



Holtek 32-Bit 带 Arm® Cortex®-M0+ 内核单片机

HT32F50220/HT32F50230 开发使用手册

Revision: V1.10 Date: 2019-07-15

www.holtek.com

目录

1 简介	18
概述	18
特性	19
单片机信息	21
方框图	22
2 文档协议	23
3 系统结构	24
Arm® Cortex®-M0+ 处理器	24
总线结构	25
存储器体系	26
存储器映射	27
嵌入式 Flash 存储器	29
嵌入式 SRAM 存储器	29
AHB 外设	29
APB 外设	29
4 Flash 存储器控制器 (FMC)	30
简介	30
特性	30
功能描述	31
Flash 存储器映射	31
Flash 存储器结构	31
启动配置	32
页擦除	33
整片擦除	34
字编程	35
选项字节描述	36
页擦除 / 编程保护	36
安全保护	38
寄存器列表	39
寄存器描述	40
Flash 目标地址寄存器 – TADR	40
Flash 写数据寄存器 – WRDR	41
Flash 操作命令寄存器 – OCMR	42
Flash 操作控制寄存器 – OPCR	43
Flash 操作中断使能寄存器 – OIER	44
Flash 操作中断状态寄存器 – OISR	45
Flash 页擦除 / 编程保护状态寄存器 – PPSR	46
Flash 安全保护状态寄存器 – CPSR	47
Flash 向量映射控制寄存器 – VMCR	48

Flash 制造商和设备 ID 寄存器 – MDID.....	49
Flash 页数状态寄存器 – PNSR	50
Flash 页大小状态寄存器 – PSSR	51
设备 ID 寄存器 – DIDR	52
自定义 ID 寄存器 n – CIDRn, n = 0 ~ 3	53
5 电源控制单元 (PWRCU).....	54
简介	54
特性	55
功能描述	55
V _{DD} 电源域	55
1.5 V 电源域	56
工作模式	57
寄存器列表	58
寄存器描述	59
电源控制状态寄存器 – PWRSR.....	59
电源控制寄存器 – PWRCR	60
低电压 / 欠压检测控制和状态寄存器 – LVDCSR.....	62
6 时钟控制单元 (CKCU).....	64
简介	64
特性	66
功能描述	66
外部高速晶振 – HSE.....	66
内部高速 RC 振荡器 – HSI	67
内部高速 RC 振荡器 (HSI) 自动微调	67
外部低速晶振 – LSE	69
内部低速 RC 振荡器 – LSI.....	69
时钟就绪标志位	69
系统时钟 (CK_SYS) 选择	70
HSE 时钟监控	70
时钟输出能力	70
寄存器列表	71
寄存器描述	72
全局时钟配置寄存器 – GCFGR.....	72
全局时钟控制寄存器 – GCCR	73
全局时钟状态寄存器 – GCSR.....	74
全局时钟中断寄存器 – GCIR.....	75
AHB 配置寄存器 – AHBCFGR	76
AHB 时钟控制寄存器 – AHBCCR	77
APB 配置寄存器 – APBCFGR	78
APB 时钟控制寄存器 0 – APBCCR0.....	79
APB 时钟控制寄存器 1 – APBCCR1.....	80
时钟源状态寄存器 – CKST	81

APB 外设时钟选择寄存器 0 – APBPCSR0	82
APB 外设时钟选择寄存器 1 – APBPCSR1	84
HSI 控制寄存器 – HSICR	85
HSI 自动微调计数器寄存器 – HSIATCR	86
APB 外设时钟选择寄存器 2 – APBPCSR2	87
单片机调试控制寄存器 – MCUDBGCR	88
7 复位控制单元 (RSTCU)	90
简介	90
功能描述	91
上电复位	91
系统复位	91
AHB 和 APB 单元复位	91
寄存器列表	92
寄存器描述	92
全局复位状态寄存器 – GRSR	92
AHB 外设复位寄存器 – AHBPRSTR	93
APB 外设复位寄存器 0 – APBPRSTR0	94
APB 外设复位寄存器 1 – APBPRSTR1	95
8 通用 I/O (GPIO)	96
简介	96
特性	97
功能描述	97
默认的 GPIO 引脚配置	97
通用 I/O – GPIO	97
GPIO 锁定机制	99
寄存器列表	99
寄存器描述	100
端口 A 数据方向控制寄存器 – PADIRCR	100
端口 A 输入功能使能控制寄存器 – PAINER	101
端口 A 上拉选择寄存器 – PAPUR	102
端口 A 下拉选择寄存器 – PAPDR	103
端口 A 漏极开路选择寄存器 – PAODR	104
端口 A 输出驱动电流选择寄存器 – PADDRV	105
端口 A 锁定寄存器 – PALOCKR	106
端口 A 数据输入寄存器 – PADINR	107
端口 A 输出数据寄存器 – PADOUTR	108
端口 A 输出置位 / 复位控制寄存器 – PASRR	109
端口 A 输出复位寄存器 – PARR	110
端口 A 灌电流增强选择寄存器 – PASCER	111
端口 B 数据方向控制寄存器 – PBDIRCR	112
端口 B 输入功能使能控制寄存器 – PBINER	113
端口 B 上拉选择寄存器 – PBPUR	114

端口 B 下拉选择寄存器 – PBPDR.....	115
端口 B 漏极开路选择寄存器 – PBODR.....	116
端口 B 输出驱动电流选择寄存器 – PBDRVR.....	117
端口 B 锁定寄存器 – PBLOCKR.....	118
端口 B 数据输入寄存器 – PBDINR.....	119
端口 B 输出数据寄存器 – PBDOUTR.....	120
端口 B 输出置位 / 复位控制寄存器 – PBSRR.....	121
端口 B 输出复位寄存器 – PBRR.....	122
端口 B 灌电流增强选择寄存器 – PBSCER.....	123
端口 C 数据方向控制寄存器 – PCDIRCR.....	124
端口 C 输入功能使能控制寄存器 – PCINER.....	125
端口 C 上拉选择寄存器 – PCPUR.....	126
端口 C 下拉选择寄存器 – PCPDR.....	127
端口 C 漏极开路选择寄存器 – PCODR.....	128
端口 C 输出驱动电流选择寄存器 – PCDRVR.....	129
端口 C 锁定寄存器 – PCLOCKR.....	130
端口 C 数据输入寄存器 – PCDINR.....	131
端口 C 输出数据寄存器 – PCDOUTR.....	132
端口 C 输出置位 / 复位控制寄存器 – PCSRR.....	133
端口 C 输出复位寄存器 – PCRR.....	134
端口 C 灌电流增强选择寄存器 – PCSCER.....	135
9 复用功能输入 / 输出控制单元 (AFIO).....	136
简介.....	136
特性.....	137
功能描述.....	137
外部中断引脚选择.....	137
复用功能.....	138
锁定机制.....	138
寄存器列表.....	138
寄存器描述.....	139
EXTI 来源选择寄存器 0 – ESSR0.....	139
EXTI 来源选择寄存器 1 – ESSR1.....	140
GPIO 端口 x 配置低字节寄存器 – GPxCFGLR, x = A, B, C.....	141
GPIO 端口 x 配置高字节寄存器 – GPxCFGHR, x = A, B, C.....	142
10 嵌套向量中断控制器 (NVIC).....	143
简介.....	143
特性.....	144
功能描述.....	145
SysTick 校准.....	145
寄存器列表.....	145

11 外部中断 / 事件控制器 (EXTI)	146
简介	146
特性	146
功能描述	147
唤醒事件管理	147
外部中断 / 事件引脚配置	148
中断和去抖	148
寄存器列表	149
寄存器描述	150
EXTI 中断配置寄存器 n – EXTICFGRn, n = 0 ~ 15	150
EXTI 中断控制寄存器 – EXTICR	151
EXTI 中断边沿标志寄存器 – EXTIEDGEFLGR	152
EXTI 中断边沿状态寄存器 – EXTIEDGESR	153
EXTI 中断软件置位命令寄存器 – EXTISSCR	154
EXTI 中断唤醒控制寄存器 – EXTIWAKUPCR	155
EXTI 中断唤醒极性寄存器 – EXTIWAKUPPOLR	156
EXTI 中断唤醒标志寄存器 – EXTIWAKUPFLG	157
12 模数转换器 (ADC)	158
简介	158
特性	159
功能描述	160
ADC 时钟设置	160
通道选择	160
转换模式	160
外部事件启动转换	163
采样时间设定	164
数据格式	164
模拟看门狗	164
中断	164
寄存器列表	165
寄存器描述	166
ADC 转换控制寄存器 – ADCCR	166
ADC 转换列表寄存器 0 – ADCLST0	167
ADC 转换列表寄存器 1 – ADCLST1	168
ADC 输入采样时间寄存器 – ADCSTR	169
ADC 转换数据寄存器 y – ADCDRy, y = 0 ~ 7	170
ADC 触发控制寄存器 – ADCTCR	171
ADC 触发源寄存器 – ADCTSR	172
ADC 看门狗控制寄存器 – ADCWCR	173
ADC 看门狗阈值寄存器 – ADCTR	174
ADC 中断使能寄存器 – ADCIER	175
ADC 中断原始状态寄存器 – ADCIRAW	176

ADC 中断状态寄存器 – ADCISR	177
ADC 中断清除寄存器 – ADCICLR	178
13 通用定时器 (GPTM).....	179
简介	179
特性	180
功能描述	180
计数器模式	180
时钟控制器	183
触发控制器	184
从机控制器	185
主机控制器	187
通道控制器	188
输入级	191
正交解码器	193
输出级	195
更新管理	199
单脉冲模式	200
非对称 PWM 模式.....	202
定时器互连	203
触发 ADC 开启.....	205
寄存器列表	206
寄存器描述	207
定时器计数器配置寄存器 – CNTCFR.....	207
定时器模式配置寄存器 – MDCFR	208
定时器触发配置寄存器 – TRCFR.....	210
定时器控制寄存器 – CTR	211
通道 0 输入配置寄存器 – CH0ICFR.....	212
通道 1 输入配置寄存器 – CH1ICFR.....	214
通道 2 输入配置寄存器 – CH2ICFR.....	215
通道 3 输入配置寄存器 – CH3ICFR.....	216
通道 0 输出配置寄存器 – CH0OCFR	217
通道 1 输出配置寄存器 – CH1OCFR	219
通道 2 输出配置寄存器 – CH2OCFR	221
通道 3 输出配置寄存器 – CH3OCFR	223
通道控制寄存器 – CHCTR.....	225
通道极性配置寄存器 – CHPOLR	226
定时器中断控制寄存器 – DICTR	227
定时器事件发生器寄存器 – EVGR	228
定时器中断状态寄存器 – INTSR.....	229
定时器计数器寄存器 – CNTR	231
定时器预分频器寄存器 – PSCR	232
定时器计数器重载寄存器 – CRR	233
通道 0 捕捉 / 比较寄存器 – CH0CCR	234
通道 1 捕捉 / 比较寄存器 – CH1CCR	235

通道 2 捕捉 / 比较寄存器 – CH2CCR	236
通道 3 捕捉 / 比较寄存器 – CH3CCR	237
通道 0 非对称比较寄存器 – CH0ACR	238
通道 1 非对称比较寄存器 – CH1ACR	239
通道 2 非对称比较寄存器 – CH2ACR	240
通道 3 非对称比较寄存器 – CH3ACR	241
14 脉冲宽度调制器 (PWM).....	242
简介	242
特性	243
功能描述	243
计数器模式	243
时钟控制器	246
触发控制器	247
从机控制器	248
重启模式	248
暂停模式	249
触发模式	249
主机控制器	250
通道控制器	251
输出级	251
更新管理	255
单脉冲模式	255
非对称 PWM 模式	258
定时器互连	258
触发外设开启	260
寄存器列表	261
寄存器描述	262
定时器计数器配置寄存器 – CNTCFR	262
定时器模式配置寄存器 – MD CFR	263
定时器触发配置寄存器 – TRCFR	265
定时器控制寄存器 – CTR	266
通道 0 输入配置寄存器 – CH0OCFR	267
通道 1 输入配置寄存器 – CH1OCFR	269
通道 2 输入配置寄存器 – CH2OCFR	271
通道 3 输入配置寄存器 – CH3OCFR	273
通道控制寄存器 – CHCTR	275
通道极性配置寄存器 – CHPOLR	276
定时器中断控制寄存器 – DICTR	277
定时器事件发生器寄存器 – EVGR	278
定时器中断状态寄存器 – INTSR	279
定时器计数器寄存器 – CNTR	280
定时器预分频器寄存器 – PSCR	281
定时器计数器重载寄存器 – CRR	281
通道 0 比较寄存器 – CH0CR	282

通道 1 比较寄存器 – CH1CR	282
通道 2 比较寄存器 – CH2CR	283
通道 3 比较寄存器 – CH3CR	283
通道 0 非对称比较寄存器 – CH0ACR	284
通道 1 非对称比较寄存器 – CH1ACR	284
通道 2 非对称比较寄存器 – CH2ACR	285
通道 3 非对称比较寄存器 – CH3ACR	285
15 基本功能定时器 (BFTM)	286
简介	286
特性	286
功能描述	287
重复模式	287
单次模式	288
触发 A/D 转换器开启	288
寄存器列表	289
寄存器描述	289
BFTM 控制寄存器 – BFTMCR	289
BFTM 状态寄存器 – BFTMSR	290
BFTM 计数器值寄存器 – BFTMCNTR	291
BFTM 比较值寄存器 – BFTMCMPR	291
16 实时时钟 (RTC)	292
简介	292
特性	292
功能描述	293
RTC 相关寄存器复位	293
低速时钟配置	293
RTC 计数器操作	293
中断和唤醒控制	293
RTCCOUT 输出引脚配置	294
寄存器列表	295
寄存器描述	295
RTC 计数器寄存器 – RTCCNT	295
RTC 比较寄存器 – RTCCMP	296
RTC 控制寄存器 – RTCCR	297
RTC 状态寄存器 – RTCSR	299
RTC 中断和唤醒使能寄存器 – RTCIWEN	300
17 看门狗定时器 (WDT)	301
简介	301
特性	301
功能描述	302
寄存器列表	303

寄存器描述.....	304
看门狗定时器控制寄存器 – WDTCR.....	304
看门狗定时器模式寄存器 0 – WDTMR0.....	305
看门狗定时器模式寄存器 1 – WDTMR1.....	306
看门狗定时器状态寄存器 – WDTSR.....	307
看门狗定时器保护寄存器 – WDTPR.....	308
看门狗定时器时钟选择寄存器 – WDTCR.....	309
18 内部集成电路 (I²C).....	310
简介.....	310
特性.....	311
功能描述.....	311
双线串行接口.....	311
START 和 STOP 条件.....	311
数据有效性.....	312
寻址格式.....	313
数据传输和确认.....	315
时钟同步.....	315
仲裁.....	316
广播呼叫寻址.....	316
总线错误.....	316
地址屏蔽使能.....	316
地址捕获.....	317
工作模式.....	317
保持 SCL 线状态的条件.....	321
I ² C 超时功能.....	322
寄存器列表.....	323
寄存器描述.....	323
I ² C 控制寄存器 – I2CCR.....	323
I ² C 中断使能寄存器 – I2CIER.....	325
I ² C 地址寄存器 – I2CADDR.....	326
I ² C 状态寄存器 – I2CSR.....	327
I ² C SCL 高电平周期发生寄存器 – I2CSHPGR.....	329
I ² C SCL 低电平周期产生寄存器 – I2CSLPGR.....	330
I ² C 数据寄存器 – I2CDR.....	331
I ² C 目标寄存器 – I2CTAR.....	332
I ² C 地址屏蔽寄存器 – I2CADDMR.....	333
I ² C 地址捕获寄存器 – I2CADDSR.....	334
I ² C 超时寄存器 – I2CTOUT.....	335
19 串行外设接口 (SPI).....	336
简介.....	336
特性.....	337
功能描述.....	337

主机模式	337
从机模式	337
SPI 串行帧格式	337
状态标志	342
寄存器列表	344
寄存器描述	345
SPI 控制寄存器 0 – SPICR0	345
SPI 控制寄存器 1 – SPICR1	347
SPI 中断使能寄存器 – SPIIER	348
SPI 时钟预分频器寄存器 – SPICPR	349
SPI 数据寄存器 – SPIDR	350
SPI 状态寄存器 – SPISR	351
SPI FIFO 控制寄存器 – SPIFCR	352
SPI FIFO 状态寄存器 – SPIFSR	353
SPI FIFO 超时计数器寄存器 – SPIFTOCR	354
20 通用异步收发器 (UART)	355
简介	355
特性	356
功能描述	356
串行数据格式	356
波特率发生器	357
中断和状态	358
寄存器列表	358
寄存器描述	359
UART 数据寄存器 – URDR	359
UART 控制寄存器 – URCCR	360
UART 中断使能寄存器 – URIER	361
UART 状态 & 中断标志位寄存器 – URSIFR	362
UART 分频器锁存寄存器 – URDLR	364
UART 测试寄存器 – URTSTR	365
21 除法器 (DIV)	366
简介	366
特性	366
功能描述	366
寄存器列表	367
寄存器描述	367
除法器控制寄存器 – CR	367
被除数数据寄存器 – DDR	368
除数数据寄存器 – DSR	368
商数据寄存器 – QTR	369
余数数据寄存器 – RMR	369

表列表

表 1. 特性及外设列表	21
表 2. 文档协议	23
表 3. 寄存器映射	28
表 4. Flash 存储器和选项字节	32
表 5. 启动模式	32
表 6. 选项字节的存储器映射	36
表 7. 受保护主 Flash 页的访问权限	37
表 8. 安全保护使能时的访问权限	38
表 9. FMC 的寄存器列表	39
表 10. 工作模式定义	57
表 11. 进入 / 退出省电模式	58
表 12. 系统复位后的电源状态	58
表 13. PWRCU 寄存器列表	58
表 14. CKOUT 的时钟源	70
表 15. CKCU 寄存器列表	71
表 16. RSTCU 寄存器列表	92
表 17. AFIO、GPIO 和 I/O 引脚控制信号真值表	98
表 18. GPIO 寄存器列表	99
表 19. 外设 AFIO 分配列表范例	138
表 20. AFIO 寄存器列表	138
表 21. 异常类型	143
表 22. NVIC 寄存器列表	145
表 23. EXTI 寄存器列表	149
表 24. ADCDR[15:0] 寄存器中的数据格式	164
表 25. A/D 转换器寄存器列表	165
表 26. 计数方向和编码信号	194
表 27. 比较匹配输出设置	195
表 28. GPTM 寄存器列表	206
表 29. GPTM 内部触发器连接	210
表 30. 比较匹配输出设置	252
表 31. PWM 寄存器列表	261
表 32. PWM 内部触发器连接	265
表 33. BFTM 寄存器列表	289
表 34. LSE 在不同启动模式下的工作电流和启动时间	293
表 35. RTCOUT 输出模式和有效电平设置	294
表 36. RTC 寄存器列表	295
表 37. 看门狗定时器寄存器列表	303
表 38. 保持 SCL 线状态的条件	321
表 39. I ² C 寄存器列表	323

表 40. I ² C 时钟设置范例	330
表 41. SPI 接口格式设置	338
表 42. SPI 模式故障触发条件	343
表 43. SPI 主机模式 SEL 引脚状态	343
表 44. SPI 寄存器列表	344
表 45. 波特率误差计算 – CK_UART = 20 MHz.....	357
表 46. 波特率误差计算 – CK_UART = 10MHz.....	358
表 47. UART 寄存器列表.....	358
表 48. DIV 寄存器列表	367

图列表

图 1. 方框图	22
图 2. Cortex®-M0+ 方框图	25
图 3. 总线结构	26
图 4. 存储器映射	27
图 5. Flash 存储器控制器方框图.....	30
图 6. Flash 存储器映射.....	31
图 7. 向量重映射	32
图 8. 页擦除操作流程图中.....	33
图 9. 整片擦除操作流程图中.....	34
图 10. 字编程操作流程图中.....	35
图 11. PWRCU 方框图	54
图 12. 上电复位 / 掉电复位波形	56
图 13. CKCU 方框图	65
图 14. HSE 外部晶体、陶瓷和谐振器.....	66
图 15. HSI 自动调整方框图.....	68
图 16. LSE 外部晶体、陶瓷和谐振器	69
图 17. RSTCU 方框图	90
图 18. 上电复位时序图	91
图 19. GPIO 方框图	96
图 20. AFIO/GPIO 控制信号.....	98
图 21. AFIO 方框图	136
图 22. EXTI 输入通道选择	137
图 23. EXTI 方框图	146
图 24. EXTI 唤醒事件管理	147
图 25. EXTI 中断去抖功能	148
图 26. ADC 方框图	158
图 27. 单次转换模式	161
图 28. 连续转换模式	161
图 29. 非连续转换模式	163
图 30. GPTM 方框图	179
图 31. 向上计数范例	181
图 32. 向下计数范例	181
图 33. 中心对齐计数范例	182
图 34. GPTM 时钟源选择	183
图 35. 触发控制器方框图	184
图 36. 从机控制器方框图	185
图 37. 重启模式下的 GPTM	185
图 38. 暂停模式下的 GPTM	186
图 39. 触发模式下的 GPTM	186

图 40. 主机 GPTMn 和从机 GPTMm / MCTMm 相连接	187
图 41. MTO 选项	187
图 42. 捕捉 / 比较方框图	188
图 43. 输入捕捉模式	189
图 44. PWM 脉冲宽度测量范例	190
图 45. 通道 0 和通道 1 输入级	191
图 46. 通道 2 和通道 3 输入级	192
图 47. TIO 数字滤波器方框图 (N = 2)	192
图 48. 输入级和正交解码器方框图	193
图 49. TIO 和 TI1 正交解码器计数	194
图 50. 输出级方框图	195
图 51. 翻转模式通道输出参考信号 - CHxPRE = 0	196
图 52. 翻转模式通道输出参考信号 - CHxPRE = 1	196
图 53. PWM 模式通道输出参考信号和计数器向上计数模式	197
图 54. PWM 模式通道输出参考信号和计数器向下计数模式	197
图 55. PWM 模式通道输出参考信号和计数器中心对齐计数模式	198
图 56. 更新事件设置方框图	199
图 57. 单脉冲模式	200
图 58. 立即有效模式的延迟	201
图 59. 非对称 PWM 模式与中心对齐计数模式	202
图 60. 用 GPTM CH0OREF 信号暂停 PWM0	203
图 61. 用 GPTM 更新事件触发 PWM0	204
图 62. 用 GPTM CH0 输入信号触发 GPTM 和 PWM0	205
图 63. PWM 方框图	242
图 64. 向上计数范例	243
图 65. 向下计数范例	244
图 66. 中心对齐计数范例	245
图 67. PWM 时钟源选择	246
图 68. 触发控制器方框图	247
图 69. 从机控制器方框图	248
图 70. 重启模式下的 PWM	248
图 71. 暂停模式下的 PWM	249
图 72. 触发模式下的 PWM	249
图 73. 主机 PWMn 和从机 PWMm / TMm 相连接	250
图 74. MTO 选项	250
图 75. 比较方框图	251
图 76. 输出级方框图	251
图 77. 翻转模式通道输出参考信号 (CHxPRE = 0)	252
图 78. 翻转模式通道输出参考信号 (CHxPRE = 1)	253
图 79. PWM 模式通道输出参考信号和计数器向上计数模式	253
图 80. PWM 模式通道输出参考信号和计数器向下计数模式	254

图 81. PWM 模式通道输出参考信号和计数器中心对齐计数模式.....	254
图 82. 更新事件设置方框图	255
图 83. 单脉冲模式	256
图 84. 立即有效模式的延迟	257
图 85. 非对称 PWM 模式与中心对齐计数模式.....	258
图 86. 用 PWM0 CH0OREF 信号暂停 PWM1	259
图 87. 用 PWM0 更新事件触发 PWM1	259
图 88. 用 PWM0 定时器使能信号触发 PWM0 和 PWM1.....	260
图 89. BFTM 方框图	286
图 90. BFTM – 重复模式	287
图 91. BFTM – 单次模式	288
图 92. BFTM – 单次模式计数器更新	288
图 93. RTC 方框图.....	292
图 94. 看门狗定时器方框图	301
图 95. 看门狗定时器的行为	303
图 96. I ² C 模块方框图	310
图 97. START 和 STOP 条件	312
图 98. 数据有效性	312
图 99. 7-bit 寻址模式.....	313
图 100. 10-bit 寻址写发送模式.....	314
图 101. 10-bit 寻址读接收模式.....	314
图 102. I ² C 总线确认	315
图 103. 仲裁期间时钟同步	315
图 104. 两个主机仲裁程序	316
图 105. 主机发送器时序图	318
图 106. 主机接收器时序图	319
图 107. 从机发送器时序图	320
图 108. 从机接收器时序图	321
图 109. SCL 时序图.....	330
图 110. SPI 方框图.....	336
图 111. SPI 单字节传输时序图 – CPOL = 0, CPHA = 0.....	338
图 112. SPI 连续传输时序图 – CPOL = 0, CPHA = 0.....	338
图 113. SPI 单个字节传输时序图 – CPOL = 0, CPHA = 1.....	339
图 114. SPI 连续传输时序图 – CPOL = 0, CPHA = 1.....	339
图 115. SPI 单个字节传输时序图 – CPOL = 1, CPHA = 0.....	340
图 116. SPI 连续数据传输时序图 – CPOL = 1, CPHA = 0.....	340
图 117. SPI 单个字节传输时序图 – CPOL = 1, CPHA = 1.....	341
图 118. SPI 连续数据传输时序图 – CPOL = 1, CPHA = 1.....	341
图 119. SPI 多主机从机环境.....	343
图 120. UART 方框图.....	355
图 121. UART 串行数据格式.....	356

图 122. UART 时钟 CK_UART 和数据帧时序	357
图 123. 除法器功能图	366

1 简介

概述

本用户手册提供的详细资料,包括如何使用该系列单片机、系统和总线结构、存储器结构和外设说明。本文档的目标读者是软件开发人员,应用开发商和硬件开发人员。欲了解更多有关引脚配置、封装和电气特性的信息,请参考系列数据手册 (Datasheet)。

该系列 Holtek 单片机是一款基于 Arm® Cortex®-M0+ 处理器内核的 32 位高性能低功耗单片机。Cortex®-M0+ 是把嵌套向量中断控制器 (NVIC), SysTick 定时器和先进的调试支持紧紧结合在一起的新一代处理器内核。

该系列单片机可工作在高达 20 MHz 的频率下,通过借助 Flash 加速器以获得较大的效能。它提供多达 32 KB 的嵌入式 Flash 存储器用作代码 / 数据存储,4 KB 的嵌入式 SRAM 存储器用作系统操作和应用程序运用。此系列单片机具有多种外设,如硬件除法器 DIV、ADC、I²C、UART、SPI、BFTM、GPTM、PWM、RTC、WDT 和 SW-DP (串行线调试端口) 等。此外该系列单片机还提供了几种省电模式,在唤醒延迟和功耗方面具有较为优化的灵活性,这是低功耗应用方面的考虑要点。

以上这些特性使该系列单片机可以广泛地适用于各种应用,如白色家电应用控制、电源监控和报警系统、消费类产品、手持式设备、数据记录应用、马达控制等。

特性

- 内核
 - 32-bit Arm® Cortex®-M0+ 处理器内核
 - 工作频率高达 20 MHz
 - 单周期乘法
 - 集成嵌套向量中断控制器 (NVIC)
 - 24-bit SysTick 定时器
- 片上存储器
 - 高达 32 KB 片上 Flash 存储器用作指令 / 数据和选项字节存储
 - 4 KB 片上 SRAM
 - 支持多种启动模式
- Flash 存储器控制器 – FMC
 - 32-bit 字编程带有在线系统烧录 (ISP) 和在线应用编程 (IAP)
 - Flash 保护能力, 防止非法访问
- 复位控制单元 – RSTCU
 - 电源监控: 上电复位 (POR)、掉电复位 (PDR)、欠压检测器 (BOD)、可编程低压检测器 (LVD)
- 时钟控制单元 – CKCU
 - 外部 4 ~ 20 MHz 晶振
 - 外部 32.768 kHz 晶振
 - 在工作温度为 25 °C 下, 内部 20 MHz RC 振荡器精度可调整为 ±2 %
 - 内部 32 kHz RC 振荡器
 - 用作外设时钟源的独立的时钟分频和门控位
- 电源管理 – PWRCU
 - 灵活供电:
 - V_{DD} 供电: 2.5 V ~ 5.5 V
 - V_{DDIO} 用作 I/O 口供电: 1.8 V ~ 5.5 V
 - 内置 1.5 V LDO 稳压器用作 CPU 内核、外设和存储器电源
 - 三个电源域: V_{DD}、V_{DDIO} 和 1.5 V
 - 三种省电模式: 休眠模式、深度休眠模式 1、深度休眠模式 2
- 外部中断 / 事件控制器 – EXTI
 - 高达 16 条 EXTI 线, 可配置触发源与触发类型
 - 所有 GPIO 引脚都可选作 EXTI 触发源
 - 触发源类型包括: 高电平、低电平、下降沿、上升沿或者双沿
 - 每条 EXTI 线都具有独立的中断使能、唤醒使能和状态标志位
 - 每条 EXTI 线都具有软件中断触发模式
 - 内置去干扰滤波器, 用于封锁短脉冲
- 模拟数字转换器 – ADC
 - 12-bit SAR ADC 内核
 - 最大 1 Msps 转换速率
 - 最多 12 个外部模拟输入通道

- 输入 / 输出 – GPIO
 - 多达 40 个通用输入 / 输出
 - 端口 A、B 和 C 映射为 16 个外部中断 – EXTI
 - 几乎所有的 I/O 引脚都具有可配置输出驱动电流
- PWM 产生和捕捉定时器 – GPTM
 - 一个 16-bit 向上、向下、向上 / 向下自动重载计数器
 - 16-bit 可编程预分频器，可以对计数器时钟进行 1 ~ 65536 之间的任何数值的分频
 - 输入捕捉功能
 - 比较匹配输出
 - 带有边沿对齐模式和中心对齐计数模式的 PWM 波形发生器
 - 单脉冲模式输出
 - 内建可处理编码器接口信号的带两个输入口的正交解码器
- 脉冲宽度调制器 – PWM
 - 1 个 16-bit 向上、向下、向上 / 向下自动重载计数器
 - 每个定时器具有多达 4 个独立通道
 - 16-bit 可编程预分频器，可以对计数器时钟进行 1 ~ 65536 之间的任何数值的分频
 - 比较匹配输出
 - PWM 波形产生功能，具有边沿对齐和中心对齐两种计数模式
 - 单脉冲模式输出
- 基础功能定时器 – BFTM
 - 1 个 32-bit 比较 / 匹配向上计数器 – 无输入 / 输出引脚
 - 单次模式 – 匹配后停止计数
 - 重复模式 – 匹配后重新开始计数
- 看门狗定时器 – WDT
 - 带 3-bit 预分频器的 12-bit 向下计数器
 - 可产生系统复位
 - 可编程看门狗定时器窗口功能
 - 寄存器写保护功能
- 实时时钟 – RTC
 - 带可编程预分频器的 24-bit 向上计数器
 - 闹钟功能
 - 中断和唤醒事件
- 内部集成电路 – I²C
 - 支持高达 1 MHz 频率的主从模式
 - 提供仲裁功能和时钟同步
 - 支持 7-bit 和 10-bit 寻址模式和广播呼叫寻址
 - 屏蔽地址功能可支持从机多寻址模式
- 串行外设接口 – SPI
 - 支持主机和从机模式
 - 主机模式频率高达 ($f_{PCLK} / 2$) MHz，从机模式频率高达 ($f_{PCLK} / 3$) MHz
 - FIFO 深度：8 级
 - 多个主机和多个从机操作

- 通用异步收发器 – UART
 - 异步串行通信工作波特率时钟频率高达 ($f_{PCLK} / 16$) MHz
 - 全双工通信能力
 - 完全可编程串行通信特性包括：字长、校验位、停止位和位顺序
 - 错误检测：奇偶校验、溢出和帧错误
- 硬件除法器 – DIV
 - 32-bit 有符号 / 无符号除法器
 - 运算需要 8 个时钟周期，加载需要 1 个时钟周期
 - 除数为零错误标志
- 调试支持
 - 串行线调试端口 – SW-DP
 - 4 个用于硬件断点或代码 / 文字补丁的比较器
 - 2 个用于硬件数据观察点的比较器
- 封装和工作温度
 - 24/28-pin SSOP, 28-pin SOP, 24/33-pin QFN 和 44/48-pin LQFP 封装类型
 - 工作温度范围：-40 °C ~ +85 °C

单片机信息

表 1. 特性及外设列表

外设		HT32F50220	HT32F50230
主 Flash (KB)		16	31
选项字节 Flash (KB)		1	1
SRAM (KB)		4	4
定时器	GPTM	1	
	PWM	2	
	BFTM	1	
	WDT	1	
	RTC	1	
通信	SPI	2	
	UART	2	
	I ² C	1	
硬件除法器		1	
EXTI		16	
12-bit ADC 通道总数		1 12 通道	
GPIO		40 (Max.)	
CPU 频率		20 MHz (Max.)	
工作电压		2.5 V ~ 5.5 V	
工作温度		-40 °C ~ +85 °C	
封装		24/28-pin SSOP, 28-pin SOP, 24/33-pin QFN 和 44/48-pin LQFP	

方框图

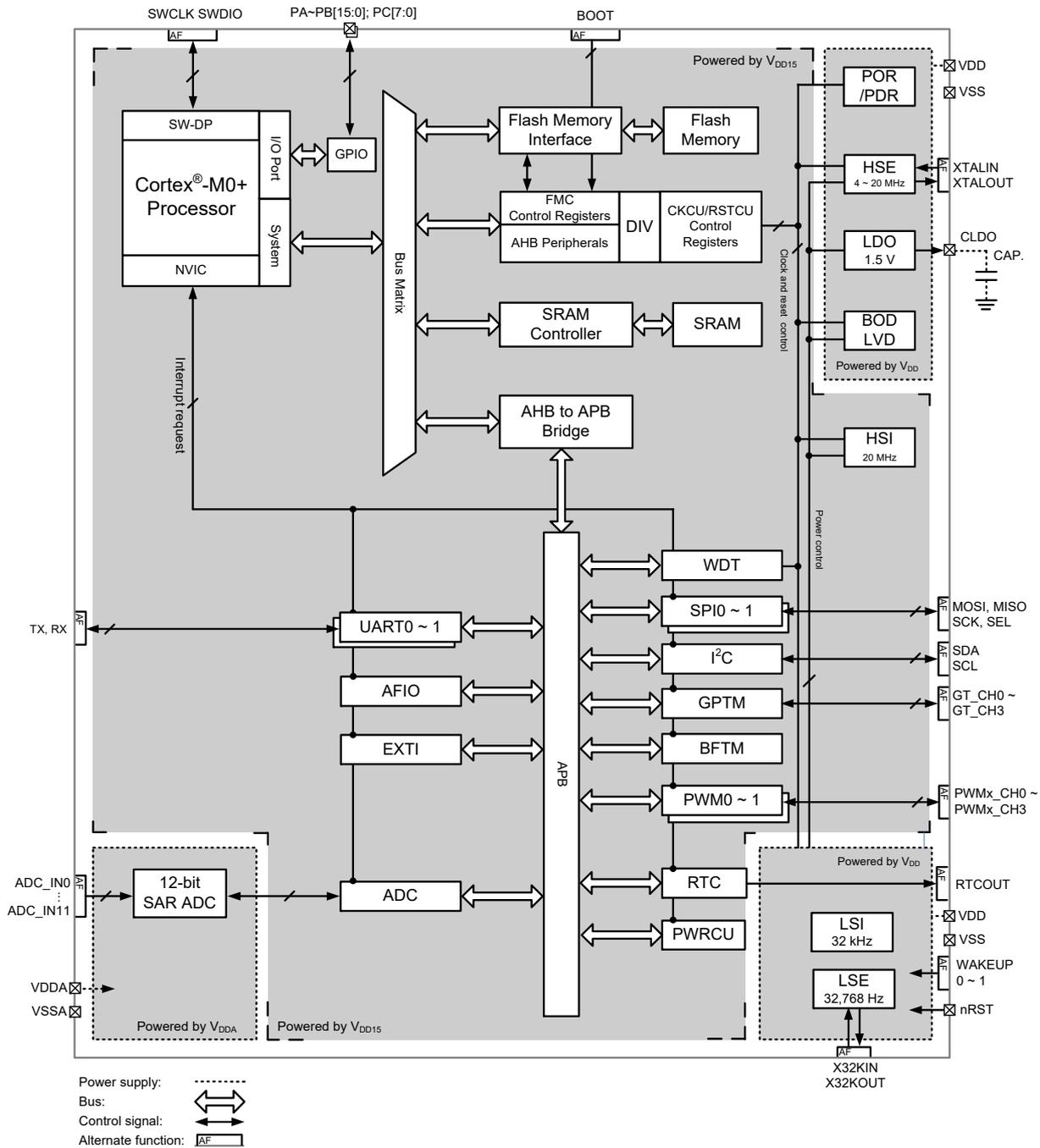


图 1. 方框图

一
简介

2 文档协议

本文档中使用的协议列于下表。

表 2. 文档协议

符号	范例	描述						
0x	0x5a05	带 0x 前缀的数字字符串表示十六进制数。						
0xnnnn_nnnn	0x2000_0100	32-bit 十六进制地址或数据。						
b	b0101	带小写字母 b 前缀的数字字符串表示二进制数。						
NAME[n]	ADDR[5]	NAME 的特定位。NAME 可以是寄存器或寄存器字段。例如，ADDR[5] 指的是 ADDR 寄存器 (字段) 的第 5 位。						
NAME[m:n]	ADDR[11:5]	NAME 的特定位。NAME 可以是寄存器或寄存器字段。例如，ADDR[11:5] 指的是 ADDR 寄存器 (字段) 的第 11 位 ~ 第 5 位。						
X	b10X1	不用在意的符号，它表示可以允许的任何值。						
RW	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="text-align: center;">19</td> <td style="text-align: center;">18</td> </tr> <tr> <td style="text-align: center;">SERDYIE</td> <td style="text-align: center;">PLLRDYIE</td> </tr> <tr> <td style="text-align: center;">RW</td> <td style="text-align: center;">0 RW 0</td> </tr> </table>	19	18	SERDYIE	PLLRDYIE	RW	0 RW 0	软件可以读取或写入该位。
19	18							
SERDYIE	PLLRDYIE							
RW	0 RW 0							
RO	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="text-align: center;">3</td> <td style="text-align: center;">2</td> </tr> <tr> <td style="text-align: center;">HSIRDY</td> <td style="text-align: center;">HSERDY</td> </tr> <tr> <td style="text-align: center;">RO</td> <td style="text-align: center;">1 RO 0</td> </tr> </table>	3	2	HSIRDY	HSERDY	RO	1 RO 0	软件只能读该位。写操作将不会有任何效果。
3	2							
HSIRDY	HSERDY							
RO	1 RO 0							
RC	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> </tr> <tr> <td style="text-align: center;">PDF</td> <td style="text-align: center;">BAK PORF</td> </tr> <tr> <td style="text-align: center;">RC</td> <td style="text-align: center;">0 RC 1</td> </tr> </table>	1	0	PDF	BAK PORF	RC	0 RC 1	软件只能读该位。读操作将自动清零该位。
1	0							
PDF	BAK PORF							
RC	0 RC 1							
WC	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="text-align: center;">3</td> <td style="text-align: center;">2</td> </tr> <tr> <td style="text-align: center;">SERDYF</td> <td style="text-align: center;">PLLRDYF</td> </tr> <tr> <td style="text-align: center;">WC</td> <td style="text-align: center;">0 WC 0</td> </tr> </table>	3	2	SERDYF	PLLRDYF	WC	0 WC 0	软件可以读该位或通过写 1 来清除它。写 0 将不会有任何效果。
3	2							
SERDYF	PLLRDYF							
WC	0 WC 0							
W0C	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> </tr> <tr> <td style="text-align: center;">RXCF</td> <td style="text-align: center;">PARF</td> </tr> <tr> <td style="text-align: center;">RO</td> <td style="text-align: center;">0 RO 0</td> </tr> </table>	1	0	RXCF	PARF	RO	0 RO 0	软件可以读该位或通过写 0 来清除它。写 1 将不会有任何效果。
1	0							
RXCF	PARF							
RO	0 RO 0							
WO	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="text-align: center;">31</td> <td style="text-align: center;">30</td> </tr> <tr> <td colspan="2" style="text-align: center;">DB_CKSRC</td> </tr> <tr> <td style="text-align: center;">WO</td> <td style="text-align: center;">0 WO 0</td> </tr> </table>	31	30	DB_CKSRC		WO	0 WO 0	软件只能写该位。读操作将总是返回 0。
31	30							
DB_CKSRC								
WO	0 WO 0							
保留位	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> </tr> <tr> <td style="text-align: center;">LLRDY</td> <td style="text-align: center;">Reserved</td> </tr> <tr> <td style="text-align: center;">RO</td> <td style="text-align: center;">0</td> </tr> </table>	1	0	LLRDY	Reserved	RO	0	保留位供将来使用。从这些位中读取的数据没有很好地被定义，应被视为随机数据处理。通常情况下，这些保留位应设置为 0。需要注意的是保留位必须保持在复位值。
1	0							
LLRDY	Reserved							
RO	0							
Word		一个字的数据长度为 32-bit。						
Half-word		一个半字的数据长度为 16-bit。						
Byte		一个字节的数据长度为 8-bit。						

3 系统结构

该系列单片机的系统结构包括 Arm® Cortex®-M0+ 处理器、总线结构和存储器体系，将在以下各节中描述。Cortex®-M0+ 处理器是新一代的处理器内核，提供了许多新的特性。集成和先进的特性使得 Cortex®-M0+ 处理器适合于需要高性能和低功耗单片机的市场产品。简而言之，Cortex®-M0+ 处理器包含 AHB-Lite 总线接口。根据不同的目的和目标存储空间，Cortex®-M0+ 处理器访问所有存储器都是在这条总线上执行的。存储器体系采用了哈佛结构，预先定义的存储器映射和高达 4 GB 的存储空间，使系统灵活和可扩展。

Arm® Cortex®-M0+ 处理器

Cortex®-M0+ 处理器具有低门数、高效能的特点，专为要求面积优化、低功耗处理器的单片机及深度嵌入式应用而设计。Cortex®-M0+ 处理器基于 ARMv6-M 架构，同时支持 Thumb® 指令集、单周期输入 / 输出端口、硬件乘法器和低延迟中断响应时间。下面列出了一些也是由 Cortex®-M0+ 提供的系统外设：

- 内部总线矩阵连接 AHB-Lite 接口，单周期输入 / 输出端口和调试访问端口 (DAP)
- 嵌套向量中断控制器 (NVIC)
- 可选唤醒中断控制器 (WIC)
- 断点和观察点单元
- 可选存储器保护单元 (MPU)
- 串行线调试端口 (SW-DP)
- 可选微型追踪缓存接口 (MTB)

下图为 Cortex®-M0+ 处理器的方框图。欲了解更多信息，请参考 Arm® Cortex®-M0+ 技术参考手册。

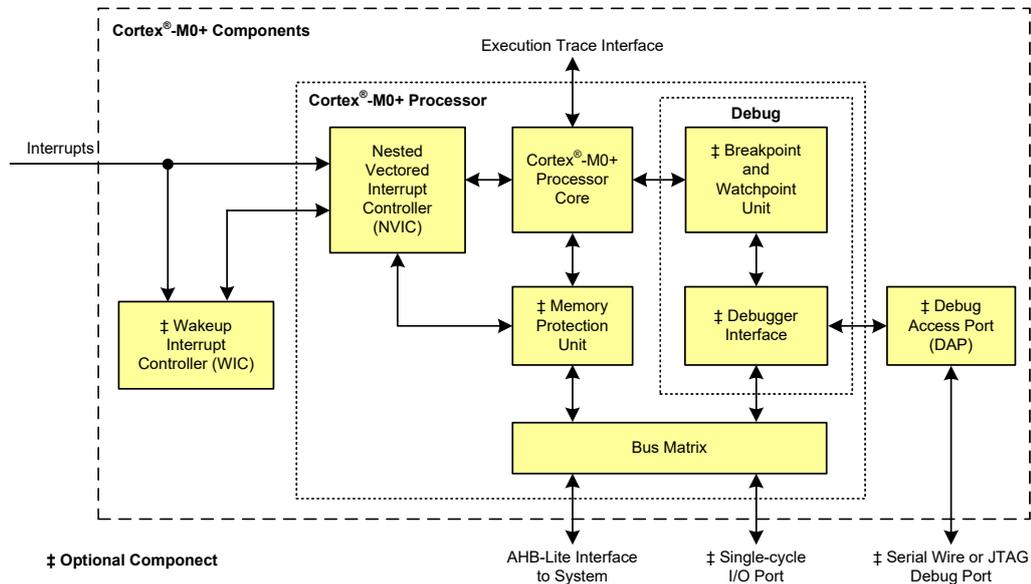


图 2. Cortex®-M0+ 方框图

总线结构

该系列单片机的总线结构包括 1 条主机总线和 4 条从机总线。Cortex®-M0+ AHB-Lite 总线为主机总线，而内部 SRAM 访问总线、内部 Flash 存储器访问总线、AHB 外设访问总线以及 AHB-APB 桥为从机总线。32-bit AHB-Lite 总线用于对所有系统区域提供简单的集成，包括内部 SRAM 域和外设域。所有的主机总线都基于 32-bit 高级的高性能总线精简 (AHB-Lite) 协议。下图显示了该系列单片机的总线结构。

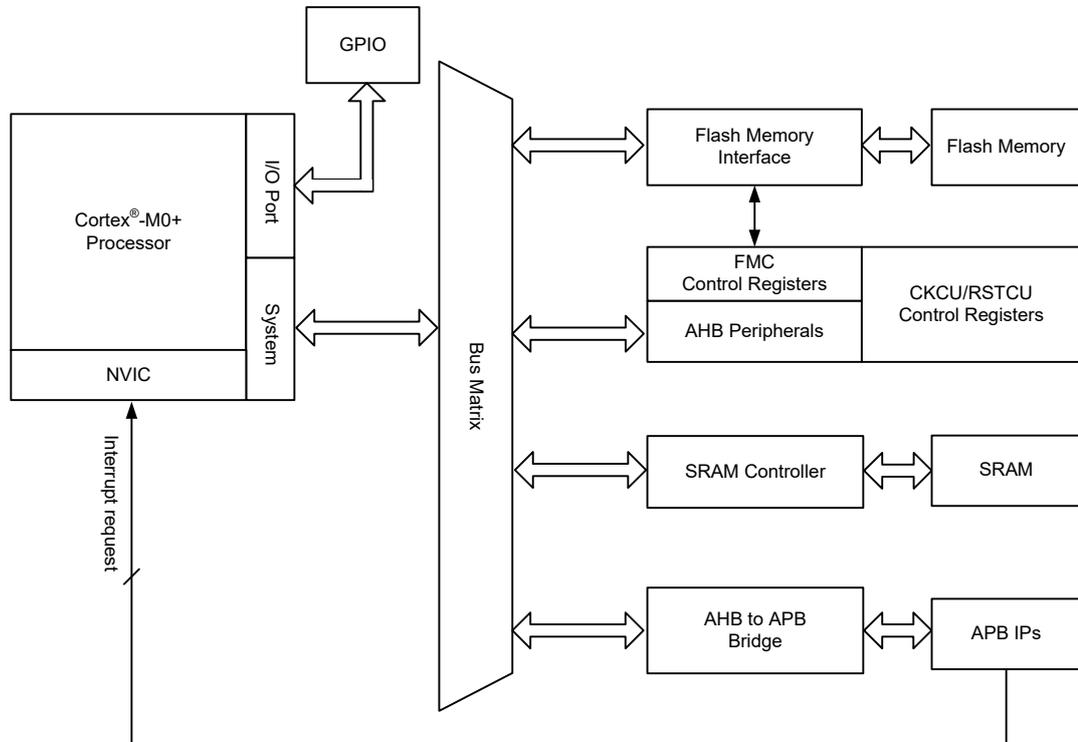


图 3. 总线结构

存储器体系

Arm® Cortex®-M0+ 处理器通过同一个外部接口对外部 AHB 外设进行访问及调试访问。处理器访问优先于调试访问。Cortex®-M0+ 的最大地址范围是 4 GB，因为它具有 32-bit 总线地址宽度。此外，预先定义的存储器映射由 Cortex®-M0+ 处理器提供，以减少软件被不同的单片机供应商重复实施的复杂性。但是，有些区域由 Arm® Cortex®-M0+ 系统外设使用。更多信息请参考 Arm® Cortex®-M0+ 技术参考手册。下图显示了该系列单片机的存储器映射，包括代码、SRAM、外设和其它预定义区域。

存储器映射

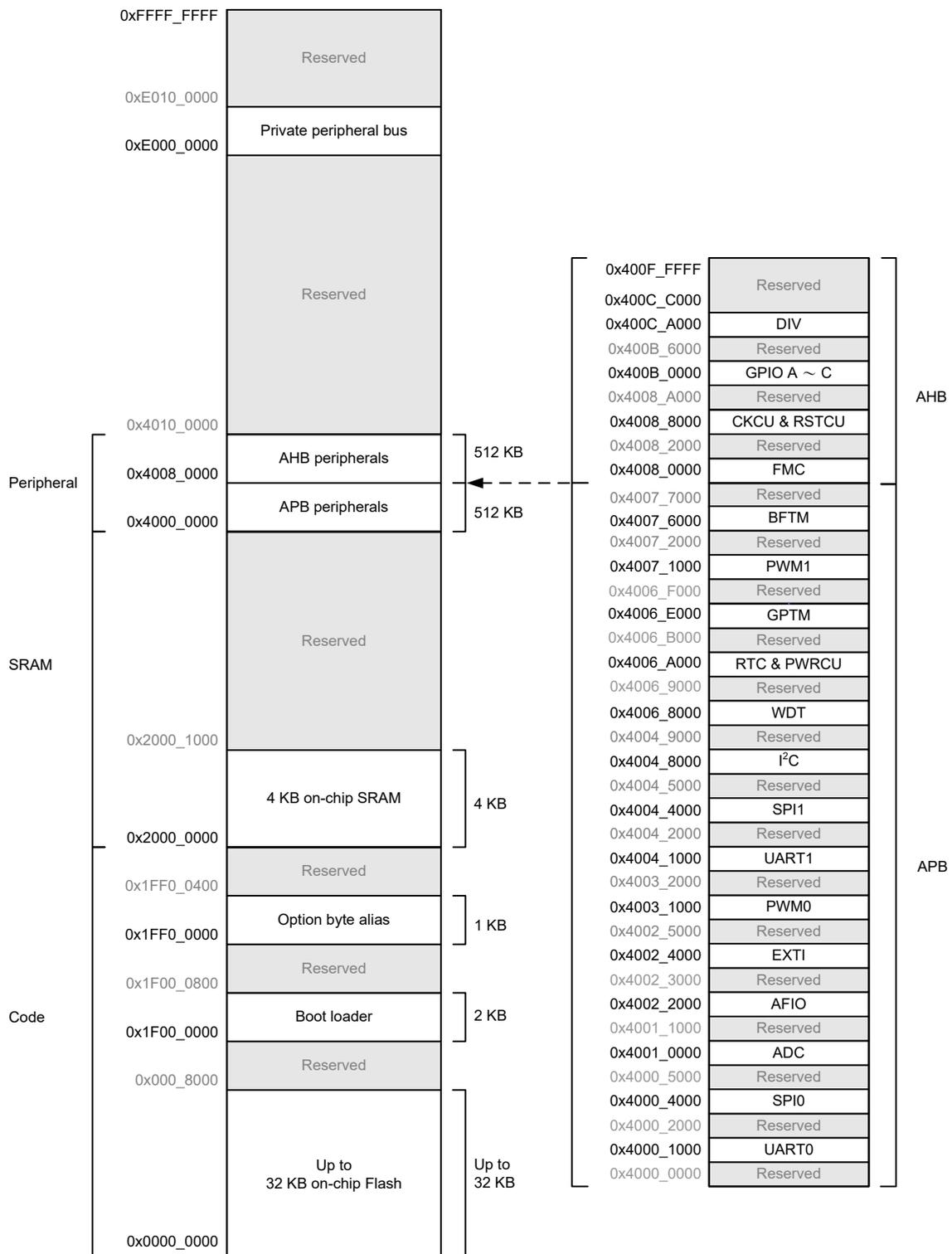


图 4. 存储器映射

表 3. 寄存器映射

起始地址	终止地址	外设	总线	
0x4000_0000	0x4000_0FFF	保留	APB	
0x4000_1000	0x4000_1FFF	UART0		
0x4000_2000	0x4000_3FFF	保留		
0x4000_4000	0x4000_4FFF	SPI0		
0x4000_5000	0x4000_5FFF	保留		
0x4001_0000	0x4001_0FFF	ADC		
0x4001_1000	0x4002_1FFF	保留		
0x4002_2000	0x4002_2FFF	AFIO		
0x4002_3000	0x4002_3FFF	保留		
0x4002_4000	0x4002_4FFF	EXTI		
0x4002_5000	0x4003_0FFF	保留		
0x4003_1000	0x4003_1FFF	PWM0		
0x4003_2000	0x4004_0FFF	保留		
0x4004_1000	0x4004_1FFF	UART1		
0x4004_2000	0x4004_3FFF	保留		
0x4004_4000	0x4004_4FFF	SPI1		
0x4004_5000	0x4004_7FFF	保留		
0x4004_8000	0x4004_8FFF	I ² C		
0x4004_9000	0x4006_7FFF	保留		
0x4006_8000	0x4006_8FFF	WDT		
0x4006_9000	0x4006_9FFF	保留		
0x4006_A000	0x4006_AFFF	RTC & PWRCU		
0x4006_B000	0x4006_DFFF	保留		
0x4006_E000	0x4006_EFFF	GPTM		
0x4006_F000	0x4007_0FFF	保留		
0x4007_1000	0x4007_1FFF	PWM1		
0x4007_2000	0x4007_5FFF	保留		
0x4007_6000	0x4007_6FFF	BFTM		
0x4007_7000	0x4007_FFFF	保留		
0x4008_0000	0x4008_1FFF	FMC		AHB
0x4008_2000	0x4008_7FFF	保留		
0x4008_8000	0x4008_9FFF	CKCU & RSTCU		
0x4008_A000	0x400A_FFFF	保留		
0x400B_0000	0x400B_1FFF	GPIOA		
0x400B_2000	0x400B_3FFF	GPIOB		
0x400B_4000	0x400B_5FFF	GPIOC		
0x400B_6000	0x400C_9FFF	保留		
0x400C_A000	0x400C_BFFF	DIV		
0x400C_C000	0x400F_FFFF	保留		

嵌入式 Flash 存储器

该系列单片机提供高达 32 KB 片上 Flash 存储器，位于地址 0x0000_0000。它支持字节、半字和字访问操作。注意，Flash 存储器仅支持读操作。通过总线对 Flash 存储器的任何写操作将导致总线错误中断发生。该 Flash 存储器容量高达 32 页，每一页都有 1 KB 的存储容量，并且可以单独擦除。32-bit 可编程接口可对单独的位进行编程。数据存储或固件升级可以通过使用在线系统烧录 (ISP)、在线应用编程 (IAP) 或在线烧录 (ICP) 来实现。欲了解更多信息，请参考 Flash 存储器控制器章节。

嵌入式 SRAM 存储器

该系列单片机包含 4 KB 片上 SRAM 存储器，位于地址 0x2000_0000。它支持字节、半字和字访问操作。

AHB 外设

AHB 外设的地址范围从 0x4008_0000 到 0x400F_FFFF。一些外设如时钟控制单元、复位控制单元和 Flash 存储器控制器直接连接到 AHB 总线。系统复位后，上列的 AHB 外设时钟总是开启的。可以通过 AHB 总线直接访问这些外设寄存器。注意，所有 AHB 总线上的外设寄存器只支持字访问。

APB 外设

APB 外设的地址范围从 0x4000_0000 到 0x4007_FFFF。通过 APB-AHB 桥可进行 CPU 和 APB 外设间的访问。此外，系统复位后，APB 外设时钟关闭。在访问相应的外设寄存器之前，软件必须通过设置时钟控制单元中的 APBCCRn 寄存器打开外设时钟。需要注意的是，当对 APB 外设寄存器进行半字或字节访问时，APB-AHB 总线桥会将半字或字节数据复制到一个字的宽度。换言之，对 APB 外设寄存器进行半字或字节访问的结果会因访问数据位宽度而有所不同。

4 Flash 存储器控制器 (FMC)

简介

Flash 存储器控制器 FMC 为嵌入式片上 Flash 存储器提供所有必要的功能和预取缓存器。下图显示了 FMC 的方框图，其中包含了烧录接口，控制寄存器，预取缓存器和访问接口。由于 Flash 存储器访问速度比 CPU 慢，故提供一个带有预取缓存器的宽访问接口以减少 CPU 等待时间，从而避免引起指令执行延迟。Flash 存储器还提供字编程 / 页擦除功能来实现指令与数据的储存。

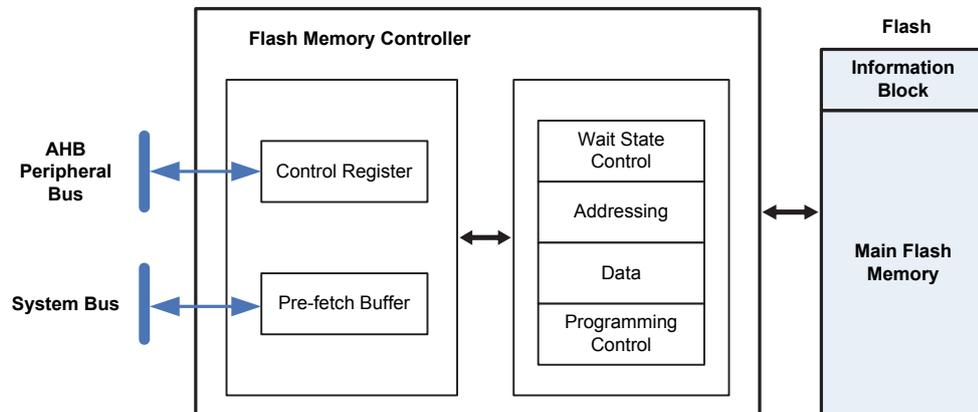


图 5. Flash 存储器控制器方框图

特性

- 多达 32 KB 片上 Flash 存储器用于存储指令 / 数据和选项字节
 - 32 KB (指令 / 数据 + 选项字节)
 - 16 KB (指令 / 数据 + 选项字节)
- 页大小为 1 KB，页数取决于主 Flash 的大小，总共多达 32 页
- 带有预取缓存器的宽访问接口，以减少指令间隙
- 页擦除和整片擦除
- 32-bit 字编程
- 中断功能用来指示 Flash 存储器操作完成或有错误发生
- Flash 读保护，防止非法代码 / 数据访问
- 页擦除 / 编程保护，防止操作不慎毁坏数据

功能描述

Flash 存储器映射

下图是该系列单片机地址从 0x0000_0000 到 0x1FFF_FFFF (0.5 GB) 的 Flash 存储器映射。0x1F00_0000 ~ 0x1F00_07FF 的地址被映射到 2 KB 容量的启动加载程序块。此外，从 0x1FF0_0000 到 0x1FF0_03FF 的地址区域是具有 1 KB 容量的选项字节块，位于主 Flash 存储块物理地址最后一页。在系统视图下的存储器映射如下图所示。

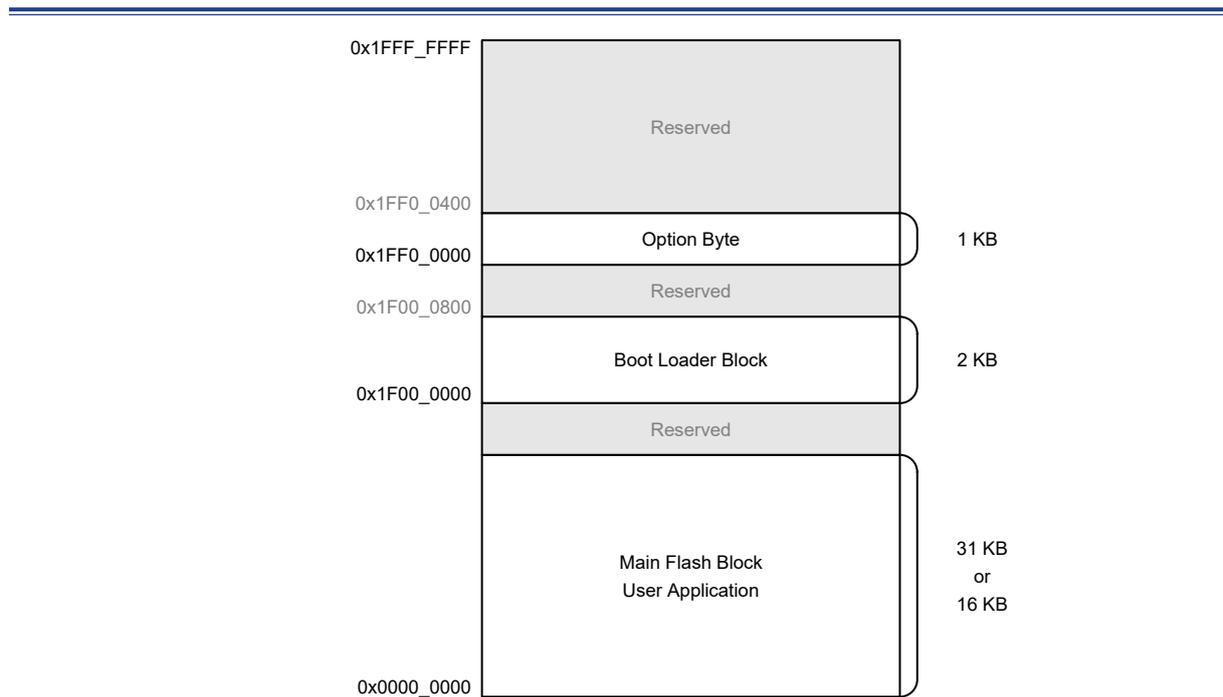


图 6. Flash 存储器映射

Flash 存储器结构

Flash 存储器由高达 32 KB 主 Flash 存储块 (每页 1 KB) 和用于启动加载器的 2 KB 信息块组成。主 Flash 存储块总共包含 32 页 (16 KB 的单片机总共有 16 页) 可单独擦除的页。下表显示了每页的基址、大小和保护设置。

表 4. Flash 存储器和选项字节

存储块	名称	地址	页保护位	大小
主 Flash 块	Page 0	0x0000_0000 ~ 0x0000_03FF	OB_PP [0]	1 KB
	Page 1	0x0000_0400 ~ 0x0000_07FF	OB_PP [1]	1 KB
	Page 2	0x0000_0800 ~ 0x0000_0BFF	OB_PP [2]	1 KB
	Page 3	0x0000_0C00 ~ 0x0000_0FFF	OB_PP [3]	1 KB
	⋮	⋮	⋮	⋮
	Page 28	0x0000_7000 ~ 0x0000_73FF	OB_PP [28]	1 KB
	Page 29	0x0000_7400 ~ 0x0000_77FF	OB_PP [29]	1 KB
	Page 30	0x0000_7800 ~ 0x0000_7BFF	OB_PP [30]	1 KB
	Page 31 (选项字节)	物理地址: 0x0000_7C00 ~ 0x0000_7FFF 别名地址: 0x1FF0_0000 ~ 0x1FF0_03FF	OB_CP [1]	1 KB
信息块	Boot Loader (启动加载器)	0x1F00_0000 ~ 0x1F00_07FF	NA	2 KB

注: 1. 该信息块存储启动加载器 – 此块不能由用户编程或擦除。
2. 选项字节始终位于主 Flash 块的最后一页。

启动配置

系统提供了两种启动模式, 通过 BOOT 引脚进行选择。在上电复位或系统复位期间对 BOOT 引脚采样。一旦该引脚上的逻辑值确定, 根据启动模式, 向量的前 4 个字将被重映射到相应的源。下表列出了启动模式。

表 5. 启动模式

启动模式选择引脚	模式	描述
BOOT		
0	启动加载器	启动加载器被选为启动区域
1	主 Flash	主 Flash 存储器被选为启动区域

Flash 向量映射控制寄存器 VMCR 用来在单片机复位后临时改变向量重映射设置。VMCR 寄存器的复位初始值是由将在复位期间被采样的 BOOT 引脚的状态定义的。

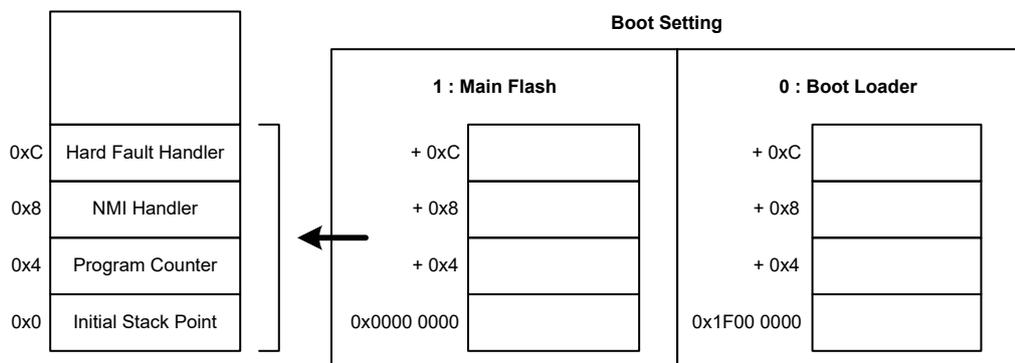


图 7. 向量重映射

页擦除

FMC 提供了页擦除功能, 用来复位 Flash 存储器的部分内容。每一页可以单独擦除而不影响其它页的内容。下列步骤显示了用于页擦除的寄存器的访问顺序。

- 检查 OPCR 寄存器, 以确认没有正在执行 Flash 存储器操作 (OPM[3:0] = 0xE 或 0x6)。否则, 需一直等待直到当前操作完成。
- 在 TADR 寄存器写入页地址。
- 在 OCMR 寄存器写入页擦除命令 (CMD[3:0] = 0x8)。
- 通过设置 OPCR 寄存器, 发送页擦除命令到 FMC (设置 OPM[3:0] = 0xA)。
- 等待, 直到通过检查 OPCR 寄存器的值确认所有操作已经完成 (OPM[3:0] = 0xE)。
- 依据需要读取并验证 Flash 存储器页的内容。

需要注意的是, 必须确定正确的目标页地址。如果目标擦除页正进行程序代码读取或数据读取, 软件可能会运行失控。如果发生这种情况, FMC 将不提供任何通知。此外, 对于受保护的页, 页擦除操作将被忽略。此时, OREF 位将被置高, 若是 OIER 寄存器中的 OREIEN 位被置位, FMC 将触发 Flash 操作错误中断。软件可以通过在中断处理程序中检查 OISR 寄存器中的 PPEF 位来检测此情况。下图显示了页擦除操作流程。

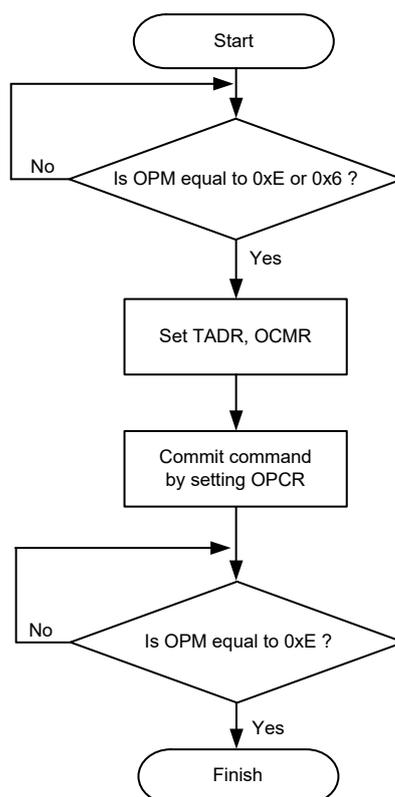


图 8. 页擦除操作流程

整片擦除

FMC 提供了一个整片擦除功能, 用来初始化整个 Flash 存储器的内容到一个高电平状态。下列步骤显示了整片擦除寄存器的访问顺序。

- 检查 OPCR 寄存器, 以确认没有正在执行 Flash 存储器操作 (OPM[3:0] = 0xE 或 0x6)。否则, 需一直等待直到 Flash 存储器操作完成。
- 在 OCMR 寄存器写入整片擦除命令 (CMD[3:0] = 0xA)。
- 通过设置 OPCR 寄存器, 发送整片擦除命令到 FMC (设置 OPM[3:0] = 0xA)。
- 等待, 直到通过检查 OPCR 寄存器的值确认所有操作已经完成 (OPM[3:0] = 0xE)。
- 依据需要读取并验证 Flash 存储器的内容。

由于所有的 Flash 数据将被复位到 0xFFFF_FFFF 值, 使用在 SRAM 上运行的程序或使用调试工具直接访问 FMC 寄存器, 可以实现整片擦除操作。在 Flash 存储器上所执行的程序无法达成整片擦除操作。下图显示了整片擦除操作流程。

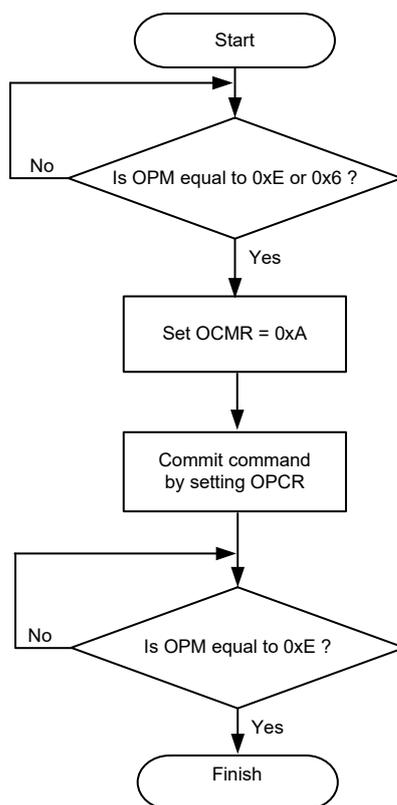


图 9. 整片擦除操作流程

字编程

FMC 提供了 32-bit 字编程功能, 用于修改指定 Flash 存储器的内容。下列步骤显示了字编程操作寄存器的访问顺序。

- 检查 OPCR 寄存器, 以确认没有正在执行 Flash 存储器操作 (OPM[3:0] = 0xE 或 0x6)。否则, 需一直等待直到当前的操作完成。
- 在 TADR 寄存器写入字地址。在 WRDR 寄存器写入字数据。
- 在 OCMR 寄存器写入字编程命令 (CMD[3:0] = 0x4)。
- 通过设置 OPCR 寄存器, 发送字编程命令到 FMC (设置 OPM[3:0] = 0xA)。
- 等待, 直到通过检查 OPCR 寄存器的值确认所有操作已经完成 (OPM[3:0] = 0xE)。
- 依据需要读取并验证 Flash 存储器的内容。

需要注意的是, 不能对相同的地址连续两次进行字编程操作。对相同地址连续进行的字编程操作必须通过页擦除操作分开。此外, 对于受保护的页, 字编程操作将被忽略。此时, OREF 位将被置高, 若是 OIER 寄存器中的 OREIEN 位被置位, FMC 将触发 Flash 操作错误中断。软件可以通过在中断处理程序中检查 OISR 寄存器中的 PPEF 位来检测此情况。下图显示了字编程操作流程。

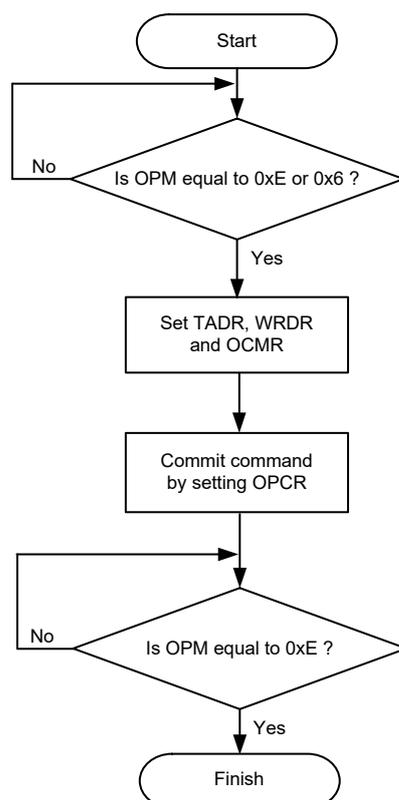


图 10. 字编程操作流程图

选项字节描述

选项字节区可视为一个独立的基址为 0x1FF0_0000 的 Flash 存储器。下表显示了功能描述和选项字节的存储器映射。

表 6. 选项字节的存储器映射

选项字节	偏移量	描述	复位值
选项字节基址 = 0x1FF0_0000			
OB_PP	0x000	Flash 页擦除 / 编程保护 (n = 0 ~ 127)	0xFFFF_FFFF 0xFFFF_FFFF 0xFFFF_FFFF 0xFFFF_FFFF
	0x004	OB_PP [n] (n = 0 ~ 30)	
	0x008	0: Flash 页 n 擦除 / 编程保护使能	
	0x00C	1: Flash 页 n 擦除 / 编程保护除能 OB_PP [n] (n = 31 ~ 127) 保留	
OB_CP	0x010	OB_CP [0]: Flash 安全保护	0xFFFF_FFFF
		0: Flash 安全保护使能	
		1: Flash 安全保护除能	
		OB_CP [1]: 选项字节保护	
OB_CK	0x020	0: 选项字节保护使能	0xFFFF_FFFF
		1: 选项字节保护除能	
		OB_CP [31:2]: 保留位	
		OB_CK [31:0]: 选项字节校验和 当 OB_PP 或 OB_CP 寄存器的内容不等于 0xFFFF_FFFF 时, OB_CK 应设置为 OB_PP 及 OB_CP 寄存器 5 个字的选项字节内容的总和, 其地址偏移量范围从 0x000 到 0x010 (0x000 + 0x004 + 0x008 + 0x00C + 0x010)。否则页擦除 / 编程保护和安全保护都将被使能。	

页擦除 / 编程保护

FMC 提供了页擦除 / 编程保护功能, 以防止 Flash 存储器上的误操作。FMC 受保护页不接受页擦除 (OCMR 寄存器中的 CMD [3:0] = 0x8) 或字编程 (CMD [3:0] = 0x4) 命令。如果页擦除或字编程命令被发送到受保护页上的 FMC, OISR 寄存器中的 PPEF 位将被 FMC 置位。如果 OIER 寄存器的 OREIEN 位也被置位, 那么 FMC 将触发 Flash 操作错误中断。通过配置选项字节中的 OB_PP 寄存器, 可以单独地为每一页使能保护功能。下表显示了页保护使能时主 Flash 页的访问权限。

表 7. 受保护主 Flash 页的访问权限

操作 \ 模式	ISP / IAP	ICP / 调试模式
读	O	O
编程	X	X
页擦除	X	X
整片擦除	O	O

- 注：1. 写保护设置是基于特定页的。上述访问权限只影响保护功能已使能的页。其它页不会受到影响。
 2. 主 Flash 页保护由 OB_PP [127:0] 字段配置。选项字节物理上存在于主 Flash 的最后一页。选项字节的页保护由 OB_CP [1] 位配置。
 3. 选项字节区域上的页擦除可除能主 Flash 的页保护功能。
 4. 选项字节的页保护只能由一个整片擦除操作除能。

下列步骤显示了页擦除 / 编程保护程序寄存器的访问顺序：

- 检查 OPCR 寄存器，以确认没有正在执行 Flash 存储器操作 (OPM [3:0] = 0xE 或 0x6)。否则，需等待直到前面的操作已经完成。
- 在 TADR 寄存器写入 OB_PP 地址 (TADR = 0x1FF0_0000)。
- 将数据写到 WRDR 寄存器，定义相应的页保护功能是使能还是除能 (0: 使能, 1: 除能)。
- 在 OCMR 寄存器写入字编程命令 (CMD [3:0] = 0x4)。
- 通过设置 OPCR 寄存器，将字编程命令发送到 FMC (OPM [3:0] = 0xA)。
- 等待，直到通过检查 OPCR 寄存器的值确认所有操作已经完成 (OPM [3:0] = 0xE)。
- 依据需要读取并验证选项字节。
- 必须根据选项字节校验和规则更新 OB_CK。
- 使系统复位以激活新的 OB_PP 设置。

安全保护

FMC 提供了安全保护功能, 防止 Flash 存储器上的非法代码 / 数据访问。该功能用于保护软件 / 固件不受非法使用者侵害。通过设置选项字节 OB_CP [0] 来激活此功能。一旦该功能被使能, 除了用户的应用程序, 所有主 Flash 的 ICP/ 调试模式、编程和页擦除操作将不会被允许。然而, 为了除能安全保护功能, 整片擦除操作仍将被 FMC 接受。下表列出了当安全保护功能使能时, 相应 Flash 存储器的访问权限。

表 8. 安全保护使能时的访问权限

操作 \ 模式	用户应用程序 ⁽¹⁾	ICP / 调试模式
读	O	X (读为 0)
编程	O ⁽¹⁾	X
页擦除	O ⁽¹⁾	X
整片擦除	O	O

注: 1. 用户应用程序是指 JTAG / SW 调试器断开时从主 Flash 存储器开始执行或启动的软件。然而, 选项字节块和第 0 页仍然受保护, 编程 / 页擦除操作不能被执行。
2. 整片擦除操作可擦除选项字节块并可除能安全保护功能。

下列步骤显示了安全保护程序寄存器的访问顺序:

- 检查 OPCR 寄存器, 以确认没有正在执行 Flash 存储器操作 (OPM [3:0] = 0xE 或 0x6)。否则, 需等待直到前面的操作已经完成。
- 在 TADR 寄存器写入 OB_CP 地址写入 (TADR = 0x1FF0_0010)。
- 在 WRDR 寄存器写入数据用于清除 OB_CP [0] 位。
- 将字编程命令写入到 OCMR 寄存器 (CMD [3:0] = 0x4)。
- 通过设置 OPCR 寄存器, 将字编程命令发送到 FMC (OPM [3:0] = 0xA)。
- 等待, 直到通过检查 OPCR 寄存器的值确认所有操作已经完成 (OPM [3:0] = 0xE)。
- 依据需要读取并验证选项字节。
- 必须根据选项字节校验和规则更新 OB_CK。
- 使系统复位以激活新的 OB_CP 设置。

寄存器列表

下表显示了 FMC 寄存器及其复位值。

表 9. FMC 的寄存器列表

寄存器	偏移量	描述	复位值
TADR	0x000	Flash 目标地址寄存器	0x0000_0000
WRDR	0x004	Flash 写数据寄存器	0x0000_0000
OCMR	0x00C	Flash 操作命令寄存器	0x0000_0000
OPCR	0x010	Flash 操作控制寄存器	0x0000_000C
OIER	0x014	Flash 操作中中断使能寄存器	0x0000_0000
OISR	0x018	Flash 操作中中断状态寄存器	0x0001_0000
PPSR	0x020 0x024 0x028 0x02C	Flash 页擦除 / 编程保护状态寄存器	0xXXXX_XXXX 0xXXXX_XXXX 0xXXXX_XXXX 0xXXXX_XXXX
CPSR	0x030	Flash 安全保护状态寄存器	0x0000_000X
VMCR	0x100	Flash 向量映射控制寄存器	0x0000_000X
MDID	0x180	Flash 制造商与设备 ID 寄存器	0x0376_XXXX
PNSR	0x184	Flash 页数量状态寄存器	0x0000_00XX
PSSR	0x188	Flash 页大小状态寄存器	0x0000_0400
DIDR	0x18C	设备 ID 寄存器	0x000X_XXXX
CIDR0	0x310	自定义 ID 寄存器 0	0xXXXX_XXXX
CIDR1	0x314	自定义 ID 寄存器 1	0xXXXX_XXXX
CIDR2	0x318	自定义 ID 寄存器 2	0xXXXX_XXXX
CIDR3	0x31C	自定义 ID 寄存器 3	0xXXXX_XXXX

注：“X”表示未知的复位值，其值取决于单片机、Flash 值、选项字节值或上电复位设置。

寄存器描述

Flash 目标地址寄存器 – TADR

该寄存器定义了页擦除和字编程操作的目标地址。

偏移量: 0x000
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
	TADB								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	23	22	21	20	19	18	17	16	
	TADB								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	15	14	13	12	11	10	9	8	
	TADB								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
	TADB								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	字段	描述
[31:0]	TADB	Flash 目标地址位 对于编程操作，TADR 寄存器指定了数据将要被写入的地址。由于编程长度是 32 位，TADR 应被设置为字对齐 (4 字节)。TADB [1:0] 在编程操作期间将被忽略。对于页擦除操作，TADR 寄存器包含将会被擦除的页地址。由于页大小为 1 KB，TADB [9:0] 将被忽略，以限制目标地址为 1 KB 对齐。对于 32 KB 主 Flash 寻址，TADB [31:15] 应为 0，而对于 16 KB 主 Flash 寻址，TADB [31:14] 应为 0。1 KB 选项字节的地址范围从 0x1FF0_0000 到 0x1FF0_03FF。有效的 Flash 存储器地址范围必须在 0x0000_0000 到 0x1FFF_FFFF 以内。否则，如果相应的中断使能位被置位，将产生一个无效目标地址中断。

Flash 写数据寄存器 – WRDR

该寄存器表示了用于编程操作的写入数据。

偏移量: 0x004

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
	WRDB								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	23	22	21	20	19	18	17	16	
	WRDB								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	15	14	13	12	11	10	9	8	
	WRDB								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
	WRDB								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

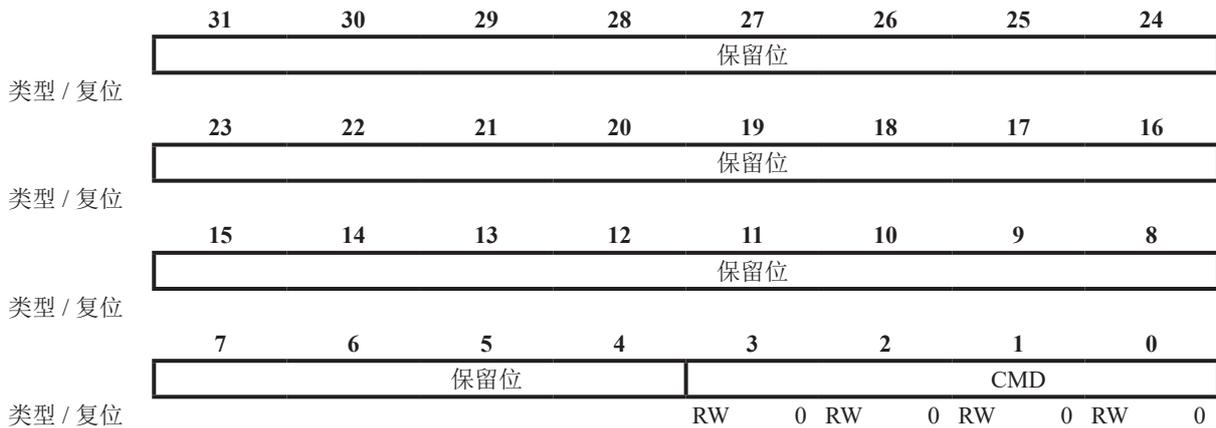
位	字段	描述
[31:0]	WRDB	Flash 写数据位 用于编程操作的数据值。

Flash 操作命令寄存器 – OCMR

该寄存器定义了 Flash 操作命令，包括字编程、页擦除和整片擦除。

偏移量： 0x00C

复位值： 0x0000_0000



位	字段	描述
[3:0]	CMD	Flash 操作命令位 下表显示了操作命令位 CMD[3:0] 的定义，它决定了 Flash 存储器的操作。如果设置了一个无效的命令，且 IOCMIEEN 位等于 1，将产生一个无效操作命令中断。

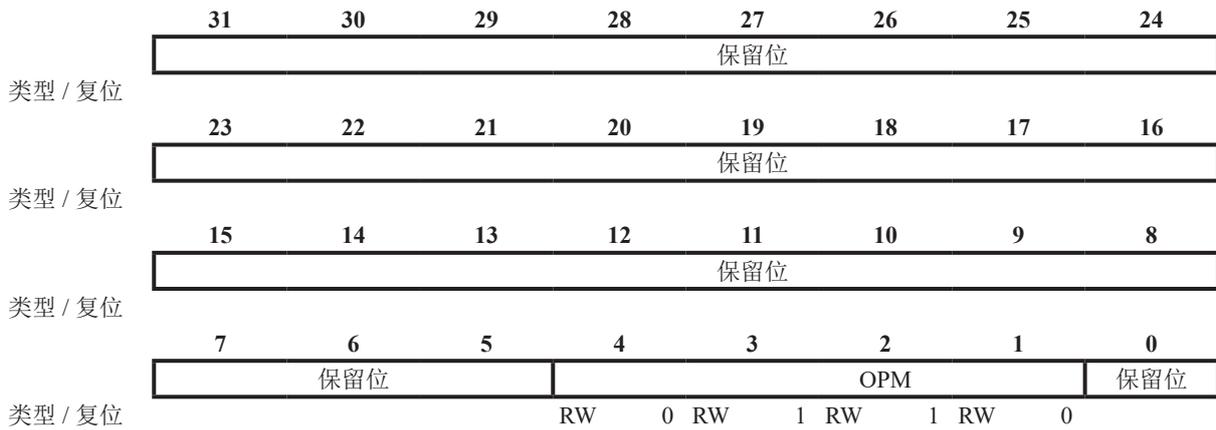
CMD[3:0]	描述
0x0	空闲 (默认)
0x4	字编程
0x8	页擦除
0xA	整片擦除
其它值	保留

Flash 操作控制寄存器 – OPCR

该寄存器用于控制命令提交和检查 FMC 操作的状态。

偏移量: 0x010

复位值: 0x0000_000C



位	字段	描述
---	----	----

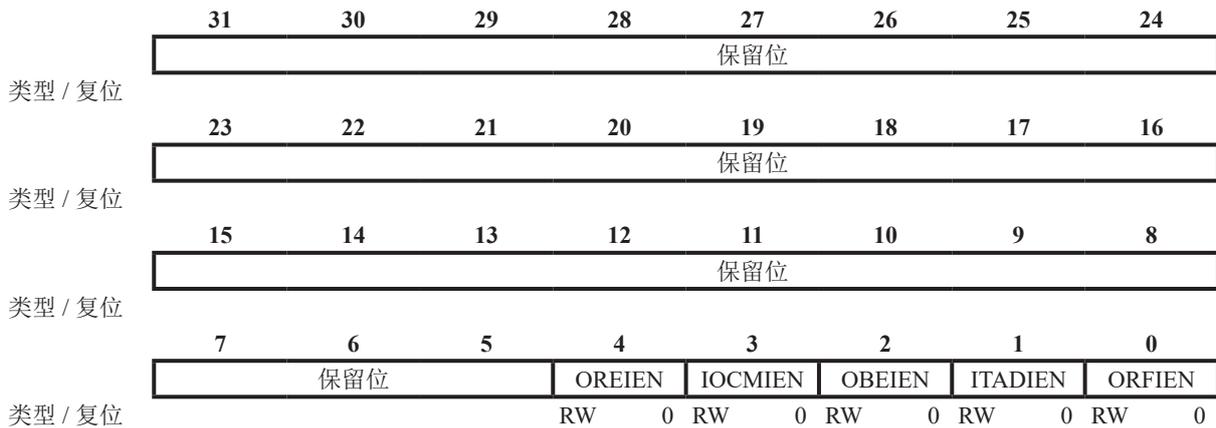
[4:1]	OPM	<p>操作模式位</p> <p>下表列出了 FMC 的操作模式。根据 TADR 寄存器的地址别名设置，用户可以提交由 OCMR 寄存器对 FMC 设置的命令。TADR、WRDR 和 OCMR 寄存器的内容应该在设置该寄存器之前先设置好。在所有操作都完成后，OPM 字段将由 FMC 硬件设置为 0xE 或 0xF。当所有操作已完成，设置进入空闲模式以降低功耗。注意，FMC 执行下一步操作之前，应检查操作状态。TADR、WRDR、OCMR 和 OPCR 寄存器的内容不应该被改变直到前面的操作完成。</p>
-------	-----	---

OPM[3:0]	描述
0x6	空闲 (默认)
0xA	提交命令给主 Flash
0xE	所有主 Flash 操作都已完成
其它值	保留

Flash 操作中断使能寄存器 – OIER

该寄存器用来使能或除能 FMC 中断功能。当相应的中断使能位被置位，当相应的事件发生时 FMC 会产生中断。

偏移量： 0x014
复位值： 0x0000_0000



位	字段	描述
[4]	OREIEN	操作错误中断使能位 0: 操作错误中断除能 1: 操作错误中断使能
[3]	IOCM IEN	无效操作命令中断使能位 0: 无效操作命令中断除能 1: 无效操作命令中断使能
[2]	OBEIEN	选项字节校验和错误中断使能位 0: 选项字节校验和错误中断除能 1: 选项字节校验和错误中断使能
[1]	ITADIEN	无效目标地址中断使能位 0: 无效目标地址中断除能 1: 无效目标地址中断使能
[0]	ORFIEN	操作完成中断使能位 0: 操作完成中断除能 1: 操作完成中断使能

Flash 操作中断状态寄存器 – OISR

该寄存器定义了 FMC 的中断状态，它用来检查一个 Flash 操作是否完成或是否有错误发生。当 OIER 寄存器中相应的中断使能位被置位时，该状态字段 [4:0] 置位后才能触发中断。

偏移量： 0x018

复位值： 0x0001_0000

	31	30	29	28	27	26	25	24		
	保留位									
类型 / 复位										
	23	22	21	20	19	18	17	16		
	保留位						PPEF	RORFF		
类型 / 复位							RO	0	RO	1
	15	14	13	12	11	10	9	8		
	保留位									
类型 / 复位										
	7	6	5	4	3	2	1	0		
	保留位		OREF		IOCMF		OBEF		ITADF	
类型 / 复位			WC		0		WC		0	
			WC		0		WC		0	

位	字段	描述
[17]	PPEF	页擦除 / 编程保护错误标志位 0: 页擦除 / 编程保护错误未发生 1: 操作错误发生，由于执行无效的页擦除 / 编程操作到受保护页一旦新的 Flash 操作命令被提交，该位由硬件复位。
[16]	RORFF	原始操作完成标志位 0: 上一个 Flash 操作命令未完成 1: 上一个 Flash 操作命令完成 该位直接连接到 Flash 存储器用于调试。
[4]	OREF	操作错误标志位 0: Flash 操作错误未发生 1: 上一个 Flash 操作失败 当任何一个 Flash 操作错误发生时，例如无效的命令、编程错误和擦除错误等，该位将被置位。如果 OIER 寄存器中的 OREIEN 位被置位，ORE 中断将发生。通过写 1 来复位该位。
[3]	IOCMF	无效操作命令标志位 0: 未设置无效的 Flash 操作命令 1: 无效的 Flash 操作命令被写入到 OCMR 寄存器 当无效的 Flash 操作命令被写入到 OCMR 寄存器时此位被置位，如果 OIER 寄存器中的 IOCMIEEN 位被置位，IOCM 中断将发生。通过写 1 来复位该位。
[2]	OBEF	选项字节校验和错误标志位 0: 选项字节校验和正确 1: 选项字节校验和错误 当出现选项字节校验和错误时此位被置位，如果 OIER 寄存器中的 OBEIEN 位被置位，OBE 中断将发生。通过写 1 来复位该位。但直到中断条件清除后 OBEF 才可被软件复位，即选项字节校验和的值需要正确调整或相应的中断控制需除能，否则将一直产生中断。

位	字段	描述
[1]	ITADF	无效目标地址标志位 0: 目标地址有效 1: 目标地址无效 TADR 字段中的数据必须在 0x0000_0000 到 0x1FFF_FFFF 范围内。否则此位将被置位, 如果 OIER 寄存器中的 ITADIEN 位被置位, ITAD 中断将发生。通过写 1 来复位该位。
[0]	ORFF	操作完成标志位 0: Flash 操作完成未发生 1: Flash 操作完成发生 当 Flash 操作完成时此位将被置位, 如果 OIER 寄存器中的 ORFIEN 位被置位, ORF 中断将发生。通过写 1 来复位该位。

Flash 页擦除 / 编程保护状态寄存器 – PPSR

该寄存器定义了 Flash 存储器的页擦除 / 编程保护功能。

偏移量: 0x020 (0) ~ 0x02C (3)
复位值: 0xXXXX_XXXX

	31	30	29	28	27	26	25	24	
	PPSBn								
类型 / 复位	RO	X RO	X RO	X RO	X RO	X RO	X RO	X RO	X
	23	22	21	20	19	18	17	16	
	PPSBn								
类型 / 复位	RO	X RO	X RO	X RO	X RO	X RO	X RO	X RO	X
	15	14	13	12	11	10	9	8	
	PPSBn								
类型 / 复位	RO	X RO	X RO	X RO	X RO	X RO	X RO	X RO	X
	7	6	5	4	3	2	1	0	
	PPSBn								
类型 / 复位	RO	X RO	X RO	X RO	X RO	X RO	X RO	X RO	X

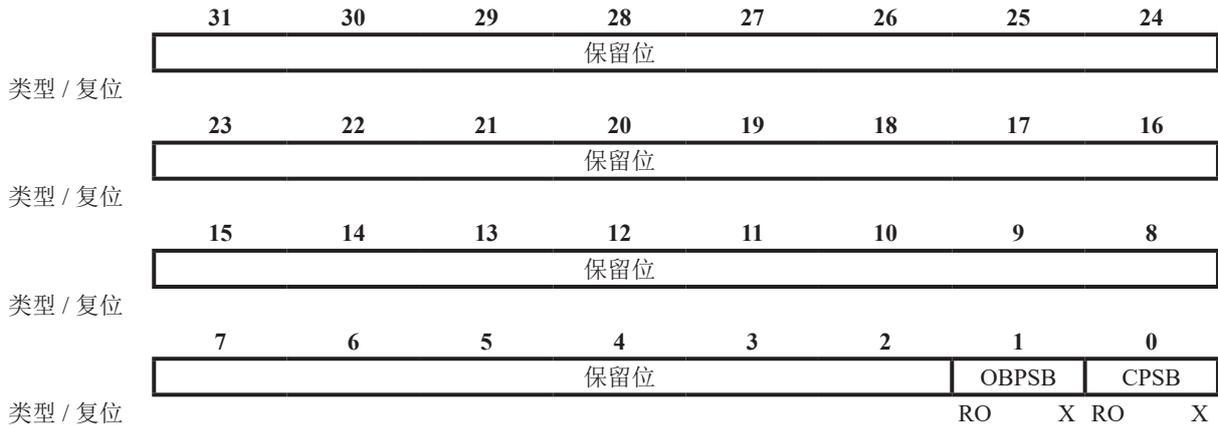
位	字段	描述
[127:0]	PPSBn	擦除 / 编程保护状态位 (n = 0 ~ 127) PPSB[n] = OB_PP[n] 0: 相应的第 n 页受保护 1: 相应的第 n 页不受保护 该寄存器的内容不是动态更新的, 只有在任意复位发生时从选项字节重新加载。当 PPSR 寄存器中的相应位被复位为零时, 特定页的擦除或编程功能不被允许。PPSR [127:0] 位的复位值由选项字节 OB_PP [127:0] 决定。基于芯片规格, 主 Flash 最大页数量不同。因此, 每个页擦除 / 编程保护状态位可保护 1 或 2 页, 这取决于芯片规格。OB_PP 和 PPSR 寄存器的其它位被保留用来作未来扩充。

Flash 安全保护状态寄存器 – CPSR

该寄存器定义了 Flash 存储器安全保护状态。它的内容不是动态更新的，只有在任意复位发生时通过选项字节加载器重新加载。

偏移量： 0x030

复位值： 0x0000_000X

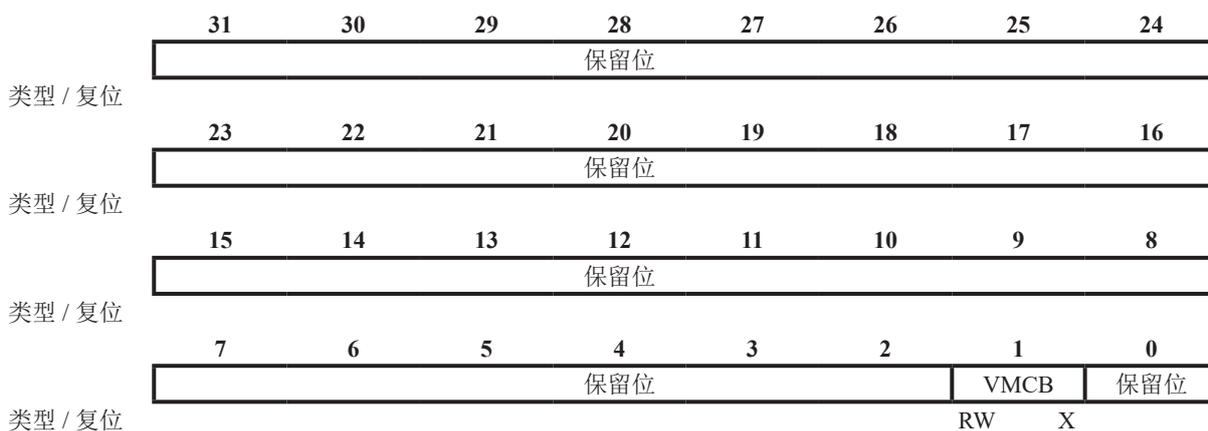


位	字段	描述
[1]	OBPSB	选项字节页擦除 / 编程保护状态位 0: 选项字节页受保护 1: 选项字节页不受保护 OBPSB 位的复位值是由选项字节中的 OB_CP [1] 位决定。
[0]	CPSB	Flash 存储器安全保护状态位 0: Flash 存储器安全保护使能 1: Flash 存储器安全保护除能 CPSB 位的复位值是由选项字节中的 OB_CP [0] 位决定。

