



Holtek 32-Bit 带 Arm® Cortex®-M3 内核单片机

HT32F12364

开发使用手册

Revision: V1.10 Date: 2021-09-28

www.holtek.com

目录

1 简介	24
概述	24
特性	25
单片机信息	29
方框图	30
2 文档协议	31
3 系统结构	32
Arm® Cortex®-M3 处理器	32
总线结构	33
存储器体系	34
存储器映射	35
嵌入式 Flash 存储器	38
嵌入式 SRAM 存储器	38
AHB 外设	38
APB 外设	38
4 Flash 存储器控制器 (FMC)	39
简介	39
特性	39
功能描述	40
Flash 存储器映射	40
Flash 存储器结构	41
等待状态设置	41
启动配置	42
页擦除	43
整片擦除	44
字编程	45
选项字节描述	46
页擦除 / 编程保护	46
安全保护	48
寄存器列表	49
寄存器描述	50
Flash 目标地址寄存器 – TADR	50
Flash 写数据寄存器 – WRDR	51
Flash 操作命令寄存器 – OCMR	52
Flash 操作控制寄存器 – OPCR	53
Flash 操作中断使能寄存器 – OIER	54
Flash 操作中断状态寄存器 – OISR	55
Flash 页擦除 / 编程保护状态寄存器 – PPSR	56
Flash 安全保护状态寄存器 – CPSR	57

Flash 向量映射控制寄存器 – VMCR.....	58
Flash 制造商和设备 ID 寄存器 – MDID.....	59
Flash 页数量状态寄存器 – PNSR	60
Flash 页大小状态寄存器 – PSSR	61
设备 ID 寄存器 – DID.....	62
Flash 缓存和预取控制寄存器 – CFCR	63
SRAM 启动向量寄存器 n – SBVTn, n = 0 ~ 3	64
自定义 ID 寄存器 n – CIDRn, n = 0 ~ 3	65
5 电源控制单元 (PWRCU).....	66
简介	66
特性	67
功能描述	67
V _{DD} 电源域	67
V _{CORE} 电源域	69
工作模式	69
寄存器列表	71
寄存器描述	71
电源控制状态寄存器 – PWRSR.....	71
电源控制寄存器 – PWRCR	72
V _{DD} 电源域测试寄存器 – PWRTTEST.....	74
低电压 / 欠压检测控制和状态寄存器 – LVDCSR.....	75
电源控制 LDO 状态寄存器 – PWRLDOSR	76
6 时钟控制单元 (CKCU).....	77
简介	77
特性	79
功能描述	79
外部高速晶振 – HSE.....	79
内部高速 RC 振荡器 – HSI	80
自动微调内部高速 RC 振荡器 – HSI	80
锁相环 – PLL	82
USB 锁相环 – USB PLL.....	84
外部低速晶振 – LSE	86
内部低速 RC 振荡器 – LSI.....	86
时钟就绪标志位	86
系统时钟选择 – CK_SYS	87
HSE 时钟监测器.....	88
时钟输出能力	88
寄存器列表	88
寄存器描述	89
全局时钟配置寄存器 – GCFGR.....	89
全局时钟控制寄存器 – GCCR	91
全局时钟状态寄存器 – GCSR.....	92

全局时钟中断寄存器 – GCIR.....	93
PLL 配置寄存器 – PLLCFGR.....	95
PLL 控制寄存器 – PLLCR.....	96
AHB 配置寄存器 – AHBCFGR.....	97
AHB 时钟控制寄存器 – AHBCCR.....	98
APB 配置寄存器 – APBCFGR.....	100
APB 时钟控制寄存器 0 – APBCCR0.....	101
APB 时钟控制寄存器 1 – APBCCR1.....	102
时钟源状态寄存器 – CKST.....	104
APB 外设时钟选择寄存器 0 – APBPCSR0.....	105
APB 外设时钟选择寄存器 1 – APBPCSR1.....	107
HSI 控制寄存器 – HSICR.....	109
HSI 自动微调计数器寄存器 – HSIATCR.....	110
APB 外设时钟选择寄存器 2 – APBPCSR2.....	111
低功耗控制寄存器 – LPCR.....	112
单片机调试控制寄存器 – MCUDBGCR.....	113
7 复位控制单元 (RSTCU).....	115
简介.....	115
功能描述.....	116
上电复位.....	116
系统复位.....	116
AHB 和 APB 单元复位.....	116
寄存器列表.....	117
寄存器描述.....	117
全局复位状态寄存器 – GRSR.....	117
AHB 外设复位寄存器 – AHBPRSTR.....	118
APB 外设复位寄存器 0 – APBPRSTR0.....	119
APB 外设复位寄存器 1 – APBPRSTR1.....	121
8 通用 I/O (GPIO).....	122
简介.....	122
特性.....	123
功能描述.....	123
默认的 GPIO 引脚配置.....	123
通用 I/O – GPIO.....	123
GPIO 锁定机制.....	125
寄存器列表.....	125
寄存器描述.....	127
端口 A 数据方向控制寄存器 – PADIRCR.....	127
端口 A 输入功能使能控制寄存器 – PAINER.....	128
端口 A 上拉选择寄存器 – PAPUR.....	129
端口 A 下拉选择寄存器 – PAPDR.....	130
端口 A 漏极开路选择寄存器 – PAODR.....	131

端口 A 驱动电流选择寄存器 – PADRVR	132
端口 A 锁定寄存器 – PALOCKR	133
端口 A 数据输入寄存器 – PADINR	134
端口 A 输出数据寄存器 – PADOUTR	135
端口 A 输出置位 / 复位控制寄存器 – PASRR	136
端口 A 输出复位寄存器 – PARR	137
端口 B 数据方向控制寄存器 – PBDIRCR	138
端口 B 输入功能使能控制寄存器 – PBINER	139
端口 B 上拉选择寄存器 – PBPUR	140
端口 B 下拉选择寄存器 – PBPDR	141
端口 B 漏极开路选择寄存器 – PBODR	142
端口 B 驱动电流选择寄存器 – PBDRVR	143
端口 B 锁定寄存器 – PBLOCKR	144
端口 B 数据输入寄存器 – PBDINR	145
端口 B 输出数据寄存器 – PBDOUTR	146
端口 B 输出置位 / 复位控制寄存器 – PBSRR	147
端口 B 输出复位寄存器 – PBRR	148
端口 C 数据方向控制寄存器 – PCDIRCR	149
端口 C 输入功能使能控制寄存器 – PCINER	150
端口 C 上拉选择寄存器 – PCPUR	151
端口 C 下拉选择寄存器 – PCPDR	152
端口 C 漏极开路选择寄存器 – PCODR	153
端口 C 驱动电流选择寄存器 – PCDRVR	154
端口 C 锁定寄存器 – PCLOCKR	155
端口 C 数据输入寄存器 – PCDINR	156
端口 C 输出数据寄存器 – PCDOUTR	157
端口 C 输出置位 / 复位控制寄存器 – PCSRR	158
端口 C 输出复位寄存器 – PCRR	159
端口 D 数据方向控制寄存器 – PDDIRCR	160
端口 D 输入功能使能控制寄存器 – PDINER	161
端口 D 上拉选择寄存器 – PDPUR	162
端口 D 下拉选择寄存器 – PDPDR	163
端口 D 漏极开路选择寄存器 – PDODR	164
端口 D 驱动电流选择寄存器 – PDDRVR	165
端口 D 锁定寄存器 – PDLOCKR	166
端口 D 数据输入寄存器 – PDDINR	167
端口 D 输出数据寄存器 – PDDOUTR	168
端口 D 输出置位 / 复位控制寄存器 – PDSRR	169
端口 D 输出复位寄存器 – PDRR	170
端口 F 数据方向控制寄存器 – PFDIRCR	171
端口 F 输入功能使能控制寄存器 – PFINER	172
端口 F 上拉选择寄存器 – PFPUR	173
端口 F 下拉选择寄存器 – PFPDR	174
端口 F 漏极开路选择寄存器 – PFODR	175
端口 F 驱动电流选择寄存器 – PFDRVR	176

端口 F 锁定寄存器 – PFLOCKR.....	177
端口 F 数据输入寄存器 – PFDINR.....	178
端口 F 输出数据寄存器 – PFDOUTR.....	179
端口 F 输出置位 / 复位控制寄存器 – PFSRR.....	180
端口 F 输出复位寄存器 – PFRR.....	181
9 复用功能输入 / 输出控制单元 (AFIO).....	182
简介.....	182
特性.....	183
功能描述.....	183
外部中断引脚选择.....	183
复用功能.....	184
锁定机制.....	184
寄存器列表.....	184
寄存器描述.....	185
EXTI 来源选择寄存器 0 – ESSR0.....	185
EXTI 来源选择寄存器 1 – ESSR1.....	186
GPIO 端口 x 配置低寄存器 – GPxCFGRLR, x = A, B, C, D, F.....	187
GPIO 端口 x 配置高寄存器 – GPxCFGHR, x = A, B, C, D, F.....	188
10 嵌套向量中断控制器 (NVIC).....	189
简介.....	189
特性.....	191
功能描述.....	192
SysTick 校准.....	192
寄存器列表.....	192
11 外部中断 / 事件控制器 (EXTI).....	194
简介.....	194
特性.....	194
功能描述.....	195
唤醒事件管理.....	195
外部中断 / 事件线配置.....	196
中断和去抖.....	196
寄存器列表.....	197
寄存器描述.....	198
EXTI 中断 n 配置寄存器 – EXTICFGRn, n = 0 ~ 15.....	198
EXTI 中断控制寄存器 – EXTICR.....	199
EXTI 中断边沿标志寄存器 – EXTIEDGEFLGR.....	200
EXTI 中断边沿状态寄存器 – EXTIEDGESR.....	201
EXTI 中断软件设置命令寄存器 – EXTISSCR.....	202
EXTI 中断唤醒控制寄存器 – EXTIWAKUPCR.....	203
EXTI 中断唤醒极性寄存器 – EXTIWAKUPPOLR.....	204
EXTI 中断唤醒标志寄存器 – EXTIWAKUPFLG.....	205

12 模数转换器 (ADC)	206
简介	206
特性	207
功能描述	207
ADC 时钟设置	207
通道选择	207
转换模式	208
外部事件启动转换	210
采样时间设定	211
数据格式	211
模拟看门狗	211
中断	211
PDMA 请求	212
内部参考电压	212
V _{DDA} 电压监测	212
寄存器列表	213
寄存器描述	214
ADC 转换控制寄存器 – ADCCR	214
ADC 转换列表寄存器 0 – ADCLST0	215
ADC 转换列表寄存器 1 – ADCLST1	216
ADC 输入采样时间寄存器 – ADCSTR	217
ADC 转换数据寄存器 y – ADCDRy, y = 0 ~ 7	218
ADC 触发控制寄存器 – ADCTCR	219
ADC 触发源寄存器 – ADCTSR	220
ADC 看门狗控制寄存器 – ADCWCR	221
ADC 看门狗阈值寄存器 – ADCTR	222
ADC 中断使能寄存器 – ADCIER	223
ADC 中断原始状态寄存器 – ADCIRAW	224
ADC 中断状态寄存器 – ADCISR	225
ADC 中断清除寄存器 – ADCICLR	226
ADC DMA 请求寄存器 – ADCDMAR	227
参考电压控制寄存器 – VREFCR	228
参考电压值寄存器 – VREFVALR	229
13 通用定时器 (GPTM)	230
简介	230
特性	231
功能描述	231
计数器模式	231
时钟控制器	234
触发控制器	235
从机控制器	236
主机控制器	238
通道控制器	239

输入级	242
正交解码器	244
输出级	245
更新管理	249
单脉冲模式	250
非对称 PWM 模式	252
定时器互连	253
触发其它外设功能启动	256
PDMA 请求	256
寄存器列表	257
寄存器描述	258
定时器计数器配置寄存器 – CNTCFR	258
定时器模式配置寄存器 – MDCFR	259
定时器触发配置寄存器 – TRCFR	261
定时器控制寄存器 – CTR	262
通道 0 输入配置寄存器 – CH0ICFR	263
通道 1 输入配置寄存器 – CH1ICFR	264
通道 2 输入配置寄存器 – CH2ICFR	266
通道 3 输入配置寄存器 – CH3ICFR	267
通道 0 输出配置寄存器 – CH0OCFR	268
通道 1 输出配置寄存器 – CH1OCFR	270
通道 2 输出配置寄存器 – CH2OCFR	272
通道 3 输出配置寄存器 – CH3OCFR	274
通道控制寄存器 – CHCTR	276
通道极性配置寄存器 – CHPOLR	277
定时器 PDMA / 中断控制寄存器 – DICTR	278
定时器事件发生器寄存器 – EVGR	279
定时器中断状态寄存器 – INTSR	280
定时器计数器寄存器 – CNTR	282
定时器预分频器寄存器 – PSCR	283
定时器计数器重载寄存器 – CRR	284
通道 0 捕捉 / 比较寄存器 – CH0CCR	285
通道 1 捕捉 / 比较寄存器 – CH1CCR	286
通道 2 捕捉 / 比较寄存器 – CH2CCR	287
通道 3 捕捉 / 比较寄存器 – CH3CCR	288
通道 0 非对称比较寄存器 – CH0ACR	289
通道 1 非对称比较寄存器 – CH1ACR	290
通道 2 非对称比较寄存器 – CH2ACR	291
通道 3 非对称比较寄存器 – CH3ACR	292
14 脉冲宽度调制定时器 (PWM)	293
简介	293
特性	294
功能描述	294
计数器模式	294

时钟控制器	297
触发控制器	298
从机控制器	299
主机控制器	301
通道控制器	302
输出级	302
更新管理	306
单脉冲模式	307
非对称 PWM 模式	309
定时器互连	310
触发其它外设功能启动	313
PDMA 请求	313
寄存器列表	314
寄存器描述	315
定时器计数器配置寄存器 – CNTCFR	315
定时器模式配置寄存器 – MDCFR	316
定时器触发配置寄存器 – TRCFR	318
定时器控制寄存器 – CTR	319
通道 0 输出配置寄存器 – CH0OCFR	320
通道 1 输出配置寄存器 – CH1OCFR	322
通道 2 输出配置寄存器 – CH2OCFR	324
通道 3 输出配置寄存器 – CH3OCFR	326
通道控制寄存器 – CHCTR	328
通道极性配置寄存器 – CHPOLR	329
定时器 PDMA / 中断控制寄存器 – DICTR	330
定时器事件发生器寄存器 – EVGR	331
定时器中断状态寄存器 – INTSR	332
定时器计数器寄存器 – CNTR	333
定时器预分频器寄存器 – PSCR	334
定时器计数器重载寄存器 – CRR	334
通道 0 比较寄存器 – CH0CR	335
通道 1 比较寄存器 – CH1CR	335
通道 2 比较寄存器 – CH2CR	336
通道 3 比较寄存器 – CH3CR	336
通道 0 非对称比较寄存器 – CH0ACR	337
通道 1 非对称比较寄存器 – CH1ACR	338
通道 2 非对称比较寄存器 – CH2ACR	339
通道 3 非对称比较寄存器 – CH3ACR	340
15 单通道定时器 (SCTM)	341
简介	341
特性	342
功能描述	342
计数器模式	342
时钟控制器	343

触发控制器	344
从机控制器	345
通道控制器	347
输入级	349
输出级	350
更新管理	352
寄存器列表	353
寄存器描述	354
定时器计数器配置寄存器 – CNTCFR	354
定时器模式配置寄存器 – MDCFR	355
定时器触发配置寄存器 – TRCFR	356
定时器控制寄存器 – CTR	357
通道输入配置寄存器 – CHICFR	358
通道输出配置寄存器 – CHOCFR	359
通道控制寄存器 – CHCTR	360
通道极性配置寄存器 – CHPOLR	361
定时器中断控制寄存器 – DICTR	362
定时器事件发生器寄存器 – EVGR	363
定时器中断状态寄存器 – INTSR	364
定时器计数器寄存器 – CNTR	365
定时器预分频器寄存器 – PSCR	365
定时器计数器重载寄存器 – CRR	366
通道捕捉 / 比较寄存器 – CHCCR	367
16 基本功能定时器 (BFTM)	368
简介	368
特性	368
功能描述	369
重复模式	369
单次模式	370
寄存器列表	371
寄存器描述	371
BFTM 控制寄存器 – BFTMCR	371
BFTM 状态寄存器 – BFTMSR	372
BFTM 计数器值寄存器 – BFTMCNTR	373
BFTM 比较值寄存器 – BFTMCMPR	374
17 实时时钟 (RTC)	375
简介	375
特性	375
功能描述	376
RTC 相关寄存器复位	376
读取 RTC 寄存器	376
低速时钟配置	376

RTC 计数器操作.....	376
中断和唤醒控制	377
RTCOUNT 输出引脚配置	377
寄存器列表.....	378
寄存器描述.....	379
RTC 计数器寄存器 – RTCCNT	379
RTC 比较寄存器 – RTCCMP.....	380
RTC 控制寄存器 – RTCCR.....	381
RTC 状态寄存器 – RTCSR	383
RTC 中断和唤醒使能寄存器 – RTCIWEN.....	384
18 看门狗定时器 (WDT)	385
简介	385
特性.....	385
功能描述.....	386
寄存器列表.....	388
寄存器描述.....	388
看门狗定时器控制寄存器 – WDTCR.....	388
看门狗定时器模式寄存器 0 – WDTMR0.....	389
看门狗定时器模式寄存器 1 – WDTMR1	390
看门狗定时器状态寄存器 – WDTSR	391
看门狗定时器保护寄存器 – WDTPR	392
看门狗定时器时钟选择寄存器 – WDTCSR.....	393
19 内部集成电路 (I²C).....	394
简介	394
特性.....	395
功能描述.....	395
双线串行接口	395
START 和 STOP 条件	395
数据有效性	396
寻址格式.....	397
数据传输和确认	399
时钟同步	400
仲裁.....	400
广播呼叫寻址	401
总线错误	401
地址屏蔽使能	401
地址捕获	401
工作模式	401
保持 SCL 线状态的条件.....	407
I ² C 超时功能	407
PDMA 接口.....	408
寄存器列表.....	408

寄存器描述	409
I ² C 控制寄存器 – I2CCR	409
I ² C 中断使能寄存器 – I2CIER	411
I ² C 地址寄存器 – I2CADDR	412
I ² C 状态寄存器 – I2CSR	413
I ² C SCL 高电平周期发生寄存器 – I2CSHPGR	416
I ² C SCL 低电平周期产生寄存器 – I2CSLPGR	417
I ² C 数据寄存器 – I2CDR	418
I ² C 目标寄存器 – I2CTAR	419
I ² C 地址屏蔽寄存器 – I2CADDRMR	420
I ² C 地址捕获寄存器 – I2CADDRSR	421
I ² C 超时寄存器 – I2CTOUT	422
20 串行外设接口 (SPI).....	423
简介	423
特性	424
功能描述	424
主机模式	424
从机模式	424
SPI 串行帧格式	425
SPI 双线读取模式	429
状态标志	431
寄存器列表	434
寄存器描述	434
SPI 控制寄存器 0 – SPICR0	434
SPI 控制寄存器 1 – SPICR1	436
SPI 中断使能寄存器 – SPIIER	438
SPI 时钟预分频器寄存器 – SPICPR	439
SPI 数据寄存器 – SPIDR	440
SPI 状态寄存器 – SPISR	441
SPI FIFO 控制寄存器 – SPIFCR	442
SPI FIFO 状态寄存器 – SPIFSR	443
SPI FIFO 超时计数器寄存器 – SPIFTOCR	444
21 通用同步异步收发器 (USART)	445
简介	445
特性	446
功能描述	447
串行数据格式	447
波特率发生器	448
硬件流控制	449
IrDA 模式	451
RS485 模式	453
同步主机模式	454

中断和状态	456
PDMA 接口	456
寄存器列表	456
寄存器描述	457
USART 数据寄存器 – USRDR	457
USART 控制寄存器 – USRCR	458
USART FIFO 控制寄存器 – USRFCR	460
USART 中断使能寄存器 – USRIER	461
USART 状态 & 中断标志位寄存器 – USRSIFR	462
USART 时序参数寄存器 – USRTPR	464
USART IrDA 控制寄存器 – IrDACR	465
USART RS485 控制寄存器 – RS485CR	466
USART 同步控制寄存器 – SYNCR	467
USART 分频器锁存寄存器 – USRDLR	468
USART 测试寄存器 – USRTSTR	469
22 通用异步收发器 (UART)	470
简介	470
特性	471
功能描述	471
串行数据格式	471
波特率发生器	472
中断和状态	473
PDMA 接口	473
寄存器列表	474
寄存器描述	474
UART 数据寄存器 – URDR	474
UART 控制寄存器 – URCR	475
UART 中断使能寄存器 – URIER	476
UART 状态 & 中断标志位寄存器 – URSIFR	477
UART 分频器锁存寄存器 – URDLR	478
UART 测试寄存器 – URTSTR	479
23 智能卡接口 (SCI)	480
简介	480
特性	480
功能描述	481
基本时间单元计数器	481
保护时间计数器	483
等待时间计数器	483
智能卡时钟和数据选择	484
智能卡检测	484
SCI 数据传输模式	485
中断发生器	486

PDMA 接口	487
寄存器列表	487
寄存器描述	488
SCI 控制寄存器 – CR	488
SCI 状态寄存器 – SR	490
SCI 通信控制寄存器 – CCR	491
SCI 基本时间单元寄存器 – ETUR	492
SCI 保护时间寄存器 – GTR	493
SCI 等待时间寄存器 – WTR	494
SCI 中断使能寄存器 – IER	495
SCI 中断挂起寄存器 – IPR	496
SCI 发送缓冲器 – TXB	497
SCI 接收缓冲器 – RXB	498
SCI 预分频器寄存器 – PSCR	498
24 USB 设备控制器 (USB)	499
简介	499
特性	499
功能描述	500
端点	500
EP_SRAM	500
串行接口引擎 – SIE	501
双缓冲	501
挂起模式和唤醒	503
远程唤醒	503
寄存器列表	503
寄存器描述	505
USB 控制和状态寄存器 – USBCSR	505
USB 中断使能寄存器 – USBIER	507
USB 中断状态寄存器 – USBISR	508
USB 帧计数寄存器 – USBFCR	509
USB 设备地址寄存器 – USBDEVAR	510
USB 端点 0 控制和状态寄存器 – USBEP0CSR	511
USB 端点 0 中断使能控制寄存器 – USBEP0IER	512
USB 端点 0 中断状态寄存器 – USBEP0ISR	513
USB 端点 0 传输计数寄存器 – USBEP0TCR	515
USB 端点 0 配置寄存器 – USBEP0CFGR	516
USB 端点 1~3 控制和状态寄存器 – USBEPnCSR, n = 1~3	517
USB 端点 1~3 中断使能寄存器 – USBEPnIER, n = 1~3	518
USB 端点 1~3 中断状态寄存器 – USBEPnISR, n = 1~3	519
USB 端点 1~3 传输计数寄存器 – USBEPnTCR, n = 1~3	520
USB 端点 1~3 配置寄存器 – USBEPnCFGR, n = 1~3	521
USB 端点 4~7 控制和状态寄存器 – USBEPnCSR, n = 4~7	522
USB 端点 4~7 中断使能寄存器 – USBEPnIER, n = 4~7	524

USB 端点 4~7 中断状态寄存器 – USBEPnISR, n = 4~7	525
USB 端点 4~7 传输计数寄存器 – USBEPnTCR, n = 4~7	526
USB 端点 4~7 配置寄存器 – USBEPnCFGR, n = 4~7	527
25 外设直接存储器访问 (PDMA).....	528
简介	528
特性	528
功能描述	529
AHB 主机	529
PDMA 通道	529
PDMA 请求映射	529
通道传输	530
通道优先级	530
传输请求	531
地址模式	531
自动重载	532
传输中断	532
寄存器列表	532
寄存器描述	533
PDMA 通道 n 控制寄存器 – PDMACHnCR, n = 0~5	533
PDMA 通道 n 源地址寄存器 – PDMACHnSADR, n = 0~5	535
PDMA 通道 n 目标地址寄存器 – PDMACHnDADR, n = 0~5	536
PDMA 通道 n 传输大小寄存器 – PDMACHnTSR, n = 0~5	537
PDMA 通道 n 当前传输大小寄存器 – PDMACHnCTSR, n = 0~5	538
PDMA 中断状态寄存器 – PDMAISR	539
PDMA 中断状态清除寄存器 – PDMAISCR	540
PDMA 中断使能寄存器 – PDMAIER	541
26 外部总线接口 (EBI).....	542
简介	542
特性	542
功能描述	543
非复用 8-bit 数据 8-bit 地址模式	544
非复用 16-bit 数据 N-bit 地址模式	545
复用 16-bit 数据 16-bit 地址模式	546
复用 8-bit 数据 20-bit 地址模式	547
写缓冲器和 EBI 状态	548
总线周转和空闲周期	548
AHB 事务宽度转换	549
EBI Bank 访问	551
PDMA 请求	552
寄存器列表	552
寄存器描述	552
EBI 控制寄存器 – EBICR	552

EBI 状态寄存器 – EBISR	554
EBI 地址时序寄存器 – EBIATR.....	555
EBI 读时序寄存器 – EBIRTR.....	556
EBI 写时序寄存器 – EBIWTR	557
EBI 极性寄存器 – EBIPR	558
27 循环冗余校验 (CRC)	559
简介	559
特性	560
功能描述	560
CRC 计算	560
CRC 计算按字节反序与位反序	560
CRC 与 PDMA 搭配工作.....	561
寄存器列表	561
寄存器描述	562
CRC 控制寄存器 – CRCCR.....	562
CRC 种子值寄存器 – CRCSDR	563
CRC 校验和寄存器 – CRCCSR.....	564
CRC 数据寄存器 – CRCDR.....	565
28 AES 加密 / 解密接口 (AES)	566
简介	566
特性	566
功能描述	567
AES 模式.....	567
AES 状态.....	570
AES PDMA 接口	570
AES 中断.....	570
AES 初始矢量.....	570
AES 字交换.....	571
寄存器列表	571
寄存器描述	572
AES 控制寄存器 – AESCR.....	572
AES 状态寄存器 – AESSR	573
AES DMA 寄存器 – AESDMAR	574
AES 中断状态寄存器 – AESISR	575
AES 中断使能寄存器 – AESIER.....	576
AES 数据输入寄存器 – AESDINR	577
AES 数据输出寄存器 – AESDOUTR	577
AES 密钥寄存器 n – AESKEYRn, n = 0 ~ 7	578
AES 初始矢量寄存器 n – AESIVRn, n = 0 ~ 3	578

表列表

表 1. 单片机的特性及外设列表	29
表 2. 文档协议	31
表 3. 寄存器列表	36
表 4. Flash 存储器和选项字节	41
表 5. 等待状态周期与 HCLK 之间的关系	41
表 6. 启动模式	42
表 7. 选项字节的存储器映射	46
表 8. 主 Flash 受保护页的访问权限	47
表 9. 安全保护使能时的访问权限	48
表 10. FMC 的寄存器列表	49
表 11. 工作模式定义	69
表 12. 进入 / 退出省电模式	70
表 13. 系统复位后的电源状态	71
表 14. PWRCU 寄存器列表	71
表 15. 输出分频器 2 值映射	83
表 16. 反馈分频器 2 值映射	83
表 17. USB PLL 输出分频器 2 值映射	85
表 18. USB PLL 反馈分频器 2 值映射	85
表 19. CKOUT 的时钟源	88
表 20. CKCU 寄存器列表	88
表 21. RSTCU 寄存器列表	117
表 22. AFIO、GPIO 和 I/O 引脚控制信号真值表	124
表 23. GPIO 寄存器列表	125
表 24. 外设 AFIO 分配列表范例	184
表 25. AFIO 寄存器列表	184
表 26. 异常类型	189
表 27. NVIC 寄存器列表	192
表 28. EXTI 寄存器列表	197
表 29. ADCDR[15:0] 寄存器中的数据格式	211
表 30. A/D 转换器寄存器列表	213
表 31. 计数方向和编码信号	245
表 32. 比较匹配输出设置	246
表 33. GPTM 寄存器列表	257
表 34. GPTM 内部触发器连接	261
表 35. 比较匹配输出设置	303
表 36. PWM 寄存器列表	314
表 37. PWM 内部触发器连接	318
表 38. 比较匹配输出设置	350
表 39. SCTM 寄存器列表	353

表 40. BFTM 寄存器列表	371
表 41. LSE 在不同启动模式下的工作电流和启动时间	376
表 42. RTCOUT 输出模式和有效电平设置	377
表 43. RTC 寄存器列表	378
表 44. 看门狗定时器寄存器列表	388
表 45. 保持 SCL 线状态的条件	407
表 46. I ² C 寄存器列表	408
表 47. I ² C 时钟设置范例	417
表 48. SPI 接口格式设置	425
表 49. SPI 模式故障触发条件	432
表 50. SPI 主机模式 SPI_SEL 引脚状态	432
表 51. SPI 寄存器列表	434
表 52. 波特率误差计算 – CK_USART = 40 MHz	448
表 53. 波特率误差计算 – CK_USART = 48 MHz	449
表 54. 波特率误差计算 – CK_USART = 60 MHz	449
表 55. USART 寄存器列表	456
表 56. 波特率误差计算 – CK_UART = 40 MHz	472
表 57. 波特率误差计算 – CK_UART = 48 MHz	473
表 58. 波特率误差计算 – CK_UART = 60 MHz	473
表 59. UART 寄存器列表	474
表 60. 基于 DI 位域编码的 Di 十进制值	481
表 61. 基于 FI 位域编码的 Fi 十进制值	481
表 62. 由 Fi/Di 比例计算得到的 ETU 可能值	482
表 63. SCI 寄存器列表	487
表 64. 端点特性	500
表 65. USB 数据类型和缓冲器容量	500
表 66. USB 寄存器列表	503
表 67. 恢复事件检测	506
表 68. PDMA 通道配置	530
表 69. PDMA 地址模式	531
表 70. PDMA 寄存器列表	532
表 71. AHB 事务宽度与外部设备事务的 EBI 映射	550
表 72. AHB 事务宽度与外部设备事务宽度的 EBI 映射	550
表 73. EBI 寄存器列表	552
表 74. CRC 寄存器列表	561
表 75. AES 寄存器列表	571

图列表

图 1. 方框图	30
图 2. Cortex®-M3 方框图	33
图 3. 总线结构	34
图 4. 存储器映射	35
图 5. Flash 存储器控制器方框图	39
图 6. Flash 存储器映射	40
图 7. 向量重映射	42
图 8. 页擦除操作流程	43
图 9. 整片擦除操作流程	44
图 10. 字编程操作流程	45
图 11. PWRCU 方框图	66
图 12. 上电复位 / 掉电复位波形	68
图 13. CKCU 方框图	78
图 14. HSE 外部晶体、陶瓷和谐振器	79
图 15. HSI 自动微调方框图	81
图 16. PLL 方框图	82
图 17. USB PLL 方框图	84
图 18. LSE 外部晶体、陶瓷和谐振器	86
图 19. RSTCU 方框图	115
图 20. 上电复位时序图	116
图 21. GPIO 方框图	122
图 22. AFIO/GPIO 控制信号	124
图 23. AFIO 方框图	182
图 24. EXTI 输入通道选择	183
图 25. EXTI 方框图	194
图 26. EXTI 唤醒事件管理	195
图 27. 唤醒中断服务子程序	196
图 28. EXTI 中断去抖功能	196
图 29. 带参考电压 V_{REF} 的 ADC 方框图	206
图 30. 单次转换模式	208
图 31. 连续转换模式	209
图 32. 非连续转换模式	210
图 33. 内部参考电压方框图	212
图 34. GPTM 方框图	230
图 35. 向上计数范例	232
图 36. 向下计数范例	232
图 37. 中心对齐计数范例	233
图 38. GPTM 时钟源选择	234
图 39. 触发控制器方框图	235

图 40. 从机控制器方框图	236
图 41. 重启模式下的 GPTM	236
图 42. 暂停模式下的 GPTM	237
图 43. 触发模式下的 GPTM	237
图 44. 主机 GPTMn 和从机 GPTMm/TMm 相连接	238
图 45. MTO 选项	238
图 46. 捕捉 / 比较方框图	239
图 47. 输入捕捉模式	240
图 48. PWM 脉冲宽度测量范例	241
图 49. 通道 0 和通道 1 输入级	242
图 50. 通道 2 和通道 3 输入级	243
图 51. TI0 数字滤波器方框图 (N = 2)	243
图 52. 输入级和正交解码器方框图	244
图 53. TI0 和 TI1 正交解码器计数	245
图 54. 输出级方框图	245
图 55. 翻转模式通道输出参考信号 – CHxPRE = 0	246
图 56. 翻转模式通道输出参考信号 – CHxPRE = 1	247
图 57. PWM 模式通道输出参考信号和计数器向上计数模式	247
图 58. PWM 模式通道输出参考信号和计数器向下计数模式	248
图 59. PWM 模式通道输出参考信号和计数器中心对齐计数模式	248
图 60. 更新事件设置方框图	249
图 61. 单脉冲模式	250
图 62. 立即有效模式的延迟	251
图 63. 非对称 PWM 模式与中心对齐计数模式	252
图 64. 用 GPTM CH0OREF 信号暂停 PWM0	253
图 65. 用 GPTM 更新事件触发 PWM0	254
图 66. 用 GPTM CH0 输入信号触发 GPTM 和 PWM0	255
图 67. GPTM PDMA 映射图	256
图 68. PWM 方框图	293
图 69. 向上计数范例	294
图 70. 向下计数范例	295
图 71. 中心对齐计数范例	296
图 72. PWM 时钟源选择	297
图 73. 触发控制器方框图	298
图 74. 从机控制器方框图	299
图 75. 重启模式下的 PWM	299
图 76. 暂停模式下的 PWM	300
图 77. 触发模式下的 PWM	300
图 78. 主机 PWMn 和从机 PWMm / TMm 相连接	301
图 79. MTO 选项	301
图 80. 比较方框图	302

图 81. 输出级方框图	302
图 82. 翻转模式通道输出参考信号 – CHxPRE = 0	303
图 83. 翻转模式通道输出参考信号 – CHxPRE = 1	304
图 84. PWM 模式通道输出参考信号和计数器向上计数模式	304
图 85. PWM 模式通道输出参考信号和计数器向下计数模式	305
图 86. PWM 模式通道输出参考信号和计数器中心对齐计数模式	305
图 87. 更新事件设置方框图	306
图 88. 单脉冲模式	307
图 89. 立即有效模式的延迟	308
图 90. 非对称 PWM 模式与中心对齐计数模式	309
图 91. 用 PWM0 CH0OREF 信号暂停 PWM1	310
图 92. 用 PWM0 更新事件触发 PWM1	311
图 93. 用 PWM0 使能信号触发 PWM0 和 PWM1	312
图 94. PWM PDMA 映射图	313
图 95. SCTM 方框图	341
图 96. 向上计数范例	342
图 97. SCTM 时钟源选择	343
图 98. 触发控制器方框图	344
图 99. 从机控制器方框图	345
图 100. 重启模式下的 SCTM	345
图 101. 暂停模式下的 SCTM	346
图 102. 触发模式下的 SCTM	346
图 103. 捕捉 / 比较方框图	347
图 104. 输入捕捉模式	348
图 105. 通道输入级	349
图 106. TI 数字滤波器方框图 (N = 2)	349
图 107. 输出级方框图	350
图 108. 翻转模式通道输出参考信号 – CHPRE = 0	351
图 109. 翻转模式通道输出参考信号 – CHPRE = 1	351
图 110. PWM 模式通道输出参考信号	352
图 111. 更新事件设置方框图	353
图 112. BFTM 方框图	368
图 113. BFTM – 重复模式	369
图 114. BFTM – 单次模式	370
图 115. BFTM – 单次模式计数器更新	370
图 116. RTC 方框图	375
图 117. 看门狗定时器方框图	385
图 118. 看门狗定时器的运转	387
图 119. I ² C 模块方框图	394
图 120. START 和 STOP 条件	396
图 121. 数据有效性	396

图 122. 7-bit 寻址模式.....	397
图 123. 10-bit 寻址写发送模式.....	398
图 124. 10-bit 寻址读接收模式.....	398
图 125. I ² C 总线确认.....	399
图 126. 仲裁期间时钟同步.....	400
图 127. 两个主机仲裁程序.....	400
图 128. 主机发送时序图.....	402
图 129. 主机接收时序图.....	404
图 130. 从机发送时序图.....	405
图 131. 从机接收时序图.....	406
图 132. SCL 时序图.....	417
图 133. SPI 方框图.....	423
图 134. SPI 单个字节传输时序图 – CPOL = 0, CPHA = 0.....	425
图 135. SPI 连续传输时序图 – CPOL = 0, CPHA = 0.....	426
图 136. SPI 单个字节传输时序图 – CPOL = 0, CPHA = 1.....	426
图 137. SPI 连续传输时序图 – CPOL = 0, CPHA = 1.....	427
图 138. SPI 单个字节传输时序图 – CPOL = 1, CPHA = 0.....	427
图 139. SPI 连续数据传输时序图 – CPOL = 1, CPHA = 0.....	428
图 140. SPI 单个字节传输时序图 – CPOL = 1, CPHA = 1.....	428
图 141. SPI 连续数据传输时序图 – CPOL = 1, CPHA = 1.....	428
图 142. SPI 双线读取模式时序图 – CPOL = 0, CPHA = 0, DFL = 0x8 (16-bit), MSB 先传.....	429
图 143. SPI 双线读取模式时序图 – CPOL = 0, CPHA = 1, DFL = 0x8 (16-bit), MSB 先传.....	429
图 144. SPI 双线读取模式时序图 – CPOL = 1, CPHA = 0, DFL = 0x8 (16-bit), MSB 先传.....	430
图 145. SPI 双线读取模式时序图 – CPOL = 1, CPHA = 1, DFL = 0x8 (16-bit), MSB 先传.....	430
图 146. SPI 双线模式读取数据范例 – CPOL = 1, CPHA = 1.....	431
图 147. SPI 多主机从机环境.....	432
图 148. USART 方框图.....	445
图 149. USART 串行数据格式.....	447
图 150. USART 时钟 CK_USART 和数据帧时序.....	448
图 151. 两个 USART 设备之间的硬件流控制.....	449
图 152. USART RTS 流控制.....	450
图 153. USART CTS 流控制.....	450
图 154. IrDA 调制与解调.....	451
图 155. USART I/O 和 IrDA 方框图.....	452
图 156. RS485 接口和波形.....	453
图 157. USART 同步传输范例.....	454
图 158. 8-bit 格式 USART 同步波形.....	455
图 159. UART 方框图.....	470
图 160. UART 串行数据格式.....	471
图 161. UART 时钟 CK_UART 和数据帧时序.....	472
图 162. SCI 方框图.....	480

