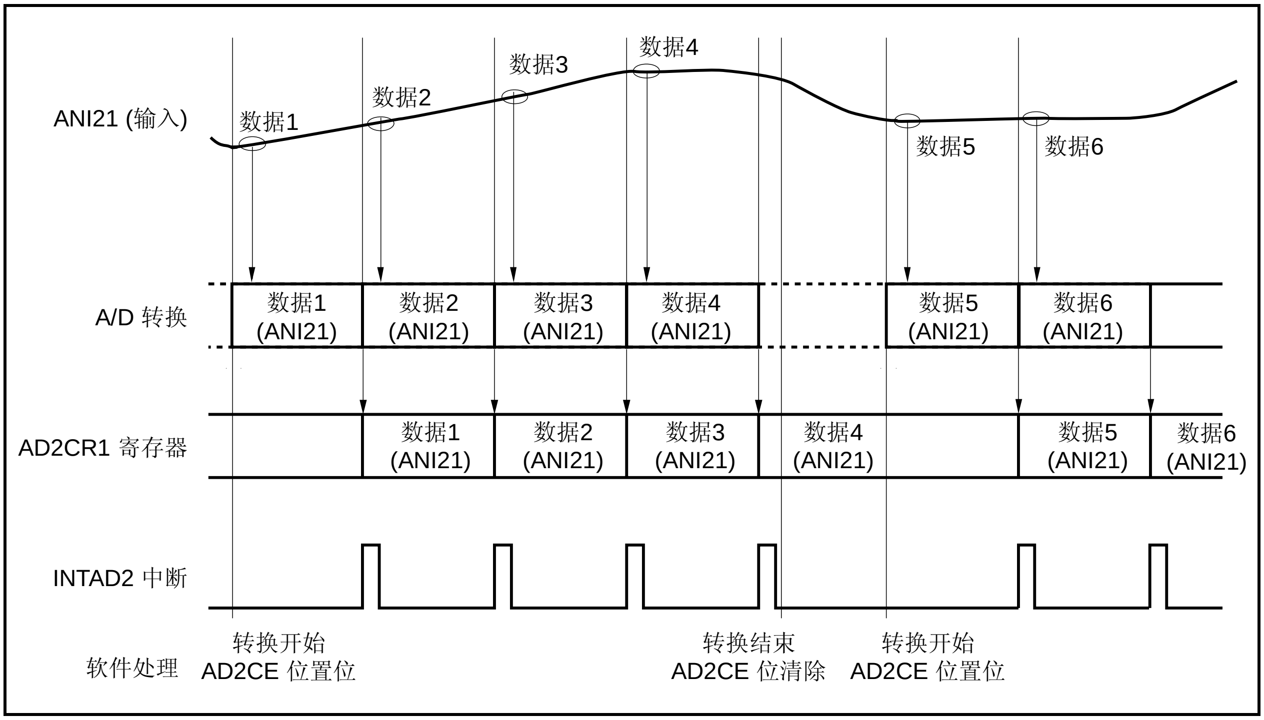
(1) 连续选择模式

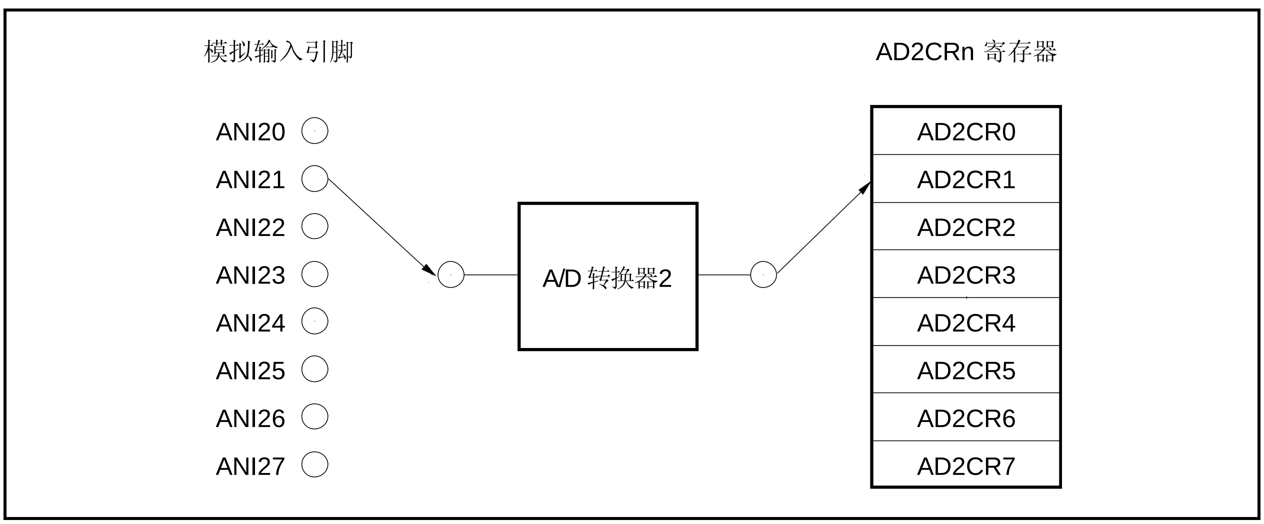
在该模式下，AD2S 寄存器指定的模拟输入引脚 (ANI2n)是连续进行 A/D 转换。转换结果存储在 ANI2n 引脚对应的 AD2CRn 寄存器。ANI2n 引脚和 AD2CRn 寄存器是一一对应的，每次一个 A/D 转换结束，则产生一个 A/D2 转换结束中断请求信号(INTAD2)。

在 A/D 转换结束之后，转换再次重复，直到 AD2M0.AD2CE 位被设置为 0。

备注　V850E/IF3：n = 0 至 3
　　　V850E/IG3：n = 0 至 7

图 13-4. 连续选择模式操作时序
(当 AD2M0.AD2MD1 和 AD2M0.AD2MD0 位 = 00，AD2S.AD2S2 至 AD2S.AD2S0 位 = 001)：V850E/IG3





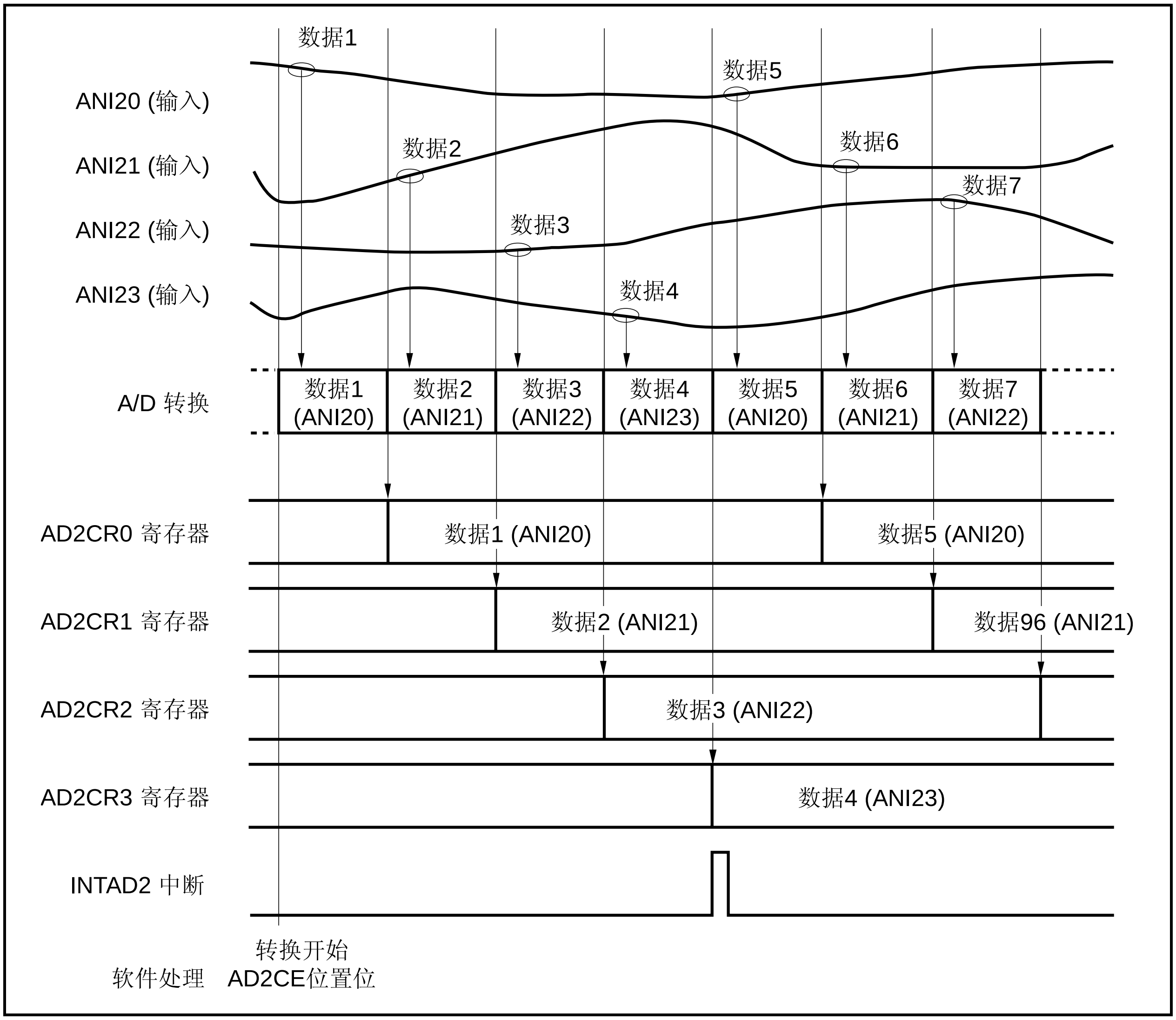
(2) 连续扫描模式

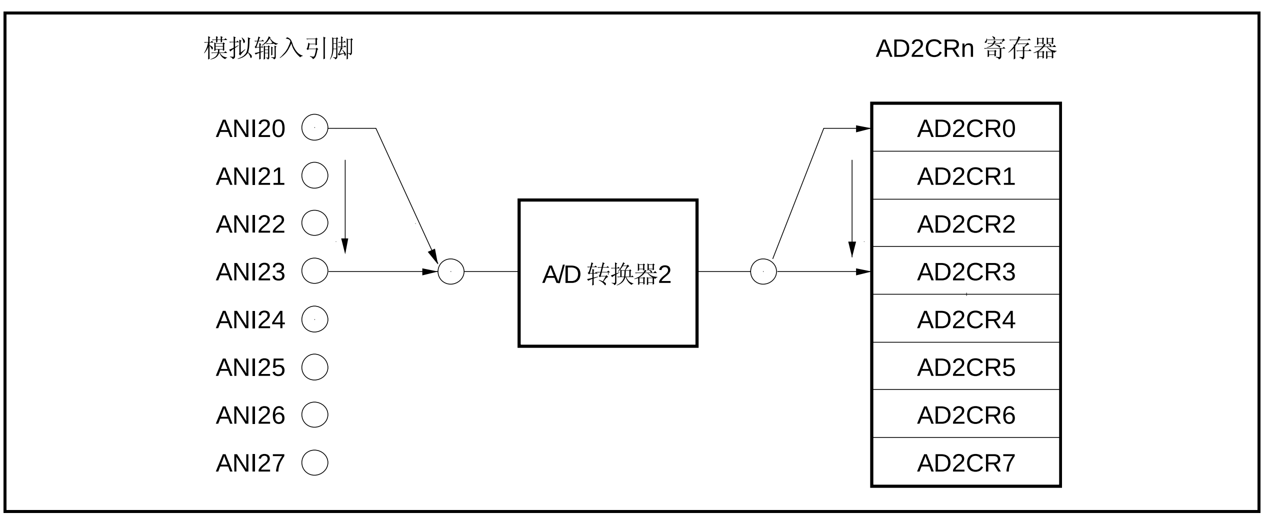
在该模式下，从 ANI20 引脚至 AD2S 寄存器指定的模拟输入引脚(ANI2n)被顺序选择，连续执行 A/D 转换。A/D 转换结果存储在模拟输入引脚对应的 AD2CRn 寄存器。当所有指定的模拟输入引脚的转换结束时，产生 A/D2 转换结束中断请求信号(INTAD2)。在 A/D 转换结束之后，转换被再次从 ANI20 引脚启动，直到 AD2M0.AD2CE 位被设置为 0。

备注 V850E/IF3：n = 0 至 3
V850E/IG3：n = 0 至 7

图 13-5. 连续扫描模式操作时序

(当 AD2M0.AD2MD1 和 AD2M0.AD2MD0 位 = 01，AD2S.AD2S2 至 AD2S.AD2S0 位 = 011)：V850E/IG3





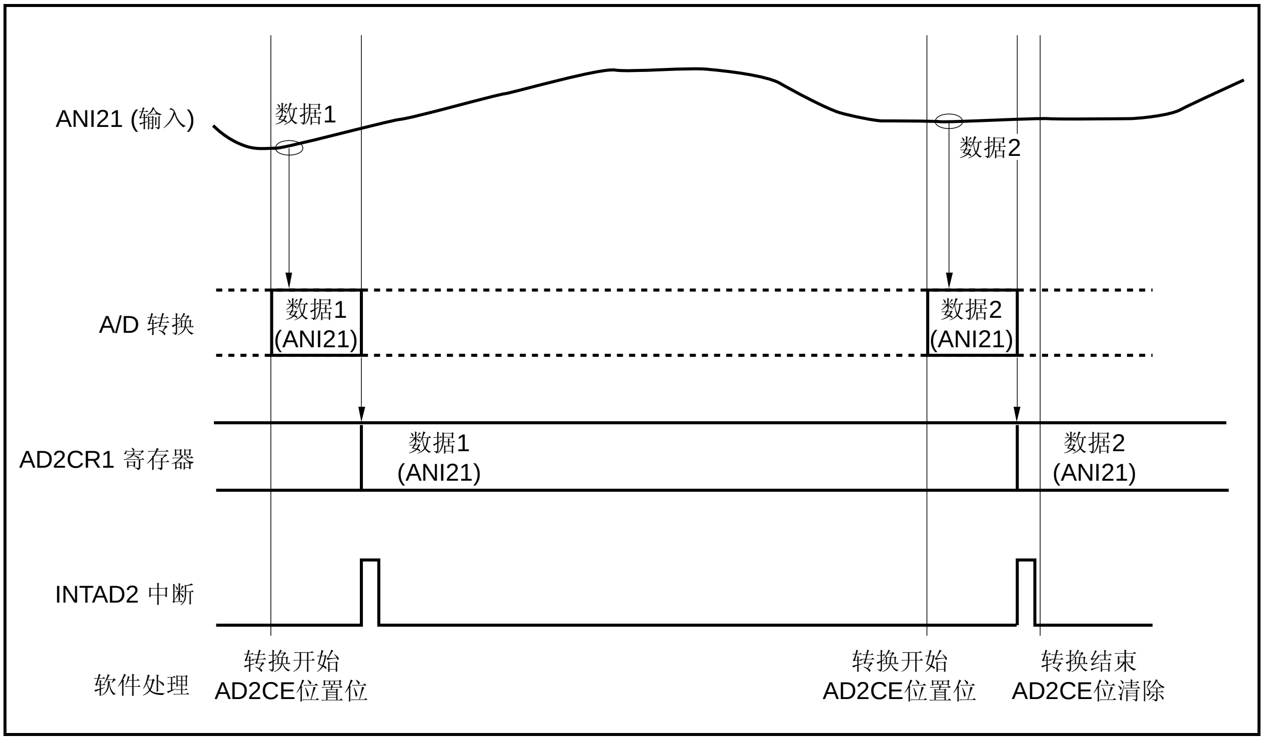
(3) 单次选择模式

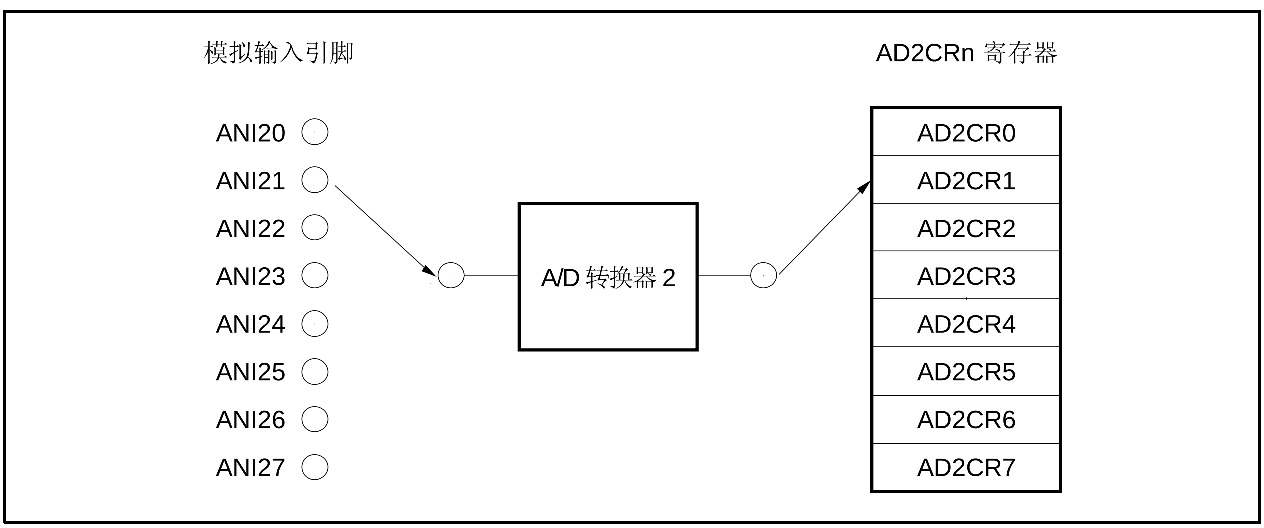
在该模式下，AD2S 寄存器指定的模拟输入引脚 (ANI2n)进行一次 A/D 转换。转换结果存储在 ANI2n 引脚对应的 AD2CRn 寄存器。ANI2n 引脚和 AD2CRn 寄存器是一一对应的，每次一个 A/D 转换结束，则产生一个 A/D2 转换结束中断请求信号(INTAD2)。

在 A/D 转换结束之后，转换操作停止。

备注　V850E/IF3：n = 0 至 3
V850E/IG3：n = 0 至 7

图 13-6. 单次选择模式操作时序
(当 AD2M0.AD2MD1 和 AD2M0.AD2MD0 位 = 10，AD2S.AD2S2 至 AD2S.AD2S0 位 = 001)：V850E/IG3





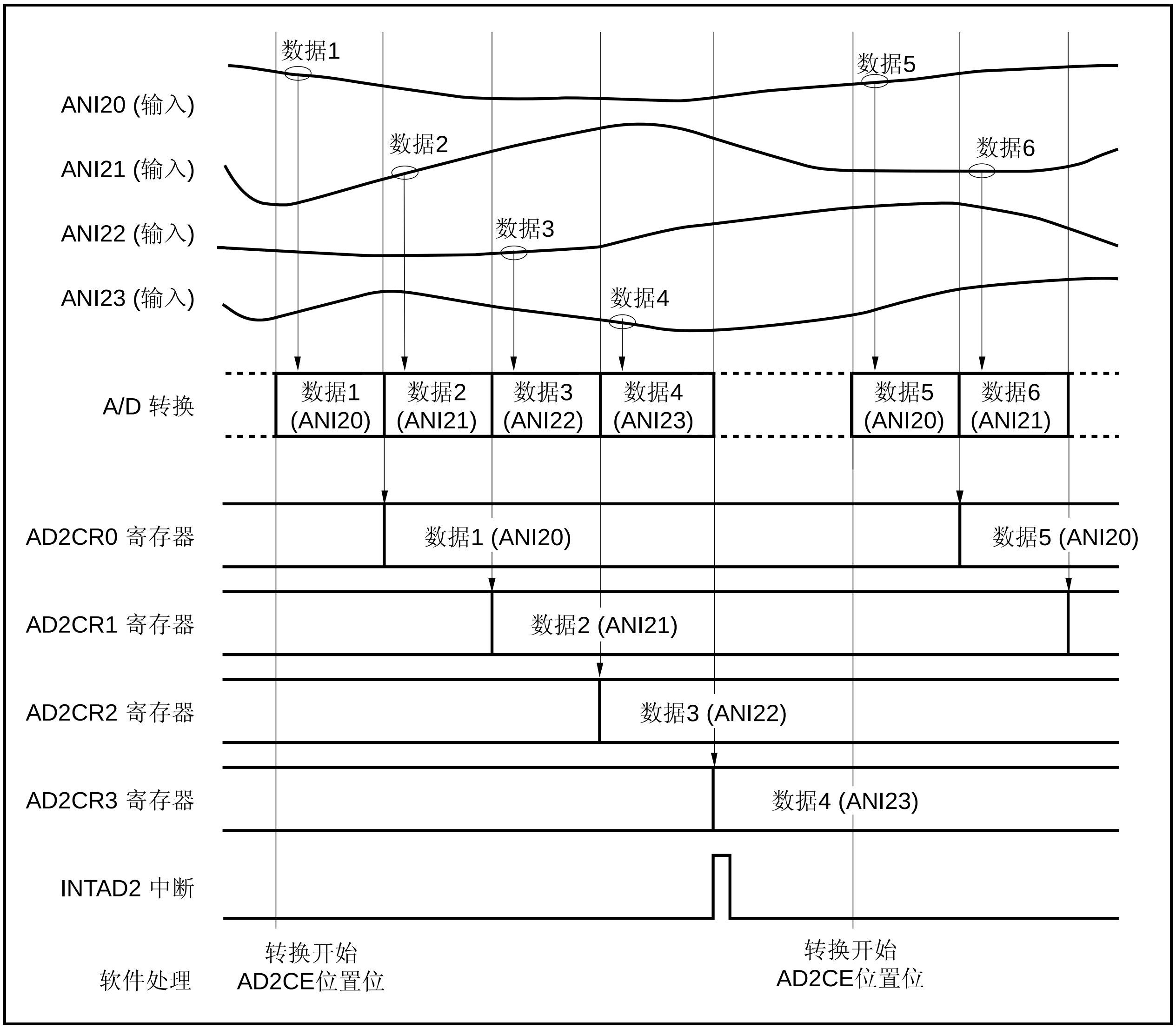
(4) 单次扫描模式

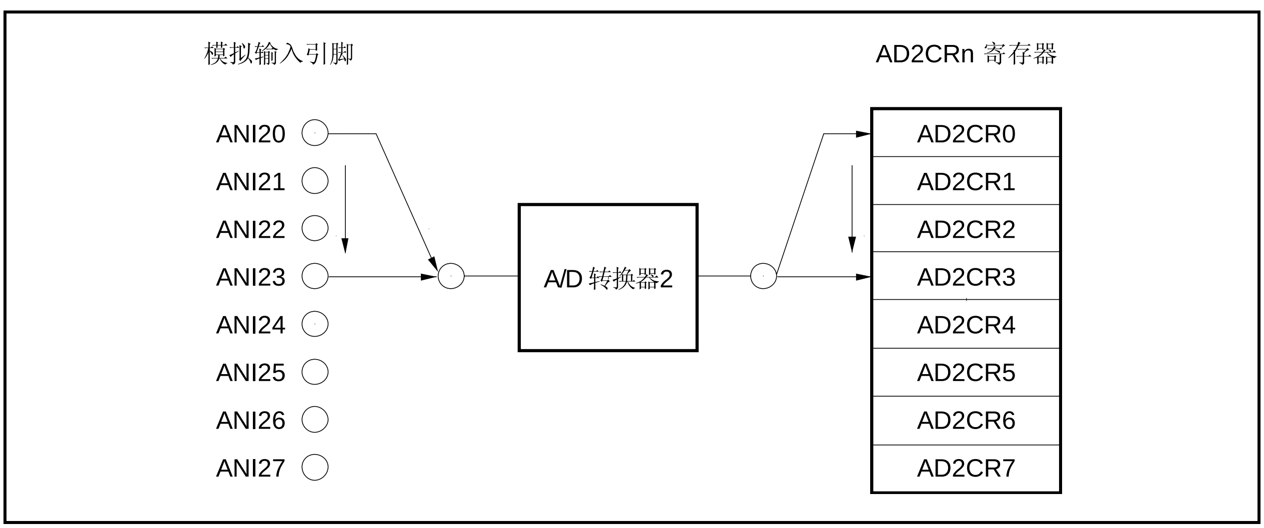
在该模式下，从 ANI20 引脚至 AD2S 寄存器指定的模拟输入引脚(ANI2n)被顺序选择，执行 A/D 转换。A/D 转换结果存储在模拟输入引脚对应的 AD2CRn 寄存器。当所有指定的模拟输入引脚的转换结束时，产生 A/D2 转换结束中断请求信号(INTAD2)。在 A/D 转换结束之后，转换操作停止。

备注　V850E/IF3：n = 0 至 3
V850E/IG3：n = 0 至 7

图 13-7. 扫描模式操作时序

(当 AD2M0.AD2MD1 和 AD2M0.AD2MD0 位 = 11，AD2S.AD2S2 至 AD2S.AD2S0 位 = 011)：V850E/IG3





13.5 软件触发模式的操作

当 AD2M0.AD2CE 位被设置为 1 时，A/D 转换启动。
当 A/D 转换被启动时，AD2M0.AD2EF 位 = 1 (转换进行中)。
在 A/D 转换期间，如果 AD2M0 和 AD2S 寄存器被写入，转换停止并重头再次执行。

(1) 软件触发连续选择模式的操作

在该模式下，AD2S 寄存器指定的模拟输入引脚 (ANI2n)进行一次 A/D 转换。转换结果存储在 ANI2n 引脚对应的 AD2CRn 寄存器。ANI2n 引脚和 AD2CRn 寄存器是一一对应的。
每次一个 A/D 转换结束，则产生一个 A/D2 转换结束中断请求信号(INTAD2)，且 A/D 转换结束。
在 A/D 转换结束之后，转换再次重复，直到 AD2M0.AD2CE 位被设置为 0。
要重启A/D转换[注]，无需将AD2M0.AD2CE位置位(1)。

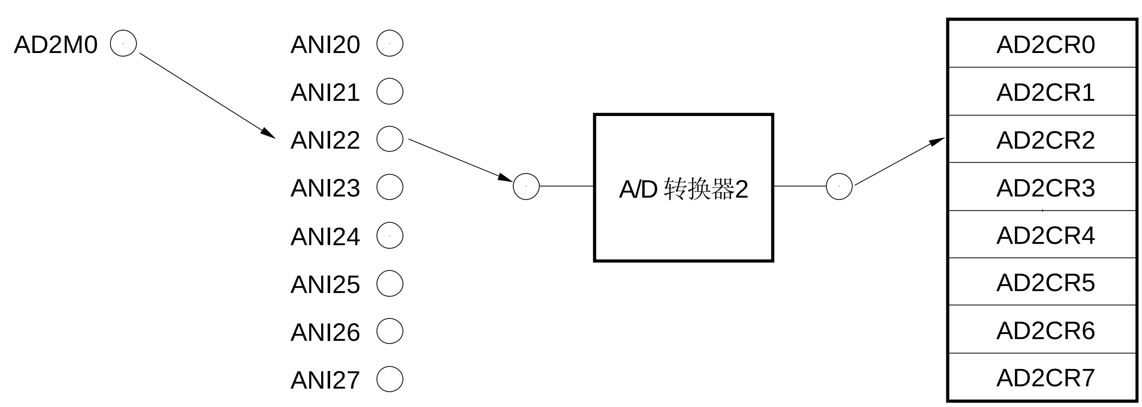
注　在软件触发连续选择模式下，A/D 转换操作不停止，除非 AD2M0.AD2CE 位被设置为 0。如果在下一次 A/D 转换结束之前，AD2CRn 寄存器没有被读取，则会被覆盖。

该模式适用于读取一个模拟输入引脚的 A/D 转换值的应用。

模拟输入引脚	A/D 转换结果寄存器
ANI2n	AD2CRn

备注　V850E/IF3：n = 0 至 3
V850E/IG3：n = 0 至 7

图 13-8. 软件触发连续选择模式的操作示例：V850E/IG3



(1) AD2CE 位 = 1 (使能)
(2) ANI22 引脚进行 A/D 转换
(3) 转换结果存储在 AD2CR2 寄存器
(4) AD2M0.AD2EF 位 = 0
(5) 产生 INTAD2 中断请求信号
(6) 返回到(2)
(7) 要结束转换，AD2CE 位 = 0 (停止)

备注　这个操作示例的设置如下。
AD2M0.AD2MD1 和 AD2M0.AD2MD0 位 = 00，AD2S.AD2S2 至 AD2S.AD2S0 位 = 010

(2) 软件触发连续扫描模式操作

在该模式下，从 ANI20 引脚至 AD2S 寄存器指定的模拟输入引脚(ANI2n)被顺序选择，连续执行 A/D 转换。A/D 转换结果存储在模拟输入引脚对应的 AD2CRn 寄存器。

当所有指定的模拟输入引脚的转换结束时，产生 A/D2 转换结束中断请求信号(INTAD2)。在 A/D 转换结束之后，转换被再次从 ANI20 引脚启动，直到 AD2M0.AD2CE 位被设置为 0。

要重启A/D转换[注]，无需将AD2M0.AD2CE位置位(1)。

注　在软件触发连续扫描模式下，A/D 转换操作不会停止，除非 AD2M0.AD2CE 位被设置为 0。如果在下一次 A/D 转换结束之前，AD2CRn 寄存器没有被读取，则会被覆盖。

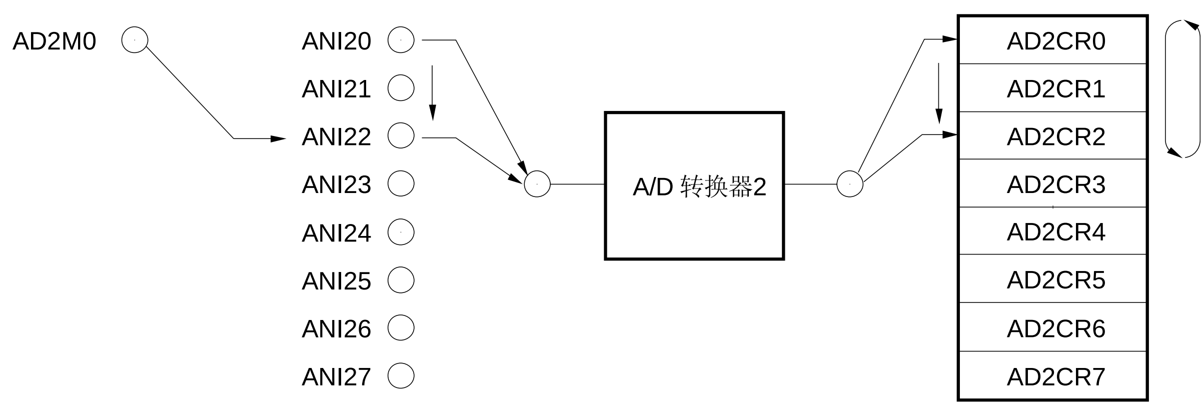
该模式适用于持续监视多个模拟输入的应用。

模拟输入引脚	A/D 转换结果寄存器
ANI20	AD2CR0
⋮	⋮
ANI2n[注]	AD2CRn

注　通过 AD2S.AD2S0 至 AD2S.AD2S2 位设置。

备注　V850E/IF3：n = 0 至 3
　　　V850E/IG3：n = 0 至 7

图 13-9. 软件触发连续扫描模式的操作示例：V850E/IG3



(1) AD2CE 位 = 1 (使能)
(2) ANI20 引脚进行 A/D 转换
(3) 转换结果存储在 AD2CR0 寄存器
(4) ANI21 引脚进行 A/D 转换
(5) 转换结果存储在 AD2CR1 寄存器
(6) ANI22 引脚进行 A/D 转换
(7) 转换结果存储在 AD2CR2 寄存器
(8) 产生 INTAD2 中断请求信号
(9) 返回到(2)
(10) 要结束转换，AD2CE 位 = 0 (停止)

备注　这个操作示例的设置如下。
AD2M0.AD2MD1 和 AD2M0.AD2MD0 位 = 01，AD2S.AD2S2 至 AD2S.AD2S0 位 = 010

(3) 软件触发单次选择模式

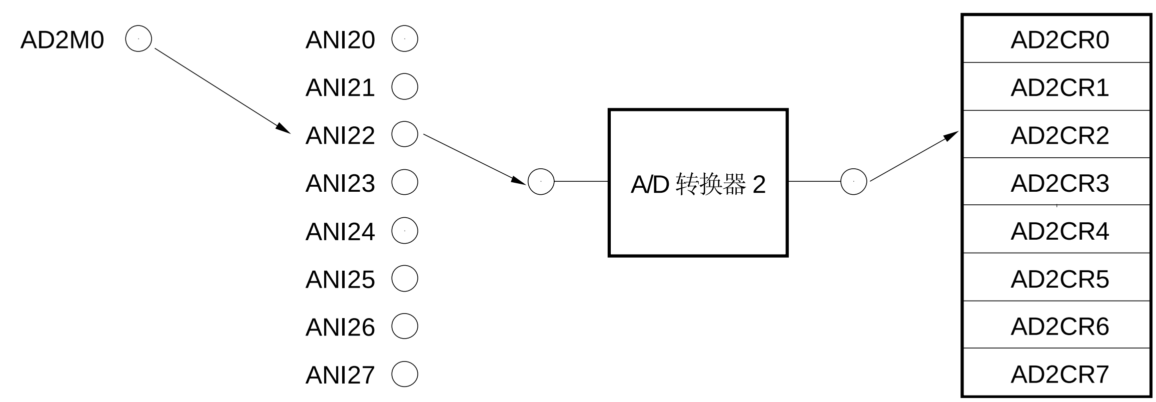
在该模式下，AD2S 寄存器指定的模拟输入引脚 (ANI2n)进行一次 A/D 转换。转换结果存储在 ANI2n 引脚对应的 AD2CRn 寄存器。ANI2n 引脚和 AD2CRn 寄存器是一一对应的。
每次一个 A/D 转换结束，则产生一个 A/D2 转换结束中断请求信号(INTAD2)，且 A/D 转换结束。
在 A/D 转换结束之后，转换操作停止。
如果 AD2M0.AD2CE 位被设置为 1，可以重启 A/D 转换。

该模式适用于读取每个第一次 A/D 转换结果的应用。

模拟输入引脚	A/D 转换结果寄存器
ANI2n	AD2CRn

备注 V850E/IF3：n = 0 至 3
V850E/IG3：n = 0 至 7

图 13-10. 软件触发单次选择模式的操作示例：V850E/IG3



(1) AD2CE 位 = 1 (使能)
(2) ANI22 引脚进行 A/D 转换
(3) 转换结果存储在 AD2CR2 寄存器
(4) AD2M0.AD2EF 位 = 0
(5) 产生 INTAD2 中断请求信号

备注 这个操作示例的设置如下。
AD2M0.AD2MD1 和 AD2M0.AD2MD0 位 = 10，AD2S.AD2S2 至 AD2S.AD2S0 位 = 010

(4) 软件触发单次扫描模式操作

在该模式下，从 ANI20 引脚至 AD2S 寄存器指定的模拟输入引脚(ANI2n)被顺序选择，连续执行 A/D 转换。A/D 转换结果存储在模拟输入引脚对应的 AD2CRn 寄存器。

当所有指定的模拟输入引脚的转换结束时，产生 A/D2 转换结束中断请求信号(INTAD2)。

在 A/D 转换结束之后，转换操作停止。

如果 AD2M0.AD2CE 位被设置为 1，可以重启 A/D 转换。

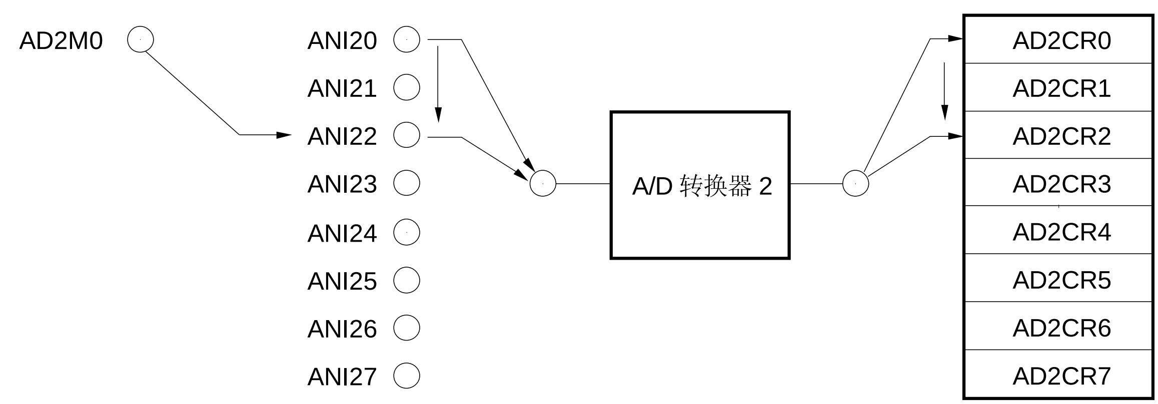
该模式适用于持续监视多个模拟输入的应用。

模拟输入引脚	A/D 转换结果寄存器
ANI20	AD2CR0
⋮	⋮
ANI2n[注]	AD2CRn

注　　通过 AD2S.AD2S0 至 AD2S.AD2S2 位设置。

备注　　V850E/IF3：n = 0 至 3
　　　　V850E/IG3：n = 0 至 7

图 13-11. 软件触发扫描模式的操作示例：V850E/IG3

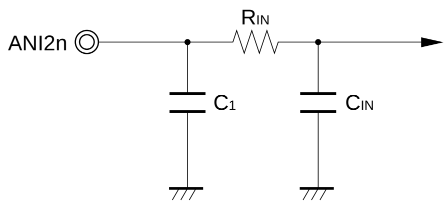


(1) AD2CE 位 = 1 (使能)
(2) ANI20 引脚进行 A/D 转换
(3) 转换结果存储在 AD2CR0 寄存器
(4) ANI21 引脚进行 A/D 转换
(5) 转换结果存储在 AD2CR1 寄存器
(6) ANI22 引脚进行 A/D 转换
(7) 转换结果存储在 AD2CR2 寄存器
(8) AD2M0.AD2EF 位 = 0
(9) 产生 INTAD2 中断请求信号

备注　　这个操作示例的设置如下。
AD2M0.AD2MD1 和 AD2M0.AD2MD0 位 = 11，AD2S.AD2S2 至 AD2S.AD2S0 位 = 010

13.6 内部等效电路

下图展示了模拟输入模块的等效电路。



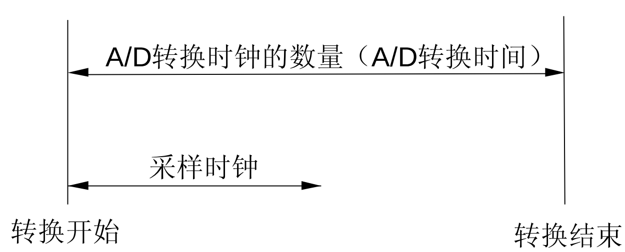
R	C1	C2
2.6 kΩ	15 pF	6.2 pF

备注 **1.** 最大值如下（参考值）。

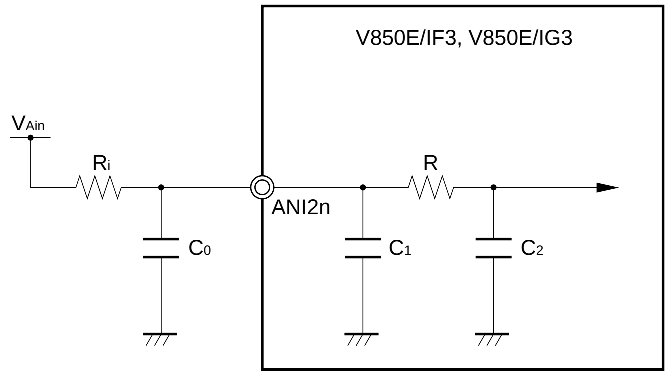
2. V850E/IF3：n = 0 至 3
V850E/IG3：n = 0 至 7

AD2M1 寄存器				A/D 转换时钟的数量 (f_{AD2})	采样时钟的数量
AD2FR3 位	AD2FR2 位	AD2FR1 位	AD2FR0 位		
0	0	0	1	62	33
0	0	1	0	93	49.5
0	0	1	1	124	66
0	1	0	0	155	82.5
0	1	0	1	186	99
0	1	1	0	217	115.5
0	1	1	1	248	132
1	0	0	0	279	148.5
1	0	0	1	310	165

注意事项 采样时钟的数量包括在 A/D 转换时钟的数量内。



计算 A/D 转换器 2 的总误差的示例如下。



f_{XX} (MHz)	A/D 转换时间 (μs)	采样 (μs)	R (kΩ)	C_1 (pF)	C_2 (pF)	C_0 (pF)	R_i (kΩ)	采样错误(LSB)[注]
64	3.88 ($62/f_{AD2}$)	1.03 ($33/f_{AD2}$)	2.6	15	6.2	100	1.0	0.1 或更低
						100	0.5	0.1 或更低
						100	0.25	0.1 或更低
						100	0.125	0.1 或更低
						50	1.0	0.1 或更低
						50	0.5	0.1 或更低
						50	0.25	0.1 或更低
						50	0.125	0.1 或更低

注　考虑信号源阻抗时，错误是"采样误差 + 总误差"。

备注
1. 这些值是通过计算得到的参考值，当 V_{Ain} 输入 0 V 至 5 V 的同时启动采样，模拟 R_i 和 C_0 对 C_2 电压的影响。
2. V850E/IF3：n = 0 至 3
V850E/IG3：n = 0 至 7
3. f_{XX}：系统时钟频率
f_{AD2}：操作时钟频率

13.7 注意事项

(1) 不使用 **A/D** 转换器时

不使用 A/D 转换器时，将 AD2M0.AD2CE 和 AD2M0.AD2PS 位清除为 0，可以降低功耗。

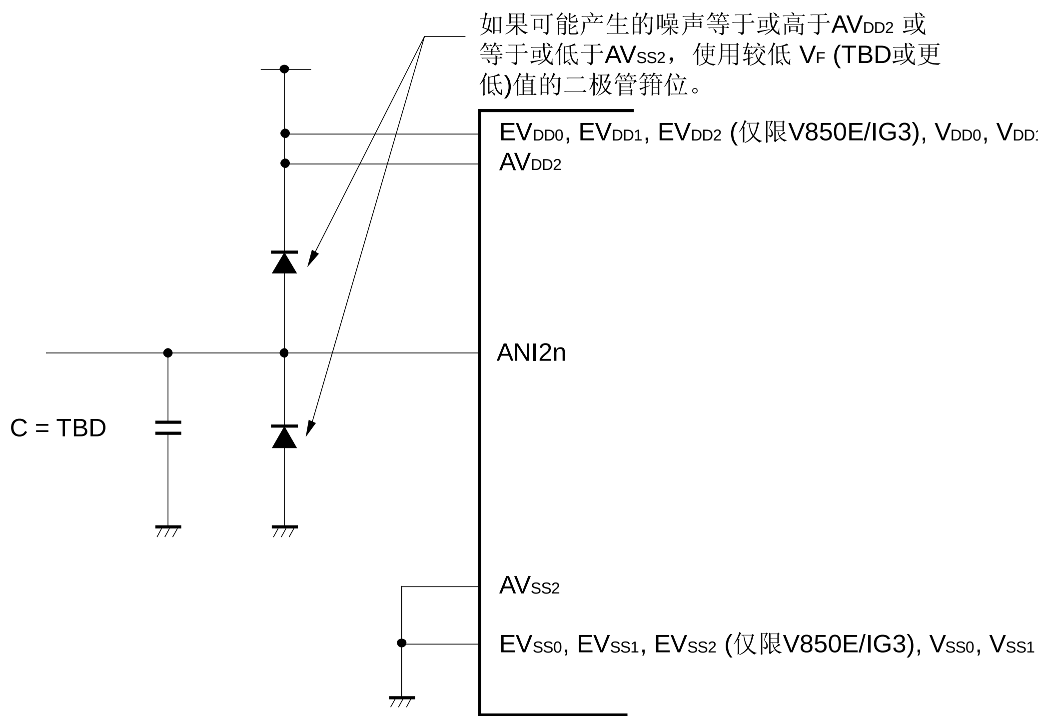
(2) ANI2n 引脚的输入范围

输入到 ANI2n 引脚的电压应该在指定范围内。如果一个大于或等于 AV_{DD2} 的电压，或者低于或等于 AV_{SS2} 的电压（即使在绝对最大额度范围内）被输入到该引脚，该通道的转换值不确定，并且其它通道的转换值可能也会受到影响。

(3) 对抗噪声的措施

要保持 10 位分辨率，必须有效保护 ANI2n 引脚以防止噪声。随着模拟输入源的输出阻抗变高，噪声的影响增加。要降低噪声，推荐连接一个外部电容，如图 13-12 所示。

图 13-12. 模拟输入引脚的处理



备注 V850E/IF3：n = 0 至 3
V850E/IG3：n = 0 至 7

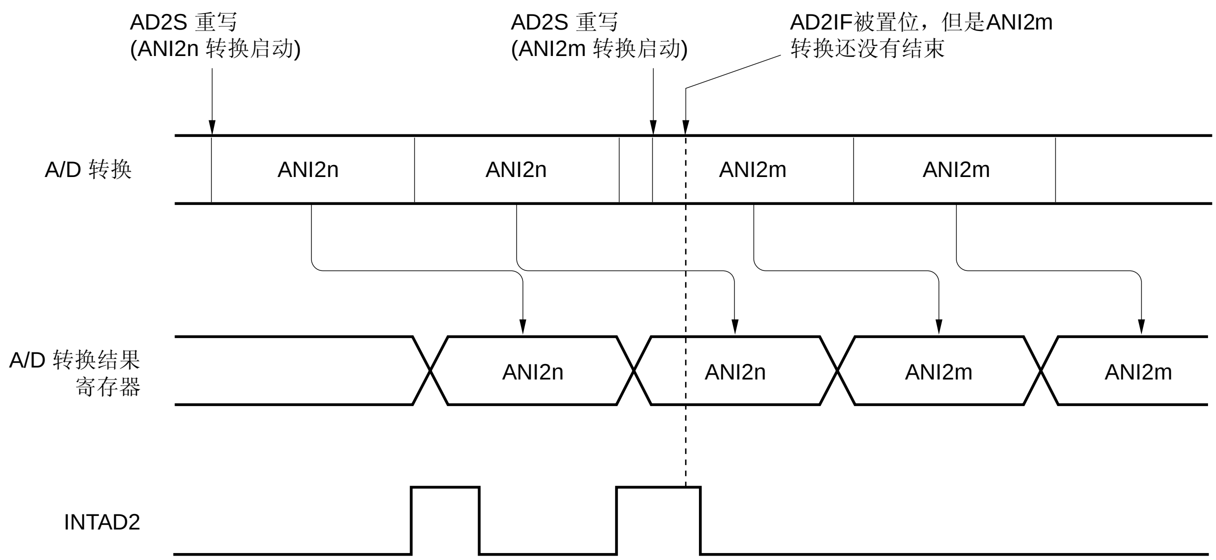
(4) 复用输入

模拟输入引脚 (ANI2n)功能复用作为输入端口(P7n)。当 ANI2n 引脚的其中一个被选作 A/D 转换时，在转换期间，不要执行端口 7 的输入指令，否则转换的分辨率会下降。

(5) 中断请求信号(AD2IF)

即使 AD2S 寄存器的内容被改变，中断请求信号(AD2IF)也不会被清除。在 A/D 转换期间，如果模拟输入引脚被改变，因此，上次选择的模拟输入信号转换结果可能被存储，且在改写 AD2S 寄存器之前 A/D2 转换结束中断请求信号有可能立刻被设置。如果在改写 AD2S 寄存器之后立即读取 AD2IF 标志，即使新选择的模拟输入引脚的 A/D 转换尚未完成，AD2IF 标志也可能被置位。当 A/D 转换停止时，恢复转换之前，清除 AD2IF 标志。

图 13-13. A/D2 转换结束中断请求的产生时序



备注 V850E/IF3：n = 0 至 3, m= 0 至 3
V850E/IG3：n = 0 至 7, m = 0 至 7

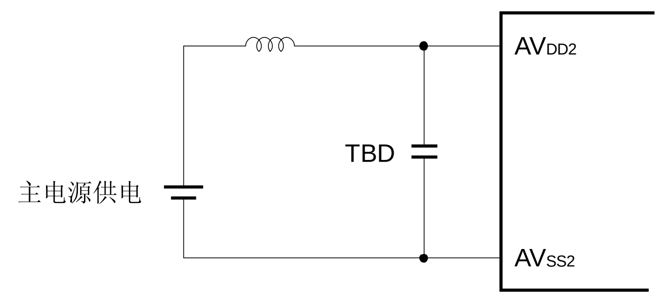
(6) AV_{DD2} 引脚

(a) AV_{DD2} 引脚用作 A/D 转换器 2 的供电电源引脚，也为复用功能端口提供电源。在使用备用供电电源的应用中，请确保向 AV_{DD2} 引脚提供的电势与 EV_{DD0}, EV_{DD1} 和 EV_{DD2} (仅限 V850E/IG3)的电势相同，如图 13-12 所示。

(b) AV_{DD2} 引脚也用作 A/D 转换器 2 的参考电压引脚。如果向 AV_{DD2} 引脚提供的供电电源是高阻抗，或提供的供电电源其电流提供能力较低，参考电压可能会波动，由于转换期间流经电流的影响(尤其在转换操作刚刚使能(AD2CE 位 = 1)之后)，转换精度可能会下降。为了防止这个问题，推荐在 AV_{DD2} 和 AV_{SS2} 引脚之间连接一个电容，用于抑制参考电压的波动，如图 13-14 所示。

(c) 如果向 AV_{DD2} 引脚提供的供电电源有较高的 DC 阻抗(比如，因为二极管的接入)，转换使能时的电压可能低于转换停止时的电压，因为 A/D 转换电流导致电压下降。

图 13-14. AV_{DD2} 引脚连接示例



(7) 读取 AD2CRn 寄存器

当 AD2M0, AD2M1 或 AD2S 寄存器被写入时，AD2CRn 寄存器的内容可能不确定。在转换完成之后且写入 AD2M0, AD2M1 和 AD2S 寄存器之前，读取转换结果。如果不同于上述时序，可能无法读取正确的转换结果。

(8) A/D 转换结果

如果在模拟输入引脚（ANI2n）或在 A/D 转换器参考电压输入引脚（AVDD2）有噪声，噪声可能会产生错误的转换结果。

需要软件处理来防止非法转换结果对系统的负面影响。

下面是一个软件处理的示例。

- 对一定数量的 A/D 转换取平均值，并将该值用作 A/D 转换结果.
- 连续执行一定数量的 A/D 转换，并使用这些结果，忽略任何可能获得的意外结果。
- 如果获得的一个 A/D 转换结果被判定已经产生系统误操作，请确保在执行相应措施之前，重新检查系统错误。

(9) 待机模式

因为在 IDLE 和 STOP 模式下，A/D 转换器 2 停止操作，转换结果无效，因此可以降低功耗。在解除 IDLE 和 STOP 模式之后，操作恢复，但是解除 IDLE 和 STOP 模式之后的 A/D 转换结果无效。在 IDLE 和 STOP 模式被解除之后使用 A/D 转换器 2 时，在设置 IDLE 和 STOP 模式之前或解除 IDLE 和 STOP 模式之前，将 AD2M0.AD2CE 位设置为 0，在 IDLE 和 STOP 模式解除之后，将 AD2CE 位设置为 1。

(10) A/D 转换结果的变化

A/D 转换的结果可能会因为供电电压的波动而变化，或者可能会受到噪声的影响。要减少这些影响，在程序中采取相应措施，比如取 A/D 转换结果的平均值。

(11) A/D 转换结果迟滞特性

<R> 连续比较型 A/D 转换器在内部采样&保持电容中保持模拟输入电压，然后执行 A/D 转换。在 A/D 转换完成之后，模拟输入电压保持在内部采样&保持电容中。如果来自模拟输入源的输出阻抗太高，可能会发生下列现象。

- 当 A/D 转换使用相同的通道时，如果电压比上次 A/D 转换更高或更低，转换结果受上次值的影响的话，可能会出现迟滞特性。即使转换执行时是相同的电势，结果可能不同。
- 当切换模拟输入通道时，转换结果受上个通道的值的影响的话，可能会出现迟滞特性。这是因为 A/D 转换使用同一个 A/D 转换器。即使转换执行时是相同的电势，结果可能不同。

<R> 要获得更准确的转换结果，降低来自模拟输入源的输出阻抗或在同一通道上连续执行两次 A/D 转换，并丢弃第一次的转换结果。

13.8 如何读取 A/D 转换器特征表

关于 A/D 转换器特征表的详情，参见 **12.7 如何读取 A/D 转换器特征表**。

14.1 UARTA 和其它串行接口的模式转换

14.1.1 UARTA0 和 CSIB0 的模式转换

在 V850E/IF3 和 V850E/IG3 中，UARTA0 和 CSIB0 是相同引脚的复用功能，因而不能同时使用。要在 UARTA0 和 CSIB0 之间切换，用 PMC4、PFC4 和 PFCE4 寄存器预先设置。

注意事项 **如果在发送或接收期间切换这些功能，则 UARTA0 或 CSIB0 的发送和接收操作得不到保证。确保禁止未使用的功能。**

图 14-1. UARTA0 和 CSIB0 模式转换设置

复位后： 00H　R/W　地址： FFFFF448H

	7	6	5	4	3	2	1	0
PMC4	PMC47	PMC46	PMC45	PMC44	PMC43	PMC42	PMC41	PMC40

复位后： 00H　R/W　地址： FFFFF468H

	7	6	5	4	3	2	1	0
PFC4	PFC47	PFC46	PFC45	PFC44	PFC43	PFC42	PFC41	PFC40

复位后： 00H　R/W　地址： FFFFF708H

	7	6	5	4	3	2	1	0
PFCE4	PFCE47	PFCE46	PFCE45	PFCE44	PFCE43	PFCE42	PFCE41	PFCE40

PMC42	PFCE42	PFC42	P42 引脚复用功能说明
0	×	×	I/O 端口
1	0	0	$\overline{\text{SCKB0}}$ I/O
1	0	1	INTP13 输入
1	1	0	禁止设置
1	1	1	禁止设置

PMC41	PFCE41	PFC41	P41 引脚复用功能说明
0	×	×	I/O 端口
1	0	0	SOB0 输出
1	0	1	TXDA0 输出
1	1	0	禁止设置
1	1	1	禁止设置

PMC40	PFC40	P40 引脚复用功能说明
0	×	I/O 端口
1	0	SIB0 输入
1	0	RXDA0 输入

备注　× = 无需理会

14.1.2 UARTA1 和I^2C的模式转换

在V850E/IF3 和V850E/IG3 中，UARTA1 和I^2C是相同引脚的复用功能，因而不能同时使用。要在UARTA1 和I^2C之间切换，用PMC3 和、PFC3 和PFCE3 寄存器预先设置。

注意事项 如果在发送或接收期间切换这些功能，则UARTA1 和I^2C的发送和接收操作得不到保证。确保禁止未使用的功能。

图 14-2. UARTA1 和I^2C模式转换设置

复位后：00H　R/W　地址：FFFFF446H

	7	6	5	4	3	2	1	0
PMC3	PMC37	PMC36	PMC35	PMC34	PMC33	PMC32	PMC31	PMC30

复位后：00H　R/W　地址：FFFFF466H

	7	6	5	4	3	2	1	0
PFC3	PFC37	PFC36	PFC35	PFC34	PFC33	PFC32	PFC31	PFC30

复位后：00H　R/W　地址：FFFFF706H

	7	6	5	4	3	2	1	0
PFCE3	PFCE37	PFCE36	PFCE35	PFCE34	0	PFCE32	PFCE31	PFCE30

PMC31	PFCE31	PFC31	P31 引脚复用功能说明
0	×	×	I/O 端口
1	0	0	TXDA1 输出
1	0	1	SDA I/O
1	1	0	禁止设置
1	1	1	禁止设置

PMC30	PFCE30	PFC30	P30 引脚复用功能说明
0	×	×	I/O 端口
1	0	0	RXDA1 输入
1	0	1	SCL I/O
1	1	0	禁止设置
1	1	1	禁止设置

备注　× = 无需理会

14.1.3 UARTA2 和 CSIB1 的模式转换

在 V850E/IF3 和 V850E/IG3 中，UARTA2 和 CSIB1 是相同引脚的复用功能，因而不能同时使用。要在 UARTA2 和 CSIB1 之间切换，用 PMC3 和、PFC3 和 PFCE3 寄存器预先设置。

注意事项　如果在发送或接收期间切换这些功能，则 **UARTA2** 和 **CSIB1** 的发送和接收操作得不到保证。确保禁止未使用的功能。

图 14-3. UARTA2 和 CSIB1 模式转换设置

复位后： 00H　　R/W　　地址： FFFFF446H

	7	6	5	4	3	2	1	0
PMC3	PMC37	PMC36	PMC35	PMC34	PMC33	PMC32	PMC31	PMC30

复位后：00H　　R/W　　地址：FFFFF466H

	7	6	5	4	3	2	1	0
PFC3	PFC37	PFC36	PFC35	PFC34	PFC33	PFC32	PFC31	PFC30

复位后： 00H　　R/W　　地址： FFFFF706H

	7	6	5	4	3	2	1	0
PFCE3	PFCE37	PFCE36	PFCE35	PFCE34	0	PFCE32	PFCE31	PFCE30

PMC34	PFCE34	PFC34	P34引脚复位功能说明
0	×	×	I/O 端口
1	0	0	$\overline{\text{SCKB1}}$ I/O
1	0	1	INTP11 输入
1	1	0	$\overline{\text{CS0}}$[注] 输出
1	1	1	禁止设置

PMC33	PFC33	P33 引脚复位功能说明
0	×	I/O 端口
1	0	SOB1 输出
1	0	TXDA2 输出

PMC32	PFCE32	PFC32	P32 引脚复位功能说明
0	×	×	I/O 端口
1	0	0	SIB1 输入
1	0	1	RXDA2 输入
1	1	0	$\overline{\text{CS1}}$[注] 输出
1	1	1	禁止设置

<R>　注　仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

备注　× = 无需理会

14.2 特性

- 传输速率： 300 bps 至 1.25 Mbps（使用 64 MHz 的外围时钟 (fxx) 和专用波特率发生器）
- 全双工通信：
  内部 UARTA 接收数据寄存器 n（UAnRX）
  内部 UARTA 发送数据寄存器 n（UAnTX）
- 2 引脚配置：
  TXDAn： 发送数据输出引脚
  RXDAn： 接收数据输入引脚
- 接收错误输出功能
  - 奇偶错误
  - 帧错误
  - 溢出错误
- 中断源：3
  - 接收错误中断（INTUAnRE）： 通过三种接收错误类型的或操作（OR），产生该中断。
  - 接收结束中断（INTUAnR）： 在接收使能状态下，串行传输结束后，当接收数据从移位寄存器传送到 UAnRX 寄存器时，产生该中断。
  - 发送使能中断（ INTUAnT）： 在发送使能状态下，当数据从 UAnTX 寄存器传送到移位寄存器时，产生该中断。

- 字符长度： 7 或 8 位
- 奇偶校验功能： 奇校验、 偶校验、 0 校验、无校验
- 发送停止位： 1 或 2 位
- 片上专用波特率发生器
- 可选择 MSB/LSB 先行传输
- 可以使发送/接收数据的输入/输出反相

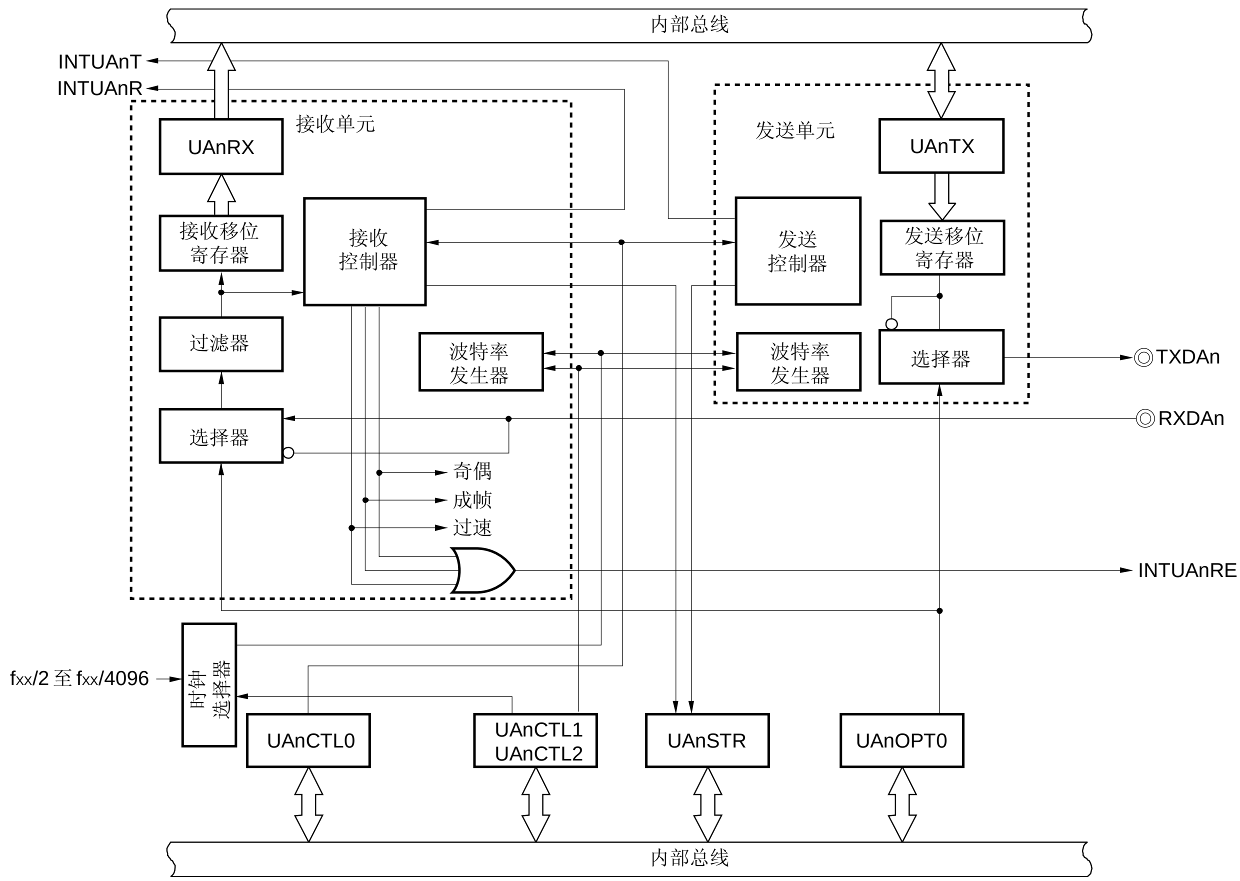
备注 n = 0 至 2

14.3 配置

UARTAn 的功能框图如下所示：

<R>

图 14-4. UARTAn 的功能框图



备注
1. n = 0 至 2
2. 关于波特率发生器的配置，参见图 **14-12**。

UARTAn 包括下列硬件单元：

表 14-1. UARTAn 的配置

项目	配置
寄存器	UARTAn 控制寄存器 0 （UAnCTL0） UARTAn 控制寄存器 1 （UAnCTL1） UARTAn 控制寄存器 2 （UAnCTL2） UARTAn 选项控制寄存器 0 （UAnOPT0） UARTAn 状态寄存器 （UAnSTR） UARTAn 接收移位寄存器 UARTAn 接收数据寄存器（UAnRX） UARTAn 发送移位寄存器 UARTAn 发送数据寄存器（UAnTX）

（1）UARTAn 控制寄存器 0 （UAnCTL0）

UAnCTL0 寄存器是 8 位寄存器，用于指定 UARTAn 的操作。

（2）UARTAn 控制寄存器 1 （UAnCTL1）

UAnCTL1 寄存器是 8 位寄存器，用于选择 UARTAn 的输入时钟。

（3）UARTAn 控制寄存器 2 （UAnCTL2）

UAnCTL2 寄存器是 8 位寄存器，用于控制 UARTAn 的波特率。

（4）UARTAn 选项控制寄存器 0 （UAnOPT0）

UAnOPT0 寄存器是 8 位寄存器，用于控制 UARTAn 的串行传输。

（5）UARTAn 状态寄存器 （UAnSTR）

UAnSTRn 寄存器由指示接收错误的标志位组成。当发生接收错误时，相应接收错误标志被置位（1）。

（6）UARTAn 接收移位寄存器

本寄存器是移位寄存器，用于将输入至 RXDAn 引脚的串行数据转换为并行数据。接收到 1 字节数据并检测到停止位时，就将接收数据传输至 UAnRX 寄存器。

不能直接操作该寄存器。

（7）UARTAn 接收数据寄存器 （UAnRX）

UAnRX 是 8 位寄存器，用于保存接收数据。当接收到 7 个字符时，最高位清 0 （当数据以 LSB 先行方式接收）。

在接收使能状态下，当 1 帧移入操作完毕时，接收数据同步从 UARTAn 接收移位寄存器传送到 UAnRX 寄存器。

传送到 UAnRX 寄存器也会引发接收结束中断请求信号（INTUAnR）。

（8）UARTAn 发送移位寄存器

UARTAn 移位寄存器用于将来自 UAnTX 寄存器的并行数据转换为串行数据。

当一个字节的数据从 UAnTX 寄存器传输过来时，UARTAn 发送移位寄存器的数据从 TXDAn 引脚输出。

不能直接操作该寄存器。

（9）UARTAn 发送数据寄存器 （UAnTX）

UAnTX 寄存器是 8 位发送数据缓存器。当发送的数据被写入 UAnTX 寄存器后，开始发送。当数据能够写入 UAnTX 寄存器时 （当一帧数据从 UAnTX 寄存器传送到 UARTAn 发送移位寄存器时），产生发送使能中断请求信号 （INTUAnT）。

14.4 控制寄存器

（1）UARTAn 控制寄存器 0（UAnCTL0）

UAnCTL0 寄存器是 8 位寄存器，用于控制 UARTAn 的串行传输操作。

该寄存器可进行字节读写或按位读写。

系统复位后，该寄存器被设为 10H。

（1/2）

复位后： 10H　R/W　地址：　UA0CTL0 FFFFFA00H, UA1CTL0 FFFFFA10H,
UA2CTL0 FFFFFA20H

	<7>	<6>	<5>	<4>	3	2	1	0
UAnCTL0 (n = 0 至 2)	UAnPWR	UAnTXE	UAnRXE	UAnDIR	UAnPS1	UAnPS0	UAnCL	UAnSL

UAnPWR	UARTAn 操作控制
0	禁止 UARTAn 操作 (UARTAn 异步复位)
1	使能 UARTAn 操作

UARTAn 操作由 UAnPWR 位控制。 通过清除 UAnPWR 位为 0 将TXDAn引脚输出固定为高电平(如果UAnOPT0.UAnTDL 位 = 1，则固定为低电平).

UAnTXE	发送操作使能
0	发送操作禁止
1	发送操作使能

- 若要开始发送，设置 UAnPWR 位为 1 然后设置UAnTXE 位为 1.
- 若要初始化发送单元，清除 UAnTXE 位为 0, 等待两个周期的基础时钟 (f_{UCLK})，然后再次设置 UAnTXE 位为 1 . 否则无法执行初始化 (关于基础时钟，参见 14.7 (1) (a) 基础时钟)。
- 如果允许操作 (UAnPWR 位 = 1)，基础时钟 (f_{UCLK})的两个或更多周期之后，因为 UAnTXE = 1，发送操作被使能。
- 当UAnPWR位被清除为0，因为UAnPWR位，即使UAnTXE位为1，内部电路的状态也和UAnTXE 位 = 0的状态相同。 当UAnPWR位被再次设置为1时，发送操作使能。

UAnRXE	接收操作使能
0	接收操作禁止
1	接收操作使能

- 若要开始发送，设置 UAnPWR 位为 1 然后设置UAnTXE 位为 1。
- 若要初始化发送单元，清除 UAnRXE 位为 0, 等待两个周期的基础时钟 (f_{UCLK})，然后再次设置 UAnRXE 位为 1 . 否则无法执行初始化 (关于基础时钟，参见 14.7 (1) (a) 基础时钟)。
- 如果允许操作 (UAnPWR 位 = 1)，基础时钟 (f_{UCLK})的两个或更多周期之后，因为 UAnRXE = 1，接收操作被使能。如果在接收操作使能之前被接收，则开始位被忽略。
- 当UAnPWR位被清除为0，因为UAnPWR位，即使UAnRXE位为1，内部电路的状态也和UAnRXE 位 = 0的状态相同。 当UAnPWR位被再次设置为1时，接收操作使能。

（2/2）

UAnDIR[注]	传输方向选择
0	MSB先行传输
1	LSB先行传输

UAnPS1[注]	UAnPS0[注]	发送期间奇偶校验的选择	接收期间奇偶校验的选择
0	0	无奇偶校验输出	无奇偶校验接收
0	1	0 奇偶校验输出	0 奇偶校验接收
1	0	奇校验输出	奇校验检测
1	1	偶校验输出	偶校验检测
如果接收期间选择“0 奇偶校验接收”，则不进行奇偶校验检查。 因此不设置 UAnSTR.UAnPE 位， 由于输出奇偶错误，没有错误中断。			

UAnCL[注]	1 帧的发送/接收数据字符长度的说明
0	7 位
1	8 位

UAnSL[注]	发送数据的停止位长度的说明
0	1 位
1	2 位
仅当检查了接收数据停止位的起始位，不管 UAnSL 位的值如何。	

注　只有当 UAnPWR 位 = 0 或 UAnTXE 位 = UAnRXE 位 = 0 时，可以写入该寄存器。但是，可以同时将 UAnPWR, UAnTXE 和 UAnRXE 的任何位或所有位设置为 1。

备注　关于奇偶校验的详情，参见 **14.6.6 奇偶类型和操作。**

（2）UARTAn 控制寄存器 1 （UAnCTL1）

更多详情，参见 **14.7 （2） UARTAn 控制寄存器 1 （UAnCTL1）**。

（3）UARTAn 控制寄存器 2 （UAnCTL2）

更多详情，参见 **14.7 （3） UARTAn 控制寄存器 2 （UAnCTL2）**。

（4）UARTAn 选项控制寄存器 0 （UAnOPT0）

UAnOPT0 寄存器是 8 位寄存器，用于控制 UARTAn 的串行传输操作。

该寄存器可按字节或位读写。

系统复位后，该寄存器设为 14H。

复位后：14H R/W 地址： UA0OPT0 FFFFFA03H, UA1OPT0 FFFFFA13H, UA2OPT0 FFFFFA23H

	7	6	5	4	3	2	1	0
UAnOPT0 (n = 0 至 2)	0	0	0	1	0	1	UAnTDL	UAnRDL

UAnTDL	发送数据电平位
0	传输数据的正常输出
1	传输数据的反相输出
• TXDAn 引脚的输出电平可以使用 UAnTDL 位进行反相。 • UAnCTL0.UAnPWR 位 = 0 或UAnCTL0.UAnTXE 位 = 0时，可设置该寄存器。	

UAnRDL	接收数据电平位
0	传输数据的正常输入
1	传输数据的反相输入
• RXDAn引脚的输出电平可以使用 UAnRDL 位进行反相。 • UAnPWR 位 = 0 或 UAnCTL0.UAnRXE 位 = 0时，可设置该寄存器。 • 接收启动时，在将UART 接收引脚(RXDAn)设置为数据接收引脚之后，当UAnRDL 位被设置为1 (接收数据的反相输入)时， 接收必须被使能(UAnCTL0.UAnRXE 位 = 1) 。在接收使能之后，引脚模式被改变，如果引脚电平为高电平，将会错误检测到起始位。	

注意事项 **请确保将第 3 位和第 5 位至第 7 位清除为“0”，并将第 2 位和第 4 位设置为“1”。其它设置时，操作无法保证。**

（5）UARTAn 状态寄存器（UAnSTR）

UAnSTR 寄存器是 8 位寄存器，显示 UARTAn 的传输状态和接收错误的内容。

该寄存器可按字节或位读写，但是，UAnTSF 位为只读位，而 UAnPE、UAnFE 和 UAnOVE 位既可读也可写。但是，这些位只能写入 0 来清除，而不能通过写 1 来置位（即便写入 1，其值仍保持不变）。

初始化条件如下：

寄存器/位	初始条件
UAnSTR 寄存器	• 复位之后 • UAnCTL0.UAnPWR = 0
UAnTSF 位	• UAnCTL0.UAnTXE = 0
UAnPE 位、UAnFE 位、UAnOVE 位	• 写入 0 • UAnCTL0.UAnRXE = 0

注意事项 **请确保读取并检查 UAnPE, UAnFE 和 UAnOVE 位的错误标志，并对其写入“0”来清除。**

复位后: 00H R/W 地址: UA0STR FFFFFA04H, UA1STR FFFFFA14H,
UA2STR FFFFFA24H

	<7>	6	5	4	3	<2>	<1>	<0>
UAnSTR (n = 0 至 2)	UAnTSF	0	0	0	0	UAnPE	UAnFE	UAnOVE

UAnTSF	传输状态标志
0	• 已经设置UAnPWR位=0 或 UAnTXE=0。 • 传输完成之后，UAnTX寄存器没有发来下一个数据。
1	写入 UAnTX 寄存器
进行连续发送时，UAnTSF位总是为1。 初始化发送单元时，在进行初始化前确保UAnTSF位= 0。 如果在UAnTSF位= 1时执行初始化，则发送数据无法保证。	

UAnPE	奇偶校验 错误标志
0	• 已经设置UAnPWR位=0 或 UAnRXE位=0。 • 已经写入 0 时。
1	接收期间数据的奇偶位与奇偶校验位不匹配时。
• 通过设置UAnCTL0.UAnPS1 和 UAnCTL0.UAnPS0 位来控制UAnPE位的操作。 • UAnPE位可读可写，但是只能写入0进行清除，而不能写入1进行置位，如果向该位写入1，其值保持不变。	

UAnFE	帧错误标志
0	• 已经设置UAnPWR位=0 或 UAnRXE=0。 • 已经写入 0 时。
1	接收期间没有检测到停止位。
• 只检测接收数据停止位的第一位，而无需考虑UAnCTL0.UAnSL位的值。 • UAnFE位可读可写，但是只能写入0进行清除，而不能写入1进行置位，如果向该位写入1，其值保持不变。	

UAnOVE	溢出错误标志
0	• 已经设置UAnPWR位=0 或 UAnRXE=0。 • 已经写入 0 时。
1	接收数据已经设置到 UAnRX 寄存器 并且在接收数据被读取之前 下一个接收操作已经完成。
• 发生溢出错误时，数据被丢弃，下一个接收数据不会被写入接收缓冲器。 • UAnOVE位可读可写，但是只能写入0进行清除，而不能写入1进行置位，如果向该位写入1，其值保持不变。	

（6）UARTAn 接收数据寄存器 （UAnRX）

UAnRX 寄存器是 8 位缓冲寄存器，用于存储由 UARTAn 接收移位寄存器转换而来的并行数据。

1 个字节的数据接收结束时，将存储于 UARTAn 接收移位寄存器的数据传送到 UAnRX 寄存器。此时会产生接收结束中断请求信号 (INTUAnR)。

LSB 接收期间，数据长度指定为 7 位时，接收数据传输至 UAnRX 寄存器的第 6 位至第 0 位，MSB 位总是为 0。

MSB 接收期间，将接收数据传输到 UAnRX 的第 7 位至第 1 位，且 LSB 位总是为 0。

发生溢出（UAnSTR.UAnOVE 位 = 1）错误时，此时的接收数据不会传送到 UAnRX 寄存器，而是丢弃。

该寄存器是只读寄存器，按字节读取。

除了复位之外，将 UAnCTL0.UAnPWR 位清除为 0， UAnRX 寄存器置为 FFH。

复位后：FFH　R　地址：　UA0RX FFFFFA06H, UA1RX FFFFFA16H,
UA2RX FFFFFA26H

	7	6	5	4	3	2	1	0
UAnRX (n = 0 至 2)								

（7）UARTAn 发送数据寄存器 （UAnTX）

UAnTX 寄存器是 8 位寄存器，用于设置发送数据。

在发送使能状态(UAnCTL0.UAnTXE 位 = 1)下，当发送数据写入到 UAnTX 寄存器时，启动发送。在 UAnTX 寄存器的数据传输到 UARTAn 发送移位寄存器的传输结束时，产生发送使能中断请求信号 (INTUAnT)。

该寄存器可以按字节进行读取或写入。

系统复位后，该寄存器设为 FFH。

复位后： FFH　R/W　地址：　UA0TX FFFFFA07H, UA1TX FFFFFA17H,
UA2TX FFFFFA27H

	7	6	5	4	3	2	1	0
UAnTX (n = 0 至 2)								

14.5 中断请求信号

UARTAn 会产生以下三种中断请求信号：

- 接收错误中断请求信号（INTUAnRE）
- 接收结束中断请求信号（INTUAnR）
- 发送使能中断请求信号（INTUAnT）

在这三个中断信号中，接收错误中断请求信号具有最高的默认优先级，接收结束中断请求信号和发送使能中断请求信号按照下列次序。

表 14-2. 中断及其默认优先级

中断	优先级
接收错误	高
接收结束	⇳
发送使能	低

（1）　接收错误中断请求信号 (INTUAnRE)

当通过对三种类型的接收错误（奇偶检验错误、帧错误以及溢出错误）或（OR）操作来使能接收时，产生一个接收错误中断请求信号。这三种错误在 UAnSTR 寄存器章节给出解释。

（2）　接收完成中断请求信号（INTUAnR）

接收使能状态下，数据转移至 UARTAn 接收移位寄存器并发送至 UAnRX 寄存器时，接收结束中断请求信号输出。

在接收禁止状态下，不产生接收结束中断请求信号。

（3）　发送使能中断请求信号 （INTUAnT）

在发送允许状态下，当发送数据从 UAnTX 寄存器传输至 UARTAn 发送移位寄存器，产生发送使能中断请求信号。

14.6 操作

14.6.1 数据格式

执行全双工串行数据的接收和发送。

如图 14-6 所示，发送/接收数据帧的数据由起始位、字符位、校验位以及停止位组成。

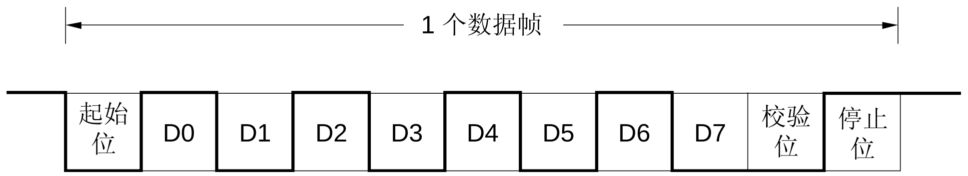
使用 UAnCTL0 寄存器指定数据帧内字符位长度、校验段，指定停止位长度，以及 MSB/LSB 传输方式。

另外，UARTAn 中关于 TXDAn 引脚的输出/反相控制由 UAnOPT0.UAnTDL 位来执行。

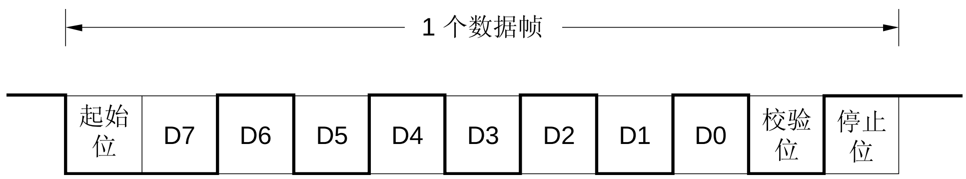
- 起始位 .... 1 位
- 字符位 .... 7 位/8 位
- 奇偶校验位 .... 奇检验/偶校验/0 校验/无校验
- 停止位 .... 1 位/2 位

图 14-5. UARTA 发送/接收数据格式

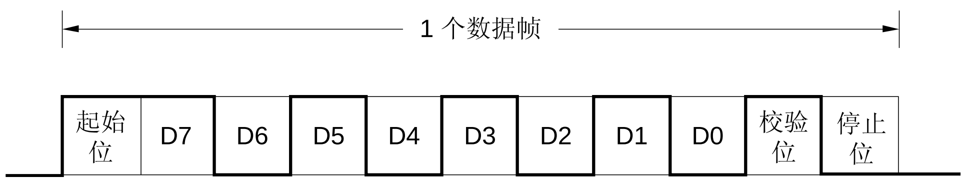
（a）　**8** 位数据长度，**LSB**，偶校验， 停止位，传输数据：**55H**



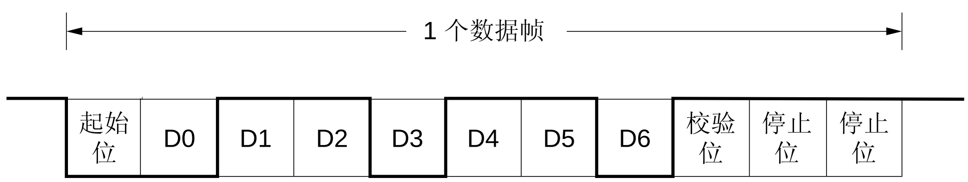
（b）　**8** 位数据长度，**MSB**，偶校验， 停止位，传输数据：**55H**



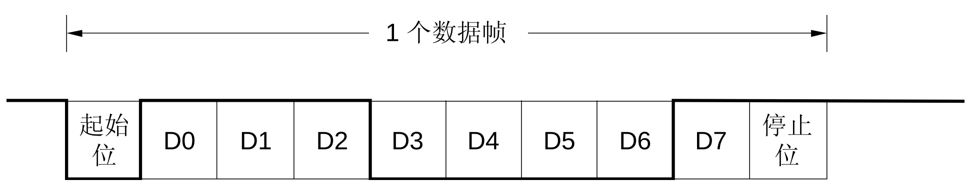
（c）　**8** 位数据长度，**MSB**，偶校验， 停止位，传输数据：**55H**，**TXDAn** 反相



（d）　**7** 位数据长度，**LSB**，奇校验， **2** 个停止位，传输数据：**36H**



（e）　**8** 位数据长度，**LSB**，无校验， 停止位，传输数据：**87H**



14.6.2 UART 发送

设置 UAnCTL0.UAnPWR 位为 1，TXDAn 引脚输出高电平。

之后，设置 UAnCTL0.UAnTXE 位为 1，进入发送允许状态，将发送数据写入 UAnTX 寄存器启动发送。起始位、校验位以及停止位自动添加。

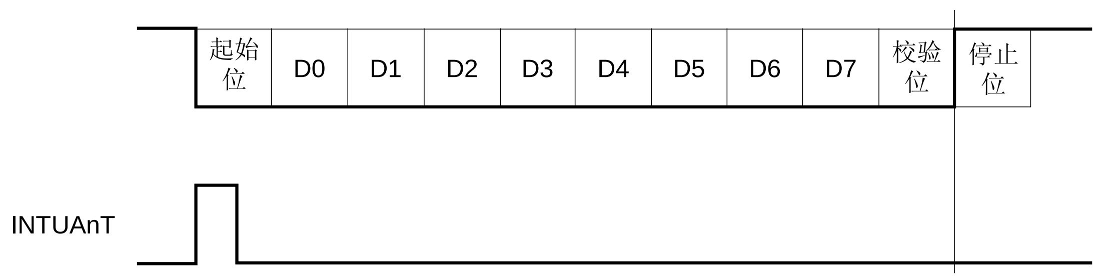
由于 UARTAn 中不提供 CTS（发送允许信号） 输入引脚，所以需要使用一个端口来检查发送目的方已经允许接收。

当发送操作启动之时，UAnTX 寄存器内的数据发送到 UARTAn 发送移位寄存器 。

UAnTX 寄存器的数据发送至 UARTAn 发送移位寄存器的操作结束时，产生发送使能中断请求信号（INTUAnT），之后 UARTAn 发送移位寄存器的内容输出至 TXDAn 引脚。

通过 INTUAnT 信号的产生，使能下一个发送数据写入 UAnTX 寄存器。

图 14-6. UART 发送



备注　**1.** LSB 先行
　　　2. n = 0 至 2

14.6.3 连续发送过程

当 UARTAn 发送移位寄存器启动移位操作时，UARTAn 可以写入下一个数据到 UAnTX 寄存器，UARTAn 发送移位寄存器的发送时序可以由发送使能中断请求信号 （INTUAnT）来判断。

在传输过程中将下一个待发送数据写入 UAnTX 寄存器使得传输效率更高。

注意事项 连续发送执行期间，在检查 UAnSTR.UAnTSF 位为 0 之后，执行初始化。在 UAnTSF 位为 1 时执行初始化，则发送数据无法保证。

备注 n = 0 至 2

图 **14-7.** 连续发送处理框图

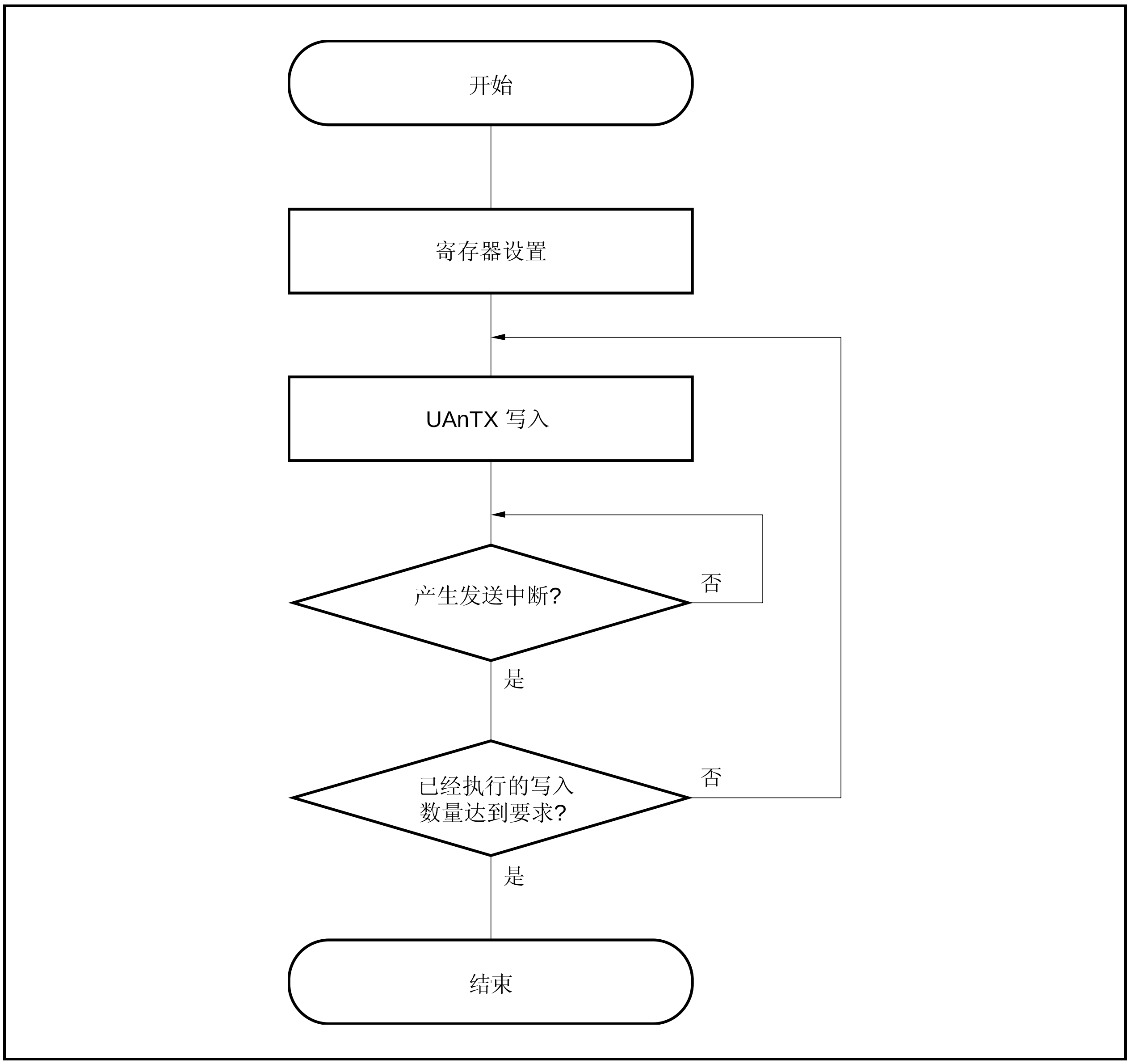
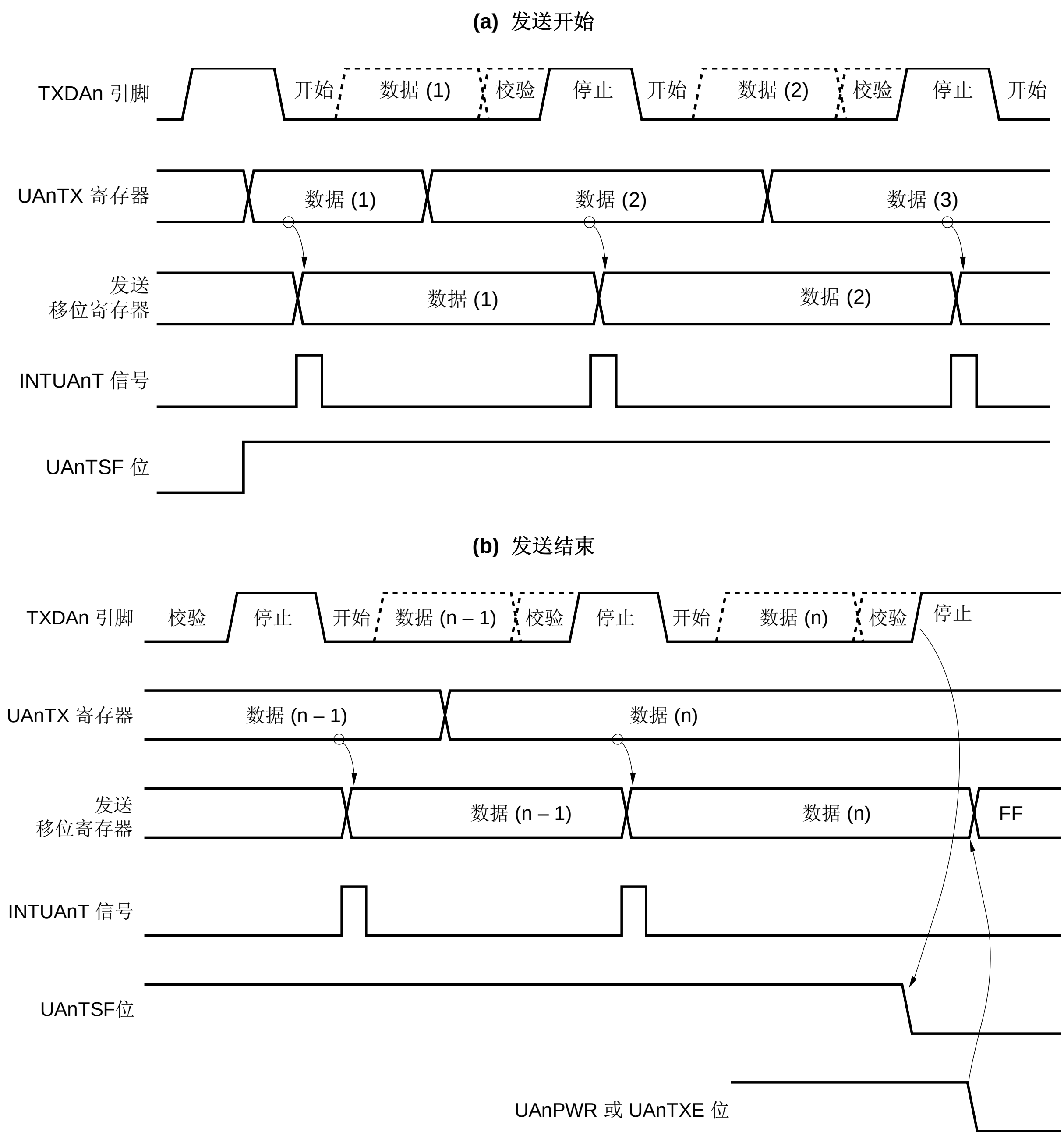


图 **14-8.** 连续发送操作时序



<R>

<R>

备注　　n = 0 至 2

14.6.4 UART 接收

设置 UAnCTL0.UAnPWR 位为 1，然后设置 UAnCTL0.UAnRXE 位为 1，进入接收等待状态。接收等待状态下，RXDAn 引脚被监测并且进行起始位的检测。

应用两步检测程序来进行起始位的检测。

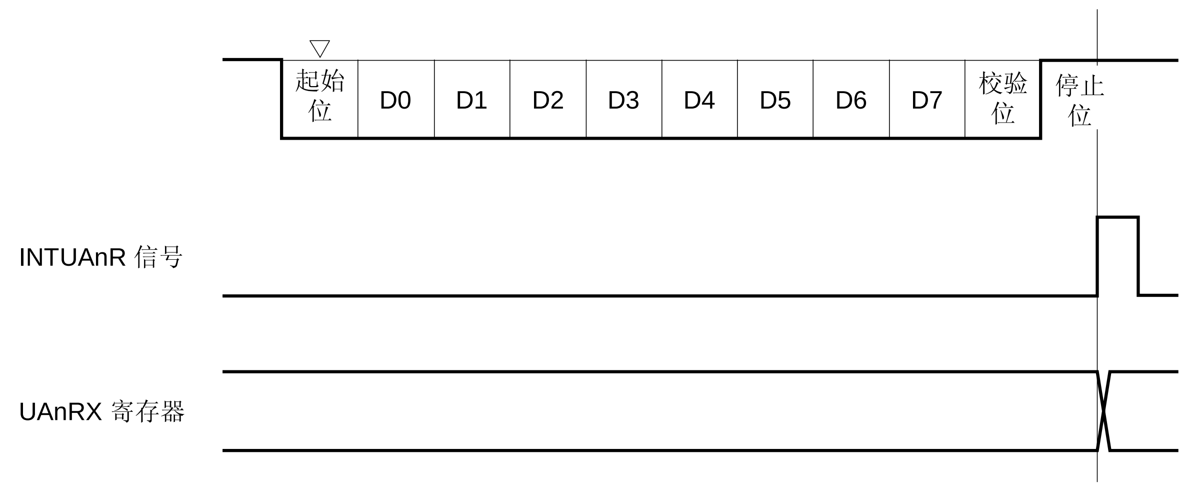
首先检测到 RXDAn 引脚的下降沿，在下降沿处开始采样。如果 RXDAn 引脚在起始位采样点为低电平，确认是起始位。确认起始位之后，开始接收操作，按照设定的波特率，串行数据存入 UARTAn 接收移位寄存器。

接收到停止位时，会输出接收结束中断请求信号（INTUAnR）时，UARTAn 接收移位寄存器的数据写入 UAnRX 寄存器。但是，如果发生溢出（UAnSTR.UAnOVE 位 = 1）错误，此时接收到的数据不写入 UAnRX 寄存器，而是丢弃。

即使接收期间发生奇偶错误（UAnSTR.UAnPE 位 = 1）或帧错误（UAnSTR.UAnFE 位 = 1），接收还会继续进行，直到接收到第一个停止位，并在接收完成后输出 INTUAnRE 信号。

备注　n = 0 至 2

图 14-9. UART 接收



注意事项

1. 即使发生接收错误，也必须读取 **UAnRX** 寄存器。如果不读取 **UAnRX**，则下一个数据接收期间发生溢出错误，并且接收错误会不断地连续发生。

2. 认为接收操作期间只有一个停止位。忽略第二个停止位。

3. 当接收完成时，在产生接收结束中断请求信号（**INTUAnR**）之后读取 **UAnRX** 寄存器，并对 **UAnPWR** 位或 **UAnRXE** 位清 **0**。如果在 **INTUAnR** 信号产生之前对 **UAnPWR** 位或 **UAnRXE** 位清 **0**，则无法保证对 **UAnRX** 寄存器值的正确读取。

4. 如果 **UARTAn** 的接收结束处理（**INTUAnR** 信号产生）和 **UAnPWR** 位**= 0** 或 **UAnRXE** 位 **= 0** 发生冲突，则即 **UAnRX** 寄存器内没有存储数据，也可能会产生 **INTUAnR** 信号。

为了结束接收而无需等待 **INTUAnR** 信号的产生，请确保将中断请求标志（**UAnRIC.UAnRIF**）清除（**0**），在中断屏蔽标志（**UAnRIC.UAnRMK**）置位（**1**）之后，然后设置 **UAnPWR** 位**= 0** 或 **UAnRXE** 位 **= 0**。

14.6.5 接收错误

接收过程中的错误有 3 种类型：奇偶检验错误、帧错误以及溢出错误。数据接收结果错误标志设置在 UAnSTR 寄存器中，当错误发生时，输出接收错误中断请求信号 （INTUAnRE）。

通过读取 UAnSTR 寄存器的内容，可以确定在接收过程中发生了哪种错误 。

读取完成后，写入 0 来清除接收错误标志。

注意事项　接收结束中断请求信号(INTUAnR)和接收错误中断请求信号(INTUAnRE)不会同时产生。当接收正常结束时，产生 INTUAnR 信号。当发生接收错误时，产生 INTUAnRE 信号，且 INTUAnR 信号不产生。

备注　　n = 0 至 2

- 接收错误原因

错误标志	接收错误	原因
UAnPE	奇偶错误	接收到的校验位和设定校验位不一致
UAnFE	帧错误	未检测到停止位
UAnOVE	溢出错误	在数据从 UAnRX 寄存器读出之前，下一个数据接收结束

14.6.6 奇偶类型和操作

奇偶位用于检测通信数据中的位错误。通常情况下，发送端和接收端应该使用相同的奇偶校验。
在使用奇校验和偶校验情况下，可以检测到奇数计数的位错误。而在 0 奇偶校验和无校验情况下，检测不到错误。

（a） 偶校验

（i） 在发送期间

发送数据之中位值为“1”的数量，包括奇偶校验位，控制其数量为偶数。奇偶位的值如下：

- 发送数据中位值为“1”的数量为奇数：　1
- 发送数据中位值为“1”的数量为偶数：　0

（ii） 在接收期间

计算接收数据中位值为“1”的数量，包括校验位，如果为奇数，则输出奇偶校验错误。

（b） 奇校验

（i） 在发送期间

与偶校验相对，发送数据中位值为“1”的数量，包括奇偶校验位，控制其数量为奇数。奇偶位的值如下：

- 发送数据中位值为“1”的数量为奇数：　0
- 发送数据中位值为“1”的数量为偶数：　1

（ii） 在接收期间

计算接收数据中位值为“1”的数量，包括校验位，如果为偶数，则输出奇偶校验错误标志。

（c） 0 奇偶校验

在发送期间，奇偶校验位总是“0”，而与发送数据无关。
在接收期间，不执行奇偶校验。因而，无论奇偶位为“0” 或 “1”，均不会产生奇偶错误。

（d） 无奇偶校验

不向发送数据附加奇偶位。
执行接收假设无无奇偶位。既然无奇偶位则无奇偶错误发生。

14.6.7　接收数据噪声过滤器

该滤波器使用预分频器输出的基础时钟(f_{UCLK})对 RXDAn 引脚进行采样。

如果读到两次相同的采样值，则一致检测器输出改变，并且采样的 RXDAn 信号作为输入数据。所以不超过两个时钟宽度的数据判定为噪声，不会传达到内部电路（参见 **图 14-11**）。有关基础时钟，参见 14.7 （1）（a） 基础时钟 。

另外，由于电路如图 14-10 所示，所以，接收操作的处理过程与外部信号状态相比，延迟了 3 个时钟周期。

备注　　n = 0 至 2

图 14-10. 噪声过滤电路

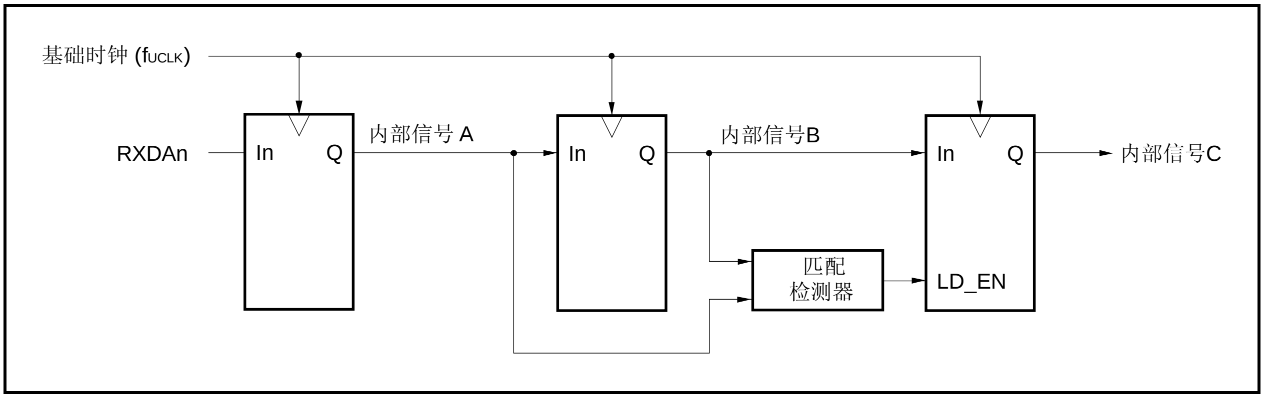
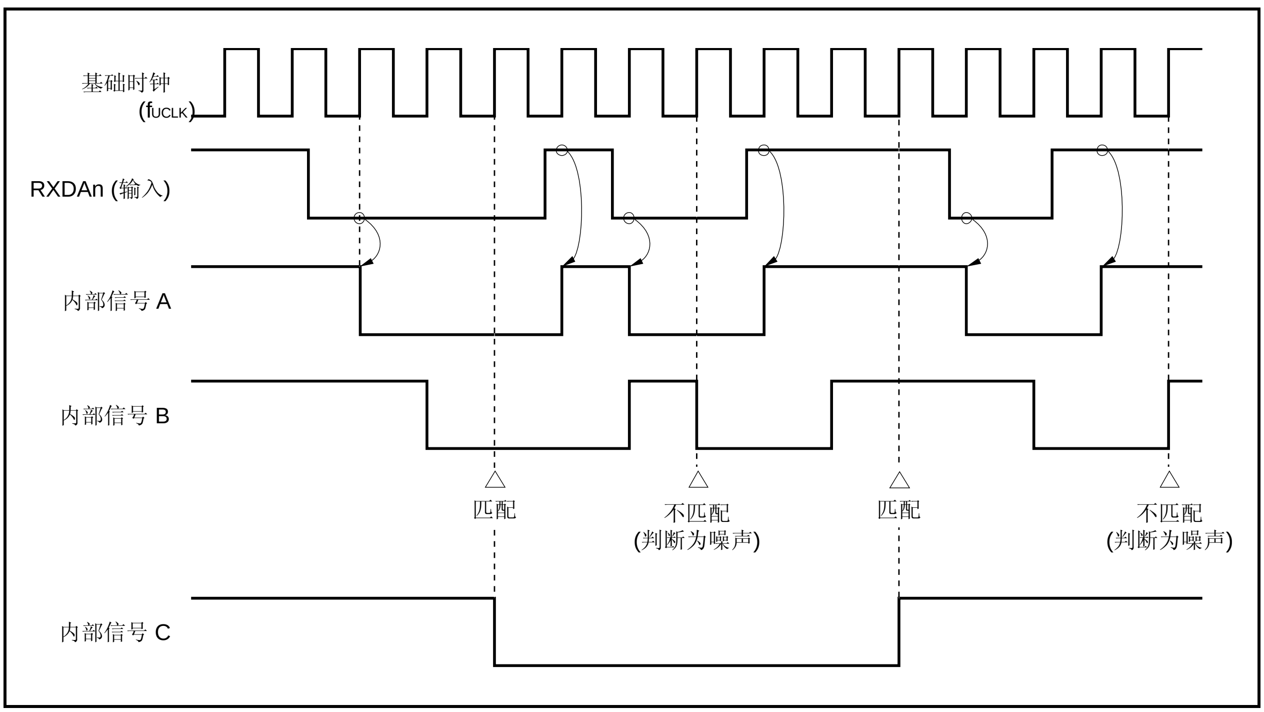


图 14-11. RXDAn 信号判定为噪声的时序



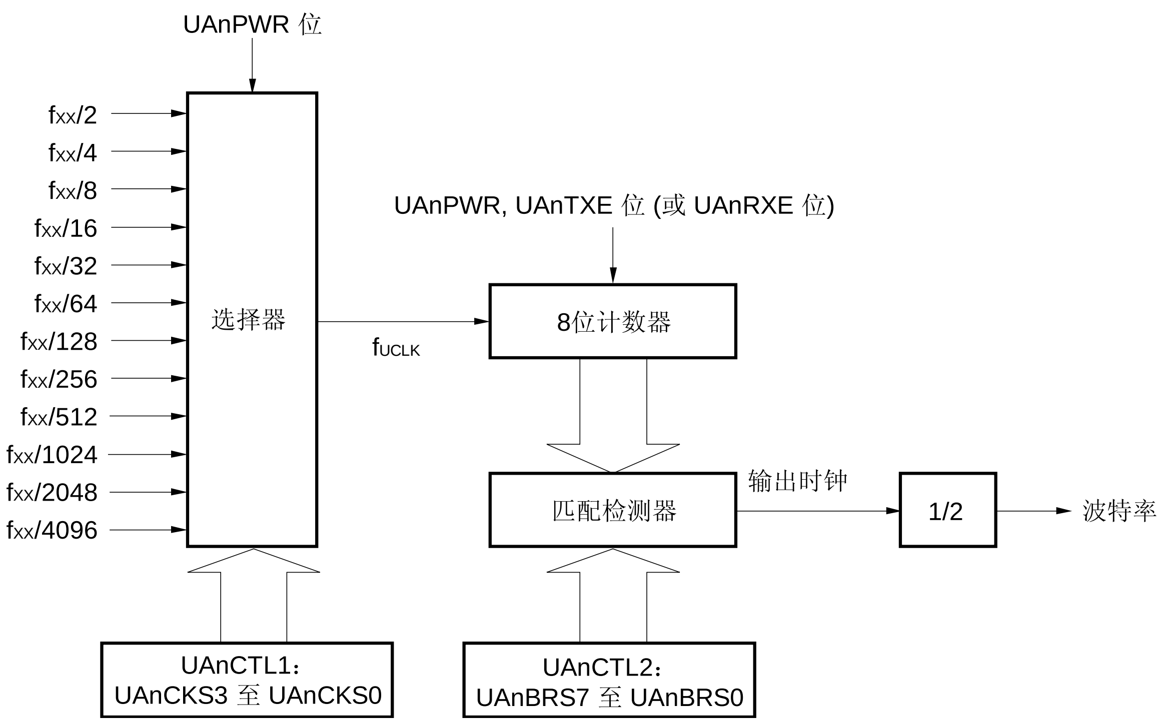
14.7 专用波特率发生器

专用波特率发生器包括一个时钟源选择器和一个 8 位可编程计数器，在 UARTAn 发送/接收期间产生串行时钟。关于串行时钟，对各个通道可以选择专用波特率发生器输出。

一个 8 位计数器用于发送，另一个用于接收。

（1）　波特率发生器的配置

图 14-12. 波特率发生器的配置



注意事项　如果 **CPU** 时钟 **(f_{CPU})** 比 **f_{UCLK}** 慢，不能使用 **UARTAn**。

备注　**1.** n = 0 至 2
　　　2. f_{XX}：　外围时钟频率

（a）　基础时钟

当 UAnCTL0.UAnPWR 位为 1 时，由 UAnCTL1.UAnCKS3 至 UAnCTL1.UAnCKS0 位选择的时钟提供给 8 位计数器。该时钟称为基础时钟（f_{UCLK}）。当 UAnPWR 位 = 0 时，f_{UCLK} 固定为低电平。

（b）　串行时钟的产生

通过设置 UAnCTL1 寄存器和 UAnCTL2 寄存器，来产生一个串行时钟。

基础时钟（f_{UCLK}）由 UAnCTL1.UAnCKS3 至 UAnCTL1.UAnCKS0 位选择。

8 位计数器的分频值可以使用 UAnCTL2.UAnBRS7 至 UAnCTL2.UAnBRS0 位进行设置。

（2） UARTAn 控制寄存器 1 （UAnCTL1）

UAnCTL1 寄存器是 8 位寄存器，用于选择 UARTAn 基础时钟。

该寄存器可以按字节或按位进行读取或写入。

系统复位后，该寄存器被设置为 00H。

注意事项 在改写 UAnCTL1 寄存器之前，将 UAnCTL0.UAnPWR 位清 0。

复位后： 00H R/W 地址： UA0CTL1 FFFFFA01H, UA1CTL1 FFFFFA11H, UA2CTL1 FFFFFA21H

	7	6	5	4	3	2	1	0
UAnCTL1 (n = 0 至 2)	0	0	0	0	UAnCKS3	UAnCKS2	UAnCKS1	UAnCKS0

UAnCKS3	UAnCKS2	UAnCKS1	UAnCKS0	基础时钟(f_{UCLK}) 选择
0	0	0	0	fxx/2
0	0	0	1	fxx/4
0	0	1	0	fxx/8
0	0	1	1	fxx/16
0	1	0	0	fxx/32
0	1	0	1	fxx/64
0	1	1	0	fxx/128
0	1	1	1	fxx/256
1	0	0	0	fxx/512
1	0	0	1	fxx/1,024
1	0	1	0	fxx/2,048
1	0	1	1	fxx/4,096
其它情况				禁止设置

备注 f_{XX}：外围时钟频率

（3） UARTAn 控制寄存器 2（UAnCTL2）

UAnCTL2 寄存器是 8 位寄存器，用于选择 UARTAn 的波特率（串行传输速度）。

该寄存器可以按字节进行读取或写入。

系统复位后，该寄存器被设置为 FFH。

注意事项 **在改写 UAnCTL2 寄存器之前，将 UAnCTL0.UAnPWR 位清除为 0，或将 UAnTXE 和 UAnRXE 位清除为 0。**

复位后： FFH　R/W　地址：　UA0CTL2 FFFFFA02H, UA1CTL2 FFFFFA12H, UA2CTL2 FFFFFA22H

UAnCTL2 (n = 0 至 2)

7	6	5	4	3	2	1	0
UAnBRS7	UAnBRS6	UAnBRS5	UAnBRS4	UAnBRS3	UAnBRS2	UAnBRS1	UAnBRS0

UAn BRS7	UAn BRS6	UAn BRS5	UAn BRS4	UAn BRS3	UAn BRS2	UAn BRS1	UAn BRS0	缺省值 (k)	串行时钟
0	0	0	0	0	0	×	×	×	禁止设置
0	0	0	0	0	1	0	0	4	$f_{UCLK}/4$
0	0	0	0	0	1	0	1	5	$f_{UCLK}/5$
0	0	0	0	0	1	1	0	6	$f_{UCLK}/6$
:	:	:	:	:	:	:	:	:	:
1	1	1	1	1	1	0	0	252	$f_{UCLK}/252$
1	1	1	1	1	1	0	1	253	$f_{UCLK}/253$
1	1	1	1	1	1	1	0	254	$f_{UCLK}/254$
1	1	1	1	1	1	1	1	255	$f_{UCLK}/255$

备注　f_{UCLK}：基础时钟的频率由 UAnCTL1.UAnCKS3 至 UAnCTL1.UAnCKS0 位选择。

（4）　波特率

波特率误差由下列公式计算。

$$波特率 = \frac{f_{UCLK}}{2 \times k} [bps]$$

f_{UCLK}： 由 UAnCTL1.UAnCKS3 至 UAnCTL1.UAnCKS0 位选择的基础时钟频率

k： 由 UAnCTL2.UAnBRS7 至 UAnCTL2.UAnBRS0 位设置的值（k = 4 至 255）

(5) 波特率误差

波特率误差由下列公式计算：

$$误差（\%）= \left(\frac{实际波特率（有错误的波特率）}{目标波特率（正确的波特率）} - 1 \right) \times 100\ [\%]$$

注意事项 1. 发送过程中，波特率误差必须在接收端的误差容限之内。

2. 接收时的波特率误差满足一定范围，该范围参见节（7）接收期间允许的波特率误差范围。

示例： 外围时钟频率 = 32 MHz = 32,000,000 Hz

UAnCTL1.UAnCKS3 至 UAnCTL1.UAnCKS0 位的设置值 = 0000B (f_{UCLK} = 16,000,000 Hz)

UAnCTL2.UAnBRS7 至 UAnCTL2.UAnBRS0 位的设置值 = 00110100B (k = 52)

目标波特率 = 153,600

波特率 = 16,000,000/ (2 × 52) = 153,846 [bps]

误差 = (153,846/153,600 – 1) × 100

= 0.160 [%]

(6) 波特率设置举例

表 **14-3.** 波特率发生器设置数据

波特率 (bps)	f_{XX} = 64 MHz			f_{XX} = 32 MHz		
	UAnCTL1	UAnCTL2	ERR (%)	UAnCTL1	UAnCTL2	ERR (%)
300	08H	D0H	0.16	07H	D0H	0.16
600	07H	D0H	0.16	06H	D0H	0.16
1,200	06H	D0H	0.16	05H	D0H	0.16
2,400	05H	D0H	0.16	04H	D0H	0.16
4,800	04H	D0H	0.16	03H	D0H	0.16
9,600	03H	D0H	0.16	02H	D0H	0.16
19,200	02H	D0H	0.16	01H	D0H	0.16
31,250	02H	80H	0	00H	80H	0
38,400	01H	D0H	0.16	00H	D0H	0.16
76,800	00H	D0H	0.16	00H	68H	0.16
153,600	00H	68H	0.16	00H	34H	0.16
312,500	00H	33H	0.39	00H	1AH	–1.54
625,000	00H	1AH	–1.54	00H	0DH	–1.54
1,250,000	00H	0DH	–1.54	00H	06H	6.67

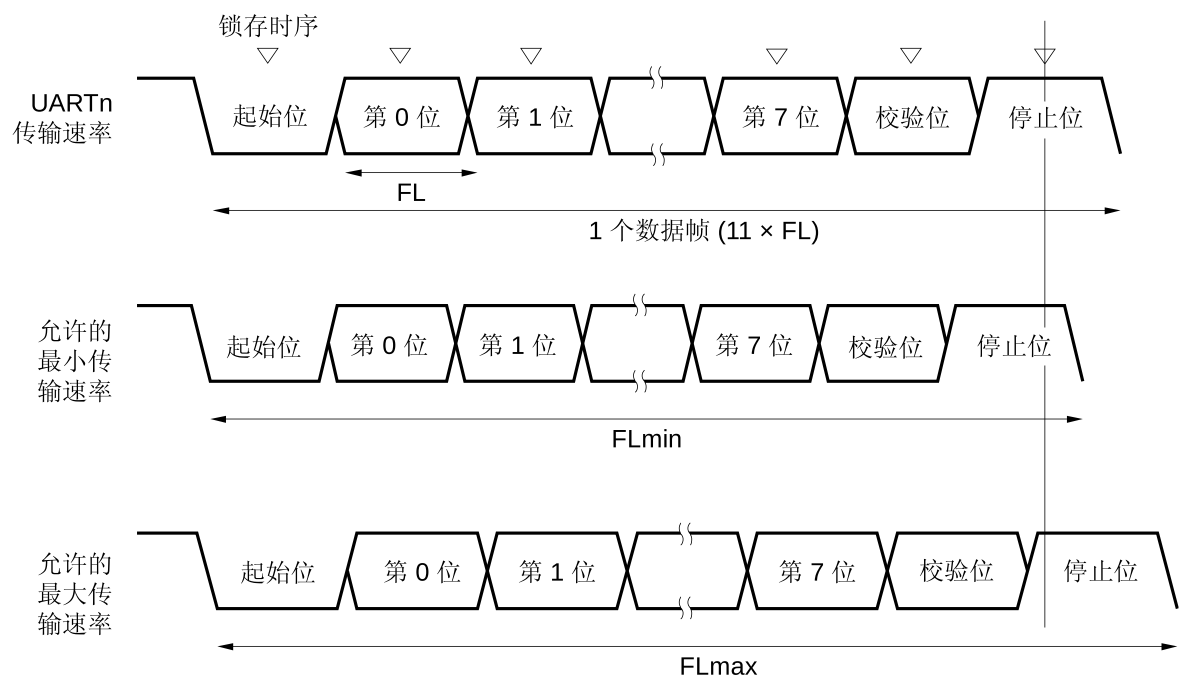
备注　f_{XX}：　外围时钟频率
　　　ERR：　波特率误差（%）

（7） 接收期间允许的波特率范围

接收过程中，在目标允许的波特率误差范围如下所示：

注意事项 接收过程中使用以下公式，波特率误差必须设置在允许范围内。

图 14-13. 接收期间允许的波特率范围



备注 n = 0 至 2

如图 14-13 所示，接收数据的锁存时序取决于检测到起始位后 UAnCTL2 寄存器对计数器的设置值。如果在此锁存时序可以及时接收直到最末位数据（停止位），发送数据就可以正常接收。

当应用于 11 位接收时，以下为理论结果。

FL = （波特率）$^{-1}$

波特率：UARTAn 波特率（n = 0 至 2）

k： UAnCTL2.UAnBRS7 至 UAnCTL2.UAnBRS0 位的设定值（n = 0 至 2）。

FL： 1 位数据长度

锁存时间余量： 2 个时钟

最小允许传输速率： $FLmin = 11 \times FL - \frac{k-2}{2k} \times FL = \frac{21k+2}{2k} FL$

所以，可以从目的端接收的最大波特率如下所示：

$$BRmax = （FLmin/11）^{-1} = \frac{22k}{21k + 2} Brate$$

同理，可以得到下列公式来计算最大允许传输速率：

$$\frac{10}{11} \times FLmax = 11 \times FL - \frac{k + 2}{2 \times k} \times FL = \frac{21k - 2}{2 \times k} FL$$

$$FLmax = \frac{21k - 2}{20\ k} FL \times 11$$

所以，可以从目的端接收的最小波特率如下所示：

$$BRmin = （FLmax/11）^{-1} = \frac{20k}{21k - 2} Brate$$

由上述公式计算 UARTAn 以及目的端的允许波特率误差，得到的最大和最小波特率值必须遵从下表。

表 14-4. 允许的最大/最小波特率误差

分频比（k）	允许的最大波特率误差	允许的最小波特率误差
4	+2.32%	- 2.43%
8	+3.52%	- 3.61%
20	+4.26%	- 4.30%
50	+4.56%	- 4.58%
100	+4.66%	- 4.67%
255	+4.72%	- 4.72%

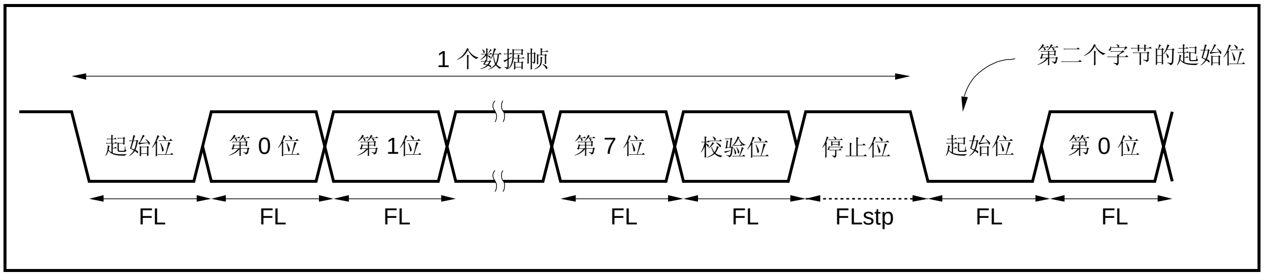
备注 **1.** 接收精度依赖于一帧内的位数量、输入时钟频率和分频比（k）。输入时钟频率越高且分频比（k）越大，则精度越高。

2. k： UAnCTL2.UAnBRS7 位至 UAnCTL2.UAnBRS0 位的设定值（n = 0 至 2）

（8）连续发送期间的传输速率

连续发送期间，从停止位到下一个起始位的传输间隔通常大于 2 个基础时钟。而时序的初始化由接收端通过检测起始位执行，因此，这对传输结果没有影响。

图 14-14. 连续发送期间的传输速率



假定 1 位数据长度为 FL；停止位长度为 FLstp；基础时钟频率为 f_{UCLK}，可以得到如下等式：

$$FLstp = FL + 2/f_{UCLK}$$

所以，连续发送期间的传输速率如下。

$$\text{传输速率} = 11 \times FL + (2/f_{UCLK})$$

14.8 注意事项

当提供给 UARTAn 的时钟停止（比如进入 IDLE 或 STOP 模式）时，则操作停止，每一个寄存器都保留时钟供应停止前的瞬间值。TXDAn 引脚输出也保持时钟供给停止前的瞬时值。然而，恢复时钟供给后，不能保证正常操作。因此，在时钟恢复之后，应该将 UAnCTL0.UAnPWR 位、UAnCTL0.UAnRXE 位和 UAnCTL0.UAnTXE 位设置为 000，对电路进行初始化。

备注　　n = 0 至 2

15.1 UARTB 和 CSIB2 的模式转换

在 V850E/IF3 和 V850E/IG3 中，UARTB 和 CSIB2 是相同引脚的复用功能，因而不能同时使用。要在 UARTB 和 CSIB2 和之间切换，用 PMC3、PFC3 和 PFCE3 寄存器预先设置。

注意事项 **如果在发送或接收期间切换这些功能，则 UARTB 或 CSIB2 的发送和接收操作得不到保证。确保禁止未使用的功能。**

图 15-1. UARTB 和 CSIB2 模式转换设置

复位后：00H　R/W　地址：FFFFF446H

	7	6	5	4	3	2	1	0
PMC3	PMC37	PMC36	PMC35	PMC34	PMC33	PMC32	PMC31	PMC30

复位后：00H　R/W　地址：FFFFF466H

	7	6	5	4	3	2	1	0
PFC3	PFC37	PFC36	PFC35	PFC34	PFC33	PFC32	PFC31	PFC30

复位后：00H　R/W　地址：FFFFF706H

	7	6	5	4	3	2	1	0
PFCE3	PFCE37	PFCE36	PFCE35	PFCE34	0	PFCE32	PFCE31	PFCE30

PMC37	PFCE37	PFC37	P37 引脚复用功能说明
0	×	×	I/O 端口
1	0	0	$\overline{\text{SCKB2}}$ I/O
1	0	1	INTP12 输入
1	1	0	ASTB[注] 输出
1	1	1	禁止设置

PMC36	PFCE36	PFC36	P36 引脚复用功能说明
0	×	×	I/O 端口
1	0	0	SOB2 输出
1	0	1	TXDB 输出
1	1	0	禁止设置
1	1	1	禁止设置

PMC35	PFCE35	PFC35	P35 引脚复用功能说明
0	×	×	I/O 端口
1	0	0	SIB2 输入
1	0	1	RXDB 输入
1	1	0	禁止设置
1	1	1	禁止设置

<R> 注　仅限 μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

备注　x = 无需理会

15.2 特性

- 传输速率： 最高 5.33 Mbps（使用专用波特率发生器）
- 全双工通信
- 单次模式和 FIFO 模式可选
  - 单次模式： 每个发送和接收使用 8 位 × 1 级数据寄存器(UBTX 寄存器或 UBRX 寄存器)
  - FIFO 模式

    发送 FIFO： UBTX 寄存器 (8 位 × 16 级).

    接收 FIFO： UBRXAP 寄存器 (16 位 × 16 级)

    UBRXAP 寄存器的高 8 位的 2 位用作错误标志
- 2 引脚配置：

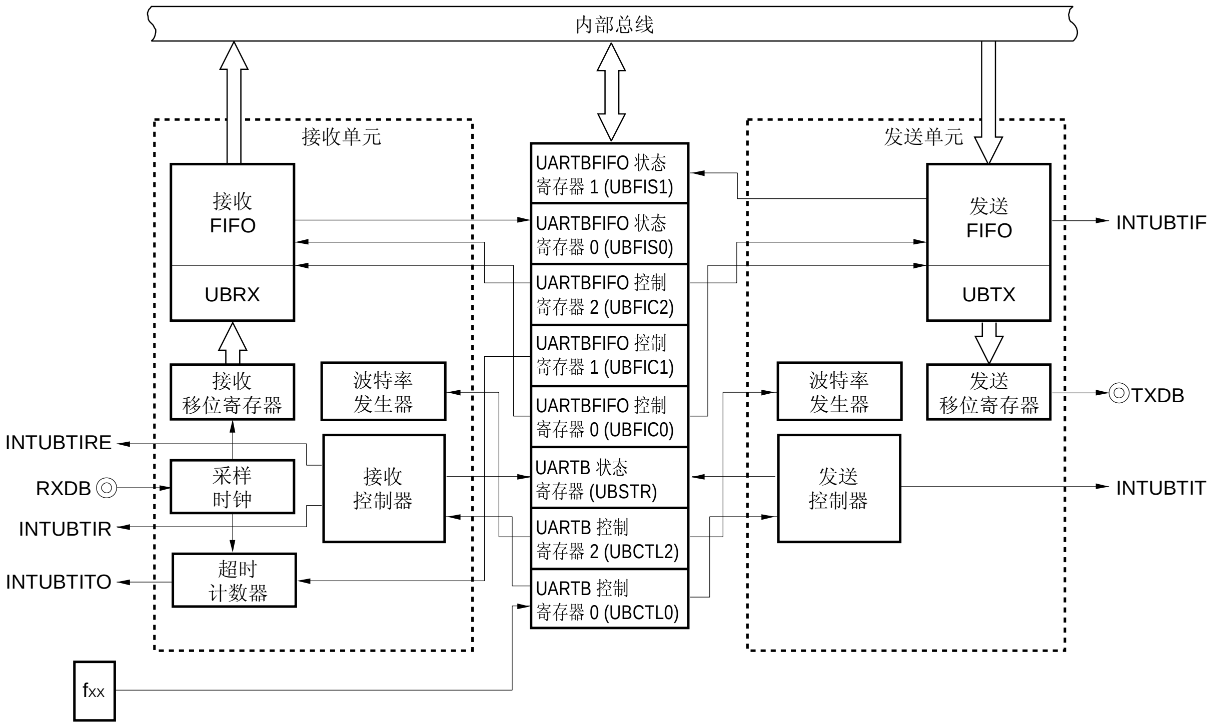
  TXDB： 发送数据输出引脚

  RXDB： 接收数据输入引脚
- 接收错误检测功能
  - 溢出错误（仅限 FIFO 模式）
  - 奇偶错误
  - 帧错误
  - 过载错误（仅限单次模式）
- 中断源：5 种类型
  - 接收错误中断请求信号(INTUBTIRE)
  - 接收结束中断请求信号(INTUBTIR)
  - 发送使能中断请求信号(INTUBTIT)
  - FIFO 发送结束中断请求信号(INTUBTIF)（仅限 FIFO 模式）
  - 接收超时中断请求信号(INTUBTITO)（仅限 FIFO 模式）
- 发送/接收数据的字符长度根据 UBCTL0 寄存器来指定
- 字符长度：7 或 8 位
- 奇偶校验功能： 奇校验、偶校验、0 校验、无校验
- 发送停止位：1 或 2 位
- 传输数据可选择 MSB/LSB 先行
- 片上专用波特率发生器

15.3 配置

UARTB 的功能框图如下所示：

图 15-2. UARTB 的功能框图



备注　**1.** fxx：外围时钟

2. 关于波特率发生器的配置，参见图 **15-9**。

UARTB 包括下列硬件：

表 15-1. UARTB 的配置

项目	配置
寄存器	UARTB 控制寄存器 0 （UBCTL0） UARTB 控制寄存器 2 （UBCTL2） UARTB 状态寄存器 （UBSTR） UARTB FIFO 控制寄存器 0 (UBFIC0) UARTB FIFO 控制寄存器 1 (UBFIC1) UARTB FIFO 控制寄存器 2 (UBFIC2) UARTB FIFO 状态寄存器 0 (UBFIS0) UARTB FIFO 状态寄存器 1 (UBFIS1) 接收移位寄存器 UARTB 接收数据寄存器 AP（UBRXAP） UARTB 接收数据寄存器（UBRX） 发送移位寄存器 UARTB 发送数据寄存器（UBTX）

（1）UARTB 控制寄存器 0（UBCTL0）

该寄存器用于控制 UARTB 的传输操作。

（2）UARTB 状态寄存器 （UBSTR）

该寄存器表示发送期间的传输状态，也表示接收错误的内容。该寄存器的状态标志，表示发送期间的传输状态，表示发送移位寄存器和发送数据寄存器(单次模式下的 UBTX 寄存器或 FIFO 模式下的发送 FIFO)的数据保留状态。当发生接收错误时，每个接收错误标志被设置为 1，当 0 写入 UBSTR 寄存器时，清除为 0。

（3）UARTB 控制寄存器 2 （UBCTL2）

该寄存器用于指定分频比率，并通过它来控制 UARTB 的波特率（串行传输速度）。

(4) UARTB FIFO 控制寄存器 0 (UBFIC0)

该寄存器用于选择 UARTB 的操作模式，在 FIFO 模式下，可以清除发送 FIFO/接收 FIFO。并指定时序模式，按照时序发生发送使能中断请求信号(INTUBTIT)/接收结束中断请求信号(INTUBTIR)。

(5) UARTB FIFO 控制寄存器 1 (UBFIC1)

在 FIFO 模式下，该寄存器有效。在接收到停止位之后，即使经过了下一数据的接收等待时间，下一个数据也没有到来(没有检测到开始位)时，如果数据存储在接收 FIFO，产生一个接收超时中断请求信号(INTUBTITO)。

(6) UARTB FIFO 控制寄存器 2 (UBFIC2)

在 FIFO 模式下，该寄存器有效。用来设置产生发送使能中断请求信号(INTUBTIT)/接收结束中断请求信号(INTUBTIR)的时序，使用已发送或接收的数据数量作为触发。

(7) UARTB FIFO 状态寄存器 0 (UBFIS0)

在 FIFO 模式下，该寄存器有效。存储在接收 FIFO 内的字节数量可以从该寄存器读取获得。

(8) UARTB FIFO 状态寄存器 1 (UBFIS1)

在 FIFO 模式下，该寄存器有效。发送 FIFO 内的空字节数量可以从该寄存器读取获得。

（9）接收移位寄存器

该寄存器是移位寄存器，用于将输入至 RXDB 引脚的串行数据转换为并行数据。接收到 1 字节数据，如果检测到停止位，就将接收到的数据传输到接收数据寄存器。

不能直接操作该寄存器。

(10) UARTB 接收数据寄存器 AP (UBRXAP)，UARTB 接收数据寄存器(UBRX)

接收数据寄存器保持接收数据。在单次模式下，使用 8 位 × 1 级 UBRX 寄存器。在 FIFO 模式下，使用 16 位 × 16 级接收 FIFO (UBRXAP 寄存器)。

接收数据存储在接收 FIFO (UBRXAP 寄存器)的低 8 位，接收数据的错误信号存储在高 8 位(第 8 位和第 9 位)。如果在 FIFO 模式下发生一个接收错误(比如奇偶校验错误或帧错误)，通过按照 16 位单元宽度（半字）读取 UBRXAP 寄存器(错误信息被附加，UBPEF 位 = 1 或 UBFEF 位 = 1)，可以识别错误数据。当 UBRXAP 寄存器的低 8 位按照 8 位单元（字节）读取时，高 8 位被丢弃。因此，如果没有发生错误，只有按照 8 位单元（字节）读取，才可以连续读取 UBRXAP 寄存器的接收数据，和读取 UBRX 寄存器的方式相同。

以 LSB 先行方式接收到 7 位长度数据时，接收到的数据从 LSB（第 0 位）开始，被传输到接收数据寄存器的第 6 位至第 0 位，MSB（第 7 位）总是 0。以 MSB 先行方式接收数据时，接收到的数据从 MSB（第 7 位）开始，被传输到接收数据寄存器的第 7 位至第 1 位，LSB（第 0 位）总是 0。如果发生过载错误，此时的接收数据不传输到接收数据寄存器。

当接收使能时，接收到的数据从接收移位寄存器传输到接收数据寄存器，和一帧的移入处理同步。

在单次模式下，通过将数据传输到 UBRX 寄存器，或者在 FIFO 模式下，将指定数量的接收数据传输到接收 FIFO，都会产生一个接收结束中断请求信号(INTUBTIR)，接收数据的数量由 UBFIC2.UBRT3 至 UBFIC2.UBRT0 位设置为触发器。在 FIFO 模式下，在经过了 UBFIC1.UBTC4 至 UBFIC1.UBTC0 位指定的下一数据接收等待时间后，下一个数据也没有到来(没有检测到开始位)时，如果数据存储在接收 FIFO，则产生一个接收超时中断请求信号(INTUBTITO)。

（11） 发送移位寄存器

该寄存器是移位寄存器，用于将来自发送数据寄存器的并行数据转换为串行数据。

当一个字节的数据从发送数据寄存器传送过来时，移位寄存器的数据从 TXDB 引脚输出。

不能直接操作该寄存器。

(12) UARTB 发送数据寄存器(UBTX)

发送数据寄存器是一个发送数据的缓冲器。在单次模式下，使用 8 位 × 1 级 UBTX 寄存器作为该缓冲器。在 FIFO 模式下，使用 8 位 × 16 级发送 FIFO。

以 LSB 先行方式发送出 7 位长度数据时，从 LSB（第 0 位）开始，发送数据寄存器的第 6 位至第 0 位作为发送数据发出，MSB（第 7 位）总是 0。以 MSB 先行方式发送数据时，从 MSB（第 7 位）开始，发送数据寄存器的第 7 位至第 1 位作为发送数据发出，LSB（第 0 位）总是 0。

在单次模式下，当发送使能 (UBCTL0.UBTXE 位 = 1)时，通过将发送数据写入 UBTX 寄存器，可以启动发送。将发送数据写入 UBTX 寄存器的操作使能时（当一个字节的数据从 UBTX 寄存器传输到发送移位寄存器时），产生一个发送使能中断请求信号(INTUBTIT)。

在 FIFO 模式下，将不少于指定数量但少于 16 字节的发送数据写入到发送 FIFO，然后使能发送（UBTXE 位 = 1），可以启动发送，发送数据的指定数量由 UBFIC2.UBTT3 至 UBFIC2.UBTT0 位设置为触发器。当由 UBFIC2.UBTT3 至 UBFIC2.UBTT0 位设置为触发器的指定数量的发送数据从发送 FIFO 传输到发送移位寄存器（设置为触发器的指定数量的发送数据可以写入）时，产生一个发送使能中断请求信号(INTUBTIT)。在 FIFO 模式下，如果在发送 FIFO 和发送移位寄存器内没有数据（当 FIFO 和该寄存器为空时），则产生一个 FIFO 发送使能中断请求信号(INTUBTIF)。

(13) 超时计数器

当接收到的数据数量没有达到 UBFIC2.UBRT3 至 UBFIC2.UBRT0 位设置作为触发的数量时，该计数器用于识别接收 FIFO 中存在（保持）的数据。只有在 FIFO 模式下有效。

在经过了 UBFIC1.UBTC4 至 UBFIC1.UBTC0 位指定的下一数据接收等待时间后，下一个数据也没有到来(没有检测到开始位)时，如果数据存储在接收 FIFO，则产生一个接收超时中断请求信号(INTUBTITO)。

(14) 采样块

这个块在外围时钟(fxx)的上升沿处对 RXDB 信号进行采样。如果两次检测到相同的采样值，一致检测器的输出改变，采样到的值作为输入数据。小于一个时钟宽度的数据被判断为噪声，不会被发送到内部电路。

15.4 寄存器

（1）UARTB 控制寄存器 0（UBCTL0）

UBCTL0 寄存器用于控制 UARTB 的传输操作。
该寄存器可以按字节或按位进行读取和写入。
系统复位后，该寄存器设为 10H。

注意事项 **1. 使用 UARTB 时，设置 UARTB 功能相关的外部引脚为复用功能模式，设置 UARTB 控制寄存器 2 (UBCTL2)。然后在设置其它位之前，将 UBPWR 位设置为 1。**
2. 设置 UARTB 功能相关的外部引脚为复用功能模式时，请确保向 RXDB 引脚输入高电平。如果输入了低电平，则被判断为在 UBRXE 位设置为 1 之后输入了一个下降沿，可能会启动接收。

备注 接收禁止时，接收移位寄存器不检测起始位。不执行向接收数据寄存器的移入处理或传输处理，接收数据寄存器的内容被保留。
接收使能时，与起始位的检测同步，启动接收移位操作，当一帧的接收完成后，接收移位寄存器的内容被传输到接收数据寄存器。
在传输到接收数据寄存器 (在 FIFO 模式下，达到接收数据设置数量而触发传输)的同时，也产生一个接收结束中断请求信号(INTUBTIR)。
在经过了 UBFIC1.UBTC4 至 UBFIC1.UBTC0 位指定的下一数据接收等待时间后，下一个数据也没有到来(没有检测到开始位)，如果数据存储在接收 FIFO，则产生一个接收超时中断请求信号(INTUBTITO)。

（1/2）

复位后： 10H　　R/W　　地址：UB0CTL0 FFFFFB80H, UB1CTL0 FFFFFBA0H

	<7>	<6>	<5>	<4>	3	2	1	0
UBnCTL0 (n = 1, 0)	UBnPWR	UBnTXE	UBnRXE	UBnDIR	UBnPS1	UBnPS0	UBnCL	UBnSL

UBnPWR	控制 UARTBn的操作时钟
0	停止向UARTBn提供时钟
1	向UARTBn提供时钟

· 当UBnPWR 位清除为0时，UARTBn可以被异步复位。
· 当UBnPWR 位 = 0时，UARTBn 处于复位状态。因此，要操作UARTBn，UBnPWR 位必须设置为1。
· 当UBnPWR位从1改变为0时，UARTBn的所有寄存器被初始化。当UBnPWR位被再次设置为1时，UARTBn 寄存器必须重新设置。
· 当UBnPWR位清除为0时，TXDBn引脚输出一个高电平。

UBnTXE	发送使能
0	发送禁止
1	发送使能

· 在启动时，将UBnPWR 位设置为1，然后将UBnTXE位设置为1。要停止发送，将UBnTXE位清除为0，然后将UBnPWR 位清除为0。
· 当发送单元状态被初始化时，发送状态可能无法被初始化，除非在UBnTXE位清除为0，经过fxx 两个周期的间隔之后，将 UBnTXE 位设置为1。

UBnRXE	接收使能
0	接收禁止
1	接收使能

· 在启动时，将UBnPWR 位设置为1，然后将UBnRXE位设置为1。要停止发送，将UBnRXE位清除为0，然后将UBnPWR 位清除为0。
· 当发送单元状态被初始化时，发送状态可能无法初始化，除非在UBnRXE位被清除为0，经过fxx 两个周期的间隔之后，将 UBnRXE 位设置为1。

（2/2）

UBDIR	传输方向模式的规范 (MSB/LSB)
0	首先传输MSB
1	首先传输LSB

· 在改变UBDIR位的设置之前，将UBPWR位或 UBTXE 和 UBRXE 位清除为0。

UBPS1	UBPS0	发送期间的校验选择	接收期间的校验选择
0	0	不输出校验位	无校验接收
0	1	输出0 校验	0校验接收
1	0	输出奇校验	作为奇校验判断
1	1	输出偶校验	作为偶校验判断

· 在重写UBPS1 和 UBPS0位之前，将 UBTXE 和UBRXE位清除为0。
· 如果接收选择 “0 校验”，则不做校验判断。因此不产生错误中断，因为 UBSTR.UBPE位不被设置为1。

UBCL	1-帧发送/接收数据的字符长度规范
0	7 位
1	8 位

在重写UBCL位之前，将UBTXE 和 UBRXE 位清除为0。

UBSL	发送数据停止位长度的规范
0	1 位
1	2 位

· 在重写UBSL位之前，将UBTXE位清除为0。
· 由于接收的操作总是使用单个停止位长度，UBSL位的设置不影响接收操作。

备注 关于奇偶校验的详情，参见 **15.7.6 奇偶类型和对应操作。**

（2）UARTB 状态寄存器 （UBSTR）

UBSTR 寄存器表示 UARTB 在发送数据时传输状态和接收错误的内容。

状态寄存器表示发送数据时的传输状态，表示发送移位寄存器和发送数据寄存器（在单次模式下的 UBTX 寄存器或 FIFO 模式下的发送 FIFO）的数据保留状态。表示接收错误的状态标志保持它的状态，直到清除为 0。

该寄存器可以按字节或按位进行读取和写入。

系统复位后，该寄存器设为 00H。

注意事项 当 UBCTL0.UBPWR 位或 UBCTL0.UBRXE 位设置为 0 时，或当 0 写入 UBSTR 寄存器时，UBSTR.UBOVF，UBSTR.UBPE，UBSTR.UBFE 和 UBSTR.UBOVE 位清除为 0。

(1/2)

复位后：00H　R/W　地址：FFFFFA44H

	<7>	6	5	4	3	<2>	<1>	<0>
UBSTR	UBTSF	0	0	0	UBOVF	UBPE	UBFE	UBOVE

UBTSF	传输状态标志
0	· 在单次模式下 (UBFIC0.UBMOD 位 = 0) 要传输到发送移位寄存器和UBTX寄存器的数据不存在 (当 UBCTL0.UBPWR 位 = 0 或UBCTL0.UBTXE 位 = 0时被清除(0))。 · 在FIFO模式下 (UBFIC0.UBMOD 位 = 1) 要传输到发送移位寄存器和发送FIFO的数据不存在 (当UBCTL0.UBPWR 位 = 0 或 UBCTL0.UBTXE 位 = 0时被清除(0))。
1	· 在单次模式下(UBFIC0.UBMOD 位 = 0) 要传输到发送移位寄存器和UBTX寄存器的数据存在 (发送进行中)。 · 在FIFO模式下(UBFIC0.UBMOD 位 = 1) 要传输到发送移位寄存器和发送FIFO的数据存在 (发送进行中)。

发送数据被写入UBTX 寄存器之后，经过 f_{XX} 的两个周期，UBTSF 位的值被反映。
因此，发送数据被写入UBTX 寄存器之后，参考UBTSF 位时要特别注意。

UBOVF	溢出标志
0	未发生溢出
1	已发生溢出 (接收期间)

· 只有在FIFO模式下 (当 UBFIC0.UBMOD 位 = 1)，UBOVF 位有效。在单次模式下 (当 UBFIC0.UBMOD 位 = 0)无效。
· 如果发生一次溢出，接收数据不被写入接收FIFO，而是被丢弃。

(2/2)

UBPE	校验错误标志
0	未发生校验错误
1	已发生校验错误 (接收期间)

· 只有在单次模式下 (当 UBFIC0.UBMOD 位= 0)UBPE 位有效。在FIFO模式下 (当 UBFIC0.UBMOD 位= 1)无效。
· 根据UBCTL0.UBPS1 和 UBCTL0.UBPS0 位的设置，UBPE 位的操作不同。

UBFE	帧错误标志
0	未发生帧错误
1	已发生帧错误 (接收期间)

· 只有在单次模式下 (当 UBFIC0.UBMOD 位= 0)UBFE 位有效。在FIFO模式下 (当 UBFIC0.UBMOD 位= 1)无效。
· 只检查接收数据停止位的第一位，不论停止位长度如何。

UBOVE	过载错误标志
0	未发生过载错误
1	已发生过载错误 (接收期间)

· 只有在单次模式下 (当 UBFIC0.UBMOD 位= 0)UBOVE 位有效。在FIFO模式下 (当 UBFIC0.UBMOD 位= 1)无效。
· 当发生一个过载错误时，下一个接收数据值不被写入UBRX 寄存器，而是被丢弃。

(3) UARTB 控制寄存器 2 (UBCTL2)

UBCTL2 寄存器用于指定分频比率，并通过它来控制 UARTB 的波特率（串行传输速度）。

该寄存器可以按 16 位宽度进行读取和写入。

系统复位后，该寄存器设为 FFFFH。

注意事项　当写入该寄存器的 UBBRS15 至 UBBRS0 位时，将 UBCTL0.UBTXE 和 UBCTL0.UBRXE 位设置为 0，或将 UBCTL0.UBPWR 位清除为 0。

复位后：FFFFH　　R/W　　地址：FFFFFA42H

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
UBCTL2	UB BRS 15	UB BRS 14	UB BRS 13	UB BRS 12	UB BRS 11	UB BRS 10	UB BRS 9	UB BRS 8	UB BRS 7	UB BRS 6	UB BRS 5	UB BRS 4	UB BRS 3	UB BRS 2	UB BRS 1	UB BRS 0

备注　关于 UBBRS15 至 UBBRS0 位的详情，参见 **表 15-2　16 位计数器的分频值**。

表 15-2　16 位计数器的分频值

UB BRS 15	UB BRS 14	UB BRS 13	UB BRS 12	UB BRS 11	UB BRS 10	UB BRS 9	UB BRS 8	UB BRS 7	UB BRS 6	UB BRS 5	UB BRS 4	UB BRS 3	UB BRS 2	UB BRS 1	UB BRS 0	k	选择的输出时钟
0	0	0	0	0	0	0	0	0	0	0	0	0	0	x	x	4	f_{XX}/k
0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	4	f_{XX}/k
0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	5	f_{XX}/k
0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	6	f_{XX}/k
• • •	• • •	• • •	• • •	• • •	• • •	• • •	• • •	• • •	• • •	• • •	• • •	• • •	• • •	• • •	• • •	• • •	• • •
1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	65532	f_{XX}/k
1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	65533	f_{XX}/k
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	65534	f_{XX}/k
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	65535	f_{XX}/k

备注
1. f_{XX}：外围时钟
2. k：UBCTL2.UBBRS15 至 UBCTL2.UBBRS0 位设置的值 (k = 4，5，6，…，65535)
3. x：无需理会

(4) **UARTB 发送数据寄存器(UBTX)**

UBTX 寄存器用于设置发送数据。在单次模式下(UBFIC0.UBMOD 位 = 0)，使用 8 位 × 1 级 UBTX 寄存器。在 FIFO 模式下(UBFIC0.UBMOD 位 = 1)，使用 8 位 × 16 级发送 FIFO。

在单次模式下，当发送使能 (UBCTL0.UBTXE 位 = 1)时，通过将发送数据写入 UBTX 寄存器，可以启动发送。当数据可以写入 UBTX 寄存器时（当一个字节的数据从 UBTX 寄存器传输到发送移位寄存器时），产生一个发送使能中断请求信号(INTUBTIT)。

在 FIFO 模式下，将不少于指定数量但少于 16 字节的发送数据写入到发送 FIFO，然后使能发送（UBTXE 位 = 1），可以启动发送，发送数据的指定数量由 UBFIC2.UBTT3 至 UBFIC2.UBTT0 位设置为触发器。当由 UBFIC2.UBTT3 至 UBFIC2.UBTT0 位设置为触发器的指定数量的发送数据从发送 FIFO 传输到发送移位寄存器（设置为触发器的指定数量的发送数据可以写入）时，产生一个发送使能中断请求信号(INTUBTIT)。在 FIFO 模式下，如果在发送 FIFO 和发送移位寄存器内没有数据（当 FIFO 和该寄存器为空时），则产生一个 FIFO 发送使能中断请求信号(INTUBTIF)。

中断的产生时序，参见 **15.5 中断请求信号。**

以 LSB 先行方式发送出 7 位长度数据时，从 LSB（第 0 位）开始，发送数据寄存器的第 6 位至第 0 位作为发送数据被发出，MSB（第 7 位）总是 0。以 MSB 先行方式发送数据时，从 MSB（第 7 位）开始，发送数据寄存器的第 7 位至第 1 位作为发送数据发出，LSB（第 0 位）总是 0。

该寄存器只能按字节进行写入。数据写入到发送数据寄存器。

系统复位后，该寄存器设为 FFH。

复位后：FFH　　W　　地址：　FFFFFA48H

	7	6	5	4	3	2	1	0
UBTX	UBTD7	UBTD6	UBTD5	UBTD4	UBTD3	UBTD2	UBTD1	UBTD0

(5) **UARTB 接收数据寄存器 AP (UBRXAP)，UARTB 接收数据寄存器(UBRX)**

这些寄存器存储接收移位寄存器转换后的并行数据。在单次模式下(UBFIC0.UBMOD 位 = 0)，作为 8 位 × 1 级 UBRX 寄存器。在 FIFO 模式下(UBFIC0.UBMOD 位 = 1)，使用 16 位 × 16 级接收 FIFO (UBRXAP 寄存器)。

接收数据存储在接收 FIFO (UBRXAP 寄存器)的低 8 位，接收数据的错误信号存储在高 8 位(第 8 位和第 9 位)。如果在 FIFO 模式下发生一个接收错误(比如奇偶校验错误或帧错误)，按照 16 位单元宽度（半字）读取 UBRXAP 寄存器。用这种方法可以检查存储在接收 FIFO 中的数据的标志(错误信息被附加，UBPEF 位 = 1 或 UBFEF 位 = 1)，可以识别错误数据（当 UBRXAP 寄存器的低 8 位按照 8 位单元（字节）读取时，高 8 位被丢弃。因此，如果没有发生错误，只有按照 8 位单元（字节）读取，才可以连续读取 UBRXAP 寄存器的接收数据，和读取 UBRX 寄存器的方式相同）。

当接收使能时 (UBCTL0.UBRXE 位 = 1)，在一帧的移入处理的完成的同时，接收到的数据从接收移位寄存器传输到接收数据寄存器。

在单次模式下，通过将数据传输到 UBRX 寄存器，或者在 FIFO 模式下，将指定数量的接收数据传输到接收 FIFO，都会产生一个接收结束中断请求信号(INTUBTIR)，接收数据的数量由 UBFIC2.UBRT3 至 UBFIC2.UBRT0 位设置为触发器。在 FIFO 模式下，在经过了 UBFIC1.UBTC4 至 UBFIC1.UBTC0 位指定的下一数据接收等待时间后，下一个数据也没有到来(没有检测到开始位)时，如果数据存储在接收 FIFO，则产生一个接收超时中断请求信号(INTUBTITO)。

这些中断请求的产生时序信息，参见 **15.5　中断请求信号。**

以 LSB 先行方式接收数据，且数据长度指定为 7 位时，接收到的数据从 LSB（第 0 位）开始，被传输到接收数据寄存器的第 6 位至第 0 位，MSB（第 7 位）总是 0。以 MSB 先行方式接收数据时，接收到的数据从 MSB（第 7 位）开始，被传输到接收数据寄存器的第 7 位至第 1 位，LSB（第 0 位）总是 0。但是，如果发生过载错误，此时的接收数据不传输到接收数据寄存器。

UBRXAP 寄存器只能按照 16 位宽度来读取。但是 UBRXAP 寄存器的低 8 位只能按字节来读取。

UBRX 寄存器只能按字节来读取。

在复位输入之外，通过将 UBCTL0.UBPWR 位清除为 0，在单次模式下，这些寄存器的值设置为 FFH，或在 FIFO 模式下，被设置为 00FFH。

注意事项 **1.** **不能读取 UBPEF 和 UBFEF 位，因为这些寄存器在单次模式下被用作 8 位寄存器。**

2. **在 FIFO 模式下，没有发生接收错误时，通过按 8 位（字节）单元读取 UBRXAP 寄存器的低 8 位，UBRXAP 寄存器的接收数据可以连续读取。对高 8 位的 8 位访问被禁止。如果访问，则操作无法保证。**

注意事项 **3.** 调试一个使用单次模式的系统时，不要执行下列操作。

- 紧跟在读取 **UBRX** 寄存器之后的指令处设置断点。
- 在 **DMA** 传输结束之前设置断点，**UBRX** 寄存器在 **DMA** 传输中被指定为传输源。
- 在数据的接收和读取 **UBRX** 寄存器之后，在下一个数据的接收结束之前设置断点，并在调试器的 **I/O** 寄存器窗口检查 **UBRX** 寄存器。

如果执行了上述任何一个操作，在后续的接收期间可能会发生过载错误。

复位后：00FFH　R　地址：FFFFFA46H

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
UBRXAP	0	0	0	0	0	0	UBPEF	UBFEF	UBRD7	UBRD6	UBRD5	UBRD4	UBRD3	UBRD2	UBRD1	UBRD0

复位后：FFH　R　地址：FFFFFA46H

	7	6	5	4	3	2	1	0
UBRX	UBRD7	UBRD6	UBRD5	UBRD4	UBRD3	UBRD2	UBRD1	UBRD0

UBPEF	校验错误标志
0	未发生校验错误
1	已发生校验错误 (接收期间)

· 只有在FIFO模式下 (当 UBFIC0.UBMOD 位= 1)UBPEF 位有效。在单次模式下 (当 UBFIC0.UBMOD 位= 0)无效。
· 根据UBCTL0.UBPS1 和 UBCTL0.UBPS0 位的设置，UBPEF 位的操作不同。

UBFEF	帧错误标志
0	未发生帧错误
1	已发生帧错误 (接收期间)

· 只有在FIFO模式下 (当 UBFIC0.UBMOD 位= 1)UBFEF 位有效。在单次模式下 (当 UBFIC0.UBMOD 位= 0)无效。
· 只检查接收数据停止位的第一位，不论停止位长度如何。

UBRD7 至 UBRD0	存储接收数据

(6) UARTB FIFO 控制寄存器 0 (UBFIC0)

该寄存器用于选择 UARTB 的操作模式，并选择在 FIFO 模式下(UBMOD 位 = 1)有效的功能。在 FIFO 模式下，可以清除发送 FIFO/接收 FIFO，并指定时序模式，按照该时序产生发送使能中断请求信号(INTUBTIT)/接收结束中断请求信号(INTUBTIR)。

该寄存器可以按字节或按位进行读取和写入。

系统复位后，该寄存器设为 00H。

(1/2)