Flash 向量映射控制寄存器 – VMCR

该寄存器用来控制向量映射。VMCR 寄存器的复位值是由上电复位期间外部启动引脚 BOOT 的状态决定。

偏移量: 0x100
复位值: 0x0000_000X



位	字段	描述
[1]	VMCB	向量映射控制位 VMCB 位用来控制地址前 4 个字 (从地址 0x0 到 0xC) 向量的映射源。下表显示了向量映射的设置。

BOOT	VMCB	描述
低	0	启动加载器模式 向量映射源是启动加载器区域
高	1	主 Flash 模式 向量映射源是主 Flash 存储器区域

VMCR 寄存器的复位值由上电复位和系统复位期间外部启动引脚 BOOT 的状态决定。但是, 当应用程序被执行时, 通过配置 VMCB 位可以暂时改变向量映射设置。

Flash 制造商和设备 ID 寄存器 – MDID

该寄存器定义了制造商 ID 和设备型号信息，可用于产品识别。

偏移量： 0x180

复位值： 0x0376_XXXX

	31	30	29	28	27	26	25	24				
	MFID											
类型 / 复位	RO	0	RO	0	RO	0	RO	0	RO	1	RO	1
	23	22	21	20	19	18	17	16				
	MFID											
类型 / 复位	RO	0	RO	1	RO	1	RO	1	RO	1	RO	0
	15	14	13	12	11	10	9	8				
	ChipID											
类型 / 复位	RO	X	RO	X	RO	X	RO	X	RO	X	RO	X
	7	6	5	4	3	2	1	0				
	ChipID											
类型 / 复位	RO	X	RO	X	RO	X	RO	X	RO	X	RO	X

位	字段	描述
[31:16]	MFID	制造商 ID 读为 0x0376
[15:0]	ChipID	芯片 ID 读取 MCU 设备型号的后四个数字编码。

Flash 页数状态寄存器 – PNSR

该寄存器用于显示 Flash 存储器的页数。

偏移量: 0x184

复位值: 0x0000_00XX

	31	30	29	28	27	26	25	24	
	PNSB								
类型 / 复位	RO	0	RO	0	RO	0	RO	0	RO
	23	22	21	20	19	18	17	16	
	PNSB								
类型 / 复位	RO	0	RO	0	RO	0	RO	0	RO
	15	14	13	12	11	10	9	8	
	PNSB								
类型 / 复位	RO	0	RO	0	RO	0	RO	0	RO
	7	6	5	4	3	2	1	0	
	PNSB								
类型 / 复位	RO	X	RO	X	RO	X	RO	X	RO

位	字段	描述
[31:0]	PNSB	Flash 页数状态位 0x0000_0010: 片上 Flash 存储器总页数为 16 页 0x0000_0020: 片上 Flash 存储器总页数为 32 页 0x0000_0040: 片上 Flash 存储器总页数为 64 页 0x0000_0080: 片上 Flash 存储器总页数为 128 页 0x0000_00FF: 片上 Flash 存储器总页数为 255 页 这些位用于表明此片上 Flash 存储设备的总页数。

Flash 页大小状态寄存器 – PSSR

该寄存器用于以字节形式显示页大小。

偏移量: 0x188

复位值: 0x0000_0400

	31	30	29	28	27	26	25	24	
	PSSB								
类型 / 复位	RO	0	RO	0	RO	0	RO	0	RO
	23	22	21	20	19	18	17	16	
	PSSB								
类型 / 复位	RO	0	RO	0	RO	0	RO	0	RO
	15	14	13	12	11	10	9	8	
	PSSB								
类型 / 复位	RO	0	RO	0	RO	0	RO	1	RO
	7	6	5	4	3	2	1	0	
	PSSB								
类型 / 复位	RO	0	RO	0	RO	0	RO	0	RO

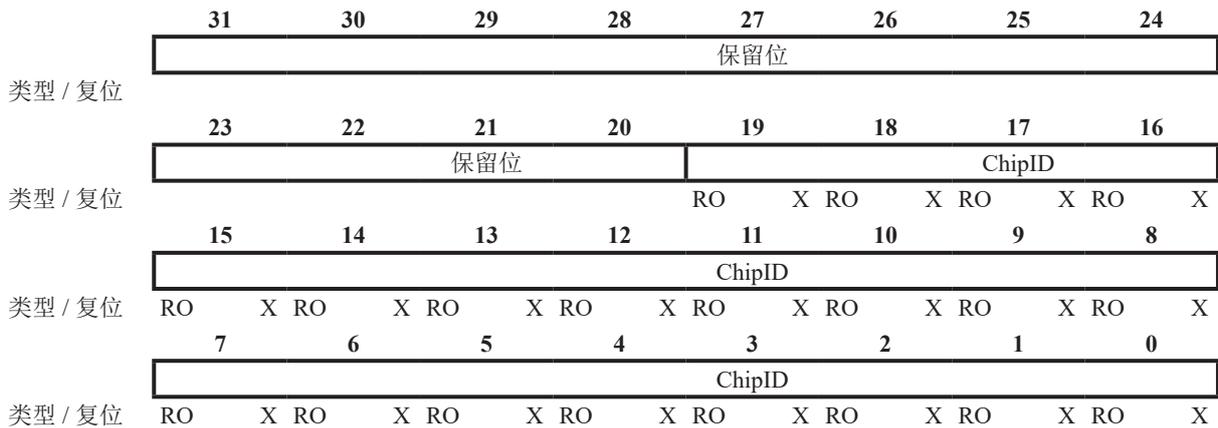
位	字段	描述
[31:0]	PSSB	Flash 页大小状态位 0x200: 每页 512 字节 0x400: 每页 1 KB 0x800: 每页 2 KB

设备 ID 寄存器 – DIDR

该寄存器用于存储设备型号信息，可用于产品识别。

偏移量： 0x18C

复位值： 0x000X_XXXX



位	字段	描述
[19:0]	ChipID	芯片 ID 读取 MCU 设备型号所有的数字编码。

自定义 ID 寄存器 n – CIDRn, n = 0 ~ 3

该寄存器用于存储自定义 ID 信息，可用于自定义标识。

偏移量： 0x310 (0) ~ 0x31C (3)

复位值： 有多种复位值，取决于 Flash 制造权限信息块。

	31	30	29	28	27	26	25	24	
	CID								
类型 / 复位	RO	X RO	X RO	X RO	X RO	X RO	X RO	X RO	X
	23	22	21	20	19	18	17	16	
	CID								
类型 / 复位	RO	X RO	X RO	X RO	X RO	X RO	X RO	X RO	X
	15	14	13	12	11	10	9	8	
	CID								
类型 / 复位	RO	X RO	X RO	X RO	X RO	X RO	X RO	X RO	X
	7	6	5	4	3	2	1	0	
	CID								
类型 / 复位	RO	X RO	X RO	X RO	X RO	X RO	X RO	X RO	X

位	字段	描述
[31:0]	CIDn	自定义 ID Flash 制造权限信息块中自定义 ID 寄存器的 CIDn[31:0] (n = 0 ~ 3) 字段。

5 电源控制单元 (PWRCU)

简介

功耗被视为许多嵌入式系统应用中很重要的问题之一。因此，在这些单片机中，电源控制单元 PWRCU 提供多种省电模式如休眠模式、深度休眠模式 1 和休眠模式 2。这些工作模式可以降低功耗，并允许应用程序在 CPU 运行时间、速度和功耗相互冲突的需求中达到较佳平衡。图 11 中的虚线显示了两个数字电源域的供电来源。

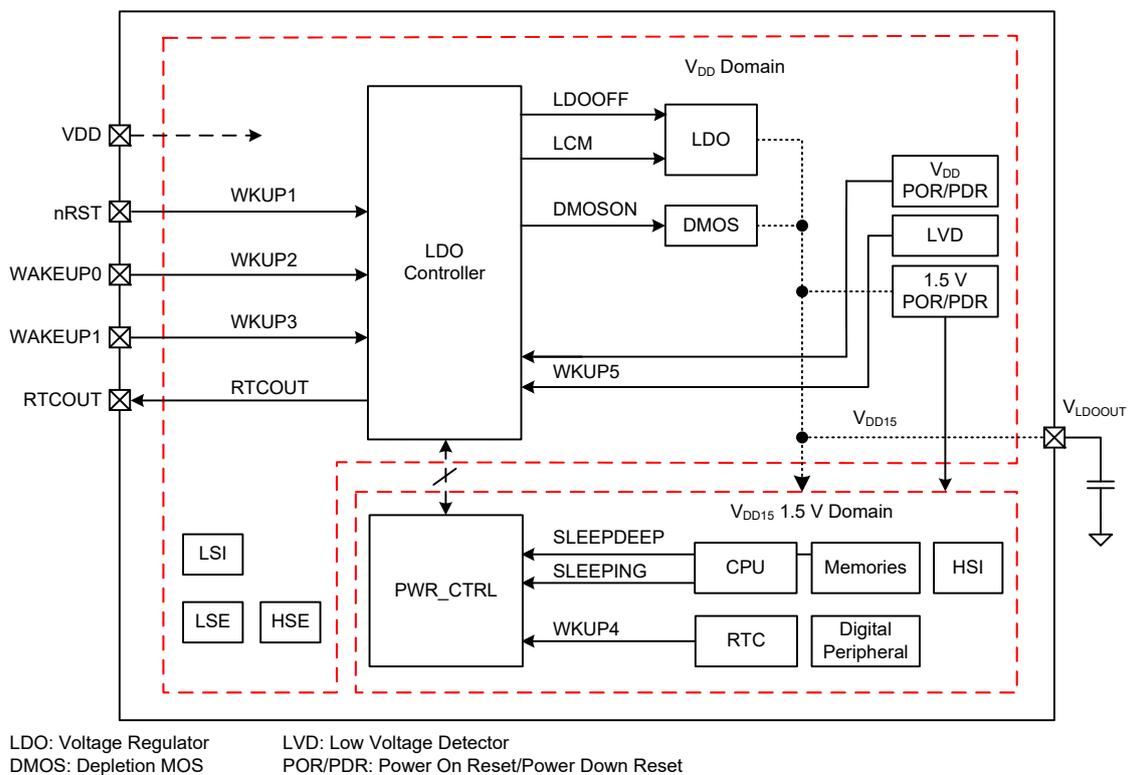


图 11. PWRCU 方框图

特性

- 三个电源域: V_{DD} , V_{DDIO} 和 V_{DD15} 1.5 V 电源域
- 三种省电模式: 休眠模式、深度休眠模式 1 和深度休眠 2
- 内部稳压器提供 1.5 V 电压源
- 额外的耗尽型 MOS 提供 1.5 V 电压源、漏电流和工作电流小
- 当下列事件发生时将产生一个电源复位
 - 上电复位 / 掉电复位 (POR / PDR)
 - 控制位 $BODEN = 1$, $BODRIS = 0$ 且 $V_{DD} \leq V_{BOD}$
- 当 V_{DD} 低于欠压检测器电压 V_{BOD} , 欠压检测器 BOD 可产生一个系统复位或中断
- 当 V_{DD} 低于可编程阈值电压 V_{LVD} , 低电压检测器 LVD 可产生一个中断或唤醒事件

功能描述

V_{DD} 电源域

LDO 电源控制

当下列条件发生时, LDO 将自动关闭:

- 进入深度休眠模式 2

当 $V_{DD} > V_{POR}$ 时, 若发生以下任一条件, LDO 将由硬件自动开启:

- 从省电模式中恢复操作 – RTC 唤醒, LVD 唤醒, EXTI 唤醒及 WAKEUP 引脚出现上升沿
- 检测到外部复位引脚 (nRST) 的下降沿
- 控制位 $BODEN = 1$ 且 $V_{DD} > V_{BOD}$

要进入深度休眠模式 1, PWRCU 会要求 LDO 工作在低电流模式 LCM 下。要进入深度休眠模式 2, PWRCU 将关闭 LDO, 然后打开 DMOS 提供另一种 1.5 V 电源。

稳压器

稳压器 LDO、耗尽型 MOS 管 DMOS、低电压检测器 LVD, 内部低速振荡器 LSI, 外部低速晶体振荡器 LSE 和外部高速晶体振荡器 HSE 都在 V_{DD} 电源域下工作。LDO 可配置为工作在正常模式 ($LDOOFF = 0$, $LDOLCM = 0$, I_{OUT} = 高电流模式) 或低电流模式 ($LDOOFF = 0$, $LDOLCM = 1$, I_{OUT} = 低电流模式), 以提供 1.5 V 电源。另一种 1.5 V 电源来自具有小静态电流和小驱动电流特性的 DMOS 输出。它通过使用 PWRCR 寄存器中的 DMOSON 位控制。DMOS 输出具有弱的输出电流和稳压能力, 在 V_{DD15} 电源域只能工作在深度休眠模式 2 来实现数据保存。

上电复位 (POR) / 掉电复位 (PDR)

该单片机内置 POR / PDR 电路, 允许单片机在从 V_{POR} 开始升压或降压至 V_{POR} 的过程中都可以正常工作。欲知详细上电 / 掉电复位的阈值电压, 请参考相关数据手册 (Datasheet) 中的电气特性部分。

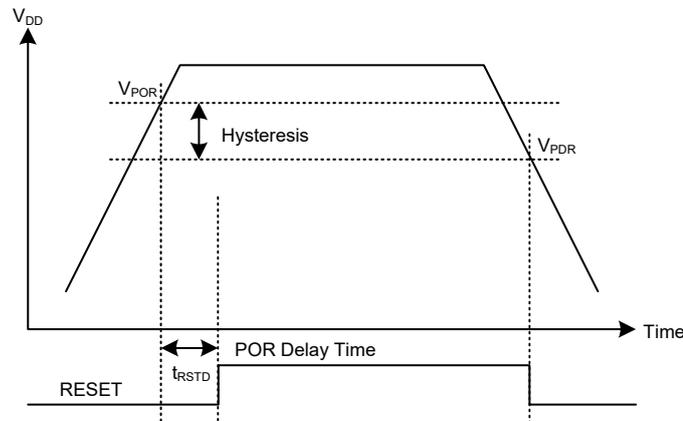


图 12. 上电复位 / 掉电复位波形

低电压检测器 / 欠压检测器

低电压检测 LVD 也可以检测电源电压 V_{DD} 是否低于可编程阈值电压 V_{LVD} 。阈值电压由 LVDCSR 寄存器中的 LVDS 字段选择。当在 V_{DD} 电源引脚上检测到一个低电压, LVDF 标志位将置位, 如果 LVDCSR 寄存器中的 LVDEN 和 LVDIWEN 位被置位, 将产生一个中断并发送给 MCU 内核。有关 LVD 可编程阈值电压 V_{LVD} 的详细信息, 可参考数据手册 (Datasheet) 中的电气特性部分。

欠压检测器 BOD 用来检测 V_{DD} 电源电压是否等于或低于 V_{BOD} 。当 LVDCSR 寄存器中的 BODEN 位被设置为 1, V_{DD} 电源电压低于 V_{BOD} 时, BODF 标志位将置位。PWRCU 就会把这个作为一个掉电复位的情况, 然后立即除能内部 LDO 稳压器 (当 BODRIS = 0) 或发出中断来通知 CPU 执行相应的程序后待机 (当 BODRIS = 1)。有关欠压检测器 BOD 电压 V_{BOD} 的详细信息, 可参考数据手册 (Datasheet) 中的电气特性部分。

外部高速振荡器

外部高速振荡器 HSE 位于 V_{DD} 电源域。通过设置 HSEEN 位可以关闭和开启该时钟, 此位位于时钟控制单元 CKCU 中的全局时钟控制寄存器 GCCR。HSE 时钟可直接用作系统时钟源。

LSE, LSI 与 RTC

实时时钟定时器 RTC 的时钟源可来自内部低速振荡器 LSI 或外部低速晶体振荡器 LSE。在执行 WFI / WFE 指令进入省电模式之前, MCU 需要在 RTC 比较寄存器设置所需要的唤醒时间, 并使能唤醒功能以完成 RTC 定时器唤醒事件。在进入省电模式一段时间后, 比较匹配标志位 CMFLAG 将会被置位以在比较匹配事件发生时唤醒单片机。关于唤醒定时器的 RTC 配置细节将在 RTC 章节中进行描述。

1.5 V 电源域

主要功能包括用于 V_{DD} 电源域的内部高速振荡器 HSI、MCU 内核逻辑、AHB / APB 外设和存储器等, 都位于此电源域。一旦 1.5 V 电源上电, POR 将在 1.5 V 电源域中产生一个复位序列。随后, 要进入相应的省电模式, 相关的控制位包括 LDOOFF、DMOSON 和 LDOLCM 位必须被配置。然后, 一旦 WFI 或 WFE 指令被执行, 单片机将进入相应的省电模式, 这个将在下一章节讨论。

内部高速振荡器

内部高速振荡器 HSI 位于 V_{DD15} 电源域。当从深度休眠模式退出时，通过设置 PSRCEN 位为 1，HSI 时钟可以在一段时间内被配置作为系统时钟，此位位于时钟控制单元 CKCU 中的全局时钟控制寄存器 GCCR。系统时钟不会切换回进入深度休眠模式之前使用的初始时钟源，直到这个初始时钟源稳定。

工作模式

运行模式

在运行模式下，系统可运行所有功能且所有电源域都可用。此模式下，有两种方法减少功耗。第一种是通过设置 CKCU AHBCFGR 寄存器中的 AHBPRE 字段以减慢系统时钟，第二种是通过设置 APBCCR0 和 APBCCR1 寄存器以关闭未使用的外设时钟或通过设置 APBPCSR0 和 APBPCSR1 寄存器以降低外设时钟频率以满足应用需求。在进入休眠模式之前减小系统时钟速度也有助于降低功耗。

此外，有几种省电模式可以提供单片机性能和功耗之间的较大优化。

表 10. 工作模式定义

模式名称	硬件动作
运行	系统复位后，CPU 获取指令执行。
休眠	1. CPU 内核时钟将停止。 2. 外设、Flash 和 SRAM 的时钟可以设置为关闭。
深度休眠 1~2	1. 停止 1.5 V 电源域的所有时钟。 2. 除能 HSI 和 HSE。 3. 通过打开 LDO 低电流模式或 DMOS 来减少 1.5 V 电源域的电流。

休眠模式

默认情况下，系统进入休眠模式后，只有 CPU 时钟会停止。若设置 CKCU AHBCCR 寄存器中的 FMCEN 或 SRAMEN 位为 0，在系统进入休眠模式后，Flash 时钟或 SRAM 时钟也将停止。如果在休眠模式 CPU 不会访问 Flash 存储器和 SRAM，建议清除 AHBCCR 寄存器中的 FMCEN 和 SRAMEN 位来降低功耗。要进入休眠模式，需清零 SLEEPDEEP 位，并执行 WFI 或 WFE 指令。通过任何中断或事件触发，系统将退出休眠模式。下表提供了有关省电模式的更多信息。

表 11. 进入 / 退出省电模式

模式	模式进入				模式退出
	CPU 指令	CPU SLEEPDEEP	LDOOFF	DMOSON	
休眠		0	X	X	WFI: 任何中断 WFE: 任何唤醒事件 ⁽¹⁾ 或 任何中断 (NVIC on) 或 SEVONPEND = 1 的任何中断 (NVIC off)
深度休眠 1	WFI 或 WFE (生效)	1	0	0	在事件模式的任何 EXTI 或 RTC 唤醒或 LVD 唤醒 ⁽²⁾ 或 WAKEUP 引脚的上升沿
深度休眠 2		1	X	1	在事件模式的任何 EXTI 或 RTC 唤醒或 LVD 唤醒 ⁽²⁾ 或 WAKEUP 引脚的上升沿

注: 1. 唤醒事件是指在事件模式的 EXTI 输入线、RTC、LVD 和 WAKEUP 引脚的上升沿。

2. 系统进入省电模式之后, 如果允许 LVD 来唤醒, LVDCSR 寄存器中的 LVDEWEN 和 LVDEN 位需使能, 以确保系统可以由一个 LVD 事件唤醒, 从深度休眠 2 中唤醒时, LDO 稳压器可以开启。

深度休眠模式

配置如上表所示的寄存器, 并执行 WFI 或 WFE 指令可进入深度休眠模式。在深度休眠模式下, 所有时钟包括高速振荡器, 即 HSI 和 HSE, 将被停止。此外, 在深度休眠模式 1, LDO 设置为低电流模式, 深度休眠模式 2 关闭 LDO, 并使用 DMOS 保持 1.5 V 电源。一旦 PWRCU 接收到一个如上表“模式退出”一栏所述的唤醒事件或中断, LDO 将工作在正常模式, 高速振荡器将使能。最后, CPU 将返回到运行模式, 并依需要来处理唤醒中断。如果 LVDCSR 寄存器中相应的唤醒控制位 LVDEWEN 使能, 低电压检测也可以被看作是唤醒事件。最后一个唤醒事件是外部 WAKEUP 引脚由低到高电平的转换, 发送到 PWRCU 以从深度休眠模式中恢复。在深度休眠模式, 保持寄存器和存储器的内容会缩短唤醒延迟。

表 12. 系统复位后的电源状态

PORF	PORSTF	描述
1	1	V _{DD} 电源域复位后首次上电: 当 V _{DD} 首次使用或在对 V _{DD} 域执行软件复位。
0	1	从 1.5 V 电源意外掉电重新启动或其它复位 (nRST, WDT 等)

寄存器列表

下表显示了 PWRCU 寄存器及其复位值。注意, 在本单元的所有寄存器都位于 V_{DD15} 电源域。

表 13. PWRCU 寄存器列表

寄存器	偏移量	描述	复位值
PWRSR	0x100	电源控制状态寄存器	0x0000_0010
PWRCCR	0x104	电源控制寄存器	0x0000_0000
LVDCSR	0x110	低电压 / 欠压检测控制和状态寄存器	0x0000_0000

寄存器描述

电源控制状态寄存器 – PWRSR

该寄存器定义了电源控制状态。

偏移量: 0x100

复位值: 0x0000_0010

	31	30	29	28	27	26	25	24		
类型 / 复位	保留位									
	23	22	21	20	19	18	17	16		
类型 / 复位	保留位									
	15	14	13	12	11	10	9	8		
类型 / 复位	保留位						RC	0	RC	0
	7	6	5	4	3	2	1	0		
类型 / 复位	保留位		PORF		保留位					
				RC	1					

位	字段	描述
[9]	WUPF1	外部 WAKEUP1 引脚标志位 0: WAKEUP1 引脚无效 1: WAKEUP1 引脚有效 当 WAKEUP1 引脚有效时, 该位由硬件置位, 当软件读取时, 该位清零。系统从省电模式中唤醒后, 软件应读取该位使其清零。
[8]	WUPF0	外部 WAKEUP0 引脚标志位 0: WAKEUP0 引脚无效 1: WAKEUP0 引脚有效 当 WAKEUP0 引脚有效时, 该位由硬件置位, 当软件读取时, 该位清零。系统从省电模式中唤醒后, 软件应读取该位使其清零。
[4]	PORF	上电复位标志位 0: V _{DD15} 电源域复位未发生 1: V _{DD15} 电源域复位发生 当 V _{DD15} 电源域复位发生时, 无论是 V _{DD15} 电源域硬件上电复位还是 V _{DD15} 电源域软件复位, 该位由硬件置位, 由软件读取清零。系统首次启动后, 该位必须被清零, 否则 V _{DD15} 电源域复位触发时将无法检测到。当该位读为 1 时, 必须执行连续软件读取, 直到此位再次回到 0 为止。连续软件读取是为了确认 V _{DD15} 电源域是否访问就绪。在 V _{DD15} 电源域首次上电后必须执行读软件循环。

电源控制寄存器 – PWRCCR

该寄存器为不同的省电模式提供了电源控制位。

偏移量: 0x104
复位值: 0x0000_0000

31	30	29	28	27	26	25	24	
保留位								
23	22	21	20	19	18	17	16	
保留位				WUP1TYPE		WUP0TYPE		
类型 / 复位				RW	0	RW	0	
15	14	13	12	11	10	9	8	
DMSOSSTS	保留位			WUP1IEN	WUP1EN	WUP0IEN	WUP0EN	
RO	0	类型 / 复位			RW	0	RW	0
7	6	5	4	3	2	1	0	
DMSOSON	保留位			LDOOFF	LDOLCM	保留位	PWCURST	
RW	0	类型 / 复位			RW	0	WO	0

位 字段 描述

[19:18] WUP1TYPE WAKEUP1 信号触发类型

WUP1TYPE [1:0]		WAKEUP1 信号触发类型
0	0	上升沿触发
0	1	下降沿触发
1	0	高电平有效
1	1	低电平有效

[17:16] WUP0TYPE WAKEUP0 信号触发类型

WUP0TYPE [1:0]		WAKEUP0 信号触发类型
0	0	上升沿触发
0	1	下降沿触发
1	0	高电平有效
1	1	低电平有效

[15] DMSOSSTS 外 DMOS 状态位

如果此寄存器中的 DMSOSON 位被设置为 1，该位被置 1。
如果 DMSOSON 位被设置为 0 或 POR / PDR 复位发生，该位被清零。

[11] WUP1IEN 外部 WAKEUP1 引脚中断使能位

0: 除能外部 WAKEUP1 引脚中断功能
1: 使能外部 WAKEUP1 引脚中断功能
当 WUP1EN 和 WUPF1 位都被设置为 1 时，软件可以设置 WUP1IEN 位为 1 来触发 NVIC 单元的 WKUP 中断。

位	字段	描述
[10]	WUP1EN	<p>外部 WAKEUP1 引脚中断使能位</p> <p>0: 除能外部 WAKEUP1 引脚功能 1: 使能外部 WAKEUP1 引脚功能</p> <p>在进入省电模式之前, 软件可以设置 WUP1EN 位为 1 以使能 WAKEUP1 引脚功能。当 WUP1EN = 1 时, WAKEUP1 引脚的上升沿将系统从省电模式中唤醒。如果 WAKEUP1 引脚是高电平有效, 当此位为高时, 该引脚应设置为输入下拉模式。在使用 WAKEUP1 引脚前, 应先通过配置位于 AFIO 的 GPBCFGHR 寄存器中的 PBCFG9[3:0] 字段设置为 0xF 以选择 WAKEUP1 引脚的复用功能。WAKEUP1 引脚上的相应上拉功能可通过将位于 GPIO 的 PBPUR 寄存器中的 PBPUR[9] 清除为 0 实现除能而通过设置位于 GPIO 的 PBPDR 寄存器中的 PBPDR[9] 为 1 可使能下拉功能。</p> <p>注: 该位由 V_{DD15} 电源域复位来复位。因为该位位于 V_{DD15} 电源域。WAKEUP1 引脚信号至少需保持三个 32 kHz 时钟周期, 直到检测到系统唤醒为止。</p>
[9]	WUP0IEN	<p>外部 WAKEUP0 引脚中断使能位</p> <p>0: 除能外部 WAKEUP0 引脚中断功能 1: 使能外部 WAKEUP0 引脚中断功能</p> <p>当 WUP0EN 和 WUPF0 位都被设置为 1 时, 软件可以设置 WUP0IEN 位为 1 来触发 NVIC 单元的 WKUP 中断。</p>
[8]	WUP0EN	<p>外部 WAKEUP0 引脚使能位</p> <p>0: 除能外部 WAKEUP0 引脚功能 1: 使能外部 WAKEUP0 引脚功能</p> <p>在进入省电模式之前, 软件可以设置 WUP0EN 位为 1 以使能 WAKEUP0 引脚功能。当 WUP0EN = 1 时, WAKEUP0 引脚的上升沿将系统从省电模式中唤醒。由于 WAKEUP0 引脚是高电平有效, 当此位为高时, 该引脚应设置为输入下拉模式。在使用 WAKEUP0 引脚前, 应先通过配置位于 AFIO 的 GPBCFGHR 寄存器中的 PBCFG12[3:0] 字段设置为 0xF 以选择 WAKEUP0 引脚的复用功能。WAKEUP0 引脚上的相应上拉功能可通过将位于 GPIO 的 PBPUR 寄存器中的 PBPUR[12] 清除为 0 实现除能而通过设置位于 GPIO 的 PBPDR 寄存器中的 PBPDR[12] 为 1 可使能下拉功能</p> <p>注: 该位由 V_{DD15} 电源域复位来复位。因为该位位于 V_{DD15} 电源域。WAKEUP0 引脚信号至少需保持三个 32 kHz 时钟周期, 直到检测到系统唤醒为止。</p>
[7]	DMOSON	<p>DMOS 控制位</p> <p>0: DMOS 关闭 1: DMOS 开启</p> <p>当 CPU 进入深度休眠模式 (SLEEPDEEP = 1), DMOS 运行作为 1.5 V 电源域的另一个电压源。控制位 DMOSON 由软件置位, 由软件或 V_{DD} 电源域复位清零。如果 DMOSON 位设置为 1, 当 CPU 进入深度休眠模式, LDO 将自动关闭。</p>
[3]	LDOOFF	<p>LDO 工作模式控制位</p> <p>0: 当 CPU 进入深度休眠模式 (SLEEPDEEP = 1), LDO 工作在低电流模式。V_{DD15} 电源可用。 1: 当 CPU 进入深度休眠模式 (SLEEPDEEP = 1), LDO 关闭。V_{DD15} 电源不可用。</p> <p>注: 该位仅在 DMOSON 位清除为 0 时有效。</p>
[2]	LDOLCM	<p>LDO 低电流模式</p> <p>0: LDO 工作在正常电流模式 1: LDO 工作在低电流模式</p> <p>注: 该位仅在 CPU 处于运行模式时有效。当 LDOLCM 置位时, LDO 输出电流能力将被限制在低于 10 mA。此举可获得更低的电流消耗, 适用于工作在较低速系统时钟下的 CPU。当 LDO 关闭或 V_{DD} 电源域复位时, 此位将被清零。</p>
[0]	PWCURST	<p>电源控制单元软件复位</p> <p>0: 无动作 1: 电源控制单元软件复位启动</p> <p>当此位置位, RTC 和 PWRCU 相关的寄存器将都会被复位。</p>

低电压 / 欠压检测控制和状态寄存器 – LVDCSR

该寄存器定义了低电压检测器的标志位、使能位和选项位。

偏移量: 0x110
复位值: 0x0000_0000

31	30	29	28	27	26	25	24
保留位							
23	22	21	20	19	18	17	16
保留位	LVDS[2]	LVDEWEN	LVDIWEN	LVDF	LVDS[1:0]		LVDEN
类型 / 复位	RW	0 RW	0 RW	0 RO	0 RW	0 RW	0 RW
15	14	13	12	11	10	9	8
保留位							
7	6	5	4	3	2	1	0
保留位				BODF	保留位	BODRIS	BODEN
类型 / 复位				RO	0	RW	0 RW

位	字段	描述
[21]	LVDEWEN	LVD 事件唤醒使能位 0: LVD 事件唤醒除能 1: LVD 事件唤醒使能 将该位设置为 1, 将使能 LVD 事件唤醒功能, 当低电压事件发生时, 会唤醒系统并导致 LVDF 置位。如果系统需要通过 LVD 条件从深度休眠模式中唤醒, 该位必须设为 1。
[20]	LVDIWEN	LVD 中断唤醒使能位 0: LVD 中断唤醒除能 1: LVD 中断唤醒使能 将该位设置为 1, 将使能 LVD 中断功能。当 LVD 条件发生且 LVDIWEN 位设为 1, 将发生 LVD 中断并发送到 CPU NVIC 单元。
[19]	LVDF	低电压检测状态标志位 0: V _{DD} 高于指定的电压 1: V _{DD} 等于或低于指定的电压 当 LVD 条件产生, LVDF 标志位将被置位。当 LVDF 标志位被置位, 如果 LVDIWEN 位设为 1, 将发生 LVD 中断并发送到 CPU。但是, 如果 LVDEWEN 位设为 1 且 LVDIWEN 清为 0, 当 LVDF 标志位被置位时, 将只发生 LVD 事件不产生 LVD 中断。
[22], [18:17]	LVDS[2:0]	低电压检测电平选择位 关于 LVD 可编程阈值电压的更多细节, 可参考相应数据手册 (Datasheet) 的电气特性。
[16]	LVDEN	低电压检测使能位 0: 除能低电压检测 1: 使能低电压检测 将该位设置为 1, 当 V _{DD} 电源等于或低于由 LVDS 位设置的电压时, 将发生 LVD 事件。因此, 当该位在系统进入深度休眠模式 2 (DMOS 打开, LDO 关闭) 之前被使能, 则 LVDEWEN 位必须被使能, 以避免在 CPU 由低电压检测功能唤醒时 LDO 无法启动。
[3]	BODF	欠压检测标志位 0: V _{DD} > V _{BOD} 1: V _{DD} ≤ V _{BOD}

位	字段	描述
[1]	BODRIS	BOD 复位或中断选择位 0: 复位整个单片机 1: 产生中断
[0]	BODEN	欠压检测器使能位 0: 除能欠压检测 1: 使能欠压检测

6 时钟控制单元 (CKCU)

简介

时钟控制单元 CKCU 提供了多种频率和时钟功能。包括内部高速 RC 振荡器 (HSI)、外部高速晶体振荡器 (HSE)、内部低速 RC 振荡器 (LSI)、外部低速晶体振荡器 (LSE)、HSE 时钟监控、时钟分频器、时钟多路复用器和时钟门控电路。AHB、APB 和 CPU 的时钟来自系统时钟 CK_SYS，而系统时钟可以来源于 HSI、HSE、LSI 或 LSE。看门狗定时器和实时时钟 RTC 使用 LSI 或 LSE 作为其时钟源。

一些内部时钟也可以通过 CKOUT 引脚引出用于调试。时钟监控电路可用于检测 HSE 时钟故障。一旦 HSE 时钟因损坏或被移除等原因停止运行，CKCU 将迫使系统时钟源切换到 HSI 时钟，以防止发生系统暂停。

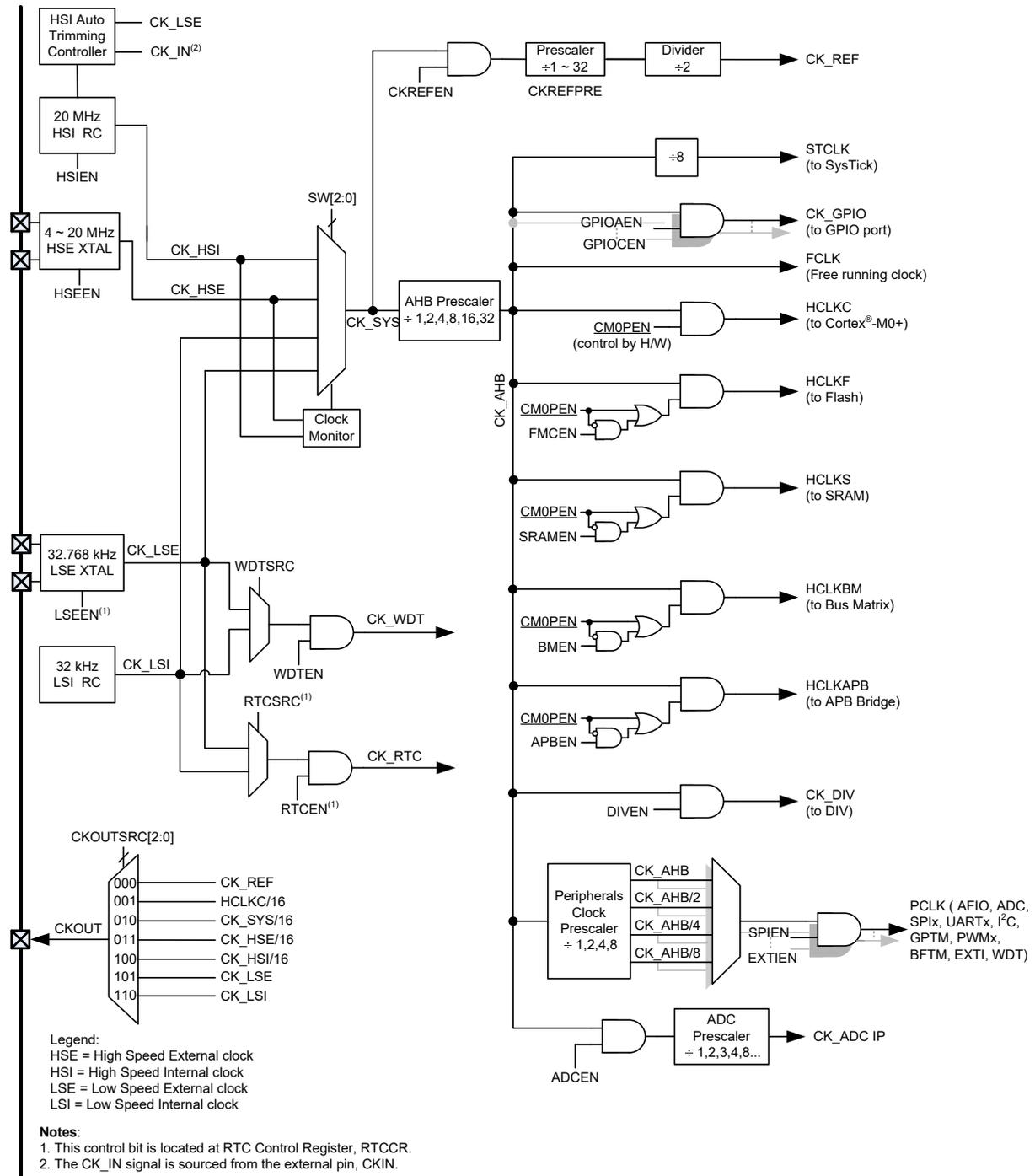


图 13. CKCU 方框图

特性

- 4 ~ 20 MHz 外部高速晶体振荡器 – HSE
- 20 MHz 内部高速 RC 振荡器 – HSI，具有配置选项校准和自定义调整能力
- 32.768 kHz 外部低速晶体振荡器 – LSE，用于看门狗定时器，RTC 或系统时钟
- 32 kHz 内部低速 RC 振荡器 – LSI，用于看门狗定时器，RTC 或系统时钟
- HSE 时钟监控

功能描述

外部高速晶振 – HSE

外部高速晶体振荡器 HSE 的频率为 4 ~ 20 MHz，该振荡器可以产生高精度度的时钟源作为系统时钟使用。相关硬件配置如下图所示。一个特定频率的晶体需连接到两个 HSE 引脚 XTALIN / XTALOUT，并靠近它们。为了确保振荡频率的正确，需连接外部电阻和电容元件。

为提高晶体振荡电路 PCB 布局的稳定性，建议遵守以下指南：

- 晶体振荡器应尽可能靠近 MCU，从而尽量缩短线路长度，减少寄生电容。
- 将在晶振附近的遮罩线连接到一个接地层，实现隔绝信号与降低噪声。
- 频率切换信号线应远离晶振区域，防止串扰。

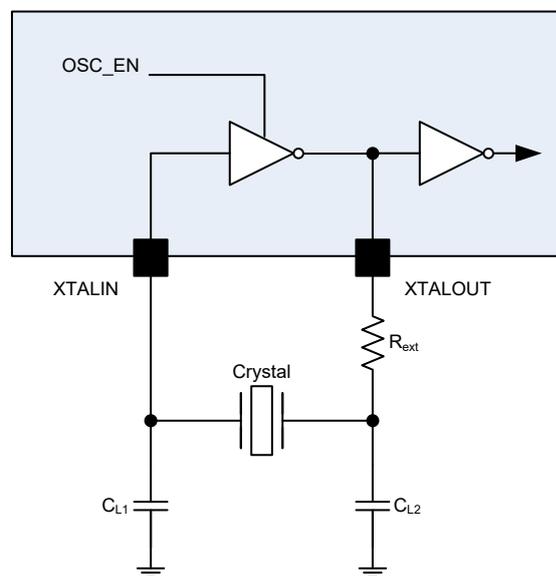


图 14. HSE 外部晶体、陶瓷和谐振器

HSE 晶体振荡器可以使用全局时钟控制寄存器 GCCR 中的 HSEEN 位开启或关闭。全局时钟状态寄存器 GCSR 中的 HSERDY 标志位用于显示外部高速晶体振荡器是否稳定。当 HSE 开启时, 直到 HSERDY 位由硬件置位, HSE 时钟才会被释放来使用。这个具体的延迟时间被称为振荡器的“启动时间”。HSE 时钟可以直接用作系统时钟源。

内部高速 RC 振荡器 – HSI

内部高速 RC 振荡器 HSI 是单片机上电时 CPU 的默认时钟源选择。HSI 振荡器提供了一个较低成本类型的时钟源, 而不需外部元件。HSI RC 振荡器可以使用全局时钟控制寄存器 GCCR 中的 HSIEN 位开启或关闭。全局时钟状态寄存器 GCSR 中的 HSIRDY 标志位用来指出内部 RC 振荡器是否稳定。HSI 振荡器的启动时间比 HSE 晶体振荡器短。

HSI 的频率精度可通过配置选项进行校准, 但其工作频率仍不如 HSE 准确。可依据应用需求、环境和成本选择使用哪种振荡器类型。

通过软件可以设置 PSRCEN 位 (省电唤醒 RC 时钟使能位) 为 1, 以迫使在系统从深度休眠中唤醒时使用 HSI 时钟作为系统时钟。随后, 当初时钟源就绪标志位被置位, 系统时钟会自动切换回初始时钟源。当使用 HSE 作为系统时钟时, 使用此功能可减少唤醒时间。

内部高速 RC 振荡器 (HSI) 自动微调

由于芯片制成工艺的不同, 每个芯片的内部高速 RC 振荡器 HSI 的频率精度都可能不同, 故每个单片机都由 Holtek 在工厂校准为以下规格: 在 $V_{DD} = 5\text{ V}$ 且 $T_A = 25\text{ }^\circ\text{C}$ 时精度为 $\pm 2\%$ 。但这种精度无法满足某些应用和环境的需求。因此, 该系列单片机提供了 HSI 微调功能, 用户可使用更精确的外部参考时钟对 HSI 频率校准。详细的方框图见图 15。

复位后, 工厂调整值载入 HSICOARSE[4:0] 和 HSI 控制寄存器 HSICR 中的 HSIFINE[7:0] 字段中。HSI 的频率精度可受电压或温度变化的影响。若应用需要一个更精确的 HSI 频率来驱动, 可使用 HSI 控制寄存器 HSICR 中的 HSIFINE[7:0] 字段手动调整, 也可通过自动调整控制器结合应用的外部参考时钟进行自动调整。参考时钟可由一个频率为 32768 Hz 的低速外部晶体 / 陶瓷谐振振荡器 LSE 或外部 CKIN 引脚的 1 kHz 脉冲提供。

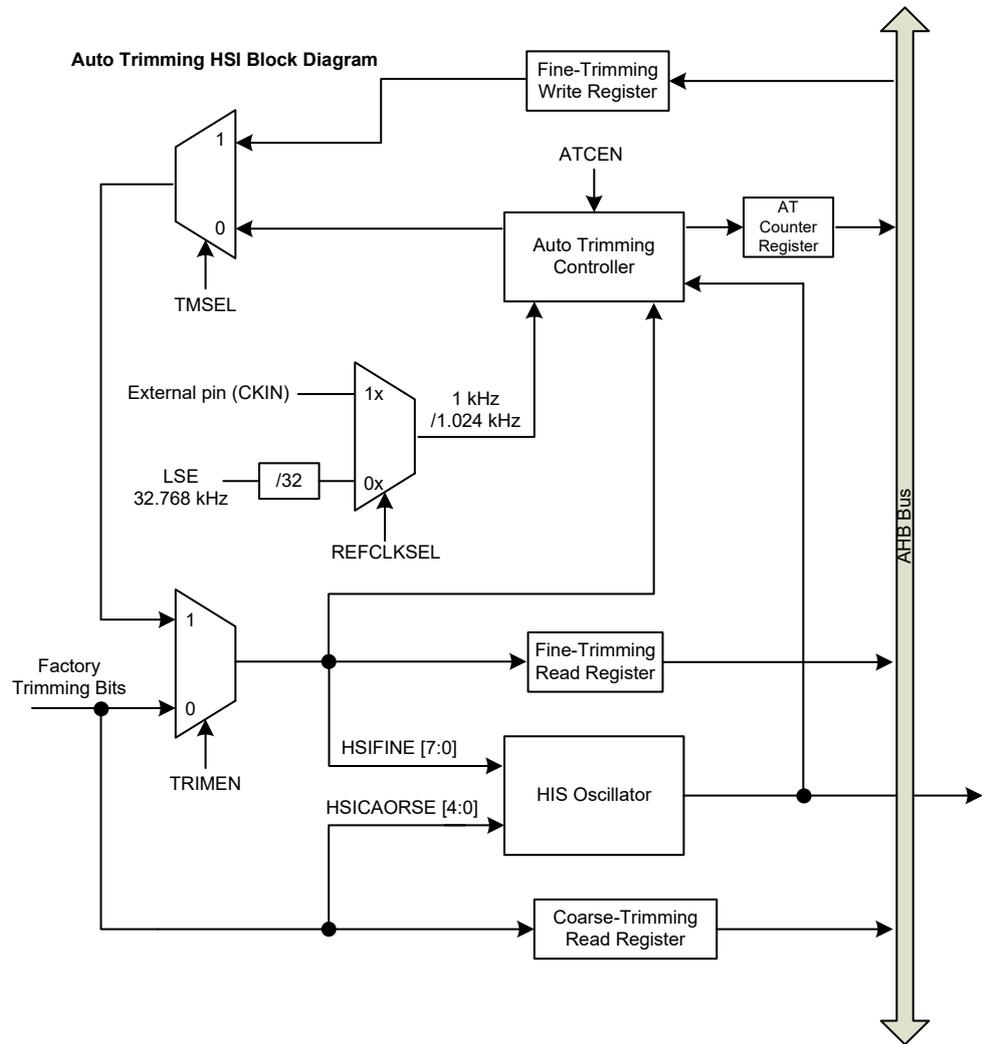


图 15. HSI 自动调整方框图

外部低速晶振 – LSE

外部低速晶体 / 陶瓷谐振振荡器，频率为 32768 Hz，可以产生一个低功率但高精确度的时钟源用于实时时钟外设，看门狗定时器或系统时钟。相关的硬件配置如下图所示。晶体或陶瓷谐振器必须靠近两个 LSE 引脚，X32KIN 和 X32KOUT。为达到适当的振荡，其外部电阻和电容元件是必要的。LSE 振荡器可以使用 RTC 控制寄存器 RTCCR 中的 LSEEN 位开启或关闭。全局时钟状态寄存器 GCSR 中的 LSERDY 标志位将指出 LSE 时钟是否稳定。

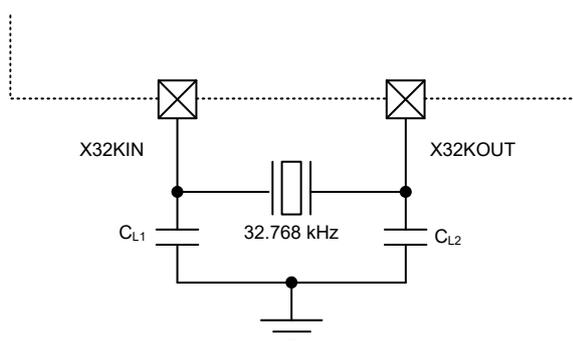


图 16. LSE 外部晶体、陶瓷和谐振器

内部低速 RC 振荡器 – LSI

内部低速 RC 振荡器的频率约为 32 kHz，是一个低功耗的时钟源用于实时时钟外设，看门狗定时器或系统时钟。LSI 提供了一个低成本的时钟源，不需外部元件。LSI 频率精度可在数据手册 (Datasheet) 中查看。全局时钟状态寄存器 GCSR 中的 LSIRDY 标志位表示 LSI 时钟是否稳定。

时钟就绪标志位

CKCU 为 HSI、HSE、LSI 和 LSE 提供相关时钟就绪标志位，用以表示这些时钟是否稳定。在用户选择这些时钟作为系统时钟或其它用途之前，必须保证指定的时钟标志位已置位。可以通过轮询 GCSR 寄存器内的各自的时钟就绪状态位来检测指定的时钟是否就绪。

系统时钟 (CK_SYS) 选择

系统复位后，默认的 CK_SYS 源为 HSI，通过改变全局时钟控制寄存器 GCCR 中的系统时钟切换位 SW 可以切换为 HSI、HSE、LSI 或 LSE。当 SW 值改变，CK_SYS 将使用初始时钟源继续运行，直到目标时钟源稳定。全局时钟状态寄存器 GCSR 中的相应的时钟就绪状态位将显示所选的时钟是否已就绪，时钟源状态寄存器 CKST 则用于显示当前作为系统时钟源的时钟。关于时钟使能功能的描述如下。

当下列任一事件发生，HSI 始终处于使能状态：

- 使能时钟监测器。(CKMEN)
- 配置时钟切换寄存器位切换到 HSI。(SW)
- 配置 HSI 使能寄存器位为 1。(HSIEN)

当下列任一事件发生，HSE 始终处于使能状态：

- 配置时钟切换寄存器位切换到 HSE。(SW)
- 配置 HSE 使能寄存器位为 1。(HSEEN)

系统时钟选择编程指南列举如下：

- 使能将作为系统时钟的任一时钟源。
- 在时钟源的相应就绪标志位置位后配置 SW 寄存器以改变系统时钟源。需注意若时钟监测器使能或配置为系统时钟的 HSE 时钟固定在 0 或 1 时，系统时钟源将强制变为 HSI。

HSE 时钟监控

HSE 时钟监控功能由全局时钟控制寄存器 GCCR 中的 HSE 时钟监控使能位 CKMEN 开启。此功能应该在 HSE 振荡器启动延迟后打开，在 HSE 停止时关闭。一旦检测到 HSE 故障，HSE 将自动除能。如果全局时钟中断寄存器 GCIR 中的时钟故障中断使能位 CKSIE 被置位，全局时钟中断寄存器 GCIR 中的 HSE 时钟故障标志位 CKSF 将被置位，并产生 HSE 故障事件。这种故障中断被连接到 CPU 的非可屏蔽中断 NMI 中。如果选择 HSE 作为 CK_SYS 时钟源，HSE 故障将迫使 CK_SYS 源通过硬件自动切换到 HSI。

时钟输出能力

该系列单片机具有时钟输出能力，允许时钟从专门的外部输出引脚 CKOUT 输出。相应的 GPIO 引脚应该被配置在适当的复用功能 I/O (AFIO) 模式以输出所选的时钟信号。有七种时钟信号可通过全局时钟配置寄存器 GCFGR 中的 CKOUT 时钟源选择位 CKOUTSRC 来选择。

表 14. CKOUT 的时钟源

CKOUTSRC[2:0]	时钟源
000	$CK_REF = CK_SYS / (CKREFPRE + 1) / 2$
001	HCLKC / 16
010	CK_SYS / 16
011	CK_HSE / 16
100	CK_HSI / 16
101	CK_LSE
110	CK_LSI

寄存器列表

下表显示了CKCU 寄存器及其复位值。

表 15. CKCU 寄存器列表

寄存器	偏移量	描述	复位值
GCFGR	0x000	全局时钟配置寄存器	0x0000_0002
GCCR	0x004	全局时钟控制寄存器	0x0000_0803
GCSR	0x008	全局时钟状态寄存器	0x0000_0028
GCIR	0x00C	全局时钟中断寄存器	0x0000_0000
AHBCFGR	0x020	AHB 配置寄存器	0x0000_0001
AHBCCR	0x024	AHB 时钟控制寄存器	0x0000_0065
APBCFGR	0x028	APB 配置寄存器	0x0001_0000
APBCCR0	0x02C	APB 时钟控制寄存器 0	0x0000_0000
APBCCR1	0x030	APB 时钟控制寄存器 1	0x0000_0000
CKST	0x034	时钟源状态寄存器	0x0100_0003
APBPCR0	0x038	APB 外设时钟选择寄存器 0	0x0000_0000
APBPCR1	0x03C	APB 外设时钟选择寄存器 1	0x0000_0000
HSICR	0x040	HSI 控制寄存器	0xXXXX_0000 X 表示未定义
HSIATCR	0x044	HSI 自动微调计数寄存器	0x0000_0000
APBPCR2	0x048	APB 外设时钟选择寄存器 2	0x0000_0000
MCUDBGCR	0x304	MCU 调试控制寄存器	0x0000_0000

寄存器描述

全局时钟配置寄存器 – GCFGR

该寄存器用于为定义低功耗模式和 CKOUT 电路指定时钟源。

偏移量: 0x000

复位值: 0x0000_0002

	31	30	29	28	27	26	25	24
	LPMOD				保留位			
类型 / 复位	RO	0	RO	0	RO	0		
	23	22	21	20	19	18	17	16
	保留位							
	15	14	13	12	11	10	9	8
	CKREFPRE				保留位			
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0
	保留位					CKOUTSRC		
类型 / 复位						RW	0	RW
							1	RW
								0

位	字段	描述
[31:29]	LPMOD	低功耗模式状态 000: 单片机处于运行模式 001: 单片机要进入休眠模式 010: 单片机要进入深度休眠模式 1 011: 单片机要进入深度休眠模式 2 其它: 保留 由硬件置位和复位。
[15:11]	CKREFPRE	CK_REF 时钟预分频选择 $CK_REF = CK_SYS / (CKREFPRE + 1) / 2$ 00000: $CK_REF = CK_SYS / 2$ 00001: $CK_REF = CK_SYS / 4$... 11111: $CK_REF = CK_SYS / 64$ 由软件设置此字段来控制 CK_REF 时钟预分频值。
[2:0]	CKOUTSRC	CKOUT 时钟源选择位 000: $CK_REF, CK_REF = CK_SYS / (CKREFPRE + 1) / 2$ 001: HCLKC / 16 010: CK_SYS / 16 011: CK_HSE / 16 100: CK_HSI / 16 101: CK_LSE 110: CK_LSI 111: 保留 这些位由软件置位和复位

全局时钟控制寄存器 – GCCR

该寄存器定义了时钟使能位。

偏移量: 0x004

复位值: 0x0000_0803

31	30	29	28	27	26	25	24
保留位							
23	22	21	20	19	18	17	16
保留位						PSRCEN	CKMEN
						RW	0 RW 0
15	14	13	12	11	10	9	8
保留位				HSIEN	HSEEN	保留位	HSEGAIN
				RW	1 RW 0		RW 0
7	6	5	4	3	2	1	0
保留位					SW		
					RW	0 RW	1 RW 1

位	字段	描述
[17]	PSRCEN	省电唤醒 RC 时钟使能位 0: 无动作 1: 从深度休眠 1/2 模式唤醒后, 使用内部 RC 时钟 (HSI) 作为系统时钟。 在进入深度休眠 1/2 模式之前, 软件可以设置 PSRCEN 位为高来减少唤醒等待时间。 当 PSRCEN = 1, 在系统从深度休眠 1/2 模式唤醒后将先选择 HSI 作为系统时钟源。因为 HSI 时钟提供给 CPU, 可以在初始时钟 CK_SYS 源稳定前执行指令。在初始时钟源 CK_SYS 就绪后, CK_SYS 将自动从 HSI 时钟切换到初始时钟源。
[16]	CKMEN	HSE 时钟监控使能位 0: 外部晶振 HSE 时钟监控除能 1: 外部晶振 HSE 时钟监控使能 当硬件检测出 HSE 时钟停留在低电平或高电平状态, 内部硬件将切换内部高速 RC 时钟 HSI 为系统时钟。恢复系统时钟的唯一方法是通过外部复位, 电源复位或通过软件清除 CKSF。
[11]	HSIEN	内部高速振荡器使能位 0: 内部 RC 振荡器除能 1: 内部 RC 振荡器使能 由软件置位和复位。如果使用 HSI 作为系统时钟, 该位不能被复位。
[10]	HSEEN	外部高速振荡器使能位 0: 外部晶振除能 1: 外部晶振使能 由软件置位和复位。如果使用 HSE 作为系统时钟, 该位不能被复位。
[8]	HSEGAIN	外部高速振荡器增益选择 0: HSE 在低增益模式 1: HSE 在高增益模式

位	字段	描述
[2:0]	SW	<p>系统时钟切换</p> <p>010: 选择 CK_HSE 作为系统时钟 011: 选择 CK_HSI 作为系统时钟 110: 选择 CK_LSE 作为系统时钟 111: 选择 CK_LSI 作为系统时钟 其它: CK_HSI 作为系统时钟</p> <p>这些位由软件置位和复位, 其用来选择 CK_SYS 源。如果直接或间接使用 HSE 振荡器作为系统时钟, 当 HSE 时钟监控使能, 如果 HSE 时钟故障被检测到, 则硬件将这些位设置为 b011, HSI 振荡器将强制作为系统时钟。</p> <p>注: 当使用 SW 切换系统时钟时, 系统时钟不会立即被切换, 会有一定的延迟时间。系统时钟的选择由 SW 位控制, 可通过软件监控寄存器 CKSTR 中的 CKSWST 位来确定当前所用的系统时钟。</p>

全局时钟状态寄存器 – GCSR

该寄存器定义了时钟就绪状态。

偏移量: 0x008

复位值: 0x0000_0028

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位								
	15	14	13	12	11	10	9	8	
类型 / 复位	保留位								
	7	6	5	4	3	2	1	0	
类型 / 复位		保留位	LSIRDY	LSERDY	HSIRDY	HSERDY		保留位	
			RO	1 RO	0 RO	1 RO	0		

位	字段	描述
[5]	LSIRDY	<p>内部低速振荡器就绪标志位。</p> <p>0: 内部 32 kHz RC 振荡器未就绪 1: 内部 32 kHz RC 振荡器就绪</p> <p>由硬件置位以表示 LSI 振荡器已稳定且准备就绪。</p>
[4]	LSERDY	<p>外部低速振荡器 LSE 就绪标志位</p> <p>0: 外部 32768 Hz 晶体振荡器未就绪 1: 外部 32768 Hz 晶体振荡器就绪</p> <p>由硬件置位以表示 LSE 振荡器已稳定且准备就绪。</p>
[3]	HSIRDY	<p>内部高速振荡器就绪标志位</p> <p>0: 内部 RC 振荡器未就绪 1: 内部 RC 振荡器就绪</p> <p>由硬件置位以表示 HSI 振荡器已稳定且准备就绪。</p>
[2]	HSERDY	<p>外部高速振荡器就绪标志位</p> <p>0: 外部振荡器未就绪 1: 外部振荡器就绪</p> <p>由硬件置位以表示 HSE 振荡器已稳定且准备就绪。</p>

全局时钟中断寄存器 – GCIR

该寄存器定义了中断使能位和标志位。

偏移量: 0x00C

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位							CKSIE	
									RW 0
	15	14	13	12	11	10	9	8	
类型 / 复位	保留位								
	7	6	5	4	3	2	1	0	
类型 / 复位	保留位							CKSF	
									WC 0

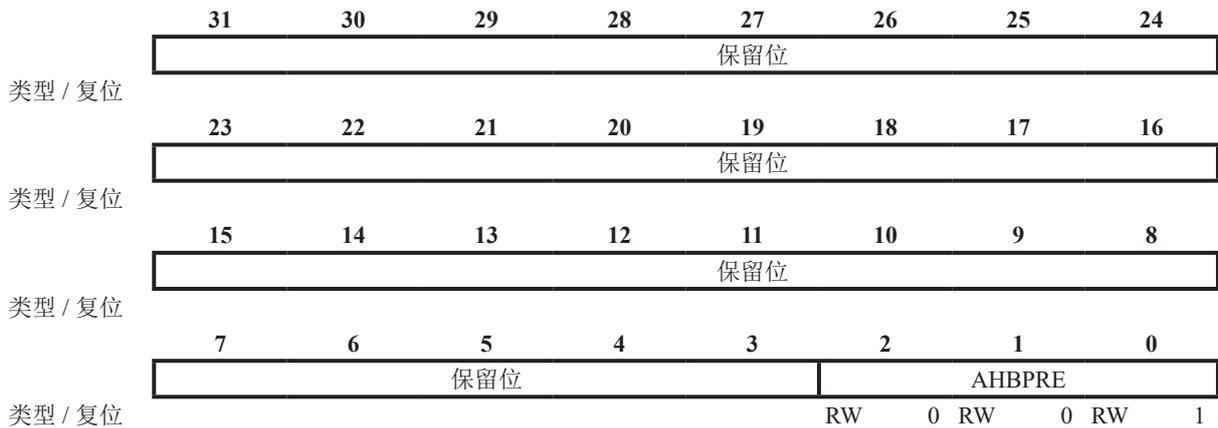
位	字段	描述
[16]	CKSIE	时钟故障中断使能位 0: 时钟故障中断除能 1: 时钟故障中断使能 该位由软件置位和复位来使能或除能时钟监控中断。
[0]	CKSF	HSE 时钟故障中断标志位 0: 时钟正常运行 1: HSE 时钟故障 通过写 1 来清零。当 HSE 时钟故障且 CKSIE 位为 1 时，硬件将其置位。

AHB 配置寄存器 – AHBCFGR

该寄存器定义了系统时钟频率。

偏移量: 0x020

复位值: 0x0000_0001



位	字段	描述
[2:0]	AHBPRE	AHB 预分频 000: CK_AHB = CK_SYS 001: CK_AHB = CK_SYS / 2 010: CK_AHB = CK_SYS / 4 011: CK_AHB = CK_SYS / 8 100: CK_AHB = CK_SYS / 16 101: CK_AHB = CK_SYS / 32 110: CK_AHB = CK_SYS / 32 111: CK_AHB = CK_SYS / 32 由软件设置此位来控制 AHB 时钟的分频比。

AHB 时钟控制寄存器 – AHBCCR

该寄存器定义了 AHB 时钟使能位。

偏移量: 0x024

复位值: 0x0000_0065

	31	30	29	28	27	26	25	24
	保留位							DIVEN
类型 / 复位								RW 0
	23	22	21	20	19	18	17	16
	保留位					PCEN	PBEN	PAEN
类型 / 复位						RW 0	RW 0	RW 0
	15	14	13	12	11	10	9	8
	保留位				CKREFEN	保留位		
类型 / 复位					RW 0			
	7	6	5	4	3	2	1	0
	保留位	APBEN	BMEN	保留位		SRAMEN	保留位	FMCEN
类型 / 复位		RW 1	RW 1			RW 1		RW 1

位	字段	描述
[24]	DIVEN	除法器时钟使能位 0: 除法器时钟除能 1: 除法器时钟使能 该位由软件置位和复位。
[18]	PCEN	通用 I/O 口 PC 时钟使能位 0: PC 时钟除能 1: PC 时钟使能 该位由软件置位和复位。
[17]	PBEN	通用 I/O 口 PB 时钟使能位 0: PB 时钟除能 1: PB 时钟使能 该位由软件置位和复位。
[16]	PAEN	通用 I/O 口 PA 时钟使能位 0: PA 时钟除能 1: PA 时钟使能 该位由软件置位和复位。
[11]	CKREFEN	CK_REF 时钟使能位 0: CK_REF 时钟除能 1: CK_REF 时钟使能 该位由软件置位和复位。
[6]	APBEN	APB 桥时钟使能位 0: 在休眠模式下 APB 桥时钟由硬件自动除能 1: 在休眠模式下 APB 桥时钟始终使能 由软件置位和复位。如果 APB 桥在休眠模式下未使用，用户可将 APBEN 位清零以降低功耗。
[5]	BMEN	总线矩阵时钟使能位 0: 在休眠模式下总线矩阵时钟由硬件自动除能 1: 在休眠模式下总线矩阵时钟始终使能 由软件置位和复位。如果总线矩阵桥在休眠模式下未使用，用户可将 BMEN 位清零以降低功耗。

位	字段	描述
[2]	SRAMEN	SRAM 时钟使能位 0: 在休眠模式下 SRAM 时钟由硬件自动除能 1: 在休眠模式下 SRAM 时钟始终使能 由软件置位和复位。如果 SRAM 在休眠模式下未使用，用户可将 SRAMEN 位清零以降低功耗。
[0]	FMCEN	Flash 存储控制器时钟使能位 0: FMC 时钟在休眠模式下由硬件自动除能 1: FMC 时钟在休眠模式下始终使能 由软件置位和复位。如果 Flash 存储在休眠模式下未使用，用户可将 FMCEN 位清零以降低功耗。

APB 配置寄存器 – APBCFGR

该寄存器定义了 A/D 转换器时钟频率。

偏移量: 0x028
复位值: 0x0001_0000

	31	30	29	28	27	26	25	24			
类型 / 复位	保留位										
	23	22	21	20	19	18	17	16			
类型 / 复位	保留位					ADCDIV					
						RW	0	RW	0	RW	1
	15	14	13	12	11	10	9	8			
类型 / 复位	保留位										
	7	6	5	4	3	2	1	0			
类型 / 复位	保留位										

位	字段	描述
[18:16]	ADCDIV	A/D 转换器时钟分频选择位 000: CK_ADC = CK_AHB / 1 001: CK_ADC = CK_AHB / 2 010: CK_ADC = CK_AHB / 4 011: CK_ADC = CK_AHB / 8 100: CK_ADC = CK_AHB / 16 101: CK_ADC = CK_AHB / 32 110: CK_ADC = CK_AHB / 64 111: CK_ADC = CK_AHB / 3 由软件设置此位来选择 A/D 转换器时钟分频比。

APB 时钟控制寄存器 0 – APBCCR0

该寄存器具体定义了 APB 外设时钟使能位。

偏移量: 0x02C

复位值: 0x0000_0000

31	30	29	28	27	26	25	24	
保留位								
23	22	21	20	19	18	17	16	
保留位								
15	14	13	12	11	10	9	8	
EXTIEN	AFIOEN	保留位		UR1EN	UR0EN	保留位		
RW	0	RW	0	RW	0	RW	0	
7	6	5	4	3	2	1	0	
保留位		SPI1EN	SPI0EN	保留位			I2CEN	
		RW	0	RW	0	RW		0

位	字段	描述
[15]	EXTIEN	外部中断时钟使能位 0: EXTI 时钟除能 1: EXTI 时钟使能 由软件置位和复位。
[14]	AFIOEN	复用功能 I/O 时钟使能位 0: AFIO 时钟除能 1: AFIO 时钟使能 由软件置位和复位。
[11]	UR1EN	UART1 时钟使能位 0: UART1 时钟除能 1: UART1 时钟使能 由软件置位和复位。
[10]	UR0EN	UART0 时钟使能位 0: UART0 时钟除能 1: UART0 时钟使能 由软件置位和复位。
[5]	SPI1EN	SPI1 时钟使能位 0: SPI1 时钟除能 1: SPI1 时钟使能 由软件置位和复位。
[4]	SPI0EN	SPI0 时钟使能位 0: SPI0 时钟除能 1: SPI0 时钟使能 由软件置位和复位。
[0]	I2CEN	I ² C 时钟使能位 0: I ² C 时钟除能 1: I ² C 时钟使能 由软件置位和复位。

APB 时钟控制寄存器 1 – APBCCR1

该寄存器定义了各种 APB 外设时钟使能位。

偏移量: 0x030
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
	保留位								ADCCEN
类型 / 复位									RW 0
	23	22	21	20	19	18	17	16	
	保留位								BFTMEN
类型 / 复位									RW 0
	15	14	13	12	11	10	9	8	
	保留位		PWM1	PWM0	保留位			GPTMEN	
类型 / 复位			RW 0	RW 0				RW 0	
	7	6	5	4	3	2	1	0	
	保留位	VDDREN	保留位	WDTREN	保留位				
类型 / 复位		RW 0		RW 0					

位	字段	描述
[24]	ADCCEN	ADC 时钟使能位 0: ADC 时钟除能 1: ADC 时钟使能 由软件置位和复位。
[16]	BFTMEN	BFTM 时钟使能位 0: BFTM 时钟除能 1: BFTM 时钟使能 由软件置位和复位。
[13]	PWM1EN	PWM1 时钟使能位 0: PWM1 时钟除能 1: PWM1 时钟使能 由软件置位和复位。
[12]	PWM0EN	PWM0 时钟使能位 0: PWM0 时钟除能 1: PWM0 时钟使能 由软件置位和复位。
[8]	GPTMEN	GPTM 时钟使能位 0: GPTM 时钟除能 1: GPTM 时钟使能 由软件置位和复位。
[6]	VDDREN	V _{DD} 域寄存器存取时钟使能位 0: V _{DD} 域寄存器存取时钟除能 1: V _{DD} 域寄存器存取时钟使能 由软件置位和复位。
[4]	WDTREN	看门狗定时器寄存器存取时钟使能位 0: 看门狗定时器寄存器存取时钟除能 1: 看门狗定时器寄存器存取时钟使能 由软件置位和复位。

时钟源状态寄存器 – CKST

该寄存器定义了时钟源状态。

偏移量: 0x034

复位值: 0x0100_0003

	31	30	29	28	27	26	25	24			
	保留位					HSIST					
类型 / 复位						RO	0	RO	0	RO	1
	23	22	21	20	19	18	17	16			
	保留位					HSEST					
类型 / 复位						RO	0	RO	0		
	15	14	13	12	11	10	9	8			
	保留位										
类型 / 复位											
	7	6	5	4	3	2	1	0			
	保留位					CKSWST					
类型 / 复位						RO	0	RO	1	RO	1

位	字段	描述
[26:24]	HSIST	内部高速时钟 CK_HSI 占用状态位 xx1: HSI 用于系统时钟 CK_SYS (SW = 0x3) x1x: 保留 1xx: HSI 用于时钟监控
[17:16]	HSEST	外部高速时钟 CK_HSE 占用状态位 x1: HSE 用于系统时钟 CK_SYS (SW = 0x2) 1x: 保留
[2:0]	CKSWST	时钟切换状态位 00x: 保留 010: CK_HSE 用于系统时钟 011: CK_HSI 用于系统时钟 110: CK_LSE 用于系统时钟 111: CK_LSI 用于系统时钟 该字段为状态位用来表示哪一个时钟源作为当前的系统时钟。

APB 外设时钟选择寄存器 0 – APBPCSR0

该寄存器定义了 APB 外设时钟分频选择。

偏移量: 0x038
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24
	UR1PCLK		UR0PCLK		保留位			
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	23	22	21	20	19	18	17	16
	保留位		GPTMPCLK		保留位			
类型 / 复位			RW	0	RW	0		
	15	14	13	12	11	10	9	8
	保留位		BFTMPCLK		保留位			
类型 / 复位			RW	0	RW	0		
	7	6	5	4	3	2	1	0
	SPI1PCLK		SPI0PCLK		保留位		I2CPCLK	
类型 / 复位	RW	0	RW	0	RW	0	RW	0

位	字段	描述
[31:30]	UR1PCLK	UART1 外设时钟选择 00: PCLK = CK_AHB 01: PCLK = CK_AHB / 2 10: PCLK = CK_AHB / 4 11: PCLK = CK_AHB / 8 PCLK = 外设时钟; CK_AHB = AHB 和 CPU 时钟
[29:28]	UR0PCLK	UART0 外设时钟选择 00: PCLK = CK_AHB 01: PCLK = CK_AHB / 2 10: PCLK = CK_AHB / 4 11: PCLK = CK_AHB / 8 PCLK = 外设时钟; CK_AHB = AHB 和 CPU 时钟
[21:20]	GPTMPCLK	GPTM 外设时钟选择 00: PCLK = CK_AHB 01: PCLK = CK_AHB / 2 10: PCLK = CK_AHB / 4 11: PCLK = CK_AHB / 8 PCLK = 外设时钟; CK_AHB = AHB 和 CPU 时钟
[13:12]	BFTMPCLK	BFTM 外设时钟选择 00: PCLK = CK_AHB 01: PCLK = CK_AHB / 2 10: PCLK = CK_AHB / 4 11: PCLK = CK_AHB / 8 PCLK = 外设时钟; CK_AHB = AHB 和 CPU 时钟
[7:6]	SPI1PCLK	SPI1 外设时钟选择 00: PCLK = CK_AHB 01: PCLK = CK_AHB / 2 10: PCLK = CK_AHB / 4 11: PCLK = CK_AHB / 8 PCLK = 外设时钟; CK_AHB = AHB 和 CPU 时钟

位	字段	描述
[5:4]	SPI0PCLK	SPI0 外设时钟选择 00: PCLK = CK_AHB 01: PCLK = CK_AHB / 2 10: PCLK = CK_AHB / 4 11: PCLK = CK_AHB / 8 PCLK = 外设时钟; CK_AHB = AHB 和 CPU 时钟
[1:0]	I2CPCLK	I ² C 外设时钟选择 00: PCLK = CK_AHB 01: PCLK = CK_AHB / 2 10: PCLK = CK_AHB / 4 11: PCLK = CK_AHB / 8 PCLK = 外设时钟; CK_AHB = AHB 和 CPU 时钟

APB 外设时钟选择寄存器 1 – APBPCSR1

该寄存器定义了 APB 外设时钟分频选择。

偏移量: 0x03C
复位值: 0x0000_0000

31	30	29	28	27	26	25	24
保留位							
23	22	21	20	19	18	17	16
保留位							
15	14	13	12	11	10	9	8
VDDRCLK		WDTRCLK		保留位			
RW	0	RW	0	RW	0	RW	0
7	6	5	4	3	2	1	0
保留位		ADCCCLK		EXTICLK		AFIOPCLK	
		RW	0	RW	0	RW	0

位	字段	描述
[15:14]	VDDRCLK	V _{DD} 域寄存器存取时钟选择 00: PCLK = CK_AHB / 4 01: PCLK = CK_AHB / 8 10: PCLK = CK_AHB / 16 11: PCLK = CK_AHB / 32 PCLK = 外设时钟; CK_AHB = AHB 和 CPU 时钟
[13:12]	WDTRCLK	WDT 寄存器访问时钟选择 00: PCLK = CK_AHB 01: PCLK = CK_AHB / 2 10: PCLK = CK_AHB / 4 11: PCLK = CK_AHB / 8 PCLK = 外设时钟; CK_AHB = AHB 和 CPU 时钟
[5:4]	ADCCCLK	ADC 控制器外设时钟选择 00: PCLK = CK_AHB 01: PCLK = CK_AHB / 2 10: PCLK = CK_AHB / 4 11: PCLK = CK_AHB / 8 PCLK = 外设时钟; CK_AHB = AHB 和 CPU 时钟
[3:2]	EXTICLK	EXTI 外设时钟选择 00: PCLK = CK_AHB 01: PCLK = CK_AHB / 2 10: PCLK = CK_AHB / 4 11: PCLK = CK_AHB / 8 PCLK = 外设时钟; CK_AHB = AHB 和 CPU 时钟
[1:0]	AFIOPCLK	AFIO 外设时钟选择 00: PCLK = CK_AHB 01: PCLK = CK_AHB / 2 10: PCLK = CK_AHB / 4 11: PCLK = CK_AHB / 8 PCLK = 外设时钟; CK_AHB = AHB 和 CPU 时钟

HSI 控制寄存器 – HSICR

该寄存器用来控制 HSI RC 振荡器频率微调。

偏移量: 0x040

复位值: 0xXXXX_0000, X 表示未定义

	31	30	29	28	27	26	25	24	
	保留位			HSICOARSE					
类型 / 复位				RO	X RO	X RO	X RO	X RO	
	23	22	21	20	19	18	17	16	
	HSIFINE								
类型 / 复位	RW	X RW	X RW	X RW	X RW	X RW	X RW	X RW	
	15	14	13	12	11	10	9	8	
	保留位								
类型 / 复位									
	7	6	5	4	3	2	1	0	
	FLOCK	REFCLKSEL		TMSEL	ATMSEL	LTRSEL	ATCEN	TRIMEN	
类型 / 复位	RO	0 RW	0 RW	0 RW	0 RW	0 RW	0 RW	0	

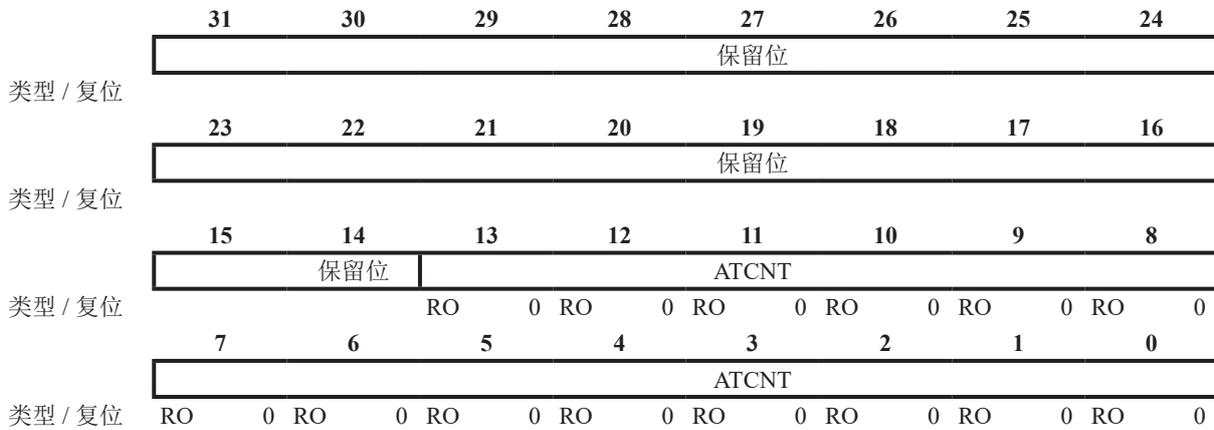
位	字段	描述
[28:24]	HSICOARSE	HSI 时钟粗调值 开机时, 这些位自动初始化。该值仅由工厂调整, 不能通过程序调整。
[23:16]	HSIFINE	HSI 时钟微调值 开机时, 这些位自动初始化。该值也由工厂调整。但这些位也支持用户通过程序微调, 微调值将与 HSICOARSE[4:0] 位共同作用, 对电压或温度变化进行补偿, 获取更加高的 HSI 频率精度。它可由软件编程或由带外部参考时钟的自动微调控制器 (ATC) 自动调整。
[7]	FLOCK	时钟频率 0: HSI 频率没有微调到目标范围 1: HSI 频率微调到目标范围
[6:5]	REFCLKSEL	参考时钟选择 0x: 选择 32.768 kHz 外部低速时钟源 (LSE) 1x: 选择外部 CKIN 引脚 1 kHz 脉冲 该位为 HSI 自动微调控制器 (ATC) 选择参考时钟源
[4]	TMSEL	微调模式选择 0: 由自动微调控制器微调 1: 由用户手动微调 该位用来选择 HSI RC 振荡器微调功能是通过 ATC 硬件或用户编程 HSI 控制寄存器中的 HSIFINE[7:0] 位来微调。
[3]	ATMSEL	自动微调模式选择 0: 自动微调控制器采用二分查找法逼近目标范围 1: 自动微调控制器采用线性查找法逼近目标范围 此位用于选择 HSI RC 振荡器微调功能的 ATC 硬件自动微调模式。
[2]	LTRSEL	频率锁定目标范围选择 0: 0.1 % 的变化范围 1: 0.2 % 的变化范围 此位用于选择内部 HSI RC 振荡器调整功能的频率锁定目标范围为 0.1 % 或 0.2 %。
[1]	ATCEN	自动微调控制器 (ATC) 使能 0: 除能自动微调控制器 1: 使能自动微调控制器

位	字段	描述
[0]	TRIMEN	微调使能 0: HSI 微调除能 1: HSI 微调使能 该位使能 HSI RC 振荡器微调功能，可由 ATC 硬件或用户编程微调。

HSI 自动微调计数器寄存器 – HSIATCR

该寄存器包含了 HSI 自动微调控制器的计数值。

偏移量: 0x044
 复位值: 0x0000_0000



位	字段	描述
[13:0]	ATCNT	自动微调计数器 这些位包含了 HSI 自动微调控制器的计数器值。

APB 外设时钟选择寄存器 2 – APBPCSR2

该寄存器包含了 HSI 自动微调控制器的计数值。

偏移量: 0x044

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位				PWM1PCLK		PWM0PCLK		
					RW	0	RW	0	RW
	15	14	13	12	11	10	9	8	
类型 / 复位	保留位								
	7	6	5	4	3	2	1	0	
类型 / 复位	保留位								

位	字段	描述
[19:18]	PWM1PCLK	PWM1 外设时钟选择 00: PCLK = CK_AHB 01: PCLK = CK_AHB / 2 10: PCLK = CK_AHB / 4 11: PCLK = CK_AHB / 8 PCLK = 外设时钟; CK_AHB = AHB 和 CPU 时钟
[17:16]	PWM0PCLK	PWM0 外设时钟选择 00: PCLK = CK_AHB 01: PCLK = CK_AHB / 2 10: PCLK = CK_AHB / 4 11: PCLK = CK_AHB / 8 PCLK = 外设时钟; CK_AHB = AHB 和 CPU 时钟

单片机调试控制寄存器 – MCUIDBGCR

该寄存器定义了单片机的调试控制。

偏移量: 0x304
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24
	DBPWM1	DBPWM0	保留位					
类型 / 复位	RW 0	RW 0						
	23	22	21	20	19	18	17	16
	保留位				DBUR1	DBUR0	保留位	DBBFTM
类型 / 复位					RW 0	RW 0		RW 0
	15	14	13	12	11	10	9	8
	保留位	DBDSLP2	保留位	DBI2C	DBSPI1	DBSPI0	保留位	
类型 / 复位		RW 0		RW 0	RW 0	RW 0		
	7	6	5	4	3	2	1	0
	保留位	DBGPTM	保留位		DBWDT	保留位	DBDSLP1	DBSLP
类型 / 复位		RW 0			RW 0		RW 0	RW 0

位	字段	描述
[31]	DBPWM1	PWM1 调试模式使能位 0: 即使内核暂停, PWM1 计数器仍继续计数 1: 当内核暂停, PWM1 计数器停止计数 由软件置位和复位。
[30]	DBPWM0	PWM0 调试模式使能位 0: 即使内核暂停, PWM0 计数器仍继续计数 1: 当内核暂停, PWM0 计数器停止计数 由软件置位和复位。
[19]	DBUR1	UART1 调试模式使能位 0: 与正常模式操作相同 1: 当内核暂停, UART1 FIFO 超时计数停止 由软件置位和复位。
[18]	DBUR0	UART0 调试模式使能位 0: 与正常模式操作相同 1: 当内核暂停, UART0 FIFO 超时计数停止 由软件置位和复位。
[16]	DBBFTM	BFTM 调试模式使能位 0: 即使内核暂停, BFTM 计数器仍继续计数 1: 当内核暂停, BFTM 计时器停止计数 由软件置位和复位。
[14]	DBDSLP2	调试深度休眠模式 2 0: 在深度休眠模式 2 下, LDO = Off, DMOS = On, FCLK = Off, HCLK = Off 1: 在深度休眠模式 2 下, LDO = On, FCLK = On, HCLK = On 由软件置位和复位。
[12]	DBI2C	I ² C 调试模式使能位 0: 与正常模式操作相同 1: 当内核暂停, I ² C 超时计数停止 由软件置位和复位。

位	字段	描述
[11]	DBSPI1	SPI1 调试模式使能位 0: 与正常模式操作相同 1: 当内核暂停, SPI1 FIFO 超时计数停止由软件置位和复位。
[10]	DBSPI0	SPI0 调试模式使能位 0: 与正常模式操作相同 1: 当内核暂停, SPI0 FIFO 超时计数停止由软件置位和复位。
[6]	DBGPTM	GPTM 调试模式使能位 0: 即使内核暂停, GPTM 计数器仍继续计数 1: 当内核暂停, GPTM 计数器停止计数由软件置位和复位。
[3]	DBWDT	看门狗定时器调试模式使能位 0: 即使内核暂停, 看门狗定时器仍继续计数 1: 当内核暂停, 看门狗定时器停止计数由软件置位和复位。
[1]	DBDSLP1	调试深度休眠模式 1 0: 在深度休眠模式 1 下, LDO = 低功耗模式, FCLK = Off, HCLK = Off 1: 在深度休眠模式 1 下, LDO = On, FCLK = On, HCLK = On 由软件置位和复位。
[0]	DBSLP	调试休眠模式 0: 在休眠模式下, LDO = On, FCLK = On, HCLK = Off 1: 在休眠模式下, LDO = On, FCLK = On, HCLK = On 由软件置位和复位。

7 复位控制单元 (RSTCU)

简介

复位控制单元 RSTCU 有三种复位方式，上电复位、系统复位和 APB 单元复位。上电复位，即冷复位，在上电时复位了整个系统。系统复位复位了处理器内核和除调试端口控制器以外的外设 IP 元件。这些复位可以通过外部信号、内部事件和复位发生器触发。有关这些复位的更多信息将在以下章节描述。

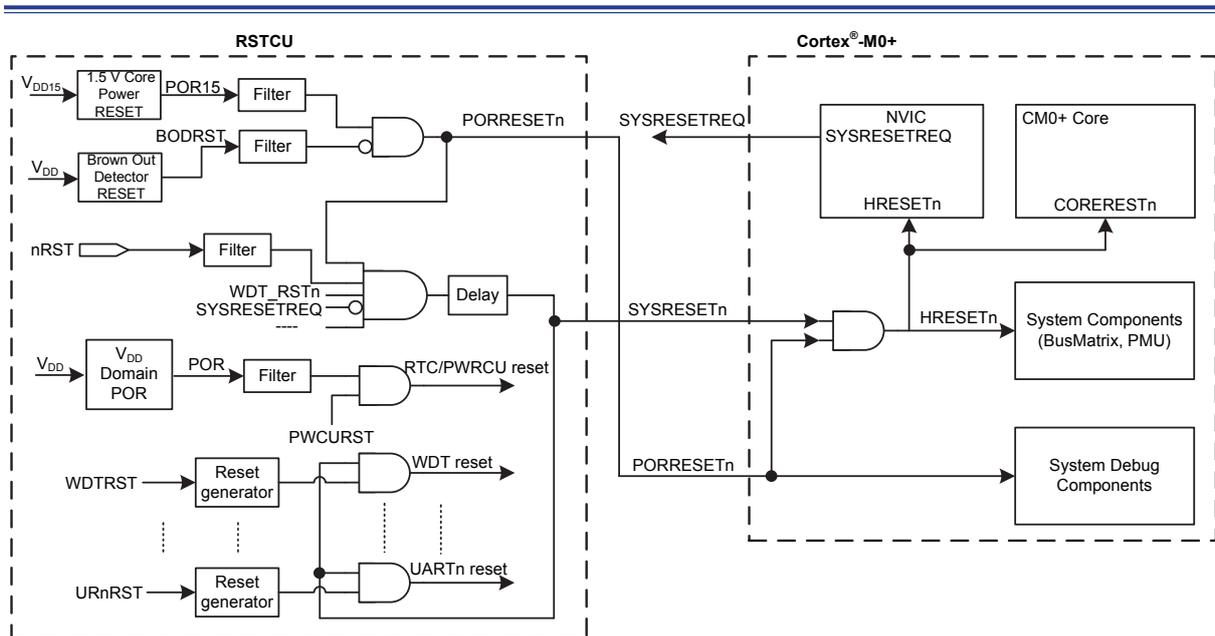


图 17. RSTCU 方框图

功能描述

上电复位

上电复位 POR 是由外部复位或内部复位发生器产生。这两种类型都有一个内部滤波器，以防止因噪声而误动作。参照图 18，POR15 信号会一直维持在低电平直到内部 LDO 稳压器准备好提供 1.5 V 电源。除了 POR15 信号，当 LVDCSR 寄存器中的 BODEN 位被置位且欠压事件发生时，电源控制单元 PWRCU 将置位 BODF 信号表明发生了掉电复位 PDR。有关 PWRCU 功能的更多细节，请参考 PWRCU 章节。

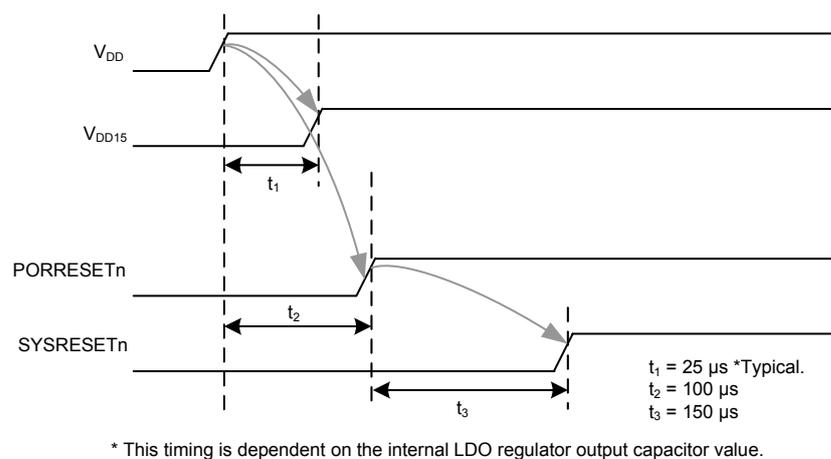


图 18. 上电复位时序图

系统复位

系统复位是由上电复位 PORRESETn、看门狗定时器复位 WDT_RSTn、nRST 引脚或软件复位 SYSRESETREQ 事件产生。有关 SYSRESETREQ 的更多信息，请参考 Cortex®-M0+ 参考手册的相关章节。

AHB 和 APB 单元复位

AHB 和 APB 单元复位可分为硬件复位和软件复位。硬件复位可以由上电复位或系统复位来对所有 AHB 和 APB 单元进行复位。连接到 AHB 和 APB 总线上的每个功能 IP 都可以通过 RSTCU 中相关的软件复位位单独复位。例如，应用软件可以通过 APBPRSTR0 寄存器中的 UR0RST 位产生一个 UART0 复位。

寄存器列表

下表显示了 RSTCU 寄存器及其复位值。

表 16. RSTCU 寄存器列表

寄存器	偏移量	描述	复位值
GRSR	0x100	全局复位状态寄存器	0x0000_0008
AHBPRSTR	0x104	AHB 外设复位寄存器	0x0000_0000
APBPRSTR0	0x108	APB 外设复位寄存器 0	0x0000_0000
APBPRSTR1	0x10C	APB 外设复位寄存器 1	0x0000_0000

寄存器描述

全局复位状态寄存器 – GRSR

该寄存器定义了复位状态的各种状况。

偏移量: 0x100

复位值: 0x0000_0008

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位								
	15	14	13	12	11	10	9	8	
类型 / 复位	保留位								
	7	6	5	4	3	2	1	0	
类型 / 复位	保留位				PORSTF	WDTRSTF	EXTRSTF	NVICRSTF	
					WC	1 WC	0 WC	0 WC	0

位	字段	描述
[3]	PORSTF	内核 1.5 V 上电复位标志位 0: 上电复位未发生 1: 上电复位发生 当上电复位发生时, 由硬件置位。通过写 1 来复位。
[2]	WDTRSTF	看门狗定时器复位标志位 0: 看门狗定时器复位未发生 1: 看门狗定时器复位发生 当看门狗定时器复位发生时, 由硬件置位。通过写 1 来复位, 或当上电复位发生时, 由硬件复位。
[1]	EXTRSTF	外部引脚复位标志位 0: 引脚复位未发生 1: 引脚复位发生 当外部引脚复位发生时, 由硬件置位。通过写 1 来复位, 或当上电复位发生时, 由硬件复位。

位	字段	描述
[0]	NVICRSTF	NVIC 复位标志位 0: NVIC 所触发的系统复位未发生 1: NVIC 所触发的系统复位发生 当系统复位发生时, 由硬件置位。通过写 1 来复位, 或当上电复位发生时, 由硬件复位。

AHB 外设复位寄存器 – AHBPRSTR

该寄存器定义了 AHB 外设的软件复位控制位。

偏移量: 0x104
复位值: 0x0000_0000

31	30	29	28	27	26	25	24			
保留位							DIVRST			
							RW	0		
23	22	21	20	19	18	17	16			
保留位										
15	14	13	12	11	10	9	8			
保留位					PCRST	PBRST	PARST			
					RW	0	RW	0	RW	0
7	6	5	4	3	2	1	0			
保留位										

位	字段	描述
[24]	DIVRST	除法器复位控制位 0: 无复位 1: 复位除法器 该位由软件置位, 由硬件自动清零。
[10]	PCRST	通用 I/O 口 PC 复位控制位 0: 无复位 1: 复位 PC 该位由软件置位, 由硬件自动清零。
[9]	PBRST	通用 I/O 口 PB 复位控制位 0: 无复位 1: 复位 PB 该位由软件置位, 由硬件自动清零。
[8]	PARST	通用 I/O 口 PA 复位控制位 0: 无复位 1: 复位 PA 该位由软件置位, 由硬件自动清零。

APB 外设复位寄存器 0 – APBPRSTR0

该寄存器定义了 APB 外设的软件复位控制位。

偏移量: 0x108
复位值: 0x0000_0000

31	30	29	28	27	26	25	24	
保留位								
23	22	21	20	19	18	17	16	
保留位								
15	14	13	12	11	10	9	8	
EXTIRST	AFIORST	保留位		UR1RST	UR0RST	保留位		
RW	0	RW	0	RW	0	RW	0	
7	6	5	4	3	2	1	0	
保留位		SPI1RST	SPI0RST	保留位			I2CRST	
		RW	0	RW	0	RW		0

位	字段	描述
[15]	EXTIRST	外部中断控制器复位控制位 0: 无复位 1: 复位 EXTI 该位由软件置位, 由硬件自动清零。
[14]	AFIORST	复用功能 I/O 复位控制位 0: 无复位 1: 复位复用功能 I/O 该位由软件置位, 由硬件自动清零。
[11]	UR1RST	UART1 复位控制位 0: 无复位 1: 复位 UART1 该位由软件置位, 由硬件自动清零。
[10]	UR0RST	UART0 复位控制位 0: 无复位 1: 复位 UART0 该位由软件置位, 由硬件自动清零。
[5]	SPI1RST	SPI1 复位控制位 0: 无复位 1: 复位 SPI1 该位由软件置位, 由硬件自动清零。
[4]	SPI0RST	SPI0 复位控制位 0: 无复位 1: 复位 SPI0 该位由软件置位, 由硬件自动清零。
[0]	I2CRST	I ² C 复位控制位 0: 无复位 1: 复位 I ² C 该位由软件置位, 由硬件自动清零。

APB 外设复位寄存器 1 – APBPRSTR1

该寄存器定义了几种 APB 外设软件复位控制位。

偏移量: 0x10C
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24		
类型 / 复位	保留位							ADCRST	RW	0
	23	22	21	20	19	18	17	16		
类型 / 复位	保留位							BFTMRST	RW	0
	15	14	13	12	11	10	9	8		
类型 / 复位	保留位		PWM1RST	PWM0RST	保留位			GPTMRST	RW	0
	7	6	5	4	3	2	1	0		
类型 / 复位	保留位			WDTRST	保留位				RW	0

位	字段	描述
[24]	ADCRST	A/D 转换器复位控制位 0: 无复位 1: 复位 A/D 转换器 该位由软件置位, 由硬件自动清零。
[16]	BFTMRST	BFTM 复位控制位 0: 无复位 1: 复位 BFTM 该位由软件置位, 由硬件自动清零。
[13]	PWM1RST	PWM1 复位控制位 0: 无复位 1: 复位 PWM1 该位由软件置位, 由硬件自动清零。
[12]	PWM0RST	PWM0 复位控制位 0: 无复位 1: 复位 PWM0 该位由软件置位, 由硬件自动清零。
[8]	GPTMRST	GPTM 复位控制位 0: 无复位 1: 复位 GPTM 该位由软件置位, 由硬件自动清零。
[4]	WDTRST	看门狗定时器复位控制位 0: 无复位 1: 复位看门狗定时器 该位由软件置位, 由硬件自动清零。

8 通用 I/O (GPIO)

简介

该系列单片机有多达 40 个通用 I/O 引脚 (GPIO), 即 PA0 ~ PA15、PB0 ~ PB15 和 PC0 ~ PC7, 可以实现逻辑输入 / 输出功能。每个 GPIO 端口都有相关的控制和配置寄存器, 以满足特定应用的需求。实际可用的通用 I/O 端口数量取决于单片机特性和封装类型。详细信息请参考该系列单片机的数据手册 (Datasheet)。

GPIO 引脚与其它复用功能 (AF) 共用, 以获得较大的封装灵活性。通过配置相应的寄存器, GPIO 端口可以被用作复用功能 AF 的输入或输出脚。

单片机 GPIO 引脚的外部中断功能可通过相关的控制和配置寄存器设置, 具体见外部中断控制单元 (EXTI)。

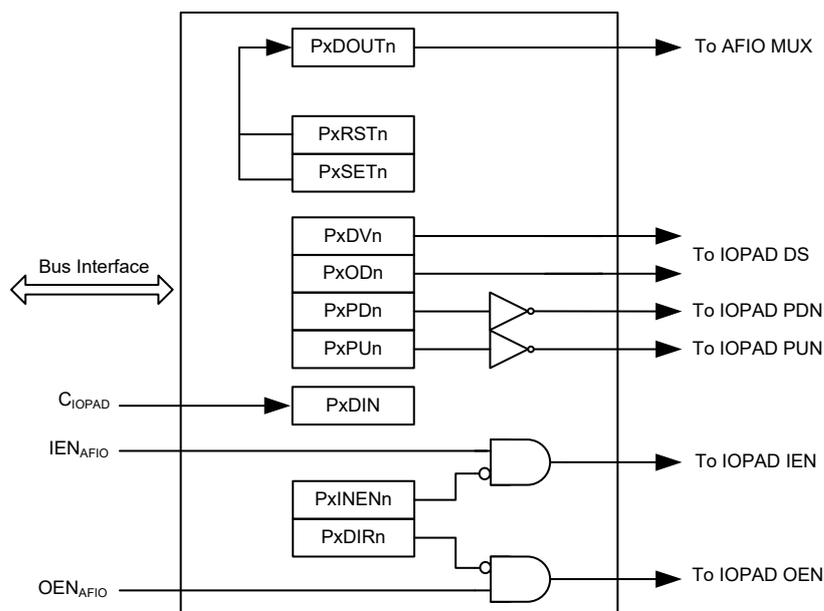


图 19. GPIO 方框图

特性

- 输入 / 输出方向控制
- 输入引脚弱上拉 / 下拉功能
- 输出推挽 / 漏极开路使能控制
- 输出置位 / 复位控制
- 输出驱动电流选择
- 可编程边沿触发的外部中断 – 使用 EXTI 配置寄存器
- 模拟输入 / 输出配置 – 使用 AFIO 配置寄存器
- 复用功能的输入 / 输出配置 – 使用 AFIO 配置寄存器
- 端口配置锁定