图 163. 字符帧和补偿模式	482
图 164. 保护时间	483
图 165. 字符和块等待时间 – CWT 和 BWT.....	484
图 166. SCI 卡检测方框图	485
图 167. SCI 中断结构	487
图 168. USB 方框图	499
图 169. 端点缓冲器分配范例	501
图 170. 双缓冲工作范例	502
图 171. PDMA 方框图	528
图 172. PDMA 请求映射架构	529
图 173. PDMA 通道仲裁和安排范例	531
图 174. EBI 方框图	543
图 175. EBI 非复用 8-bit 数据 8-bit 地址读操作	544
图 176. EBI 非复用 8-bit 数据 8-bit 地址写操作	544
图 177. EBI 非复用 16-bit 数据 N-bit 地址读操作	545
图 178. EBI 非复用 16-bit 数据 N-bit 地址写操作	545
图 179. EBI 地址锁存器设置方框图	546
图 180. EBI 复用 16-bit 数据 16-bit 地址读操作	546
图 181. EBI 复用 16-bit 数据 16-bit 地址写操作	547
图 182. EBI 复用 8-bit 数据 20-bit 地址读操作	547
图 183. EBI 复用 8-bit 数据 20-bit 地址写操作	548
图 184. EBI 在同一个 bank 的各个事务之间插入一个 IDLE 周期 (NOIDLE = 0)	549
图 185. EBI 在同一个 bank 的各个事务之间不插入一个 IDLE 周期 (NOIDLE = 1)	549
图 186. EBI Bank 存储器映射	551
图 187. CRC 方框图	559
图 188. CRC 数据按字节反序和按位反序范例	561
图 189. AES 方框图	566
图 190. AES-ECB 模式	567
图 191. AES-CBC 模式	568
图 192. AES-CTR 模式	569
图 193. AES 中断	570
图 194. CTR 模式初始矢量	570
图 195. AES 字交换功能	571

1 简介

概述

本用户手册提供的详细资料, 包括如何使用 HT32F12364 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、应用开发人员和硬件开发人员。欲了解更多有关引脚配置、封装和电气特性的信息, 请参考该单片机数据手册 (Datasheet)。

该单片机是一款基于 Arm® Cortex®-M3 处理器内核的 32 位高性能低功耗单片机。Cortex®-M3 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。

该单片机可工作在高达 72 MHz 频率下, 通过借助 Flash 加速器以获得较大的效能。它提供多达 256 KB 的嵌入式 Flash 存储器用作代码 / 数据存储, 128 KB 的嵌入式 SRAM 存储器用作系统操作和应用程序运用。此单片机具有多种外设, 如 ADC、I²C、USART、UART、SPI、SCI、PDMA、GPTM、PWM、SCTM、EBI、CRC-16/32、AES-128/256、USB2.0 FS 和 SW-DP (串行线调试端口) 等。提供了几种省电模式, 在唤醒延迟和功耗方面具有较优化的灵活性, 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、指纹识别、智能门锁等。

特性

- 内核
 - 32-bit Arm® Cortex®-M3 处理器内核
 - 高达 72 MHz 的工作频率
 - 单周期乘法和硬件除法器
 - 内建嵌套向量中断控制器 (NVIC)
 - 24-bit SysTick 定时器
- 片上存储器
 - 高达 256 KB 片上 Flash 存储器用作指令 / 数据和选项存储
 - 128 KB 片上 SRAM
 - 支持多种启动模式
- Flash 存储器控制器 – FMC
 - Flash 加速器用以提升效能
 - 具有在系统编程 (ISP) 和在应用编程 (IAP) 的 32 位字编程功能
 - 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 ~ 16 MHz 晶振
 - 外部 32.768 kHz 晶振
 - 在工作电压为 3.3 V, 工作温度为 25 °C 下, 内部 8 MHz RC 振荡器精度度可调整为 $\pm 1.5\%$
 - 内部 32 kHz RC 振荡器
 - 集成的系统时钟 PLL 与 USB PLL
 - 用作外设时钟源的独立的时钟分频和门控位
- 电源管理 – PWRCU
 - 单个 V_{DD} 电源: 1.65 V ~ 3.6 V
 - 集成的 1.5 V LDO 稳压器用作 MCU 内核、外设和存储器电源
 - V_{DD} 电源供电给 RTC
 - 两个电源域: V_{DD} 和 V_{CORE}
 - 四种省电模式: 休眠模式、深度休眠模式 1、深度休眠模式 2、暂停模式
- 外部中断 / 事件控制器 – EXTI
 - 高达 16 个可配置触发源和触发类型的 EXTI 输入线
 - 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
 - 每个 EXTI 输入线都可单独进行中断使能、唤醒和状态位设置
 - 每个 EXTI 输入线都有软件中断触发模式
 - 集成抗干扰滤波器, 用于短脉冲封锁

- 模数转换器 – ADC
 - 12-bit SAR ADC 内核
 - 最大 1 Msps 转换速率
 - 最多 8 个外部模拟输入通道
- 输入 / 输出 – GPIO
 - 多达 52 个通用输入 / 输出
 - 端口 A ~ D、F 映射为 16 个外部中断 – EXTI
 - 几乎所有的 I/O 引脚可配置输出驱动电流
- 通用定时器 – GPTM
 - 16-bit 向上 / 向下自动重载计数器
 - 多达 4 个独立通道用于每个定时器
 - 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
 - 捕捉输入功能
 - 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
 - 内建正交解码器带两个输入接口
- 脉冲宽度调制定时器 – PWM
 - 16-bit 向上 / 向下自动重载计数器
 - 每个定时器多达 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
 - 单脉冲输出模式
- 单通道定时器 – SCTM
 - 16-bit 向上自动重载计数器
 - 每个定时器具有 1 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
 - 输入捕捉功能
 - 比较匹配输出
 - PWM 波形产生功能，具有边沿对齐计数模式
 - 单脉冲输出模式
- 基础功能定时器 – BFTM
 - 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
 - 用于系统的复位事件
 - 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 寻址模式和扩播呼叫寻址
 - 支持带有屏蔽地址的从机多寻址模式
- 串行外设接口 – SPI
 - 支持主机和从机模式
 - 主机模式频率高达 $f_{\text{PCLK}}/2$ MHz，从机模式频率高达 $f_{\text{PCLK}}/3$ MHz
 - FIFO 深度：8 级
 - 多个主机和多个从机操作
- 通用同步异步收发器 – USART
 - 同时支持异步和时钟同步串行通信模式
 - 异步工作波特率时钟频率高达 $f_{\text{PCLK}}/16$ MHz，同步工作波特率时钟频率高达 $f_{\text{PCLK}}/8$ MHz
 - 全双工通信能力
 - 完全可编程串行通信功能包括：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
 - 支持自动硬件流控制模式 – RTS、CTS
 - IrDA SIR 编码器和解码器
 - 带有输出使能控制的 RS485 模式
 - FIFO 深度：8 级，用于发送和接收
- 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 $f_{\text{PCLK}}/16$ MHz
 - 全双工通信能力
 - 完全可编程串行通信功能包括：字长、校验位、停止位和位顺序
 - 错误检测：奇偶检验、溢出和帧错误
- 智能卡接口 – SCI
 - 支持 ISO 7816-3 标准
 - 字符传输模式
 - 一个发送缓冲器和一个接收缓冲器
 - 11-bit 基本时间单元 (ETU) 计数器
 - 9-bit 保护时间计数器
 - 24-bit 通用等待时间计数器
 - 奇偶校验位生成和检测功能
 - 发送和接收模式下检测到奇偶校验错误时字符自动重发
- 循环冗余校验 – CRC
 - 支持 CRC16 多项式： $0x8005$ ， $X^{16}+X^{15}+X^2+1$
 - 支持 CCITT CRC16 多项式： $0x1021$ ， $X^{16}+X^{12}+X^5+1$
 - 支持 IEEE-802.3 CRC32 多项式： $0x04C11DB7$ ， $X^{32}+X^{26}+X^{23}+X^{22}+X^{16}+X^{12}+X^{11}+X^{10}+X^8+X^7+X^5+X^4+X^2+X+1$
 - 支持对数据和校验和进行反码、字节反序和位反序操作

- 支持字节、半字和字数据大小
- 可编程 CRC 初始种子值
- 对 8-bit 数据执行 CRC 计算需要 1 个 AHB 时钟周期，32-bit 数据需要 4 个 AHB 时钟周期
- 支持 PDMA 完成指定存储器区块的 CRC 计算
- 外设直接存储器访问 – PDMA
 - 6 通道含触发源组
 - 数据传输宽度：8/16/32-bit
 - 支持线性地址、环形地址和固定地址模式
 - 4 级可编程通道优先级
 - 自动重载模式
 - 支持触发源：
ADC、SPI、USART、UART、I²C、SCI、GPTM、PWM、AES 和软件请求
- 外部总线接口 – EBI
 - 可编程接口适用于各种存储器类型
 - 将 AHB 传输协议转换成适当的外部设备协议
 - 每一个存储器 Bank 有独立的片选控制
 - 可编程时序，支持多种设备
 - 当 AHB 事务宽度和外部存储器接口宽度不同时，可自动转换
 - 写缓存器可减少因 AHB 写突发事件而停滞的状况
 - 支持复用和非复用的地址和数据线配置
 - ◆ 多达 21 条地址线
 - ◆ 多达 16-bit 数据总线宽度
- 通用串行总线设备控制器 – USB
 - 兼容 USB 2.0 全速模式 (12 Mbps) 设备的技术规范
 - 片上 USB 全速收发器
 - 1 个用于控制传输的控制端点 (EP0)
 - 3 个用于批量和中断传输的单缓冲端点 (EP1 ~ EP3)
 - 4 个用于批量、中断和等时传输的双缓冲端点 (EP4 ~ EP7)
 - 1024-byte EP_SRAM 作为端点数据缓冲器
- 高级加密标准 – AES
 - 支持 AES 加密 / 解密功能
 - 支持 AES ECB / CBC / CTR 三种模式
 - 支持 128 位、192 位和 256 位密钥长度
 - 支持 4 字长度的初始矢量用于 CBC 和 CTR 模式
 - 8 × 32 位 (输入 / 输出 FIFO) 用于 2 个 AES 数据块
 - 支持按字的数据交换功能
 - 支持 PDMA 接口
- 调试支持
 - 串行线调试端口 – SW-DP
 - 6 个指令比较器和 2 个用于硬件断点或程序 / 文字补丁的比较器
 - 4 个用于硬件数据观察点的比较器
 - 1-bit 异步追踪用于串行线调试模式 – TRACESWO

- 封装和工作温度
 - 40-pin QFN 和 48/64-pin LQFP 封装
 - 工作温度范围：-40 °C ~ +85 °C

单片机信息

表 1. 单片机的特性及外设列表

外设		HT32F12364
主 Flash (KB)		255
选项字节 Flash (KB)		1
SRAM (KB)		128
定时器	MCTM	1
	GPTM	1
	SCTM	2
	BFTM	2
	RTC	1
	WDT	1
通信	USB	1
	USART	1
	UART	2
	SPI	2
	I ² C	2
	SCI	1
PDMA		6 通道
AES		1
EBI		1
CRC		1
GPIO		52
EXTI		16
12-bit ADC		1
通道总数		8 通道 (Max.)
比较器		2
CPU 频率		高达 72 MHz
工作电压		1.65 V ~ 3.6 V
工作温度		-40 °C ~ +85 °C
封装		40-pin QFN 和 48/64-pin LQFP



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位域。例如，ADDR [5] 指的是 ADDR 寄存器 (位域) 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位域。例如，ADDR [11:5] 指的是 ADDR 寄存器 (位域) 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	<div> <div>19</div> <div>18</div> <div>SERDYIE</div> <div>PLLRDYIE</div> <div>RW 0</div> <div>RW 0</div> </div>	软件可以读取或写入该位。
RO	<div> <div>3</div> <div>2</div> <div>HSIRDY</div> <div>HSERDY</div> <div>RO 1</div> <div>RO 0</div> </div>	软件只能读该位。写操作将不会有任何效果。
RC	<div> <div>1</div> <div>0</div> <div>PDF</div> <div>BAK_PORF</div> <div>RC 0</div> <div>RC 1</div> </div>	软件只能读该位。读操作将自动清零该位。
WC	<div> <div>3</div> <div>2</div> <div>SERDYF</div> <div>PLLRDYF</div> <div>WC 0</div> <div>WC 0</div> </div>	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。
W0C	<div> <div>1</div> <div>0</div> <div></div> <div>MIF</div> <div>W0C 0</div> </div>	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。
WO	<div> <div>31</div> <div>30</div> <div></div> <div>DB_CKSRC</div> <div>WO 0</div> <div>WO 0</div> </div>	软件只能写该位。读操作将总是返回 0。
保留位	<div> <div>1</div> <div>0</div> <div>LLRDY</div> <div>Reserved</div> <div>RO 0</div> </div>	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。
Word (字)		一个字的数据长度为 32-bit。
Half-word (半字)		一个半字的数据长度为 16-bit。
Byte (字节)		一个字节的数据长度为 8-bit。

3 系统结构

该单片机的系统结构包括 Arm® Cortex®-M3 处理器、总线结构和存储器体系，将在以下各节中描述。Cortex®-M3 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®-M3 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®-M3 处理器包含三条 AHB-Lite 总线，即 ICode 总线、DCode 总线和系统总线。根据不同的目的和目标存储空间，Cortex®-M3 处理器访问所有存储器都是在这三条总线上执行的。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®-M3 处理器

Cortex®-M3 处理器是一种通用的 32-bit 处理器内核，特别适合于要求高性能和低功耗单片机的产品。它提供了许多新的特性，如 Thumb-2 指令集、硬件除法器、低延迟中断响应时间、原子位带访问和适用于同时访问的多条总线。Cortex®-M3 处理器基于 ARMv7 结构，同时支持 Thumb 和 Thumb-2 指令集。下面列出了一些也是由 Cortex®-M3 提供的系统外设：

- 内部总线矩阵连接的 ICode 总线、DCode 总线、系统总线、专用外设总线 (PPB) 和调试访问 (AHB-AP)
- 嵌套向量中断控制器 (NVIC)
- Flash 补丁和断点 (FPB)
- 数据观察点及追踪 (DWT)
- 仪器跟踪宏单元 (ITM)
- 存储器保护单元 (MPU)
- 串行线调试端口 (SW-DP)
- 嵌入式跟踪宏单元 (ETM)
- 跟踪端口接口单元 (TPIU)

下图为 Cortex®-M3 处理器的方框图。欲了解更多信息，请参考 Arm® Cortex®-M3 技术参考手册。

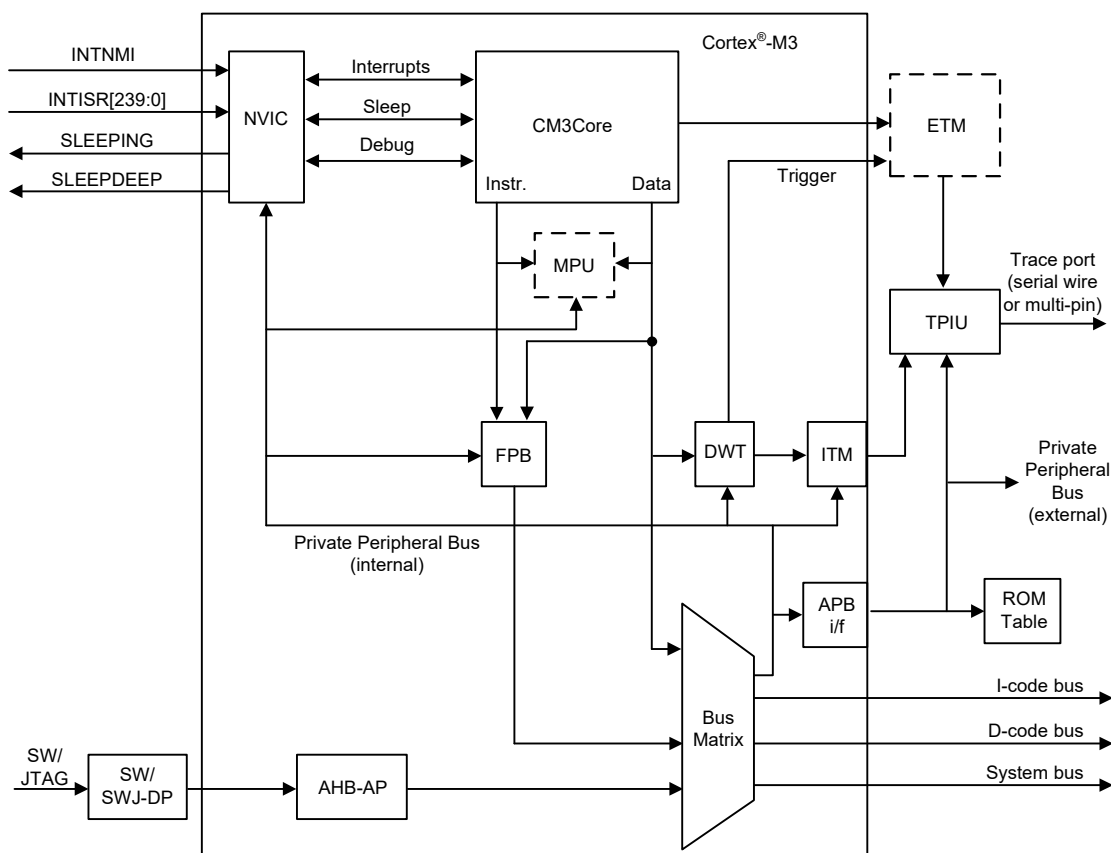


图 2. Cortex®-M3 方框图

总线结构

该单片机的总线结构包括四个主机总线和六个从机总线。Cortex®-M3 ICode、DCode、系统总线 and 外设直接存储器访问 (PDMA) 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、外部总线接口和两个 AHB 到 APB 总线桥为从机总线。ICode 总线用于从代码区域 (0x0000_0000 ~ 0x1FFF_FFFF) 取指令和向量到 Cortex®-M3 内核。DCode 总线用于对代码区域进行数据加载 / 存储以及调试访问。同样地，系统总线用于对系统区域执行取指令和向量、数据加载 / 存储以及调试访问。系统区域包括内部 SRAM 区域和外设区域。主机总线都是基于 32-bit 先进的高性能总线精简 (AHB-Lite) 协议。下图显示了该单片机的总线结构。

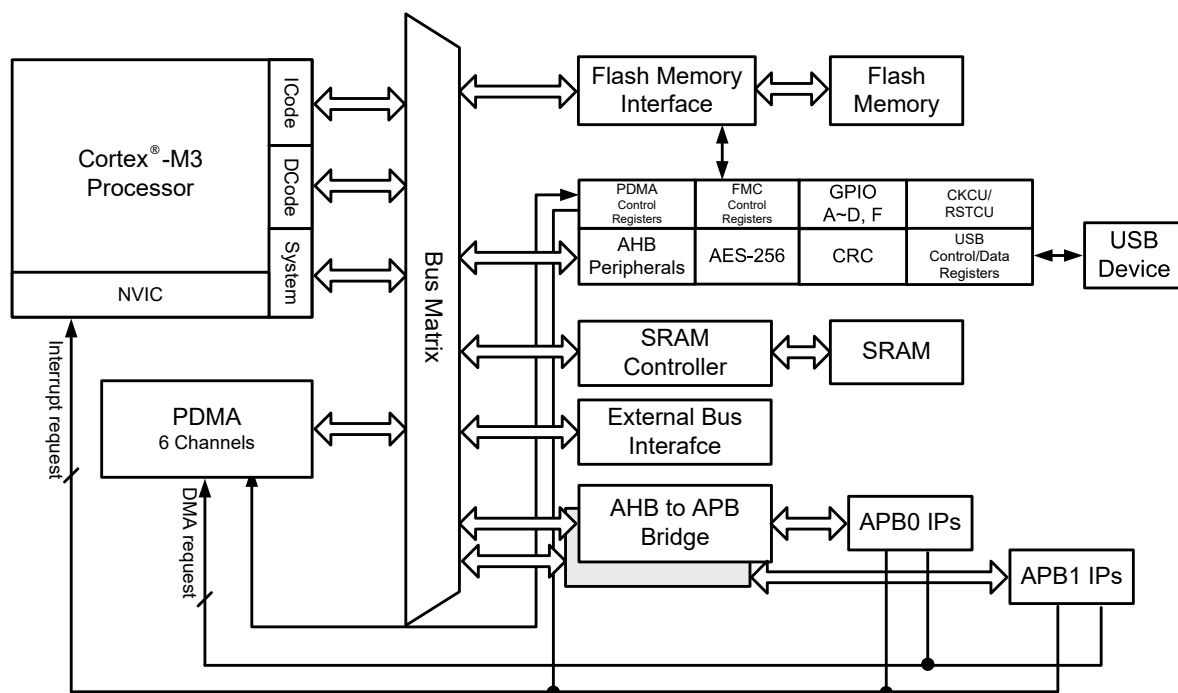


图 3. 总线结构

存储器体系

Arm® Cortex®-M3 处理器的结构是哈佛结构, 可以使用不同的总线进行取指令和加载 / 存储数据。指令代码和数据都位于相同的存储地址空间, 但在不同的地址范围。Cortex®-M3 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®-M3 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®-M3 系统外设使用。更多信息请参考 Arm® Cortex®-M3 技术参考手册。下图显示了该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

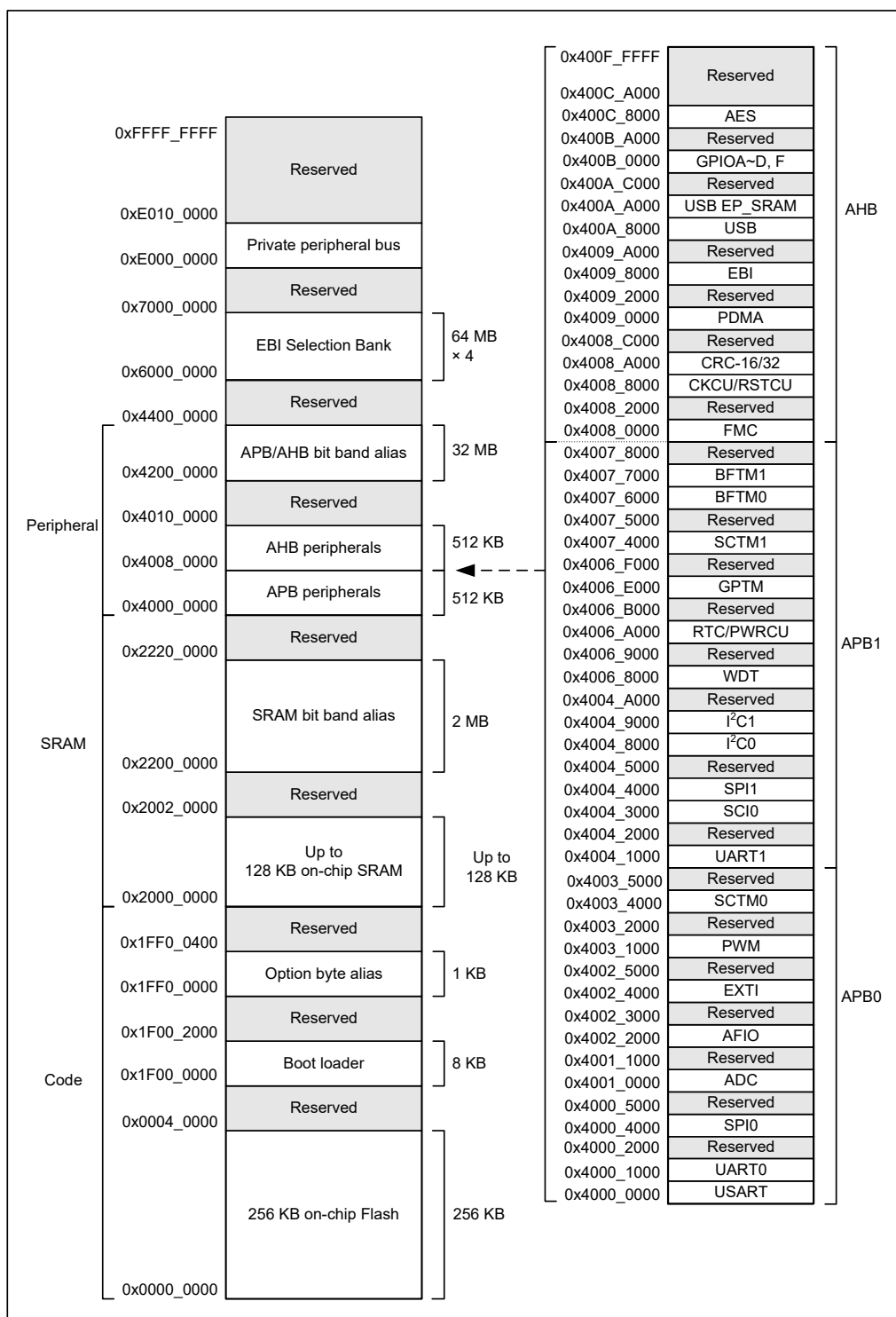


图 4. 存储器映射

表 3. 寄存器列表

起始地址	结束地址	外设	总线
0x4000_0000	0x4000_0FFF	USART	APB0
0x4000_1000	0x4000_1FFF	UART0	
0x4000_2000	0x4000_3FFF	保留	
0x4000_4000	0x4000_4FFF	SPI0	
0x4000_5000	0x4000_FFFF	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	PWM	
0x4003_2000	0x4003_3FFF	保留	
0x4003_4000	0x4003_4FFF	SCTM0	
0x4003_5000	0x4003_FFFF	保留	
0x4004_0000	0x4004_0FFF	保留	APB1
0x4004_1000	0x4004_1FFF	UART1	
0x4004_2000	0x4004_2FFF	保留	
0x4004_3000	0x4004_3FFF	SCI	
0x4004_4000	0x4004_4FFF	SPI1	
0x4004_5000	0x4004_7FFF	保留	
0x4004_8000	0x4004_8FFF	I ² C0	
0x4004_9000	0x4004_9FFF	I ² C1	
0x4004_A000	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_3FFF	保留	
0x4007_4000	0x4007_4FFF	SCTM1	
0x4007_5000	0x4007_5FFF	保留	
0x4007_6000	0x4007_6FFF	BFTM0	
0x4007_7000	0x4007_7FFF	BFTM1	
0x4007_8000	0x4007_FFFF	保留	

起始地址	结束地址	外设	总线
0x4008_0000	0x4008_1FFF	FMC	AHB
0x4008_2000	0x4008_7FFF	保留	
0x4008_8000	0x4008_9FFF	CKCU/RSTCU	
0x4008_A000	0x4008_BFFF	CRC-16/32	
0x4008_C000	0x4008_FFFF	保留	
0x4009_0000	0x4009_1FFF	PDMA	
0x4009_2000	0x4009_7FFF	保留	
0x4009_8000	0x4009_9FFF	EBI	
0x4009_A000	0x400A_7FFF	保留	
0x400A_8000	0x400A_9FFF	USB	
0x400A_A000	0x400A_BFFF	USB EP_SRAM	
0x400A_C000	0x400A_FFFF	保留	
0x400B_0000	0x400B_1FFF	GPIOA	
0x400B_2000	0x400B_3FFF	GPIOB	
0x400B_4000	0x400B_5FFF	GPIOC	
0x400B_6000	0x400B_7FFF	GPIOD	
0x400B_8000	0x400C_9FFF	保留	
0x400B_A000	0x400B_BFFF	GPIOF	
0x400B_A000	0x400C_7FFF	保留	
0x400C_8000	0x400C_9FFF	AES	
0x400C_A000	0x400F_FFFF	保留	

嵌入式 Flash 存储器

该单片机提供高达 256 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，对于 Cortex®-M3 ICode 或 DCode 总线访问，Flash 存储器仅支持读操作。任何对 Flash 存储器的写操作（通过 DCode 总线）将导致总线故障发生。该 Flash 存储器容量高达 256 页，每一页都有 1 KB 的存储容量，并且可以单独擦除。32-bit 可编程接口拥有使位从 1 到 0 改变的能力。数据存储或固件升级可以通过使用在线系统编程 (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

该单片机包含 128 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。为了减少读 - 修改 - 写操作的时间，Cortex®-M3 处理器提供一个位带功能来执行单个原子位操作。用户可以通过访问相应的位带别名来修改在 SRAM 位带区的单个位。有关位带的更多信息，请参考 Arm® Cortex®-M3 技术参考手册。下面的公式和例子说明了如何通过计算位带别名来访问位带区中的一个位。

$$\text{位带别名} = \text{位带基址} + (\text{字节偏移量} \times 32) + (\text{位序号} \times 4)$$

例如，要访问地址 0x2000_0200 的第 7 位，位带别名为：

$$\text{位带别名} = 0x2200_0000 + (0x200 \times 32) + (7 \times 4) = 0x2200_401C$$

写数据到地址 0x2200_401C 将导致地址 0x2000_0200 的第 7 位改变，而读取地址 0x2200_401C 的数据，根据 SRAM 地址 0x2000_0200 第 7 位的值将返回 0x01 或 0x00。

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，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。APB-AHB 总线桥在 CPU 和 APB 外设之间有进行访问的能力。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增加到一个字的宽度。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的外设寄存器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, 预取缓存器和分支缓存。下图显示了 FMC 的方框图, 包括烧录接口、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 的等待时间, 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。

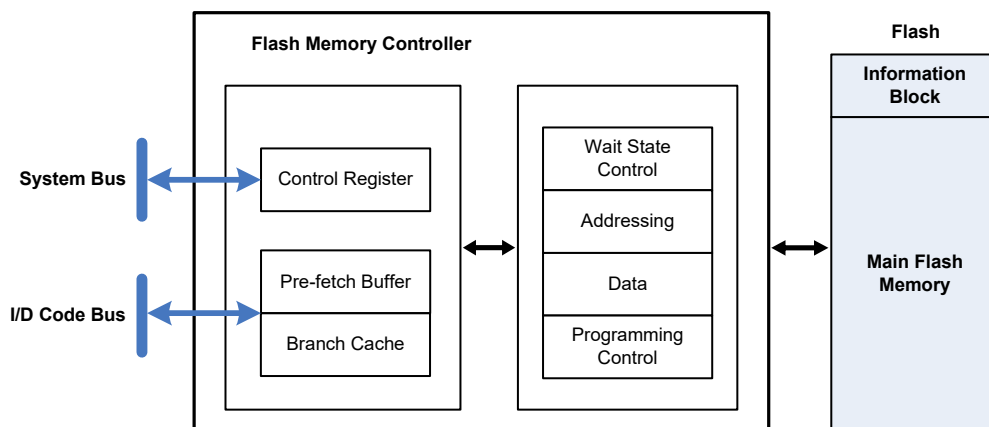


图 5. Flash 存储器控制器方框图

特性

- 多达 256 KB 片上 Flash 存储器用于存储指令 / 数据和选项字节
 - 256 KB: 255 KB + 1 KB (指令 / 数据 + 选项字节)
- 页大小为 1 KB, 总计多达 256 页
- 带有预取缓存器和分支缓存访问接口, 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 存储器运行结束或错误发生
- 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

下图是系统地址从 0x0000_0000 到 0x1FFF_FFFF (0.5 GB) 的 Flash 存储器映射。0x1F00_0000 ~ 0x1F00_1FFF 的地址被映射到启动加载块 (8 KB)。此外, 从 0x1FF0_0000 到 0x1FF0_03FF 的地址区域是选项字节块 (1 KB) 在系统视图下的存储器映射如下图所示。

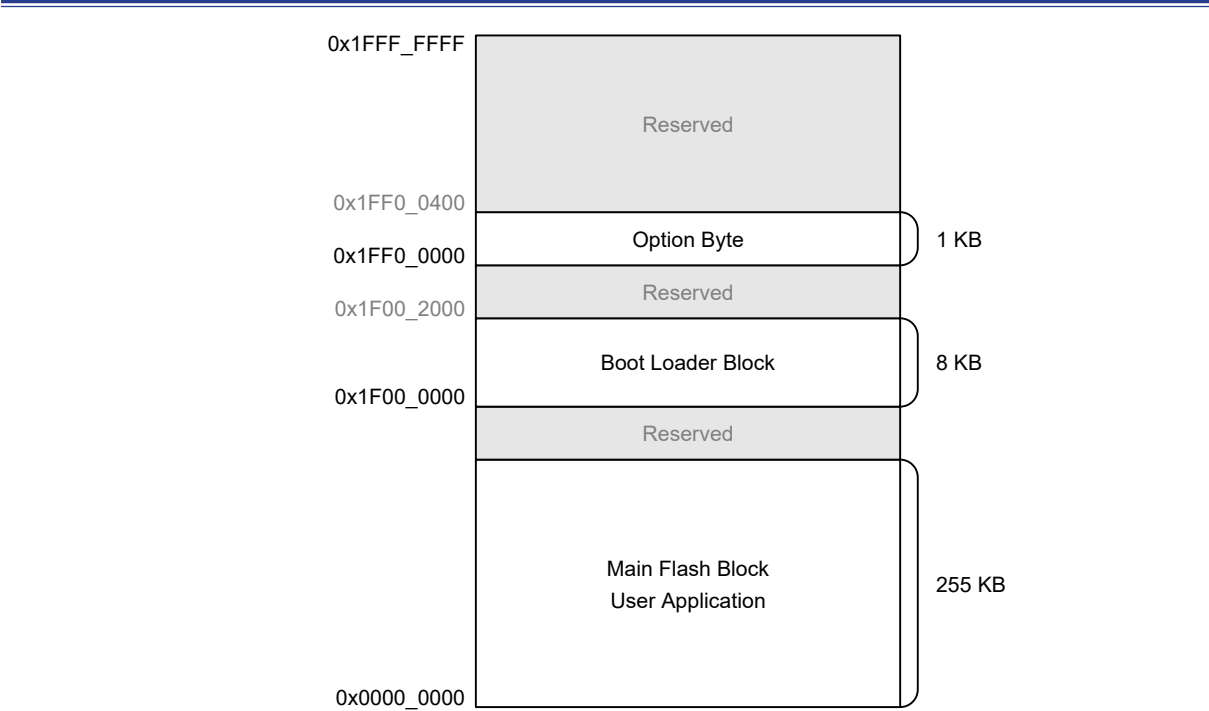


图 6. Flash 存储器映射

Flash 存储器结构

Flash 存储器由高达 256 KB 主 Flash 存储块 (每页 1 KB) 和用于启动加载器的 8 KB 信息块组成。主 Flash 存储块总共包含 256 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		1 KB
	Page 2	0x0000_0800 ~ 0x0000_0BFF	OB_PP [1]	1 KB
	Page 3	0x0000_0C00 ~ 0x0000_0FFF		1 KB
	⋮	⋮	⋮	⋮
	Page 252	0x0003_F000 ~ 0x0003_F3FF	OB_PP [126]	1 KB
	Page 253	0x0003_F400 ~ 0x0003_F7FF		1 KB
	Page 254	0x0003_F800 ~ 0x0003_FBFF	OB_PP [127]	1 KB
	Page 255 (选项字节)	物理地址: 0x0003_FC00 ~ 0x0003_FFFF 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_1FFF	NA	8 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。

等待状态设置

当 CPU 时钟 HCLK 比 Flash 存储器的访问速度快时, 必须在 CPU 取指令或从 Flash 存储器加载数据时插入等待状态周期。通过设置 Flash 缓存和预取控制寄存器 CFCR 中的 WAIT[2:0], 可以改变等待状态。为了满足等待状态的需要, 以下两条规则应该予以考虑。

- 从低频到高频切换 HCLK 时钟:
先更改等待状态设置, 然后再切换 HCLK 时钟。
- 从高频到低频切换 HCLK 时钟:
先切换 HCLK 时钟, 然后再更改等待状态设置。

下表显示了等待状态周期和 CPU 时钟 HCLK 之间的关系。默认的等待状态为 0, 因为系统复位后 HSI (8 MHz) 被选择作为 HCLK 的时钟源。

表 5. 等待状态周期与 HCLK 之间的关系

等待状态周期	HCLK
0	0 MHz < HCLK ≤ 20 MHz
1	20 MHz < HCLK ≤ 40 MHz
2	40 MHz < HCLK ≤ 60 MHz
3	60 MHz < HCLK ≤ 72 MHz

启动配置

系统提供了三种启动模式，通过 BOOT0~BOOT1 引脚进行选择。在上电复位或系统复位期间对 BOOT0 ~ BOOT1 引脚状态采样。一旦这些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启动区域。下表列出了启动模式。

表 6. 启动模式

启动模式选择引脚		模式	描述
BOOT1	BOOT0		
0	0	SRAM	SBVT0 ~ SBVT3 被选为启动区域
0	1	启动加载器	启动加载器被选为启动区域
1	X	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 用来在芯片复位后临时改变向量重映射设置。VMCR 的初始复位值是由将在复位时被采样的 BOOT0 ~ BOOT1 引脚定义的。