复位后：00H　　R/W　　地址：　FFFFFA4AH

	7	6	5	4	3	2	1	0
UBFIC0	UBMOD	0	0	0	UBTFC	UBRFC	UBITM	UBIRM

UBMOD	UARTB 操作模式的规范
0	单次模式
1	FIFO 模式

UBTFC	发送FIFO清除触发位
0	正常状态
1	清除(发送FIFO清除之后，该位自动返回为0)

- 只有在FIFO模式下(UBMOD 位= 1)UBTFC 位有效。在单次模式下(UBMOD 位= 0)该位无效。
- 向UBTFC 位写入1时，发送FIFO的指针清除为0。在挂起模式下(UBITM 位= 0)，挂起的中断请求信号 (INTUBTIT) 被清除[注]。但是，中断控制寄存器(UTIC) 的第7位 (UTIF) 不清除为0。必需将该位清除为0。
  向UBTFC 位写入0时，状态保持。不执行操作，比如清除或设置。
- 向UBTFC 位写入1时，请确保将UBCTL0.UBTXE 位清除为0 (禁止发送)。如果当UBTXE 位是1时 (发送使能)，1被写入到UBTFC 位，操作无法保证。

注　发送 FIFO 清除之后(UBTFC 位 = 1)，在 f_{XX} 的四个周期内，或直到通过读取 UBFIC0 寄存器确认 UBTFC 位清除 (自动恢复)，禁止访问这些和 UARTB 相关的寄存器。如果访问了这些寄存器，则操作无法保证。

备注　f_{XX}：外围时钟

(2/2)

UBRFC	接收FIFO (UBRXAP) 清除触发位
0	正常状态
1	清除(接收FIFO清除之后，该位自动返回为0)

· 只有在FIFO模式下(UBMOD 位= 1)UBRFC 位有效。在单次模式下(UBMOD 位= 0)该位无效。
· 向UBRFC 位写入1时，接收FIFO的指针清除为0。在挂起模式下(UBIRM 位= 0)，挂起的中断请求信号 (INTUBTIR) 被清除[注]。但是，中断控制寄存器(URIC) 的第7位 (URIF) 不清除为0。必需将该位清除为0。
向UBRFC 位写入0时，状态保持。不执行操作，比如清除或设置。
· 向UBRFC 位写入1时，请确保将UBCTL0.UBRXE 位清除为0 (禁止接收)。如果当UBRXE 位为1时 (接收使能)，1被写入到UBRFC 位，操作无法保证。

UBITM	在FIFO模式下 INTUBTIT 中断产生时序的规范
0	挂起模式
1	指针模式

在FIFO模式下，由UBFIC2.UBTT3 至 UBFIC2.UBTT0 位设置作为触发器的指定数量的发送数据从发送FIFO传输到发送移位寄存器，则马上产生INTUBTIT信号。 产生INTUBTIT 信号请求之后， 根据挂起模式或指针模式指定实际产生INTUBTIT 信号的时序。详情参见15.6 (2) 挂起模式/指针模式。

UBIRM	在FIFO模式下 INTUBTIR中断产生时序的规范
0	挂起模式
1	指针模式

在FIFO模式下，由UBFIC2.UBRT3 至UBFIC2.UBRT0 位设置作为触发器的指定数量的接收数据从接收移位寄存器传输到接收FIFO，则马上产生INTUBTIR 信号。产生INTUBTIR信号请求之后， 根据挂起模式或指针模式指定实际产生INTUBTIR信号的时序。详情参见15.6 (2) 挂起模式/指针模式。

注　接收 FIFO(UBRXAP) 清除之后(UBRFC 位 = 1)，在 fxx 的四个周期内，或直到通过读取 UBFIC0 寄存器确认 UBRFC 位清除 (自动恢复)，禁止访问这些和 UARTB 相关的寄存器。如果访问了这些寄存器，则操作无法保证。

备注　fxx：外围时钟

(7) UARTB FIFO 控制寄存器 1 (UBFIC1)

在 FIFO 模式下(UBFIC0.UBMOD 位 = 1)，UBFIC1 寄存器有效。在接收到停止位之后，即使经过了 UBTC4 至 UBTC0 位设置的等待时间，下一个数据也没有到来(没有检测到开始位)时，如果数据存储在接收 FIFO，产生一个接收超时中断请求信号(INTUBTITO)。

该寄存器可以按字节或按位进行读取和写入。

系统复位后，该寄存器设为 00H。

复位后： 00H　　R/W　　地址：FFFFFA4BH

	7	6	5	4	3	2	1	0
UBFIC1	UBTCE	0	0	UBTC4	UBTC3	UBTC2	UBTC1	UBTC0

UBTCE	超时计数器功能禁止/使能的规范
0	禁止超时计数器功能的使用
1	使能超时计数器功能的使用

UBTC4	UBTC3	UBTC2	UBTC1	UBTC0	下一个数据接收等待时间
0	0	0	0	0	32 字节 (32 × 8/波特率)
0	0	0	0	1	31 字节 (31 × 8/波特率)
0	0	0	1	0	30 字节 (30 × 8/波特率)
0	0	0	1	1	29 字节 (29 × 8/波特率)
⋮	⋮	⋮	⋮	⋮	⋮
1	1	1	0	0	4 字节 (4 × 8/波特率)
1	1	1	0	1	3 字节 (3 × 8/波特率)
1	1	1	1	0	2 字节 (2 × 8/波特率)
1	1	1	1	1	1 字节 (1 × 8/波特率)

由UBTC4 至UBTC0位设置的接收等待时间向上计数完成时，超时计数器的计数值被清除为0，不论接收FIFO内的数据存储状态如何。之后检测到下一个起始位时，从该数据的停止位再次开始计数。

(8) UARTB FIFO 控制寄存器 2 (UBFIC2)

在 FIFO 模式下(UBFIC0.UBMOD 位 = 1)，UBFIC2 寄存器有效。用于设置产生中断的时序，使用发送/接收数据的数量作为触发。当数据发送时，从发送 FIFO 传输的数据的数量被指定作为产生该中断的条件。当接收到数据时，存储在接收 FIFO 中的数据的数量被指定作为该中断的产生条件。

该寄存器可以按照 16 位宽度来读取或写入。

当 UBFIC2 寄存器的高 8 位可以用作 UBFIC2H 寄存器，低 8 位可以用作 UBFIC2L 寄存器时，这些寄存器可以按字节来读取或写入。

复位输入会将 UBFIC2 寄存器设置为 0000H，并将 UBFIC2H 和 UBFIC2L 寄存器设置为 00H。

注意事项　**在将数据写入 UBFIC2 寄存器之前，请确保将 UBCTL0.UBTXE 位 (禁止发送) 和 UBCTL0.UBRXE 位 (禁止接收)设置为 0。如果当 UBTXE 或 UBRXE 位设置为 1 时，将数据写入 UBFIC2 寄存器，操作无法保证。**

(1/2)

复位后：0000H　　R/W　　地址：FFFFFA4CH

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
UBFIC2	0	0	0	0	UBTT3	UBTT2	UBTT1	UBTT0	0	0	0	0	UBRT3	UBRT2	UBRT1	UBRT0

UBTT3	UBTT2	UBTT1	UBTT0	被设置作为触发器的发送 FIFO的数据数量	指针模式	挂起模式
0	0	0	0	1 字节	可以设置	可以设置
0	0	0	1	2 字节	禁止设置	
0	0	1	0	3 字节		
0	0	1	1	4 字节		
0	1	0	0	5 字节		
0	1	0	1	6 字节		
0	1	1	0	7 字节		
0	1	1	1	8 字节		
1	0	0	0	9 字节		
1	0	0	1	10 字节		
1	0	1	0	11 字节		
1	0	1	1	12 字节		
1	1	0	0	13 字节		
1	1	0	1	14 字节		
1	1	1	0	15 字节		
1	1	1	1	16 字节		

· 被设置作为触发器的发送FIFO的发送数据数量。

· 每当指定数量的数据从发送FIFO移出到发送移位寄存器时，产生INTUBTIT信号。在挂起模式下 (UBFIC0.UBITM 位 = 0)，在挂起模式的条件下产生INTUBTIT信号。

· 在指针模式下 (UBFIC0.UBITM 位 = 1)，被设置作为触发器的发送数据数量只能是1个字节 (UBTT3 至 UBTT0 位 = 0000)，其它设置被禁止。如果做出了1字节之外的设置，操作无法保证。

(2/2)

UBRT3	UBRT2	UBRT1	UBRT0	被设置作为触发器的接收 FIFO的数据数量	指针模式	挂起模式
0	0	0	0	1 字节	可以设置	可以设置
0	0	0	1	2 字节	禁止设置	
0	0	1	0	3 字节		
0	0	1	1	4 字节		
0	1	0	0	5 字节		
0	1	0	1	6 字节		
0	1	1	0	7 字节		
0	1	1	1	8 字节		
1	0	0	0	9 字节		
1	0	0	1	10 字节		
1	0	1	0	11 字节		
1	0	1	1	12 字节		
1	1	0	0	13 字节		
1	1	0	1	14 字节		
1	1	1	0	15 字节		
1	1	1	1	16 字节		

· 被设置作为触发器的接收FIFO的接收数据数量。

· 每当指定数量的数据从接收移位寄存器移出到接收FIFO时，产生INTUBTIR 信号。在挂起模式下 (UBFIC0.UBIRM 位 = 0)，在挂起模式的条件下产生INTUBTIR 信号。

· 在指针模式下 (UBFIC0.UBIRM 位 = 1)，被设置作为触发器的接收数据数量只能是1个字节 (UBRT3 至 UBRT0 位 = 0000)，其它设置被禁止。如果做出了1字节之外的设置，操作无法保证。

(9) UARTB FIFO 状态寄存器 0 (UBFIS0)

在 FIFO 模式下(UBFIC0.UBMOD 位 = 1)，UBFIS0 寄存器有效。用于读取存储在接收 FIFO 内的数据的字节数量。

该寄存器只能按字节进行读取。

系统复位后，该寄存器设为 00H。

复位后：10H　　R　　地址：FFFFFA4FH

	7	6	5	4	3	2	1	0
UBFIS1	0	0	0	UBTB4	UBTB3	UBTB2	UBTB1	UBTB0

UBTB4	UBTB3	UBTB2	UBTB1	UBTB0	发送FIFO 指针
0	0	0	0	0	0 字节
0	0	0	0	1	1 字节
0	0	0	1	0	2 字节
0	0	0	1	1	3 字节
0	0	1	0	0	4 字节
0	0	1	0	1	5 字节
0	0	1	1	0	6 字节
0	0	1	1	1	7 字节
0	1	0	0	0	8 字节
0	1	0	0	1	9 字节
0	1	0	1	0	10 字节
0	1	0	1	1	11 字节
0	1	1	0	0	12 字节
0	1	1	0	1	13 字节
0	1	1	1	0	14 字节
0	1	1	1	1	15 字节
1	0	0	0	0	16 字节
其它情况					无效

表示发送FIFO内的空余字节数量(可以写入的字节)作为发送FIFO指针。

(10) UARTB FIFO 状态寄存器 1 (UBFIS1)

在 FIFO 模式下(UBFIC0.UBMOD 位 = 1)，UBFIS1 寄存器有效。用于读取发送 FIFO 内的空字节数量。
该寄存器只能按字节进行读取。
系统复位后，该寄存器设为 00H。

注意事项　**在发送数据写入 UBTX 寄存器之后经过外围时钟 (fxx) 的两个周期，UBTB4 至 UBTB0 位的值被反映。因此，在发送数据写入 UBTX 寄存器之后参考 UBFIS1 寄存器时，必须特别注意。**

复位后：00H　　R　　地址：FFFFFA4EH

	7	6	5	4	3	2	1	0
UBFIS0	0	0	0	UBRB4	UBRB3	UBRB2	UBRB1	UBRB0

UBRB4	UBRB3	UBRB2	UBRB1	UBRB0	接收FIFO指针
0	0	0	0	0	0 字节
0	0	0	0	1	1 字节
0	0	0	1	0	2 字节
0	0	0	1	1	3 字节
0	0	1	0	0	4 字节
0	0	1	0	1	5 字节
0	0	1	1	0	6 字节
0	0	1	1	1	7 字节
0	1	0	0	0	8 字节
0	1	0	0	1	9 字节
0	1	0	1	0	10 字节
0	1	0	1	1	11 字节
0	1	1	0	0	12 字节
0	1	1	0	1	13 字节
0	1	1	1	0	14 字节
0	1	1	1	1	15 字节
1	0	0	0	0	16 字节
其它情况					无效

表示存储在接收FIFO内的数据的字节数量 (可读取的字节)作为接收FIFO指针。

15.5 中断请求信号

UARTB 会产生以下五种类型的中断请求信号。

- 接收错误中断请求信号(INTUBTIRE)
- 接收结束中断请求信号(INTUBTIR)
- 发送使能中断请求信号(INTUBTIT)
- FIFO 发送结束中断请求信号(INTUBTIF)
- 接收超时中断请求信号(INTUBTITO)

在这五种类型的中断请求中，默认优先级从高到低为，接收错误中断请求信号，接收结束中断请求信号，发送使能中断请求信号，FIFO 发送结束中断请求信号，和接收超时中断请求信号。

为接收完成中断请求信号优先，然后为发送使能中断请求信号。

表 15-3. 产生的中断及默认优先级

中断	优先级
接收错误	1
接收结束	2
发送使能	3
FIFO 发送结束	4
接收超时	5

(1) 接收错误中断请求信号(INTUBTIRE)

(a) 单次模式

当接收使能时，根据三种类型的接收错误(奇偶错误，帧错误，过载错误)的逻辑或（OR）结果，产生一个接收错误中断请求信号，三种错误类型在 UBSTR 寄存器中解释。

当接收禁止时，不产生接收错误中断请求信号。

(b) FIFO 模式

当接收使能时，根据三种类型的接收错误(奇偶错误，帧错误，过载错误)的逻辑或（OR）结果，产生一个接收错误中断请求信号，三种错误类型在 UBSTR 寄存器中解释。

当接收禁止时，不产生接收错误中断请求信号。

(2) 接收结束中断请求信号 **(INTUBTIR)**

(a) 单次模式

当接收使能时，如果数据转移到接收移位寄存器并存储在 UBRX 寄存器(如果接收数据可以读取)，产生一个接收结束中断请求信号。

当接收禁止时，不产生接收结束中断请求信号。

(b) FIFO 模式

当接收使能时，如果数据转移到接收移位寄存器，且 UBFIC2.UBRT3 至 UBFIC2.UBRT0 位设置为触发器的指定数量接收数据传输到接收 FIFO(如果指定数量的接收数据可以读取)，产生一个接收结束中断请求信号。

当接收禁止时，不产生接收结束中断请求信号。

(3) 发送使能中断请求信号**(INTUBTIT)**

(a) 单次模式

如果一帧包括 7 或 8 位字符的发送数据从发送移位寄存器转移出去，且清空 UBTX 寄存器（如果可以写入发送数据），产生发送使能中断请求信号。

(b) FIFO 模式

如果 UBFIC2.UBTT3 至 UBFIC2.UBTT0 位设置为触发器的指定数量发送数据从发送 FIFO 转移到发送移位寄存器（如果可以写入指定数量的发送数据），产生发送使能中断请求信号。

(4) **FIFO** 发送结束中断请求信号 (INTUBTIF)

(a) 单次模式

不能使用。

(b) FIFO 模式

当发送 FIFO 和发送移位寄存器内没有数据（当 FIFO 和寄存器为空）时，产生一个 FIFO 发送使能中断请求信号。在 FIFO 发送使能中断请求信号发生之后，在挂起模式(UBFIC0.UBITM 位 = 0)下，通过清除 FIFO(UBFIC0.UBTFC 位 = 1)对中断请求信号(INTUBTIT)的清除被挂起。

注意事项　如果因为向发送 FIFO 写入下一个发送数据被延迟，产生 FIFO 发送结束中断请求信号(所有发送数据没有发送)，不要清除 FIFO。

(5) 接收超时中断请求信号 (INTUBTITO)

(a) 单次模式

不能使用。

(b) FIFO 模式

当使用超时计数功能(UBFIC1.UBTCE 位 = 1)时，即使经过了 UBFIC1.UBTC4 至 UBFIC1.UBTC0 位指定的下一数据接收等待时间后，下一个数据也没有到来(没有检测到开始位)时，如果数据存储在接收 FIFO，则产生接收超时中断请求信号。

当接收禁止时，不产生接收超时中断请求信号。

如果没有接收到 UBFIC2.UBRT3 至 UBFIC2.UBRT0 位设置为触发器的指定数量的接收数据，可以通过接收超时中断请求信号来设置读取少于设置数量的接收数据的时序，该数量小于指定的数量。

由于在检测到起始位时，超时计数器开始计数，如果没有接收到 1 个字符的数据，不发生接收超时中断请求信号。

15.6 控制模式

(1) 单次模式/FIFO 模式

可以通过 UBFIC0.UBMOD 位来选择单次模式或 FIFO 模式。

(a) 单次模式

- 每个 UBRX 和 UBTX 寄存器包括 8 位 × 1 级。
- 当接收到 1 字节的数据时，产生 INTUBTIR 信号。
- 在产生 INTUBTIR 信号之后，如果在 UBRX 寄存器的接收数据被读取之前，UARTB 的下一次接收操作结束，产生 INTUBTIRE 信号并发生过载错误。

(b) FIFO 模式

- 接收 FIFO (UBRXAP 寄存器) 包括 16 位 × 16 级，发送 FIFO 包括 8 位 × 16 级。
- 只有在发生接收错误（奇偶错误或帧错误）时，通过读取 16 位 UBRXAP 寄存器，接收 FIFO 可以识别错误数据。

(将不少于指定数量但少于 16 字节的发送数据写入到发送 FIFO 之后，当发送使能（UBCTL0.UBTXE 位 = 1）时，可以启动发送，发送数据的指定数量由 UBFIC2.UBTT3 至 UBFIC2.UBTT0 位设置为触发器。

(可以为 INTUBTIT 和 INTUBTIR 信号的产生时序选择挂起模式或指针模式。

(2) 挂起模式/指针模式

在 FIFO 模式下(UBFIC0.UBMOD 位 = 1)，通过 UBFIC0.UBITM 和 UBFIC0.UBIRM 位可以选择挂起模式或指针模式。

如果通过 UBFIC2.UBTT3 至 UBFIC2.UBTT0 位设置为触发器的数量的两倍以上的数据写入发送 FIFO，启动发送，可能会发生一次以上的发送使能中断请求信号(INTUBTIT)。如果 UBFIC2.UBRT3 至 UBFIC2.UBRT0 位设置为触发器的在接收 FIFO 内的接收数据的数量是 8 个字节或更少，可能会发生一次以上的接收结束中断请求信号(INTUBTIR)。在挂起或指针模式下，可以指定在中断被挂起后如何处理。

(a) 挂起模式

(i) 发送期间(写入发送 FIFO)

- 在中断发生之后，如果第一个发送使能中断请求信号(INTUBTIT)的数据没有写入到发送 FIFO，即使满足第二个 INTUBTIT 信号的产生条件(当 UBFIC2.UBTT3 至 UBFIC2.UBTT0 位设置为触发器的指定数量的发送数据从发送 FIFO 被传输到发送移位寄存器时)，也不会发生第二个 INTUBTIT 信号(保持挂起)。

  当第一个INTUBTIT信号对应的数据之后写入到发送FIFO时，产生挂起的INTUBTIT信号[注]。

  注　挂起中断的数量如下。

  当触发器设置为 1 个字节时(UBFIC2.UBTT3 至 UBFIC2.UBTT0 位 = 0000)：最多 15 次
  当触发器设置为 2 个字节时(UBFIC2.UBTT3 至 UBFIC2.UBTT0 位 = 0001)：最多 7 次
  :
  当触发器设置为 6 个字节时(UBFIC2.UBTT3 至 UBFIC2.UBTT0 位 = 0101)：最多 1 次
  当触发器设置为 7 个字节时(UBFIC2.UBTT3 至 UBFIC2.UBTT0 位 = 0110)：最多 1 次
  当触发器设置为 8 个字节时(UBFIC2.UBTT3 至 UBFIC2.UBTT0 位 = 0111)：最多 1 次

- 在挂起模式下，当发生发送使能中断请求信号(INTUBTIT)时，总是向发送 FIFO 写入 UBFIC2.UBTT3 至 UBFIC2.UBTT0 位设置为触发器的指定数量的发送数据。如果数据多于或少于指定数量，禁止将数据写入发送 FIFO。如果写入的数据多于或少于指定数量，操作无法保证。
- 将 UBFIC2.UBTT3 至 UBFIC2.UBTT0 位固定为 0000 (发送数据的设置数量：1 个字节)，通过 DMA 将发送数据写入发送 FIFO。如果有任何其它设置，操作无法保证。

(ii) 接收期间 (从接收 FIFO 读取)

- 如果第一个接收结束中断请求信号(INTUBTIR)对应的数据没有从接收FIFO读取，即使满足第二个INTUBTIR信号的产生条件(如果可以从接收FIFO读取UBFIC2.UBRT3 至UBFIC2.UBRT0 位设置为触发器的指定数量的接收数据)，也不会发生第二个INTUBTIR信号(保持挂起)。当第一个INTUBTIR信号对应的数据之后从接收FIFO读取时，产生挂起的INTUBTIR信号[注]。

注　挂起中断的数量如下。

当触发器设置为 1 个字节时(UBFIC2.UBRT3 至 UBFIC2.UBRT0 位 = 0000)：最多 15 次
当触发器设置为 2 个字节时(UBFIC2.UBRT3 至 UBFIC2.UBRT0 位 = 0001)：最多 7 次
:
当触发器设置为 6 个字节时(UBFIC2.UBRT3 至 UBFIC2.UBRT0 位 = 0101)：最多 1 次
当触发器设置为 7 个字节时(UBFIC2.UBRT3 至 UBFIC2.UBRT0 位 = 0110)：最多 1 次
当触发器设置为 8 个字节时(UBFIC2.UBRT3 至 UBFIC2.UBRT0 位 = 0111)：最多 1 次

- 在挂起模式下，当发生接收结束中断请求信号(INTUBTIR)时，总是从接收 FIFO 读取 UBFIC2.UBRT3 至 UBFIC2.UBRT0 位设置为触发器的指定数量的接收数据。如果数据多于或少于指定数量，禁止从接收 FIFO 读取数据。如果读取的数据多于或少于指定数量，操作无法保证。
- 将 UBFIC2.UBRT3 至 UBFIC2.UBRT0 位固定为 0000 (接收数据的设置数量：1 个字节)，通过 DMA 从接收 FIFO 读取接收数据。如果有任何其它设置，操作无法保证。

(b) 指针模式

(i) 发送期间(写入发送 FIFO)

- 每次当 1 个字节的数据从发送 FIFO 传输到发送移位寄存器，发生一个发送使能中断请求信号(INTUBTIT)。
- 在指针模式下，当发生发送使能中断请求信号(INTUBTIT)时，请确保将 UBFIC2.UBTT3 至 UBFIC2.UBTT0 位固定为 0000 (发送数据的设置数量：1 个字节)作为发送 FIFO 触发器的发送数据的数量。如果有任何其它设置，操作无法保证。
- 禁止通过 DMA 将发送数据写入到发送 FIFO。如果使用 DMA 控制，操作无法保证。
- 在发送使能中断请求信号(INTUBTIT)响应之后，通过参考 UBFIS1 寄存器，发送 FIFO 中空余字节数量的数据可以写入到发送 FIFO。

(ii) 接收期间 (从接收 FIFO 读取)

- 每次当 1 个字节的数据从接收移位寄存器传输到接收 FIFO，发生一个接收结束中断请求信号 (INTUBTIR)。
- 在指针模式下，当发生接收结束中断请求信号(INTUBTIR)时，请确保将 UBFIC2.UBRT3 至 UBFIC2.UBRT0 位固定为 0000 (接收数据的设置数量：1 个字节)作为接收 FIFO 设置为触发器的接收数据的数量。如果有任何其它设置，操作无法保证。
- 禁止通过 DMA 从接受 FIFO 读取数据。如果使用 DMA 控制，操作无法保证。
- 在接收结束中断请求信号(INTUBTIR)响应之后，通过参考 UBFIS0 寄存器，可以从接收 FIFO 中读取存储在接收 FIFO 中字节数量的数据。但是，在某些情况下，即使产生了 INTUBTIR 信号 (UBFIS0.UBRB4 至 UBFIS0.UBRB0 位 = 00000)，数据没有存储在接收 FIFO。在这些情况下，不要从接收 FIFO 读取数据。当存储在接收 FIFO 内的字节数量是 1 个字节或更多(UBRB4 至 UBRB0 位 = 除 00000 之外)时，总是从接收 FIFO 读取数据。

15.7 操作

15.7.1 数据格式

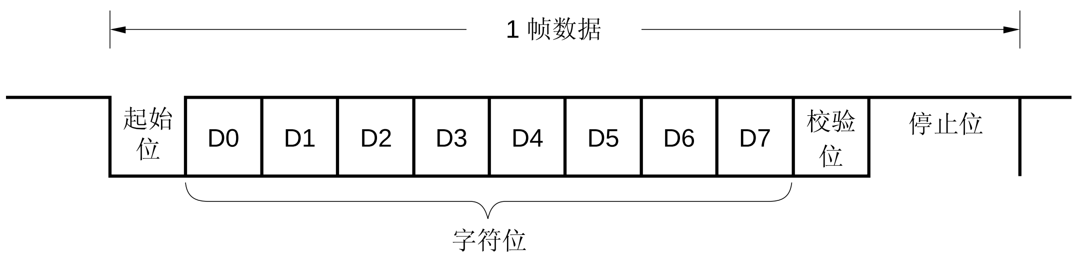
可以执行全双工串行数据的接收和发送。

如图 15-3 所示，发送/接收数据帧的数据由起始位、字符位、校验位以及停止位组成。

通过 UARTB 控制寄存器 0（UBCTL0）指定一个数据帧内的字符位长度、校验类型和停止位长度。

另外，使用 MSB/LSB 先行来传输数据。

图 15-3. 异步串行接口发送/接收数据格式 (LSB 先行传输)



- 起始位 .... 1 位
- 字符位 .... 7 位或 8 位
- 奇偶校验位 .... 奇检验、偶校验、0 校验或无校验
- 停止位 .... 1 位或 2 位

15.7.2 发送操作

在单次模式下(UBFIC0.UBMOD 位 = 0)，当 UBCTL0.UBTXE 位设置为 1 时，使能发送。且当发送数据写入 UBTX 寄存器时，启动发送。

在 FIFO 模式下 (UBFIC0.UBMOD 位 = 1)，将不少于指定数量但少于 16 字节的发送数据写入到发送 FIFO，然后将 UBTXE 位设置为 1，可以启动发送，发送数据的指定数量由 UBFIC2.UBTT3 至 UBFIC2.UBTT0 位设置为触发器。

注意事项　在 FIFO 模式下，将发送数据写入发送 FIFO 之前，禁止将 UBCTL0.UBTXE 位设置为 1。如果有该设置，操作无法保证。

(1) 发送使能状态

该状态由 UBCTL0.UBTXE 位设置。

- UBTXE = 1：发送使能状态
- UBTXE = 0：发送禁止状态

但是，因为 CSIB2 也使用该位，在将 CB2CTL0.CB2PWR 位设置为 0 之后，使能发送。

由于 UARTB 不具有 CTS (发送使能信号)输入引脚，应该使用一个端口来确认目标方是否处于接收使能状态。

(2) 启动一个发送操作

- 在单次模式下 (UBFIC0.UBMOD 位 = 0)
  在单次模式下，发送使能时，当发送数据写入到 UBTX 寄存器，启动发送。
- 在 FIFO 模式下(UBFIC0.UBMOD 位 = 1)
  在 FIFO 模式下，将不少于指定数量但少于 16 字节的发送数据写入到发送 FIFO，然后使能发送(UBTXE 位 = 1)，可以启动发送，发送数据的指定数量由 UBFIC2.UBTT3 至 UBFIC2.UBTT0 位设置为触发器。

当发送启动时，发送数据寄存器中的数据(单次模式下的 UBTX 寄存器，或 FIFO 模式下的发送 FIFO) 传输到发送移位寄存器。然后，发送移位寄存器从 LSB 开始顺序将数据输出到 TXDB 引脚(发送数据从起始位开始顺序传输)。起始位、校验位和停止位自动添加。

(3) 发送中断请求信号

(a) 发送使能中断请求信号 (INTUBTIT)

- **在单次模式下 (UBFIC0.UBMOD 位 = 0)**
  在单次模式下，当发送数据可以写入 UBTX 寄存器时(当 1 个字节的数据从 UBTX 寄存器传输到发送移位寄存器)，发生发送使能中断请求信号 (INTUBTIT)。

- **在 FIFO 模式下(UBFIC0.UBMOD 位 = 1)**
  在 FIFO 模式下，当 UBFIC2.UBTT3 至 UBFIC2.UBTT0 位设置为触发器的指定数量的发送数据从发送 FIFO 传输到发送移位寄存器 (如果可以写入设置为触发器数量的发送数据)时，发生 INTUBTIT 信号。

- **如果在 FIFO 模式下指定为挂起模式 (UBFIC0.UBITM 位 = 0)**
  如果在 FIFO 模式下指定为挂起模式，在发生第一个 INTUBTIT 信号之后，即满足第二个 INTUBTIT 信号的产生条件。第二个 INTUBTIT 信号挂起，直到 UBFIC2.UBTT3 至 UBFIC2.UBTT0 位设置为触发器的指定数量的发送数据写入到发送 FIFO。当第一个 INTUBTIT 信号对应的设置为触发器的指定数量的发送数据写入到发送 FIFO，产生第二个挂起的 INTUBTIT 信号。

- **如果在 FIFO 模式下被指定为指针模式(UBFIC0.UBITM 位 = 1)**
  如果在 FIFO 模式下指定为指针模式，即使当第一个 INTUBTIT 信号发生时，由 UBFIC2.UBTT3 至 UBFIC2.UBTT0 位设置为触发器的指定数量的发送数据没有写入到发送 FIFO，在满足第二个 INTUBTIT 信号的产生条件时，还会发生第二个 INTUBTIT 信号。

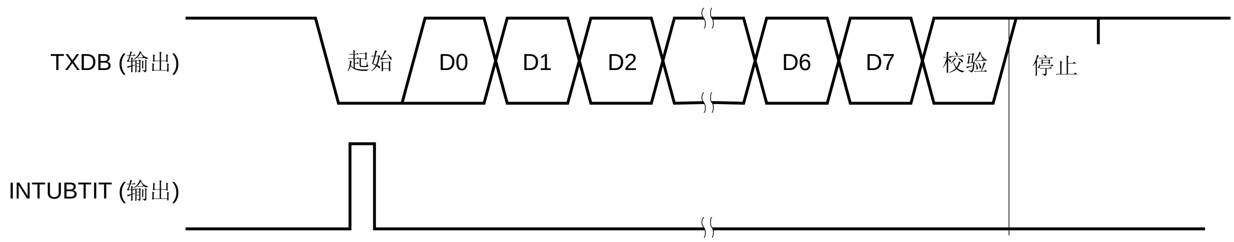
(b) FIFO 发送结束中断请求信号 (INTUBTIF)

在 FIFO 模式下(UBFIC0.UBMOD 位 = 1)，如果在发送 FIFO 和发送移位寄存器内没有数据（当 FIFO 和该寄存器为空时），则产生一个 FIFO 发送结束中断请求信号 (INTUBTIF)。在 INTUBTIF 信号发生之后，通过清除 FIFO (UBFIC0.UBTFC 位 = 1)，来清除在挂起模式下(UBFIC0.UBITM 位 = 0)挂起的 INTUBTIT 信号。如果因为向发送 FIFO 写入下一个发送数据延迟，发生 INTUBTIF 信号(如果没有发送所有发送数据)，不要清除 FIFO

如果下一个待发送的数据还没有写入到发送数据寄存器，发送操作被暂停。

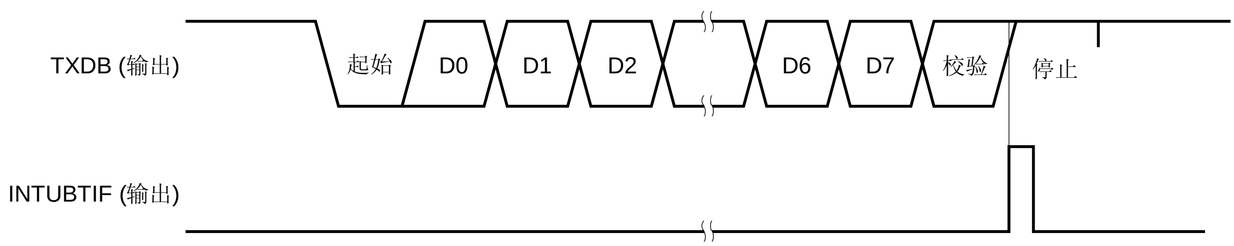
注意事项 **在单次模式下，当 UBTX 寄存器为空时(当 1 个字节的数据从 UBTX 寄存器传输到发送移位寄存器)，发生发送使能中断请求信号 (INTUBTIT)。在 FIFO 模式下，当数据不在发送 FIFO 和发送移位寄存器 (当 FIFO 和寄存器为空)时，发生 FIFO 发送结束中断请求信号 (INTUBTIF)。但是，如果发送数据寄存器是因为 $\overline{\text{RESET}}$ 输入被清空，不产生 INTUBTIT 信号或 INTUBTIF 信号。**

图 15-4. 步串行接口发送使能中断请求信号 (INTUBTIT)的时序



备注　　在 FIFO 模式下，当 UBFIC2.UBTT3 至 UBFIC2.UBTT0 位设置为触发器的指定数量的发送数据被串行传输时，在上述时序发生 INTUBTIT 信号。

图 15-5. 异步串行接口 FIFO 发送结束中断请求信号 (INTUBTIF)的时序



备注　　如果在发送 FIFO 和发送移位寄存器内没有数据（当 FIFO 和该寄存器为空）时，在上述时序发生 INTUBTIF 信号。

15.7.3 连续发送操作

- **在单次模式下 (UBFIC0.UBMOD 位 = 0)**

  在单次模式下，当发送移位寄存器启动一个移位操作，下一个数据马上可以写入到 UBTX 寄存器。传输的时序可以通过发送使能中断请求信号 (INTUBTIT)来识别。借助 INTUBTIT 信号，在一个数据帧发送时期内将下一个发送数据写入到 UBTX 寄存器，数据可以无间隔发送，实现有效的通信速率。

  注意事项 在发送处理期间，执行初始化之前，确认 UBSTR.UBTSF 位为 0。如果在 UBTSF 位为 1 时执行初始化，发送数据无法保证。

- **如果在 FIFO 模式下被指定为挂起模式 (UBFIC0.UBITM 位 = 0)**

  如果将不少于指定数量但少于 16 字节的发送数据写入到发送 FIFO，可以启动发送，发送数据的指定数量由 UBFIC2.UBTT3 至 UBFIC2.UBTT0 位设置为触发器。

  如果在 FIFO 模式下指定为挂起模式，发送移位寄存器开始对指定数量的最后一个数据移位时，马上可以将 UBFIC2.UBTT3 至 UBFIC2.UBTT0 位设置为触发器的指定数量的下一批发送数据写入到发送 FIFO。传输的时序可以通过 INTUBTIT 信号来识别。通过设置为触发器的指定数量的下一批发送数据写入到发送 FIFO，或借助 INTUBTIT 信号，在发送 FIFO 内的数据发送期间将数据写入到 FIFO，数据可以无间隔发送，实现有效的通信速率。

  注意事项 在发送处理期间，执行初始化之前，确认 UBSTR.UBTSF 位为 0 (也可以通过 FIFO 发送结束中断请求信号 (INTUBTIF)来完成)。如果在 UBTSF 位为 1 时执行初始化，发送数据无法保证。通过 DMA 将发送数据写入发送 FIFO，由 UBFIC2.UBTT3 至 UBFIC2.UBTT0 位设置为触发器的发送数据指定数量是 1 个字节；否则操作无法保证。

- **如果在 FIFO 模式下指定为指针模式 (UBFIC0.UBITM 位 = 1)**

  如果在 FIFO 模式下指定为指针模式，发送移位寄存器开始对设置为触发器的指定数量的发送数据移位，发生 INTUBTIT 信号，且下一个数据马上可以写入发送 FIFO。此时通过参考 UBFIS1 寄存器，可以写入的数据数量等于发送 FIFO 内空余的字节数量。传输的时序可以通过 INTUBTIT 信号来识别。通过设置为触发器的指定数量的下一批发送数据写入到发送 FIFO，或借助 INTUBTIT 信号，在发送 FIFO 内的数据发送期间将数据写入到 FIFO，数据可以无间隔发送，实现有效的通信速率。

  注意事项 在发送处理期间，执行初始化之前，确认 UBSTR.UBTSF 位为 0 (也可以通过 FIFO 发送结束中断请求信号 (INTUBTIF)来完成)。如果在 UBTSF 位为 1 时执行初始化，发送数据无法保证。

15.7.4 接收操作

设置 UBCTL0.UBPWR 位为 1，然后设置 UBCTL0.UBRXE 位为 1，设置为等待接收状态。开始对 RXDB 引脚采样，并检测到起始位。当检测到起始位时，接收操作开始，根据设置的波特率，数据被依次存储在接收移位寄存器。

在单次模式下(UBFIC0.UBMOD 位 = 0)，每次完成一帧数据的接收时，产生一个接收结束中断请求信号。正常情况下，在该中断服务程序中，接收数据从 UBRX 寄存器传输到存储器。

在 FIFO 模式下(UBFIC0.UBMOD 位 = 1)，将指定数量的接收数据传输到接收 FIFO，会产生 INTUBTIR 信号，接收数据的数量由 UBFIC2.UBRT3 至 UBFIC2.UBRT0 位设置为触发器。

如果在 FIFO 模式下指定为挂起模式(UBFIC0.UBIRM 位 = 0)，可以从接收 FIFO 读取指定数量的接收数据，读取数据的数量由 UBFIC2.UBRT3 至 UBFIC2.UBRT0 位设置为触发器。

如果在 FIFO 模式下指定为指针模式(UBFIC0.UBIRM 位 = 1)，通过参考 UBRT3 至 UBRT0 位设置为触发器的接收数据数量(1 个字节)或参考 UBFIS0 寄存器，可以从接收 FIFO 读取的数据数量是存储在接收 FIFO 内的字节数量(0 字节或更多)。

注意事项 如果在 FIFO 模式下指定为指针模式，如果通过参考 UBFIS0 寄存器，可以从接收 FIFO 读取的数据数量是存储在接收 FIFO 内的字节数量。即使发生接收结束中断请求信号 (INTUBTIR)，没有数据会存储在接收 FIFO 内(UBFIS0.UBRB4 至 UBFIS0.UBRB0 位 = 00000)。这种情况下，不要从接收 FIFO 读取数据。请确认存储在接收 FIFO 内的字节数量 = 1 字节(UBRB4 至 UBRB0 位 = 除 00000 之外)或更多之后，从接收 FIFO 读取数据。

(1) 接收使能状态

该状态由 UBCTL0.UBRXE 位设置。

- UBRXE = 1：接收使能状态
- UBRXE = 0：接收禁止状态

但是，因为 CSIB2 也使用该位，在将 CB2CTL0.CB2PWR 位设置为 0 并禁止 CSIB2 操作之后，使能发送。

在接收禁止状态下，接收硬件处于初始状态。此时，不产生接收结束中断请求信号或接收错误中断请求信号，接收数据寄存器 (单次模式下的 UBRX 寄存器或 FIFO 模式下的接收 FIFO(UBRXAP 寄存器))的内容被保留。

(2) 启动一个接收操作

通过检测到起始位，启动接收操作。

使用 UARTB 控制寄存器 2 (UBCTL2)决定的串行时钟对 RXDB 引脚进行采样。

(3) 接收中断请求信号

(a) 接收结束中断请求信号 (INTUBTIR)

- **在单次模式下 (UBFIC0.UBMOD 位 = 0)**
  在单次模式下，当 UBCTL0.UBRXE 位 = 1 且一帧数据的接收结束（检测到停止位）时，产生一个接收结束中断请求信号 (INTUBTIR)，同时接收移位寄存器内的接收数据被传输到 UBRX 寄存器。
  同样，如果发生过载错误，此时的接收数据不会传输到 UBRX 寄存器，并产生接收错误中断请求信号 (INTUBTIRE)。
  如果在接收操作期间发生奇偶错误或帧错误，接收操作继续，直到接收到停止位。接收完成之后，发生 INTUBTIRE 信号(接收移位寄存器内的接收数据传输到 UBRX 寄存器)。
  如果在接收操作期间 UBRXE 位重置(0)，接收操作马上停止。此时，UBRX 寄存器的内容保持不变，UARTB 状态寄存器 (UBSTR)的内容清除，并且不产生 INTUBTIR 和 INTUBTIRE 信号。
  当 UBRXE 位 = 0 (接收禁止)时，不产生 INTUBTIR 信号。

- **在 FIFO 模式下(UBFIC0.UBMOD 位 = 1)**
  在 FIFO 模式下，当一帧数据已经接收(检测到停止位)时，且当由 UBFIC2.UBRT3 至 UBFIC2.UBRT0 位设置为触发器的指定数量的接收数据从接收移位寄存器传输到接收 FIFO 时，发生接收结束中断请求信号 (INTUBTIR)。如果发生溢出错误，接收数据不传输到接收 FIFO，且发生接收错误中断请求信号 (INTUBTIRE)。
  如果在接收期间发生奇偶错误或帧错误，接收操作继续，直到接收到停止位。收完成之后，发生 INTUBTIRE 信号，且接收移位寄存器内的接收数据传输接收 FIFO。此时，会附加错误信息，UBRXAP.UBPEF 或 UBRXAP.UBFEF 位 = 1。如果发生 INTUBTIRE 信号，通过将接收 FIFO 作为 16 位寄存器 UBRXAP 来读取，可以识别错误数据。

(b) 接收超时中断请求信号 (INTUBTITO) (仅限 FIFO 模式下)

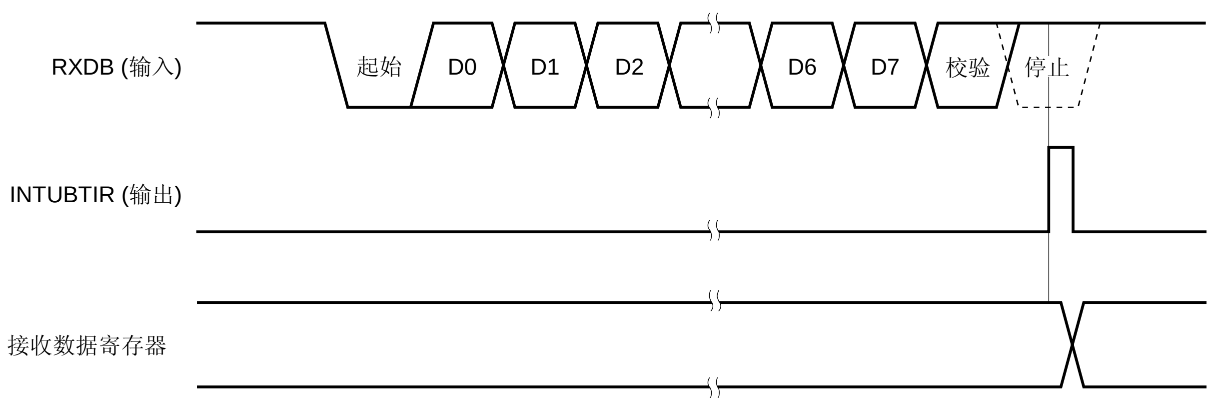
当使用超时计数功能(UBFIC1.UBTCE 位 ＝ 1)时，即使经过了 UBFIC1.UBTC4 至 UBFIC1.UBTC0 位指定的下一数据接收等待时间后，下一个数据也没有到来(没有检测到开始位)时，如果数据存储在接收 FIFO，则产生接收超时中断请求信号。

当接收被禁止时，不产生 INTUBTITO 信号。

如果没有接收到 UBFIC2.UBRT3 至 UBFIC2.UBRT0 位设置为触发器的指定数量的接收数据，可以通过 INTUBTITO 信号来设置读取少于设置数量的接收数据的时序，该数量小于指定的数量。

由于在检测到起始位时，超时计数器开始计数，如果没有接收到 1 个字符的数据，不发生接收超时中断请求信号。

图 15-6. 异步串行接口接收结束中断请求信号 **(INTUBTIR)**的时序



注意事项 **1.** 即使发生了接收错误，请确保读取存储在接收数据寄存器 **(**单次模式下的 **UBRX** 寄存器，或 **FIFO** 模式下的接收 **FIFO(UBRXAP** 寄存器**))**中的所有数据**(**由 **UBFIS0.UBRB4** 至 **UBFIS0.UBRB0** 位指示的数据数量**)**。

除非读取接收数据寄存器，当接收到下一个数据时，发生过载错误，使得接收错误状态保持。

如果在 **FIFO** 模式下指定为挂起模式，但是，请确保在读取接收 **FIFO** 内存储的数据之后，清除 **FIFO (UBFIC0.UBRFC** 位 **= 1)**。

在 **FIFO** 模式下，即使没有读取接收 **FIFO** 内存储的数据，也可以清除 **FIFO**。

如果在 **FIFO** 模式下发生奇偶错误或帧错误，可以按照 **16** 位（半字）单元读取 **UBRXAP** 寄存器。

2. 数据总是按照一个停止位**(1)**来接收。

忽略第二个停止位。

15.7.5 接收错误

在单次模式下(UBFIC0.UBMOD 位 = 0)，在一次接收操作期间，可能发生的错误的三种类型是奇偶错误、帧错误和过载错误。在 FIFO 模式下 (UBFIC0.UBMOD 位 = 1)，在一次接收操作期间，可能发生的错误的三种类型是奇偶错误、帧错误和溢出错误。

作为数据接收的结果，如果在单次模式下发生奇偶错误、帧错误或过载错误，UBSTR.UBPE，UBSTR.UBFE 或 UBSTR.UBOVE 位设置为 1。如果在 FIFO 模式下发生溢出错误，UBSTR.UBOVF 位设置为 1。如果在 FIFO 模式下发生奇偶错误或帧错误，UBRXAP.UBPEF 或 UBRXAP.UBFEF 位设置为 1。同时，发生接收错误中断请求信号(INTUBTIRE)。通过读取 UBSTR 或 UBRXAP 寄存器的内容，可以检测错误的内容。

当 0 写入 UBOVF，UBPE，UBFE 或 UBOVE 位，或 UBCTL0.UBPWR 或 UBCTL0.UBRXE 位时，重置 UBSTR 寄存器的内容。当 0 写入 UBCTL0.UBPWR 位时，重置 UBRXAP 寄存器的内容。

表 15-4. 接收错误原因

错误标志	有效操作模式	错误标志	接收错误	原因
UBPE	单次模式	UBPE	奇偶错误	发送期间的奇偶规范和接收数据的奇偶校验不一致
UBFE		UBFE	帧错误	没有检测到停止位
UBOVE		UBOVE	过载错误	在数据从 UBRX 寄存器读出之前，下一个数据的接收结束
UBOVF	FIFO 模式	UBOVF	溢出错误	接收 FIFO 处于 Full 状态下，数据读取之前，下一个数据的接收结束
UBPEF		UBPEF	奇偶错误	发送期间的奇偶规范 和待接收的数据的奇偶校验不一致
UBFEF		UBFEF	帧错误	当目标数据装载时，没有检测到停止位

15.7.6 奇偶类型和对应的操作

奇偶位用于检测通信数据中的位错误。通常情况下，发送端和接收端应该使用相同类型的奇偶校验位。

（1）偶校验

（a）在发送期间

奇偶校验位的控制是为了让发送数据之中位值为“1”的数量为偶数，包括奇偶校验位。奇偶位的值如下：

(如果发送数据中位的值为“1”的数量是奇数： 1
• 如果发送数据中位的值为“1”的数量是偶数： 0

（b）在接收期间

计算接收数据中位的值为“1”的数量，包括校验位，如果为奇数，则产生奇偶校验错误。

（2）奇校验

（a）在发送期间

与偶校验相反，奇偶校验位的控制是为了让发送数据之中位值为“1”的数量为奇数，包括奇偶校验位。奇偶位的值如下：

(如果发送数据中位的值为“1”的数量是奇数： 0
(如果发送数据中位的值为“1”的数量是偶数： 1

（b）在接收期间

计算接收数据中位的值为“1”的数量，包括校验位，如果为偶数，则产生奇偶校验错误。

（3）0 奇偶校验

在发送期间，奇偶校验位设置为“0”，而与发送数据无关。
在接收期间，不执行奇偶校验检查。因此，无论奇偶位为“0” 或 “1”，均不会产生奇偶错误。

（4）无奇偶校验

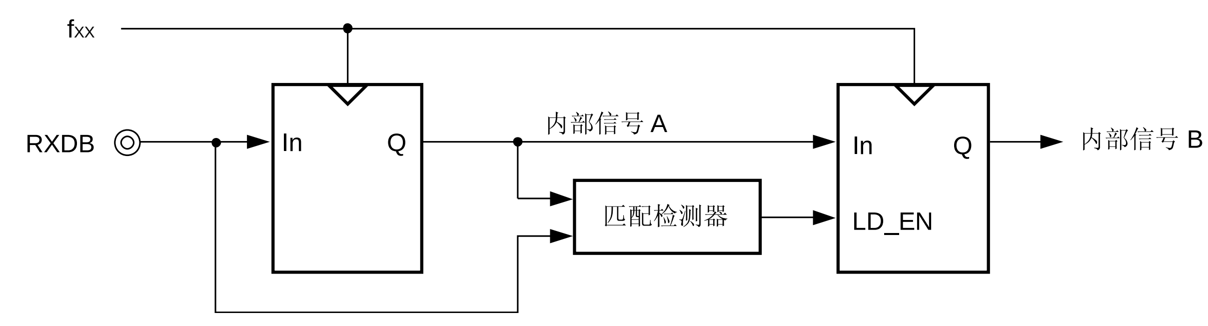
不向发送数据附加奇偶位。
在接收期间，接收的执行认为没有奇偶校验位。因为无奇偶位，所以也不产生奇偶错误。

15.7.7　接收数据噪声过滤器

在外围时钟(fxx)的上升沿处对 RXDB 引脚进行采样。如果两次读到相同的采样值，则一致检测器输出改变，并且该输出采样作为输入数据。因此，不超过一个时钟宽度的数据判定为噪声，不会传送给内部电路（参见 **图 15-8**）。

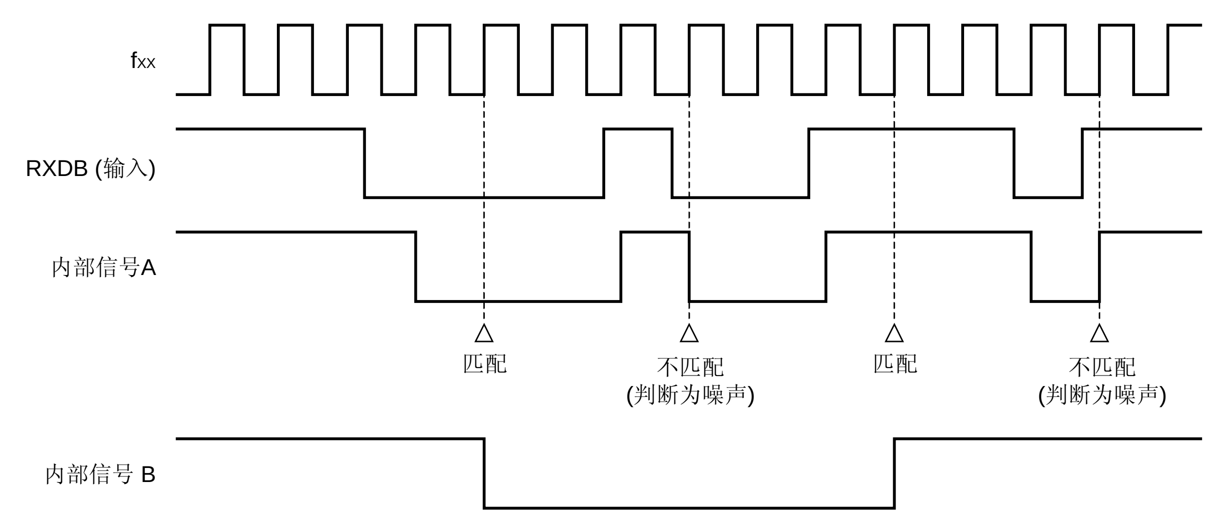
同样，由于电路配置如图 15-7 所示，接收操作期间的内部处理会延迟最多 2 个时钟周期，延迟情况根据外部信号状态。

图 15-7. 噪声过滤电路



备注　fxx：外围时钟

图 15-8. RXDB 信号判定为噪声的时序



备注　fxx：外围时钟

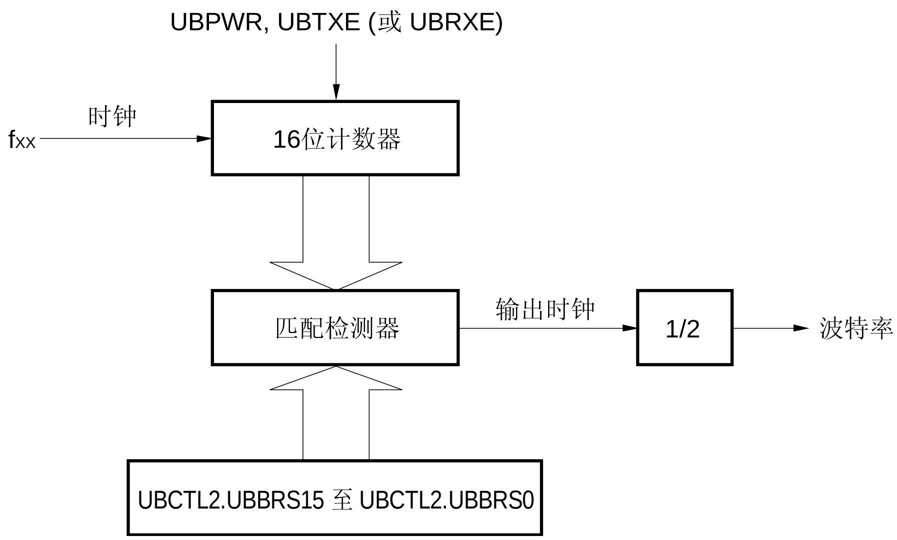
15.8 专用波特率发生器（BRG）

专用波特率发生器包括 16 位可编程计数器，在 UARTB 发送/接收期间产生串行时钟。专用波特率发生器的输出可以选择作为各个通道的串行时钟。

用于发送和接收存在独立的 16 位计数器。同一通道的发送/接收使用的波特率相同。

（1）波特率发生器的配置

图 15-9. 波特率发生器的配置



备注 fxx：外围时钟

（a） 基准时钟

当 UBCTL0.UBPWR 位为 1 时，外围时钟 (fxx)提供发送/接收单元。该时钟称为基准时钟。当 UBPWR 位为 0 时，时钟信号固定为低电平。

（2） 串行时钟的产生

根据 UBCTL2 寄存器的设置，可以产生串行时钟。

16 位计数器的分频值可以通过 UBCTL2.UBBRS15 至 UBCTL2.UBBRS0 位来选择。

（a） 波特率

波特率是根据下列公式计算得到的值。

$$波特率 = \frac{基础时钟频率}{2 \times k} \text{ [bps]}$$

基准时钟频率 = f_{XX}

k = 由 UBCTL2.UBBRS15 至 UBCTL2.UBBRS0 位设置的值 (k = 4, 5, 6, ..., 65535)

（b） 波特率误差

波特率误差是根据下列公式计算得到的值。

$$误差（\%）= \left(\frac{实际波特率（有错误的波特率）}{目标波特率 （正常的波特率）} - 1 \right) \times 100 \text{ [\%]}$$

注意事项 **1.** **发送期间，请确保波特率误差不超过接收端的误差容限。**

2. **接收期间，请确保波特率误差满足一定范围，该范围参见"（4）接收期间允许的波特率误差范围"。**

示例：　基准时钟(f_{XX}) = 64 MHz = 64,000,000 Hz

UBCTL2.UBBRS15 至 UBCTL2.UBBRS0 位的设置 = 0000000001100110B (k = 102)

目标波特率 = 312500 bps

波特率 = 64 M/(2 × 102)
= 64000000/(2 × 102) = 313725 [bps]

误差 = (313725/312500 − 1) × 100
= 0.392 [%]

当基准时钟(f_{XX}) = 60 MHz 并且 k = 96 时，误差是 0% 。

（3）波特率设置举例

表 15-5. 波特率发生器设置数据 (1/2)

波特率 (bps)	f_{XX} = 64 MHz			f_{XX} = 60 MHz			f_{XX} = 50 MHz		
	k (十进制)	k (十六进制)	ERR	k (十进制)	k (十六进制)	ERR	k (十进制)	k (十六进制)	ERR
300	–	–	–	–	–	–	–	–	–
600	53333	D055H	0.001	50000	C350H	0.000	41667	A2C3H	–0.001
1200	26667	682BH	–0.001	25000	61A8H	0.000	20833	5161H	0.002
2400	13333	3415H	0.003	12500	30D4H	0.000	10417	28B1H	–0.003
4800	6667	1A0BH	–0.005	6250	186AH	0.000	5208	1458H	0.006
9600	3333	0D05H	0.010	3125	0C35H	0.000	2604	0A2CH	0.006
19200	1667	0683H	–0.020	1563	061BH	0.000	1302	0516H	0.006
31250	1024	0400H	0.000	960	03C0H	0.000	800	0320H	0.000
38400	833	0341H	0.040	781	030DH	0.000	651	028BH	0.006
76800	417	01A1H	–0.080	391	0187H	0.000	326	0146H	–0.147
153600	208	00D0H	0.160	195	00C3H	0.000	163	00A3H	–0.147
312500	102	0066H	0.392	96	0060H	0.000	80	0050H	0.000
500000	64	0040H	0.000	60	003CH	0.000	50	0032H	0.000
1000000	32	0020H	0.000	30	001EH	0.000	25	0019H	0.000
2000000	16	0010H	0.000	15	000FH	0.000	13	000DH	–3.846
3000000	11	000BH	–3.030	10	000AH	0.000	8	0008H	4.167
4000000	8	0008H	0.000	8	0008H	–6.250	–	–	–
5000000	6	0006H	6.667	6	0006H	0.000	–	–	–
5333333	6	0006H	0.000	–	–	–	–	–	–

注意事项　**外围时钟(f_{XX})的最大允许频率是 64 MHz。**
波特率的最大传输速度是 5.33 Mbps。

备注　f_{XX}：　外围时钟
K：　UBCTL2.UBBRS15 至 UBCTL2.UBBRS0 位的设置
ERR：　波特率误差 （%）

表 **15-5.** 波特率发生器设置数据 **(2/2)**

波特率 (bps)	f_{XX} = 40 MHz			f_{XX} = 32 MHz		
	k (十进制)	k (十六进制)	ERR	k (十进制)	k (十六进制)	ERR
300	–	–	–	53333	D055H	0.001
600	33333	8235H	0.001	26667	682BH	–0.001
1200	16667	411BH	–0.002	13333	3415H	0.003
2400	8333	208DH	0.004	6667	1A0BH	–0.005
4800	4167	1047H	–0.008	3333	0D05H	0.010
9600	2083	0823H	0.016	1667	0683H	–0.020
19200	1042	0412H	–0.032	833	0341H	0.040
31250	640	0280H	0.000	512	0200H	0.000
38400	521	0209H	–0.032	417	01A1H	–0.080
76800	260	0104H	0.160	208	00D0H	0.160
153600	130	0082H	0.160	104	0068H	0.160
312500	64	0040H	0.000	51	0033H	0.392
500000	40	0028H	0.000	32	0020H	0.000
1000000	20	0014H	0.000	16	0010H	0.000
2000000	10	000AH	0.000	8	0008H	0.000
3000000	7	0007H	–4.762	–	–	–
4000000	–	–	–	–	–	–
5000000	–	–	–	–	–	–
5333333	–	–	–	–	–	–

注意事项　**外围时钟(f_{XX})的最大允许频率是 64 MHz。**
波特率的最大传输速度是 5.33 Mbps。

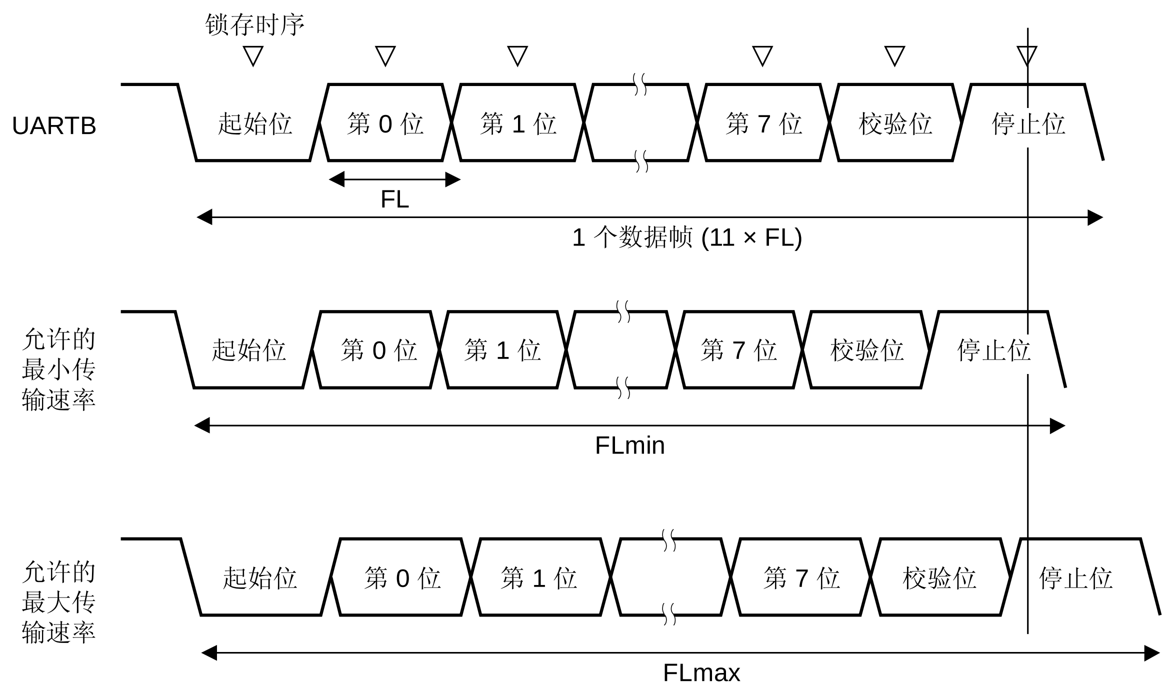
备注　f_{XX}：　外围时钟
K：　UBCTL2.UBBRS15 至 UBCTL2.UBBRS0 位的设置
ERR：　波特率误差（%）

（4）接收期间允许的波特率范围

接收期间，允许从目的端波特率偏离的程度如下所示。

注意事项 接收期间应该使用下述公式来设置波特率错误，使它总是处于允许的误差范围内。

图 15-10. 接收期间允许的波特率范围



备注 n = 0 至 3

如图 15-10 所示，检测到起始位之后，接收数据的锁存时序取决于 UBCTL2 寄存器设置的计数器。如果在此锁存时序内可以及时接收直到最末位数据（停止位），数据就可以正常接收。

当应用于 11 位接收时，以下为理论结果。

FL = （波特率）$^{-1}$

波特率：	UARTB 波特率
k：	UBCTL2 的设定值
FL：	1 位数据长度
锁存时间余量：	2 个时钟

最小允许传输速率： $FLmin = 11 \times FL - \frac{k-2}{2k} \times FL = \frac{21k+2}{2k} FL$

因此，目的端可以接收的最大波特率如下所示。

$$BRmax = (FLmin/11)^{-1} = \frac{22k}{21k + 2} Brate$$

同理，最大允许传输速率的计算如下。

$$\frac{10}{11} \times FLmax = 11 \times FL - \frac{k + 2}{2 \times k} \times FL = \frac{21k - 2}{2 \times k} FL$$

$$FLmax = \frac{21k - 2}{20\ k} FL \times 11$$

因此，目的端可以接收的最小波特率如下所示。

$$BRmin = (FLmax/11)^{-1} = \frac{20k}{21k - 2} Brate$$

UARTB 以及目的端的允许波特率误差可以通过上述公式计算获得，得到的最大和最小波特率值必须遵守下列表格。

表 15-6. 最大和最小的允许波特率误差

分频比（k）	最大允许波特率误差	最小允许波特率误差
4	+2.33 %	–2.44
8	+3.53 %	–3.61
16	+4.14 %	–4.19
32	+4.45 %	–4.48
64	+4.61 %	–4.62
128	+4.68 %	–4.69
256	+4.72 %	–4.73
512	+4.74 %	–4.74
1024	+4.75 %	–4.75
2048	+4.76 %	–4.76
4096	+4.76 %	–4.76
8192	+4.76 %	–4.76
16384	+4.76 %	–4.76
32768	+4.76 %	–4.76
65535	+4.76 %	–4.76

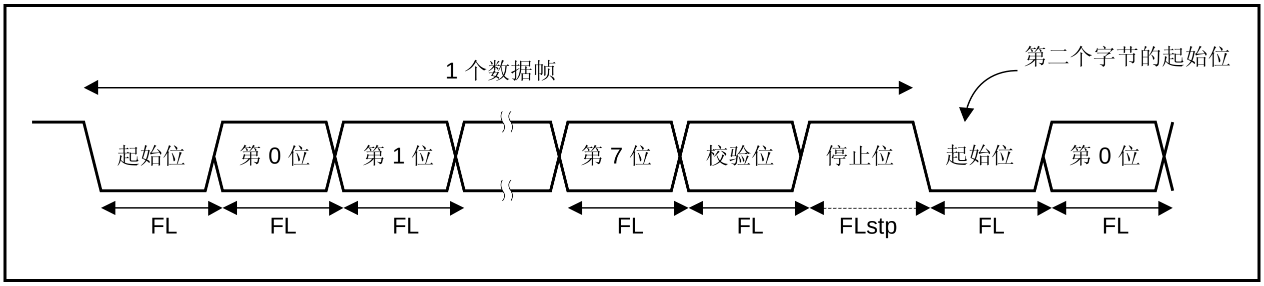
备注　**1.** **接收精度取决于一帧内的位数量、基准时钟频率和分频比（k）。基准时钟频率越高且分频比（k）越大，则精度越高。**

2. **k：UBCTL2 的设定值**

（5）连续发送期间的波特率

连续发送期间，从停止位到下一个起始位的传输间隔要比正常情况扩展 2 个基准时钟。但是，由于该时序通过起始位的检测来启动初始化，因此，对传输结果没有影响。

图 15-11. 连续发送期间的传送速率



假定 1 位数据长度为 FL；停止位长度为 FLstp；基准时钟频率为 f_{xx}，可以得到如下等式：

$$FLstp = FL + 2/f_{xx}$$

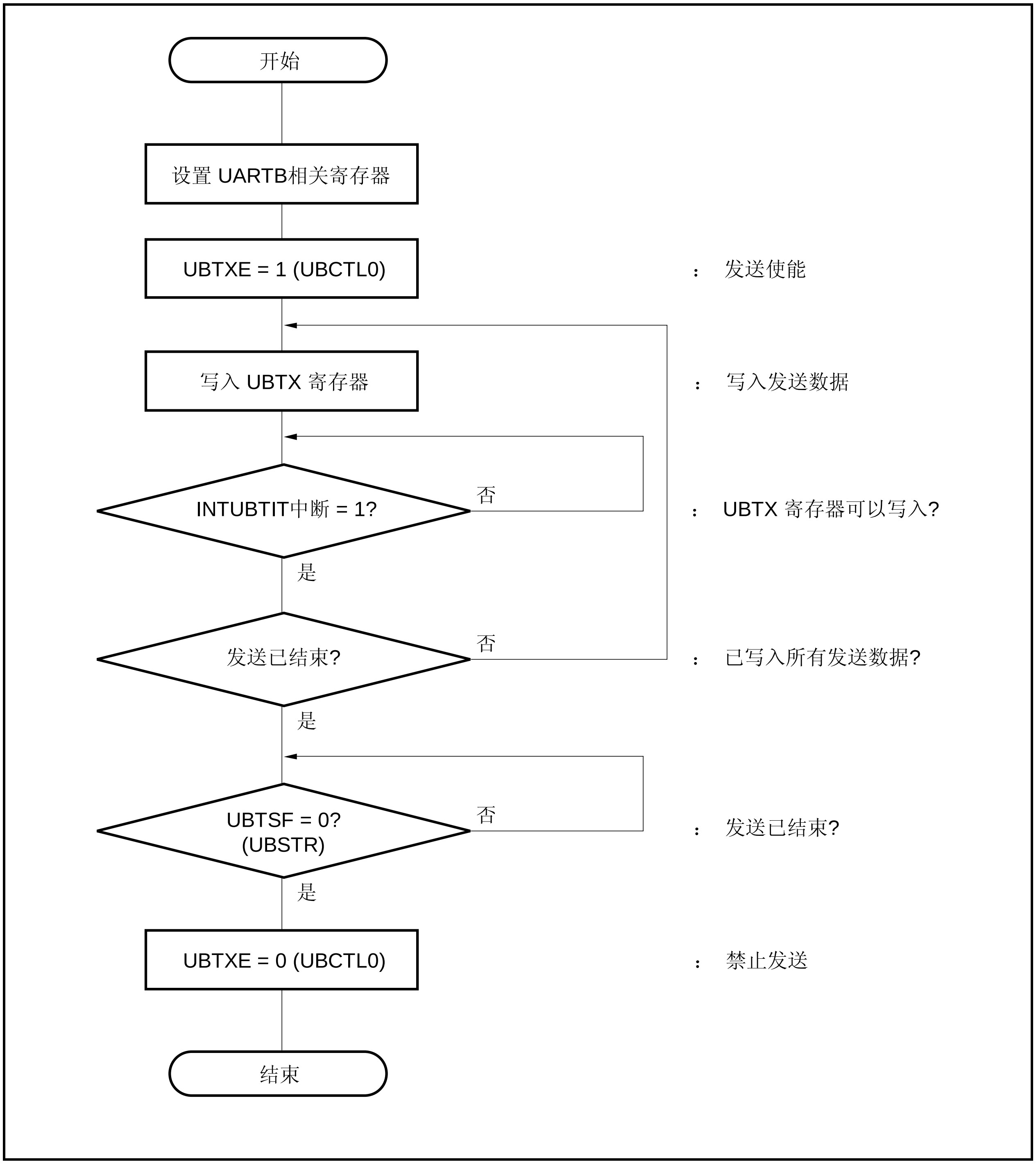
因此，连续发送期间的传输速率如下。

$$传输速率 = 11 \times FL + (2/f_{xx})$$

15.9 控制流程

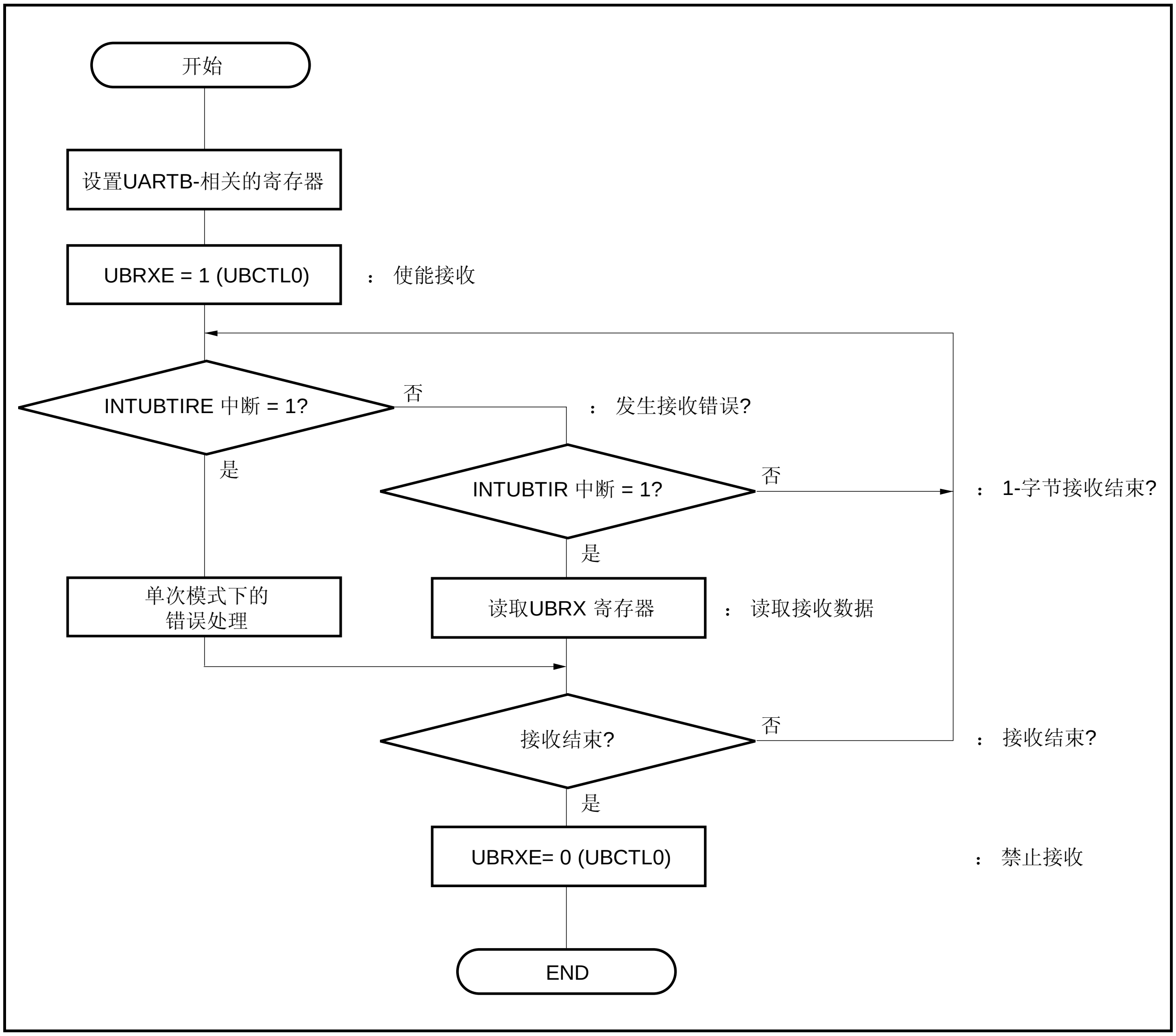
(1) 单次模式下连续发送处理流程的示例 **(CPU** 控制**)**

图 **15-12.** 单次模式下连续发送处理流程的示例 **(CPU** 控制**)**



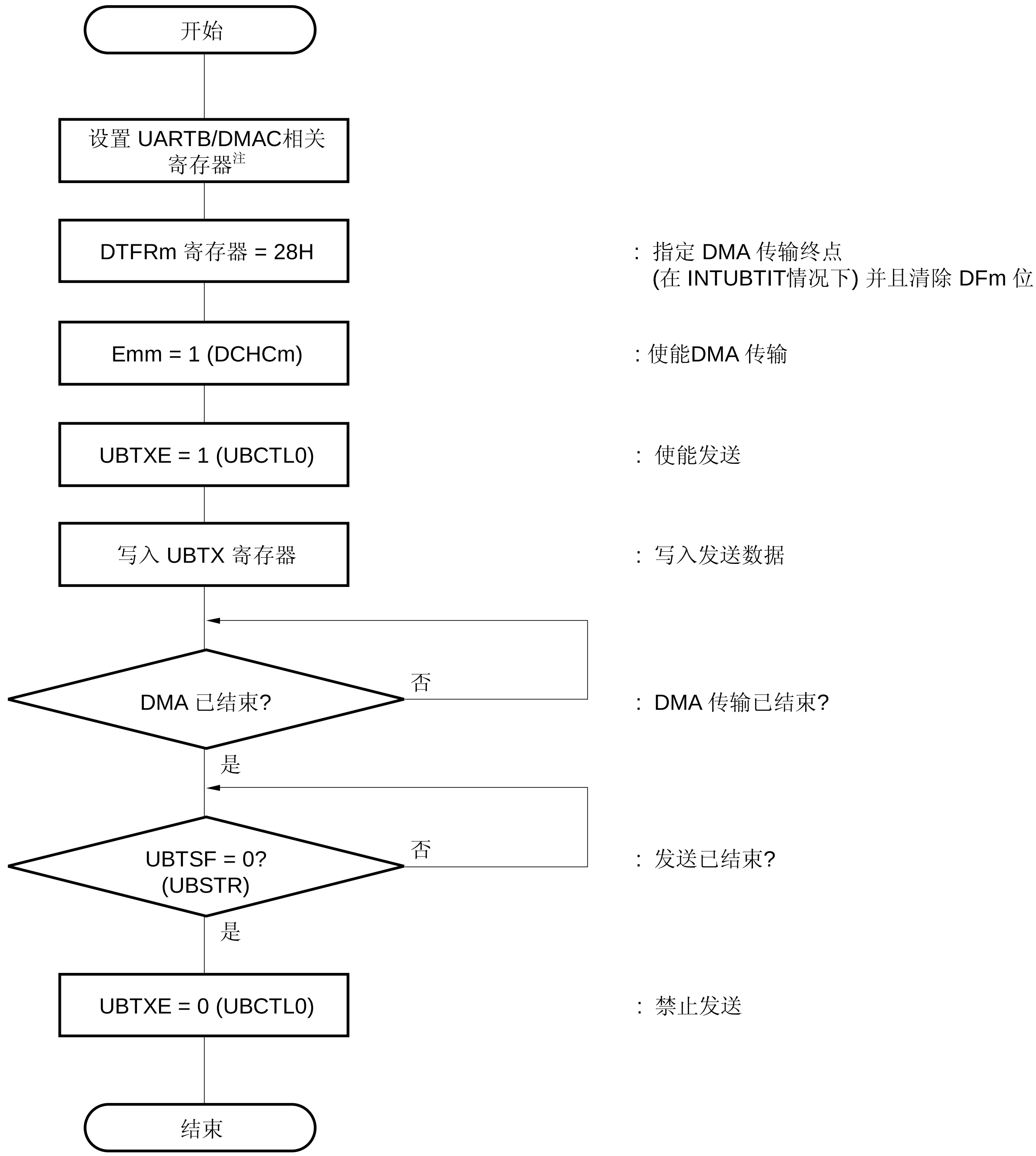
(2)　单次模式下连续接收处理流程的示例 (CPU 控制)

图 **15-13.** 单次模式下连续接收处理流程的示例 **(CPU** 控制**)**



(3) 单次模式下连续发送处理流程的示例 **(DMA** 控制**)**

图 **15-14.** 单次模式下连续发送处理流程的示例 **(DMA** 控制**)**

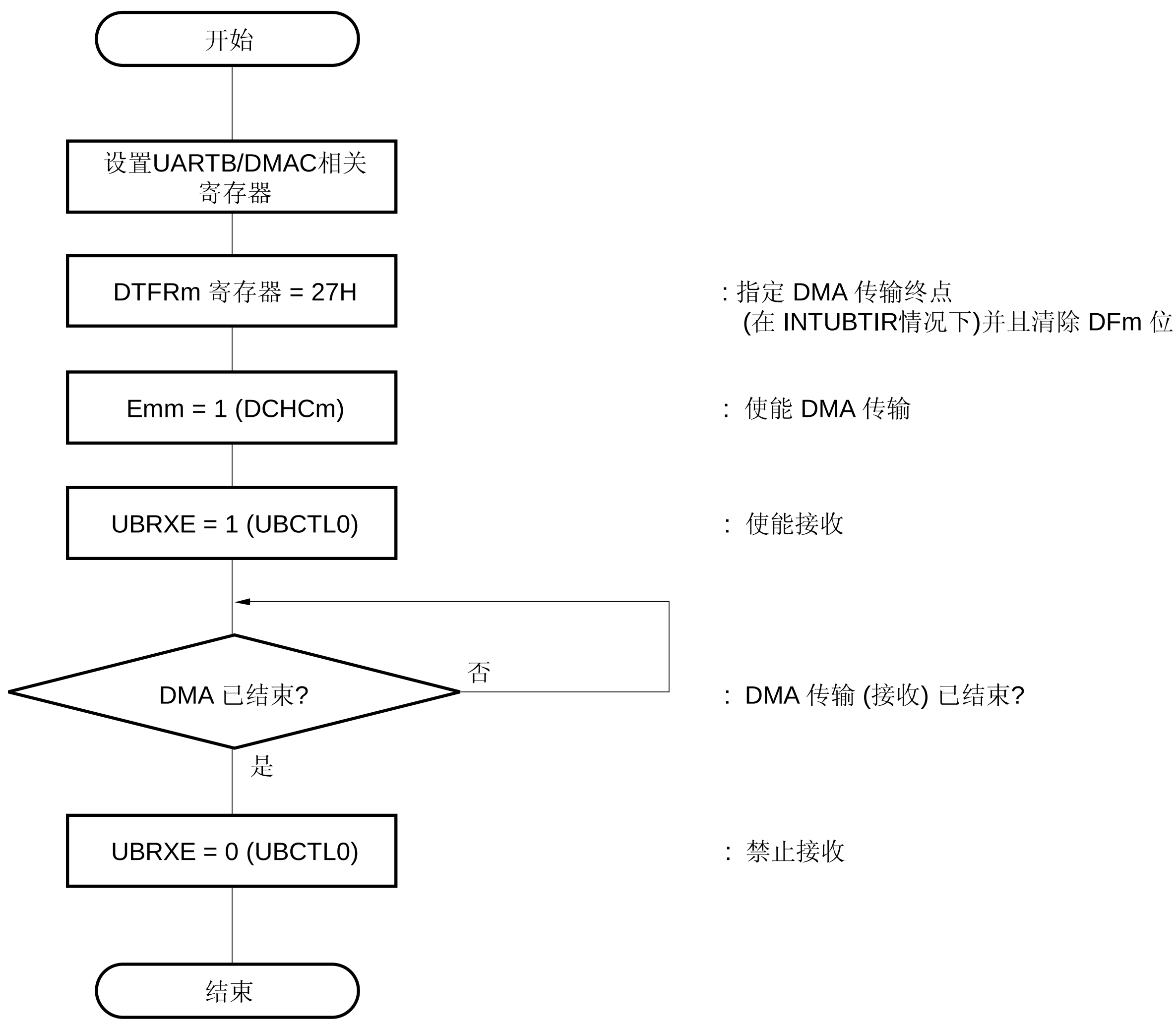


注　　在这个控制流程示例中，第一个字节数据的发送是通过 CPU 写入操作执行的。设置 DMA 传输的数据数量(DBCm 寄存器)和源地址(DSAmH, DSAmL 寄存器)时，必须小心。

备注　　m = 0 至 3

(4) 单次模式下连续接收处理流程的示例 **(DMA** 控制**)**

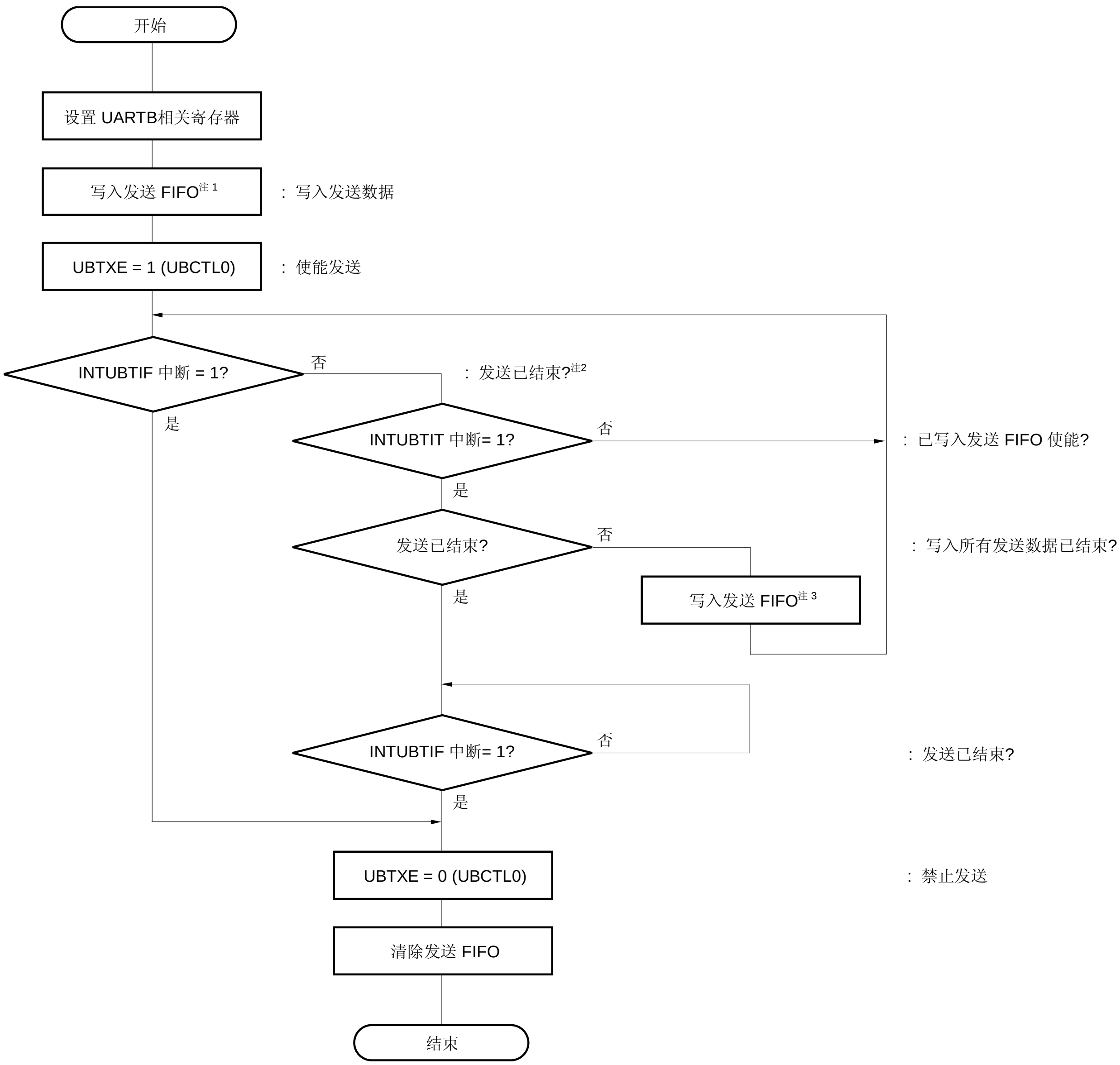
图 **15-15.** 单次模式下连续接收处理流程的示例 **(DMA** 控制**)**



备注　m = 0 至 3

(5) FIFO 模式下连续发送处理流程的示例 (CPU 控制)

图 15-16. FIFO 模式下连续发送处理流程的示例 (CPU 控制)



注 **1.** 写入到发送 FIFO 的发送数据的数量大于 UBFIC2.UBTT3 至 UBFIC2.UBTT0 位设置为触发器的数量。

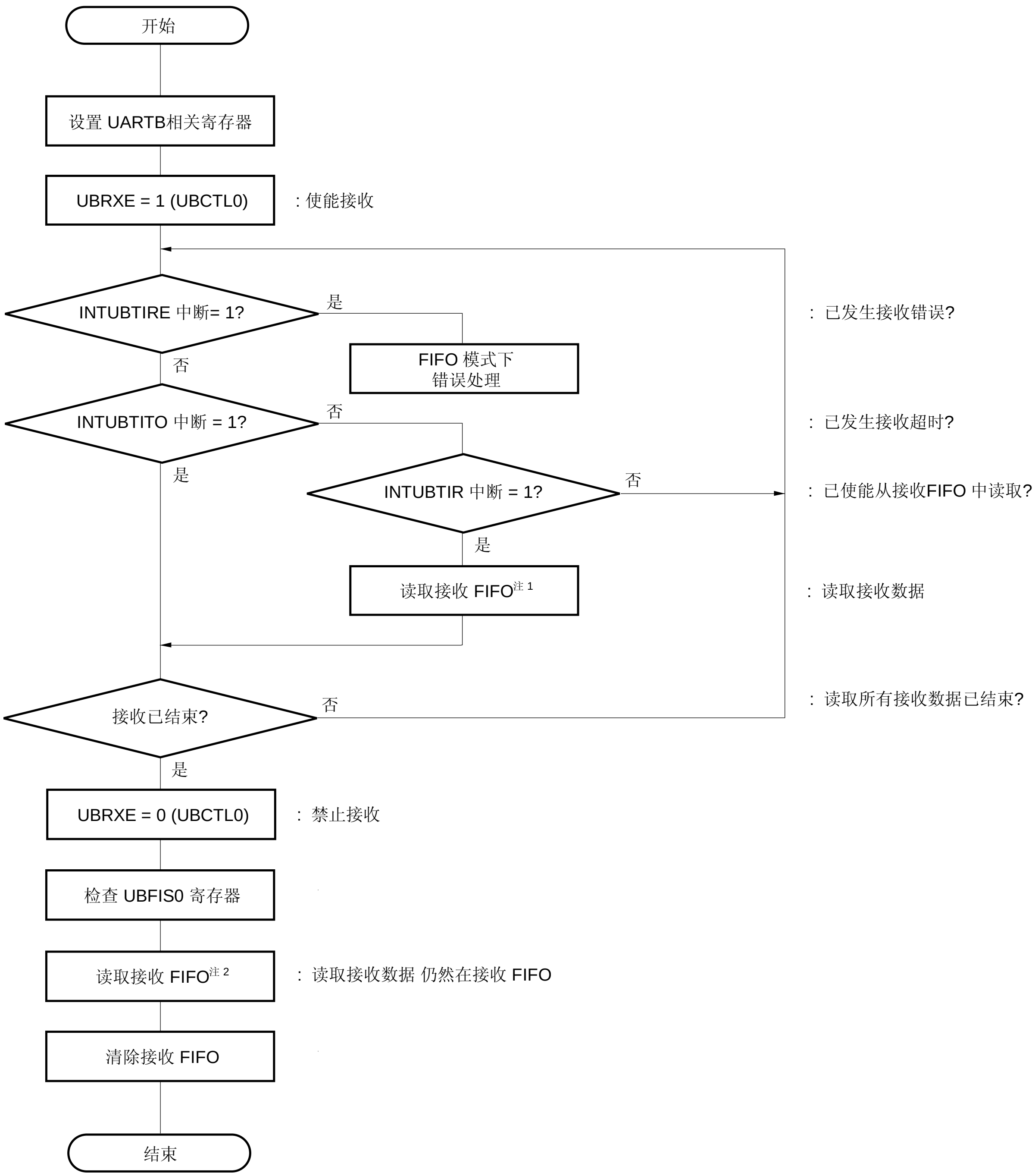
2. 这种情况是在下一个发送数据写入之前，发送结束(发送 FIFO 和发送移位寄存器为空)。要继续数据的发送，清除 INTUBTIF 和 INTUBTIT 信号，并将下一个数据写入到发送 FIFO。

3. 在挂起模式下(UBFIC0.UBITM 位 = 0)，向发送 FIFO 写入的发送数据数量等于 UBFIC2.UBTT3 至 UBFIC2.UBTT0 位设置为触发器的数量。在指针模式下(UBITM 位 = 1)，参考 UBFIS1.UBTB4 至 UBFIS1.UBTB0 位，向发送 FIFO 写入的发送数据数量等于发送 FIFO 内的空余字节数量。

写入 16 字节数据，可以充分使用 8 位 × 16 级 FIFO 功能。

(6) FIFO 模式下连续接收处理的示例 (CPU 控制)

图 15-17. FIFO 模式下连续接收处理的示例 (CPU 控制)

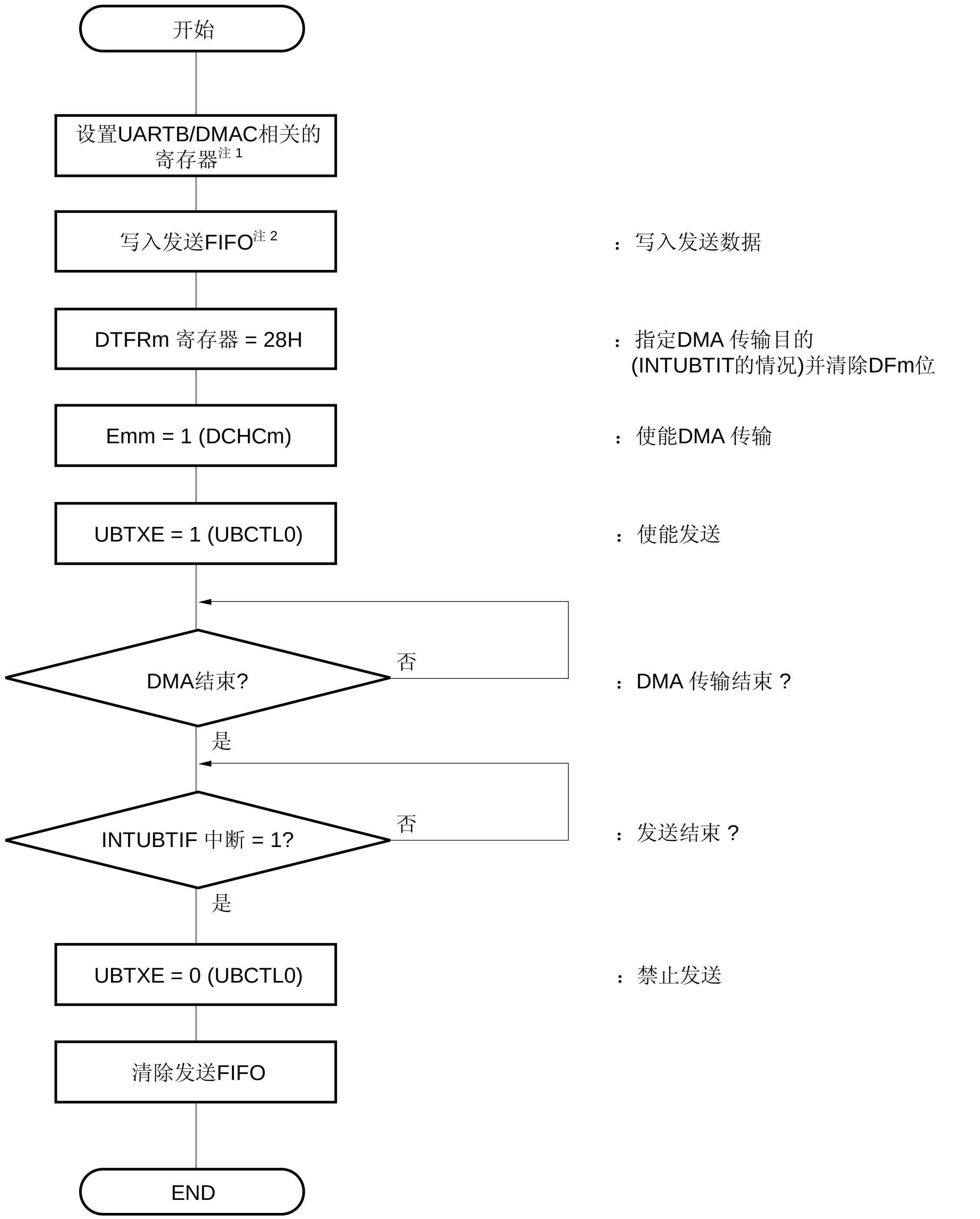


注 **1.** 挂起模式下(UBFIC0.UBIRM 位 = 0)，从接收 FIFO 读取的接收数据的数量等于 UBFIC2.UBRT3 至 UBFIC2.UBRT0 位设置为触发器的数量。在指针模式下(UBIRM 位 = 1)，参考 UBFIS0.UBRB4 至 UBFIS0.UBRB0 位，从接收 FIFO 读取的接收数据的数量等于存储在接收 FIFO 内的字节数量。

2. 通过参考 UBFIS0.UBRB4 至 UBFIS0.UBRB0 位，从接收 FIFO 读取的接收数据的数量等于存储在接收 FIFO 内的字节数量（剩余的接收数据小于设置为触发器的数量）。

(7) FIFO 模式下连续发送(挂起模式)处理的示例 (DMA 控制)

图 15-18. FIFO 模式下连续发送(挂起模式)处理的示例 (DMA 控制)



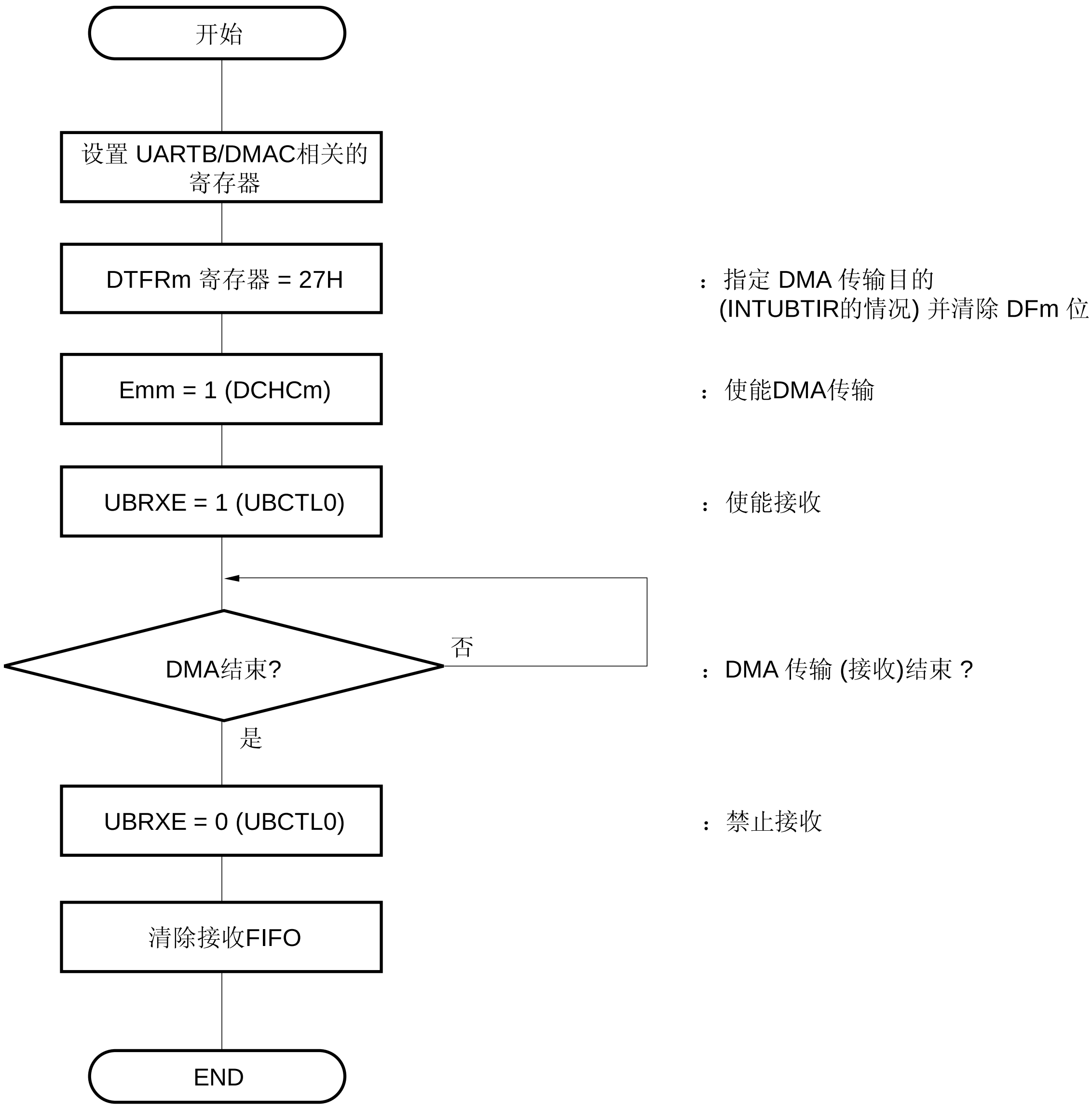
注　**1.** 在这个控制流程示例中，注 **2** 中描述数据发送是通过 CPU 写入操作执行的。设置 DMA 传输的数据数量(DBCm 寄存器)和源地址(DSAmH, DSAmL 寄存器)时，必须小心。

　　2. 向发送 FIFO 写入的发送数据数量等于 UBFIC2.UBTT3 至 UBFIC2.UBTT0 位设置为触发器的数量(= 1 字节)。

备注　m = 0 至 3

(8) **FIFO 模式下连续接收(挂起模式) 处理流程的示例 (DMA 控制)**

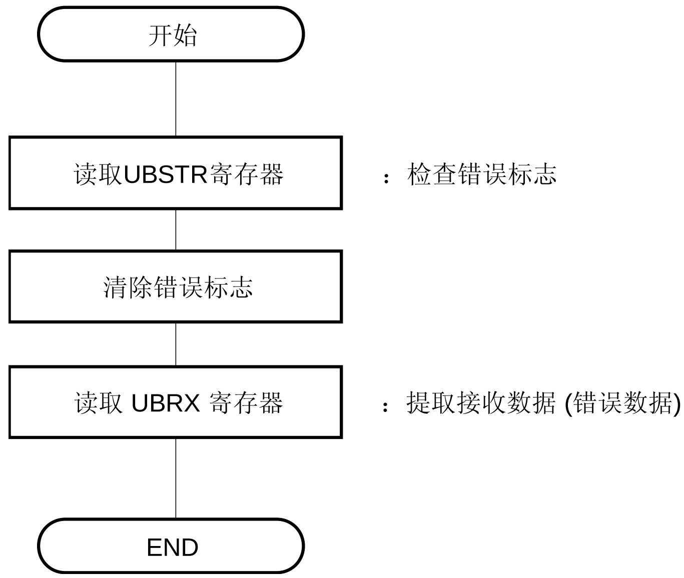
图 **15-19. FIFO** 模式下连续接收**(**挂起模式**)** 处理流程的示例 **(DMA** 控制**)**



备注 m = 0 至 3

(9) 单次模式下接收错误处理的示例

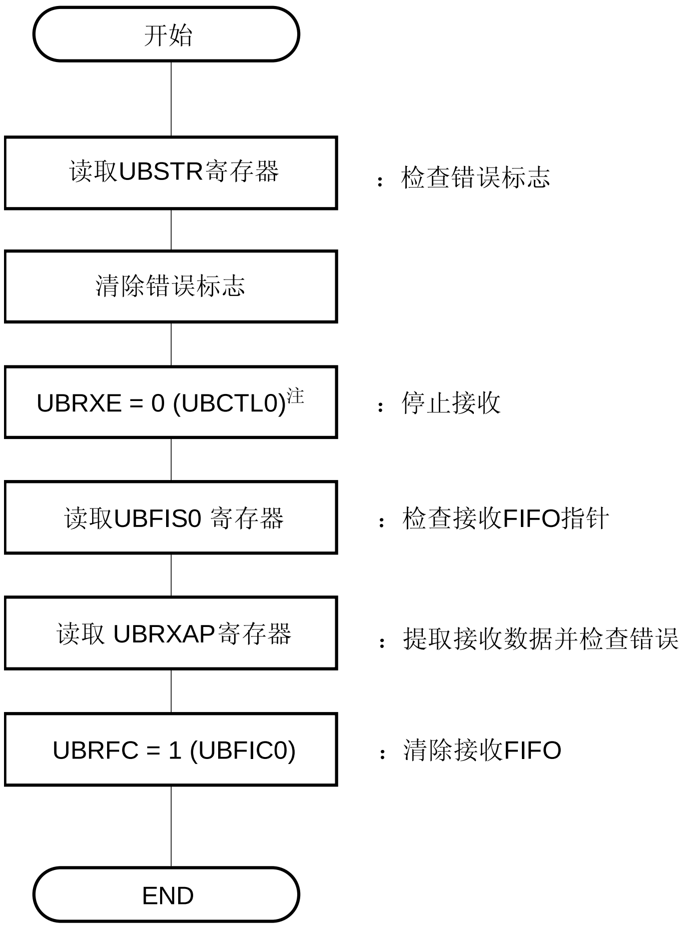
图 **15-20.** 单次模式下接收错误处理的示例



注意事项　在下一个数据的接收结束之前完成这个控制流程，接收可以继续。如果在这个控制流程完成之前接收到下一个数据，即使数据正确接收，也可能会发生接收错误中断请求信号**(INTUBTIRE)**。

(10) FIFO 模式下接收错误处理流程的示例 (1)

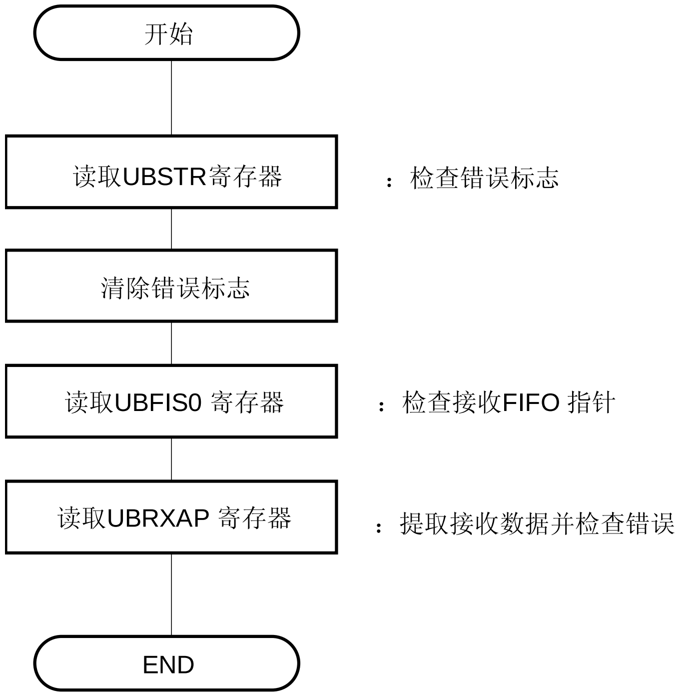
图 **15-21. FIFO** 模式下接收错误处理流程的示例 **(1)**



注　　如果当 UBRXE 位 = 0 时，清除错误标志，UBCTL0 寄存器不必设置。

(11) FIFO 模式下接收错误处理流程的示例(2)

图 15-22. FIFO 模式下接收错误处理流程的示例 (2)



注　在下一个数据的接收结束之前完成这个控制流程，接收可以继续。在接收 FIFO 为空之前，提取接收数据并检查是否发生接收错误。请注意只有当发生奇偶错误或帧错误时，该控制流程有效。如果发生溢出错误，必须清除接收 FIFO(UBFIC0.UBRFC 位 = 1)。

如果在这个控制流程完成之前接收到下一个数据，即使数据正确接收，也可能会发生接收错误中断请求信号(INTUBTIRE)。

15.10 注意事项

关于 UARTB 的注意事项展示如下。

(1) 当 UARTB 的时钟供给停止时

当提供给 UARTB 的时钟停止时(比如，IDLE 和 STOP 模式)，操作停止，各个寄存器保持时钟供给停止时的瞬时值。TXDB 引脚输出也保持，输出的值是时钟供给停止时的瞬时值。但是，时钟供给重启之后，操作无法保证。因此，时钟供给重启之后，应该通过设置 UBPWR 位= 0, UBRXE 位= 0 和 UBTXE 位= 0，对电路进行初始化。

(2) 设置 UBCTL0 寄存器的注意事项

- 使用 UARTB 时，设置 UARTB 功能相关的外部引脚为复用功能模式，并设置 UBCTL2 寄存器。然后在设置其它位之前，将 UBCTL0.UBPWR 位设置为 1。
- 设置 UARTB 功能相关的外部引脚为复用功能模式时，请确保向 RXDB 引脚输入高电平。如果输入了低电平，则判断为在 UBCTL0.UBRXE 位设置为 1 之后输入了下降沿，可能会启动接收。

(3) 设置 UBFIC2 寄存器的注意事项

在将数据写入 UBFIC2 寄存器之前，请确保将 UBCTL0.UBTXE 位 (禁止发送) 和 UBCTL0.UBRXE 位 (禁止接收)设置为 0。如果当 UBTXE 或 UBRXE 位设置为 1 时，将数据写入 UBFIC2 寄存器，操作无法保证。

(4) 发送中断请求信号

在单次模式下，当 UBTX 寄存器为空时(当 1 个字节的数据从 UBTX 寄存器传输到发送移位寄存器)，发生发送使能中断请求信号 (INTUBTIT)。在 FIFO 模式下，当发送 FIFO 和发送移位寄存器没有数据 (当 FIFO 和寄存器为空)时，发生 FIFO 发送结束中断请求信号 (INTUBTIF)。但是，如果发送数据寄存器是因为 $\overline{\text{RESET}}$ 输入清空，不产生 INTUBTIT 信号或 INTUBTIF 信号。

(5) 单次模式下连续发送期间的初始化

在发送处理期间，执行初始化之前，确认 UBSTR.UBTSF 位为 0。如果在 UBTSF 位为 1 时执行初始化，发送数据无法保证。

(6) FIFO 模式下连续发送（挂起模式）期间的初始化

在发送处理期间，执行初始化之前，确认 UBSTR.UBTSF 位为 0 (也可以通过 FIFO 发送结束中断请求信号(INTUBTIF)来完成)。如果在 UBTSF 位为 1 时执行初始化，发送数据无法保证。

通过 DMA 控制将发送数据写入发送 FIFO，由 UBFIC2.UBTT3 至 UBFIC2.UBTT0 位设置为触发器的发送数据指定数量是 1 个字节；否则操作无法保证。

(7) FIFO 模式下连续发送（指针模式）期间的初始化

在发送处理期间，执行初始化之前，确认 UBSTR.UBTSF 位为 0 (也可以通过 FIFO 发送结束中断请求信号(INTUBTIF)来完成)。如果在 UBTSF 位为 1 时执行初始化，发送数据无法保证。

(8) FIFO 模式下的接收操作 (指定为指针模式)

如果在 FIFO 模式下指定为指针模式，如果通过参考 UBFIS0 寄存器，从接收 FIFO 读取的数据数量是存储在接收 FIFO 内的字节数量。即使发生接收结束中断请求信号 (INTUBTIR)，没有数据可能存储在接收 FIFO 内 (UBFIS0.UBRB4 至 UBFIS0.UBRB0 位 = 00000)。这种情况下，不要从接收 FIFO 读取数据。请确认存储在接收 FIFO 内的字节数量 = 1 字节(UBRB4 至 UBRB0 位 = 除 00000 之外)或更多之后，从接收 FIFO 读取数据。

16.1 CSIB 和其它串行接口的模式转换

16.1.1 CSIB0 和 UARTA0 的模式转换

在 V850E/IF3 和 V850E/IG3 中，CSIB0 和 UARTA0 是复用功能，这些功能不能同时使用。要使用 CSIB0 和 UARTA0，必须预先设置 PMC4、PFC4 和 PFCE4 寄存器。

注意事项　如果在发送或接收期间切换模式，则 CSIB0 和 UARTA0 相关的发送和接收操作得不到保证。确保禁止未使用的功能。

图 16-1. CSIB0 和 UARTA0 模式转换设置

复位后：00H　　R/W　　地址：FFFFF448H

	7	6	5	4	3	2	1	0
PMC4	PMC47	PMC46	PMC45	PMC44	PMC43	PMC42	PMC41	PMC40

复位后：00H　　R/W　　地址：FFFFF468H

	7	6	5	4	3	2	1	0
PFC4	PFC47	PFC46	PFC45	PFC44	PFC43	PFC42	PFC41	PFC40

复位后：00H　　R/W　　地址：FFFFF708H

	7	6	5	4	3	2	1	0
PFCE4	PFCE47	PFCE46	PFCE45	PFCE44	PFCE43	PFCE42	PFCE41	PFCE40

PMC42	PFCE42	PFC42	P42引脚复用功能的规范
0	×	×	I/O 端口
1	0	0	$\overline{\text{SCKB0}}$ I/O
1	0	1	INTP13 输入
1	1	0	禁止设置
1	1	1	禁止设置

PMC41	PFCE41	PFC41	P41引脚复用功能的规范
0	×	×	I/O 端口
1	0	0	SOB0 输出
1	0	1	TXDA0 输出
1	1	0	禁止设置
1	1	1	禁止设置

PMC40	PFC40	P40 引脚复用功能的规范
0	×	I/O 端口
1	0	SIB0 输入
1	0	RXDA0 输入

备注　　× = 无需理会

16.1.2 CSIB1 和 UARTA2 的模式转换

在 V850E/IF3 和 V850E/IG3 中，CSIB1 和 UARTA2 是复用功能，这些功能不能同时使用。要使用 CSIB1 和 UARTA2，必须预先设置 PMC3、PFC3 和 PFCE3 寄存器。

注意事项 如果在发送或接收期间切换模式，则 **CSIB1** 和 **UARTA2** 相关的发送和接收操作得不到保证。确保禁止未使用的功能。

图 16-2. CSIB1 和 UARTA2 模式转换设置

复位后：00H R/W 地址： FFFFF446H

	7	6	5	4	3	2	1	0
PMC3	PMC37	PMC36	PMC35	PMC34	PMC33	PMC32	PMC31	PMC30

复位后：00H R/W 地址： FFFFF466H

	7	6	5	4	3	2	1	0
PFC3	PFC37	PFC36	PFC35	PFC34	PFC33	PFC32	PFC31	PFC30

复位后：00H R/W 地址： FFFFF706H

	7	6	5	4	3	2	1	0
PFCE3	PFCE37	PFCE36	PFCE35	PFCE34	0	PFCE32	PFCE31	PFCE30

PMC34	PFCE34	PFC34	P34 引脚复用功能的规范
0	×	×	I/O 端口
1	0	0	$\overline{\text{SCKB1}}$ I/O
1	0	1	INTP11输入
1	1	0	$\overline{\text{CS0}}$注 输出
1	1	1	禁止设置

PMC33	PFC33	P33引脚复用功能的规范
0	×	I/O 端口
1	0	SOB1 输出
1	0	TXDA2 输出

PMC32	PFCE32	PFC32	P32 引脚复用功能的规范
0	×	×	I/O 端口
1	0	0	SIB1输入
1	0	1	RXDA2输入
1	1	0	$\overline{\text{CS1}}$注 输出
1	1	1	禁止设置

<R> 注 仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

备注 × = 无需理会

16.1.3 CSIB2 和 UARTB 的模式转换

在 V850E/IF3 和 V850E/IG3 中，CSIB2 和 UARTB 是复用功能，这些功能不能同时使用。要使用 CSIB2 和 UARTB，必须预先设置 PMC3、PFC3 和 PFCE3 寄存器。

注意事项 如果在发送或接收期间切换模式，则 CSIB2 和 UARTB 相关的发送和接收操作得不到保证。确保禁止未使用的功能。

图 16-3. CSIB2 和 UARTB 模式转换设置

复位后：00H　R/W　地址：FFFFF446H

	7	6	5	4	3	2	1	0
PMC3	PMC37	PMC36	PMC35	PMC34	PMC33	PMC32	PMC31	PMC30

复位后：00H　R/W　地址：FFFFF466H

	7	6	5	4	3	2	1	0
PFC3	PFC37	PFC36	PFC35	PFC34	PFC33	PFC32	PFC31	PFC30

复位后：00H　R/W　地址：FFFFF706H

	7	6	5	4	3	2	1	0
PFCE3	PFCE37	PFCE36	PFCE35	PFCE34	0	PFCE32	PFCE31	PFCE30

PMC37	PFCE37	PFC37	P37 引脚复用功能的规范
0	×	×	I/O 端口
1	0	0	$\overline{\text{SCKB2}}$ I/O
1	0	1	INTP12 输入
1	1	0	ASTB[注] 输出
1	1	1	禁止设置

PMC36	PFCE36	PFC36	P36引脚复用功能的规范
0	×	×	I/O 端口
1	0	0	SOB2 输出
1	0	1	TXDB 输出
1	1	0	禁止设置
1	1	1	禁止设置

PMC35	PFCE35	PFC35	P35引脚复用功能的规范
0	×	×	I/O 端口
1	0	0	SIB2 输入
1	0	1	RXDB 输入
1	1	0	禁止设置
1	1	1	禁止设置

<R> 注　仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

备注　× = 无需理会

16.2 特性

- 传输速率：最大 8 Mbps（使用内部时钟）
- 主机、从机模式可选
- 8-位至 16-位传输，3-线串行接口
- 中断请求信号（INTCBnRE，INTCBnT，INTCBnR）
- 串行时钟和数据相位可翻转
- 传输数据长度可以在 8 位至 16 位之间选择任意位长度
- 可切换 MSB/LSB 先行传输数据
- 3-线传输 SOBn： 串行数据输出
  SIBn： 串行数据输入
  $\overline{\text{SCKBn}}$： 串行时钟 I/O

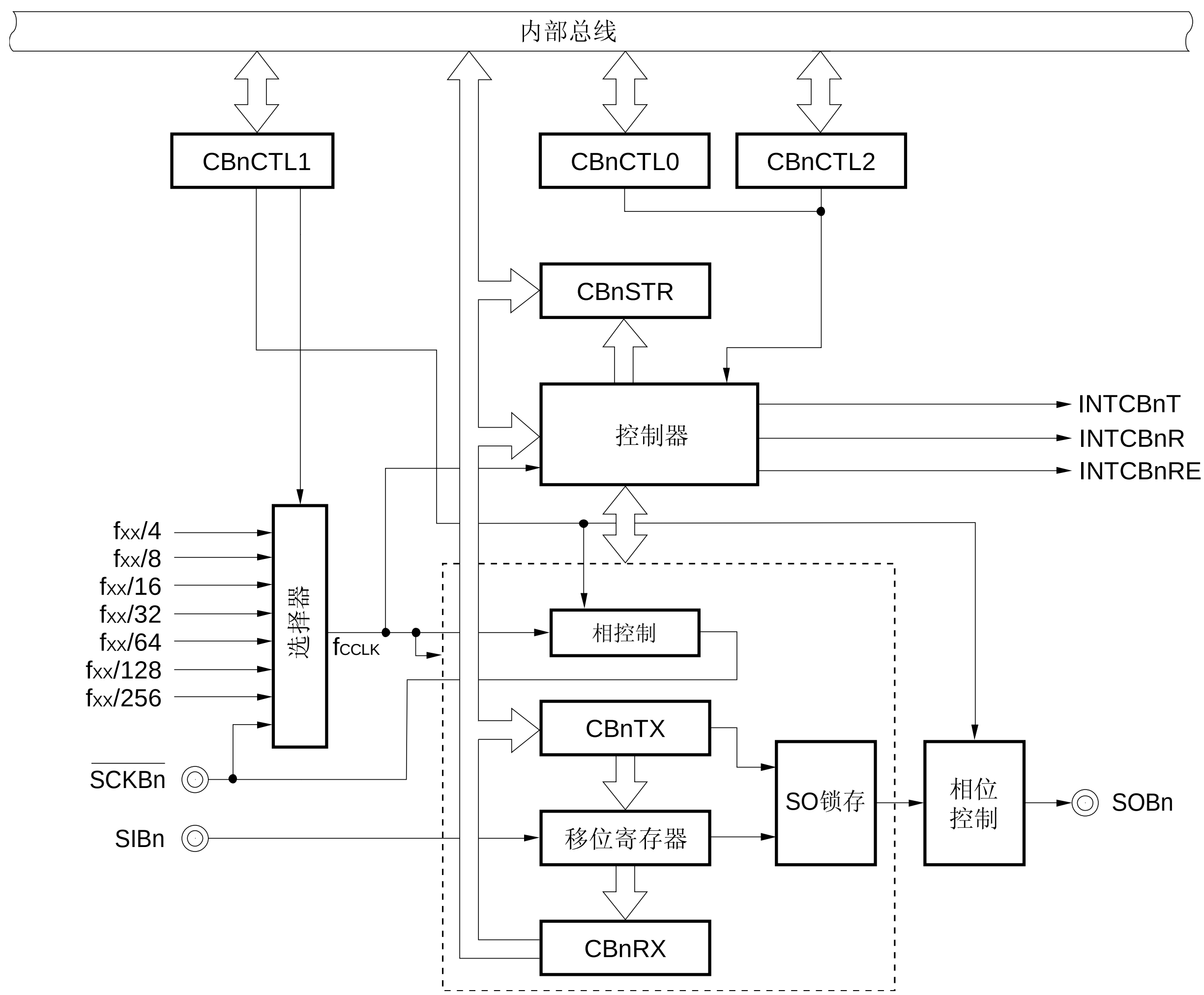
  可指定为发送模式、接收模式以及发送/接收模式。

备注 n = 0 至 2

16.3 配置

以下显示 CSIBn 的功能框图。

图 16-4. CSIBn 框图



备注 1. n = 0 至 2

2. fCCLK：通信时钟(8 MHz (最大值))

CSIBn 包括下列硬件：

表 16-1. CSIBn 的配置

项目	配置
寄存器	CSIBn 接收数据寄存器 （CBnRX） CSIBn 发送数据寄存器 （CBnTX）
控制寄存器	CSIBn 控制寄存器 0 （CBnCTL0） CSIBn 控制寄存器 1 （CBnCTL1） CSIBn 控制寄存器 2 （CBnCTL2） CSIBn 状态寄存器 （CBnSTR）

（1） CSIBn 接收数据寄存器（CBnRX）

CBnRX 是 16-位缓存寄存器，用于存放接收数据。

该寄存器是只读寄存器，以 16 位宽度读取。

在接收允许状态下，通过读取 CBnRX 寄存器启动接收操作。

如果传输数据长度为 8 位，该寄存器的低 8 位作为寄存器 CBnRXL，且只能以 8 位宽度进行读取。

系统复位后，该寄存器被设置为 0000H。

除复位之外，清除 CBnCTL0.CBnPWR 位（为 0），也可以对 CBnRX 寄存器进行初始化。

复位后：0000H　R　地址：　CB0RX FFFFFD04H, CB0RXL FFFFFD04H,
CB1RX FFFFFD14H, CB1RXL FFFFFD14H,
CB2RX FFFFFD24H, CB2RXL FFFFFD24H

	15 … 8	7 … 0
CBnRX (n = 0 至2)		

（2） CSIBn 发送数据寄存器（CBnTX）

CBnTX 寄存器是 16 位缓冲寄存器，用于写入 CSIBn 的发送数据。

该寄存器可以按照 16 位宽度进行读取或写入。

在发送允许状态下，通过向 CBnTX 寄存器写入数据启动发送操作。

如果传输数据长度为 8 位，该寄存器的低 8 位作为寄存器 CBnTXL 按字节进行读取或写入。

系统复位后，该寄存器被设置为 0000H。

复位后：0000H　R/W　地址：　CB0TX FFFFFD06H, CB0TXL FFFFFD06H,
CB1TX FFFFFD16H, CB1TXL FFFFFD16H,
CB2TX FFFFFD26H, CB2TXL FFFFFD26H

	15 … 8	7 … 0
CBnTX (n = 0 至2)		

备注　通信开始条件如下所示。

发送模式（CBnTXE 位 = 1，CBnRXE 位 = 0）：	写入 CBnTX 寄存器
发送/接收模式 （CBnTXE 位 = 1，CBnRXE 位 = 1）：	写入 CBnTX 寄存器
接收模式（CBnTXE 位 = 0，CBnRXE 位 = 1）：	从 CBnRX 寄存器读取数据

16.4 控制寄存器

以下寄存器用来控制 CSIBn。

- CSIBn 控制寄存器 0 （CBnCTL0）
- CSIBn 控制寄存器 1 （CBnCTL1）
- CSIBn 控制寄存器 2 （CBnCTL2）
- CSIBn 状态寄存器 （CBnSTR）

（1） CSIBn 控制寄存器 0 （CBnCTL0）

CBnCTL0 寄存器用于控制 CSIBn 串行传输操作。
该寄存器可以按字节或按位进行读取或写入。
系统复位后，该寄存器被设置为 01H。

（1/2）

复位后： 01H R/W 地址： CB0CTL0 FFFFFD00H, CB1CTL0 FFFFFD10H, CB2CTL0 FFFFFD20H

	<7>	<6>	<5>	<4>	3	2	1	<0>
CBnCTL0 (n = 0 至 2)	CBnPWR	CBnTXE[注]	CBnRXE[注]	CBnDIR[注]	0	0	CBnTMS[注]	CBnSCE

CBnPWR	CSIBn操作禁止/使能的规范
0	禁止CSIBn操作并重置CBnSTR 寄存器
1	使能CSIBn操作
• CBnPWR 位控制CSIBn操作，并复位内部电路。	

CBnTXE[注]	发送操作禁止/使能的规范
0	禁止发送操作
1	使能发送操作
• 当CBnTXE 位为0时，SOBn输出低电平。	

CBnRXE[注]	接收操作禁止/使能的规范
0	禁止接收操作
1	使能接收操作
• 当CBnRXE 位被清除为0时，即使传输了为禁止接收操作而预置的数据，也不输出接收结束中断，并且接收数据 (CBnRX 寄存器)不被更新。	

注 只有当 CBnPWR 位= 0 时，这些位才可重写。但是，在重写这些位的同时，可以设置 CBnPWR 位= 1 。

注意事项 请确保将第 3 位和第 2 位设置为“0”。

（2/2）

CBnDIR注1	传输方向模式的规范(MSB/LSB)
0	首先传输MSB
1	首先传输LSB

CBnTMS注1	传输模式规范
0	单次传输模式
1	连续传输模式

- 通信类型2或4(CBnCTL1.CBnDAP位 = 1)情况下使用发送或发送/接收模式时，在检查BnSTR.CBnTSF位为0之后，将传输数据写入CBnTX寄存器。
- 使用DMA时，选择连续传输模式。

CBnSCE	启动传输禁止/使能的规范
0	通信启动触发无效
1	通信启动触发有效

- 在主模式下
  该位使能或禁止通信启动触发。
  (a) 在单次接收模式下
  在读取接收数据(CBnRX寄存器)注 2之前，设置CBnSCE位为0。
  (b) 在连续接收模式下
  在最后数据的接收结束注 3之前的一个通信时钟，设置CBnSCE位为0。

- 在从模式下
  该位使能或禁止通信启动触发。
  (a) 在单次接收模式或连续接收模式下
  将CBnSCE位设置为1注4。
- 在单次发送或发送/接收模式、或连续发送或发送/接收模式下
  CBnSCE位的功能无效。推荐将该位设置为1。

注
1. 只有当 CBnPWR 位= 0 时，这些位才可重写。但是，重写这些位的同时，CBnPWR 位可以设置为 1 。
2. 如果其值为 1 时读取 CBnSCE 位，则启动下一次通信操作。
3. 最末数据接收结束之前的一个通信时钟，CBnSCE 位不清 0，则下一次通信操作自动开始。
   在读取了最后数据之后，要再次启动通信操作，将 CBnSCE 位设置为 1，并对 CBnRX 寄存器执行一次伪读取。
4. 要启动接收，必需一个伪读取。

(a) 如何使用 **CBnSCE** 位

(i) 在单次接收模式下

<1> 当接收最后数据以 INTCBnR 中断服务结束时，将 CBnSCE 位清除为 0，再读取 CBnRX 寄存器。

<2> 最后数据的接收结束后，当接收禁止时，检查 CBnSTR.CBnTSF 位是 0，再将 CBnPWR 和 CBnRXE 位清除为 0。要继续接收，将 CBnSCE 位设置为 1，并执行对 CBnRX 寄存器的伪读取来启动下一次接收操作。

(ii) 在连续接收模式下

<1> 当接收以 INTCBnR 中断服务的最后数据期间，由接收将 CBnSCE 位清除为 0，再读取 CBnRX 寄存器。

<2> 接收最后接收的 INTCBnR 信号之后，从 CBnRX 寄存器读取最后数据。

<3> 最后数据的接收结束后，当接收禁止时，检查 CBnSTR.CBnTSF 位是 0，再将 CBnPWR 和 CBnRXE 位清除为 0。要继续接收，将 CBnSCE 位设置为 1，并执行对 CBnRX 寄存器的伪读取来启动下一次接收操作。

注意事项 **在连续接收模式下，串行时钟不会停止，直到伪读取启动的接收之后因 CBnSCE 位被清除为 0 而执行的接收结束时。**

（2） CSIBn 控制寄存器 1 （CBnCTL1）

CBnCTL 是 8 位寄存器，用于控制 CSIBn 串行传输操作。

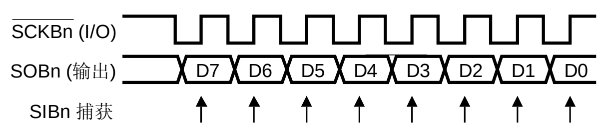
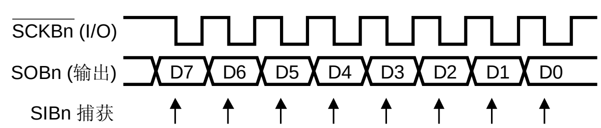
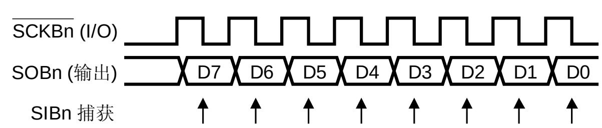
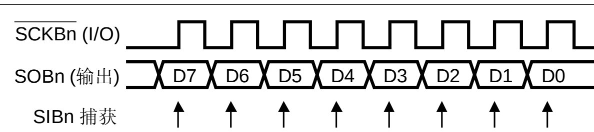
该寄存器可以按字节或按位进行读取或写入。

系统复位后，该寄存器被设置为 00H。

注意事项 仅当 CBnCTL0.CBnPWR 位= 0 时，CBnCTL1 寄存器才可以重写。

复位后：00H　R/W　地址：　CB0CTL1 FFFFFD01H, CB1CTL1 FFFFFD11H, CB2CTL1 FFFFFD21H

	7	6	5	4	3	2	1	0
CBnCTL1 (n = 0 至 2)	0	0	0	CBnCKP	CBnDAP	CBnCKS2	CBnCKS1	CBnCKS0

	CBnCKP	CBnDAP	和$\overline{\text{SCKBn}}$相关的数据发送/接收时序的规范
通信类型 1	0	0	
通信类型 2	0	1	
通信类型 3	1	0	
通信类型 4	1	1	

CBnCKS2	CBnCKS1	CBnCKS0	通信时钟 (f_{CCLK})	模式
0	0	0	$f_{XX}/4$	主模式
0	0	1	$f_{XX}/8$	主模式
0	1	0	$f_{XX}/16$	主模式
0	1	1	$f_{XX}/32$	主模式
1	0	0	$f_{XX}/64$	主模式
1	0	1	$f_{XX}/128$	主模式
1	1	0	$f_{XX}/256$	主模式
1	1	1	外部时钟 ($\overline{\text{SCKBn}}$)	从模式

注　设置 f_{CCLK} 为 8 MHz 或更低。

（3） CSIBn 控制寄存器 2 （CBnCTL2）

CBnCTL2 是 8 位寄存器，用于控制 CSIBn 串行传输的位数量。

该寄存器可进行字节读取或写入。

系统复位后，该寄存器被设置为 00H。

注意事项 **仅当 CBnCTL0.CBnPWR 位= 0 或 CBnTXE 和 CBnRXE 位均为 0 时，CBnCTL2 寄存器才可以重写。**

复位后：00H R/W 地址： CB0CTL2 FFFFFD02H, CB1CTL2 FFFFFD12H,
CB2CTL2 FFFFFD22H

	7	6	5	4	3	2	1	0
CBnCTL2	0	0	0	0	CBnCL3	CBnCL2	CBnCL1	CBnCL0

(n = 0 至 2)

CBnCL3	CBnCL2	CBnCL1	CBnCL0	串行寄存器位长度
0	0	0	0	8 位
0	0	0	1	9 位
0	0	1	0	10 位
0	0	1	1	11 位
0	1	0	0	12 位
0	1	0	1	13 位
0	1	1	0	14 位
0	1	1	1	15 位
1	×	×	×	16 位

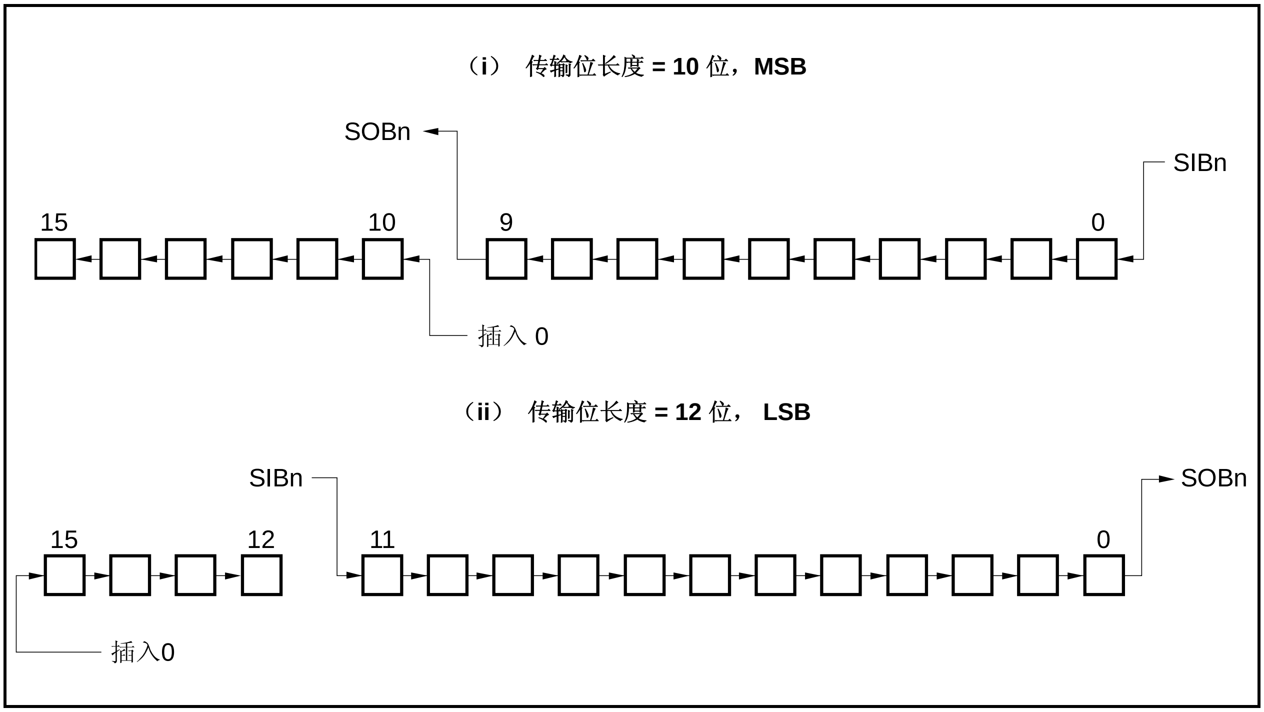
备注 如果传输位数不是 8 位或 16 位，从 CBnTX 和 CBnRX 寄存器的 LSB 起进行填充。

（a）　传输数据长度改变功能

可以使用 CBnCTL2.CBnCL3 至 CBnCTL2.CBnCL0 位，在 8 位和 16 位之间以位为单位设置 CSIBn 传输的数据长度。

当设置的传输比特长度不是 16 位时，从 LSB 开始向 CBnTX 或 CBnRX 寄存器设置数据，而忽略传输起始位是 MSB 还是 LSB。没有使用的高位可以设置为任意数据，但是，在随后的串行传输中接收的对应数据变为 0。

备注　　n = 0 至 2



（4） CSIBn 状态寄存器 （CBnSTR）

CBnSTR 是 8 位寄存器，用于显示 CSIBn 的状态。

该寄存器可以按 8 位或 16 位宽度进行读取或写入，但是，CBnTSF 标志只读。

系统复位后，该寄存器被设为 00H。

除复位输入之外，可以通过清除 CBnCTL0.CBnPWR 位（为 0）对 CBnSTR 寄存器进行初始化。

复位后： 00H R/W 地址： CB0STR FFFFFD03H, CB1STR FFFFFD13H,
CB2STR FFFFFD23H

	<7>	6	5	4	3	2	1	<0>
CBnSTR (n = 0 至 2)	CBnTSF	0	0	0	0	0	0	CBnOVE

CBnTSF	通信状态标志
0	通信停止
1	通信中

• 在发送期间，当在CBnTX中准备好该数据时，设置该寄存器。在接收期间，当执行CBnRX寄存器的伪读取时，设置该寄存器。
当传输结束时，该标志在时钟的最后边沿被清除为0。

CBnOVE	过载错误标志
0	未过载
1	过载

- 在接收操作结束时，启动下一个接收而没有执行CPU对CBnRX寄存器值的读取，发生一个过载错误。
  CBnOVE标志表示这种情况下的过载错误发生状态。
- 通过写入0来清除CBnOVE标志，即使写入1也不能设置该位。

注意事项 在单次传输模式下，**CBnTSF** 位设置为 **1** 时，忽略向 **CBnTX** 寄存器的写入。在传输期间，对操作没有影响。

比如，通过 **INTCBnR** 信号的产生来启动 **DMA** 时，如果下一个数据写入 **CBnTX** 寄存器，则不会传输所写入的数据，因为 **CBnTSF** 位设置为 **1**。

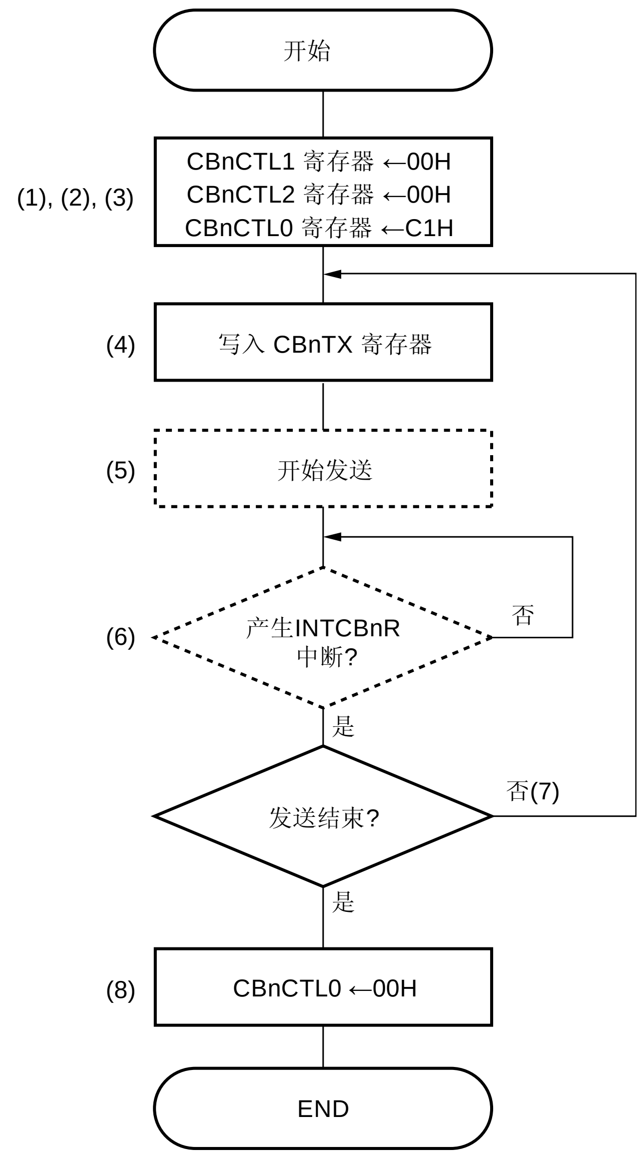
对这样的应用，使用连续传输模式，而不是单次传输模式。

16.5 操作

16.5.1 单次传输模式（主机模式，发送模式）

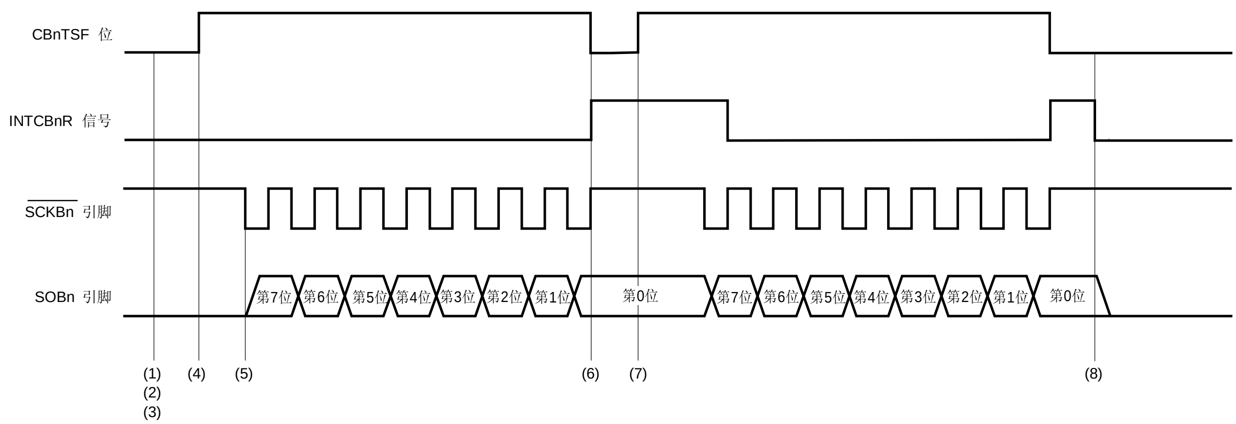
MSB 先行（CBnCTL0.CBnDIR 位= 0），通信类型 1（CBnCTL1.CBnCKP 和 CBnCTL1.CBnDAP 位= 00），通信时钟（f_{CCLK}）= f_{XX}/4（CBnCTL1.CBnCKS2 至 CBnCTL1.CBnCKS0 位 = 000），传输数据长度= 8 位（CBnCTL2.CBnCL3 至 CBnCTL2.CBnCL0 位= 0000）。

（1）　操作流程



备注　**1.** 虚线部分表示硬件处理。
2. 图中的数字对应于（**2**）**操作时序**中的数字。
3. n = 0 至 2

（2）操作时序



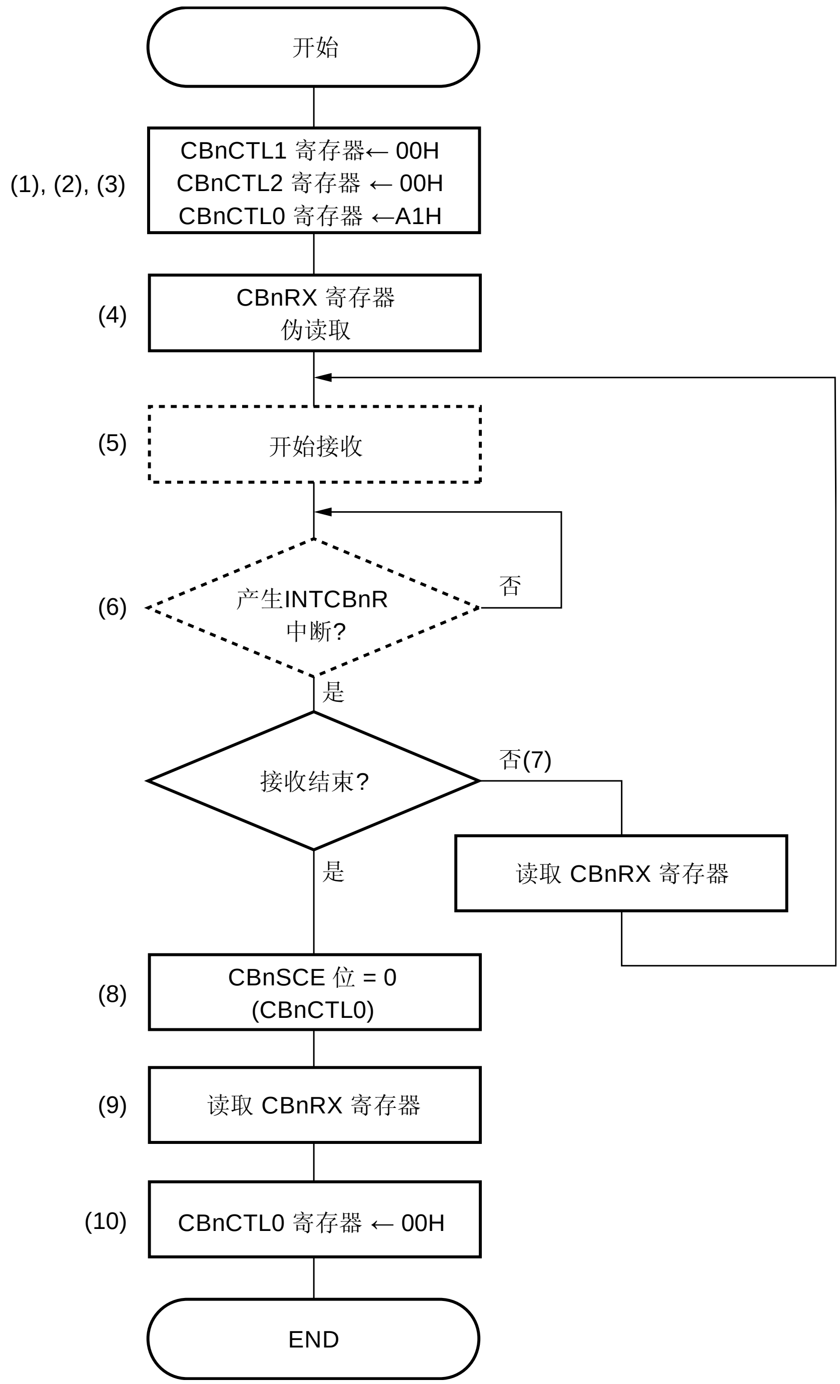
（1）向 CBnCTL1 寄存器写入 00H，选择通信类型 1，通信时钟（f_{CCLK}） = f_{XX}/4，且设置为主机模式。
（2）向 CBnCTL2 寄存器写入 00H，设定传输数据长度为 8 位。
（3）向 CBnCTL0 寄存器写入 C1H，并在使能通信时钟（f_{CCLK}）的同时，选择为发送模式和 MSB 先行方式。
（4）将待发送数据写入 CBnTX 寄存器使得 CBnSTR.CBnTSF 位置为 1，启动发送。
（5）当发送开始后，串行时钟输出至 $\overline{\text{SCKBn}}$ 引脚，待发送数据与串行时钟同步，由 SOBn 引脚输出。
（6）当 CBnCTL2 寄存器设定的传输数据长度发送完毕时，停止串行时钟和发送数据的输出，在串行时钟最末边沿产生接收结束中断请求信号 （INTCBnR），并将 CBnTSF 位清为 0。
（7）若继续进行发送，在 INTCBnR 信号产生之后，通过将发送数据写入 CBnTX 寄存器启动下一次发送。
（8）若要结束发送，设置 CBnCTL0.CBnPWR 位= 0 以及 CBnCTL0.CBnTXE 位= 0。

备注　　n = 0 至 2

16.5.2 单次传输模式（主机模式，接收模式）

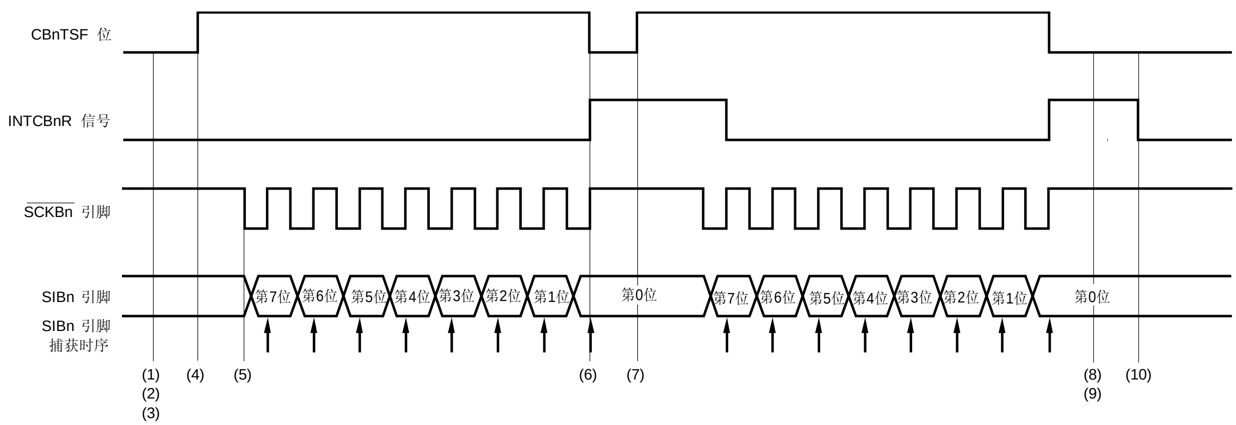
MSB 先行（CBnCTL0.CBnDIR 位= 0），通信类型 1（CBnCTL1.CBnCKP 和 CBnCTL1.CBnDAP 位= 00），通信时钟（f_{CCLK}） = $f_{XX}/4$（CBnCTL1.CBnCKS2 至 CBnCTL1.CBnCKS0 位 = 000），传输数据长度= 8 位（CBnCTL2.CBnCL3 至 CBnCTL2.CBnCL0 位= 0000）。

（1） 操作流程



备注
1. 虚线部分表示硬件处理。
2. 图中的数字对应于（2） **操作时序**中的数字。
3. n = 0 至 2

（2）操作时序



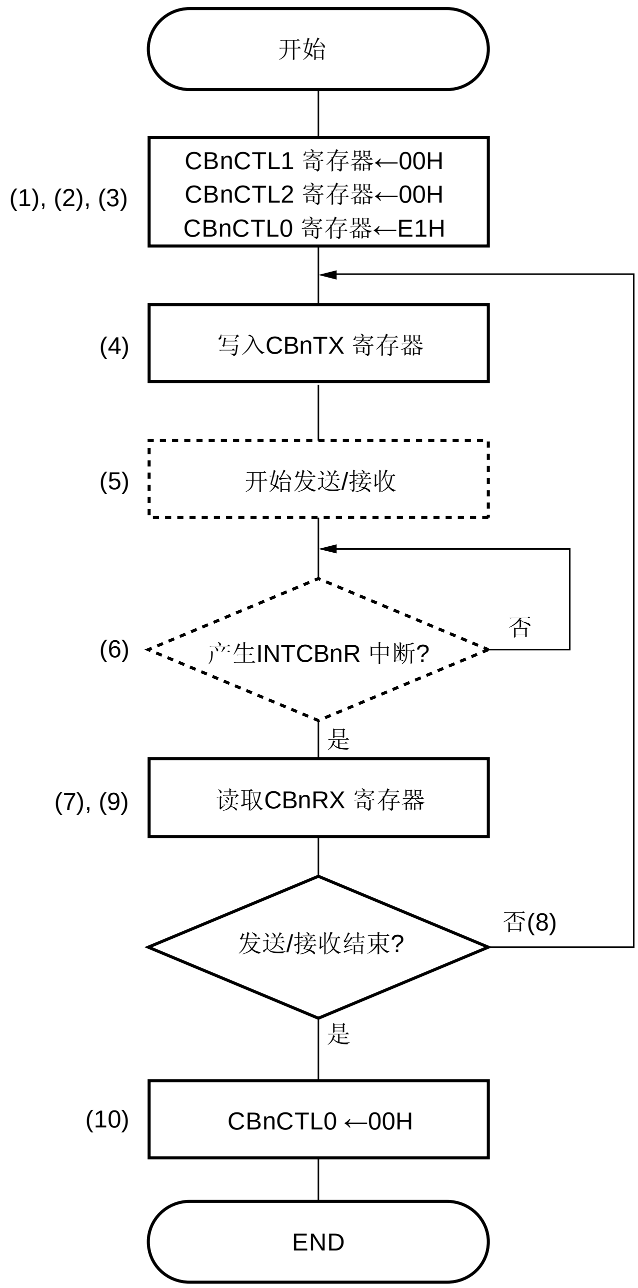
（1） 向 CBnCTL1 寄存器写入 00H，选择通信类型 1，通信时钟（f_{CCLK}） = f_{XX}/4，并且设置为主机模式。
（2） 向 CBnCTL2 寄存器写入 00H，设定传输数据长度为 8 位。
（3） 向 CBnCTL0 寄存器写入 A1H，并在使能通信时钟（f_{CCLK}）的同时，选择为接收模式和 MSB 先行。
（4） 通过对 CBnRX 寄存器执行一次伪读取，使得 CBnSTR.CBnTSF 位为 1，并启动接收。
（5） 当接收开始后，串行时钟输出至 $\overline{\text{SCKBn}}$ 引脚，并与串行时钟同步捕获 SIBn 引脚的接收数据。
（6） 当 CBnCTL2 寄存器设定的传输数据长度接收完毕时，停止串行时钟输出和数据捕获，在串行时钟最末边沿产生接收结束中断请求信号 （INTCBnR），并将 CBnTSF 位清为 0。
（7） 若要连续接收，在 INTCBnR 信号产生之后且 CBnCTL0.CBnSCE 位= 1 时，读取 CBnRX 寄存器。
（8） 若要读取 CBnRX 寄存器，而不启动下一次接收，写入 CBnSCE 位= 0。
（9） 读取 CBnRX 寄存器。
（10）若要结束接收，设置 CBnCTL0.CBnPWR 位= 0 以及 CBnCTL0.CBnRXE 位= 0。

