CMS32M65xx用户手册

基于ARM® Cortex®-M0+的超低功耗32位微控制器

V0.9.0

请注意以下有关CMS知识产权政策

＊中微半导体（深圳）股份有限公司（以下简称本公司）已申请了专利，享有绝对的合法权益。与本公司MCU或其他产品有关的专利权并未被同意授权使用，任何经由不当手段侵害本公司专利权的公司、组织或个人，本公司将采取一切可能的法律行动，遏止侵权者不当的侵权行为，并追讨本公司因侵权行为所受的损失、或侵权者所得的不法利益。

＊中微半导体（深圳）股份有限公司的名称和标识都是本公司的注册商标。
＊本公司保留对规格书中产品在可靠性、功能和设计方面的改进作进一步说明的权利。然而本公司对于规格内容的使用不负责任。文中提到的应用其目的仅仅是用来做说明，本公司不保证和不表示这些应用没有更深入的修改就能适用，也不推荐它的产品使用在会由于故障或其它原因可能会对人身造成危害的地方。本公司的产品不授权适用于救生、维生器件或系统中作为关键器件。本公司拥有不事先通知而修改产品的权利，对于最新的信息，请参考官方网站 www.mcu.com.cn

文档使用说明

本手册是CMS32M65xx微控制器产品的技术参考手册，技术参考手册是有关如何使用本系列产品的应用说明资料，包含各个功能模块的结构、功能描述、工作模式以及寄存器配置等详细信息,并对每种功能模块都有专门的章节进行介绍。

技术参考手册是针对这一系列产品所有功能模块的说明，若要了解特定型号产品的特征说明（即功能搭载情况），可参考相应的数据手册。

数据手册信息如下：

CMS32M65xx：CMS32M65xx_数据手册_vx.x.x. pdf

通常在芯片选型的初期，首先要看数据手册，以评估该产品是否能够满足设计上的功能需求；在基本选定所需产品后，需要查看技术参考手册，以确定各功能模块的工作模式是否符合要求；在确定选型进入编程设计阶段时，需要详细阅读技术参考手册，以获知各项功能的具体实现方式和寄存器配置。在设计硬件时可参考数据手册以获得电压，电流，驱动能力以及管脚分配等信息。

关于Cortex-M0+核心、SysTick定时器和NVIC的详细说明，请参照对应ARM的文档。

目录

文档使用说明 2
第1章 CPU 15
1.1 概述 15
1.2 Cortex-M0+内核特性 15
第2章 调试特性 16
2.1 SWD接口引脚 17
2.2 ARM参考文档 18
第3章 引脚功能 19
3.1 端口功能 19
3.2 端口复用功能 19
3.3 寄存器映射 21
3.3.1 控制功能寄存器映射 21
3.3.2 输出输入复用功能寄存器映射 22
3.3.3 特殊功能端口RESINB控制寄存器映射 24
3.4 寄存器说明 25
3.4.1 端口寄存器（Px） 26
3.4.2 端口模式寄存器（PMx） 27
3.4.3 上拉电阻选择寄存器（PUx） 28
3.4.4 下拉电阻选择寄存器（PDx） 29
3.4.5 端口输出模式寄存器（POMx） 30
3.4.6 端口模式控制寄存器（PMCx） 31
3.4.7 端口置位控制寄存器（PSETx） 32
3.4.8 端口清零控制寄存器（PCLRx） 33
3.4.9 端口输出复用功能配置寄存器（PmnCFG） 34
3.4.10 端口输入复用功能配置寄存器（PSxx_CFG） 35
3.4.11 TTL和施密特输入选择(PxTTLCFG,x=0和2) 39
3.4.12 特殊功能端口RESINB说明(RSTM) 39
第4章 系统结构 40
4.1 概述 40
4.2 系统地址划分 41
第5章 时钟发生电路 44
5.1 时钟发生电路的功能 44
5.2 时钟发生电路的结构 45
5.3 寄存器映射 47
5.4 寄存器说明 48
5.4.1 系统时钟控制寄存器（CKC） 48
5.4.2 时钟运行状态控制寄存器（CSC） 49
5.4.3 外围允许寄存器0、1（PER0、PER11，PER12，PER13） 49
5.4.4 12位间隔定时器运行时钟选择寄存器（OSMC） 53

5.4.5 高速内部振荡器的频率选择寄存器（HOCODIV）.....53
5.4.6 低速内部振荡器时钟选择寄存器（SUBCKSEL）.....54
5.4.7 供电模式控制保护寄存器（PMUKEY）.....54
5.4.8 供电模式控制寄存器（PMUCTL）.....54
5.5 系统时钟振荡电路.....55
5.5.1 高速内部振荡器.....55
5.5.2 低速内部振荡器.....55
5.6 时钟发生电路的运行.....56
5.7 时钟控制.....57
5.7.1 高速内部振荡器的设置例子.....57
5.7.2 CPU时钟的状态转移图.....59
5.7.3 CPU时钟转移前的条件和转移后的处理.....62
5.7.4 CPU时钟和主系统时钟的切换所需时间.....63
5.7.5 时钟振荡停止前的条件.....63
第6章 通用定时器单元的功能.....64
6.1 通用定时器的功能.....64
6.1.1 独立通道运行功能.....64
6.1.2 多通道联动运行功能.....66
6.1.3 8 位定时器运行功能（只限于通道1和通道3）.....67
6.2 通用定时器单元的结构.....68
6.2.1 寄存器映射.....70
6.2.2 定时器计数寄存器mn（TCRmn）.....71
6.2.3 定时器数据寄存器mn（TDRmn）.....72
6.2.4 外围允许寄存器0（PER0）.....73
6.2.5 定时器时钟选择寄存器m（TPSm）.....74
6.2.6 定时器模式寄存器mn（TMRmn）.....76
6.2.7 定时器状态寄存器mn（TSRmn）.....86
6.2.8 定时器通道允许状态寄存器m（TEm）.....87
6.2.9 定时器通道开始寄存器m（TSm）.....88
6.2.10 定时器通道停止寄存器m（TTm）.....89
6.2.11 定时器输入输出选择寄存器（TIOS0）.....90
6.2.12 定时器输出允许寄存器m（TOEm）.....91
6.2.13 定时器输出寄存器m（TOm）.....92
6.2.14 定时器输出电平寄存器m（TOLm）.....93
6.2.15 定时器输出模式寄存器m（TOMm）.....94
6.2.16 噪声滤波器允许寄存器1（NFEN1）.....95
6.2.17 控制定时器输入/输出引脚端口功能的寄存器.....96
6.3 通用定时器单元的基本规则.....97
6.3.1 多通道联动运行功能的基本规则.....97
6.3.2 8 位定时器运行功能的基本规则（只限于通道1和通道3）.....99
6.4 计数器的运行.....100
6.4.1 计数时钟（F_{TCLK}）.....100

6.4.2 计数器的开始时序 102
6.4.3 计数器的运行 103
6.5 通道输出（TOmn引脚）的控制 108
6.5.1 TOmn引脚输出电路的结构 108
6.5.2 TOmn引脚的输出设定 109
6.5.3 通道输出运行的注意事项 110
6.5.4 TOmn位的一次性操作 114
6.5.5 有关开始计数时的定时器中断和TOmn引脚输出 115
6.6 定时器输入（TImn）的控制 116
6.6.1 TImn引脚输入电路的结构 116
6.6.2 噪声滤波器 116
6.6.3 操作通道输入时的注意事项 117
6.7 通用定时器单元的独立通道运行功能 118
6.7.1 作为间隔定时器/ 方波输出的运行 118
6.7.2 作为外部事件计数器的运行 122
6.7.3 作为分频器的运行 125
6.7.4 作为输入脉冲间隔测量的运行 128
6.7.5 作为输入信号高低电平宽度测量的运行 131
6.7.6 作为延迟计数器的运行 135
6.8 通用定时器单元的多通道联动运行功能 138
6.8.1 作为单触发脉冲输出功能的运行 138
6.8.2 作为 PWM 功能的运行 145
6.8.3 作为多重 PWM 输出功能的运行 152
第7章 LSITIMER 12位间隔定时器 160
7.1 12位间隔定时器的功能 160
7.2 12位间隔定时器的结构 160
7.3 寄存器映射 160
7.4 12位间隔定时器的控制寄存器（CON0） 161
7.5 12位间隔定时器的运行 162
7.5.1 12位间隔定时器的运行时序 162
7.5.2 从睡眠模式返回后开始计数器的运行并且再次向睡眠模式的转移 163
第8章 时钟输出/蜂鸣器输出控制电路 164
8.1 时钟输出/蜂鸣器输出控制电路的功能 164
8.2 时钟输出/蜂鸣器输出控制电路的结构 165
8.3 寄存器映射 166
8.3.1 时钟输出选择寄存器（CKS0） 166
8.3.2 时钟输出选择寄存器（CKS1） 167
8.4 配置时钟输出/蜂鸣器输出引脚端口功能的寄存器 168
8.5 时钟输出/蜂鸣器输出控制电路的运行 169
8.5.1 输出引脚的运行 169
8.6 时钟输出/蜂鸣器输出控制电路的注意事项 169
第9章 看门狗定时器 170

9.1 看门狗定时器的功能 170
9.2 看门狗定时器的结构 170
9.3 寄存器映射 172
9.3.1 看门狗定时器的允许寄存器（WDTE） 172
9.3.2 LOCKUP控制寄存器（LOCKCTL） 173
9.3.3 保护寄存器（PRCR） 173
9.4 看门狗定时器的运行 174
9.4.1 看门狗定时器的运行控制 174
9.4.2 看门狗定时器上溢时间的设定 175
9.4.3 看门狗定时器窗口打开期间的设定 176
9.4.4 看门狗定时器间隔中断的设定 177
9.4.5 LOCKUP期间看门狗定时器的运行 177
第10章 除法与开方运算单元（DIVSQRT） 178
10.1 概述 178
10.2 特性 178
10.3 功能说明 178
10.4 寄存器映射 179
10.5 寄存器说明 180
10.5.1 除法开方器控制寄存器(DIVSQRTCON) 180
10.5.2 除法开方器数据A寄存器(DIVSQRTALUA) 180
10.5.3 除法开方器数据B寄存器(DIVSQRTALUB) 180
10.5.4 除法开方器结果0寄存器(DIVSQRTRES0) 180
10.5.5 除法开方器结果1寄存器(DIVSQRTRES1) 180
第11章 定时器（TIMER0/1） 181
11.1 概述 181
11.2 特性 181
11.3 功能描述 181
11.3.1 单次触发模式 181
11.3.2 周期计数模式 181
11.3.3 连续计数模式 181
11.3.4 延迟加载功能 181
11.4 寄存器映射 182
11.5 寄存器说明 183
11.5.1 定时器控制寄存器（CON0/1） 183
11.5.2 定时器加载寄存器（LOAD0/1） 183
11.5.3 定时器当前值寄存器（VAL0/1） 183
11.5.4 定时器中断源状态寄存器（RIS0/1） 184
11.5.5 定时器已使能中断状态寄存器（MIS0/1） 184
11.5.6 定时器中断清零寄存器（ICLR0/1） 184
11.5.7 定时器延迟加载寄存器（BGLOAD0/1） 184
第12章 捕捉/比较/脉宽调制模块（CCP0/1） 185

12.1 概述 185
12.2 特性 185
12.3 功能描述 186
12.3.1 脉宽调制模式（PWM） 186
12.3.2 方波输出模式 186
12.3.3 捕捉模式0 187
12.3.4 捕捉模式1 187
12.3.5 捕捉模式2 189
12.3.6 PWM配置过程 190
12.3.7 中断 190
12.4 寄存器映射 191
12.5 寄存器说明 192
12.5.1 CCPx控制寄存器（CCPCONx）（x=0,1） 192
12.5.2 CCP重加载寄存器（CCPLOADx）（x=0,1） 193
12.5.3 CCPxA数据寄存器（CCPDxA）（x=0,1） 193
12.5.4 CCPxB数据寄存器（CCPDxB）（x=0,1） 193
12.5.5 CCP中断使能寄存器（CCPIMSC） 194
12.5.6 CCP中断源状态寄存器（CCPRIS） 195
12.5.7 CCP已使能中断状态寄存器（CCPMIS） 196
12.5.8 CCP中断清零寄存器（CCPICLR） 197
12.5.9 CCP运行寄存器（CCPRUN） 197
12.5.10 CCP写使能控制寄存器(LOCK) 197
12.5.11 CAP控制寄存器（CAPCON） 198
12.5.12 CAP通道选择寄存器（CAPCHS） 200
12.5.13 CAP数据寄存器（CAP0DATA） 201
12.5.14 CAP数据寄存器（CAPnDATA）（n=1-3） 201
第13章 增强型PWM（EPWM） 202
13.1 概述 202
13.2 特性 202
13.3 功能描述 203
13.3.1 结构框图 204
13.3.2 时钟分频 204
13.3.3 独立输出模式 204
13.3.4 互补输出模式 205
13.3.5 同步输出模式 205
13.3.6 成组输出模式 205
13.3.7 加载更新模式 206
13.3.8 边沿对齐计数模式 208
13.3.9 中心对齐计数模式 209
13.3.10 独立计数器比较功能 212
13.3.11 可编程死区发生器 214
13.3.12 掩码及掩码预设功能 215

13.3.13 霍尔传感器接口功能 215
13.3.14 故障保护功能（刹车及恢复功能） 218
13.3.15 调试模式下的输出状态 220
13.3.16 输出通道重映射功能 220
13.3.17 EPWM配置过程 220
13.3.18 中断 221
13.4 寄存器映射 222
13.5 寄存器说明 223
13.5.1 EPWM预分频寄存器(CLKPSC) 223
13.5.2 EPWM时钟选择寄存器(CLKDIV) 224
13.5.3 EPWM控制寄存器(CON) 226
13.5.4 EPWM控制寄存器(CON2) 228
13.5.5 EPWM控制寄存器(CON3) 229
13.5.6 EPWM周期寄存器0-5(PERIOD0-5) 232
13.5.7 EPWM比较寄存器0-5(CMPDAT0-5) 232
13.5.8 EPWM输出控制寄存器(POEN) 232
13.5.9 EPWM输出通道重映射寄存器(POREMAP) 233
13.5.10 EPWM故障保护控制寄存器(BRKCTL) 235
13.5.11 EPWM死区长度寄存器(DTCTL) 237
13.5.12 EPWM掩码输出控制寄存器(MASK) 238
13.5.13 EPWM掩码输出控制预设寄存器(MASKNXT) 239
13.5.14 EPWM触发比较寄存器(CMPTGD0-1) 241
13.5.15 EPWM中断使能寄存器(IMSC) 242
13.5.16 EPWM中断源状态寄存器(RIS) 243
13.5.17 EPWM已使能中断状态寄存器(MIS) 244
13.5.18 EPWM中断清零控制寄存器(ICLR) 245
13.5.19 EPWM中断累加控制寄存器(IFA) 246
13.5.20 EPWM写使能控制寄存器(LOCK) 246
13.5.21 EPWM故障保护恢复延时寄存器(BRKRDT) 247

第14章 通用异步收发器（UART） 248

14.1 概述 248
14.2 特性 248
14.3 功能描述 248
14.3.1 UART功能模式 248
14.3.2 UART中断和状态 248
14.4 寄存器映射 249
14.5 寄存器说明 250
14.5.1 接收缓存寄存器（RBR） 250
14.5.2 发送缓存寄存器（THR） 250
14.5.3 波特率分频寄存器（DLR） 250
14.5.4 中断使能寄存器（IER） 250
14.5.5 中断状态寄存器（IIR） 251

14.5.6 线控制寄存器（LCR） 251
14.5.7 Modem控制寄存器（MCR） 252
14.5.8 线状态寄存器（LSR） 252
14.5.9 UART访问结束寄存器（END） 253

第15章 I²C串行接口控制器（I²C） 254

15.1 概述 254
15.2 特性 254
15.3 寄存器映射 254
15.4 寄存器说明 255
15.4.1 I²C控制置位寄存器（CONSET） 255
15.4.2 I²C控制清零寄存器（CONCLR） 256
15.4.3 I²C状态寄存器（STAT） 257
15.4.4 I²C数据寄存器（DAT） 257
15.4.5 I²C时钟控制寄存器（CLK） 258
15.4.6 I²C从机地址寄存器（ADR0/ADR1/ADR2/ADR3） 258
15.4.7 I²C从机地址掩码寄存器（ADM0/ADM1/ADM2/ADM3） 258
15.4.8 I²C扩展从机地址寄存器（XADR0） 258
15.4.9 I²C扩展从机地址掩码寄存器（XADM0） 258
15.4.10 I²C软件复位寄存器（RST） 259

第16章 串行外围接口控制器（SSP/SPI） 260

16.1 概述 260
16.2 特性 260
16.3 寄存器映射 260
16.4 寄存器说明 261
16.4.1 SSP控制寄存器（CON） 261
16.4.2 SSP状态寄存器（STAT） 262
16.4.3 SSP数据寄存器（DAT） 262
16.4.4 SSP时钟控制器（CLK） 262
16.4.5 SSP中断使能寄存器（IMSC） 263
16.4.6 SSP中断源状态寄存器（RIS） 263
16.4.7 SSP已使能中断状态寄存器（MIS） 263
16.4.8 SSP中断清零寄存器（ICLR） 264
16.4.9 SSP软件片选信号寄存器（CSCR） 264

第17章 快速模数转换（ADC） 265

17.1 概述 265
17.2 特性 265
17.3 功能描述 266
17.3.1 ADC的通道 266
17.3.2 ADC结构框图 267
17.3.3 ADC的功耗模式 268
17.3.4 ADC的转换模式 268

17.3.5 ADC的时钟 268
17.3.6 ADC的通道选择及中断产生 269
17.3.7 ADC软件启动 269
17.3.8 ADC硬件触发启动 270
17.4 寄存器映射 273
17.5 寄存器说明 274
17.5.1 ADC控制寄存器(CON) 274
17.5.2 ADC控制寄存器2(CON2) 276
17.5.3 ADC硬件触发控制寄存器(HWTG) 277
17.5.4 ADC EPWM触发延时寄存器(EPWMTGDLY) 278
17.5.5 ADC扫描寄存器(SCAN) 278
17.5.6 ADC EPWM输出触发转换通道使能寄存器(CHEPWM) 278
17.5.7 ADC EPWM比较器0触发转换通道使能寄存器(CHPTG0) 279
17.5.8 ADC EPWM比较器1触发转换通道使能寄存器(CHPTG1) 279
17.5.9 ADC转换结果寄存器(DATAx) x=23-0,x≠19 279
17.5.10 ADC比较控制寄存器0(CMPx) x=0～1 280
17.5.11 ADC中断使能寄存器(IMSC) 281
17.5.12 ADC中断源状态寄存器(RIS) 281
17.5.13 ADC已使能中断状态寄存器(MIS) 281
17.5.14 ADC中断清零寄存器(ICLR) 282
17.5.15 ADC写使能控制寄存器(LOCK) 282

第18章 可编程增益放大器（PGA0/1/2） 283

18.1 概述 283
18.2 特性 283
18.3 结构框图 284
18.4 寄存器映射 285
18.5 寄存器说明 286
18.5.1 PGA0控制寄存器0 286
18.5.2 PGA0控制寄存器1 286
18.5.3 PGA0访问寄存器使能 287
18.5.4 PGA1控制寄存器0 287
18.5.5 PGA2控制寄存器0 288
18.5.6 PGA12控制寄存器0 288
18.5.7 PGA1/PGA2访问寄存器使能 288

第19章 模拟比较器（ACMP0/1） 289

19.1 概述 289
19.2 结构框图 289
19.3 特性 291
19.4 功能说明 291
19.5 寄存器映射 292
19.6 寄存器说明 293
19.6.1 模拟比较器0控制寄存器0(C0CON0) 293

19.6.2 模拟比较器0控制寄存器1(C0CON1) 294
19.6.3 模拟比较器1控制寄存器0(C1CON0) 295
19.6.4 模拟比较器1控制寄存器1(C1CON1) 296
19.6.5 模拟比较器事件控制寄存器(CEVCON) 297
19.6.6 模拟比较器中断使能寄存器(IMSC) 297
19.6.7 模拟比较器中断源状态寄存器(RIS) 297
19.6.8 模拟比较器已使能中断源状态寄存器(MIS) 298
19.6.9 模拟比较器中断清零控制寄存器(ICLR) 298
19.6.10 模拟比较器写使能控制寄存器(LOCK) 298

第20章 DAC 299

20.1 概述 299
20.2 结构框图 299
20.3 特性 299
20.4 寄存器映射 299
20.5 寄存器说明 300
20.5.1 DAC控制寄存器0(CON0) 300
20.5.2 DAC写使能控制寄存器(LOCK) 300

第21章 ADCLDO概述 301

21.1 特性 301
21.2 结构框图 301
21.3 寄存器映射 301
21.4 寄存器说明 302
21.4.1 ADCLDO控制寄存器0(CON0) 302
21.4.2 ADCLDO写使能控制寄存器(LOCK) 302

第22章 嵌套向量中断控制器（NVIC） 303

22.1 特性 303
22.2 异常模式和系统中断映射 303
22.3 向量表 304
22.4 寄存器映射 305
22.5 寄存器说明 306
22.5.1 中断设置使能控制寄存器（ISER） 306
22.5.2 中断清使能控制寄存器（ICER） 306
22.5.3 中断设置挂起控制寄存器（ISPR） 306
22.5.4 中断清挂起控制寄存器（ICPR） 307
22.5.5 IRQ0~IRQ3中断优先级寄存器（IPR0） 307
22.5.6 IRQ4~IRQ7中断优先级寄存器（IPR1） 307
22.5.7 IRQ8~IRQ11中断优先级寄存器（IPR2） 308
22.5.8 IRQ12~IRQ15中断优先级寄存器（IPR3） 308
22.5.9 IRQ16~IRQ19中断优先级寄存器（IPR4） 309
22.5.10 IRQ20~IRQ23中断优先级寄存器（IPR5） 309
22.5.11 IRQ24~IRQ27中断优先级寄存器（IPR6） 309

22.5.12 IRQ28~IRQ31中断优先级寄存器（IPR7） 310
22.5.13 外部中断上升沿允许寄存器（EGP0） 310
22.5.14 外部中断下降沿允许寄存器（EGN0） 311
第23章 待机功能 312
23.1 待机功能 312
23.2 睡眠模式 312
23.2.1 睡眠模式的设定 312
23.2.2 睡眠模式的解除 314
23.3 深度睡眠模式 315
23.3.1 深度睡眠模式的设定 315
23.3.2 深度睡眠模式的解除 317
23.4 部分掉电的深度睡眠模式 318
23.4.1 部分掉电的深度睡眠模式的设定 318
23.4.2 部分掉电的深度睡眠模式的解除 320
第24章 复位功能 321
24.1 确认复位源的寄存器 327
24.1.1 寄存器映射 327
24.1.2 复位控制标志寄存器（RESF） 327
第25章 上电复位电路 330
25.1 上电复位电路的功能 330
25.2 上电复位电路的结构 331
25.3 上电复位电路的运行 332
第26章 电压检测电路 335
26.1 电压检测电路的功能 335
26.2 电压检测电路的结构 336
26.3 寄存器映射 336
26.4 控制电压检测电路的寄存器 337
26.4.1 电压检测寄存器（LVIM） 337
26.4.2 电压检测电平寄存器（LVIS） 338
26.5 电压检测电路的运行 339
26.5.1 用作复位模式时的设定 339
26.6 用作中断模式时的设定 340
26.7 用作中断&复位模式时的设定 341
26.8 电压检测电路的注意事项 347
第27章 安全功能 349
27.1 安全功能的概要 349
27.2 寄存器映射 349
27.3 安全功能的运行 351
27.3.1 闪存CRC运算功能（高速CRC） 351
27.3.1.1 闪存 CRC 运算结果寄存器 L（PGCRCL） 352

27.3.1.2 CRC 运算功能（通用 CRC） 354
27.3.1.3 CRC 输入寄存器（CRCIN） 354
27.3.1.4 CRC 数据寄存器（CRCD） 355
27.3.2 SFR保护功能 356
27.3.2.1 SFR 保护控制寄存器（SFRGD） 356
27.3.3 频率检测功能 357
27.3.3.1 定时器输入输出选择寄存器 0（TIOS0） 357
27.3.4 A/D测试功能 358
27.3.4.1 ADC 寄存器 358
27.3.5 输入/输出引脚的数字输出信号电平检测功能 359
27.3.5.1 端口模式选择寄存器（PMS） 359
27.3.6 产品唯一身份标识寄存器 360

第28章 温度传感器 361

28.1 温度传感器的功能 361
28.2 寄存器映射 361
28.3 温度传感器的寄存器 361
28.3.1 温度传感器校准数据寄存器 TSN25 361
28.4 温度传感器的使用说明 362

第29章 选项字节 363

29.1 选项字节的功能 363
29.1.1 用户选项字节（000C0H～000C2H） 363
29.2 闪存数据保护选项字节（000C3H，500004H） 364
29.3 寄存器映射 364
29.4 用户选项字节 365
29.4.1 用户选项字节（000C0H） 365
29.4.2 用户选项字节（000C1H） 366
29.4.3 用户选项字节（000C2H） 369
29.4.4 闪存数据保护选项字节（000C3H） 369
29.4.5 闪存数据保护选项字节（500004H） 369

第30章 FLASH控制 370

30.1 FLASH控制功能描述 370
30.2 FLASH存储器结构 370
30.3 寄存器映射 371
30.4 寄存器说明 372
30.4.1 Flash写保护寄存器(FLPROT) 372
30.4.2 FLASH操作控制寄存器(FLOPMD1) 372
30.4.3 FLASH操作控制寄存器(FLOPMD2) 372
30.4.4 Flash擦除控制寄存器(FLERMD) 373
30.4.5 Flash状态寄存器(FLSTS) 373
30.4.6 Flash全片擦除时间控制寄存器(FLCERCNT) 374
30.4.7 Flash页擦除时间控制寄存器（FLSERCNT） 374

30.4.8 Flash 写入时间控制寄存器（FLPROCNT） 375
30.5 FLASH操作方法 376
30.5.1 页擦除（sector erase） 376
30.5.2 全片擦除（chip erase） 377
30.5.3 编程（word program） 377
30.6 闪存读取 378
30.7 FLASH操作的注意事项 378
附录 修订记录 379

第1章 CPU

1.1 概述

本章节简单介绍本制品搭载的ARM Cortex-M0+内核的特性及调试特性，详情请参考ARM相关文档。

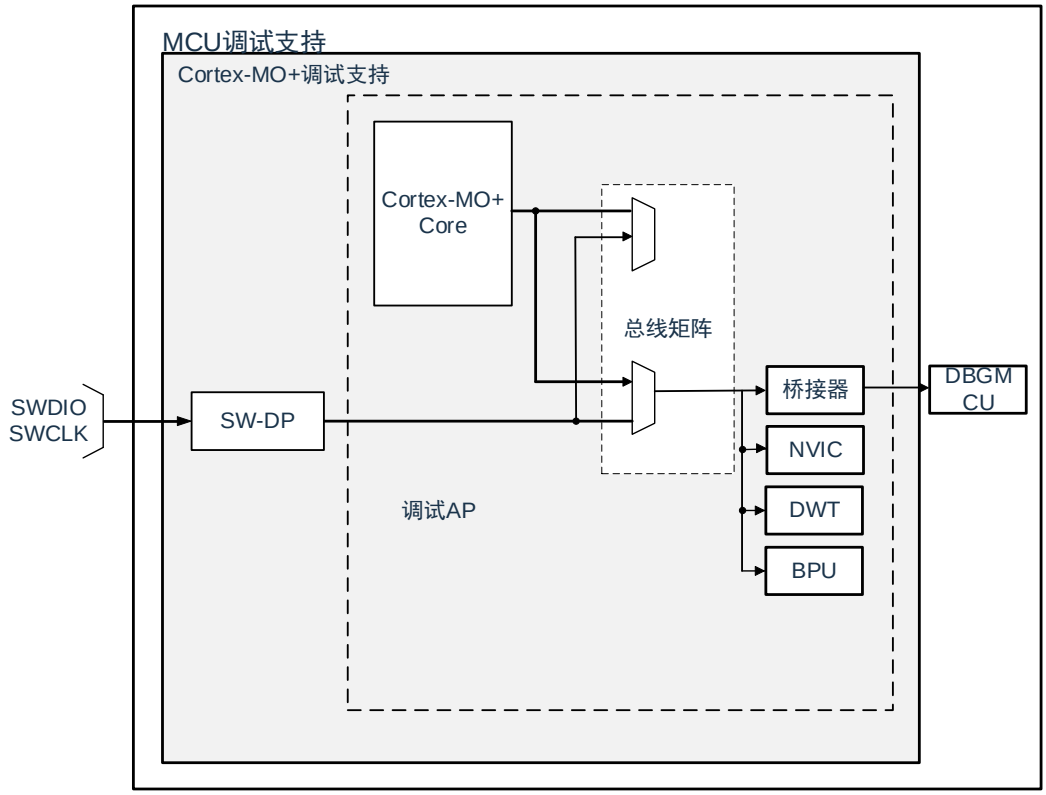
1.2 Cortex-M0+内核特性

- ARM Cortex-M0+处理器是32位RISC结构，采用2级流水线，仅支持特权模式
- 1周期硬件乘法器
- 嵌套向量中断控制器（NVIC）
  - 1个不可屏蔽中断(NMI)
  - 支持21个可屏蔽中断请求(IRQ)
  - 4个中断优先级
- 系统定时器SysTick是一个24位倒计时定时器，可选择F_{CLK}或F_{IL}计数时钟
- 向量表偏移寄存器(VTOR)
  - 软件可以写VTOR将向量表起始地址重新定位到不同的位置
  - 该寄存器的默认值为0x0000_0000,低8位写忽略，读为零，也就是偏移量256字节对齐

第2章 调试特性

- 2线SWD调试接口
- 支持暂停、恢复和单步执行程序
- 访问处理器的内核寄存器和特殊功能寄存器
- 4个硬件断点(BPU)
- 无限个软件断点(BKPT指令)
- 2个数据观察点(DWT)
- 内核执行的时候访问存储器

图2-1：Cortex-M0+的调试框图



注：SWD在Deep Sleep模式下不能工作，请在active和sleep模式下进行调试操作。

2.1 SWD接口引脚

本产品的2个GPIO可用作SWD接口引脚，这些引脚在所有的封装里都存在。

表2-1：SWD调试端口引脚

SWD 端口名称	调试功能	引脚分配
SWDCLK	串行时钟	P06
SWDIO	串行数据输入 /输出	P07

不使用SWD功能时，可以通过设置debug停止控制寄存器（DBGSTOPCR）来禁用SWD。

位	符号	描述	复位值
31:25	-	保留	-
24	SWDIS	SWD调试接口禁用 0: SWD调试接口使能。在连接调试器的状态下，P06/P07不能用作GPIO（因为此时该IOBUF的ENO和DOUT由调试器控制） 1: SWD调试接口禁用。P06/P07可用作GPIO	0
23:2	-	保留	0x0
1	FRZEN1	在调试器连接的状态下，并且 CPU 处于调试状态时（HALTED=1)，通信系周边模块动作/停止 注1	0
		0： 周边动作	0
		1： 周边停止	0
0	FRZEN0	在调试器连接的状态下，并且CPU处于调试状态时（HALTED=1)，定时器系周边模块动作/停止 注2	0
		0： 周边动作	0
		1： 周边停止	0

注1：本制品的通信系周边模块包括：通信串行通信单元，串行IICA

注2：本制品的定时器系周边模块包括：通用定时器单元Timer4

2.2 ARM参考文档

Cortex®-M0+ 内核中内置的调试功能是ARM® CoreSight 设计套件的一部分。相关文档请参考：

- Cortex®-M0+技术参考手册(TRM)
- ARM®调试接口V5
- ARM® CoreSight设计套件版本r1p1 技术参考手册
- ARM® CoreSight™ MTB-M0+ Technical Reference Manual

第3章 引脚功能

3.1 端口功能

具体端口功能参见各产品系列的数据手册。

3.2 端口复用功能

具体端口复用功能参见各产品系列的数据手册。详细端口复用功能参见下表。

表3-1：端口复用功能数字映射表

功能名称	输入	复用功能 PmnCFG					
		0	1	2	3	4	5
P00	BKIN	默认兼用输出	-	-	-	-	-
P01	-		-	-	-	-	-
P02	CCPnA_I / CCPnB_I		C1_O	SSIO	CCP0A_O	CCP1A_O	-
P03	RXD		--	SCK			SDA
P04	CCPnA_I / CCPnB_I		TXD	MISO	CCP0B_O	CCP1B_O	SCL
P05			C0_O	MOSI	-	-	-
P06	CCPnA_I / CCPnB_I / RXD		-	-	CCP0A_O	CCP1A_O	-
P07	CCPnA_I / CCPnB_I		TXD		CCP0B_O	CCP1B_O	
P10	-		EPWM0	-	-	-	-
P11	-		EPWM1	-	-	-	-
P12	-		EPWM2	-	-	-	-
P13	-		EPWM3	-	-	-	-
P14	-		EPWM4	-	-	-	-
P15	-		EPWM5	-	-	-	-
P16	-		EPWM0	-	-	-	-
P20	-		-	-	-	-	-
P21	-		-	-	-	-	-
P22	-		-	-	-	-	-
P23	-		CCP0A_O	CCP1A_O	-	-	-
P24	BKIN		CCP0B_O	CCP1B_O	-	-	-
P25	-		-	-	-	-	-
P26	CCPnA_I / CCPnB_I /BKIN		C1_O	-	CCP0A_O	CCP1A_O	
P27	-		-	-	-	-	-

注1：本产品IO复用功能，需要用户单独配置PMC，PM，等寄存器
注2：当选择到IIC功能时，开漏功能自动开启
注3：针对复用功能，即可做输入又可做输出功能，选择好PmnCFG 后，输入通道自动开启

表3-2：模拟功能和特殊功能管脚

管脚	模拟			特殊功能管脚
	ADC	PGA	ACMP	
P00	AN8	A0P	C1P3	
P01	AN9	A0GND		
P02	AN10	A0O		RESINB
P03	AN11			
P04	AN12	A12O		
P05	AN13			
P06				SWDCLK
P07				SWDIO
P10				
P11				
P12				
P13				
P14				
P15				
P16	AN18			
P20	AN4		C0P0	
P21	AN5		C0P1	
P22	AN6		C0P2	
P23	AN7		C0N	
P24	AN14	A1P	C1P0	
P25	AN15	A1GND	C1P1	
P26	AN16	A2P	C1P2	
P27	AN17	A2GND	C1N	

3.3 寄存器映射

3.3.1 控制功能寄存器映射

（端口控制寄存器的基地址=0x40040000） RO：只读，WO：只写，R/W：读写

寄存器	偏移量	读/写	描述	复位值
P0	0x000	R/W	以位为单位，设定端口输出锁存器值的寄存器；输入模式中读此寄存器可以得到引脚电平，在输出模式中读此寄存器可得到端口的输出锁存器的值。	0x00
P1	0x001	R/W		0x00
P2	0x002	R/W		0x00
PM0	0x020	R/W	在端口作为数字通道使用时，以位为单位，设定端口输入或输出的寄存器。	0xFF
PM1	0x021	R/W		0xFF
PM2	0x022	R/W		0xFF
PU0	0x030	R/W	端口内部上拉电阻选择寄存器，只能在对应PMCmn=0时，设置上拉电阻有效。在产生复位信号后，P02的上拉功能默认打开。	0x04
PU1	0x031	R/W		0x00
PU2	0x032	R/W		0x00
PD0	0x040	R/W	端口内部下拉电阻选择寄存器，只能在对应PMCmn=0时，设置下拉电阻有效。在产生复位信号后，P06、P07的下拉功能默认打开。RESINB(P02)端口无下拉功能，配置PD0的bit2位无效	0xC0
PD1	0x041	R/W		0x00
PD2	0x042	R/W		0x00
POM0	0x050	R/W	开漏模式寄存器，只有在端口配置为输出模式时，N沟道漏极开路才会开启。当P03CFG=0X05或P04CFG=0X05，会强制开启P03或P04的开漏模式。	0x00
POM1	0x051	R/W		0x00
POM2	0x052	R/W		0x00
PMC0	0x060	R/W	端口模式寄存器，以位为单位设定端口作为数字或模拟通道使用；P00、P01、P02、P06、P07默认作为数字通道使用。	0x38
PMC1	0x061	R/W		0xFF
PMC2	0x062	R/W		0xFF
PSET0	0x070	W	以位为单位来置位端口输出锁存器的寄存器。	0x00
PSET1	0x071	W		0x00
PSET2	0x072	W		0x00
PCLR0	0x080	W	以位为单位来清零端口输出锁存器的寄存器。	0x00
PCLR1	0x081	W		0x00
PCLR2	0x082	W		0x00

3.3.2 输出输入复用功能寄存器映射

（输出输入复用功能寄存器的基地址=0x40040800） RO：只读，WO：只写，R/W：读写

寄存器	偏移量	读/写	描述	复位值
P00CFG	0x00	R/W	端口输出复用功能配置寄存器，可实现将一部分外围模块的输出功能映射到对应的端口上，具体管脚功能数字映射，参见表表3-1；端口输出复用功能寄存器的复位值为0x00，此时端口为默认的兼用功能和GPIO功能。	0x00
P01CFG	0x02	R/W		0x00
P02CFG	0x04	R/W		0x00
P03CFG	0x06	R/W		0x00
P04CFG	0x08	R/W		0x00
P05CFG	0x0A	R/W		0x00
P06CFG	0x0C	R/W		0x00
P07CFG	0x0E	R/W		0x00
P10CFG	0x10	R/W		0x00
P11CFG	0x12	R/W		0x00
P12CFG	0x14	R/W		0x00
P13CFG	0x16	R/W		0x00
P14CFG	0x18	R/W		0x00
P15CFG	0x1A	R/W		0x00
P16CFG	0x1C	R/W		0x00
P20CFG	0x20	R/W		0x00
P21CFG	0x22	R/W		0x00
P22CFG	0x24	R/W		0x00
P23CFG	0x26	R/W		0x00
P24CFG	0x28	R/W		0x00
P25CFG	0x2A	R/W		0x00
P26CFG	0x2C	R/W		0x00
P27CFG	0x2E	R/W		0x00
PS_{int0}_CFG	0x60	R/W	外部中断0输入端口复用寄存器，可以映射到任意端口。	0x3f
PS_{int1}_CFG	0x61	R/W	外部中断1输入端口复用寄存器，可以映射到任意端口。	0x3f
PS_{int2}_CFG	0x62	R/W	外部中断2输入端口复用寄存器，可以映射到任意端口。	0x3f
PS_{int3}_CFG	0x63	R/W	外部中断3输入端口复用寄存器，可以映射到任意端口。	0x3f
$PS_{tau0tin0}$_CFG	0x64	R/W	TAU0外部输入通道0的端口输入复用寄存器，可以映射到任意端口。	0x3f
$PS_{tau0tin1}$_CFG	0x65	R/W	TAU0外部输入通道1的端口输入复用寄存器，可以映射到任意端口。	0x3f
$PS_{tau0tin2}$_CFG	0x66	R/W	TAU0外部输入通道2的端口输入复用寄存器，可以映射到任意端口。	0x3f
$PS_{tau0tin3}$_CFG	0x67	R/W	TAU0外部输入通道3的端口输入复用寄存器，可以映射到任意端口。	0x3f
$PS_{uart0rxd}$_CFG	0x68	R/W	UART的rxd信号输入端口复用寄存器，可以映射到指定端口，具体映射参见表表3-1。	0x07
$PS_{epwmnkin}$_CFG	0x69	R/W	EPWM外部刹车输入端口复用寄存器，可以映射到指定端口，具体映射参见表3-1。	0x07
$PS_{ccp0ain}$_CFG	0x6A	R/W	CCP0通道a输入端口复用寄存器，可以映射到指定端口，具体映射参见表3-1。	0x07

$PS_{ccp0bin}$_CFG	0x6B	R/W	CCP0通道b输入端口复用寄存器，可以映射到指定端口，具体映射参见表3-1。	0x07
$PS_{ccp1ain}$_CFG	0x6C	R/W	CCP1通道a输入端口复用寄存器，可以映射到指定端口，具体映射参见表3-1。	0x07
$PS_{ccp1bin}$_CFG	0x6D	R/W	CCP1通道b输入端口复用寄存器，可以映射到指定端口，具体映射参见表3-1。	0x07
P0TTLCFG	0x70	R/W	P0端口输入电平选择寄存器。	0x00
P2TTLCFG	0x72	R/W	P2端口输入电平选择寄存器。	0x00
PMS	0x7B	R/W	端口读取模式选择寄存器；在端口为输出模式时，该寄存器选择读端口锁存器的值或引脚输出电平。详见27.3章节	0x00

3.3.3 特殊功能端口RESINB控制寄存器映射

（寄存器的基地址=0x40020400） RO：只读，WO：只写，R/W：读写

寄存器	偏移量	读/写	描述	复位值
RSTM	0x0B	R/W	选择RESINB(P02)口作为外部复位口或GPIO口的寄存器	0x00

3.4 寄存器说明

通过以下寄存器控制端口：

(1) 端口寄存器（Px）
(2) 端口模式寄存器（PMx）
(3) 上拉电阻选择寄存器（PUx）
(4) 下拉电阻选择寄存器（PDx）
(5) 端口输出模式寄存器（POMx）
(6) 端口模式控制寄存器（PMCx）
(7) 端口置位控制寄存器（PSETx）
(8) 端口清零控制寄存器（PCLRx）
(9) 端口输出复用功能配置寄存器（PxxCFG）
(10) 端口输入复用功能配置寄存器（PSxx_CFG）
(11) 端口电平选择寄存器（PxTTLCFG）
(12) 特殊功能端口RESINB控制寄存器(RSTM)

3.4.1 端口寄存器（Px）

这是以位为单位设定端口输出锁存器值的寄存器 Px（x=0～2）。在输入模式中读此寄存器可以得到引脚电平，而在输出模式中读可以得到端口的输出锁存器的值。在产生复位信号后，这些寄存器的值变为“00H”。

具体寄存器描述如下：

位	符号	描述	复位值
7	Px7	端口x的模式寄存器第7位 0：输出模式时输出“0”；输入模式时输入低电平 1：输出模式时输出“1”；输入模式时输入高电平	0
6	Px6	端口x的模式寄存器第6位 0：输出模式时输出“0”；输入模式时输入低电平 1：输出模式时输出“1”；输入模式时输入高电平	0
5	Px5	端口x的模式寄存器第5位 0：输出模式时输出“0”；输入模式时输入低电平 1：输出模式时输出“1”；输入模式时输入高电平	0
4	Px4	端口x的模式寄存器第4位 0：输出模式时输出“0”；输入模式时输入低电平 1：输出模式时输出“1”；输入模式时输入高电平	0
3	Px3	端口x的模式寄存器第3位 0：输出模式时输出“0”；输入模式时输入低电平 1：输出模式时输出“1”；输入模式时输入高电平	0
2	Px2	端口x的模式寄存器第2位 0：输出模式时输出“0”；输入模式时输入低电平 1：输出模式时输出“1”；输入模式时输入高电平	0
1	Px1	端口x的模式寄存器第1位 0：输出模式时输出“0”；输入模式时输入低电平 1：输出模式时输出“1”；输入模式时输入高电平	0
0	Px0	端口x的模式寄存器第0位 0：输出模式时输出“0”；输入模式时输入低电平 1：输出模式时输出“1”；输入模式时输入高电平	0

注：必须给未分配的位设定初始值。

3.4.2 端口模式寄存器（PMx）

当端口作为数字通道使用时，这是以位为单位设定其输入/输出的寄存器PMx（x=0~2）。在产生复位信号后，所有端口默认为输入状态。

具体寄存器描述如下：

位	符号	描述	复位值
7	PMx7	端口x的模式寄存器第7位 0：输出模式(用作输出端口（输出缓冲器ON）) 1：输入模式(用作输入端口（输出缓冲器OFF）)	1
6	PMx6	端口x的模式寄存器第6位 0：输出模式(用作输出端口（输出缓冲器ON）) 1：输入模式(用作输入端口（输出缓冲器OFF）)	1
5	PMx5	端口x的模式寄存器第5位 0：输出模式(用作输出端口（输出缓冲器ON）) 1：输入模式(用作输入端口（输出缓冲器OFF）)	1
4	PMx4	端口x的模式寄存器第4位 0：输出模式(用作输出端口（输出缓冲器ON）) 1：输入模式(用作输入端口（输出缓冲器OFF）)	1
3	PMx3	端口x的模式寄存器第3位 0：输出模式(用作输出端口（输出缓冲器ON）) 1：输入模式(用作输入端口（输出缓冲器OFF）)	1
2	PMx2	端口x的模式寄存器第2位 0：输出模式(用作输出端口（输出缓冲器ON）) 1：输入模式(用作输入端口（输出缓冲器OFF）)	1
1	PMx1	端口x的模式寄存器第1位 0：输出模式(用作输出端口（输出缓冲器ON）) 1：输入模式(用作输入端口（输出缓冲器OFF）)	1
0	PMx0	端口x的模式寄存器第0位 0：输出模式(用作输出端口（输出缓冲器ON）) 1：输入模式(用作输入端口（输出缓冲器OFF）)	1

注1：必须给未分配的位设定初始值。

注2：P17端口无效，PM1的bit7保持为1。

3.4.3 上拉电阻选择寄存器（PUx）

内部上拉电阻的选择寄存器 PUx（x=0～2）。只能在对应的 PMCx 对应位等于 0 的时候，设置上拉电阻有效。

在产生复位信号后，P02 端口上拉功能自动打开，其他端口的上拉功能默认不打开。

具体寄存器描述如下：

位	符号	描述	复位值
7	PUx7	Px引脚的内部上拉电阻选择第7位 0：不连接内部上拉电阻 1：连接内部上拉电阻	0
6	PUx6	Px引脚的内部上拉电阻选择第6位 0：不连接内部上拉电阻 1：连接内部上拉电阻	0
5	PUx5	Px引脚的内部上拉电阻选择第5位 0：不连接内部上拉电阻 1：连接内部上拉电阻	0
4	PUx4	Px引脚的内部上拉电阻选择第4位 0：不连接内部上拉电阻 1：连接内部上拉电阻	0
3	PUx3	Px引脚的内部上拉电阻选择第3位 0：不连接内部上拉电阻 1：连接内部上拉电阻	0
2	PUx2	Px引脚的内部上拉电阻选择第2位 0：不连接内部上拉电阻 1：连接内部上拉电阻	x=0,复位值为1 x=1或2,复位值为0
1	PUx1	Px引脚的内部上拉电阻选择第1位 0：不连接内部上拉电阻 1：连接内部上拉电阻	0
0	PUx0	Px引脚的内部上拉电阻选择第0位 0：不连接内部上拉电阻 1：连接内部上拉电阻	0

注 1：必须给未分配的位设定初始值。

注 2：P17 端口无效，

注 3：PU1 的 bit7 保持为 0。

3.4.4 下拉电阻选择寄存器（PDx）

内部下拉电阻的选择寄存器PDx（x=0～2）。只能在对应的PMCx对应位等于0 的时候，设置下拉电阻有效；RESINB(P02)端口无下拉功能。

在产生复位信号后，P06，P07端口下拉功能自动打开，其他端口的下拉功能默认不打开。

具体寄存器描述如下：

位	符号	描述	复位值
7	PDx7	Px引脚的内部下拉电阻选择第7位 0：不连接内部下拉电阻 1：连接内部下拉电阻	x=0,复位值为1 x=1或2,复位值为0
6	PDx6	Px引脚的内部下拉电阻选择第6位 0：不连接内部下拉电阻 1：连接内部下拉电阻	x=0,复位值为1 x=1或2,复位值为0
5	PDx5	Px引脚的内部下拉电阻选择第5位 0：不连接内部下拉电阻 1：连接内部下拉电阻	0
4	PDx4	Px引脚的内部下拉电阻选择第4位 0：不连接内部下拉电阻 1：连接内部下拉电阻	0
3	PDx3	Px引脚的内部下拉电阻选择第3位 0：不连接内部下拉电阻 1：连接内部下拉电阻	0
2	PDx2	Px引脚的内部下拉电阻选择第2位 0：不连接内部下拉电阻 1：连接内部下拉电阻 注：RESINB(P02)端口无下拉功能	0
1	PDx1	Px引脚的内部下拉电阻选择第1位 0：不连接内部下拉电阻 1：连接内部下拉电阻	0
0	PDx0	Px引脚的内部下拉电阻选择第0位 0：不连接内部下拉电阻 1：连接内部下拉电阻	0

注1：必须给未分配的位设定初始值。

注2：P17端口无效，PD1的bit7保持为0。

3.4.5 端口输出模式寄存器（POMx）

端口输出模式寄存器POMx（x=0~2），只有在配置成输出模式N沟道漏极开路才会开启。当P03CFG=0x05或P04CFG=0x05，会强制开启P03或P04的N沟道漏极开路模式。

在产生复位信号后，这些寄存器的值变为“00H”。

注：对于设定N沟道漏极开路输出模式（POMmn=1）的位，不连接内部上拉电阻。

具体寄存器描述如下：

位	符号	描述	复位值
7	POMx7	Px引脚输出模式选择第7位 0：通常输出模式 1：N沟道漏极开路输出模式	0
6	POMx6	Px引脚输出模式选择第6位 0：通常输出模式 1：N沟道漏极开路输出模式	0
5	POMx5	Px引脚输出模式选择第5位 0：通常输出模式 1：N沟道漏极开路输出模式	0
4	POMx4	Px引脚输出模式选择第4位 0：通常输出模式 1：N沟道漏极开路输出模式	0
3	POMx3	Px引脚输出模式选择第3位 0：通常输出模式 1：N沟道漏极开路输出模式	0
2	POMx2	Px引脚输出模式选择第2位 0：通常输出模式 1：N沟道漏极开路输出模式	0
1	POMx1	Px引脚输出模式选择第1位 0：通常输出模式 1：N沟道漏极开路输出模式	0
0	POMx0	Px引脚输出模式选择第0位 0：通常输出模式 1：N沟道漏极开路输出模式	0

注：P17端口无效，POM1的bit7保持为0

3.4.6 端口模式控制寄存器（PMCx）

端口模式寄存器 PMCx(x=0~2)，以位为单位设定端口作为数字（输入/输出）或者作为模拟（输入）通道。

在产生复位信号后，P00，P01，P02，P06，P07 默认作为数字通道使用（PMC00，PMC01，PMC02，PMC06，PMC07 复位值为“0”），其他端口默认作为模拟通道使用即 PMCx 对应位等于 1。

具体寄存器描述如下：

位	符号	描述	复位值
7	PMCx7	Px引脚数字（输入/输出）或模拟（输入）第7位 0：数字输入/输出（模拟输入以外的复用功能） 1：模拟输入	x=0,复位值为0 x=1或2,复位值为1
6	PMCx6	Px引脚数字（输入/输出）或模拟（输入）第6位 0：数字输入/输出（模拟输入以外的复用功能） 1：模拟输入	x=0,复位值为0 x=1或2,复位值为1
5	PMCx5	Px引脚数字（输入/输出）或模拟（输入）第5位 0：数字输入/输出（模拟输入以外的复用功能） 1：模拟输入	1
4	PMCx4	Px引脚数字（输入/输出）或模拟（输入）第4位 0：数字输入/输出（模拟输入以外的复用功能） 1：模拟输入	1
3	PMCx3	Px引脚数字（输入/输出）或模拟（输入）第3位 0：数字输入/输出（模拟输入以外的复用功能） 1：模拟输入	1
2	PMCx2	Px引脚数字（输入/输出）或模拟（输入）第2位 0：数字输入/输出（模拟输入以外的复用功能） 1：模拟输入	x=0,复位值为0 x=1或2,复位值为1
1	PMCx1	Px引脚数字（输入/输出）或模拟（输入）第1位 0：数字输入/输出（模拟输入以外的复用功能） 1：模拟输入	x=0,复位值为0 x=1或2,复位值为1
0	PMCx0	Px引脚数字（输入/输出）或模拟（输入）第0位 0：数字输入/输出（模拟输入以外的复用功能） 1：模拟输入	x=0,复位值为0 x=1或2,复位值为1

注1：必须给未分配的位设定初始值。

注2：P17端口无效，PMC1的bit7保持为1。

3.4.7 端口置位控制寄存器（PSETx）

这是以位为单位来置位端口输出锁存器的寄存器PSETx（x=0～2）。在产生复位信号后，这些寄存器的值变为“00H”。

具体寄存器描述如下：

位	符号	描述	复位值
7	PSETx7	Px引脚的置位控制第7位 0：无操作 1：对应的Px7置1	0
6	PSETx6	Px引脚的置位控制第6位 0：无操作 1：对应的Px6置1	0
5	PSETx5	Px引脚的置位控制第5位 0：无操作 1：对应的Px5置1	0
4	PSETx4	Px引脚的置位控制第4位 0：无操作 1：对应的Px4置1	0
3	PSETx3	Px引脚的置位控制第3位 0：无操作 1：对应的Px3置1	0
2	PSETx2	Px引脚的置位控制第2位 0：无操作 1：对应的Px2置1	0
1	PSETx1	Px引脚的置位控制第1位 0：无操作 1：对应的Px1置1	0
0	PSETx0	Px引脚的置位控制第0位 0：无操作 1：对应的Px0置1	0

注1：必须给未分配的位设定初始值。

注2：P17端口无效，PSET1的bit7保持为0。

3.4.8 端口清零控制寄存器（PCLRx）

这是以位为单位来置位端口输出锁存器的寄存器PCLRx（x=0～2）。在产生复位信号后，这些寄存器的值变为“00H”。

具体寄存器描述如下：

位	符号	描述	复位值
7	PCLRx7	Px引脚的清零控制第7位 0：无操作 1：对应的Px7清零	0
6	PCLRx6	Px引脚的清零控制第6位 0：无操作 1：对应的Px6清零	0
5	PCLRx5	Px引脚的清零控制第5位 0：无操作 1：对应的Px5清零	0
4	PCLRx4	Px引脚的清零控制第4位 0：无操作 1：对应的Px4清零	0
3	PCLRx3	Px引脚的清零控制第3位 0：无操作 1：对应的Px3清零	0
2	PCLRx2	Px引脚的清零控制第2位 0：无操作 1：对应的Px2清零	0
1	PCLRx1	Px引脚的清零控制第1位 0：无操作 1：对应的Px1清零	0
0	PCLRx0	Px引脚的清零控制第0位 0：无操作 1：对应的Px0清零	0

注1：必须给未分配的位设定初始值。

注2：P17端口无效，PCLR1的bit7保持为0。

3.4.9 端口输出复用功能配置寄存器（PmnCFG）

端口输出复用功能配置寄存器PmnCFG(m=0~2,n=0~7.m=1时,n!=7)，可实现将一部分外围模块的输出功能映射到端口，具体映射功能参见表3-1；端口输出复用功能配置寄存器的复位值为“00H”，此时端口为默认的兼用功能和GPIO功能。设置步骤如下：

（1)设置PMCmn=0，选择数字输入/输出

（2)设置PMmn=0，设置输出模式

（3)设置PmnCFG选择对应管脚的复位功能输出

注：IIC的SDA，SCL以及SPI所有功能脚既可以做输出也可做输入，在设置PmnCFG后，输入功能将自动开启，除以上3个步骤外，无需其他操作。

具体寄存器描述如下：

位	符号	描述	复位值
7:3	--	保留	--
2:0	PmnCFG[2:0]	Pmn引脚输出复用功能 0x00： Pmn对应输出映射表=0x00的功能 0x01： Pmn对应输出映射表=0x01的功能 0x02： Pmn对应输出映射表=0x02的功能 0x03： Pmn对应输出映射表=0x03的功能 0x04： Pmn对应输出映射表=0x04的功能 0x05： Pmn对应输出映射表=0x05的功能 其他： 禁止选择	0x0

3.4.10 端口输入复用功能配置寄存器（PSxx_CFG）

INTP0、INTP1、INTP2、INTP3、TI00、TI01、TI02、TI03可映射到任意的GPIO输入；RXD、BKIN、CCP0AIN、CCP0BIN、CCP1AIN、CCP1BIN可映射到指定GPIO输入，具体输入功能映射参见表3-1；由于每个功能都有多管脚输入，因此在用输入复用功能时需要选择具体的GPIO输入，设置步骤如下：

（1)设置PMCmn = 0

（2)设置PMmn=1

（3)设置PSxx_CFG选择对应的管脚输入

PSintp0_CFG 描述如下：

位	符号	描述	复位值
7:6	--	保留	--
5:0	PSintp0_CFG[5:0]	INTP0选择GPIO输入 0x00： 选择P00作为INTP0输入 0x01： 选择P01作为INTP0输入 … … 0x26： 选择P26作为INTP0输入 0x27： 选择P27作为INTP0输入 其他： 输入低电平	0x3F

PSintp1_CFG 描述如下：

位	符号	描述	复位值
7:6	--	保留	--
5:0	PSintp1_CFG[5:0]	INTP1选择GPIO输入 0x00： 选择P00作为INTP1输入 0x01： 选择P01作为INTP1输入 … … 0x26： 选择P26作为INTP1输入 0x27： 选择P27作为INTP1输入 其他： 输入低电平	0x3F

PSintp2_CFG 描述如下：

位	符号	描述	复位值
7:6	--	保留	--
5:0	PSintp2_CFG[5:0]	INTP2选择GPIO输入 0x00： 选择P00作为INTP2输入 0x01： 选择P01作为INTP2输入 … … 0x26： 选择P26作为INTP2输入 0x27： 选择P27作为INTP2输入 其他： 输入低电平	0x3F

PSintp3_CFG 描述如下：

位	符号	描述	复位值
7:6	--	保留	--
5:0	PSintp3_CFG[5:0]	INTP3选择GPIO输入 0x00： 选择P00作为INTP3输入 0x01： 选择P01作为INTP3输入 … … 0x26： 选择P26作为INTP3输入 0x27： 选择P27作为INTP3输入 其他： 输入低电平	0x3F

PStau0tin0_CFG 描述如下：

位	符号	描述	复位值
7:6	--	保留	--
5:0	PStau0tin0_CFG [5:0]	TI00选择GPIO输入 0x00： 选择P00作为TI00输入 0x01： 选择P01作为TI00输入 … … 0x26： 选择P26作为TI00输入 0x27： 选择P27作为TI00输入 其他： TI00输入低电平	0x3F

PStau0tin1_CFG 描述如下：

位	符号	描述	复位值
7:6	--	保留	--
5:0	PStau0tin1_CFG[5:0]	TI01选择GPIO输入 0x00： 选择P00作为TI01输入 0x01： 选择P01作为TI01输入 … … 0x26： 选择P26作为TI01输入 0x27： 选择P27作为TI01输入 其他： TI01输入低电平	0x3F

PStau0tin2_CFG 描述如下：

位	符号	描述	复位值
7:6	--	保留	--
5:0	PStau0tin2_CFG [5:0]	TI02选择GPIO输入 0x00： 选择P00作为TI02输入 0x01： 选择P01作为TI02输入 … … 0x26： 选择P26作为TI02输入 0x27： 选择P27作为TI02输入 其他： TI02输入低电平	0x3F

PStau0tin3_CFG 描述如下：

位	符号	描述	复位值
7:6	--	保留	--
5:0	PStau0tin3_CFG [5:0]	TI03选择GPIO输入 0x00： 选择P00作为TI03输入 0x01： 选择P01作为TI03输入 … … 0x26： 选择P26作为TI03输入 0x27： 选择P27作为TI03输入 其他： TI03输入低电平	0x3F

PSuart0rxd_CFG 描述如下：

位	符号	描述	复位值
7:3	--	保留	--
2:0	PSuart0rxd_CFG [2:0]	UART0_RXD选择GPIO输入 0x00： 选择P03作为UART_RXD输入 0x01： 选择P06作为UART_RXD输入 其他： UART0_RXD输入高电平	0x07

PSepwmbkin_CFG 描述如下：

位	符号	描述	复位值
7:3	--	保留	--
2:0	PSepwmbkin_CFG [2:0]	EPWM_BKIN选择GPIO输入 0x00： 选择P00作为EPWM_BKIN输入 0x01： 选择P24作为EPWM_BKIN输入 0x02： 选择P26作为EPWM_BKIN输入 其他： EPWM_BKIN输入低电平	0x07

PSccp0ain_CFG 描述如下：

位	符号	描述	复位值
7:3	--	保留	--
2:0	PSccp0a_i_CFG [2:0]	CCP0AIN选择GPIO输入 0x00： 选择P02作为CCP0A_I输入 0x01： 选择P04作为CCP0A_I输入 0x02： 选择P06作为CCP0A_I输入 0x03： 选择P07作为CCP0A_I输入 0x04： 选择P26作为CCP0A_I输入 其他： CCP0AIN输入低电平	0x7

PSccp0bin_CFG 描述如下：

位	符号	描述	复位值
7:3	--	保留	--
2:0	PSccp0b_i_CFG [2:0]	CCP0BIN选择GPIO输入 0x00： 选择P02作为CCP0BIN输入 0x01： 选择P04作为CCP0BIN输入 0x02： 选择P06作为CCP0BIN输入 0x03： 选择P07作为CCP0BIN输入 0x04： 选择P26作为CCP0BIN输入 其他： CCP0BIN输入低电平	0x7

PSccp1ain_CFG 描述如下：

位	符号	描述	复位值
7:3	--	保留	--
2:0	PSccp1ain_CFG [2:0]	CCP1AIN选择GPIO输入 0x00： 选择P02作为CCP1AIN输入 0x01： 选择P04作为CCP1AIN输入 0x02： 选择P06作为CCP1AIN输入 0x03： 选择P07作为CCP1AIN输入 0x04： 选择P26作为CCP1AIN输入 其他： CCP1AIN输入低电平	0x7

PSccp1bin_CFG 描述如下：

位	符号	描述	复位值
7:3	--	保留	--
2:0	PSccp1bin_CFG [2:0]	CCP1BIN选择GPIO输入 0x00： 选择P02作为CCP1BIN输入 0x01： 选择P04作为CCP1BIN输入 0x02： 选择P06作为CCP1BIN输入 0x03： 选择P07作为CCP1BIN输入 0x04： 选择P26作为CCP1BIN输入 其他： CCP1BIN输入低电平	0x7

3.4.11 TTL和施密特输入选择(PxTTLCFG,x=0和2)

PxTTLCFG 选择寄存器描述如下：

位	符号	描述	复位值
7	PxTTL7	Px7 输入电平选择 0： 施密特输入 1： TTL输入	0
6	PxTTL6	Px6 输入电平选择 0： 施密特输入 1： TTL输入	0
5	PxTTL5	Px5 输入电平选择 0： 施密特输入 1： TTL输入	0
4	PxTTL4	Px4 输入电平选择 0： 施密特输入 1： TTL输入	0
3	PxTTL3	Px3 输入电平选择 0： 施密特输入 1： TTL输入	0
2	PxTTL2	Px2 输入电平选择 0： 施密特输入 1： TTL输入	0
1	PxTTL1	Px1 输入电平选择 0： 施密特输入 1： TTL输入	0
0	PxTTL0	Px0 输入电平选择 0： 施密特输入 1： TTL输入	0

3.4.12 特殊功能端口RESINB说明(RSTM)

本产品上电默认RESINB（P02）有效，如果需要把该口做GPIO使用，那么需要通过寄存器关掉复位功能，寄存器描述如下：

位	符号	描述	复位值
7：1	--	保留为0	0x0
0	RSTM	RESINB引脚的外部复位功能屏蔽 0： RESINB外部复位引脚 1： RESINB做GPIO引脚	0

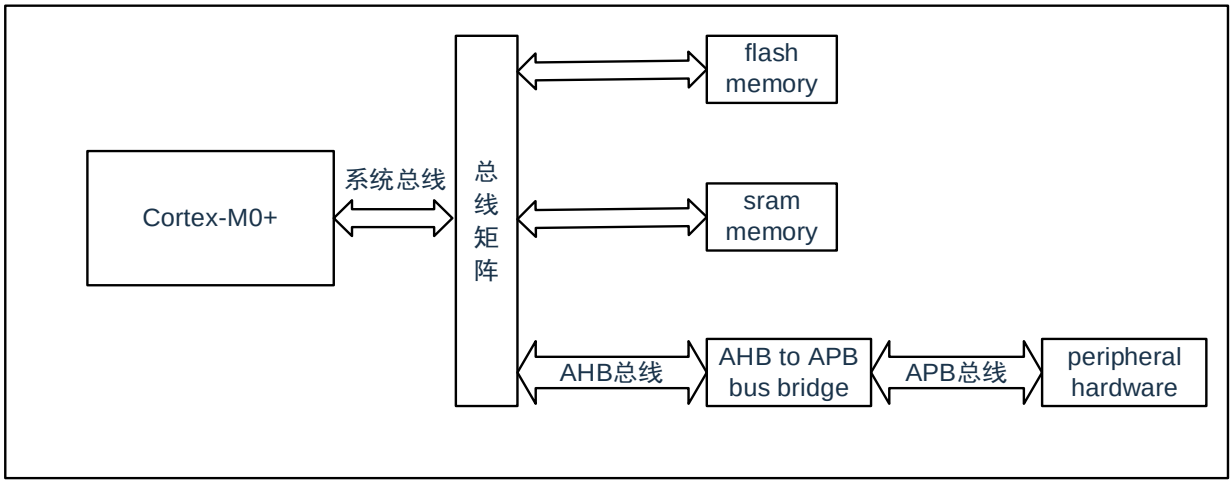
第4章 系统结构

4.1 概述

本产品系统由以下部分组成：

- 1个AHB总线Master：
  - Cortex-M0+
- 3个AHB总线Slaves：
  - FLASH存储器
  - SRAM存储器
  - AHB to APB Bridge，包含所有APB接口外设

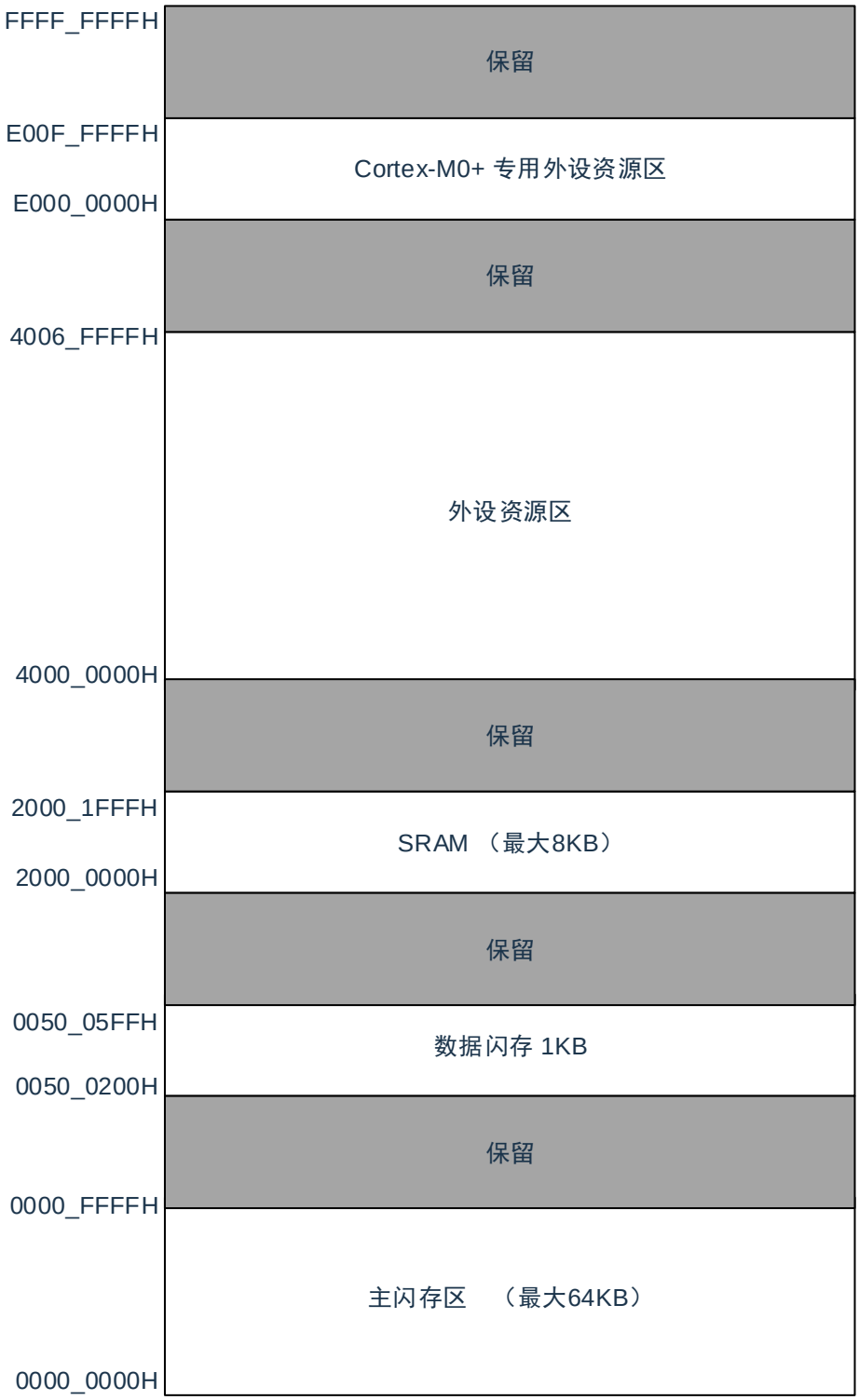
图4-1：系统结构示意图



1) 系统总线: 此总线连接Cortex-M0+内核的系统总线(外设总线)到总线矩阵。
2) 总线矩阵: 总线矩阵协调内核系统总线其他总线的访问。
3) AHB to APB Bridge ：AHB to APB Bridge 在AHB和APB总线间提供同步连接。有关连接到每个桥的不同外设的地址映射请参考表4-1。

4.2 系统地址划分

图4-2：地址区域划分示意图



外设地址分配

表4-1：外设的寄存器组起始地址

起始地址	外设	备注
0x4000_0000 - 0x4000_4FFF	保留	
0x4000_5000 - 0x4000_5FFF	保留	
0x4000_6000 - 0x4000_6FFF	中断控制	
0x4000_7000 - 0x4001_8FFF	保留	
0x4001_9000 - 0x4001_9FFF	保留	
0x4001_A000 - 0x4001_FFFF	保留	
0x4002_0000 - 0x4002_03FF	FLASH 控制	
0x4002_0400 - 0x4002_0FFF	时钟控制	
0x4002_1000 - 0x4002_1001	看门狗定时器	
0x4002_1002 - 0x4002_1800	保留	
0x4002_1800 - 0x4002_1BFF	高速 CRC	详见第 27 章 安全功能
0x4002_1C00 - 0x4002_1FFF	时钟控制	
0x4002_2000 - 0x4003_FFFF	保留	
0x4004_0000 - 0x4004_0FFF	GPIO	
0x4004_1D00 - 0x4004_1FFF	定时器阵列 0	
0x4004_2200 - 0x4004_23FF	保留	
0x4004_2800 - 0x4004_31FF	保留	
0x4004_3200 - 0x4004_32FF	通用 CRC	详见第 27 章 安全功能
0x4004_3300 - 0x4004_33FF	保留	
0x4004_3C00 - 0x4004_3FFF	保留	
0x4004_4000 - 0x4004_43FF	保留	
0x4004_4B50 -0x4004_4B50	LSITIMER	
0x4004_4800 - 0x4004_4EFF	保留	
0x4004_5400 - 0x4004_5AFF	保留	
0x4004_5B00 - 0x4004_5BFF	外部中断控制	
0x4008_0000 - 0x4008_01FF	保留	
0x4008_0200 - 0xDFFF_FFFF	保留	
-		
0x4006_1000 – 0x4006_1FFF	TIMER0	
0x4006_2000 – 0x4006_2FFF	保留	
0x4006_3000 – 0x4006_3FFF	SPI	
0x4006_4000 – 0x4006_40FF	UART	
0x4006_4100 – 0x4006_41FF	保留	
0x4006_4200 – 0x4006_427F	EPWM	
0x4006_4280 – 0x4006_42FF	CCP	
0x4006_4300 – 0x4006_433F	IIC	
0x4006_4340 – 0x4006_437F	保留	

0x4006_4380 – 0x4006_43BF	DIV	
	保留	
0x4006_8000 – 0x4006_80FF	ADC	
0x4006_8100 – 0x4006_81FF	保留	
0x4006_8200 – 0x4006_823F	ACMP0	
0x4006_8240 – 0x4006_827F	保留	
-	保留	
0x4006_8300 – 0x4006_831F	PGA0	
0x4006_8320 – 0x4006_833F	PGA12	
0x4006_8340 – 0x4006_835F	ADCLDO	
0x4006_8360 – 0x4006_837F	DAC	
-		
-		

第5章 时钟发生电路

5.1 时钟发生电路的功能

时钟发生电路是产生给CPU和外围硬件提供时钟的电路。有以下2种系统时钟和时钟振荡电路。

(1) 主系统时钟高速内部振荡器（高速OCO）

能通过选项字节（000C2H）从F_{HOCO}=64MHz、48MHz、32MHz、24MHz、16MHz、12MHz、8MHz、6MHz、4MHz、3MHz和2MHz (TYP.)中选择频率进行振荡。在解除复位后，CPU一定以此高速内部振荡器时钟开始运行。能通过进入深度睡眠模式或者设置HIOSTOP位（CSC寄存器的bit0）使振荡停止。能通过高速内部振荡器的频率选择寄存器（HOCODIV）更改选项字节设置的频率。有关频率设置，请参照“高内部振荡器的频率选择寄存器（HOCODIV）的设置”。

(2) 低速内部振荡器时钟（低速OCO）

能使F_{IL}=15kHz的时钟振荡。

可以将低速内部振荡器时钟用作系统时钟。

当选项字节（000C0H）的bit4（WDTON）或者副系统时钟提供模式控制寄存器（OSMC）的bit4（WUTMMCK0）为“1”时，或者副系统时钟选择寄存器(SUBCKSEL)的bit0(SELLOSC)为“1”时，低速内部振荡器振荡。

但是，在WDTON位为“1”并且WUTMMCK0位为“0”而且选项字节（000C0H）的bit0（WDSTBYON）为“0”时，如果进入深度睡眠模式或睡眠模式，低速内部振荡器就停止振荡。

注：F_{HOCO}：高速内部振荡器的时钟频率

F_{IH}：高速内部振荡器的时钟频率

F_{IL}：低速内部振荡器的时钟频率

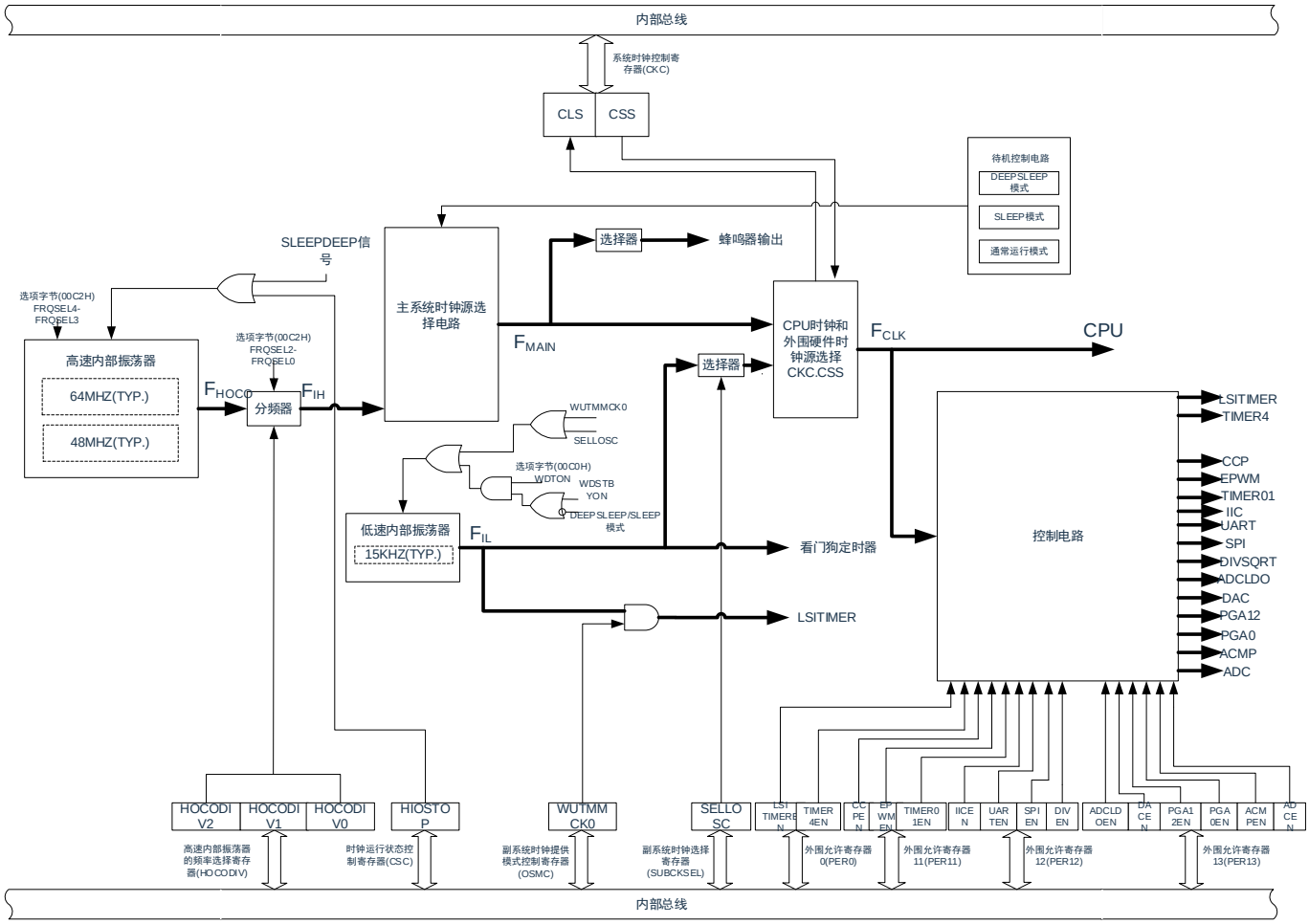
5.2 时钟发生电路的结构

时钟发生电路由以下硬件构成。

表5-1：时钟发生电路的结构

项目	结构
控制寄存器	系统时钟控制寄存器（CKC） 时钟运行状态控制寄存器（CSC） 外围允许寄存器0、1（PER0、PER11、PER12、PER13） 副系统时钟提供模式控制寄存器（OSMC） 高速内部振荡器的频率选择寄存器（HOCODIV） 副系统时钟选择寄存器（SUBCKSEL）
振荡电路	高速内部振荡器 低速内部振荡器

图5-1：时钟发生电路的框图



注：F_{HOCO}：高速内部振荡器的时钟频率

F_{IH}：高速内部振荡器的时钟频率

F_{MAIN}：主系统时钟频率

F_{CLK}：CPU/外围硬件的时钟频率

F_{IL}：低速内部振荡器的时钟频率

5.3 寄存器映射

(以下寄存器基地址 = 0x4002_0400)　　RO：只读，WO：只写，R/W：读写

寄存器	偏移量	读/写	描述	复位值
CSC	0x001	R/W	时钟运行状态控制寄存器	0xC0
CKC	0x004	R/W	系统时钟控制寄存器	0x0
SUBCKSEL	0x007	R/W	副系统时钟选择寄存器	0x0
PMUKEY	0x008	R/W	供电模式控制保护寄存器	0x0
PMCCTL	0x00A	R/W	供电模式控制寄存器	0x0
PER0	0x020	R/W	外围允许寄存器0	0x0
OSMC	0x023	R/W	副系统时钟提供模式控制寄存器	0x0

(以下寄存器基地址 = 0x4002_0810)　　RO：只读，WO：只写，R/W：读写

寄存器	偏移量	读/写	描述	复位值
PER11	0x001	R/W	外围允许寄存器1	0x0
PER12	0x002	R/W	外围允许寄存器1	0x0
PER13	0x003	R/W	外围允许寄存器1	0x0

(以下寄存器基地址 = 0x4002_1C00)　　RO：只读，WO：只写，R/W：读写

寄存器	偏移量	读/写	描述	复位值
HOCODIV	0x020	R/W	高速内部振荡器的频率选择寄存器	选项字节（000C2H）的FRQSEL2~FRQSEL0 位的设置值

5.4 寄存器说明

通过以下寄存器控制时钟发生电路。

- 系统时钟控制寄存器（CKC）
- 时钟运行状态控制寄存器（CSC）
- 外围允许寄存器0、1（PER0、PER11、PER12、PER13）
- 副系统时钟提供模式控制寄存器（OSMC）
- 高速内部振荡器的频率选择寄存器（HOCODIV）
- 副系统时钟选择寄存器（SUBCKSEL）

注：分配的寄存器和位因产品而不同。必须给未分配的位设置初始值。

5.4.1 系统时钟控制寄存器（CKC）

这是选择CPU/外围硬件时钟和主系统时钟的寄存器。

通过8位存储器操作指令设置CKC寄存器。

位	符号	描述	复位值
7	CLS[注1]	CPU/外围硬件时钟(F_{CLK})的状态 0：主系统时钟(F_{MAIN}) 1：低速内部振荡器时钟 (F_{IL})	0
6	CSS	CPU/外围硬件时钟(F_{CLK})的选择 0：主系统时钟(F_{MAIN}) 1：低速内部振荡器时钟 (F_{IL})	0
5:0	--	保留	0x0

注1：bit7是只读位。

注2：必须将bit0～5置“0”。

注3：给CPU和外围硬件提供CSS位设置的时钟。如果更改CPU时钟，就同时更改外围硬件的时钟（时钟输出/蜂鸣器输出和看门狗定时器除外）。因此，如果要更改CPU/外围硬件的时钟，就必须停止各外围功能。

5.4.2 时钟运行状态控制寄存器（CSC）

这是控制高速内部振荡器时钟运行的寄存器。通过8位存储器操作指令设置CSC寄存器。

在产生复位信号后，此寄存器的值变为“C0H”。

位	符号	描述	复位值
7:1	--	保留	0x60
0	HIOSTOP	高速内部振荡器时钟的运行控制 0：高速内部振荡器运行 1：高速内部振荡器停止	0

注1：不能通过CSC寄存器停止被选择为CPU/外围硬件时钟（F_{CLK}）的时钟。

注2：有关用于停止时钟振荡的寄存器标志设置和停止前的条件，请参照表5-2。

表5-2：时钟停止方法

时钟	时钟停止前的条件	设置CSC寄存器的标志
高速内部振荡器时钟	CPU/外围硬件时钟以高速内部振荡器时钟以外的时钟运行。（CLS=1）	HIOSTOP=1

5.4.3 外围允许寄存器0、1（PER0、PER11，PER12，PER13）

这是设置允许或者禁止给各外围硬件提供时钟的寄存器。通过停止给不使用的硬件提供时钟，以降低功耗和噪声。当使用由这些寄存器控制的以下外围功能时，必须在进行外围功能的初始设置前将对应位置“1”。

- LSITIMER 12位间隔定时器
- 通用定时器单元TIMER4
- CCP0/1
- EPWM
- TIMER01
- IIC
- UART
- SPI
- DIVSQRT
- ADCLDO
- DAC
- PGA12
- PGA0
- ACMP0/1
- ADC

通过8位存储器操作指令设置PER0寄存器和PER11、PER12、PER13寄存器。

在产生复位信号后，这些寄存器的值变为“00H”。

外围允许寄存器0（PER0）

位	符号	描述	复位值
7	LSITIMEREN	提供LSITIMER输入时钟的控制(掉电睡眠可行) 0：停止提供输入时钟 •不能写LSITIMER使用的SFR。 •LSITIMER处于复位状态。 1：提供输入时钟 •能写LSITIMER使用的SFR。	0
6:1	--	保留	0x0
0	TIM40EN	提供通用定时器单元4输入时钟的控制 0：停止提供输入时钟 •不能写通用定时器单元4使用的SFR。 •通用定时器单元4处于复位状态。 1：提供输入时钟 •能写通用定时器单元4使用的SFR。	0

外围允许寄存器1（PER11）

位	符号	描述	复位值
7:5	--	保留	0x0
4	CCPEN	提供CCP模块输入时钟的控制 0：停止提供输入时钟 • CCP不能运行 1：提供输入时钟 • CCP能运行	0
3	EPWMEN	提供EPWM模块输入时钟的控制 0：停止提供输入时钟 • EPWM不能运行 1：提供输入时钟 • EPWM能运行	0
2:1	--	保留	0x0
0	TIMER01EN	提供TIMER01的输入时钟的控制 0：停止提供输入时钟 • TIMER01不能运行 1：提供输入时钟 • TIMER01能运行	0

外围允许寄存器1（PER12）

位	符号	描述	复位值
7:5	--	保留	0x0
4	IICEN	提供IIC模块输入时钟的控制 0：停止提供输入时钟 • IIC不能运行 1：提供输入时钟 • IIC能运行	0
3	--	保留	0
2	UARTEN	提供UART模块输入时钟的控制 0：停止提供输入时钟 • UART不能运行 1：提供输入时钟 •UART能运行	0
1	SPIEN	提供SPI模块输入时钟的控制 0：停止提供输入时钟 • SPI不能运行 1：提供输入时钟 • SPI能运行	0
0	DIVEN	提供除法与开方运算单元DIVSQRT的输入时钟的控制 0：停止提供输入时钟 • DIVSQRT不能运行 1：提供输入时钟 • DIVSQRT能运行	0

外围允许寄存器1（PER13）

位	符号	描述	复位值
7	--	保留	0
6	ADCLDOEN	提供ADCLDO模块输入时钟的控制 0：停止提供输入时钟 • ADCLDO不能运行 1：提供输入时钟 • ADCLDO能运行	0
5	DACEN	提供DAC模块输入时钟的控制 0：停止提供输入时钟 • DAC不能运行 1：提供输入时钟 • DAC能运行	
4	PGA12EN	提供PGA12模块输入时钟的控制 0：停止提供输入时钟 • PGA12不能运行 1：提供输入时钟 • PGA12能运行	0
3	PGA0EN	提供PGA0模块输入时钟的控制 0：停止提供输入时钟 • PGA0不能运行 1：提供输入时钟 • PGA0能运行	0
2	--	保留	0
1	ACMPEN	提供ACMP0/1模块输入时钟的控制 0：停止提供输入时钟 •ACMP0/1不能运行 1：提供输入时钟 • ACMP0/1能运行	0
0	ADCEN	提供ADC模块的输入时钟的控制 0：停止提供输入时钟 • ADC不能运行 1：提供输入时钟 • ADC能运行	0

5.4.4 12位间隔定时器运行时钟选择寄存器（OSMC）

通过OSMC寄存器选择12位间隔定时器LSITIMER的运行时钟。

通过8位存储器操作指令设置OSMC寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

位	符号	描述	复位值
7:5	--	保留	0x0
4	WUTMMCK0	12位间隔定时器的运行时钟的选择 0：低速内部振荡时钟停止给12位间隔定时器LSITIMER提供时钟 1：低速内部振荡时钟为12位间隔定时器LSITIMER提供时钟	0
3:0	--	保留	0x0

5.4.5 高速内部振荡器的频率选择寄存器（HOCODIV）

这是更改选项字节（000C2H）设置的高速内部振荡器频率的寄存器。但是，能选择的频率因选项字节（000C2H）的FRQSEL4位和FRQSEL3位的值而不同。

通过8位存储器操作指令设置HOCODIV寄存器。

在产生复位信号后，此寄存器的值变为选项字节（000C2H）的FRQSEL2～FRQSEL0位的设置值。

<table>
<tr><th>位</th><th>符号</th><th colspan="3">描述</th><th>复位值</th></tr>
<tr><td>7:3</td><td>--</td><td colspan="3">保留</td><td>0x0</td></tr>
<tr><td rowspan="8">2:0</td><td rowspan="8">HOCODIV2~
HOCODIV0</td><td colspan="3">高速内部振荡器时钟频率的选择</td><td rowspan="8">选项字节（000C2H）的FRQSEL2～FRQSEL0位的设置值</td></tr>
<tr><td></td><td>FSQSEL4,3=00</td><td>FSQSEL4,3=01</td></tr>
<tr><td>000</td><td>F_{IH}=48MHZ
F_{HOCO}=48MHZ</td><td>F_{IH}=64MHZ
F_{HOCO}=64MHZ</td></tr>
<tr><td>001</td><td>F_{IH}=24MHZ
F_{HOCO}=48MHZ</td><td>F_{IH}=32MHZ
F_{HOCO}=64MHZ</td></tr>
<tr><td>010</td><td>F_{IH}=12MHZ
F_{HOCO}=48MHZ</td><td>F_{IH}=16MHZ
F_{HOCO}=64MHZ</td></tr>
<tr><td>011</td><td>F_{IH}=6MHZ
F_{HOCO}=48MHZ</td><td>F_{IH}=8MHZ
F_{HOCO}=64MHZ</td></tr>
<tr><td>100</td><td>F_{IH}=3MHZ
F_{HOCO}=48MHZ</td><td>F_{IH}=4MHZ
F_{HOCO}=64MHZ</td></tr>
<tr><td>101</td><td>禁止设置</td><td>F_{IH}=2MHZ
F_{HOCO}=64MHZ</td></tr>
<tr><td>上述以外</td><td colspan="2">禁止设置</td></tr>
</table>

注1：必须在选择高速内部振荡器时钟（F_{IH}）作为CPU/外围硬件时钟（F_{CLK}）的状态下设置HOCODIV寄存器。

注2：在通过HOCODIV寄存器更改频率后，经过以下转移时间之后进行频率切换：

- 以更改前的频率，最多进行3个时钟的运行。
- 以更改后的频率，最多等待3个CPU/外围硬件的时钟。

5.4.6 低速内部振荡器时钟选择寄存器（SUBCKSEL）

SUBCKSEL寄存器是选择低速内部振荡器时钟FIL的寄存器。

通过8位存储器操作指令设置SUBCKSEL寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

位	符号	描述	复位值
7:1	--	保留	0x0
0	SELLOSC	低速内部振荡器时钟选择 0：禁止选择低速内部振荡器时钟 1：选择低速内部振荡器时钟	0

5.4.7 供电模式控制保护寄存器（PMUKEY）

PMUKEY寄存器是供电模式控制PMUCTL的保护的寄存器。

通过16位存储器操作指令设置PMUKEY寄存器。

在产生复位信号后，此寄存器的值变为“0000H”。

位	符号	描述	复位值
15:0	PMUKEY	供电模式控制保护寄存器的选择 •解除 PMUCTL写入保护。通过先后写入192AH、3E4FH到PMUKEY使能PMUCTL的PWDNEN位的写入控制 •其他。PMUCTL写入设置无效	0x0

5.4.8 供电模式控制寄存器（PMUCTL）

PMUCTL寄存器是控制使能供电控制模式的寄存器。

通过8位存储器操作指令设置PMUCTL寄存器。

在产生复位信号后，此寄存器的值变为“00H”并且写入保护打开，通过PMUKEY解除写入控制。

位	符号	描述	复位值
7:1	--	保留	0x0
0	PWDNEN	供电模式控制寄存器的选择 0：部分掉电模式禁止 1：部分掉电模式使能	0

注：通过PMUKEY解除PMUCTL的写入保护。

5.5 系统时钟振荡电路

5.5.1 高速内部振荡器

CMS32M65xx内置高速内部振荡器。能通过选项字节(000C2H)从64MHz、48MHz、32MHz、24MHz、16MHz、12MHz、8MHz、6MHz、4MHz、3MHz和2MHz中选择频率。能通过时钟运行状态控制寄存器(CSC)的bit0(HIOSTOP)控制振荡。

在解除上电复位后，高速内部振荡器自动开始振荡。

5.5.2 低速内部振荡器

CMS32M65xx内置低速内部振荡器。

低速内部振荡器时钟用作看门狗定时器、LSITIMER，以及SysTick定时器的外部参考时钟，也可用作CPU时钟和外围模块时钟。

当选项字节(000C0H)的bit4(WDTON)或者副系统时钟提供模式控制寄存器(OSMC)的bit4（WUTMMCK0）为“1”时，或者副系统时钟选择寄存器(SUBCKSEL)的bit0(SELLOSC)为“1”时，低速内部振荡器振荡。

当看门狗定时器停止运行并且WUTMMCK0位不为“0”时，低速内部振荡器继续振荡。但是，如果看门狗定时器运行而WUTMMCK0位为“0”， SELLOSC位为“0”，就在WDSTBYON位为“0”并且处于睡眠模式、深度睡眠模式时低速内部振器停止振荡。在看门狗定时器运行时，即使程序失控，低速内部振荡器时钟也不停止运行。

5.6 时钟发生电路的运行

时钟发生电路产生以下所示各种时钟，并且控制待机模式等CPU的运行模式（参照图5-1）。

F_{MAIN}：主系统时钟

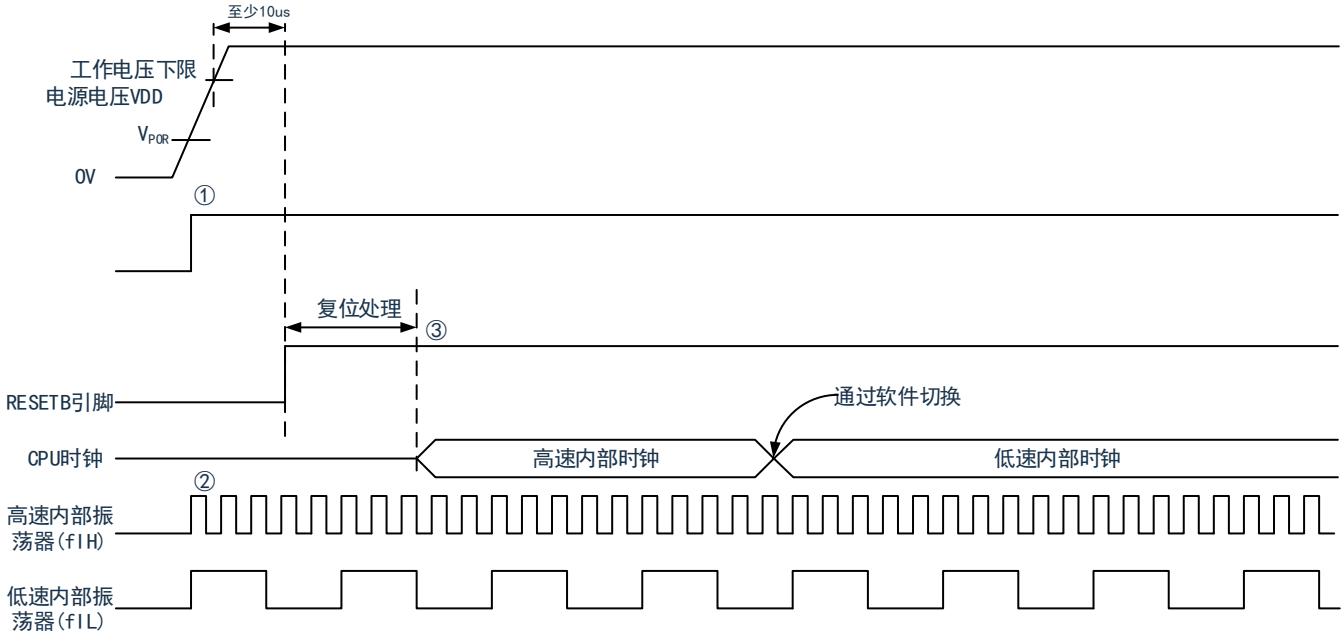
F_{IH}：高速内部振荡器时钟

F_{IL}：低速内部振荡器时钟

F_{CLK}：CPU/外围硬件时钟

CMS32M65xx 在解除复位后，CPU 通过高速内部振荡器的输出开始运行。接通电源时的时钟发生电路的运行如图 5-2 所示。

图5-2：接通电源时的时钟发生电路的运行



1) 在接通电源后，通过上电复位(POR)电路产生内部复位信号。
2) 但是，在达到数据手册的 AC 特性所示的工作电压范围前，通过电压检测电路或者外部复位保持复位状态（上图是使用外部复位时的例子）。
3) 如果解除复位，高速内部振荡器就自动开始振荡。
4) 在解除复位后，进行电压稳定等待和复位处理，然后 CPU 以高速内部振荡器时钟开始运行。

5.7 时钟控制

5.7.1 高速内部振荡器的设置例子

在解除复位后，CPU/外围硬件时钟（F_{CLK}）一定以高速内部振荡器时钟运行。能通过选项字节（000C2H）的FRQSEL0～FRQSEL4位，从64MHz、48MHz、32MHz、24MHz、16MHz、12MHz、8MHz、6MHz、4MHz、3MHz和2MHz中选择高速内部振荡器的频率。另外，能通过高速内部振荡器的频率选择寄存器（HOCODIV）更改频率。

【选项字节000C2的设置】

<table>
<tr><th>位</th><th>符号</th><th colspan="3">描述</th><th>复位值</th></tr>
<tr><td>7:5</td><td>--</td><td colspan="3">保留</td><td>--</td></tr>
<tr><td rowspan="14">4:0</td><td rowspan="14">FRQSEL4~ FRQSEL0</td><td colspan="3">高速内部振荡器时钟频率的选择</td><td rowspan="14">--</td></tr>
<tr><td></td><td>F_{HOCO}</td><td>F_{IH}</td></tr>
<tr><td>01000</td><td>64MHZ</td><td>64MHZ</td></tr>
<tr><td>00000</td><td>48MHZ</td><td>48MHZ</td></tr>
<tr><td>01001</td><td>64MHZ</td><td>32MHZ</td></tr>
<tr><td>00001</td><td>48MHZ</td><td>24MHZ</td></tr>
<tr><td>01010</td><td>64MHZ</td><td>16MHZ</td></tr>
<tr><td>00010</td><td>48MHZ</td><td>12MHZ</td></tr>
<tr><td>01011</td><td>64MHZ</td><td>8MHZ</td></tr>
<tr><td>00011</td><td>48MHZ</td><td>6MHZ</td></tr>
<tr><td>01100</td><td>64MHZ</td><td>4MHZ</td></tr>
<tr><td>00100</td><td>48MHZ</td><td>3MHZ</td></tr>
<tr><td>01101</td><td>64MHZ</td><td>2MHZ</td></tr>
<tr><td>上述以外</td><td colspan="2">禁止设置</td></tr>
</table>