功能描述

默认的 GPIO 引脚配置

在复位期间或刚复位后，复用功能都无效，GPIO 端口被配置为输入除能浮空模式，即没有上拉 / 下拉电阻且输入除能。单片机复位后，只有启动和串行线调试引脚这几根与 I/O 引脚共用的引脚是有效的。

- BOOT：带有内部上拉的输入使能
- SWCLK：带有内部上拉的输入使能
- SWDIO：带有内部上拉的输入使能

通用 I/O – GPIO

GPIO 引脚可以通过数据方向控制寄存器 PxDIRCR (其中 x = A ~ C) 配置为输入或输出。当 GPIO 引脚配置为输入引脚时，如果输入使能功能寄存器 PxINER 的使能位被置位，则外部引脚上的数据可读。GPIO 上拉 / 下拉寄存器 PxPUR / PxPDR 可配置以满足特定应用。当上拉和下拉功能同时使能，上拉功能具有较高的优先级，下拉功能将被封锁直到上拉功能被除能。

GPIO 引脚可以配置为输出引脚，输出数据锁存到数据寄存器 PxDOUTr。通过漏极开路选择寄存器 PxODR，输出类型可设置为推挽式或漏极开路式。通过配置端口输出置位 / 复位控制寄存器 PxSRR 或端口输出复位控制寄存器 PxRR，可在不影响未被选中的位的情况下，对选中的一个或几个特定位进行输出数据的置位 / 复位。端口输出置位 / 复位功能同时使能时，端口输出置位功能具有更高的优先级，端口输出复位功能将被封锁。通过配置驱动电流选择寄存器 PxDRVR，选择 GPIO 引脚输出的驱动电流。

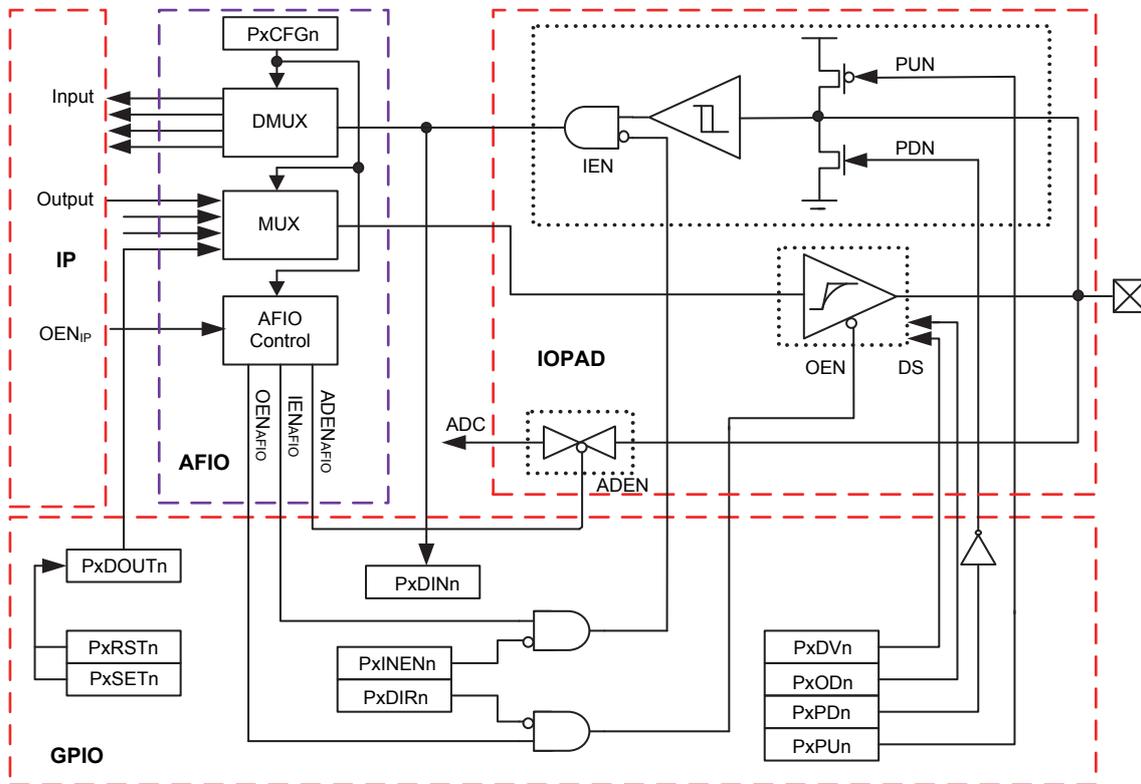


图 20. AFIO/GPIO 控制信号

PxDINn / PxDOUn (x = A ~ C): 数据输入 / 数据输出

PxRSTn / PxSETn (x = A ~ C): 复位 / 置位

PxDIRn (x = A ~ C): 方向

PxINENn (x = A ~ C): 输入使能

PxDVn (x = A ~ C): 输出驱动

PxODn (x = A ~ C): 漏极开路

PxPDn / PxPUn (x = A ~ C): 下拉 / 上拉

PxCFGn (x = A ~ C): AFIO 配置

表 17. AFIO、GPIO 和 I/O 引脚控制信号真值表

类型	AFIO			GPIO		PAD		
	ADEN _{AFIO}	OEN _{AFIO}	IEN _{AFIO}	PxDIRn	PxINENn	ADEN	OEN	IEN
GPIO 输入 (注)	1	1	1	0	1	1	1	0
GPIO 输出 (注)	1	1	1	1	0 (1 如需要)	1	0	1 (0)
AFIO 输入	1	1	0	0	X	1	1	0
AFIO 输出	1	0	1	X	0 (1 如需要)	1	0	1 (0)
ADC 输入	0	1	1	0	0 (1 如需要)	0	1	1 (0)
OSC 输出	0	1	1	0	0 (1 如需要)	0	1	1 (0)

注: 当相关引脚被配置为 GPIO 输入 / 输出模式时, I/O 引脚的 IEN 和 OEN 信号分别来自 GPIO 寄存器 PxINENn 和 PxDIRn 位。

GPIO 锁定机制

GPIO 还提供了锁定功能用来锁定端口，直到复位事件发生。PxLOCKR (x = A ~ C) 寄存器是用来锁定端口 x 和锁定控制选项。在 PxLOCKR 寄存器 PxLKEY 字段写入值 0x5FA0 冻结 PxDIRCR、PxINER、PxPUR、PxPDR、PxODR、PxDRVR 控制和 AFIO 的模式配置 (GPxCFGHR 或 GPxCFGLR, 其中 x = A ~ C)。如果 PxLOCKR 寄存器写入值为 0x5FA0_0001, 这意味着端口 x 锁定功能使能, 端口 x 引脚 0 被冻结。

寄存器列表

下表显示 GPIO 寄存器和端口 A ~ C 的复位值。

表 18. GPIO 寄存器列表

寄存器	偏移量	描述	复位值
GPIO A 基址 = 0x400B_0000			
PADIRCR	0x000	端口 A 数据方向控制寄存器	0x0000_0000
PAINER	0x004	端口 A 输入功能使能控制寄存器	0x0000_0200
PAPUR	0x008	端口 A 上拉选择寄存器	0x0000_3200
PAPDR	0x00C	端口 A 下拉选择寄存器	0x0000_0000
PAODR	0x010	端口 A 漏极开路选择寄存器	0x0000_0000
PADRVR	0x014	端口 A 驱动电流选择寄存器	0x0000_0000
PALOCKR	0x018	端口 A 锁定寄存器	0x0000_0000
PADINR	0x01C	端口 A 数据输入寄存器	0x0000_3200
PADOUTR	0x020	端口 A 数据输出寄存器	0x0000_0000
PASRR	0x024	端口 A 输出置位 / 复位控制寄存器	0x0000_0000
PARR	0x028	端口 A 输出复位控制寄存器	0x0000_0000
PASCER	0x02C	端口 A 灌电流增强选择寄存器	0x0000_0000
GPIO B 基址 = 0x400B_2000			
PBDIRCR	0x000	端口 B 数据方向控制寄存器	0x0000_0000
PBINER	0x004	端口 B 输入功能使能控制寄存器	0x0000_0000
PBPUR	0x008	端口 B 上拉选择寄存器	0x0000_0000
PBPDR	0x00C	端口 B 下拉选择寄存器	0x0000_0000
PBODR	0x010	端口 B 漏极开路选择寄存器	0x0000_0000
PBDRVR	0x014	端口 B 驱动电流选择寄存器	0x0000_0000
PBLOCKR	0x018	端口 B 锁定寄存器	0x0000_0000
PBDINR	0x01C	端口 B 数据输入寄存器	0x0000_0000
PBDOUTR	0x020	端口 B 数据输出寄存器	0x0000_0000
PBSRR	0x024	端口 B 输出置位 / 复位控制寄存器	0x0000_0000
PBRR	0x028	端口 B 输出复位控制寄存器	0x0000_0000
PBSCER	0x02C	端口 B 灌电流增强选择寄存器	0x0000_0000
GPIO C 基址 = 0x400B_4000			
PCDIRCR	0x000	端口 C 数据方向控制寄存器	0x0000_0000
PCINER	0x004	端口 C 输入功能使能控制寄存器	0x0000_0000
PCPUR	0x008	端口 C 上拉选择寄存器	0x0000_0000
PCPDR	0x00C	端口 C 下拉选择寄存器	0x0000_0000
PCODR	0x010	端口 C 漏极开路选择寄存器	0x0000_0000
PCDRVR	0x014	端口 C 驱动电流选择寄存器	0x0000_0000
PCLOCKR	0x018	端口 C 锁定寄存器	0x0000_0000

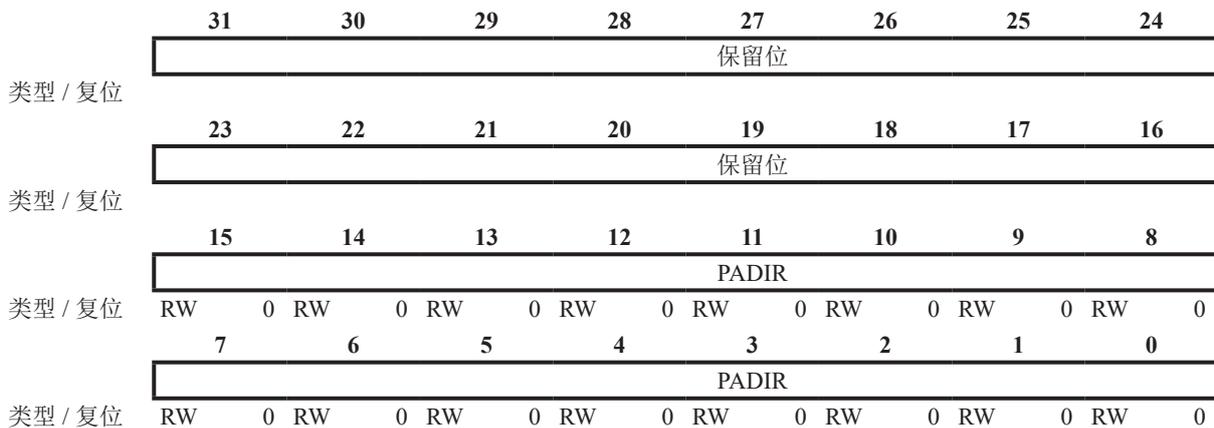
寄存器	偏移量	描述	复位值
PCDINR	0x01C	端口 C 数据输入寄存器	0x0000_0000
PCDOUTR	0x020	端口 C 数据输出寄存器	0x0000_0000
PCSRR	0x024	端口 C 输出置位 / 复位控制寄存器	0x0000_0000
PCRR	0x028	端口 C 输出复位控制寄存器	0x0000_0000
PCSCER	0x02C	端口 C 灌电流增强选择寄存器	0x0000_0000

寄存器描述

端口 A 数据方向控制寄存器 – PADIRCR

该寄存器用来控制 GPIO 端口 A 引脚方向为输入或输出。

偏移量: 0x000
复位值: 0x0000_0000



位	字段	描述
[15:0]	PADIRn	GPIO 端口 A 引脚 n 的方向控制位 (n = 0 ~ 15) 0: 引脚 n 处于输入模式 1: 引脚 n 处于输出模式

端口 A 输入功能使能控制寄存器 – PAINER

该寄存器用来使能或除能 GPIO 端口 A 的输入功能。

偏移量: 0x004

复位值: 0x0000_0200

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位								
	15	14	13	12	11	10	9	8	
类型 / 复位	PAINEN								
	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
类型 / 复位	PAINEN								
	RW	0	RW	0	RW	0	RW	0	RW

位	字段	描述
[15:0]	PAINEN _n	GPIO 端口 A 引脚 n 的输入使能控制位 (n = 0 ~ 15) 0: 除能引脚 n 的输入功能 1: 使能引脚 n 的输入功能 当除能引脚 n 的输入功能时, 输入施密特触发器将关闭, 施密特触发器输出将保持在零状态。

端口 A 上拉选择寄存器 – PAPUR

该寄存器用来使能或除能 GPIO 端口 A 的上拉功能。

偏移量: 0x008

复位值: 0x0000_3200

	31	30	29	28	27	26	25	24									
类型 / 复位	保留位																
	23	22	21	20	19	18	17	16									
类型 / 复位	保留位																
	15	14	13	12	11	10	9	8									
类型 / 复位	PAPU																
	RW	0	RW	0	RW	1	RW	1	RW	0	0	RW	0	RW	1	RW	0
	7	6	5	4	3	2	1	0									
类型 / 复位	PAPU																
	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	

位	字段	描述
[15:0]	PAPUn	GPIO 端口 A 引脚 n 的上拉选择控制位 (n = 0 ~ 15) 0: 除能引脚 n 的上拉功能 1: 使能引脚 n 的上拉功能 注: 当上拉和下拉功能都使能, 上拉功能将有更高的优先级, 因此下拉功能将被封锁和除能。

端口 A 下拉选择寄存器 – PAPDR

该寄存器用来使能或除能 GPIO 端口 A 的下拉功能。

偏移量: 0x00C

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位								
	15	14	13	12	11	10	9	8	
类型 / 复位	PAPD								
	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
类型 / 复位	PAPD								
	RW	0	RW	0	RW	0	RW	0	RW

位	字段	描述
[15:0]	PAPDn	GPIO 端口 A 引脚 n 的下拉选择控制位 (n = 0 ~ 15) 0: 除能引脚 n 的下拉功能 1: 使能引脚 n 的下拉功能 注: 当上拉和下拉功能都使能, 上拉功能将有更高的优先级, 因此下拉功能将被封锁和除能。

端口 A 漏极开路选择寄存器 – PAODR

该寄存器用来使能或除能 GPIO 端口 A 的漏极开路功能。

偏移量: 0x010

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位								
	15	14	13	12	11	10	9	8	
类型 / 复位	PAOD								
	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
类型 / 复位	PAOD								
	RW	0	RW	0	RW	0	RW	0	RW

位	字段	描述
[15:0]	PAODn	GPIO 端口 A 引脚 n 的漏极开路选择控制位 (n = 0 ~ 15) 0: 除能引脚 n 的漏极开路输出 (输出类型为 CMOS 输出) 1: 使能引脚 n 的漏极开路输出 (输出类型为漏极开路)

端口 A 输出驱动电流选择寄存器 – PADDRV

该寄存器用来定义 GPIO 端口 A 的输出驱动电流。

偏移量: 0x014
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
	PADV15		PADV14		PADV13		PADV12		
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	23	22	21	20	19	18	17	16	
	PADV11		PADV10		PADV9		PADV8		
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	15	14	13	12	11	10	9	8	
	PADV7		PADV6		PADV5		PADV4		
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
	PADV3		PADV2		PADV1		PADV0		
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	字段	描述
[31:0]	PADV _n [1:0]	GPIO 端口 A 引脚 n 的输出驱动电流选择控制位 (n = 0 ~ 15) 00: 4 mA 源 / 灌电流 01: 8 mA 源 / 灌电流 10: 12 mA 源 / 灌电流 11: 16 mA 源 / 灌电流

端口 A 锁定寄存器 – PALOCKR

该寄存器定义了 GPIO 端口 A 的锁定配置。

偏移量: 0x018
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	PALKEY					
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0
	23	22	21	20	19	18	17	16	PALKEY					
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0
	15	14	13	12	11	10	9	8	PALOCK					
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0	PALOCK					
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0

位	字段	描述
[31:16]	PALKEY	GPIO 端口 A 锁定键 0x5FA0: 使能端口 A 锁定功能 其它: 除能端口 A 锁定功能 为了锁定端口 A 功能, 在该寄存器 PALKEY 字段写入值 0x5FA0。为了在这个锁定寄存器执行一个成功的写操作, 写入 PALKEY 字段的值必须为 0x5FA0。如果写入这个字段值不等于 0x5FA0, PALOCKR 寄存器上的任何写操作将被中止。对 PALKEY 字段的一个读操作的结果将返回 GPIO 端口 A 锁定状态, 表示 GPIO 端口 A 是否被锁定。如果读值, PALKEY 字段为 0, 表明 GPIO 端口 A 锁定功能除能。否则, 它表明 GPIO 端口 A 锁定功能使能, 读值等于 1。
[15:0]	PALOCK _n	GPIO 端口 A 引脚 n 的锁定控制位 (n = 0 ~ 15) 0: 端口 A 引脚 n 未锁定 1: 端口 A 引脚 n 锁定 当正确的锁定键应用于 PALKEY 字段, PALOCK _n 位用来锁定相应 GPIO 引脚的配置。被锁定的配置包括 PADIR _n 、PAINEN _n 、PAPUn、PAPD _n 、PAOD _n 和 PADV _n 设置相关的 GPIO 寄存器。另外, 用来配置为相关 GPIO 引脚的复用功能的 AFIO 寄存器 GPACFGHR 或 GPACFGLR 寄存器也将被锁定。请注意, PALOCKR 寄存器只能写入一次, 这意味着 PALKEY 和 PALOCK _n (锁定控制位) 应一起写入, 不能改变直到系统复位或 GPIO 端口 A 发生复位。

端口 A 数据输入寄存器 – PADINR

此寄存器定义了 GPIO 端口 A 的输入数据。

偏移量: 0x01C

复位值: 0x0000_3200

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位								
	15	14	13	12	11	10	9	8	
类型 / 复位	PADIN								
	RO	0	RO	0	RO	1	RO	1	RO
	0	RO	0	RO	0	RO	0	RO	1
	RO	0	RO	0	RO	0	RO	0	RO
	0	RO	0	RO	0	RO	0	RO	0
	7	6	5	4	3	2	1	0	
类型 / 复位	PADIN								
	RO	0	RO	0	RO	0	RO	0	RO
	0	RO	0	RO	0	RO	0	RO	0
	RO	0	RO	0	RO	0	RO	0	RO
	0	RO	0	RO	0	RO	0	RO	0

位	字段	描述
[15:0]	PADINn	GPIO 端口 A 引脚 n 的数据输入位 (n = 0 ~ 15) 0: 引脚 n 的输入数据为 0 1: 引脚 n 的输入数据为 1

端口 A 输出数据寄存器 – PADOUTR

该寄存器定义了 GPIO 端口 A 的输出数据。

偏移量: 0x020

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位								
	15	14	13	12	11	10	9	8	
类型 / 复位	PADOUT								
	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
类型 / 复位	PADOUT								
	RW	0	RW	0	RW	0	RW	0	RW

位	字段	描述
[15:0]	PADOUT _n	GPIO 端口 A 引脚 n 的数据输出位 (n = 0 ~ 15) 0: 引脚 n 的输出数据为 0 1: 引脚 n 的输出数据为 1

端口 A 输出置位 / 复位控制寄存器 – PASRR

该寄存器用来置位或复位 GPIO 端口 A 输出数据相应的位。

偏移量: 0x024

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
	PARST								
类型 / 复位	WO	0	WO	0	WO	0	WO	0	WO
	23	22	21	20	19	18	17	16	
	PARST								
类型 / 复位	WO	0	WO	0	WO	0	WO	0	WO
	15	14	13	12	11	10	9	8	
	PASET								
类型 / 复位	WO	0	WO	0	WO	0	WO	0	WO
	7	6	5	4	3	2	1	0	
	PASET								
类型 / 复位	WO	0	WO	0	WO	0	WO	0	WO

位	字段	描述
[31:16]	PARSTn	GPIO 端口 A 引脚 n 的输出复位控制位 (n = 0 ~ 15) 0: 不影响 PADOUTn 位 1: 复位 PADOUTn 位 注意, 当该寄存器中的 PARSTn 位或 PARR 寄存器中的 PARSTn 位置位时, PADOUTn 位上的复位功能将生效。
[15:0]	PASETn	GPIO 端口 A 引脚 n 的输出置位控制位 (n = 0 ~ 15) 0: 不影响 PADOUTn 位 1: 置位 PADOUTn 位 注意, 如果 PASETn 和 PARSTn 位同时被置位, PASETn 位使能的功能有比较高的优先级。

端口 A 输出复位寄存器 – PARR

该寄存器用来复位 GPIO 端口 A 输出数据相应的位。

偏移量: 0x028
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	保留位							
类型 / 复位																
	23	22	21	20	19	18	17	16	保留位							
类型 / 复位																
	15	14	13	12	11	10	9	8	PARST							
类型 / 复位	WO	0	WO	0	WO	0	WO	0	WO	0	WO	0	WO	0	WO	0
	7	6	5	4	3	2	1	0	PARST							
类型 / 复位	WO	0	WO	0	WO	0	WO	0	WO	0	WO	0	WO	0	WO	0

位	字段	描述
[15:0]	PARSTn	GPIO 端口 A 引脚 n 的输出复位控制位 (n = 0 ~ 15) 0: 不影响 PADOUTn 位 1: 复位 PADOUTn 位

端口 A 灌电流增强选择寄存器 – PASCER

该寄存器用来定义 GPIO 端口 A 的增强灌电流驱动。

偏移量: 0x02C

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位								
	15	14	13	12	11	10	9	8	
类型 / 复位	PASCE15	PASCE14	PASCE13	PASCE12	PASCE11	PASCE10	PASCE9	PASCE8	
	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
类型 / 复位	PASCE7	PASCE6	PASCE5	PASCE4	PASCE3	PASCE2	PASCE1	PASCE0	
	RW	0	RW	0	RW	0	RW	0	RW

位	字段	描述
[15:0]	PASCE _n	GPIO 端口 A 引脚 n 的灌电流增强选择控制位 (n = 0 ~ 15) 0: 无增强灌电流 1: 增强灌电流

端口 B 数据方向控制寄存器 – PBDIRCR

该寄存器用来控制 GPIO 端口 B 引脚的方向作为输入或输出。

偏移量: 0x000

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位								
	15	14	13	12	11	10	9	8	
类型 / 复位	PBDIR								
	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
类型 / 复位	PBDIR								
	RW	0	RW	0	RW	0	RW	0	RW

位	字段	描述
[15:0]	PBDIRn	GPIO 端口 B 引脚 n 的方向控制位 (n = 0 ~ 15) 0: 引脚 n 处于输入模式 1: 引脚 n 处于输出模式

端口 B 输入功能使能控制寄存器 – PBINER

此寄存器用来使能或除能 GPIO 端口 B 的输入功能。

偏移量: 0x004

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位								
	15	14	13	12	11	10	9	8	
类型 / 复位	PBINEN								
	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
类型 / 复位	PBINEN								
	RW	0	RW	0	RW	0	RW	0	RW

位	字段	描述
[15:0]	PBINEN _n	GPIO 端口 B 引脚 n 的输入使能控制位 (n = 0 ~ 15) 0: 除能引脚 n 的输入功能 1: 使能引脚 n 的输入功能 当除能引脚 n 的输入功能时, 输入施密特触发器将关闭, 施密特触发器输出将保持在零状态。

端口 B 上拉选择寄存器 – PBPUR

该寄存器用来使能或除能 GPIO 端口 B 的上拉功能。

偏移量: 0x008

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位								
	15	14	13	12	11	10	9	8	
类型 / 复位	PBPU								
	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
类型 / 复位	PBPU								
	RW	0	RW	0	RW	0	RW	0	RW

位	字段	描述
[15:0]	PBPUn	GPIO 端口 B 引脚 n 的上拉选择控制位 (n = 0 ~ 15) 0: 除能引脚 n 的上拉功能 1: 使能引脚 n 的上拉功能 注: 当上拉和下拉功能都使能, 上拉功能将有更高的优先级, 因此下拉功能将被封锁和除能。

端口 B 下拉选择寄存器 – PBPDR

该寄存器用来使能或除能 GPIO 端口 B 的下拉功能。

偏移量: 0x00C

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位								
	15	14	13	12	11	10	9	8	
类型 / 复位	PBPDR								
	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
类型 / 复位	PBPDR								
	RW	0	RW	0	RW	0	RW	0	RW

位	字段	描述
[15:0]	PBPDn	GPIO 端口 B 引脚 n 的下拉选择控制位 (n = 0 ~ 15) 0: 除能引脚 n 的下拉功能 1: 使能引脚 n 的下拉功能 注: 当上拉和下拉功能都使能时, 上拉功能将有更高的优先级, 因此下拉功能将被封锁和除能。

端口 B 漏极开路选择寄存器 – PBODR

该寄存器用来使能或除能 GPIO 端口 B 的漏极开路功能。

偏移量: 0x010

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位								
	15	14	13	12	11	10	9	8	
类型 / 复位	PBOD								
	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
类型 / 复位	PBOD								
	RW	0	RW	0	RW	0	RW	0	RW

位	字段	描述
[15:0]	PBODn	GPIO 端口 B 引脚 n 的漏极开路选择控制位 (n = 0 ~ 15) 0: 除能引脚 n 的漏极开路输出 (输出类型为 CMOS 输出) 1: 使能引脚 n 的漏极开路输出 (输出类型为漏极开路)

端口 B 输出驱动电流选择寄存器 – PBDRVR

该寄存器用来定义 GPIO 端口 B 的输出驱动电流。

偏移量: 0x014
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
	PBDV15		PBDV14		PBDV13		PBDV12		
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	23	22	21	20	19	18	17	16	
	PBDV11		PBDV10		PBDV9		PBDV8		
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	15	14	13	12	11	10	9	8	
	PBDV7		PBDV6		PBDV5		PBDV4		
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
	PBDV3		PBDV2		PBDV1		PBDV0		
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	字段	描述
[31:0]	PBDVn[1:0]	GPIO 端口 B 引脚 n 的输出驱动电流选择控制位 (n = 0 ~ 15) 00: 4 mA 源 / 灌电流 01: 8 mA 源 / 灌电流 10: 12 mA 源 / 灌电流 11: 16 mA 源 / 灌电流

端口 B 锁定寄存器 – PBLOCKR

该寄存器定义了 GPIO 端口 B 的锁定配置。

偏移量: 0x018
复位值: 0x0000_0000

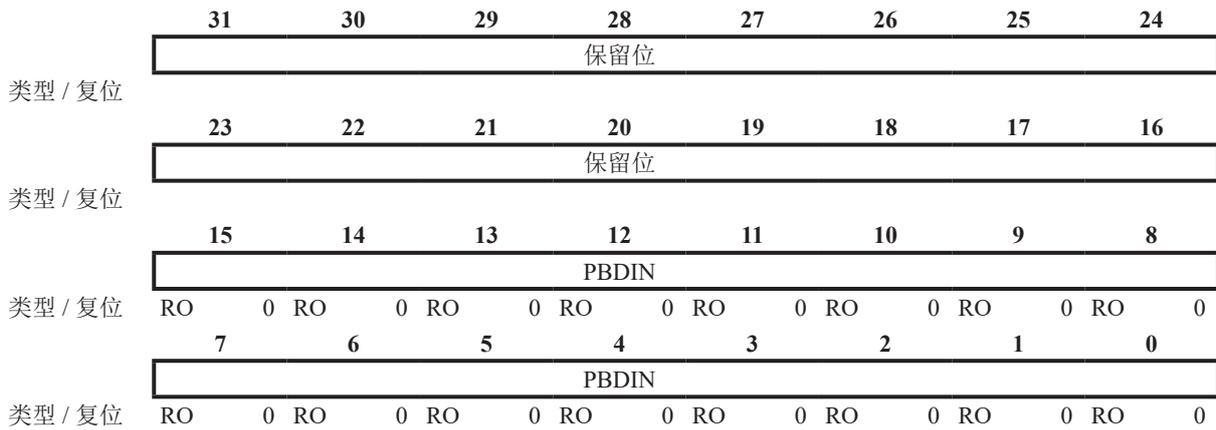
	31	30	29	28	27	26	25	24	PBLKEY													
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0								
	23	22	21	20	19	18	17	16	PBLKEY													
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0								
	15	14	13	12	11	10	9	8	PBLOCK													
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0								
	7	6	5	4	3	2	1	0	PBLOCK													
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0								

位	字段	描述
[31:16]	PBLKEY	GPIO 端口 B 锁定键 0x5FA0: 使能端口 B 锁定功能 其它: 除能端口 B 锁定功能 为了锁定端口 B 功能, 在该寄存器 PBLKEY 字段写入值 0x5FA0。为了在这个锁定寄存器执行一个成功的写操作, 写入 PBLKEY 字段的值必须为 0x5FA0。如果写入这个字段值不等于 0x5FA0, PBLOCKR 寄存器上的任何写操作将被中止。对 PBLKEY 字段的一个读操作的结果将返回 GPIO 端口 B 锁定状态, 表示 GPIO 端口 B 是否被锁定。如果读值, PBLKEY 字段为 0, 表明 GPIO 端口 B 锁定功能除能。否则, 它表明 GPIO 端口 B 锁定功能使能, 读值等于 1。
[15:0]	PBLOCKn	GPIO 端口 B 引脚 n 的锁定控制位 (n = 0 ~ 15) 0: 端口 B 引脚 n 未锁定 1: 端口 B 引脚 n 锁定 当正确的锁定键应用于 PBLKEY 字段, PBLOCKn 位用来锁定相应 GPIO 引脚的配置。被锁定的配置包括 PBDIRn、PBINENn、PBPUn、PBPDn、PBODn 和 PBDVn 设置相关的 GPIO 寄存器。另外, 用来配置为相关 GPIO 引脚的复用功能的 AFIO 寄存器 GPBCFGHR 或 GPBCFGRLR 寄存器也将被锁定。请注意, PBLOCKR 寄存器只能写入一次, 这意味着 PBLKEY 和 PBLOCKn (锁定控制位) 应一起写入, 不能改变直到系统复位或 GPIO 端口 B 发生复位。

端口 B 数据输入寄存器 – PBDINR

该寄存器指定 GPIO 端口 B 的输入数据。

偏移量: 0x01C
复位值: 0x0000_0000



位	字段	描述
[15:0]	PBDINn	GPIO 端口 B 引脚 n 的数据输入位 (n = 0 ~ 15) 0: 引脚 n 的输入数据为 0 1: 引脚 n 的输入数据为 1

端口 B 输出数据寄存器 – PBDOUTR

该寄存器指定 GPIO 端口 B 的输出数据。

偏移量: 0x020

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位								
	15	14	13	12	11	10	9	8	
类型 / 复位	PBDOUT								
	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
类型 / 复位	PBDOUT								
	RW	0	RW	0	RW	0	RW	0	RW

位	字段	描述
[15:0]	PBDOUTn	GPIO 端口 B 引脚 n 的数据输出位 (n = 0 ~ 15) 0: 引脚 n 的输出数据为 0 1: 引脚 n 的输出数据为 1

端口 B 输出置位 / 复位控制寄存器 – PBSRR

该寄存器用来置位或复位 GPIO 端口 B 输出数据相应的位。

偏移量: 0x024

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
	PBRST								
类型 / 复位	WO	0	WO	0	WO	0	WO	0	WO
	23	22	21	20	19	18	17	16	
	PBRST								
类型 / 复位	WO	0	WO	0	WO	0	WO	0	WO
	15	14	13	12	11	10	9	8	
	PBSET								
类型 / 复位	WO	0	WO	0	WO	0	WO	0	WO
	7	6	5	4	3	2	1	0	
	PBSET								
类型 / 复位	WO	0	WO	0	WO	0	WO	0	WO

位	字段	描述
[31:16]	PBRSTn	GPIO 端口 B 引脚 n 的输出复位控制位 (n = 0 ~ 15) 0: 不影响 PBDOUTn 位 1: 复位 PBDOUTn 位 注意, 当该寄存器中的 PBRSTn 位或 PBRR 寄存器中的 PBRSTn 位置位时, PBDOUTn 位上的复位功能将生效。
[15:0]	PBSETn	GPIO 端口 B 引脚 n 的输出置位控制位 (n = 0 ~ 15) 0: 不影响 PBDOUTn 位 1: 置位 PBDOUTn 位 注意, 如果 PBSETn 和 PBRSTn 位同时被置位, PBSETn 位使能的功能有比较高的优先级。

端口 B 输出复位寄存器 – PBRR

该寄存器用来复位 GPIO 端口 B 输出数据相应的位。

偏移量: 0x028

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位								
	15	14	13	12	11	10	9	8	
类型 / 复位	PBRST								
	WO	0	WO	0	WO	0	WO	0	WO
	7	6	5	4	3	2	1	0	
类型 / 复位	PBRST								
	WO	0	WO	0	WO	0	WO	0	WO

位	字段	描述
[15:0]	PBRSTn	GPIO 端口 B 引脚 n 的输出复位控制位 (n = 0 ~ 15) 0: 不影响 PBDOUTn 位 1: 复位 PBDOUTn 位

端口 B 灌电流增强选择寄存器 – PBSCER

该寄存器用来定义 GPIO 端口 B 的增强灌电流驱动。

偏移量: 0x02C

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	保留位							
类型 / 复位																
	23	22	21	20	19	18	17	16	保留位							
类型 / 复位																
	15	14	13	12	11	10	9	8	PBSCE15	PBSCE14	PBSCE13	PBSCE12	PBSCE11	PBSCE10	PBSCE9	PBSCE8
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0	PBSCE7	PBSCE6	PBSCE5	PBSCE4	PBSCE3	PBSCE2	PBSCE1	PBSCE0
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0

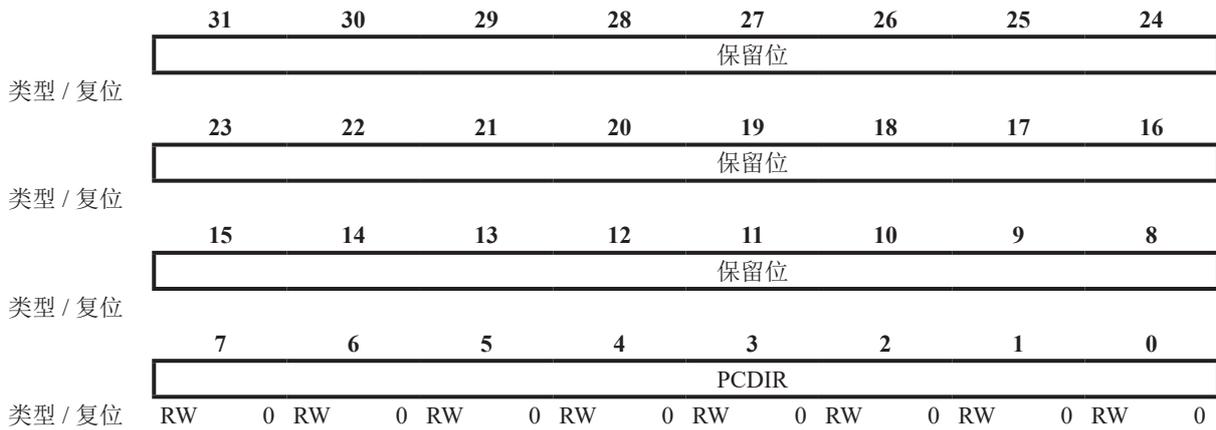
位	字段	描述
[15:0]	PBSCE _n	GPIO 端口 B 引脚 n 的灌电流增强选择控制位 (n = 0 ~ 15) 0: 无增强灌电流 1: 增强灌电流

端口 C 数据方向控制寄存器 – PCDIRCR

该寄存器用来控制 GPIO 端口 C 引脚的方向作为输入或输出。

偏移量: 0x000

复位值: 0x0000_0000



位	字段	描述
[7:0]	PCDIRn	GPIO 端口 C 引脚 n 的方向控制位 (n = 0 ~ 7) 0: 引脚 n 处于输入模式 1: 引脚 n 处于输出模式

端口 C 输入功能使能控制寄存器 – PCINER

此寄存器用来使能或除能 GPIO 端口 C 的输入功能。

偏移量: 0x004

复位值: 0x0000_0000

31	30	29	28	27	26	25	24							
保留位														
23	22	21	20	19	18	17	16							
保留位														
15	14	13	12	11	10	9	8							
保留位														
7	6	5	4	3	2	1	0							
PCINEN														
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0

位	字段	描述
[7:0]	PCINEN _n	GPIO 端口 C 引脚 n 的输入使能控制位 (n = 0 ~ 7) 0: 除能引脚 n 的输入功能 1: 使能引脚 n 的输入功能 当除能引脚 n 的输入功能时, 输入施密特触发器将关闭, 施密特触发器输出将保持在零状态。

端口 C 上拉选择寄存器 – PCPUR

该寄存器用来使能或除能 GPIO 端口 C 的上拉功能。

偏移量: 0x008

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位								
	15	14	13	12	11	10	9	8	
类型 / 复位	保留位								
	7	6	5	4	3	2	1	0	
类型 / 复位	PCPU								
	RW	0	RW	0	RW	0	RW	0	RW

位	字段	描述
[7:0]	PCPU _n	GPIO 端口 C 引脚 n 的上拉选择控制位 (n = 0 ~ 7) 0: 除能引脚 n 的上拉功能 1: 使能引脚 n 的上拉功能 注: 当上拉和下拉功能都使能, 上拉功能将有更高的优先级, 因此下拉功能将被封锁和除能。

端口 C 下拉选择寄存器 – PCPDR

该寄存器用来使能或除能 GPIO 端口 C 的下拉功能。

偏移量: 0x00C

复位值: 0x0000_0000

31	30	29	28	27	26	25	24							
保留位														
23	22	21	20	19	18	17	16							
保留位														
15	14	13	12	11	10	9	8							
保留位														
7	6	5	4	3	2	1	0							
PCPD														
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0

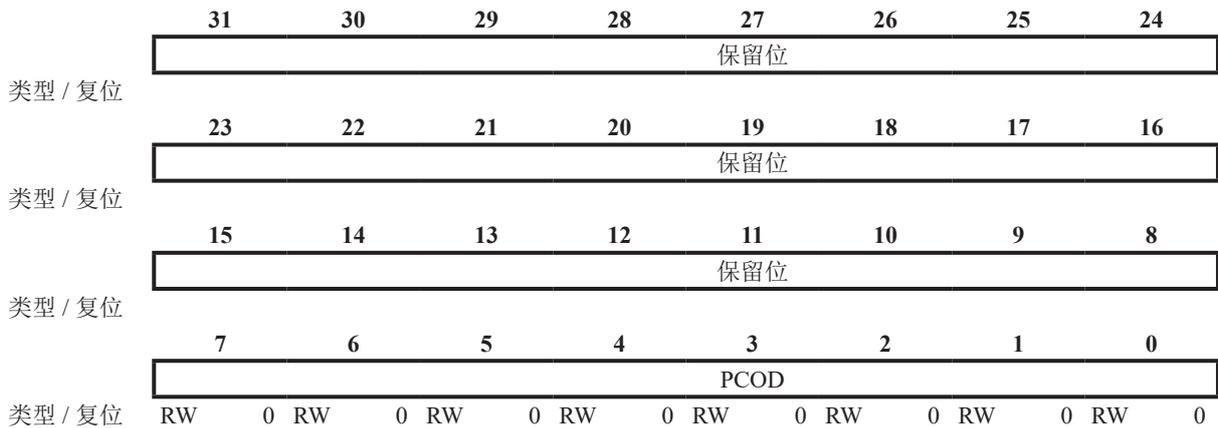
位	字段	描述
[7:0]	PCPDn	GPIO 端口 C 引脚 n 的下拉选择控制位 (n = 0 ~ 7) 0: 除能引脚 n 的下拉功能 1: 使能引脚 n 的下拉功能 注: 当上拉和下拉功能都使能时, 上拉功能将有更高的优先级, 因此下拉功能将被封锁和除能。

端口 C 漏极开路选择寄存器 – PCODR

该寄存器用来使能或除能 GPIO 端口 C 的漏极开路功能。

偏移量: 0x010

复位值: 0x0000_0000



位	字段	描述
[7:0]	PCODn	GPIO 端口 C 引脚 n 的漏极开路选择控制位 (n = 0 ~ 7) 0: 除能引脚 n 的漏极开路输出 (输出类型为 CMOS 输出) 1: 使能引脚 n 的漏极开路输出 (输出类型为漏极开路)

端口 C 输出驱动电流选择寄存器 – PCDRVR

该寄存器用来定义 GPIO 端口 C 的输出驱动电流。

偏移量: 0x014

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位								
	15	14	13	12	11	10	9	8	
类型 / 复位	PCDV7		PCDV6			PCDV5		PCDV4	
	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
类型 / 复位	PCDV3		PCDV2			PCDV1		PCDV0	
	RW	0	RW	0	RW	0	RW	0	RW

位	字段	描述
[15:0]	PCDVn[1:0]	GPIO 端口 C 引脚 n 的输出驱动电流选择控制位 (n = 0 ~ 7) 00: 4 mA 源 / 灌电流 01: 8 mA 源 / 灌电流 10: 12 mA 源 / 灌电流 11: 16 mA 源 / 灌电流

端口 C 锁定寄存器 – PCLOCKR

该寄存器定义了 GPIO 端口 C 的锁定配置。

偏移量: 0x018
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
	PCLKEY								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	23	22	21	20	19	18	17	16	
	PCLKEY								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	15	14	13	12	11	10	9	8	
	保留位								
类型 / 复位									
	7	6	5	4	3	2	1	0	
	PCLOCK								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	字段	描述
[31:16]	PCLKEY	GPIO 端口 C 锁定键 0x5FA0: 使能端口 C 锁定功能 其它: 除能端口 C 锁定功能 为了锁定端口 C 功能, 在该寄存器 PCLKEY 字段写入值 0x5FA0。为了在这个锁定寄存器执行一个成功的写操作, 写入 PCLKEY 字段的值必须为 0x5FA0。如果写入这个字段值不等于 0x5FA0, PCLOCKR 寄存器上的任何写操作将被中止。对 PCLKEY 字段的一个读操作的结果将返回 GPIO 端口 C 锁定状态, 表示 GPIO 端口 C 是否被锁定。如果读值, PCLKEY 字段为 0, 表明 GPIO 端口 C 锁定功能除能。否则, 它表明 GPIO 端口 C 锁定功能使能, 读值等于 1。
[7:0]	PCLOCK _n	GPIO 端口 C 引脚 n 锁定控制位 (n = 0 ~ 7) 0: 端口 C 引脚 n 未锁定 1: 端口 C 引脚 n 锁定 当正确的锁定键应用于 PCLKEY 字段, PCLOCK _n 位用来锁定相应 GPIO 引脚的配置。被锁定的配置包括 PCDIR _n 、PCINEN _n 、PCPU _n 、PCPD _n 、PCOD _n 和 PCDV _n 相关的 GPIO 寄存器。另外, 用来配置为相关 GPIO 引脚的复用功能的 AFIO 寄存器 GPCCFGHR 或 GPCCFGLR 寄存器也将被锁定。请注意, PCLOCKR 寄存器只能写入一次, 这意味着 PCLKEY 和 PCLOCK _n (锁定控制位) 应一起写入, 不能改变直到系统复位或 GPIO 端口 C 发生复位。

端口 C 数据输入寄存器 – PCDINR

该寄存器指定 GPIO 端口 C 的输入数据。

偏移量: 0x01C

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位								
	15	14	13	12	11	10	9	8	
类型 / 复位	保留位								
	7	6	5	4	3	2	1	0	
类型 / 复位	PCDIN								
	RO	0 RO	0 RO	0 RO	0 RO	0 RO	0 RO	0 RO	0

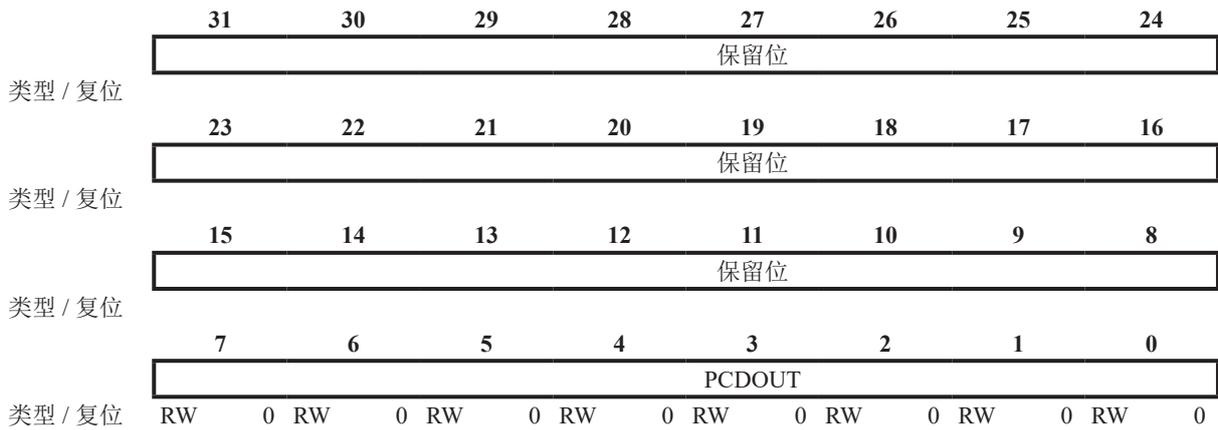
位	字段	描述
[7:0]	PCDINn	GPIO 端口 C 引脚 n 的数据输入位 (n = 0 ~ 7) 0: 引脚 n 的输入数据为 0 1: 引脚 n 的输入数据为 1

端口 C 输出数据寄存器 – PCDOUTR

该寄存器指定 GPIO 端口 C 的输出数据。

偏移量: 0x020

复位值: 0x0000_0000



位	字段	描述
[7:0]	PCDOUTn	GPIO 端口 C 引脚 n 的数据输出位 (n = 0 ~ 7) 0: 引脚 n 的输出数据为 0 1: 引脚 n 的输出数据为 1

端口 C 输出置位 / 复位控制寄存器 – PCSRR

该寄存器用来置位或复位 GPIO 端口 C 输出数据相应的位。

偏移量: 0x024

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	PCRST								
	15	14	13	12	11	10	9	8	
类型 / 复位	保留位								
	7	6	5	4	3	2	1	0	
类型 / 复位	PCSET								
	WO	0	WO	0	WO	0	WO	0	WO

位	字段	描述
[23:16]	PCRSTn	GPIO 端口 C 引脚 n 的输出复位控制位 (n = 0 ~ 7) 0: 不影响 PCDOUn 位 1: 复位 PCDOUn 位 注意, 当该寄存器中的 PCRSTn 位或 PCRR 寄存器中的 PCRSTn 位置位时, PCDOUn 位上的复位功能将生效。
[7:0]	PCSETn	GPIO 端口 C 引脚 n 的输出置位控制位 (n = 0 ~ 7) 0: 不影响 PCDOUn 位 1: 置位 PCDOUn 位 注意, 如果 PCSETn 和 PCRSTn 位同时被置位, PCSETn 位使能的功能有比较高的优先级。

端口 C 输出复位寄存器 – PCRR

该寄存器用来复位 GPIO 端口 C 输出数据相应的位。

偏移量: 0x028

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位								
	15	14	13	12	11	10	9	8	
类型 / 复位	保留位								
	7	6	5	4	3	2	1	0	
类型 / 复位	PCRST								
	WO	0	WO	0	WO	0	WO	0	WO

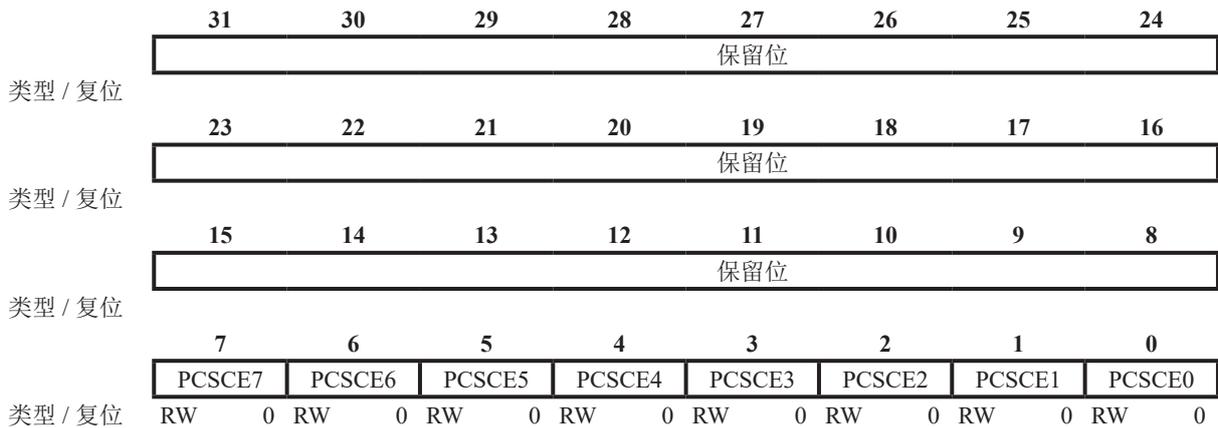
位	字段	描述
[7:0]	PCRSTn	GPIO 端口 C 引脚 n 的输出复位控制位 (n = 0 ~ 7) 0: 不影响 PCDOUTn 位 1: 复位 PCDOUTn 位

端口 C 灌电流增强选择寄存器 – PCSCER

该寄存器用来定义 GPIO 端口 C 的增强灌电流驱动。

偏移量: 0x02C

复位值: 0x0000_0000



位	字段	描述
[7:0]	PCSCE _n	GPIO 端口 C 引脚 n 的灌电流增强选择控制位 (n = 0 ~ 7) 0: 无增强灌电流 1: 增强灌电流

9 复用功能输入 / 输出控制单元 (AFIO)

简介

为了扩大 GPIO 或外设功能使用的灵活性, 每个 I/O 引脚可通过设置 GPxCFGCLR 或 GPxCFGHR 寄存器 (x = A, B, C) 配置为多达 16 个不同的 GPIO 或 IP 功能。根据 IP 资源的使用情况和应用需求, 可以使用外设 I/O 重映射机制来选择各功能从哪个引脚引出。此外, 通过设置 ESSRn 寄存器中 EXTInPIN[3:0] 字段, 可以选择任意的 GPIO 引脚作为 EXTI 中断输入线来触发中断或事件。更多详细信息请参阅 EXTI 部分。

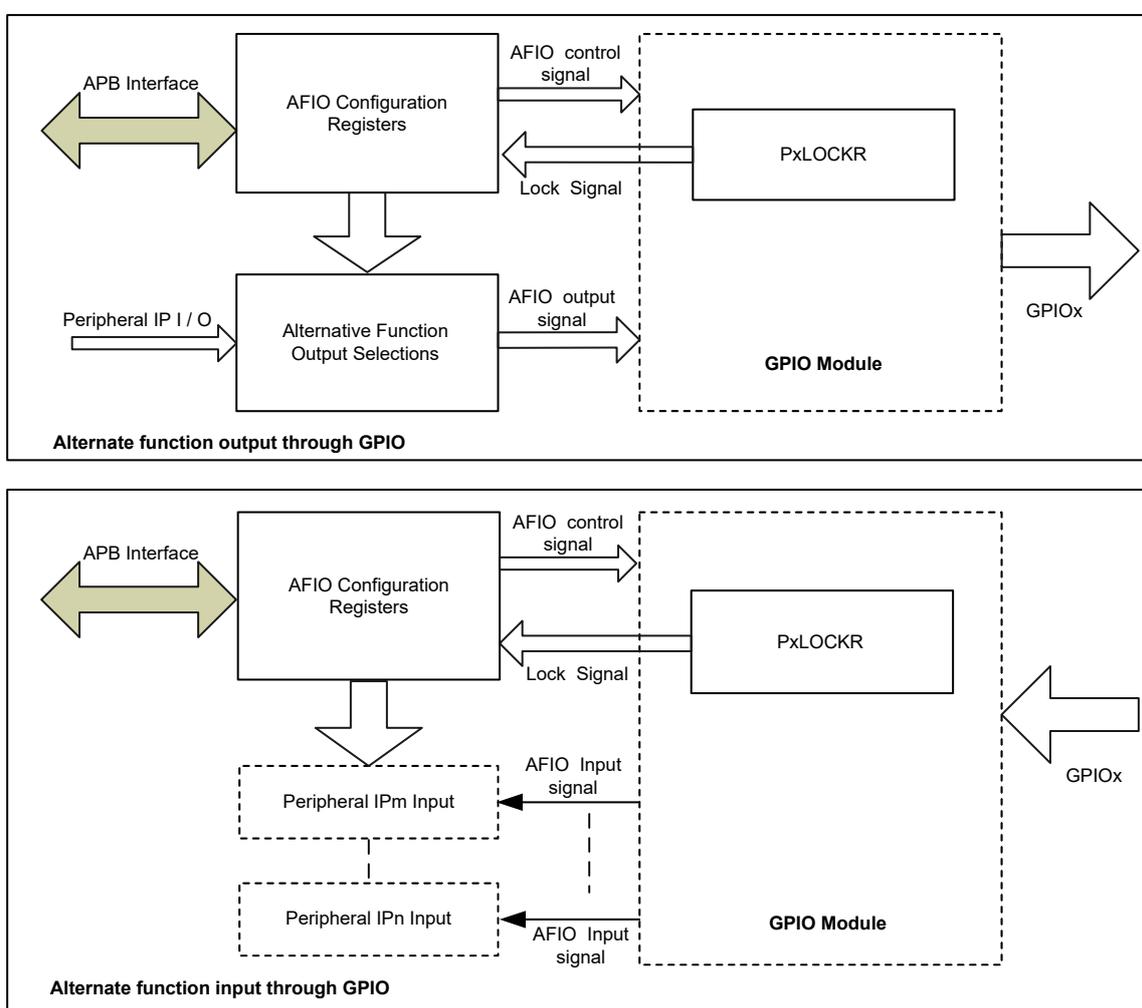


图 21. AFIO 方框图

特性

- APB 从机接口用于寄存器访问
- EXTI 来源选择
- 每个 GPIO 引脚的可配置功能，每个引脚上多达 16 个复用功能
- AFIO 锁定机制

功能描述

外部中断引脚选择

GPIO 引脚连接到 16 个 EXTI 输入线如附图所示。例如，用户可以设置 ESSR0 寄存器中 EXTI0PIN[3:0] 字段为 b0000 来选择 GPIO PA0 引脚作为 EXTI 0 输入。对于所有封装类型并非端口 A ~ C 的全部引脚都可用，详细的引脚信息请参阅数据手册 (Datasheet) 中的引脚图。当相应的引脚不存在时，EXTI_nPIN[3:0] 字段的设置是无效的。

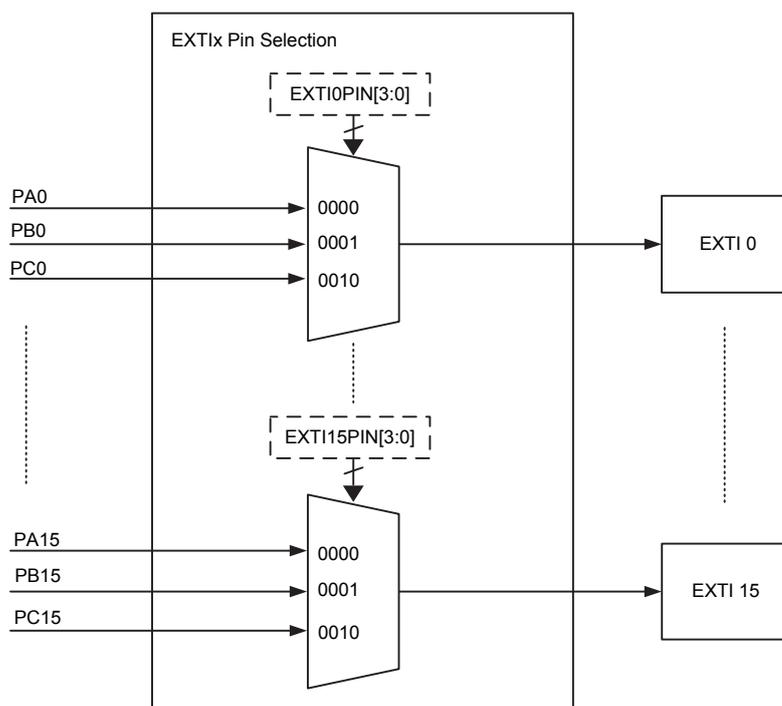


图 22. EXTI 输入通道选择

复用功能

每个 I/O 引脚有多达 16 种可供选择的功能, 可以通过设置 GPxCFGLR 或 GPxCFGHR 寄存器中 PxCFGn[3:0] (n = 0 ~ 15, x = A ~ C) 字段选择。如果引脚选择的备选功能不可用, 即在“复用功能分配”列表中标识为“N/A”, 则该引脚将用作默认复用功能。可选功能 I/O 引脚的详细分配请参阅数据手册 (Datasheet) 中的复用功能列表说明部分。除了这种机制外, 每个外设的可选功能被分配到不同的引脚, 在小封装单片机中优化了外设的可用数量。下面的说明显示 PxCFGn[3:0] 字段的设置。

- PxCFGn[3:0] = 0000: 默认复用功能 (复位后, AF0)
- PxCFGn[3:0] = 0001: 复用功能 1 (AF1)
- PxCFGn[3:0] = 0010: 复用功能 2 (AF2)
-
- PxCFGn[3:0] = 1110: 复用功能 14 (AF14)
- PxCFGn[3:0] = 1111: 复用功能 15 (AF15)

表 19. 外设 AFIO 分配列表范例

AF0	AF1	AF2	AF3	AF4	AF5	AF6	AF7	AF8	AF9	AF10	AF11	AF12	AF13	AF14	AF15
系统默认	GPIO	ADC	N/A	GPTM	SPI	UART	I ² C	N/A	N/A	N/A	N/A	N/A	PWM	N/A	系统其它

锁定机制

GPIO 锁定寄存器 PxLOCKR 同时可以用来锁定 AFIO 配置, 直到复位发生。请参考 GPIO 章节中的锁定机制部分获得更多的信息。

寄存器列表

下表显示 AFIO 寄存器和复位值。

表 20. AFIO 寄存器列表

寄存器	偏移量	描述	复位值
ESSR0	0x000	EXTI 来源选择寄存器 0	0x0000_0000
ESSR1	0x004	EXTI 来源选择寄存器 1	0x0000_0000
GPACFGLR	0x020	GPIO 端口 A 配置低字节寄存器	0x0000_0000
GPACFGHR	0x024	GPIO 端口 A 配置高字节寄存器	0x0000_0000
GPBCFGLR	0x028	GPIO 端口 B 配置低字节寄存器	0x0000_0000
GPBCFGHR	0x02C	GPIO 端口 B 配置高字节寄存器	0x0000_0000
GPCCFGLR	0x030	GPIO 端口 C 配置低字节寄存器	0x0000_0000
GPCCFGHR	0x034	GPIO 端口 C 配置高字节寄存器	0x0000_0000

寄存器描述

EXTI 来源选择寄存器 0 – ESSR0

该寄存器定义了 EXTI0 ~ EXTI7 的 I/O 选项。

偏移量: 0x000

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
	EXTI7PIN				EXTI6PIN				
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	23	22	21	20	19	18	17	16	
	EXTI5PIN				EXTI4PIN				
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	15	14	13	12	11	10	9	8	
	EXTI3PIN				EXTI2PIN				
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
	EXTI1PIN				EXTI0PIN				
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	字段	描述
[31:0]	EXTInPIN[3:0]	EXTIn 引脚选择 (n = 0 ~ 7) 0000: PAn 位被选作 EXTIn 来源信号 0001: PBn 位被选作 EXTIn 来源信号 0010: PCn 位被选作 EXTIn 来源信号 其它: 保留

注: 由于并非所有端口的引脚在所有产品和封装类型都是存在的, 详细的引脚信息请参阅数据手册 (Datasheet) 中的引脚图。当相应的引脚不存在时, EXTIInPIN[3:0] 字段的设置是无效的。

EXTI 来源选择寄存器 1 – ESSR1

该寄存器定义了 EXTI8 ~ EXTI15 的 I/O 选项。

偏移量: 0x004

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
	EXTI15PIN				EXTI14PIN				
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	23	22	21	20	19	18	17	16	
	EXTI13PIN				EXTI12PIN				
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	15	14	13	12	11	10	9	8	
	EXTI11PIN				EXTI10PIN				
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
	EXTI9PIN				EXTI8PIN				
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	字段	描述
[31:0]	EXTInPIN[3:0]	EXTIn 引脚选择 (n = 8 ~ 15) 0000: PAn 位被选作 EXTIn 来源信号 0001: PBn 位被选作 EXTIn 来源信号 0010: PCn 位被选作 EXTIn 来源信号 其它: 保留 注: 由于并非所有端口的引脚在所有产品和封装类型都是存在的, 详细的引脚信息请参阅数据手册 (Datasheet) 中的引脚图。当相应的引脚不存在时, EXTIInPIN[3:0] 字段的设置是无效的。

GPIO 端口 x 配置低字节寄存器 – GPxCFGLR, x = A, B, C

该低字节寄存器定义了 GPIO 端口 x (x = A, B, C) 的复用功能。

偏移量: 0x020, 0x028, 0x030
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
	PxCFG7				PxCFG6				
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	23	22	21	20	19	18	17	16	
	PxCFG5				PxCFG4				
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	15	14	13	12	11	10	9	8	
	PxCFG3				PxCFG2				
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
	PxCFG1				PxCFG0				
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	字段	描述
[31:0]	PxCFGn[3:0]	<p>端口 x 引脚 n 复用功能选择 (n = 0 ~ 7)</p> <p>0000: 端口 x 引脚 n 选择为 AF0</p> <p>0001: 端口 x 引脚 n 选择为 AF1</p> <p>...</p> <p>1110: 端口 x 引脚 n 选择为 AF14</p> <p>1111: 端口 x 引脚 n 选择为 AF15</p> <p>如果该引脚选择作为不可用项目, 即数据手册 (Datasheet) 复用功能列表中被标记为“N/A”的项目时, 该引脚将作为默认的复用功能引脚。请参考单片机数据手册 (Datasheet) 中的复用功能分配列表获取复用功能的更详细信息。</p>

GPIO 端口 x 配置高字节寄存器 – GPxCFGHR, x = A, B, C

该高字节寄存器定义了 GPIO 端口 x (x = A, B, C) 的复用功能。

偏移量: 0x024, 0x02C, 0x034

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
	PxCFG15				PxCFG14				
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	23	22	21	20	19	18	17	16	
	PxCFG13				PxCFG12				
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	15	14	13	12	11	10	9	8	
	PxCFG11				PxCFG10				
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
	PxCFG9				PxCFG8				
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	字段	描述
[31:0]	PxCFGn[3:0]	<p>端口 x 引脚 n 复用功能选择 (n = 8 ~ 15)</p> <p>0000: 端口 x 引脚 n 选择为 AF0</p> <p>0001: 端口 x 引脚 n 选择为 AF1</p> <p>...</p> <p>1110: 端口 x 引脚 n 选择为 AF14</p> <p>1111: 端口 x 引脚 n 选择为 AF15</p> <p>如果该引脚选择作为不可用项目, 即数据手册 (Datasheet) 复用功能列表中被标记为“N/A”的项目时, 该引脚将作为默认的复用功能引脚。请参考单片机数据手册 (Datasheet) 中的复用功能分配列表获取复用功能的更详细信息。</p>

10 嵌套向量中断控制器 (NVIC)

简介

为了减少延迟并提高中断处理效率, Cortex®-M0+ 与内部集成的嵌套向量中断控制器 (NVIC) 紧密耦合。NVIC 对系统异常及外设中断提供的控制包括如使能 / 除能控制、优先级、清除 - 挂起、有效状态报告、软件触发和向量表重置等功能。更多信息请参考 Cortex®-M0+ 技术参考手册。

此外, Cortex®-M0+ 内置简单的 24-bit 向下计数定时器 (SysTick) 用来作为实时操作系统 (RTOS) 的节拍定时器, 或作为一个简单的计数器。SysTick 从预设值向下计数, 当它到达零时, 产生一个系统中断。附表中列出系统异常类型和多个外设中断。

表 21. 异常类型

中断号	异常号	异常类型	优先级	向量地址	描述
—	0	—	—	0x000	初始堆栈指针值
—	1	复位	-3 (最高)	0x004	复位
-14	2	NMI	-2	0x008	不可屏蔽中断。时钟故障中断信号 (时钟控制单元提供的时钟监控功能) 连接到 NMI 输入
-13	3	硬件故障	-1	0x00C	所有故障类型
—	4-10	保留	—	—	—
-5	11	SVCcall	可配置 ⁽¹⁾	0x02C	SVC 指令的系统服务呼叫
—	12-13	保留	—	—	—
-2	14	PendSV	可配置 ⁽¹⁾	0x038	系统服务的可挂起请求
-1	15	SysTick	可配置 ⁽¹⁾	0x03C	SysTick 定时器值递减到零
0	16	LVD	可配置 ⁽²⁾	0x040	低压检测中断
1	17	RTC	可配置 ⁽²⁾	0x044	RTC 总中断
2	18	FMC	可配置 ⁽²⁾	0x048	FMC 总中断
3	19	WKUP	可配置 ⁽²⁾	0x04C	EXTI 事件唤醒中断或外部 WAKEUP 引脚中断
4	20	EXTI0 ~ 1	可配置 ⁽²⁾	0x050	EXTI 输入线 0 ~ 1 中断
5	21	EXTI2 ~ 3	可配置 ⁽²⁾	0x054	EXTI 输入线 2 ~ 3 中断
6	22	EXTI4 ~ 15	可配置 ⁽²⁾	0x058	EXTI 输入线 4 ~ 15 中断
7	23	保留	—	—	—
8	24	ADC	可配置 ⁽²⁾	0x060	ADC 总中断
9	25	保留	—	0x064	—
10	26	保留	—	0x068	—
11	27	保留	—	—	—
12	28	GPTM	可配置 ⁽²⁾	0x070	GPTM 总中断
13	29	保留	—	0x074	—
14	30	保留	—	0x078	—
15	31	PWM0	可配置 ⁽²⁾	0x07C	PWM0 总中断
16	32	PWM1	可配置 ⁽²⁾	0x080	PWM1 总中断
17	33	BFTM	可配置 ⁽²⁾	0x084	BFTM 总中断
18	34	保留	—	0x088	—
19	35	I ² C	可配置 ⁽²⁾	0x08C	I ² C 总中断
20	36	保留	—	0x090	—

中断号	异常号	异常类型	优先级	向量地址	描述
21	37	SPI0	可配置 ⁽²⁾	0x094	SPI0 总中断
22	38	SPI1	可配置 ⁽²⁾	0x098	SPI1 总中断
23	39	保留	—	0x09C	—
24	40	保留	—	0x0A0	—
25	41	UART0	可配置 ⁽²⁾	0x0A4	UART0 总中断
26	42	UART1	可配置 ⁽²⁾	0x0A8	UART1 总中断
27	43	保留	—	0x0AC	—
28	44	保留	—	0x0B0	—
29	45	保留	—	—	—
30	46	保留	—	0x0B8	—
31	47	保留	—	0x0BC	—

注：1. 使用 NVIC 系统处理程序优先级寄存器，异常类型的优先级可改变。更多信息请查阅 Arm® “Cortex®-M0+ Devices Generic User Guide” 文件。

2. 使用 NVIC 中断优先级寄存器，中断优先级可改变。更多信息请查阅 Arm® “Cortex®-M0+ Devices Generic User Guide” 文件。

特性

- 7 个 Cortex®-M0+ 系统异常
- 多达 32 个可屏蔽的外设中断
- 4 个可编程优先级 (2 位用于中断优先级设置)
- 不可屏蔽中断
- 低延时异常和中断处理
- 向量表重置功能
- 内置简单，24-bit 系统定时器，SysTick
 - 24-bit 向下计数器
 - 自动重载功能
 - 当计数器递减到 0，可产生可屏蔽的系统中断
 - HCLK 时钟除以 8 得到 SysTick 的参考时钟源

功能描述

SysTick 校准

NVIC 提供 SysTick 校准值寄存器 SYST_CALIB, 给 RTOS 节拍定时器一个 1 ms 的参考时基, 也可用作其它用途。若 SysTick 参考输入时钟 STCLK 频率为 2.5 MHz (20 MHz 除以 8), 在 SYST_CALIB 寄存器 TENMS 字段有一个固定值 2500, 此重载值用来产生 1 ms 的时间基准。

寄存器列表

下表显示 NVIC 寄存器和复位值。

表 22. NVIC 寄存器列表

寄存器	偏移量	描述	复位值
NVIC 基址 = 0xE000_E000			
SYST_CSR	0x010	SysTick 控制和状态寄存器	0x0000_0000
SYST_RVR	0x014	SysTick 重载值寄存器	未知
SYST_CVR	0x018	SysTick 当前值寄存器	未知
SYST_CALIB	0x01C	SysTick 校准值寄存器	0x4000_09C4
NVIC_ISER	0x100	中断设置使能寄存器	0x0000_0000
NVIC_ICER	0x180	中断清除使能寄存器	0x0000_0000
NVIC_ISPR	0x200	中断设置挂起寄存器	0x0000_0000
NVIC_ICPR	0x280	中断清除挂起寄存器	0x0000_0000
NVIC_IPR0	0x400	中断 0 ~ 3 优先级寄存器	0x0000_0000
NVIC_IPR1	0x404	中断 4 ~ 7 优先级寄存器	0x0000_0000
NVIC_IPR2	0x408	中断 8 ~ 11 优先级寄存器	0x0000_0000
NVIC_IPR3	0x40C	中断 12 ~ 15 优先级寄存器	0x0000_0000
NVIC_IPR4	0x410	中断 16 ~ 19 优先级寄存器	0x0000_0000
NVIC_IPR5	0x414	中断 20 ~ 23 优先级寄存器	0x0000_0000
NVIC_IPR6	0x418	中断 24 ~ 27 优先级寄存器	0x0000_0000
NVIC_IPR7	0x41C	中断 28 ~ 31 优先级寄存器	0x0000_0000
CPUID	0xD00	CPUID 寄存器	0x410C_C601
ICSR	0xD04	中断控制和状态寄存器	0x0000_0000
VTOR	0xD08	向量表偏移寄存器	0x0000_0000
AIRCR	0xD0C	程序中断和复位控制寄存器	0xFA05_0000
SCR	0xD10	系统控制寄存器	0x0000_0000
CCR	0xD14	配置和控制寄存器	0x0000_0204
SHPR2	0xD1C	系统处理程序优先级寄存器 2	0x0000_0000
SHPR3	0xD20	系统处理程序优先级寄存器 3	0x0000_0000

注：更多以上寄存器的详细描述，请参阅 Arm® 文档 “Cortex®-M0+ Devices Generic User Guide”。

11 外部中断 / 事件控制器 (EXTI)

简介

外部中断 / 事件控制器 (EXTI) 包括 16 个边沿 / 电平检测器，都可以分别产生唤醒事件或中断请求。在中断模式中，外部中断有 5 种触发类型，分别为低电平、高电平、下降沿、上升沿和双沿，通过 EXTICFGR_n (n = 0 ~ 15) 寄存器中 SRCnTYPE 字段选择。在唤醒事件模式中，唤醒事件的极性可以通过 EXTIWAKUPPOLR 寄存器中 EXTInWPOL (n = 0 ~ 15) 字段设置。如果 EXTIWAKUPCR 寄存器中 EVWUPIEN 位置位，当相关的唤醒事件发生时及相应的 EXTI 唤醒使能位置位时，将产生 EVWUP 中断。每个 EXTI 输入线也可以个别地被屏蔽。

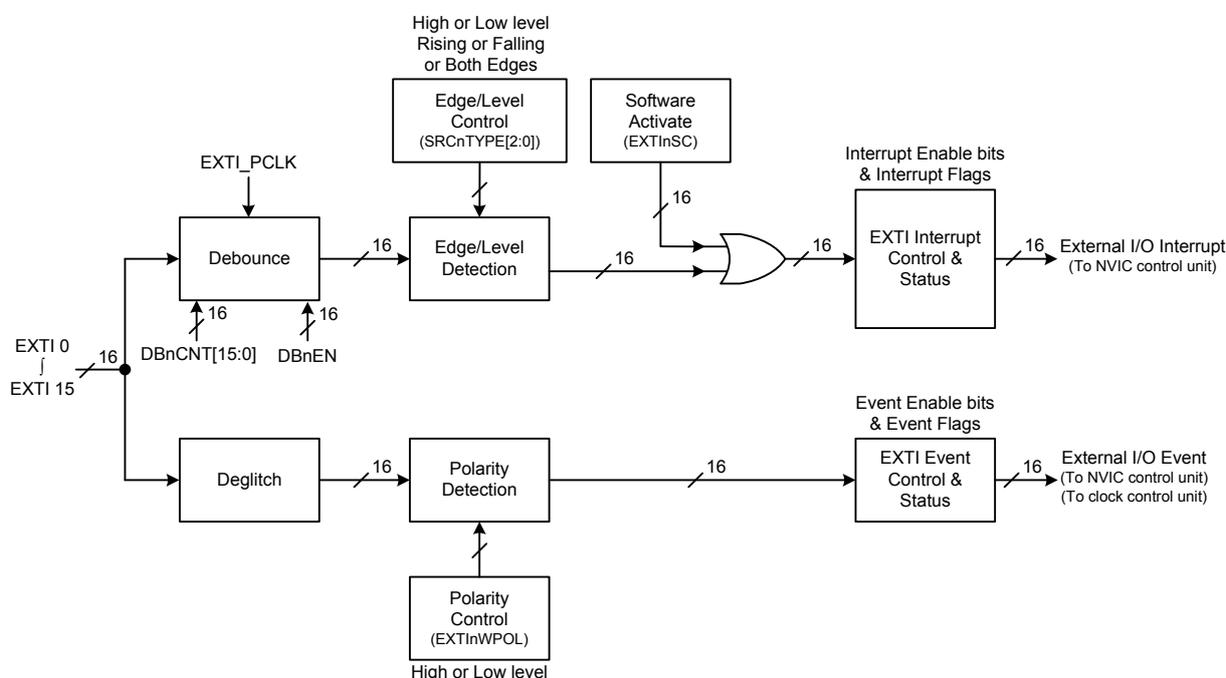


图 23. EXTI 方框图

特性

- 多达 16 个带可配置触发源和类型的 EXTI 输入线
 - 所有 GPIO 引脚可以选择作为 EXTI 触发源
 - 触发源类型包括高电平、低电平、下降沿、上升沿或双沿
- 每个 EXTI 输入线有独立的中断使能、唤醒使能及状态位
- 每个 EXTI 输入线有软件中断触发模式
- 内置去干扰滤波器，用于封锁短脉冲

功能描述

唤醒事件管理

为了从省电模式中唤醒系统，EXTI 控制器提供可以监控外部事件的功能，并把监控到的外部事件送到 CPU 内核和时钟控制单元 CKCU。这些外部事件包括 EXTI 事件、低电压检测事件、WAKEUP 输入引脚和 RTC 唤醒功能。通过在相应的外设配置唤醒事件的使能位，当相应的唤醒事件发生时，通过 EXTI 控制器，唤醒信号将被发送到 CPU 和 CKCU。此外，可以通过软件设置 EXTIWAKUPCR 寄存器 EVWUPIEN 位使能事件唤醒中断功能，当唤醒事件发生时，EXTI 控制器将产生中断。

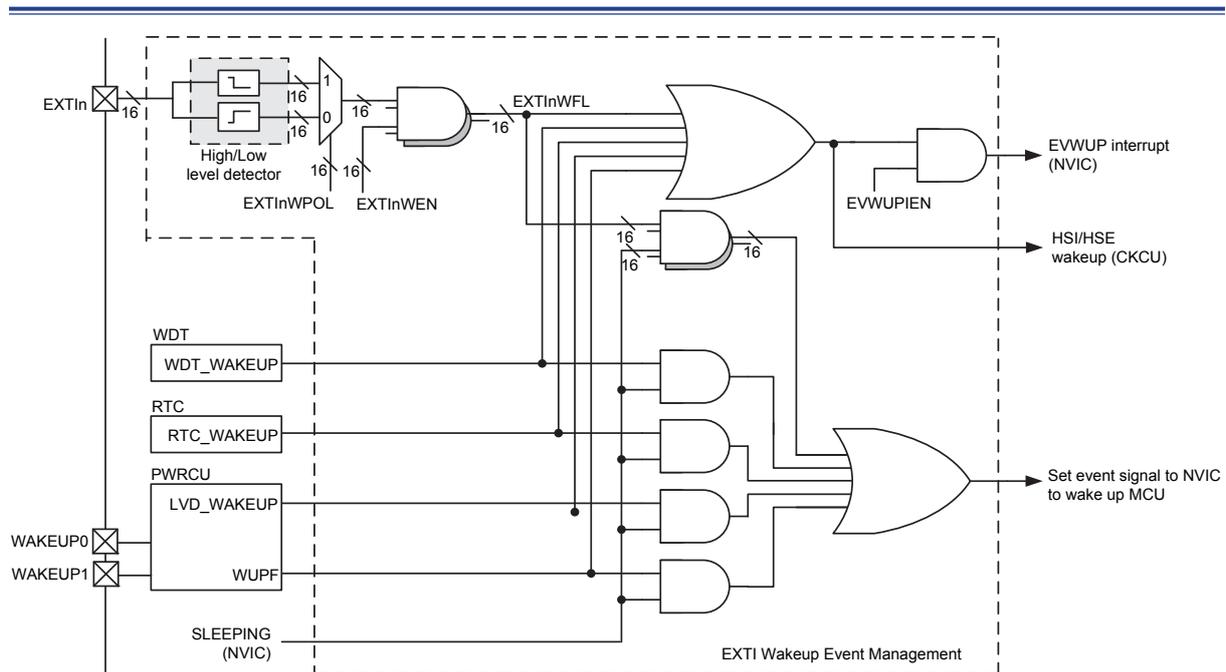


图 24. EXTI 唤醒事件管理

外部中断 / 事件引脚配置

通过配置 AFIO ESSRn (n = 0 ~ 1) 寄存器 EXTInPIN[3:0] 字段, 所有 GPIO 引脚都可被选择作为 EXTI 触发源来触发中断或事件。更多信息请参考 AFIO 部分。

中断和去抖

应用软件可以设置 EXTIn 中断配置寄存器 EXTICFGRn (n = 0 ~ 15) 的 DBnEN 位, 使能相应引脚的去抖功能, 配置 EXTICFGRn 寄存器中 DBnCNT 字段, 以便为特定应用选择一个合适的去抖时间。但中断信号将因去抖功能被延迟。当单片机通过外部中断从省电模式中唤醒时, EXTI 唤醒标志位使中断请求产生。单片机被唤醒且时钟已经恢复后, 被 EXTI 输入线触发的 EXTI 唤醒标志位必须用应用程序对相应位写入 1 来将其清零。所附图表显示 EXTI 输入信号与 EXTI 中断 / 事件请求信号之间的关系。

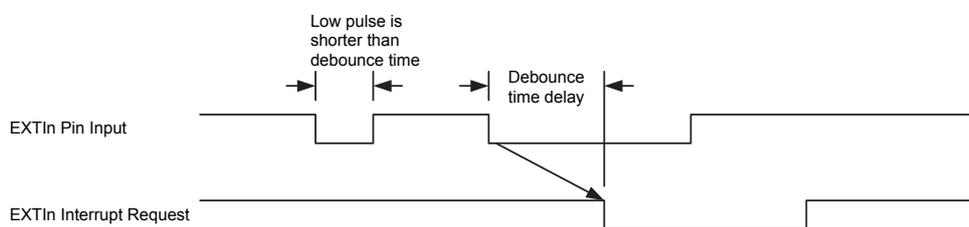


图 25. EXTI 中断去抖功能

寄存器列表

下表显示 EXTI 寄存器和复位值。

表 23. EXTI 寄存器列表

寄存器	偏移量	描述	复位值
EXTICFGR0	0x000	EXTI 中断 0 配置寄存器	0x0000_0000
EXTICFGR1	0x004	EXTI 中断 1 配置寄存器	0x0000_0000
EXTICFGR2	0x008	EXTI 中断 2 配置寄存器	0x0000_0000
EXTICFGR3	0x00C	EXTI 中断 3 配置寄存器	0x0000_0000
EXTICFGR4	0x010	EXTI 中断 4 配置寄存器	0x0000_0000
EXTICFGR5	0x014	EXTI 中断 5 配置寄存器	0x0000_0000
EXTICFGR6	0x018	EXTI 中断 6 配置寄存器	0x0000_0000
EXTICFGR7	0x01C	EXTI 中断 7 配置寄存器	0x0000_0000
EXTICFGR8	0x020	EXTI 中断 8 配置寄存器	0x0000_0000
EXTICFGR9	0x024	EXTI 中断 9 配置寄存器	0x0000_0000
EXTICFGR10	0x028	EXTI 中断 10 配置寄存器	0x0000_0000
EXTICFGR11	0x02C	EXTI 中断 11 配置寄存器	0x0000_0000
EXTICFGR12	0x030	EXTI 中断 12 配置寄存器	0x0000_0000
EXTICFGR13	0x034	EXTI 中断 13 配置寄存器	0x0000_0000
EXTICFGR14	0x038	EXTI 中断 14 配置寄存器	0x0000_0000
EXTICFGR15	0x03C	EXTI 中断 15 配置寄存器	0x0000_0000
EXTICR	0x040	EXTI 中断控制寄存器	0x0000_0000
EXTIEDGFLGR	0x044	EXTI 中断边沿标志寄存器	0x0000_0000
EXTIEDGESR	0x048	EXTI 中断边沿状态寄存器	0x0000_0000
EXTISSCR	0x04C	EXTI 中断软件置位命令寄存器	0x0000_0000
EXTIWAKUPCR	0x050	EXTI 中断唤醒控制寄存器	0x0000_0000
EXTIWAKUPPOLR	0x054	EXTI 中断唤醒极性寄存器	0x0000_0000
EXTIWAKUPFLG	0x058	EXTI 中断唤醒标志寄存器	0x0000_0000

寄存器描述

EXTI 中断配置寄存器 n – EXTICFGRn, n = 0 ~ 15

该寄存器用来定义去抖功能和选择触发类型。

偏移量: 0x000 (0) ~ 0x03C (15)

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24
	DBnEN		SRCnTYPE				保留位	
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	23	22	21	20	19	18	17	16
	保留位							
类型 / 复位								
	15	14	13	12	11	10	9	8
	DBnCNT							
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0
	DBnCNT							
类型 / 复位	RW	0	RW	0	RW	0	RW	0

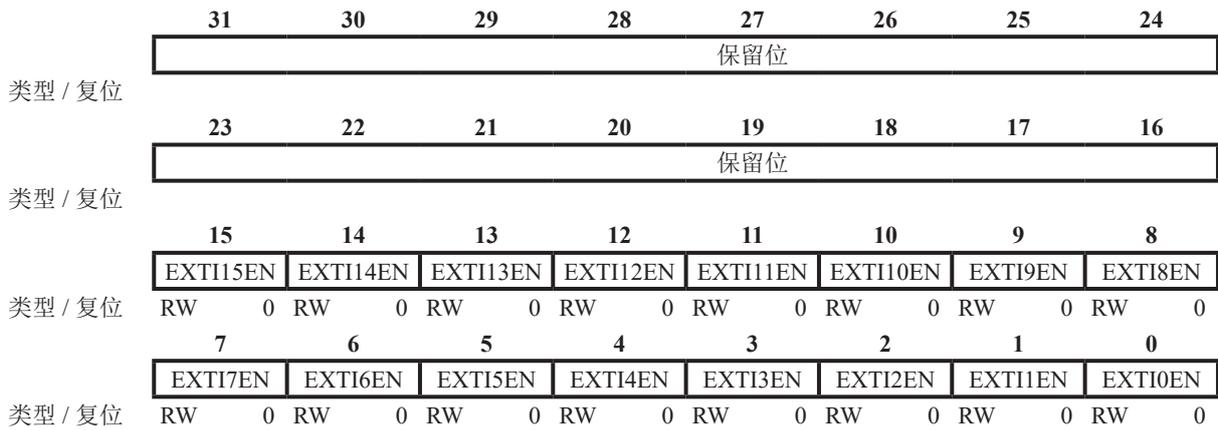
位	字段	描述																								
[31]	DBnEN	EXTIn 去抖电路使能位 (n = 0 ~ 15) 0: 除能去抖电路 1: 使能去抖电路																								
[30:28]	SRCnTYPE	EXTIn 中断源触发类型 (n = 0 ~ 15)																								
		<table border="1"> <thead> <tr> <th colspan="3">SRCnTYPE[2:0]</th> <th>中断源类型</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>低电平有效</td> </tr> <tr> <td>0</td> <td>0</td> <td>1</td> <td>高电平有效</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> <td>下降沿触发</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> <td>上升沿触发</td> </tr> <tr> <td>1</td> <td>X</td> <td>X</td> <td>双沿触发</td> </tr> </tbody> </table>	SRCnTYPE[2:0]			中断源类型	0	0	0	低电平有效	0	0	1	高电平有效	0	1	0	下降沿触发	0	1	1	上升沿触发	1	X	X	双沿触发
SRCnTYPE[2:0]			中断源类型																							
0	0	0	低电平有效																							
0	0	1	高电平有效																							
0	1	0	下降沿触发																							
0	1	1	上升沿触发																							
1	X	X	双沿触发																							
[15:0]	DBnCNT	EXTIn 去抖计数器 (n = 0 ~ 15) 去抖时间根据 DBnCNT × APB (EXTI_PCLK) 时钟周期计算, 并保持足够长的时间以确保对输入信号产生影响。																								

EXTI 中断控制寄存器 – EXTICR

该寄存器用来控制 EXTI 中断。

偏移量: 0x040

复位值: 0x0000_0000



位	字段	描述
[15:0]	EXTInEN	EXTIn 中断使能位 (n = 0 ~ 15) 0: 除能 EXTI 输入线 n 中断 1: 使能 EXTI 输入线 n 中断

EXTI 中断边沿标志寄存器 – EXTIEDGEFLGR

该寄存器用来说明是否已检测到 EXTI 边沿。

偏移量: 0x044

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位								
	15	14	13	12	11	10	9	8	
类型 / 复位	EXTI15EDF	EXTI14EDF	EXTI13EDF	EXTI12EDF	EXTI11EDF	EXTI10EDF	EXTI9EDF	EXTI8EDF	
	WC	0	WC	0	WC	0	WC	0	WC
	7	6	5	4	3	2	1	0	
类型 / 复位	EXTI7EDF	EXTI6EDF	EXTI5EDF	EXTI4EDF	EXTI3EDF	EXTI2EDF	EXTI1EDF	EXTI0EDF	
	WC	0	WC	0	WC	0	WC	0	WC

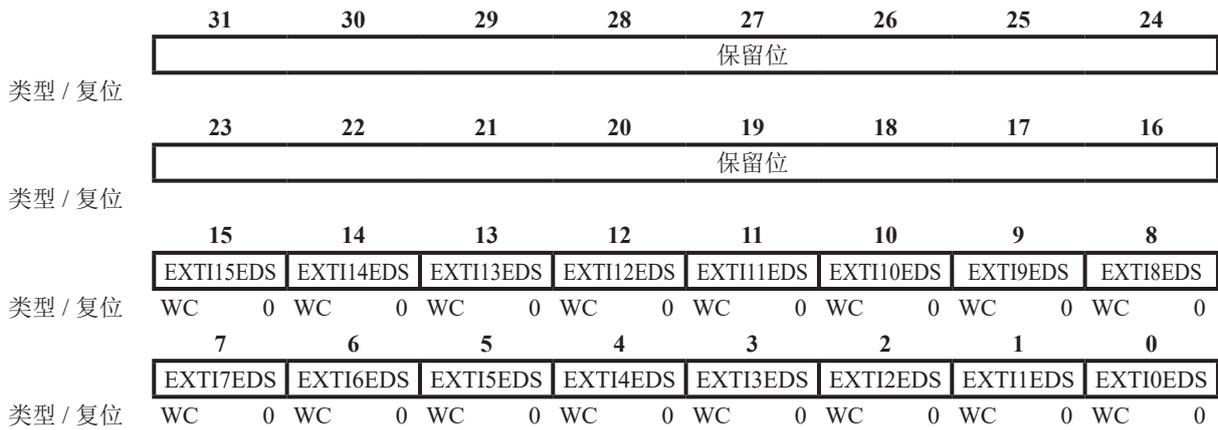
位	字段	描述
[15:0]	EXTInEDF	EXTIn 边沿检测标志位 (n = 0 ~ 15) 0: 没有检测到边沿 1: 检测到上升沿或下降沿 当相应的 EXTI 输入线检测到上升沿或下降沿, 此位被硬件置位。通过写 1 来清零。

EXTI 中断边沿状态寄存器 – EXTIEDGESR

该寄存器用来说明检测到的 EXTI 边沿的极性。

偏移量: 0x048

复位值: 0x0000_0000



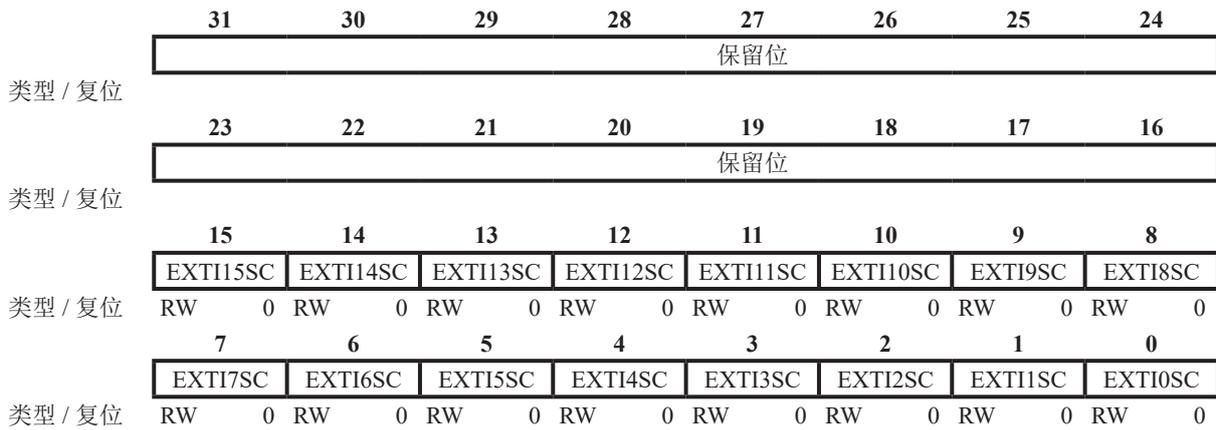
位	字段	描述
[15:0]	EXTInEDS	EXTIn 边沿检测状态位 (n = 0 ~ 15) 0: 检测到下降沿 1: 检测到上升沿 通过写 1 来清零。

EXTI 中断软件置位命令寄存器 – EXTISSCR

该寄存器用来激活 EXTI 中断。

偏移量: 0x04C

复位值: 0x0000_0000



位	字段	描述
[15:0]	EXTInSC	EXTIn 软件置位命令 (n = 0 ~ 15) 0: 未激活相关 EXTI 中断 1: 激活相关 EXTI 中断

EXTI 中断唤醒控制寄存器 – EXTIWAKUPCR

该寄存器用来控制 EXTI 中断和唤醒功能。

偏移量: 0x050

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24
	E VWUPIEN		保留位					
类型 / 复位	RW	0						
	23	22	21	20	19	18	17	16
	保留位							
	15	14	13	12	11	10	9	8
	EXTI15WEN		EXTI14WEN		EXTI13WEN		EXTI12WEN	
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0
	EXTI7WEN		EXTI6WEN		EXTI5WEN		EXTI4WEN	
类型 / 复位	RW	0	RW	0	RW	0	RW	0

位	字段	描述
[31]	E VWUPIEN	EXTI 事件唤醒中断使能位 0: 除能 EVWUP 中断 1: 使能 EVWUP 中断
[15:0]	EXTInWEN	EXTIn 唤醒使能位 (n = 0 ~ 15) 0: 除能省电模式唤醒 1: 使能省电模式唤醒

EXTI 中断唤醒极性寄存器 – EXTIWAKUPPOLR

该寄存器用来选择 EXTI 输入线中断唤醒极性。

偏移量: 0x054

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位								
	15	14	13	12	11	10	9	8	
类型 / 复位	EXTI15WPOL	EXTI14WPOL	EXTI13WPOL	EXTI12WPOL	EXTI11WPOL	EXTI10WPOL	EXTI9WPOL	EXTI8WPOL	
	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
类型 / 复位	EXTI7WPOL	EXTI6WPOL	EXTI5WPOL	EXTI4WPOL	EXTI3WPOL	EXTI2WPOL	EXTI1WPOL	EXTI0WPOL	
	RW	0	RW	0	RW	0	RW	0	RW

位	字段	描述
[15:0]	EXTInWPOL	EXTIn 唤醒极性 (n = 0 ~ 15) 0: 高电平有效 1: 低电平有效

EXTI 中断唤醒标志寄存器 – EXTIWAKUPFLG

该寄存器是 EXTI 中断唤醒标志寄存器。

偏移量: 0x058

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位								
	15	14	13	12	11	10	9	8	
类型 / 复位	EXTI15WFL	EXTI14WFL	EXTI13WFL	EXTI12WFL	EXTI11WFL	EXTI10WFL	EXTI9WFL	EXTI8WFL	
	WC	0	WC	0	WC	0	WC	0	WC
	7	6	5	4	3	2	1	0	
类型 / 复位	EXTI7WFL	EXTI6WFL	EXTI5WFL	EXTI4WFL	EXTI3WFL	EXTI2WFL	EXTI1WFL	EXTI0WFL	
	WC	0	WC	0	WC	0	WC	0	WC

位	字段	描述
[15:0]	EXTInWFL	EXTIn 唤醒标志位 (n = 0 ~ 15) 0: 没发生唤醒 1: 系统被 EXTIn 唤醒 通过写 1 来清零。

12 模数转换器 (ADC)

简介

单片机内建一个 12-bit 多通道 A/D 转换器，总共有 14 个复用通道，包括 12 个可供外部模拟信号输入的外部通道和 2 个内部通道。模拟看门狗功能用来确保 A/D 转换器的输入电压保持在一个特定的阈值。当输入电压高于或低于设定的阈值，将产生中断。有三种转换模式用来把模拟信号转换成数字信号。A/D 转换器可以工作在单次转换，连续和非连续转换模式，且提供一个 16-bit 寄存器用来储存转换后的数据。

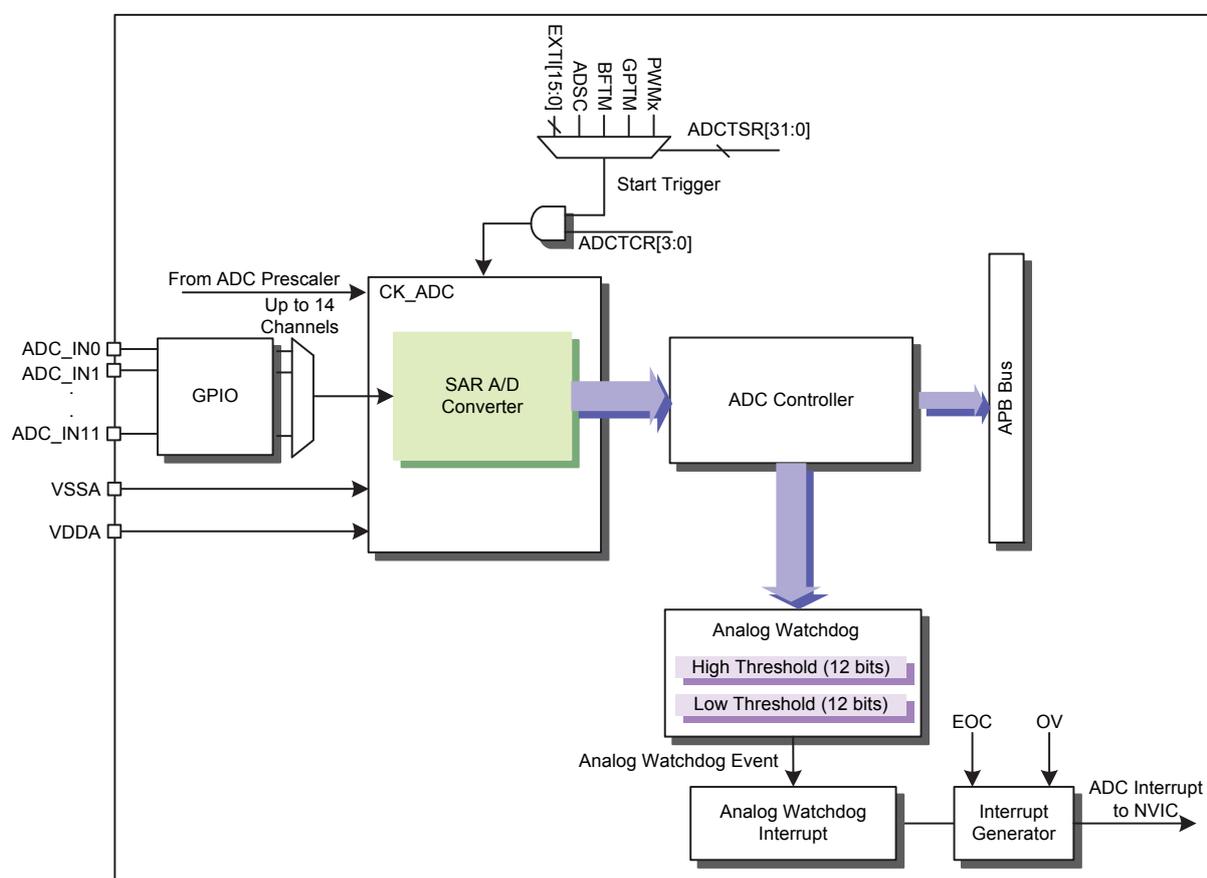


图 26. ADC 方框图

特性

- 12-bit SAR ADC 内核
- 高达 1 Msps 转换速率
- 12 个外部模拟输入通道
- 2 个内部模拟输入通道用于参考电压检测
- 通道采样时间可编程
- 多达 8 个可编程转换通道序列和转换结果专用寄存器
- 3 种转换模式
 - 单次转换模式
 - 连续转换模式
 - 非连续转换模式
- 用于预定义的电压范围监控的模拟看门狗
 - 下限 / 上限阈值寄存器
 - 产生中断
- 多种触发启动源用于转换模式
 - 软件触发
 - EXTI – 外部中断输入引脚
 - GPTM 触发
 - PWM0 / PWM1 触发
 - BFTM 触发
- 多种中断产生
 - 单次转换结束
 - 子组转换结束
 - 周期转换结束
 - 模拟看门狗
 - 数据寄存器覆盖