备注　　n = 0 至 2

16.5.3 单次传输模式 （主机模式，发送/接收模式）

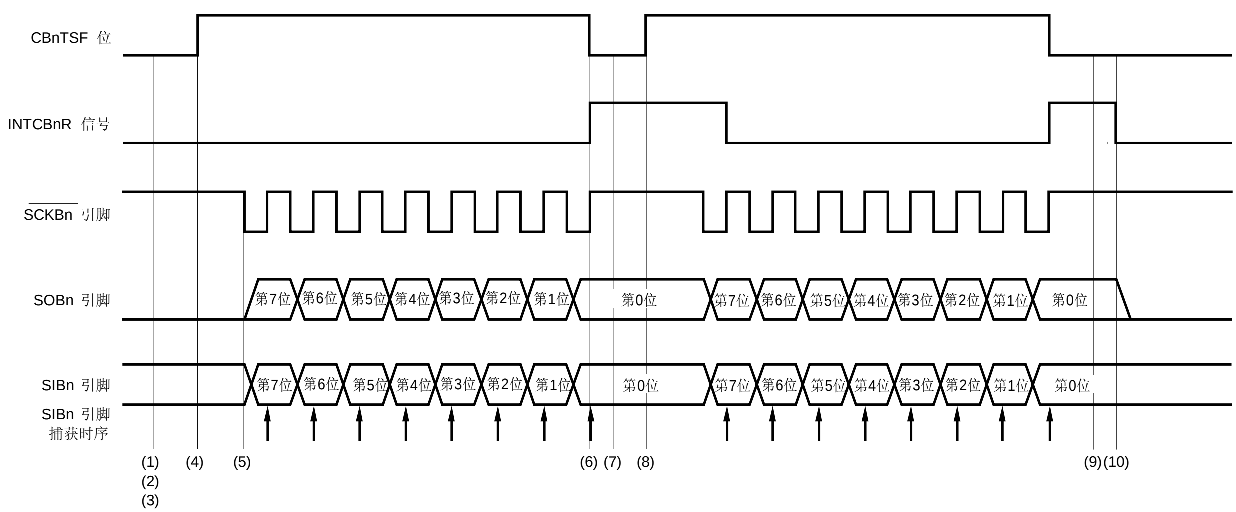
MSB 先行 （CBnCTL0.CBnDIR 位= 0），通信类型 1 （CBnCTL1.CBnCKP 和 CBnCTL1.CBnDAP 位= 00），通信时钟（f_{CCLK}） = f_{XX}/4 （CBnCTL1.CBnCKS2 至 CBnCTL1.CBnCKS0 位 = 000），传输数据长度= 8 位（CBnCTL2.CBnCL3 至 CBnCTL2.CBnCL0 位= 0000）。

（1） 操作流程



备注 **1.** 虚线部分表示硬件处理。
2. 图中的数字对应于（**2**） **操作时序**中的数字。
3. n = 0 至 2

（2） 操作时序



（1） 向 CBnCTL1 寄存器写入 00H，选择通信类型 1，通信时钟（f_{CCLK}） = f_{XX}/4，并且设置为主机模式。

（2） 向 CBnCTL2 寄存器写入 00H，设定传输数据长度为 8 位。

（3） 向 CBnCTL0 寄存器写入 E1H，使在使能通信时钟（f_{CCLK}）的同时，选择为发送/接收模式和 MSB 先行。

（4） 将发送数据写入 CBnTX 寄存器，使得 CBnSTR.CBnTSF 位置为 1，启动发送/接收。

（5） 当发送/接收启动之后，串行时钟输出至 $\overline{\text{SCKBn}}$ 引脚，待发送数据与串行时钟同步，由 SOBn 引脚输出，并且同步捕获 SIBn 引脚的接收数据。

（6） 当 CBnCTL2 寄存器设定的传输数据长度发送/接收完毕时，停止串行时钟输出、发送数据输出以及数据捕获，在串行时钟最末边沿产生接收结束中断请求信号 （INTCBnR），并将 CBnTSF 位清为 0。

（7） 读取 CBnRX 寄存器。

（8） 若继续发送/接收，再次将发送数据写入 CBnTX 寄存器。

（9） 读取 CBnRX 寄存器。

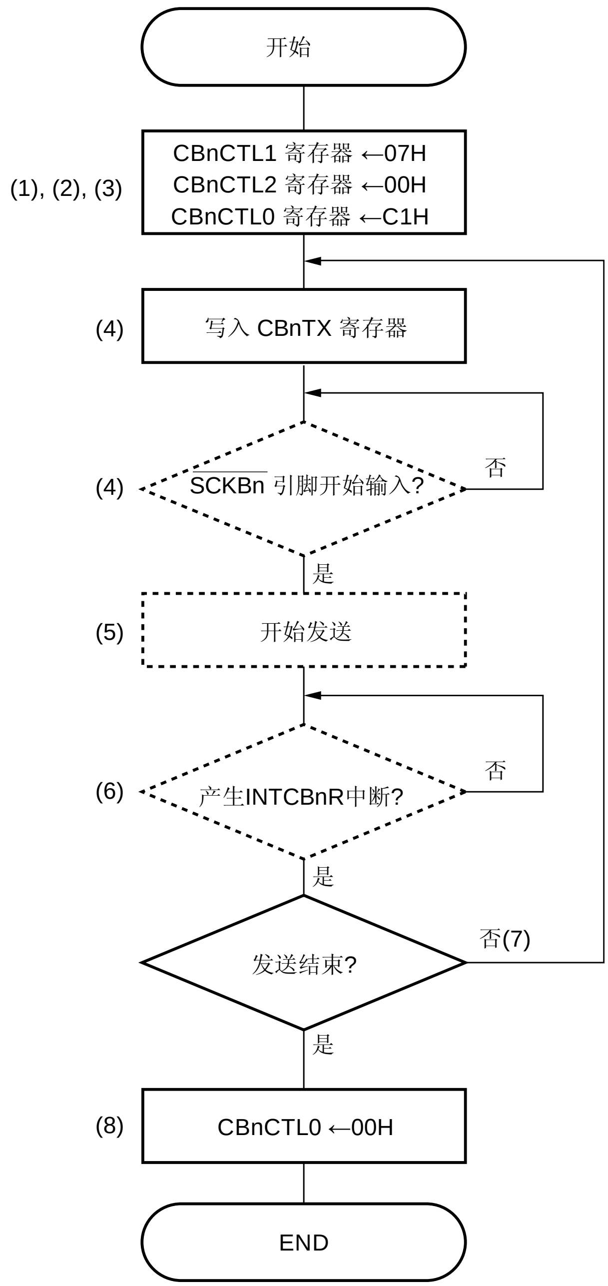
（10）若要结束数据发送/接收，设置 CBnCTL0.CBnPWR 位= 0、CBnCTL0.CBnTXE 位= 0 以及 CBnCTL0.CBnRXE 位= 0。

备注 n = 0 至 2

16.5.4 单次传输模式（从机模式，发送模式）

MSB 先行（CBnCTL0.CBnDIR 位= 0），通信类型 1（CBnCTL1.CBnCKP 和 CBnCTL1.CBnDAP 位= 00），通信时钟（f_{CCLK}） = 外部时钟（$\overline{SCKBn}$）（CBnCTL1.CBnCKS2 至 CBnCTL1.CBnCKS0 位 = 111），传输数据长度= 8 位（CBnCTL2.CBnCL3 至 CBnCTL2.CBnCL0 位= 0000）。

（1） 操作流程

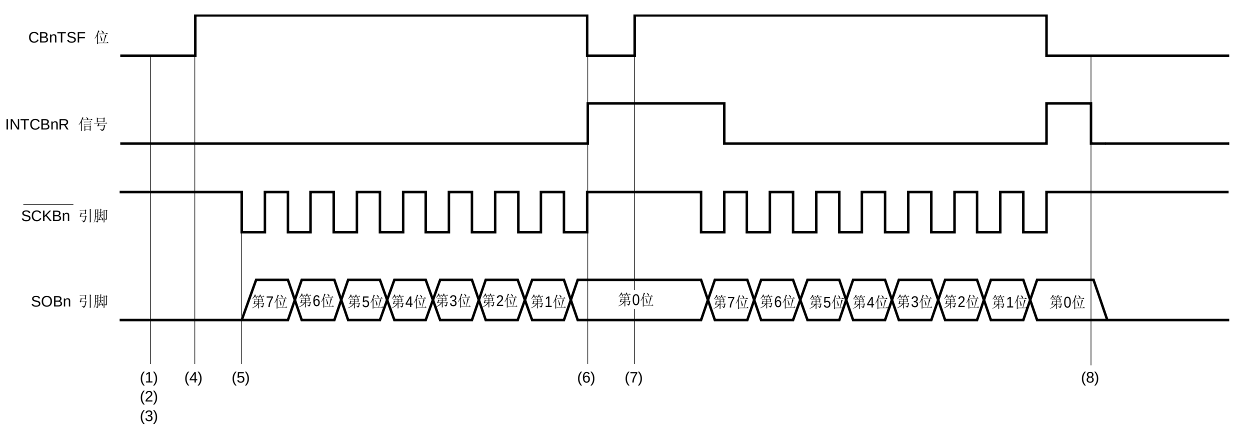


备注 **1.** 虚线部分表示硬件处理。

2. 图中的数字对应于（**2**） **操作时序**中的数字。

3. n = 0 至 2

（2）操作时序



（1）向 CBnCTL1 寄存器写入 07H，选择通信类型 1，通信时钟（f_{CCLK}） = 外部时钟 （$\overline{\text{SCKBn}}$），并设置为从机模式。

（2）向 CBnCTL2 寄存器写入 00H，设定传输数据长度为 8 位。

（3）向 CBnCTL0 寄存器写入 C1H，并在使能通信时钟（f_{CCLK}）的同时，选择为发送模式和 MSB 先行方式。

（4）将发送数据写入 CBnTX 寄存器，使得 CBnSTR.CBnTSF 位为 1，设备等待串行时钟输入。

（5）当串行时钟输入时，待发送的数据与串行时钟同步，由 SOBn 引脚输出。

（6）当 CBnCTL2 寄存器设定的传输数据长度发送完毕时，停止串行时钟的输入和发送数据的输出，在串行时钟最末边沿产生接收结束中断请求信号（INTCBnR），并将 CBnTSF 位清为 0。

（7）若要继续进行发送，在 INTCBnR 信号产生后，将发送数据写入 CBnTX 寄存器，并等待串行时钟的输入。

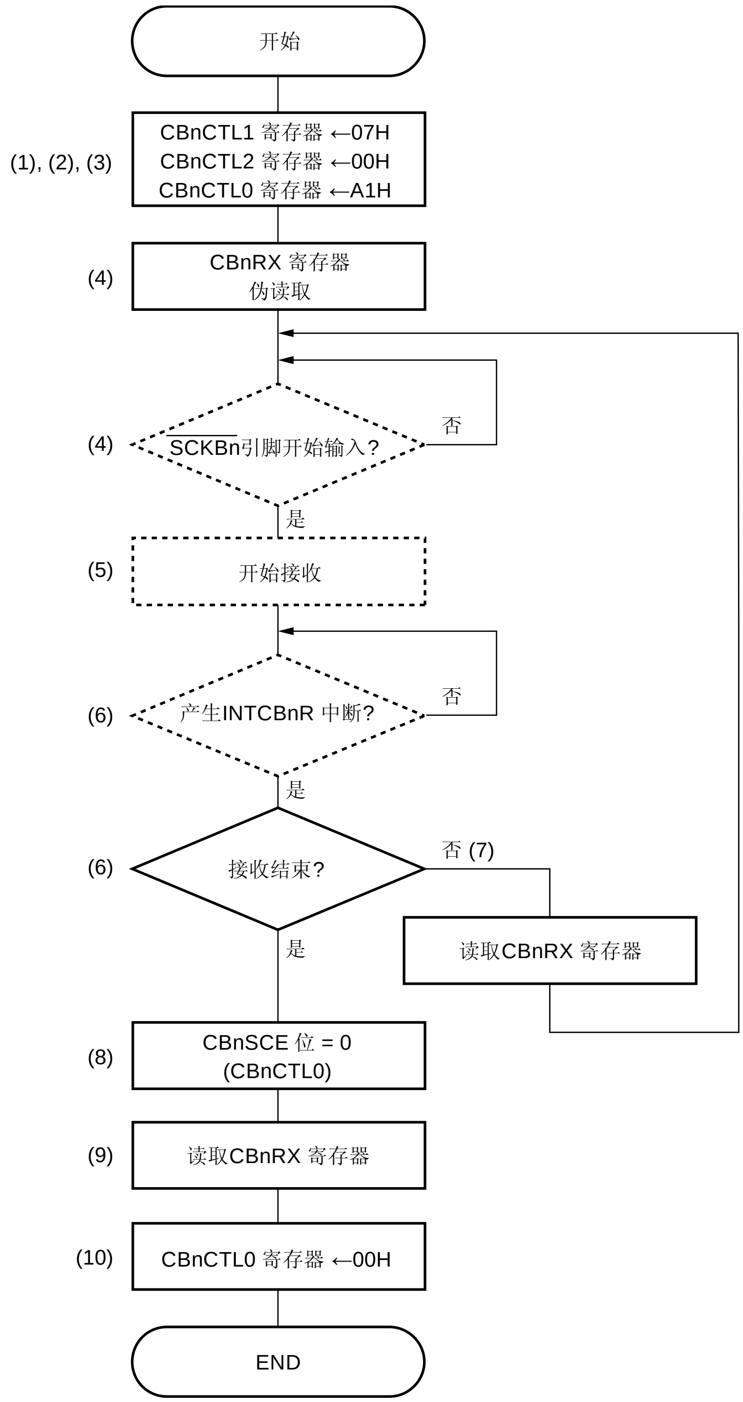
（8）若要结束发送，设置 CBnCTL0.CBnPWR 位= 0 以及 CBnCTL0.CBnTXE 位= 0。

备注　　n = 0 至 2

16.5.5 单次传输模式 （从机模式，接收模式）

MSB 先行 （CBnCTL0.CBnDIR 位= 0），通信类型 1 （CBnCTL1.CBnCKP 和 CBnCTL1.CBnDAP 位= 00），通信时钟（f_{CCLK}） = 外部时钟（$\overline{\text{SCKBn}}$） （CBnCTL1.CBnCKS2 至 CBnCTL1.CBnCKS0 位 = 111），传输数据长度= 8 位（CBnCTL2.CBnCL3 至 CBnCTL2.CBnCL0 位= 0000）。

（1） 操作流程

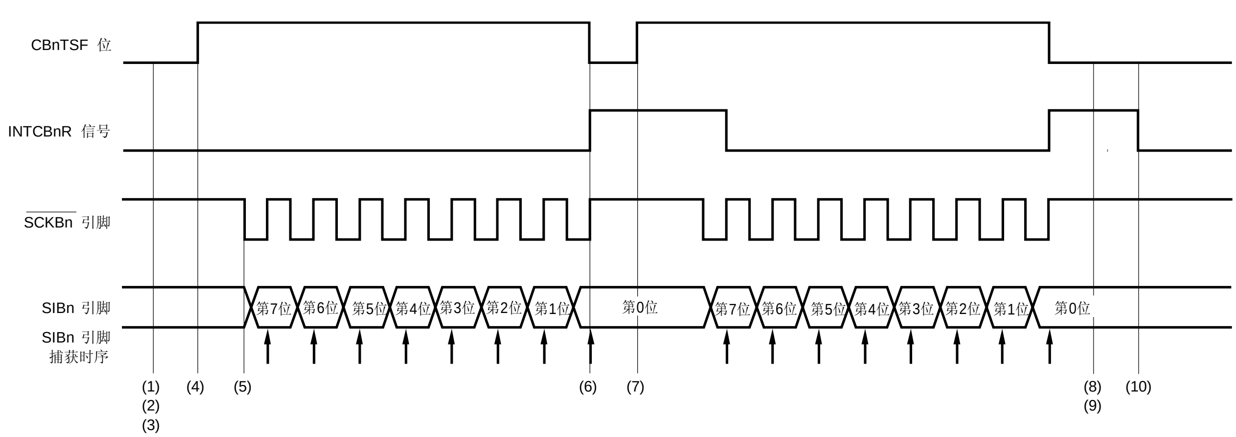


备注 **1.** 虚线部分表示硬件处理。

2. 图中的数字对应于**（2） 操作时序**中的数字。

3. n = 0 至 2

（2） 操作时序



（1） 向CBnCTL1 寄存器写入 07H，选择通信类型 1，通信时钟（f_{CCLK}） = 外部时钟 （$\overline{\text{SCKBn}}$），并设置为从机模式。

（2） 向 CBnCTL2 寄存器写入 00H，设定传输数据长度为 8 位。

（3） 向 CBnCTL0 寄存器写入 A1H，并在使能通信时钟（f_{CCLK}）的同时，选择为接收模式和 MSB 先行方式。

（4） 通过对 CBnRX 寄存器进行伪读取使得 CBnSTR.CBnTSF 位置为 1，设备等待串行时钟输入。

（5） 当串行时钟输入时，与串行时钟同步捕获 SIBn 引脚输入的接收数据。

（6） 当 CBnCTL2 寄存器设定的传输数据长度接收完毕时，停止串行时钟输入和数据捕获，在串行时钟最末边沿产生接收结束中断请求信号（INTCBnR），并将 CBnTSF 位清为 0。

（7） 若要继续接收，在 INTCBnR 信号产生后，保持 CBnCTL0.CBnSCE 位= 1 ，读取 CBnRX 寄存器，并等待串行时钟输入。

（8） 若要结束接收，写入 CBnSCE 位= 0。

（9） 读取 CBnRX 寄存器。

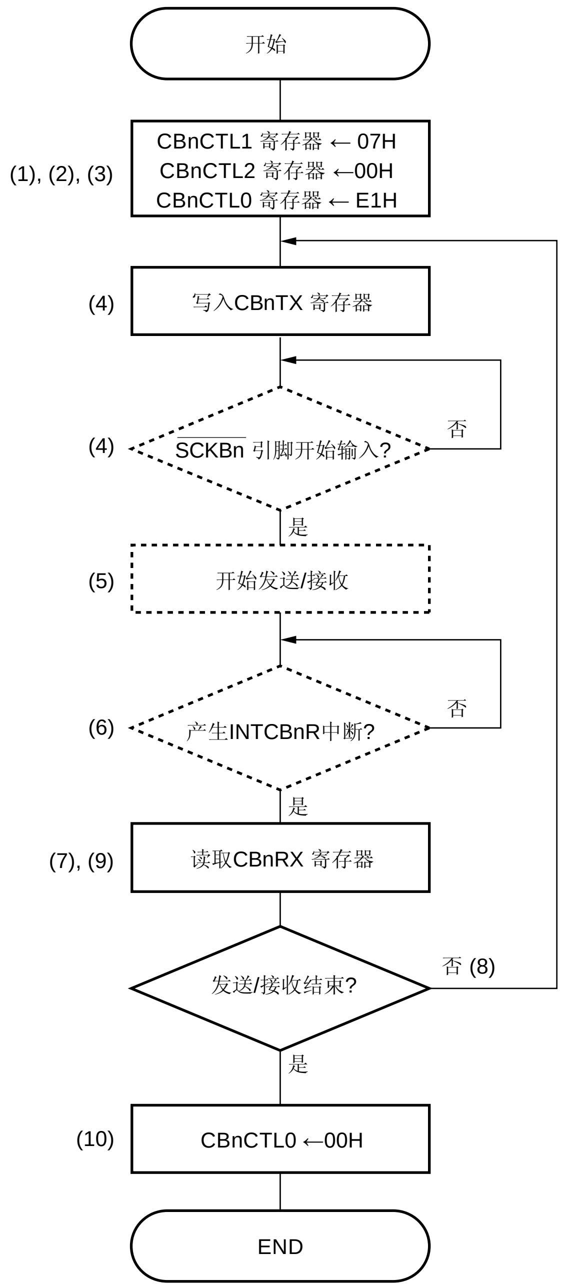
（10）若要结束接收，设置 CBnCTL0.CBnPWR 位= 0 以及 CBnCTL0.CBnRXE 位= 0。

备注 n = 0 至 2

16.5.6 单次传输模式（从机模式，发送/接收模式）

MSB 先行 （CBnCTL0.CBnDIR 位= 0），通信类型 1 （CBnCTL1.CBnCKP 和 CBnCTL1.CBnDAP 位= 00），通信时钟（f_{CCLK}） = 外部时钟（$\overline{SCKBn}$）（CBnCTL1.CBnCKS2 至 CBnCTL1.CBnCKS0 位 = 111），传输数据长度= 8 位（CBnCTL2.CBnCL3 至 CBnCTL2.CBnCL0 位= 0000）。

（1）　操作流程

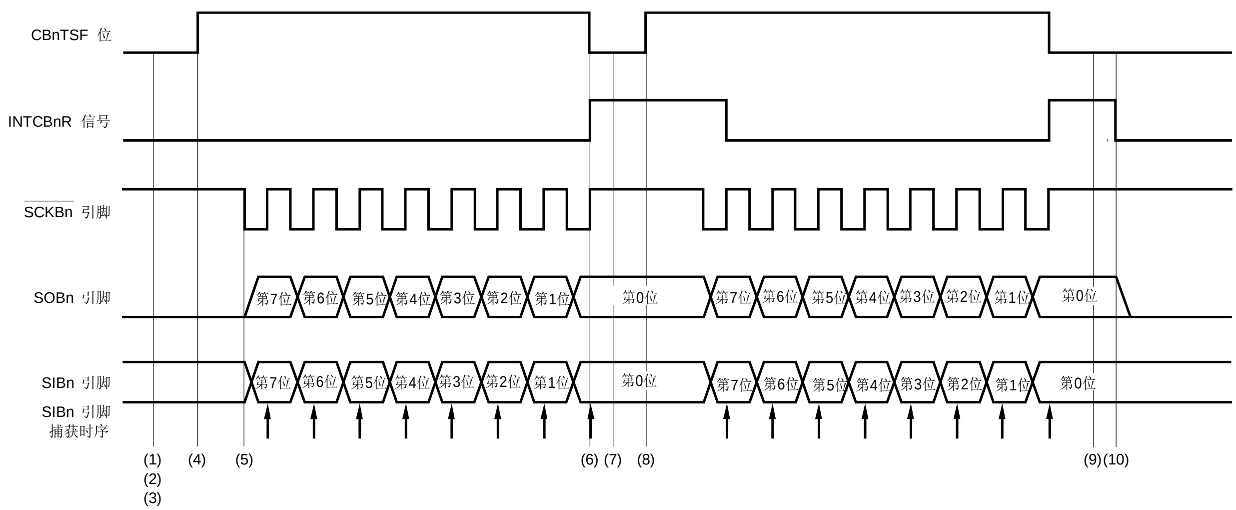


备注　**1.** 虚线部分表示硬件处理。

2. 图中的数字对应于（**2**）　**操作时序**中的数字。

3. n = 0 至 2

（2）操作时序



（1） 向 CBnCTL1 寄存器写入 07H，选择通信类型 1，通信时钟（f_{CCLK}） = 外部时钟 （$\overline{SCKBn}$），并设置为从机模式。

（2） 向 CBnCTL2 寄存器写入 00H，设定传输数据长度为 8 位。

（3） 向 CBnCTL0 寄存器写入 E1H，并在使能通信时钟（f_{CCLK}）的同时，选择为发送/接收模式和 MSB 先行方式。

（4）将待发送数据写入 CBnTX 寄存器，使得 CBnSTR.CBnTSF 位置为 1，设备等待串行时钟的输入。

（5） 当串行时钟输入时，待发送数据与串行时钟同步，由 SOBn 引脚输出，并且同步捕获 SIBn 引脚的接收数据。

（6） 当 CBnCTL2 寄存器设定的传输数据长度发送/接收完毕时，停止串行时钟输入、发送数据输出以及数据捕获，在串行时钟最末边沿产生接收结束中断请求信号 （INTCBnR），并将 CBnTSF 位清为 0。

（7） 读取 CBnRX 寄存器。

（8） 若要继续发送/接收，将待发送数据再次写入 CBnTX 寄存器，并等待串行时钟输入。

（9） 读取 CBnRX 寄存器。

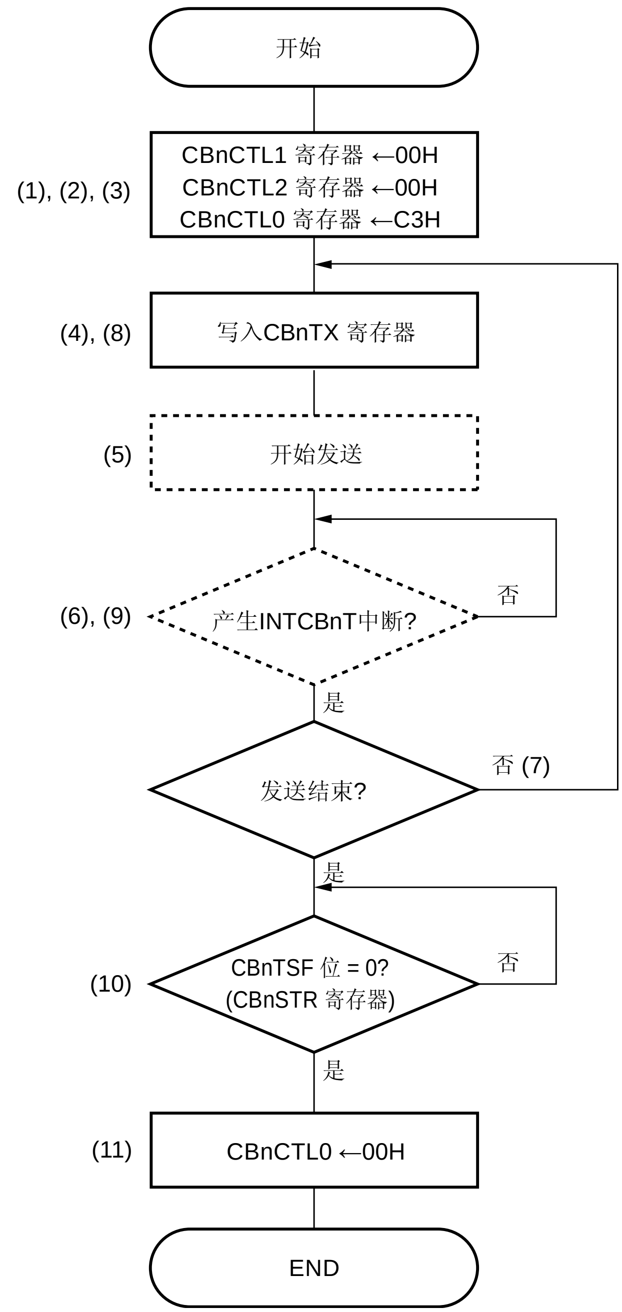
（10）若要结束数据发送/接收，设置 CBnCTL0.CBnPWR 位= 0、CBnCTL0.CBnTXE 位= 0 以及 CBnCTL0.CBnRXE 位= 0。

备注　　n = 0 至 2

16.5.7 连续传输模式（主机模式，发送模式）

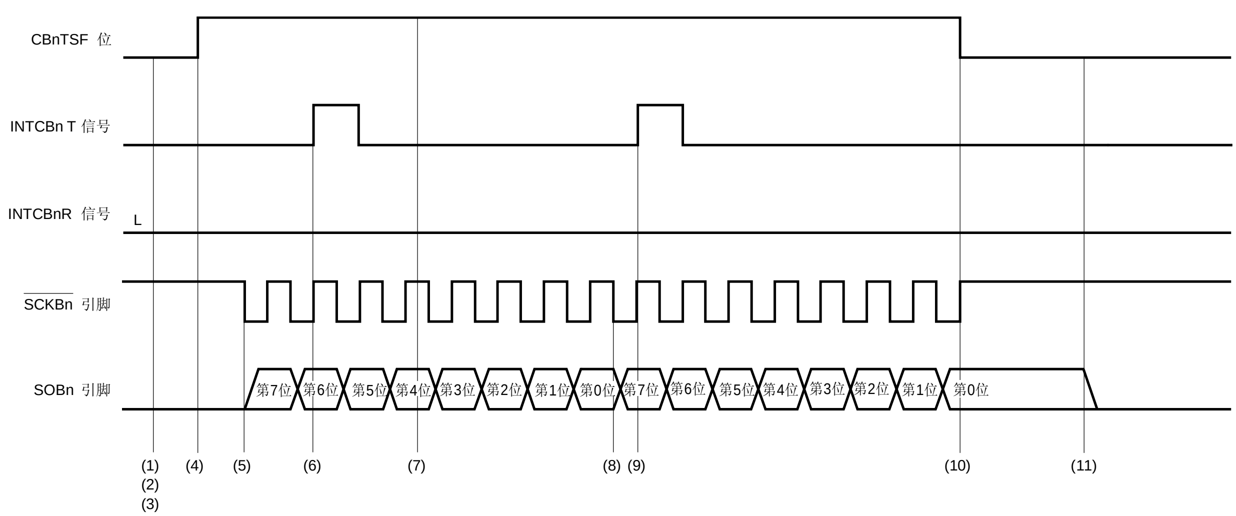
MSB 先行（CBnCTL0.CBnDIR 位= 0），通信类型 1（CBnCTL1.CBnCKP 和 CBnCTL1.CBnDAP 位= 00），通信时钟（f_{CCLK}）= $f_{XX}/4$（CBnCTL1.CBnCKS2 至 CBnCTL1.CBnCKS0 位 = 000），传输数据长度= 8 位（CBnCTL2.CBnCL3 至 CBnCTL2.CBnCL0 位= 0000）。

（1） 操作流程



备注 **1.** 虚线部分表示硬件处理。
2. 图中的数字对应于 **（2） 操作时序**中的数字。
3. n = 0 至 2

（2） 操作时序



（1） 向 CBnCTL1 寄存器写入 00H，选择通信类型 1，通信时钟（f_{CCLK}） = f_{XX}/4，并设置为主机模式。

（2） 向 CBnCTL2 寄存器写入 00H，设定传输数据长度为 8 位。

（3） 向 CBnCTL0 寄存器写入 C3H，在使能通信时钟（f_{CCLK}）操作的同时，选择为发送模式、MSB 先行以及连续传输模式。

（4）将待发送数据写入 CBnTX 寄存器使得 CBnSTR.CBnTSF 位置为 1，启动发送。

（5） 当发送开始后，输出串行时钟至 $\overline{\text{SCKBn}}$ 引脚，待发送数据与串行时钟同步，由 SOBn 引脚输出。

（6） 当发送数据从 CBnTX 寄存器传输至移位寄存器结束，且允许写入 CBnTX 寄存器时，产生发送允许中断请求信号 （INTCBnT） 。

（7） 若要连续发送，在产生 INTCBnT 信号之后，再次将发送数据写入 CBnTX 寄存器。

（8） 当通信完成之前，新的发送数据已经写入 CBnTX 寄存器，则在本次通信结束后立即开始下一次通信。

（9） 从 CBnTX 寄存器传输发送数据至移位寄存器完毕，且产生 INTCBnT 信号。若要结束当前的连续发送传输，不要写入 CBnTX 寄存器。

（10）当传输完毕之前下一个发送数据没有被写入 CBnTX 寄存器，传输结束后，则停止输入到 $\overline{\text{SCKBn}}$ 引脚的串行时钟，并将 CBnTSF 位清除为 0。

（11）若要解除发送使能状态，在检查 CBnTSF 位= 0 之后，设置 CBnCTL0.CBnPWR 位= 0 和 CBnCTL0.CBnTXE 位= 0 。

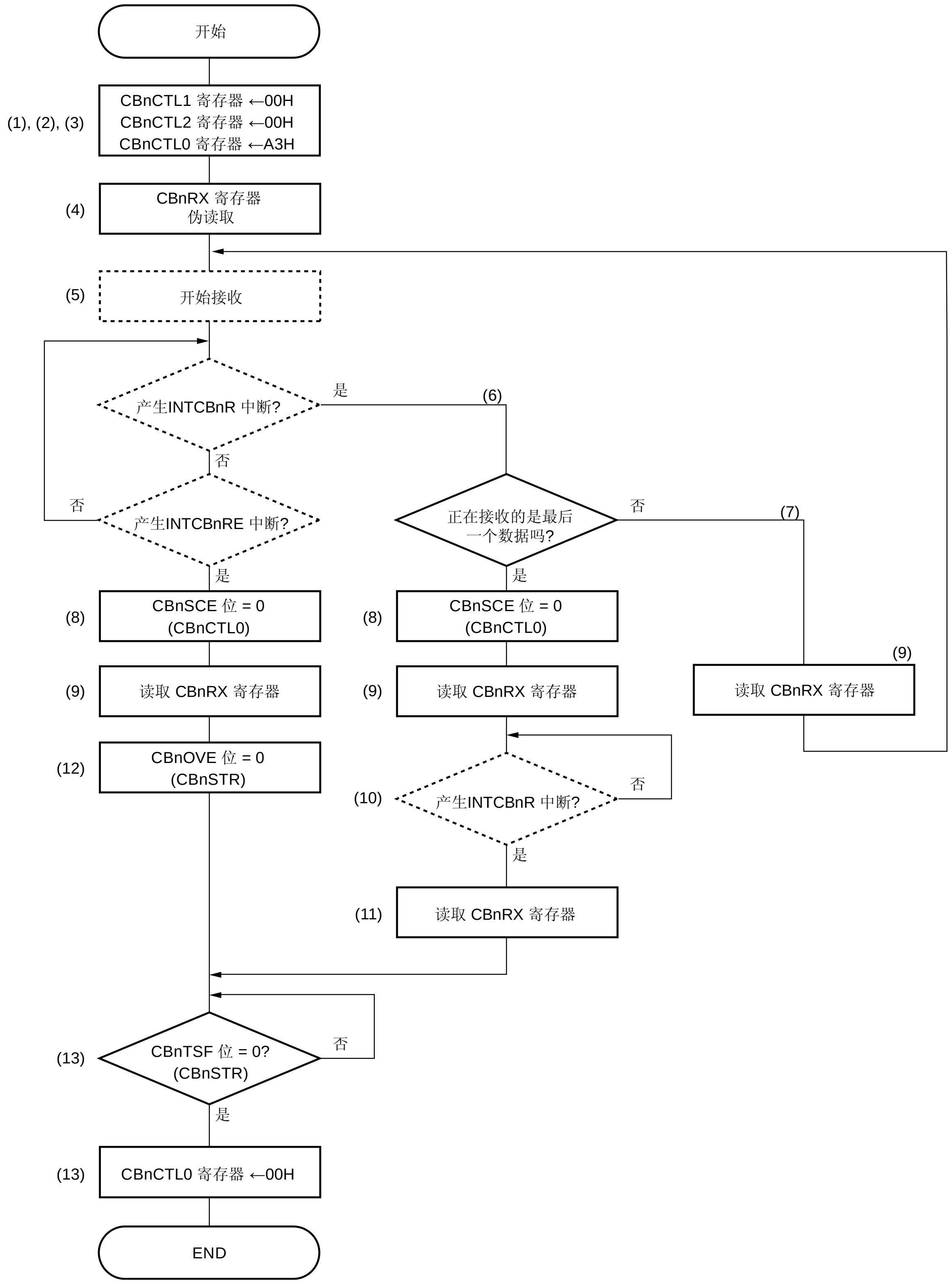
注意事项 连续发送模式下，不产生接收结束中断请求信号 （INTCBnR）。

备注 n = 0 至 2

16.5.8 连续传输模式（主机模式，接收模式）

MSB 先行（CBnCTL0.CBnDIR 位= 0），通信类型 1（CBnCTL1.CBnCKP 和 CBnCTL1.CBnDAP 位= 00），通信时钟（f_{CCLK}）= f_{XX}/4（CBnCTL1.CBnCKS2 至 CBnCTL1.CBnCKS0 位 = 000），传输数据长度= 8 位（CBnCTL2.CBnCL3 至 CBnCTL2.CBnCL0 位= 0000）。

（1） 操作流程

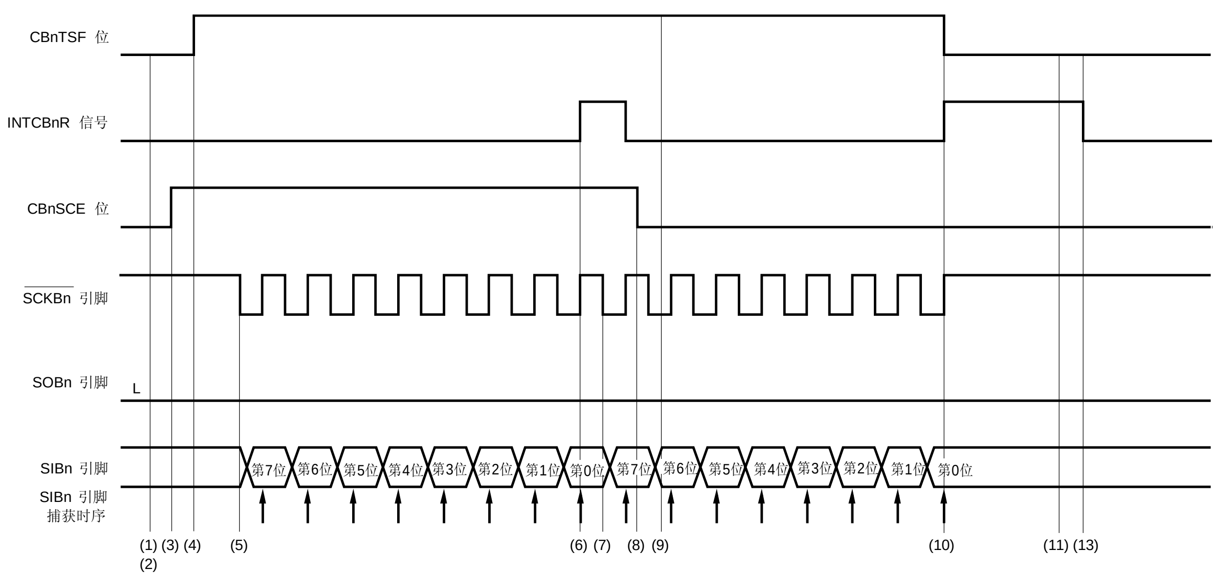


备注 **1.** 虚线部分表示硬件处理。

2. 图中的数字对应于（**2**） **操作时序**中的数字。

3. n = 0 至 2

（2） 操作时序



（1） 向 CBnCTL1 寄存器写入 00H，选择通信类型 1，通信时钟（f_{CCLK}） = f_{XX}/4，并设置为主机模式。

（2） 向 CBnCTL2 寄存器写入 00H，设定传输数据长度为 8 位。

（3） 向 CBnCTL0 寄存器写入 A3H，在使能通信时钟（f_{CCLK}）操作的同时，选择为接收模式、MSB 先行以及连续传输模式。

（4） 对 CBnRX 寄存器执行一次伪读取，使得 CBnSTR.CBnTSF 位置为 1，并启动接收。

（5） 当接收开始后，输出串行时钟至 $\overline{\text{SCKBn}}$ 引脚，并与串行时钟同步捕获 SIBn 引脚的接收数据。

（6） 当接收结束之后，产生接收结束中断请求信号（INTCBnR），允许读取 CBnRX 寄存器。

（7） 当通信结束时 CBnCTL0.CBnSCE 位= 1，则在本次通信结束之后立即开始下一次通信。

（8） 若要结束当前连续接收传输，写入 CBnSCE 位= 0。

（9） 读取 CBnRX 寄存器。

（10）当接收结束时，产生 INTCBnR 信号，并且允许读取 CBnRX 寄存器。如果在通信结束之前设置 CBnSCE 位= 0，停止输出串行时钟至 $\overline{\text{SCKBn}}$ 引脚，并清除 CBnTSF 为 0，从而结束接收操作。

（11）读取 CBnRX 寄存器。

（12）如果发生溢出错误，则 CBnSTR.CBnOVE 位= 0，并且清除错误标志。

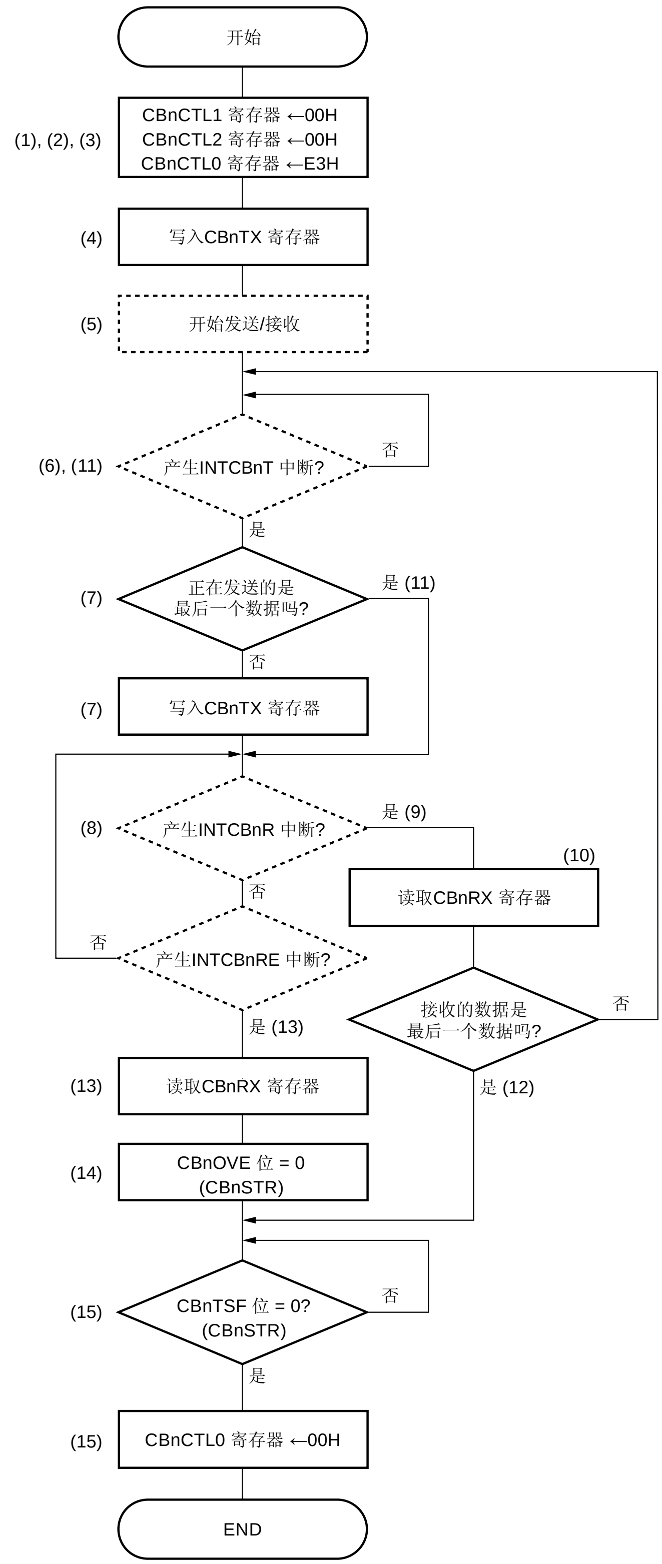
（13）若要解除接收使能状态，在检查确认 CBnTSF 位= 0 之后，设置 CBnCTL0.CBnPWR 位= 0 和 CBnCTL0.CBnRXE 位= 0 。

备注 n = 0 至 2

16.5.9 连续传输模式（主机模式，发送/接收模式）

MSB 先行 （CBnCTL0.CBnDIR 位= 0），通信类型 1 （CBnCTL1.CBnCKP 和 CBnCTL1.CBnDAP 位= 00），通信时钟（f_{CCLK}）= f_{XX}/4 （CBnCTL1.CBnCKS2 至 CBnCTL1.CBnCKS0 位 = 000），传输数据长度= 8 位（CBnCTL2.CBnCL3 至 CBnCTL2.CBnCL0 位= 0000）。

（1） 操作流程



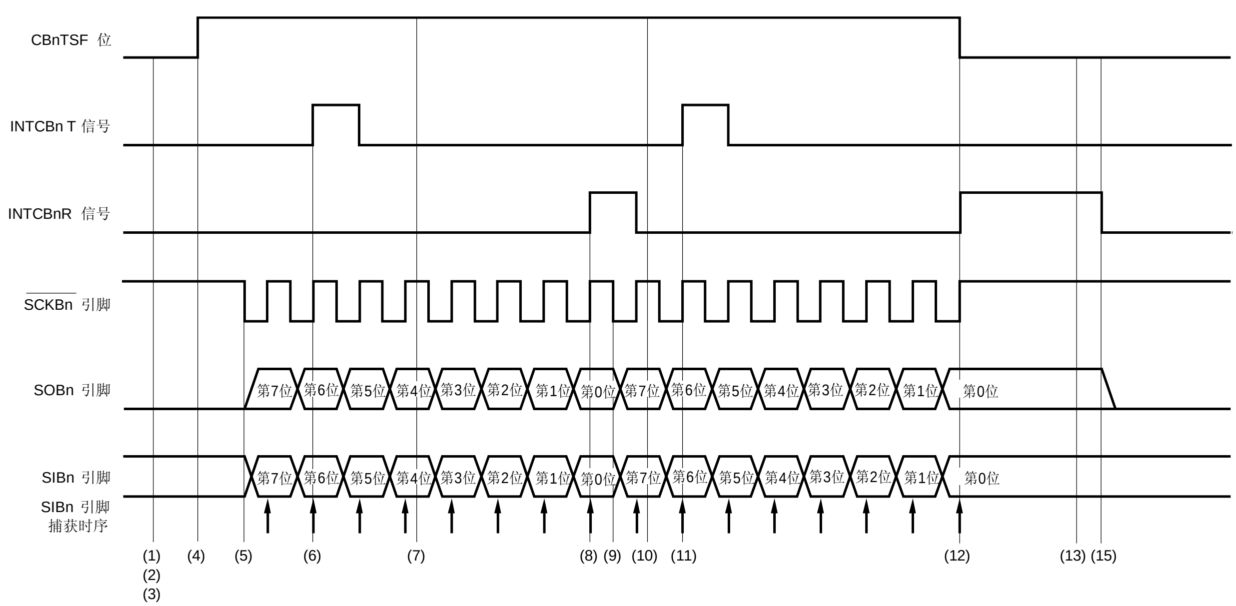
备注 **1.** 虚线部分表示硬件处理。

2. 图中的数字对应于（2） **操作时序**中的数字。

3. n = 0 至 2

（2）　操作时序

（1/2）



（1） 向 CBnCTL1 寄存器写入 00H，选择通信类型 1，通信时钟（f_{CCLK}） = f_{XX}/4，并设置为主机模式。

（2） 向 CBnCTL2 寄存器写入 00H，设定传输数据长度为 8 位。

（3） 向 CBnCTL0 寄存器写入 E3H，并且在使能通信时钟（f_{CCLK}）的同时，选择为发送/接收模式、MSB 先行以及连续传输模式。

（4） 将待发送数据写入 CBnTX 寄存器，使得 CBnSTR.CBnTSF 位置为 1 ，启动发送/接收。

（5） 当发送/接收启动之后，输出串行时钟至 $\overline{\text{SCKBn}}$ 引脚，待发送数据与串行时钟同步，由 SOBn 引脚输出 ，并且同步捕获 SIBn 引脚的接收数据。

（6） 当发送数据从 CBnTX 寄存器传输至移位寄存器结束，且允许写入 CBnTX 寄存器时，产生发送允许中断请求信号 （INTCBnT） 。

（7） 若要连续发送/接收，在产生 INTCBnT 信号之后，再次将待发送数据写入 CBnTX 寄存器。

（8） 当一次发送/接收结束之后，产生接收结束中断请求信号 （INTCBnR）），并且允许读取 CBnRX 寄存器。

（9） 当通信结束之前，新的待发送数据已经写入 CBnTX 寄存器，则在本次通信结束之后立即开始下一次通信。

（10）读取 CBnRX 寄存器。

备注　　n = 0 至 2

（2/2）

（11）从 CBnTX 寄存器传输发送数据至移位寄存器完毕，且产生 INTCBnT 信号。若要结束当前的发送/接收传输，则不写入 CBnTX 寄存器。

（12）当传输结束之前下一个待发送数据未被写入 CBnTX 寄存器，传输结束之后，则停止输出串行时钟至 $\overline{\text{SCKBn}}$ 引脚，并将 CBnTSF 位清除为 0。

（13）发生接收错误中断请求信号（INTCBnR）时，读取 CBnRX 寄存器。

（14）如果发生溢出错误，则 CBnSTR.CBnOVE 位= 0，并且清除错误标志。

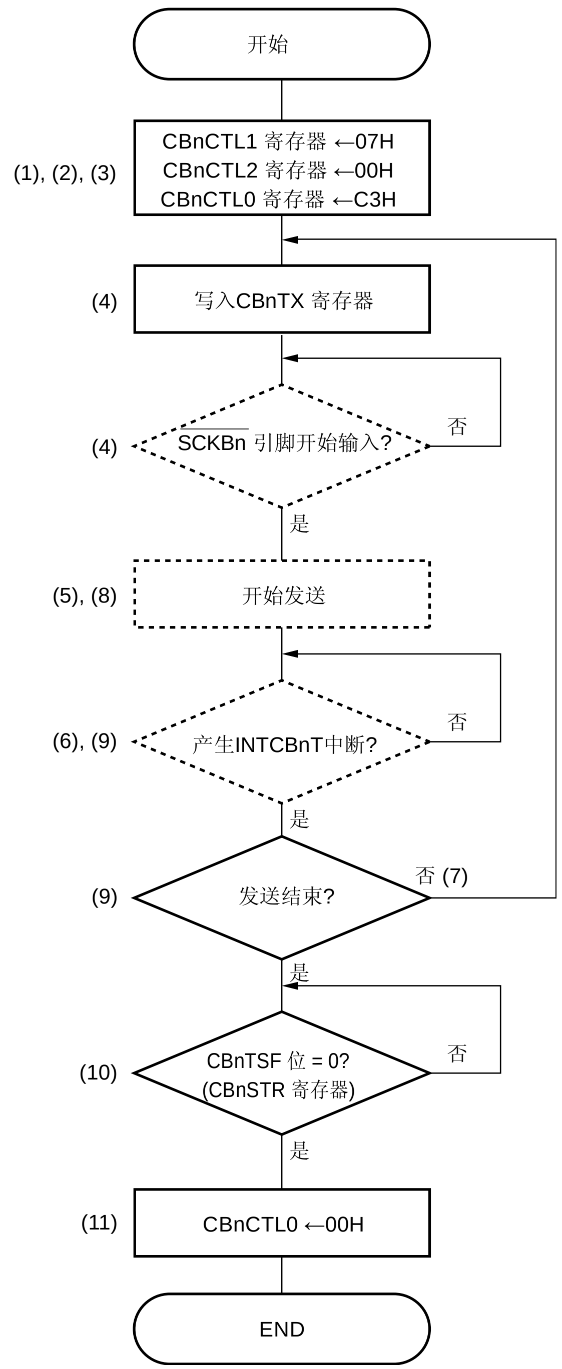
（15）若要解除发送/接收使能状态，在检查确认 CBnTSF 位= 0 之后， 设置 CBnCTL0.CBnPWR 位= 0，CBnCTL0.CBnTXE 位= 0 以及 CBnCTL0.CBnRXE 位= 0 。

备注 n = 0 至 2

16.5.10 连续传输模式（从机模式，发送模式）

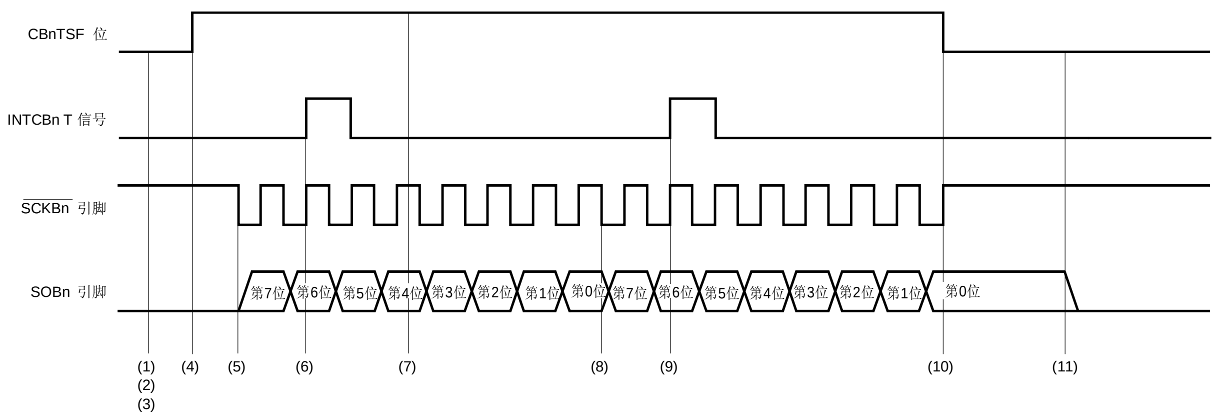
MSB 先行（CBnCTL0.CBnDIR 位= 0），通信类型 1（CBnCTL1.CBnCKP 和 CBnCTL1.CBnDAP 位= 00），通信时钟（f_{CCLK}）= 外部时钟（$\overline{SCKBn}$）（CBnCTL1.CBnCKS2 至 CBnCTL1.CBnCKS0 位 = 111），传输数据长度= 8 位（CBnCTL2.CBnCL3 至 CBnCTL2.CBnCL0 位= 0000）。

（1） 操作流程



备注
1. 虚线部分表示硬件处理。
2. 图中的数字对应于（2） **操作时序**中的数字。
3. n = 0 至 2

（2） 操作时序



（1） 向CBnCTL1 寄存器写入 07H，选择通信类型 1，通信时钟（f_{CCLK}） = 外部时钟 （$\overline{SCKBn}$），并设置为从机模式。

（2） 向 CBnCTL2 寄存器写入 00H，设定传输数据长度为 8 位。

（3） 向 CBnCTL0 寄存器写入 C3H，在使能通信时钟（f_{CCLK}）操作的同时，选择为发送模式、MSB 先行以及连续传输模式。

（4） 将待发送数据写入 CBnTX 寄存器，使得 CBnSTR.CBnTSF 位为 1，设备等待串行时钟输入。

（5） 当串行时钟输入时，待发送数据与串行时钟同步，由 SOBn 引脚输出。

（6） 当发送数据从 CBnTX 寄存器传输至移位寄存器结束，且允许写入 CBnTX 寄存器时，产生发送允许中断请求信号（INTCBnT）。

（7） 若要连续发送，在产生 INTCBnT 信号之后，再次将待发送数据写入 CBnTX 寄存器。

（8） 当 CBnCTL2 寄存器设定的传输数据长度发送结束后，有串行时钟输入，则启动连续发送。

（9） 当发送数据从 CBnTX 寄存器传输至移位寄存器结束，并且允许写入 CBnTX 寄存器，产生 INTCBnT 信号。 若要结束当前的连续发送传输，不要写入 CBnTX 寄存器。

（10）当 CBnCTL2 寄存器设定的传输数据长度对应的时钟输入时，而未写入 CBnTX 寄存器，清除 CBnTSF 位为 0，从而结束发送。

（11）若要解除发送使能状态，在检查确认 CBnTSF 位= 0 之后，设置 CBnCTL0.CBnPWR 位= 0 和 CBnCTL0.CBnTXE 位= 0 。

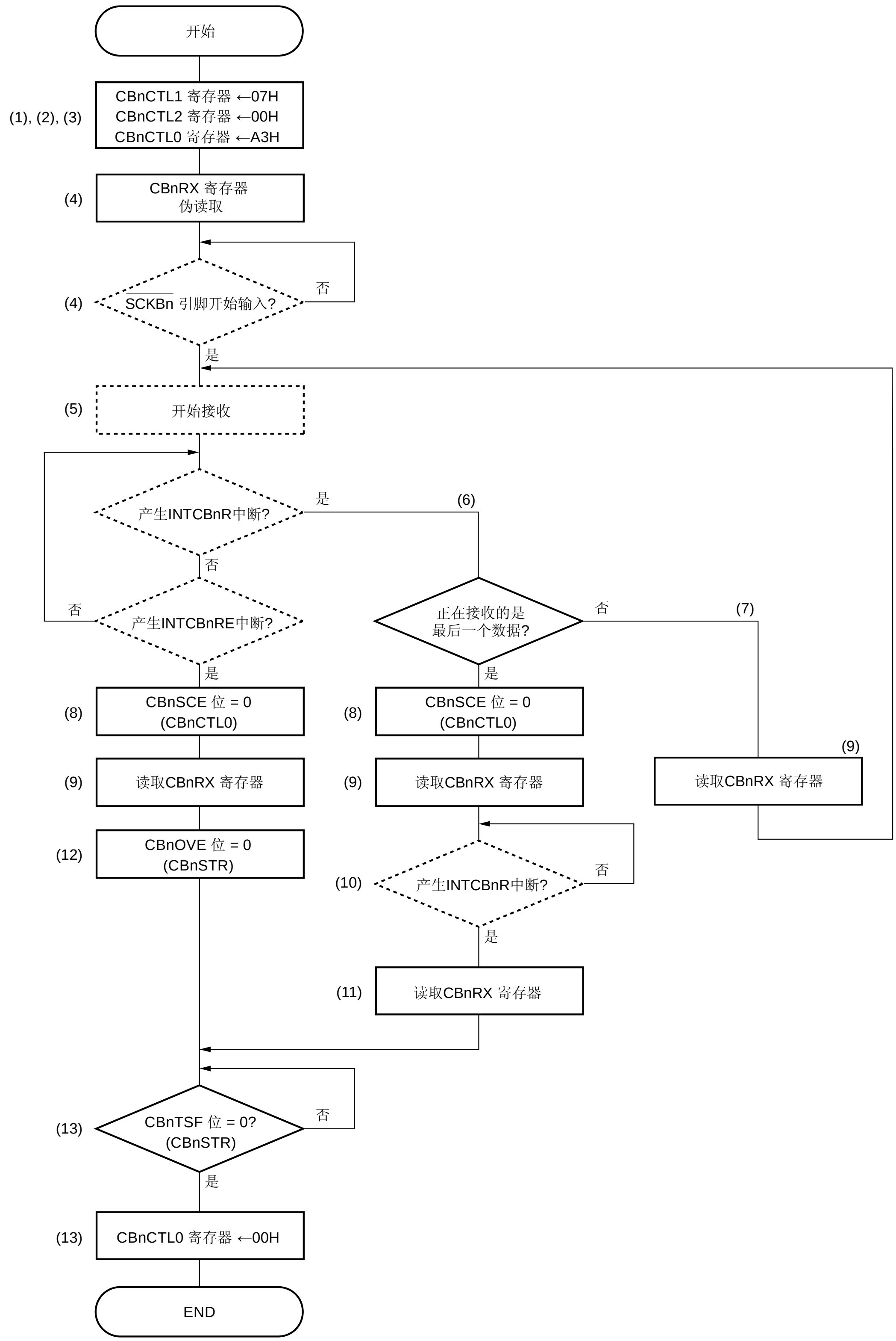
注意事项 连续发送模式下，不产生接收结束中断请求信号（INTCBnR）。

备注 n = 0 至 2

16.5.11 连续传输模式（从机模式，接收模式）

MSB 先行 （CBnCTL0.CBnDIR 位= 0），通信类型 1 （CBnCTL1.CBnCKP 和 CBnCTL1.CBnDAP 位= 00），通信时钟（f_{CCLK}） = 外部时钟（$\overline{\text{SCKBn}}$） （CBnCTL1.CBnCKS2 至 CBnCTL1.CBnCKS0 位 = 111）， 传输数据长度= 8 位（CBnCTL2.CBnCL3 至 CBnCTL2.CBnCL0 位= 0000）。

（1） 操作流程

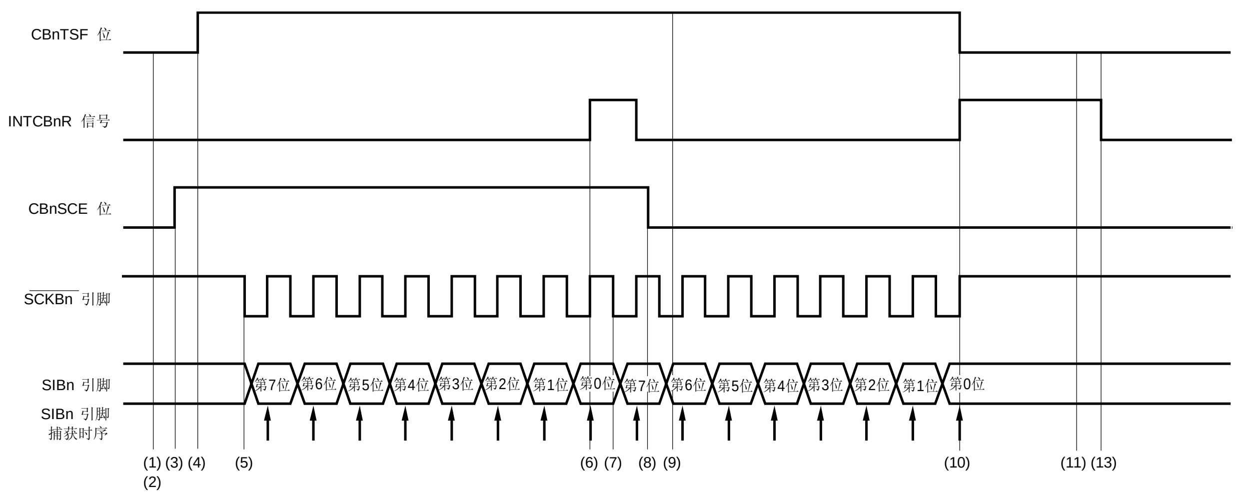


备注 **1.** 虚线部分表示硬件处理。

2. 图中的数字和（**2**） **操作时序**中的处理数字相对应。

3. n = 0 至 2

（2） 操作时序



（1） 向CBnCTL1 寄存器写入 07H，选择通信类型 1，通信时钟（f_{CCLK}） = 外部时钟 （$\overline{SCKBn}$），并设置为从机模式。

（2） 向 CBnCTL2 寄存器写入 00H，设定传输数据长度为 8 位。

（3） 向 CBnCTL0 寄存器写入 A3H，在使能通信时钟（f_{CCLK}）操作的同时，选择为接收模式、MSB 先行以及连续传输模式。

（4） 通过对 CBnRX 寄存器进行伪读取，使得 CBnSTR.CBnTSF 位为 1，设备等待串行时钟输入。

（5） 当串行时钟输入时，与串行时钟同步从 SIBn 引脚捕获接收数据。

（6） 当接收结束之后，产生接收结束中断请求信号 （INTCBnR），并且允许读取 CBnRX 寄存器。

（7）在 CBnCTL0.CBnSCE 位= 1 状态下输入串行时钟，启动连续接收。

（8） 若要结束当前连续接收传输，写入 CBnSCE 位= 0。

（9） 读取 CBnRX 寄存器。

（10）当接收完结束时，产生 INTCBnR 信号，并且允许读取 CBnRX 寄存器。如果在通信结束之前设置 CBnSCE 位= 0 ，则清除 CBnTSF 位为 0，从而结束操作。

（11）读取 CBnRX 寄存器。

（12）如果发生溢出错误， 则 CBnSTR.CBnOVE 位= 0，并且清除错误标志。

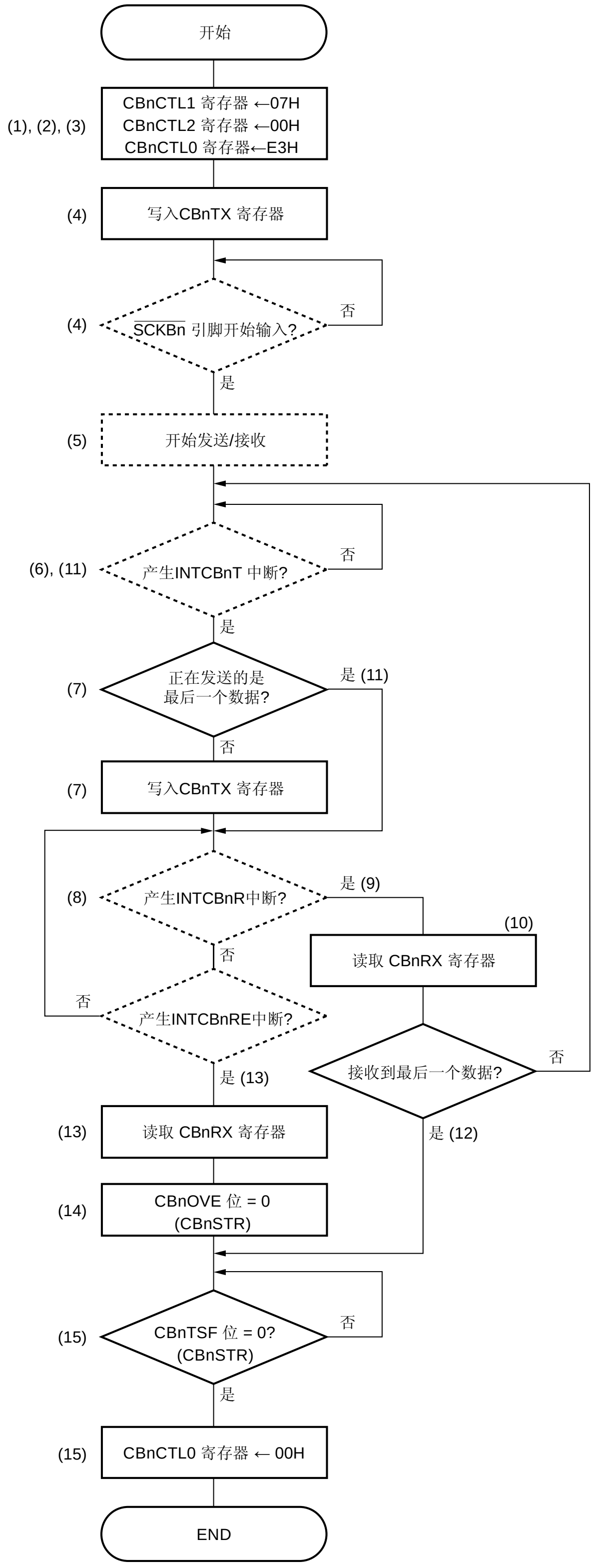
（13）若要解除接收使能状态，在检查确认 CBnTSF 位= 0 之后，设置 CBnCTL0.CBnPWR 位= 0 和 CBnCTL0.CBnRXE 位= 0 。

备注 n = 0 至 2

16.5.12 连续传输模式 （从机模式，发送/接收模式）

MSB 先行 （CBnCTL0.CBnDIR 位= 0），通信类型 1 （CBnCTL1.CBnCKP 和 CBnCTL1.CBnDAP 位= 00），通信时钟（f_{CCLK}） = 外部时钟（$\overline{\text{SCKBn}}$） （CBnCTL1.CBnCKS2 至 CBnCTL1.CBnCKS0 位 = 111），传输数据长度= 8 位 （CBnCTL2.CBnCL3 至 CBnCTL2.CBnCL0 位= 0000）。

（1） 操作流程



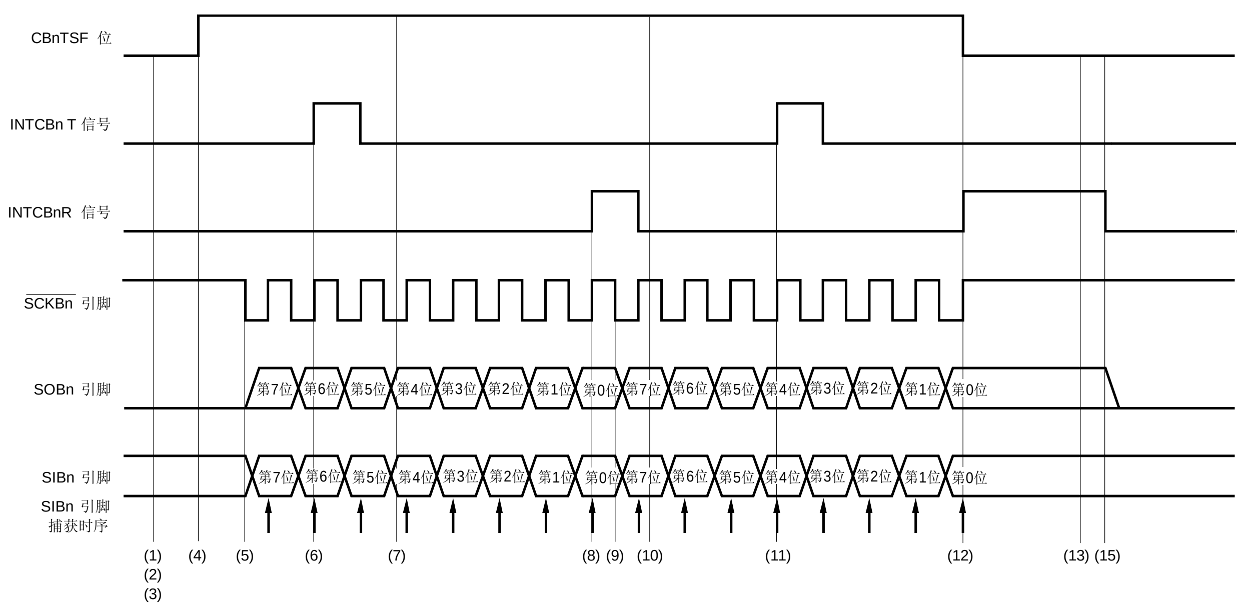
备注 **1.** 虚线部分表示硬件处理。

2. 图中的数字对应于（**2**） **操作时序**中的数字。

3. n = 0 至 2

（2） 操作时序

（1/2）



（1） 向 CBnCTL1 寄存器写入 07H，选择通信类型 1，通信时钟（f_{CCLK}） = 外部时钟 （$\overline{SCKBn}$），并设置为从机模式。

（2） 向 CBnCTL2 寄存器写入 00H，设定传输数据长度为 8 位。

（3） 向 CBnCTL0 寄存器写入 E3H，在使能通信时钟（f_{CCLK}）的同时，选择为发送/接收模式、MSB 先行以及连续传输模式。

（4） 将待发送数据写入 CBnTX 寄存器，使得 CBnSTR.CBnTSF 位置为 1，设备等待串行时钟输入。

（5） 当串行时钟输入时，与串行时钟同步，输出发送数据至 SOBn 引脚，并且捕获 SIBn 引脚的接收数据。

（6） 当发送数据从 CBnTX 寄存器传输至移位寄存器结束，且允许写入 CBnTX 寄存器时，产生发送允许中断请求信号 （INTCBnT） 。

（7） 若要连续发送，在产生 INTCBnT 信号之后，再次将发送数据写入 CBnTX 寄存器。

（8）当 CBnCTL2 寄存器设定的传输数据长度接收结束后，产生接收完成中断请求信号（INTCBnR），并且允许读取 CBnRX 寄存器。

（9） 当串行时钟连续输入，启动连续发送/接收。

（10）读取 CBnRX 寄存器。

（11）当发送数据从 CBnTX 寄存器至移位寄存器传输结束，并且允许写入 CBnTX 寄存器，产生 INTCBnT 信号。若要结束当前的发送/接收传输，则不写入 CBnTX 寄存器。

备注 n = 0 至 2

（2/2）

（12）当 CBnCTL2 设定的传输数据长度相应的时钟输入时，而未写入 CBnTX 寄存器，则产生 INTCBnR 信号。 将 CBnTSF 位清除为 0，从而结束发送/接收。
（13）产生接收错误中断请求信号（INTCBnRE）信号时，读取 CBnRX 寄存器。
（14）如果发生溢出错误，则 CBnSTR.CBnOVE 位= 0，并且清除错误标志。
（15）若要解除发送/接收使能状态，在检查确认 CBnTSF 位= 0 之后， 设置 CBnCTL0.CBnPWR 位= 0，CBnCTL0.CBnTXE 位= 0 以及 CBnCTL0.CBnRXE 位= 0 。

备注　　n = 0 至 2

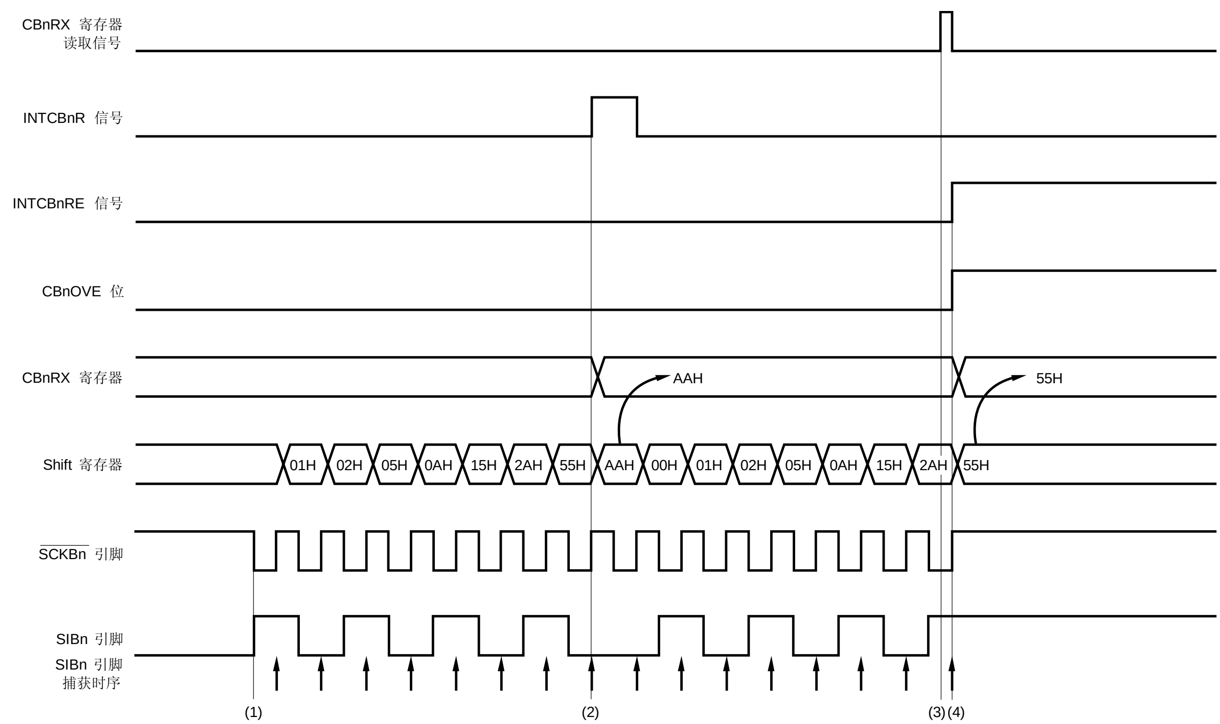
16.5.13 接收错误

连续传输模式下，在接收使能（CBnCTL0.CBnRXE 位= 1）时执行传输，在读取 CBnRX 寄存器之前下一次接收操作结束，则产生接收错误中断请求信号（INTCBnRE），并且过载错误标志（CBnSTR.CBnOVE）被置为 1。

即使发生了过载错误，由于 CBnRX 已经更新，先前接收到的数据丢失。即使发生了接收错误，如果在读取 CBnRX 寄存器之前下次接收结束，则再次产生 INTCBnRE 信号。

为了避免发生过载错误，从产生 INTCBnR 信号到采样下次接收最末位数据之前的半个时钟，结束对 CBnRX 寄存器的读取。

（1） 操作时序



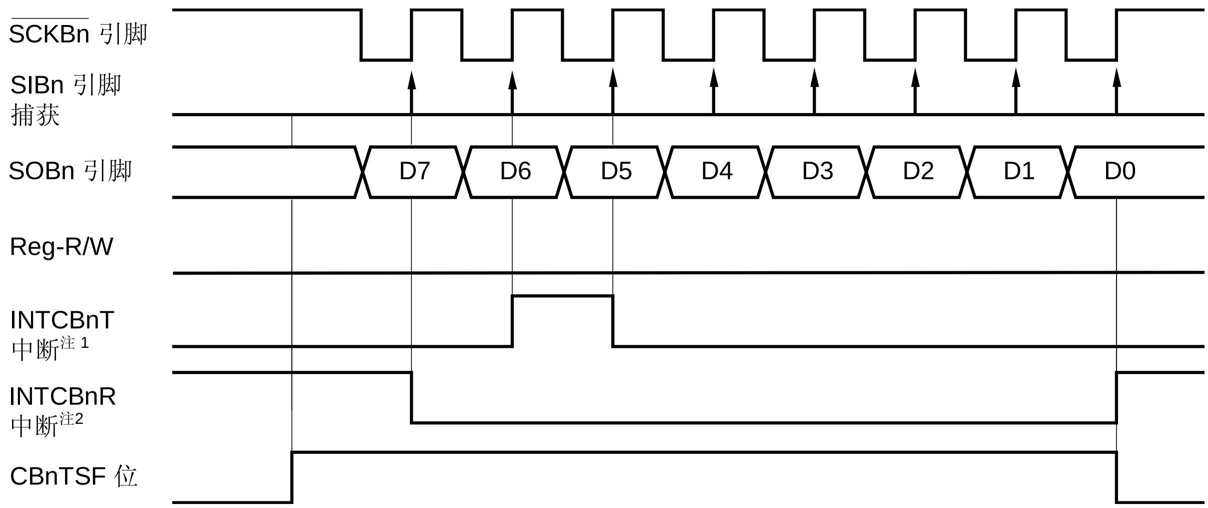
（1） 启动连续传输。
（2） 结束首次传输。
（3） 直到第二次传输结束前的半个时钟，没有读取 CBnRX 寄存器。
（4）当发生过载错误时，产生接收错误中断请求信号（INTCBnRE），过载错误标志（CBnSTR.CBnOVE）被置位。重写接收数据。

备注 n = 0 至 2

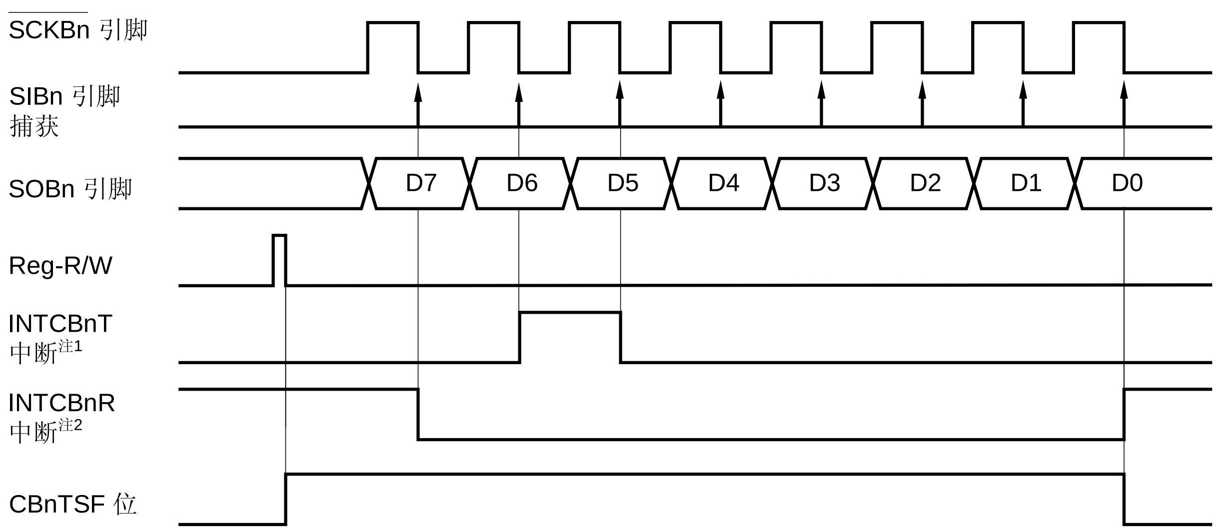
16.5.14 时钟时序

（1/2）

（i） 通信类型 1（CBnCKP 和 CBnDAP 位 = 00）



（ii） 通信类型 3 （CBnCKP 和 CBnDAP 位 = 10）



注 **1.** 在连续发送模式或连续发送/接收模式下，当写入 CBnTX 寄存器的数据传输到数据移位寄存器时，INTCBnT 中断置位。在单次发送模式或单次发送/接收模式下，不产生 INTCBnT 中断请求信号，但是在通信结束时产生 INTCBnR 中断请求信号。

2. 如果接收正确结束，将产生 INTCBnR 中断，而且在接收使能的情况下，接收数据在 CBnRX 寄存器内就绪。在单次模式下，即使在发送模式下，仍然在通信结束时产生 INTCBnR 中断请求信号。

注意事项 **在单次传输模式下，CBnTSF 位置 1 时，忽略对 CBnTX 寄存器的写入。这对传输过程的操作没有影响。**

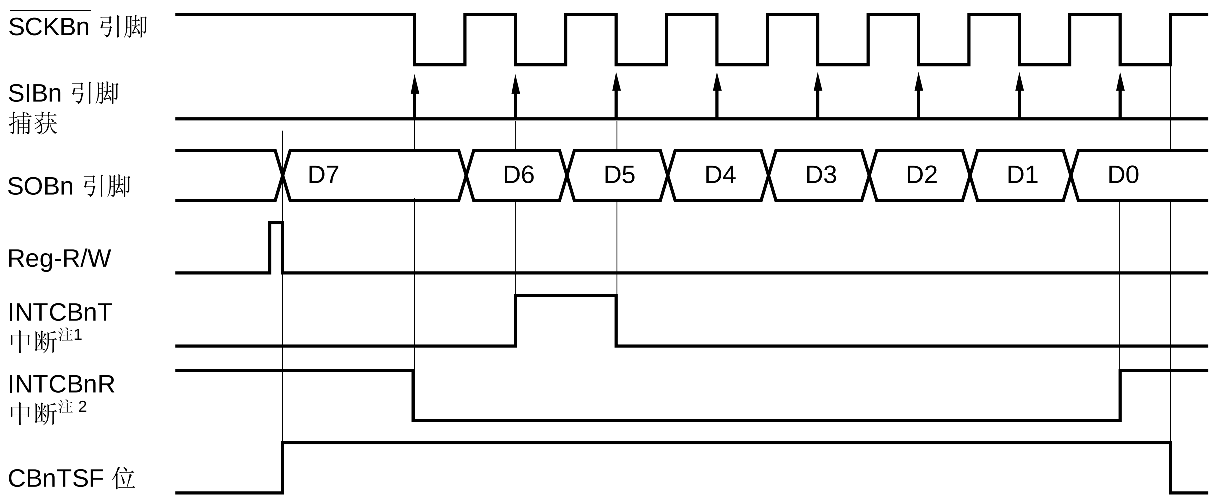
例如，通过产生的 INTCBnR 信号启动 DMA 时，若写入下一个数据到 CBnTX 寄存器，因为 CBnTSF 位为 1，不能传输写入数据。

对于这样的应用，应该使用连续传输模式，而不要使用单次传输模式。

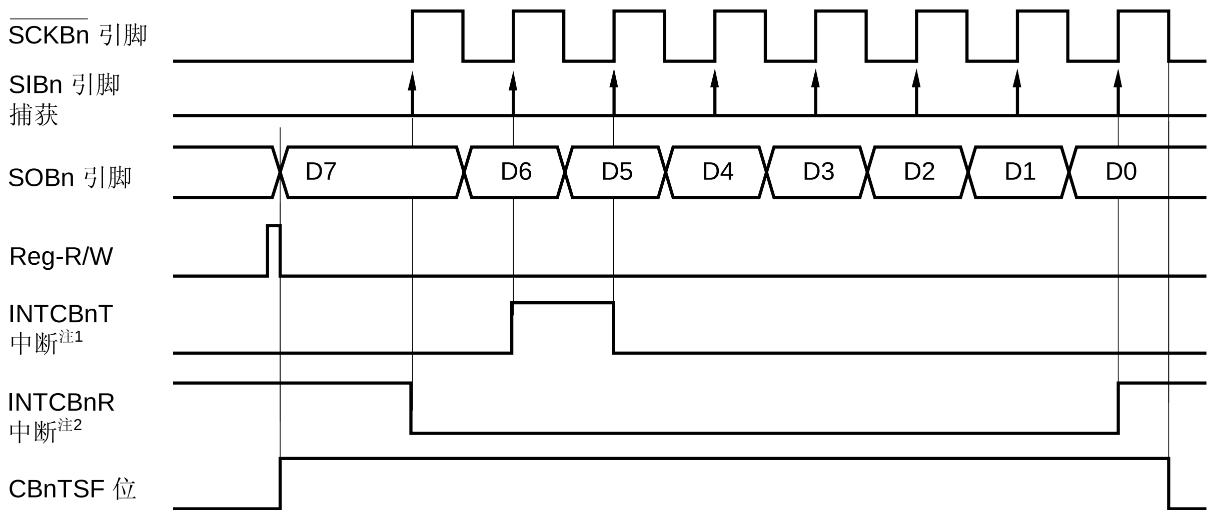
备注 n = 0 至 2

（2/2）

（iii） 通信类型 2 （CBnCKP 和 CBnDAP 位 = 01）



（iv） 通信类型 4 （CBnCKP 和 CBnDAP 位 = 11）



注 **1.** 在连续发送模式或连续发送/接收模式下，当写入 CBnTX 寄存器的数据传输到数据移位寄存器时，INTCBnT 中断置位。在单次发送模式或单次发送/接收模式下，不产生 INTCBnT 中断请求信号，但是在通信结束时产生 INTCBnR 中断请求信号。

2. 如果接收正确结束，将产生 INTCBnR 中断，而且在接收使能的情况下，接收数据在 CBnRX 寄存器内就绪。在单次模式下，即使在发送模式下，仍然在通信结束时产生 INTCBnR 中断请求信号。

注意事项 **在单次传输模式下，当 CBnTSF 位置为 1 时，忽略对 CBnTX 寄存器的写入。这对传输过程的操作没有影响。**

例如，通过产生的 INTCBnR 信号启动 DMA 时，若写入下一个数据到 CBnTX 寄存器，因为 CBnTSF 位为 1，不能传输写入数据。

对于这样的应用，应该使用连续传输模式，而不要使用单次传输模式。

备注 n = 0 至 2

16.6 输出引脚

（1） $\overline{\text{SCKBn}}$ 引脚

当 CSIBn 操作禁止 （CBnCTL0.CBnPWR 位= 0）时，$\overline{\text{SCKBn}}$ 引脚输出状态如下所示。

备注 n = 0 至 2

CBnCKP	CBnCKS2	CBnCKS1	CBnCKS0	$\overline{\text{SCKBn}}$ 引脚输出
0	1	1	1	高阻抗状态
	其它			固定为高电平
1	1	1	1	高阻抗状态
	其它			固定为低电平

备注 如果 CBnCTL1.CBnCKP 和 CBnCTL1.CBnCKS2 至 CBnCTL1.CBnCKS0 位中有任何一位重写，$\overline{\text{SCKBn}}$ 引脚的输出电平改变。

（2）SOBn 引脚

当 CSIBn 操作禁止（CBnPWR 位= 0）时，SOBn 引脚输出状态如下所示。

备注 n = 0 至 2

CBnTXE	CBnDAP	CBnDIR	SOBn 引脚输出
0	×	×	固定为低电平
1	0	×	SOBn 锁存值（低电平）。
	1	0	CBnTX0 值 （MSB）
		1	CBnTX0 值 （LSB）

备注 **1.** 如果 **CBnCTL0.CBnTXE**、**CBnCTL0.CBnDIR** 或 **CBnCTL1.CBnDAP** 位中有任何一位重写，**SOBn** 引脚输出会改变。

2. ×：无需理会

要使用I²C总线功能，P30/SCL和P31/SDA引脚用于串行发送/接收数据，并将它们设置为N沟道漏极开路输出。

在V850E/IF3 和V850E/IG3 中，提供一个通道的I²C 总线。

17.1 I²C和UARTA1 的模式转换

在V850E/IF3 和V850E/IG3 中，I²C和UARTA1 是相同引脚的复用功能，这些引脚不能同时使用。要在I²C和UARTA1 和之间切换，必须预先设置PMC3、PFC3 和PFCE3 寄存器。

注意事项 **如果发送或接收期间切换这些功能，则I²C和UARTA1 相关的发送和接收操作得不到保证。确保禁止未使用的功能。**

图 17-1. I²C和UARTA1 模式转换设置

复位后：00H R/W 地址：FFFFF446H

	7	6	5	4	3	2	1	0
PMC3	PMC37	PMC36	PMC35	PMC34	PMC33	PMC32	PMC31	PMC30

复位后：00H R/W 地址：FFFFF466H

	7	6	5	4	3	2	1	0
PFC3	PFC37	PFC36	PFC35	PFC34	PFC33	PFC32	PFC31	PFC30

复位后：00H R/W 地址：FFFFF706H

	7	6	5	4	3	2	1	0
PFCE3	PFCE37	PFCE36	PFCE35	PFCE34	0	PFCE32	PFCE31	PFCE30

PMC31	PFCE31	PFC31	P31 引脚复用功能的说明
0	×	×	I/O 端口
1	0	0	TXDA1 输出
1	0	1	SDA I/O
1	1	0	禁止设置
1	1	1	禁止设置

PMC30	PFCE30	PFC30	P30 引脚复用功能的说明
0	×	×	I/O 端口
1	0	0	RXDA1 输入
1	0	1	SCL I/O
1	1	0	禁止设置
1	1	1	禁止设置

备注 × = 无需理会

17.2 特性

I²C具有下列两种模式：

- 操作停止模式
- I²C （IC内部）总线模式（支持多主设备）

（1） 操作停止模式

在该模式下，不执行串行传输，因此可以用来降低功耗。

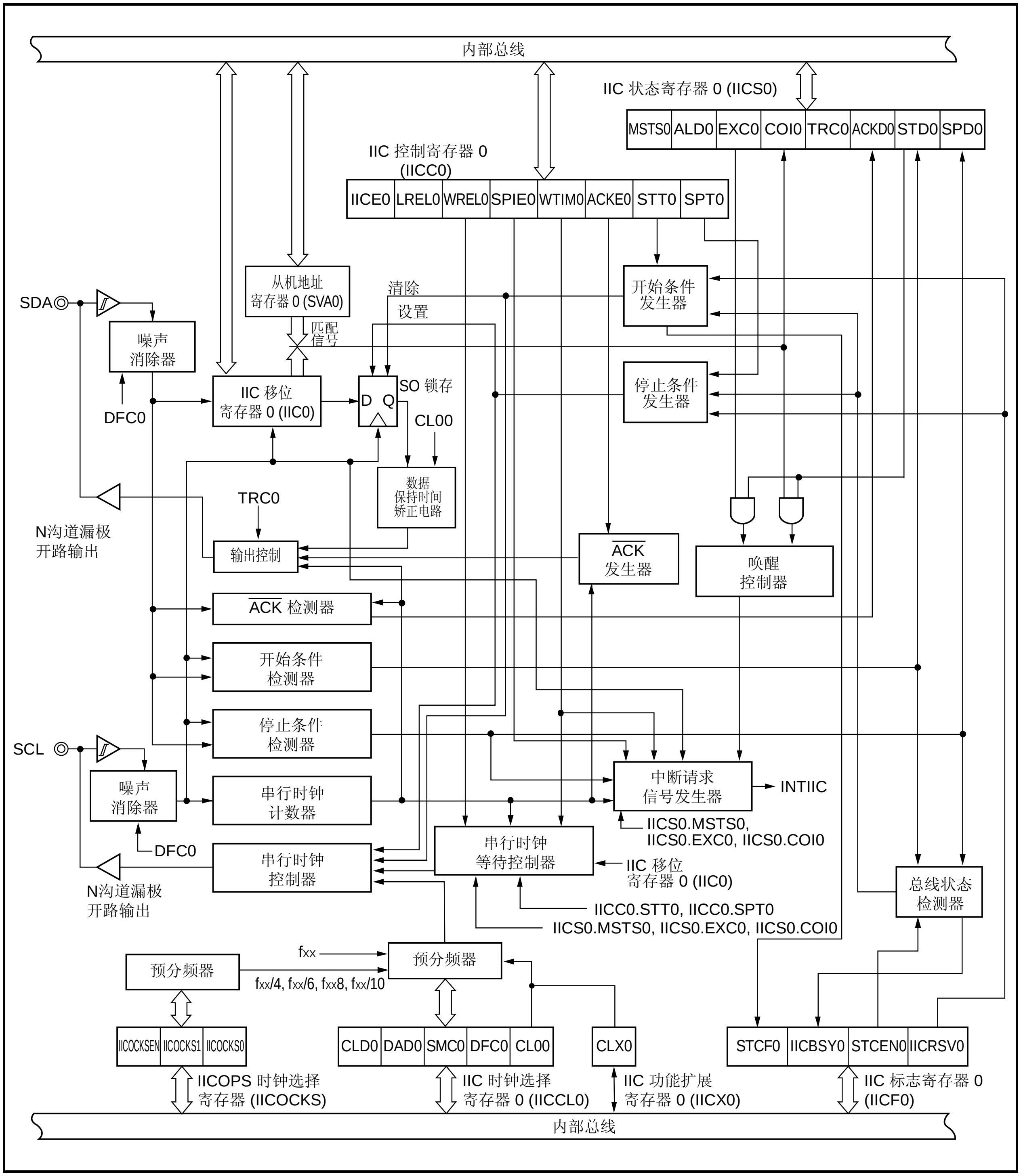
（2） I²C总线模式（支持多主设备）

该模式通过两条信号线在多个设备之间进行 8 位数据传输：串行时钟（SCL）线和串行数据总线（SDA）线。

该模式符合I²C 总线规范格式，主设备可以通过其串行数据总线，为从设备产生"开始条件"、"地址"、"传输方向规范"、"传据"和 "停止条件"等数据。从设备通过硬件自动检测接收到的状态和数据。这种功能可以使控制I²C总线的应用程序得到简化。

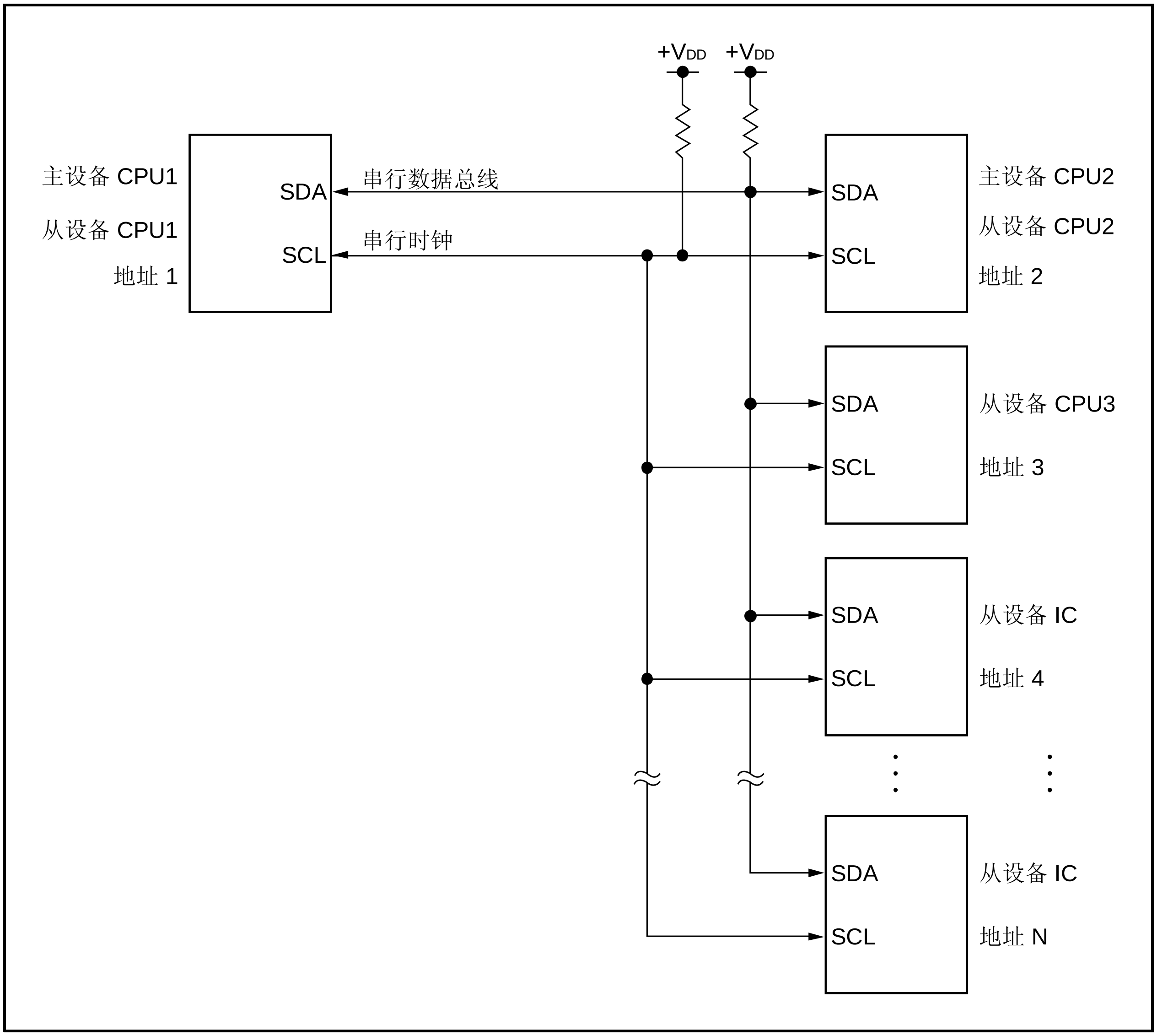
由于SCL和SDA引脚用于 N沟道漏极开路输出，因此I²C 的串行时钟线和串行数据线都需要上拉电阻。

图 17-2. I2C的框图


内部总线
IIC 状态寄存器 0 (IICS0)
MSTS0 ALD0 EXC0 COI0 TRC0 ACKD0 STD0 SPD0
IIC 控制寄存器 0 (IICC0)
IICE0 LREL0 WREL0 SPIE0 WTIM0 ACKE0 STT0 SPT0
从机地址寄存器 0 (SVA0)
清除
设置
匹配信号
SDA
噪声消除器
DFC0
IIC 移位寄存器 0 (IIC0)
D Q
SO 锁存
CL00
数据保持时间矫正电路
开始条件发生器
停止条件发生器
TRC0
N沟道漏极开路输出
输出控制
ACK 发生器
唤醒控制器
ACK 检测器
开始条件检测器
停止条件检测器
SCL
噪声消除器
串行时钟计数器
中断请求信号发生器
INTIIC
DFC0
串行时钟控制器
串行时钟等待控制器
IICS0.MSTS0, IICS0.EXC0, IICS0.COI0
IIC 移位寄存器 0 (IIC0)
N沟道漏极开路输出
IICC0.STT0, IICC0.SPT0
IICS0.MSTS0, IICS0.EXC0, IICS0.COI0
总线状态检测器
预分频器
fxx
预分频器
fxx/4, fxx/6, fxx8, fxx/10
IICOCKSEN IICOCKS1 IICOCKS0
CLD0 DAD0 SMC0 DFC0 CL00
CLX0
STCF0 IICBSY0 STCEN0 IICRSV0
IICOPS 时钟选择寄存器 (IICOCKS)
IIC 时钟选择寄存器 0 (IICCL0)
IIC 功能扩展寄存器 0 (IICX0)
IIC 标志寄存器 0 (IICF0)
内部总线

串行总线结构配置的示例如下。

图 17-3. 使用I²C 总线的串行总线配置示例



17.3 配置

I^2C包括下列硬件。

表 17-1. I^2C的配置

项目	配置
寄存器	IIC 移位寄存器 0（IIC0） 从设备地址寄存器 0 （SVA0）
控制寄存器	IIC 控制寄存器 0 （IICC0） IIC 状态寄存器 0（IICS0） IIC 标志寄存器 0（IICF0） IIC 时钟选择寄存器 0 （IICCL0） IIC 功能扩展寄存器 0 （IICX0） IIC 时钟分频选择寄存器（IICOCKS）

（1） IIC 移位寄存器 0（IIC0）

IIC0 寄存器用于将 8 位串行数据转换为 8 位并行数据，也可将 8 位并行数据转换为 8 为串行数据，发送和接收都可以使用 IIC0 寄存器。

对 IIC0 寄存器的写入和读取操作用来控制实际的发送和接收操作。

该寄存器可以按字节进行读取或写入。

系统复位后，IIC0 置为 00H。

（2） 从设备地址寄存器 0 （SVA0）

在从模式下，SVA0 寄存器用于设定本地地址。

该寄存器可以按字节进行读取或写入。

系统复位后，SVA0 置为 00H。

（3） SO 锁存器

SO 锁存器用于保持 SDA 引脚的输出电平。

（4） 唤醒控制器

该寄存器收到的地址和 SVA0 寄存器设定的地址相一致时，或接收到扩展码时，该电路就会产生中断请求信号（INTIIC）。

（5） 预分频器

用于选择使用的采样时钟。

（6） 串行时钟计数器

在发送/接收操作过程中，串行时钟计数器对输出的串行时钟和输入的串行时钟进行计数，且用于验证发送或接收到的 8 位数据。

（7） 中断请求信号发生器

该电路控制中断请求信号（INTIIC）的产生。

以下两种触发都可以产生I^2C 中断。

- 串行时钟的第 8 个或第 9 个（由 IICC0.WTIM0 位设定）时钟的下降沿。
- 检测到停止条件时（由 IICC0.SPIE0 位设定），产生中断请求。

（8） 串行时钟控制器

在主模式下，该电路从采样时钟生成时钟并经由 SCL 引脚输出。

（9） 串行时钟等待控制器

该电路控制等待时序。

（10） $\overline{\text{ACK}}$ 发生器、停止条件检测器、开始条件检测器以及 $\overline{\text{ACK}}$ 检测器

这些电路用来产生和检测各种状态。

（11） 数据保持时间修正电路

该电路根据对应的串行时钟下降沿，为数据产生保持时间。

（12） 开始条件发生器

当 IICC0.STT0 位被置位时，该电路产生开始条件。

但是，在通信保留禁止状态（IICF0.IICRSV0 位 = 1）下，如果总线未释放（IICF0.IICBSY0 位 = 1）时，则忽略开始条件请求，且 IICF0.STCF0 位设置为 1。

（13） 停止条件发生器

当 IICC0.SPT0 位被置位（1）时，产生停止条件。

（14） 总线状态检测器

通过检测开始条件和停止条件，该电路检测总线是否释放。

但是，总线状态不能在操作后立即检测到，初始状态由 IICF0.STCEN0 位设置。

17.4 寄存器

I²C由下列寄存器进行控制：

- IIC 控制寄存器 0（IICC0）
- IIC 状态寄存器 0（IICS0）
- IIC 标志寄存器 0（IICF0）
- IIC 时钟选择寄存器 0（IICCL0）
- IIC 功能扩展寄存器 0（IICX0）
- IICOPS 时钟选择寄存器（IICOCKS）

以下寄存器也可以使用：

- IIC 移位寄存器 0（IIC0）
- 从设备地址寄存器 0（SVA0）

备注 关于复用功能引脚的设置， 参见 **表 4-14 端口引脚用作复用功能的设置**。

（1） **IIC 控制寄存器 0（IICC0）**

IICC0 寄存器用于使能/停止I²C操作，设置等待时间并设置其它I²C操作。

IICC0 寄存器可以按字节或按位进行读取和写入。但是，当 IICE0 位为 0 时或在等待期间，对 SPIE0、WTIM0 和 ACKE0 位置位。当 IICE0 位的设置由“0”变为“1”时，这些位也可以同时置位。

系统复位后，该寄存器置为 00H。

（1/4）

复位后： 00H　　R/W　　地址：FFFFFD82H

	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
IICC0	IICE0	LREL0	WREL0	SPIE0	WTIM0	ACKE0	STT0	SPT0

IICE0	I²C操作允许/禁止的说明
0	操作停止。复位IICS0 寄存器[注 1]。内部操作停止。
1	使能操作。
SCL 和 SDA 信号线是高电平时，请确保将该位设置为 1。	
清零条件（IICE0 位 = 0）	置位条件（IICE0 位= 1）
• 由指令清除。 • 复位。	• 由指令设置。

LREL0[注 2]	退出通信
0	正常操作
1	从当前通信退出，并设置为待机模式。在执行之后该设置自动清除为 0。当接收到本地无关扩展码的情况下也可以使用。 SCL 和 SDA 信号线设为高阻状态。 STT0, SPT0, IICS0.MSTS0, IICS0.EXC0, IICS0.COI0, IICS0.TRC0, IICS0.ACKD0 和 IICS0.STD0 位清除为 0。
从通信中退出之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。 • 检测到停止条件后，在主模式下重新开始。 • 在开始条件以后，发生地址一致或接收到扩展码。	
清零条件（LREL0 位 = 0）	置位条件 （LREL0 位 = 1）
• 执行后自动清除。 • 复位。	• 由指令设置。

注 **1.** IICS0 寄存器、IICF0.STCF0, IICF0.IICBSY0, IICCL0.CLD0 和 IICCL0.DAD0 位重置。
2. 当 IICE0 位 = 0 时，该标志信号无效。

注意事项 **当SCL信号线为高电平且SDA信号线为低电平时，如果允许I²C操作（IICE0 位 = 1），则立即检测到开始条件。为了避免这种情况，I²C操作使能之后，立刻使用位处理指令将LREL0 位设置为 1 。**

（2/4）

WREL0[注]	等待取消的控制
0	不取消等待。
1	取消等待。等待取消之后，该设置自动清除为 0。
清零条件（WREL0 位 = 0）	置位条件（WREL0 位 = 1）
• 执行后自动清除。 • 复位。	• 由指令设置。

SPIE0[注]	当检测到停止条件时，使能/禁止中断请求的产生
0	禁止
1	使能
清零条件 （SPIE0 位 = 0）	置位条件 （SPIE0 位 = 1）
• 由指令清除。 • 复位。	• 由指令设置。

WTIM0[注]	等待和产生中断请求的控制
0	在第 8 个时钟的下降沿产生中断请求信号。 主模式： 在输出 8 个时钟以后，时钟输出置为低电平，并且设置等待状态。 从模式： 在输入 8 个时钟以后，时钟设置为低电平，并为主设备设置等待状态。
1	在第 9 个时钟信号的下降沿产生中断请求信号。 主模式： 在输出 9 个时钟以后，时钟输出设置为低电平，并且设置等待状态。 从模式： 在输入 9 个时钟以后，时钟设置为低电平，并为主设备设置等待状态。
地址传输期间，不论该位如何设置，在第 9 个时钟下降沿处发生中断。地址传输完成后该位的设置生效。主模式下，在地址传输期间的第 9 个时钟下降沿处，插入一个等待状态。对于接收到本地地址的从设备，产生 $\overline{ACK}$ 信号之后的第 9 个时钟的下降沿处插入一个等待状态。但是，当从设备收到扩展码后，在第 8 个时钟的下降沿插入等待状态。	
清零条件 （WTIM0 位 = 0）	置位条件 （WTIM0 位 = 1）
• 由指令清除。 • 复位。	• 由指令设置。

ACKE0[注]	应答控制
0	禁止应答。
1	使能应答。在第 9 个时钟期间，SDA 信号线设置为低电平。
ACKE0 位的设置对地址接收是无效的，这种情况下，地址一致时产生 $\overline{ACK}$ 信号。 但是，对于扩展码的地址接收，ACKE0 位的设置有效。	
清零条件 （ACKE0 位 = 0）	置位条件 （ACKE0 位 = 1）
• 由指令清除。 • 复位。	• 由指令设置。

注 当 IICE0 位 = 0 时，该标志信号无效。

（3/4）

<table>
<tr><th>STT0</th><th colspan="2">开始条件触发</th></tr>
<tr><td>0</td><td colspan="2">不产生开始条件。</td></tr>
<tr><td>1</td><td colspan="2">当总线释放（在 STOP 模式下）：
产生开始条件（作为主设备启动）。当 SCL 信号线为高电平时，SDA 信号线由高电平变为低电平，这样就可以产生开始条件。标准时间过后，SCL 信号线变为低电平（等待状态）。
与第三方通信时：
• 当通信保留功能使能时（IICF0.IICRSV0 位 = 0）
作为开始条件的保留标志。当设置为 1 时，在总线释放之后，自动产生开始条件。
• 当通信保留功能禁止时（IICRSV0 = 1）
IICF0.STCF0 位设置为 1，且设置（1）在 STT0 位的信息清除，不产生开始条件。
处于等待状态 （作为主设备）：
等待状态释放之后，产生重新开始条件。</td></tr>
<tr><td colspan="3">设置时序的有关注意事项：
主设备接收：　传输期间不能设置为 1，只能在 ACKE0 位已经清除为 0，且从设备已经得知最末接收时，才可以置为 1 。
主设备发送：　在 ACK 期间，可能不会正常产生开始条件。在第 9 个时钟输出后的等待期间置为 1。
• 不能和 SPT0 位同时设置为 1。
• 当 STT0 位设置为 1 时，禁止再次将 STT0 位置 1，直到设置清 0 为止。</td></tr>
<tr><td colspan="2">清零条件（STT0 位 = 0）</td><td>置位条件（STT0 位 = 1）</td></tr>
<tr><td colspan="2">• 在通信保留禁止状态，当 STT0 位设置为 1 时。
• 仲裁失败后清除。
• 主设备产生开始条件后清除。
• 当 LREL0 位 = 1 时 （从通信中退出）。
• 当 IICE0 位 = 0（操作停止）时。
• 复位</td><td>• 由指令设置。</td></tr>
</table>

备注　如果在数据设置后读取 STT0 位，其值为 0。

（4/4）

<table>
<tr><td>SPT0</td><td colspan="2">停止条件触发</td></tr>
<tr><td>0</td><td colspan="2">不产生停止条件。</td></tr>
<tr><td>1</td><td colspan="2">产生停止条件 （主设备传输的终结）。
SDA0 信号线变为低电平之后，将 SCL 信号线设置为高电平，或者一直等待直到 SCL 引脚变为高电平。标准时间过后，SDA 信号线由低电平变为高电平，产生停止条件。</td></tr>
<tr><td colspan="3">设置时序的有关注意事项
主设备接收：　传输期间不能设置为 1，
只能在 ACKE0 位已经设置为 0，且从设备已经得知最末接收后的等待期间，才可以置为 1 。
主设备发送：　在 $\overline{\text{ACK}}$ 接收期间，通常不会产生停止条件。在第 9 个时钟输出后的等待期间置为 1。
• 不能和 STT0 位同时置 1。
• 只有在主模式[注]下， SPT0 位才可以设置为 1。
• 在 WTIM0 位已经设置 0 时，如果在 8 个时钟输出后的等待期间将 SPT0 位设置为 1，注意，将在第 9 个时钟的高电平期间产生停止条件。
在 8 个时钟输出后的等待期间， WTIM0 位应该从 0 变为 1，且在第 9 个时钟输出后的等待期间应该将 SPT0 位设置为 1。
• 当 SPT0 位设置为 1 时，禁止再次将 SPT0 位置为 1，直到设置清除为 0。</td></tr>
<tr><td colspan="2">清零条件 （SPT0 位 = 0）</td><td>置位条件 （SPT0 位 = 1）</td></tr>
<tr><td colspan="2">• 仲裁失败后清除。
• 检测到停止条件后，自动清除。
• 当 LREL0 位 = 1 时 （从通信中退出）。
• IICE0 位 = 0（操作停止）时。
• 复位后。</td><td>• 由指令设置。</td></tr>
</table>

注　只有在主设备模式下，才可以将 SPT0 位设置为 1。但是，改变操作使能状态之后检测到第一个停止条件前，SPT0 位必须设置为 1 并产生停止条件。详情参见 **17.15 注意事项。**

注意事项　当 IICS0.TRC0 位设置为 1 时，在第 9 个时钟期间 WREL0 位置 1 且等待状态被取消，之后 TRC0 位清除为 0 且 SDA 信号线置为高阻状态。

备注　如果在数据设置以后读取 SPT0 位，其值为 0。

（2） IIC 状态寄存器 0（IICS0）

IICS0 寄存器表示I²C总线的状态。

IICS0 寄存器只能按字节或按位进行读取。但是，只有当 IICC0.STT0 位为 1 时或在等待期间，才能读取 IICS0 寄存器。

系统复位后，该寄存器置为 00H。

（1/3）

复位后： 00H　　R　　地址：FFFFFD86H

	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
IICS0	MSTS0	ALD0	EXC0	COI0	TRC0	ACKD0	STD0	SPD0

MSTS0	主设备状态
0	从设备状态或通信待机状态
1	主设备通信状态

清零条件 （MSTS0 位 = 0）	置位条件 （MSTS0 位 = 1）
• 当检测到停止条件时。 • 当 ALD0 位= 1 时 （仲裁失败）。 • 由 IICC0.LREL0 位 = 1 清除 （从通信中退出）。 • 当 IICC0.IICE0 位 由 1 变为 0 （工作停止）时。 • 复位。	• 当产生开始条件时。

ALD0	仲裁失败的检测
0	该状态意味着不存在总线仲裁，或仲裁结果为“胜利”。
1	该状态表示仲裁结果为“失败”。MSTS0 位清除为 0。

清零条件 （ALD0 位 = 0）	置位条件 （ALD0 位= 1）
• IICS0 寄存器读取[注]后， 自动清除。 • 当 IICE0 位由 1 变为 0 （工作停止）时。 • 复位后。	• 当总线仲裁结果为“失败”。

注　对 IICS0 寄存器的其它位进行位操作时，该位也清除。

（2/3）

EXC0	扩展码接收的检测
0	未接收到扩展码。
1	接收到扩展码。

清零条件 （EXC0 位 = 0）	置位条件 （EXC0 位= 1）
• 当检测到开始条件时。 • 当检测到停止条件时。 • 通过 LREL0 位 = 1 清除 （从通信中退出）。 • 当 IICE0 位由 1 变为 0 时（操作停止）。 • 复位。	• 当接收地址数据的高 4 位是“0000” 或 “1111”时（在第 8 个时钟的上升沿置位）。

COI0	一致地址的检测
0	地址不一致
1	地址一致

清零条件 （COI0 位 = 0）	置位条件 （COI0 位= 1）
• 当检测到开始条件时。 • 当检测到停止条件时。 • 通过 LREL0 位 = 1 清除 （从通信中退出）。 • 当 IICE0 位由 1 变为 0 时。 • 复位。	• 当接收到的地址和本地地址（SVA0 寄存器）一致时（在第 8 个时钟的上升沿置位）。

TRC0	发送/接收状态的检测
0	接收状态（除发送状态外）。SDA 信号线置为高阻态。
1	发送状态。允许 SO 锁存器内的数值从 SDA 信号线输出（在第一个字节的第 9 个时钟上升沿开始生效）。

清零条件 （TRC0 位 = 0）	置位条件 （TRC0 位= 1）
• 当检测到停止条件时。 • 通过 LREL0 位 = 1 清除 （从通信中退出）。 • 当 IICE0 位由 1 变为 0 （工作停止）时。 • 通过IICC0.WREL0 位 = 1[注]清除（等待释放）。 • 当 ALD0 位从 0 变为 1 （仲裁失败）时。 • 复位。 主设备： • 当第一个字节的 LSB（传输方向指示位） 输出为“1”时。 从设备： • 当检测到开始条件时。 当不用于通信时。	主设备： • 当产生开始条件时。 • 当第一个字节的 LSB（传输方向指示位） 输出为“0”时。 从设备： • 当第一个字节的 LSB（传输方向指示位） 输入为“1”时。

注 当 IICC0.WREL0 位设置为 1，TRC0 清除为 0，而且 SDA 信号线变为高阻态，且在第九个时钟通过 IICS0.TRC0 位 = 1 释放等待状态。

（3/3）

ACKD0	$\overline{ACK}$ 的检测
0	未检测到 $\overline{ACK}$
1	已检测到 $\overline{ACK}$

清零条件 （ACKD0 位= 0）	置位条件 （ACKD0 位 = 1）
• 当检测到停止条件时。 • 在下一字节第一个时钟的上升沿。 • 通过 LREL0 位 = 1 清除 （从通信中退出）。 • 当 IICE0 位由 1 变为 0 （工作停止）时。 • 复位。	•在 SCL 引脚的第 9 个时钟的上升沿 SDA 引脚设置为低电平后。

STD0	开始条件的检测
0	未检测到开始条件。
1	检测到开始条件。这表示在地址传输时期有效时段内。

清零条件 （STD0 位 = 0）	置位条件（STD0 位 = 1）
• 当检测到停止条件时。 • 在地址传输后下一个字节第一个时钟的上升沿。 • 通过 LREL0 位 = 1 清除（从通信中退出）。 • 当 IICE0 位 由 1 变为 0 （工作停止）时。 • 复位。	• 当检测到开始条件时。

SPD0	停止条件的检测
0	未检测到停止条件。
1	检测到停止条件。主设备的通信终结，并且释放总线。

清零条件 （SPD0 位 = 0）	置位条件 （SPD0 位 = 1）
• 在位置该位之后的地址传输第一个时钟上升沿处，以及检测到开始条件时。 • 当 IICE0 位由 1 变为 0 （工作停止）时。 • 复位。	• 当检测到停止条件时。

（3） IIC 标志寄存器 0（IICF0）

IICF0 寄存器用于设置I2C操作模式，并指示I2C总线的状态。

该寄存器可以按字节或按位进行读取和写入。但是，STCF0 和 IICBSY0 位只读。

IICRSV0 可以用于使能/禁止通信保留功能（参见 **17.14 通信保留**）。

IICBSY0 位的初始值通过 STCEN0 位设置（参见 **17.15 注意事项**）。

IICRSV0 和 STCEN0 位只能在I2C操作禁止（IICC0.IICE0 位 = 0）时写入。当操作使能时，可以读取IICF0 寄存器。

系统复位后，该寄存器置为 00H。

复位后： 00H　　R/W注　　地址：FFFFFD8AH

	<7>	<6>	5	4	3	2	<1>	<0>
IICF0	STCF0	IICBSY0	0	0	0	0	STCEN0	IICRSV0

STCF0	IICC0.STT0 清除标志
0	产生开始条件
1	开始条件的产生不成功，清除 STT0 标志

清零条件 （STCF0 位 = 0）	置位条件（STCF0 位= 1）
(通过 IICC0.STT0 位 = 1 清除。 • 当 IICE0 位 =1 → 0 时（操作停止） • 复位。	• 当通信保留（IICRSV0 位 = 1）禁止期间，开始条件的产生不成功且 STT0 标志清除为 0 时。

IICBSY0	I²C 总线状态
0	总线释放状态（当 STCEN0 位 = 1 时的初始通信状态）
1	总线通信状态（当 STCEN0 位 = 0 时的初始通信状态）

清零条件 （IICBSY0 位 = 0）	置位条件 （IICBSY0 位 = 1）
• 当检测到停止条件时。 • 当 IICE0 位 =1 → 0 时（操作停止） • 复位	• 当检测到开始条件时。 • 当 STCEN0 位 = 0 时，设置 IICE0 位

STCEN0	初始开始使能触发
0	操作使能（IICE0 位 = 1）之后，在检测到停止条件时使能开始条件的产生。
1	操作使能（IICE0 位 = 1）之后，无需检测到停止条件，就使能开始条件的产生。

清零条件 （STCEN0 位 = 0）	置位条件 （STCEN0 位 = 1）
• 当检测到开始条件。 • 复位	• 由指令设置。

IICRSV0	通信保留功能禁止位
0	使能通信保留
1	禁止通信保留

清零条件 （IICRSV0 位 = 0）	置位条件 （IICRSV0 位 = 1）
• 由指令清除。 • 复位	• 由指令设置。

注　　第 6 位和第 7 位为只读位。

注意事项
1. **仅当操作停止时（IICE0 位 = 0），才可以写入 STCEN0 位。**
2. **当 STCEN0 位 = 1 时，不管总线的实际状态如何，认为处于总线释放状态（IICF0.IICBSY0 位 = 0）。当产生第一个开始条件（STT0 位 = 1）时，必需确认没有第三方通信在进行中，为了防止这样的通信被破坏。**
3. **仅当操作停止时（IICE0 位 = 0），才可以写入 IICRSV0 位。**

（4） IIC 时钟选择寄存器 0（IICCL0）

IICCL0 寄存器用于设置I²C总线的传输时钟。

该寄存器可以按字节或按位进行读取或写入。但是，CLD0 和 DAD0 位是只读位。SMC0 和CL00 位和IICX0.CLX0、IICOCKS.IICOCKS1 及IICOCKS.IICOCKS0 位组合使用进行设置（参见 **17.4 （7） I²C传输时钟设置方法**）。

当 IICC0.IICE0 位 = 0 时，设置 IICCL0 寄存器。

系统复位后，该寄存器置为 00H。

复位后： 00H R/W注 地址： FFFFFD84H

	7	6	<5>	<4>	3	2	1	0
IICCL0	0	0	CLD0	DAD0	SMC0	DFC0	0	CL00

CLD0	SCL 引脚电平的检测 （仅当 IICC0.IICE0 位 = 1 时有效）
0	SCL 引脚检测为低电平。
1	SCL 引脚检测为高电平。

清零条件 （CLD0 位 = 0）	置位条件 （CLD0 位 = 1）
• 当 SCL 引脚为低电平时。 • 当 IICE0 位 =1 → 0 时（操作停止） • 复位	• 当 SCL 引脚为高电平时。

DAD0	SDA 引脚电平的检测 （仅当 IICE0 位 = 1 时有效）
0	SDA 引脚检测为低电平。
1	SDA 引脚检测为高电平。

清零条件 （DAD0 位 = 0）	置位条件 （DAD0 位 = 1）
• 当 SDA 引脚为低电平。 • 当 IICE0 位 =1 → 0 时（操作停止） • 复位	• 当 SDA 引脚为高电平时。

SMC0	工作模式切换
0	工作于标准模式。
1	工作于高速模式。

DFC0	数字滤波器操作控制
0	数字滤波器关闭。
1	数字滤波器开启。

数字滤波器只能在高速模式下使用。
在高速模式下，不论 DFC0 位如何设置（开启/关闭），传输时钟都不会改变。
数字滤波器用于消除高速模式下的噪声。

CL00	通信时钟选择	
	正常模式	高速模式
0	Fxx/44	Fxx/24
1	Fxx/86	Fxx/24

注 第 4 位和第 5 位是只读位。

备注 Fxx：选择时钟

（5） IIC 功能扩展寄存器 0（IICX0）

该寄存器用于设置I²C的功能扩展（仅在高速模式下有效）。

该寄存器可以按字节或按位进行读取或写入。CLX0 位与IICCL0.SMC0, IICCL0.CL00, IICOCKS.IICOCKS1 和 IICOCKS.IICOCKS0 位组合设置（参见 **17.4 （7） I²C传输时钟设置方法**）。

当 IICC0.IICE0 位 = 0 时，设置 IICX0 寄存器。

系统复位后，该寄存器置为 00H。

复位后： 00H　　R/W　　地址： FFFFFD85H

	7	6	5	4	3	2	1	<0>
IICX0	0	0	0	0	0	0	0	CLX0

CLX0	时钟选择扩展位
0	按照 IICCL0.CL00 位设置的传输速率通信
1	在高速模式下，按照 IICCL0.CL00 位设置的传输速率的两倍速率通信

(6) IICOPS 时钟选择寄存器 (IICOCKS)

该寄存器用于控制I²C的分频时钟。

该寄存器可以按字节或按位进行读取或写入。IICOCKS1 和IICOCKS0 位与IICCL0.SMC0, IICCL0.CL00 和 IICX0.CLX0 位组合设置 (参见 **17.4 （7） I²C传输时钟设置方法**)。

系统复位后，该寄存器置为 00H。

复位后： 00H　　R/W　　地址： FFFFFD90H

	7	6	5	4	3	2	1	<0>
IICOCKS	0	0	0	IICOCKSEN	0	0	IICOCKS1	IICOCKS0

IICOCKSEN	I²C的分频时钟操作的规范
0	I²C的分频时钟操作停止
1	I²C的分频时钟操作使能

IICOCKS1	IICOCKS0	I²C的分频时钟选择
0	0	$f_{xx}/4$
0	1	$f_{xx}/6$
1	0	$f_{xx}/8$
1	1	$f_{xx}/10$

（7） I²C传输时钟设置方法

使用下列表达式计算I²C传输时钟频率（f_{SCL}）。

$f_{SCL} = 1/（m \times T + t_R + t_F）$

m = 96, 120, 144, 192, 240, 344, 352, 440, 516, 688, 860（参见表 **17-2 选择时钟设置**）。

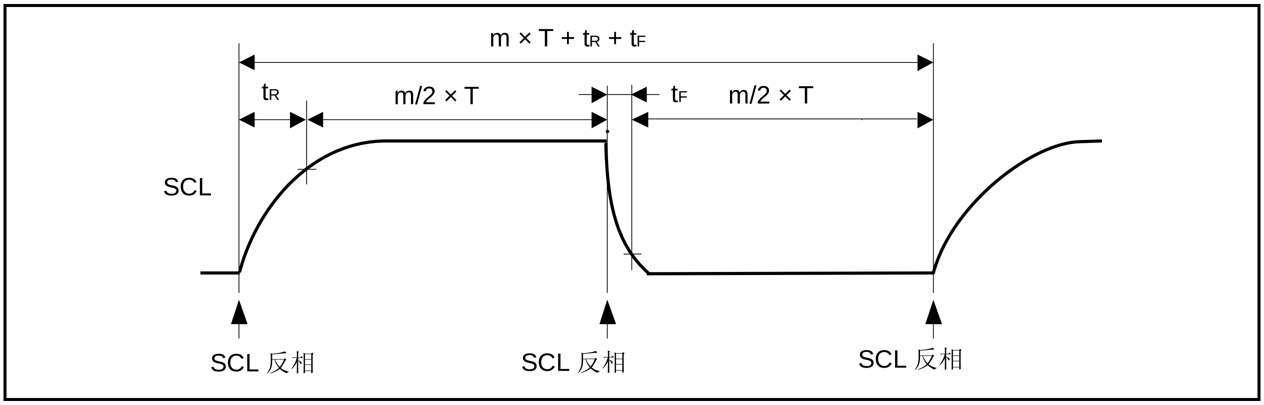
T： $1/f_{XX}$

t_R： SCL 上升时间

t_F： SCL 下降时间

例如，当f_{XX} = 64 MHz， m = 192， t_R = 200 ns 且 t_F = 50 ns时，I²C传输时钟频率（f_{SCL}）使用下面表达式计算。

$f_{SCL} = 1/（192 \times 15.6 \text{ ns} + 200 \text{ ns} + 50 \text{ ns}） \cong 308 \text{ kHz}$



使用 IICCL0.SMC0, IICCL0.CL00, IICX0.CLX0, IICOCKS.IICOCKS1 和 IICOCKS.IICOCKS0 位的组合设置来选择时钟。

表 17-2. 选择时钟设置

IICX0	IICCL0		选择时钟	传输时钟 (f_{XX}/m)	可设置的内部系统时钟频率 （f_{XX}） 范围	操作模式
第 0 位	第 3 位	第 0 位				
CLX0	SMC0	CL00				
0	0	0	f_{XX}/8 (当 IICOCKS = 12H 时)	f_{XX}/352	32.00 MHz 至 33.52 MHz	正常模式 (SMC0 位 = 0)
			f_{XX}/10 (当 IICOCKS = 13H 时)	f_{XX}/440	32.00 MHz 至 41.90 MHz	
0	0	1	f_{XX}/4 (当 IICOCKS = 10H 时)	f_{XX}/344	32.00 MHz 至 33.52 MHz	
			f_{XX}/6 (当 IICOCKS = 11H 时)	f_{XX}/516	32.00 MHz 至 50.28 MHz	
			f_{XX}/8 (当 IICOCKS = 12H 时)	f_{XX}/688	33.52 MHz 至 64.00 MHz	
			f_{XX}/10 (当 IICOCKS = 13H 时)	f_{XX}/860	41.90 MHz 至 64.00 MHz	
0	1	x	f_{XX}/4 (当 IICOCKS = 10H 时)	f_{XX}/96	32.00 MHz 至 33.52 MHz	高速模式 (SMC0 位= 1)
			f_{XX}/6 (当 IICOCKS = 11H 时)	f_{XX}/144	32.00 MHz 至 50.28 MHz	
			f_{XX}/8 (当 IICOCKS = 12H 时)	f_{XX}/192	32.00 MHz 至 64.00 MHz	
			f_{XX}/10 (当 IICOCKS = 13H 时)	f_{XX}/240	40.00 MHz 至 64.00 MHz	
1	0	x	禁止设置			
1	1	x	f_{XX}/8 (当 IICOCKS = 12H 时)	f_{XX}/96	32.00 MHz 至 33.52 MHz	高速模式 (SMC0 位 = 1)
			f_{XX}/10 (当 IICOCKS = 13H 时)	f_{XX}/120	40.00 MHz 至 41.90 MHz	

备注 x：无需理会

（8）　IIC 移位寄存器 0（IIC0）

IIC0 移位寄存器用于与串行时钟同步进行串行发送/接收（移位操作）。

IIC0 移位寄存器可以按字节读取或写入，但是，在数据传输期间，不应该向 IIC0 移位寄存器写入数据。

只有在等待期间，可以访问（读/写）IIC0 移位寄存器。在等待期间以外的其它通信状态时，禁止访问该寄存器。

但是，对于主设备，只有在传输触发位（IICC0.STT0 位）置为 1 后， IIC0 移位寄存器才可以写入一次。

在等待期间写入 IIC0 寄存器，取消等待且启动传输。

系统复位后，该寄存器置为 00H。

复位后： 00H　　R/W　　地址： FFFFFD80H

	7	6	5	4	3	2	1	0
IIC0								

（9）　从设备地址寄存器 0（SVA0）

SVA0 寄存器保持I^2C 总线的从地址。

但是，当 IICS0.STD0 位 = 1（开始条件检测）时，禁止重写该寄存器。

系统复位后，该寄存器置为 00H。

复位后： 00H　　R/W　　地址： FFFFFD83H

	7	6	5	4	3	2	1	0
SVA0								0

17.5 功能

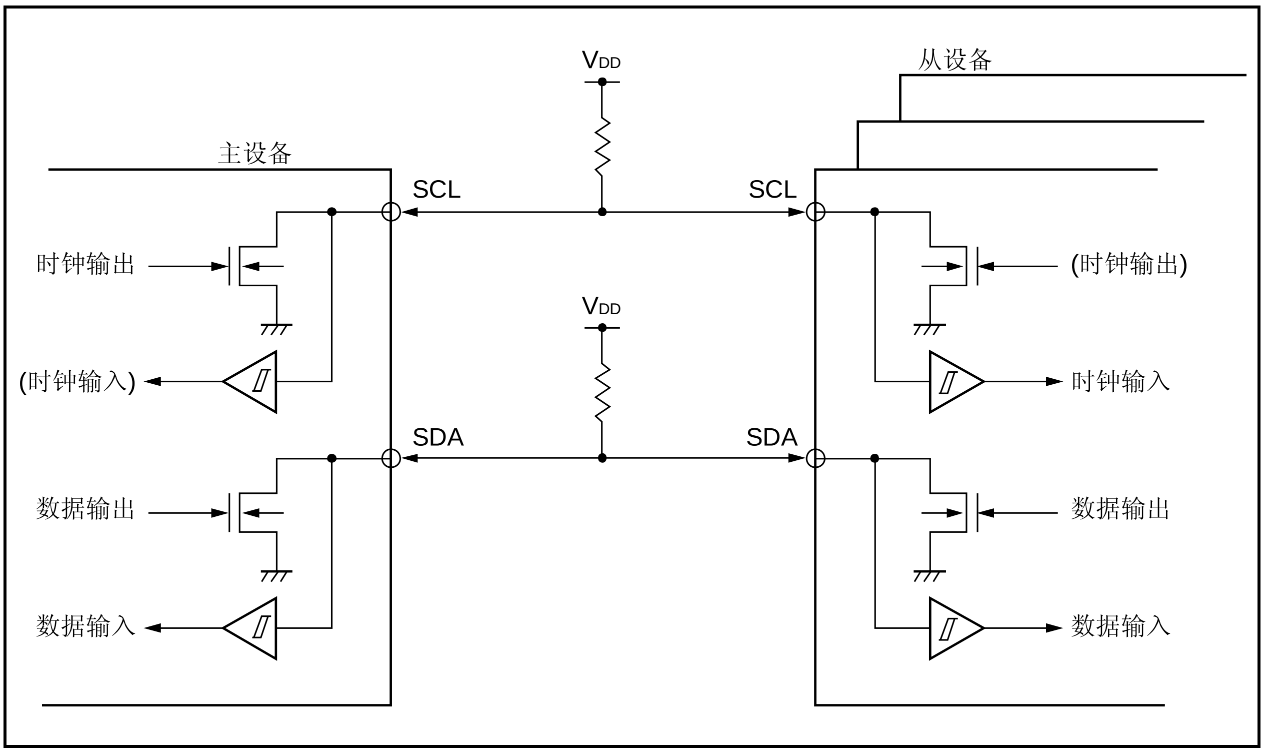
17.5.1 引脚配置

串行时钟引脚（SCL） 和串行数据总线引脚 （SDA）配置如下。

SCL .... 该引脚用于串行时钟输入和输出。
对于主设备和从设备，该引脚都是 N 沟道漏极开路输出，输入为施密特输入。
SDA .... 该引脚用于串行数据输入和输出。
对于主设备和从设备，该引脚都是 N 沟道漏极开路输出，输入为施密特输入。

由于串行时钟信号线和串行数据总线输出都是 N 沟道漏极开路输出，所以都需要外接上拉电阻。

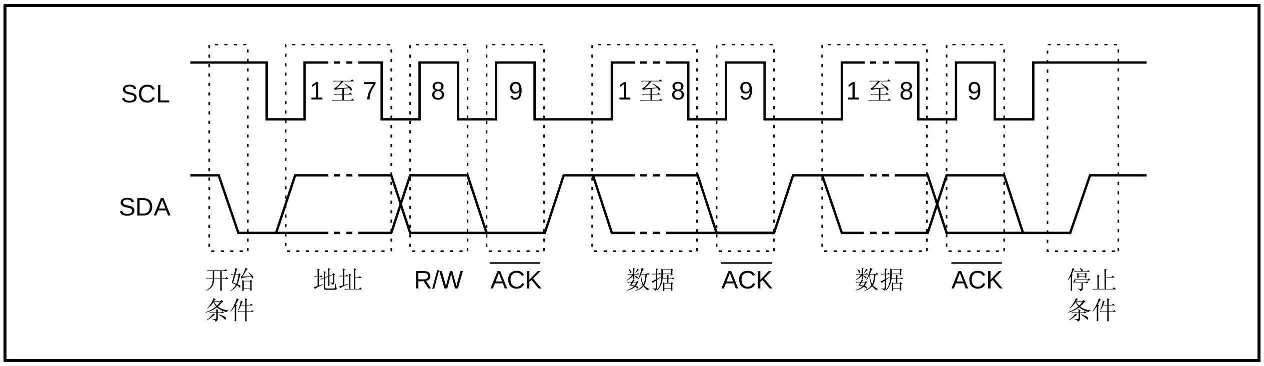
图 17-4. 引脚配置图



17.6 I^2C总线的定义和控制方法

以下章节描述I^2C 总线的串行数据通信格式和I^2C 总线产生的状态。关于I^2C 总线的串行数据总线产生的“开始条件”、“地址”、“传输方向”、“数据”以及“停止条件”的传输时序，如下所示。

图 17-5. I^2C 总线串行数据传输时序



主设备产生开始条件、从设备地址和停止条件。

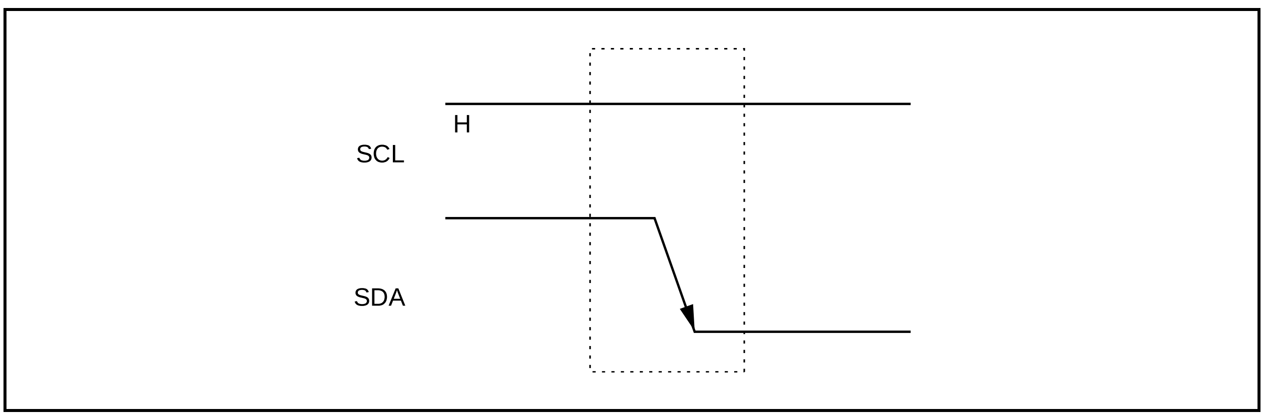
$\overline{\text{ACK}}$ 信号既可由主设备产生，也可由从设备产生。（通常情况下，它由接收 8 位数据的设备产生）。

串行时钟（SCL）由主设备连续输出。但是，在从设备中 SCL 引脚的低电平时段可以延长，可以插入一个等待状态。

17.6.1 开始条件

当 SCL 引脚为高电平而且 SDA 引脚由高电平变为低电平时，形成开始条件。当主设备启动一个至从设备的串行传输时，产生 SCL 和 SDA 引脚的开始条件。当设备用作从设备时，可以检测开始条件。

图 17-6. 开始条件



当检测到停止条件（IICS0.SPD0 位 = 1）之后，当 IICC0.STT0 位置位（1）时，产生开始条件。当检测到开始条件时，IICS0.STD0 位置位（1）。

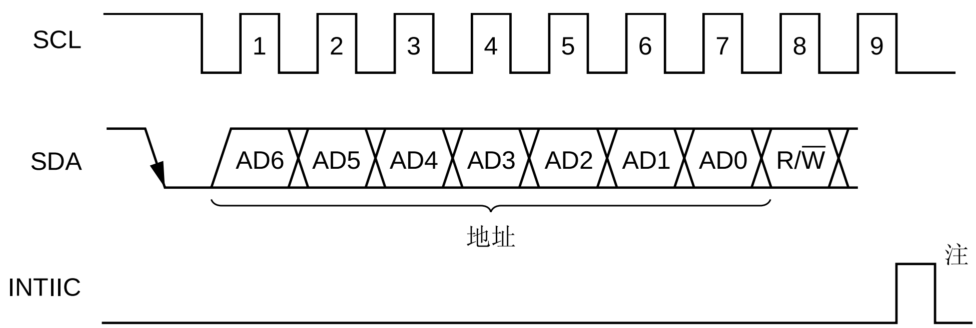
17.6.2 地址

紧随开始条件之后的 7 位数据定义为地址。

地址是 7 位数据段，输出地址用来选择通过总线连接到主设备的其中一个从设备。所以，每一个接入总线的从设备必须具有一个独一无二的地址。

从设备包含检测开始条件的硬件，并检查 7 位地址数据是否和 SVA0 的值一致。如果地址数据和 SVA0 寄存器内储存的数据值相一致，则选择该从设备，并且和主设备进行通信，直到主设备产生开始条件或结束条件。

图 17-7. 地址



注 如果在从设备操作期间，接收到本地地址或扩展码，就会产生中断请求信号 （INTIIC）。

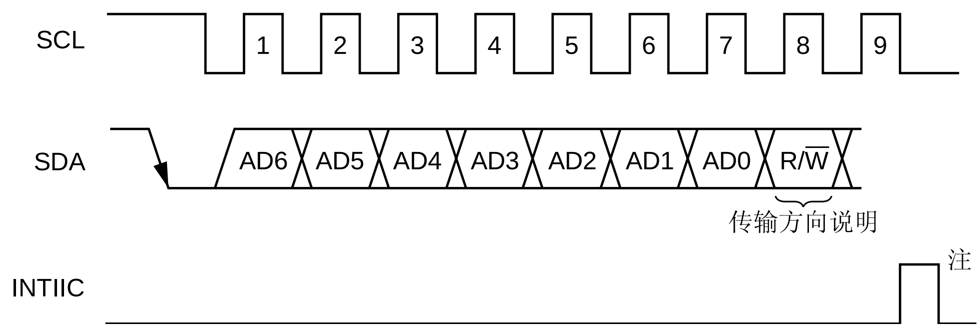
从设备地址和第 8 位一起写入 IIC0 寄存器，然后输出。接收到的地址数据写入 IIC0 寄存器。第 8 位用来指示传输方向，在 **17.6.3 指定传输方向**中描述。

从设备地址分配至 IIC0 寄存器的高 7 位 。

17.6.3 传输方向指示

除了 7 位地址数据之外，主设备发送 1 位数据指示传输方向。当该传输方向指示位为 0 时，表示主设备向从设备传输数据。当该传输方向指示位为 1 时，表示主设备由从设备接收数据。

图 17-8. 传输方向指示



注　如果在从设备操作期间，接收到本地地址或扩展码，就会产生中断请求信号（INTIIC）。

17.6.4 $\overline{\text{ACK}}$

$\overline{\text{ACK}}$ 用来确认发送设备和接收设备的串行数据传输状态。

每接收到的 8 位数据，接收设备返回一次 $\overline{\text{ACK}}$ 信号。

通常情况下，发送设备发送 8 位数据后，接收到一个 $\overline{\text{ACK}}$ 信号。当 $\overline{\text{ACK}}$ 信号由接收端返回后，接收判定为正常并且继续进行操作。使用 IICS0.ACKD0 位对 $\overline{\text{ACK}}$ 信号的检测进行确认。

当主设备是接收设备时，收到最后一个数据之后，不返回 $\overline{\text{ACK}}$ 信号且产生停止条件。当从设备为接收设备且不返回 $\overline{\text{ACK}}$ 信号时，主设备产生停止条件或重新开始条件，然后停止当前的通信传输。返回 ACK 失败的情况可由以下因素导致：

（a） 接收没有正常执行。
（b） 接收到最末数据。
（c） 指定地址上的接收设备（从设备）不存在。

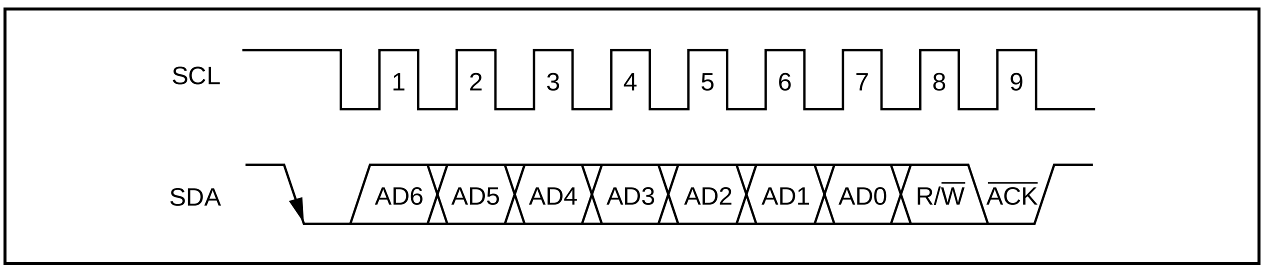
在第 9 个时钟期间，当接收设备将 SDA 信号线设置为低电平时，产生 $\overline{\text{ACK}}$ 信号（接收正常）。

当 IICC0.ACKE0 位设置为 1 时，使能自动产生 $\overline{\text{ACK}}$。发送 7 位地址之后的第 8 位会导致 IICS0.TRC0 位置位。通常情况下，设置 ACKE0 位 为 1 作为接收的应答（TRC0 位 = 0）。

当从设备进行接收（TRC0 位 = 0）时，如果从设备不能接收数据或不再需要接收更多数据，将 ACKE0 位清除为 0，提示主设备，不能再接收更多数据。

与上述相似，当主设备进行接收（TRC0 位 = 0）时，并且不再需要后续数据，将 ACKE0 位清除为 0，来防止 $\overline{\text{ACK}}$ 信号的产生。这样可以通知从设备 （发送设备）数据发送结束（发送停止）。

图 17-9. $\overline{\text{ACK}}$



接收到本地地址后，不管 ACKE0 位的设置值如何，都会自动产生 $\overline{\text{ACK}}$ 信号。如果收到的地址不是本地地址（NACK），不产生 $\overline{\text{ACK}}$ 信号。

接收到扩展码时，在产生 $\overline{\text{ACK}}$ 信号之前，将 ACKE0 位设置为 1。

数据接收期间 $\overline{\text{ACK}}$ 的产生方式取决于等待时序的设置，描述如下：

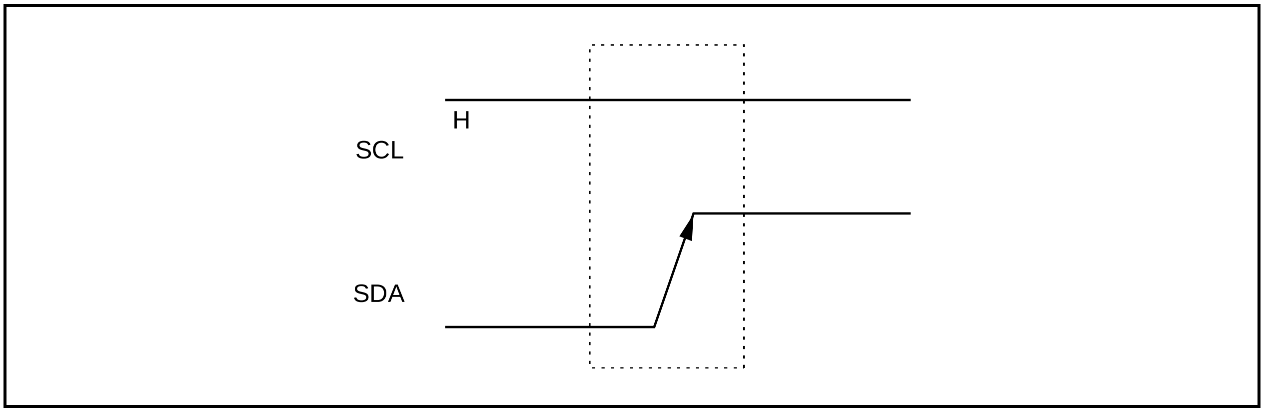
- 当选择 8 时钟等待时（IICC0.WTIM0 位 = 0）：
  如果等待状态取消前将 ACKE0 位设置为 1，$\overline{\text{ACK}}$ 信号在 SCL 引脚第 8 个时钟下降沿产生。
- 当选择 9 时钟等待时（IICC0.WTIM0 位 = 1）：
  如果 ACKE0 位预先设置为 1，产生 $\overline{\text{ACK}}$ 信号。

17.6.5 停止条件

当 SCL 引脚为高电平时，SDA 引脚由低电平变为高电平，即产生停止条件。

当主设备发向从设备的串行传输完成之后，产生停止条件。当设备用作从设备时，可以检测停止条件。

图 **17-10.** 停止条件



当 IICC0.SPT0 位设置为 1 时，发生停止条件。当检测到停止条件时，IICS0.SPD0 位置为 1，且当 IICC0.SPIE0 位置为 1 时，产生中断请求信号（INTIIC）。

17.6.6 等待状态

等待状态用来通知通信伙伴，设备（主设备或从设备）已经准备就绪可以进行收发数据（即：处于等待状态）。

将 SCL 引脚设为低电平，通知通信伙伴本机已经处于等待状态。当主设备和从设备的等待状态都取消时，可以开始下一次传输。

图 17-11. 等待状态 （1/2）

（a） 当主设备为 9 时钟等待而从设备为 8 时钟等待时
（主设备：发送，从设备：接收，而且 IICC0.ACKE0 位 = 1）

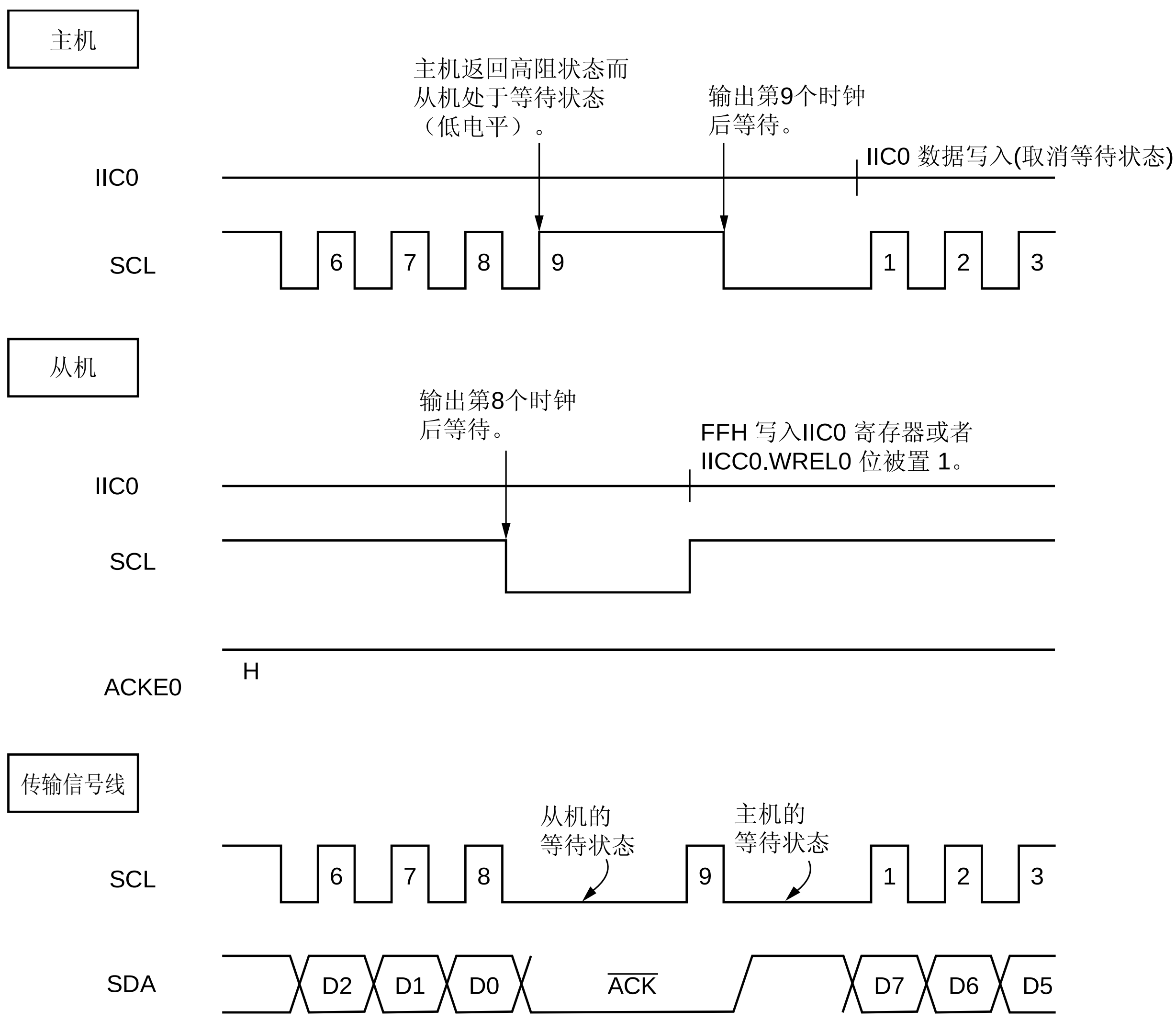
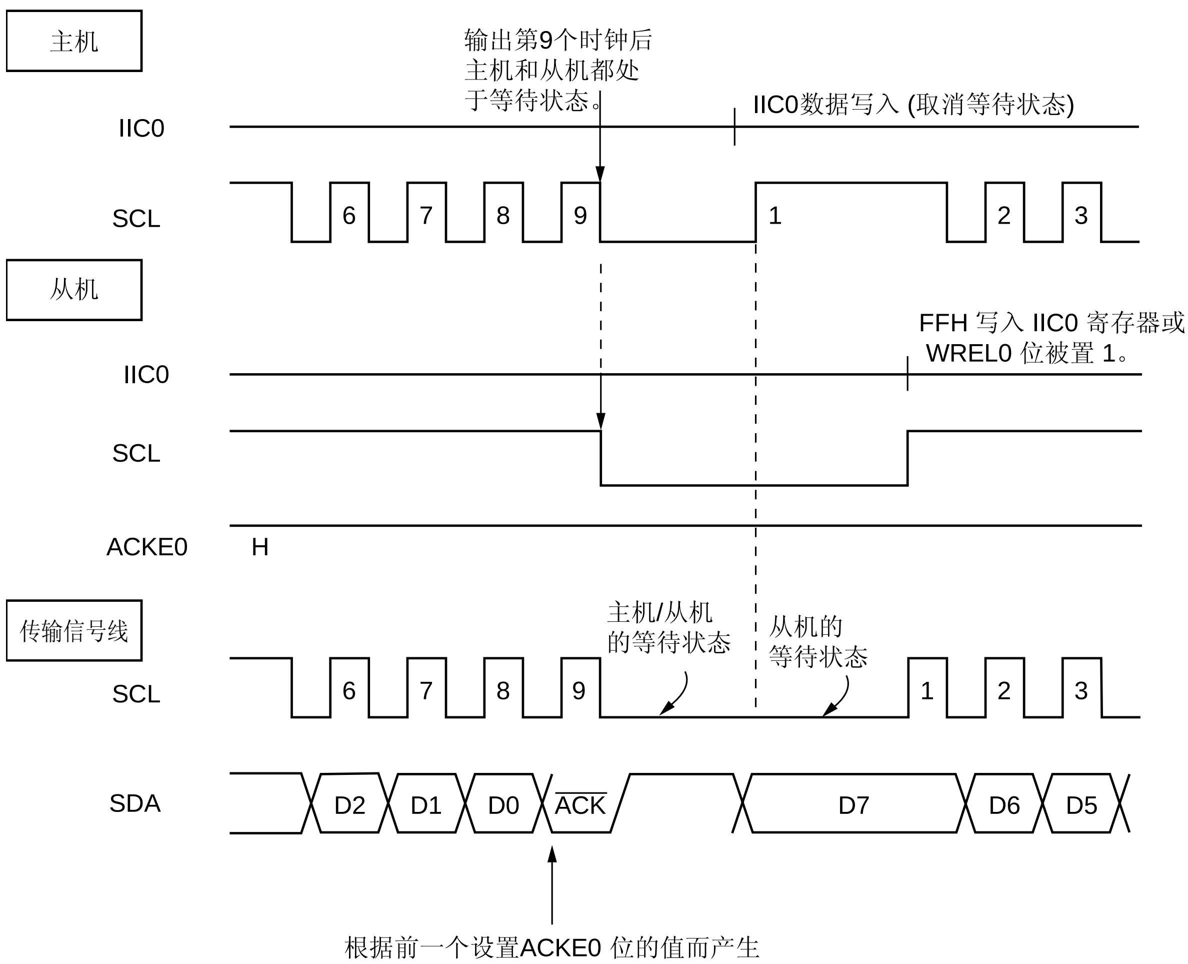