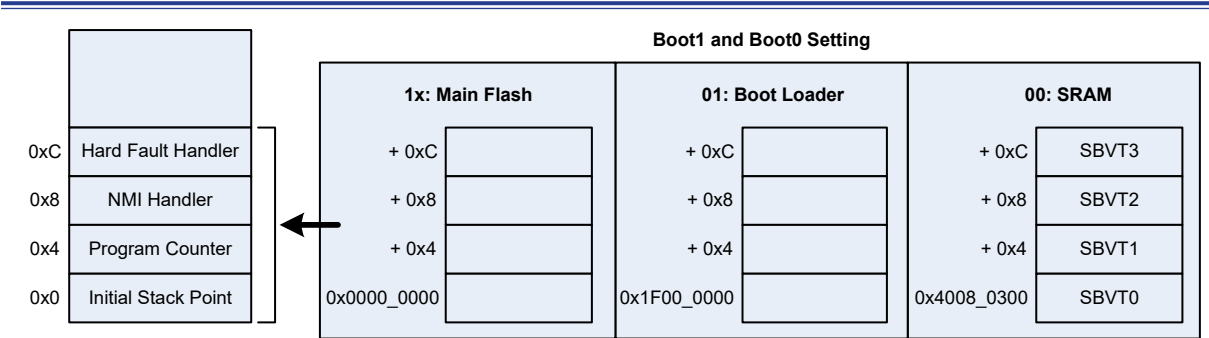


图 7. 向量重映射

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操作的寄存器的访问顺序。

- 检查 OPCR 寄存器，以确认没有 Flash 存储器的操作正在进行 (OPM [3:0] = 0xE 或 0x6)。否则需一直等待直到前一操作已经完成为止。
- 写入页地址到 TADR 寄存器。
- 写入页擦除命令到 OCMR 寄存器 (CMD [3:0] = 0x8)。
- 通过设置 OPCR 寄存器，发送页擦除命令到 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，通过访问 DCODE 来读取并验证 Flash 存储器的内容。

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。发生该情况时，OREF 位将被置位。如果 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。

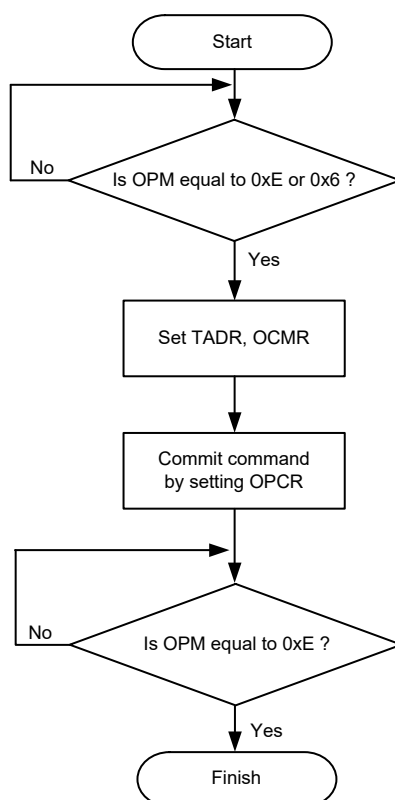


图 8. 页擦除操作流程

整片擦除

FMC 提供了一个整片擦除功能，用来初始化整个 Flash 存储器的内容为高状态。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 Flash 存储器的操作正在进行 (OPM [3:0] = 0xE 或 0x6)。否则需一直等待直到 Flash 存储器操作已经完成为止。
- 写入整片擦除命令到 OCMR 寄存器 (CMD [3:0] = 0xA)。
- 通过设置 OPCR 寄存器，发送整片擦除命令到 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，通过访问 DCODE 来读取和验证 Flash 存储器的内容。

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

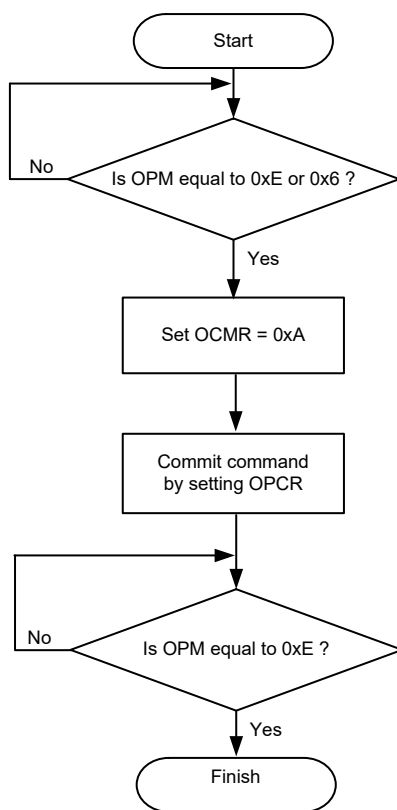


图 9. 整片擦除操作流程

字编程

FMC 提供了 32-bit 字编程功能，用于修改 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 Flash 存储器的操作正在进行 (OPM [3:0] = 0xE 或 0x6)。否则需一直等待直到前面的操作已经完成为止。
- 写入字地址到 TADR 寄存器。写入数据到 WRDR 寄存器。
- 写入字编程命令到 OCMR 寄存器 (CMD [3:0] = 0x4)。
- 通过设置 OPCR 寄存器，发送字编程命令到 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，通过访问 DCODE 来读取和验证 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。发生该情况时，OREF 位将被置位。如果 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。

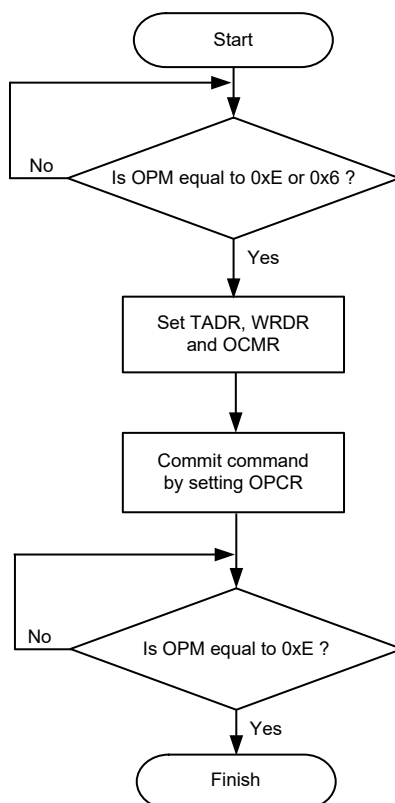


图 10. 字编程操作流程

选项字节描述

选项字节可视为一个基址为 0x1FF0_0000 的独立的 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 7. 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 0x004 0x008 0x00C	OB_PP [n]: 主 Flash 页擦除 / 编程保护 (n = 0 ~ 127) 0: Flash 页 2n 和 2n+1 擦除 / 编程保护使能 1: Flash 页 2n 和 2n+1 擦除 / 编程保护除能	0xFFFF_FFFF 0xFFFF_FFFF 0xFFFF_FFFF 0xFFFF_FFFF
OB_CP	0x010	OB_CP [0]: Flash 安全保护 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 0: 选项字节保护使能 1: 选项字节保护除能 OB_CP [31:2]: 保留位	0xFFFF_FFFF
OB_CK	0x020	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 选项字节 5 个寄存器内容的总和, 其地址偏移量范围在选项字节中从 0x000 到 0x010 (0x000 + 0x004 + 0x008 + 0x00C + 0x010)。否则, 页擦除 / 程序保护和安全保护将被使能。	0xFFFF_FFFF
OB_WDT	0x02C	Flash 选项 WDT 功能使能 OB_WDT[15:0]: 0x7A92 若设置 OB_WDT[15:0] 的值为 0x7A92, 当发生 MCU 上电复位或系统复位, WDT 功能会立即使能。可通过软件除能 WDT 功能。 OB_WDT[31:16]: 保留	0xFFFF_FFFF
OB_TOOL	0x030 ~ 0x04C	保留, 用于 Flash 烧录工具和引导装载程	0xFFFF_FFFF

页擦除 / 编程保护

FMC 提供了页擦除 / 编程保护, 以防止 Flash 存储器上的误操作。FMC 受保护页不接受页擦除命令 (OCMR 寄存器中的 CMD [3:0] = 0x8) 或字编程命令 (CMD [3:0] = 0x4), 当向 FMC 发送对受保护页的页擦除或字编程命令时, 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 [127:0] 位域, 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 8. 主 Flash 受保护页的访问权限

操作 \ 模式	ISP / IAP	ICP / 调试模式	从 SRAM 启动
DCODE 读操作	O	O	O
编程	X	X	X
页擦除	X	X	X
整片擦除	O	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] 配置。选项字节物理上位于主 Flash 的最后一页。选项字节页保护由 OB_CP[1] 位设置。
3. 选项字节区域中的页擦除可除能主 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 Flash 存储器的操作正在进行 (OPM [3:0] = 0xE 或 0x6)。否则，等待直到前面的操作已经完成。
- 将 OB_PP 地址写入 TADR 寄存器 (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)。
- 将字编程命令写到 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，通过访问 DCODE 来读取和验证操作字节。
- 在激活新的 OB_PP 设置之前，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。该功能对保护软件 / 固件不受非法使用者侵害非常有用。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 DCODE 访问，编程和页擦除的数据访问将不会被允许。但是，为了除能安全保护功能，整片擦除操作仍将被 FMC 接受。下表显示了安全保护使能时 Flash 存储器的访问权限。

表 9. 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	从 SRAM 启动
DCODE 读操作	O	X (读为 0)	X (读为 0)
编程	O ⁽¹⁾	X	X
页擦除	O ⁽¹⁾	X	X
整片擦除	O	O	O

注：1. 用户应用程序是指在 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，不能编程 / 页擦除。
2. 整片擦除操作可以擦除选项字节块和除能安全保护功能。

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 Flash 存储器的操作正在进行 (OPM [3:0] = 0xE 或 0x6)。否则，需一直等待到前面的操作已经完成。
- 将 OB_CP 地址写入到 TADR 寄存器 (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，通过访问 DCODE 来读取和验证操作字节。
- 在激活安全保护功能之前，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安全保护设置。

寄存器列表

下表显示了 FMC 寄存器及其复位值。

表 10. 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	0xFFFF_FFFF 0xFFFF_FFFF 0xFFFF_FFFF 0xFFFF_FFFF
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 制造商和设备 ID 寄存器	0x0376_XXXX
PNSR	0x184	Flash 页数量状态寄存器	0x0000_00FF
PSSR	0x188	Flash 页大小状态	0x0000_0400
DID	0x18C	设备 ID 寄存器	0x000X_XXXX
CFCR	0x200	Flash 缓存和预取控制寄存器	0x0000_1091
SBVT0	0x300	SRAM 启动向量 0 (堆栈指针)	0x200X_0000
SBVT1	0x304	SRAM 启动向量 1 (程序计数器)	0x2000_0159
SBVT2	0x308	SRAM 启动向量 2 (NMI 处理器)	0x0000_0000
SBVT3	0x30C	SRAM 启动向量 3 (硬故障处理器)	0x0000_0000
CIDR0	0x310	自定义 ID 寄存器 0	0xFFFF_FFFF
CIDR1	0x314	自定义 ID 寄存器 1	0xFFFF_FFFF
CIDR2	0x318	自定义 ID 寄存器 2	0xFFFF_FFFF
CIDR3	0x31C	自定义 ID 寄存器 3	0xFFFF_FFFF

注：“X”表示未知的复位值，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	域	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 256 KB，TADB [31:18] 应设置为 0。从地址 0x1FF0_0000 到 0x1FF0_03FF 为 1 KB 选项字节。对于这个区域的可用 Flash 地址，必须在 0x0000_0000 ~ 0x1FFF_FFFF。否则，若相应中断使能，将产生一个无效目标地址中断。

Flash 写数据寄存器 – WRDR

该寄存器用于存储编程操作中将写入 TADR 寄存器的数据。

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

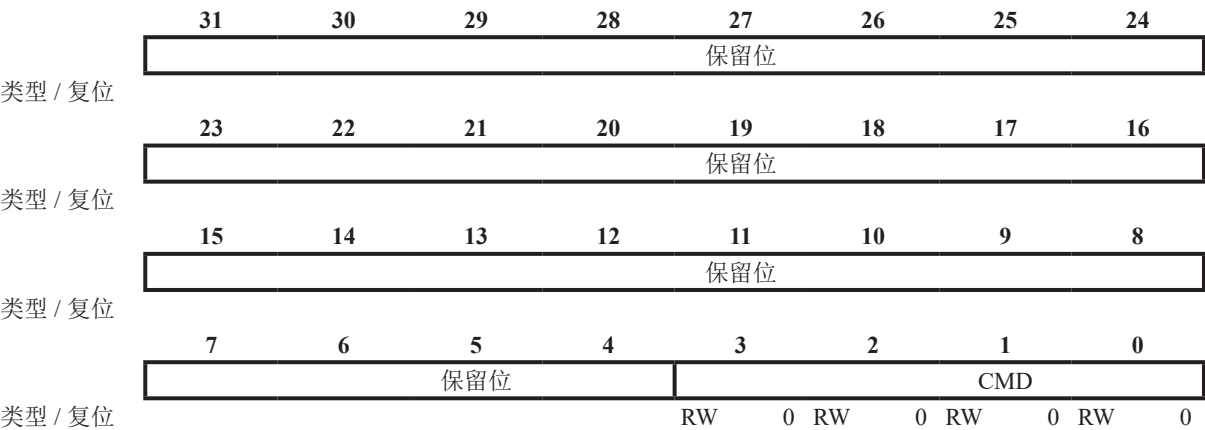
位	域	描述
[31:0]	WRDB	Flash 写数据位 用于编程操作的数据值。

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

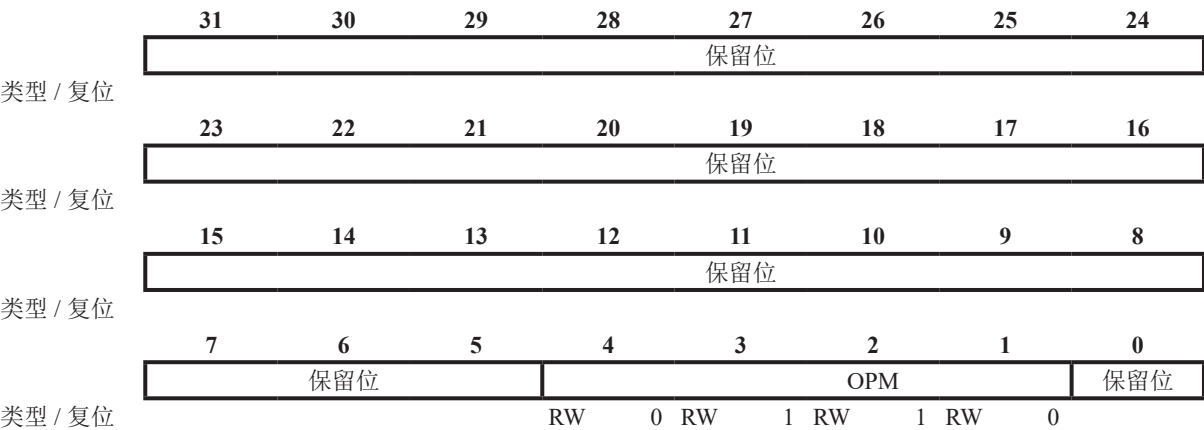


位	域	描述												
[3:0]	CMD	Flash 操作命令位 下表显示了操作命令位 CMD [3:0] 的定义，它决定了 Flash 存储器的操作。如果设置了一个无效的命令，且 IOCMIEN 位等于 1，将产生一个无效操作命令中断。												
<table><tr><th>CMD [3:0]</th><th>描述</th></tr><tr><td>0x0</td><td>空闲 (默认)</td></tr><tr><td>0x4</td><td>字编程</td></tr><tr><td>0x8</td><td>页擦除</td></tr><tr><td>0xA</td><td>整片擦除</td></tr><tr><td>其它值</td><td>保留</td></tr></table>			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
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

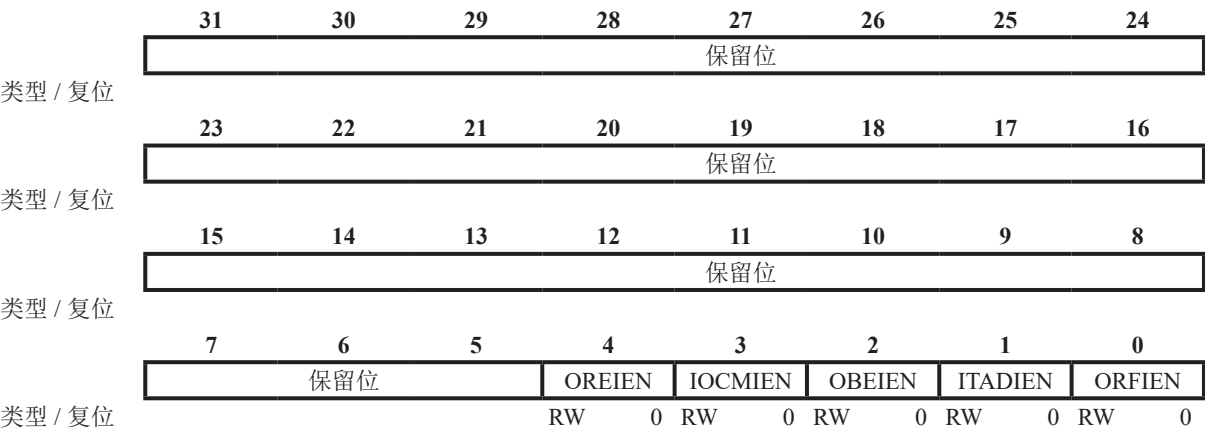


位	域	描述										
[4:1]	OPM	<p>操作模式位</p> <p>下表列出了 FMC 的操作模式。根据 TADR 寄存器的地址别名设置，用户可以提交由 OCMR 寄存器对 FMC 设置的命令。TADR、WRDR 和 OCMR 寄存器的内容应该在设置该寄存器之前准备好。在所有操作都完成后，OPM 位域将由 FMC 硬件设置为 0xE。当所有操作已完成，设置进入空闲模式以降低功耗。注意，执行下一步动作之前，应检查操作状态。TADR、WRDR、OCMR 和 OPCR 寄存器的内容不应该被改变直到前面的操作完成。</p> <table><tr><th>OPM [3:0]</th><th>描述</th></tr><tr><td>0x6</td><td>空闲 (默认)</td></tr><tr><td>0xA</td><td>提交命令给主 Flash</td></tr><tr><td>0xE</td><td>所有操作在主 Flash 中完成</td></tr><tr><td>其它值</td><td>保留</td></tr></table>	OPM [3:0]	描述	0x6	空闲 (默认)	0xA	提交命令给主 Flash	0xE	所有操作在主 Flash 中完成	其它值	保留
OPM [3:0]	描述											
0x6	空闲 (默认)											
0xA	提交命令给主 Flash											
0xE	所有操作在主 Flash 中完成											
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且中断事件发生时 FMC 会产生中断。

偏移量： 0x014
复位值： 0x0000_0000

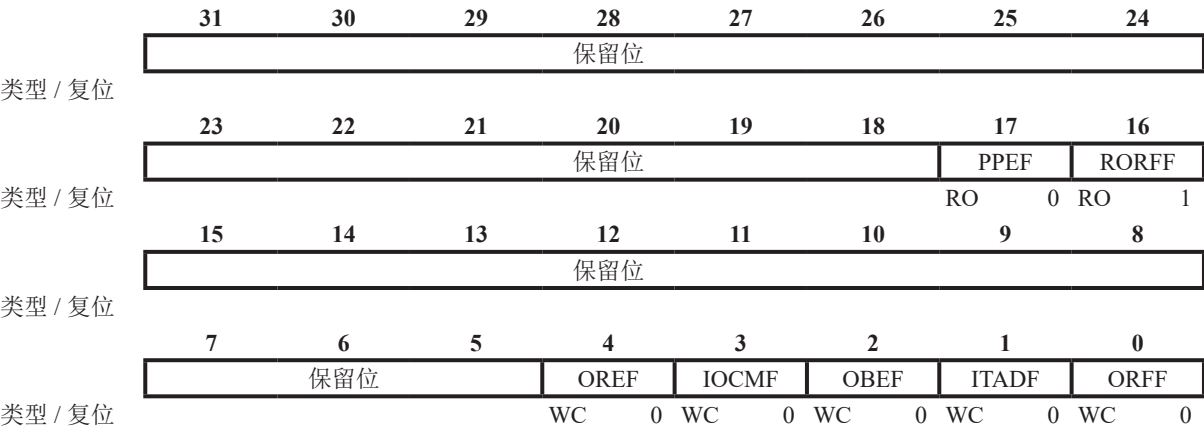


位	域	描述
[4]	OREIEN	操作错误中断使能位 0: 操作错误中断除能 1: 操作错误中断使能
[3]	IOCMIEN	无效操作命令中断使能位 0: 无效操作命令中断除能 1: 无效操作命令中断使能
[2]	OBEIEN	选项字节校验和错误中断使能位 0: 选项字节校验和错误中断除能 1: 选项字节校验和错误中断使能
[1]	ITADIEN	无效目标地址中断使能位 0: 无效目标地址中断除能 1: 无效目标地址中断使能
[0]	ORFIEN	操作完成中断使能位 0: 操作完成中断除能 1: 操作完成中断使能

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 操作是否完成或者是否有错误发生。当 OIER 寄存器中相应的位被置位时，该状态位 (位 [4:0]) 置位后才能触发中断。

偏移量: 0x018
复位值: 0x0001_0000



位	域	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，由于无效的页擦除 / 编程操作被应用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 操作命令写入到 OCMR 寄存器 1: 无效的 Flash 操作命令被写入到 OCMR 寄存器 当在 OCMR 寄存器写入无效 Flash 操作命令时，该位被置位。如果 OIER 寄存器中的 IOCMIEN 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 当选项字节校验和错误时，此位将被置高。如果 OIER 寄存器中的 OBEIEN 位被置位，OBE 中断将发生。可通过写 1 来复位该位。但直到中断条件清除后 OBEF 才可通过软件写 1 来复位，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	域	描述
[1]	ITADF	无效目标地址标志位 0: 目标地址 TADR 有效 1: 目标地址 TADR 无效 TADR 位域中的数据必须在 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, 该位被置位。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 存储器页擦除 / 程序保护的状态。

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

	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
	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
	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
	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

位	域	描述
[127:0]	PPSBn	第 n 页擦除 / 编程保护状态位 (n = 0 ~ 127) PPSB[n] = OB_PP[n] 0: 相应的页受保护 1: 相应的页不受保护 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时, 特定页的擦除或编程功能不被允许。PPSR [127:0] 的复位值由选项字节 OB_PP [127:0] 决定。因该单片机主 Flash 最大页数量为 256 KB, 因此每个页擦除 / 编程保护状态位可对应 2 页。OB_PP 和 PPSR 寄存器的未使用到的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，只能在任何一种复位发生后通过选项字节加载器重新加载。

偏移量： 0x030
复位值： 0x0000_000X

	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						OBPSB	CPSB
类型 / 复位							RO X	RO X

位	域	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设置引脚 BOOT0 ~ BOOT1 决定。

偏移量: 0x100
复位值: 0x0000_000X

	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						VMCB	
							RW	X
							RW	X

位	域	描述																				
[1:0]	VMCB	向量映射控制位 VMCB 位用来控制地址从 0x0 到 0xC 的前 4 个字向量的映射源。下表显示了向量映射的设置。																				
<table><tr><th>BOOT1</th><th>BOOT0</th><th>VMCB [1:0]</th><th>描述</th></tr><tr><td>低</td><td>低</td><td>00</td><td>SRAM 启动模式 向量映射源是 SBVT0 ~ 3。</td></tr><tr><td>低</td><td>高</td><td>01</td><td>启动加载器模式 向量映射源是启动加载器区域</td></tr><tr><td>高</td><td>低</td><td>10</td><td>主 Flash 模式</td></tr><tr><td>高</td><td>高</td><td>11</td><td>模式向量映射源是主 Flash 存储器区域</td></tr></table>			BOOT1	BOOT0	VMCB [1:0]	描述	低	低	00	SRAM 启动模式 向量映射源是 SBVT0 ~ 3。	低	高	01	启动加载器模式 向量映射源是启动加载器区域	高	低	10	主 Flash 模式	高	高	11	模式向量映射源是主 Flash 存储器区域
BOOT1	BOOT0	VMCB [1:0]	描述																			
低	低	00	SRAM 启动模式 向量映射源是 SBVT0 ~ 3。																			
低	高	01	启动加载器模式 向量映射源是启动加载器区域																			
高	低	10	主 Flash 模式																			
高	高	11	模式向量映射源是主 Flash 存储器区域																			
VMCR 寄存器的复位值是由上电复位和系统复位期间外部启动引脚 BOOT1 和 BOOT0 的状态决定。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	1
	23	22	21	20	19	18	17	16
	MFID							
类型 / 复位	RO	0	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
	7	6	5	4	3	2	1	0
	ChipID							
类型 / 复位	RO	X	RO	X	RO	X	RO	X

位	域	描述
[31:16]	MFID	制造商 ID 读为 0x0376。
[15:0]	ChipID	芯片 ID 读取 MCU 型号最后四位数字值。

Flash 页数量状态寄存器 – PNSR

该寄存器定义了 Flash 存储器的页数。

偏移量: 0x184
复位值: 0x0000_00FF

	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	1	RO	1	RO	1	RO	1	RO

位	域	描述
[31:0]	PNSB	Flash 页数状态位 0x0000_0020: 片上 Flash 存储器总页数为 32 0x0000_0040: 片上 Flash 存储器总页数为 64 0x0000_0080: 片上 Flash 存储器总页数为 128 0x0000_00FF: 片上 Flash 存储器总页数为 255

Flash 页大小状态寄存器 – PSSR

该寄存器用于显示每页包含多少字节。

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	域	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 – DID

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

	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				ChipID				
类型 / 复位					RO	X RO	X RO	X RO	X
	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 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 RO	X RO	X

位	域	描述
[19:0]	ChipID	芯片 ID MCU 设备型号。

Flash 缓存和预取控制寄存器 – CFCR

该寄存器用于控制 FMC 缓存和预取模块。

偏移量：0x200

复位值：0x0000_1091

	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			CE	保留位			
	RW 1							
	7	6	5	4	3	2	1	0
类型 / 复位	DCDB	保留位		PFBE	保留位	WAIT		
	RW 1			RW 1	RW 0		RW 0	RW 1

位	域	描述
[12]	CE	分支缓存使能位 0: 分支缓存除能 1: 分支缓存使能
[7]	DCDB	DCODE 数据缓存使能位 0: DCODE 数据缓存使能 1: DCODE 数据缓存除能
[4]	PFBE	预取缓冲器使能位 0: 预取缓冲器除能 1: 预取缓冲器使能 (默认) 在默认状态下, 预取缓冲器使能。当预取缓冲器除能, 指令和数据直接由 Flash 存储器提供。
[2:0]	WAIT	Flash 等待周期设置位 WAIT [2:0] 用于设置在非连续 Flash 访问期间 HCLK 等待时钟。等待时钟的实际值是 (WAIT [2:0]-1)。由于提供了一个带预取缓存器和分支缓存的宽访问接口, 连续 Flash 访问的等待状态非常接近于零。

WAIT [2:0]	等待状态	HCLK 允许的范围
001	0	0 MHz < HCLK ≤ 20 MHz
010	1	20 MHz < HCLK ≤ 40 MHz
011	2	40 MHz < HCLK ≤ 60 MHz
100	3	60 MHz < HCLK ≤ 72 MHz
其他值	保留	保留

SRAM 启动向量寄存器 n – SBVTn, n = 0 ~ 3

这些寄存器定义了用于 SRAM 启动模式的堆栈指针、程序计数器、NMI 处理器地址和硬故障处理器地址。

偏移量： 0x300 (0) ~ 0x30C (3)

复位值： 取决于地址偏移量的变化

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

位	域	描述
---	---	----

[31:0]	SBVTn	SRAM 启动向量 n (n = 0 ~ 3) SRAM 启动向量 0 ~ 3 提供了一个用于应用调试的 SRAM 启动能力。在 SRAM 启动模式下，SBVTn 寄存器的内容被重映射到地址 0x0 到 0xC 的 Flash 存储器的代码区。请参考 VMCR 寄存器和 BOOT1 / BOOT0 启动引脚的描述。下表列出了 SBVTn 寄存器的用途和复位值。复位值提供了在 SRAM 启动模式下程序执行的固定设置。这些寄存器可以由调试工具修改，以更改程序执行的设置。SBVTn 寄存器的复位值只能由上电复位重新加载，其它复位源将没有任何效果。
--------	-------	--

名称	地址偏移量	用途描述	复位值
SBVT0	0x300	堆栈指针	128 KB SRAM: 0x2002_0000 64 KB SRAM: 0x2001_0000
SBVT1	0x304	程序计数器	0x2000_0159
SBVT2	0x308	NMI 处理器地址	0x0000_0000
SBVT3	0x30C	硬故障处理器地址	0x0000_0000

SBVT0 ~ SBVT3 寄存器的访问宽度必须是 32-bit 访问 (字访问)。8 或 16 位访问 (字节或半字访问) 是不允许的。

自定义 ID 寄存器 n – CIDRn, n = 0 ~ 3

该寄存器定义了自定义 ID 信息，用来作为自定义标识。

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	域	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) 位域。

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，在该单片机中，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、深度休眠模式 2 和暂停模式。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。下图中的虚线指出了两个数字电源域的供电来源。

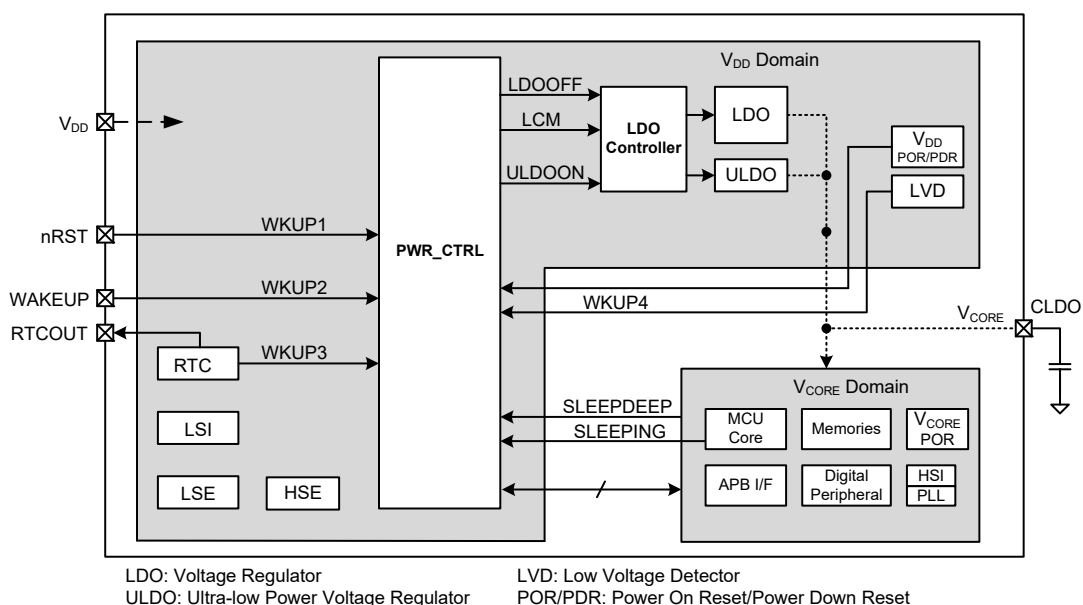


图 11. PWRCU 方框图

特性

- 两个电源域: V_{DD} 和 V_{CORE} 电源域
- 四种省电模式: 休眠模式、深度休眠模式 1、深度休眠模式 2 和暂停模式
- 内部稳压器提供 V_{CORE} 电压源
- 额外提供 V_{CORE} 低静态电流, 低工作电流的超低功耗稳压器 (ULDO)
- 下列条件之一成立时, 将发生电源复位
 - 上电 / 掉电复位 (POR / PDR 复位)
 - 当退出暂停模式
 - 控制位 $BODEN = 1$, $BODRIS = 0$ 且电源电压 $V_{DD} \leq V_{BOD}$
- 当 V_{DD} 低于欠压检测器电压 V_{BOD} 时, 欠压检测器可以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} 时, 低电压检测器可以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

当有下列情况之一发生, 主 LDO (MLDO) 和超低功耗 LDO (ULDO) 将自动关闭:

- 进入暂停模式
- 电源电压 $V_{DD} \leq V_{PDR}$

当 $V_{DD} > V_{POR}$ 时, 如果下列任一情况发生, 主 LDO 会自动由硬件开启:

- 从省电模式恢复运作 – RTC 唤醒, LVD 唤醒和 WAKEUP 引脚的上升沿
- 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 或深度休眠模式 2, PWRCU 将关闭主 LDO, 同时使 ULDO 工作在高待机电流模式提供另外一个 V_{CORE} 替代电源。

稳压器

主稳压器 LDO、超低功耗 LDO (ULDO)、低电压检测器 LVD、内部低速振荡器 LSI、外部高速振荡器 HSE 和外部低速晶体振荡器 LSE 都在 V_{DD} 电源域下工作。主 LDO 可配置为工作在正常模式 ($LDOOFF = 0$, $LDOLCM = 0$, I_{OUT} = 大电流模式), 超低功耗 ULDO 可配置为工作在小电流模式 ($LDOOFF = 0$, $LDOLCM = 1$, I_{OUT} = 小电流模式), 以提供 V_{CORE} 电源。在深度休眠模式 2, 使用具有超低静态电流的 ULDO, 提供电源用于 V_{CORE} 电源域数据保存。ULDO 由 PWRCR 寄存器中的 ULDOON 位控制。

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 1.65 V 开始升压或者降压至 1.65 V 的过程中都可以适当工作。当单片机进入暂停模式时, 若是发生 V_{DD} 低于特定的阈值电压 V_{PDR} 时, 单片机将维持在暂停模式, 无需外部复位电路。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。

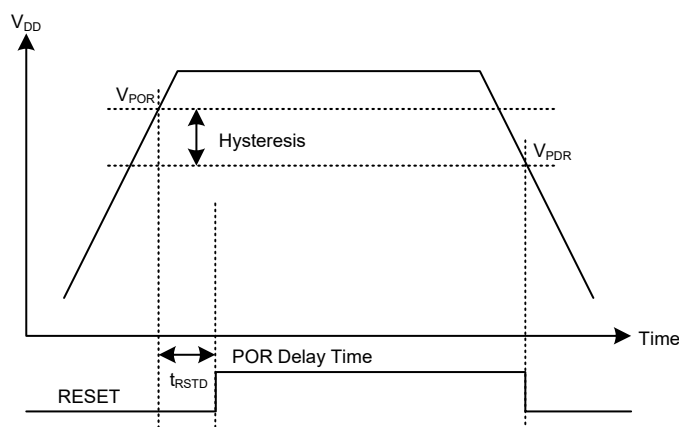


图 12. 上电复位 / 掉电复位波形

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 位域选择。当在 V_{DD} 电源引脚上检测到一个低电压, LVDF 标志将置位, 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 执行进入暂停模式的程序 (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或 PLL 时钟源。

LSE、LSI 和 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。

V_{CORE} 电源域

主要功能包括用于 V_{DD} 电源域的 APB 接口、MCU 内核逻辑、AHB / APB 外设和存储器等，都位于此电源域。一旦 V_{CORE} 电源上电，POR 将产生一个在 V_{CORE} 电源域的复位序列。随后，要进入相应的省电模式，相关的控制位包括 LDOOFF、ULDOON 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_{CORE} 电源域。当从深度休眠模式退出时，通过设置 PSRCEN 位 (位于时钟控制单元 CKCU 中的全局时钟控制寄存器 GCCR) 为 1，HSI 时钟可以在一段时间内被用作系统时钟。当进入深度休眠模式之前使用的初始时钟源稳定后，系统时钟切换回这个可能来自 PLL 或 HSE 的初始时钟源。此外，系统从暂停模式唤醒后，会发生 V_{CORE} 上电复位，此时会迫使 HSI 振荡器作为系统时钟。

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 AHBPRES 位域以减慢系统时钟，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 寄存器减慢外围时钟以满足应用需求。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 11. 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 的时钟通过设置可以停止。
深度休眠 1 ~ 2	1. 停止 V _{CORE} 电源域的所有时钟。 2. 除能 HSI、HSE 和 PLL。 3. 通过打开超低功耗 ULDO 来减少 V _{CORE} 电源域的电流。
暂停	关闭 V _{CORE} 电源域。

休眠模式

默认情况下，系统进入休眠模式，只有 CPU 时钟会停止。清除 CKCU AHBCCR 寄存器中的 FMCEN 或 SRAMEN 位为 0，在系统进入休眠模式后将会停止 Flash 时钟或 SRAM 时钟。如果在休眠模式 CPU 没有必要访问 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 12. 进入 / 退出省电模式

模式	模式进入				模式退出
	CPU 指令	CPU SLEEPDEEP	LDOOFF	ULDOON	
休眠	WFI 或 WFE (生效)	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		1	0	0	在事件模式的任何 EXTI 或 RTC 唤醒或 LVD 唤醒 ⁽²⁾ 或 WAKEUP 引脚的上升沿或 USB 恢复
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
暂停		1	1	0	RTC 唤醒或 LVD 唤醒 ⁽²⁾ 或 WAKEUP 引脚的上升沿或 外部复位 (nRST)

注: 1. 唤醒事件是指 EXTI 线触发事件、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 位需设为 1, 以确保系统可以由一个 LVD 事件唤醒, 从深度休眠 2 和暂停模式中唤醒时, 主 LDO 稳压器可以开启。

深度休眠模式

要进入深度休眠模式, 配置如上表所示的寄存器, 并执行 WFI 或 WFE 指令。在深度休眠模式下, 所有时钟包括 PLL 和高速振荡器, 即 HSI 和 HSE 将被停止。此外, 在深度休眠 1 和深度休眠 2 模式, 会关闭主 LDO, 使用超低功耗 LDO (ULDO) 提供 V_{CORE} 电源。一旦 PWRUCU 接收到一个如上表模式退出一栏所述的唤醒事件或中断, 主 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 PWRUCU 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。

暂停模式

暂停模式衍生于 CPU 深度休眠模式外加控制位 LDOOFF 和 ULDOON。要进入暂停模式, 用户可以依前面的表 12 所示的模式对相关寄存器进行配置, 并执行 WFI 或 WFE 指令。RTC 唤醒触发事件、LVD 唤醒, 外部 WAKEUP 引脚从低到高电平的转换或外部复位引脚 (nRST) 信号将迫使 MCU 退出暂停模式。在暂停模式下, V_{CORE} 电源将被关闭。其余的有效电源是 V_{DD}/V_{DDA} 电源。

系统复位后, 应通过软件检测 RSTCU GRSR 寄存器中的 PORSTF 位以及 PWRSR 寄存器中的 PDF 和 PORF 位, 以确定单片机是否通过上电复位或其它复位 (nRST, WDT 等) 从暂停模式中恢复。如果单片机在正确操作下进入暂停模式, 那么 PDF 位将被置位。系统信息可以被保存在 V_{DD} 电源域寄存器中, 当 V_{CORE} 电源域再次上电时可以被重新取回。有关 PWRSR 寄存器中 PDF 位和 PORF 位以及 RSTCU GRSR 寄存器中的 PORSTF 位的更多信息如下表所示。