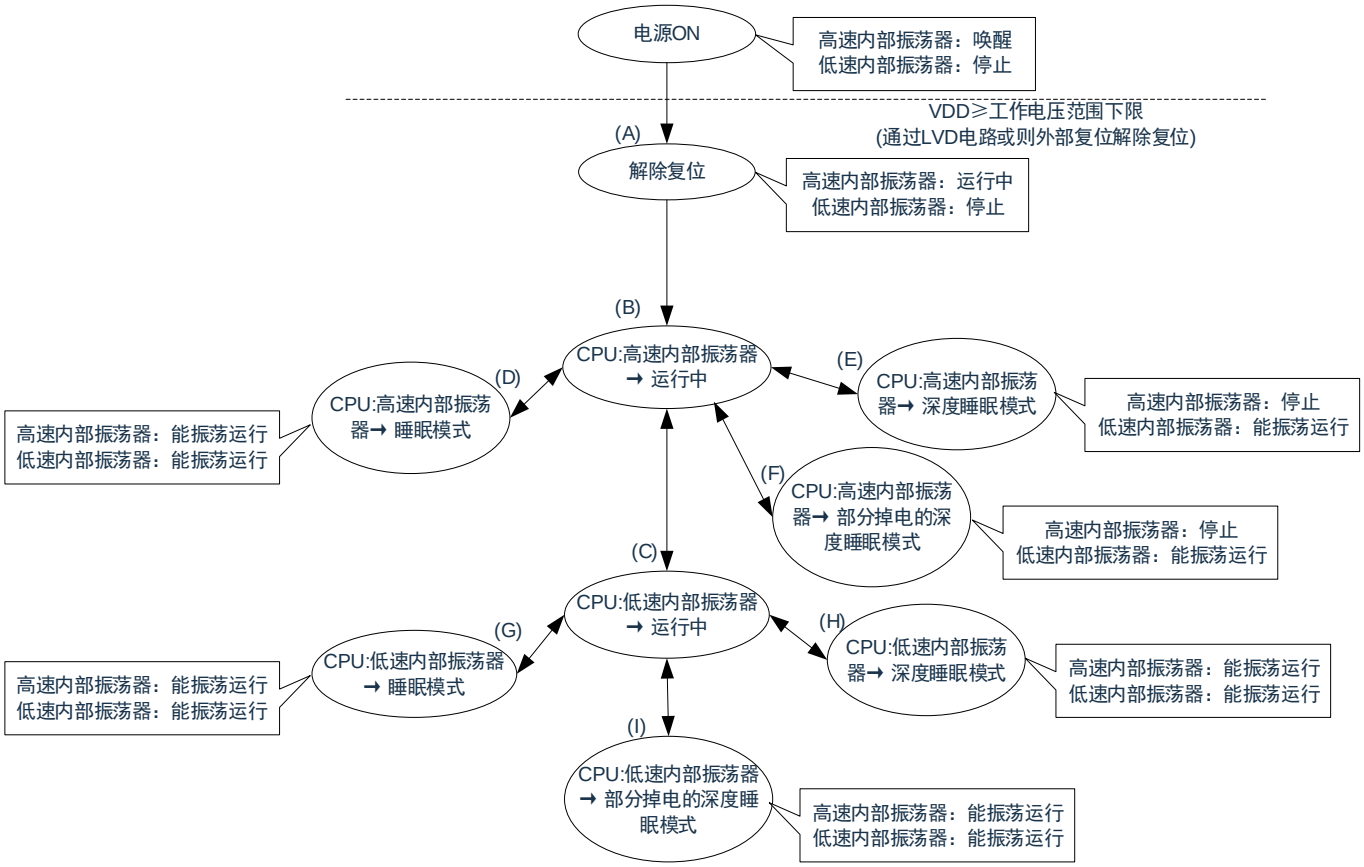
【高速内部振荡器的频率选择寄存器（HOCODIV）的设置】

<table>
<tr><th>位</th><th>符号</th><th colspan="3">描述</th><th>复位值</th></tr>
<tr><td>7:3</td><td>--</td><td colspan="3">保留</td><td>0x0</td></tr>
<tr><td rowspan="9">2:0</td><td rowspan="9">HOCODIV2~
HOCODIV0</td><td colspan="3">高速内部振荡器时钟频率的选择</td><td rowspan="9">选项字节
（000C2H）
的
FRQSEL2～
FRQSEL0位
的设置值</td></tr>
<tr><td rowspan="2"></td><td colspan="2">FSQSEL4=0</td></tr>
<tr><td>FSQSEL3=0</td><td>FSQSEL3=1</td></tr>
<tr><td>000</td><td>F_{IH}=48MHZ
F_{HOCO}=48MHZ</td><td>F_{IH}=64MHZ
F_{HOCO}=64MHZ</td></tr>
<tr><td>001</td><td>F_{IH}=24MHZ
F_{HOCO}=48MHZ</td><td>F_{IH}=32MHZ
F_{HOCO}=64MHZ</td></tr>
<tr><td>010</td><td>F_{IH}=12MHZ
F_{HOCO}=48MHZ</td><td>F_{IH}=16MHZ
F_{HOCO}=64MHZ</td></tr>
<tr><td>011</td><td>F_{IH}=6MHZ
F_{HOCO}=48MHZ</td><td>F_{IH}=8MHZ
F_{HOCO}=64MHZ</td></tr>
<tr><td>100</td><td>F_{IH}=3MHZ
F_{HOCO}=48MHZ</td><td>F_{IH}=4MHZ
F_{HOCO}=64MHZ</td></tr>
<tr><td>101</td><td>禁止设置</td><td>F_{IH}=2MHZ
F_{HOCO}=64MHZ</td></tr>
<tr><td>上述以外</td><td colspan="2">禁止设置</td></tr>
</table>

5.7.2 CPU时钟的状态转移图

本产品的CPU时钟状态转移图如图5-3所示。

图5-3：CPU时钟的状态转移图



CPU时钟的转移和SFR寄存器的设置例子等如表5-3所示。

表5-3：CPU时钟的转移和SFR寄存器的设置例子(1/3)

(1)在解除复位(A)后，CPU转移到高速内部振荡器时钟运行(B)。

状态转移	SFR寄存器的设置
(A)→(B)	不需要设置SFR寄存器（解除复位后的初始状态）。

(2)在解除复位(A)后，CPU转移到低速内部振荡器时钟运行(C)。（CPU在解除复位后立即以高速内部振荡器时钟运行(B)）

（SFR寄存器的设置顺序）→

SFR寄存器的设置标志 / 状态转移	SUBCKSEL寄存器	CKC寄存器
	SELLOSC	CSS
(B)→(C)	1	1

表5-3：CPU时钟的转移和SFR寄存器的设置例子(2/3)

(3) CPU从高速内部时钟运行(B)转移到低速内部时钟运行(C)。

（SFR寄存器的设置顺序）→

SFR寄存器的设置标志 / 状态转移	SUBCKSEL寄存器	CKC寄存器
	SELLOSC	CSS
(B)→(C)	1	1

(4) CPU从低速内部时钟运行(C)转移到高速内部时钟运行(B)。

(SFR寄存器的设置顺序）→

SFR寄存器的设置标志 / 状态转移	CSC寄存器	振荡精度稳定的等待	CKC寄存器
	HIOSTOP		CSS
(C)→(B)	0	1us	0

注1：表5-3的(A)～(I)对应图5-3的(A)～(I)。

注2：高速内部振荡器时钟的振荡精度稳定等待因温度条件和深度睡眠模式期间而变。

(5) CPU从高速内部时钟运行(B)转移到睡眠模式(D)。

CPU从低速内部时钟运行(C)转移到睡眠模式(G)。

状态转移	设置内容
(B)→(D) (C)→(G)	执行WFI指令。

注：表5-3的(A)～(I)对应图5-3的(A)～(I)。

(6) CPU从高速内部时钟运行(B)转移到深度睡眠模式(E)。

CPU从低速内部时钟运行(C)转移到深度睡眠模式(H)。

（设置顺序）→

状态转移	设置内容	
(B)→(E) (C)→(H)	停止不能在深度睡眠模式中运行的外围功能。	SCR寄存器bit2（SLEEPDEEP）置为1，并执行WFI指令。

注：表5-3的(A)～(I)对应图5-3的(A)～(I)。

表5-3：CPU时钟的转移和SFR寄存器的设置例子(3/3)

(7) CPU从高速内部时钟运行(B)转移到部分掉电的深度睡眠模式(F)。
CPU从低速内部时钟运行(C)转移到部分掉电的深度睡眠模式(I)。

（设置顺序）

状态转移	设置内容		
(B)→(E) (C)→(H)	停止不能在深度睡眠模式中运行的外围功能。	PMUKEY=0x192A; PMUKEY=0x3E4F; PMUCTL=0x01;	SCR 寄存器 bit2（SLEEPDEEP）置为1，并执行WFI指令。

注：表5-3的(A)～(I)对应图5-3的(A)～(I)。

5.7.3 CPU时钟转移前的条件和转移后的处理

CPU时钟转移前的条件和转移后的处理如下所示。

表5-4：有关CPU时钟的转移

CPU时钟		转移前的条件	转移后的处理
转移前	转移后		
高速内部振荡时钟	低速内部振荡时钟	选择低速内部振荡时钟 SELLOSC =1	如果停止高速内部振荡器的振荡（HIOSTOP=1），就能减小工作电流。
低速内部振荡时钟	高速内部振荡时钟	允许高速内部振荡器振荡。 • HIOSTOP=0 •经过振荡稳定时间后	若看门狗和LSITIMER没有运行，可关闭低速内部时钟振荡(SELLOSC =0) 就能减小工作电流。

5.7.4 CPU时钟和主系统时钟的切换所需时间

能通过设置系统时钟控制寄存器(CKC)的bit6 (CSS)进行CPU时钟的切换(主系统时钟和低速内部振荡时钟，即高速内部振荡时钟和低速内部振荡时钟)

在改写CKC寄存器后不立即进行实际的切换，而是在更改CKC寄存器后仍然以切换前的时钟继续运行数个时钟(参照表5-5)。

能通过CKC寄存器的bit7(CLS)来判断CPU是以主系统时钟 (高速内部振荡时钟)还是以低速内部振荡时钟运行。

如果切换CPU时钟，就同时切换外围硬件时钟。

表5-5：F_{MAIN} ←→ F_{IL}所需要的最大时钟数

切换前的设置值	切换后的设置值	
CSS	CLS	
	0 （F_{CLK}=F_{MAIN}）	1 （F_{CLK}=F_{IL}）
0 （F_{CLK}=F_{MAIN}）		2.5个F_{MAIN}/F_{IL}时钟
1 （F_{CLK}=F_{IL}）	2个时钟	

注1：表5-5中的时钟数是切换前的CPU时钟数。

注2：表5-5中的时钟数是舍入小数部分的时钟数。

例：CPU 从主系统时钟切换到副系统时钟的情况（选择 F_{IH}=2MHz、F_{SUB}= F_{IL} =15KHz 振荡的情况）

2.5F_{MAIN}/F_{SUB}=2.5(2000/15)=333.3→334个时钟

5.7.5 时钟振荡停止前的条件

用于停止时钟振荡的寄存器标志设置和停止前的条件如下所示。

表5-6：时钟振荡停止前的条件和标志设置

时钟	时钟停止前的条件	SFR寄存器的标志
高速内部振荡器时钟	CLS=1 （CPU以高速内部振荡器时钟以外的时钟运行）	HIOSTOP=1
低速内部振荡器时钟	CLS=0,WDTON=0,WUMMCK0=0 （CPU以低速内部振荡器时钟以外的时钟运行）	SELLOSC =0

第6章 通用定时器单元的功能

6.1 通用定时器的功能

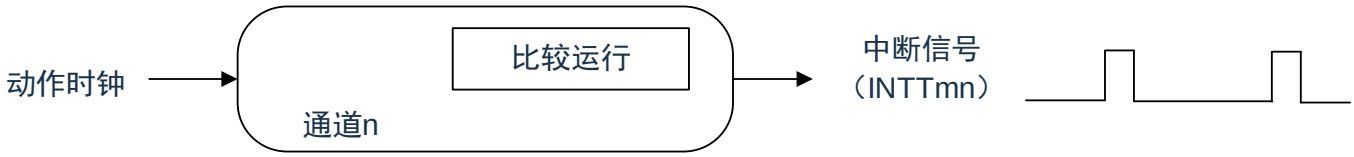
通用定时器单元有以下功能：

6.1.1 独立通道运行功能

独立通道运行功能是不受其他通道运行模式的影响而能独立使用任意通道的功能。

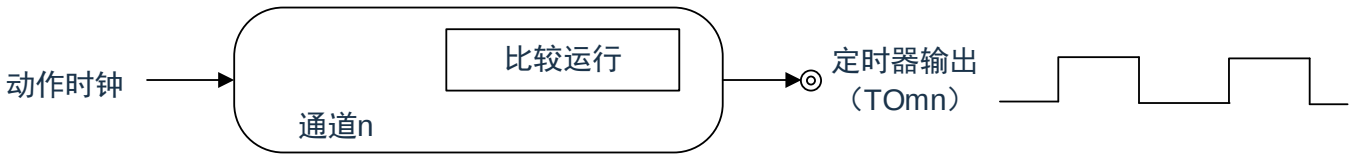
(1) 间隔定时器

能用作以固定间隔产生中断（INTTMmn）的基准定时器。



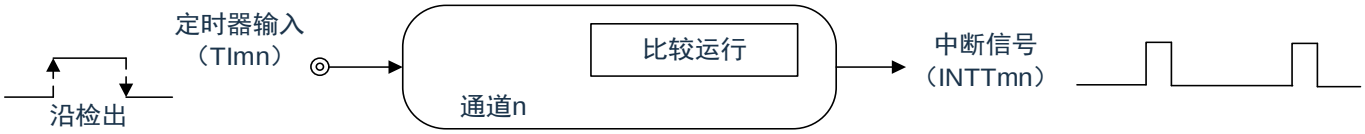
(2) 方波输出

每当产生INTTMmn 中断时，就进行交替运行并且从定时器的输出引脚（TOmn）输出50% 占空比的方波。



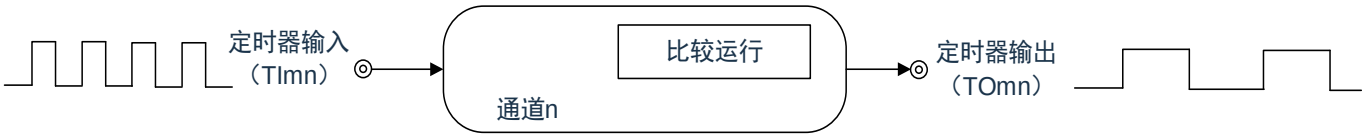
(3) 外部事件计数器

对定时器输入引脚 （TImn）的输入信号的有效边沿进行计数，如果达到规定次数，就能用作产生中断的事件计数器。



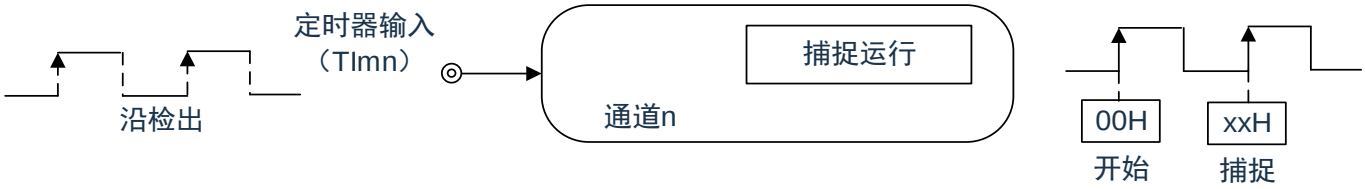
(4) 分频器功能 （只限于单元 0 的通道 0）

对定时器输入引脚（TI00）的输入时钟进行分频，然后从输出引脚（TO00）输出。



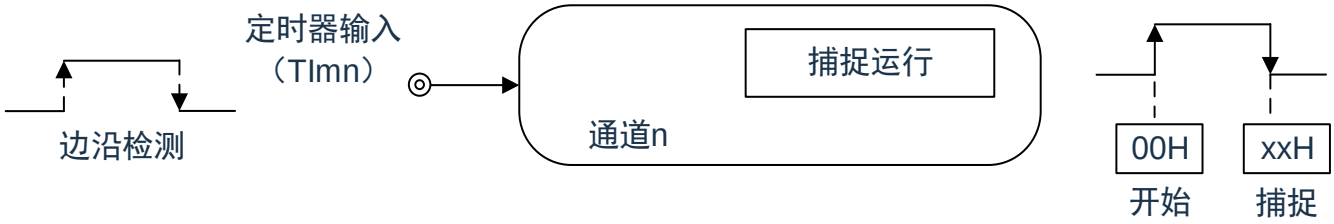
(5) 输入脉冲间隔的测量

在定时器输入引脚 （TImn）的输入脉冲信号的有效边沿开始计数并且在下一个脉冲的有效边沿捕捉计数值，从而测量输入脉冲的间隔。



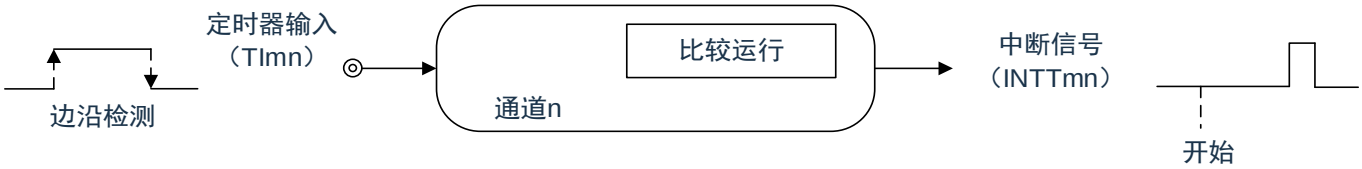
(6) 输入信号的高低电平宽度的测量

在定时器输入引脚（TImn）的输入信号的一个边沿开始计数并且在另一个边沿捕捉计数值，从而测量输入信号的高低电平宽度。



(7) 延迟计数器

在定时器输入引脚（TImn）的输入信号的有效边沿开始计数并且在经过任意延迟期间后产生中断。



注1：m：单元号（m=0） n：通道号（n=0～3）

注2： 通道0～3的定时器输入/输出引脚可配置请参照“第3章 引脚功能”。

6.1.2 多通道联动运行功能

多通道联动运行功能是将主控通道（主要控制周期的基准定时器）和从属通道（遵从主控通道运行的定时器）组合实现的功能。

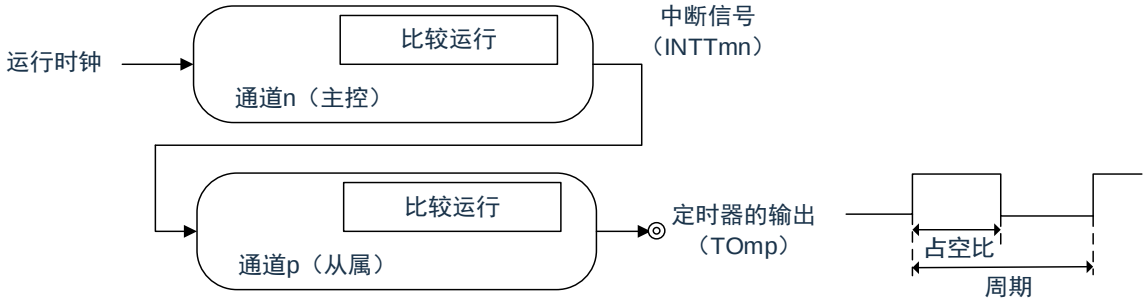
多通道联动运行功能可用作以下模式。

(1) 单触发脉冲输出

将2 个通道成对使用，生成能任意设定输出时序和脉宽的单触发脉冲。

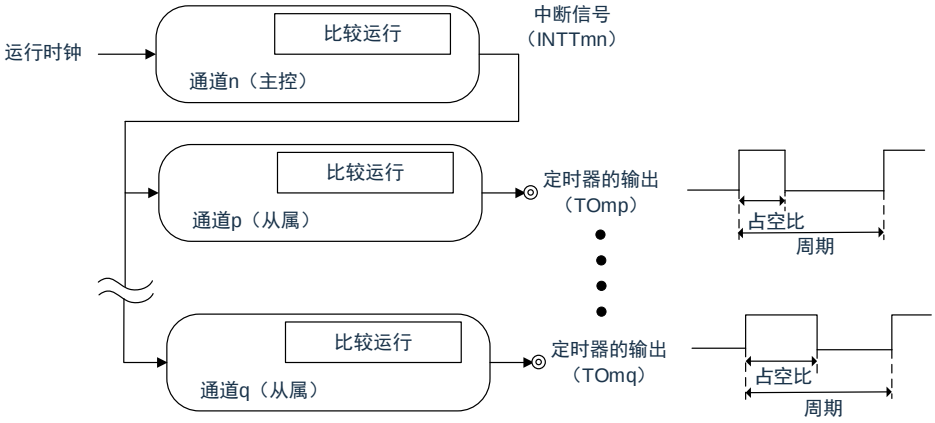
(2) PWM （Pulse Width Modulation）输出

将2 个通道成对使用，生成能任意设定周期和占空比的脉冲。



(3) 多重 PWM （Pulse Width Modulation）输出

能通过扩展PWM 功能并且使用1 个主控通道和多个从属通道，以固定周期生成最多3 种任意占空比的PWM信号。



注1：有关多通道联动运行功能规则的详细内容，请参照 “6.3.1 多通道联动运行功能的基本规则”。

注2：m：单元号（m=0） n：通道号（n=0～3） p、q：从属通道号（n＜p＜q≤3）

6.1.3 8 位定时器运行功能（只限于通道1和通道3）

8 位定时器运行功能是将16 位定时器通道用作2 个8 位定时器通道的功能。只能使用通道1 和通道3。

注：在使用8 位定时器运行功能时，有几个规则。
详细内容请参照“6.3.2 8位定时器运行功能的基本规则（只限于通道1和通道3）”。

6.2 通用定时器单元的结构

通用定时器单元由以下硬件构成。

表6-1：通用定时器单元的结构

项目	结构
计数器	定时器计数寄存器mn（TCRmn）
寄存器	定时器数据寄存器mn（TDRmn）
定时器的输入	TI00～TI03[注1]
定时器的输出	TO00～TO03[注1]、输出控制电路
控制寄存器	＜单元设定部的寄存器＞ • 外围允许寄存器0（PER0） • 定时器时钟选择寄存器m（TPSm） • 定时器通道允许状态寄存器m（TEm） • 定时器通道开始寄存器m（TSm） • 定时器通道停止寄存器m（TTm） • 定时器输入选择寄存器0（TIOS0）[注2] • 定时器输出允许寄存器m（TOEm） • 定时器输出寄存器m（TOm） • 定时器输出电平寄存器m（TOLm） • 定时器输出模式寄存器m（TOMm）
	＜每个通道的寄存器＞ • 定时器模式寄存器mn（TMRmn） • 定时器状态寄存器mn（TSRmn） • 噪声滤波器允许寄存器1、2（NFEN1、NFEN2） • 端口模式控制寄存器（PMCxx）[注3] • 端口模式寄存器（PMxx）[注3] • 端口输出复用功能配置寄存器（PxxCFG）[注3] • 端口输入复用功能配置寄存器（TI1XPCFG）[注3]

注1：通用定时器单元0的输入/输出引脚复用到固定端口。详细内容请参照“第3章 引脚功能”。

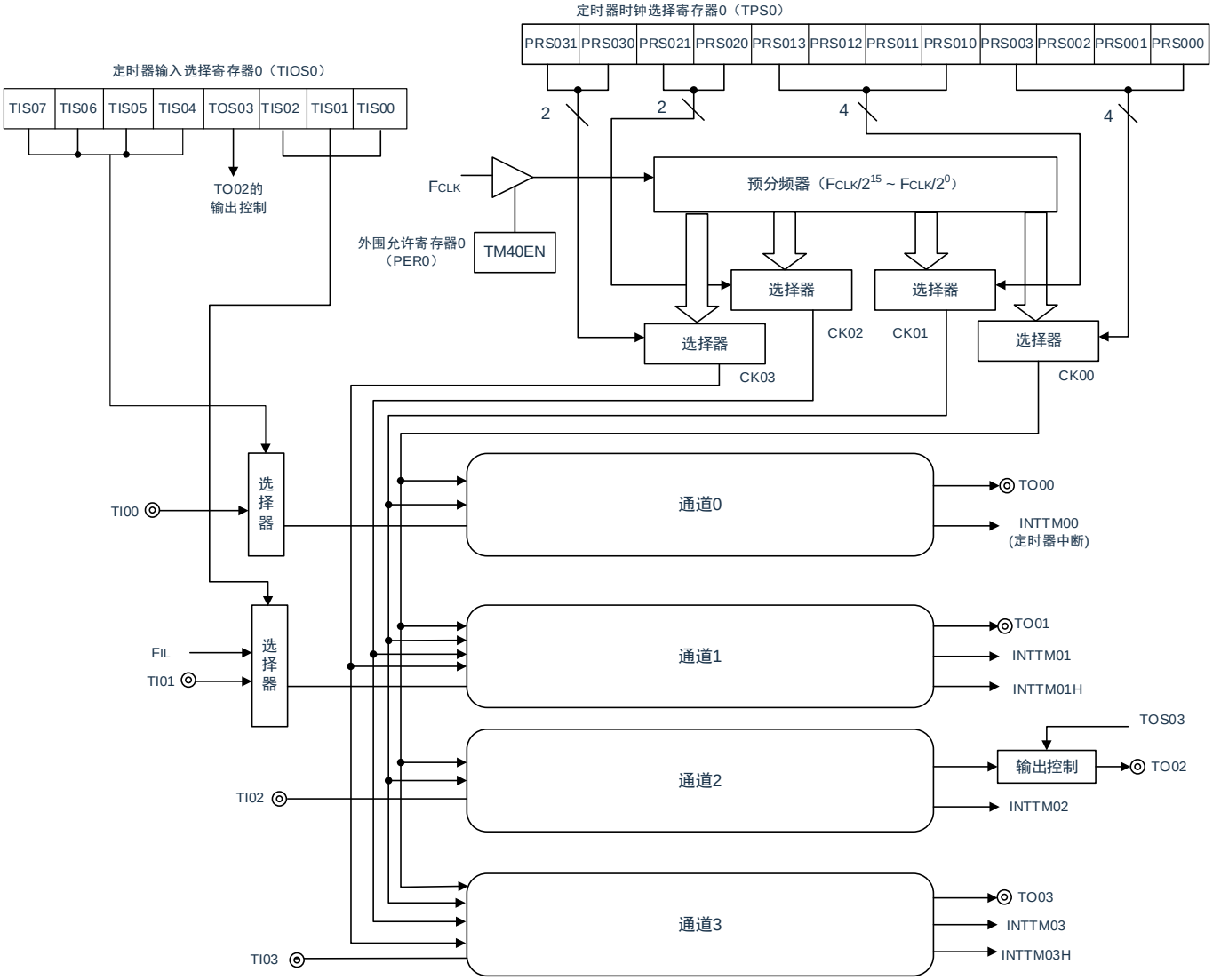
注2：仅用于单元0的通道选择

注3：用于通道0～3的定时器输入/输出引脚配置。详细内容请参照“第3章 引脚功能”。

注4：m：单元号（m=0） n：通道号（n=0～3）

通用定时器单元的框图如图 6-1 所示。

图6-1：通用定时器单元0的整体框图



6.2.1 寄存器映射

(以下寄存器基地址 = 0x4004_1D80)　　RO：只读，WO：只写，R/W：读写

寄存器名	偏移地址	读写属性	位宽	描述	复位值
TCR00	0x000	R	16	定时器通道 0 计数寄存器	FFFFH
TCR01	0x002	R	16	定时器通道 1 计数寄存器	FFFFH
TCR02	0x004	R	16	定时器通道 2 计数寄存器	FFFFH
TCR03	0x006	R	16	定时器通道 3 计数寄存器	FFFFH
TMR00	0x010	R/W	16	定时器通道 0 模式寄存器	0000H
TMR01	0x012	R/W	16	定时器通道 1 模式寄存器	0000H
TMR02	0x014	R/W	16	定时器通道 2 模式寄存器	0000H
TMR03	0x016	R/W	16	定时器通道 3 模式寄存器	0000H
TSR00	0x020	R	16	定时器通道 0 状态寄存器	0000H
TSR01	0x022	R	16	定时器通道 1 状态寄存器	0000H
TSR02	0x024	R	16	定时器通道 2 状态寄存器	0000H
TSR03	0x026	R	16	定时器通道 3 状态寄存器	0000H
TE0	0x030	R	16	定时器通道允许状态寄存器	0000H
TS0	0x032	R/W	16	定时器通道开始寄存器	0000H
TT0	0x034	R/W	16	定时器通道停止寄存器	0000H
TPS0	0x036	R/W	16	定时器时钟选择寄存器	0000H
TO0	0x038	R/W	16	定时器输出寄存器	0000H
TOE0	0x03A	R/W	16	定时器输出允许寄存器	0000H
TOL0	0x03C	R/W	16	定时器输出电平寄存器	0000H
TOM0	0x03E	R/W	16	定时器输出模式寄存器	0000H
TOM0L	0x03E	R/W	8	定时器输出模式寄存器低 8 位	00H
TDR00	0x198	R/W	16	定时器通道 0 数据寄存器	0000H
TDR01	0x19A	R/W	16	定时器通道 1 数据寄存器	0000H
TDR02	0x1E4	R/W	16	定时器通道 2 数据寄存器	0000H
TDR03	0x1E6	R/W	16	定时器通道 3 数据寄存器	0000H

(以下寄存器基地址 = 0x4004_0470)　　RO：只读，WO：只写，R/W：读写

寄存器	偏移量	读/写	描述	复位值
NFEN1	0x001	R/W	噪声滤波允许寄存器1	0x0
TIOS0	0x004	R/W	定时器输入输出选择寄存器	0x0

6.2.2 定时器计数寄存器mn（TCRmn）

TCRmn寄存器是对计数时钟进行计数的16位只读寄存器。与计数时钟的上升沿同步进行递增或者递减计数。

通过定时器模式寄存器mn（TMRmn）的MDmn3～MDmn0位来选择运行模式，进行递增和递减计数的切换（参照“6.2.6定时器模式寄存器mn（TMRmn）”）。

表6-2：定时器计数寄存器 mn （TCRmn）的表格

位	符号	描述	复位值
15:0	TCRmn	计数时钟计数寄存器(只读)	0xFFFF

注：m：单元号（m=0） n：通道号（n=0～3）

能通过读定时器计数寄存器mn（TCRmn）来读计数值。

在以下情况下，计数值变为“FFFFH”。

- 当产生复位信号时
- 当清除外围允许寄存器0（PER0）的TM4mEN位时
- 在PWM输出模式中从属通道的计数结束时
- 在延迟计数模式中从属通道的计数结束时
- 在单触发脉冲输出模式中主控/ 从属通道的计数结束时
- 在多重PWM输出模式中从属通道的计数结束时

在以下情况下，计数值变为“0000H”。

- 在捕捉模式中输入开始触发时
- 在捕捉模式中捕捉结束时

注：即使读TCRmn寄存器，也不将计数值捕捉到定时器数据寄存器mn（TDRmn）。

如下所示，TCRmn寄存器的读取值因运行模式和运行状态而不同。

表6-3：各运行模式中的定时器计数寄存器mn（TCRmn）的读取值

运行模式	计数方式	定时器计数寄存器mn（TCRmn）的读取值注			
		解除复位后更改运行模式时的值	计数暂停（TTmn=1）时的值	计数暂停（TTmn=1）后更改运行模式时的值	单次计数后等待 开始触发时的值
间隔定时器模式	递减计数	FFFFH	停止时的值	不定值	—
捕捉模式	递增计数	0000H	停止时的值	不定值	—
事件计数器模式	递减计数	FFFFH	停止时的值	不定值	—
单次计数模式	递减计数	FFFFH	停止时的值	不定值	FFFFH
捕捉&单次计数模式	递增计数	0000H	停止时的值	不定值	TDRmn寄存器的捕捉值+1

注1：通道n处于定时器运行停止状态（TEmn=0）和计数允许状态（TSmn=1）时的TCRmn寄存器的读取值。将此值保持在TCRmn寄存器，直到开始计数为止。

注2：单元号（m=0） n：通道号（n=0～3）

6.2.3 定时器数据寄存器mn（TDRmn）

这是能进行捕捉功能和比较功能切换使用的 16 位寄存器。通过定时器模式寄存器 mn（TMRmn）的 MDmn3～MDmn0 位来选择运行模式，进行捕捉功能和比较功能的切换。

能随时改写 TDRmn 寄存器。

能以 16 位为单位读写此寄存器。

在 8 位定时器模式中（定时器模式寄存器 m1、m3（TMRm1、TMRm3）的 SPLIT 位为“1”），能以 8 位为单位读写 TDRm1 寄存器和 TDRm3 寄存器，其中 TDRm1H 和 TDRm3H 用作高 8 位，TDRm1L 和 TDRm3L 用作低 8 位。

在产生复位信号后，TDRmn 寄存器的值变为“0000H”。

表6-4：通道0定时器数据寄存器TDR00

位	符号	描述	复位值
15:0	TDR00	定时器通道0数据寄存器	0x0

表6-5：通道1定时器数据寄存器TDR01

位	符号	描述	复位值
15:8	TDR01H	定时器通道1数据寄存器bit15:8	0x0
7:0	TDR01L	定时器通道1数据寄存器bit7:0	0x0

表6-6：通道2定时器数据寄存器TDR02

位	符号	描述	复位值
15:0	TDR02	定时器通道2数据寄存器	0x0

表6-7：通道1定时器数据寄存器TDR03

位	符号	描述	复位值
15:8	TDR03H	定时器通道3数据寄存器bit15:8	0x0
7:0	TDR03L	定时器通道3数据寄存器bit7:0	0x0

(1) 定时器数据寄存器mn（TDRmn）用作比较寄存器的情况

从TDRmn寄存器的设定值开始递减计数，当计数值变为“0000H”时，产生中断信号（INTTMmn）。保持TDRmn寄存器的值，直到被改写为止。

注：即使输入捕捉触发信号，设定为比较功能的TDRmn寄存器也不进行捕捉运行。

(2) 定时器数据寄存器mn（TDRmn）用作捕捉寄存器的情况

通过输入捕捉触发，将定时器计数寄存器mn（TCRmn）的计数值捕捉到TDRmn寄存器。

能选择TImn引脚的有效边沿作为捕捉触发信号。通过定时器模式寄存器mn（TMRmn）来设定捕捉触发的选择。

注：m：单元号（m=0） n：通道号（n=0～3）

6.2.4 外围允许寄存器0（PER0）

PER0 寄存器是设定允许或者禁止给各外围硬件提供时钟的寄存器。通过停止给不使用的硬件提供时钟，以降低功耗和噪声。

要使用通用定时器单元 4 时，必须将 bit0（TM40EN）置“1”。通过 8 位存储器操作指令设定 PER0 寄存器。在产生复位信号后，PER0 寄存器的值变为“00H”。

表6-8：外围允许寄存器0（PER0）的表格

位	符号	描述	复位值
7	LSITIMEEN	12位间隔定时器的输入时钟的控制 0：停止提供输入时钟。 • 12位间隔定时器使用的SFR。 •12位间隔定时器处于复位状态。 1：提供输入时钟。 • 12位间隔定时器使用的SFR。	0
6:0	-	保留	0x0
0	TM40EN	通用定时器单元4的输入时钟的控制 0：停止提供输入时钟。 • 不能写通用定时器单元使用的SFR。 • 通用定时器单元4处于复位状态。 1：提供输入时钟。 • 能读写通用定时器单元4使用的SFR	0

注：要设定通用定时器单元时，必须先在TM4mEN位为“1”的状态下设定以下的寄存器。当TM4mEN位为“0”时，定时器阵列单元的控制寄存器的值为初始值，忽视写操作（定时器输入输出选择寄存器0（TIOS0）、噪声滤波器允许寄存器1（NFEN1）、端口模式控制寄存器PMCx、端口模式寄存器PMx和端口复用功能配置寄存器PxxCFG除外）。

•定时器状态寄存器mn（TSRmn）
•定时器通道允许状态寄存器m（TEm）
•定时器通道开始寄存器m（TSm）
•定时器通道停止寄存器m（TTm）
•定时器输出允许寄存器m（TOEm）
•定时器输出寄存器m（TOm）
•定时器输出电平寄存器m（TOLm）
•定时器输出模式寄存器m（TOMm）

6.2.5 定时器时钟选择寄存器m（TPSm）

TPSm 寄存器是 16 位寄存器，选择提供给各通道的 2 种或者 4 种公共运行时钟（CKm0、CKm1、CKm2、CKm3）。通过 TPSm 寄存器的 bit3～0 选择 CKm0，通过 TPSm 寄存器的 bit7～4 选择 CKm1。另外，只有通道 1 和通道 3 才能选择 CKm2 和 CKm3，通过 TPSm 寄存器的 bit9～8 选择 CKm2，通过 TPSm 寄存器的 bit13 和 bit12 选择 CKm3。

只有在以下情况下才能改写定时器运行中的 TPSm 寄存器。

能改写 PRSm00～PRSm03 位的情况（n=0～3）：

选择 CKm0 作为运行时钟（CKSmn1、CKSmn0=0、0）的通道全部处于停止状态（TEmn=0）。

能改写 PRSm10～PRSm13 位的情况（n=0～3）：

选择 CKm2 作为运行时钟（CKSmn1、CKSmn0=0、1）的通道全部处于停止状态（TEmn=0）。

能改写 PRSm20 位和 PRSm21 位的情况（n=1、3）：

选择 CKm1 作为运行时钟（CKSmn1、CKSmn0=1、0）的通道全部处于停止状态（TEmn=0）。

能改写 PRSm30 位和 PRSm31 位的情况（n=1、3）：

选择 CKm3 作为运行时钟（CKSmn1、CKSmn0=1、1）的通道全部处于停止状态（TEmn=0）。

通过 16 位存储器操作指令设定 TPSm 寄存器。在产生复位信号后，TPSm 寄存器的值变为“0000H”。

表 6-9：定时器时钟选择寄存器 m（TPSm）的表格

位	符号	描述	复位值
15:14	-	必须为0	0x0
13:12	CKm3	定时器运行时钟选择 CKm3: 00H：Fclk=Fclk/2^8 01H：Fclk=Fclk/2^{10} 02H：Fclk=Fclk/2^{12} 03H：Fclk=Fclk/2^{14}	0x0
11:10	-	必须为0	0x0
9:8	CKm2	定时器运行时钟选择 CKm2: 00H：Fclk=Fclk/2 01H：Fclk=Fclk/2^2 02H：Fclk=Fclk/2^4 03H：Fclk=Fclk/2^6	0x0
7:4	CKm1	定时器运行时钟选择 Fclk=Fclk/2^{CKm1}	0x0
3:0	CKm0	定时器运行时钟选择 Fclk=Fclk/2^{CKm0}	0x0

注1：在更改选择为F_{CL}K的时钟（更改系统时钟控制寄存器（CKC）的值）的情况下，必须停止通用定时器单元（TTm=000FH）。即使在选择运行时钟（F_{MCK}）或者TImn引脚输入信号的有效边沿时，也需要停止通用定时器单元。

注2：F_{CLK}：CPU/外围硬件的时钟频率

注3：TPSm寄存器选择的时钟波形从上升沿开始只有1个F_{CLK}周期为高电平。详细内容请参照“6.4.1 计数时钟（F_{TCLK}）”。

注4：必须将bit15、14、11、10置“0”。

注5：如果选择F_{CLK}（无分频）作为运行时钟（CKmk）并且将TDRmn置“0000H”（m=0、1，n=0～3），就不能使用通用定时器单元的中断请求。

如果在 8 位定时器模式中使用通道 1 和通道 3 并且将 CKm2 和 CKm3 作为运行时钟，就能通过间隔定时器功能实现下表所示的间隔时间。

表6-10：运行时钟CKSm2和CKSm3能设定的间隔时间

时钟		间隔时间注（F_{CLK}=32MHz）			
		10us	100us	1ms	10ms
CKm2	$F_{CLK}/2$	○	—	—	—
	$F_{CLK}/2^2$	○	—	—	—
	$F_{CLK}/2^4$	○	○	—	—
	$F_{CLK}/2^6$	○	○	—	—
CKm3	$F_{CLK}/2^8$	—	○	○	—
	$F_{CLK}/2^{10}$	—	○	○	—
	$F_{CLK}/2^{12}$	—	—	○	○
	$F_{CLK}/2^{14}$	—	—	○	○

注1：○包含5%以内的误差。

注2：F_{CLK}：CPU/外围硬件的时钟频率

注3：有关TPSm寄存器所选$F_{CLK}/2^r$波形的详细内容，请参照“6.4.1 计数时钟（F_{TCLK}）”。

6.2.6 定时器模式寄存器mn（TMRmn）

TMRmn 寄存器是设定通道 n 运行模式的寄存器，进行运行时钟（F_{MCK}）的选择、计数时钟的选择、主控/从属的选择、16 位/8 位定时器的选择（只限于通道 1 和通道 3）、开始触发和捕捉触发的设定、定时器输入有效边沿的选择以及运行模式（间隔、捕捉、事件计数器、单次计数、捕捉&单次计数）的设定。

禁止在运行中（TEmn=1）改写 TMRmn 寄存器。但是，能在一部分的功能运行中（TEmn=1）改写 bit7 和 bit6（CISmn1、CISmn0）（详细内容请参照“6.7 通用定时器单元的独立通道运行功能”和“6.8 通用定时器的多通道联动运行功能”）。

通过 16 位存储器操作指令设定 TMRmn 寄存器。在产生复位信号后，TMRmn 寄存器的值变为“0000H”。

注意：TMRmn寄存器的bit11因通道而不同。

TMRm：MASTERmn位（n=2）

TMRm1、TMRm3：SPLITmn位（n=1、3）

TMRm0：固定为“0”。

表 6-11：定时器通道 0 模式寄存器 TMR00

位	符号	描述	复位值
15:14	CKS001- CKS000	通道n运行时钟（F_{MCK}）的选择 CKS001- CKS000: 00H：定时器时钟选择寄存器m（TPSm）设定的运行时钟CKm0 01H：定时器时钟选择寄存器m（TPSm）设定的运行时钟CKm2 02H：定时器时钟选择寄存器m（TPSm）设定的运行时钟CKm1 03H：定时器时钟选择寄存器m（TPSm）设定的运行时钟CKm3 运行时钟（F_{MCK}）用于边沿检测电路。通过设定CCSmn位来产生采样时钟和计数时钟（F_{TCLK}）。只有通道1和通道3才能选择运行时钟CKm2和CKm3。	0x0
13	0	必须为0	0
12	CCS00	通道n计数时钟（F_{TCLK}）的选择 00H：CKS000位和CKS001位指定的运行时钟（F_{MCK}） 01H：通道0：TIOS0选择的输入信号的有效边沿 通道1：TIOS0选择的输入信号的有效边沿 计数时钟（F_{TCLK}）用于计数器、输出控制电路和中断控制电路	0
11	0	必须为0	0
10:8	STS002- STS000	通道n的开始触发和捕捉触发的设定 STS002-STS000: 00H：只有软件触发开始有效（不选择其他触发源） 01H：将TI00引脚输入的有效边沿用于开始触发和捕捉触发。 02H：将TI00引脚输入的双边沿分别用于开始触	0x0

		发和捕捉触发。 04H：使用主控通道的中断信号（多通道联动运行功能的从属通道的情况） 上述以外, 禁止设定。	
7:6	CIS001- CIS000	TI00引脚的有效边沿选择 CIS001-CIS000: 00H：下降沿 01H：上升沿 02H：双边沿（测量低电平宽度时） 开始触发：下降沿，捕捉触发：上升沿 03H：双边沿（测量高电平宽度时） 开始触发：上升沿，捕捉触发：下降沿	0x0
5:4	0	保留	0x0
3:0	MD003- MD000	通道n运行模式及中断的设定 MD003-MD000： 00H：间隔定时器模式，在开始计数时不产生定时器中断。 01H：间隔定时器模式，在开始计数时产生定时器中断。 04H：捕捉模式，在开始计数时不产生定时器中断。 05H：捕捉模式，在开始计数时不产生定时器中断。 06H：事件计数器模式，在开始计数时不产生定时器中断。 08H：单次计数模式，计数运行中的开始触发无效。此时不产生中断。 09H：单次计数模式，计数运行中的开始触发有效。此时不产生中断。 0CH：捕捉&单次计数模式，在开始计数时不产生定时器中断。	

MD003- MD000 详细说明见下表

MD 003	MD 002	MD 001	通道n运行模式的设定	对应功能	TCR的计数运行
0	0	0	间隔定时器模式	间隔定时器/方波输出/分频器功能/PWM输出（主控）	递减计数
0	1	0	捕捉模式	输入脉冲间隔的测量	递增计数
0	1	1	事件计数器模式	外部事件计数器	递减计数
1	0	0	单次计数模式	延迟计数器/单触发脉冲输出/PWM输出（从属）	递减计数
1	1	0	捕捉&单次计数模式	输入信号的高低电平宽度的测量	递增计数
上述以外			禁止设定。		
各模式的运行因MD000位而变（参照下表）。					

运行模式（MD003～MD001位的设定（参照上表））	MD 000	开始计数和中断的设定
• 间隔定时器模式（0、0、0） • 捕捉模式（0、1、0）	0	在开始计数时不产生定时器中断（定时器的输出也不发生变化）。
	1	在开始计数时产生定时器中断（定时器的输出也发生变化）。
• 事件计数器模式（0、1、1）	0	在开始计数时不产生定时器中断（定时器的输出也不发生变化）。
• 单次计数模式[注1]（1、0、0）	0	计数运行中的开始触发无效。此时不产生中断。
	1	计数运行中的开始触发有效[注2]。此时不产生中断。
• 捕捉&单次计数模式（1、1、0）	0	在开始计数时不产生定时器中断（定时器的输出也不发生变化）。 计数运行中的开始触发无效。 此时不产生中断。

注1：在单次计数模式中，不控制开始计数时的中断输出（INTTM00）和TO00输出。

注2：如果在运行中产生开始触发（TS00=1），就对计数器进行初始化并且重新开始计数（不产生中断请求）。

表 6-12：定时器通道 0 模式寄存器 TMR01

位	符号	描述	复位值
31:16	-	保留	-
15:14	CKS011- CKS010	通道n运行时钟（F_{MCK}）的选择 CKS011- CKS010: 00H: 定时器时钟选择寄存器m（TPSm）设定的运行时钟CKm0 01H: 定时器时钟选择寄存器m（TPSm）设定的运行时钟CKm2 02H: 定时器时钟选择寄存器m（TPSm）设定的运行时钟CKm1 03H: 定时器时钟选择寄存器m（TPSm）设定的运行时钟CKm3 运行时钟（F_{MCK}）用于边沿检测电路。通过设定CCSmn位来产生采样时钟和计数时钟（F_{TCLK}）。只有通道1和通道3才能选择运行时钟CKm2和CKm3。	0x0
13	0	必须为0	0
12	CCS01	通道n计数时钟（F_{TCLK}）的选择 00H: CKS010位和CKS011位指定的运行时钟（F_{MCK}） 01H: TI01引脚输入信号的有效边沿 通道0：TIOS0选择的输入信号的有效边沿 通道1：TIOS0选择的输入信号的有效边沿 计数时钟（F_{TCLK}）用于计数器、输出控制电路和中断控制电路	0
11	SPLIT0	通道1的8位定时器/16位定时器的运行选择 00H用作16位定时器。 （用作独立通道运行功能或者多通道联动运行功能的从属通道） 01H:用作8位定时器。	0
10:8	STS012- STS010	通道n的开始触发和捕捉触发的设定 STS012- STS010: 00H: 只有软件触发开始有效（不选择其他触发源） 01H: 将TI01引脚输入的有效边沿用于开始触发和捕捉触发。 02H: 将TI01引脚输入的双边沿分别用于开始触发和捕捉触发。 04H: 使用主控通道的中断信号（多通道联动运行功能的从属通道的情况） 上述以外, 禁止设定。	0x0
7:6	CIS011- CIS010	TI01引脚的有效边沿选择 CIS011- CIS010 00H: 下降沿 01H: 上升沿 02H: 双边沿（测量低电平宽度时） 开始触发：下降沿，捕捉触发：上升沿 03H: 双边沿（测量高电平宽度时） 开始触发：上升沿，捕捉触发：下降沿	0x0
5:4	0	保留	0x0

3:0	MD013- MD010	通道n运行模式及中断的设定 MD013- MD010： 通道n运行模式及中断的设定 MD013- MD010： 00H: 间隔定时器模式，在开始计数时不产生定时器中断。 01H: 间隔定时器模式，在开始计数时产生定时器中断。 04H: 捕捉模式，在开始计数时不产生定时器中断。 05H: 捕捉模式，在开始计数时不产生定时器中断。 06H: 事件计数器模式，在开始计数时不产生定时器中断。 08H: 单次计数模式，计数运行中的开始触发无效。此时不产生中断。 09H: 单次计数模式，计数运行中的开始触发有效。此时不产生中断。 0CH: 捕捉&单次计数模式，在开始计数时不产生定时器中断。	

MD013- MD010 详细说明见下表

MD 013	MD 012	MD 011	通道n运行模式的设定	对应功能	TCR的计数运行
0	0	0	间隔定时器模式	间隔定时器/方波输出/ 分频器功能/PWM输出（主控）	递减计数
0	1	0	捕捉模式	输入脉冲间隔的测量	递增计数
0	1	1	事件计数器模式	外部事件计数器	递减计数
1	0	0	单次计数模式	延迟计数器/单触发脉冲输出/PWM输出（从属）	递减计数
1	1	0	捕捉&单次计数模式	输入信号的高低电平宽度的测量	递增计数
上述以外			禁止设定。		
各模式的运行因MD010位而变（参照下表）。					

运行模式（MD013～MD011位的设定（参照上表））	MD 010	开始计数和中断的设定
• 间隔定时器模式（0、0、0） • 捕捉模式（0、1、0）	0	在开始计数时不产生定时器中断（定时器的输出也不发生变化）。
	1	在开始计数时产生定时器中断（定时器的输出也发生变化）。
• 事件计数器模式（0、1、1）	0	在开始计数时不产生定时器中断（定时器的输出也不发生变化）。
• 单次计数模式[注1]（1、0、0）	0	计数运行中的开始触发无效。此时不产生中断。
	1	计数运行中的开始触发有效[注2]。此时不产生中断。
• 捕捉&单次计数模式（1、1、0）	0	在开始计数时不产生定时器中断（定时器的输出也不发生变化）。 计数运行中的开始触发无效。 此时不产生中断。

注1：单次计数模式中，不控制开始计数时的中断输出（INTTM01）和TO01输出。

注2：如果在运行中产生开始触发（TS01=1），就对计数器进行初始化并且重新开始计数（不产生中断请求）。

表 6-13：定时器通道 1 模式寄存器 TMR02

位	符号	描述	复位值
31:16	-	保留	-
15:14	CKS021- CKS020	通道n运行时钟（F_{MCK}）的选择 CKS021- CKS020: 00H: 定时器时钟选择寄存器m（TPSm）设定的运行时钟CKm0 01H: 定时器时钟选择寄存器m（TPSm）设定的运行时钟CKm2 02H: 定时器时钟选择寄存器m（TPSm）设定的运行时钟CKm1 03H: 定时器时钟选择寄存器m（TPSm）设定的运行时钟CKm3 运行时钟（F_{MCK}）用于边沿检测电路。通过设定CCSmn位来产生采样时钟和计数时钟（F_{TCLK}）。只有通道1和通道3才能选择运行时钟CKm2和CKm3。	0x0
13	0	必须为0	0
12	CCS02	通道n计数时钟（F_{TCLK}）的选择 00H：CKS020位和CKS021位指定的运行时钟（F_{MCK}） 01H：TI02引脚输入信号的有效边沿 道0：TIOS0选择的输入信号的有效边沿 通道1：TIOS0选择的输入信号的有效边沿 计数时钟（F_{TCLK}）用于计数器、输出控制电路和中断控制电路	0
11	MASTER	通道2的独立通道运行/多通道联动运行（从属或者主控）的选择 00H：用作独立通道运行功能或者多通道联动运行功能的从属通道。 01H：用作多通道联动运行功能的主控通道。 通道0固定为“0”（因为通道0为最高位的通道，所以与此位的设定无关，用作主控通道） 只能将通道2设定为主控通道（MASTERmn=1）	0
10:8	STS022- STS020	通道n的开始触发和捕捉触发的设定 STS022- STS020: 00H：只有软件触发开始有效（不选择其他触发源） 01H：将TI02引脚输入的有效边沿用于开始触发和捕捉触发。 02H：将TI02引脚输入的双边沿分别用于开始触发和捕捉触发。 04H：使用主控通道的中断信号（多通道联动运行功能的从属通道的情况） 上述以外，禁止设定。	0x0
7:6	CIS021- CIS020	TI01引脚的有效边沿选择 CIS021- CIS020 00H：下降沿 01H：上升沿	0x0

		02H：双边沿（测量低电平宽度时） 开始触发：下降沿，捕捉触发：上升沿 03H：双边沿（测量高电平宽度时） 开始触发：上升沿，捕捉触发：下降沿	
5:4	0	保留	0x0
3:0	MD023- MD020	通道n运行模式及中断的设定 MD023-MD020： 00H：间隔定时器模式，在开始计数时不产生定时器中断。 01H：间隔定时器模式，在开始计数时产生定时器中断。 04H：捕捉模式，在开始计数时不产生定时器中断。 05H：捕捉模式，在开始计数时不产生定时器中断。 06H：事件计数器模式，在开始计数时不产生定时器中断。 08H：单次计数模式，计数运行中的开始触发无效。此时不产生中断。 09H：单次计数模式，计数运行中的开始触发有效。此时不产生中断。 0CH：捕捉&单次计数模式，在开始计数时不产生定时器中断	

MD023- MD020 详细说明见下表

MD 023	MD 022	MD 021	通道n运行模式的设定	对应功能	TCR的计数运行
0	0	0	间隔定时器模式	间隔定时器/方波输出/ 分频器功能/PWM输出（主控）	递减计数
0	1	0	捕捉模式	输入脉冲间隔的测量	递增计数
0	1	1	事件计数器模式	外部事件计数器	递减计数
1	0	0	单次计数模式	延迟计数器/单触发脉冲输出/PWM输出（从属）	递减计数
1	1	0	捕捉&单次计数模式	输入信号的高低电平宽度的测量	递增计数
上述以外			禁止设定。		
各模式的运行因MD020位而变（参照下表）。					

运行模式（MD023～MD021位的设定（参照上表））	MD020	开始计数和中断的设定
• 间隔定时器模式（0、0、0） • 捕捉模式（0、1、0）	0	在开始计数时不产生定时器中断（定时器的输出也不发生变化）。
	1	在开始计数时产生定时器中断（定时器的输出也发生变化）。
• 事件计数器模式（0、1、1）	0	在开始计数时不产生定时器中断（定时器的输出也不发生变化）。
• 单次计数模式[注1]（1、0、0）	0	计数运行中的开始触发无效。此时不产生中断。
	1	计数运行中的开始触发有效[注2]。此时不产生中断。
• 捕捉&单次计数模式（1、1、0）	0	在开始计数时不产生定时器中断（定时器的输出也不发生变化）。 计数运行中的开始触发无效。 此时不产生中断。

注 1：在单次计数模式中，不控制开始计数时的中断输出（INTTM02）和 TO02 输出。

注 2：如果在运行中产生开始触发（TS02=1），就对计数器进行初始化并且重新开始计数（不产生中断请求）。

表 6-14：定时器通道 3 模式寄存器 TMR03

位	符号	描述	复位值
31:16	-	保留	-
15:14	CKS031- CKS030	通道n运行时钟（F_{MCK}）的选择 CKS031- CKS030: 00H：定时器时钟选择寄存器m（TPSm）设定的运行时钟CKm0 01H：定时器时钟选择寄存器m（TPSm）设定的运行时钟CKm2 02H：定时器时钟选择寄存器m（TPSm）设定的运行时钟CKm1 03H：定时器时钟选择寄存器m（TPSm）设定的运行时钟CKm3 运行时钟（F_{MCK}）用于边沿检测电路。通过设定CCSmn位来产生采样时钟和计数时钟（F_{TCLK}）。只有通道1和通道3才能选择运行时钟CKm2和CKm3。	0x0
13	0	必须为0	0
12	CCS03	通道n计数时钟（F_{TCLK}）的选择 00H：CKS030位和CKS031位指定的运行时钟（F_{MCK}） 01H：TI03引脚输入信号的有效边沿 通道0：TIOS0选择的输入信号的有效边沿 通道1：TIOS0选择的输入信号的有效边沿 计数时钟（F_{TCLK}）用于计数器、输出控制电路和中断控制电路	0
11	SPLIT0	通道3的8位定时器/16位定时器的运行选择 00H：用作16位定时器。 （用作独立通道运行功能或者多通道联动运行功能的从属通道） 01H：用作8位定时器。	0
10:8	STS032- STS030	通道n的开始触发和捕捉触发的设定 STS032- STS030: 00H：只有软件触发开始有效（不选择其他触发源） 01H：将TI03引脚输入的有效边沿用于开始触发和捕捉触发。 02H：将TI03引脚输入的双边沿分别用于开始触发和捕捉触发。 04H：使用主控通道的中断信号（多通道联动运行功能的从属通道的情况） 上述以外，禁止设定。	0x0
7:6	CIS031- CIS030	TI01引脚的有效边沿选择 CIS031- CIS030 00H：下降沿 01H：上升沿 02H：双边沿（测量低电平宽度时） 开始触发：下降沿，捕捉触发：上升沿 03H：双边沿（测量高电平宽度时） 开始触发：上升沿，捕捉触发：下降沿	0x0
5:4	0	保留	0x0

3:0	MD033- MD030	通道n运行模式及中断的设定 MD033-MD030： 00H：间隔定时器模式，在开始计数时不产生定时器中断。 01H：间隔定时器模式，在开始计数时产生定时器中断。 04H：捕捉模式，在开始计数时不产生定时器中断。 05H：捕捉模式，在开始计数时不产生定时器中断。 06H：事件计数器模式，在开始计数时不产生定时器中断。 08H：单次计数模式，计数运行中的开始触发无效。此时不产生中断。 09H：单次计数模式，计数运行中的开始触发有效。此时不产生中断。 0CH：捕捉&单次计数模式，在开始计数时不产生定时器中断	

MD033- MD030 详细说明见下表

MD 033	MD 322	MD 031	通道n运行模式的设定	对应功能	TCR的计数运行
0	0	0	间隔定时器模式	间隔定时器/方波输出/ 分频器功能/PWM输出（主控）	递减计数
0	1	0	捕捉模式	输入脉冲间隔的测量	递增计数
0	1	1	事件计数器模式	外部事件计数器	递减计数
1	0	0	单次计数模式	延迟计数器/单触发脉冲输出/PWM输出（从属）	递减计数
1	1	0	捕捉&单次计数模式	输入信号的高低电平宽度的测量	递增计数
上述以外			禁止设定。		
各模式的运行因MD030位而变（参照下表）。					

运行模式（MD023～MD021位的设定（参照上表））	MD030000	开始计数和中断的设定
• 间隔定时器模式（0、0、0） • 捕捉模式（0、1、0）	0	在开始计数时不产生定时器中断（定时器的输出也不发生变化）。
	1	在开始计数时产生定时器中断（定时器的输出也发生变化）。
• 事件计数器模式（0、1、1）	0	在开始计数时不产生定时器中断（定时器的输出也不发生变化）。
• 单次计数模式[注1]（1、0、0）	0	计数运行中的开始触发无效。此时不产生中断。
	1	计数运行中的开始触发有效[注2]。此时不产生中断。
• 捕捉&单次计数模式（1、1、0）	0	在开始计数时不产生定时器中断（定时器的输出也不发生变化）。计数运行中的开始触发无效。 此时不产生中断。

注 1：在单次计数模式中，不控制开始计数时的中断输出（INTTM03）和 TO03 输出。

注 2：如果在运行中产生开始触发（TS03=1），就对计数器进行初始化并且重新开始计数（不产生中断请求）。

6.2.7 定时器状态寄存器mn（TSRmn）

TSRmn 寄存器是表示通道 n 计数器的上溢状态的寄存器。

TSRmn 寄存器只在捕捉模式（MDmn3～MDmn1=010B）和捕捉&单次计数模式（MDmn3～MDmn1=110B）中有效。有关各运行模式中的 OVF 位的变化和置位/清除条件，请参照表 6-16。

通过 16 位存储器操作指令读 TSRmn 寄存器。

能用 TSRmnL 并且通过 8 位存储器操作指令读 TSRmn 寄存器的低 8 位。在产生复位信号后，TSRmn 寄存器的值变为“0000H”。

表6-15：定时器状态寄存器mn（TSRmn）的表格

位	符号	描述	复位值
15:1	-	保留	-
0	OVF	通道n的计数器上溢状态、 0：没有发生上溢。 1：发生上溢。 如果OVF位为“1”，就在下一次计数不发生上溢并且捕捉到计数值时清除此标志（OVF=0）。	0

注：m：单元号（m=0）n：通道号（n=0～3）

表6-16：各运行模式中的OVF位的变化和置位/清除条件

定时器运行模式	OVF位	置位/清除条件
• 捕捉模式 • 捕捉&单次计数模式	清除	在捕捉时没有发生上溢的情况
	置位	在捕捉时发生上溢的情况
• 间隔定时器模式 • 事件计数器模式 • 单次计数模式	清除	（不能使用）
	置位	

注：即使计数器发生上溢，OVF位也不立即发生变化，而在此后的捕捉时发生变化。

6.2.8 定时器通道允许状态寄存器m（TEm）

TEm 寄存器是表示各通道定时器运行的允许或者停止状态的寄存器。

TEm 寄存器的各位对应定时器通道开始寄存器 m（TSm）和定时器通道停止寄存器 m（TTm）的各位。如果将 TSm 寄存器的各位置“1”，就将 TEm 寄存器的对应位置“1”。如果将 TTm 寄存器的各位置“1”，就将其对应位清“0”。

通过 16 位存储器操作指令读 TEm 寄存器。

能用 TEmL 并且通过 8 位存储器操作指令读 TEm 寄存器的低 8 位。在产生复位信号后，TEm 寄存器的值变为“0000H”。

表6-17：定时器通道允许状态寄存器m（TEm）的表格

位	符号	描述	复位值
15:12	-	保留	-
11	TEH03	通道3为8位定时器模式时的高8位定时器的运行允许或者停止状态的表示 0：运行停止状态 1：运行允许状态	-
10	-	保留	0
9	TEH01	通道1为8位定时器模式时的高8位定时器的运行允许或者停止状态的表示 0：运行停止状态 1：运行允许状态	0
8:4	-	保留	0x0
3:0	TE03-TE00	通道n的运行允许或者停止状态的表示: 0：运行停止状态 1：运行允许状态	0x0

6.2.9 定时器通道开始寄存器m（TSm）

TSm 寄存器是对定时器计数寄存器 mn（TCRmn）进行初始化并且设定各通道计数运行开始的触发寄存器。如果将各位置“1”，定时器通道允许状态寄存器 m（TEm）的对应位就被置“1”。因为 TSmn 位、TSHm1 位和 TSHm3 位是触发位，所以如果变为运行允许状态（TEmn、TEHm1、TEHm3=1），就立即清除 TSmn 位、TSHm1 位和 TSHm3 位。

通过 16 位存储器操作指令设定 TSm 寄存器。

能用 TSmL 并且通过 8 位存储器操作指令设定 TSm 寄存器的低 8 位。在产生复位信号后，TSm 寄存器的值变为“0000H”。

表6-18：定时器通道开始寄存器m（TSm）的表格

位	符号	描述	复位值
15:12	-	保留	-
11	TSHm3	通道3为8位定时器模式时的高8位定时器的运行允许（开始）触发 0：没有触发 1：将TEHm3位置“1”，进入计数允许状态。如果在计数允许状态下开始TCRm3寄存器的计数，就进入间隔定时器模式（参照“6.4.2 计数器的开始时序”的表6-26）	-
10	-	保留	0
9	TSHm1	通道1为8位定时器模式时的高8位定时器的运行允许（开始）触发 0：没有触发 1：将TEHm1位置“1”，进入计数允许状态。如果在计数允许状态下开始TCRm1寄存器的计数，就进入间隔定时器模式（参照“6.4.2 计数器的开始时序”的表6-26）。	0
8:4	-	保留	0x0
3:0	TSm3-TSm0	通道n的运行允许（开始）触发示: 0：没有触发 1：将TEmn位置“1”，进入计数允许状态。计数允许状态下的TCRmn寄存器的计数开始因各运行模式而不同（参照“6.4.2 计数器的开始时序”的表6-26）。在通道1和通道3为8位定时器模式时，TSm1和TSm3为低8位定时器的运行允许（开始）触发。	0x0

注1：必须将bit15～12、10、8～4置“0”。

注2：从不使用TImn引脚输入的功能切换到使用TImn引脚输入的功能时，从设定定时器模式寄存器mn（TMRmn）到将TSmn（TSHm1、TSHm3）位置“1”为止，需要以下期间的等待：

TImn引脚噪声滤波器有效时（TNFENmn=1）：4个运行时钟（F_{MCK}）

TImn引脚噪声滤波器无效时（TNFENmn=0）：2个运行时钟（F_{MCK}）

注3：TSm寄存器的读取值总是“0”。

注4：m：单元号（m=0）

6.2.10 定时器通道停止寄存器m（TTm）

TTm 寄存器是设定各通道计数停止的触发寄存器。

如果将各位置“1”，定时器通道允许状态寄存器 m（TEm）的对应位就被清“0”。因为 TTmn 位、TTHm1 位和 TTHm3 位是触发位，所以如果变为运行停止状态（TEmn、TEHm1、TEHm3=0），就立即清除 TTmn 位、TTHm1 位和 TTHm3 位。

通过 16 位存储器操作指令设定 TTm 寄存器。

能用 TTmL 并且通过 8 位存储器操作指令设定 TTm 寄存器的低 8 位。在产生复位信号后，TTm 寄存器的值变为“0000H”。

表6-19：定时器通道停止寄存器m（TTm）的表格

位	符号	描述	复位值
15:12	-	保留	-
11	TTHm3	通道3为8位定时器模式时的高8位定时器的运行停止触发0: 没有触发 1: 将TEHm3位清“0”，进入计数停止状态	-
10	-	保留	0
9	TTHm1	通道1为8位定时器模式时的高8位定时器的运行停止触发 0: 没有触发 1: 将TEHm1位清“0”，进入计数停止状态。	0
8:4	-	保留	0x0
3:0	TTm3-TTm0	通道n的运行停止触发: 0: 没有触发 1: 将TEmn位清“0”，进入计数停止状态。 在通道1和通道3为8位定时器模式时，TTm1和TTm3为低8位定时器的运行停止触发。	0x0

注1：必须将bit15～12、10、8～4置“0”。

注2：TTm寄存器的读取值总是“0”。

注3：m：单元号（m=0）

6.2.11 定时器输入输出选择寄存器（TIOS0）

TIOS0 寄存器用于对单元 0 的输入输出进行选择。选择单元 0 的通道 0 和通道 1 的定时器输入以及通道 2 的定时器输出。通过 8 位存储器操作指令设定 TIOS0 寄存器。在产生复位信号后，TIOS0 寄存器的值变为“00H”。

表6-20：定时器输入输出选择寄存器0（TIOS0）的表格

位	符号	描述	复位值
7:5	TIS07- TIS05	通道0使用的定时器输入的选择 0: 定时器输入引脚（TI00）的输入信号 其他 ：禁止设定	-
4	TIS04	通道0使用的定时器输入的选择 0：通过TIS07~TIS05选择的输入信号	0
3	TOS03	通道2的定时器输出的使能 0: 允许输出 1: 禁止输出（输出固定为0）。	0
2:0	TIS02- TIS00	通道1使用的定时器输入的选择: 00H: 定时器输入引脚（TI01）的输入信号 02H: 定时器输入引脚（TI01）的输入信号 03H: 定时器输入引脚（TI01）的输入信号 04H: 低速内部振荡器时钟（FIL） 上述以外,禁止设定	0x0

注1：选择的定时器输入的高低电平宽度需要大于等于1/F_{MCK}+10ns。因此，在选择F_{IL}作为F_{CLK}时（CKC寄存器的CSS=1），不能将TIS02位置“1”。

6.2.12 定时器输出允许寄存器m（TOEm）

TOEm 寄存器是设定允许或者禁止各通道定时器输出的寄存器。

对于允许定时器输出的通道 n，无法通过软件改写后述的定时器输出寄存器 m（TOm）的 TOmn 位的值，并且由计数运行的定时器输出功能反映的值从定时器的输出引脚（TOmn）输出。

通过 16 位存储器操作指令设定 TOEm 寄存器。

能用 TOEmL 并且通过 8 位存储器操作指令设定 TOEm 寄存器的低 8 位。在产生复位信号后，TOEm 寄存器的值变为“0000H”。

表6-21：定时器输出允许寄存器m（TOEm）的表格

位	符号	描述	复位值
15:4		必须为0	0x0
3:0	TOEmn	通道n的定时器输出的允许/禁止: 0：定时器输出。 定时器的运行不反映到TOmn位，固定输出。 能写TOmn位，并且从TOmn引脚输出TOmn位设定的电平。 1：允许定时器输出 定时器的运行反映到TOmn位，产生输出波形。 忽视TOmn位的写操作	0x0

注1：必须将bit15～4置“0”。

注2：单元号（m=0） n：通道号（n=0～3）

6.2.13 定时器输出寄存器m（TOm）

TOm 寄存器是各通道定时器输出的缓冲寄存器。

此寄存器各位的值从各通道定时器的输出引脚（TOmn）输出。

只有在禁止定时器输出（TOEmn=0）时才能通过软件改写此寄存器的 TOmn 位。当允许定时器输出时（TOEmn=1），忽视通过软件的改写操作，而只通过定时器的运行更改其值。

要将 TI00、TO00、TI01/TO01、TI02/TO02、TI03/TO03 引脚用作端口功能时，必须将相应的 TOmn 位置“0”。

通过 16 位存储器操作指令设定 TOm 寄存器。

能用 TOmL 并且通过 8 位存储器操作指令设定 TOm 寄存器的低 8 位。在产生复位信号后，TOm 寄存器的值变为“0000H”。

表6-22：定时器输出寄存器m（TOm）的表格

位	符号	描述	复位值
15:4		必须为0	0x0
3:0	TOmn	通道n的定时器输出: 0：定时器的输出值为“0”。 1：定时器的输出值为“1”	0x0

注：m：单元号（m=0） n：通道号（n=0～3）

6.2.14 定时器输出电平寄存器m（TOLm）

TOLm 寄存器是控制各通道定时器输出电平的寄存器。

当允许定时器输出（TOEmn=1）并且使用多通道联动运行功能（TOMmn=1）时，在定时器输出信号的置位和复位时序，反映此寄存器进行的各通道 n 的反相设定。在主控通道输出模式（TOMmn=0）中，此寄存器的设定无效。

通过 16 位存储器操作指令设定 TOLm 寄存器。

能用 TOLmL 并且通过 8 位存储器操作指令设定 TOLm 寄存器的低 8 位。在产生复位信号后，TOLm 寄存器的值变为“0000H”。

表6-23：定时器输出电平寄存器m（TOLm）的表格

位	符号	描述	复位值
15:4		必须为0	0x0
3:1	TOL03- TOL01	通道n的定时器输出电平的控制 0：正逻辑输出（高电平有效）。 1：反相输出（低电平有效）	0x0
0	0	保留为0	0

注1：在定时器运行中改写此寄存器的值，会在下一次定时器输出信号发生变化时反相定时器的输出逻辑，而不是在改写后立即反相。

注2：m：单元号（m=0） n：通道号（n=0～3）

6.2.15 定时器输出模式寄存器m（TOMm）

TOMm 寄存器是控制各通道定时器输出模式的寄存器。当用作独立通道运行功能时，将所用通道的对应位置“0”。

当用作多通道联动运行功能（PWM 输出、单触发脉冲输出和多重 PWM 输出）时，将主控通道的对应位置“0”并且将从属通道的对应位置“1”。

当允许定时器输出（TOEmn=1）时，在定时器输出信号的置位和复位时序，反映此寄存器进行的各通道 n 的设定。

通过 16 位存储器操作指令设定 TOMm 寄存器。

能用 TOMmL 并且通过 8 位存储器操作指令设定 TOMm 寄存器的低 8 位。在产生复位信号后，TOMm 寄存器的值变为“0000H”。

表6-24：定时器输出模式寄存器m（TOMm）的表格

位	符号	描述	复位值
15:4		必须为0	0x0
3:1	TOM03- TOM01	通道n的定时器输出模式的控制 0: 主控通道输出模式（通过定时器中断请求信号（INTTMmn）进行交替输出）。 1：从属通道输出模式（通过主控通道的定时器中断请求信号（INTTMmn）将输出置位，并且通过从属通道的定时器中断请求信号（INTTMmp）对输出进行复位）	0x0
0	0	保留为0	0

注：m：单元号（m=0） n：通道号 n=0～3（主控通道时：n=0、2）

p：从属通道号

n=0：p=1、2、3

n=2：p=3

（有关主控通道和从属通道关系的详细内容，请参照“6.3.1 多通道联动运行功能的基本规则”）

6.2.16 噪声滤波器允许寄存器1（NFEN1）

NFEN1寄存器设定噪声滤波器是否用于单元0各通道定时器输入引脚的输入信号。对于需要消除噪声的引脚，必须将对应位置“1”，使噪声滤波器有效。当噪声滤波器有效时，在通过对象通道的运行时钟（F_{MCK}）进行同步后检测2个时钟是否一致；当噪声滤波器无效时，只通过对象通道的运行时钟（F_{MCK}）进行同步[注]。

通过8位存储器操作指令设定NFEN1寄存器。在产生复位信号后，NFEN1寄存器的值变为“00H”。

注：详细内容请参照“6.4.1(2) 选择TImn引脚输入信号的有效边沿的情况（CCSmn=1）”、“6.4.2 计数器的开始时序”和“6.6 定时器输入（TImn）的控制”。

表6-25：噪声滤波器允许寄存器1（NFEN1）的表格

位	符号	描述	复位值
7:4	-	保留	0x0
3	TNFEN03	TI03引脚的输入信号噪声滤波器的使用与否 0: 噪声滤波器OFF 1: 噪声滤波器ON	0
2	TNFEN02	TI02引脚的输入信号噪声滤波器的使用与否 0: 噪声滤波器OFF 1: 噪声滤波器ON	0
1	TNFEN01	TI01引脚的输入信号噪声滤波器的使用与否 0: 噪声滤波器OFF 1: 噪声滤波器ON	0
0	TNFEN00	TI00引脚的输入信号噪声滤波器的使用与否 0: 噪声滤波器OFF 1: 噪声滤波器ON	0

注：0～3的定时器输入/输出引脚的配置参照“第3章 引脚功能”。

6.2.17 控制定时器输入/输出引脚端口功能的寄存器

在使用通用定时器单元时，定时器 0 的输出引脚复用到固定端口，定时器 0 的输入引脚可以配置到任意端口。详细内容请参照“第 3 章 引脚功能”

在将定时器 0 的输出引脚复用到端口时，必须将该端口对应得端口模式控制寄存器（PMCxx）的位、端口模式寄存器（PMxx）的位、端口寄存器（Pxx）的位置“0”。

在将定时器 0 的输出引脚复用到某端口时，必须将该端口对应的端口模式控制寄存器(PMCxx)的位，端口模式寄存器(PMxx)的位置“0”。并设置端口复用功能配置寄存器(PxxCFG)。此时,端口寄存器(Pxx)的位可以是“0”或者"1"。

(例)

将 P20 配置为 TO00 用作定时器输出的情况。

将端口模式控制寄存器 2 的 PMC20 位置“0”。

将端口模式寄存器 2 的 PM20 位置“0”。

将端口输出复用功能配置寄存器 P20CFG 置“0x02”。”

在将定时器 0 输入引脚的复用端口用作定时器的输入时,必须将各端口对应的端口模式奇存器(PMx)的位置1”并且将端口模式控制寄存器(PMCxx)的位置口。并设置端口复用功能配置寄存器(PStau0tin0_CFG)。此时，端口寄存器(Pxx)的位可以是 0 或者“1”。

(例)

将 P20/TI00用作定时器输入的情况。

将端口模式控制寄存器2的PMC20位置“0”。

将端口模式寄存器2的PM20位置“1”。

将端口输入复用功能配置寄存器 PStau0tin0 CFG 置*0x207。

6.3 通用定时器单元的基本规则

6.3.1 多通道联动运行功能的基本规则

多通道联动运行功能是将主控通道（主要对周期进行计数的基准定时器）和从属通道（遵从主控通道运行的定时器）组合实现的功能，使用时需要遵守几个规则。

多通道联动运行功能的基本规则如下所示。

(1)只能将偶数通道（通道 0、通道 2）设定为主控通道。

(2)通道 0 以外的任何通道都能设定为从属通道。

(3)只能将主控通道的低位通道设定为从属通道。

例在将通道 0 设定为主控通道时，能将通道 1 开始的通道（通道 1、通道 2、通道 3）设定为从属通道。

(4)能对 1 个主控通道设定多个从属通道。

(5)当使用多个主控通道时，不能设定跨越主控通道的从属通道。

例在将通道 0 和通道 2 设定为主控通道时，能将通道 1 设定为主控通道 0 的从属通道，而不能将通道 3 设定为主控通道 0 的从属通道。

(6)和主控通道联动的从属通道需要设定相同的运行时钟。和主控通道联动的从属通道的 CKSmn0 位和 CKSmn1 位（定时器模式寄存器 mn（TMRmn）的 bit15 和 bit14）的值需要是相同的设定值。

(7)主控通道能将 INTTMmn（中断）、开始软件触发和计数时钟传给低位通道。

(8)从属通道能将主控通道的 INTTMmn（中断）、开始软件触发和计数时钟用作源时钟，但是不能将自己的 INTTMmn（中断）、开始软件触发和计数时钟传给低位通道。

(9)主控通道不能将其他高位主控通道的 INTTMmn（中断）、开始软件触发和计数时钟用作源时钟。

(10)为了同时启动要联动的通道，需要同时设定联动通道的通道开始触发位（TSmn）。

(11)只有联动的全部通道或者主控通道才能使用计数运行中的 TSmn 位的设定。不能只使用从属通道的 TSmn 位的设定。

(12)为了同时停止要联动的通道，需要同时设定联动通道的通道停止触发位（TTmn）。

(13)在联动运行时，因为主控通道和从属通道需要相同的运行时钟，所以不能选择 CKm2/CKm3。

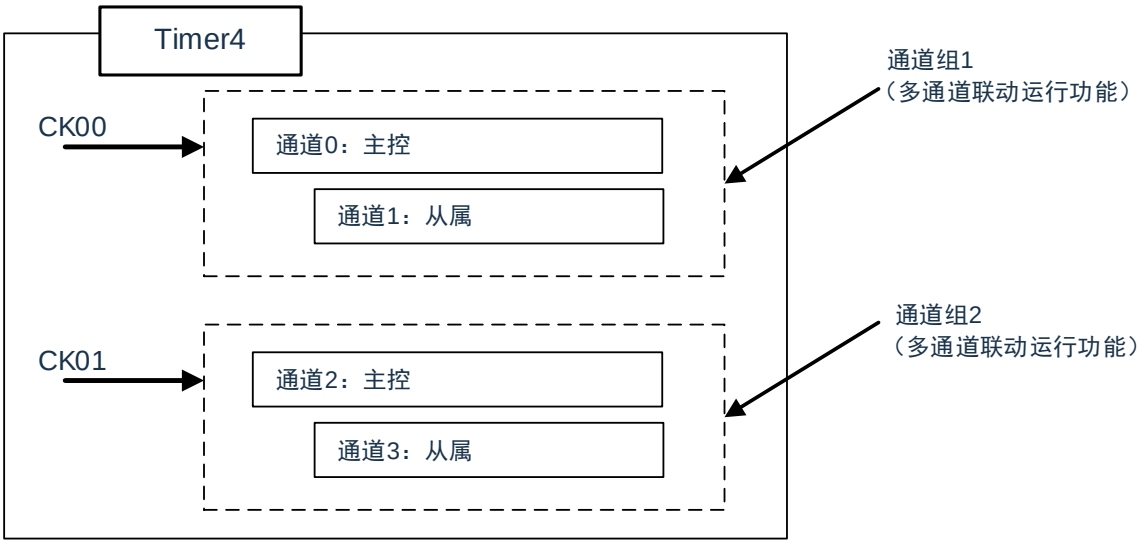
(14)定时器模式寄存器 m0（TMRm0）没有主控位而固定为“0”。但是，因为通道 0 是最高位的通道，所以在联动运行时能将通道 0 用作主控通道。

多通道联动运行功能的基本规则是适用于通道组群（形成 1 个多通道联动运行功能的主控通道和从属通道的集合）的规则。

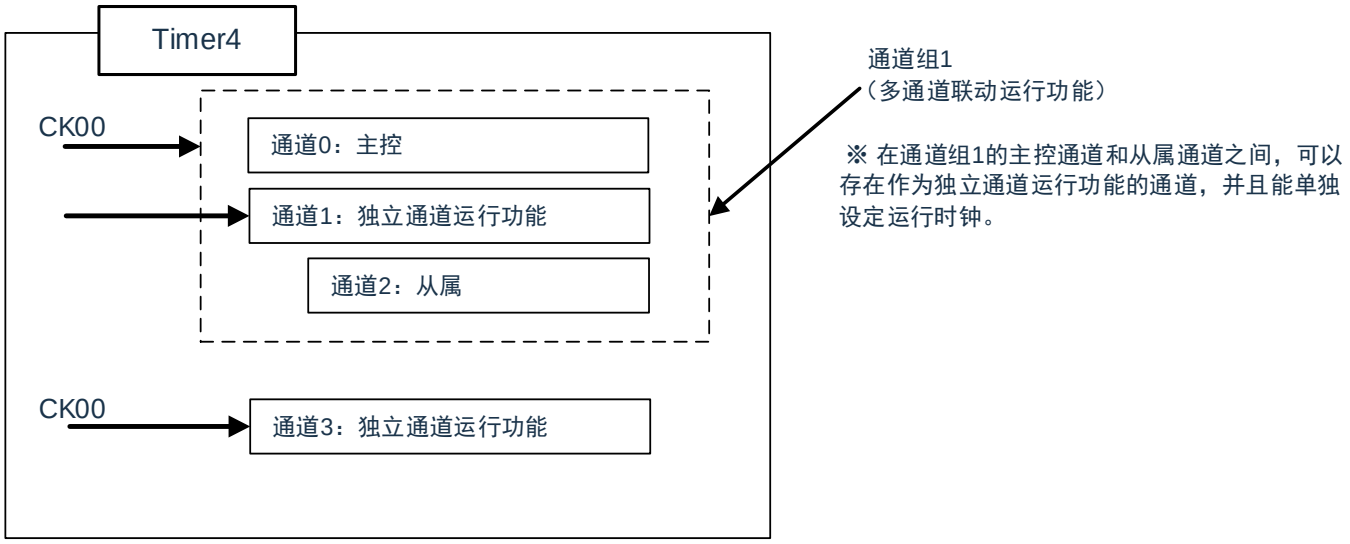
如果设定 2 个或者更多的相互不联动的通道群，通道群之间就不适用上述的基本规则。

注：m：单元号（m=0） n：通道号（n=0～3）

例 1


Timer4
CK00
通道0：主控
通道1：从属
通道组1
（多通道联动运行功能）
CK01
通道2：主控
通道3：从属
通道组2
（多通道联动运行功能）

例2


Timer4
CK00
通道0：主控
通道1：独立通道运行功能
通道2：从属
CK00
通道3：独立通道运行功能
通道组1
（多通道联动运行功能）
※ 在通道组1的主控通道和从属通道之间，可以存在作为独立通道运行功能的通道，并且能单独设定运行时钟。

6.3.2 8 位定时器运行功能的基本规则（只限于通道1和通道3）

8 位定时器运行功能是将 16 位定时器的通道用作 2 个 8 位定时器的通道的功能。

只有通道 1 和通道 3 才能使用 8 位定时器运行功能，使用时需要遵守几个规则。

8 位定时器运行功能的基本规则如下所示。

(1)8 位定时器运行功能只适用于通道 1 和通道 3。

(2)当用作 8 位定时器时，将定时器模式寄存器 mn（TMRmn）的 SPLIT 位置“1”。

(3)高 8 位定时器能用作间隔定时器功能。

(4)在开始运行时，高 8 位定时器输出 INTTMm1H（中断）（和 MDmn0 位为“1”的运行相同）。

(5)高 8 位定时器的运行时钟的选择取决于低位 TMRmn 寄存器的 CKSmn1 位和 CKSmn0 位的设定。

(6)对于高 8 位定时器，通过操作 TSHm1 位来开始通道的运行，并且通过操作 TTHm1 位来停止通道的运行。能通过 TEHm1 位确认通道的状态。

(7)低 8 位定时器的运行取决于 TMRmn 寄存器的设定，有以下 3 种支持低 8 位定时器运行的功能：

- 间隔定时器功能
- 外部事件计数器功能
- 延迟计数功能

(8)对于低 8 位定时器，通过操作 TSm1/TSm3 位来开始通道的运行，并且通过操作 TTm1/TTm3 位来停止通道的运行。能通过 TEm1/TEm3 位确认通道的状态。

(9)在 16 位定时器运行时，TSHm1/TSHm3/TTHm1/TTHm3 位的操作无效。通过操作 TSm1/TSm3 位和 TTm1/TTm3 位使通道 1 和通道 3 运行。TEHm3 位和 TEHm1 位不变。

(10)8 位定时器功能不能使用联动运行功能（单触发脉冲、PWM 和多重 PWM）。

注：m：单元号（m=0） n：通道号（n=1、3）

6.4 计数器的运行

6.4.1 计数时钟（F_{TCLK}）

通用定时器单元的计数时钟（F_{TCLK}）能通过定时器模式寄存器 mn（TMRmn）的 CCSmn 位选择以下任意一个时钟：

① CKSmn0 位和 CKSmn1 位指定的运行时钟（F_{MCK}）

② TImn 引脚输入信号的有效边沿

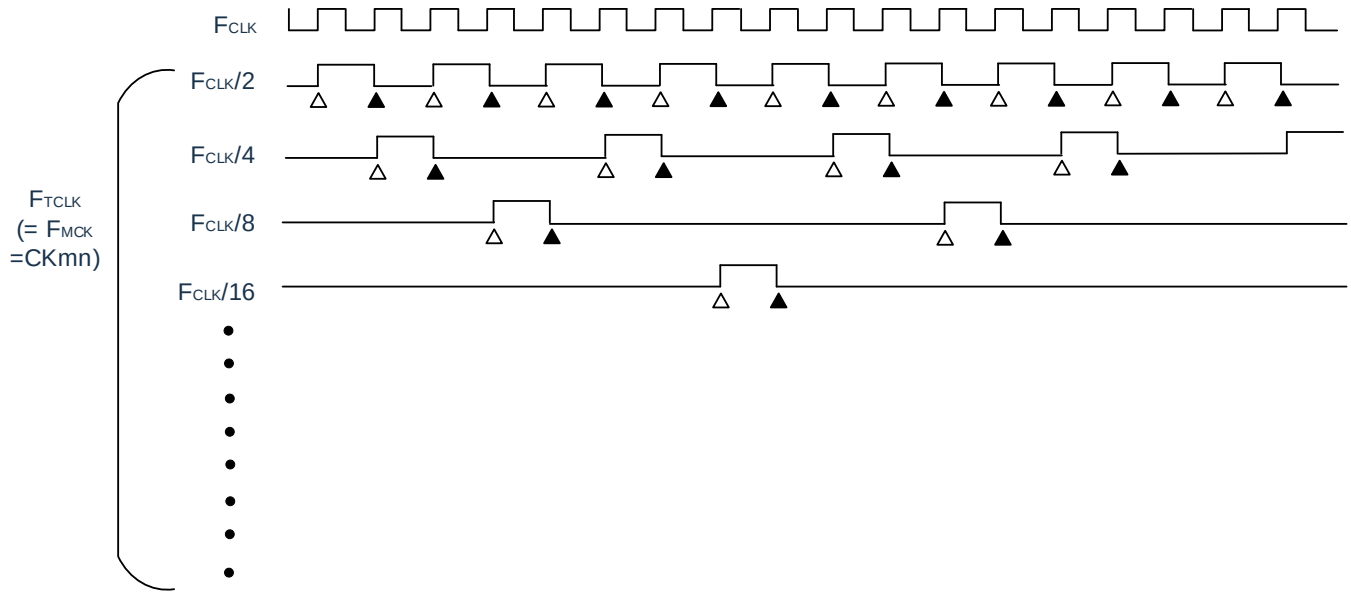
通用定时器单元被设计为与 F_{CLK} 同步运行，因此计数时钟（F_{TCLK}）的时序如下。

(1)选择 CKSmn0 位和 CKSmn1 位指定的运行时钟（F_{MCK}）的情况（CCSmn=0）

根据定时器时钟选择寄存器 m（TPSm）的设定，计数时钟（F_{TCLK}）为 F_{CLK}～$F_{CLK}/2^{15}$。但是，当选择 F_{CLK} 的分频时，TPSm 寄存器选择的时钟是从上升沿开始只有 1 个 F_{CLK} 周期为高电平的信号。当选择 F_{CLK} 时，固定为高电平。

为了取得与 F_{CLK} 的同步，定时器计数寄存器 mn（TCRmn）从计数时钟的上升沿开始延迟 1 个 F_{CLK} 时钟后进行计数，出于方便而将其称为“在计数时钟的上升沿进行计数”。

图6-2：F_{CLK}和计数时钟（F_{TCLK}）的时序（CCSmn=0的情况）



注1：△：计数时钟的上升沿

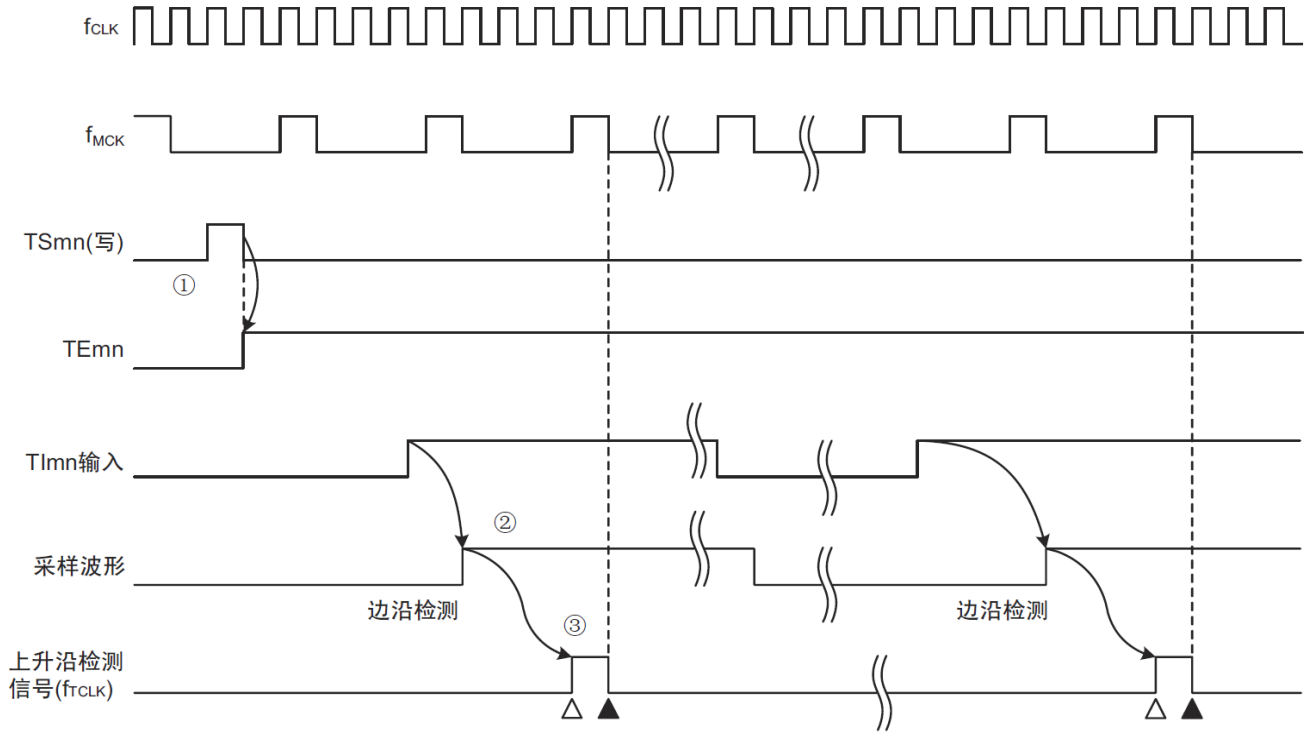
注2：▲：同步、计数器的递增/递减

注3：F_{CLK}：CPU/外围硬件的时钟

(2)选择 TImn 引脚输入信号的有效边沿的情况（CCSmn=1）

计数时钟（F_{TCLK}）是检测 TImn 引脚输入信号的有效边沿并且与下一个 F_{MCK} 上升沿同步的信号。实际上，这是比 TImn 引脚的输入信号延迟了 1～2 个 F_{MCK} 时钟的信号（在使用噪声滤波器时，延迟 3～4 个 F_{MCK} 时钟）。为了取得与 F_{CLK} 的同步，定时器计数寄存器 mn（TCRmn）从计数时钟的上升沿开始延迟 1 个 F_{CLK} 时后进行计数，出于方便而将其称为“在 TImn 引脚输入信号的有效边沿进行计数”。

图6-3：计数时钟（F_{TCLK}）的时序（CCSmn=1，未使用噪声滤波器的情况）



①通过将 TSmn 位置位来开始定时器的运行，并且等待 TImn 输入的有效边沿。

②通过 F_{MCK} 对 TImn 输入的上升沿进行采样。

③ 在采样信号的上升沿检测边沿，并且输出检测信号（计数时钟）。

注1：△：计数时钟的上升沿

注2：▲：同步、计数器的递增/递减

注3：F_{CLK}：CPU外围硬件时钟

注4：F_{MCK}：通道n的运行时钟

注5：输入脉冲间隔的测量、输入信号高低电平的测量、延迟计数器和单触发脉冲输出功能的TImn输入也是同样的波形。

6.4.2 计数器的开始时序

通过将定时器通道开始寄存器 m（TSm）的 TSmn 位置位，定时器计数寄存器 mn（TCRmn）进入运行允许状态。

从计数允许状态到定时器计数寄存器 mn（TCRmn）开始计数为止的运行如表 6-26 所示。

表6-26：从计数允许状态到定时器计数寄存器mn（TCRmn）开始计数为止的运行

定时器的运行模式	将TSmn位置“1”后的运行
•间隔定时器模式	从检测到开始触发（TSmn=1）到产生计数时钟为止，不执行任何操作。 通过第1个计数时钟将TDRmn寄存器的值装入TCRmn寄存器，并且通过后续的计数时钟进行递减计数（参照“6.4.3 间隔定时器模式的运行”）。
•事件计数器模式	通过给TSmn位写“1”，将TDRmn寄存器的值装入TCRmn寄存器。 如果检测到TImn的输入边沿，就通过后续的计数时钟进行递减计数。（参照“6.4.3 事件计数器模式的运行”）。
•捕捉模式	从检测到开始触发到产生计数时钟为止，不执行任何操作。 通过第1个计数时钟将“0000H”装入TCRmn寄存器，并且通过后续的计数时钟进行递增计数（参照“6.4.3 捕捉模式的运行（输入脉冲的间隔测量）”）。
•单次计数模式	通过在定时器停止运行（TEmn=0）的状态下给TSmn位写“1”，进入开始触发的等待状态。 从检测到开始触发到产生计数时钟为止，不执行任何操作。 通过第1个计数时钟将TDRmn寄存器的值装入TCRmn寄存器，并且通过后续的计数时钟进行递减计数（参照“6.4.3 单次计数模式的运行”）。
•捕捉&单次计数模式	通过在定时器停止运行（TEmn=0）的状态下给TSmn位写“1”，进入开始触发的等待状态。 从检测到开始触发到产生计数时钟为止，不执行任何操作。 通过第1个计数时钟将“0000H”装入TCRmn寄存器，并且通过后续的计数时钟进行递增计数（参照“6.4.3 捕捉&单次计数模式的运行（高电平宽度的测量）。

6.4.3 计数器的运行

以下说明各模式的计数器运行。

(1) 间隔定时器模式的运行

① 通过给 TSmn 位写“1”，进入运行允许状态（TEmn=1）。定时器计数寄存器 mn（TCRmn）保持初始值，直到产生计数时钟为止。

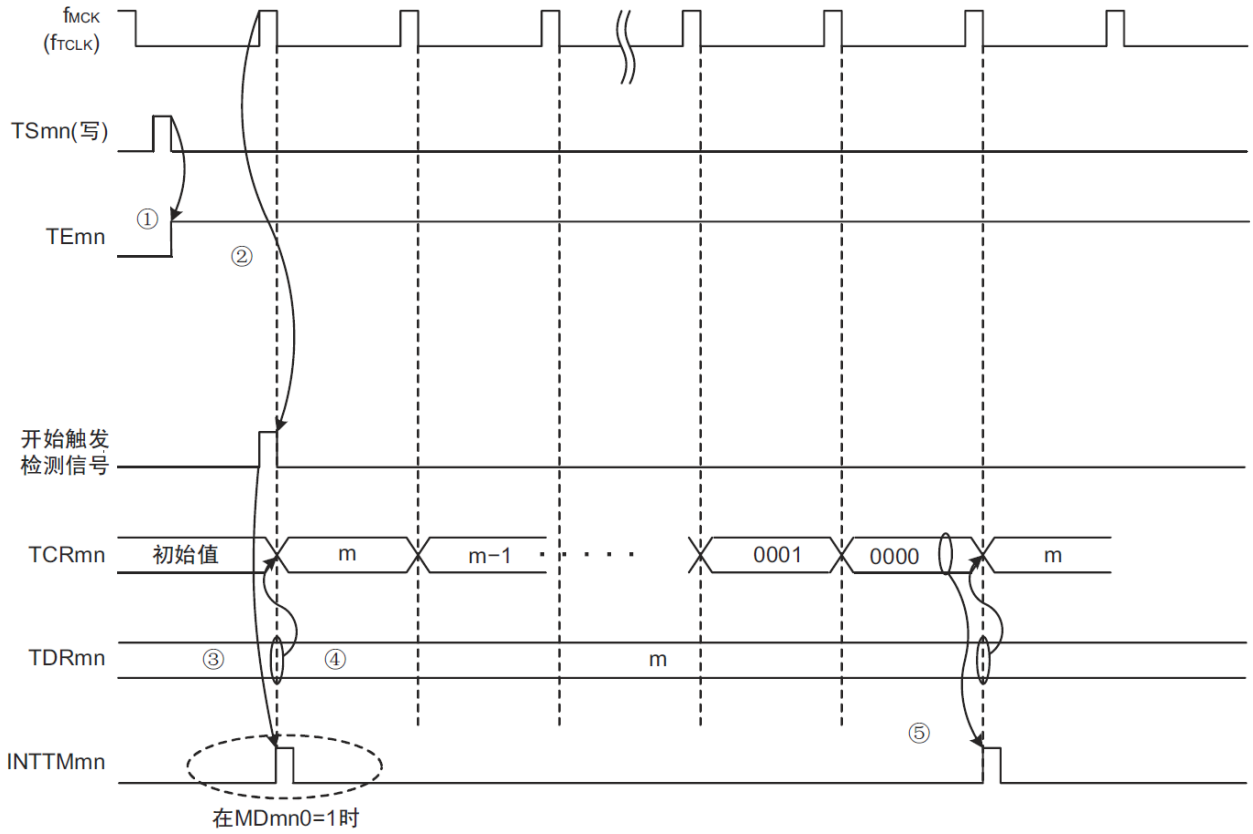
② 通过允许运行后的第 1 个计数时钟（F_{MCK}）产生开始触发信号。

③ 当 MDmn0 位为“1”时，通过开始触发信号产生 INTTMmn。

④ 通过允许运行后的第 1 个计数时钟将定时器数据寄存器 mn（TDRmn）的值装入 TCRmn 寄存器，并且以间隔定时器模式开始计数。

⑤ 如果 TCRmn 寄存器递减计数到“0000H”，就通过下一个计数时钟（F_{MCK}）产生 INTTMmn 并且在将定时器数据寄存器 mn（TDRmn）的值装入 TCRmn 寄存器后继续计数。

图6-4：运行时序（间隔定时器模式）



注1：因为第1个计数时钟周期的运行在写TSmn位后并且在产生计数时钟前延迟计数的开始，所以产生最大为1个时钟周期的误差。另外，如果需要开始计数时序的信息，就将MDmn0位置“1”，以便能在开始计数时产生中断。

注2：F_{MCK}、开始触发检测信号和INTTMmn与F_{CLK}同步并且在1个时钟内有效。

(2) 事件计数器模式的运行

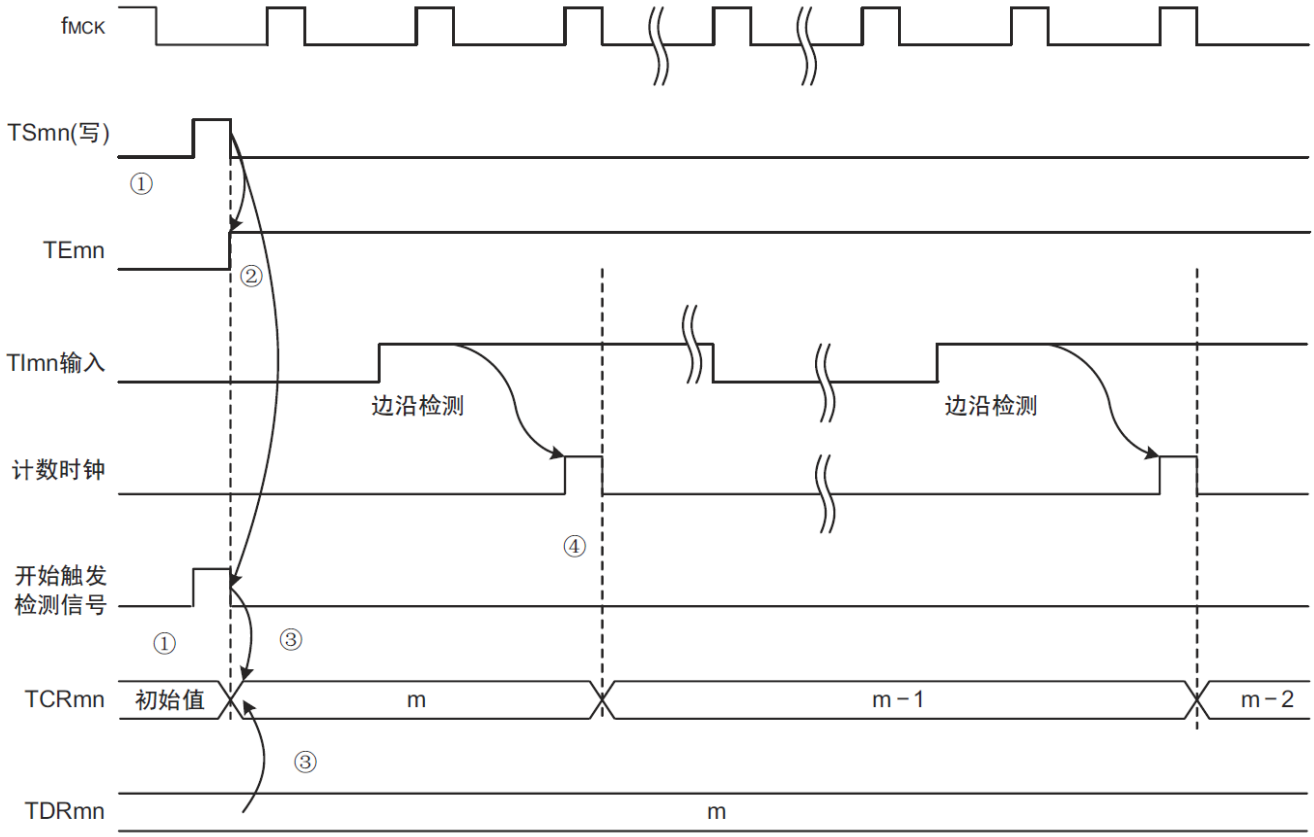
① 在运行停止状态（TEmn=0）的期间，定时器计数寄存器 mn（TCRmn）保持初始值。

② 通过给 TSmn 位写“1”，进入运行允许状态（TEmn=1）。

③ 在 TSmn 位和 TEmn 位都变为“1”的同时将定时器数据寄存器 mn（TDRmn）的值装入 TCRmn 寄存器，并且开始计数。

此后，在 TImn 输入的有效边沿，通过计数时钟对 TCRmn 寄存器的值进行递减计数。

图6-5：运行时序（事件计数器模式）



注：这是不使用噪声滤波器时的时序。如果使用噪声滤波器，边沿检测就从TImn输入开始再延迟2个F_{MCK}周期（合计3～4个周期）。1个周期的误差是因为TImn输入与计数时钟（F_{MCK}）不同步。

(3) 捕捉模式的运行（输入脉冲的间隔测量）

① 通过给 TSmn 位写“1”，进入运行允许状态（TEmn=1）。

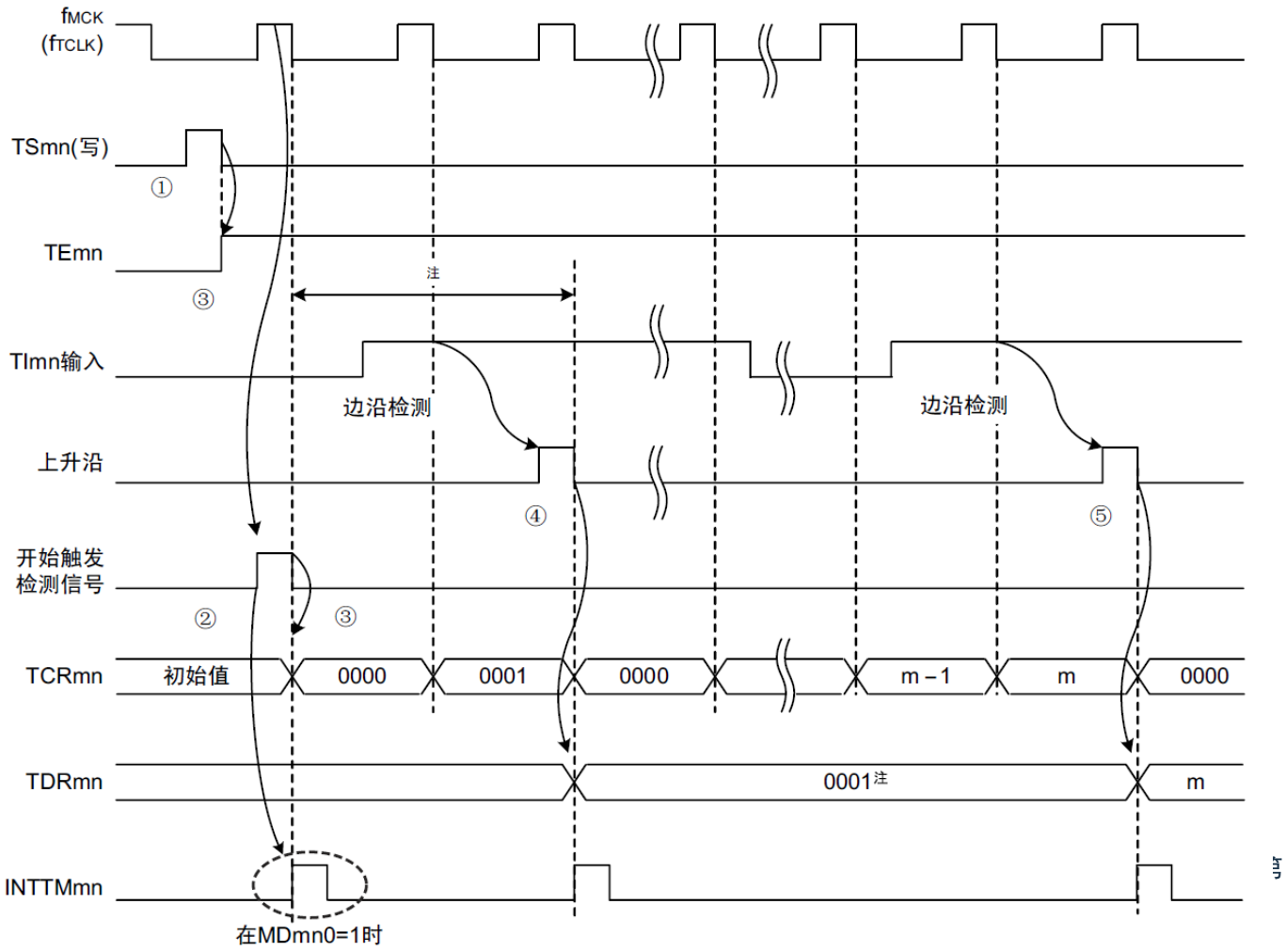
② 定时器计数寄存器 mn（TCRmn）保持初始值，直到产生计数时钟为止。

③ 通过允许运行后的第 1 个计数时钟（F_{MCK}）产生开始触发信号。然后，将“0000H”装入 TCRmn 寄存器并且以捕捉模式开始计数（当 MDmn0 位为“1”时，通过开始触发信号产生 INTTMmn）。

④ 如果检测到 TImn 输入的有效边沿，就将 TCRmn 寄存器的值捕捉到 TDRmn 寄存器，并且产生 INTTMmn 中断。此时的捕捉值没有意义。TCRmn 寄存器从“0000H”开始继续计数。

⑤ 如果检测到下一个 TImn 输入的有效边沿，就将 TCRmn 寄存器的值捕捉到 TDRmn 寄存器，并且产生 INTTMmn 中断。

图6-6：运行时序（捕捉模式：输入脉冲的间隔测量）



注1：第1个计数时钟周期的运行在写TSmn位后并且在产生计数时钟前延迟计数的开始，所以产生最大为1个时钟周期的误差。另外，如果需要开始计数时序的信息，就将MDmn0位置“1”，以便能在开始计数时产生中断。

注2：不使用噪声滤波器时的时序。如果使用噪声滤波器，边沿检测就从TImn输入开始再延迟2个F_{MCK}周期（合计3～4个周期）。1个周期的误差是因为TImn输入与计数时钟（F_{MCK}）不同步。

(4) 单次计数模式的运行

① 通过给 TSmn 位写“1”，进入运行允许状态（TEmn=1）。

② 定时器计数寄存器 mn（TCRmn）保持初始值，直到产生开始触发信号为止。

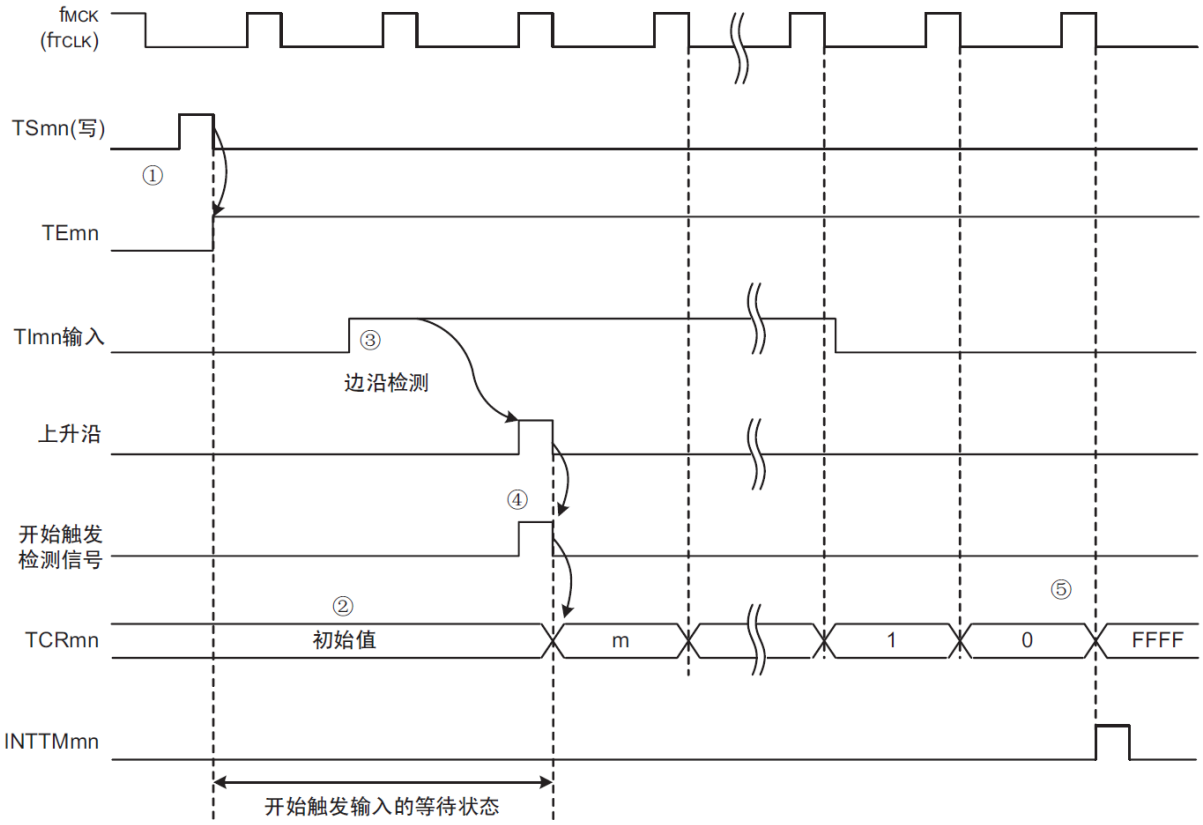
③ 检测 TImn 输入的上升沿。

④ 在产生开始触发信号后将 TDRmn 寄存器的值（m）装入 TCRmn 寄存器，并且开始计数。

⑤ 当 TCRmn 寄存器递减计数到“0000H”时，产生 INTTMmn 中断，并且 TCRmn 寄存器的值变为“FFFFH”，停止计数。

.