图 **17-11.** 等待状态 （**2/2**）

（**b**） 当主设备和从设备都为 **9** 时钟等待
（主设备：发送，从设备：接收，**ACKE0 = 1**）



在开始条件产生之后，自动产生一个等待状态。并且根据 IICC0.WTIM0 位的设置，会自动产生等待状态。

通常情况下，当 IICC0.WREL0 位置为 1，或当 IIC0 寄存器写入 FFH 时，取消等待状态，发送端向 IIC0 寄存器写入数据，清除等待状态。

主设备还可以通过以下方式取消等待状态。

- 将 IICC0.STT0 位置为 1。
- 将 IICC0.SPT0 位置为 1。

17.6.7 等待状态取消方法

在使用I²C的情况下，一般使用以下方式取消等待状态。

- 向 IIC0 寄存器写入数据。
- 将 IICC0.WREL0 位设置为 1 （取消等待状态）。
- 将 IICC0.STT0 位设置为 1 （产生开始条件）[注]
- 将 IICC0.SPT0 位设置为 1 （产生停止条件）[注]

注　　仅限主设备

如果执行任何一种等待取消动作，I²C将取消等待状态，并重新开始通信。

当取消等待状态并发送数据（包括地址）时，向 IIC0 寄存器写入数据。

为了在取消等待状态之后接收数据，或结束数据发送，将 WREL0 位置为 1。

为了在取消等待状态之后产生重新开始条件，将 STT0 位设置为 1。

为了在取消等待状态之后产生停止条件，将 SPT0 位设置为 1。

对每个等待状态只可执行一次取消操作。

例如，如果将 WREL0 位设置为 1，取消等待状态，然后将数据写入 IIC0 寄存器，SDA 信号线的时序变化和 IIC0 寄存器的写入时序可能发生冲突，可能导致输出到 SDA 信号线上的数据不正确。

即使在其它操作情况下，如果通信中途停止，将 IICC0.IICE0 位清除为 0 也会停止通信，使能等待状态的取消。

如果I²C 总线因噪声等原因被死锁，将IICC0.LREL0 位设置为 1，会退出通信操作，使能等待状态的取消。

17.7 I2C中断请求信号（INTIIC）

以下是 INTIIC 中断请求信号产生时序处和 INTIIC 信号时序处的 IICS0 寄存器的值。

备注	ST：	开始条件
	AD6 至 AD0：	地址
	R/$\overline{\text{W}}$：	传输方向规范
	$\overline{\text{ACK}}$：	应答
	D7 至 D0：	数据
	SP：	停止条件

17.7.1 主设备操作

（1） 开始 ~ 地址 ~ 数据 ~ 数据 ~ 停止（正常发送/接收）

<1> 当 IICC0.WTIM0 位 = 0 时

IICC0.SPT0 位 = 1
↓

ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	SP

▲1 ▲2 ▲3 ▲4 Δ5

▲1： IICS0 寄存器 = 1000X110B

▲2： IICS0 寄存器 = 1000X000B

▲3： IICS0 寄存器 = 1000X000B （WTIM0 位 = 1[注]）

▲4： IICS0 寄存器 = 1000XX00B

Δ 5： IICS0 寄存器 = 00000001B

注 要产生停止条件，将 WTIM0 位设置为 1，并改变中断请求信号 (INTIIC)的产生时序。

备注 ▲：总是产生
Δ： 仅当 SPIE0 位 = 1 时产生
X： 无需理会

<2> 当 WTIM0 位 = 1 时

SPT0 位 = 1
↓

ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	SP

▲1 ▲2 ▲3 Δ4

▲1： IICS0 寄存器 = 1000X110B

▲2： IICS0 寄存器 = 1000X100B

▲3： IICS0 寄存器 = 1000XX00B

Δ 4： IICS0 寄存器 = 00000001B

备注 ▲：总是产生
Δ： 仅当 SPIE0 位 = 1 时产生
X： 无需理会

（2）　开始 ~ 地址 ~ 数据 ~ 开始 ~ 地址 ~ 数据 ~ 停止（重新开始）

<1> 当 WTIM0 位 = 0 时

STT0 位 = 1 ↓（第二个 ST 之前）　　SPT0 位 = 1 ↓（SP 之前）

ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	SP

▲1　▲2　▲3　▲4　▲5　▲6　Δ7

▲1： IICS0 寄存器 = 1000X110B

▲2： IICS0 寄存器 = 1000X000B （WTIM0 位 = 1[注 1]）

▲3： IICS0 寄存器 = 1000XX00B （WTIM0 位 = 0[注 2]）

▲4： IICS0 寄存器 = 1000X110B

▲5： IICS0 寄存器 = 1000X000B （WTIM0 位 = 1[注 3]）

▲6： IICS0 寄存器 = 1000XX00B

Δ 7： IICS0 寄存器 = 00000001B

注　**1.** 要产生开始条件，将 WTIM0 位设置为 1，并改变中断请求信号 (INTIIC)的产生时序。
2. 将 WTIM0 位清除为 0，使设置回复初始。
3. 要产生停止条件，将 WTIM0 位设置为 1，并改变中断请求信号 (INTIIC)的产生时序。

备注　▲：总是产生
Δ：　仅当 SPIE0 位 = 1 时产生
X：　无需理会

<2> 当 WTIM0 位 = 1 时

STT0 位 = 1 ↓（第二个 ST 之前）　　SPT0 位 = 1 ↓（SP 之前）

ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	SP

▲1　▲2　▲3　▲4　Δ5

▲1： IICS0 寄存器 = 1000X110B

▲2： IICS0 寄存器 = 1000XX00B

▲3： IICS0 寄存器 = 1000X110B

▲4： IICS0 寄存器 = 1000XX00B

Δ 5： IICS0 寄存器 = 00000001B

备注　▲：总是产生
Δ：　仅当 SPIE0 位 = 1 时产生
X：　无需理会

（3） 开始 ~ 扩展码 ~ 数据 ~ 数据 ~ 停止（发送扩展码）

<1> 当 WTIM0 位 = 0 时

SPT0 位 = 1
↓

ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	SP

▲1　▲2　▲3　▲4　Δ5

▲1： IICS0 寄存器 = 1010X110B

▲2： IICS0 寄存器 = 1010X000B

▲3： IICS0 寄存器 = 1010X000B （WTIM0 位 = 1[注]）

▲4： IICS0 寄存器 = 1010XX00B

Δ 5： IICS0 寄存器 = 00000001B

注　要产生停止条件，将 WTIM0 位设置为 1，并改变中断请求信号 (INTIIC)的产生时序。

备注　▲： 总是产生
Δ： 仅当 SPIE0 位 = 1 时产生
X： 无需理会

<2> 当 WTIM0 位 = 1 时

SPT0 位 = 1
↓

ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	SP

▲1　▲2　▲3　Δ4

▲1： IICS0 寄存器 = 1010X110B

▲2： IICS0 寄存器 = 1010X100B

▲3： IICS0 寄存器 = 1010XX00B

Δ 4： IICS0 寄存器 = 00000001B

备注　▲： 总是产生
Δ： 仅当 SPIE0 位 = 1 时产生
X： 无需理会

17.7.2 从设备操作 （当接收从设备地址数据（地址一致）时）

（1） 开始 ~ 地址 ~ 数据 ~ 数据 ~ 停止

<1> 当 IICC0.WTIM0 位 = 0 时

ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	SP

▲1 ▲2 ▲3 Δ4

▲1： IICS0 寄存器 = 0001X110B

▲2： IICS0 寄存器 = 0001X000B

▲3： IICS0 寄存器 = 0001X000B

Δ 4： IICS0 寄存器 = 00000001B

备注 ▲：总是产生
Δ： 仅当 IICC0.SPIE0 位 = 1 时产生
X： 无需理会

<2> 当 WTIM0 位 = 1 时

ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	SP

▲1 ▲2 ▲3 Δ4

▲1： IICS0 寄存器 = 0001X110B

▲2： IICS0 寄存器 = 0001X100B

▲3： IICS0 寄存器 = 0001XX00B

Δ 4： IICS0 寄存器 = 00000001B

备注 ▲：总是产生
Δ： 仅当 SPIE0 位 = 1 时产生
X： 无需理会

（2） 开始 ~ 地址 ~ 数据 ~ 开始 ~ 地址 ~ 数据 ~ 停止

<1> 当 WTIM0 位 = 0 （重新开始后，地址一致）时

ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	SP
			▲1	▲2					▲3	▲4		Δ5

▲1： IICS0 寄存器 = 0001X110B

▲2： IICS0 寄存器 = 0001X000B

▲3： IICS0 寄存器 = 0001X110B

▲4： IICS0 寄存器 = 0001X000B

Δ 5： IICS0 寄存器 = 00000001B

备注 ▲： 总是产生
Δ： 仅当 SPIE0 位 = 1 时产生
X： 无需理会

<2> 当 WTIM0 位 = 1 （重新开始后，地址一致）时

ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	SP
			▲1		▲2				▲3		▲4	Δ5

▲1： IICS0 寄存器 = 0001X110B

▲2： IICS0 寄存器 = 0001XX00B

▲3： IICS0 寄存器 = 0001X110B

▲4： IICS0 寄存器 = 0001XX00B

Δ 5： IICS0 寄存器 = 00000001B

备注 ▲： 总是产生
Δ： 仅当 SPIE0 位 = 1 时产生
X： 无需理会

（3） 开始 ~ 地址 ~ 数据 ~ 开始 ~ 扩展码 ~ 数据 ~ 停止

<1> 当 **WTIM0** 位 **= 0** （重新开始后，地址不一致（扩展码））

ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	SP
				▲1	▲2				▲3		▲4	Δ5

▲1： IICS0 寄存器 = 0001X110B

▲2： IICS0 寄存器 = 0001X000B

▲3： IICS0 寄存器 = 0010X010B

▲4： IICS0 寄存器 = 0010X000B

Δ 5： IICS0 寄存器 = 00000001B

备注 ▲：总是产生
Δ： 仅当 SPIE0 位 = 1 时产生
X： 无需理会

<2> 当 **WTIM0** 位 **= 1** （重新开始后，地址不一致（扩展码））

ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	SP
				▲1		▲2			▲3	▲4		▲5 Δ6

▲1： IICS0 寄存器 = 0001X110B

▲2： IICS0 寄存器 = 0001XX00B

▲3： IICS0 寄存器 = 0010X010B

▲4： IICS0 寄存器 = 0010X110B

▲5： IICS0 寄存器 = 0010XX00B

Δ 6： IICS0 寄存器 = 00000001B

备注 ▲：总是产生
Δ： 仅当 SPIE0 位 = 1 时产生
X： 无需理会

（4） 开始 ~ 地址 ~ 数据 ~ 开始 ~ 地址 ~ 数据 ~ 停止

<1> 当 WTIM0 位 = 0 （重新开始后， 地址不一致 （**=** 非扩展码））时

ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	SP

▲1 ▲2 ▲3 Δ4

▲1： IICS0 寄存器 = 0001X110B

▲2： IICS0 寄存器 = 0001X000B

▲3： IICS0 寄存器 = 00000110B

Δ 4： IICS0 寄存器 = 00000001B

备注 ▲： 总是产生
Δ： 仅当 SPIE0 位 = 1 时产生
X： 无需理会

<2> 当 WTIM0 位 = 1 （重新开始后，地址不一致 （**=** 无扩展码））时

ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	SP

▲1 ▲2 ▲3 Δ4

▲1： IICS0 寄存器 = 0001X110B

▲2： IICS0 寄存器 = 0001XX00B

▲3： IICS0 寄存器 = 00000110B

Δ 4： IICS0 寄存器 = 00000001B

备注 ▲： 总是产生
Δ： 仅当 SPIE0 位 = 1 时产生
X： 无需理会

17.7.3 从设备操作 （当接收扩展码时）

当接收扩展码时，总是处于通信中。

（1） 开始 ~ 扩展码 ~ 数据 ~ 数据 ~ 停止

<1> 当 IICC0.WTIM0 位 = 0 时

ST	AD6 至 AD0	R/$\overline{\text{W}}$	$\overline{\text{ACK}}$	D7 至 D0	$\overline{\text{ACK}}$	D7 至 D0	$\overline{\text{ACK}}$	SP
		▲1		▲2		▲3		Δ4

▲1： IICS0 寄存器 = 0010X010B

▲2： IICS0 寄存器 = 0010X000B

▲3： IICS0 寄存器 = 0010X000B

Δ 4： IICS0 寄存器 = 00000001B

备注　▲：总是产生

Δ： 仅当 IICC0.SPIE0 位 = 1 时产生

X： 无需理会

<2> 当 WTIM0 位 = 1 时

ST	AD6 至 AD0	R/$\overline{\text{W}}$	$\overline{\text{ACK}}$	D7 至 D0	$\overline{\text{ACK}}$	D7 至 D0	$\overline{\text{ACK}}$	SP
		▲1	▲2		▲3		▲4	Δ5

▲1： IICS0 寄存器 = 0010X010B

▲2： IICS0 寄存器 = 0010X110B

▲3： IICS0 寄存器 = 0010X100B

▲4： IICS0 寄存器 = 0010XX00B

Δ 5： IICS0 寄存器 = 00000001B

备注　▲：总是产生

Δ： 仅当 SPIE0 位 = 1 时产生

X： 无需理会

（2） 开始 ~ 扩展码 ~ 数据 ~ 开始 ~ 地址 ~ 数据 ~ 停止

<1> 当 WTIM0 位 = 0 （重新开始后，地址一致）时

ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	SP

▲1 ▲2 ▲3 ▲4 △5

▲1： IICS0 寄存器 = 0010X010B

▲2： IICS0 寄存器 = 0010X000B

▲3： IICS0 寄存器 = 0001X110B

▲4： IICS0 寄存器 = 0001X000B

△ 5： IICS0 寄存器 = 00000001B

备注 ▲： 总是产生
△： 仅当 SPIE0 位 = 1 时产生
X： 无需理会

<2> 当 WTIM0 位 = 1 （重新开始后， 地址一致）时

ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	SP

▲1 ▲2 ▲3 ▲4 ▲5 △6

▲1： IICS0 寄存器 = 0010X010B

▲2： IICS0 寄存器 = 0010X110B

▲3： IICS0 寄存器 = 0010XX00B

▲4： IICS0 寄存器 = 0001X110B

▲5： IICS0 寄存器 = 0001XX00B

△ 6： IICS0 寄存器 = 00000001B

备注 ▲： 总是产生
△： 仅当 SPIE0 位 = 1 时产生
X： 无需理会

（3） 开始 ~ 扩展码 ~ 数据 ~ 开始 ~ 扩展码 ~ 数据 ~ 停止

<1> 当 WTIM0 位 = 0 （重新开始后，扩展码接收）时

ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	SP

▲1 ▲2 ▲3 ▲4 Δ5

▲1： IICS0 寄存器 = 0010X010B

▲2： IICS0 寄存器 = 0010X000B

▲3： IICS0 寄存器 = 0010X010B

▲4： IICS0 寄存器 = 0010X000B

Δ 5： IICS0 寄存器 = 00000001B

备注 ▲：总是产生

Δ： 仅当 SPIE0 位 = 1 时产生

X： 无需理会

<2> 当 WTIM0 位 = 1 （重新开始后，扩展码接收）时

ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	SP

▲1 ▲2 ▲3 ▲4 ▲5 ▲6 Δ7

▲1： IICS0 寄存器 = 0010X010B

▲2： IICS0 寄存器 = 0010X110B

▲3： IICS0 寄存器 = 0010XX00B

▲4： IICS0 寄存器 = 0010X010B

▲5： IICS0 寄存器 = 0010X110B

▲6： IICS0 寄存器 = 0010XX00B

Δ 7： IICS0 寄存器 = 00000001B

备注 ▲：总是产生

Δ： 仅当 SPIE0 位 = 1 时产生

X： 无需理会

（4） 开始 ~ 扩展码 ~ 数据 ~ 开始 ~ 地址 ~ 数据 ~ 停止

<1> 当 WTIM0 位 = 0 （重新开始后，地址不一致 （= 无扩展码））时

ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	SP

▲1 ▲2 ▲3 Δ4

▲1： IICS0 寄存器 = 0010X010B

▲2： IICS0 寄存器 = 0010X000B

▲3： IICS0 寄存器 = 00000110B

Δ 4： IICS0 寄存器 = 00000001B

备注 ▲：总是产生
Δ： 仅当 SPIE0 位 = 1 时产生
X： 无需理会

<2> 当 WTIM0 位 = 1 （重新开始后， 地址不一致 （= 无扩展码））时

ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	SP

▲1 ▲2 ▲3 ▲4 Δ5

▲1： IICS0 寄存器 = 0010X010B

▲2： IICS0 寄存器 = 0010X110B

▲3： IICS0 寄存器 = 0010XX00B

▲4： IICS0 寄存器 = 00000110B

Δ 5： IICS0 寄存器 = 00000001B

备注 ▲：总是产生
Δ： 仅当 SPIE0 位 = 1 时产生
X： 无需理会

17.7.4 不通信时的操作

（1） 开始 ~ 扩展码 ~ 数据 ~ 数据 ~ 停止

ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	SP
								Δ1

Δ 1： IICS0 寄存器 = 00000001B

备注 Δ： 仅当 SPIE0 位 = 1 时产生

17.7.5 仲裁失败操作（仲裁失败后作为从设备操作）

在多主系统中用作主设备时，在每次发生 INTIIC 中断时检查仲裁结果，通过读取 IICS0.MSTS0 位检查仲裁结果。

（1） 当在从设备地址数据传输期间发生仲裁失败

<1> 当 IICC0.WTIM0 位 = 0 时

ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	SP

▲1 ▲2 ▲3 Δ4

▲1： IICS0 寄存器 = 0101X110B

▲2： IICS0 寄存器 = 0001X000B

▲3： IICS0 寄存器 = 0001X000B

Δ 4： IICS0 寄存器 = 00000001B

备注 ▲： 总是产生
Δ： 仅当 IICC0.SPIE0 位 = 1 时产生
X： 无需理会

<2> 当 WTIM0 位 = 1 时

ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	SP

▲1 ▲2 ▲3 Δ4

▲1： IICS0 寄存器 = 0101X110B

▲2： IICS0 寄存器 = 0001X100B

▲3： IICS0 寄存器 = 0001XX00B

Δ 4： IICS0 寄存器 = 00000001B

备注 ▲： 总是产生
Δ： 仅当 SPIE0 位 = 1 时产生
X： 无需理会

（2） 在扩展码的传输期间发生仲裁失败

<1> 当 WTIM0 位 = 0 时

ST	AD6 至 AD0	R/$\overline{\text{W}}$	$\overline{\text{ACK}}$	D7 至 D0	$\overline{\text{ACK}}$	D7 至 D0	$\overline{\text{ACK}}$	SP

▲1 ▲2 ▲3 Δ4

▲1： IICS0 寄存器 = 0110X010B

▲2： IICS0 寄存器 = 0010X000B

▲3： IICS0 寄存器 = 0010X000B

Δ 4： IICS0 寄存器 = 00000001B

备注 ▲：总是产生
Δ： 仅当 SPIE0 位 = 1 时产生
X： 无需理会

<2> 当 WTIM0 位 = 1 时

ST	AD6 至 AD0	R/$\overline{\text{W}}$	$\overline{\text{ACK}}$	D7 至 D0	$\overline{\text{ACK}}$	D7 至 D0	$\overline{\text{ACK}}$	SP

▲1 ▲2 ▲3 ▲4 Δ5

▲1： IICS0 寄存器 = 0110X010B

▲2： IICS0 寄存器 = 0010X110B

▲3： IICS0 寄存器 = 0010X100B

▲4： IICS0 寄存器 = 0010XX00B

Δ 5： IICS0 寄存器 = 00000001B

备注 ▲：总是产生
Δ： 仅当 SPIE0 位 = 1 时产生
X： 无需理会

17.7.6 发生仲裁失败时的操作（在仲裁失败后不进行通信）

在多主系统中用作主设备时，在每次发生 INTIIC 中断时检查仲裁结果，通过读取 IICS0.MSTS0 位检查仲裁结果。

（1） 在从设备地址数据传输期间发生仲裁失败

ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	SP

▲1　　Δ2

▲1： IICS0 寄存器 = 01000110B

Δ 2： IICS0 寄存器 = 00000001B

备注　▲：总是产生

Δ： 仅当 IICC0.SPIE0 位 = 1 时产生

（2） 在扩展码的传输期间发生仲裁失败

ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	SP

▲1　　Δ2

▲1：IICS0 寄存器 = 0110X010B

由软件设置 IICC0.LREL0 位。

Δ 2：IICS0 寄存器 = 00000001B

备注　▲：总是产生

Δ： 仅当 SPIE0 位 = 1 时产生

X： 无需理会

（3）　在数据传输期间发生仲裁失败

<1> 当 **IICC0.WTIM0** 位 **= 0** 时

ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	SP

▲1　▲2　Δ3

▲1：IICS0 寄存器 = 10001110B

▲2：IICS0 寄存器 = 01000000B

Δ 3：IICS0 寄存器 = 00000001B

备注　▲：总是产生

Δ：仅当 SPIE0 位 = 1 时产生

<2> 当 **WTIM0** 位 **= 1** 时

ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	SP

▲1　▲2　Δ3

▲1：IICS0 寄存器 = 10001110B

▲2：IICS0 寄存器 = 01000100B

Δ 3：IICS0 寄存器 = 00000001B

备注　▲：总是产生

Δ：仅当 SPIE0 位 = 1 时产生

（4） 在数据传输期间，因为重新开始条件导致发生仲裁失败

<1> 无扩展码（例如： 地址不一致）

ST	AD6 至 AD0	$R/\overline{W}$	$\overline{ACK}$	D7 至 Dn	ST	AD6 至 AD0	$R/\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	SP

▲1 ▲2 Δ3

▲1： IICS0 寄存器 = 1000X110B

▲2： IICS0 寄存器 = 01000110B

Δ 3： IICS0 寄存器 = 00000001B

备注 ▲： 总是产生

Δ： 仅当 SPIE0 位 = 1 时产生

X： 无需理会

Dn = D6 至 D0

<2> 扩展码

ST	AD6 至 AD0	$R/\overline{W}$	$\overline{ACK}$	D7 至 Dn	ST	AD6 至 AD0	$R/\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	SP

▲1 ▲2 Δ3

▲1： IICS0 寄存器 = 1000X110B

▲2： IICS0 寄存器 = 0110X010B

由软件设置 IICC0.LREL0 位。

Δ 3： IICS0 寄存器 = 00000001B

备注 ▲： 总是产生

Δ： 仅当 SPIE0 位 = 1 时产生

X： 无需理会

Dn = D6 至 D0

（**5**） 在数据传输期间，因为停止条件导致发生仲裁失败

ST	AD6 至 AD0	$R/\overline{W}$	$\overline{ACK}$	D7 至 Dn	SP

▲1 Δ2

▲1： IICS0 寄存器 = 1000X110B

Δ 2： IICS0 寄存器 = 01000001B

备注 ▲：总是产生

Δ： 仅当 SPIE0 位 = 1 时产生

X： 无需理会

Dn = D6 至 D0

（6） 当试图产生重新开始条件时，因为 **SDAn** 引脚的低电平导致发生仲裁失败

<1> 当 WTIM0 位 = 0 时

IICC0.STT0 位 = 1
↓

ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	SP

▲1 ▲2 ▲3 ▲4 Δ5

▲1： IICS0 寄存器 = 1000X110B

▲2： IICS0 寄存器 = 1000X000B （WTIM0 位 = 1）

▲3： IICS0 寄存器 = 1000X100B （WTIM0 位 = 0）

▲4： IICS0 寄存器 = 01000000B

Δ 5： IICS0 寄存器 = 00000001B

备注 ▲： 总是产生
Δ： 仅当 SPIE0 位 = 1 时产生
X： 无需理会

<2> 当 WTIM0 位 = 1 时

IICC0.STT0 位 = 1
↓

ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	SP

▲1 ▲2 ▲3 Δ4

▲1： IICS0 寄存器 = 1000X110B

▲2： IICS0 寄存器 = 1000X100B

▲3： IICS0 寄存器 = 01000100B

Δ 4： IICS0 寄存器 = 00000001B

备注 ▲： 总是产生
Δ： 仅当 SPIE0 位 = 1 时产生
X： 无需理会

（7） 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

<1> 当 WTIM0 位 = 0 时

STT0 位 = 1 ↓（SP 前）

ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	SP

▲1（$\overline{ACK}$ 后） ▲2（D7 至 D0 后） ▲3（$\overline{ACK}$ 后） Δ4（SP 后）

▲1： IICS0 寄存器 = 1000X110B

▲2： IICS0 寄存器 = 1000X000B （WTIM0 位 = 1）

▲3： IICS0 寄存器 = 1000XX00B

Δ 4： IICS0 寄存器 = 01000001B

备注 ▲：总是产生
Δ： 仅当 SPIE0 位 = 1 时产生
X： 无需理会

<2> 当 WTIM0 位 = 1 时

STT0 位 = 1 ↓（SP 前）

ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	SP

▲1（$\overline{ACK}$ 后） ▲2（$\overline{ACK}$ 后） Δ3（SP 后）

▲1： IICS0 寄存器 = 1000X110B

▲2： IICS0 寄存器 = 1000XX00B

Δ 3： IICS0 寄存器 = 01000001B

备注 ▲：总是产生
Δ： 仅当 SPIE0 位 = 1 时产生
X： 无需理会

（8） 当试图产生停止条件时，由于 **SDAn** 引脚为低电平导致发生仲裁失败

<1> 当 WTIM0 位 = 0 时

IICC0.SPT0 位 = 1
↓

ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	SP

▲1 ▲2 ▲3 ▲4 Δ5

▲1： IICS0 寄存器 = 1000X110B

▲2： IICS0 寄存器 = 1000X000B （WTIM0 位 = 1）

▲3： IICS0 寄存器 = 1000X100B （WTIM0 位 = 0）

▲4： IICS0 寄存器 = 01000100B

Δ 5： IICS0 寄存器 = 00000001B

备注 ▲： 总是产生
Δ： 仅当 SPIE0 位 = 1 时产生
X： 无需理会

<2> 当 WTIM0 位 = 1 时

IICC0.SPT0 位 = 1
↓

ST	AD6 至 AD0	R/$\overline{W}$	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	D7 至 D0	$\overline{ACK}$	SP

▲1 ▲2 ▲3 Δ4

▲1： IICS0 寄存器 = 1000X110B

▲2： IICS0 寄存器 = 1000X100B

▲3： IICS0 寄存器 = 01000100B

Δ 4： IICS0 寄存器 = 00000001B

备注 ▲： 总是产生
Δ： 仅当 SPIE0 位 = 1 时产生
X： 无需理会

17.8 中断请求信号（INTIIC）的产生时序和等待控制

IICC0.WTIM0 位的设置可以决定 INTIIC 信号的产生时序和对应的等待控制，如下所示。

表 17-3. INTIIC 信号产生时序和等待控制

WTIM0 位	从设备操作期间			主设备操作期间		
	地址	数据接收	数据发送	地址	数据接收	数据发送
0	9[注 1, 2]	8[注 2]	8[注 2]	9	8	8
1	9[注 1, 2]	9[注 2]	9[注 2]	9	9	9

注 **1.** 仅当接收的地址和 SVA0 寄存器设定的值相一致时，在第 9 个时钟的下降沿产生从设备的 INTIIC 信号和等待时期。
此刻，不论 IICC0.ACKE0 位如何设置，都产生 $\overline{ACK}$ 信号，对于已经接收到扩展码的从设备，INTIIC 信号在第 8 个时钟的下降沿发生。
当重新开始后地址不一致时，INTIIC 信号在第 9 个时钟的下降沿发生，但是，不发生等待。

2. 如果接收地址和 SVA0 寄存器的内容不一致，而且不能收到扩展码，则既不产生 INTIIC 信号，也不产生等待状态。

备注 表中的数量表示串行时钟信号的时钟数目，中断请求信号和等待控制都和这些时钟信号的下降沿同步。

（1） 地址发送/接收期间

- 从设备操作： 根据上述注 1 和注 2 的条件来决定中断请求和等待时序，无需考虑 WTIM0 位的设置。
- 主设备操作： 不论 WTIM0 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

- 主设备/从设备操作： 中断请求和等待时序由 WTIM0 位的设置决定。

（3） 数据发送期间

- 主设备/从设备操作： 中断请求和等待时序由 WTIM0 位的设置决定。

（4） 等待取消方法

等待状态的四种取消方法如下所示：

- 向 IIC0 寄存器写入数据。
- 设置 IICC0.WREL0 位（取消等待状态）。
- 设置 IICC0.STT0 位（产生开始条件）[注]。
- 设置IICC0.SPT0 位（产生停止条件）[注]。

注 仅限主设备。

当已经选择 8 时钟等待时（WTIM0 位 = 0），在取消等待之前必须确定是否已经产生 $\overline{ACK}$ 信号。

（5） 停止条件检测

当检测到停止条件时，会产生 INTIIC 信号。

17.9 地址一致检测方法

在I^2C 总线模式下，通过发送从设备的地址，主设备可以选择特定的从设备。

地址一致的检测由硬件自动执行。当本地地址置入 SVA0 寄存器以后，且置入 SVA0 寄存器的本地地址和主设备发送的从设备地址一致时，或接收到扩展码时，产生 INTIIC 中断请求信号。

17.10 错误检测

在I^2C 总线模式下，串行数据总线（SDA）在数据传送期间的状态被发送设备的IIC0 寄存器捕获，于是，发送之前IIC0 寄存器内的数据就可以和已发送的IIC0 数据进行比较，从而能够检测传输错误。当比较值不一致时，判断为发生传输错误。

17.11 扩展码

（1）当接收地址的高 4 位是 0000 或 1111 时，扩展码标志（IICS0.EXC0 位）被置位，表示扩展码接收，且在第 8 个时钟的下降沿处产生一个中断请求信号 （INTIIC）。
存储在 SVA0 寄存器中的本地地址不受影响。

（2）如果通过 10 位地址传输将 11110xx0 设置到 SVA0 寄存器，且 11110xx0 数据由主设备发出，其结果如下所示。 注意， INTIIC 信号在第 8 个时钟的下降沿处产生。

- 数据一致的高 4 位： IICS0.EXC0 位 = 1
- 数据一致的第 7 位： IICS0.COI0 位 = 1

（3） 根据扩展码之后的数据，INTIIC 信号发生后的处理也各不相同，这些处理由软件执行。接收到扩展码的从设备总是处于通信中，即使地址不一致。
例如，接收到扩展码以后，当不希望作为从设备操作时，将 IICC0.LREL0 位设置为 1，CPU 将进入下一次通信等待状态。

表 17-4. 主扩展码位的位定义

从设备地址	$R/\overline{W}$ 位	描述
0000 000	0	通用呼叫地址
1111 0xx	0	10 位从设备地址规范 (地址认证时)
1111 0xx	1	10 位从设备地址规范 (地址一致之后的读取命令发出时)

备注 关于上述情况之外的扩展码，参见NXP 半导体发布的 I^2C 总线规范。

17.12 仲裁

当多个主设备同时产生开始条件时（在 IICS0.STD0 位置 1 之前，IICC0.STT0 位已置为 1），调整时钟数目，直到数据不同，多个主设备之间的通信才能执行，这种操作称为仲裁 。

当一个主设备在仲裁中失败时，在仲裁失败发生时序处将仲裁失败标志（IICS0.ALD0 位）设置为 1，且 SCL 信号线和 SDA 信号线都设置为高阻状态，释放总线。

仲裁失败的检测基于下次中断请求信号 （INTIIC）的时序（当检测到停止条件时，第 8 或第 9 个时钟）和通过软件对 ALD0 位置 1 的设置。

关于中断请求信号时序的详情， 参见 **17.7 I²C 中断请求信号 （INTIIC）**。

图 **17-12.** 仲裁时序示例

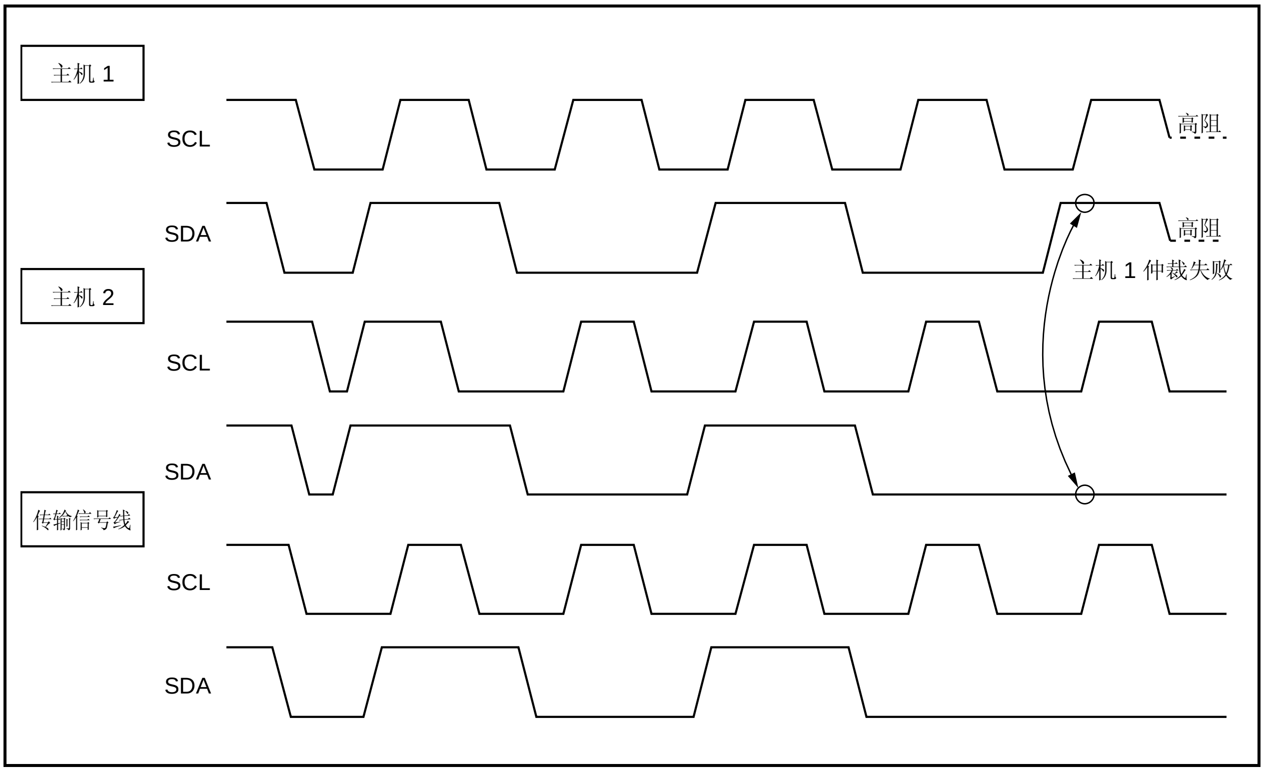


表 17-5. 仲裁期间的状态和中断请求产生时序

仲裁期间的状态	中断请求产生时序
在地址发送期间	字节传输之后的第 8 个或第 9 个时钟的下降沿[注 1]
地址发送后读取/写入数据	
在扩展码发送期间	
扩展码发送后读取/写入数据	
在数据发送期间	
数据接收后 $\overline{\text{ACK}}$ 传送期间	
在数据传输期间检测到重新开始条件	
在数据传输期间检测到停止条件	当产生停止条件时（当IICC0.SPIE0 位 = 1 时）[注 2]
试图产生重新开始条件时 SDA 引脚是低电平	字节传输之后的第 8 个或第 9 个时钟的下降沿[注 1]
试图产生重新开始条件时检测到停止条件	当产生停止条件时（当IICC0.SPIE0 位 = 1 时）[注 2]
试图产生停止条件时 SDA 引脚是低电平	字节传输之后的第 8 个或第 9 个时钟的下降沿[注 1]
试图产生重新开始条件时 SCL 引脚是低电平	

注 **1.** 当 IICC0.WTIM0 位 = 1 时，中断请求在第 9 个时钟的下降沿发生。当 WTIM0 位 = 0 且接收到扩展码的从地址时，中断请求在第 8 个时钟的下降沿发生。

2. 当有可能发生仲裁时，在主设备操作要设置 SPIE0 位 = 1。

17.13 唤醒功能

已经接收到本地地址或扩展码时，I²C总线从设备的功能是产生中断请求信号（INTIIC） 。

当地址数据不一致时，该功能就会阻止不必要的中断请求，从而使处理更有效率。

当检测到开始条件后，待机模式被唤醒。仲裁失败可能将主设备（已经产生开始条件）变为从设备，在这种可能情况下发送地址时，待机模式唤醒有效。

但是，当检测到停止条件后，不论唤醒功能如何设置，IICC0.SPIE0 位总是置位，这可以决定中断请求的使能或禁止 。

17.14 通信保留

17.14.1 使能通信保留功能（IICF0.IICRSV0 位 = 0）时

在目前不能使用总线的情况下，开始主设备通信，通信保留功能可以在总线释放后使能开始条件的传输。在两种模式下，总线不能使用。

- 当仲裁结果既不是主设备操作，又不是从设备操作。
- 当接收到扩展码后且从设备操作禁止时（当 IICC0.LREL0 位置为 1 时，不返回 $\overline{\text{ACK}}$ 信号且总线释放）。

如果在不能使用总线时，将 IICC0.STT0 位置位（1），将自动产生开始条件，并且在总线释放后（检测到停止条件后）设置等待状态。

将 IICC0.SPIE0 位设置为 1，自动启动通信并作为主设备，通过中断请求（INTIIC）的发生来检测总线释放（检测停止条件），然后将地址写入 IIC0 寄存器。在检测到停止条件之前，认为写入 IIC0 寄存器的数据无效。

当 STT0 位设置为 1 后，根据总线状态决定工作模式（开始条件或通信保留）。

如果总线已经释放..产生开始条件
如果总线未释放（待机模式）..通信保留

要检测为 STT0 位决定何种工作模式，将 STT0 位设置为 1，待等待时期完毕后，检查 IICS0.MSTS0 位。

应该由软件设置的等待期间在表 17-6 中列出。

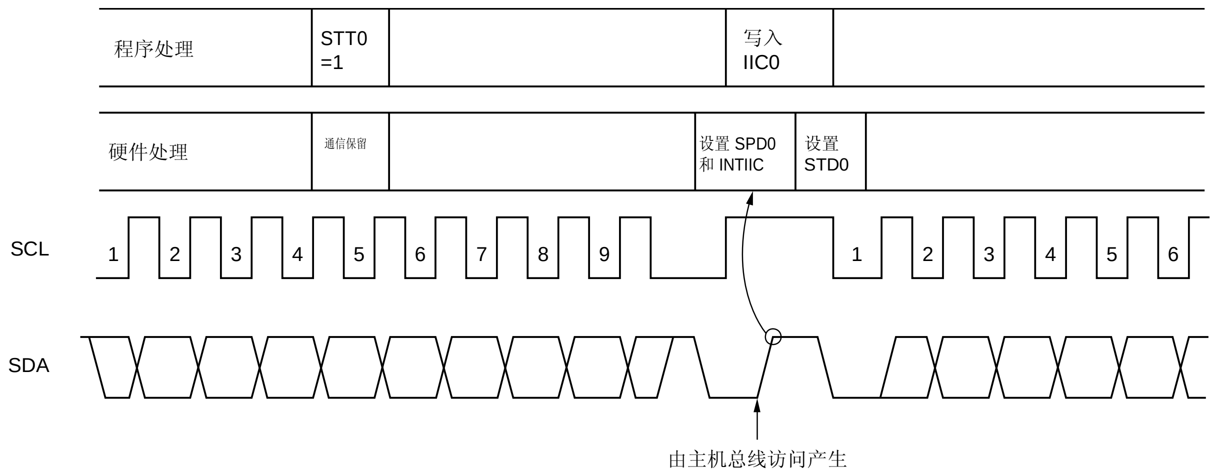
这些等待时期可以通过 IICX0.CLX0，IICCL0.SMC0 和 IICCL0.CL00 位进行设置。

表 17-6. 等待时期

时钟选择	CLX0	SMC0	CL00	等待时钟	等待时间 当 f_{XX} = 64 MHz 时
f_{XX}/8 (IICOCKS = 12H)	0	0	0	23 个时钟	2.88 μs
f_{XX}/10 (IICOCKS = 13H)	0	0	0	23 个时钟	3.59 μs
f_{XX}/4 (IICOCKS = 10H)	0	0	1	43 个时钟	2.69 μs
f_{XX}/6 (IICOCKS = 11H)	0	0	1	43 个时钟	4.03 μs
f_{XX}/8 (IICOCKS = 12H)	0	0	1	43 个时钟	5.38 μs
f_{XX}/10 (IICOCKS = 13H)	0	0	1	43 个时钟	6.72 μs
f_{XX}/4 (IICOCKS = 10H)	0	1	x	15 个时钟	0.94 μs
f_{XX}/6 (IICOCKS = 11H)	0	1	x	15 个时钟	1.41 μs
f_{XX}/8 (IICOCKS = 12H)	0	1	x	15 个时钟	1.88 μs
f_{XX}/10 (IICOCKS = 13H)	0	1	x	15 个时钟	2.34 μs
f_{XX}/8 (IICOCKS = 12H)	1	1	x	9 个时钟	1.13 μs
f_{XX}/10 (IICOCKS = 13H)	1	1	x	9 个时钟	1.41 μs

通信保留时序如下所示。

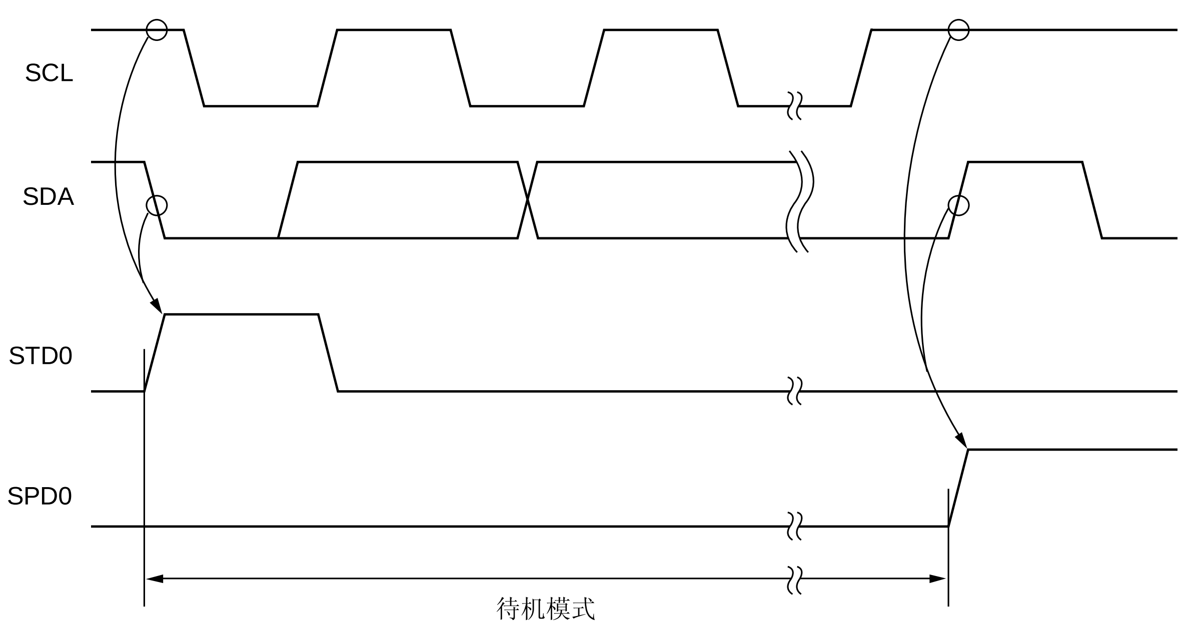
图 17-13. 通信保留时序



IIC0： IIC 移位寄存器 0
STT0： IIC 控制寄存器 0（IICC0）第 1 位
STD0： IIC 状态寄存器 0（IICS0）第 1 位
SPD0： IIC 状态寄存器 0（IICS0）第 0 位

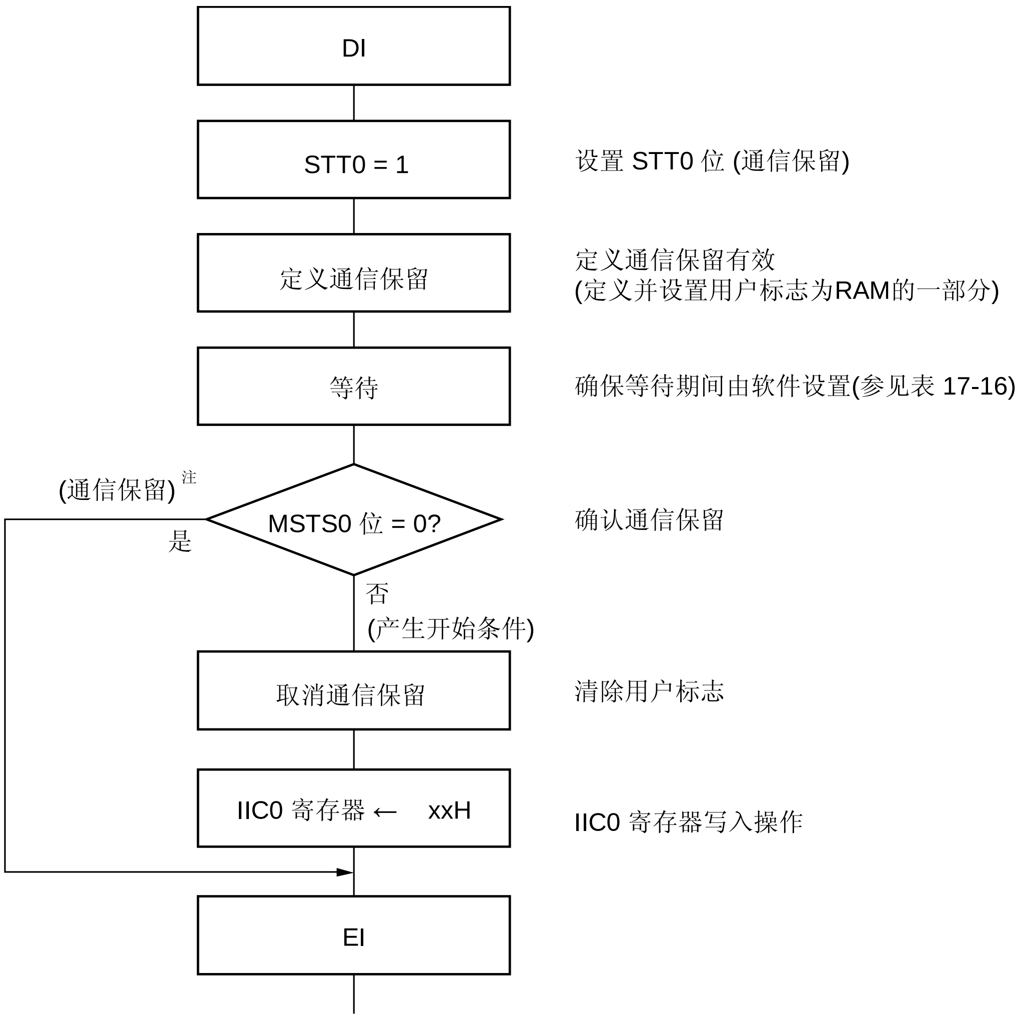
通过以下时序受理通信保留。IICS0.STD0 位设置为 1 后，在检测到停止条件前，设置 IICC0.STT0 位为 1 就可以产生一次通信保留。

图 17-14. 受理通信保留的时序



通信保留的流程图展示如下。

图 17-15. 通信保留的流程图



注 当发生停止条件中断请求时，通信保留执行写入到 IIC0 寄存器的操作。

17.14.2 禁止通信保留功能时（IICF0.IICRSV0 位 = 1）

在总线通信中总线没有使用时，当 IICC0.STT0 位置位时，通信保留请求被拒绝，且不会产生开始条件。在下列两种状态下不能使用总线。

- 当仲裁结果既不是主设备操作，又不是从设备操作。
- 当接收到扩展码后且从设备操作禁止时（当 IICC0.LREL0 位置为 1 时，不返回 $\overline{\text{ACK}}$ 信号且总线释放）。

为了确认是产生了开始条件，还是请求遭到拒绝，检查 IICF0.STCF0 标志。在 STT0 位 = 1 后，直到 STCF0 标志置位，需要的时间在表 17-7 中列出。 因而，通过软件来保证这段需要的时间。

表 17-7. 等待期

选择时钟	CLX0	SMC0	CL00	等待时钟	等待时间 当 f_{XX} = 64 MHz 时
f_{XX}/8 (IICOCKS = 12H)	0	0	0	5 个时钟	0.63 μs
f_{XX}/10 (IICOCKS = 13H)	0	0	0	5 个时钟	0.78 μs
f_{XX}/4 (IICOCKS = 10H)	0	0	1	5 个时钟	0.31 μs
f_{XX}/6 (IICOCKS = 11H)	0	0	1	5 个时钟	0.47 μs
f_{XX}/8 (IICOCKS = 12H)	0	0	1	5 个时钟	0.63 μs
f_{XX}/10 (IICOCKS = 13H)	0	0	1	5 个时钟	0.78 μs

17.15 注意事项

（1）当 IICF0.STCEN0 位 = 0 时

I²C操作使能之后，不管总线的实际状态如何，总是立即认为处于总线通信状态（IICF0.IICBSY0 位 = 1）。 要在尚未检测到停止条件的情况下执行主设备通信，先产生停止条件，在开始主设备通信之前释放总线。使用以下次序来产生停止条件。

<1> 设置 IICCL0 寄存器。
<2> 设置 IICC0.IICE0 位。
<3> 设置 IICC0.SPT0 位。

（2）当 IICF0.STCEN0 位 = 1 时

I²C操作使能之后，不管总线的实际状态如何，总是立即认为处于总线释放状态（IICF0.IICBSY0 位 = 0）。若要产生首个开始条件 （IICC0.STT0 位 = 1），必需确认总线已经被释放，以免干扰其它通信。

（3）在和其它设备进行通信时，当 V850E/IF3 和 V850E/IG3 的 IICC0.IICE0 位设置为 1，根据通信线的状态，可能会检测到开始条件。当 SCL 和 SDA 信号线是高电平时，请确保将 IICC0.IICE0 位设置为 1。

（4）使能操作（IICC0.IICE0 位 = 1）之前，通过 IICCL0 ，IICX0 和 IICOCKS 寄存器决定工作时钟频率。要改变操作时钟频率，要将 IICC0.IICE0 位清 0 一次。

（5）当 IICC0.STT0 和 IICC0.SPT0 位已经设置为 1，若不首先将其清 0，则一定不要对其再次置位。

（6）如果传输已经保留，设置IICC0.SPIE0 位为 1，这样，检测到停止条件就会产生一个中断请求信号。当产生中断请求信号后，向I²C写入通信数据将会释放等待状态，然后传输开始。如果检测到停止条件后不产生中断请求信号，由于未产生中断请求，传输将暂停在等待状态。但是，没有必要为了软件对IICS0.MSTS0 位的检测而将SPIE0 位置为 1 。

17.16 通信操作

以下展示 3 种操作过程的流程图。

（1）主设备工作于单主系统

当使用 V850E/IF3 和 V850E/IG3 单片机作为单主系统中的主设备时，流程图如下所示。

流程图大体可以分为初始设置和通信处理两部分。在启动时执行初始设置。如果需要和从设备进行通信，进行通信准备后，执行通信处理。

（2）主设备工作于多主系统

在I²C总线多主系统中，当总线参与通信之后，不能由I²C总线规范判定总线是释放还是使用。这里，当数据线和时钟线都在一定的时期（1 帧）内保持为高电平，V850E/IF3 和V850E/IG3 就认为总线已释放并可以进行通信。

操作流程可以大体分为初始设置、通信等待和通信处理。V850E/IF3 和 V850E/IG3 在仲裁中失败并指定为从设备的处理情况这里予以省略，而只是展示作为主设备的处理操作。启动时执行初始设置以便参与通信。然后，作为主设备等待通信请求或作为从设备等待指定。在通信处理中执行实际的通信，且支持与从设备的发送/接收操作和与其它主设备间的仲裁处理。

（3）从设备操作

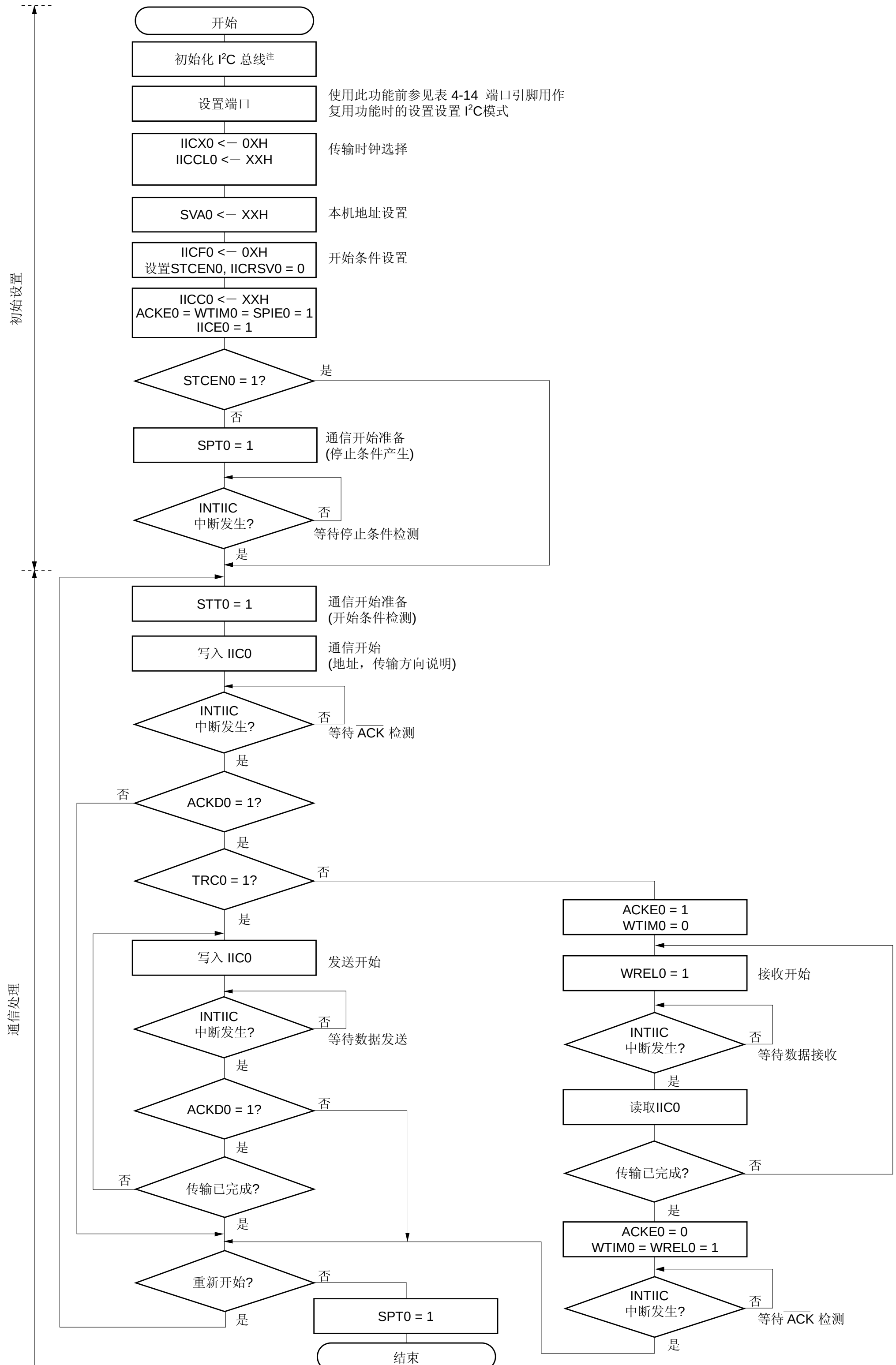
当 V850E/IF3 和 V850E/IG3 用作从设备时的示例如下。

当用作从设备时，由一个中断启动操作。在启动时执行初始设置，然后，等待 INTIIC 中断信号的发生（通信等待）。当 INTIIC 中断发生时，通信状态被判定，并且判定结果作为一个标志传送给主处理程序。

通过检查该标志，可以执行必需的通信处理。

17.16.1 主设备工作于单主系统

图 17-16. 单主系统的主设备操作

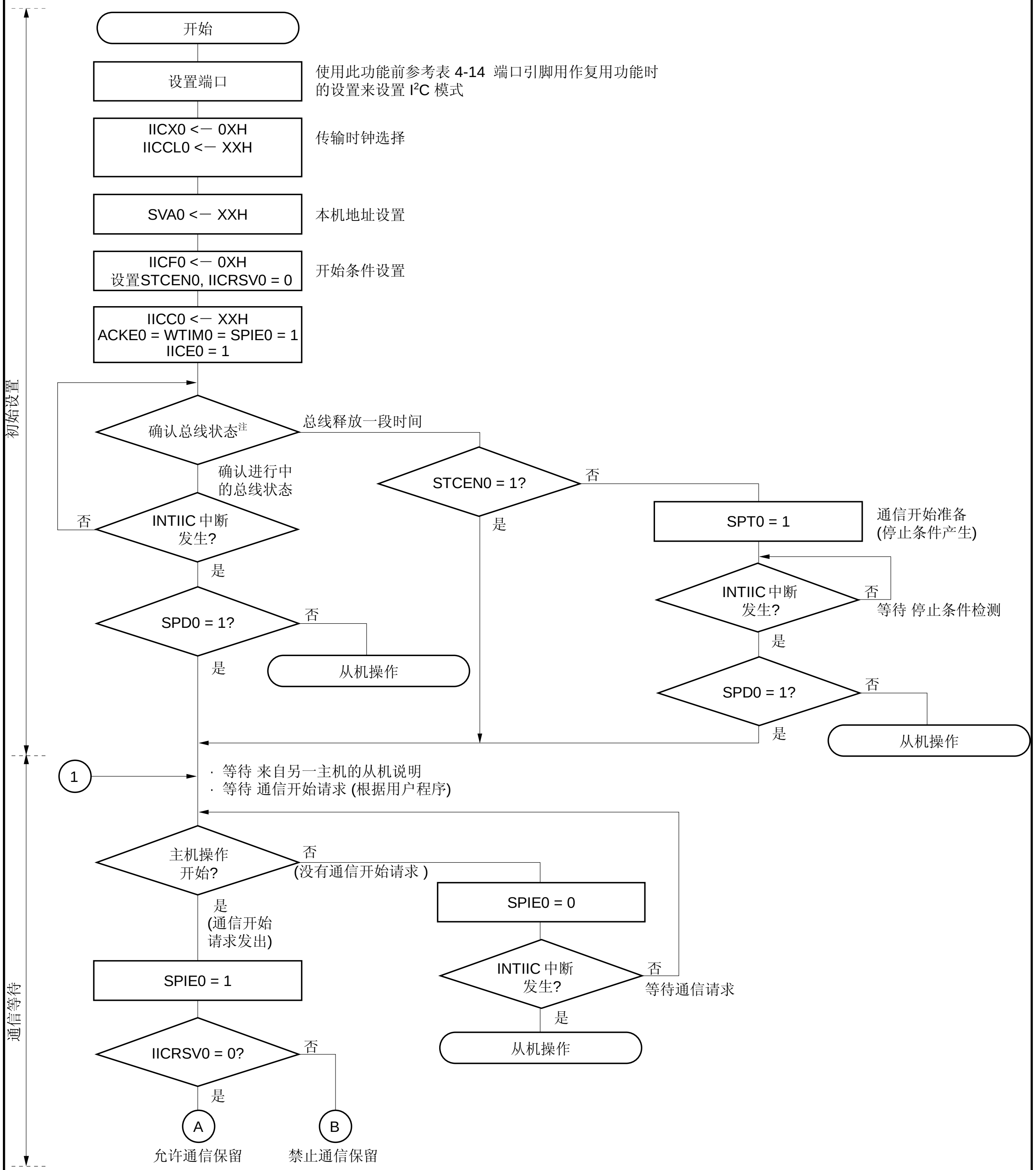


注　　释放I²C总线（SCL，SDA 引脚 = 高电平），与进行通信的产品规格一致。例如，当EEPROM™ 输出低电平到SDA 引脚时，设置SCL引脚为输出端口并从中输出时钟脉冲，直到SDA引脚为稳定的高电平。

备注　　　至于发送格式和接收格式，符合进行通信的产品的规格。

17.16.2 主设备工作于多主系统

图 17-17. 多主系统中的主设备操作（1/3）



注　确定总线释放状态（IICCL0.CLD0 位 = 1，IICCL0.DAD0 位 = 1）已经维持了一段时间（比如：1 帧）。当SDA 引脚稳定为低电平时，参考通信产品的规格书，决定是否释放I²C总线（SCL，SDA引脚 = 高电平）。

图 17-17. 多主系统中的主设备操作（2/3）

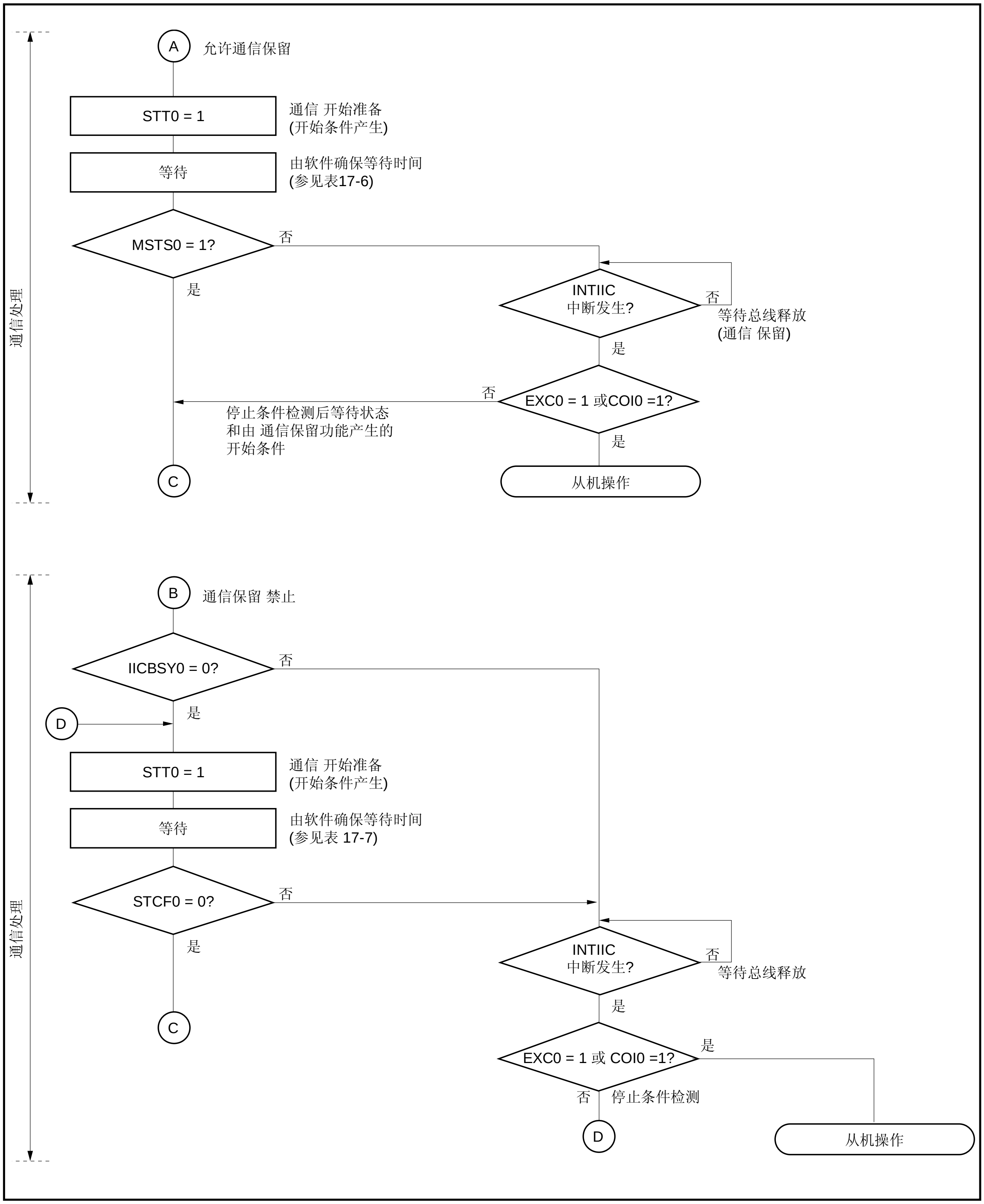
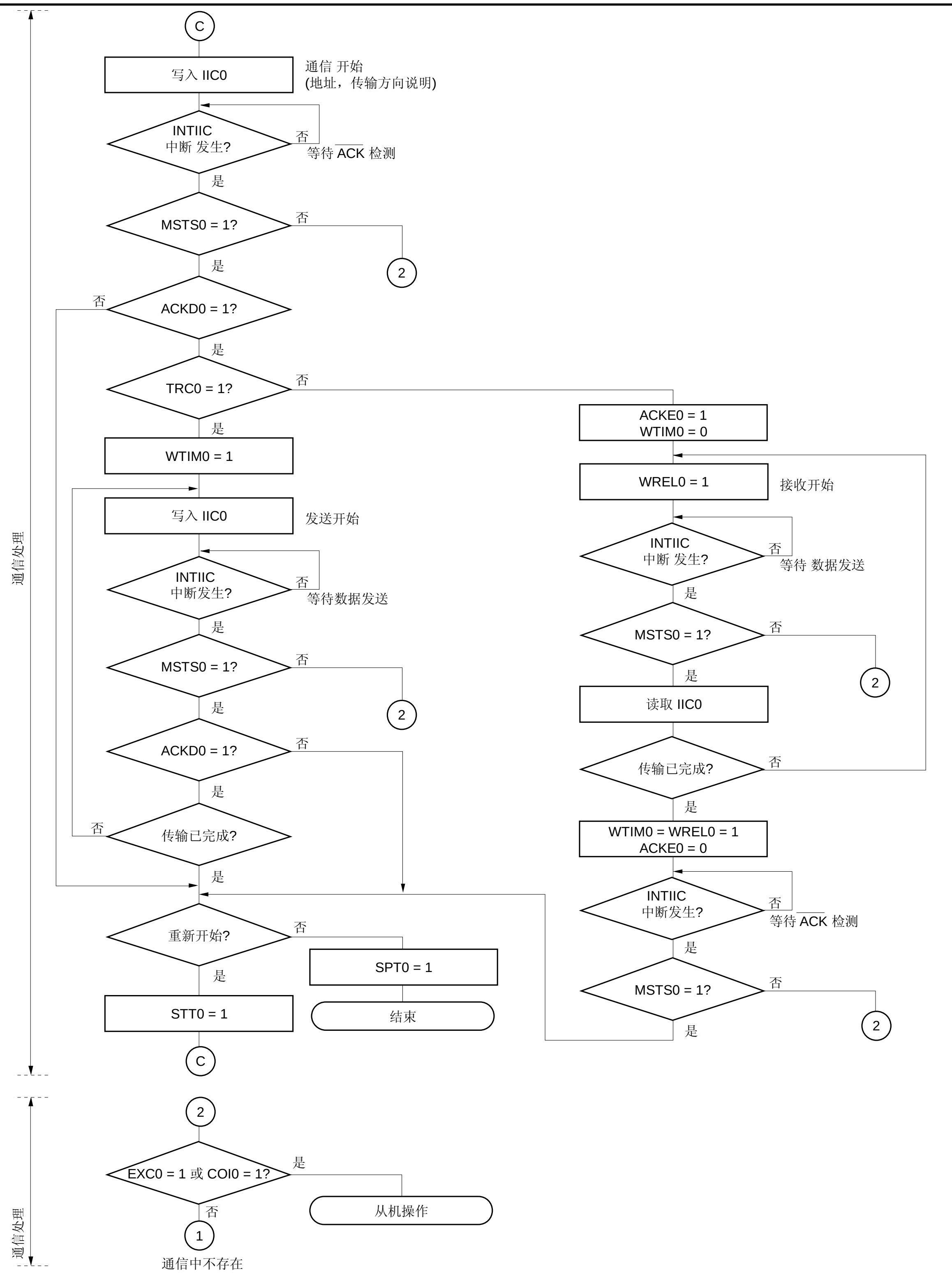


图 17-17. 多主系统中的主设备操作（3/3）



备注

1. 确认发送和接收格式与参与通信的产品规格一致。
2. 当 V850E/IF3 和 V850E/IG3 单片机在多主系统中用作主设备时，每次发生 INTIIC 中断时都要读取 IICS0.MSTS0 位以确定仲裁结果。
3. 当 V850E/IF3 和 V850E/IG3 单片机在多主系统中用作从设备时，每次发生 INTIIC 中断时都要使用 IICS0 和 IICF0 寄存器确认状态以决定下一步的处理。

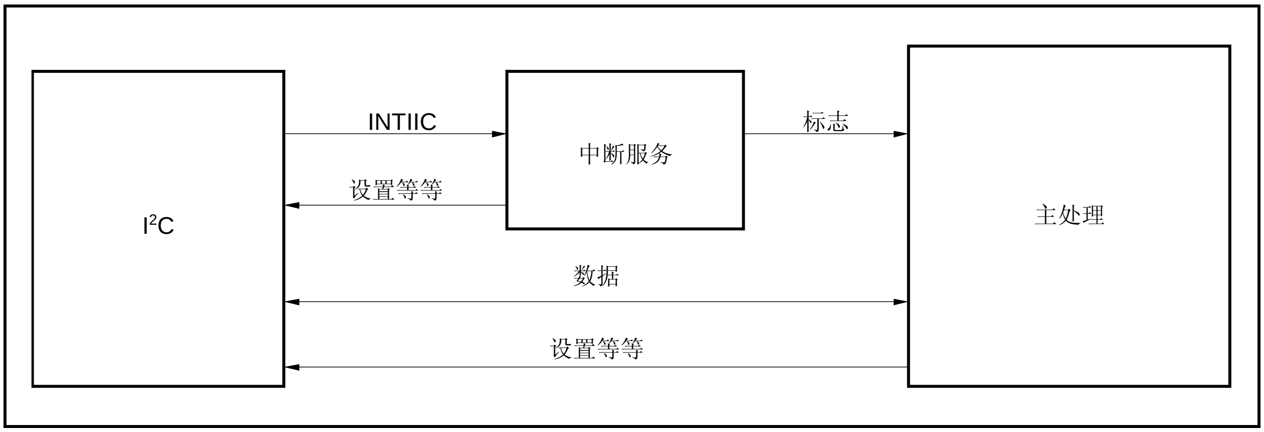
17.16.3 从设备操作

以下是从设备操作的处理过程。

基本而言，从设备的操作都是由事件驱动的，所以，必需通过 INTIIC 中断进行处理（处理需要操作状态的重大改变，比如，通信过程中检测到停止条件）。

以下描述假定数据通信不支持扩展码，还假设只有在状态改变处理时才执行 INTIIC 中断服务，并且在主程序处理期间执行实际的数据通信。

图 17-18. 从设备操作期间的软件概况



所以，系统准备了以下 3 个标志，以便将其传送给主程序，从而执行数据传输处理，而不使用 INTIIC 信号。

（1） 通信模式标志

该标志表示以下通信状态：

清除模式：　　　未进行数据通信

通信模式：　　　正在进行数据通信（检测到有效地址，检测到停止条件，未检测到来自主设备的 $\overline{\text{ACK}}$ 以及地址不一致）

（2） 就绪标志

该标志表示使能数据通信。这和正常数据传输期间发生 INTIIC 中断的状态相同。该标志在中断处理模块内被置位，在主程序模块中清除。第一个数据发送的就绪标志并不在中断处理模块内置位，所以，第一个数据发送后不需要清除处理（地址一致被视为下一个数据的请求）。

（3） 通信方向标志

该标志表示数据通信的方向，和 IICS0.TRC0 位的值相同。

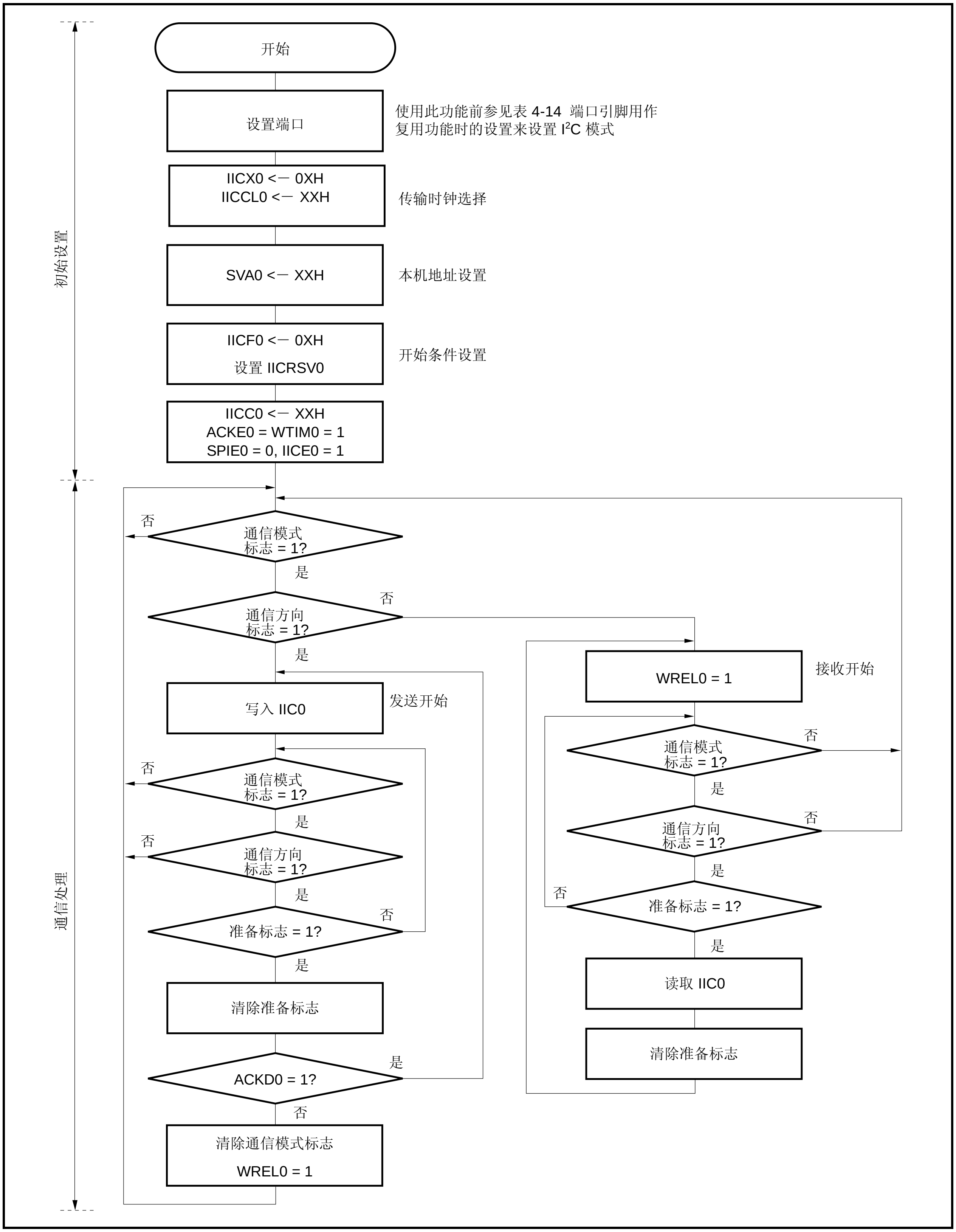
下面展示了从设备操作期间主程序模块的操作。

启动I^{2}C并且等待通信使能状态。当通信使能后，使用通信模式标志和就绪标志执行传输（停止条件和开始条件的处理由中断处理执行，条件通过标志位确认）。

对于发送操作，重复发送直到主设备停止返回 $\overline{\text{ACK}}$ 信号，当主设备停止返回 $\overline{\text{ACK}}$ 信号时，传输结束。

对于接收操作，接收指定数量的数据，且在传输结束后，不为下一个数据立即返回 $\overline{\text{ACK}}$ 信号。此后，主设备产生停止条件或重新开始条件。这将导致退出通信。

图 17-19. 从设备操作流程图（1）

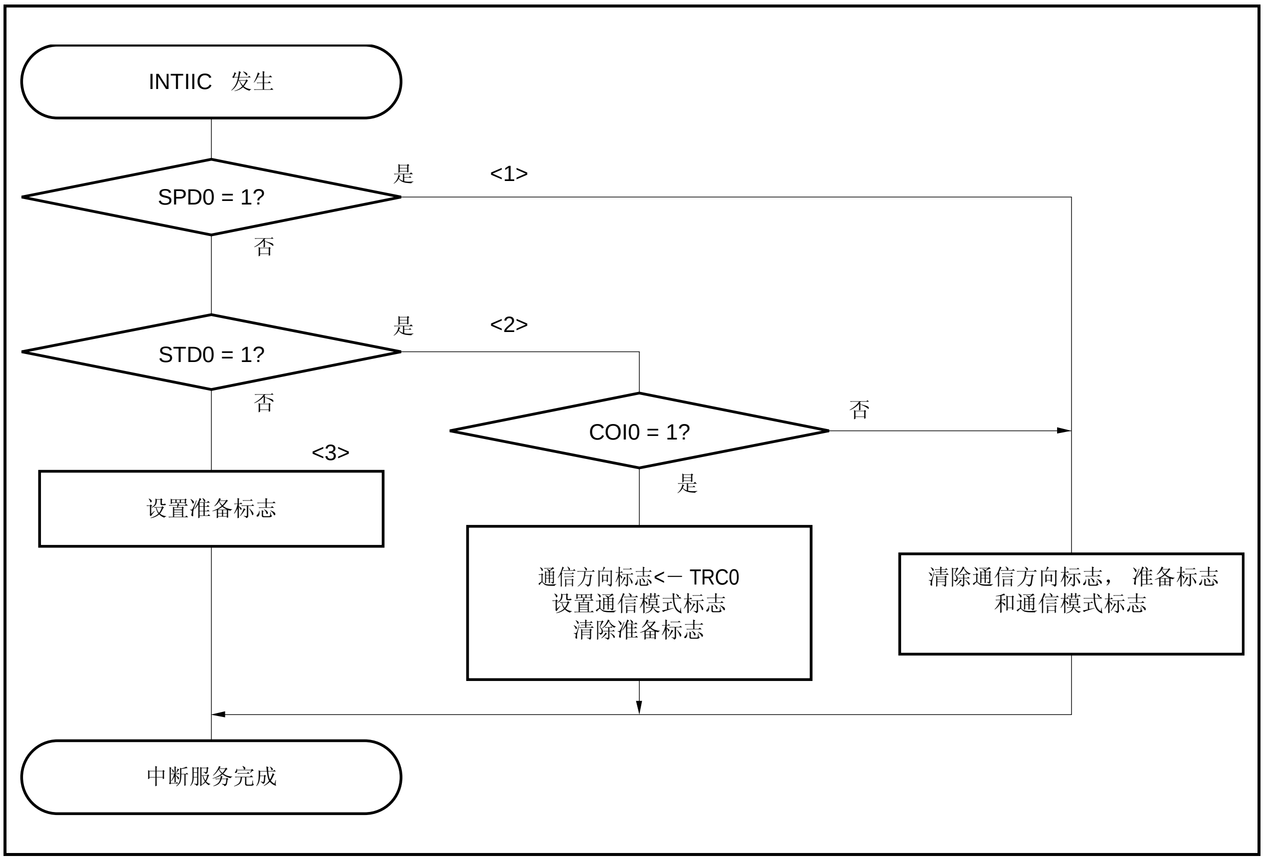


下面展示了从设备使用 INTIIC 中断进行处理的一个示例（这里假定不使用扩展码）。在 INTIIC 中断期间，状态得以确认并且执行下列步骤。

<1> 当检测到停止条件时，通信终止。
<2> 当检测到开始条件后，进行地址确认。如果地址不一致，则通信终止。如果地址一致，设置通信模式并且释放等待状态，从中断中返回操作处理（就绪标志被清除）。
<3> 对于数据的发送/接收过程，当就绪标志置位时，I²C总线保持为等待状态，从中断中返回操作处理。

备注 上述<1> 至<3> 步骤对应**图 17-20 从设备操作流程（2）**的<1> 至<3>步骤。

图 17-20. 从设备操作流程 （2）



17.17 数据通信的时序

当使用I^2C 总线模式时，主设备通过串行总线产生一个地址，在多个从设备中选择其中一个作为自己的通信伙伴。

输出从设备地址以后，主设备发送 IICS0.TRC0 位指定数据传输方向，然后，开始与从设备进行串行通信。

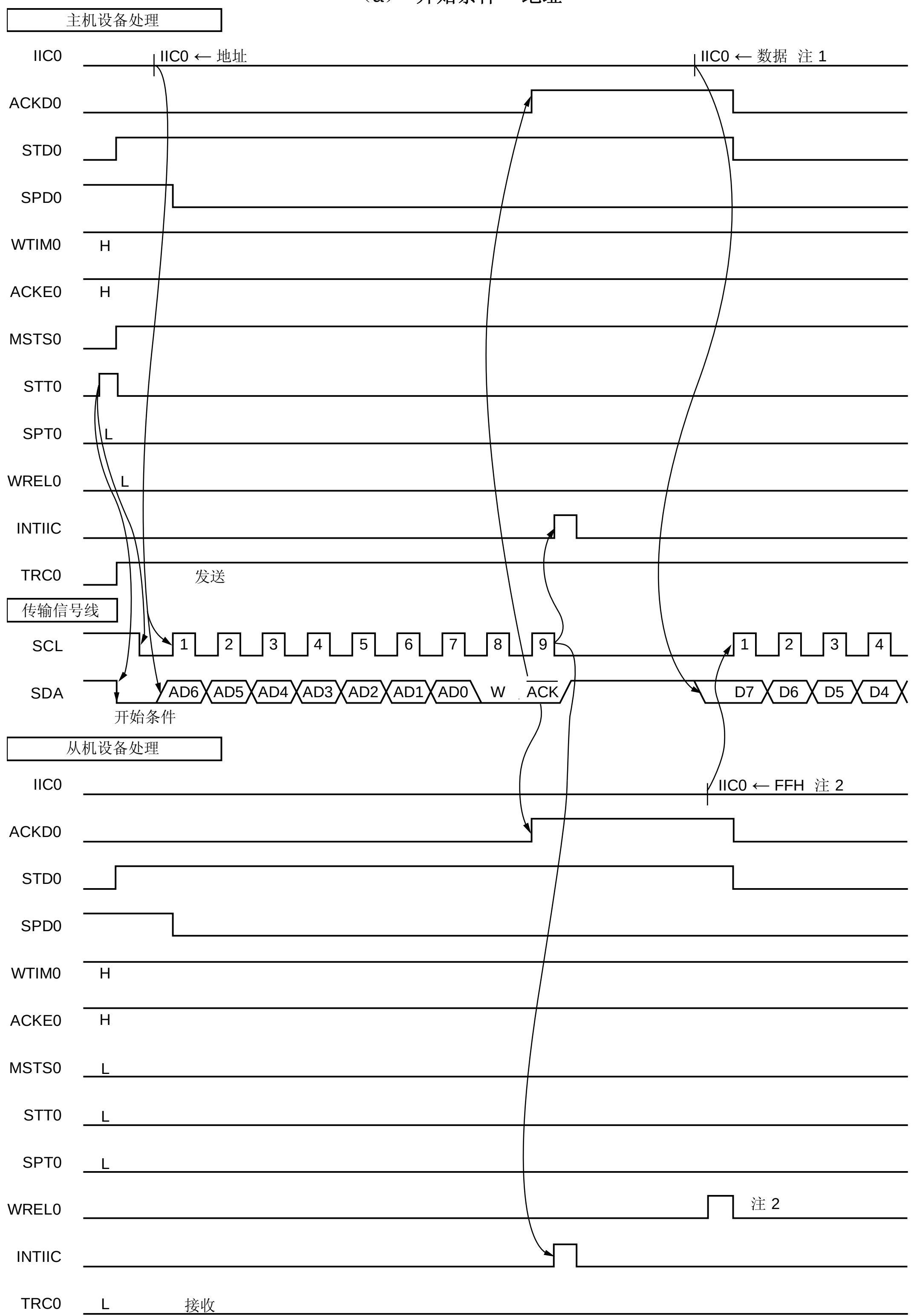
IIC0 寄存器的移位操作和串行时钟引脚（SCL 引脚）信号的下降沿同步。发送数据传送到 SO 锁存器并经由 SDA 引脚输出（MSB 先行）。

由 SDA 引脚输入的数据在 SCL 引脚信号的上升沿被 IIC0 寄存器捕获。

数据通信时序如下所示。

图 **17-21.** 主设备向从设备传输示例
（当主设备和从设备都选择 **9** 时钟等待时）（**1/3**）

（a） 开始条件 ~ 地址



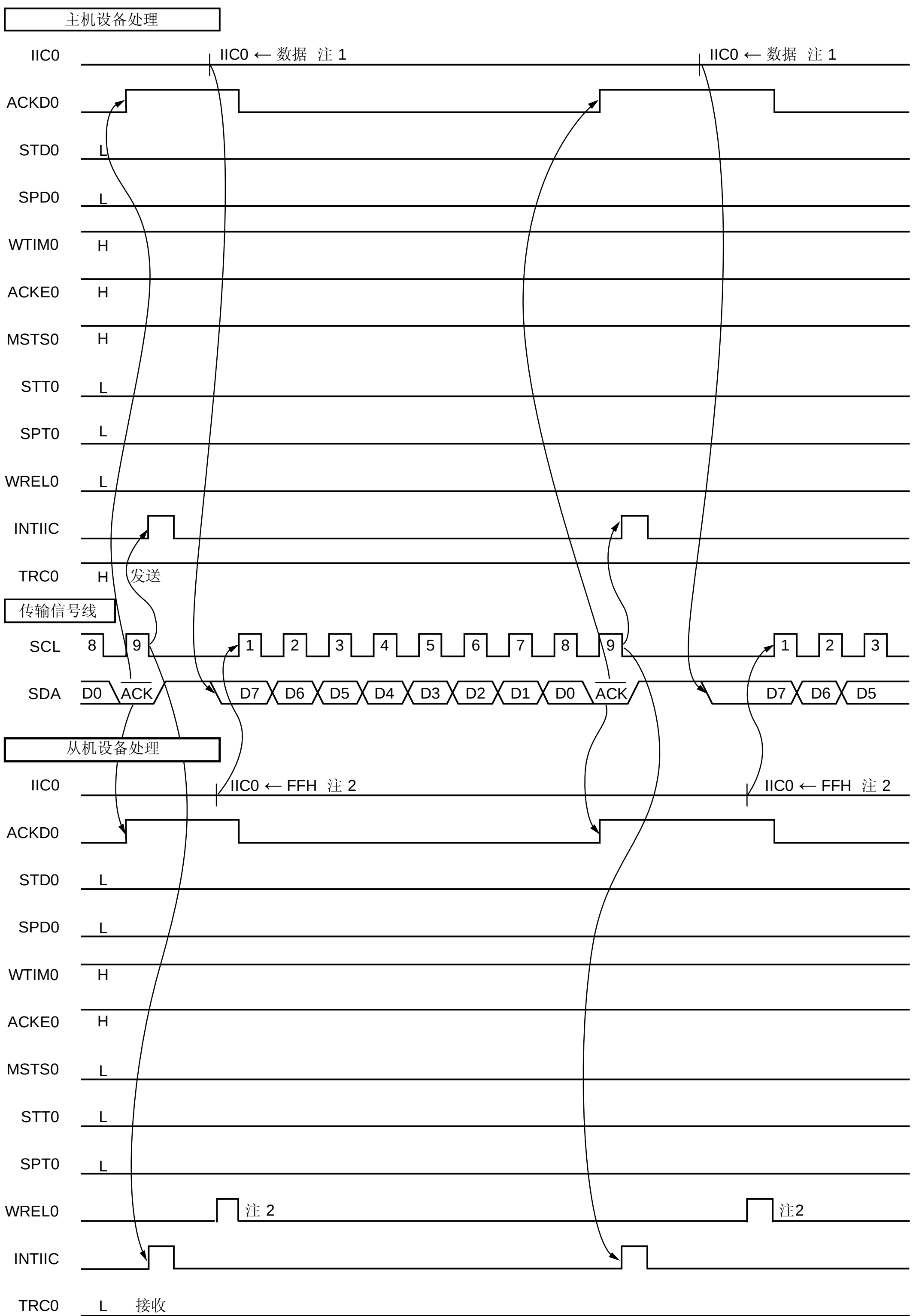
注 **1.** 在主设备发送期间，通过向 IIC0 写入数据，而不是通过设置 WREL0，来取消等待。

2. 要取消从设备等待，向 IIC0 写入 FFH ，或将 WREL0 置位。

图 **17-21.** 主设备向从设备传输示例

（当主设备和从设备都选择 **9** 时钟等待时）（**2/3**）

（**b**） 数据



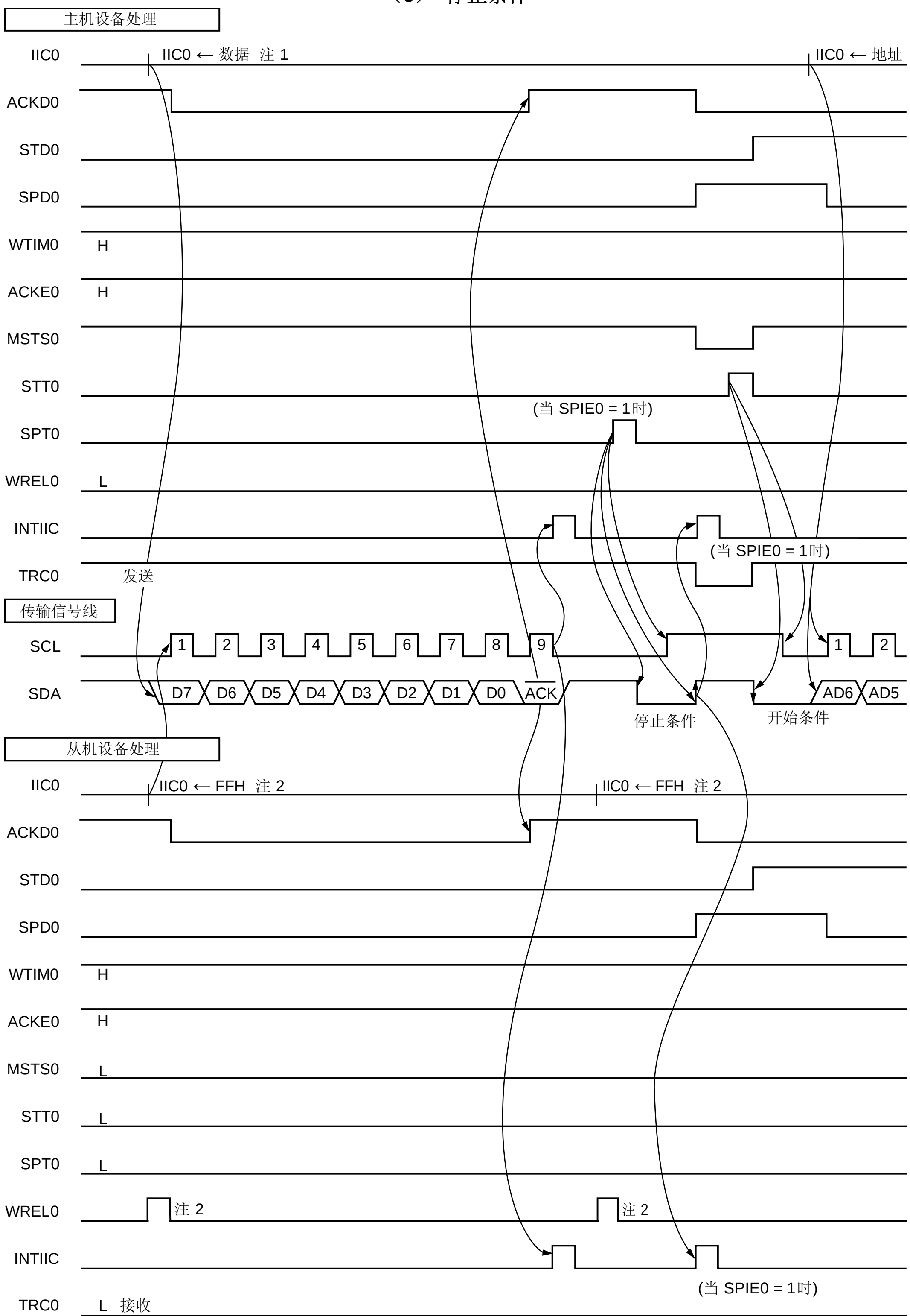
注 **1.** 在主设备发送期间，通过向 IIC0 写入数据，而不是通过设置 WREL0，来取消等待。

2. 要取消从设备等待，向 IIC0 写入 FFH ，或将 WREL0 置位。

图 17-21. 主设备向从设备传输示例

（当主设备和从设备都选择 9 时钟等待时）（3/3）

（c） 停止条件

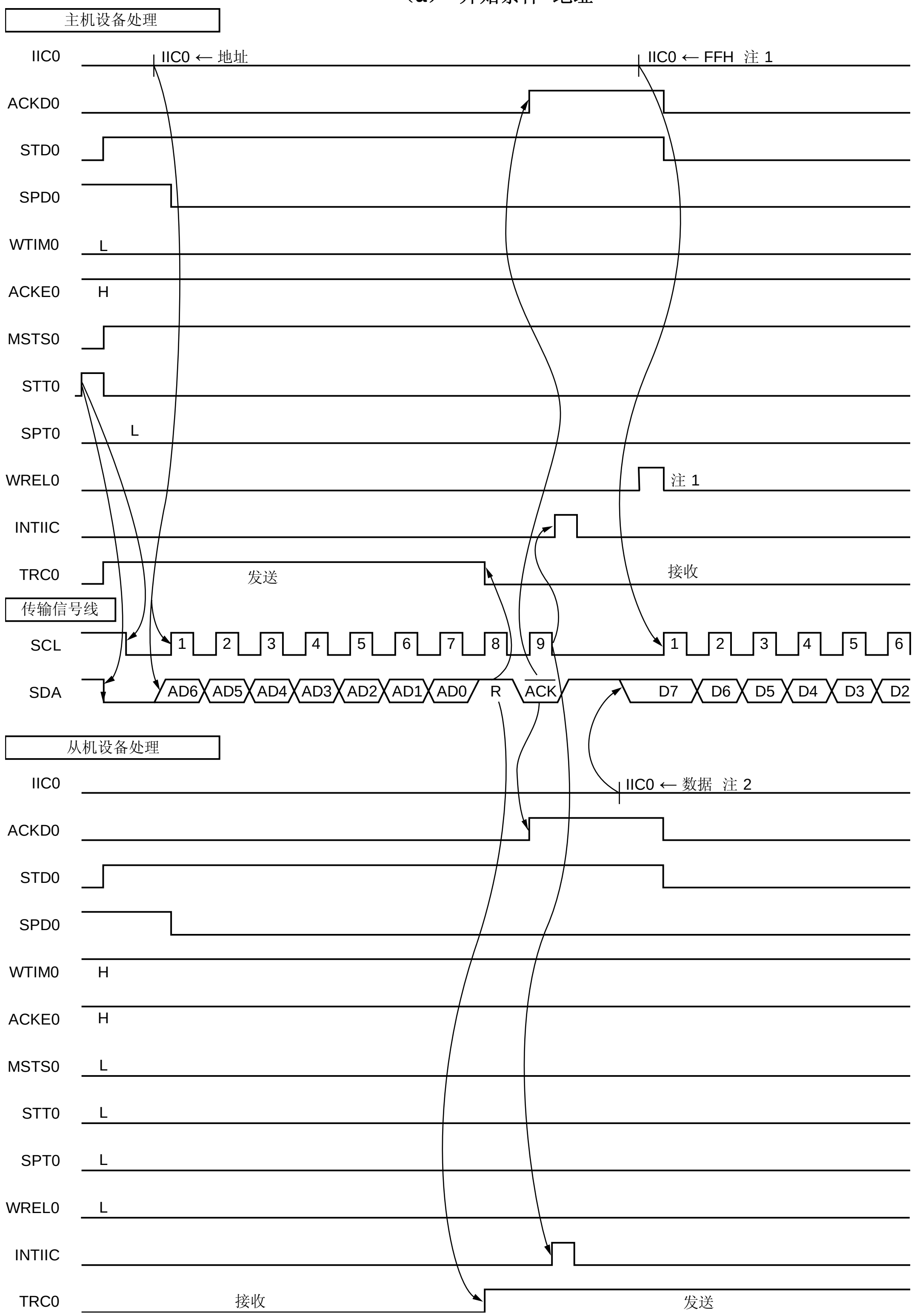


注 1. 在主设备发送期间，通过向 IIC0 写入数据，而不是通过设置 WREL0，来取消等待。

2. 要取消从设备等待，向 IIC0 写入 FFH ，或将 WREL0 置位。

图 **17-22.** 从设备向主设备传输示例
（选择主设备 **8** 时钟等待，从设备 **9** 时钟等待）（**1/3**）

（**a**） 开始条件 地址

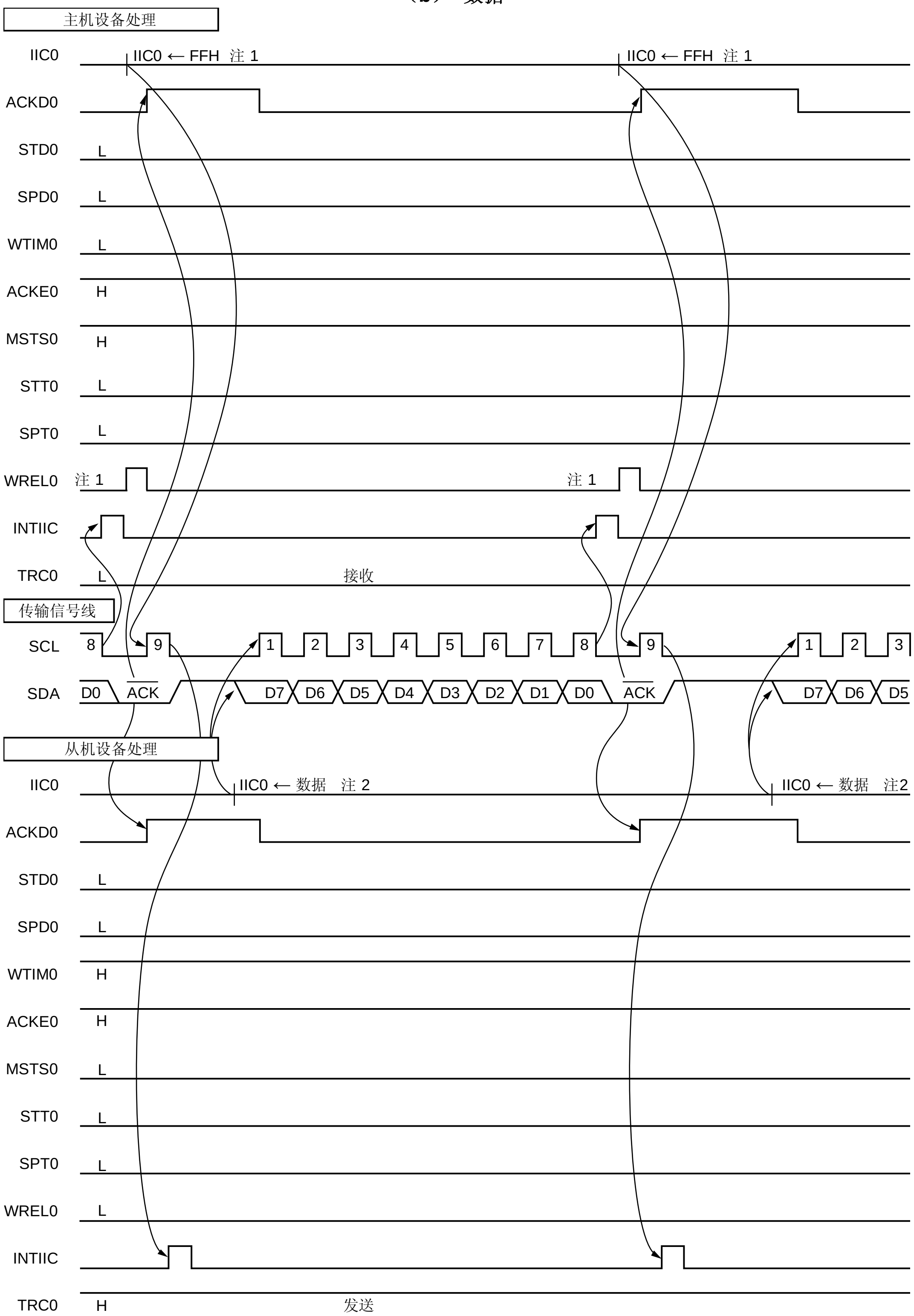


注 **1.** 要取消从设备等待，向 IIC0 写入 FFH，或将 WREL0 置位。
2. 在主设备发送期间，通过向 IIC0 写入数据，而不是通过设置 WREL0，来取消等待。

图 17-22. 从设备向主设备传输示例

（选择主设备 8 时钟等待，从设备 9 时钟等待）（2/3）

（b） 数据



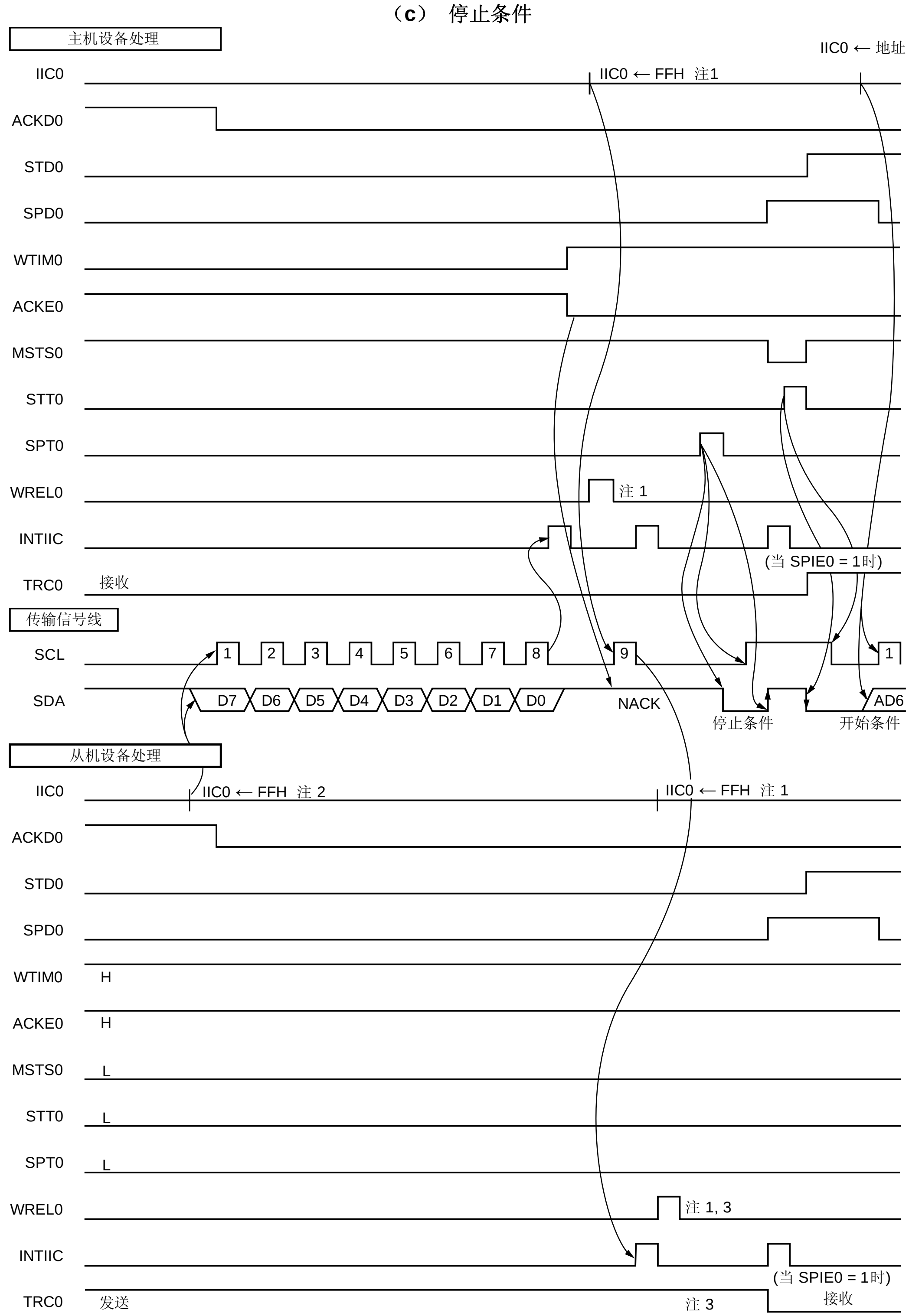
注 1. 要取消从设备等待，向 IIC0 写入 FFH，或将 WREL0 置位。

2. 在主设备发送期间，通过向 IIC0 写入数据，而不是通过设置 WREL0，来取消等待。

图 **17-22.** 从设备向主设备传输示例

（选择主设备 **8** 时钟等待，从设备 **9** 时钟等待）（**3/3**）

（**c**） 停止条件



注 **1.** 要取消从设备等待，向 IIC0 写入 FFH ，或将 WREL0 置位。

2. 在主设备发送期间，通过向 IIC0 写入数据，而不是通过设置 WREL0，来取消等待。

3. 在从设置发送期间，如果通过设置 **WREL0** 取消了等待，清除 **TRC0** 。

第十八章　总线控制功能

<R> μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A 提供外部总线接口功能，通过该接口可以连接外部存储器设备，诸如 ROM、RAM 和 I/O 设备等。V850E/IG3 的 GF 版本μPD70F3453GC-8EA-A，和 V850E/IF3 不提供外部总线接口功能。

<R> 本章将 μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A 作为目标微控制器进行描述。

18.1 特性

- 16 位/8 位数据总线功能
- 2 空间片选功能
- 等待功能：
  - 对每个存储器块最高可插入 7 个等待状态的可编程等待功能
  - 地址建立等待和地址保持等待插入功能，通过该功能可以为每个存储器块插入一个等待状态
  - 使用等待 引脚的外部等待功能
- 空闲状态插入功能
- 通过总线控制/端口复用功能引脚使能外部设备连接
- 分离总线模式(8 位地址总线，8 位/16 位数据总线)
- 复用总线模式(16 位地址总线，8 位/16 位数据总线)
- 支持小尾存储
- 外部总线时钟频率 32 MHz/16 MHz 可选功能

18.2 总线控制引脚

下列引脚用于连接外部设备。

(1) 分离总线模式

总线控制引脚 (复用功能模式的功能)	端口模式下的功能	切换端口模式/复用功能的寄存器
数据总线 (AD0 至 AD15)	PDL0 至 PDL9, PDL10 至 PDL15	PMCDL
地址总线 (A0 至 A7)	P10 至 P17	PMC1
片选 ($\overline{\text{CS0}}$, $\overline{\text{CS1}}$)	P34, P32	PMC3
读取/写入控制(ASTB, $\overline{\text{WR0}}$, $\overline{\text{WR1}}$, $\overline{\text{RD}}$)	P37, P46, P45, P47	PMC3, PMC4
外部等待控制 ($\overline{\text{等待}}$)	P44	PMC4
外部总线时钟输出 (CLKOUT)	P07	PMC0

(2) 复用总线模式

总线控制引脚 (复用功能模式的功能)	端口模式下的功能	切换端口模式/复用功能的寄存器
地址数据总线 (AD0 至 AD15)	PDL0 至 PDL9, PDL10 至 PDL15	PMCDL
片选 ($\overline{\text{CS0}}$, $\overline{\text{CS1}}$)	P34, P32	PMC3
读取/写入控制(ASTB, $\overline{\text{WR0}}$, $\overline{\text{WR1}}$, $\overline{\text{RD}}$)	P37, P46, P45, P47	PMC3, PMC4
外部等待控制 ($\overline{\text{等待}}$)	P44	PMC4
外部总线时钟输出 (CLKOUT)	P07	PMC0

18.2.1 访问内部 ROM，内部 RAM 和片上外围 I/O 期间的引脚状态

访问片内 ROM、片内 RAM 和片上外围 I/O 时，各个引脚的状态如下：

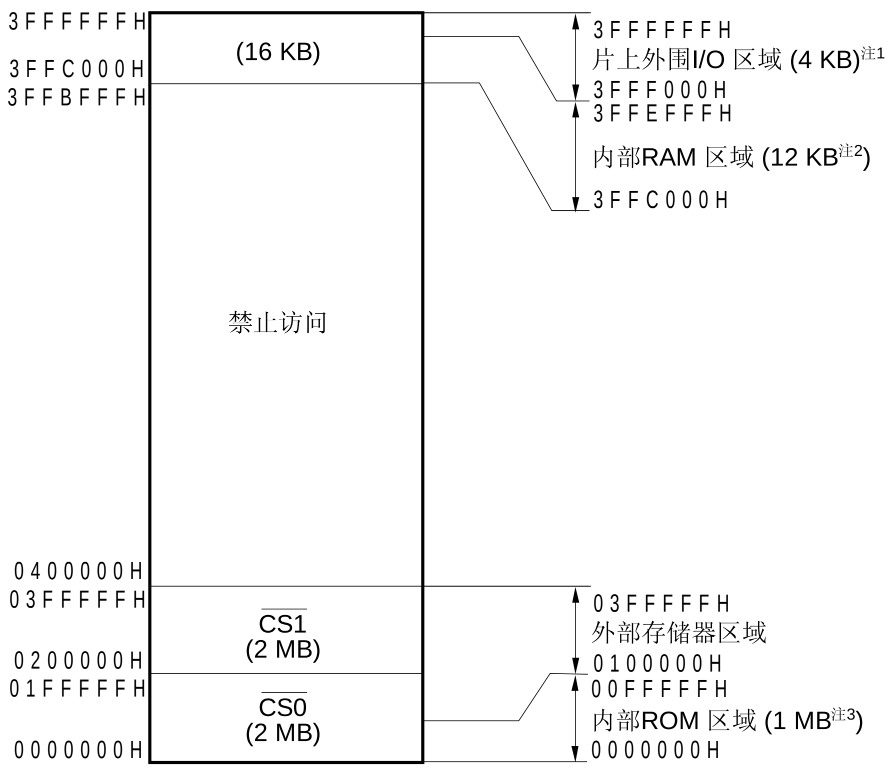
表 18-1. 访问内部 ROM，内部 RAM 和片上外围 I/O 时的引脚状态列表

访问目标	内部 ROM	内部 RAM	片上外围 I/O
地址总线	不确定	不确定	**注 1**
数据总线	高阻	高阻	高阻
外部总线控制信号	无效[注 2]	无效[注 2]	无效[注 2]

注　**1.** 访问片上外围 I/O 时，访问的片上外围 I/O 的地址也输出到外部地址总线。
　　2. 在该操作期间，$\overline{\text{等待}}$ 引脚不输入任何信号。

18.3 存储块功能

64 MB 存储空间被划分为低 2 MB 单元的存储器块。对每一个块的可编程等待功能和总线周期操作模式都可以独立控制。



注　**1.** 地址 3FFF000H 至 3FFFFFFH 禁止访问。要访问片上外围 I/O，指定地址 FFFF000H 至 FFFFFFFH。

2. μPD70F3453：　8 KB
μPD70F3454：　12 KB

3. μPD70F3453：　128 KB
μPD70F3454：　256 KB

18.3.1 片选控制功能

64 MB 地址空间(线性)中，低地址的 4 MB (0000000H 至 03FFFFFH) 具有两个片选功能，$\overline{\text{CS0}}$ 和 $\overline{\text{CS1}}$。该区域通过 $\overline{\text{CS0}}$ 和 $\overline{\text{CS1}}$ 固定选择。

使用片选控制功能将存储器区域划分为存储器块，可以高效使用存储器区域。存储器块的分配描述如下。

片选引脚	区域
$\overline{\text{CS0}}$	0000000H 至 01FFFFFH (2 MB)
$\overline{\text{CS1}}$	0200000H 至 03FFFFFH (2 MB)

18.4 总线周期类型控制功能

<R> 在 μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A 中，SRAM、外部 ROM 和外部 I/O 可以直接连接。

(1) 总线周期类型配置寄存器 0 (BCT0)

该寄存器可以按照 16 位宽度进行读取或写入。

系统复位后，该寄存器被设置为 CCCCH。

注意事项 **1.** 在 **BCT0** 寄存器的初始设置完成之前,不要访问外部存储器区域。但是，已经完成初始化设置的外部存储器区域可以被访问。

2. 在禁止操作 **CSn** 空间的情况下， 为 **CSn** 空间的各个寄存器设置的内容无效。

复位后：CCCCH　　R/W　地址：FFFFF480H

	15	14	13	12	11	10	9	8
BCT0	1	1	0	0	1	1	0	0
$\overline{CSn}$ 信号								

	7	6	5	4	3	2	1	0
	ME1	1	0	0	ME0	1	0	0
$\overline{CSn}$ 信号	$\overline{CS1}$				$\overline{CS0}$			

MEn	各个CSn空间的存储器控制器操作使能 (n = 0, 1)
0	操作禁止
1	操作使能

注意事项 请确保将第 **0, 1, 4, 5, 8, 9, 12** 位和第 **13** 位清除为 **“ 0”**，并将第 **2, 6, 10, 11, 14** 位和第 **15** 位设置为 **“1”**。如果它们被设置为其他值，操作无法保证。

18.5 总线访问

18.5.1 访问时钟的数量

访问各个资源所需的基准时钟的数量(最小值)如下。

资源(总线宽度) \ 总线周期配置		取指 (正常访问)	取指 (跳转)	操作数数据访问
内部 ROM (32 位)		1	2	5
内部 RAM (32 位)		1[注 1]	1[注 1]	1
片上外围 I/O (16 位)		–	–	3[注 2]
外部存储器 (16 位)	分离总线模式	3 + n	3 + n	3 + n
	复用总线模式	3 + n	3 + n	3 + n

注　**1.** 如果和数据访问有冲突，则该值为 2。
　　2. 取决于 VSWC 寄存器的设置值。

备注　**1.** 单元：时钟/访问
　　　2. n：插入等待状态的数量

18.5.2 总线宽度功能

总线宽度功能用于控制各个 CS 空间的数据总线宽度。数据总线宽度通过 BSC 寄存器指定。

(1) 总线宽度配置寄存器 (BSC)

该寄存器按 16 位宽度进行读取或写入。

系统复位后，该寄存器被设置为 5555H。

注意事项　复位后写入 BSC 寄存器，然后不要改变其设置值。同样，不要访问外部存储器区域，直到 BSC 寄存器的初始设置完成。但是，已经完成初始化设置的外部存储器区域可以被访问。

复位后：5555H　　R/W　地址：　FFFFF066H

	15	14	13	12	11	10	9	8
BSC	0	1	0	1	0	1	0	1
$\overline{\text{CSn}}$ 信号								

	7	6	5	4	3	2	1	0
	0	1	0	1	0	BS10	0	BS00
$\overline{\text{CSn}}$ 信号						$\overline{\text{CS1}}$		$\overline{\text{CS0}}$

BSn0	各个CSn空间的数据总线宽度规范 (n = 0, 1)
0	8 位
1	16 位

注意事项　请确保将第 1, 3, 5, 7, 9, 11, 13 位和第 15 位清除为 " 0"，并将第 4, 6, 8, 10, 12 位和第 14 位设置为 "1"。如果它们被设置为其他值，操作无法保证。

18.5.3 字节存储次序功能

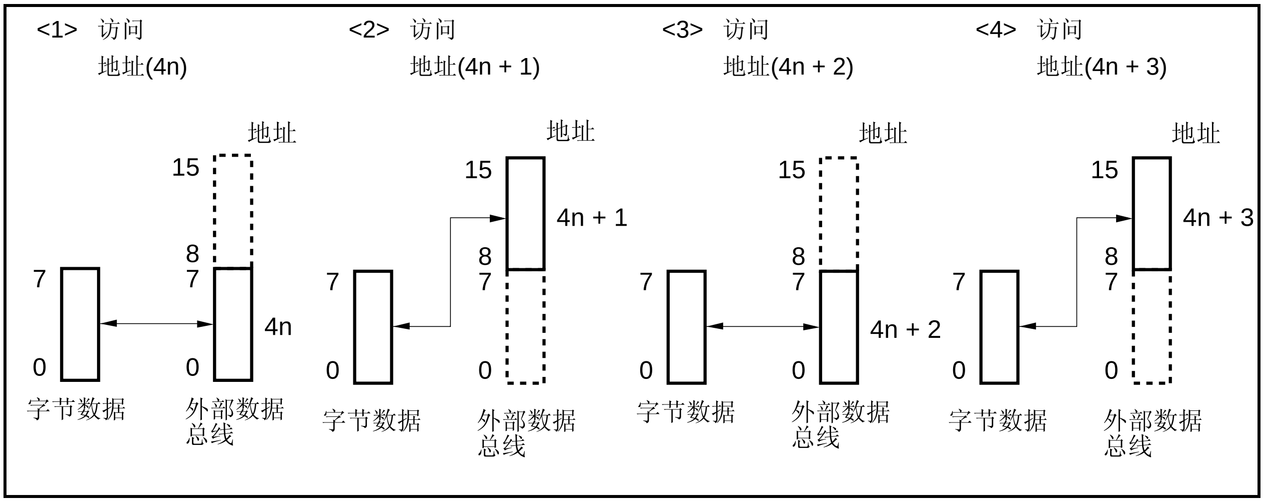
<R> The μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A 支持小尾存储。

18.5.4 总线宽度

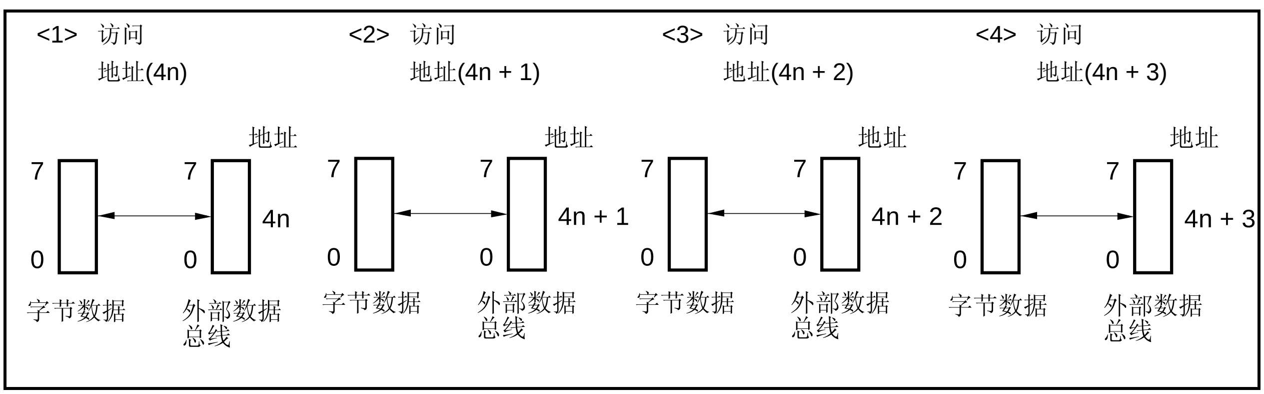
<R> μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A 按照 8 位、16 位或 32 位宽度访问片上外围 I/O 和外部存储器，各种类型访问时的操作描述如下。所有数据访问都从低位开始。

(1) 字节访问(8 位)

(a) 16 位数据总线宽度

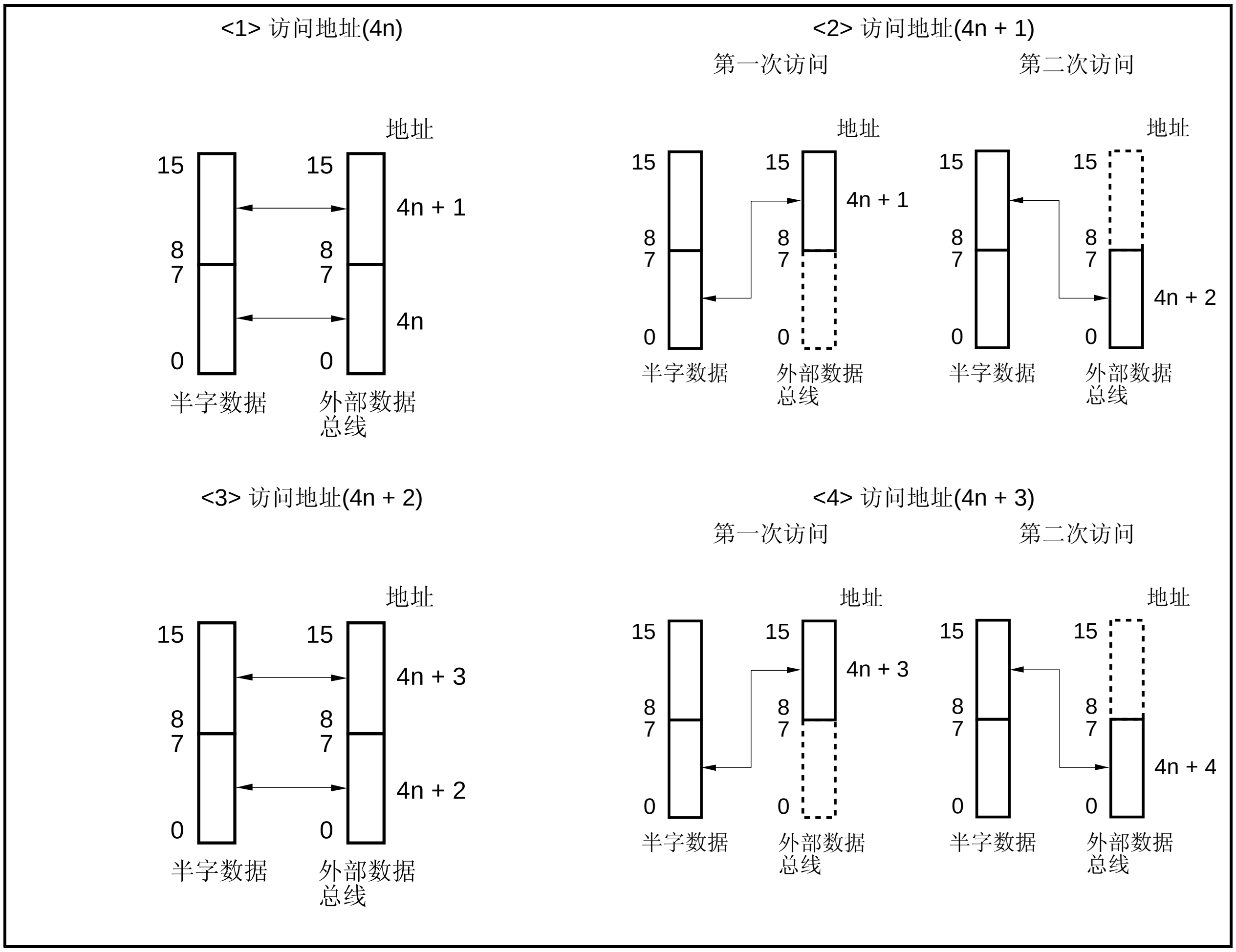


(b) 8 位数据总线宽度

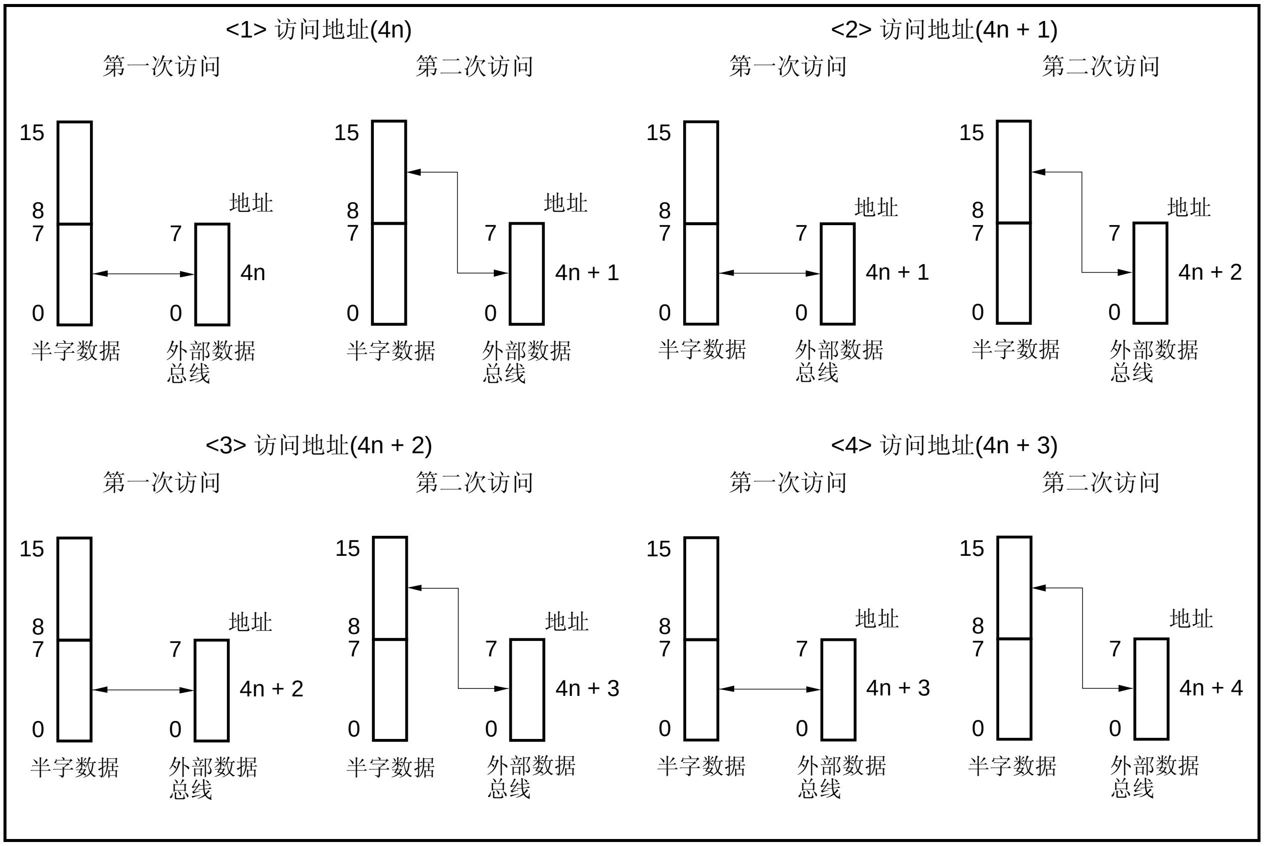


(2) 半字访问**(16** 位**)**

(a) 16 位数据总线宽度

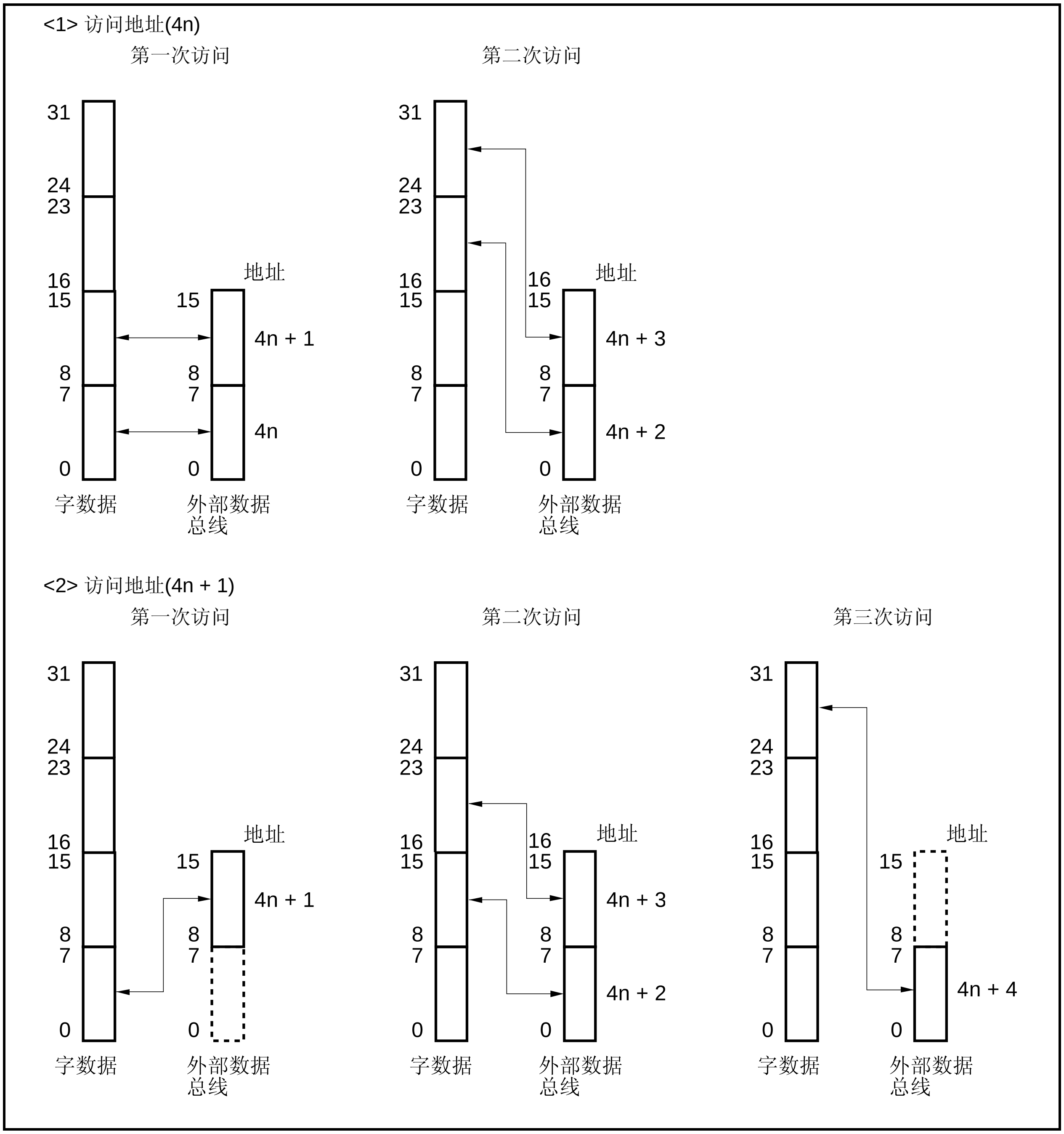


(b) 8 位数据总线宽度

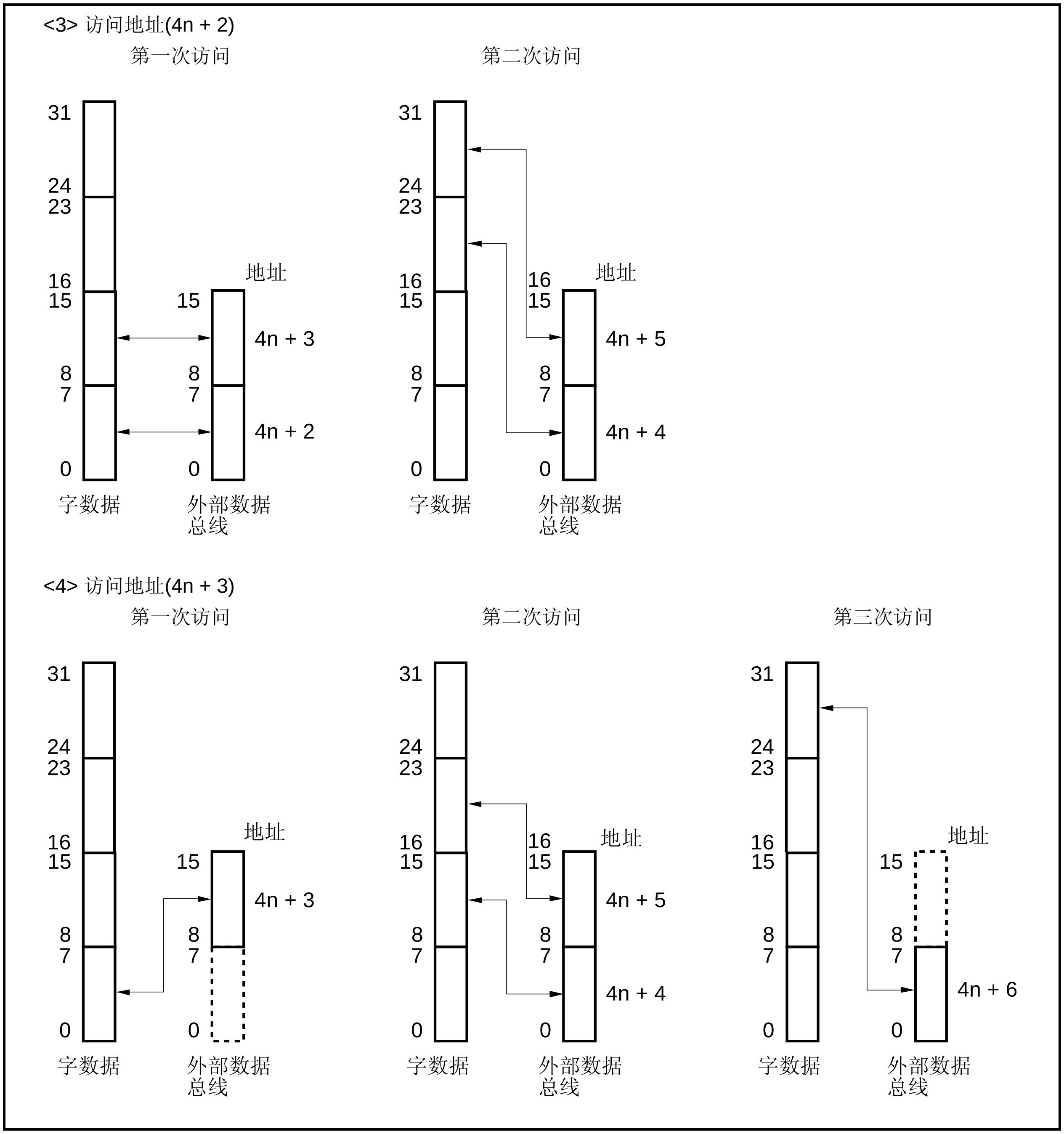

<1> 访问地址(4n)
第一次访问
第二次访问
15
8
7
0
半字数据
地址
7
4n
0
外部数据
总线
4n + 1
<2> 访问地址(4n + 1)
4n + 2
<3> 访问地址(4n + 2)
4n + 3
<4> 访问地址(4n + 3)
4n + 4

(3) 字访问(32 位)

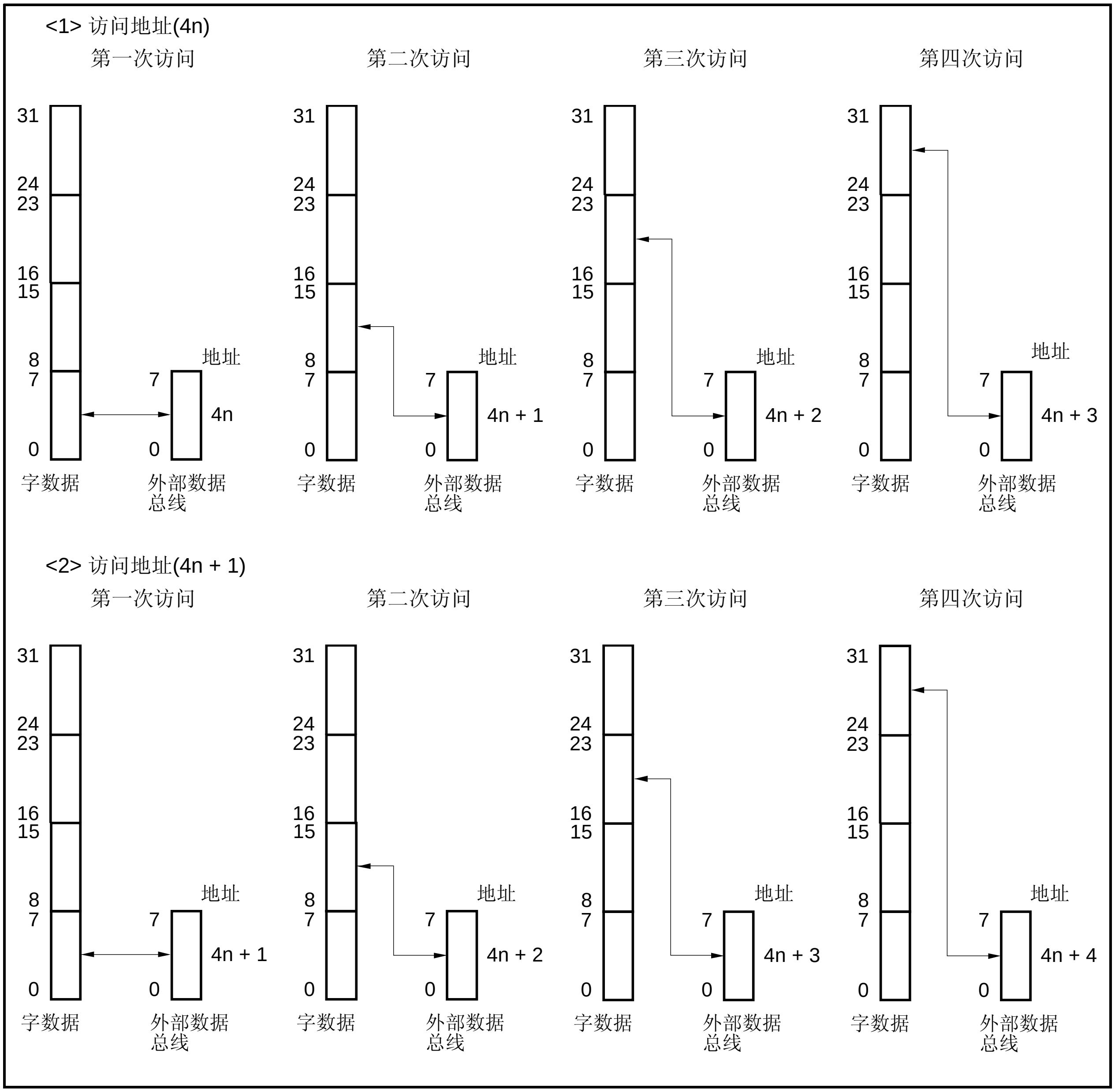
(a) 16 位数据总线宽度(1/2)



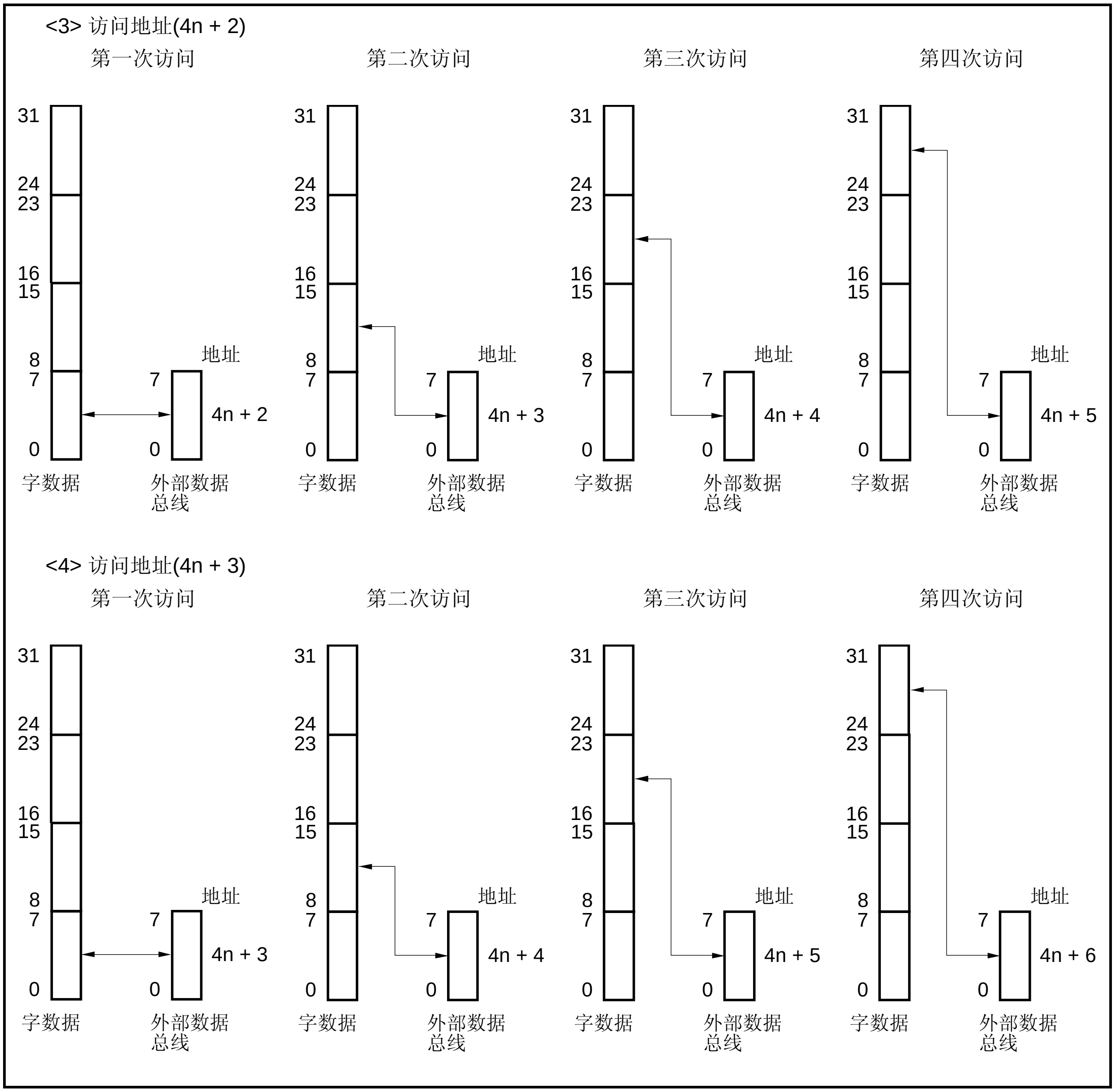
(a)　16 位数据总线宽度 (2/2)



(b) 8 位数据总线宽度(1/2)



(b) 8 位数据总线宽度(2/2)



18.6 等待功能

18.6.1 可编程等待功能

(1) 数据等待控制寄存器 0 (DWC0)

为了和低速存储器或者和I/O端口实现连接，访问每个CS空间时，都可以在启动总线周期[注]内插入多达 7 个数据等待状态。

等待状态的数量可以使用 DWC0 寄存器编程指定。在系统复位后，所有存储块都插入 7 个数据等待状态。

DWC0 寄存器可以按 16 位宽度进行读取或写入。

系统复位后，该寄存器被置为 7777H。

注　SRAM 读取/写入周期

注意事项 **1.** 内部 **ROM** 和内部 **RAM** 区域无需插入可编程等待，平常不进行等待访问。
片上外围 **I/O** 区域无需插入可编程等待，仅通过各个片上外围功能执行等待控制。

2. 复位后写入 **DWC0** 寄存器，然后不要改变其设置值。同样，不要访问外部存储器区域，直到 **DWC0** 寄存器的初始设置完成。但是，已经完成初始化设置的外部存储器区域可以访问。

复位后：7777H　　R/W　地址：FFFFF484H

	15	14	13	12	11	10	9	8
DWC0	0	1	1	1	0	1	1	1
$\overline{CSn}$ 信号								
	7	6	5	4	3	2	1	0
	0	DW12	DW11	DW10	0	DW02	DW01	DW00
$\overline{CSn}$ 信号		$\overline{CS1}$				$\overline{CS0}$		

DWn2	DWn1	DWn0	各个CSn空间插入的等待状态数量规范 (n = 0, 1)
0	0	0	无插入
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7

注意事项　**请确保将第 3，7，11 位和第 15 位清除为“0”，并将第 8 位至第 10 位和第 12 位至第 14 位设置为“1”。如果它们被设置为其他值，操作无法保证。**

(2) 地址等待控制寄存器 (AWC)

通过 AWC 寄存器，可以设置插入到各个总线周期中的地址建立等待或地址保持等待。在 T1 状态之前插入地址建立等待，在 T1 状态之后插入地址保持等待。

访问各个 CS 空间所需插入的地址建立等待状态和地址保持等待状态可以由 AWC 寄存器设置。

AWC 寄存器按 16 位宽度进行读取或写入。

系统复位后，该寄存器被设置为 FFFFH。

注意事项 **1.** 内部 **ROM**、内部 **RAM** 和片上外围 **I/O** 区域无需插入地址建立等待状态和地址保持等待状态 。

2. 在地址建立等待状态和地址保持等待状态期间，禁止基于 **$\overline{\text{WAIT}}$** 引脚的外部等待功能。

3. 复位后写入 **AWC** 寄存器，然后不要改变其设置值。

复位后：FFFFH　　R/W　地址：FFFFF488H

	15	14	13	12	11	10	9	8
AWC	1	1	1	1	1	1	1	1
$\overline{\text{CSn}}$ 信号								

	7	6	5	4	3	2	1	0
	1	1	1	1	AHW1	ASW1	AHW0	ASW0
$\overline{\text{CSn}}$ 信号					$\overline{\text{CS1}}$		$\overline{\text{CS0}}$	

AHWn	各个CSn空间插入的地址保持等待状态规范 (n = 0, 1)
0	未插入
1	已插入

ASWn	各个CSn空间插入的地址建立等待状态规范 (n = 0, 1)
0	未插入
1	已插入

注意事项 **请确保将第 4 位至第 15 位设置为“1”。如果它们被设置为“0”，操作无法保证。**

18.6.2 外部等待功能

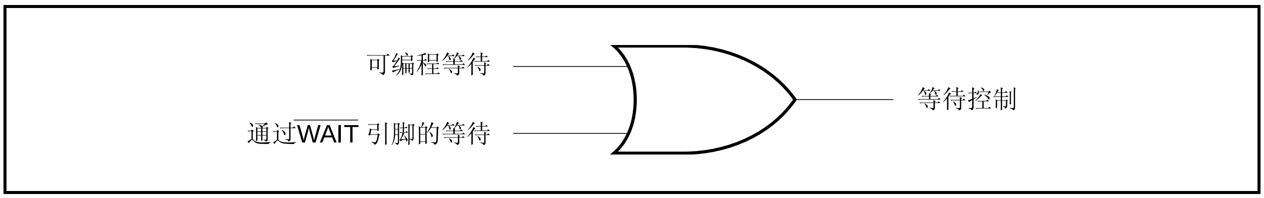
当连接一个极低速的存储器、I/O 或异步系统时，可以通过外部等待引脚（$\overline{\text{WAIT}}$）在总线周期中插入任意数目的等待状态，以实现和外部设备同步。

访问内部 ROM、内部 RAM 的任何区域以及片上外围 I/O 区域时，不能通过外部等待功能来控制，这一点和可编程等待功能一样。

外部 $\overline{\text{WAIT}}$ 信号的 输入可以和外部总线时钟频率不同步。

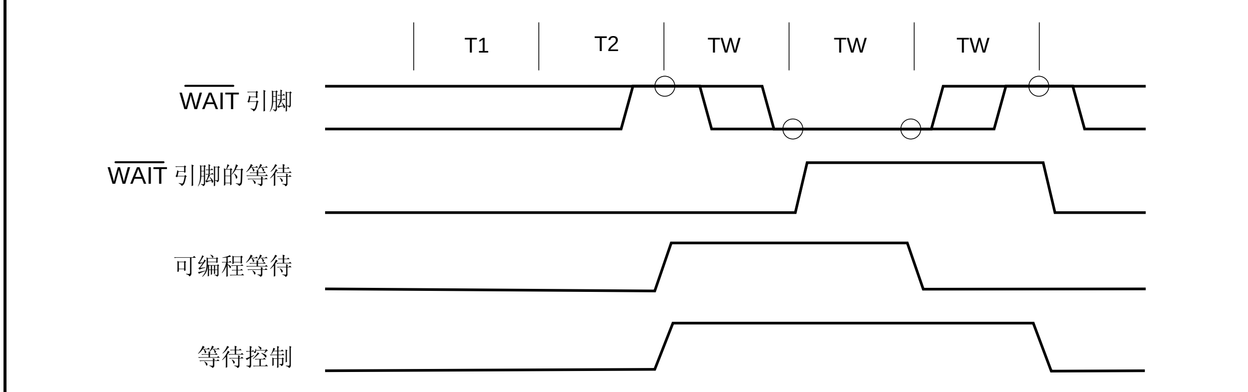
18.6.3 可编程等待和外部等待之间的关系

对可编程等待的设置值指定的等待周期和 $\overline{\text{WAIT}}$ 引脚控制的等待周期执行或(OR)操作，得到的结果是插入的等待周期数量。



例如，如果可编程等待的时序和 $\overline{\text{WAIT}}$ 引脚信号如下所示，则插入到总线访问周期数目为 3 个等待状态。

图 18-1. 等待插入的示例



备注 圆圈 ○ 表示采样时序

18.6.4 等待功能有效的总线周期

<R> 对于μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A，可以指定各个存储器块的等待数量。下面说明了等待功能有效的总线周期和用于设置等待的寄存器。

表 18-2. 等待功能有效的总线周期

总线周期	等待类型	可编程等待设置			$\overline{\text{WAIT}}$ 引脚的等待
		寄存器	位	等待的数量	
SRAM、外部 ROM、外部 I/O 周期	地址建立等待	AWC	ASWn	0, 1	× (无效)
	地址保持等待	AWC	AHWn	0, 1	× (无效)
	数据等待	DWC0	DWn2 至 DWn0	0 至 7	√ (有效)

备注 n = 0, 1

18.7 空闲状态插入功能

在对 SRAM、外部 ROM 或外部 I/O 进行一个读取/写入周期之后，插入空闲状态。

(1) 总线周期控制寄存器 (BCC)

为了方便和低速存储器设备的连接，对各个 CS 空间的存储器进行读取访问时，在 T2 状态之后（如果有一个数据等待状态插入,则在 TW 状态之后）的当前总线周期插入一个空闲状态（TI），可以保证数据输出的浮动延迟时间。在插入一个空闲状态之后，总线周期在 T2 状态（或 TW 状态）之后。

使用总线时钟分频控制寄存器 (DVC)可以在写入访问之后插入一个空闲状态。

可以使用 BCC 寄存器编程指定空闲状态插入的设置。在系统复位后，对所有存储器块都会立刻自动进行空闲状态插入。关于空闲状态插入的时序，参见 **18.8 总线时序**。

该寄存器按 16 位宽度进行读取或写入。

系统复位后，该寄存器被置为 AAAAH。

注意事项 1. 内部 ROM 、内部 RAM、片上外围 I/O 区域无需空闲状态插入。

2. 复位后写入 BCC 寄存器，然后不要改变其设置值。同样，不要访问外部存储器区域，直到 BCC 寄存器的初始设置完成。但是，已经完成初始化设置的外部存储器区域可以访问。

3. 在空闲状态下，片选信号 ($\overline{CSn}$)不会变为有效(n = 0, 1)。

复位后： AAAAH R/W 地址： FFFFF48AH

	15	14	13	12	11	10	9	8
BCC	1	0	1	0	1	0	1	0
$\overline{CSn}$ 信号								

	7	6	5	4	3	2	1	0
	1	0	1	0	BC11	0	BC01	0
$\overline{CSn}$ 信号					$\overline{CS1}$		$\overline{CS0}$	

BCn1	各个CSn 空间插入的空闲状态规范 (n = 0, 1)
0	未插入
1	已插入

在一个读取周期或一个写入周期之后，可为各个CSn空间插入空闲状态。
如果DVC.BCWI 位 = 0 仅可在一个读取周期完成之后插入空闲状态，而在一个写入周期完成之后不能插入。

注意事项 请确保将第 0, 2, 4, 6, 8, 10, 12 位和第 14 位清除为“0”，并将第 5, 7, 9, 11, 13 位和第 15 位设置为“1”。如果它们被设置为其他值，操作无法保证。

(2) 总线时钟分频控制寄存器 (DVC)

DVC 寄存器用于指定在一个写入周期或一个外部总线时钟频率之后插入空闲状态(TI)。
该寄存器按字节进行读取或写入。
系统复位后，该寄存器被置为 81H。

注意事项 **1.** 内部 **ROM** 、内部 **RAM**、片上外围 **I/O** 区域无需空闲状态插入。
2. 复位后写入 **DVC** 寄存器（初始设置），然后不要改变其设置值。同样，不要访问外部存储器区域，直到 **DVC** 寄存器的初始设置完成。
但是，已经完成初始化设置的外部存储器区域可以访问。

复位后：81H　R/W　地址：FFFFF48EH

	7	6	5	4	3	2	1	0
DVC	BCWI	0	0	0	0	0	DVC1	DVC0

BCWI	写入周期结束之后插入的空闲状态规范
0	未插入
1	已插入 (仅当BCC.BCn1 位 = 1)