功能描述

ADC 时钟设置

ADC 时钟 (CK_ADC) 由时钟控制器提供, 与 AHB 时钟 HCLK 同步, 为 HCLK 时钟分频。更多信息请参考时钟控制单元的章节。注意, ADC 需要至少两个 ADC 时钟周期在上电和掉电情况下相互切换 (ADEN 位 = 0)。

通道选择

A/D 转换器支持 14 个多路复用通道, 转换结果依组转换次序存放对应数据寄存器中。一个转换组可以安排一个转换序列, 此序列长度可为 1 到 8 个通道。例如, 可依 CH2、CH4、CH7、CH5、CH6、CH3、CH0 和 CH1 的序列依次进行转换。

一个组可由多达 8 次的转换组成。一个转换组使用的通道通过 ADCLST0 ~ ADCLST1 寄存器选择。转换序列的总长度通过 ADCCR 寄存器 ADSEQL[2:0] 位设置。

在转换过程中, 修改 ADCCR 或 ADCLSTn 寄存器值将复位当前的转换。之后需要一个新的转换启动触发信号重新开始一个新的转换。

转换模式

A/D 转换器有三种工作转换模式。这三种转换模式分别是单次转换模式、连续转换模式和非连续转换模式。详细信息以下逐一描述。

单次转换模式

在单次转换模式中, 当 A/D 转换触发事件发生时, A/D 转换器在 A/D 转换列表寄存器 ADCLSTn 指定的通道中按照特定的顺序执行转换周期。当 ADCCR 寄存器中的 A/D 转换模式字段的 ADMODE[1:0] 设置为 0x0, A/D 转换器将运行在单次转换模式。所有的触发源都可启动单次转换, 触发源由触发控制寄存器 ADCTCR 和触发源寄存器 ADCTSR 决定。

转换后:

- 转换后的数据将被储存在 16-bit 寄存器 ADCDRy (y = 0 ~ 7)。
- 单次采样转换完成时, ADCIRAW 寄存器中的 ADC 单次采样转换结束事件原始状态标志位 ADIRAWS 将被置位。
- 单次采样转换完成后, 如果 ADCIER 寄存器中的 ADIES 位使能, 将产生中断。
- 一个组的转换周期结束后, 如果 ADCIER 寄存器中 ADIEC 位使能, 将产生中断。

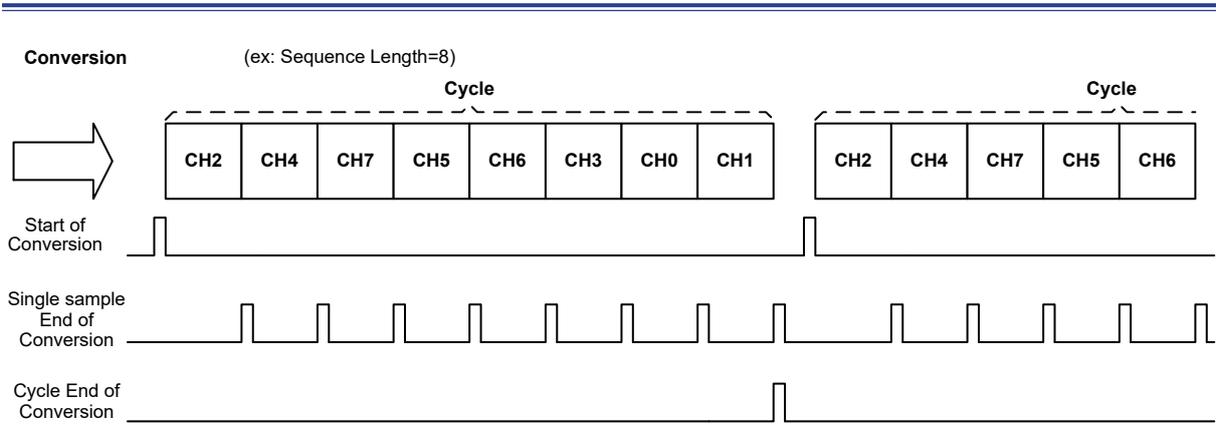


图 27. 单次转换模式

连续转换模式

在连续转换模式下，一个通道组转换完成后，无需额外的 A/D 转换启动触发信号，重复转换周期将自动重新启动。当 A/D 转换模式字段的 $ADM\text{MODE}[1:0]$ 设置为 $0x2$ ，A/D 转换器将运行在连续转换模式。所有的触发源都可启动连续转换，触发源由触发控制寄存器 $ADCTCR$ 和触发源寄存器 $ADCTSR$ 决定。

转换后：

- 转换后的数据将被储存在 16-bit $ADCDRy$ ($y = 0 \sim 7$) 寄存器。
- 当转换周期结束时， $ADCIRAW$ 寄存器中的 ADC 组转换事件周期结束原始状态标志位 $ADIRAWC$ 将被置位。
- 组转换周期结束后，如果 $ADCIER$ 寄存器 $ADIEC$ 位使能，将产生中断。

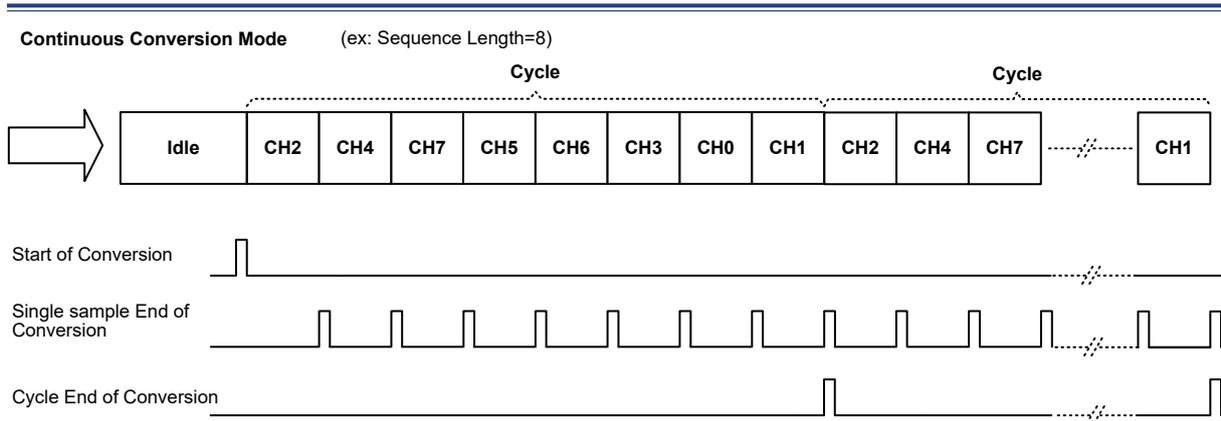


图 28. 连续转换模式

非连续转换模式

ADCCR 寄存器中 A/D 转换模式位字段 ADMODE[1:0] 设置为 0x3, A/D 转换器将运行在通道组的非连续转换模式。要转换的组最多可以有 8 个通道, 可以以一个特定的顺序排列, 通过 ADCLSTn ($n = 0 \sim 1$) 寄存器配置。在这种模式下, 每次触发事件发生时, 以短序列形式转换组数据, 称为 A/D 转换子组。子组长度由 ADCCR 寄存器中的 ADSUBL[2:0] 字段定义。所有的触发源都可启动非连续转换, 触发源由触发控制寄存器 ADCTCR 和触发源寄存器 ADCTSR 决定。

在非连续转换模式, A/D 转换器开始下一组的 n 个转换, 数字 n 是 ADSUBL 字段定义子组长度。当触发事件发生时, 以一个特定序列转换的通道由 ADCLSTn 寄存器定义。此 n 次转换完成后, ADCIRAW 寄存器中子组 EOC 中断原始标志位 ADIRAWG 将被置位。此时 A/D 转换器无法继续执行后面的 n 次转换, 需等到下一个触发事件发生。所有的组通道 (总数由 ADCCR 寄存器 ADSEQL[2:0] 位定义) 完成它们的转换后, 转换周期结束, 此时, ADCIRAW 寄存器中周期 EOC 中断的原始标志位 ADIRAWC 将被置位。如果所有子组通道都被转换后, 即一个完整的转换周期已完成, 当一个新触发事件发生, 转换将从第一子组重新开始。

例如:

A/D 转换子组的长度 = 3 ($ADSUBL = 2$), 序列长度 = 8 ($ADSEQL = 7$), 被转换的通道 = 2、4、7、5、6、3、0、1 (ADCLSTn 寄存器中定义的转换顺序)。

- 触发 1: 被转换的子组通道是 CH2、CH4 和 CH7, 子组转换完成后 ADIRAWG 标志位被置位。
- 触发 2: 被转换的子组通道是 CH5、CH6 和 CH3, 子组转换完成后 ADIRAWG 标志位被置位。
- 触发 3: 被转换的子组通道是 CH0 和 CH1, 子组转换完成后 ADIRAWG 标志位被置位。一个转换周期结束 (EOC) 中断原标志位 ADIRAWC 也被置位。
- 触发 4: 被转换的子组通道是 CH2、CH4 和 CH7, 子组转换完成后 ADIRAWG 标志位被置位 – 从第一组重新开始转换序列。

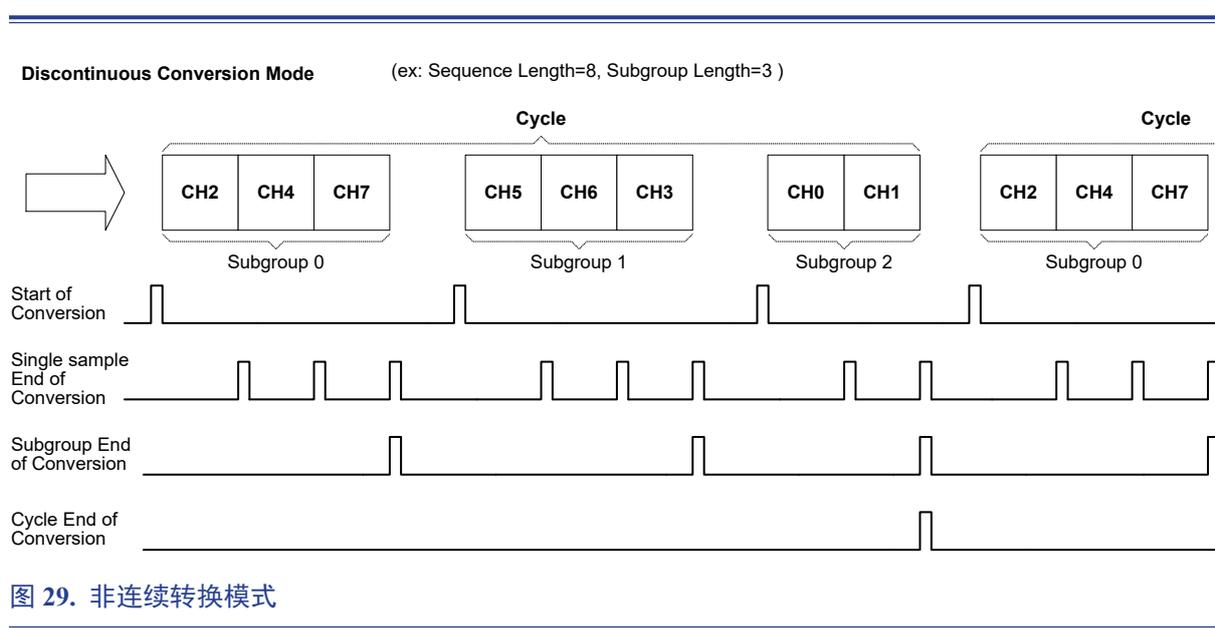


图 29. 非连续转换模式

外部事件启动转换

A/D 转换可以通过软件触发, 通用定时器模块 (GPTM) 事件, PWM 触发事件, 基本功能定时器模块 (BFTM) 事件或外部触发 (EXTI) 启动。每个触发源可以通过设置 ADCTCR 寄存器中相应的使能控制位启用, 然后通过配置 ADCTSRS 寄存器相关的选择位选择触发源从而开始一个组通道转换。

当 ADCTCR 寄存器中软件触发使能位 ADSW 置为 1 时, 可通过设置用于组通道的 ADCTSRS 寄存器的软件触发位 ADSC 来启动 A/D 转换器的转换。A/D 转换器开始转换模拟数据后, 软件触发位 ADSC 将被自动清零。

定时器事件也可以触发组转换开始 A/D 转换。定时器事件包括一个 GPTM 或 PWM 的主机触发输出 MTO, 四个 GPTM 或 PWM 通道输出 CH0 ~ CH3 和一个 BFTM 触发输出。如果相应的定时器触发使能位置为 1, 触发输出或定时器通道事件已通过相应的定时器事件选择位选择, 当一个选定的触发事件的上升沿发生时, A/D 转换器启动转换。

除了内部触发源, 外部触发事件也可以触发 A/D 转换器开始转换。外部触发事件是来自外部输入线 EXTI_n。如果外部触发使能位 ADEXTI 设置为 1, 并通过配置 ADCTSRS 寄存器中的 ADEXTIS 字段, EXTI 单元中定义的 EXTI 输入线的有效边沿发生时, A/D 转换器将启动一次转换。

采样时间设定

可根据输入电压源的输入阻抗配置转换通道采样时间。采样时间必须足够，以确保输入电压源可将 A/D 转换器内部采样保持电容充电至输入电压的水平。每个转换通道都使用相同的采样时间采样。通过修改 ADCSTR 寄存器中 ADST[7:0] 位，可以设置模拟输入信号的采样时间。

总转换时间 (T_{conv}) 使用下列公式计算：

$$T_{conv} = T_{sampling} + T_{Latency}$$

最小采样时间 $T_{sampling} = 1.5$ 个周期 ($ADST[7:0] = 0$)，通道转换的最小延迟 $T_{Latency} = 12.5$ 个周期。

例如：

A/D 转换时钟 $CK_ADC = 14$ MHz，采样时间 = 1.5 个周期：

$$T_{conv} = 1.5 + 12.5 = 14 \text{ 周期} = 1 \mu s$$

数据格式

A/D 转换结果通过 ADCDRy 寄存器读取，其数据格式见表 24。

表 24. ADCDR[15:0] 寄存器中的数据格式

描述	ADCDR 寄存器数据格式
右对齐	“0_0_0_0_d11_d10_d9_d8_d7_d6_d5_d4_d3_d2_d1_d0”

模拟看门狗

A/D 转换器，包括一个看门狗功能用来监视正被转换的数据。看门狗监视功能有两种阈值，即看门狗下限阈值和看门狗上限阈值，它们分别由看门狗阈值寄存器 ADCTR 中的 ADLT 和 ADUT 字段定义。看门狗监视功能通过看门狗控制寄存器 ADCWCR 中看门狗上限阈值和下限阈值监视功能使能位 ADWUE 和 ADWLE 设置使能。通过配置 ADWCH 和 ADWALL 位可以指定要监控的通道。当转换后的数据小于或大于由看门狗阈值寄存器 ADCTR 中的 ADLT 和 ADUT 字段定义的下限阈值或上限阈值时，若看门狗下限或上限阈值监控功能使能，则置位 ADCIRAW 寄存器中的下限阈值中断标志 ADIRAWL 或上限阈值中断标志 ADIRAWU。如果下限阈值或上限阈值中断原始位被置位，且通过设置 ADCIER 寄存器中 ADIEL 或 ADIEU 使相应的中断使能，将产生 A/D 看门狗下限阈值或上限阈值中断。

中断

当 A/D 转换完成，一个转换结束 EOC 事件将发生。有三种 EOC 事件分别为单次采样 EOC、子组 EOC 和周期 EOC。当单个通道转换已经完成，单次采样 EOC 事件发生，且 ADCIRAW 寄存器中单次采样 EOC 中断原始标志位 ADIRAWS 被置位。当子组转换已经完成，子组 EOC 事件发生，且 ADCIRAW 寄存器中子组 EOC 中断原始标志位 ADIRAWG 被置位。当周期转换已经完成，周期 EOC 事件发生，且 ADCIRAW 寄存器中周期 EOC 中断原始标志位 ADIRAWC 被置位。当单次采样 EOC，子组 EOC 或周期 EOC 原始标志位被置位，且 ADCIER 寄存器中对应的中断使能位 ADIES、ADIEG 或 ADIEC 位置 1，将产生相关中断。

转换完成后，将 12 位数字数据存储在与相关的 ADCDRy 寄存器中，相关的数据有效标志位 ADVLDy 将被置位。转换后的数据通过应用程序读取，读取后，数据有效标志位 ADVLDy 将自动清零。若是 ADVLDy 置位后数据尚未被读取，此时又有新数据写入，则将发生数据覆盖事件，ADCIRAW 寄存器中数据覆盖中断原始标志位 ADIRAWO 位将被置位。当相关的数据覆盖原标志位被置位，如果 ADCIER 寄存器 ADIEO 中断使能位置为 1，将产生数据覆盖中断。

如果 A/D 看门狗监视功能使能，通道转换后的数据是低于下限阈值或高于上限阈值，ADCIRAW 寄存器中看门狗下限或上限阈值中断原始标志位 ADIRAWL 或 ADIRAWU 被置位。当 ADIRAWL 或 ADIRAWU 位被置位，ADCIER 寄存器中相应的中断使能位 ADIEL 或 ADIEU 置 1，将产生看门狗下限或上限阈值中断。

A/D 转换中断清除位用来清除相关的 A/D 转换中断原始位和中断状态位。在 A/D 转换中断清除寄存器 ADCICLR 中将对应的 A/D 转换中断清除位写入 1，将清除相应的 A/D 转换中断原始状态和中断状态位。这些位被软件设置为 1 后，由硬件自动清零。

寄存器列表

下表显示 A/D 转换器寄存器和复位值。

表 25. A/D 转换器寄存器列表

寄存器	偏移量	描述	复位值
ADCCR	0x000	ADC 转换控制寄存器	0x0000_0000
ADCLST0	0x004	ADC 转换列表寄存器 0	0x0000_0000
ADCLST1	0x008	ADC 转换列表寄存器 1	0x0000_0000
ADCSTR	0x020	ADC 输入采样时间寄存器	0x0000_0000
ADCDR0	0x030	ADC 转换数据寄存器 0	0x0000_0000
ADCDR1	0x034	ADC 转换数据寄存器 1	0x0000_0000
ADCDR2	0x038	ADC 转换数据寄存器 2	0x0000_0000
ADCDR3	0x03C	ADC 转换数据寄存器 3	0x0000_0000
ADCDR4	0x040	ADC 转换数据寄存器 4	0x0000_0000
ADCDR5	0x044	ADC 转换数据寄存器 5	0x0000_0000
ADCDR6	0x048	ADC 转换数据寄存器 6	0x0000_0000
ADCDR7	0x04C	ADC 转换数据寄存器 7	0x0000_0000
ADCTCR	0x070	ADC 触发控制寄存器	0x0000_0000
ADCTSR	0x074	ADC 触发源寄存器	0x0000_0000
ADCWCR	0x078	ADC 看门狗控制寄存器	0x0000_0000
ADCTR	0x07C	ADC 看门狗阈值寄存器	0x0000_0000
ADCIER	0x080	ADC 中断使能寄存器	0x0000_0000
ADCIRAW	0x084	ADC 中断原始状态寄存器	0x0000_0000
ADCISR	0x088	ADC 中断状态寄存器	0x0000_0000
ADCICLR	0x08C	ADC 中断清除寄存器	0x0000_0000

寄存器描述

ADC 转换控制寄存器 – ADCCR

该寄存器定义了模式的设置，序列长度和 ADC 转换模式的子组长度。请注意，A/D 转换进行中时，若 ADCCR 寄存器的内容发生了变化，转换将被迫终止，A/D 转换器将返回到空闲状态。应用程序发出下一个命令前须等待至少一个 A/D 转换时钟 CK_ADC。

偏移量： 0x000
复位值： 0x0000_0000

	31	30	29	28	27	26	25	24				
类型 / 复位	保留位											
	23	22	21	20	19	18	17	16				
类型 / 复位	保留位					ADSUBL						
						RW	0	RW	0	RW	0	
	15	14	13	12	11	10	9	8				
类型 / 复位	保留位					ADSEQL						
						RW	0	RW	0	RW	0	
	7	6	5	4	3	2	1	0				
类型 / 复位	ADCEN	ADCRST	保留位				ADMODE					
	RW	0	RW	0					RW	0	RW	0

位	字段	描述										
[18:16]	ADSUBL	ADC 转换子组长度 非连续模式下的每个子组的转换通道长度由 ADSUBL 字段定义。子组长度 = ADSUBL[2:0] + 1。如果序列长度 (ADSEQL[2:0] + 1) 不是子组长度 (ADSUBL[2:0] + 1) 的倍数，最后一个子组将是没有被转换的剩余的通道。										
[10:8]	ADSEQL	ADC 转换长度 000：由 ADCLST0 寄存器中的 ADSEQ0 位指定的通道进行转换 其它：序列长度 = ADSEQL[2:0] + 1 ADSEQL 字段指定转换组的整个转换序列长度。										
[7]	ADCEN	ADC 使能 0：除能 1：使能 当该位清零，ADC 将除能，且 CK_ADC 时钟也将关闭。										
[6]	ADCRST	ADC 复位 0：无影响 1：复位 A/D 转换器 (A/D 转换控制器除外)										
[1:0]	ADMODE	ADC 转换模式										
	ADMODE[1:0]	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 15%;">模式</th> <th style="width: 85%;">描述</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">00</td> <td>单次模式 启动触发后，在指定通道整体转换一次</td> </tr> <tr> <td style="text-align: center;">01</td> <td>保留</td> </tr> <tr> <td style="text-align: center;">10</td> <td>连续模式 启动触发后，在特定通道按序列进行连续转换直到转换模式改变。</td> </tr> <tr> <td style="text-align: center;">11</td> <td>非连续模式 启动触发后，当前的子组将进行转换。当最后一个子组完成转换，若其它开始触发发生，从第一子组重新启动转换。</td> </tr> </tbody> </table>	模式	描述	00	单次模式 启动触发后，在指定通道整体转换一次	01	保留	10	连续模式 启动触发后，在特定通道按序列进行连续转换直到转换模式改变。	11	非连续模式 启动触发后，当前的子组将进行转换。当最后一个子组完成转换，若其它开始触发发生，从第一子组重新启动转换。
模式	描述											
00	单次模式 启动触发后，在指定通道整体转换一次											
01	保留											
10	连续模式 启动触发后，在特定通道按序列进行连续转换直到转换模式改变。											
11	非连续模式 启动触发后，当前的子组将进行转换。当最后一个子组完成转换，若其它开始触发发生，从第一子组重新启动转换。											

ADC 转换列表寄存器 0 – ADCLST0

该寄存器定义了 ADC 的转换序列顺序 No.0 ~ No.3。

偏移量: 0x004
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24		
	保留位			ADSEQ3						
类型 / 复位				RW	0	RW	0	RW	0	
	23	22	21	20	19	18	17	16		
	保留位			ADSEQ2						
类型 / 复位				RW	0	RW	0	RW	0	
	15	14	13	12	11	10	9	8		
	保留位			ADSEQ1						
类型 / 复位				RW	0	RW	0	RW	0	
	7	6	5	4	3	2	1	0		
	保留位			ADSEQ0						
类型 / 复位				RW	0	RW	0	RW	0	

位	字段	描述
[28:24]	ADSEQ3	ADC 转换序列选择 3 定义 ADC 转换第三序列的 ADC 输入通道 0x00: ADC_IN0 0x01: ADC_IN1 0x02: ADC_IN2 0x03: ADC_IN3 0x04: ADC_IN4 0x05: ADC_IN5 0x06: ADC_IN6 0x07: ADC_IN7 0x08: ADC_IN8 0x09: ADC_IN9 0x0A: ADC_IN10 0x0B: ADC_IN11 0x0C: 模拟地, VSSA (V _{REF-}) 0x0D: 模拟电源, VDDA (V _{REF+}) 0x0E ~ 0x1F: 设置无效。这些值不能被选择, 因为它们可能导致 ADC 异常操作。
[20:16]	ADSEQ2	ADC 转换序列选择 2
[12:8]	ADSEQ1	ADC 转换序列选择 1
[4:0]	ADSEQ0	ADC 转换序列选择 0

ADC 转换列表寄存器 1 – ADCLST1

该寄存器定义了 ADC 的转换序列顺序 No.4 ~ No.7。

偏移量: 0x008

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24			
	保留位			ADSEQ7							
类型 / 复位				RW	0	RW	0	RW	0	RW	0
	23	22	21	20	19	18	17	16			
	保留位			ADSEQ6							
类型 / 复位				RW	0	RW	0	RW	0	RW	0
	15	14	13	12	11	10	9	8			
	保留位			ADSEQ5							
类型 / 复位				RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0			
	保留位			ADSEQ4							
类型 / 复位				RW	0	RW	0	RW	0	RW	0

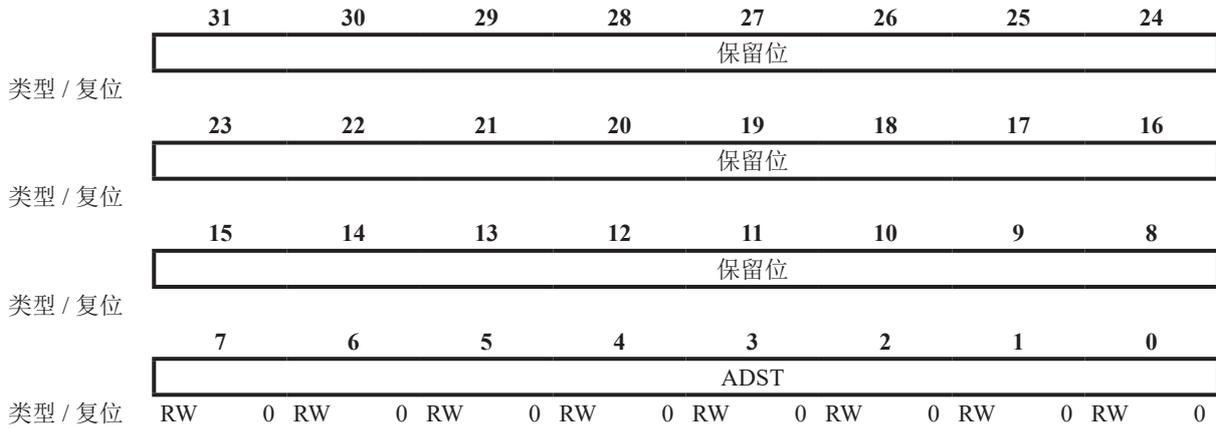
位	字段	描述
[28:24]	ADSEQ7	ADC 转换序列选择 7 定义 ADC 转换第七序列的 ADC 输入通道 0x00: ADC_IN0 0x01: ADC_IN1 0x02: ADC_IN2 0x03: ADC_IN3 0x04: ADC_IN4 0x05: ADC_IN5 0x06: ADC_IN6 0x07: ADC_IN7 0x08: ADC_IN8 0x09: ADC_IN9 0x0A: ADC_IN10 0x0B: ADC_IN11 0x0C: 模拟地, VSSA (V _{REF-}) 0x0D: 模拟电源, VDDA (V _{REF+}) 0x0E ~ 0x1F: 设置无效。这些值不能被选择, 因为它们可能导致 ADC 异常操作。
[20:16]	ADSEQ6	ADC 转换序列选择 6
[12:8]	ADSEQ5	ADC 转换序列选择 5
[4:0]	ADSEQ4	ADC 转换序列选择 4

ADC 输入采样时间寄存器 – ADCSTR

该寄存器定义了 A/D 转换通道的采样时间。

偏移量: 0x020

复位值: 0x0000_0000



位	字段	描述
[7:0]	ADST	ADC 输入通道采样时间 采样时间 = (ADST[7:0] + 1.5) × CK_ADC 时钟周期。

ADC 转换数据寄存器 y – ADCDRy, y = 0 ~ 7

该寄存器存储了 ADCLSTn (n = 0 ~ 1) 寄存器中的 ADSEQy 字段的第 y 转换序列的转换数据。

偏移量: 0x030 ~ 0x04C

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24						
	ADVLDy		保留位											
类型 / 复位	RC	0												
	23	22	21	20	19	18	17	16						
	保留位													
类型 / 复位														
	15	14	13	12	11	10	9	8						
	ADDy													
类型 / 复位	RO	0	RO	0	RO	0	RO	0	RO	0	RO	0	RO	0
	7	6	5	4	3	2	1	0						
	ADDy													
类型 / 复位	RO	0	RO	0	RO	0	RO	0	RO	0	RO	0	RO	0

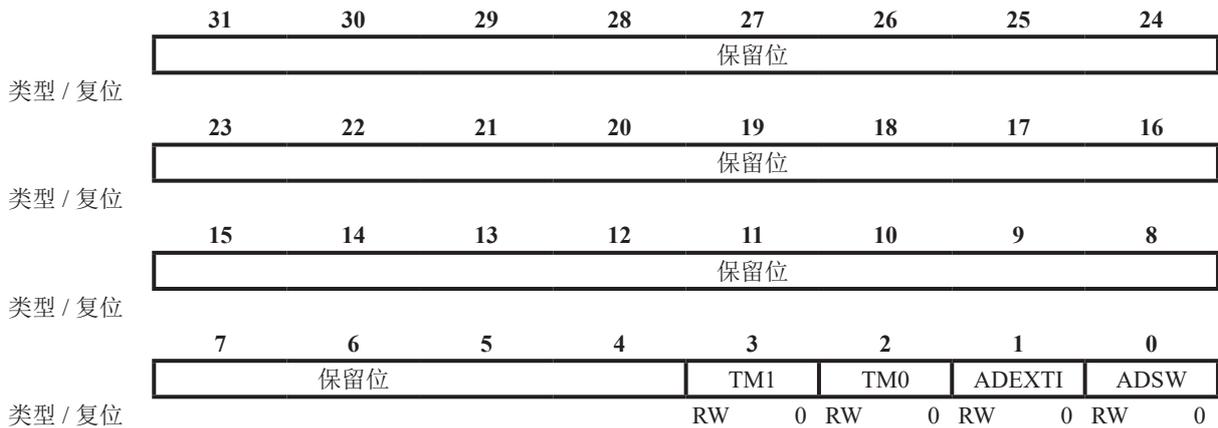
位	字段	描述
[31]	ADVLDy	ADC 序列 y 转换数据有效位 (y = 0 ~ 7) 0: 数据无效或已经被读出 1: 新数据有效
[15:0]	ADDy	ADC 序列 y 转换数据 (y = 0 ~ 7) ADCLSTn (n = 0 ~ 1) 寄存器中 ADSEQy 字段的序列转换结果。

ADC 触发控制寄存器 – ADCTCR

该寄存器包含了 ADC 转换启动转换的触发使能位。

偏移量: 0x070

复位值: 0x0000_0000



位	字段	描述
[3]	TM1	BFTM 或 PWM 事件触发的 ADC 转换使能控制 0: 除能 BFTM 或 PWM 事件触发转换 1: 使能 BFTM 或 PWM 事件触发转换
[2]	TM0	GPTM 事件触发的 ADC 转换使能控制 0: 除能 GPTM 事件触发转换 1: 使能 GPTM 事件触发转换
[1]	ADEXTI	EXTI 事件触发的 ADC 转换使能控制 0: 除能 EXTI 事件触发转换 1: 使能 EXTI 事件触发转换
[0]	ADSW	软件触发的 ADC 转换使能控制 0: 除能软件触发转换 1: 使能软件触发转换

ADC 触发源寄存器 – ADCTSR

该寄存器包含了触发源的选择和转换的软件触发位。

偏移量: 0x074
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24
	保留位		TM1E				TM0E	
类型 / 复位			RW	0	RW	0	RW	0
	23	22	21	20	19	18	17	16
	TM1S [2:1]		保留位		TM1S [0]	TM0S		
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	15	14	13	12	11	10	9	8
	保留位				ADEXTIS			
类型 / 复位					RW	0	RW	0
	7	6	5	4	3	2	1	0
	保留位							ADSC
类型 / 复位								RW

位	字段	描述
[29:27]	TM1E	ADC 转换的 PWM 触发事件的选择 000: PWM MTO 事件 001: PWM CH0 事件 010: PWM CH1 事件 011: PWM CH2 事件 100: PWM CH3 事件 其它: 保留 – 不使用, 否则, 转换结果将是不可预知的
[26:24]	TM0E	ADC 转换的 GPTM 触发事件的选择 000: GPTM MTO 事件 001: GPTM CH0 事件 010: GPTM CH1 事件 011: GPTM CH2 事件 100: GPTM CH3 事件 其它: 保留 – 不使用, 否则, 转换结果将是不可预知的
[23:22], [19]	TM1S	ADC 转换的 BFTM 或 PWM 触发定时器的选择 000: BFTM 001: 保留 010: PWM0 011: PWM1 其它: 保留 – 不使用, 否则, 转换结果将是不可预知的
[18:16]	TM0S	ADC 转换的 GPTM 触发定时器的选择 010: GPTM 其它: 保留 – 不使用, 否则, 转换结果将是不可预知的
[11:8]	ADEXTIS	ADC 转换的 EXTI 触发源的选择 0x00: EXTI 输入线 0 0x01: EXTI 输入线 1 ... 1111: EXTI 输入线 15 注意, EXTI 输入有效边沿是否触发 A/D 转换是由外部中断 / 事件控制单元 EXTI 决定的。

位	字段	描述
[0]	ADSC	ADC 转换的软件触发位 0: 无操作 1: 立即启动转换 通过软件设置手动启动转换。转换启动后由硬件清零。

ADC 看门狗控制寄存器 – ADCWCR

该寄存器定义了 ADC 看门狗功能的控制位和状态。

偏移量: 0x078
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24				
	保留位				ADUCH							
类型 / 复位					RO	0	RO	0	RO	0	RO	0
	23	22	21	20	19	18	17	16				
	保留位				ADLCH							
类型 / 复位					RO	0	RO	0	RO	0	RO	0
	15	14	13	12	11	10	9	8				
	保留位				ADWCH							
类型 / 复位					RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0				
	保留位				ADWALL	ADWUE	ADWLE					
类型 / 复位					RW	0	RW	0	RW	0		

位	字段	描述
[27:24]	ADUCH	上限阈值通道状态 0000: ADC_IN0 高于上限阈值 0001: ADC_IN1 高于上限阈值 ... 1011: ADC_IN11 高于上限阈值 其它: 保留 如果 ADWUE 及 ADWALL 状态位都被软件设为 1, 则 ADUCH 状态字段值应首先被存储在相关 ISR 的用户自定义存储器地址。否则, 如果另外一个输入通道转换的数据高于上限阈值时, ADUCH 字段将会改变。
[19:16]	ADLCH	下限阈值通道状态 0000: ADC_IN0 低于下限阈值 0001: ADC_IN1 低于下限阈值 ... 1011: ADC_IN11 低于下限阈值 其它: 保留 如果 ADWLE 及 ADWALL 状态位都被软件设为 1, 则 ADLCH 状态字段值应首先被存储在相关 ISR 的用户自定义存储器地址。否则, 如果另外一个输入通道转换的数据低于上限阈值时, ADLCH 字段将会改变。

位	字段	描述
[11:8]	ADWCH	ADC 看门狗指定通道选择 0000: ADC_IN0 被监控 0001: ADC_IN1 被监控 ... 1011: ADC_IN11 被监控 其它: 保留
[2]	ADWALL	ADC 看门狗功能指定的通道或所有通道设置 0: 仅 ADWCH 字段指定的通道被监控 1: 所有通道被监控
[1]	ADWUE	ADC 看门狗上限阈值使能位 0: 除能上限阈值功能 1: 使能上限阈值功能
[0]	ADWLE	ADC 看门狗下限阈值使能位 0: 除能下限阈值功能 1: 使能下限阈值功能

ADC 看门狗阈值寄存器 – ADCTR

该寄存器定义了 ADC 看门狗功能的上限和下限阈值。

偏移量: 0x07C
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24				
	保留位				ADUT							
类型 / 复位					RW	0	RW	0	RW	0	RW	0
	23	22	21	20	19	18	17	16				
	ADUT											
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0
	15	14	13	12	11	10	9	8				
	保留位				ADLT							
类型 / 复位					RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0				
	ADLT											
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0

位	字段	描述
[27:16]	ADUT	ADC 看门狗的上限阈值 指定 ADC 看门狗监测功能的上限阈值。
[11:0]	ADLT	ADC 看门狗的下限阈值 指定 ADC 看门狗监测功能的下限阈值。

ADC 中断使能寄存器 – ADCIER

该寄存器包含了 ADC 的中断使能位。

偏移量: 0x080
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24			
类型 / 复位	保留位							ADIEO	0		
	23	22	21	20	19	18	17	16			
类型 / 复位	保留位						ADIEU	0	ADIEL	0	
	15	14	13	12	11	10	9	8			
类型 / 复位	保留位										
	7	6	5	4	3	2	1	0			
类型 / 复位	保留位					ADIEC	0	ADIEG	0	ADIES	0

位	字段	描述
[24]	ADIEO	ADC 数据寄存器覆盖中断使能位 0: ADC 数据寄存器覆盖中断被除能 1: ADC 数据寄存器覆盖中断被使能
[17]	ADIEU	ADC 看门狗上限阈值中断使能位 0: ADC 看门狗上限阈值中断被除能 1: ADC 看门狗上限阈值中断被使能
[16]	ADIEL	ADC 看门狗下限阈值中断使能位 0: ADC 看门狗下限阈值中断被除能 1: ADC 看门狗下限阈值中断被使能
[2]	ADIEC	ADC 周期 EOC 中断使能位 0: ADC 周期转换结束中断被除能 1: ADC 周期转换结束中断被使能
[1]	ADIEG	ADC 子组 EOC 中断使能位 0: ADC 子组转换结束中断被除能 1: ADC 子组转换结束中断被使能
[0]	ADIES	ADC 单次 EOC 中断使能位 0: ADC 单次转换结束中断被除能 1: ADC 单次转换结束中断被使能

ADC 中断原始状态寄存器 – ADCIRAW

该寄存器包含了 ADC 的中断原始状态位。

偏移量: 0x084

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24			
	保留位							ADIRAWO			
类型 / 复位								RO	0		
	23	22	21	20	19	18	17	16			
	保留位						ADIRAWU	ADIRAWL			
类型 / 复位							RO	0	RO	0	
	15	14	13	12	11	10	9	8			
	保留位										
类型 / 复位											
	7	6	5	4	3	2	1	0			
	保留位					ADIRAWC	ADIRAWG	ADIRAWS			
类型 / 复位						RO	0	RO	0	RO	0

位	字段	描述
[24]	ADIRAWO	ADC 数据寄存器覆盖中断原始状态位 0: 未发生 ADC 数据寄存器覆盖中断 1: 发生 ADC 数据寄存器覆盖中断
[17]	ADIRAWU	ADC 看门狗上限阈值中断原始状态位 0: 未发生 ADC 看门狗上限阈值中断 1: 发生 ADC 看门狗上限阈值中断
[16]	ADIRAWL	ADC 看门狗下限阈值中断原始状态位 0: 未发生 ADC 看门狗下限阈值中断 1: 发生 ADC 看门狗下限阈值中断
[2]	ADIRAWC	ADC 周期 EOC 中断原始状态位 0: 未发生 ADC 周期转换结束中断 1: 发生 ADC 周期转换结束中断
[1]	ADIRAWG	ADC 子组 EOC 中断原始状态位 0: 未发生 ADC 子组转换结束中断 1: 发生 ADC 子组转换结束中断
[0]	ADIRAWS	ADC 单次 EOC 中断原始状态位 0: 未发生 ADC 单次转换结束中断 1: 发生 ADC 单次转换结束中断

ADC 中断状态寄存器 – ADCISR

该寄存器包含了 ADC 的中断状态位。如果相关的中断事件发生且相关的使能位置 1，相应的状态位将被置 1。

偏移量： 0x088

复位值： 0x0000_0000

	31	30	29	28	27	26	25	24
	保留位							ADISRO
类型 / 复位								RO 0
	23	22	21	20	19	18	17	16
	保留位						ADISRU	ADISRL
类型 / 复位							RO 0	RO 0
	15	14	13	12	11	10	9	8
	保留位							
类型 / 复位								
	7	6	5	4	3	2	1	0
	保留位					ADISRC	ADISRG	ADISRS
类型 / 复位						RO 0	RO 0	RO 0

位	字段	描述
[24]	ADISRO	ADC 数据寄存器覆盖中断状态位 0: 未发生 ADC 数据寄存器覆盖中断或相关中断控制除能 1: 发生 ADC 数据寄存器覆盖中断且相关中断控制使能
[17]	ADISRU	ADC 看门狗上限阈值中断状态位 0: 未发生 ADC 看门狗上限阈值中断或相关中断控制除能 1: 发生 ADC 看门狗上限阈值中断且相关中断控制使能
[16]	ADISRL	ADC 看门狗下限阈值中断状态位 0: 未发生 ADC 看门狗下限阈值中断或相关中断控制除能 1: 发生 ADC 看门狗下限阈值中断且相关中断控制使能
[2]	ADISRC	ADC 周期 EOC 中断状态位 0: 未发生 ADC 周期转换结束中断或相关中断控制除能 1: 发生 ADC 周期转换结束中断且相关中断控制使能
[1]	ADISRG	ADC 的子组 EOC 中断状态位 0: 未发生 ADC 子组转换结束中断或相关中断控制除能 1: 发生 ADC 子组转换结束中断且相关中断控制使能
[0]	ADISRS	ADC 的单个 EOC 中断状态位 0: 未发生 ADC 的单个转换结束中断或相关中断控制除能 1: 发生 ADC 的单个转换结束中断且相关中断控制使能

ADC 中断清除寄存器 – ADCICLR

该寄存器用来清除 ADC 的中断原始位和中断状态位。这些位由软件置位来清除中断状态位，被置 1 后，由硬件自动清零。

偏移量: 0x08C
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位							ADICLRO	0
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位						ADICLRU	ADICLRL	0
	15	14	13	12	11	10	9	8	
类型 / 复位	保留位								
	7	6	5	4	3	2	1	0	
类型 / 复位	保留位					ADICLRC	ADICLRG	ADICLRS	0

位	字段	描述
[24]	ADICLRO	ADC 数据寄存器覆盖中断状态清除位 0: 不影响 1: 清除 ADISRO 和 ADIRAWO 位
[17]	ADICLRU	ADC 看门狗上限阈值中断状态清除位 0: 不影响 1: 清除 ADISRU 和 ADIRAWU 位
[16]	ADICLRL	ADC 看门狗下限阈值中断状态清除位 0: 不影响 1: 清除 ADISRL 和 ADIRAWL 位
[2]	ADICLRC	ADC 周期 EOC 中断状态清除位 0: 不影响 1: 清除 ADISRC 和 ADIRAWC 位
[1]	ADICLRG	ADC 子组 EOC 中断状态清除位 0: 不影响 1: 清除 ADISRG 和 ADIRAWG 位
[0]	ADICLRS	ADC 单次 EOC 中断状态清除位 0: 不影响 1: 清除 ADISRS 和 ADIRAWS 位

13 通用定时器 (GPTM)

简介

通用定时器由一个 16-bit 向上 / 向下计数器、四个 16-bit 捕捉 / 比较寄存器 (CCR)、一个 16-bit 计数器重载寄存器 (CRR) 和几个控制 / 状态寄存器组成。GPTM 可用于多种用途, 包括通用计时、测量输入信号脉冲宽度或产生输出波形, 如单脉冲或 PWM 输出。GPTM 还内建可处理编码器接口信号的带两个输入口的正交解码器。

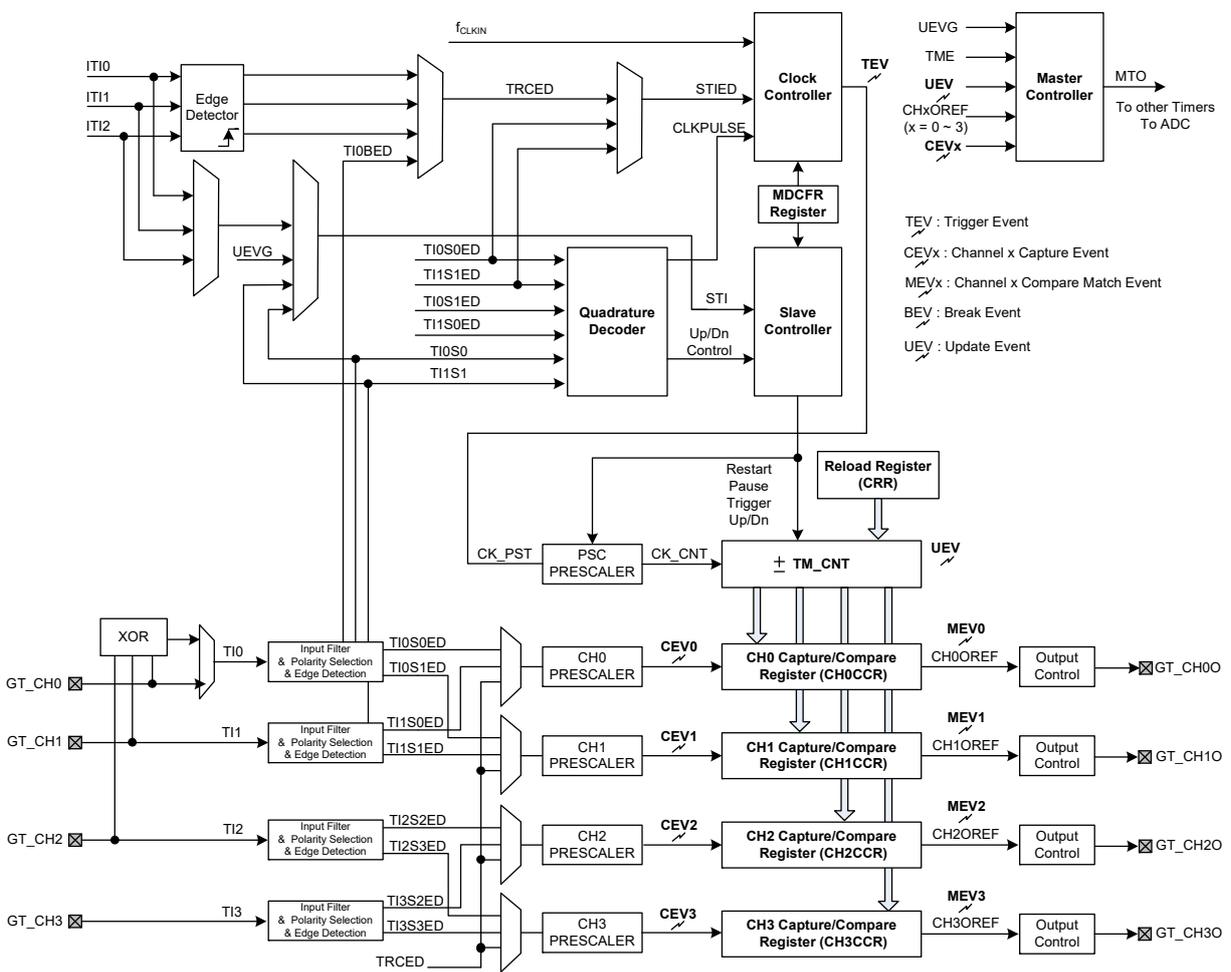


图 30. GPTM 方框图

特性

- 16-bit 向上 / 向下自动重载计数器
- 16-bit 可编程预分频器，可以对计数器时钟进行 1 ~ 65536 之间的任意数值的分频
- 多达 4 个独立通道用于：
 - 输入捕捉功能
 - 比较匹配输出
 - PWM 波形生成 – 边沿对齐计数和中心对齐计数模式
 - 单脉冲模式输出
- 内建可处理编码器接口信号的带两个输入口的正交解码器
- 同步电路可实现通过外部信号控制定时器和定时器间的互联
- 下列事件发生时将产生中断请求：
 - 更新事件
 - 触发事件
 - 输入捕捉事件
 - 比较匹配输出事件
- GPTM 主机 / 从机模式控制器

功能描述

计数器模式

向上计数

在向上计数模式里，计数器从 0 开始连续向上计数，一直计数到 CRR 寄存器定义的计数器的重载值。一旦计数器的值达到了计数器重载值，定时器模块将产生溢出事件，并从 0 开始重新计数。这一动作会反复执行。CNTCFR 寄存器中的计数方向位 DIR 设置为 0 可选择向上计数模式。

当通过设置 EVGR 寄存器中的 UEVG 位为 1 触发更新事件时，计数器的值将被初始化为 0。

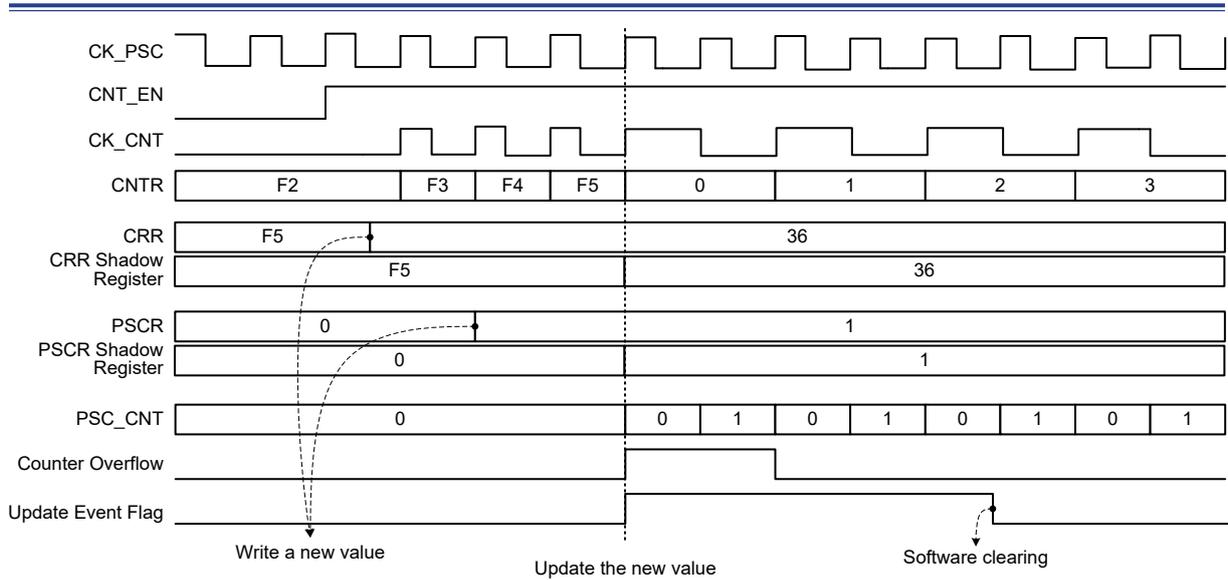


图 31. 向上计数范例

向下计数

在向下计数模式里，计数器从 CRR 寄存器定义的计数器重载值开始向下计数，一直计数到 0。一旦计数器的值达到 0，定时器模块将产生一个下溢事件，并从计数器重载值开始重新计数。这一动作会反复执行。CNTCFR 寄存器中的计数方向位 DIR 设置为 1 可选择向下计数模式。

当通过设置 EVGR 寄存器中的 UEVG 位为 1 触发更新事件时，计数器的值将被初始化为计数器重载值。

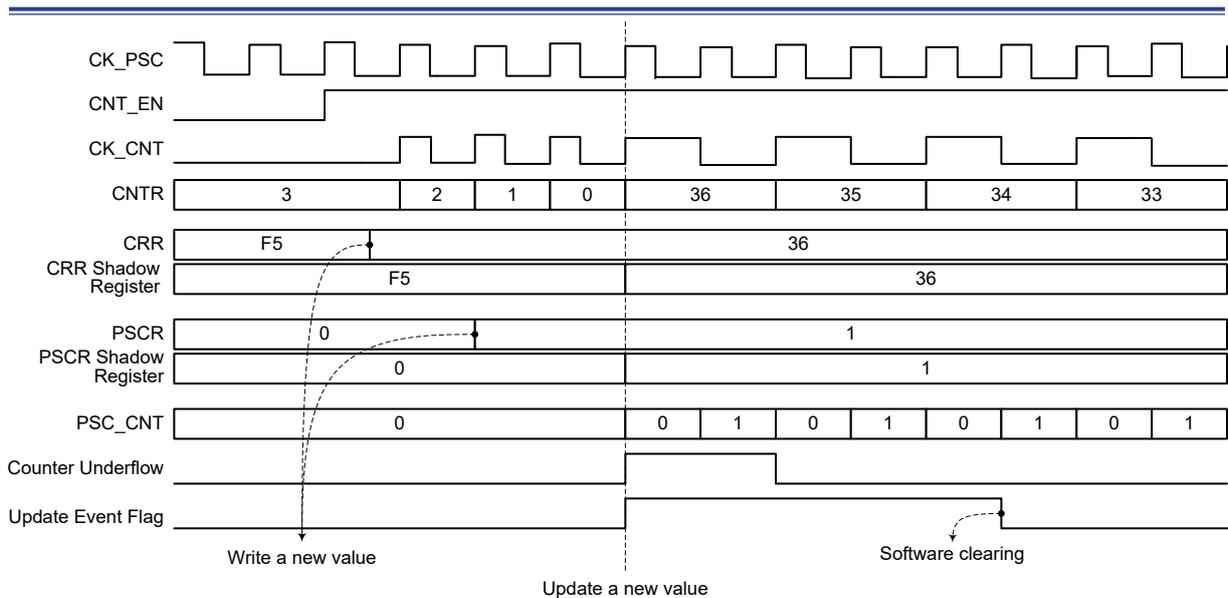


图 32. 向下计数范例

中心对齐计数

在中心对齐计数模式里，计数器先从 0 向上计数到重载值然后向下计数到 0，如此交替。在向上计数模式下，当计数器计数到计数器重载值时，计数器模块会产生一个上溢事件；而在向下计数模式下，当计数器计数到 0 时，计数器模块会产生一个下溢事件。CNTCFR 寄存器中的 DIR 位是只读位，其值表明了中心对齐计数模式下的计数方向。计数方向由硬件自动更新。

在中心对齐计数模式下，如果将 EVGR 寄存器中的 UEVG 位置位，那么不管计数器是正在向上计数还是向下计数，计数器的值都将初始化为 0。

可经由 CNTCFR 寄存器中的 CMSEL 字段来控制当发生上溢事件 (CMSEL = 0x1)、下溢事件 (CMSEL = 0x2) 或两个事件都发生 (CMSEL = 0x3) 时，INTSR 寄存器中的 UEVIF 位被置为 1。

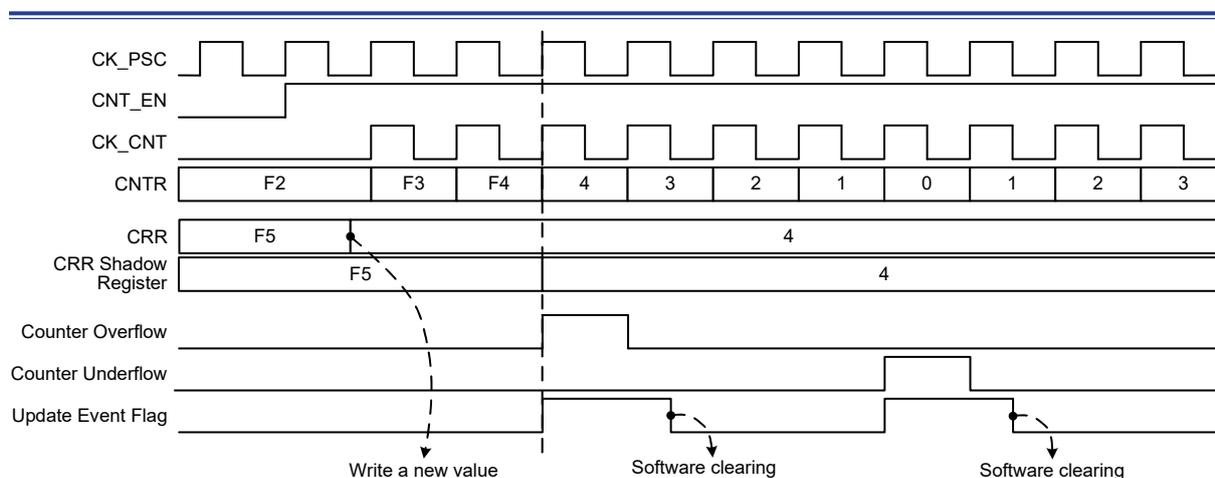


图 33. 中心对齐计数范例

时钟控制器

以下描述了定时器模块的时钟控制器，它用来选择内部预分频计数器的时钟源。

■ 内部 APB 时钟 f_{CLKIN} :

当从机模式除能时，默认内部时钟源是 APB 时钟 f_{CLKIN} ，用来驱动计数器预分频器。当从机模式选择位 $SMSEL$ 被置为 $0x4$ 、 $0x5$ 或 $0x6$ 时，内部 APB 时钟 f_{CLKIN} 将作为驱动计数器预分频器的时钟源。如果设置 $MDCFR$ 寄存器中的 $SMSEL$ 字段为 $0x1$ 、 $0x2$ 、 $0x3$ 或 $0x7$ 来使能从机模式控制器，那么预分频器时钟将由 $TRCFR$ 寄存器中的 $TRSEL$ 字段设置为其它时钟源，具体描述如下。

■ 正交解码器:

要选择正交解码器模式，应把 $MDCFR$ 寄存器中的 $SMSEL$ 字段设为 $0x1$ 、 $0x2$ 或 $0x3$ 。正交解码器功能通过 GT_CH0 和 GT_CH1 引脚的输入状态来产生时钟脉冲以驱动计数器预分频器。在每一个输入源信号发生转换时，计数器方向位 DIR 的值会由硬件自动依当前状态设置。输入源信号可以只来自于 GT_CH0 引脚或 GT_CH1 引脚，也可以来自于两个引脚。

■ STIED:

计数器预分频器在每一个 STI 信号的上升沿计数。此模式可通过将 $MDCFR$ 寄存器中的 $SMSEL$ 字段设为 $0x7$ 来选择。这时计数器将作为一个事件计数器使用。输入事件，即 STI ，可通过把 $TRSEL$ 字段设成除 $0x0$ 以外的可用值来选择。当 STI 信号被选择作为时钟源使用时，在每一个 STI 信号上升沿时，内部边沿检测电路将会产生一个时钟脉冲来驱动计数器预分频器。值得注意的是，如果 $TRSEL$ 字段设成 $0x0$ 来选择软件 $UEVG$ 位作为触发源，那么当 $SMSEL$ 字段设成 $0x7$ 时，计数器将会被更新而非计数。

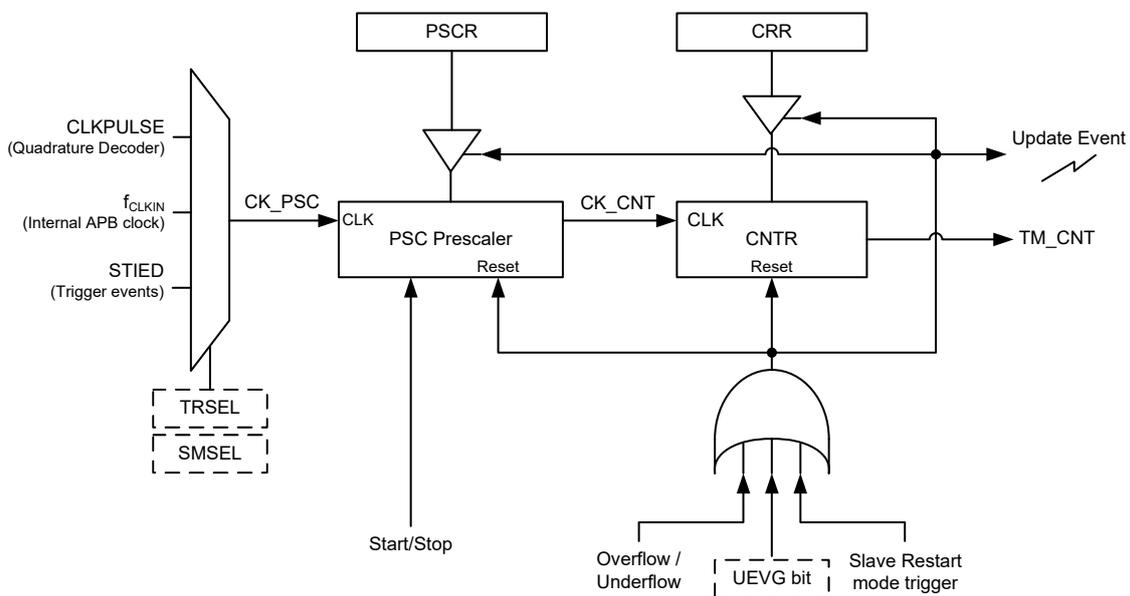


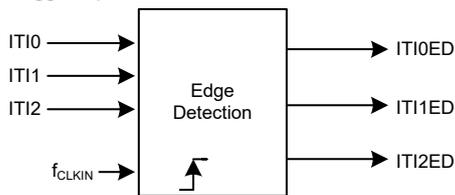
图 34. GPTM 时钟源选择

触发控制器

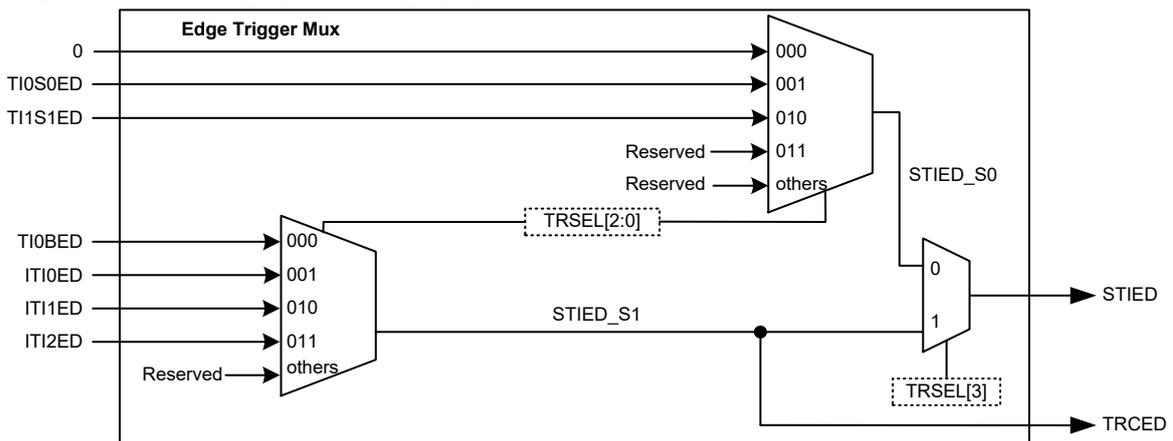
触发控制器用来选择触发源以及设置电平触发或边沿触发条件。内部触发输入可通过 TRCFR 寄存器中的触发选择位 TRSEL 进行选择。除了 UEVG 位软件触发之外的所有触发源，内部边沿检测电路将会在每个触发信号上升沿作用期间产生一个时钟脉冲，以激活某些因触发信号上升沿而触发的 GPTM 功能。

Trigger Controller Block = Edge Trigger Mux + Level Trigger Mux

Internal Trigger Input



Edge Trigger Source = Internal (ITix) + Channel input (TIn)



Level Trigger Source = Internal (ITix) + Channel input (TIn) + Software UEVG bit

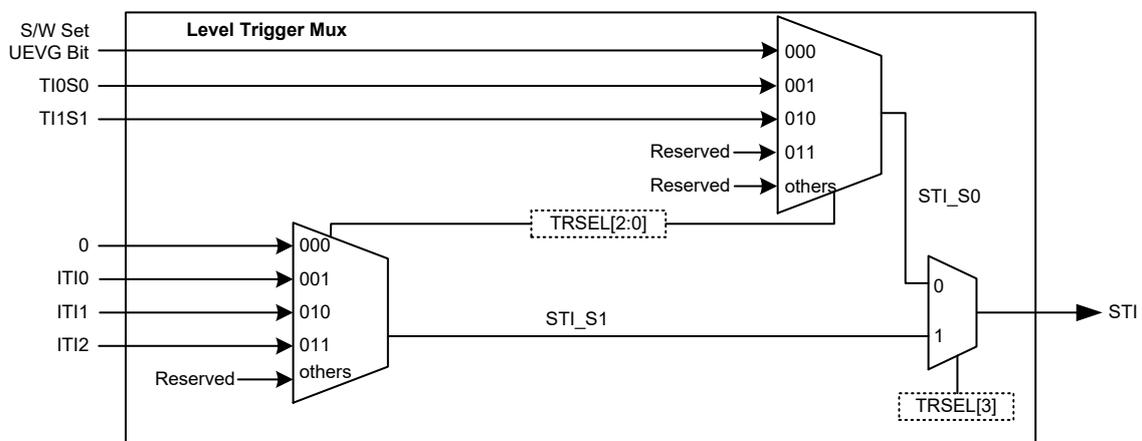


图 35. 触发控制器方框图

从机控制器

在几种模式下，GPTM 可以与一个外部触发器进行同步。这些模式包括重启模式、暂停模式和触发模式，是通过 MDCFR 寄存器中的 SMSEL 字段选择的。这些模式的触发输入来自于 STI 信号，通过 TRCFR 寄存器中的 TRSEL 字段选择。从机控制器中的工作模式在相关章节中有所描述。

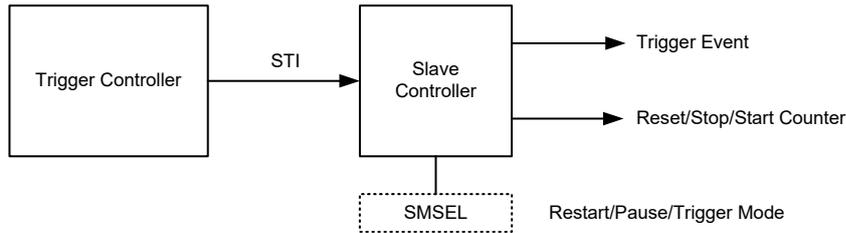


图 36. 从机控制器方框图

重启模式

作为对 STI 信号上升沿的响应，计数器及其预分频器会被重新初始化。当一个 STI 上升沿到来时，更新事件软件产生位 UEVG 将被硬件自动置位，触发事件标志位也将被置位，计数器和预分频器将被重新初始化。虽然 UEVG 位被硬件置 1，但是更新事件是否真正发生还要取决于 CNTCFR 寄存器中的更新事件除能控制位 UEVDIS 的设置。如果 UEVDIS 位被置 1 来除能更新事件，那么更新事件将不会发生，然而当 STI 上升沿到来时，计数器和预分频器仍会被重新初始化。如果 UEVDIS 位被清零来使能更新事件，则当 STI 上升沿到来时，发生更新事件，所有预加载的寄存器将被更新。

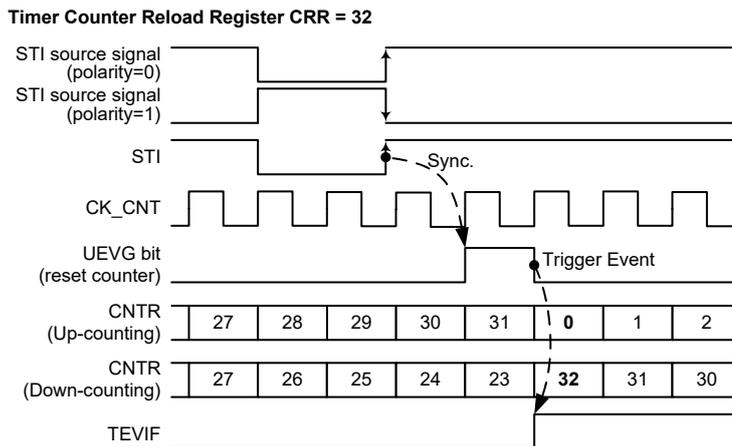


图 37. 重启模式下的 GPTM

暂停模式

在暂停模式下, 所选择的 STI 输入信号电平用来控制计数器的开始 / 停止操作。当 STI 信号处于高电平时, 计数器开始计数; 当 STI 信号转换为低电平时, 计数器停止计数并保持当前值不变, 不会被复位。由于计数器的开始 / 停止操作是由 STI 电平控制的, 则所选择的 STI 信号不能来自于 TI0BED。

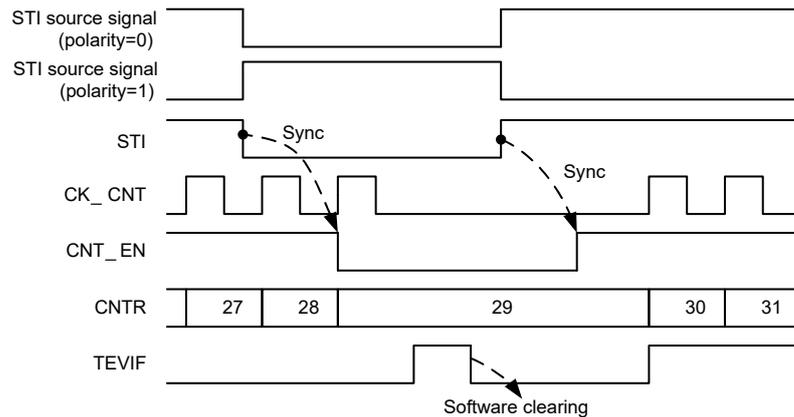


图 38. 暂停模式下的 GPTM

触发模式

在计数器停止计数后, 当一个 STI 上升沿信号到来时, 计数器将从当前值继续开始计数。注意, 如果 STI 信号选择来自于 UEVG 位软件触发, 计数器不会重新计数。当 STI 源信号通过 UEVG 位选择为软件触发时, 不会产生使计数器重新计数的时钟脉冲。还要注意, STI 信号只是用来使计数器开始继续计数, 而没有使计数器停止计数的作用。

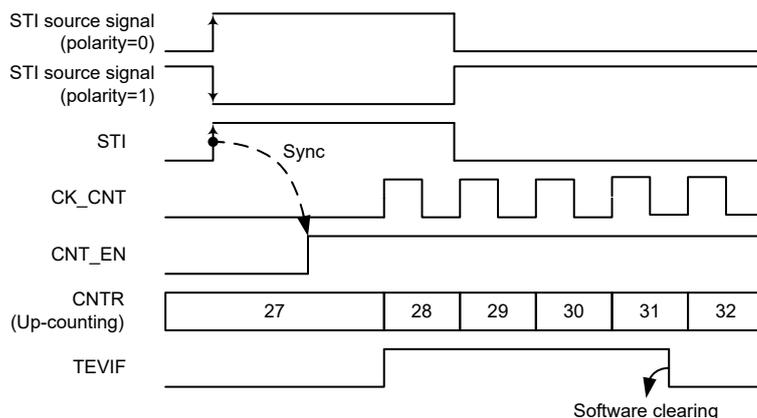


图 39. 触发模式下的 GPTM

主机控制器

GPTM 和 MCTM 可在内部连接在一起用于定时器同步或链接。当一个 GPTM 被配置在主机模式下时, GPTM 主机控制器将会产生一个主机触发输出 (MTO) 信号, 通过 MDCFR 寄存器中的 MMSEL 字段选择可以触发或驱动处于从机模式下的其它 GPTM 或 MCTM 定时器复位、启动、停止或为其提供时钟源。

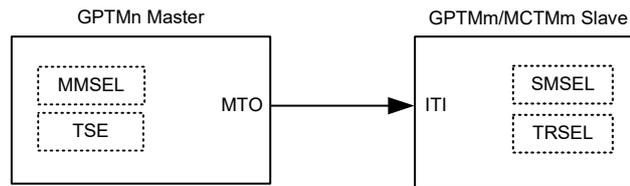


图 40. 主机 GPTMn 和从机 GPTMm / MCTMm 相连接

MDCFR 寄存器中的主机模式选择字段 (MMSEL) 用来选择同步另外一个从机 GPTM 或 MCTM 的 MTO 源。

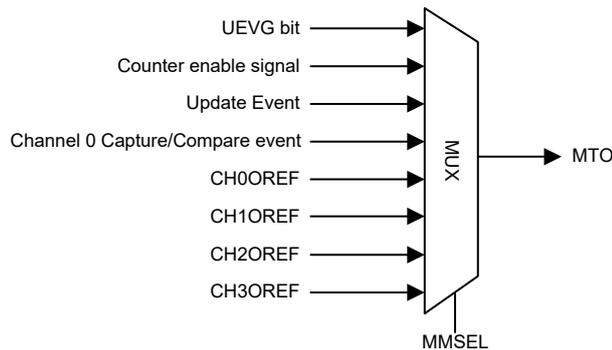


图 41. MTO 选项

例如, 把 MMSEL 字段设为 0x5, 以选择 CH1OREF 信号作为 MTO 信号来同步另外一个从机 GPTM 或 MCTM。欲知更多详细描述, 请参考相关的 MDCFR 寄存器中的 MMSEL 字段定义。

通道控制器

GPTM 有四个独立的通道，用来选择作为捕捉输入或比较匹配输出。每个捕捉输入或比较匹配输出通道都由一个预载寄存器和一个影子寄存器组成。APB 总线只能通过读 / 写预载寄存器来进行数据访问。

当工作在输入捕捉模式，捕捉事件发生时，计数器的值会首先被捕捉到 CHxCCR 影子寄存器中，其值再被传送到 CHxCCR 预载寄存器中。

当工作在比较匹配输出模式，CHxCCR 预载寄存器的内容会被复制到相应的影子寄存器中，然后计数器的值会与寄存器的值进行比较。

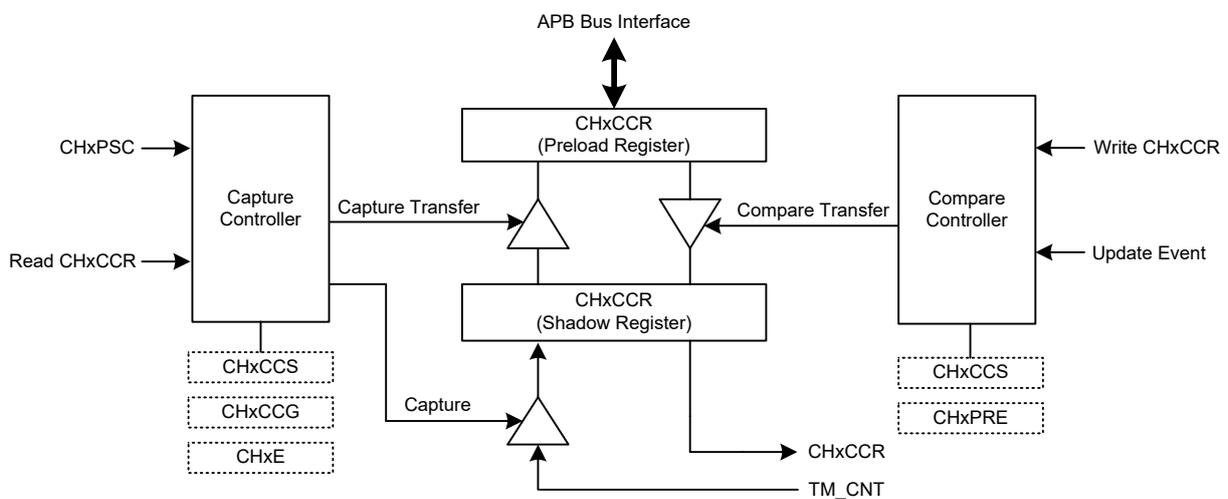


图 42. 捕捉 / 比较方框图

捕捉计数器值传送到 CHxCCR

当通道用作捕捉输入,有效输入信号跳变发生时,计数器的值会被捕捉到通道捕捉/比较寄存器(CHxCCR)中。一旦捕捉事件发生,INTSR寄存器中的CHxCCIF标志位会相应的被置位。如果CHxCCIF位已经被置位,即标志位还未被软件清零,而此通道的另外一个捕捉事件发生,则相应的通道过度捕捉标志位 CHxOCF 将被置位。

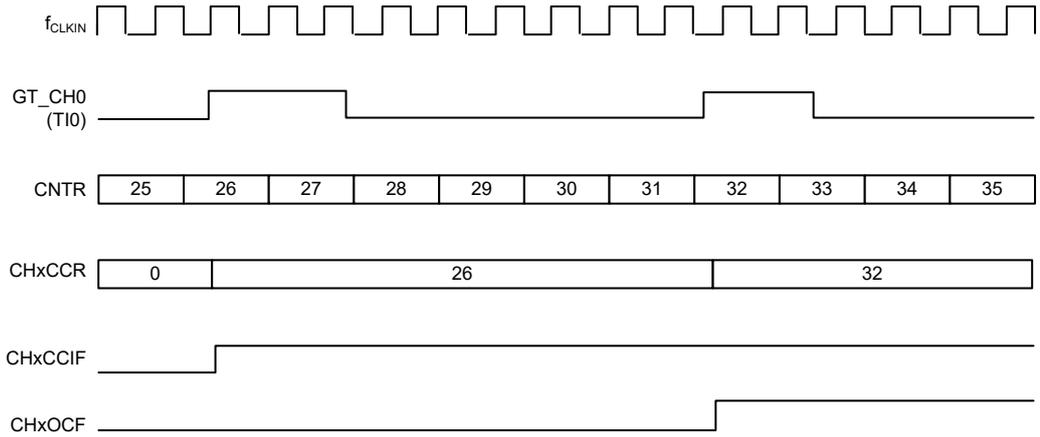


图 43. 输入捕捉模式

脉冲宽度测量

输入捕捉模式也可以用于 GT_CHx 引脚上的信号 (TIx) 脉冲宽度测量。下面的例子说明了如何配置 GPTM 使其工作于输入捕捉模式，使用通道 0 和通道 1 测量在 GT_CH0 引脚上的高脉冲宽度和输入周期。基本步骤如下：

- 配置捕捉通道 0 (CH0CCS = 0x1)，选择 TI0 信号作为捕捉输入
- 设置 CH0P 位为 0，选择 TI0 输入的上升沿作为有效极性
- 配置捕捉通道 1 (CH1CCS = 0x2)，选择 TI0 信号作为捕捉输入
- 设置 CH1P 位为 1，选择 TI0 输入的下沿作为有效极性
- 设置 TRSEL 位为 0x1，选择 TI0S0 作为触发输入
- 把 MDCFR 寄存器的 SMSEL 字段置为 0x4，使从机控制器工作在重启模式下
- 把 CHCTR 寄存器中的 CH0E 和 CH1E 位置为 1 来开启输入捕捉模式

如下图所示，输入捕捉操作之后，GT_CH0 引脚上的高脉冲宽度将被捕捉到 CH1CCR 寄存器中，输入信号将被捕捉到 CH0CCR 寄存器中。

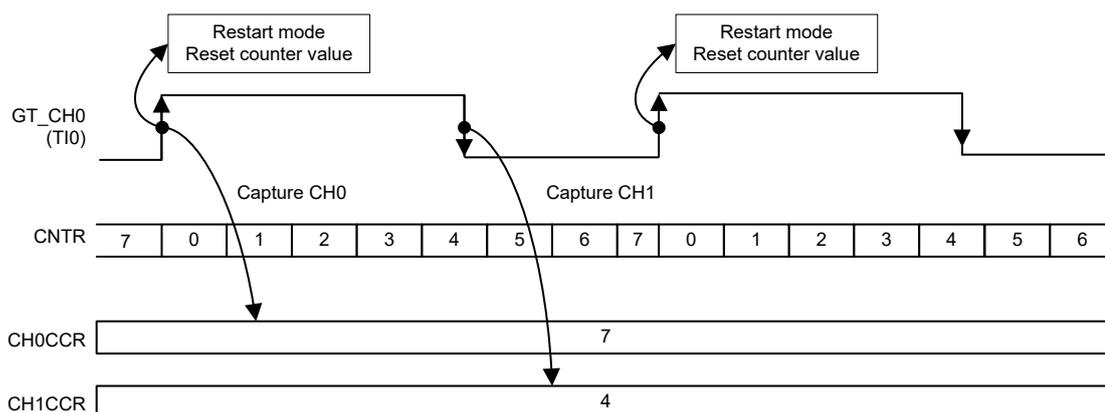


图 44. PWM 脉冲宽度测量范例

输入级

输入级由一个数字滤波器、一个通道极性选择器、边沿检测电路和一个通道预分频器组成。通道 0 输入信号 TI0 可以来自于 GT_CH0 信号或 GT_CH0、GT_CH1 和 GT_CH2 信号的异或。通道输入信号 TIx 被一个数字滤波器采样，产生一个滤波输入信号 TIxFP。通道极性和边沿检测模块可以产生一个 TIxS0ED 或 TIxS1ED 信号提供给输入捕捉功能。有效输入事件数量可通过通道输入预分频寄存器 CHxPSC 设置。

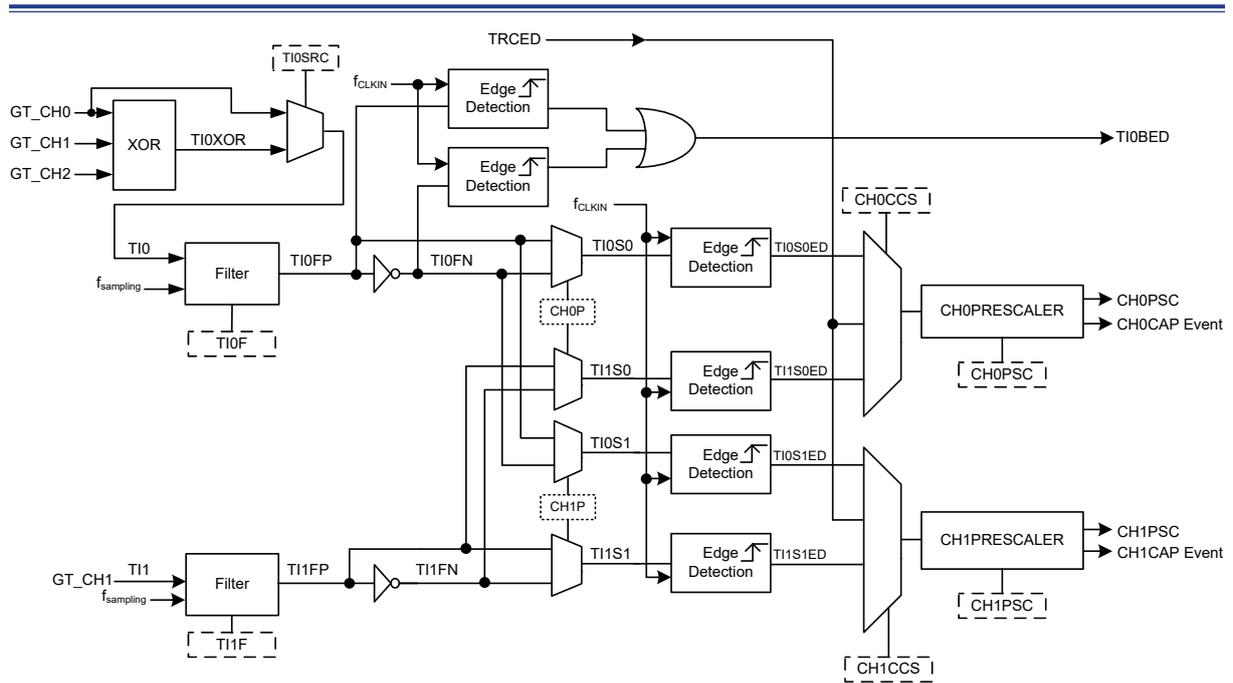


图 45. 通道 0 和通道 1 输入级

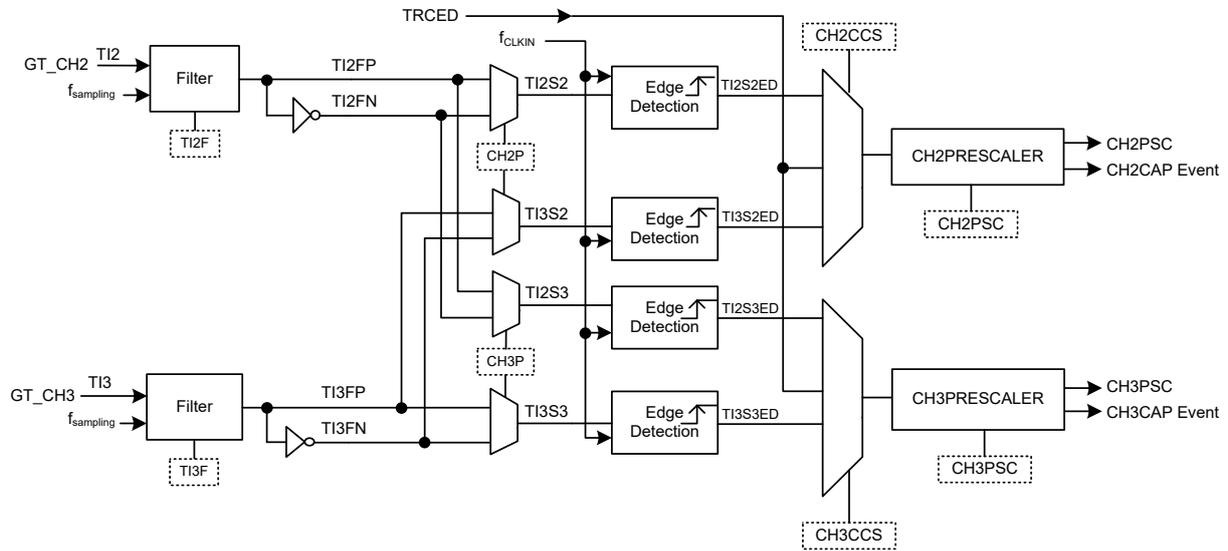


图 46. 通道 2 和通道 3 输入级

数字滤波器

数字滤波器嵌入在输入级，分别用于 GT_CH0 ~ GT_CH3 引脚。GPTM 中的数字滤波器是一个 N 事件计数器，N 指的是它记录到 N 个有效事件后会产生一个输出的跳变。根据每个滤波器的用户选择，N 的值可以是 0、2、4、5、6 或 8。

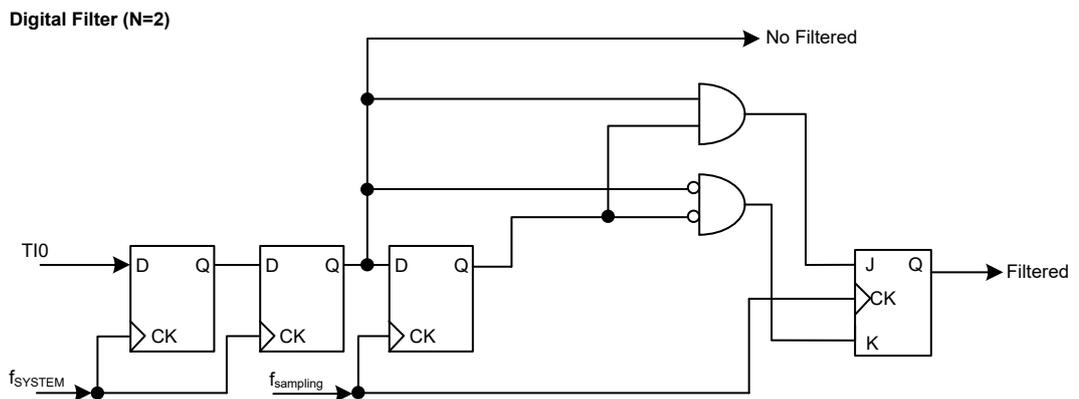


图 47. TI0 数字滤波器方框图 (N = 2)

正交解码器

正交解码器功能使用两个分别来自于 GT_CH0 和 GT_CH1 引脚的正交输入 TI0 和 TI1 进行交互以生成计数器的值。在每个输入源信号转换时，DIR 位会被硬件依状态自动更改。输入源可以是只来自于 TI0 或只来自于 TI1，也可来自于 TI0 和 TI1，通过设置 SMSEL 字段为 0x1、0x2 或 0x3 来选择模式。下表列出了改变计数器方向的几种情况。正交解码器可视为一个带有方向性选择的外部时钟。这意味着，计数器将在 0 和计数器重载值之间连续计数。因此，用户必须在计数器开始计数之前设置 CRR 寄存器。

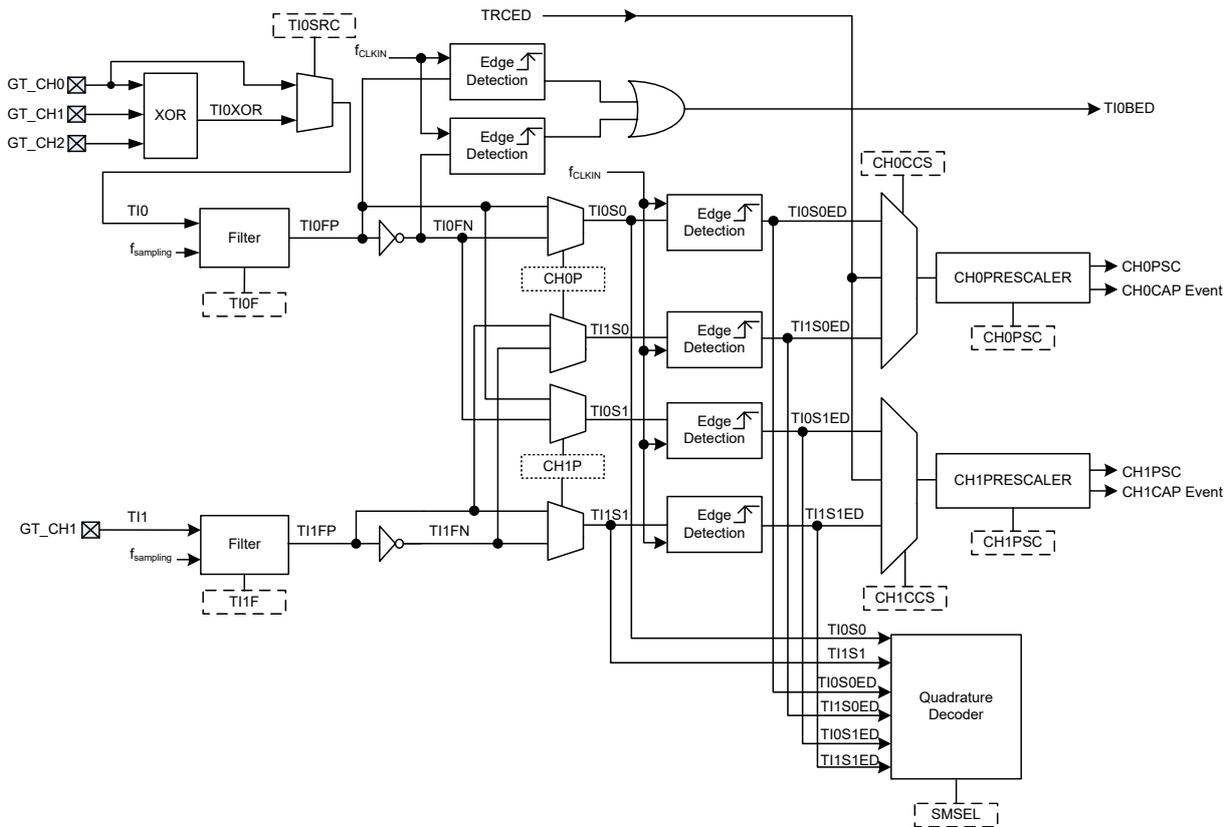


图 48. 输入级和正交解码器方框图

表 26. 计数方向和编码信号

计数模式	电平	TI0S0		TI1S1	
		上升沿	下降沿	上升沿	下降沿
仅 TI0 计数 (SMSEL = 0x1)	TI1S1 = 高	向下	向上	—	—
	TI1S1 = 低	向上	向下	—	—
仅 TI1 计数 (SMSEL = 0x2)	TI0S0 = 高	—	—	向上	向下
	TI0S0 = 低	—	—	向下	向上
TI0 和 TI1 计数 (SMSEL = 0x3)	TI1S1 = 高	向下	向上	X	X
	TI1S1 = 低	向上	向下	X	X
	TI0S0 = 高	X	X	向上	向下
	TI0S0 = 低	X	X	向下	向上

注：“—” → 没有计数；“X” → 不可能

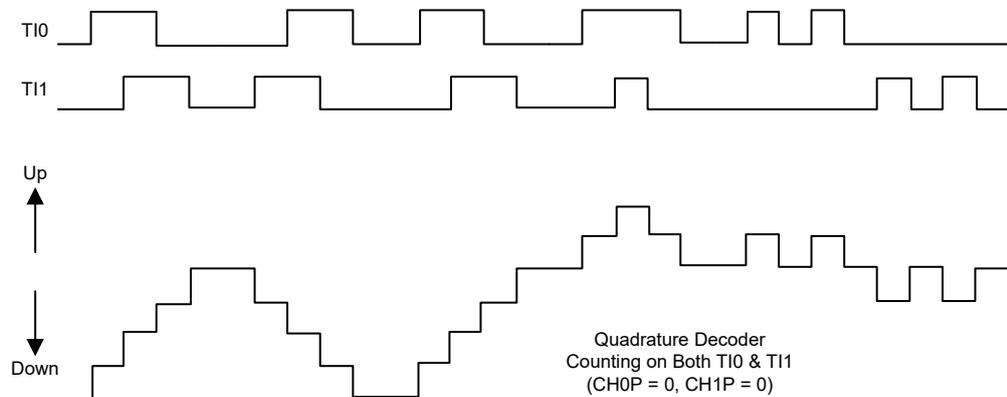


图 49. TI0 和 TI1 正交解码器计数

输出级

GPTM 有四个通道, 用于比较匹配、单脉冲或 PWM 输出功能。通道输出 GT_CHxO 由 CHxO CFR、CHPOLR 和 CHCTR 寄存器分别对应的 CHxOM、CHxP 和 CHxE 位控制。

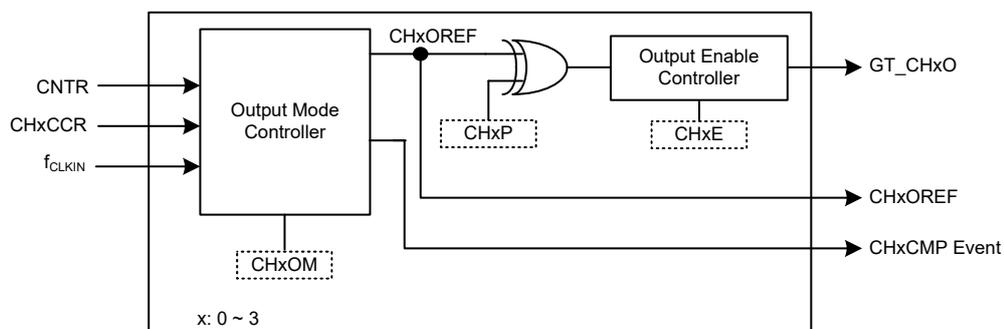


图 50. 输出级方框图

通道输出参考信号

当 GPTM 用在比较匹配输出模式时, CHxOREF 信号 (通道 x 输出参考信号) 通过设置 CHxOM 字段来定义。CHxOREF 信号有多种输出功能类型, 即当计数器的值与 CHxCCR 寄存器的内容匹配时, CHxOREF 输出可为低电平, 高电平或者翻转, 除此之外, 也有 PWM 模式 1 和 PWM 模式 2 输出。在这些模式中, CHxOREF 信号的电平都是根据计数方向以及计数器值与 CHxCCR 内容的关系而改变。还有两种模式, 不论计数器和 CHxCCR 的值是什么, 输出都会被强制为一个无效或有效的电平。更多详细说明请参考相应位的定义。输出类型设置如表 27 所示。

表 27. 比较匹配输出设置

CHxOM 值	比较匹配输出电平
0x00	无变化
0x01	输出 0
0x02	输出 1
0x03	输出翻转
0x04	强制无效电平
0x05	强制有效电平
0x06	PWM 模式 1
0x07	PWM 模式 2