图6-7：运行时序（单次计数模式）



注1：不使用噪声滤波器时的时序。如果使用噪声滤波器，边沿检测就从TImn输入开始再延迟2个F_{MCK}周期（合计3～4个周期）。1个周期的误差是因为TImn输入与计数时钟（F_{MCK}）不同步。

(5) 捕捉&单次计数模式的运行（高电平宽度的测量）

① 通过给定时器通道开始寄存器 m（TSm）的 TSmn 位写“1”，进入运行允许状态（TEmn=1）。

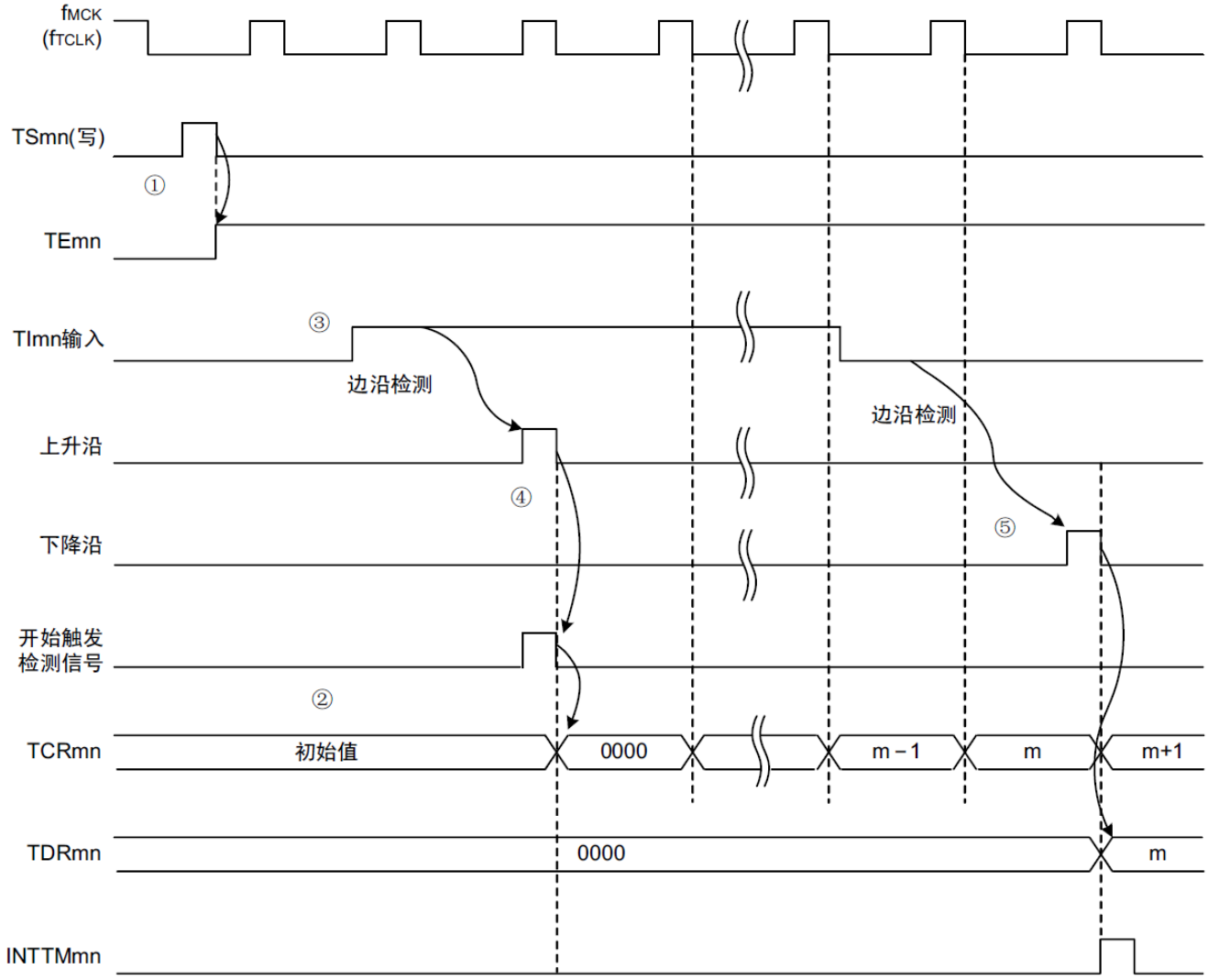
② 定时器计数寄存器 mn（TCRmn）保持初始值，直到产生开始触发信号为止。

③ 检测 TImn 输入的上升沿。

④ 在产生开始触发信号后将“0000H”装入 TCRmn 寄存器，并且开始计数。

⑤ 如果检测到 TImn 输入的下降沿，就将 TCRmn 寄存器的值捕捉到 TDRmn 寄存器，并且产生 INTTMmn 中断。

图6-8：运行时序（捕捉&单次计数模式：高电平宽度的测量）

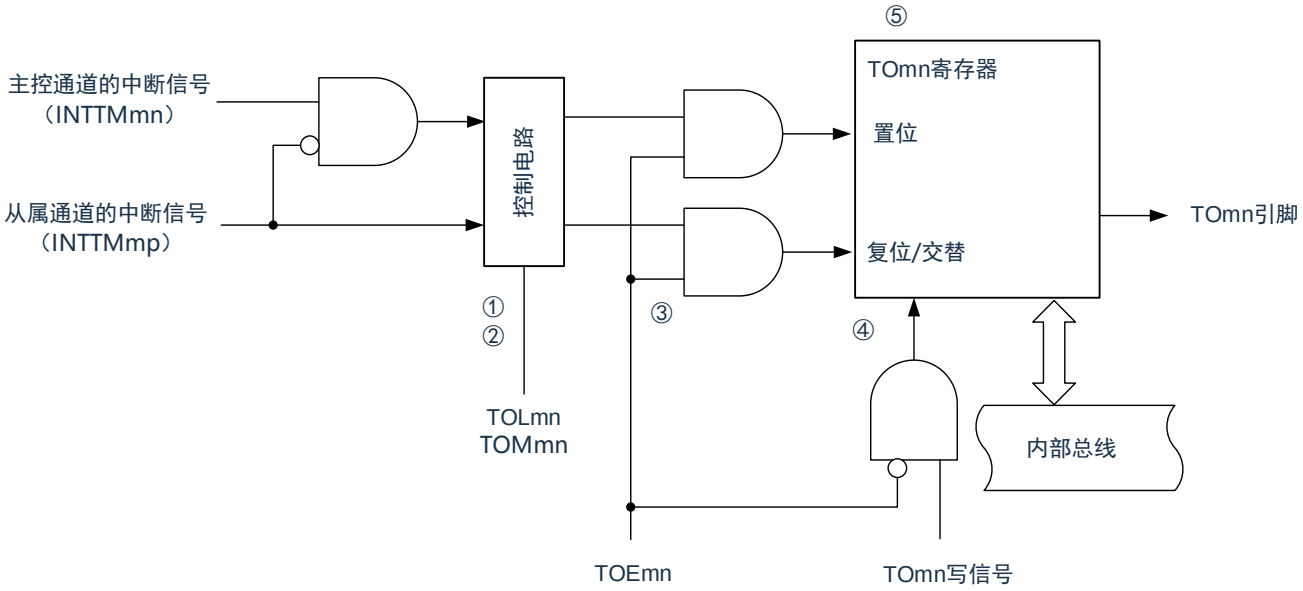


注：不使用噪声滤波器时的时序。如果使用噪声滤波器，边沿检测就从TImn输入开始再延迟2个F_{MCK}周期（合计3～4个周期）。1个周期的误差是因为TImn输入和计数时钟（F_{MCK}）不同步。

6.5 通道输出（TOmn引脚）的控制

6.5.1 TOmn引脚输出电路的结构

图6-9：输出电路的结构



① 当 TOMmn 位为“0”（主控通道输出模式）时，忽视定时器输出电平寄存器 m（TOLm）的设定值，只将 INTTMmp（从属通道定时器中断）传给定时器输出寄存器 m（TOm）。

② 当 TOMmn 位为“1”（从属通道输出模式）时，将 INTTMmn（主控通道定时器中断）和 INTTMmp（从属通道定时器中断）传给 TOm 寄存器。

此时，TOLm 寄存器有效并且进行以下信号的控制：

TOLmn=0 时：正相运行（INTTMmn→置位、INTTMmp→复位）TOLmn=1 时：反相运行（INTTMmn→复位、INTTMmp→置位）

当同时产生 INTTMmn 和 INTTMmp 时（PWM 输出的 0%输出），优先 INTTMmp（复位信号）而屏蔽 INTTMmn（置位信号）。

③ 在允许定时器输出（TOEmn=1）的状态下，将 INTTMmn（主控通道定时器中断）和 INTTMmp（从属通道定时器中断）传给 TOm 寄存器。TOm 寄存器的写操作（TOmn 写信号）无效。

当 TOEmn 位为“1”时，除了中断信号以外，不改变 TOmn 引脚的输出。

要对 TOmn 引脚的输出电平进行初始化时，需要在设定为禁止定时器输出（TOEmn=0）后给 TOm 寄存器写值。

④ 在禁止定时器输出（TOEmn=0）的状态下，对象通道的 TOmn 位的写操作（TOmn 写信号）有效。当定时器输出为禁止状态（TOEmn=0）时，不将 INTTMmn（主控通道定时器中断）和 INTTMmp（从属通道定时器中断）传给 TOm 寄存器。

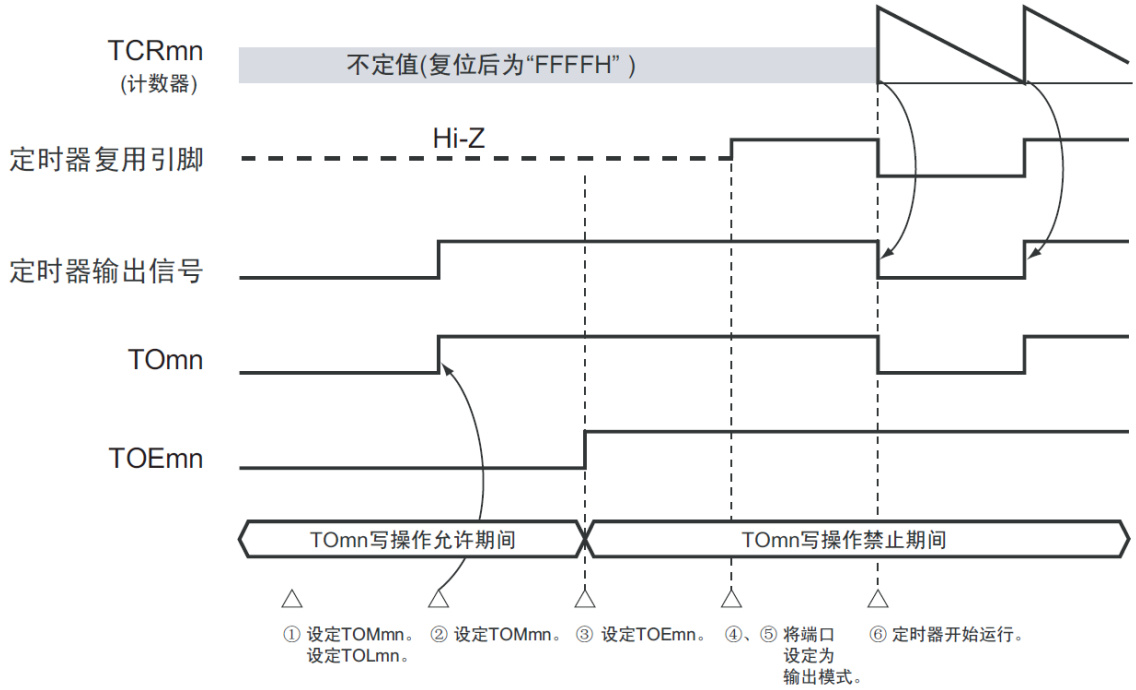
⑤ 能随时读 TOm 寄存器，并且能确认 TOmn 引脚的输出电平。

注：m：单元号（m=0，1） n：通道号 n=0~3（主控通道：n=0、2） p：从属通道号（n=0：p=1、2、3 n=2：p=3）

6.5.2 TOmn引脚的输出设定

从TOmn输出引脚的初始设定到定时器开始运行的步骤和状态变化如下所示。

图6-10：从设定定时器的输出到开始运行的状态变化



① 设定定时器输出的运行模式。

TOMmn 位（0：主控通道输出模式、1：从属通道输出模式）

TOLmn 位（0：正逻辑输出、1：负逻辑输出）

② 通过设定定时器输出寄存器 m（TOm），将定时器输出信号设定为初始状态。

③ 给 TOEmn 位写“1”，允许定时器输出（禁止写 TOm 寄存器）。

④ 通过端口模式控制寄存器（PMCxx）将端口设定为数字输入/输出。

⑤ 将端口的输入/输出设定为输出。

⑥ 允许定时器运行（TSmn=1）。

注：m：单元号（m=0） n：通道号（n=0～3）

6.5.3 通道输出运行的注意事项

(1) 有关定时器运行中的 TOm、TOEm、TOLm、TOMm 寄存器的设定值变更

定时器的运行（定时器计数寄存器 mn（TCRmn）和定时器数据寄存器 mn（TDRmn）的运行）和 TOmn 输出电路相互独立。因此，定时器输出寄存器 m（TOm）、定时器输出允许寄存器 m（TOEm）和定时器输出电平寄存器 m（TOLm）的设定值的变更不会影响定时器的运行，能在定时器运行中更改设定值。但是，为了在各定时器的运行中从 TOmn 引脚输出期待的波形，必须设定为 6.7 通用定时器单元的独立通道运行功能 和 6.8 通用定时器单元的多通道联动功能所示的各运行的寄存器设定内容例子的值。

如果在产生各通道的定时器中断（INTTMmn）信号前后更改除了 TOm 寄存器以外的 TOEm 寄存器和 TOLm 寄存器的设定值，就根据是在产生定时器中断（INTTMmn）信号前更改还是在产生后更改，TOmn 引脚输出的波形可能不同。

注：m：单元号（m=0） n：通道号（n=0～3）

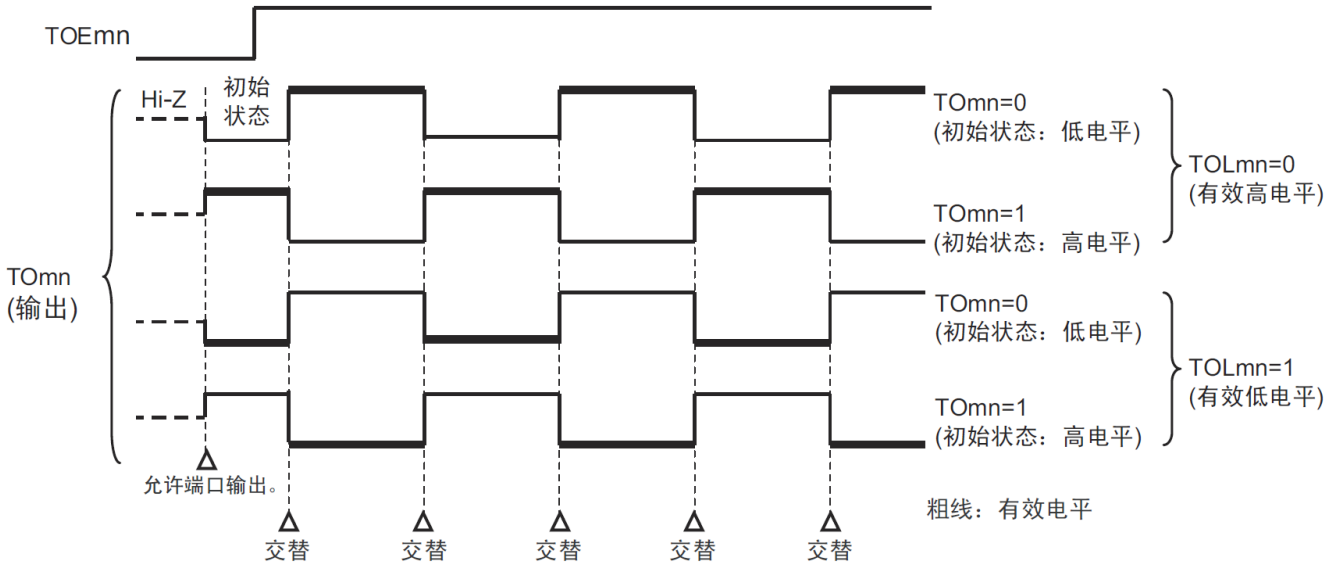
(2) 有关 TOmn 引脚的初始电平和定时器开始运行后的输出电平

在允许端口输出前并且在禁止定时器输出（TOEmn=0）的状态下写定时器输出寄存器 m（TOm），在更改初始电平后设定为定时器输出允许状态（TOEmn=1）时的 TOmn 引脚输出电平的变化如下所示。

1) 在主控通道输出模式（TOMmn=0）中开始运行的情况

在主控通道输出模式（TOMmn=0）中，定时器输出电平寄存器 m（TOLm）的设定无效。如果在设定初始电平后开始定时器的运行，就通过产生交替信号反相 TOmn 引脚的输出电平。

图6-11：交替输出时（TOMmn=0）的TOmn引脚输出状态



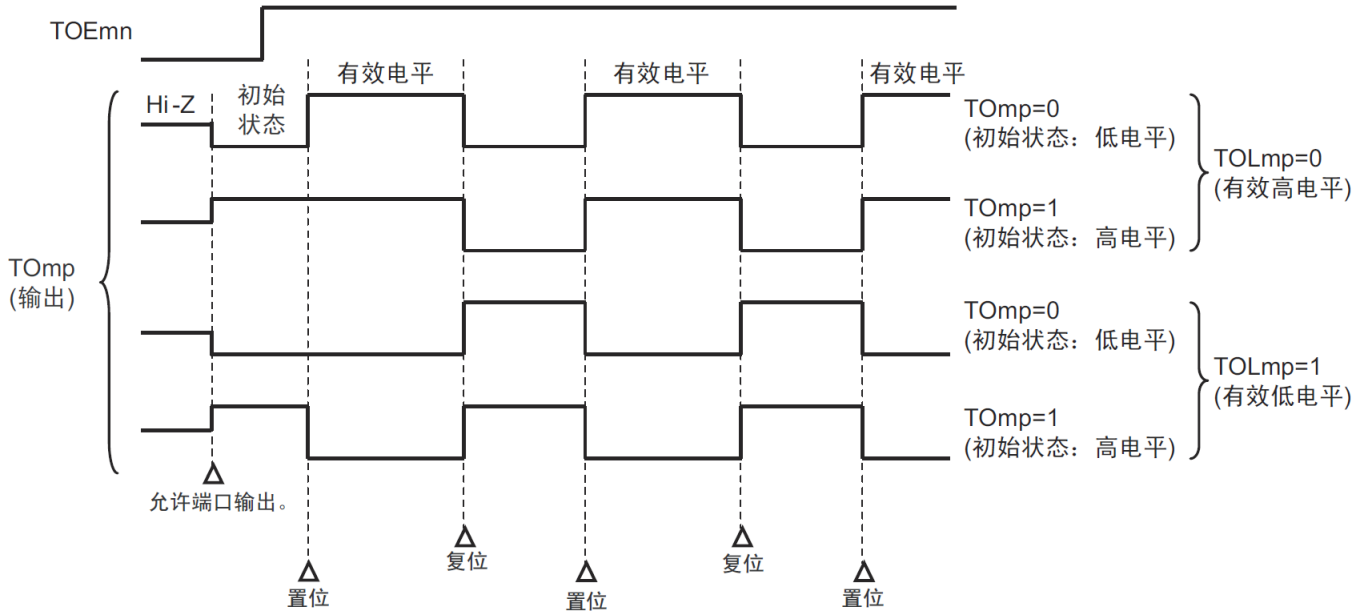
注1：交替：反相TOmn引脚的输出状态。

注2：m：单元号（m=0） n：通道号（n=0～3）

1) 在从属通道输出模式（TOMmn=1）中开始运行的情况（PWM 输出）

在从属通道输出模式（TOMmn=1）中，有效电平取决于定时器输出电平寄存器 m（TOLmn）的设定。

图6-12：PWM输出时（TOMmn=1）的TOmn引脚输出状态



注1：置位：TOmp引脚的输出信号从无效电平变为有效电平。

注2：复位：TOmp引脚的输出信号从有效电平变为无效电平。

注3：m：单元号（m=0） n：通道号（p=1～3）

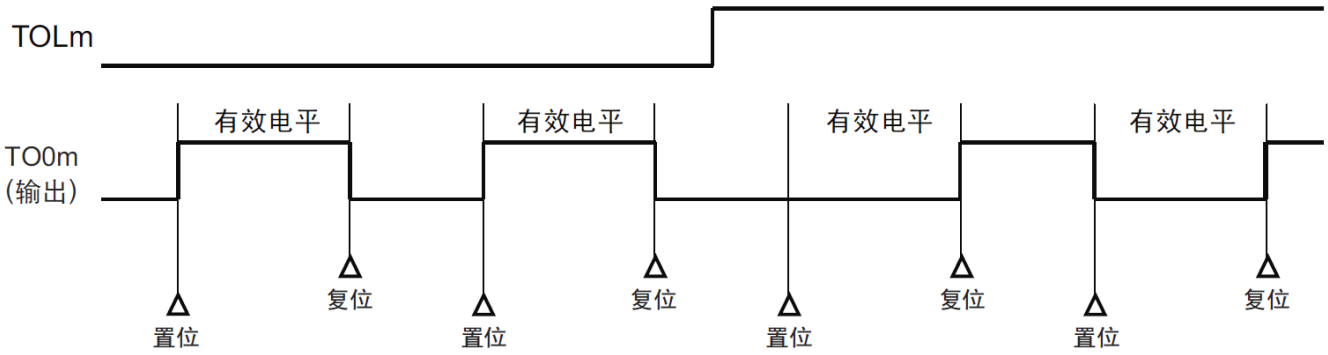
(3) 有关从属通道输出模式（TOMmn=1）的 TOmn 引脚变化

1) 在定时器运行中更改定时器输出电平寄存器 m（TOLm）的设定的情况

如果在定时器运行中更改 TOLm 寄存器的设定，就在产生 TOmn 引脚变化条件时设定有效。无法通过改写 TOLm 寄存器来改变 TOmn 引脚的输出电平。

当 TOMmn 位为“1”时，在定时器运行中（TEmn=1）更改 TOLm 寄存器的值时的运行如下所示。

图6-13：在定时器运行中更改TOLm寄存器的内容时的运行



注1：置位：TOmn引脚的输出信号从无效电平变为有效电平。

注2：复位：TOmn引脚的输出信号从有效电平变为无效电平。

注3：m：单元号（m=0）n：通道号（n=0～3）

2) 置位/复位时序

3) 为了在 PWM 输出时实现 0%和 100%的输出，通过从属通道将产生主控通道定时器中断（INTTMmn）时的 TOmn 引脚/TOmn 位的置位时序延迟 1 个计数时钟。

当置位条件和复位条件同时产生时，优先复位条件。

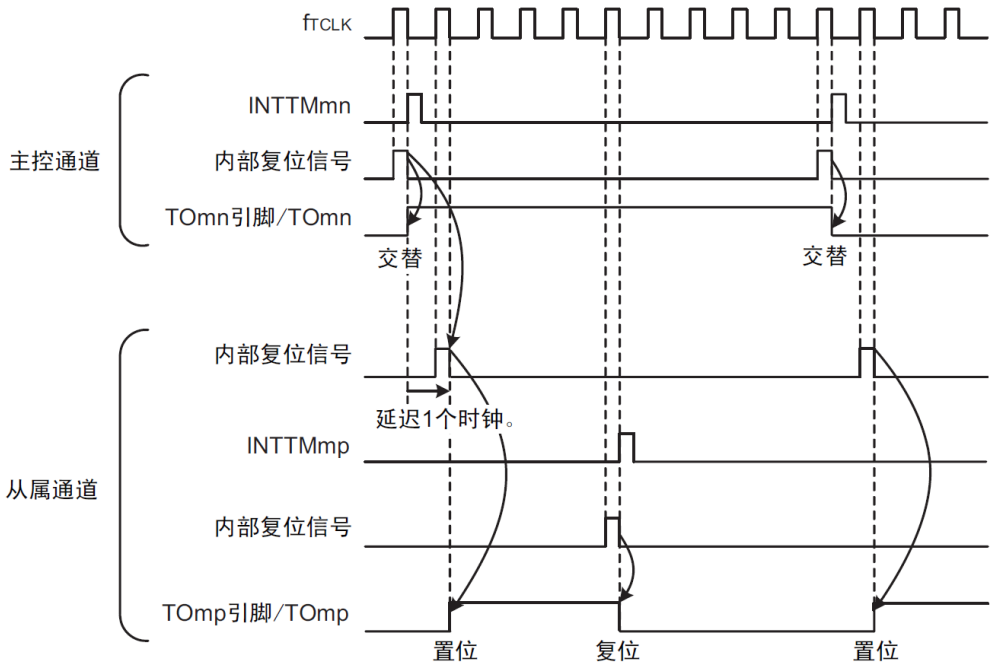
按照以下方法设定主控/从属通道时的置位/复位运行状态如图 6-2814 所示。

主控通道：TOEmn=1、TOMmn=0、TOLmn=0

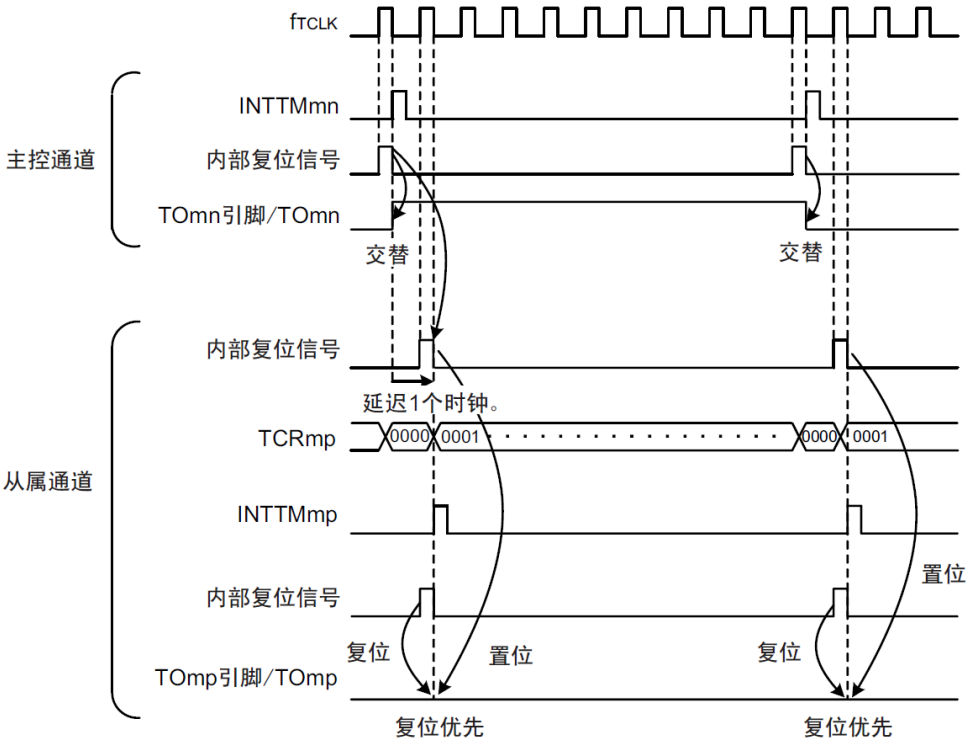
从属通道：TOEmp=1、TOMmp=1、TOLmp=0

图 6-14：置位/复位时序运行状态

(1) 基本运行时序



(2) 0%占空比的运行时序



注1：内部复位信号：TOmn引脚的复位/交替信号

注2：内部置位信号：TOmn引脚的置位信号

注3：m：单元号（m=0）n：通道号 n=0～3（主控通道：n=0、2）

p：从属通道号n=0：p=1、2、3 n=2：p=3

6.5.4 TOmn位的一次性操作

和定时器通道开始寄存器 m（TSm）相同，定时器输出寄存器 m（TOm）有全部通道的设定位（TOmn），因此能一次性地操作全部通道的 TOmn 位。

表6-27：TO0n位的一次性操作例子

写前

TO0	0	0	0	0	0	0	0	0	0	0	0	0	TO03 1	TO02 0	TO01 1	TO00 0
TOE0	0	0	0	0	0	0	0	0	0	0	0	0	TOE03 0	TOE02 0	TOE01 0	TOE00 1

要写的数据

0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1

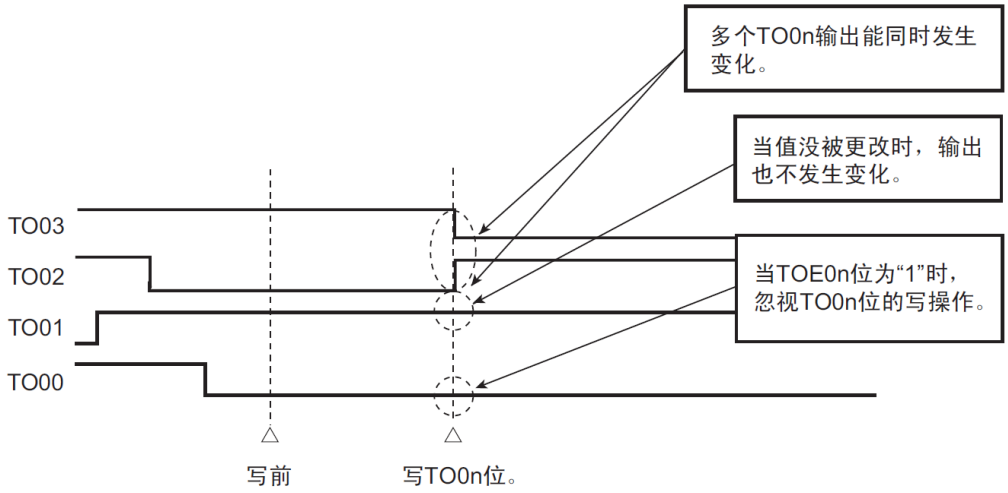
写后

TO0	0	0	0	0	0	0	0	0	0	0	0	0	TO03 0	TO02 1	TO01 1	TO00 0

只能写 TOEmn 位为“0”的 TOmn 位，忽视 TOEmn 位为“1”的 TOmn 位的写操作。

TOEmn 位为“1”的 TOmn（通道输出）不受写操作的影响，即使写 TOmn 位也被忽视，由定时器运行引起的输出变化正常进行。

图6-15：一次性操作TO0n位时的TO0n引脚状态



注：m：单元号（m=0） n：通道号（n=0～3）

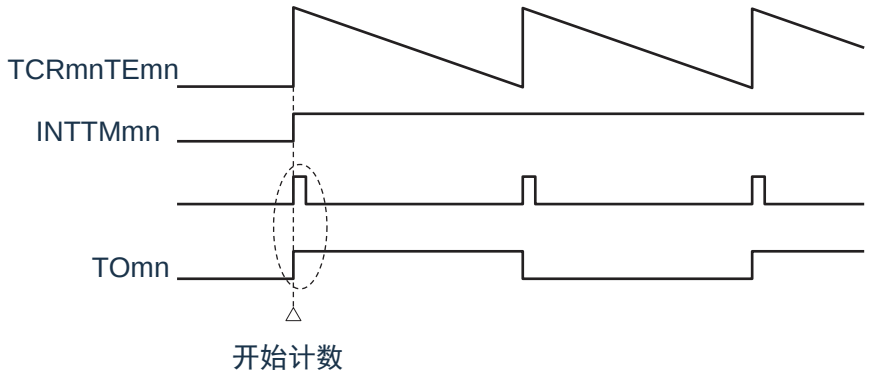
6.5.5 有关开始计数时的定时器中断和TOmn引脚输出

在间隔定时器模式或者捕捉模式中，定时器模式寄存器 mn（TMRmn）的 MDmn0 位是设定是否在开始计数时产生定时器中断的位。

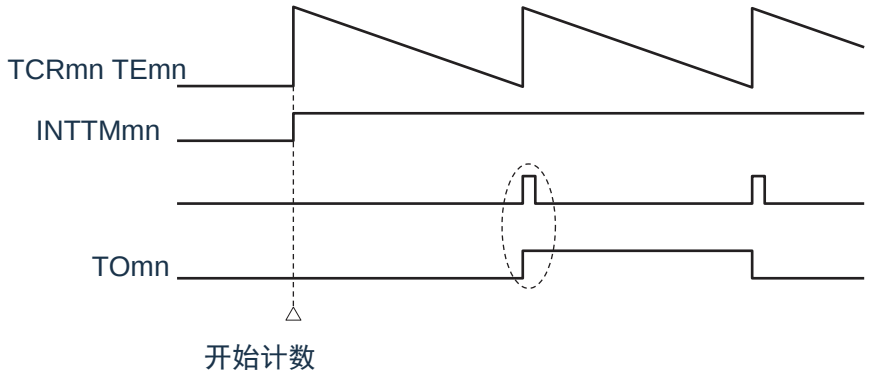
当 MDmn0 位为“1”时，能通过产生定时器中断（INTTMmn）得知计数的开始时序。在其他模式中，不控制开始计数时的定时器中断和 TOmn 输出。设定为间隔定时器模式（TOEmn=1、TOMmn=0）时的运行例子如下所示。

图6-16：开始计数时的定时器中断和TOmn输出的运行例子

(a) MDmn0位为“1”的情况



(b) MDmn0位为“0”的情况



当 MDmn0 位为“1”时，在开始计数时输出定时器中断（INTTMmn）并且 TOmn 进行交替输出。

当 MDmn0 位为“0”时，在开始计数时不输出定时器中断（INTTMmn）并且 TOmn 也不发生变化，而在对 1 个周期进行计数后输出 INTTMmn 并且 TOmn 进行交替输出。

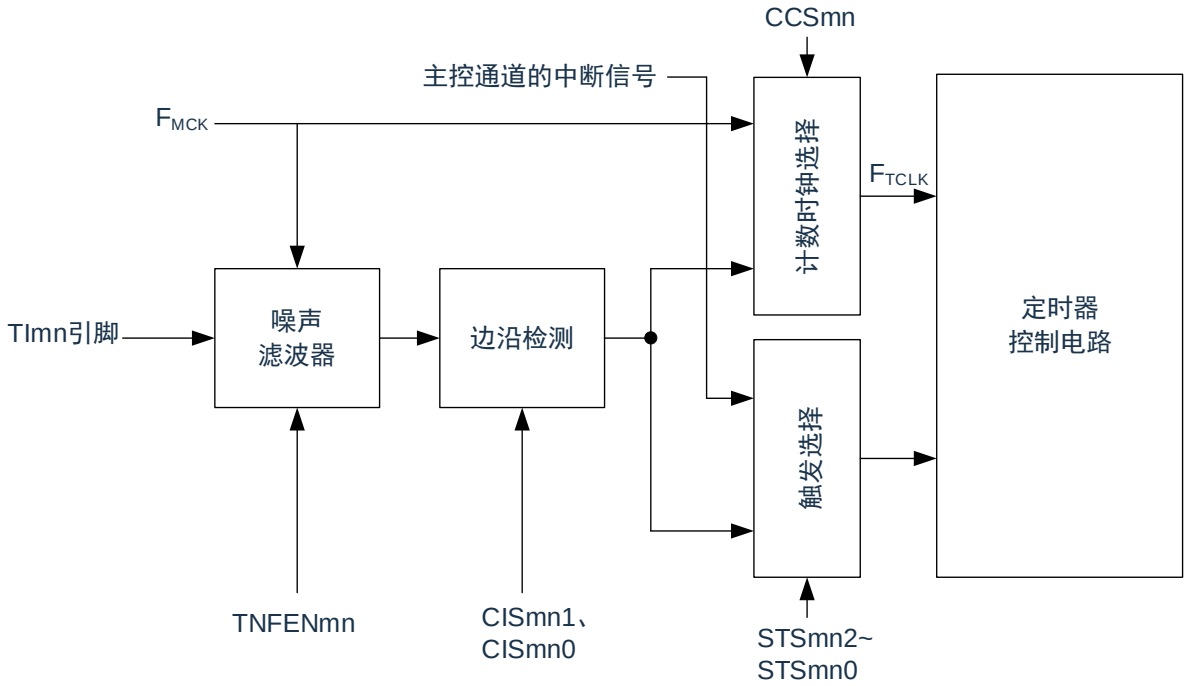
注：m：单元号（m=0）n：通道号（n=0～3）

6.6 定时器输入（TImn）的控制

6.6.1 TImn引脚输入电路的结构

定时器输入引脚的信号通过噪声滤波器和边沿检测电路输入到定时器控制电路。对于需要消除噪声的引脚，必须将对应的引脚噪声滤波器置为有效。输入电路的结构如下所示。

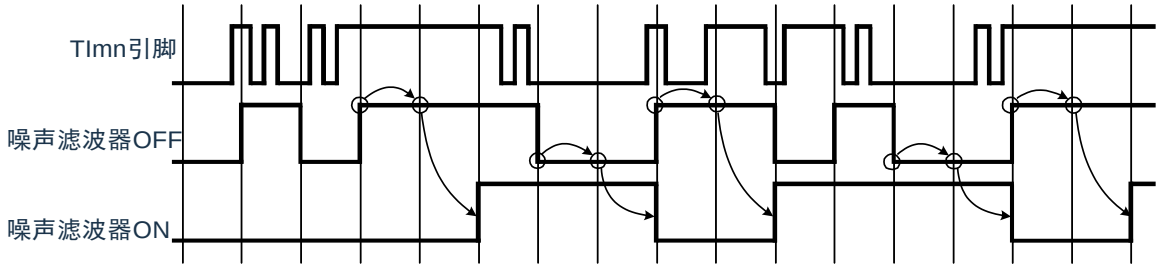
图6-17：输入电路的结构



6.6.2 噪声滤波器

当噪声滤波器无效时，只通过通道 n 的运行时钟（F_{MCK}）进行同步；当噪声滤波器有效时，在通过通道 n 的运行时钟（F_{MCK}）进行同步后检测 2 个时钟是否一致。TM4Imn 输入引脚在噪声滤波器 ON 或者 OFF 的情况下，经过噪声滤波器电路后的波形如下所示。

图6-18：TImn输入引脚在噪声滤波器ON或者OFF情况下的采样波形



注：TImn引脚的输入波形用于说明噪声滤波器ON或者OFF的运行。实际使用时，必须按照数据手册的AC特性所示的TImn输入高低电平宽度进行输入。

6.6.3 操作通道输入时的注意事项

在设定为不使用定时器输入引脚时，不给噪声滤波器电路提供运行时钟。因此，从设定为使用定时器输入引脚到设定定时器输入引脚对应的通道运行允许触发，需要以下的等待时间。

(1) 噪声滤波器为 OFF 的情况

如果在定时器模式寄存器 mn（TMRmn）的 bit12（CCSmn）、bit9（STSmn1）和 bit8（STSmn0）全都为“0”的状态下将任意一位置位，就必须至少在经过 2 个运行时钟（F_{MCK}）周期后将定时器通道开始寄存器（TSm）的运行允许触发置位。

(2) 噪声滤波器为 ON 的情况

如果在定时器模式寄存器 mn（TMRmn）的 bit12（CCSmn）、bit9（STSmn1）和 bit8（STSmn0）全都为“0”的状态下将任意一位置位，就必须至少在经过 4 个运行时钟（F_{MCK}）周期后将定时器通道开始寄存器（TSm）的运行允许触发置位。

6.7 通用定时器单元的独立通道运行功能

6.7.1 作为间隔定时器/ 方波输出的运行

(1) 间隔定时器

能用作以固定间隔产生 INTTMmn （定时器中断）的基准定时器。 中断产生周期能用以下计算式进行计算：

INTTMmn （定时器中断）的产生周期 = 计数时钟周期 × (TDRmn 的设定值 +1)

(2) 作为方波输出的运行

TOmn 在产生 INTTMmn 的同时进行交替输出，输出占空比为 50% 的方波。

TOmn 输出方波的周期和频率能用以下计算式进行计算：

TOmn 输出的方波周期 = 计数时钟周期 × (TDRmn 的设定值 +1)x2

TOmn 输出的方波频率 = 计数时钟频率 / {(TDRmn 的设定值 +1)x2}

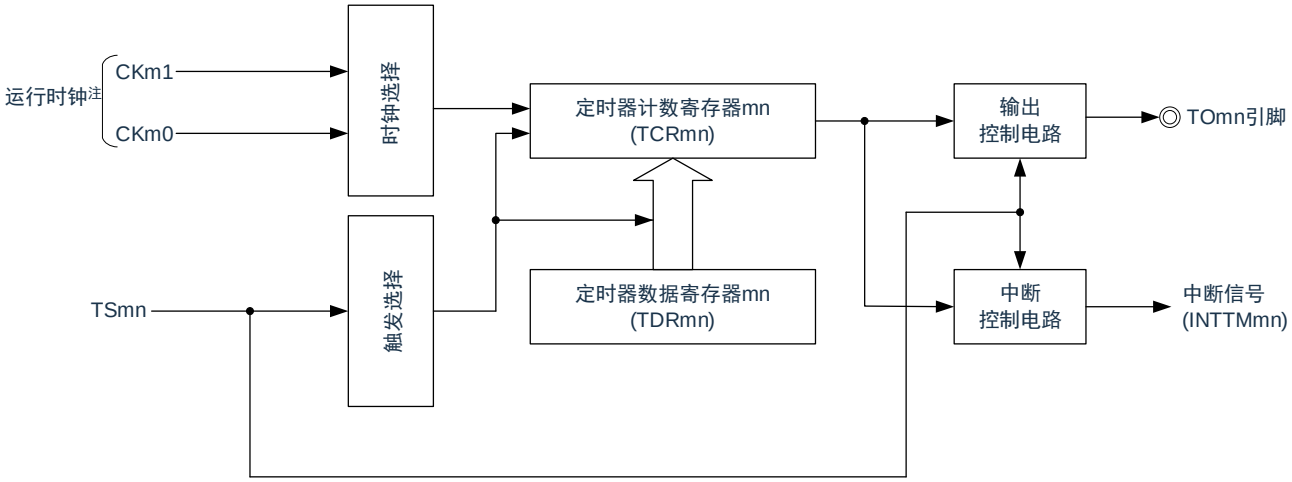
在间隔定时器模式中，定时器计数寄存器mn （TCRmn）用作递减计数器。

在将定时器通道开始寄存器m （TSm）的通道开始触发位（TSmn、TSHm1、TSHm3）置“1” 后，通过第 1 个计数时钟将定时器数据寄存器 mn （TDRmn）的值装入 TCRmn 寄存器。此时，如果定时器模式寄存器 n （TMRmn）的 MDmn0 位为 “0”，就不输出 INTTMmn 并且 TOmn 也不进行交替输出。如果 TMRmn 寄存器的 MDmn0 位为“1”，就输出INTTMmn 并且TOmn 进行交替输出。 然后，TCRmn 寄存器通过计数时钟进行递减计数。

如果TCRmn 变为“0000H”，就通过下一个计数时钟输出INTTMmn 并且TOmn 进行交替输出。同时，再次将 TDRmn 寄存器的值装入TCRmn 寄存器。此后，继续同样的运行。

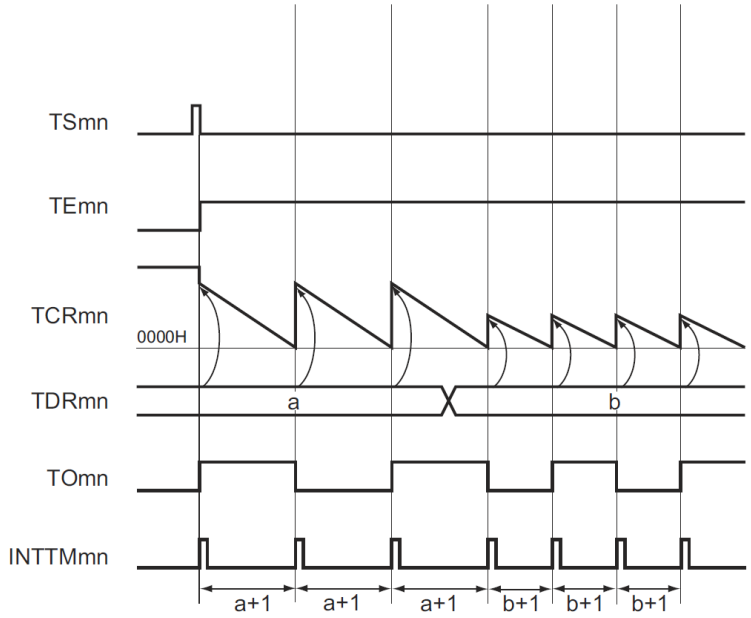
能随时改写TDRmn 寄存器，改写的TDRmn 寄存器的值从下一个周期开始有效。

图6-19：作为间隔定时器/ 方波输出运行的基本时序例子（MDmn0=1）



注：在通道1 和通道3 时，能从CKm0、CKm1、CKm2 和CKm3 中选择时钟。

图6-20：作为间隔定时器/ 方波输出运行的基本时序例子（MDmn0=1）



注1：m：单元号（m= 0） n：通道号（n=0 ～3）

注2：TSmn：定时器通道开始寄存器m （TSm）的bit n

TEmn：定时器通道允许状态寄存器m （TEm）的bit n

TCRmn：定时器计数寄存器 mn （TCRmn）

TDRmn：定时器数据寄存器 mn （TDRmn）

TOmn：TOmn 引脚输出信号

图6-21：间隔定时器/ 方波输出时的寄存器设定内容例子

(a) 定时器模式寄存器mn （TMRmn）

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMRmn	CKSmn1 1/0	CKSmn0 1/0	0	CCSmn 0	M/S 注 0/1	STSmn2 0	STSmn1 0	STSmn0 0	CISmn1 0	CISmn0 0	0	0	MDmn3 0	MDmn2 0	MDmn1 0	MDmn0 1/0

通道 n 的运行模式
000B: 间隔定时器

开始计数时的运行设定
0：在开始计数时，不产生 INTTMmn
并且不反相定时器的输出。
1：在开始计数时，产生 INTTMmn
并且反相定时器的输出。

Timn 引脚输入边沿的选择
00B：因不使用而置 “00”。

开始触发的选择
000B: 只选择软件开始触发。

MASTERmn位的设定（通道2）
0： 独立通道运行
SPLITmn位的设定（通道1、3）
0： 16位定时器
1： 8位定时器

计数时钟的选择
0：选择运行时钟 （fMCK）。

运行时钟 （fMCK）的选择
00B：选择 CKm0 作为通道 n 的运行时钟。
10B：选择 CKm1 作为通道 n 的运行时钟。
01B：选择 CKm2 作为通道 1、 3 的运行时钟 （只有通道 1、 3 才能选择）。
11B：选择 CKm3 作为通道 1、 3 的运行时钟 （只有通道 1、 3 才能选择）。

(b) 定时器输出寄存器m （TOm）

	bitn
TOm	TOmn 1/0

0：由TOmn 输出“0”。
1：由TOmn 输出“1”。

(c) 定时器输出允许寄存器m （TOEm）

	bit n
TOEm	TOEmn 1/0

0：停止由计数运行进行的 TOmn 输出。
1：允许由计数运行进行的 TOmn 输出。

(d) 定时器输出电平寄存器m （TOLm）

	bit n
TOLm	TOLmn 0

0：在 TOMmn=0 （主控通道输出模式）时置 “0”。

(e) 定时器输出模式寄存器m （TOMm）

	bit n
TOMm	TOMmn 0

0：设定主控通道输出模式。

注1：TMRm2：MASTERmn 位

注2：TMRm1、TMRm3：SPLITmn位

注3：TMRm0：固定为“0”。

注4：m：单元号（m= 0） n：通道号（n=0 ～3）

表6-28：间隔定时器/ 方波输出功能时的操作步骤

	软件操作	硬件状态
TAU 初始设定		定时器单元 m 的输入时钟处于停止提供状态。 （停止提供时钟，不能写各寄存器）
	将外围允许寄存器0 （PER0）的TM4mEN 位置“1”。 →	定时器单元m 的输入时钟处于提供状态。 （开始提供时钟，能写各寄存器）
	设定定时器时钟选择寄存器m （TPSm）。 确定CKm0 ～ CKm3 的时钟频率。	
通道初始设定	设定定时器模式寄存器mn （TMRmn）（确定通道的运行模式）。 给定时器数据寄存器mn（TDRmn）设定间隔（周期）值。	通道处于运行停止状态。 （提供时钟，消耗一部分电力）
	使用TOmn 输出的情况： 将定时器输出模式寄存器m（TOMm）的TOMmn位置“0”（主控通道输出模式）。 将TOLmn 位置“0”。 设定TOmn 位，确定TOmn 输出的初始电平。 →	TOmn 引脚处于Hi-Z 输出状态。 当端口模式寄存器为输出模式并且端口寄存器为“0”时，输出TOmn 初始设定的电平。
	将TOEmn 位置“1”，允许TOmn 输出。 →	因为通道处于运行停止状态，所以TOmn 不变。
	将端口寄存器和端口模式寄存器置“0”。 →	TOmn 引脚输出TOmn 设定的电平。
开始运行	（只在使用TOmn 输出并且重新开始时，将TOEmn位置“1”） 将TSmn （TSHm1、TSHm3）位置“1”。 →	TEmn（TEHm1、TEHm3）位变为“1” 并且开始计数。 将TDRmn 寄存器的值装入定时器计数寄存器mn（TCRmn）。当TMRmn寄存器的MDmn0位为“1”时，产生INTTMmn 并且TOmn 进行交替输出。
运行中	能任意更改TDRmn 寄存器的设定值。 能随时读TCRmn 寄存器。 不使用TSRmn 寄存器。 能更改TOm 寄存器和TOEm 寄存器的设定值。 禁止更改TMRmn 寄存器、TOMmn 位和TOLmn 位的设定值。	计数器（TCRmn）进行递减计数。如果计数到“0000H”，就再次将TDRmn寄存器的值装入TCRmn寄存器并且继续计数。当检测到TCRmn 为“0000H”时，产生INTTMmn 并且TOmn 进行交替输出。 此后，重复此运行。
停止运行	将TTmn （TTHm1、TTHm3）位置“1”。 →	TEmn（TEHm1、TEHmn）位变为“0” 并且停止计数。 TCRmn 寄存器保持计数值而停止计数。 TOmn 输出不被初始化而保持状态。
	因为TTmn（TTHm1、TTHm3）位是触发位，所以自动返回到“0”。	
	将TOEmn 位置“0” 并且给TOmn 位设定值。 →	TOmn 引脚输出TOmn 位设定的电平。
TAU 停止	要保持TOmn 引脚输出电平的情况： 在给端口寄存器设定要保持的值后将TOmn 位置“0”。 →	通过端口功能保持TOmn 引脚的输出电平。
	不需要保持TOmn 引脚输出电平的情况： 不需要设定。	
	将PER0 寄存器的TM4mEN 位置“0”。 →	定时器单元m 的输入时钟处于停止提供状态。 对全部电路和各通道的SFR 进行初始化。 （TOmn位变为“0”并且TOmn引脚变为端口功能）

重新开始运行

注：m：单元号 （m= 0） n：通道号 （n=0 ～ 3）

6.7.2 作为外部事件计数器的运行

能用作事件计数器，对检测到的 TImn 引脚输入的有效边沿 （外部事件）进行计数，如果达到规定的计数值，就产生中断。规定的计数值能用以下计算式进行计算：

规定的计数值 = TDRmn 的设定值 +1

在事件计数器模式中，定时器计数寄存器mn （TCRmn）用作递减计数器。

通过将定时器通道开始寄存器m （TSm）的任意通道开始触发位（TSmn、TSHm1、TSHm3）置“1”，将定时器数据寄存器mn （TDRmn）的值装入TCRmn 寄存器。

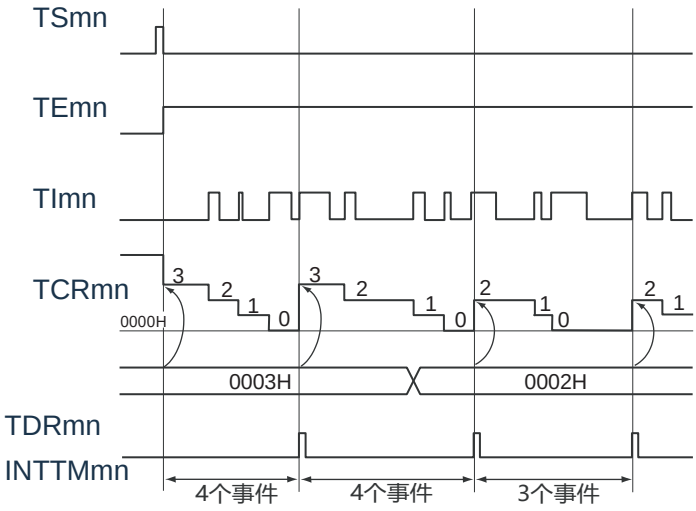
TCRmn 寄存器在检测到 TImn 引脚输入的有效边沿的同时进行递减计数。如果 TCRmn 变为 “0000H”，就再次装入TDRmn 寄存器的值并且输出INTTMmn。

此后，继续同样的运行。

因为TOmn 引脚根据外部事件输出不规则的波形，所以必须将定时器输出允许寄存器m（TOEm）的TOEmn 位置“0”，停止输出。

能随时改写TDRmn 寄存器，改写的TDRmn 寄存器的值在下一个计数期间有效。

图6-22：作为外部事件计数器运行的基本时序例子



注1：m：单元号（m= 0） n：通道号（n=0 ～3）

注2：TSmn：定时器通道开始寄存器 m （TSm）的 bit n

TEmn：定时器通道允许状态寄存器 m （TEm）的 bit n

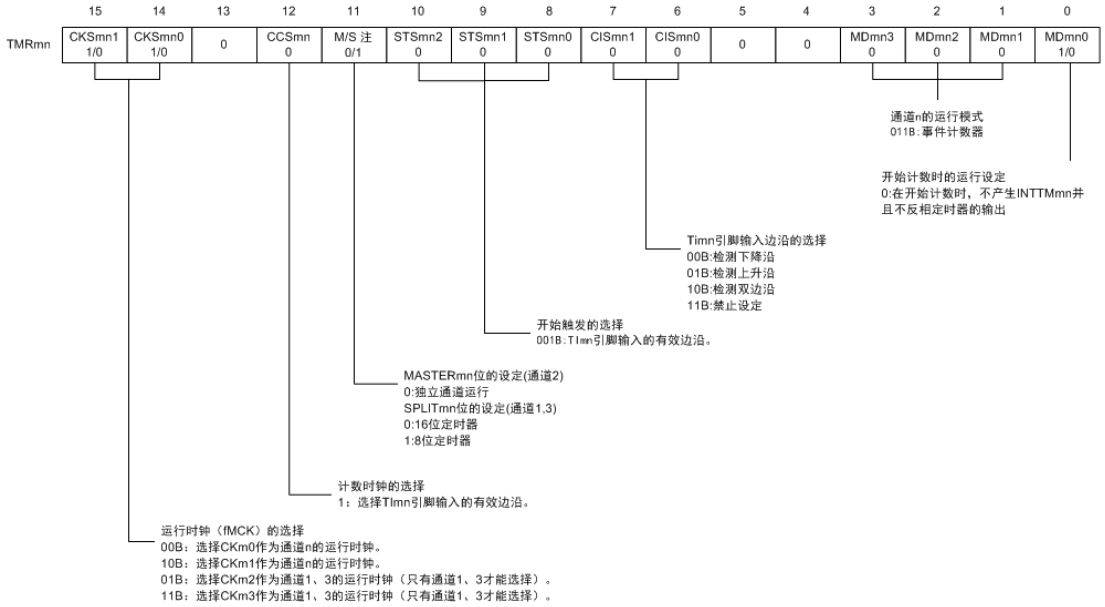
TImn：TImn 引脚输入信号

TCRmn ：定时器计数寄存器 mn （TCRmn）

TDRmn ：定时器数据寄存器 mn （TDRmn）

图6-23：外部事件计数器模式时的寄存器设定内容例子

(a) 定时器模式寄存器mn （TMRmn）



(b) 定时器输出寄存器m （TOm）

	bit n	
TOm	TOmn 1/0	0：由 TOmn 输出 “0”。

(c) 定时器输出允许寄存器m （TOEm）

	bit n	
TOEm	TOEmn 1/0	0：停止由计数运行进行的 TOmn 输出。 1：允许由计数运行进行的 TOmn 输出。

(d) 定时器输出电平寄存器m （TOLm）

	bit n	
TOLm	TOLmn 0	0：在 TOMmn=0 （主控通道输出模式）时置 “0”。

(e) 定时器输出模式寄存器m （TOMm）

	bit n	
TOMm	TOMmn 0	0：设定主控通道输出模式。

注1：TMRm2：MASTERmn 位

注2：TMRm1、 TMRm3 ：SPLITmn位

注3：TMRm0：固定为“0”。

注4：m：单元号（m= 0）n：通道号（n=0 ～3）

表6-29：外部事件计数器功能时的操作步骤

	软件操作	硬件状态
Timer4 初始设定		定时器单元m 的输入时钟处于停止提供的状态。 （停止提供时钟，不能写各寄存器）
	将外围允许寄存器0 （PER0）的TM4mEN 位置“1”。 →	定时器单元m的输入时钟处于提供状态，各通道处于运行停止状态。 （开始提供时钟，能写各寄存器）
	设定定时器时钟选择寄存器m （TPSm）。 确定CKm0 ~ CKm3 的时钟频率。	
通道初始设定	将噪声滤波器允许寄存器1（NFEN12）的对应位置“0” （OFF）或者“1” （ON）。 设定定时器模式寄存器mn （TMRmn）（确定通道的运行模式）。 给定时器数据寄存器mn （TDRmn）设定计数值。 将定时器输出允许寄存器m （TOEm）的TOEmn 位置“0”。	通道处于运行停止状态。 （提供时钟，消耗一部分电力）
开始运行	将TSmn 位置“1”。 因为TSmn 位是触发位，所以自动返回到“0”。 →	TEmn 位变为“1” 并且开始计数。 将TDRmn 寄存器的值装入定时器计数寄存器mn（TCRmn），进入TImn 引脚输入边沿的检测等待状态。
运行中	能任意更改TDRmn 寄存器的设定值。 能随时读TCRmn 寄存器。 不使用TSRmn 寄存器。 禁止更改TMRmn 寄存器、TOMmn位、TOLmn 位、TOmn 位和TOEmn 位的设定值。	每当检测到TImn 引脚的输入边沿时，计数器（TCRmn）就进行递减计数。如果计数到“0000H”，就再次将TDRmn 寄存器的值装入TCRmn 寄存器并且继续计数。当检测到TCRmn 为“0000H” 时，产生INTTMmn。 此后，重复此运行。
停止运行	将TTmn 位置“1”。 因为TTmn 位是触发位，所以自动返回到“0”。 →	TEmn 位变为“0” 并且停止计数。 TCRmn 寄存器保持计数值而停止计数。
Timer4 停止	将PER0 寄存器的TM4mEN 位置“0”。 →	定时器单元m 的输入时钟处于停止提供状态。 对全部电路和各通道的SFR 进行初始化。

重新开始运行

6.7.3 作为分频器的运行

能对TI00 引脚输入的时钟进行分频并且用作TO00 引脚输出的分频器。

TO00 输出的分频时钟频率能用以下计算式进行计算：

- 选择上升沿或者下降沿的情况：

分频时钟频率 = 输入时钟频率 / {(TDR00 的设定值 +1) ´ 2}

- 选择双边沿的情况：

分频时钟频率≈输入时钟频率 / (TDR00 的设定值 +1)

在间隔定时器模式中，定时器计数寄存器 00 （TCR00）用作递减计数器。

在将定时器通道开始寄存器 0 （TS0）的通道开始触发位 （TS00）置 “1” 后，通过检测到 TI00 的有效边沿将定时器数据寄存器 00 （TDR00）的值装入 TCR00 寄存器。此时，如果定时器模式寄存器 00 （TMR00）的 MD000 位为 “0”，就不输出 INTTM00 并且 TO00 不进行交替输出；如果 TMR00 寄存器的 MD000 位为“1”，就输出 INTTM00 并且 TO00 进行交替输出。

然后，TCR00 寄存器通过 TI00 引脚输入的有效边沿进行递减计数。如果 TCR00 变为 “0000H”，TO00 就进行交替输出。同时，将 TDR00 寄存器的值装入 TCR00 寄存器并且继续计数。

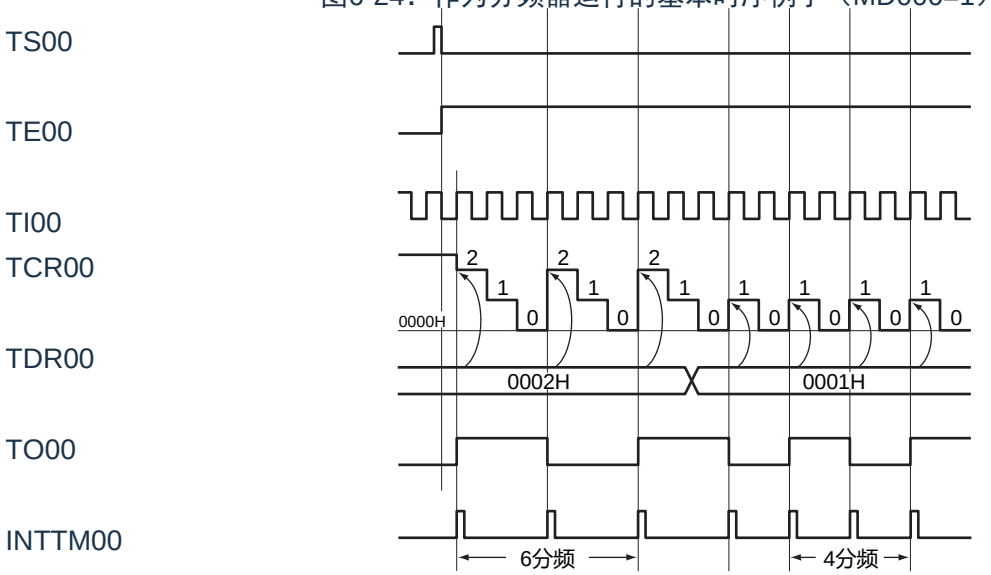
如果选择 TI00 引脚输入的双边沿检测，输入时钟的占空比误差就会影响 TO00 输出的分频时钟周期。

TO00 输出的时钟周期包含 1 个运行时钟周期的采样误差。

TO00 输出的时钟周期= 理想的TO00 输出时钟周期± 运行时钟周期 （误差）

能随时改写TDR00 寄存器，改写的TDR00 寄存器的值在下一个计数期间有效。

图6-24：作为分频器运行的基本时序例子（MD000=1）



注：TS00：定时器通道开始寄存器0 （TS0）的bit0

TE00：定时器通道允许状态寄存器 0 （TE0）的 bit0

TI00：TI00 引脚输入信号

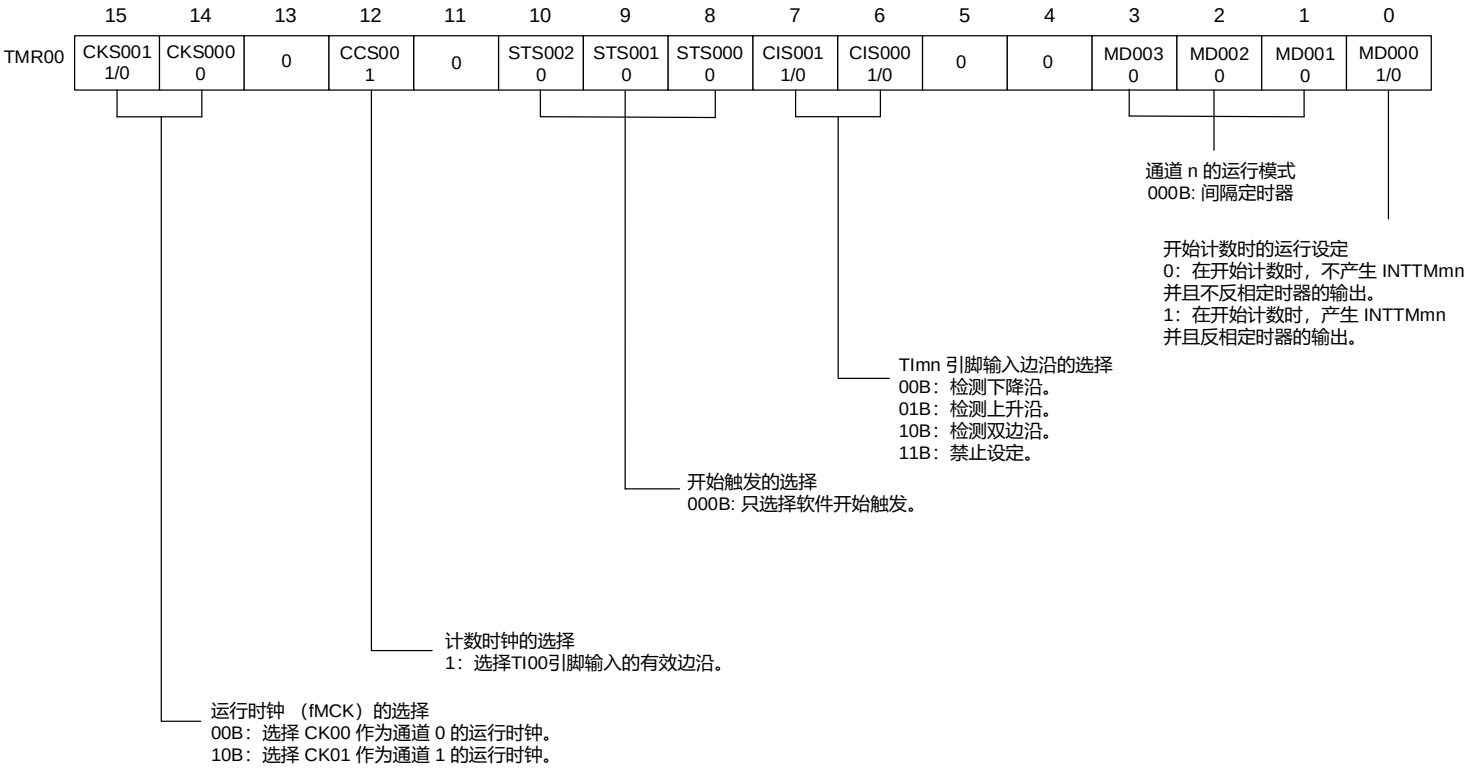
TCR00：定时器计数寄存器 00 （TCR00）

TDR00：定时器数据寄存器 00 （TDR00）

TO00：TO00 引脚输出信号

图6-25：作为分频器运行时的寄存器设定内容例子

(a) 定时器模式寄存器00 （TMR00）



(b) 定时器输出寄存器0 （TO0）

	bit 0	
TO0	TO00 1/0	0：由TO00 输出“0”。 1：由TO00 输出“1”。

(c) 定时器输出允许寄存器0 （TOE0）

	bit 0	
TOE0	TOE00 1/0	0：停止由计数运行进行的 TO00 输出。 1：允许由计数运行进行的 TO00 输出。

(d) 定时器输出电平寄存器0 （TOL0）

	bit 0	
TOL0	TOL00 0	0：在主控通道输出模式 （TOM00=0）中置 “0”。

(e) 定时器输出模式寄存器0 （TOM0）

	bit n	
TOM0	TOM00 0	0：设定主控通道输出模式。

表6-30：分频器功能时的操作步骤

	软件操作	硬件状态
Timer4初始设定		定时器单元 0 的输入时钟处于停止提供状态。 （停止提供时钟，不能写各寄存器）
	将外围允许寄存器0 （PER0）的TM4mEN 位置“1”。 →	定时器单元0 的输入时钟处于提供状态。 （开始提供时钟，能写各寄存器）
	设定定时器时钟选择寄存器0 （TPS0）。 确定CK00 ~ CK03 的时钟频率。	
通道初始设定	将噪声滤波器允许寄存器1（NFEN1）的对应位置“0”（OFF）或者“1” （ON）。 设定定时器模式寄存器00（TMR00）（确定通道的运行模式，选择检测边沿）。 给定时器数据寄存器00 （TDR00）设定间隔（周期）值。	通道处于运行停止状态。 （提供时钟，消耗一部分电力）
	将定时器输出模式寄存器0 （TOM0）的TOM00 位置“0” （主控通道输出模式）。 将TOL00 位置“0”。 设定TO00 位并且确定TO00 输出的初始电平。 →	TO00 引脚处于Hi-Z 输出状态。 当端口模式寄存器为输出模式并且端口寄存器为“0”时，输出TO00 初始设定的电平。
	将TOE00 位置“1”，允许TO00 输出。 →	因为通道处于运行停止状态，所以TO00 不变。
	将端口寄存器和端口模式寄存器置“0”。 →	TO00 引脚输出TO00 设定的电平。
开始运行	将TOE00 位置“1” （只限于重新开始运行）。 将TS00 位置“1”。 → 因为TS00 位是触发位，所以自动返回到“0”。	TE00位变为“1” 并且开始计数。 将TDR00 寄存器的值装入定时器计数寄存器00（TCR00）。当TMR00寄存器的MD000位为“1”时，产生INTTM00 并且TO00 进行交替输出。
运行中	能任意更改TDR00 寄存器的设定值。 能随时读TCR00 寄存器。 不使用TSR00 寄存器。 能更改TO0 寄存器和TOE0 寄存器的设定值。 禁止更改TMR00 寄存器、TOM00 位和TOL00 位的设定值。	计数器（TCR00）进行递减计数。如果计数到“0000H”，就再次将TDR00寄存器的值装入TCR00 寄存器并且继续计数。当检测到TCR00 为“0000H”时，产生INTTM00 并且TO00 进行交替输出。 此后，重复此运行。
停止运行	将TT00 位置“1”。 → 因为TT00位是触发位，所以自动返回到“0”。	TE00位变为“0” 并且停止计数。 TCR00 寄存器保持计数值而停止计数。 TO00 输出不被初始化而保持状态。
	将TOE00 位置“0” 并且给TO00 位设定值。 →	TO00 引脚输出TO00 位设定的电平。
Timer4停止	要保持TO00 引脚输出电平的情况： 在给端口寄存器设定要保持的值后将TO00 位置“0”。 → 不需要保持TO00 引脚输出电平的情况： 不需要设定。	通过端口功能保持TO00 引脚的输出电平。
	将PER0 寄存器的TM4mEN 位置“0”。 →	定时器单元0 的输入时钟处于停止提供状态。 对全部电路和各通道的SFR 进行初始化。 （TO00位变为“0”并且TO00引脚变为端口功能）

重新开始运行

6.7.4 作为输入脉冲间隔测量的运行

能在 TImn 有效边沿捕捉计数值，测量 TImn 输入脉冲的间隔。在 TEmn 位为 “1” 的期间，也能将软件操作（TSmn=1）设定为捕捉触发，捕捉计数值。

脉冲间隔能用以下计算式进行计算：

TImn 输入脉冲间隔= 计数时钟的周期´ ((10000H ´ TSRmn:OVF) + (TDRmn 的捕捉值+1))

注：因为通过定时器模式寄存器 mn（TMRmn）的 CKSmn 位选择的运行时钟对 TImn 引脚输入进行采样，所以产生1 个运行时钟的误差。

在捕捉模式中，定时器计数寄存器mn （TCRmn）用作递增计数器。

如果将定时器通道开始寄存器m （TSm）的通道开始触发位（TSmn）置“1”，TCRmn 寄存器就通过计数时钟从“0000H” 开始递增计数。

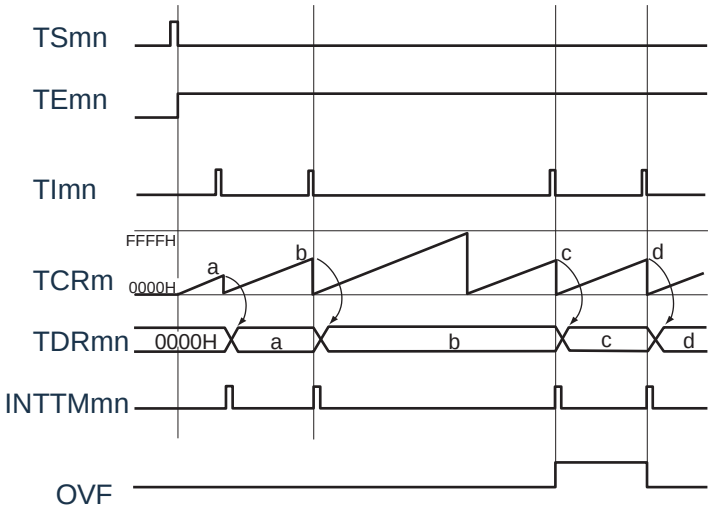
如果检测到 TImn 引脚输入的有效边沿，就将 TCRmn 寄存器的计数值传送 （捕捉）到定时器数据寄存器mn （TDRmn），同时将 TCRmn 寄存器清 “0000H”，然后输出 INTTMmn。此时，如果计数器发生上溢，就将定时器状态寄存器mn （TSRmn）的OVF 位置“1”。如果计数器没有发生上溢，就清除OVF 位。此后，继续同样的运行。

在将计数值捕捉到 TDRmn 寄存器的同时，根据在测量期间是否发生上溢，更新 TSRmn 寄存器的 OVF 位，并且能确认捕捉值的上溢状态。

即使计数器进行了2 个周期或者2 个周期以上的完整计数，也认为发生上溢而将TSRmn 寄存器的OVF 位置“1”。但是，在发生2 次或者2 次以上的上溢时，无法通过OVF 位正常测量间隔值。

将TMRmn 寄存器的STSmn2 ～STSmn0 位置“001B”，并且将TImn 的有效边沿用于开始触发和捕捉触发。

图6-26：作为输入脉冲间隔测量的运行基本时序例子（MDmn0=0）



注1：m：单元号（m= 0） n：通道号（n=0 ～3）

注2：TSmn：定时器通道开始寄存器m （TSm）的bit n TEmn ：定时器通道允许状态寄存器 m（TEm）的bit n TImn：TImn 引脚输入信号

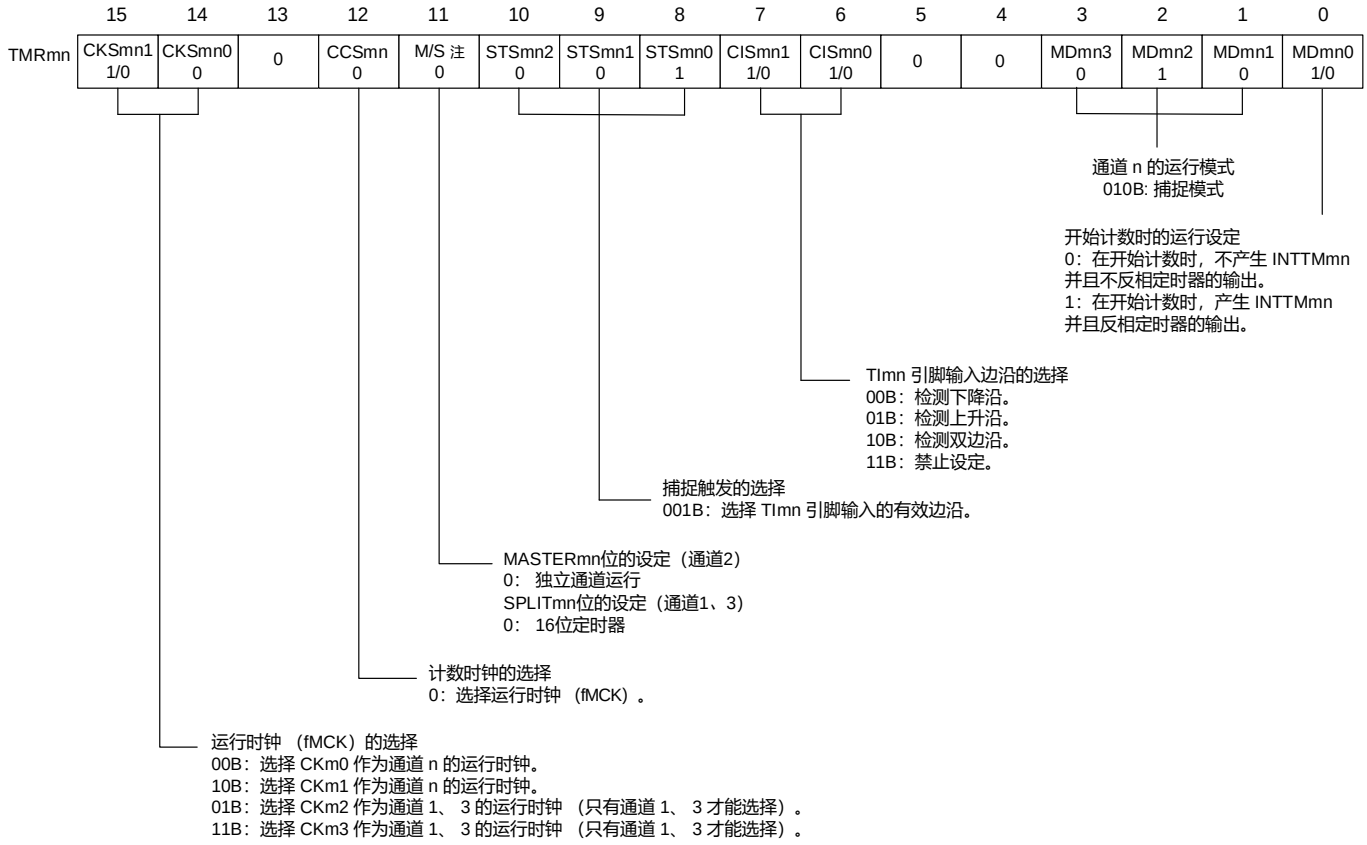
注3：TCRmn ：定时器计数寄存器mn （TCRmn）

TDRmn ：定时器数据寄存器 mn （TDRmn）

OVF：定时器状态寄存器 mn （TSRmn）的 bit0

图6-27：测量输入脉冲间隔时的寄存器设定内容例子

(a) 定时器模式寄存器mn （TMRmn）



(b) 定时器输出寄存器m （TOm）

	bit n	
TOm	TOmn 0	0：由TOmn 输出“0”。

(c) 定时器输出允许寄存器m （TOEm）

	bit n	
TOEm	TOEmn 0	0：停止由计数运行进行的 TOmn 输出。

(d) 定时器输出电平寄存器m （TOLm）

	bit n	
TOLm	TOLmn 0	0：在主控通道输出模式 （TOMmn=0）中置 “0”。

(e) 定时器输出模式寄存器m （TOMm）

	bit n	
TOMm	TOMmn 0	0：设定主控通道输出模式。

注1：TMRm2：MASTERmn 位

注2：TMRm1、 TMRm3 ：SPLITmn位

注3：TMRm0：固定为“0”。

注4：单元号（m= 0） n：通道号（n=0 ～3）

表6-31：输入脉冲间隔测量功能时的操作步骤

	软件操作	硬件状态
Timer4初始设定		定时器单元 m 的输入时钟处于停止提供状态。 （停止提供时钟，不能写各寄存器）
	将外围允许寄存器0 （PER0）的TM4mEN 位置“1”。 →	定时器单元m 的输入时钟处于提供状态，各通道处于运行停止状态。 （开始提供时钟，能写各寄存器）
	设定定时器时钟选择寄存器m （TPSm）。 确定CKm0 ~ CKm3 的时钟频率。	
通道初始设定	将噪声滤波器允许寄存器1（NFEN1）的对应位置“0”（OFF）或者“1” （ON）。 设定定时器模式寄存器mn （TMRmn） （确定通道n 的运行模式）。	通道处于运行停止状态。 （提供时钟，消耗一部分电力）
开始运行	将TSmn位置“1”。 → 因为TSmn位是触发位，所以自动返回到“0”。	TEmn 位变为 “1” 并且开始计数。 将定时器计数寄存器 mn（TCRmn）清 “0000H”。 当 TMRmn 寄存器的 MDmn0 位为 “1” 时，产生INTTMmn。
运行中	只能更改 TMRmn 寄存器的 CISmn1 位和 CISmn0 位的设定值。 能随时读 TDRmn 寄存器。 能随时读 TCRmn 寄存器。 能随时读 TSRmn 寄存器。 禁止更改 TOMmn 位、 TOLmn 位、 TOmn 位和 TOEmn 位的设定值。	计数器（TCRmn）从 “0000H” 开始递增计数，如果检测到 TImn 引脚输入的有效边沿或者将 TSmn 位置 “1”，就将计数值传送（捕捉）到定时器数据寄存器mn（TDRmn），同时将TCRmn寄存器清“0000H” 并且产生 INTTMmn。 此时，如果发生上溢，就将定时器状态寄存器 mn（TSRmn）的 OVF 位置位。如果没有发生上溢，就清除 OVF 位。 此后，重复此运行。
停止运行	将TTmn位置“1”。 → 因为TTmn位是触发位，所以自动返回到“0”。	TEmn 位变为 “0” 并且停止计数。 TCRmn 寄存器保持计数值而停止计数。 保持 TSRmn 寄存器的 OVF 位。
Timer4停止	将PER0 寄存器的TM4mEN 位置“0”。 →	定时器单元m 的输入时钟处于停止提供状态。 对全部电路和各通道的SFR 进行初始化。

重新开始运行

注：m：单元号（m= 0） n：通道号（n=0 ～3）

6.7.5 作为输入信号高低电平宽度测量的运行

注：当用作 LIN-bus 支持功能时，必须将输入切换控制寄存器 （ISC）的 bit1 （ISC1）置 “1”，并且在以下说明中，请 用RxD0 代替TImn。

能通过在 TImn 引脚输入的一个边沿开始计数并且在另一个边沿捕捉计数值，测量 TImn 的信号宽度（高低电平宽度）。TImn 的信号宽度能用以下计算式进行计算。

TImn 输入的信号宽度 = 计数时钟的周期×((10000H×TSRmn:OVF) + (TDRmn 的捕捉值 +1))

注：因为通过定时器模式寄存器 mn（TMRmn）的 CKSmn 位选择的运行时钟对 TImn 引脚输入进行采样，所以产生1 个 运行时钟的误差。

在捕捉 & 单次计数模式中，定时器计数寄存器 mn（TCRmn）用作递增计数器。 如果将定时器通道开始寄存器 m （TSm）的通道开始触发位 （TSmn）置 “1”， TEmn 位就变为 “1”，并且进入 TImn 引脚的开始边沿检测等待状态。

如果检测到 TImn 引脚输入的开始边沿（在测量高电平宽度时为 TImn 引脚输入的上升沿），就与计数时钟同步，从 “0000H” 开始递增计数。然后，如果检测到有效捕捉边沿 （在测量高电平宽度时为 TImn 引脚输入的下降沿），就在将计数值传送到定时器数据寄存器mn （TDRmn）的同时，输出INTTMmn。此时，如果计数器发生上溢，就将定时器状态寄存器 mn （TSRmn）的 OVF 位置位。如果计数器没有发生上溢，就清除 OVF 位。TCRmn 寄存器的值变为 “ 传送到 TDRmn 寄存器的值 +1” 而停止计数，并且进入 TImn 引脚的开始边沿检测等待状态。此后，继续同样的运行。

在将计数值捕捉到 TDRmn 寄存器的同时，根据在测量期间是否发生上溢，更新 TSRmn 寄存器的 OVF 位，并且能确认捕捉值的上溢状态。

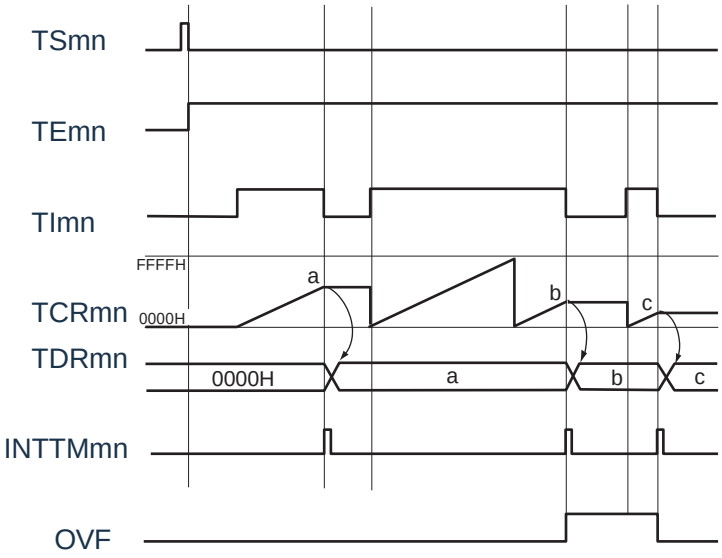
即使计数器进行了 2 个周期或者 2 个周期以上的完整计数，也认为发生上溢而将TSRmn 寄存器的OVF 位置 “1”。但是，在发生2 次或者2 次以上的上溢时，无法通过OVF 位正常测量间隔值。

能通过 TMRmn 寄存器的 CISmn1 位和 CISmn0 位来设定是测量 TImn 引脚的高电平宽度还是低电平宽度。此功能是以测量TImn 引脚的输入信号宽度为目的，因此不能在TEmn 位为“1” 的期间将TSmn 位置“1”。