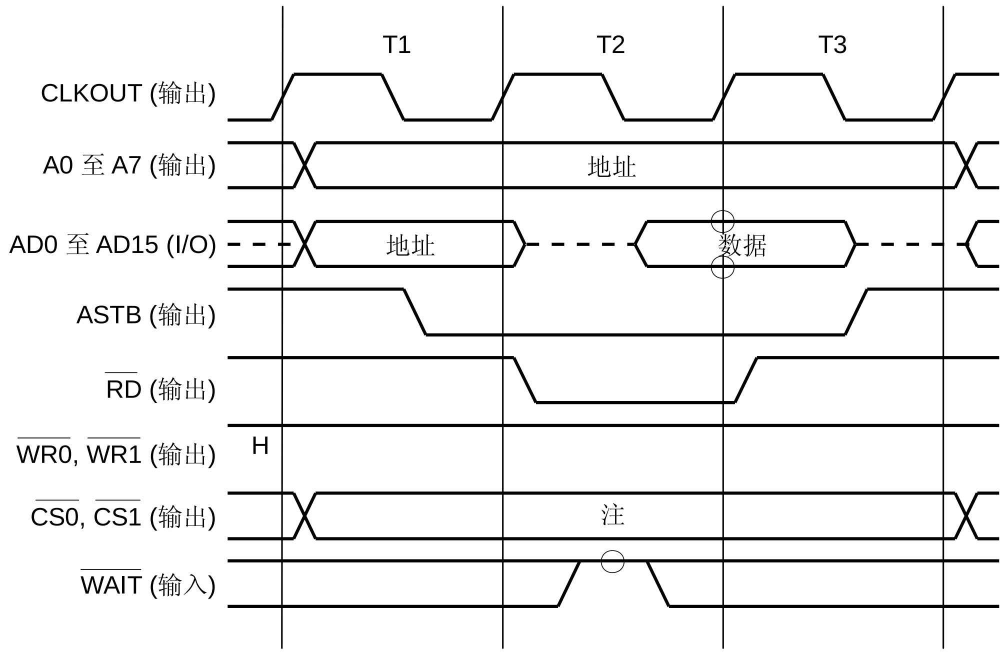
DVC1	DVC0	外部总线时钟频率(f_{BUS})的规范
0	0	f_{CLK}/1[注]
0	1	f_{CLK}/2
1	0	禁止设置
1	1	f_{CLK}/4

注　只有当 $f_{CLK} \leq 32$ MHz 时可以设置。
当 32 MHz < $f_{CLK} \leq 64$ MHz 时，禁止设置。

注意事项 **1.** 请确保在改变 **DVC1** 和 **DVC0** 位的设置之前，将 **CLKOUT** 引脚设置为端口模式。禁止在使用复用功能**(CLKOUT)**时改变 **DVC1** 和 **DVC0** 位的设置。
2. 设置外部总线时钟频率 **(f_{BUS})** 的范围是 **16 MHz ≤ f_{BUS} ≤ 32 MHz.**
3. 请确保将第 **2** 位至第 **6** 位清除为**“0”**，如果它们被设置为**“1”**，操作无法保证。

18.8 总线时序

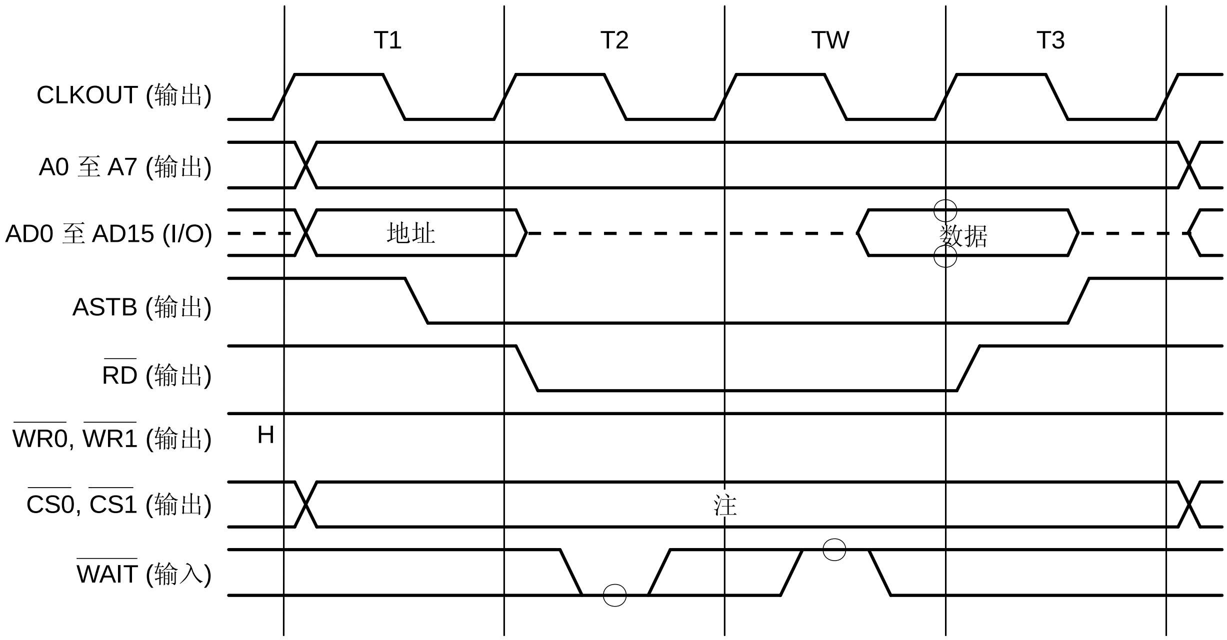
(1) 读取周期 (基本周期)



注　只有可以访问的 CS 空间有效。

备注　**1.** 圆圈 O 表示采样时序。
　　　2. 虚线表示高阻抗状态。

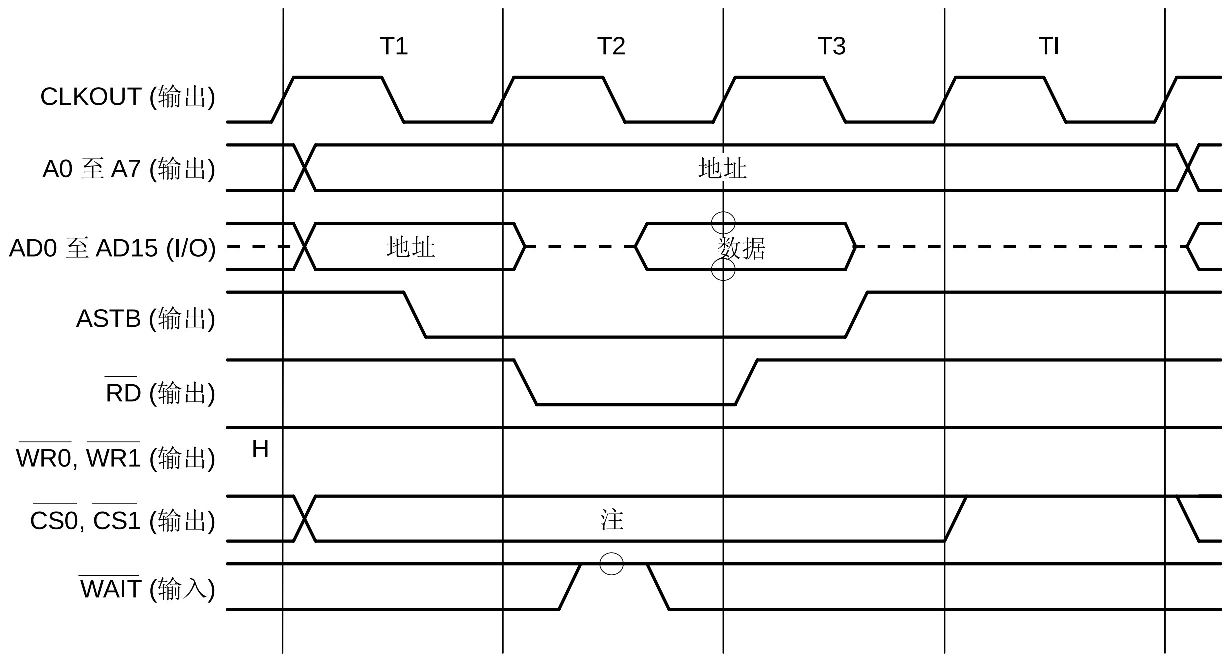
(2) 读取周期 (当数据等待状态 (1 等待)插入时)



注　　只有可以访问的 CS 空间有效。

备注　　**1.** 圆圈 O 表示采样时序。
　　　　2. 虚线表示高阻抗状态。

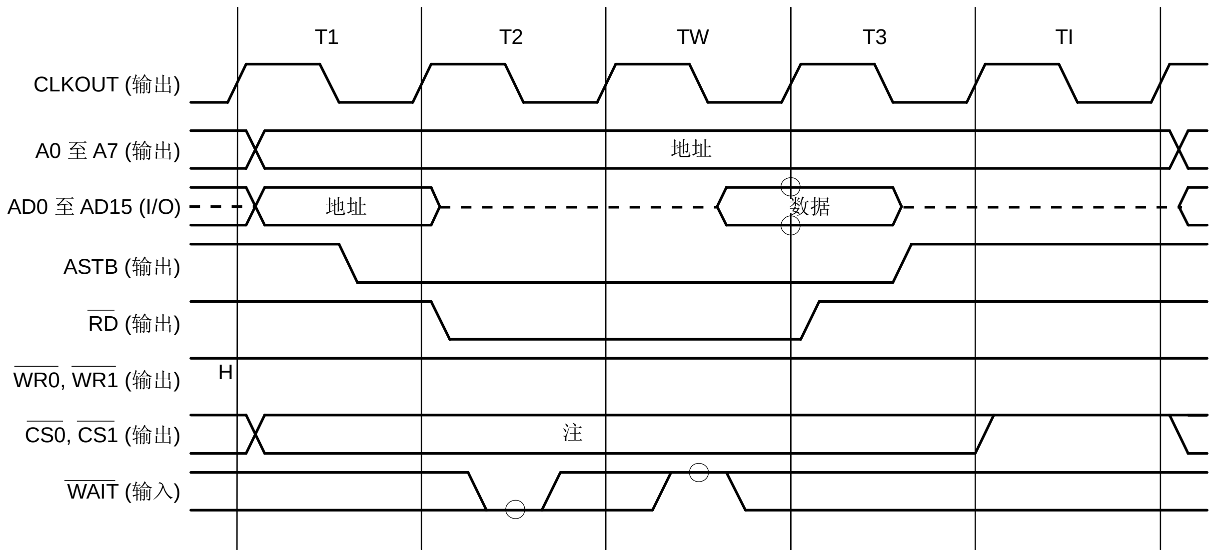
(3)　读取周期 (当空闲状态插入时)



注　　只有可以访问的 CS 空间有效。

备注　　**1.**　圆圈 ○ 表示采样时序。
　　　　2.　虚线表示高阻抗状态。

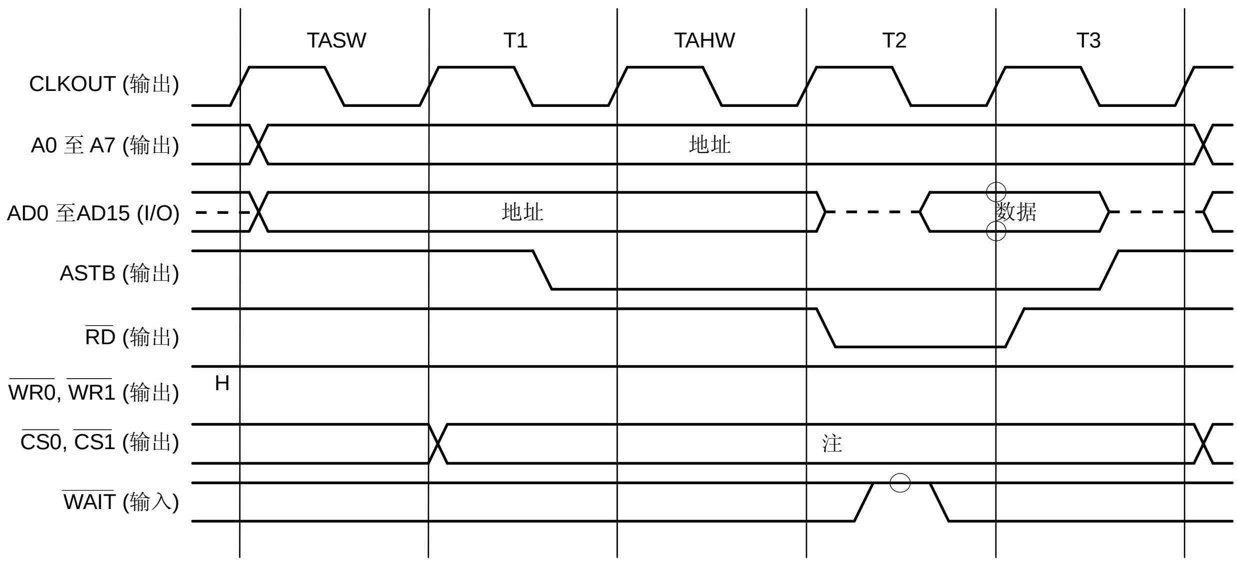
(4)　读取周期 (当数据等待状态 (1 等待)，空闲状态插入时)



注　　只有可以访问的 CS 空间有效。

备注　　**1.**　圆圈 ○ 表示采样时序。
　　　　2.　虚线表示高阻抗状态。

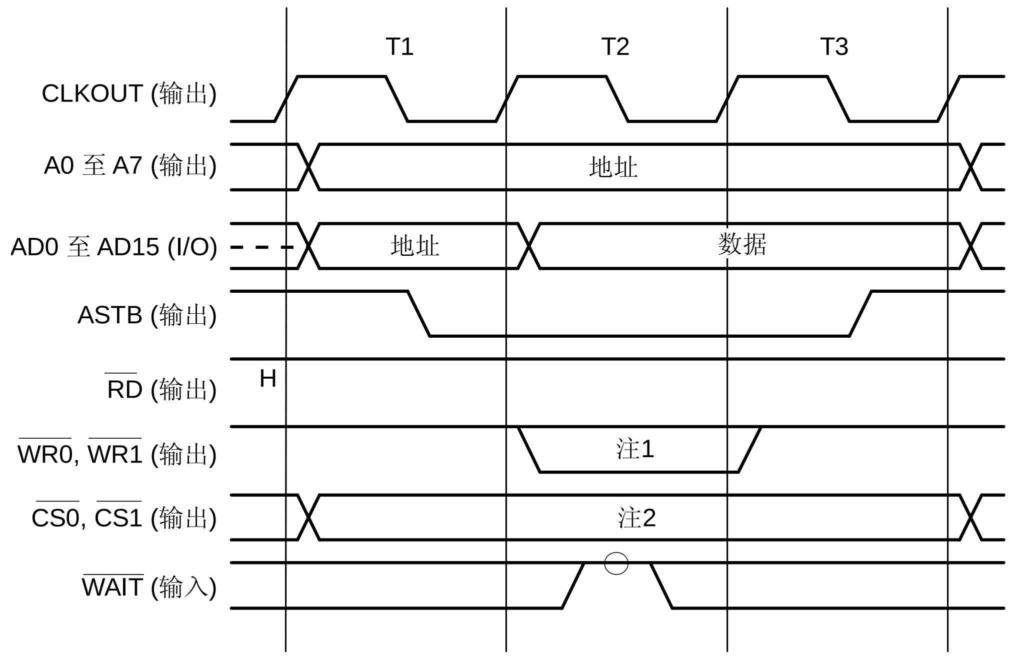
(5)　读取周期 (当地址建立等待状态，地址保持等待状态插入时)



注　　只有可以访问的 CS 空间有效。

备注　　**1.**　圆圈 O 表示采样时序。
　　　　2.　虚线表示高阻抗状态。

(6) 写入周期 **(**基本周期**)**



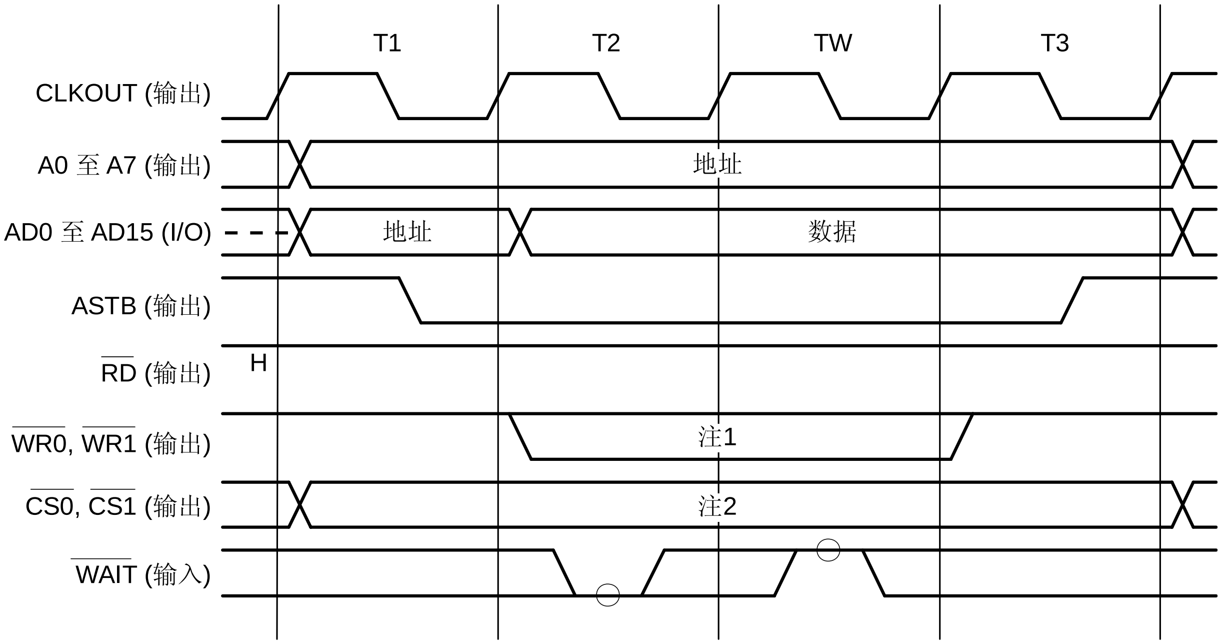
注　**1.** 这些信号的电平取决于访问数据总线宽度, 如下。

访问数据总线宽度	$\overline{WR1}$	$\overline{WR0}$
16位	低电平	低电平
8位	高电平	低电平

2. 只有可以访问的 CS 空间有效。

备注　**1.** 圆圈 ○ 表示采样时序。
2. 虚线表示高阻抗状态。

(7) 写入周期 **(**当数据等待状态 **(1** 等待**)** 插入时**)**



注　**1.**　这些信号的电平取决于访问数据总线宽度, 如下。

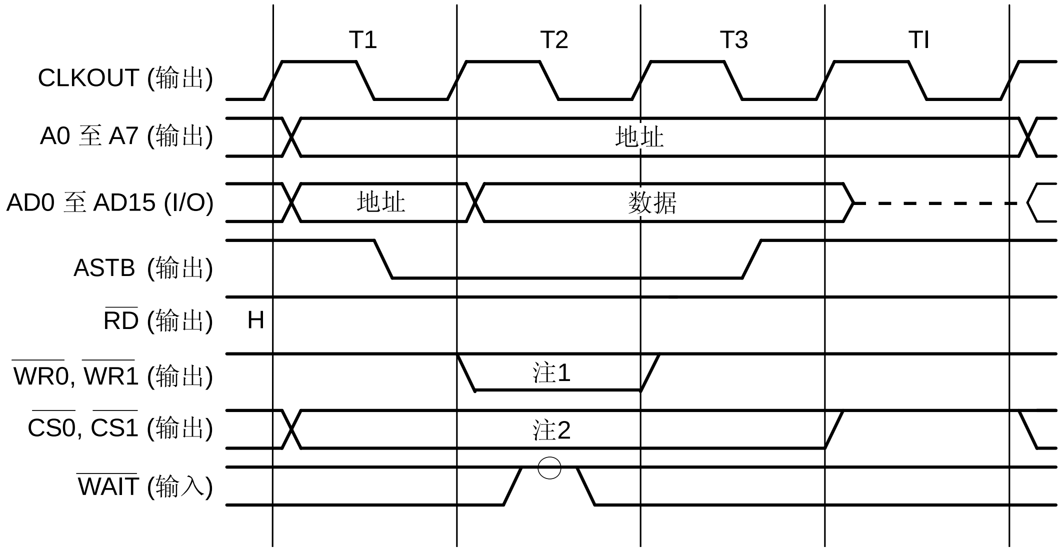
访问数据总线宽度	$\overline{WR1}$	$\overline{WR0}$
16位	低电平	低电平
8位	高电平	低电平

2.　只有可以访问的 CS 空间有效。

备注　**1.**　圆圈 ○ 表示采样时序。

2.　虚线表示高阻抗状态。

(8) 写入周期 **(**当空闲状态插入时**)**



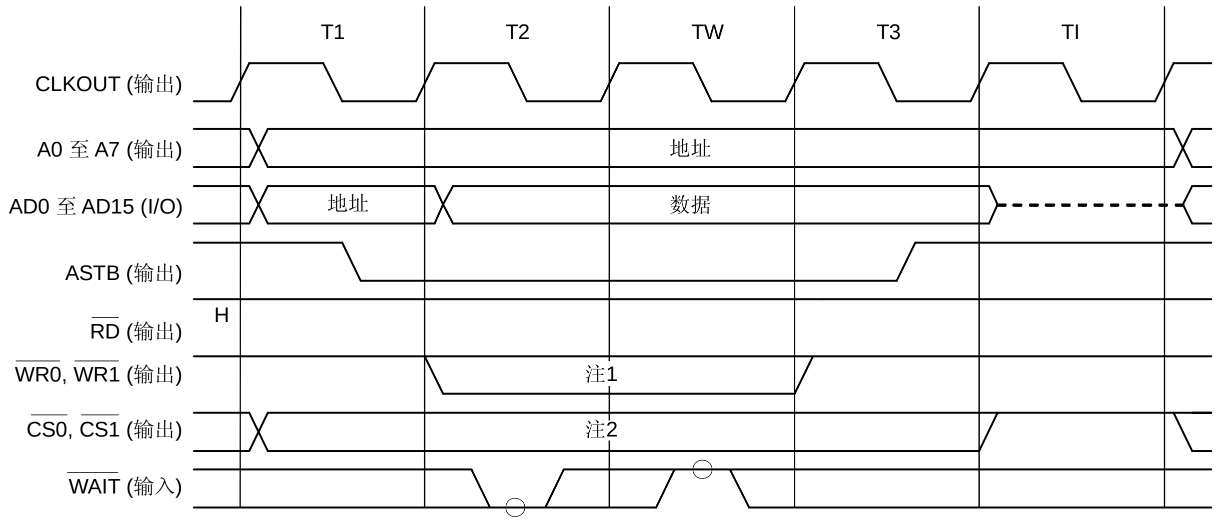
注　**1.** 这些信号的电平取决于访问数据总线宽度, 如下。

访问数据总线宽度	WR1	WR0
16位	低电平	低电平
8位	高电平	低电平

2. 只有可以访问的 CS 空间有效。

备注　**1.** 圆圈 ○ 表示采样时序。
2. 虚线表示高阻抗状态。

(9)　写入周期 (当数据等待状态 (1 等待)，空闲状态插入时)



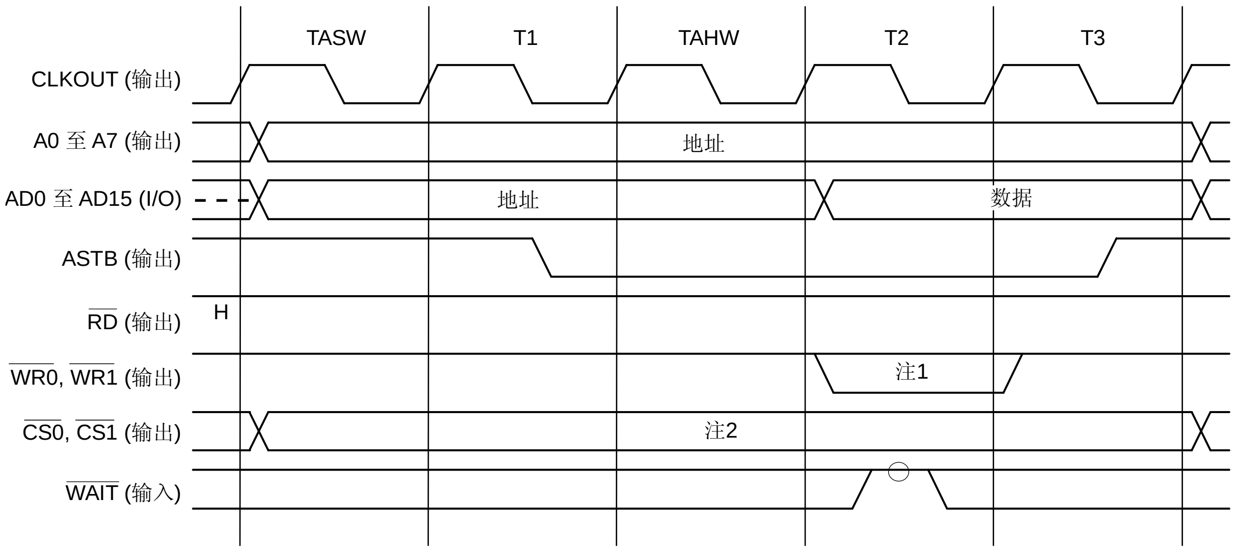
注　**1.**　这些信号的电平取决于访问数据总线宽度, 如下。

访问数据总线宽度	$\overline{\text{WR1}}$	$\overline{\text{WR0}}$
16位	低电平	低电平
8位	高电平	低电平

2.　只有可以访问的 CS 空间有效。

备注　**1.**　圆圈 O 表示采样时序。
2.　虚线表示高阻抗状态。

(10) 写入周期 **(**当地址建立等待状态，地址保持等待状态插入时**)**



注　**1.** 这些信号的电平取决于访问数据总线宽度, 如下。

访问数据总线宽度	$\overline{\text{WR1}}$	$\overline{\text{WR0}}$
16位	低电平	低电平
8位	高电平	低电平

2. 只有可以访问的 CS 空间有效。

备注　**1.** 圆圈 ○ 表示采样时序。
2. 虚线表示高阻抗状态。

18.9 总线优先次序

有两个外部总线周期：取指和操作数数据访问。
按照优先级次序，操作数数据访问较高，取指较低。
在读取修改为写入访问期间，可能在读取访问和写入访问之间插入一次取指。

表 18-3. 总线优先次序

优先级次序	外部总线周期	总线主控器
高	操作数数据访问	CPU
低	取指	CPU

18.10 边界操作条件

18.10.1 程序空间

禁止跳转到片上外围 I/O 区域。如果执行了上述操作，取到的是不确定数据，并且不执行从外部存储器取指。

18.10.2 数据空间

<R> μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A 提供地址移位功能。

借助这个功能，数据可以被分配到所有的地址，不论数据格式如何(字或半字)。但是，在字数据和半字数据的情况下，如果数据没有和边界对齐，至少产生两次总线周期，这样会导致总线效率下降。

(1) 在半字长度数据访问的情况下

当地址的 LSB 是 1，将会产生 2 次字节长度总线周期。

(2) 在字长度数据访问的情况下

(a) 当地址的 LSB 是 1，产生总线周期的顺序是字节长度总线周期、半字长度总线周期和字节长度总线周期。
(b) 当地址的低 2 位是 10，将会产生 2 次半字长度总线周期。

第十九章　DMA 功能（DMA 控制器）

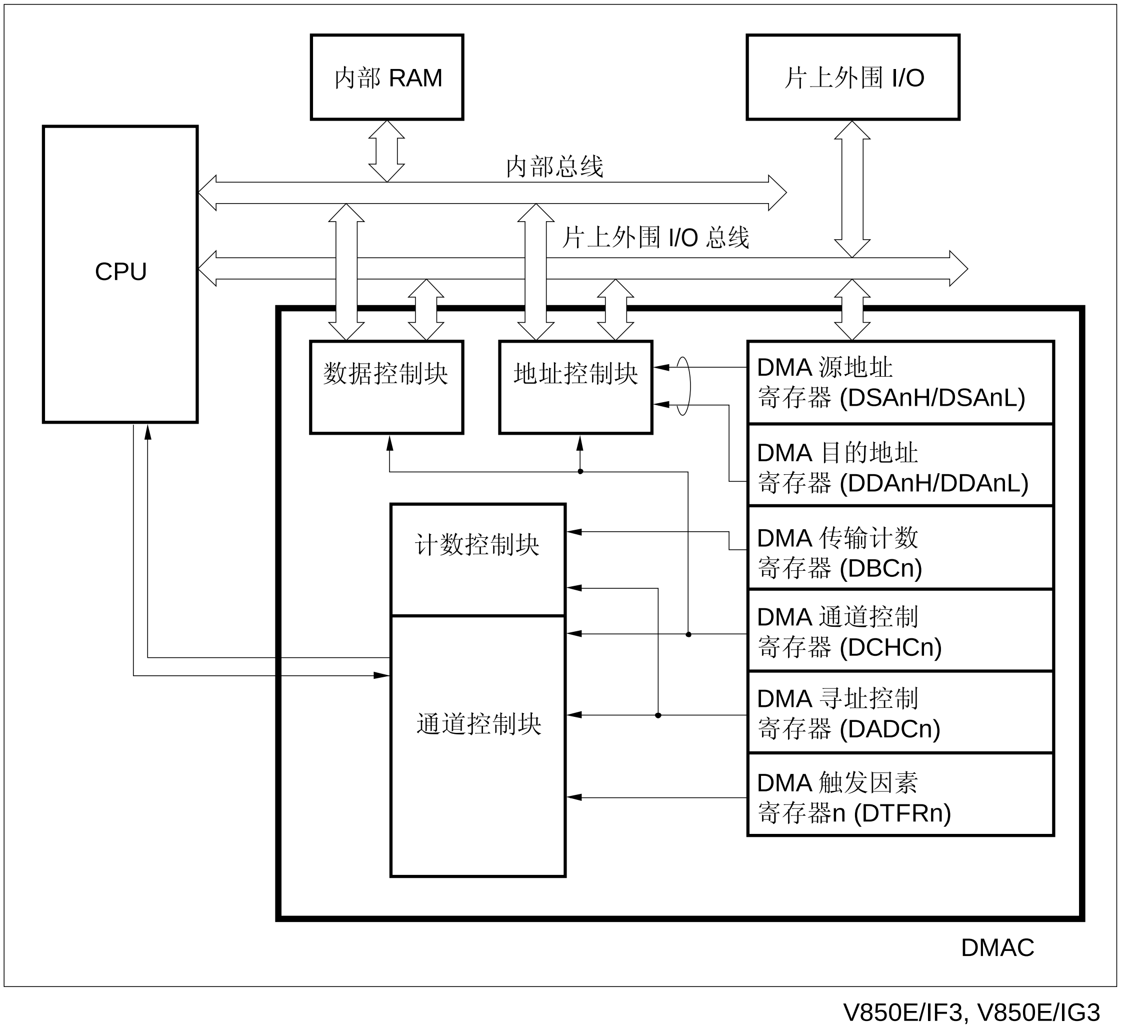
V850E/IF3 和 V850E/IG3 单片机包括一个直接存储器访问（DMA）控制器（DMAC），用于执行和控制 DMA 传输。

根据软件触发发出的片上外围 I/O 请求（串行接口、定时器和 A/D 转换器）或 DMA 请求中断，在 DMAC 控制内部存储器与外围 I/O 之间、或外围 I/O 们之间的数据传输。

19.1 特性

- 4 路独立 DMA 通道
- 传输单位：8/16 位
- 最大传输计数：65,536 （2^{16}）
- 传输类型：双周期传输
- 三种传输模式：
  - 单次传输模式
  - 单步传输模式
  - 块传输模式
- 传输请求：
  - 来自片上外围 I/O （串行接口、定时器和 A/D 转换器）
  - 来自软件触发的请求。
- 传输对象：
  - 内部存储器 ↔ 外围 I/O
  - 外围 I/O ↔ 外围 I/O
- 下一个地址设置功能

19.2 配置



备注　　n＝0 至 3

19.3 控制寄存器

19.3.1 DMA 源地址寄存器 0 至 3（DSA0 至 DSA3）

DSA0 至 DSA3 寄存器用于为 DMA 通道 n 设置 DMA 传输源地址（28 位）（n＝0 至 3）。这些寄存器分为两个 16 位寄存器，即 DSAnH 和 DSAnL。

由于这些寄存器配置为 2 级 FIFO 缓冲寄存器，包括主寄存器和从寄存器，在 DMA 传输期间，可以为 DMA 传输指定一个新的传输源地址（参见 **19.8 下一个地址设置功能**）。当设置下一个地址时，DSAn 寄存器的新的设置值传输到从寄存器，只有在 DMA 传输正常完成且 DCHCn.TCn 位设置为 1，或当 DCHCn.INITn 位设置为 1（n = 0 至 3）时新值变为有效。但是，即使 DCHCn.Enn 位清除为 0 以禁止 DMA 传输，然后设置 DSAn 寄存器，DSAn 寄存器的设置值无效。

（1）DMA 源地址寄存器 0 至 30H 至 3H（DSA0H 至 DSA3H）

DSA0H 至 DSA3H 寄存器可以按照 16 位宽度进行读取或写入。

系统复位后，该寄存器设置为不确定。

注意事项 1. 当设置一个片上外围 I/O 寄存器的地址作为源地址，请确保指定的地址在 FFFF000H 和 FFFFFFFH 之间，不能指定片上外围 I/O 寄存器的地址映像（3FFF000H 至 3FFFFFFH）。

2. 当 DMA 挂起暂停时，不要设置 DSAnH 寄存器。

复位后：不确定　R/W　地址：DSA0H FFFFF082H, DSA1H FFFFF08AH,
DSA2H FFFFF092H, DSA3H FFFFF09AH

DSAnH (n = 0 至 3)

15	14	13	12	11	10	9	8
IRSn	0	0	0	SAn27	SAn26	SAn25	SAn24

7	6	5	4	3	2	1	0
SAn23	SAn22	SAn21	SAn20	SAn19	SAn18	SAn17	SAn16

IRSn	DMA 传输源规范
0	片上外围 I/O
1	内部 RAM

SAn27 至 SAn16	设置DMA 传输源地址 (A27 至 A16)。在DMA传输期间，这些位存储下一个DMA 传输源地址。

注意事项 请确保将第 14 位至第 12 位清除为“0”。如果它们设置为“1”，操作无法保证。

（2）DMA 源地址寄存器 0L 至 3L（DSA0L 至 DSA3L）

DSA0L 至 DSA3L 寄存器可以按照 16 位宽度进行读取或写入。

系统复位后，该寄存器设置为不确定。

复位后：不确定 R/W 地址： DSA0L FFFFF080H, DSA1L FFFFF088H, DSA2L FFFFF090H, DSA3L FFFFF098H

DSAnL (n = 0 至 3)

15	14	13	12	11	10	9	8
SAn15	SAn14	SAn13	SAn12	SAn11	SAn10	SAn9	SAn8

7	6	5	4	3	2	1	0
SAn7	SAn6	SAn5	SAn4	SAn3	SAn2	SAn1	SAn0

SAn15 至 SAn0	设置DMA 传输源地址(A15 至 A0)。在DMA传输期间，这些位存储下一个DMA 传输源地址。

19.3.2 DMA 目的地址寄存器 0 至 3（DDA0 至 DDA3）

DDA0 至 DDA3 寄存器用于为 DMA 通道 n 设置 DMA 传输目的地址（28 位）（n＝0 至 3）。这些寄存器分为两个 16 位寄存器，即 DDAnH 和 DDAnL。

由于这些寄存器配置为 2 级 FIFO 缓冲寄存器，包括主寄存器和从寄存器，在 DMA 传输期间，可以为 DMA 传输指定一个新的传输源地址（参见 **19.8 下一个地址设置功能**）。当设置下一个地址时，DDAn 寄存器的新的设置值传输到从寄存器，只有在 DMA 传输正常完成且 DCHCn.TCn 位设置为 1，或当 DCHCn.INITn 位设置为 1（n ＝ 0 至 3）时新值变为有效。但是，即使 DCHCn.Enn 位清除为 0 以禁止 DMA 传输，然后设置 DDAn 寄存器，DDAn 寄存器的设置值无效。

（1）DMA 目的地址寄存器 0H 至 3H（DSA0H 至 DSA3H）

DDA0H 至 DDA3H 寄存器可以按照 16 位宽度进行读取或写入。

系统复位后，该寄存器设置为不确定。

注意事项 1. 当设置一个片上外围 I/O 寄存器的地址作为源地址，请确保指定的地址在 FFFF000H 和 FFFFFFFH 之间，不能指定片上外围 I/O 寄存器的地址映像（3FFF000H 至 3FFFFFFH）。

2. 当 DMA 挂起暂停时，不要设置 DDAnH 寄存器。

复位后：不确定　R/W　地址：　DDA0H FFFFF086H, DDA1H FFFFF08EH,
DDA2H FFFFF096H, DDA3H FFFFF09EH

DDAnH (n = 0 至 3)

15	14	13	12	11	10	9	8
IRAn	0	0	0	DAn27	DAn26	DAn25	DAn24

7	6	5	4	3	2	1	0
DAn23	DAn22	DAn21	DAn20	DAn19	DAn18	DAn17	DAn16

IRAn	DMA 传输目的规范
0	片上外围 I/O
1	内部RAM

DAn27 至 DAn16	设置DMA 传输目的地址 (A27 至 A16)。在DMA传输期间，这些位存储下一个DMA 传输目的地址。

注意事项 请确保将第 14 位至第 12 位清除为“0”。如果它们设置为“1”，操作无法保证。

（2）DMA 目的地址寄存器 0L 至 3L（DDA0L 至 DDA3L）

DDA0L 至 DDA3L 寄存器可以按照 16 位宽度进行读取或写入。

系统复位后，该寄存器设置为不确定。

复位后：不确定　R/W　地址：DDA0L FFFFF084H, DDA1L FFFFF08CH, DDA2L FFFFF094H, DDA3L FFFFF09CH

DDAnL (n = 0 至 3)

15	14	13	12	11	10	9	8
DAn15	DAn14	DAn13	DAn12	DAn11	DAn10	DAn9	DAn8

7	6	5	4	3	2	1	0
DAn7	DAn6	DAn5	DAn4	DAn3	DAn2	DAn1	DAn0

DAn15 至 DAn0	设置DMA 传输目的地址 (A15 至 A0)。在DMA传输期间，这些位存储下一个DMA 传输目的地址。

19.3.3 DMA 传输计数寄存器 0 至 3（DBC0 至 DBC3）

DBC0 至 DBC3 寄存器是 16 位寄存器，用于设置 DMA 通道 n 的字节传输计数（n＝0 至 3）。在 DMA 传输期间，这些寄存器存储剩余传输计数。

由于这些寄存器被配置为 2 级 FIFO 缓冲寄存器，包括主寄存器和从寄存器，在 DMA 传输期间，可以为 DMA 传输指定一个新的传输源地址（参见 **19.8 下一个地址设置功能**）。当设置下一个地址时，DBCn 寄存器的新的设置值传输到从寄存器，只有在 DMA 传输正常完成且 DCHCn.TCn 位设置为 1，或当 DCHCn.INITn 位设置为 1（n ＝ 0 至 3）时新值变为有效。但是，即使 DCHCn.Enn 位清除为 0 以禁止 DMA 传输，然后设置 DBCn 寄存器，DBCn 寄存器的设置值无效。

每次传输，这些寄存器就递减 1，当发生借位时，传输结束。

这些寄存器可以按照 16 位宽度进行读取或写入。

系统复位后，该寄存器设置为不确定。

注意事项 当 DMA 挂起暂停时，不要设置 DBCn 寄存器。

备注 终止计数时没有重写该寄存器，之后如果在 DMA 传输期间读取 DBCn 寄存器，读取到的是在 DMA 传输之前设置的瞬时值（即使 DMA 传输已经结束，也不会读到 0000H）。

复位后：不确定 R/W 地址： DBC0 FFFFF0C0H, DBC1 FFFFF0C2H, DBC2 FFFFF0C4H, DBC3 FFFFF0C6H

DBCn (n = 0 至 3)

15	14	13	12	11	10	9	8
BCn15	BCn14	BCn13	BCn12	BCn11	BCn10	BCn9	BCn8

7	6	5	4	3	2	1	0
BCn7	BCn6	BCn5	BCn4	BCn3	BCn2	BCn1	BCn0

BCn15 至 BCn0	传输计数设置 (存储DMA传输期间剩余的传输计数)
0000H	传输计数1 或剩余的传输计数
0001H	传输计数2 或剩余的传输计数
:	:
FFFFH	传输计数65536 (2^{16}) 或剩余的传输计数

19.3.4 DMA 寻址控制寄存器 0 至 3（DADC0 至 DADC3）

DADC0 至 DADC3 寄存器是 16 位寄存器，用于控制 DMA 通道 n 的 DMA 传输模式（n＝0 至 3）。在 DMA 传输期间，不能访问这些寄存器。

这些寄存器可以按照 16 位宽度进行读取或写入。

系统复位后，该寄存器设置为 0000H。

注意事项 **1. DSn0** 位设置有多少位数据需要传输。
如果传输数据宽度设置为 **16** 位，总是从最低位和**"0"**对齐的地址开始传输，不能从奇地址开始传输。
2. 当目标通道处于下列时期，设置 **DADCn** 寄存器（如果在其它时间设置该寄存器，操作无法保证）。
- 从系统复位到第一次产生 **DMA** 传输请求
- 从 **DMA** 传输结束（在终止计数后）到产生下一次 **DMA** 传输请求
- 从 **DMA** 传输的强制终止（当 **DCHCn.INITn** 位被设置为 **1** 之后）到产生下一次 **DMA** 传输请求

复位后： 0000H　R/W　地址：　DADC0 FFFFF0D0H, DADC1 FFFFF0D2H,
DADC2 FFFFF0D4H, DADC3 FFFFF0D6H

DADCn (n = 0 至 3)

15	14	13	12	11	10	9	8
0	DSn0	0	0	0	0	0	0

7	6	5	4	3	2	1	0
SADn1	SADn0	DADn1	DADn0	TMn1	TMn0	0	0

DSn0	DMA 传输的传输数据宽度的设置
0	8 位
1	16 位

SADn1	SADn0	DMA通道n的传输源地址计数方向设置
0	0	递增
0	1	递减
1	0	固定
1	1	禁止设置

DADn1	DADn0	DMA通道n的传输目的地址计数方向设置
0	0	递增
0	1	递减
1	0	固定
1	1	禁止设置

TMn1	TMn0	DMA传输期间的传输模式设置
0	0	单次传输模式
0	1	单步传输模式
1	0	禁止设置
1	1	块传输模式

注意事项 请确保将第 **15** 位、第 **13** 位至第 **8** 位、第 **1** 位和第 **0** 位清除为**"0"**。如果它们被设置为**"1"**，操作无法保证。

19.3.5 DMA 通道控制寄存器 0 至 3（DCHC0 至 DCHC3）

DCHC0 至 DCHC3 寄存器是 8 位寄存器，用于控制 DMA 通道 n 的 DMA 传输操作模式（n = 0 至 3）。

这些寄存器可以按字节或按位进行读取或写入。（但是，第 7 位是只读的）

系统复位后，该寄存器 0

.设置为 00H。

注意事项 **1.** 如果因为 **MLEn** 位设置为 **1** 而结束传输，且下一次传输请求由 **DMA** 传输（硬件 **DMA**）产生，通过片上外围 **I/O** 的中断启动该传输，则 **TCn** 位设置为 **1**（不自动清除为 **0**）时执行下一次传输。

2. 当目标通道处于下列时期，设置 **MLEn** 位（如果在其它时间设置该位，操作无法保证）。

- 从系统复位到第一次产生 **DMA** 传输请求
- 从 **DMA** 传输结束（在终止计数后）到产生下一次 **DMA** 传输请求
- 从 **DMA** 传输的强制终止（当 **INITn** 位被设置为 **1** 之后）到产生下一次 **DMA** 传输请求

3. 当 **MLEn** 位设置为 **1**，如果在最后一个传输周期内 **DMA** 传输强制终止，执行的操作和传输结束时（**TCn** 位设置为 **1**）的操作相同（强制终止时 **Enn** 位清除为 **0**，无论 **MLEn** 位的值如何）。

在这种情况下，当产生下一个 **DMA** 传输请求时，**Enn** 位必须设置为 **1**，且必须读取 **TCn** 位（清除为 **0**）。

4. 在 **DMA** 传输结束时（在终止计数期间），随着 **Enn** 位清除为 **0** 然后 **TCn** 位设置为 **1**，各个位更新。如果轮询 **TCn** 位和 **Enn** 位的状态，且各个位更新后读取 **DCHCn** 寄存器，因此，可能读取到的值表示该状态**“**传输未结束并禁止**”**（**TCn** 位 **= 0** 和 **Enn** 位 **= 0**）（这不是异常）

5. 在 **DMA** 传输结束之后（终止计数之后），请确保读取（清除为 **0**）**TCn** 位。只有当满足下列两个条件时，**TCn** 位才必须需要读取（清除为 **0**）。

- 在 **DMA** 传输结束时（终止计数期间），**MLEn** 位设置为 **1**。
- 下一次 **DMA** 传输（硬件 **DMA**）启动因素是来自片上外围 **I/O**（硬件 **DMA**）的中断。

如果任何一个条件都不满足，在下一次 **DMA** 传输请求产生之前，请确保读取（清除为 **0**）**TCn** 位。

如果当 **TCn** 位设置为 **1** 时，产生下一次 **DMA** 传输请求，则操作无法保证。

6. 当 **DMA** 挂起暂停时，不要设置 **Enn** 和 **STGn** 位，否则，操作无法保证。

7. 不要通过将 **Enn** 位清除为 **0** 来结束 **DMA** 传输。

8. **DMA** 传输的状态和该寄存器的值之间的关系如下。

- **DMA** 传输在进行中： **TCn** 位 **= 0**，**Enn** 位 **= 1**
- **DMA** 传输放弃： **TCn** 位 **= 0**，**Enn** 位 **= 0**
- **DMA** 传输停止（结束）： **TCn** 位 **= 1**

复位后： 00H　R/W　地址：　DCHC0 FFFFF0E0H, DCHC1 FFFFF0E2H,
DCHC2 FFFFF0E4H, DCHC3 FFFFF0E6H

	<7>	6	5	4	<3>	<2>	<1>	<0>
DCHCn (n = 0 至 3)	TCn	0	0	0	MLEn	INITn	STGn	Enn

TCn[注1]	表示DMA通道n的DMA 传输是否结束的状态位
0	DMA 传输尚未结束。
1	DMA 传输已经结束。

在最后的DMA传输时，该位置位(1)，当读取该位时，则被清除(0)。如果执行的DMA传输是从内部RAM传输数据，在最后的传输结束之后4个时钟该位置位(1)。

MLEn	
	当该位在DMA 传输结束(在终止计数输出时)置位(1)，则Enn位不被清除(0)且保持DMA 传输使能状态。 如果下一个DMA传输启动因素是从片上外围I/O(硬件DMA)输入，即使TCn 位未被读取，响应DMA传输请求。 如果下一个DMA传输启动因素是通过STGn 位设置为1(软件DMA)输入，如果TCn位被读取且清除(0)，则响应DMA 传输请求。 当该位在DMA传输结束(在终止计数输出时)时清除(0)，则Enn 位清除(0)且进入DMA传输禁止状态。在下一次DMA传输请求时，必须读取TCn位且Enn 位必须置位(1)。

INITn[注2]	如果在DMA 传输期间或DMA被挂起时该位置位(1)， 强制终止DMA 传输。

STGn[注2]	在DMA 传输使能状态(TCn 位 = 0, Enn 位 = 1)下， 将该位置位(1)，则启动DMA 传输。

Enn	DMA通道n的DMA 传输使能或禁止的设置
0	DMA 传输禁止
1	DMA 传输使能

- 当DMA 传输结束时，该位被清除(0)。当DMA传输因为INITn 位的置位 (1)强制终止时，也会清除(0)该位。
- 直到DMA 传输已经结束DBCn寄存器设置的次数或DMA 传输被INITn 位强制终止，不要将Enn 位置位(1)。

注　**1.** TCn 位只读。
　　2. INITn 和 STGn 位只写。如果读取这些位，读到的是 0。

注意事项　1.　请确保将第 6 位至第 4 位清除为“0”。如果它们设置为“1”，操作无法保证。

19.3.6 DMA 触发因素寄存器 0 至 3（DTFR0 至 DTFR3）

DTFR0 至 DTFR3 寄存器是 8 位寄存器，用于通过片上外围 I/O 的中断请求信号控制 DMA 传输启动触发源。

这些寄存器设置的中断请求信号用作 DMA 传输启动因素。

这些寄存器可以按字节或按位进行读取或写入。但是，只有第 7 位（DFn）可以按位进行读取或写入；第 5 位至第 0 位（IFCn5 至 IFCn0）仅可以按字节进行读取或写入。

系统复位后，该寄存器设置为 00H。

注意事项 **1.** **请确保按照下列步骤来改变 DTFRn 寄存器的设置。**

- **当 IFCn5 至 IFCn0 位的设置值没有设置到另一个通道的 IFCm5 至 IFCm0 位（n = 0 至 3，m = 0 至 3，n ≠ m）**

<R> **<1> 当 DCHCn.Enn 位清除为 0 时进行步骤<3>至<5>，当 Enn 位设置为 1 时进行步骤 <2>至<5> 。**

<2> 停止待写入的通道（DCHCn.INITn 位 = 1）的 DMAn 操作。

<3> 改变 DTFRn 寄存器的设置（请确保设置 DFn 位 = 0，并按字节操作改变设置）。

<4> 要清除 DMA 传输请求，将 DMA 传输请求标志（DTFRn.DFn）清除为 0。

<5> 使能 DMAn 操作（Enn 位 = 1）。

- **当 IFCn5 至 IFCn0 位的设置值没有设置到另一个通道的（n = 0 至 3，m = 0 至 3，n ≠ m）**

<R> **<1> 当 DCHCn.Enn 位清除为 0 时进行步骤<4>至<6> ，当 Enn 位设置为 1 时进行步骤 <2>至<6> 。**

<2> 停止待写入的通道（DCHCn.INITn 位 = 1）的 DMAn 操作。

<3> 设置到 IFCm5 至 IFCm0 位的值用来重写 IFCn5 至 IFCn0 位的值相同（DCHCm.INITm 位 = 1），停止该通道的 DMAm 操作。

<4> 改变 DTFRn 寄存器的设置（请确保设置 DFn 位 = 0，并按字节操作改变设置）。

<5> 要清除 DMA 传输请求，将 DMA 传输请求标志（DTFRn.DFn）清除为 0。

<6> 使能 DMAn 操作（Enn 和 Emm 位 = 1）。

2. **在待机模式下（IDLE 或 STOP 模式），来自片上外围 I/O 的中断请求作为 DMA 传输启动因素而挂起。在恢复正常操作模式之后，执行该挂起的 DMA 启动因素。**

3. **使用 IFCn5 至 IFCn0 位改变 DMA 传输的启动因素，请确保在改变之后，立即使用指令设置 DFn 位（0）。**

复位后：00H　R/W　地址：　DTFR0 FFFFF810H, DTFR1 FFFFF812H,
DTFR2 FFFFF814H, DTFR3 FFFFF816H

	<7>	6	5	4	3	2	1	0
DTFRn (n = 0 至 3)	DFn	0	IFCn5	IFCn4	IFCn3	IFCn2	IFCn1	IFCn0

DFn[注]	DMA传输请求标志
0	未请求DMA传输
1	请求DMA传输

注　不要通过软件将 DFn 位设置为“1”。
当 DMA 传输禁止时（包括软件强制终止），如果发生了一个指定作为启动 DMA 传输的触发因素，且必需清除 DMA 传输请求，则停止中断源的操作，将 0 写入 DFn 位（比如，在串行接收的情况下禁止接收）。如果明确在 DMA 传输再次恢复之前不会发生中断，则无需停止中断源的操作。

注意事项　1.　关于 IFCn5 至 IFCn0 位，请参见表 19-1 DMA 启动因素。

2.　请确保将第 6 位清除为“0”。如果它们设置为“1”，操作无法保证。

表 19-1. DMA 启动因素（1/2）

IFCn5	IFCn4	IFCn3	IFCn2	IFCn1	IFCn0	中断源
0	0	0	0	0	0	禁止来自片上外围 I/O 的 DMA 请求
0	0	0	0	0	1	INTLVIL
0	0	0	0	1	0	INTLVIH
0	0	0	0	1	1	INTP11
0	0	0	1	0	0	INTP12
0	0	0	1	0	1	INTP13
0	0	0	1	1	0	INTP15
0	0	0	1	1	1	INTTB0OV_BASE[注]
0	0	1	0	0	0	INTTB1OV_BASE[注]
0	0	1	0	0	1	INTCMP0L
0	0	1	0	1	0	INTCMP0F
0	0	1	0	1	1	INTCMP1L
0	0	1	1	0	0	INTCMP1F
0	0	1	1	0	1	INTTB0CC0
0	0	1	1	1	0	INTTB0CC1
0	0	1	1	1	1	INTTB0CC2
0	1	0	0	0	0	INTTB0CC3
0	1	0	0	0	1	INTTB1CC0
0	1	0	0	1	0	INTTB1CC1
0	1	0	0	1	1	INTTB1CC2
0	1	0	1	0	0	INTTB1CC3
0	1	0	1	0	1	INTTTEQC00
0	1	0	1	1	0	INTTTEQC01
0	1	0	1	1	1	INTTTEQC10
0	1	1	0	0	0	INTTTEQC11
0	1	1	0	0	1	INTTA0CC0
0	1	1	0	1	0	INTTA0CC1
0	1	1	0	1	1	INTTA1CC0
0	1	1	1	0	0	INTTA1CC1
0	1	1	1	0	1	INTTA2CC0
0	1	1	1	1	0	INTTA2CC1
0	1	1	1	1	1	INTTA3CC0
1	0	0	0	0	0	INTTA3CC1
1	0	0	0	0	1	INTTA4CC0
1	0	0	0	1	0	INTTA4CC1
1	0	0	0	1	1	INTDMA0
1	0	0	1	0	0	INTDMA1
1	0	0	1	0	1	INTDMA2
1	0	0	1	1	0	INTDMA3

备注　n＝0 至 3

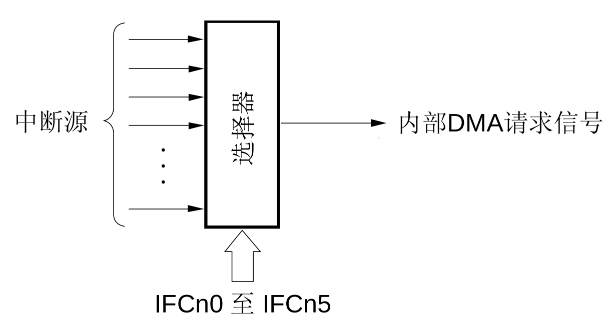
<R>　注　6 相 PWM 输出模式下，INTTBaOV_BASE 是一个中断请求信号，在 INTTBaOV 被 TMQa 选项分选之前产生（a = 0，1）。INTTBaCC0 也是一个在分选之前的中断请求信号。详情参见图 **10-2 TMQn** 选项。

表 19-1. DMA 启动因素（2/2）

IFCn5	IFCn4	IFCn3	IFCn2	IFCn1	IFCn0	中断源
1	0	0	1	1	1	INTUBTIR
1	0	1	0	0	0	INTUBTIT
1	0	1	0	0	1	INTUBTIF
1	0	1	0	1	0	INTUA0R
1	0	1	0	1	1	INTUA0T
1	0	1	1	0	0	INTCB0R
1	0	1	1	0	1	INTCB0T
1	0	1	1	1	0	INTUA1R
1	0	1	1	1	1	INTUA1T
1	1	0	0	0	0	INTCB1R
1	1	0	0	0	1	INTCB1T
1	1	0	0	1	0	INTUA2R
1	1	0	0	1	1	INTUA2T
1	1	0	1	0	0	INTCB2R
1	1	0	1	0	1	INTCB2T
1	1	0	1	1	0	INTIIC
1	1	0	1	1	1	INTAD0
1	1	1	0	0	0	INTAD1
1	1	1	0	0	1	INTAD2
1	1	1	0	1	0	INTTM0EQ0
1	1	1	0	1	1	INTTM1EQ0
1	1	1	1	0	0	INTTM2EQ0
1	1	1	1	0	1	INTTM3EQ0
其它情况						禁止设置

备注　　n＝0 至 3

中断源和 DMA 传输触发之间的关系如下（n = 0 至 3）。



注意事项 **1.** 当 **IFCn5** 至 **IFCn0** 位设置的中断因素发生时，**DMA** 传输启动。要防止中断服务的执行，通过中断控制寄存器屏蔽该中断。即使中断屏蔽，**DMA** 传输也会启动。

2. 如果因为时钟发生器的预分频器 **2** 的设置，**CPU** 时钟的频率低于各个片上外围 **I/O** 的时钟，**DMA** 传输启动因素可能无法响应。

19.4 传输模式

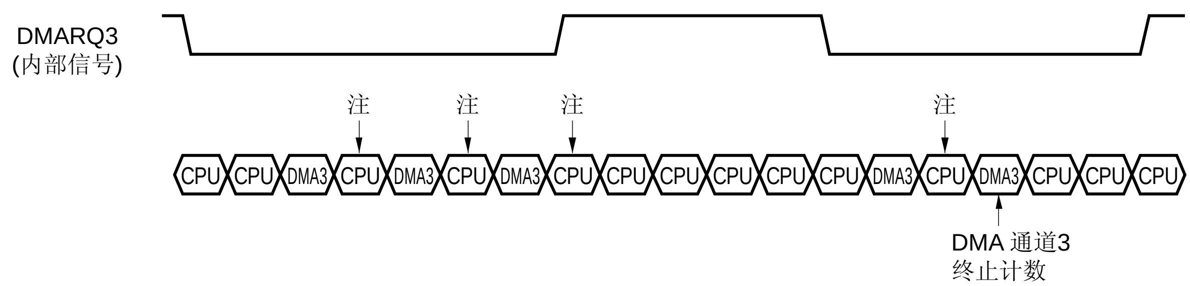
19.4.1 单次传输模式

在单次传输模式下，在每次字节/半字传输时 DMAC 释放总线。如果有后续的 DMA 传输请求，则再执行一次传输。该操作一直进行下去，直到发生一个终止计数。

当 DMAC 释放总线后，如果发生另一个更高优先级的 DMA 传输请求，则较高优先级的 DMA 请求总是优先处理。如果在单次传输完成之后一个时钟，发生另一个更低优先级的 DMA 传输请求，但是，即使上一个较高优先级的 DMA 传输请求信号保持有效，该请求不会优先。当总线释放给 CPU 后，执行具有低优先级的通道的 DMA 传输。

图 19-1 和图 19-4 是单次传输的示例。

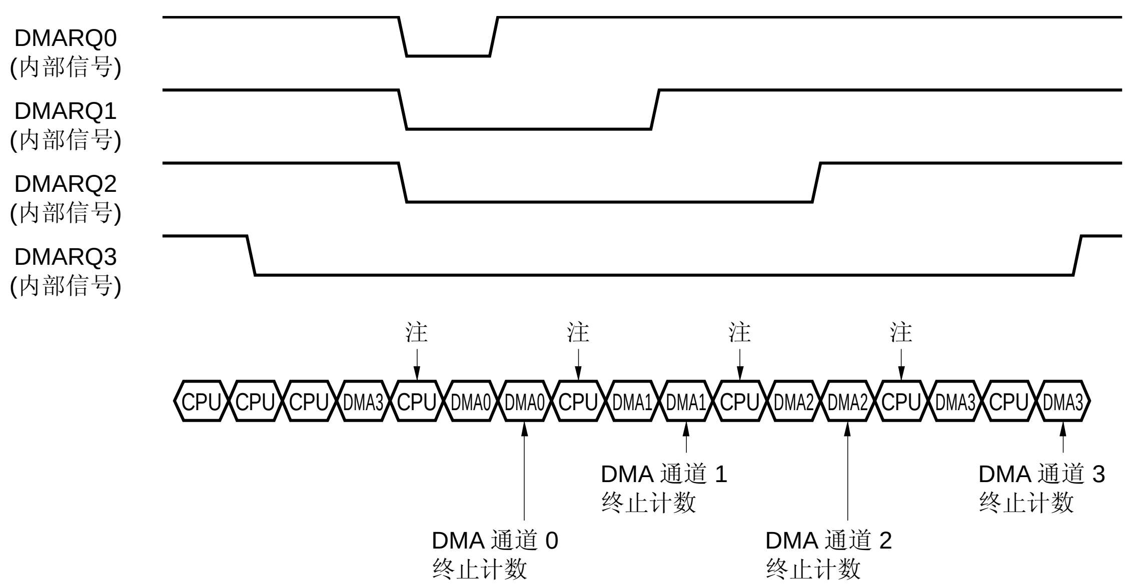
图 19-1. 单次传输示例 1



注 总线总是释放。

图 19-2 是单次传输的示例，其中产生了一个较高优先级的 DMA 请求。DMA 通道 0 至 2 为块传输模式，通道 3 为单次传输模式。

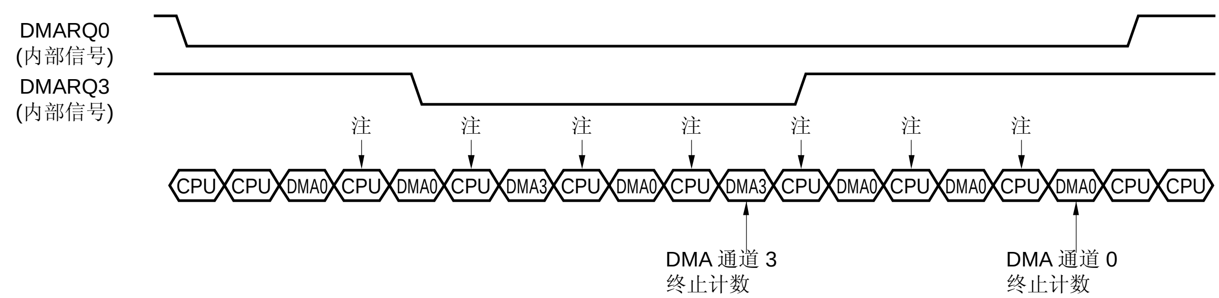
图 19-2. 单次传输示例 2



注 总线总是释放。

图 19-3 是单次传输的示例，在单次传输完成之后一个时钟，产生了一个较低优先级的 DMA 请求。DMA 通道 0 和 3 被用作单次传输。如果有两个 DMA 传输请求信号同时变为有效，两个 DMA 传输操作交替执行。

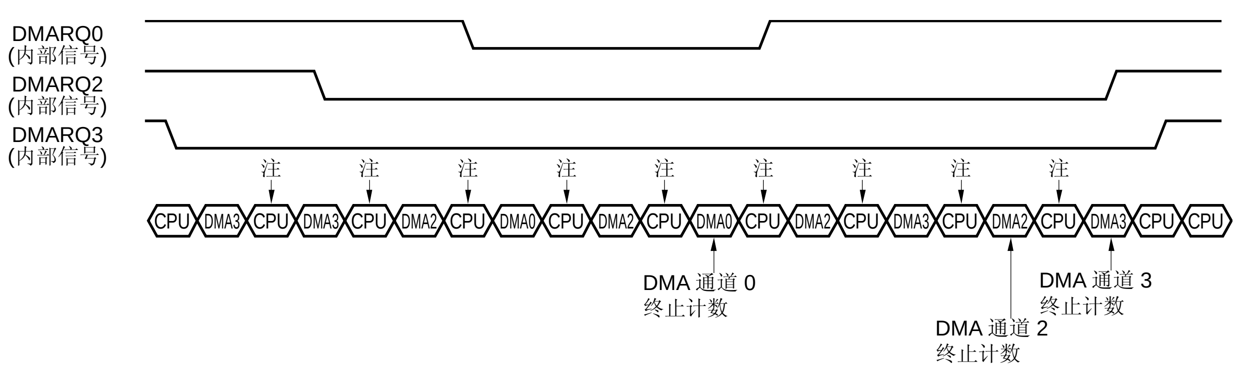
图 19-3. 单次传输示例 3



注 总线总是释放。

图 19-4 是单次传输的示例，在单次传输完成之后一个时钟，产生了两个或更多较低优先级的 DMA 请求。DMA 通道 0、2 和 3 用作单次传输。如果有三个或更多的 DMA 传输请求信号同时变为有效，两个 DMA 传输操作交替执行，从最高优先级的开始。

图 19-4. 单次传输示例 4



注 总线总是释放。

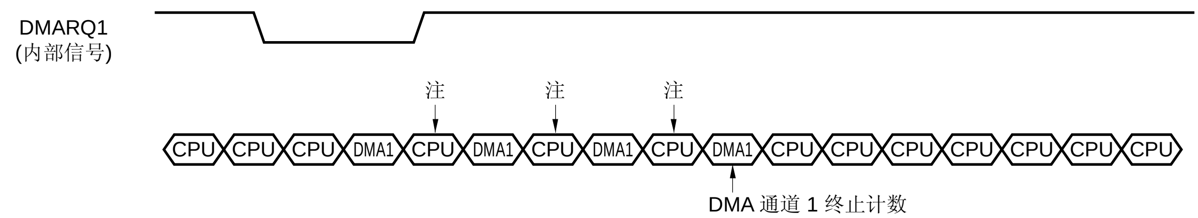
19.4.2 单步传输模式

在单次传输模式下，在每次字节/半字传输时 DMAC 释放总线。如果有后续的 DMA 传输请求，则再执行一次传输。该操作一直进行下去，直到发生一个终止计数。

当 DMAC 释放总线后，如果发生另一个更高优先级的 DMA 传输请求，则较高优先级的 DMA 请求总是优先处理。

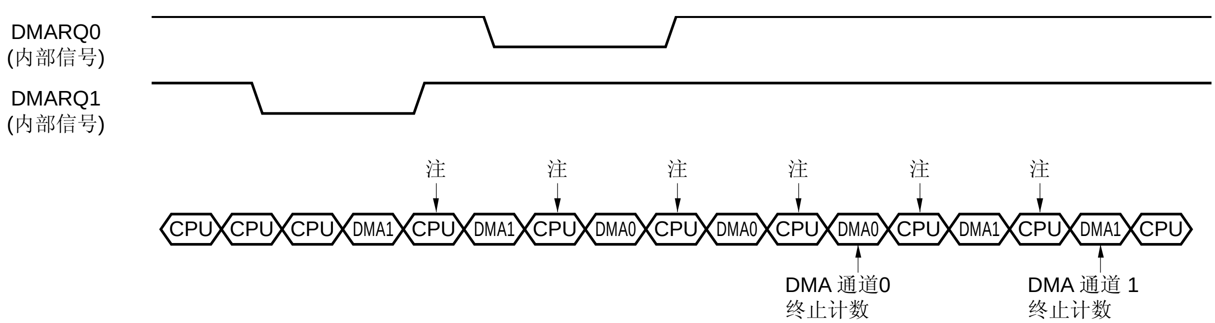
下面时一个单步传输的示例，图 19-6 展示了单步传输的示例，其中产生了一个较高优先级的 DMA 请求。DMA 通道 0 和 1 为单步传输模式。

图 19-5. 单步传输示例 1



注 总线总是释放。

图 19-6. 单步传输示例 2



注 总线总是释放。

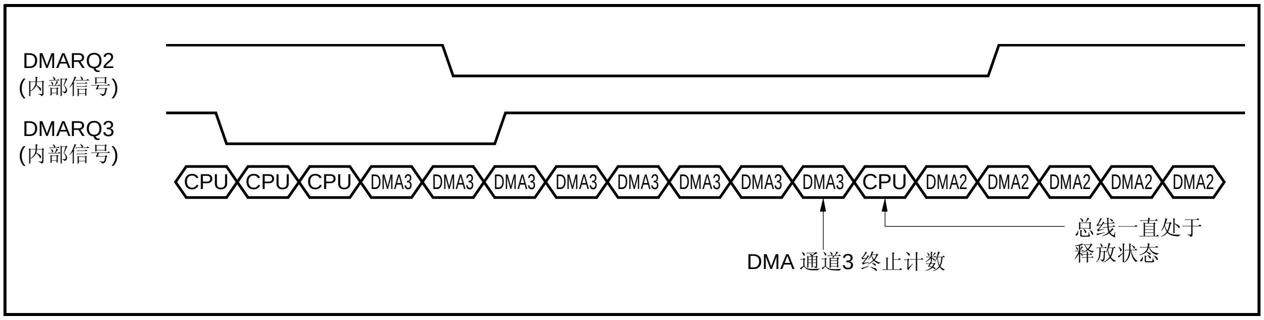
19.4.3 块传输模式

在块传输模式下，一旦传输启动，DMAC 继续传输操作而不释放总线，直到发生一个终止计数。在块传输期间，不会响应其他 DMA 请求。

在块传输结束之后，DMAC 释放总线，可以响应另一个 DMA 传输。

下面是一个块传输的示例，其中发生一个较高优先级的 DMA 请求。DMA 通道 2 和 3 处于块传输模式。

图 19-7. 块传输示例



19.5 传输类型

19.5.1 2 周期传输

在 2 周期传输中，数据传输在两个周期内执行，一个读取周期（至 DMAC 的源）和一个写入周期（DMAC 至目的）。

在第一个周期中，输出源地址，并执行从源地址向 DMAC 的读操作。在第二个周期中，输出传输目的地址，并执行从 DMAC 向目的地的写操作。

注意事项 在读周期和写周期之间总是插入 1 至 2 个时钟的空闲周期。

19.6 传输目标

19.6.1 传输类型和传输目标

表 19-2 列出了传输类型和传输目标之间的关系。标记“√” 表示允许传输，“×”表示禁止传输）。

表 19-2 传输类型和传输目标之间的关系

		传输目的地		
		内部 ROM	片上外围I/O[注]	内部 RAM
传输源	片上外围 I/O[注]	×	√	√
	内部 RAM	×	√	×
	内部 ROM	×	×	×

注 如果传输目标是片上外围 I/O，只能使用单次传输模式。

注意事项 1. 在表 19-2 中，用“×”标记的传输目的地和传输源组合，其操作将无法保证。

2. 3FFF000H 和 3FFFFFFH 之间的地址不能指定作为 DMA 传输的源地址和目的地址。
请确保指定的地址位于 FFFF000H 和 FFFFFFFH 之间。

备注 如果执行 DMA 传输来传输一个片上外围 I/O 寄存器（作为传输源或目标）的数据，请确保指定的传输宽度和寄存器宽度一致。比如，要执行 8 位寄存器的 DMA 传输，请确保指定字节（8 位）传输。

19.7 DMA 通道优先级

DMA 通道优先级固定如下所示：

DMA 通道 0 > DMA 通道 1 > DMA 通道 2 > DMA 通道 3

在块传输模式下，决不可以切换传输通道。
在单步传输模式下，如果在总线已释放时发生一个较高优先级的 DMA 请求，响应这个较高优先级的 DMA 请求。

19.8 下一个地址设置功能

DSAnH、DSAnL、DDAnH、DDAnL 和 DBCn 寄存器时两级 FIFO 缓冲寄存器，包括一个主寄存器和一个从寄存器（n = 0 至 3）。

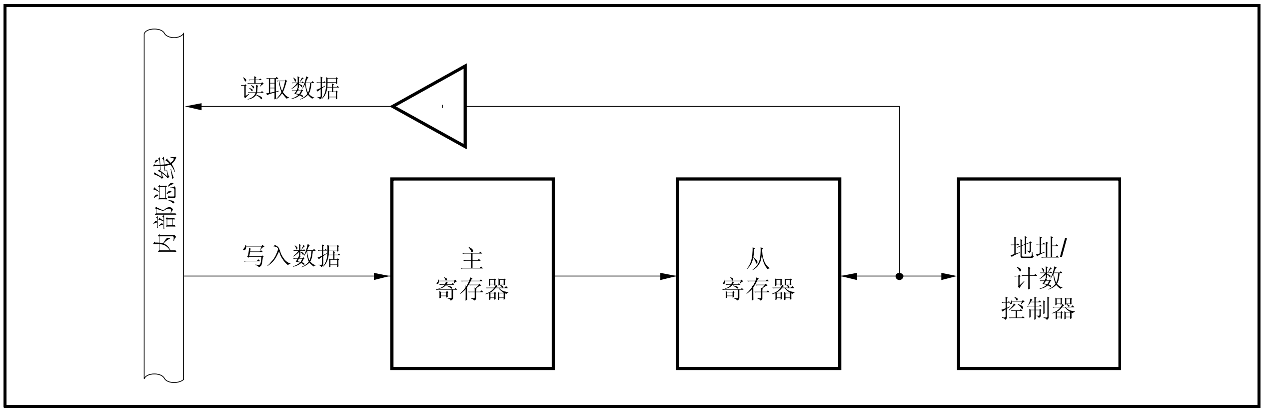
当发生终止计数时，这些寄存器自动重写，重写的值就是之前的瞬时值。

在DMA传输期间，如果对这些寄存器有新的DMA传输设置，因此，在传输[注]完成之后，这些寄存器的值自动更新为新设置的值。

注 要进行新 DMA 传输设置，需要确认 DMA 传输已经启动。如果在 DMA 传输已经启动之前，就进行新设置，主寄存器和从寄存器的设置值都需要重写。

图 19-8 是缓冲寄存器的配置。

图 19-8. 缓冲寄存器配置



实际上，DMA 传输的执行会根据从寄存器的内容决定。

设置值反映在主寄存器和从寄存器是不同的，取决于设置的时序（时期），如下所示。

- **从系统复位到第一次产生 DMA 传输请求**
- **从 DMA 传输结束（在终止计数后）到产生下一次 DMA 传输请求**
- **从 DMA 传输的强制终止（当 DCHCn.INITn 位被设置为 1 之后）到产生下一次 DMA 传输请求**

（1） 从系统复位到第一次产生 DMA 传输请求的时期

设置值在主寄存器和从寄存器均有反映。

（2） 在 DMA 传输期间（从 DMA 传输请求的产生开始，到 DMA 传输完成）

设置值仅在主寄存器反映，而不反映在从寄存器（从寄存器保持下一个 DMA 传输的设置值）。

但是，在 DMA 传输完成之后，主寄存器的内容自动覆盖从寄存器。

在此期间，如果读取寄存器的值，读到的是从寄存器的值。

要检查 DMA 传输已经启动，确认第一个传输已经通过读取 DBCn 寄存器（n = 0 至 3）而开始执行。

（3） 从 DMA 传输完成到产生下一次 DMA 传输请求的时期

设置值在主寄存器和从寄存器均有反映。

备注 “DMA 传输完成”表示有下列情况。
- DMA 传输结束（终止计数）。
- DMA 传输的强制终止（将 DCHCn.INITn 位设置为 1）。

19.9 DMA 传输启动因素

DMA 传输启动因素有两种，如下所示。

注意事项 **1.** **不要为同一个通道使用启动因素（（1）和（2））的组合（如果两个启动因素同时产生，只有一个有效，但是生效的启动因素无法识别）。**
如果两个启动因素组合使用，操作无法保证。
2. **如果 DMA 传输是通过软件启动，且软件没有通过 DCHCn.STGn 位的操作（设置为 1）正确检测到预期的 DMA 传输操作已经结束，则无法保证 STGn 位的下一个（第二个）操作是否对应启动“软件期待的下一个 DMA 传输”（n = 0 至 3）。**
比如，假设通过 STGn 位的操作启动单次传输。即使 STGn 位的下一次（第二次）操作没有通过软件去检查单次传输是否真正执行，则下一次（第二次）DMA 传输不是每次都可以执行。这是因为 STGn 位的第二次操作可能在第一个 DMA 传输开始或结束之前，因为，比如当第一次操作 STGn 位时，带有较高优先级的 DMA 传输已经启动。
因此，必需在检查通过第一次操作 STGn 位而启动的 DMA 传输操作是否已经结束之后，下一次（第二次）操作 STGn 位。
DMA 传输的结束可以通过检查 DBCn 寄存器的内容获知。

（1）软件请求

如果 DCHCn.STGn，DCHCn.Enn 和 DCHCn.TCn 位的设置如下，DMA 传输启动（n = 0 至 3）。

- STGn 位 = 1
- Enn 位 = 1
- TCn 位 = 0

（2）片上外围 I/O 请求

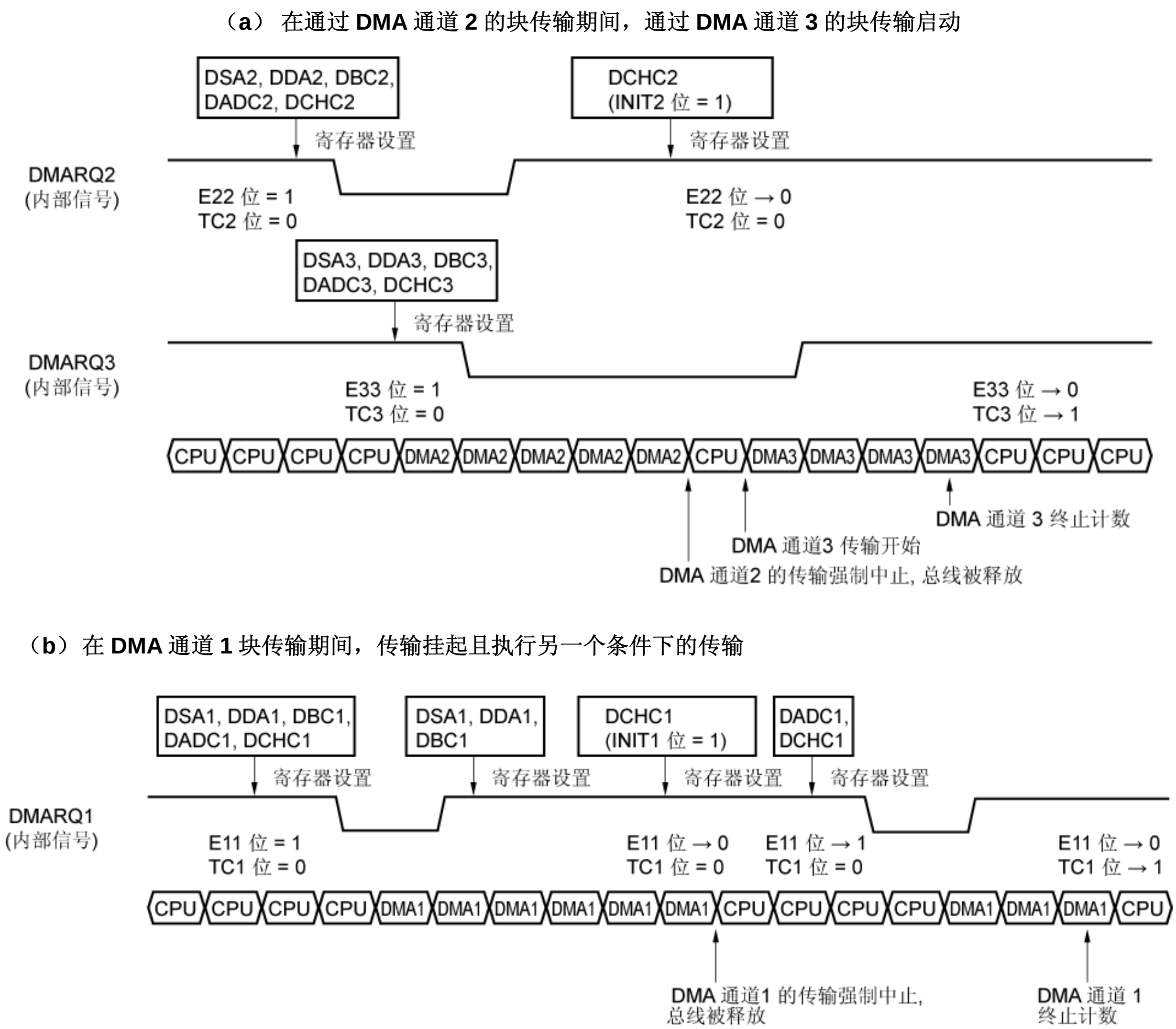
如果当 DCHCn.Enn 和 DCHCn.TCn 位的设置如下时，由 DTFRn 寄存器内设置的片上外围 I/O 产生一个中断请求，则 DMA 传输启动（n = 0 至 3）。

- Enn 位 = 1
- TCn 位 = 0

19.10 强制终止

通过 DCHCn.INITn 位可以强制终止 DMA 传输（n = 0 至 3）。
通过 DCHCn.INITn 位强制终止的示例如下（n = 0 至 3）。

图 19-9. DMA 传输强制终止的示例



备注 即使 DMA 传输强制终止，DSAn、DDAn 和 DBCn 寄存器（n = 0 至 3）的值也会保持，因为这些寄存器是 FIFO 格式的缓冲寄存器。即使 DMA 传输在进行中，下一个传输条件也可以设置到这些寄存器。也就是说，在 DMA 传输期间 DADCn 和 DCHCn 寄存器无效，因为这些寄存器不是缓冲寄存器（参见 **19.8 下一个地址设置功能**，**19.3.4 DMA 寻址控制寄存器 0 至 3**（**DADC0** 至 **DADC3**）和 **19.3.5 DMA 通道控制寄存器 0 至 3**（**DCHC0** 至 **DCHC3**））。

19.11 与 DMA 传输有关的时间

DMA 传输前后的所需时间和执行 DMA 传输需要的最少时钟数如下所示。

表 19-3. DMA 周期内最小执行时钟数量

DMA 周期		最少执行时钟数量
<1> DMA 请求响应时间		4 个时钟[注 1]
<2> 存储器访问	内部 RAM 访问	2 个时钟[注 2]
	片上外围 I/O 寄存器访问	4 个时钟＋VSWC 寄存器指定的等待周期数量

注 **1.** 如果指定一个外部中断（INTPn）作为启动 DMA 传输的触发因素，则加上噪声消除时间（n ＝ 11 至 13，15，17，18）.

2. 一个 DMA 周期需要两个时钟。

各个模式下 DMA 周期内最小执行时钟数量如下。

单次传输： DMA响应时间（<1>）＋ 传输源存储器访问（<2>）＋1[注]＋传输目的地存储器访问（<2>）

块传输： DMA响应时间（<1>）＋ 传输源存储器访问（<2>）＋1[注]＋传输目的地存储器访问（<2>）× 传输次数

注 DMA 传输的读周期和写周期之间总是插入一个时钟。

19.12 注意事项

（1）存储边界

在 DMA 传输过程中，如果传输源或目的地的地址超出了 DMA 目标（内部 RAM 或片上外围 I/O）的区域，则操作将无法保证。

（2）传输非对齐的数据

不支持采用 16 位总线宽度进行非对齐数据的 DMA 传输。如果奇地址指定为传输源或目的地，则强制处理地址的 LSB 为 0。

<R> **（3）CPU 的总线仲裁**

因为 DMA 控制器拥有比 CPU 更高优先级的总线掌控权限，在 DMA 传输期间，发生的 CPU 访问将保持挂起状态直到 DMA 传输周期完成，总线释放给 CPU 为止。

但是，CPU 可以访问未执行 DMA 传输的内部 ROM 和内部 RAM。

- 在片上外围 I/O 与内部 RAM 之间执行 DMA 传输时，CPU 可以访问内部 ROM 。

<R> **（4）DMA 启动因素**

当设置两个或更多 DMA 通道使用相同的启动因素时，要特别小心。

如果使用相同的启动因素开启了两个或更多的 DMA 通道，具有较低优先级的 DMA 通道将在具有较高优先级的 DMA 通道之前应答。

（5）在内部 RAM 执行程序并执行 DMA 传输

不要在内部 RAM 执行指令的同时，执行到/从内部 RAM 的 DMA 传输。

（6）设置 DCHCn.TCn 位的时序

在 DMA 传输结束时，DCHCn.TCn 位通常设置为 1。但是，在 DMA 传输从内部 RAM 开始的情况下，在 DMA 传输结束之后 4 个时钟，将其置位（n = 0 至 3）。

（7）读取 DSAn 和 DDAn 寄存器的值

在 DMA 传输过程中，如果读取 DSAn 和 DDAn 寄存器，读取到的可能是正在更新的值（n＝0 至 3）。

例如，当 DMA 传输源地址（DSAn 寄存器）为“0000FFFFH”，且计数方向是递增（DADCn 寄存器的 SAD1 和 SAD0 位＝00）时，如果先读取 DSAnH 寄存器，然后再读取 DSAnL 寄存器，则根据是否在读取 DSAnH 寄存器之后立即执行 DMA 传输，DSAnL 寄存器的值会有所不同，描述如下。

（a） 如果在读取 DSAn 寄存器时未发生 DMA 传输

<1> 读取 DSAnH 寄存器： DSAnH＝0000H

<2> 读取 DSAnL 寄存器： DSAnL＝FFFFH

（b） 如果在读取 DSAn 寄存器时发生 DMA 传输

<1> 读取 DSAnH 寄存器： DSAnH＝0000H

<2> 发生 DMA 传输

<3> DSAn 寄存器递增： DSAn＝00100000H

<4> 读取 DSAnL 寄存器： DSAnL＝0000H

（8）CLR1，NOT1 和 SET1 指令

在读取寄存器后写入 CLR1，NOT1 和 SET1 指令，再操作目标位。因此，要使用 SET1 指令来将 DCHCn.Enn 位设置为 1，当 DCHCn.TCn 位 = 1 时，TCn 位清除为 0（n = 0 至 3）。

19.13 DMA 传输结束

当 DMA 传输结束且 DCHCn.TCn 位设置为 1 时，中断控制器（INTC）将会发出 DMA 传输结束中断（INTDMAn）（n＝0 至 3）。

第二十章　中断/异常处理功能

V850E/IF3 和 V850E/IG3 为中断服务提供专用的中断控制器（INTC），可以处理总共 89 至 96 个中断请求。

所谓中断，是独立于程序执行而发生的事件，而异常则是与程序执行相关的事件。

V850E/IF3 和 V850E/IG3 能够处理来自于片上外围硬件设备和外部源的中断请求。不仅如此，还可以通过 TRAP 指令（软件异常）或产生一个异常事件（比如，取非法操作码）（异常陷阱）来启动异常处理。

20.1 特性

○ 中断

- 不可屏蔽中断：　　1 个源（外部：无，内部：1 个源）
- 可屏蔽中断（可屏蔽中断源的数量根据产品不同而有差别）
  V850E/IF3：　　88 个源（外部：15 个源，内部：73 个源）
  V850E/IG3：　　95 个源（外部：21 个源，内部：74 个源）
- 8 级可编程优先级（可屏蔽中断）
- 依照优先级控制多重中断
- 可以对每一个可屏蔽中断请求指定屏蔽
- 噪声消除、边沿检测以及为外部中断请求信号指定有效边沿

○ 异常

- 软件异常：　　32 个源
- 异常陷阱：　　2 源（非法操作码异常和调试陷阱）

中断源如表 20-1 所列。

表 20-1. 中断源列表（1/4）

类型	分类	中断/异常源				默认优先级	异常码	句柄地址	被恢复的 PC
		名称	控制寄存器	产生源	产生单元				
复位	中断	RESET	–	RESET 引脚输入	引脚	–	0000H	00000000H	不确定
			–	WDT 溢出（WDTRES）	WDT				
不可屏蔽	中断	INTWDT	–	WDT 溢出	WDT		0010H	00000010H	下一 PC 值
软件异常	异常	TRAP0n注 1	–	TRAP 指令	–	–	004nH	00000040H	下一 PC 值
	异常	TRAP1n注 1	–	TRAP 指令	–	–	005nH	00000050H	下一 PC 值
异常陷阱	异常	ILGOP/DBG0	–	非法指令码/DBTRAP 指令	–	–	0060H	00000060H	下一 PC 值
可屏蔽	中断	INTLVIL	LVILIC	LVI 低电平电压检测	LVI	0	0080H	00000080H	下一 PC 值
	中断	INTLVIH	LVIHIC	LVI 高电平电压检测	LVI	1	0090H	00000090H	下一 PC 值
	中断	INTP00	PIC00	INTP00 引脚有效边沿输入	引脚	2	00A0H	000000A0H	下一 PC 值
	中断	INTP01	PIC01	INTP01 引脚有效边沿输入	引脚	3	00B0H	000000B0H	下一 PC 值
	中断	INTP02注 2	PIC02注 2	INTP02 引脚有效边沿输入	引脚	4	00C0H	000000C0H	下一 PC 值
	中断	INTP03注 2	PIC03注 2	INTP03 引脚有效边沿输入	引脚	5	00D0H	000000D0H	下一 PC 值
	中断	INTP04注 2	PIC04注 2	INTP04 引脚有效边沿输入	引脚	6	00E0H	000000E0H	下一 PC 值
	中断	INTP05注 2	PIC05注 2	INTP05 引脚有效边沿输入	引脚	7	00F0H	000000F0H	下一 PC 值
	中断	INTP06注 2	PIC06注 2	INTP06 引脚有效边沿输入	引脚	8	0100H	00000100H	下一 PC 值
	中断	INTP07注 2	PIC07注 2	INTP07 引脚有效边沿输入	引脚	9	0110H	00000110H	下一 PC 值
	中断	INTP08	PIC08	INTP08 引脚有效边沿输入	引脚	10	0120H	00000120H	下一 PC 值
	中断	INTP09	PIC09	INTP09 引脚有效边沿输入	引脚	11	0130H	00000130H	下一 PC 值
	中断	INTP10	PIC10	INTP10 引脚有效边沿输入	引脚	12	0140H	00000140H	下一 PC 值
	中断	INTP11	PIC11	INTP11 引脚有效边沿输入	引脚	13	0150H	00000150H	下一 PC 值
	中断	INTP12	PIC12	INTP12 引脚有效边沿输入	引脚	14	0160H	00000160H	下一 PC 值
	中断	INTP13	PIC13	INTP13 引脚有效边沿输入	引脚	15	0170H	00000170H	下一 PC 值
	中断	INTP14	PIC14	INTP14 引脚有效边沿输入	引脚	16	0180H	00000180H	下一 PC 值
	中断	INTP15	PIC15	INTP15 引脚有效边沿输入	引脚	17	0190H	00000190H	下一 PC 值
	中断	INTP16	PIC16	INTP16 引脚有效边沿输入	引脚	18	01A0H	000001A0H	下一 PC 值
	中断	INTP17	PIC17	INTP17 引脚有效边沿输入	引脚	19	01B0H	000001B0H	下一 PC 值
	中断	INTP18	PIC18	INTP18 引脚有效边沿输入	引脚	20	01C0H	000001C0H	下一 PC 值
	中断	INTCMP0L	CMPIC0L	ADC0 过电压检测 L（比较器输出）	ADC0（比较器）	21	01D0H	000001D0H	下一 PC 值
	中断	INTCMP0F	CMPIC0F	ADC0 过电压检测 F（比较器输出）	ADC0（比较器）	22	01E0H	000001E0H	下一 PC 值
	中断	INTCMP1L	CMPIC1L	ADC1 过电压检测 L（比较器输出）	ADC1（比较器）	23	01F0H	000001F0H	下一 PC 值
	中断	INTCMP1F	CMPIC1F	ADC1 过电压检测 F（比较器输出）	ADC1（比较器）	24	0200H	00000200H	下一 PC 值

注 **1.** n = 0 至 FH

2. 仅限 V850E/IG3

表 20-1. 中断源列表（2/4）

类型	分类	中断/异常源				默认优先级	异常码	句柄地址	被恢复的 PC
		名称	控制寄存器	产生源	产生单元				
可屏蔽	中断	INTTB0OV	TB0OVIC	TAB0 溢出注 2	TAB0	25	0210H	00000210H	下一 PC 值
	中断	INTTB0CC0	TB0CCIC0	TAB0CCR0 捕获输入/比较一致注 3	TAB0	26	0220H	00000220H	下一 PC 值
	中断	INTTB0CC1	TB0CCIC1	TAB0CCR1 捕获输入/比较一致	TAB0	27	0230H	00000230H	下一 PC 值
	中断	INTTB0CC2	TB0CCIC2	TAB0CCR2 捕获输入/比较一致	TAB0	28	0240H	00000240H	下一 PC 值
	中断	INTTB0CC3	TB0CCIC3	TAB0CCR3 捕获输入/比较一致	TAB0	29	0250H	00000250H	下一 PC 值
	中断	INTTB1OV	TB1OVIC	TAB1 溢出注 2	TAB1	30	0260H	00000260H	下一 PC 值
	中断	INTTB1CC0	TB1CCIC0	TAB1CCR0 捕获输入/比较一致注 3	TAB1	31	0270H	00000270H	下一 PC 值
	中断	INTTB1CC1	TB1CCIC1	TAB1CCR1 捕获输入/比较一致	TAB1	32	0280H	00000280H	下一 PC 值
	中断	INTTB1CC2	TB1CCIC2	TAB1CCR2 捕获输入/比较一致	TAB1	33	0290H	00000290H	下一 PC 值
	中断	INTTB1CC3	TB1CCIC3	TAB1CCR3 捕获输入/比较一致	TAB1	34	02A0H	000002A0H	下一 PC 值
	中断	INTTTIOV0	TT0OVIC	TMT0 溢出	TMT0	35	02B0H	000002B0H	下一 PC 值
	中断	INTTTEQC00	TT0CCIC0	TT0CCR0 捕获输入注 4/比较一致	TMT0	36	02C0H	000002C0H	下一 PC 值
	中断	INTTTEQC01	TT0CCIC1	TT0CCR1 捕获输入注 4/比较一致	TMT0	37	02D0H	000002D0H	下一 PC 值
	中断	INTTIEC0注 1	TT0IECIC注 1	编码器输入 中断 0	TMT0	38	02E0H	000002E0H	下一 PC 值
	中断	INTTTIOV1	TT1OVIC	TMT1 溢出	TMT1	39	02F0H	000002F0H	下一 PC 值
	中断	INTTTEQC10	TT1CCIC0	TT1CCR0 捕获输入/比较一致	TMT1	40	0300H	00000300H	下一 PC 值
	中断	INTTTEQC11	TT1CCIC1	TT1CCR1 捕获输入/比较一致	TMT1	41	0310H	00000310H	下一 PC 值
	中断	INTTIEC1	TT1IECIC	编码器输入 中断 1	TMT1	42	0320H	00000320H	下一 PC 值
	中断	INTTA0OV	TA0OVIC	TAA0 溢出	TAA0	43	0330H	00000330H	下一 PC 值

注 **1.** 仅限 V850E/IG3

2. 在 6 相 PWM 输出模式下使用 TABm 时，作为来自 TMQm 选项（TMQOPm）的 INTTBmOV（波谷中断）（m = 0，1）。

3. 在 6 相 PWM 输出模式下使用 TABm 时，作为来自 TMQm 选项（TMQOPm）的 INTTBmCC0（波峰中断）（m = 0，1）。

4. 仅限 V850E/IG3

在 V850E/IF3 内，只有比较一致

表 20-1. 中断源列表（3/4）

类型	分类	中断/异常源				默认优先级	异常码	句柄地址	被恢复的 PC
		名称	控制寄存器	产生源	产生单元				
可屏蔽	中断	INTTA0CC0	TA0CCIC0	TA0CCR0 比较一致	TAA0	44	0340H	00000340H	下一 PC 值
	中断	INTTA0CC1	TA0CCIC1	TA0CCR1 比较一致	TAA0	45	0350H	00000350H	下一 PC 值
	中断	INTTA1OV	TA1OVIC	TAA1 溢出	TAA1	46	0360H	00000360H	下一 PC 值
	中断	INTTA1CC0	TA1CCIC0	TA1CCR0 比较一致	TAA1	47	0370H	00000370H	下一 PC 值
	中断	INTTA1CC1	TA1CCIC1	TA1CCR1 比较一致	TAA1	48	0380H	00000380H	下一 PC 值
	中断	INTTA2OV	TA2OVIC	TAA2 溢出	TAA2	49	0390H	00000390H	下一 PC 值
	中断	INTTA2CC0	TA2CCIC0	TA2CCR0 捕获输入/比较一致	TAA2	50	03A0H	000003A0H	下一 PC 值
	中断	INTTA2CC1	TA2CCIC1	TA2CCR1 捕获输入/比较一致	TAA2	51	03B0H	000003B0H	下一 PC 值
	中断	INTTA3OV	TA3OVIC	TAA3 溢出	TAA3	52	03C0H	000003C0H	下一 PC 值
	中断	INTTA3CC0	TA3CCIC0	TA3CCR0 捕获输入[注]/比较一致	TAA3	53	03D0H	000003D0H	下一 PC 值
	中断	INTTA3CC1	TA3CCIC1	TA3CCR1 捕获输入[注]/比较一致	TAA3	54	03E0H	000003E0H	下一 PC 值
	中断	INTTA4OV	TA4OVIC	TAA4 溢出	TAA4	55	03F0H	000003F0H	下一 PC 值
	中断	INTTA4CC0	TA4CCIC0	TA4CCR0 捕获输入/比较一致	TAA4	56	0400H	00000400H	下一 PC 值
	中断	INTTA4CC1	TA4CCIC1	TA4CCR1 捕获输入/比较一致	TAA4	57	0410H	00000410H	下一 PC 值
	中断	INTDMA0	DMAIC0	DMA 通道 0 传输结束	DMA0	58	0420H	00000420H	下一 PC 值
	中断	INTDMA1	DMAIC1	DMA 通道 1 传输结束	DMA1	59	0430H	00000430H	下一 PC 值
	中断	INTDMA2	DMAIC2	DMA 通道 2 传输结束	DMA2	60	0440H	00000440H	下一 PC 值
	中断	INTDMA3	DMAIC3	DMA 通道 3 传输结束	DMA3	61	0450H	00000450H	下一 PC 值
	中断	INTUBTIRE	UREIC	UARTB 接收错误	UARTB	62	0460H	00000460H	下一 PC 值
	中断	INTUBTIR	URIC	UARTB 接收 结束	UARTB	63	0470H	00000470H	下一 PC 值
	中断	INTUBTIT	UTIC	UARTB 发送使能	UARTB	64	0480H	00000480H	下一 PC 值
	中断	INTUBTIF	UIFIC	UARTB FIFO 发送结束	UARTB	65	0490H	00000490H	下一 PC 值
	中断	INTUBTITO	UTOIC	UARTB 接收超时	UARTB	66	04A0H	000004A0H	下一 PC 值
	中断	INTUA0RE	UA0REIC	UARTA0 接收错误	UARTA0	67	04B0H	000004B0H	下一 PC 值
	中断	INTUA0R	UA0RIC	UARTA0 接收 结束	UARTA0	68	04C0H	000004C0H	下一 PC 值
	中断	INTUA0T	UA0TIC	UARTA0 发送使能	UARTA0	69	04D0H	000004D0H	下一 PC 值
	中断	INTCB0RE	CB0REIC	CSIB0 接收错误	CSIB0	70	04E0H	000004E0H	下一 PC 值
	中断	INTCB0R	CB0RIC	CSIB0 接收 结束	CSIB0	71	04F0H	000004F0H	下一 PC 值
	中断	INTCB0T	CB0TIC	CSIB0 发送使能	CSIB0	72	0500H	00000500H	下一 PC 值
	中断	INTUA1RE	UA1REIC	UARTA1 接收错误	UARTA1	73	0510H	00000510H	下一 PC 值
	中断	INTUA1R	UA1RIC	UARTA1 接收 结束	UARTA1	74	0520H	00000520H	下一 PC 值
	中断	INTUA1T	UA1TIC	UARTA1 发送使能	UARTA1	75	0530H	00000530H	下一 PC 值

注　仅限 V850E/IG3
在 V850E/IF3 内，仅有比较一致

表 20-1. 中断源列表（4/4）

类型	分类	中断/异常源				默认优先级	异常码	句柄地址	被恢复的 PC
		名称	控制寄存器	产生源	产生单元				
可屏蔽	中断	INTCB1RE	CB1REIC	CSIB1 接收错误	CSIB1	76	0540H	00000540H	下一 PC 值
	中断	INTCB1R	CB1RIC	CSIB1 接收 结束	CSIB1	77	0550H	00000550H	下一 PC 值
	中断	INTCB1T	CB1TIC	CSIB1 发送使能	CSIB1	78	0560H	00000560H	下一 PC 值
	中断	INTUA2RE	UA2REIC	UARTA2 接收错误	UARTA2	79	0570H	00000570H	下一 PC 值
	中断	INTUA2R	UA2RIC	UARTA2 接收 结束	UARTA2	80	0580H	00000580H	下一 PC 值
	中断	INTUA2T	UA2TIC	UARTA2 发送使能	UARTA2	81	0590H	00000590H	下一 PC 值
	中断	INTCB2RE	CB2REIC	CSIB2 接收错误	CSIB2	82	05A0H	000005A0H	下一 PC 值
	中断	INTCB2R	CB2RIC	CSIB2 接收 结束	CSIB2	83	05B0H	000005B0H	下一 PC 值
	中断	INTCB2T	CB2TIC	CSIB2 发送使能	CSIB2	84	05C0H	000005C0H	下一 PC 值
	中断	INTIIC	IICIC	IIC 串行传输结束	IIC	85	05D0H	000005D0H	下一 PC 值
	中断	INTAD0	AD0IC	ADC0 转换结束	ADC0	86	05E0H	000005E0H	下一 PC 值
	中断	INTAD1	AD1IC	ADC1 转换结束	ADC1	87	05F0H	000005F0H	下一 PC 值
	中断	INTAD2	AD2IC	ADC2 转换结束	ADC2	88	0600H	00000600H	下一 PC 值
	中断	INTTM0EQ0	TM0EQIC0	TM0CMP0 比较一致	TMM0	89	0610H	00000610H	下一 PC 值
	中断	INTTM1EQ0	TM1EQIC0	TM0CMP1 比较一致	TMM1	90	0620H	00000620H	下一 PC 值
	中断	INTTM2EQ0	TM2EQIC0	TM0CMP2 比较一致	TMM2	91	0630H	00000630H	下一 PC 值
	中断	INTTM3EQ0	TM3EQIC0	TM0CMP3 比较一致	TMM3	92	0640H	00000640H	下一 PC 值
	中断	INTADT0	ADT0IC	ADTRG0 引脚有效边沿输入	引脚	93	0650H	00000650H	下一 PC 值
	中断	INTADT1	ADT1IC	ADTRG1 引脚有效边沿输入	引脚	94	0660H	00000660H	下一 PC 值

备注

1. 默认优先级：当两个或两个以上可屏蔽中断请求同时发生时，系统指定的优先级次序。最高优先级是 0。

被恢复的 PC：当中断服务开始时，程序计数器（PC）的值被存入 CPU 的 EIPC、FEPC 或 DBPC。但是，需要注意的是：在执行以下任意一条指令时，如果程序响应一个不可屏蔽中断或可屏蔽中断，则被恢复的 PC 不是下一 PC 值。（如果在中断执行期间响应一个中断，执行停止，之后在中断服务完成后恢复执行。这种情况下，被放弃指令的地址是恢复的 PC 值）

- 装载指令（SLD.B，SLD.BU，SLD.H，SLD.HU，SLD.W）。
- 除法指令（DIV，DIVH，DIVU，DIVHU）。
- PREPARE，DISPOSE 指令（仅当在堆栈指针被更新前产生中断）。

下一 PC 值：在中断/异常处理之后，由此开始进行处理的 PC 值。

2. 当发生非法操作码异常时，非法指令的执行地址计算方法为：（被恢复的 PC－4）。

20.2 不可屏蔽中断

即便处于中断禁止（DI）状态，不可屏蔽中断请求信号也可以无条件地得到响应。不可屏蔽中断（NMI）并不遵从优先级控制，而且，比所有其他中断请求信号更优先。

V850E/IF3 和 V850E/IG3 的不可屏蔽中断请求信号是看门狗定时器（INTWDT）的溢出产生的不可屏蔽中断请求信号。

当 WDTM.WDM1 和 WDTM.WDM0 位被设置为“01”时，INTWDT 生效。

20.2.1 操作

如果产生一个不可屏蔽中断请求信号（INTWDT），则 CPU 执行以下处理过程，并将控制权移交给句柄程序。

<1> 将待恢复的 PC 保存至 FEPC。
<2> 将当前 PSW 保存至 FEPSW。
<3> 将异常码（0010H）写入 ECR 的高半字（FECC）。
<4> 将 PSW.NP 和 PSW.ID 位置位（1），并将 PSW.EP 位清除（0）。
<5> 将不可屏蔽中断相应的句柄地址 （00000010H）载入 PC，并转移控制。

不可屏蔽中断的服务流程如下所示：

图 20-1. 不可屏蔽中断服务

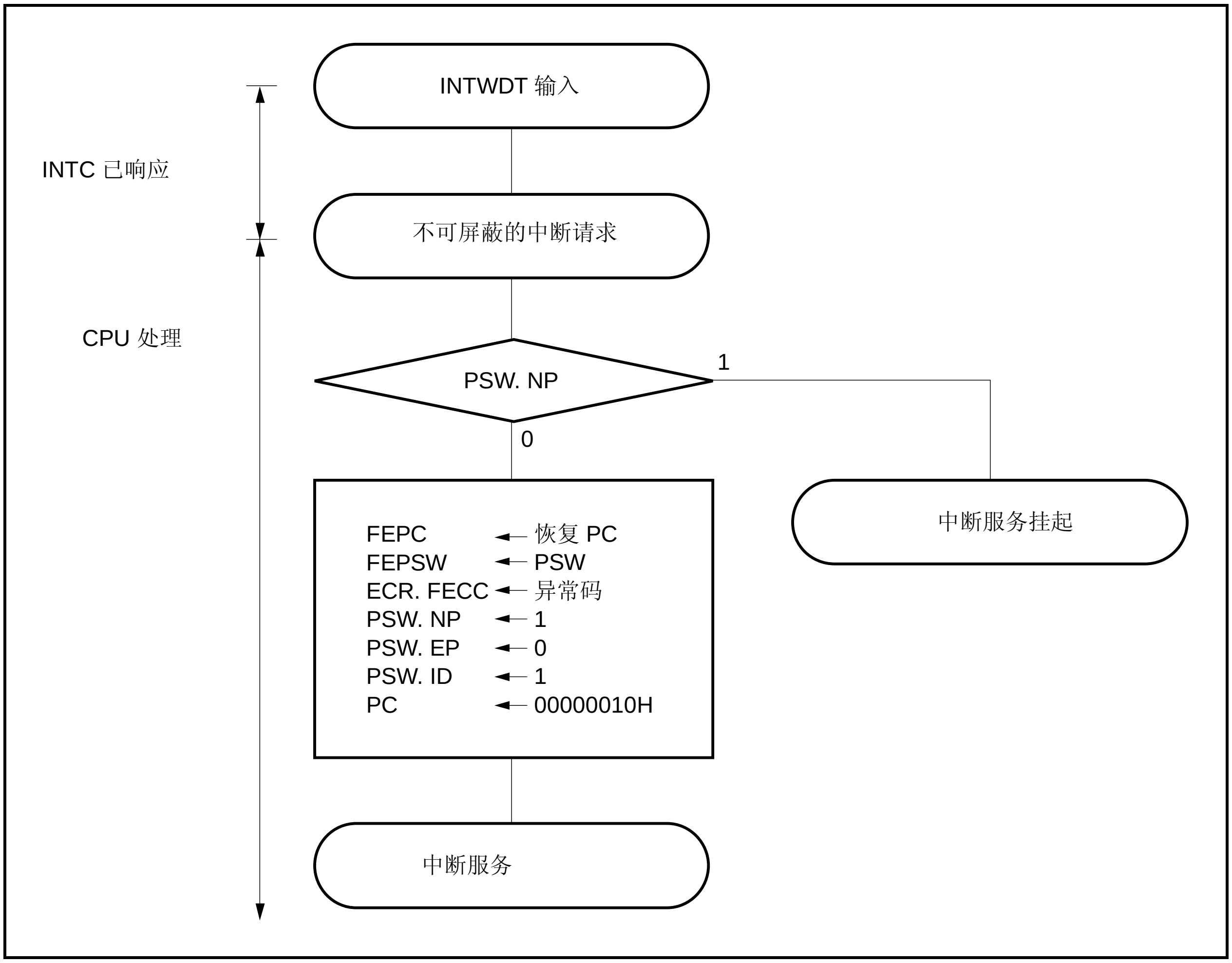
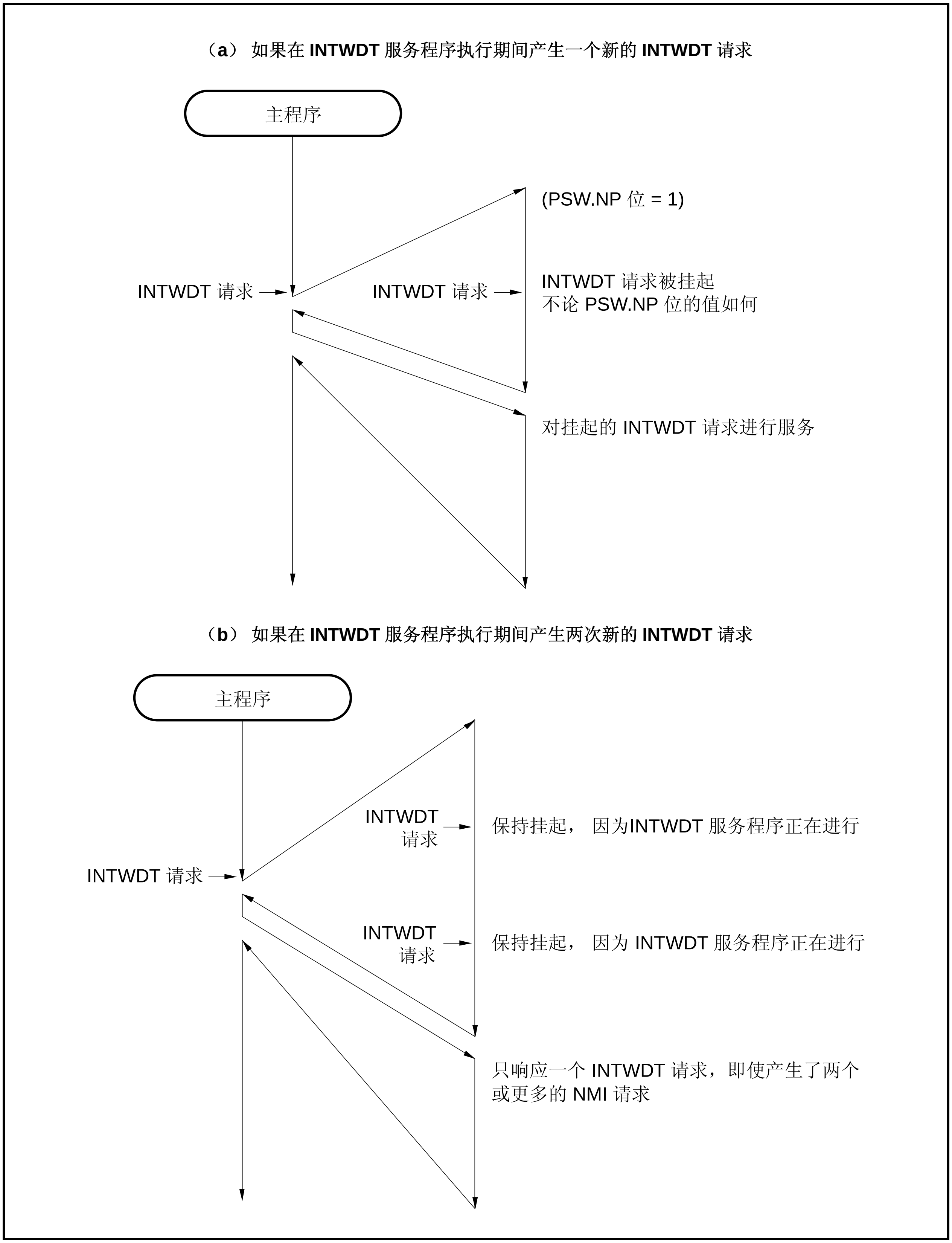


图 **20-2.** 不可屏蔽中断请求信号的响应



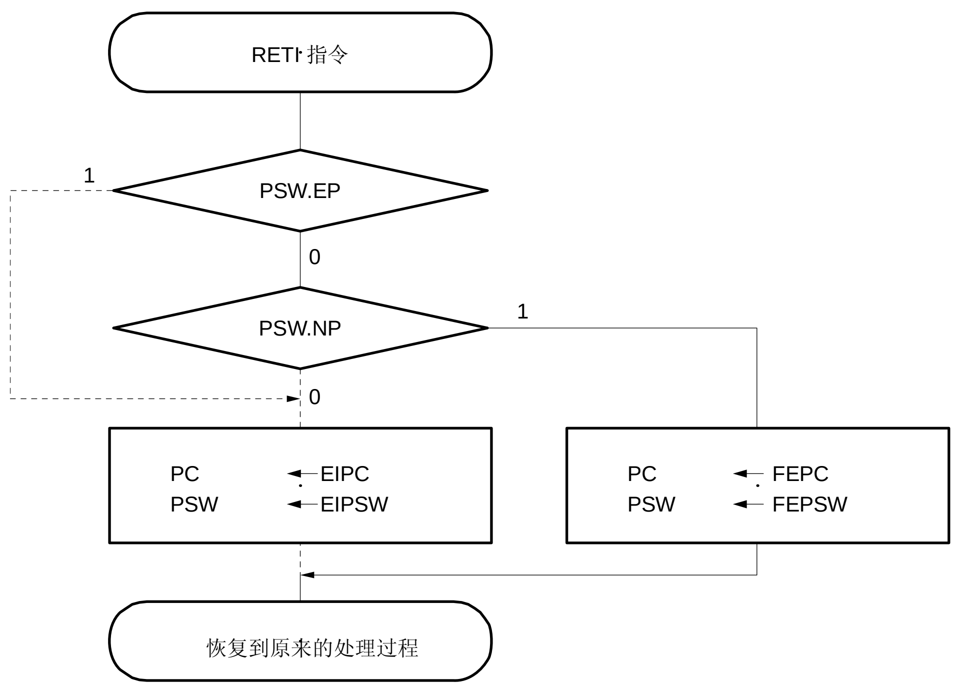
20.2.2 恢复

通过 RETI 指令，从不可屏蔽中断服务的恢复处理。
当执行 RETI 指令时，CPU 执行以下处理过程，并且控制转移到被恢复的 PC 地址。

<1> 因为 PSW.EP 位是 0 且 PSW.NP 位是 1，所以，由 FEPC 和 FEPSW 载入相应的 PC 和 PSW 的恢复值。
<2> 控制转移返回到被恢复的 PC 和 PSW 的地址。

下面展示如何处理 RETI 指令的过程。

图 20-3. RETI 指令处理



注意事项　**在不可屏蔽中断服务期间，如果 EP 位和 NP 位被 LDSR 指令改变时，为了在 RETI 恢复期间正确恢复 PC 和 PSW，必须在紧邻 RETI 指令之前，使用 LDSR 指令将 EP 位设置为 0，并将 NP 位置回 1。**

备注　实线显示的是 CPU 的处理流程。

20.2.3 不可屏蔽中断状态标志(NP)

NP 标志是一个状态标志，用于指示不可屏蔽中断（INTWDT）服务正在进行。NP 标志分配在 PSW 内。
当 INTWDT 中断请求信号被响应之后，该标志置位，并且，屏蔽所有中断请求和异常，禁止响应多重中断。
系统复位后，该标志被清除为 00000020H。

复位后： 00000020H

	31 ... 8	7	6	5	4	3	2	1	0
PSW	0	NP	EP	ID	SAT	CY	OV	S	Z

NP	不可屏蔽中断 (INTWDT) 服务状态
0	没有不可屏蔽中断在服务
1	不可屏蔽中断正在处理中

20.3 可屏蔽中断

可屏蔽中断请求信号可以由中断控制寄存器进行屏蔽。V850E/IF3 和 V850E/IG3 单片机有 95 个可屏蔽中断源。

如果同时发生两个或更多的可屏蔽中断请求信号，根据默认的优先级进行响应。除了默认优先级之外，使用中断控制寄存器（可编程优先级控制）可以指定 8 个优先等级。

当一个中断请求信号已经得到响应，则禁止响应其他可屏蔽中断请求信号，并且中断禁止（DI）状态置位。

当在中断服务程序中执行 EI 指令，中断使能（EI）状态置位，可以使能那些比正在处理的中断请求信号（由中断控制寄存器指定）具有更高的优先级的中断服务。注意，只有具有较高优先级的中断才有这种能力；具有相同优先等级的中断不能用作多重中断。

但是，为了允许多重中断（中断嵌套）服务，在执行 EI 指令之前有必要将 EIPC 和 EIPSW 保存到存储器或寄存器中，并在使用 RETI 指令之前执行 DI 指令，以恢复 EIPC 和 EIPSW 原始值。

20.3.1 操作

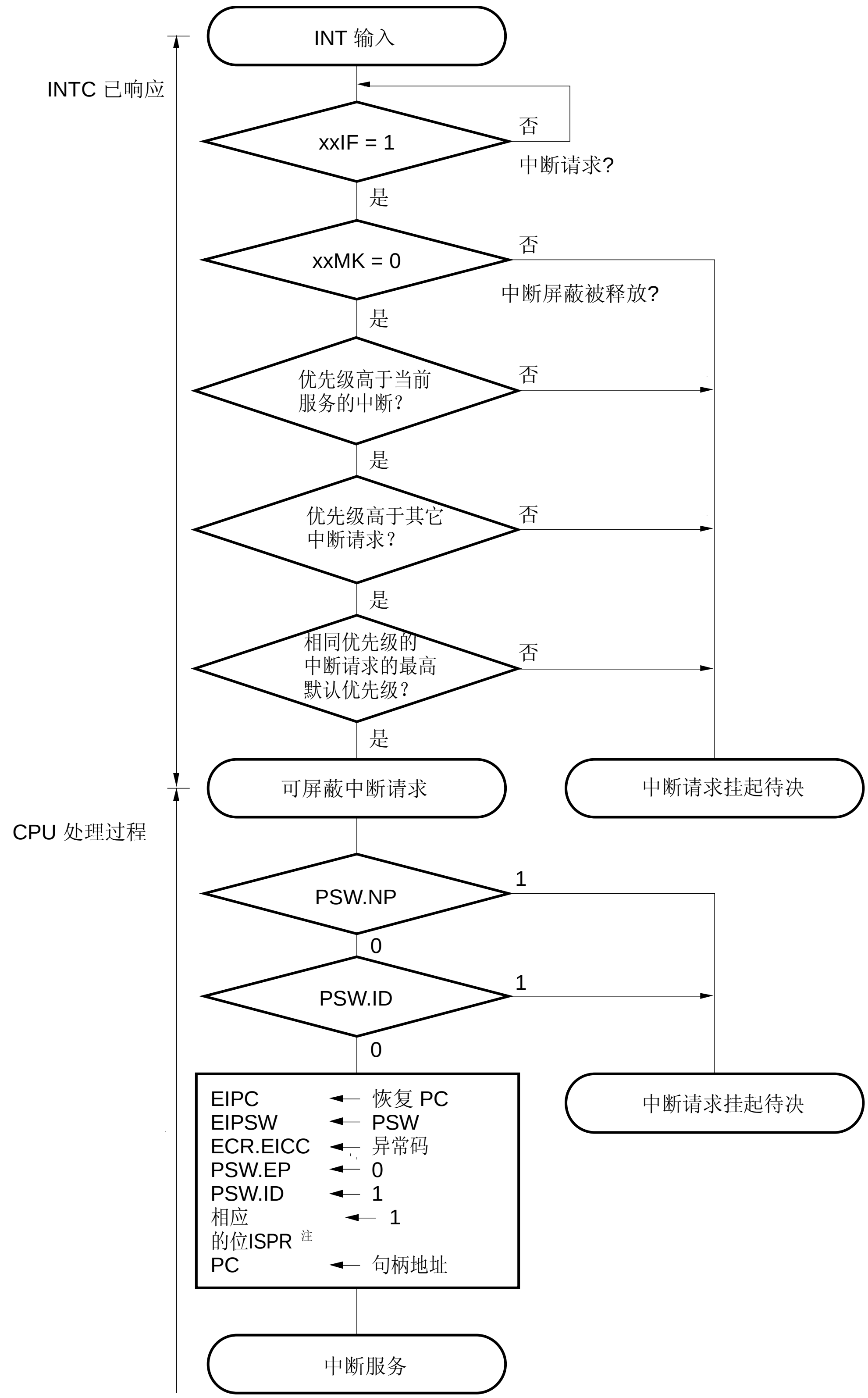
如果发生可屏蔽中断，CPU 执行以下处理过程，并控制转移至句柄程序。

<1> 将 PC 保存至 EIPC。
<2> 将 PSW 保存至 EIPSW。
<3> 将异常码写入 ECR（EICC）的低半字。
<4> 对 PSW.ID 位置 1，并对 PSW.EP 位清 0。
<5> 将各个中断对应的处理程序地址设置到 PC 指针，并进行控制转移。

由中断控制寄存器（INTC）屏蔽的可屏蔽中断请求信号，和在另外的中断服务处理时（当 PSW.NP 位＝1 或 ID 位＝1）产生的可屏蔽中断请求信号，都被保持挂起状态存放在 INTC 内。这种情况下，如果取消对可屏蔽中断的屏蔽或使用 RETI 或 LDSR 指令将 NP 或 ID 清除为 0，则根据被挂起的可屏蔽中断请求的优先级判断，来启动一个新的可屏蔽中断服务。

下图显示了可屏蔽中断的服务过程：

图 **20-4.** 可屏蔽中断服务



注　　ISPR 寄存器的详情，参见 **20.3.6　服务中优先级寄存器（ISPR）**。

20.3.2 恢复

执行 RETI 指令，可以从可屏蔽中断服务中恢复。

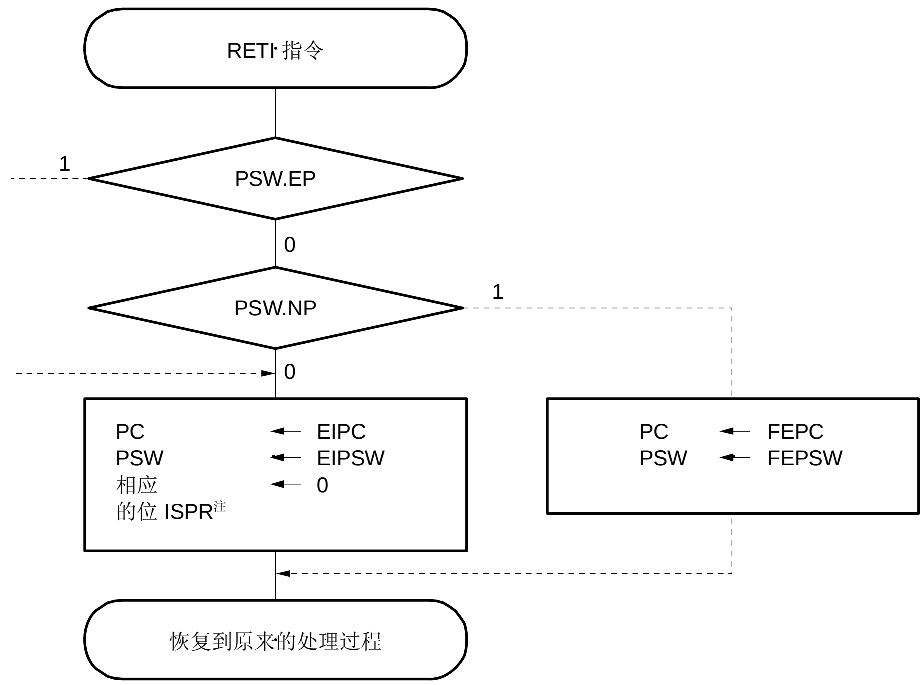
当执行 RETI 指令时，CPU 执行以下步骤，并且控制移交给被恢复的 PC 地址。

<1> 因为 PSW.EP 位为 0 且 PSW.NP 位为 0，所以由 EIPC 和 EIPSW 载入 PC 和 PSW 的值。

<2> 控制转移到被恢复的 PC 和 PSW 的地址。

RETI 指令的处理过程如下所示：

图 20-5. RETI 指令处理



注　　关于 ISPR 寄存器的详情，参见 **20.3.6 服务中优先级寄存器（ISPR）**。

注意事项　在可屏蔽中断服务期间，如果 EP 位和 NP 位被 LDSR 指令改变时，为了在 RETI 指令恢复期间正确恢复 PC 和 PSW，必须在紧邻 RETI 指令之前，使用 LDSR 指令将 EP 位和 NP 位都清为 0。

备注　　实线显示的是 CPU 的处理流程。

20.3.3 可屏蔽中断的优先级

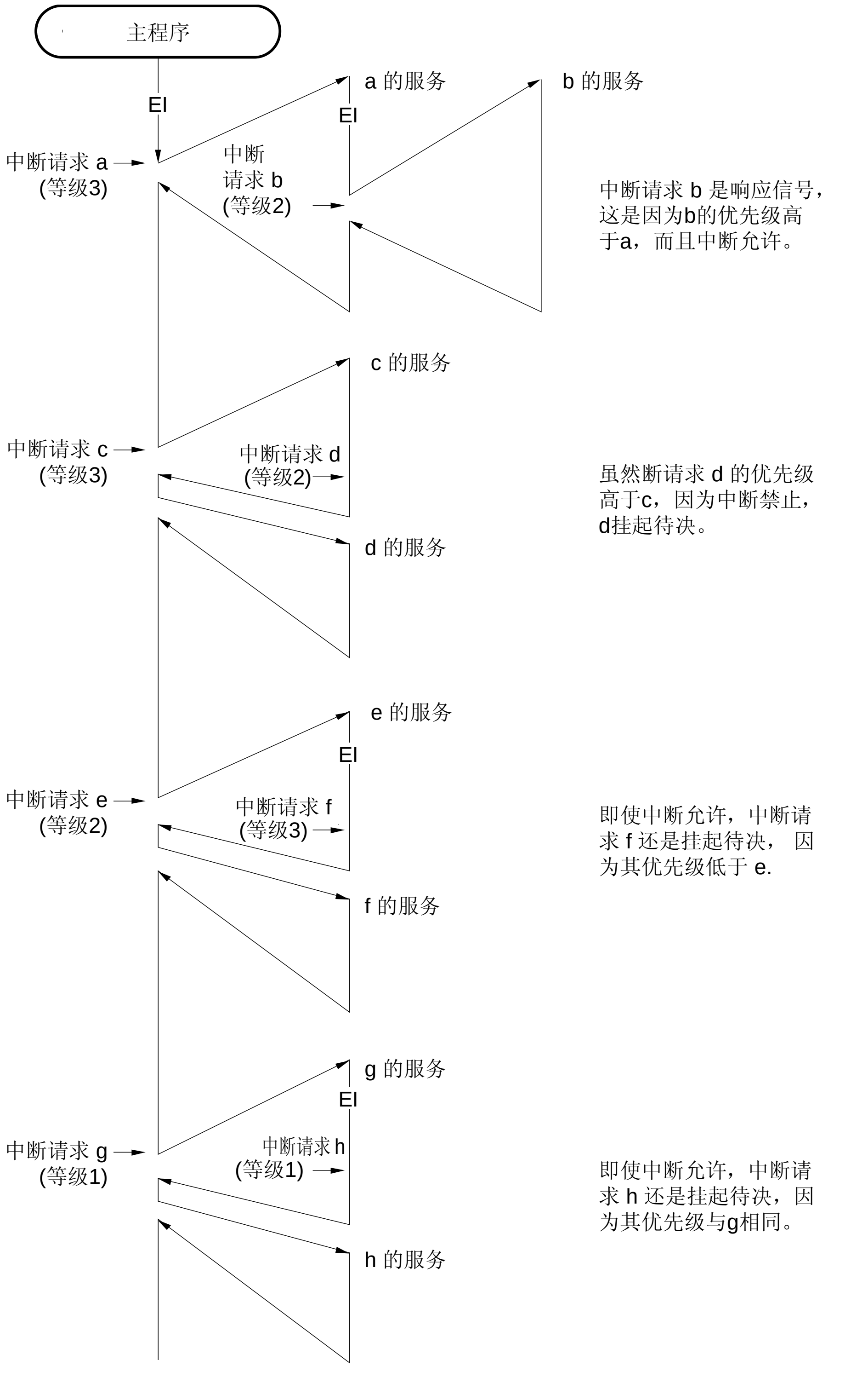
INTC 提供多重中断服务，也就是在一个中断正在处理服务时响应另一个中断。多重中断可以通过优先等级进行控制。

有两种类型的优先级控制方式：基于默认优先级的控制，以及基于中断控制寄存器（xxICn）的中断优先级指定位（xxPRn）确定的可编程优先级控制。当两个或两个以上中断同时发生，且 xxPRn 位指定的优先级都相同时，按照每个中断请求类型预先设定（默认优先级）的优先级次序进行处理。更多信息参见 **表 20-1 中断源列表**。通过设置优先级标志，可编程优先级控制将中断请求信号划分为 8 个等级。

注意，当中断请求信号被响应时，PSW.ID 标志自动被置 1。所以，当使用多重中断时，应该预先将 ID 标志清为 0（比如，在中断服务程序中插入一条 EI 指令）来设置多重中断允许模式。

备注 xx： 每一个外围单元的标志名称（参见 **表 20-2**）。
n： 外围单元编号（参见**表 20-2**）。

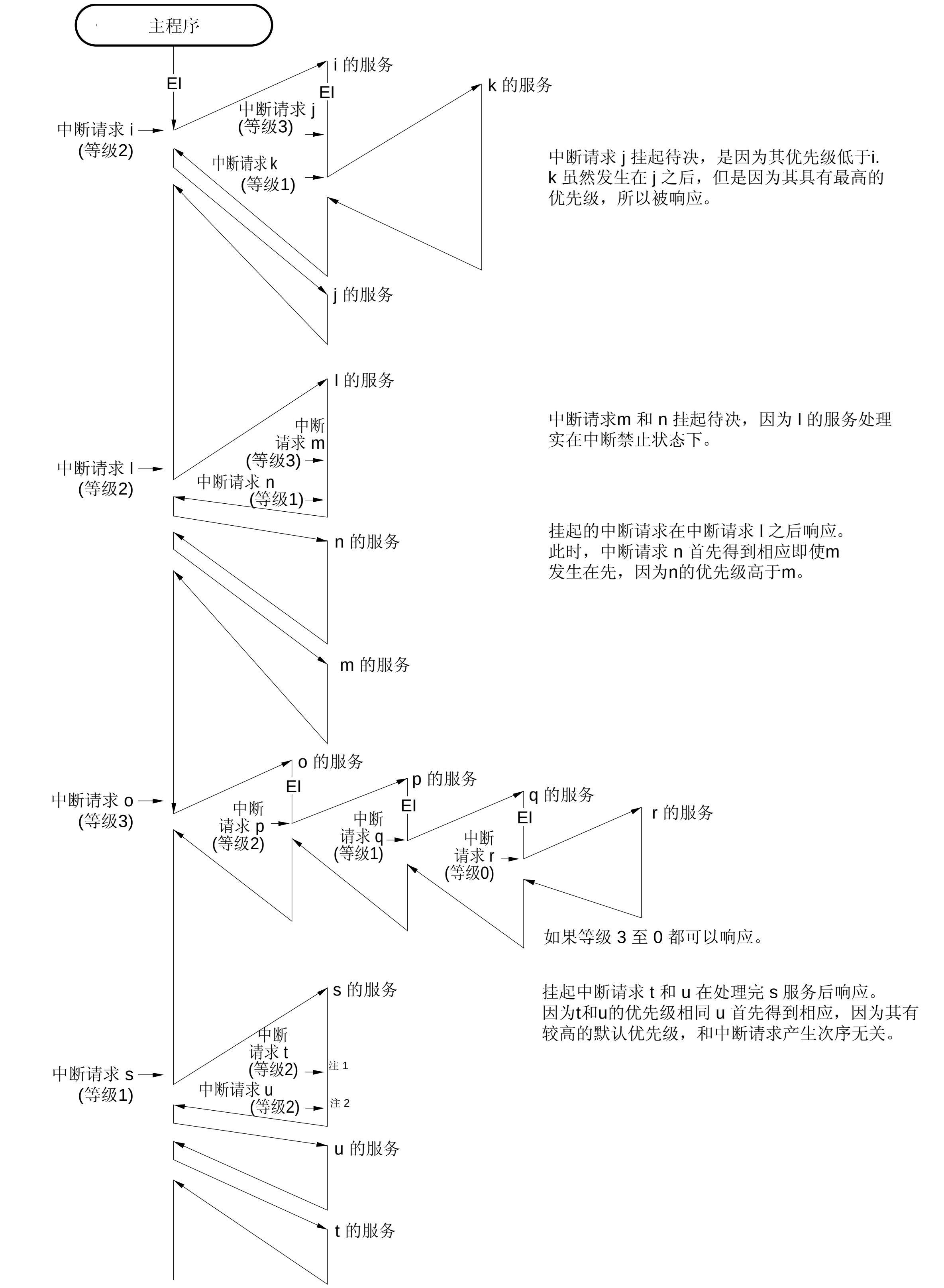
图 20-6. 当一个中断服务时发生另一个中断请求的处理示例（1/2）



注意事项 **为了执行多重中断服务，EIPC 和 EIPSW 寄存器的值必须在执行 EI 指令前进行保存。当从多重中断服务中返回时，在执行 DI 指令之后，恢复 EIPC 和 EIPSW 寄存器的值。**

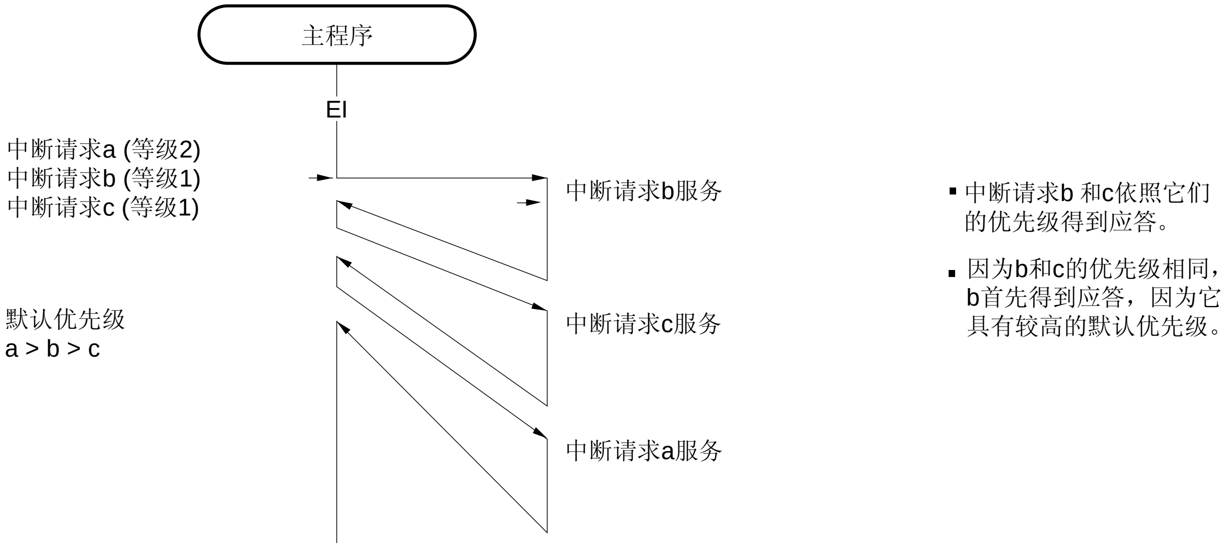
备注
1. 为了解释说明，上图中 a 至 u 是中断请求信号的暂用名。
2. 上图的默认优先级表明两个中断请求信号之间的相对优先级。

图 **20-6.** 一个中断服务当中而另一个中断请求发生的处理举例（**2/2**）



注意事项　为了执行多重中断服务，**EIPC** 和 **EIPSW** 寄存器的值必须在执行 **EI** 指令前进行保存。当从多重中断服务中返回时，在执行 **DI** 指令之后，恢复 **EIPC** 和 **EIPSW** 寄存器的值。

图 **20-7.** 同时产生的中断请求信号服务举例



注意事项 为了执行多重中断服务，**EIPC** 和 **EIPSW** 寄存器的值必须在执行 **EI** 指令前进行保存。当从多重中断服务中返回时，在执行 **DI** 指令之后，恢复 **EIPC** 和 **EIPSW** 寄存器的值。

备注 **1.** 为了解释说明，图中的 a 至 c 是中断请求信号的假设名称。

2. 图中的默认优先级表明两个中断请求信号之间的相对优先级。

20.3.4 中断控制寄存器（xxICn）

每一个可屏蔽中断请求信号（可屏蔽中断）都被指定一个 xxICn 寄存器，用来设置每一个可屏蔽中断请求的控制条件。

该寄存器可以按字节或按位进行读取或写入。

系统复位后，寄存器被置为 47H。

注意事项 **读取 xxICn.xxIFn 位时，禁止中断（DI）。如果在中断允许（EI）时的情况下读取 xxIFn 位，当响应一个中断与读取该位的时序发生冲突时，可能读不出正确值。**

<R> **2. 当中断请求可能发生时，操作 xxICn.xxMKn 位（包括中断被禁止（DI）的状态），请确保使用位操作指令或使用 IMRm.xxMKn 位（m = 0 至 5）。**

复位后： 47H R/W 地址： FFFFF110H 至 FFFFF1CCH

	<7>	<6>	5	4	3	2	1	0
xxICn	xxIFn	xxMKn	0	0	0	xxPRn2	xxPRn1	xxPRn0

xxIFn	中断请求标志[注]
0	不产生中断请求
1	产生中断请求

xxMKn	中断屏蔽标志
0	中断服务使能
1	中断服务禁止 (挂起待决)

xxPRn2	xxPRn1	xxPRn0	中断优先级指定位
0	0	0	指定等级 0 (最高).
0	0	1	指定等级 1
0	1	0	指定等级 2
0	1	1	指定等级 3
1	0	0	指定等级 4
1	0	1	指定等级 5
1	1	0	指定等级 6
1	1	1	指定等级 7 (最低)

注 如果一个中断请求信号得到响应，xxIFn 标志由硬件自动复位。

备注 xx： 每一个外围单元的标志名称（参见 表 **20-2**）。
n ： 外围单元编号（参见表 **20-2**）。

中断控制寄存器的地址及位如下所示：

表 **20-2.** 中断控制寄存器的地址和位（**1/3**）

地址	寄存器	位							
		<7>	<6>	5	4	3	2	1	0
FFFFF110H	LVILIC	LVILIF	LVILMK	0	0	0	LVILPR2	LVILPR1	LVILPR0
FFFFF112H	LVIHIC	LVIHIF	LVIHMK	0	0	0	LVIHPR2	LVIHPR1	LVIHPR0
FFFFF114H	PIC00	PIF00	PMK00	0	0	0	PPR002	PPR001	PPR000
FFFFF116H	PIC01	PIF01	PMK01	0	0	0	PPR012	PPR011	PPR010
FFFFF118H	PIC02[注]	PIF02	PMK02	0	0	0	PPR022	PPR021	PPR020
FFFFF11AH	PIC03[注]	PIF03	PMK03	0	0	0	PPR032	PPR031	PPR030
FFFFF11CH	PIC04[注]	PIF04	PMK04	0	0	0	PPR042	PPR041	PPR040
FFFFF11EH	PIC05[注]	PIF05	PMK05	0	0	0	PPR052	PPR051	PPR050
FFFFF120H	PIC06[注]	PIF06	PMK06	0	0	0	PPR062	PPR061	PPR060
FFFFF122H	PIC07[注]	PIF07	PMK07	0	0	0	PPR072	PPR071	PPR070
FFFFF124H	PIC08	PIF08	PMK08	0	0	0	PPR082	PPR081	PPR080
FFFFF126H	PIC09	PIF09	PMK09	0	0	0	PPR092	PPR091	PPR090
FFFFF128H	PIC10	PIF10	PMK10	0	0	0	PPR102	PPR101	PPR100
FFFFF12AH	PIC11	PIF11	PMK11	0	0	0	PPR112	PPR111	PPR110
FFFFF12CH	PIC12	PIF12	PMK12	0	0	0	PPR122	PPR121	PPR120
FFFFF12EH	PIC13	PIF13	PMK13	0	0	0	PPR132	PPR131	PPR130
FFFFF130H	PIC14	PIF14	PMK14	0	0	0	PPR142	PPR141	PPR140
FFFFF132H	PIC15	PIF15	PMK15	0	0	0	PPR152	PPR151	PPR150
FFFFF134H	PIC16	PIF16	PMK16	0	0	0	PPR162	PPR161	PPR160
FFFFF136H	PIC17	PIF17	PMK17	0	0	0	PPR172	PPR171	PPR170
FFFFF138H	PIC18	PIF18	PMK18	0	0	0	PPR182	PPR181	PPR180
FFFFF13AH	CMPIC0L	CMPIF0L	CMPMK0L	0	0	0	CMPPR0L2	CMPPR0L1	CMPPR0L0
FFFFF13CH	CMPIC0F	CMPIF0F	CMPMK0F	0	0	0	CMPPR0F2	CMPPR0F1	CMPPR0F0
FFFFF13EH	CMPIC1L	CMPIF1L	CMPMK1L	0	0	0	CMPPR1L2	CMPPR1L1	CMPPR1L0
FFFFF140H	CMPIC1F	CMPIF1F	CMPMK1F	0	0	0	CMPPR1F2	CMPPR1F1	CMPPR1F0
FFFFF142H	TB0OVIC	TB0OVIF	TB0OVMK	0	0	0	TB0OVPR2	TB0OVPR1	TB0OVPR0
FFFFF144H	TB0CCIC0	TB0CCIF0	TB0CCMK0	0	0	0	TB0CCPR02	TB0CCPR01	TB0CCPR00
FFFFF146H	TB0CCIC1	TB0CCIF1	TB0CCMK1	0	0	0	TB0CCPR12	TB0CCPR11	TB0CCPR10
FFFFF148H	TB0CCIC2	TB0CCIF2	TB0CCMK2	0	0	0	TB0CCPR22	TB0CCPR21	TB0CCPR20
FFFFF14AH	TB0CCIC3	TB0CCIF3	TB0CCMK3	0	0	0	TB0CCPR32	TB0CCPR31	TB0CCPR30
FFFFF14CH	TB1OVIC	TB1OVIF	TB1OVMK	0	0	0	TB1OVPR2	TB1OVPR1	TB1OVPR0
FFFFF14EH	TB1CCIC0	TB1CCIF0	TB1CCMK0	0	0	0	TB1CCPR02	TB1CCPR01	TB1CCPR00
FFFFF150H	TB1CCIC1	TB1CCIF1	TB1CCMK1	0	0	0	TB1CCPR12	TB1CCPR11	TB1CCPR10
FFFFF152H	TB1CCIC2	TB1CCIF2	TB1CCMK2	0	0	0	TB1CCPR22	TB1CCPR21	TB1CCPR20
FFFFF154H	TB1CCIC3	TB1CCIF3	TB1CCMK3	0	0	0	TB1CCPR32	TB1CCPR31	TB1CCPR30
FFFFF156H	TT0OVIC	TT0OVIF	TT0OVMK	0	0	0	TT0OVPR2	TT0OVPR1	TT0OVPR0
FFFFF158H	TT0CCIC0	TT0CCIF0	TT0CCMK0	0	0	0	TT0CCPR02	TT0CCPR01	TT0CCPR00
FFFFF15AH	TT0CCIC1	TT0CCIF1	TT0CCMK1	0	0	0	TT0CCPR12	TT0CCPR11	TT0CCPR10
FFFFF15CH	TT0IECIC[注]	TT0IECIF	TT0IECMK	0	0	0	TT0IECPR2	TT0IECPR1	TT0IECPR0
FFFFF15EH	TT1OVIC	TT1OVIF	TT1OVMK	0	0	0	TT1OVPR2	TT1OVPR1	TT1OVPR0
FFFFF160H	TT1CCIC0	TT1CCIF0	TT1CCMK0	0	0	0	TT1CCPR02	TT1CCPR01	TT1CCPR00
FFFFF162H	TT1CCIC1	TT1CCIF1	TT1CCMK1	0	0	0	TT1CCPR12	TT1CCPR11	TT1CCPR10

注　　仅限 V850E/IG3

表 **20-2.** 中断控制寄存器的地址和位（**2/3**）

地址	寄存器	位							
		<7>	<6>	5	4	3	2	1	0
FFFFF164H	TT1IECIC	TT1IECIF	TT1IECMK	0	0	0	TT1IECPR2	TT1IECPR1	TT1IECPR0
FFFFF166H	TA0OVIC	TA0OVIF	TA0OVMK	0	0	0	TA0OVPR2	TA0OVPR1	TA0OVPR0
FFFFF168H	TA0CCIC0	TA0CCIF0	TA0CCMK0	0	0	0	TA0CCPR02	TA0CCPR01	TA0CCPR00
FFFFF16AH	TA0CCIC1	TA0CCIF1	TA0CCMK1	0	0	0	TA0CCPR12	TA0CCPR11	TA0CCPR10
FFFFF16CH	TA1OVIC	TA1OVIF	TA1OVMK	0	0	0	TA1OVPR2	TA1OVPR1	TA1OVPR0
FFFFF16EH	TA1CCIC0	TA1CCIF0	TA1CCMK0	0	0	0	TA1CCPR02	TA1CCPR01	TA1CCPR00
FFFFF170H	TA1CCIC1	TA1CCIF1	TA1CCMK1	0	0	0	TA1CCPR12	TA1CCPR11	TA1CCPR10
FFFFF172H	TA2OVIC	TA2OVIF	TA2OVMK	0	0	0	TA2OVPR2	TA2OVPR1	TA2OVPR0
FFFFF174H	TA2CCIC0	TA2CCIF0	TA2CCMK0	0	0	0	TA2CCPR02	TA2CCPR01	TA2CCPR00
FFFFF176H	TA2CCIC1	TA2CCIF1	TA2CCMK1	0	0	0	TA2CCPR12	TA2CCPR11	TA2CCPR10
FFFFF178H	TA3OVIC	TA3OVIF	TA3OVMK	0	0	0	TA3OVPR2	TA3OVPR1	TA3OVPR0
FFFFF17AH	TA3CCIC0	TA3CCIF0	TA3CCMK0	0	0	0	TA3CCPR02	TA3CCPR01	TA3CCPR00
FFFFF17CH	TA3CCIC1	TA3CCIF1	TA3CCMK1	0	0	0	TA3CCPR12	TA3CCPR11	TA3CCPR10
FFFFF17EH	TA4OVIC	TA4OVIF	TA4OVMK	0	0	0	TA4OVPR2	TA4OVPR1	TA4OVPR0
FFFFF180H	TA4CCIC0	TA4CCIF0	TA4CCMK0	0	0	0	TA4CCPR02	TA4CCPR01	TA4CCPR00
FFFFF182H	TA4CCIC1	TA4CCIF1	TA4CCMK1	0	0	0	TA4CCPR12	TA4CCPR11	TA4CCPR10
FFFFF184H	DMAIC0	DMAIF0	DMAMK0	0	0	0	DMAPR02	DMAPR01	DMAPR00
FFFFF186H	DMAIC1	DMAIF1	DMAMK1	0	0	0	DMAPR12	DMAPR11	DMAPR10
FFFFF188H	DMAIC2	DMAIF2	DMAMK2	0	0	0	DMAPR22	DMAPR21	DMAPR20
FFFFF18AH	DMAIC3	DMAIF3	DMAMK3	0	0	0	DMAPR32	DMAPR31	DMAPR30
FFFFF18CH	UREIC	UREIF	UREMK	0	0	0	UREPR2	UREPR1	UREPR0
FFFFF18EH	URIC	URIF	URMK	0	0	0	URPR2	URPR1	URPR0
FFFFF190H	UTIC	UTIF	UTMK	0	0	0	UTPR2	UTPR1	UTPR0
FFFFF192H	UIFIC	UIFIF	UIFMK	0	0	0	UIFPR2	UIFPR1	UIFPR0
FFFFF194H	UTOIC	UTOIF	UTOMK	0	0	0	UTOPR2	UTOPR1	UTOPR0
FFFFF196H	UA0REIC	UA0REIF	UA0REMK	0	0	0	UA0REPR2	UA0REPR1	UA0REPR0
FFFFF198H	UA0RIC	UA0RIF	UA0RMK	0	0	0	UA0RPR2	UA0RPR1	UA0RPR0
FFFFF19AH	UA0TIC	UA0TIF	UA0TMK	0	0	0	UA0TPR2	UA0TPR1	UA0TPR0
FFFFF19CH	CB0REIC	CB0REIF	CB0REMK	0	0	0	CB0REPR2	CB0REPR1	CB0REPR0
FFFFF19EH	CB0RIC	CB0RIF	CB0RMK	0	0	0	CB0RPR2	CB0RPR1	CB0RPR0
FFFFF1A0H	CB0TIC	CB0TIF	CB0TMK	0	0	0	CB0TPR2	CB0TPR1	CB0TPR0
FFFFF1A2H	UA1REIC	UA1REIF	UA1REMK	0	0	0	UA1REPR2	UA1REPR1	UA1REPR0
FFFFF1A4H	UA1RIC	UA1RIF	UA1RMK	0	0	0	UA1RPR2	UA1RPR1	UA1RPR0
FFFFF1A6H	UA1TIC	UA1TIF	UA1TMK	0	0	0	UA1TPR2	UA1TPR1	UA1TPR0
FFFFF1A8H	CB1REIC	CB1REIF	CB1REMK	0	0	0	CB1REPR2	CB1REPR1	CB1REPR0
FFFFF1AAH	CB1RIC	CB1RIF	CB1RMK	0	0	0	CB1RPR2	CB1RPR1	CB1RPR0
FFFFF1ACH	CB1TIC	CB1TIF	CB1TMK	0	0	0	CB1TPR2	CB1TPR1	CB1TPR0
FFFFF1AEH	UA2REIC	UA2REIF	UA2REMK	0	0	0	UA2REPR2	UA2REPR1	UA2REPR0
FFFFF1B0H	UA2RIC	UA2RIF	UA2RMK	0	0	0	UA2RPR2	UA2RPR1	UA2RPR0
FFFFF1B2H	UA2TIC	UA2TIF	UA2TMK	0	0	0	UA2TPR2	UA2TPR1	UA2TPR0
FFFFF1B4H	CB2REIC	CB2REIF	CB2REMK	0	0	0	CB2REPR2	CB2REPR1	CB2REPR0
FFFFF1B6H	CB2RIC	CB2RIF	CB2RMK	0	0	0	CB2RPR2	CB2RPR1	CB2RPR0
FFFFF1B8H	CB2TIC	CB2TIF	CB2TMK	0	0	0	CB2TPR2	CB2TPR1	CB2TPR0

表 **20-2.** 中断控制寄存器的地址和位（**3/3**）

地址	寄存器	位							
		<7>	<6>	5	4	3	2	1	0
FFFFF1BAH	IICIC	IICIF	IICMK	0	0	0	IICPR2	IICPR1	IICPR0
FFFFF1BCH	AD0IC	AD0IF	AD0MK	0	0	0	AD0PR2	AD0PR1	AD0PR0
FFFFF1BEH	AD1IC	AD1IF	AD1MK	0	0	0	AD1PR2	AD1PR1	AD1PR0
FFFFF1C0H	AD2IC	AD2IF	AD2MK	0	0	0	AD2PR2	AD2PR1	AD2PR0
FFFFF1C2H	TM0EQIC0	TM0EQIF0	TM0EQMK0	0	0	0	TM0EQPR02	TM0EQPR01	TM0EQPR00
FFFFF1C4H	TM1EQIC0	TM1EQIF0	TM1EQMK0	0	0	0	TM1EQPR02	TM1EQPR01	TM1EQPR00
FFFFF1C6H	TM2EQIC0	TM2EQIF0	TM2EQMK0	0	0	0	TM2EQPR02	TM2EQPR01	TM2EQPR00
FFFFF1C8H	TM3EQIC0	TM3EQIF0	TM3EQMK0	0	0	0	TM3EQPR02	TM3EQPR01	TM3EQPR00
FFFFF1CAH	ADT0IC	ADT0IF	ADT0MK	0	0	0	ADT0PR2	ADT0PR1	ADT0PR0
FFFFF1CCH	ADT1IC	ADT1IF	ADT1MK	0	0	0	ADT1PR2	ADT1PR1	ADT1PR0

20.3.5　中断屏蔽寄存器 0 至 5（IMR0 至 IMR5）

IMR0 至 IMR5 寄存器用于设置可屏蔽中断的中断屏蔽状态。IMR0.xxMKn 至 IMR3.xxMKn 位等同于 xxICn.xxMKn 位。

IMRm 寄存器可以按 16 位宽度进行读取或写入（m = 0 至 5）.。

如果 IMRm 寄存器的高 8 位用作 IMRmH 寄存器，低 8 位用作 IMRmL 寄存器，则这些寄存器可以按字节或按位进行读取或写入。

系统复位后，这些寄存器被设置为 FFFFH。

注意事项　**设备文件将 xxICn.xxMKn 位定义为保留字。如果使用 xxMKn 进行位操作，xxICn 寄存器的内容而不是 IMRm 寄存器被改写（结果，IMRm 寄存器的内容也被改写）。**

（1/2）

复位后: FFFFH　R/W　地址:　IMR5 FFFFF10AH
IMR5L FFFFF10AH, IMR5H FFFFF10BH

	15	14	13	12	11	10	9	8
IMR5 (IMR5H注 1)	1	ADT1MK	ADT0MK	TM3EQMK0	TM2EQMK0	TM1EQMK0	TM0EQMK0	AD2MK
	7	6	5	4	3	2	1	0
(IMR5L)	AD1MK	AD0MK	IICMK	CB2TMK	CB2RMK	CB2REMK	UA2TMK	UA2RMK

复位后: FFFFH　R/W　地址:　IMR4 FFFFF108H
IMR4L FFFFF108H, IMR4H FFFFF109H

	15	14	13	12	11	10	9	8
IMR4 (IMR4H注 1)	UA2REMK	CB1TMK	CB1RMK	CB1REMK	UA1TMK	UA1RMK	UA1REMK	CB0TMK
	7	6	5	4	3	2	1	0
(IMR4L)	CB0RMK	CB0REMK	UA0TMK	UA0RMK	UA0REMK	UTOMK	UIFMK	UTMK