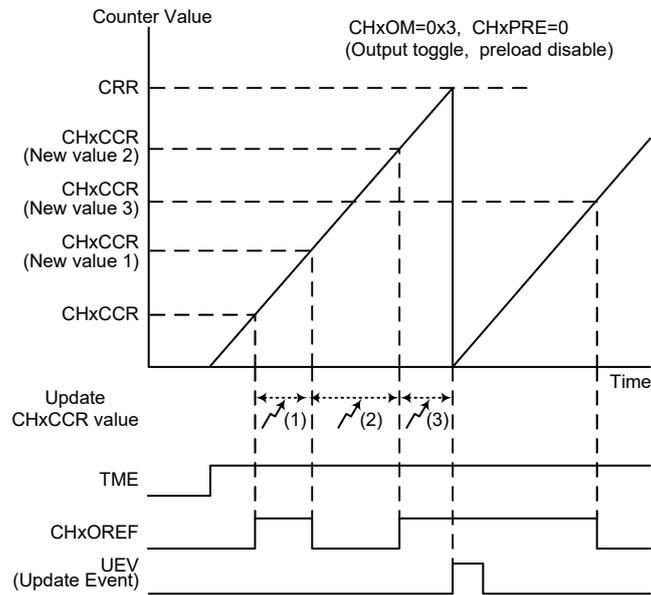


图 51. 翻转模式通道输出参考信号 - CHxPRE = 0

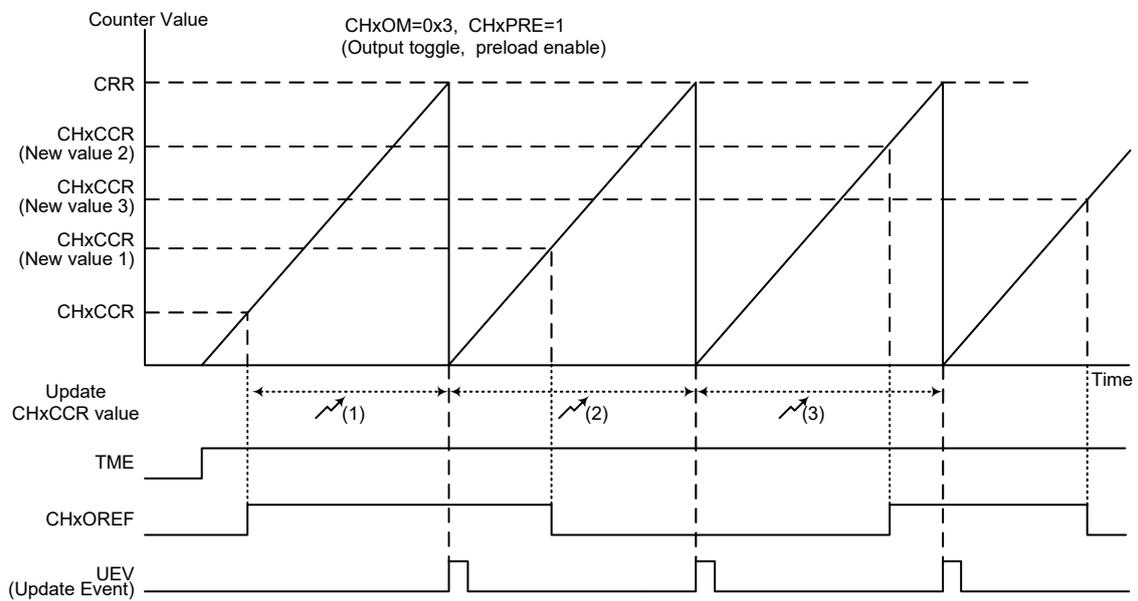


图 52. 翻转模式通道输出参考信号 - CHxPRE = 1

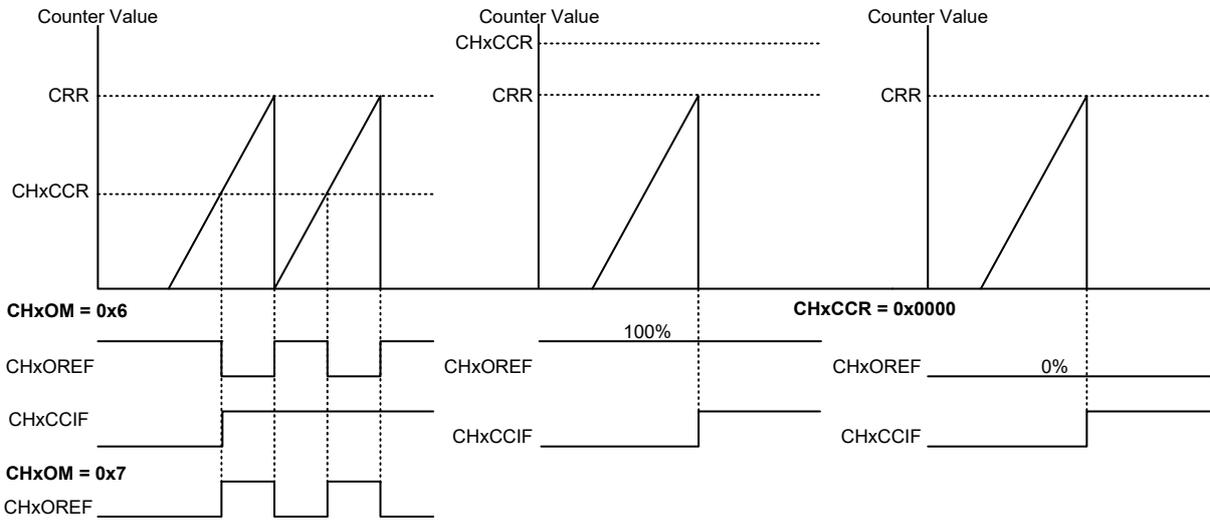


图 53. PWM 模式通道输出参考信号和计数器向上计数模式

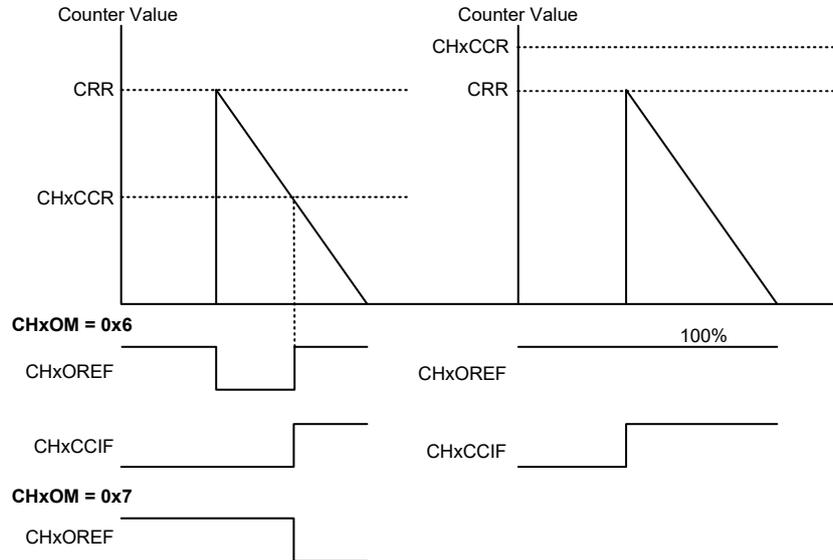


图 54. PWM 模式通道输出参考信号和计数器向下计数模式

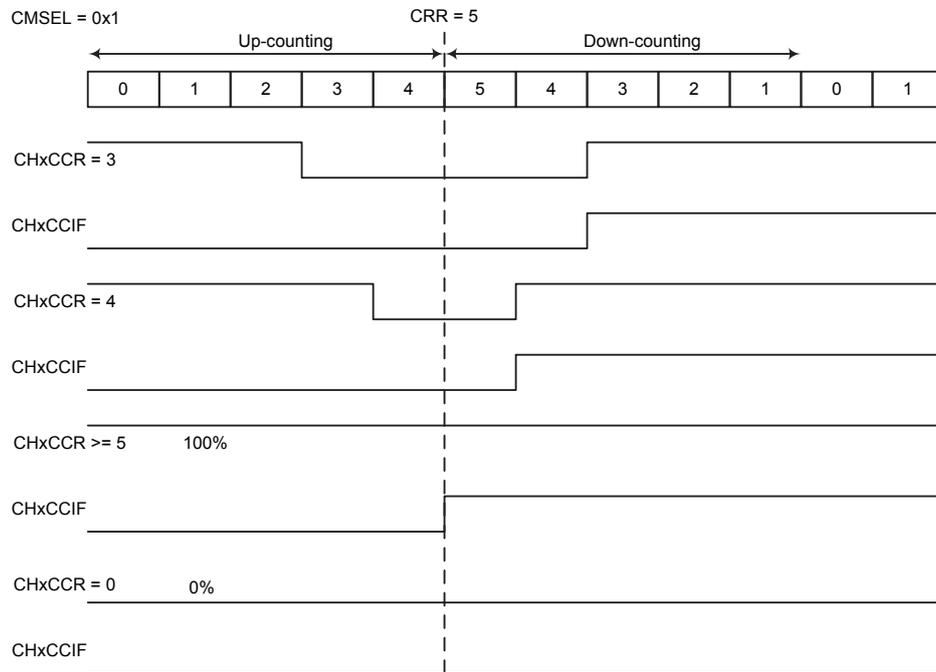


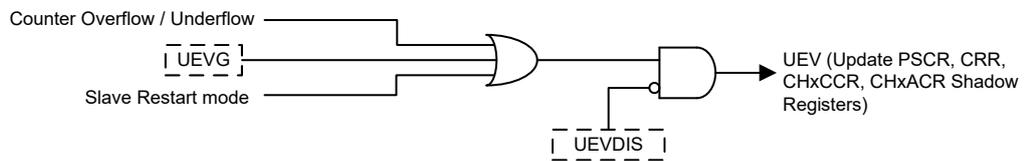
图 55. PWM 模式通道输出参考信号和计数器中心对齐计数模式

更新管理

更新事件用来把实际寄存器 CRR、PSCR、CHxACR 和 CHxCCR 中的值更新到相应的影子寄存器中。更新事件可在计数器上溢 / 下溢、软件更新控制位被触发或从机控制器的更新事件产生时发生。

更新事件是否发生由 CNTCFR 寄存器的 UEVDIS 位控制。通过设置 CNTCFR 寄存器中的 UGDIS 位, 当更新事件发生, 可产生相应的更新事件中断。欲知更多详细信息, 请参考 CNTCFR 寄存器中的 UEVDIS 位和 UGDIS 位的相关定义。

Update Event Management



Update Event Interrupt Management

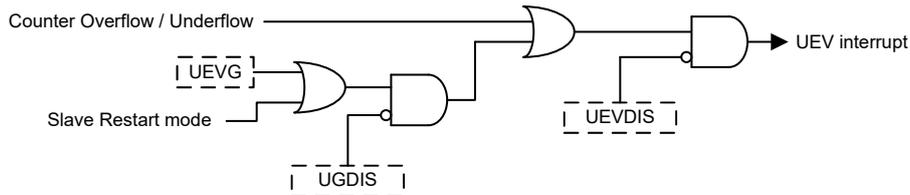


图 56. 更新事件设置方框图

单脉冲模式

一旦定时器被设置工作在单脉冲模式下, 则无需把 CTR 寄存器中的定时器使能位 TME 置为 1 来使能定时器。当 STI 信号上升沿发生时或通过软件把 TME 位设置为 1 时, 触发器将会产生一个脉冲, 然后 TME 位一直保持为高电平直到更新事件发生或使用软件将 TME 清零。如果使用软件将 TME 位清零, 计数器将停止且保持当前值不变。如果 TME 位是由硬件的更新事件自动清零, 计数器将被重新初始化。

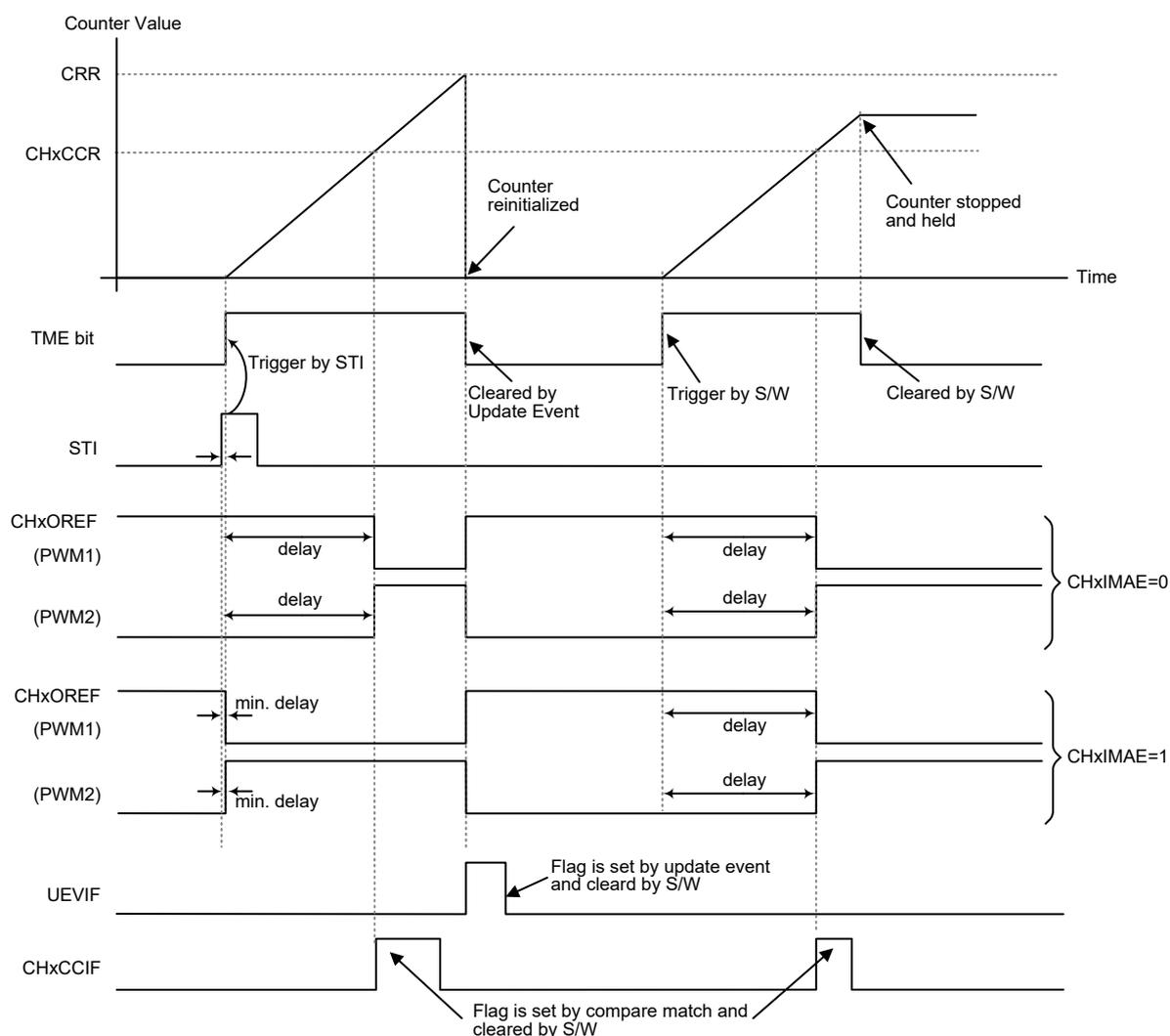


图 57. 单脉冲模式

在单脉冲模式下，STI 有效边沿使 TME 置为 1 时，将使能计数器。然而，由于要执行计数器值和 CHxCCR 值的比较结果，会存在几个时钟的延迟。用户可以通过设置 CHxOCFR 寄存器中的 CHxIMAE 位来使延迟时间减小。单脉冲模式下，STI 上升沿触发发生后，CHxOREF 信号将立即被强制转变为与比较匹配事件发生时相同的电平，而无需考虑比较结果为何。只有当输出通道被配置工作在 PWM 模式 1 或 PWM 模式 2 下且触发源来自于 STI 信号时，CHxIMAE 位才可用。

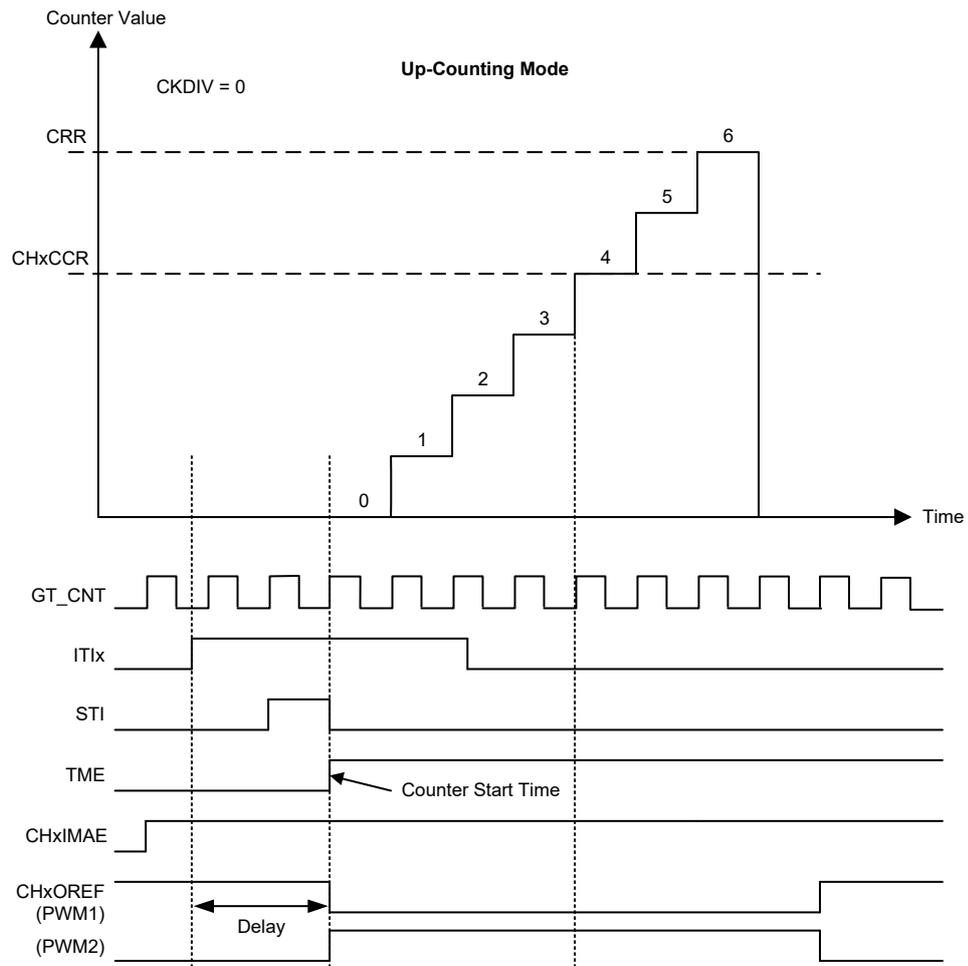


图 58. 立即有效模式的延迟

非对称 PWM 模式

非对称 PWM 模式允许两个中心对齐的 PWM 信号产生一个可编程的相位偏移。PWM 频率由 CRR 寄存器的值决定，占空比和相移由 CHxCCR 和 CHxACR 寄存器决定。当计数器向上计数时，PWM 使用 CHxCCR 的值作为向上计数的比较值。当计数器进入向下计数阶段，PWM 使用 CHxACR 的值作为向下计数的比较值。图 59 是在中心对齐计数模式下的非对称 PWM 模式的一个例子。

注：非对称 PWM 模式只能在中心对齐计数模式下运行。

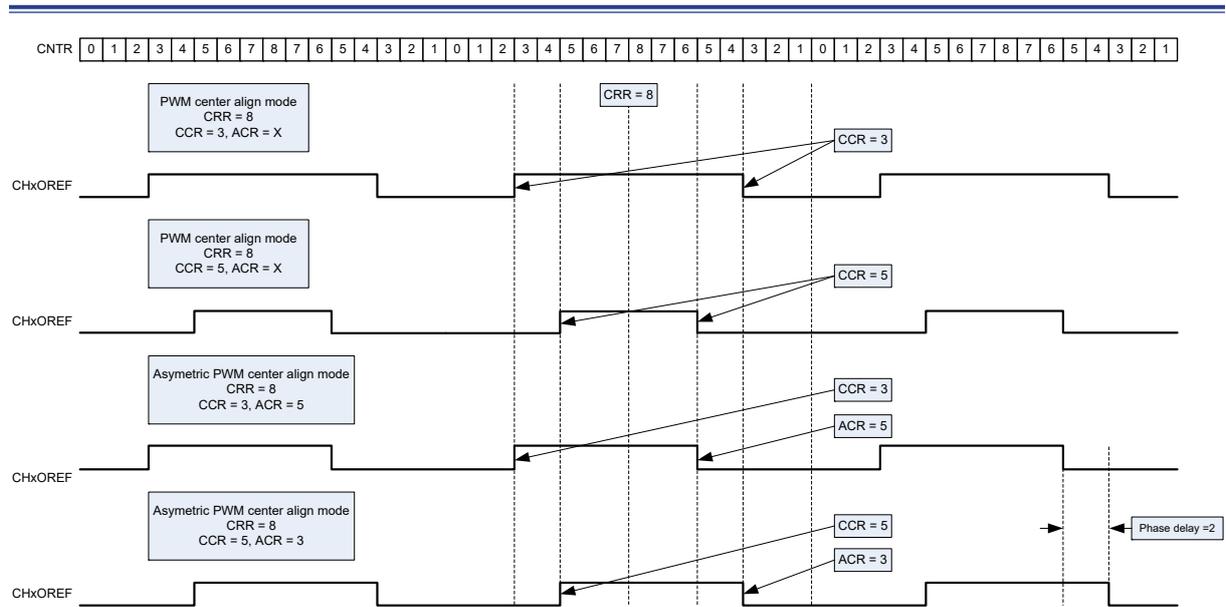


图 59. 非对称 PWM 模式与中心对齐计数模式

定时器互连

定时器可以在内部互相连接用于使定时器链接和同步。它可通过配置一个定时器工作在主机模式，而配置另一个定时器工作在从机模式来实现。下面是主 / 从模式触发器选择的几个示例。

使用一个定时器使能或除能另外一个定时器使其开始或停止计数

- 配置 GPTM 工作在主机模式，发送通道 0 输出参考信号 CH0OREF 作为触发输出 (MMSEL = 0x4)
- 配置 GPTM CH0OREF 波形
- 配置 PWM0 接收的输入触发源来自于 GPTM 触发器输出信号 (TRSEL = 0xA)
- 配置 PWM0 工作在暂停模式下 (SMSSEL = 0x5)
- 通过向 TME 位写入 1 来使能 PWM0
- 通过向 TME 位写入 1 来使能 GPTM

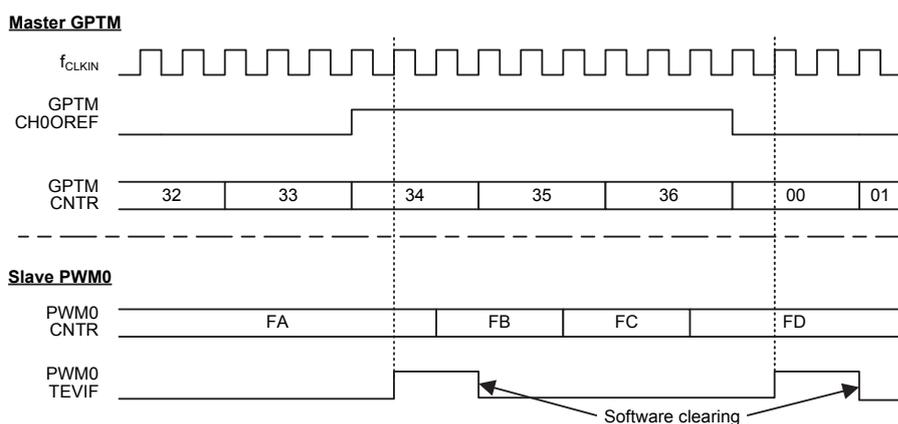


图 60. 用 GPTM CH0OREF 信号暂停 PWM0

使用一个定时器触发另一个定时器开始计数

- 配置 GPTM 工作在主机模式，发送更新事件 UEV 作为触发输出 (MMSEL = 0x2)
- 通过设置 CRR 寄存器配置 GPTM 周期
- 配置 PWM0 的输入触发源来自 GPTM 触发输出信号 (TRSEL = 0xA)
- 配置 PWM0 工作在从机触发模式 (SMSEL = 0x6)
- 向 TME 位写入 1 启动 GPTM

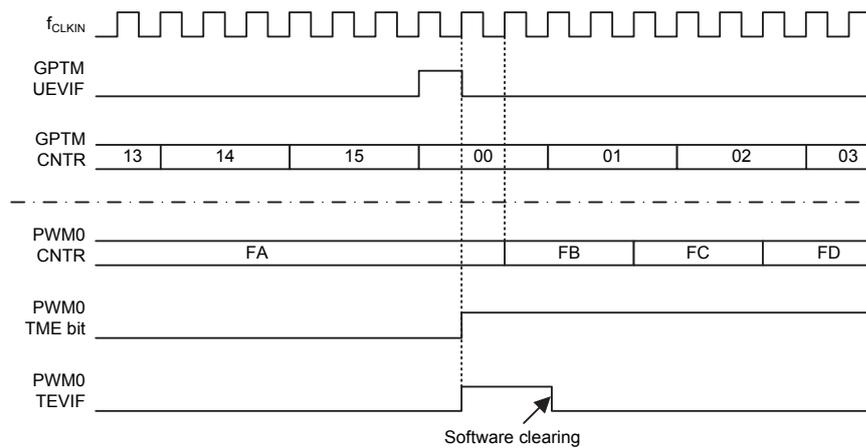


图 61. 用 GPTM 更新事件触发 PWM0

通过外部触发同时启动两个定时器

- 配置 GPTM 工作在主机模式下，发送使能信号作为触发输出信号 (MMSEL = 0x1)
- 配置 GPTM 从机模式，从 GT_CH0 引脚获得其输入触发源 (TRSEL = 0x1)
- 配置 GPTM 工作在从机触发模式 (SMSEL = 0x6)
- 设置 MDCFR 寄存器中的 TSE 位为 1，使能 GPTM 主机定时器同步功能使其与从机定时器同步
- 配置 PWM0 接收的输入触发源来自于 GPTM 的触发输出信号 (TRSEL = 0xA)
- 配置 PWM0 工作在从机触发模式下 (SMSEL = 0x6)

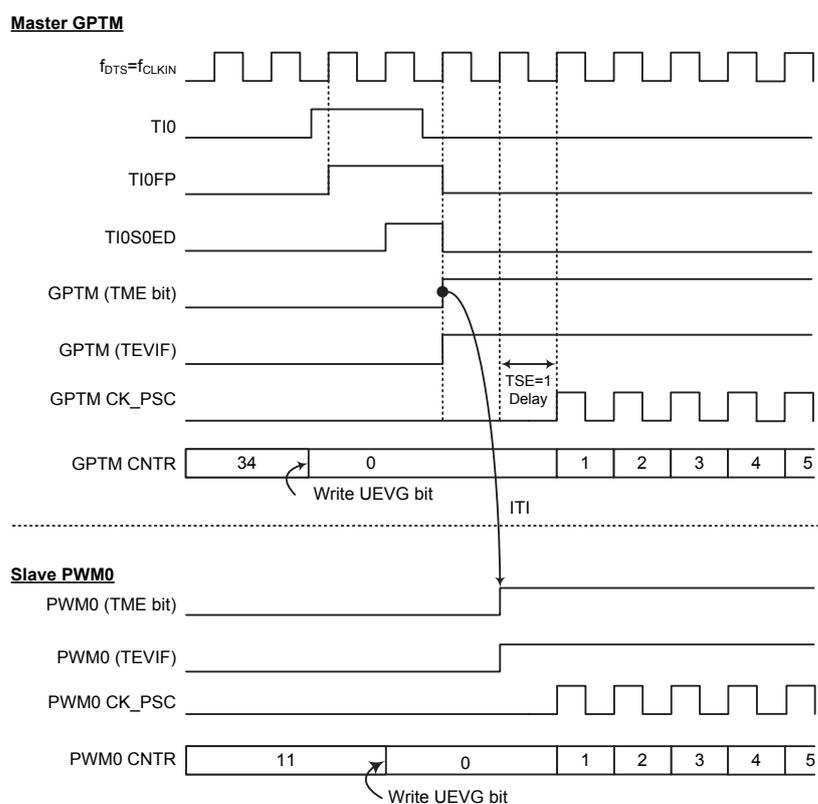


图 62. 用 GPTM CH0 输入信号触发 GPTM 和 PWM0

触发 ADC 开启

为了与模拟 / 数字转换器相连接，GPTM 可以输出 MTO 信号或通道比较匹配输出信号 CHxOREF (x = 0 ~ 3) 作为模拟 / 数字转换器的输入触发信号。

寄存器列表

下表显示了GPTM 寄存器及其复位值。

表 28. GPTM 寄存器列表

寄存器	偏移量	描述	复位值
CNTCFR	0x000	定时器计数器配置寄存器	0x0000_0000
MDCFR	0x004	定时器模式配置寄存器	0x0000_0000
TRCFR	0x008	定时器触发配置寄存器	0x0000_0000
CTR	0x010	定时器控制寄存器	0x0000_0000
CH0ICFR	0x020	通道 0 输入配置寄存器	0x0000_0000
CH1ICFR	0x024	通道 1 输入配置寄存器	0x0000_0000
CH2ICFR	0x028	通道 2 输入配置寄存器	0x0000_0000
CH3ICFR	0x02C	通道 3 输入配置寄存器	0x0000_0000
CH0OCFR	0x040	通道 0 输出配置寄存器	0x0000_0000
CH1OCFR	0x044	通道 1 输出配置寄存器	0x0000_0000
CH2OCFR	0x048	通道 2 输出配置寄存器	0x0000_0000
CH3OCFR	0x04C	通道 3 输出配置寄存器	0x0000_0000
CHCTR	0x050	通道控制寄存器	0x0000_0000
CHPOLR	0x054	通道极性配置寄存器	0x0000_0000
DICTR	0x074	定时器中断控制寄存器	0x0000_0000
EVGR	0x078	定时器事件发生器寄存器	0x0000_0000
INTSR	0x07C	定时器中断状态寄存器	0x0000_0000
CNTR	0x080	定时器计数器寄存器	0x0000_0000
PSCR	0x084	定时器预分频器寄存器	0x0000_0000
CRR	0x088	定时器计数器重载寄存器	0x0000_FFFF
CH0CCR	0x090	通道 0 捕捉 / 比较寄存器	0x0000_0000
CH1CCR	0x094	通道 1 捕捉 / 比较寄存器	0x0000_0000
CH2CCR	0x098	通道 2 捕捉 / 比较寄存器	0x0000_0000
CH3CCR	0x09C	通道 3 捕捉 / 比较寄存器	0x0000_0000
CH0ACR	0x0A0	通道 0 非对称比较寄存器	0x0000_0000
CH1ACR	0x0A4	通道 1 非对称比较寄存器	0x0000_0000
CH2ACR	0x0A8	通道 2 非对称比较寄存器	0x0000_0000
CH3ACR	0x0AC	通道 3 非对称比较寄存器	0x0000_0000

寄存器描述

定时器计数器配置寄存器 – CNTCFR

该寄存器定义了 GPTM 计数器配置。

偏移量: 0x000
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24		
	保留位							DIR		
类型 / 复位								RW	0	
	23	22	21	20	19	18	17	16		
	保留位						CMSEL			
类型 / 复位							RW	0	RW	0
	15	14	13	12	11	10	9	8		
	保留位						CKDIV			
类型 / 复位							RW	0	RW	0
	7	6	5	4	3	2	1	0		
	保留位						UGDIS		UEVDIS	
类型 / 复位							RW	0	RW	0

位	字段	描述
[24]	DIR	计数方向 0: 向上计数 1: 向下计数 注: 当定时器工作在中心对齐计数模式下或作为正交解码器使用时, 此位为只读位。
[17:16]	CMSEL	计数器模式选择 00: 边沿对齐计数模式。此模式下可正常向上和向下计数。计数方向由 DIR 位定义。 01: 中心对齐计数模式 1。计数器向上向下交替计数。比较匹配中断标志位在向下计数期间被置位。 10: 中心对齐计数模式 2。计数器向上向下交替计数。比较匹配中断标志位在向上计数期间被置位。 11: 中心对齐计数模式 3。计数器向上向下交替计数。比较匹配中断标志位在向上和向下计数期间被置位。
[9:8]	CKDIV	时钟分频 这两位定义了定时器时钟 (f_{CLKIN}) 和死区时钟 (f_{DTS}) 之间的频率比值。死区时钟也可作为数字滤波器采样时钟使用。 00: $f_{DTS} = f_{CLKIN}$ 01: $f_{DTS} = f_{CLKIN} / 2$ 10: $f_{DTS} = f_{CLKIN} / 4$ 11: 保留
[1]	UGDIS	更新事件中产生除能控制 0: 以下任何一个事件都可产生一个更新中断 – 计数器上溢 / 下溢 – 设置 UEVG 位 – 通过从机模式产生更新 1: 只在计数器上溢 / 下溢时产生一个更新中断

位	字段	描述
[0]	UEVDIS	更新事件除能控制 0: 以下任何一个事件都可使能更新事件请求 - 计数器上溢 / 下溢 - 设置 UEVG 位 - 通过从机模式产生更新 1: 除能更新事件 (如果 UEVG 位被置位或从从机模式收到硬件重启, 那么计数器和预分频器将被初始化)

定时器模式配置寄存器 – MDCFR

该寄存器定义了 GPTM 主机和从机模式选项以及单脉冲模式。

偏移量: 0x004

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
	保留位							SPMSET	
类型 / 复位								RW	0
	23	22	21	20	19	18	17	16	
	保留位					MMSEL			
类型 / 复位						RW	0	RW	0
	15	14	13	12	11	10	9	8	
	保留位					SMSEL			
类型 / 复位						RW	0	RW	0
	7	6	5	4	3	2	1	0	
	保留位							TSE	
类型 / 复位								RW	0

位	字段	描述
[24]	SPMSET	单脉冲模式设置 0: 无论更新事件是否发生, 计数器正常计数 1: 下一个更新事件到来时, 计数器停止计数, 接着 TME 位被硬件清零

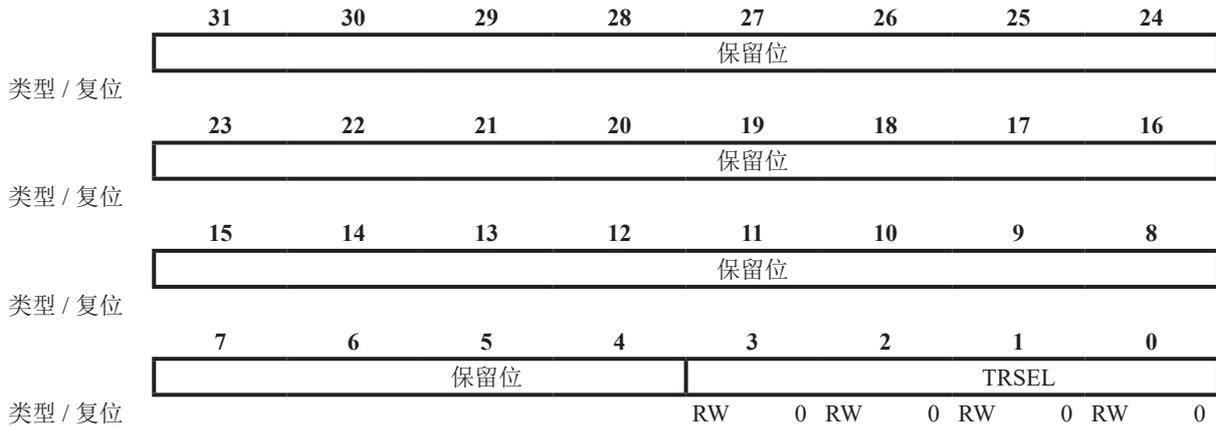
位	字段	描述																											
[18:16]	MMSEL	<p>主机模式选项</p> <p>主机模式选项用来选择与其它从机定时器同步的 MTO 信号源。</p> <table border="1"> <thead> <tr> <th>MMSEL[2:0]</th> <th>模式</th> <th>描述</th> </tr> </thead> <tbody> <tr> <td>000</td> <td>复位模式</td> <td>复位模式下的 MTO 输出信号由以下条件之一产生： 1. 软件设置 UEVG 位 2. 当定时器用于从机重启模式，STI 触发输入信号被输出到 MTO 信号线</td> </tr> <tr> <td>001</td> <td>使能模式</td> <td>计数器使能信号作为触发输出</td> </tr> <tr> <td>010</td> <td>更新模式</td> <td>当 UEVDIS 位被清零时，下列任一事件发生时更新事件将作为触发输出： 1. 计数器上溢 / 下溢 2. 软件设置 UEVG 位 3. 从机重启模式下从机触发输入</td> </tr> <tr> <td>011</td> <td>捕捉 / 比较模式</td> <td>当通道 0 捕捉或比较匹配事件发生，将产生一个正脉冲作为主机触发输出。</td> </tr> <tr> <td>100</td> <td>比较模式 0</td> <td>通道 0 输出参考信号 CH0OREF 作为触发输出。</td> </tr> <tr> <td>101</td> <td>比较模式 1</td> <td>通道 1 输出参考信号 CH1OREF 作为触发输出。</td> </tr> <tr> <td>110</td> <td>比较模式 2</td> <td>通道 2 输出参考信号 CH2OREF 作为触发输出。</td> </tr> <tr> <td>111</td> <td>比较模式 3</td> <td>通道 3 输出参考信号 CH3OREF 作为触发输出。</td> </tr> </tbody> </table>	MMSEL[2:0]	模式	描述	000	复位模式	复位模式下的 MTO 输出信号由以下条件之一产生： 1. 软件设置 UEVG 位 2. 当定时器用于从机重启模式，STI 触发输入信号被输出到 MTO 信号线	001	使能模式	计数器使能信号作为触发输出	010	更新模式	当 UEVDIS 位被清零时，下列任一事件发生时更新事件将作为触发输出： 1. 计数器上溢 / 下溢 2. 软件设置 UEVG 位 3. 从机重启模式下从机触发输入	011	捕捉 / 比较模式	当通道 0 捕捉或比较匹配事件发生，将产生一个正脉冲作为主机触发输出。	100	比较模式 0	通道 0 输出参考信号 CH0OREF 作为触发输出。	101	比较模式 1	通道 1 输出参考信号 CH1OREF 作为触发输出。	110	比较模式 2	通道 2 输出参考信号 CH2OREF 作为触发输出。	111	比较模式 3	通道 3 输出参考信号 CH3OREF 作为触发输出。
MMSEL[2:0]	模式	描述																											
000	复位模式	复位模式下的 MTO 输出信号由以下条件之一产生： 1. 软件设置 UEVG 位 2. 当定时器用于从机重启模式，STI 触发输入信号被输出到 MTO 信号线																											
001	使能模式	计数器使能信号作为触发输出																											
010	更新模式	当 UEVDIS 位被清零时，下列任一事件发生时更新事件将作为触发输出： 1. 计数器上溢 / 下溢 2. 软件设置 UEVG 位 3. 从机重启模式下从机触发输入																											
011	捕捉 / 比较模式	当通道 0 捕捉或比较匹配事件发生，将产生一个正脉冲作为主机触发输出。																											
100	比较模式 0	通道 0 输出参考信号 CH0OREF 作为触发输出。																											
101	比较模式 1	通道 1 输出参考信号 CH1OREF 作为触发输出。																											
110	比较模式 2	通道 2 输出参考信号 CH2OREF 作为触发输出。																											
111	比较模式 3	通道 3 输出参考信号 CH3OREF 作为触发输出。																											
[10:8]	SMSEL	<p>从机模式选项</p> <table border="1"> <thead> <tr> <th>SMSEL[2:0]</th> <th>模式</th> <th>描述</th> </tr> </thead> <tbody> <tr> <td>000</td> <td>除能模式</td> <td>预分频器直接用内部时钟计时。</td> </tr> <tr> <td>001</td> <td>正交解码模式 1</td> <td>计数器使用由 TI0 和 TI1 信号交互产生的时钟脉冲来驱动计数器预分频器。此模式下的 TI0 边沿转换取决于 TI1 的电平。</td> </tr> <tr> <td>010</td> <td>正交解码模式 2</td> <td>计数器使用由 TI0 和 TI1 信号交互产生的时钟脉冲来驱动计数器预分频器。此模式下的 TI1 边沿转换取决于 TI0 的电平。</td> </tr> <tr> <td>011</td> <td>正交解码模式 3</td> <td>计数器使用由 TI0 和 TI1 信号交互产生的时钟脉冲来驱动计数器预分频器。在此模式下的一个通道边沿转换取决于另外一个通道的电平。</td> </tr> <tr> <td>100</td> <td>重启模式</td> <td>计数器在 STI 信号上升沿，依计数模式从 0 或 CRR 影子寄存器的值重新开始计数。寄存器也将被更新。</td> </tr> <tr> <td>101</td> <td>暂停模式</td> <td>当所选择的触发输入信号 STI 是高电平时，计数器开始计数。当 STI 信号转换成低电平时，计数器立即停止计数，但不会产生复位。计数器开始和停止计数是由 STI 信号控制的。</td> </tr> <tr> <td>110</td> <td>触发模式</td> <td>在所选择的 STI 触发信号上升沿处，计数器从初始值开始计数。STI 信号只能控制计数器的开启。</td> </tr> <tr> <td>111</td> <td>STIED</td> <td>计数器使用所选的触发信号 STI 的上升沿计时。</td> </tr> </tbody> </table>	SMSEL[2:0]	模式	描述	000	除能模式	预分频器直接用内部时钟计时。	001	正交解码模式 1	计数器使用由 TI0 和 TI1 信号交互产生的时钟脉冲来驱动计数器预分频器。此模式下的 TI0 边沿转换取决于 TI1 的电平。	010	正交解码模式 2	计数器使用由 TI0 和 TI1 信号交互产生的时钟脉冲来驱动计数器预分频器。此模式下的 TI1 边沿转换取决于 TI0 的电平。	011	正交解码模式 3	计数器使用由 TI0 和 TI1 信号交互产生的时钟脉冲来驱动计数器预分频器。在此模式下的一个通道边沿转换取决于另外一个通道的电平。	100	重启模式	计数器在 STI 信号上升沿，依计数模式从 0 或 CRR 影子寄存器的值重新开始计数。寄存器也将被更新。	101	暂停模式	当所选择的触发输入信号 STI 是高电平时，计数器开始计数。当 STI 信号转换成低电平时，计数器立即停止计数，但不会产生复位。计数器开始和停止计数是由 STI 信号控制的。	110	触发模式	在所选择的 STI 触发信号上升沿处，计数器从初始值开始计数。STI 信号只能控制计数器的开启。	111	STIED	计数器使用所选的触发信号 STI 的上升沿计时。
SMSEL[2:0]	模式	描述																											
000	除能模式	预分频器直接用内部时钟计时。																											
001	正交解码模式 1	计数器使用由 TI0 和 TI1 信号交互产生的时钟脉冲来驱动计数器预分频器。此模式下的 TI0 边沿转换取决于 TI1 的电平。																											
010	正交解码模式 2	计数器使用由 TI0 和 TI1 信号交互产生的时钟脉冲来驱动计数器预分频器。此模式下的 TI1 边沿转换取决于 TI0 的电平。																											
011	正交解码模式 3	计数器使用由 TI0 和 TI1 信号交互产生的时钟脉冲来驱动计数器预分频器。在此模式下的一个通道边沿转换取决于另外一个通道的电平。																											
100	重启模式	计数器在 STI 信号上升沿，依计数模式从 0 或 CRR 影子寄存器的值重新开始计数。寄存器也将被更新。																											
101	暂停模式	当所选择的触发输入信号 STI 是高电平时，计数器开始计数。当 STI 信号转换成低电平时，计数器立即停止计数，但不会产生复位。计数器开始和停止计数是由 STI 信号控制的。																											
110	触发模式	在所选择的 STI 触发信号上升沿处，计数器从初始值开始计数。STI 信号只能控制计数器的开启。																											
111	STIED	计数器使用所选的触发信号 STI 的上升沿计时。																											
[0]	TSE	<p>定时器同步使能</p> <p>0: 无动作</p> <p>1: 主机定时器 (当前定时器) 将通过 MTO 信号产生一个延时以同步其从机定时器</p>																											

定时器触发配置寄存器 – TRCFR

该寄存器定义了 GPTM 的触发源选项。

偏移量: 0x008

复位值: 0x0000_0000



位	字段	描述
[3:0]	TRSEL	触发源选择 这些位用来选择用于计数器同步的触发输入源 (STI) 0000: 通过设置 UEVG 位软件触发 0001: 通道 0 滤波输入 (TI0S0) 0010: 通道 1 滤波输入 (TI1S1) 0011: 保留 1000: 通道 0 边沿检测器 (TI0BED) 1001: 内部定时器模块触发器 0 (ITI0) 1010: 内部定时器模块触发器 1 (ITI1) 1011: 内部定时器模块触发器 2 (ITI2) 其它: 保留 注: 当 SMSEL 字段为 0x0 除能从机模式时, 这些位才能被更新。

表 29. GPTM 内部触发器连接

从机定时器模块	ITI0	ITI1	ITI2
GPTM	PWM0	—	PWM1

定时器控制寄存器 – CTR

该寄存器定义了定时器使能位 (TME) 和 CRR 缓冲器使能位 (CRBE)。

偏移量: 0x010

复位值: 0x0000_0000

31	30	29	28	27	26	25	24
保留位							
类型 / 复位							
23	22	21	20	19	18	17	16
保留位							
类型 / 复位							
15	14	13	12	11	10	9	8
保留位							
类型 / 复位							
7	6	5	4	3	2	1	0
保留位						CRBE	TME
类型 / 复位						RW	0 RW 0

位	字段	描述
[1]	CRBE	计数器重载寄存器缓冲器使能位 0: 计数器重载寄存器能立即被更新 1: 直到更新事件发生时计数器重载寄存器才会被更新
[0]	TME	定时器使能位 0: GPTM 关闭 1: GPTM 开启 – GPTM 功能正常 当 TME 位被清零, 计数器停止计数且 GPTM 无功耗 (单脉冲模式和从机触发模式除外)。在单脉冲模式和从机触发模式中, TME 位可通过硬件自动置 1, 允许所有的 GPTM 寄存器正常工作。

通道 0 输入配置寄存器 – CH0ICFR

该寄存器定义了通道 0 输入模式配置。

偏移量: 0x020

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24		
	TI0SRC		保留位							
类型 / 复位	RW	0								
	23	22	21	20	19	18	17	16		
	保留位			CH0PSC			CH0CCS			
类型 / 复位				RW	0	RW	0	RW	0	
	15	14	13	12	11	10	9	8		
	保留位									
类型 / 复位										
	7	6	5	4	3	2	1	0		
	保留位				TIOF					
类型 / 复位					RW	0	RW	0	RW	0

位	字段	描述
[31]	TI0SRC	通道 0 输入源 TI0 选择 0: GT_CH0 引脚连接到通道 0 输入 TI0 1: GT_CH0、GT_CH1 和 GT_CH2 引脚异或操作后的输出连接到通道 0 输入 TI0
[19:18]	CH0PSC	通道 0 捕捉输入源预分频器设置 这些位定义了通道 0 捕捉输入的有效事件。注意，一旦通道控制寄存器 CHCTR 中的通道 0 捕捉 / 比较使能位 CH0E 被清零，则预分频器将被复位。 00: 无分频，通道 0 每个有效事件都进行一个输入捕捉 01: 通道 0 每 2 个事件发生才进行一个输入捕捉 10: 通道 0 每 4 个事件发生才进行一个输入捕捉 11: 通道 0 每 8 个事件发生才进行一个输入捕捉
[17:16]	CH0CCS	通道 0 捕捉 / 比较选择 00: 通道 0 配置为输出 01: 通道 0 配置为来自 TI0 信号的一个输入 10: 通道 0 配置为来自 TI1 信号的一个输入 11: 通道 0 配置为来自触发控制器产生的 TRCED 信号的一个输入 注：仅当 CH0E 位清零时，才可以访问 CH0CCS 字段。

位	字段	描述
[3:0]	TI0F	<p>通道 0 输入源 TI0 滤波器设置</p> <p>这些位定义了用来采样 TI0 信号的采样频率和对 TI0 信号数字滤波的带宽。GPTM 中的数字滤波器是一个事件计数器，它记录到 N 个事件后会产生一个输出的跳变。</p> <p>0000: 无滤波，采样时钟是 f_{SYSTEM}</p> <p>0001: $f_{sampling} = f_{CLKIN}$, N = 2</p> <p>0010: $f_{sampling} = f_{CLKIN}$, N = 4</p> <p>0011: $f_{sampling} = f_{CLKIN}$, N = 8</p> <p>0100: $f_{sampling} = f_{DTS} / 2$, N = 6</p> <p>0101: $f_{sampling} = f_{DTS} / 2$, N = 8</p> <p>0110: $f_{sampling} = f_{DTS} / 4$, N = 6</p> <p>0111: $f_{sampling} = f_{DTS} / 4$, N = 8</p> <p>1000: $f_{sampling} = f_{DTS} / 8$, N = 6</p> <p>1001: $f_{sampling} = f_{DTS} / 8$, N = 8</p> <p>1010: $f_{sampling} = f_{DTS} / 16$, N = 5</p> <p>1011: $f_{sampling} = f_{DTS} / 16$, N = 6</p> <p>1100: $f_{sampling} = f_{DTS} / 16$, N = 8</p> <p>1101: $f_{sampling} = f_{DTS} / 32$, N = 5</p> <p>1110: $f_{sampling} = f_{DTS} / 32$, N = 6</p> <p>1111: $f_{sampling} = f_{DTS} / 32$, N = 8</p>

通道 1 输入配置寄存器 – CH1ICFR

该寄存器定义了通道 1 输入模式配置。

偏移量: 0x024
复位值: 0x0000_0000

31	30	29	28	27	26	25	24
保留位							
23	22	21	20	19	18	17	16
保留位				CHIPSC		CH1CCS	
				RW	0	RW	0
				RW	0	RW	0
				RW	0	RW	0
15	14	13	12	11	10	9	8
保留位							
7	6	5	4	3	2	1	0
保留位				TIIF			
				RW	0	RW	0
				RW	0	RW	0
				RW	0	RW	0

位	字段	描述
[19:18]	CHIPSC	通道 1 捕捉输入源预分频器设置 这些位定义了通道 1 捕捉输入的有效事件。注意，一旦通道控制寄存器 CHCTR 中的通道 1 捕捉 / 比较使能位 CH1E 被清零，则预分频器将被复位。 00: 无分频，通道 1 每个有效事件都进行一个输入捕捉 01: 通道 1 每 2 个事件发生才进行一个输入捕捉 10: 通道 1 每 4 个事件发生才进行一个输入捕捉 11: 通道 1 每 8 个事件发生才进行一个输入捕捉
[17:16]	CH1CCS	通道 1 捕捉 / 比较选项 00: 通道 1 被配置为输出 01: 通道 1 被配置为来自 TI1 信号的一个输入 10: 通道 1 被配置为来自 TI0 信号的一个输入 11: 通道 1 被配置为来自触发控制器产生的 TRCED 信号的一个输入 注: 仅当 CH1E 位清零时，才可以访问 CH1CCS 字段。
[3:0]	TIIF	通道 1 输入源 TI1 滤波器设置 这些位定义了用来采样 TI1 信号的采样频率和对 TI1 信号数字滤波的带宽。GPTM 中的数字滤波器是一个事件计数器，它记录到 N 个事件后会产生一个输出的跳变。 0000: 无滤波，采样时钟是 f_{SYSTEM} 0001: $f_{sampling} = f_{CLKIN}$, $N = 2$ 0010: $f_{sampling} = f_{CLKIN}$, $N = 4$ 0011: $f_{sampling} = f_{CLKIN}$, $N = 8$ 0100: $f_{sampling} = f_{DTS} / 2$, $N = 6$ 0101: $f_{sampling} = f_{DTS} / 2$, $N = 8$ 0110: $f_{sampling} = f_{DTS} / 4$, $N = 6$ 0111: $f_{sampling} = f_{DTS} / 4$, $N = 8$ 1000: $f_{sampling} = f_{DTS} / 8$, $N = 6$ 1001: $f_{sampling} = f_{DTS} / 8$, $N = 8$ 1010: $f_{sampling} = f_{DTS} / 16$, $N = 5$ 1011: $f_{sampling} = f_{DTS} / 16$, $N = 6$ 1100: $f_{sampling} = f_{DTS} / 16$, $N = 8$ 1101: $f_{sampling} = f_{DTS} / 32$, $N = 5$ 1110: $f_{sampling} = f_{DTS} / 32$, $N = 6$ 1111: $f_{sampling} = f_{DTS} / 32$, $N = 8$

通道 2 输入配置寄存器 – CH2ICFR

该寄存器定义了通道 2 输入模式配置。

偏移量: 0x028
复位值: 0x0000_0000

31	30	29	28	27	26	25	24
保留位							
23	22	21	20	19	18	17	16
保留位				CH2PSC		CH2CCS	
				RW	0	RW	0
				RW	0	RW	0
				RW	0	RW	0
15	14	13	12	11	10	9	8
保留位							
7	6	5	4	3	2	1	0
保留位				TI2F			
				RW	0	RW	0
				RW	0	RW	0
				RW	0	RW	0

位	字段	描述
[19:18]	CH2PSC	通道 2 捕捉输入源预分频器设置 这些位定义了通道 2 捕捉输入的有效事件。注意，一旦通道控制寄存器 CHCTR 中的通道 2 捕捉 / 比较使能位 CH2E 被清零，则预分频器将被复位。 00: 无分频，通道 2 每个有效事件都进行一个输入捕捉 01: 通道 2 每 2 个事件发生才进行一个输入捕捉 10: 通道 2 每 4 个事件发生才进行一个输入捕捉 11: 通道 2 每 8 个事件发生才进行一个输入捕捉
[17:16]	CH2CCS	通道 2 捕捉 / 比较选项 00: 通道 2 被配置为输出 01: 通道 2 被配置为来自 TI2 信号的一个输入 10: 通道 2 被配置为来自 TI3 信号的一个输入 11: 通道 2 被配置为来自触发控制器产生的 TRCED 信号的一个输入 注: 仅当 CH2E 位清零时，才可以访问 CH2CCS 字段。
[3:0]	TI2F	通道 2 输入源 TI2 滤波器设置 这些位定义了用来采样 TI2 信号的采样频率和对 TI2 信号数字滤波的带宽。GPTM 中的数字滤波器是一个事件计数器，它记录到 N 个事件后会产生一个输出的跳变。 0000: 无滤波，采样时钟是 f_{SYSTEM} 0001: $f_{sampling} = f_{CLKIN}$, $N = 2$ 0010: $f_{sampling} = f_{CLKIN}$, $N = 4$ 0011: $f_{sampling} = f_{CLKIN}$, $N = 8$ 0100: $f_{sampling} = f_{DTS} / 2$, $N = 6$ 0101: $f_{sampling} = f_{DTS} / 2$, $N = 8$ 0110: $f_{sampling} = f_{DTS} / 4$, $N = 6$ 0111: $f_{sampling} = f_{DTS} / 4$, $N = 8$ 1000: $f_{sampling} = f_{DTS} / 8$, $N = 6$ 1001: $f_{sampling} = f_{DTS} / 8$, $N = 8$ 1010: $f_{sampling} = f_{DTS} / 16$, $N = 5$ 1011: $f_{sampling} = f_{DTS} / 16$, $N = 6$ 1100: $f_{sampling} = f_{DTS} / 16$, $N = 8$ 1101: $f_{sampling} = f_{DTS} / 32$, $N = 5$ 1110: $f_{sampling} = f_{DTS} / 32$, $N = 6$ 1111: $f_{sampling} = f_{DTS} / 32$, $N = 8$

通道 3 输入配置寄存器 – CH3ICFR

该寄存器定义了通道 3 输入模式配置。

偏移量: 0x02C
复位值: 0x0000_0000

31	30	29	28	27	26	25	24
保留位							
23	22	21	20	19	18	17	16
保留位				CH3PSC		CH3CCS	
				RW	0	RW	0
				RW	0	RW	0
15	14	13	12	11	10	9	8
保留位							
7	6	5	4	3	2	1	0
保留位				TI3F			
				RW	0	RW	0
				RW	0	RW	0

位	字段	描述
[19:18]	CH3PSC	通道 3 捕捉输入源预分频器设置 这些位定义了通道 3 捕捉输入的有效事件。注意，一旦通道控制寄存器 CHCTR 中的通道 3 捕捉 / 比较使能位 CH3E 被清零，则预分频器将被复位。 00: 无分频，通道 3 每个有效事件都进行一个输入捕捉 01: 通道 3 每 2 个事件发生才进行一个输入捕捉 10: 通道 3 每 4 个事件发生才进行一个输入捕捉 11: 通道 3 每 8 个事件发生才进行一个输入捕捉
[17:16]	CH3CCS	通道 3 捕捉 / 比较选项 00: 通道 3 被配置为输出 01: 通道 3 被配置为来自 TI3 信号的一个输入 10: 通道 3 被配置为来自 TI2 信号的一个输入 11: 通道 3 被配置为来自触发控制器产生的 TRCED 信号的一个输入 注: 仅当 CH3E 位清零时，才可以访问 CH3CCS 字段。
[3:0]	TI3F	通道 3 输入源 TI3 滤波器设置 这些位定义了用来采样 TI3 信号的采样频率和对 TI3 信号数字滤波的带宽。GPTM 中的数字滤波器是一个事件计数器，它记录到 N 个事件后会产生一个输出的跳变。 0000: 无滤波，采样时钟是 f_{SYSTEM} 0001: $f_{sampling} = f_{CLKIN}$, N = 2 0010: $f_{sampling} = f_{CLKIN}$, N = 4 0011: $f_{sampling} = f_{CLKIN}$, N = 8 0100: $f_{sampling} = f_{DTS} / 2$, N = 6 0101: $f_{sampling} = f_{DTS} / 2$, N = 8 0110: $f_{sampling} = f_{DTS} / 4$, N = 6 0111: $f_{sampling} = f_{DTS} / 4$, N = 8 1000: $f_{sampling} = f_{DTS} / 8$, N = 6 1001: $f_{sampling} = f_{DTS} / 8$, N = 8 1010: $f_{sampling} = f_{DTS} / 16$, N = 5 1011: $f_{sampling} = f_{DTS} / 16$, N = 6 1100: $f_{sampling} = f_{DTS} / 16$, N = 8 1101: $f_{sampling} = f_{DTS} / 32$, N = 5 1110: $f_{sampling} = f_{DTS} / 32$, N = 6 1111: $f_{sampling} = f_{DTS} / 32$, N = 8

通道 0 输出配置寄存器 – CH0OCFR

该寄存器定义了通道 0 输出模式配置。

偏移量: 0x040

复位值: 0x0000_0000

31	30	29	28	27	26	25	24	保留位												
类型 / 复位																				
23	22	21	20	19	18	17	16	保留位												
类型 / 复位																				
15	14	13	12	11	10	9	8	保留位				CH0OM[3]								
											RW	0								
7	6	5	4	3	2	1	0	保留位	CH0IMAE	CH0PRE	保留位	CH0OM[2:0]								
											RW	0	RW	0	RW	0	RW	0	RW	0
类型 / 复位																				

位	字段	描述
[5]	CH0IMAE	通道 0 立即有效使能位 0: 无动作 1: 单脉冲立即有效模式使能 无论 CNTR 和 CH0CCR 值的比较结果如何, 在一个有效触发事件发生后, CH0OREF 会立即强制为比较匹配电平。 比较匹配电平将维持到下一个上溢或下溢事件的到来。 注: 只有当通道 0 被配置工作在 PWM 模式 1 或 PWM 模式 2 时, CH0IMAE 位可用。
[4]	CH0PRE	通道 0 捕捉 / 比较寄存器 (CH0CCR) 预载使能位 0: CH0CCR 预载功能除能 当 CH0PRE 位清零, CH0CCR 寄存器将立即被更新为新的值, 且立即可用。 1: CH0CCR 预载功能使能 直到更新事件发生后, 新的 CH0CCR 值才会被传送到影子寄存器中。

位	字段	描述
[8][2:0]	CH0OM[3:0]	<p>通道 0 输出模式设置</p> <p>这些位定义了输出参考信号 CH0OREF 的功能类型</p> <p>0000: 无变化</p> <p>0001: 比较匹配时输出 0</p> <p>0010: 比较匹配时输出 1</p> <p>0011: 比较匹配时输出翻转</p> <p>0100: 强制无效 – CH0OREF 强制为 0</p> <p>0101: 强制有效 – CH0OREF 强制为 1</p> <p>0110: PWM 模式 1</p> <ul style="list-style-type: none"> – 在向上计数期间, 当 CNTR < CH0CCR, 通道 0 处于有效电平, 否则将处于无效电平。 – 在向下计数期间, 当 CNTR > CH0CCR, 通道 0 处于无效电平, 否则将处于有效电平。 <p>0111: PWM 模式 2</p> <ul style="list-style-type: none"> – 在向上计数期间, 当 CNTR < CH0CCR, 通道 0 处于无效电平, 否则将处于有效电平。 – 在向下计数期间, 当 CNTR > CH0CCR, 通道 0 处于有效电平, 否则将处于无效电平。 <p>1110: 非对称 PWM 模式 1</p> <ul style="list-style-type: none"> – 在向上计数期间, 当 CNTR < CH0CCR, 通道 0 处于有效电平, 否则将处于无效电平。 – 在向下计数期间, 当 CNTR > CH0CCR, 通道 0 处于无效电平, 否则将处于有效电平。 <p>1111: 非对称 PWM 模式 2</p> <ul style="list-style-type: none"> – 在向上计数期间, 当 CNTR < CH0CCR, 通道 0 处于无效电平, 否则将处于有效电平。 – 在向下计数期间, 当 CNTR > CH0CCR, 通道 0 处于有效电平, 否则将处于无效电平。 <p>注: 当通道 0 用作非对称 PWM 输出模式时, 计数器配置寄存器中的计数器模式选择位必须配置为中心对齐计数模式 (CMSEL = 0x1 / 0x2 / 0x3)。</p>

通道 1 输出配置寄存器 – CH1OCFR

该寄存器定义了通道 1 输出模式配置。

偏移量: 0x044

复位值: 0x0000_0000

31	30	29	28	27	26	25	24	保留位													
类型 / 复位																					
23	22	21	20	19	18	17	16	保留位													
类型 / 复位																					
15	14	13	12	11	10	9	8	保留位				CH1OM[3]									
类型 / 复位												RW	0								
7	6	5	4	3	2	1	0	保留位	CH1IMAE	CH1PRE	保留位	CH1OM[2:0]									
类型 / 复位												RW	0	RW	0	RW	0	RW	0	RW	0

位	字段	描述
[5]	CH1IMAE	通道 1 立即有效使能位 0: 无动作 1: 单脉冲立即有效模式使能 无论 CNTR 和 CH1CCR 值的比较结果如何, 在一个有效触发事件发生后, CH1OREF 会立即强制为比较匹配电平。 比较匹配电平将维持到下一个上溢或下溢事件的到来。 注: 只有当通道 1 被配置工作在 PWM 模式 1 或 PWM 模式 2 时, CH1IMAE 位可用。
[4]	CH1PRE	通道 1 捕捉 / 比较寄存器 (CH1CCR) 预载使能位 0: CH1CCR 预载功能除能 当 CH1PRE 位清零, CH1CCR 寄存器将立即被更新为新的值, 且立即可用。 1: CH1CCR 预载功能使能 直到更新事件发生后, 新的 CH1CCR 值才会被传送到影子寄存器中。

位	字段	描述
[8][2:0]	CH1OM[3:0]	<p>通道 1 输出模式设置 这些位定义了输出参考信号 CH1OREF 的功能类型</p> <p>0000: 无变化</p> <p>0001: 比较匹配时输出 0</p> <p>0010: 比较匹配时输出 1</p> <p>0011: 比较匹配时输出翻转</p> <p>0100: 强制无效 – CH1OREF 强制为 0</p> <p>0101: 强制有效 – CH1OREF 强制为 1</p> <p>0110: PWM 模式 1</p> <ul style="list-style-type: none"> – 在向上计数期间, 当 CNTR < CH1CCR, 通道 1 处于有效电平, 否则将处于无效电平。 – 在向下计数期间, 当 CNTR > CH1CCR, 通道 1 处于无效电平, 否则将处于有效电平。 <p>0111: PWM 模式 2</p> <ul style="list-style-type: none"> – 在向上计数期间, 当 CNTR < CH1CCR, 通道 1 处于无效电平, 否则将处于有效电平。 – 在向下计数期间, 当 CNTR > CH1CCR, 通道 1 处于有效电平, 否则将处于无效电平。 <p>1110: 非对称 PWM 模式 1</p> <ul style="list-style-type: none"> – 在向上计数期间, 当 CNTR < CH1CCR, 通道 1 处于有效电平, 否则将处于无效电平。 – 在向下计数期间, 当 CNTR > CH1ACR, 通道 1 处于无效电平, 否则将处于有效电平。 <p>1111: 非对称 PWM 模式 2</p> <ul style="list-style-type: none"> – 在向上计数期间, 当 CNTR < CH1CCR, 通道 1 处于无效电平, 否则将处于有效电平。 – 在向下计数期间, 当 CNTR > CH1ACR, 通道 1 处于有效电平, 否则将处于无效电平。 <p>注: 当通道 1 用作非对称 PWM 输出模式时, 计数器配置寄存器中的计数器模式选择位必须配置为中心对齐计数模式 (CMSEL = 0x1 / 0x2 / 0x3)。</p>

通道 2 输出配置寄存器 – CH2O CFR

该寄存器定义了通道 2 输出模式配置。

偏移量: 0x048

复位值: 0x0000_0000

31	30	29	28	27	26	25	24	保留位													
类型 / 复位																					
23	22	21	20	19	18	17	16	保留位													
类型 / 复位																					
15	14	13	12	11	10	9	8	保留位				CH2OM[3]									
类型 / 复位												RW	0								
7	6	5	4	3	2	1	0	保留位	CH2IMAE	CH2PRE	保留位	CH2OM[2:0]									
类型 / 复位												RW	0	RW	0	RW	0	RW	0	RW	0

位	字段	描述
[5]	CH2IMAE	通道 2 立即有效使能位 0: 无动作 1: 单脉冲立即有效模式使能 无论 CNTR 和 CH2CCR 值的比较结果如何, 在一个有效触发事件发生后, CH2OREF 会立即强制为比较匹配电平。 比较匹配电平将维持到下一个上溢或下溢事件的到来。 注: 只有当通道 2 被配置工作在 PWM 模式 1 或 PWM 模式 2 时, CH2IMAE 位可用。
[4]	CH2PRE	通道 2 捕捉 / 比较寄存器 (CH2CCR) 预载使能位 0: CH2CCR 预载功能除能 当 CH2PRE 位清零, CH2CCR 寄存器将立即被更新为新的值, 且立即可用。 1: CH2CCR 预载功能使能 直到更新事件发生后, 新的 CH2CCR 值才会被传送到影子寄存器中。

位	字段	描述
[8][2:0]	CH2OM[3:0]	<p>通道 2 输出模式设置 这些位定义了输出参考信号 CH2OREF 的功能类型</p> <p>0000: 无变化</p> <p>0001: 比较匹配时输出 0</p> <p>0010: 比较匹配时输出 1</p> <p>0011: 比较匹配时输出翻转</p> <p>0100: 强制无效 – CH2OREF 强制为 0</p> <p>0101: 强制有效 – CH2OREF 强制为 1</p> <p>0110: PWM 模式 1</p> <ul style="list-style-type: none"> – 在向上计数期间, 当 CNTR < CH2CCR, 通道 2 处于有效电平, 否则将处于无效电平。 – 在向下计数期间, 当 CNTR > CH2CCR, 通道 2 处于无效电平, 否则将处于有效电平。 <p>0111: PWM 模式 2</p> <ul style="list-style-type: none"> – 在向上计数期间, 当 CNTR < CH2CCR, 通道 2 处于无效电平, 否则将处于有效电平。 – 在向下计数期间, 当 CNTR > CH2CCR, 通道 2 处于有效电平, 否则将处于无效电平。 <p>1110: 非对称 PWM 模式 1</p> <ul style="list-style-type: none"> – 在向上计数期间, 当 CNTR < CH2CCR, 通道 2 处于有效电平, 否则将处于无效电平。 – 在向下计数期间, 当 CNTR > CH2CCR, 通道 2 处于无效电平, 否则将处于有效电平。 <p>1111: 非对称 PWM 模式 2</p> <ul style="list-style-type: none"> – 在向上计数期间, 当 CNTR < CH2CCR, 通道 2 处于无效电平, 否则将处于有效电平。 – 在向下计数期间, 当 CNTR > CH2CCR, 通道 2 处于有效电平, 否则将处于无效电平。 <p>注: 当通道 2 用作非对称 PWM 输出模式时, 计数器配置寄存器中的计数器模式选择位必须配置为中心对齐计数模式 (CMSEL = 0x1 / 0x2 / 0x3)。</p>

通道 3 输出配置寄存器 – CH3OCFR

该寄存器定义了通道 3 输出模式配置。

偏移量: 0x04C

复位值: 0x0000_0000

31	30	29	28	27	26	25	24	
保留位								
23	22	21	20	19	18	17	16	
保留位								
15	14	13	12	11	10	9	8	
保留位							CH3OM[3]	
							RW	0
7	6	5	4	3	2	1	0	
保留位		CH3IMAE	CH3PRE	保留位	CH3OM[2:0]			
		RW	0	RW	0	RW	0	
			0		0		0	

位	字段	描述
[5]	CH3IMAE	通道 3 立即有效使能位 0: 无动作 1: 单脉冲立即有效模式使能 无论 CNTR 和 CH3CCR 值的比较结果如何, 在一个有效触发事件发生后, CH3OREF 会立即强制为比较匹配电平。 比较匹配电平将维持到下一个上溢或下溢事件的到来。 注: 只有当通道 3 被配置工作在 PWM 模式 1 或 PWM 模式 2 时, CH3IMAE 位可用。
[4]	CH3PRE	通道 3 捕捉 / 比较寄存器 (CH3CCR) 预载使能位 0: CH3CCR 预载功能除能 当 CH3PRE 位清零, CH3CCR 寄存器将立即被更新为新的值, 且立即可用。 1: CH3CCR 预载功能使能 直到更新事件发生后, 新的 CH3CCR 值才会被传送到影子寄存器中。

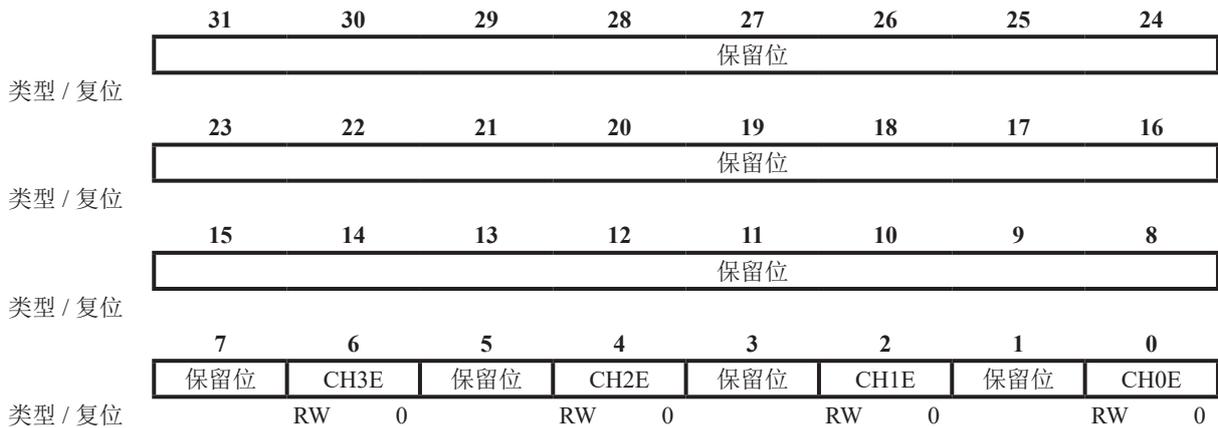
位	字段	描述
[8][2:0]	CH3OM[3:0]	<p>通道 3 输出模式设置 这些位定义了输出参考信号 CH3OREF 的功能类型</p> <p>0000: 无变化</p> <p>0001: 比较匹配时输出 0</p> <p>0010: 比较匹配时输出 1</p> <p>0011: 比较匹配时输出翻转</p> <p>0100: 强制无效 – CH3OREF 强制为 0</p> <p>0101: 强制有效 – CH3OREF 强制为 1</p> <p>0110: PWM 模式 1</p> <ul style="list-style-type: none"> – 在向上计数期间, 当 CNTR < CH3CCR, 通道 3 处于有效电平, 否则将处于无效电平。 – 在向下计数期间, 当 CNTR > CH3CCR, 通道 3 处于无效电平, 否则将处于有效电平。 <p>0111: PWM 模式 2</p> <ul style="list-style-type: none"> – 在向上计数期间, 当 CNTR < CH3CCR, 通道 3 处于无效电平, 否则将处于有效电平。 – 在向下计数期间, 当 CNTR > CH3CCR, 通道 3 处于有效电平, 否则将处于无效电平。 <p>1110: 非对称 PWM 模式 1</p> <ul style="list-style-type: none"> – 在向上计数期间, 当 CNTR < CH3CCR, 通道 3 处于有效电平, 否则将处于无效电平。 – 在向下计数期间, 当 CNTR > CH3CCR, 通道 3 处于无效电平, 否则将处于有效电平。 <p>1111: 非对称 PWM 模式 2</p> <ul style="list-style-type: none"> – 在向上计数期间, 当 CNTR < CH3CCR, 通道 3 处于无效电平, 否则将处于有效电平。 – 在向下计数期间, 当 CNTR > CH3CCR, 通道 3 处于有效电平, 否则将处于无效电平。 <p>注: 当通道 3 用作非对称 PWM 输出模式时, 计数器配置寄存器中的计数器模式选择位必须配置为中心对齐计数模式 (CMSEL = 0x1 / 0x2 / 0x3)。</p>

通道控制寄存器 – CHCTR

该寄存器包含了通道捕捉输入和比较输出功能使能控制位。

偏移量: 0x050

复位值: 0x0000_0000



位	字段	描述
[6]	CH3E	通道 3 捕捉 / 比较使能位 – 通道 3 被配置为输入 (CH3CCS = 0x1 / 0x2 / 0x3) 0: 输入捕捉模式除能 1: 输入捕捉模式使能 – 通道 3 被配置为输出 (CH3CCS = 0x0) 0: 关闭 – 通道 3 输出信号 CH3O 无作用 1: 开启 – 通道 3 输出信号 CH3O 在相应的输出脚产生
[4]	CH2E	通道 2 捕捉 / 比较使能位 – 通道 2 被配置为输入 (CH2CCS = 0x1 / 0x2 / 0x3) 0: 输入捕捉模式除能 1: 输入捕捉模式使能 – 通道 2 被配置为输出 (CH2CCS = 0x0) 0: 关闭 – 通道 2 输出信号 CH2O 无作用 1: 开启 – 通道 2 输出信号 CH2O 在相应的输出脚产生
[2]	CH1E	通道 1 捕捉 / 比较使能位 – 通道 1 被配置为输入 (CH1CCS = 0x1 / 0x2 / 0x3) 0: 输入捕捉模式除能 1: 输入捕捉模式使能 – 通道 1 被配置为输出 (CH1CCS = 0x0) 0: 关闭 – 通道 1 输出信号 CH1O 无作用 1: 开启 – 通道 1 输出信号 CH1O 在相应的输出脚产生
[0]	CH0E	通道 0 捕捉 / 比较使能位 – 通道 0 被配置为输入 (CH0CCS = 0x1 / 0x2 / 0x3) 0: 输入捕捉模式除能 1: 输入捕捉模式使能 – 通道 0 被配置为输出 (CH0CCS = 0x0) 0: 关闭 – 通道 0 输出信号 CH0O 无作用 1: 开启 – 通道 0 输出信号 CH0O 在相应的输出脚产生

通道极性配置寄存器 – CHPOLR

该寄存器包含了通道捕捉输入或比较输出极性的控制。

偏移量: 0x054

复位值: 0x0000_0000

31	30	29	28	27	26	25	24
保留位							
23	22	21	20	19	18	17	16
保留位							
15	14	13	12	11	10	9	8
保留位							
7	6	5	4	3	2	1	0
保留位	CH3P	保留位	CH2P	保留位	CH1P	保留位	CH0P
	RW 0		RW 0		RW 0		RW 0

位	字段	描述
[6]	CH3P	通道 3 捕捉 / 比较极性 – 通道 3 被配置为输入时 (CH3CCS = 0x1 / 0x2 / 0x3) 0: 捕捉事件发生在通道 3 上升沿 1: 捕捉事件发生在通道 3 下降沿 – 通道 3 被配置为输出时 (CH3CCS = 0x0) 0: 通道 3 输出高电平有效 1: 通道 3 输出低电平有效
[4]	CH2P	通道 2 捕捉 / 比较极性 – 通道 2 被配置为输入时 (CH2CCS = 0x1 / 0x2 / 0x3) 0: 捕捉事件发生在通道 2 上升沿 1: 捕捉事件发生在通道 2 下降沿 – 通道 2 被配置为输出时 (CH2CCS = 0x0) 0: 通道 2 输出高电平有效 1: 通道 2 输出低电平有效
[2]	CH1P	通道 1 捕捉 / 比较极性 – 通道 1 被配置为输入时 (CH1CCS = 0x1 / 0x2 / 0x3) 0: 捕捉事件发生在通道 1 上升沿 1: 捕捉事件发生在通道 1 下降沿 – 通道 1 被配置为输出时 (CH1CCS = 0x0) 0: 通道 1 输出高电平有效 1: 通道 1 输出低电平有效
[0]	CH0P	通道 0 捕捉 / 比较极性 – 通道 0 被配置为输入时 (CH0CCS = 0x1 / 0x2 / 0x0) 0: 捕捉事件发生在通道 0 上升沿 1: 捕捉事件发生在通道 0 下降沿 – 通道 0 被配置为输出时 (CH0CCS = 0x0) 0: 通道 0 输出高电平有效 1: 通道 0 输出低电平有效

定时器中断控制寄存器 – DICTR

该寄存器包含了定时器中断使能控制位。

偏移量: 0x074

复位值: 0x0000_0000

31	30	29	28	27	26	25	24		
保留位									
类型 / 复位									
23	22	21	20	19	18	17	16		
保留位									
类型 / 复位									
15	14	13	12	11	10	9	8		
保留位					TEVIE	保留位	UEVIE		
					RW	0	RW	0	
7	6	5	4	3	2	1	0		
保留位				CH3CCIE	CH2CCIE	CH1CCIE	CH0CCIE		
				RW	0	RW	0	RW	0
类型 / 复位									

位	字段	描述
[10]	TEVIE	触发事件中断使能位 0: 除能触发事件中断 1: 使能触发事件中断
[8]	UEVIE	更新事件中断使能位 0: 除能更新事件中断 1: 使能更新事件中断
[3]	CH3CCIE	通道 3 捕捉 / 比较中断使能位 0: 除能通道 3 中断 1: 使能通道 3 中断
[2]	CH2CCIE	通道 2 捕捉 / 比较中断使能位 0: 除能通道 2 中断 1: 使能通道 2 中断
[1]	CH1CCIE	通道 1 捕捉 / 比较中断使能位 0: 除能通道 1 中断 1: 使能通道 1 中断
[0]	CH0CCIE	通道 0 捕捉 / 比较中断使能位 0: 除能通道 0 中断 1: 使能通道 0 中断

定时器事件发生器寄存器 – EVGR

该寄存器包含了软件事件发生控制位。

偏移量: 0x078
复位值: 0x0000_0000

31	30	29	28	27	26	25	24
保留位							
23	22	21	20	19	18	17	16
保留位							
15	14	13	12	11	10	9	8
保留位				TEVG		保留位	
				WO 0		WO 0	
7	6	5	4	3	2	1	0
保留位				CH3CCG		CH0CCG	
				WO 0		WO 0	

位	字段	描述
[10]	TEVG	触发事件发生 触发事件 TEV 可由置位此位来发生。它由硬件自动清零。 0: 无动作 1: TEVIF 标志位置位
[8]	UEVG	更新事件发生 更新事件 UEV 可由置位此位来发生。它由硬件自动清零。 0: 无动作 1: 初始化计数器 依据当前定时器使用的计数模式，计数器的值返回到 0 或 CRR 预载值同时更新相关寄存器。详细描述请参考对应章节。
[3]	CH3CCG	通道 3 捕捉 / 比较发生 通道 3 捕捉 / 比较事件可由置位此位来发生。它由硬件自动清零。 0: 无动作 1: 通道 3 发生捕捉 / 比较事件 如果通道 3 被配置为输入，计数器的值将被捕捉到 CH3CCR 寄存器，接着 CH3CCIF 位被置位。如果通道 3 被配置为输出，则 CH3CCIF 被置位。
[2]	CH2CCG	通道 2 捕捉 / 比较发生 通道 2 捕捉 / 比较事件可由置位此位来发生。它由硬件自动清零。 0: 无动作 1: 通道 2 发生捕捉 / 比较事件 如果通道 2 被配置为输入，计数器的值将被捕捉到 CH2CCR 寄存器，接着 CH2CCIF 位被置位。如果通道 2 被配置为输出，则 CH2CCIF 被置位。
[1]	CH1CCG	通道 1 捕捉 / 比较发生 通道 1 捕捉 / 比较事件可由置位此位来发生。它由硬件自动清零。 0: 无动作 1: 通道 1 发生捕捉 / 比较事件 如果通道 1 被配置为输入，计数器的值将被捕捉到 CH1CCR 寄存器，接着 CH1CCIF 位被置位。如果通道 1 被配置为输出，则 CH1CCIF 被置位。

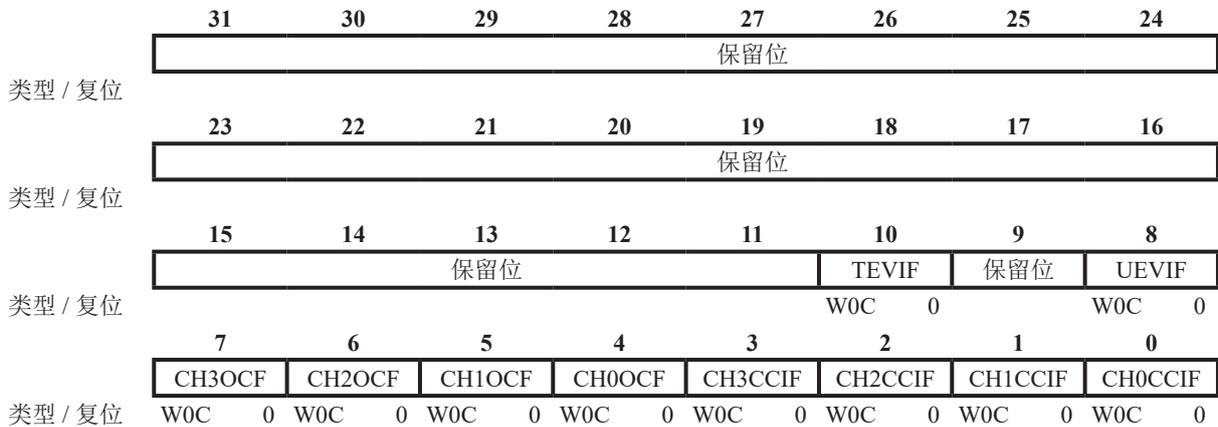
位	字段	描述
[0]	CH0CCG	通道 0 捕捉 / 比较发生 通道 0 捕捉 / 比较事件可由置位此位来发生。它由硬件自动清零。 0: 无动作 1: 通道 0 发生捕捉 / 比较事件 如果通道 0 被配置为输入, 计数器的值将被捕捉到 CH0CCR 寄存器, 接着 CH0CCIF 位被置位。如果通道 0 被配置为输出, 则 CH0CCIF 被置位。

定时器中断状态寄存器 – INTSR

该寄存器存储了定时器中断的状态。

偏移量: 0x07C

复位值: 0x0000_0000



位	字段	描述
[10]	TEVIF	触发事件中断标志位 此位在触发事件发生时通过硬件置位, 由软件清零。 0: 无触发事件发生 1: 触发事件发生
[8]	UEVIF	更新事件中断标志位 此位在更新事件发生时通过硬件置位, 由软件清零。 0: 无更新事件发生 1: 更新事件发生 注: 更新事件在以下情况下发生: - 计数器上溢或下溢 - UEVG 位被置位 - 来自从机触发输入的重启触发事件发生
[7]	CH3OCF	通道 3 过度捕捉标志位 此位由硬件置位, 由软件清零。 0: 没有检测到过度捕捉事件 1: 当 CH3CCIF 位已被置位且还没有被软件清零时, 捕捉事件再次发生。
[6]	CH2OCF	通道 2 过度捕捉标志位 此位由硬件置位, 由软件清零。 0: 没有检测到过度捕捉事件 1: 当 CH2CCIF 位已被置位且还没有被软件清零时, 捕捉事件再次发生。

位	字段	描述
[5]	CH1OCF	通道 1 过度捕捉标志位 此位由硬件置位，由软件清零。 0: 没有检测到过度捕捉事件 1: 当 CH1CCIF 位已被置位且还没有被软件清零时，捕捉事件再次发生。
[4]	CH0OCF	通道 0 过度捕捉标志位 此位由硬件置位，由软件清零。 0: 没有检测到过度捕捉事件 1: 当 CH0CCIF 位已被置位且还没有被软件清零时，捕捉事件再次发生。
[3]	CH3CCIF	通道 3 捕捉 / 比较中断标志位 - 通道 3 配置为输出时: 0: 无匹配事件发生 1: 计数器 CNTR 内容与 CH3CCR 寄存器内容匹配 除中心对齐计数模式外，当计数器的值与 CH3CCR 的值匹配时，此位被硬件置位。 通过软件清零。 - 通道 3 配置为输入时: 0: 无输入捕捉发生 1: 输入捕捉发生 此位在捕捉事件发生时被硬件置位。通过软件或通过读取 CH3CCR 寄存器清零。
[2]	CH2CCIF	通道 2 捕捉 / 比较中断标志位 - 通道 2 配置为输出时: 0: 无匹配事件发生 1: 计数器 CNTR 内容与 CH2CCR 寄存器内容匹配 除中心对齐计数模式外，当计数器的值与 CH2CCR 的值匹配时，此位被硬件置位。 通过软件清零。 - 通道 2 配置为输入时: 0: 无输入捕捉发生 1: 输入捕捉发生 此位在捕捉事件发生时被硬件置位。通过软件或通过读取 CH2CCR 寄存器清零。
[1]	CH1CCIF	通道 1 捕捉 / 比较中断标志位 - 通道 1 配置为输出时: 0: 无匹配事件发生 1: 计数器 CNTR 内容与 CH1CCR 寄存器内容匹配 除中心对齐计数模式外，当计数器的值与 CH1CCR 的值匹配时，此位被硬件置位。 通过软件清零。 - 通道 1 配置为输入时: 0: 无输入捕捉发生 1: 输入捕捉发生 此位在捕捉事件发生时被硬件置位。通过软件或通过读取 CH1CCR 寄存器清零。
[0]	CH0CCIF	通道 0 捕捉 / 比较中断标志位 - 通道 0 配置为输出时: 0: 无匹配事件发生 1: 计数器 CNTR 内容与 CH0CCR 寄存器内容匹配 除中心对齐计数模式外，当计数器的值与 CH0CCR 的值匹配时，此位被硬件置位。 通过软件清零。 - 通道 0 配置为输入时: 0: 无输入捕捉发生 1: 输入捕捉发生 此位在捕捉事件发生时被硬件置位。通过软件或通过读取 CH0CCR 寄存器清零。

定时器计数器寄存器 – CNTR

该寄存器存储了定时器计数器的值。

偏移量: 0x080

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位								
	15	14	13	12	11	10	9	8	
类型 / 复位	CNTV								
	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
类型 / 复位	CNTV								
	RW	0	RW	0	RW	0	RW	0	RW

位	字段	描述
[15:0]	CNTV	计数器的值

定时器预分频器寄存器 – PSCR

该寄存器定义了定时器预分频器的值以产生计数器时钟。

偏移量: 0x084

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24								
类型 / 复位	保留位															
	23	22	21	20	19	18	17	16								
类型 / 复位	保留位															
	15	14	13	12	11	10	9	8								
类型 / 复位	PSCV															
	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0								
类型 / 复位	PSCV															
	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0

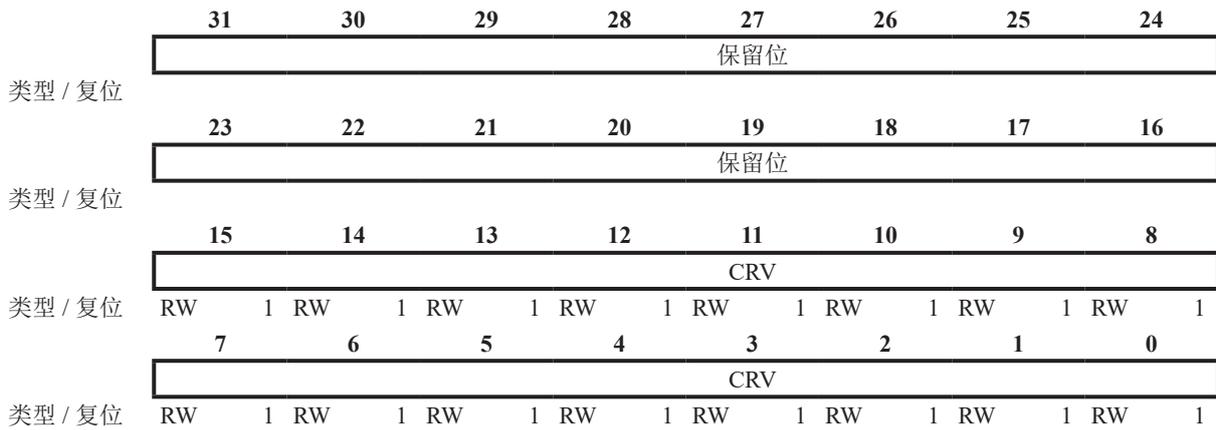
位	字段	描述
[15:0]	PSCV	<p>预分频器的值</p> <p>这些位用来定义预分频器的值以产生计数器的时钟频率 f_{CK_CNT}。</p> $f_{CK_CNT} = \frac{f_{CK_PSC}}{PSCV[15:0] + 1}$ <p>f_{CK_PSC} 代表预分频器时钟源。</p>

定时器计数器重载寄存器 – CRR

该寄存器定义了定时器计数器重载值。

偏移量: 0x088

复位值: 0x0000_FFFF



位	字段	描述
[15:0]	CRV	计数器重载值 CRV 定义计数器重载值，被加载到实际计数器寄存器中。

通道 0 捕捉 / 比较寄存器 – CH0CCR

该寄存器定义了定时器通道 0 捕捉 / 比较值。

偏移量: 0x090

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位								
	15	14	13	12	11	10	9	8	
类型 / 复位	CH0CCV								
	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
类型 / 复位	CH0CCV								
	RW	0	RW	0	RW	0	RW	0	RW

位	字段	描述
[15:0]	CH0CCV	通道 0 捕捉 / 比较值 – 当通道 0 配置为输出时 CH0CCR 值和计数器的值相比较，比较结果用于触发 CH0OREF 输出信号。 – 当通道 0 配置为输入时 CH0CCR 寄存器存储由最新一次通道 0 捕捉事件捕捉到的计数器值。

通道 1 捕捉 / 比较寄存器 – CH1CCR

该寄存器定义了定时器通道 1 捕捉 / 比较值。

偏移量: 0x094

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位								
	15	14	13	12	11	10	9	8	
类型 / 复位	CH1CCV								
	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
类型 / 复位	CH1CCV								
	RW	0	RW	0	RW	0	RW	0	RW

位	字段	描述
[15:0]	CH1CCV	通道 1 捕捉 / 比较值 – 当通道 1 配置为输出时 CH1CCR 值和计数器的值相比较，比较结果用于触发 CH1OREF 输出信号。 – 当通道 1 配置为输入时 CH1CCR 寄存器存储由最新一次通道 1 捕捉事件捕捉到的计数器值。

通道 2 捕捉 / 比较寄存器 – CH2CCR

该寄存器定义了定时器通道 2 捕捉 / 比较值。

偏移量: 0x098

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位								
	15	14	13	12	11	10	9	8	
类型 / 复位	CH2CCV								
	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
类型 / 复位	CH2CCR								
	RW	0	RW	0	RW	0	RW	0	RW

位	字段	描述
[15:0]	CH2CCV	通道 2 捕捉 / 比较值 – 当通道 2 配置为输出时 CH2CCR 值和计数器的值相比较, 比较结果用于触发 CH2OREF 输出信号。 – 当通道 2 配置为输入时 CH2CCR 寄存器存储由最新一次通道 2 捕捉事件捕捉到的计数器值。

通道 3 捕捉 / 比较寄存器 – CH3CCR

该寄存器定义了定时器通道 3 捕捉 / 比较值。

偏移量: 0x09C

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位								
	15	14	13	12	11	10	9	8	
类型 / 复位	CH3CCV								
	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
类型 / 复位	CH3CCV								
	RW	0	RW	0	RW	0	RW	0	RW

位	字段	描述
[15:0]	CH3CCV	通道 3 捕捉 / 比较值 – 当通道 3 配置为输出时 CH3CCR 值和计数器的值相比较, 比较结果用于触发 CH3OREF 输出信号。 – 当通道 3 配置为输入时 CH3CCR 寄存器存储由最新一次通道 3 捕捉事件捕捉到的计数器值。

通道 0 非对称比较寄存器 – CH0ACR

该寄存器定义了定时器通道 0 非对称比较值。

偏移量: 0x0A0

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位								
	15	14	13	12	11	10	9	8	
类型 / 复位	CH0ACV								
	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
类型 / 复位	CH0ACV								
	RW	0	RW	0	RW	0	RW	0	RW

位	字段	描述
[15:0]	CH0ACV	通道 0 非对称比较值 当通道 0 被配置为非对称 PWM 模式且计数器向下计数时，写入该寄存器的值将与计数器的值进行比较。

通道 1 非对称比较寄存器 – CH1ACR

该寄存器定义了定时器通道 1 非对称比较值。

偏移量: 0x0A4

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位								
	15	14	13	12	11	10	9	8	
类型 / 复位	CH1ACV								
	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
类型 / 复位	CH1ACV								
	RW	0	RW	0	RW	0	RW	0	RW

位	字段	描述
[15:0]	CH1ACV	通道 1 非对称比较值 当通道 1 被配置为非对称 PWM 模式且计数器向下计数时，写入该寄存器的值将与计数器的值进行比较。

通道 2 非对称比较寄存器 – CH2ACR

该寄存器定义了定时器通道 2 非对称比较值。

偏移量: 0x0A8

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位								
	15	14	13	12	11	10	9	8	
类型 / 复位	CH2ACV								
	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
类型 / 复位	CH2ACV								
	RW	0	RW	0	RW	0	RW	0	RW

位	字段	描述
[15:0]	CH2ACV	通道 2 非对称比较值 当通道 2 被配置为非对称 PWM 模式且计数器向下计数时，写入该寄存器的值将与计数器的值进行比较。

通道 3 非对称比较寄存器 – CH3ACR

该寄存器定义了定时器通道 3 非对称比较值。

偏移量: 0x0AC

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位								
	15	14	13	12	11	10	9	8	
类型 / 复位	CH3ACV								
	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
类型 / 复位	CH3ACV								
	RW	0	RW	0	RW	0	RW	0	RW

位	字段	描述
[15:0]	CH3ACV	通道 3 非对称比较值 当通道 3 被配置为非对称 PWM 模式且计数器向下计数时，写入该寄存器的值将与计数器的值进行比较。

14 脉冲宽度调制器 (PWM)

简介

脉冲宽度调制器由一个 16-bit 向上 / 向下计数器、四个 16-bit 比较寄存器 (CR)、一个 16-bit 计数器重载寄存器 (CRR) 和几个控制 / 状态寄存器组成。脉冲宽度调制器可用于多种用途，包括通用计时和产生输出波形，如单脉冲或 PWM 输出。