TMRmn 寄存器的CISmn1、CISmn0=10B：测量低电平宽度。

TMRmn 寄存器的CISmn1、CISmn0=11B：测量高电平宽度。

图6-28：作为输入信号高低电平宽度测量的运行基本时序例子



注1：m：单元号（m= 0） n：通道号（n=0 ～3）

注2：TSmn：定时器通道开始寄存器m（TSm）的 bit n

TEmn：定时器通道允许状态寄存器 m （TEm）的 bit n

TImn：TImn 引脚输入信号

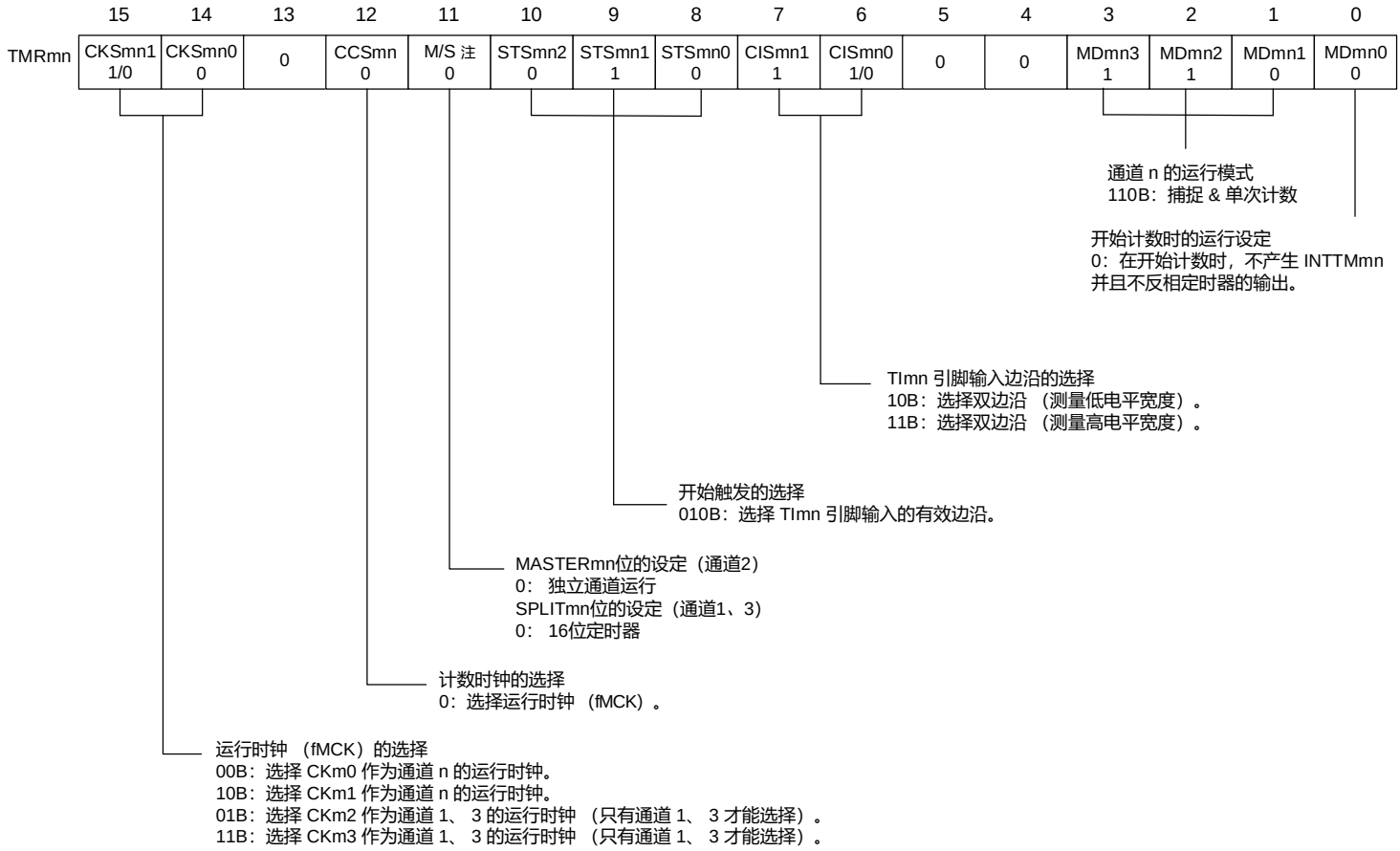
TCRmn ：定时器计数寄存器mn（TCRmn）

TDRmn ：定时器数据寄存器mn（TDRmn）

OVF：定时器状态寄存器 mnTSRmn）的bit0

图6-29：测量输入信号的高低电平宽度时的寄存器设定内容例子

(a) 定时器模式寄存器mn （TMRmn）



(b) 定时器输出寄存器m （TOm）

	bit n	
TOm	TOmn 0	0：由TOmn 输出“0”。

(c) 定时器输出允许寄存器m （TOEm）

	bit n	
TOEm	TOEmn 0	0：停止由计数运行进行的 TOmn 输出。

(d) 定时器输出电平寄存器m （TOLm）

	bit n	
TOLm	TOLmn 0	0：在主控通道输出模式 （TOMmn=0）中置 “0”。

(e) 定时器输出模式寄存器m （TOMm）

	bit n	
TOMm	TOMmn 0	0：设定主控通道输出模式。

注1：TMRm2：MASTERmn 位

注2：TMRm1、TMRm3 ：SPLITmn位

注3：TMRm0：固定为“0”。

注4：m：单元号（m= 0） n：通道号（n=0 ～3）

表6-32：输入信号高低电平宽度测量功能时的操作步骤

	软件操作	硬件状态
Timer4 初始设定		定时器单元 m 的输入时钟处于停止提供状态。 （停止提供时钟，不能写各寄存器）
	将外围允许寄存器 0（PER0）的 TM4mEN 位置“1”。 →	定时器单元 m 的输入时钟处于提供状态，各通道处于运行停止状态。 （开始提供时钟，能写各寄存器）
	设定定时器时钟选择寄存器 m（TPSm）。 确定 CKm0~CKm3 的时钟频率。	
通道初始设定	将噪声滤波器允许寄存器 1（NFEN1）的对应位置“0”（OFF）或者 “1” (ON)。 设定定时器模式寄存器 mn（TMRmn）（确定通道 n 的运行模式）。	通道处于运行停止状态。 （提供时钟，消耗一部分电力）
开始运行	将 TSmn 位置 “1” 。 因为 TSmn 位是触发位，所以自动返回到 “0” 。 →	TEmn 位变为 “1” 并且进入开始触发（检测 TImn 引脚输入的有效边沿或者将 TSmn 位置 “1” ）的检测等待状态。
	检测 TImn 引脚输入的计数开始边沿。 →	将定时器计数寄存器 mn（TCRmn）清 “0000H” 并且开始递增计数。
运行中	能任意更改 TDRmn 寄存器的设定值。 能随时读 TCRmn 寄存器。 不使用 TSRmn 寄存器。 禁止更改 TMRmn 寄存器、TOMmn 位、TOLmn 位、TOmn 位和 TOEmn 位的设定值。	在检测到 TImn 引脚的开始边沿后，计数器（TCRmn）从 “0000H” 开始递增计数。如果检测到 TImn 引脚的捕捉边沿，就将计数值传送到定时器数据寄存器 mn（TDRmn），并且产生 INTTMmn。此时，如果发生上溢，就将定时器状态寄存器 mn（TSRmn）的 OVF 位置位。如果没有发生上溢，就清除 OVF 位。TCRmn 寄存器在检测到下一个 TImn 引脚的开始边沿前停止计数。 此后，重复此运行。
停止运行	将 TTmn 位置 “1” 。 因为 TTmn 位是触发位，所以自动返回到 “0” 。 →	TEmn 位变为 “0” 并且停止计数。 TCRmn 寄存器保持计数值而停止计数。保持 TSRmn 寄存器的 OVF 位。
Timer4 停止	将 PER0 寄存器的 TM4mEN 位置 “0” 。 →	定时器单元 m 的输入时钟处于停止提供状态。 对全部电路和各通道的 SFR 进行初始化。

注：m：单元号（m= 0） n：通道号（n=0 ～3）

6.7.6 作为延迟计数器的运行

能通过 TImn 引脚输入的有效边沿检测（外部事件）开始递减计数，并且以任意的设定间隔产生 INTTMmn（定时器中断）。

在 TEmn 位为“1”的期间，能通过软件将 TSmn 位置“1”，开始递减计数，并且以任意的设定间隔产生 INTTMmn （定时器中断）。

中断产生周期能用以下计算式进行计算：

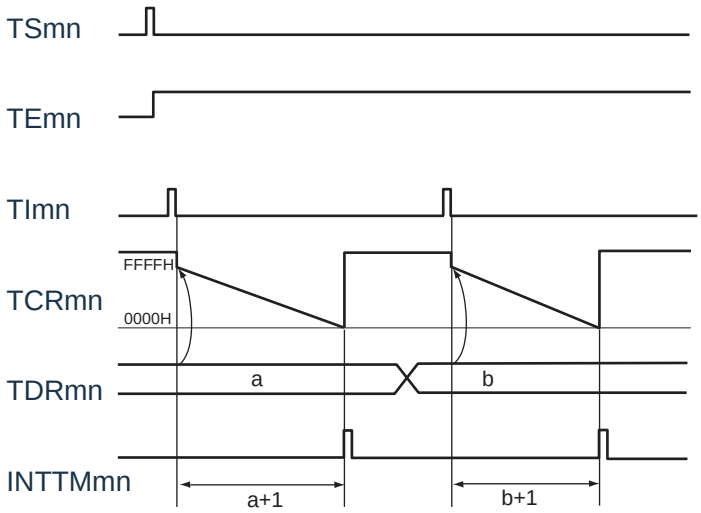
INTTMmn （定时器中断）的产生周期= 计数时钟的周期´ (TDRmn 的设定值+1)

在单次计数模式中，定时器计数寄存器mn （TCRmn）用作递减计数器。

如果将定时器通道开始寄存器 m （TSm）的通道开始触发位 （TSmn、 TSHm1、 TSHm3）置“1”，TEmn 位、TEHm1 位和 TEHm3 位就变为“1”，并且进入 TImn 引脚的有效边沿检测等待状态。通过 TImn 引脚输入的有效边沿检测，开始 TCRmn 寄存器的运行，并且装入定时器数据寄存器 mn（TDRmn）的值。TCRmn 寄存器通过计数时钟，从装入的 TDRmn 寄存器的值开始递减计数。如果 TCRmn 变为“0000H”，就输出 INTTMmn，并且在检测到下一个 TImn 引脚输入的有效边沿前停止计数。

能随时改写TDRmn 寄存器，改写的TDRmn 寄存器的值从下一个周期开始有效。

图6-30：作为延迟计数器的运行基本时序例子



注1：m：单元号（m= 0） n：通道号（n=0～3）

注2：TSmn：定时器通道开始寄存器 m （TSm）的 bit n

TEmn：定时器通道允许状态寄存器 m （TEm）的 bit n

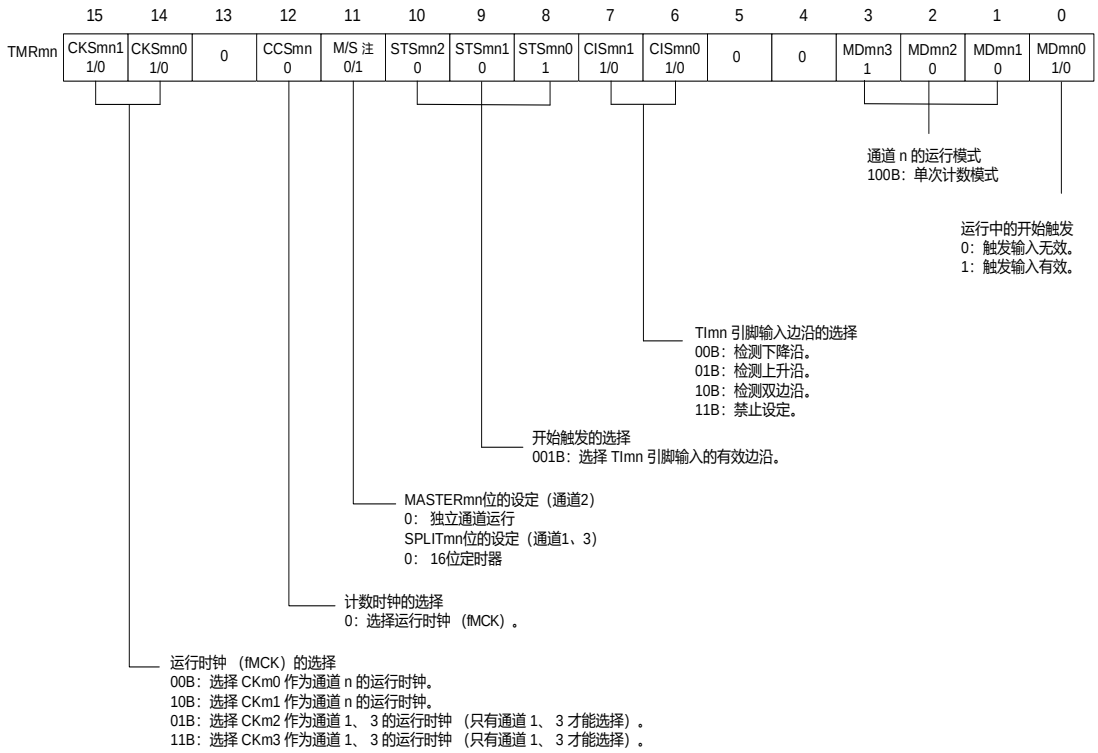
TImn：TImn 引脚输入信号

TCRmn ：定时器计数寄存器 mn （TCRmn）

TDRmn ：定时器数据寄存器 mn （TDRmn）

图6-31：延迟计数器功能时的寄存器设定内容例子

(a) 定时器模式寄存器mn （TMRmn）



(b) 定时器输出寄存器m （TOm）

	bit n	
TOm	TOmn 0	0：由TOmn 输出“0”。

(c) 定时器输出允许寄存器m （TOEm）

	bit n	
TOEm	TOEmn 0	0：停止由计数运行进行的 TOmn 输出。

(d) 定时器输出电平寄存器m （TOLm）

	bit n	
TOLm	TOLmn 0	0：在主控通道输出模式 （TOMmn=0）中置 “0”。

(e) 定时器输出模式寄存器m （TOMm）

	bit n	
TOMm	TOMmn 0	0：设定主控通道输出模式。

注1：TMRm2：MASTERmn 位

注2：TMRm1、 TMRm3 ：SPLITmn位

注3：TMRm0：固定为“0”。

注4：m：单元号（m= 0） n：通道号（n=0 ～3）

表6-33：延迟计数器功能时的操作步骤

	软件操作	硬件状态
Timer4 初始设定		定时器单元 m 的输入时钟处于停止提供状态。 （停止提供时钟，不能写各寄存器）
	将外围允许寄存器0 （PER0）的TM4mEN 位置“1”。 →	定时器单元m 的输入时钟处于提供状态，各通道处于运行停止状态。 （开始提供时钟，能写各寄存器）
	设定定时器时钟选择寄存器m （TPSm）。 确定CKm0 ~ CKm3 的时钟频率。	
通道初始设定	将噪声滤波器允许寄存器1（NFEN1）的对应位置“0”（OFF）或者“1” （ON）。 设定定时器模式寄存器mn （TMRmn）（确定通道n 的运行模式）。 给定时器数据寄存器mn （TDRmn）设定输出延迟时间。 将TOEmn 位置“0” 并且停止TOmn 的运行。	通道处于运行停止状态。 （提供时钟，消耗一部分电力）
开始运行	将TSmn位置“1”。 因为TSmn位是触发位，所以自动返回到“0”。 →	TEmn 位变为“1” 并且进入开始触发（检测TImn 引脚输入的有效边沿或者将TSmn 位置“1”）的检测等待状态。
	通过检测到下一个开始触发，开始递减计数。 . 检测TImn 引脚输入的有效边沿。 . 通过软件将TSmn 位置“1”。 →	将TDRmn 寄存器的值装入定时器计数寄存器mn（TCRmn）。
运行中	能任意更改TDRmn 寄存器的设定值。 能随时读TCRmn 寄存器。 不使用TSRmn 寄存器。	计数器（TCRmn）进行递减计数。如果TCRmn 计数到“0000H”，就产生INTTMmn，并且在检测到下一次开始触发（检测TImn 引脚输入的有效边沿或者将TSmn 位置“1”）前TCRmn 为“0000H” 而停止计数。
停止运行	将TTmn位置“1”。 因为TTmn位是触发位，所以自动返回到“0”。 →	TEmn 位变为“0” 并且停止计数。 TCRmn 寄存器保持计数值而停止计数。
Timer4 停止	将PER0 寄存器的TM4mEN 位置“0”。 →	定时器单元m 的输入时钟处于停止提供状态。 对全部电路和各通道的SFR 进行初始化。

重新开始运行

注：m：单元号（m= 0） n：通道号（n=0 ～3）

6.8 通用定时器单元的多通道联动运行功能

6.8.1 作为单触发脉冲输出功能的运行

将2个通道成对使用，能通过TImn引脚的输入生成任意延迟脉宽的单触发脉冲。 延迟和脉宽能用以下计算式进行计算：

延迟= {TDRmn （主控）的设定值+2} ´ 计数时钟周期 脉宽= {TDRmp （从属）的设定值} ´ 计数时钟周期

在单次计数模式中，主控通道运行并且对延迟进行计数。通过检测开始触发，主控通道的定时器计数寄存器mn （TCRmn）开始运行并且装入定时器数据寄存器mn （TDRmn）的值。TCRmn寄存器通过计数时钟，从装入的TDRmn寄存器的值开始递减计数。如果TCRmn变为“0000H”，就输出INTTMmn，并且在检测到下一个开始触发前停止计数。

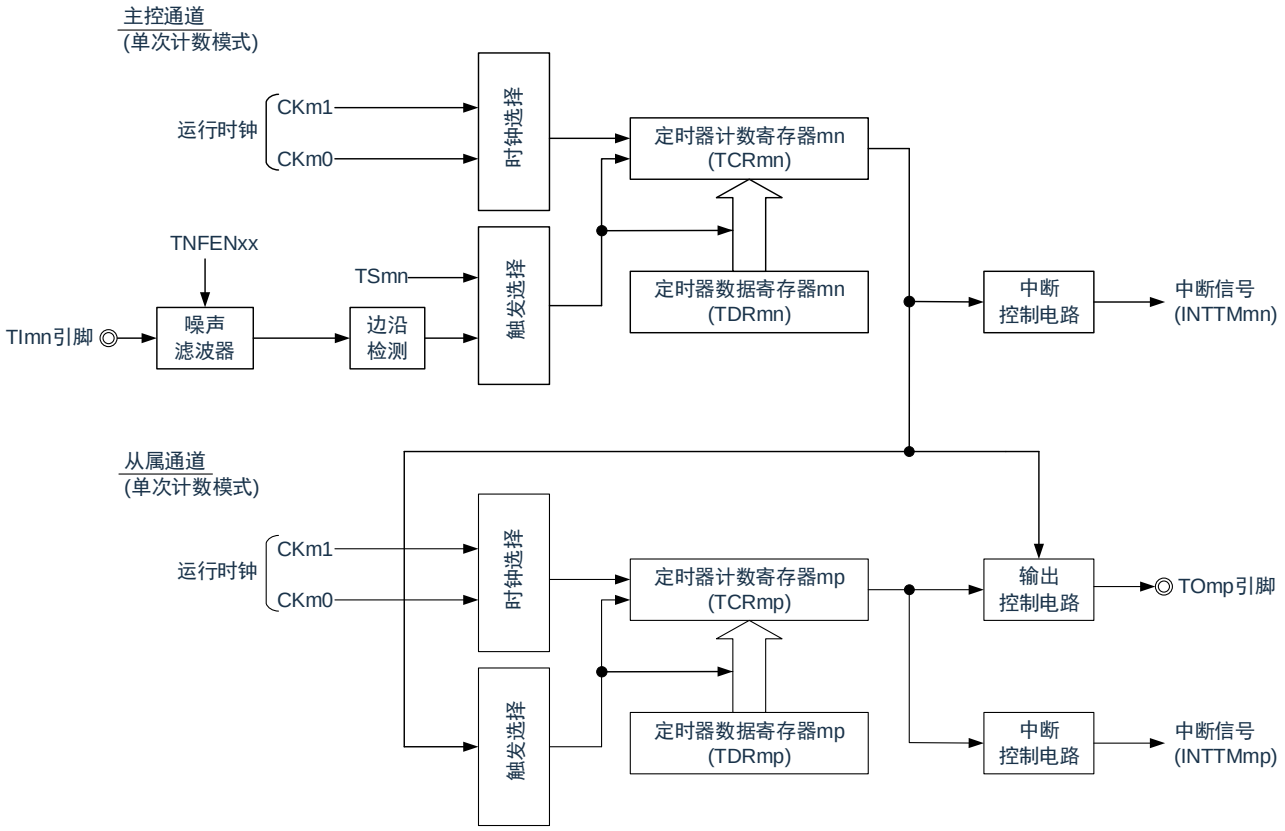
在单次计数模式中，从属通道运行并且对脉宽进行计数。将主控通道的 INTTMmn 作为开始触发，从属通道的TCRmp寄存器开始运行并且装入TDRmp寄存器的值。TCRmp寄存器通过计数时钟，从装入的TDRmp寄存器值开始递减计数。如果计数值变为 “0000H”，就输出 INTTMmp，并且在检测到下一个开始触发 （主控通道的INTTMmn）前停止计数。在从主控通道产生INTTMmn并且经过1个计数时钟后，TOmp的输出电平变为有效电平，如果TCRmp变为“0000H”，就变为无效电平。

不使用TImn引脚输入也能将软件操作（TSmn=1）作为开始触发来输出单触发脉冲。

注1：因为主控通道的TDRmn寄存器和从属通道的TDRmp寄存器的装入时序不同，所以如果在计数过程中改写 TDRmn 寄存器和 TDRmp 寄存器，就可能与装入时序发生竞争，输出不正常的波形。必须在产生INTTMmn后改写TDRmn寄存器，并且在产生INTTMmp后改写TDRmp寄存器。

注2：m：单元号（m= 0） n：主控通道号（n=0、2） p：从属通道号（n=0：p=1、2、3，n=2：p=3）

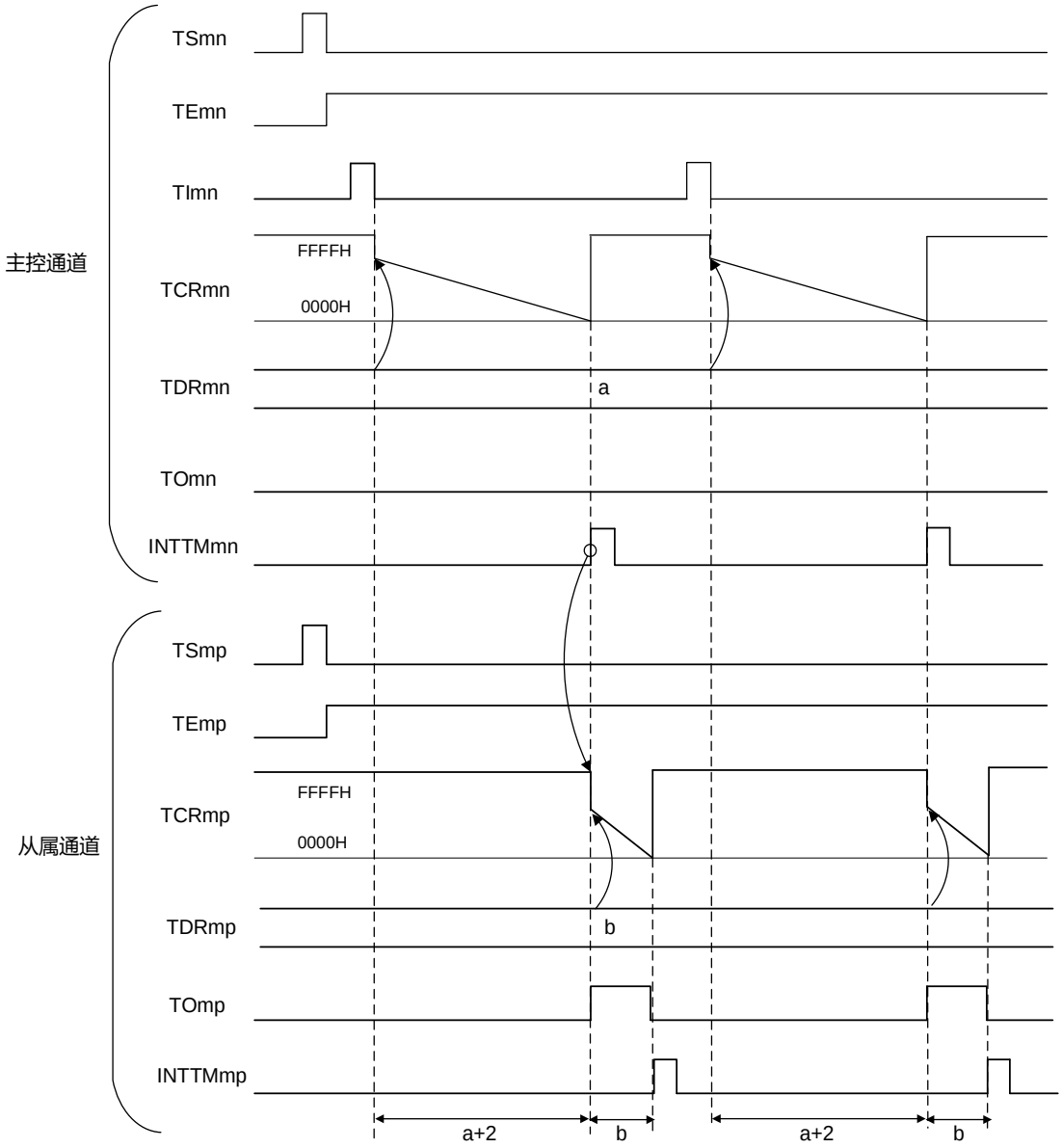
图6-32：作为单触发脉冲输出功能运行的框图



注：m：单元号（m= 0）n：主控通道号（n=0、2）

p：从属通道号（n=0：p=1、2、3， n=2：p=3）

图6-33：作为单触发脉冲输出功能的运行基本时序例子



注1：m：单元号（m= 0） n：主控通道号（n=0、2） p：从属通道号（n=0：p=1、2、3，n=2：p=3）

注2：TSmn、TSmp：定时器通道开始寄存器m （TSm）的bit n、p

TEmn、 TEmp：定时器通道允许状态寄存器 m （TEm）的 bit n、 p

TImn、 TImp：TImn 引脚和 TImp 引脚的输入信号

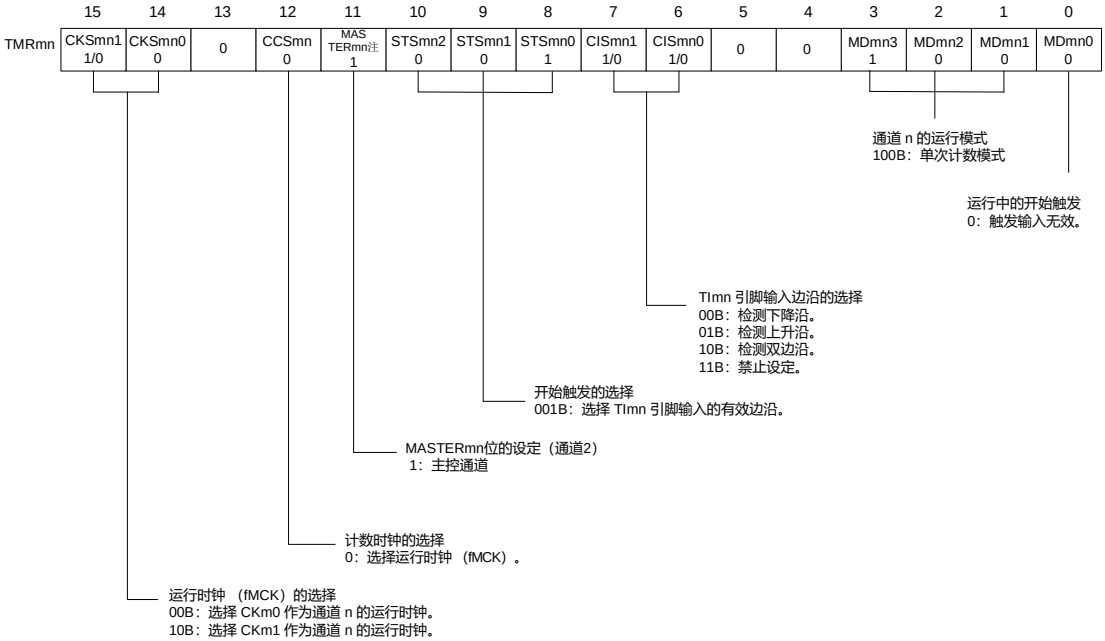
TCRmn、 TCRmp ：定时器计数寄存器 mn、 mp （TCRmn、 TCRmp）

TDRmn、 TDRmp ：定时器数据寄存器 mn、 mp （TDRmn、 TDRmp）

TOmn、 TOmp：TOmn 引脚和 TOmp 引脚的输出信号

图6-34：单触发脉冲输出功能时（主控通道）的寄存器设定内容例子

(a) 定时器模式寄存器mn （TMRmn）



(b) 定时器输出寄存器m （TOm）

	bit n	
TOm	TOmn 0	0：由TOmn 输出“0”。

(c) 定时器输出允许寄存器m （TOEm）

	bit n	
TOEm	TOEmn 0	0：停止由计数运行进行的 TOmn 输出。

(d) 定时器输出电平寄存器m （TOLm）

	bit n	
TOLm	TOLmn 0	0：在 TOMmn=0 （主控通道输出模式）时置 “0”。

(e) 定时器输出模式寄存器m （TOMm）

	bit n	
TOMm	TOMmn 0	0：设定主控通道输出模式。

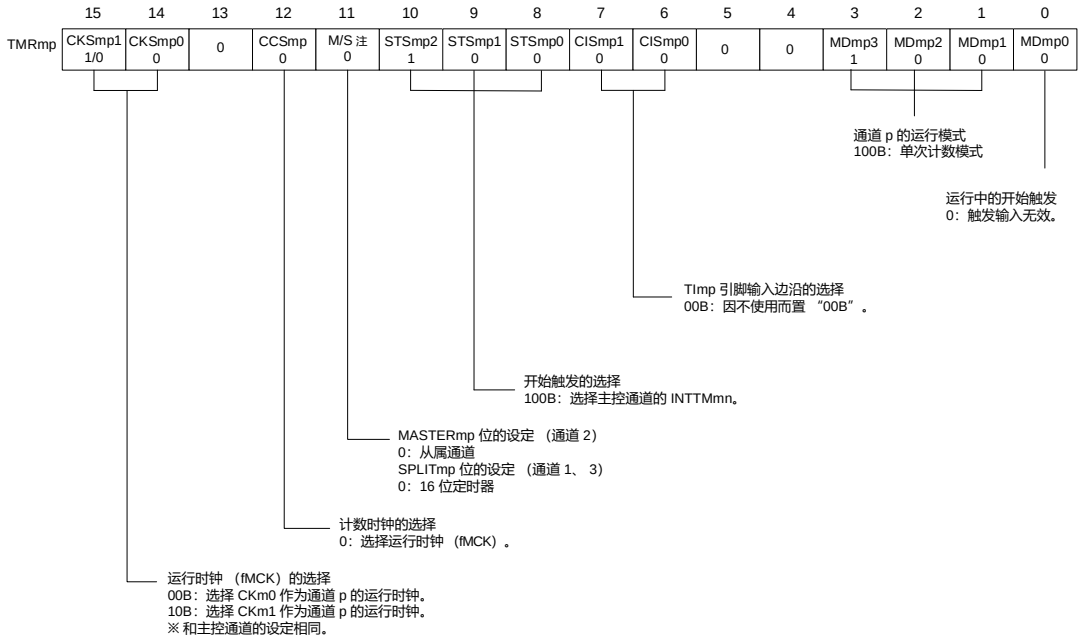
注1：m：单元号（m= 0，1）n：主控通道号（n=0、2）

注2：TMRm2：MASTERmn=1

TMRm0：固定为“0”。

图6-35：单触发脉冲输出功能时（从属通道）的寄存器设定内容例子

(a) 定时器模式寄存器mp （TMRmp）



(b) 定时器输出寄存器m （TOm）

	bit p	
TOm	TOmp 1/0	0：由TOmp 输出“0”。 1：由TOmp 输出“1”。

(c) 定时器输出允许寄存器m （TOEm）

	bit p	
TOEm	TOEmp 1/0	0：停止由计数运行进行的 TOmp 输出。 1：允许由计数运行进行的 TOmp 输出。

(d) 定时器输出电平寄存器m （TOLm）

	bit p	
TOLm	TOLmp 1/0	0：正逻辑输出 （高电平有效） 1：负逻辑输出 （低电平有效）

(e) 定时器输出模式寄存器m （TOMm）

	bit p	
TOMm	TOMmp 1	1：设定从属通道输出模式。

注1：TMRm2：MASTERmp 位

TMRm1、TMRm3：SPLITmp位

注2：m：单元号（m= 0）n：主控通道号（n=0、2）p：从属通道号（n=0：p=1、2、3，n=2：p=3）

表6-34：单触发脉冲输出功能时的操作步骤(1/2)

	软件操作	硬件状态
Timer4初始设定		定时器单元 m 的输入时钟处于停止提供状态。 （停止提供时钟，不能写各寄存器）
	将外围允许寄存器0 （PER0）的TM4mEN 位置“1”。	定时器单元m 的输入时钟处于提供状态，各通道处于运行停止状态。 （开始提供时钟，能写各寄存器）
	设定定时器时钟选择寄存器m （TPSm）。 确定CKm0 ~ CKm3 的时钟频率。	
通道初始设定	将噪声滤波器允许寄存器 1（NFEN1）的对应位置 “1”。 设定使用的 2 个通道的定时器模式寄存器 mn、 mp（TMRmn、 TMRmp）（确定通道的运行模式）。 给主控通道的定时器数据寄存器 mn （TDRmn）设定输出延迟时间，并且给从属通道的 TDRmp 寄存器设定脉宽。	通道处于运行停止状态。 （提供时钟，消耗一部分电力）
	从属通道的设定 将定时器输出模式寄存器 m （TOMm）的 TOMmp 位置 “1” （从属通道输出模式）。 设定 TOLmp 位。 设定 TOmp 位并且确定 TOmp 输出的初始电平。 将 TOEmp 位置 “1”，允许 TOmp 输出。 将端口寄存器和端口模式寄存器置 “0”。	TOmp 引脚处于 Hi-Z 输出状态。 当端口模式寄存器为输出模式并且端口寄存器为“0”时，输出 TOmp 初始设定的电平。 因为通道处于运行停止状态，所以 TOmp 不变。 TOmp 引脚输出 TOmp 设定的电平。

表6-34：单触发脉冲输出功能时的操作步骤(2/2)

重新开始运行	开始运行	将 TOEmp 位 （从属）置 “1” （只限于重新开始运行）。 将定时器通道开始寄存器 m （TSm）的 TSmn （主 控）和 TSmp （从属）位同时置 “1”。 因为 TSmn 位和 TSmp 位是触发位，所以自动返 回到 “0”。	TEmn 位和 TEmp 位都变为 “1”，主控通道进入开始触发 （检测 TImn 引脚输入的有效边沿或者将主控 通道的 TSmn 位置 “1”）的检测等待状态。 计数器还处于停止状态。
		通过检测主控通道的开始触发，开始主控通道的计 数。 • 检测 TImn 引脚输入的有效边沿。 • 通过软件将主控通道的 TSmn 位置 “1” 注。	主控通道开始计数。
	运行中	只能更改 TMRmn 寄存器的 CISmn1 位和 CISmn0 位的设定值。 禁止更改 TMRmp、 TDRmn、 TDRmp 寄存器以及 TOMmn 位、 TOMmp 位、 TOLmn 位和 TOLmp 位的设定值。 能随时读 TCRmn 寄存器和 TCRmp 寄存器。 不使用 TSRmn 寄存器和 TSRmp 寄存器。 能更改从属通道的 TOm 寄存器和 TOEm 寄存器的设定值。	主控通道通过检测开始触发 （检测 TImn 引脚输入的有效边沿或者将主控通道的 TSmn 位置 “1”），将 TDRmn 寄存器的值装入定时器计数寄存器 mn （TCRmn），并且进行递减计数。如果 TCRmn 计数到 “0000H”，就产生 INTTMmn，并且在下一次 TImn 引脚输入前停止计数。 从属通道以主控通道的INTTMmn为触发，将TDRmp 寄存器的值装入 TCRmp 寄存器并且计数器开始递减计数。在从主控通道输出 INTTMmn 并且经过 1 个计数时钟后，将TOmp的输出电平置为有效电平。 然后，如果 TCRmp 计数到 “0000H”，就在将 TOmp的输出电平置为无效电平后停止计数。 此后，重复此运行。
	停止运行	将 TTmn 位 （主控）和 TTmp 位 （从属）同时置 “1”。 因为 TTmn 位和 TTmp 位是触发位，所以自动返回到 “0”。	TEmn 位和 TEmp 位都变为 “0” 并且停止计数。 TCRmn 寄存器和 TCRmp 寄存器保持计数值而停止计数。 TOmp 输出不被初始化而保持状态。
		将从属通道的 TOEmp 位置 “0” 并且给 TOmp 位设 定值。	TOmp 引脚输出 TOmp 设定的电平。
	Timer4 停止	要保持 TOmp 引脚输出电平的情况： 在给端口寄存器设定要保持的值后将 TOmp 位置 “0”。 不需要保持 TOmp 引脚输出电平的情况： 不需要设定。	通过端口功能保持 TOmp 引脚的输出电平。
		将 PER0 寄存器的 TM4mEN 位置 “0”。	定时器单元 m 的输入时钟处于停止提供状态。 对全部电路和各通道的 SFR 进行初始化。

注 不能将从属通道的 TSmn 位置 “1”。

注：m：单元号（m= 0） n：主控通道号（n=0）p：从属通道号 q：从属通道号n＜p＜q≤3 （p和q是大于n的整数）

6.8.2 作为 PWM 功能的运行

将 2 个通道成对使用，能生成任意周期和占空比的脉冲。输出脉冲的周期和占空比能用以下计算式进行计算：

脉冲周期 = {TDRmn （主控）的设定值 +1} × 计数时钟周期
占空比 [%]= {TDRmp （从属）的设定值 } / {TDRmn （主控）的设定值 +1} ×100
0% 输出 ：TDRmp （从属）的设定值 = 0000H
100% 输出 ：TDRmp （从属）的设定值≥ {TDRmn （主控）的设定值 +1}
输出 ：TDRmp （从属）的设定值≥{TDRmn （主控）的设定值+1}

注：当TDRmp （从属）的设定值＞ {TDRmn （主控）的设定值 +1} 时，占空比超过 100%，但是为 100% 输出。

主控通道用作间隔定时器模式。如果将定时器通道开始寄存器 m （TSm）的通道开始触发位 （TSmn）置“1”，就输出中断（INTTMmn），然后将定时器数据寄存器mn （TDRmn）的设定值装入定时器计数寄存器mn （TCRmn），并且通过计数时钟进行递减计数。当计数到“0000H” 时，在输出INTTMmn 后再次将TDRmn 寄存 器的值装入 TCRmn 寄存器，并且进行递减计数。此后，在将定时器通道停止寄存器 m （TTm）的通道停止触 发位（TTmn）置“1” 前，重复此运行。

当用作 PWM 功能时，主控通道进行递减计数，在计数到 “0000H” 为止的期间为 PWM 输出（TOmp）周期。从属通道用作单次计数模式。以主控通道的 INTTMmn 为开始触发，将 TDRmp 寄存器的值装入 TCRmp 寄存器，并且进行递减计数，计数到 “0000H” 为止。当计数到 “0000H” 时，输出 INTTMmp，并且等待下一个开始触发（主控通道的INTTMmn）。

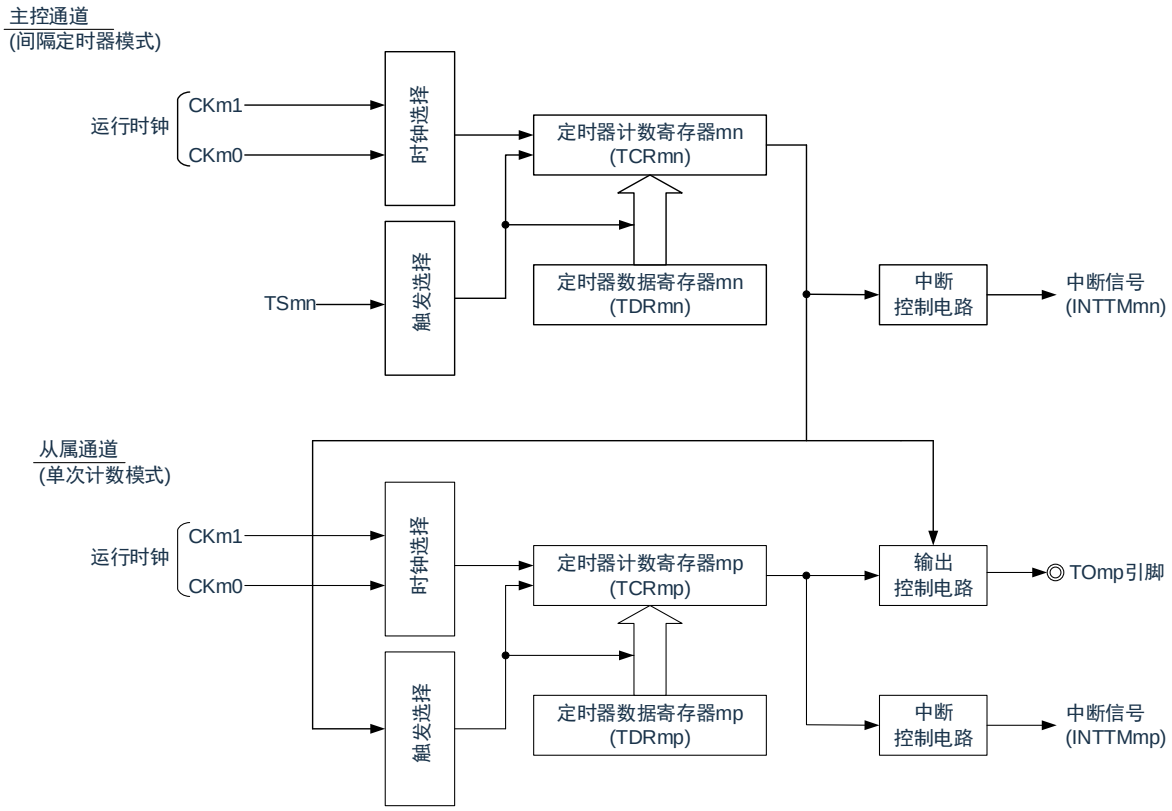
当用作 PWM 功能时，从属通道进行递减计数，在计数到 “0000H” 为止的期间为 PWM 输出 （TOmp）的占空比。

在从主控通道产生 INTTMmn 并且经过 1 个时钟后， PWM 输出（TOmp）变为有效电平，并且在从属通道的TCRmp 寄存器的值为“0000H” 时变为无效电平。

注 1：要同时改写主控通道的定时器数据寄存器 mn （TDRmn）和从属通道的 TDRmp 寄存器时，需要 2 次写存取。因为 在主控通道产生 INTTMmn 时将 TDRmn 寄存器和 TDRmp 寄存器的值装入 TCRmn 寄存器和 TCRmp 寄存器，所 以如果分别在主控通道产生 INTTMmn 前后进行改写， TOmp 引脚就不能输出期待的波形。因此，要同时改写主控 的 TDRmn 寄存器和从属的 TDRmp 寄存器时，必须在主控通道产生 INTTMmn 后立即改写这2 个寄存器。

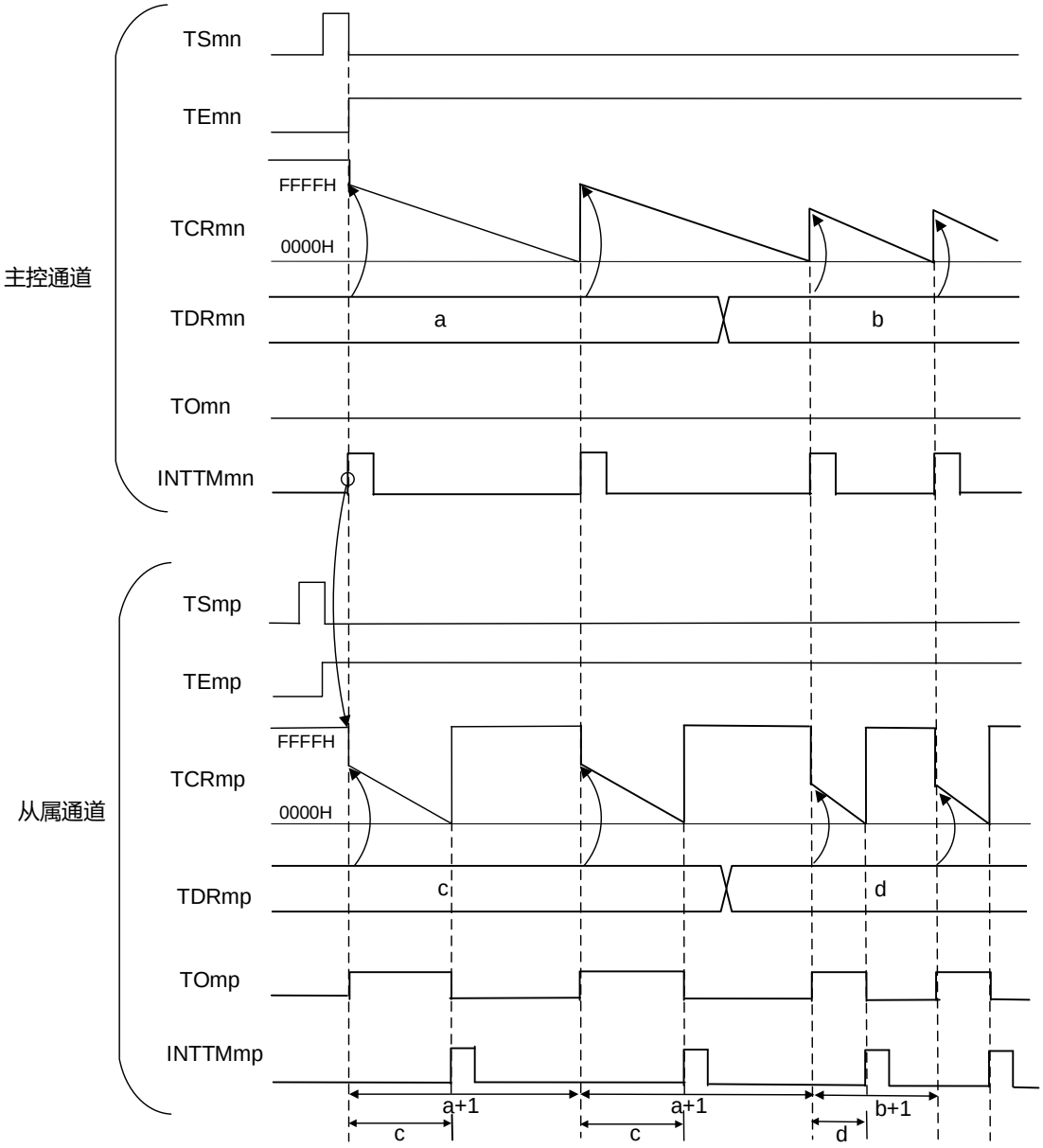
注 2：m：单元号（m= 0） n：主控通道号（n=0、2）p：从属通道号（n=0：p=1、2、3，n=2：p=3）

图6-36：作为PWM 功能运行的框图



注：m：单元号（m= 0）n：主控通道号（n=0、2） p：从属通道号（n=0：p=1、2、3， n=2：p=3）

图6-37：作为PWM 功能的运行基本时序例子



注1：m：单元号（m= 0） n：主控通道号（n=0、2） p：从属通道号（n=0：p=1、2、3，n=2：p=3）

注2：TSmn、TSmp：定时器通道开始寄存器m （TSm）的bit n、p

TEmn、TEmp：定时器通道允许状态寄存器 m （TEm）的 bit n、 p

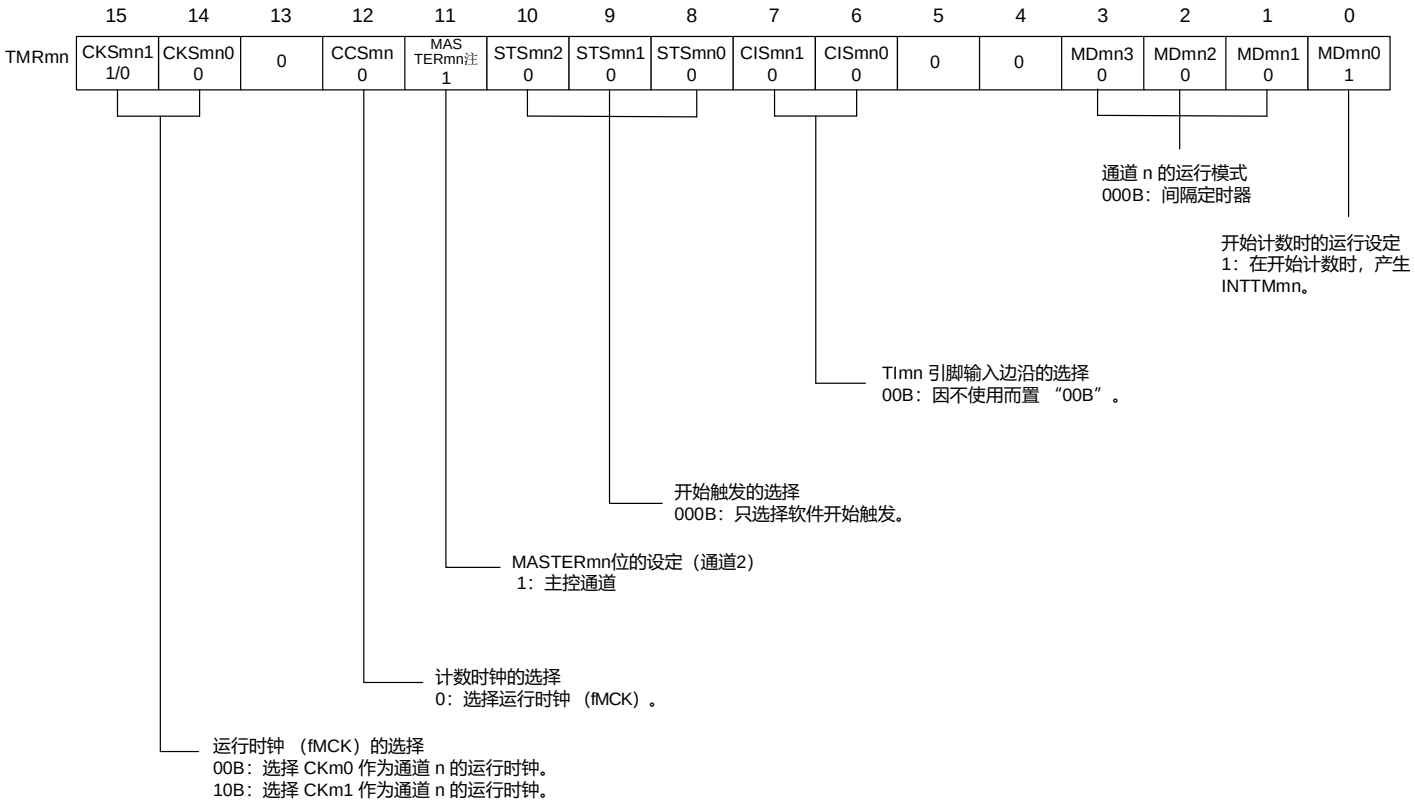
TCRmn、TCRmp：定时器计数寄存器 mn、 mp （TCRmn、 TCRmp）

TDRmn、TDRmp：定时器数据寄存器 mn、 mp （TDRmn、 TDRmp）

TOmn、TOmp：TOmn 引脚和 TOmp 引脚的输出信号

图6-38：PWM 功能时（主控通道）的寄存器设定内容例子

(a) 定时器模式寄存器mn （TMRmn）



(b) 定时器输出寄存器m （TOm）

	bit n	
TOm	TOmn 0	0：由TOmn 输出“0”。

(c) 定时器输出允许寄存器m （TOEm）

	bit n	
TOEm	TOEmn 0	0：停止由计数运行进行的 TOmn 输出。

(d) 定时器输出电平寄存器m （TOLm）

	bit n	
TOLm	TOLmn 0	0：在 TOMmn=0 （主控通道输出模式）时置 “0”。

(e) 定时器输出模式寄存器m （TOMm）

	bit n	
TOMm	TOMmn 0	0：设定主控通道输出模式。

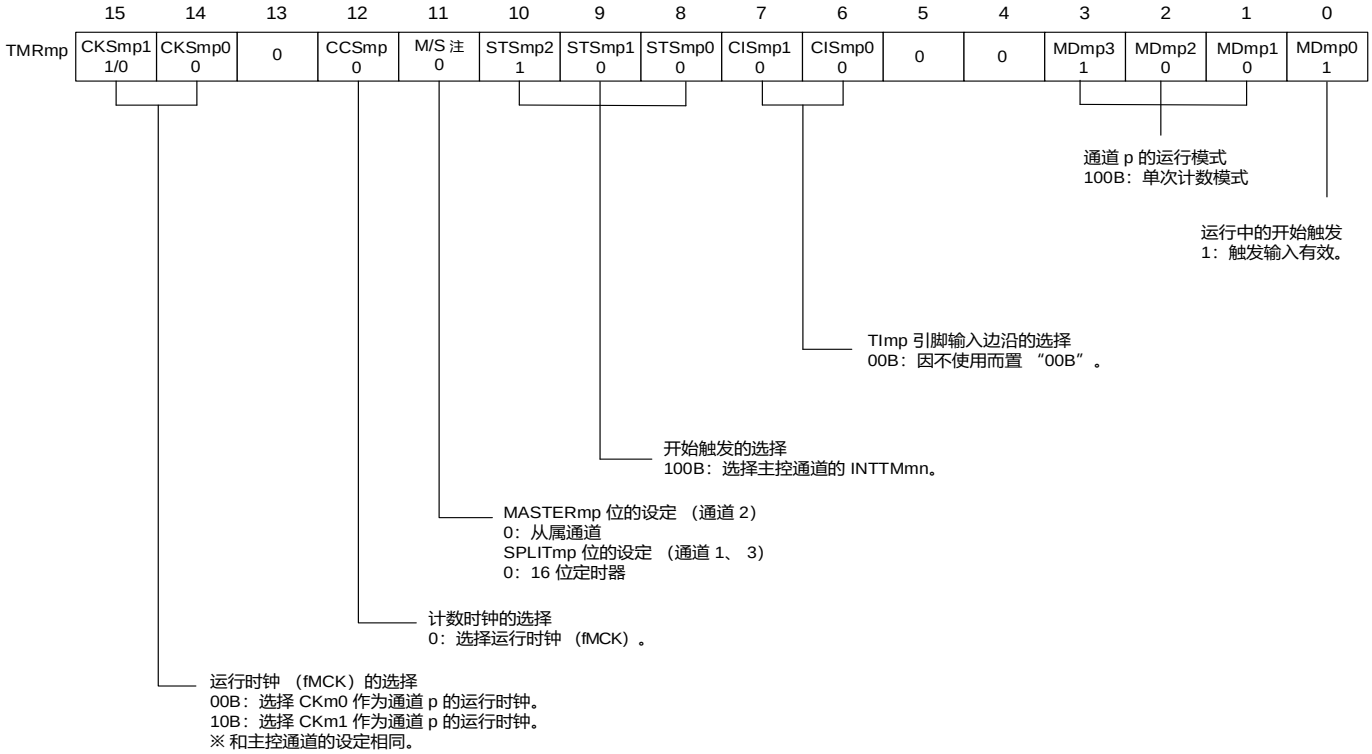
注1：m：单元号（m=0，1）n：主控通道号（n=0、2）

注2：TMRm2：MASTERmn=1

TMRm0：固定为“0”。

图6-39：PWM 功能时（从属通道）的寄存器设定内容例子

(a) 定时器模式寄存器mp （TMRmp）



(b) 定时器输出寄存器m （TOm）

	bit p	
TOm	TOmp 1/0	0：由TOmp 输出“0”。 1：由TOmp 输出“1”。

(c) 定时器输出允许寄存器m （TOEm）

	bit p	
TOEm	TOEmp 1/0	0：停止由计数运行进行的 TOmp 输出。 1：允许由计数运行进行的 TOmp 输出。

(d) 定时器输出电平寄存器m （TOLm）

	bit p	
TOLm	TOLmp 1/0	0：正逻辑输出 （高电平有效） 1：负逻辑输出 （低电平有效）

(e) 定时器输出模式寄存器m （TOMm）

	bit p	
TOMm	TOMmp 1	1：设定从属通道输出模式。

注1：TMRm2：MASTERmp 位

注2：TMRm1、 TMRm3 ：SPLITmp位

注3：m：单元号（m= 0）n：主控通道号（n=0、2） p：从属通道号（n=0：p=1、2、3，n=2：p=3）

表6-35：PWM 功能时的操作步骤(1/2)

	软件操作	硬件状态
Timer4 初始设定		定时器单元 m 的输入时钟处于停止提供状态。 （停止提供时钟，不能写各寄存器）
	将外围允许寄存器0 （PER0）的TM4mEN 位置“1”。 →	定时器单元m 的输入时钟处于提供状态，各通道处于运行停止状态。 （开始提供时钟，能写各寄存器）
	设定定时器时钟选择寄存器m （TPSm）。 确定CKm0 ~ CKm3 的时钟频率。	
通道初始设定	设定使用的 2 个通道的定时器模式寄存器 mn、 mp （TMRmn、 TMRmp）（确定通道的运行模式）。 给主控通道的定时器数据寄存器 mn （TDRmn）设 定间隔（周期）值，并且给从属通道的 TDRmp 寄 存器设定占空比的值。	通道处于运行停止状态。 （提供时钟，消耗一部分电力）
	从属通道的设定 将定时器输出模式寄存器 m （TOMm）的 TOMmp 位置 “1” （从属通道输出模式）。 设定 TOLmp 位。 设定 TOmp 位并且确定 TOmp 输出的初始电平。 将 TOEmp 位置 “1” ，允许 TOmp 输出。 将端口寄存器和端口模式寄存器置 “0” 。	TOmp 引脚处于 Hi-Z 输出状态。 当端口模式寄存器为输出模式并且端口寄存器为“0”时，输出 TOmp 初始设定的电平。 因为通道处于运行停止状态，所以 TOmp 不变。 TOmp 引脚输出 TOmp 设定的电平。

表6-35：PWM 功能时的操作步骤(2/2)

重新开始运行	开始运行	将 TOEmp 位 （从属）置 “1” （只限于重新开始运行）。 将定时器通道开始寄存器 m （TSm）的 TSmn （主 控）和 TSmp （从属）位同时置 “1”。 因为 TSmn 位和 TSmp 位是触发位，所以自动返 回到 “0”。	TEmn 位和 TEmp 位都变为 “1”。 主控通道开始计数并且产生 INTTMmn。以此为触 发，从属通道也开始计数。
	运行中	禁止更改 TMRmn 寄存器和 TMRmp 寄存器以及 TOMmn 位、 TOMmp 位、 TOLmn 位和 TOLmp 位的设定值。 能在主控通道产生 INTTMmn 后更改 TDRmn 寄存器和 TDRmp 寄存器的设定值。 能随时读 TCRmn 寄存器和 TCRmp 寄存器。 不使用 TSRmn 寄存器和 TSRmp 寄存器。	主控通道将 TDRmn 寄存器的值装入定时器计数寄存器 mn （TCRmn），并且进行递减计数。如果 TCRmn 计数到 “0000H”，就产生 INTTMmn。同时， 将 TDRmn 寄存器的值装入 TCRmn 寄存器，并且重新开始递减计数。 从属通道以主控通道的 INTTMmn 为触发，将 TDRmp 寄存器的值装入 TCRmp 寄存器，并且计数器进行递减计数。在从主控通道输出 INTTMmn 并且经过 1 个计数时钟后，将 TOmp 的输出电平置为有效电平。然后，如果 TCRmp 计数到 “0000H”，就在将 TOmp 的输出电平置为无效电平后停止计数。 此后，重复此运行。
	停止运行	将 TTmn 位 （主控）和 TTmp 位 （从属）同时置 “1”。 因为 TTmn 位和 TTmp 位是触发位，所以自动返回到 “0”。	TEmn 位和 TEmp 位都变为 “0” 并且停止计数。 TCRmn 寄存器和 TCRmp 寄存器保持计数值而停止计数。 TOmp 输出不被初始化而保持状态。
		将从属通道的 TOEmp 位置 “0” 并且给 TOmp 位设 定值。	TOmp 引脚输出 TOmp 设定的电平。
	Timer4 停止	要保持 TOmp 引脚输出电平的情况： 在给端口寄存器设定要保持的值后将 TOmp 位置 “0”。 不需要保持 TOmp 引脚输出电平的情况： 不需要设定。	通过端口功能保持 TOmp 引脚的输出电平。
		将 PER0 寄存器的 TM4mEN 位置 “0”。	定时器单元 m 的输入时钟处于停止提供状态。 对全部电路和各通道的 SFR 进行初始化。 （TOmp位变为 “0” 并且TOmp 引脚变为端口功能）

注：m：单元号（m= 0） n：主控通道号（n=0）
p：从属通道号 q：从属通道号n＜p＜q≤3 （p和q是大于n的整数）

6.8.3 作为多重 PWM 输出功能的运行

这是通过扩展PWM 功能并且使用多个从属通道进行不同占空比的多个PWM 输出的功能。 例如，当将2 个从属通道成对使用时，输出脉冲的周期和占空比能用以下计算式进行计算：

脉冲周期 = {TDRmn （主控）的设定值 +1} × 计数时钟周期 占空比 1[%] = {TDRmp （从属 1）的设定值 } / {TDRmn （主控）的设定值 +1} ×100 占空比 2[%] = {TDRmq （从属 2）的设定值 } / {TDRmn （主控）的设定值 +1} ×100

注：当TDRmp （从属1）的设定值＞{TDRmn （主控）的设定值+1} 或者{TDRmq （从属2）的设定值} ＞{TDRmn（主控）的设定值+1} 时，占空比超过100%，但是为100% 输出。

在间隔定时器模式中，主控通道的定时器计数寄存器 mn（TCRmn）运行并且对周期进行计数。 在单次计数模式中，从属通道1 的TCRmp 寄存器运行并且对占空比进行计数以及从 TOmp 引脚输出 PWM 波形。以主控通道的 INTTMmn 为开始触发，将定时器数据寄存器 mp （TDRmp）的值装入 TCRmp 寄存器并 且进行递减计数。如果 TCRmp 变为 “0000H”，就输出 INTTMmp，并且在输入下一个开始触发 （主控通道的 INTTMmn）前停止计数。在从主控通道产生 INTTMmn 并且经过 1 个计数时钟后， TOmp 的输出电平变为有效电平，如果TCRmp 变为“0000H”，就变为无效电平。

和从属通道1 的TCRmp 寄存器相同，在单次计数模式中，从属通道2 的TCRmq 寄存器运行并且对占空比进行计数以及从 TOmq 引脚输出 PWM 波形。以主控通道的 INTTMmn 为开始触发，将 TDRmq 寄存器的值装入TCRmq 寄存器并且进行递减计数。如果TCRmq 变为“0000H”，就输出INTTMmq，并且在输入下一个开始触发（主控通道的INTTMmn）前停止计数。在从主控通道产生 INTTMmn 并且经过 1 个计数时钟后，TOmq 的 输出电平变为有效电平，如果TCRmq 变为“0000H”，就变为无效电平。

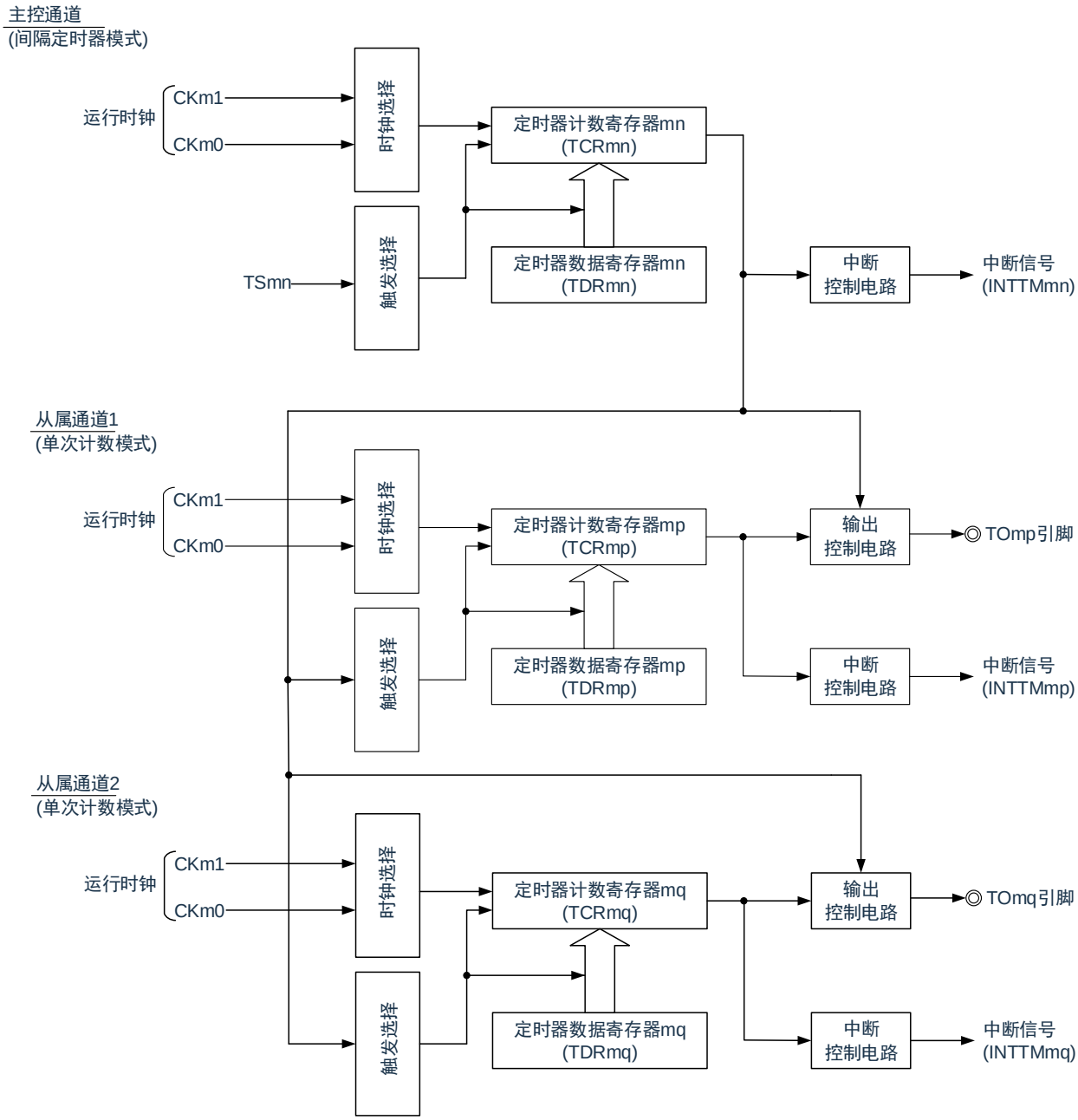
当通过如此的运行将通道0 用作主控通道时，最多能同时输出3 种PWM 信号。

注 1：要同时改写主控通道的定时器数据寄存器 mn （TDRmn）和从属通道 1 的 TDRmp 寄存器时，至少需要 2 次写存 取。因为在主控通道产生 INTTMmn 时将 TDRmn 寄存器和 TDRmp 寄存器的值装入 TCRmn 寄存器和 TCRmp 寄 存器，所以如果分别在主控通道产生 INTTMmn 前和产生后进行改写， TOmp 引脚就不能输出期待的波形。因此， 要同时改写主控的 TDRmn 寄存器和从属的 TDRmp 寄存器时，必须在主控通道产生 INTTMmn 后立即改写这 2 个 寄存器 （同样也适用于从属通道2 的TDRmq 寄存器）。

注 2：m：单元号 （m= 0） n：主控通道号 （n=0）

p：从属通道号 q：从属通道号n＜p＜q≤ 3 （p和q是大于n的整数）

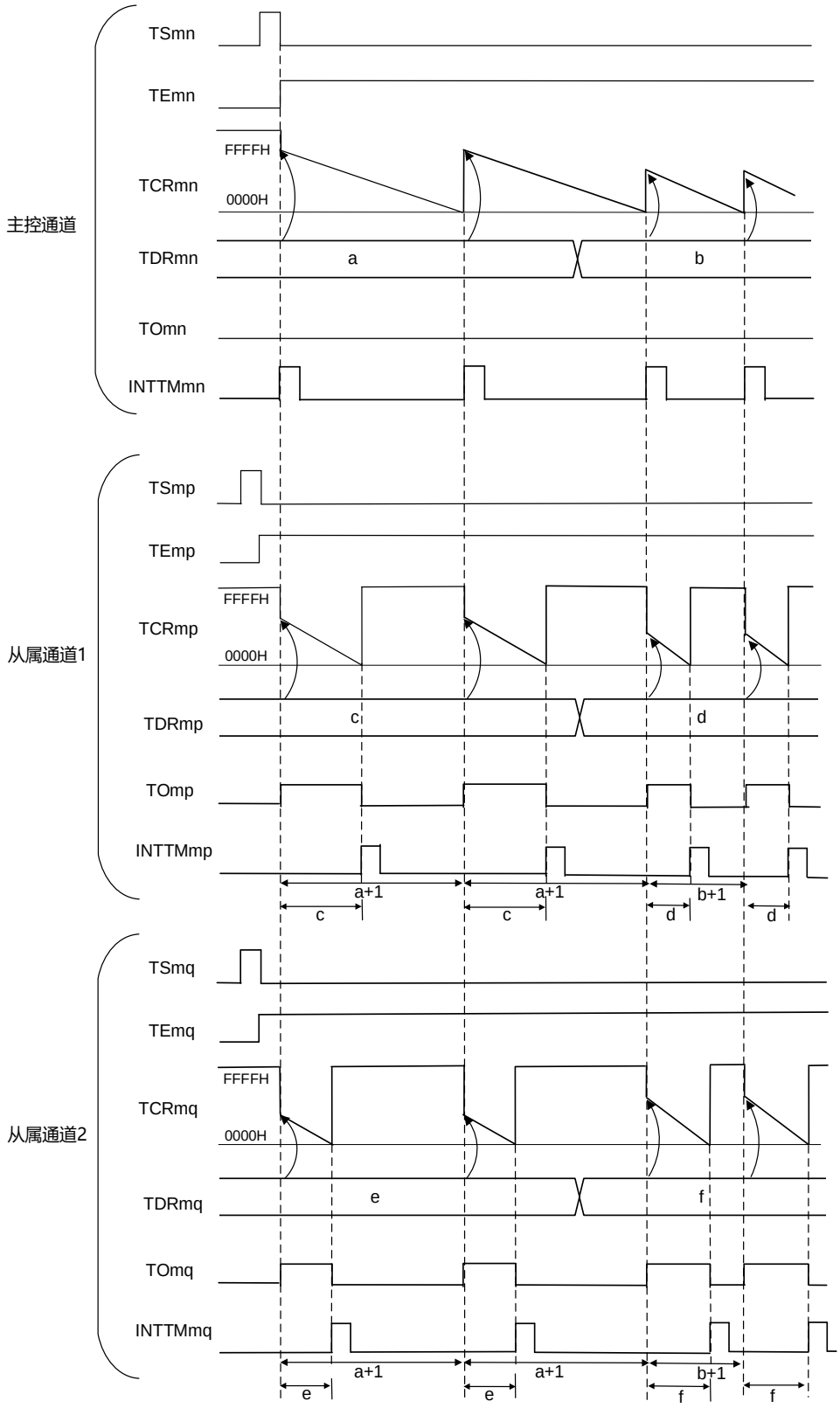
图6-40：作为多重PWM 输出功能运行的框图（输出2 种PWM 的情况）



注：m：单元号（m= 0） n：主控通道号（n=0）

p：从属通道号 q：从属通道号n＜p＜q≤3 （p和q是大于n的整数）

图6-41：作为多重PWM 输出功能的运行基本时序例子（输出2 种PWM 的情况）



注1：m：单元号（m= 0） n：主控通道号（n=0）

p：从属通道号 q：从属通道号n ＜p ＜q ≤3 （p 和q 是大于n 的整数）

注2：TSmn、TSmp、TSmq：定时器通道开始寄存器m （TSm）的bit n、p、q

TEmn、TEmp、 TEmq：定时器通道允许状态寄存器 m （TEm）的 bit n、 p、 q

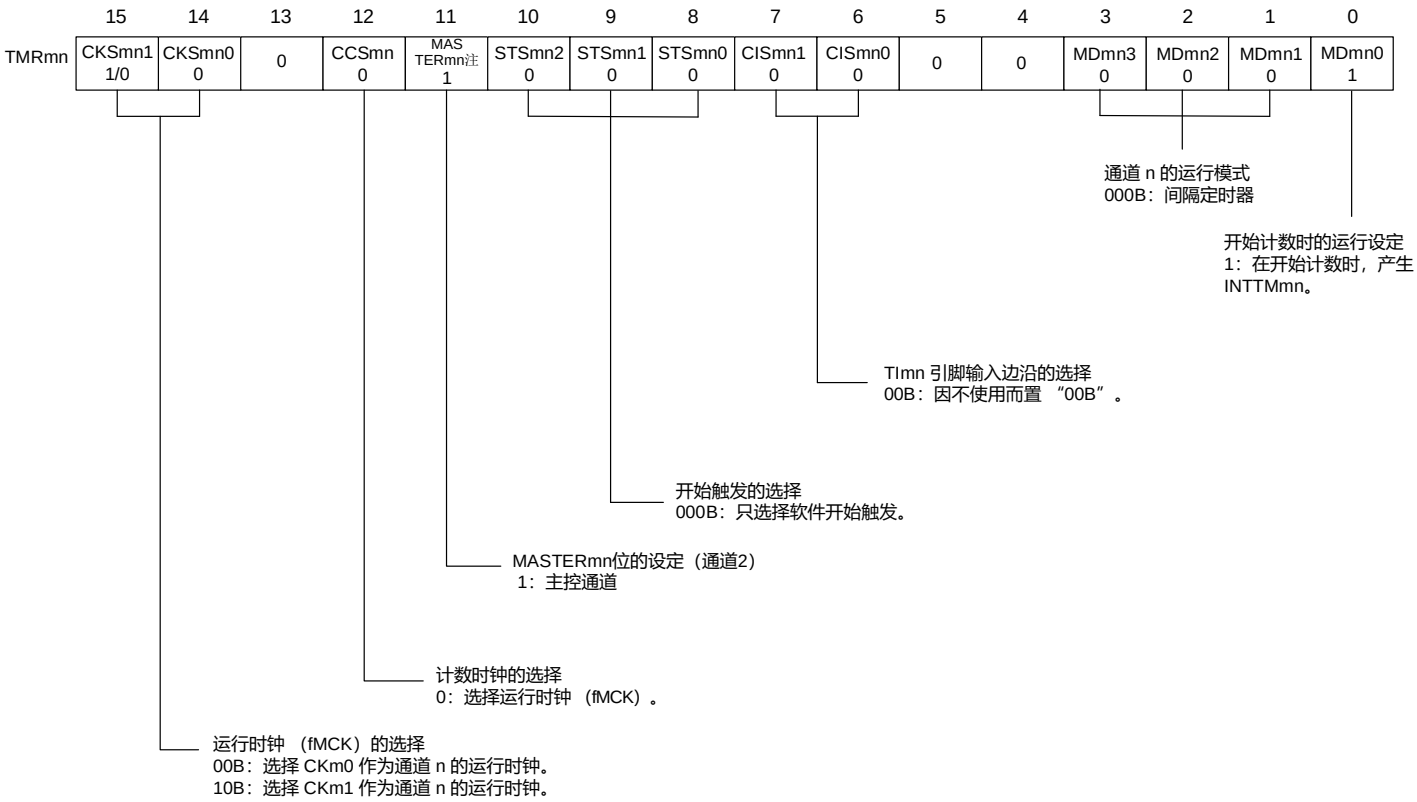
TCRmn、TCRmp、TCRmq：定时器计数寄存器 mn、 mp、 mq （TCRmn、 TCRmp、 TCRmq）

TDRmn、TDRmp、TDRmq：定时器数据寄存器 mn、 mp、 mq （TDRmn、 TDRmp、 TDRmq）

TOmn、TOmp、TOmq：TOmn、TOmp、TOmq 引脚的输出信号

图6-42：多重PWM 输出功能时（主控通道）的寄存器设定内容例子

(a) 定时器模式寄存器mn （TMRmn）



(b) 定时器输出寄存器m （TOm）

	bit n	
TOm	TOmn 0	0：由TOmn 输出“0”。

(c) 定时器输出允许寄存器m （TOEm）

	bit n	
TOEm	TOEmn 0	0：停止由计数运行进行的 TOmn 输出。

(d) 定时器输出电平寄存器m （TOLm）

	bit n	
TOLm	TOLmn 0	0：在 TOMmn=0 （主控通道输出模式）时置 “0”。

(e) 定时器输出模式寄存器m （TOMm）

	bit n	
TOMm	TOMmn 0	0：设定主控通道输出模式。

注1：m：单元号（m= 0，1） n：主控通道号（n=0）

注2：TMRm2：MASTERmn=1

TMRm0：固定为“0”。

图6-43：多重PWM 输出功能时（从属通道）的寄存器设定内容例子（输出2 种PWM 的情况）

(a) 定时器模式寄存器mp、mq （TMRmp、TMRmq）

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMRmp	CKSmp1 1/0	CKSmp0 0	0	CCSmp 0	M/S 注 0	STSmp2 1	STSmp1 0	STSmp0 0	CISmp1 0	CISmp0 0	0	0	MDmp3 1	MDmp2 0	MDmp1 0	MDmp0 1
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMRmq	CKSmq1 1/0	CKSmq0 0	0	CCSmq 0	M/S 注 0	STSmq2 1	STSmq1 0	STSmq0 0	CISmq1 0	CISmq0 0	0	0	MDmq3 1	MDmq2 0	MDmq1 0	MDmq0 1

通道 p、 q 的运行模式
100B：单次计数模式

运行中的开始触发
1：触发输入有效。

TImp、 TImq 引脚输入边沿的选择
00B：因不使用而置 “00B”。

开始触发的选择
100B：选择主控通道的 INTTMmn。

MASTERmp 位和 MASTERmq 位的设定 （通道 2）
0：从属通道
SPLITmp 位和 SPLITmq 位的设定 （通道 1、 3）
0：16 位定时器

计数时钟的选择
0：选择运行时钟 （fMCK）。

运行时钟 （fMCK）的选择
00B：选择 CKm0 作为通道 p、 q 的运行时钟。
10B：选择 CKm1 作为通道 p、 q 的运行时钟。
※ 和主控通道的设定相同。

(b) 定时器输出寄存器 m （TOm）

	bit q	bit p
TOm	TOmq 1/0	TOmp 1/0

0：由TOmp 和TOmq 输出“0”。
1：由TOmp 和TOmq 输出“1”。

(c) 定时器输出允许寄存器 m （TOEm）

	bit q	bit p
TOEm	TOEmq 1/0	TOEmp 1/0

0：停止由计数运行进行的 TOmp 和 TOmq 输出。
1：允许由计数运行进行的 TOmp 和 TOmq 输出。

(d) 定时器输出电平寄存器 m （TOLm）

	bit q	bit p
TOLm	TOLmq 1/0	TOLmp 1/0

0：正逻辑输出 （高电平有效）
1：负逻辑输出 （低电平有效）

(e) 定时器输出模式寄存器 m （TOMm）

	bit q	bit p
TOMm	TOMmq 1	TOMmp 1

1：设定从属通道输出模式。

注：m：单元号 （m= 0） n：主控通道号 （n=0）

p：从属通道号 q：从属通道号n＜p＜q≤3 （p和q是大于n的整数）

表 6-36：多重PWM 输出功能时的操作步骤（输出2 种PWM 的情况）(1/2)

	软件操作	硬件状态
Timer4 初始设定		定时器单元 m 的输入时钟处于停止提供状态。 （停止提供时钟，不能写各寄存器）
	将外围允许寄存器0 （PER0）的TM4mEN 位置“1”。 →	定时器单元m 的输入时钟处于提供状态，各通道处于运行停止状态。 （开始提供时钟，能写各寄存器）
	设定定时器时钟选择寄存器m （TPSm）。 确定CKm0 ~ CKm3 的时钟频率。	
通道初始设定	设定使用的 2 个通道的定时器模式寄存器 mn、 mp（TMRmn、 TMRmp）（确定通道的运行模式）。 给主控通道的定时器数据寄存器 mn （TDRmn）设 定间隔（周期）值，并且给从属通道的 TDRmp 寄 存器设定占空比的值。	通道处于运行停止状态。 （提供时钟，消耗一部分电力）
	从属通道的设定 将定时器输出模式寄存器m（TOMm）的TOMmp 位和TOMmq 位置“1” （从属通道输出模式）。 将TOLmp 位和TOLmq 位置“0”。 设定TOmp 位和TOmq 位，并且确定TOmp 和 TOmq 输出的初始电平。 →	TOmp 引脚处于 Hi-Z 输出状态。 当端口模式寄存器为输出模式并且端口寄存器为“0”时，输出TOmp 和TOmq 初始设定的电平。
	将TOEmp 位和TOEmq 位置“1”，允许TOmp 和 TOmq 的输出。 →	因为通道处于运行停止状态，所以TOmp 和TOmq 不变。
	将端口寄存器和端口模式寄存器置“0”。 →	TOmp 引脚和TOmq 引脚输出TOmp 和TOmq 设定的电平。

表6-36：多重PWM 输出功能时的操作步骤（输出2 种PWM 的情况）(2/2)

重新开始运行	开始运行	（只在重新开始运行时将TOEmp 位和TOEmq 位（从属）置“1”） 将定时器通道开始寄存器m（TSm）的TSmn 位（主控）、TSmp 位和TSmq 位（从属）同时置“1”。 因为TSmn 位、TSmp 位和TSmq 位是触发位，所以自动返回到“0”。	TEmn 位、TEmp 位和TEmq 位都变为“1”。 主控通道开始计数并且产生INTTMmn。以此为触发，从属通道也开始计数。
	运行中	禁止更改TMRmn、TMRmp、TMRmq 寄存器以及TOMmn 位、TOMmp 位、TOMmq 位、TOLmn 位、TOLmp、TOLmq 位的设定值。 能在主控通道产生INTTMmn 后更改TDRmn、TDRmp、TDRmq 寄存器的设定值。 能随时读TCRmn、TCRmp、TCRmq 寄存器。 不使用TSRmn、TSRmp、TSRmq 寄存器。	主控通道将TDRmn 寄存器的值装入定时器计数寄存器mn （TCRmn），并且进行递减计数。如果TCRmn 计数到“0000H”，就产生INTTMmn。同时，将TDRmn 寄存器的值装入TCRmn 寄存器，并且重新开始递减计数。 从属通道1 以主控通道的INTTMmn 信号为触发，将TDRmp 寄存器的值传送到TCRmp 寄存器，并且计数器开始递减计数。在从主控通道输出INTTMmn 并且经过1 个计数时钟后，将TOmp 的输出电平置为有效电平。然后，如果计数到“0000H”，就在将TOmp的输出电平置为无效电平后停止计数。 从属通道2 以主控通道的INTTMmn 信号为触发，将TDRmq 寄存器的值传送到TCRmq 寄存器，并且计数器开始递减计数。在从主控通道输出INTTMmn 并且经过1 个计数时钟后，将TOmq 的输出电平置为有效电平。然后，如果计数到“0000H”，就在将TOmq的输出电平置为无效电平后停止计数。 此后，重复此运行。
	停止运行	将TTmn 位（主控）、TTmp 位和TTmq 位（从属）位同时置“1”。 因为TTmn 位、TTmp 位和TTmq 位是触发位，所以自动返回到“0”。	TEmn 位、TEmp 位和TEmq 位都变为“0” 并且停止计数。 TCRmn、TCRmp、TCRmq 寄存器保持计数值而停止计数。 TOmp 和TOmq 输出不被初始化而保持状态。
		将从属通道的TOEmp 位和TOEmq 位置“0” 并且给TOmp 位和TOmq 位设定值。	TOmp 引脚和TOmq 引脚输出TOmp 和TOmq 设定的电平。
	Timer4 停止	要保持TOmp 引脚和TOmq 引脚的输出电平的情况： 在给端口寄存器设定要保持的值后将TOmp 位和TOmq 位置“0”。 不需要保持TOmp 引脚和TOmq 引脚的输出电平的情况： 不需要设定。	通过端口功能保持TOmp 引脚和TOmq 引脚的输出电平。
		将 PER0 寄存器的 TM4mEN 位置 “0”。	定时器单元m 的输入时钟处于停止提供状态。 对全部电路和各通道的SFR 进行初始化。 （TOmp 位和TOmq 位变为“0” 并且TOmp 引脚和TOmq 引脚变为端口功能）

注：m：单元号（m= 0）n：主控通道号（n=0）
p：从属通道号q：从属通道号n＜p＜q≤3 （p和q是大于n的整数）

第7章　LSITIMER 12位间隔定时器

7.1 12位间隔定时器的功能

以事先设定的任意时间间隔产生中断（INTIT），能用于从睡眠模式、深度睡眠模式和部分掉电模式的唤醒。

7.2 12位间隔定时器的结构

12位间隔定时器由以下硬件构成。

表7-1：12位间隔定时器的结构

项目	结构
计数器	12位计数器
控制寄存器	12位间隔定时器的控制寄存器（CON0）

7.3 寄存器映射

(CON0基地址 = 0x4004_4B50)

RO：只读，WO：只写，R/W：读写

寄存器	偏移量	读/写	描述	复位值
CON0	0x000	R/W	12位间隔定时器的控制寄存器	0xFFF

7.4 12位间隔定时器的控制寄存器（CON0）

这是设定12位间隔定时器的运行开始和停止以及比较值的寄存器。
通过12位存储器操作指令设定CON0寄存器。
在产生复位信号后，此寄存器的值变为“FFFH”。

位	符号	描述	复位值
15	RINTE	12位间隔定时器的运行控制 0: 停止计数器的运行（清除计数） 1: 开始计数器的运行	0
14:12	--	保留	--
11:0	ITCMP	12位间隔定时器比较值的设定	0xFFF

有关ITCMP比较值的设定以及相关中断周期计算如下表：（仅供参考）

ITCMP[11]～ITCMP[0]	12位间隔定时器比较值的设定
001H	这些位产生“计数时钟周期（ITCMP设定值+1）”的固定周期中断。
• • •	
FFFH	
000H	禁止设定。
ITCMP[11]～ITCMP[0]为“001H”或者“FFFH”时的中断周期例子 ITCMP[11]～ITCMP[0]=001H，计数时钟：Fclk=15kHz 1/15[kHz]*(1+1)=0.13333 [ms] ITCMP[11]～ITCMP[0]=FFFH，计数时钟：Fclk=15kHz 1/15[kHz]*(4095+1)=273. 06667[ms]	

注 1：要将 RINTE 位从“1”改为“0”时，必须在通过中断屏蔽标志寄存器将 INTIT 设定为禁止中断处理后进行改写。要重新开始运行（从“0”改为“1”）时，必须在清除 ITIF 标志后设定为允许中断处理。

注 2：RINTE 位的读取值在设定 RINTE 位后的 1 个计数时钟之后被反映。

注 3：在从睡眠模式转移到通常运行模式后，如果要设定 CON0 寄存器并且再次转移到睡眠模式，就必须在确认 CON0 寄存器的写入值被反映后或者在设定 CON0 寄存器后至少经过 1 个计数时钟之后再转移到睡眠模式。

注 4：要更改 ITCMP11～ITCMP0 位的设定时，必须在 RINTE 位为“0”的状态下进行。

注 5：但是，能在将 RINTE 位从“0”改为“1”或者从“1”改为“0”的同时更改 ITCMP11～ITCMP0 位的设定。

7.5 12位间隔定时器的运行

7.5.1 12位间隔定时器的运行时序

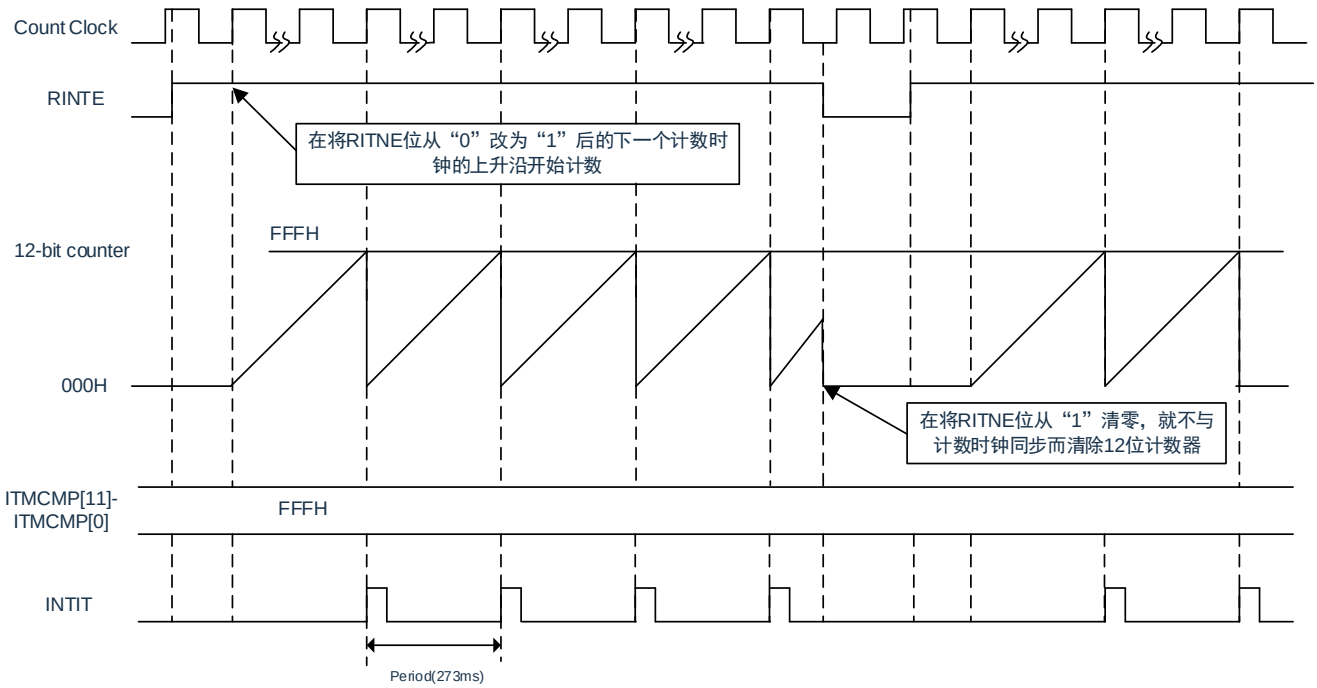
以ITCMP11～ITCMP0位设定的计数值为间隔，作为重复产生中断请求（INTIT）的12位间隔定时器运行。如果将RINTE位置“1”，12位计数器就开始计数。

当12位计数值和ITCMP11～ITCMP0位的设定值相同时，将12位计数值清“0”并且继续计数，同时产生中断请求信号（INTIT）。

12 位间隔定时器的基本运行如图 7-1 所示。

图7-1：12位间隔定时器的运行时序

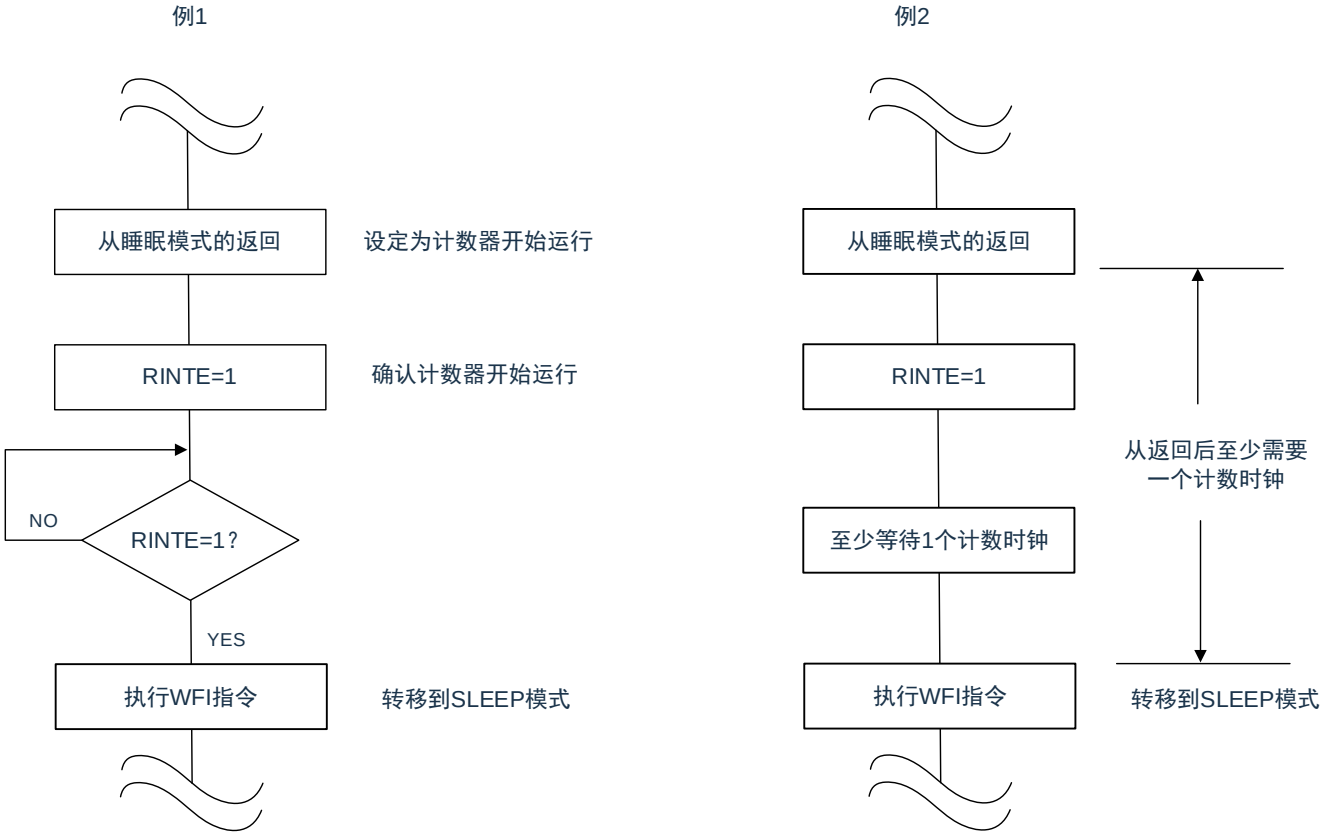
（ITCMP[11]～ITCMP[0]=FFFH，计数时钟：Fclk=15KHz）



7.5.2 从睡眠模式返回后开始计数器的运行并且再次向睡眠模式的转移

在从睡眠模式返回后，如果要将 RINTE 位置“1”并且再次转移到睡眠模式，就必须在将 RINTE 位置“1”后确认 RINTE 位的写入值被反映，或者在返回后至少经过 1 个计数时钟的时间之后再转移到睡眠模式。

- 在将 RINTE 位置“1”后，通过轮询确认 RINTE 位变为“1”，然后转移到睡眠模式（参照下图的例 1）。
- 在将 RINTE 位置“1”后至少经过 1 个计数时钟的时间之后转移到睡眠模式（参照下图的例 2）。



第8章 时钟输出/蜂鸣器输出控制电路

8.1 时钟输出/蜂鸣器输出控制电路的功能

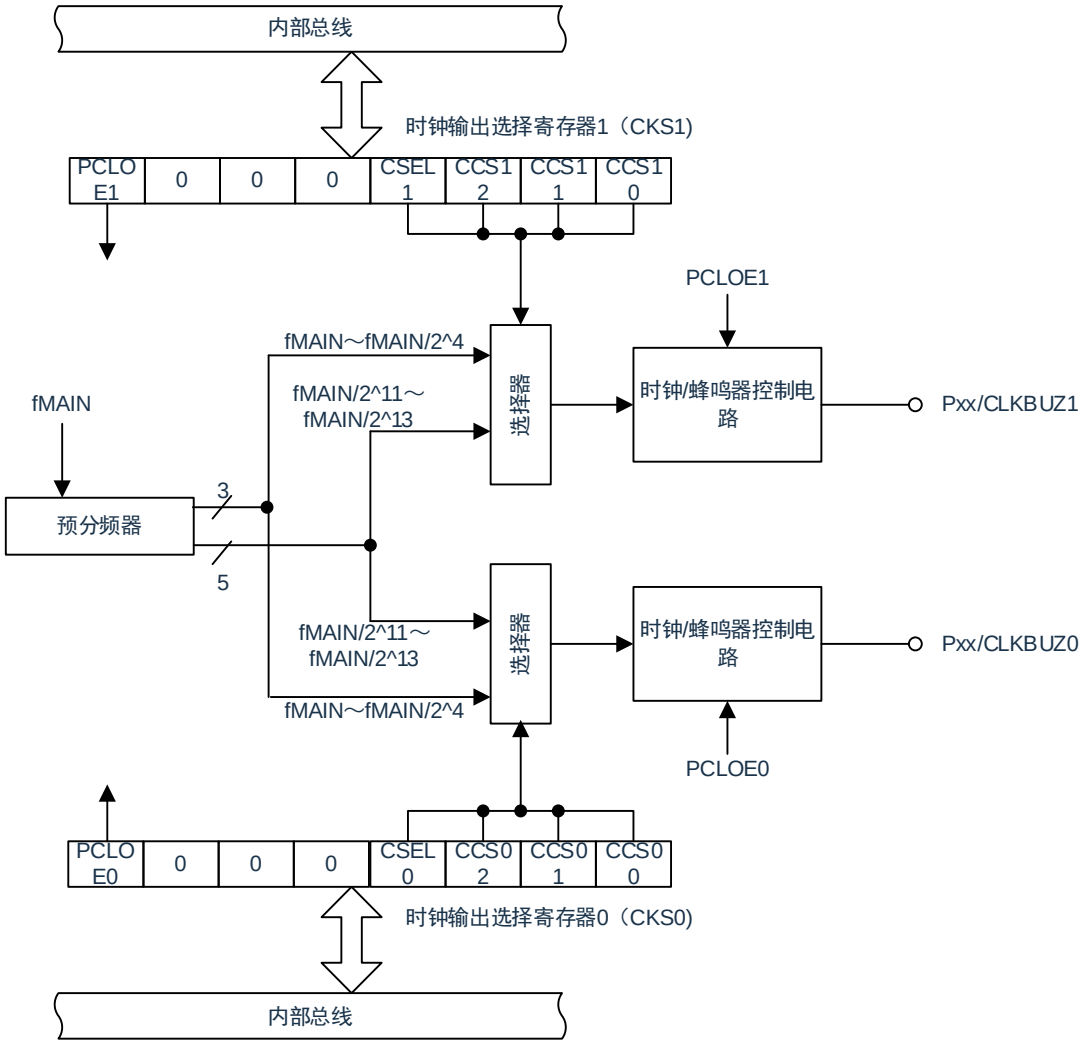
时钟输出是输出提供给外围 IC 时钟的功能，蜂鸣器输出是输出蜂鸣器频率方波的功能。

本产品有两个时钟输出/蜂鸣器输出引脚，其中 CLKBUZ0 能从 P20、P25、P00、P06、P22 中选择引脚用作时钟输出或者蜂鸣器输出，CLKBUZ1 能从 P21、P25、P06、P22 中选择引脚用作时钟输出或者蜂鸣器输出。

CLKBUZn 引脚输出由时钟输出选择寄存器 n（CKSn）选择的时钟。

时钟输出/蜂鸣器输出控制电路的框图如图 8-1 所示。

图8-1：时钟输出/蜂鸣器输出控制电路的框图



注：有关能从CLKBUZ0引脚和CLKBUZ1引脚输出的频率，请参照“数据手册的AC特性”。

8.2 时钟输出/蜂鸣器输出控制电路的结构

时钟输出/蜂鸣器输出控制电路由以下硬件构成。

表8-1：时钟输出/蜂鸣器输出控制电路的结构

项目	结构
控制寄存器	时钟输出选择寄存器n（CKSn） 端口模式控制寄存器（PMCxx）、端口模式寄存器（PMxx）、端口复用控制寄存器（PxxCFG）

8.3 寄存器映射

(CKS0/1 基地址 = 0x4004_0FA5)

RO：只读，WO：只写，R/W：读写

寄存器	偏移量	读/写	描述	复位值
CKS0	0x000	R/W	时钟输出选择寄存器0	0x0
CKS1	0x001	R/W	时钟输出选择寄存器1	0x0

8.3.1 时钟输出选择寄存器（CKS0）

这是允许或者禁止时钟输出引脚或者蜂鸣器频率输出引脚（CLKBUZn）的输出以及设定输出时钟的寄存器。

通过CKSn寄存器选择CLKBUZn引脚输出的时钟。通过8位存储器操作指令设定CKSn寄存器。在产生复位信号后，此寄存器的值变为“00H”。

位	符号	描述	复位值
7	PCLOE0	CLKBUZ0引脚输出允许/禁止的指定 0: 禁止输出（默认值） 1: 允许输出	0
6:4	--	保留	--
3	CSEL0	CLKBUZ0引脚输出时钟的选择	0
2:0	CCS0	CLKBUZ0引脚输出时钟的选择	0x0

具体CLKBUZn引脚输出时钟的选择如下表：

CSEL0	CCS0[2]	CCS0[1]	CCS0[0]	CLKBUZ0引脚输出时钟的选择
0	0	0	0	F_{MAIN}
0	0	0	1	$F_{MAIN}/2$
0	0	1	0	$F_{MAIN}/2^2$
0	0	1	1	$F_{MAIN}/2^3$
0	1	0	0	$F_{MAIN}/2^4$
0	1	0	1	$F_{MAIN}/2^{11}$
0	1	1	0	$F_{MAIN}/2^{12}$
0	1	1	1	$F_{MAIN}/2^{13}$