复位后: FFFFH　R/W　地址:　IMR3 FFFFF106H
IMR3L FFFFF106H, IMR3H FFFFF107H

	15	14	13	12	11	10	9	8
IMR3 (IMR3H注 1)	URMK	UREMK	DMAMK3	DMAMK2	DMAMK1	DMAMK0	TA4CCMK1	TA4CCMK0
	7	6	5	4	3	2	1	0
(IMR3L)	TA4OVMK	TA3CCMK1	TA3CCMK0	TA3OVMK	TA2CCMK1	TA2CCMK0	TA2OVMK	TA1CCMK1

复位后: FFFFH　R/W　地址:　IMR2 FFFFF104H
IMR2L FFFFF104H, IMR2H FFFFF105H

	15	14	13	12	11	10	9	8
IMR2 (IMR2H注 1)	TA1CCMK0	TA1OVMK	TA0CCMK1	TA0CCMK0	TA0OVMK	TT1IECMK	TT1CCMK1	TT1CCMK0
	7	6	5	4	3	2	1	0
(IMR2L)	TT1OVMK	TT0IECMK注 2	TT0CCMK1	TT0CCMK0	TT0OVMK	TB1CCMK3	TB1CCMK2	TB1CCMK1

复位后: FFFFH　R/W　地址:　IMR1 FFFFF102H
IMR1L FFFFF102H, IMR1H FFFFF103H

	15	14	13	12	11	10	9	8
IMR1 (IMR1H注 1)	TB1CCMK0	TB1OVMK	TB0CCMK3	TB0CCMK2	TB0CCMK1	TB0CCMK0	TB0OVMK	CMPMK1F
	7	6	5	4	3	2	1	0
(IMR1L)	CMPMK1L	CMPMK0F	CMPMK0L	PMK18	PMK17	PMK16	PMK15	PMK14

复位后: FFFFH　R/W　地址:　IMR0 FFFFF100H
IMR0L FFFFF100H, IMR0H FFFFF101H

	15	14	13	12	11	10	9	8
IMR0 (IMR0H注 1)	PMK13	PMK12	PMK11	PMK10	PMK09	PMK08	PMK07注 2	PMK06注 2
	7	6	5	4	3	2	1	0
(IMR0L)	PMK05注 2	PMK04注 2	PMK03注 2	PMK02注 2	PMK01	PMK00	LVIHMK	LVILMK

注　**1.** 若要按字节或按位读取/写入 IMR3 至 IMR5 寄存器的第 8 位至第 15 位，指定这些位作为 IMR3H 至 IMR5H 寄存器的第 0 位至第 7 位。

2. 仅在 V850E/IG3 内这些位有效。
在 V850E/IF3 内，请确保将这些位设置为 1。

注意事项　将 IMR5 寄存器的第 15 位（IMR5H 寄存器的第 7 位）设置为 1。如果改变这些位的设置，操作将无法保证。

（2/2）

xxMKn	中断屏蔽标志设置
0	中断服务使能
1	中断服务禁止

备注 xx： 每一个外围单元的标志名称（参见 **表 20-2**）。
n ： 外围单元编号（参见 **表 20-2**）。

20.3.6 正在服务中优先级寄存器（ISPR）

ISPR 寄存器保存当前响应的可屏蔽中断的优先级。当响应一个中断请求信号时，该寄存器中与中断请求信号优先级对应的位被置为 1，并且在中断服务期间一直保持置位。

当执行 RETI 指令时，对应最高优先级中断请求信号的位被硬件自动清 0。但是，从不可屏蔽中断服务或异常处理中返回时，并不被清除为 0。

该寄存器为只读寄存器，按字节或按位读取。

系统复位后，该寄存器被设为 00H。

注意事项　中断允许（EI）状态下，如果在读取 ISPR 寄存器的时候响应一个中断，读取到的 ISPR 寄存器值可能是响应中断对该位置位（1）后的值。为了在响应中断前准确读出 ISPR 寄存器的值，应在中断禁止（DI）的状态下读取。

复位后：00H　　R　　地址：FFFFF1FAH

	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
ISPR	ISPR7	ISPR6	ISPR5	ISPR4	ISPR3	ISPR2	ISPR1	ISPR0

ISPRn	当前正在响应中断的优先级
0	没有响应优先级为n的中断请求信号
1	正在响应优先级为n的中断请求信号

备注　　n＝0 至 7（优先级等级）

20.3.7 可屏蔽中断状态标志（ID）

ID 标志用于控制可屏蔽中断的操作状态，并且存储有关中断请求使能或禁止的控制信息。ID 标志位于 PSW 内。

系统复位后，该标志被置为 00000020H。

复位后： 00000020H

	31–8	7	6	5	4	3	2	1	0
PSW	0	NP	EP	ID	SAT	CY	OV	S	Z

ID	可屏蔽中断服务说明[注]
0	可屏蔽中断请求信号响应使能
1	可屏蔽中断请求信号响应禁止 (挂起)

注　中断禁止标志（ID）功能

ID 位由 DI 指令置位（1），由 EI 指令清除（0）。其值也可由 RETI 指令或 PSW 相关的 LDSR 指令来改变。

不管该标志如何，不可屏蔽中断请求信号和异常事件都可以得到响应。当一个可屏蔽中断请求信号被响应时，ID 标志由硬件自动置位（1）。

当 xxICn.xxIFn 位被置位（1）时，在中断响应禁止期间（ID 标志＝1）产生的中断请求信号可以得到响应，并且 ID 标志被清除（0）。

20.4 外部中断请求输入引脚（INTP00 至 INTP18，INTADT0，INTADT1）

20.4.1 噪声消除

（1） INTP00，INTP01，INTPa（仅限 V850E/IG3），INTP08 至 INTP13，INTP17，INTP18，INTADT0 和 INTADT1 引脚的噪声消除

INTP00，INTP01，INTPa（仅限 V850E/IG3），INTP08 至 INTP13，INTP17，INTP18，INTADT0 和 INTADT1 引脚包含使用模拟滤波器的噪声消除电路。因此，除非各个引脚的输入电平保持特定的或更长一段时间，否则将不会被检测为边沿。经过一定的时间之后，才能检测到边沿。

（3）INTP14 至 INTP16 引脚的噪声消除

INTP14 至 INTP16 引脚包含数字噪声消除电路。
使用 INTNFCm.INTNFCm2 至 INTNFCm.INTNFCm0 位可以选择执行数字采样的采样时钟（m = 14 至 16）。
在 IDLE 和 STOP 模式下，系统时钟停止，所以 INTP14 至 INTP16 引脚不能用来解除 IDLE 和 STOP 模式。

20.4.2 边沿检测

INTPn 引脚的有效边沿可以通过程序选择（V850E/IF3：n = 00，01，08 至 18，V850E/IG3：n = 00 至 18）。可以选择作为有效边沿的边沿是下列之一。

- 上升沿
- 下降沿
- 上升和下降双沿

边沿被检测的 INTPn 信号成为一个中断源。
通过 INTR0 至 INTR2 和 INTF0 至 INTF2 寄存器来指定有效边沿。

（1）外部中断上升沿指定寄存器 0（INTR0），外部中断下降沿指定寄存器 0（INTF0）

INTF0 和 INTR0 寄存器用于指定 INTP00，INTP01 和 INTPa（仅限 V850E/IG3）引脚的触发模式，也可以为各个引脚单独指定有效边沿（上升沿、下降沿或上升和下降双沿）（a = 02 至 07）。

这些寄存器可以按字节或按位进行读取或写入。

系统复位后，这些寄存器被设置为 00H。

注意事项　当引脚功能由外部中断功能（复用功能）改变为端口功能时，可能会检测到一个边沿。所以，将 INTF0n 和 INTR0n 位清为 00，然后再设置端口模式（V850E/IF3：n = 0，1，V850E/IG3：n = 0 至 7）。

复位后：00H　　R/W　　地址：　FFFFFC20H

	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
INTR0	INTR07[注]	INTR06[注]	INTR05[注]	INTR04[注]	INTR03[注]	INTR02[注]	INTR01	INTR00

复位后：00H　　R/W　　地址：　FFFFFC00H

	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
INTF0	INTF07[注]	INTF06[注]	INTF05[注]	INTF04[注]	INTF03[注]	INTF02[注]	INTF01	INTF00

注　　仅限 V850E/IG3 有效。
　　　在 V850E/IF3 内，请确保将这些位设置为 0。

备注　　关于有效边沿规范，参见 **表 20-3**。

表 20-3.　INTP00 至 INTP07 引脚的有效边沿规范

INTF0n	INTR0n	有效边沿规范
0	0	无边沿检测
0	1	上升沿
1	0	下降沿
1	1	上升和下降双边沿

注意事项　当这些寄存器不用作 INTPn 引脚时，请确保将 INTF0n 和 INTR0n 位设置为 00。

备注　　V850E/IF3：n = 0，1
　　　　V850E/IG3：n = 0 至 7

（2）外部中断上升沿指定寄存器 1（INTR1），外部中断下降沿指定寄存器 1（INTF1）

INTF1 和 INTR1 寄存器用于指定 INTP08 至 INTP13，INTP17 和 INTP18 引脚的触发模式，也可以为各个引脚单独指定有效边沿（上升沿、下降沿或上升和下降双沿）。

这些寄存器可以按字节或按位进行读取或写入。

系统复位后，这些寄存器被设置为 00H。

注意事项　**当引脚功能由外部中断功能（复用功能）改变为端口功能时，可能会检测到一个边沿。所以，将 INTFn 和 INTRn 位清为 00，然后再设置端口模式（n = 08 至 13，17，18）。**

复位后：00H　　R/W　　地址：　FFFFFC22H

	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
INTR1	INTR18	INTR17	INTR13	INTR12	INTR11	INTR10	INTR09	INTR08

复位后：00H　　R/W　　地址：　FFFFFC02H

	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
INTF1	INTF18	INTF17	INTF13	INTF12	INTF11	INTF10	INTF09	INTF08

备注　　关于有效边沿规范，参见 **表 20-4**。

表 20-4. INTP08 至 INTP13，INTP17 和 INTP18 引脚的有效边沿规范

INTFn	INTRn	有效边沿规范
0	0	无边沿检测
0	1	上升沿
1	0	下降沿
1	1	上升和下降双边沿

注意事项　**当这些寄存器不用作 INTPn 引脚时，请确保将 INTFn 和 INTRn 位清为 00。**

备注　　n = 08 至 13，17，18

（3）外部中断上升沿指定寄存器 2（INTR2），外部中断下降沿指定寄存器 2（INTF2）

INTF2 和 INTR2 寄存器用于指定 INTP14 至 INTP16 引脚的触发模式，也可以为各个引脚单独指定有效边沿（上升沿、下降沿或上升和下降双沿）。

这些寄存器可以按字节或按位进行读取或写入。

系统复位后，这些寄存器被设置为 00H。

注意事项　**当引脚功能由外部中断功能（复用功能）改变为端口功能时，可能会检测到一个边沿。所以，将 INTF1n 和 INTR1n 位清为 00，然后再设置端口模式（n = 4 至 6）。**

复位后：00H　　R/W　　地址：FFFFFC24H

	7	6	5	4	3	<2>	<1>	<0>
INTR2	0	0	0	0	0	INTR16	INTR15	INTR14

复位后：00H　　R/W　　地址：FFFFFC04H

	7	6	5	4	3	<2>	<1>	<0>
INTF2	0	0	0	0	0	INTF16	INTF15	INTF14

备注　关于有效边沿规范，参见 **表 20-5**。

表 20-5. INTP14 至 INTP16 引脚的有效边沿规范

INTF1n	INTR1n	有效边沿规范
0	0	无边沿检测
0	1	上升沿
1	0	下降沿
1	1	上升和下降双边沿

注意事项　**当这些寄存器不用作 INTP1n 引脚时，请确保将 INTF1n 和 INTR1n 位清为 00。**

备注　n = 4 至 6

20.5 软件异常

当 CPU 执行 TRAP 指令时，产生软件异常，并且总是能够得到响应。

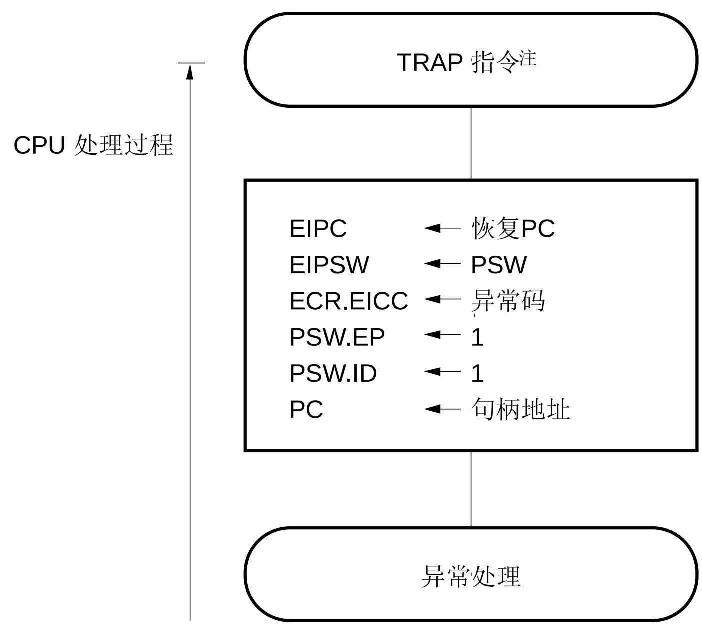
20.5.1 操作

如果发生软件异常，CPU 执行以下处理，并控制移交给句柄程序。

<1>　将 PC 保存至 EIPC。
<2>　将 PSW 保存至 EIPSW。
<3>　写入异常码到 ECR（中断源）的低 16 位（EICC）。
<4>　将 PSW.EP 和 PSW.ID 位置位（1）。
<5>　将软件异常对应的句柄地址（00000040H 或 00000050H）设置到 PC 指针，并进行控制转移。

软件异常的处理过程如下所示：

图 20-8. 软件异常的处理过程



注　　TRAP 指令格式：TRAP 向量（向量是从 00H 至 1FH 的值）

句柄地址由 TRAP 指令的操作数（向量）决定。如果向量为 00H 至 0FH，句柄地址是 00000040H，而如果向量为 10H 至 1FH，句柄地址是 00000050H。

20.5.2 恢复

执行 RETI 指令，可以从软件异常处理中恢复。

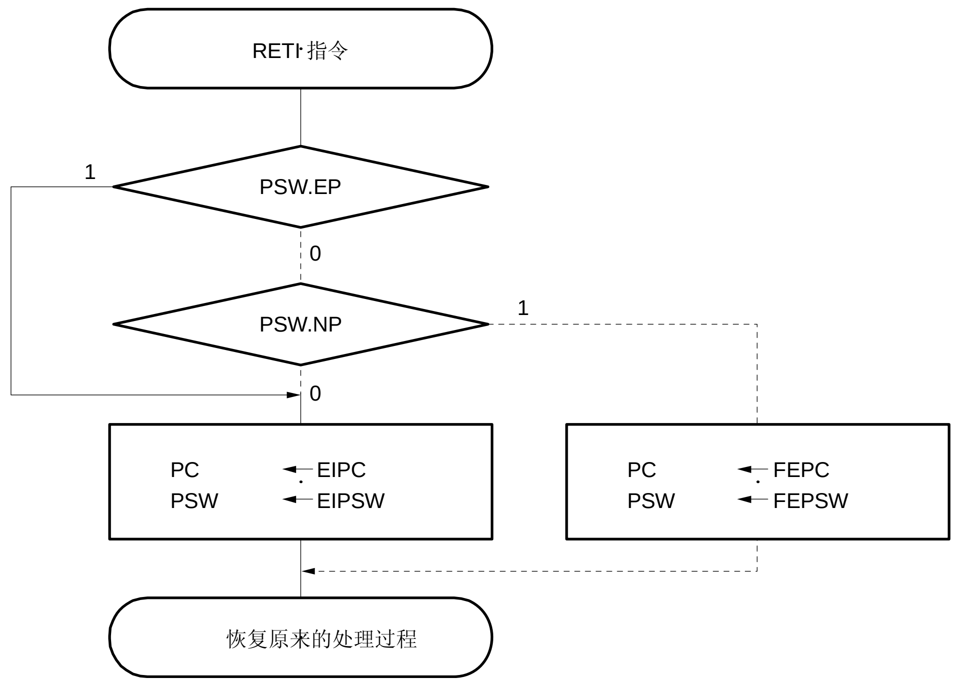
通过执行 RETI 指令，CPU 执行以下处理并将控制转移至被恢复的 PC 地址。

<1> 由于 PSW.EP 位为 1，由 EIPC 和 EIPSW 载入被恢复的 PC 和 PSW。

<2> 控制转移至被恢复的 PC 和 PSW 的地址。

RETI 指令的处理过程如下所示：

图 20-9. RETI 指令处理



注意事项 在软件异常处理期间，如果 **EP** 位和 **NP** 位被 **LDSR** 指令改变时，为了在 **RETI** 指令的恢复过程中正确恢复 **PC** 和 **PSW**，必须在紧邻 **RETI** 指令之前，使用 **LDSR** 指令将 **EP** 置回为 **1** 以及将 **NP** 清除为 **0**。

备注 实线显示的是 CPU 的处理流程。

20.5.3 异常状态标志（EP）

EP 标志是一个状态标志，用于指示异常处理正在处理中。当异常发生时，该位被置位。EP 标志分配在 PSW 内。系统复位后，该标志被设置为 00000020H。

复位后: 00000020H

	31–8	7	6	5	4	3	2	1	0
PSW	0	NP	EP	ID	SAT	CY	OV	S	Z

EP	异常处理状态
0	没有进行异常处理。
1	正在进行异常处理。

20.6 异常陷阱

异常陷阱是一个中断，当非法执行指令时产生异常陷阱中断请求。在 V850E/IF3 和 V850E/IG3 中，非法操作码陷阱（ILGOP：非法操作码陷阱）被认为是一个异常陷阱。

20.6.1 非法操作码定义

非法指令的操作码（第 10 位至第 5 位）为 111111B，副操作码（第 26 位至第 23 位）为 0111B 至 1111B，另一个副操作码（第 16 位）是 0B。当执行这样一条指令时，就会产生一个异常陷阱。

15 11	10 5	4 0	31 27	26 23	22	16
X X X X X	1 1 1 1 1 1	X X X X X	X X X X X	0 1 1 1 至 1 1 1 1	X X X X X X	0

×： 任意

注意事项 建议不要使用非法操作码，因为该指令将来可能被指定为非法操作码。

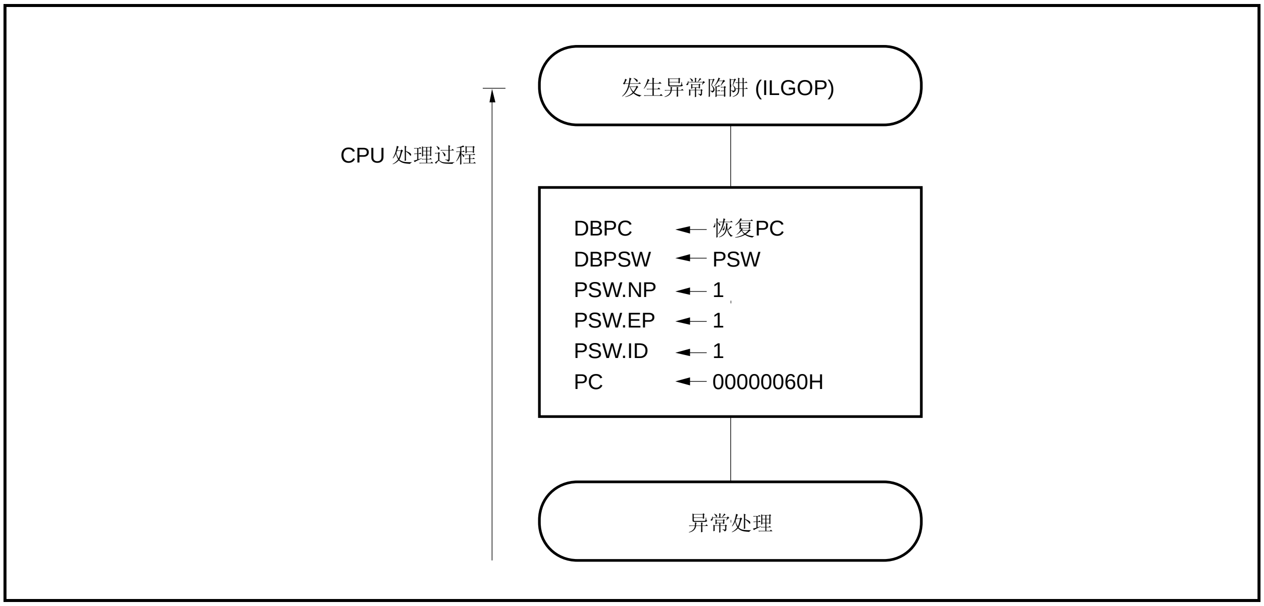
（1）操作

当发生异常陷阱时，CPU 执行以下处理，并将控制转移到句柄程序：

<1> 将待恢复的 PC 保存至 DBPC。
<2> 将当前 PSW 保存至 DBPSW。
<3> 将 PSW.NP，PSW.EP 和 PSW.ID 位置位（1）。
<4> 将异常陷阱对应的处理程序地址（00000060H）设置到 PC 指针，并进行控制转移。

异常陷阱的处理过程如下所示：

图 **20-10.** 异常陷阱处理



（**2**）恢复

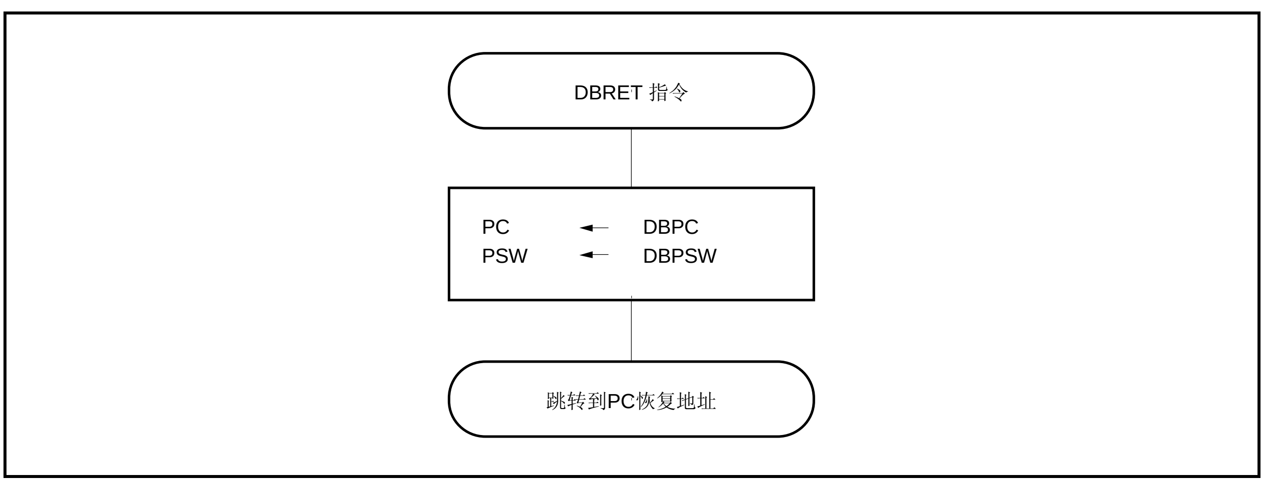
执行 DBRET 指令，可以从异常陷阱中恢复。执行 DBRET 指令时，CPU 进行以下处理并将控制转移至被恢复的 PC 地址。

<1> 从 DBPC 和 DBPSW 中载入被恢复的 PC 和 PSW。
<2> 控制转移至被恢复的 PC 和 PSW 所指定的地址。

注意事项　只有在执行非法操作码和执行 DBRET 指令之间的间隔，才可以访问 DBPC 和 DBPSW。

从异常陷阱中恢复的处理过程如下所示：

图 **20-11.** 异常陷阱恢复处理



20.6.2 调试陷阱

调试陷阱是一个异常，在执行 DBTRAP 指令时产生，总是可以得到响应。

当发生调试陷阱时，CPU 执行以下处理过程。

（1）操作

<1> 将待恢复的 PC 保存至 DBPC。

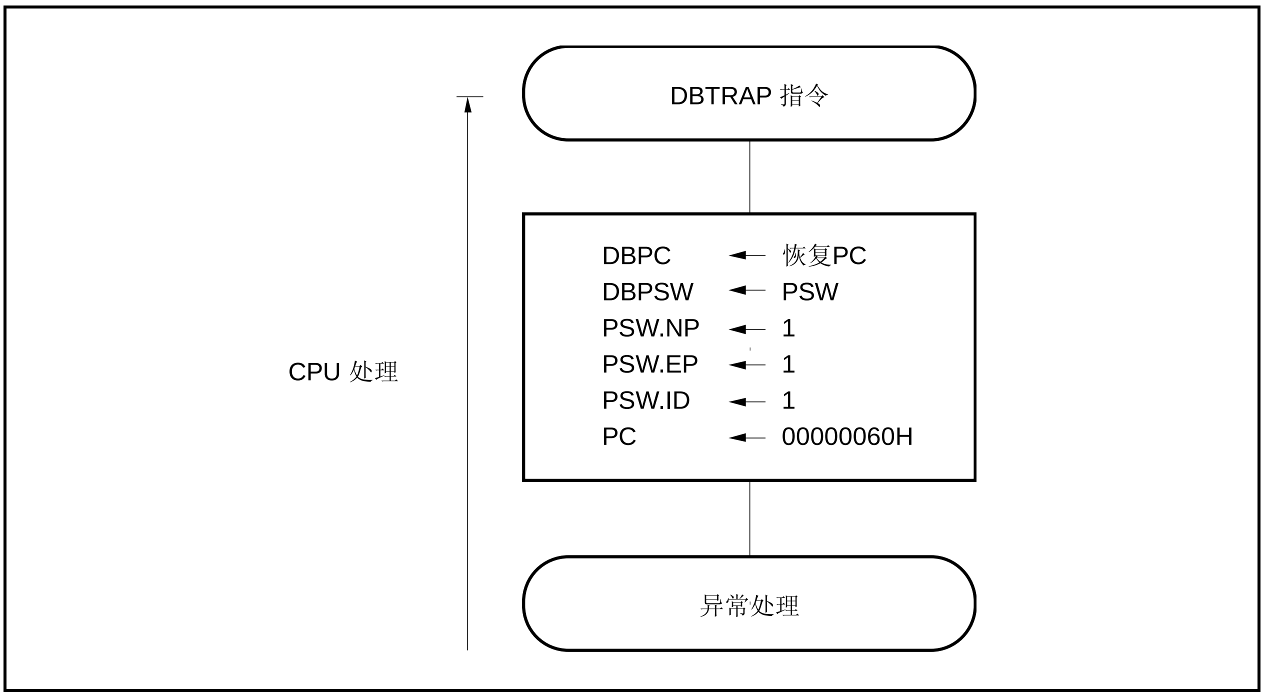
<2> 将当前 PSW 保存至 DBPSW。

<3> 将 PSW.NP，PSW.EP 和 PSW.ID 位置位（1）。

<4> 将调试陷阱的句柄地址（00000060H）设置到 PC 指针，并进行控制转移。

调试陷阱的处理过程如下所示：

图 20-12. 调试陷阱处理



（2）恢复

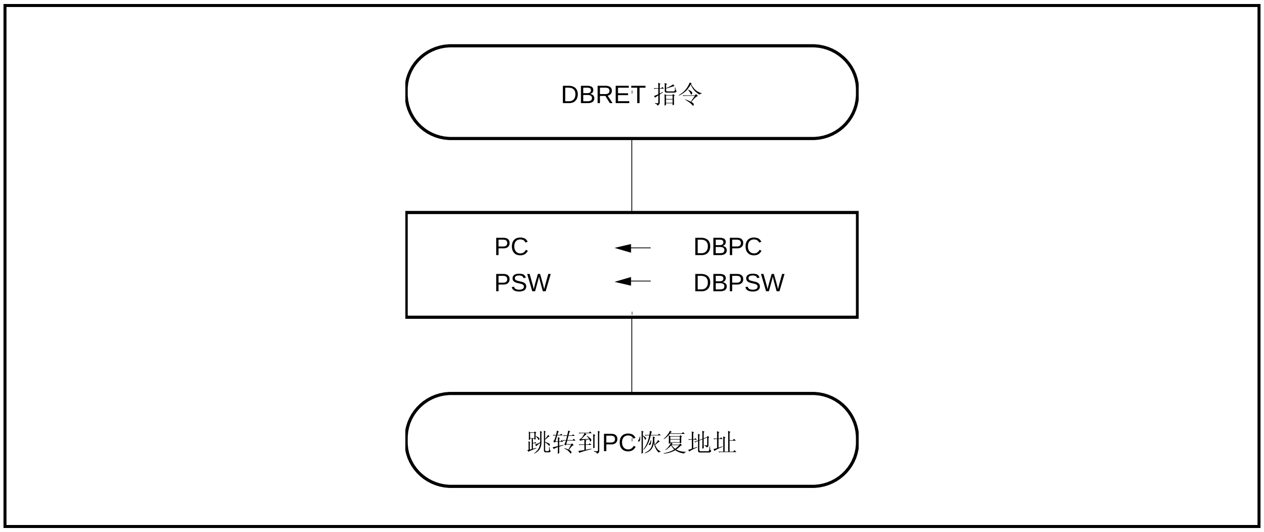
执行 DBRET 指令，可以从异常陷阱中进行恢复。当执行 DBRET 指令时，CPU 执行以下处理过程并将控制转移到恢复 PC 地址。

<1> 从 DBPC 和 DBPSW 中读取被恢复的 PC 和 PSW。
<2> 控制转移返回到被恢复的 PC 和 PSW 的地址。

注意事项　只有在执行 DBTRAP 和执行 DBRET 指令之间的间隔，才可以访问 DBPC 和 DBPSW。

从调试陷阱中恢复的处理过程如下所示：

图 20-13. 从调试陷阱中恢复的处理



20.7 多重中断服务控制

多重中断服务控制是一个过程，在服务期间如果由更高优先级的中断请求产生，则可以打断当前正在服务处理的中断请求，更高优先级的中断请求被优先响应且进行中断处理服务。

如果产生的中断请求优先级低于当前正在服务处理的中断请求，则该中断请求信号被挂起。

当中断使能时（PSW.ID 位 = 0），执行可屏蔽中断的多重中断服务控制。于是，要执行多重中断，即使在中断服务程序中，也必需设置为中断使能状态（PSW.ID 位 = 0）。

在一个可屏蔽中断或软件异常服务程序中，如果可屏蔽中断被使能，或产生软件异常，则必需保存 EIPC 和 EIPSW。

通过下列过程来完成。

（1）在服务程序中，可屏蔽中断的响应

可屏蔽中断或异常的服务程序

...
...
- EIPC 保存到存储器或寄存器
- EIPSW 保存到存储器或寄存器
- EI 指令（中断响应被使能）

...
... ← 可屏蔽中断响应
...
...
- DI 指令（中断响应被禁止）
- 已保存的值恢复到 EIPSW
- 保存的值恢复到 EIPC
- RETI 指令

（2）在服务程序中，异常的产生

可屏蔽中断或异常的服务程序

```
        ...
        ...
• EIPC 保存到存储器或寄存器
• EIPSW 保存到存储器或寄存器
        ...
• TRAP 指令                          ← 响应异常，比如 TRAP 指令
        ...
• 已保存的值恢复到 EIPSW
• 已保存的值恢复到 EIPC
• RETI 指令
```

多重中断服务控制的优先级次序有 8 个级别，各个可屏蔽中断请求信号从 0 至 7（0 是最高优先级），但是可以通过软件按需设置。使用中断控制寄存器（xxICn）的 xxPRn0 至 xxPRn2 位为各个可屏蔽中断请求信号设置优先级次序。系统复位后，一个中断请求信号通过 xxMKn 位被屏蔽，并通过 xxPRn0 至 xxPRn2 位将优先级次序设置为 7 级。

可屏蔽中断的优先级次序如下。

（高）　0 级 > 1 级 > 2 级 > 3 级 > 4 级 > 5 级 > 6 级 > 7 级　（低）

较高优先级中断的服务完成且已经执行 RETI 指令之后，中断服务被暂停，因为多重中断服务控制已被恢复。
当前中断完成且已经执行 RETI 指令之后，一个挂起的中断请求信号被响应。

注意事项　在不可屏蔽中断服务程序中（直到执行 RETI 指令），可屏蔽中断被暂停而不响应。

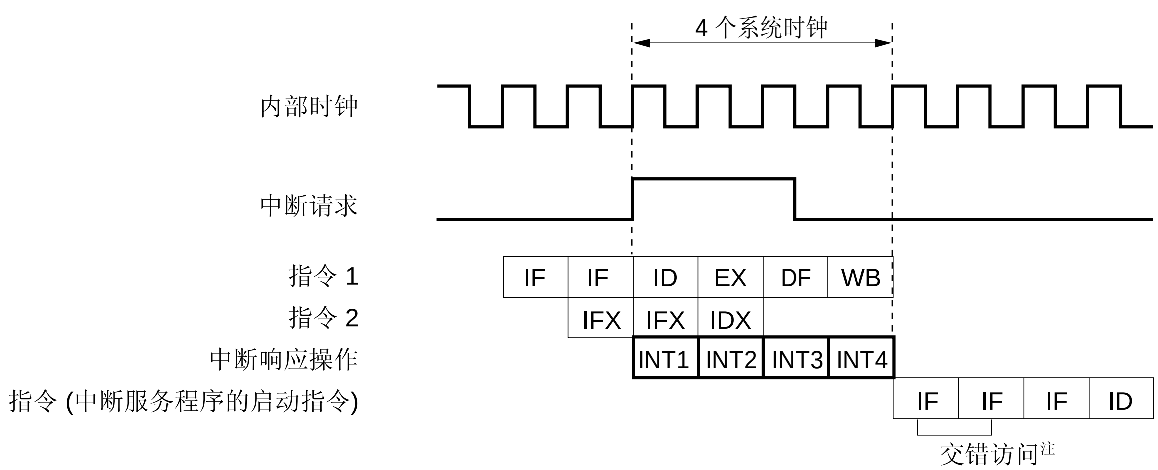
备注　xx：　每一个外围单元的标志名称（参见 **表 20-2**）。
　　　　n ：　外围单元编号（参见**表 20-2**）。

20.8 CPU 的中断响应时间

除了下列情况外，CPU 的中断响应时间最小为 4 个时钟周期。若要连续输入中断请求信号，至少需要在前一中断之后 4 个时钟周期才能输入下一中断请求信号。

- 在 IDLE/STOP 模式下
- 当中断请求无采样指令被连续执行时（参见 **20.9 CPU 不响应中断时**）。
- 当访问片上外围 I/O 寄存器时

图 20-14. 中断请求信号响应的流水线操作（概述）



注　关于交错访问，参见 **V850E1 架构用户手册（U14559E）**的 **8.1.2 2 时钟跳转**。

备注　INT1 至 INT4：中断响应处理
IFX：　无效指令取指
IDX：　无效指令解码

	中断等待时间（内部系统时钟）			条件
	内部中断	外部中断		
		INTP00，INTP01，INTPa[注 1]，INTP08 至INTP13，INTP17，INTP18，INTADT0，INTADT1	INTP14 至 INTP16	
最小值	4	4 + 模拟滤波时间	4 + **注 3** + 数字噪声滤波	下列情况除外： • IDLE/STOP 模式下 • 访问外部总线 • 2 个或更多个中断请求无采样指令连续执行 • 访问片上外围 I/O 寄存器
最大值	7[注 2]	7 + 模拟滤波时间	7 + **注 3** + 数字噪声滤波	

注　**1.** 仅限 V850E/IG3
2. 向内部 ROM 执行的 LD 指令（对齐访问期间）。
3. 关于内部系统时钟的数量，参见 **4.6（1） 数字噪声消除 0 控制寄存器 n（INTNFCn）**。

备注　a = 02 至 07

20.9 CPU 不响应中断的时期

当指令正在执行时，CPU 可以响应中断。但是，在中断请求无采样指令和下一条指令之间不会响应中断（中断被挂起）。

中断请求无采样指令如下所示：

- EI 指令
- DI 指令
- LDSR reg2，0x5 指令（对于 PSW）
- 命令寄存器（PRCMD）的存储指令
- 对下列寄存器的存储指令或除 tst1 之外的位操作指令
  - 中断相关的寄存器：
    中断控制寄存器（xxICn），中断屏蔽寄存器 0 至 5（IMR0 至 IMR5）
  - 节电控制寄存器（PSC）

备注　xx：　每一个外围单元的标志名称（参见 **表 20-2**）。
　　　n ：　外围单元编号（参见 **表 20-2**）。

20.10注意事项

请注意如果一个端口被设置为外部中断输入（INTPn），定时器/计数器相关中断、串行接口相关中断和 A/D 转换相关中断不会发生，它们都是复用功能（V850E/IF3：n = 00，01，08 至 18，ADT0，ADT1，V850E/IG3：n = 00 至 18，ADT0，ADT1）。

第二十一章　待机功能

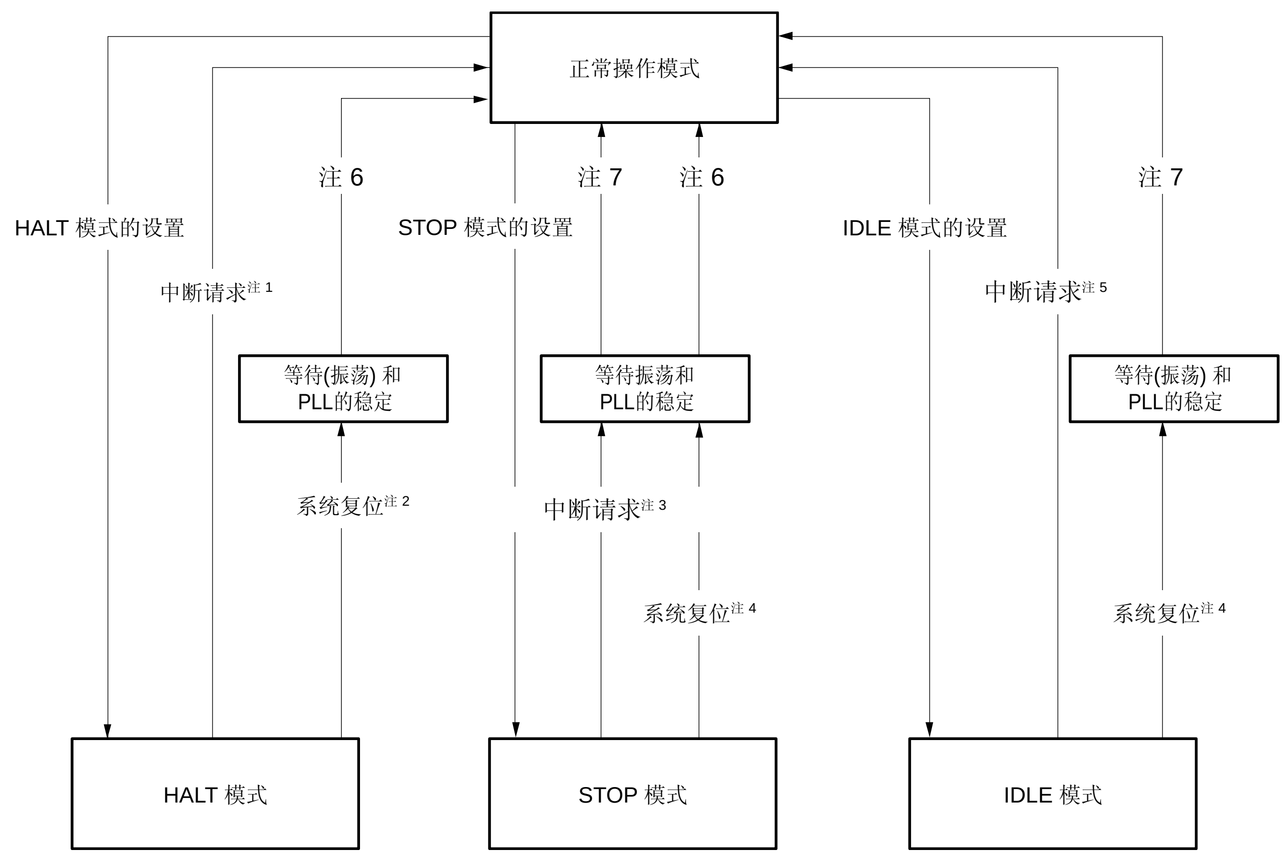
21.1 概述

通过结合运用待机模式并为实际应用选择恰当的模式，可以有效减少系统的功耗。表 21-1 列出了待机模式的类型：

表 21-1. 待机模式

模式	功能概述
HALT 模式	此模式下，仅停止 CPU 的工作时钟。
IDLE 模式	此模式下，除振荡器、PLL、从模式下的 CSIB、时钟监视器、低电压检测器(LVI)、上电清零电路(POC)之外，停止其它所有内部电路的操作。
STOP 模式	此模式下，除从模式下的 CSIB、低电压检测器(LVI)、上电清零电路(POC)外，停止其它所有内部电路的操作。

图 21-1. 状态转换



注

1. 不可屏蔽中断请求信号(INTWDT)或未屏蔽的可屏蔽中断请求信号
2. $\overline{\text{RESET}}$ 引脚输入、看门狗定时器溢出产生的复位信号(WDTRES)、低电压检测器(LVI)产生的复位信号(LVIRES)或上电清零电路(POC)产生的复位信号(POCRES)
3. 来自 STOP 模式下可操作的外围功能的未屏蔽外部中断请求信号(INTP00, INTP01, INTP02 至 INTP07 (仅限 V850E/IG3), INTP08 至 INTP13, INTP17, INTP18, INTADT0 或 INTADT1)或未屏蔽内部中断请求信号(从模式下的 CSIB 相关的中断请求信号)
4. $\overline{\text{RESET}}$ 引脚输入、低电压检测器(LVI)产生的复位信号(LVIRES)或上电清零电路(POC)产生的复位信号(POCRES)
5. 来自 IDLE 模式下可操作的外围功能的未屏蔽外部中断请求信号(INTP00, INTP01, INTP02 至 INTP07 (仅限 V850E/IG3), INTP08 至 INTP13, INTP17, INTP18, INTADT0 或 INTADT1)或未屏蔽内部中断请求信号(从模式下的 CSIB 相关的中断请求信号)
6. 振荡稳定时间通过振荡稳定时间等待控制(OST)进行计数
   复位释放后的振荡稳定时间是必需的，因为 PLL 被复位信号重新初始化。稳定时间由缺省值决定。
7. 振荡稳定时间通过振荡稳定时间等待控制(OST)进行计数
   振荡稳定时间由 OSTS 寄存器的设置决定。

21.2 控制寄存器

（1）节电控制寄存器（PSC）

PSC 寄存器是 8 位寄存器，用于控制待机功能。该寄存器的 STB 位用于指定待机模式。该寄存器是专用寄存器（参见 **3.4.8 专用寄存器**），只有通过特定的流程才能将数据写入该寄存器。

该寄存器可以按字节或位进行读取或写入。

系统复位后，该寄存器被设置为 00H。

复位后： 00H　　R/W　　地址： FFFFF1FEH

	7	6	5	<4>	3	2	<1>	0
PSC	0	0	0	INTM	0	0	STB	0

INTM	通过可屏蔽的中断请求(INTxx注 1)控制注 2待机模式
0	允许通过INTxx 请求解除待机模式
1	禁止通过INTxx 请求解除待机模式

STB	设置操作模式
0	正常模式
1	待机模式

注　**1.** 详情参见表 **20-1 中断源列表**。
　　2. 只有在 IDLE 模式和 STOP 模式下，该设置有效。

注意事项 1. 请确保将第 0, 2, 3 位和第 5 位至第 7 位设置为“0”。

2. 通过将 STB 位设置为 1 进入待机模式之前，请确保设置 PCC 寄存器为 03H 然后将 STB 位设置为 1。否则，待机模式可能无法被设置进入或解除。在解除待机模式之后，可以任意改变 PCC 寄存器的值。

3. 要设置 IDLE 模式或 STOP 模式，按顺序设置 PCC 寄存器为 03H，再设置 PSMR.PSM0 位，然后设置 STB 位为 1。

（2）　节电模式寄存器（PSMR）

PSMR 寄存器是 8 位寄存器，用于控制待机模式下的操作。

该寄存器可按字节或按位进行读取或写入。

复位输入将该寄存器设置为 00H。

复位后：00H　R/W　地址：FFFFF820H

	7	6	5	4	3	2	1	<0>
PSMR	0	0	0	0	0	0	0	PSM0

PSM0	在软件待机模式下的操作说明
0	IDLE 模式
1	STOP 模式

注意事项　1.　请确保将第 1 位至第 7 位清除为“0”。

2.　仅当 PSC.STB 位为 1 时，PSM0 位有效。

21.3 HALT 模式

21.3.1 设置和操作状态

在正常操作模式下，执行一条专用指令（HALT），可以设置进入 HALT 模式。

当设置为 HALT 模式时，只是停止 CPU 的时钟供给，时钟发生器和 PLL 继续工作。提供给其它片上外围功能的时钟继续。

结果，程序停止执行，内部 RAM 保持设置为 HALT 模式之前的内容。独立于 CPU 指令处理的片上外围功能继续工作。

表 21-3 显示了 HALT 模式下的操作状态。

通过 HALT 模式和正常操作模式的切换操作，可以减少系统的平均电流消耗。

注意事项 1. 在 HALT 指令后，插入五个或更多的 NOP 指令。

2. 如果中断请求被挂起时执行 HALT 指令，则设置为 HALT 模式，但是立即被该中断请求解除。

21.3.2 解除 HALT 模式

HALT 模式的解除，可以通过不可屏蔽中断请求信号(INTWDT)、未屏蔽的可屏蔽中断请求信号和复位信号($\overline{\text{RESET}}$引脚输入、看门狗定时器溢出产生的复位信号(WDTRES)、低电压检测器(LVI)产生的复位信号(LVIRES)或上电清零电路(POC)产生的复位信号(POCRES))。

HALT 模式释放后，恢复为正常操作模式。

（1）通过不可屏蔽中断请求信号或未屏蔽的可屏蔽中断请求信号解除 HALT 模式

通过不可屏蔽中断请求信号（INTWDT）或未屏蔽的可屏蔽中断请求信号，可以解除 HALT 模式，且与中断请求的优先级无关。但是，如果在一个中断服务程序中设置 HALT 模式，则之后发生的中断请求的服务如下。

（a）如果产生的中断请求信号优先级低于或等于当前正在服务的中断，则 HALT 模式被解除，但是不响应这个新产生的中断请求信号。该中断请求信号被保持。因此，在 HALT 指令之后的下一条指令开始执行。

（b）如果发出的中断请求信号优先级高于当前正在服务的中断（包括不可屏蔽中断请求信号），则 HALT 模式解除，且响应该中断请求信号。因此，执行跳转到句柄地址。

表 21-2. 用中断请求信号解除 HALT 模式后的操作

解除源	中断允许（EI）状态	中断禁止（DI）状态
不可屏蔽中断请求信号	执行对句柄地址的跳转。	
未屏蔽的可屏蔽中断请求信号	执行对句柄地址的跳转或执行下一条指令。	执行下一条指令。

（2）通过 $\overline{\text{RESET}}$ 引脚输入或 WDTRES 信号的产生来解除 HALT 模式

执行的操作与正常的复位操作相同。

表 21-3. HALT 模式下的操作状态

项目 \ HALT 模式的设置		操作状态
时钟发生器，PLL		操作中
系统时钟(f_{XX})		提供
CPU		停止操作
外部总线接口[注]		参见 **表 2-2 操作模式下的引脚操作状态**
DMA		可操作
中断控制器		可操作
定时器	TAA0 至 TAA4	可操作
	TAB0, TAB1	可操作
	TMT0, TMT1	可操作
	TMM0 至 TMM3	可操作
看门狗定时器		可操作
串行接口	CSIB0 至 CSIB2	可操作
	UARTA0 至 UARTA2	可操作
	UARTB	可操作
	I^2C	可操作
A/D 转换器 0 至 2		可操作
时钟监视器		可操作
低电压检测器		可操作
上电清零电路		可操作
端口功能		保持设置为 HALT 模式前的状态。
内部数据		CPU 寄存器、状态、数据及其它内部数据，例如内部 RAM 的内容，被保持为设置 HALT 模式之前的状态。

<R> 注 仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

21.4 IDLE 模式

21.4.1 设置和操作状态

在正常操作模式下，通过清除（0）PSMR.PSM0 位，并设置（1）PSC.STB 位，可以设置进入 IDLE 模式。

在 IDLE 模式下，时钟发生器和 PLL 继续操作，但停止 CPU 和其它片上外围功能的时钟供给。

结果，程序停止执行，且内部 RAM 保持与设置为 IDLE 模式前的内容。CPU 及其它片上外围功能停止工作。但是，可以在外部时钟下工作的片上外围功能继续工作。

表 21-5 显示了 IDLE 模式下的操作状态。

与 HALT 模式相比，IDLE 模式可以减少更多功耗，因为停止了片上外围功能的运行。时钟发生器和 PLL 不停止，所以在解除 IDLE 模式后，无需等待振荡稳定时间就可以恢复到正常操作模式，HALT 模式解除时也是如此。

注意事项　**为了设置 IDLE 模式，需要将数据存储到 PSC 寄存器，在存储指令之后，应该插入五个或更多 NOP 指令。**

21.4.2 解除 IDLE 模式

IDLE 模式的解除，可以通过来自 IDLE 模式下可操作的外围功能的未屏蔽外部中断请求信号(INTP00, INTP01, INTP02 至 INTP07 (仅限 V850E/IG3), INTP08 至 INTP13, INTP17, INTP18, INTADT0 或 INTADT1 引脚输入)，未屏蔽内部中断请求信号(从模式下的 CSIB 相关的中断请求信号)，或复位信号($\overline{\text{RESET}}$ 引脚输入、低电压检测器(LVI)产生的复位信号(LVIRES)或上电清零电路(POC)产生的复位信号(POCRES))。

IDLE 模式释放后，恢复为正常操作模式。

（1）通过未屏蔽的可屏蔽中断请求信号来解除 IDLE 模式

通过未屏蔽的可屏蔽中断请求信号，可以解除 IDLE 模式，且与中断请求信号的优先级无关。如果在一个中断服务程序中设置 IDLE 模式，则之后产生的中断请求的处理如下：

注意事项　**当 PSC.INTM 位 = 1 时，不能通过未屏蔽的可屏蔽中断请求信号来解除 IDLE 模式。**

（a）如果产生的中断请求信号优先级低于或等于当前正在服务的中断请求，则解除 IDLE 模式，但是不响应这个新产生的中断请求。该中断请求信号被保持。因此，因此，在 IDLE 指令之后的下一条指令开始执行。

（b） 如果发出的中断请求信号优先级高于当前正在服务的中断（包括不可屏蔽中断请求信号），则解除 IDLE 模式，且响应该中断请求信号。

表 21-4.通过中断请求信号解除 IDLE 模式后的操作

解除源	中断允许（EI）状态	中断禁止（DI）状态
可屏蔽中断请求信号	执行对句柄地址的跳转或执行下一条指令。	执行下一条指令。

（2）通过 $\overline{\text{RESET}}$ 引脚输入来解除 IDLE 模式

执行的操作与正常的复位操作相同。

表 21-5. IDLE 模式下的操作状态

项目 \ IDLE 模式的设置		操作状态
时钟发生器，PLL		操作
系统时钟(f_{XX})		停止提供
CPU		停止操作
外部总线接口[注]		参见 **表 2-2 操作模式下的引脚操作状态**
DMA		停止操作
中断控制器		停止操作
定时器	TAA0 至 TAA4	停止操作
	TAB0, TAB1	停止操作
	TMT0, TMT1	停止操作
	TMM0 至 TMM3	停止操作
看门狗定时器		停止操作
串行接口	CSIB0 至 CSIB2	当 $\overline{\text{SCKBn}}$ 输入时钟被选为计数时钟（在从模式下）时，可操作(n = 0 至 2)
	UARTA0 至 UARTA2	停止操作
	UARTB	停止操作
	I^2C	停止操作
A/D 转换器 0 至 2		停止操作
时钟监视器		可操作
低电压检测器		可操作
上电清零电路		可操作
端口功能		保持设置为 IDLE 模式前的状态
内部数据		CPU 寄存器、状态位、数据及其它内部数据，例如内部 RAM 的内容，都保持为设置 IDLE 模式之前的状态。

<R>　注　仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

21.5 STOP 模式

21.5.1 设置和操作状态

在正常操作模式下，通过设置（1）PSMR.PSM0 位，并设置（1）PSC.STB 位，可以设置进入 STOP 模式。

在 STOP 模式下，时钟发生器停止操作。停止对 CPU 和片上外围功能的时钟供给。

结果，程序停止执行，且内部 RAM 保持为设置 STOP 模式之前的内容。CPU 和片上外围功能停止操作。但是，运行于外部时钟的片上外围功能继续工作。

表 21-7 显示了在 STOP 模式下的操作状态。

因为 STOP 模式使时钟发生器停止操作，所以可将功耗降低到低于 IDLE 模式的水平。如果不使用外部时钟，功耗达到最小化，仅有漏电流通过。

注意事项　　为了设置 STOP 模式，需要将数据存储到 PSC 寄存器，在存储指令之后，应该插入五个或更多 NOP 指令。

21.5.2 解除 STOP 模式

STOP 模式的解除，可以通过来自 STOP 模式下可操作的外围功能的未屏蔽外部中断请求信号(INTP00, INTP01, INTP02 至 INTP07 (仅限 V850E/IG3), INTP08 至 INTP13, INTP17, INTP18, INTADT0 或 INTADT1 引脚输入)，未屏蔽内部中断请求信号(从模式下的 CSIB 相关的中断请求信号)，或复位信号($\overline{\text{RESET}}$ 引脚输入、低电压检测器(LVI)产生的复位信号(LVIRES)或上电清零电路(POC)产生的复位信号(POCRES))。

STOP 模式释放后，在确保经过振荡稳定时间之后，恢复为正常操作模式。

（1）通过未屏蔽的可屏蔽中断请求信号来解除 STOP 模式

通过未屏蔽中断请求信号，可解除 STOP 模式，与中断请求信号的优先级无关。但是，如果在一个中断服务程序中设置 STOP 模式，则对之后发出的中断请求信号的服务如下。

注意事项　当 PSC.INTM 位 = 1 时，不能通过未屏蔽的可屏蔽中断请求信号来解除 STOP 模式。

（a）如果产生的中断请求信号优先级低于或等于当前正在服务的中断请求，则解除 STOP 模式，但是不响应这个新产生的中断请求。该中断请求信号被保持。因此，因此，在 STOP 指令之后的下一条指令开始执行。

（b）如果发出的中断请求信号优先级高于当前正在服务的中断请求，则解除 STOP 模式，且响应该中断请求信号。因此，执行跳转到句柄地址。

表 **21-6.** 通过中断请求信号解除 **STOP** 模式之后的操作

解除源	中断允许（EI）状态	中断禁止（DI）状态
可屏蔽中断请求信号	在经过确保的振荡稳定时间之后，执行对句柄地址的跳转或执行下一条指令。	在经过确保的振荡稳定时间之后，执行下一条指令。

（2）通过 $\overline{\text{RESET}}$ 引脚输入来解除 STOP 模式

执行的操作与正常的复位操作相同。

表 **21-7. STOP** 模式的操作状态

项目 \ STOP 模式的设置		操作状态
时钟发生器，PLL		停止操作
系统时钟(f_{XX})		停止供给
CPU		停止操作
外部总线接口[注]		参见 **表 2-2 操作模式下的引脚操作状态**
DMA		停止操作
中断控制器		停止操作
定时器	TAA0 至 TAA4	停止操作
	TAB0, TAB1	停止操作
	TMT0, TMT1	停止操作
	TMM0 至 TMM3	停止操作
看门狗定时器		停止操作
串行接口	CSIB0 至 CSIB2	当 $\overline{\text{SCKBn}}$ 输入时钟被选为计数时钟（在从模式下）时，可操作(n = 0 至 2)
	UARTA0 至 UARTA2	停止操作
	UARTB	停止操作
	I^{2}C	停止操作
A/D 转换器 0 至 2		停止操作
时钟监视器		停止操作
低电压检测器		可操作
上电清零电路		可操作
端口功能		保持设置为 STOP 模式前的状态。
内部数据		CPU 寄存器、状态、数据及其它内部数据，例如内部 RAM 的内容，都保持为设置 STOP 模式之前的状态。

<R>　注　仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

21.6 确保振荡稳定时间

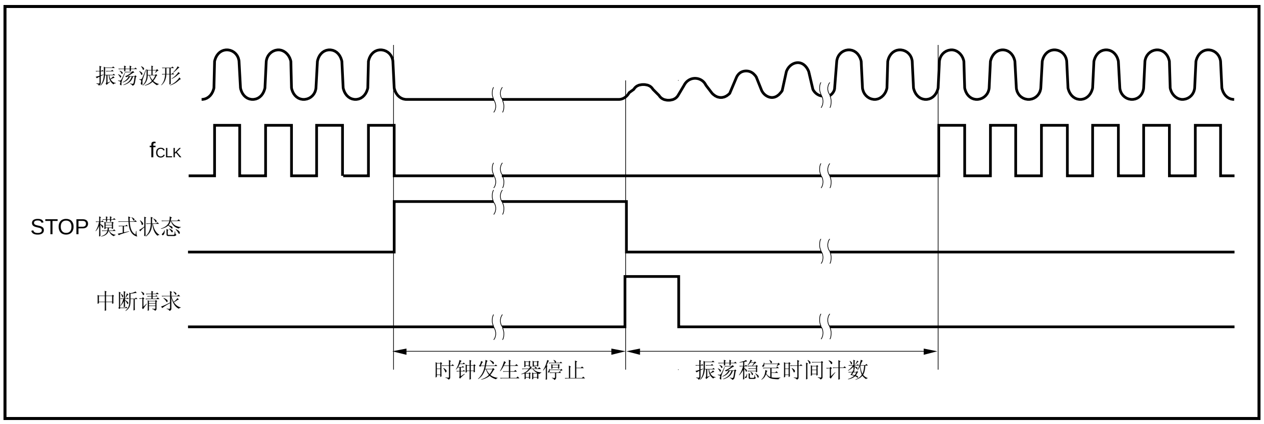
解除STOP模式时，需要经过一段由OSTS寄存器设置的振荡稳定时间。如果STOP模式是通过$\overline{\text{RESET}}$引脚输入来解除，则振荡稳定时间是OSTS寄存器的复位值，$2^{14}/f_X$ (当f_X = 8 MHz时，2.048 ms)。

但是，实际的振荡稳定时间是该值的一半（复位后：$2^{13}/f_X$ (当f_X = 8 MHz时，1.024 ms)），另一半的时间是PLL的稳定时间。解除STOP模式时，振荡稳定时间的设置应该是所使用振荡器的振荡稳定时间的两倍。如果通过$\overline{\text{RESET}}$引脚输入解除STOP模式时，使用的振荡器所需的振荡稳定时间长于 $2^{13}/f_X$ ，使用$\overline{\text{RESET}}$信号的低电平宽度来确保振荡稳定时间。

振荡稳定时间的计数定时器确保振荡稳定时间等于看门狗定时器的溢出时间。

通过一个中断请求信号解除 STOP 模式时，执行的操作如下。

图 21-2. 振荡稳定时间



注意事项　OSTS 寄存器的详情，参见 5.3 (5) 振荡稳定时间选择寄存器(OSTS)。

第二十二章　复位功能

22.1 概述

- $\overline{\text{RESET}}$ 引脚输入引发的系统复位
- 看门狗定时器(WDT)溢出产生的系统复位信号 (WDTRES)
- 低电压检测器(LVI)产生的系统复位信号 (LVIRES)
- 上电清零电路 (POC)产生的系统复位信号 (POCRES)
- 通过片上调试功能和复位屏蔽功能的强制复位(参见 **第二十六章　片上调试功能**)

22.2 控制寄存器

(1) 复位源标志寄存器 (RESF)

RESF 寄存器是一个 8 位寄存器，用于表示发生的一个复位请求来自看门狗定时器(WDT)或低电压检测器(LVI)。当来自 WDT 或 LVI 的内部复位源信号有效时，该寄存器的 WDTRF 或 LVIRF 位被设置为 1。WDTRF 或 LVIRF 位被 $\overline{\text{RESET}}$ 引脚引发的复位清除，或使用位操作指令或存储操作指令清除 (向 WDTRF 或 LVIRF 位写入 0)。

RESF 寄存器是特殊寄存器，它只能通过一个特定序列的组合来写入（参见 **3.4.8 特殊寄存器**）。

该寄存器可按字节或按位进行读取或写入。但是，第 0 位和第 4 位只能通过写入来清除(0)。

$\overline{\text{RESET}}$ 引脚输入和上电清零电路 (POC)引发的复位会将该寄存器清为 00H。如果复位源不是 $\overline{\text{RESET}}$ 引脚输入和上电清零电路 (POC)引发的复位信号，其默认值将不同。复位冲突的详情，参见下面的**注意事项**。

复位后： 00H注　　R/W　　地址： FFFFF888H

	7	6	5	4	3	2	1	0
RESF	0	0	0	WDTRF	0	0	0	LIVRF

WDTRF	产生看门狗定时器(WDT)引发的复位信号
0	读取：没有复位请求， 写入：清除
1	复位请求

LIVRF	产生低电压检测器(LVI)引发的复位信号
0	读取：没有复位请求， 写入：清除
1	复位请求

注　$\overline{\text{RESET}}$ 引脚输入或上电清零电路 (POC)引发的复位之后：00H
看门狗定时器溢出引发的复位之后：10H
低电压检测器(LVI)引发的复位之后：01H

注意事项　**如果 RESF 寄存器的设置(发生设置源的复位)和清除 (系统复位的发生或向 WDTRF 或 LVIRF 位写入 0)发生冲突，优先级如下：**

1. 产生 $\overline{\text{RESET}}$ 引脚输入引发的复位 (清除 RESF 寄存器)
2. 产生 WDT 或 LVI 引发的复位 (设置 RESF 寄存器)
3. 通过位操作或存储操作指令向 WDTRF 或 LVIRF 位写入 0 (清除 RESF 寄存器)

如果 $\overline{\text{RESET}}$ 引脚输入引发的复位和 WDT 或 LVI 引发的复位发生冲突，RESF 寄存器不置位而是清零 (00H)。

22.3 操作

(1) $\overline{\text{RESET}}$ 引脚输入的复位操作

当一个低电平输入 $\overline{\text{RESET}}$ 引脚时，V850E/IF3 和 V850E/IG3 被复位，各个硬件单元被初始化为指定状态。

即使当一个低电平输入到 $\overline{\text{RESET}}$ 引脚，但是振荡模式被初始化为时钟直通模式(PLLCTL 寄存器 = 01H)，且 CPU 时钟 (f_{CPU}) 分频是 $f_{XX}/8$ (PCC 寄存器 = 03H)，振荡器继续振荡。

当$\overline{\text{RESET}}$ 引脚输入从低变为高时，复位状态释放。在复位状态被释放之后，经过振荡器的振荡稳定时间和PLL 的入锁时间 (OSTS寄存器的默认值设置总体时间：$2^{14}/f_X$ (2.05 ms (f_X = 8 MHz))，然后CPU开始执行程序。因此，在复位被释放后，在时钟直通模式启动操作，且运行于$f_{XX}/8$。

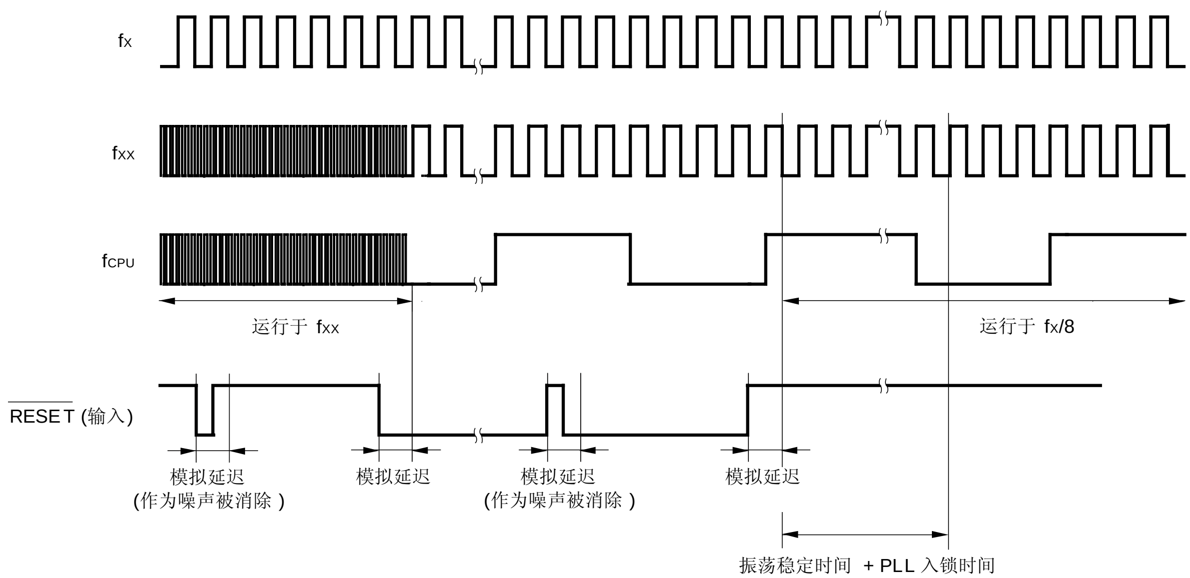
复位期间和复位被释放后各个硬件单元的状态如下所示。

硬件	复位期间	复位被释放后
时钟发生器： 振荡器(f_X) 内部系统时钟 (f_{CLK}) CPU 时钟 (f_{CPU}) 外部总线时钟 (f_{BUS})注	振荡/供给继续 但是，CPU 时钟 (f_{CPU})被初始化为 $f_{XX}/8$	
时钟发生器： 外围时钟 (f_{XX} 至 $f_{XX}/4096$)	振荡/供给停止	确保振荡稳定时间之后，振荡/供给启动
时钟发生器： 看门狗定时器时钟 ($f_{XX}/1024$)	振荡/供给停止	振荡/供给启动
CPU	被初始化	确保振荡稳定时间之后，开始执行程序
内部 RAM	只有在 STOP 模式下复位输入期间，保持复位输入之前的瞬时值。否则，不确定。	
端口 (包括复用功能引脚)	高阻抗	
片上外围 I/O 寄存器(除端口外)	初始化为指定状态	
上述之外的片上外围功能	停止操作	可以启动操作

<R> 注 仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

$\overline{\text{RESET}}$ 引脚输入引发的复位操作展示如下。

图 22-1. $\overline{\textbf{RESET}}$ 引脚输入的复位操作



注意事项　**在复位释放后，确保经过振荡稳定时间 (1.024 ms (当 f_X = 8 MHz 时)) 和 PLL 入锁时间 (1.024 ms (当 f_X = 8 MHz 时)) 。如果 1.024 ms 的振荡稳定时间不足以确保振荡稳定，保持** $\overline{\textbf{RESET}}$ **引脚为低的时间来补足。**

备注　在上述时序图中的 f_{XX} 和 f_X 之间的关系是 $f_{XX} = 8 \times f_X$

在 PLL 模式和时钟直通模式下，复位释放后的操作相同，都作为时钟直通模式启动。通过软件控制来设置 PLL 模式 (将 PLLCTL.SELPLL 位设置为 1)。要改善抗噪性能，推荐设置 PLL 模式，然后提高 CPU 时钟速度(示例：PCC 寄存器 = 00H (f_{XX} 操作))。

(2) 看门狗定时器(WDT)溢出引发的复位操作(WDTRES)

如果设置的复位模式是在看门狗定时器(WDT) 溢出时(WDTM.WDM1 和 WDTM.WDM0 位 = 10 或 11)进行复位，当 WDT 溢出时(INTWDT)，系统被复位，各个硬件单元被初始化为指定状态。

如果产生 INTWDT 中断请求信号，RESF.WDTRF 位被设置为 1，表示已经发生内部复位。

除了 RESF 寄存器的操作，在复位期间和复位被释放后，各个硬件的状态和 $\overline{\text{RESET}}$ 引脚引发的复位操作相同(参见 **(1) $\overline{\text{RESET}}$ 引脚输入引发的复位操作**)。

(3) 低电压检测器(LVI)引发的复位操作(LVIRES)

当 LVI 操作使能时，供电电压(V_{DD0}, V_{DD1})和检测电压(V_{LVI})进行比较，如果供电电压跌落到检测电压以下，系统被复位 (当 LVIM.LVIMD 位被设置为“1”)，且各个硬件单元被初始化为指定状态。

当V_{DD0}, V_{DD1} < V_{LVI} 时，系统被复位，当V_{DD0}, V_{DD1} ≥ V_{LVI} 时，复位被释放。复位被释放之后，经过振荡器的振荡稳定时间 (OSTS寄存器的默认值：2^{14}/fx)，CPU开始执行程序。

在复位期间，振荡器停止，所以要确保振荡稳定时间。

复位期间和复位被释放后，各个硬件的状态和 $\overline{\text{RESET}}$ 引脚引发的复位操作相同 (参见 **(1) $\overline{\text{RESET}}$ 引脚输入引发的复位操作**)。

低电压检测器(LVI)引发的复位操作详情，参见 **第二十三章 低电压检测器**。

(4) 上电清零电路 (POC)引发的复位操作(POCRES)

当供电电压(V_{DD0}, V_{DD1})和检测电压(V_{POC0})进行比较时，如果供电电压跌落到检测电压以下(包括上电)，系统复位，且各个硬件单元被初始化为指定状态。

当V_{DD0}, V_{DD1} < V_{POC0} 时，系统复位，当V_{DD0}, V_{DD1} ≥ V_{POC0}时，复位被释放。复位被释放之后，经过振荡器的振荡稳定时间 (OSTS寄存器的默认值：2^{14}/fx)，CPU开始执行程序。

在复位期间，振荡器停止，所以要确保振荡稳定时间。

复位期间和复位被释放后，各个硬件的状态和 $\overline{\text{RESET}}$ 引脚引发的复位操作相同 (参见 **(1) $\overline{\text{RESET}}$ 引脚输入引发的复位操作**)。

上电清零电路 (POC)引发的复位操作详情，参见 **第二十四章 上电清零电路**。

第二十三章　低电压检测器（LVI）

23.1 功能

低电压检测器（LVI）具有以下功能。

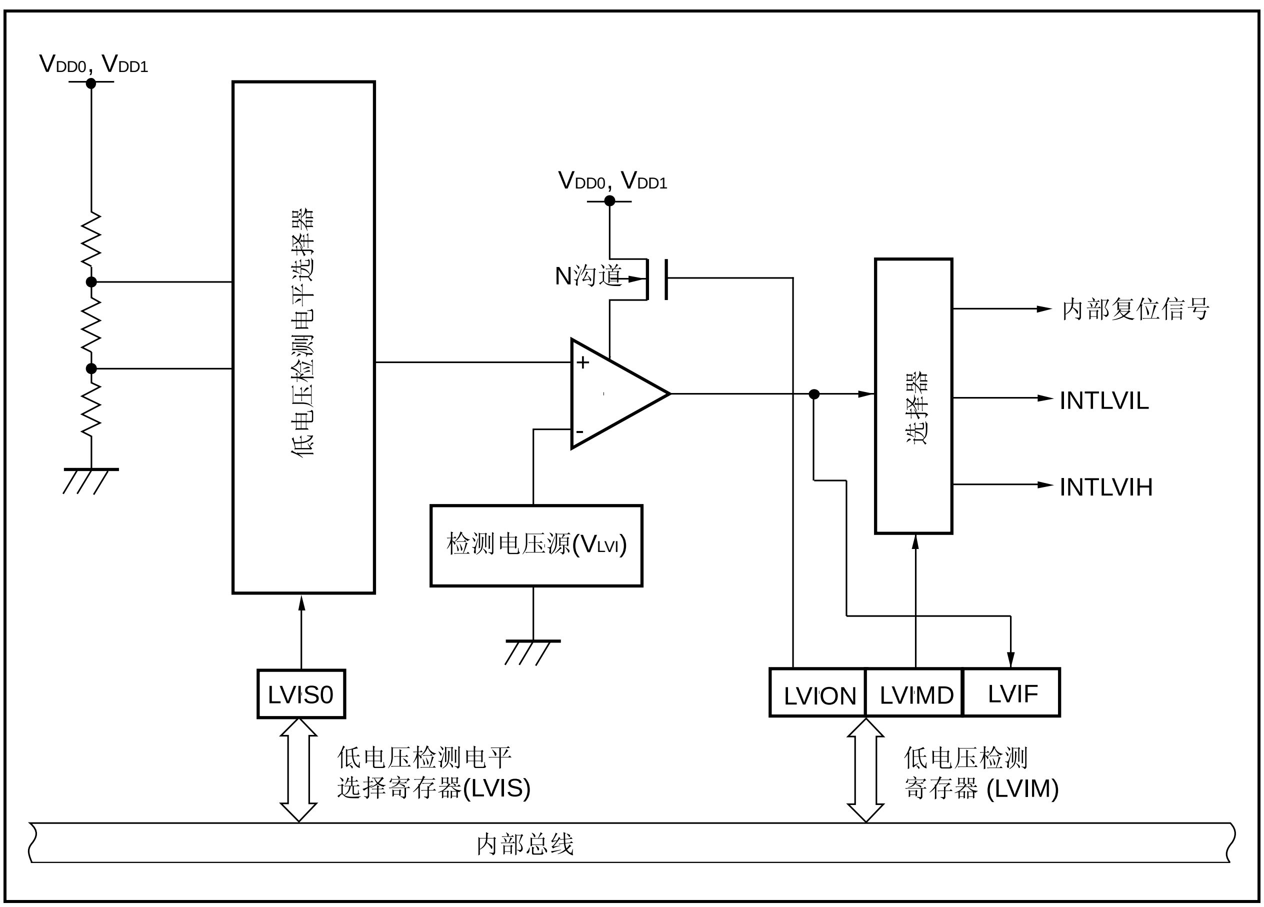
- 比较供电电压(V_{DD0}, V_{DD1})和检测电压（V_{LVI}），并在 V_{DD0}, V_{DD1} < V_{LVI} 时，产生一个中断请求信号(INTLVIL, INTLVIH)或内部复位信号(LVIRES)。
- 待检测的供电电压电平可以通过软件来改变(分为两级)。
- 可以选择中断请求信号(INTLVIL, INTLVIH)或内部复位信号(LVIRES)。
- 可以在 STOP 模式下工作。
- 可以通过软件停止其工作。

如果低电压检测器用于产生复位信号，当复位信号产生时，RESF.LVIRF 位设置为 1。关于 RESF 寄存器的详情，参见**第二十二章 复位功能**。

23.2 配置

低电压检测器的框图如下所示。

图 23-1. 低电压检测器的框图



23.3 控制寄存器

（1）低电压检测寄存器 （LVIM）

LVIM 寄存器用于使能或禁止低电压检测，并且设置低电压检测器的工作模式。LVIM 是一个专用寄存器。只有通过特定序列的组合才能写入该寄存器（参见 **3.4.78 专用寄存器** ）。

该寄存器可以按字节或按位进行读取或写入，但是，第 0 位是只读的。

低电压检测器（LVI）引发的复位之外的复位，会将该寄存器设置为 00H。

复位后： 00H　　R/W　　地址： FFFFF890H

	<7>	6	5	4	3	2	<1>	<0>
LVIM	LVION	0	0	0	0	0	LVIMD	LVIF

LVION	使能或禁止低电压检测操作
0	禁止工作
1	使能工作

LVIMD	选择低电压检测的工作模式
0	当供电电压 < 检测电压时，产生中断请求信号 INTLVIL 当供电电压 > 检测电压时，产生中断请求信号 INTLVIH
1	当供电电压 < 检测电压时，产生内部复位信号 LVIRES

LVIF	低电压检测标志
0	当供电电压 > 检测电压，或操作禁止时
1	当供电电压 < 检测电压时

注意事项

1. 将 **LVION** 位设置为 **1** 后，在使用 **LVIF** 位检查电压之前，等待 **TBD**。

2. 当 **LVION** 位 **= 1** 且 **LVIMD** 位 **= 0** 时，**LVIF** 标志的值输出作为输出信号 **INTLVIL** 或 **INTLVIH**。

3. 如果 **LVION** 位 **= 1** 且 **LVIMD** 位 **= 1**，低电压检测器**(LVI)**不能停止，除非 **LVI** 之外的其他复位源产生复位信号。

4. 请确保将第 **2** 位至第 **6** 位清除为 **"0"**。

（2）低电压检测电平选择寄存器 （LVIS）

LVIS 寄存器用于选择低电压检测的检测电平值。

该寄存器能够按字节进行读取或写入。

低电压检测器（LVI）引发的复位之外的复位，会将该寄存器设置为 00H。

复位后： 00H　　R/W　　地址： FFFFF891H

	7	6	5	4	3	2	1	0
LVIS	0	0	0	0	0	0	0	LVIS0

LVIS0	检测电平
0	4.4 V ± 0.2 V
1	4.2 V ± 0.2 V

注意事项 **1.** 在 **LVIM.LVION** 和 **LVIM.LVIMD** 位被设置为 **1** 之后，低电压检测器（**LVI**）之外的其他复位源产生复位请求，否则 **LVIS** 寄存器不能写入。

2. 请确保将第 **1** 位至第 **7** 位清除为**“0”**。

23.4 操作

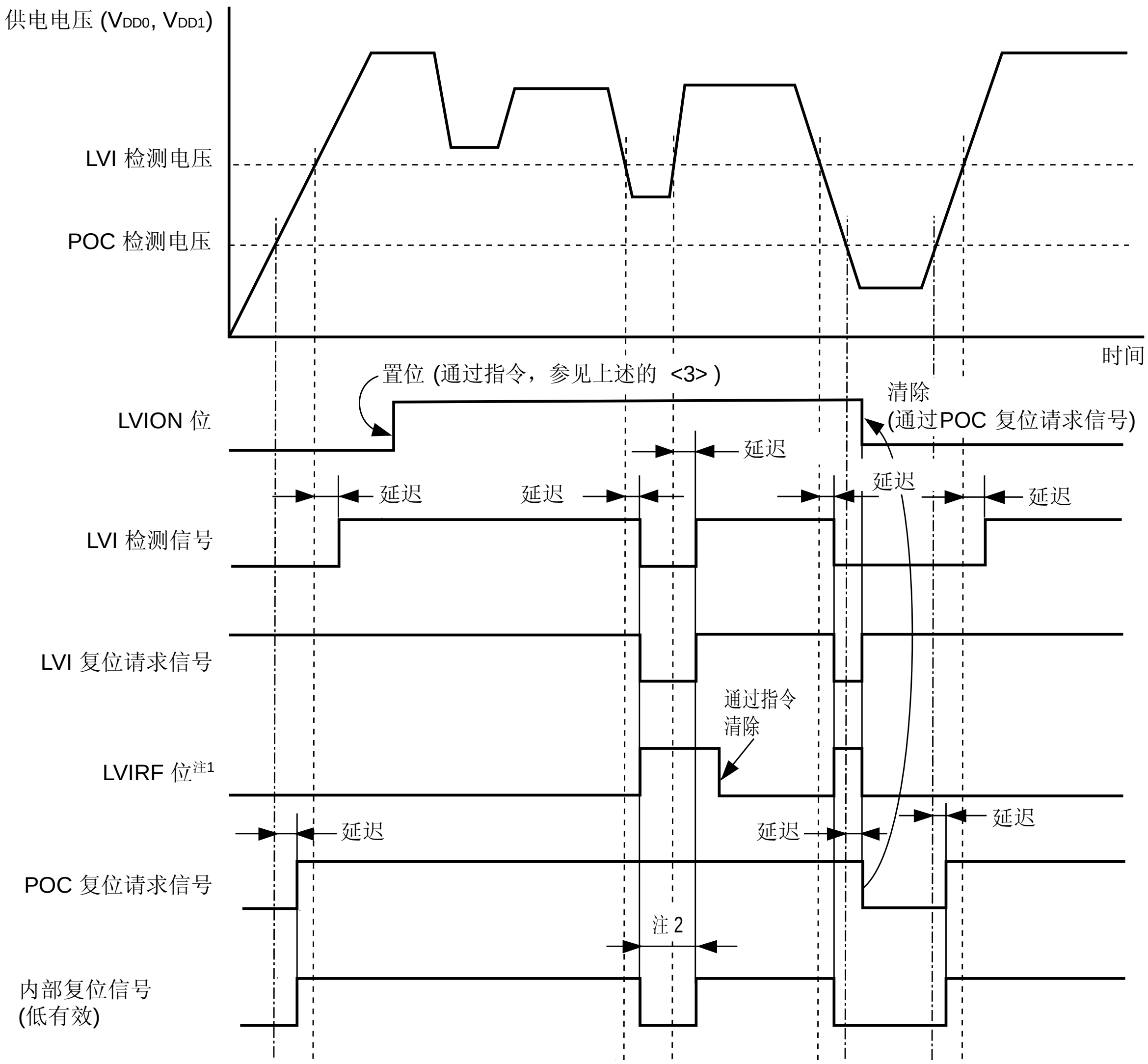
根据 LVIM.VIMD 位的设置，产生中断请求信号(INTLVIL, INTLVIH)或内部复位信号(LVIRES)。

23.4.1 用作内部复位信号

<开始操作>
<1> 屏蔽低电压检测器（LVI）的中断。
<2> 使用 LVIS.LVIS0 位选择待检测的电压。
<3> 将 LVIM.LVION 位设置为 1（使能操作）。
<4> 通过软件插入一个 TBD 或更长的等待周期。
<5> 使用 LVIM.LVIF 位检测是否供电电压 > 检测电压。
<6> 将 LVIMD 位设置为 1 （产生一个内部复位信号）。

注意事项 **如果 LVIMD 位设置为 1，LVIM 和 LVIS 寄存器的内容不能改变，除非低电压检测器（LVI）之外的其他复位源产生复位请求。**

图 23-2. 低电压检测器的操作时序（LVIMD 位= 1）



注　**1.** LVIRF 位是复位源标志寄存器(RESF)的第 0 位。关于 RESF 的详情，参见**第二十二章 复位功能**。

　　2. 供电电压为设置电压或更低的期间，内部复位信号被保持(内部复位状态)。

23.4.2 用作中断

<开始操作>

<1> 屏蔽低电压检测器（LVI）的中断。

<2> 使用 LVIS.LVIS0 位选择待检测的电压。

<3> 将 LVIM.LVION 位设置为 1（使能操作）。

<4> 通过软件插入一个 TBD 或更长的等待周期。

<5> 使用 LVIM.LVIF 位检测是否供电电压 > 检测电压。

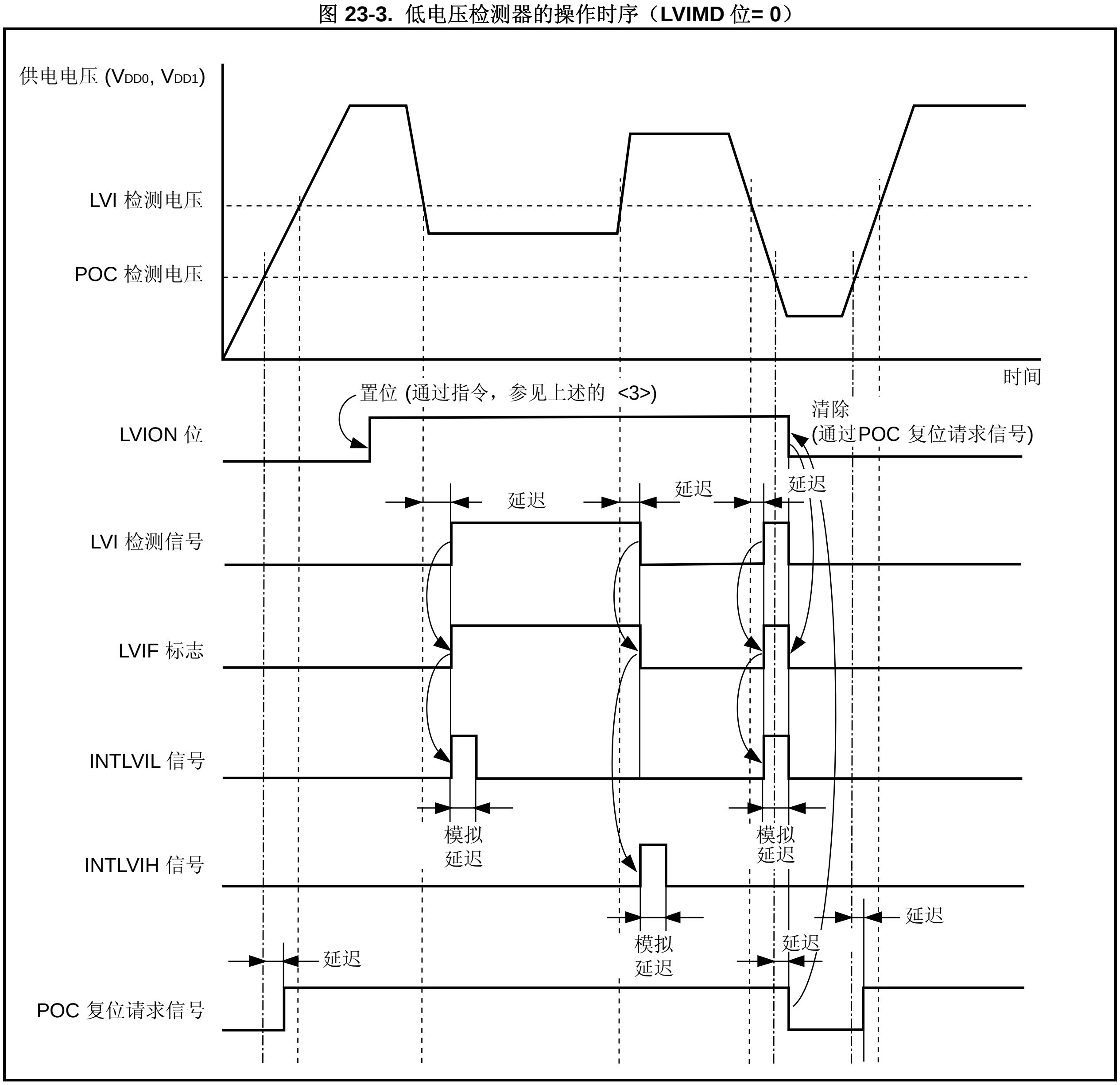
<6> 清除 LVI 的中断请求标志。

<7> 解除 LVI 中断的屏蔽。

<停止操作>

将 LVION 位清除为 0。

图 23-3. 低电压检测器的操作时序（LVIMD 位= 0）



第二十四章　上电清零电路

24.1 功能

上电清零电路(POC)的功能如下。

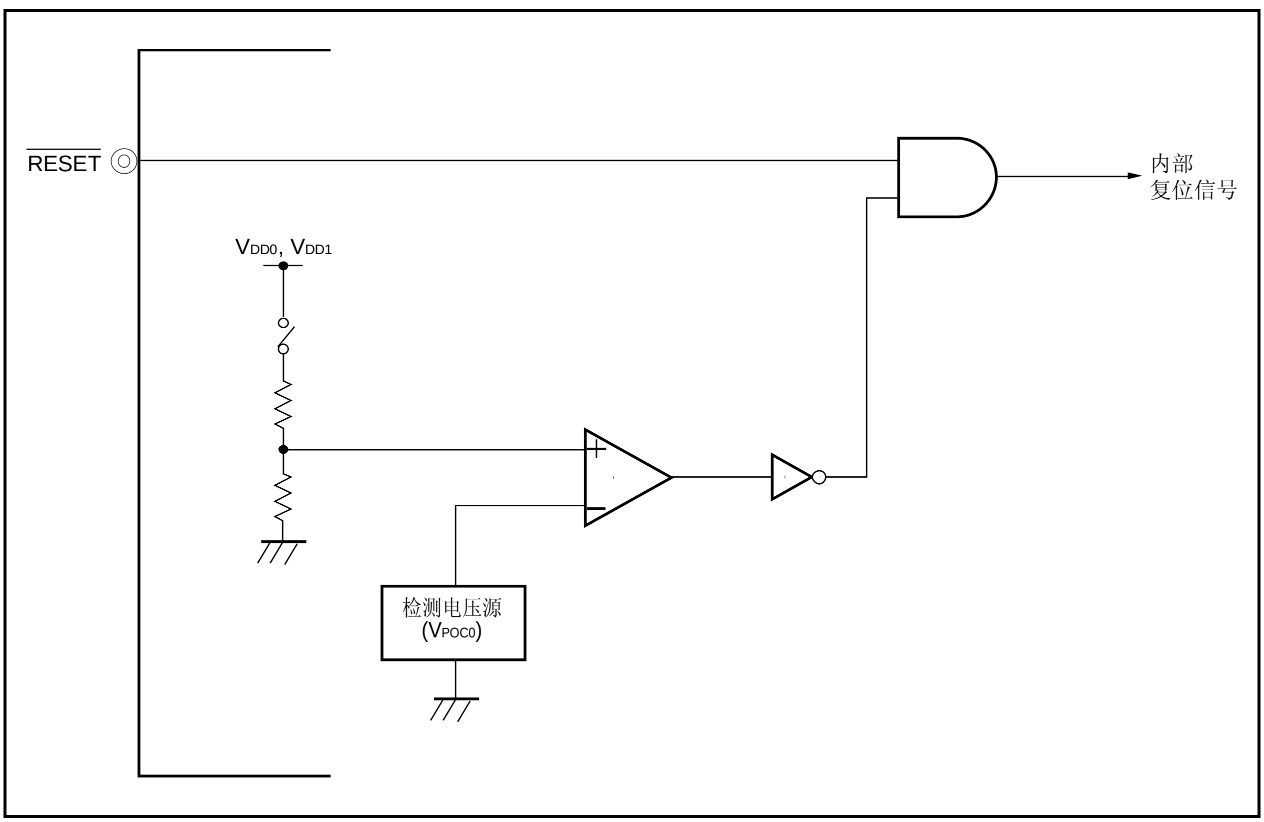
- 在上电时产生复位信号 (POCRES)
- 比较供电电压（V_{DD0}, V_{DD1}）和检测电压（V_{POC0}），并在 V_{DD0}, V_{DD1} < V_{POC0} 时(检测电压(V_{POC0})：3.7 V ± 0.2 V)产生复位信号。

备注　V850E/IF3 和 V850E/IG3 具有复位源标志寄存器(RESF)，用于指示产生一个看门狗定时器溢出引发的复位信号(WDTRES) 和一个低电压检测器(LVI)引发的复位信号(LVIRES)。
当产生复位信号(WDTRES 或 LVIRES)时，RESF 寄存器不会被清除为 00H，且对应复位源的标志位被设置为 1。
当产生上电清零电路(POC)引发的复位信号(POCRES)时，RESF 寄存器被清除（00H）。
如需了解 RESF 的详细内容，参见**第二十二章　复位功能**。

24.2 配置

框图如下所示。

图 24-1. 上电清零电路的框图



24.3 操作

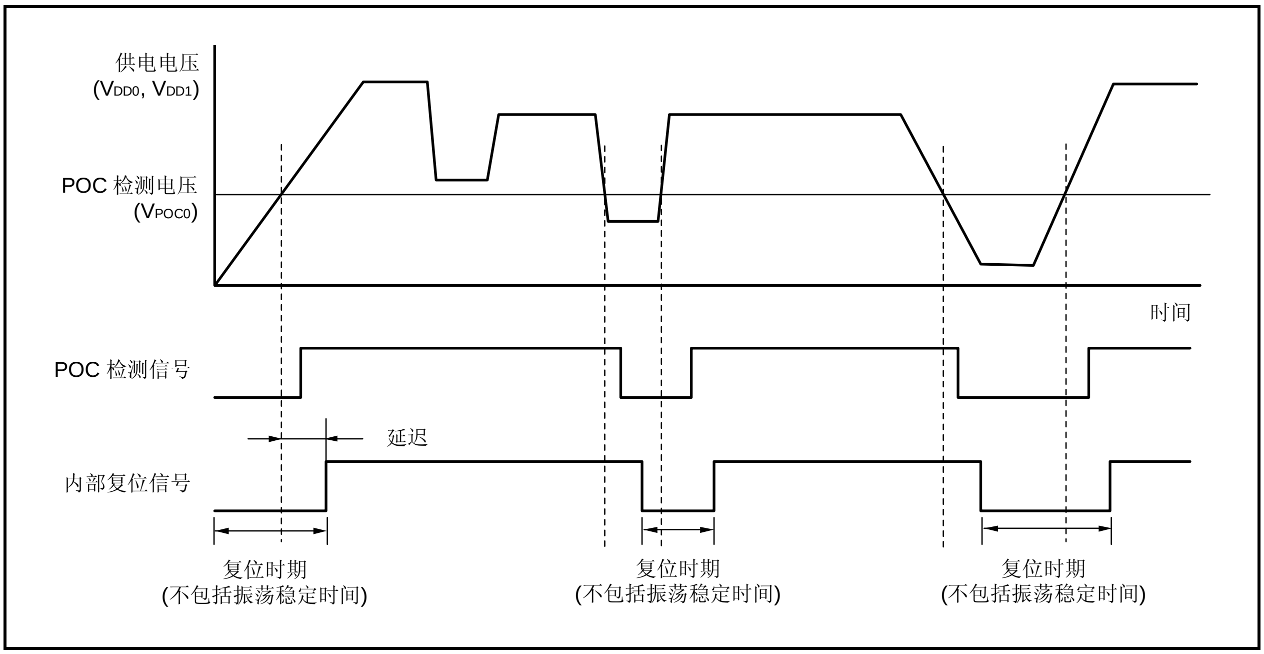
当供电电压和检测电压进行比较时，如果供电电压跌落到检测电压以下(包括上电)，系统被复位，且各个硬件单元被初始化为指定状态。

从检测到低电压直到供电电压高于检测电压，系统被复位。复位被释放之后，经过振荡器的振荡稳定时间 (OSTS寄存器的默认值：2^{14}/fx)，CPU开始执行程序。

在复位期间和复位被释放后，各个硬件的状态和 $\overline{\text{RESET}}$ 引脚引发的复位操作相同 (参见 **22.3 (1) $\overline{\text{RESET}}$ 引脚输入引发的复位操作**)。

时序图展示如下。

图 **24-2.** 上电清零电路产生复位信号的时序

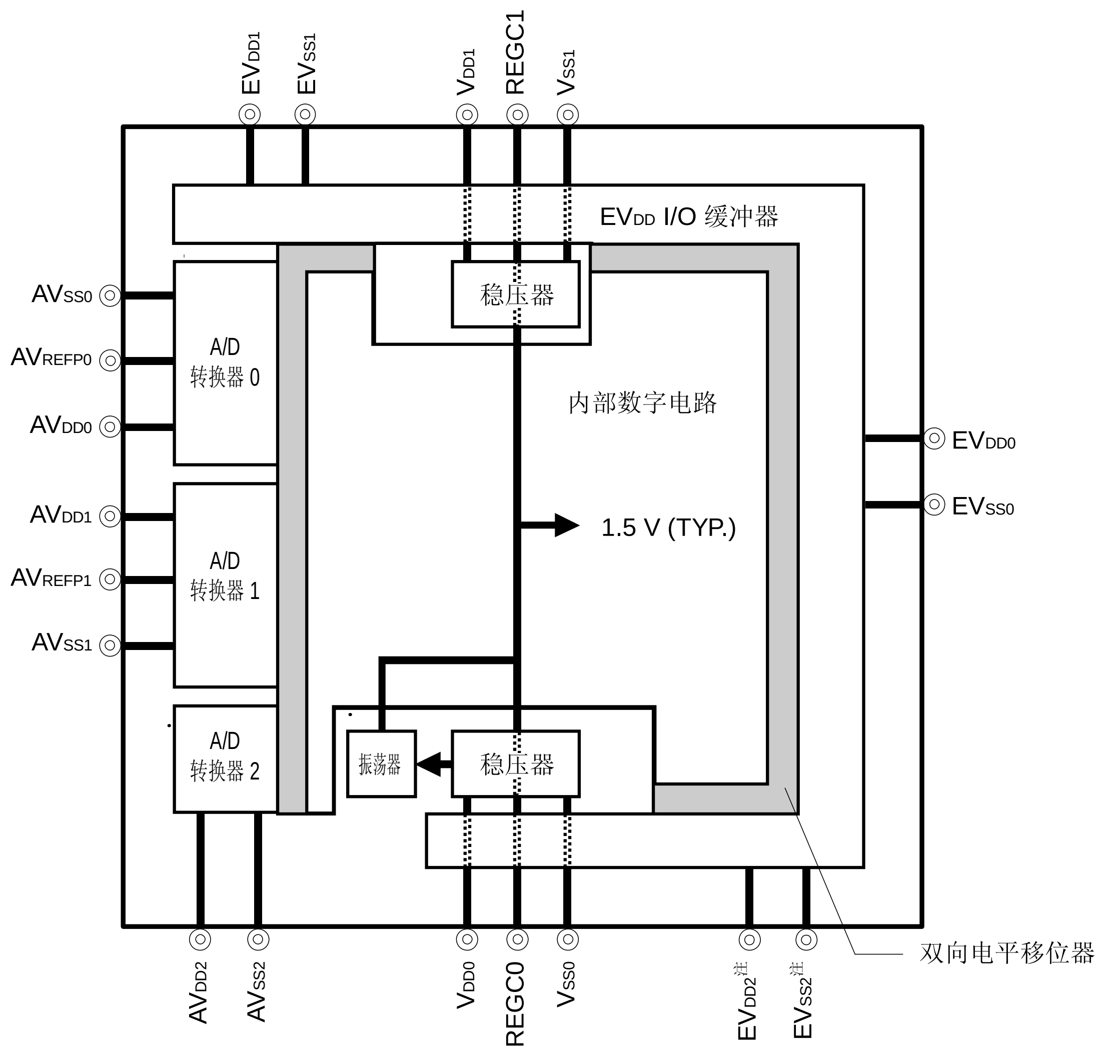


25.1 概述

V850E/IF3 和 V850E/IG3 具有一个内部稳压器，用于实现 5 V 单电源供应操作。

该稳压器为振荡模块和内部逻辑电路（除了 A/D 转换器 0 至 2 和 I/O 缓存器之外）提供一个阶梯下降的 V_{DD0} 和 V_{DD1} 电源供电电压。稳压器输出电压 (REGC0, REGC1 引脚)被设置为 1.5 V（典型值）。

图 25-1. 稳压器



注　　仅限 V850E/IG3

注意事项　使用稳压器应该设置 V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = EV_{DD2} (仅限 V850E/IG3) = AV_{DD0} = AV_{DD1} = AV_{DD2} = AV_{REF0} = AV_{REF1}

25.2 操作

V850E/IF3 和 V850E/IG3 的稳压器能在任何模式（比如正常工作模式、HALT 模式、IDLE 模式、STOP 模式或者在复位期间）下保持运行。

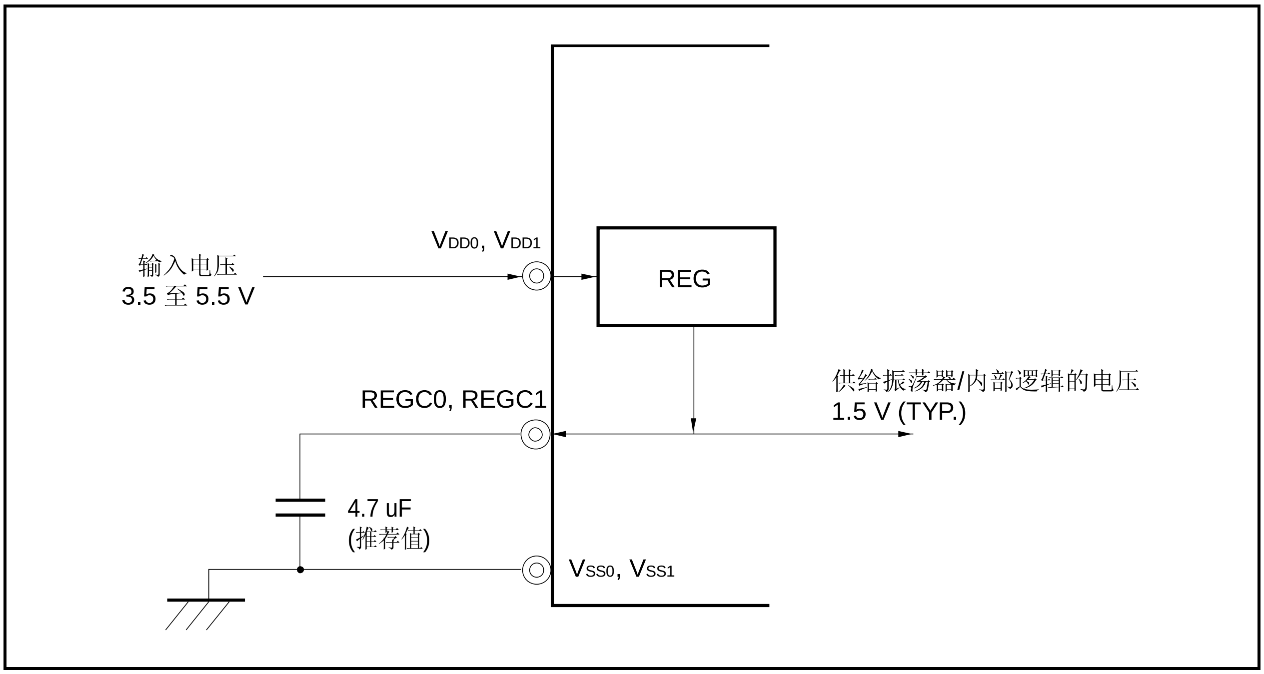
请确保在REGC0 和REGC1 引脚接一个电容（4.7μF（推荐值））[注]，来稳定该稳压器的输出。

稳压器引脚的连接方式如下图所示。

注　　使用串联电阻元件 ESR 的较低 ESR (0.5 Ω 或更低)。

注意事项　V850E/IF3 和 **V850E/IG3** 各有两个稳压器。因此，为 **REGC0** 和 **REGC1** 引脚各自连接一个电容。

图 **25-2. REGC0** 和 **REGC1** 引脚的连接



第二十六章　片上调试功能

V850E/IF3 和 V850E/IG3 单片机的片上调试功能可由下面两种方法实现。

- 使用 DCU 调试（调试控制单元）（使用 MINICUBE）
  使用 $\overline{\text{DRST}}$、DCK、DMS、DDI和DDO引脚作为调试接口引脚，通过V850E/IG3[注]的内部DCU来实现片上调试。
- 不使用 DCU 调试（使用 MINICUBE2）
  通过 MINICUBE2 而不使用 DCU 来实现片上调试，需要使用用户资源。

注　　V850E/IF3 不具有内部 DCU。

下表显示了两种片上调试功能的特性。

表 26-1. 片上调试功能特性

		使用 DCU 调试	不使用 DCU 调试
目标产品		V850E/IG3	V850E/IF3, V850E/IG3
调试接口引脚		$\overline{\text{DRST}}$, DCK, DMS, DDI, DDO	• 使用 UARTA0 时 RXDA0, TXDA0 • 使用 CSIB0 时 SIB0, SOB0, $\overline{\text{SCKB0}}$, HS (P43)
用户资源的保障		不需要	需要
硬件断点功能		2 点	2 点 (V850E/IG3 only)
软件断点功能	内部 ROM 区域	4 点	4 点
	RAM 区域	2000 点	2000 点
实时RAM 监控功能[注1]		可用	可用
动态内存修改（DMM）功能[注2]		可用	可用
屏蔽功能		Reset, INTWDT, $\overline{\text{WAIT}}$[注3]	$\overline{\text{RESET}}$，$\overline{\text{WAIT}}$[注3]
ROM 安全功能		10-字节 ID 码鉴定	10-字节 ID 码鉴定
使用的硬件		MINICUBE	MINICUBE2
跟踪功能		不支持	不支持
调试中断接口功能（DBINT）		不支持	不支持

注　**1.　程序执行时读出存储器内容的功能。**
　　2.　程序执行时重写 RAM 内容的功能。
<R>　3.　仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

26.1 使用 DCU 调试

通过调试接口引脚（$\overline{\text{DRST}}$, DCK, DMS, DDI 和 DDO）连接片上调试仿真器（MINICUBE），可以调试程序。

注意事项　只有 V850E/IG3 具有 DCU。

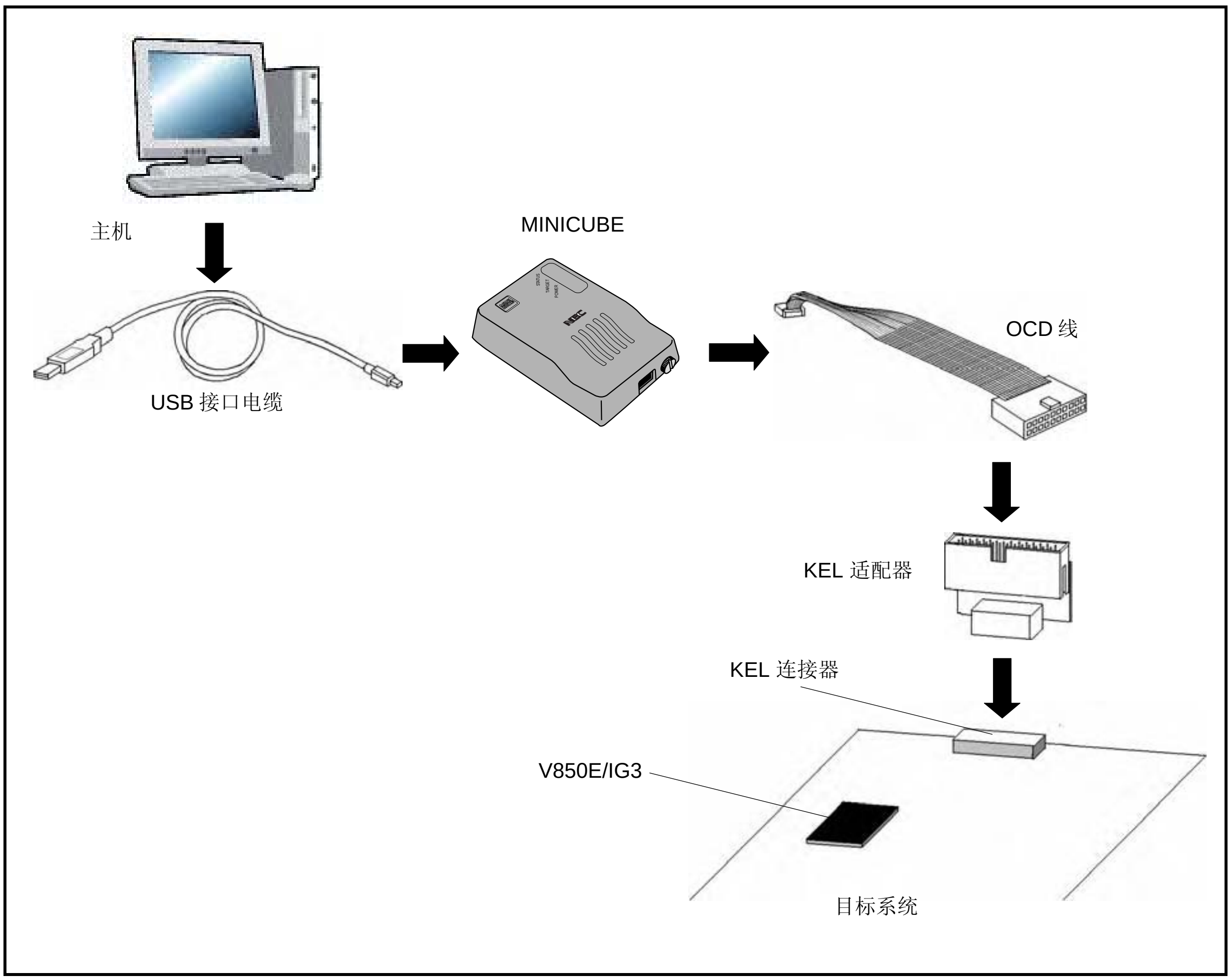
26.1.1 电路连接示例

使用 MINICUBE 时，推荐使用下列 KEL 连接件。

○ 零件登记号
- 8830E-026-170S：竖直类型
- 8830E-026-170L：直角类型

必需安装仿真器和电路，用于目标系统的连接。

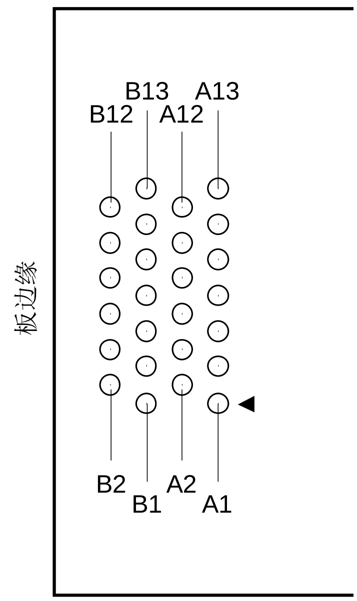
图 26-1. 片上调试接口的连接示例（MINICUBE）



(1) 引脚配置

下图是仿真器连接器的引脚配置(在目标系统侧)。

图 26-2. 仿真器连接器的引脚配置(在目标系统侧)



(俯视图)

注意事项　当连接器实际安装在板上时，基于连接器的尺寸来设置目标板。

(2) 引脚功能

下表是仿真器连接器的引脚功能(在目标系统侧)。

表 26-2. 仿真器连接器的引脚功能(在目标系统侧)

引脚编号	引脚名称	I/O	引脚功能
A1	(保留 1)	–	(连接至 GND)
A2	(保留 2)	–	(连接至 GND)
A3	(保留 3)	–	(连接至 GND)
A4	(保留 4)	–	(连接至 GND)
A5	(保留 5)	–	(连接至 GND)
A6	(保留 6)	–	(连接至 GND)
A7	DDI	输出	调试串行接口的数据输出
A8	DCK	输出	调试串行接口的时钟输出
A9	DMS	输出	调试串行接口的传输模式选择输出
A10	DDO	输入	调试串行接口的数据输入
A11	$\overline{\text{DRST}}$	输出	DCU 复位输出
A12	(保留 7)	–	(开路)
A13	FLMD0	输出	flash 存储器下载的控制信号
B1	GND	–	–
B2	GND	–	–
B3	GND	–	–
B4	GND	–	–
B5	GND	–	–
B6	GND	–	–
B7	GND	–	–
B8	GND	–	–
B9	GND	–	–
B10	GND	–	–
B11	PORT0_IN	–	(连接至 GND)
B12	PORT1_IN	–	(连接至 GND)
B13	V_{DD}	–	5 V 输入 (监视目标板的上电)

注意事项　1. 根据所使用的仿真器，V850E/IG3 不支持该引脚的连接。
2. 目标板的模式必须满足下列条件。

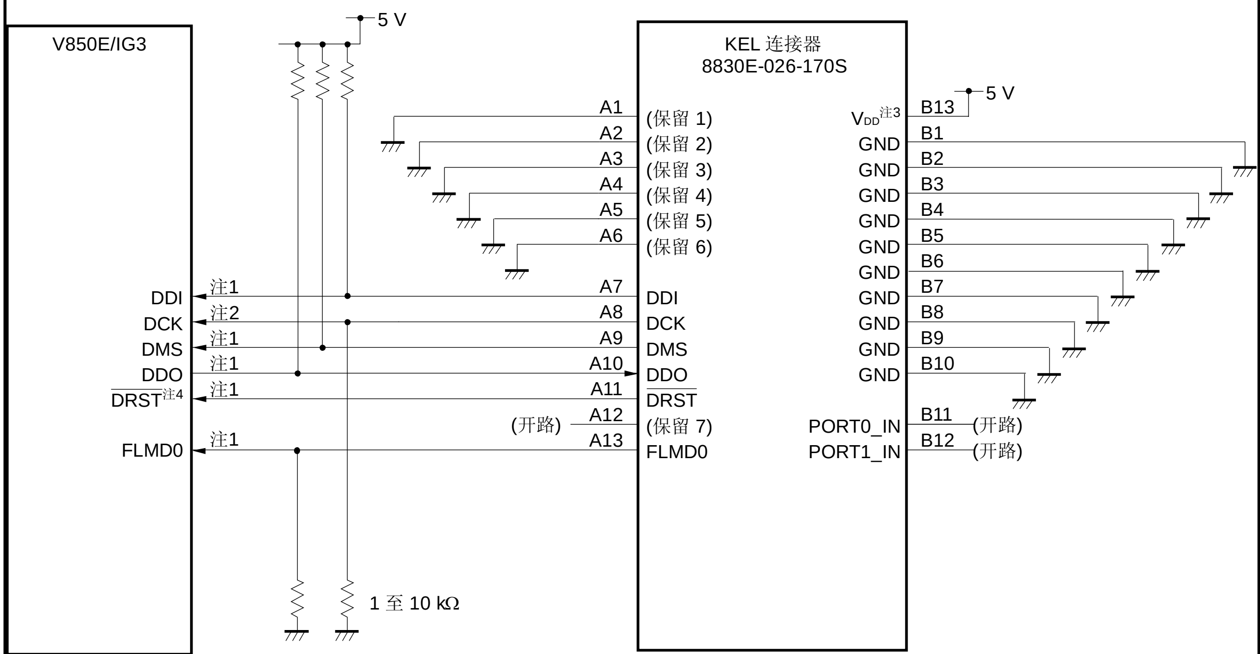
- 保持模式长度在 **100 mm** 内
- 用 **GND** 防护时钟信号

备注　输入/输出是从仿真器侧的角度。

(3) 推荐电路示例

下图是仿真器连接器的推荐电路示例(在目标系统侧)。

图 26-3. 仿真器的推荐连接示例



注 **1.** 保持模式长度在 100 mm 内

2. 用 GND 防护 DCK 信号

3. 检测对目标板的供电

4. 当 $\overline{\text{DRST}}$ 引脚为高电平时：片上调试模式

当 $\overline{\text{DRST}}$ 引脚为低电平时：正常操作模式

$\overline{\text{DRST}}$ 引脚在 V850E/IG3 中被内部下拉。

注意事项 DDO 信号是 5 V 输出，DDI, DCK, DMS 和 $\overline{\text{DRST}}$ 信号的输入电平是 TTL 电平。

26.1.2　接口信号

V850E/IG3 侧的接口信号的描述如下：

（1）$\overline{\text{DRST}}$

它是片上调试单元的复位输入信号。它是负逻辑信号，异步初始化调试单元（DCU）。

集成调试器启动后，当 MINICUBE 检测到目标系统的 V_{DD} 时，MINICUBE 将 $\overline{\text{DRST}}$ 信号的电平从低变为高，输出并启动 V850E/IG3 的片上调试单元。如果没有检测到目标系统的 V_{DD}，来自 MINICUBE 输出信号($\overline{\text{DRST}}$, DCK, DMS, DDI 和 FLMD0 引脚) 变为高阻抗状态。

集成调试器启动后，当 MINICUBE 检测到目标系统的 V_{DD} 时使得 $\overline{\text{DRST}}$ 上升，并且启动设备上的片上调试单元。

当 $\overline{\text{DRST}}$ 信号上升为高后，V850E/IG3 内产生一个复位信号。

在启动集成调试器开始调试后，总是会产生复位信号。

（2）DCK

它是时钟输入信号。由 MINICUBE 为其提供 20 MHz 的时钟。在片上调试单元，在 DCK 信号的上升沿对 DMS 和 DDI 信号进行采样，而在其下降沿输出数据 DDO。

（3）DMS

它是传输模式选择信号。调试单元的传输状态随 DMS 信号的电平改变。

（4）DDI

这是数据输入信号。在片上调试单元内在 DCK 的上升沿对其进行采样。

（5）DDO

这是数据输出信号。在 DCK 信号的下降沿由片上调试单元输出该信号。

（6）FLMD0

flash 存储器的自编程功能用于通过集成调试器把数据下载到 flash 存储器。在 flash 存储器自编程期间，FLMD0 引脚必须保持为高。此外，在 FLMD0 引脚接一个下拉电阻。

FLMD0 引脚可通过下面两种方式进行控制：

<1>　由 MINICUBE 控制：

将 MINICUBE 的 FLMD0 信号连接到 V850E/IG3 的 FLMD0 引脚。

在正常模式下，MINICUBE 不作任何驱动（高阻）。

在中断期间，当集成调试器执行下载功能时 MINICUBE 将 FLMD0 引脚上升为高电平。

<2>　由端口控制：

将设备的任一端口连接到 V850E/IG3 的 FLMD0 引脚。

可能会用到由用户程序实现 flash 存储器自编程功能的同一个端口。

在集成调试器的控制端，在执行下载功能之前，对端口引脚进行设置使端口引脚为高电平，或在执行下载功能之后，降低端口引脚的电平。

详情参见 **ID850QB Ver.3.20 集成调试器操作用户手册（U17964E）。**

26.1.3 可屏蔽功能

Reset、INTWDT和WAIT[注]信号都可以屏蔽。

调试器（ID850QB）的可屏蔽功能以及对应功能列举如下：

<R> 注　仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

表 **26-3.** 可屏蔽功能

调试器（ID850QB）的可屏蔽功能	对应 V850E/IG3 的功能
NMI0	产生不可屏蔽中断请求信号(INTWDT)
NMI1	×
NMI2	×
STOP	×
HOLD	×
RESET	RESET 引脚输入、看门狗定时器溢出产生的系统复位信号 (WDTRES)、低电压检测器(LVI)产生的系统复位信号 (LVIRES)、上电清零电路 (POC)产生的系统复位信号 (POCRES)
WAIT	WAIT 引脚[注]输入

<R> 注　仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

26.1.6 注意事项

（1）如果在 RUN（程序执行）期间输入复位信号（来自目标系统或来自内部复位源的复位信号），则断点功能可能会误操作。

（2）即使屏蔽复位信号屏蔽功能，如果在引脚输入复位信号也可能使得 I/O 缓存器（端口引脚）复位。

（3）在片上 flash 存储器设置的一个软件断点会因为复位信号而临时无效($\overline{\text{RESET}}$ 引脚输入、看门狗定时器溢出产生的系统复位信号 (WDTRES)、低电压检测器(LVI)产生的系统复位信号 (LVIRES)、上电清零电路 (POC)产生的系统复位信号 (POCRES))。当发生一个硬件断点或强制断点时，该断点再次有效，但是直到那时也不会发生软件断点。

（4）中断时屏蔽 reset 引脚，且 CPU 和外围 I/O 不复位。如果在执行用户程序时，flash 存储器由 DMA 重写或由 RAM 监控功能读取之后立刻产生引脚复位或内部复位，那么 CPU 和外围 I/O 可能无法正确复位。

（5） 在片上调试模式下，DDO 引脚强制设为高电平输出。

（6）在调试期间，设备中重写调试所用的 flash 存储器，所以 flash 存储器的重写次数无法保证。因此，不要在量产的产品中使用曾经用来调试的设备。

（7）因为 DDI 和 DCK 引脚功能可以复用为 CSIB0 I/O 引脚(SOB0，$\overline{\text{SCKB0}}$)、UARTA0 输出引脚(TXDA0)和外部中断引脚(INTP13)，所以当使用片上调试功能时，CSIB0、UARTA0 和 INTP13 无法使用。

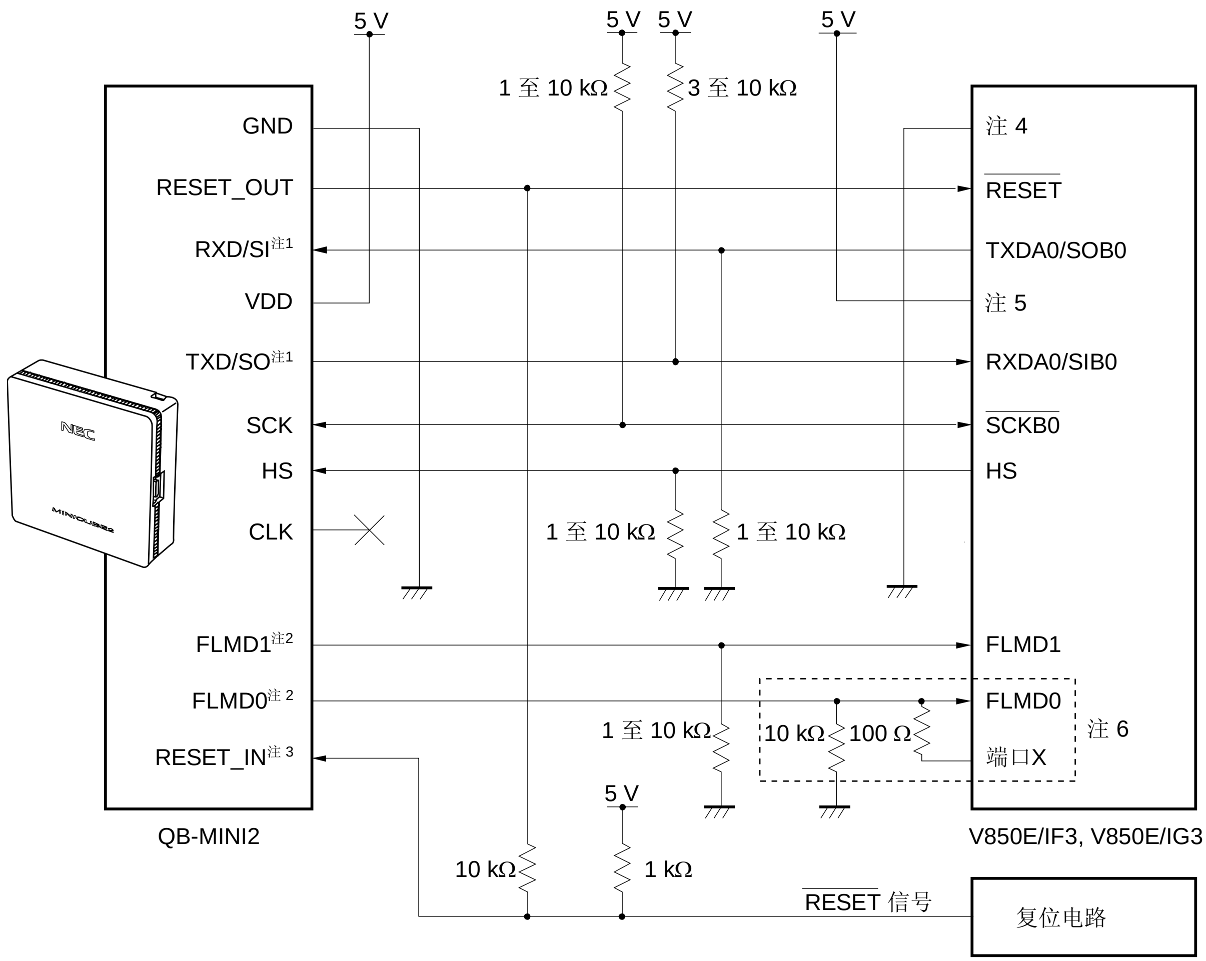
（8）使用片上调试功能时，即使设置为 STOP 模式，时钟发生器和 PLL 继续工作。

26.2　不使用 DCU 调试

下面描述如何使用 MINICUBE2 而不使用 DCU 来实现片上调试功能。通过 UARTA0 引脚（RXDA0 和 TXDA0）或 CSIB0 引脚（SIB0、SOB0、$\overline{\text{SCKB0}}$、HS（P43））作为调试接口。

26.2.1 电路连接示例

图 26-4. UARTA0/CSIB0 用作通信接口时的电路连接示例



注

1. 连接 V850E/IF3 和 V850E/IG3 的 TXDA0/SOB0（发送端）和目标连接器的 RXD/SI （接收端），还要将目标连接器的 TXD/SO（发送端）连接到 V850E/IF3 和 V850E/IG3 的 RXDA0/SIB0（接收端）。
2. 除了重写存储器在调试的中断时，连接到这个引脚（FLMD0，FLMD1）的 V850E/IF3 和 V850E/IG3 引脚可用作复用功能引脚，因为该引脚处于高阻状态。
3. 该连接设计是假设 $\overline{\text{RESET}}$ 信号由 N 通道漏极开路缓存器输出 （输出电阻：100Ω或更小）。
4. EV_{SS0}, EV_{SS1}, EV_{SS2} (仅限 V850E/IG3), V_{SS0}, V_{SS1}, AV_{SS0}, AV_{SS1}, AV_{SS2}
5. EV_{DD0}, EV_{DD1}, EV_{DD2} (仅限 V850E/IG3), V_{DD0}, V_{DD1}, AV_{DD0}, AV_{DD1}, AV_{DD2}
6. 虚线内的电路是为 flash 存储器自编程设计的，通过端口控制 FLMD0 引脚。使用该端口输入或输出高电平。当不执行 flash 存储器自编程时，接到 FLMD0 引脚的下拉电阻可以在 1 kΩ至 10 kΩ之间

备注　当 UARTA0 或 CSIB0 用作通信接口时，涉及的引脚参见表 **26-4**。

表 26-4. V850E/IF3、V850E/IG3 和 MINICUBE2 之间的连线（1/2）

MINICUBE2 (QB-MINI2)的引脚配置			当使用 UARTA0 时				
信号名称	I/O	引脚功能	引脚名称	引脚编号			
				IF3	IG3		
				GC	GC	GF	F1
SI/R$_x$D	输入	该引脚用于接收来自 V850E/IF3 和 V850E/IG3 的指令和数据	TXDA0	39	48	76	N12
SO/T$_x$D	输出	该引脚用于发送指令和数据至 V850E/IF3 和 V850E/IG3	RXDA0	38	47	75	P12
SCK	输出	3-线串行通信的时钟输出引脚	不需要	不需要	不需要	不需要	不需要
CLK	输出	到 V850E/IF3 和 V850E/IG3 的时钟输出引脚	不需要	不需要	不需要	不需要	不需要
			不需要	不需要	不需要	不需要	不需要
RESET_OUT	输出	到 V850E/IF3 和 V850E/IG3 的复位输出引脚	$\overline{\text{RESET}}$	35	40	68	P9
FLMD0	输出	该输出引脚用于将 V850E/IF3 和 V850E/IG3 设置为调试模式或编程模式	FLMD0	37	46	74	N9
FLMD1	输出	该输出引脚用于设置编程模式	FLMD1	60	76	4	A12
HS	输入	CSI0 + HS 通信的握手信号	不需要	不需要	不需要	不需要	不需要
GND	–	接地	V_{SS0}	32	37	65	N7, P7
			V_{SS1}	66	85	13	A9, B9
			AV_{SS0}	6	7	35	E1, E2
			AV_{SS1}	11	12	40	H1, H2
			AV_{SS2}	22	27	55	P1, N2, M3, L4
			EV_{SS0}	54	64	92	A14, G13, G14
			EV_{SS1}	79	1	29	A1, B1, C1, C2, D3, E4
			EV_{SS2}	不需要	42	70	N11, P11, P14
RESET_IN	输入	目标系统的复位输入引脚					

<R> **表 26-4. V850E/IF3、V850E/IG3 和 MINICUBE2 之间的连线（2/2）**

MINICUBE2 (QB-MINI2)的引脚配置			当使用 CSIB0-HS 时				
信号名称	I/O	引脚功能	引脚名称	引脚编号			
				IF3	IG3		
				GC	GC	GF	F1
SI/R_XD	输入	该引脚用于接收来自 V850E/IF3 和 V850E/IG3 的指令和数据	SOB0	39	48	76	N12
SO/T_XD	输出	该引脚用于发送指令和数据至 V850E/IF3 和 V850E/IG3	SIB0	38	47	75	P12
SCK	输出	3-线串行通信的时钟输出引脚	$\overline{\text{SCKB0}}$	40	49	77	M14
CLK	输出	到 V850E/IF3 和 V850E/IG3 的时钟输出引脚	不需要	不需要	不需要	不需要	不需要
			不需要	不需要	不需要	不需要	不需要
RESET_OUT	输出	到 V850E/IF3 和 V850E/IG3 的复位输出引脚	$\overline{\text{RESET}}$	35	40	68	P9
FLMD0	输出	该输出引脚用于将 V850E/IF3 和 V850E/IG3 设置为调试模式或编程模式	FLMD0	37	46	74	N9
FLMD1	输出	该输出引脚用于设置编程模式	FLMD1	60	76	4	A12
HS	输入	CSI0 + HS 通信的握手信号	P43	41	50	78	M13
GND	–	接地	V_{SS0}	32	37	65	N7, P7
			V_{SS1}	66	85	13	A9, B9
			AV_{SS0}	6	7	35	E1, E2
			AV_{SS1}	11	12	40	H1, H2
			AV_{SS2}	22	27	55	P1, N2, M3, L4
			EV_{SS0}	54	64	92	A14, G13, G14
			EV_{SS1}	79	1	29	A1, B1, C1, C2, D3, E4
			EV_{SS2}	不需要	42	70	N11, P11, P14
RESET_IN	输入	目标系统的复位输入引脚					

26.2.2 可屏蔽功能

只有复位信号可以屏蔽。

调试器（ID850QB）的可屏蔽功能以及对应功能列举如下：

表 26-5. 可屏蔽功能

调试器（ID850QB）可屏蔽的功能	对应 V850E/IF3 和 V850E/IG3 的功能
NMI0	×
NMI1	×
NMI2	×
STOP	×
HOLD	×
RESET	产生 $\overline{\text{RESET}}$ 引脚输入引发的复位信号
WAIT	$\overline{\text{WAIT}}$ 引脚[注]输入

<R> 注　仅限 μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

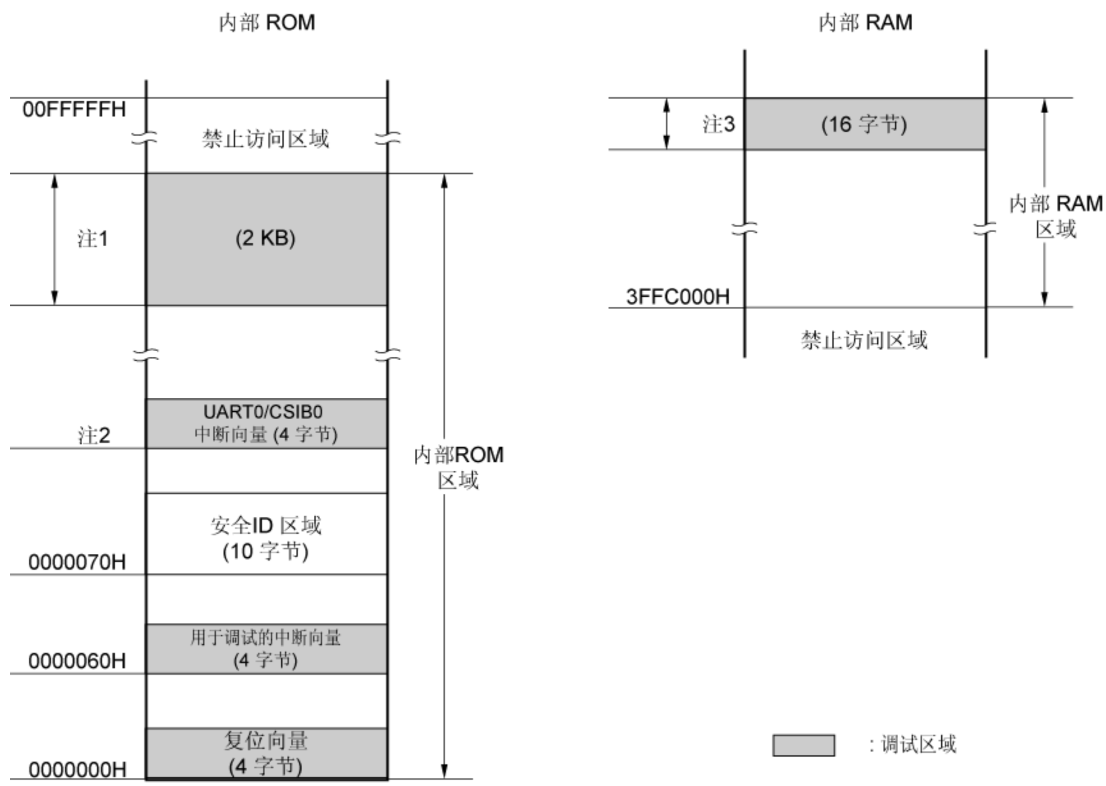
26.2.3 用户资源的保留

用户必须准备下列资源来实现 MINICUBE2 和 V850E/IF3、V850E/IG3 之间的通信功能，并完成各调试功能。这些内容需要在用户程序中设置或使用编译器选项设置。

（1）存储器空间的保留

图 26-5 中的阴影部分是预留区，用于放置调试监控程序，因此用户程序和数据不能分配在这些空间。必须保证这些空间不被用户程序使用。

图 **26-5.** 调试监控程序的内存空间分配



注　**1.** 地址值因产品不同而有所差异。

	内部 ROM 大小	调试区域
μPD70F3451 (V850E/IF3) μPD70F3453 (V850E/IG3)	128 KB	001F800H 至 001FFFFH
μPD70F3452 (V850E/IF3) μPD70F3454 (V850E/IG3)	256 KB	003F800H 至 003FFFFH

2. 使用 CSIB0 和 UARTA0 时的起始地址如下。

目标串行接口	中断名称	起始地址
UARTA0	INTUA0RE	000004B0H
	INTUA0R	000004C0H
	INTUA0T	000004D0H
CSIB0	INTCB0RE	000004E0H
	INTCB0R	000004F0H
	INTCB0T	00000500H

3. 地址值因产品不同而有所差异。

	内部 RAM 大小	调试区域
μPD70F3451 (V850E/IF3) μPD70F3453 (V850E/IG3)	8 KB	3FFDFF0H 至 3FFDFFFH
μPD70F3452 (V850E/IF3) μPD70F3454 (V850E/IG3)	12 KB	3FFEFF0H 至 3FFEFFFH

- 安全 ID 设置

ID 码必须嵌入到图 26-5 所示的 0000070H 至 0000079H 的区域，以防止存储内容被未授权读取。详情参见 **26.3 ROM** 安全功能。

（2）复位向量

复位向量包括调试监控程序所需的转移指令。

[如何保留区域]

没有必要刻意保护该区域。但是，当下载一个程序时，调试器根据下列情况重写复位向量。若重写的模式与下列情况不符，则调试器会产生错误（使用 ID850QB 时为 F0C34）。

（a）当从 0 地址开始连续放置两个空指令时

重写前		重写后
0x0 nop	→	跳转到 0x0 处的调试监控程序
0x2 nop		0x4 xxxx
0x4 xxxx		

（b）当从 0 地址开始连续放置两个 0xFFFF 时（已擦除器件）

重写前		重写后
0x0 0xFFFF	→	跳转到 0x0 处的调试监控程序
0x2 0xFFFF		0x4 xxxx
0x4 xxxx		

（c）*jr* 指令放在 0 地址（使用 CA850 时）

重写前		重写后
0x0 jr disp22	→	跳转到 0x0 处的调试监控程序
		0x4 jr disp22 - 4

（d）从 0 地址开始连续放置 *mov32* 和 *jmp*（使用 IAR 编译 ICCV850 时）

重写前		重写后
0x0 mov imm32,reg1	→	跳转到 0x0 处的调试监控程序
0x6 jmp [reg1]		0x4 mov imm32,reg1
		0xa jmp [reg1]

（e）在 0 地址处放置用于跳转到调试监控程序的指令

重写前		重写后
跳转到 0x0 处的调试监控程序	→	没有改变

（3）调试监控程序的保留区域

图 26-5 中的阴影部分是分配给调试监控程序的区域。监控程序对调试通信接口和 CPU 的 RUN 或中断处理过程执行初始化。内部 ROM 区必须填充为 0xFF。该区域禁止被用户程序重写。

[如何保留区域]

若用户程序不使用该区域，则没有必要进行保护。

但是，为了避免在调试器启动时可能出现的问题，建议预先使用编译器保留本区域。

下面给出保留区域的示例，使用 NEC 电子的编译器 CA850。加上汇编源文件和连接伪指令代码，如下所示。

• 汇编源文件（添加下列代码作为汇编源文件。）

```
-- Secures 2 KB space for monitor ROM section
.section "MonitorROM", const
.space     0x800, 0xff

-- Secures interrupt vector for debugging
.section "DBG0"
.space   4, 0xff

-- Secures interrupt vector for serial communication
-- Change the section name according to the serial communication mode used
.section "INTCB0RE"
.space   4, 0xff
.section "INTCB0R"
.space   4, 0xff
.section "INTCB0T"
.space   4, 0xff

-- Secures 16-byte space for monitor RAM section
.section "MonitorRAM", bss
.lcomm   monitorramsym, 16, 4;      -- defines symbol monitorramsym
```

• 连接伪指令（将下列代码添加到连接伪指令文件）

下面给出一个例子，它的内部 ROM 为 256 KB（结束地址为 003FFFFH）且内部 RAM 为 12 KB（结束地址为 3FFEFFFH）。

```
MROMSEG       : !LOAD ?R V0x03f800{
                MonitorROM      = $PROGBITS    ?A MonitorROM;
};

MRAMSEG       : !LOAD ?RW V0x03ffeff0{
                MonitorRAM      = $NOBITS      ?AW MonitorRAM;
};
```

（4）通信串行接口的保留

UARTA0 或 CSIB0 用于 MINICUBE2 和 V850E/IF3, V850E/IG3 之间的通信。与串行接口模式相关的设置通过调试监控程序来完成，但如果设置被用户程序改变，则可能发生通信错误。
为了防止发生这样的问题，必须在用户程序中对通信串行接口进行保留。

[如何保留通信串行接口]

- 串行接口寄存器
  在用户程序中不要设置 UARTA0 和 CSIB0 相关的寄存器。

- 中断屏蔽寄存器
  使用 UARTA0 时，不要屏蔽接收结束中断（INTUA0R）。使用 CSIB0 时，不要屏蔽接收结束中断（INTCB0R）。

（a） 使用 **UARTA0** 时

	7	6	5	4	3	2	1	0
UA0RIC	×	0	×	×	×	×	×	×

（b） 使用 **CSIB0** 时

	7	6	5	4	3	2	1	0
CB0RIC	×	0	×	×	×	×	×	×

备注　×：无须理会

- 使用 UARTA0 时的端口寄存器
  当 UARTA0 用于通信时，通过调试监控程序对端口寄存器进行设置，使 TXDA0 和 RXDA0 引脚有效。在调试期间用户程序不要改变下列寄存器的设置。（同样的值可覆盖写入）

	7	6	5	4	3	2	1	0
PFCE4	×	×	×	×	×	×	0	0
PFC4	×	×	×	×	×	×	1	1
PMC4	×	×	×	×	×	×	1	1

备注　×：无需理会

- 使用 CSIB0 的端口寄存器

  使用 CSIB0 时，通过调试监控程序对端口寄存器进行设置，使 SIB0、SOB0、$\overline{\text{SCKB0}}$ 和 HS（P43）引脚生效。在调试期间用户程序不要改变下列寄存器的设置。（同样的值可覆盖写入）

（a） SIB0、SOB0 和 $\overline{\text{SCKB0}}$ 设置

	7	6	5	4	3	2	1	0
PFCE4	×	×	×	×	×	0	0	0

	7	6	5	4	3	2	1	0
PFC4	×	×	×	×	×	0	0	0

	7	6	5	4	3	2	1	0
PMC4	×	×	×	×	×	1	1	1

（b） HS（P43 引脚）设置

	7	6	5	4	3	2	1	0
PM4	×	×	×	×	×	×	×	0

	7	6	5	4	3	2	1	0
P4	×	×	×	×	×	×	×	注

注　禁止写入该位。

HS 引脚对应的值会根据调试器的状态由监控程序改变。为了按照字节进行端口寄存器的设置，用户程序通常会使用读取－修改－写入方法。但是，如果在写入之前发生一个调试中断，就可能会执行一个意外的操作。

备注　×：无需理会

26.2.4 注意事项

（1）用于调试的设备处理

不要将用于调试的设备安装在批量生产的产品上，因为在调试过程中 flash 存储器被重写过，所以不能保证 flash 存储器的重写次数。此外，不要将调试监控程序嵌入到批量生产的产品中。

（2）当中断不能执行时

如果满足下列条件之一，则强制断点无法执行。

- 中断禁止（DI）。
- 发给串行接口发出的中断被屏蔽，该中断用于 MINICUBE2 和 V850E/IF3, V850E/IG3 之间的通信。
- 在通过可屏蔽中断释放待机时，禁止进入待机模式。
- MINICUBE2 和 V850E/IF3, V850E/IG3 之间的通信模式为 UARTA0，且外围时钟已停止。

（3）　当不执行伪实时 RAM 监控（RRM） 功能和 DMM 功能时

如果满足下列条件之一，则伪 RRM 功能和 DMM 功能无法操作。

- 中断禁止（DI）。
- 串行接口发出的中断被屏蔽，该中断用于 MINICUBE2 和 V850E/IF3, V850E/IG3 之间的通信。
- 禁止在通过可屏蔽中断解除待机时进入待机模式。
- MINICUBE2 和 V850E/IF3, V850E/IG3 之间的通信模式为 UARTA0，且外围时钟已停止。
- MINICUBE2 和 V850E/IF3, V850E/IG3 之间的通信模式为 UARTA0，并且用于通信的时钟和调试器中指定的时钟不同。

（4）允许由伪 RRM 和 DMM 功能解除待机模式。

如果满足下列条件之一，则可以由伪 RRM 功能和 DMM 功能来解除待机模式。

- MINICUBE2 和 V850E/IF3, V850E/IG3 之间的通信模式为 CSIB0。
- MINICUBE2 和 V850E/IF3, V850E/IG3 之间的通信模式为 UARTA0，且外围时钟不停止。

（5）使用 DMM 功能写入需要特定序列写入的外围 I/O 寄存器。

需要特定序列的外围 I/O 寄存器，不能用 DMM 功能写入。

（6） Flash 自编程

如果分配给调试监控程序的空间被 flash 存储器自编程重写，那么调试器将不能再正常运行。

26.3 ROM 安全功能

26.3.1 安全 ID

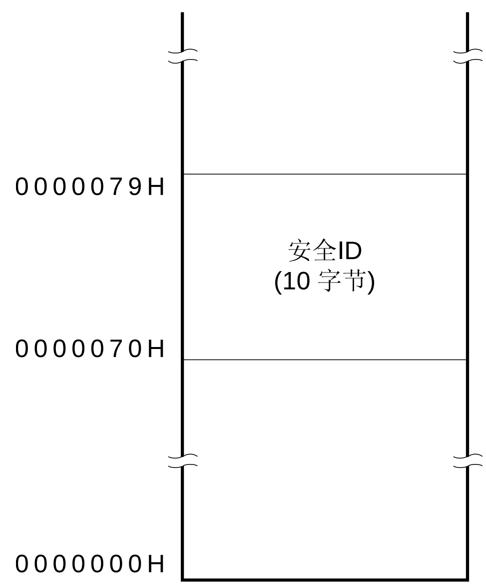
V850E/IF3 和 V850E/IG3 的 flash 存储器版本执行鉴定，通过使用一个 10 字节的 ID 码，防止 flash 存储器中的内容在由片上仿真器进行片上调试时被未经授权读取。

在 10 字节的片上 flash 存储区地址从 0000070H 至 0000079H 设置 ID 码，以便允许调试器执行 ID 鉴定。

如果 ID 匹配，则安全保护解除，且允许对 flash 存储器进行读取，也允许使用片上调试仿真器。

- 在 0000070H 至 0000079H 之间设置 10 字节的 ID 码。
- 0000079H 的第 7 位是片上调试仿真器的使能标志。
  （0：禁止，1：使能）
- 当启动片上调试仿真器后，调试器要求输入 ID。当输入的 ID 和 0000070H 至 0000079H 间设置的 ID 码匹配时，调试器启动。
- 即使 ID 码匹配，当片上调试仿真器的使能标志为 0 时，也不能进行调试.