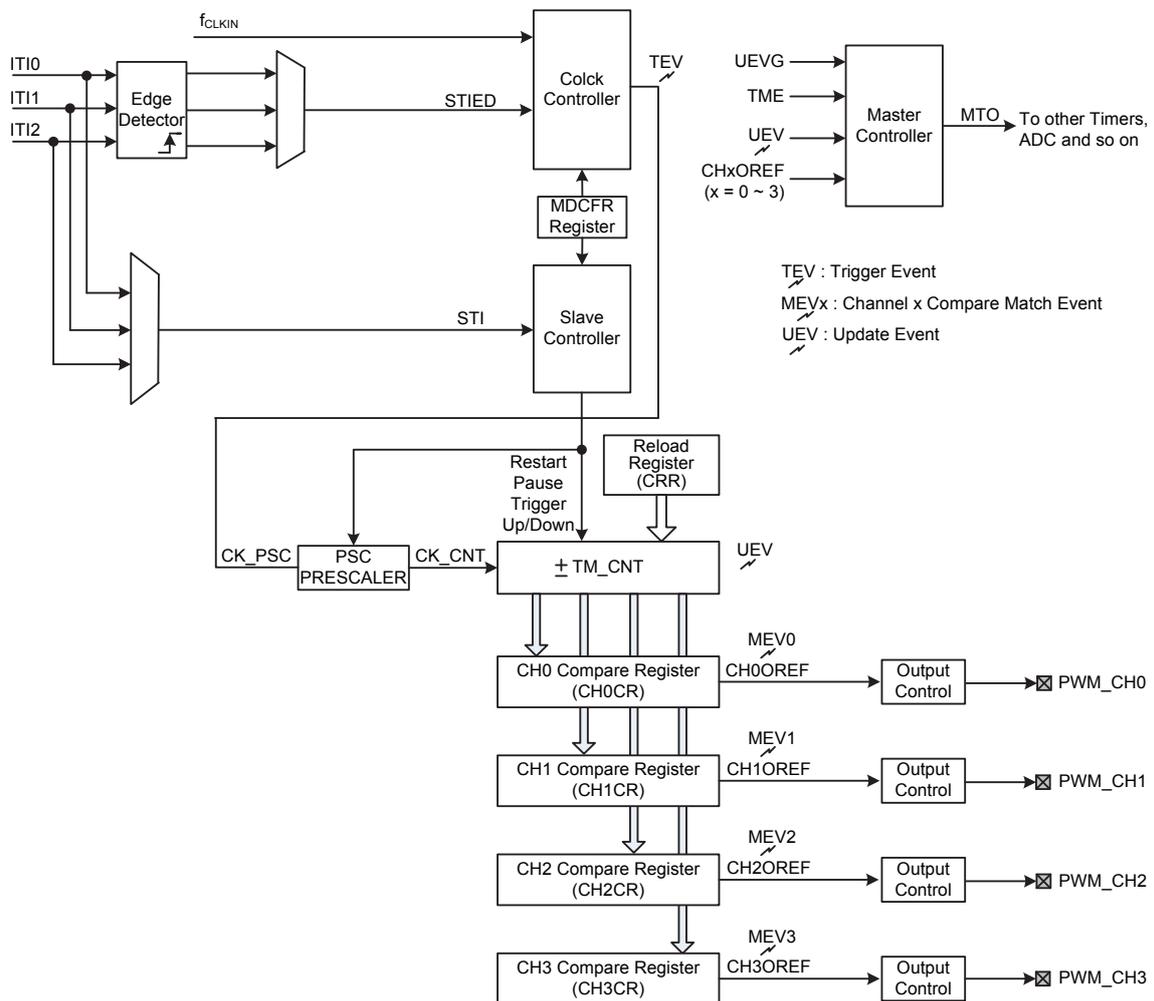


图 63. PWM 方框图

特性

- 16-bit 向上 / 向下自动重载计数器
- 16-bit 可编程预分频器，可以对计数器时钟进行 1 ~ 65536 之间的任意数值的分频
- 多达 4 个独立通道用于：
 - 比较匹配输出
 - PWM 波形生成 – 边沿对齐计数和中心对齐计数模式
 - 单脉冲模式输出
- 使用外部信号控制定时器和定时器互联的同步电路
- 下列事件发生时将产生中断：
 - 更新事件
 - 触发事件
 - 比较匹配输出事件
- PWM 主机 / 从机模式控制器

功能描述

计数器模式

向上计数

在此模式下，计数器从 0 连续向上计数，一直计数到 CRR 寄存器定义的计数器的重载值。一旦计数器的值达到了计数器重载值，定时器模块将发生溢出事件，并从 0 开始重新计数。这一动作会反复执行。向上计数模式时，CNTCFR 寄存器中的计数方向位 DIR 应置为 0。

当通过设置 EVGR 寄存器中的 UEVG 位为 1 触发更新事件时，计数器的值将被初始化为 0。

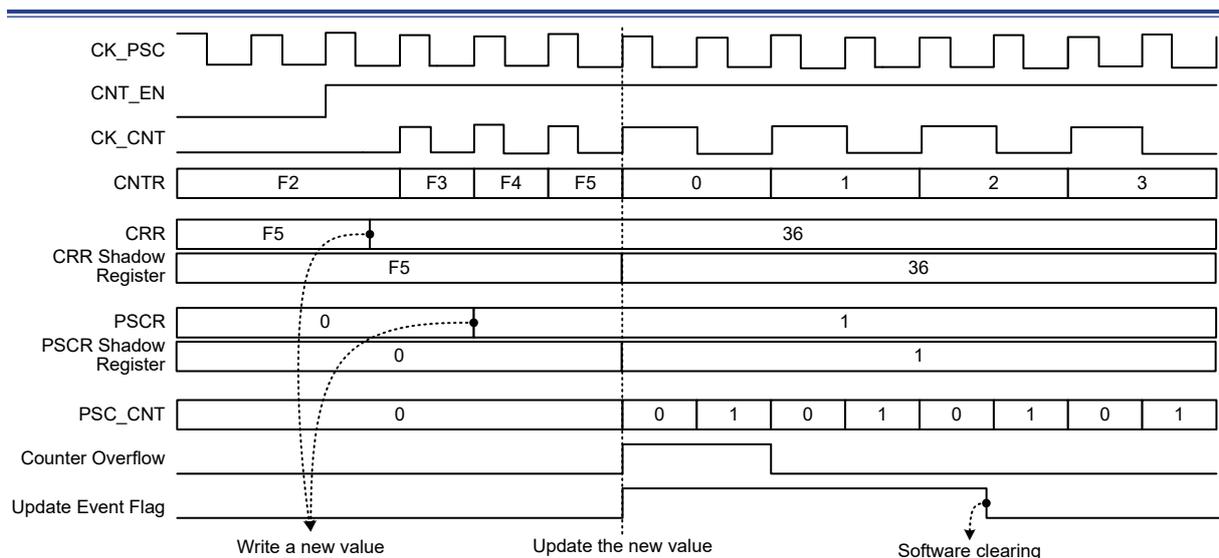


图 64. 向上计数范例

向下计数

在此模式下，计数器将从 CRR 寄存器定义的计数器重载值开始向下计数，一直计数到 0。一旦计数器的值达到 0，定时器模块将产生一个下溢事件，并从计数器重载值开始重新计数。这一动作会反复执行。向下计数模式时，CNTCFR 寄存器中的计数方向位 DIR 应置为 1。

当通过设置 EVGR 寄存器中的 UEVG 位为 1 触发更新事件时，计数器的值将被初始化为计数器重载值。

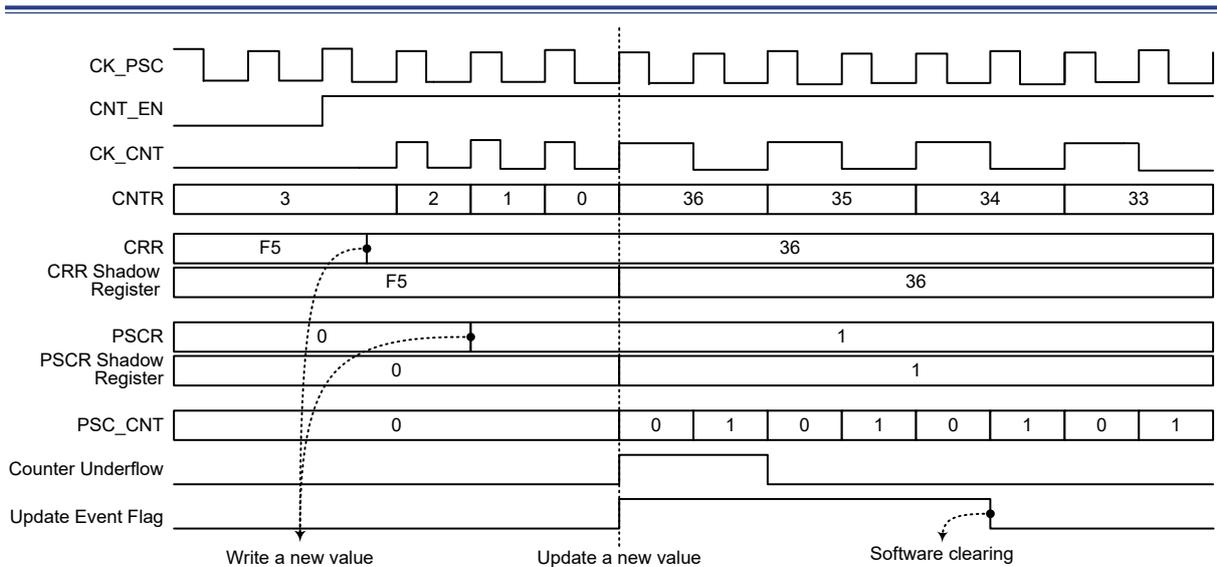


图 65. 向下计数范例

中心对齐计数

在中心对齐计数模式下，计数器将交替地从 0 计数到重载值而后从重载值计数到 0。在向上计数模式下，当计数器计数到计数器重载值时，计数器模块会产生一个上溢事件；而在向下计数模式下，当计数器计数到 0 时，计数器模块会产生一个下溢事件。CNTCFR 寄存器中的 DIR 位是只读位，其值表明了中心对齐计数模式下的计数方向。计数方向由硬件自动更新。

在中心对齐计数模式下，如果将 EVGR 寄存器中的 UEVG 位置位，那么不管计数器正在向上计数还是向下计数，计数器的值都将初始化为 0。

可经由 CNTCFR 寄存器中的 CMSEL 字段来控制当发生上溢事件 (CMSEL = 0x1)、下溢事件 (CMSEL = 0x2) 或两个事件都发生 (CMSEL = 0x3) 时，INTSR 寄存器中的 UEVIF 位将被置为 1。

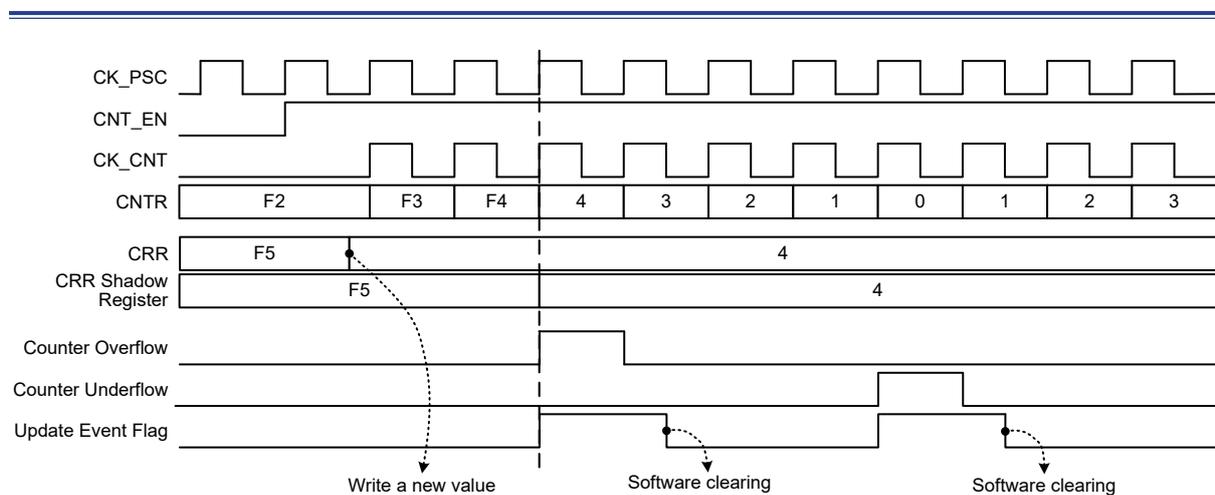


图 66. 中心对齐计数范例

时钟控制器

以下描述了定时器模块的时钟控制器，它用来选择内部预分频计数器的时钟源。

■ 内部 APB 时钟 f_{CLKIN} :

默认内部时钟源是 APB 时钟 f_{CLKIN} ，用来驱动计数器预分频器。

■ STIED:

计数器预分频器在每一个 STI 信号的上升沿期间计数。此模式可通过将 MDCFR 寄存器中的 SMSEL 字段设为 0x7 来选择。这时计数器将作为一个事件计数器使用。输入事件，即 STI，可通过把 TRSEL 字段设成除 0x0 以外的可用值来选择。当 STI 信号被选择作为时钟源使用时，在每一个 STI 信号上升沿时，内部边沿检测电路将会产生一个时钟脉冲来驱动计数器预分频器。值得注意的是，如果 TRSEL 字段设成 0x0 来选择软件 UEVG 位作为触发源，那么当 SMSEL 字段设成 0x7 时，计数器将会被更新而非计数。

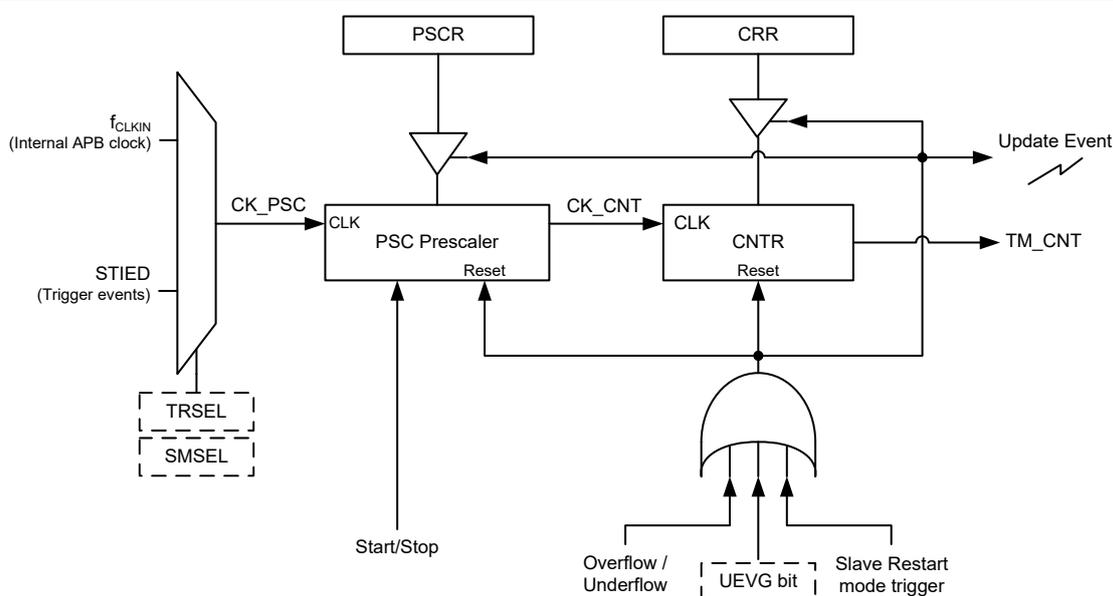


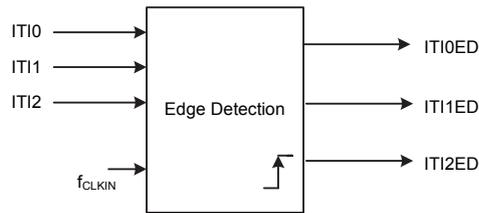
图 67. PWM 时钟源选择

触发控制器

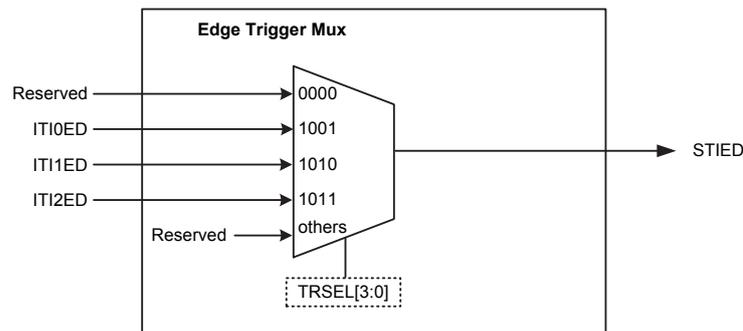
触发控制器用来选择触发源以及设置电平触发或边沿触发条件。内部触发输入可通过 TRCFR 寄存器中的触发选择位 TRSEL 进行选择。除了 UEVG 位软件触发之外的所有触发源，内部边沿检测电路将会在每个触发信号上升沿作用期间产生一个时钟脉冲，以激活某些因触发信号上升沿而触发的 PWM 功能。

Trigger Controller Block = Edge Trigger Mux + Level Trigger Mux

Internal Trigger Input



Edge Trigger = Internal (ITIx)



Level Trigger Source = Internal (ITIx) + Software UEVG bit

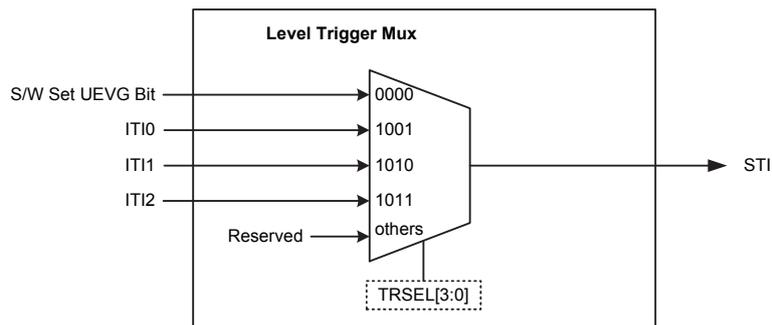


图 68. 触发控制器方框图

从机控制器

在几种模式下, PWM 可以与一个外部触发器进行同步。这些模式包括重启模式、暂停模式和触发模式, 是通过 MDCFR 寄存器中的 SMSEL 字段选择的。这些模式的触发输入来自于 STI 信号, 通过 TRCFR 寄存器中的 TRSEL 字段选择。从机控制器中的工作模式在相关章节中有所描述。

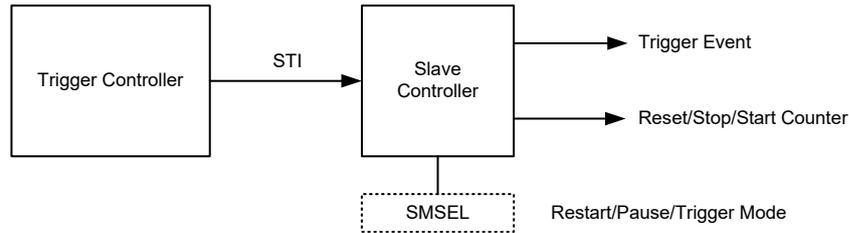


图 69. 从机控制器方框图

重启模式

作为对 STI 信号上升沿的响应, 计数器及其预分频器会被重新初始化。当一个 STI 上升沿到来时, 更新事件软件产生位 UEVG 将被硬件自动置位, 触发事件标志位也将被置位, 计数器和预分频器将被重新初始化。虽然 UEVG 位被硬件置 1, 但是更新事件还没有真正发生, 它取决于更新事件除能控制位 UEVDIS 是否被置 1。如果 UEVDIS 位被置 1 来除能更新事件, 那么更新事件将不会发生, 然而当 STI 上升沿到来时, 计数器和预分频器仍会被重新初始化。如果 CNTCFR 寄存器中的 UEVDIS 位被清零来使能更新事件, 则更新事件将伴随 STI 上升沿一起发生, 所有预加载的寄存器将被更新。

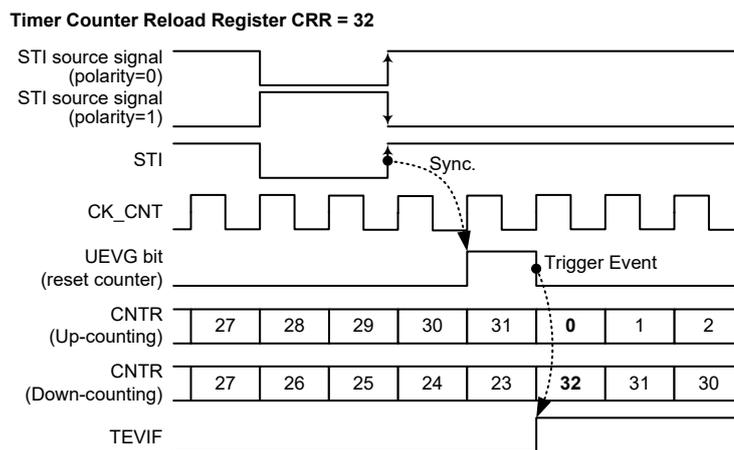


图 70. 重启模式下的 PWM

暂停模式

在暂停模式下，所选择的 STI 输入信号电平用来控制计数器的开始 / 停止操作。当 STI 信号处于高电平时，计数器开始计数；当 STI 信号转换为低电平时，计数器停止计数并保持当前值不变，且不会被复位。计数器的开始 / 停止操作是由 STI 电平控制。

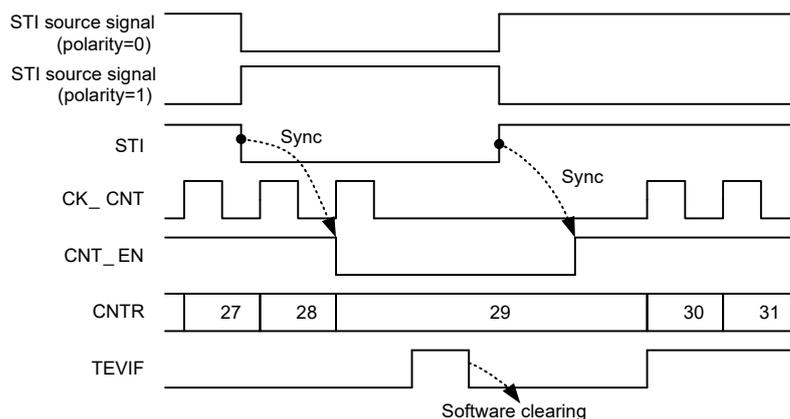


图 71. 暂停模式下的 PWM

触发模式

在计数器停止计数后，当一个 STI 上升沿信号到来时，计数器将从当前值继续开始计数。注意，如果 STI 信号来自于 UEVG 位软件触发，计数器不会重新计数。当 STI 源信号通过 UEVG 位选择为软件触发时，不会产生使计数器重新计数的时钟脉冲。还要注意，STI 信号只是用来使计数器继续计数，而没有使计数器停止计数的作用。

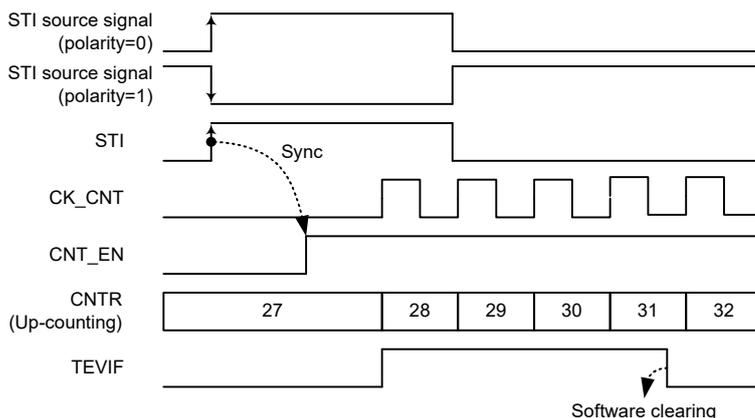


图 72. 触发模式下的 PWM

主机控制器

PWM 和 TM 可在内部连接在一起用于定时器同步或链接。当一个 PWM 被配置在主机模式下时, PWM 主机控制器将会产生一个主机触发输出 (MTO) 信号, 通过 MDCFR 寄存器中的 MMSEL 字段选择可以触发或驱动被设置在从机模式下的另一个 PWM 或 TM 的定时器复位、启动、停止或为其提供时钟源。

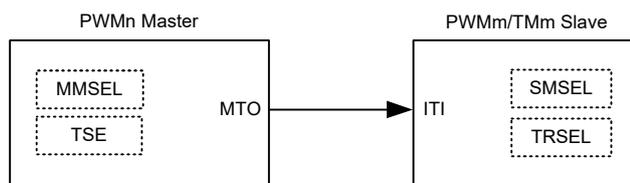


图 73. 主机 PWMn 和从机 PWMm / TMm 相连接

MDCFR 寄存器中的主机模式字段 (MMSEL) 用来选择同步另外一个从机 PWM 或 TM 的 MTO 源。

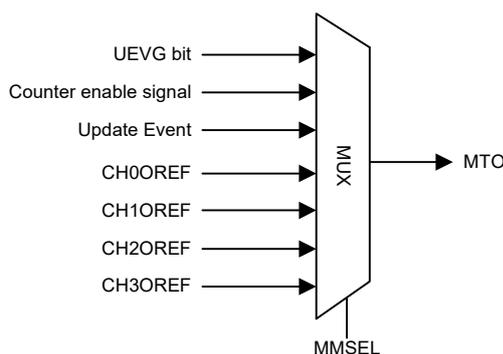


图 74. MTO 选项

例如, 把 MMSEL 字段设为 0x5, 以选择 CH1OREF 信号作为 MTO 信号来同步另外一个从机 PWM 或 TM。欲知更多详细描述, 请参考相关的 MDCFR 寄存器中的 MMSEL 字段定义。

通道控制器

PWM 有四个独立的通道，用来选择作为比较匹配输出。每个比较匹配输出通道都由一个预载寄存器和一个影子寄存器组成。APB 总线只能通过读 / 写预载寄存器来进行数据访问的。

当工作在比较匹配输出模式，CHxCR 预载寄存器的内容会被复制到相应的影子寄存器中，然后计数器的值会与寄存器的值进行比较。

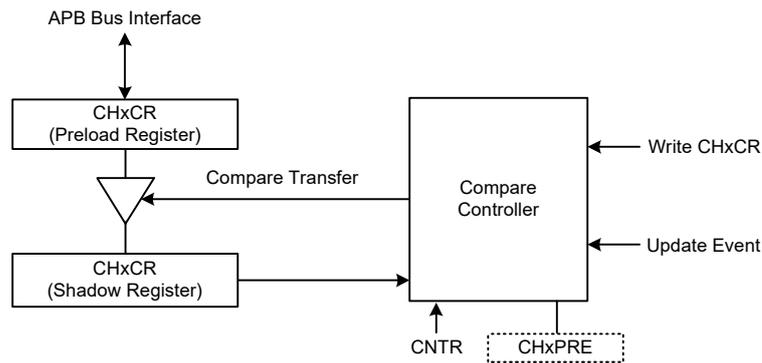


图 75. 比较方框图

输出级

PWM 有四个通道，用于比较匹配、单脉冲或 PWM 输出功能。通道输出 PWM_CHx 由 CHxOCFR、CHPOLR 和 CHCTR 寄存器分别对应的 CHxOM、CHxP 和 CHxE 位控制。

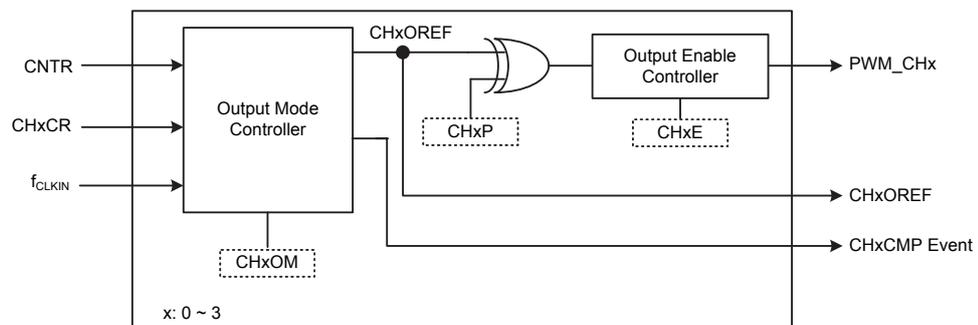


图 76. 输出级方框图

通道输出参考信号

当 PWM 用在比较匹配输出模式时, CHxOREF 信号 (通道 x 输出参考信号) 通过设置 CHxOM 位来定义。当计数器的值与 CHxCR 寄存器的内容匹配时, CHxOREF 信号有几种输出功能类型, 这些类型包括 CHxOREF 输出可为低电平, 高电平或者翻转, 除此之外, 也有 PWM 模式 1 和 PWM 模式 2 输出。在这些模式中, CHxOREF 信号的电平都是根据计数方向以及计数器值与 CHxCR 内容的关系而改变。还有两种模式, 不论计数器和 CHxCR 的值是什么, 输出都会被强制为一个无效或有效的电平。至于更详细的说明请参考相应位的定义。输出类型设置如表 30 所示。

表 30. 比较匹配输出设置

CHxOM 值	比较匹配输出电平
0x0	无变化
0x1	输出 0
0x2	输出 1
0x3	输出翻转
0x4	强制无效电平
0x5	强制有效电平
0x6	PWM 模式 1
0x7	PWM 模式 2

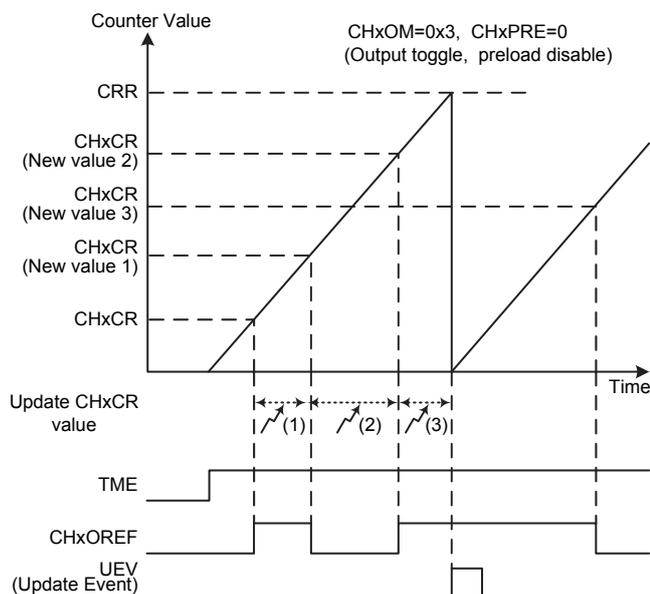


图 77. 翻转模式通道输出参考信号 (CHxPRE = 0)

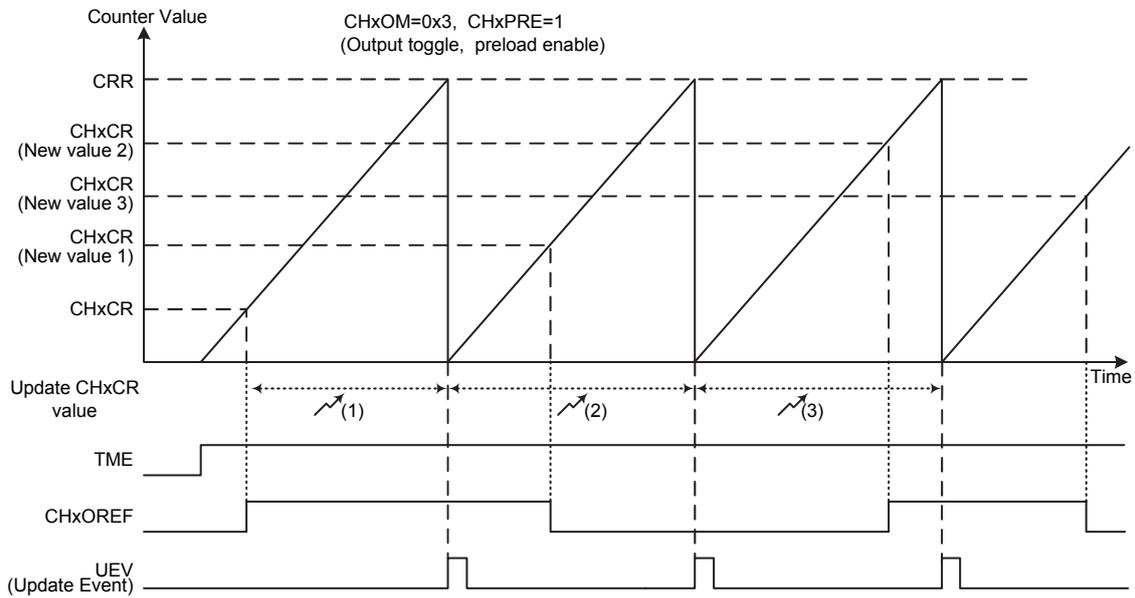


图 78. 翻转模式通道输出参考信号 (CHxPRE = 1)

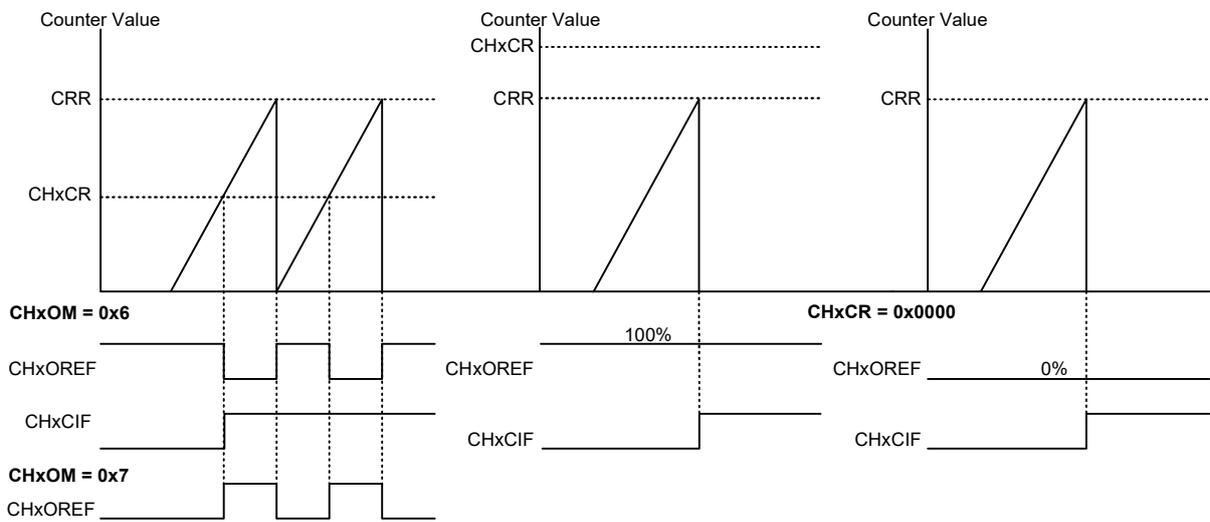


图 79. PWM 模式通道输出参考信号和计数器向上计数模式

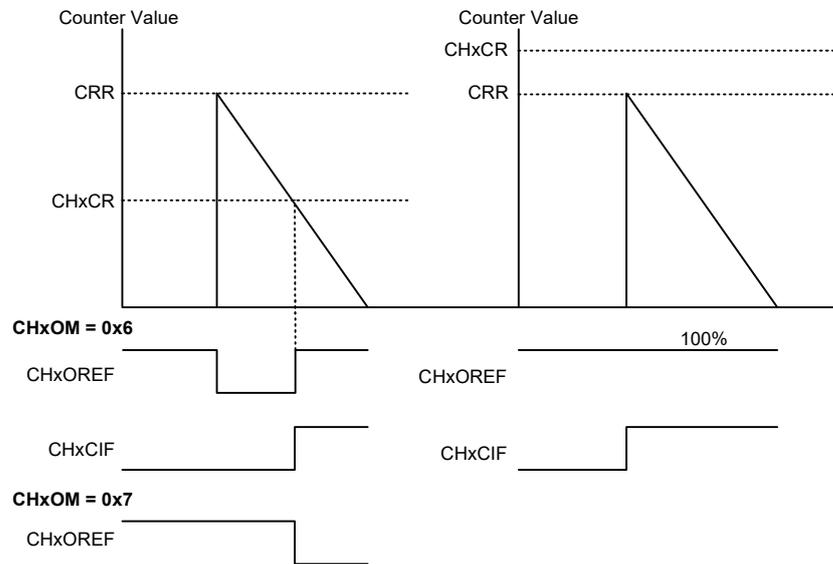


图 80. PWM 模式通道输出参考信号和计数器向下计数模式

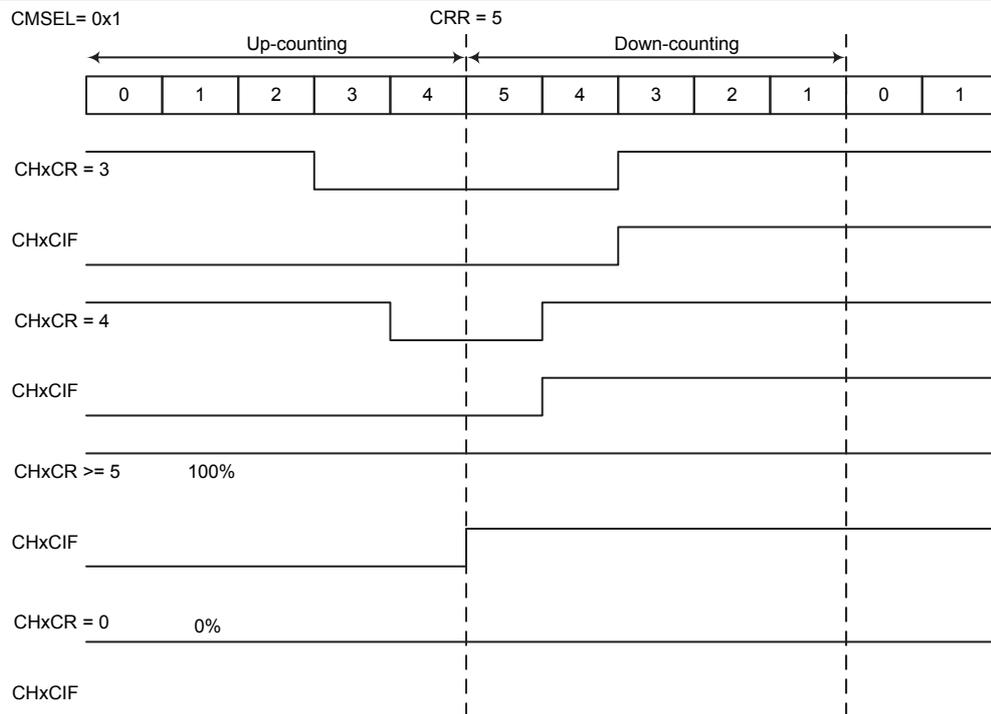


图 81. PWM 模式通道输出参考信号和计数器中心对齐计数模式

更新管理

更新事件用来把实际寄存器 CRR、PSCR、CHxACR 和 CHxCR 中的值更新到相应的影子寄存器中。更新事件可在计数器上溢 / 下溢、软件更新控制位被触发或从机控制器的更新事件产生时发生。

更新事件是否发生由 CNTCFR 寄存器的 UEVDIS 位控制。通过设置 CNTCFR 寄存器中的 UGDIS 位, 当更新事件发生, 可产生相应的更新事件中断。欲知更多详细信息, 请参考 CNTCFR 寄存器中的 UEVDIS 位和 UGDIS 位的相关定义。

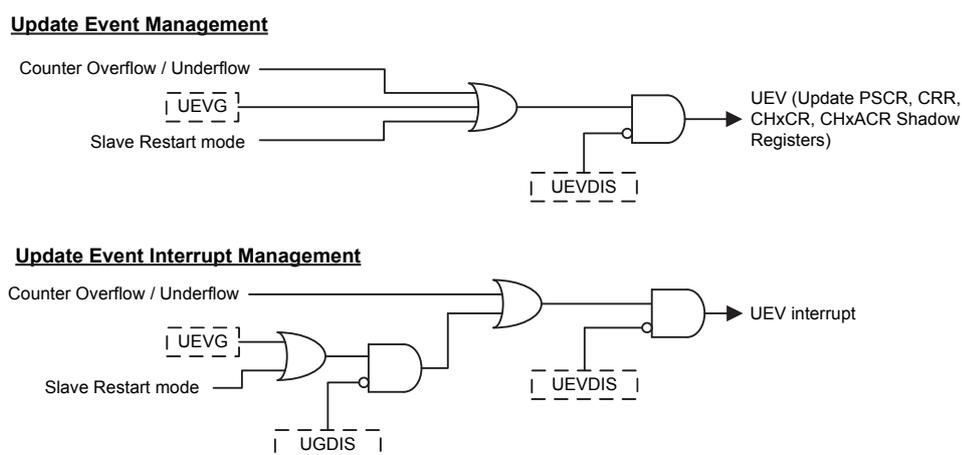


图 82. 更新事件设置方框图

单脉冲模式

一旦定时器被设置工作在单脉冲模式下, 则无需把 CTR 寄存器中的定时器使能位 TME 置为 1 来使能定时器。当 STI 信号上升沿发生时或通过软件把 TME 位设置为 1 时, 触发器将会产生一个脉冲, 然后 TME 位一直保持为高电平直到更新事件发生或使用软件将 TME 清零。如果使用软件将 TME 位清零, 计数器将停止且保持当前值不变。如果 TME 位是由硬件的更新事件自动清零, 计数器将被重新初始化。

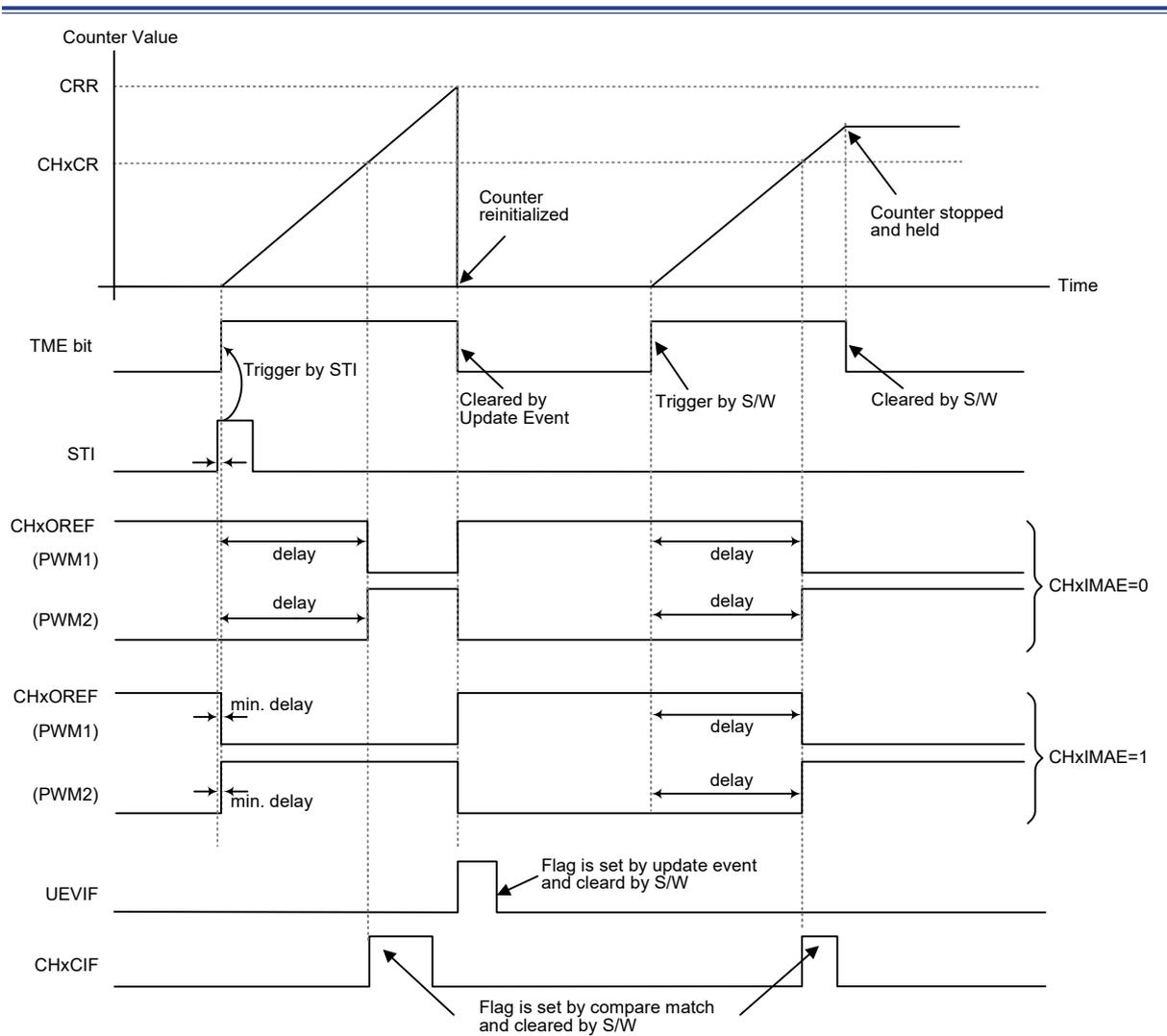


图 83. 单脉冲模式

在单脉冲模式下，STI 有效边沿使 TME 置为 1 时，将使能计数器。然而，由于要执行计数器值和 CHxCR 值的比较结果，会存在几个时钟的延迟。用户可以通过设置 CHxOCFR 寄存器中的 CHxIMAE 位来使延迟时间减小。单脉冲模式下，STI 上升沿触发发生后，CHxOREF 信号将立即被强制转变为与比较匹配事件发生时相同的电平，而无需考虑比较结果为何。只有当输出通道被配置工作在 PWM 模式 1 或 PWM 模式 2 下且触发源自于 STI 信号时，CHxIMAE 位才可用。

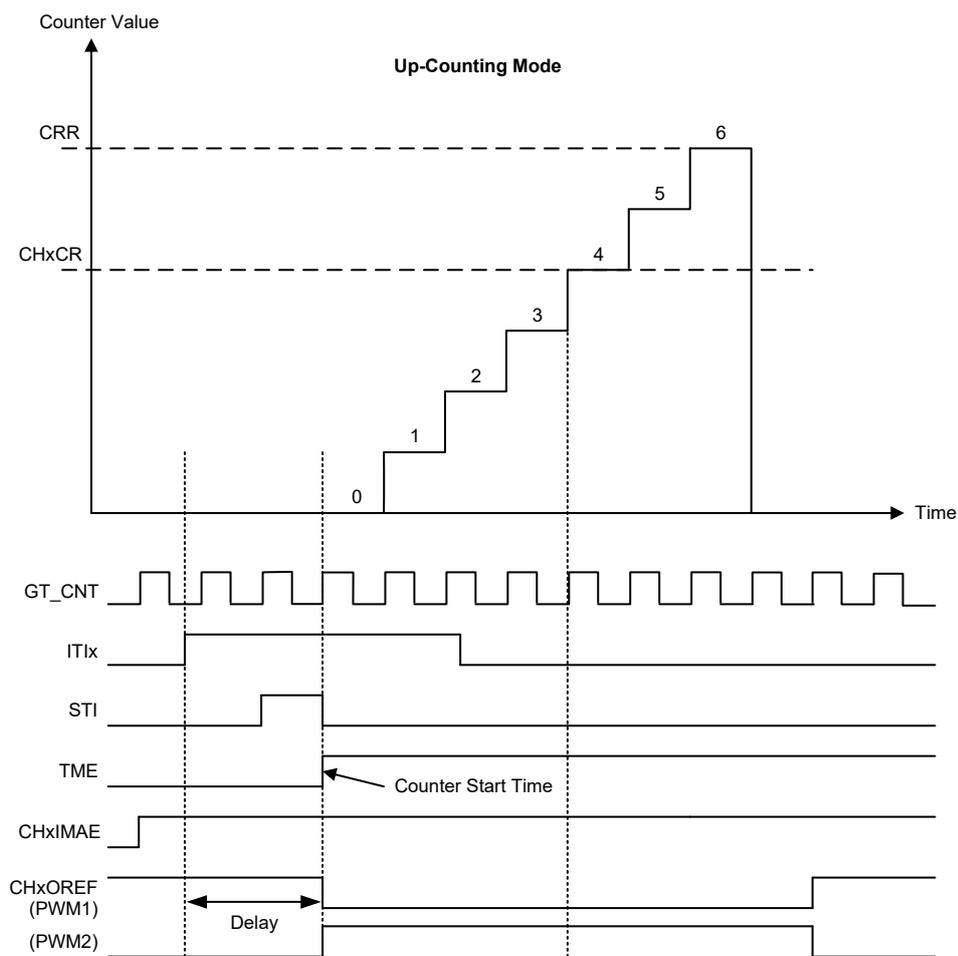


图 84. 立即有效模式的延迟

非对称 PWM 模式

非对称 PWM 模式允许两个中心对齐的 PWM 信号产生一个可编程的相位偏移。PWM 频率由 CRR 寄存器的值决定，占空比和相移由 CHxCR 和 CHxACR 寄存器决定。当计数器向上计数时，PWM 使用 CHxCR 的值作为向上计数的比较值。当计数器进入向下计数阶段，PWM 使用 CHxACR 的值作为向下计数的比较值。图 85 是在中心对齐计数模式下的非对称 PWM 模式的一个例子。

注：非对称 PWM 模式只能在中心对齐计数模式下运行。

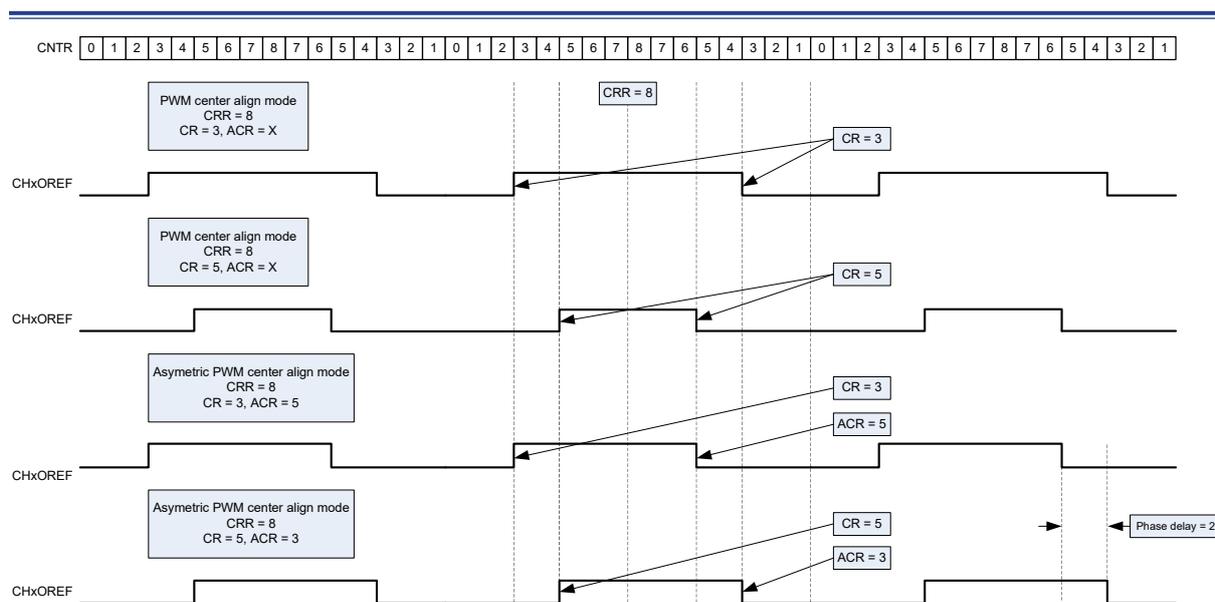


图 85. 非对称 PWM 模式与中心对齐计数模式

定时器互连

定时器可以在内部互相连接用于使定时器链接和同步。它可通过配置一个定时器工作在主机模式，而配置另一个定时器工作在从机模式来实现。下面是主 / 从模式触发器选择的几个示例。

使用一个定时器使能或除能另外一个定时器使其开始或停止计数

- 配置 PWM0 工作在主机模式，发送通道 0 输出参考信号 CH0OREF 作为触发器输出 (MMSEL = 0x4)
- 配置 PWM0 CH0OREF 波形
- 配置 PWM1 接收来自于 PWM0 触发器输出信号的输入触发器源 (TRSEL = 0x9)
- 配置 PWM1 工作在暂停模式下 (SMSEL = 0x5)
- 通过向 TME 位写入 1 来使能 PWM1
- 通过向 TME 位写入 1 来使能 PWM0

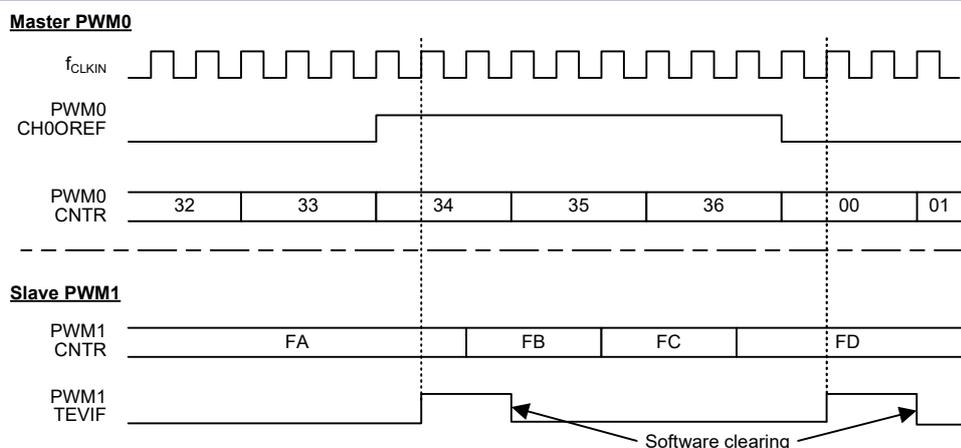


图 86. 用 PWM0 CH0OREF 信号暂停 PWM1

使用一个定时器触发另一个定时器开始计数

- 配置 PWM0 工作在主机模式，发送更新事件 UEV 作为触发输出 (MMSEL = 0x2)
- 通过设置 CRR 寄存器配置 PWM0 周期
- 配置 PWM1 从 PWM0 触发输出信号获得输入触发源 (TRSEL = 0x9)
- 配置 PWM1 工作在从机触发模式 (SMSEL = 0x6)
- 向 TME 位写入 1 启动 PWM0

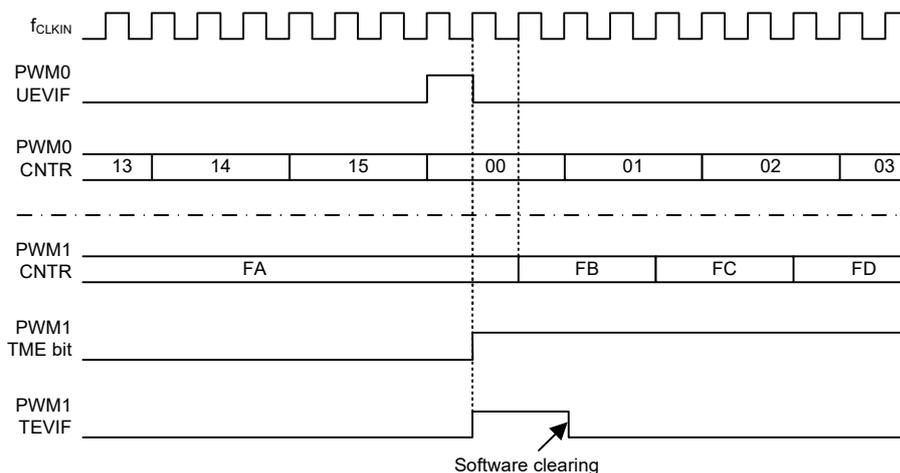


图 87. 用 PWM0 更新事件触发 PWM1

启动两个定时器同步响应外部触发

- 配置 PWM0 工作在主机模式下，发送使能信号作为一个触发输出信号 (MMSEL = 0x1)
- 设置 MDCFR 寄存器中的 TSE 位为 1，使能 PWM0 主机定时器同步功能使其与从机定时器同步
- 配置 PWM1 接收来自于 PWM0 的触发输出信号作为其输入触发源 (TRSEL = 0x9)
- 配置 PWM1 工作在从机触发模式下 (SMSEL = 0x6)
- 向 TME 位写入 1 启动 PWM0

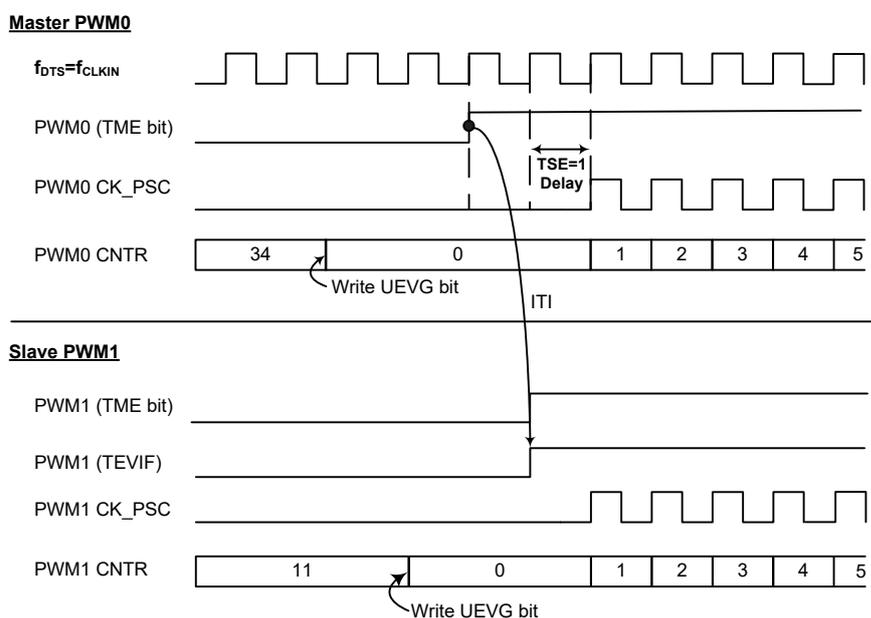


图 88. 用 PWM0 定时器使能信号触发 PWM0 和 PWM1

触发外设开启

为了与外设相连接 (如模拟 / 数字转换器), PWM 可以输出 MTO 信号或通道比较匹配输出信号 CHxOREF ($x = 0 \sim 3$) 作为外设的输入触发信号。

寄存器列表

下表显示了 PWM 寄存器及其复位值。

表 31. PWM 寄存器列表

寄存器	偏移量	描述	复位值
CNTCFR	0x000	定时器计数器配置寄存器	0x0000_0000
MDCFR	0x004	定时器模式配置寄存器	0x0000_0000
TRCFR	0x008	定时器触发配置寄存器	0x0000_0000
CTR	0x010	定时器控制寄存器	0x0000_0000
CH0OCFR	0x040	通道 0 输出配置寄存器	0x0000_0000
CH1OCFR	0x044	通道 1 输出配置寄存器	0x0000_0000
CH2OCFR	0x048	通道 2 输出配置寄存器	0x0000_0000
CH3OCFR	0x04C	通道 3 输出配置寄存器	0x0000_0000
CHCTR	0x050	通道控制寄存器	0x0000_0000
CHPOLR	0x054	通道极性配置寄存器	0x0000_0000
DICTR	0x074	定时器中断控制寄存器	0x0000_0000
EVGR	0x078	定时器事件发生器寄存器	0x0000_0000
INTSR	0x07C	定时器中断状态寄存器	0x0000_0000
CNTR	0x080	定时器计数器寄存器	0x0000_0000
PSCR	0x084	定时器预分频器寄存器	0x0000_0000
CRR	0x088	定时器计数器重载寄存器	0x0000_FFFF
CH0CR	0x090	通道 0 比较寄存器	0x0000_0000
CH1CR	0x094	通道 1 比较寄存器	0x0000_0000
CH2CR	0x098	通道 2 比较寄存器	0x0000_0000
CH3CR	0x09C	通道 3 比较寄存器	0x0000_0000
CH0ACR	0x0A0	通道 0 非对称比较寄存器	0x0000_0000
CH1ACR	0x0A4	通道 1 非对称比较寄存器	0x0000_0000
CH2ACR	0x0A8	通道 2 非对称比较寄存器	0x0000_0000
CH3ACR	0x0AC	通道 3 非对称比较寄存器	0x0000_0000

寄存器描述

定时器计数器配置寄存器 – CNTCFR

该寄存器定义了 PWM 计数器配置。

偏移量: 0x000
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24		
	保留位							DIR		
类型 / 复位								RW	0	
	23	22	21	20	19	18	17	16		
	保留位						CMSEL			
类型 / 复位							RW	0	RW	0
	15	14	13	12	11	10	9	8		
	保留位									
类型 / 复位										
	7	6	5	4	3	2	1	0		
	保留位						UGDIS	UEVDIS		
类型 / 复位							RW	0	RW	0

位	字段	描述
[24]	DIR	计数方向 0: 向上计数 1: 向下计数 注: 当定时器工作在中心对齐计数模式下或作为正交解码器使用时, 此位为只读位。
[17:16]	CMSEL	计数器模式选择 00: 边沿对齐计数模式。此模式下可正常向上和向下计数。计数方向由 DIR 位定义。 01: 中心对齐计数模式 1。计数器向上向下交替计数。比较匹配中断标志位在向下计数期间被置位。 10: 中心对齐计数模式 2。计数器向上向下交替计数。比较匹配中断标志位在向上计数期间被置位。 11: 中心对齐计数模式 3。计数器向上向下交替计数。比较匹配中断标志位在向上和向下计数期间被置位。
[1]	UGDIS	更新事件中断产生除能控制 0: 以下任何一个事件都可产生一个更新中断 – 计数器上溢 / 下溢 – 设置 UEVG 位 – 通过从机模式产生更新 1: 只在计数器上溢 / 下溢时产生一个更新中断
[0]	UEVDIS	更新事件除能控制 0: 以下任何一个事件都可能使能更新事件请求 – 计数器上溢 / 下溢 – 设置 UEVG 位 – 通过从机模式产生更新 1: 除能更新事件 (如果 UEVG 位被置位或从从机模式收到硬件重启, 那么计数器和预分频器将被重新初始化)

定时器模式配置寄存器 – MDCFR

该寄存器定义了 PWM 主机和从机模式选项以及单脉冲模式。

偏移量: 0x004

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24			
	保留位							SPMSET			
类型 / 复位								RW	0		
	23	22	21	20	19	18	17	16			
	保留位					MMSEL					
类型 / 复位						RW	0	RW	0	RW	0
	15	14	13	12	11	10	9	8			
	保留位					SMSEL					
类型 / 复位						RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0			
	保留位							TSE			
类型 / 复位								RW	0		

位 字段 描述

- [24] SPMSET 单脉冲模式设置
0: 无论更新事件是否发生, 计数器正常计数。
1: 下一个更新事件到来时, 计数器停止计数, 接着 TME 位被硬件清零。
- [18:16] MMSEL 主机模式选项
主机模式选项可选择用于同步其它从机定时器的 MTO 信号源。

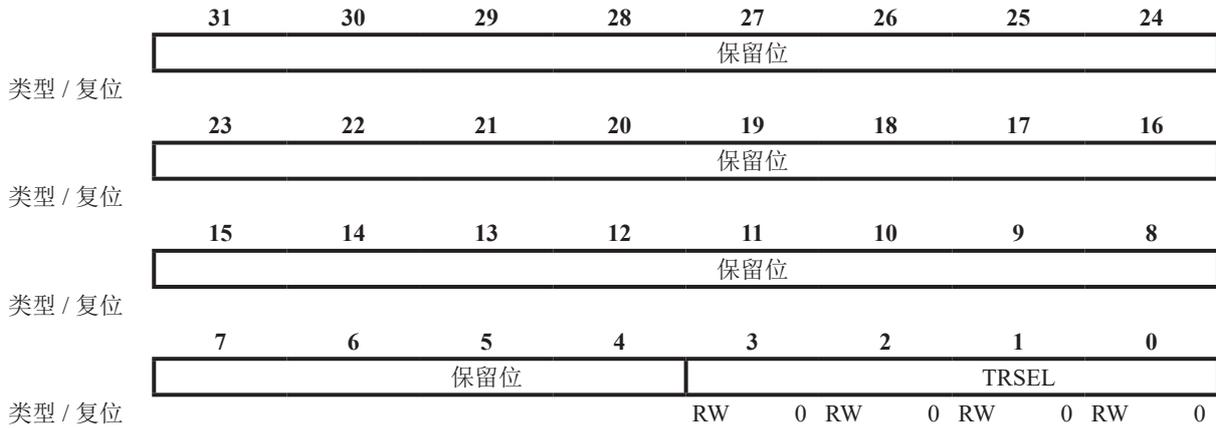
MMSEL[2:0]	模式	描述
000	复位模式	复位模式下的 MTO 输出信号由以下条件之一产生: 1. 软件置位 UEVG 位 2. 当定时器用于从机重启模式, STI 触发输入信号被输出到 MTO 信号线
001	使能模式	计数器使能信号作为触发输出。
010	更新模式	当 UEVDIS 位被清零时, 下列任一事件发生时更新事件将作为触发输出: 1. 计数器上溢 / 下溢 2. 软件设置 UEVG 位 3. 从机重启模式下从机触发输入
011	—	保留
100	比较输出 0	通道 0 输出参考信号 CH0OREF 作为触发输出。
101	比较输出 1	通道 1 输出参考信号 CH1OREF 作为触发输出。
110	比较输出 2	通道 2 输出参考信号 CH2OREF 作为触发输出。
111	比较输出 3	通道 3 输出参考信号 CH3OREF 作为触发输出。

位	字段	描述																											
[10:8]	SMSEL	从机模式选项																											
		<table border="1"> <thead> <tr> <th>SMSEL[2:0]</th> <th>模式</th> <th>描述</th> </tr> </thead> <tbody> <tr> <td>000</td> <td>除能模式</td> <td>预分频器直接用作内部时钟计时。</td> </tr> <tr> <td>001</td> <td>—</td> <td>保留</td> </tr> <tr> <td>010</td> <td>—</td> <td>保留</td> </tr> <tr> <td>011</td> <td>—</td> <td>保留</td> </tr> <tr> <td>100</td> <td>重启模式</td> <td>计数器从 0 或 CRR 影子寄存器的值重新开始计数，这取决于 STI 信号上升沿的计数模式。寄存器也将被更新。</td> </tr> <tr> <td>101</td> <td>暂停模式</td> <td>当所选择的触发输入信号 STI 是高电平时，计数器开始计数。当 STI 信号转换成低电平时，计数器立即停止计数，而不会产生复位。计数器开始和停止计数是由 STI 信号控制的。</td> </tr> <tr> <td>110</td> <td>触发模式</td> <td>在所选择的 STI 触发信号上升沿处，计数器从初始值开始计数。只有计数器的开启是由 STI 信号控制的。</td> </tr> <tr> <td>111</td> <td>STIED</td> <td>计数器使用所选的触发信号 STI 的上升沿计时。</td> </tr> </tbody> </table>	SMSEL[2:0]	模式	描述	000	除能模式	预分频器直接用作内部时钟计时。	001	—	保留	010	—	保留	011	—	保留	100	重启模式	计数器从 0 或 CRR 影子寄存器的值重新开始计数，这取决于 STI 信号上升沿的计数模式。寄存器也将被更新。	101	暂停模式	当所选择的触发输入信号 STI 是高电平时，计数器开始计数。当 STI 信号转换成低电平时，计数器立即停止计数，而不会产生复位。计数器开始和停止计数是由 STI 信号控制的。	110	触发模式	在所选择的 STI 触发信号上升沿处，计数器从初始值开始计数。只有计数器的开启是由 STI 信号控制的。	111	STIED	计数器使用所选的触发信号 STI 的上升沿计时。
SMSEL[2:0]	模式	描述																											
000	除能模式	预分频器直接用作内部时钟计时。																											
001	—	保留																											
010	—	保留																											
011	—	保留																											
100	重启模式	计数器从 0 或 CRR 影子寄存器的值重新开始计数，这取决于 STI 信号上升沿的计数模式。寄存器也将被更新。																											
101	暂停模式	当所选择的触发输入信号 STI 是高电平时，计数器开始计数。当 STI 信号转换成低电平时，计数器立即停止计数，而不会产生复位。计数器开始和停止计数是由 STI 信号控制的。																											
110	触发模式	在所选择的 STI 触发信号上升沿处，计数器从初始值开始计数。只有计数器的开启是由 STI 信号控制的。																											
111	STIED	计数器使用所选的触发信号 STI 的上升沿计时。																											
[0]	TSE	定时器同步使能 0: 无动作 1: 主机定时器 (当前定时器) 将通过 MTO 信号产生一个延时以同步其从机定时器																											

定时器触发配置寄存器 – TRCFR

该寄存器定义了 PWM 的触发源选项。

偏移量: 0x008
复位值: 0x0000_0000



位	字段	描述
[3:0]	TRSEL	触发源选择 这些位用来选择用于计数器同步的触发输入源 (STI) 0000: 通过设置 UEVG 位软件触发 1001: 内部定时器模块触发器 0 (ITI0) 1010: 内部定时器模块触发器 1 (ITI1) 1011: 内部定时器模块触发器 2 (ITI2) 其它: 保留 注: 当 SMSEL 字段为 0x0 除能从机模式时, 这些位才能被更新。

表 32. PWM 内部触发器连接

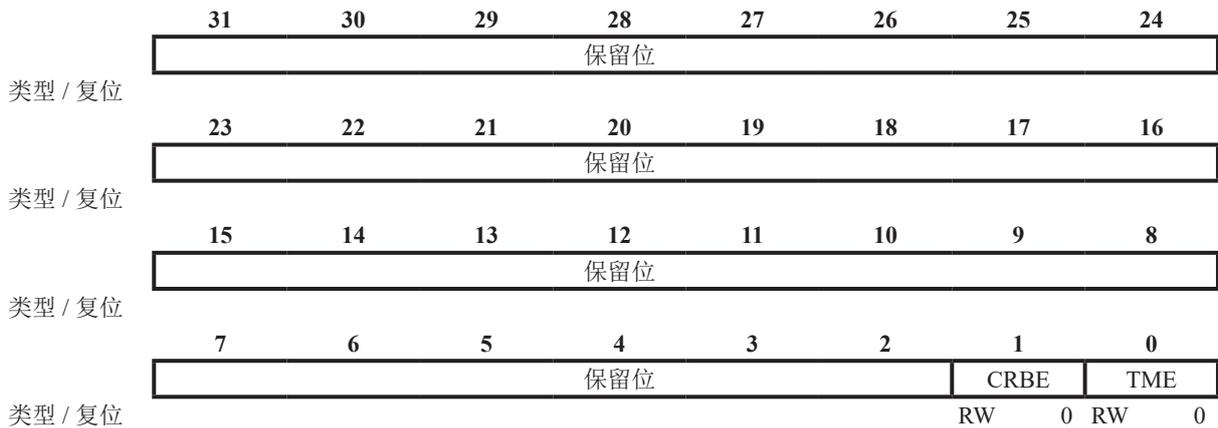
从机定时器模块	ITI0	ITI1	ITI2
PWM0	PWM1	GPTM	—
PWM1	PWM0	GPTM	—

定时器控制寄存器 – CTR

该寄存器定义了定时器使能位 (TME) 和 CRR 缓冲器使能位 (CRBE)。

偏移量: 0x010

复位值: 0x0000_0000



位	字段	描述
[1]	CRBE	计数器重载寄存器缓冲器使能位 0: 计数器重载寄存器能立即被更新 1: 直到更新事件发生时计数器重载寄存器才会被更新
[0]	TME	定时器使能位 0: PWM 关闭 1: PWM 开启 当 TME 位被清零, 计数器停止计数且 PWM 无功耗 (单脉冲模式和从机触发模式除外)。在单脉冲模式和从机触发模式中, TME 位可通过硬件自动置 1, 允许所有的 PWM 寄存器正常工作。

通道 0 输入配置寄存器 – CH0OCFR

该寄存器定义了通道 0 输出模式配置。

偏移量: 0x040
复位值: 0x0000_0000

31	30	29	28	27	26	25	24	保留位									
类型 / 复位																	
23	22	21	20	19	18	17	16	保留位									
类型 / 复位																	
15	14	13	12	11	10	9	8	CH0OM[3]	0								
类型 / 复位																	
7	6	5	4	3	2	1	0	保留位	CH0IMAE	CH0PRE	保留位	CH0OM[2:0]					
类型 / 复位																	
								RW	0	RW	0	RW	0	RW	0	RW	0

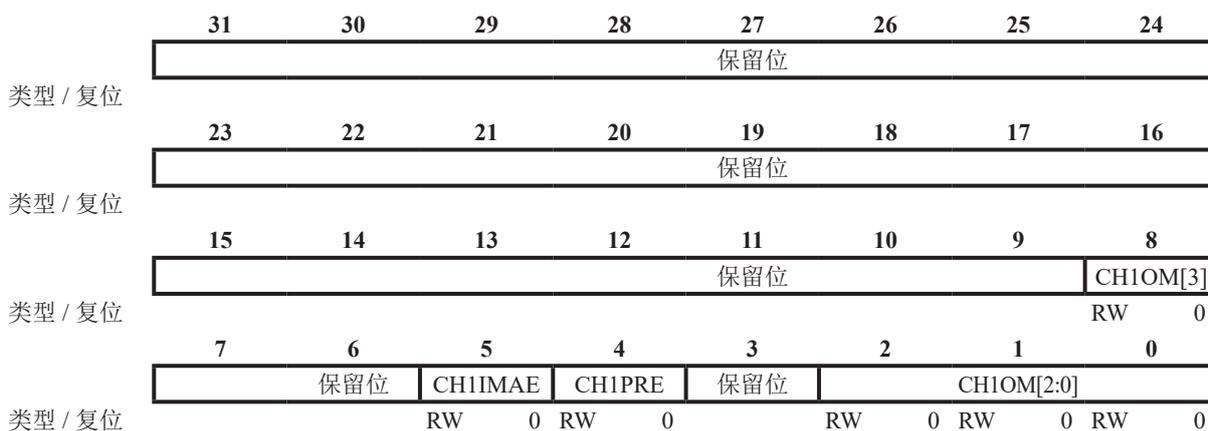
位	字段	描述
[5]	CH0IMAE	通道 0 立即有效使能位 0: 无动作 1: 单脉冲立即有效模式使能 无论 CNTR 和 CH0CR 值的比较结果如何, 在一个有效触发事件发生后, CH0OREF 会立即强制为比较匹配电平。 比较匹配电平将维持到下一个上溢或下溢事件的到来。 注: 只有当通道 0 被配置工作在 PWM 模式 1 或 PWM 模式 2 时, CH0IMAE 位可用。
[4]	CH0PRE	通道 0 比较寄存器 (CH0CR) 预载使能位 0: CH0CR 预载功能除能 当 CH0PRE 位清零, CH0CR 寄存器将立即被更新为新的值, 且立即可用。 1: CH0CR 预载功能使能 直到更新事件发生后, 新的 CH0CR 值才会被传送到影子寄存器中。

位	字段	描述
[8][2:0]	CH0OM[3:0]	<p>通道 0 输出模式设置</p> <p>这些位定义了输出参考信号 CH0OREF 的功能类型</p> <p>0000: 无变化</p> <p>0001: 比较匹配时输出 0</p> <p>0010: 比较匹配时输出 1</p> <p>0011: 比较匹配时输出翻转</p> <p>0100: 强制无效 – CH0OREF 强制为 0</p> <p>0101: 强制有效 – CH0OREF 强制为 1</p> <p>0110: PWM 模式 1</p> <ul style="list-style-type: none"> – 在向上计数期间, 当 CNTR < CH0CR, 通道 0 处于有效电平, 否则将处于无效电平。 – 在向下计数期间, 当 CNTR > CH0CR, 通道 0 处于无效电平, 否则将处于有效电平。 <p>0111: PWM 模式 2</p> <ul style="list-style-type: none"> – 在向上计数期间, 当 CNTR < CH0CR, 通道 0 处于无效电平, 否则将处于有效电平。 – 在向下计数期间, 当 CNTR > CH0CR, 通道 0 处于有效电平, 否则将处于无效电平。 <p>1110: 非对称 PWM 模式 1</p> <ul style="list-style-type: none"> – 在向上计数期间, 当 CNTR < CH0CR, 通道 0 处于有效电平, 否则将处于无效电平。 – 在向下计数期间, 当 CNTR > CH0ACR, 通道 0 处于无效电平, 否则将处于有效电平。 <p>1111: 非对称 PWM 模式 2</p> <ul style="list-style-type: none"> – 在向上计数期间, 当 CNTR < CH0CR, 通道 0 处于无效电平, 否则将处于有效电平。 – 在向下计数期间, 当 CNTR > CH0ACR, 通道 0 处于有效电平, 否则将处于无效电平。 <p>注: 当通道 0 用作非对称 PWM 输出模式时, 计数器配置寄存器中的计数器模式选择位必须配置为中心对齐计数模式 (CMSEL = 0x1 / 0x2 / 0x3)。</p>

通道 1 输入配置寄存器 – CH1OCFR

该寄存器定义了通道 1 输出模式配置。

偏移量: 0x044
复位值: 0x0000_0000



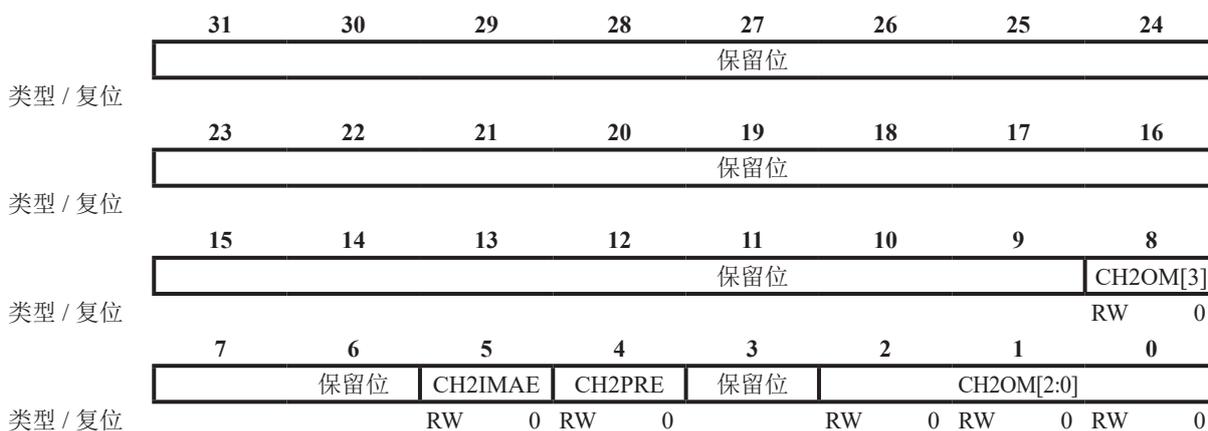
位	字段	描述
[5]	CH1IMAE	通道 1 立即有效使能位 0: 无动作 1: 单脉冲立即有效模式使能 无论 CNTR 和 CH1CR 值的比较结果如何, 在一个有效触发事件发生后, CH1OREF 会立即强制为比较匹配电平。 比较匹配电平将维持到下一个上溢或下溢事件的到来。 注: 只有当通道 1 被配置工作在 PWM 模式 1 或 PWM 模式 2 时, CH1IMAE 位可用。
[4]	CH1PRE	通道 1 比较寄存器 (CH1CR) 预载使能位 0: CH1CR 预载功能除能 当 CH1PRE 位清零, CH1CR 寄存器将立即被更新为新的值, 且立即可用。 1: CH1CR 预载功能使能 直到更新事件发生后, 新的 CH1CR 值才会被传送到影子寄存器中。

位	字段	描述
[8][2:0]	CH1OM[3:0]	<p>通道 1 输出模式设置</p> <p>这些位定义了输出参考信号 CH1OREF 的功能类型</p> <p>0000: 无变化</p> <p>0001: 比较匹配时输出 0</p> <p>0010: 比较匹配时输出 1</p> <p>0011: 比较匹配时输出翻转</p> <p>0100: 强制无效 – CH1OREF 强制为 0</p> <p>0101: 强制有效 – CH1OREF 强制为 1</p> <p>0110: PWM 模式 1</p> <ul style="list-style-type: none"> – 在向上计数期间, 当 CNTR < CH1CR, 通道 1 处于有效电平, 否则将处于无效电平。 – 在向下计数期间, 当 CNTR > CH1CR, 通道 1 处于无效电平, 否则将处于有效电平。 <p>0111: PWM 模式 2</p> <ul style="list-style-type: none"> – 在向上计数期间, 当 CNTR < CH1CR, 通道 1 处于无效电平, 否则将处于有效电平。 – 在向下计数期间, 当 CNTR > CH1CR, 通道 1 处于有效电平, 否则将处于无效电平。 <p>1110: 非对称 PWM 模式 1</p> <ul style="list-style-type: none"> – 在向上计数期间, 当 CNTR < CH1CR, 通道 1 处于有效电平, 否则将处于无效电平。 – 在向下计数期间, 当 CNTR > CH1ACR, 通道 1 处于无效电平, 否则将处于有效电平。 <p>1111: 非对称 PWM 模式 2</p> <ul style="list-style-type: none"> – 在向上计数期间, 当 CNTR < CH1CR, 通道 1 处于无效电平, 否则将处于有效电平。 – 在向下计数期间, 当 CNTR > CH1ACR, 通道 1 处于有效电平, 否则将处于无效电平。 <p>注: 当通道 1 用作非对称 PWM 输出模式时, 计数器配置寄存器中的计数器模式选择位必须配置为中心对齐计数模式 (CMSEL = 0x1 / 0x2 / 0x3)。</p>

通道 2 输入配置寄存器 – CH2OCFR

该寄存器定义了通道 2 输出模式配置。

偏移量: 0x048
复位值: 0x0000_0000



位	字段	描述
[5]	CH2IMAE	通道 2 立即有效使能位 0: 无动作 1: 单脉冲立即有效模式使能 无论 CNTR 和 CH2CR 值的比较结果如何, 在一个有效触发事件发生后, CH2OREF 会立即强制为比较匹配电平。 比较匹配电平将维持到下一个上溢或下溢事件的到来。 注: 只有当通道 1 被配置工作在 PWM 模式 1 或 PWM 模式 2 时, CH2IMAE 位可用。
[4]	CH2PRE	通道 2 比较寄存器 (CH2CR) 预载使能位 0: CH2CR 预载功能除能 当 CH1PRE 位清零, CH2CR 寄存器将立即被更新为新的值, 且立即可用。 1: CH2CR 预载功能使能 直到更新事件发生后, 新的 CH2CR 值才会被传送到影子寄存器中。

位	字段	描述
[8][2:0]	CH2OM[3:0]	<p>通道 2 输出模式设置</p> <p>这些位定义了输出参考信号 CH2OREF 的功能类型</p> <p>0000: 无变化</p> <p>0001: 比较匹配时输出 0</p> <p>0010: 比较匹配时输出 1</p> <p>0011: 比较匹配时输出翻转</p> <p>0100: 强制无效 – CH2OREF 强制为 0</p> <p>0101: 强制有效 – CH2OREF 强制为 1</p> <p>0110: PWM 模式 1</p> <ul style="list-style-type: none"> – 在向上计数期间, 当 CNTR < CH2CR, 通道 2 处于有效电平, 否则将处于无效电平。 – 在向下计数期间, 当 CNTR > CH2CR, 通道 2 处于无效电平, 否则将处于有效电平。 <p>0111: PWM 模式 2</p> <ul style="list-style-type: none"> – 在向上计数期间, 当 CNTR < CH2CR, 通道 2 处于无效电平, 否则将处于有效电平。 – 在向下计数期间, 当 CNTR > CH2CR, 通道 2 处于有效电平, 否则将处于无效电平。 <p>1110: 非对称 PWM 模式 1</p> <ul style="list-style-type: none"> – 在向上计数期间, 当 CNTR < CH2CR, 通道 2 处于有效电平, 否则将处于无效电平。 – 在向下计数期间, 当 CNTR > CH2ACR, 通道 2 处于无效电平, 否则将处于有效电平。 <p>1111: 非对称 PWM 模式 2</p> <ul style="list-style-type: none"> – 在向上计数期间, 当 CNTR < CH2CR, 通道 2 处于无效电平, 否则将处于有效电平。 – 在向下计数期间, 当 CNTR > CH2ACR, 通道 2 处于有效电平, 否则将处于无效电平。 <p>注: 当通道 2 用作非对称 PWM 输出模式时, 计数器配置寄存器中的计数器模式选择位必须配置为中心对齐计数模式 (CMSEL = 0x1 / 0x2 / 0x3)。</p>

通道 3 输入配置寄存器 – CH3OCFR

该寄存器定义了通道 3 输出模式配置。

偏移量: 0x04C

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位								
	15	14	13	12	11	10	9	8	
类型 / 复位	保留位							CH3OM[3]	
	7	6	5	4	3	2	1	0	
类型 / 复位		保留位	CH3IMAE	CH3PRE	保留位	CH3OM[2:0]			
			RW	0	RW	0	RW	0	RW
									0

位	字段	描述
[5]	CH3IMAE	通道 3 立即有效使能位 0: 无动作 1: 单脉冲立即有效模式使能 无论 CNTR 和 CH3CR 值的比较结果如何, 在一个有效触发事件发生后, CH3OREF 会立即强制为比较匹配电平。 比较匹配电平将维持到下一个上溢或下溢事件的到来。 注: 只有当通道 3 被配置工作在 PWM 模式 1 或 PWM 模式 2 时, CH3IMAE 位可用。
[4]	CH3PRE	通道 3 比较寄存器 (CH3CR) 预载使能位 0: CH3CR 预载功能除能 当 CH3PRE 位清零, CH3CR 寄存器将立即被更新为新的值, 且立即可用。 1: CH3CR 预载功能使能 直到更新事件发生后, 新的 CH3CR 值才会被传送到影子寄存器中。

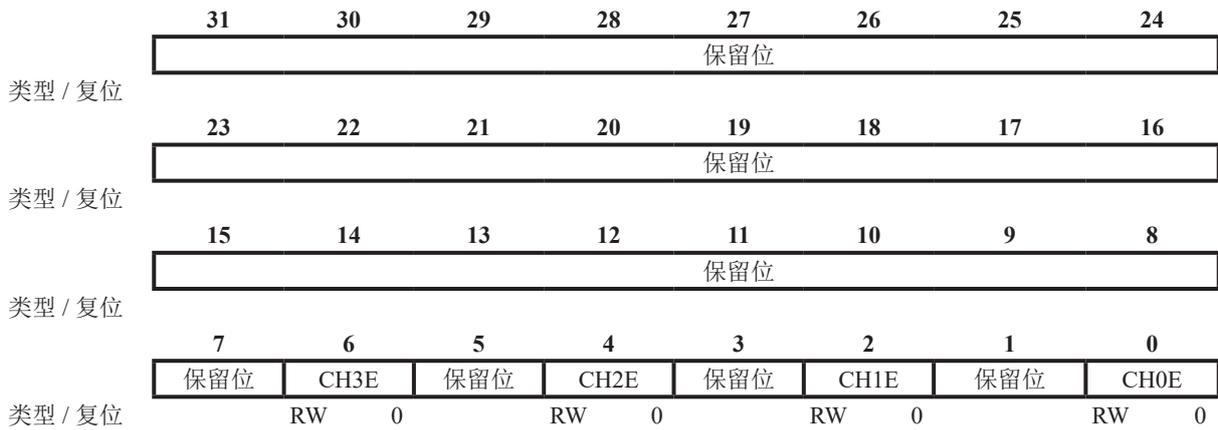
位	字段	描述
[8][2:0]	CH3OM[3:0]	<p>通道 3 输出模式设置</p> <p>这些位定义了输出参考信号 CH3OREF 的功能类型</p> <p>0000: 无变化</p> <p>0001: 比较匹配时输出 0</p> <p>0010: 比较匹配时输出 1</p> <p>0011: 比较匹配时输出翻转</p> <p>0100: 强制无效 – CH3OREF 强制为 0</p> <p>0101: 强制有效 – CH3OREF 强制为 1</p> <p>0110: PWM 模式 1</p> <ul style="list-style-type: none"> – 在向上计数期间, 当 CNTR < CH3CR, 通道 3 处于有效电平, 否则将处于无效电平。 – 在向下计数期间, 当 CNTR > CH3CR, 通道 3 处于无效电平, 否则将处于有效电平。 <p>0111: PWM 模式 2</p> <ul style="list-style-type: none"> – 在向上计数期间, 当 CNTR < CH3CR, 通道 3 处于无效电平, 否则将处于有效电平。 – 在向下计数期间, 当 CNTR > CH3CR, 通道 3 处于有效电平, 否则将处于无效电平。 <p>1110: 非对称 PWM 模式 1</p> <ul style="list-style-type: none"> – 在向上计数期间, 当 CNTR < CH3CR, 通道 3 处于有效电平, 否则将处于无效电平。 – 在向下计数期间, 当 CNTR > CH3ACR, 通道 3 处于无效电平, 否则将处于有效电平。 <p>1111: 非对称 PWM 模式 2</p> <ul style="list-style-type: none"> – 在向上计数期间, 当 CNTR < CH3CR, 通道 3 处于无效电平, 否则将处于有效电平。 – 在向下计数期间, 当 CNTR > CH3ACR, 通道 3 处于有效电平, 否则将处于无效电平。 <p>注: 当通道 3 用作非对称 PWM 输出模式时, 计数器配置寄存器中的计数器模式选择位必须配置为中心对齐计数模式 (CMSEL = 0x1 / 0x2 / 0x3)。</p>

通道控制寄存器 – CHCTR

该寄存器包含了通道比较输出功能使能控制位。

偏移量: 0x050

复位值: 0x0000_0000



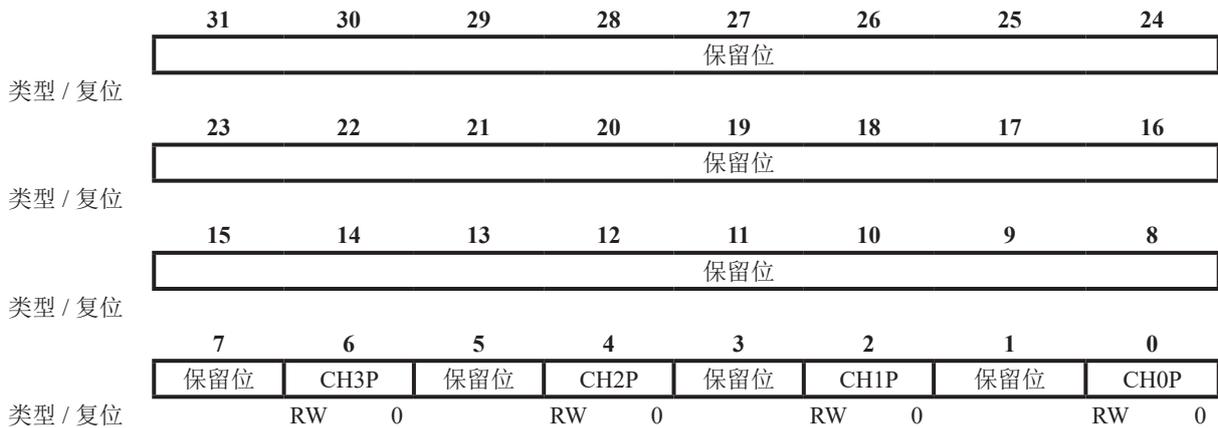
位	字段	描述
[6]	CH3E	通道 3 比较使能位 0: 关闭 – 通道 3 输出信号 CH3O 无作用 1: 开启 – 通道 3 输出信号 CH3O 在相应的输出脚产生
[4]	CH2E	通道 2 比较使能位 0: 关闭 – 通道 2 输出信号 CH2O 无作用 1: 开启 – 通道 2 输出信号 CH2O 在相应的输出脚产生
[2]	CH1E	通道 1 比较使能位 0: 关闭 – 通道 1 输出信号 CH1O 无作用 1: 开启 – 通道 1 输出信号 CH1O 在相应的输出脚产生
[0]	CH0E	通道 0 比较使能位 0: 关闭 – 通道 0 输出信号 CH0O 无作用 1: 开启 – 通道 0 输出信号 CH0O 在相应的输出脚产生

通道极性配置寄存器 – CHPOLR

该寄存器包含了通道比较输出极性的控制

偏移量: 0x054

复位值: 0x0000_0000



位	字段	描述
[6]	CH3P	通道 3 比较极性 0: 通道 3 输出高电平有效 1: 通道 3 输出低电平有效
[4]	CH2P	通道 2 比较极性 0: 通道 2 输出高电平有效 1: 通道 2 输出低电平有效
[2]	CH1P	通道 1 比较极性 0: 通道 1 输出高电平有效 1: 通道 1 输出低电平有效
[0]	CH0P	通道 0 比较极性 0: 通道 0 输出高电平有效 1: 通道 0 输出低电平有效

定时器中断控制寄存器 – DICTR

该寄存器包含了定时器中断使能控制位。

偏移量: 0x074

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24
类型 / 复位	保留位							
	23	22	21	20	19	18	17	16
类型 / 复位	保留位							
	15	14	13	12	11	10	9	8
类型 / 复位	保留位					TEVIE	保留位	UEVIE
	7	6	5	4	3	2	1	0
类型 / 复位	保留位				CH3CIE	CH2CIE	CH1CIE	CH0CIE
					RW	0	RW	0
					RW	0	RW	0

位	字段	描述
[10]	TEVIE	触发事件中断使能位 0: 除能触发事件中断 1: 使能触发事件中断
[8]	UEVIE	更新事件中断使能位 0: 除能更新事件中断 1: 使能更新事件中断
[3]	CH3CIE	通道 3 比较中断使能位 0: 除能通道 3 中断 1: 使能通道 3 中断
[2]	CH2CIE	通道 2 比较中断使能位 0: 除能通道 2 中断 1: 使能通道 2 中断
[1]	CH1CIE	通道 1 比较中断使能位 0: 除能通道 1 中断 1: 使能通道 1 中断
[0]	CH0CIE	通道 0 比较中断使能位 0: 除能通道 0 中断 1: 使能通道 0 中断

定时器事件发生器寄存器 – EVGR

该寄存器包含了软件事件发生控制位。

偏移量: 0x078
复位值: 0x0000_0000

31	30	29	28	27	26	25	24
保留位							
23	22	21	20	19	18	17	16
保留位							
15	14	13	12	11	10	9	8
保留位				TEVG		保留位	UEVG
				WO 0		WO 0	
7	6	5	4	3	2	1	0
保留位				CH3CG	CH2CG	CH1CG	CH0CG
				WO 0		WO 0	WO 0

位	字段	描述
[10]	TEVG	触发事件发生 触发事件 TEV 可由置位此位来发生。它由硬件自动清零。 0: 无动作 1: TEVIF 标志位置位
[8]	UEVG	更新事件发生 更新事件 UEV 可由置位此位来发生。它由硬件自动清零。 0: 无动作 1: 初始化计数器 计数器的值回到 0 或 CRR 预载值，这取决于当前定时器使用的计数模式。相关寄存器也会被更新。详细描述请参考相关章节。
[3]	CH3CG	通道 3 比较发生 通道 3 比较事件可由置位此位来发生。它由硬件自动清零。 0: 无动作 1: 通道 3 发生比较事件
[2]	CH2CG	通道 2 比较发生 通道 2 比较事件可由置位此位来发生。它由硬件自动清零。 0: 无动作 1: 通道 2 发生比较事件
[1]	CH1CG	通道 1 比较发生 通道 1 比较事件可由置位此位来发生。它由硬件自动清零。 0: 无动作 1: 通道 1 发生比较事件
[0]	CH0CG	通道 0 比较发生 通道 0 比较事件可由置位此位来发生。它由硬件自动清零。 0: 无动作 1: 通道 0 发生比较事件

定时器中断状态寄存器 – INTSR

该寄存器存储了定时器中断的状态。

偏移量: 0x07C
复位值: 0x0000_0000

31	30	29	28	27	26	25	24
保留位							
23	22	21	20	19	18	17	16
保留位							
15	14	13	12	11	10	9	8
保留位				W0C 0		保留位	UEVIF
7	6	5	4	3	2	1	0
保留位				CH3CIF	CH2CIF	CH1CIF	CH0CIF
				W0C 0	W0C 0	W0C 0	W0C 0

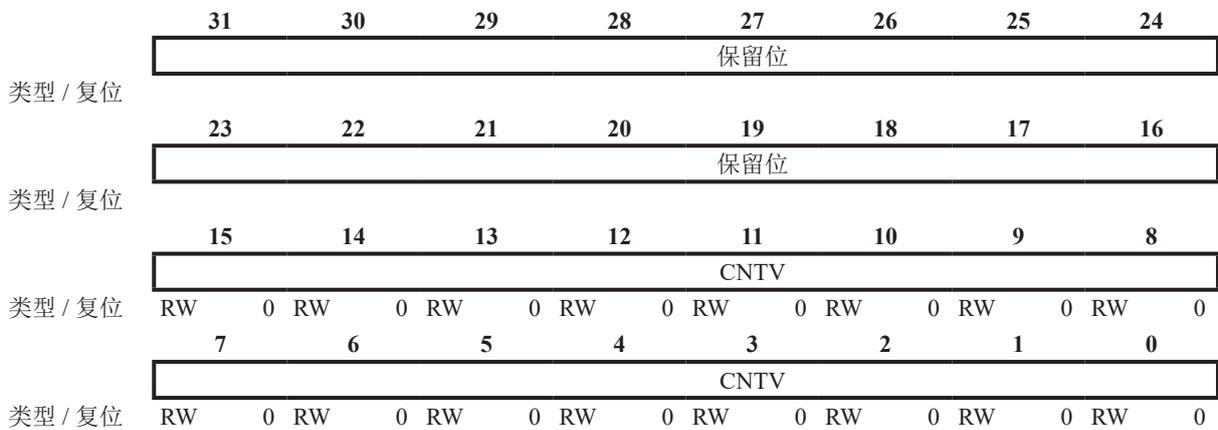
位	字段	描述
[10]	TEVIF	触发事件中中断标志位 此位在触发事件发生时通过硬件置位，由软件清零。 0: 无触发事件发生 1: 触发事件发生
[8]	UEVIF	更新事件中中断标志位 此位在更新事件发生时通过硬件置位，由软件清零。 0: 无更新事件发生 1: 更新事件发生 注: 更新事件在以下情况下发生: – 计数器上溢或下溢 – UEVG 位被置位 – 来自从机触发输入的重启触发事件发生
[3]	CH3CIF	通道 3 比较中断标志位 0: 无匹配事件发生 1: 计数器 CNTR 内容与 CH3CR 寄存器内容匹配 除中心对齐计数模式外，当计数器的值与 CH3CR 的值匹配时，此位被硬件置位。通过软件清零。
[2]	CH2CIF	通道 2 比较中断标志位 0: 无匹配事件发生 1: 计数器 CNTR 内容与 CH2CR 寄存器内容匹配 除中心对齐计数模式外，当计数器的值与 CH2CR 的值匹配时，此位被硬件置位。通过软件清零。
[1]	CH1CIF	通道 1 比较中断标志位 0: 无匹配事件发生 1: 计数器 CNTR 内容与 CH1CR 寄存器内容匹配 除中心对齐计数模式外，当计数器的值与 CH1CR 的值匹配时，此位被硬件置位。通过软件清零。