注1：必须在16MHz以内的范围内使用输出时钟。详细内容请参照“数据手册的AC特性”。

注2：钟的切换必须在设定为禁止输出（PCLOE0=0)后进行。

注3：主系统时钟（CSEL0=0）时，如果要转移到深度睡眠模式，就必须在执行WFI指令前将PCLOE0置“0”。

注4：F_{MAIN}：主系统时钟频率

8.3.2 时钟输出选择寄存器（CKS1）

这是允许或者禁止时钟输出引脚或者蜂鸣器频率输出引脚（CLKBUZ1）的输出以及设定输出时钟的寄存器。

通过 CKS1 寄存器选择 CLKBUZ1 引脚输出的时钟。通过 32 位存储器操作指令设定 CKS1 寄存器。在产生复位信号后，此寄存器的值变为“00H”。

位	符号	描述	复位值
7	PCLOE1	CLKBUZ1引脚输出允许/禁止的指定 0: 禁止输出（默认值） 1: 允许输出	0
6:4	--	保留	--
3	CSEL1	CLKBUZ1引脚输出时钟的选择	0
2:0	CCS1	CLKBUZ1引脚输出时钟的选择	0x0

具体CLKBUZn引脚输出时钟的选择如下表：

CSEL1	CCS1[2]	CCS1[1]	CCS1[0]	CLKBUZ1引脚输出时钟的选择
0	0	0	0	F_{MAIN}
0	0	0	1	$F_{MAIN}/2$
0	0	1	0	$F_{MAIN}/2^2$
0	0	1	1	$F_{MAIN}/2^3$
0	1	0	0	$F_{MAIN}/2^4$
0	1	0	1	$F_{MAIN}/2^{11}$
0	1	1	0	$F_{MAIN}/2^{12}$
0	1	1	1	$F_{MAIN}/2^{13}$

注1：16MHz以内的范围内使用输出时钟。详细内容请参照“数据手册的AC特性”。

注2：出时钟的切换必须在设定为禁止输出（PCLOE1=0)后进行。

注3：择主系统时钟（CSEL1=0）时，如果要转移到深度睡眠模式，就必须在执行WFI指令前将PCLOE1置“0”。

注4：F_{MAIN}：主系统时钟频率

8.4 配置时钟输出/蜂鸣器输出引脚端口功能的寄存器

本产品有两个时钟输出/蜂鸣器输出引脚，其中 CLKBUZ0 能从 P20、P25、P00、P06、P22 中选择引脚用作时钟输出或者蜂鸣器输出，CLKBUZ1 能从 P21、P25、P06、P22 中选择引脚用作时钟输出或者蜂鸣器输出。

使用时钟输出/蜂鸣器输出功能时，必须设定端口复用功能配置寄存器（PxxCFG）,端口寄存器（Pxx），端口模式寄存器（PMxx）和端口模式控制寄存器（PMCxx）。详细内容请参照“第 3 章 引脚功能”。

被配置为时钟输出/蜂鸣器输出引脚的复用端口，其对应的端口寄存器（Pxx），端口模式寄存器（PMxx）的位和端口模式控制寄存器（PMCxx）的位必须置“0”。

（例）将 P20 用作时钟输出/蜂鸣器输出（CLKBUZ0）的情况：

将端口寄存器 2 的 P20 位置“0”。

将端口模式寄存器 2 的 PM20 位置“0”。

将端口模式控制寄存器 2 的 PMC20 位置“0”。

将端口复用功能配置寄存器 P20CFG 置“0x01”。

（例）将 P21 用作时钟输出/蜂鸣器输出（CLKBUZ1）的情况：

将端口寄存器 2 的 P21 位置“0”。

将端口模式寄存器 2 的 PM21 位置“0”。

将端口模式控制寄存器 2 的 PMC21 位置“0”。

将端口复用功能配置寄存器 P21CFG 置“0x01”。

8.5 时钟输出/蜂鸣器输出控制电路的运行

能用 1 个引脚选择用作时钟输出或者蜂鸣器输出。

CLKBUZ0 引脚输出由时钟输出选择寄存器 0（CKS0）选择的时钟/蜂鸣器。

CLKBUZ1 引脚输出由时钟输出选择寄存器 1（CKS1）选择的时钟/蜂鸣器。

8.5.1 输出引脚的运行

CLKBUZn 引脚按照以下步骤进行输出：

1）设定端口复用功能配置寄存器（PmnCFG）,将用作 CLKBUZ0 引脚的端口对应的端口寄存器（Pxx），端口模式寄存器（PMxx）和端口模式控制寄存器（PMCxx）的位置“0”。

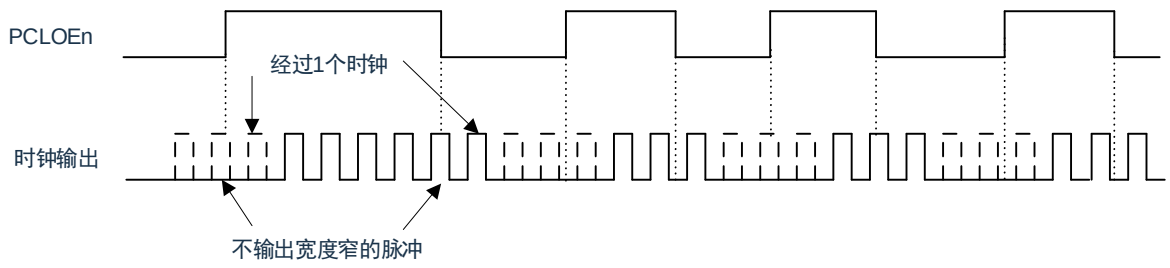
2）通过 CLKBUZn 引脚的时钟输出选择寄存器（CKSn）的 bit0～3（CCSn0～CCSn2、CSELn）选择输出频率（输出为禁止状态）。

3）将 CKSn 寄存器的 bit7（PCLOEn）置“1”，允许时钟/蜂鸣器的输出。

注 1：用作时钟输出时的控制电路在允许或者禁止时钟输出（PCLOEn 位）后的 1 个时钟之后，开始或者停止时钟输出。此时不输出宽度窄的脉冲。通过 PCLOEn 位允许或者停止输出以及时钟输出的时序如图 8-2 所示。

注 2：n=0、1

图8-2：CLKBUZn引脚的时钟输出时序



8.6 时钟输出/蜂鸣器输出控制电路的注意事项

当选择主系统时钟作为 CLKBUZn 输出（CSELn=0）时，如果在设定停止输出（PCLOEn=0）后的 1.5 个 CLKBUZn 引脚的输出时钟内转移到深度睡眠模式，CLKBUZn 的输出宽度就变窄。

第9章 看门狗定时器

9.1 看门狗定时器的功能

看门狗定时器通过选项字节（000C0H）设定计数运行。看门狗定时器以低速内部振荡器时钟（F_{IL}）运行。看门狗定时器用于检测程序失控。在检测到程序失控时，产生内部复位信号。

下述情况判断为程序失控。

(1) 当看门狗定时器的计数器发生上溢时

(2) 当给 WDTE 寄存器写“ACH”以外的数据时

(3) 在窗口关闭期间给 WDTE 寄存器写数据时

当因看门狗定时器而发生复位时，将复位控制标志寄存器（RESF）的 bit4（WDTRF）置“1”。有关 RESF 寄存器的详细内容，请参照“第 24 章 复位功能”。 当达到上溢时间的 75%+1/2F_{IL} 时，能产生间隔中断。

9.2 看门狗定时器的结构

看门狗定时器由以下硬件构成。

表9-1：看门狗定时器的结构

项目	结构
计数器	内部计数器（17 位）
控制寄存器	看门狗定时器的允许寄存器（WDTE）

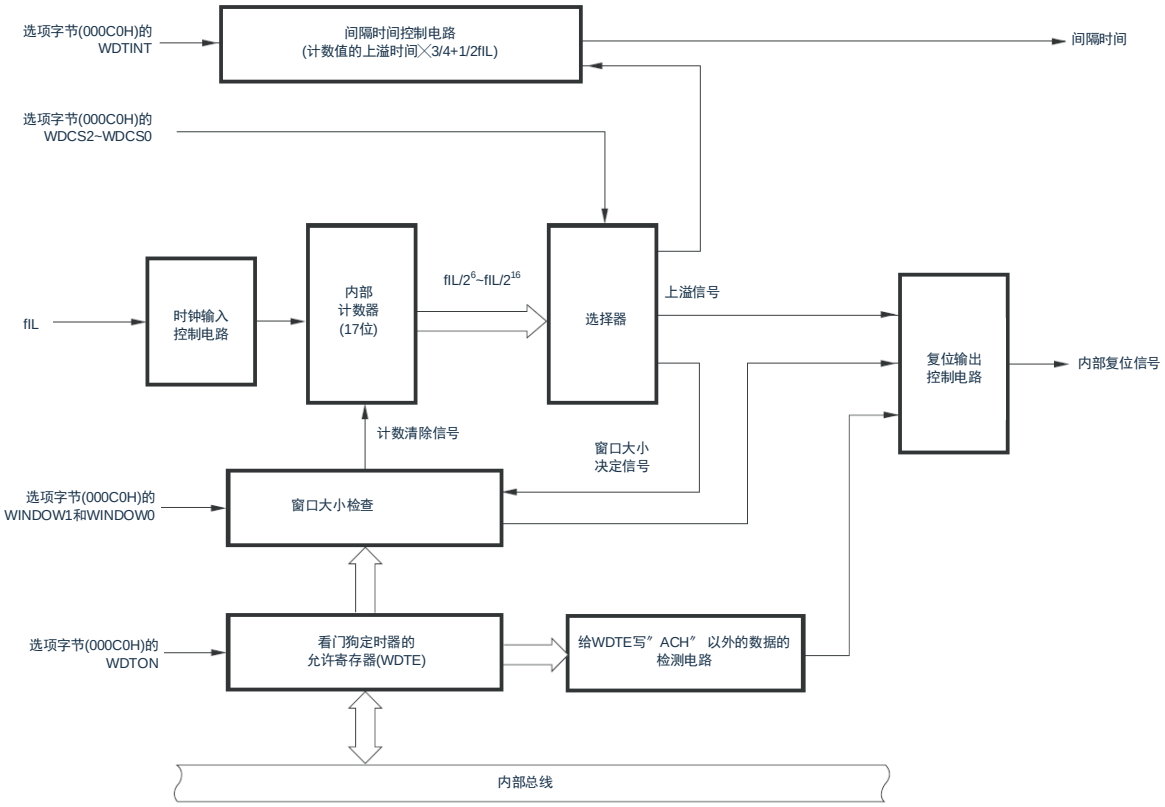
通过选项字节控制计数器的运行以及设定上溢时间、窗口打开期间和间隔中断。

表9-2：选项字节和看门狗定时器的设定内容

看门狗定时器的设定内容	选项字节（000C0H）
看门狗定时器的间隔中断的设定	bit7（WDTINT）
窗口打开期间的设定	bit6和bit5（WINDOW1、WINDOW0）
看门狗定时器的计数器运行控制	bit4（WDTON）
看门狗定时器的上溢时间的设定	bit3～1（WDCS2～WDCS0）
看门狗定时器的计数器运行控制（睡眠时）	bit0（WDSTBYON）

注：有关选项字节，请参照“第29章 选项字节”。

图9-1：看门狗定时器的框图



注：F_{IL}：低速内部振荡器的时钟频率

9.3 寄存器映射

(WDTE 基地址 = 0x4002_1001)

寄存器	偏移量	读/写	描述	复位值
WDTE	0x000	R/W	看门狗定时器的允许寄存器	0x1A/0x9A

(LOCKCTL基地址 = 0x4002_0405)

RO：只读，WO：只写，R/W：读写

寄存器	偏移量	读/写	描述	复位值
LOCKCTL	0x000	R/W	控制寄存器	0x1

(PRCR基地址 = 0x4002_0406)

RO：只读，WO：只写，R/W：读写

寄存器	偏移量	读/写	描述	复位值
PRCR	0x000	R/W	保护寄存器	0x0

9.3.1 看门狗定时器的允许寄存器（WDTE）

通过给 WDTE 寄存器写“ACH”，清除看门狗定时器的计数器并且重新开始计数。通过 8 位存储器操作指令设定 WDTE 寄存器。在产生复位信号后，此寄存器的值变为“9AH”或者“1AH”[注]。

位	符号	描述	复位值
7:0	WDTE	写0xAC清除看门狗定时器的计数器并且重新开始计数	0x1A/0x9A

注：WDTE寄存器的复位值因选项字节（000C0H）的WDTON位的设定值而不同。要使看门狗定时器运行时，必须将WDTON位置“1”。

WDTON位的设定值	WDTE寄存器的复位值
0（禁止看门狗定时器的计数运行）	1AH
1（允许看门狗定时器的计数运行）	9AH

注1：当给WDTE寄存器写“ACH”以外的值时，产生内部复位信号。

注2：WDTE寄存器的读取值为“9AH/1AH”（和写入值（“ACH”）不同）。

9.3.2 LOCKUP控制寄存器（LOCKCTL）

LOCKCTL 寄存器是 Cortex-M0+ LockUp 功能是否引起看门狗定时器运行的配置寄存器，PRCR 是其写保护寄存器。

通过 8 位存储器操作指令设置 LOCKCTL 寄存器。

在产生复位信号后，LOCKCTL 寄存器的值变为“01H”。

位	符号	描述	复位值
7:1	-	保留	-
0	lockup_rst	LOCKUP功能的配置 0: LOCKUP不导致WDT复位 1: LOCKUP导致WDT复位	1

9.3.3 保护寄存器（PRCR）

LOCKCTL 寄存器是 Cortex-M0+ LockUp 功能是否引起看门狗定时器运行的配置寄存器，PRCR 是其写保护寄存器。

通过 8 位存储器操作指令设置 PRCR 寄存器。

在产生复位信号后 PRCR 寄存器的值变为“00H”。

位	符号	描述	复位值
7:1	PRTKEY	PRCR的写保护 78H: PRCR可写 其他: PRCR不可写	0x0
0	PRCR	LOCKUP控制寄存器写保护 0: LOCKCTL寄存器不可写 1: LOCKCTL寄存器可写	0

9.4 看门狗定时器的运行

9.4.1 看门狗定时器的运行控制

1. 当使用看门狗定时器时，通过选项字节（000C0H）设定以下内容：
   (1) 必须将选项字节（000C0H）的 bit4（WDTON）置“1”，允许看门狗定时器的计数运行（在解除复位后，计数器开始运行）（详细内容请参照第 29 章 选项字节）。

WDTON	看门狗定时器的计数器
0	禁止计数运行（解除复位后停止计数）。
1	允许计数运行（解除复位后开始计数）。

   (2) 必须通过选项字节（000C0H）的 bit3～1（WDCS2～WDCS0）设定上溢时间（详细内容请参照 9.4.2 和第 29 章）。
   (3) 必须通过选项字节（000C0H）的 bit6 和 bit5（WINDOW1、WINDOW0）设定窗口打开期间（详细内容请参照 9.4.2 和第 29 章）。
2. 在解除复位后，看门狗定时器开始计数。
3. 在开始计数后并且在选项字节所设上溢时间前，如果给看门狗定时器的允许寄存器（WDTE）写“ACH”，就清除看门狗定时器并且重新开始计数。
4. 此后，解除复位后第 2 次以后的 WDTE 寄存器的写操作必须在窗口打开期间进行。如果在窗口关闭期间写 WDTE 寄存器，就产生内部复位信号。
5. 如果不给 WDTE 寄存器写“ACH”而超过上溢时间，就产生内部复位信号。以下情况会产生内部复位信号：
   (1) 当给 WDTE 寄存器写“ACH”以外的数据时

注 1：只在解除复位后第 1 次写看门狗定时器的允许寄存器（WDTE）时，与窗口打开期间无关，只要在上溢时间前的任意时候写 WDTE，就清除看门狗定时器并且重新开始计数。
注 2：给 WDTE 寄存器写“ACH”到清除看门狗定时器的计数器为止，有可能产生最大 2 个 FIL 时钟的误差。
注 3：数值发生上溢前，都能清除看门狗定时器。
注 4：所示，看门狗定时器在睡眠或者深度睡眠模式中的运行因选项字节（000C0H）的 bit0（WDSTBYON）的设定值而不同。

	WDSTBYON=0	WDSTBYON=1
睡眠模式	停止看门狗定时器运行。	继续看门狗定时器运行。
深度睡眠模式		

当 WDSTBYON 位为“0”时，在解除睡眠或者深度睡眠模式后重新开始看门狗定时器的计数。此时，将计数器清“0”，开始计数。

如果从解除深度睡眠模式到看门狗定时器发生上溢的时间较短，就会在振荡稳定时间内发生看门狗上溢而产生复位。

9.4.2 看门狗定时器上溢时间的设定

通过选项字节（000C0H）的 bit3～1（WDCS2～WDCS0）设定看门狗定时器的上溢时间。

在发生上溢时，产生内部复位信号。如果在上溢时间前的窗口打开期间给看门狗定时器的允许寄存器（WDTE）写“ACH”，就清除计数并且重新开始计数。能设定的上溢时间如下所示。

表9-3：看门狗定时器上溢时间的设定

WDCS2	WDCS1	WDCS0	看门狗定时器的上溢时间 （F_{IL}=20kHz(MAX.)的情况）
0	0	0	$2^6/F_{IL}$（3.2ms）
0	0	1	$2^7/F_{IL}$（6.4ms）
0	1	0	$2^8/F_{IL}$（12.8ms）
0	1	1	$2^9/F_{IL}$（25.6ms）
1	0	0	$2^{11}/F_{IL}$（102.4ms）
1	0	1	$2^{13}/F_{IL}$（409.6ms）
1	1	0	$2^{14}/F_{IL}$（819.2ms）
1	1	1	$2^{16}/F_{IL}$（3276.8ms）

注：F_{IL}：低速内部振荡器的时钟频率

9.4.3 看门狗定时器窗口打开期间的设定

通过选项字节（000C0H）的 bit6 和 bit5（WINDOW1、WINDOW0）设定看门狗定时器的窗口打开期间。窗口概要如下：

- 如果在窗口打开期间给看门狗定时器的允许寄存器（WDTE）写“ACH”，就清除看门狗定时器并且重新开始计数。
- 在窗口关闭期间，即使给WDTE寄存器写“ACH”，也会检测到异常并且产生内部复位信号。

注：只在解除复位后第1次写WDTE寄存器时，与窗口打开期间无关，只要在上溢时间前的任意时候写WDTE，就清除看门狗定时器并且重新开始计数。

能设定的窗口打开期间如下所示。

表9-4：看门狗定时器窗口打开期间的设定

WINDOW1	WINDOW0	看门狗定时器的窗口打开期间
0	-	禁止设定
1	0	75%
1	1	100%

注1：当选项字节（000C0H）的bit0（WDSTBYON）为“0”时，与WINDOW1位和WINDOW0位的值无关，窗口打开期间为100%。

注2：当将上溢时间设定为$2^9/F_{IL}$的情况时，窗口关闭时间和打开时间如下所示。

	窗口打开期间的设定	
	75%	100%
窗口关闭时间	0～12.8ms	无
窗口打开时间	12.8～25.6ms	0～25.6ms

＜当窗口打开期间为75%时＞

(1) 上溢时间：

$2^9/F_{IL}$(MAX.)=2^9/20kHz(MAX.)=25.6ms

(2) 窗口关闭时间：

0～$2^9/F_{IL}$(MIN.)×(1–0.75)=0～2^9/10kHz×0.25=0～12.8ms

(3) 窗口打开时间：

$2^9/F_{IL}$(MIN.)×(1–0.75)～$2^9/F_{IL}$(MAX.)=12.8～25.6ms

9.4.4 看门狗定时器间隔中断的设定

能通过设定选项字节（000C0H）的 bit7（WDTINT），在达到上溢时间的 75%+1/2F_{IL} 时产生间隔中断（INTWDTI）。

表9-5：看门狗定时器间隔中断的设定

WDTINT	看门狗定时器间隔中断的使用/不使用
0	不使用间隔中断。
1	在达到上溢时间的75%+1/2F_{IL}时，产生间隔中断。

注1：当解除深度睡眠模式后以X1振荡时钟运行时，CPU在经过振荡稳定时间后开始运行。
如果从解除深度睡眠模式到看门狗定时器发生上溢的时间较短，就会在振荡稳定时间内发生看门狗上溢而产生复位。

注2：即使在产生INTWDTI后也继续计数（继续到给看门狗定时器的允许寄存器（WDTE）写“ACH”为止）。如果在上溢时间前不给WDTE寄存器写“ACH”，就产生内部复位信号。

9.4.5 LOCKUP期间看门狗定时器的运行

当 LOCKUP 控制寄存器 LOCKCTL 的 lockup_rst 位设置为 1 时，一旦内核进入 LOCKUP 状态，低速内部振荡器开始发振，看门狗定时器的计时器自动开始运行，并将上溢时间的控制位（WDCS2~WDCS0）设置为 3’b010，即设置上溢时间为 12.8ms。

第10章 除法与开方运算单元（DIVSQRT）

10.1 概述

芯片包含一个 32bit/32bit 的硬件除法器与一个 32bit 的硬件开方器。

10.2 特性

- 支持有/无符号除法与开方运算。
- 商和余数均为 32 位宽度。
- 除法器除零标志指示位。
- 22 个 APB 时钟运算完成。
- 写 ALUB 寄存器启动运算。

10.3 功能说明

运算单元可以通过寄存器 DIVSQRT->CON[4]选择除法模式或者开根号模式，除法模式下寄存器 DIVSQRT->RES0 保存商和寄存器 DIVSQRT->RES1 保存余数；可以通过寄存器 DIVSQRT->CON[2]判别除数是否为 0，该位为只读位；开根号模式下 DIVSQRT->RES0 保存开根号结果，DIVSQRT->RES1 未用。

注意在开根号模式下，被开根号数最高位为 1，则被当成有符号数处理，先取绝对值，再进行开根号运算：$RES0 = \sqrt[2]{absval(ALUB)}$

同时可以通过寄存器 DIVSQRT->CON[3]判别是否运算完毕，该位为只读位，读取值为 0 表示正在运算，为 1 表示运算完毕，当除法器处于空闲状态时该位也为 1。

除法器可通过 DIVSQRT->CON[1]选择有符号还是无符号除法模式。

需要注意的是运算单元的时钟使能位在外围允许寄存器 PER12 中设置。

注：在计算期间请不要写 ALUA 或 ALUB 寄存器，也不要读 RES0 或 RES1 寄存器，否则结果不可预知。

寄存器在不同模式下的定义：

运算单元模式	ALUA	ALUB	RES0	RES1
除法模式	被除数	除数	商	余数
开根号模式	-	被开根号数	开根号结果(低16位有效)	-

10.4 寄存器映射

(DIVSQRT 基地址 = 0x4006_4380)
RO：只读，WO：只写，R/W：读写

寄存器	偏移量	读/写	描述	复位值
CON	0x000	R/W	运算单元控制寄存器	0xC
ALUA	0x004	R/W	运算单元数据A寄存器	0x0
ALUB	0x008	R/W	运算单元数据B寄存器	0x0
RES0	0x00C	RO	运算单元结果0寄存器	0x0
RES1	0x010	RO	运算单元结果1寄存器	0x0

10.5 寄存器说明

10.5.1 除法开方器控制寄存器(DIVSQRTCON)

位	符号	描述	复位值
31:5	-	保留	-
4	MODE	运算模式选择位 0: 除法模式 1: 开根号模式	0
3	READY	运算完毕指示位 0: 运算正在进行 1: 运算完毕或空闲状态	1
2	DIVBY0	除法模式除零指示位（写除数后自动更新该位） 0: 除数不为0 1: 除数为0	1
1	SIGN	除法模式符号选择位 0: 无符号模式 1: 有符号模式	0
0	-	保留	0

10.5.2 除法开方器数据A寄存器(DIVSQRTALUA)

位	符号	描述	复位值
31:0	ALUA	32位数据A	0x0

10.5.3 除法开方器数据B寄存器(DIVSQRTALUB)

位	符号	描述	复位值
31:0	ALUB	32位数据B	0x0

10.5.4 除法开方器结果0寄存器(DIVSQRTRES0)

位	符号	描述	复位值
31:0	RES0	32位结果0	0x0

10.5.5 除法开方器结果1寄存器(DIVSQRTRES1)

位	符号	描述	复位值
31:0	RES1	32位结果1	0x0

第11章 定时器（TIMER0/1）

11.1 概述

包含2路可编程的32位/16位计数器，即TIMER0/TIMER1，为用户提供便捷的定时计数功能。

11.2 特性

- 可配置 32 位/16 位向下计数器。
- 每个定时器都有独立预分频器。
- 提供单次触发，周期计数，连续计数三种计数操作模式。
- 支持芯片从休眠模式唤醒。

11.3 功能描述

11.3.1 单次触发模式

如果定时器工作在单次触发模式，使能定时器后，计数器从加载寄存器加载初值，向下计数，当计数器递减到0时，停止工作，同时产生中断。若要再次启动单次触发模式，需清零TMROS位，再置位TMROS位。

（再次启动单次触发模式时，需注意TMROS位清零时，保持为0的时间须大于一个定时器计数周期）

11.3.2 周期计数模式

如果定时器工作在周期计数模式，使能定时器后，计数器从加载寄存器加载初值，向下计数，当计数器递减到0时，计数器从加载寄存器加载初值，并继续计数，同时产生中断。

11.3.3 连续计数模式

如果定时器工作在连续计数模式，使能定时器后，计数器从加载寄存器加载初值，向下计数，当计数器递减到0时，计数器加载最大值作为初值，并继续计数，同时产生中断。

11.3.4 延迟加载功能

当数据写入加载寄存器时，计数器不会继续递减，会在下一个TIMER_CLK上升沿从加载寄存器中加载初值，再递减计数。

当数据写入延迟加载寄存器时，数据在下一个TIMER_CLK上升沿写入加载寄存器，若计数器已经开始计数，则会等待当前周期计数为0，再从加载寄存器中加载初值。

11.4 寄存器映射

（Timer0基地址= 0x4006_1000，Timer1基地址= 0x4006_1100）

RO：只读；WO：只写；R/W：读写；

寄存器	偏移量	读/写	描述	复位值
CON	0x000	R/W	定时器控制寄存器	0x20
LOAD	0x004	R/W	定时器加载寄存器	0x0
VAL	0x008	RO	定时器当前值寄存器	0xFFFFFFFF
RIS	0x00C	RO	定时器中断源状态寄存器	0x0
MIS	0x010	RO	定时器已使能中断状态寄存器	0x0
ICLR	0x014	WO	定时器中断清零寄存器	-
BGLOAD	0x018	R/W	定时器延迟加载寄存器	0x0

11.5 寄存器说明

11.5.1 定时器控制寄存器（CON0/1）

位	符号	描述	复位值
31:8	-	保留	-
7	TMREN	定时器使能位 0: 禁止 1: 使能	0
6	TMRMS	定时器模式选择位 0: 连续计数模式 1: 周期计数模式	0
5	TMRIE	定时器中断使能位 0: 禁止中断 1: 使能中断	1
4	-	保留	-
3:2	TMRPRE	定时器预分频 00: 1分频 01: 16分频 10: 256分频 11: 保留	0x0
1	TMRSZ	定时器计数位数选择 0: 16位计数器 1: 32位计数器	0
0	TMROS	单次触发模式选择位 0: 模式由TMRMS位确定 1: 单次触发模式 （再次触发单次模式，其初始值由TMRMS位确定）	0

11.5.2 定时器加载寄存器（LOAD0/1）

位	符号	描述	复位值
31:0	TMRxLOAD	定时器加载寄存器	0x0

11.5.3 定时器当前值寄存器（VAL0/1）

位	符号	描述	复位值
31:0	TMRxVAL	定时器当前计数值	0xFFFFFFFF

11.5.4 定时器中断源状态寄存器（RIS0/1）

位	符号	描述	复位值
31:1	-	保留	-
0	TMRxRIS	定时器中断源状态 1: 产生中断 0: 未产生中断	0

11.5.5 定时器已使能中断状态寄存器（MIS0/1）

位	符号	描述	复位值
31:1	-	保留	-
0	TMRxMIS	定时器已使能中断状态位 1: 中断使能并产生中断 0: 未产生中断	0

11.5.6 定时器中断清零寄存器（ICLR0/1）

位	符号	描述	复位值
31:0	TMRxICLR	写入任意数，清零定时器中断	-

11.5.7 定时器延迟加载寄存器（BGLOAD0/1）

位	符号	描述	复位值
31:0	TMRxBGLOAD	定时器延迟加载寄存器（读取值为最近一次写入TMRxLOAD或TIMERxBGLOAD的值）	0x0

第12章 捕捉/比较/脉宽调制模块（CCP0/1）

12.1 概述

包含2组CCP模块CCP0/CCP1，每组CCP对应A，B两路通道。CCP0对应CCP0A/CCP0B，CCP1对应CCP1A/CCP1B。

12.2 特性

- 多达 2 组 CCP，最多支持 4 路 PWM 输出。
- 每组 CCP 都可设置独立的周期。
- CCPn 内部有 16 位计数器，可产生比较/溢出中断。
- CCPn 具有独立捕捉功能，可选在 A 路或 B 路管脚输入信号。
- CCP1 具有 4 通道捕捉功能，可同时捕获 CCP0A/CCP0B/CCP1A/CCP1B 输入信号。
- 捕捉模式 1 下支持捕获操作重新加载 CCP0 计数器功能。
- 内部通道 CAP3 支持模拟比较器输出捕获功能。
- 内部通道 CAP0-CAP3 支持软件捕获功能。

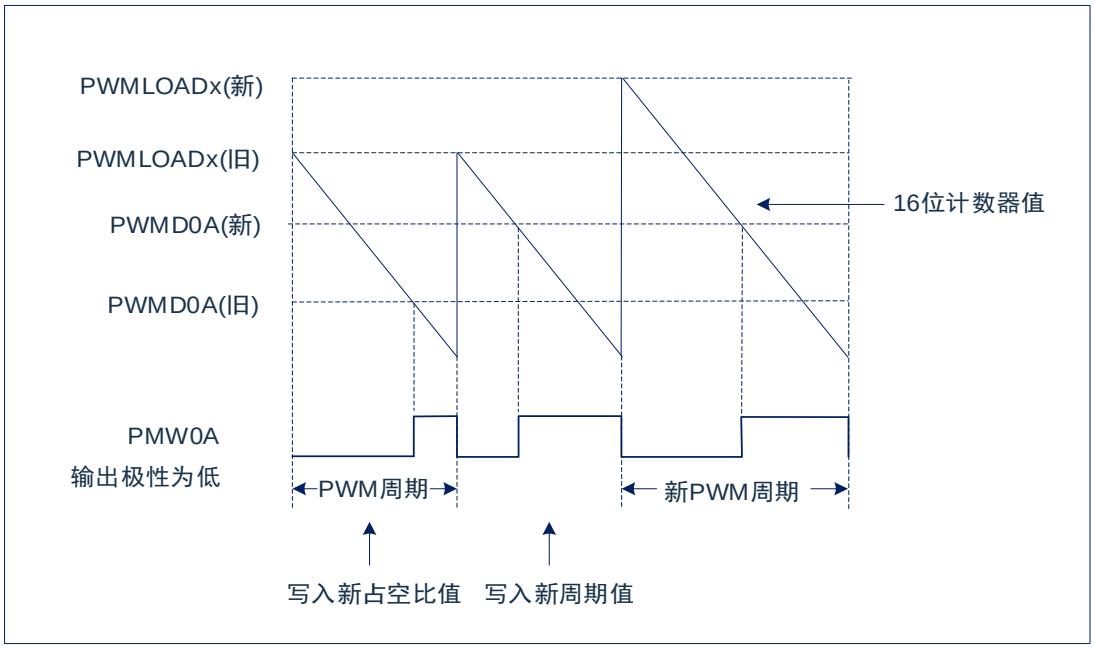
12.3 功能描述

12.3.1 脉宽调制模式（PWM）

每组CCP可输出A、B两路PWM：PWMxA，PWMxB，这两路共用一个周期，输出占空比可以通过CCPDxA，CCPDxB独立设置。PWMxA/PWMxB输出极性可通过PWMxAO/PWMxBO位设置，且分别对应CCPxA/CCPxB通道输出。

当CCPx运行位置1后，16位计数器加载CCPx重加载寄存器的值，向下计数，当计数值等于CCPDxA/B的值时，PWMxA/PWMxB输出电平发生改变。

图 12-1：PWM 时序图



周期和占空比计算方式如下：

周期=CCPLOAD×CCP时钟周期。

PWMxA占空比=CCPDxA/CCPLOADx (支持0%~100%)。

PWMxB占空比=CCPDxB/CCPLOADx (支持0%~100%)。

CCPLOADx=0时，PWMxA，PWMxB占空比为0%。

CCPDxA/CCPDxA>CCPLOADx是时，占空比为100%。

12.3.2 方波输出模式

方波输出模式属于脉宽调制模式的一种，该模式周期自由可调，占空比固定为50%。

每组CCP均可设置从A路或B为BUZ输出模式。设置方波输出模式时需要配置PWM模式，且将CCPxCON.ZAEN或CCPxCON.ZAEN置位。周期由CCPLOADx来配置，占空比默认加载CCPLOADx值的一半（最低位忽略）。

12.3.3 捕捉模式0

该捕捉模式为外部捕捉。

每组CCP可设置从A路或B路作为外部捕捉信号管脚，CCPRUNx置位后，16位计数从0xFFFF开始向下计数，当触发捕捉条件时，计数器停止计数，CCPxA或CCPxB返回当前计数器的值。若需要进行下一次捕捉，需将CCPRUNx清零，再置位。

捕捉时间计算方式为：

CCPLOADx.RELOAD=0，捕捉时间=（0xFFFF -CCPDxA/B）×CCPx时钟周期

CCPLOADx.RELOAD=1，捕捉时间=（CCPxLOAD[15:0]-CCPDxA/B）×CCPx时钟周期。

12.3.4 捕捉模式1

CCP1包括4路内部通道有CAP0，CAP1，CAP2，CAP3。其中一路通道可选择外部通道中ECAP00-02或ECAP10-13中任意一路作为捕获通道。也可分别选择CCP0A/CCP0B/CCP1A/CCP1B作为捕获通道。

ECAP00-02对应模拟比较器0的正端输入C0P0-C0P2。

ECAP10-13对应模拟比较器1的正端输入C1P0-C1P3。

使用ECAP外部捕获时需要将相应的端口设置为GPIO功能。

使用CCP0A/CCP0B/CCP1A/CCP1B捕获时，需要将相应口设置为CCP口。

CAPn与外部通道的对应关系：

内部通道	外部通道	
CAP0	CAP0CHS=n:	选择ECAP0n (n=0-2)@ECAPS=0
	CAP0CHS=n:	选择ECAP1n (n=0-3)@ECAPS=1
	CAP0CHS=F:	选择CAP0A
	CAP0CHS=其他值:	保留
CAP1	CAP1CHS=n:	选择ECAP0n (n=0-2)@ECAPS=0
	CAP1CHS=n:	选择ECAP1n (n=0-3)@ECAPS=1
	CAP1CHS=F:	选择CAP0B
	CAP1CHS=其他值:	保留
CAP2	CAP2CHS=n:	选择ECAP0n (n=0-2)@ECAPS=0
	CAP2CHS=n:	选择ECAP1n (n=0-3)@ECAPS=1
	CAP2CHS=F:	选择CAP1A
	CAP2CHS=其他值:	保留
CAP3	CAP3CHS=n:	选择ECAP0n (n=0-2)@ECAPS=0
	CAP3CHS=n:	选择ECAP1n (n=0-3)@ECAPS=1
	CAP3CHS=8:	选择ACMP0滤波选择后输出
	CAP3CHS=9:	选择ACMP1滤波选择后输出
	CAP3CHS=F:	选择CAP1B
	CAP3CHS=其他值:	保留

在捕获模式1下，CCP0与CCP1的PWM模式输出与外部捕捉模式0禁止。

该模式需要CCP1工作在计数模式下，捕获操作将CCP1计数中值装入到相关寄存器中。

另外CCP0可选择工作在计数模式下，可分别设置CAP0-CAP3捕获触发加载功能。即设置的通道有捕获操作产生时，将重新加载CCP0的计数器初值。多个通道可同时设置该功能，软件触发捕捉则不会重新加载CCP0的初值。

在捕获模式1下，CCP0与CCP1的比较/溢出中断功能可正常使用。

该捕捉方式分为两种：一种外部信号触发捕捉，一种为软件触发捕捉。

1) 外部信号触发捕捉：

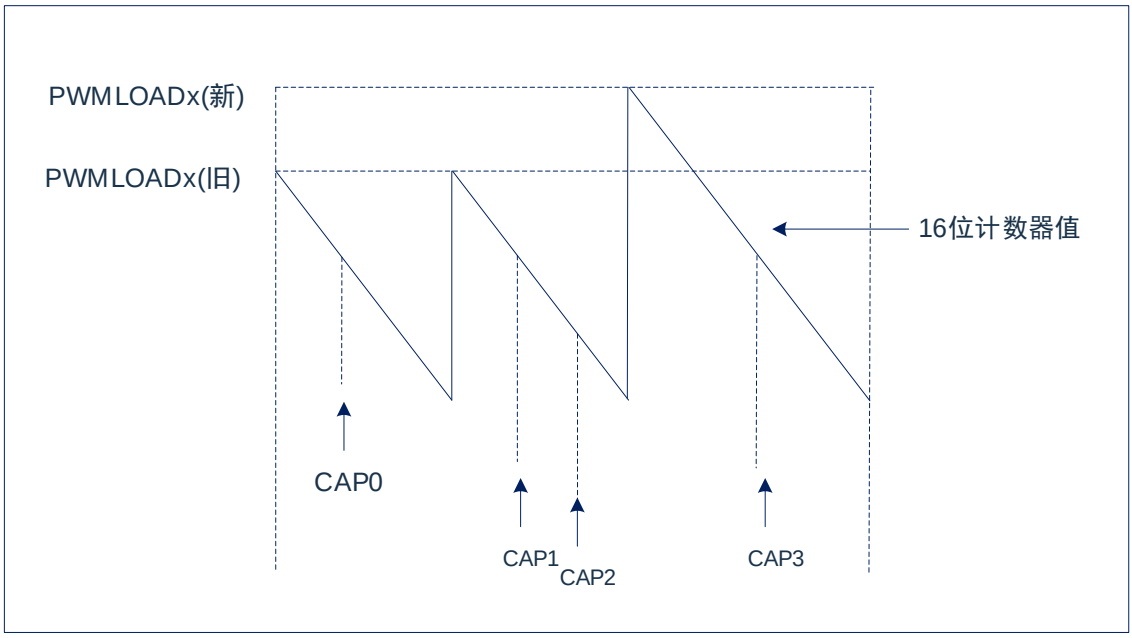
   CAP0-CAP3均可选择上升沿/下降沿或双沿捕获。当产生信号时，将CCP1计数器的值捕获到相应的寄存器中，且产生中断标志。4个通道与捕获寄存器的对应关系如下：

   CAP0/CAP1/CAP2/CAP3分别对应CAP0DAT/CAP1DAT/CAP2DAT/CAP3DAT寄存器。

2) 软件触发捕捉：

   对CAP0DAT-CAP3DAT进行写操作，则分别对CAP0-CAP3通道产生捕获操作。将CCP1计数器的值捕获到相应的寄存器中。且写入的31-16位必须为0x55AA，才能触发捕获操作，与写入的低16位数据不相关。软件触发捕捉不会产生中断标志。

图 12-2：CAP0-CAP3 通道捕获操作



12.3.5 捕捉模式2

该捕捉模式为外部捕捉。该模式主要用来捕捉外部输入的PWM波形信息。

在捕捉模式下2，CAP2，CAP3的通道映射到CAP1，即CAP1-3为同一捕获通道，CAP0禁止使用。

CCP0可自由设置，不受影响。

CCP1工作在计数模式下，CCPRUN1置位后，16位计数从初始值开始向下计数。如果溢出且没有产生捕获操作，则计数器重新加载初值继续向下计数。

启动捕获模式2的操作如下：

（1）CAP1发生捕获之后，CCP1计数器重新加载初值，同时将该值装载到CAP1DATA，允许CAP2产生捕获；

（2）CAP2发生捕获之后，将捕获的计数器的值装载到CAP2DATA，允许CAP3发生捕获，

（3）CAP3发生捕获之后，将捕获的计数器的值装载到CAP3DATA， 完成整次捕获。此时将CAP1DATA-CAP3DATA的值装入CAP0DATA[31:16]，将CAP1DATA-CAP2DATA的值装入CAP0DATA[15:0]。然后CAP1产生捕获动作，回到第（1）步。

注：

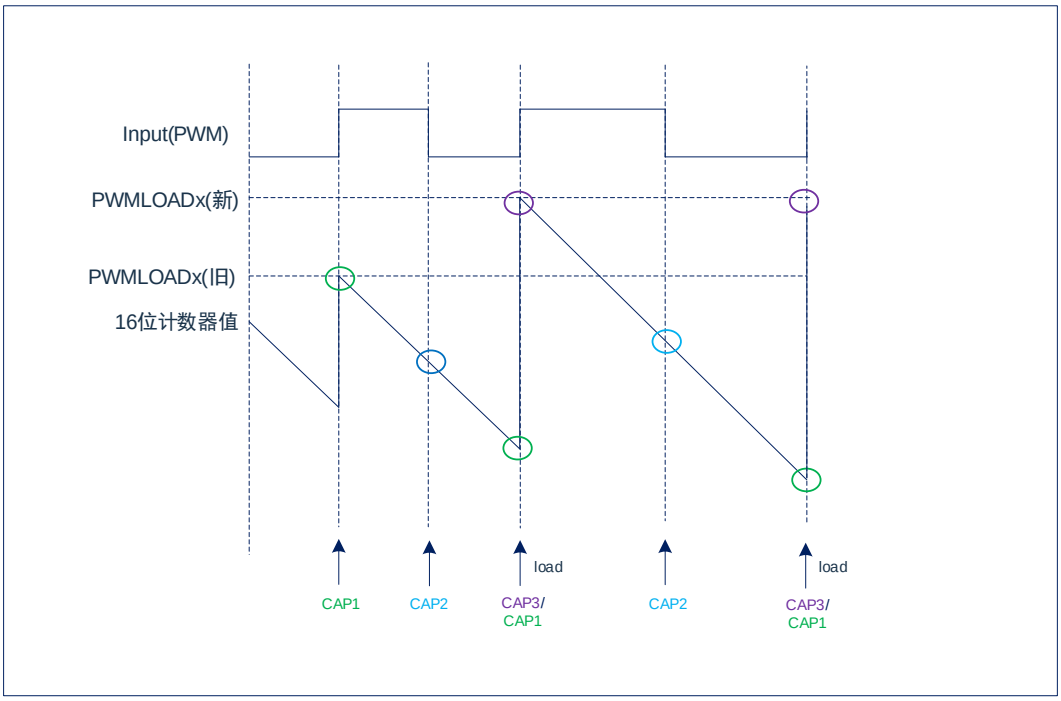
a：首次CAP1的捕获操作也会将更新CAP0DATA的值，但此时值无参考意义，建议丢弃。

b：如果完成CAP3产生的捕获的时间接超过了CCP1计数器的一个周期，将会发生溢出，计算出的CAP0DATA的值是不准确的。建议设置计数器的周期远大于需要捕获的PWM周期。

c：CAP1到CAP2，CAP2到CAP3捕获的间隔时间需要大于8个CCP1计数值。

d：捕获模式2支持CAP1-3的软件捕获动作。

e：CAP1-CPA3共用同一个捕获通道，捕获操作会同时在CAP1-CAP3上产生相应的捕获标志位。



12.3.6 PWM配置过程

- 配置 PWM 控制寄存器，设置预分频，选择 PWM 模式，使能 PWM。
- 配置 PWM 周期，写入 CCPLOADx 寄存器。
- 配置 PWM 占空比，写入 CCPDxA/CCPDxB 寄存器。
- 若需要中断，使能相关中断位，清零中断状态寄存器。
- 设置相应 I/O 口为 PWM 输出。
- 设置 PWM 运行寄存器，开始输出。

12.3.7 中断

在 PWM 模式下，CCPx 可产生两种中断：

- 当计数器递减到 0 时，产生溢出中断。
- 当计数器的值与 CCPDxA 或 CCPDxB 的值相等时，产生比较中断。

捕捉模式 0/1 下，可产生两种中断：

- 当计数器递减到 0 时，产生溢出中断。
- 触发捕捉条件时，产生捕捉中断。

12.4 寄存器映射

（CCP基地址 = 0x4006_4280）RO：只读；WO：只写；R/W：读写。

寄存器	偏移量	读/写	描述	复位值
CCPCON0(P1B)	0x000	R/W	CCP0控制寄存器	0x0
CCPLOAD0(P1A)	0x004	R/W	CCP0重加载寄存器	0x0
CCPD0A(P1A)	0x008	R/W	CCP0通道A数据寄存器	0x0
CCPD0B(P1A)	0x00C	R/W	CCP0通道B数据寄存器	0x0
CCPCON1(P1B)	0x010	R/W	CCP1控制寄存器	0x0
CCPLOAD1(P1A)	0x014	R/W	CCP1重加载寄存器	0x0
CCPD1A(P1A)	0x018	R/W	CCP1通道A数据寄存器	0x0
CCPD1B(P1A)	0x01C	R/W	CCP1通道B数据寄存器	0x0
CCPIMSC(P1B)	0x040	R/W	CCP中断使能寄存器	0x0
CCPRIS	0x044	RO	CCP中断源状态寄存器	0x0
CCPMIS	0x048	RO	CCP已使能中断状态寄存器	0x0
CCPICLR	0x04C	WO	CCP中断清零寄存器	0x0
CCPRUN(P1B)	0x050	R/W	CCP运行寄存器	0x0
CCPLOCK	0x054	R/W	CPP0/1写使能寄存器	0x0
CAPCON(P1B)	0x058	R/W	捕获控制寄存器	0x0
CAPCHS(P1B)	0x05C	R/W	捕获通道选择寄存器	0x0
CAP0DAT0(P1A)	0x060	R/W	捕获通道0数据寄存器	0x0
CAP1DAT0(P1A)	0x064	R/W	捕获通道1数据寄存器	0x0
CAP2DAT0(P1A)	0x068	R/W	捕获通道2数据寄存器	0x0
CAP3DAT0(P1A)	0x06C	R/W	捕获通道3数据寄存器	0x0

注：
(P1A/P1B)标注的寄存器为被保护的寄存器。
(P1A)：LOCK==55H 或 AAH 时，标注的寄存器允许写入；=其他值，禁止写入。
(P1B)：LOCK==55H 时，标注的寄存器允许写入；=其他值，禁止写入。

12.5 寄存器说明

12.5.1 CCPx控制寄存器（CCPCONx）（x=0,1）

位	符号	描述	复位值
31:10	-	保留	-
9	CCPxZBEN	通道B的方波模式使能位（PWM模式有效） 0: 禁止 1: 使能，占空比加载值为LOADx/2	0
8	CCPxZAEN	通道A的方波模式使能位（PWM模式有效） 0: 禁止 1: 使能，占空比加载值为LOADx/2	0
7	-	保留	-
6	CCPxEN	CCPx使能位 0: 禁止 1: 使能	0
5:4	CCPxPS	CCPx预分频选择 0x0: PCLk 0x1: PCLk/4 0x2: PCLk/16 0x3: PCLk/64	0x0
3	CCPxMS	CCPx模式选择 0: 捕捉模式0（CAPEN=0时生效） 1: PWM模式（CAPEN=0时生效）	0
2	CCPxCM0CS	CCPx捕捉模式0捕捉通道选择 0: 通道CCPxA 1: 通道CCPxB	0
1:0	CCPxCM0ES	CCPx捕捉模式0捕捉方式选择 0x0: CCPRUNx=1开始计数，上升沿捕捉并产生中断 0x1: CCPRUNx=1开始计数，下降沿捕捉并产生中断 0x2: 上升沿开始计数，下降沿捕捉并产生中断 0x3: 下降沿开始计数，上升沿捕捉并产生中断	0x0

12.5.2 CCP重加载寄存器（CCPLOADx）（x=0,1）

位	符号	描述	复位值
31:17	-	保留	-
16	RELOAD	**CCP0模块:** PWM模式下: 重加载使能位 0: 计数器重加载值为0xFFFF 1: 计数器重加载值为CCP0LOAD 捕捉模式0下: 0: 计数器重加载值为0xFFFF 1: 计数器重加载值为CCP0LOAD **CCP1模块:** PWM模式下: 重加载使能位 0: 计数器重加载值为0xFFFF 1: 计数器重加载值为CCP1LOAD 捕捉模式0、1下: 0: 计数器重加载值为0xFFFF 1: 计数器重加载值为CCP1LOAD	0
15:0	CCPxLOAD	CCPx计数器的加载值（建议加载值不为0）	0x0

12.5.3 CCPxA数据寄存器（CCPDxA）（x=0,1）

位	符号	描述	复位值
31:17	-	保留	-
16	PWMxAOP	PWMxA输出极性选择 0: 正常输出 1: 反相输出	0
15:0	CCPxADATA	PWM模式时: PWMxA的占空比 捕捉模式0时: 捕捉结果	0x0

12.5.4 CCPxB数据寄存器（CCPDxB）（x=0,1）

位	符号	描述	复位值
31:17	-	保留	-
16	PWMxBOP	PWMxB输出极性选择 0: 正常输出 1: 反相输出	0
15:0	CCPxBDATA	PWM模式时: PWMxB的占空比 捕捉模式0时: 捕捉结果	0x0

12.5.5 CCP中断使能寄存器（CCPIMSC）

位	符号	描述	复位值
31:12	-	保留	-
11	CAP3IMSC	CAP3捕获中断使能位 0: 禁止 1: 使能	0
10	CAP2IMSC	CAP2捕获中断使能位 0: 禁止 1: 使能	0
9	CAP1IMSC	CAP1捕获中断使能位 0: 禁止 1: 使能	0
8	CAP0IMSC	CAP0捕获中断使能位 0: 禁止 1: 使能	0
7:6	-	保留	0x0
5	PWMIMSC	PWM1溢出中断使能位 0: 禁止 1: 使能	0
4	PWMIMSC4	PWM0溢出中断使能位 0: 禁止 1: 使能	0
3:2	-	保留	-
1	PWMIMSC1	PWM1比较/捕捉中断使能位 0: 禁止 1: 使能	0
0	PWMIMSC0	PWM0比较/捕捉中断使能位 0: 禁止 1: 使能	0

12.5.6 CCP中断源状态寄存器（CCPRIS）

位	符号	描述	复位值
31:12	-	保留	-
11	CAP3RIS	CAP3捕获中断状态位 1: 产生中断 0: 未产生中断	0
10	CAP2RIS	CAP2捕获中断状态位 1: 产生中断 0: 未产生中断	0
9	CAP1RIS	CAP1捕获中断状态位 1: 产生中断 0: 未产生中断	0
8	CAP0RIS	CAP0捕获中断状态位 1: 产生中断 0: 未产生中断	0
7:6	-	保留	0x0
5	PWMRIS5	PWM1溢出中断状态位 1: 产生中断 0: 未产生中断	0
4	PWMRIS4	PWM0溢出中断状态位 1: 产生中断 0: 未产生中断	0
3:2	-	保留	-
1	PWMRIS1	PWM1比较/捕捉中断状态位 1: 产生中断 0: 未产生中断	0
0	PWMRIS0	PWM0比较/捕捉中断状态位 1: 产生中断 0: 未产生中断	0

12.5.7 CCP已使能中断状态寄存器（CCPMIS）

位	符号	描述	复位值
31:12	-	保留	-
11	CAP3MIS	CAP3已使能捕获中断状态位 1: 中断使能并产生中断 0: 未产生中断	0
10	CAP2MIS	CAP2已使能捕获中断状态位 1: 中断使能并产生中断 0: 未产生中断	0
9	CAP1MIS	CAP1已使能捕获中断状态位 1: 中断使能并产生中断 0: 未产生中断	0
8	CAP0MIS	CAP0已使能捕获中断状态位 1: 中断使能并产生中断 0: 未产生中断	0
7:6	-	保留	-
5	PWMMIS5	PWM1已使能溢出中断状态位 1: 中断使能并产生中断 0: 未产生中断	0
4	PWMMIS4	PWM0已使能溢出中断状态位 1: 中断使能并产生中断 0: 未产生中断	0
3:2	-	保留	-
1	PWMMIS1	PWM1已使能比较/捕捉中断状态位 1: 中断使能并产生中断 0: 未产生中断	0
0	PWMMIS0	PWM0已使能比较/捕捉中断状态位 1: 中断使能并产生中断 0: 未产生中断	0

12.5.8 CCP中断清零寄存器（CCPICLR）

位	符号	描述	复位值
31:12	-	保留	-
11	CAP3ICLR	清除CAP3捕获中断状态位	0
10	CAP2ICLR	写1清除CAP2捕获中断状态位	0
9	CAP1ICLR	写1清除CAP1捕获中断状态位	0
8	CAP0ICLR	写1清除CAP0捕获中断状态位	0
7:6	-	保留	-
5	PWMICLR5	写1清除PWM1溢出中断状态位	0
4	PWMICLR4	写1清除PWM0溢出中断状态位	0
3:2	-	保留	-
1	PWMICLR1	写1清除PWM1比较/捕捉中断状态位	0
0	PWMICLR0	写1清除PWM0比较/捕捉中断状态位	0

12.5.9 CCP运行寄存器（CCPRUN）

位	符号	描述	复位值
31:2	-	保留	-
1	CCPRUN1	CCP1运行控制位 0: 停止 1: 运行	0
0	CCPRUN0	CCP0运行控制位 0: 停止 1: 运行	0

12.5.10 CCP写使能控制寄存器(LOCK)

位	符号	描述	复位值
31:8	-	保留	-
7:0	LOCK	当LOCK=0xaa时，使能操作保护级别为P1A的寄存器； 当LOCK=0x55时，使能操作保护级别为P1B与P1A的寄存器； 当LOCK=其他值时，禁止操作有保护级别的寄存器。	0x0

12.5.11 CAP控制寄存器（CAPCON）

位	符号	描述	复位值
31:14	-	保留	-
13	CAPEN2	捕捉模式2使能位（仅对CCP1有效） 0： -- 1： 捕捉模式2使能位，同时禁止捕捉模式1	0
12	CAPEN	捕捉模式1使能位 0: CCP0/CCP1为PWM模式或捕获模式0使能 1: 捕获模式1使能，即全通道捕获模式 CCP0可设置为连续计数模式 CCP1可设置为连续计数模式	0
11	CAP3RLEN	捕捉模式1下CAP3捕获触发CCP0的计数器加载使能位 0: 禁止 1: 使能，（需要在捕获模式1，且CCP0运行状态下生效） CAP3出现捕获触发信号，则CCP0在计数器的运行过程中，将重新加载CCP0LOAD寄存器中数据。	0
10	CAP2RLEN	捕捉模式1下CAP2捕获触发CCP0的计数器加载使能位 0: 禁止 1: 使能，（需要在捕获模式1，且CCP0运行状态下生效） CAP2出现捕获触发信号，则CCP0在计数器的运行过程中，将重新加载CCP0LOAD寄存器中数据。	0
9	CAP1RLEN	捕捉模式1下CAP1捕获触发CCP0的计数器加载使能位 0: 禁止 1: 使能，（需要在捕获模式1，且CCP0运行状态下生效） CAP1出现捕获触发信号，则CCP0在计数器的运行过程中，将重新加载CCP0LOAD寄存器中数据。	0
8	CAP0RLEN	捕捉模式1下CAP0捕获触发CCP0的计数器加载使能位 0: 禁止 1: 使能，（需要在捕获模式1，且CCP0运行状态下生效） CAP0出现捕获触发信号，则CCP0在计数器的运行过程中，将重新加载CCP0LOAD寄存器中数据。	0
7:6	CAP3ES	CAP3捕捉模式选择 0x0: 禁止 0x1: 上升沿捕捉 0x2: 下降沿捕捉 0x3: 双沿	0x0
5:4	CAP2ES	CAP2捕捉模式选择 0x0: 禁止 0x1: 上升沿捕捉 0x2: 下降沿捕捉 0x3: 双沿	0x0
3:2	CAP1ES	CAP1捕捉模式选择	0x0

		0x0: 禁止 0x1: 上升沿捕捉 0x2: 下降沿捕捉 0x3: 双沿	
1:0	CAP0ES	CAP0捕捉模式选择 0x0: 禁止 0x1: 上升沿捕捉 0x2: 下降沿捕捉 0x3: 双沿	0x0

12.5.12 CAP通道选择寄存器（CAPCHS）

位	符号	描述	复位值
31:17	-	保留	-
16	ECAPS	ECAP捕获通道组选择 0: 选择ECAP00-ECAP02 1: 选择ECAP10-ECAP13	0
15:12	CAP3CHS	CAP3捕捉通道选择 0x0: ECAPx0（x=0或1，由ECAPS决定） 0x1: ECAPx1 0x2: ECAPx2 0x3: ECAPx3 0x4: 禁止 0x5: 禁止 0x8: ACMP0的输出（非事件输出） 0x9: ACMP1的输出（非事件输出） 0xF: CCP1B 其他值: 保留	0x0
11:8	CAP2CHS	CAP2捕捉通道选择 0x0: ECAPx0（x=0或1，由ECAPS决定） 0x1: ECAPx1 0x2: ECAPx2 0x3: ECAPx3 0x4: 禁止 0x5: 禁止 0xF: CCP1A 其他值: 保留	0x0
7:4	CAP1CHS	CAP1捕捉通道选择 0x0: ECAPx0（x=0或1，由ECAPS决定） 0x1: ECAPx1 0x2: ECAPx2 0x3: ECAPx3 0x4: 禁止 0x5: 禁止 0xF: CCP0B 其他值: 保留	0x0
3:0	CAP0CHS	CAP0捕捉通道选择 0x0: ECAPx0（x=0或1，由ECAPS决定） 0x1: ECAPx1 0x2: ECAPx2 0x3: ECAPx3 0x4: 禁止 0x5: 禁止 0xF: CCP0A 其他值: 保留	0x0

12.5.13 CAP数据寄存器（CAP0DATA）

位	符号	描述	复位值
31:16	CAPXDATA	读: 捕获模式2：捕获完成后，存放CAPXDATA =CAP1的捕获值-CAP3的捕获值 其他：- 写: 捕获模式2：- 其他：0x55aa，产生CAPn的捕获操作 写: 其他值，则无效	0x0
15:0	CAP0DATA/ CAPYDATA	读: 捕获模式2：捕获完成后，存放CAPYDATA =CAP1的捕获值-CAP2的捕获值 其他：为CAP0捕捉CCP1计数器的16bit值 写: 无效	0x0

12.5.14 CAP数据寄存器（CAPnDATA）（n=1-3）

位	符号	描述	复位值
31:16	-	读: 无效 写: 0x55aa，产生CAPn的捕获操作 写: 其他值，则无效	0x0
15:0	CAPnDATA	读: 为CAPn捕捉CCP1计数器的16bit值 写: 无效	0x0

第13章 增强型PWM（EPWM）

13.1 概述

增强性PWM模块支持6路PWM发生器，可以配置成相互独立的6路PWM输出（EPWM0-EPWM5），也可以配置成3对分别带有编程死区发生器的互补PWM（EPWM0-EPWM1，EPWM2-EPWM3，EPWM4-EPWM5）。

每一对PWM共用8位预分频器，有6组时钟分频器，提供5种分频系数（1、1/2、1/4、1/8、1/16）。每一路PWM输出有独立的16位计数器进行控制，另外16位的比较器用以调节占空比。6路PWM发生器提供28个中断标志，相关PWM通道的周期或占空比与计数器相符，将产生中断标志，每一路PWM有单独的使能位。

每路PWM可配置成单次模式（产生一个PWM信号周期）或者循环模式（连续输出PWM波形）。

13.2 特性

增强性PWM模块有如下特性：

- ◆ 6 路独立的 16 位 PWM 控制模式。
  - 6 路独立输出：EPWM0、EPWM1、EPWM2、EPWM3、EPWM4、EPWM5；
    - 3组互补PWM对：（EPWM0-EPWM1）、（EPWM2-EPWM3）、（EPWM4-EPWM5），可插入可编程死区时间；
    - 3组同步PWM对：（EPWM0-EPWM1）、（EPWM2-EPWM3）、（EPWM4-EPWM5），每组PWM对引脚同步。
- ◆ 支持组控制，EPWM0，EPWM2，EPWM4 输出同步，EPWM1，EPWM3，EPWM5 输出同步。
- ◆ 单次模式（仅支持边沿对齐）或者自动装载模式。
- ◆ 支持边沿对齐，中心对齐 2 种模式。
- ◆ 中心对齐模式支持对称计数和非对称计数。
- ◆ 互补的 PWM 中，支持可编程死区发生器。
- ◆ 每路 PWM 有独立的极性控制。
- ◆ 故障刹车保护以及恢复功能（软/硬件触发以及软/硬件恢复）。
- ◆ ACMP 模拟比较器可触发硬件刹车保护。
- ◆ PWM 边沿或周期可触发启动 AD 转换。

13.3 功能描述

相关名称说明：

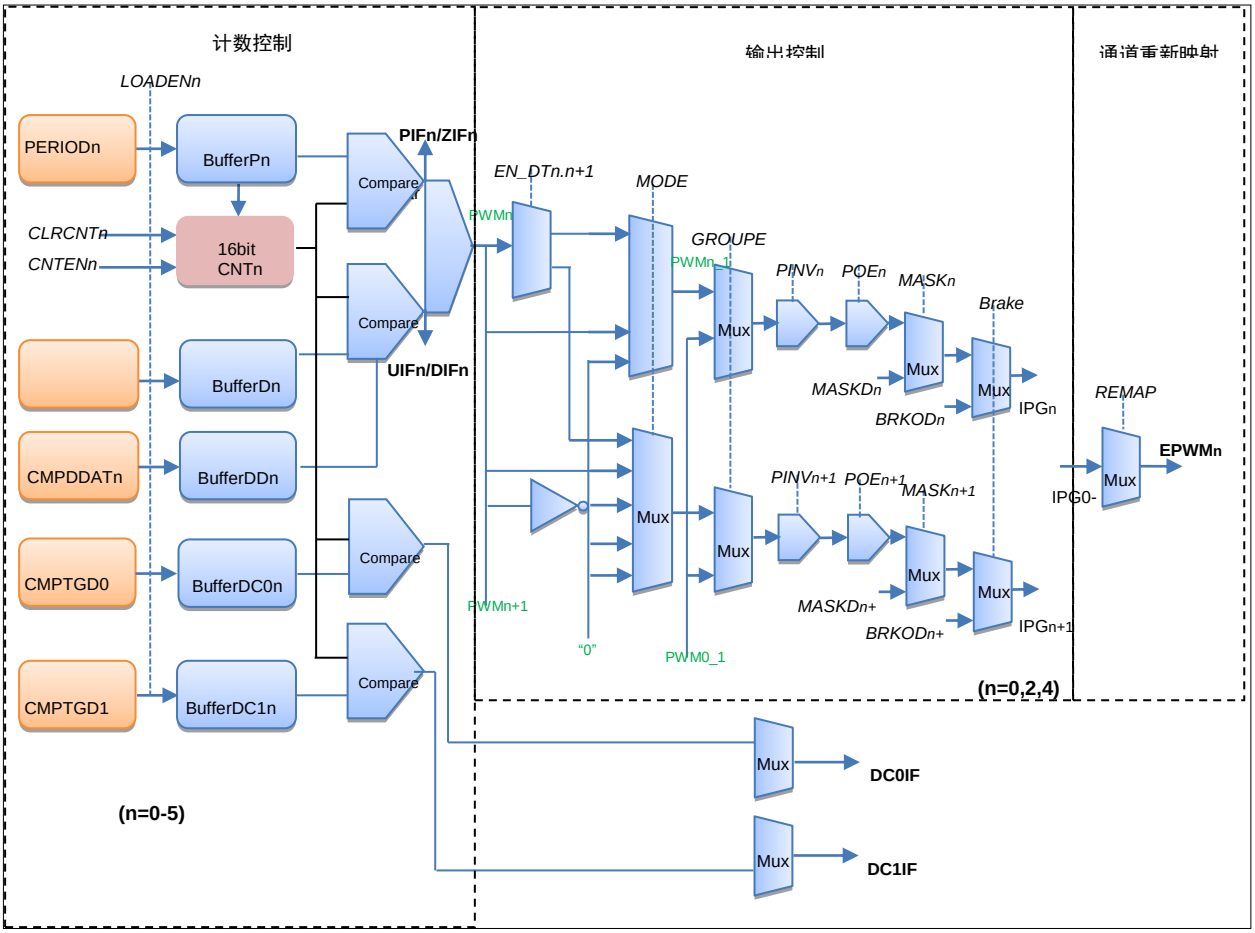
1) 周期点：计数器 CNTn 计数到与周期 PERIODn 相等时，称之为周期点。所产生的中断为 PIFn.
2) 零点：计数器 CNTn 计数到 0 时，称之为零点。所产生的中断为 ZIFn.
3) 向上比较点：计数器 CNTn 加计数到与 CMPDATn 相等时，称之为向上比较点。所产生的中断为 UIFn。边沿对齐计数方式无向上比较点。
4) 向下比较点：计数器 CNTn 减计数到与 CMPDATn 或 CMPDDATn 相等时，称之为向下比较点。所产生的中断为 DIFn.
5) 中点：中点即中间点，中点为中心对齐计数方式时 CNTn 计数到与 CMPDATn 或 CMPDDATn 相等的时刻，由于之后 CNTn 会减计数到 0，所以将该时刻称之为中点，也为周期点。边沿对齐计数方式无中点，但有周期点。

注：

1) 边沿对齐方式时，第一次计数开始时加载周期数据，将产生周期点；其他时刻由于计数器计数到 0 后，需要立即加载周期数据。所以之后的周期点与零点的位置是相同的。该对齐方式存在向下比较点，不存在向上比较点。
2) 中心对齐时，第一次计数开始由0开始向上计数，将产生零点。在计数到周期数据时，将产生周期点 (中点)。零点与中点交替存在。该对齐方式存在向上比较点与向下比较点，对称计数时，向上比较点与向下比较点均由CMPDATn决定；非对称计数时，向上比较点由CMPDATn决定，向下比较点由CMDDATn决定。

13.3.1 结构框图

图13-1：IPGn的信号为EPWMn重映射前的信号。



13.3.2 时钟分频

每一对PWM共用8位预分频器，在预分频之后，每一路PWM可以选择（1，1/2,1/4,1/8,1/16）5种分频比。PWM_CLK = PCLK / (CLKPSCxx + 1) / CLKDIVn, 这里xx可以是01, 23, 45,n=0-5。

13.3.3 独立输出模式

6路EPWM通道输出相互不影响，按照各自的周期/占空比数据运行

13.3.4 互补输出模式

互补输出模式下，6路PWM分为3对，EPWM0与EPWM1成1对，EPWM2与EPWM3成1对，EPWM4与EPWM5成1对。共有3对PWM。

EPWM0-EPWM1按EPWM0的周期/占空比数据运行，EPWM0与EPWM1波形反相。

EPWM2-EPWM3按EPWM2的周期/占空比数据运行，EPWM2与EPWM3波形反相。

EPWM4-EPWM5按EPWM4的周期/占空比数据运行，EPWM4与EPWM5波形反相。

在此模式下，EPWM1/EPWM3/EPWM5输出与自己的相关运行数据寄存器无关，但输出控制仍然有效。如输出使能，掩码，刹车等控制。

互补模式下支持死区延时控制。

13.3.5 同步输出模式

同步输出模式下，6路PWM分为3对，EPWM0与EPWM1成1对，EPWM2与EPWM3成1对，EPWM4与EPWM5成1对。共有3对PWM。

EPWM0-EPWM1按EPWM0的周期/占空比数据运行，EPWM0与EPWM1波形同相。

EPWM2-EPWM3按EPWM2的周期/占空比数据运行，EPWM2与EPWM3波形同相.

EPWM4-EPWM5按EPWM4的周期/占空比数据运行，EPWM4与EPWM5波形同相。

在此模式下，EPWM1/EPWM3/EPWM5输出与自己的相关运行数据寄存器无关，但输出控制仍然有效。如输出使能，掩码，刹车等控制。

13.3.6 成组输出模式

GROUPEN=1使能成组功能，6路PWM分为2组，EPWM0，EPWM2，EPWM4为1组，EPWM1，EPWM3，EPWM5为1组。

EPWM0-EPWM2-EPWM4按EPWM0的周期/占空比数据运行，3个通道波形同相。

EPWM1-EPWM3-EPWM5按EPWM1的周期/占空比数据运行，3个通道波形同相。

成组功能打开时，EPWM2/EPWM4/EPWM3/EPWM5输出与自己的相关运行数据寄存器无关，但输出控制仍然有效。如输出使能，掩码，刹车等控制。

13.3.7 加载更新模式

计数器加载模式有两种：单次模式(One-shot)与连续模式(自动加载模式)

单次模式：

周期占空比相关数据在计数器开始加载一次，输出的PWM周期与加载方式有关。

LOADTYPn=0时，边沿对齐为1个周期，中心对齐为0.5个周期。

LOADTYPn=1时，边沿对齐为2个周期，中心对齐为1个周期。

LOADTYPn=2时，边沿对齐为3个周期，中心对齐为1.5个周期。

LOADTYPn=3时，边沿对齐为4个周期，中心对齐为2个周期。

连续模式：

周期占空比数据在PWM周期内零点与中点自动加载。中点加载只存在中心对齐计数模式下。

边沿对齐计数模式下，产生零点的同时也会产生周期点，此时计数比较电路会重新加载CMPDATn/PERIODn/CMPTGD0/CMPTGD1的值。

中心对齐计数模式下，中点和零点均会自动加载相关寄存器的值。这样的结构支持前半波形周期占空比与后半波形周期占空比设置不一致，当周期占空比相关寄存器值未改变时，则保持一致。

由于EPWM存在双缓存结构，在EPWM运行的过程中，改变相关运行寄存器：CMPDATn/CMPDDATn/PERIODn/CMPTGD0/CMPTGD1的值，PWM输出波形不会立即改变，只有在零点或周期点时这些寄存器的值才会加载到相应的缓存中。

这样的结构在改变周期占空比数据后，不会立即改变当前PWM周期或半周期内的输出波形，在下个周期或半周期内PWM波形才会做出相应的变化。即任何PWM相关数据的改变不会影响当前一个完整PWM周期或半周期。

在高速的应用中，有可能会出现加载点已经到来，但写入运行寄存器的操作还未完成。此时不期望出现部分运行数据已经加载，另外一部分运行数据没有加载的情况。

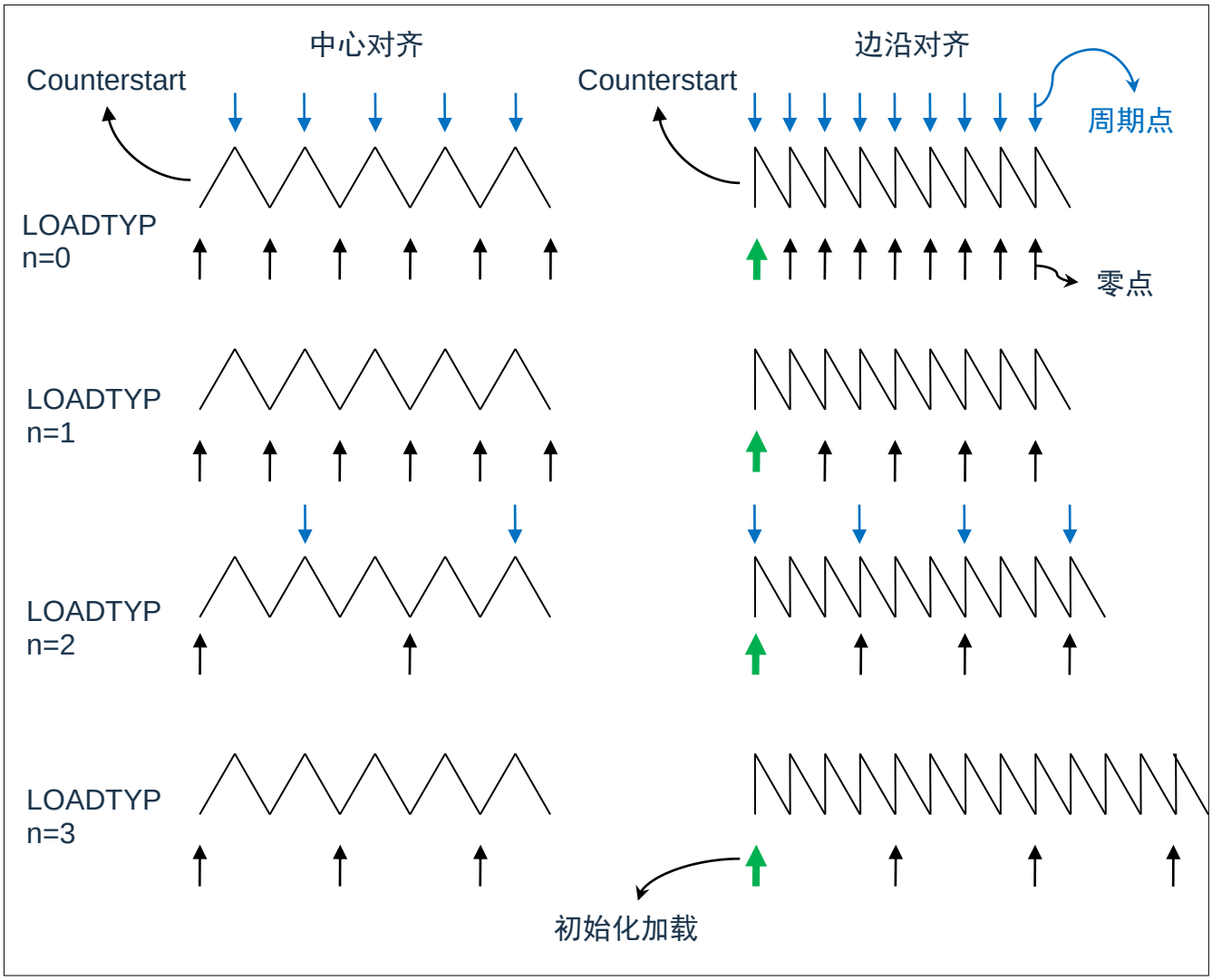
针对该高速应用情况。该EPWM模块提供了加载使能位，改变相关运行寄存器后，需要将加载使能位LOADENn置1，加载完毕后LOADENn位自动清零。另外可以读取该位来判断是否将相关寄存器的值加载到实际电路中。如果LOADENn=0，则表示已经加载，将影响正在输出的PWM波形；如果LOADENn=1，则表示还未加载，当前的PWM波形还未会发生变化，将在下一个加载点才会加载之前改变的寄存器的值。如果再次改变相关运行寄存器的值，也需重新将LOADENn置1。

默认条件下，PWM在零点与周期点均会加载相关寄存器的运行数据，以及产生零点与周期点中断。为了适应更灵活应用需求。PWM支持不同方式的加载方式与零点/周期点中断产生方式。

在寄存器EPWMCON3中LOADTYPn(0-5)可设置加载方式与零点/周期点的中断方式：

LOADTYEn	中心对齐加载	边沿对齐加载
00	每个零点与周期点均加载与产生零点与周期点中断标志	每个零点或周期点均加载与产生零点与周期点中断标志
01	每个零点加载与产生零点中断标志	每2个零点加载与产生零点中断标志
10	第一个零点与下一个周期点交替加载与产生零点与周期点中断标志	每3个零点或周期点加载与产生零点与周期点中断标志
11	每两个零点加载与产生相关零点中断标志	每4零点加载与产生零点中断标志

图13-2：PWM周期/占空比加载更新框图



13.3.8 边沿对齐计数模式

边沿对齐模式下，计数方式为向下计数，即减1计数。16位PWM计数器CNTn在每个周期开始向下计数，与锁存CMPDATn值进行比较，当CNTn=CMPDATn时EPWMn输出高电平，CMPnDIF置1。CNTn继续向下计数至0，此时EPWMn将输出低电平，当前CMPDATn和PERIODn在PWMnCNTM=1的情况会重新加载，PIF周期中断标志置位。

边沿对齐相关参数如下：

高电平时间=（CMPDATn+1）×Tpwm

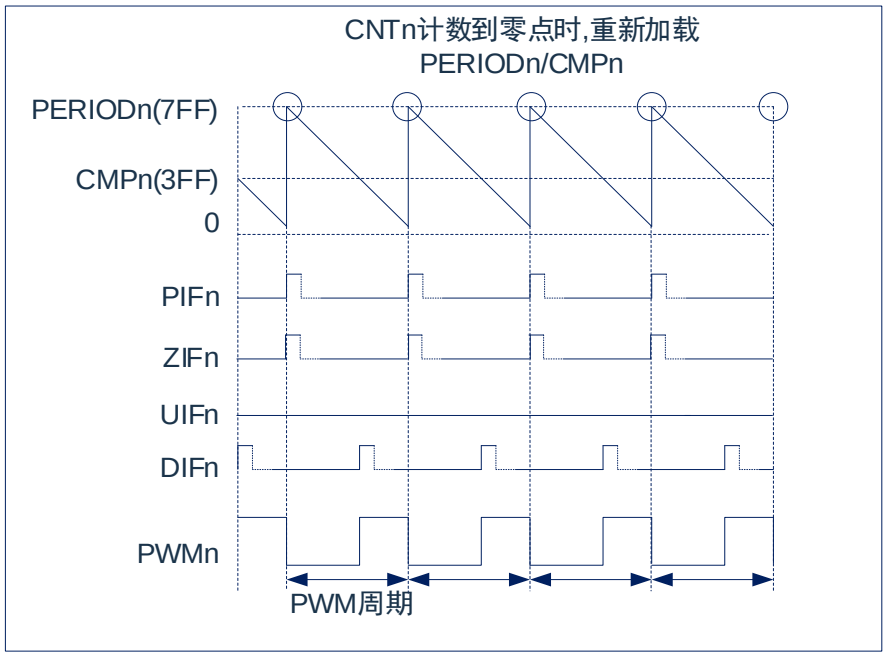
周期 = （PERIODn+1）×Tpwm

$$占空比 = \frac{CMPDATn + 1}{PERIODn + 1}$$

若CMPDATn>PERIODn，占空比为100%，EPWMn通道一直为高。且不会产生向下比较中断。

若CMPDATn=0，则占空比为0%。

图13-3：边沿对齐模式波形图



13.3.9 中心对齐计数模式

中心对齐模式下，计数方式为先向上计数再向下计数。

中心对齐模式按对称方式又分为两种：对称计数方式与非对称计数方式。

对称计数方式(ASYMEN=0)占空比由CMPDATAn决定。

非对称计数方式(ASYMEN=1)占空比由CMPDATAn与CMPDDATn共同决定。

中心对齐对称计数方式下，16位PWM计数器CNTn从0开始向上计数，当CNTn=CMPDATn时，EPWMn输出高电平，之后CNTn继续向上计数至与PERIODn相等，然后CNTn开始向下计数，在向下计数的过程中CNTn=CMPDATn时，EPWMn输出低电平，之后继续向下计数至0。

高电平时间 = （PERIODn×2-CMPDATn×2-1）×Tpwm

周期 = （PERIODn）×2×Tpwm

$$占空比 = \frac{PERIODn \times 2 - CMPDATn \times 2 - 1}{PERIODn \times 2}$$

若CMPDATn>=PERIODn，占空比为0%，EPWMn通道一直为低，且不会产生向上比较中断与向下比较中断。

若PERIODn=0，占空比为0%，EPWMn通道一直为低，且CNTn使能时零点中断与周期点中断一直存在。

若CMPDATn=0，则占空比为100%。

图13-4：中心对齐模式对称计数波形图

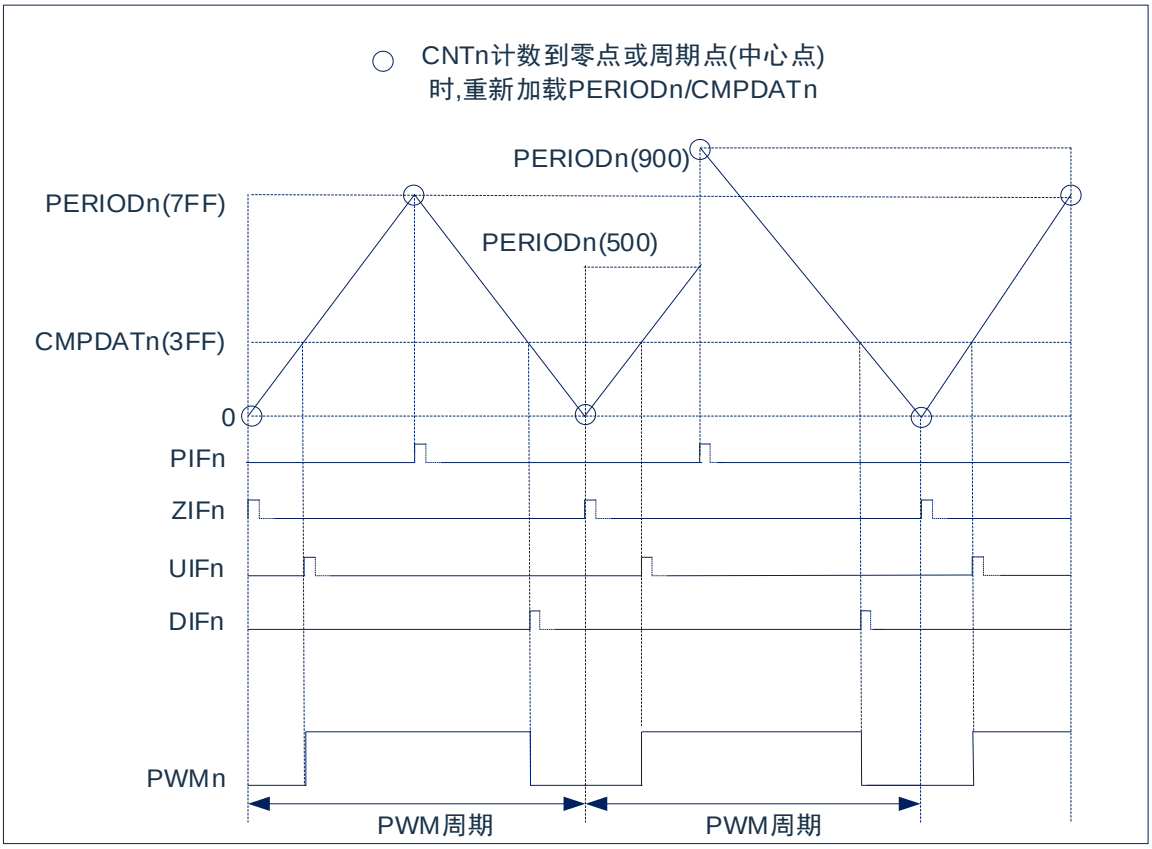
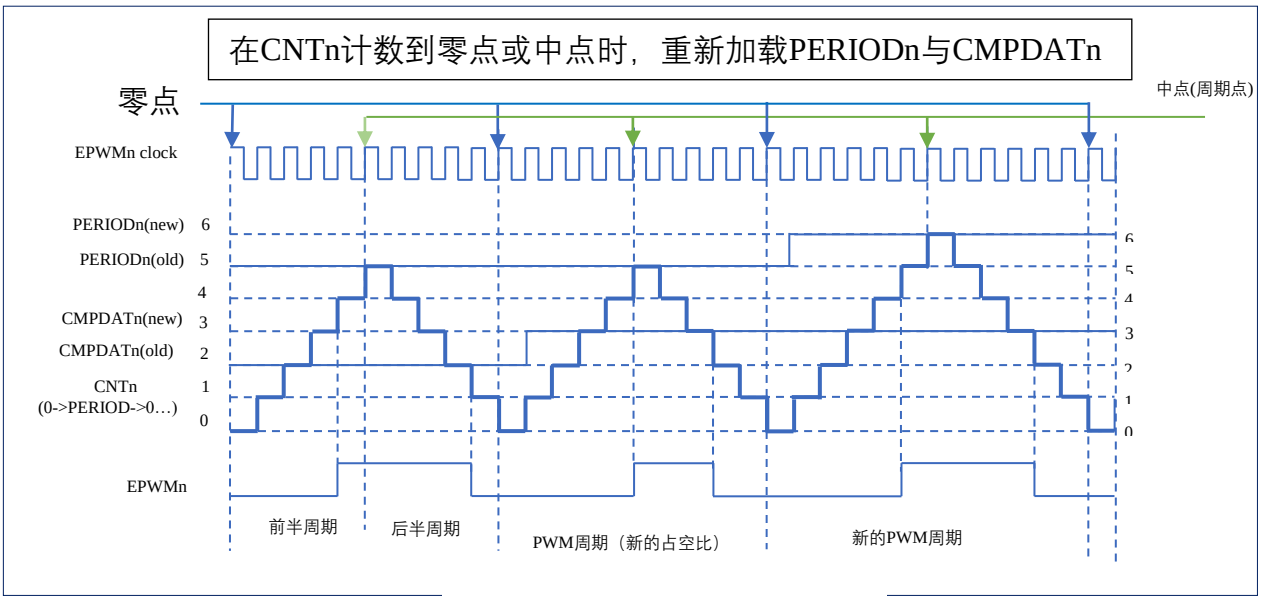


图13-5：中心对齐计数器波形（对称计数）



中心对齐非对称计数方式下，16位PWM计数器CNTn从0开始向上计数，当CNTn=CMPDATn时，EPWMn输出高电平，之后CNTn继续向上计数至与PERIODn相等，然后CNTn开始向下计数，在向下计数的过程中CNTn=CMPDDATn时，EPWMn输出低电平，之后继续向下计数至0。开启非对称计数方式需要将ASYMEN置1，非对称计数方式下可实现精确的中心对齐波形。

中心对齐非对称计数方式相关参数如下：

高电平时间 = （PERIODn×2-CMPDDATn-CMPDATn-1）×Tpwm

周期 = （PERIODn）×2×Tpwm

$$占空比 = \frac{PERIODn\times2-CMPDDATn-CMPDATn-1}{PERIODn\times2}，（CMPDATAn<PERIODn,CMPDDATn<PERIODn）$$

$$占空比 = \frac{PERIODn-CMPDDATn-1}{PERIODn\times2}，（CMPDATAn\geq PERIODn,CMPDDATn<PERIODn）$$

$$占空比 = \frac{PERIODn-CMPDATn}{PERIODn\times2}，（CMPDATAn<PERIODn,CMPDDATn\geq PERIODn）$$

占空比=0%，（CMPDATAn≥PERIODn,CMPDDATn≥PERIODn）

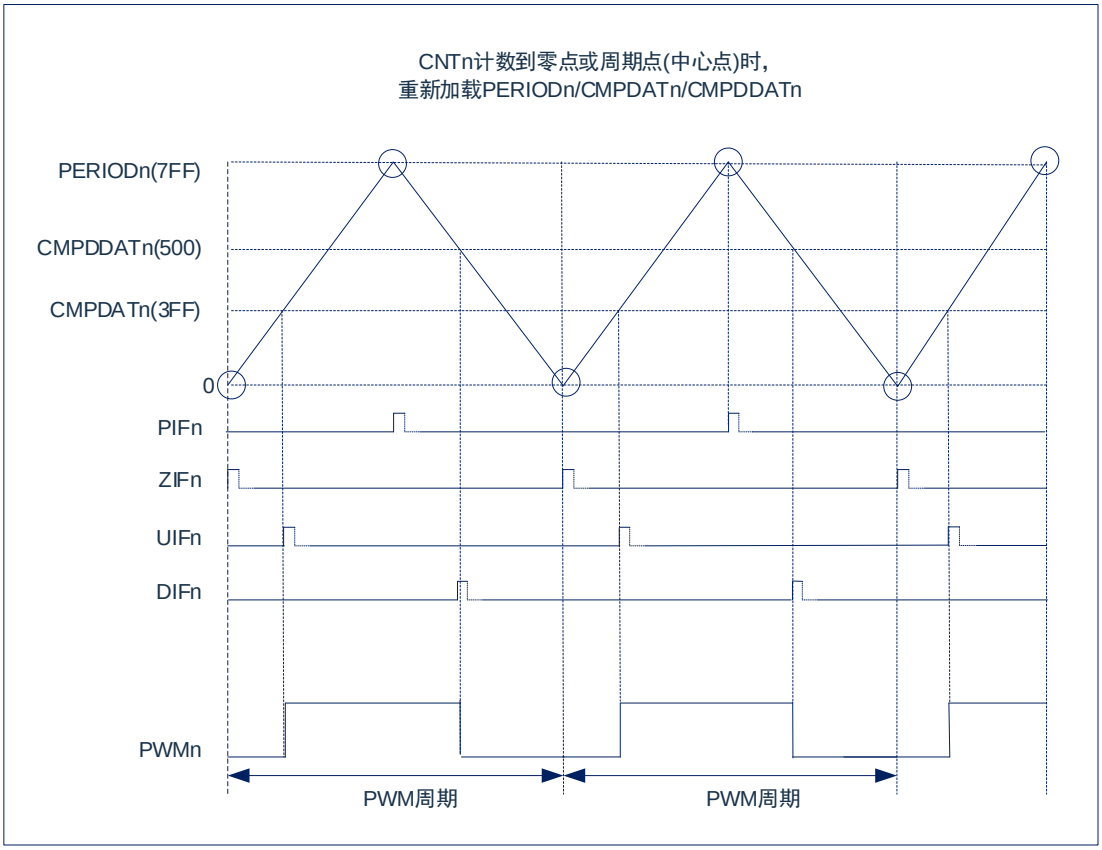
CMPDATAn>=PERIODn时不会产生向上比较中断。

CMPDDATAn>=PERIODn时不会产生向下比较中断。

若PERIODn=0，占空比为0%，EPWMn通道一直为低，且CNTn使能时零点中断与周期点中断一直存在。

若CMPDATn=0与CMDATDn=0，则占空比为100%。

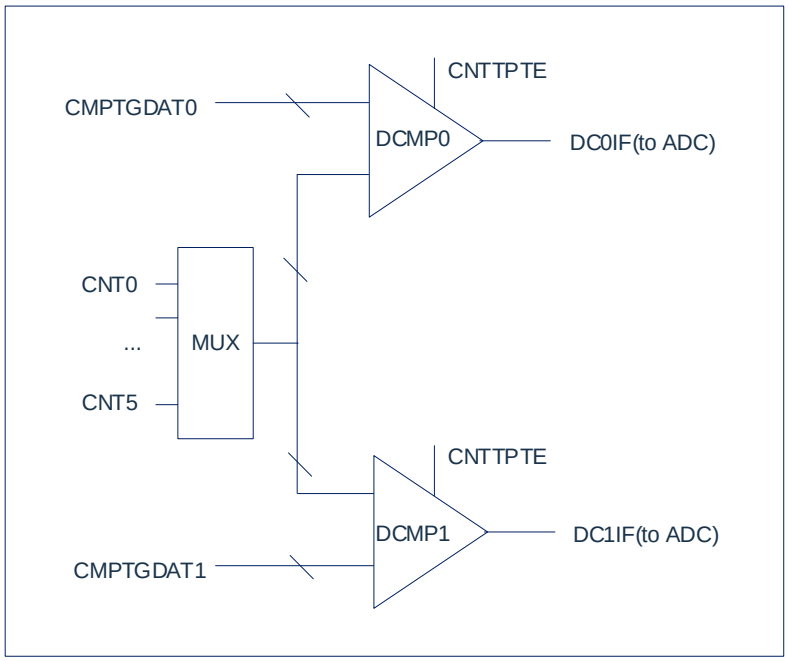
图13-6：中心对齐模式非对称计数波形图



13.3.10 独立计数器比较功能

在PWMn通道计数器(CNTn)计数器期间，提供了两个数字比较器，计数器CNTn与预设的值进行比较，若计数器的值与预设值相等则可产生中断信号或者触发ADC启动。该功能不影响PWM的输出。

图13-7：独立计数器比较功能

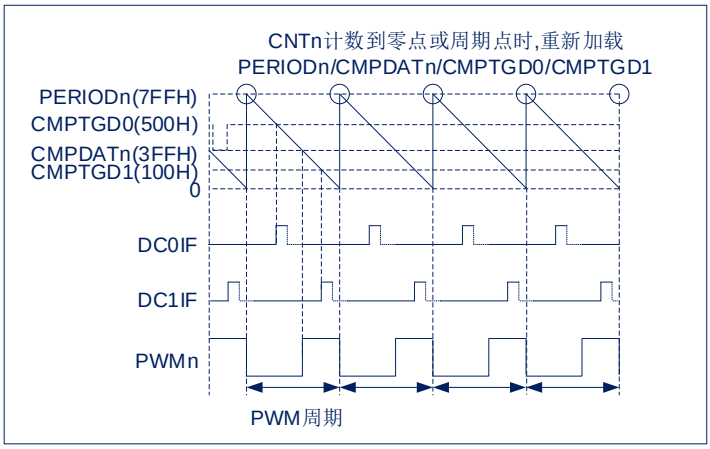


数字比较器0比较计数器CNTn的值与CMPTGDAT0的值。若相等则产生中断标志位DC0IF，CMPTGD0[10:8]选择PWM0-5通道计数器之一与CMPTGDAT0进行比较。

数字比较器1比较计数器CNTn的值与CMPTGDAT1的值。若相等则产生中断标志位DC1IF.，CMPTGD1[10:8]选择PWM0-5通道计数器之一与CMPTGDAT1进行比较

1) 边沿对齐模式，数字比较器的工作方式：

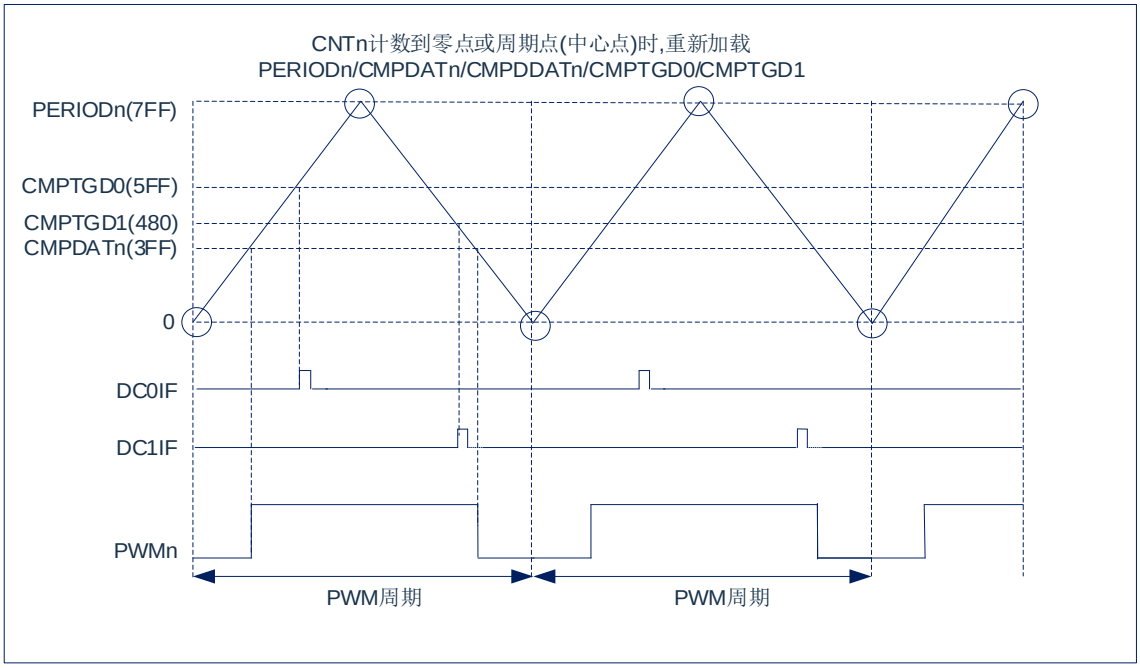
图13-8：边沿对齐模式，数字比较器的工作方式



在边沿计数模式下，数字比较器0/1可设置为在任意计数时刻产生比较中断。

2) 中心对齐模式，数字比较器的工作方式：

图13-9：中心对齐模式，数字比较器的工作方式



在中心对齐计数模式下，数字比较器0/1可各自设置为在向上或向下计数模式比较触发。即都可在前半周期触发或后半周期触发，也可一个在前半周期触发，另一个在后半周期触发。由CMPTGD0[19]位CMPTGDSn决定。

13.3.11 可编程死区发生器

6通道PWM可设置为3组互补对。在互补输出模式下，PWM1，PWM3，PWM5的周期与占空比分别由PWM0，PWM2，PWM4相关寄存器决定，同时死区延时寄存器也可影响PWM互补对的占空比。此时除了对应的输出使能控制位（PWMnOE），PWM1/PWM3/PWM5输出波形不再受自己的寄存器控制。

在互补模式下，每组互补PWM对均支持插入死区延时，插入的死区时间如下：

PWM0/1死区时间：（PWM01DT[9:0]+1）*TPWM0

PWM2/3死区时间：（PWM23DT[9:0]+1）*TPWM2

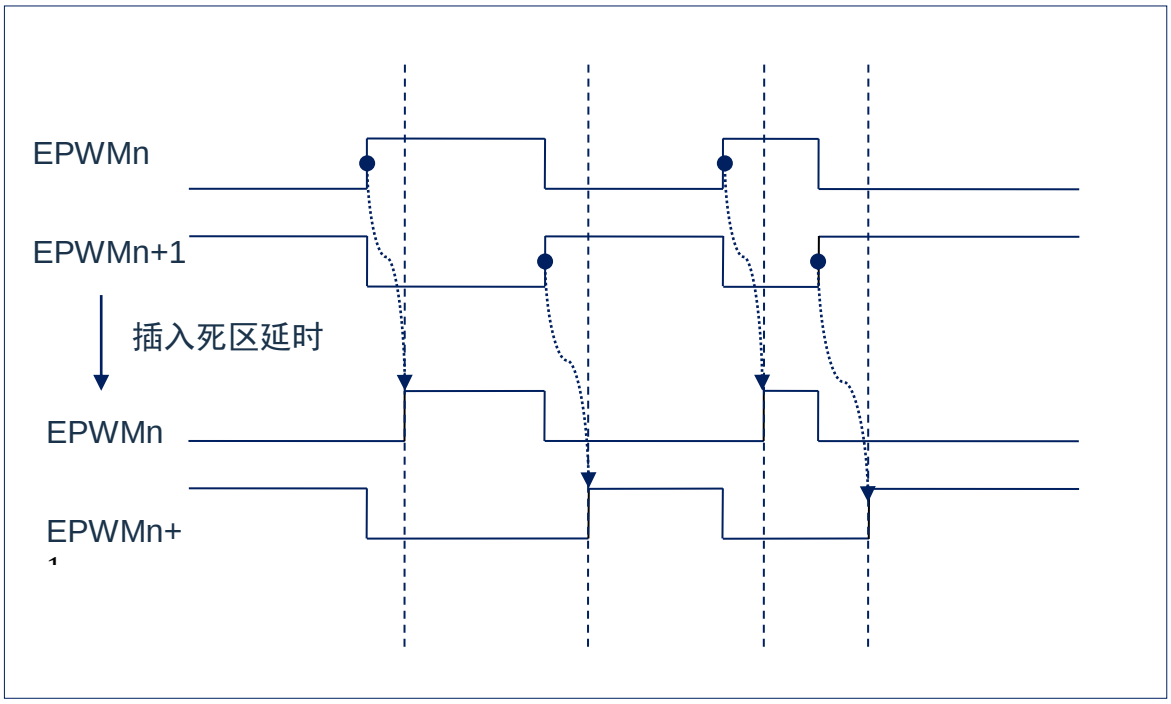
PWM4/5死区时间：（PWM45DT[9:0]+1）*TPWM4

TPWM0/TPWM2/TPWM4分别为PWM0/PWM2/PWM4的时钟源周期。

死区时间可设置的范围：0.021us～21us（Fpwmn=48MHz）

输出模式不影响计数器的模式，所以中心对齐与边沿对齐均支持互补输出模式。

图13-10：中心对齐与边沿对齐均支持互补输出模式



13.3.12 掩码及掩码预设功能

EPWM支持掩码功能。EPWM0-EPWM5每个通道有单独的控制，EPWMn对应的控制位为MASKENn，MASKDn（在寄存器MASK中）。

当MASKENn=0时，EPWMn通道输出正常的PWM波形；

当MASKENn=1时，EPWMn通道输出MASKDn的数据；

掩码功能的控制寄存器MASK也支持自动加载预设值的功能。开启该功能需要将输出控制寄存器POEN的MASKLE位置1，允许MASK自动加载MASKNXT寄存器的值，同时禁止写MASK寄存器。