图 26-6. 安全 ID 区



注意事项　flash 存储器擦除后，整个区域都写入 1。

26.3.2 设置

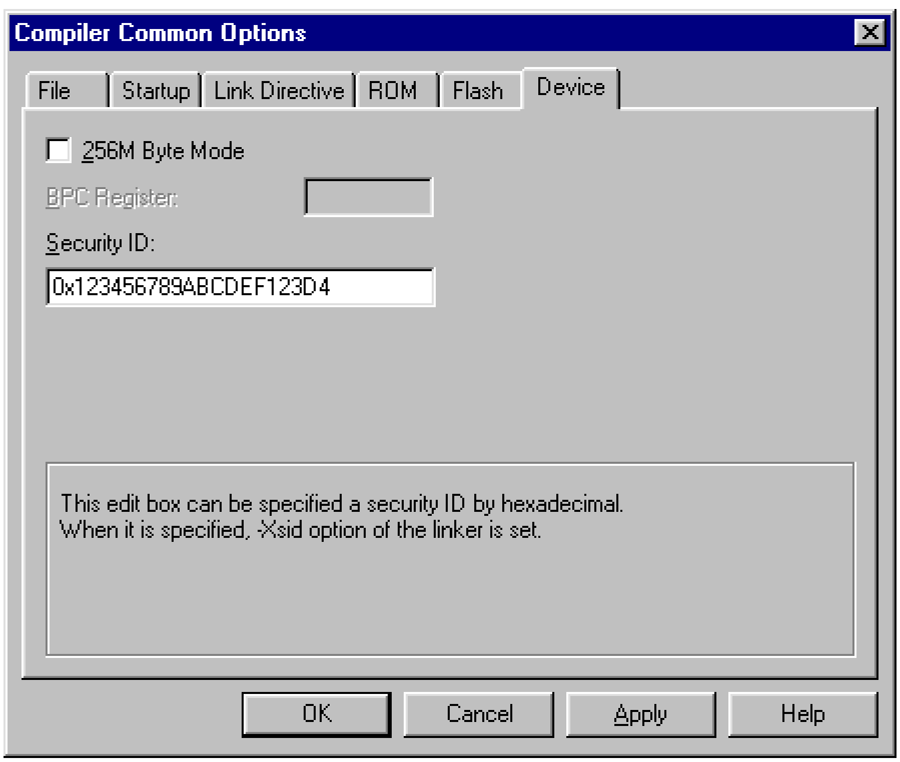
下面在表 26-6 中显示如何设置 ID 码。

当 ID 码按照表 26-6 进行设置时，在 ID850QB 的配置对话框中的 ID 码输入为 “123456789ABCDEF123D4”（ID 码不区分大小写）。

表 26-6. ID 码

地址	值
0x70	0x12
0x71	0x34
0x72	0x56
0x73	0x78
0x74	0x9A
0x75	0xBC
0x76	0xDE
0x77	0XF1
0x78	0x23
0x79	0xD4

<R> 如果使用的设备文件可以支持 CA850 Ver. 3.10 或更新版本以及安全 ID，可以使用 PM+内的编译器通用选项对话框指定 ID 码。



第二十七章　FLASH 存储器

27.1 特征

- 所有区域批量擦除或按块（2 KB）为单元擦除
- 与 flash 存储器编程器通过串行接口通信
- 擦除/写入电压：可以使用单电压电源进行擦除/写入
- 在板编程
- Flash 存储器可以自编程
- 使用引导交换功能通过自编程对整个 flash 存储器区域安全重写
- 重写方法
  - 通过 flash 存储器编程器从串行接口进行重写（在板/离线编程）
  - 通过用户程序重写 flash 存储器（自编程）
- 支持重写 flash 存储器功能并禁止读取功能（安全性增强）
- 自编程期间可以响应中断

表 27-1. 重写方式

重写方式	功能概述	工作模式
在板编程	设备安装到目标系统之后，flash 存储器可使用专用 flash 存储器编程器重写。	Flash 存储器编程模式
离线编程	在设备安装到目标系统之前，flash 存储器可通过专用 flash 存储器编程器和专用程序适配板（FA 系列）重写。	
自编程	Flash 存储器可通过执行用户程序来重写，用户程序通过离线/在板编程预先写入 flash 存储器。（在自编程时，不能从片上 flash 存储器区取指令，也不能对内部 flash 存储器区进行数据存取。因此，重写程序必须预先传送到内部 RAM 或外部存储器（仅限 μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A））。 <R>	正常操作模式

备注　FA 系列为 Naito Densei Machida Mfg. Co., Ltd 的产品。

表 27-2. 基本功能

功能	功能概述	支持（√：可支持，×：不支持）	
		在板/离线编程	自编程
块擦除	指定存储块的内容被擦除。	√	√
片擦除	一次全部擦除整个存储器的内容。	√	× （通过指定的区域支持块擦除）
写入	写入到指定地址，并校验检查写入的等级是否安全执行	√	√
校验/校验和	将从 flash 存储器读出的数据与 flash 存储器编程器传输的数据进行比较。	√	× （可以通过用户程序读取）
空白检测	检查整个存储器的擦除状态。	√	√
安全设置	禁止使用块擦除命令、片擦除命令、编程命令以及读取指令。	√	× （只有通过在板/离线编程设置的值可以保留）

表 27-3. 安全功能

功能	功能概述	支持	
		在板/离线编程	自编程
禁止块擦除命令	在所有块上禁止执行块擦除指令。关于禁止的设置可以通过执行片擦除指令来初始化。	详情参见 **27.1.2 安全功能**	
禁止片擦除命令	在所有块上禁止执行片擦除命令和块擦除命令。一旦设定为禁止之后，所有的关于禁止的设置都无法重新初始化设定， 因为片擦除命令不能执行。		
禁止编程命令	所有模块上禁止执行写入和块擦除命令。关于禁止的设置可以通过执行片擦除命令来初始化。		
禁止读取命令	所有块上禁止执行读取命令。关于禁止的设置可以通过执行片擦除命令来初始化。		

V850E/IF3 和 V850E/IG3 具有 128 KB 或 256 KB 的片上 flash 存储器。

- μPD70F3451（V850E/IF3），70F3453（V850E/IG3）：128 KB 片上 flash 存储器版本

（（PD70F3452（V850E/IF3），70F3454（V850E/IG3）：256 KB 片上 flash 存储器版本

Flash 存储器可以使用 Flash 存储器编程器重写，或使用自编程模式重写。
可以用连接到目标系统的 Flash 存储器编程器来执行写入 Flash 存储器。
以自编程模式写入可以通过应用程序来执行，而无须使用 flash 存储器编程器。
在下列开发环境下和量产的产品应用中经常使用 Flash 存储器版本。

○ 当 V850E/IF3 和 V850E/IG3 焊接到目标系统之后，要改动软件。
○ 根据不同模型小规模生产的规范，要改动软件。
○ 当开始量产时，要调整数据。

27.1.1 擦除单元

（1） 所有区域批量擦除

可以同时擦除 Flash 存储器区域 128 KB 或 256 KB。

（2） 按块单元擦除

可以按照块单元擦除。

- μPD70F3451（V850E/IF3），70F3453（V850E/IG3）：块 0 至块 63：每个 2 KB
- μPD70F3452（V850E/IF3），70F3454（V850E/IG3）：块 0 至块 127：每个 2 KB

27.1.2 安全功能

当重写 flash 存储器时，命令和功能可以得到保障。

作为 V850E/IF3 和 V850E/IG3 的出厂设置条件，选择了“全都使能”，且保证 flash 存储器内没有写入任何内容。

（1）在 flash 存储器编程模式下

安全设置（标志） 命令	全都使能	禁止读取	禁止写入	禁止片擦除	禁止块擦除	禁止重写引导块簇
读取	√	×	√	√	√	√
写入	√	√	×	√	√	△
片擦除	√	√	√	×	√	×
块擦除	√	√	×	×	×	△
安全设置的改变	√	**注 1**	**注 1**	**注 1**	**注 1**	**注 1, 2**
其他命令（比如空白检查与校验）	√	√	√	√	√	√

√： 命令可以接受

△： 该功能在引导块簇之外的区域可以执行

×： 保护错误

注 **1.** 可以禁止使能的设置。但是，如果禁止的设置使能时，会发生保护错误。要允许已经禁止的安全功能，必须擦除芯片。如果选择了禁止片擦除或禁止重写引导块簇，不能允许安全设置。

2. 如果设置了禁止重写引导块簇，引导块簇最后块编号（128 KB 版本：63，256 KB 版本：127）不能改变。

（2）在自编程模式下

安全设置（标志） 命令	全都使能	禁止读取	禁止写入	禁止片擦除	禁止块擦除	禁止重写引导块簇
FlashWordRead	√	√	√	√	√	√
FlashWordWrite	√	√	√	√	√	△
FlashBlockErase	√	√	√	√	√	△
FlashBootSwap	√	√	√	√	√	×
FlashSetInfo	√	**注 1**	**注 1**	**注 1**	**注 1**	**注 1, 2**
其他功能（比如 FlashBlockBlankCheck 和 FlashBlockVerify）	√	√	√	√	√	√

√： 命令可以接受

△： 该功能在引导块簇之外的区域可以执行

×： 保护错误

注 **1.** 可以禁止使能的设置。但是，如果禁止的设置使能时，会发生保护错误。要允许已经禁止的安全功能，必须在自编程模式下擦除芯片。如果选择了禁止片擦除，不能允许安全设置。

2. 如果设置了禁止重写引导块簇，引导块簇最后块编号（128 KB 版本：63，256 KB 版本：127）和引导交换簇设置标志不能改变。

27.2 使用专用 flash 存储器编程器写入

<R> 使用专用 flash 存储器编程器（PG-FP4，PG-FP5，FL-PR4 或 FL-PR5）和 MINICUBE2，可以执行在板或离线写入。

（1） 在板（On-board）编程

在 V850E/IF3 或 V850E/IG3 安装到目标系统之后，重写 flash 存储器的内容。在目标系统上安装连接器等，以连接 flash 存储器编程器。

（2） 离线（Off-board）编程

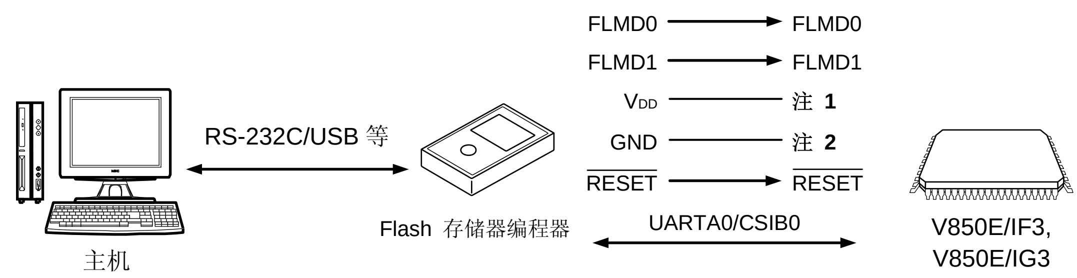
在 V850E/IF3 或 V850E/IG3 安装到目标系统之前，执行 flash 存储器的写入。

<R> 备注 FL-PR4 和 FL-PR5 是 Naito Densei Machida Mfg. Co., Ltd 的产品。

27.3 Flash 存储器编程环境

下面是将程序写入到 V850E/IF3 和 V850E/IG3 的 flash 存储器所需的环境。

<R>



注 **1.** V_{DD0}, V_{DD1}, EV_{DD0}, EV_{DD1}, EV_{DD2}（仅限 V850E/IG3）, AV_{DD0}, AV_{DD1}, AV_{DD2}, AV_{REFP0}, AV_{REFP1}

2. V_{SS0}, V_{SS1}, EV_{SS0}, EV_{SS1}, EV_{SS2}（仅限 V850E/IG3）, AV_{SS0}, AV_{SS1}, AV_{SS2}

需要一台主机用来控制专用 flash 存储器编程器。

UARTA0 或 CSIB0 用作专用 flash 存储器编程器和 V850E/IF3 或 V850E/IG3 之间的接口，执行写入、擦除等操作。通过 V850E/IF3 或 V850E/IG3 板上的振荡器与电容组成的振荡器来向 V850E/IF3 或 V850E/IG3 提供操作时钟。

表 27-4. 环境和通信模式

环境	通信模式		
	UARTA0	CSIB0	CSIB0 支持握手
<R> Flash 存储器编程器（PG-FP4, PG-FP5, FL-PR4 和 FL-PR5）	√	√	√
MINICUBE2	√	×	√

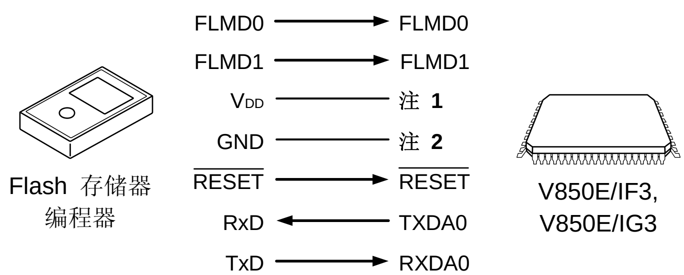
备注 √：支持，×：不支持

27.4 Flash 存储器编程的通信方式

（**1**） **UARTA0** 通信方式

传输速率： 9,600 至 153,600 bps（LSB 先行）

<R>



注 **1.** V_{DD0}, V_{DD1}, EV_{DD0}, EV_{DD1}, EV_{DD2}(仅限 V850E/IG3), AV_{DD0}, AV_{DD1}, AV_{DD2}, AV_{REFP0}, AV_{REFP1}

2. V_{SS0}, V_{SS1}, EV_{SS0}, EV_{SS1}, EV_{SS2}(仅限 V850E/IG3), AV_{SS0}, AV_{SS1}, AV_{SS2}

注意事项 **1.** 通过 **V850E/IF3** 或 **V850E/IG3** 板上的振荡器与电容组成的振荡器向 **V850E/IF3** 或 **V850E/IG3** 提供操作时钟。

2. 详情参见各个编程器的用户手册。

<R>

表 27-5. 专用 Flash 存储器编程器和 V850E/IF3, V850E/IG3 之间的对应连线

引脚编号	专用 Flash 存储器编程器（PG-FP4 或 PG-FP5）	I/O（PG-FP4 或 PG-FP5 侧）	V850E/IF3, V850E/IG3				
			引脚名称	引脚编号			
				V850E/IF3	V850E/IG3		
				GC	GC	GF	F1
1	GND	–	V_{SS0}	32	37	65	N7, P7
			V_{SS1}	66	85	13	A9, B9
			EV_{SS0}	54	64	92	A14, G13, G14
			EV_{SS1}	79	1	29	**注 1**
			EV_{SS2}	–	42	70	N11, P11, P14
			AV_{SS0}	6	7	35	E1, E2
			AV_{SS1}	11	12	40	H1, H2
			AV_{SS2}	22	27	55	L4, M3, N2, P1
2	$\overline{\text{RESET}}$	输出	$\overline{\text{RESET}}$	35	40	68	P9
3	SI/RxD	输入	TXDA0	39	48	76	N12
4	V_{DD}	–	V_{DD0}	30	35	63	N5, P5
			V_{DD1}	68	87	15	A7, B7
			EV_{DD0}	55	65	93	F13, F14
			EV_{DD1}	80	100	28	A3, B2
			EV_{DD2}	–	41	69	N10, P10
			AV_{DD0}	8	9	37	F1
			AV_{DD1}	9	10	38	G1
			AV_{DD2}	21	26	54	M1, M2
			AV_{REFP0}	7	8	36	F2
			AV_{REFP1}	10	11	39	G2
5	SO/TxD	输出	RXDA0	38	47	75	P12
6	V_{PP}	×	NC	–	–	–	–
7	SCK	×	NC	–	–	–	–
8	H/S	×	NC	–	–	–	–
9	CLK[注 2]	输出	X1[注 2]	33	38	66	P8
10	VDE	×	NC	–	–	–	–
11	V_{DD2}	–	NC	–	–	–	–
12	FLMD1	输出	**注 3**	60	76	4	A12
13	RFU-1	×	NC	–	–	–	–
14	FLMD0	输出	FLMD0	37	46	74	N9
15	未使用	×	NC	–	–	–	–
16	未使用	×	NC	–	–	–	–

注 **1.** A1, B1, C1, C2, D3 和 E4

2. 在 V850E/IF3 和 V850E/IG3 内，禁止外部时钟输入。在板上安装振荡器。

3. 通过一个电阻连接到 FLMD1 或 GND。

备注 NC：无连接

GC（V850E/IF3）：80-脚塑封 LQFP（14 × 14）

GC（V850E/IG3）：100-脚塑封 LQFP(密间距)(14 × 14)

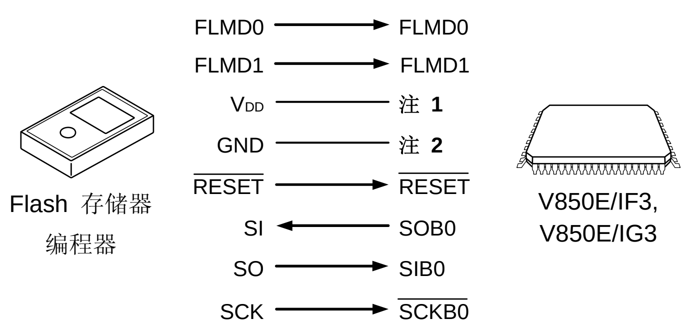
GF（V850E/IG3）：100-脚塑封 LQFP（14 × 20）

F1（V850E/IG3）：161-脚塑封 FBGA（10 × 10）

（2） CSIB0 通信方式

串行时钟：最高 2.5 MHz（MSB 先行）

<R>



注 **1.** V_{DD0}, V_{DD1}, EV_{DD0}, EV_{DD1}, EV_{DD2}（仅限 V850E/IG3）, AV_{DD0}, AV_{DD1}, AV_{DD2}, AV_{REFP0}, AV_{REFP1}

2. V_{SS0}, V_{SS1}, EV_{SS0}, EV_{SS1}, EV_{SS2}（仅限 V850E/IG3）, AV_{SS0}, AV_{SS1}, AV_{SS2}

注意事项 **1.** 通过 **V850E/IF3** 或 **V850E/IG3** 板上的振荡器与电容组成的振荡器向 **V850E/IF3** 或 **V850E/IG3** 提供操作时钟。

2. 详情参见各个编程器的用户手册。

Flash 存储器编程器输出（主）传输时钟，V850E/IF3 或 V850E/IG3 作为从设备工作。

<R>

表 27-6. 专用 Flash 存储器编程器和 V850E/IF3, V850E/IG3 之间的对应连线

引脚编号	Flash 存储器编程器（PG-FP4 或 PG-FP5）	I/O（PG-FP4 或 PG-FP5 侧）	V850E/IF3, V850E/IG3				
			引脚名称	引脚编号			
				V850E/IF3	V850E/IG3		
				GC	GC	GF	F1
1	GND	–	V_{SS0}	32	37	65	N7, P7
			V_{SS1}	66	85	13	A9, B9
			EV_{SS0}	54	64	92	A14, G13, G14
			EV_{SS1}	79	1	29	注 **1**
			EV_{SS2}	–	42	70	N11, P11, P14
			AV_{SS0}	6	7	35	E1, E2
			AV_{SS1}	11	12	40	H1, H2
			AV_{SS2}	22	27	55	L4, M3, N2, P1
2	$\overline{\text{RESET}}$	输出	$\overline{\text{RESET}}$	35	40	68	P9
3	SI/RxD	输入	SOB0	39	48	76	N12
4	V_{DD}	–	V_{DD0}	30	35	63	N5, P5
			V_{DD1}	68	87	15	A7, B7
			EV_{DD0}	55	65	93	F13, F14
			EV_{DD1}	80	100	28	A3, B2
			EV_{DD2}	–	41	69	N10, P10
			AV_{DD0}	8	9	37	F1
			AV_{DD1}	9	10	38	G1
			AV_{DD2}	21	26	54	M1, M2
			AV_{REFP0}	7	8	36	F2
			AV_{REFP1}	10	11	39	G2
5	SO/TxD	输出	SIB0	38	47	75	P12
6	V_{PP}	×	NC	–	–	–	–
7	SCK	输出	$\overline{\text{SCKB0}}$	40	49	77	M14
8	H/S	×	NC	–	–	–	–
9	CLKZ[注 2]	输出	X1[注 2]	33	38	66	P8
10	VDE	×	NC	–	–	–	–
11	V_{DD2}	–	NC	–	–	–	–
12	FLMD1	输出	注 **3**	60	76	4	A12
13	RFU-1	×	NC	–	–	–	–
14	FLMD0	输出	FLMD0	37	46	74	N9
15	未使用	×	NC	–	–	–	–
16	未使用	×	NC	–	–	–	–

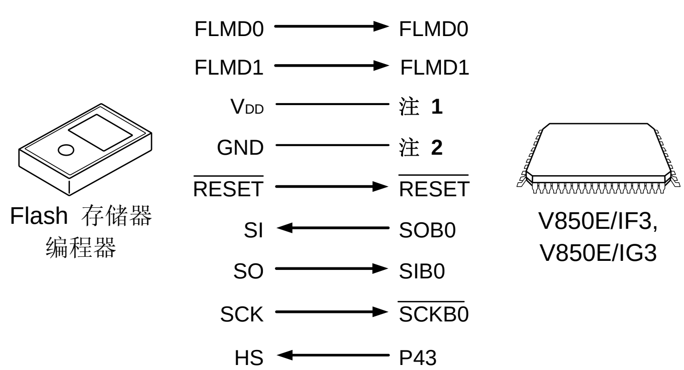
注 **1.** A1, B1, C1, C2, D3 和 E4
2. 在 V850E/IF3 和 V850E/IG3 内，禁止外部时钟输入。在板上安装振荡器。
3. 通过一个电阻连接到 FLMD1 或 GND。

备注 NC：无连接
GC（V850E/IF3）：80-脚塑封 LQFP（14 × 14）
GC（V850E/IG3）：100-脚塑封 LQFP（密间距）（14 × 14）
GF（V850E/IG3）：100-脚塑封 LQFP（14 × 20）
F1（V850E/IG3）：161-脚塑封 FBGA（10 × 10）

（**3**） **CSIB0** 通信方式支持握手

串行时钟：最高 2.5 MHz（MSB 先行）

<R>



注 **1.** V_{DD0}, V_{DD1}, EV_{DD0}, EV_{DD1}, EV_{DD2}（仅限 V850E/IG3）, AV_{DD0}, AV_{DD1}, AV_{DD2}, AV_{REFP0}, AV_{REFP1}

2. V_{SS0}, V_{SS1}, EV_{SS0}, EV_{SS1}, EV_{SS2}（仅限 V850E/IG3）, AV_{SS0}, AV_{SS1}, AV_{SS2}

注意事项 1. 通过 V850E/IF3 或 V850E/IG3 板上的振荡器与电容组成的振荡器向 V850E/IF3 或 V850E/IG3 提供操作时钟。

2. 详情参见各个编程器的用户手册。

<R> Flash 存储器编程器输出传输时钟，V850E/IF3 或 V850E/IG3 作为从设备操作。

<R> 使用 PG-FP4 或 PG-FP5 时，它向 V850E/IF3 或 V850E/IG3 发送下列信号。详情参见 **PG-FP4** 用户手册（**U15260E**）或 **PG-FP5** 用户手册（**U18865E**）。

<R>

表 27-7. 专用 Flash 存储器编程器和 V850E/IF3, V850E/IG3 之间的对应连线

引脚编号	Flash 存储器编程器（PG-FP4 或 PG-FP5）	I/O（PG-FP4 或 PG-FP5 侧）	V850E/IF3, V850E/IG3				
			引脚名称	引脚编号			
				V850E/IF3	V850E/IG3		
				GC	GC	GC	F1
1	GND	–	V_{SS0}	32	37	65	N7, P7
			V_{SS1}	66	85	13	A9, B9
			EV_{SS0}	54	64	92	A14, G13, G14
			EV_{SS1}	79	1	29	注 **1**
			EV_{SS2}	–	42	70	N11, P11, P14
			AV_{SS0}	6	7	35	E1, E2
			AV_{SS1}	11	12	40	H1, H2
			AV_{SS2}	22	27	55	L4, M3, N2, P1
2	$\overline{\text{RESET}}$	输出	$\overline{\text{RESET}}$	35	40	68	P9
3	SI/RxD	输入	SOB0	39	48	76	N12
4	V_{DD}	–	V_{DD0}	30	35	63	N5, P5
			V_{DD1}	68	87	15	A7, B7
			EV_{DD0}	55	65	93	F13, F14
			EV_{DD1}	80	100	28	A3, B2
			EV_{DD2}	–	41	69	N10, P10
			AV_{DD0}	8	9	37	F1
			AV_{DD1}	9	10	38	G1
			AV_{DD2}	21	26	54	M1, M2
			AV_{REFP0}	7	8	36	F2
			AV_{REFP1}	10	11	39	G2
5	SO/TxD	输出	SIB0	38	47	75	P12
6	V_{PP}	×	NC	–	–	–	–
7	SCK	输出	$\overline{\text{SCKB0}}$	40	49	77	M14
8	H/S	输入	P43	41	50	78	M13
9	CLK[注 2]	输出	X1[注 2]	33	38	66	P8
10	VDE	×	NC	–	–	–	–
11	V_{DD2}	–	NC	–	–	–	–
12	FLMD1	输出	注 **3**	60	76	4	A12
13	RFU-1	×	NC	–	–	–	–
14	FLMD0	输出	FLMD0	37	46	74	N9
15	未使用	×	NC	–	–	–	–
16	未使用	×	NC	–	–	–	–

注 **1.** A1, B1, C1, C2, D3 和 E4
2. 在 V850E/IF3 和 V850E/IG3 内，禁止外部时钟输入。在板上安装振荡器。
3. 通过一个电阻连接到 FLMD1 或 GND。

备注 NC：无连接
GC（V850E/IF3）：80-脚塑封 LQFP（14 × 14）
GC（V850E/IG3）：100-脚塑封 LQFP（密间距）（14 × 14）
GF（V850E/IG3）：100-脚塑封 LQFP（14 × 20）
F1（V850E/IG3）：161-脚塑封 FBGA（10 × 10）

27.5 Flash 存储器编程期间的引脚处理

当执行在板写入时，在目标系统上安装一个连接器用以连接专用 flash 存储器编程器。

在 flash 存储器编程模式下，所有在 flash 存储器编程过程中未使用的引脚，其状态与正常操作模式复位后的瞬时状态相同。因为所有的端口变为高阻抗状态，所以，当外部设备不能响应高阻抗状态时，需要对引脚进行处理。

27.5.1 供电电源

在正常操作模式下，向（V_{DD0}, V_{DD1}, V_{SS0}, V_{SS1}, EV_{DD0}, EV_{DD1}, EV_{DD2}(仅限 V850E/IG3), EV_{SS0}, EV_{SS1}, EV_{SS2}(仅限 V850E/IG3), AV_{DD0}, AV_{DD1}, AV_{DD2}, AV_{SS0}, AV_{SS1}, AV_{SS2}, AV_{REFP0} 和 AV_{REFP1}）提供相同的供电电源。将 V_{DD} 和 flash 存储器编程器的 GND 连接到 V_{DD0}, V_{DD1}, V_{SS0}, V_{SS1}, EV_{DD0}, EV_{DD1}, EV_{DD2}(仅限 V850E/IG3), EV_{SS0}, EV_{SS1}, EV_{SS2}(仅限 V850E/IG3), AV_{DD0}, AV_{DD1}, AV_{DD2}, AV_{SS0}, AV_{SS1}, AV_{SS2}, AV_{REFP0} 和 AV_{REFP1}。（flash 存储器编程器的 V_{DD} 提供供电电源监视功能。）

在 flash 存储器编程模式下（包括 flash 存储器自编程），在 V_{DD0}, V_{DD1} 引脚和 V_{SS0}, V_{SS1} 引脚之间插入电容，在 EV_{DD0}, EV_{DD1}, EV_{DD2}（仅限 V850E/IG3）引脚和 EV_{SS0}, EV_{SS1}, EV_{SS2}（仅限 V850E/IG3）引脚之间插入电容，用以稳定供电电源的电压。

27.5.2 使用的引脚

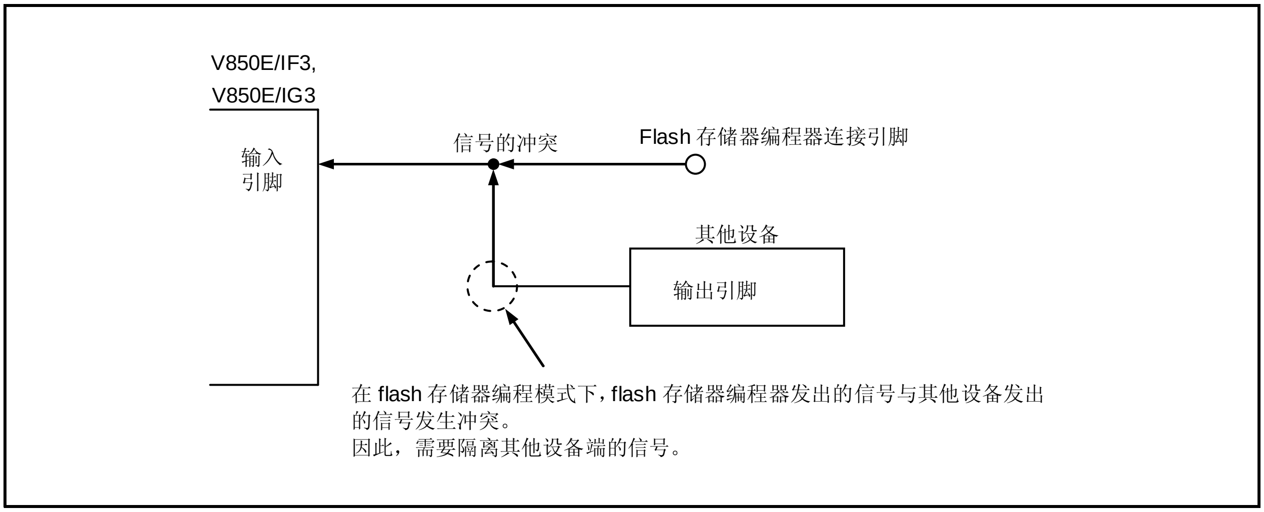
下面展示各个接口所使用的引脚。

通信模式	使用的引脚
UARTA0	TXD0, RXD0
CSIB0	SOB0, SIB0, $\overline{\text{SCKB0}}$
CSIB0 支持握手	SOB0, SIB0, $\overline{\text{SCKB0}}$, P43

将 flash 存储器编程器接到某个串行接口引脚，该接口已经与板上其它设备连接时，应注意避免信号冲突或其它设备的误操作。

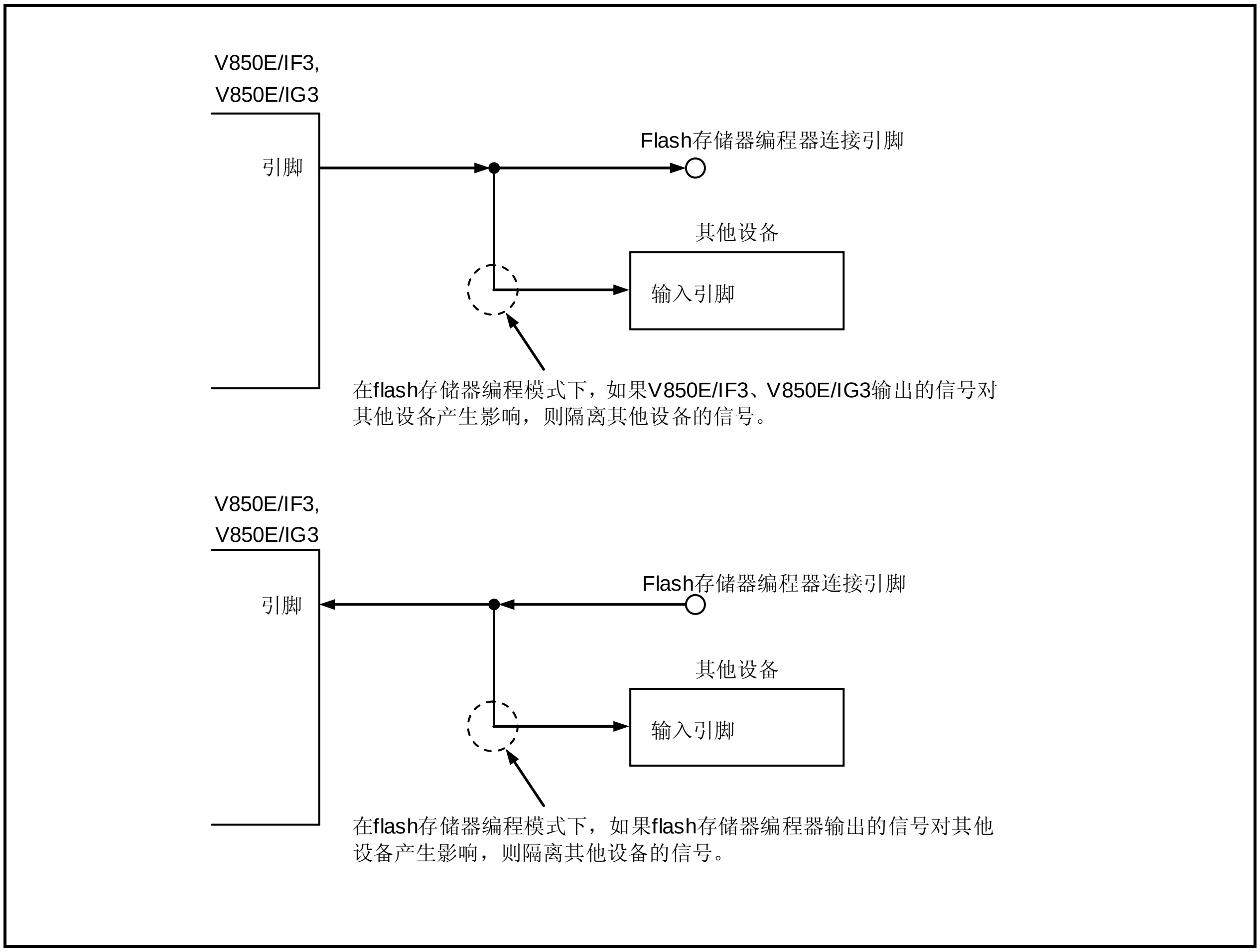
（1） 信号冲突

当专用 flash 存储器编程器（输出）连接到某个串行接口引脚（输入）时，该接口已经与板上其它设备（输出）连接，则会发生信号冲突。为避免信号冲突，隔离其与其它设备的连接，或将其它设备设为输出高阻抗状态。



（2） 其它设备的误操作

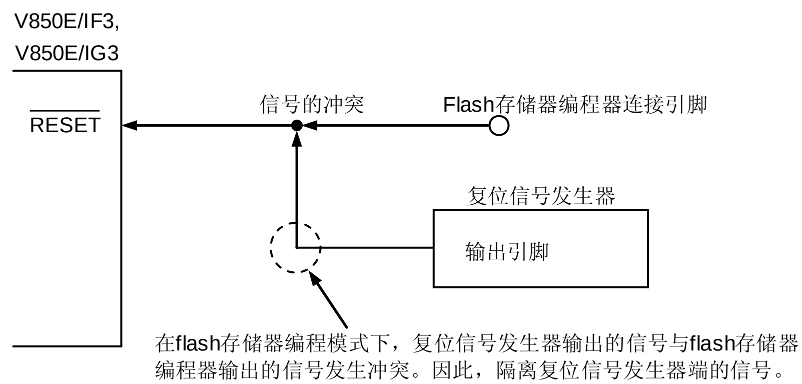
当专用 flash 存储器编程器（输出或输入）接到到某个串行接口引脚（输入或输出）时，该接口已经与板上另一个设备（输入）连接，则信号被输出到其它设备，导致设备的误操作。为避免这种情况。隔离与其它设备的连接，或对其设置使得输入到其他设备的信号被忽略。



27.5.3 $\overline{\text{RESET}}$ 引脚

当 flash 存储器编程器的复位信号接到 $\overline{\text{RESET}}$ 引脚时，该引脚已经连接到板上的复位信号发生器，则会发生信号冲突。为避免信号间的冲突，隔离其与复位信号发生器的连接。

在 Flash 存储器编程模式下，当复位信号由用户系统输入时，编程操作将不能正确执行。因此，不要输入任何 flash 存储器编程器复位信号之外的信号。

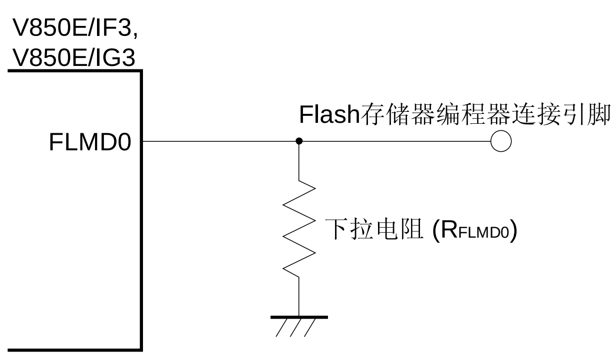


27.5.4 FLMD0 和 FLMD1 引脚

（1） FLMD0 引脚

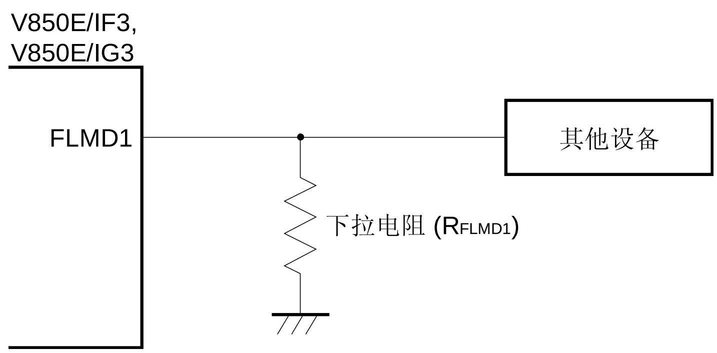
在正常操作模式下，向 FLMD0 引脚输入一个的 EV_{SS0}, EV_{SS1} 或 EV_{SS2}（V850E/IG3）电平的电压。在 flash 存储器编程模式，向 FLMD0 引脚提供 EV_{DD0}, EV_{DD1}, EV_{DD2}（仅限 V850E/IG3）电平的写入电压。

因为在自编程模式时，FLMD0 引脚被用作写入保护引脚，因此在写入 flash 存储器之前，必须通过端口控制等手段向 FLMD0 引脚提供 EV_{DD0}, EV_{DD1} 或 EV_{DD2}（仅限 V850E/IG3）电平的电压。详情参见 **27.7.7（1） FLMD0 引脚**。



（2） FLMD1 引脚

当 FLMD0 引脚输入 0V 时，FLMD1 引脚不起作用。当向 FLMD0 引脚提供 EV_{DD0}, EV_{DD1} 或 EV_{DD2}（仅限 V850E/IG3）时，进入 flash 存储器编程模式，因此必须向 FLMD1 引脚输入 0V。下面是 FLMD1 引脚的连接示例。



注意事项　在板编程期间，或复位刚结束时，如果从其它器件向 FLMD1 引脚输入 EV_{DD0}, EV_{DD1} 或 EV_{DD2}（仅限 V850E/IG3）信号，应将该信号隔离。

表 27-8. 复位解除后 FLMD0 和 FLMD1 引脚与操作模式之间的关系

FLMD0	FLMD1	操作模式
0	无需理会	正常操作模式
EV_{DD}	0	Flash 存储器编程模式
EV_{DD}	EV_{DD}	禁止设置

备注　EV_{DD}：EV_{DD0}, EV_{DD1}, EV_{DD2}（仅限 V850E/IG3）

27.5.5 端口引脚

当设置为 flash 存储器编程模式时，除了和 flash 存储器编程器通信的引脚之外，所有端口引脚变为高阻抗状态。这些端口引脚无需处理。如果外部设备连接到该端口时发生错误，比如高阻抗状态的禁止，通过电阻将他们连接到 EV_{DD0}, EV_{DD1}, EV_{DD2}（仅限 V850E/IG3）或 EV_{SS0}, EV_{SS1}, EV_{SS2}（仅限 V850E/IG3）。

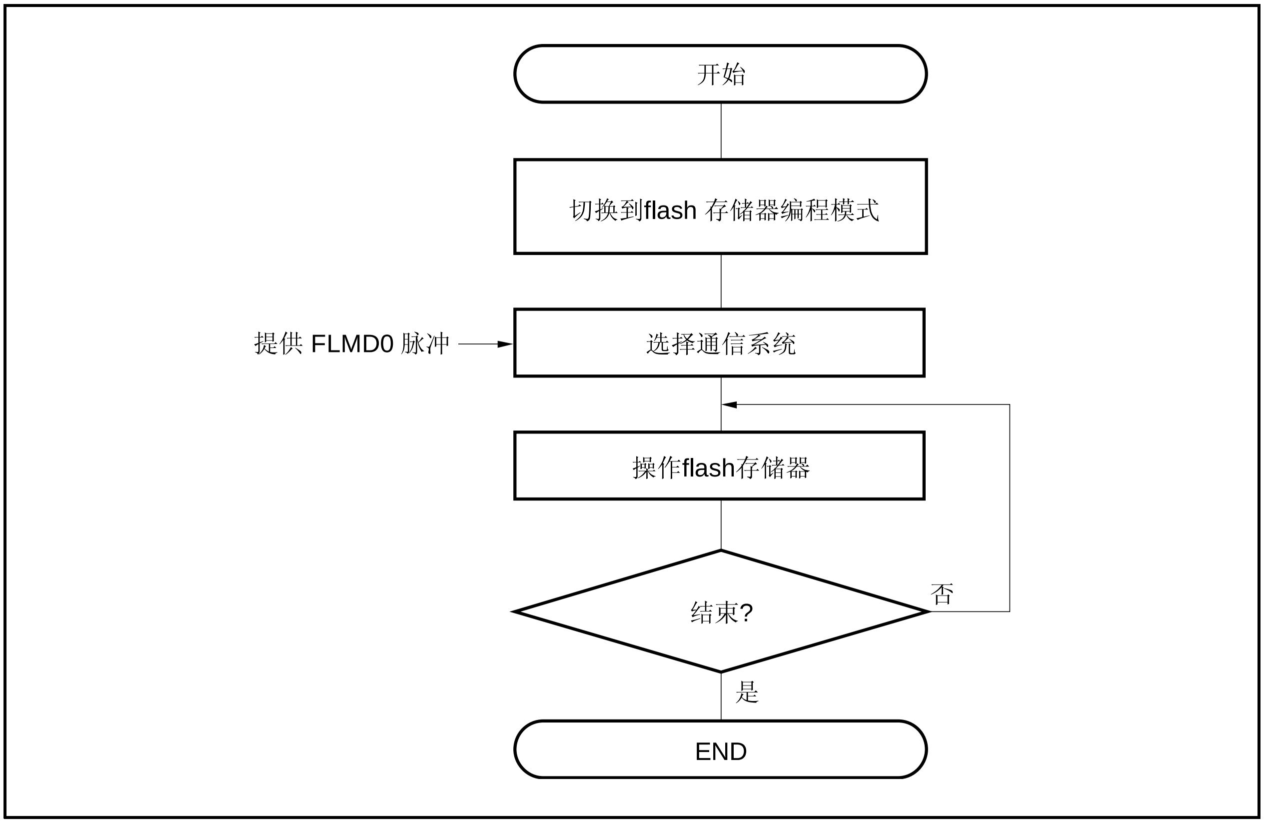
27.5.6 其它信号引脚

X1 和 X2 的连接状态，与正常操作模式相同。

27.6 Flash 存储器编程模式

27.6.1 Flash 存储器控制

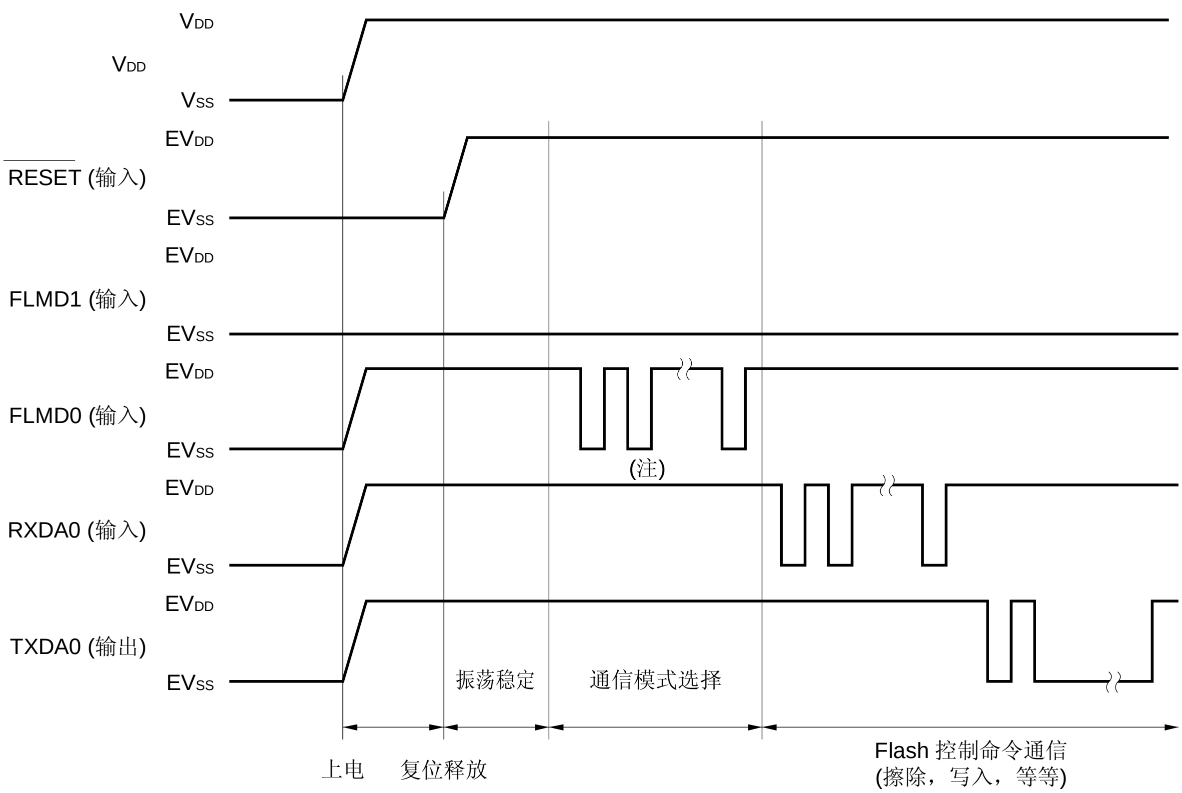
以下显示了操作 flash 存储器的过程。



27.6.2 通信模式的选择

在 V850E/IF3 和 V850E/IG3 中，切换为 flash 存储器编程模式后，通过向 FLMD0 引脚输入脉冲（最多 11 个脉冲）来选择通信模式。FLMD0 脉冲由 flash 存储器编程器产生。

下面显示了脉冲数量和通信模式之间的关系。



注　　时钟数目取决于通信模式，关系如下：

FLMD0 脉冲	通信模式	备注
0	UARTA0	通信速度：9,600 bps（复位后），LSB 先行
8	CSIB0	V850E/IF3 和 V850E/IG3 执行从机操作，MSB 先行
11	CSIB0 + HS	V850E/IF3 和 V850E/IG3 执行从机操作，MSB 先行
其它	RFU	禁止设置

注意事项　选择 UARTA0 时，在接收到 FLMD0 脉冲以后，基于 flash 存储器编程器发出的复位命令来计算接收时钟。

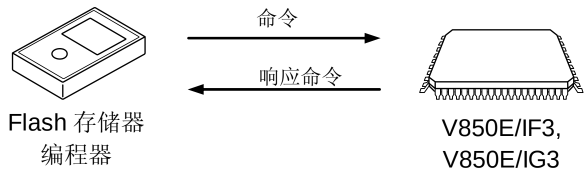
备注　V_{DD}：V_{DD0}, V_{DD1}

EV_{DD}：EV_{DD0}, EV_{DD1}, EV_{DD2}（仅限 V850E/IG3）

EV_{SS}：EV_{SS0}, EV_{SS1}, EV_{SS2}（仅限 V850E/IG3）

27.6.3 通信命令

V850E/IF3 和 V850E/IG3 通过指令与 flash 存储器编程器进行通信。从 flash 存储器编程器发至 V850E/IF3 和 V850E/IG3 的信号称为“命令”。从 V850E/IF3 和 V850E/IG3 发出至 flash 专用编程器的应答信号称为“应答命令”。



下面显示了 V850E/IF3 和 V850E/IG3 内用于控制 flash 存储器的命令。所有的这些命令都由 flash 存储器编程器发出，且 V850E/IF3 和 V850E/IG3 执行命令相应的处理。

类别	命令名称	支持			功能
		UARTA0	CSIB0	注	
校验	块校验命令	√	√	√	比较指定块的内容和输入数据
擦除	片擦除命令	√	√	√	擦除整个 flash 存储器的内容
	块擦除命令	√	√	√	擦除指定块的内容
空白检查	块空白检查命令	√	√	√	检查指定块的擦除状态。
数据写入	写入命令	√	√	√	向指定块写入数据
数据读取	读取命令	√	√	√	读出指定块的数据
系统设置和控制	状态命令	√	×	√	获取操作的状态
	振荡频率设置命令	√	√	√	设置振荡频率
	波特率设置命令	√	×	×	当选择 UART0 时，改变波特率
	硅签字命令	√	√	√	读出硅签字信息
	版本获取命令	√	√	√	读出设备版本和固件版本
	安全设置命令	√	√	√	设置安全信息和引导块大小
	校验和命令	√	√	√	发送读取块区域数据的校验和的值
	复位命令	√	√	√	用于检测通信同步

注　　CSIB0 支持握手

对于 flash 存储器编程器发出的命令，V850E/IF3 和 V850E/IG3 发回应答命令。从 V850E/IF3 和 V850E/IG3 发回的应答命令列举如下。

应答命令名称	功能
ACK（响应）	响应命令/数据，等
NAK（无响应）	响应非法帧，等
命令编号错误	响应非法命令/数据，等
参数错误	响应非法参数，等
校验和错误	响应帧的校验和
保护错误	当保护生效时，响应
在处理期间（BUSY）	在处理期间响应
其它情况	错误

27.7 Flash 存储器自编程

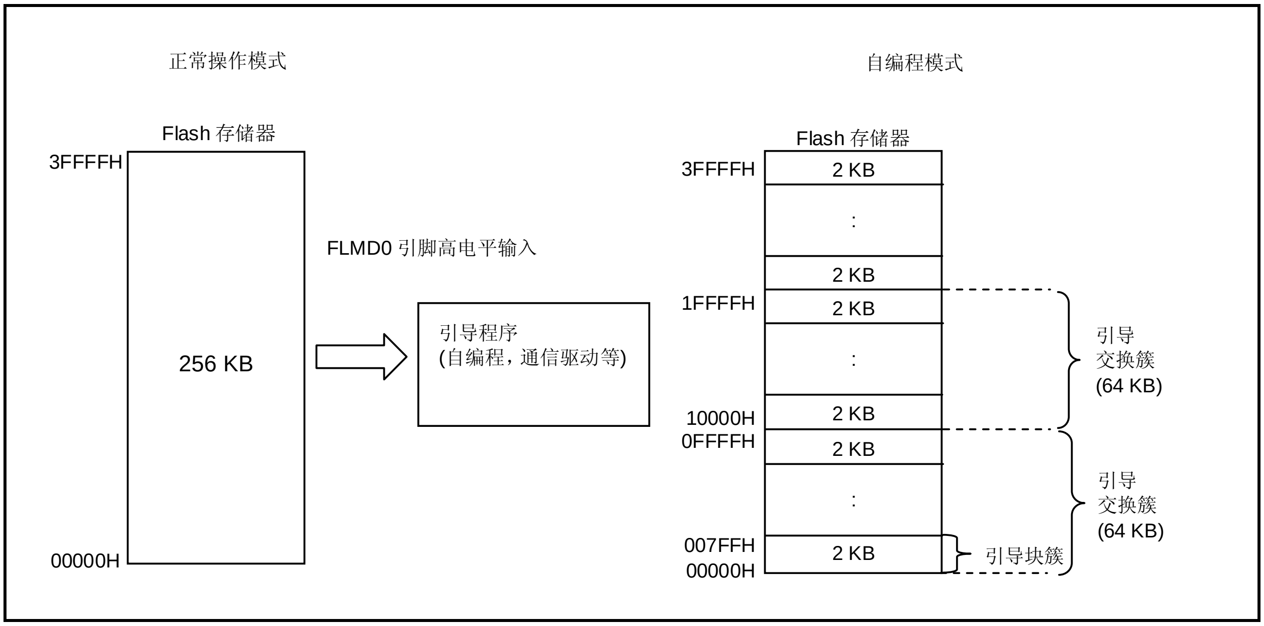
V850E/IF3 和 V850E/IG3 支持自编程功能，通过用户程序来重写 flash 存储器。通过使用 flash 存储器自编程，flash 存储器可以通过用户应用程序来重写。可以使用 flash 存储器自编程现场升级程序。

27.7.1 Flash 存储器自编程的概述

通过从一个程序调用 flash 功能，Flash 存储器自编程用于擦除或写入 flash 存储器，该程序存储于待擦除或写入的 flash 存储器区域之外的区域。要存储原来位于待擦除或写入的区域的实现自编程的程序，将程序拷贝到内部 RAM 区域，在拷贝目的地址执行改程序，并调用 flash 功能。

要调用 flash 功能，使用 flash 编程模式控制寄存器从正常操作模式改变为自编程模式。

图 27-1. 自编程的概述



（1）引导交换簇

执行 flash 存储器编程时，低地址侧（00000H 至 0FFFFH）的引导交换簇的内容和高地址侧（10000H 至 1FFFFH）的引导交换簇的内容可以内部交换。

（2）引导块簇

通过从 00000H 按照 2 KB 单元指定引导块簇，可以保护引导块簇的内容不被重写。

27.7.2 自编程库

Flash 存储器版本提供的自编程函数在表 27-9 中列出。通过调用应用程序的库函数来执行 flash 存储器的写入/擦除。

自编程库的特征如下。

- flash 存储器的内容可以通过应用程序而无需专用写入工具来改变。
- 可以由用户定义外部接口，因为它有独立的结构。
- 可以通过 C 语言来使用 NEC 电子提供的基本库功能。

图 27-2. 概述

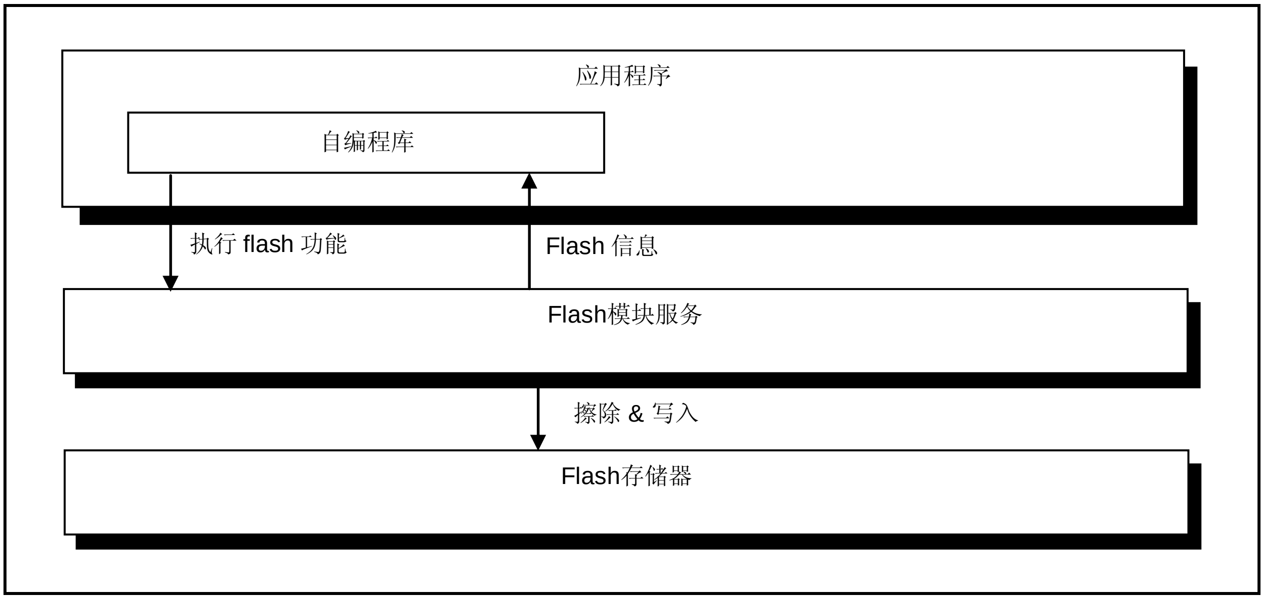


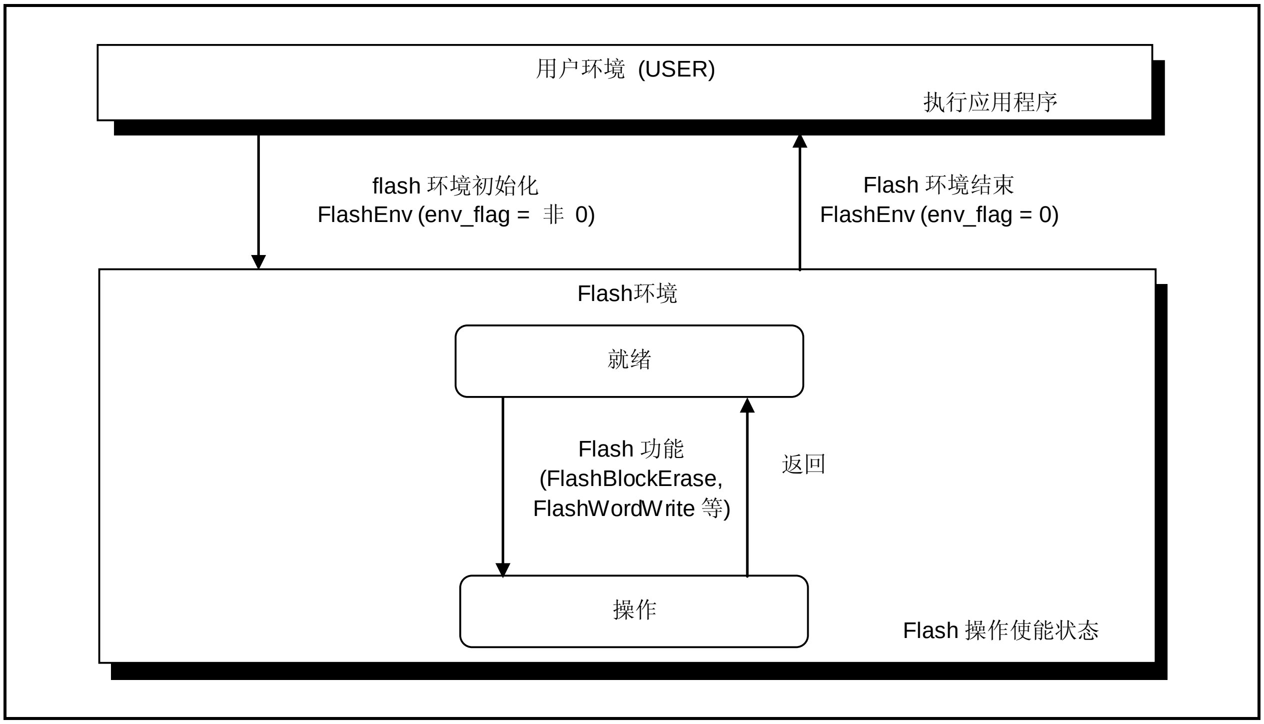
表 27-9. Flash 函数列表

类型	函数名称	缩写	功能
初始化	FlashEnv	FLE	初始化 flash 控制
擦除	FlashBlockErase	FLBE	擦除指定的块
写入	FlashWordWrite	FLWW	从指定的 flash 存储器地址开始，连续写入指定存储器内容，字的数量按照 4 字节单元指定
检查	FlashBlockBlankCheck	FLBBC	检查指定块的擦除状态
	FlashBlockIVerify	FLBIV	执行指定块的内部校验
	FlashFLMDCheck	FLFC	输入 FLMD0 引脚并检查 FLMD0 设置寄存器值
获取信息	FlashGetInfo	FLGI	读出 flash 存储器相关的信息
设置	FlashSetInfo	FLSI	设置 flash 信息
	FlashBootSwap	FLBS	内部改变引导交换簇的内容
	FlashWordRead	FLWR	读出指定地址的数据

27.7.3 Flash 环境

要执行 flash 存储器自编程，必须从用户环境改变到 flash 环境。Flash 环境是指通过使用 flash 函数（自编程库）重写或擦除 flash 存储器的环境。

图 27-3. Flash 环境



（1）用户环境

在该状态下用户应用程序的操作不执行 flash 存储器操作。要设置 flash 环境，使得 FLMD0 引脚为高，并在内部 RAM 执行 flash 环境启动处理函数 FlashEnv（env_flag = 非 0）。

（2）Flash 环境

在该环境下，可以通过 flash 函数（自编程库）擦除或写入 flash 存储器。要改变该环境为用户环境，执行 flash 环境结束处理 FlashEnv（env_flag = 0），然后在用户环境下，向 FLMD0 引脚输入一个低电平。

27.7.4 限制

flash 存储器自编程模式的限制解释如下。

（1）状态有关的限制

表 27-10. 状态有关的限制

类型	状态名称	限制
用户环境	USER	运行与正常操作模式
Flash 环境	READY	• 被 flash 自编程库使用的 RAM，不要写入该区域 • 向 FLMD0 引脚输入稳定的高电平电压
	OPERATION	• 不要输入复位信号 • 被 flash 自编程库使用的 RAM，不要写入该区域 • 向 FLMD0 引脚输入稳定的高电平电压

（2）flash 环境有关的限制

表 27-11. Flash 环境有关的限制

项目	限制
可屏蔽的中断	必需设置 JMP 指令来跳转到位于内部 RAM 起始地址的第四个字节的中断服务
输入到 X1 引脚的时钟	在 flash 环境下，不要停止输入到 X1 引脚的时钟
省电模式	不要使用省电模式
Flash 存储器区域	在 flash 环境下，不要在 flash 存储器内执行程序，在内部 RAM 中执行程序。 不要读取 flash 存储器区域。要读取 flash 存储器区域，使用 flash 函数 Flash Word Read

（**3**） 其他限制

（**a**） 内部 **RAM** 中的程序执行和内部 **RAM** 相关的 **DMA** 传输

如果满足下列所有条件，则 CPU 可能操作不正确。在该状态下，只可以接受复位信号。

[条件]

- 内部 RAM 中的位操作指令（SET1, CLR1 或 NOT1）
- 对内部 RAM 未对齐地址的数据访问指令

当执行上面任一指令时，执行对内部 RAM 的 DMA 传输。
因此，采取下列规避动作。

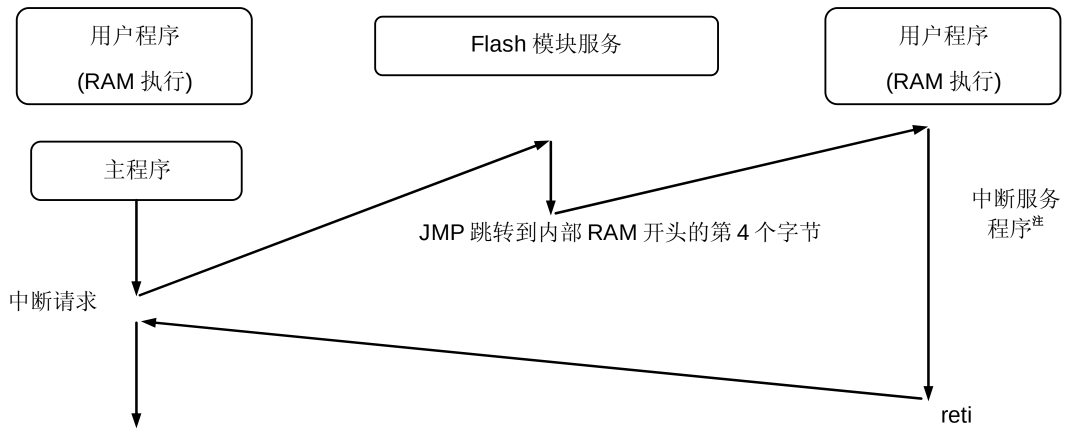
[动作]

- 要执行内部 RAM 相关的 DMA 传输，不要在内部 RAM 执行位操作指令（SET1，CLR1 或 NOT1），不要对内部 RAM 未对齐地址执行数据访问指令。
- 在内部 RAM 执行位操作指令（SET1，CLR1 或 NOT1）时，或对未对齐地址执行数据访问指令时，不要执行内部 RAM 相关的 DMA 传输。

27.7.5 Flash 环境下的中断服务

在 flash 环境下，可以执行可屏蔽中断服务。但是，这个服务和正常中断的服务不同。flash 环境下的中断服务的概述如下。

图 27-4. 中断服务的概述



注　通过中断服务程序读取 CPU 的 ECR 寄存器，并检查中断源。

在 flash 环境下产生一个中断时，flash 模块服务无条件跳转到内部 RAM 起始地址的第 4 个字节。判断中断是否于中断控制寄存器的中断请求位相对应，如果对应，执行中断服务，结束时使用 reti 指令。

（1）flash 环境下使用中断服务的注意事项

- 在用户程序中没有设置中断使能状态时，必须在内部 RAM 首地址的第 4 个字节设置中断服务。如果没有这样做，程序可能会因为中断发生跑飞。
- 不要为中断服务访问 flash 存储器区域。
- 不要为中断服务执行 flash 功能。
- 中断服务要使用通用寄存器，中断服务中必须执行保存和恢复。
- 不要使用复位的句柄地址（00000000H）和软件异常（Trap10）的句柄地址（00000050H），因为这些地址被 flash 模块服务使用。
- 在 FlashEnv 执行期间产生一个中断，该中断被挂起的时间最长为 50 μs。
- FlashEnv 函数包括中断句柄切换的执行（flash 存储器的起始地址 ⇔ 内部 RAM 起始地址的第 4 个字节）。请注意切换/返回自编程模式时，有可能对两个中断句柄都执行跳转。
- 每次执行 flash 功能，都会发生大约 100 个时钟（f_{CPU}）的中断保持时间。
- 不提供中断句柄的库代码。

（2）flash 环境下的中断反应时间

和正常中断的情况不同，flash 环境下的中断通过 flash 模块服务来处理，导致较长的中断反应时间。

Flash 模块服务处理时间（最大值）= 3 个 [时钟（f_{CPU}）]

备注　f_{CPU}：CPU 时钟

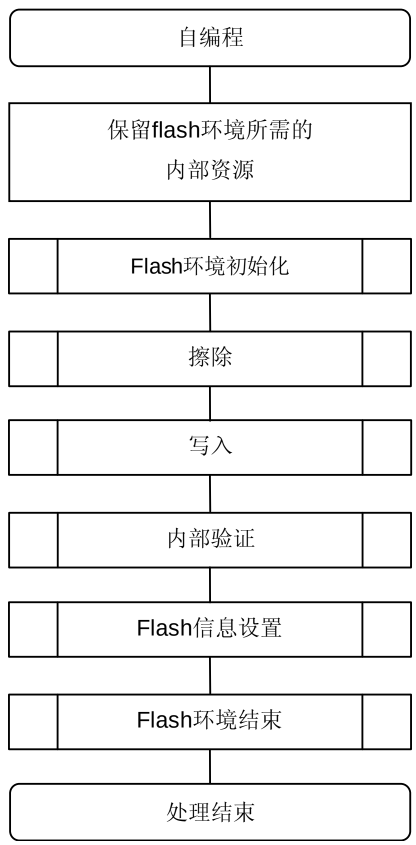
27.7.6 Flash 存储器自编程流程

（1） Flash 存储器自编程总流程（推荐）概述如下。

注意事项 flash 存储器自编程模式的使用要根据 **27.7.4 限制**。

（a） 单次执行重写时

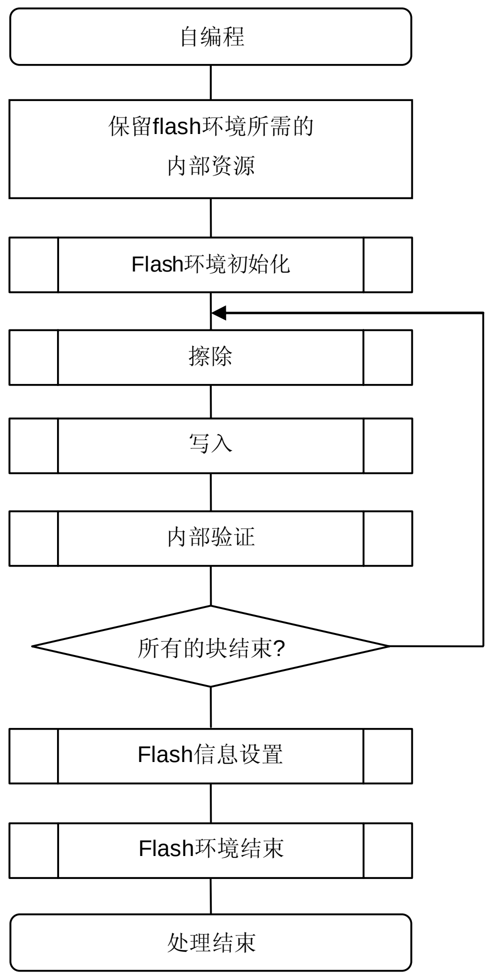
图 27-5. Flash 存储器自编程总流程（1/2）



备注 未设置安全功能时，不需要执行 flash 信息设置。

（**b**） 按块单元执行重写时

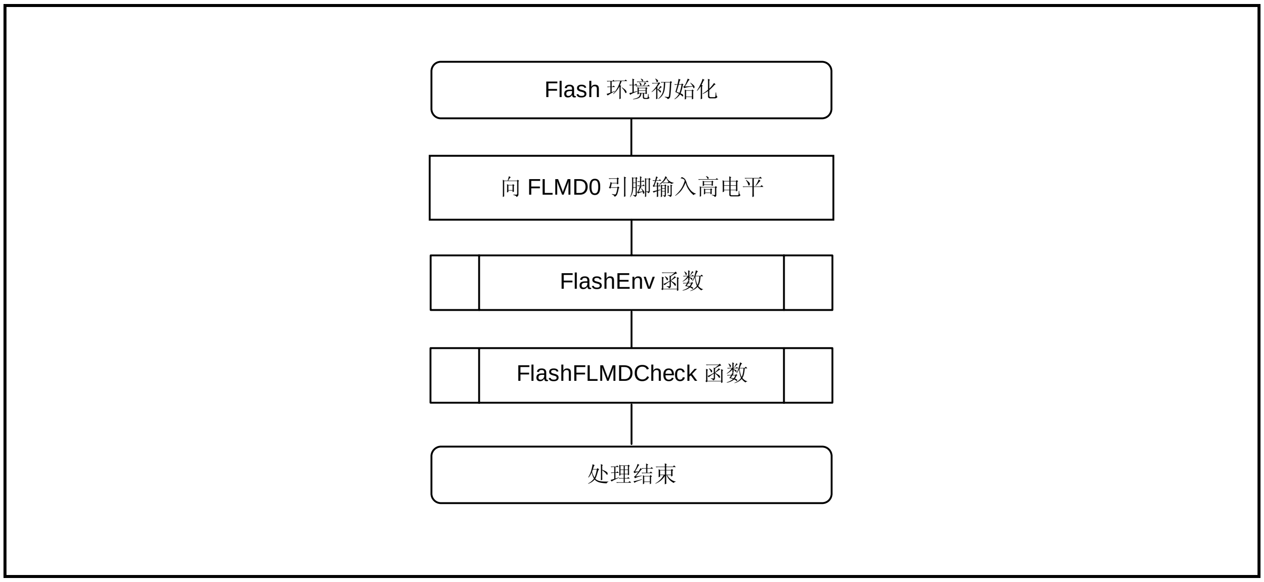
图 **27-5. Flash** 存储器自编程总流程（**2/2**）



备注 未设置安全功能时，不需要执行 flash 信息设置。

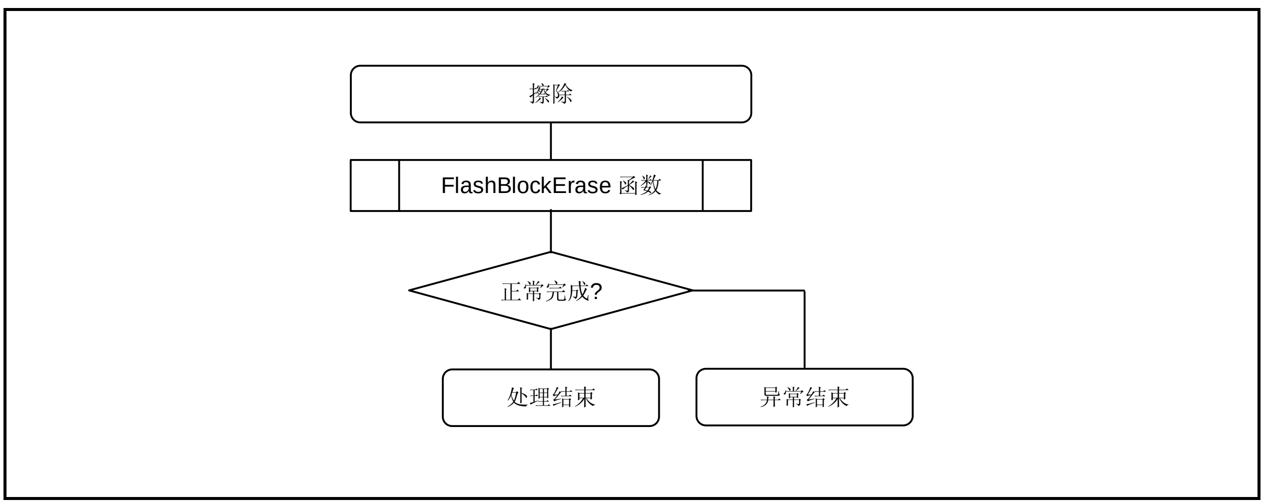
（2） **Flash** 环境初始化流程

图 **27-6.** **Flash** 环境初始化处理流程



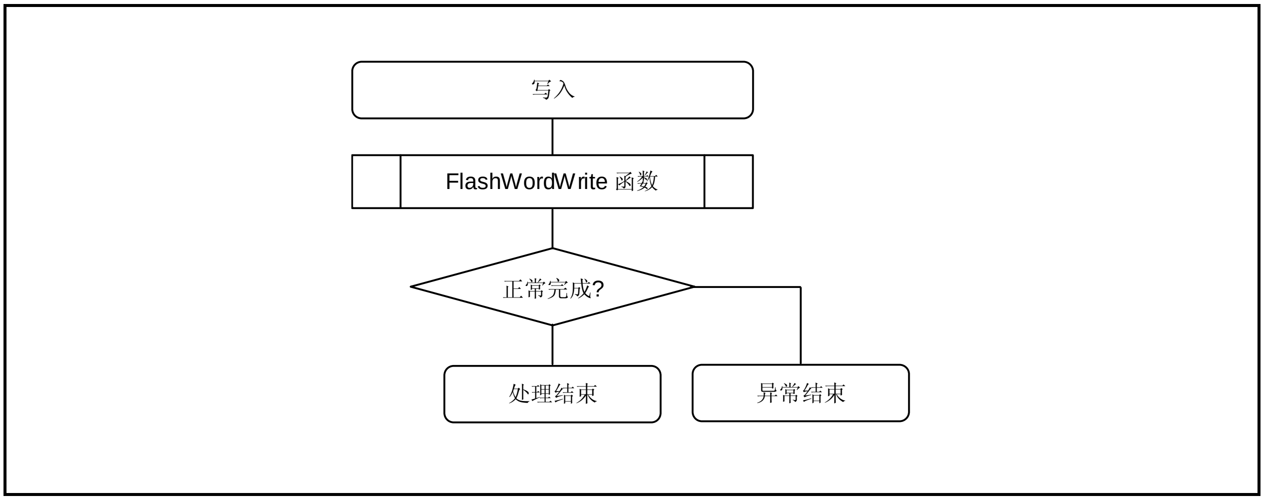
（3） 擦除流程

图 **27-7.** 擦除处理流程



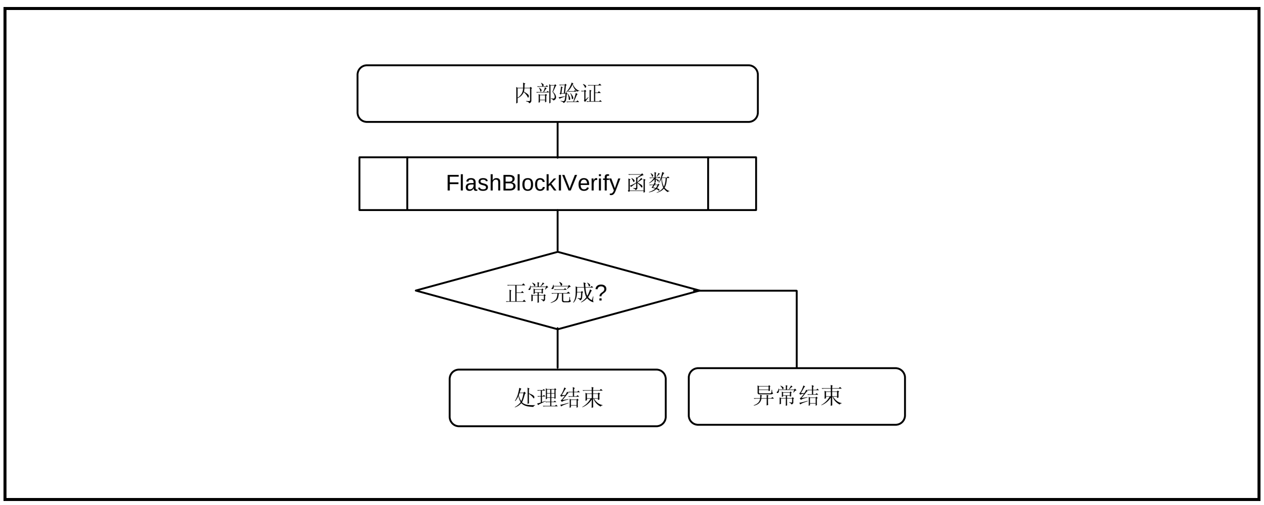
（**4**） 写入流程

图 **27-8.** 写入处理流程



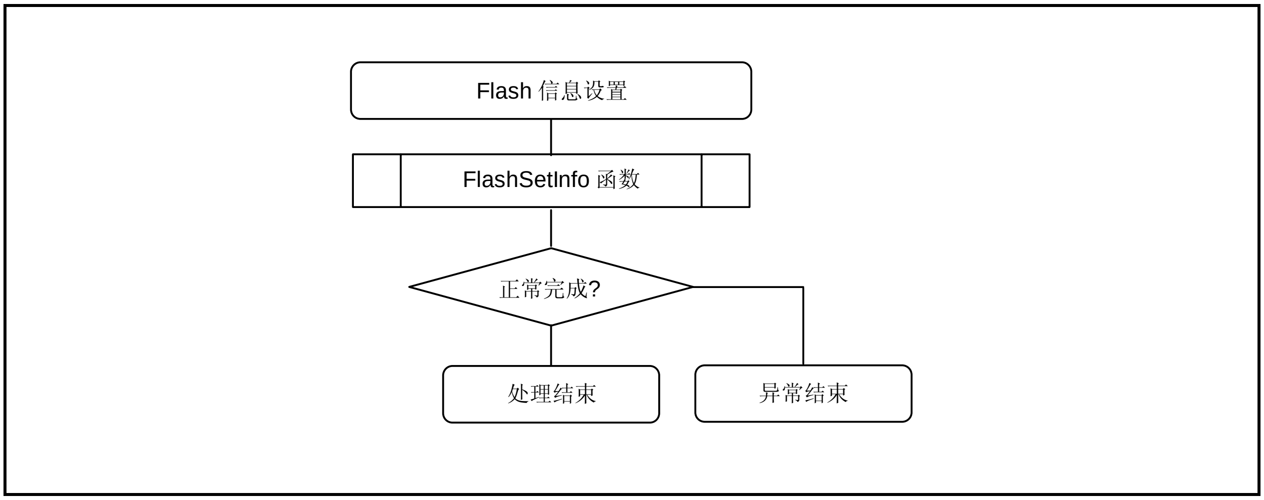
（**5**） 内部检验处理流程

图 **27-9.** 内部检验处理流程



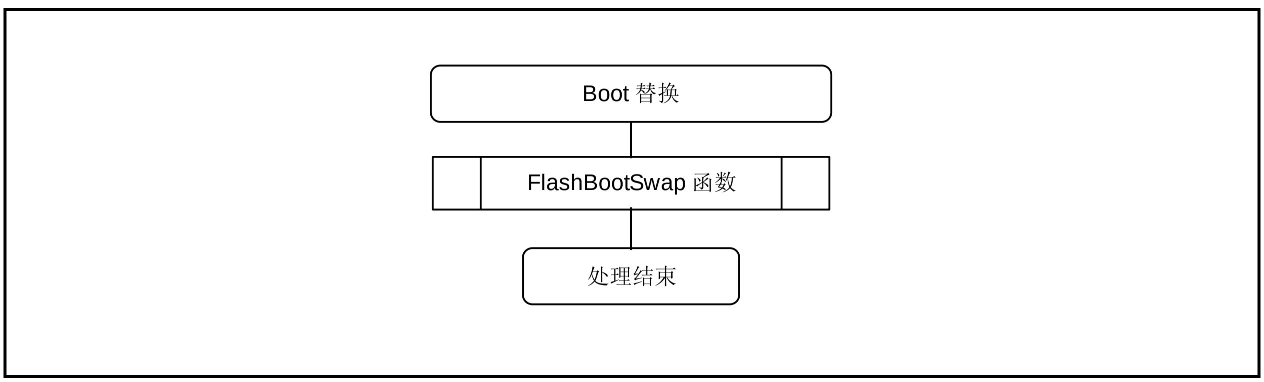
（6） Flash 信息设置流程

图 27-10. Flash 信息设置流程



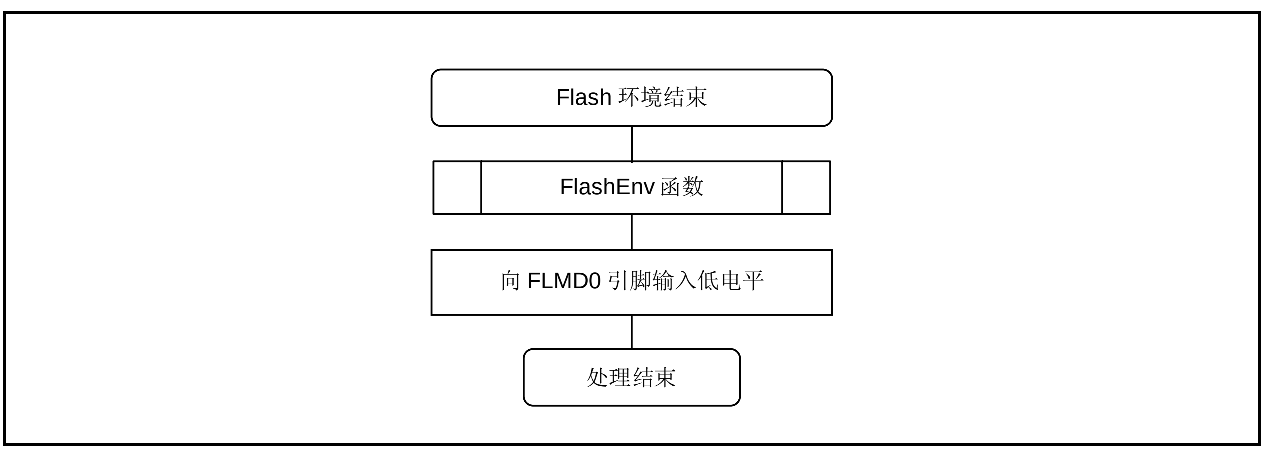
（7） 引导交换处理流程

图 27-11. 引导交换处理流程



（8） Flash 环境结束处理流程

图 27-12. Flash 环境结束处理流程



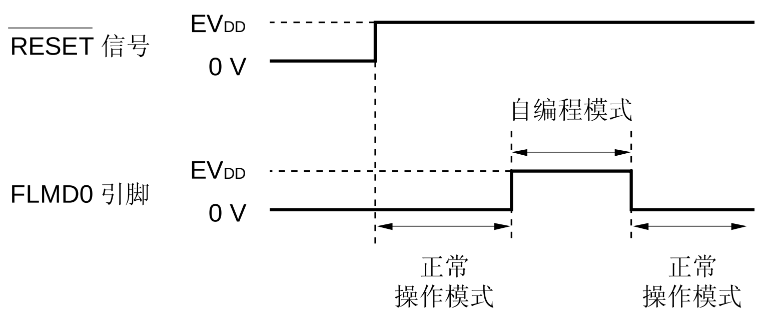
27.7.7 引脚处理

（1） FLMD0 引脚

FLMD0 引脚用于在复位解除后设置操作模式，并在自重写期间保护 flash 存储器不被写入。因此，在复位释放后且执行正常操作时，有必要保持输入到 FLMD0 引脚的电压为 0V。重写存储器之前，通过端口控制自编程模式期间，有必要向 FLMD0 引脚输入 EV_{DD0}, EV_{DD1}, EV_{DD2}(仅限 V850E/IG3)电平的电压。

当完成自编程时，FLMD0 引脚的电压值必须返回 0V。

图 27-13. 模式变换时序



备注　EV_{DD}：EV_{DD0}, EV_{DD1}, EV_{DD2}（仅限 V850E/IG3）

注意事项　当复位释放后，确保 FLMD0 引脚为 0V。

28.1 V850E/IF3

28.1.1 最大额定值

(T_A = 25°C)

参数	符号	条件		额定值范围	单位
供电电压	V_{DD}	$V_{DDa} = EV_{DDb} = AV_{DDk}$		–0.5 至+6.5	V
	V_{SS}	$V_{SSa} = EV_{SSb} = AV_{SSk}$		–0.5 至+0.5	V
	EV_{DD}	$V_{DDa} = EV_{DDb} = AV_{DDk}$		–0.5 至+6.5	V
	EV_{SS}	$V_{SSa} = EV_{SSb} = AV_{SSk}$		–0.5 至+0.5	V
	AV_{DD}	$V_{DDa} = EV_{DDb} = AV_{DDk}$		–0.5 至+6.5	V
	AV_{SS}	$V_{SSa} = EV_{SSb} = AV_{SSk}$		–0.5 至+0.5	V
输入电压	V_{I1}	注 **1**		–0.5 至 EV_{DD} + 0.5[注 2]	V
	V_{I2}	X1, X2		–0.5 至 V_{RO} + 0.35	V
输出电流，低	I_{OL}	所有引脚	每个引脚	4	mA
			全部引脚总和	42	mA
输出电流，高	I_{OH}	所有引脚	每个引脚	–4	mA
			全部引脚总和	–42	mA
模拟输入电压	V_{IAN}	P70/ANI20 至 P73/ANI23, ANI00 至 ANI05, ANI10 至 ANI17		–0.5 至 AV_{DD} + 0.5[注 2]	V
模拟参考输入电压	V_{IREF}	AV_{REFP0}, AV_{REFP1}		–0.5 至 AV_{DD} + 0.5[注 2]	V
比较器参考输入电压	V_{CREF}	CREF0L, CREF1L, CREF0F, CREF1F		–0.5 至 AV_{DD} + 0.5[注 2]	V
工作环境温度	T_A	正常操作模式		–40 至+85	°C
		Flash 存储器编程模式		–40 至+85	°C
工作环境温度	T_{stg}			–40 至+125	°C

注　**1.** P00, P01, P10 至 P17，P20 至 P27，P30 至 P37，P40 至 P47, PDL0 至 PDL9, $\overline{RESET}$, FLMD0

　　2. 请确保各个供电电压不超过最大额定值 (MAX.值)。

注意事项 **1.** 不要将 **IC** 产品的输出引脚（或处于输出状态下的 **I/O** 引脚）直接连接到其它输出引脚（包括处于输出状态下的 **I/O** 引脚），或直接连接至 **V_{DD}**、**EV_{DD}** 或 **GND**。如果输出引脚可以设置为高阻抗状态，并且设计外部电路的输出时序时已经考虑避免输出冲突，芯片和外部电路之间的输出引脚才可以直接相连。

2. 任何参数，哪怕瞬间超过最大额定值，都可能使产品性能受损。就是说，产品工作在最大额定值时处于物理损坏的边缘，因此，产品必须在不超出最大额定值的条件下使用。

DC 和 **AC** 特性的额定值和工作条件代表在正常工作期间的性能保障范围。

备注　a = 0, 1

b = 0, 1

k = 0 至 2

28.1.2 电容

(T_A = 25°C, V_{DD0} = V_{SS0} = V_{DD1} = V_{SS1} = EV_{DD0} = EV_{SS0} = EV_{DD1} = EV_{SS1} = AV_{DD0} = AV_{SS0} = AV_{DD1} = AV_{SS1} = AV_{DD2} = AV_{SS2} = 0 V)

参数	符号	条件		MIN.	TYP.	MAX.	单位
输入电容	C_I	fc = 1 MHz 未测量的引脚返回至 0 V	**注 1**			15	pF
I/O 电容	C_{IO}		**注 2**			15	pF

注 **1.** ANI00 至 ANI05, ANI10 至 ANI17, $\overline{\text{RESET}}$

2. P00, P01, P10 至 P17，P20 至 P27，P30 至 P37，P40 至 P47, P70 至 P73, PDL0 至 PDL9

注意事项 **1.** 不包括 **FLMD0, X1** 和 **X2** 引脚。

2. 除了输入电容，**ANI00** 至 **ANI05, ANI10** 至 **ANI17** 和 **ANI20** 至 **ANI27** 引脚在采样时还要加上采样电容。

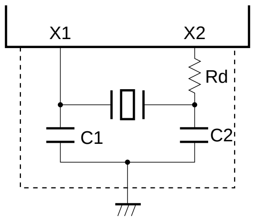
28.1.3 工作条件

(T_A = –40 至+85°C, V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V)

参数	符号	条件	MIN.	TYP.	MAX.	单位
系统时钟频率	f_{XX}	PLL 模式	32		64	MHz
		时钟直通模式	4		8	MHz
CPU 时钟频率	f_{CPU}	PLL 模式	4		64	MHz
		时钟直通模式	0.5		8	MHz
V_{DD}, EV_{DD} 电压	V_{DD}, EV_{DD}	V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = AV_{DD0} = AV_{DD1} = AV_{DD2}	3.5		5.5	V
AV_{DD} 电压	AV_{DD}	当 A/D 转换器 0 至 2 工作时	4.0		5.5	V
		当 A/D 转换器 0 至 2 不工作时	3.5		5.5	V

28.1.4 时钟振荡器特性

(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = AV_{DD0} = AV_{DD1} = AV_{DD2} = 3.5 至 5.5 V, V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V)

谐振器	推荐电路	参数	条件	MIN.	TYP.	MAX.	单位
陶瓷/晶体谐振器		振荡频率(f_X)		4		8	MHz
		振荡稳定时间	复位释放后		$2^{14}/f_X$		ms
			STOP 模式解除后		注		ms

注　　该值根据振荡稳定时间选择寄存器 (OSTS)的设置而不同。

注意事项
1. **振荡器的连接要尽量靠近 X1 和 X2 引脚。**
2. **上图中虚线框内的连线应避免和其它的信号线互相交叉。**
3. **谐振器的选择和振荡器常数，客户需要自行评估振荡性能，或向振荡器生产商申请对实现电路的评估。**
4. **禁止向 V850E/IF3 输入外部时钟。**

<R> **(i) Murata Mfg. Co., Ltd.：陶瓷谐振器 (T_A = –40 至+85°C)**

类型	零件登记号	振荡频率 f_X (MHz)	推荐电路常数			振荡电压范围	
			C1	C2	Rd (kΩ)	MIN. (V)	MAX. (V)
引线式	CSTLS4M00G56-B0	4	片上 (47 pF)	片上 (47 pF)	0	3.5	5.5
	CSTLS5M00G56-B0	5	片上 (47 pF)	片上 (47 pF)	0	3.5	5.5
	CSTLS8M00G56-B0	8	片上 (47 pF)	片上 (47 pF)	0	3.5	5.5
表面贴装	CSTCR4M00G55-R0	4	片上 (39 pF)	片上 (39 pF)	0	3.5	5.5
	CSTCR5M00G55-R0	5	片上 (39 pF)	片上 (39 pF)	0	3.5	5.5
	CSTCE8M00G55-R0	8	片上 (33 pF)	片上 (33 pF)	0	3.5	5.5

注意事项　**该振荡常数是振荡器生产商在特定环境下进行评估得出的参考值。如果在实际应用中必须对振荡器特性进行优化，请向振荡器生产商申请对实现电路的评估。**
振荡电压和振荡频率仅表示振荡器特性。使用 V850E/IF3 时内部工作条件不要超出 AC 特性和 DC 特性的额定值。

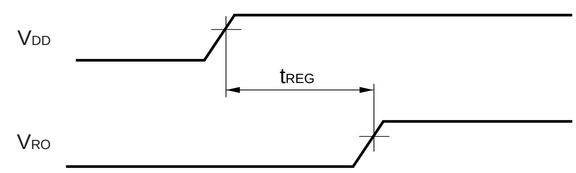
28.1.5 稳压器特性

(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = 3.5 至 5.5 V, V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = 0 V)

参数	符号	条件	MIN.	TYP.	MAX.	单位
输入电压	V_{DD}		3.5		5.5	V
输出电压	V_{RO}			1.5		V
输出电压稳定时间	t_{REG}	稳定电容：C = 4.7 μF[注 1]			1[注 2]	ms

注　**1.** 在 REGC0 引脚和 V_{SS0} 引脚、REGC1 引脚和 V_{SS1} 引脚之间各连接一个稳定电容。

　　2. 在 t_{REG} 时期，输出内部复位信号，直到上电清零电路(POC) 的输出电压稳定。



28.1.6 DC 特性

(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = AV_{DD0} = AV_{DD1} = AV_{DD2} = 3.5 至 5.5 V, V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V) (1/2)

参数	符号	条件		MIN.	TYP.	MAX.	单位
输入电压，高	V_{IH1}	**注 1**		0.7EV_{DD}		EV_{DD}	V
	V_{IH2}	**注 2**		0.8EV_{DD}		EV_{DD}	V
	V_{IH3}	**注 3**		0.7AV_{DD}		AV_{DD}	V
输入电压，低	V_{IL1}	**注 1**		EV_{SS}		0.3EV_{DD}	V
	V_{IL2}	**注 2**		EV_{SS}		0.2EV_{DD}	V
	V_{IL3}	**注 3**		AV_{SS}		0.3AV_{DD}	V
输入漏电流，高	I_{LIH1}	V_I = **注 4**	除 X1 之外			5	μA
	I_{LIH2}		X1			20	μA
输入漏电流，低	I_{LIL1}	V_I = 0 V	除 X1 之外			–5	μA
	I_{LIL2}		X1			–20	μA
输出漏电流，高	I_{LOH}	V_O = **注 4**				5	μA
输出漏电流，低	I_{LOL}	V_O = 0 V				–5	μA
输出电压，高	V_{OH1}	**注 5**	I_{OH} = –1.0 mA	EV_{DD} – 1.0			V
输出电压，低	V_{OL1}	**注 5**	I_{OL} = 1.0 mA			0.4	V
上拉电阻	R_{L1}			10	30	100	kΩ

注
1. P33, P36, P41 和 PDL0 至 PDL9 引脚
2. P00, P01, P10 至 P17，P20 至 P27，P30 至 P32, P34, P35, P37, P40, P42 至 P47, $\overline{\text{RESET}}$ 和 FLMD0 引脚
3. P70 至 P73 引脚
4. AV_{DD0} = AV_{DD1} = AV_{DD2} = EV_{DD0} = EV_{DD1}
5. P00, P01, P10 至 P17，P20 至 P27，P30 至 P37，P40 至 P47 和 PDL0 至 PDL9 引脚

备注　复用功能引脚的特性和端口引脚的特性相同。

(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = AV_{DD0} = AV_{DD1} = AV_{DD2} = 3.5 至 5.5 V,
V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V) (2/2)

参数	符号	条件		MIN.	TYP.注 1	MAX.	单位
V_{DD} 供电电流注 2	I_{DD1}	f_{XX} = 64 MHz	正常工作		64	93	mA
	I_{DD2}		HALT 模式		42	60	mA
	I_{DD3}		IDLE 模式		5	10	mA
	I_{DD4}	STOP 模式			40	800	μA

注　**1.** TYP.值是 V_{DD0} = V_{DD1} = 5.0 V 和 T_A = 25°C 情况下的参考值。

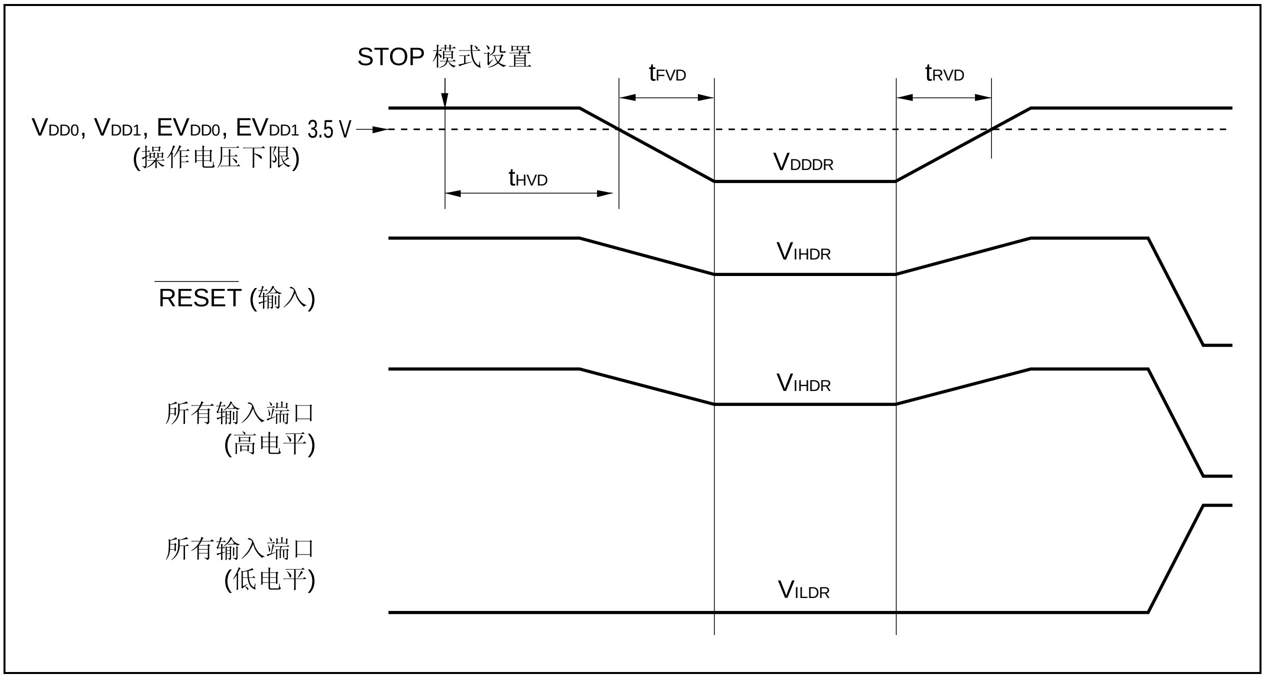
2. 不包括 EV_{DD} 系统(输出缓冲和上拉电阻)消耗的电流，也不包括 A/D 转换器 0 至 2，运算放大器和比较器的工作电流。

28.1.7 数据保持特性

STOP 模式 (T_A = –40 至+85°C, V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V)

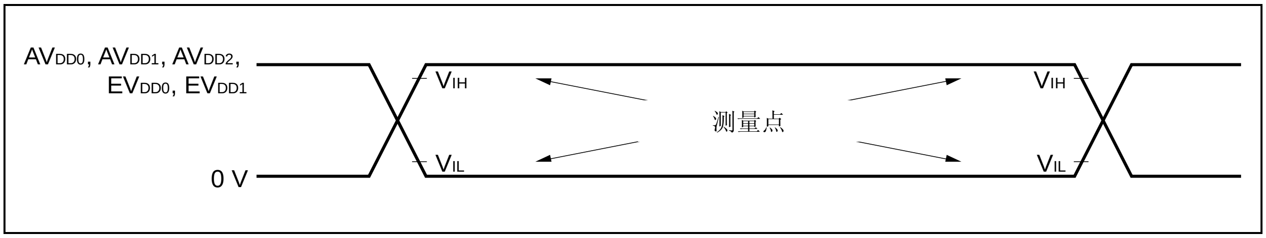
参数	符号	条件	MIN.	TYP.	MAX.	单位
数据保持电压	V_{DDDR}	STOP 模式	注		5.5	V
数据保持电流	I_{DDDR}	V_{DD0} = V_{DD1} = V_{DDDR}		40	800	μA
供电电压上升时间	t_{RVD}		1			μs
供电电压下降时间	t_{FVD}		1			μs
供电电压保持时间 (从设置 STOP 模式)	t_{HVD}		0			ms
数据保持输入电压，高	V_{IHDR}	All 输入端口	0.9V_{DDDR}		V_{DDDR}	V
数据保持输入电压，低	V_{ILDR}	All 输入端口	EV_{SS}		0.1V_{DDDR}	V

注 不使用低电压检测器 (LVI)复位模式 (LVIM.LVIMD 位 = 0)时： POC 检测电压 (V_{POC0})
使用低电压检测器 (LVI)复位模式(LVIM.LVIMD 位 = 1)时： LVI 检测电压 (V_{LVI0}/V_{LVI1})

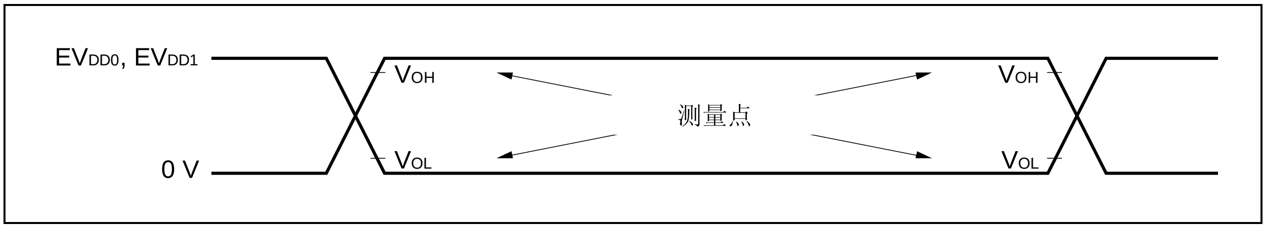


28.1.8 AC 特性

AC 测试输入测量点(除 CSIB0 至 CSIB2 之外的引脚)

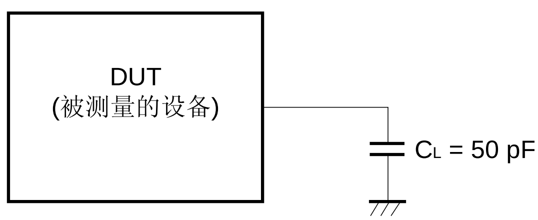


AC 测试输出测量点(除 CSIB0 至 CSIB2 之外的引脚)



AC 测试 I/O 测量点 (CSIB0 至 CSIB2 引脚)

负载条件

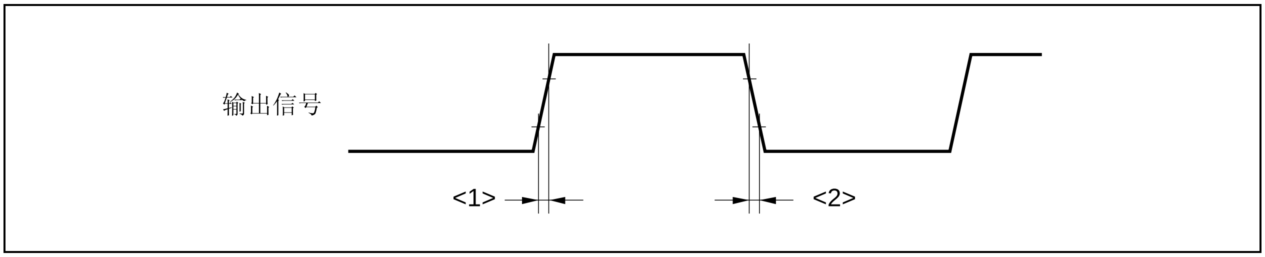


注意事项　如果由于电路配置使得负载电容超过 **50 pF**，可插入一个缓冲器或其它手段使设备的负载电容等于或小于 **50 pF**。

(1) 输出信号时序

(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = AV_{DD0} = AV_{DD1} = AV_{DD2} = 4.0 至 5.5 V, V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V, C_L = 50 pF)

参数	符号		条件	MIN.	MAX.	单位
输出上升时间	t_{OR}	<1>	PDL0 至 PDL9		8	ns
			其它情况		15	ns
输出下降时间	t_{OF}	<2>	PDL0 至 PDL9		8	ns
			其它情况		15	ns



(2) 复位，外部中断时序

(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = AV_{DD0} = AV_{DD1} = AV_{DD2} = 4.0 至 5.5 V, V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V, C_L = 50 pF)

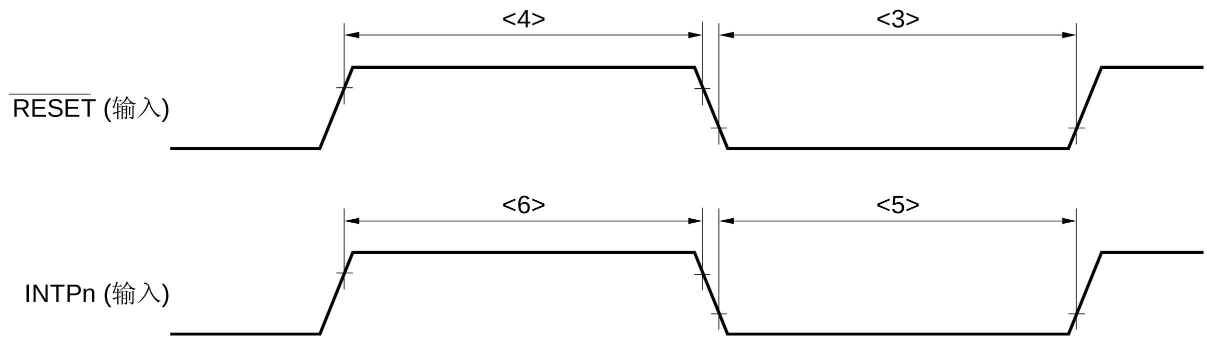
参数	符号		条件	MIN.	MAX.	单位
$\overline{\text{RESET}}$ 低电平宽度	t_{WRSL}	<3>	上电，解除 STOP 模式	500 + T_{os}		ns
			其它情况	500		ns
$\overline{\text{RESET}}$ 高电平宽度	t_{WRSH}	<4>		500		ns
INTPn 低电平宽度	t_{WITL}	<5>	n = 00, 01, 08 至 13, 17, 18 (模拟噪声消除)	500		ns
			n = 14 至 16 (数字噪声消除)	$4T_{smp}$		ns
INTPn 高电平宽度	t_{WITH}	<6>	n = 00, 01, 08 至 13, 17, 18 (模拟噪声消除)	500		ns
			n = 14 至 16 (数字噪声消除)	$4T_{smp}$		ns

备注　**1.** T_{os}：振荡稳定时间

T_{smp}：噪声消除采样时钟周期(由 INTNFCn 寄存器设置)

2. 复位释放后，当振荡频率(fx) = 8 MHz.时，内部保障 1 ms 振荡稳定时间。因此，振荡稳定时间是(T_{os} + 1) ms。STOP 模式解除之后，设置到 OSTS 寄存器振荡稳定时间值的一半由内部保障。因此，如果可以通过 OSTS 寄存器的设置保障足够稳定时间，可以接受 T_{os} = 0 ns。

复位/中断



备注　　n = 00, 01, 08 至 13, 17, 18

(3) 定时器时序

(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = AV_{DD0} = AV_{DD1} = AV_{DD2} = 4.0 至 5.5 V,

V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V, C_L = 50 pF)

参数	符号		条件	MIN.	MAX.	单位
TIn高电平宽度注 1, 2	t_{WTIHn}	<7>	n = B00 至 B03, B10 至 B13	12T + 10		ns
			n = A20, A21, A40, A41	3T_{smp1} + 10		ns
TIn 低电平宽度注 1, 2	t_{WTILn}	<8>	n = B00 至 B03, B10 至 B13	12T + 10		ns
			n = A20, A21, A40, A41	3T_{smp1} + 10		ns
EVTBm高电平宽度注 1	t_{WEVBHm}	<9>	m = 0, 1	12T + 10		ns
EVTBm 低电平宽度注 1	t_{WEVBLm}	<10>	m = 0, 1	12T + 10		ns
TRGBm高电平宽度注 1	t_{WTRHm}	<11>	m = 0, 1	12T + 10		ns
TRGBm 低电平宽度注 1	t_{WTRLm}	<12>	m = 0, 1	12T + 10		ns
TENC10/TENC11 高电平宽度注 3	t_{WENCH1}	<13>		3T_{smp2} + 10		ns
TENC10/TENC11 低电平宽度注 3	t_{WENCL1}	<14>		3T_{smp2} + 10		ns
TECR1 高电平宽度注 3	t_{WCRH1}	<15>		3T_{smp2} + 10		ns
TECR1 低电平宽度注 3	t_{WCRL1}	<16>		3T_{smp2} + 10		ns
TIT10/TIT11 高电平宽度注 3	t_{WTITH1}	<17>		3T_{smp2} + 10		ns
TIT10/TIT11 低电平宽度注 3	t_{WTITL1}	<18>		3T_{smp2} + 10		ns
EVTT1 高电平宽度注 3	t_{WEVTH1}	<19>		3T_{smp2} + 10		ns
EVTT1 低电平宽度注 3	t_{WEVTL1}	<20>		3T_{smp2} + 10		ns
TENC10/TENC11 输入时间差别注 3	t_{PHUD1}	<21>		3T_{smp2} + 10		ns

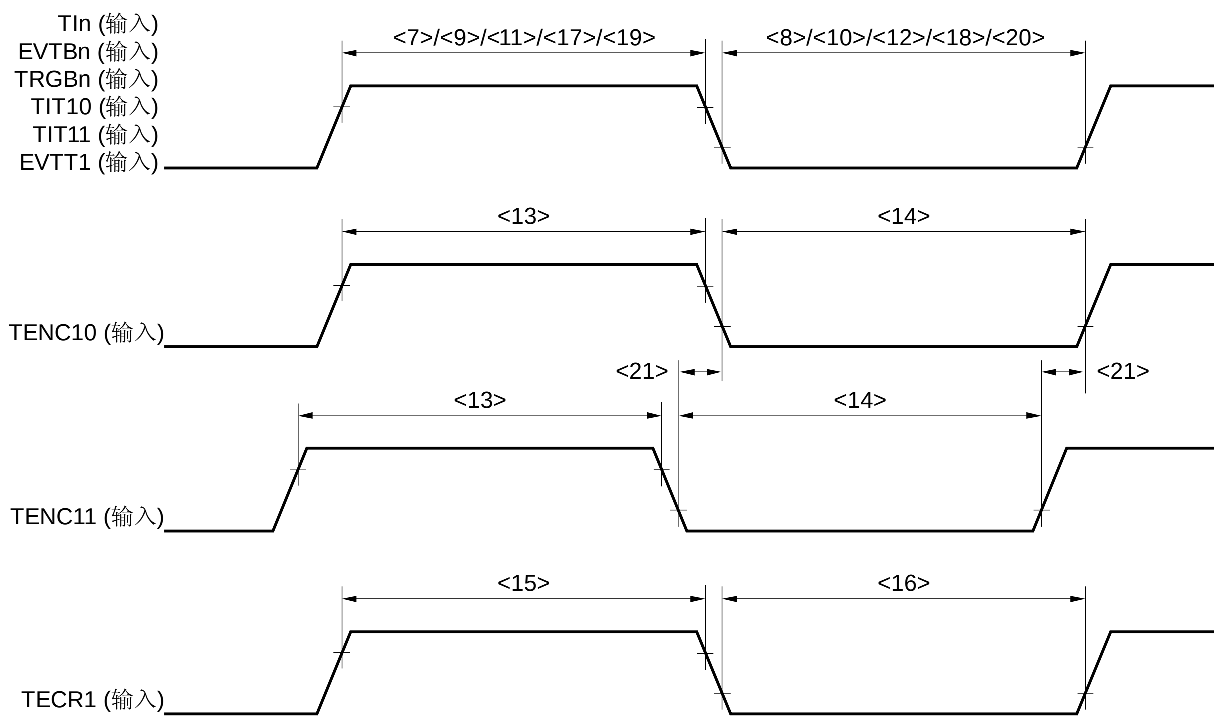
注　**1.** T = 1/f_{XX}

2. T_{smp1}：噪声消除采样时钟周期(由 TANFC2 和 TANFC4 寄存器设置)

3. T_{smp2}：噪声消除采样时钟周期(由 TTNFC1 寄存器设置)

备注　上述规范是以有效沿来精确检测后的脉冲宽度。因此，即使输入的脉冲窄于上述规范，也可以以有效沿来检测。

定时器输入时序



备注　n = A20, A21, A40, A41, B00 至 B03, B10 至 B13

(4) CSIB 时序

(a) 主模式

(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = AV_{DD0} = AV_{DD1} = AV_{DD2} = 4.0 至 5.5 V, V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V, C_L = 50 pF)

参数	符号		条件	MIN.	MAX.	单位
$\overline{\text{SCKBn}}$ 周期	t_{KCYM}	<22>		125		ns
$\overline{\text{SCKBn}}$ 高/低电平宽度	t_{KWHM}, t_{KWLM}	<23>		$t_{KCYM}/2 - 10$		ns
SIBn 建立时间(至 $\overline{\text{SCKBn}}$↑)	t_{SSIM}	<24>		30		ns
SIBn 建立时间(至 $\overline{\text{SCKBn}}$↓)				30		ns
SIBn 保持时间 (自 $\overline{\text{SCKBn}}$↑)	t_{HSIM}	<25>		30		ns
SIBn 保持时间 (自 $\overline{\text{SCKBn}}$↓)				30		ns
SOBn 输出延迟时间 (自 $\overline{\text{SCKBn}}$↓)	t_{DSOM}	<26>			30	ns
SOBn 输出延迟时间 (自 $\overline{\text{SCKBn}}$↑)					30	ns
SOBn 输出保持时间 (自 $\overline{\text{SCKBn}}$↑)	t_{HSOM}	<27>		$t_{KCYM}/2 - 10$		ns
SOBn 输出保持时间 (自 $\overline{\text{SCKBn}}$↓)				$t_{KCYM}/2 - 10$		ns

备注　　n = 0 至 2

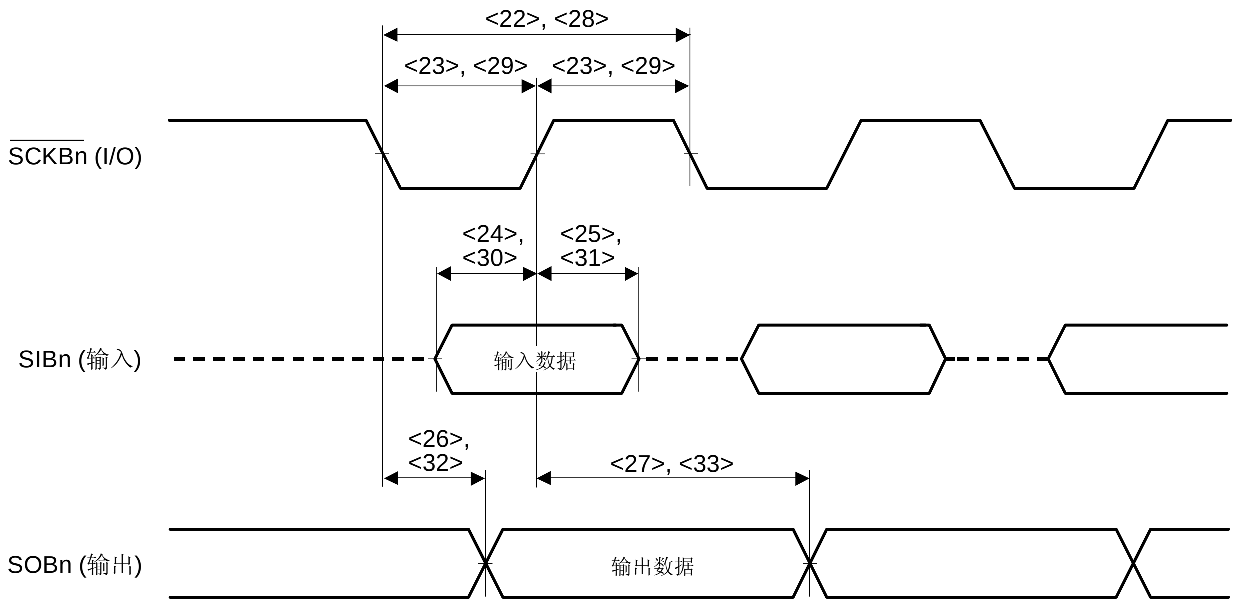
(b) 从模式

(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = AV_{DD0} = AV_{DD1} = AV_{DD2} = 4.0 至 5.5 V, V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V, C_L = 50 pF)

参数	符号		条件	MIN.	MAX.	单位
$\overline{\text{SCKBn}}$ 周期	t_{KCYS}	<28>		125		ns
$\overline{\text{SCKBn}}$ 高/低电平宽度	t_{KWHS}, t_{KWLS}	<29>		$t_{KCYS}/2 - 10$		ns
SIBn 建立时间(至 $\overline{\text{SCKBn}}$↑)	t_{SSIS}	<30>		30		ns
SIBn 建立时间(至 $\overline{\text{SCKBn}}$↓)				30		ns
SIBn 保持时间 (自 $\overline{\text{SCKBn}}$↑)	t_{HSIS}	<31>		30		ns
SIBn 保持时间 (自 $\overline{\text{SCKBn}}$↓)				30		ns
SOBn 输出延迟时间 (自 $\overline{\text{SCKBn}}$↓)	t_{DSOS}	<32>			30	ns
SOBn 输出延迟时间 (自 $\overline{\text{SCKBn}}$↑)					30	ns
SOBn 输出保持时间 (自 $\overline{\text{SCKBn}}$↑)	t_{HSOS}	<33>		$t_{KCYS}/2 - 10$		ns
SOBn 输出保持时间 (自 $\overline{\text{SCKBn}}$↓)				$t_{KCYS}/2 - 10$		ns

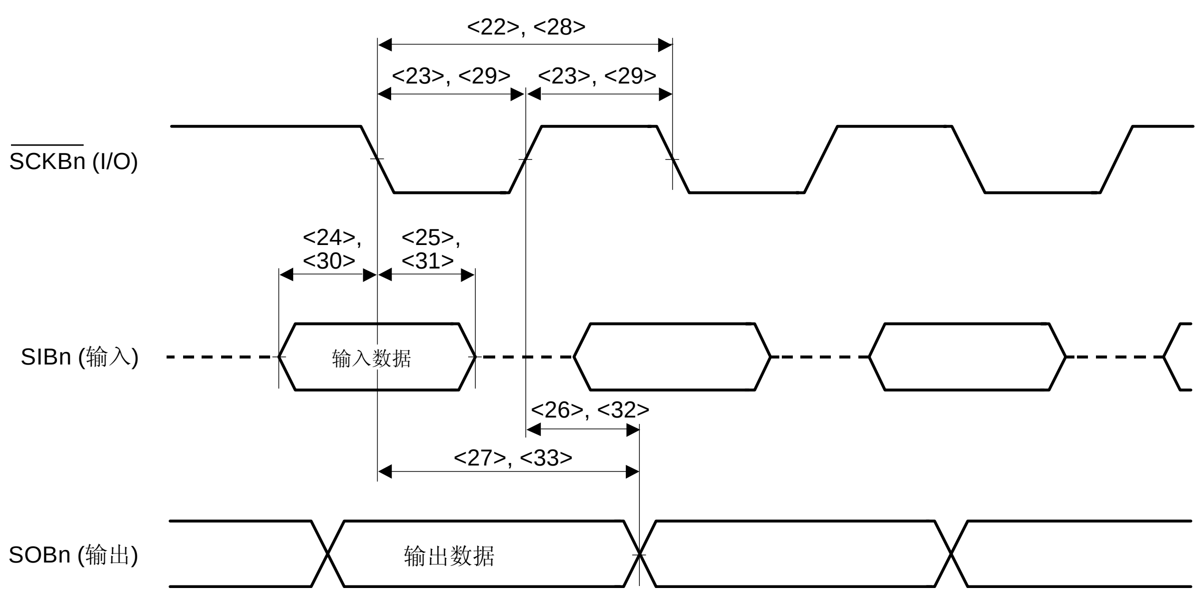
备注　　n = 0 至 2

当 **CBnCTL1** 寄存器的 **CBnCKP** 和 **CBnDAP** 位 **= 00** 时的 **CSIB** 时序



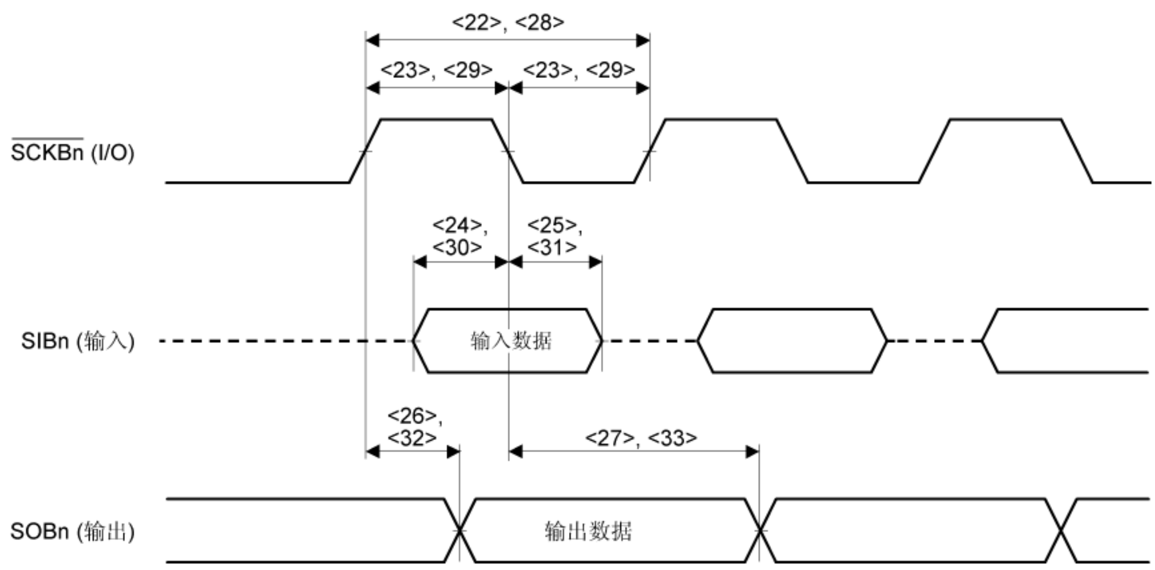
备注　**1.** 虚线表示高阻抗
2. n = 0 至 2

当 **CBnCTL1** 寄存器的 **CBnCKP** 和 **CBnDAP** 位 **= 01** 时的 **CSIB** 时序



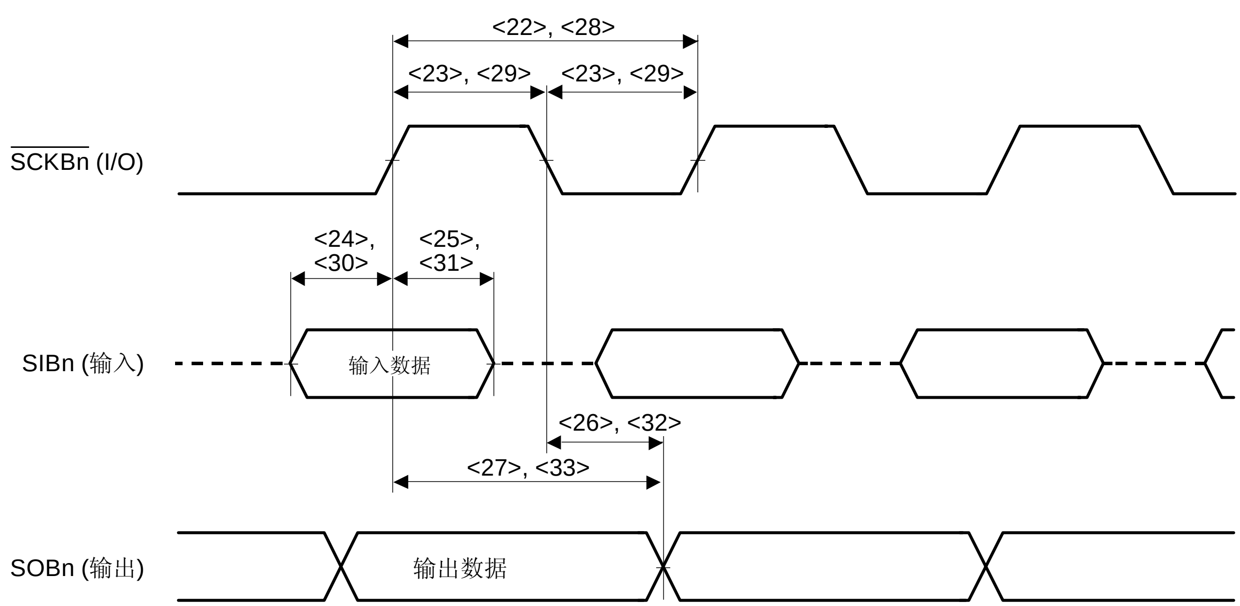
备注　**1.** 虚线表示高阻抗
2. n = 0 至 2

当 CBnCTL1 寄存器的 CBnCKP 和 CBnDAP 位 = 10 时的 CSIB 时序



备注 **1.** 虚线表示高阻抗
2. n = 0 至 2

当 CBnCTL1 寄存器的 CBnCKP 和 CBnDAP 位 = 11 时的 CSIB 时序



备注 **1.** 虚线表示高阻抗
2. n = 0 至 2

(5) I²C 总线时序

(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = AV_{DD0} = AV_{DD1} = AV_{DD2} = 4.0 至 5.5 V, V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V, C_L = 50 pF)

参数		符号		标准模式		高速模式		单位
				MIN.	MAX.	MIN.	MAX.	
SCL 时钟频率		f_{CLK}	–	0	100	0	400	kHz
总线自由时间 (停止条件和开始条件之间)		t_{BUF}	<34>	4.7	–	1.3	–	μs
保持时间[注1]		$t_{HD:STA}$	<35>	4.0	–	0.6	–	μs
SCL 时钟低电平宽度		t_{LOW}	<36>	4.7	–	1.3	–	μs
SCL 时钟高电平宽度		t_{HIGH}	<37>	4.0	–	0.6	–	μs
开始/重启条件建立时间		$t_{SU:STA}$	<38>	4.7	–	0.6	–	μs
数据保持时间	CBUS-兼容主	$t_{HD:DAT}$	<39>	5.0	–	–	–	μs
	I²C模式			0[注2]	–	0[注2]	0.9[注3]	μs
数据建立时间		$t_{SU:DAT}$	<40>	250	–	100[注4]	–	ns
SDA，SCL 信号上升时间		t_R	<41>	–	1000	20 + 0.1Cb[注5]	300	ns
SDA，SCL 信号下降时间		t_F	<42>	–	300	20 + 0.1Cb[注5]	300	ns
停止条件建立时间		$t_{SU:STO}$	<43>	4.0	–	0.6	–	μs
输入滤波器滤除的毛刺脉冲宽度		t_{SP}	<44>	–	–	0	50	ns
每条总线的容性负载		Cb	–	–	400	–	400	pF

注

1. 开始条件期间，在保持时间之后产生第一个时钟脉冲。

2. 系统必须为 SDA 信号内部提供至少 300 ns 的保持时间，(在 SCL 信号的 V_{IHmin}) 用于在 SCL 下降沿处填充不确定区域。

3. 如果系统不延长 SCL 信号的低状态保持时间 (t_{LOW})，必须满足最大数据保持时间($t_{HD:DAT}$)。

4. 在标准模式 I²C 总线系统下可以使用高速模式 I²C 总线。在这种情况下，请确保满足下列条件。

- 如果系统不延长 SCL 信号的低状态保持时间。

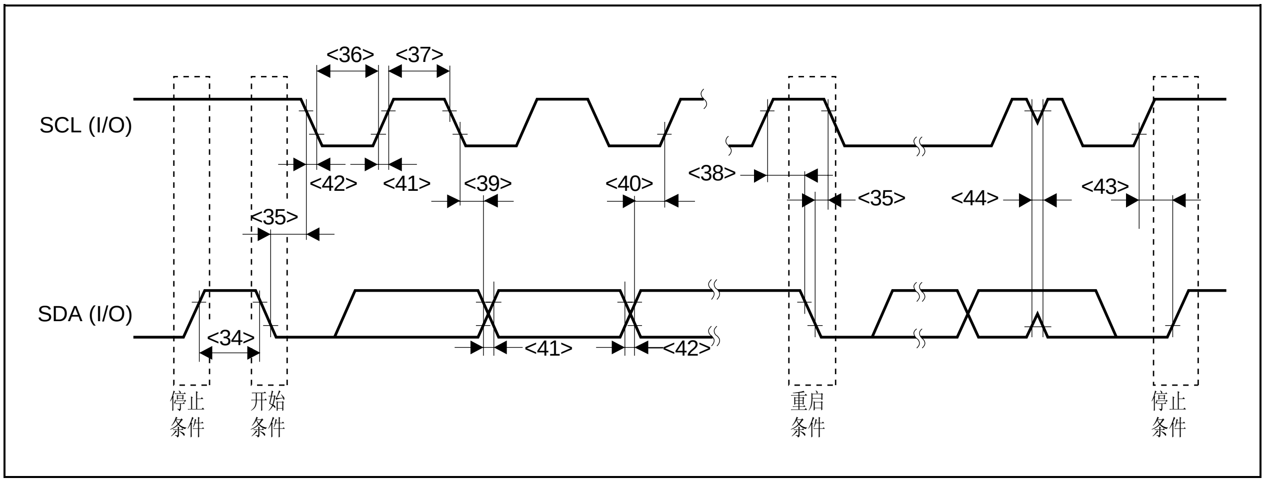
  $t_{SU:DAT} \geq 250$ ns

- 如果系统延长 SCL 信号的低状态保持时间。

  在SCL线释放之前，将下一个数据位发送到SDA线($t_{Rmax.}$ + $t_{SU:DAT}$ = 1000 + 250 = 1250 ns：标准模式I²C总线规范)。

5. Cb：一条总线的线间总电容(单位：pF)

I^{2}C 总线时序



(6) 高阻抗控制时序

(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = AV_{DD0} = AV_{DD1} = AV_{DD2} = 4.0 至 5.5 V,
V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V, C_L = 50 pF)

参数	符号	条件	MIN.	MAX.	单位
振荡停止→定时器输出高阻抗	t_{CLM}	当时钟监视器工作时		65	μs
输入到 TOBnOFF →定时器输出高阻抗	t_{HTQn}			300	ns
输入到 TOA2OFF →定时器输出高阻抗	t_{HTP2}			300	ns
输入到 ANI00/ANI05 →定时器输出高阻抗	t_{ANI0}			10	μs
输入到 ANI10/ANI15 至 ANI12/ANI17 →定时器输出高阻抗	t_{ANI1}			10	μs

备注　　n = 0, 1

28.1.9 A/D 转换器 0 和 1 的特性

(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = AV_{DD0} = AV_{DD1} = AV_{DD2} = AV_{REFP0} = AV_{REFP1} = 4.0 至 5.5 V, V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V, C_L = 50 pF)

参数	符号	条件	MIN.	TYP.	MAX.	单位
分辨率			12	12	12	位
总体误差[注 1]					±10	LSB
转换时间	t_{CONV}	f_{AD01} = 16 MHz，ADAnCTC 寄存器 = 0BH 或 0CH	2.0			μs
		f_{AD01} = 12 MHz，ADAnCTC 寄存器 = 00H			7.42	μs
零标度误差[注 1]					±10	LSB
满量程误差[注 1]					±10	LSB
积分线性误差[注 1]					±4	LSB
微分线性误差[注 1]					±2.5	LSB
模拟参考电压	AV_{DD}		4.0		5.5	V
模拟输入电压	V_{IAN}		AV_{SS}		AV_{DD}	V
AV_{DD} 供电电流[注 2]	AI_{DD}	工作		4.5	7.5	mA
	AI_{DDS}	在STOP模式下[注 3]		3.5	17.5	μs

注　**1.** 不包括量化误差 (±0.5 LSB)
2. 该值仅对一路 A/D 转换器 (A/D 转换器 0 或 1)
3. 在设置 STOP 模式之前，停止 A/D 转换器 0 和 1 的操作(ADnSCM.ADnCE 位 = 0) 。

备注　**1.** LSB：最低有效位
2. f_{AD01}：A/D 转换器 0 和 1 的基准时钟
3. n = 0, 1

28.1.10　A/D 转换器 2 的特性

(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = AV_{DD0} = AV_{DD1} = AV_{DD2} = 4.0 至 5.5 V,
V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V, C_L = 50 pF)

参数	符号	条件	MIN.	TYP.	MAX.	单位
分辨率			10	10	10	位
总体误差注 1					±4.0	LSB
转换时间	t_{CONV}		3.88		10	μs
零标度误差注 1					±4.0	LSB
满量程误差注 1					±4.0	LSB
积分线性误差注 1					±4.0	LSB
微分线性误差注 1					±2.0	LSB
模拟参考电压	AV_{DD}		4.0		5.5	V
模拟输入电压	V_{IAN}		AV_{SS}		AV_{DD}	V
AV_{DD} 供电电流	AI_{DD}	在工作期间		3.5	7	mA
	AI_{DDS}	在STOP模式下注 2		1	10	μA

注　**1.** 不包括量化误差 (±0.5 LSB).
　　2. 在设置 STOP 模式之前，停止 A/D 转换器 2 的操作(AD2M0.AD2CE 位 = 0) 。

备注　LSB：最低有效位

28.1.11 运算放大器特性

(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = AV_{DD0} = AV_{DD1} = AV_{DD2} = 4.0 至 5.5 V, V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V, C_L = 50 pF)

参数	符号	条件		MIN.	TYP.	MAX.	单位
输入偏置电压	V_{IO}				± 9.0		mV
输入电压范围	V_I	增益 = 2.500		0.04AV_{DD}		0.36AV_{DD}	V
		增益 = 5.000		0.02AV_{DD}		0.18AV_{DD}	V
		增益 = 10.00		0.01AV_{DD}		0.085AV_{DD}	V
转换速率[注 1]	S_R			10	15		V/μs
增益误差		**注 2**	增益 = 2.500 至 4.444		±1.0	±1.3	%
			增益 = 5.000 至 6.667		±1.0	±1.5	%
			增益 = 8.000, 10.00		±1.0	±1.7	%
		注 3	增益 = 2.500 至 4.444		±1.0	±2.0	%
			增益 = 5.000 至 6.667		±1.0	±2.1	%
			增益 = 8.000, 10.00		±1.0	±2.2	%
工作电流[注 4]	I_{OPDD}	在工作期间			1.8	2.6	mA
	A_{IDDS}	在STOP模式下[注 5]			1.0	10	μA

注
1. 输出电压从 10%至 90%的斜率特性。
2. AV_{DD0} = AV_{DD1} = 4.5 至 5.5 V
3. AV_{DD0} = AV_{DD1} = 4.0 至 5.5 V
4. 总共提供四路运算放大器。该值为每个运算放大器的工作电流。
5. 设置 STOP 模式之前，停止运算放大器的操作(OP0CTL0.OP0EN 位 = 0, OP1CTL0.OP12EN, OP11EN 和 OP10EN 位 s = 0)) 。

备注　运算放大器使用的供电电压是 AV_{DD0} 和 AV_{DD1}

28.1.12　比较器特性

(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = AV_{DD0} = AV_{DD1} = AV_{DD2} = 4.0 至 5.5 V, V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V, C_L = 50 pF)

参数	符号	条件	MIN.	TYP.	MAX.	单位
输入偏置电压	V_{IO}			±3.0		mV
输入电压范围	V_I		AV_{SS}		AV_{DD}	V
比较器参考电压 (全范围)	CREFnF		$0.02AV_{DD} + 0.1$		$0.92AV_{DD} – 0.1$	V
比较器参考电压 (低范围)	CREFnL		$0.02AV_{DD} + 0.1$		$0.5AV_{DD} – 0.1$	V
响应时间	t_{CR}	输入幅度 = 100 mV, 在上升沿注 1		1.0		μs
	t_{CF}	输入幅度 = 100 mV, 在下降沿注 2		1.0		μs
工作电流注 3	I_{CPDD}	在工作期间			250	μA
	A_{IDDS}	在STOP模式下注 4		2.0	20	nA

注　**1.** 当 ANIm 输入从比较器参考电压 – 100 mV 改变为比较器参考电压 + 100 mV 时，脉冲响应的特性。

2. 当 ANIm 输入从比较器参考电压 + 100 mV 改变为比较器参考电压 – 100 mV 时，脉冲响应的特性。

3. 总共提供四路比较器。该值为每个比较器的工作电流。

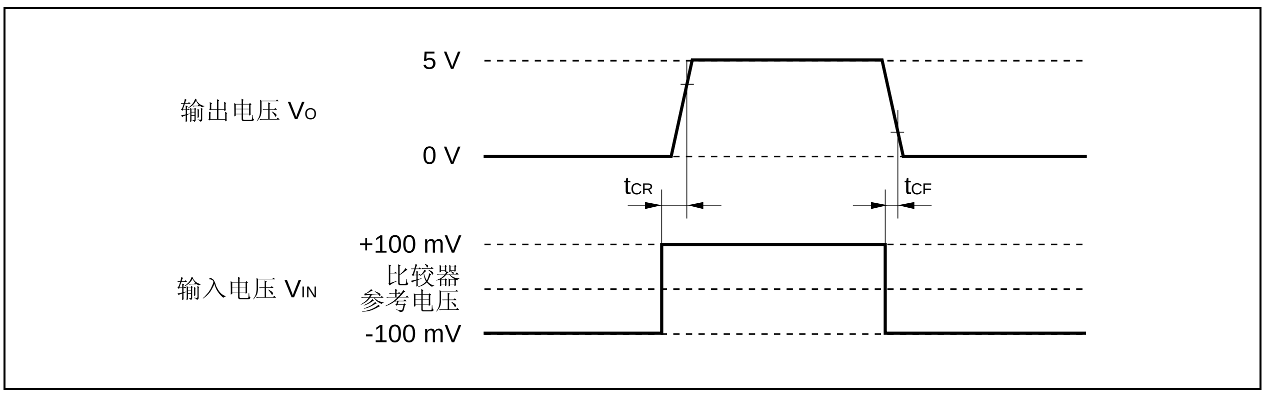
4. 设置 STOP 模式之前，停止比较器操作(CMPnCTL0 寄存器 = 00H)。

备注　**1.** 比较器使用的供电电压是 AV_{DD0} 和 AV_{DD1}

2. m = 05, 15 至 17

n = 0, 1

比较器特性

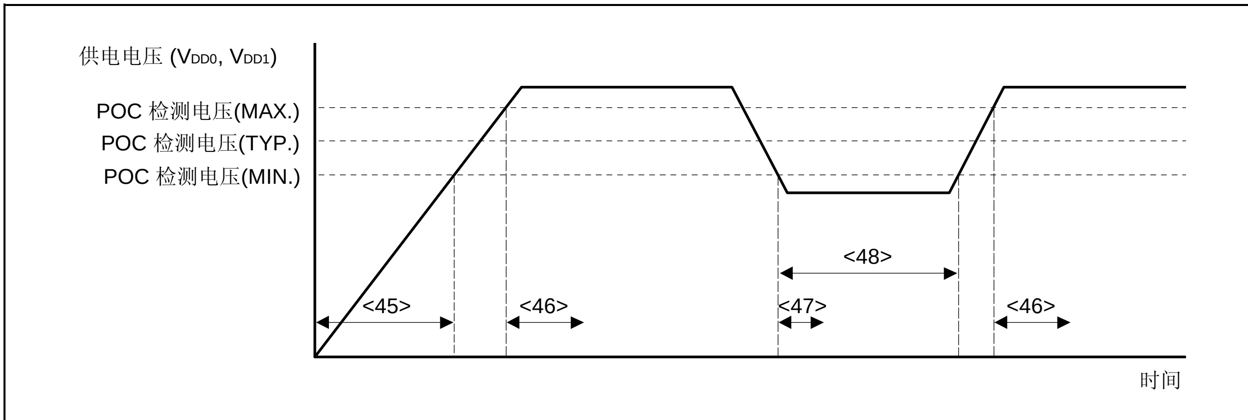


28.1.13　上电清零电路(POC)

(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = AV_{DD0} = AV_{DD1} = AV_{DD2} = 3.5 至 5.5 V,
V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V, C_L = 50 pF)

参数	符号		条件	MIN.	TYP.	MAX.	单位
POC 检测电压	V_{POC0}			3.5	3.7	3.9	V
供电电压上升时间	t_{PTH}	<45>	V_{DD0}, V_{DD1} = 0 至 3.5 V	2.5 μs		1.8 s	
响应时间 1[注1]	t_{PTHD}	<46>	上电时当 V_{DD0} 和 V_{DD1} 达到 3.9 V 之后			3.0	ms
响应时间 2[注2]	t_{PD}	<47>	下电时当 V_{DD0} 和 V_{DD1} 下降到 3.5 V 之后			1.0	ms
V_{DD0}, V_{DD1} 的最小宽度	t_{PW}	<48>		0.2			ms

注　**1.** 检测到 POC 检测电压之后，释放复位信号(POCRES)所需的时间。
　　2. 检测到 POC 检测电压之后，输出复位信号(POCRES)所需的时间。

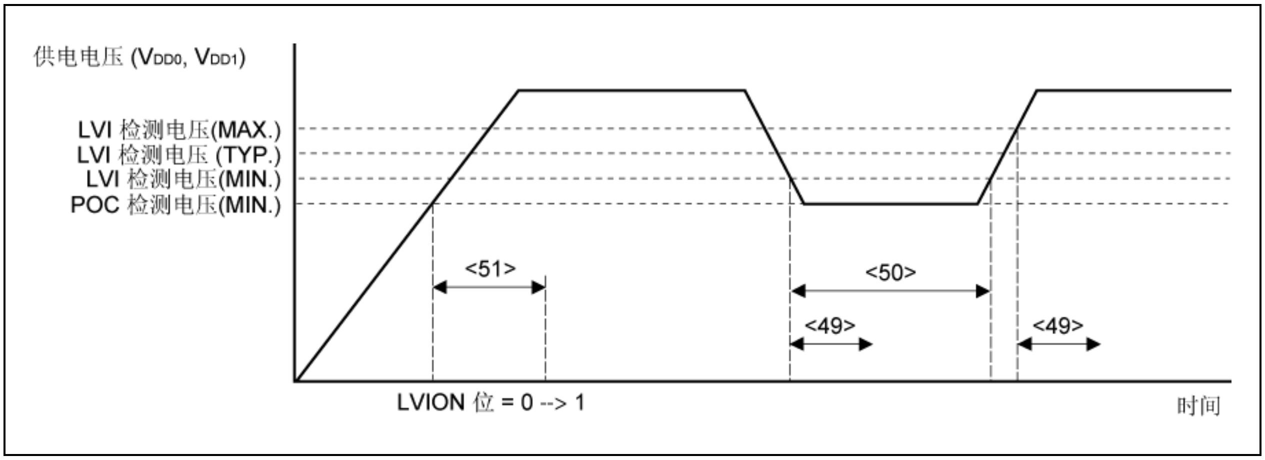


28.1.14　低电压检测器 (LVI)

(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = AV_{DD0} = AV_{DD1} = AV_{DD2} = 3.5 至 5.5 V, V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V, C_L = 50 pF)

参数	符号		条件	MIN.	TYP.	MAX.	单位
LVI 检测电压	V_{LVI0}		LVIS.LVIS0 位 = 0	4.2	4.4	4.6	V
	V_{LVI1}		LVIS.LVIS0 位 = 1	4.0	4.2	4.4	V
响应时间 1[注]	t_{LD}	<49>	当 V_{DD0} 和 V_{DD1} 达到 V_{LVI0}/V_{LVI1} (MAX.) 或下降到 V_{LVI0}/V_{LVI1} (MIN.)之后		0.2	2.0	ms
V_{DD0}, V_{DD1} 的最小宽度	t_{LW}	<50>		0.2			ms
参考电压稳定等待时间	t_{LWAIT}	<51>	当 V_{DD0} 和 V_{DD1} 达到 POC 检测电压(MIN.)之后，且 LVIM.LVION 位从 0 改变为 1		0.1		ms

注　　检测到 LVI 检测电压之后，输出一个中断请求信号(INTLVIL，INTLVIH)或内部复位信号(LVIRES)所需的时间。



28.1.15　Flash 存储器编程特性

(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = AV_{DD0} = AV_{DD1} = AV_{DD2} = 3.5 至 5.5 V,

V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V, C_L = 50 pF)

参数	符号	条件	MIN.	TYP.	MAX.	单位
重写次数	C_{ERWR}	注	100			次

注　重写如下。

三次重写的示例：出厂产品→E→P→E→P→E→P (P：写入，E：擦除)

28.2 V850E/IG3

28.2.1 最大额定值

(T_A = 25°C)

参数	符号	条件		额定值范围	单位
供电电压	V_{DD}	$V_{DDa} = EV_{DDb} = AV_{DDk}$		–0.5 至+6.5	V
	V_{SS}	$V_{SSa} = EV_{SSb} = AV_{SSk}$		–0.5 至+0.5	V
	EV_{DD}	$V_{DDa} = EV_{DDb} = AV_{DDk}$		–0.5 至+6.5	V
	EV_{SS}	$V_{SSa} = EV_{SSb} = AV_{SSk}$		–0.5 至+0.5	V
	AV_{DD}	$V_{DDa} = EV_{DDb} = AV_{DDk}$		–0.5 至+6.5	V
	AV_{SS}	$V_{SSa} = EV_{SSb} = AV_{SSk}$		–0.5 至+0.5	V
输入电压	V_{I1}	**注 1**		–0.5 至 EV_{DD} + 0.5[注 2]	V
	V_{I2}	X1，X2		–0.5 至 V_{RO} + 0.35	V
输出电流，低	I_{OL}	所有引脚	每个引脚	4	mA
			全部引脚总和	63	mA
输出电流，高	I_{OH}	所有引脚	每个引脚	–4	mA
			全部引脚总和	–63	mA
模拟输入电压	V_{IAN}	P70/ANI20 至 P77/ANI27， ANI00 至 ANI05, ANI10 至 ANI17		–0.5 至 AV_{DD} + 0.5[注 2]	V
模拟参考输入电压	V_{IREF}	AV_{REFP0}，AV_{REFP1}		–0.5 至 AV_{DD} + 0.5[注 2]	V
比较器参考输入电压	V_{CREF}	CREF0L, CREF1L, CREF0F, CREF1F		–0.5 至 AV_{DD} + 0.5[注 2]	V
工作环境温度	T_A	正常操作模式		–40 至+85	°C
		Flash 存储器编程模式		–40 至+85	°C
工作环境温度	T_{stg}			–40 至+125	°C

注　**1.** P00 至 P07, P10 至 P17，P20 至 P27，P30 至 P37，P40 至 P47, PDL0 至 PDL15, $\overline{\text{RESET}}$, FLMD0, $\overline{\text{DRST}}$

　　2. 请确保各个供电电压不超过最大额定值 (MAX.值)。

注意事项 **1.** 不要将 **IC** 产品的输出引脚（或处于输出状态下的 **I/O** 引脚）直接连接到其它输出引脚（包括处于输出状态下的 **I/O** 引脚），或直接连接至 **V_{DD}**、**EV_{DD}** 或 **GND**。如果输出引脚可以设置为高阻抗状态，并且设计外部电路的输出时序时已经考虑避免输出冲突，芯片和外部电路之间的输出引脚才可以直接相连。

2. 任何参数，哪怕瞬间超过最大额定值，都可能使产品性能受损。就是说，产品工作在最大额定值时处于物理损坏的边缘，因此，产品必须在不超出最大额定值的条件下使用。

DC 和 **AC** 特性的额定值和工作条件代表在正常工作期间的性能保障范围。

备注　a = 0, 1

b = 0 至 2

k = 0 至 2

28.2.2 电容

(T_A = 25°C, V_{DD0} = V_{SS0} = V_{DD1} = V_{SS1} = EV_{DD0} = EV_{SS0} = EV_{DD1} = EV_{SS1} = EV_{DD2} = EV_{SS2} = AV_{DD0} = AV_{SS0} = AV_{DD1} = AV_{SS1} = AV_{DD2} = AV_{SS2} = 0 V)

参数	符号	条件		MIN.	TYP.	MAX.	单位
输入电容	C_I	fc = 1 MHz 未测量的引脚返回至 0 V	**注 1**			15	pF
I/O 电容	C_{IO}		**注 2**			15	pF
输出电容	C_O		**注 3**			15	pF

注 **1.** ANI00 至 ANI05, ANI10 至 ANI17, $\overline{\text{RESET}}$

2. P00 至 P07, P10 至 P17，P20 至 P27，P30 至 P37，P40 至 P47, P70 至 P77, PDL0 至 PDL15

3. DDO

注意事项 **1.** 不包括 **FLMD0, $\overline{\textbf{DRST}}$, X1** 和 **X2** 引脚**.**

2. 除了输入电容，**ANI00** 至 **ANI05, ANI10** 至 **ANI17** 和 **ANI20** 至 **ANI27** 引脚在采样时还要加上采样电容。

28.2.3 工作 条件

(T_A = –40 至+85°C, V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = EV_{SS2} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V)

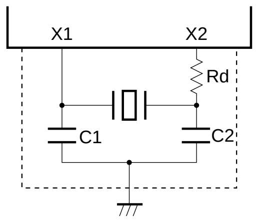
参数	符号	条件		MIN.	TYP.	MAX.	单位
系统时钟频率	f_{XX}	PLL 模式		32		64	MHz
		时钟直通模式		4		8	MHz
CPU 时钟频率	f_{CPU}	PLL 模式		4		64	MHz
		时钟直通模式		0.5		8	MHz
V_{DD}, EV_{DD} 电压	V_{DD}, EV_{DD}	**注 1**	未使用外部总线时	3.5		5.5	V
			使用外部总线时**注 2**	4.0		5.5	V
AV_{DD} 电压	AV_{DD}	当 A/D 转换器 0 至 2 工作时		4.0		5.5	V
		当 A/D 转换器 0 至 2 不工作时		3.5		5.5	V

注 **1.** V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = EV_{DD2} = AV_{DD0} = AV_{DD1} = AV_{DD2}

<R> **2.** 仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A

28.2.4 时钟振荡器特性

(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = EV_{DD2} = AV_{DD0} = AV_{DD1} = AV_{DD2} = 3.5 至 5.5 V，V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = EV_{SS2} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V)

谐振器	推荐电路	参数	条件	MIN.	TYP.	MAX.	单位
陶瓷/晶体谐振器		振荡频率(f_X)		4		8	MHz
		振荡稳定时间	复位释放后		$2^{14}/f_X$		ms
			STOP 模式解除后		注		ms

注　　该值根据振荡稳定时间选择寄存器 (OSTS)的设置而不同。

注意事项 **1.** 振荡器的连接要尽量靠近 **X1** 和 **X2** 引脚。

2. 上图中虚线框内的连线应避免和其它的信号线互相交叉。

3. 谐振器的选择和振荡器常数，客户需要自行评估振荡性能，或向振荡器生产商申请对实现电路的评估。

4. 禁止向 **V850E/IG3** 输入外部时钟。

<R> **(i) Murata Mfg. Co., Ltd.**：陶瓷谐振器 **(T_A = –40 至+85°C) (除μPD70F3454F1-DA9-A** 之外的产品**)**

类型	零件登记号	振荡频率 f_X (MHz)	推荐电路常数			振荡电压范围	
			C1	C2	Rd (kΩ)	MIN. (V)	MAX. (V)
引线式	CSTLS4M00G56-B0	4	片上 (47 pF)	片上 (47 pF)	0	3.5	5.5
	CSTLS5M00G56-B0	5	片上 (47 pF)	片上 (47 pF)	0	3.5	5.5
	CSTLS8M00G56-B0	8	片上 (47 pF)	片上 (47 pF)	0	3.5	5.5
表面贴装	CSTCR4M00G55-R0	4	片上 (39 pF)	片上 (39 pF)	0	3.5	5.5
	CSTCR5M00G55-R0	5	片上 (39 pF)	片上 (39 pF)	0	3.5	5.5
	CSTCE8M00G55-R0	8	片上 (33 pF)	片上 (33 pF)	0	3.5	5.5

注意事项 该振荡常数是振荡器生产商在特定环境下进行评估得出的参考值。如果在实际应用中必须对振荡器特性进行优化，请向振荡器生产商申请对实现电路的评估。

振荡电压和振荡频率仅表示振荡器特性。使用 **V850E/IG3** 时内部工作条件不要超出 **AC** 特性和 **DC** 特性的额定值。

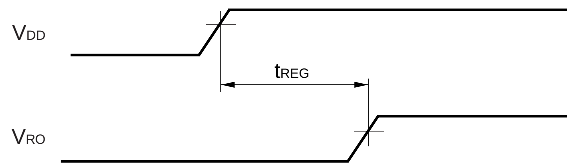
28.2.5 稳压器特性

(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = EV_{DD2} = 3.5 至 5.5 V, V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = EV_{SS2} = 0 V)

参数	符号	条件	MIN.	TYP.	MAX.	单位
输入电压	V_{DD}		3.5		5.5	V
输出电压	V_{RO}			1.5		V
输出电压 稳定时间	t_{REG}	稳定电容 C = 4.7 μF 注 1			1 注 2	ms

注 **1.** 在 REGC0 引脚和 V_{SS0} 引脚、REGC1 引脚和 V_{SS1} 引脚之间各连接一个稳定电容。

2. 在 t_{REG} 时期，输出内部复位信号，直到上电清零电路(POC) 的输出电压稳定。



28.2.6 DC 特性

(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = EV_{DD2} = AV_{DD0} = AV_{DD1} = AV_{DD2} = 3.5 至 5.5 V, V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = EV_{SS2} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V) (1/2)

参数	符号	条件			MIN.	TYP.	MAX.	单位
输入电压，高	V_{IH1}	注 **1**			0.7EV_{DD}		EV_{DD}	V
	V_{IH2}	注 **2**			0.8EV_{DD}		EV_{DD}	V
	V_{IH3}	注 **3**			2.2		EV_{DD}	V
	V_{IH4}	注 **4**			0.7AV_{DD}		AV_{DD}	V
输入电压，低	V_{IL1}	注 **1**			EV_{SS}		0.3EV_{DD}	V
	V_{IL2}	注 **2**			EV_{SS}		0.2EV_{DD}	V
	V_{IL3}	注 **3**			EV_{SS}		0.8	V
	V_{IL4}	注 **4**			AV_{SS}		0.3AV_{DD}	V
输入漏电流，高	I_{LIH1}	V_I = 注 **5**，注 **6**	除 X1 之外				5	μA
	I_{LIH2}		X1				20	μA
输入漏电流，低	I_{LIL1}	V_I = 0 V	除 X1 之外				–5	μA
	I_{LIL2}		X1				–20	μA
输出漏电流，高	I_{LOH}	V_O = 注 **5**					5	μA
输出漏电流，低	I_{LOL}	V_O = 0 V					–5	μA
输出电压，高	V_{OH1}	注 **7**	I_{OH} = –1.0 mA	引脚的总电流 = –57 mA	EV_{DD} – 1.0			V
输出电压，低	V_{OL1}	注 **7**	I_{OL} = 1.0 mA	引脚的总电流 = 57 mA			0.4	V
上拉电阻	R_{L1}				10	30	100	kΩ
下拉电阻注 8	R_{L2}				10	30	100	kΩ

注　**1.** P33, P36, P41 和 PDL0 至 PDL15 引脚

2. P00 至 P07, P10 至 P17，P20 至 P27，P30 至 P32, P34, P35, P37，P40, P42 至 P47, $\overline{\text{RESET}}$ 和 FLMD0 引脚

3. $\overline{\text{DRST}}$, DDI, DCK 和 DMS 引脚

4. P70 至 P77 引脚

5. AV_{DD0} = AV_{DD1} = AV_{DD2} = EV_{DD0} = EV_{DD1} = EV_{DD2}

6. 除 $\overline{\text{DRST}}$ 引脚之外

7. P00 至 P07, P10 至 P17，P20 至 P27，P30 至 P37，P40 至 P47, PDL0 至 PDL15 和 DDO 引脚

8. 仅限 $\overline{\text{DRST}}$ 引脚

备注　复用功能引脚的特性和端口引脚的特性相同。

(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = EV_{DD2} = AV_{DD0} = AV_{DD1} = AV_{DD2} = 3.5 至 5.5 V,
V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = EV_{SS2} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V) (2/2)

参数	符号	条件		MIN.	TYP.注 1	MAX.	单位
V_{DD} 供电电流注 2	I_{DD1}	f_{XX} = 64 MHz	正常工作		64	93	mA
	I_{DD2}		HALT 模式		42	60	mA
	I_{DD3}		IDLE 模式		5	10	mA
	I_{DD4}	STOP 模式			40	800	μA

注　**1.** TYP.值是 V_{DD0} = V_{DD1} = 5.0 V 和 T_A = 25°C 情况下的参考值。

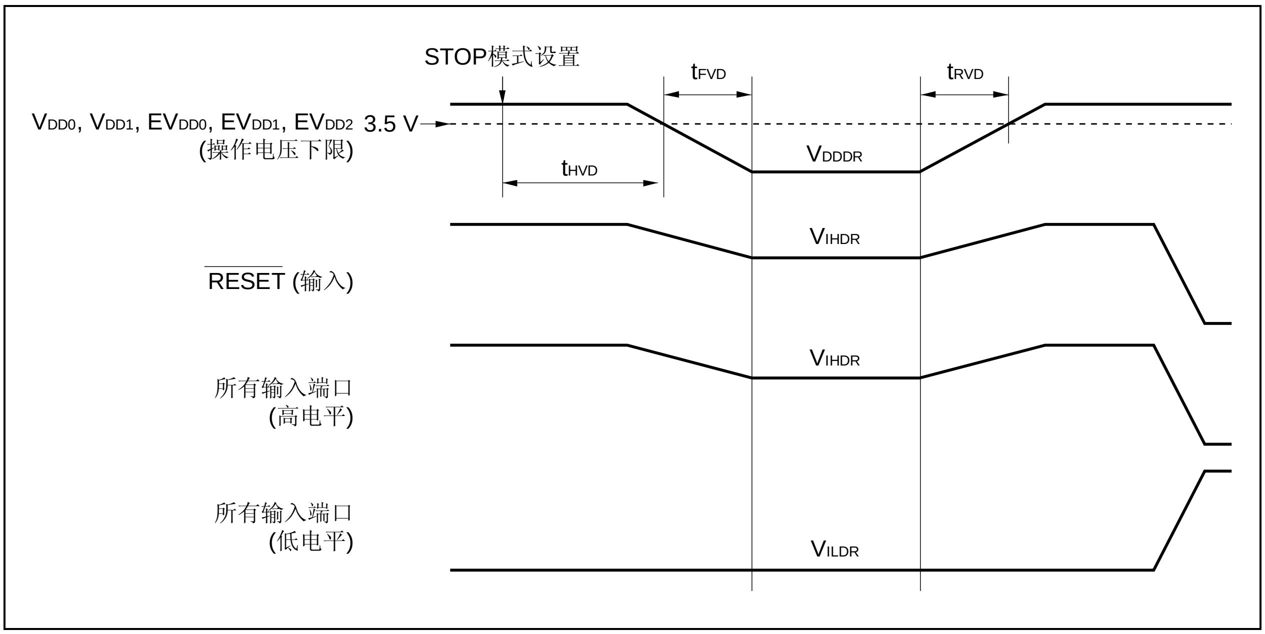
2. 不包括 EV_{DD} 系统(输出缓冲和上拉电阻)消耗的电流，也不包括 A/D 转换器 0 至 2，运算放大器和比较器的工作电流。

28.2.7 数据保持特性

STOP 模式 (T_A = –40 至+85°C, V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = EV_{SS2} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V)

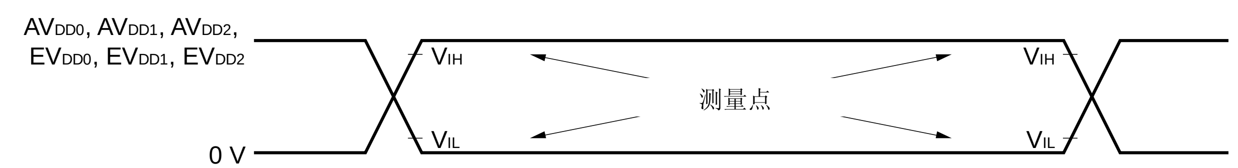
参数	符号	条件	MIN.	TYP.	MAX.	单位
数据保持电压	V_{DDDR}	STOP 模式	注		5.5	V
数据保持电流	I_{DDDR}	V_{DD0} = V_{DD1} = V_{DDDR}		40	800	μA
供电电压上升时间	t_{RVD}		1			μs
供电电压下降时间	t_{FVD}		1			μs
供电电压保持时间 (从设置 STOP 模式)	t_{HVD}		0			ms
数据保持输入电压，高	V_{IHDR}	所有输入端口	$0.9V_{DDDR}$		V_{DDDR}	V
数据保持输入电压，低	V_{ILDR}	所有输入端口	EV_{SS}		$0.1V_{DDDR}$	V

注　不使用低电压检测器 (LVI)复位模式 (LVIM.LVIMD 位 = 0)时：　POC 检测电压 (V_{POC0})
　　使用低电压检测器 (LVI)复位模式(LVIM.LVIMD 位 = 1)时：　LVI 检测电压 (V_{LVI0}/V_{LVI1})

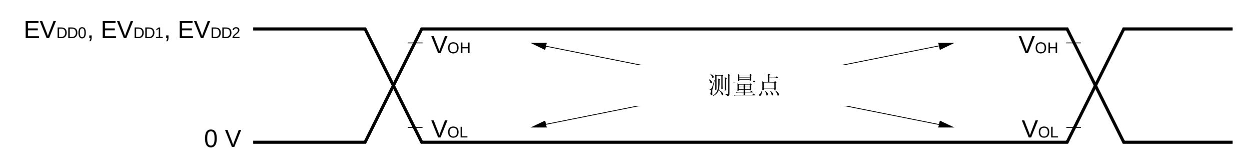


28.2.8 AC 特性

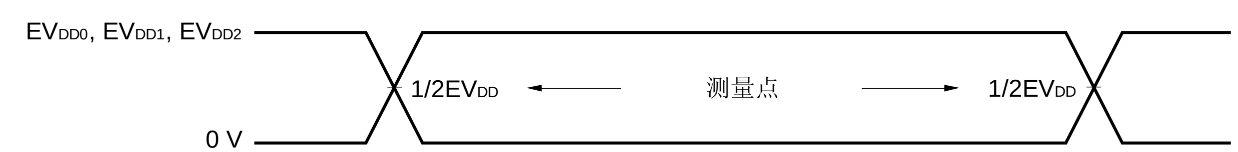
<R> **AC** 测试输入测量点**(**外部总线 **(**仅限μ**PD70F3454GC-8EA-A** 和 **70F3454F1-DA9-A)**，除 **CSIB0** 至 **CSIB2** 之外的引脚**)**