表 13. 系统复位后的电源状态

PORF	PDF	PORSTF	描述
1	0	1	V _{DD} 电源域复位后首次上电： 当 V _{DD} 首次使用或对 V _{DD} 域执行软件复位
0	0	1	从 V _{CORE} 电源意外掉电重新启动或其它复位 (nRST, WDT 等)
0	1	1	从暂停模式重新启动
1	1	X	保留

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位于 V_{DD} 电源域。

表 14. 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_0001
PWRCR	0x104	电源控制寄存器	0x0000_0000
PWRTEST	0x108	V _{DD} 电源域测试寄存器	0x0000_0027
LVDCSR	0x110	低电压 / 欠压检测控制和状态寄存器	0x0000_0000
PWRLDOSR	0x11C	电源控制 LDO 状态寄存器	0x30XX_XXXX

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_0001 (仅由 V_{DD} 电源域上电复位来复位)

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							WUPF
类型 / 复位							RC 0
7	6	5	4	3	2	1	0
保留位						PDF	PORF
类型 / 复位						RC 0	RC 1

位	域	描述
[8]	WUPF	外部 WAKEUP 引脚标志位 0: WAKEUP 引脚无效 1: WAKEUP 引脚有效 当 WAKEUP 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。

位	域	描述
[1]	PDF	暂停标志位 0: 从异常 V _{CORE} 关机中唤醒 (V _{CORE} 掉电是意料之外的) 1: 从暂停模式中唤醒 (V _{CORE} 掉电是预期的) 当系统成功进入暂停模式, 该位由硬件置位。该位由软件读取清零。
[0]	PORF	上电复位标志位 0: V _{DD} 电源域复位未发生 1: V _{DD} 电源域复位发生 当 V _{DD} 复位发生时, 无论是 V _{DD} 电源域硬件上电复位还是 V _{DD} 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 _{DD} 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 _{DD} 电源域是否访问就绪。

电源控制寄存器 – PWRCR

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 (仅由 V_{DD} 域上电 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
ULDOSTS	保留位					WUPIEN	WUPEN
RO	0					RW	0
7	6	5	4	3	2	1	0
ULDOON	保留位	LDOFTRM		LDOOFF	LDOLCM	保留位	PWRST
RW	0	RW	0	RW	0	RW	0

位	域	描述
[15]	ULDOSTS	超低功耗稳压器 ULDO 状态位 如果该寄存器中的 ULDOON 位为 1, 当 MCU 从深度休眠模式 2 唤醒后, 该位被置为 1。 如果 ULDOON 位为 0, 或者发生上电复位 / 掉电复位, 该位被清零。
[9]	WUPIEN	外部 WAKEUP 引脚中断使能位 0: 除能外部 WAKEUP 引脚中断功能 1: 使能外部 WAKEUP 引脚中断功能 当 WUPEN 和 WUPF 位都被设置为 1 时, 软件可以设置 WUPIEN 位为 1 来触发 NVIC 单元的 WKUP 中断。

位	域	描述
[8]	WUPEN	<p>外部 WAKEUP 引脚使能位</p> <p>0: 除能外部 WAKEUP 引脚功能 1: 使能外部 WAKEUP 引脚功能</p> <p>在进入省电模式之前, 软件可以设置 WUPEN 位为 1 以使能 WAKEUP 引脚功能。当 WUPEN = 1 时, WAKEUP 引脚的上升沿将系统从省电模式中唤醒。由于 WAKEUP 引脚是高电平有效, 该引脚应设置为输入下拉模式。</p> <p>注: 该位由 V_{DD} 电源域复位来复位。因为该位位于 V_{DD} 电源域, 复位启动后, 会有一定的延迟, 直到该位有效。该位将一直无效, 直到系统复位完成, V_{DD} 电源域 ISO 信号除能。这意味着在系统复位完成, V_{DD} 域 ISO 信号除能后, 该位不能立即由软件置位。至少需延迟三个 32 kHz 时钟周期, 直到该位的复位活动结束。</p>
[7]	ULDOON	<p>超低功耗稳压器 ULDO 控制位</p> <p>0: ULDO 关闭 1: ULDO 开启</p> <p>当 CPU 进入深度休眠模式 (SLEEPDEEP = 1), ULDO 运行作为 V_{CORE} 电源域的一个电压源。控制位 ULDOON 由软件置位, 由软件或 V_{DD} 电源域复位清零。如果 ULDOON 位设置为 1, 当 CPU 进入深度休眠模式, 主 LDO 将自动关闭。</p>
[5:4]	LDOFTRM	<p>LDO 输出电压微调</p> <p>00: LDO 默认输出电压 01: LDO 默认输出电压偏移量 -5 % 10: LDO 默认输出电压偏移量 +3 % 11: LDO 默认输出电压偏移量 +7 %</p> <p>当 LDO 进入暂停模式或 V_{DD} 电源域复位, 该位域将被清零。</p>
[3]	LDOOFF	<p>主 LDO 工作模式控制位</p> <p>0: 当 CPU 进入深度休眠模式 (SLEEPDEEP = 1), LDO 工作在小电流模式。V_{CORE} 电源可用, 由 ULDO 提供 1: 当 CPU 进入深度休眠模式 (SLEEPDEEP = 1), LDO 关闭。V_{CORE} 电源不可用, 系统进入暂停模式</p> <p>注: 该位仅在 ULDOON 位清除为 0 时有效。</p>
[2]	LDOLCM	<p>LDO 小电流模式</p> <p>0: 在正常电流模式下, 使用主 LDO 提供 V_{CORE} 电源域电压 1: 在小电流模式下, 使用 ULDO 提供 V_{CORE} 电源域电压</p> <p>注: 该位仅在 MCU 运行模式时可用。当 LDOLCM 置位时, ULDO 输出电流的能力将限制在 5 mA 以下, 有较小的静态电流。适用于 MCU 工作在低速系统时钟时, 可具有较小电流功耗。该位当 MCU 进入暂停模式或 V_{DD} 电源域复位时清零。</p>
[0]	PWRST	<p>V_{DD} 电源域软件复位</p> <p>0: 无动作 1: V_{DD} 电源域软件复位启动</p> <p>当设置此位为高, 所有相关的 RTC 和 PWRCU 寄存器都会被复位。</p>

V_{DD} 电源域测试寄存器 – PWRTEST

该寄存器为软件定义了一个只读值来确认 V_{DD} 电源域是否就绪可进行访问。

偏移量：0x108

复位值：0x0000_0027

	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								
	PWRTEST															
类型 / 复位	RO	0	RO	0	RO	1	RO	0	RO	0	RO	1	RO	1	RO	1

位	域	描述
[7:0]	PWRTEST	V _{DD} 电源域测试位 当 V _{DD} 电源域为 CPU 的访问做好准备，读取该寄存器将得到恒定值 0x27。

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和欠压检测器的标志位、使能位和选项位。

偏移量： 0x110
复位值： 0x0000_0000 (仅由 V_{DD} 电源域上电复位而复位)

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	LVDS [2]	LVDEWEN	LVDIWEN	LVDF	LVDS [1:0]		LVDEN		
类型 / 复位									
	RW	0	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				BODF	保留位	BODRIS	BODEN		
类型 / 复位									
				RO	0	RW	0	RW	0

位	域	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或暂停模式中唤醒，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 标志位将有效。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 有关 LVD 可编程阈值电压的详细信息，请参考相关数据手册 (Datasheet) 中的电气特性部分。
[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 (ULDO 打开，主 LDO 关闭) 或暂停模式 (ULDO 和 主 LDO 都关闭) 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期间 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	域	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: 除能欠压检测器 1: 使能欠压检测器

电源控制 LDO 状态寄存器 – PWRLDOSR

该寄存器定义了主 LDO 和 ULDO 运行状态。

偏移量: 0x11C
复位值: 0x30XX_XXXX

	31	30	29	28	27	26	25	24
	保留位		LDOPST		保留位		ULDOPST	
类型 / 复位	RO		1	RO	1	RO	0	RO
	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							

位	域	描述
[29:28]	LDOPST	主 LDO 运行状态 00: MLDO 关闭 01: MLDO 正在开启 10: MLDO 正在关闭 11: MLDO 开启
[25:24]	ULDOPST	超低功耗 LDO 运行状态 00: ULDO 关闭 01: ULDO 正在开启 10: ULDO 正在关闭 11: ULDO 开启

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)、锁相环 (PLL)、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、LSE 或 PLL。看门狗定时器和实时时钟 RTC 使用 LSI 或 LSE 作为其时钟源。系统时钟 f_{CK_AHB} 的最大工作频率可达 72 MHz。

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。

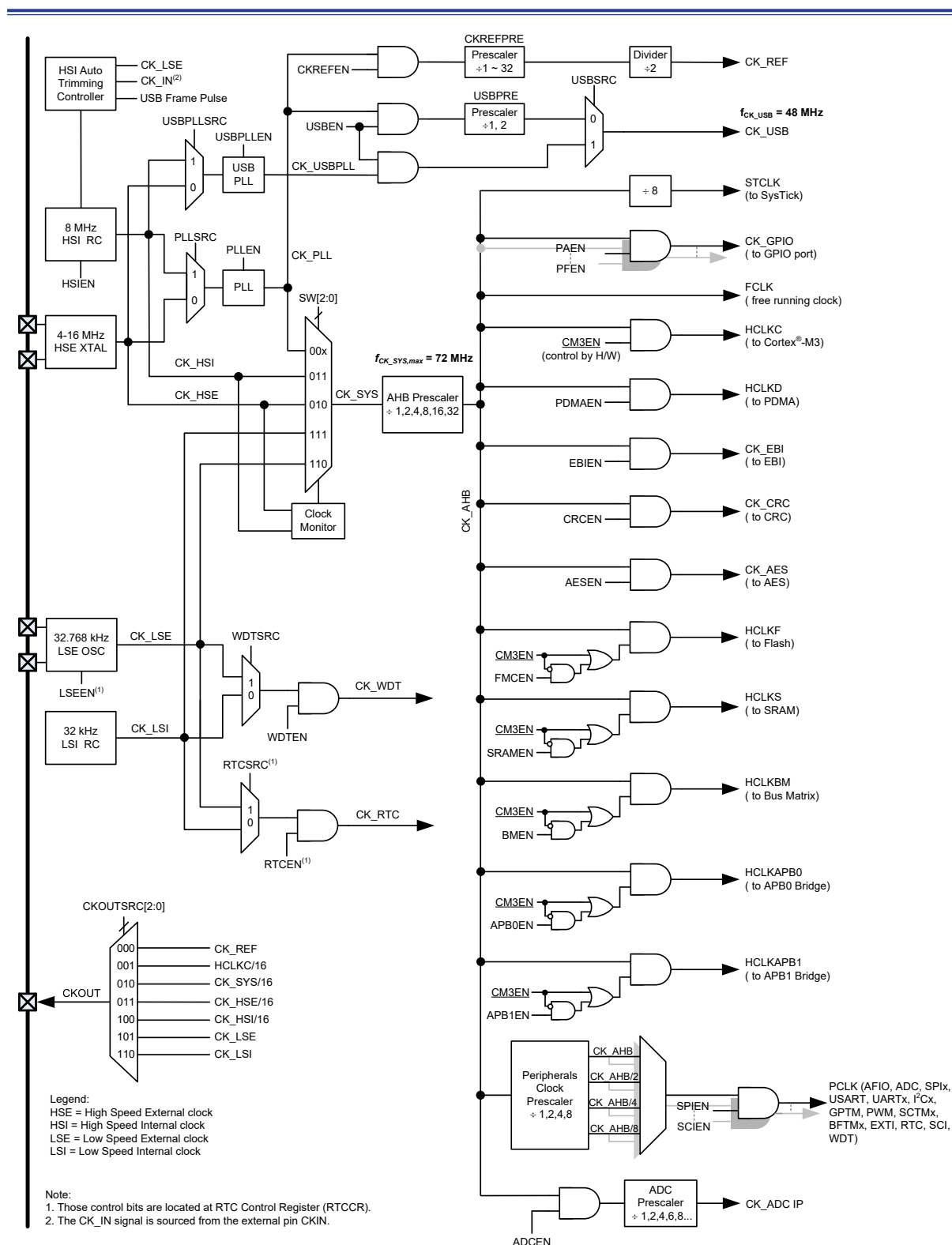


图 13. CKCU 方框图

特性

- 4 ~ 16 MHz 外部高速晶体振荡器 – HSE
- 8 MHz 内部高速 RC 振荡器 – HSI，具有配置选项校准和自定义微调功能
- PLL 可选时钟源 (来自于 HSE 或 HSI) 用于系统时钟
- 32,768 Hz 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 ~ 16 MHz，可以产生高精度度的时钟源作为系统时钟使用。相关硬件配置详见下图。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。为了适当的振荡，外部电阻和电容元件连接到晶体上是必要的。

下面内容介绍了提高晶体电路布局的稳定性。

- 晶体振荡器应尽可能接近单片机，布线长度尽量缩短以减少寄生电容。
- 通过铺地接线来屏蔽晶体附近的任何线路以隔离信号和减小噪声。
- 保持频繁切换信号线离开晶体区，以防止串扰。

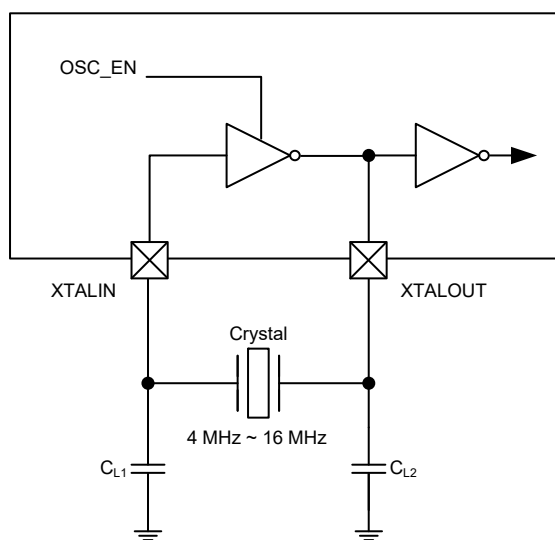


图 14. HSE 外部晶体、陶瓷谐振器

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 标志位表示外部高速晶体振荡器是否稳定。当开启 HSE 时, HSE 时钟不会被释放来使用, 直到 HSERDY 位由硬件置位。这个特定的延迟时间被称为振荡器的“启动时间”。当 HSE 变稳定, 如果全局时钟中断寄存器 GCIR 中相关的中断使能位 HSERDYIE 被置位, 将产生一个中断。此时, HSE 时钟可以直接用作系统时钟源或 PLL 输入时钟。

内部高速 RC 振荡器 – HSI

内部高速 8 MHz RC 振荡器 HSI 是单片机上电时 CPU 的默认时钟源。HSI RC 振荡器提供了一个较低成本类型的时钟源, 无需外部元件。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。当 HSI 变得稳定, 如果全局时钟中断寄存器 GCIR 中相关的中断使能位 HSIRDYIE 被置位, 将产生一个中断。HSI 时钟还可以用作 PLL 的输入时钟。

HSI 的频率精准度可由配置选项校准, 但其工作频率仍不如 HSE 准确。由应用需求、环境和成本决定选择哪种振荡器类型。

通过软件可以设置 PSRCEN 位 (省电唤醒 RC 时钟使能位) 为 1, 以迫使系统最初唤醒时使用 HSI 时钟作为系统时钟。随后, 当初时钟源 HSE 或 PLL 就绪标志位被置位, 系统时钟会自动切换回初始时钟源 HSE 或 PLL。当使用 HSE 或 PLL 作为系统时钟, 此功能会减少唤醒时间。

自动微调内部高速 RC 振荡器 – HSI

内部高速 RC 振荡器 HSI 的频率精准度可随着单片机制造过程的变化而不同。也就是说, 工厂校准后的单片机精准度在 $V_{DD} = 3.3V$ 和 $T_A = 25^\circ C$ 条件下是 $\pm 1.5\%$, 但这个精准度不一定会满足一些应用领域和环境需求。因此, 单片机提供了 HSI 频率校准的微调机制, 使用更加精准的外部参考时钟。详细的方框图请参考下图。

复位后, 工厂微调值加载到 HSI 控制寄存器 HSICR 中的 HSICOARSE[4:0] 位和 HSIFINE[7:0] 位。HSI 的频率精准度会受到电压和温度改变而影响。如果应用需要更精准的 HSI 频率, 用户可以使用 HSI 控制寄存器 HSICR 中的 HSIFINE[7:0] 位手动微调, 或通过外部参考时钟使用自动微调控制器 ATC 来自动调整。参考时钟可以来自于以下时钟源:

- 32,768 Hz 的外部低速晶振或陶瓷谐振器振荡器 LSE 输出时钟
- 1 kHz USB 帧脉冲
- 外部引脚 CKIN 1 kHz 脉冲

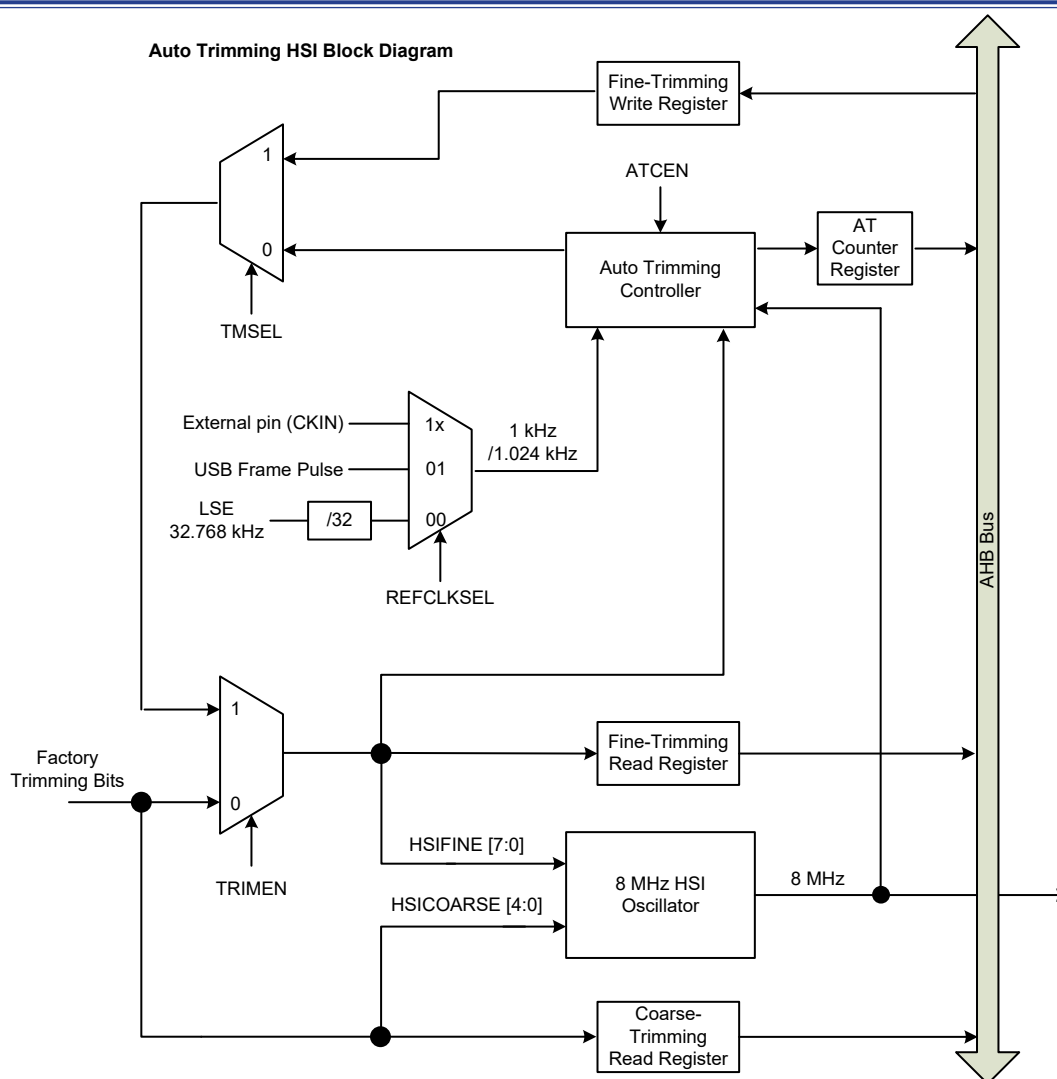


图 15. HSI 自动微调方框图

锁相环-PLL

内部锁相环 PLL 可以提供 8 ~ 80 MHz 的时钟输出, 是 4 ~ 16 MHz 基本参考频率的 2 ~ 32 倍。时钟合成器的原理依赖于 PLL, 包括参考分频器、反馈分频器、数字相位频率检测器 PFD、电流控制的充电泵 CP、内部环路滤波器和电压控制的振荡器 VCO, 以实现稳定的锁相状态。

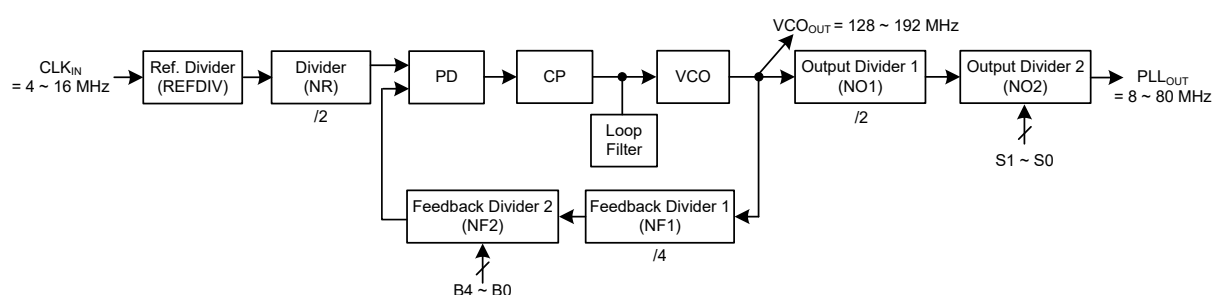


图 16. PLL 方框图

6 时钟控制单元 (CKCU)

PLL 输出时钟频率可由以下公式决定:

$$\begin{aligned}\text{PLL}_{\text{OUT}} &= \text{CLK}_{\text{IN}} \times \frac{\text{NF1} \times \text{NF2}}{\text{REFDIV} \times \text{NR} \times \text{NO1} \times \text{NO2}} = \text{CLK}_{\text{IN}} \times \frac{4 \times \text{NF2}}{\text{REFDIV} \times 2 \times 2 \times \text{NO2}} \\ &= \text{CLK}_{\text{IN}} \times \frac{\text{NF2}}{\text{REFDIV} \times \text{NO2}}\end{aligned}$$

这里, REFDIV = 1 或 2, NR = 参考分频器 = 2, NF1 = 反馈分频器 1 = 4, NF2 = 反馈分频器 2 = 1 ~ 32, NO1 = 输出分频器 1 = 2, NO2 = 输出分频器 2 = 1、2、4 或 8。

考虑到占空比为 50 % 时输入和输出频率都除以 2。如果一个给定的 PLL 输入时钟源频率 CLK_{IN} , 将产生一个特定的 PLL 输出频率, 那么建议使用一个更大的 NF2 值, 以增加 PLL 稳定性并且利用调整时间来减少抖动。对输出和反馈分频器 2 设置位的描述如下两表。下表中所有的设置位 S1 ~ S0 和 B4 ~ B0 以及旁路模式控制, 定义在寄存器定义章节的 PLL 配置寄存器 PLLCFGR 和 PLL 控制寄存器 PLLCR 中。注意, VCO_{OUT} 频率必须在 128 MHz ~ 192 MHz 范围内。如果所选的配置超出这个范围, 则 PLL 输出频率不能保证符合上述 PLL_{OUT} 公式。

PLL 可以通过使用全局时钟控制寄存器 GCCR 中的 PLLEN 位开启或关闭。全局时钟状态寄存器 GCSR 中的 PLLRDY 标志位表示 PLL 时钟是否稳定。当 PLL 变得稳定, 如果全局时钟中断寄存器 GCIR 中相关的中断使能位 PLLRDYIE 被置位, 将产生一个中断。

表 15. 输出分频器 2 值映射

输出分频器 2 设置位 S1 ~ S0 (PLLCFGR 寄存器中的 POTD 位)	NO2 (输出分频器 2 的值)
00	1
01	2
10	4
11	8

表 16. 反馈分频器 2 值映射

反馈分频器 2 设置位 B4 ~ B0 (PLLCFGR 寄存器中的 PFBD 位)	NF2 (反馈分频器 2 的值)
00000	32
00001	1
00010	2
00011	3
00100	4
00101	5
00110	6
00111	7
01000	8
01001	9
01010	10
01011	11
:	:
:	:
11110	30
11111	31

USB 锁相环 – USB PLL

USB PLL 可以提供 48 MHz 的 USB 外设时钟, 是 4 ~ 16 MHz 基本参考频率的 3 ~ 12 倍。时钟合成器的原理依赖于数字 PLL, 包括参考分频器、反馈分频器、数字相位频率检测器 PFD、电流控制的充电泵 CP、内部环路滤波器和电压控制的振荡器 VCO, 以实现稳定的锁相状态。

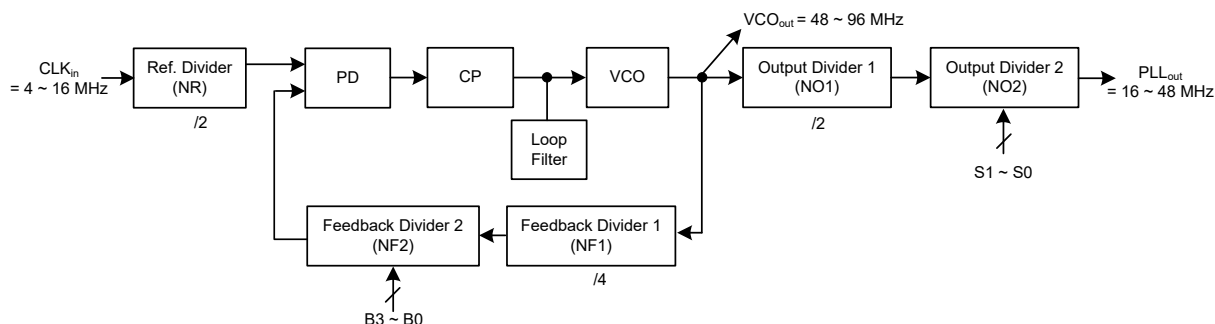


图 17. USB PLL 方框图

PLL 输出时钟频率可由以下公式决定:

$$PLL_{out} = CLK_{in} \times \frac{NF1 \times NF2}{NR \times NO1 \times NO2} = CLK_{in} \times \frac{4 \times NF2}{2 \times 2 \times NO2} = CLK_{in} \times \frac{NF2}{NO2}$$

这里, NR = 参考分频器 = 2, NF1 = 反馈分频器 1 = 4, NF2 = 反馈分频器 2 = 1 ~ 16, NO1 = 输出分频器 1 = 2, NO2 = 输出分频器 2 = 1、2、4 或 8。

考虑到占空比为 50 % 时输入和输出频率都除以 2。如果一个给定的 USB PLL 输入时钟源频率 CLK_{in} , 将产生一个特定的 USB PLL 输出频率, 那么建议使用一个更大的 NF2 值, 以增加 PLL 稳定性并且利用调整时间来减少抖动。对输出和反馈分频器 2 设置位的描述如下两表。下表中所有的设置位 S1 ~ S0 和 B3 ~ B0, 定义在寄存器定义章节的 PLL 配置寄存器 PLLCFGR 和 PLL 控制寄存器 PLLCR 中。注意, VCO_{OUT} 频率必须在 48 MHz ~ 96 MHz 范围内。如果所选的配置超出这个范围, 则 PLL 输出频率不能保证符合上述 PLL_{OUT} 公式。

USB PLL 可以通过使用全局时钟控制寄存器 GCCR 中的 USBPllen 位开启或关闭。全局时钟状态寄存器 GCSR 中的 USBPLLrdy 标志位表示 USB PLL 时钟是否稳定。当 USB PLL 变得稳定, 如果全局时钟中断寄存器 GCIR 中相关的中断使能位 USBPLLrdyIE 被置位, 将产生一个中断。

表 17. USB PLL 输出分频器 2 值映射

输出分频器 2 设置位 S1 ~ S0 (PLLCFGR 寄存器中的 USBPOTD 位)	NO2 (输出分频器 2 的值)
00	1
01	2
10	4
11	8

表 18. USB PLL 反馈分频器 2 值映射

反馈分频器 2 设置位 B3 ~ B0 (PLLCFGR 寄存器中的 USBPFBD 位)	NF2 (反馈分频器 2 的值)
0000	16
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7
1000	8
1001	9
1010	10
1011	11
1100	12
:	:
:	:
1111	15

外部低速晶振 – LSE

外部低速晶体振荡器或陶瓷谐振器频率为 32768 Hz, 可以产生一个低功率但高精度的时钟源用于实时时钟电路、看门狗定时器或系统时钟。相关的硬件配置见下图。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。当 LSE 变得稳定, 如果全局时钟中断寄存器 GCIR 中相关的中断使能位 LSERDYIE 被置位, 将产生一个中断。

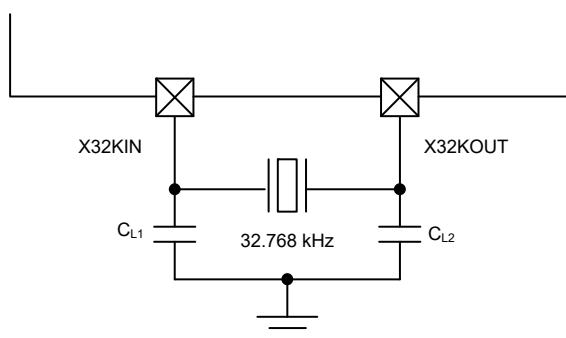


图 18. LSE 外部晶体、陶瓷和谐振器

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电路或看门狗定时器。LSI 提供了一个低成本的时钟源, 无需外部元件。LSI 频率精准度可在数据手册 (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。当 LSI 变得稳定, 如果全局时钟中断寄存器 GCIR 中相关的中断使能位 LSIRDYIE 被置位, 将产生一个中断。

时钟就绪标志位

CKCU 为 HSI、HSE、PLL、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。在使用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。此外, 如果 GCIR 寄存器的相关中断使能位置位, 则 CKCU 产生中断用以表示指定的时钟已就绪。可通过中断服务子程序来软件清零 GCIR 寄存器中的中断状态位。

系统时钟选择 – CK_SYS

系统复位后, 默认的系统时钟源 CK_SYS 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 值可以切换系统时钟为 HSI、HSE、LSE、LSI 或 PLL 输出时钟。当 SW 值改变, 在目标时钟源稳定之前, CK_SYS 仍会先使用原来的时钟源继续运行。相应的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。当一个时钟源或 PLL 时钟被用作系统时钟源, 停止它是不可能的。关于时钟使能功能的描述如下。

如果下面的任一动作发生时, HSI 始终处于使能状态。

- 使能 PLL 并配置 HSI 为其时钟源 (PLLEN, PLLSRC)
- 使能时钟监控器 (CKMEN)
- 配置时钟切换位切换到 HSI (SW)
- 配置 HSI 使能位为 1 (HSIEN)

如果下面的任一动作发生时, HSE 始终处于使能状态。

- 使能 PLL 并配置 HSE 为其时钟源 (PLLEN, PLLSRC)
- 配置时钟切换位到 HSE (SW)
- 配置 HSE 使能位为 1 (HSEEN)

如果下面的任一动作发生时, PLL 始终处于使能状态。

- 使能 USB 使能位 (USBEN)
- 配置时钟切换位到 PLL (SW)
- 配置 PLL 使能位为 1 (PLLEN)

系统时钟选择的编程指南如下所示。

1. 使能将作为系统时钟或 PLL 输入时钟的任意一个时钟源。
2. 在 HSI 和 HSE 的就绪标志位都有效之后, 配置 PLLSRC 寄存器。
3. 在时钟源相应的就绪标志位置起后配置 SW 位以改变系统时钟源。注意, 如果时钟监控器使能, 且被配置为系统时钟的 PLL 输出或 HSE 时钟卡在了 0 或 1 时, 将强制 HSI 作为系统时钟。

HSE 时钟监测器

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 振荡器启动延迟后开启, 在 HSE 停止时关闭。一旦检测到 HSE 故障, HSE 将自动除能。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。此时如果 GCIR 寄存器中的时钟故障中断使能位 CKSIE 位为高, 则会产生时钟故障中断。这种故障中断被连接到 CPU 的不可屏蔽中断 NMI 中。如果选择 HSE 直接作为系统时钟来源, 当 HSE 振荡器故障发生, HSE 将关闭且系统时钟将由硬件自动切换到 HSI。如果选择 HSE 作为 PLL 电路的时钟输入, PLL 时钟输出作为系统时钟, 当 HSE 故障发生时, PLL 电路将关闭, 同时, HSE 也将关闭。

时钟输出能力

单片机有时钟输出能力, 允许时钟输出在特定的外部输出引脚 CKOUT 上。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 19. CKOUT 的时钟源

CKOUTSRC[2:0]	时钟源
000	$CK_REF = CK_PLL / (CKREFPRE + 1) / 2$
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 20. 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_0302
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
PLLCFGR	0x018	PLL 配置寄存器	0x0000_0000
PLLCR	0x01C	PLL 控制寄存器	0x0000_0000
AHBCFGR	0x020	AHB 配置寄存器	0x0000_0000
AHBCCR	0x024	AHB 时钟控制寄存器	0x0000_00E5
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
APBPCSR0	0x038	APB 外设时钟选择寄存器 0	0x0000_0000
APBPCSR1	0x03C	APB 外设时钟选择寄存器 1	0x0000_0000
HSICR	0x040	HSI 控制寄存器	0xFFFF_0000 (X 表示未定义)
HSIATCR	0x044	HSI 自动微调计数器寄存器	0x0000_0000

寄存器	偏移量	描述	复位值
APBPCSR2	0x048	APB 外设时钟选择寄存器 2	0x0000_0000
LPCR	0x300	低功耗控制寄存器	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 PLL / USB / CKOUT 电路指定时钟源。

偏移量: 0x000

复位值: 0x0000_0302

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

位	域	描述
[31:29]	LPMOD	低功耗模式状态 000: 单片机处于运行状态 001: 单片机曾经进入休眠模式 010: 单片机曾经进入深度休眠模式 1 011: 单片机曾经进入深度休眠模式 2 100: 单片机曾经进入暂停模式 其它值: 保留 由硬件置位和写入 b110 复位。
[23:22]	USBPRE	USB 时钟预分频选择 00: CK_USB = CK_PLL 01: CK_USB = CK_PLL / 2 其它值: 保留 由软件置位和复位来控制 USB 时钟的预分频值。
[15:11]	CKREFPRE	CK_REF 时钟预分频选择 $CK_REF = CK_PLL / (CKREFPRE + 1) / 2$ 00000: CK_REF = CK_PLL / 2 00001: CK_REF = CK_PLL / 4 ... 11111: CK_REF = CK_PLL / 64 由软件置位和复位来控制 CK_REF 时钟预分频值。

位	域	描述
[10]	USBSRC	USB 时钟源选择位 0: CK_PLL 时钟 1: CK_USBPLL 时钟 由软件置位和复位来控制 USB 的时钟源。
[9]	USBPLLSRC	USB PLL 时钟源选择位 0: 外部 4 ~ 16 MHz 晶振 (HSE) 1: 内部 8 MHz RC 振荡器 (HSI) 由软件置位和复位来控制 USB PLL 的时钟源。
[8]	PLLSRC	PLL 时钟源选择位 0: 外部 4 ~ 16 MHz 晶振 (HSE) 1: 内部 8 MHz RC 振荡器 (HSI) 由软件置位和复位来控制 PLL 的时钟源。
[2:0]	CKOUTSRC	CKOUT 时钟源选择位 000: CK_REF, $CK_REF = CK_PLL / (CKREFPRE + 1) / 2$ 001: HCLKC / 16 010: CK_SYS / 16 011: CK_HSE / 16 100: CK_HSI / 16 101: CK_LSE 110: CK_LSI 111: 保留 由软件置位和复位来控制 CKOUT 的时钟源。

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

类型 / 复位	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						PSRCEN	CKMEN
							RW	0 RW 0
类型 / 复位	15	14	13	12	11	10	9	8
	保留位				HSIEN	HSEEN	PLLEN	HSEGAIN
					RW	1 RW	0 RW	0 RW 0
类型 / 复位	7	6	5	4	3	2	1	0
	保留位				USBPLLEN	SW		
					RW	0 RW	0 RW	1 RW 1

位	域	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 / 深度休眠 2 唤醒后, 使用内部 8 MHz RC 时钟 (HSI) 作为系统时钟。在进入深度休眠 1 / 深度休眠 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 / 深度休眠 2 模式唤醒后将选择 HSI 作为系统时钟源。因为 HSI 提供时钟给 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 4 ~ 16 MHz 晶振 HSE 时钟监控除能 1: 外部 4 ~ 16 MHz 晶振 HSE 时钟监控使能 当硬件检测出 HSE 时钟卡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。
[11]	HSIEN	内部高速振荡器使能位 0: 内部 8 MHz RC 振荡器除能 1: 内部 8 MHz 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 4 ~ 16 MHz 晶振除能 1: 外部 4 ~ 16 MHz 晶振使能 由软件置位和复位。如果使用 HSE 作为系统时钟或 PLL 的输入时钟, 该位不能被复位。
[9]	PLLEN	PLL 使能 0: PLL 关闭 1: PLL 开启 由软件置位和复位。如果 PLL 时钟被用作系统时钟, 此位不会被复位。
[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	域	描述
[3]	USBPLEN	USB PLL 使能 0: USB PLL 关闭 1: USB PLL 开启 由软件置位和复位 USB PLL。如果 PLL 时钟被用作系统时钟，此位不会被复位。
[2:0]	SW	系统时钟切换 00x: 选择 CK_PLL 作为系统时钟 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 其它值: CK_HSI 作为系统时钟 通过软件置位和复位该位域来选择 CK_SYS 时钟源。当 HSE 直接或间接作为系统时钟来源，且 HSE 时钟监控使能，如果 HSE 时钟故障被检测到，则 HSI 振荡器将强制作为系统时钟。 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 位域控制，由时钟源状态寄存器 CKST 的 CKSWST 位域来显示当前作为系统时钟源的时钟。