加载时刻在POEN中MASKLS<2:0>设置，可选择与EPWM0-EPWM5其中之一的周期/占空比加载时刻（加载点）相同。

13.3.13 霍尔传感器接口功能

EPWM考虑与霍尔传感器的接口。内部包含一个HALL位置状态检测电路，该电路检测CCP0/1模块中内部捕获通道CAP0，CAP1，CAP2经过滤波后的电平。

检测电路内部处理后存在状态称之为HALLST：

HALLST存在8种状态，与HALL位置状态对应关系如下：

HALLST	对应的状态
000	HALL检测电路未启动或初始状态
001	{CAP2-CAP0}=001
010	{CAP2-CAP0}=010
011	{CAP2-CAP0}=011
100	{CAP2-CAP0}=100
101	{CAP2-CAP0}=101
110	{CAP2-CAP0}=110
111	{CAP2-CAP0}变化过程中出现错误的状态或者出现了错误的序列

HALLST的值可从MASKNXT寄存器中读出，在任何时间可判断HALL位置或序列状态，

HALL状态检测序列支持以下两种({CAP2，CAP1，CAP0}出现的顺序)：

- -6-2-3-1-5-4-6-
- -6-4-5-1-3-2-6-

如出现其他序列则认为出现了错误，HALLST将进入111的状态后停止检测。同时将产生中断标志HALLIF。如果需要重新启动HALL检测电路，需要将MASKNXT寄存器中的HALLCLR位写1，HALLST即从111的状态进入000的初始状态重新启动检测电路。

HALL检测电路提供了可与掩码自动加载相关功能。该功能不需要软件介入便可以控制EPWM的输出通道波形。

HALLST每个有效状态对应一个掩码预设缓存，一共有7个掩码预设缓存：

HALLST(HALLEN=1)	对应的掩码预设缓存
000	掩码预设缓存7
001	掩码预设缓存1
010	掩码预设缓存2
011	掩码预设缓存3
100	掩码预设缓存4
101	掩码预设缓存5
110	掩码预设缓存6
111	掩码预设缓存7
HALLEN=0	掩码预设缓存0

如果开启掩码自动加载预设值的功能，则在相应的状态下，且在选择的加载点时刻，对应的掩码预设缓存中的数据将加载到MASK寄存器中。例如：

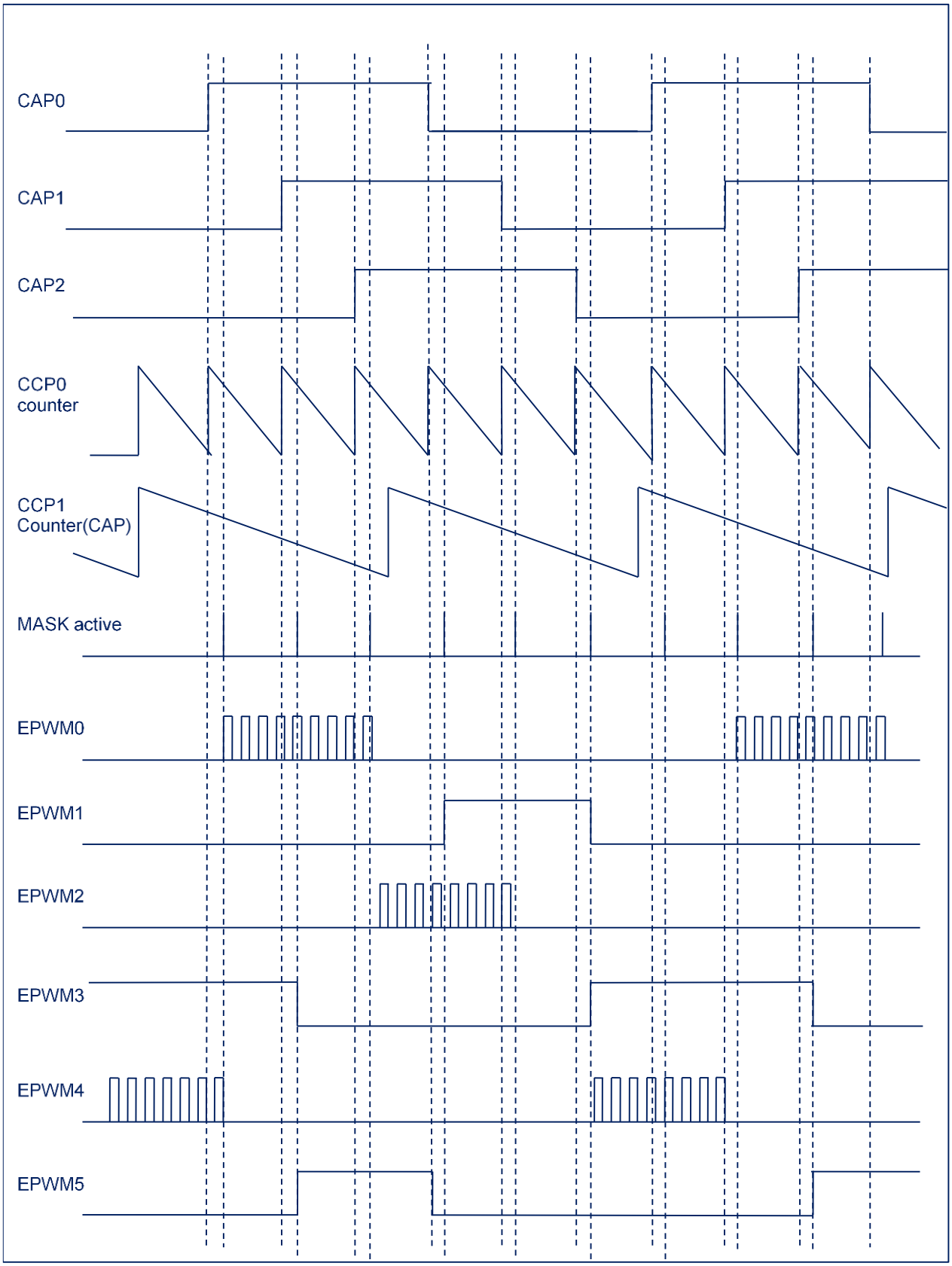
HALLST中位置状态从000改变到001时，在进入001状态第一个加载点，掩码预设缓存1的数据则加载到MASK寄存器中。

之后HALLST中位置状态从001改变到101时，在进入101状态第一个加载点，掩码预设缓存5的数据则加载到MASK寄存器中。

如果出现错误的序列，如CAP2-CAP0输入从101改变到010时，这不是正确的序列，HALLST中位置状态则从101改变到111，并将中断标志位HALLIF置1。在进入111状态第一个加载点，掩码预设缓存7的数据则加载到MASK寄存器中。

初始状态下，掩码预设缓存7的数据在加载点加载到MASK寄存器中。

图13-11：HALL检测时序示例（不代表实际运行波形）



13.3.14 故障保护功能（刹车及恢复功能）

EPWM支持故障保护功能，BKODn控制6个通道的刹车阀值。故障保护功能由BRKCTL寄存器控制。

EPWM故障保护触发来源有：

电平触发源：

1) 外部 BKIN 电平信号（高电平或低电平）
2) 软件刹车信号（SWBRK 位置为 1）
3) 模拟比较器 0 的输出（输出高或输出低）
4) 模拟比较器 1 的输出（输出高或输出低）脉冲触发源：
5) 外部 BKIN 的边沿信号（上升沿或下降沿）
6) 模拟比较器 0 的输出事件（上升沿或下降沿或双沿）
7) 模拟比较器 1 的输出事件（上升沿或下降沿或双沿）
8) ADC 结果比较器 0 事件（结果比较事件）

故障中断标志位BRKIF（软件清0）：

检测到有效的刹车触发源信号后，故障中断标志BRKIF置1，需由软件清零。

故障信号标志位BRKAF（只读）：

故障信号标志位BRKAF置1，刹车信号撤销后，BRKAF自动清零0。BRKAF为只读位。

故障保护输出状态标志位BRKOSF（只读）：

BRKOSF=1，表示EPWMn通道输出BRKODn数据状态；

BRKOSF=0，表示EPWMn为正常输出状态。

指示EPWM输出在刹车状态还是正常状态。检测到有效的刹车信号时BRKOSF将置1。软件恢复模式下，执行刹车状态清除操作（BRKCLR=1）会影响到该位的状态。

故障保护模式可分为4种，以适应不同故障保护场合的需要。

BRKMS	故障保护模式
00	停止模式（软件恢复）
01	暂停模式（软件恢复）
10	恢复模式（硬件恢复）
11	延时恢复模式（硬件恢复）

注：故障中断标志（BRKIF）与恢复功能无关，仅代表产生了刹车信号。故障中断标志也支持累加功能。

停止模式：

产生故障保护及故障中断标志，将CNTENn位清零，停止计数器运行。恢复输出需要刹车信号撤销，且执行故障状态清除操作（BRKCLR=1），然后重新将CNTENn置1。

暂停模式：

产生故障保护及故障中断标志，但计数器继续运行。恢复输出需要刹车信号撤销，执行故障状态清除操作（BRKCLR=1）后，在最近一次的加载更新点恢复正常输出。

恢复模式：

产生故障保护及故障中断标志，但计数器继续运行。刹车信号撤销后，自动在最近一次的加载更新点恢复正常输出。不需要执行故障状态清除操作。

需要注意区分刹车信号是脉冲信号还是电平信号：如果刹车源为电平信号，则需要等待刹车撤销后才能恢复输出；如果为脉冲信号，则EPWM输出在出发刹车之后最近一次加载更新点恢复输出，除非期间再次产生刹

车脉冲信号。

延时恢复模式：

产生故障保护及故障中断标志，但计数器继续运行。刹车信号撤销后延时一段时间EPWM在最近一次的加载更新点恢复正常输出。不需要执行故障状态清除操作。

延时时间可以自由设置，BRKRDT的低16位RDT控制延时时间。延时时间如下：

$$Tdelay = RDT*TAPBCLK$$

需要注意区分刹车信号是脉冲信号还是电平信号：如果刹车源为电平信号，则需要等待刹车撤销后才能恢复输出；如果为脉冲信号，则EPWM输出等待完成延时后最近一次加载更新点恢复输出，除非期间再次产生刹车脉冲信号。

产生刹车保护后，EPWMn通道输出BRKODn中数据，每个通道可单独设置输出高/低电平。

13.3.15 调试模式下的输出状态

在调试模式下，CPU 的状态有运行状态与暂停状态。一种为正常运行状态；一种状态为执行 STOP 命令/运行到断点/单步之后的暂停状态。

其中暂停状态下 EPWMn 工作时（POEn=1）输出状态可以通过寄存器 CON 中的 HALTMS 位来配置。

HALTMS=0 时，EPWMn 的输出状态在暂停时为正常输出。

HALTMS=1 时，EPWMn 的输出状态在暂停时输出刹车数据，但此时不会产生故障相关的标志位。EPWMn 的计数器将继续运行，在恢复运行状态后最近的加载更新点恢复 EPWMn 输出。

需要注意的是，在调试模式下暂停时，EPWMn 的相关运行数据寄存器的值不会自动改变，还会保持之前的状态。

13.3.16 输出通道重映射功能

输出通道重映射功能可满足应用中更加灵活的排版需求。芯片管脚分布图中 EPWM0-EPWM5 的管脚默认为对应的 PWM 通道输出。也可通过输出通道重映射功能来重新配置所需要的通道。

EPWM0-EPWM5 默认对应的内部通道分别为 IPG0-IPG5，通过 EPWM 输出通道重映射寄存器 POREMAP 可将 IPG0-IPG5 任意一路通道重新分配到 EPWMn(n=0-5)。输出通道重映射功能仅对端口输出通道重新分配，其内部控制及中断不会重映射。

13.3.17 EPWM配置过程

- 在 LOCK 寄存器中写入 0x55 使能 EPWM 寄存器操作
- 配置 EPWM 时钟分频，设置预分频比和独立分频比
- 选择模式，独立模式或互补模式
- 设置 EPWM 周期和占空比
- 设置 EPWM 输出极性
- 使能 EPWM 计数器
- 配置相关 IO 口为 EPWM 功能口
- 使能相关 EPWM 通道输出
- 在 LOCK 寄存器写入 0x00，避免 EPWM 相关寄存器被误操作，直至下一次需要操作 EPWM 相关寄存器时才重新使能

13.3.18 中断

EPWM单元有八个中断源：

- ZIFn—EPWM 计数器计数为零时产生的中断标志
- UIFn—EPWM 计数器向上计数到 CMPDATn 中断标志
- PIFn—EPWM 计数器边沿对齐计数中断标志，中心对齐计数中断标志
- DIFn—EPWM 计数器向下计数到 CMPDATn/CMPDDATn 中断标志
- DC0IF—EPWM 计数器计数到与 CMPTGD0 相等的中断标志
- DC1IF—EPWM 计数器计数到与 CMPTGD1 相等的中断标志
- HALLIF—霍尔状态错误中断标志位
- BRKIF—故障中断标志位

所有的中断标志都由硬件置位，且必须通过软件清零。

13.4 寄存器映射

(EPWM 基地址 = 0x4006_4200)

RO：只读，WO：只写，R/W：读写

寄存器	偏移量	读/写	描述	复位值
CLKPSC(P1B)	0x000	R/W	EPWM预分频寄存器	0x0
CLKDIV(P1B)	0x004	R/W	EPWM时钟选择寄存器	0x0
CON(P1B)	0x008	R/W	EPWM控制寄存器	0x0
CON2(P1B)	0x00C	R/W	EPWM控制寄存器2	0x0
CON3(P1B)	0x010	R/W	EPWM控制寄存器3	0x0
PERIOD0(P1A)	0x014	R/W	EPWM周期寄存器0	0x0
PERIOD1(P1A)	0x018	R/W	EPWM周期寄存器1	0x0
PERIOD2(P1A)	0x01C	R/W	EPWM周期寄存器2	0x0
PERIOD3(P1A)	0x020	R/W	EPWM周期寄存器3	0x0
PERIOD4(P1A)	0x024	R/W	EPWM周期寄存器4	0x0
PERIOD5(P1A)	0x028	R/W	EPWM周期寄存器5	0x0
CMPDAT0(P1A)	0x02C	R/W	EPWM比较寄存器0	0x0
CMPDAT1(P1A)	0x030	R/W	EPWM比较寄存器1	0x0
CMPDAT2(P1A)	0x034	R/W	EPWM比较寄存器2	0x0
CMPDAT3(P1A)	0x038	R/W	EPWM比较寄存器3	0x0
CMPDAT4(P1A)	0x03C	R/W	EPWM比较寄存器4	0x0
CMPDAT5(P1A)	0x040	R/W	EPWM比较寄存器5	0x0
POREMAP(P1B)	0x044	R/W	EPWM输出通道重映射寄存器	0x543210
POEN(P1B)	0x048	R/W	EPWM输出控制寄存器	0x0
BRKCTL(P1B)	0x04C	R/W	EPWM故障保护控制寄存器	0x0
DTCTL(P1B)	0x050	R/W	EPWM死区长度寄存器	0x0
MASK(P1B)	0x054	R/W	EPWM输出掩码寄存器	0x0
MASKNXT(P1B)	0x058	R/W	EPWM输出掩码预设寄存器	0x0
CMPTGD0(P1B)	0x05c	R/W	EPWM计数器比较寄存器0	0x0
CMPTGD1(P1B)	0x060	R/W	EPWM计数器比较寄存器1	0x0
IMSC(P1B)	0x064	R/W	EPWM中断使能寄存器	0x0
RIS	0x068	RO	EPWM中断源状态寄存器	0x0
MIS	0x06c	RO	EPWM已使能中断状态寄存器	0x0
ICLR	0x070	WO	EPWM中断清零寄存器	0x0
IFA(P1B)	0x074	R/W	EPWM中断累加控制寄存器	0x0
LOCK	0x078	R/W	EPWM写使能控制寄存器	0x0
BRKRDT(P1B)	0x07C	R/W	EPWM故障保护恢复延时寄存器	0x0

注：

(P1A/P1B)标注的寄存器为被保护的寄存器。

(P1A)：LOCK==55H 或 AAH 时，标注的寄存器允许写入；=其他值，禁止写入。

(P1B)：LOCK==55H时，标注的寄存器允许写入；=其他值，禁止写入。

13.5 寄存器说明

13.5.1 EPWM预分频寄存器(CLKPSC)

位	符号	描述	复位值
31:24	-	保留	-
23:16	CLKPSC45	EPWM计数器4和5 时钟预分频 CLK_PSC45 = PCLK/(CLKPSC45+1) 如果CLKPSC45=0，预分频器没有时钟输出，CLKDIVn位若选择跟PSC相关的时钟时，计数器不工作	0x0
15:8	CLKPSC23	EPWM计数器2和3 时钟预分频 CLK_PSC23 = PCLK/(CLKPSC23+1) 如果CLKPSC23=0，预分频器没有时钟输出，CLKDIVn位若选择跟PSC相关的时钟时，计数器不工作	0x0
7:0	CLKPSC01	EPWM计数器0和1 时钟预分频 CLK_PSC01 = PCLK/(CLKPSC01+1) 如果CLKPSC01=0，预分频器没有时钟输出，CLKDIVn位若选择跟PSC相关的时钟时，计数器不工作	0x0

13.5.2 EPWM时钟选择寄存器(CLKDIV)

位	符号	描述	复位值
31:23	-	保留	-
22:20	CLKDIV5	计数器5时钟分频选择 000: CLK_PSC45/2 001: CLK_PSC45/4 010: CLK_PSC45/8 011: CLK_PSC45/16 100: CLK_PSC45/1 其他值: PCLK	0x0
19	-	保留	-
18:16	CLKDIV4	计数器4时钟分频选择 000: CLK_PSC45/2 001: CLK_PSC45/4 010: CLK_PSC45/8 011: CLK_PSC45/16 100: CLK_PSC45/1 其他值: PCLK	0x0
15	-	保留	-
14:12	CLKDIV3	计数器3时钟分频选择 000: CLK_PSC23/2 001: CLK_PSC23/4 010: CLK_PSC23/8 011: CLK_PSC23/16 100: CLK_PSC23/1 其他值: PCLK	0x0
11	-	保留	-
10:8	CLKDIV2	计数器2时钟分频选择 000: CLK_PSC23/2 001: CLK_PSC23/4 010: CLK_PSC23/8 011: CLK_PSC23/16 100: CLK_PSC23/1 其他值: PCLK	0x0
7	-	保留	-
6:4	CLKDIV1	计数器1时钟分频选择 000: CLK_PSC01/2 001: CLK_PSC01/4 010: CLK_PSC01/8 011: CLK_PSC01/16 100: CLK_PSC01/1 其他值: PCLK	0x0
3	-	保留	-
2:0	CLKDIV0	计数器0时钟分频选择 000: CLK_PSC01/2 001: CLK_PSC01/4	0x0

		010: CLK_PSC01/8 011: CLK_PSC01/16 100: CLK_PSC01/1 其他值: PCLK	

13.5.3 EPWM控制寄存器(CON)

位	符号	描述	复位值
31:27	-	保留	-
26	HALTMS	HALT（调试暂停）时EPWMn通道状态控制位 （若POENn=0，EPWMn的输出为高阻态） 0: 所有通道正常输出（POENn=1） 1: 所有通道输出刹车数据（POENn=1） （在调试态下，运行至断点/单步后或者操作STOP按钮后暂停时，EPWMn的输出为刹车数据。）	0
25:24	MODE	EPWM工作模式选择 00: 独立模式 01: 互补模式 10: 同步模式 11: 保留	0x0
23	GROUNPEN	EPWM成组功能使能位 0: 所有的PWM通道相互独立 1: EPWM0控制 EPWM2,EPWM4,EPWM1控制 EPWM3,EPWM5	0
22	ASYMEN	EPWM中心对齐方式下非对称计数使能 0: 对称计数使能 1: 非对称计数使能	0
21	CNTTYPE	EPWM计数对齐方式选择 0: 边沿对齐 1: 中心对齐	0
20:19	-	保留	-
18	EN_DT45	EPWM计数器4和5死区使能位 0: 禁止计数器4和5死区 1: 使能计数器4和5死区	0
17	EN_DT23	EPWM计数器2和3死区使能位 0: 禁止计数器2和3死区 1: 使能计数器2和3死区	0
16	EN_DT01	EPWM计数器0和1死区使能位 0: 禁止计数器0和1死区 1: 使能计数器0和1死区	0
15:14	-	保留	-
13	PINV5	EPWM5输出极性控制位 0: 正常输出 1: 反相输出	0
12	PINV4	EPWM4输出极性控制位 0: 正常输出 1: 反相输出	0
11	PINV3	EPWM3输出极性控制位 0: 正常输出 1: 反相输出	0

10	PINV2	EPWM2输出极性控制位 0: 正常输出 1: 反相输出	0
9	PINV1	EPWM1输出极性控制位 0: 正常输出 1: 反相输出	0
8	PINV0	EPWM0输出极性控制位 0: 正常输出 1: 反相输出	0
7:6	-	保留	-
5	CNTMODE5	EPWM5自动加载/单次模 0: 单次模式 1: 自动加载模式	0
4	CNTMODE4	EPWM4自动加载/单次模式 0: 单次模式 1: 自动加载模式	0
3	CNTMODE3	EPWM3自动加载/单次模式 0: 单次模式 1: 自动加载模式	0
2	CNTMODE2	EPWM2自动加载/单次模式 0: 单次模式 1: 自动加载模式	0
1	CNTMODE1	EPWM1自动加载/单次模式 0: 单次模式 1: 自动加载模式	0
0	CNTMODE0	EPWM0自动加载/单次模式 0: 单次模式 1: 自动加载模式	0

13.5.4 EPWM控制寄存器(CON2)

位	符号	描述	复位值
31:6	-	保留	-
5	CNTEN5	EPWM5计数器使能位 0: 禁止 1: 使能 (单次模式完成后自动清除该位)	0
4	CNTEN4	EPWM4计数器使能位 0: 禁止 1: 使能 (单次模式完成后自动清除该位)	0
3	CNTEN3	EPWM3计数器使能位 0: 禁止 1: 使能 (单次模式完成后自动清除该位)	0
2	CNTEN2	EPWM2计数器使能位 0: 禁止 1: 使能 (单次模式完成后自动清除该位)	0
1	CNTEN1	EPWM1计数器使能位 0: 禁止 1: 使能 (单次模式完成后自动清除该位)	0
0	CNTEN0	EPWM0 计数器使能位 0: 禁止 1: 使能 (单次模式完成后自动清除该位)	0

13.5.5 EPWM控制寄存器(CON3)

位	符号	描述	复位值
31	LOADNWINT	EPWM加载与中断标志相关控制位 1: 加载控制与中断标志不相关 0: 加载控制与中断标志相关 加载动作产生后，中断标志是否随加载产生，不相关则每个周期和零点都会产生中断标志。	0
30	LETGHALL	HALL状态触发LOADENn使能位 0: 禁止 1: 使能HALL状态变化触发LOADENn=1 注：如果HALL检测状态改变，则将EPWM0-EPWM5的加载使能位置1。	0
29	LETGACMP1	ACMP1触发LOADENn使能位 0: 禁止 1: 使能ACMP1时触发LOADENn=1 注：如果ACMP1事件产生，则将EPWM0-EPWM5的加载使能位置1。	0
28	LETGACMP0	ACMP0触发LOADENn使能位 0: 禁止 1: 使能ACMP0时触发LOADENn=1 注：如果ACMP0事件产生，则将EPWM0-EPWM5的加载使能位置1。	0
27:26	LOADTYP5	EPWM5加载/中断方式选择位 00: 每个零点与周期点均加载与产生中断标志 01: 每个零点加载与产生中断标志 10: 第一个零点与下一个周期点交替加载与产生中断标志 11: 每两个零点加载与产生中断标志	0x0
25:24	LOADTYP4	EPWM4加载/中断方式选择位 00: 每个零点与周期点均加载与产生中断标志 01: 每个零点加载与产生中断标志 10: 第一个零点与下一个周期点交替加载与产生中断标志 11: 每两个零点加载与产生中断标志	0x0
23:22	LOADTYP3	EPWM3加载/中断方式选择 00: 每个零点与周期点均加载与产生中断标志 01: 每个零点加载与产生中断标志 10: 第一个零点与下一个周期点交替加载与产生中断标志 11: 每两个零点加载与产生中断标志	0x0
21:20	LOADTYP2	EPWM2加载/中断方式选择 00: 每个零点与周期点均加载与产生中断标志 01: 每个零点加载与产生中断标志	0x0

		10: 第一个零点与下一个周期点交替加载与产生中断标志 11: 每两个零点加载与产生中断标志	
19:18	LOADTYP1	EPWM1加载/中断方式选择位 00: 每个零点与周期点均加载与产生中断标志 01: 每个零点加载与产生中断标志 10: 第一个零点与下一个周期点交替加载与产生中断标志 11: 每两个零点加载与产生中断标志	0x0
17:16	LOADTYP0	EPWM0加载/中断方式选择 00: 每个零点与周期点均加载与产生中断标志 01: 每个零点加载与产生中断标志 10: 第一个零点与下一个周期点交替加载与产生中断标志 11: 每两个零点加载与产生中断标志	0x0
15:14	-	保留	-
13	LOADEN5	EPWM5周期/比较器加载使能 0: 禁止 1: 使能（加载完毕硬件自动清零）	0
12	LOADEN4	EPWM4周期/比较器加载使能位 0: 禁止 1: 使能（加载完毕硬件自动清零）	0
11	LOADEN3	EPWM3周期/比较器加载使能位 0: 禁止 1: 使能（加载完毕硬件自动清零）	0
10	LOADEN2	EPWM2周期/比较器加载使能位 0: 禁止 1: 使能（加载完毕硬件自动清零）	0
9	LOADEN1	EPWM1周期/比较器加载使能位 0: 禁止 1: 使能（加载完毕硬件自动清零）	0
8	LOADEN0	EPWM0周期/比较器加载使能位 0: 禁止 1: 使能（加载完毕硬件自动清零）	0
7:6	-	保留	-
5	CNTCLR5	EPWM5 计数器清零位 0: 禁止 1: 使能（硬件自动清零）	0
4	CNTCLR4	EPWM4 计数器清零位 0: 禁止 1: 使能（硬件自动清零）	0
3	CNTCLR3	EPWM3 计数器清零位 0: 禁止 1: 使能（硬件自动清零）	0
2	CNTCLR2	EPWM2 计数器清零位 0: 禁止 1: 使能（硬件自动清零）	0

1	CNTCLR1	EPWM1 计数器清零位 0: 禁止 1: 使能（硬件自动清零）	0
0	CNTCLR0	EPWM0 计数器清零位 0: 禁止 1: 使能（硬件自动清零）	0

13.5.6 EPWM周期寄存器0-5(PERIOD0-5)

位	符号	描述	复位值
31:16	-	保留	-
15:0	PERIODn	EPWMn计数器周期值	0x0

13.5.7 EPWM比较寄存器0-5(CMPDAT0-5)

位	符号	描述	复位值
31:16	CMPDDATn	EPWMn计数器向下比较值	0x0
15:0	CMPDATn	EPWMn计数器比较值	0x0

13.5.8 EPWM输出控制寄存器(POEN)

位	符号	描述	复位值
31:12	-	保留	-
11	MASKLE	EPWM掩码控制预设数据加载使能位 0: 禁止 1: 使能 （允许MASKNXT寄存器的数据加载到MASK寄存器中，同时禁止写MASK寄存器。另外该位为1时并不立即加载掩码数据，需要到相应加载点时才加载）	0
10:8	MASKLS	EPWM掩码控制数据加载时刻选择位 000: 在EPWM0的加载点加载 001: 在EPWM1的加载点加载 010: 在EPWM2的加载点加载 011: 在EPWM3的加载点加载 100: 在EPWM4的加载点加载 101: 在EPWM5的加载点加载 11x: 保留	0x0
7:6	-	保留	-
5:0	POENn	EPWMn输出使能位 0: EPWM通道n输出禁止 1: EPWM通道n输出使能	0x0

13.5.9 EPWM输出通道重映射寄存器(POREMAP)

位	符号	描述	复位值
31:24	PWMRMEN	EPWM通道重映射功能使能控制 AAH: 重映射功能使能 EPWMn由PWMnRM选择何种通道输出 其他: 重映射功能禁止 EPWMn固定通道输出如下: EPWM0<- IPG0 EPWM1<- IPG1 EPWM2<- IPG2 EPWM3<- IPG3 EPWM4<- IPG4 EPWM5<- IPG5	0x0
23	-	保留	-
22:20	PWM5RM	EPWM通道5重映射选择位 000: 映射IPG0的输出 001: 映射IPG1的输出 010: 映射IPG2的输出 011: 映射IPG3的输出 100: 映射IPG4的输出 101: 映射IPG5的输出 11x: 保留	0x5
19	-	保留	-
18:16	PWM4RM	EPWM通道4重映射选择位 000: 映射IPG0的输出 001: 映射IPG1的输出 010: 映射IPG2的输出 011: 映射IPG3的输出 100: 映射IPG4的输出 101: 映射IPG5的输出 11x: 保留	0x4
15	-	保留	-
14:12	PWM3RM	EPWM通道3重映射选择位 000: 映射IPG0的输出 001: 映射IPG1的输出 010: 映射IPG2的输出 011: 映射IPG3的输出 100: 映射IPG4的输出 101: 映射IPG5的输出 11x: 保留	0x3
11	-	保留	-
10:8	PWM2RM	EPWM通道2重映射选择位 000: 映射IPG0的输出 001: 映射IPG1的输出 010: 映射IPG2的输出 011: 映射IPG3的输出 100: 映射IPG4的输出 101: 映射IPG5的输出 11x:	0x2

		保留	
7	-	保留	-
6:4	PWM1RM	EPWM通道1重映射选择位 000: 映射IPG0的输出 001: 映射IPG1的输出 010: 映射IPG2的输出 011: 映射IPG3的输出 100: 映射IPG4的输出 101: 映射IPG5的输出 11x: 保留	0x1
3	-	保留	-
2:0	PWM0RM	EPWM通道0重映射选择位 000: 映射IPG0的输出 001: 映射IPG1的输出 010: 映射IPG2的输出 011: 映射IPG3的输出 100: 映射IPG4的输出 101: 映射IPG5的输出 11x: 保留	0x0

13.5.10 EPWM故障保护控制寄存器(BRKCTL)

位	符号	描述	复位值
31	BRKEN	EPWM故障保护功能总使能位 0: 禁止（复位故障保护电路） 1: 使能	0
30	BRKAF	EPWM故障信号标志位（只读） 0: 没有产生故障 1: 产生了故障信号或刹车信号保持有效	0
29:28	BRKMS	EPWM故障保护模式选择位 00: 停止模式 01: 暂停模式 10: 恢复模式 11: 延时恢复模式 注：切换故障保护模式时，必须禁止故障保护总使能后，然后切换故障保护模式，最后打开故障保护总使能位。	0x0
27	BRKCLR	EPWM故障保护清除位（只写） 0: -- 1: 清除故障保护状态 注：仅当BRKAF=0时才能写1，进行故障清除操作，否则操作无效。	0
26:24	BRKRCS	EPWM故障恢复加载点选择位 000: EPWM0的加载点恢复 001: EPWM1的加载点恢复 010: EPWM2的加载点恢复 011: EPWM3的加载点恢复 100: EPWM4的加载点恢复 101: EPWM5的加载点恢复 其他: 禁止选择	0x0
23	ACMP1BKLE	模拟比较器1输出电平控制刹车使能位 0: 禁止 1: 使能	0
22	ACMP1BKLS	模拟比较器1输出电平控制刹车选择位 0: 低电平产生刹车 1: 高电平产生刹车	0
21	ACMP0BKLE	模拟比较器0输出电平控制刹车使能位 0: 禁止 1: 使能	0
20	ACMP0BKLS	模拟比较器0输出电平控制刹车选择位 0: 低电平产生刹车 1: 高电平产生刹车	0
19	ACMP1BKEN	模拟比较器1输出事件控制刹车使能位 0: 禁止 1: 使能 （比较器输出事件指产生上升	0

		沿/下降沿/双沿，可在ACMP->CEVCON选择何种边沿）	
18	ACMP0BKEN	模拟比较器0输出事件控制刹车使能位 0: 禁止 1: 使能 （比较器输出事件指产生上升沿/下降沿/双沿，可在ACMP->CEVCON选择何种边沿）	0
17	ADCMP1BKEN	ADC比较器1输出刹车使能位 0: 禁止 1: 使能	0
16	ADCMP0BKEN	ADC比较器0输出刹车使能位 0: 禁止 1: 使能	0
15	--	必须为0	0
14	--	保留	0
13	BRKOSF	EPWM故障保护输出状态标志位（只读） 0: EPWMn通道为正常输出状态 1: EPWMn通道为输出BRKODn的数据状态	0
12	SWBRK	软件刹车使能位 0: 禁止软件刹车 1: 立即产生软件刹车	0
11	EXTBRKEE	外部硬件刹车边沿检测使能位 0: 禁止 1: 使能	0
10	EXTBRKES	外部硬件刹车边沿检测选择位 0: 下降沿触发刹车 1: 上升沿触发刹车	0
9	EXTBRKLE	外部硬件刹车电平检测使能位 0: 禁止 1: 使能	0
8	EXTBRKLS	外部硬件刹车电平检测选择位 0: 低电平产生刹车 1: 高电平产生刹车	0
7:6	-	必须为0	0x0
5:0	BRKODn	EPWMn刹车输出电平选择位 0: 当故障刹车后，通道n输出低电平 1: 当故障刹车后，通道n输出高电平	0x0

13.5.11 EPWM死区长度寄存器(DTCTL)

位	符号	描述	复位值
31:30	-	保留	-
29:20	DTI45	通道4和通道5死区长度寄存器 死区时间 = PWM_CLK45 × DTI45	0x0
19:10	DTI23	通道2和通道3死区长度寄存器 死区时间 = PWM_CLK23 × DTI23	0x0
9:0	DTI01	通道0和通道1死区长度寄存器 死区时间 = PWM_CLK01 × DTI01	0x0

13.5.12 EPWM掩码输出控制寄存器(MASK)

位	符号	描述	复位值
31:14	-	保留	-
13	MASKEN5	EPWM5掩码输出使能位 0: 禁止 1: 使能	0
12	MASKEN4	EPWM4掩码输出使能位 0: 禁止 1: 使能	0
11	MASKEN3	EPWM3掩码输出使能位 0: 禁止 1: 使能	0
10	MASKEN2	EPWM2掩码输出使能位 0: 禁止 1: 使能	0
9	MASKEN1	EPWM1掩码输出使能位 0: 禁止 1: 使能	0
8	MASKEN0	EPWM0掩码输出使能位 0: 禁止 1: 使能	0
7:6	-	保留	-
5	MASKD5	EPWM5 掩码数据 0: 输出0 1: 输出1	0
4	MASKD4	EPWM4 掩码数据 0: 输出0 1: 输出1	0
3	MASKD3	EPWM3 掩码数据 0: 输出0 1: 输出1	0
2	MASKD2	EPWM2 掩码数据 0: 输出0 1: 输出1	0
1	MASKD1	EPWM1 掩码数据 0: 输出0 1: 输出1	0
0	MASKD0	EPWM0 掩码数据 0: 输出0 1: 输出1	0

13.5.13 EPWM掩码输出控制预设寄存器(MASKNXT)

位	符号	描述	复位值
31:25	-	保留	-
24	HALLEN	HALL检测模式使能位 0: 禁止 1: 使能	0
23	HALLCLR	HALL错误状态清除位 0: 写0无效 1: 写1清除HALLST的错误状态，让其回到初始状态000。 读为0。 注1：如果出现的错误的状态或者序列时，HALLST=111，HALL检测功能停止。再次检测开启HALL状态时，需要写1清除111的状态。	0
22:20	HALLST	HALL接口的状态位（只读） 检测对应{CAP2,CAP1,CAP0}的状态 000: 状态为0（初始状态） 001: 状态为1 010: 状态为2 011: 状态为3 100: 状态为4 101: 状态为5 110: 状态为6 111: 错误状态 注1：该状态为芯片内部检测到HALL接口的状态，可由此判断是否进入了有效状态，如果3个HALL传感器的状态出现错误或者状态的顺序出现错误，则该状态位为111。 有效序列1: 6-2-3-1-5-4-6- 有效序列2: 6-4-5-1-3-2-6- 注2：在有效的状态位下，如果使能掩码预设数据加载功能，则相应的掩码预设缓存数据在加载点装入到MASK寄存器中。如HALL检测改变为状态3后，在进入状态3后第一个加载点将掩码预设缓存3的数据装入MASK寄存器中。 注3：在初始状态000或错误状态111下输出掩码预设缓存7的数据。	0x0
19	-	保留	-
18:16	PMASKSEL	掩码预设缓存选择位； 000: 选择掩码预设缓存0 001: 选择掩码预设缓存1 010: 选择掩码预设缓存2 011: 选择掩码预设缓存3 100: 选择掩码预设缓存4 101: 选择掩码预设缓存5	0x0

		110: 选择掩码预设缓存6 111: 选择掩码预设缓存7 注1：该选择位影响读写低16位的数据，EPWM内部存在6个掩码预设缓存 如为000: 则该寄存器读写低16位为掩码缓存0中的数据， 如为001: 则该寄存器读写低16位为掩码缓存1中的数据 如为110: 则该寄存器读写低16位为掩码缓存6中的数据。 注2：HALLEN=0时，默认加载掩码预设缓存0中的数据。	
15:14	-	保留	-
13	PMASKEN5	EPWM5掩码输出使能预设位 （该位可设置在EPWMn的加载点加载到MASK寄存器中）	0
12	PMASKEN4	EPWM4掩码输出使能预设位 （该位可设置在EPWMn的加载点加载到MASK寄存器中）	0
11	PMASKEN3	EPWM3掩码输出使能预设位 （该位可设置在EPWMn的加载点加载到MASK寄存器中）	0
10	PMASKEN2	EPWM2掩码输出使能预设位 （该位可设置在EPWMn的加载点加载到MASK寄存器中）	0
9	PMASKEN1	EPWM1掩码输出使能预设位 （该位可设置在EPWMn的加载点加载到MASK寄存器中）	0
8	PMASKEN0	EPWM0掩码输出使能预设位 （该位可设置在EPWMn的加载点加载到MASK寄存器中）	0
7:6	-	保留	-
5	PMASKD5	EPWM5 掩码数据预设位 （该位可设置在EPWMn的加载点加载到MASK寄存器中）	0
4	PMASKD4	EPWM4 掩码数据预设位 （该位可设置在EPWMn的加载点加载到MASK寄存器中）	0
3	PMASKD3	EPWM3 掩码数据预设位 （该位可设置在EPWMn的加载点加载到MASK寄存器中）	0
2	PMASKD2	EPWM2 掩码数据预设位 （该位可设置在EPWMn的加载点加载到MASK寄存器中）	0
1	PMASKD1	EPWM1 掩码数据预设位 （该位可设置在EPWMn的加载点加载到MASK寄存器中）	0
0	PMASKD0	EPWM0 掩码数据预设位 （该位可设置在EPWMn的加载点加载到MASK寄存器中）	0

13.5.14 EPWM触发比较寄存器(CMPTGD0-1)

位	符号	描述	复位值
31:20	-	保留	-
19	CMPTGDSn	EPWM计数比较器n触发模式 (在中心对齐计数方式下生效) 0: 向下计数时触发 1: 向上计数时触发	0
18:16	CMPPCHSn	EPWM 数字比较器n比较通道选择 000: PWM0的计数器 001: PWM1的计数器 010: PWM2的计数器 011: PWM3的计数器 100: PWM4的计数器 101: PWM5的计数器 其他值: PWM0的计数器	0x0
15:0	CMPTGDn	EPWM计数比较器n触发比较值	0x0

13.5.15 EPWM中断使能寄存器(IMSC)

位	符号	描述	复位值
31	EN_BRKIF	EPWM故障中断使能位 0: 禁止 1: 使能	0
30	EN_HALLIF	HALL状态错误中断使能位 0: 禁止 1: 使能	0
29:24	EN_DIFn （n=5-0）	EPWMn向下比较中断使能位 0: 禁止 1: 使能	0x0
23:22	-	保留	-
21:16	EN_UIFn （n=5-0）	EPWMn向上比较中断使能位 0: 禁止 1: 使能	0x0
15	EN_DC1IF	计数比较器1中断使能位 0: 禁止 1: 使能	0
14	EN_DC0IF	计数比较器0中断使能位 0: 禁止 1: 使能	0
13:8	EN_PIFn （n=5-0）	EPWMn周期中断使能位 0: 禁止 1: 使能	0x0
7:6	-	保留	-
5:0	EN_ZIFn （n=5-0）	EPWMn零点中断使能位 0: 禁止 1: 使能	0x0

13.5.16 EPWM中断源状态寄存器(RIS)

位	符号	描述	复位值
31	RIS_BRKIF	EPWM故障中断源状态位 0: 未产生中断 1: 已产生中断	0
30	RIS_HALLIF	HALL状态错误中断源状态位 0: 未产生中断 1: 已产生中断	0
29:24	RIS_DIFn （n=5-0）	EPWMn向下比较中断源状态位 0: 未产生中断 1: 已产生中断	0x0
23:22	-	保留	-
21:16	RIS_UIFn （n=5-0）	EPWMn向上比较中断源状态位 0: 未产生中断 1: 已产生中断	0x0
15	RIS_DC1IF	计数比较器1中断状态位 0: 禁止 1: 使能	0
14	RIS_DC0IF	计数比较器0中断状态位 0: 禁止 1: 使能	0
13:8 n=5-0	RIS_PIFn （n=5-0）	EPWMn周期中断源状态位 0: 未产生中断 1: 已产生中断	0x0
7:6	-	保留	-
5:0 n=5-0	RIS_ZIFn （n=5-0）	EPWMn零点中断源状态位 0: 未产生中断 1: 已产生中断	0x0

13.5.17 EPWM已使能中断状态寄存器(MIS)

位	符号	描述	复位值
31	MIS_BRKIF	EPWM故障已使能中断状态位 0: 未产生中断 1: 已使能并产生中断	0
30	MIS_HALLIF	HALL状态错误已使能中断状态位 0: 未产生中断 1: 已使能并产生中断	0
29:24	MIS_DIFn （n=5-0）	EPWMn向下比较已使能中断状态位 0: 未产生中断 1: 已使能并产生中断	0x0
23:22	-	-	-
21:16	MIS_UIFn （n=5-0）	EPWMn向上比较已使能中断状态位 0: 未产生中断 1: 已使能并产生中断	0x0
15	RIS_DC1IF	计数比较器1已使能中断状态位 0: 禁止 1: 使能	0
14	RIS_DC0IF	计数比较器0已使能中断状态位 0: 禁止 1: 使能	0
13:8	MIS_PIFn （n=5-0）	EPWMn周期已使能中断状态位 0: 未产生中断 1: 已使能并产生中断	0x0
7:6	-	-	-
5:0	MIS_ZIFn （n=5-0）	EPWMn零点已使能中断状态位 0: 未产生中断 1: 已使能并产生中断	0x0

13.5.18 EPWM中断清零控制寄存器(ICLR)

位	符号	描述	复位值
31	ICLR_BRKIF	EPWM故障中断清零控制位 0: 不影响 1: 清零RIS_BRKIF标志位	0
30	ICLR_HALLIF	HALL状态错误中断清零控制位 0: 不影响 1: 清零RIS_HALLIF标志位 注：如果HALLST=111，则无法清除RIS_HALLIF标志位	0
29:24	ICLR_DIFn （n=5-0）	EPWMn向下比较中断清零控制位 0: 不影响 1: 清零RIS_DIFn标志位	0x0
23:22	-	-	-
21:16	ICLR_UIFn （n=5-0）	EPWMn向上比较中断清零控制位 0: 不影响 1: 清零RIS_UIFn标志位	0x0
15	ICLR_DC1IF	计数比较器1中断清零控制位 0: 不影响 1: 清零RIS_DC1IF标志位	0
14	ICLR_DC0IF	计数比较器0中断清零控制位 0: 不影响 1: 清零RIS_DC0IF标志位	0
13:8	ICLR_PIFn （n=5-0）	EPWMn周期中断清零控制位 0: 不影响 1: 清零RIS_PIFn标志位	0x0
7:6	-	-	-
5:0	ICLR_ZIFn （n=5-0）	EPWMn零点中断清零控制位 0: 不影响 1: 清零RIS_ZIFn标志位	0x0

13.5.19 EPWM中断累加控制寄存器(IFA)

位	符号	描述	复位值
31:16	-	保留	-
15:12	BRKIFCMP	故障保护中断累加比较值 当故障中断累加到(BRKIFCMP+1)时，BRKIF中断标志位置1	0x0
11:9	-	保留	0x0
8	BRKIFAEN	故障保护中断累加使能位 0: 禁止 1: 使能	0
7:4	ZIFCMP	零点中断累加比较值 当相应通道的零点中断累加到(ZIFCMP+1)时，ZIFn中断标志位置1（所有的通道为同一个比较值）	0x0
3:1	-	保留	-
0	ZIFAEN	零点中断累加使能位 0: 禁止 1: 使能	0

13.5.20 EPWM写使能控制寄存器(LOCK)

位	符号	描述	复位值
31:8	-	保留	-
7:0	LOCK	当LOCK=0x55时，使能操作EPWM其他寄存器； 当LOCK=0xaa时，只使能操作EPWM周期寄存器和比较寄存器； 当LOCK=其他值时，禁止操作EPWM相关寄存器。	0x0

13.5.21 EPWM故障保护恢复延时寄存器(BRKRDT)

位	符号	描述		复位值
31:20	-	必须为0		0x0
19:16	FILS	故障保护（刹车）信号滤波时间选择位		0x0
		0000:	(0~1)* TPCLK	
		0001:	(1~2)* TPCLK	
		0010:	(2~3)* TPCLK	
		0011:	(4~5)* TPCLK	
		0100:	(8~9)* TPCLK	
		0101:	(16~17)* TPCLK	
		0110:	(24~25)* TPCLK	
		0111:	(32~33)* TPCLK	
		1000:	(48~49)* TPCLK	
		1001:	(64~65)* TPCLK	
		1010:	(80~81)* TPCLK	
		1011:	(96~97)* TPCLK	
		1100:	(112~113)* TPCLK	
		其他:	(0~1)* TPCLK	
15:0	RDT	故障保护恢复延时（仅延时恢复模式有效） 延时时间 = RDT ×TAPBCLK		0x0

第14章 通用异步收发器（UART）

14.1 概述

包含1路通用异步串行接口。

14.2 特性

- 全双工，异步通信。
- 可编程串行接口特性。
  - 数据位长度可设为 5~8 位。
  - 校验位可设为奇偶校验、无校验或固定校验位的产生和检测。
  - 停止位长度可设置为为 1 位、1.5 位或 2 位。

14.3 功能描述

14.3.1 UART功能模式

UART为全双工异步通讯接口。UART收发器各包含一个缓冲区，可灵活设置发送字节长度和停止位长度。全双工串行接口通讯参数可设。

14.3.2 UART中断和状态

UART支持3种类型的中断，具体中断类型包括如下：

- Line 状态中断（奇偶校验错误、帧错误、打断中断）。
- 接收数据有效中断。
- 发送保持寄存器空中断。

14.4 寄存器映射

（UART0基地址 = 0x4006_4000）

RO：只读；WO：只写，R/W：读写；

寄存器	偏移量	读/写	描述	复位值
RBR	0x000	RO	接收缓存寄存器	-
THR	0x004	WO	发送缓存寄存器	-
DLR	0x008	R/W	波特率分频寄存器	0x1
IER	0x00c	R/W	中断使能寄存器	0x0
IIR	0x010	RO	中断状态寄存器	0x1
LCR	0x018	R/W	线控制寄存器	0x0
MCR	0x01C	R/W	Modem控制寄存器	0x0
LSR	0x020	RO	线状态寄存器	0x60
END	0x030	W	UART访问结束寄存器	-

14.5 寄存器说明

14.5.1 接收缓存寄存器（RBR）

位	符号	描述	复位值
31:8	-	保留	-
7:0	RBR	读操作，返回接收到的来自接收Buffer的数据	-

14.5.2 发送缓存寄存器（THR）

位	符号	描述	复位值
31:8	-	保留	-
7:0	THR	写数据到发送Buffer，UART模块随后会将Buffer中数据发送出去	-

14.5.3 波特率分频寄存器（DLR）

位	符号	描述	复位值
31:16	-	保留	-
15:0	DLR	波特率= PCLK/16×DLR	0x1

14.5.4 中断使能寄存器（IER）

位	符号	描述	复位值
31:3	-	保留	-
2	RLSIE	接收线状态中断使能位 0: 禁止 1: 使能	0
1	THREIE	发送保持寄存器空中断使能位 0: 禁止 1: 使能	0
0	RBRIE	接收数据有效中断/接收定时器溢出中断使能位 0: 禁止 1: 使能	0

14.5.5 中断状态寄存器（IIR）

位	符号	描述	复位值
31:4	-	保留	-
3:1	INTID	中断状态指示 0x0: Modem状态发生改变 0x1: 发送保持寄存器为空 0x2: 接收数据有效 0x3: 接收到线状态	0x0
0	INT STATUS	中断状态 0: 至少一个中断在队列中 1: 没有中断在队列中	1

14.5.6 线控制寄存器（LCR）

位	符号	描述	复位值
31:7	-	保留	-
6	BCON	Break控制位 当该位写1，使能Break传输，TXD口将会强制输出逻辑0	0
5:4	PSEL	奇偶校验位选择 0x0: 奇校验，逻辑1的奇数数目在每个字节中被发送和检测 0x1: 偶校验，逻辑1的偶数数目在每个字节中被发送和检测 0x2: 校验位强制为1 0x3: 校验位强制为0	0x0
3	PEN	奇偶校验位使能 0: 禁止校验位的产生的检测 1: 使能校验位的产生和检测	0
2	SBS	停止位选择 0: 1位停止位 1: 当发送字长为5位时，停止位为1.5位；当发送字长为其他时，停止位为2位	0
1:0	WLS	字长度选择位 0x0: 5位字长 0x1: 6位字长 0x2: 7位字长 0x3: 8位字长	0x0

14.5.7 Modem控制寄存器（MCR）

位	符号	描述	复位值
31:5	-	保留	-
4	MLBM	Modem回环模式 0: 禁止Modem回环模式 1: 使能Modem回环模式	0
3:0	-	保留	-

14.5.8 线状态寄存器（LSR）

位	符号	描述	复位值
31:7	-	保留	-
6	TEMT	发送缓存空标志位（只读） 0: 发送缓存有未发送完毕的数据 1: 发送缓存为空	1
5	THRE	表示为发送寄存器空标志位（只读） 0: 发送寄存器有未发送的数据 1: 发送寄存器为空	1
4	BI	打断中断标志位（只读） 0: 未检测到打断中断 1: 检测到打断中断 当UART数据输入口在一个传输过程中（起始位，数据，校验位，停止位）保持为低电平，则触发打断中断。UART保持空闲状态直到数据输入口为高电平。 可通过读LSR寄存器清零该位	0
3	FE	帧错误标志位（只读） 0: 未检测到帧错误 1: 检测到帧错误 可通过读LSR寄存器清零该位	0
2	PE	校验位错误标志位（只读） 0: 未检测到校验位错误 1: 检测到校验位错误 可通过读LSR寄存器清零该位	0
1	-	保留	0
0	RDR	接收数据有效标志位（只读） 0: 接收区没有未读数据 1: 接收区有未读数据	0

14.5.9 UART访问结束寄存器（END）

位	符号	描述	复位值
31：0	END	所有UART寄存器访问结束寄存器 0x0: 可以访问UART以外的寄存器 其他: 禁止写入	-

注：

1. UART相关寄存器后，若要操作UART以外的寄存器，必须先对END寄存器进行写0操作。
2. 操作END寄存器时，如果存在被中断打断的情况，可能会对总线有影响，所以未避免程序出错，有两种如下方法解决：

   （1）在操作END寄存器前先关掉中断使能，操作完成后，再恢复中断使能。

   （2）在中断服务程序里执行一次写END寄存器。

第15章 I²C串行接口控制器（I²C）

15.1 概述

I²C是一种两线双向串行总线，为设备之间的数据交换提供了一种简单有效的连接方式。I²C是一个真正的多主机总线，包含了冲突检测和仲裁机制。冲突检测和仲裁机制用来在两个或多个主机同时尝试控制总线的情况下，防止数据损坏。

15.2 特性

- 支持主机/从机模式。
- 主从机之间双向数据传送。
- 多主机总线。
- 多主机间同时传输数据仲裁，避免总线上串行数据损坏。
- 总线采用串行同步时钟，可实现设备之间以不同的速率传输。
- 串行同步时钟可以被用来作为握手机制实现挂起和恢复串行传输。
- 可编程的时钟可以用于多种速率控制。
- 支持 7 位/10 位从地址模式。
- 支持多地址识别（4 组从机地址和 1 组从机扩展地址带 mask 选项）。
- 支持唤醒模式。

15.3 寄存器映射

（I2C0 基地址 = 0x4006_4300）RO：只读；WO：只写；R/W：读写。

寄存器	偏移量	读/写	描述	复位值
CONSET	0x000	R/W	I²C控制置位寄存器	0x0
CONCLR	0x004	WO	I²C控制清零寄存器	0x0
STAT	0x008	RO	I²C状态寄存器	0xF8
DAT	0x00C	R/W	I²C数据寄存器	0x0
CLK	0x010	R/W	I²C时钟控制寄存器	0x0
ADR0	0x014	R/W	I²C从机地址寄存器0	0x0
ADM0	0x018	R/W	I²C从机地址掩码寄存器0	0xFE
XADR0	0x01C	R/W	I²C扩展从机地址寄存器0	0x0
XADM0	0x020	R/W	I²C扩展从机地址掩码寄存器0	0x1FE
RST	0x024	WO	I²C软件复位寄存器	0x0
ADR1	0x028	R/W	I²C从机地址寄存器1	0x0
ADM1	0x02C	R/W	I²C从机地址掩码寄存器1	0xFE
ADR2	0x030	R/W	I²C从机地址寄存器2	0x0
ADM2	0x034	R/W	I²C从机地址掩码寄存器2	0xFE
ADR3	0x038	R/W	I²C从机地址寄存器3	0x0
ADM3	0x03C	R/W	I²C从机地址掩码寄存器3	0xFE

15.4 寄存器说明

15.4.1 I²C控制置位寄存器（CONSET）

位	符号	描述	复位值
31:9	-	保留	-
8	GCF	I²C广播呼叫标志位 只读 0: 未接收到广播呼叫 1: 广播呼叫地址匹配 当收到停止位/重启动位/复位信号时该位清零	0
7	I2CIE	中断使能位 0: 禁止 1: 使能	0
6	I2CEN	I²C接口使能位 0: 禁止I²C接口 1: 使能I²C接口 注：通过在I2CEN位写1使能I2C接口，通过在I2CENC位（I2CxCONCLR）写1禁止I2C接口，只在从机模式下有效(地址匹配不成功)。	0
5	STA	启动标志位 1：I²C进入主机模式并发送启动信号； - 当I²C已经处于主机模式，则发送重启动信号。 - 当I²C处于从机模式时，写1会结束当前传输并等待总线空闲时进入主机模式。 0：不影响。 - 当启动位或重启动位发送完成时，该位自动清零。	0
4	STO	停止标志位 在主机模式下写1时，会发送一个停止位。 在从机模式下写1时，I²C模块会当作接收到一个停止位 - 当同时将STA和STO置位时，I²C模块会先发送一个停止位，接着发送一个启动位。 - 当停止位发送完成时，该位自动清零。	0
3	SI	I²C中断标志位 只读 当I²C发生总线状态改变时该位置位，可通过在SIC位写1清零。	0
2	AA	应答标志位 0: 没有接收到ACK信号 1: 在以下情况回复ACK信号 ● 从机地址匹配时 ● 使能广播呼叫且接收到广播地址时 ● 在主机或从机模式下接收到数据时可通过在AAC位写1清零该位	0
1	XADRF	I²C 从机10位地址标志位 只读 0: I²C地址不匹配	0

		1: 1²C 10位地址匹配 当收到停止位/复位信号时该位清零	
0	ADRF	I²C从机7位地址标志位，只读 0: I²C地址不匹配 1: 1²C 7位地址匹配 当收到停止位/复位信号时该位清零	0

15.4.2 I²C控制清零寄存器（CONCLR）

位	符号	描述	复位值
31:8	-	保留	-
7	I2CIEC	I²C中断禁止位 0: 清零I2CIE位 1: 不影响	0
6	I2CENC	I²C接口禁止位 1: 清零I2CEN位 0: 不影响	0
5	STAC	启动标志清零位 0: 不影响 1: 清零STA位	-
4	-	保留	
3	SIC	I²C中断标志清零位 0: 不影响 1: 清零SI位	0
2	AAC	I²C应答标志清零位 0: 不影响 1: 清零AA位	0
1:0	-	保留	-

注：I²C的操作需要清除相应的标志位才能进入下一状态。

15.4.3 I²C状态寄存器（STAT）

位	符号	描述	复位值
31:8	-	保留	-
7:0	Status	I²C状态代码 00H: 总线错误（只在主机模式有效） 08H: 启动位发送完成 10H: 重启动位发送完成 18H: 地址+写位发送完成，接收到ACK 20H: 地址+写位发送完成，未接收到ACK 28H: 主机模式下数据发送完成，接收到ACK 30H: 主机模式下数据发送完成，未接收到ACK 38H: 在地址或数据传输过程中仲裁失败 40H: 地址+读位发送完成，接收到ACK 48H: 地址+读位发送完成，未接收到ACK 50H: 主机模式下接收到数据，回复ACK 58H: 主机模式下接收到数据，不回复ACK 60H: 从机模式下接收到地址+写位，回复ACK 68H: 主机仲裁失败，接收到从机地址+写位，回复ACK 70H: 接收到广播呼叫地址，回复ACK 78H: 主机仲裁失败，接收到广播呼叫地址，回复ACK 80H: 从机地址匹配后接收到数据，回复ACK 88H: 从机地址匹配后接收到数据，不回复ACK 90H: 从机接收广播呼叫地址后接收到数据，回复ACK 98H: 从机接收广播呼叫地址后接收到数据，不回复ACK A0H: 从机模式下接收到停止信号或重启动信号 A8H: 从机模式下接收到地址+读位，回复ACK B0H: 主机仲裁失败，接收到从机地址+读位，回复ACK B8H: 从机模式下发送数据后，接收到ACK C0H: 从机模式下发送数据后，未接收到ACK C8H: 从机模式下发送完最后一个数据，接收到ACK D0H: 从机模式下发送完最后一个数据，未接收到ACK D8H: 未用 E0H: 主机模式下发送完第二个地址，接收到ACK E8H: 主机模式下发送完第二个地址，未接收到ACK F0H: 未用 F8H: 不确切的状态 其他: 保留	0xF8

15.4.4 I²C数据寄存器（DAT）

位	符号	描述	复位值
31:8	-	保留	-
7:0	Data	接收到的数据或将被发送的数据(读取接收到的数据需在数据接收完成后立即读取）。	0x0

15.4.5 I²C时钟控制寄存器（CLK）

位	符号	描述	复位值
31:7	-	保留	-
6:4	M	采样时钟= PCLK/（2^M×（N+1）） SCL时钟= PCLK/（2^M×（N+1）×10）	0x0
3:0	N		0x0

15.4.6 I²C从机地址寄存器（ADR0/ADR1/ADR2/ADR3）

位	符号	描述	复位值
31:8	-	保留	-
7:1	Address	从机地址	0x0
0	GC	1: 使能广播呼叫地址识别 0: 禁止广播呼叫地址识别	0

15.4.7 I²C从机地址掩码寄存器（ADM0/ADM1/ADM2/ADM3）

位	符号	描述	复位值
31:8	-	保留	-
7:1	MASK	掩码位 0: 不比较该位地址 1: 比较该位地址	0x7F
0	-	保留	-

15.4.8 I²C扩展从机地址寄存器（XADR0）

位	符号	描述	复位值
31:11	-	保留	-
10:1	Address	10位从机地址	0x0
0	GC	1: 使能广播呼叫地址识别 0: 禁止广播呼叫地址识别	0

15.4.9 I²C扩展从机地址掩码寄存器（XADM0）

位	符号	描述	复位值
31:9	-	保留	-
8:1	MASK	掩码位 0: 不比较该位地址 1: 比较该位地址	0xFF
0	-	保留	-

15.4.10 I²C软件复位寄存器（RST）

位	符号	描述	复位值
31:8	-	保留	-
7:0	RST	写入任意值，产生软件复位	0x0

第16章 串行外围接口控制器（SSP/SPI）

16.1 概述

串行外围设备接口（SPI）是一个工作于全双工模式的同步串行数据通讯协议。设备可工作在主/从模式，利用4线双向接口相互通讯。当从一个外围设备接收数据时，SPI执行串--并的转换，而在数据向外围设备发送时执行并-串的转换。该SPI控制器可以配置为主设备或从设备。

16.2 特性

- 支持主机或从机模式。
- 全双工。
- 可配置发送的位长度（4bit-16bit）。
- MSB 优先发送/接收。
- 内置一个接收 Buffer 和一个发送 Buffer

16.3 寄存器映射

（SSP0基地址= 0x4006_3000）RO：只读；WO：只写；R/W：读写。

寄存器	偏移量	读/写	描述	复位值
CON	0x000	R/W	SSP控制寄存器	0x0
STAT	0x004	RO	SSP 状态寄存器	0x3
DAT	0x008	R/W	SSP 数据寄存器	0x0
CLK	0x00C	R/W	SSP 时钟控制寄存器	0x0
IMSC	0x010	R/W	SSP 中断使能寄存器	0x0
RIS	0x014	RO	SSP 中断源状态寄存器	0x8
MIS	0x018	RO	SSP 已使能中断状态寄存器	0x0
ICLR	0x01C	WO	SSP 中断清零寄存器	0x0
CSCR	0x028	R/W	SSP软件片选信号寄存器	0x0

16.4寄存器说明

16.4.1 SSP控制寄存器（CON）

位	符号	描述	复位值
31:12	-	保留	-
11	LBM	回环模式使能位 0: 正常工作模式 1: 回环模式，串行输入接到串行输出	0
10	SSPEN	SSP使能位 0: 禁止 1: 使能	0
9	MS	主机/从机模式选择位 0: 主机模式 1: 从机模式	0
8		保留	0
7	CPH	时钟相位控制位 0: SSP在第一个时钟边沿采样数据 1: SSP在第二个时钟边沿采样数据	0
6	CPO	时钟输出极性选择位 0: SPI_CLK在空闲时为低电平 1: SPI_CLK在空闲时为高电平	0
5:4	FRF	帧格式 0x0: SPI–兼容帧格式 0x1: TISS-兼容帧格式 0x2: Microwire –兼容帧格式 0x3: 保留	0x0
3:0	DSS	数据传输长度选择位 0x0: 保留 0x1: 保留 0x2: 保留 0x3: 4位长度 0x4: 5位长度 0x5: 6位长度 0x6: 7位长度 0x7: 8位长度 0x8: 9位长度 0x9: 10位长度 0xA: 11位长度 0xB: 12位长度 0xC: 13位长度 0xD: 14位长度 0xE: 15位长度 0xF: 16位长度	0x0

16.4.2 SSP状态寄存器（STAT）

位	符号	描述	复位值
31:5	-	保留	-
4	BSY	忙标志位，只读 0: SSP空闲 1: SSP正在发送/接收数据或发送Buffer已写入数据	0
3:0		保留	0x3

16.4.3 SSP数据寄存器（DAT）

位	符号	描述	复位值
31:16	-	保留	-
15:0	DATA	写数据到该寄存器，当总线上没有数据在发送时，该数据会写入到发送寄存器被发送出去；当总线上有数据在发送时，该数据会存入Buffer并在上一次数据传输完成后发送。发送时间的间隔最短为3个SSPCLK时钟。 当数据长度小于16位时，需右对齐。 读该寄存器，读到的是最近接收到的数据，当数据长度小于16位时，需右对齐。	0x0

16.4.4 SSP时钟控制器（CLK）

位	符号	描述	复位值
31:16	-	保留	-
15:8	M	SSPCLK = PCLK /（（M+1）×N） N为2-254的偶数	0x0
7:0	N		0x0

16.4.5 SSP中断使能寄存器（IMSC）

位	符号	描述	复位值
31:4	-	保留	-
3	TXIM	发送Buffer中断使能位 0: 禁止发送Buffer中无数据中断 1: 使能发送Buffer中无数据中断	0
2	RXIM	接收Buffer中断使能位 0: 禁止接收Buffer收到数据中断 1: 使能接收Buffer收到数据中断	0
1	RTIM	接收Buffer定时器溢出中断使能位 0: 禁止接收Buffer定时器溢出中断 1: 使能接收Buffer定时器溢出中断（溢出时间为：32×SSPCLK）	0
0	RORIM	接收Buffer溢出中断使能位 0: 禁止接收Buffer溢出中断 1: 使能接收Buffer溢出中断	0

16.4.6 SSP中断源状态寄存器（RIS）

位	符号	描述	复位值
31:4	-	保留	-
3	TXRIS	当发送 Buffer 无数据时或发送 Buffer 中数据被发送后该位置位 (当发送 Buffer 中有数据时，该位自动清零)	1
2	RXRIS	当接收 Buffer 收到数据时该位置位 （当接收 Buffer 没有数据时或接收 Buffer 中数据被读取后，该位自动清零）	0
1	RTRIS	当接收 Buffer 收到数据，且超时未被读取时该位置位 (读取数据寄存器或写 ICLR 寄存器清零)	0
0	RORRIS	当接收 Buffer 收到数据且未被读取期间，又接收到一帧数据时该位置位，新数据将会丢失 (写 ICLR 寄存器清零)	0

16.4.7 SSP已使能中断状态寄存器（MIS）

位	符号	描述	复位值
31:4	-	保留	-
3	TXMIS	= TXIM & TXRIS	0
2	RXMIS	= RXIM & RXRIS	0
1	RTMIS	= RTIM & RTRIS	0
0	RORMIS	= RORIM & RORRIS	0

16.4.8 SSP中断清零寄存器（ICLR）

位	符号	描述	复位值
31:2	-	保留	-
1	RTIC	1：清零RTRIS标志位	0
0	RORIC	1：清零RORRIS标志位	0

16.4.9 SSP软件片选信号寄存器（CSCR）

位	符号	描述	复位值
31:4	-	保留	-
3	SWCS	主机模式下软件片选信号控制位 0: 输出低电平 1: 输出高电平	0
2	SWSEL	主机模式下片选信号选择 0: 片选信号由SPI模块自动控制 1: 片选信号由SWCS位控制	0
1:0	-	保留	-

第17章 快速模数转换（ADC）

17.1 概述

芯片包含一个12位23通道快速逐次逼近型模数转换器(ADC)。

17.2 特性

- 模拟输入电压范围： VSS ~ AVDD。
- 最大采样速率：1.2Msps。
- 多达23路单端模拟输入通道。
- 支持两种功耗模式：高速模式与低电流模式。
- 高速模式下单次采样与转换时间为：52*T_{ADCK}(采样时间设置为13.5*T_{ADCK})。
- 单次模式：对指定通道执行一次A/D转换。
- 连续模式：对所有选定的通道都执行A/D转换。
- 支持外部输入信号触发ADC转换。
- 支持转换完毕产生中断。
- 内置AD转换结果比较器。
- 每个通道的转换结果都存储在对应的数据寄存器中。

17.3 功能描述

17.3.1 ADC的通道

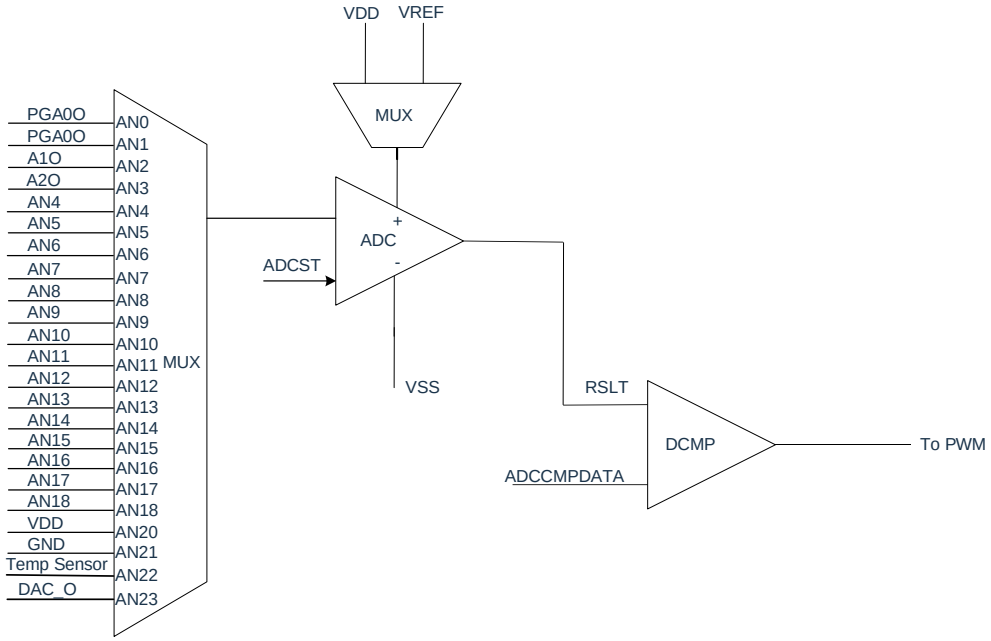
ADC通道号（支持硬件触发）	ADC通道	ADC通道优先级	说明
0	AN0（PGA0O）	最高	PGA0通道(详见第18章)
1	AN1（PGA0O）	-	PGA0通道(详见第18章)
2	AN2（A1O）	-	PGA1通道(详见第18章)
3	AN3（A2O）	-	PGA2通道(详见第18章)
4	AN4（P20）	-	外部通道4
5	AN5（P21）	-	外部通道5
6	AN6（P22）	-	外部通道6
7	AN7（P23）	-	外部通道7
8	AN8（P00）	-	外部通道8
9	AN9（P01）	-	外部通道9
10	AN10（P02）	-	外部通道10
11	AN11（P03）	-	外部通道11
12	AN12（P04）	-	外部通道12
13	AN13（P05）	-	外部通道13
14	AN14（P24）	-	外部通道14
15	AN15（P25）	-	外部通道15
16	AN16（P26）	-	外部通道16
17	AN17（P27）	-	外部通道17
18	AN18（P16）	-	外部通道18
19	-	-	禁止选择
20	AN20（VDD）	-	
21	AN21（GND）	-	
22	AN22（TS）	-	温度传感器通道
23	AN23	最低	内部通道

注：AN0-AN23 任意组合通道支持连续模式转换。

ADC 的内部通道

ADC内部通道号	ADC内部通道	说明
1-3	-	禁止选择
4	IAN_4（DAC_O）	DAC输出通道(详见第20章)

17.3.2 ADC结构框图



17.3.3 ADC的功耗模式

ADC工作模式分为两种：高速模式与低电流模式。

高速模式：该模式下转换速度较快。

低电流模式：该模式转换速度稍慢，ADC的工作电流显著下降。对转换速率要求不高的应用来说，可采用此模式来降低ADC的功耗。此模式的逐次比较时间比高速模式多10个TADCK的时间。

17.3.4 ADC的转换模式

ADC 转换模式分为两种：单次转换模式与连续转换模式。

单次转换模式：

对已使能优先级最高的通道进行一次转换之后结束操作并产生中断标志位。

连续转换模式：

对所有已使能的通道进行转换之后结束操作并产生中断标志位，未使能的通道忽略跳过。

ADCSWCHE=0 时，软件通道开启禁止，ADC 通道的选择与开启由硬件自动控制。

ADCSWCHE=1 时，软件通道开启使能，ADC 通道的选择与开启由 ADCSWCHS 来控制，ADCSWCHS 选择某一通道后，该通道自动开启（ADCEN 必须为 1）。单次模式与连续模式在此条件下实际是对 ADCSWCHS 选择的通道进行转换。

17.3.5 ADC的时钟

ADC 的时钟来源于 APB 时钟，可选择 8 种分频 1/2/4/8/16/32/64/128，通过 ADCCON.ADCDIV 来配置。

高速模式下单次转换模式 AD 转换的时间（TADC）：

2*TADCK（默认开关稳定时间）+13.5*TADCK（默认采样时间）+31.5*TADCK（逐次比较时间）+5*TADCK

高速模式模式下连续转换模式完成一次 AD 转换的时间（TADC）：

2*TADCK（默认开关稳定时间）+13.5*TADCK（默认采样时间）+31.5*TADCK（逐次比较时间）+3*TADCK

ADCSWCHE=1 时，实际的开关稳定时间为选择到某一通道后到开始转换的时间。

17.3.6 ADC的通道选择及中断产生

<table>
<tr><th>ADCCON
(ADCSWCHE)</th><th>ADCSCAN
(ADEn)</th><th>ADCCON
(ADCMS)</th><th>通道说明</th><th>结果存放</th><th>中断产生</th></tr>
<tr><td>0</td><td>0</td><td>0/1</td><td>关闭所有通道</td><td>-</td><td>-</td></tr>
<tr><td rowspan="2">0</td><td rowspan="2">1</td><td>0</td><td>转换 SCAN 中已使能的优先级最高的通道</td><td rowspan="2">单个通道转换完成后，结果存在转换的通道对应的结果寄存器中</td><td rowspan="2">单个通道转换完成后，中断产生在转换的通道对应的中断源中(ADCRISn)</td></tr>
<tr><td>1</td><td>按优先级从高到低依次转换 SCAN 中所有已使能的通道</td></tr>
<tr><td rowspan="2">1</td><td rowspan="2">X</td><td>0</td><td>转换 ADCSWCHS 中设置的通道一次</td><td>结果存在 SCAN 已使能优先级最高的通道对应的结果寄存器中</td><td>转换完成后，中断产生在 SCAN 已使能优先级最高的通道对应的中断源中(ADCRISn)</td></tr>
<tr><td>1</td><td>SCAN 使能多少个有效通道，就连续转换多少次，转换的通道始终为 ADCSWCHS 设置的通道</td><td>单个通道转换完成后，结果按优先级从高到低依次存在 SCAN 使能的通道对应的结果寄存器中</td><td>单个通道转换完成后，中断按优先级从高到低依产生在 SCAN 使能的通道对应的中断源中(ADCRISn)</td></tr>
</table>

注：若 X 等于 0，仍会开始 ADCSWCHS 设置的通道的转换，结果和中断都不会更新到任何寄存器中。

17.3.7 ADC软件启动

在寄存器ADCCON2.ADCST位中写入1，将启动ADC转换。转换完毕后，该位硬件自动清零。
在ADC转换期间，任何软件和硬件触发启动信号将被忽略。

17.3.8 ADC硬件触发启动

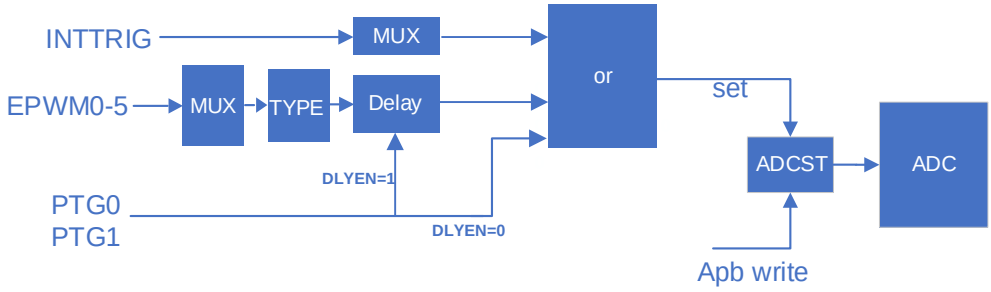
触发源：

ADC除了软件启动转换之外，还可通过硬件触发ADC转换。硬件触发源的种类有：

1) 内部触发
2) EPWM 输出通道触发
3) EPWM 计数比较器 0 触发
4) EPWM 计数比较器 1 触发

不同种类的触发源可同时有效，同一种类的触发源可能含有不同触发信号，如EPWM输出通道触发，可选择EPWM0-EPWM5其中之一的触发信号。

图17-1：ADC硬件触发启动



内部触发：

内部触发包括： ADC，ACMP0，ACMP1，TIMER0/1触发。

ADC：ADC的转换结束

ACMP0：ACMP0的事件输出

ACMP1：ACMP1的事件输出

Timer0：Timer0的已使能中断(TMR0MIS)

Timer1：Timer1的已使能中断(TMR1MIS)

EPWM输出通道触发：

EPWM 输出通道触发可选择上升沿，下降沿，零点，周期点启动 ADC，如检测到 EPWM 触发信号，可选择经过一定延时之后启动 ADC 转换。若 EPWM 的输出通道经过重映射，则 EPWM 触发信号为重映射前的信号，IPGn 信号。

EPWM 输出通道触发仅支持硬件选择通道，可设置单独的 ADC 转换通道。即 EPWM 输出通道触发信号产生后，将按照独有的设置通道转换。EPWM 输出通道触发 ADC 的转换通道在 ADCCHEPWM 寄存器中设置。转换完毕后将恢复为 ADCSCAN 中通道设置。

EPWM 计数器比较器触发：

EPWM 计数比较器 0/1 触发可设置在 EPWMn 周期内任意时刻触发启动 ADC，方式同 EPWM 通道触发相同，也可选择经过一定延时之后启动 ADC 转换。

EPWM 计数比较器 0/1 触发仅支持硬件选择通道，可设置单独的 ADC 转换通道。即触发信号产生后，将按照独自的设置通道转换。EPWM 计数比较器 0 触发 ADC 的转换通道在 ADCCHEPTG0 寄存器中设置。EPWM 计数比较器 1 触发 ADC 的转换通道在 ADCCHEPTG1 寄存器中设置。转换完毕后将恢复为 ADCSCAN 中通道设置。

EPWM触发延时：

ADCEPWMTGDLY 寄存器决定 EPWM 触发启动 ADC 延时的时间：(ADCEPWMTGDLY[9:0]+3)*PCLK

EPWM触发延时的范围如下：

pclk 48MHz（延时范围）	pclk 64MHz（延时范围）
0.041us~21.34us	0.031us~16.03us
0.02us~31.32us	0.015US~16.01us

若 ADCEPWMTGDLY=0,EPWM 比较器 0/EPWM 比较器 1/上升沿/下降沿/周期点/零点延时 3 个 PCLK 时钟启动 ADC 转换。

EPWM 触发启动 ADC 设置

EPWM触发ADC转换在一些应用中有特殊的时间要求。针对该需求，ADC内部支持不同的EPWM触发条件可设置独立的转换通道。例如：

EPWM输出通道触发可选择AN0、AN1、AN2通道转换。

EPWM比较器0触发可选择AN17通道转换。

EPWM比较器1触发可选择AN18通道转换。

软件启动或其他触发启动选择的通道为AN5、AN6、AN7、AN8。

没有EPWM触发条件时，默认的转换通道为AN5-AN8。

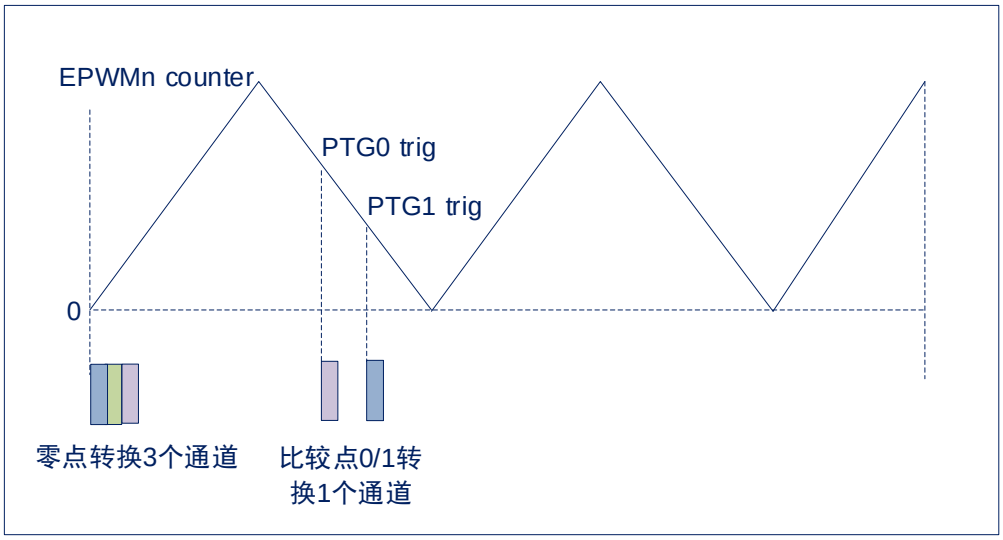
若EPWM的输出通道触发后将仅选择AN0-AN2 共3个通道进行AD转换，转换完毕后自动切换为AN5-AN8通道使能。

若EPWM的比较0触发后，仅选择AN17通道进行AD转换，转换完毕后自动切换为AN5-AN8通道使能。

若EPWM的比较1触发后，仅选择AN18通道进行AD转换，转换完毕后自动切换为AN5-AN8通道使能。

需要注意的是，在AD转换未结束期间，其他的任何触发信号将被忽略。

图17-2：EPWM触发启动ADC设置



注 1：零点时触发 ADC 使能转换的通道由 ADCCHPEM 决定；

注 2：比较点 0 触发 ADC 使能的转换通道由 ADCCHPTG0 决定；

注 3：比较点 1 触发 ADC 使能的转换通道由 ADCCHPTG1 决定；

其他方式启动 ADC 使能的转换通道由 ADCSCAN 或 ADCSWCHS 决定。

17.4 寄存器映射

(ADC基地址 = 0x4006_8000)RO：只读，WO：只写，R/W：读写

寄存器	偏移量	读/写	描述	复位值
CON(P1B)	0x000	R/W	ADC控制寄存器	0xD0000
CON2(P1B)	0x004	R/W	ADC控制寄存器2	0x0
HWTG(P1B)	0x008	R/W	ADC 硬件触发控制寄存器	0x0
PWMTGDLY(P1B)	0x00C	R/W	ADC EPWM触发延时数据寄存器	0x0
SCAN(P1B)	0x010	R/W	ADC 扫描寄存器	0x0
CMP0(P1B)	0x014	R/W	ADC 比较器0控制寄存器	0x0
CMP1(P1B)	0x018	R/W	ADC 比较器1控制寄存器	0x0
IMSC(P1B)	0x01C	R/W	ADC 中断使能寄存器	0x0
RIS	0x020	RO	ADC 中断源状态寄存器	0x0
MIS	0x024	RO	ADC 已使能中断状态寄存器	0x0
ICLR	0x028	WO	ADC 中断清零寄存器	0x0
LOCK	0x02C	R/W	ADC 写使能控制寄存器	0x0
CHEPWM(P1B)	0x030	R/W	ADC EPWM输出触发转换通道寄存器	0x0
CHPTG0(P1B)	0x034	R/W	ADC EPWM 比较器0触发转换通道寄存器	0x0
CHPTG1(P1B)	0x038	R/W	ADC EPWM比较器1触发转换通道寄存器	0x0
--	--	--	--	--
DATA0	0x080	RO	ADC 通道0转换结果寄存器	0x0
DATA1	0x084	RO	ADC 通道1转换结果寄存器	0x0
DATA2	0x088	RO	ADC 通道2转换结果寄存器	0x0
DATA3	0x08C	RO	ADC 通道3转换结果寄存器	0x0
DATA4	0x090	RO	ADC 通道4转换结果寄存器	0x0
DATA5	0x094	RO	ADC 通道5转换结果寄存器	0x0
DATA6	0x098	RO	ADC 通道6转换结果寄存器	0x0
DATA7	0x09C	RO	ADC 通道7转换结果寄存器	0x0
DATA8	0x0A0	RO	ADC 通道8转换结果寄存器	0x0
DATA9	0x0A4	RO	ADC 通道9转换结果寄存器	0x0
DATA10	0x0A8	RO	ADC 通道10转换结果寄存器	0x0
DATA11	0x0AC	RO	ADC 通道11转换结果寄存器	0x0
DATA12	0x0B0	RO	ADC 通道12转换结果寄存器	0x0
DATA13	0x0B4	RO	ADC 通道13转换结果寄存器	0x0
DATA14	0x0B8	RO	ADC 通道14转换结果寄存器	0x0
DATA15	0x0BC	RO	ADC 通道15转换结果寄存器	0x0
DATA16	0x0C0	RO	ADC 通道16转换结果寄存器	0x0
DATA17	0x0C4	RO	ADC 通道17转换结果寄存器	0x0
DATA18	0x0C8	RO	ADC 通道18转换结果寄存器	0x0
--	--	--	--	-
DATA20	0x0D0	RO	ADC 通道20转换结果寄存器	0x0
DATA21	0x0D4	RO	ADC 通道21转换结果寄存器	0x0
DATA22	0x0D8	RO	ADC 通道22转换结果寄存器	0x0
DATA22	0x0DC	RO	ADC 通道23转换结果寄存器	0x0

注：

1 (P1B)标注的寄存器为被保护的寄存器。

(P1B)LOCK==55H时，标注的寄存器允许写入；=其他值，禁止写入。

17.5 寄存器说明

17.5.1 ADC控制寄存器(CON)

位	符号	描述	复位值
31	ADCRST	ADC模块复位控制位 0: --- 1: ADC模块复位	0
30:26	-	保留，必须为0	0x0
25:24	ADMODE10	ADC功耗模式选择位 00: 高速模式 01: 保留，禁止选择 10: 保留，禁止选择 11: 低电流模式	0x0
23:16	ADCNSMP	ADC内部采样时间选择位 00000000: 至 禁止选择 00000100: 00000101: 5.5 ADC clockcycles 00000110: 6.5 ADC clockcycles 00000111: 7.5 ADC clockcycles 00001000: 8.5 ADC clockcycles 00001001: 9.5 ADC clockcycles 00001010: 10.5 ADC clockcycles 00001011: 11.5 ADC clockcycles 00001100: 12.5 ADC clockcycles 00001101: 13.5 ADC clockcycles 00001000: - 11111110: 254.5 ADC clockcycles 11111111: 255.5 ADC clockcycles	0xD
15	-	保留	0
14	-	保留	0
13	ADCSWCHE	ADC通道软件开启使能位 0: 由硬件自动开启 1: 通道开启由ADCSWCHS决定	0
12	ADCNDISEN	ADC充放电功能选择位 0: 放电 1: 充电	0
11:8	ADCNDISTS	ADC充放电时间选择位 0000: 不进行充电或放电 0001: 禁止选择 0010: 2 ADC clockcycles 0011: 3 ADC clockcycles … 1111: 15 ADC clockcycles	0x0
7:6	ADCVS	ADC正端参考选择位 00: 选择VDD 01: 选择VREF	0x0

		10: 保留 11: 禁止选择	
5	-	保留，必须为0	0
4	ADCEN	ADC使能控制位 0: 禁止 1: 使能	0
3	ADCMS	ADC转换模式选择位 0: 单次转换 1: 连续转换 （一次转换完所有使能的ADC通道，顺序为通道0至通道23，没有使能的通道硬件自动忽略，不会产生转换操作）	0
2:0	ADCDIV	ADC时钟预分频选择位 $F_{ADC} = PCLK/2^{ADCDIV}$	0x0

17.5.2 ADC控制寄存器2(CON2)

位	符号	描述	复位值
31:16	-	保留	-
15:13	ADCICHES	ADC内部通道(AN23)选择位 100: 选择内部通道4 其他: 禁止选择	0x0
12	ADCSF4	ADC转换状态标志位4（只读） 0: - 1: 单次转换完成	0
11	ADCSF3	ADC转换状态标志位3（只读） 0：- 1: 转换完成前两个ADC clock cycles	0
10	ADCSF2	ADC转换状态标志位2（只读） 0：- 1：转换完成前两个ADC clock cycles	0
9	ADCSF1	ADC转换状态标志位1（只读） 0: - 1: 转换过程中	0
8	ADCSF0	ADC转换状态标志位1（只读） 0: - 1: 采样过程中	0
7	ADCST	ADC转换开始(转换结束后硬件自动清零) 0: 转换结束或ADC处于空闲模式(写0无效) 1: 开始转换(ADCEN必须为1)	0
6	ADCSMPWAIT	ADC采样时间延长控制位 0: - 1: 采样过程中强制保持采样状态	0
5	BG2ADSEL-	ADC TS通道(AN22)选择位 0: TS温度传感器 1: 保留	0
4:0	ADCSWCHS	ADC通道软件选择选择位 （需要ADCSWCHE=1才能生效） 注：10011禁止选择 00000: 选择通道0 00001: 选择通道1 … … 10111: 选择通道23 其他 保留	0x0

17.5.3 ADC硬件触发控制寄存器(HWTG)

位	符号	描述	复位值
31:16	-	保留	-
15	ADCINTTGEN	ADC 内部功能触发使能位 0: 禁止 1: 使能	0
14:12	ADCINTTGSS	ADC 内部功能触发源通道选择位 000: 保留 001: ADC转换结束信号 010: ACMP0事件 011: ACMP1事件 100: Timer0中断信号 101: Timer1中断信号	0x0
11	ADCPTG1DLYEN	ADC EPWM计数比较器1延时触发使能位 0: 使能 1: 禁止（不经过延时）	-
10	ADCPTG0DLYEN	ADC EPWM计数比较器0延时触发使能位 0: 使能 1: 禁止（不经过延时）	-
9	ADCPTG1EN	ADC EPWM计数比较器1触发使能位 0: 禁止 1: 使能	0
8	ADCPTG0EN	ADC EPWM计数比较器0触发使能位 0: 禁止 1: 使能	0
7	ADCEPWMTEN	ADC EPWM输出触发使能位 0: 禁止 1: 使能	0
6:4	ADCEPWMTSS	ADC EPWM输出触发源通道选择位 000: 触发源为EPWM0 001: 触发源为EPWM1 010: 触发源为EPWM2 011: 触发源为EPWM3 100: 触发源为EPWM4 101: 触发源为EPWM5 11x: 保留	0x0
3:2	-	保留	-
1:0	ADCPEWMTPS	ADC EPWMn触发方式选择位(n=0-5) 00: EPWMn波形的上升沿 01: EPWMn周期点（IPGn） 10: EPWMn波形的下降沿 11: EPWMn的零点（IPGn）	0x0

17.5.4 ADC EPWM触发延时寄存器(EPWMTGDLY)

位	符号	描述	复位值
31:10	-	保留	-
9:0	ADCEPWMTGDLY	ADC EPWM触发延时数据 EPWM（包括输出通道触发与EPWM比较器0/1的触发）延时触发ADC延时数据（详见17.3.8EPWM触发延时部分）。	0x0

注：

上芯片上电后需对EPWMTGDLY寄存器bit12进行写1操作。

17.5.5 ADC扫描寄存器(SCAN)

位	符号	描述	复位值
31:24	-	保留	-
23:0	ADCEn	ADC通道n使能位(n=23-0,n≠19) 0: 禁止 1: 使能	0x0

注：

Bit19保留，且必须为0。

17.5.6 ADC EPWM输出触发转换通道使能寄存器(CHEPWM)

位	符号	描述	复位值
31:24	-	保留	-
23:0	ADCCHEPWMn	ADC EPWM输出触发转换通道使能位(n=23-0,n≠19) 0: 禁止 1: 使能	0x0

注：

Bit19保留，且必须为0。

17.5.7 ADC EPWM比较器0触发转换通道使能寄存器(CHPTG0)

位	符号	描述	复位值
31:24	-	保留	-
23:0	ADCCHPTG0n	ADC EPWM比较器0触发转换通道使能位(n=23-0,n≠19) 0: 禁止 1: 使能	0x0

注：
Bit19保留，且必须为0。

17.5.8 ADC EPWM比较器1触发转换通道使能寄存器(CHPTG1)

位	符号	描述	复位值
31:24	-	保留	-
23:0	ADCCHPTG1n	ADC EPWM比较器1触发转换通道使能位(n=23-0,n≠19) 0: 禁止 1: 使能	0x0

注：
Bit19保留，且必须为0。

17.5.9 ADC转换结果寄存器(DATAx) x=23-0,x≠19

位	符号	描述	复位值
31:12	-	保留	-
11:0	RSLT	ADC 转换结果	0x0

17.5.10 ADC比较控制寄存器0(CMPx) x=0～1

位	符号	描述	复位值
31	ADCCMPxEN	ADC比较器x使能位 0: - 1: 使能	0
30	ADCCMPxO	ADC 比较器 x 结果位（只读） (选择的通道转换完毕后自动更新该位) 0: 不满足比较的条件 1: 满足比较条件	0
29	-	保留	-
28	ADCCMPxCOND	ADC比较器x比较条件选择位 0: ADC结果<预设值 1: ADC结果>=预设值	0
27:24	ADCCMPxMCNT	ADC 比较器 x 匹配次数预设值 指定的通道的模数转换结果和比较条件匹配时，内部计数器加 1，当内部计数器等于 ADCCMPxMCNT+1 的值时，之后内部计数器值自动清零。若累加过程中不满足匹配条件，内部计数器值也将自动清零，即该功能具有滤波功能。 匹配的同时产生 ADC 比较事件，该事件可以作为触发 EPWM 的刹车操作的信号。 注：ADC 比较器 0 比较事件将置中断标志 ADCCMP0IF 为 1；	0x0
23:21	-	保留	-
20:16	ADCCMPxCHS	ADC比较器x比较通道选择位 注：10011禁止选择 00000: 通道0 …… …… 10111: 通道23 其他: 保留	0x0
15:12	-	保留	-
11:0	ADCCMPxDATA	ADC比较器x数据预设值(12位)	0x0

17.5.11 ADC中断使能寄存器(IMSC)

位	符号	描述	复位值
31	ADCIMSC31	ADC比较器0中断使能位 0: 禁止 1: 使能	0
30:24	-	保留	0x0
23:0	ADCIMSCn	ADC 通道n中断使能位(n=23-0,n≠19) 0: 禁止 1: 使能	0x0

注：

Bit19保留，且必须为0。

17.5.12 ADC中断源状态寄存器(RIS)

位	符号	描述	复位值
31	ADCRIS31	ADC比较器0中断源状态 0: 中断源未产生中断 1: 中断源产生中断	0
30:24	-	保留	0x0
23:0	ADCRISn	ADC通道n中断源状态(n=23-0,n≠19) 0: 中断源未产生中断 1: 中断源产生中断	0x0

注：

Bit19保留，且必须为0。

17.5.13 ADC已使能中断状态寄存器(MIS)

位	符号	描述	复位值
31	ADCMIS31	ADC比较器0中断状态 0: 未产生中断 1: 使能且产生中断	0
30:24	-	保留	-
23:0	ADCMISn	ADC通道n中断状态(n=23-0,n≠19) 0: 未产生中断 1: 使能且产生中断	0x0

注：

Bit19保留，且必须为0。

17.5.14 ADC中断清零寄存器(ICLR)

位	符号	描述	复位值
31	ADCICLR31	ADC比较器0中断状态 1: 清零ADC比较器0中断状态 0: 不影响	0
30:24	-	保留	-
23:0	ADCICLRn	ADC通道n中断状态(n=23-0,n≠19) 0: 不影响 1: 清零	0x0

注：

Bit19保留，且必须为0。

17.5.15 ADC写使能控制寄存器(LOCK)

位	符号	描述	复位值
31:8	-	保留	-
7:0	LOCK	当LOCK=0x55时，使能操作ADC相关寄存器 （详见寄ADC存器映射说明） LOCK=其他值时，禁止操作ADC相关寄存器	0x0

第18章 可编程增益放大器（PGA0/1/2）

18.1 概述

芯片包含三个基本运放模块与三个可编程增益放大器。利用少量外围元器件可实现基本的信号放大与信号运算功能。

18.2 特性

PGA0 (可编程增益放大器0)

- 增益可调节：1X/2X/2.5X/5X/7.5X/10X/15X。
- 正端输入可选：从 A0P 输入、PGA_VREF(VREF/2 或 BG)可选。
- 支持伪差分结构，反馈地可选择从外部端口接入。
- PGA0 可选择多种输出方式：
  （1） 可直接输出到 ADC 第 0、1 通道
  （2） 可直接输出到比较器
  （3） 可直接输出到 PAD（A0O）
  （4） 可通过 10K 电阻后输出到 PAD（A0O）

PGA1 (可编程增益放大器1)

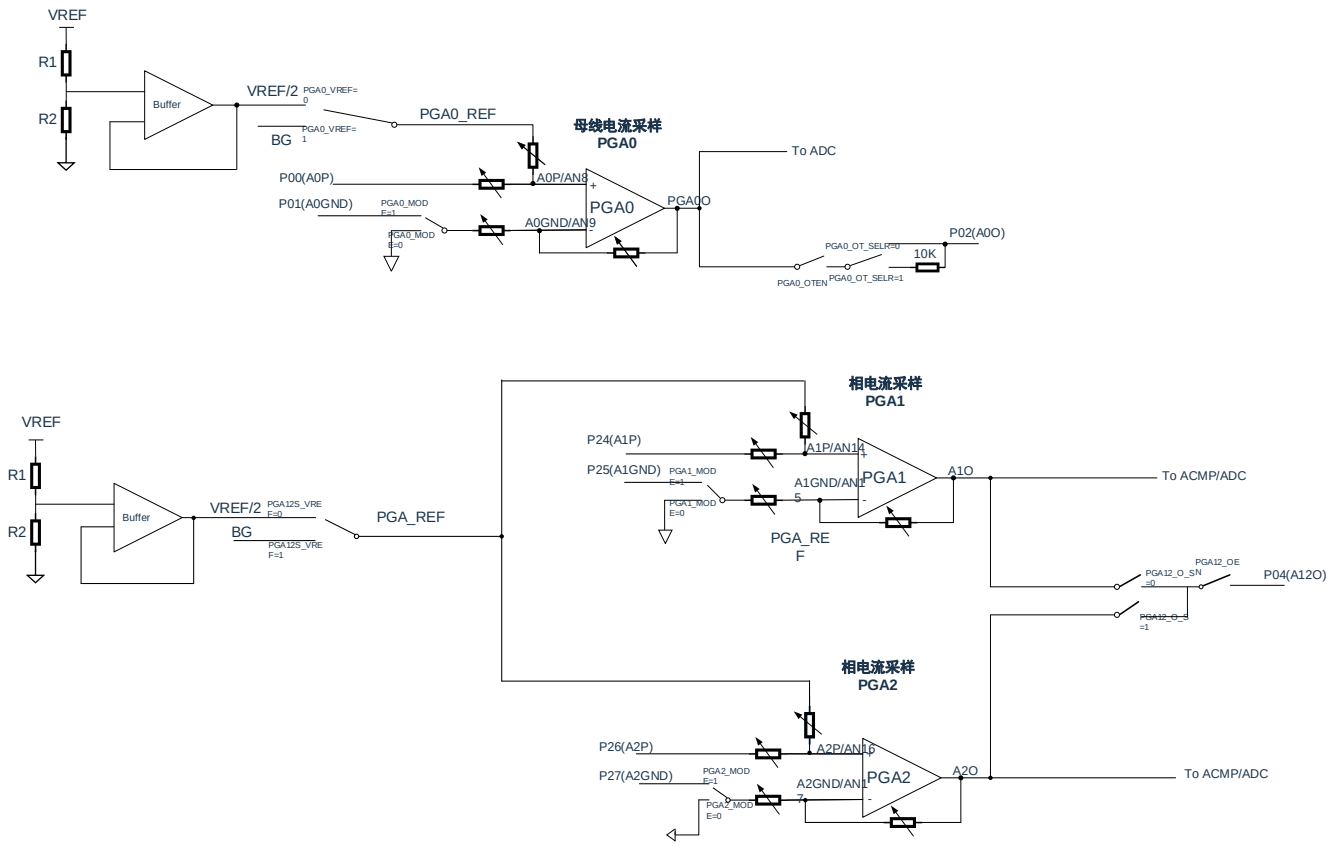
- 增益可调节：1X/2X/2.5X/5X/7.5X/10X/15X。
- 正端输入可选：A1P 输入、PGA_VREF(VREF/2 或 BG)可选。
- 支持伪差分结构，反馈地可选择从外部端口接入。
- PGA1 输出方式可选：
  （1） 可直接输出到 ADC 第 2 通道
  （2） 可直接输出到比较器
  （3） 可直接输出到 PAD（A12O）