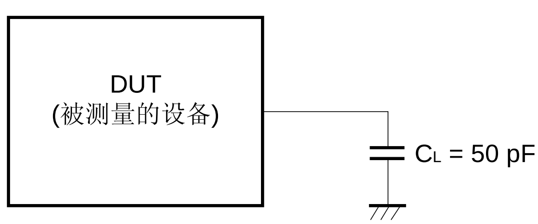
<R> **AC** 测试输出测量点**(**外部总线 **(**仅限μ**PD70F3454GC-8EA-A** 和 **70F3454F1-DA9-A)**，除 **CSIB0** 至 **CSIB2** 之外的引脚**)**



<R> **AC test I/O** 测量点 **(**外部总线 **(**仅限μ**PD70F3454GC-8EA-A** 和 **70F3454F1-DA9-A)**，**CSIB0** 至 **CSIB2** 引脚**)**



负载条件

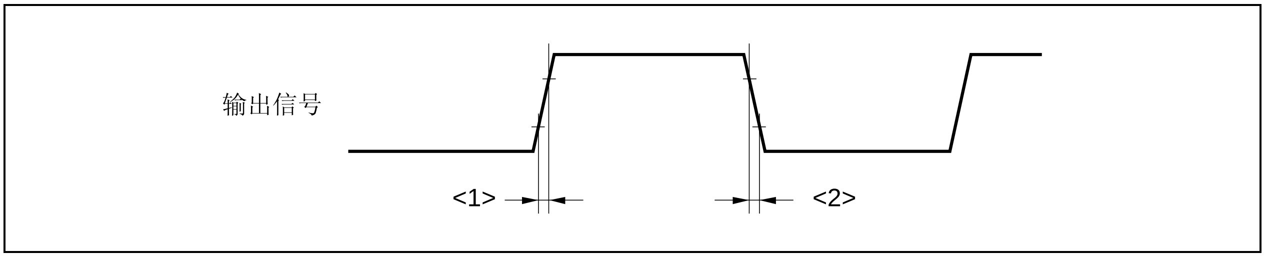


注意事项 如果由于电路配置使得负载电容超过 **50 pF**，可插入一个缓冲器或其它手段使设备的负载电容等于或小于 **50 pF**。

(1) 输出信号时序

(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = EV_{DD2} = AV_{DD0} = AV_{DD1} = AV_{DD2} = 4.0 至 5.5 V, V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = EV_{SS2} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V, C_L = 50 pF)

参数	符号		条件	MIN.	MAX.	单位
输出上升时间	t_{OR}	<1>	P07，PDL0 至 PDL15，DDO		8	ns
			其它情况		15	ns
输出下降时间	t_{OF}	<2>	P07，PDL0 至 PDL15，DDO		8	ns
			其它情况		15	ns



(2) 复位，外部中断时序

(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = EV_{DD2} = AV_{DD0} = AV_{DD1} = AV_{DD2} = 4.0 至 5.5 V, V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = EV_{SS2} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V, C_L = 50 pF)

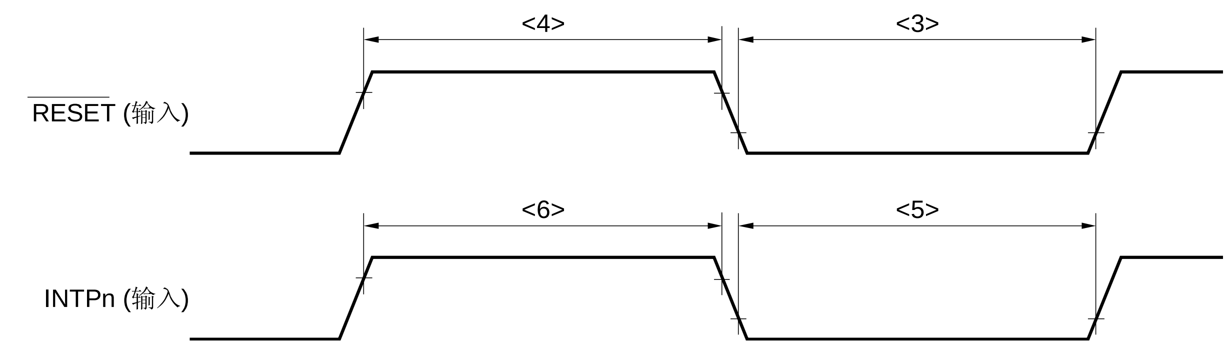
参数	符号		条件	MIN.	MAX.	单位
$\overline{\text{RESET}}$ 低电平宽度	t_{WRSL}	<3>	上电，STOP 模式被解除	500 + T_{os}		ns
			其它情况	500		ns
$\overline{\text{RESET}}$ 高电平宽度	t_{WRSH}	<4>		500		ns
INTPn 低电平宽度	t_{WITL}	<5>	n = 00 至 13, 17, 18 (模拟噪声消除)	500		ns
			n = 14 至 16 (数字噪声消除)	4T_{smp}		ns
INTPn 高电平宽度	t_{WITH}	<6>	n = 00 至 13, 17, 18 (模拟噪声消除)	500		ns
			n = 14 至 16 (数字噪声消除)	4T_{smp}		ns

备注　**1.** T_{os}：振荡稳定时间

T_{smp}：噪声消除采样时钟周期(由 INTNFCn 寄存器设置)

2. 复位释放后，当振荡频率(fx) = 8 MHz 时，内部保障 1 ms 振荡稳定时间。因此，振荡稳定时间是(T_{os} + 1) ms。STOP 模式解除之后，设置到 OSTS 寄存器振荡稳定时间值的一半由内部保障。因此，如果可以通过 OSTS 寄存器的设置保障足够稳定时间，可以接受 T_{os} = 0 ns。

复位/中断

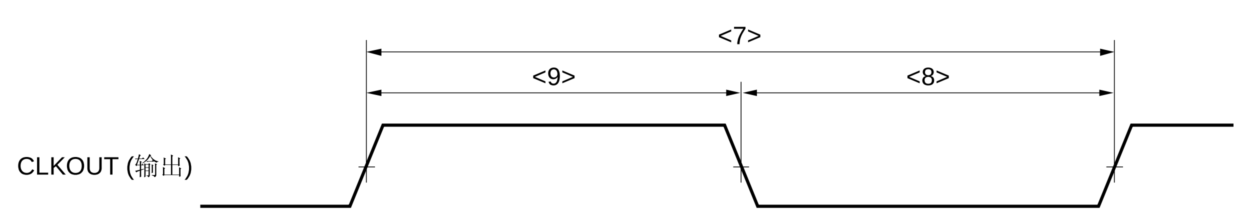


备注　　n = 00 至 13, 17, 18

<R> **(3) CLKOUT 输出时序 (仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A)**

(T_A = –40 至+85 °C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = EV_{DD2} = AV_{DD0} = AV_{DD1} = AV_{DD2} = 4.0 至 5.5 V, V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = EV_{SS2} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V, C_L = 50 pF)

参数	符号		条件	MIN.	MAX.	单位
输出周期	t_{CYK}	<7>		31.25 ns	8 μs	
低电平宽度	t_{WKH}	<8>		t_{CYK}/2 – 6.2		ns
高电平宽度	t_{WKL}	<9>		t_{CYK}/2 – 6.2		ns



<R> **(4) 总线时序 (仅限μPD70F3454GC-8EA-A 和 70F3454F1-DA9-A)**

(a) 读取周期 (CLKOUT 异步)

(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = EV_{DD2} = AV_{DD0} = AV_{DD1} = AV_{DD2} = 4.0 至 5.5 V, V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = EV_{SS2} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V, C_L = 50 pF)

参数	符号		条件	MIN.	MAX.	单位
自地址到 ASTB↓的延迟时间	t_{DAST2}	<10>		$(0.5 + w_{AS})$ T – 20		ns
ASTB 高电平宽度	t_{WSTH}	<11>		$(1 + w_{AS} + i)$ T – 17		ns
自 ASTB↓的地址保持时间	t_{HSTA}	<12>		$(0.5 + w_{AH})$ T – 19		ns
自 $\overline{\text{RD}}$↑的地址保持时间	t_{HRDA2}	<13>		(1 + i) T – 29		ns
自地址到 $\overline{\text{RD}}$↓的延迟时间	t_{DARD2}	<14>		$(1 + w_{AS} + w_{AH})$ T – 36		ns
从 $\overline{\text{RD}}$↓ 到地址悬空的延迟时间	t_{FRDA}	<15>			16	ns
自地址的数据输入建立时间	t_{DAID2}	<16>			$(2 + w_D + w + w_{AS} + w_{AH})$ T – 37	ns
自 ASTB↓的数据输入建立时间	t_{DSTID}	<17>			$(1.5 + w_D + w + w_{AH})$ T – 37	ns
自 $\overline{\text{RD}}$↓的数据输入建立时间	t_{DRDID2}	<18>			$(1 + w_D + w)$ T – 37	ns
从 ASTB↓到 $\overline{\text{RD}}$↓的延迟时间	t_{DSTRD3}	<19>		$(0.5 + w_{AH})$ T – 16		ns
数据 输入保持时间 (自 $\overline{\text{RD}}$↑)	t_{HRDID2}	<20>		2		ns
从 $\overline{\text{RD}}$↑到总线 输出的延迟时间	t_{DRDOD2}	<21>		(1 + i) T – 19		ns
从 $\overline{\text{RD}}$↑到 ASTB↑的延迟时间	t_{DRDST}	<22>		0.5T – 16		ns
$\overline{\text{RD}}$ 低电平宽度	t_{WRDL2}	<23>		$(1 + w_D + w)$ T – 20		ns
$\overline{\text{RD}}$ 高电平宽度	t_{WRDH2}	<24>		$(2 + i + w_{AS} + w_{AH})$ T – 20		ns
从 $\overline{\text{RD}}$↑到 $\overline{\text{WRn}}$ 的高电平保持时间	t_{HRDWR2}	<25>		$(2 + i + w_{AS} + w_{AH})$ T – 20		ns
$\overline{\text{WAIT}}$ 建立时间(至 地址)	t_{DAWT2}	<26>			$(1.5 + w_D + w + w_{AS} + w_{AH})$ T – 45	ns
$\overline{\text{WAIT}}$ 保持时间 (自 地址)	t_{HAWT2}	<27>		$(1.5 + w_D + w + w_{AS} + w_{AH})$ T – 1		ns
$\overline{\text{WAIT}}$ 建立时间(至 ASTB↓)	t_{DSTWT}	<28>			$(1 + w_D + w + w_{AH})$ T – 37	ns
$\overline{\text{WAIT}}$ 保持时间 (自 ASTB↓)	t_{HSTWT}	<29>		$(1 + w_D + w + w_{AH})$ T + 2		ns
$\overline{\text{WAIT}}$ 建立时间(至 $\overline{\text{RD}}$↓)	t_{DRDWT2}	<30>			$(0.5 + w_D + w)$ T – 37	ns
$\overline{\text{WAIT}}$ 保持时间 (自 $\overline{\text{RD}}$↓)	t_{HRDWT2}	<31>		$(0.5 + w_D + w)$ T + 2		ns

注意事项　1. T 的设置符合下列条件。

31.25 ns ≥ T

2. 请确保插入地址建立等待和地址保持等待。

备注　**1.** w_{AS}：AWC 寄存器设置的地址建立等待数量

w_{AH}：AWC 寄存器设置的地址保持等待数量

w_D：DWC0 寄存器设置的数据等待数量

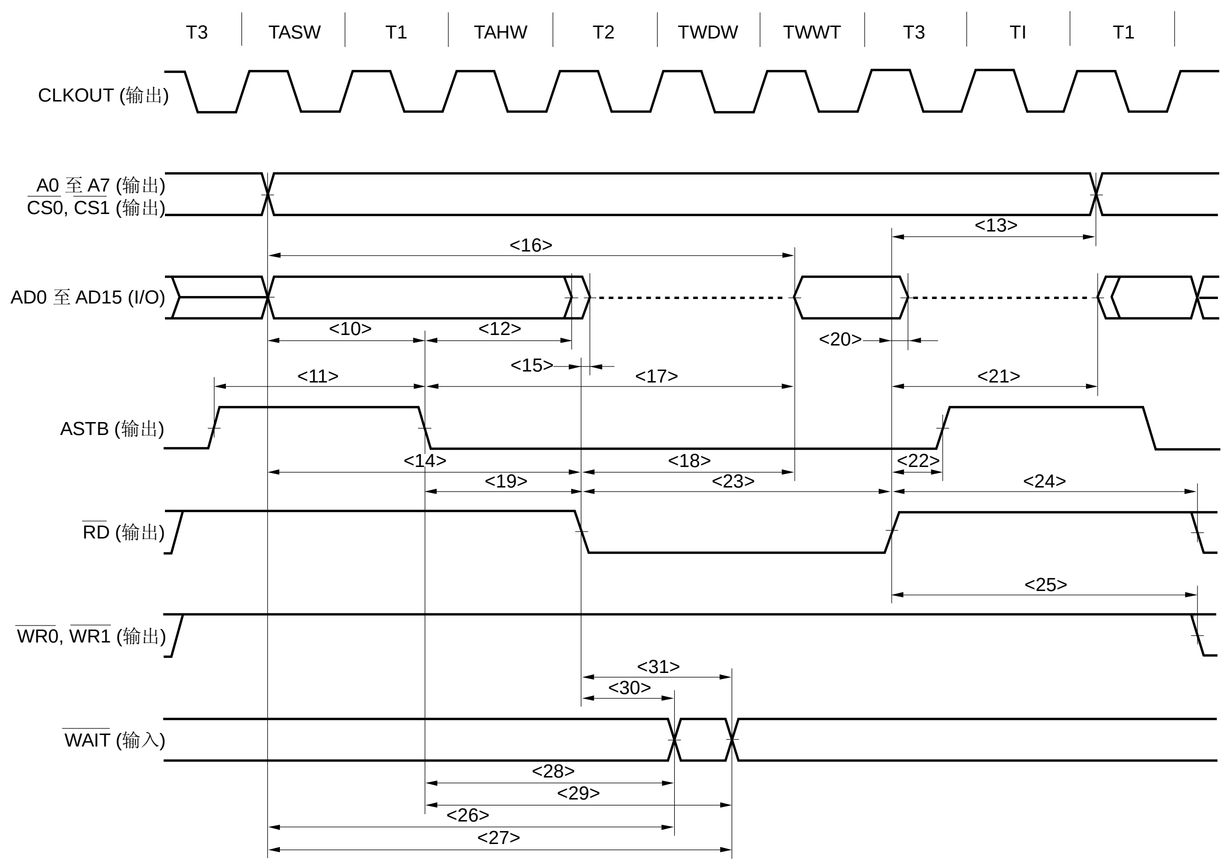
w：$\overline{\text{WAIT}}$ 引脚导致的外部等待数量

2. T = $1/f_{CPU}$ (f_{CPU}：CPU 工作时钟频率)

3. n = 0，1

4. i：空闲状态的数量

读取周期 **(CLKOUT 异步)**



备注 当地址设置等待的数量为 1，地址保持等待的数量为 1，数据等待的数量为 1，$\overline{\text{WAIT}}$ 引脚的等待数量为 1(在已决定的等待时期，当一个有效电平(低电平)的输入持续一个周期)，且空闲状态的数量为 1 时，上述时序图展示了时序，

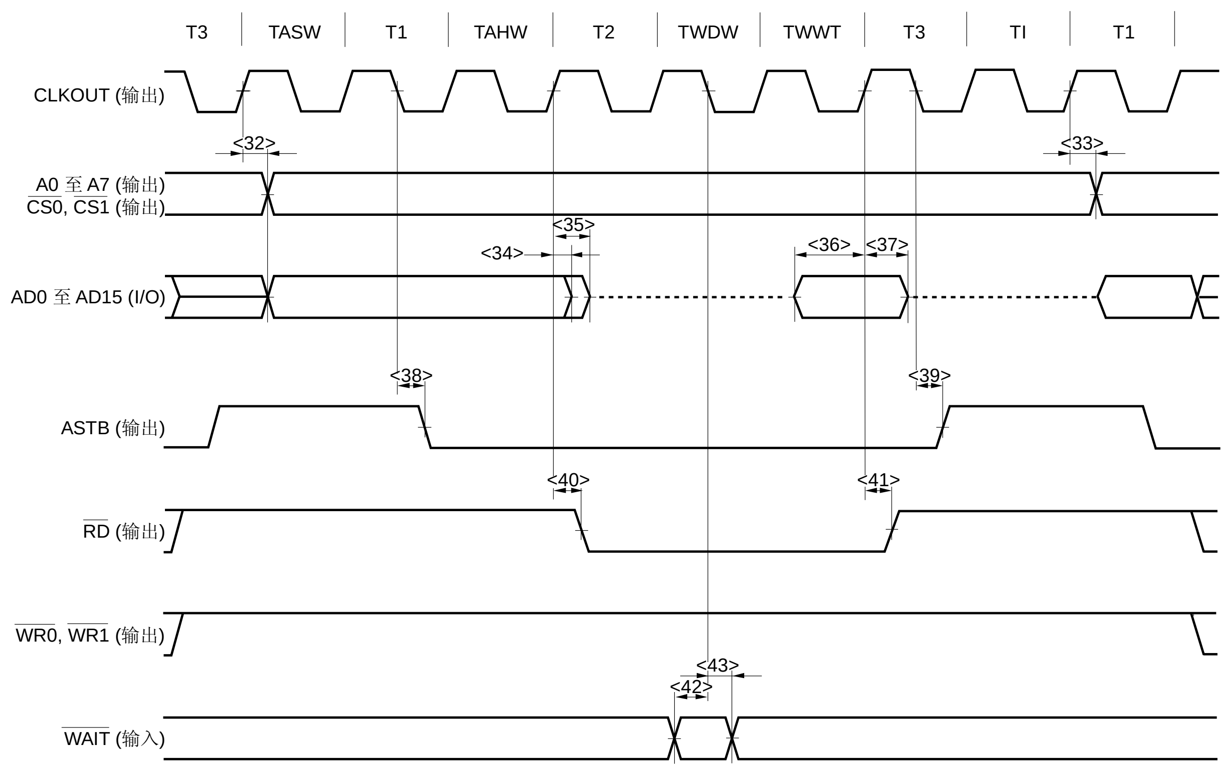
(b)　读取周期 (CLKOUT 同步)

(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = EV_{DD2} = AV_{DD0} = AV_{DD1} = AV_{DD2} = 4.0 至 5.5 V, V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = EV_{SS2} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V, C_L = 50 pF)

参数	符号		条件	MIN.	MAX.	单位
从 CLKOUT↑到地址的延迟时间	t_{DKA2}	<32>			22	ns
自 CLKOUT↑的地址保持时间	t_{HKA2}	<33>		–5		ns
自 CLKOUT↑的地址保持时间	t_{HKA3}	<34>		–8		ns
从 CLKOUT↑到地址悬空的延迟时间	t_{FKA}	<35>			15	ns
数据 输入建立时间 (至 CLKOUT↑)	t_{SIDK2}	<36>		30		ns
数据 输入保持时间 (自 CLKOUT↑)	t_{HKID2}	<37>		9		ns
从 CLKOUT↓到 ASTB↓的延迟时间	t_{DKST3}	<38>		–8	18	ns
从 CLKOUT↓到 ASTB↑的延迟时间	t_{DKST4}	<39>		–8	18	ns
从 CLKOUT↑到 $\overline{RD}$↓的延迟时间	t_{DKRD3}	<40>		–10	17	ns
从 CLKOUT↑到 $\overline{RD}$↑的延迟时间	t_{DKRD4}	<41>		–10	17	ns
$\overline{WAIT}$ 建立时间(至 CLKOUT↓)	t_{SWTK2}	<42>		30		ns
$\overline{WAIT}$ 保持时间 (自 CLKOUT↓)	t_{HKWT2}	<43>		9		ns

<R>　注意事项　请确保插入地址建立等待和地址保持等待。

读取周期 **(CLKOUT 同步)**



备注　当地址设置等待的数量为 1，地址保持等待的数量为 1，数据等待的数量为 1，$\overline{\text{WAIT}}$ 引脚的等待数量为 1(在已决定的等待时期，当一个有效电平(低电平)的输入持续一个周期)，且空闲状态的数量为 1 时，上述时序图展示了时序，

(c) 写入周期 **(CLKOUT** 异步**)**

(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = EV_{DD2} = AV_{DD0} = AV_{DD1} = AV_{DD2} = 4.0 至 5.5 V, V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = EV_{SS2} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V, C_L = 50 pF)

参数	符号		条件	MIN.	MAX.	单位
从地址到 ASTB↓ 的延迟时间	t_{DAST2}	<10>		(0.5 + w_{AS}) T – 20		ns
ASTB 高电平宽度	t_{WSTH}	<11>		(1 + w_{AS} + i) T – 16		ns
自 ASTB↓ 的地址保持时间	t_{HSTA}	<12>		(0.5 + w_{AH}) T – 19		ns
自 $\overline{\text{WRn}}$↑ 的地址保持时间	t_{HWRA2}	<44>		T – 19		ns
从地址到 $\overline{\text{WRn}}$↓的延迟时间	t_{DAWR2}	<45>		(1 + w_{AS} + w_{AH}) T – 36		ns
从 $\overline{\text{WRn}}$↓ 到数据输出的延迟时间	t_{DWROD3}	<46>			15	ns
从 ASTB↓到 $\overline{\text{WRn}}$↓的延迟时间	t_{DSTWR3}	<47>		(0.5 + w_{AH}) T – 16		ns
从数据输出到 $\overline{\text{WRn}}$↑的延迟时间	t_{DODWR2}	<48>		(1 + w_D + w) T – 25		ns
自 $\overline{\text{WRn}}$↑的数据输出保持时间	t_{HWROD2}	<49>		T – 19		ns
从 $\overline{\text{WRn}}$↑到 ASTB↑的延迟时间	t_{DWRST}	<50>		0.5T – 16		ns
$\overline{\text{WRn}}$ 低电平宽度	t_{WWRL2}	<51>		(1 + w_D + w) T – 20		ns
$\overline{\text{WRn}}$ 高电平宽度	t_{WWRH2}	<52>		(2 + w_{AS} + w_{AH}) T – 20		ns
从 $\overline{\text{WRn}}$↑到 $\overline{\text{RD}}$ 的高电平保持时间	t_{HWRRD2}	<53>		(2 + w_{AS} + w_{AH}) T – 20		ns
$\overline{\text{WAIT}}$ 建立时间(至 地址)	t_{DAWT2}	<26>			(1.5 + w_D + w + w_{AS} + w_{AH}) T – 45	ns
$\overline{\text{WAIT}}$ 保持时间 (自 地址)	t_{HAWT2}	<27>		(1.5 + w_D + w + w_{AS} + w_{AH}) T – 1		ns
$\overline{\text{WAIT}}$ 建立时间(至 ASTB↓)	t_{DSTWT}	<28>			(1 + w_D + w + w_{AH}) T – 37	ns
$\overline{\text{WAIT}}$ 保持时间 (自 ASTB↓)	t_{HSTWT}	<29>		(1 + w_D + w + w_{AH}) T + 2		ns
$\overline{\text{WAIT}}$ 建立时间(至 $\overline{\text{WRn}}$↓)	t_{DWRWT2}	<54>			(0.5 + w_D + w) T – 37	ns
$\overline{\text{WAIT}}$ 保持时间 (自 $\overline{\text{WRn}}$↓)	t_{HWRWT2}	<55>		(0.5 + w_D + w) T + 2		ns

注意事项　**1. T** 的设置符合下列条件。

31.25 ns ≥ T

2. 请确保插入地址建立等待和地址保持等待。

备注　**1.** w_{AS}：AWC 寄存器设置的地址建立等待数量

w_{AH}：AWC 寄存器设置的地址保持等待数量

w_D：DWC0 寄存器设置的数据等待数量

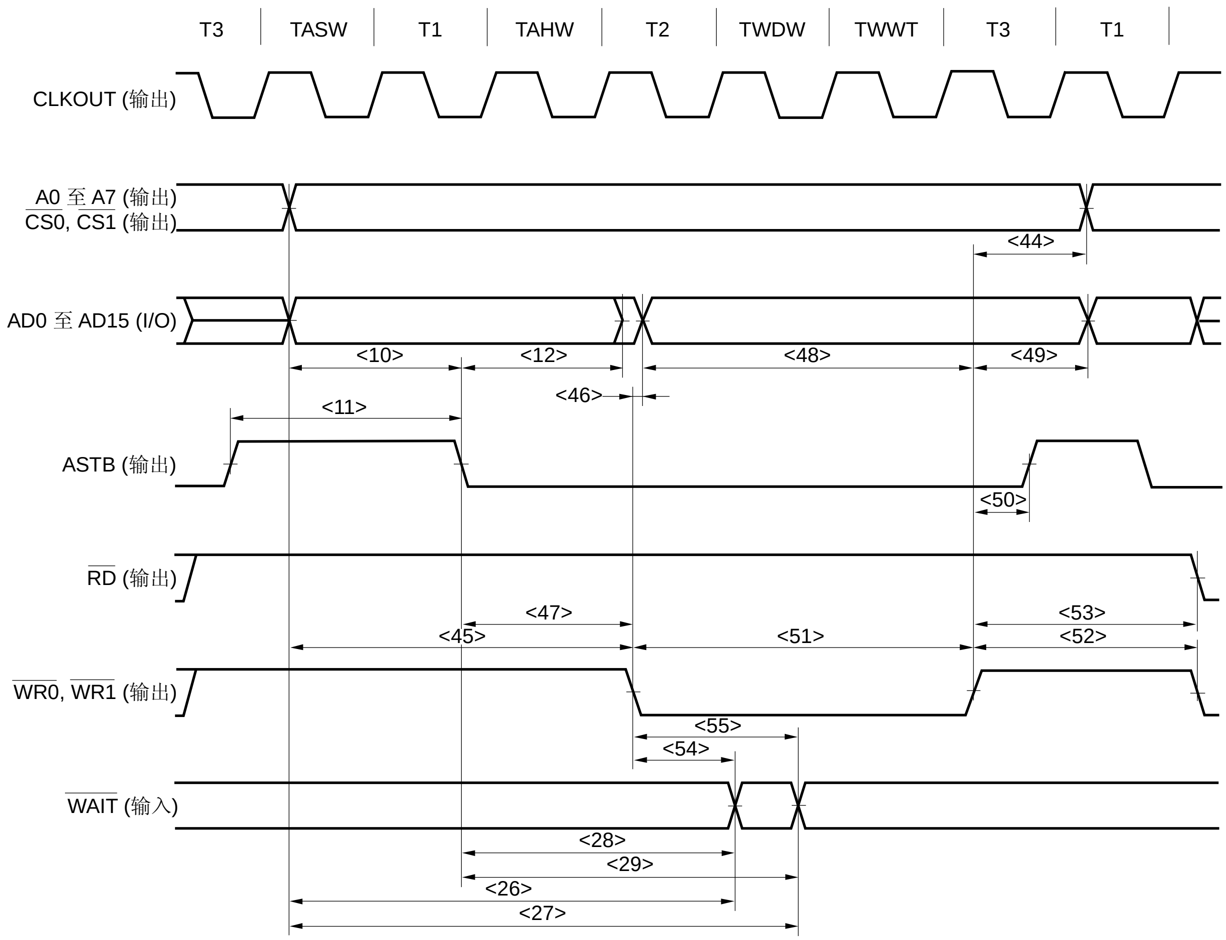
w：$\overline{\text{WAIT}}$ 引脚导致的外部等待数量

2. T = 1/f_{CPU} (f_{CPU}：CPU 工作时钟频率)

3. n = 0，1

4. i：空闲状态的数量

写入周期 **(CLKOUT 异步)**



备注　当地址设置等待的数量为 1，地址保持等待的数量为 1，数据等待的数量为 1，且 $\overline{\text{WAIT}}$ 引脚的等待数量为 1 时(在已决定的等待时期，当一个有效电平(低电平)的输入持续一个周期)，上述时序图展示了时序，

(d) 写入周期 (CLKOUT 同步)

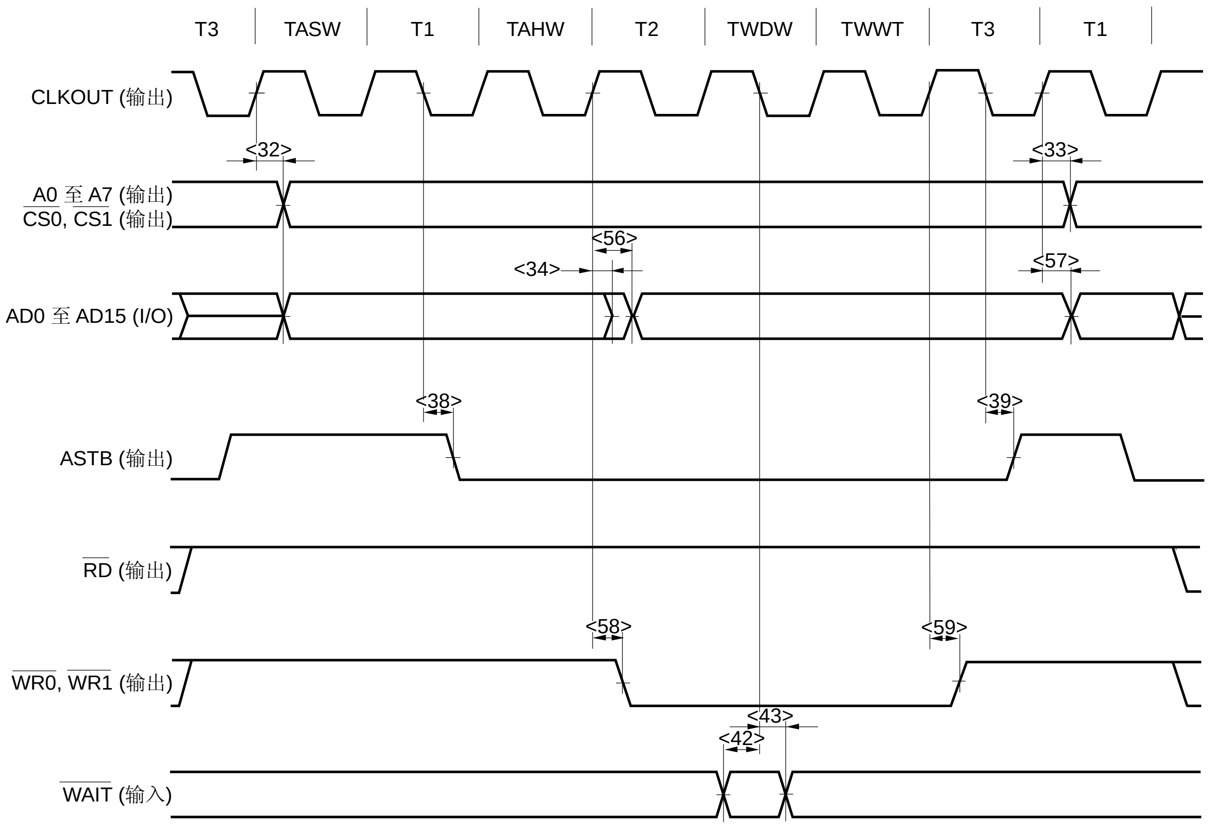
(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = EV_{DD2} = AV_{DD0} = AV_{DD1} = AV_{DD2} = 4.0 至 5.5 V, V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = EV_{SS2} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V, C_L = 50 pF)

参数	符号		条件	MIN.	MAX.	单位
从 CLKOUT↑ 到地址的延迟时间	t_{DKA2}	<32>			22	ns
自 CLKOUT↑ 的地址保持时间	t_{HKA2}	<33>		–5		ns
自 CLKOUT↑ 的地址保持时间	t_{HKA3}	<34>		–8		ns
从 CLKOUT↓ 到 ASTB↓ 的延迟时间	t_{DKST3}	<38>		–8	18	ns
从 CLKOUT↓ 到 ASTB↑ 延迟时间	t_{DKST4}	<39>		–8	18	ns
从 CLKOUT↑ 到数据输出的延迟时间	t_{DKOD3}	<56>			22	ns
从 CLKOUT↑的数据输出保持时间	t_{HKOD2}	<57>		–9		ns
从 CLKOUT↑ 到 $\overline{\text{WRn}}$↓的延迟时间	t_{DKWR3}	<58>		–10	17	ns
从 CLKOUT↑ 到 $\overline{\text{WRn}}$↑的延迟时间	t_{DKWR4}	<59>		–10	17	ns
$\overline{\text{WAIT}}$ 建立时间(至 CLKOUT↓)	t_{SWTK2}	<42>		30		ns
$\overline{\text{WAIT}}$ 保持时间 (自 CLKOUT↓)	t_{HKWT2}	<43>		9		ns

<R> 注意事项　请确保插入地址建立等待和地址保持等待。

备注　　n = 0, 1

写入周期 **(CLKOUT 同步)**



备注　当地址设置等待的数量为 1，地址保持等待的数量为 1，数据等待的数量为 1，且 WAIT 引脚的等待数量为 1 时(在已决定的等待时期，当一个有效电平(低电平)的输入持续一个周期)，上述时序图展示了时序，

(5) 定时器时序

(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = EV_{DD2} = AV_{DD0} = AV_{DD1} = AV_{DD2} = 4.0 至 5.5 V, V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = EV_{SS2} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V, C_L = 50 pF)

参数	符号		条件	MIN.	MAX.	单位
TIn高电平宽度[注 1, 2]	t_{WTIHn}	<60>	n = B00 至 B03, B10 至 B13	12T + 10		ns
			n = A20, A21, A30, A31, A40, A41	3T_{smp1} + 10		ns
TIn 低电平宽度[注 1, 2]	t_{WTILn}	<61>	n = B00 至 B03, B10 至 B13	12T + 10		ns
			n = A20, A21, A30, A31, A40, A41	3T_{smp1} + 10		ns
EVTBm高电平宽度[注 1]	t_{WEVBHm}	<62>	m = 0, 1	12T + 10		ns
EVTBm 低电平宽度[注 1]	t_{WEVBLm}	<63>	m = 0, 1	12T + 10		ns
TRGBm高电平宽度[注 1]	t_{WTRHm}	<64>	m = 0, 1	12T + 10		ns
TRGBm 低电平宽度[注 1]	t_{WTRLm}	<65>	m = 0, 1	12T + 10		ns
TENCm0/TENCm1 高电平宽度[注 3]	t_{WENCHm}	<66>	m = 0, 1	3T_{smp2} + 10		ns
TENCm0/TENCm1 低电平宽度[注 3]	t_{WENCLm}	<67>	m = 0, 1	3T_{smp2} + 10		ns
TECRm高电平宽度[注 3]	t_{WCRHm}	<68>	m = 0, 1	3T_{smp2} + 10		ns
TECRm 低电平宽度[注 3]	t_{WCRLm}	<69>	m = 0, 1	3T_{smp2} + 10		ns
TITm0/TITm1 高电平宽度[注 3]	t_{WTITHm}	<70>	m = 0, 1	3T_{smp2} + 10		ns
TITm0/TITm1 低电平宽度[注 3]	t_{WTITLm}	<71>	m = 0, 1	3T_{smp2} + 10		ns
EVTTm高电平宽度[注 3]	t_{WEVTHm}	<72>	m = 0, 1	3T_{smp2} + 10		ns
EVTTm 低电平宽度[注 3]	t_{WEVTLm}	<73>	m = 0, 1	3T_{smp2} + 10		ns
TENCm0/TENCm1 输入时间差别[注 3]	t_{PHUDm}	<74>	m = 0, 1	3T_{smp2} + 10		ns

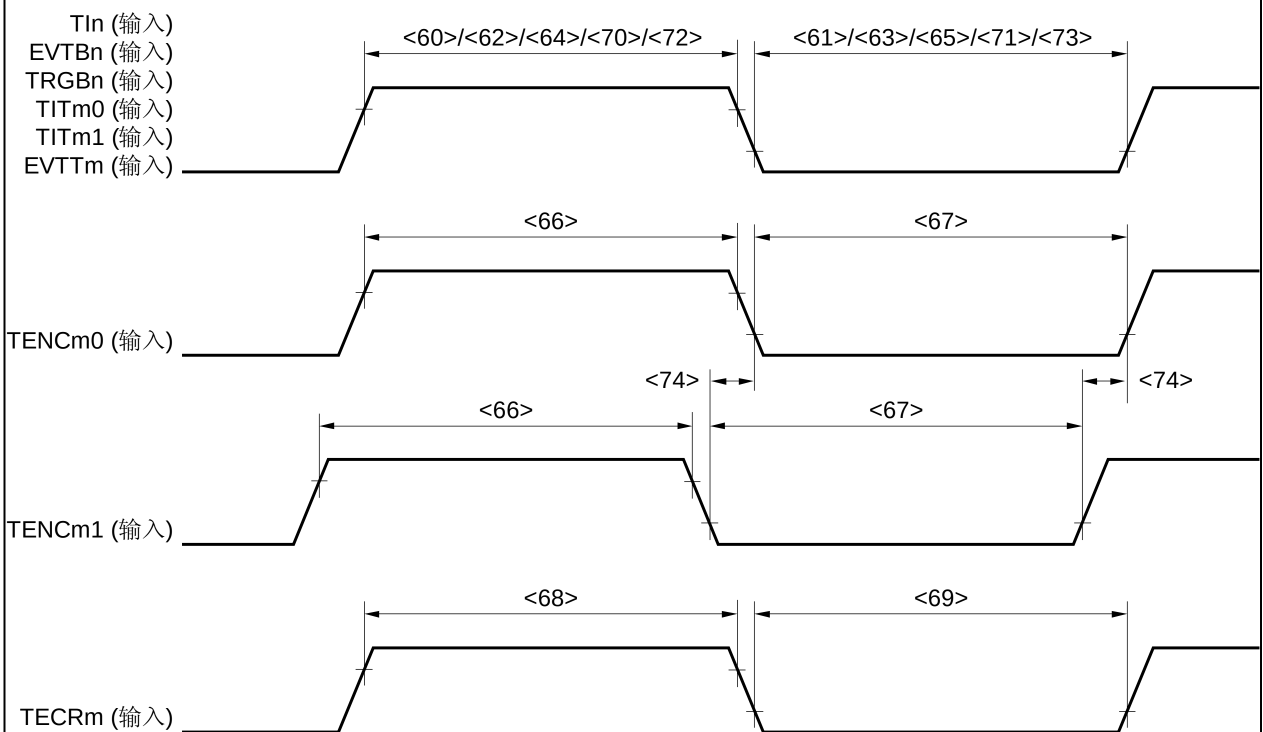
注　**1.** T = 1/f_{XX}

2. T_{smp1}：噪声消除采样时钟周期(由 TANFC2 至 TANFC4 寄存器设置)

3. T_{smp2}：噪声消除采样时钟周期(由 TTNFC0 和 TTNFC1 寄存器设置)

备注　上述规范展示了被准确检测为有效边沿的脉冲宽度。因此，即使输入的脉冲窄于上述规范，也可能被检测作为有效边沿。

定时器输入时序



备注　　n = A20, A21, A30, A31, A40, A41, B00 至 B03, B10 至 B13
　　　　m = 0, 1

(6) CSIB 时序

(a) 主模式

(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = EV_{DD2} = AV_{DD0} = AV_{DD1} = AV_{DD2} = 4.0 至 5.5 V, V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = EV_{SS2} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V, C_L = 50 pF)

参数	符号		条件	MIN.	MAX.	单位
$\overline{\text{SCKBn}}$ 周期	t_{KCYM}	<75>		125		ns
$\overline{\text{SCKBn}}$ 高/低电平宽度	t_{KWHM}, t_{KWLM}	<76>		t_{KCYM}/2 – 10		ns
SIBn 建立时间(至 $\overline{\text{SCKBn}}$↑)	t_{SSIM}	<77>		30		ns
SIBn 建立时间(至 $\overline{\text{SCKBn}}$↓)				30		ns
SIBn 保持时间 (自 $\overline{\text{SCKBn}}$↑)	t_{HSIM}	<78>		30		ns
SIBn 保持时间 (自 $\overline{\text{SCKBn}}$↓)				30		ns
SOBn 输出延迟时间 (自 $\overline{\text{SCKBn}}$↓)	t_{DSOM}	<79>			30	ns
SOBn 输出延迟时间 (自 $\overline{\text{SCKBn}}$↑)					30	ns
SOBn 输出保持时间 (自 $\overline{\text{SCKBn}}$↑)	t_{HSOM}	<80>		t_{KCYM}/2 – 10		ns
SOBn 输出保持时间 (自 $\overline{\text{SCKBn}}$↓)				t_{KCYM}/2 – 10		ns

备注　　n = 0 至 2

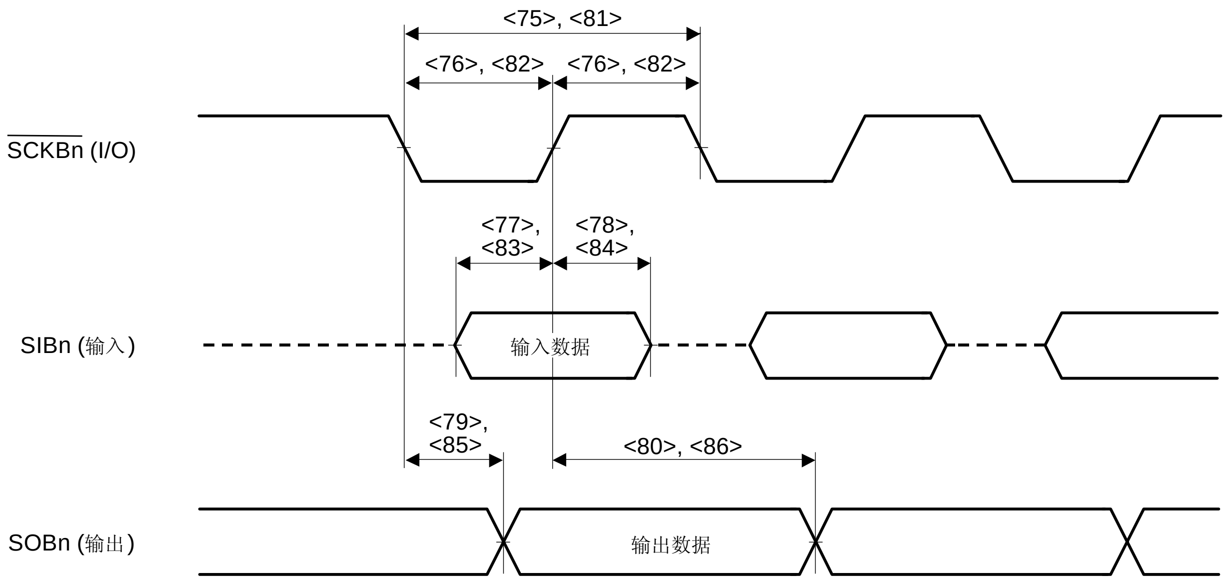
(b) 从模式

(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = EV_{DD2} = AV_{DD0} = AV_{DD1} = AV_{DD2} = 4.0 至 5.5 V, V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = EV_{SS2} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V, C_L = 50 pF)

参数	符号		条件	MIN.	MAX.	单位
$\overline{\text{SCKBn}}$ 周期	t_{KCYS}	<81>		125		ns
$\overline{\text{SCKBn}}$ 高/低电平宽度	t_{KWHS}, t_{KWLS}	<82>		t_{KCYS}/2 – 10		ns
SIBn 建立时间(至 $\overline{\text{SCKBn}}$↑)	t_{SSIS}	<83>		30		ns
SIBn 建立时间(至 $\overline{\text{SCKBn}}$↓)				30		ns
SIBn 保持时间 (自 $\overline{\text{SCKBn}}$↑)	t_{HSIS}	<84>		30		ns
SIBn 保持时间 (自 $\overline{\text{SCKBn}}$↓)				30		ns
SOBn 输出延迟时间 (自 $\overline{\text{SCKBn}}$↓)	t_{DSOS}	<85>			30	ns
SOBn 输出延迟时间 (自 $\overline{\text{SCKBn}}$↑)					30	ns
SOBn 输出保持时间 (自 $\overline{\text{SCKBn}}$↑)	t_{HSOS}	<86>		t_{KCYS}/2 – 10		ns
SOBn 输出保持时间 (自 $\overline{\text{SCKBn}}$↓)				t_{KCYS}/2 – 10		ns

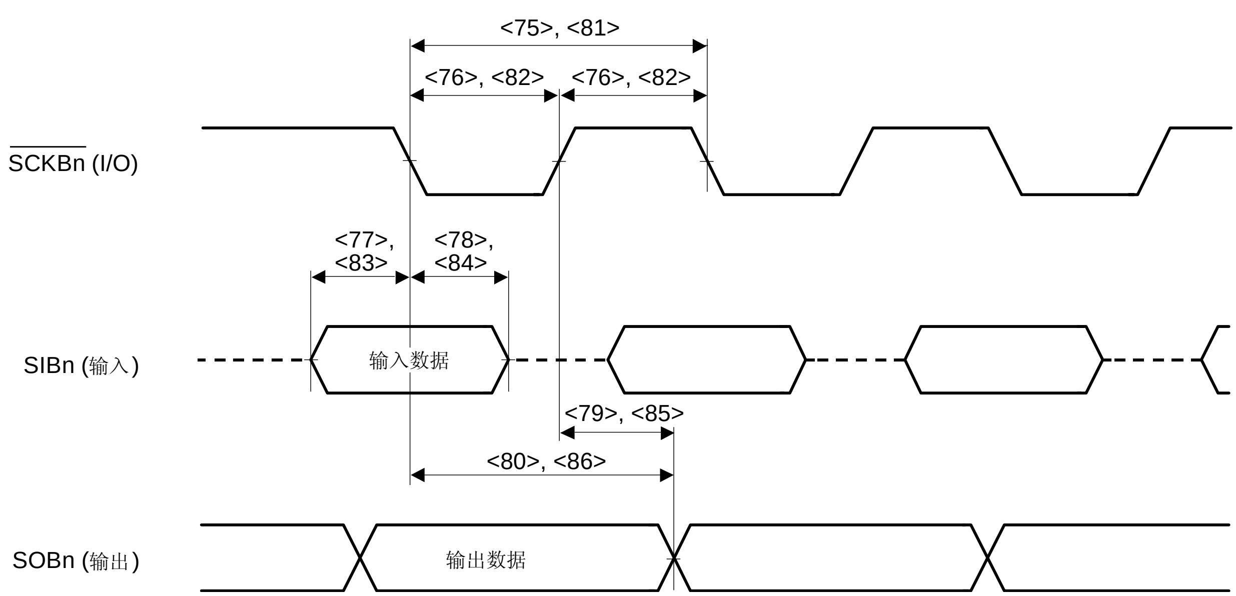
备注　　n = 0 至 2

当 CBnCTL1 寄存器的 CBnCKP 和 CBnDAP 位 = 00 时的 CSIB 时序



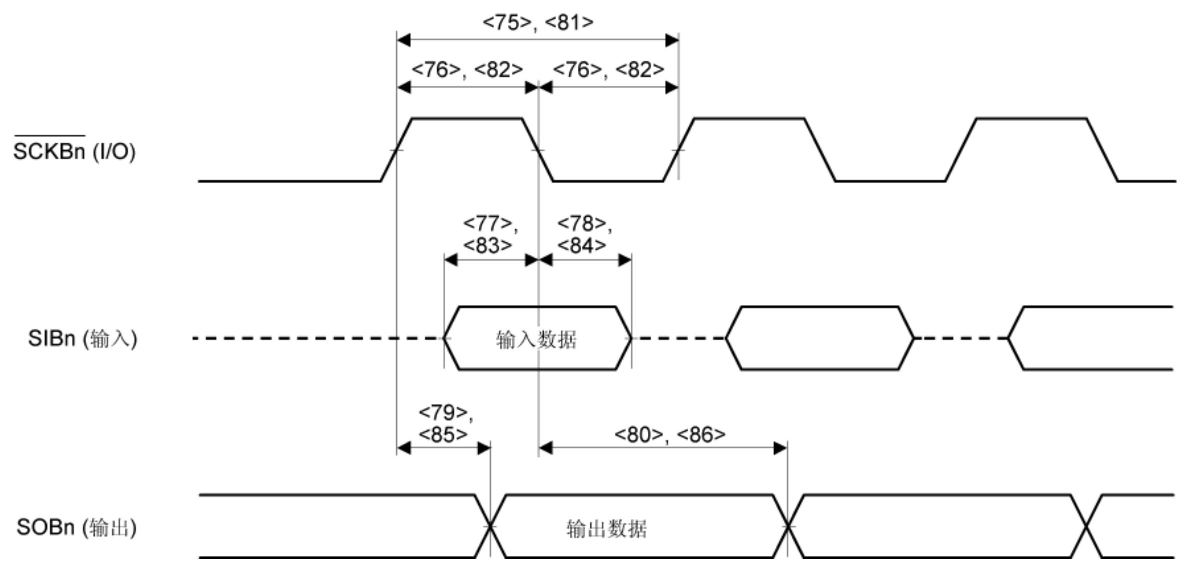
备注 **1.** 虚线表示高阻抗
2. n = 0 至 2

当 CBnCTL1 寄存器的 CBnCKP 和 CBnDAP 位 = 01 时的 CSIB 时序



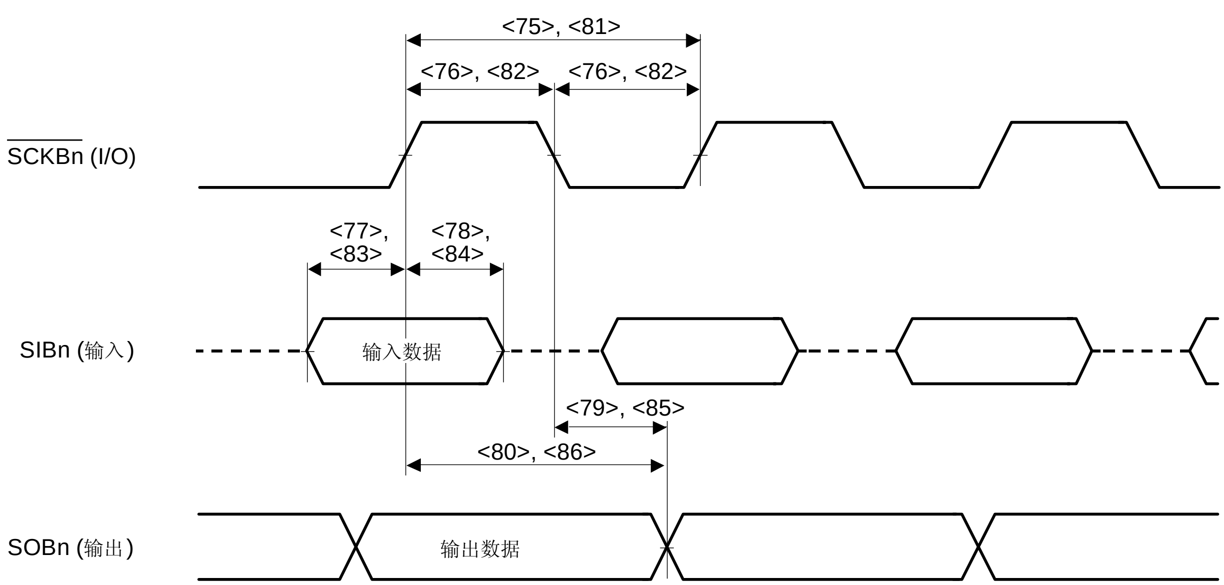
备注 **1.** 虚线表示高阻抗
2. n = 0 至 2

当 CBnCTL1 寄存器的 CBnCKP 和 CBnDAP 位 = 10 时的 CSIB 时序



备注 **1.** 虚线表示高阻抗
2. n = 0 至 2

当 CBnCTL1 寄存器的 CBnCKP 和 CBnDAP 位 = 11 时的 CSIB 时序



备注 **1.** 虚线表示高阻抗
2. n = 0 至 2

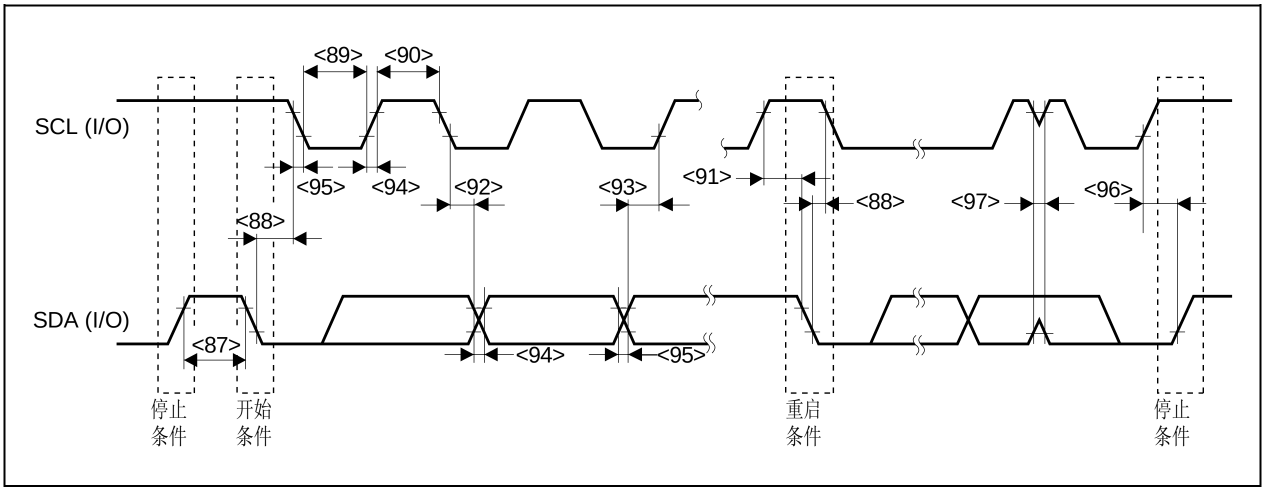
(7) I²C 总线时序

(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = EV_{DD2} = AV_{DD0} = AV_{DD1} = AV_{DD2} = 4.0 至 5.5 V, V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = EV_{SS2} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V, C_L = 50 pF)

参数		符号		标准模式		高速模式		单位
				MIN.	MAX.	MIN.	MAX.	
SCL 时钟频率		f_{CLK}	–	0	100	0	400	kHz
总线自由时间 (停止条件和开始条件之间)		t_{BUF}	<87>	4.7	–	1.3	–	μs
保持时间[注 1]		$t_{HD:STA}$	<88>	4.0	–	0.6	–	μs
SCL 时钟低电平宽度		t_{LOW}	<89>	4.7	–	1.3	–	μs
SCL 时钟高电平宽度		t_{HIGH}	<90>	4.0	–	0.6	–	μs
开始/重启条件建立时间		$t_{SU:STA}$	<91>	4.7	–	0.6	–	μs
数据保持时间	CBUS-兼容主	$t_{HD:DAT}$	<92>	5.0	–	–	–	μs
	I²C模式			0[注 2]	–	0[注 2]	0.9[注 3]	μs
数据建立时间		$t_{SU:DAT}$	<93>	250	–	100[注 4]	–	ns
SDA，SCL 信号上升时间		t_R	<94>	–	1000	20 + 0.1Cb[注 5]	300	ns
SDA，SCL 信号下降时间		t_F	<95>	–	300	20 + 0.1Cb[注 5]	300	ns
停止条件建立时间		$t_{SU:STO}$	<96>	4.0	–	0.6	–	μs
输入滤波器滤除的毛刺脉冲宽度		t_{SP}	<97>	–	–	0	50	ns
每条总线的容性负载		Cb	–	–	400	–	400	pF

注
1. 开始条件期间，在保持时间之后产生第一个时钟脉冲。
2. 系统必须为 SDA 信号内部提供至少 300 ns 的保持时间，(在 SCL 信号的 V_{IHmin}) 用于在 SCL 下降沿处填充不确定区域。
3. 如果系统不延长 SCL 信号的低状态保持时间 (t_{LOW})，必须满足最大数据保持时间($t_{HD:DAT}$)。
4. 在标准模式 I²C 总线系统下可以使用高速模式 I²C 总线。在这种情况下，请确保满足下列条件。
   - 如果系统不延长 SCL 信号的低状态保持时间。
     $t_{SU:DAT} \geq 250$ ns
   - 如果系统延长 SCL 信号的低状态保持时间。
     在SCL线释放之前，将下一个数据位发送到SDA线($t_{Rmax.}$ + $t_{SU:DAT}$ = 1000 + 250 = 1250 ns：标准模式I²C总线规范)。
5. Cb：一条总线的线间总电容(单位：pF)

I²C 总线时序



(8) 高阻抗控制时序

(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = EV_{DD2} = AV_{DD0} = AV_{DD1} = AV_{DD2} = 4.0 至 5.5 V,
V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = EV_{SS2} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V, C_L = 50 pF)

参数	符号	条件	MIN.	MAX.	单位
振荡停止 →定时器输出高阻抗	t_{CLM}	当时钟监视器工作时		65	μs
输入到 TOBnOFF →定时器输出高阻抗	t_{HTQn}			300	ns
输入到 TOAmOFF →定时器输出高阻抗	t_{HTPm}			300	ns
输入到 ANI00/ANI05 →定时器输出高阻抗	t_{ANI0}			10	μs
输入到 ANI10/ANI15 至 ANI12/ANI17 →定时器输出高阻抗	t_{ANI1}			10	μs

备注 n = 0, 1
m = 2, 3

28.2.9 A/D 转换器 0, 1 的特性

(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = EV_{DD2} = AV_{DD0} = AV_{DD1} = AV_{DD2} = AV_{REFP0} = AV_{REFP1} = 4.0 至 5.5 V, V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = EV_{SS2} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V, C_L = 50 pF)

参数	符号	条件	MIN.	TYP.	MAX.	单位
分辨率			12	12	12	位
总体误差[注 1]					±10	LSB
转换时间	t_{CONV}	f_{AD01} = 16 MHz, ADAnCTC 寄存器 = 0BH 或 0CH	2.0			μs
		f_{AD01} = 12 MHz, ADAnCTC 寄存器 = 00H			7.42	μs
零标度误差[注 1]					±10	LSB
满量程误差[注 1]					±10	LSB
积分线性误差[注 1]					±4	LSB
微分线性误差[注 1]					±2.5	LSB
模拟参考电压	AV_{DD}		4.0		5.5	V
模拟输入电压	V_{IAN}		AV_{SS}		AV_{DD}	V
AV_{DD} 供电电流[注 2]	AI_{DD}	在工作期间		4.5	7.5	mA
	AI_{DDS}	在STOP模式下[注 3]		3.5	17.5	μA

注 **1.** 不包括量化误差 (±0.5 LSB)

2. 该值仅对一路 A/D 转换器 (A/D 转换器 0 或 1)

3. 在设置 STOP 模式之前，停止 A/D 转换器 0 和 1 的操作(ADnSCM.ADnCE 位 = 0) 。

备注 **1.** LSB：最低有效位

2. f_{AD01}：A/D 转换器 0 和 1 的基准时钟

3. n = 0, 1

28.2.10 A/D 转换器 2 的特性

(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = EV_{DD2} = AV_{DD0} = AV_{DD1} = AV_{DD2} = 4.0 至 5.5 V, V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = EV_{SS2} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V, C_L = 50 pF)

参数	符号	条件	MIN.	TYP.	MAX.	单位
分辨率			10	10	10	位
总体误差注 1					±4.0	LSB
转换时间	t_{CONV}		3.88		10	μs
零标度误差注 1					±4.0	LSB
满量程误差注 1					±4.0	LSB
积分线性误差注 1					±4.0	LSB
微分线性误差注 1					±2.0	LSB
模拟参考电压	AV_{DD}		4.0		5.5	V
模拟输入电压	V_{IAN}		AV_{SS}		AV_{DD}	V
AV_{DD} 供电电流	AI_{DD}	在工作期间		3.5	7	mA
	AI_{DDS}	在STOP模式下注 2		1	10	μA

注 **1.** 不包括量化误差 (±0.5 LSB).

2. 在设置 STOP 模式之前，停止 A/D 转换器 2 的操作(AD2M0.AD2CE 位 = 0) 。

备注 LSB：最低有效位

28.2.11　运算放大器特性

(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = EV_{DD2} = AV_{DD0} = AV_{DD1} = AV_{DD2} = 4.0 至 5.5 V, V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = EV_{SS2} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V, C_L = 50 pF)

参数	符号	条件		MIN.	TYP.	MAX.	单位
输入偏置电压	V_{IO}				±9.0		mV
输入电压范围	V_I	增益 = 2.500		$0.04AV_{DD}$		$0.36AV_{DD}$	V
		增益 = 5.000		$0.02AV_{DD}$		$0.18AV_{DD}$	V
		增益 = 10.00		$0.01AV_{DD}$		$0.085AV_{DD}$	V
转换速率注 1	S_R			10	15		V/µs
增益误差		**注 2**	增益 = 2.500 至 4.444		±1.0	±1.3	%
			增益 = 5.000 至 6.667		±1.0	±1.5	%
			增益 = 8.000, 10.00		±1.0	±1.7	%
		注 3	增益 = 2.500 至 4.444		±1.0	±2.0	%
			增益 = 5.000 至 6.667		±1.0	±2.1	%
			增益 = 8.000, 10.00		±1.0	±2.2	%
工作电流注 4	I_{OPDD}	在工作期间			1.8	2.6	mA
	A_{IDDS}	在STOP模式下注 5			1.0	10	µA

注
1. 输出电压从 10%至 90%的斜率特性。
2. AV_{DD0} = AV_{DD1} = 4.5 至 5.5 V
3. AV_{DD0} = AV_{DD1} = 4.0 至 5.5 V
4. 总共提供四路运算放大器。该值为每个运算放大器的工作电流。
5. 设置 STOP 模式之前，停止运算放大器的操作(OP0CTL0.OP0EN 位 = 0, OP1CTL0.OP12EN, OP11EN 和 OP10EN 位 s = 0))。

备注　运算放大器使用的供电电压是 AV_{DD0} 和 AV_{DD1}

28.2.12　比较器特性

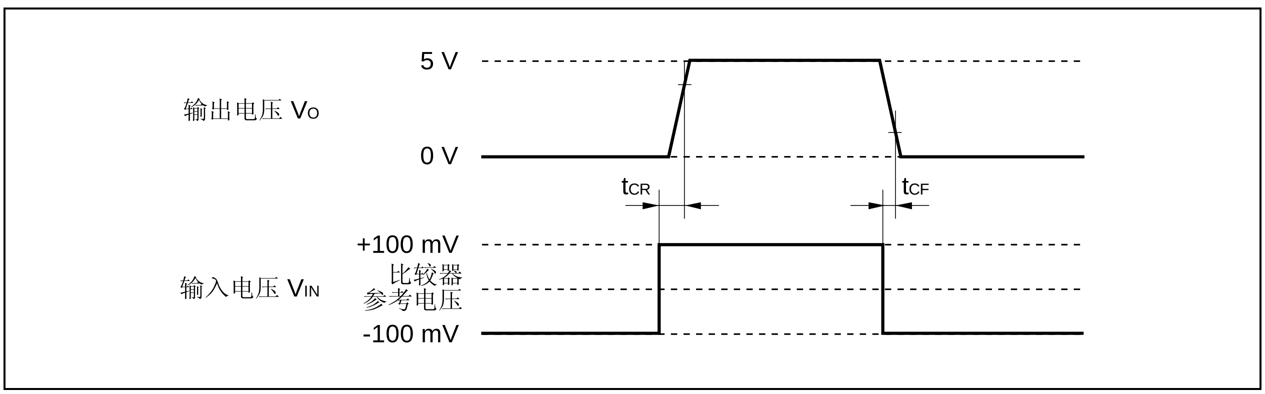
(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = EV_{DD2} = AV_{DD0} = AV_{DD1} = AV_{DD2} = 4.0 至 5.5 V,
V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = EV_{SS2} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V, C_L = 50 pF)

参数	符号	条件	MIN.	TYP.	MAX.	单位
输入偏置电压	V_{IO}			±3.0		mV
输入电压范围	V_I		AV_{SS}		AV_{DD}	V
比较器参考电压 (全范围)	CREFnF		$0.02AV_{DD}$ + 0.1		$0.92AV_{DD}$ – 0.1	V
比较器参考电压 (低范围)	CREFnL		$0.02AV_{DD}$ + 0.1		$0.5AV_{DD}$ – 0.1	V
响应时间	t_{CR}	输入幅度 = 100 mV, 在上升沿注 1		1.0		μs
	t_{CF}	输入幅度 = 100 mV, 在下降沿注 2		1.0		μs
工作电流注 3	I_{CPDD}	在工作期间			250	μA
	A_{IDDS}	在STOP模式下注 4		2.0	20	nA

注　**1.** 当 ANIm 输入从比较器参考电压 – 100 mV 改变为比较器参考电压 + 100 mV 时，脉冲响应的特性。
2. 当 ANIm 输入从比较器参考电压 + 100 mV 改变为比较器参考电压 – 100 mV 时，脉冲响应的特性。
3. 总共提供四路比较器。该值为每个比较器的工作电流。
4. 设置 STOP 模式之前，停止比较器操作(CMPnCTL0 寄存器 = 00H)。

备注　**1.** 比较器使用的供电电压是 AV_{DD0} 和 AV_{DD1}
2. m = 05, 15 至 17
n = 0, 1

比较器特性

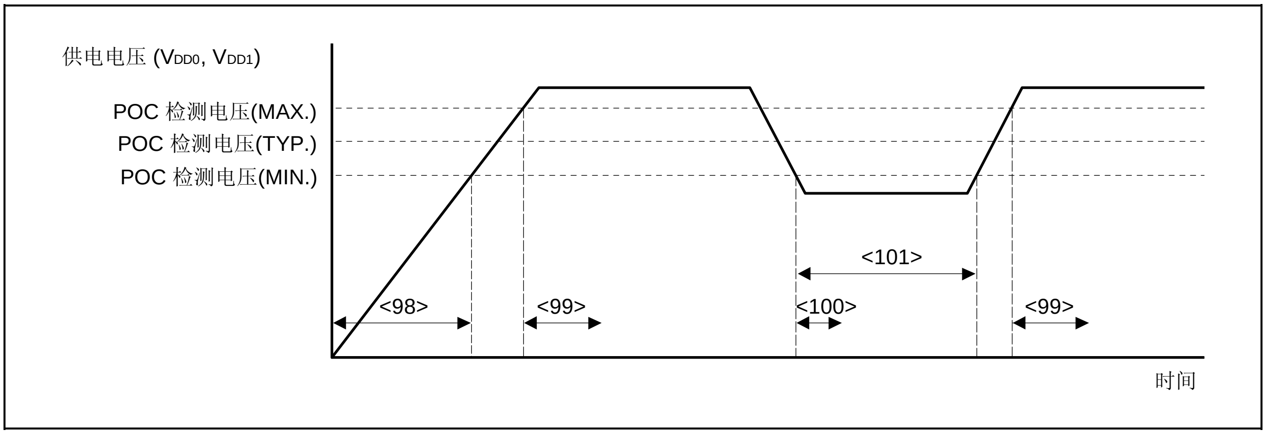


28.2.13　上电清零电路(POC)

(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = EV_{DD2} = AV_{DD0} = AV_{DD1} = AV_{DD2} = 3.5 至 5.5 V,
V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = EV_{SS2} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V, C_L = 50 pF)

参数	符号		条件	MIN.	TYP.	MAX.	单位
POC 检测电压	V_{POC0}			3.5	3.7	3.9	V
供电电压上升时间	t_{PTH}	<98>	V_{DD0}, V_{DD1} = 0 至 3.5 V	2.5 μs		1.8 s	
响应时间 1注 1	t_{PTHD}	<99>	上电时当 V_{DD0} 和 V_{DD1} 达到 3.9 V 之后			3.0	ms
响应时间 2注 2	t_{PD}	<100>	下电时当 V_{DD0} 和 V_{DD1} 下降到 3.5 V 之后			1.0	ms
V_{DD0}, V_{DD1} 的最小宽度	t_{PW}	<101>		0.2			ms

注　**1.** 检测到 POC 检测电压之后，释放复位信号(POCRES)所需的时间。
　　2. 检测到 POC 检测电压之后，输出复位信号(POCRES)所需的时间。

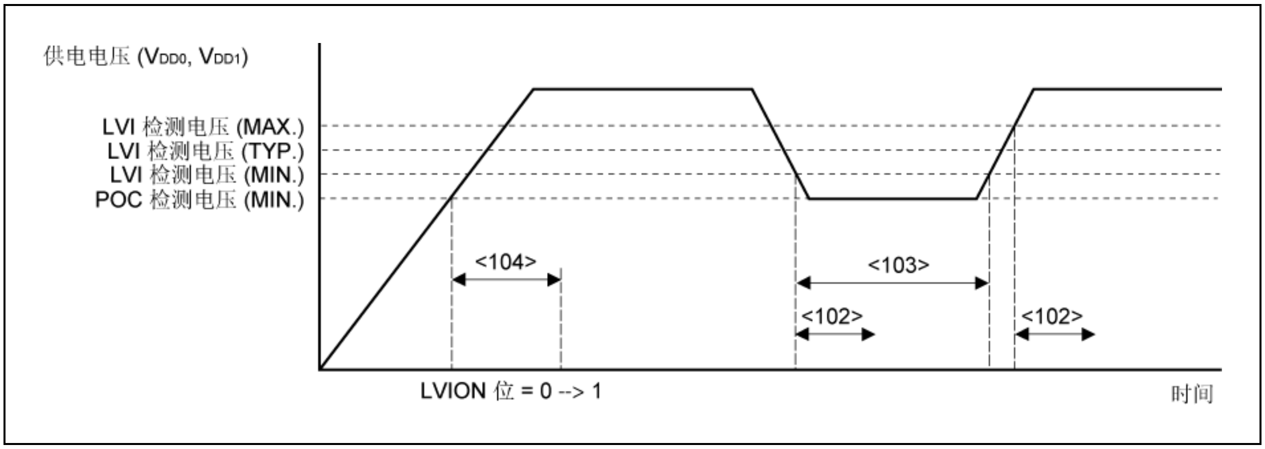


28.2.14 低电压检测器 (LVI)

(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = EV_{DD2} = AV_{DD0} = AV_{DD1} = AV_{DD2} = 3.5 至 5.5 V, V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = EV_{SS2} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V, C_L = 50 pF)

参数	符号		条件	MIN.	TYP.	MAX.	单位
LVI 检测电压	V_{LVI0}		LVIS.LVIS0 位 = 0	4.2	4.4	4.6	V
	V_{LVI1}		LVIS.LVIS0 位 = 1	4.0	4.2	4.4	V
响应时间 1[注]	t_{LD}	<102>	当 V_{DD0} 和 V_{DD1} 达到 V_{LVI0}/V_{LVI1} (MAX.) 或下降到 V_{LVI0}/V_{LVI1} (MIN.) 之后		0.2	2.0	ms
V_{DD0}, V_{DD1} 的最小宽度	t_{LW}	<103>		0.2			ms
参考电压稳定等待时间	t_{LWAIT}	<104>	当 V_{DD0} 和 V_{DD1} 达到 POC 检测电压 (MIN.)之后，且 LVIM.LVION 位从 0 改变为 1		0.1		ms

注　检测到 LVI 检测电压之后，输出一个中断请求信号(INTLVIL，INTLVIH)或内部复位信号(LVIRES)所需的时间。



28.2.15 Flash 存储器编程特性

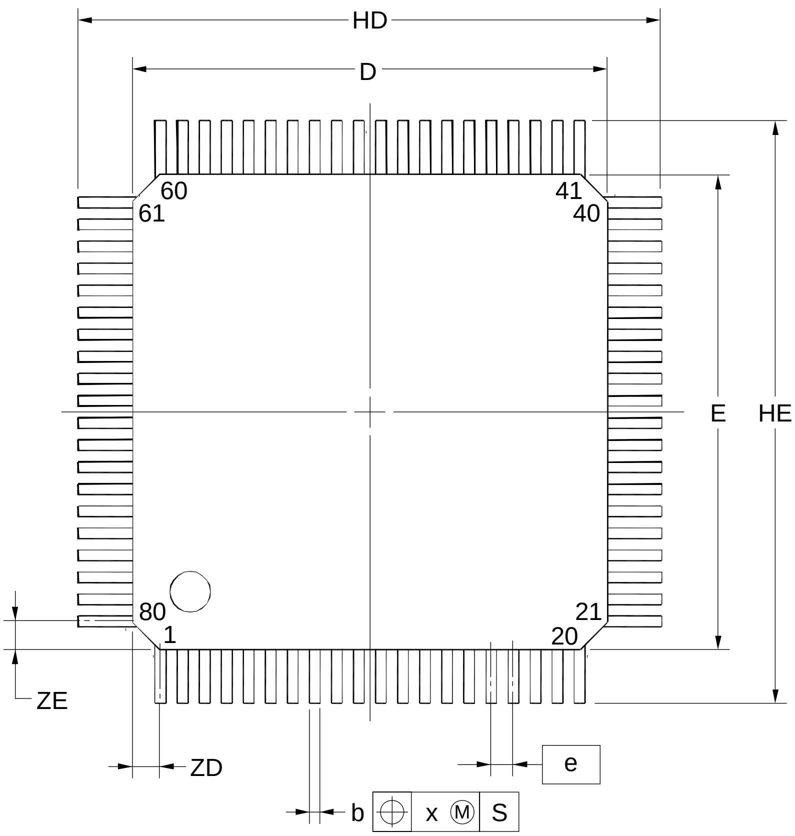
(T_A = –40 至+85°C, V_{DD0} = V_{DD1} = EV_{DD0} = EV_{DD1} = EV_{DD2} = AV_{DD0} = AV_{DD1} = AV_{DD2} = 3.5 至 5.5 V,
V_{SS0} = V_{SS1} = EV_{SS0} = EV_{SS1} = EV_{SS2} = AV_{SS0} = AV_{SS1} = AV_{SS2} = 0 V, C_L = 50 pF)

参数	符号	条件	MIN.	TYP.	MAX.	单位
重写次数	C_{ERWR}	注	100			次

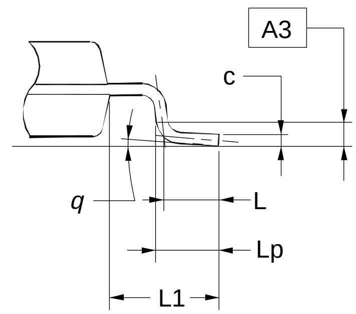
注　　重写如下。

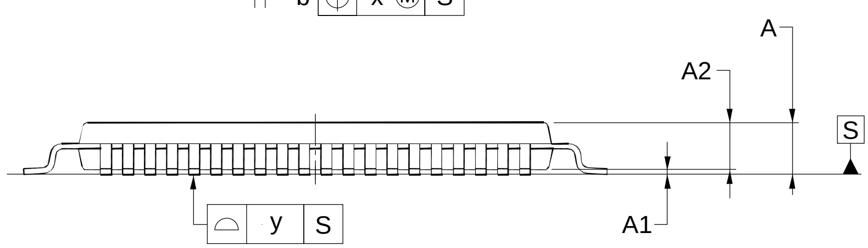
三次重写的示例：出厂产品→E→P→E→P→E→P (P：写入，E：擦除)

80-针塑封 LQFP(14x14)



引脚端的细节图





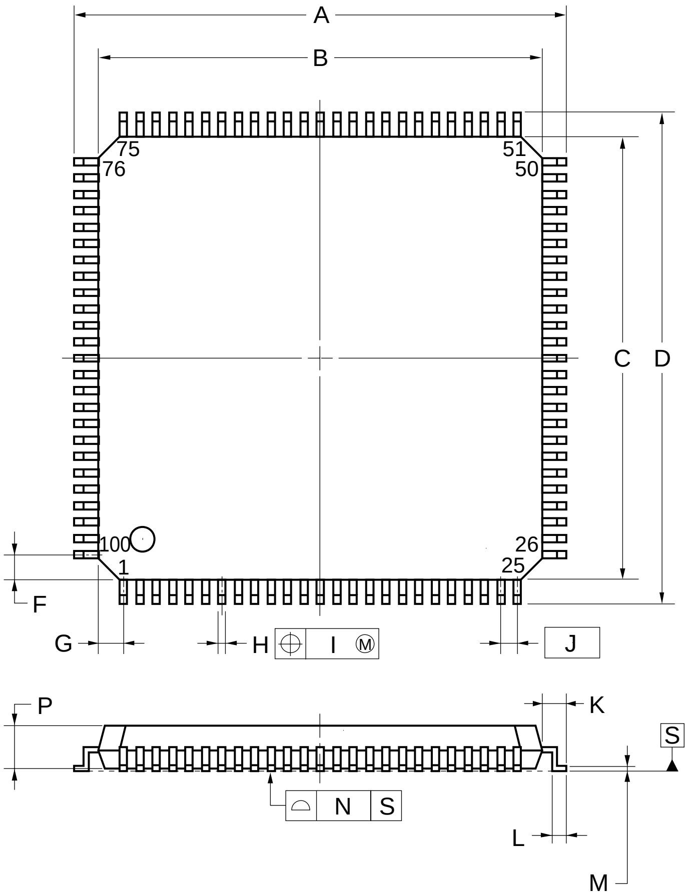
注
在最大公差条件下，每个引脚的中心线位于真实位置的
0.13 mm范围之内。

(单元：mm)

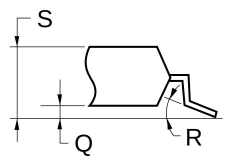
项目	尺寸
D	14.00±0.20
E	14.00±0.20
HD	17.20±0.20
HE	17.20±0.20
A	1.70 MAX.
A1	0.125±0.075
A2	1.40±0.05
A3	0.25
b	0.32±0.06
c	$0.17^{+0.03}_{-0.06}$
L	0.80
Lp	0.886±0.15
L1	1.60±0.20
q	$3°^{+5°}_{-3°}$
e	0.65
x	0.13
y	0.10
ZD	0.825
ZE	0.825

P80GC-65-UBT

100-针塑封LQFP (密间距) (14x14)



引脚端的细节图

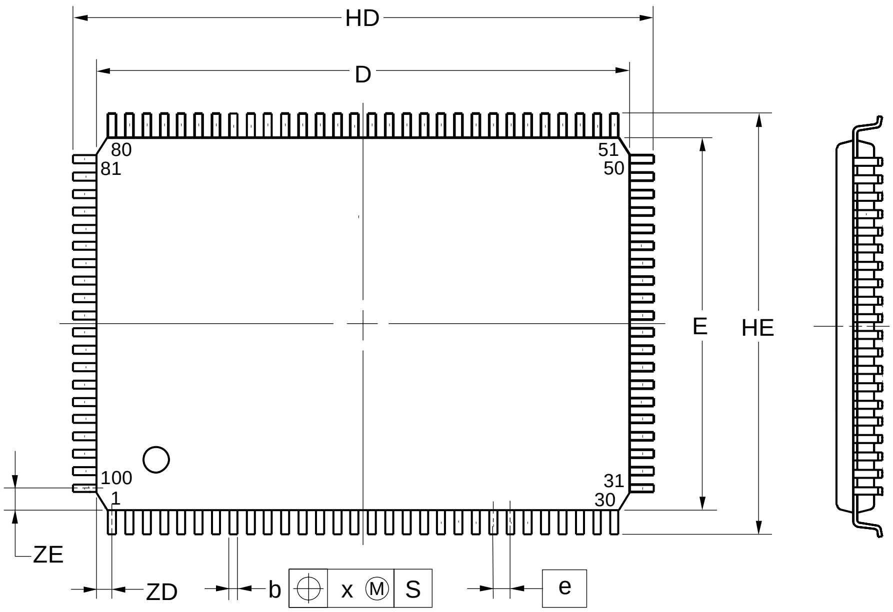


注
在最大公差条件下，每个引脚的中心线位于真实位置(T.P.)的0.08 mm范围之内。

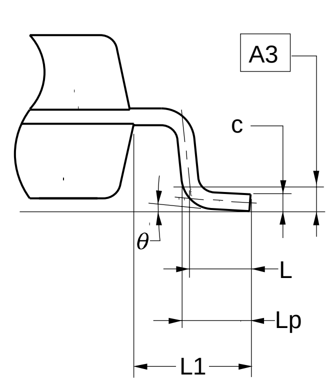
项目	毫米
A	16.00±0.20
B	14.00±0.20
C	14.00±0.20
D	16.00±0.20
F	1.00
G	1.00
H	$0.22^{+0.05}_{-0.04}$
I	0.08
J	0.50 (T.P.)
K	1.00±0.20
L	0.50±0.20
M	$0.17^{+0.03}_{-0.07}$
N	0.08
P	1.40±0.05
Q	0.10±0.05
R	$3°^{+7°}_{-3°}$
S	1.60 MAX.

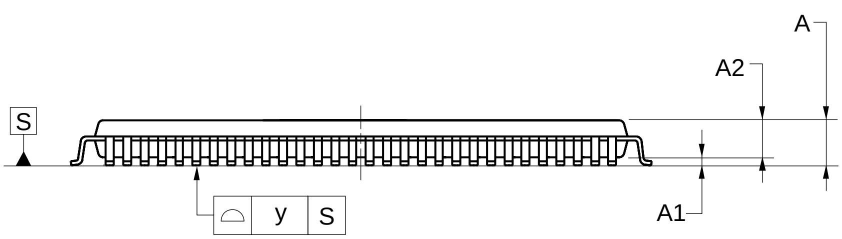
S100GC-50-8EU, 8EA-2

100-针塑封 LQFP (14x20)



引脚端的细节图





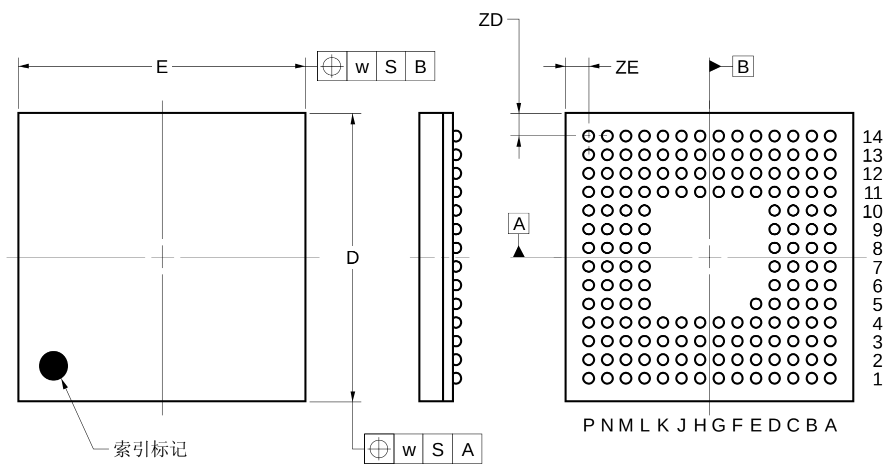
(单位:mm)

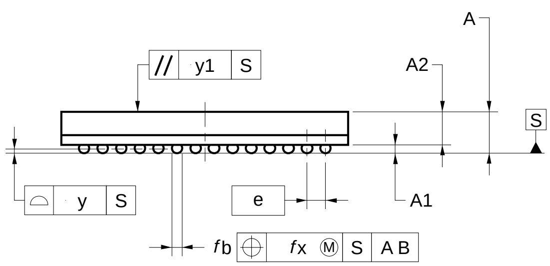
项目	尺寸
D	20.00±0.20
E	14.00±0.20
HD	22.00±0.20
HE	16.00±0.20
A	1.60 MAX.
A1	0.10±0.05
A2	1.40±0.05
$\boxed{A3}$	0.25
b	$0.30^{+0.08}_{-0.04}$
c	$0.125^{+0.075}_{-0.025}$
L	0.50
Lp	0.60±0.15
L1	1.00±0.20
θ	$3°^{+5°}_{-3°}$
$\boxed{e}$	0.65
x	0.13
y	0.10
ZD	0.575
ZE	

P100GF-65-GAS

注
在最大公差条件下，每个引脚的中心线位于真实位置的
0.13 mm范围之内。

161-针塑封 FBGA (10x10)





项目	毫米
D	10.00±0.10
E	10.00±0.10
w	0.20
A	1.43±0.10
A1	0.30±0.05
A2	1.13
e	0.65
b	0.40±0.05
x	0.08
y	0.10
y1	0.20
ZD	0.775
ZE	0.775

P161F1-65-DA1

<R>

第三十章 推荐焊接条件

V850E/IF3 和 V850E/IG3 产品应该在下列推荐条件下进行焊接和安装。
相关技术信息，请浏览下面的网页。

半导体设备装配手册（http：//www.necel.com/pkg/en/mount/index.html）

表 30-1. 表面贴装类型焊接条件(1/2)

μPD70F3451GC-UBT-A：80 针塑封 LQFP (14 × 14)
μPD70F3452GC-UBT-A：80 针塑封 LQFP (14 × 14)
μPD70F3453GC-8EA-A：100 针塑封 LQFP (密间距) (14 × 14)
μPD70F3454GC-8EA-A：100 针塑封 LQFP (密间距) (14 × 14)

焊接方法	焊接条件	推荐条件符号
红外回流	封装峰值温度： 260°C， 时间： 最长 60 秒 （ 大于等于 220°C）， 次数： 小于等于 3 次， 曝光限制： 7 天[注]（此后，在 125°C 预烘 10 至 72 小时）	IR60-207-3
波峰焊	详情请与 NEC 电子公司销售代理联系。	–
局部加热	引脚温度： 350°C 最大值.， 时间： 最大 3 秒 （每行引脚）	–

注　打开干燥包装之后，在有效存放期内将其存储在 25°C 或更低温度，65% RH 或更低湿度的环境中。

注意事项　不要一起使用不同的焊接方法（局部加热法除外）。

备注　1. 部件登记号末尾带有-A 的产品是无铅产品。
　　2. 如果不使用以上推荐的焊接方法和条件，请与 NEC 电子公司销售代理联系。

表 30-1. 表面贴装类型焊接条件(2/2)

***μ*PD70F3453GF-GAS-AX：　100 针塑封 LQFP (14 × 20)**
***μ*PD70F3454GF-GAS-AX：　100 针塑封 LQFP (14 × 20)**
***μ*PD70F3454F1-DA9-A：　161 针塑封 FBGA (10 × 10)**

焊接方法	焊接条件	推荐条件符号
红外回流	封装峰值温度： 260°C， 时间： 最长 60 秒 （ 大于等于 220°C）， 次数： 小于等于 3 次， 曝光限制： 7 天[注] （此后，在 125°C 预烘 20 至 72 小时）	IR60-207-3
波峰焊	详情请与 NEC 电子公司销售代理联系。	–
局部加热	引脚温度： 350°C 最大值.， 时间： 最大 3 秒 （每行引脚）	–

注　打开干燥包装之后，在有效存放期内将其存储在 25°C 或更低温度，65% RH 或更低湿度的环境中。

注意事项　不要一起使用不同的焊接方法（局部加热法除外）。

备注
1. 部件登记号末尾带有-AX 或-A 的产品是无铅产品。
2. 如果不使用以上推荐的焊接方法和条件，请与 NEC 电子公司销售代理联系。

附录 A　注意事项

A.1　sld 指令和中断请求之间的冲突限制

A.1.1　描述

如果在指令<1>执行完毕之前，指令<2>的译码操作和某个中断请求产生冲突，那么，指令<1>的执行结果就可能无法储存到寄存器中。指令<2>的位置紧邻在 sld 指令之前，且位于指令<1>之后。

指令 <1>

- ld 指令：　ld.b，ld.h，ld.w，ld.bu，ld.hu
- sld 指令：　sld.b，sld.h，sld.w，sld.bu，sld.hu
- 乘法指令：　Mul，mulh，mulhi，mulu

指令<2>

mov reg1，reg2	not reg1，reg2	satsubr reg1, reg2	satsub reg1, reg2
satadd reg1, reg2	satadd imm5, reg2	or reg1，reg2	xor reg1，reg2
and reg1，reg2	tst reg1，reg2	subr reg1，reg2	sub reg1，reg2
add reg1，reg2	add imm5，reg2	cmp reg1，reg2	cmp imm5，reg2
mulh reg1，reg2	shr imm5，reg2	sar imm5，reg2	shl imm5，reg2

<举例>

```
<i>   ld.w   [r11]，r10
              •
              •
              •
<ii>  mov   r10，r28
<iii> sld.w 0x28，r10
```

如果在 ld 指令<i>执行完毕之前，mov 指令<ii>的译码操作和某个中断请求产生冲突，那么，指令<i>的执行结果就可能无法储存到寄存器中。mov 指令<ii>紧邻在 sld 指令<iii>之前。

A.1.2　对策

(1) 当使用编译器（CA850）时

请使用 CA850 Ver. 2.61 或更高版本，因为它将自动禁止相应指令序列的产生。

(2) 使用汇编器时

在指令<ii>之后就立即执行 sld 指令，可以采用以下任一种方法来避免上述操作。

- 紧贴在 sld 指令之前插入 nop 指令。
- 在紧邻 sld 指令之前的指令<ii>中，不要使用同一个寄存器作为 sld 指令目标寄存器。

附录 B 寄存器索引

(1/13)

(2/13)

符号	名称	单元	页码
AD0ECR0	A/D0 转换结果扩展寄存器 0	ADC0	637
AD0ECR0H	A/D0 转换结果扩展寄存器 0H	ADC0	637
AD0ECR1	A/D0 转换结果扩展寄存器 1	ADC0	637
AD0ECR1H	A/D0 转换结果扩展寄存器 1H	ADC0	637
AD0ECR2	A/D0 转换结果扩展寄存器 2	ADC0	637
AD0ECR2H	A/D0 转换结果扩展寄存器 2H	ADC0	637
AD0ECR3	A/D0 转换结果扩展寄存器 3	ADC0	637
AD0ECR3H	A/D0 转换结果扩展寄存器 3H	ADC0	637
AD0ECR4	A/D0 转换结果扩展寄存器 4	ADC0	637
AD0ECR4H	A/D0 转换结果扩展寄存器 4H	ADC0	637
AD0FLG	A/D 转换器 0 标志寄存器	ADC0	639
AD0FLGB	A/D 转换器 0 标志缓冲寄存器	ADC0	640
AD0IC	中断控制寄存器	ADC0	1000
AD0OCKS	A/D 转换器 0 时钟选择寄存器	ADC0	651
AD0SCM	A/D 转换器 0 扫描模式寄存器	ADC0	622
AD0SCMH	A/D 转换器 0 扫描模式寄存器 H	ADC0	622
AD0SCML	A/D 转换器 0 扫描模式寄存器 L	ADC0	622
AD0TSEL	A/D 转换器 0 触发选择寄存器	ADC0	632
AD1CH1	A/D 转换器 1 通道指定寄存器 1	ADC1	633
AD1CH2	A/D 转换器 1 通道指定寄存器 2	ADC1	635
AD1CHEN	A/D 转换器 1 转换通道指定寄存器	ADC1	625
AD1CHENH	A/D 转换器 1 转换通道指定寄存器 H	ADC1	627
AD1CHENL	A/D 转换器 1 转换通道指定寄存器 L	ADC1	627
AD1CR0	A/D1 转换结果寄存器 0	ADC1	627
AD1CR0H	A/D1 转换结果寄存器 0H	ADC1	627
AD1CR1	A/D1 转换结果寄存器 1	ADC1	627
AD1CR10	A/D1 转换结果寄存器 10	ADC1	627
AD1CR10H	A/D1 转换结果寄存器 10H	ADC1	627
AD1CR11	A/D1 转换结果寄存器 11	ADC1	627
AD1CR11H	A/D1 转换结果寄存器 11H	ADC1	627
AD1CR12	A/D1 转换结果寄存器 12	ADC1	627
AD1CR12H	A/D1 转换结果寄存器 12H	ADC1	627
AD1CR13	A/D1 转换结果寄存器 13	ADC1	627
AD1CR13H	A/D1 转换结果寄存器 13H	ADC1	627
AD1CR14	A/D1 转换结果寄存器 14	ADC1	627
AD1CR14H	A/D1 转换结果寄存器 14H	ADC1	627
AD1CR15	A/D1 转换结果寄存器 15	ADC1	627
AD1CR15H	A/D1 转换结果寄存器 15H	ADC1	627
AD1CR1H	A/D1 转换结果寄存器 1H	ADC1	627
AD1CR2	A/D1 转换结果寄存器 2	ADC1	627
AD1CR2H	A/D1 转换结果寄存器 2H	ADC1	627
AD1CR3	A/D1 转换结果寄存器 3	ADC1	627
AD1CR3H	A/D1 转换结果寄存器 3H	ADC1	627
AD1CR4	A/D1 转换结果寄存器 4	ADC1	627

(3/13)

符号	名称	单元	页码
AD1CR4H	A/D1 转换结果寄存器 4H	ADC1	627
AD1CR5	A/D1 转换结果寄存器 5	ADC1	627
AD1CR5H	A/D1 转换结果寄存器 5H	ADC1	627
AD1CR6	A/D1 转换结果寄存器 6	ADC1	627
AD1CR6H	A/D1 转换结果寄存器 6H	ADC1	627
AD1CR7	A/D1 转换结果寄存器 7	ADC1	627
AD1CR7H	A/D1 转换结果寄存器 7H	ADC1	627
AD1CR8	A/D1 转换结果寄存器 8	ADC1	627
AD1CR8H	A/D1 转换结果寄存器 8H	ADC1	627
AD1CR9	A/D1 转换结果寄存器 9	ADC1	627
AD1CR9H	A/D1 转换结果寄存器 9H	ADC1	627
AD1CTC	A/D 转换器 1 转换时间控制寄存器	ADC1	624
AD1CTL0	A/D 转换器 1 控制寄存器	ADC1	631
AD1ECR0	A/D1 转换结果扩展寄存器 0	ADC1	637
AD1ECR0H	A/D1 转换结果扩展寄存器 0H	ADC1	637
AD1ECR1	A/D1 转换结果扩展寄存器 1	ADC1	637
AD1ECR1H	A/D1 转换结果扩展寄存器 1H	ADC1	637
AD1ECR2	A/D1 转换结果扩展寄存器 2	ADC1	637
AD1ECR2H	A/D1 转换结果扩展寄存器 2H	ADC1	637
AD1ECR3	A/D1 转换结果扩展寄存器 3	ADC1	637
AD1ECR3H	A/D1 转换结果扩展寄存器 3H	ADC1	637
AD1ECR4	A/D1 转换结果扩展寄存器 4	ADC1	637
AD1ECR4H	A/D1 转换结果扩展寄存器 4H	ADC1	637
AD1FLG	A/D 转换器 1 标志寄存器	ADC1	639
AD1FLGB	A/D 转换器 1 标志缓冲寄存器	ADC1	640
AD1IC	中断控制寄存器	ADC1	1000
AD1OCKS	A/D 转换器 1 时钟选择寄存器	ADC1	651
AD1SCM	A/D 转换器 1 扫描模式寄存器	ADC1	622
AD1SCMH	A/D 转换器 1 扫描模式寄存器 H	ADC1	622
AD1SCML	A/D 转换器 1 扫描模式寄存器 L	ADC1	622
AD1TSEL	A/D 转换器 1 触发选择寄存器	ADC1	632
AD2CR0	A/D2 转换结果寄存器 0	ADC2	695
AD2CR0H	A/D2 转换结果寄存器 0H	ADC2	695
AD2CR1	A/D2 转换结果寄存器 1	ADC2	695
AD2CR1H	A/D2 转换结果寄存器 1H	ADC2	695
AD2CR2	A/D2 转换结果寄存器 2	ADC2	695
AD2CR2H	A/D2 转换结果寄存器 2H	ADC2	695
AD2CR3	A/D2 转换结果寄存器 3	ADC2	695
AD2CR3H	A/D2 转换结果寄存器 3H	ADC2	695
AD2CR4	A/D2 转换结果寄存器 4	ADC2	695
AD2CR4H	A/D2 转换结果寄存器 4H	ADC2	695
AD2CR5	A/D2 转换结果寄存器 5	ADC2	695
AD2CR5H	A/D2 转换结果寄存器 5H	ADC2	695
AD2CR6	A/D2 转换结果寄存器 6	ADC2	695

(4/13)

符号	名称	单元	页码
AD2CR6H	A/D2 转换结果寄存器 6H	ADC2	695
AD2CR7	A/D2 转换结果寄存器 7	ADC2	695
AD2CR7H	A/D2 转换结果寄存器 7H	ADC2	695
AD2IC	中断控制寄存器	ADC2	1000
AD2M0	A/D 转换器 2 模式寄存器 0	ADC2	692
AD2M1	A/D 转换器 2 模式寄存器 1	ADC2	693
AD2S	A/D 转换器 2 通道指定寄存器	ADC2	694
ADLTS1	A/DLDTRG1 输入选择寄存器	ADC0	641
ADLTS2	A/DLDTRG2 输入选择寄存器	ADC1	641
ADT0IC	中断控制寄存器	INTC	1000
ADT1IC	中断控制寄存器	INTC	1000
ADTF	A/D 触发下降沿指定寄存器	ADC0	653
ADTR	A/D 触发上升沿指定寄存器	ADC0	653
AWC	地址等待控制寄存器	BCU	943
BCC	总线周期控制寄存器	BCU	946
BCT0	总线周期类型配置寄存器 0	BCU	931
BSC	总线宽度配置寄存器	BCU	933
CB0CTL0	CSIB0 控制寄存器 0	CSIB	812
CB0CTL1	CSIB0 控制寄存器 1	CSIB	815
CB0CTL2	CSIB0 控制寄存器 2	CSIB	816
CB0REIC	中断控制寄存器	INTC	1000
CB0RIC	中断控制寄存器	INTC	1000
CB0RX	CSIB0 接收数据寄存器	CSIB	811
CB0RXL	CSIB0 接收数据寄存器 L	CSIB	811
CB0STR	CSIB0 状态寄存器	CSIB	818
CB0TIC	中断控制寄存器	INTC	992
CB0TX	CSIB0 发送数据寄存器	CSIB	811
CB0TXL	CSIB0 发送数据寄存器 L	CSIB	811
CB1CTL0	CSIB1 控制寄存器 0	CSIB	812
CB1CTL1	CSIB1 控制寄存器 1	CSIB	815
CB1CTL2	CSIB1 控制寄存器 2	CSIB	816
CB1REIC	中断控制寄存器	CSIB	1000
CB1RIC	中断控制寄存器	CSIB	1000
CB1RX	CSIB1 接收数据寄存器	CSIB	811
CB1RXL	CSIB1 接收数据寄存器 L	CSIB	811
CB1STR	CSIB1 状态寄存器	CSIB	818
CB1TIC	中断控制寄存器	CSIB	1000
CB1TX	CSIB1 发送数据寄存器	CSIB	811
CB1TXL	CSIB1 发送数据寄存器 L	CSIB	811
CB2CTL0	CSIB2 控制寄存器 0	CSIB	812
CB2CTL1	CSIB2 控制寄存器 1	CSIB	815
CB2CTL2	CSIB2 控制寄存器 2	CSIB	816
CB2REIC	中断控制寄存器	CSIB	1000
CB2RIC	中断控制寄存器	CSIB	1000

(5/13)

符号	名称	单元	页码
CB2RX	CSIB2 接收数据寄存器	CSIB	811
CB2RXL	CSIB2 接收数据寄存器 L	CSIB	811
CB2STR	CSIB2 状态寄存器	CSIB	818
CB2TIC	中断控制寄存器	CSIB	1000
CB2TX	CSIB2 发送数据寄存器	CSIB	811
CB2TXL	CSIB2 发送数据寄存器 L	CSIB	811
CLM	时钟监视器模式寄存器	CG	183
CMP0CTL0	比较器 0 控制寄存器 0	ADC0	644
CMP0CTL1	比较器 0 控制寄存器 1	ADC0	646
CMP0CTL2	比较器 0 控制寄存器 2	ADC0	648
CMP0CTL3	比较器 0 控制寄存器 3	ADC0	649
CMP1CTL0	比较器 1 控制寄存器 0	ADC0	644
CMP1CTL1	比较器 1 控制寄存器 1	ADC0	646
CMP1CTL2	比较器 1 控制寄存器 2	ADC0	648
CMP1CTL3	比较器 1 控制寄存器 3	ADC0	649
CMPIC0F	中断控制寄存器	INTC	1000
CMPIC0L	中断控制寄存器	INTC	1000
CMPIC1F	中断控制寄存器	INTC	1000
CMPIC1L	中断控制寄存器	INTC	1000
CMPNFC0F	比较器输出数字噪声消除寄存器 0F	ADC0	652
CMPNFC0L	比较器输出数字噪声消除寄存器 0L	ADC0	652
CMPNFC1F	比较器输出数字噪声消除寄存器 1F	ADC0	652
CMPNFC1L	比较器输出数字噪声消除寄存器 1L	ADC0	652
CMPOF	比较器输出中断下降沿指定寄存器	ADC0	654
CMPOR	比较器输出中断上升沿指定寄存器	ADC0	654
DADC0	DMA 寻址控制寄存器 0	DMAC	966
DADC1	DMA 寻址控制寄存器 1	DMAC	966
DADC2	DMA 寻址控制寄存器 2	DMAC	966
DADC3	DMA 寻址控制寄存器 3	DMAC	966
DBC0	DMA 传输计数寄存器 0	DMAC	965
DBC1	DMA 传输计数寄存器 1	DMAC	965
DBC2	DMA 传输计数寄存器 2	DMAC	965
DBC3	DMA 传输计数寄存器 3	DMAC	965
DCHC0	DMA 通道控制寄存器 0	DMAC	967
DCHC1	DMA 通道控制寄存器 1	DMAC	967
DCHC2	DMA 通道控制寄存器 2	DMAC	967
DCHC3	DMA 通道控制寄存器 3	DMAC	967
DDA0H	DMA 目的地址寄存器 0H	DMAC	963
DDA0L	DMA 目的地址寄存器 0L	DMAC	964
DDA1H	DMA 目的地址寄存器 1H	DMAC	963
DDA1L	DMA 目的地址寄存器 1L	DMAC	964
DDA2H	DMA 目的地址寄存器 2H	DMAC	963
DDA2L	DMA 目的地址寄存器 2L	DMAC	964
DDA3H	DMA 目的地址寄存器 3H	DMAC	963

(6/13)

符号	名称	单元	页码
DDA3L	DMA 目的地址寄存器 3L	DMAC	964
DMAIC0	中断控制寄存器	INTC	1000
DMAIC1	中断控制寄存器	INTC	1000
DMAIC2	中断控制寄存器	INTC	1000
DMAIC3	中断控制寄存器	INTC	1000
DSA0H	DMA 源地址寄存器 0H	DMAC	961
DSA0L	DMA 源地址寄存器 0L	DMAC	962
DSA1H	DMA 源地址寄存器 1H	DMAC	961
DSA1L	DMA 源地址寄存器 1L	DMAC	962
DSA2H	DMA 源地址寄存器 2H	DMAC	961
DSA2L	DMA 源地址寄存器 2L	DMAC	962
DSA3H	DMA 源地址寄存器 3H	DMAC	961
DSA3L	DMA 源地址寄存器 3L	DMAC	962
DTFR0	DMA 触发因素寄存器 0	DMAC	969
DTFR1	DMA 触发因素寄存器 1	DMAC	969
DTFR2	DMA 触发因素寄存器 2	DMAC	969
DTFR3	DMA 触发因素寄存器 3	DMAC	969
DVC	总线时钟分频控制寄存器	BCU	947
DWC0	数据等待控制寄存器 0	BCU	941
HZA0CTL0	高阻抗输出控制寄存器 00	定时器	556
HZA0CTL1	高阻抗输出控制寄存器 01	定时器	556
HZA1CTL0	高阻抗输出控制寄存器 10	定时器	556
HZA1CTL1	高阻抗输出控制寄存器 11	定时器	556
HZA2CTL0	高阻抗输出控制寄存器 20	定时器	556
HZA2CTL1	高阻抗输出控制寄存器 21	定时器	556
HZA3CTL0	高阻抗输出控制寄存器 30	定时器	556
HZA3CTL1	高阻抗输出控制寄存器 31	定时器	556
IIC0	IIC 移位寄存器 0	I^2C	857
IICC0	IIC 控制寄存器 0	I^2C	860
IICCL0	IIC 时钟选择寄存器 0	I^2C	869
IICF0	IIC 标志寄存器 0	I^2C	867
IICIC	中断控制寄存器	INTC	1000
IICOCKS	IICOPS 时钟选择寄存器	I^2C	870
IICS0	IIC 状态寄存器 0	I^2C	864
IICX0	IIC 功能扩展寄存器 0	I^2C	870
IMR0	中断屏蔽寄存器 0	INTC	1005
IMR0H	中断屏蔽寄存器 0H	INTC	1005
IMR0L	中断屏蔽寄存器 0L	INTC	1005
IMR1	中断屏蔽寄存器 1	INTC	1005
IMR1H	中断屏蔽寄存器 1H	INTC	1005
IMR1L	中断屏蔽寄存器 1L	INTC	1005
IMR2	中断屏蔽寄存器 2	INTC	1005
IMR2H	中断屏蔽寄存器 2H	INTC	1005
IMR2L	中断屏蔽寄存器 2L	INTC	1005

(7/13)

符号	名称	单元	页码
IMR3	中断屏蔽寄存器 3	INTC	1005
IMR3H	中断屏蔽寄存器 3H	INTC	1005
IMR3L	中断屏蔽寄存器 3L	INTC	1005
IMR4	中断屏蔽寄存器 4	INTC	1005
IMR4H	中断屏蔽寄存器 4H	INTC	1005
IMR4L	中断屏蔽寄存器 4L	INTC	1005
IMR5	中断屏蔽寄存器 5	INTC	1005
IMR5H	中断屏蔽寄存器 5H	INTC	1005
IMR5L	中断屏蔽寄存器 5L	INTC	1005
INTF0	外部中断下降沿指定寄存器 0	INTC	1011
INTF1	外部中断下降沿指定寄存器 1	INTC	1012
INTF2	外部中断下降沿指定寄存器 2	INTC	1013
INTNFC14	数字噪声消除 0 控制寄存器 14	端口	169
INTNFC15	数字噪声消除 0 控制寄存器 15	端口	169
INTNFC16	数字噪声消除 0 控制寄存器 16	端口	169
INTR0	外部中断上升沿指定寄存器 0	INTC	1011
INTR1	外部中断上升沿指定寄存器 1	INTC	1012
INTR2	外部中断上升沿指定寄存器 2	INTC	1013
ISPR	正在服务的中断优先级寄存器	INTC	1008
LVIHIC	中断控制寄存器	INTC	1000
LVILIC	中断控制寄存器	INTC	1000
LVIM	低电压检测寄存器	LVI	1042
LVIS	低电压检测电平选择寄存器	LVI	1043
OP0CTL0	运算放大器 0 控制寄存器 0	ADC0	642
OP1CTL0	运算放大器 1 控制寄存器 0	ADC0	642
OSTS	振荡稳定时间选择寄存器	CG	182
P0	端口 0 寄存器	端口	108
P1	端口 1 寄存器	端口	114
P2	端口 2 寄存器	端口	120
P3	端口 3 寄存器	端口	126
P4	端口 4 寄存器	端口	132
P7	端口 7 寄存器	端口	138
PCC	处理器时钟控制寄存器	CG	179
PDL	DL 端口寄存器	端口	140
PDLH	DLH 端口寄存器	端口	140
PDLL	DLL 端口寄存器	端口	140
PF3	端口 3 功能寄存器	端口	130
PFC0	端口 0 功能控制寄存器	端口	110
PFC1	端口 1 功能控制寄存器	端口	116
PFC2	端口 2 功能控制寄存器	端口	122
PFC3	端口 3 功能控制寄存器	端口	128
PFC4	端口 4 功能控制寄存器	端口	134
PFCE0	端口 0 功能控制扩展寄存器	端口	110
PFCE1	端口 1 功能控制扩展寄存器	端口	116

(8/13)

符号	名称	单元	页码
PFCE2	端口 2 功能控制扩展寄存器	端口	122
PFCE3	端口 3 功能控制扩展寄存器	端口	128
PFCE4	端口 4 功能控制扩展寄存器	端口	134
PIC00	中断控制寄存器	INTC	1000
PIC01	中断控制寄存器	INTC	1000
PIC02	中断控制寄存器	INTC	1000
PIC03	中断控制寄存器	INTC	1000
PIC04	中断控制寄存器	INTC	1000
PIC05	中断控制寄存器	INTC	1000
PIC06	中断控制寄存器	INTC	1000
PIC07	中断控制寄存器	INTC	1000
PIC08	中断控制寄存器	INTC	1000
PIC09	中断控制寄存器	INTC	1000
PIC10	中断控制寄存器	INTC	1000
PIC11	中断控制寄存器	INTC	1000
PIC12	中断控制寄存器	INTC	1000
PIC13	中断控制寄存器	INTC	1000
PIC14	中断控制寄存器	INTC	1000
PIC15	中断控制寄存器	INTC	1000
PIC16	中断控制寄存器	INTC	1000
PIC17	中断控制寄存器	INTC	1000
PIC18	中断控制寄存器	INTC	1000
PLLCTL	PLL 控制寄存器	CG	178
PM0	端口 0 模式寄存器	端口	108
PM1	端口 1 模式寄存器	端口	114
PM2	端口 2 模式寄存器	端口	120
PM3	端口 3 模式寄存器	端口	126
PM4	端口 4 模式寄存器	端口	132
PMC0	端口 0 模式控制寄存器	端口	109
PMC1	端口 1 模式控制寄存器	端口	115
PMC2	端口 2 模式控制寄存器	端口	121
PMC3	端口 3 模式控制寄存器	端口	127
PMC4	端口 4 模式控制寄存器	端口	133
PMC7	端口 7 模式控制寄存器	端口	138
PMCDL	DL 端口模式控制寄存器	端口	142
PMCDLH	DL 端口模式控制寄存器 H	端口	142
PMCDLL	DL 端口模式控制寄存器 L	端口	142
PMDL	DL 端口模式寄存器	端口	141
PMDLH	DL 端口模式寄存器 H	端口	141
PMDLL	DL 端口模式寄存器 L	端口	141
PRCMD	命令寄存器	CPU	95
PSC	省电控制寄存器	CPU	180, 1027
PSMR	省电模式寄存器	CPU	181, 1028
PU0	上拉电阻选项寄存器 0	端口	112

(9/13)

符号	名称	单元	页码
PU1	上拉电阻选项寄存器 1	端口	118
PU2	上拉电阻选项寄存器 2	端口	124
PU3	上拉电阻选项寄存器 3	端口	130
PU4	上拉电阻选项寄存器 4	端口	136
PUDL	上拉电阻选项寄存器 DL	端口	143
PUDLH	上拉电阻选项寄存器 DLH	端口	143
PUDLL	上拉电阻选项寄存器 DLL	端口	143
RESF	复位源标志寄存器	复位	1037
SVA0	从地址寄存器 0	I^2C	857
SYS	系统状态寄存器	CPU	96
TA0CCIC0	中断控制寄存器	INTC	1000
TA0CCIC1	中断控制寄存器	INTC	1000
TA0OVIC	中断控制寄存器	INTC	1000
TA1CCIC0	中断控制寄存器	INTC	1000
TA1CCIC1	中断控制寄存器	INTC	1000
TA1OVIC	中断控制寄存器	INTC	1000
TA2CCIC0	中断控制寄存器	INTC	1000
TA2CCIC1	中断控制寄存器	INTC	1000
TA2OVIC	中断控制寄存器	INTC	1000
TA3CCIC0	中断控制寄存器	INTC	1000
TA3CCIC1	中断控制寄存器	INTC	1000
TA3OVIC	中断控制寄存器	INTC	1000
TA4CCIC0	中断控制寄存器	INTC	1000
TA4CCIC1	中断控制寄存器	INTC	1000
TA4OVIC	中断控制寄存器	INTC	1000
TAA0CCR0	TAA0 捕获/比较寄存器 0	TAA	207
TAA0CCR1	TAA0 捕获/比较寄存器 1	TAA	209
TAA0CNT	TAA0 计数器读取缓冲寄存器	TAA	211
TAA0CTL0	TAA0 控制寄存器 0	TAA	199
TAA0CTL1	TAA0 控制寄存器 1	TAA	200
TAA0OPT0	TAA0 选项寄存器 0	TAA	206
TAA1CCR0	TAA1 捕获/比较寄存器 0	TAA	207
TAA1CCR1	TAA1 捕获/比较寄存器 1	TAA	209
TAA1CNT	TAA1 计数器读取缓冲寄存器	TAA	211
TAA1CTL0	TAA1 控制寄存器 0	TAA	199
TAA1CTL1	TAA1 控制寄存器 1	TAA	200
TAA1OPT0	TAA1 选项寄存器 0	TAA	206
TAA2CCR0	TAA2 捕获/比较寄存器 0	TAA	207
TAA2CCR1	TAA2 捕获/比较寄存器 1	TAA	209
TAA2CNT	TAA2 计数器读取缓冲寄存器	TAA	211
TAA2CTL0	TAA2 控制寄存器 0	TAA	199
TAA2CTL1	TAA2 控制寄存器 1	TAA	200
TAA2IOC0	TAA2 I/O 控制寄存器 0	TAA	202
TAA2IOC1	TAA2 I/O 控制寄存器 1	TAA	204

(10/13)

符号	名称	单元	页码
TAA2IOC2	TAA2 I/O 控制寄存器 2	TAA	205
TAA2OPT0	TAA2 选项寄存器 0	TAA	206
TAA3CCR0	TAA3 捕获/比较寄存器 0	TAA	207
TAA3CCR1	TAA3 捕获/比较寄存器 1	TAA	209
TAA3CNT	TAA3 计数器读取缓冲寄存器	TAA	211
TAA3CTL0	TAA3 控制寄存器 0	TAA	199
TAA3CTL1	TAA3 控制寄存器 1	TAA	200
TAA3IOC0	TAA3 I/O 控制寄存器 0	TAA	202
TAA3IOC1	TAA3 I/O 控制寄存器 1	TAA	204
TAA3IOC2	TAA3 I/O 控制寄存器 2	TAA	205
TAA3OPT0	TAA3 选项寄存器 0	TAA	206
TAA4CCR0	TAA4 捕获/比较寄存器 0	TAA	207
TAA4CCR1	TAA4 捕获/比较寄存器 1	TAA	209
TAA4CNT	TAA4 计数器读取缓冲寄存器	TAA	211
TAA4CTL0	TAA4 控制寄存器 0	TAA	199
TAA4CTL1	TAA4 控制寄存器 1	TAA	200
TAA4IOC0	TAA4 I/O 控制寄存器 0	TAA	202
TAA4IOC1	TAA4 I/O 控制寄存器 1	TAA	204
TAA4IOC2	TAA4 I/O 控制寄存器 2	TAA	205
TAA4OPT0	TAA4 选项寄存器 0	TAA	206
TAB0CCR0	TAB0 捕获/比较寄存器 0	TAB	303
TAB0CCR1	TAB0 捕获/比较寄存器 1	TAB	305
TAB0CCR2	TAB0 捕获/比较寄存器 2	TAB	307
TAB0CCR3	TAB0 捕获/比较寄存器 3	TAB	309
TAB0CNT	TAB0 计数器读取缓冲寄存器	TAB	311
TAB0CTL0	TAB0 控制寄存器 0	TAB	297
TAB0CTL1	TAB0 控制寄存器 1	TAB	298
TAB0DTC	TAB0 死区时间比较寄存器	定时器	547
TAB0IOC0	TAB0 I/O 控制寄存器 0	TAB	299
TAB0IOC1	TAB0 I/O 控制寄存器 1	TAB	300
TAB0IOC2	TAB0 I/O 控制寄存器 2	TAB	301
TAB0IOC3	TAB0 I/O 控制寄存器 3	定时器	553
TAB0OPT0	TAB0 选项寄存器 0	TAB	302, 549
TAB0OPT1	TAB0 选项寄存器 1	定时器	549
TAB0OPT2	TAB0 选项寄存器 2	定时器	550
TAB0OPT3	TAB0 选项寄存器 3	定时器	552
TAB1CCR0	TAB1 捕获/比较寄存器 0	TAB	303
TAB1CCR1	TAB1 捕获/比较寄存器 1	TAB	305
TAB1CCR2	TAB1 捕获/比较寄存器 2	TAB	307
TAB1CCR3	TAB1 捕获/比较寄存器 3	TAB	309
TAB1CNT	TAB1 计数器读取缓冲寄存器	TAB	311
TAB1CTL0	TAB1 控制寄存器 0	TAB	297
TAB1CTL1	TAB1 控制寄存器 1	TAB	298
TAB1DTC	TAB0 死区时间比较寄存器	TAB	547

(11/13)

(12/13)

符号	名称	单元	页码
TT0IOC2	TMT0 I/O 控制寄存器 2	TMT	416
TT0IOC3	TMT0 I/O 控制寄存器 3	TMT	417
TT0OPT0	TMT0 选项寄存器 0	TMT	419
TT0OPT1	TMT0 选项寄存器 1	TMT	420
TT0OVIC	中断控制寄存器	INTC	1000
TT0TCW	TMT0 计数器写入寄存器	TMT	427
TT1CCIC0	中断控制寄存器	INTC	1000
TT1CCIC1	中断控制寄存器	INTC	1000
TT1CCR0	TMT1 捕获/比较寄存器 0	TMT	423
TT1CCR1	TMT1 捕获/比较寄存器 1	TMT	425
TT1CNT	TMT1 计数器读取缓冲寄存器	TMT	427
TT1CTL0	TMT1 控制寄存器 0	TMT	408
TT1CTL1	TMT1 控制寄存器 1	TMT	409
TT1CTL2	TMT1 控制寄存器 2	TMT	411
TT1IECIC	中断控制寄存器	INTC	1000
TT1IOC0	TMT1 I/O 控制寄存器 0	TMT	413
TT1IOC1	TMT1 I/O 控制寄存器 1	TMT	415
TT1IOC2	TMT1 I/O 控制寄存器 2	TMT	416
TT1IOC3	TMT1 I/O 控制寄存器 3	TMT	417
TT1OPT0	TMT1 选项寄存器 0	TMT	419
TT1OPT1	TMT1 选项寄存器 1	TMT	420
TT1OVIC	中断控制寄存器	INTC	1000
TT1TCW	TMT1 计数器写入寄存器	TMT	427
TTISL0	TMT0 捕获输入选择寄存器	TMT	422
TTISL1	TMT1 捕获输入选择寄存器	TMT	422
TTNFC0	数字噪声消除 2 控制寄存器 0	端口	171
TTNFC1	数字噪声消除 2 控制寄存器 1	端口	171
UA0CTL0	UARTA0 控制寄存器 0	UARTA	723
UA0CTL1	UARTA0 控制寄存器 1	UARTA	739
UA0CTL2	UARTA0 控制寄存器 2	UARTA	740
UA0OPT0	UARTA0 选项控制寄存器 0	UARTA	725
UA0REIC	中断控制寄存器	INTC	1000
UA0RIC	中断控制寄存器	INTC	1000
UA0RX	UARTA0 接收数据寄存器	UARTA	727
UA0STR	UARTA0 状态寄存器	UARTA	725
UA0TIC	中断控制寄存器	INTC	1000
UA0TX	UARTA0 发送数据寄存器	UARTA	727
UA1CTL0	UARTA1 控制寄存器 0	UARTA	723
UA1CTL1	UARTA1 控制寄存器 1	UARTA	739
UA1CTL2	UARTA1 控制寄存器 2	UARTA	740
UA1OPT0	UARTA1 选项控制寄存器 0	UARTA	725
UA1REIC	中断控制寄存器	INTC	1000
UA1RIC	中断控制寄存器	INTC	1000
UA1RX	UARTA1 接收数据寄存器	UARTA	727

(13/13)

符号	名称	单元	页码
UA1STR	UARTA1 状态寄存器	UARTA	725
UA1TIC	中断控制寄存器	INTC	1000
UA1TX	UARTA1 发送数据寄存器	UARTA	727
UA2CTL0	UARTA2 控制寄存器 0	UARTA	723
UA2CTL1	UARTA2 控制寄存器 1	UARTA	739
UA2CTL2	UARTA2 控制寄存器 2	UARTA	740
UA2OPT0	UARTA2 选项控制寄存器 0	UARTA	725
UA2REIC	中断控制寄存器	INTC	1000
UA2RIC	中断控制寄存器	INTC	1000
UA2RX	UARTA2 接收数据寄存器	UARTA	727
UA2STR	UARTA2 状态寄存器	UARTA	725
UA2TIC	中断控制寄存器	INTC	1000
UA2TX	UARTA2 发送数据寄存器	UARTA	727
UBCTL0	UARTB 控制寄存器 0	UARTB	749
UBCTL2	UARTB 控制寄存器 2	UARTB	757
UBFIC0	UARTB FIFO 控制寄存器 0	UARTB	761
UBFIC1	UARTB FIFO 控制寄存器 1	UARTB	763
UBFIC2	UARTB FIFO 控制寄存器 2	UARTB	764
UBFIC2H	UARTB FIFO 控制寄存器 2H	UARTB	764
UBFIC2L	UARTB FIFO 控制寄存器 2L	UARTB	764
UBFIS0	UARTB FIFO 控制寄存器 0	UARTB	766
UBFIS1	UARTB FIFO 控制寄存器 1	UARTB	767
UBRX	UARTB 接收数据寄存器	UARTB	759
UBRXAP	UARTB 接收数据寄存器 AP	UARTB	759
UBSTR	UARTB 状态寄存器	UARTB	755
UBTX	UARTB 发送数据寄存器	UARTB	758
UIFIC	中断控制寄存器	INTC	1000
UREIC	中断控制寄存器	INTC	1000
URIC	中断控制寄存器	INTC	1000
UTIC	中断控制寄存器	INTC	1000
UTOIC	中断控制寄存器	INTC	1000
VSWC	系统等待控制寄存器	BCU	97
WDTE	看门狗定时器使能寄存器	WDT	609
WDTM	看门狗定时器模式寄存器	WDT	608

附录 C 指令集列表

C.1 习惯用法

(1) 用于描述操作数的寄存器符号

寄存器符号	解释
reg1	通用寄存器：用作源寄存器。
reg2	通用寄存器：主要用作目的寄存器，在某指令里也用作源寄存器。
reg3	通用寄存器：主要用作存储除法结果的余数，以及乘法结果的高 32 位。
bit#3	指明位号的 3 位数据。
immX	X 位立即数。
dispX	X 位置换数据。
regID	系统寄存器号。
vector	用来指明陷阱向量的 5 位数据 (00H 至 1FH)。
cccc	用来显示状态码的 4 位数据。
sp	堆栈指针 (SP)。
ep	元指针 (r30)。
listX	X 项寄存器列表。

(2) 用于描述操作码的寄存器符号

寄存器符号	解释
R	指定 reg1 或 regID 的 1 位代码数据。
r	指定 reg2 的 1 位代码数据。
w	指定 reg3 的 1 位代码数据。
d	1 位置换数据。
I	1 位立即数 (指示立即数的高位)。
i	1 位立即数。
cccc	显示条件码的 4 位数据。
CCCC	显示 Bcond 指令条件码的 4 位数据。
bbb	指定位号的 3 位数据。
L	在寄存器列表中指示程序寄存器的 1 位数据。
S	在寄存器列表中指示系统寄存器的 1 位数据

(3) 用于操作的寄存器符号

寄存器符号	解释
←	为之输入
GR [ ]	通用寄存器
SR [ ]	系统寄存器
zero-extend (n)	用零扩展 n 直至字长。
sign-extend (n)	用符号扩展 n 直至字长。
load-memory (a, b)	从地址 a 起读取存储长度为 b 的数据。
store-memory (a, b, c)	向地址 c 起存储长度为 a 的字节写入数据 b。
load-memory-bit (a, b)	从地址 a 载入位 b 。
store-memory-bit (a, b, c)	写入 c 至地址 a 的 b 位。
saturated (n)	执行 n 的饱和运算 (n 为 2 的补码)。 如此的话，结果如下处理： n ≥ 7FFFFFFFH, 令其为 7FFFFFFFH。 n ≤ 80000000H, 令其为 80000000H。
result	反映标志的结果。
Byte	字节 (8 位)
Halfword	半字 (16 位)
Word	字(32 位)
+	加
–	减
ll	位并接
×	乘
÷	除
%	除法结果的余数
AND	逻辑乘
OR	逻辑加
XOR	逻辑异或
NOT	逻辑非
logically shift left by	逻辑左移
logically shift right by	逻辑右移
arithmetically shift right by	算术右移

(4) 用于执行时钟的寄存器符号

寄存器符号	解释
i	如果执行第一条指令后紧接着执行另一条指令 (流式执行)。
r	如果执行第一条指令后紧接着重复执行相同指令(重复执行)。
l	如果紧接之后的指令执行使用当前指令执行结果 (等待执行)。

(5) 用于标志操作的寄存器符号

标识符	说明
(Blank)	无改变。
0	清为 0。
X	根据结果置位或清 0。
R	恢复先前的保存值。

(6) 条件码

条件码 (cond)	条件公式 (cccc)	条件公式	说明
V	0 0 0 0	OV = 1	溢出
NV	1 0 0 0	OV = 0	无溢出
C/L	0 0 0 1	CY = 1	进位 低(小于)
NC/NL	1 0 0 1	CY = 0	无进位 不小(大于或等于)
Z/E	0 0 1 0	Z = 1	0
NZ/NE	1 0 1 0	Z = 0	非 0
NH	0 0 1 1	(CY or Z) = 1	不高 (小于或等于)
H	1 0 1 1	(CY or Z) = 0	高 (相比大于)
N	0 1 0 0	S = 1	负
P	1 1 0 0	S = 0	正
T	0 1 0 1	–	恒为 (无条件的)
SA	1 1 0 1	SAT = 1	饱和
LT	0 1 1 0	(S xor OV) = 1	有符号数小于。
GE	1 1 1 0	(S xor OV) = 0	有符号数大于或等于。
LE	0 1 1 1	((S xor OV) or Z) = 1	有符号数小于或等于。
GT	1 1 1 1	((S xor OV) or Z) = 0	有符号数大于。

C.2 指令集 (按字母先后次序)

(1/6)

助记符	操作数	操作码	操作	执行时钟			标志				
				i	r	l	CY	OV	S	Z	SAT
ADD	reg1,reg2	rrrrr001110RRRRR	GR[reg2]←GR[reg2]+GR[reg1]	1	1	1	×	×	×	×	
	imm5,reg2	rrrrr010010iiiii	GR[reg2]←GR[reg2]+sign-extend(imm5)	1	1	1	×	×	×	×	
ADDI	imm16,reg1,reg2	rrrrr110000RRRRR iiiiiiiiiiiiiiii	GR[reg2]←GR[reg1]+sign-extend(imm16)	1	1	1	×	×	×	×	
AND	reg1,reg2	rrrrr001010RRRRR	GR[reg2]←GR[reg2]AND GR[reg1]	1	1	1		0	×	×	
ANDI	imm16,reg1,reg2	rrrrr110110RRRRR iiiiiiiiiiiiiiii	GR[reg2]←GR[reg1]AND zero-extend(imm16)	1	1	1		0	0	×	
Bcond	disp9	ddddd1011dddcccc **注 1**	如果条件满足 则： PC←PC+sign-extend(disp9) 如果条件满足	3 注2	3 注2	3 注2					
			如果条件不满足	1	1	1					
BSH	reg2,reg3	rrrrr11111100000 wwwww01101000010	GR[reg3]←GR[reg2] (23 : 16) ll GR[reg2] (31 : 24) ll GR[reg2] (7 : 0) ll GR[reg2] (15 : 8)	1	1	1	×	0	×	×	
BSW	reg2,reg3	rrrrr11111100000 wwwww01101000000	GR[reg3]←GR[reg2] (7 : 0) ll GR[reg2] (15 : 8) ll GR [reg2] (23 : 16) ll GR[reg2] (31 : 24)	1	1	1	×	0	×	×	
CALLT	imm6	0000001000iiiiii	CTPC←PC+2(return PC) CTPSW←PSW adr←CTBP+zero-extend(imm6 logically shift left by 1) PC←CTBP+zero-extend(Load-memory(adr,Halfword))	5	5	5					
CLR1	bit#3,disp16[reg1]	10bbb111110RRRRR dddddddddddddddd	adr←GR[reg1]+sign-extend(disp16) Z flag←Not(Load-memory-bit(adr,bit#3)) Store-memory-bit(adr,bit#3,0)	3 注3	3 注3	3 注3				×	
	reg2,[reg1]	rrrrr111111RRRRR 0000000011100100	adr←GR[reg1] Z flag←Not(Load-memory-bit(adr,reg2)) Store-memory-bit(adr,reg2,0)	3 注3	3 注3	3 注3				×	
CMOV	cccc,imm5,reg2,reg3	rrrrr111111iiiii wwwww011000cccc0	如果条件满足 则 GR[reg3]←sign-extended(imm5) 否则 GR[reg3]←GR[reg2]	1	1	1					
	cccc,reg1,reg2,reg3	rrrrr111111RRRR wwwww011001cccc0	如果条件满足 则 GR[reg3]←GR[reg1] 否则 GR[reg3]←GR[reg2]	1	1	1					
CMP	reg1,reg2	rrrrr001111RRRRR	result←GR[reg2]–GR[reg1]	1	1	1	×	×	×	×	
	imm5,reg2	rrrrr010011iiiii	result←GR[reg2]–sign-extend(imm5)	1	1	1	×	×	×	×	
CTRET		0000011111100000 0000000101000100	PC←CTPC PSW←CTPSW	4	4	4	R	R	R	R	R
DBRET		0000011111100000 0000000101000110	PC←DBPC PSW←DBPSW	4	4	4	R	R	R	R	R

(2/6)

助记符	操作数	操作码	操作	执行时钟			标志				
				i	r	l	CY	OV	S	Z	SAT
DBTRAP		1111100001000000	DBPC←PC+2 (return PC) DBPSW←PSW PSW.NP←1 PSW.EP←1 PSW.ID←1 PC←00000060H	4	4	4					
DI		0000011111100000 0000000101100000	PSW.ID←1	1	1	1					
DISPOSE	imm5,list12	0000011001iiiiiL LLLLLLLLLLL00000	sp←sp+zero-extend(imm5 logically shift left by 2) GR[reg in list12]←Load-memory(sp,Word) sp←sp+4 重复以上两个步骤直到表 12 中所有寄存器被载入。	n+1 注4	n+1 注4	n+1 注4					
	imm5,list12,[reg1]	0000011001iiiiiL LLLLLLLLLLLRRRRR 注 5	sp←sp+zero-extend(imm5 logically shift left by 2) GR[reg in list12]←Load-memory(sp,Word) sp←sp+4 重复以上两个步骤直到表 12 中所有寄存器载入 PC←GR[reg1]	n+3 注4	n+3 注4	n+3 注4					
DIV	reg1,reg2,reg3	rrrrr111111RRRRR wwwww01011000000	GR[reg2]←GR[reg2]÷GR[reg1] GR[reg3]←GR[reg2]%GR[reg1]	35	35	35		×	×	×	
DIVH	reg1,reg2	rrrrr000010RRRRR	GR[reg2]←GR[reg2]÷GR[reg1][注 6]	35	35	35		×	×	×	
	reg1,reg2,reg3	rrrrr111111RRRRR wwwww01010000000	GR[reg2]←GR[reg2]÷GR[reg1][注 6] GR[reg3]←GR[reg2]%GR[reg1]	35	35	35		×	×	×	
DIVHU	reg1,reg2,reg3	rrrrr111111RRRRR wwwww01010000010	GR[reg2]←GR[reg2]÷GR[reg1][注 6] GR[reg3]←GR[reg2]%GR[reg1]	34	34	34		×	×	×	
DIVU	reg1,reg2,reg3	rrrrr111111RRRRR wwwww01011000010	GR[reg2]←GR[reg2]÷GR[reg1] GR[reg3]←GR[reg2]%GR[reg1]	34	34	34		×	×	×	
EI		1000011111100000 0000000101100000	PSW.ID←0	1	1	1					
HALT		0000011111100000 0000000100100000	停止	1	1	1					
HSW	reg2,reg3	rrrrr11111100000 wwwww01101000100	GR[reg3]←GR[reg2](15 : 0) ll GR[reg2] (31 : 16)	1	1	1	×	0	×	×	
JARL	disp22,reg2	rrrrr11110dddddd ddddddddddddddd0 注 7	GR[reg2]←PC+4 PC←PC+sign-extend(disp22)	3	3	3					
JMP	[reg1]	00000000011RRRRR	PC←GR[reg1]	4	4	4					
JR	disp22	0000011110dddddd ddddddddddddddd0 注 7	PC←PC+sign-extend(disp22)	3	3	3					
LD.B	disp16[reg1],reg2	rrrrr111000RRRRR dddddddddddddddd	adr←GR[reg1]+sign-extend(disp16) GR[reg2]←sign-extend(Load-memory(adr,Byte))	1	1	注 11					
LD.BU	disp16[reg1],reg2	rrrrr11110bRRRRR ddddddddddddddd1 注 8, 10	adr←GR[reg1]+sign-extend(disp16) GR[reg2]←zero-extend(Load-memory(adr,Byte))	1	1	注 11					

(3/6)

助记符	操作数	操作码	操作		执行时钟			标志				
					i	r	l	CY	OV	S	Z	SAT
LD.H	disp16[reg1],reg2	rrrrr111001RRRRR ddddddddddddddd0 注 8	adr←GR[reg1]+sign-extend(disp16) GR[reg2]←sign-extend(Load-memory(adr,Halfword))		1	1	注 11					
LDSR	reg2,regID	rrrrr111111RRRRR 0000000000100000 注 12	SR[regID]←GR[reg2]	除非 regID = PSW	1	1	1					
				regID = PSW	1	1	1	×	×	×	×	×
LD.HU	disp16[reg1],reg2	rrrrr111111RRRRR ddddddddddddddd1 注 8	adr←GR[reg1]+sign-extend(disp16) GR[reg2]←zero-extend(Load-memory(adr,Halfword))		1	1	注 11					
LD.W	disp16[reg1],reg2	rrrrr111001RRRRR ddddddddddddddd1 注 8	adr←GR[reg1]+sign-extend(disp16) GR[reg2]←Load-memory(adr,Word)		1	1	注 11					
MOV	reg1,reg2	rrrrr000000RRRRR	GR[reg2]←GR[reg1]		1	1	1					
	imm5,reg2	rrrrr010000iiiii	GR[reg2]←sign-extend(imm5)		1	1	1					
	imm32,reg1	00000110001RRRRR iiiiiiiiiiiiiiii IIIIIIIIIIIIIIII	GR[reg1]←imm32		2	2	2					
MOVEA	imm16,reg1,reg2	rrrrr110001RRRRR iiiiiiiiiiiiiiii	GR[reg2]←GR[reg1]+sign-extend(imm16)		1	1	1					
MOVHI	imm16,reg1,reg2	rrrrr110010RRRRR iiiiiiiiiiiiiiii	GR[reg2]←GR[reg1]+(imm16 ll 0^{16})		1	1	1					
MUL[注 22]	reg1,reg2,reg3	rrrrr111111RRRRR wwwww01000100000	GR[reg3] ll GR[reg2]←GR[reg2]xGR[reg1]		1	2 注 14	2					
	imm9,reg2,reg3	rrrrr111111iiiii wwwww01001IIII00 注 13	GR[reg3] ll GR[reg2]←GR[reg2]xsign-extend(imm9)		1	2 注 14	2					
MULH	reg1,reg2	rrrrr000111RRRRR	GR[reg2]←GR[reg2][注 6]xGR[reg1][注 6]		1	1	2					
	imm5,reg2	rrrrr010111iiiii	GR[reg2]←GR[reg2][注 6]xsign-extend(imm5)		1	1	2					
MULHI	imm16,reg1,reg2	rrrrr110111RRRRR iiiiiiiiiiiiiiii	GR[reg2]←GR[reg1][注 6]ximm16		1	1	2					
MULU[注 22]	reg1,reg2,reg3	rrrrr111111RRRRR wwwww01000100010	GR[reg3] ll GR[reg2]←GR[reg2]xGR[reg1]		1	2 注 14	2					
	imm9,reg2,reg3	rrrrr111111iiiii wwwww01001IIII10 注 13	GR[reg3] ll GR[reg2]←GR[reg2]xzero-extend(imm9)		1	2 注 14	2					
NOP		0000000000000000	至少一个时钟周期内不作任何操作。		1	1	1					
NOT	reg1,reg2	rrrrr000001RRRRR	GR[reg2]←NOT(GR[reg1])		1	1	1		0	×	×	
NOT1	bit#3,disp16[reg1]	01bbb111110RRRRR dddddddddddddddd	adr←GR[reg1]+sign-extend(disp16) Z flag←Not(Load-memory-bit(adr,bit#3)) Store-memory-bit(adr,bit#3,Z flag)		3 注 3	3 注 3	3 注 3				×	
	reg2,[reg1]	rrrrr111111RRRRR 0000000011100010	adr←GR[reg1] Z flag←Not(Load-memory-bit(adr,reg2)) Store-memory-bit(adr,reg2,Z flag)		3 注 3	3 注 3	3 注 3				×	

(4/6)

助记符	操作数	操作码	操作	执行时钟			标志				
				i	r	l	CY	OV	S	Z	SAT
OR	reg1,reg2	rrrrr001000RRRRR	GR[reg2]←GR[reg2]OR GR[reg1]	1	1	1		0	×	×	
ORI	imm16,reg1,reg2	rrrrr110100RRRRR iiiiiiiiiiiiiiii	GR[reg2]←GR[reg1]OR zero-extend(imm16)	1	1	1		0	×	×	
PREPARE	list12,imm5	0000011110iiiiiL LLLLLLLLLLL00001	Store-memory(sp–4,GR[reg in list12],Word) sp←sp–4 repeat 1 step above until all regs in list12 is stored sp←sp-zero-extend(imm5)	n+1 注4	n+1 注4	n+1 注4					
	list12,imm5, sp/imm注 15	0000011110iiiiiL LLLLLLLLLLLff011 imm16/imm32 注 16	Store-memory(sp–4,GR[reg in list12],Word) GR[reg in list 12]←Load-memory(sp,Word) sp←sp+4 repeat 2 step above until all regs in list12 is loaded PC←GR[reg1]	n+2 注4 注17	n+2 注4 注17	n+2 注4 注17					
RETI		0000011111100000 0000000101000000	如果 PSW.EP=1 那么 PC ←EIPC PSW ←EIPSW 否则，如果 PSW.NP=1 则 PC ←FEPC PSW ←FEPSW 否则 PC ←EIPC PSW ←EIPSW	4	4	4	R	R	R	R	R
SAR	reg1,reg2	rrrrr111111RRRRR 0000000010100000	GR[reg2]←GR[reg2]arithmetically shift right by GR[reg1]	1	1	1	×	0	×	×	
	imm5,reg2	rrrrr010101iiiii	GR[reg2]←GR[reg2]arithmetically shift right by zero-extend (imm5)	1	1	1	×	0	×	×	
SASF	cccc,reg2	rrrrr1111110cccc 0000001000000000	如果条件满足 则 GR[reg2]←(GR[reg2]Logically shift left by 1) OR 00000001H 否则 GR[reg2]←(GR[reg2]Logically shift left by 1) OR 00000000H	1	1	1					
SATADD	reg1,reg2	rrrrr000110RRRRR	GR[reg2]←saturated(GR[reg2]+GR[reg1])	1	1	1	×	×	×	×	×
	imm5,reg2	rrrrr010001iiiii	GR[reg2]←saturated(GR[reg2]+sign-extend(imm5)	1	1	1	×	×	×	×	×
SATSUB	reg1,reg2	rrrrr000101RRRRR	GR[reg2]←saturated(GR[reg2]–GR[reg1])	1	1	1	×	×	×	×	×
SATSUBI	imm16,reg1,reg2	rrrrr110011RRRRR iiiiiiiiiiiiiiii	GR[reg2]←saturated(GR[reg1]–sign-extend(imm16)	1	1	1	×	×	×	×	×
SATSUBR	reg1,reg2	rrrrr000100RRRRR	GR[reg2]←saturated(GR[reg1]–GR[reg2])	1	1	1	×	×	×	×	×
SETF	cccc,reg2	rrrrr1111110cccc 0000000000000000	如果条件满足 则 GR[reg2]←00000001H 否则 GR[reg2]←00000000H	1	1	1					

(5/6)

助记符	操作数	操作码	操作	执行时钟			标志				
				i	r	l	CY	OV	S	Z	SAT
SET1	bit#3,disp16[reg1]	00bbb111110RRRRR dddddddddddddddd	adr←GR[reg1]+sign-extend(disp16) Z flag←Not (Load-memory-bit(adr,bit#3)) Store-memory-bit(adr,bit#3,1)	3 注3	3 注3	3 注3				×	
	reg2,[reg1]	rrrrr111111RRRRR 0000000011100000	adr←GR[reg1] Z flag←Not(Load-memory-bit(adr,reg2)) Store-memory-bit(adr,reg2,1)	3 注3	3 注3	3 注3				×	
SHL	reg1,reg2	rrrrr111111RRRRR 0000000011000000	GR[reg2]←GR[reg2] logically shift left by GR[reg1]	1	1	1	×	0	×	×	
	imm5,reg2	rrrrr010110iiiii	GR[reg2]←GR[reg2] logically shift left by zero-extend(imm5)	1	1	1	×	0	×	×	
SHR	reg1,reg2	rrrrr111111RRRRR 0000000010000000	GR[reg2]←GR[reg2] logically shift right by GR[reg1]	1	1	1	×	0	×	×	
	imm5,reg2	rrrrr010100iiiii	GR[reg2]←GR[reg2] logically shift right by zero-extend(imm5)	1	1	1	×	0	×	×	
SLD.B	disp7[ep],reg2	rrrrr0110ddddddd	adr←ep+zero-extend(disp7) GR[reg2]←sign-extend(Load-memory(adr,Byte))	1	1	注9					
SLD.BU	disp4[ep],reg2	rrrrr0000110dddd 注 18	adr←ep+zero-extend(disp4) GR[reg2]←zero-extend(Load-memory(adr,Byte))	1	1	注9					
SLD.H	disp8[ep],reg2	rrrrr1000ddddddd 注 19	adr←ep+zero-extend(disp8) GR[reg2]←sign-extend(Load-memory(adr,Halfword))	1	1	注9					
SLD.HU	disp5[ep],reg2	rrrrr0000111dddd 注 18, 20	adr←ep+zero-extend(disp5) GR[reg2]←zero-extend(Load-memory(adr,Halfword))	1	1	注9					
SLD.W	disp8[ep],reg2	rrrrr1010dddddd0 注 21	adr←ep+zero-extend(disp8) GR[reg2]←Load-memory(adr,Word)	1	1	注9					
SST.B	reg2,disp7[ep]	rrrrr0111ddddddd	adr←ep+zero-extend(disp7) Store-memory(adr,GR[reg2],Byte)	1	1	1					
SST.H	reg2,disp8[ep]	rrrrr1001ddddddd 注 19	adr←ep+zero-extend(disp8) Store-memory(adr,GR[reg2],Halfword)	1	1	1					
SST.W	reg2,disp8[ep]	rrrrr1010dddddd1 注 21	adr←ep+zero-extend(disp8) Store-memory(adr,GR[reg2],Word)	1	1	1					
ST.B	reg2,disp16[reg1]	rrrrr111010RRRRR dddddddddddddddd	adr←GR[reg1]+sign-extend(disp16) Store-memory(adr,GR[reg2],Byte)	1	1	1					
ST.H	reg2,disp16[reg1]	rrrrr111011RRRRR ddddddddddddddd0 注 8	adr←GR[reg1]+sign-extend(disp16) Store-memory (adr,GR[reg2], Halfword)	1	1	1					
ST.W	reg2,disp16[reg1]	rrrrr111011RRRRR ddddddddddddddd1 注 8	adr←GR[reg1]+sign-extend(disp16) Store-memory (adr,GR[reg2], Word)	1	1	1					
STSR	regID,reg2	rrrrr111111RRRRR 0000000001000000	GR[reg2]←SR[regID]	1	1	1					

(6/6)

助记符	操作数	操作码	操作	执行时钟			标志				
				i	r	l	CY	OV	S	Z	SAT
SUB	reg1,reg2	rrrrr001101RRRRR	GR[reg2]←GR[reg2]–GR[reg1]	1	1	1	×	×	×	×	
SUBR	reg1,reg2	rrrrr001100RRRRR	GR[reg2]←GR[reg1]–GR[reg2]	1	1	1	×	×	×	×	
SWITCH	reg1	00000000010RRRRR	adr←(PC+2) + (GR [reg1] logically shift left by 1) PC←(PC+2) + (sign-extend (Load-memory (adr,Halfword)) logically shift left by 1	5	5	5					
SXB	reg1	00000000101RRRRR	GR[reg1]←sign-extend (GR[reg1] (7 : 0))	1	1	1					
SXH	reg1	00000000111RRRRR	GR[reg1]←sign-extend (GR[reg1] (15 : 0))	1	1	1					
TRAP	vector	00000111111iiiii 0000000100000000	EIPC ←PC+4(return PC) EIPSW ←PSW ECR.EICC ←Exception code (40H to 4FH, 50H to 5FH) PSW.EP ←1 PSW.ID ←1 PC ←00000040H (when vector is 00H to 0FH (exception code: 40H to 4FH)) 00000050H (when vector is 10H to 1FH (exception code: 50H to 5FH))	4	4	4					
TST	reg1,reg2	rrrrr001011RRRRR	result←GR[reg2] AND GR[reg1]	1	1	1		0	×	×	
TST1	bit#3,disp16[reg1]	11bbb111110RRRRR dddddddddddddddd	adr←GR[reg1]+sign-extend(disp16) Z flag←Not (Load-memory-bit (adr,bit#3))	3 注3	3 注3	3 注3				×	
	reg2, [reg1]	rrrrr111111RRRRR 0000000011100110	adr←GR[reg1] Z flag←Not (Load-memory-bit (adr,reg2))	3 注3	3 注3	3 注3				×	
XOR	reg1,reg2	rrrrr001001RRRRR	GR[reg2]←GR[reg2] XOR GR[reg1]	1	1	1		0	×	×	
XORI	imm16,reg1,reg2	rrrrr110101RRRRR iiiiiiiiiiiiiiii	GR[reg2]←GR[reg1] XOR zero-extend (imm16)	1	1	1		0	×	×	
ZXB	reg1	00000000100RRRRR	GR[reg1]←zero-extend (GR[reg1] (7 : 0))	1	1	1					
ZXH	reg1	00000000110RRRRR	GR[reg1]←zero-extend (GR[reg1] (15 : 0))	1	1	1					

注
1. dddddddd： disp9 的高 8 位。
2. 如果之前紧邻一条指令改写 PSW 的内容，则为 3。
3. 如果没有等待状态 (3 + 读取访问等待状态数)。
4. n 为列表 12 中加载寄存器的总数。 (根据等待状态数目， n 为列表 12 中加载寄存器的总数。 如果 n = 0，和 n = 1 时的操作相同)
5. RRRRR: 除了 00000 之外。
6. 只有低半字有效。
7. dddddddddddddddddddddd：disp22 的高 21 位。
8. ddddddddddddddd：disp16 的高 15 位。
9. 依照等待状态的数目(如果没有等待状态，则为 1)。
10. b：disp16 的位 0。
11. 依照等待状态的数目(如果没有等待状态，则为 2)。

注 **12.** 本指令中，为了助记符描述方便，使源寄存器为 reg2，而操作码位置上使用 reg1 寄存器。所以，助记符描述中寄存器的含义和其它指令有所不同。

rrrrr = 指定 regID

RRRRR = 指定 reg2

13. iiiii: imm9 的低 5 位。

IIII: imm9 的高 4 位。

14. 在 reg2 = reg3 (结果的低 32 位不写入该寄存器)或 reg3 = r0 (结果的高 32 位不写入该寄存器)的情况下，缩短 1 个时钟。

15. sp/imm: 由次操作码的位 19 和位 20 指定。

16. ff = 00: 将 sp 载入 ep。

01: 将符号扩展的 16 位立即数(位 47 至 32)载入 ep。

10: 将经过 16 位逻辑左移的 16 位立即数 (位 47 至 32) 载入 ep。

11: 将 32 位立即数 (位 63 至 32) 载入 ep。

17. If imm = imm32， n + 3 时钟。

18. rrrrr: 非 00000。

19. ddddddd: disp8 的高 8 位。

20. dddd: disp5 的高 4 位。

21. dddddd: disp8 的高 6 位。

22. 当使用“MUL reg1, reg2, reg3” 指令和“MULU reg1, reg2, reg3”指令时，不要同时满足下列所有条件。当执行的指令满足下列条件时，操作无法保证。

- Reg1 = reg3
- Reg1 ≠ reg2
- Reg1 ≠ r0
- Reg3 ≠ r0

<R>

附录 D 修订历史

D.1 本版的主要修订

(1/2)

页码	描述内容
全文	• 增加产品 μPD70F3454F1-DA9-A • 增加封装 161 针塑封 FBGA (10 × 10) • 增加 flash 存储器编程器 PG-FP5 和 FL-PR5
p. 33	增加 **1.3.4 引脚配置 (V850E/IG3)**的注
p. 65	**3.2.2 (2) NMI 状态保存寄存器 (FEPC, FEPSW)**中描述内容的修改
p. 166	**表 4-15 噪声消除器 (1/2)**中注意事项 **1** 和 **2** 的修改
p. 167	**表 4-15 噪声消除器 (2/2)**中注意事项 **1** 和 **2** 的修改
p. 169	增加 **4.6 (1) 数字噪声消除 0 控制寄存器 n (INTNFCn)**的注
p. 214	**6.6 (1) (a) 计数器开始操作**中描述内容的修改
p. 236	增加**图 6-19 外部事件计数模式的寄存器设置 (2/2)**中的**注意事项 2**
p. 244	增加 **图 6-24 外部触发脉冲输出模式的配置**中的**注意事项**，并修改描述内容
p. 246	**图 6-26 外部触发脉冲输出模式的寄存器设置 (1/2)**中描述内容的修改
p. 247	**图 6-26 外部触发脉冲输出模式的寄存器设置 (2/2)**中描述内容的修改
p. 256	增加**图 6-28 单次触发脉冲输出模式的配置**中的**注意事项**，并修改描述内容
p. 258	**图 6-30 单次触发脉冲输出模式的寄存器设置 (1/2)**中描述内容的修改
p. 259	增加**图 6-30 单次触发脉冲输出模式的寄存器设置 (2/2)**中的**注意事项**，并修改描述内容
p. 260	**图 6-31 单次触发脉冲输出模式的软件处理流程**中描述内容的修改
p. 288	增加 **图 6-42 脉冲宽度测量模式的配置**中的**注意事项**，并修改描述内容
p. 290	**图 6-44 脉冲宽度测量模式的寄存器设置 (1/2)**中描述内容的修改
p. 291	**图 6-44 脉冲宽度测量模式的寄存器设置 (2/2)**中描述内容的修改
p. 292	**图 6-45 <1> 计数操作启动流程**中描述内容的修改
p. 314	**7.6 (1) (a) 计数器开始操作**中描述内容的修改
p. 361	增加**图 7-26 单次触发脉冲输出模式的寄存器设置 (3/3)**中的**注意事项**
p. 430	**8.6 (1) (a) 计数器开始操作**中描述内容的修改
p. 449	增加**图 8-16 (f) TMTm 捕获/比较寄存器 1 (TTmCCR1)**中的**注意事项**
p. 451	增加 **8.6.2 (2) 外部事件计数模式的操作时序**中的**注意事项**
p. 473	增加**图 8-27 单次触发脉冲输出模式的寄存器设置 (2/2)**中的**注意事项**
p. 474	修改**图 8-28 单次触发脉冲输出模式的软件处理流程**中的图
p. 510	增加 **8.6.8 (1) 0%/100%的 PWM 输出**
p. 549	增加 **10.3 (2) TABn 选项寄存器 1 (TABnOPT1)**中的注，并修改描述内容
p. 652	增加 **12.3 (20) 比较器输出数字噪声消除寄存器 nL, nF (CMPNFCnL, CMPNFCnF)**中的注
p. 680	**12.6.8 A/D 转换结果迟滞特性**中描述内容的修改
p. 693	**13.3 (2) A/D 转换 2 模式寄存器 1 (AD2M1)**中**注意事项**的修改

(2/2)

页码	描述内容
p. 716	**13.7 (11) A/D 转换结果迟滞特性**中描述内容的修改
p. 721	图 **14-4 UARTAn 的框图**中描述内容的修改
p. 733	图 **14-8 连续发送操作时序**中描述内容的修改
p. 922	图 **17-21 (a) 开始条件 ~ 地址**的修改图
p. 923	图 **17-21 (b) 数据**的修改图
p. 924	图 **17-21 (c) 停止条件**的修改图
p. 925	图 **17-22 (a) 开始条件 ~ 地址**的修改图
p. 926	图 **17-22 (b) 数据**的修改图
p. 927	图 **17-22 (c) 停止条件**的修改图
p. 969	**19.3.6 DMA 触发因素寄存器 0 至 3 (DTFR0 至 DTFR3)**中注意事项 **1** 的修改
p. 971	表 **19-1 DMA 启动因素 (1/2)**中注的修改
p. 981	**19.12 (3) CPU 的总线仲裁**中描述内容的修改
p. 981	增加 **19.12 (4) DMA 启动因素**
p. 1000	**20.3.4 中断控制寄存器 (xxICn)**中注意事项 **2** 的修改
p. 1107	增加 **28.1.4 (i) Murata Mfg. Co., Ltd.：陶瓷谐振器 (T_A = –40 至+85°C)**
p. 1131	增加 **28.2.4 (i) Murata Mfg. Co., Ltd.：陶瓷谐振器 (T_A = –40 至+85°C) (除μPD70F3454F1-DA9-A 之外的产品)**
p. 1141	**28.2.8 (4) (b) 读取周期 (CLKOUT 同步)**增加注意事项
p. 1145	**28.2.8 (4) (d) 写入周期 (CLKOUT 同步)**增加注意事项
p. 1165	增加 **第三十章 推荐焊接条件** 的描述内容
p. 1191	增加**附录 D 修订历史**

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

[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

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

[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

[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

[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

[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

[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