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
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		LSIRDY	LSERDY	HSIRDY	HSERDY	PLLRDY	USBPLLRDY
		RO	1 RO	0 RO	1 RO	0 RO	0 RO

位	域	描述
[5]	LSIRDY	内部低速振荡器就绪标志位 0: 内部 32 kHz RC 振荡器未就绪 1: 内部 32 kHz RC 振荡器就绪 由硬件置位以表示 LSI 振荡器已稳定可使用。
[4]	LSERDY	外部低速振荡器 LSE 就绪标志位 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 由硬件置位以表示 LSE 振荡器已稳定可使用。

位	域	描述
[3]	HSIRDY	内部高速振荡器就绪标志位 0: 内部 8 MHz RC 振荡器未就绪 1: 内部 8 MHz RC 振荡器就绪 由硬件置位以表示 HSI 振荡器已稳定可使用。
[2]	HSERDY	外部高速振荡器就绪标志位 0: 外部 4 ~ 16 MHz 振荡器未就绪 1: 外部 4 ~ 16 MHz 振荡器就绪 由硬件置位以表示 HSE 振荡器已稳定可使用。
[1]	PLLRDY	PLL 时钟就绪标志位 0: PLL 未就绪 1: PLL 就绪 由硬件置位以表示 PLL 时钟已稳定可使用。
[0]	USBPLLRDY	USB PLL 时钟就绪标志位 0: USB PLL 未就绪 1: USB PLL 就绪 由硬件置位以表示 USB PLL 时钟已稳定可使用。

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
类型 / 复位	
	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 LSIRDYIE LSERDYIE HSIRDYIE HSERDYIE PLLRDYIE USBPLLRDYIE CKSIE
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
类型 / 复位	
	7 6 5 4 3 2 1 0
	保留位 LSIRDYF LSERDYF HSIRDYF HSERDYF PLLRDYF USBPLLRDYF CKSF
类型 / 复位	WC 0 WC 0 WC 0 WC 0 WC 0 WC 0

位	域	描述
[22]	LSIRDYIE	LSI 就绪中断使能位 0: LSI 稳定中断除能 1: LSI 稳定中断使能 该位由软件置位和复位来使能或除能 LSI 稳定中断。
[21]	LSERDYIE	LSE 就绪中断使能位 0: LSE 稳定中断除能 1: LSE 稳定中断使能 该位由软件置位和复位来使能或除能 LSE 稳定中断。

位	域	描述
[20]	HSIRDYIE	HSI 就绪中断使能位 0: HSI 稳定中断除能 1: HSI 稳定中断使能 该位由软件置位和复位来使能或除能 HSI 稳定中断。
[19]	HSERDYIE	HSE 就绪中断使能位 0: HSE 稳定中断除能 1: HSE 稳定中断使能 该位由软件置位和复位来使能或除能 HSE 稳定中断。
[18]	PLLRDYIE	PLL 就绪中断使能位 0: PLL 稳定中断除能 1: PLL 稳定中断使能 该位由软件置位和复位来使能或除能 PLL 稳定中断。
[17]	USBPLLRDYIE	USB PLL 就绪中断使能位 0: USB PLL 就绪中断除能 1: USB PLL 就绪中断使能 该位由软件置位和复位来使能或除能 USB PLL 就绪中断。
[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监测到的时钟故障中断。
[6]	LSIRDYF	LSI 就绪中断标志位 0: 不产生 LSI 就绪中断 1: LSI 时钟稳定, 产生 LSI 就绪中断 通过写 1 来清零。当内部 32 kHz RC 振荡器时钟稳定且 LSIRDYIE 位被置位时, 可通过硬件置位。
[5]	LSERDYF	LSE 就绪中断标志位 0: 不产生 LSE 就绪中断 1: LSE 时钟稳定, 产生 LSE 就绪中断 通过写 1 来清零。当外部 32,768 Hz 晶振时钟稳定且 LSERDYIE 位被置位时, 可通过硬件置位。
[4]	HSIRDYF	HSI 就绪中断标志位 0: 不产生 HSI 就绪中断 1: HSI 时钟稳定, 产生 HSI 就绪中断 通过写 1 来清零。当内部 8 MHz RC 振荡器时钟稳定且 HSIRDYIE 位被置位时, 可通过硬件置位。
[3]	HSERDYF	HSE 就绪中断标志位 0: 不产生 HSE 就绪中断 1: HSE 时钟稳定, 产生 HSE 就绪中断 通过写 1 来清零。当外部 4 ~ 16 MHz 晶振时钟稳定且 HSERDYIE 位被置位时, 可通过硬件置位。
[2]	PLLRDYF	PLL 就绪中断标志位 0: 不产生 PLL 就绪中断 1: PLL 时钟稳定, 产生 PLL 就绪中断 通过写 1 来清零。当 PLL 稳定且 PLLRDYIE 位被置位时, 可通过硬件置位。
[1]	USBPLLRDYF	USB PLL 就绪中断标志位 0: 不产生 USB PLL 就绪中断 1: USB PLL 时钟稳定, 产生 USB PLL 就绪中断 通过写 1 来清零。当 USB PLL 稳定且 USBPLLRDYIE 位被置位时, 可通过硬件置位。

位	域	描述
[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 CKMEN 位被置位时，可通过硬件置位。

PLL 配置寄存器 – PLLCFGR

该寄存器定义了 PLL 的配置。

偏移量: 0x018
复位值: 0x0000_0000

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

位	域	描述
[28]	REFDIV	PLL 输入参考时钟分频选择 0: 参考分频器 NR = 2 1: 参考分频器 NR = 4
[27:23]	PFBD	PLL VCO 输出时钟反馈分频器 (B4 ~ B0) PLL VCO 输出时钟经反馈分频器分频。
[22:21]	POTD	PLL 输出时钟分频器 (S1 ~ S0)
[10:7]	USBPFBD	USB PLL VCO 输出时钟反馈分频器 (B3 ~ B0) USB PLL VCO 输出时钟经反馈分频器分频。
[6:5]	USBPOTD	USB PLL 输出时钟分频器 (S1 ~ S0)

PLL 控制寄存器 – PLLCR

该寄存器定义了 PLL 旁路模式。

偏移量: 0x01C
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24
	PLLBPS	保留位						
类型 / 复位	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
	保留位							
类型 / 复位								
	7	6	5	4	3	2	1	0
	保留位							
类型 / 复位								

位	域	描述
[31]	PLLBPS	PLL 旁路模式使能位 0: PLL 旁路模式除能 1: PLL 旁路模式使能, $F_{OUT} = F_{IN}$

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_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	保留位				AHBPRE			
					RW	0	RW	0

位	域	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 由软件置位和复位来控制 AHB 时钟的分频比。

AHB 时钟控制寄存器 – AHBCCR

该寄存器定义了 AHB 外设时钟使能位。

偏移量： 0x024

复位值： 0x0000_00E5

	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		PFEN	保留位	PDEN	PCEN	PBEN	PAEN
类型 / 复位			RW 0		RW 0	RW 0	RW 0	RW 0
	15	14	13	12	11	10	9	8
	AESSEN	保留位	CRCEN	EBIEN	CKREFEN	USBEN		保留位
类型 / 复位	RW 0		RW 0	RW 0	RW 0	RW 0		
	7	6	5	4	3	2	1	0
	APB1EN	APB0EN	BMEN	PDMAEN	保留位	SRAMEN	保留位	FMCEN
类型 / 复位	RW 1	RW 1	RW 1	RW 0		RW 1		RW 1

位	域	描述
[21]	PFEN	通用 I/O 口 PF 时钟使能位 0: PF 时钟除能 1: PF 时钟使能 该位由软件置位和复位。
[19]	PDEN	通用 I/O 口 PD 时钟使能位 0: PD 时钟除能 1: PD 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。
[15]	AESSEN	AES 模块时钟使能位 0: AES 时钟除能 1: AES 时钟使能 该位由软件置位和复位。
[13]	CRCEN	CRC 模块时钟使能位 0: CRC 时钟除能 1: CRC 时钟使能 该位由软件置位和复位。

位	域	描述
[12]	EBIEN	EBI 时钟使能位 0: EBI 时钟除能 1: EBI 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。
[10]	USBEN	USB 时钟使能位 0: USB 时钟除能 1: USB 时钟使能 该位由软件置位和复位。
[7]	APB1EN	APB1 桥时钟使能位 0: 在休眠模式下 APB1 桥时钟由硬件自动除能 1: 在休眠模式下 APB1 桥时钟始终使能 由软件置位和复位。如果 APB1 桥在休眠模式下未使用，用户可将 APB1EN 位清零以降低功耗。
[6]	APB0EN	APB0 桥时钟使能位 0: 在休眠模式下 APB0 桥时钟由硬件自动除能 1: 在休眠模式下 APB0 桥时钟始终使能 由软件置位和复位。如果 APB0 桥在休眠模式下未使用，用户可将 APB0EN 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。如果总线矩阵在休眠模式下未使用，用户可将 BMEN 位清零以降低功耗。
[4]	PDMAEN	外设 DMA 时钟使能位 0: PDMA 时钟除能 1: PDMA 时钟使能 由软件置位和复位。注意，当处理器处于休眠模式时，PDMA 可独立工作。但是，相关的 AHB 总线从机时钟或外设时钟必须使能。
[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 存储器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		
	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	域	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 / 6 由软件置位和复位来选择 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几种 APB 外设时钟使能位。

偏移量: 0x02C

复位值: 0x0000_0000

类型 / 复位	31	30	29	28	27	26	25	24
	保留位							SCIEN
								RW 0
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
	EXTIEN	AFIOEN	保留位		UR1EN	UR0EN	保留位	USREN
	RW 0	RW 0			RW 0	RW 0		RW 0
类型 / 复位	7	6	5	4	3	2	1	0
	保留位		SPI1EN	SPI0EN	保留位		I2C1EN	I2C0EN
			RW 0	RW 0			RW 0	RW 0

位	域	描述
[24]	SCIEN	智能卡接口时钟使能位 0: SCI 时钟除能 1: SCI 时钟使能 由软件置位和复位。
[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。
[8]	USREN	USART 时钟使能位 0: USART 时钟除能 1: USART 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。

位	域	描述
[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。
[1]	I2C1EN	I ² C1 时钟使能位 0: I ² C1 时钟除能 1: I ² C1 时钟使能 由软件置位和复位。
[0]	I2C0EN	I ² C0 时钟使能位 0: I ² C0 时钟除能 1: I ² C0 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	SCTM1EN	SCTM0EN		保留位		ADCCEN
类型 / 复位			RW 0	RW 0				RW 0
	23	22	21	20	19	18	17	16
				保留位			BFTM1EN	BFTM0EN
类型 / 复位							RW 0	RW 0
	15	14	13	12	11	10	9	8
		保留位		PWMEN		保留位		GPTMEN
类型 / 复位				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	域	描述
[29]	SCTM1EN	SCTM1 时钟使能位 0: SCTM1 时钟除能 1: SCTM1 时钟使能 由软件置位和复
[28]	SCTM0EN	SCTM0 时钟使能位 0: SCTM0 时钟除能 1: SCTM0 时钟使能 由软件置位和复位。
[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。

位	域	描述
[17]	BFTM1EN	BFTM1 时钟使能位 0: BFTM1 时钟除能 1: BFTM1 时钟使能 由软件置位和复位。
[16]	BFTM0EN	BFTM0 时钟使能位 0: BFTM0 时钟除能 1: BFTM0 时钟使能 由软件置位和复位。
[12]	PWMEN	PWM 时钟使能位 0: PWM 时钟除能 1: PWM 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	看门狗定时器时钟使能位用于寄存器访问 0: 看门狗定时器时钟除能 1: 看门狗定时器时钟使能 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
	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				PLLST			
类型 / 复位					RO	0	RO	0
	7	6	5	4	3	2	1	0
	保留位				CKSWST			
类型 / 复位					RO	0	RO	1

位	域	描述
[27:24]	HSIST	内部高速时钟 CK_HSI 占用状态位 xxx1: HSI 用于系统时钟 CK_SYS (SW = 0x3) xx1x: HSI 用于 PLL x1xx: HSI 用于时钟监控 1xxx: HSI 用于 USB PLL
[18:16]	HSEST	外部高速时钟 CK_HSE 占用状态位 xx1: HSE 用于系统时钟 CK_SYS (SW = 0x2) x1x: HSE 用于 PLL 1xx: HSE 用于 USB PLL
[11:8]	PLLST	PLL 时钟占用状态位 xxx1: PLL 用于系统时钟 CK_SYS x1xx: PLL 用于 USB 1xxx: PLL 用于 CK_REF
[2:0]	CKSWST	时钟切换状态位 00x: CK_PLL 用于系统时钟 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 该位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		USRCLK	
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	23	22	21	20	19	18	17	16
	保留位		GPTMCLK		保留位			
类型 / 复位			RW	0	RW	0		
	15	14	13	12	11	10	9	8
	BFTM1PCLK		BFTM0PCLK		保留位			
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0
	SPI1PCLK		SPI0PCLK		I2C1PCLK		I2C0PCLK	
类型 / 复位	RW	0	RW	0	RW	0	RW	0

位	域	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
[25:24]	USRCLK	USART 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]	GPTMCLK	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
[15:14]	BFTM1PCLK	BFTM1 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	域	描述
[13:12]	BFTM0PCLK	BFTM0 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
[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
[3:2]	I2C1PCLK	I ² C1 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]	I2C0PCLK	I ² C0 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				SCTM1PCLK		SCTM0PCLK	
类型 / 复位					RW	0	RW	0
	23	22	21	20	19	18	17	16
	保留位						SCIPCLK	
类型 / 复位							RW	0
	15	14	13	12	11	10	9	8
	VDDRCLK		WDTRPCLK		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		ADCCPCLK		EXTIPCLK		AFIOPCLK	
类型 / 复位			RW	0	RW	0	RW	0

位	域	描述
[27:26]	SCTM1PCLK	SCTM1 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
[25:24]	SCTM0PCLK	SCTM0 外设时钟选择 00: PCLK = CK_AHB / 4 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]	SCIPCLK	SCI 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
[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]	WDTRPCLK	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

位	域	描述
[5:4]	ADCCPCLK	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]	EXTIPCLK	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： 0xFFFF_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	X RO	X	
	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	X RW	X	
	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 RW	0		

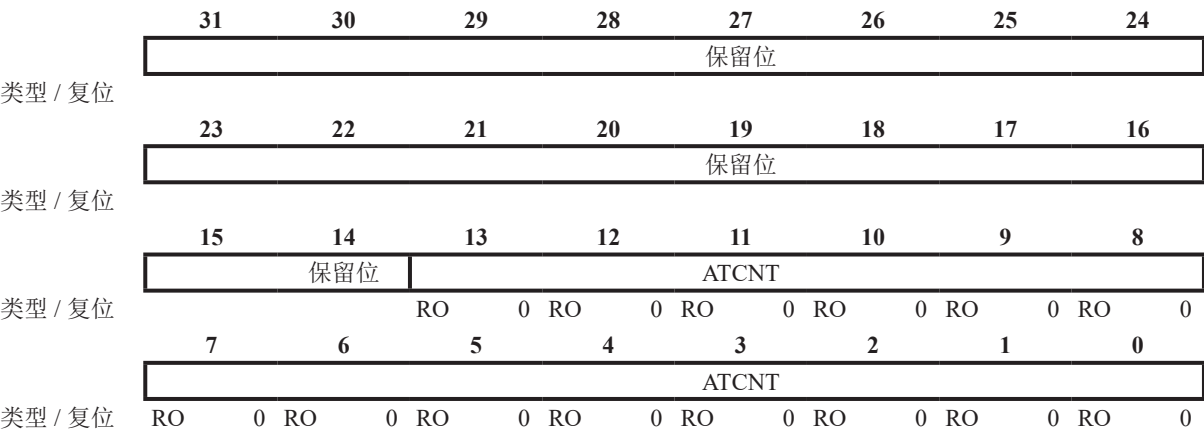
位	域	描述
[28:24]	HSICOARSE	HSI 时钟粗调值 开机时，这些位自动初始化。该值仅由工厂调整，不能通过程序调整。
[23:16]	HSIFINE	HSI 时钟细调值 开机时，这些位自动初始化。该值仅由工厂调整。但这些位也提供了额外的用户程序微调值将增加到 HSICOARSE[4:0] 位，在 HSI 频率受电压或温度影响时，获取更加高的精准度或补偿的变量。它可由软件或带外部参考时钟的自动微调控制器 (ATC) 编程。
[7]	FLOCK	频率锁定 0: HSI 频率没有微调到目标范围 1: HSI 频率微调到目标范围
[6:5]	REFCLKSEL	参考时钟选择 00: 选择 32.768 kHz 外部低速时钟源 (LSE) 01: 选择 1 kHz USB 帧脉冲 1x: 选择外部引脚 CKIN 1 kHz 脉冲 该位为 HSI 自动微调控制器选择参考时钟源
[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，通过 ATC 硬件或用户使用 HSI 控制寄存器中的 HSIFINE[7:0] 位编程来微调。
[3]	ATMSEL	自动微调方式选择 0: 使用二分查找法接近目标范围 1: 使用线性查找法接近目标范围 该位用于选择 HSI RC 振荡器 ATC 硬件的自动微调方式。
[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%。

位	域	描述
[1]	ATCEN	ATC 使能 0: 除能自动微调控制器 1: 使能自动微调控制器
[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	域	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的值。

APB 外设时钟选择寄存器 2 – APBPCSR2

该寄存器定义了 APB 外设时钟分频选择。

偏移量： 0x048
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						PWMPCLK		
	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	域	描述
[17:16]	PWMPCLK	PWM 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。

低功耗控制寄存器 – LPCR

该寄存器定义了低功耗控制位。

偏移量： 0x300

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							USBSLEEP
	7	6	5	4	3	2	1	0
类型 / 复位	保留位							

位	域	描述
[8]	USBSLEEP	USB 休眠软件控制使能位 0: 除能 1: 使能 USB 软件休眠 由软件置位和复位。详细信息请参考电源控制单元章节。

单片机调试控制寄存器 – MCUDBGCR

该寄存器定义了单片机的调试控制。

偏移量: 0x304

复位值: 0x0000_0000 (仅由 V_{DD} 域上电复位来复位)

	31	30	29	28	27	26	25	24
	保留位	DBPWM	保留位					
类型 / 复位	RW 0							
	23	22	21	20	19	18	17	16
	DBSCTM1	DBSCTM0	保留位	DBTRACE	DBUR1	DBUR0	DBBFTM1	DBBFTM0
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
	DBSCI	DBDSLP2	DBI2C1	DBI2C0	DBSPI1	DBSPI0	保留位	DBUSR
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
	保留位	DBGPTM	保留位		DBWDT	DBPD	DBDSLP1	DBSLP
类型 / 复位		RW 0			RW 0	RW 0	RW 0	RW 0

位	域	描述
[30]	DBPWM	PWM 调试模式使能位 0: 即使内核暂停, PWM 计数器仍继续计数 1: 当内核暂停, PWM 计数器停止计数 由软件置位和复位。
[23]	DBSCTM1	SCTM1 调试模式使能位 0: 即使内核暂停, SCTM1 计数器仍继续计数 1: 当内核暂停, SCTM1 计数器停止计数 由软件置位和复位。
[22]	DBSCTM0	SCTM0 调试模式使能位 0: 即使内核暂停, SCTM0 计数器仍继续计数 1: 当内核暂停, SCTM0 计数器停止计数 由软件置位和复位。
[20]	DBTRACE	TRACESWO 调试模式使能位 0: 除能 TRACESWO 输出 1: 使能 TRACESWO 输出 由软件置位和复位。
[19]	DBUR1	UART1 调试模式使能位 0: 与正常模式相同 1: 当内核暂停, UART1 FIFO 溢出计时停止 由软件置位和复位。
[18]	DBUR0	UART0 调试模式使能位 0: 与正常模式相同 1: 当内核暂停, UART0 FIFO 溢出计时停止 由软件置位和复位。
[17]	DBBFTM1	BFTM1 调试模式使能位 0: 即使内核暂停, BFTM1 计数器仍继续计数 1: 当内核暂停, BFTM1 计时器停止计数 由软件置位和复位。

位	域	描述
[16]	DBBFTM0	BFTM0 调试模式使能位 0: 即使内核暂停, BFTM0 计数器仍继续计数 1: 当内核暂停, BFTM0 计时器停止计数 由软件置位和复位。
[15]	DBSCI	SCI 调试模式使能位 0: 与正常模式相同 1: 当内核暂停, SCI 溢出计时停止 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, CM3EN = 0 1: 在深度休眠模式 2 下, LDO = On, FCLK = On, CM3EN = 1 由软件置位和复位。
[13]	DBI2C1	I ² C1 调试模式使能位 0: 超时计数与正常模式相同 1: 当内核暂停时, I ² C1 超时计数停止 由软件置位和复位。
[12]	DBI2C0	I ² C0 调试模式使能位 0: 超时计数与正常模式相同 1: 当内核暂停时, I ² C0 超时计数停止 由软件置位和复位。
[11]	DBSPI1	SPI1 调试模式使能位 0: 与正常模式相同 1: 当内核暂停时, SPI1 FIFO 溢出计数停止 由软件置位和复位。
[10]	DBSPI0	SPI0 调试模式使能位 0: 与正常模式相同 1: 当内核暂停时, SPI0 FIFO 溢出计数停止 由软件置位和复位。
[8]	DBUSR	USART 调试模式使能位 0: 与正常模式相同 1: 当内核暂停时, USART FIFO 溢出计数停止 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。
[2]	DBPD	调试暂停模式 0: 在暂停模式下, LDO = Off, FCLK = Off, CM3EN = 0 1: 在暂停模式下, LDO = On, FCLK = On, CM3EN = 1 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, CM3EN = 0 1: 在深度休眠模式 1 下, LDO = On, FCLK = On, CM3EN = 1 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, CM3EN = 0 1: 在休眠模式下, LDO = On, FCLK = On, CM3EN = 1 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下一章节描述。

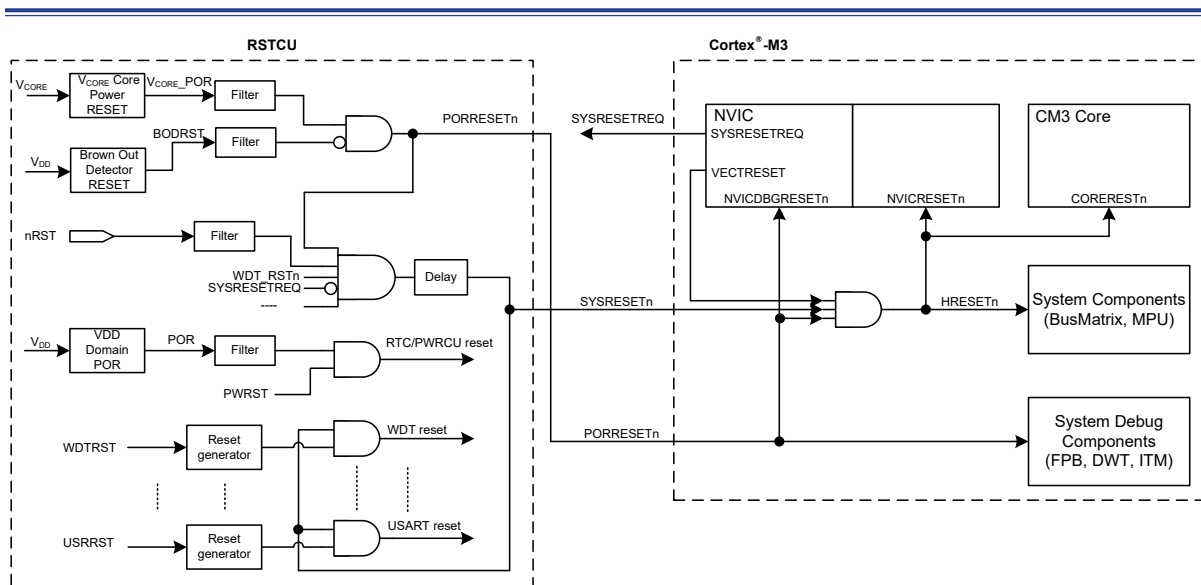


图 19. RSTCU 方框图

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，以防止因噪声而导致错误的复位操作。参照下图， V_{CORE_POR} 信号会一直维持在低电平直到内部 LDO 稳压器准备好提供 V_{CORE} 电源时。除了 V_{CORE_POR} 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。

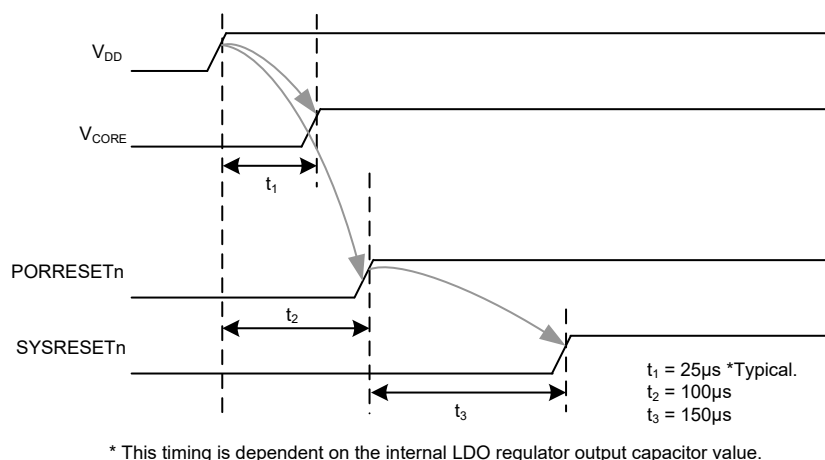


图 20. 上电复位时序图

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 和 VECTRESET 事件的更多信息，请参考 Cortex®-M3 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 USRRST 位产生一个 USART 复位。

寄存器列表

下表显示了 RSTCU 寄存器及其复位值。

表 21. 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

该寄存器定义了多种复位状态位。

偏移量:	0x100
复位值:	0x0000_0008
类型 / 复位	
	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 PORSTF WDTRSTF EXTRSTF SYSRSTF
类型 / 复位	WC 1 WC 0 WC 0 WC 0

位	域	描述
[3]	PORSTF	内核 V _{CORE} 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	域	描述
[0]	SYSRSTF	系统复位标志位 0: 通过 NVIC 建立的系统复位未发生 1: 通过 NVIC 建立的系统复位发生 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

类型 / 复位	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
	AESRST	保留位	PFRST	保留位	PDRST	PCRST	PBRST	PARST
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0
类型 / 复位	CRCRST	EBIRST	USBRST	保留位				DMARST
	RW	0	RW	0	RW	0	RW	0

位	域	描述
[15]	AESRST	AES 复位控制位 0: 无复位 1: AES 复位 该位由软件置位, 由硬件自动清零。
[13]	PFRST	通用 I/O 口 PF 复位控制位 0: 无复位 1: PF 复位 该位由软件置位, 由硬件自动清零。
[11]	PDRST	通用 I/O 口 PD 复位控制位 0: 无复位 1: PD 复位 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: PC 复位 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: PB 复位 该位由软件置位, 由硬件自动清零。

位	域	描述
[8]	PARST	通用 I/O 口 PA 复位控制位 0: 无复位 1: PA 复位 该位由软件置位, 由硬件自动清零。
[7]	CRCRST	CRC 复位控制位 0: 无复位 1: CRC 复位 该位由软件置位, 由硬件自动清零。
[6]	EBIRST	EBI 复位控制位 0: 无复位 1: EBI 复位 该位由软件置位, 由硬件自动清零。
[5]	USBRST	USB 复位控制位 0: 无复位 1: USB 复位 该位由软件置位, 由硬件自动清零。
[0]	DMARST	外设 DMA (PDMA) 复位控制位 0: 无复位 1: 复位外设 DMA (PDMA) 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							SCIRST	
类型 / 复位								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	
	EXTIRST	AFIORST	保留位		UR1RST	UR0RST	保留位	USRRST	
类型 / 复位	RW	0	RW	0		RW	0	RW	0
	7	6	5	4	3	2	1	0	
	保留位		SPI1RST	SPI0RST	保留位		I2C1RST	I2C0RST	
类型 / 复位			RW	0	RW	0		RW	0

位	域	描述
[24]	SCIRST	SCI 复位控制位 0: 无复位 1: 复位 SCI 该位由软件置位, 由硬件自动清零。

位	域	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。
[8]	USRRST	USART 复位控制位 0: 无复位 1: 复位 USART 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。
[1]	I2C1RST	I ² C1 复位控制位 0: 无复位 1: 复位 I ² C1 该位由软件置位, 由硬件自动清零。
[0]	I2C0RST	I ² C0 复位控制位 0: 无复位 1: 复位 I ² C0 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
保留位	SCTM1RST	SCTM0RST	保留位	保留位	保留位	保留位	ADCRST
类型 / 复位	RW	0	RW	0			RW 0
23	22	21	20	19	18	17	16
保留位	保留位	保留位	保留位	保留位	保留位	BFTM1RST	BFTM0RST
类型 / 复位						RW 0	RW 0
15	14	13	12	11	10	9	8
保留位	保留位	保留位	PWMRST	保留位	保留位	保留位	GPTMRST
类型 / 复位			RW 0				RW 0
7	6	5	4	3	2	1	0
保留位	保留位	保留位	WDTRST	保留位	保留位	保留位	保留位
类型 / 复位			RW 0				

位	域	描述
[29]	SCTM1RST	SCTM1 复位控制位 0: 无复位 1: 复位 SCTM1 该位由软件置位，由硬件自动清零。
[28]	SCTM0RST	SCTM0 复位控制位 0: 无复位 1: 复位 SCTM0 该位由软件置位，由硬件自动清零。
[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。
[17]	BFTM1RST	BFTM1 复位控制位 0: 无复位 1: 复位 BFTM1 该位由软件置位，由硬件自动清零。
[16]	BFTM0RST	BFTM0 复位控制位 0: 无复位 1: 复位 BFTM0 该位由软件置位，由硬件自动清零。
[12]	PWMRST	PWM 复位控制位 0: 无复位 1: 复位 PWM 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

单片机有多达 52 个通用 I/O 引脚 (GPIO), 即 PA0 ~ PA15、PB0 ~ PB15、PC0 ~ PC15、PD0 ~ PD2 和 PF0, 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。请参考单片机的数据手册 (Datasheet) 获得更多信息。

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

单片机 GPIO 引脚的外部中断在外部中断控制单元 (EXTI) 都有相关的控制和配置寄存器。

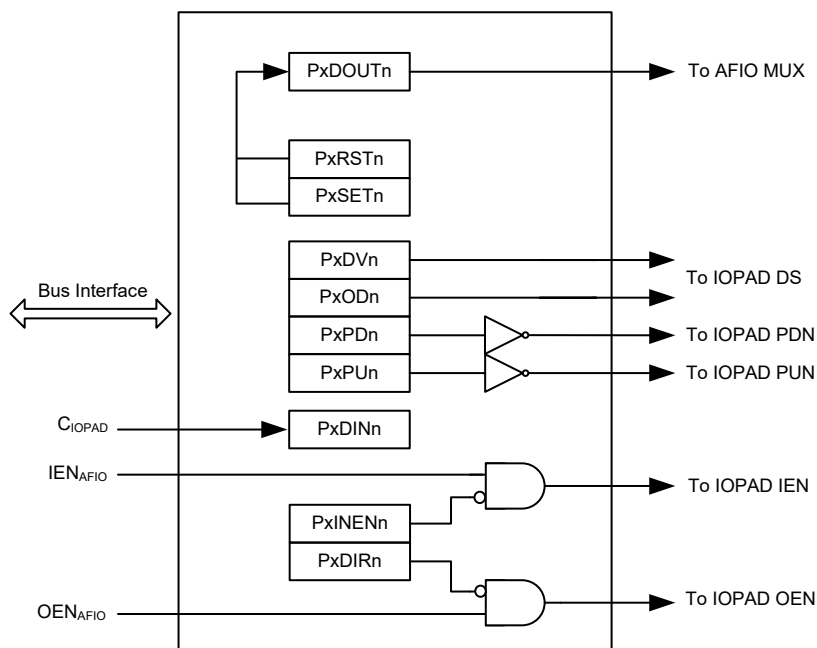


图 21. GPIO 方框图

特性

- 输入 / 输出方向控制
- 施密特触发器输入功能使能控制
- 输入引脚弱上拉 / 下拉控制
- 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 – 通过 EXTI 配置寄存器
- 模拟输入 / 输出配置 – 通过 AFIO 配置寄存器
- 复用功能的输入 / 输出配置 – 通过 AFIO 配置寄存器
- 端口配置锁定

功能描述

默认的 GPIO 引脚配置

在复位期间或之后, 复用功能都无效, GPIO 端口被配置为输入除能浮空模式, 即不带有上拉 / 下拉电阻的输入除能。单片机复位后, 只有启动和串行线调试引脚与 I/O 引脚共用是有效的。

- BOOT0: 带有内部上拉的输入使能
- BOOT1: 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 ~ D、F) 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。当上拉和下拉功能都使能, 上拉功能具有较高的优先级, 上拉功能被除能时才能使能下拉功能。

GPIO 引脚可以配置为输出引脚, 输出数据锁存到数据寄存器 PxDOCTR。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和复位控制寄存器 PxSRR 或端口输出复位控制寄存器 PxRR, 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。端口输出置位和复位功能都使能时, 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。

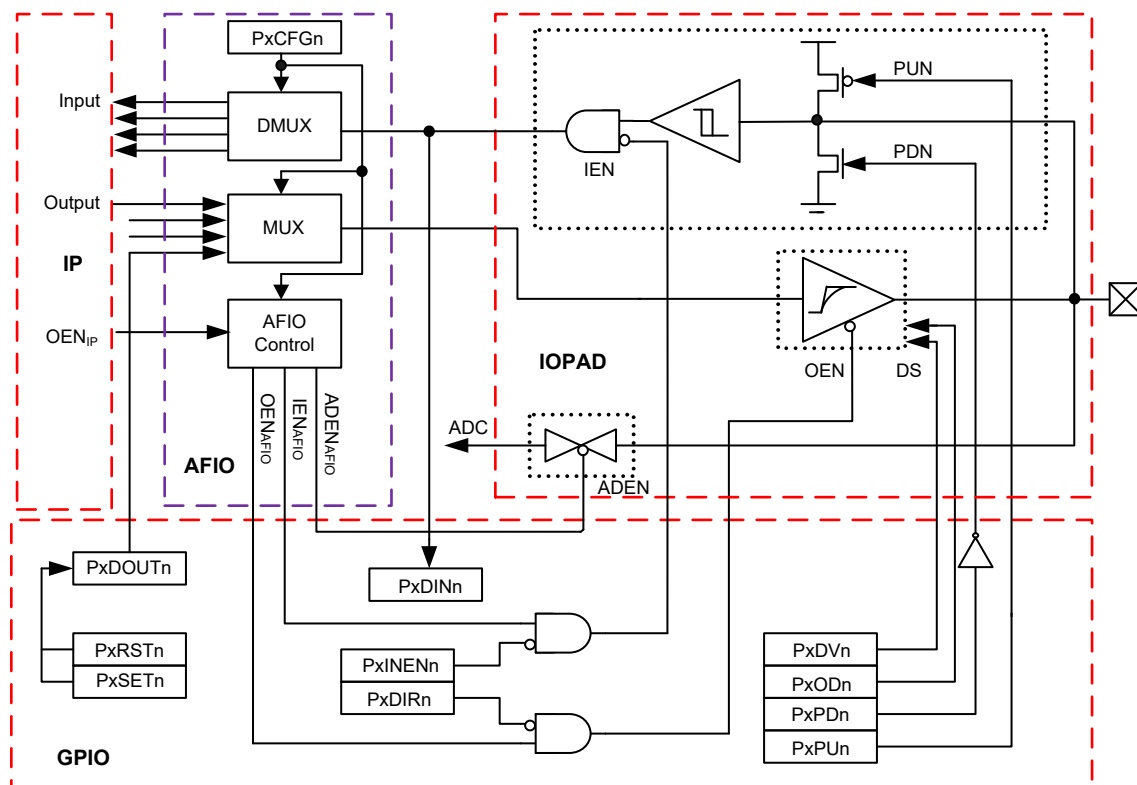


图 22. AFIO/GPIO 控制信号

PxDINn / PxDOUn (x = A ~ D, F): 数据输入 / 数据输出

PxRSTn / PxSETn (x = A ~ D, F): 复位 / 置位

PxDIRn (x = A ~ D, F): 方向

PxDVn (x = A ~ D, F): 输出驱动

PxPDn / PxPUn (x = A ~ D, F): 拉低 / 拉高

PxINEn (x = A ~ D, F): 输入使能

PxODn (x = A ~ D, F): 漏极开路

PxCFGn (x = A ~ D, F): AFIO 配置

表 22. 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	PxINEn	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 PxINEn 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 ~ D、F) 寄存器是用来锁定端口 x 和锁定控制选项。在 PxLOCKR 寄存器 PxLKEY 位域写入值 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 ~ D、F)。如果 PxLOCKR 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 ~ D、F 的复位值。

表 23. 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_0300
PAPUR	0x008	端口 A 上拉选择寄存器	0x0000_3300
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_3300
PADOUTR	0x020	端口 A 数据输出寄存器	0x0000_0000
PASRR	0x024	端口 A 输出置位和复位控制寄存器	0x0000_0000
PARR	0x028	端口 A 输出复位控制寄存器	0x0000_0000
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 输出置位和复位控制寄存器	0x0000_0000
PBRR	0x028	端口 B 输出复位控制寄存器	0x0000_0000
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
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