位	字段	描述
[0]	CH0CIF	通道 0 比较中断标志位 0: 无匹配事件发生 1: 计数器 CNTR 内容与 CH0CR 寄存器内容匹配 除中心对齐计数模式外，当计数器的值与 CH0CR 的值匹配时，此位被硬件置位。通过软件清零。

定时器计数器寄存器 – CNTR

该寄存器存储了定时器计数器的值。

偏移量: 0x080
复位值: 0x0000_0000



位	字段	描述
[15:0]	CNTV	计数器的值

定时器预分频器寄存器 – PSCR

该寄存器定义了定时器预分频器的值以产生计数器时钟。

偏移量: 0x084
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	保留位							
类型 / 复位																
	23	22	21	20	19	18	17	16	保留位							
类型 / 复位																
	15	14	13	12	11	10	9	8	PSCV							
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0	PSCV							
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0

位	字段	描述
[15:0]	PSCV	预分频器的值 这些位用来定义预分频器的值以产生计数器的时钟频率 f_{CK_CNT} 。 $f_{CK_CNT} = \frac{f_{CK_PSC}}{PSCV[15:0] + 1}$ f_{CK_PSC} 代表预分频器时钟源。

定时器计数器重载寄存器 – CRR

该寄存器定义了定时器计数器重载值。

偏移量: 0x088
复位值: 0x0000_FFFF

	31	30	29	28	27	26	25	24	保留位							
类型 / 复位																
	23	22	21	20	19	18	17	16	保留位							
类型 / 复位																
	15	14	13	12	11	10	9	8	CRV							
类型 / 复位	RW	1	RW	1	RW	1	RW	1	RW	1	RW	1	RW	1	RW	1
	7	6	5	4	3	2	1	0	CRV							
类型 / 复位	RW	1	RW	1	RW	1	RW	1	RW	1	RW	1	RW	1	RW	1

位	字段	描述
[15:0]	CRV	计数器重载值 CRV 是加载到实际计数器寄存器中的重载值。

通道 0 比较寄存器 – CH0CR

该寄存器定义了定时器通道 0 比较值。

偏移量: 0x090

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	保留位					
类型 / 复位														
	23	22	21	20	19	18	17	16	保留位					
类型 / 复位														
	15	14	13	12	11	10	9	8	CH0CV					
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0	CH0CV					
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0

位	字段	描述
[15:0]	CH0CV	通道 0 比较值 CH0CR 值和计数器的值相比较，比较结果用于触发 CH0OREF 输出信号。

通道 1 比较寄存器 – CH1CR

该寄存器定义了定时器通道 1 比较值。

偏移量: 0x094

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	保留位					
类型 / 复位														
	23	22	21	20	19	18	17	16	保留位					
类型 / 复位														
	15	14	13	12	11	10	9	8	CH1CV					
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0	CH1CV					
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0

位	字段	描述
[15:0]	CH1CV	通道 1 比较值 CH1CR 值和计数器的值相比较，比较结果用于触发 CH1OREF 输出信号。

通道 2 比较寄存器 – CH2CR

该寄存器存储了定时器中断的状态。

偏移量: 0x098

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24								
类型 / 复位	保留位															
	23	22	21	20	19	18	17	16								
类型 / 复位	保留位															
	15	14	13	12	11	10	9	8								
类型 / 复位	CH2CV															
	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0								
类型 / 复位	CH2CV															
	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0

位	字段	描述
[15:0]	CH2CV	通道 2 比较值 CH2CR 值和计数器的值相比较，比较结果用于触发 CH2OREF 输出信号。

通道 3 比较寄存器 – CH3CR

该寄存器存储了定时器计数器的值。

偏移量: 0x09C

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24								
类型 / 复位	保留位															
	23	22	21	20	19	18	17	16								
类型 / 复位	保留位															
	15	14	13	12	11	10	9	8								
类型 / 复位	CH3CV															
	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0								
类型 / 复位	CH3CV															
	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0

位	字段	描述
[15:0]	CH3CV	通道 3 比较值 CH3CR 值和计数器的值相比较，比较结果用于触发 CH3OREF 输出信号。

通道 0 非对称比较寄存器 – CH0ACR

该寄存器定义了定时器通道 0 非对称比较值。

偏移量: 0x0A0
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	保留位							
类型 / 复位																
	23	22	21	20	19	18	17	16	保留位							
类型 / 复位																
	15	14	13	12	11	10	9	8	CH0ACV							
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0	CH0ACV							
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0

位	字段	描述
[15:0]	CH0ACV	通道 0 非对称比较值 当通道 0 被配置为非对称 PWM 模式且计数器向下计数时，写入该寄存器的值将与计数器的值进行比较。

通道 1 非对称比较寄存器 – CH1ACR

该寄存器定义了定时器通道 1 非对称比较值。

偏移量: 0x0A4
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	保留位							
类型 / 复位																
	23	22	21	20	19	18	17	16	保留位							
类型 / 复位																
	15	14	13	12	11	10	9	8	CH1ACV							
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0	CH1ACV							
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0

位	字段	描述
[15:0]	CH1ACV	通道 1 非对称比较值 当通道 1 被配置为非对称 PWM 模式且计数器向下计数时，写入该寄存器的值将与计数器的值进行比较。

通道 2 非对称比较寄存器 – CH2ACR

该寄存器定义了定时器通道 2 非对称比较值。

偏移量: 0x0A8
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	保留位							
类型 / 复位																
	23	22	21	20	19	18	17	16	保留位							
类型 / 复位																
	15	14	13	12	11	10	9	8	CH2ACV							
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0	CH2ACV							
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0

位	字段	描述
[15:0]	CH2ACV	通道 2 非对称比较值 当通道 2 被配置为非对称 PWM 模式且计数器向下计数时，写入该寄存器的值将与计数器的值进行比较。

通道 3 非对称比较寄存器 – CH3ACR

该寄存器定义了定时器通道 3 非对称比较值。

偏移量: 0x0AC
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	保留位							
类型 / 复位																
	23	22	21	20	19	18	17	16	保留位							
类型 / 复位																
	15	14	13	12	11	10	9	8	CH3ACV							
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0	CH3ACV							
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0

位	字段	描述
[15:0]	CH3ACV	通道 3 非对称比较值 当通道 3 被配置为非对称 PWM 模式且计数器向下计数时，写入该寄存器的值将与计数器的值进行比较。

15 基本功能定时器 (BFTM)

简介

基本功能定时器模块 (BFTM) 是一个 32-bit 向上计数型计数器, 用来测量时间间隔、产生单次或重复中断。BFTM 有两种工作模式, 即重复模式和单次模式。在内部比较器每次产生比较匹配事件时重复模式都会重启计数器。BFTM 也支持单次模式, 当比较匹配事件发生时, 计数器会被强制停止计数。

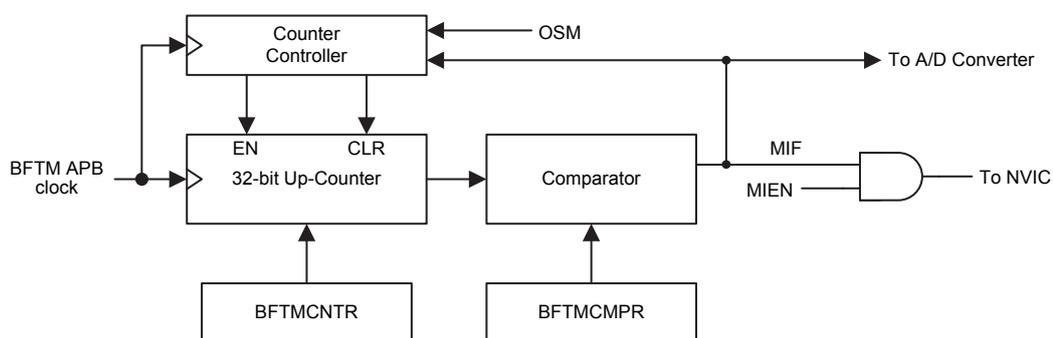


图 89. BFTM 方框图

特性

- 32-bit 向上计数型计数器
- 比较匹配功能
- 包含调试模式
- 时钟源: BFTM APB 时钟
- 运行时计数器值可读写
- 单次模式: 计数器在比较匹配发生时停止计数
- 重复模式: 计数器在比较匹配发生时重启
- 比较匹配中断使能 / 除能控制

功能描述

BFTM 是一个由 BFTM APB 时钟 (PCLK) 驱动的 32-bit 向上计数型计数器。任何时候计数器的值都可被更改或读取，即使定时器还在计数。BFTM 支持两种工作模式，即重复模式和单次模式，可测量时间间隔或产生周期性时间周期。

重复模式

BFTM 从 0 开始向上计数到某一指定的比较值，该值由 BFTMCMPR 寄存器预定义。当 BFTM 工作于重复模式时且计数器值已达到 BFTMCMPR 寄存器指定的比较值，定时器将会产生一个比较匹配事件信号 MIF，此时计数器将被复位为 0 并重新开始计数。当 MIF 信号产生时，如果通过设置相关中断控制位 MIEN 为 1 使能比较匹配中断，则 BFTM 比较匹配中断将周期性地产生。如果通过清零 CEN 位除能计数器，计数器将停止计数并保持当前值不变。

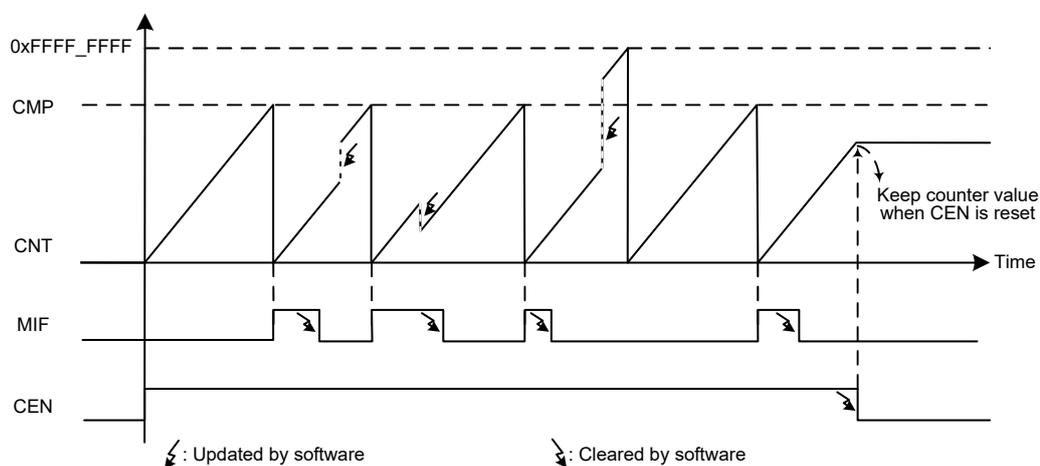


图 90. BFTM – 重复模式

单次模式

通过设置 BFTMCR 寄存器中的 OSM 位为 1, BFTM 将工作于单次模式。通过应用程序把 CEN 位置为 1 时, BFTM 开始计数。如果 CEN 位通过应用程序清零, 则计数器保持当前值不变。然而, 如果计数器比较匹配事件发生时 CEN 位通过内部硬件自动清零, 则计数器停止计数且其值将会被复位。

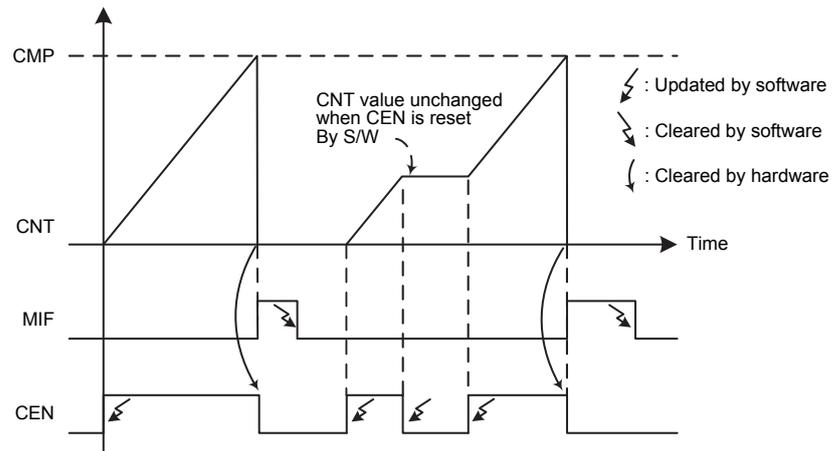


图 91. BFTM – 单次模式

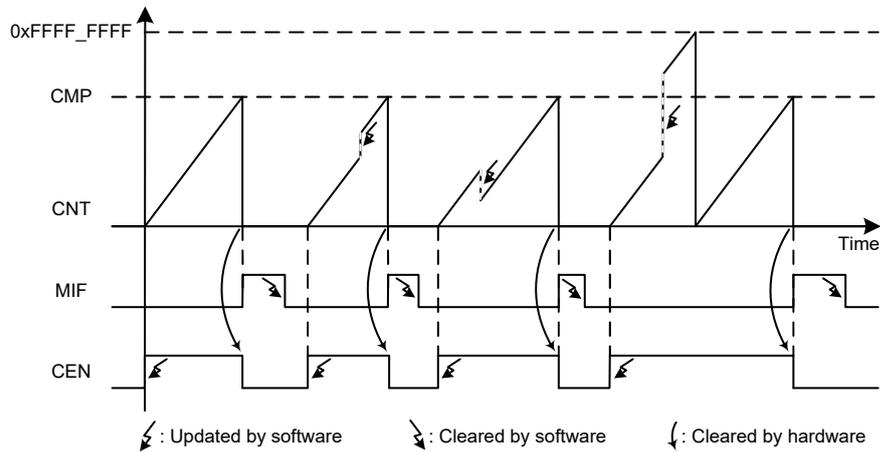


图 92. BFTM – 单次模式计数器更新

触发 A/D 转换器开启

当 BFTM 比较匹配事件发生时, 将会产生比较匹配事件中断标志 MIF, 可用作 A/D 转换器输入触发源。

寄存器列表

下表所示为 BFTM 寄存器及其复位值。

表 33. BFTM 寄存器列表

寄存器	偏移量	描述	复位值
BFTMCR	0x000	BFTM 控制寄存器	0x0000_0000
BFTMSR	0x004	BFTM 状态寄存器	0x0000_0000
BFTMCNTR	0x008	BFTM 计数器值寄存器	0x0000_0000
BFTMCMR	0x00C	BFTM 比较值寄存器	0xFFFF_FFFF

寄存器描述

BFTM 控制寄存器 – BFTMCR

该寄存器定义了所有的 BFTM 控制位。

偏移量: 0x000

复位值: 0x0000_0000

31	30	29	28	27	26	25	24	保留位				
类型 / 复位												
23	22	21	20	19	18	17	16	保留位				
类型 / 复位												
15	14	13	12	11	10	9	8	保留位				
类型 / 复位												
7	6	5	4	3	2	1	0	保留位		CEN	OSM	MIEN
类型 / 复位												
					RW	0	RW	0	RW	0		

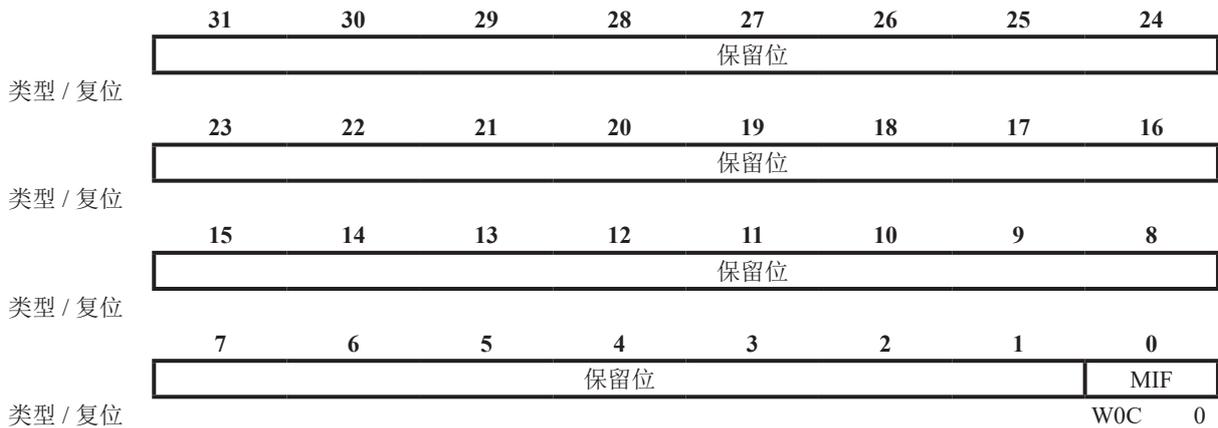
位	字段	描述
[2]	CEN	BFTM 计数器使能控制位 0: BFTM 除能 1: BFTM 使能 该位置 1 时, BFTM 计数器开始计数。通过应用程序把 CEN 位清零时, 不管处于重复模式还是单次模式, 计数器都将停止计数且保持当前值不变。但在单次模式中, 如果是因为比较匹配事件发生, 定时器硬件电路把 CEN 位清零, 计数器将停止计数并复位为 0。
[1]	OSM	BFTM 单次模式选项 0: 计数器工作于重复模式 1: 计数器工作于单次模式
[0]	MIEN	BFTM 比较匹配中断使能控制位 0: 比较匹配中断除能 1: 比较匹配中断使能

BFTM 状态寄存器 – BFTMSR

该寄存器定义了 BFTM 的各种状态。

偏移量: 0x004

复位值: 0x0000_0000



位	字段	描述
[0]	MIF	BFTM 比较匹配中断标志位 0: 无比较匹配事件发生 1: 比较匹配事件发生 当计数器值 CNT 等于比较寄存器值 CMP 时, 将会发生比较匹配事件, 相关中断标志位 MIF 通过硬件置位, 由软件清零。

BFTM 计数器值寄存器 – BFTMCNTR

该寄存器定义了 BFTM 计数器的值。

偏移量: 0x008
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	CNT			
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0
	23	22	21	20	19	18	17	16	CNT			
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0
	15	14	13	12	11	10	9	8	CNT			
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0	CNT			
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0

位	字段	描述
[31:0]	CNT	BFTM 计数器值寄存器 该字段在运行时都可读取或写入，用来存储 32-bit BFTM 计数器的值。

BFTM 比较值寄存器 – BFTMCMPR

该寄存器定义了 BFTM 的比较值。

偏移量: 0x00C
复位值: 0xFFFF_FFFF

	31	30	29	28	27	26	25	24	CMP			
类型 / 复位	RW	1	RW	1	RW	1	RW	1	RW	1	RW	1
	23	22	21	20	19	18	17	16	CMP			
类型 / 复位	RW	1	RW	1	RW	1	RW	1	RW	1	RW	1
	15	14	13	12	11	10	9	8	CMP			
类型 / 复位	RW	1	RW	1	RW	1	RW	1	RW	1	RW	1
	7	6	5	4	3	2	1	0	CMP			
类型 / 复位	RW	1	RW	1	RW	1	RW	1	RW	1	RW	1

位	字段	描述
[31:0]	CMP	BFTM 比较值寄存器 该寄存器用于设置 32-bit BFTM 的比较值，与 BFTM 计数器值作比较。

16 实时时钟 (RTC)

简介

实时时钟 (RTC) 电路包括 APB 接口、一个 24-bit 向上计数器、一个控制寄存器、一个预分频器、一个比较寄存器和一个状态寄存器。RTC 电路位于 V_{DD15} 域中, 如下图所示。RTC 计数器用作系统从省电模式恢复的一个唤醒定时器。详细的 RTC 功能会在以下内容中描述。

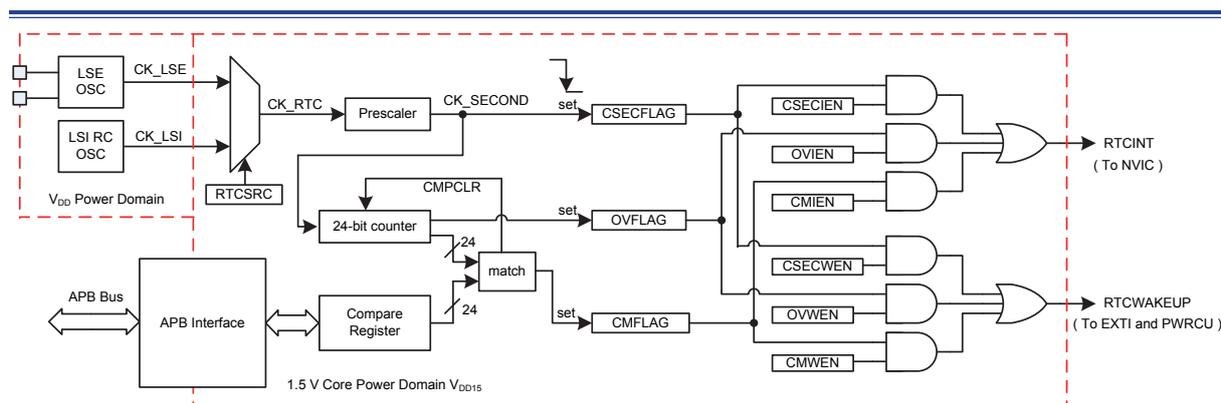


图 93. RTC 方框图

特性

- 24-bit 向上计数器用于计算所用的时间
- 可编程时钟预分频器
 - 分频系数: 1, 2, 4, 8, ..., 32768
- 24-bit 比较寄存器用于闹钟功能
- RTC 时钟源
 - LSE 振荡器时钟
 - LSI 振荡器时钟
- 三个 RTC 中断 / 唤醒设置
 - RTC 秒时钟中断 / 唤醒
 - RTC 比较匹配中断 / 唤醒
 - RTC 计数器溢出中断 / 唤醒
- RTC 中断 / 唤醒事件可与电源管理一起工作使单片机从省电模式中唤醒

功能描述

RTC 相关寄存器复位

RTC 寄存器只能由 V_{DD15} 域上电复位 (POR15) 或通过设置 PWRCCR 寄存器中的 PWCURST 位使 V_{DD15} 域软件复位的方式来复位。其它复位事件无法复位 RTC 寄存器。

低速时钟配置

默认的 RTC 时钟源, CK_RTC, 来自于 LSI 振荡器。CK_RTC 时钟可以来自于外部 32.768 kHz 晶体振荡器 LSE 或内部 32 kHz RC 振荡器 LSI, 由 RTCCR 寄存器中的 RTCSRC 位来设置。预分频器由 RPRE[3:0] 字段设置 CK_RTC 时钟分频比, 分频范围是 $2^0 \sim 2^{15}$ 。例如, 当 CK_RTC 时钟频率是 32.768 kHz 时, 设置预分频值 RPRE[3:0] 为 0xF, 将产生一个精确的 1 Hz CK_SECOND 时钟。LSE 振荡器由 RTCCR 寄存器中的 LSEEN 控制位使能。另外, LSE 振荡器启动模式由 RTCCR 寄存器中的 LSESM 位进行选择, 可以缩短 LSE 振荡器启动时间或降低功耗, 两者需根据特殊应用的需求来权衡。下表是不同启动模式下启动时间和功耗的一个例子, 可作为参考。

表 34. LSE 在不同启动模式下的工作电流和启动时间

启动模式	RTCCR 寄存器中 LSESM 的设置	工作电流	启动时间
正常启动	0	2.0 μ A	500 ms 以上
快速启动	1	3.5 μ A	300 ms 以下

@ $V_{DD} = 3.3$ V 和 LSE 时钟 = 32.768 kHz; 这些值仅供参考, 实际值取决于外部 32.768 kHz 晶体的规格。

RTC 计数器操作

RTC 提供了一个 24-bit 向上计数器, 它在 CK_SECOND 时钟的下降沿递增并可以通过 APB 总线从 RTCCNT 寄存器中异步读取计数器的值。还提供了一个 24-bit 比较寄存器, RTCCMP, 用来存储特定值并与 RTCCNT 的内容进行比较。这一操作用于预定义一个时间间隔。当 RTCCNT 内容等于 RTCCMP 寄存器的值时, RTCSR 寄存器中的匹配标志位 CMFLAG 将通过硬件置位并由 RTCIWEN 寄存器中相应的使能位决定发送一个中断或唤醒事件。当比较匹配事件发生时, 由 RTCCR 寄存器中的 CMPCLR 位决定 RTC 计数器复位为 0 或继续计数。例如, 如果将 RPRE[3:0] 设为 0xF, RTCCMP 设为十进制值 60, CMPCLR 位置 1, 则 CMFLAG 位将每分钟被置位 1 次。此外, 当 RTC 计数器溢出时, RTCSR 寄存器中的 OVFLAG 位将被置位。RTCSR 寄存器的读取操作会清除包括 CSECFLAG、CMFLAG 和 OVFLAG 位在内的状态标志位。

中断和唤醒控制

CK_SECOND 时钟下降沿会引起 RTCSR 寄存器中的 CSECFLAG 位置位, 如果 RTCIWEN 寄存器中的相关中断使能位 CSECIEN 置位, 还会产生中断。当相应的唤醒使能位 CSECWEN 置位时, 将产生唤醒事件来唤醒 HSI / HSE 振荡器、LDO 以及 CPU 内核。当 RTC 计数器溢出或比较匹配事件发生时, 将产生中断或唤醒事件, 这取决于 RTCIWEN 寄存器中相应的中断使能控制位或唤醒使能控制位, OVIEN / OVWEN 或 CMIEN / CMWEN。欲知详细信息, 请参考相关的寄存器定义。

RTCOUT 输出引脚配置

下表显示了由模式、极性和事件选择设置的 RTCOUT 的输出格式。

表 35. RTCOUT 输出模式和有效电平设置

ROWM	ROES	RTCOUT 输出波形	
0 (脉冲模式)	0 比较匹配	RTCCMP	----- 4 -----
		RTCCNT	----- 3 ----- 4 ----- 5 -----
		RTCOUT (ROAP = 0)	----- T_R -----
		RTCOUT (ROAP = 1)	----- \bar{T}_R -----
		ROLF	-----
	1 秒时钟	RTCCMP	----- X -----
		RTCCNT	----- 3 ----- 4 ----- 5 -----
		RTCOUT (ROAP = 0)	----- T_R ----- T_R ----- T_R -----
		RTCOUT (ROAP = 1)	----- \bar{T}_R ----- \bar{T}_R ----- \bar{T}_R -----
		ROLF	-----
1 (电平模式)	0 比较匹配	RTCCMP	----- 4 -----
		RTCCNT	----- 3 ----- 4 ----- 5 -----
		RTCOUT (ROAP = 0)	----- 高 -----
		RTCOUT (ROAP = 1)	----- 低 -----
		ROLF	----- \rightarrow -----
	1 秒时钟	RTCCMP	----- X -----
		RTCCNT	----- 3 ----- 4 ----- 5 -----
		RTCOUT (ROAP = 0)	----- 低 ----- 高 ----- 低 -----
		RTCOUT (ROAP = 1)	----- 高 ----- 低 ----- 高 -----
		ROLF	----- \rightarrow ----- \rightarrow -----

T_R : RTCOUT 输出脉冲时间 = $1 / f_{CK_RTC}$
 \rightarrow : 由软件读取 ROLF 位清零

寄存器列表

下表显示了 RTC 寄存器及其复位值。注意，此单元中的寄存器都位于 V_{DD15} 电源域中。

表 36. RTC 寄存器列表

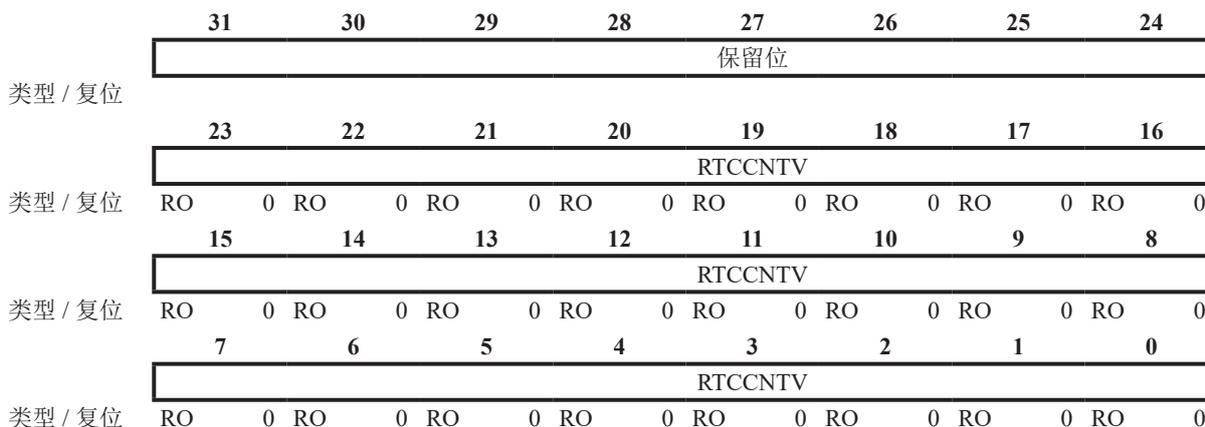
寄存器	偏移量	描述	复位值
RTCCNT	0x000	RTC 计数器寄存器	0x0000_0000
RTCCMP	0x004	RTC 比较寄存器	0x0000_0000
RTCCR	0x008	RTC 控制寄存器	0x0000_0F00
RTCSR	0x00C	RTC 状态寄存器	0x0000_0000
RTCIWEN	0x010	RTC 中断和唤醒使能寄存器	0x0000_0000

寄存器描述

RTC 计数器寄存器 – RTCCNT

该寄存器定义了一个按 CK_SECOND 时钟递增的 24-bit 向上计数器。

地址： 0x000
复位值： 0x0000_0000 (仅由 V_{DD15} 电源域复位)



位	字段	描述
[23:0]	RTCCNTV	RTC 计数器值 读取 RTCCNT 寄存器时，将返回 RTC 计数器的当前值。在 CK_SECOND 下降沿作用期间，RTCCNT 寄存器的值将被更新。该寄存器在以下情况被复位： <ul style="list-style-type: none"> - V_{DD15} 域软件复位 - 在 PWCURST 寄存器中设置 PWRST 位 - V_{DD15} 域上电复位 - POR15 - 当 CMPCLR = 1 (位于 RTCCR 寄存器) 且比较匹配发生 (RTCCNT = RTCCMP) - RTCEN 位从 0 变为 1

RTC 比较寄存器 – RTCCMP

该寄存器定义了一个与 RTC 寄存器的值进行比较的特定值。

地址： 0x004
复位值： 0x0000_0000 (仅由 V_{DD15} 域复位来复位)

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	RTCCMPV								
	RW	0	RW	0	RW	0	RW	0	RW
	15	14	13	12	11	10	9	8	
类型 / 复位	RTCCMPV								
	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
类型 / 复位	RTCCMPV								
	RW	0	RW	0	RW	0	RW	0	RW

位	字段	描述
[23:0]	RTCCMPV	RTC 比较匹配值 RTCCNT 寄存器的值与 RTCCMP 寄存器的值相等时，匹配条件发生。如果 RTCIWEN 寄存器中的 CMIEN 位被置位，则将产生中断。当匹配条件发生时，RTCSR 寄存器中的 CMFLAG 位被置位，若是此时 RTCCR 寄存器中的 CMPCLR 位为零则 RTCCNT 寄存器的值不受影响，计数器将继续计数直到溢出为止。若是 RTCCR 寄存器中的 CMPCLR 位置 1，则 RTCCNT 寄存器的值将复位到 0 然后计数器将继续计数。

RTC 控制寄存器 – RTCCR

该寄存器定义了 RTC 电路的控制位。

地址： 0x008
复位值： 0x0000_0F00 (仅由 V_{DD15} 域复位来复位)

31	30	29	28	27	26	25	24
保留位							
23	22	21	20	19	18	17	16
保留位		RC	ROLF	ROAP	ROWM	ROES	ROEN
		0 RW	0 RW	0 RW	0 RW	0 RW	0
15	14	13	12	11	10	9	8
保留位				RPRE			
				RW	1 RW	1 RW	1 RW
7	6	5	4	3	2	1	0
保留位		LSESM	CMPCLR	LSEEN	保留位	RTCSRC	RTCEN
		RW	0 RW	0 RW	0	RW	0 RW

位	字段	描述
[20]	ROLF	RTCCOUT 电平模式标志位 0: RTCCOUT 输出无效 1: RTCCOUT 输出保持在有效电平 当处于电平模式 (ROWM = 1) 且一个 RTCCOUT 输出事件发生时, 此位被硬件置位。软件读取此位会使其清零。软件读取此位后, RTCCOUT 信号将回到无效电平。
[19]	ROAP	RTCCOUT 输出有效极性 0: 高电平有效 1: 低电平有效
[18]	ROWM	RTCCOUT 输出波形模式 0: 脉冲模式 输出脉冲时间是一个 RTC 时钟 (CK_RTC) 周期。 1: 电平模式 在软件读取 ROLF 位使其清零之前, RTCCOUT 信号将保持在有效电平。
[17]	ROES	RTCCOUT 输出事件选择 0: RTC 比较匹配事件被选择 1: RTC 秒时钟 (CK_SECOND) 事件被选择 此位用于确定 RTCCOUT 信号是在发生 RTC 比较匹配事件时还是在发生 RTC 秒时钟 (CK_SECOND) 事件时被输出到 RTCCOUT 引脚。
[16]	ROEN	RTCCOUT 输出引脚使能位 0: 除能 RTCCOUT 输出引脚 1: 使能 RTCCOUT 输出引脚 当 ROEN 位设为 1 时, 一旦 RTC 比较匹配或 RTC 秒时钟 (CK_SECOND) 事件发生, RTCCOUT 信号将处于一个有效电平状态。有效极性和输出波形模式可分别由 ROAP 和 ROWM 位设置。当 ROEN 位清零时, RTCCOUT 引脚将处于浮空状态。

位	字段	描述
[11:8]	RPRE	RTC 时钟预分频器选择 $CK_SECOND = CK_RTC / 2^{RPRE}$ 0000: $CK_SECOND = CK_RTC / 2^0$ 0001: $CK_SECOND = CK_RTC / 2^1$ 0010: $CK_SECOND = CK_RTC / 2^2$... 1111: $CK_SECOND = CK_RTC / 2^{15}$
[5]	LSESM	LSE 振荡器启动模式 0: 正常启动且功耗较小 1: 快速启动但工作电流较大
[4]	CMPCLR	比较匹配计数器清零 0: 比较匹配条件发生时, 对 RTC 计数器无影响 1: 比较匹配条件发生时, RTC 计数器清零
[3]	LSEEN	LSE 振荡器使能控制位 0: LSE 振荡器除能 1: LSE 振荡器使能
[1]	RTCSRC	RTC 时钟源选择 0: LSI 振荡器作为 RTC 时钟源 1: LSE 振荡器作为 RTC 时钟源
[0]	RTCEN	RTC 使能控制位 0: RTC 除能 1: RTC 使能

RTC 状态寄存器 – RTCSR

该寄存器存储了计数器的标志位。

地址： 0x00C

复位值： 0x0000_0000 (由 V_{DD15} 域复位来复位且 RTCEN 位从 1 变为 0)

31	30	29	28	27	26	25	24	保留位				
类型 / 复位												
23	22	21	20	19	18	17	16	保留位				
类型 / 复位												
15	14	13	12	11	10	9	8	保留位				
类型 / 复位												
7	6	5	4	3	2	1	0	保留位		OVFLAG	CMFLAG	CSECFLAG
类型 / 复位												
					RC	0	RC	0	RC	0		

位	字段	描述
[2]	OVFLAG	计数器溢出标志位 0: 自上一次 RTCSR 寄存器读操作后无计数器溢出发生 1: 自上一次 RTCSR 寄存器读操作后计数器溢出发生 当计数器的值 RTCCNT 从 0xFF_FFFF 变为 0x00_0000 时, 此位将被硬件置位, 并通过软件读操作清零。建议在 RTC IRQ 程序中读取此位, 在使用软件轮询时应注意。
[1]	CMFLAG	比较匹配条件标志位 0: 自上一次 RTCSR 寄存器读操作后无比较匹配条件发生 1: 自上一次 RTCSR 寄存器读操作后比较匹配条件发生 当寄存器 RTCCNT 的值与寄存器 RTCCMP 的值相等时, 此位在 CK_SECOND 时钟下降沿被硬件置位。通过软件读取此位使其清零。建议在相应的 RTC 中断服务程序中对位访问 – 软件自由运行时, 不要使用软件轮询操作。
[0]	CSECFLAG	CK_SECOND 发生标志位 0: 自上一次 RTCSR 寄存器读操作后 CK_SECOND 没有发生 1: 自上一次 RTCSR 寄存器读操作后 CK_SECOND 发生 此位在 CK_SECOND 时钟下降沿被硬件置位。通过软件读取此位使其清零。建议在相应的 RTC 中断服务程序中对位访问 – 软件自由运行时, 不要使用软件轮询操作。

RTC 中断和唤醒使能寄存器 – RTCIWEN

该寄存器包含了中断和唤醒使能位。

地址： 0x010

复位值： 0x0000_0000 (仅由 V_{DD15} 域复位来复位)

31	30	29	28	27	26	25	24			
保留位										
23	22	21	20	19	18	17	16			
保留位										
15	14	13	12	11	10	9	8			
保留位					OVWEN	CMWEN	CSECWEN			
					RW	0	RW	0	RW	0
7	6	5	4	3	2	1	0			
保留位					OVIEN	CMIEN	CSECIEN			
					RW	0	RW	0	RW	0

位	字段	描述
[10]	OVWEN	计数器溢出唤醒使能位 0: 计数器溢出唤醒除能 1: 计数器溢出唤醒使能
[9]	CMWEN	比较匹配唤醒使能位 0: 比较匹配唤醒除能 1: 比较匹配唤醒使能
[8]	CSECWEN	计数器时钟 CK_SECOND 唤醒使能位 0: 计数器时钟 CK_SECOND 唤醒除能 1: 计数器时钟 CK_SECOND 唤醒使能
[2]	OVIEN	计数器溢出中断使能位 0: 计数器溢出中断除能 1: 计数器溢出中断使能
[1]	CMIEN	比较匹配中断使能位 0: 比较匹配中断除能 1: 比较匹配中断使能
[0]	CSECIEN	计数器时钟 CK_SECOND 中断使能位 0: 计数器时钟 CK_SECOND 中断除能 1: 计数器时钟 CK_SECOND 中断使能

17 看门狗定时器 (WDT)

简介

看门狗定时器是一个硬件定时电路，可用于检测因软件陷入死锁导致的系统锁定。看门狗定时器可工作在复位模式。当计数器向下计数到零时，看门狗定时器将产生复位。因此，软件应在看门狗定时器下溢前重新加载计数器的值。此外，如果在到达 WDT 增量值之前软件重新加载计数器，也会产生复位。其目的是看门狗定时器要防止软件死锁在不断对计数器进行加载的区段，计数器必须在看门狗定时器的值为 0 ~ WDTD 有限的时间窗口内重新加载。当处理器处于调试或休眠模式，看门狗定时器计数器可以停止。寄存器的写保护功能可以开启来防止看门狗定时器配置非预期地被任意改变。

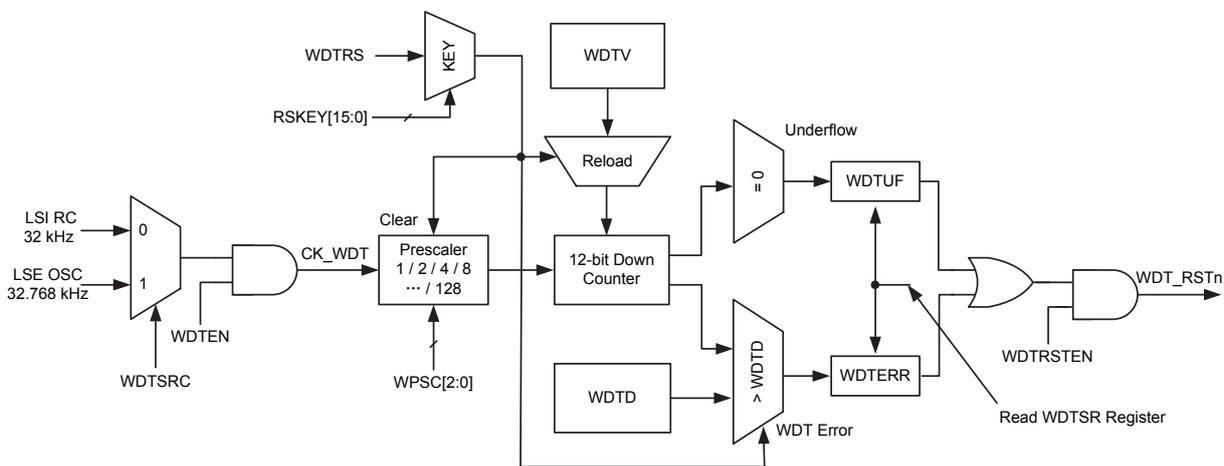


图 94. 看门狗定时器方框图

特性

- 时钟源来自于内部 32 kHz RC 振荡器 (LSI) 或外部 32.768 kHz 振荡器 (LSE)
- 进入休眠模式或深度休眠模式 1 时，可以独立设置继续运行或停止
- 带有 3-bit 预分频器的 12-bit 向下计数器结构
- 为系统提供复位信号
- 有限的重载窗口设置功能，可依需求配置重载窗口的时间长短
- 当处理器在调试时，看门狗定时器会根据 CKCU DBWDT 位的状态继续工作或停止
- 重载锁定码可以防止计数器被意外地重新加载
- 配置寄存器写保护功能用于保护计数器值、复位使能、增量值和预分频器

功能描述

看门狗定时器由一个 12-bit 向下计数器和一个固定的 3-bit 预分频器。使用 LSI 或 LSE 时钟和 1/128 最大分频值时，最长溢出周期为 16 秒。

看门狗定时器的配置包括一个可编程的计数器重载值、复位使能、窗口值和预分频值。看门狗定时器开始计数之前，这些配置都需通过 WDTMR0 和 WDTMR1 寄存器进行正确设置。为防止这些配置被意外的写入，可以通过向 WDTPR 寄存器中的 PROTECT[15:0] 位写入除 0x35CA 以外的值来使能寄存器写保护功能。在访问配置寄存器之前，向 PROTECT[15:0] 位写入 0x35CA 值，可以除能寄存器写保护功能。读取 PROTECT[0] 位可以得到寄存器写保护功能的使能 / 除能状态。

正常操作期间，看门狗定时器计数器应在下溢之前被重载，以防止看门狗复位的发生。12-bit 向下计数器可通过设置 WDTCR 寄存器中的 WDTRS 位为 1 和设置锁定码位段 (RSKEY) 为 0x5FA0，来载入所需看门狗定时器计数器的值 (WDTV)。

如果软件死锁发生在看门狗定时器重载程序里，重载操作仍将继续执行，因此软件死锁不会被看门狗定时器检测到。为了避免这一情况的发生，必须限定重载操作只能在看门狗定时器计数器的值小于或等于增量值 (WDTD) 时执行。如果看门狗定时器计数器的值大于增量值时执行重载操作，会导致看门狗定时器发生错误，如果相关功能控制使能还将产生复位。然而，通过对 WDTD 编程，使其值大于或等于 WDTV 的值，可除能上述的功能。

当看门狗定时器下溢或看门狗定时器发生错误时，WDTSR 寄存器中的 WDTERR 标志位和 WDTUF 标志位会被置位。系统复位或对 WDTSR 寄存器的写一操作会清除 WDTERR 标志位和 WDTUF 标志位。

看门狗定时器有两种时钟源：PCLK 和 CK_WDT。PCLK 时钟用于 APB 访问看门狗寄存器。CK_WDT 时钟用于看门狗定时器功能和计数。这两个时钟域之间有一些同步逻辑关系。

当系统进入休眠模式或深度休眠模式 1 时，看门狗定时器计数器是继续计数或停止取决于 WDTMR0 寄存器中的 WDTSHLT 位。而当系统进入深度休眠模式 2 时，看门狗定时器始终停止。当看门狗定时器停止计数，计数值会维持不变，使得在系统从这三种休眠模式中唤醒后能继续计数。当看门狗定时器有一个工作时钟源，看门狗复位会发生在看门狗定时器运行的任何时间。当系统进入调试模式时，看门狗定时器计数器是继续计数或停止取决于时钟控制单元 CKCU 中 MCUDBGCR 寄存器的 DBWDT 位。

看门狗定时器应按下列方式使用：

- 设置 WDTMR0 寄存器中的看门狗定时器重载值 (WDTV) 和复位使能。
- 设置 WDTMR1 寄存器中的看门狗定时器增量值 (WDTD) 和预分频器选项。
- 向 WDTCR 寄存器中写入 WDTRS = 1 和 RSKEY = 0x5FA0 来启动看门狗定时器。
- 向 WDTPR 寄存器中写入值来锁定所有的看门狗定时器寄存器，但不包括 WDTCR 和 WDTPR 寄存器。
- 看门狗定时器计数器应在计数器值小于或等于增量值 (WDTD) 时再进行重载。

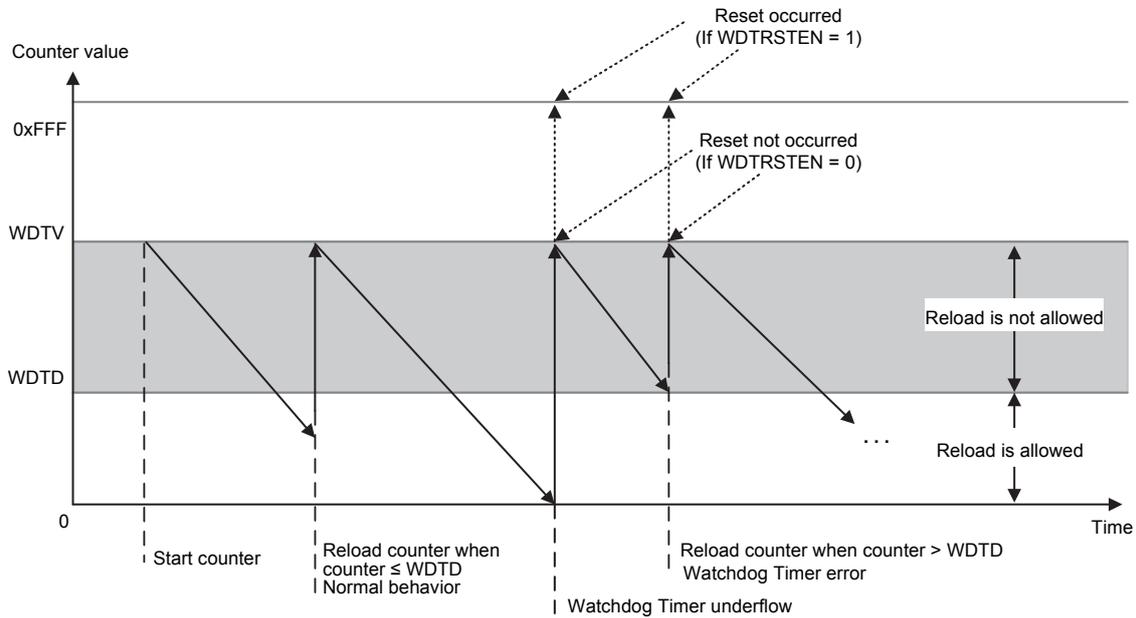


图 95. 看门狗定时器的行为

寄存器列表

下表显示了看门狗定时器寄存器及其复位值。

表 37. 看门狗定时器寄存器列表

寄存器	偏移量	描述	复位值
WDTCR	0x000	看门狗定时器控制寄存器	0x0000_0000
WDTMR0	0x004	看门狗定时器模式寄存器 0	0x0000_0FFF
WDTMR1	0x008	看门狗定时器模式寄存器 1	0x0000_5FFF
WDTSR	0x00C	看门狗定时器状态寄存器	0x0000_0000
WDTPR	0x010	看门狗定时器保护寄存器	0x0000_0000
WDTCSR	0x018	看门狗定时器时钟选择寄存器	0x0000_0000

寄存器描述

看门狗定时器控制寄存器 – WDTCR

该寄存器用来重载看门狗定时器。

偏移量: 0x000

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
	RSKEY								
类型 / 复位	WO	0	WO	0	WO	0	WO	0	WO
	23	22	21	20	19	18	17	16	
	RSKEY								
类型 / 复位	WO	0	WO	0	WO	0	WO	0	WO
	15	14	13	12	11	10	9	8	
	保留位								
类型 / 复位									
	7	6	5	4	3	2	1	0	
	保留位								WDTRS
类型 / 复位									WO
									0

位	字段	描述
[31:16]	RSKEY	看门狗定时器重载锁定码 向 RSKEY[15:0] 位写入 0x5FA0 值来开启 WDT 重载功能，写入其它值将中止写操作。
[0]	WDTRS	看门狗定时器重载 0: 无操作 1: 重载看门狗定时器 此位用来将存储在 WDTMR0 寄存器中的 WDTV 值重载入看门狗定时器计数器。此位由软件置 1 并由硬件自动清零。

看门狗定时器模式寄存器 0 – WDTMR0

该寄存器定义了看门狗定时器计数器重载值和复位使能控制。

偏移量: 0x004
复位值: 0x0000_0FFF

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位							WDTEN	
									RW 0
	15	14	13	12	11	10	9	8	
类型 / 复位	WDTSHLT		WDRSTEN	保留位	WDTV				
	RW 0	RW 0	RW 0		RW 1	RW 1	RW 1	RW 1	1
	7	6	5	4	3	2	1	0	
类型 / 复位	WDTV								
	RW 1	RW 1	RW 1	RW 1	RW 1	RW 1	RW 1	RW 1	1

位	字段	描述
[16]	WDTEN	看门狗定时器运行使能位 0: 看门狗定时器除能 1: 看门狗定时器使能运行 当看门狗定时器除能, 计数器将被重置为硬件默认状态。当 WDTEN 位被置位, 看门狗定时器将重载入 WDTV 值并向下计数。
[15:14]	WDTSHLT	看门狗定时器休眠暂停 00: 看门狗定时器在系统处于休眠模式或深度休眠模式 1 时运行 01: 看门狗定时器在系统处于休眠模式时运行, 在系统处于深度休眠模式 1 时暂停 10 或 11: 看门狗定时器在系统处于休眠模式或深度休眠模式 1 时暂停 注意, 看门狗定时器在深度休眠模式 2 时总是暂停的。进入休眠模式时, 当 WDTSHLT 位设置在休眠模式或深度休眠模式 1 下看门狗定时器停止计数, 计数值会被保留以便在系统从这三种休眠模式下唤醒后继续计数。如果看门狗定时器复位在休眠模式或深度休眠模式 1 发生时, 将会唤醒单片机。
[13]	WDRSTEN	看门狗定时器复位使能位 0: 看门狗定时器下溢或错误事件对系统复位无影响 1: 看门狗定时器下溢或错误事件触发看门狗定时器系统复位
[11:0]	WDTV	看门狗定时器计数器值 WDTV 定义了载入 12-bit 看门狗向下计数器的值。

看门狗定时器模式寄存器 1 – WDTMR1

该寄存器定义了看门狗增量值和预分频器选项。

偏移量: 0x008

复位值: 0x0000_5FFF

	31	30	29	28	27	26	25	24					
类型 / 复位	保留位												
	23	22	21	20	19	18	17	16					
类型 / 复位	保留位												
	15	14	13	12	11	10	9	8					
类型 / 复位	保留位	WPSC				WDTD							
		RW	1	RW	0	RW	1	RW	1	RW	1	RW	1
	7	6	5	4	3	2	1	0					
类型 / 复位	WDTD												
	RW	1	RW	1	RW	1	RW	1	RW	1	RW	1	

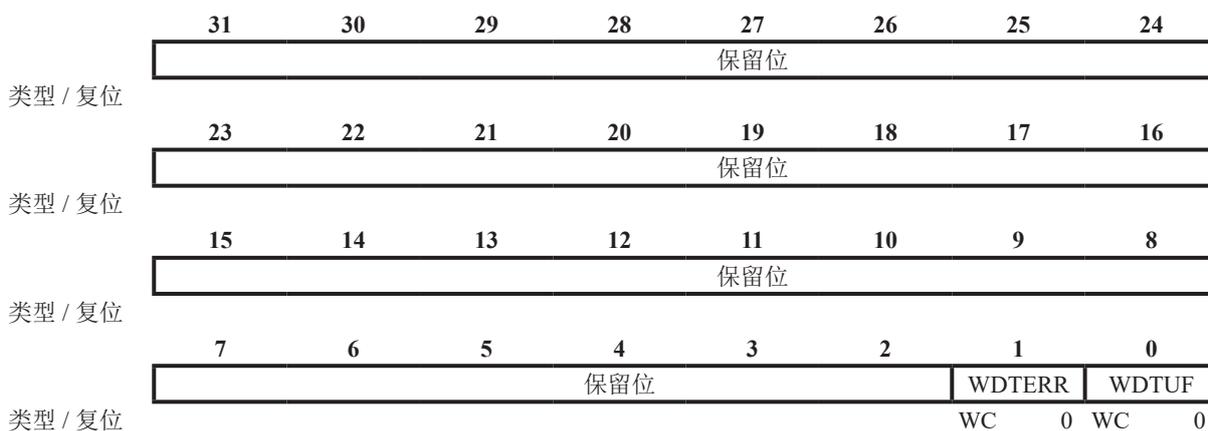
位	字段	描述
[14:12]	WPSC	看门狗定时器预分频器选项 000: 1/1 001: 1/2 010: 1/4 011: 1/8 100: 1/16 101: 1/32 110: 1/64 111: 1/128
[11:0]	WDTD	看门狗定时器增量值 WDTD 用来定义重载到看门狗定时器所允许的时间窗口范围。如果看门狗定时器计数器的值小于或等于 WDTD 的值，那么通过向 WDTCR 寄存器中写入 WDTRS = 1 和 RSKEY = 0x5FA0 会重载定时器。如果看门狗定时器计数器的值大于 WDTD 的值，那么通过向 WDTCR 寄存器中写入 WDTRS = 1 和 RSKEY = 0x5FA0 会导致看门狗定时器发生错误。此功能可通过编程使 WDTD 值大于或等于 WDTV 值来除能。

看门狗定时器状态寄存器 – WDTSR

该寄存器定义了看门狗定时器的状态。

偏移量: 0x00C

复位值: 0x0000_0000



位	字段	描述
[1]	WDTERR	看门狗定时器错误 0: 自上次对该寄存器进行读操作后无看门狗定时器错误发生 1: 自上次对该寄存器进行读操作后看门狗定时器错误发生 注: 当看门狗定时器计数器的值大于 WDTD 值时, 重载操作会导致看门狗定时器发生错误。 需要注意的是, 该位是写 1 清零标志位。
[0]	WDTUF	看门狗定时器下溢 0: 自上次对该寄存器进行读操作后无看门狗定时器下溢发生 1: 自上次对该寄存器进行读操作后看门狗定时器下溢发生 需要注意的是, 该位是写 1 清零标志位。

看门狗定时器保护寄存器 – WDTPR

该寄存器定义了看门狗定时器保护码的配置。

偏移量: 0x010

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位								
	15	14	13	12	11	10	9	8	
类型 / 复位	PROTECT								
	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
类型 / 复位	PROTECT								
	RW	0	RW	0	RW	0	RW	0	RW

位	字段	描述
[15:0]	PROTECT	看门狗定时器寄存器保护 写操作: 0x35CA: 除能看门狗定时器寄存器写保护功能 其它: 使能看门狗定时器寄存器写保护功能 读操作: 0x0000: 看门狗定时器功能除能 0x0001: 看门狗定时器功能使能 该寄存器用来除能或使能看门狗定时器配置寄存器的写保护功能。当寄存器写保护功能使能时, 除了 WDTCR 和 WDTPR 以外的所有配置寄存器将变成只读寄存器。此外, 读取 PROTECT[0] 位可以获得寄存器写保护功能的使能 / 除能状态。

看门狗定时器时钟选择寄存器 – WDTCSR

该寄存器定义了看门狗定时器时钟源选择和锁定配置。

偏移量: 0x018

复位值: 0x0000_0000

31	30	29	28	27	26	25	24
保留位							
23	22	21	20	19	18	17	16
保留位							
15	14	13	12	11	10	9	8
保留位							
7	6	5	4	3	2	1	0
保留位		WDTLOCK		保留位		WDTSRC	
		RW 0				RW 0	

位	字段	描述
[4]	WDTLOCK	看门狗定时器锁定模式 0: 该位只能由任意复位清零, 不能由软件清零 1: 该位只能由软件置位一次并锁定看门狗定时器功能 该位在任何时候都可以由软件设置为1。一旦 WDTLOCK 位被置位, 看门狗定时器的功能和寄存器都不能被修改或除能, 包括看门狗定时器的时钟源, 只有等到系统复位才能将锁定模式除能。
[0]	WDTSRC	看门狗定时器时钟源选择 0: 选择内部 32 kHz RC 振荡器时钟 (LSI) 1: 选择外部 32.768 kHz 晶体振荡器时钟 (LSE) 选择使用软件来控制看门狗定时器的时钟源。

18 内部集成电路 (I²C)

简介

I²C 模块是一个允许与外部 I²C 接口通信的内部电路，此 I²C 模块是一个符合工业标准并用于连接外部硬件的两线串行接口。这两条串行线被称为串行数据线 SDA 和串行时钟线 SCL。I²C 模块提供了三种数据传输速率：标准模式下 100 kHz、快速模式下 400 kHz 和高速模式下 1 MHz。SCL 周期产生寄存器用于设置不同的占空比以得到不同的 SCL 脉冲。

SDA 线是一条双向数据线，它连接整个 I²C 总线，在主机和从机之间用于数据的发送和接收。I²C 模块还具有仲裁检测功能，以防止多个主机试图同时传送数据到 I²C 总线的情况。

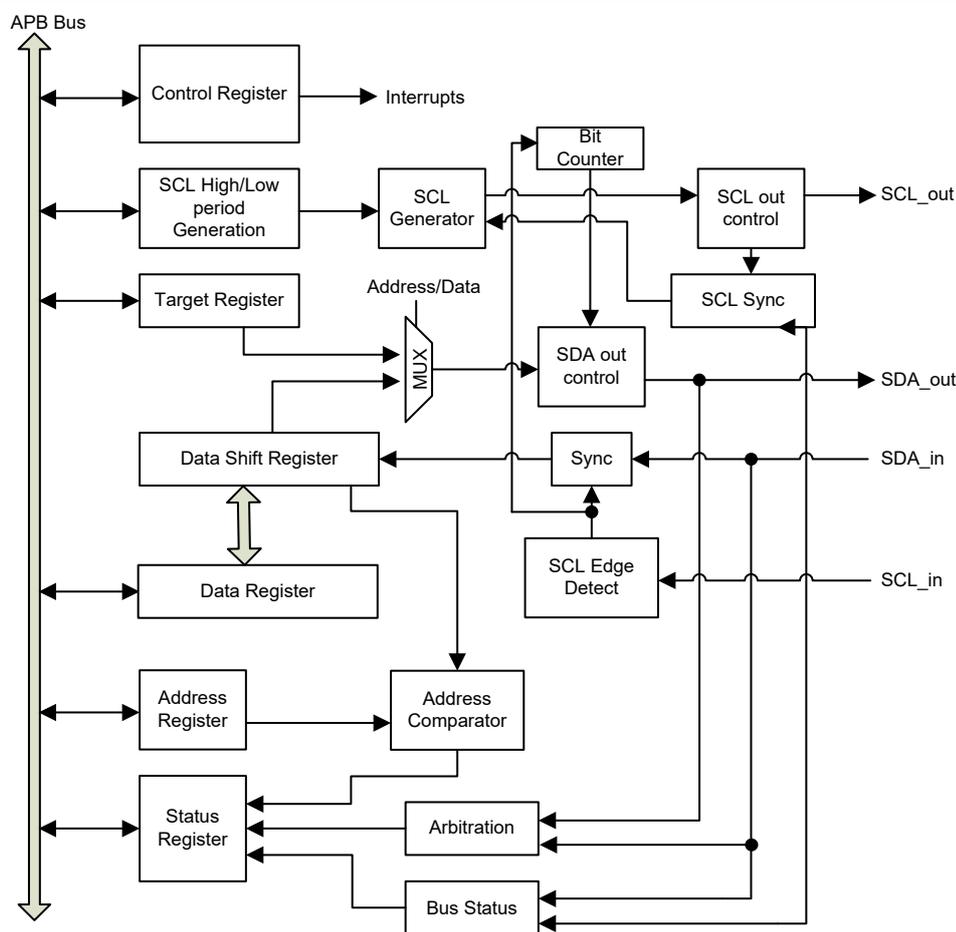


图 96. I²C 模块方框图

特性

- 双线 I²C 串行接口
 - 串行数据线 (SDA) 和串行时钟线 (SCL)
- 多种速率模式
 - 标准模式 – 100 kHz
 - 快速模式 – 400 kHz
 - 高速模式 – 1 MHz
- 主机和从机之间的双向数据传输
- 多主机总线 – 无中心主机
 - 相同的接口可作为主机或从机
- 多主机同时传输的仲裁功能使其在总线上不会产生串行数据损坏
- 时钟同步
 - 允许不同速率的设备通过一条串行总线进行通信
- 支持 7-bit 和 10-bit 寻址模式以及广播呼叫寻址
- 地址屏蔽功能使从机拥有多个地址
- 超时功能

功能描述

双线串行接口

I²C 模块有两条外部线，串行数据线 SDA 和串行时钟线 SCL，用来承载连接到总线的设备之间的信息。SCL 和 SDA 线是双向的，且必须连接一个上拉电阻。当 I²C 总线处于自由或空闲状态时，两个引脚都处于高电平状态，执行用于多个互连设备所要求的线与功能。

START 和 STOP 条件

主机可以通过发送一个 START 信号来启动传输并通过发送一个 STOP 信号来终止传输。START 信号通常被称为“S”位，定义为当 SCL 线为高电平时，SDA 线上发生从高到低的电平变化。STOP 信号通常被称为“P”位，定义为当 SCL 线为高电平时，SDA 线上从低到高的电平变化。

重复的 START，称为“Sr”位，功能上与 START 条件相同。一个重复的 START 信号允许 I²C 接口在不释放总线控制权之下与其它从机进行通信或者与同一 I²C 设备在不同传输方向上进行通信。

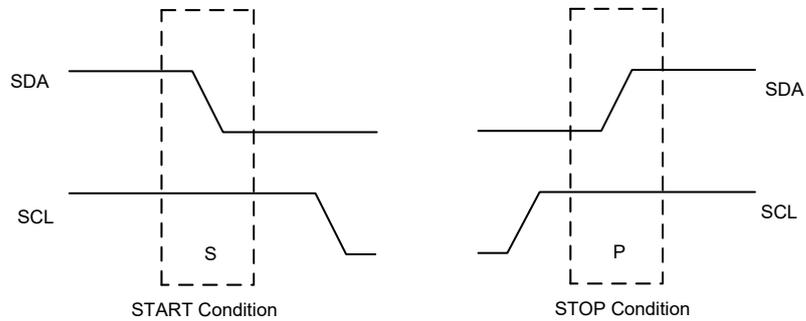


图 97. START 和 STOP 条件

数据有效性

在传送数据时, 在 SCL 时钟的高电平期间, SDA 线上的数据必须保持稳定。只有当 SCL 线上的时钟信号处于低电平状态时, SDA 数据状态才能改变。

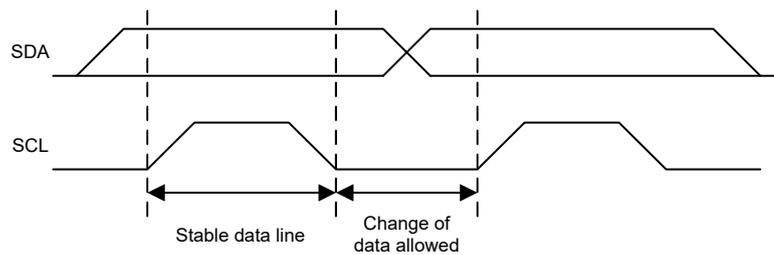


图 98. 数据有效性

寻址格式

主机发送地址确认目标从机之后, I²C 接口开始传输数据。地址帧在主机发送 START 信号后被发出。寻址模式可经由配置 I2CCR 寄存器中的寻址模式选择位 ADRM 来选择 7-bit 或 10-bit 寻址模式。

7-bit 地址格式

7-bit 地址格式由以下各部分组成: 与主机通信的 7-bit 长度的从机地址, 一个 R/\overline{W} 位和一个 ACK 位。 R/\overline{W} 位定义了数据传输方向。

$R/\overline{W} = 0$ (写): 主机发送数据到被寻址的从机。

$R/\overline{W} = 1$ (读): 主机从被寻址的从机接收数据。

从机地址可通过 I2CADDR 寄存器中的 ADDR 字段分配。如果从机地址与主机发出的地址相匹配, 从机将会返回一个确认位 (ACK)。

注意, 不允许两个从机有相同的地址。

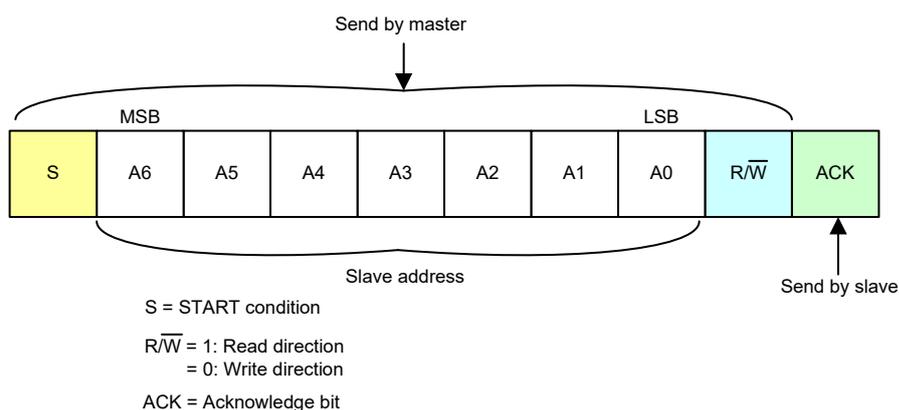


图 99. 7-bit 寻址模式

10-bit 地址格式

因 7-bit 地址范围有限, 为了防止地址冲突, 10-bit 地址格式是一种新的方案。该改进的方案与 7-bit 寻址模式一起使用, 增加了十倍左右的可用地址范围。对于 10-bit 寻址模式, START 信号后的前两个字节包括一个头字节和一个地址字节, 用来决定主机会选择哪个从机。头字节由先头的“11110”和从机地址的第 10 位和第 9 位组成。第二个字节由从机的其余 8 位地址组成。

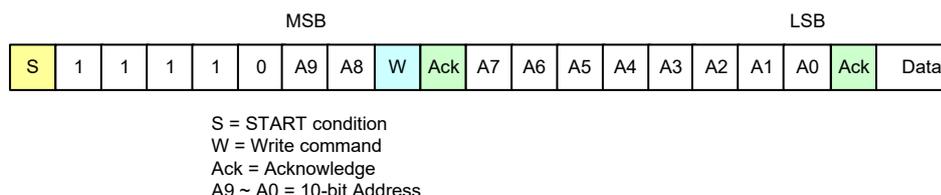


图 100. 10-bit 寻址写发送模式

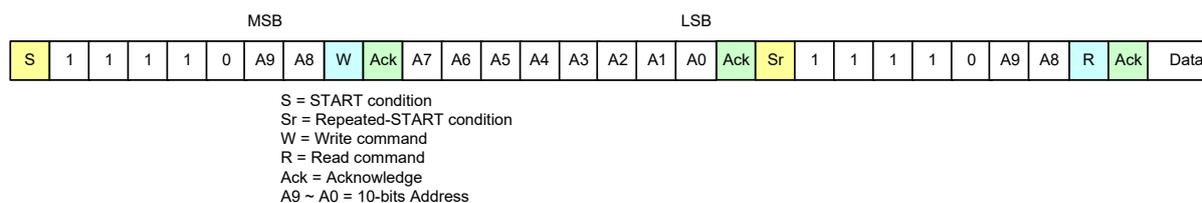


图 101. 10-bit 寻址读接收模式

数据传输和确认

一旦从机地址匹配,可以根据由 R/\bar{W} 位定义的传输方向,由从机发送或接收数据。每一个字节后面都在第 9 个 SCL 时钟上跟随一个确认位。

如果从机返回一个不确认信号 (NACK) 给主机,则主机会产生一个 STOP 信号来终止数据传输或产生一个重复 START 信号重新开始传输。

如果主机发送一个不确认信号 (NACK) 给从机,从机将释放 SDA 线给主机,主机将产生一个 STOP 信号来终止传输。

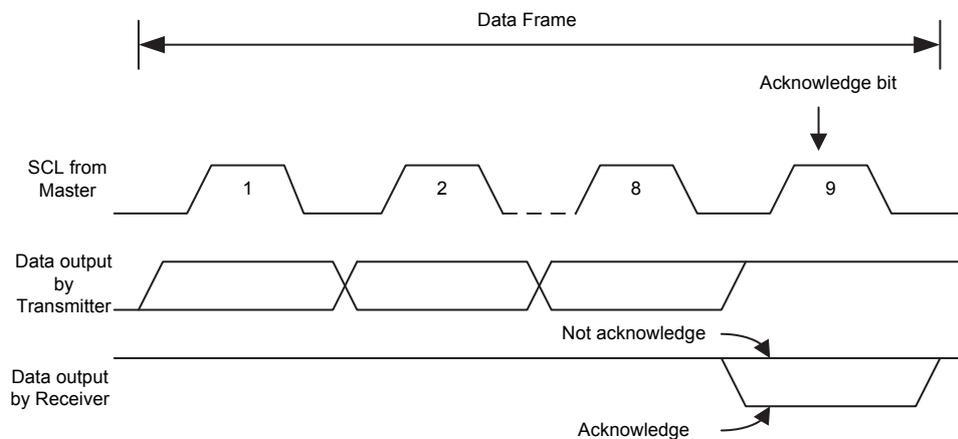


图 102. I²C 总线确认

时钟同步

正常工作时,只有一个主机可以产生 SCL 时钟。然而,当有多个主机试图产生 SCL 时钟,时钟应该同步以便比较输出数据。时钟同步可通过 I²C 接口到 SCL 线的线与连接来执行。

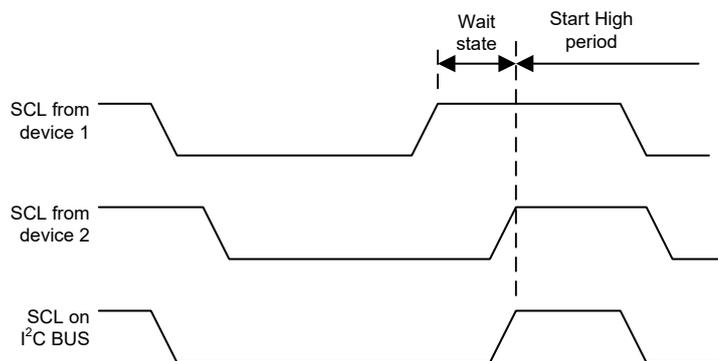


图 103. 仲裁期间时钟同步

仲裁

只有在 I²C 总线处于自由或空闲状态时, 主机才能开始传输。如果两个或两个以上的主机几乎同时产生 START 信号, 仲裁程序将会发生。

仲裁发生在 SDA 线上且可以持续很多位。仲裁程序赋予发送二进制低位 (逻辑低) 串行数据的设备更高的优先级。其它想要发送二进制高位 (逻辑高) 数据的主机将失去仲裁。只要主机输掉仲裁, I²C 模块就会将 I2CSR 寄存器中的 ARBLOS 位置位, 并且在 I2CIER 寄存器中的中断使能位 ARBLOSIE 被置为 1 时产生中断。同时, 它会停止发送数据, 并监听总线, 以检测 I²C 停止信号。当停止信号被检测到, 输掉仲裁的主机会再次尝试访问总线。

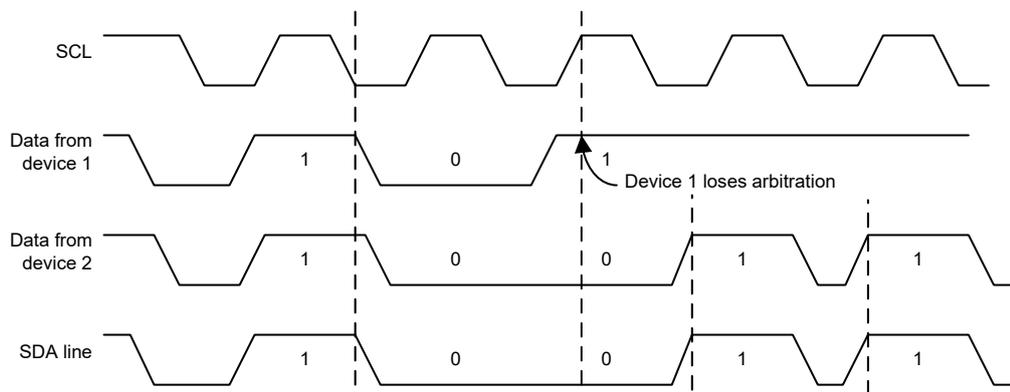


图 104. 两个主机仲裁程序

广播呼叫寻址

广播呼叫寻址功能可以用来寻址所有连接到 I²C 总线的设备。在寻址帧上, 主机可通过向 I2CTAR 寄存器中的 TAR 写入 00 和设置 RWD 位为 0 来激活广播呼叫寻址功能。

设备可通过设置相应的使能控制位 GCEN 为 1, 以使其支持广播呼叫寻址功能。如果 GCEN 位已经置 1 以支持广播呼叫寻址功能, I2CCR 寄存器中的 AA 位也应置 1, 当设备接收到值为 00H 的地址帧时将返回一个确认信号。当此条件发生时, 广播呼叫标志位 GCS 将被置 1, 但 ADRS 标志位不会被置位。

总线错误

如果数据正在 I²C 总线上传输时, 出现意外的 START 或 STOP 条件, 将会视作总线错误, 数据传输将被中止。当总线错误事件发生, I2CSR 寄存器中相应的总线错误标志位 BUSERR 将被置为 1, 且 SDA 和 SCL 线都将被释放。BUSERR 标志位应通过写 1 使其清零以使 I²C 模块进入空闲状态。

地址屏蔽使能

I²C 模块提供给用户地址屏蔽的功能, 用来决定在和主机发送来的地址帧作比较时可忽略地址的哪个位。当未屏蔽地址位和主机发送来的地址帧匹配时, ADRS 标志位被置位。注意, 此功能仅在从机模式下可用。

例如, 用户将数据传输设置为 7-bit 寻址模式, 并设置 I2CADDRMR 寄存器的值为 0x05h 和 I2CADDR 寄存器的值为 0x55h, 这意味着如果总线上 I²C 主机发送的地址等于 0x50h,

0x51h、0x54h 或 0x55h，所有 I²C 从机地址将被认为是匹配的，在地址帧匹配之后，I2CSR 寄存器的 ADRS 标志位将被置位。

地址捕获

地址捕获寄存器 I2CADDRSR 用于监控整个数据传输期间 I²C 总线上的呼叫地址，不论 I²C 模块是作为主机还是从机。注意，I2CADDRSR 寄存器是只读寄存器，每个 I²C 总线上的呼叫地址自动存储在 I2CADDRSR 寄存器，即使 I²C 设备没被寻址。

工作模式

I²C 模块可工作在以下模式：

- 主机发送器
- 主机接收器
- 从机发送器
- 从机接收器

I²C 模块默认工作在从机模式。产生 START 信号后接口将自动切换到主机模式。

主机发送器模式

开始条件

用户在设置 I2CCR 寄存器中的 I2CEN 位之后，向 I2CTAR 寄存器中写入目标从机地址和通信方向。开始条件发生之后，I2CSR 寄存器中的 STA 标志位由硬件置位。为了发送接下来的地址帧，如果 STA 标志位已被设为 1，则必须将其清零。STA 标志位通过读取 I2CSR 寄存器来清零。

地址帧

主机发送地址帧且接收到来自于地址匹配的从机的确认信号后 I2CSR 寄存器中的 ADRS 标志位被置位。为了发送接下来的数据帧，如果 ADRS 标志位已被设为 1，则必须将其清零。ADRS 标志位通过读取 I2CSR 寄存器来清零。

数据帧

要发送到从机的数据必须被写入到 I2CDR 寄存器中。

I2CSR 寄存器中的 TXDE 位被置位，表明 I2CDR 寄存器是空的，这会使 SCL 线保持在逻辑低电平状态。新数据必须被写入到 I2CDR 寄存器以继续数据的传输。写入数据到 I2CDR 寄存器将清除 TXDE 标志位。

关闭 / 继续发送

最后的数据字节发送完成后，I2CCR 寄存器中的 STOP 位被置位来中止发送或通过配置 I2CTAR 寄存器重新选定另外一个从机以开始新的传输。

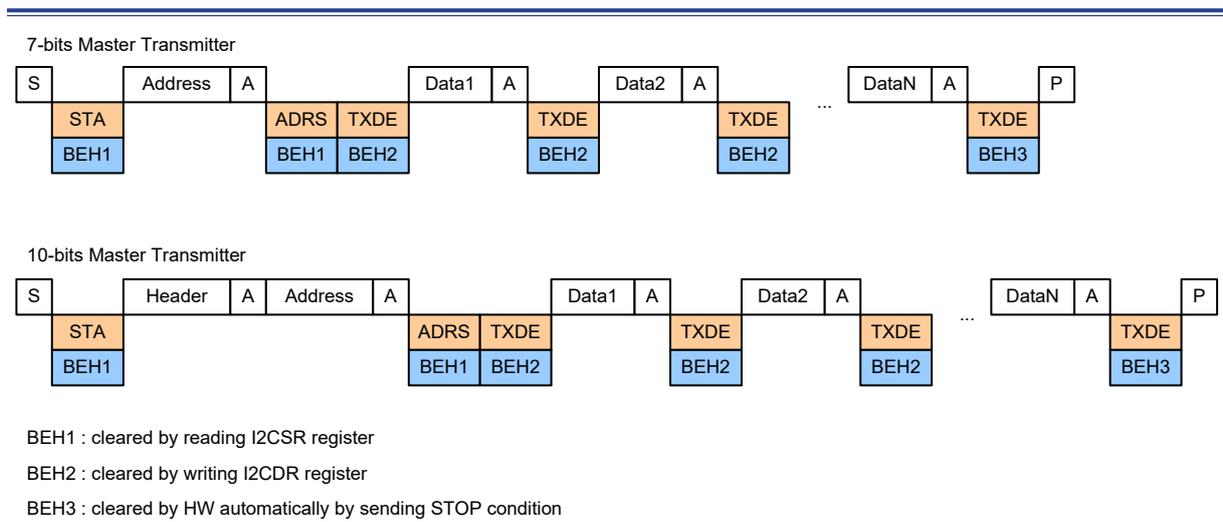


图 105. 主机发送器时序图

18 内部集成电路 (I2C)

主机接收器模式

开始条件

目标从机地址和通信方向必须被写入 I2CTAR 寄存器。在开始条件产生之后, I2CSR 寄存器中的 STA 位被置位。为了发送接下来的地址帧, 如果 STA 标志位已被设为 1, 则必须将其清零。STA 标志位通过读取 I2CSR 寄存器来清零。

地址帧

7-bit 寻址模式: 主机发送地址帧且接收到来自于地址匹配的从机的确认信号后 I2CSR 寄存器中的 ADRS 标志位被置位。为了接收接下来的数据帧, 如果 ADRS 标志位已被设为 1, 则必须将其清零。ADRS 标志位通过读取 I2CSR 寄存器来清零。

10-bit 寻址模式: 在此模式下, I2CSR 寄存器中的 ADRS 标志位会被置位两次。第一次是当 10-bit 地址被发送且接收到来自于从机的确认信号时, ADRS 位被置位。第二次是当头字节被发送且从机确认信号被接收时, ADRS 位被置位。为了接收接下来的数据帧, 如果 ADRS 标志位已被设为 1, 则必须将其清零。ADRS 标志位在读取 I2CSR 寄存器之后清零。详细的主机接收器模式时序图如下图所示。

数据帧

在主机接收器模式, 数据由从机发送。一旦数据被主机接收, I2CSR 寄存器中的 RXDNE 标志位会被置位, 但不会保持 SCL 线的状态。然而, 如果设备接收一个完整的新数据字节且 RXDNE 位已被置 1, 则 I2CSR 寄存器中的 RXBF 位将被置 1 且 SCL 线会保持在一个逻辑低的状态。当此情况发生时, 应读取 I2CDR 寄存器的数据以继续数据的传输。读取 I2CDR 寄存器后, RXDNE 标志位被清零。

关闭 / 继续发送

在传送最后一个数据字节前, 主机应使 I2CCR 寄存器中的 AA 位复位来发送一个 NACK 信号到从机。最后一个来自从机的数据字节被接收后, 一旦主机发送完一个 NACK 信号给从机, 主机将保持 SCL 线在逻辑低的状态。STOP 位被置位来终止数据传送或重新配置 I2CTAR 寄存器以开始新的传输。

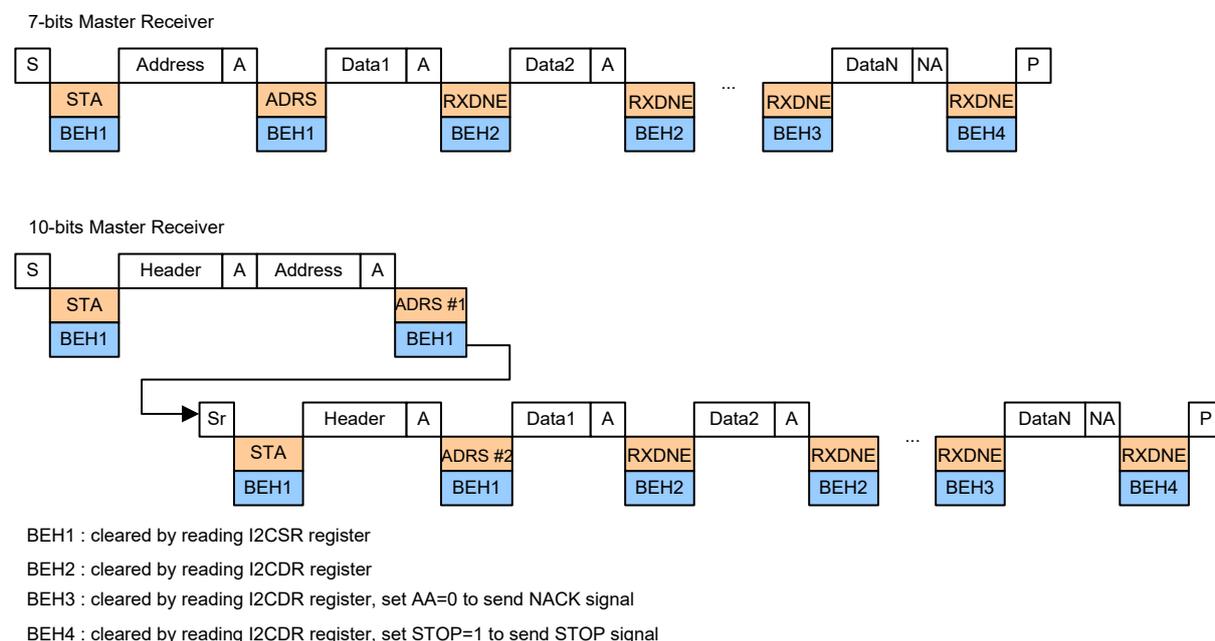


图 106. 主机接收器时序图

从机发送器模式

地址帧

7-bit 寻址模式, 在从机接收到与从机地址相匹配的呼叫地址后, I2CSR 寄存器中的 ADRS 位被置位。10-bit 寻址模式, 当第一个头字节和第二个地址字节都匹配时, ADRS 位被置位。应注意, 当第二个头字节匹配时, ADRS 位将再次被置位。在 ADRS 位已被置 1 后必须清零此位来继续数据的传输。读取 I2CSR 寄存器后, ADRS 位被清零。

数据帧

在从机发送器模式, TXDE 位被置位表明 I2CDR 是空的, 这将导致 SCL 线处于一个逻辑低的状态。此时新的发送数据必须写入到 I2CDR 寄存器以继续数据的传输。向 I2CDR 中写入一个数据将使 TXDE 位清零。

接收不确认信号

当从机接收到一个不确认信号时, I2CSR 寄存器中的 RXNACK 位被置位, 但不会保持 SCL 线的状态。向 RXNACK 写入 1 会使 RXNACK 标志位清零。

STOP 条件

当从机检测到一个 STOP 条件时, I2CSR 寄存器中的 STO 位会被置位以表明 I²C 接口发送结束。读取 I2CSR 寄存器可以使 STO 标志位清零。

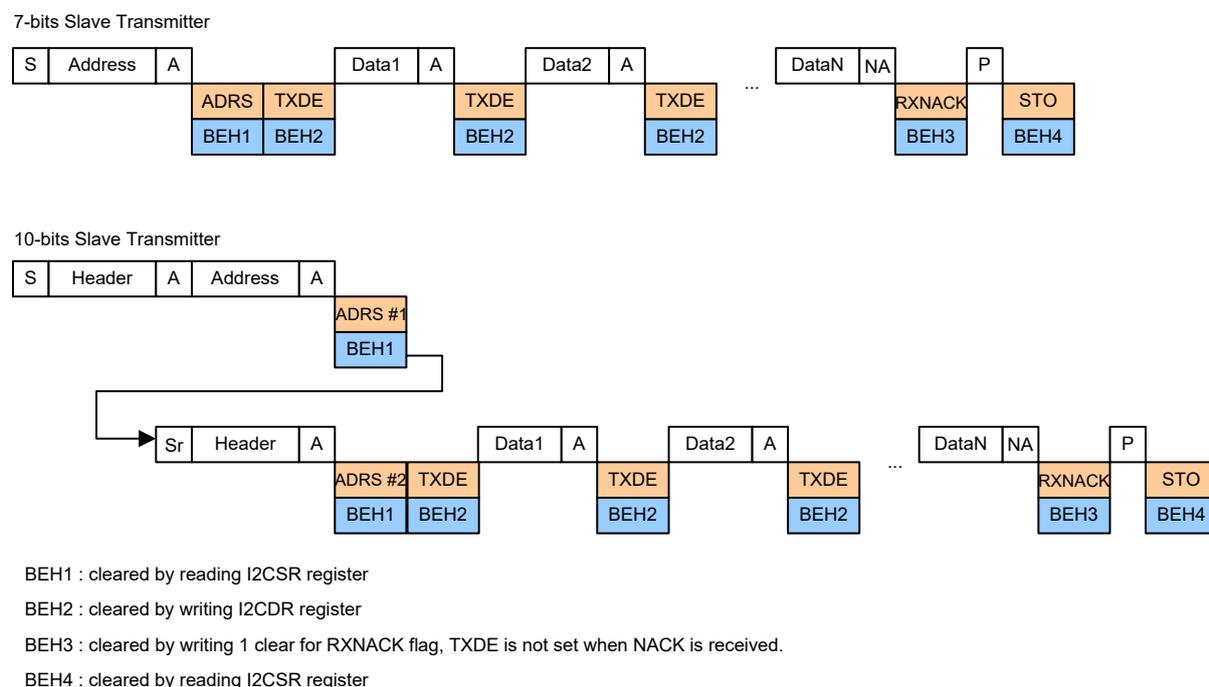


图 107. 从机发送器时序图

从机接收器模式

地址帧

在从机接收到与从机地址相匹配的呼叫地址后, I2CSR 寄存器中的 ADRS 位被置位。在 ADRS 位已被置 1 后必须将其清零来继续数据的传输。读取 I2CSR 寄存器之后, ADRS 标志位被清零。

数据帧

在从机接收模式, 数据从主机发送。一旦一个数据字节被从机接收, I2CSR 寄存器中的 RXDNE 标志位被置位, 但不会保持 SCL 线的状态。然而, 如果设备接收一个完整的新数据字节且 RXDNE 位被置 1, 则 I2CSR 寄存器中的 RXBF 位将被置 1 且 SCL 线会保持在一个逻辑低的状态。当此情况发生时, 应读取 I2CDR 寄存器中的数据来继续数据的传输。读取 I2CDR 寄存器之后, RXDNE 标志位被清零。

STOP 条件

当从机检测到 STOP 条件时, I2CSR 寄存器中的 STO 标志位会被置位以表明 I²C 接口发送结束。读取 I2CSR 寄存器可以使 STO 位清零。

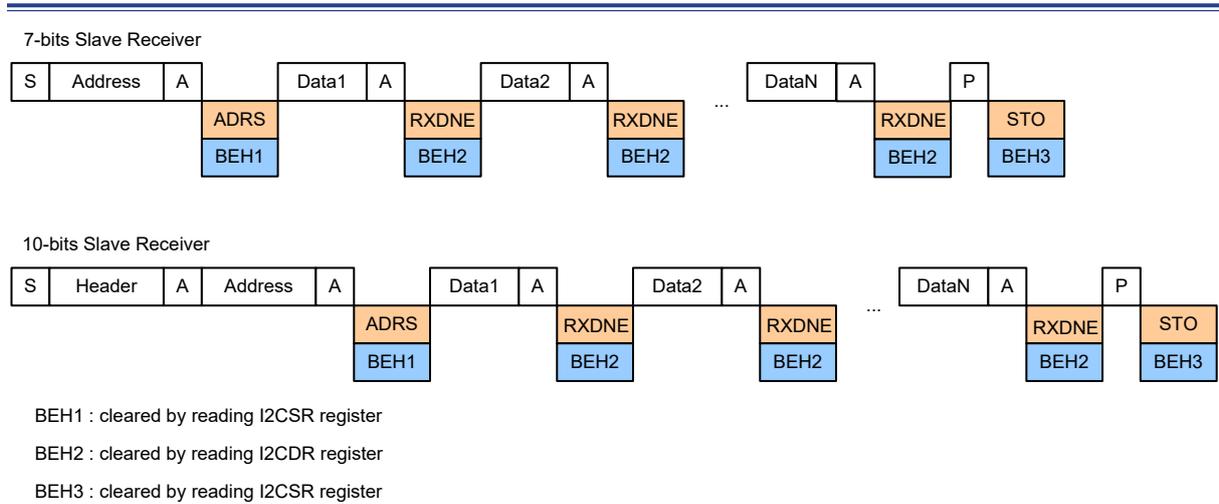


图 108. 从机接收器时序图

保持 SCL 线状态的条件

下列条件将使 SCL 线由硬件保持在一个逻辑低电平状态，从而使得所有的 I²C 传输停止。数据传输将在下列条件消除后继续进行。

表 38. 保持 SCL 线状态的条件

类型	条件	描述	消除条件
标志位	TXDE	I ² C 模块工作在发送模式，需要向 I2CDR 寄存器写入要发送的数据。 (注：接收到 NACK 信号后，TXDE 位将不会被置位。)	主机情况： 写数据到 I2CDR 寄存器 配置 TAR 置位 STOP 从机情况： 写数据到 I2CDR 寄存器
	GCS	I ² C 模块作为从机通过广播呼叫被寻址。	读取 I2CSR 寄存器
	ADRS	主机： I ² C 模块发送地址帧并接收从从机返回的一个 ACK 信号 (注：请参考图 105 和图 106) 从机： I ² C 模块作为从机被寻址 (注：请参考图 107 和图 108)	读取 I2CSR 寄存器
	STA	主机发送 START 信号	读取 I2CSR 寄存器
	RXBF	设备接收完整的新数据，同时 RXDNE 标志位已被置位。	读取 I2CDR 寄存器
事件	主机接收 NACK	无论是地址还是数据帧，一旦接收到 NACK 信号，SCL 线将保持在主机模式。	配置 TAR 置位 STOP
	主机在接收模式发送 NACK	在主机接收模式接收最后一个数据字节时发生。 (注：请参考图 106。此情况下，RXNACK 标志将不会被置位。)	配置 TAR 置位 STOP

I²C 超时功能

超时功能可减少由于接收错误导致 SCL 线持续被拉低造成 I²C 被锁住的问题。如果在超时周期里 I²C 总线时钟 SCL 一直保持在一个逻辑低的状态, 相关 I²C 超时标志位将被置位。超时周期是由一个带有可编程预载值的 16-bit 向下计数型计数器决定的。该超时计数器由 I²C 超时时钟 f_{I2CTO} 驱动, 时钟频率是由 I2CTOUT 寄存器的超时预分频器字段定义的。I2CTOUT 寄存器的 TOUT 字段用于定义超时计数器预载值。通过设置 I2CCR 寄存器的 ENTOUT 位来使能超时功能。当 ENTOUT 位被置为 1 且以下其中一种情况发生时, 超时计数器将从预载值开始向下计数:

- I²C 主机模块发送 START 信号。
- I²C 从机模块检测到 START 信号。
- RXBF、TXDE、RXDNE、RXNACK、GCS 或 ADRS 标志位被置位。

ENTOUT 位清零时, 超时计数器将停止计数。然而, 当下列所示任一情况发生时, 计数器也会停止计数:

- I²C 从机模块没被寻址。
- I²C 从机模块检测到 STOP 信号。
- I²C 主机模块发送 STOP 信号。
- I2CSR 寄存器的 ARBLOS 或 BUSERR 标志位被置位。

如果超时计数器下溢, I2CSR 寄存器的相关超时标志位 TOUTF 将会被置为 1, 若相关中断使能将产生超时中断。

寄存器列表

下表所示为 I²C 寄存器及其复位值。

表 39. I²C 寄存器列表

寄存器	偏移量	描述	复位值
I2CCR	0x000	I ² C 控制寄存器	0x0000_2000
I2CIER	0x004	I ² C 中断使能寄存器	0x0000_0000
I2CADDR	0x008	I ² C 地址寄存器	0x0000_0000
I2CSR	0x00C	I ² C 状态寄存器	0x0000_0000
I2CSHPGR	0x010	I ² C SCL 高电平周期发生寄存器	0x0000_0000
I2CSLPGR	0x014	I ² C SCL 低电平周期发生寄存器	0x0000_0000
I2CDR	0x018	I ² C 数据寄存器	0x0000_0000
I2CTAR	0x01C	I ² C 目标寄存器	0x0000_0000
I2CADDRMR	0x020	I ² C 地址屏蔽寄存器	0x0000_0000
I2CADDRSR	0x024	I ² C 地址捕获寄存器	0x0000_0000
I2CTOUT	0x028	I ² C 超时寄存器	0x0000_0000

寄存器描述

I²C 控制寄存器 – I2CCR

该寄存器定义了相应的 I²C 功能使能控制位。

偏移量: 0x000 (0)
复位值: 0x0000_2000

31	30	29	28	27	26	25	24					
保留位												
23	22	21	20	19	18	17	16					
保留位												
15	14	13	12	11	10	9	8					
SEQFILTER		COMBFILTEREN		ENTOUT		保留位						
RW	0	RW	0	RW	1	RW	0					
7	6	5	4	3	2	1	0					
ADRM		保留位			I2CEN	GCEN	STOP	AA				
RW	0				RW	0	RW	0	RW	0	RW	0

位	字段	描述
[15:14]	SEQFILTER	SDA 或 SCL 输入连续滤波器配置位 00: 连续滤波器除能 01: 1 个 PCLK 干扰滤波器 1x: 2 个 PCLK 干扰滤波器 注: 该设置将影响 SCL 的频率。详细的描述在 I2CSLPGR 寄存器部分。
[13]	COMBFILTEREN	SDA 或 SCL 输入组合滤波器使能位 0: 组合滤波器除能 1: 组合滤波器使能

位	字段	描述
[12]	ENTOUT	I ² C 超时功能使能控制位 0: 超时功能除能 1: 超时功能使能 该位用于使能或除能 I ² C 超时功能。当 I2CEN 位清零, ENTOUT 位将由硬件自动清零。建议用户在超时计数器设置 ENOUT 位为 1 开始计数之前要配置好 I2CTOUT 寄存器的 PSC 和 TOUT 字段。
[7]	ADRM	寻址模式 0: 7-bit 寻址模式 1: 10-bit 寻址模式 当 I ² C 主机 / 从机模块工作在 7-bit 寻址模式时, 它只能发出和响应一个 7-bit 地址, 反之亦然。当 I2CEN 除能, ADRM 位会由硬件自动清零。
[3]	I2CEN	I ² C 接口使能位 0: I ² C 接口除能 1: I ² C 接口使能
[2]	GCEN	广播呼叫使能位 0: 广播呼叫功能除能 1: 广播呼叫功能使能 当设备接收到 0x00 的呼叫地址且 GCEN 和 AA 位都被置 1, 则 I ² C 接口将作为一个从机且 I2CSR 寄存器中的 GCS 位将被置 1。当 I2CEN 位被清零时, GCEN 位由硬件自动清零。
[1]	STOP	STOP 条件控制位 0: 无动作 1: 在主机模式下发送 STOP 条件 此位被软件置 1 来产生一个 STOP 条件, 通过硬件自动清零。STOP 位只用于主机。
[0]	AA	确认位 0: 在接收到一个字节后发送一个不确认信号 (NACK) 1: 在接收到一个字节后发送一个确认信号 (ACK) 当 I2CEN 位清零, AA 位由硬件自动清零。