PGA2 (可编程增益放大器2)

- 增益可调节：1X/2X/2.5X/5X/7.5X/10X/15X。
- 正端输入可选：A2P 输入、PGA_VREF(VREF/2 或 BG)可选。
- 支持伪差分结构，反馈地可选择从外部端口接入。
- PGA2 输出方式可选：
  （1） 可直接输出到 ADC 第 3 通道
  （2） 可直接输出到比较器
  （3） 可直接输出到 PAD（A12O）

18.3 结构框图

图 18-1：PGA 结构图



18.4 寄存器映射

(PGA0基地址 = 0x4006_8300)

RO：只读，WO：只写，R/W：读写

寄存器	偏移量	读/写	描述	复位值
PGA0CON0	0x000	R/W	PGA0控制寄存器0	0x0
PGA0CON1	0x004	R/W	PGA0控制寄存器1	0x0
PGA0LOCK	0x008	R/W	PGA0访问寄存器使能	0x0

(PGA1/2基地址 = 0x4006_8320)

RO：只读，WO：只写，R/W：读写

寄存器	偏移量	读/写	描述	复位值
PGA1CON0	0x000	R/W	PGA1控制寄存器	0x0
PGA2CON0	0x004	R/W	PGA2控制寄存器	0x0
PGA12CON	0x008	R/W	PGA12控制寄存器	0x0
PGA12LOCK	0x00C	R/W	PGA12访问寄存器使能	0x0

18.5 寄存器说明

18.5.1 PGA0控制寄存器0

位	符号	描述	复位值
31:8	-	保留	-
7	PGA0_EN	PGA0使能位 0: 禁止 1: 使能	0
6	-	保留 -	-
5	PGA0_MODE	PG0模式选择 0: 单端模式 1: 全差分模式	0
4	PGA0S_VREF	PGA0参考电压选择位 0: VREF/2 1: BG（0.8V）	0
3	-	-	-
2:0	PGA0_S	PGA0增益选择 000: 1X 001: 2X 010: 2.5X 011: 5X 100: 7.5X 101: 10X 110: 15X 111: 15X	0x0

18.5.2 PGA0控制寄存器1

位	符号	描述	复位值
31:4	-	保留	-
3	PGA0_OTEN	PGA0输出到PAD通道使能 0: 禁止 1: 使能	0
2:1	-	保留	-
0	PGA0_OT_SELR	PGA0输出到PAD串联电阻选择 0: 内部不串电阻 1: 内部串10K电阻	0

18.5.3 PGA0访问寄存器使能

位	符号	描述	复位值
31:8	-	保留	-
7:0	PGA0_LOCK	PGA0寄存器访问使能位 0x55: 运行访问PGA0相关寄存器 其他: 禁止访问	0x0

18.5.4 PGA1控制寄存器0

位	符号	描述	复位值
31:8	-	保留	-
7	PGA1_EN	PGA1使能位 0: 禁止 1: 使能	0
6:5	-	保留	-
4	PGA1_MODE	PGA1模式选择 0: 单端模式 1: 全差分模式	0
3	-	-	-
2:0	PGA1_S	PGA1增益选择 000: 1X 001: 2X 010: 2.5X 011: 5X 100: 7.5X 101: 10X 110: 15X 111: 15X	0x0

18.5.5 PGA2控制寄存器0

位	符号	描述	复位值
31:8	-	保留	-
7	PGA2_EN	PGA2使能位 0: 禁止 1: 使能	0
6:5	-	保留 -	-
4	PGA2_MODE	PGA2模式选择 0: 单端模式 1: 全差分模式	0
3	-	-	-
2:0	PGA2_S	PGA2增益选择 000: 1X 001: 2X 010: 2.5X 011: 5X 100: 7.5X 101: 10X 110: 15X 111: 15X	0x0

18.5.6 PGA12控制寄存器0

位	符号	描述	复位值
31:18	-	保留	-
17	PGA12_OEN	PGA2/PGA1输出使能位 0: 禁止 1: 使能	0
16	PGA12_O_S	PGA2/PGA1输出选择位 0: PGA1输出 1: PGA2输出	0
15:10	-	- -	-
0	PGA12S_VREF	PGA1/PGA2 参考电压选择 0: VREF/2 1: BG（0.8V）	0

18.5.7 PGA1/PGA2访问寄存器使能

位	符号	描述	复位值
31:8	-	保留	-
7:0	PGA12_LOCK	PGA1/PGA2寄存器访问使能位 0x55: 运行访问PGA1/PGA2相关寄存器 其他: 禁止访问	0x0

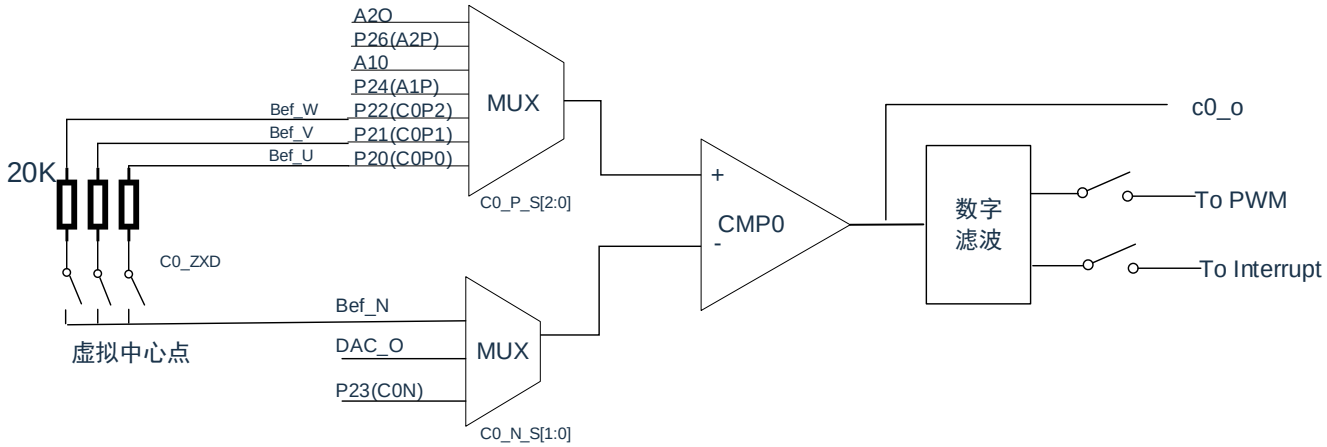
第19章 模拟比较器（ACMP0/1）

19.1 概述

芯片内部包含两个模拟比较器。可按照比较器的配置适用于不同的应用场合。当正端电压大于负端电压时，比较器输出逻辑1，反之输出0，也可以通过输出极性选择位进行改变。当比较器输出值发生改变时，每路比较器都可通过配置产生中断。

19.2 结构框图

图19-1：比较器结构框图



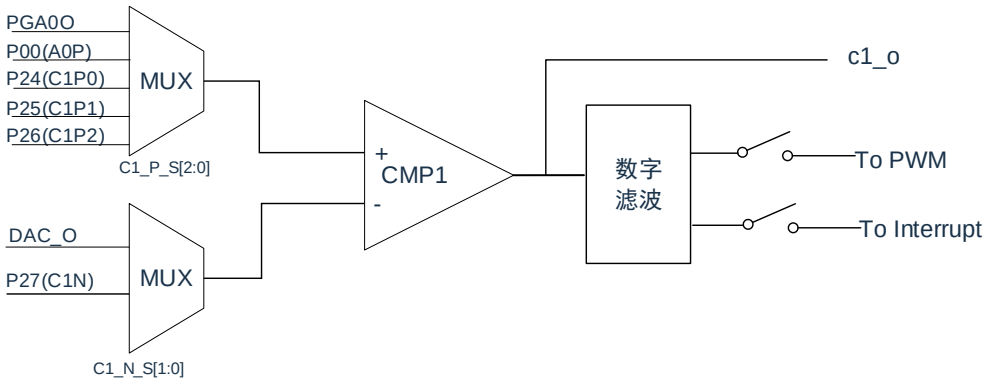
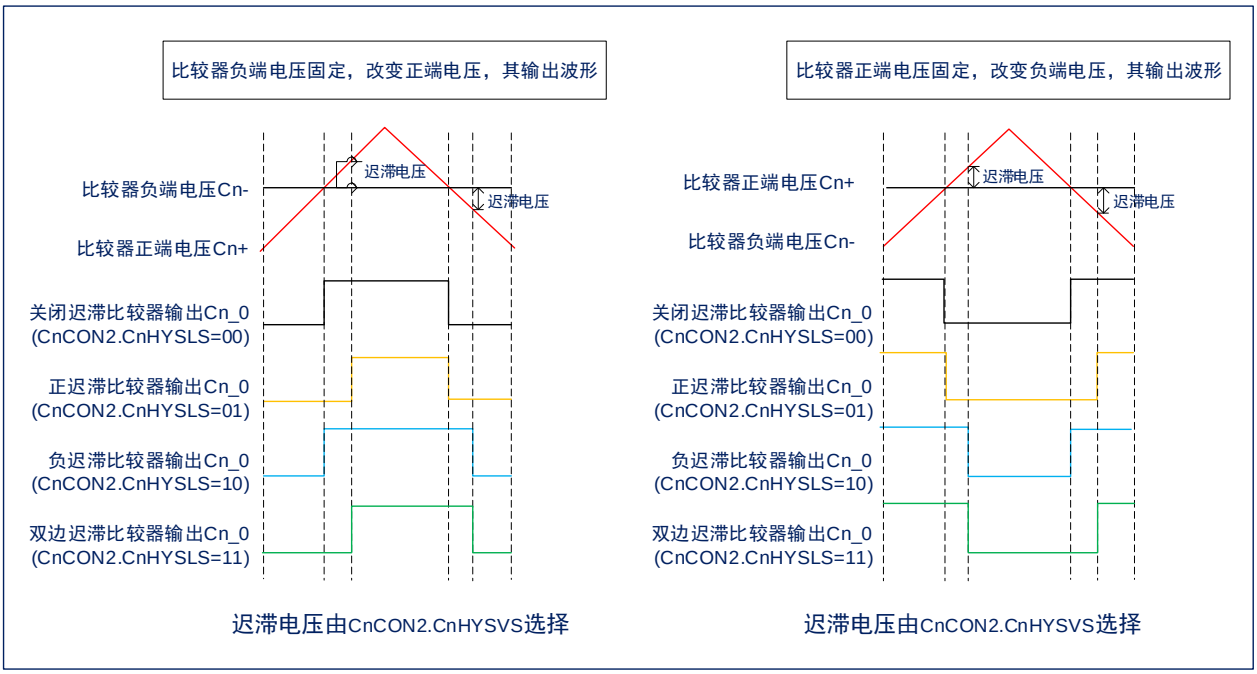


图19-2：比较器迟滞功能结构框图

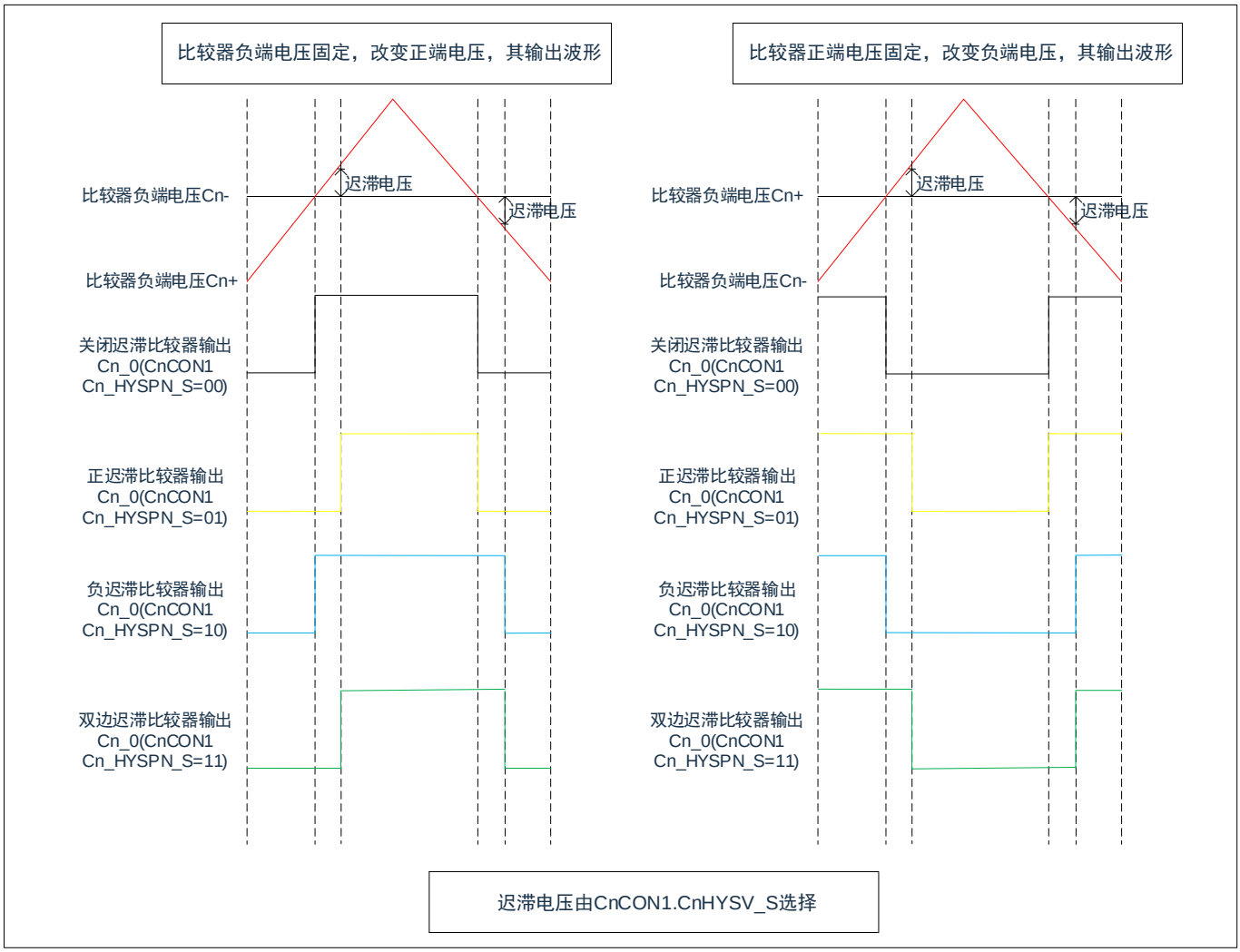


19.3 特性

- 模拟输入电压范围：(0~VDD)V。
- 支持单边/双边迟滞功能。
- 支持迟滞电压选择(10mV/20mV/60mV-典型值)。
- 每个比较器正端可多路选择
- 每个比较器负端可选择端口输入与内部参考电压。
- 输出可滤波时间可选择：0~512*Tsys。
- 比较器事件输出可作为增强型 PWM 的刹车触发信号。
- 输出改变可产生中断。

19.4 功能说明

图19-3：比较器迟滞功能结构框图



19.5 寄存器映射

(ACMP基地址 = 0x4006_8200)RO：只读，WO：只写，R/W：读写

寄存器	偏移量	读/写	描述	复位值
C0CON0(P1B)	0x000	R/W	模拟比较器0控制寄存器0	0x0
C0CON1(P1B)	0x004	R/W	模拟比较器0控制寄存器1	0x0
C1CON0(P1B)	0x08	R/W	模拟比较器1控制寄存器0	0x0
C1CON1(P1B)	0x0C	R/W	模拟比较器1控制寄存器1	0x0
CEVCON(P1B)	0x010	R/W	模拟比较器事件控制寄存器	0x0
IMSC(P1B)	0x014	R/W	模拟比较器中断使能寄存器	0x0
RIS	0x018	RO	模拟比较器中断源状态寄存器	0x0
MIS	0x01C	RO	模拟比较器已使能中断状态寄存器	0x0
ICLR	0x020	WO	模拟比较器中断清零寄存器	0x0
LOCK	0x024	R/W	模拟比较器写使能寄存器	0x0

注：

(P1B)标注的寄存器为被保护的寄存器。

(P1B)：LOCK==55H时，标注的寄存器允许写入；=其他值，禁止写入。

19.6 寄存器说明

19.6.1 模拟比较器0控制寄存器0(C0CON0)

位	符号	描述	复位值
31:17	-	保留	-
16	C0_ZXD	模拟比较器0中心点选择使能 0: 禁止 1: 使能	
15	C0_EN	模拟比较器0使能位 0: 禁止 1: 使能	0
14	C0_OEN	模拟比较器0输出使能位 0: 禁止 1: 使能	0
13:9	-	保留 -	-
8	C0_OUT	模拟比较器0结果位(只读）	
7	-	保留	-
6:4	C0_P_S	模拟比较器0正端通道选择 000: C0P0 001: C0P1 010: C0P2 011: A1P 100: A1O 101: A2P 110: A2O 111: 禁止	0x0
3:2	-	保留	-
1:0	C0_N_S	模拟比较器0负端通道选择 00: C0N 01: DAC_O 10: Bef_N虚拟中心点（比较器内部信号） 11: 禁止	0x0

19.6.2 模拟比较器0控制寄存器1(C0CON1)

位	符号	描述	复位值
31:14	-	保留	-
13:12	C0_HYSPN_S	模拟比较器0迟滞方式选择 00: 无迟滞 01: 正迟滞 10: 负迟滞 11: 正负迟滞	0x0
11:10	C0_HYSV_S	模拟比较器0迟滞电压选择 00: 无迟滞 01: 10mV 10: 20mV 11: 60mV	0x0
9	C0_POS	模拟比较器0输出极性选择位 0: 正常输出 1: 反相输出	0
8	C0_FE	模拟比较器0输出滤波使能位 0: 禁止 1: 使能	0
7:4	-	保留	-
3:0	C0_FS	模拟比较器0输出滤波时间选择位 0000: (0~1)*Tpclk 0001: (1~2)*Tpclk 0010: (2~3)*Tpclk 0011: (4~5)*Tpclk 0100: (8~9)*Tpclk 0101: (16~17)*Tpclk 0110: (32~33)*Tpclk 0111: (64~65)*Tpclk 1000: (128~129)*Tpclk 1001: (256~257)*Tpclk 1010: (512~513)*Tpclk 其他: (0~1)*Tpclk	0x0

19.6.3 模拟比较器1控制寄存器0(C1CON0)

位	符号	描述	复位值
31:16	-	保留	-
15	C1_EN	模拟比较器1使能位 0: 禁止 1: 使能	0
14	C1_OEN	模拟比较器1输出使能位 0: 禁止 1: 使能	0
13:9	-	保留 -	-
8	C1_OUT	模拟比较器1结果位(只读）	
7	-	保留 -	-
6:4	C1_P_S	模拟比较器1正端通道选择 000: C1P0 001: C1P1 010: C1P2 011: C1P3/A0P 100: PGA00（PGA0输出） 其他: 禁止	0x0
3:2	-	保留	-
1:0	C1_N_S	模拟比较器1负端通道选择 00: C1N 01: DAC_O 其他: 禁止	0x0

19.6.4 模拟比较器1控制寄存器1(C1CON1)

位	符号	描述	复位值
31:14	-	保留	-
13:12	C1HYSPN_S	模拟比较器1迟滞方式选择 00: 无迟滞 01: 正迟滞 10: 负迟滞 11: 正负迟滞	0x0
11:10	C1_HYSV_S	模拟比较器1迟滞电压选择 00: 无迟滞 01: 10mV 10: 20mV 11: 60mV	0x0
9	C1_POS	模拟比较器1输出极性选择位 0: 正常输出 1: 反相输出	0
8	C1_FE	模拟比较器1输出滤波使能位 0: 禁止 1: 使能	0
7:4	-	保留	-
3:0	C1_FS	模拟比较器1输出滤波时间选择位 0000: (0~1)*Tpclk 0001: (1~2)*Tpclk 0010: (2~3)*Tpclk 0011: (4~5)*Tpclk 0100: (8~9)*Tpclk 0101: (16~17)*Tpclk 0110: (32~33)*Tpclk 0111: (64~65)*Tpclk 1000: (128~129)*Tpclk 1001: (256~257)*Tpclk 1010: (512~513)*Tpclk 其他: (0~1)*Tpclk	0x0

19.6.5 模拟比较器事件控制寄存器(CEVCON)

位	符号	描述	复位值
31:6	-	保留	-
5	EVE1	模拟比较器1事件输出使能位 (不影响中断产生) 0: 禁止 1: 使能	0
4	EVE0	模拟比较器0事件输出使能位 (不影响中断产生) 0: 禁止 1: 使能	0
3:2	EVS1	模拟比较器1事件产生条件选择位 00: 比较器1输出从0->1的跳变 01: 比较器1输出从1->0的跳变 10: 比较器1输出从0->1的跳变或从1->0的跳变 11: 保留	0x0
1:0	EVS0	模拟比较器0事件产生条件选择位 00: 比较器0输出从0->1的跳变 01: 比较器0输出从1->0的跳变 10: 比较器0输出从0->1的跳变或从1->0的跳变 11: 保留	0x0

19.6.6 模拟比较器中断使能寄存器(IMSC)

位	符号	描述	复位值
31:2	-	保留	-
1	EN_C1IF	模拟比较器1中断使能位 0: 禁止 1: 允许	0
0	EN_C0IF	模拟比较器0中断使能位 0: 禁止 1: 允许	0

19.6.7 模拟比较器中断源状态寄存器(RIS)

位	符号	描述	复位值
31:2	-	保留	-
1	RIS_C1IF	模拟比较器1中断源状态位 0: 未产生中断 1: 已产生中断(事件产生)	0
0	RIS_C0IF	模拟比较器0中断源状态位 0: 未产生中断 1: 已产生中断(事件产生)	0

19.6.8 模拟比较器已使能中断源状态寄存器(MIS)

位	符号	描述	复位值
31:2	-	保留	-
1	MIS_C1IF	模拟比较器1已使能中断状态位 0: 未产生中断 1: 已产生中断	0
0	MIS_C0IF	模拟比较器0已使能中断状态位 0: 未产生中断 1: 已产生中断	0

19.6.9 模拟比较器中断清零控制寄存器(ICLR)

位	符号	描述	复位值
31:2	-	保留	-
1	ICLR_C1IF	模拟比较器1中断清零控制位 0: 不影响 1: 清零RIS_C1IF标志位	0
0	ICLR_C0IF	模拟比较器0中断清零控制位 0: 不影响 1: 清零RIS_C0IF标志位	0

19.6.10 模拟比较器写使能控制寄存器(LOCK)

位	符号	描述	复位值
31:8	-	保留	-
7:0	LOCK	当LOCK=0x55时，使能操作ACMP相关寄存器（详见寄ACMP存器映射说明） LOCK=其他值时，禁止操作ACMP相关寄存器	0x0

第20章 DAC

20.1 概述

芯片内部包含一个数模转换器

20.2 结构框图

20.3 特性

- 模拟参考电压输入为 ADCLDO 的输出。
- 输出电压多级可选。

20.4 寄存器映射

(DAC基地址 = 0x4006_8360)RO：只读，WO：只写，R/W：读写

寄存器	偏移量	读/写	描述	复位值
CON0	0x000	R/W	DAC控制寄存器0	0x0
LOCK	0x004	R/W	DAC寄存器使能控制位	0x0

注：

(P1B)标注的寄存器为被保护的寄存器。

(P1B)：LOCK==55H时，标注的寄存器允许写入；=其他值，禁止写入。

20.5 寄存器说明

20.5.1 DAC控制寄存器0(CON0)

位	符号	描述	复位值
31:9	-	保留	-
8	DAC_EN	DAC模块使能 0: 禁止 1: 使能	0
7:0	DAC_S	DAC数字信号输入	0x0

20.5.2 DAC写使能控制寄存器(LOCK)

位	符号	描述	复位值
31:8	-	保留	-
7:0	LOCK	当LOCK=0x55时，使能操作DAC相关寄存器 （详见寄ACMP存器映射说明） LOCK=其他值时，禁止操作DAC相关寄存器	0x0

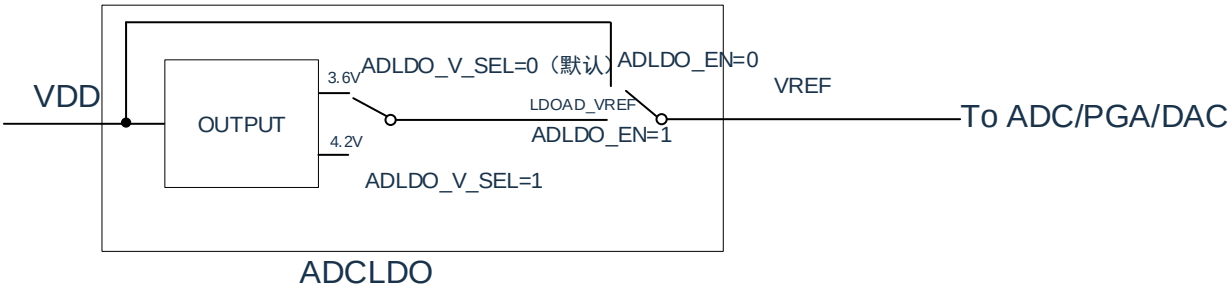
第21章 ADCLDO概述

芯片内部包含一个LDO，可向部分模块提供参考电压

21.1 特性

- 模拟输入电压范围：VDD。
- 输出电压：VDD，4.2V，3.6V 可选。

21.2 结构框图



21.3 寄存器映射

(ADCLDO基地址 = 0x4006_8340)RO：只读，WO：只写，R/W：读写

寄存器	偏移量	读/写	描述	复位值
CON0	0x000	R/W	ADCLDO控制寄存器0	0x0
LOCK	0x00C	R/W	ADCLDO寄存器使能控制位	0x0

注：LOCK==55H时，CON0允许写入；=其他值，禁止写入。

21.4 寄存器说明

21.4.1 ADCLDO控制寄存器0(CON0)

位	符号	描述	复位值
31:21	-	保留	-
20:16	ADCLDO_AJ[4:0]	ADCLDO修调位（只读）	
15:9	-	保留	0x0
8	ADCLDO_EN	ADCLDO模块使能 0: 禁止，ADLDO输出VDD 1: 使能，ADLDO输出LDO电压	0
7:0	ADCLDO_V_SEL	ADCLDO输出电压选择 0x55: LDO电压输出4.2V 其他: LDO电压输出3.6V	0x0

21.4.2 ADCLDO写使能控制寄存器(LOCK)

位	符号	描述	复位值
31:8	-	保留	-
7:0	LOCK	当LOCK=0x55时，使能操作ADCLDO相关寄存器（详见寄ACMP存器映射说明） LOCK=其他值时，禁止操作ADCLDO相关寄存器	0x0

第22章 嵌套向量中断控制器（NVIC）

Cortex®-M0+ CPU提供一个用于中断处理的嵌套向量中断控制器（NVIC）。

22.1 特性

- 支持嵌套向量中断。
- 自动保存和恢复处理器状态。
- 动态改变优先级。
- 简化和确定的中断时间。

NVIC按照优先级处理所有支持的异常。所有的异常在“Handler模式”处理。NVIC支持21个(IRQ[31:0])离散中断，每个中断支持4级中断优先级。所有的中断和大部分系统异常可以配置成不同的优先级。当一个中断发生时，NVIC将比较新中断与当前中断的优先级，如果新中断优先级高，则立即处理新中断。

当接受一个中断后，中断服务程序（ISR）的开始地址可从内存中的向量表取得。软件不需要决定哪个中断被响应，也不用分配相关ISR的开始地址。当取得开始地址后，NVIC将自动保存处理器状态寄存器（PC、PSR、LR、R0~R3、R12）的值到栈中。在ISR结束后，NVIC将从栈中恢复相关寄存器的值，并运行在正常状态。因此花费少量且确定的时间处理中断请求。

NVIC支持“末尾连锁”，可以有效的处理背对背中断，即无需保存和恢复当前状态，从而减少结束当前ISR切换到挂起ISR的延迟时间。NVIC还支持“Late Arrival”，因此可以提高并发中断的效率。当较高优先级中断请求发生在当前ISR开始执行之前（保存处理器状态和获取起始地址阶段），NVIC将立即处理更高优先级的中断，从而提高实时性。

更多详细信息，请参考“ARM®Cortex®-M0+ 技术参考手册”和“ARM®v6-M 架构参考手册”。

22.2 异常模式和系统中断映射

下表列出了该系列产品支持的异常模式。与所有中断一样，软件可以为其中一些异常设置4级优先级。用户可配置的最高优先级为0，最低优先级为3。所有用户可配置中断的默认优先级为0。

异常名称	异常编号	优先级
Reset	1	-3
NMI	2	-2
Hard Fault	3	-1
Reserved	4~10	保留
SVCall	11	可配置
Reserved	12~13	保留
PendSV	14	可配置
SysTick	15	可配置
Interrupt (IRQ0~IRQ31)	16~47	可配置

注：优先级0在系统为第4优先级，排在“Reset”、““NMI”和“Hard Fault”三个系统异常之后。

22.3 向量表

异常编号	中断号	向量地址	异常类型	描述
1-15	-	0x00-0x3c	系统异常	-
16	0	0x40	INTLVI	电压检测
17	1	0x44	INTP0	引脚输入的边沿检测
18	2	0x48	INTP1	引脚输入的边沿检测
19	3	0x4c	INTP2	引脚输入的边沿检测
20	4	0x50	INTP3	引脚输入的边沿检测
21	5	0x54	INTTM01H	定时器通道01计数或捕捉结束（高8位定时器工作）
22	6	0x58	INTCCP	CCP中断
23	7	0x5c	INTEPWM	EPWM中断
24	8	0x60	INTADC	ADC中断
25	9	0x64	INTACMP	比较器完成中断
26	10	0x68	INTUART0	UART0中断
27	11	0x6c	INTI2C	IIC中断
28	12	0x70	INTSPI	SPI中断
29	13	0x74	INTTIMER0	TIMER0中断
30	14	0x78	INTTIMER1	TIMER1中断
31	15	0x7c	INTLSITIMER	LSI定时器中断
32	16	0x80	保留	-
33	17	0x84	INTTM00	定时器通道00计数结束或捕捉结束
34	18	0x88	INTTM01	定时器通道01计数结束或捕捉结束
35	19	0x8c	INTTM02	定时器通道02计数结束或捕捉结束
36	20	0x90	INTTM03	定时器通道03计数结束或捕捉结束
37	21	0x94	保留	-
38	22	0x98	保留	-
39	23	0x9c	保留	-
40	24	0xa0	保留	-
41	25	0xa4	保留	-
42	26	0xa8	保留	-
43	27	0xac	保留	-
44	28	0xb0	保留	-
45	29	0xb4	保留	-
46	30	0xb8	保留	-
47	31	0xbc	INTFL	FLASH编程完成

22.4 寄存器映射

(NVIC基地址 = 0xE000_E000)RO：只读；WO：只写；R/W：读写。

寄存器	偏移量	读/写	描述	复位值
ISER	0x100	R/W	中断设置使能控制寄存器	0x0
ICER	0x180	R/W	中断清除使能控制寄存器	0x0
ISPR	0x200	R/W	中断设置挂起控制寄存器	0x0
ICPR	0x280	R/W	中断清除挂起控制寄存器	0x0
IPR0	0x400	R/W	IRQ0~IRQ3中断优先级寄存器	0x0
IPR1	0x404	R/W	IRQ4~IRQ7中断优先级寄存器	0x0
IPR2	0x408	R/W	IRQ8~IRQ11中断优先级寄存器	0x0
IPR3	0x40C	R/W	IRQ12~IRQ15中断优先级寄存器	0x0
IPR4	0x410	R/W	IRQ16~IRQ19中断优先级寄存器	0x0
IPR5	0x414	R/W	IRQ20~IRQ23中断优先级寄存器	0x0
IPR6	0x418	R/W	IRQ24~IRQ27中断优先级寄存器	0x0
IPR7	0x41C	R/W	IRQ28~IRQ31中断优先级寄存器	0x0

(INTM基地址 = 0x4004_5B38) RO：只读；WO：只写；R/W：读写。

寄存器	偏移量	读/写	描述	复位值
EGP0	0x000	R/W	外部中断上升沿允许寄存器	0x0
EGN0	0x001	R/W	外部中断下降沿允许寄存器	0x0

22.5 寄存器说明

22.5.1 中断设置使能控制寄存器（ISER）

位	符号	描述	复位值
31:0	SETENA	中断使能位 使能一个或多个中断。每一位代表一个从IRQ0~IRQ31的中断 （向量号从16~47）。 写操作： 0: 无效 1: 使能相关中断 读操作： 0: 相关中断状态禁止 1: 相关中断状态使能 注：读该寄存器值表明当前使能状态。	0x0

22.5.2 中断清使能控制寄存器（ICER）

位	符号	描述	复位值
31:0	CLRENA	中断禁用位 禁用一个或多个中断。每一位表示一个从IRQ0~IRQ31的中断（向量号从16~47）。 写操作： 0: 无效 1: 禁止相关中断 读操作： 0: 相关中断状态禁止 1: 相关中断状态使能 注：读该寄存器值表明当前使能状态。	0x0

22.5.3 中断设置挂起控制寄存器（ISPR）

位	符号	描述	复位值
31:0	SETPEND	设置中断挂起位 写操作： 0: 无效 1: 设置挂起状态。每一位表示一个从IRQ0~IRQ31的中断（向量号从16~47） 读操作： 0: 相关中断不在挂起状态 1: 相关中断在挂起状态 注：读该寄存器值表明当前挂起状态。	0x0

22.5.4 中断清挂起控制寄存器（ICPR）

位	符号	描述	复位值
31:0	CLRPEND	清中断挂起位 写操作： 0: 无效 1: 清除挂起状态。每一位表示一个从IRQ0~IRQ31的中断（向量号从16~47） 读操作： 0: 相关中断不在挂起状态 1: 相关中断在挂起状态 注：读该寄存器值表明当前挂起状态。	0x0

22.5.5 IRQ0~IRQ3中断优先级寄存器（IPR0）

位	符号	描述	复位值
31:30	PRI_3	IRQ3优先级 0表示最高优先级，3表示最低优先级	0x0
29:24	-	保留	-
23:22	PRI_2	IRQ2优先级 0表示最高优先级，3表示最低优先级	0x0
21:16	-	保留	-
15:14	PRI_1	IRQ1优先级 0表示最高优先级，3表示最低优先级	0x0
13:8	-	保留	-
7:6	PRI_0	IRQ0优先级 0表示最高优先级，3表示最低优先级	0x0
5:0	-	保留	-

22.5.6 IRQ4~IRQ7中断优先级寄存器（IPR1）

位	符号	描述	复位值
31:30	PRI_7	IRQ7 优先级 0 表示最高优先级，3 表示最低优先级	0x0
29:24	-	保留	-
23:22	PRI_6	IRQ6 优先级 0 表示最高优先级，3 表示最低优先级	0x0
21:16	-	保留	-
15:14	PRI_5	IRQ5 优先级 0 表示最高优先级，3 表示最低优先级	0x0
13:8	-	保留	-
7:6	PRI_4	IRQ4 优先级 0 表示最高优先级，3 表示最低优先级	0x0
5:0	-	保留	-

22.5.7 IRQ8~IRQ11中断优先级寄存器（IPR2）

位	符号	描述	复位值
31:30	PRI_11	IRQ11 优先级 0 表示最高优先级，3 表示最低优先级	0x0
29:24	-	保留	-
23:22	PRI_10	IRQ10 优先级 0 表示最高优先级，3 表示最低优先级	0x0
21:16	-	保留	-
15:14	PRI_9	IRQ9 优先级 0 表示最高优先级，3 表示最低优先级	0x0
13:8	-	保留	-
7:6	PRI_8	IRQ8 优先级 0 表示最高优先级，3 表示最低优先级	0x0
5:0	-	保留	-

22.5.8 IRQ12~IRQ15中断优先级寄存器（IPR3）

位	符号	描述	复位值
31:30	PRI_15	IRQ15 优先级 0 表示最高优先级，3 表示最低优先级	0x0
29:24	-	保留	-
23:22	PRI_14	IRQ14 优先级 0 表示最高优先级，3 表示最低优先级	0x0
21:16	-	保留	-
15:14	PRI_13	IRQ13 优先级 0 表示最高优先级，3 表示最低优先级	0x0
13:8	-	保留	-
7:6	PRI_12	IRQ12 优先级 0 表示最高优先级，3 表示最低优先级	0x0
5:0	-	保留	-

22.5.9 IRQ16~IRQ19中断优先级寄存器（IPR4）

位	符号	描述	复位值
31:30	PRI_19	IRQ19 优先级 0 表示最高优先级，3 表示最低优先级	0x0
29:24	-	保留	-
23:22	PRI_18	IRQ18 优先级 0 表示最高优先级，3 表示最低优先级	0x0
21:16	-	保留	-
15:14	PRI_17	IRQ17 优先级 0 表示最高优先级，3 表示最低优先级	0x0
13:8	-	保留	-
7:6	PRI_16	IRQ16 优先级 0 表示最高优先级，3 表示最低优先级	0x0
5:0	-	保留	-

22.5.10 IRQ20~IRQ23中断优先级寄存器（IPR5）

位	符号	描述	复位值
31:30	PRI_23	IRQ23 优先级 0 表示最高优先级，3 表示最低优先级	0x0
29:24	-	保留	-
23:22	PRI_22	IRQ22 优先级 0 表示最高优先级，3 表示最低优先级	0x0
21:16	-	保留	-
15:14	PRI_21	IRQ21 优先级 0 表示最高优先级，3 表示最低优先级	0x0
13:8	-	保留	-
7:6	PRI_20	IRQ20 优先级 0 表示最高优先级，3 表示最低优先级	0x0
5:0	-	保留	-

22.5.11 IRQ24~IRQ27中断优先级寄存器（IPR6）

位	符号	描述	复位值
31:30	PRI_27	IRQ27优先级 0表示最高优先级，3表示最低优先级	0x0
29:24	-	保留	-
23:22	PRI_26	IRQ26优先级 0表示最高优先级，3表示最低优先级	0x0
21:16	-	保留	-
15:14	PRI_25	IRQ25优先级 0表示最高优先级，3表示最低优先级	0x0
13:8	-	保留	-
7:6	PRI_24	IRQ24优先级 0表示最高优先级，3表示最低优先级	0x0
5:0	-	保留	-

22.5.12 IRQ28~IRQ31中断优先级寄存器（IPR7）

位	符号	描述	复位值
31:30	PRI_31	IRQ31 优先级 0 表示最高优先级，3 表示最低优先级	0x0
29:24	-	保留	-
23:22	PRI_30	IRQ30 优先级 0 表示最高优先级，3 表示最低优先级	0x0
21:16	-	保留	-
15:14	PRI_29	IRQ29 优先级 0 表示最高优先级，3 表示最低优先级	0x0
13:8	-	保留	-
7:6	PRI_28	IRQ28 优先级 0 表示最高优先级，3 表示最低优先级	0x0
5:0	-	保留	-

22.5.13 外部中断上升沿允许寄存器（EGP0）

EGP0 和 EGN0 寄存器用于设定 INTP0～INTP3 的有效边沿。通过 8 位存储器操作指令设定 EGP0、EGN0 寄存器。

在产生复位信号后，这些寄存器的值变为“00H”。

图22-1：外部中断上升沿允许寄存器（EGP0）和外部中断下降沿允许寄存器（EGN0）的格式

位	符号	描述	复位值
7:4	-	保留	0x0
3	EGP3	INTP3 中断上升沿允许寄存器： 0：禁止外部中断上升沿 1：允许外部中断上升沿	0
2	EGP2	INTP2 部中断上升沿允许寄存器： 0：禁止外部中断上升沿 1：允许外部中断上升沿	0
1	EGP1	INTP1 外部中断上升沿允许寄存器： 0：禁止外部中断上升沿 1：允许外部中断上升沿	0
0	EGP0	INTP0 外部中断上升沿允许寄存 0：禁止外部中断上升沿 1：允许外部中断上升沿	0

22.5.14 外部中断下降沿允许寄存器（EGN0）

位	符号	描述	复位值
7:4	-	保留	0x0
3	EGN3	INTP3 外部中断下降沿允许寄存器： 0：禁止外部中断下降沿 1：允许外部中断下降沿	0
2	EGN	INTP2 外部中断下降沿允许寄存器： 0：禁止外部中断下降沿 1：允许外部中断下降沿	0
1	EGN	INTP1 外部中断下降沿允许寄存器： 0：禁止外部中断下降沿 1：允许外部中断下降沿	0
0	EGN	INTP0 外部中断下降沿允许寄存器： 0：禁止外部中断下降沿 1：允许外部中断下降沿	0

EGPn	EGNn	INTPn引脚的有效边沿选择（n=0～3）
0	0	禁止检测边沿。
0	1	下降沿
1	0	上升沿
1	1	上升和下降的双边沿

对应EGPn位和EGNn位的端口如表22-1所示。

表22-1：对应EGPn位和EGNn位的中断请求信号

检测允许位		中断请求信号
EGP0	EGN0	INTP0
EGP1	EGN1	INTP1
EGP2	EGN2	INTP2
EGP3	EGN3	INTP3

注1：如果将外部中断功能使用的输入端口切换到输出模式，就可能检测到有效边沿而产生INTPn中断。当切换到输出模式时，必须在禁止检测边沿后（EGPn、EGNn=0、0）将端口模式寄存器（PMxx）置“0”。

注2：有关边沿检测的端口，请参照“3.1端口功能”。

注3：n=0～3

第23章 待机功能

23.1 待机功能

待机功能是进一步降低系统工作电流的功能，有以下2种模式。

(1)睡眠模式

睡眠模式是停止 CPU 运行时钟的模式。在设定睡眠模式前，如果高速内部振荡器或者低速内部振荡器正在振荡，各时钟就继续振荡。虽然此模式无法让工作电流降到深度睡眠模式的程度，但是在想要通过中断请求立即重新开始处理或者想要频繁地进行间歇运行时是一种有效的模式。

(2)深度睡眠模式

深度睡眠模式是停止高速内部振荡器的振荡并且停止整个系统的模式。能大幅度地减小 CPU 的工作电流。

因为深度睡眠模式能通过中断请求来解除，所以也能进行间歇运行。但是，因为在解除深度睡眠模式时需要确保振荡稳定的等待时间，所以如果需要通过中断请求立即开始处理，就必须选择睡眠模式。

在任何一种模式中，寄存器、标志和数据存储器全部保持设定为待机模式前的内容，并且还保持输入/输出端口的输出锁存器和输出缓冲器的状态。

注1：在转移到深度睡眠模式时，必须在停止以系统时钟运行的外围硬件后执行WFI指令。

注2：为了减小A/D转换器的工作电流，必须将A/D转换器控制寄存器（CON）bit4（ADCEN）和控制寄存器（CON2）bit7（ADCST转换完成硬件清0）清“0”后，再停止A/D转换运行后执行WFI指令。

注3：能通过选项字节选择在睡眠模式或者深度睡眠模式中是继续还是停止低速内部振荡器的振荡。详细内容请参照“第29章 选项字节”。

23.2 睡眠模式

23.2.1 睡眠模式的设定

在 SCR 寄存器的 SLEEPDEEP 位为 0 时，执行 WFI 指令，就进入了睡眠模式。在睡眠模式，CPU 停止动作，但是内部寄存器的值仍被保持，周边模块也保持进入睡眠模式之前的状态。周边模块，振荡器等在睡眠模式下的状态见表 23-1.

无论设定前的 CPU 时钟是高速内部振荡器时钟或者低速内部振荡器时钟，都能设定睡眠模式。

注意：中断请求标志为“1”（产生中断请求信号）时，中断请求信号用于解除睡眠模式。因此，即使在此情况下执行 WFI 指令，也不转移到睡眠模式。

表23-1：睡眠模式中的运行状态

项目		睡眠模式
系统时钟	F_{IH}	继续运行
	F_{IL}	通过 OSMC 寄存器和 SUBCKSEL 寄存器设置运行状态，保持设定好的状态
CPU		停止运行
代码闪存		停止运行
RAM		停止运行
端口（锁存器）		保持休眠之前的设定
通用定时单元 TIMER4		能运行
LSI_定时器		能运行
时钟输出/蜂鸣器输出		能运行
看门狗定时器		休眠前设定了继续计数则休眠后可以继续计数
除法与开方运算单元		停止运算
定时器 TIMER0/1		能运行
捕捉/比较/脉宽调制模块(CCP0/1)		能运行
增强型 PWM(EPWM)		能运行
通用异步收发器(UART)		能运行
I^2C 串行接口控制器(I^2C)		能运行
串行外围接口控制器(SSP/SPI)		能运行
快速模数转换(ADC)		能运行
可编程增益放大器(PGA0/1/2)		能运行
模拟比较器(ACMP0/1)		能运行
DAC		能运行
ADCLDO		能运行
上电复位功能		能运行
电压检测功能		能运行
外部中断		能运行
CRC 运算	高速 CRC	能运行
	通用 CRC	停止运行
SFR 保护功能		停止运行

注：停止运行：在转移到睡眠模式时停止运行。

F_{IH}：高速内部振荡器时钟

F_{IL}：低速内部振荡器时钟

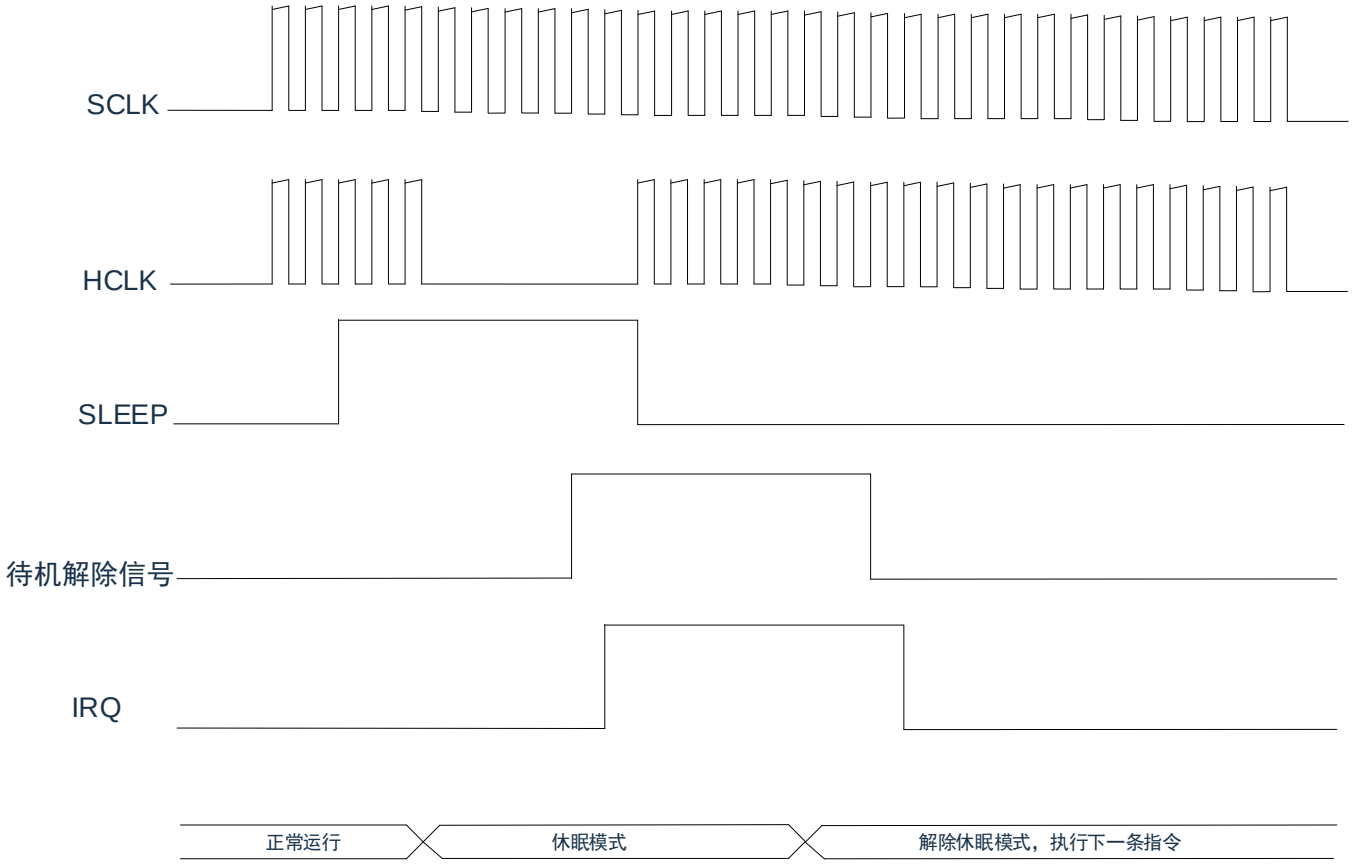
23.2.2 睡眠模式的解除

睡眠模式可以被任意中断以及外部复位，POR 复位，低电压检测复位以及 WDT 复位解除。

（1）通过中断解除

当产生一个中断请求，且处于允许接受中断的状态时，睡眠模式就被解除，CPU 开始处理中断服务程序。

图23-1：通过中断请求解除睡眠模式



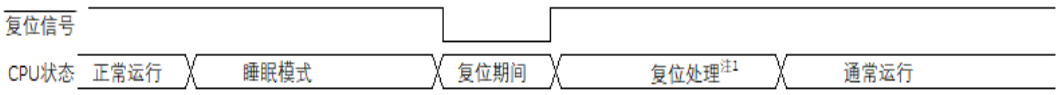
注 1：从待机解除信号产生到睡眠模式解除，开始执行中断服务程序，需要 16 个时钟。

注 2：部分待机解除信号不能自己清除，必须写寄存器清除。通常是在中断服务程序中写寄存器清除。

（2）通过复位解除

当有复位信号产生时，CPU 处于复位状态，睡眠模式被解除。和通常的复位相同，在转移到复位向量地址后执行程序。

图23-2：通过复位解除睡眠模式



注：有关复位处理，请参照“第24章 复位功能”。有关上电复位（POR）电路和电压检测（LVD）电路的复位处理，请参照“第25章 上电复位电路”。

23.3 深度睡眠模式

23.3.1 深度睡眠模式的设定

在 SCR 寄存器的 SLEEPDEEP 位为 1 时，执行 WFI 指令，就进入了深度睡眠模式。在这个模式，CPU，大多数的周边模块，以及振荡器都停止运行。但是，CPU 内部寄存器的值，RAM 数据，周边模块，I/O 的状态被保持。周边模块，振荡器在深度睡眠模式的运行状态见表 23-2。

注：当中断请求标志为“1”（产生中断请求信号）时，中断请求信号用于解除深度睡眠模式。因此，如果在此情况下执行WFI指令，就在一旦进入深度睡眠模式后立即被解除。在执行WFI指令并且经过深度睡眠模式解除时间后返回到运行模式。

表23-2：深度睡眠模式中的运行状态

项目		深度睡眠模式
系统时钟	F_{IH}	停止运行
	F_{IL}	通过 OSMC 寄存器和 SUBCKSEL 寄存器设置运行状态，保持设定好的状态
CPU		停止运行
代码闪存		停止运行
RAM		停止运行
端口（锁存器）		保持休眠之前的设定
通用定时单元 TIMER4		禁止运行
LSI_定时器		能运行
时钟输出/蜂鸣器输出		禁止运行
看门狗定时器		能运行
除法与开方运算单元		停止运算
定时器 TIMER0/1		禁止运行
捕捉/比较/脉宽调制模块(CCP0/1)		禁止运行
增强型 PWM(EPWM)		禁止运行
通用异步收发器(UART)		禁止运行
I^2C 串行接口控制器(I^2C)		禁止运行
串行外围接口控制器(SSP/SPI)		禁止运行
快速模数转换(ADC)		禁止运行
可编程增益放大器(PGA0/1/2)		禁止运行
模拟比较器(ACMP0/1)		禁止运行
DAC		禁止运行
ADCLDO		禁止运行
上电复位功能		能运行
电压检测功能		能运行
外部中断		能运行
CRC 运算	高速 CRC	停止运行
	通用 CRC	停止运行

SFR 保护功能	停止运行

注：停止运行：在转移到深度睡眠模式时停止运行。

禁止运行：在转移到深度睡眠模式前停止运行。

F_{IH} ：高速内部振荡器时钟　　F_{IL} ：低速内部振荡器时钟

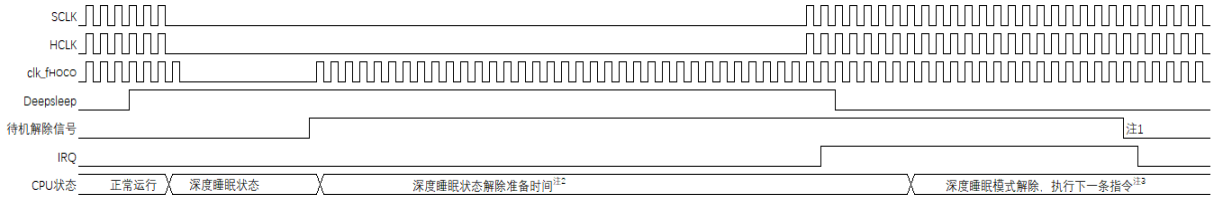
23.3.2 深度睡眠模式的解除

能通过以下2种方法解除深度睡眠模式。

(a) 通过未屏蔽的中断请求进行的解除

如果发生 LVD 检测、INTP0-3、LSI 定时器或者 WDT 中断请求，就解除深度睡眠模式。在经过振荡稳定时间后，如果处于允许接受中断的状态，就进行向量中断的处理。如果处于禁止接受中断的状态，就执行下一个地址的指令。

图23-3：通过中断请求解除深度睡眠模式



注 1：待机解除信号：有关待机解除信号的详细内容，请参照中断相关内容 。

注 2：深度睡眠状态解除准备时间：

进入深度睡眠模式前 CPU 时钟为高速内部振荡时时：至少 20us

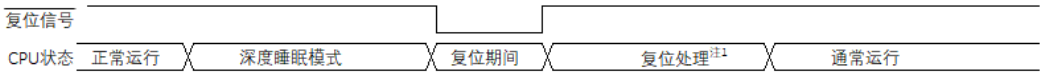
注 3：等待：从 CPU.IRQ 有效到开始执行中断服务程序，需要 14 个时钟。

注4：高速内部振荡器时钟的振荡精度稳定等待因温度条件和深度睡眠模式期间而变。

(b) 通过产生复位信号进行的解除

通过产生复位信号来解除深度睡眠模式。然后，和通常的复位相同，在转移到复位向量地址后执行程序。

图23-4：通过复位解除深度睡眠模式



注：有关复位处理，请参照“第24章 复位功能”。有关上电复位（POR）电路和电压检测（LVD）电路的复位处理，请参照“第25章 上电复位电路”。

23.4 部分掉电的深度睡眠模式

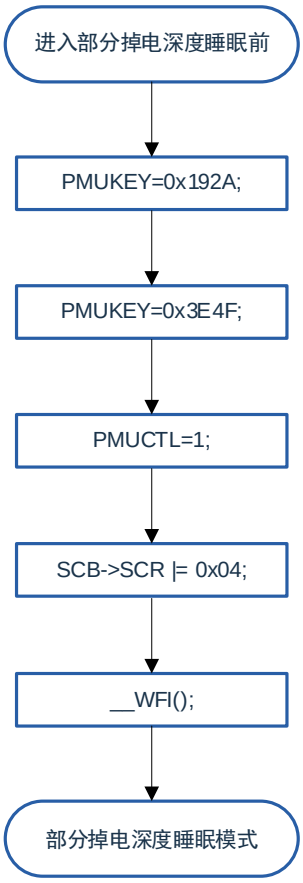
23.4.1 部分掉电的深度睡眠模式的设定

部分掉电的深度睡眠模式是在深度睡眠模式的基础上关闭部分周边的电源供电而进一步节省电源消耗的深度睡眠模式。进入部分掉电的深度睡眠模式需要配置 PMUCTL 寄存器的 PWDNEN 位，该控制位的写入受供电模式控制保护寄存器（PMUKEY）的保护，解除部分掉电的深度睡眠模式时要求重新初始化掉电的周边后才能重新正常运行，需要重新初始化的周边模块详情请参考表 23-3：部分掉电的深度睡眠模式中的运行状态。

在 SCR 寄存器的 SLEEPDEEP 位为 1 并且 PMUCTL 寄存器的 PWDNEN 位也为 1 时，执行 WFI 指令就可进入部分掉电的深度睡眠模式。在这个模式，CPU，以及发振器都停止运行，大多数的周边模块会被关闭电源供给。但是，CPU 内部寄存器的值，RAM 数据，I/O 的状态会被保持。周边模块，发振器在部分掉电的深度睡眠模式的运行状态见表 23-3。

PMUCTL 寄存器的 PWDNEN 位的控制参考 5.4.7 供电模式控制保护寄存器（PMUKEY）和 5.4.8 供电模式控制寄存器（PMUCTL）部分。

图 23-5：进入部分掉电的深度睡眠模式流程图



注：当中断请求标志为“1”（产生中断请求信号）时，中断请求信号用于解除深度睡眠模式。因此，如果在此情况下执行WFI指令，就在一旦进入深度睡眠模式后立即被解除，此种情况下不会部分掉电模式。在执行WFI指令并且经过深度睡眠模式解除时间后返回到运行模式。

表23-3：部分掉电的深度睡眠模式中的运行状态

项目		部分掉电深度睡眠模式
系统时钟	F_{IH}	停止运行
	F_{IL}	通过 OSMC 寄存器和 SUBCKSEL 寄存器设置运行状态，保持设定好的状态
CPU		停止运行
代码闪存		停止运行
RAM		停止运行
端口（锁存器）		保持休眠之前的设定
通用定时单元 TIMER4		禁止运行
LSI_定时器		能运行
时钟输出/蜂鸣器输出		停止运行
看门狗定时器		能运行
除法与开方运算单元		停止运算
定时器 TIMER0/1		禁止运行
捕捉/比较/脉宽调制模块(CCP0/1)		禁止运行
增强型 PWM(EPWM)		禁止运行
通用异步收发器(UART)		禁止运行
I^2C 串行接口控制器(I^2C)		禁止运行
串行外围接口控制器(SSP/SPI)		禁止运行
快速模数转换(ADC)		停止运行
可编程增益放大器(PGA0/1/2)		禁止运行
模拟比较器(ACMP0/1)		禁止运行
DAC		禁止运行
ADCLDO		禁止运行
上电复位功能		能运行
电压检测功能		能运行
外部中断		能运行
CRC 运算	高速 CRC	停止运行
	通用 CRC	停止运行
SFR 保护功能		停止运行

注：停止运行：在转移到部分掉电的深度睡眠模式时自动停止运行。
禁止运行：在转移到部分掉电的深度睡眠模式前停止运行。
转移到部分掉电的深度睡眠模式后停止模块的电源供应，解除模式后需要重新初始化模块。
F_{IH} ：高速内部振荡器时钟 F_{IL} ：低速内部振荡器时钟

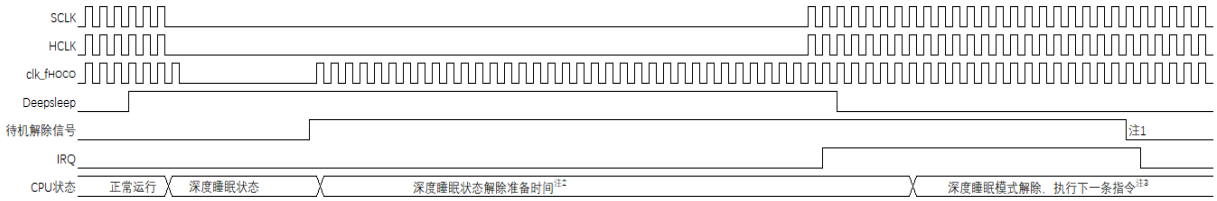
23.4.2 部分掉电的深度睡眠模式的解除

能通过以下 2 种方法解除部分掉电的深度睡眠模式。

(a) 通过中断请求进行部分掉电的深度睡眠的解除

如果 INTP0-3、LSITIMER 定时器中断、LVI 中断及 WDT 中断请求，可解除部分掉电的深度睡眠模式。在经过振荡稳定时间后，如果处于允许接受中断的状态，就进行向量中断的处理。如果处于禁止接受中断的状态，就执行下一个地址的指令。

图23-6：通过中断请求解除深度睡眠模式



注 1：待机解除信号：INTP0-3、LSITIMER 定时器中断、LVI 中断及 WDT 中断请求信号。

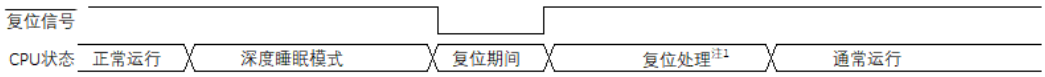
注 2：部分掉电的深度睡眠状态解除准备：

需要重新初始化周边功能，才能保证程序继续正常运行。

(b) 通过产生复位信号进行的解除

通过产生复位信号来解除部分掉电的深度睡眠模式。然后，和通常的复位相同，在转移到复位向量地址后执行程序。

图23-7：通过复位解除部分掉电的深度睡眠模式



注：有关复位处理，请参照“第24章 复位功能”。有关上电复位（POR）电路和电压检测（LVD）电路的复位处理，请参照“第25章 上电复位电路”。

第24章 复位功能

以下6种方法产生复位信号。

(1)通过 RESETB 引脚输入外部复位。

(2)通过看门狗定时器的程序失控检测产生内部复位。

(3)通过上电复位(POR)电路的电源电压和检测电压的比较产生内部复位。

(4)通过电压检测电路(LVD)的电源电压和检测电压的比较产生内部复位。

(5)因系统复位请求寄存器位(AIRCR.SYSRESETREQ)被置为 1 而产生内部复位。

(6)因存取非法存储器而产生内部复位。

内部复位和外部复位相同，在产生复位信号后，从用户自定义的程序起始地址开始执行程序。

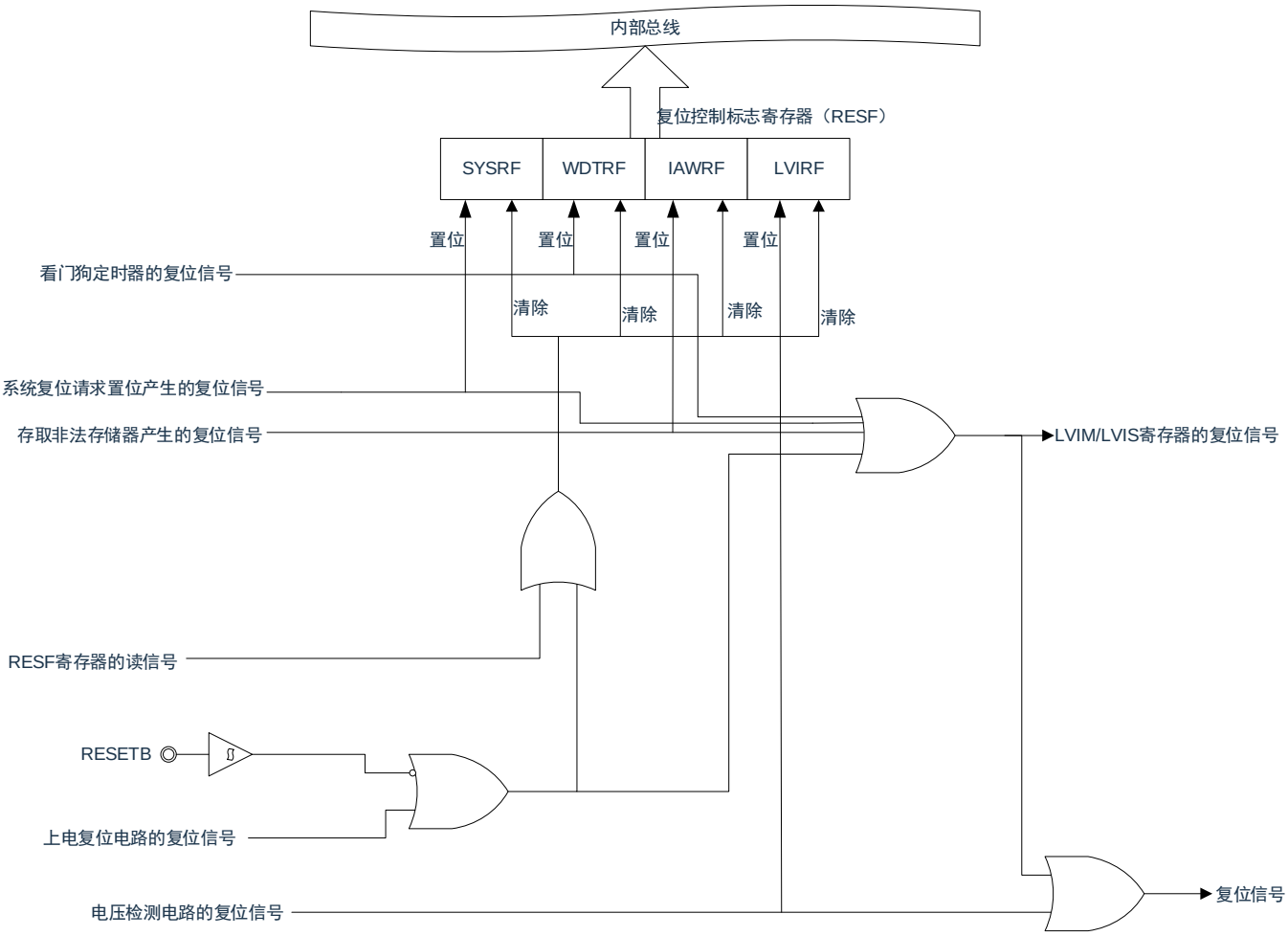
当给 RESETB 引脚输入低电平，或者看门狗定时器检测到程序失控，或者检测到 POR 电路和 LVD 电路的电压，或者系统复位请求位被置位，或者存取非法存储器时，产生复位并且各硬件变为如表 24-1 所示的状态。

注 1：在进行外部复位时，必须至少给 RESETB 引脚输入 10us 的低电平。如果在电源电压上升时进行外部复位，就必须在给 RESETB 引脚输入低电平后接通电源，而且在用户手册的 AC 特性所示的工作电压范围内至少保持 10us 的低电平，然后输入高电平。

注 2：如果发生复位，就对各 SFR 进行初始化，因此端口引脚变为以下状态：

- 在外部复位或者 POR 复位的期间除 P02 接内部上拉，P06，P07 接内部下拉外，其他 IO 均处于 高阻态 。

图24-1：复位功能的框图



注1：LVD电路的内部复位不会对LVD电路进行复位。

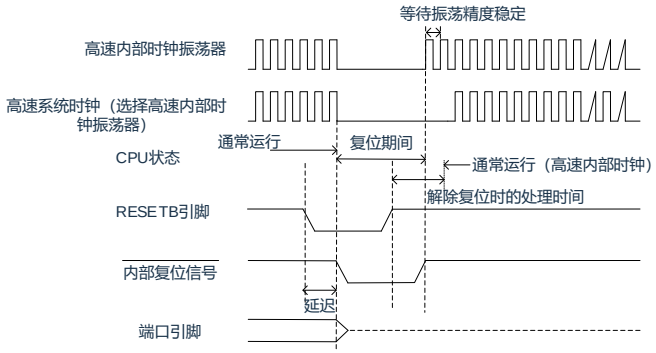
注2：LVIM：电压检测寄存器

注3：LVIS：电压检测电平寄存器

1. 复位时序

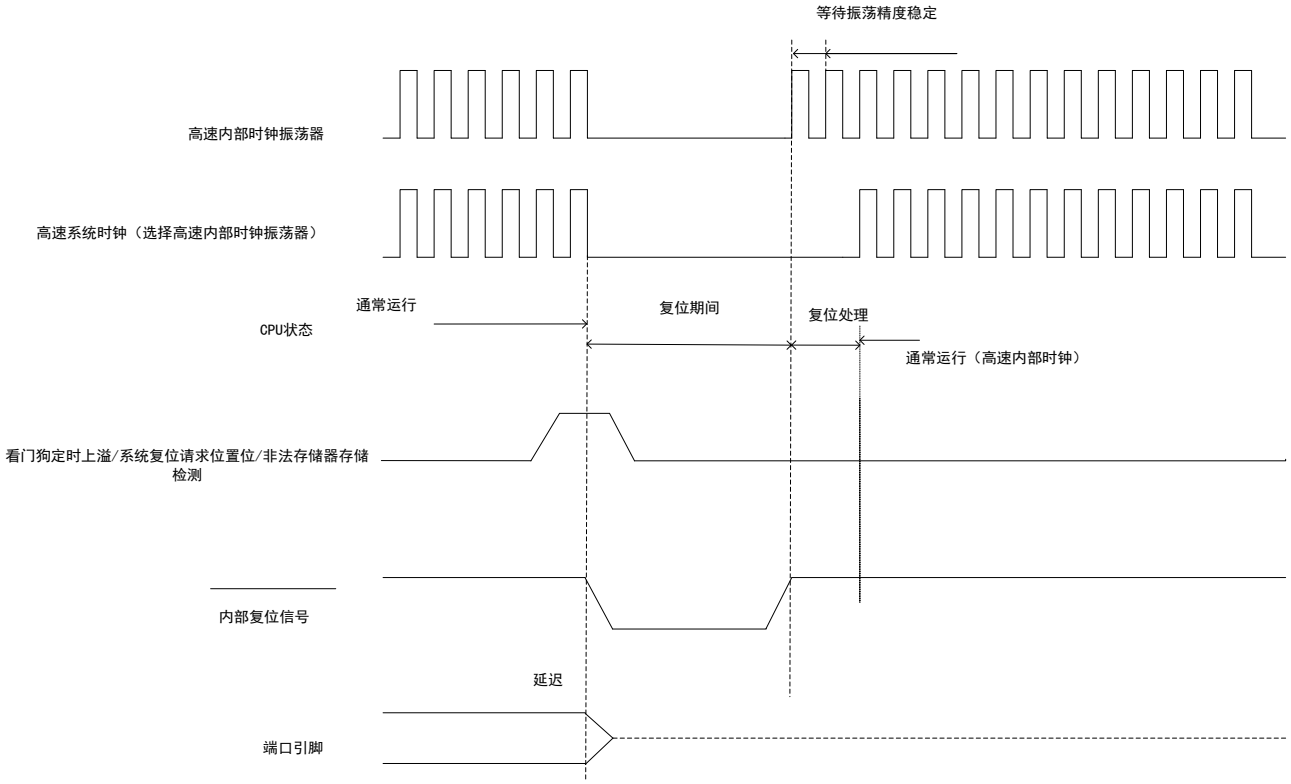
当给 RESETB 引脚输入低电平时，产生复位。然后，如果给 RESETB 引输入高电平就解除复位状态，并且在复位处理结束后以高速内部振荡器时钟开始执行程序。

图 24-2：RESETB 输入的复位时序



对于因看门狗定时器的上溢、系统复位请求位的置位或者非法存储器存取的检测而产生的复位，自动解除复位状态，在复位处理结束后以高速内部振荡器时钟开始执行程序。

图 24-3：因看门狗定时器的上溢、系统复位请求位的置位或者非法存储器存取的检测而产生的复位时序。



注1：端口引脚P02,P06,P07变为以下状态：

- 在外部复位期间P02为外部输入信号确定，P06，P07为低电平。
- POR复位的期间为P02为高电平，P06，P07为低电平。

注2：看门狗定时器也不例外，在发生内部复位时进行复位。

对于由 POR 电路和 LVD 电路的电压检测产生的复位，如果在复位后满足 VDD≥VPOR 或者 VDD≥VLVD，就解除复位状态，并且在复位处理后以高速内部振荡器时钟开始执行程序。详细内容请参照“第 24 章 上电复位

电路”和“第 26 章 电压检测电路”。

注3：VPOR：POR电源电压上升检测电压

VLVD： LVD检测电压

表 24-1：复位期间的运行状态

项目		复位期间					
		外部复位 RESINB	看门狗复位	上电复位	LVD 电压检测复位	写复位寄存器复位	存取非法存储器复位
系统时钟	F_{IH}	停止运行	停止运行	停止运行	停止运行	停止运行	停止运行
	F_{IL}	停止运行	停止运行	停止运行	停止运行	停止运行	停止运行
CPU		停止运行	停止运行	停止运行	停止运行	停止运行	停止运行
代码闪存		停止运行	停止运行	停止运行	停止运行	停止运行	停止运行
RAM		停止运行	停止运行	停止运行	停止运行	停止运行	停止运行
端口（锁存器）		P02 上电后为高，P06,P07 为低，其他脚复位前中后均为高阻态。	P02 上电后为高，P06,P07 为低，其他脚复位前中后均为高阻态。	P02 为低电平，其他引脚为高阻态	未配置的情况下 P02 上电后为高，P06,P07 为低，其他脚复位前中后均为高阻态。	P02 上电后为高，P06,P07 为低，其他脚复位前中后均为高阻态。	P02 上电后为高，P06,P07 为低，其他脚复位前中后均为高阻态。
通用定时单元 TIMER4		停止运行	停止运行	停止运行	停止运行	停止运行	停止运行
LSI_定时器		停止运行	停止运行	停止运行	停止运行	停止运行	停止运行
时钟输出/蜂鸣器输出		停止运行	停止运行	停止运行	停止运行	停止运行	停止运行
看门狗定时器		计数停止	停止运行	停止运行	停止运行	停止运行	停止运行
除法与开方运算单元		停止运行	停止运行	停止运行	停止运行	停止运行	停止运行
定时器 TIMER0/1		停止运行	停止运行	停止运行	停止运行	停止运行	停止运行
捕捉/比较/脉宽调制模块(CCP0/1)		停止运行	停止运行	停止运行	停止运行	停止运行	停止运行
增强型 PWM(EPWM)		停止运行	停止运行	停止运行	停止运行	停止运行	停止运行
通用异步收发器 (UART)		停止运行	停止运行	停止运行	停止运行	停止运行	停止运行
I^2C 串行接口控制器 (I^2C)		停止运行	停止运行	停止运行	停止运行	停止运行	停止运行
串行外围接口控制器 (SSP/SPI)		停止运行	停止运行	停止运行	停止运行	停止运行	停止运行
快速模数转换(ADC)		停止运行	停止运行	停止运行	停止运行	停止运行	停止运行
可编程增益放大器 (PGA0/1/2)		停止运行	停止运行	停止运行	停止运行	停止运行	停止运行
模拟比较器 (ACMP0/1)		停止运行	停止运行	停止运行	停止运行	停止运行	停止运行
DAC		停止运行	停止运行	停止运行	停止运行	停止运行	停止运行
ADCLDO		停止运行	停止运行	停止运行	停止运行	停止运行	停止运行
上电复位功能		能进行检测运行	能进行检测运行	能进行检测运行	能进行检测运行	能进行检测运行	能进行检测运行
电压检测功能		不能运行			能运行	不能运行	不能运行
外部中断		停止运行	停止运行	停止运行	停止运行	停止运行	停止运行
CRC 运算功能	高速 CRC	停止运行	停止运行	停止运行	停止运行	停止运行	停止运行
	通用 CRC	停止运行	停止运行	停止运行	停止运行	停止运行	停止运行
SFR 保护功能		停止运行	停止运行	停止运行	停止运行	停止运行	停止运行
复位完成后上电状态(选项字节配置)		重新配置	重新配置	进行选项字节配置	重新配置	重新配置	重新配置

注 ：P02上电后为高（外部复位时为外部输入的状态），P06,P07为低，其他脚复位前中后均为高阻态。

F_{IH}：高速内部时钟

F_{IL}：低速内部时钟

24.1 确认复位源的寄存器

24.1.1 寄存器映射

(复位控制基地址 = 0x4002_0440)RO：只读；WO：只写；R/W：读写。

寄存器	偏移量	读/写	描述	复位值
RESF	0x000	RO	复位控制标志寄存器	-

24.1.2 复位控制标志寄存器（RESF）

CMS32M65xx 微控制器存在多种内部复位发生源。复位控制标志寄存器（RESF）保存发生复位请求的复位源。能通过 8 位存储器操作指令读 RESF 寄存器。

通过 RESETB 的输入、上电复位（POR）电路的复位和 RESF 寄存器的读取，清除 SYSRF、WDTRF、IAWRF、LVIRF 标志。要判断复位源时，必须将 RESF 寄存器的值保存到任意的 RAM，然后通过其 RAM 值进行判断。

位	符号	描述	复位值[注1]
7	SYSRF	系统复位请求位被置位而产生的内部复位请求 0: 没有产生内部复位请求或者清除了 RESF 寄存器。 1: 产生内部复位请求。	-
6:5	-	保留	-
4	WDTRF	看门狗定时器（WDT）产生的内部复位请求 0: 没有产生内部复位请求或者清除了 RESF寄存器。 1: 产生内部复位请求。	-
3	-	保留	-
2	RPERF	RAM 奇偶校验错误产生的内部复位请求 0: 没有产生内部复位请求或者清除了 RESF寄存器。 1: 产生内部复位请求。	-
1	IAWRF	存取非法存储器产生的内部复位请求 0: 没有产生内部复位请求或者清除了 RESF寄存器。 1: 产生内部复位请求。	-
0	LVIRF	电压检测电路（LVD）产生的内部复位请求 0: 没有产生内部复位请求或者清除了 RESF寄存器。 1: 产生内部复位请求。	-

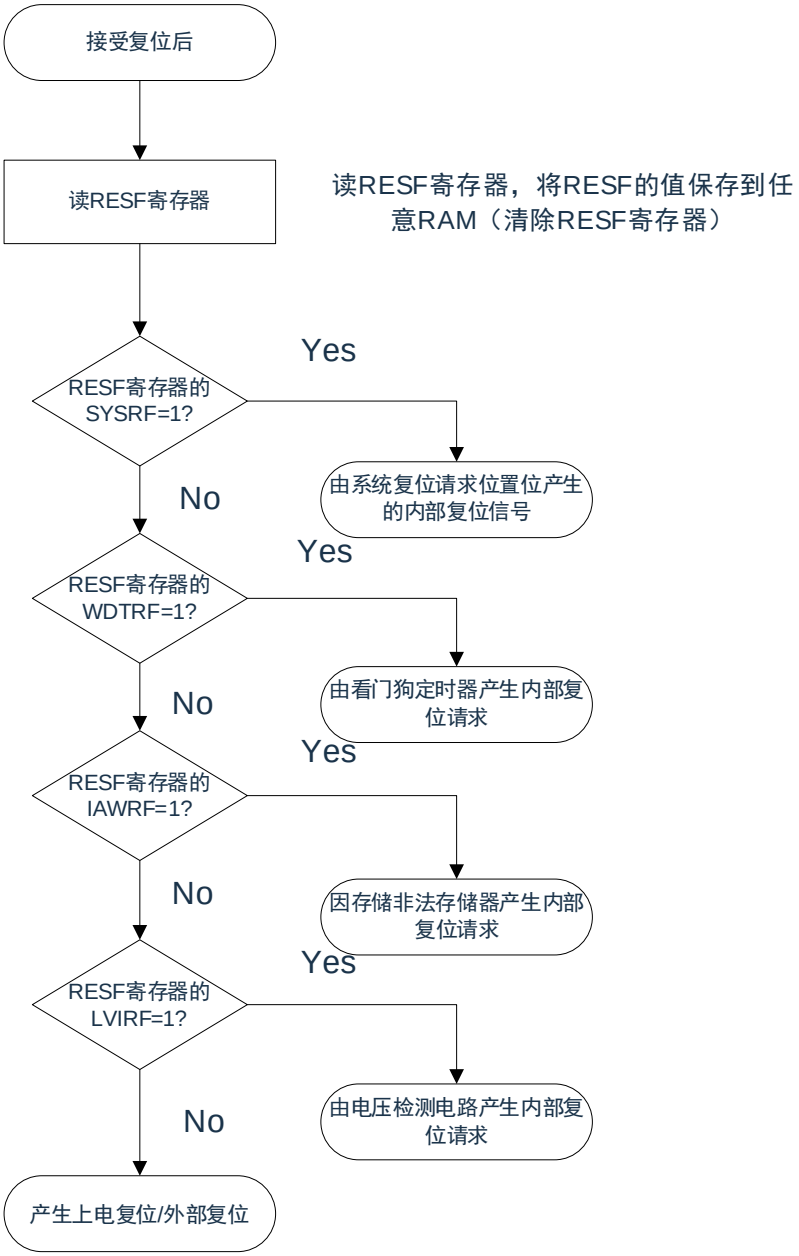
注：因复位源而不同。请参照表24-2。

表 24-2：发生复位请求时的 RESF 寄存器状态表

<table>
<tr><th>复位源
标志</th><th>RESETB输入</th><th>POR产生的复位</th><th>系统复位请求位置位产生的复位</th><th>WDT产生的复位</th><th>存取非法存储器产生的复位</th><th>LVD产生的复位</th></tr>
<tr><td>SYSRF</td><td rowspan="4">清“0”</td><td rowspan="4">清“0”</td><td>置“1”</td><td>保持</td><td rowspan="2">保持</td><td rowspan="3">保持</td></tr>
<tr><td>WDTRF</td><td rowspan="3">保持</td><td>置“1”</td></tr>
<tr><td>IAWRF</td><td rowspan="2"></td><td>置“1”</td></tr>
<tr><td>LVIRF</td><td>保持</td><td>置“1”</td></tr>
</table>

复位源的确认步骤如图24-4所示。

图24-4：复位源的确认步骤



注：上述流程是确认步骤的例子。

第25章 上电复位电路

25.1 上电复位电路的功能

上电复位电路（POR）有以下功能。

① 在接通电源时产生内部复位信号。

如果电源电压（VDD）超过检测电压（VPOR），就解除复位。但是，必须在电源电压达到数据手册的 AC 特性所示的工作电压范围前，通过电压检测电路或者外部复位保持复位状态。

② 将电源电压（VDD）和检测电压（VPDR）进行比较。当 VDD＜VPDR 时，产生内部复位信号。但是，当电源电压下降时，必须在电源电压低于数据手册的 AC 特性所示的工作电压范围前，通过深度睡眠模式的转移、电压检测电路或者外部复位，置为复位状态。在重新开始运行时，必须确认电源电压是否恢复到工作电压范围。

注 1：当上电复位电路产生内部复位信号时，将复位控制标志寄存器（RESF）清“00H”。

注 2：CMS32M65xx 内置多个产生内部复位信号的硬件。当由看门狗定时器（WDT）、电压检测（LVD）电路、系统复位请求位置位或者非法存储器的存取而产生内部复位信号时，用于表示复位源的标志分配在 RESF 寄存器；当由 WDT、LVD、系统复位请求位的置位或者非法存储器的存取而产生内部复位信号时，不将 RESF 寄存器清“00H”，而将标志置“1”。有关 RESF 寄存器的详细内容，请参照“第 24 章 复位功能”。

注 3：VPOR：POR 电源电压上升检测电压

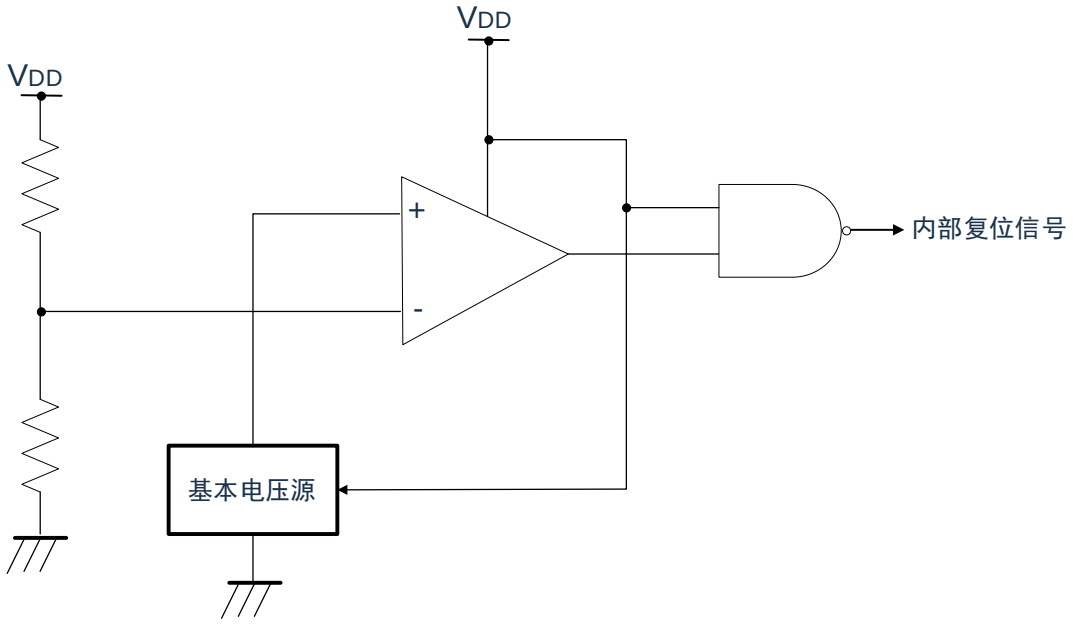
VPDR：POR 电源电压下降检测电压

详细内容请参照数据手册的 POR 电路特性。

25.2 上电复位电路的结构

上电复位电路的框图如图25-1所示。

图25-1：上电复位电路的框图

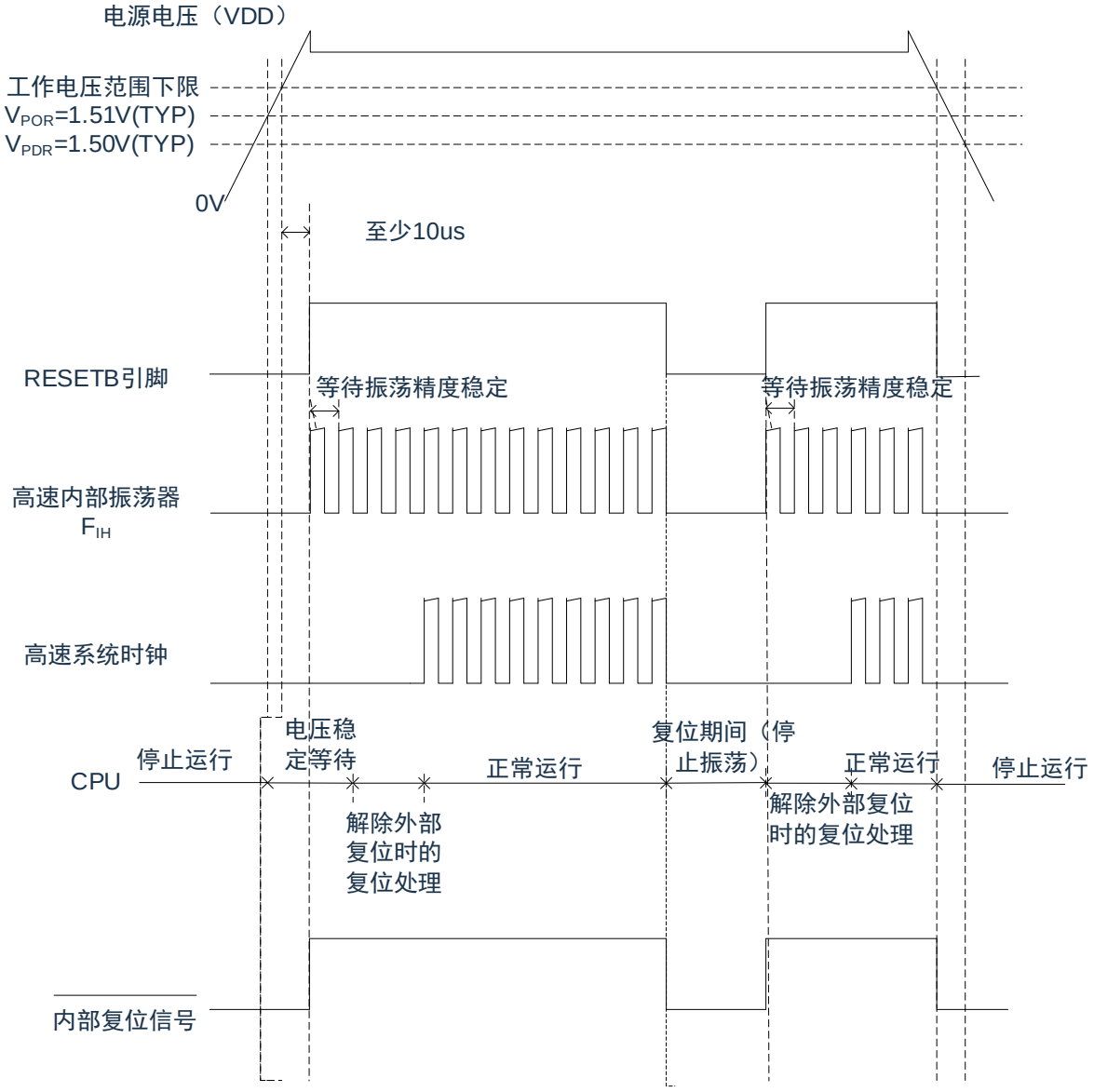


25.3 上电复位电路的运行

上电复位电路和电压检测电路的内部复位信号的产生时序如下所示。

图 25-2：上电复位电路和电压检测电路的内部复位信号的产生时序(1/3)

(1) 使用 RESETB 引脚的外部复位输入的情况



注1：内部复位处理时间包含高速内部振荡器时钟的振荡精度稳定等待时间。

注2：当电源电压上升时，必须在电源电压达到数据手册的AC特性所示的工作电压范围前，通过外部复位保持复位状态；当电源电压下降时，必须在电源电压低于工作电压范围前，通过深度睡眠模式的转移、电压检测电路或者外部复位，置为复位状态。在重新开始运行时，必须确认电源电压是否恢复到工作电压范围。

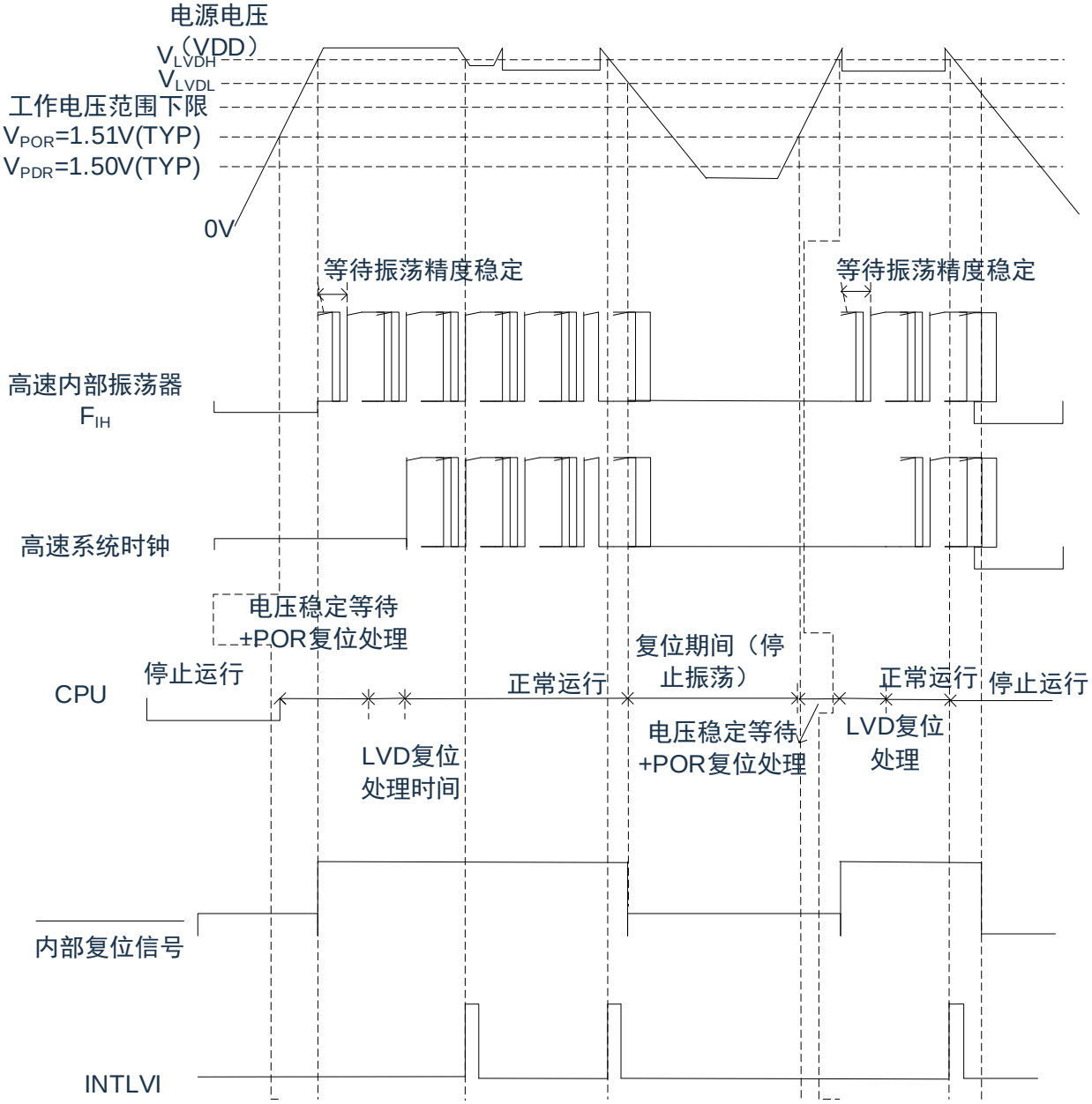
注3：VPOR：POR电源电压上升检测电压

VPDR：POR电源电压下降检测电压

注4：在LVD为OFF时，必须使用RESETB引脚的外部复位。详细内容请参照“第26章 电压检测电路”

图 25-2：上电复位电路和电压检测电路的内部复位信号的产生时序(2/3)

(2) LVD为中断&复位模式的情况（选项字节000C1H的LVIMDS1、LVIMDS0=1、0）



注1：内部复位处理时间包含高速内部振荡器时钟的振荡精度稳定等待时间。

注2：在产生中断请求信号（INTLVI）后，自动将电压检测电平寄存器（LVIS）的LVILV位和LVIMD位置“1”。因此，必须考虑可能出现电源电压在不低于低电压检测电压（VLVDL）的状态下恢复到高电压检测电压（VLVDH）或者更高的情况，在产生INTLVI后按照“图26-5：工作电压的确认/复位的设定步骤”和“图26-6：中断&复位模式的初始设定步骤”进行设定。

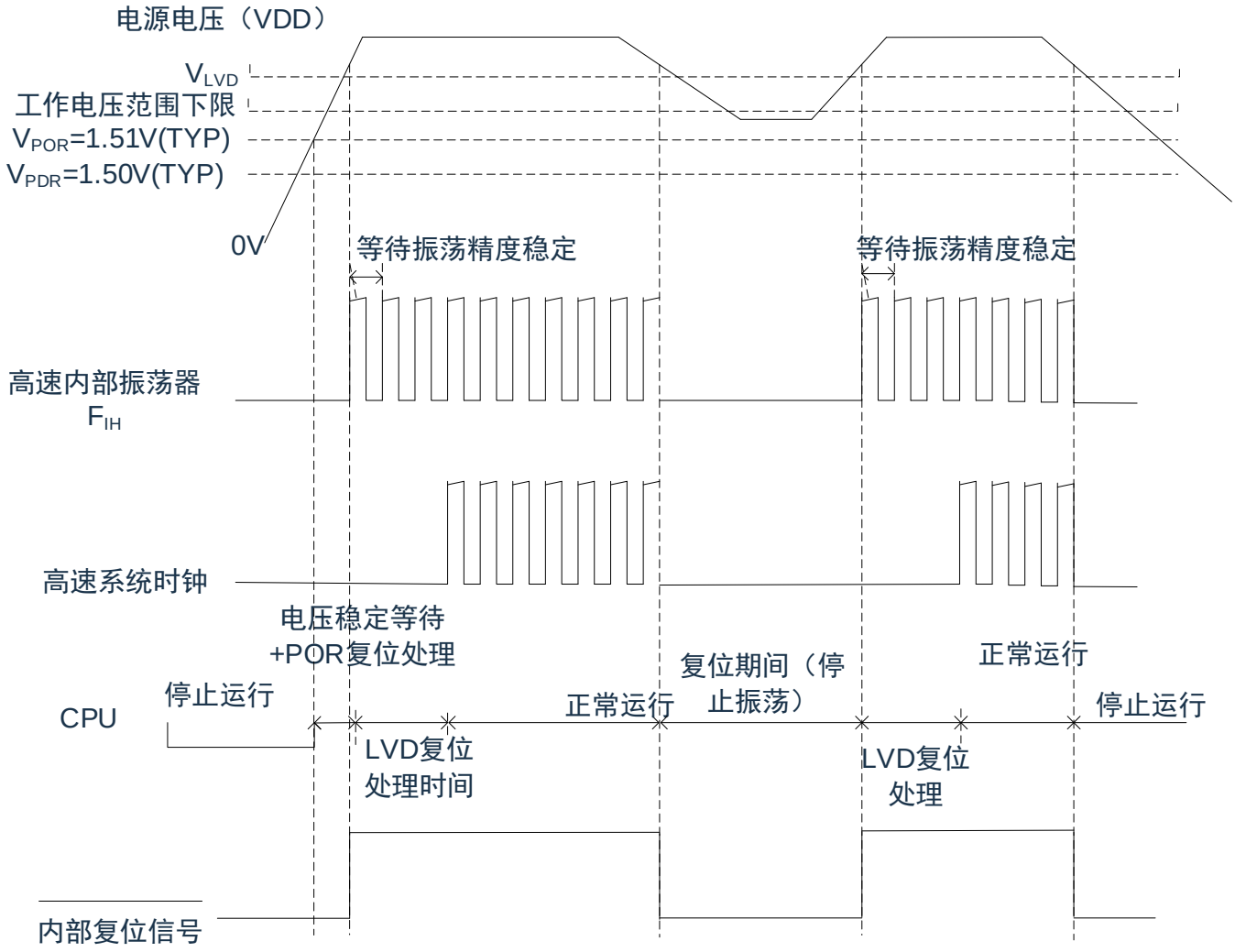
注3：到开始通常运行为止的时间除了达到VPOR（1.51V(TYP.)）后的“电压稳定等待+POR复位处理”以外，在达到LVD检测电平（VLVDH）后还需要以下的“LVD复位处理”。

注4：VLVDH、VLVDL　：LVD检测电压

VPOR　：POR电源电压上升检测电压

VPDR　：POR电源电压下降检测电

图25-2：位电路和电压检测电路的内部复位信号的产生时序(3/3)



(3)LVD复位模式的情况（选项字节000C1H的LVIMDS1、LVIMDS0=1、1）

注1：.内部复位处理时间包含高速内部振荡器时钟的振荡精度稳定等待时间。

注2：到开始通常运行为止的时间除了达到VPOR（1.51V(TYP.)）后的“电压稳定等待+POR复位处理”以外，在达到LVD检测电平（VLVD）后还需要“LVD复位处理”。

注3：在电源电压下降时，如果只在发生电压检测电路（LVD）的内部复位后恢复电源电压，就在达到LVD检测电平（VLVD）后需要“LVD复位处理”。

注4：VLVDH、VLVDL：LVD检测电压

VPOR：POR电源上升检测电压

VPDR：POR电源下降检测电压

注5：当选择LVD中断模式（选项字节000C1H的LVIMD1、LVIMD0=0、1）时，从接通电源到开始正常运行的时间和“图25-2(3/3)LVD复位模式的情况”的LVD复位处理的时间相同。