寄存器	偏移量	描述	复位值
PCSRR	0x024	端口 C 输出置位和复位控制寄存器	0x0000_0000
PCRR	0x028	端口 C 输出复位控制寄存器	0x0000_0000
GPIO D 基址 = 0x400B_6000			
PDDIRCR	0x000	端口 D 数据方向控制寄存器	0x0000_0000
PDINER	0x004	端口 D 输入功能使能控制寄存器	0x0000_0000
PDPUR	0x008	端口 D 上拉选择寄存器	0x0000_0000
PDPDR	0x00C	端口 D 下拉选择寄存器	0x0000_0000
PDODR	0x010	端口 D 漏极开路选择寄存器	0x0000_0000
PDDRVCR	0x014	端口 D 驱动电流选择寄存器	0x0000_0000
PDLOCKR	0x018	端口 D 锁定寄存器	0x0000_0000
PDDINR	0x01C	端口 D 数据输入寄存器	0x0000_0000
PDDOUTR	0x020	端口 D 数据输出寄存器	0x0000_0000
PDSRR	0x024	端口 D 输出置位和复位控制寄存器	0x0000_0000
PDRR	0x028	端口 D 输出复位控制寄存器	0x0000_0000
GPIO F 基址 = 0x400B_A000			
PFDIRCR	0x000	端口 F 数据方向控制寄存器	0x0000_0000
PFINER	0x004	端口 F 输入功能使能控制寄存器	0x0000_0000
PFPUR	0x008	端口 F 上拉选择寄存器	0x0000_0000
PFPDR	0x00C	端口 F 下拉选择寄存器	0x0000_0000
PFODR	0x010	端口 F 漏极开路选择寄存器	0x0000_0000
PFDRVCR	0x014	端口 F 驱动电流选择寄存器	0x0000_0000
PFLOCKR	0x018	端口 F 锁定寄存器	0x0000_0000
PFDINR	0x01C	端口 F 数据输入寄存器	0x0000_0000
PFDOUTR	0x020	端口 F 数据输出寄存器	0x0000_0000
PFSRR	0x024	端口 F 输出置位和复位控制寄存器	0x0000_0000
PFRR	0x028	端口 F 输出复位控制寄存器	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

	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	
	PADIR								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
	PADIR								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	域	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_0300

	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
	7	6	5	4	3	2	1	0
类型 / 复位	PAINEN							
	RW	0	RW	0	RW	0	RW	0

位	域	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_3300

	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
	7	6	5	4	3	2	1	0
类型 / 复位	PAPU							
	RW	0	RW	0	RW	0	RW	0

位	域	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	
	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	
	PAPD								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
	PAPD								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	域	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	域	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) 注: 当漏极开路功能使能时, 引脚 n 内部上拉或下拉设置将无效。

端口 A 驱动电流选择寄存器 – PADRVR

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	域	描述
[31:0]	PADVn[1:0]	GPIO 端口 A 引脚 n 的输出电流驱动选择控制位 (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
	23	22	21	20	19	18	17	16	
	PALKEY								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	15	14	13	12	11	10	9	8	
	PALOCK								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
	PALOCK								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	域	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 位域写入值 0x5FA0。为了在这个锁定寄存器执行一个成功的写操作, 写入 PALKEY 位域的值必须为 0x5FA0。如果写入这个位域值不等于 0x5FA0, PALOCKR 寄存器上的任何写操作将被中止。对 PALKEY 位域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 位域为 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]	PALOCKn	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 位域, PALOCKn 位用来锁定相应 GPIO 引脚的配置。被锁定的配置包括 PADIRn、PAINENn、PAPUn、PAPDn、PAODn 和 PADVn 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 PALOCKn (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_3300

	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								
	PADIN															
类型 / 复位	RO	0	RO	0	RO	1	RO	1	RO	0	RO	0	RO	1	RO	1
	7	6	5	4	3	2	1	0								
	PADIN															
类型 / 复位	RO	0	RO	0	RO	0	RO	0	RO	0	RO	0	RO	0	RO	0

位	域	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	
	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	
	PADOUT								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
	PADOUT								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	域	描述
[15:0]	PADOUTn	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	域	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 位被置位, 都会使 PADOUTn 位复位。
[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	
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	PARST								
	7	6	5	4	3	2	1	0	
类型 / 复位	PARST								
	WO	0	WO	0	WO	0	WO	0	WO
	0		0		0		0		0

位	域	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 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
	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
	PBDIR							
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0
	PBDIR							
类型 / 复位	RW	0	RW	0	RW	0	RW	0

位	域	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	
	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	
	PBINEN								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
	PBINEN								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	域	描述
[15:0]	PBINENn	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	
	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	
	PBPUR								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
	PBPUR								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	域	描述
[15:0]	PBPURn	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	
	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	
	PBPDR								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
	PBPDR								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	域	描述
[15:0]	PBPDRn	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	
	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	
	PBOD								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
	PBOD								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	域	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) 注: 当漏极开路功能使能时, 引脚 n 内部上拉或下拉设置将无效。

端口 B 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				PBDV12
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	23	22	21	20	19	18	17	16	
	PBDV11				PBDV10				PBDV8
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	15	14	13	12	11	10	9	8	
	PBDV7				PBDV6				PBDV4
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
	PBDV3				PBDV2				PBDV0
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	域	描述
[31:0]	PBDVn[1:0]	GPIO 端口 B 引脚 n 的输出电流驱动选择控制位 (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
	23	22	21	20	19	18	17	16	
	PBLKEY								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	15	14	13	12	11	10	9	8	
	PBLOCK								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
	PBLOCK								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	域	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 位域写入值 0x5FA0。为了在这个锁定寄存器执行一个成功的写操作, 写入 PBLKEY 位域的值必须为 0x5FA0。如果写入这个位域值不等于 0x5FA0, PBLOCKR 寄存器上的任何写操作将被中止。对 PBLKEY 位域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 位域为 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 位域, 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

	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	
	PBDIN								
类型 / 复位	RO	0	RO	0	RO	0	RO	0	RO
	7	6	5	4	3	2	1	0	
	PBDIN								
类型 / 复位	RO	0	RO	0	RO	0	RO	0	RO

位	域	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	
	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	
	PBDOUT								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
	PBDOUT								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	域	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	域	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 位被置位, 都会使 PBDOUTn 位复位。
[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	
	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	
	PBRST								
类型 / 复位	WO	0	WO	0	WO	0	WO	0	WO
	7	6	5	4	3	2	1	0	
	PBRST								
类型 / 复位	WO	0	WO	0	WO	0	WO	0	WO

位	域	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 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

	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
	PCDIR							
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0
	PCDIR							
类型 / 复位	RW	0	RW	0	RW	0	RW	0

位	域	描述
[15:0]	PCDIRn	GPIO 端口 C 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 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

类型 / 复位	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
	PCINEN							
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0
类型 / 复位	PCINEN							
	RW	0	RW	0	RW	0	RW	0

位	域	描述
[15:0]	PCINEN _n	GPIO 端口 C 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 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	PCPU								
	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
类型 / 复位	PCPU								
	RW	0	RW	0	RW	0	RW	0	RW

位	域	描述
[15:0]	PCPU _n	GPIO 端口 C 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 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								
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	
	PCPD								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
	PCPD								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	域	描述
[15:0]	PCPDn	GPIO 端口 C 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 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

	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	PCOD								
	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
类型 / 复位	PCOD								
	RW	0	RW	0	RW	0	RW	0	RW

位	域	描述
[15:0]	PCODn	GPIO 端口 C 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) 注: 当漏极开路功能使能时, 引脚 n 内部上拉或下拉设置将无效。

端口 C 驱动电流选择寄存器 – PCDVR

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	
	PCDV15				PCDV14				PCDV13				PCDV12			
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0
	23		22		21		20		19		18		17		16	
	PCDV11				PCDV10				PCDV9				PCDV8			
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0
	15		14		13		12		11		10		9		8	
	PCDV7				PCDV6				PCDV5				PCDV4			
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	
	PCDV3				PCDV2				PCDV1				PCDV0			
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0

位	域	描述
[31:0]	PCDVn[1:0]	GPIO 端口 C 引脚 n 的输出电流驱动选择控制位 (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 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	
	PCLOCK								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
	PCLOCK								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	域	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 位域写入值 0x5FA0。为了在这个锁定寄存器执行一个成功的写操作, 写入 PCLKEY 位域的值必须为 0x5FA0。如果写入这个位域值不等于 0x5FA0, PCLOCKR 寄存器上的任何写操作将被中止。对 PCLKEY 位域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 位域为 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。
[15:0]	PCLOCKn	GPIO 端口 C 引脚 n 锁定控制位 (n = 0 ~ 15) 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 位域, PCLOCKn 位用来锁定相应 GPIO 引脚的配置。被锁定的配置包括 PCDIRn、PCINENn、PCPUn、PCPDn、PCODn 和 PCDVn 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 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 PCLOCKn (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	
	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	
	PCDIN								
类型 / 复位	RO	0	RO	0	RO	0	RO	0	RO
	7	6	5	4	3	2	1	0	
	PCDIN								
类型 / 复位	RO	0	RO	0	RO	0	RO	0	RO

位	域	描述
[15:0]	PCDINn	GPIO 端口 C 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 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

	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	
	PCDOUT								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
	PCDOUT								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	域	描述
[15:0]	PCDOUTn	GPIO 端口 C 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 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	
	PCRST								
类型 / 复位	WO	0	WO	0	WO	0	WO	0	WO
	23	22	21	20	19	18	17	16	
	PCRST								
类型 / 复位	WO	0	WO	0	WO	0	WO	0	WO
	15	14	13	12	11	10	9	8	
	PCSET								
类型 / 复位	WO	0	WO	0	WO	0	WO	0	WO
	7	6	5	4	3	2	1	0	
	PCSET								
类型 / 复位	WO	0	WO	0	WO	0	WO	0	WO

位	域	描述
[31:16]	PCRSTn	GPIO 端口 C 引脚 n 的输出复位控制位 (n = 0 ~ 15) 0: 不影响 PCDOUTn 位 1: 复位 PCDOUTn 位 注意, 当该寄存器中的 PCRSTn 位或 PCRR 寄存器中的 PCRSTn 位被置位, 都会使 PCDOUTn 位复位。
[15:0]	PCSETn	GPIO 端口 C 引脚 n 的输出置位控制位 (n = 0 ~ 15) 0: 不影响 PCDOUTn 位 1: 置位 PCDOUTn 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	PCRST								
	7	6	5	4	3	2	1	0	
类型 / 复位	PCRST								
	WO	0	WO	0	WO	0	WO	0	WO
	0		0		0		0		0

位	域	描述
[15:0]	PCRSTn	GPIO 端口 C 引脚 n 的输出复位控制位 (n = 0 ~ 15) 0: 不影响 PCDOUn 位 1: 复位 PCDOUn 位

端口 D 数据方向控制寄存器 – PDDIRCR

该寄存器用来控制 GPIO 端口 D 引脚的方向作为输入或输出。

偏移量：0x000

复位值：0x0000_0000

类型 / 复位	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					PDDIR		
						RW	0	RW
							0	RW
								0

位	域	描述
[2:0]	PDDIRn	GPIO 端口 D 引脚 n 的方向控制位 (n = 0 ~ 2) 0: 引脚 n 处于输入模式 1: 引脚 n 处于输出模式

端口 D 输入功能使能控制寄存器 – PDINER

此寄存器用来使能或除能 GPIO 端口 D 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	保留位							
	7	6	5	4	3	2	1	0
类型 / 复位	保留位				PDINEN			
					RW	0	RW	0

位	域	描述
[2:0]	PDINENn	GPIO 端口 D 引脚 n 的输入使能控制位 (n = 0 ~ 2) 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 D 上拉选择寄存器 – PDPUR

该寄存器用来使能或除能 GPIO 端口 D 的上拉功能。

偏移量： 0x008

复位值： 0x0000_0000

类型 / 复位	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				PDPUn			
					RW	0	RW	0

位	域	描述
[2:0]	PDPUn	GPIO 端口 D 引脚 n 的上拉选择控制位 (n = 0 ~ 2) 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 D 下拉选择寄存器 – PDPDR

该寄存器用来使能或除能 GPIO 端口 D 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	保留位							
	7	6	5	4	3	2	1	0
类型 / 复位	保留位				PDPD			
					RW	0	RW	0
						0	RW	0

位	域	描述
[2:0]	PDPDn	GPIO 端口 D 引脚 n 的下拉选择控制位 (n = 0 ~ 2) 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 D 漏极开路选择寄存器 – PDODR

该寄存器用来使能或除能 GPIO 端口 D 的漏极开路功能。

偏移量： 0x010

复位值： 0x0000_0000

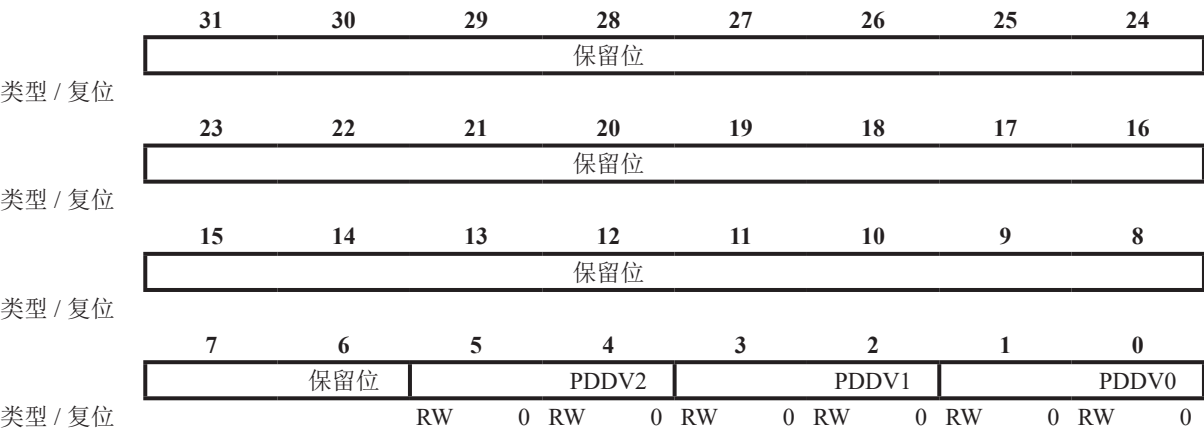
	31	30	29	28	27	26	25	24
	保留位							
类型 / 复位								
	23	22	21	20	19	18	17	16
	保留位							
类型 / 复位								
	15	14	13	12	11	10	9	8
	保留位							
类型 / 复位								
	7	6	5	4	3	2	1	0
	保留位				PDOD			
类型 / 复位					RW	0	RW	0

位	域	描述
[2:0]	PDODn	GPIO 端口 D 引脚 n 的漏极开路选择控制位 (n = 0 ~ 2) 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) 注: 当漏极开路功能使能时, 引脚 n 内部上拉或下拉设置将无效。

端口 D 驱动电流选择寄存器 – PDDVR

该寄存器用来定义 GPIO 端口 D 的输出驱动电流。

偏移量： 0x014
复位值： 0x0000_0000



位	域	描述
[5:0]	PDDVn[1:0]	GPIO 端口 D 引脚 n 的输出电流驱动选择控制位 (n = 0 ~ 2) 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 D 锁定寄存器 – PDLOCKR

该寄存器定义了 GPIO 端口 D 的锁定配置。

偏移量： 0x018
复位值： 0x0000_0000

	31	30	29	28	27	26	25	24	
	PDLKEY								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	23	22	21	20	19	18	17	16	
	PDLKEY								
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	
	保留位					PDLOCK			
类型 / 复位						RW	0	RW	0

位	域	描述
[31:16]	PDLKEY	GPIO 端口 D 锁定键 0x5FA0：使能端口 D 锁定功能 其它：除能端口 D 锁定功能 为了锁定端口 D 功能，在该寄存器 PDLKEY 位域写入值 0x5FA0。为了在这个锁定寄存器执行一个成功的写操作，写入 PDLKEY 位域的值必须为 0x5FA0。如果写入这个位域值不等于 0x5FA0，PDLOCKR 寄存器上的任何写操作将被中止。对 PDLKEY 位域的一个读操作的结果将返回 GPIO 端口 D 锁定状态，表示 GPIO 端口 D 是否被锁定。如果读值，PDLKEY 位域为 0，表明 GPIO 端口 D 锁定功能除能。否则，它表明 GPIO 端口 D 锁定功能使能，读值等于 1。
[2:0]	PDLOCKn	GPIO 端口 D 引脚 n 的锁定控制位 (n = 0 ~ 2) 0：端口 D 引脚 n 未锁定 1：端口 D 引脚 n 锁定 当正确的锁定键应用于 PDLKEY 位域，PDLOCKn 位用来锁定相应 GPIO 引脚的配置。被锁定的配置包括 PDDIRn、PDINENn、PDPUn、PDPDn、PDODn 和 PDDVn 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 GPDCFGHR 或 GPDCFGRLR 寄存器也将被锁定。请注意，PDLOCKR 寄存器只能写入一次，这意味着 PDLKEY 和 PDLOCKn (锁定控制位) 应一起写入，不能改变直到系统复位或 GPIO 端口 D 发生复位。

端口 D 数据输入寄存器 – PDDINR

该寄存器指定 GPIO 端口 D 的输入数据。

偏移量: 0x01C
复位值: 0x0000_0000

类型 / 复位	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					PDDIN		
						RO	0 RO	0 RO 0

位	域	描述
[2:0]	PDDINn	GPIO 端口 D 引脚 n 的数据输入位 (n = 0 ~ 2) 0: 引脚 n 的输入数据为 0 1: 引脚 n 的输入数据为 1

端口 D 输出数据寄存器 – PDDOUTR

该寄存器指定 GPIO 端口 D 的输出数据。

偏移量： 0x020
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				PDDOUT				
类型 / 复位					RW	0	RW	0	RW
									0

位	域	描述
[2:0]	PDDOUTn	GPIO 端口 D 引脚 n 的数据输出位 (n = 0 ~ 2) 0: 引脚 n 的输出数据为 0 1: 引脚 n 的输出数据为 1

端口 D 输出置位 / 复位控制寄存器 – PDSRR

该寄存器用来置位或复位 GPIO 端口 D 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
	保留位							
类型 / 复位								
	23	22	21	20	19	18	17	16
	保留位					PDRST		
类型 / 复位						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					PDSET		
类型 / 复位						WO	0	WO

位	域	描述
[18:16]	PDRSTn	GPIO 端口 D 引脚 n 的输出复位控制位 (n = 0 ~ 2) 0: 不影响 PDDOUTn 位 1: 复位 PDDOUTn 位 注意, 当该寄存器中的 PDRSTn 位或 PDRR 寄存器中的 PDRSTn 位被置位, 都会使 PDDOUTn 位复位。
[2:0]	PDSETn	GPIO 端口 D 引脚 n 的输出置位控制位 (n = 0 ~ 2) 0: 不影响 PDDOUTn 位 1: 置位 PDDOUTn 位 注意, 如果 PDSETn 和 PDRSTn 位同时被置位, PDSETn 位使能的功能有比较高的优先级。

端口 D 输出复位寄存器 – PDRR

该寄存器用来复位 GPIO 端口 D 输出数据相应的位。

偏移量： 0x028
复位值： 0x0000_0000

类型 / 复位	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					PDRST		
						WO	0 WO	0 WO 0

位	域	描述
[2:0]	PDRSTn	GPIO 端口 D 引脚 n 的输出复位控制位 (n = 0 ~ 2) 0: 不影响 PDDOUTn 位 1: 复位 PDDOUTn 位

端口 F 数据方向控制寄存器 – PFDIRCR

该寄存器用来控制 GPIO 端口 F 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
	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							PFDIR
类型 / 复位								RW 0

位	域	描述
[0]	PFDIR	GPIO 端口 F 引脚 n 的方向控制位 (n = 0) 0: 引脚 n 处于输入模式 1: 引脚 n 处于输出模式

端口 F 输入功能使能控制寄存器 – PFINER

此寄存器用来使能或除能 GPIO 端口 F 的输入功能。

偏移量： 0x004

复位值： 0x0000_0000

类型 / 复位	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							PFINEN
								RW 0

位	域	描述
[0]	PFINENn	GPIO 端口 F 引脚 n 的输入使能控制位 (n = 0) 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 F 上拉选择寄存器 – PFPUR

该寄存器用来使能或除能 GPIO 端口 F 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	保留位							PFPUn
								RW 0

位	域	描述
[0]	PFPUn	GPIO 端口 F 引脚 n 的上拉选择控制位 (n = 0) 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 F 下拉选择寄存器 – PFPDR

该寄存器用来使能或除能 GPIO 端口 F 的下拉功能。

偏移量： 0x00C

复位值： 0x0000_0000

类型 / 复位	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							PFPD
								RW 0

位	域	描述
[0]	PFPDn	GPIO 端口 F 引脚 n 的下拉选择控制位 (n = 0) 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 F 漏极开路选择寄存器 – PFODR

该寄存器用来使能或除能 GPIO 端口 F 的漏极开路功能。

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							PFOD
类型 / 复位								RW 0

位	域	描述
[0]	PFODn	GPIO 端口 F 引脚 n 的漏极开路选择控制位 (n = 0) 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) 注: 当漏极开路功能使能时, 引脚 n 内部上拉或下拉设置将无效。

端口 F 驱动电流选择寄存器 – PFDRVR

该寄存器用来定义 GPIO 端口 F 的输出驱动电流。

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
	保留位							
类型 / 复位								
	7	6	5	4	3	2	1	0
	保留位						PFDV0	
类型 / 复位							RW	0 RW 0

位	域	描述
[1:0]	PFDVn[1:0]	GPIO 端口 F 引脚 n 的输出电流驱动选择控制位 (n = 0) 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 F 锁定寄存器 – PFLOCKR

该寄存器定义了 GPIO 端口 F 的锁定配置。

偏移量: 0x018

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
	PFLKEY								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	23	22	21	20	19	18	17	16	
	PFLKEY								
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	
	保留位							PFLOCK	
类型 / 复位								RW	0

位	域	描述
[31:16]	PFLKEY	GPIO 端口 F 锁定键 0x5FA0: 使能端口 F 锁定功能 其它: 除能端口 F 锁定功能 为了锁定端口 F 功能, 在该寄存器 PFLKEY 位域写入值 0x5FA0。为了在这个锁定寄存器执行一个成功的写操作, 写入 PFLKEY 位域的值必须为 0x5FA0。如果写入这个位域值不等于 0x5FA0, PFLOCKR 寄存器上的任何写操作将被中止。对 PFLKEY 位域的一个读操作的结果将返回 GPIO 端口 F 锁定状态, 表示 GPIO 端口 F 是否被锁定。如果读值, PFLKEY 位域为 0, 表明 GPIO 端口 F 锁定功能除能。否则, 它表明 GPIO 端口 F 锁定功能使能, 读值等于 1。
[0]	PFLOCKn	GPIO 端口 F 引脚 n 的锁定控制位 (n = 0) 0: 端口 F 引脚 n 未锁定 1: 端口 F 引脚 n 锁定 当正确的锁定键应用于 PFLKEY 位域, PFLOCKn 位用来锁定相应 GPIO 引脚的配置。被锁定的配置包括 PFDIRn、PFINENn、PFPUn、PFPDn、PFODn 和 PFDVn 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 GPF CFGHR 或 GPF CFGLR 寄存器也将被锁定。请注意, PFLOCKR 寄存器只能写入一次, 这意味着 PFLKEY 和 PFLOCKn (锁定控制位) 应一起写入, 不能改变直到系统复位或 GPIO 端口 F 发生复位。

端口 F 数据输入寄存器 – PFDINR

该寄存器指定 GPIO 端口 F 的输入数据。

偏移量： 0x01C

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							PFDIN	
类型 / 复位								RO	0

位	域	描述
[0]	PFDINn	GPIO 端口 F 引脚 n 的数据输入位 (n = 0) 0: 引脚 n 的输入数据为 0 1: 引脚 n 的输入数据为 1

端口 F 输出数据寄存器 – PFDOUTR

该寄存器指定 GPIO 端口 F 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	保留位							
	7	6	5	4	3	2	1	0
类型 / 复位	保留位							PFDOUT
								RW 0

位	域	描述
[0]	PFDOUTn	GPIO 端口 F 引脚 n 的数据输出位 (n = 0) 0: 引脚 n 的输出数据为 0 1: 引脚 n 的输出数据为 1

端口 F 输出置位 / 复位控制寄存器 – PFSRR

该寄存器用来置位或复位 GPIO 端口 F 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
	保留位							
类型 / 复位								
	23	22	21	20	19	18	17	16
	保留位							PFRST
类型 / 复位								WO 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							PFSET
类型 / 复位								WO 0

位	域	描述
[16]	PFRSTn	GPIO 端口 F 引脚 n 的输出复位控制位 (n = 0) 0: 不影响 PFDOUTn 位 1: 复位 PFDOUTn 位 注意, 当该寄存器中的 PFRSTn 位或 PFRR 寄存器中的 PFRSTn 位被置位, 都会使 PFDOUTn 位复位。
[0]	PFSETn	GPIO 端口 F 引脚 n 的输出置位控制位 (n = 0) 0: 不影响 PFDOUTn 位 1: 置位 PFDOUTn 位 注意, 如果 PFSETn 和 PFRSTn 位同时被置位, PFSETn 位使能的功能有比较高的优先级。

端口 F 输出复位寄存器 – PFRR

该寄存器用来复位 GPIO 端口 F 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	保留位							PFRST
								WO 0

位	域	描述
[0]	PFRSTn	GPIO 端口 F 引脚 n 的输出复位控制位 (n = 0) 0: 不影响 PFDOUTn 位 1: 复位 PFDOUTn 位

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 GPxCFGLR 或 GPxCFGHR 寄存器 (其中 x 为不同引脚的名称) 配置为多达 16 个不同的功能, 如 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] 位域, 可以选择任意的 GPIO 引脚作为 EXTI 中断引脚来触发一个中断或事件。更多详细信息请参阅 EXTI 部分。

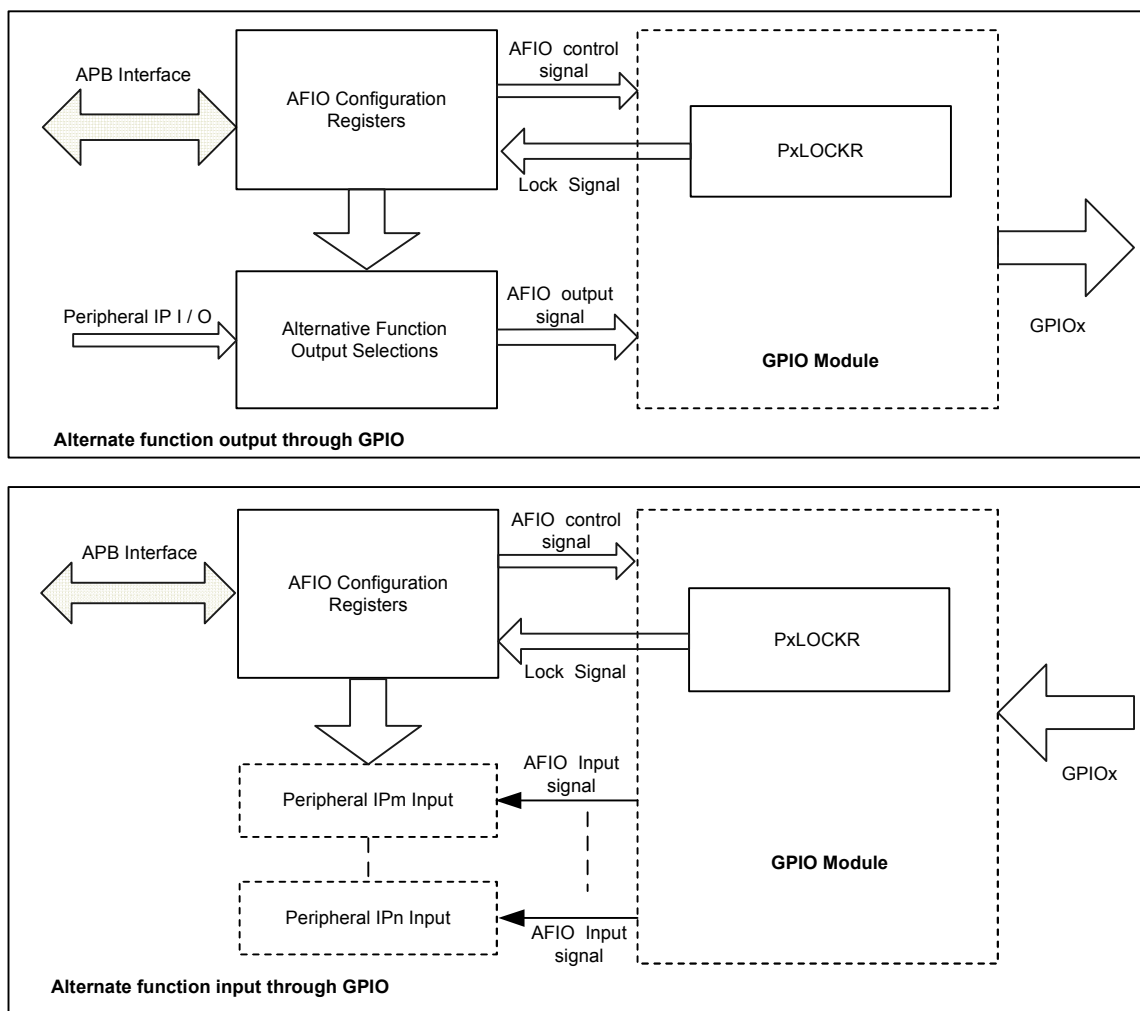


图 23. AFIO 方框图

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 引脚如附图所示。例如，用户可以设置 ESSR0 寄存器中 EXTI0PIN[3:0] 位域为 b0000 来选择 GPIO PA0 引脚作为 EXTI 0 输入。由于并非所有端口 A ~ D、F 的引脚在所有封装类型都是可用的，详细的引脚信息请参阅数据手册 (Datasheet) 中的引脚图。当相应的引脚不可用时，EXTInPIN[3:0] 位域的设置是无效的。

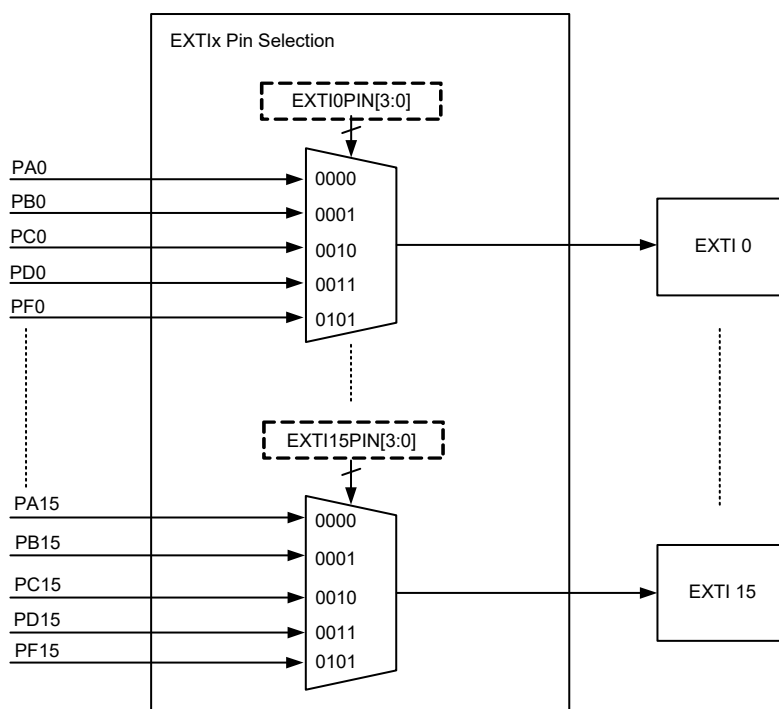


图 24. EXTI 输入通道选择

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 (n = 0 ~ 15, x = A ~ D、F) 寄存器中 PxCFGn[3:0] 位域选择。如果选择的引脚复用功能不可用，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) 中复用功能列表说明部分。除了这种机制外，每个外设可选功能分别分配到不同的引脚，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] 位域的设置。

- 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 24. 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	USART /UART	I ² C	SCI	EBI	N/A	N/A	N/A	SCTM/PWM	N/A	系统其它

锁定机制

该单片机具有锁定功能，使用 GPIO 锁定寄存器 PxLOCKR (x = A ~ D、F)，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 25. 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 AFIO 配置低寄存器	0x0000_0000
GPACFGHR	0x024	GPIO 端口 A AFIO 配置高寄存器	0x0000_0000
GPBCFGLR	0x028	GPIO 端口 B AFIO 配置低寄存器	0x0000_0000
GPBCFGHR	0x02C	GPIO 端口 B AFIO 配置高寄存器	0x0000_0000
GPCCFGLR	0x030	GPIO 端口 C AFIO 配置低寄存器	0x0000_0000
GPCCFGHR	0x034	GPIO 端口 C AFIO 配置高寄存器	0x0000_0000
GPDCFGLR	0x038	GPIO 端口 D AFIO 配置低寄存器	0x0000_0000
GPDCFGHR	0x03C	GPIO 端口 D AFIO 配置高寄存器	0x0000_0000
GPFCFGLR	0x048	GPIO 端口 F AFIO 配置低寄存器	0x0000_0000
GPFCFGHR	0x04C	GPIO 端口 F AFIO 配置高寄存器	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
	23	22	21	20	19	18	17	16
	EXTI5PIN				EXTI4PIN			
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	15	14	13	12	11	10	9	8
	EXTI3PIN				EXTI2PIN			
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0
	EXTI1PIN				EXTI0PIN			
类型 / 复位	RW	0	RW	0	RW	0	RW	0

位	域	描述
[31:0]	EXTInPIN[3:0]	EXTIn 引脚选择 (n = 0 ~ 7) 0000: PA 位 n 被选作 EXTIn 来源信号 0001: PB 位 n 被选作 EXTIn 来源信号 0010: PC 位 n 被选作 EXTIn 来源信号 0011: PD 位 n 被选作 EXTIn 来源信号 0101: PF 位 n 被选作 EXTIn 来源信号 其它: 保留位 注: 由于并非所有端口的引脚在所有产品和封装类型都是可用的, 详细的引脚信息请参阅数据手册 (Datasheet) 中的引脚图。当相应的引脚不可用时, EXTIInPIN[3:0] 位域的设置是无效的。

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
	23	22	21	20	19	18	17	16
	EXTI13PIN				EXTI12PIN			
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	15	14	13	12	11	10	9	8
	EXTI11PIN				EXTI10PIN			
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0
	EXTI9PIN				EXTI8PIN			
类型 / 复位	RW	0	RW	0	RW	0	RW	0

位	域	描述
[31:0]	EXTInPIN[3:0]	EXTIn 引脚选择 (n = 8 ~ 15) 0000: PA 位 n 被选作 EXTIn 来源信号 0001: PB 位 n 被选作 EXTIn 来源信号 0010: PC 位 n 被选作 EXTIn 来源信号 0011: PD 位 n 被选作 EXTIn 来源信号 0101: PF 位 n 被选作 EXTIn 来源信号 其它: 保留位 注: 由于并非所有端口的引脚在所有产品和封装类型都是可用的, 详细的引脚信息请参阅数据手册 (Datasheet) 中的引脚图。当相应的引脚不可用时, EXTIInPIN[3:0] 位域的设置是无效的。

GPIO 端口 x 配置低寄存器 – GPxCFGnLR, x = A, B, C, D, F

该寄存器定义了 GPIO 端口 x (x = A, B, C, D, F) 的复用功能。

偏移量: 0x020, 0x028, 0x030, 0x038, 0x048

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
	23	22	21	20	19	18	17	16
	PxCFG5				PxCFG4			
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	15	14	13	12	11	10	9	8
	PxCFG3				PxCFG2			
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0
	PxCFG1				PxCFG0			
类型 / 复位	RW	0	RW	0	RW	0	RW	0

位	域	描述
[31:0]	PxCFGn[3:0]	端口 x 引脚 n 复用功能选择 (n = 0 ~ 7) 0000: 端口 x 引脚 n 选择作为 AF0 0001: 端口 x 引脚 n 选择作为 AF1 ... 1110: 端口 x 引脚 n 选择作为 AF14 1111: 端口 x 引脚 n 选择作为 AF15 请参考单片机数据手册 (Datasheet) 中的复用功能列表获取复用功能的更多信息。

GPIO 端口 x 配置高寄存器 – GPxCFGHR, x = A, B, C, D, F

该寄存器定义了 GPIO 端口 x (x = A, B, C, D, F) 的复用功能。

偏移量: 0x024, 0x02C, 0x034, 0x03C, 0x04C

复位值: 0x0000_0000

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

位	域	描述
[31:0]	PxCFGn[3:0]	端口 x 引脚 n 复用功能选择 (n = 8 ~ 15) 0000: 端口 x 引脚 n 选择作为 AF0 0001: 端口 x 引脚 n 选择作为 AF1 ... 1110: 端口 x 引脚 n 选择作为 AF14 1111: 端口 x 引脚 n 选择作为 AF15 请参考单片机数据手册 (Datasheet) 中的复用功能列表获取复用功能的更多信息。

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®-M3 提供了紧密耦合集成部分, 即嵌套向量中断控制器 (NVIC)。NVIC 控制系统异常及包括如使能 / 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功能的外设中断。更多信息请参考“Cortex®-M3 技术参考手册”。

此外, Cortex®-M3 内置简单的 24-bit 向下计数定时器 (SysTick) 用来作为实时定时器操作系统 (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 26. 异常类型