I²C 中断使能寄存器 – I2CIER

该寄存器定义了相应的 I²C 中断使能位。

偏移量: 0x004
复位值: 0x0000_0000

31	30	29	28	27	26	25	24			
保留位										
23	22	21	20	19	18	17	16			
保留位					RXBFIE	TXDEIE	RXDNEIE			
					RW	0	RW	0	RW	0
15	14	13	12	11	10	9	8			
保留位					TOUTIE	BUSERRIE	RXNACKIE	ARBLOSIE		
					RW	0	RW	0	RW	0
7	6	5	4	3	2	1	0			
保留位					GCSIE	ADRSIE	STOIE	STAIE		
					RW	0	RW	0	RW	0

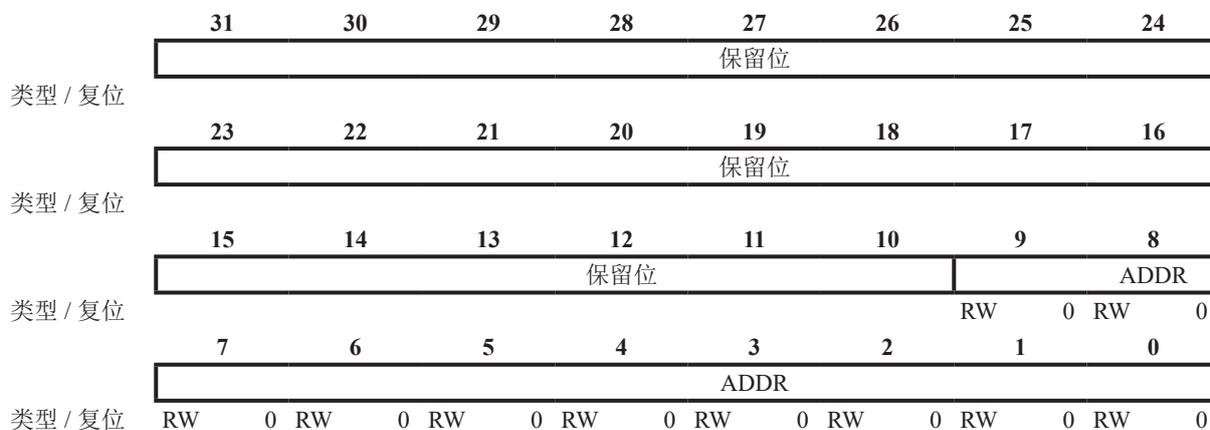
位	字段	描述
[18]	RXBFIE	RX 缓冲器已满中断使能位 0: 中断除能 1: 中断使能 当 I2CCR 寄存器中的 I2CEN 位清零时, 此位将由硬件清零。
[17]	TXDEIE	发送器模式下数据寄存器空中断使能位 0: 中断除能 1: 中断使能 当 I2CCR 寄存器中的 I2CEN 位清零时, 此位将由硬件清零。
[16]	RXDNEIE	接收器模式下数据寄存器非空中断使能位 0: 中断除能 1: 中断使能 当 I2CCR 寄存器中的 I2CEN 位清零时, 此位将由硬件清零。
[11]	TOUTIE	超时中断使能位 0: 中断除能 1: 中断使能 I2CCR 寄存器中的 I2CEN 位清零时, 此位将由硬件清零。
[10]	BUSERRIE	总线错误中断使能位 0: 中断除能 1: 中断使能 I2CCR 寄存器中的 I2CEN 位清零时, 此位将由硬件清零。
[9]	RXNACKIE	接收不确认信号中断使能位 0: 中断除能 1: 中断使能 I2CCR 寄存器中的 I2CEN 位清零时, 此位将由硬件清零。
[8]	ARBLOSIE	I ² C 多主机模式下仲裁丢失中断使能位 0: 中断除能 1: 中断使能 I2CCR 寄存器中的 I2CEN 位清零时, 此位将由硬件清零。

位	字段	描述
[3]	GCSIE	广播呼叫从机中断使能位 0: 中断除能 1: 中断使能 当 I2CCR 寄存器中的 I2CEN 位清零时, 此位将由硬件清零。
[2]	ADRSIE	从机地址匹配中断使能位 0: 中断除能 1: 中断使能 当 I2CCR 寄存器中的 I2CEN 位清零时, 此位将由硬件清零。
[1]	STOIE	STOP 条件检测中断使能位 0: 中断除能 1: 中断使能 当 I2CCR 寄存器中的 I2CEN 位清零时, 此位将由硬件清零。该位仅用在 I ² C 从机模式。
[0]	STAIE	START 条件发送中断使能位 0: 中断除能 1: 中断使能 当 I2CCR 寄存器中的 I2CEN 位清零时, 此位将由硬件清零。该位仅用在 I ² C 主机模式。

I²C 地址寄存器 – I2CADDR

该寄存器定义了 I²C 设备地址。

偏移量: 0x008
复位值: 0x0000_0000



位	字段	描述
[9:0]	ADDR	设备地址 该寄存器定义了 I ² C 设备地址。当 I ² C 设备用在 7-bit 寻址模式时, 只有 ADDR[6:0] 位与 I ² C 主机发送的地址相比较。

I²C 状态寄存器 – I2CSR

该寄存器包含了 I²C 的工作状态。

偏移量: 0x00C
复位值: 0x0000_0000

31	30	29	28	27	26	25	24
保留位							
23	22	21	20	19	18	17	16
保留位		TXNRX	MASTER	BUSBUSY	RXBF	TXDE	RXDNE
		RO	0 RO	0 RO	0 RO	0 RO	0 RO
15	14	13	12	11	10	9	8
保留位				TOUTF	BUSERR	RXNACK	ARBLOS
				WC	0 WC	0 WC	0 WC
7	6	5	4	3	2	1	0
保留位				GCS	ADRS	STO	STA
				RC	0 RC	0 RC	0 RC

位	字段	描述
[21]	TXNRX	发送器 / 接收器模式 0: 接收器模式 1: 发送器模式 只读位。
[20]	MASTER	主机模式 0: I ² C 在从机模式或空闲 1: I ² C 在主机模式 当 I2CTAR 寄存器被赋值且 I ² C 总线空闲时, I ² C 接口将切换作为 I ² C 总线上的主机。 当通过软件将 I2CEN 位清零或发送一个 STOP 条件给 I ² C 总线或检测出 I ² C 总线错误时, MASTER 位将由硬件清零。此位通过硬件置位或清零, 且为只读位。
[19]	BUSBUSY	总线忙 0: I ² C 总线空闲 1: I ² C 总线忙 当通过将 I2CEN 位置 1 使能 I ² C 接口时, I ² C 接口硬件开始检测 I ² C 总线的状态。当 SDA 或 SCL 信号被检测到一个逻辑低状态时此位会被置 1, 当 STOP 条件被检测到时此位会被清零。
[18]	RXBF	接收器模式下缓冲器已满标志位 0: 数据缓冲器未满 1: 数据缓冲器已满 当数据寄存器 I2CDR 已经存储了一个数据字节, 同时数据移位寄存器也已经接收到了一个完整的新数据字节, 此位会被置位。RXBF 位通过软件读取 I2CDR 寄存器来清零。
[17]	TXDE	发送器模式下数据寄存器为空 0: 数据寄存器 I2CDR 非空 1: 数据寄存器 I2CDR 为空 在发送器模式下, 当 I2CDR 寄存器为空时, 此位被置位。需注意的是, 在地址帧被发送后此位会被硬件置位, 此时后需要发送的数据应该被载入到 I2CDR 寄存器。此位可通过以下方式清零: 在主机和从机模式下通过软件向 I2CDR 寄存器写入数据使其清零; 在主机模式下发送 STOP 信号, 终止数据传输后由硬件自动清零; 主机模式下设置 I2CTAR 寄存器重新开始新的数据传输使其清零。

位	字段	描述
[16]	RXDNE	接收器模式下数据寄存器非空 0: 数据寄存器 I2CDR 为空 1: 数据寄存器 I2CDR 非空 在接收器模式下, 当 I2CDR 寄存器不为空时, 此位被置位。软件读取 I2CDR 寄存器中的数据字节使 RXDNE 位清零。
[11]	TOUTF	超时计数器下溢标志位 0: 无超时计数器下溢发生 1: 超时计数器下溢发生 此位写入 1 将清零 TOUTF 标志位。
[10]	BUSERR	总线错误标志位 0: 无总线错误发生 1: 总线错误发生 在传输过程中, 当 I ² C 接口检测到一个在不当时机发生的 START 或 STOP 条件时, 此位将被置位。此位写入 1 将使 BUSERR 标志位清零。 在主机模式下: 一旦总线错误事件发生, SDA 线和 SCL 线都将通过硬件被释放且 BUSERR 位被置位。软件必须在下一个地址字节被发送之前清除 BUSERR 标志位。 在从机模式下: 一旦从机检测到一个在不当时机发生的 START 或 STOP 条件时, 软件必须在下一个地址字节被接收之前清除 BUSERR 标志位。
[9]	RXNACK	接收到 NACK 信号标志位 0: 从接收器返回 ACK 信号 1: 从接收器返回 NACK 信号 RXNACK 位表明在主机或从机发送器模式下接收到 NACK 信号。向此位写入 1 来清除 RXNACK 标志位。
[8]	ARBLOS	仲裁丢失标志位 0: 无仲裁丢失被检测到 1: 位仲裁丢失被检测到 在地址或数据帧发送过程中, 当 I ² C 接口在主机模式下于当前的时钟周期输掉一个主机总线仲裁时, 此位由硬件置位。向此位写入 1 来清除 ARBLOS 标志位。一旦 ARBLOS 标志位由硬件置位, 则必须在下一次发送之前清除此标志位。
[3]	GCS	广播呼叫从机标志位 0: 无广播呼叫从机发生 1: I ² C 接口通过广播呼叫命令被寻址 在 7-bit 寻址模式或 10-bit 寻址模式下, I ² C 接口收到 0x00 或 0x000 的地址时, 如果 GCEN 和 AA 位都被置为 1, 则将切换为广播呼叫从机。此标志位在被读取后自动清零。
[2]	ADRS	地址发送 (主机模式) / 地址接收 (从机模式) 标志位 主机模式下地址发送 0: 地址帧没被发送 1: 地址帧已被发送 对于 7-bit 寻址模式, 此位在主机接收到从机发送的地址帧确认后被置位。对于 10-bit 寻址模式, 此位在接收到第一个头字节和第二个地址字节的确认后被置位。 从机模式下地址匹配 0: I ² C 接口没有被寻址 1: I ² C 接口作为从机被寻址 当 I ² C 接口接收到与 I2CADDR 寄存器里定义地址相匹配的呼叫地址, 且 I2CCR 寄存器中的 AA 位被置 1 时, 将切换到从机模式。此标志位在读取 I2CSR 寄存器后被清零。
[1]	STO	STOP 条件检测标志位 0: 无 STOP 条件被检测 1: 从机模式下 STOP 条件被检测 此位只在从机模式下可用, 且在读取 I2CSR 寄存器后自动清零。

位	字段	描述
[0]	STA	START 条件发送标志位 0: 无 START 条件被发送 1: 在主机模式 START 条件被发送 此位只在主机模式下可用, 且在读取 I2CSR 寄存器后自动清零。

I²C SCL 高电平周期发生寄存器 – I2CSHPGR

该寄存器定义了 I²C SCL 时钟高电平时间长短。

偏移量: 0x010
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24										
类型 / 复位	保留位																	
	23	22	21	20	19	18	17	16										
类型 / 复位	保留位																	
	15	14	13	12	11	10	9	8										
类型 / 复位	SHPG																	
	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0										
类型 / 复位	SHPG																	
	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0

位	字段	描述
[15:0]	SHPG	SCL 时钟高电平时间产生 高电平周期持续时间设置 $SCL_{HIGH} = T_{PCLK} \times (SHPG + d)$ 这里 T_{PCLK} 是 APB 总线外设时钟 (PCLK) 周期, d 值取决于 I ² C 控制寄存器 (I2CCR) 中 SEQFILTER 的设置。 若 SEQFILTER = 00, d = 6 若 SEQFILTER = 01, d = 8 若 SEQFILTER = 10 或 11, d = 9

I²C SCL 低电平周期产生寄存器 – I2CSLPGR

该寄存器定义了 I²C SCL 时钟低电平时间长短。

偏移量: 0x014

复位值: 0x0000_0000

31	30	29	28	27	26	25	24	保留位							
类型 / 复位															
23	22	21	20	19	18	17	16	保留位							
类型 / 复位															
15	14	13	12	11	10	9	8	SLPG							
类型 / 复位															
RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0
7	6	5	4	3	2	1	0	SLPG							
类型 / 复位															
RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0

位	字段	描述
[15:0]	SLPG	<p>SCL 时钟低电平时间产生</p> <p>低电平周期持续时间设置 $SCL_{LOW} = T_{PCLK} \times (SLPG + d)$</p> <p>这里 T_{PCLK} 是 APB 总线外设时钟 (PCLK) 周期, d 值取决于 I²C 控制寄存器 (I2CCR) 中 SEQFILTER 的设置。</p> <p>若 SEQFILTER = 00, d = 6</p> <p>若 SEQFILTER = 01, d = 8</p> <p>若 SEQFILTER = 10 或 11, d = 9</p>

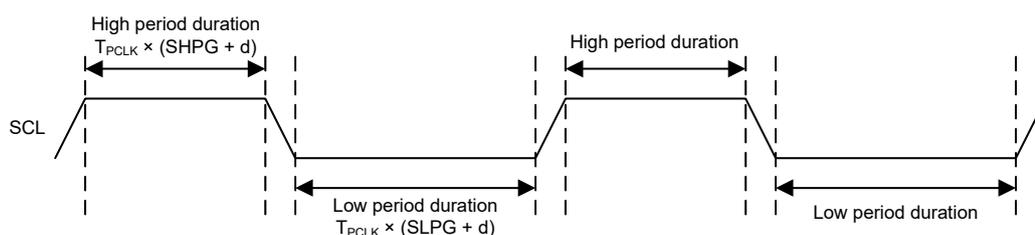


图 109. SCL 时序图

表 40. I²C 时钟设置范例

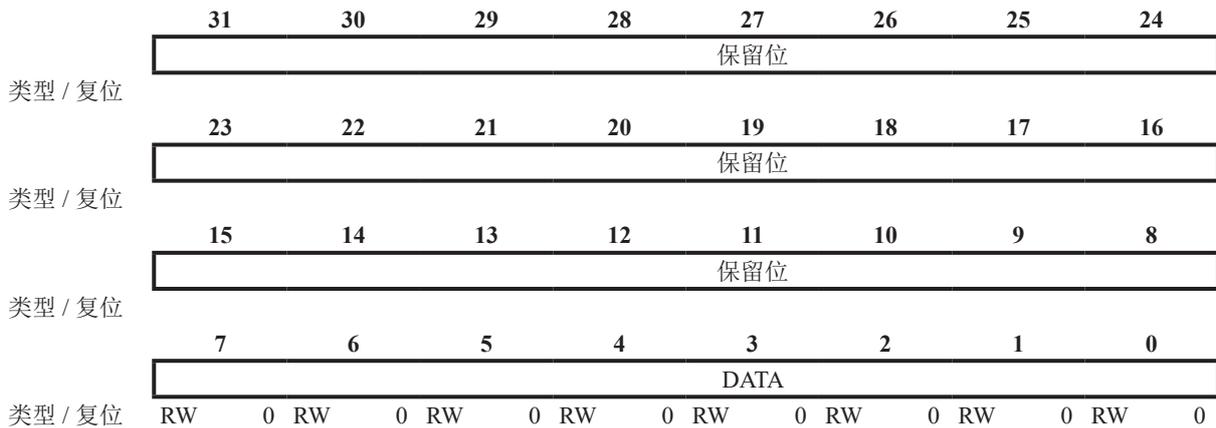
I ² C 时钟	$T_{SCL} = T_{PCLK} \times [(SHPG + d) + (SLPG + d)]$ (d = 6) PCLK 时钟下 SHPG + SLPG 的值	
	10 MHz	20 MHz
100 kHz (标准模式)	88	188
400 kHz (快速模式)	13	38
1 MHz (高速模式)	N/A	8

I²C 数据寄存器 – I2CDR

该寄存器定义了由 I²C 模块发送和接收的数据。

偏移量: 0x018

复位值: 0x0000_0000



位	字段	描述
[7:0]	DATA	I ² C 数据寄存器 在发送器模式中，发送到从机的一个数据字节可以配置给这些位。如果软件配置新的数据给 I2CDR 寄存器时，TXDE 标志位将被清零。在接收器模式中，一个数据字节从 MSB 到 LSB 逐位的通过 I ² C 接口被接收且被储存在数据移位寄存器中。一旦接收到了确认位，当 RXDNE 标志位为 0 时，数据移位寄存器的值将被发送到 I2CDR 寄存器。

I²C 目标寄存器 – I2CTAR

该寄存器定义了要与之通信的目标设备地址。

偏移量: 0x01C

复位值: 0x0000_0000

31	30	29	28	27	26	25	24			
保留位										
23	22	21	20	19	18	17	16			
保留位										
15	14	13	12	11	10	9	8			
保留位					RWD	TAR				
					RW	0	RW	0	RW	0
7	6	5	4	3	2	1	0			
TAR										
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0

位	字段	描述
[10]	RWD	读或写方向位 0: 写入目标从机地址 1: 读取目标从机地址 如果在 10-bit 主机接收器模式此位被置 1, 那么 I ² C 接口将使第一个头帧中的字节初始化为 b11110XX0, 并由硬件自动发送一个值为 b11110XX1 的字节给第二个头帧。
[9:0]	TAR	目标从机地址 一旦数据写入该寄存器, I ² C 接口将会自动发送一个 START 信号和依据写入数据发送一个目标从机地址。当系统想要发送一个重复的 START 信号给 I ² C 总线时, 建议在一个字节传输完成之后再设置 I2CTAR 寄存器。不允许在地址帧设置 TAR。I2CTAR[9:7] 在 7-bit 寻址模式中不可用。

I²C 地址屏蔽寄存器 – I2CADDR

该寄存器定义了 I²C 地址哪个位被屏蔽且不与接收到的地址帧相关位作比较。

偏移量: 0x020

复位值: 0x0000_0000

31	30	29	28	27	26	25	24		
保留位									
23	22	21	20	19	18	17	16		
保留位									
15	14	13	12	11	10	9	8		
保留位						ADDR			
						RW	0	RW	0
7	6	5	4	3	2	1	0		
ADDR									
RW	0	RW	0	RW	0	RW	0	RW	0

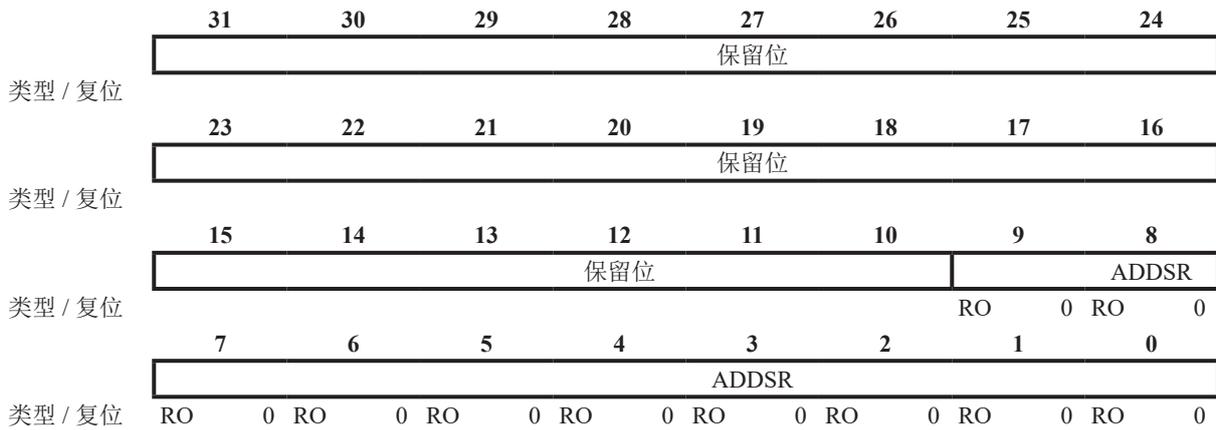
位	字段	描述
[9:0]	ADDR	地址屏蔽控制位 ADDR[i] 用于定义 I2CADDR 寄存器 ADDR 字段的第 i 位是否被屏蔽以及是否与接收到的 I ² C 总线上的地址帧作比较。该寄存器仅用于 I ² C 从机模式。 0: ADDR 第 i 位与 I ² C 总线上的地址帧作比较 1: ADDR 第 i 位被屏蔽且不与 I ² C 总线上的地址帧作比较

I²C 地址捕获寄存器 – I2CADDR

该寄存器用于描述 I²C 总线上地址帧的值。

偏移量: 0x024

复位值: 0x0000_0000



位	字段	描述
[9:0]	ADDR	地址捕获 一旦 I2CEN 位使能, I ² C 总线上的呼叫地址值将被自动载入到该 ADDR 字段。

I²C 超时寄存器 – I2CTOUT

该寄存器定义了 I²C 超时计数器预载值和时钟预分频比。

偏移量: 0x028
复位值: 0x0000_0000

31	30	29	28	27	26	25	24
保留位							
23	22	21	20	19	18	17	16
保留位				PSC			
				RW	0	RW	0
15	14	13	12	11	10	9	8
TOUT							
RW	0	RW	0	RW	0	RW	0
7	6	5	4	3	2	1	0
TOUT							
RW	0	RW	0	RW	0	RW	0

位	字段	描述
[18:16]	PSC	I ² C 超时计数器预分频选项 该 PSC 字段用于定义 I ² C 超时计数器时钟频率 f_{I2CTO} 。超时时钟频率通过以下公式获得。 $f_{I2CTO} = \frac{f_{PCLK}}{2^{PSC}}$ PSC = 0 → $f_{I2CTO} = f_{PCLK} / 2^0 = f_{PCLK}$ PSC = 1 → $f_{I2CTO} = f_{PCLK} / 2^1 = f_{PCLK} / 2$ PSC = 2 → $f_{I2CTO} = f_{PCLK} / 2^2 = f_{PCLK} / 4$... PSC = 7 → $f_{I2CTO} = f_{PCLK} / 2^7 = f_{PCLK} / 128$
[15:0]	TOUT	I ² C 超时计数器预载值 TOUT 字段用于定义计数器预载值。 下列任一情况发生时，计数器值重载： 1. I2CSR 寄存器的 RXBF、TXDE、RXDNE、RXNACK、GCS 或 ADRS 标志位被置位。 2. I ² C 主机模块发送 START 信号。 3. I ² C 从机模块检测到 START 信号。 下列任一情况发生时，计数器停止计数： 1. I ² C 从机模块没被寻址。 2. I ² C 主机模块发送 STOP 信号。 3. I ² C 从机模块检测到 STOP 信号。 4. I2CSR 寄存器的 ARBLOS 或 BUSERR 标志位被置位。

19 串行外设接口 (SPI)

简介

串行外设接口 SPI 在主从模式下均提供了一个符合 SPI 通信协议的数据传输与接收功能。SPI 接口使用 4 个引脚，其中有串行数据输入和输出线 MISO 和 MOSI，时钟线 SCK 和从机选择线 SEL。作为主机 SPI 的设备，使用 SEL 和 SCK 信号控制数据流来执行数据通信启动和控制数据采样率。为了接收数据流，SPICR1 寄存器的 DFL 字段定义的 1 位到 16 位数据帧长度在特定的时钟边沿时被锁存且存储在数据寄存器或 RX FIFO。数据传输也是通过类似的方式，但以相反的顺序。模式故障检测功能使其适用于多主机应用。

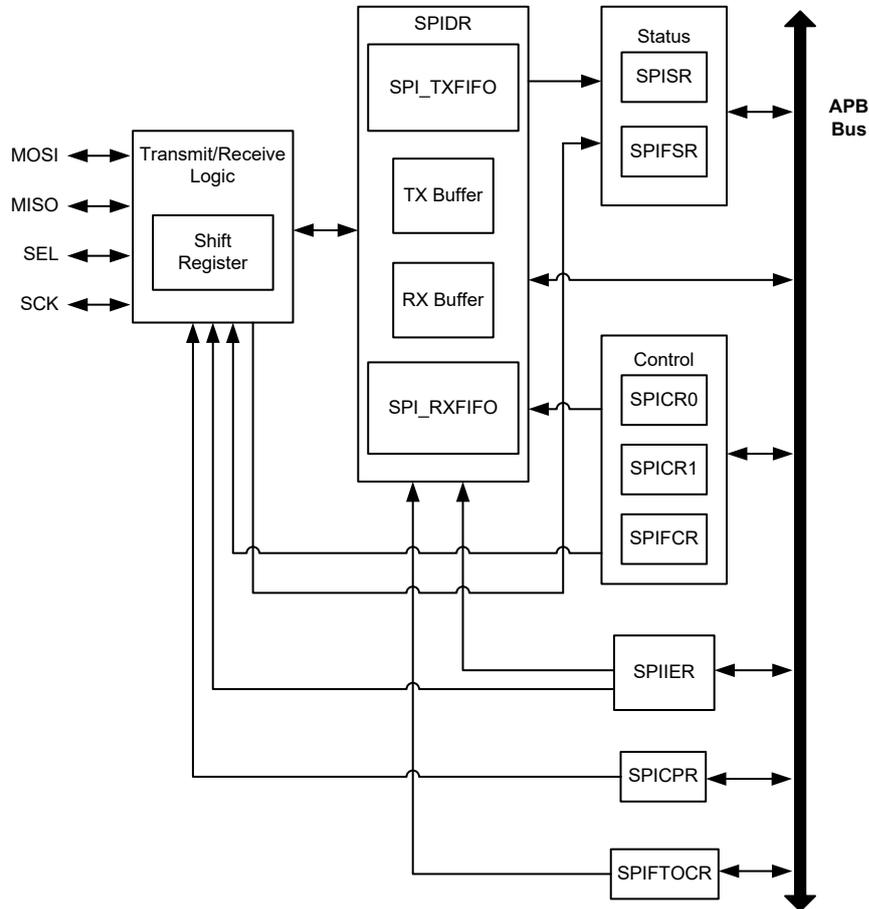


图 110. SPI 方框图

特性

- 主机或从机模式
- 主机模式频率高达 $f_{PCLK} / 2$
- 从机模式频率高达 $f_{PCLK} / 3$
- 可编程数据帧长度达 16 位
- FIFO 深度: 8 级
- MSB 或 LSB 优先传输选择
- 可编程从机选择高或低极性有效
- 多个主机和多个从机工作模式
- 主机模式支持 SPI NOR Flash 双输出读取模式
- 四个错误标志带有各自的中断
 - 读溢出
 - 写冲突
 - 模式故障
 - 从机中止

功能描述

主机模式

每个数据帧的数据长度范围是 1 ~ 16 位。被发送的数据的第一位可以是 MSB 或 LSB, 这由 SPICR1 寄存器的 FIRSTBIT 位决定。SPI 模块可以由 SPICR1 寄存器中的 MODE 位配置作为主机或从机。当 MODE 位被置位, SPI 模块被配置作为主机且在 SCK 引脚产生串行时钟。在串行时钟边沿发生时, 移位寄存器中的数据被发送到 MOSI 引脚。在数据传输的整个过程中, SEL 引脚保持有效电平。当 SPICR1 寄存器中的 SELAP 位被置位, 在完整的数据处理过程中 SEL 引脚为高电平有效。当 SPICR1 寄存器中的 SELM 位被置位, SEL 引脚将被硬件自动驱动, 在 SEL 有效边沿和 SCK 的第一个边沿之间的时间间隔等于 SCK 周期的一半。

从机模式

在从机模式, SCK 引脚作为输入脚且串行时钟来自于外部主机。SEL 引脚也作为输入脚。当 SELAP 位被清零, 在完整的数据流接收期间 SEL 引脚为低电平有效。当 SELAP 位被置为 1, 在完整数据流接收期间, SEL 引脚为高电平有效。

注: 在从机模式, APB 时钟, 即 f_{PCLK} , 必须至少为外部输入 SCK 时钟频率的 3 倍。

SPI 串行帧格式

SPI 接口帧格式取决于时钟极性位 CPOL 和时钟相位位 CPHA 的配置。

- 时钟极性 – CPOL
当时钟极性为零, SCK 线空闲状态为低电平。当时钟极性位为高, SCK 线空闲状态为高电平。
- 时钟相位 – CPHA
当时钟相位为零, 数据会在第一个 SCK 时钟转换信号跳变时被采样。当时钟相位为高, 数据会在第二个 SCK 时钟信号跳变时被采样。

SPI 接口有四种帧格式。表 41 显示了如何通过 SPICR1 寄存器中的 FORMAT 字段设置这些格式。

表 41. SPI 接口格式设置

FORMAT [2:0]	CPOL	CPHA
001	0	0
010	0	1
110	1	0
101	1	1
其它	保留	

CPOL = 0, CPHA = 0

在此格式下，接收到的数据在 SCK 信号上升沿被采样，而被发送的数据在 SCK 信号下降沿改变数据位。在主机模式中，当数据写入 SPIDR 寄存器时，第一个位被驱动。在从机模式中，当 SEL 信号变为有效电平时，第一个位被驱动。图 111 显示了此格式下单个字节数据传输时序图。

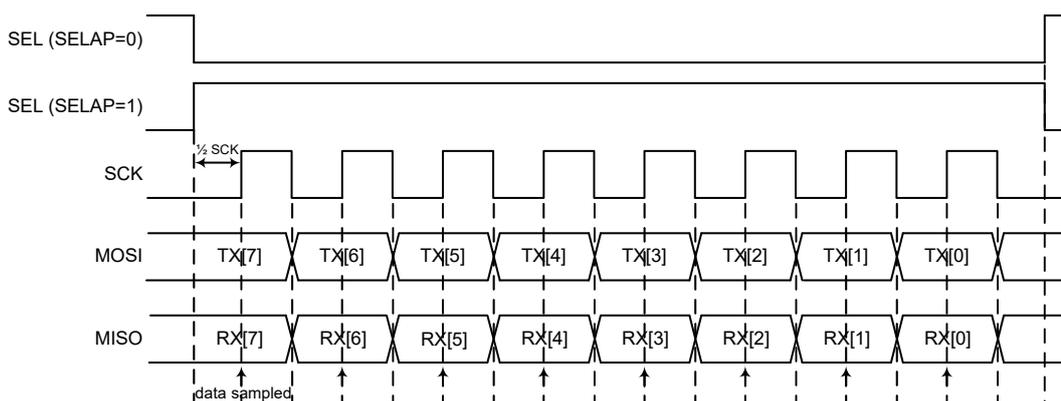


图 111. SPI 单字节传输时序图 – CPOL = 0, CPHA = 0

图 112 显示了此格式下连续数据传输的时序图。注意，在每个数据帧之间 SEL 信号必须转变为无效电平。

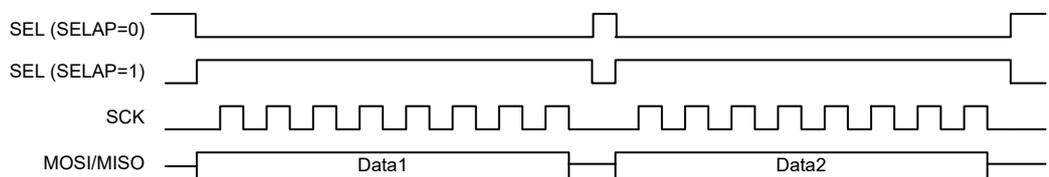


图 112. SPI 连续传输时序图 – CPOL = 0, CPHA = 0

CPOL = 0, CPHA = 1

在此格式下, 接收到的数据在 SCK 信号下降沿被采样, 而被发送的数据在 SCK 信号上升沿改变数据位。在主机模式中, 当数据写入 SPIDR 寄存器时, 第一个位被驱动。在从机模式中, 第一个 SCK 时钟上升沿到来时, 第一个位被驱动。图 113 显示了单个数据字节传输的时序。

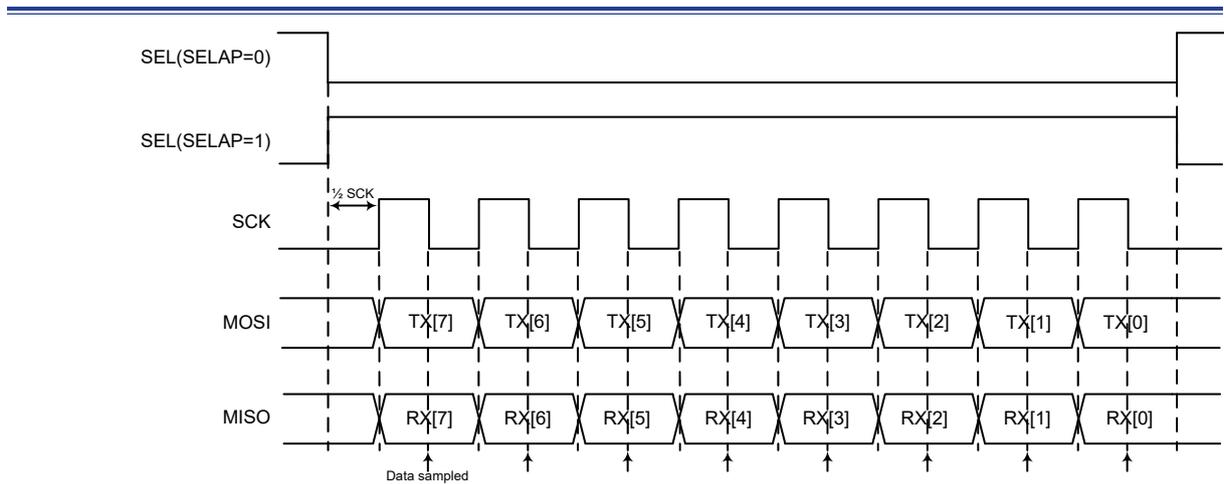


图 113. SPI 单个字节传输时序图 – CPOL = 0, CPHA = 1

图 114 显示了连续数据传输时序图。注意, SEL 信号必须保持在有效电平直到最后一个数据传输结束。

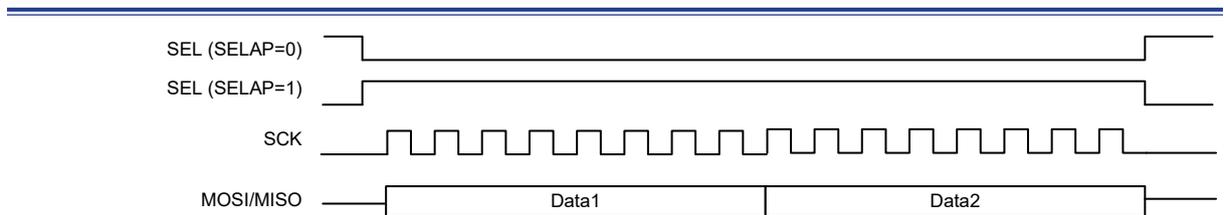


图 114. SPI 连续传输时序图 – CPOL = 0, CPHA = 1

CPOL = 1, CPHA = 0

在此格式下, 接收到的数据在 SCK 信号下降沿被采样, 而被发送的数据在 SCK 信号上升沿改变数据位。在主机模式中, 当数据写入 SPIDR 寄存器时, 第一个位被驱动。在从机模式中, 在 SEL 信号转换为有效电平时, 第一个位被驱动。图 115 显示了此格式下单个字节数据传输的时序图。

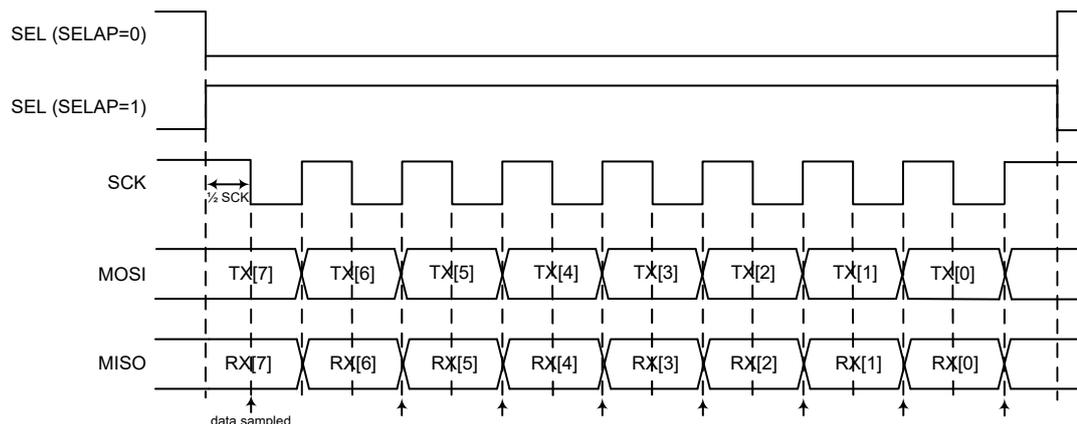


图 115. SPI 单个字节传输时序图 – CPOL = 1, CPHA = 0

图 116 显示了此格式下连续数据传输的时序图。注意, SEL 信号必须在每个数据帧之间转换为无效电平。

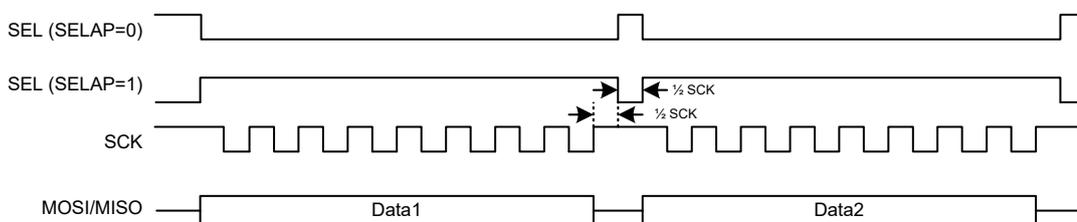


图 116. SPI 连续数据传输时序图 – CPOL = 1, CPHA = 0

CPOL = 1, CPHA = 1

在此格式下, 接收到的数据在 SCK 信号上升沿被采样, 而被发送的数据在 SCK 信号下降沿改变数据位。在主机模式中, 当数据写入 SPIDR 寄存器时, 第一个位被驱动。在从机模式中, 第一个 SCK 时钟下降沿到来时, 第一个位被驱动。图 117 显示了此格式下单个字节数据传输的时序图。

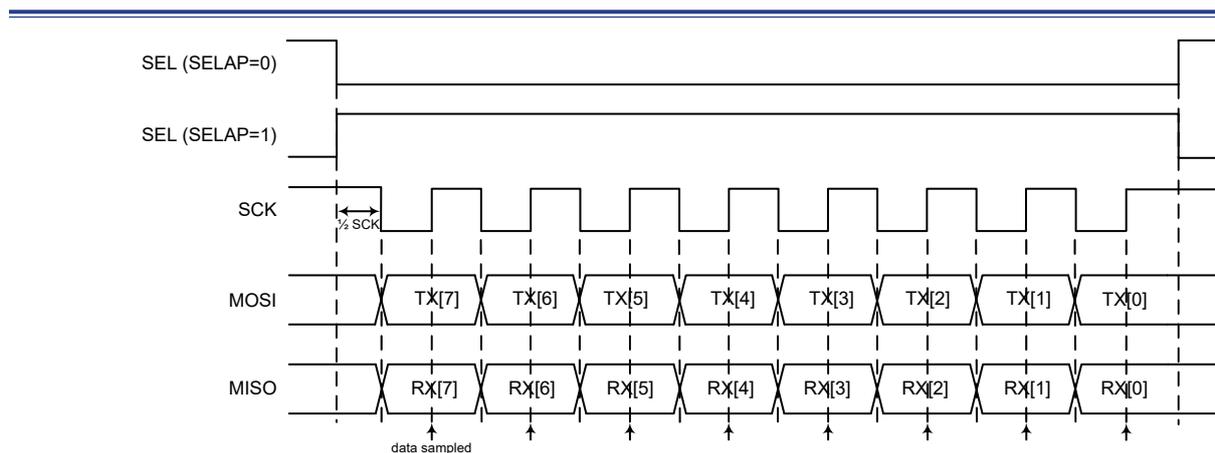


图 117. SPI 单个字节传输时序图 – CPOL = 1, CPHA = 1

图 118 显示了此格式下连续数据传输的时序图。注意, SEL 信号必须保持在有效电平直到最后一个数据传输结束。

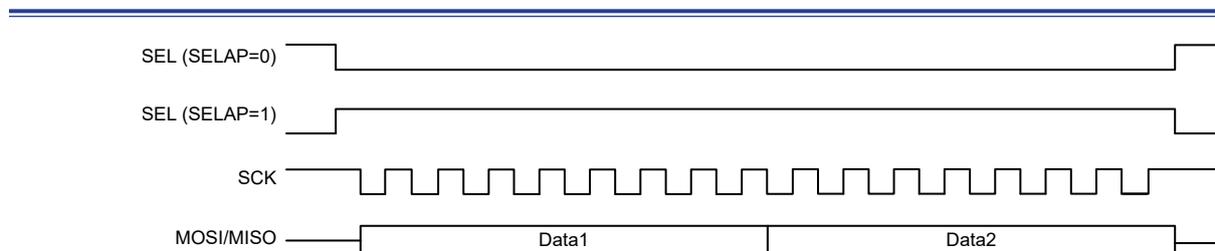


图 118. SPI 连续数据传输时序图 – CPOL = 1, CPHA = 1

状态标志

TX 缓存器空 – TXBE

在非 FIFO 模式下 TX 缓存器为空时，或在 FIFO 模式下当 TX FIFO 数据长度等于或小于 SPIFCR 寄存器中 TXFTLS 字段定义的 TX FIFO 阈值级别时，TXBE 标志位被置位。接着要发送的数据可被载入缓存器。此外，在非 FIFO 模式下 TX 缓存器已经包含了一个新的数据时，或在 FIFO 模式下当 TX FIFO 数据长度大于 TXFTLS 字段定义的 TX FIFO 阈值级别时，TXBE 标志位被复位。

传输寄存器空 – TXE

当 TX 缓存器和 TX 移位寄存器都为空时，TXE 标志位被置位。当 TX 缓存器或 TX 移位寄存器包含等待被发送的数据时，TXE 标志位被复位。

RX 缓存器非空 – RXBNE

在非 FIFO 模式下 RX 缓存器中接收到有效数据，或在 SPI FIFO 模式下 RX FIFO 数据长度等于或大于 SPIFCR 寄存器中 RXFTLS 字段定义的 RX FIFO 阈值时，RXBNE 标志位被置位。在非 FIFO 模式下当接收到的数据已从 RX 缓存器中被全部读取，或在 FIFO 模式下当 RX FIFO 数据长度小于由 RXFTLS 字段设置的 RX FIFO 阈值时，此标志位将由硬件自动清零。

超时标志 – TO

超时功能仅在 SPI FIFO 模式下有效，可通过载 0 到超时计数器寄存器的 TOC 字段来除能。当超时功能使能，如果 SPI RX FIFO 非空，超时计数器将开始计数，一旦数据从 SPIDR 寄存器被读出，或者接收到新的数据，超时计数器将复位为 0 并再次开始计数。当超时计数器的值等于 SPIFTOCR 寄存器中的 TOC 字段指定的值时，TO 标志位将被置位。此标志位通过向此位写 1 来清零。

模式故障标志 – MF

模式故障标志可用于检测 SPI 多主机模式下的 SPI 总线使用情况。在多主机模式下，SPI 模块被配置作为主机且 SEL 信号被设置作为输入信号。当 SEL 引脚突然被另一个主机变换为有效电平时，模式故障标志位将被置位。这意味着另一个 SPI 主机正在请求使用 SPI 总线。因此，当 SPI 模式故障发生，将迫使此 SPI 模块工作在从机模式且除能此 SPI 接口所有的信号来避免 SPI 总线信号冲突。同样地，在多主机模式下，如果 SPI 主机想要取得总线使用权传输数据，有必要通过驱动其它主机的 SEL 信号变为一个有效状态来告知其它的 SPI 主机。SPI 多主机模式下的详细配置图如下图所示。

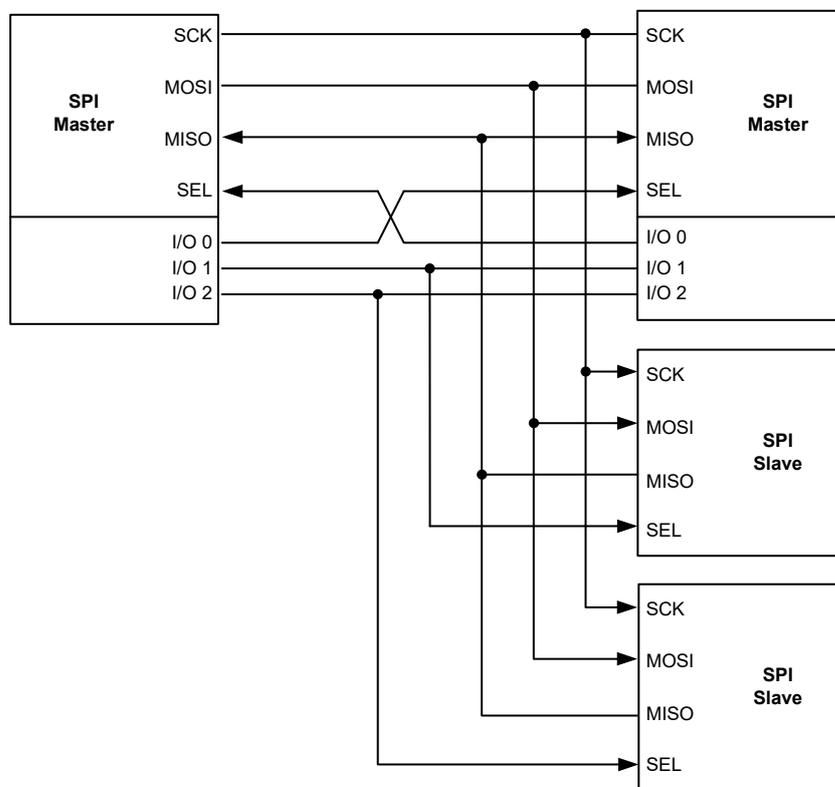


图 119. SPI 多主机从机环境

表 42. SPI 模式故障触发条件

模式故障	描述
触发条件	<ol style="list-style-type: none"> 1. SPI 主机模式 2. SPICR0 寄存器的 SELOEN = 0 – SEL 引脚被配置为输入模式 3. SEL 信号被外部 SPI 主机驱动，转换为有效电平
SPI 行为	<ol style="list-style-type: none"> 1. 模式故障标志位置位。 2. SPICR0 寄存器中的 SPIEN 位被复位。这将除能 SPI 接口，使 SPI 接口停止所有的信号输出。 3. SPICR1 寄存器中的 MODE 位被复位。这将迫使单片机进入从机模式。

表 43. SPI 主机模式 SEL 引脚状态

	SEL 作为输入 – SELOEN = 0	SEL 作为输出 – SELOEN = 1								
多主机	支持	不支持								
SPI SEL 控制信号	使用另一个 GPIO 代替 SEL 引脚功能	SEL 引脚处于硬件控制或软件控制模式 – 使用 SELM 位设置								
连续传输	<table border="1"> <tr> <td>情况 1</td> <td>情况 2</td> </tr> <tr> <td>不支持</td> <td>支持</td> </tr> </table>	情况 1	情况 2	不支持	支持	<table border="1"> <tr> <td>情况 1</td> <td>情况 2</td> </tr> <tr> <td>使用硬件控制</td> <td>硬件或软件控制</td> </tr> </table>	情况 1	情况 2	使用硬件控制	硬件或软件控制
情况 1	情况 2									
不支持	支持									
情况 1	情况 2									
使用硬件控制	硬件或软件控制									

情况 1：每次数据传输之间，SEL 信号必须为无效。

情况 2：直到最后一个数据帧传输完成后，SEL 信号才会无效。

注：当 SPI 模块工作在从机模式时，SEL 信号始终作为输入且不受 SPICR0 寄存器中 SELOEN 位的影响。

写冲突标志 – WC

下列情况将使写冲突标志置位：

- SPIFCR 寄存器中的 FIFOEN 位被清零 (FIFO 除能)
TX 缓存器和移位寄存器都已经为满时，当有新的数据写入到 SPIDR 寄存器，写冲突标志会被置位。任何被写入 TX 缓存器的新的数据将丢失。
- SPIFCR 寄存器中的 FIFOEN 位被置位 (FIFO 使能)
TX FIFO 和 TX 移位寄存器都已经为满时，当有新的数据写入到 SPIDR 寄存器，写冲突标志会被置位。任何被写入 TX FIFO 的新的数据都将丢失。

读溢出标志 – RO

- SPIFCR 寄存器中的 FIFOEN 位被清零 (FIFO 除能)
当 RX 移位寄存器和 RX 缓存器都已满，如果再一笔数据被接收时读溢出标志位被置位。这意味着最新接收的数据不会被移到 SPI 移位寄存器中。所以最新的接收数据将丢失。
- SPIFCR 寄存器中的 FIFOEN 位被置位 (FIFO 使能)
当 RX 移位寄存器和 RX FIFO 都已满，如果再一笔数据被接收时读溢出标志位被置位。这意味着最新的接收数据不会被移到 SPI 移位寄存器中。所以最新的接收数据将丢失。

从机中止标志 – SA

在 SPI 从机模式，从机中止标志位被置位表明在数据帧传输期间 SEL 引脚被突然转换到一个无效的状态。数据帧长度由 SPICR1 寄存器中的 DFL 字段定义。

寄存器列表

下表显示了 SPI 寄存器及其复位值。

表 44. SPI 寄存器列表

寄存器	偏移量	描述	复位值
SPICR0	0x000	SPI 控制寄存器 0	0x0000_0000
SPICR1	0x004	SPI 控制寄存器 1	0x0000_0000
SPIIER	0x008	SPI 中断使能寄存器	0x0000_0000
SPICPR	0x00C	SPI 时钟预分频器寄存器	0x0000_0000
SPIDR	0x010	SPI 数据寄存器	0x0000_0000
SPISR	0x014	SPI 状态寄存器	0x0000_0003
SPIFCR	0x018	SPI FIFO 控制寄存器	0x0000_0000
SPIFSR	0x01C	SPI FIFO 状态寄存器	0x0000_0000
SPIFTOCR	0x020	SPI FIFO 超时计数器寄存器	0x0000_0000

寄存器描述

SPI 控制寄存器 0 – SPICR0

该寄存器定义了 SEL 控制位和 SPI 使能位。

偏移量: 0x000

复位值: 0x0000_0000

31	30	29	28	27	26	25	24	保留位							
类型 / 复位															
23	22	21	20	19	18	17	16	保留位							
类型 / 复位															
15	14	13	12	11	10	9	8	SELHT				GUADT			
类型 / 复位															
RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0
7	6	5	4	3	2	1	0	GUADTEN	DUALEN	保留位	SSELC	SELOEN	保留位	SPIEN	
类型 / 复位															
RW	0	RW	0					RW	0	RW	0			RW	0

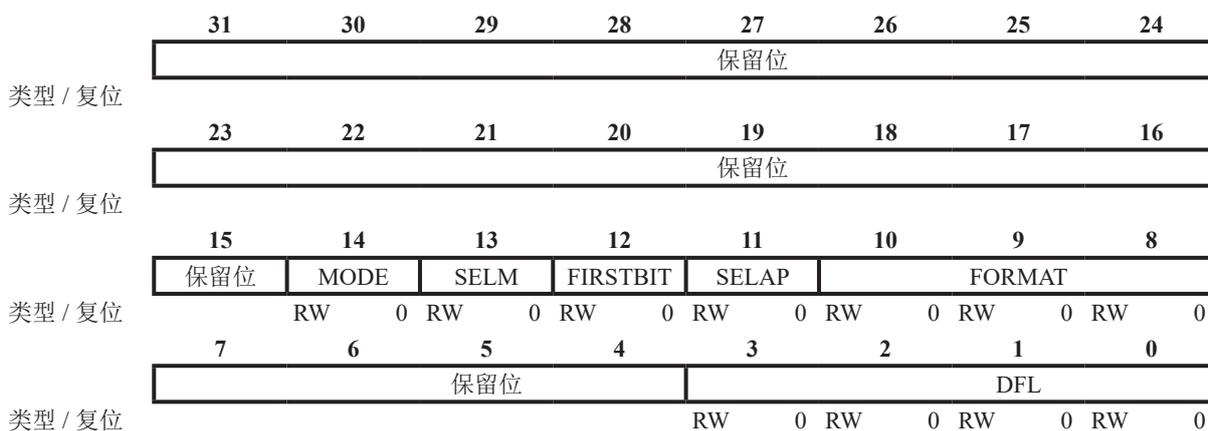
位	字段	描述
[15:12]	SELHT	芯片选择保持时间控制 0x0: 1/2 SCK 0x1: 1 SCK 0x2: 3/2 SCK 0x3: 2 SCK ... 注意, SELHT 位仅适用于主机模式。
[11:8]	GUADT	保护时间控制 GUADTEN = 1 0x0: 1 SCK 0x1: 2 SCK 0x2: 3 SCK ... 注意, GUADT 位仅适用于主机模式。
[7]	GUADTEN	保护时间使能位 0: 保护时间为 1/2 SCK 1: 当该位置高, 保护时间由 GUADT 位控制 注意, GUADTEN 位仅适用于主机模式。
[6]	DUALEN	双端口使能位 0: 双端口除能 1: 双端口使能 该控制位用于支持串行 SPI NOR Flash 双输出读取模式。当该位置位, MOSI 信号将改变方向由输出变为输入并接收串行数据流。这就意味着 DUALEN 控制位仅适用于主机模式。

位	字段	描述
[4]	SSELC	软件从机选择控制位 0: 设置 SEL 输出为一个无效电平 1: 设置 SEL 输出为一个有效电平 软件可以通过设置 SSELC 位使 SEL 输出为一个有效或无效电平。有效电平通过 SPICR1 寄存器中的 SELAP 位设置。注意，只有当 SELOEN 位置 1 使能 SEL 输出且同时 SELM 位清零使用软件控制 SEL 信号，SSELC 位才是可用的。否则，SSELC 位无影响。
[3]	SELOEN	从机选择输出使能位 0: 设置 SEL 信号为输入模式用于多主机模式 1: 设置 SEL 信号为输出模式用于从机选择 SELOEN 仅在主机模式可用，用来设置 SEL 信号作为输入或输出信号。当 SEL 信号被配置为工作在输出模式时，根据 SPICR1 寄存器中 SELM 位的设置，它可作为硬件控制或软件控制模式下的从机选择信号。当 SEL 信号被配置工作在输入模式时，它可在多主机环境中用作模式故障检测功能。
[0]	SPIEN	SPI 使能位 0: SPI 接口除能 1: SPI 接口使能

SPI 控制寄存器 1 – SPICR1

该寄存器定义了 SPI 的参数，包括数据长度、传输格式、SEL 有效极性/模式、LSB / MSB 控制以及主机 / 从机模式。

偏移量： 0x004
复位值： 0x0000_0000



位	字段	描述
[14]	MODE	主机或从机模式 0: 从机模式 1: 主机模式
[13]	SELM	从机选择模式 0: SEL 信号由软件控制 – 电平由 SSEL 位控制 1: SEL 信号由硬件控制 – 由 SPI 硬件自动产生 注意，只有在主机模式下 SELM 位可用 – MODE = 1
[12]	FIRSTBIT	LSB 或 MSB 优先发送 0: MSB 优先发送 1: LSB 优先发送
[11]	SELAP	从机选择有效极性 0: SEL 低电平有效 1: SEL 高电平有效
[10:8]	FORMAT	SPI 数据传输格式 这些位用来决定 SPI 接口数据传输格式

FORMAT[2:0]	CPOL	CPHA
001	0	0
010	0	1
110	1	0
101	1	1
其它	保留	

CPOL: 时钟极性
0: SCK 空闲状态为低
1: SCK 空闲状态为高

CPHA: 时钟相位
0: 数据在第一个 SCK 时钟沿被捕捉
1: 数据在第二个 SCK 时钟沿被捕捉

位	字段	描述
[3:0]	DFL	数据帧长度 在 1~16 位之间选择数据传输的数据帧长度 0001: 1 位 0010: 2 位 ... 1111: 15 位 0000: 16 位

SPI 中断使能寄存器 – SPIER

该寄存器包含了相关的 SPI 中断使能控制位。

偏移量: 0x008
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24
类型 / 复位	保留位							
	23	22	21	20	19	18	17	16
类型 / 复位	保留位							
	15	14	13	12	11	10	9	8
类型 / 复位	保留位							
	7	6	5	4	3	2	1	0
类型 / 复位	TOIEN	SAIEN	MFIEN	ROIEN	WCIEN	RXBNEIEN	TXEIEIN	TXBEIEN
	RW	0 RW	0 RW	0 RW	0 RW	0 RW	0 RW	0 RW

位	字段	描述
[7]	TOIEN	超时中断使能位 0: 除能 1: 使能
[6]	SAIEN	从机中止中断使能位 0: 除能 1: 使能
[5]	MFIEN	模式故障中断使能位 0: 除能 1: 使能
[4]	ROIEN	读溢出中断使能位 0: 除能 1: 使能
[3]	WCIEN	写冲突中断使能位 0: 除能 1: 使能

位	字段	描述
[2]	RXBNEIEN	RX 缓存器非空中断使能位 0: 除能 1: 使能 RXBNE 标志位和 RXBNEIEN 位被置位时, 会产生一个中断请求。在 FIFO 模式下, 中断请求标志的产生取决于 RX FIFO 触发级别的设置。
[1]	TXEIEN	TX 空中断使能位 0: 除能 1: 使能 TXE 标志位和 TXEIEN 位被置位时, 将产生 TX 寄存器空中断请求。
[0]	TXBEIEN	TX 缓存器空中断使能位 0: 除能 1: 使能 TXBE 标志位和 TXBEIEN 位被置位时, TX 缓存器空中断请求将会产生。在 FIFO 模式下, 中断请求标志的产生取决于 TX FIFO 触发级别的设置。

SPI 时钟预分频器寄存器 – SPICPR

该寄存器定义了 SPI 时钟预分频比。

偏移量: 0x00C
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24						
类型 / 复位	保留位													
	23	22	21	20	19	18	17	16						
类型 / 复位	保留位													
	15	14	13	12	11	10	9	8						
类型 / 复位	CP													
	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0						
类型 / 复位	CP													
	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0

位	字段	描述
[15:0]	CP	SPI 时钟预分频器 SPI 时钟 SCK 频率由以下等式决定: $f_{SCK} = f_{PCLK} / (2 \times (CP + 1))$, CP 范围: 0 ~ 65535 注: 在 SPI 从机模式下, 系统时钟 (f_{PCLK}) 必须至少为外部输入 SPI SCK 频率的 3 倍。

SPI 数据寄存器 – SPIDR

该寄存器存储了 SPI 接收或发送的数据。

偏移量: 0x010

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位								
	15	14	13	12	11	10	9	8	
类型 / 复位	DR								
	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
类型 / 复位	DR								
	RW	0	RW	0	RW	0	RW	0	RW

位	字段	描述
[15:0]	DR	数据寄存器 SPI 数据寄存器用来存储串行总线发送或接收的数据。在非 FIFO 模式下，写数据到 SPI 数据寄存器也会使数据加载到数据发送缓存器即 TX 缓存器内。从 SPI 数据寄存器中读取数据将会返回接收缓存器即 RX 缓存器中的数据。

SPI 状态寄存器 – SPISR

该寄存器包含了相关的 SPI 状态。

偏移量: 0x014

复位值: 0x0000_0003

31	30	29	28	27	26	25	24	
保留位								
23	22	21	20	19	18	17	16	
保留位								
15	14	13	12	11	10	9	8	
保留位							BUSY	
							RO	0
7	6	5	4	3	2	1	0	
TO	SA	MF	RO	WC	RXBNE	TXE	TXBE	
WC	0	WC	0	WC	0	RO	0	
WC	0	WC	0	WC	0	RO	1	
WC	0	WC	0	WC	0	RO	1	

位	字段	描述
[8]	BUSY	SPI 忙标志位 0: SPI 不忙 1: SPI 忙 在主机模式, 当 TX 缓存器和 TX 移位寄存器都为空时, 此位被复位。当 TX 缓存器或 TX 移位寄存器不为空时, 此位被置位。 在从机模式, 当 SEL 信号转变为有效电平时, 此位被置位。当 SEL 信号转变为无效电平时, 此位被复位。
[7]	TO	超时标志位 0: 无 RX FIFO 超时 1: RX FIFO 超时发生 一旦超时计数器的值等于 SPIFTOCR 寄存器中 TOC 字段的设定时, 超时标志位将置位, SPIIER 寄存器中的 TOIEN 位使能, 则将产生中断。此位通过写 1 使其清零。 注: 只有在 SPI FIFO 模式下, 超时标志功能可用。
[6]	SA	从机中止标志 0: 无从机中止发生 1: 从机中止发生 此位由硬件置位且通过写 1 清零。
[5]	MF	模式故障标志 0: 无模式故障 1: 模式故障发生 此位由硬件置位且通过写 1 清零。
[4]	RO	读溢出标志 0: 无读溢出 1: 读溢出发生 此位由硬件置位且通过写 1 清零。
[3]	WC	写冲突标志 0: 无写冲突 1: 写冲突发生 此位由硬件置位且通过写 1 清零。

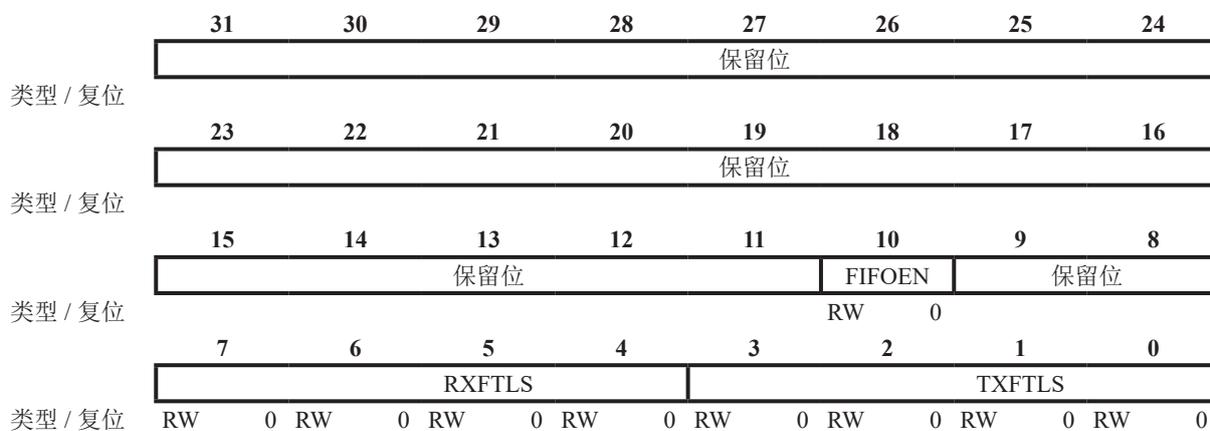
位	字段	描述
[2]	RXBNE	接收缓存器非空标志位 0: RX 缓存器为空 1: RX 缓存器非空 此位表明了在非 FIFO 模式下 RX 缓存器的状态。它也用来说明在 FIFO 模式下是否达到 RX FIFO 触发级别。在非 FIFO 模式下，当 SPI RX 缓存器为空时或在 SPI FIFO 模式下当存放在 FIFO 的数据量小于由 SPIFCR 寄存器中的 RXFTLS 字段定义的触发级别时，此位将被清零。
[1]	TXE	发送寄存器空标志位 0: TX 缓存器或 TX 移位寄存器非空 1: TX 缓存器和 TX 移位寄存器都为空
[0]	TXBE	发送缓存器空标志位 0: TX 缓存器非空 1: TX 缓存器为空 在 FIFO 模式下，此位置位则表明存放在 TX FIFO 的数据量等于或小于 SPIFCR 寄存器中 TXFTLS 字段定义的触发级别。

SPI FIFO 控制寄存器 – SPIFCR

该寄存器包含了 SPI FIFO 相关的控制，包括 FIFO 使能控制和 FIFO 触发级别选项。

偏移量: 0x018

复位值: 0x0000_0000



位	字段	描述
[10]	FIFOEN	FIFO 使能位 0: FIFO 除能 1: FIFO 使能 当 SPI 接口正在发送时，此位不可被置位或复位。

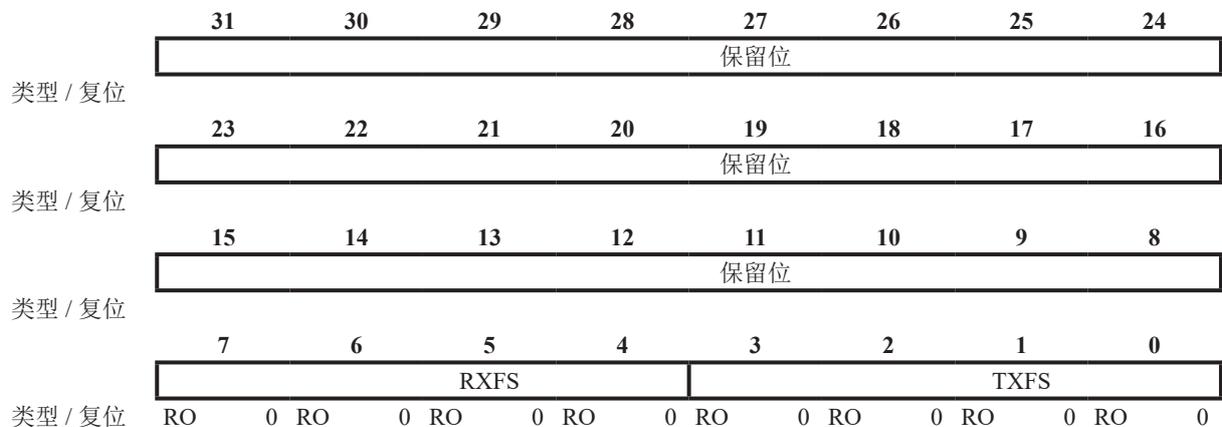
位	字段	描述
[7:4]	RXFTLS	<p>RX FIFO 触发级别选择</p> <p>0000: 触发级别为 0</p> <p>0001: 触发级别为 1</p> <p>...</p> <p>1000: 触发级别为 8</p> <p>其它: 保留</p> <p>RXFTLS 字段用来定义 RX FIFO 触发级别。当 RX FIFO 中的数据量等于或大于 RXFTLS 字段定义的触发级别时, RXBNE 标志将被置位。</p>
[3:0]	TXFTLS	<p>TX FIFO 触发级别选择</p> <p>0000: 触发级别为 0</p> <p>0001: 触发级别为 1</p> <p>...</p> <p>1000: 触发级别为 8</p> <p>其它: 保留</p> <p>TXFTLS 字段用来定义 TX FIFO 触发级别。当 TX FIFO 中的数据量等于或小于 TXFTLS 字段定义的触发级别时, TXBE 标志位被置位。</p>

SPI FIFO 状态寄存器 – SPIFSR

该寄存器包含了 SPI FIFO 相关的状态。

偏移量: 0x01C

复位值: 0x0000_0000



位	字段	描述
[7:4]	RXFS	<p>RX FIFO 状态</p> <p>0000: RX FIFO 为空</p> <p>0001: RX FIFO 包含 1 个数据</p> <p>...</p> <p>1000: RX FIFO 包含 8 个数据</p> <p>其它: 保留</p>
[3:0]	TXFS	<p>TX FIFO 状态</p> <p>0000: TX FIFO 为空</p> <p>0001: TX FIFO 包含 1 个数据</p> <p>...</p> <p>1000: TX FIFO 包含 8 个数据</p> <p>其它: 保留</p>

SPI FIFO 超时计数器寄存器 – SPIFCR

该寄存器存储了 SPI RX FIFO 超时计数器的值。

偏移量: 0x020

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位								
	15	14	13	12	11	10	9	8	
类型 / 复位	TOC								
	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
类型 / 复位	TOC								
	RW	0	RW	0	RW	0	RW	0	RW

位	字段	说明
[15:0]	TOC	<p>超时计数器比较值</p> <p>超时计数器会在 SPI RX FIFO 接收到数据后从 0 开始计数，且一旦通过软件从 SPIDR 寄存器读取数据，或者接收到新的数据时，会复位计数器。如果 FIFO 没接收到新的数据或数据没有通过软件从 SPIDR 寄存器被读取，超时计数器值连续递增。超时计数器值等于 TOC 设置值时，SPISR 寄存器的 TO 标志位将被置位，若 SPIIER 寄存器的 TOIEN 位已置位，将会产生中断。当 RX FIFO 为空时，超时计数器将被停止。通过清零 TOC 字段，SPI FIFO 超时功能可除能。超时计数器由系统 APB 时钟 f_{PCLK} 驱动。</p>

20 通用异步收发器 (UART)

简介

通用异步收发器 UART, 提供了一个灵活的异步传输的全双工数据交换。UART 用来转换并行和串行接口之间的数据, 通常也被用作 RS232 标准通信。UART 外设功能支持多种类型的中断。

UART 模块包含一个发送数据寄存器 TDR 和发送移位寄存器 TSR 以及一个接收数据寄存器 RDR 和接收移位寄存器 RSR。通过读取 UART 状态 & 中断标志位寄存器 URSIFR, 软件可检测 UART 的错误状态。这些状态包括传输类型和状况以及因奇偶校验、溢出、帧错误和线中止事件造成的错误状况。

UART 有一个可编程的波特率发生器, 能对 CK_APB (CK_UART) 的 UART 时钟进行分频以产生 UART 发送器和接收器所需的波特率时钟。

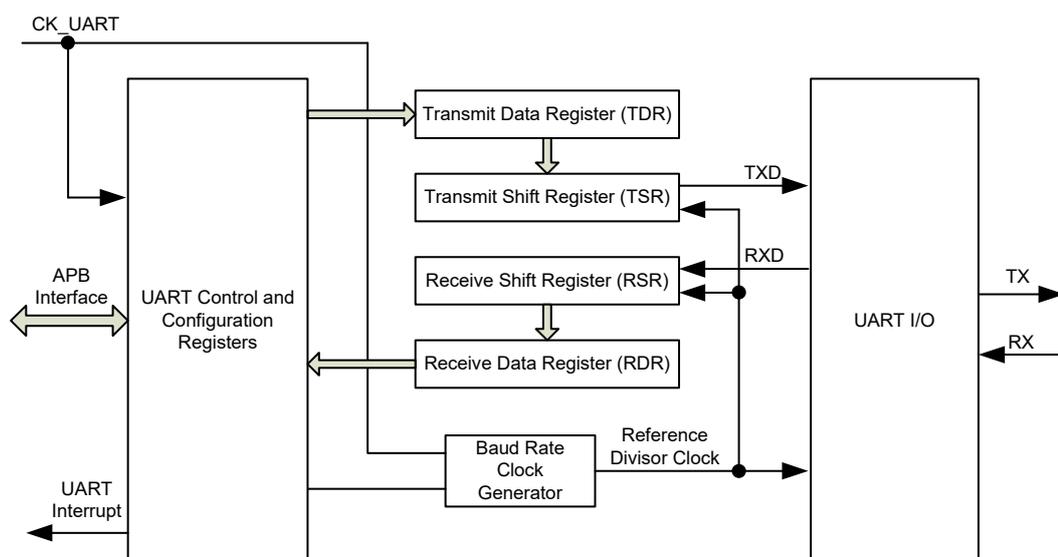


图 120. UART 方框图

特性

- 同时支持异步串行通信模式
- 全双工通信能力
- 可编程波特率时钟频率高达 ($f_{PCLK} / 16$) MHz
- 完全可编程串行通信功能包括：
 - 字长：7、8 或 9-bit 字符
 - 校验位：偶校验、奇校验或无奇偶校验位的产生和检测
 - 停止位：1 或 2 个停止位
 - 位顺序：LSB 优先或 MSB 优先传输
- 错误检测：奇偶校验、溢出和帧错误

功能描述

串行数据格式

UART 模块对写入发送数据寄存器中的数据进行并行到串行的转换，然后发送具有以下格式的数据：起始位，7~9 个 LSB/MSB 优先的数据位，可选奇偶校验位和最后 1~2 个停止位。起始位的极性与数据线空闲状态相反。停止位与数据线空闲状态相同，并在下一步起始位发生之前提供延迟。开始和停止位都用于异步数据传输过程中的数据同步。

UART 模块对从接收数据寄存器中读取的数据进行串行到并行的转换。它会首先检查校验位，然后将寻找一个停止位。如果停止位没有找到，UART 模块会认为整个字的传输失败并以帧错误作为响应。

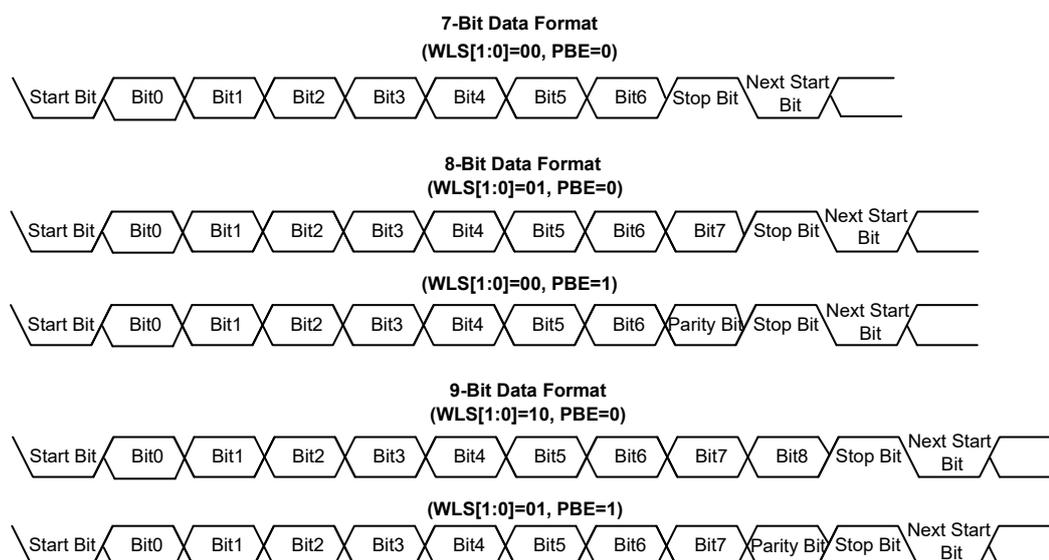


图 121. UART 串行数据格式

波特率发生器

UART 接收器和发送器的波特率都设置为相同值。波特率分频器 BRD 与 UART 时钟 CK_UART 的关系如下。

$$\text{波特率时钟} = \text{CK_UART} / \text{BRD}$$

CK_UART 时钟是连接到 UART 模块的 APB 时钟, BRD 的范围是 16 ~ 65535。

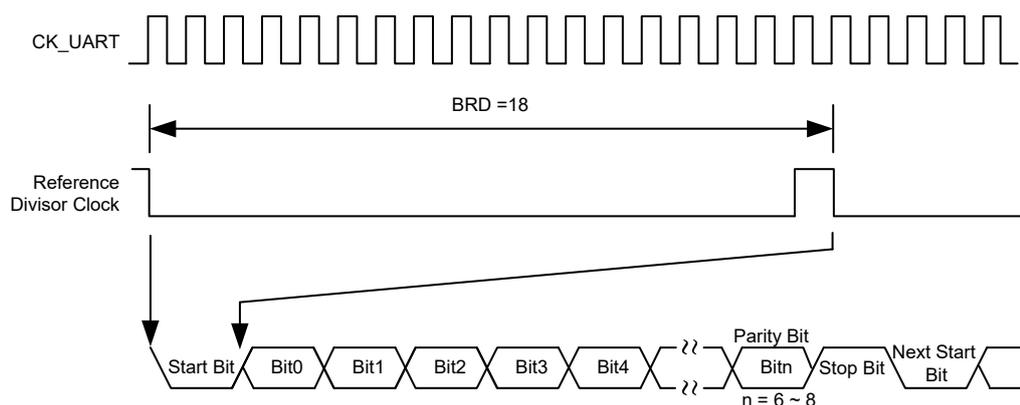


图 122. UART 时钟 CK_UART 和数据帧时序

表 45. 波特率误差计算 – CK_UART = 20 MHz

波特率		CK_UART = 20 MHz		
No.	Kbps	实际值	BRD	误差率
1	2.4	2.4	8333	0.00%
2	9.6	9.6	2083	0.02%
3	19.2	19.2	1042	-0.03%
4	57.6	57.6	347	0.06%
5	115.2	114.9	174	-0.22%
6	230.4	229.9	87	-0.22%
7	460.8	465.1	43	0.94%
8	921.6	909.1	22	-1.36%
9	1250	1250	16	0%

表 46. 波特率误差计算 – CK_UART = 10MHz

波特率		CK_UART = 10 MHz		
No.	Kbps	实际值	BRD	误差率
1	2.4	2.4	4167	-0.01%
2	9.6	9.6	1042	-0.03%
3	19.2	19.2	521	-0.03%
4	57.6	57.6	174	-0.22%
5	115.2	114.9	87	-0.22%
6	230.4	232.6	43	0.94%
7	460.8	454.5	22	-1.36%
8	625	625	16	0%

中断和状态

下列事件发生时，若相应的中断使能位被置位，UART 模块可产生一个中断：

- 接收器线路状态中断：UART 接收器溢出错误、奇偶错误、帧错误或线中止事件发生。
- 发送数据寄存器为空中断：UART 发送数据寄存器的内容传输至发送移位寄存器 TSR。
- 发送完成中断：发送数据寄存器 TDR 为空且发送移位寄存器 TSR 的内容已全部移出。
- 接收数据就绪中断：接收移位寄存器 RSR 的内容已全部传入 URDR 寄存器且读取准备就绪。

寄存器列表

下表显示了 UART 寄存器及其复位值。

表 47. UART 寄存器列表

寄存器	偏移量	描述	复位值
URDR	0x000	UART 数据寄存器	0x0000_0000
URCR	0x004	UART 控制寄存器	0x0000_0000
URIER	0x00C	UART 中断使能寄存器	0x0000_0000
URSIFR	0x010	UART 状态 & 中断标志位寄存器	0x0000_0180
URDLR	0x024	UART 分频器锁存寄存器	0x0000_0010
URTSTR	0x028	UART 测试寄存器	0x0000_0000

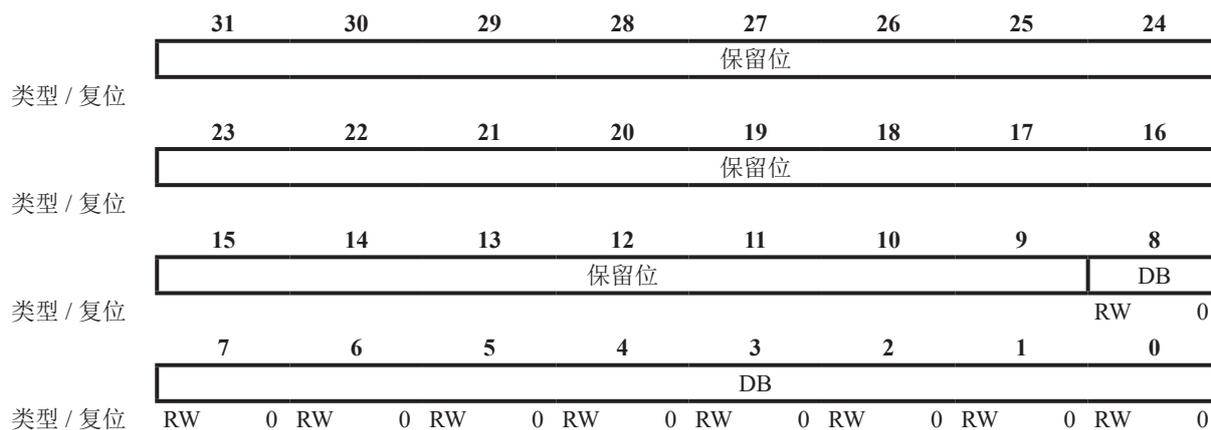
寄存器描述

UART 数据寄存器 – URDR

该寄存器用来存取 UART 发送与接收到的数据。

偏移量: 0x000

复位值: 0x0000_0000



位	字段	描述
[8:0]	DB	通过读取该寄存器，UART 将返回 7、8 或 9 位接收到的数据。DB 字段第 8 位仅在 9-bit 模式下有效，在 8-bit 模式下固定为 0。在 7-bit 模式下，接收器缓冲寄存器的 DB[6:0] 包含可用位。 通过写入该寄存器，UART 将送出 7、8 或 9 位发送的数据。DB 字段的第 8 位只有在 9-bit 模式下有效，在 8-bit 模式下将被忽略。在 7-bit 模式下，发送器缓存寄存器 DB[6:0] 包含可用位。

UART 控制寄存器 – URCR

该寄存器用于定义串口参数，如 UART 的数据长度，奇偶和停止位。

偏移量： 0x004
复位值： 0x0000_0000

31	30	29	28	27	26	25	24
保留位							
23	22	21	20	19	18	17	16
保留位							
15	14	13	12	11	10	9	8
保留位	BCB	SPE	EPE	PBE	NSB	WLS	
	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
7	6	5	4	3	2	1	0
	保留位	URRXEN	URTXEN	保留位	TRSM	保留位	
		RW 0	RW 0		RW 0		

位	字段	描述
[14]	BCB	线中止控制位 当此位被设置为 1，UART TX 引脚上的串行数据输出将被迫进入空白状态（逻辑 0）。 该位仅作用于 UART TX 输出脚，对发送器逻辑无影响。
[13]	SPE	强制奇偶校验使能 0：除能强制奇偶校验 1：使能强制奇偶校验 当 PBE 位设置为 1 时，此位才可用。如果 PBE 和 SPE 位都被设置为 1，并且 EPE 位被清零，则被发送的奇偶校验位将被强制为 1。当 PBE 和 SPE 位被设置为 1 并且 EPE 位也被设置为 1，则被传送的奇偶校验位将被强制清零。
[12]	EPE	偶校验使能 0：在数据字和奇偶校验位中共有奇数个逻辑“1”被发送或被检测到 1：在数据字和奇偶校验位中共有偶数个逻辑“1”被发送或被检测到 当 PBE 位设置为 1 时，此位才可用。
[11]	PBE	奇偶校验位使能 0：在传输过程中，奇偶校验位不会生成（发送数据）或被检查（接收数据） 1：在传输过程中，奇偶校验位生成或被检查 注：当 WLS 字段设置为“10”来选择 9-bit 数据格式时，对 PBE 位的写操作没有任何影响。
[10]	NSB	停止位的个数 0：为传输的数据生成一个停止位 1：选择 8 位或 9 位字长时，生成两个停止位
[9:8]	WLS	字长选择 00：7 位 01：8 位 10：9 位 11：保留
[5]	URRXEN	UART RX 使能 0：除能 1：使能

位	字段	描述
[4]	URTXEN	UART TX 使能 0: 除能 1: 使能
[2]	TRSM	传输模式选择 此位用于选择数据传输协议。 0: LSB 优先 1: MSB 优先

UART 中断使能寄存器 – URIER

此寄存器用于使能相关 UART 中断功能。当相应的事件发生且相应的中断使能位置位时，UART 模块产生中断。

偏移量: 0x00C

复位值: 0x0000_0000

31	30	29	28	27	26	25	24
保留位							
类型 / 复位							
23	22	21	20	19	18	17	16
保留位							
类型 / 复位							
15	14	13	12	11	10	9	8
保留位							
类型 / 复位							
7	6	5	4	3	2	1	0
保留位	BIE	FEIE	PEIE	OEIE	TXCIE	TXDEIE	RXDRIE
类型 / 复位	RW	0 RW	0 RW	0 RW	0 RW	0 RW	0 RW

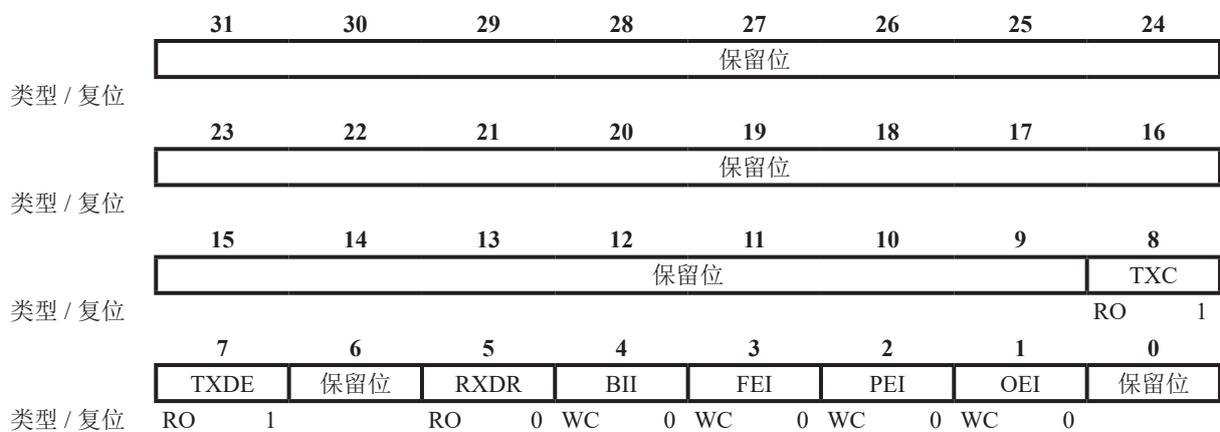
位	字段	描述
[6]	BIE	线中止中断使能 0: 除能中断 1: 使能中断 若此位置位，URSIFR 寄存器中的 BII 位置位且线中止中断使能时将产生中断。
[5]	FEIE	帧错误中断使能 0: 除能中断 1: 使能中断 若此位置位，URSIFR 寄存器中的 FEI 位置位且帧错误中断使能时将产生中断。
[4]	PEIE	奇偶错误中断使能 0: 除能中断 1: 使能中断 若此位置位，URSIFR 寄存器中的 PEI 位置位且奇偶错误中断使能时将产生中断。
[3]	OEIE	溢出错误中断使能 0: 除能中断 1: 使能中断 若此位置位，URSIFR 寄存器中的 OEI 位置位且溢出错误中断使能时将产生中断。

位	字段	描述
[2]	TXCIE	发送完成中断使能 0: 除能中断 1: 使能中断 若此位置位, URSIFR 寄存器中的 TXC 位置位且发送完成中断使能时将产生中断。
[1]	TXDEIE	发送数据空中断使能 0: 除能中断 1: 使能中断 若此位置位, URSIFR 寄存器中的 TXDE 位置位且发送数据寄存器空中断使能时将产生中断。
[0]	RXDRIE	接收数据就绪中断使能 0: 除能中断 1: 使能中断 若此位置位, URSIFR 寄存器中的 RXDR 位置位且接收数据就绪中断使能时将产生中断。

UART 状态 & 中断标志位寄存器 – URSIFR

该寄存器包含了相应的 UART 状态。

偏移量: 0x010
复位值: 0x0000_0180



位	字段	描述
[8]	TXC	发送完成 0: 发送数据寄存器 TDR 或发送移位寄存器 TSR 都不为空 1: 发送数据寄存器 TDR 与发送移位寄存器 TSR 都为空 若 URIER 寄存器中的 TXCIE = 1, 则将产生中断。向 URDR 寄存器写入新数据将清除此标志位。
[7]	TXDE	发送数据寄存器为空 0: 发送数据寄存器不为空 1: 发送数据寄存器为空 当发送数据寄存器 TDR 的内容传输至发送移位寄存器 TSR 时 TXDE 位将置位。若 URIER 寄存器中的 TXEIE = 1 将产生中断, 向 URDR 寄存器写入一个新数据将清除此位。

位	字段	描述
[5]	RXDR	<p>RX 数据就绪标志位</p> <p>0: 接收数据寄存器为空</p> <p>1: 读取接收数据寄存器中的所接收数据就绪</p> <p>当接收移位寄存器 RSR 的内容传输至 URDR 寄存器时 RXDE 位将置位。当从 URDR 寄存器中读取数据时此位清零。若 URIER 寄存器中的 RXDRIE = 1 则产生中断。</p>
[4]	BII	<p>线中止中断指示位</p> <p>当接收数据输入保持为“空白状态”(逻辑 0)的时间长于一个完整字传输所用时间,此位将置位。完整字传输时间包括起始位,数据位,奇偶校验位和停止位的总持续时间。对此位写 1 将清除此标志位。</p>
[3]	FEI	<p>帧错误指示位</p> <p>接收到的字符没有正确的停止位,即检测到最后一个数字位或奇偶校验位后面的停止位为逻辑 0 时,此位置位。对此位写 1 将清除此标志位。</p>
[2]	PEI	<p>奇偶错误指示位</p> <p>接收到的字符没有正确的奇偶校验位时,此位置位。对此位写 1 将清除此标志位。</p>
[1]	OEI	<p>溢出错误指示位</p> <p>当接收数据寄存器已满,且 RX 移位寄存器又收到一个新的字符,将发生溢出错误。当发生溢出错误事件之后,若又接收到新字符,移位寄存器中的数据将会被覆盖,接收移位寄存器中的数据将不传送至接收数据寄存器。OEI 位用于溢出事件发生时的即时显示。对此位写 1 将清除此标志位。</p>

UART 分频器锁存寄存器 – URDLR

该寄存器用来定义 UART 时钟分频比，以产生适当的波特率。

偏移量： 0x024

复位值： 0x0000_0010

	31	30	29	28	27	26	25	24	
类型 / 复位	保留位								
	23	22	21	20	19	18	17	16	
类型 / 复位	保留位								
	15	14	13	12	11	10	9	8	
类型 / 复位	BRD								
	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
类型 / 复位	BRD								
	RW	0	RW	0	RW	0	RW	0	RW

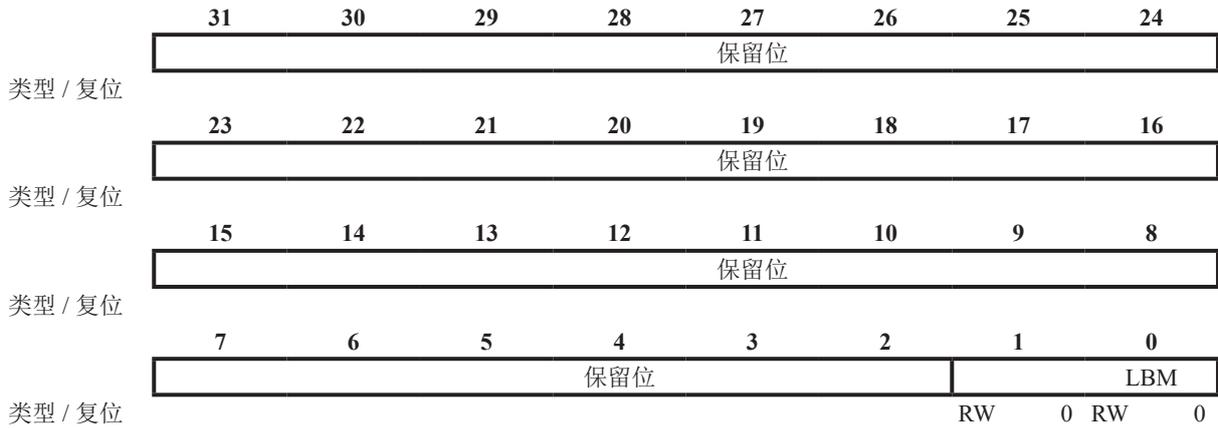
位	字段	描述
[15:0]	BRD	波特率分频器 这 16 位定义了 UART 时钟分频比例。 波特率 = CK_UART / BRD CK_UART 时钟是连接到 UART 模块的系统时钟。 在 UART 模式中 BRD = 16 ~ 65535。

UART 测试寄存器 – URTSTR

该寄存器控制 UART 调试模式。

偏移量: 0x028

复位值: 0x0000_0000



位	字段	描述
[1:0]	LBM	回送测试模式选择 00: 正常工作模式 01: 保留 10: 自动回应模式 11: 回送模式

21 除法器 (DIV)

简介

该系列单片机内置一个除法器以提高单片机性能。除法器可以实现 32 位有符号 / 无符号数据的除法运算。当除数为零时会产生错误标志。

特性

- 32-bit 有符号 / 无符号除法器
- 在 8 个时钟周期内完成计算并且在 1 个时钟周期内完成加载
- 除数为零错误标志

功能描述

舍尾除法的除法函数和取模函数都是通过下面的公式进行运算：

$$A / B = Q \cdots R$$

其中“**A**”为被除数，“**B**”为除数，“**Q**”为商，“**R**”为余数。置高除法器控制寄存器中的 **START** 位将触发除法器开始计算。8 个时钟周期后除法器计算结束，完成标志位将被置高，但若计算时除数寄存器内数据为零，那么除数为零错误标志将被置位。

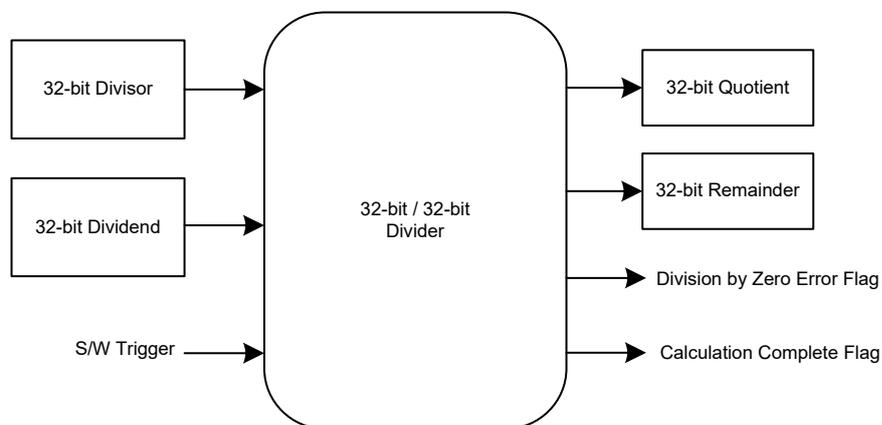


图 123. 除法器功能图

寄存器列表

下表显示了 DIV 寄存器及其复位值。

表 48. DIV 寄存器列表

寄存器	偏移量	描述	复位值
CR	0x000	除法器控制寄存器	0x0000_0008
DDR	0x004	被除数数据寄存器	0x0000_0000
DSR	0x008	除数数据寄存器	0x0000_0000
QTR	0x00C	商数据寄存器	0x0000_0000
RMR	0x010	余数数据寄存器	0x0000_0000

寄存器描述

除法器控制寄存器 – CR

该寄存器包含了除法器触发控制位和计算状态。

偏移量: 0x000

复位值: 0x0000_0008

31	30	29	28	27	26	25	24
保留位							
类型 / 复位							
23	22	21	20	19	18	17	16
保留位							
类型 / 复位							
15	14	13	12	11	10	9	8
保留位							
类型 / 复位							
7	6	5	4	3	2	1	0
保留位				COM	ZEF	保留位	START
类型 / 复位				RO	1 RO	0	RW 0

位	字段	描述
[3]	COM	计算完成且数据有效标志 0: 数据无效 1: 新数据有效 若此位置 1, 表示除法器计算完成且数据有效。此位在计算开始后由硬件清零。
[2]	ZEF	除数为零错误标志 0: 除数不为零 1: 除数为零 该位在计算开始后由硬件清零。
[0]	START	除法器计算开始触发位 0: 无动作 1: 触发除法器开始计算 该位被置高后除法器将开始计算。

被除数数据寄存器 – DDR

该寄存器用来定义被除数。

偏移量: 0x004
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	DDR					
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0
	23	22	21	20	19	18	17	16	DDR					
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0
	15	14	13	12	11	10	9	8	DDR					
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0	DDR					
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0

位	字段	描述
[31:0]	DDR	被除数数据寄存器 此字段用于定义除法器计算的被除数。

除数数据寄存器 – DSR

该寄存器用来定义除数。

偏移量: 0x008
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	DSR					
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0
	23	22	21	20	19	18	17	16	DSR					
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0
	15	14	13	12	11	10	9	8	DSR					
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0	DSR					
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0

位	字段	描述
[31:0]	DSR	除数数据寄存器 此字段用于定义除法器计算的除数。

商数据寄存器 – QTR

该寄存器用来存储商。

偏移量: 0x00C
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
	QTR								
类型 / 复位	RO	0	RO	0	RO	0	RO	0	RO
	23	22	21	20	19	18	17	16	
	QTR								
类型 / 复位	RO	0	RO	0	RO	0	RO	0	RO
	15	14	13	12	11	10	9	8	
	QTR								
类型 / 复位	RO	0	RO	0	RO	0	RO	0	RO
	7	6	5	4	3	2	1	0	
	QTR								
类型 / 复位	RO	0	RO	0	RO	0	RO	0	RO

位	字段	描述
[31:0]	QTR	商数据寄存器 此字段用于存放除法器计算结果的商。

余数数据寄存器 – RMR

该寄存器用来存储余数。

偏移量: 0x010
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
	RMR								
类型 / 复位	RO	0	RO	0	RO	0	RO	0	RO
	23	22	21	20	19	18	17	16	
	RMR								
类型 / 复位	RO	0	RO	0	RO	0	RO	0	RO
	15	14	13	12	11	10	9	8	
	RMR								
类型 / 复位	RO	0	RO	0	RO	0	RO	0	RO
	7	6	5	4	3	2	1	0	
	RMR								
类型 / 复位	RO	0	RO	0	RO	0	RO	0	RO

位	字段	描述
[31:0]	RMR	余数数据寄存器 此字段用于存放除法器计算结果的余数。

Copyright© 2019 by HOLTEK SEMICONDUCTOR INC.

使用指南中所出现的信息在出版当时相信是正确的，然而 **Holtek** 对于说明书的使用不负任何责任。文中提到的应用目的仅仅是用来做说明，**Holtek** 不保证或表示这些没有进一步修改的应用将是适当的，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。**Holtek** 产品不授权使用于救生、维生从机或系统中做为关键从机。**Holtek** 拥有不事先通知而修改产品的权利，对于最新的信息，请参考我们的网址 <http://www.holtek.com/zh/>.