第26章 电压检测电路

26.1 电压检测电路的功能

电压检测电路通过选项字节（000C1H）设定运行模式和检测电压（VLVDH、VLVDL、VLVD）。电压检测（LVD）电路有以下功能。

1) 将电源电压（VDD）和检测电压（VLVDH、VLVDL、VLVD）进行比较，产生内部复位或者内部中断信号。
2) 电源电压的检测电压（VLVDH、VLVDL）能通过选项字节选择 12 种检测电平（参照“第 29 章 选项字节”）。
3) 也能在深度睡眠模式中运行
4) 当电源电压上升时，必须在电源电压达到数据手册的 AC 特性所示的工作电压范围前，通过电压检测电路或者外部复位保持复位状态；当电源电压下降时，必须在电源电压低于工作电压范围前，通过深度睡眠模式的转移、电压检测电路或者外部复位，置为复位状态。工作电压范围取决于用户选项字节（000C2H）的设定。

(a) 中断&复位模式（选项字节的 LVIMDS1、LVIMDS0=1、0）

通过选项字节 000C1H 选择 2 个检测电压（VLVDH、VLVDL），高电压检测电平（VLVDH）用于解除复位或者产生中断，低电压检测电平（VLVDL）用于产生复位。

(b) 复位模式（选项字节的 LVIMDS1、LVIMDS0=1、1）

将选项字节 000C1H 选择的 1 个检测电压（VLVD）用于产生或者解除复位。

(c) 中断模式（选项字节的 LVIMDS1、LVIMDS0=0、1）

将选项字节 000C1H 选择的 1 个检测电压（VLVD）用于产生中断或者解除复位。在各模式中，产生以下的中断信号和内部复位信号。

模式	中断&复位模式	复位模式	中断模式
设置	（LVIMDS1、LVIMDS0=1、0）	（LVIMDS1、LVIMDS0=1、1）	（LVIMDS1、LVIMDS0=0、1）
运行过程	在工作电压下降时，当检测到VDD＜VLVDH时，产生中断请求信号；当检测到VDD＜VLVDL时，产生内部复位；当检测到VDD≥VLVDH时，解除内部复位。	当检测到VDD≥VLVD时，解除内部复位；当检测到VDD＜VLVD时，产生内部复位。	在发生复位后，LVD的内部复位状态继续保持到VDD≥VLVD为止。当检测到VDD≥VLVD时，解除LVD的内部复位。在解除LVD的内部复位后，如果检测到VDD ＜ VLVD 或者VDD≥VLVD时，就产生中断请求信号（INTLVI）。

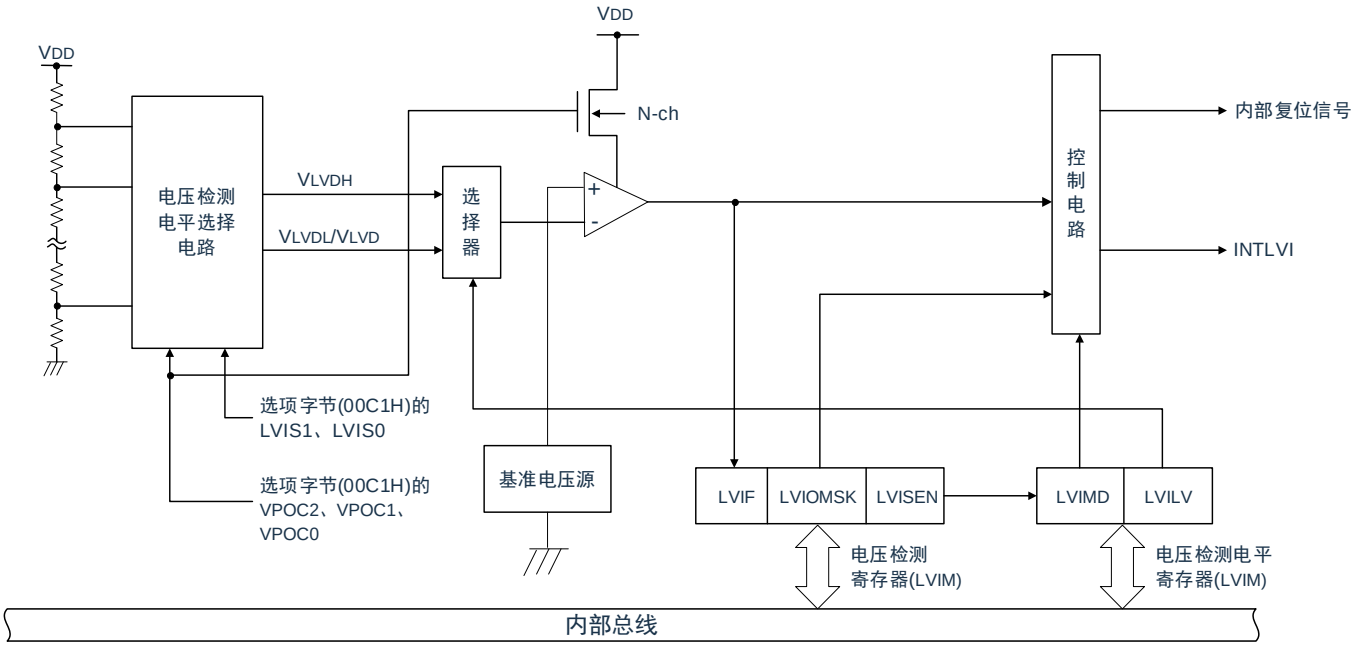
在电压检测电路运行时，能通过读电压检测标志（LVIF：电压检测寄存器（LVIM）的 bit0）来确认电源电压是大于等于检测电压还是小于检测电压。

如果发生复位，就将复位控制标志寄存器（RESF）的 bit0（LVIRF）置“1”。有关 RESF 寄存器的详细内容，请参照“第 24 章 复位功能”。

26.2 电压检测电路的结构

电压检测电路的框图如图26-1所示。

图26-1：电压检测电路的框图



26.3 寄存器映射

(电压检测基地址 = 0x4002_0441)
RO：只读，WO：只写，R/W：读写

寄存器	偏移量	读/写	描述	复位值
LVIM	0x000	R/W	电压检测寄存器	0x0
LVIS	0x001	R/W	电压检测电平寄存器	0x0

26.4 控制电压检测电路的寄存器

通过以下寄存器控制电压检测电路。

- 电压检测寄存器（LVIM）
- 电压检测电平寄存器（LVIS）

26.4.1 电压检测寄存器（LVIM）

此寄存器设定允许或者禁止改写电压检测电平寄存器（LVIS），并且确认LVD输出的屏蔽状态。通过8位存储器操作指令设定LVIM寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

位	符号	描述	复位值
7	LVISEN[注1]	电压检测电平寄存器（LVIS）的允许/禁止改写的设定 0: 禁止 1: 允许	0
6:2	-	保留	-
1	LVIOMSK	LVD 输出的屏蔽状态标志 0: 屏蔽无效 1: 屏蔽有效[注2]	0
0	LVIF	电压检测标志 0: 电源电压（VDD）≥检测电压（VLVD）或者 LVD 为 OFF。 1: 电源电压（VDD）＜检测电压（VLVD）	0

注 1.只有在选择中断&复位模式（选项字节的LVIMDS1位和LVIMDS0位分别为“1”和“0”）时才能设定，在其他模式中不能更改初始值。

2.只有在选择中断&复位模式（选项字节的LVIMDS1位和LVIMDS0位分别为“1”和“0”）时，LVIOMSK位才在以下期间自动变为“1”，屏蔽LVD产生的复位或者中断。

- LVISEN=1的期间
- 从发生LVD中断开始到LVD检测电压稳定为止的等待时间
- 从更改LVILV位的值到LVD检测电压稳定为止的等待时间

26.4.2 电压检测电平寄存器（LVIS）

这是设定电压检测电平的寄存器。

通过 8 位存储器操作指令设定 LVIS 寄存器。在产生复位信号后，此寄存器的值变为“00H/01H/81H”[注1]。

位	符号	描述	复位值
7	LVIMD[注1]	电压检测的运行模式 0: 中断模式 1: 复位模式	0
6:1	-	保留	-
0	LVILV[注2]	LVD 检测电平 0: 高电压检测电平（VLVDH） 1: 低电压检测电平（VLVDL 或者 VLVD）	0

注 1.复位值因复位源和选项字节的设定而变。在发生LVD复位时，不将此寄存器清“00H”。

在发生LVD以外的复位时，此寄存器的值如下：

- 选项字节的LVIMDS1、LVIMDS0=1、0时：00H
- 选项字节的LVIMDS1、LVIMDS0=1、1时：81H
- 选项字节的LVIMDS1、LVIMDS0=0、1时：01H

2.只有在选择中断&复位模式（选项字节的LVIMDS1位和LVIMDS0位分别为“1”和“0”）时才能写“0”。在其他情况下不能设定。在中断&复位模式中，通过产生复位或者中断自动进行值的替换。

3.要改写LVIS寄存器时，必须按照图26-5和图26-6的步骤进行。

4.通过选项字节000C1H选择LVD的运行模式和各模式的检测电压（VLVDH、VLVDL、VLVD）。用户选项字节（000C1H/010C1H）的详细内容，请参照“第29章 选项字节”。

26.5 电压检测电路的运行

26.5.1 用作复位模式时的设定

通过选项字节 000C1H 设定运行模式（复位模式（LVIMDS1、LVIMDS0=1、1））和检测电压（VLVD）。如果设定复位模式，就在以下初始设定的状态下开始运行。

- 将电压检测寄存器（LVIM）的 bit7（LVISEN）置“0”（禁止改写电压检测电平寄存器（LVIS））。
- 将电压检测电平寄存器（LVIS）的初始值置“81H”。bit7（LVIMD）为“1”（复位模式）。

bit0（LVILV）为“1”（电压检测电平：VLVD）。

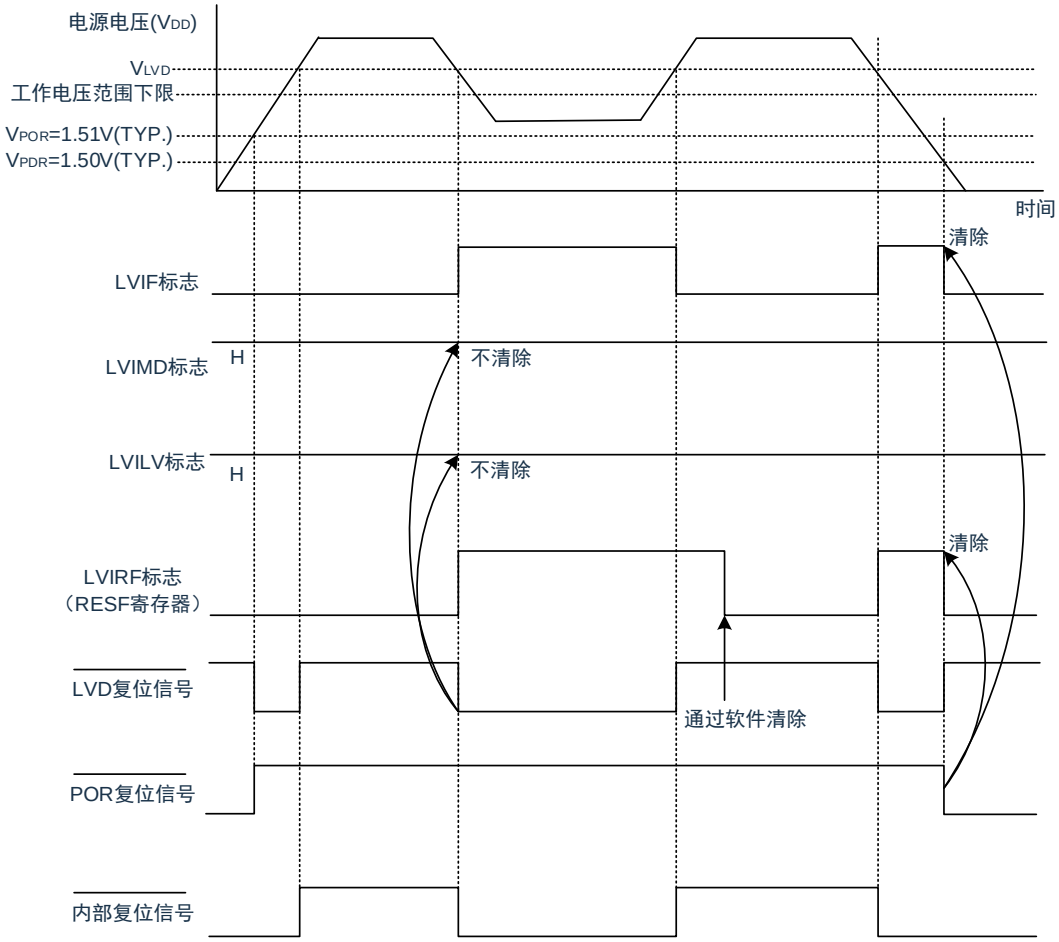
- LVD 复位模式的运行

当接通电源时，复位模式（选项字节的 LVIMDS1、LVIMDS0=1、1）在电源电压（VDD）超过电压检测电平（VLVD）前保持 LVD 的内部复位状态。如果电源电压（VDD）超过电压检测电平（VLVD），就解除内部复位。

当工作电压下降时，如果电源电压（VDD）低于电压检测电平（VLVD），就产生 LVD 的内部复位。

LVD 复位模式的内部复位信号的产生时序如图 26-2 所示。

图26-2：内部复位信号的产生时序（选项字节的LVIMDS1、LVIMDS0=1、1）



注 ：VPOR：POR电源电压上升检测电压

VPDR：POR电源电压下降检测电压

26.6 用作中断模式时的设定

通过选项字节 000C1H 设定运行模式（中断模式（LVIMDS1、LVIMDS0=0、1））和检测电压（VLVD）。如果设定中断模式，就在以下初始设定的状态下开始运行。

- 将电压检测寄存器（LVIM）的 bit7（LVISEN）置“0”（禁止改写电压检测电平寄存器（LVIS））。
- 将电压检测电平寄存器（LVIS）的初始值置“01H”。bit7（LVIMD）为“0”（中断模式）。
  bit0（LVILV）为“1”（电压检测电平：VLVD）。

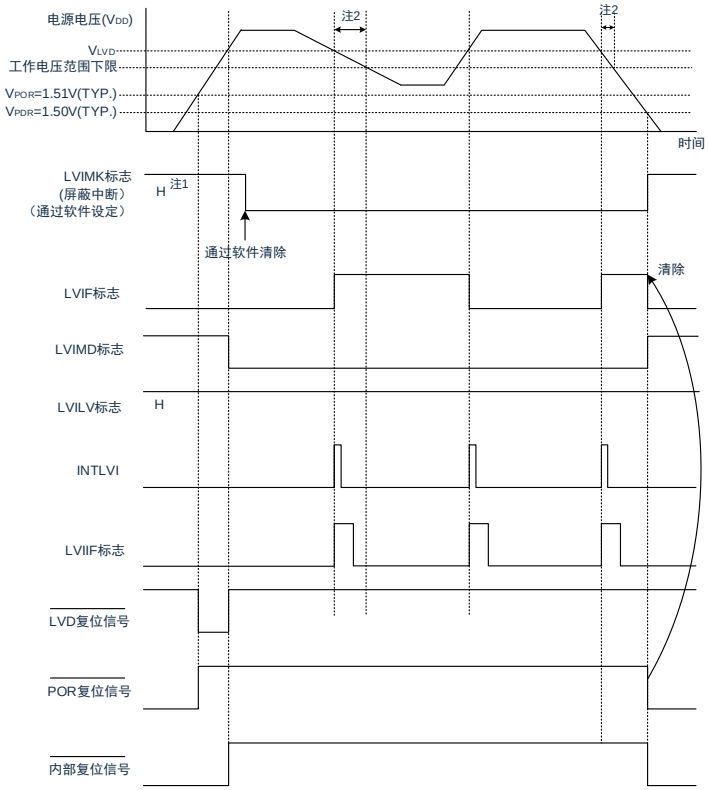
● LVD 中断模式的运行

在产生复位后，中断模式（选项字节的 LVIMDS1、LVIMDS0=0、1）在电源电压（VDD）超过电压检测电平（VLVD）前保持 LVD 的内部复位状态。如果电源电压（VDD）超过电压检测电平（VLVD），就解除 LVD 的内部复位。

在解除 LVD 的内部复位后，如果电源电压（VDD）超过电压检测电平（VLVD），就产生 LVD 的中断请求信号（INTLVI）。当工作电压下降时，必须在工作电压低于数据手册的 AC 特性所示的工作电压范围前，通过深度睡眠模式的转移或者外部复位，置为复位状态。在重新开始运行时，必须确认电源电压是否恢复到工作电压范围。

LVD 中断模式的中断请求信号的产生时序如图 26-3 所示

图26-3：中断信号的产生时序（选项字节的LVIMDS1、LVIMDS0=0、1）



注 1.在产生复位信号后，LVIMK标志变为“1”。

2.当工作电压下降时，必须在工作电压低于数据手册的AC特性所示的工作电压范围前，通过深度睡眠模式的转移或者外部复位，置为复位状态。在重新开始运行时，必须确认电源电压是否恢复到工作电压范围。

3.VPOR：POR电源电压上升检测电压

VPDR：POR电源电压下降检测电压

26.7 用作中断&复位模式时的设定

通过选项字节 000C1H 设定运行模式（中断&复位模式（LVIMDS1、LVIMDS0=1、0））和检测电压（VLVDH、VLVDL）。

如果设定中断&复位模式，就在以下初始设定的状态下开始运行。

- 将电压检测寄存器（LVIM）的 bit7（LVISEN）置“0”（禁止改写电压检测电平寄存器（LVIS））。
- 将电压检测电平寄存器（LVIS）的初始值置“00H”。bit7（LVIMD）为“0”（中断模式）。

bit0（LVILV）为“0”（高电压检测电平：VLVDH）。

- LVD 中断&复位模式的运行

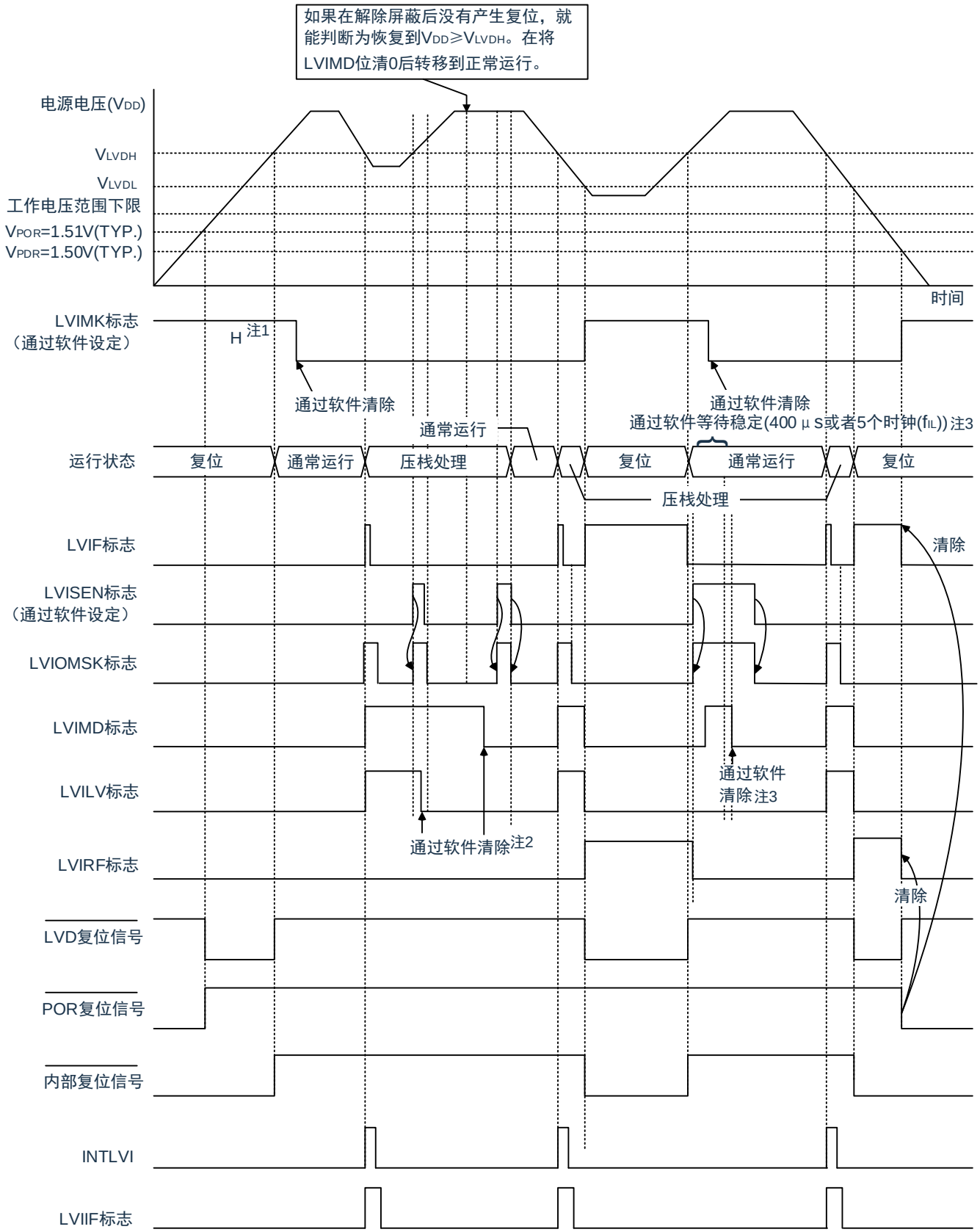
当接通电源时，中断&复位模式（选项字节的 LVIMDS1、LVIMDS0=1、0）在电源电压（VDD）超过高电压检测电平（VLVDH）前保持 LVD 的内部复位状态。如果电源电压（VDD）超过高电压检测电平（VLVDH），就解除内部复位。

当工作电压下降时，如果电源电压（VDD）低于高电压检测电平（VLVDH），就产生 LVD 的中断请求信号（INTLVI）并且能进行任意的压栈处理。此后，如果电源电压（VDD）低于低电压检测电平（VLVDL），就产生 LVD 的内部复位。但是，在发生 INTLVI 后，即使电源电压（VDD）在不低于低电压检测电压（VLVDL）的状态下恢复到高电压检测电压（VLVDH）或者更高，也不产生中断请求信号。

当使用 LVD 中断&复位模式时，必须按照“图 26-5：工作电压的确认/复位的设定步骤”和“图 26-6：中断&复位模式的初始设定步骤”所示的流程图的步骤进行设定。

LVD 中断&复位模式的内部复位信号和中断信号的产生时序如图 26-4 示。

图 26-4：复位&中断信号的产生时序（选项字节的 LVIMDS1、LVIMDS0=1、0）(1/2)



注 1.在产生复位信号后，LVIMK标志变为“1”。

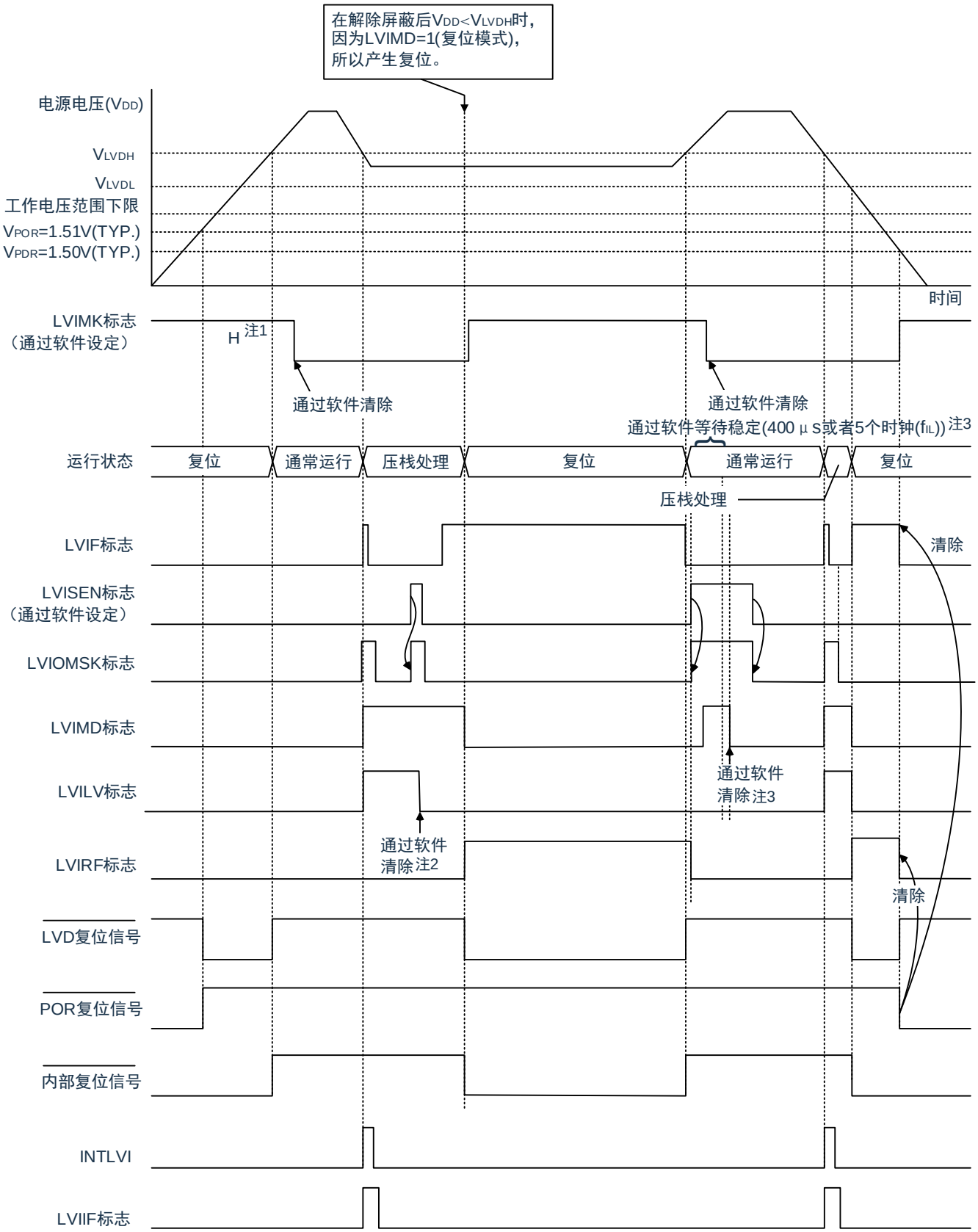
2.当使用中断&复位模式时，必须在发生中断后按照“图26-5：工作电压的确认/复位的设定步骤”进行设定。

3.当使用中断&复位模式时，必须在解除复位后按照“图26-6：中断&复位模式的初始设定步骤”进行设定。

4.VPOR：POR电源电压上升检测电压

VPDR：POR电源电压下降检测电压

图 26-4：中断&复位信号的产生时序（选项字节的 LVIMDS1、LVIMDS0=1、0）(2/2)



注 1.在产生复位信号后，LVIMK标志变为“1”。

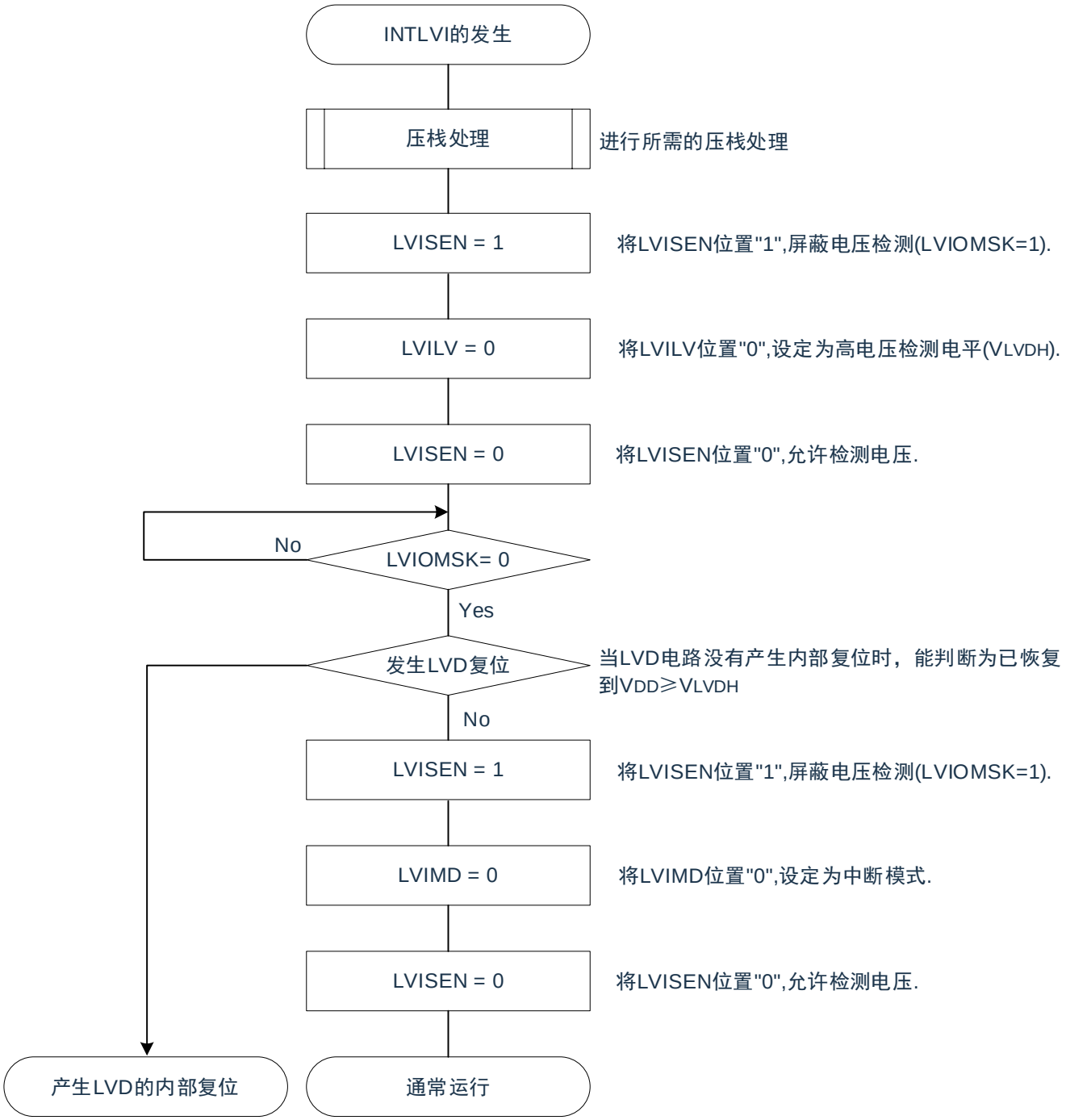
2.当使用中断&复位模式时，必须在发生中断后按照“图26-5：工作电压的确认/复位的设定步骤”进行设定。

3.当使用中断&复位模式时，必须在解除复位后按照“图26-6：中断&复位模式的初始设定步骤”进行设定。

4.VPOR：POR电源电压上升检测电压

VPDR：POR电源电压下降检测电压

图26-5：工作电压的确认/复位的设定步骤

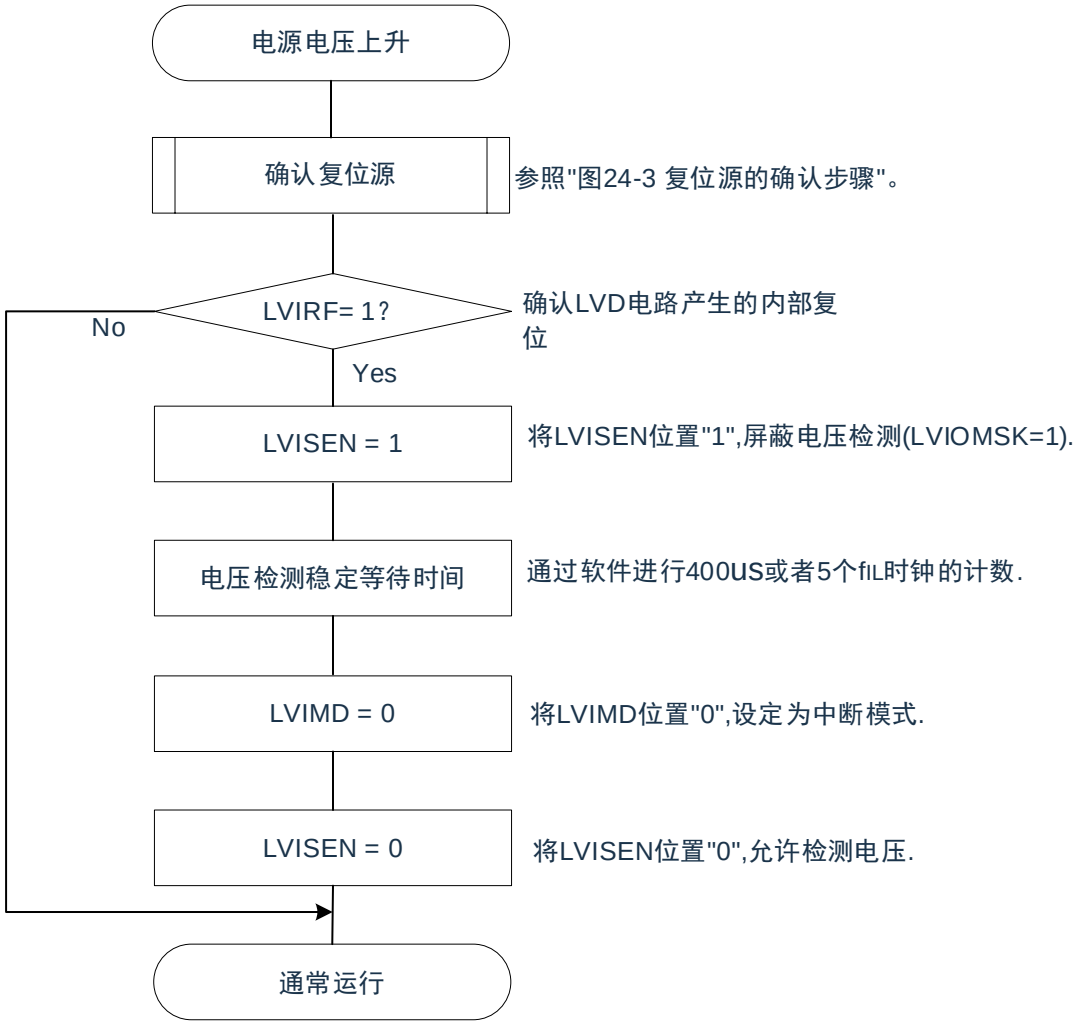


如果设定中断&复位模式（LVIMDS1、LVIMDS0=1、0），就在解除 LVD 复位（LVIRF=1）后需要 400us 或者 5 个 F_{IL} 时钟的电压检测稳定等待时间。必须在等待电压检测稳定后将 LVIMD 位清“0”进行初始化。在电压检

测稳定等待时间的计数过程中以及在改写 LVIMD 位时，必须将 LVISEN 位置“1”，屏蔽 LVD 产生的复位或者中断的产生。

中断&复位模式的初始设定步骤如图 26-6 所示。

图26-6：中断&复位模式的初始设定步骤



注：F_{IL}：低速内部振荡器时钟频率

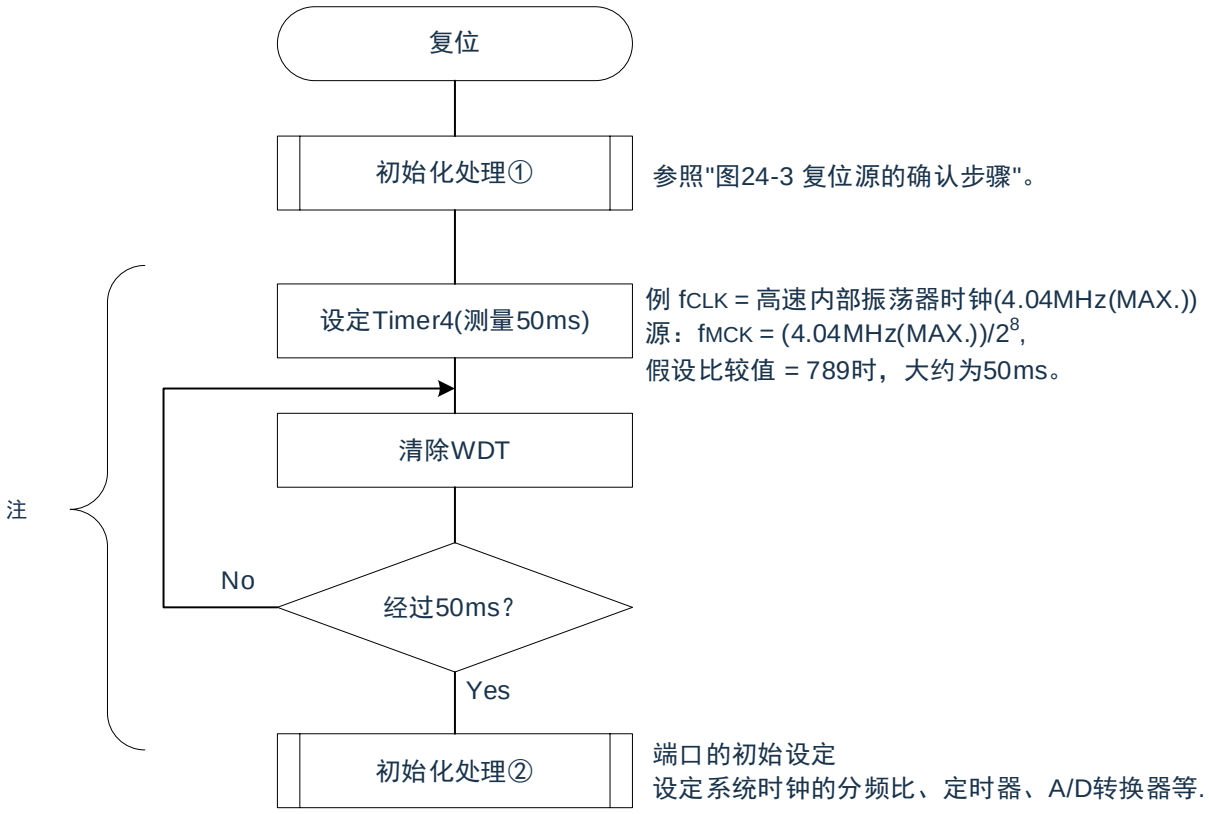
26.8 电压检测电路的注意事项

(1) 有关接通电源时的电压波动

对于电源电压（VDD）在 LVD 检测电压附近发生一定时间波动的系统，有可能重复进入复位状态和复位解除状态。能通过以下的处理，任意设定解除复位到单片机开始运行的时间。

在解除复位后，必须通过使用定时器的软件计数器，在等待各系统不同的电源电压波动时间后进行端口等的初始设定。

图 26-7：LVD 检测电压附近的电源电压波动不超过 50ms 时的软件处理例子

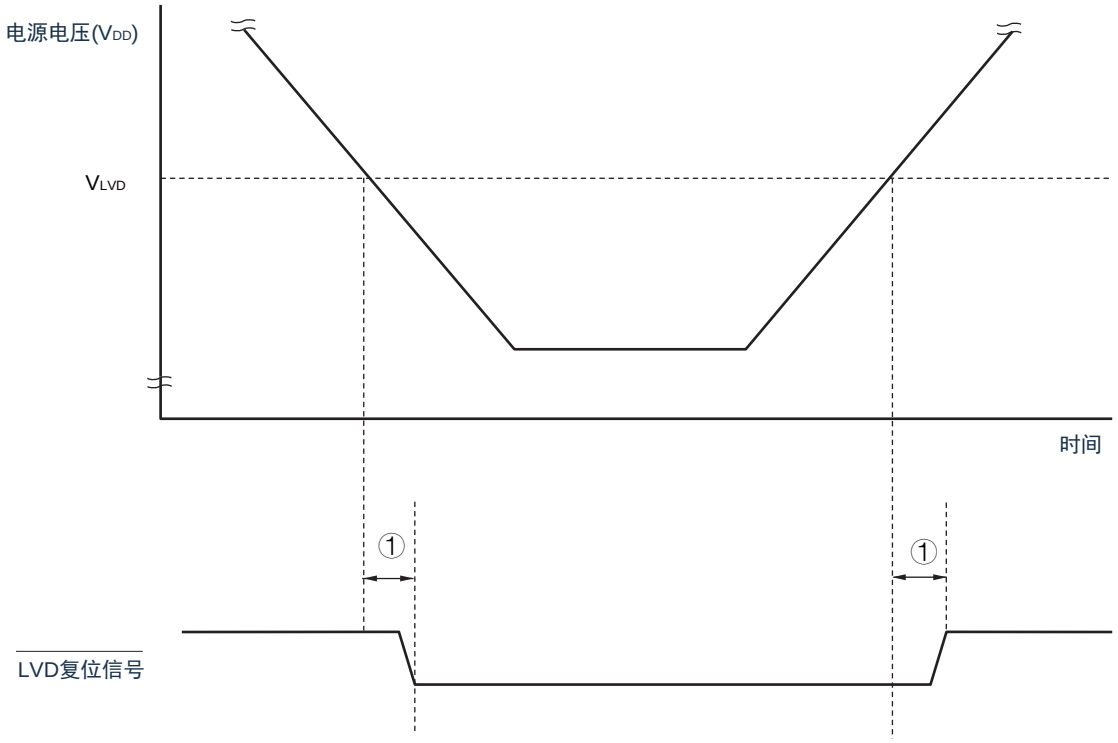


注：如果在此期间再次发生复位，就不转移到初始化处理②。

(2) 从产生 LVD 复位源到产生或者解除 LVD 复位的延迟

从满足电源电压（VDD）＜LVD 检测电压（VLVD）到产生 LVD 复位为止会发生延迟。同样，从 LVD 检测电压（VLVD）≤电源电压（VDD）到解除 LVD 复位为止也会发生延迟（参照图 26-8）。

图 26-8：从产生 LVD 复位源到产生或者解除 LVD 复位的延迟



① 检测延迟（300us(MAX.)）

(3) 有关将 LVD 置为 OFF 时接通电源的情况

当将 LVD 置为 OFF 时，必须使用 RESETB 引脚的外部复位。

在进行外部复位时，必须至少给 RESETB 引脚输入 10us 的低电平。如果在电源电压上升时进行外部复位，就必须在给 RESETB 引脚输入低电平后接通电源，而且在数据手册的 AC 特性所示的工作电压范围内至少保持 10us 的低电平，然后输入高电平。

(4) 有关将 LVD 置为 OFF 并且设定为 LVD 中断模式时工作电压下降的情况

在将 LVD 置为 OFF 并且设定为 LVD 中断模式的情况下，如果工作电压下降，就必须在工作电压低于数据手册的 AC 特性所示的工作电压范围前，通过深度睡眠模式的转移或者外部复位，置为复位状态。在重新开始运行时，必须确认电源电压是否恢复到工作电压范围。

第27章 安全功能

27.1 安全功能的概要

为了对应 IEC60730 和 IEC61508 安全标准，CMS32M65xx 内置以下安全功能。

此安全功能的目的是通过单片机的自诊断，在检测到故障时安全地停止工作。

(1)闪存 CRC 运算功能（高速 CRC、通用 CRC）

通过 CRC 运算检测闪存的数据错误。能根据不同的用途和使用条件，分别使用以下 2 个 CRC。

- “高速 CRC”…在初始化程序中，能停止 CPU 的运行并且高速检查整个代码闪存区。
- “通用 CRC”…在 CPU 运行中，不限于代码闪存区而能用于多用途的检查。

(2)SFR 保护功能

防止因 CPU 失控而改写 SFR。

(3)频率检测功能

能使用通用定时器单元进行 CPU/外围硬件时钟频率的自检。

(4)A/D 测试功能

能通过 A/D 转换器的正（+）基准电压、负（–）基准电压、模拟输入通道（ANI）、温度传感器输出和内部基准电压输出的 A/D 转换进行 A/D 转换器的自检。

(5)输入/输出端口的数字输出信号电平检测功能

在输入/输出端口为输出模式时，能读引脚的输出电平。

27.2 寄存器映射

安全功能的各功能使用以下寄存器。

寄存器名	安全功能的各功能
•闪存CRC控制寄存器（CRC0CTL） •闪存CRC运算结果寄存器L（PGCRCL）	闪存CRC运算功能（高速CRC）
•CRC输入寄存器（CRCIN） •CRC数据寄存器（CRCD）	CRC运算功能（通用CRC）
•特殊SFR保护控制寄存器（SFRGD）	SFR保护功能
•定时器输入选择寄存器0（TIOS0）	频率检测功能
•A/D测试寄存器（CON2）	A/D测试功能
•端口模式选择寄存器（PMS）	输入/输出引脚的数字输出信号电平检测功能

有关各寄存器的内容，在“27.3 安全功能的运行”中进行说明。

(闪存CRC基地址 = 0x4002_1810)

RO：只读，WO：只写，R/W：读写

寄存器	偏移量	读/写	描述	复位值
CRC0CTL	0x000	R/W	闪存CRC控制寄存器	0x0
PGCRCL	0x002	R/W	闪存CRC运算结果寄存器L	0x0

(通用CRC基地址 = 0x4004_32FA)

RO：只读，WO：只写，R/W：读写

寄存器	偏移量	读/写	描述	复位值
CRCD	0x000	R/W	闪存CRC运算结果寄存器	0x0
CRCIN	0x0B2	R/W	闪存CRC控制寄存器	0x0

(SFR基地址 = 0x4004_0478)

RO：只读，WO：只写，R/W：读写

寄存器	偏移量	读/写	描述	复位值
SFRGD	0x000	R/W	SFR保护控制寄存器	0x0

(端口控制基地址 = 0x4004_087B)

RO：只读，WO：只写，R/W：读写

寄存器	偏移量	读/写	描述	复位值
PMS	0x000	R/W	端口模式选择寄存器	0x0

(UID基地址 = 0x0050_0894)

RO：只读，WO：只写，R/W：读写

寄存器	偏移量	读/写	描述	复位值
UID0	0x000	RO	产品唯一的身份标识的第[31:0]位	-
UID1	0X004	RO	产品唯一的身份标识的第[63:32]位	-
UID2	0X008	RO	产品唯一的身份标识的第[95:64]位	-
UID3	0X00C	RO	产品唯一的身份标识的第[127:96]位	-

27.3 安全功能的运行

27.3.1 闪存CRC运算功能（高速CRC）

IEC60730 标准要求确认闪存中的数据，并且建议 CRC 为确认手段。此高速 CRC 能在初始设定（初始化）程序中检查整个代码闪存区。

高速 CRC 停止 CPU 的运行并且通过 1 个时钟从闪存读 32 位数据进行运算。因此，其特点是完成检查的时间较短（例如，64KB 闪存：1024us@64MHz）。

CRC 生成多项式对应 CRC-16-CCITT 的“$X^{16}+X^{12}+X^{5}+1$”。

以 bit31→bit0 的 MSB 优先进行运算。

注1：此为系统时钟，系统时钟与闪存CRC运算的时钟的关系为4 : 1，系统时钟为64MHZ，则闪存CRC的运算时钟为16MHZ。

注2：因为通用CRC为LSB优先，所以运算结果不同。:

闪存 CRC 控制寄存器（CRC0CTL）

这是设定高速 CRC 运算器的运行控制和运算范围的寄存器。通过 8 位存储器操作指令设定 CRC0CTL 寄存器。在产生复位信号后，此寄存器的值变为“00H”。

位	符号	描述	复位值
7	CRC0EN	高速 CRC 运算器的运行控制 0: 停止运行 1: 通过执行 WFE 指令开始运算	0
6	CRCCHK60	60K演算范围选择 0: 演算范围由[2:0]位控制 1: 00000H ~EFFBH(60K-4byte)	0
5:3	-	保留	-
2:0	FEA	高速CRC的演算范围 000: 00000H ~1FFBH(8K-4byte) 001: 00000H ~3FFBH(16K-4byte) 010: 00000H ~5FFBH(24K-4byte) 011: 00000H ~7FFBH(32K-4byte) 100: 00000H ~9FFBH(40K-4byte) 101: 00000H ~BFFBH(48K-4byte) 110: 00000H ~DFFBH(56K-4byte) 111: 00000H ~FFFBH(64K-4byte)	0x0

注1：bit3~5必须设置为0.

注2：必须事先将用于比较的CRC运算结果期待值存入闪存的最后4字节，因此运算范围为减去4字节的范围。

27.3.1.1 闪存CRC运算结果寄存器L（PGCRCL）

这是保存高速 CRC 低 16 位运算结果的寄存器。

通过 16 位存储器操作指令设定 PGCRCL 寄存器。

在产生复位信号后，此寄存器的值变为“0000H”。

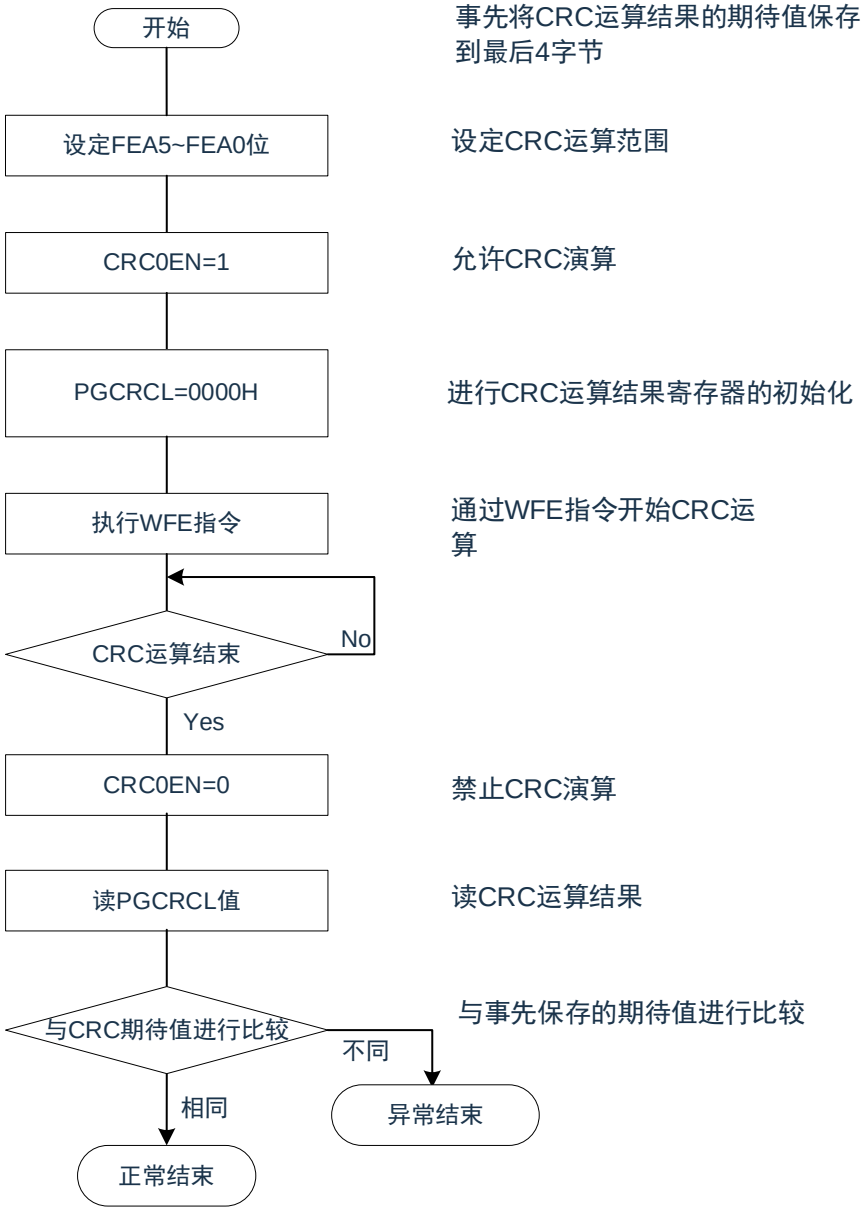
位	符号	描述	复位值
15:0	PGCRCL	保存高速 CRC 低 16 位运算结果 0000H ~FFFFH	0x0

注：只有在CRC0EN（CRC0CTL寄存器的bit7）位为“1”时才能写PGCRCL寄存器。

闪存CRC运算功能（高速CRC）的流程图如图27-1所示。

＜操作流程＞

图27-1：闪存CRC运算功能（高速CRC）的流程图



注1：只以代码闪存为CRC运算的对象。

注2：必须将CRC运算的期待值保存在代码闪存中的运算范围后的区域。

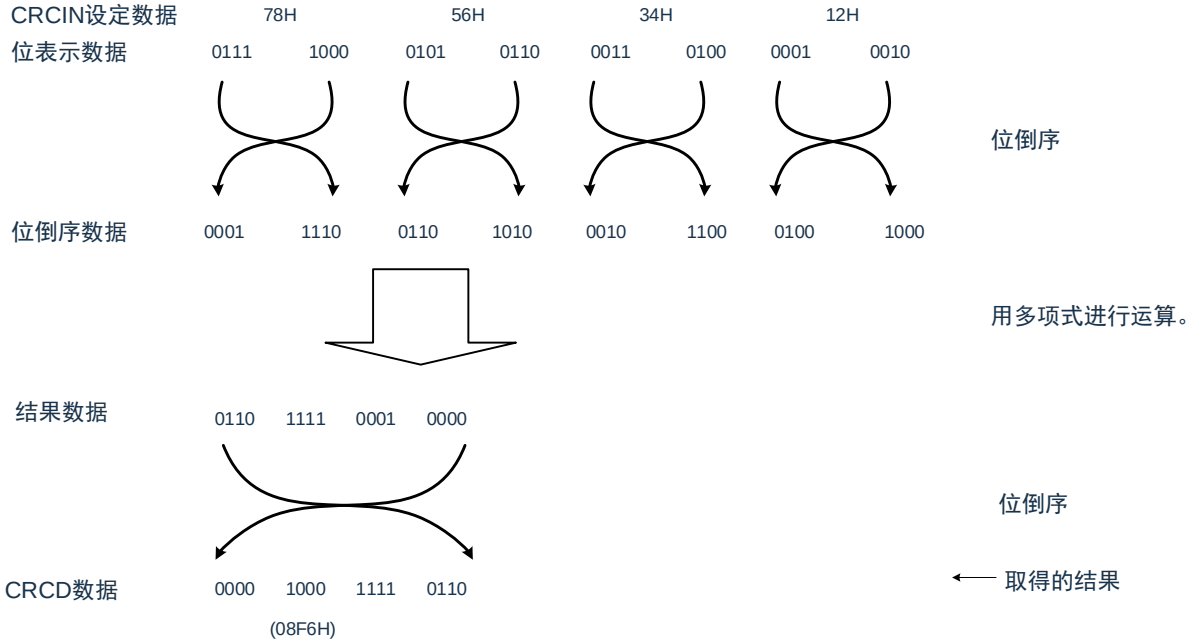
27.3.1.2 CRC运算功能（通用CRC）

为了必须保证运行过程中的安全，IEC61508 标准要求即使在 CPU 运行中也需要确认数据。

此通用 CRC 能在 CPU 运行中作为外围功能进行 CRC 运算。通用 CRC 不限于代码闪存区而能用于多用途的检查。通过软件（用户程序）指定要确认的数据。

在主系统时钟运行模式中，都能使用 CRC 运算功能。

CRC 生成多项式使用 CRC-16-CCITT 的“$X^{16}+X^{12}+X^{5}+1$”。因为考虑到是以 LSB 优先进行的通信，所以在将输入数据的位序颠倒后进行计算。例如，从 LSB 发送数据“12345678H”的情况，按照“78H”、“56H”、“34H”、“12H”的顺序给 CRCIN 寄存器写值，从 CRCD 寄存器得到“08F6H”的值。这是针对颠倒了数据“12345678H”的位序后的以下位序进行 CRC 运算的结果。



注：在执行程序的过程中，因为调式程序将软件断点的设定行改写为断点指令，所以如果在CRC运算的对象区设定软件断点，CRC的运算结果就不同。

27.3.1.3 CRC输入寄存器（CRCIN）

这是设定通用CRC的CRC计算数据的8位寄存器。能设定的范围为“00H～FFH”。

通过8位存储器操作指令设定CRCIN寄存器。在产生复位信号后，此寄存器的值变为“00H”。

位	符号	描述	复位值
7:0	CRCIN	通用 CRC 数据输入 00H ~FFH	0x0

27.3.1.4 CRC数据寄存器（CRCD）

这是保存通用 CRC 运算结果的寄存器。能设定的范围为“0000H～FFFFH”。

在写 CRCIN 寄存器后经过 1 个 CPU/外围硬件时钟（F_{CLK}），将 CRC 运算结果保存到 CRCD 寄存器。通过 16 位存储器操作指令设定 CRCD 寄存器。

在产生复位信号后，此寄存器的值变为“0000H”。

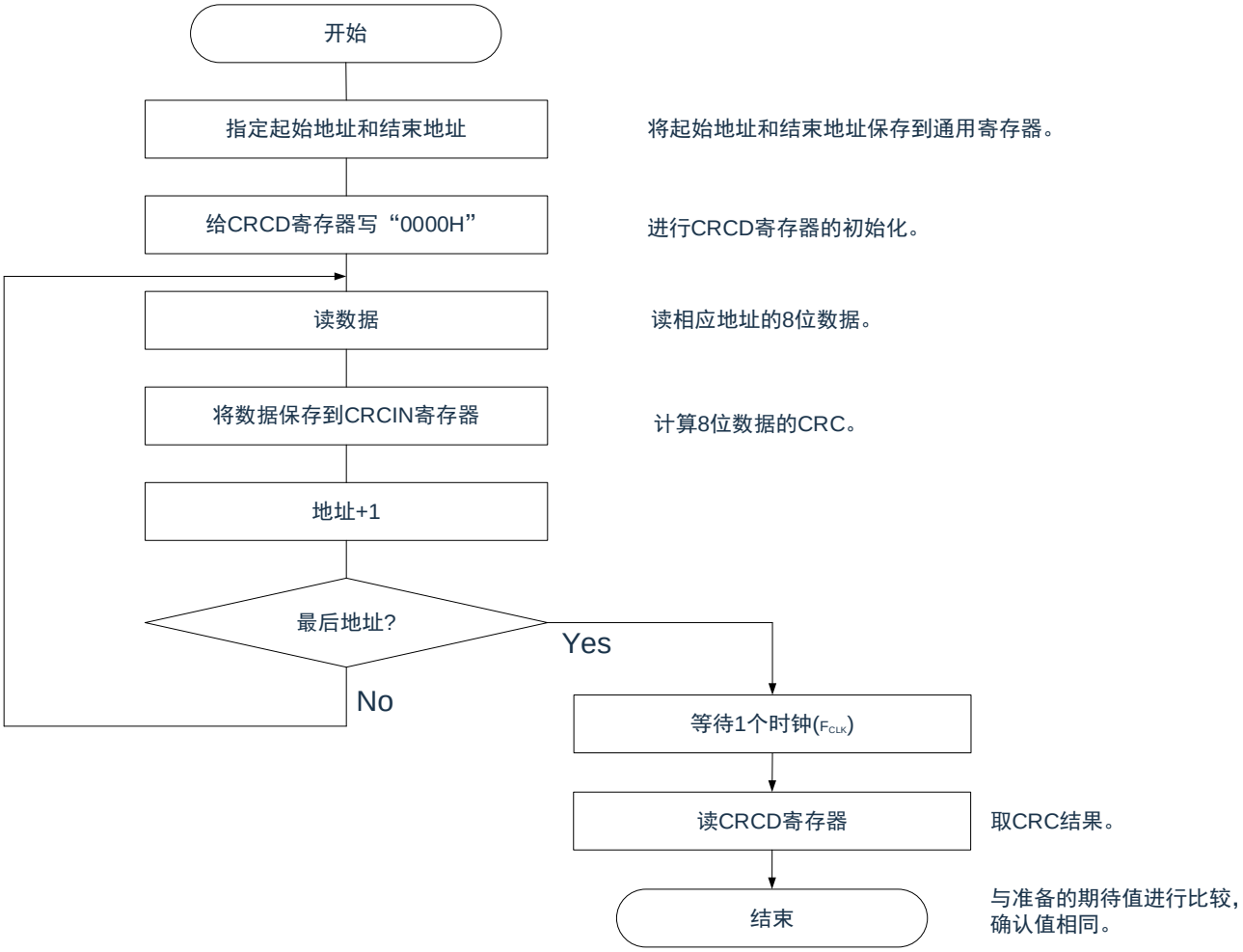
位	符号	描述	复位值
15:0	CRCD	保存通用 CRC 运算结果 0000H ~FFFFH	0x0

注1：要读CRCD寄存器的写入值时，必须在写CRCIN寄存器前读CRCD寄存器。

注2：如果CRCD寄存器的写操作与运算结果的保存发生竞争，就忽视写操作。

＜操作流程＞

图27-2：CRC运算功能（通用CRC）的流程图



27.3.2 SFR保护功能

为了必须保证运行过程中的安全，IEC61508 标准要求即使 CPU 失控也需要保护重要的 SFR，使其免遭改写 SFR 保护功能用于保护端口功能、时钟控制功能和电压检测电路的控制寄存器的数据。

如果设定为 SFR 保护功能，被保护的 SFR 的写操作就无效，但是能正常读取。

27.3.2.1 SFR保护控制寄存器（SFRGD）

此寄存器控制 SFR 保护功能是否有效。

SFR 保护功能使用 GPORT 位和 GCSC 位。

通过 8 位存储器操作指令设定 SFRGD 寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

位	符号	描述	复位值
7:3	保留	-	-
2	GPORT	端口功能的控制寄存器的保护 0: 无效，能读写端口功能的控制寄存器。 1: 有效，端口功能的控制寄存器的写操作无效，能读。 [被保护的 SFR]PMxx、PUxx、PDxx、POMxx、PMCxx、PxxCFG [注]。	0
1	-	保留	-
0	GCSC	时钟控制功能、电压检测电路的控制寄存器的保护 0: 无效，能读写时钟控制功能和电压检测电路的控制寄存器。 1: 有效。时钟控制功能和电压检测电路的控制寄存器的写操作无效，能读。 [被保护的 SFR]CSC、OSTS、CKC、PERx、OSMC、LVIM、LVIS。	0

注 ：不保护Pxx（端口寄存器）。

27.3.3 频率检测功能

IEC60730 标准要求确认振荡频率是否正常。

频率检测功能可使用 CPU/外围硬件的时钟频率（F_{CLK}），并且能通过测量 Timer40 通道 1 输入脉冲，判断 2 个时钟的比率关系是否正确。

但是，如果某 1 个时钟或者 2 个时钟停止振荡，就不能判断 2 个时钟的比率关系。

＜要比较的时钟＞

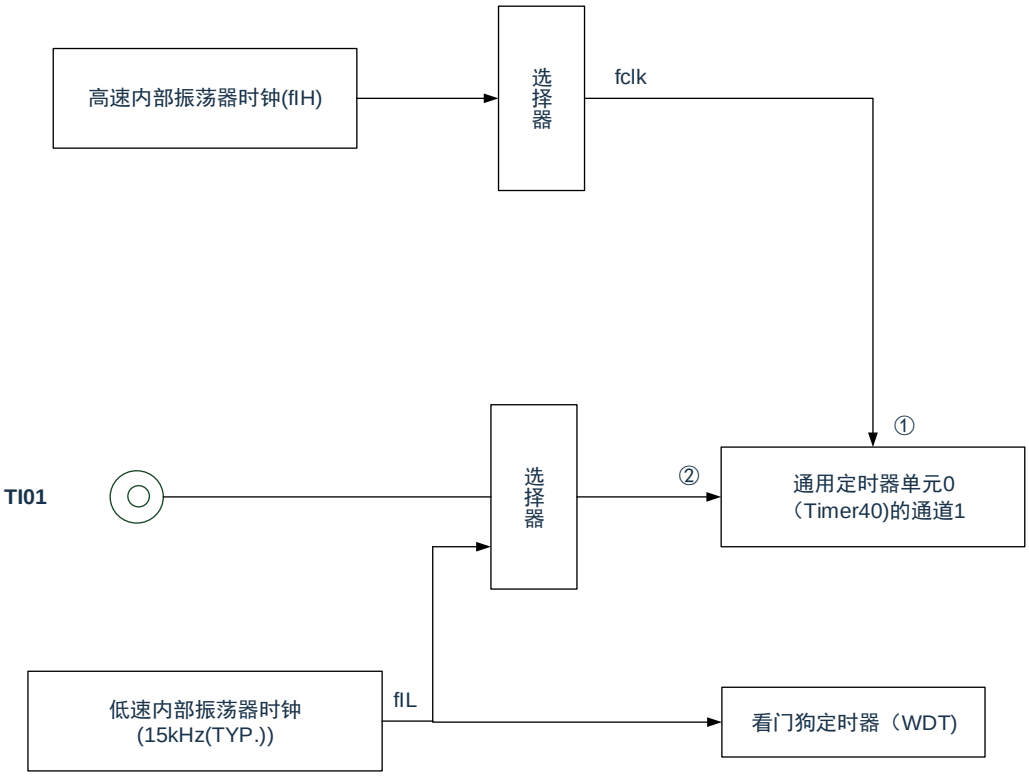
① CPU/外围硬件的时钟频率（F_{CLK}）：

- 高速内部振荡器时钟（F_{IH}）

②Timer40 通道 1 输入：

- 通道 1 的定时器输入（TI01）
- 低速内部振荡器时钟（F_{IL}：15kHz(TYP.）

图27-3：频率检测功能的结构



当输入脉冲间隔的测量结果为异常值时，能判断为“时钟频率异常”。有关输入脉冲间隔的测量方法，请参照“6.7.4 为输入脉冲间隔测量的运行”。

注：只有内置副系统时钟的产品才能选择。

27.3.3.1 定时器输入输出选择寄存器0（TIOS0）

寄存器说明请参考6.2.11节。

27.3.4 A/D测试功能

IEC60730标准要求进行A/D转换器的测试。此A/D测试功能通过对A/D转换器的正（+）基准电压、负（–）基准电压、模拟输入通道（ANI）、温度传感器的输出电压和内部基准电压进行A/D转换，确认A/D转换器是否正常运行。

能通过以下步骤确认模拟多路转换器：

① 通过CON2寄存器选择ANIx引脚作为A/D转换对象（ADCSWCHS = 00100）。

② 对ANIx引脚进行A/D转换（转换结果1-1）。

③ 通过CON2寄存器选择A/D转换器的负（–）基准电压作为A/D转换对象（ADCSWCHS = 10101）。

④ 对A/D转换器的负（–）基准电压进行A/D转换（转换结果2-1）。

⑤ 通过CON2寄存器选择ANIx引脚作为A/D转换对象（ADCSWCHS = 00100）。

⑥ 对ANIx引脚进行A/D转换（转换结果1-2）。

⑦ 通过CON2寄存器选择A/D转换器的正（+）基准电压作为A/D转换对象（ADCSWCHS = 10100）。

⑧ 对A/D转换器的正（+）基准电压进行A/D转换（转换结果2-2）。

⑨ 通过CON2寄存器选择ANIx引脚作为A/D转换对象（ADCSWCHS = 00100）。

⑩ 对ANIx引脚进行A/D转换（转换结果1-3）。

⑪ 确认“转换结果1-1”、“转换结果1-2”和“转换结果1-3”相同。

⑫ 确认“转换结果2-1”的A/D转换结果全部为“0”并且“转换结果2-2”的A/D转换结果全部为“1”。通过以上步骤，能选择模拟多路转换器以及确认布线没有断线。

注1：在①～⑩的转换过程中，如果模拟输入电压可变，就必须采用其他方法来确认模拟多路转换器。

注2：转换结果含有误差，因此必须在比较转换结果时要适当考虑误差。

27.3.4.1 ADC寄存器

ADCCON2 寄存器选择 A/D 转换器的正（+）基准电压、负（–）基准电压、模拟输入通道（ANIxx）、温度传感器的输出电压和内部基准电压（1.45V）作为 A/D 转换对象。

当用作 A/D 测试功能时，进行以下的设定：

- 在测量零刻度时，选择负（–）基准电压作为 A/D 转换对象。
- 在测量满刻度时，选择正（+）基准电压作为 A/D 转换对象。

注1：使用该模式时，A/D的正（+）基准电压需选择VDD。

注2：使用该模式时，ADCSWCHE需置一（通道开启由ADCSWCHS控制）。

A/D寄存器及使用说明请参考17章内容。

27.3.5 输入/输出引脚的数字输出信号电平检测功能

IEC60730 标准要求确认 I/O 功能是否正常。

输入/输出引脚的数字输出信号电平检测功能在引脚为输出模式时，能读引脚的数字输出电平。

27.3.5.1 端口模式选择寄存器（PMS）

此寄存器选择在引脚为输出模式（端口模式寄存器（PMm）的 PMmn 位为“0”）时是读端口的输出锁存器的值还是读引脚的输出电平。

通过 8 位存储器操作指令设定 PMS 寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

位	符号	描述	复位值
7:1	-	保留	-
0	PMS0	在引脚为输出模式时读数据的选择 0: 读 Pmn 寄存器的值。 1: 读引脚的数字输出电平。	0

注1：对于使用定时器M的脉冲输出强制截止功能而使引脚变为高阻抗状态的引脚，如果读引脚的数字输出电平，读取值就为“0”。

注2：m=0～2

n=0～7

27.3.6 产品唯一身份标识寄存器

产品唯一的身份标识非常适合：

用来作为序列号(例如 USB 字符序列号或者其他的终端应用)

用来作为密码，在编写闪存时，将此唯一标识与软件加解密算法结合使用，提高代码在闪存存储器内的安全性。

用来激活带安全机制的自举过程

128 位的产品唯一身份标识所提供的参考号码对任意一个 BAT32 微控制器，在任何情况下都是唯一的。用户在何种情况下，都不能修改这个身份标识。

产品唯一身份标识寄存器 0（UID0）

位	符号	描述	复位值
31:0	-	产品唯一身份标识的第[31:0]位，其值在出厂时编写	-

产品唯一身份标识寄存器 1（UID1）

位	符号	描述	复位值
31:0	-	产品唯一身份标识的第[63:32]位，其值在出厂时编写	-

产品唯一身份标识寄存器 2（UID2）

位	符号	描述	复位值
31:0	-	产品唯一身份标识的第[95:64]位，其值在出厂时编写	-

产品唯一身份标识寄存器 3（UID3）

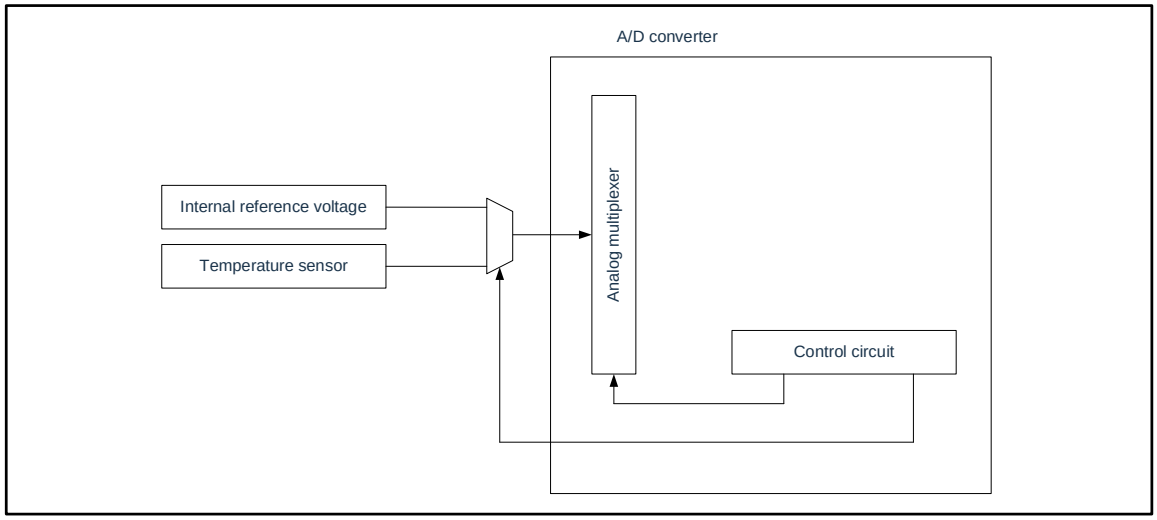
位	符号	描述	复位值
31:0	-	产品唯一身份标识的第[127:96]位，其值在出厂时编写	-

第28章 温度传感器

28.1 温度传感器的功能

片上的温度传感器可以对产品的核心温度进行测量和监控，从而保证产品的可靠运行。温度传感器输出的电压与核心温度成正比，并且电压和温度之间是线性关系。其输出电压提供给 ADC 进行转换。图 28-1 显示了温度传感器框图。

图28-1：温度传感器框图



28.2 寄存器映射

RO：只读，WO：只写，R/W：读写

寄存器	地址	读/写	描述	复位值
TSN25	0x0050066C	RO	温度传感器校准数据寄存器	-

28.3 温度传感器的寄存器

28.3.1 温度传感器校准数据寄存器 TSN25

位	符号	描述	复位值
15:12	-	保留	-
11:0	TSN25	校准数据 1，在接通电源或者复位启动时自动载入，每颗芯片有自己的校准数据。	-

28.4 温度传感器的使用说明

温度（T）与传感器电压输出（Vs）成正比，因此温度的计算公式如下：

T = (Vs - V1) / slope + 25℃

T：测量的温度（℃）

Vs：温度传感器在温度测量时的输出电压（V）

V1：温度传感器测量25℃时的电压输出（V）

Slope：温度传感器的温度斜率(V/°C)，slope = -3.5 mV/℃

备注：温度传感器的精度较低，不建议使用在精度要求高的场合。

第29章 选项字节

29.1 选项字节的功能

CMS32M65xx 的闪存 000C0H～000C3H，500004H 为选项字节区。

选项字节由用户选项字节（000C0H～000C2H）和闪存数据保护选项字节（000C3H，500004H）构成。在接通电源或者复位启动时，自动参照选项字节进行指定功能的设定。在使用本产品时，必须通过选项字节进行以下功能的设定。对于没有配置功能的位，不能更改初始值。

注意 与是否使用各功能无关，必须设定选项字节。

29.1.1 用户选项字节（000C0H～000C2H）

(1) 000C0H

- 看门狗定时器的运行
  - 允许或者禁止计数器的运行。
  - 在睡眠/深度睡眠模式中允许或者停止计数器的运行。
- 看门狗定时器的上溢时间的设定
  - 看门狗定时器的窗口打开期间的设定
- 看门狗定时器的间隔中断的设定
  - 使用或者不使用间隔中断。

(2) 000C1H

- LVD运行模式的设定
  - 中断&复位模式
  - 复位模式
  - 中断模式
  - LVD为OFF（使用RESETB引脚的外部复位输入）。
- LVD检测电平（VLVDH、VLVDL、VLVD）的设定

注：当电源电压上升时，必须在电源电压达到数据手册的AC特性所示的工作电压范围前，通过电压检测电路或者外部复位保持复位状态；当电源电压下降时，必须在电源电压低于工作电压范围前，通过深度睡眠模式的转移、电压检测电路或者外部复位，置为复位状态。

工作电压范围取决于用户选项字节（000C2H）的设定。

(3) 000C2H

- 高速内部振荡器的频率设定
  - 从2MHz～32MHz、48MHz、64MHz中选择。

29.2 闪存数据保护选项字节（000C3H，500004H）

- 片上调试时闪存数据保护的控制

Level0：允许通过 debugger 对闪存数据进行读出/写入/擦除操作

Level1：允许通过 debugger 对闪存数据进行 chip 全擦除操作，不允许进行读写操作。

Level2：不允许通过 debugger 对闪存数据进行操作。

29.3 寄存器映射

寄存器	地址	读/写	描述	复位值
选项字节0	0x00C0H	R/W	看门狗定时器状态控制寄存器	0xFF
选项字节1	0x00C1H	R/W	LVD状态控制寄存器	0xFF
选项字节2	0x00C2H	R/W	高速内部振荡器频率控制寄存器	0xEC
选项字节3	0x00C3H	R/W	片上调试闪存数据保护控制寄存器1	0xFF
选项字节4	0x500004H	R/W	片上调试闪存数据保护控制寄存器2	0xFF

29.4 用户选项字节

29.4.1 用户选项字节（000C0H）

位	符号	描述	复位值
7	WDTINT	看门狗定时器的间隔中断的使用/不使用 0: 不使用间隔中断 1: 当达到上溢时间的75%+1/2F_{IL}时，产生间隔中断	1
6:5	WINDOW[1:0]	看门狗定时器的窗口打开期间 0X: 禁止设定。 10: 75% 11: 100%	0x3
4	WDTON	看门狗定时器的计数器运行控制 0: 禁止计数器的运行（解除复位后停止计数） 1: 允许计数器的运行（解除复位后开始计数）	1
3:1	WDTCS[2:0]	看门狗定时器的上溢时间（F_{IL}=20kHz(MAX.)） 000: $2^6/F_{IL}$（3.2ms） 001: $2^7/F_{IL}$（6.4ms） 010: $2^8/F_{IL}$（12.8ms） 011: $2^9/F_{IL}$（25.6ms） 100: $2^{11}/F_{IL}$（102.4ms） 101: $2^{13}/F_{IL}$（409.6ms） 110: $2^{14}/F_{IL}$（819.2ms） 111: $2^{16}/F_{IL}$（3276.8ms）	0x7
0	WDSTBYON	看门狗定时器的计数器运行控制（睡眠模式） 0: 在睡眠模式中，停止计数器的运行 （注1 1: 在睡眠模式中，允许计数器的运行	1

注1：当WDSTBYON位为“0”时，与WINDOW1位和WINDOW0位的值无关，窗口打开期间为100%。
注2：IL：低速内部振荡器的时钟频率

29.4.2 用户选项字节（000C1H）

位	符号	描述		复位值
7:5	VPOC[2:0]	检测电压设定		0x7
4	-	保留（须为1）		1
3:2	LVIS[1:0]	检测电压设定		0x2
1:0	LVIMDS[1:0]	模式选择		0x3
		10:	中断&复位模式	
		11:	复位模式	
		01:	中断模式	

LVD检测电压设定（中断&复位模式）

检测电压			选项字节的设定值						
VLVDH		VLVDL	VPOC2	VPOC1	VPOC0	LVIS1	LVIS0	模式设定	
上升	下降	下降						LVIMDS1	LVIMDS0
1.77V	1.73V	1.63V	0	0	0	1	0	1	0
1.88V	1.84V					0	1		
2.92V	2.86V					0	0		
1.98V	1.94V	1.84V		0	1	1	0		
2.09V	2.04V					0	1		
3.13V	3.06V					0	0		
2.61V	2.55V	2.45V		1	0	1	0		
2.71V	2.65V					0	1		
3.75V	3.67V					0	0		
2.92V	2.86V	2.75V		1	1	1	0		
3.02V	2.96V					0	1		
4.06V	3.98V					0	0		
—			禁止设定上述以外的值。						

注1：：必须给bit4写“1”。

注2：有关LVD电路的详细内容，请参照“第26章 电压检测电路”。

注3：测电压是TYP.值。详细内容请参照数据手册的LVD电路特性。

LVD检测电压设定（复位模式）

检测电压		选项字节的设定值						
VLVD		VPOC2	VPOC1	VPOC0	LVIS1	LVIS0	模式设定	
上升	下降						LVIMDS1	LVIMDS0
1.67V	1.63V	0	0	0	1	1	1	1
1.77V	1.73V		0	0	1	0		
1.88V	1.84V		0	1	1	1		
1.98V	1.94V		0	1	1	0		
2.09V	2.04V		0	1	0	1		
2.50V	2.45V		1	0	1	1		
2.61V	2.55V		1	0	1	0		
2.71V	2.65V		1	0	0	1		
2.81V	2.75V		1	1	1	1		
2.92V	2.86V		1	1	1	0		
3.02V	2.96V		1	1	0	1		
3.13V	3.06V		0	1	0	0		
3.75V	3.67V		1	0	0	0		
4.06V	3.98V		1	1	0	0		
—		禁止设定上述以外的值。						

注1：必须给bit4写“1”。

注2：有关LVD电路的详细内容，请参照“第26章 电压检测电路”。

注3：检测电压是TYP.值。详细内容请参照数据手册的LVD电路特性。

LVD检测电压设定（中断模式）

检测电压		选项字节的设定值						
VLVD		VPOC2	VPOC1	VPOC0	LVIS1	LVIS0	模式设定	
上升	下降						LVIMDS1	LVIMDS0
1.67V	1.63V	0	0	0	1	1	0	1
1.77V	1.73V		0	0	1	0		
1.88V	1.84V		0	1	1	1		
1.98V	1.94V		0	1	1	0		
2.09V	2.04V		0	1	0	1		
2.50V	2.45V		1	0	1	1		
2.61V	2.55V		1	0	1	0		
2.71V	2.65V		1	0	0	1		
2.81V	2.75V		1	1	1	1		
2.92V	2.86V		1	1	1	0		
3.02V	2.96V		1	1	0	1		
3.13V	3.06V		0	1	0	0		
3.75V	3.67V		1	0	0	0		
4.06V	3.98V		1	1	0	0		
—		禁止设定上述以外的值。						

注1：必须给bit4写“1”。

注2：有关LVD电路的详细内容，请参照“第26章 电压检测电路”。

注3：.检测电压是TYP.值。详细内容请参照数据手册的LVD电路特性。

LVD为OFF时的设定（使用RESETB引脚的外部复位输入）

检测电压		选项字节的设置值						
V_{LVDH}		VPOC2	VPOC1	VPOC0	LVIS1	LVIS0	模式设定	
上升	下降						LVIMDS1	LVIMDS0
—	—	1	×	×	×	×	×	1
—		禁止设置上述以外的值。						

注1：必须给bit4写“1”。

注2：当电源电压上升时，必须在电源电压达到数据手册的AC特性所示的工作电压范围前，通过电压检测电路或者外部复位保持复位状态；当电源电压下降时，必须在电源电压低于工作电压范围前，通过睡眠模式的转移、电压检测电路或者外部复位，置为复位状态。

工作电压范围取决于用户选项字节（000C2H）的设定。

注3：×：忽略

注4：有关LVD电路的详细内容，请参照“第26章 电压检测电路”。

注5：检测电压是TYP.值。详细内容请参照数据手册的LVD电路特性。

29.4.3 用户选项字节（000C2H）

位	符号	描述	复位值
7:5	-	保留（须为1）	0x7
4:0	FRQSE[4:0]	高速内部震荡器时钟频率选择	0x0C

FRQSE4	FRQSEL3	FRQSEL2	FRQSEL1	FRQSEL0	高速内部振荡器的时钟频率	
					F_{HOCO}	F_{IH}
0	1	0	0	0	64MHz	64MHz
0	0	0	0	0	48MHz	48MHz
0	1	0	0	1	64MHz	32MHz
0	0	0	0	1	48MHz	24MHz
0	1	0	1	0	64MHz	16MHz
0	0	0	1	0	48MHz	12MHz
0	1	0	1	1	64MHz	8MHz
0	0	0	1	1	48MHz	6MHz
0	1	1	0	0	64MHz	4MHz
0	0	1	0	0	48MHz	3MHz
0	1	1	0	1	64MHz	2MHz
上述以外					禁止设定。	

注1：必须给bit7~5写“1”。

注2：工作频率范围和工作电压范围因闪存的各运行模式而不同。详细内容请参照数据手册的AC特性。

29.4.4 闪存数据保护选项字节（000C3H）

位	符号	描述	复位值
7:0	OCDEN[7:0]	闪存数据保护的控制	0xFF

29.4.5 闪存数据保护选项字节（500004H）

位	符号	描述	复位值
7:0	OCDM[7:0]	闪存数据保护的控制	0xFF

OCDM	OCDEN	闪存数据保护的控制
3C	C3	不允许通过debugger对闪存数据进行操作。
3C以外的值	C3	允许通过debugger对闪存数据进行chip全擦除操作，不允许进行读写操作。
上记以外		允许通过debugger对闪存数据进行读出/写入/擦除操作

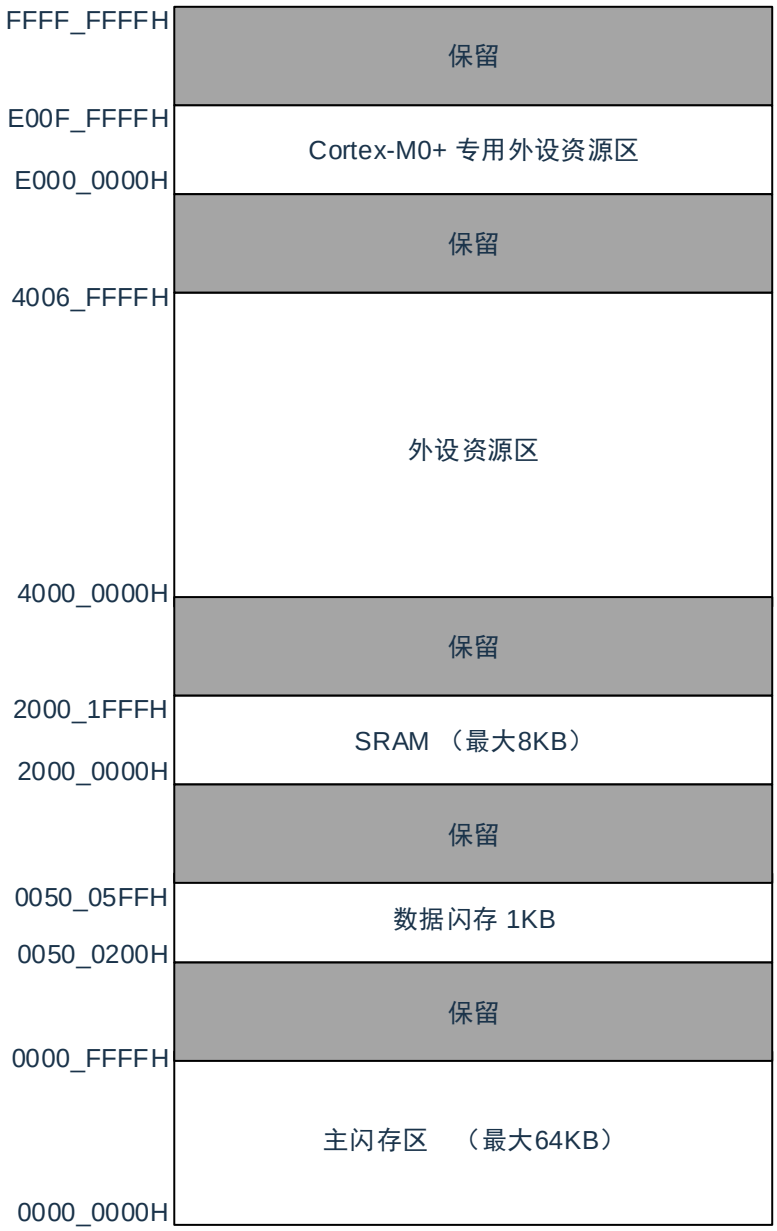
注：50_0004H地址属于数据闪存区，如果使用该地址做数据存储用，需先确定数值不会引起保护选项的误设。

第30章 FLASH控制

30.1 FLASH控制功能描述

本制品包含一颗 64KByte 容量的 FLASH 存储器，共划分为 128 个 Sector，每个 Sector 容量为 512Byte。可做为程序存储器，数据存储器。本模块支持对该存储器的擦除、编程以及读取操作。

30.2 FLASH存储器结构



30.3 寄存器映射

(FLASH控制基地址 = 0x4002_0000)RO：只读，WO：只写，R/W：读写

寄存器	偏移量	读/写	描述	复位值
FLSTS	0x000	R/W	FLASH状态寄存器	0x0
FLOPMD1	0x004	R/W	FLASH操作控制寄存器1	0x0
FLOPMD2	0x008	R/W	FLASH操作控制寄存器2	0x0
FLERMD	0x00C	R/W	FLASH擦除控制寄存器	0x0
FLCERCNT	0x010	R/W	FLASH全片擦除时间控制寄存器	-
FLSERCNT	0x014	R/W	FLASH页擦除时间控制寄存器	-
FLPROCNT	0x01C	R/W	FLASH写入时间控制寄存器	-
FLPROT	0x020	R/W	FLASH写保护寄存器	0x0

30.4 寄存器说明

30.4.1 Flash写保护寄存器(FLPROT)

Flash保护寄存器用于对FLASH操作控制寄存器进行保护。

位	符号	描述	复位值
31:8	-	保留	-
7:1	PRKEY	WRP写保护 78h: 允许改写WRP 其他 不允许改写WRP	0x0
0	WRP	操作寄存器（FLOPMD1/FLOPMD2）写保护 0: 不允许改FLOPMD1/FLOPMD2 1: 允许改写FLOPMD1/ FLOPMD2	0

30.4.2 FLASH操作控制寄存器(FLOPMD1)

Flash操作控制寄存器，用于设定FLASH的擦除和写入操作。

位	符号	描述	复位值
31:8	-	保留	-
7:0	FLOPMD1	FLASH操作选择位： 0x55: 当FLOPMD2=0xAA时：擦除操作 0xAA: 当FLOPMD2=0x55时：写入操作 0x00: 当FLOPMD2=0x00时：读取操作 上记以外: 禁止设定	0x0

30.4.3 FLASH操作控制寄存器(FLOPMD2)

Flash操作控制寄存器，用于设定FLASH的擦除和写入操作。

位	符号	描述	复位值
31:8	-	保留	-
7:0	FLOPMD2	FLASH操作选择位： 0xAA: 当FLOPMD1=0x55时：擦除操作 0x55: 当FLOPMD1=0xAA时：写入操作 0x00: 当FLOPMD1=0x00时：读取操作 上记以外: 禁止设定	0x0

30.4.4 Flash擦除控制寄存器(FLERMD)

Flash擦除控制寄存器，用于设定FLASH擦除操作的类型。

位	符号	描述	复位值
7:5	-	保留	-
4:3	ERMD	擦除操作控制位： 0: sector擦除，擦除后不进行硬件校验 1: chip擦除（注： 2: sector擦除，擦除后进行硬件校验 3: 禁止设定	0x0
2:0	-	保留	-

注：chip擦除只擦除代码闪存区域，不擦除数据闪存区域。且chip擦除不支持硬件校验。

30.4.5 Flash状态寄存器(FLSTS)

通过状态寄存器可以查询FLASH控制器的状态。

位	符号	描述	复位值
7:3	-	保留	-
2	EVF	FLASH擦除硬件校验错误标志位： 0: 擦除后，硬件校验没有发生错误 1: 擦除后，硬件校验发生错误	0
1	-	保留	-
0	OVF	FLASH擦写操作完成标志位： 0: 擦写操作未完成 1: 擦完操作完成擦除	0

注1：OVF需要软件写”1”进行清除。若不清除，不能进行下一次擦写操作。
注2：EVF需要软件写”1”进行清除。

30.4.6 Flash全片擦除时间控制寄存器(FLCERCNT)

通过FLCERCNT寄存器可以设置FLASH全片擦除的时间。

位	符号	描述	复位值
31	Load	全片擦除时间设定选择（注： 0: 使用硬件设定的擦除时间 1: 使用软件设定的擦除时间（FLCERCNT[9:0]）	0
30:10	-	保留	-
9:0	FLCERCNT	软件擦除时间设定： Chip擦除时间=（FLCERCNT*2048*Tfclk），需满足>30ms的硬件要求	不定值

注：在主时钟为内部高速OCO时钟<=20M时，可以使用硬件设定时间，不设定FLCERCNT

30.4.7 Flash页擦除时间控制寄存器（FLSERCNT）

通过FLSERCNT寄存器可以设置FLASH全片擦除的时间。

位	符号	描述	复位值
31	Load	页擦除时间设定选择（注： 0: 使用硬件设定的擦除时间 1: 使用软件设定的擦除时间（FLSERCNT[10:0]）	0
30:11	-	保留	-
10:0	FLSERCNT	软件擦除时间设定： sector擦除时间=（FLSERCNT*256*Tfclk），需满足>2ms的硬件要求	不定值

注：在主时钟为内部高速OCO时钟<=20M时，可以使用硬件设定时间，不设定FLSERCNT。

30.4.8 Flash 写入时间控制寄存器（FLPROCNT）

通过FLPROCNT寄存器可以设置FLASH WORD写入的时间。

位	符号	描述	复位值
31	Load1	写入动作建立时间（Tpgs）设定选择（注1： 0: 使用硬件设定的时间 1: 使用软件设定的时间 FLPGSCNT[12:0]	0
30:29	-	保留	-
28:16	FLPGSCNT	软件写入动作建立时间设定： 写入动作建立时间=（FLPGSCNT*Tfclk），写字节需满足>70us的硬件要求，写半字需满足>40us的硬件要求	不定值
15	Load0	写入时间（Tprog）设定选择 （注2： 0: 使用硬件设定的时间 1: 使用软件设定的时间 FLPROCNT[8:0]	0
14:9	-	保留	-
8:0	FLPROCNT	软件写入时间设定： 写入时间=（FLPROCNT*Tfclk），需满足>7us的硬件要求	不定值

注1：在主时钟为内部高速OCO或者外部输入时钟<=20M时，可以使用硬件设定时间，不设定FLPGSCNT。

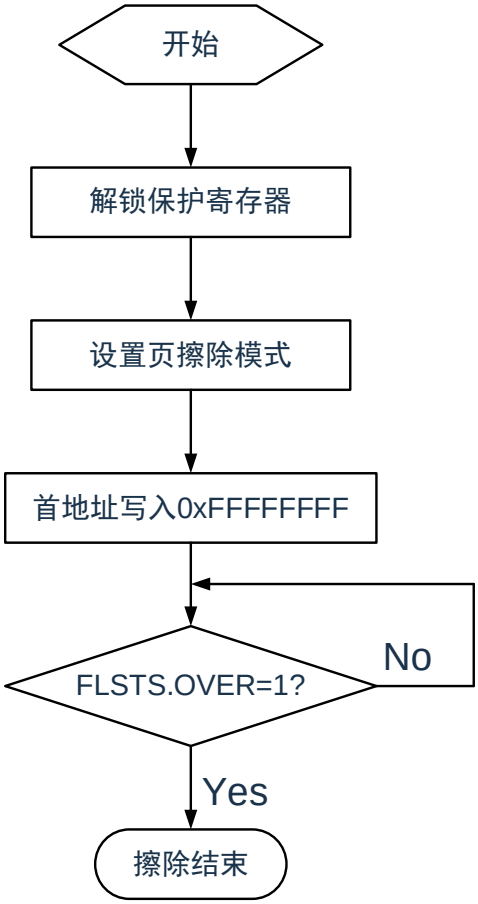
注2：在主时钟为内部高速OCO或者外部输入时钟<=20M时，可以使用硬件设定时间，不设定FLPROCNT。

30.5 FLASH操作方法

30.5.1 页擦除（sector erase）

Sector 擦除，擦除时间由硬件实现，也可以通过 FLSERCNT 配置。操作流程如下：

1) 设置 FLERMD.ERMD0 为 1'b0，选择 sector 擦除模式,根据是否需要硬件校验选择设置 ERMD1 的值；
2) 设置 FLPROT 为 0xF1，解除 FLOPMD 的保护。然后将 FLOPMD1 设置为 0x55，FLOPMD2 设置为 0xAA，
3) 向擦除目标 sector 的首地址写入任意数据。例：*（(unsigned long *）0x00000200）=0xffffffff。
4) 软件查询状态寄存器 FLSTS.OVF,OVF=1 时，表示擦除操作完成。
5) 如果设置了擦除后进行硬件校验（ERMD1=1），可以通过软件判断 FLSTS.EVF，查询是否校验正确。
6) 进行下次操作前，软件置"1"来清除 FLSTS。



30.5.2 全片擦除（chip erase）

Chip擦除，擦除时间由硬件实现，也可以通过FLCERCNT配置。操作流程如下：

1)设置 FLERMD.ERMD0 为 1’b1，选择 chip 擦除模式；

2)设置 FLPROT 为 0xF1，解除 FLOPMD 的保护。然后将 FLOPMD1 设置为 0x55，FLOPMD2 设置为 0xAA，

3)向代码闪存区域的任意地址写入任意数据。

4)软件查询状态寄存器 FLSTS.OVF,OVF=1 时，表示擦除操作完成。

5)进行下次操作前，软件置”1”来清除 FLSTS。

30.5.3 编程（word program）

Word编程，写入时间由硬件实现，也可以通过PROCNT配置。操作流程如下：

1)设置 FLPROT 为 0xF1，解除 FLOPMD 的保护。然后将 FLOPMD1 设置为 0xAA，FLOPMD2 设置为 0x55，

2)向目标地址写入相应的数据。

3)软件查询状态寄存器 FLSTS.OVF,OVF=1 时，表示写入操作完成。

4)进行下次操作前，软件置”1”来清除 FLSTS。

30.6 闪存读取

本设备内置的 FLASH 支持的最快取指频率为 32MHz。当 HCLK 频率超过 32MHz 时，硬件会在 CPU 访问 FLASH 时插入 1 等待周期。

30.7 FLASH操作的注意事项

1)FLASH 存储器对擦除和编程操作的控制信号具有严格的时间要求，控制信号的时序不合格会造成擦除操作和编程操作失败。擦写参数的设置可以由硬件实现，也可通过修改参数寄存器进行软件修改；在使用内部高速 OCO，MAINOSC/外部输入时钟=20M 时，推荐使用硬件设置的擦写参数，无需设置参数寄存器。

2)如果擦写操作从 FLASH 内执行，则 CPU 会停止取指，硬件自动等待操作完成后，继续下一条指令。如果该操作从 RAM 里执行，CPU 不会停止取指，当前可以继续下一条指令。

3)在 FLASH 处于编程操作中时，如果 CPU 执行进入深睡眠的指令，系统将等待编程动作结束后才会进入深睡眠。

附录 修订记录

版本	日期	修订内容
V0.1.0	2023年2月	初始版本
V0.5.0	2023年8月	1) 优化格式 2) 更正全文内容中引用链接 3) 修改3.4.9寄存器、表3-2描述 4) 更正5.4.6、5.7符号SELLOSC 5) 增加二级标题6.1 6) 更正1.2、2.1、3.2、5.4.3、6.2.4、7.4、8.3.2、12.3.1、14.1、17.1、17.2、19.5、20.3章节中有误内容 7) 14.4、14.5.9章节新增END寄存器 8) 修改17.3.1、17.3.2、17.3.6、17.5.1章节中内容 9) 补充17.3.8 图17-1：ADC硬件触发启动 10) 修改17.4 寄存器映射表中内容并删除DATA19寄存器 11) 修改17.5.2、17.5.4寄存器内容并添加“注” 12) 17.5.5 通道使能寄存器新增说明 13) 17.5.5~17.5.14新增条件或备注 14) 删除17.5.9 ADC测试寄存器(TEST)章节 15) 删除LPF相关内容 16) 更正20.5.2寄存器名称 17) 修改21.2结构框图 18) 第27章安全功能章节新增PGCRCH寄存器相关说明 19) 修改27.3.4.1ADC寄存器内容 20) 更正29.3 寄存器选项字节2、29.4.3 bit[4:0]复位值 21) 更正30.4.1、30.4.6~30.4.8章节寄存器内容 22) 删除RAM1，修改RAM0名称 23) 删除28.3温度传感器使用方法1，并只开放常温数据 24) 删除第9章WDT三处关于WDTE位操作复位功能描述 25) 修改17.3.2ADC框图中ADGO为ADCST 26) 修改17.5.1ADC控制寄存器描述 27) 更正29.4.1用户选项字节寄存器名称 28) 更正6.5.3描述参考图6-28为6-14 29) 更正6.8.3章节中图6-43下的(d)(e)内容 30) 更正26.4.1 寄存器LVIM和26.4.2寄存器LVIS的内容
V0.9.0	2023年10月	1) 14.5.9章节：增加END寄存器的注意事项 2) 更正PMCxx寄存器描述 3) 将3.4.10端口输入复用功能配置寄存器的TAU0_TINO-3改为TI00、TI01、TI02、TI03 4) 更正 图7-1：12位间隔定时器的运行时序 5) 更正NIVC支持离散中断个数 6) 更正LVIM寄存器的描述 7) 修改6.2.11章节内容 8) 修改第27章章节内容 9) 修改28.2章节内容