异常类型	优先级	中断号	异常号	向量地址	描述
—	—	—	0	0x000	初始堆栈指针值
复位	-3 (最高)	—	1	0x004	复位
NMI	-2	—	2	0x008	不可屏蔽中断。时钟故障中断信号 (时钟控制单元提供时钟监控功能) 连接到 NMI 输入
硬件故障	-1	—	3	0x00C	所有故障类型
存储器管理	可配置 ⁽¹⁾	—	4	0x010	存储器保护单元 (MPU) 不匹配, 包括访问冲突和不匹配
总线故障	可配置 ⁽¹⁾	—	5	0x014	预抓取故障, 存储器访问故障和其它地址 / 存储器相关的故障
使用故障	可配置 ⁽¹⁾	—	6	0x018	使用故障, 例如执行未定义的指令或试图进行非法的状态转变
—	—	—	7	0x01C	保留
—	—	—	8	0x020	保留
—	—	—	9	0x024	保留
—	—	—	10	0x028	保留
SVCCall	可配置 ⁽¹⁾	—	11	0x02C	SVC 指令的系统服务呼叫
调试监控	可配置 ⁽¹⁾	—	12	0x030	未处于 halt 时, 为调试监控器
—	—	—	13	0x034	保留
PendSV	可配置 ⁽¹⁾	—	14	0x038	系统服务的可挂起请求
SysTick	可配置 ⁽¹⁾	—	15	0x03C	SysTick 定时器递减为零
CKRDY	可配置 ⁽²⁾	0	16	0x040	时钟就绪中断 (HSE、HIS、LSE、LSI 或 PLL)
LVD	可配置 ⁽²⁾	1	17	0x044	低压检测中断
BOD	可配置 ⁽²⁾	2	18	0x048	掉电检测中断
—	—	3	19	0x04C	保留
RTC	可配置 ⁽²⁾	4	20	0x050	RTC 总中断
FMC	可配置 ⁽²⁾	5	21	0x054	FMC 总中断
EVWUP	可配置 ⁽²⁾	6	22	0x058	EXTI 事件唤醒中断
LPWUP	可配置 ⁽²⁾	7	23	0x05C	WAKEUP 引脚中断
EXTI0	可配置 ⁽²⁾	8	24	0x060	EXTI 引脚 0 中断
EXTI1	可配置 ⁽²⁾	9	25	0x064	EXTI 引脚 1 中断

异常类型	优先级	中断号	异常号	向量地址	描述
EXTI2	可配置 ⁽²⁾	10	26	0x068	EXTI 引脚 2 中断
EXTI3	可配置 ⁽²⁾	11	27	0x06C	EXTI 引脚 3 中断
EXTI4	可配置 ⁽²⁾	12	28	0x070	EXTI 引脚 4 中断
EXTI5	可配置 ⁽²⁾	13	29	0x074	EXTI 引脚 5 中断
EXTI6	可配置 ⁽²⁾	14	30	0x078	EXTI 引脚 6 中断
EXTI7	可配置 ⁽²⁾	15	31	0x07C	EXTI 引脚 7 中断
EXTI8	可配置 ⁽²⁾	16	32	0x080	EXTI 引脚 8 中断
EXTI9	可配置 ⁽²⁾	17	33	0x084	EXTI 引脚 9 中断
EXTI10	可配置 ⁽²⁾	18	34	0x088	EXTI 引脚 10 中断
EXTI11	可配置 ⁽²⁾	19	35	0x08C	EXTI 引脚 11 中断
EXTI12	可配置 ⁽²⁾	20	36	0x090	EXTI 引脚 12 中断
EXTI13	可配置 ⁽²⁾	21	37	0x094	EXTI 引脚 13 中断
EXTI14	可配置 ⁽²⁾	22	38	0x098	EXTI 引脚 14 中断
EXTI15	可配置 ⁽²⁾	23	39	0x09C	EXTI 引脚 15 中断
—	—	24	40	0x0A0	保留
ADC	可配置 ⁽²⁾	25	41	0x0A4	A/D 转换器总中断
—	—	26	42	0x0A8	保留
—	—	27	43	0x0AC	保留
—	—	28	44	0x0B0	保留
—	—	29	45	0x0B4	保留
—	—	30	46	0x0B8	保留
—	—	31	47	0x0BC	保留
—	—	32	48	0x0C0	保留
—	—	33	49	0x0C4	保留
—	—	34	50	0x0C8	保留
GPTM	可配置 ⁽²⁾	35	51	0x0CC	GPTM 总中断
—	—	36	51	0x0CC	保留
SCTM0	可配置 ⁽²⁾	37	52	0x0D0	SCTM0 总中断
SCTM1	可配置 ⁽²⁾	38	53	0x0D4	SCTM1 总中断
PWM	可配置 ⁽²⁾	39	54	0x0D8	PWM 总中断
—	—	40	56	0x0E0	保留
BFTM0	可配置 ⁽²⁾	41	57	0x0E4	BFTM0 总中断
BFTM1	可配置 ⁽²⁾	42	58	0x0E8	BFTM1 总中断
I ² C0	可配置 ⁽²⁾	43	59	0x0EC	I2C0 总中断
I ² C1	可配置 ⁽²⁾	44	60	0x0F0	I2C1 总中断
SPI0	可配置 ⁽²⁾	45	61	0x0F4	SPI0 总中断
SPI1	可配置 ⁽²⁾	46	62	0x0F8	SPI1 总中断
USART	可配置 ⁽²⁾	47	63	0x0FC	USART 总中断
—	—	48	64	0x100	保留
UART0	可配置 ⁽²⁾	49	65	0x104	UART0 总中断
UART1	可配置 ⁽²⁾	50	66	0x108	UART1 总中断
SCI	可配置 ⁽²⁾	51	67	0x10C	SCI0 & SCI1 总中断
—	—	52	68	0x110	保留
USB	可配置 ⁽²⁾	53	69	0x114	USB 总中断
—	—	54	70	0x118	保留

异常类型	优先级	中断号	异常号	向量地址	描述
PDMA_CH0	可配置 ⁽²⁾	55	71	0x11C	PDMA 通道 0 总中断
PDMA_CH1	可配置 ⁽²⁾	56	72	0x120	PDMA 通道 1 总中断
PDMA_CH2	可配置 ⁽²⁾	57	73	0x124	PDMA 通道 2 总中断
PDMA_CH3	可配置 ⁽²⁾	58	74	0x128	PDMA 通道 3 总中断
PDMA_CH4	可配置 ⁽²⁾	59	75	0x12C	PDMA 通道 4 总中断
PDMA_CH5	可配置 ⁽²⁾	60	76	0x130	PDMA 通道 5 总中断
—	—	61	77	0x134	保留
—	—	62	78	0x138	保留
—	—	63	79	0x13C	保留
—	—	64	80	0x140	保留
—	—	65	81	0x144	保留
—	—	66	82	0x148	保留
—	—	67	83	0x14C	保留
EBI	可配置 ⁽²⁾	68	84	0x150	EBI 总中断
AES	可配置 ⁽²⁾	69	85	0x154	AES 总中断

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®-M3 技术参考手册” 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®-M3 技术参考手册” 文件。

特性

- 16 个 Cortex®-M3 系统异常
- 多达 64 个可屏蔽的外设中断
- 16 个可编程优先级 (4 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，产生可屏蔽的系统中断
 - HCLK 或 AHB 时钟除以 8 作为 SysTick 的参考时钟源

功能描述

SysTick 校准

NVIC 提供 SysTick 校准值寄存器 SCALIB, 给 RTOS 节拍定时器一个 1 ms 参考时基或用作其它用途。当时钟源来自带有一个频率为 10 MHz (80 MHz 除以 8) 的 SysTick 参考输入时钟 STCLK, 在 SCALIB 寄存器 TENMS 位域有一个固定值 10000, 这是计数器重载值用来表示 1 ms。

寄存器列表

下表显示 NVIC 寄存器和复位值。

表 27. NVIC 寄存器列表

寄存器	偏移量	描述	复位值
NVIC 基址 = 0xE000_E000			
ICTR	0x004	中断控制类型寄存器	0x0000_0001
SCTRL	0x010	SysTick 控制和状态寄存器	0x0000_0000
SLOAD	0x014	SysTick 重载值寄存器	不可预知
SVAL	0x018	SysTick 电流值寄存器	不可预知
SCALIB	0x01C	SysTick 校准值寄存器	0x4000_2710
ISER0_31	0x100	Irq 0 ~ 31 置位使能寄存器	0x0000_0000
ISER32_63	0x104	Irq 32 ~ 63 置位使能寄存器	0x0000_0000
ISER64_95	0x108	Irq 64 ~ 95 置位使能寄存器	0x0000_0000
ICER0_31	0x180	Irq 0 ~ 31 清除使能寄存器	0x0000_0000
ICER32_63	0x184	Irq 32 ~ 63 清除使能寄存器	0x0000_0000
ICER64_95	0x188	Irq 64 ~ 95 清除使能寄存器	0x0000_0000
ISPR0_31	0x200	Irq 0 ~ 31 置位挂起寄存器	0x0000_0000
ISPR32_63	0x204	Irq 32 ~ 63 置位挂起寄存器	0x0000_0000
ISPR64_95	0x208	Irq 64 ~ 95 置位挂起寄存器	0x0000_0000
ICPR0_31	0x280	Irq 0 ~ 31 清除挂起寄存器	0x0000_0000
ICPR32_63	0x284	Irq 32 ~ 63 清除挂起寄存器	0x0000_0000
ICPR64_95	0x288	Irq 64 ~ 95 清除挂起寄存器	0x0000_0000
IABR0_31	0x300	Irq 0 ~ 31 有效位寄存器	0x0000_0000
IABR32_63	0x304	Irq 32 ~ 63 有效位寄存器	0x0000_0000
IABR64_95	0x308	Irq 64 ~ 95 有效位寄存器	0x0000_0000
IRQ0_3	0x400	Irq 0 ~ 3 优先级寄存器	0x0000_0000
IRQ4_7	0x404	Irq 4 ~ 7 优先级寄存器	0x0000_0000
IRQ8_11	0x408	Irq 8 ~ 11 优先级寄存器	0x0000_0000
IRQ12_15	0x40C	Irq 12 ~ 15 优先级寄存器	0x0000_0000
IRQ16_19	0x410	Irq 16 ~ 19 优先级寄存器	0x0000_0000
IRQ20_23	0x414	Irq 20 ~ 23 优先级寄存器	0x0000_0000
IRQ24_27	0x418	Irq 24 ~ 27 优先级寄存器	0x0000_0000
IRQ28_31	0x41C	Irq 28 ~ 31 优先级寄存器	0x0000_0000
IRQ32_35	0x420	Irq 32 ~ 35 优先级寄存器	0x0000_0000
IRQ36_39	0x424	Irq 36 ~ 39 优先级寄存器	0x0000_0000
IRQ40_43	0x428	Irq 40 ~ 43 优先级寄存器	0x0000_0000
IRQ44_47	0x42C	Irq 44 ~ 47 优先级寄存器	0x0000_0000

寄存器	偏移量	描述	复位值
IRQ48_51	0x430	Irq 48 ~ 51 优先级寄存器	0x0000_0000
IRQ52_55	0x434	Irq 52 ~ 55 优先级寄存器	0x0000_0000
IRQ56_59	0x438	Irq 56 ~ 59 优先级寄存器	0x0000_0000
IRQ60_63	0x43C	Irq 60 ~ 63 优先级寄存器	0x0000_0000
IRQ64_67	0x440	Irq 64 ~ 67 优先级寄存器	0x0000_0000
ICSR	0xD04	中断控制状态寄存	0x0000_0000
VTOR	0xD08	向量表偏移寄存器	0x0000_0000
AIRCR	0xD0C	应用中断 / 复位控制寄存器	0xFA05_0000
SCR	0xD10	系统控制寄存器	0x0000_0000
CCR	0xD14	配置控制寄存器	0x0000_0000
SHPR4_7	0xD18	系统处理程序 4 ~ 7 优先级寄存器	0x0000_0000
SHPR8_11	0xD1C	系统处理程序 8 ~ 11 优先级寄存器	0x0000_0000
SHPR12_15	0xD20	系统处理程序 12 ~ 15 优先级寄存器	0x0000_0000
SHCSR	0xD24	系统处理程序的控制和状态寄存器	0x0000_0000
CFSR	0xD28	可配置的故障状态寄存器	0x0000_0000
HFSR	0xD2C	硬件故障状态寄存器	0x0000_0000
DFSR	0xD30	调试故障状态寄存器	0x0000_0000
MMFAR	0xD34	分配信息管理地址寄存器	不可预知
BFAR	0xD38	总线故障地址寄存器	不可预知
AFSR	0xD3C	辅助异常状态寄存器	0x0000_0000
STIR	0xF00	软件触发中断寄存器	0x0000_0000

注：更多以上寄存器的详细描述，请参阅 Arm 文档“Cortex®-M3 技术参考手册”。

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 个边沿检测器, 都可以分别产生唤醒事件或独立的中断请求。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 EXTICFGRn (n = 0 ~ 15) 寄存器中 SRCnTYPE 位域选择。在唤醒事件模式, 唤醒事件的极性可以通过 EXTIWAKUPPOLR 寄存器中 EXTInWPOL (n = 0 ~ 15) 位域设置。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。

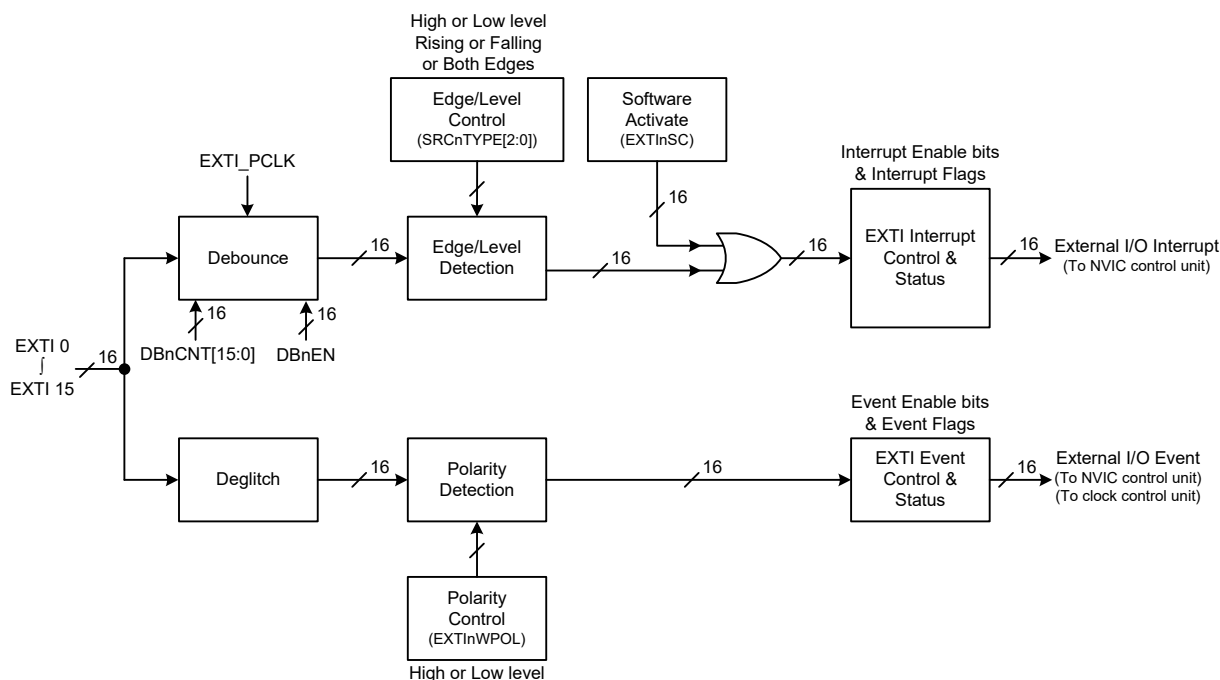


图 25. EXTI 方框图

特性

- 多达 16 个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, 并把它送到 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、WAKEUP 输入引脚、USB 和 RTC 唤醒功能。通过在相应的外设内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 控制器将确立中断。

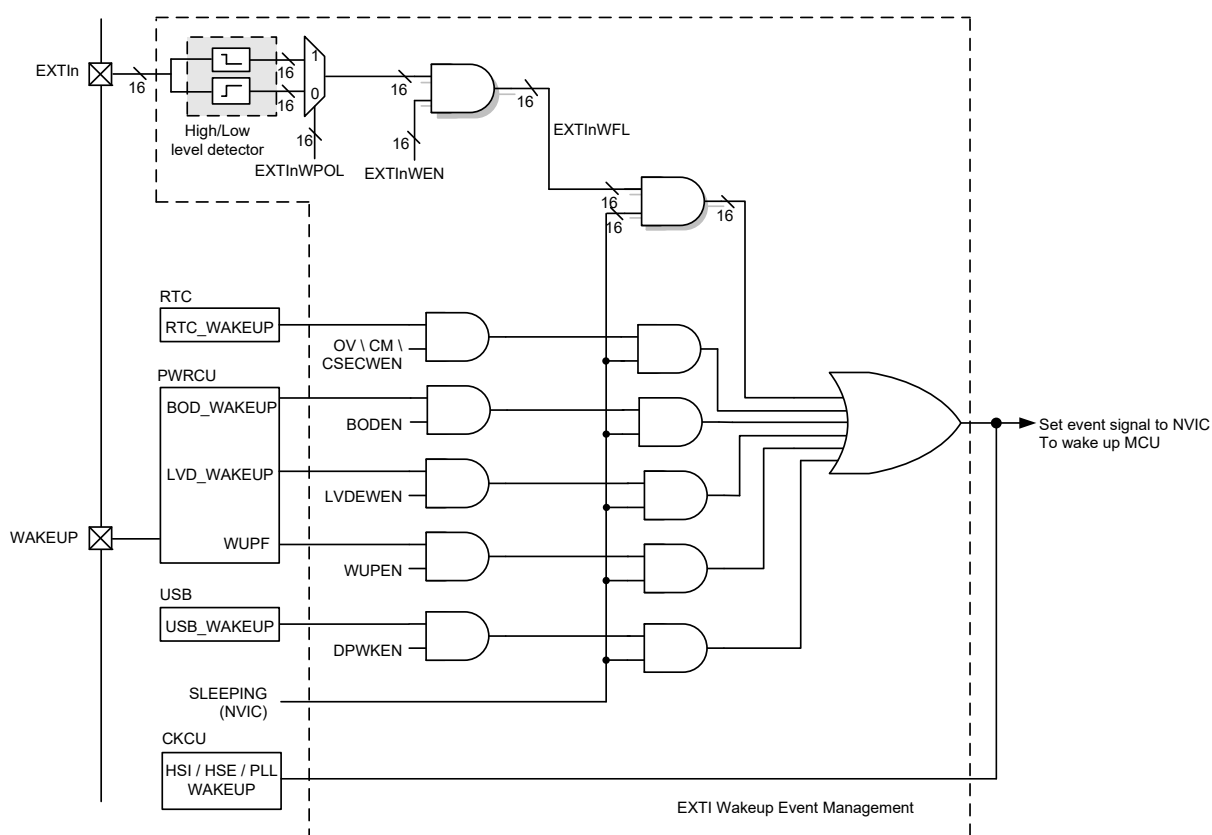


图 26. EXTI 唤醒事件管理

外部中断 / 事件线配置

所有 GPIO 引脚通过配置 AFIO_ESSRn (n = 0 ~ 1) 寄存器 EXTInPIN[3:0] 位域选择哪个引脚作为 EXTI 触发源来触发中断或事件。更多信息请参考 AFIO 部分。

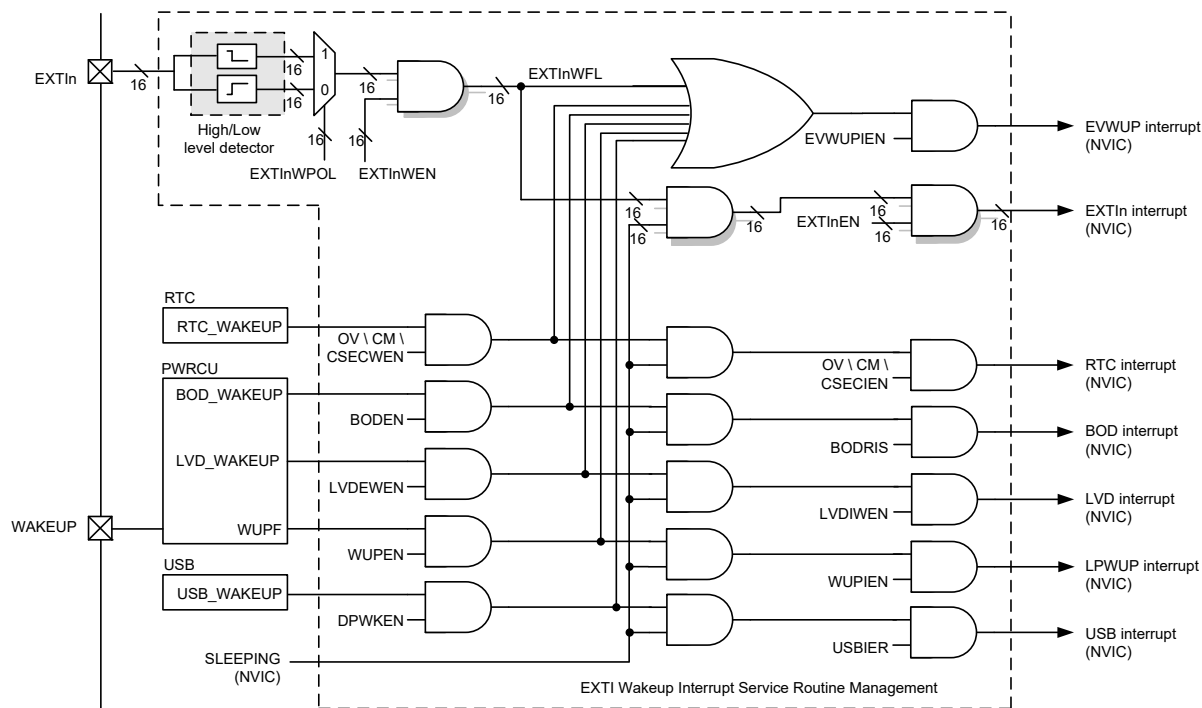


图 27. 唤醒中断服务子程序

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 位域, 以便为特定应用选择一个合适的去抖时间。中断信号也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 唤醒标志位产生中断请求。单片机被唤醒且时钟已经恢复后, 被 EXTI 引脚触发的 EXTI 唤醒标志位必须读出, 然后用应用程序清零。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。

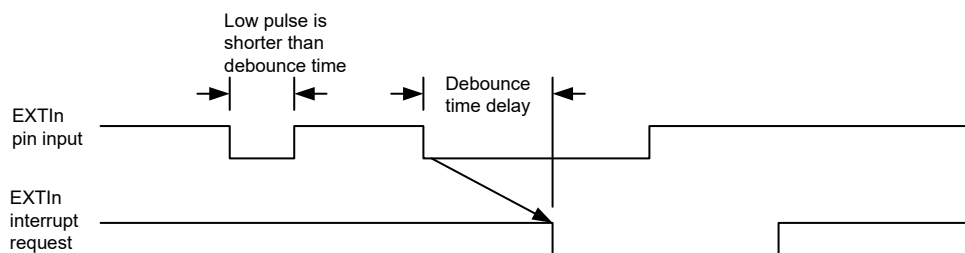


图 28. EXTI 中断去抖功能

寄存器列表

下表显示 EXTI 寄存器和复位值。

表 28. 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
EXTIEDGEFLGR	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 中断软件设置命令寄存器	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 n 配置寄存器 – 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	域	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><tr><th colspan="3">SRCnTYPE[2:0]</th><th>中断源类型</th></tr><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></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 时钟 (EXTI_PCLK) 周期计算，并应足够长来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

	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
	EXTI15EN	EXTI14EN	EXTI13EN	EXTI12EN	EXTI11EN	EXTI10EN	EXTI9EN	EXTI8EN
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0
	EXTI7EN	EXTI6EN	EXTI5EN	EXTI4EN	EXTI3EN	EXTI2EN	EXTI1EN	EXTI0EN
类型 / 复位	RW	0	RW	0	RW	0	RW	0

位	域	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 0	WC 0	WC 0	WC 0
	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 0	WC 0	WC 0	WC 0

位	域	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 引脚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

	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	EXTI15EDS	EXTI14EDS	EXTI13EDS	EXTI12EDS	EXTI11EDS	EXTI10EDS	EXTI9EDS	EXTI8EDS
	WC 0	WC 0	WC 0	WC 0	WC 0	WC 0	WC 0	WC 0
	7	6	5	4	3	2	1	0
类型 / 复位	EXTI7EDS	EXTI6EDS	EXTI5EDS	EXTI4EDS	EXTI3EDS	EXTI2EDS	EXTI1EDS	EXTI0EDS
	WC 0	WC 0	WC 0	WC 0	WC 0	WC 0	WC 0	WC 0

位	域	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 中断软件设置命令寄存器 – EXTISSCR

该寄存器用来激活 EXTI 中断。

偏移量： 0x04C
复位值： 0x0000_0000

	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
	EXTI15SC	EXTI14SC	EXTI13SC	EXTI12SC	EXTI11SC	EXTI10SC	EXTI9SC	EXTI8SC
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0
	EXTI7SC	EXTI6SC	EXTI5SC	EXTI4SC	EXTI3SC	EXTI2SC	EXTI1SC	EXTI0SC
类型 / 复位	RW	0	RW	0	RW	0	RW	0

位	域	描述
[15:0]	EXTInSC	EXTIn 软件设置命令 (n = 0 ~ 15) 0: 停用相关 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
	EVWUPIEN	保留位						
类型 / 复位	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
	EXTI15WEN	EXTI14WEN	EXTI13WEN	EXTI12WEN	EXTI11WEN	EXTI10WEN	EXTI9WEN	EXTI8WEN
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0
	EXTI7WEN	EXTI6WEN	EXTI5WEN	EXTI4WEN	EXTI3WEN	EXTI2WEN	EXTI1WEN	EXTI0WEN
类型 / 复位	RW	0	RW	0	RW	0	RW	0

位	域	描述
[31]	EVWUPIEN	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 引脚中断唤醒极性。

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
	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

位	域	描述
[15:0]	EXTInWPOL	EXTIn 唤醒极性 (n = 0 ~ 15) 0: EXTIn 高电平时唤醒 1: EXTIn 低电平时唤醒

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
	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
	EXTI15WFL	EXTI14WFL	EXTI13WFL	EXTI12WFL	EXTI11WFL	EXTI10WFL	EXTI9WFL	EXTI8WFL
类型 / 复位	WC	0	WC	0	WC	0	WC	0
	7	6	5	4	3	2	1	0
	EXTI7WFL	EXTI6WFL	EXTI5WFL	EXTI4WFL	EXTI3WFL	EXTI2WFL	EXTI1WFL	EXTI0WFL
类型 / 复位	WC	0	WC	0	WC	0	WC	0

位	域	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

单片机内建一个带参考电压发生器 (V_{REF}) 的 12-bit 多通道 A/D 转换器, 总共有 11 个复用通道, 包括 8 个可供外部模拟信号输入的外部通道和 3 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指定的阈值范围。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数据。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。

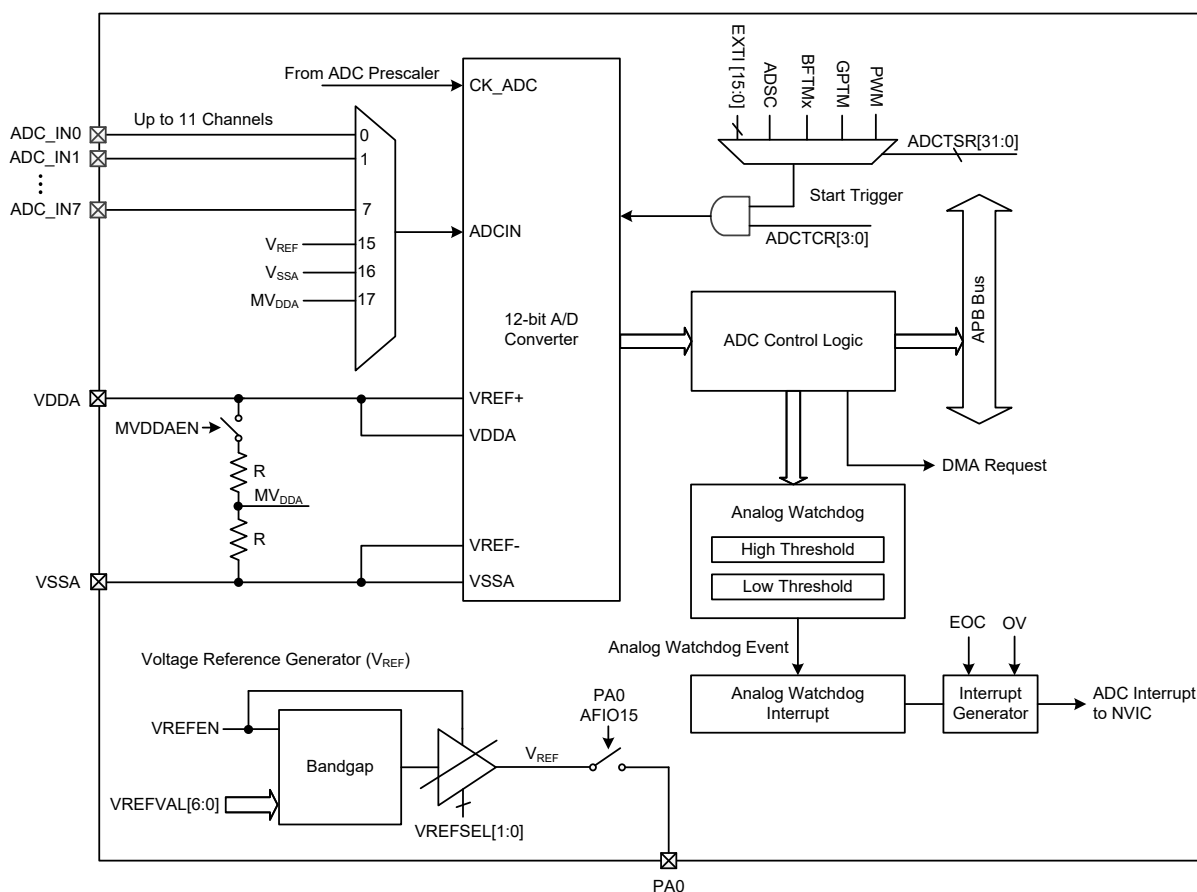


图 29. 带参考电压 V_{REF} 的 ADC 方框图

特性

- 12-bit SAR ADC 内核
- 高达 1 Msps 转换速率
- 多达 8 个外部模拟输入通道
- 1 个通道用于内部参考电压 V_{REF}
- 1 个通道用于监测外部引脚输入电源电压 V_{DDA}
- 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 – 外部事件输入
 - GPTM 触发
 - PWM 触发
 - BFTM0 / BFTM1 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
- A/D 转换结束产生 PDMA 请求

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 (ADCEN bit = 0) 情况下相互切换。

通道选择

A/D 转换器支持 11 个多路复用通道, 转换结果依组转换次序存放到对应数据寄存器中。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 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 转换模式位域的 ADMODE[1:0] 设置为 0x0, A/D 转换器将运行在单次转换模式。启动单次转换的触发源可来自软件触发, 外部 EXTI 事件或者 TM 事件, 通过触发控制寄存器 ADCTCR 和触发源寄存器 ADCTSR 设置。

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。

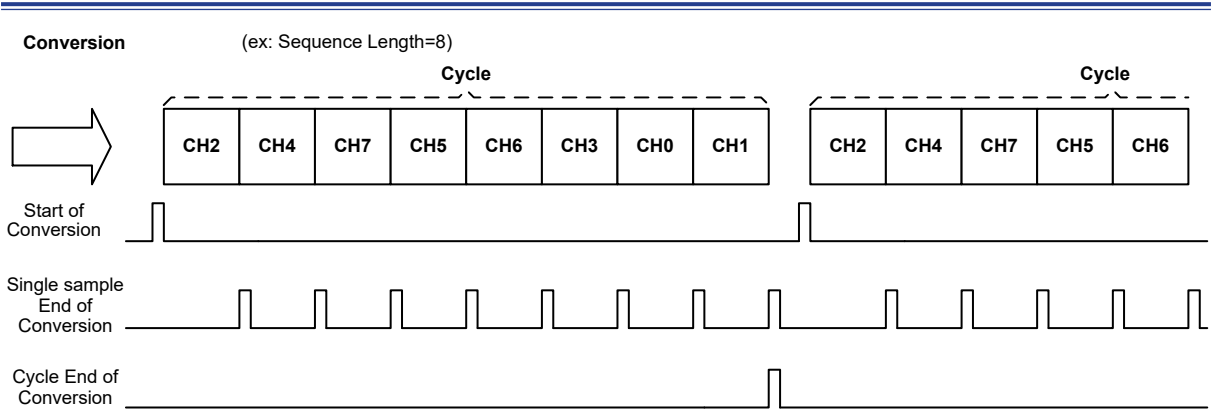


图 30. 单次转换模式

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 转换模式位域的 ADMODE[1:0] 设置为 0x2, A/D 转换器将运行在连续转换模式。启动连续转换的触发源可来自软件触发, 外部 EXTI 事件或者 TM 事件, 通过触发控制寄存器 ADCTCR 和触发源寄存器 ADCTSR 设置。

- 转换后:
- 转换后的数据将被储存在 16-bit ADCDRy (y = 0 ~ 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。

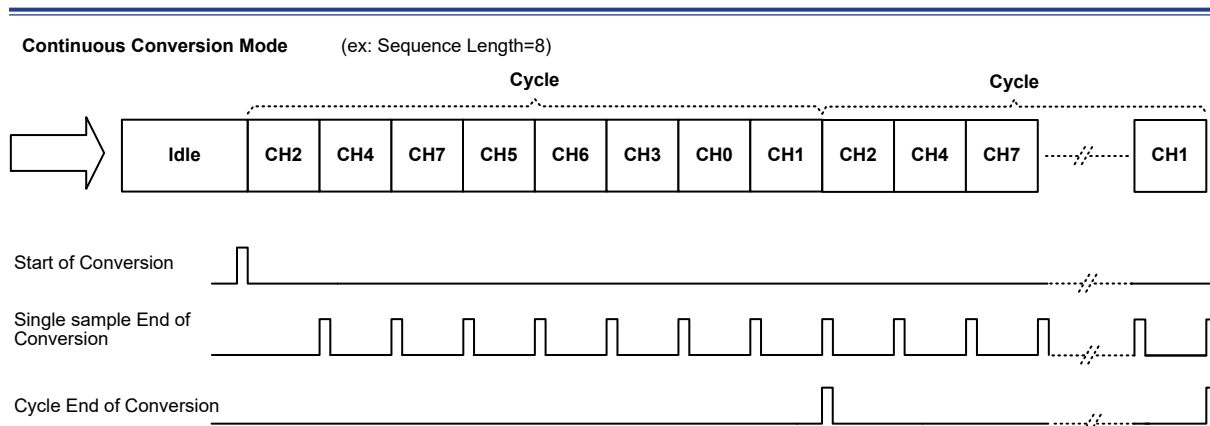


图 31. 连续转换模式

非连续转换模式

ADCCR 寄存器中 A/D 转换模式位域 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 ~ 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] 位域定义。在非连续转换模式, 启动每个子组转换的触发源可来自软件触发, 外部 EXTI 事件或者 TM 事件, 通过触发控制寄存器 ADCTCR 和触发源寄存器 ADCTSR 设置。

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 位域定义的子组长度。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。

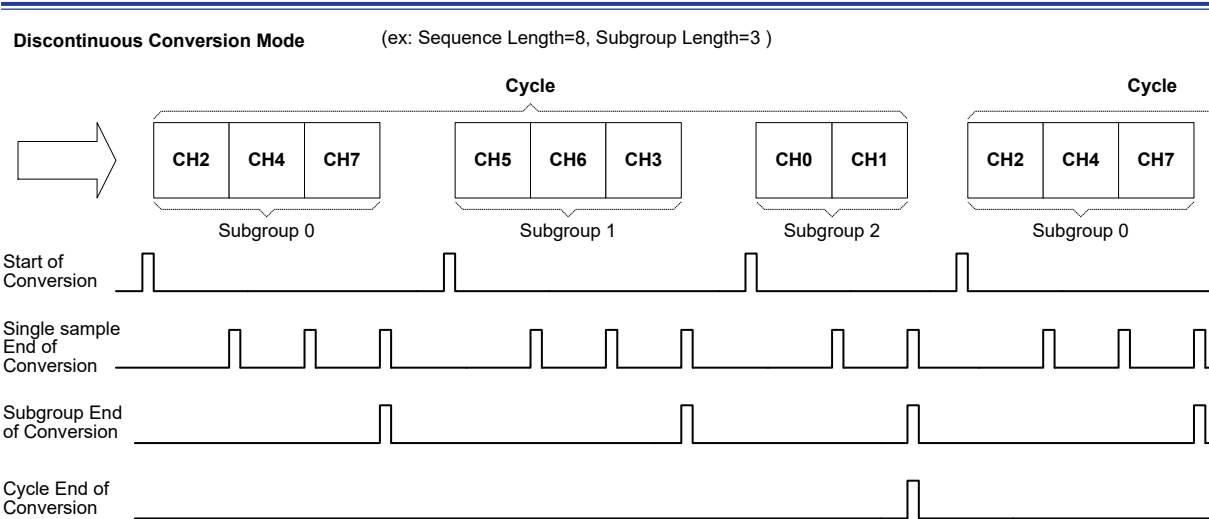


图 32. 非连续转换模式

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、脉冲宽度调制定时器 (PWM) 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 ADCTSR 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 ADCTSR 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信号后，软件触发位 ADSC 将被自动清零。

定时器事件也可以触发 A/D 开始组转换。定时器事件包括一个 PWM 或 GPTM 的主机触发输出 MTO，四个 PWM 或 GPTM 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 EXTI_n 线的外部输入。如果外部触发使能位 ADEXTI 设置为 1，并通过配置 ADCTSR 寄存器中的 ADEXTIS 位域，选定的外部 EXTI 输入线 n 有效边沿 (有效边沿由 EXTI 单元相关寄存器设置) 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：

$$T_{\text{conv}} = T_{\text{Sampling}} + T_{\text{Latency}}$$

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

例如：

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

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

数据格式

A/D 转换结果通过 ADCDRy 寄存器读取，其数据格式如下表所示。

表 29. 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，即看门狗上限阈值和看门狗下限阈值，它们分别由看门狗上限和下限阈值寄存器定义。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。若看门狗下限或上限阈值监控功能使能，当转换后的数据小于定义的下限阈值或大于定义的上限阈值时，则 ADCIRAW 寄存器中的下限阈值中断原始标志 ADIRAWL 或上限阈值中断原始标志 ADIRAWU 会被置位。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 转换中断原始位和中断状态位。这些位被软件设置为 1 后, 由硬件自动清零。

PDMA 请求

通道转换后的值将存储在对应的数据寄存器中。如果有新的转换数据存储到 ADCDRy 寄存器中, A/D 转换器可以通过 A/D 转换 EOC 中断告知单片机。用户也可以通过设置 ADCDMAR 寄存器中的 ADDMAC、ADDMAg 或 ADDMAS 位决定是否发出 PDMA 请求。若使能相关 PDMA 请求, 当上述转换完成事件发生时, PDMA 请求将自动产生。更多信息请参考 PDMA 章节以及本章节中 ADCDMAR 寄存器描述部分。

内部参考电压

内部参考电压电路可提供稳定的电压输出 V_{REF} , 供 ADC 功能使用。 V_{REF} 连接于 A/D 转换器的一个内部输入通道。制造商在生产测试过程中会对每个器件的 V_{REF} 测量校准, 并将精准的 V_{REF} 值储存在 Flash 制造权限信息块, 可通过 VREFVALR 只读寄存器进行读取。更多信息, 请参阅单片机数据手册 (Datasheet) 中相关内容。当内部参考电压未使用或者使用外部 VREF 引脚输入的电压作为参考时, 建议将内部参考电压发生电路配置为休眠模式以节省功耗。

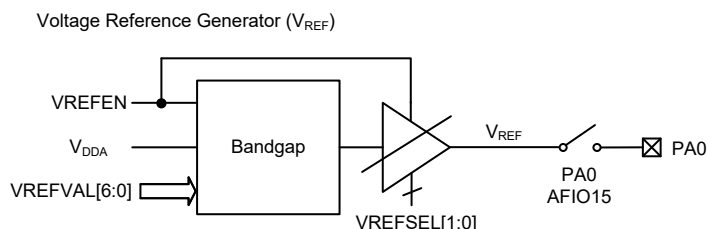


图 33. 内部参考电压方框图

V_{DDA} 电压监测

应用中若需要测量 V_{DDA} 引脚上电压值 V_{DDA} , 可通过设置 VREFCR 寄存器中的 MVDDAEN 位开启 V_{DDA} 电压测量功能。由于 V_{DDA} 电压可以高于 ADC 参考电压, 为了确保 A/D 转换器能正确测量 V_{DDA} 的值, V_{DDA} 引脚内部连接到一个分压比为 2 的桥式分压器。当置位 MVDDAEN 位时, 该桥式分压器会自动使能, 将电压 $V_{DDA}/2$ 连接到 ADC 输入通道进行转换。转换后得到 $V_{DDA}/2$ 电压对应的数字值。为避免不必要的电池耗电, 建议只有在此 A/D 转换需求时, 再使能 V_{DDA} 电压监测功能。

寄存器列表

下表显示 A/D 转换器寄存器和复位值。

表 30. 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
ADCDMAR	0x090	ADC DMA 请求寄存器	0x0000_0000
VREFCR	0x0A0	参考电压控制寄存器	0x0000_0000
VREFVALR	0x0A4	参考电压值寄存器	0x0000_00XX

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，ADC 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

类型 / 复位	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					ADSUBL		
类型 / 复位	15	14	13	12	11	10	9	8
	保留位					ADSEQL		
类型 / 复位	7	6	5	4	3	2	1	0
	ADCEN	ADCRST	保留位				ADMODE	
类型 / 复位	RW	0	RW	0			RW	0

位	域	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 位域定义。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) 的整数倍，最后一个子组将只会转换剩余的通道。
[10:8]	ADSEQL	ADC 转换长度 0x0: 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 位域指定一个转换组的整个序列长度。
[7]	ADCEN	ADC 使能 0: ADC 除能 1: ADC 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]	模式	描述
00	单次模式	启动触发后，在指定通道按顺序整体转换一次
01	保留	
10	连续模式	启动触发后，在指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，在当前的子组所有通道依序转换一次。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	域	描述
[28:24]	ADSEQ3	ADC 转换序列 NO.3 通道选择 定义 ADC 转换序列中排序 No.3 的输入通道 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 ~ 0x0E: 保留 0x0F: 内部参考电压 (V _{REF}) 0x10: 模拟地, V _{SSA} 0x11: 分压后的模拟电源电压, V _{DDA} /2 0x12 ~ 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 NO.2 通道选择
[12:8]	ADSEQ1	ADC 转换序列 NO.1 通道选择
[4:0]	ADSEQ0	ADC 转换序列 NO.0 通道选择

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	域	描述
[28:24]	ADSEQ7	ADC 转换序列 NO.7 通道选择 定义 ADC 转换序列中排序 No.7 的输入通道 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 ~ 0x0E: 保留 0x0F: 内部参考电压 (V _{REF}) 0x10: 模拟地, V _{SSA} 0x11: 分压后的模拟电源电压, V _{DDA} /2 0x12 ~ 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 NO.6 通道选择
[12:8]	ADSEQ5	ADC 转换序列 NO.5 通道选择
[4:0]	ADSEQ4	ADC 转换序列 NO.4 通道选择

ADC 输入采样时间寄存器 – ADCSTR

该寄存器定义了 A/D 转换器输入通道的采样时间。

偏移量： 0x020
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	
	ADST								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
									0

位	域	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 位域的第 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
	7	6	5	4	3	2	1	0
	ADDy							
类型 / 复位	RO	0	RO	0	RO	0	RO	0

位	域	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 位域的序列转换结果。

ADC 触发控制寄存器 – ADCTCR

该寄存器包含了 ADC 转换启动的触发使能位。

偏移量： 0x070
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				TM1	TM0	ADEXTI	ADSW
类型 / 复位					RW	0 RW	0 RW	0 RW

位	域	描述
[3]	TM1	BFTM 或 PWM 事件触发 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 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 0

位	域	描述
[29:27]	TM1E	ADC 转换的 PWM 触发事件的选择 000: PWM MTO 事件 001: PWM CH0O 事件 010: PWM CH1O 事件 011: PWM CH2O 事件 100: PWM CH3O 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 CH0O 事件 010: GPTM CH1O 事件 011: GPTM CH2O 事件 100: GPTM CH3O 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: BFTM0 001: BFTM1 010: PWM 011: 保留 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 0000: EXTI 输入线 0 0001: EXTI 输入线 1 ... 1111: EXTI 输入线 15 注意, 触发 A/D 转换开始的 EXTI 输入有效边沿是由外部中断 / 事件控制单元 EXTI 中相关寄存器决定的。

位	域	描述
[0]	ADSC	ADC 转换的软件触发位 0: 无操作 1: 立即启动转换 通过软件设置此位为高, 可手动启动 A/D 转换。转换启动后由硬件自动清零。

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
	23	22	21	20	19	18	17	16
	保留位				ADLCH			
类型 / 复位					RO	0	RO	0
	15	14	13	12	11	10	9	8
	保留位				ADWCH			
类型 / 复位					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

位	域	描述
[27:24]	ADUCH	上限阈值通道状态 0000: ADC_IN0 转换的数据高于上限阈值 0001: ADC_IN1 转换的数据高于上限阈值 ... 0101: ADC_IN5 转换的数据高于上限阈值 0110: ADC_IN6 转换的数据高于上限阈值 0111: ADC_IN7 转换的数据高于上限阈值 其它: 保留 若通过看门狗监测功能将该位域中某一位置 1, 则 ADUCH 状态位域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 位域将会改变。
[19:16]	ADLCH	下限阈值通道状态 0000: ADC_IN0 转换的数据低于下限阈值 0001: ADC_IN1 转换的数据低于下限阈值 ... 0101: ADC_IN5 转换的数据高于上限阈值 0110: ADC_IN6 转换的数据高于上限阈值 0111: ADC_IN7 转换的数据高于上限阈值 其它: 保留 若通过看门狗监测功能将该位域中某一位置 1, 则 ADLCH 状态位域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下限阈值时, ADLCH 位域将会改变。

位	域	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 ... 0101: ADC_IN5 转换的数据高于上限阈值 0110: ADC_IN6 转换的数据高于上限阈值 0111: ADC_IN7 转换的数据高于上限阈值 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 位域指定的通道被监控 1: 所有通道被监控
[1]	ADWUE	ADC 看门狗上限阈值使能位 0: 除能上限阈值监控功能 1: 使能上限阈值监控功能
[0]	ADWLE	ADC 看门狗下限阈值使能位 0: 除能下限阈值监控功能 1: 使能下限阈值监控功能

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

位	域	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

类型 / 复位	31	30	29	28	27	26	25	24
	保留位							ADIEO
								RW 0
类型 / 复位	23	22	21	20	19	18	17	16
	保留位						ADIEU	ADIEL
							RW 0	RW 0
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					ADIEC	ADIEG	ADIES
						RW 0	RW 0	RW 0

位	域	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

类型 / 复位	31	30	29	28	27	26	25	24
	保留位							ADIRAWO
								RO 0
类型 / 复位	23	22	21	20	19	18	17	16
	保留位						ADIRAWU	ADIRAWL
							RO 0	RO 0
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					ADIRAWC	ADIRAWG	ADIRAWS
						RO 0	RO 0	RO 0

位	域	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

类型 / 复位	31	30	29	28	27	26	25	24
	保留位							ADISRO
								RO 0
类型 / 复位	23	22	21	20	19	18	17	16
	保留位						ADISRU	ADISRL
							RO 0	RO 0
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					ADISRC	ADISRG	ADISRS
						RO 0	RO 0	RO 0

位	域	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

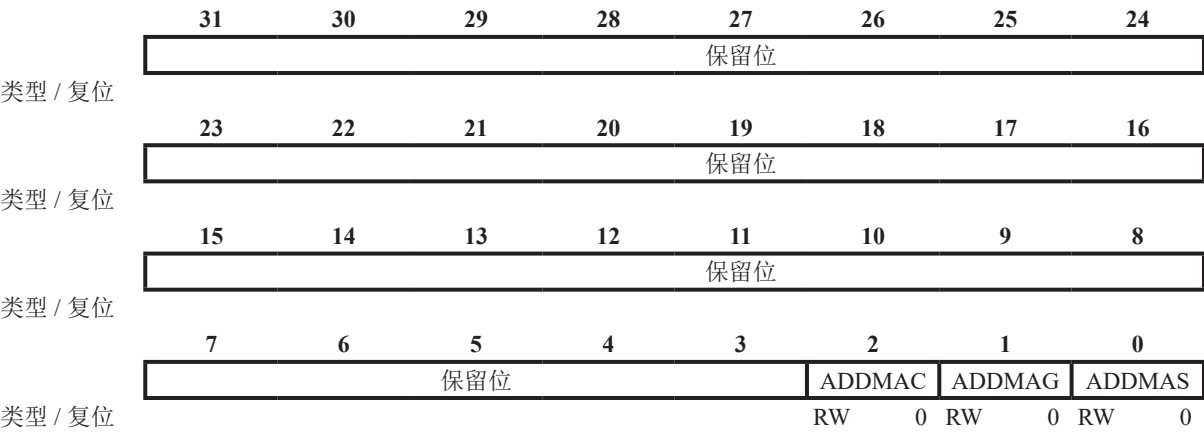
类型 / 复位	31	30	29	28	27	26	25	24
	保留位							ADICLRO
								WO 0
类型 / 复位	23	22	21	20	19	18	17	16
	保留位						ADICLRU	ADICLRL
							WO 0	WO 0
类型 / 复位	15	14	13	12	11	10	9	8
	保留位							
类型 / 复位	7	6	5	4	3	2	1	0
	保留位					ADICLRC	ADICLRG	ADICLRS
						WO 0	WO 0	WO 0

位	域	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

ADC DMA 请求寄存器 – ADCDMAR

该寄存器包含了 ADC DMA 请求使能位。

偏移量： 0x090
复位值： 0x0000_0000



位	域	描述
[2]	ADDMAC	ADC 周期转换结束 DMA 请求使能位 0: ADC 周期转换结束 DMA 请求被除能 1: ADC 周期转换结束 DMA 请求被使能
[1]	ADDMAG	ADC 子组转换结束 DMA 请求使能位 0: ADC 子组转换结束 DMA 请求被除能 1: ADC 子组转换结束 DMA 请求被使能
[0]	ADDMAS	ADC 单次转换结束 DMA 请求使能位 0: ADC 单次转换结束 DMA 请求被除能 1: ADC 单次转换结束 DMA 请求被使能

参考电压控制寄存器 – VREFCR

该寄存器包含了内部参考电压 V_{REF} 控制位。

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	
	保留位							MVDDAEN	
类型 / 复位								RW	0
	7	6	5	4	3	2	1	0	
	保留位		VREFSEL		保留位		VREFEN		
类型 / 复位			RW 0 RW 0				RW 0		

位	域	描述
[8]	MVDDAEN	电压 ($V_{DDA}/2$) 测量功能使能位 0: 除能 1: 使能
[5:4]	VREFSEL	参考电压输出值选择位 00: 1.215 V 01: 2.0 V 10: 2.5 V 11: 2.7 V 该位域用于选择参考电压输出的电平值。
[0]	VREFEN	内部参考电压电路使能位 0: 除能 1: 使能

参考电压值寄存器 – VREFVALR

该寄存器储存内部参考电压校准后值。

偏移量:	0x0A4
复位值:	0x0000_00XX (XX 为未知，取决于 Flash 制造权限信息块储存的值)
	3130292827262524
类型 / 复位	保留位
	2322212019181716
类型 / 复位	保留位
	15141312111098
类型 / 复位	保留位
	76543210
类型 / 复位	保留位VREFVAL
	ROX ROX ROX ROX ROX ROX ROX

位	域	描述
[6:0]	VREFVAL	内部参考电压校准值 制造过程中，会对每个单片机内部参考电压校准，并将校准值储存在 Flash 制造权限信息块中。系统上电后，该校准值会被载入到此寄存器。

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。

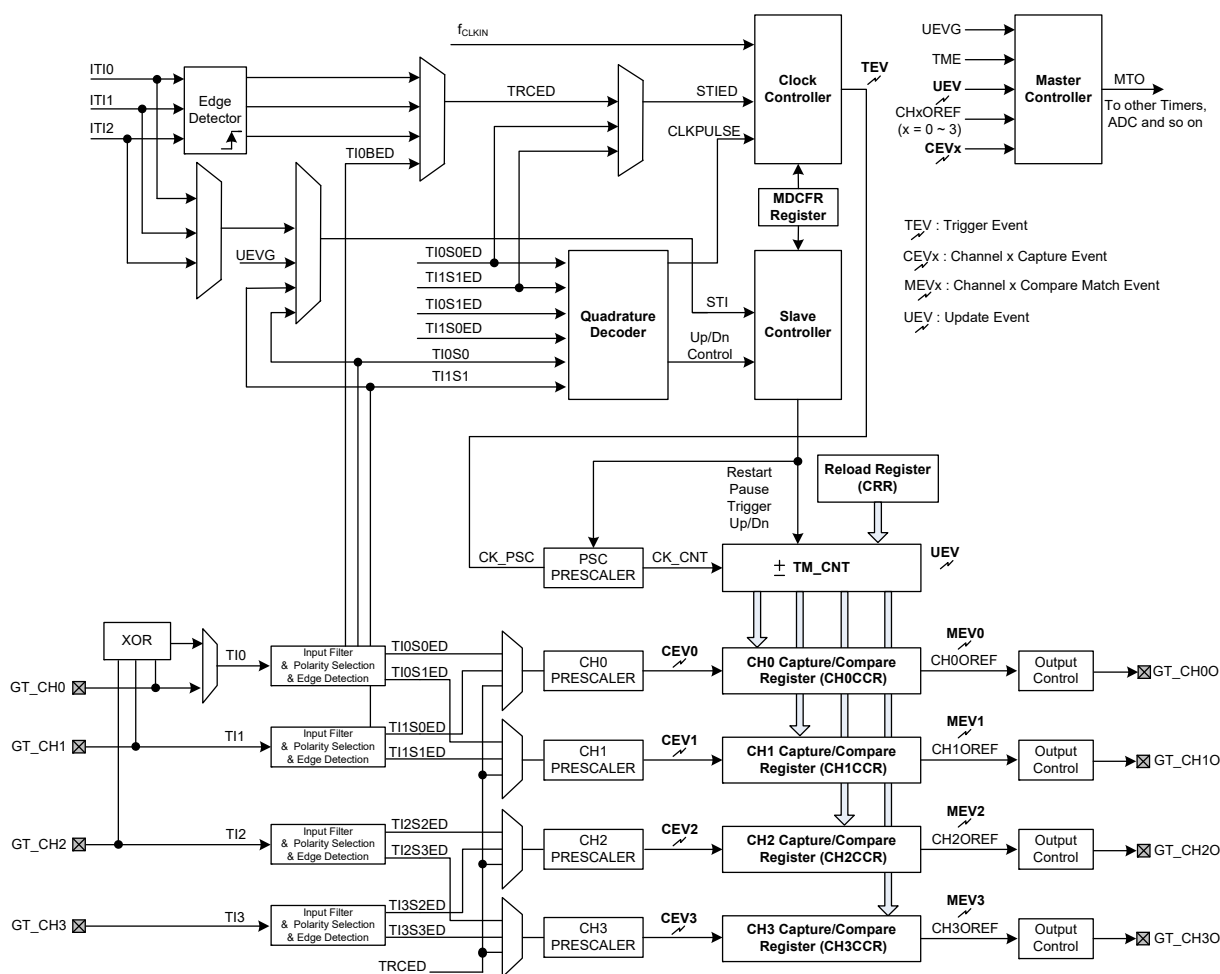


图 34. GPTM 方框图

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 / PDMA 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，接着从 0 开始重新计数并产生一个溢出事件。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。

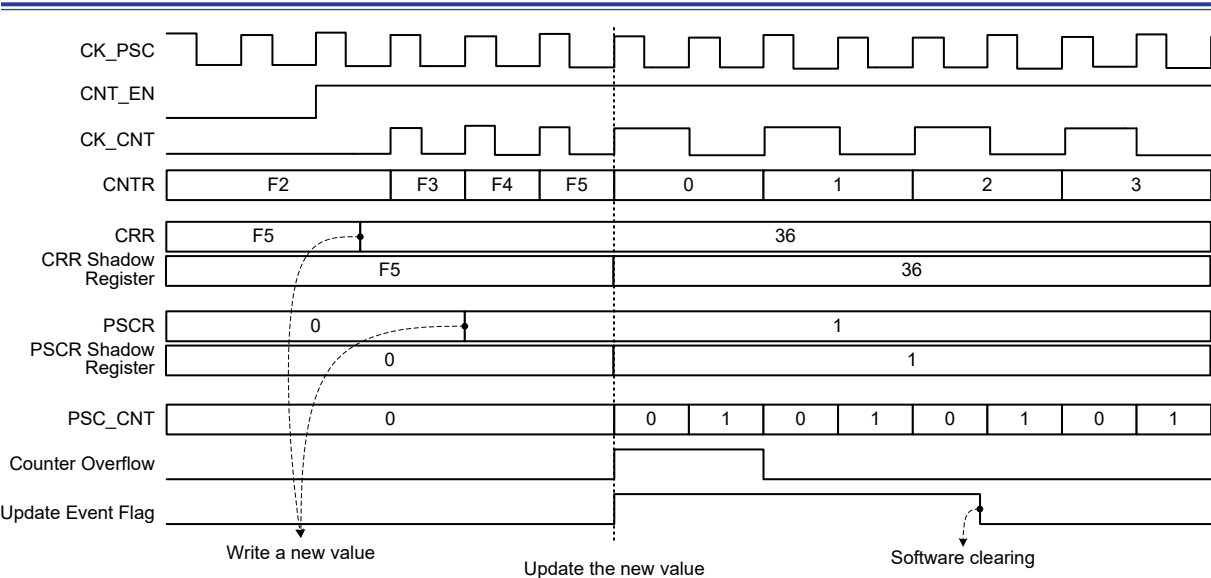


图 35. 向上计数范例

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，接着从计数器重载值开始重新计数并产生一个下溢事件。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。

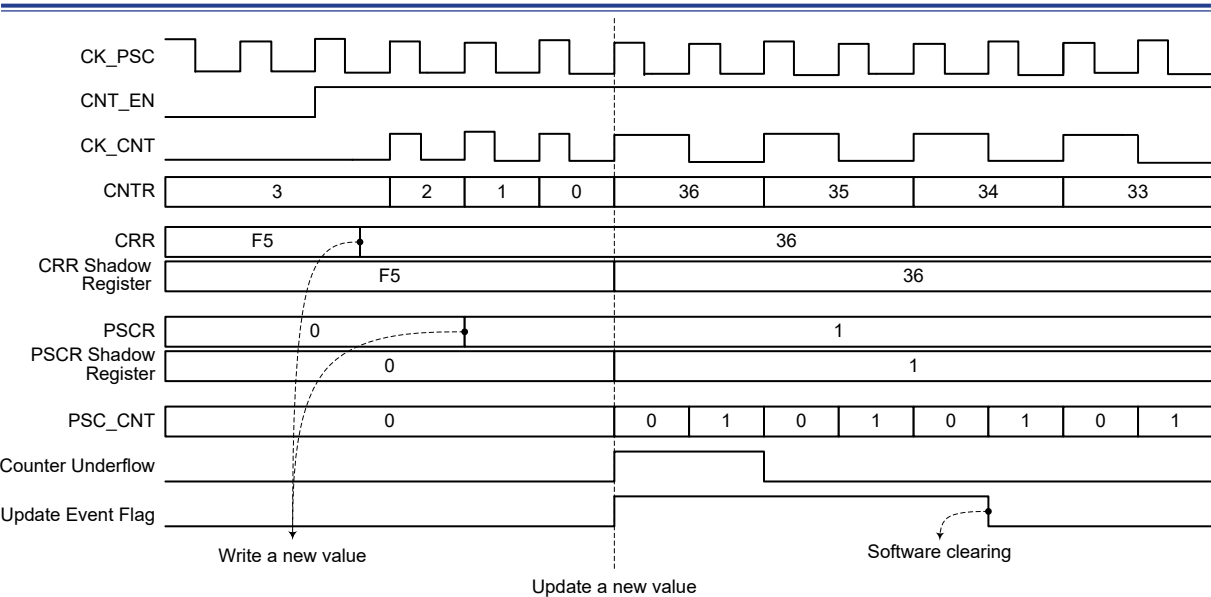


图 36. 向下计数范例

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。

当发生上溢事件或下溢事件，INTSR 寄存器中的更新事件中断标志位 UEVIF 将被置为 1。

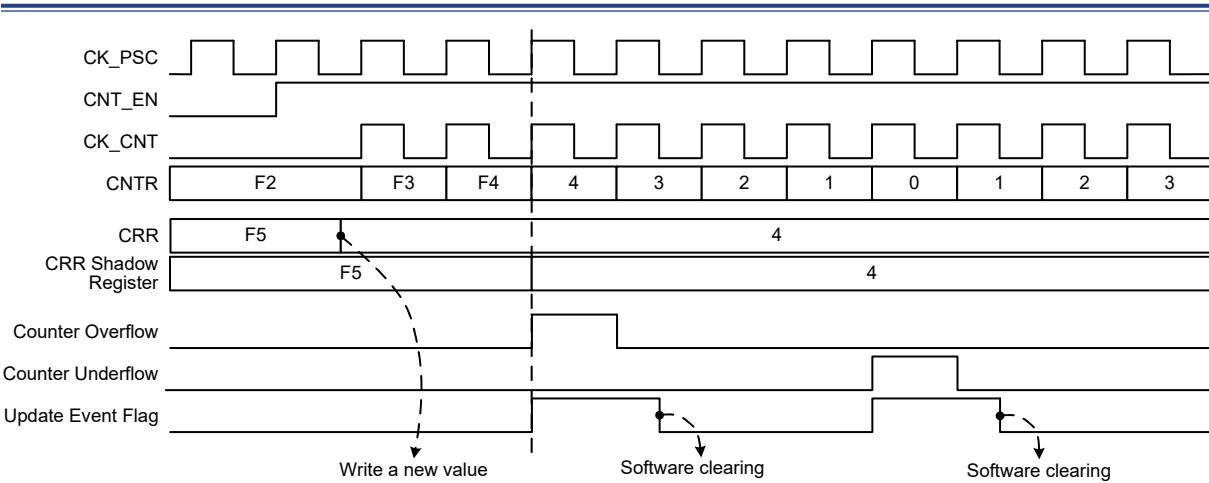


图 37. 中心对齐计数范例

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 位域为 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 位域设置为其它时钟源，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 位域设为 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，也可以来自于这两个引脚。

■ 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 位域设为 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 位域设成除 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 位域设成 0x0 来选择软件 UEVG 位作为触发源，那么当 SMSEL 位域设成 0x7 时，计数器将会被更新而非计数。

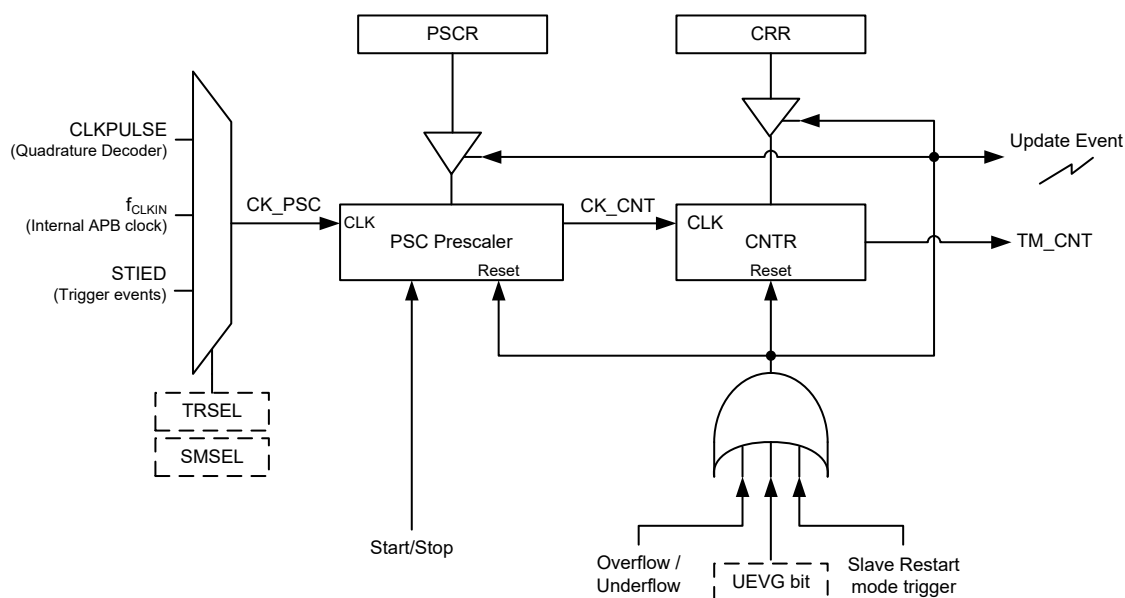


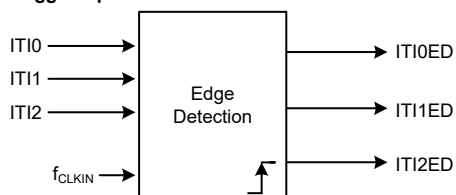
图 38. GPTM 时钟源选择

触发控制器

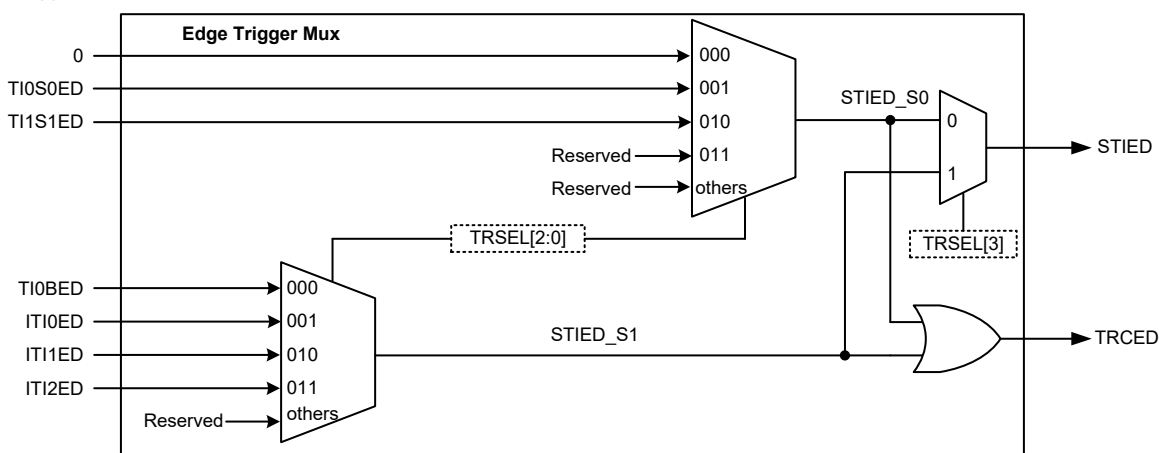
触发控制器用来选择触发源以及设置电平触发或边沿触发条件。内部触发输入可通过 TRCFR 寄存器中的触发选择位 TRSEL 进行选择。除了 UEVG 位软件触发之外的所有触发源，内部边沿检测电路将会在每个触发信号上升沿作用期间产生一个时钟脉冲，以激活某些因触发信号上升沿而触发的 GPTM 功能。

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

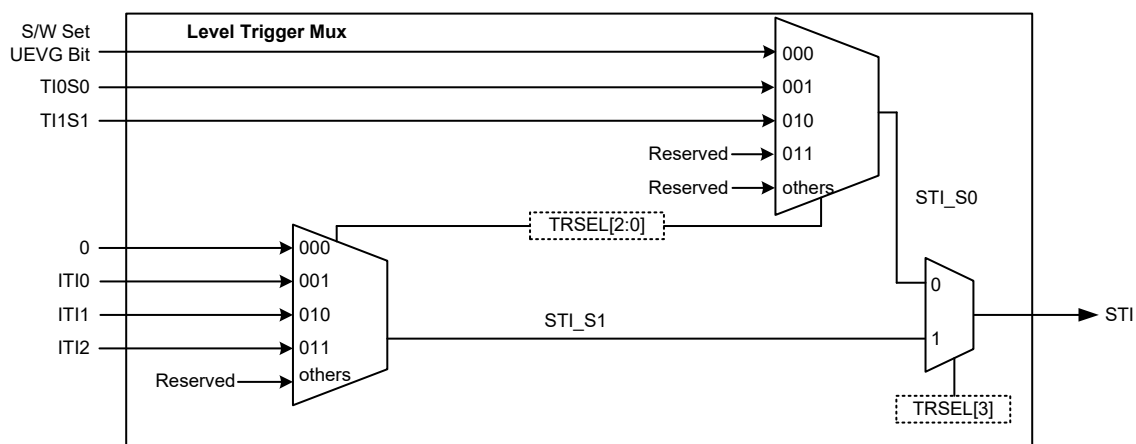


图 39. 触发控制器方框图

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 位域选择的。这些模式的触发输入来自于 STI 信号，通过 TRCFR 寄存器中的 TRSEL 位域选择。从机控制器中的工作模式在相关章节中有所描述。

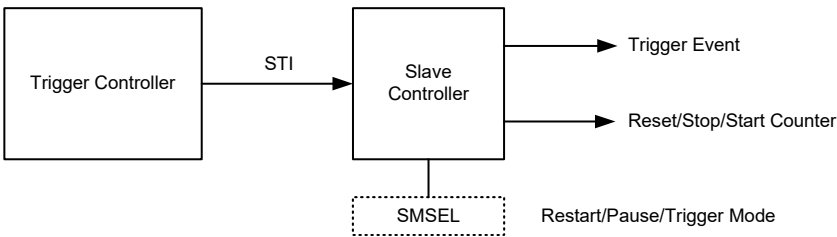


图 40. 从机控制器方框图

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。

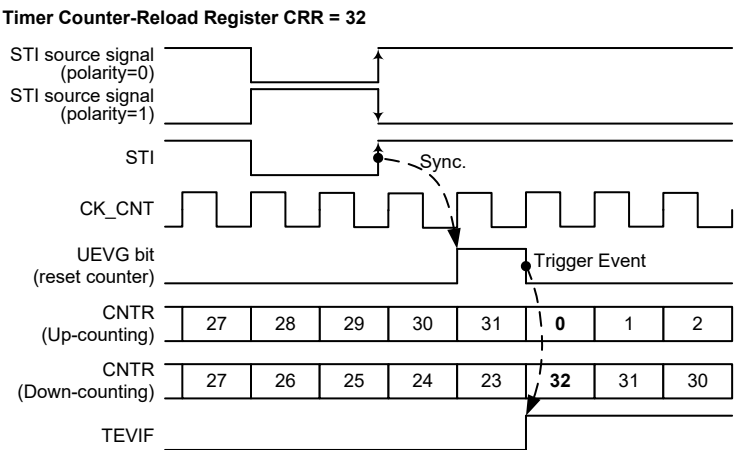


图 41. 重启模式下的 GPTM

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。

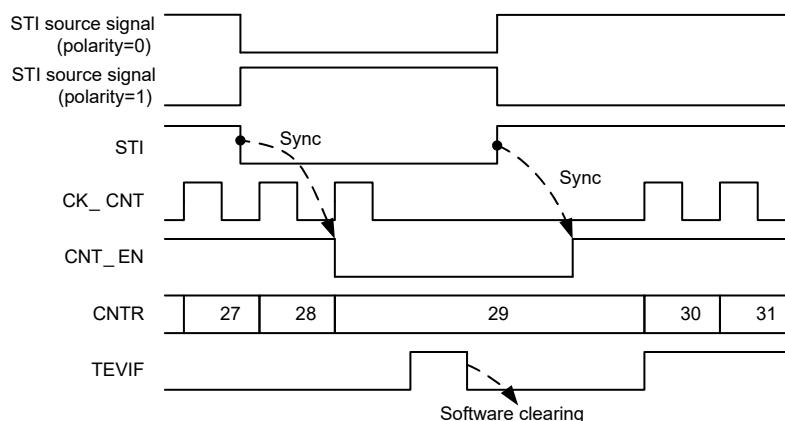


图 42. 暂停模式下的 GPTM

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, 计数器不会继续计数。当 STI 源信号通过 UEVG 位选择为软件触发时, 不会产生使计数器继续计数的时钟脉冲。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。

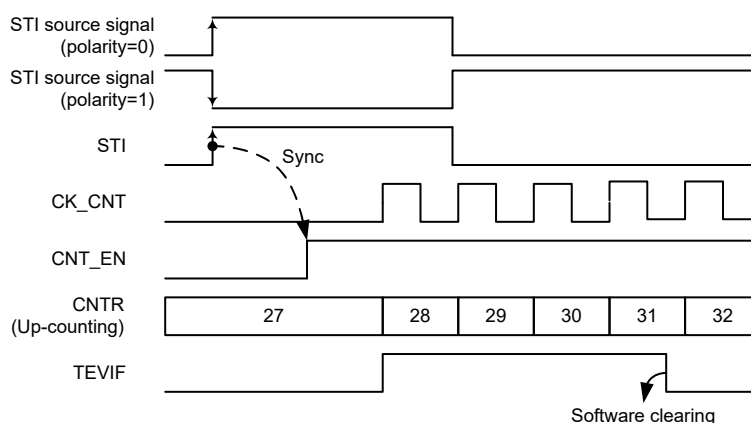


图 43. 触发模式下的 GPTM

主机控制器

GPTM 和其它定时器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 位域选择可以触发或驱动处于从机模式下的 GPTM 或其它定时器、启动、停止或为其提供时钟源。

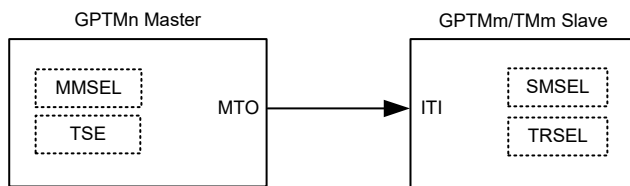


图 44. 主机 GPTMn 和从机 GPTMm/TMm 相连接

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

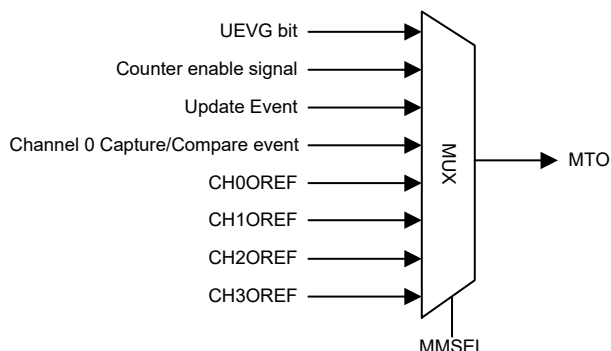


图 45. MTO 选项

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

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。

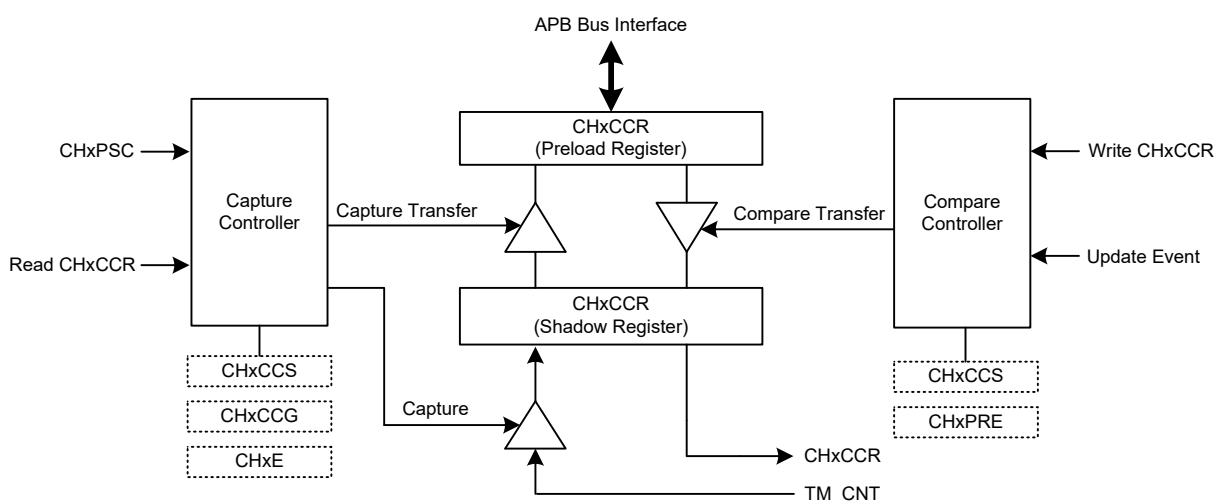


图 46. 捕捉 / 比较方框图

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。

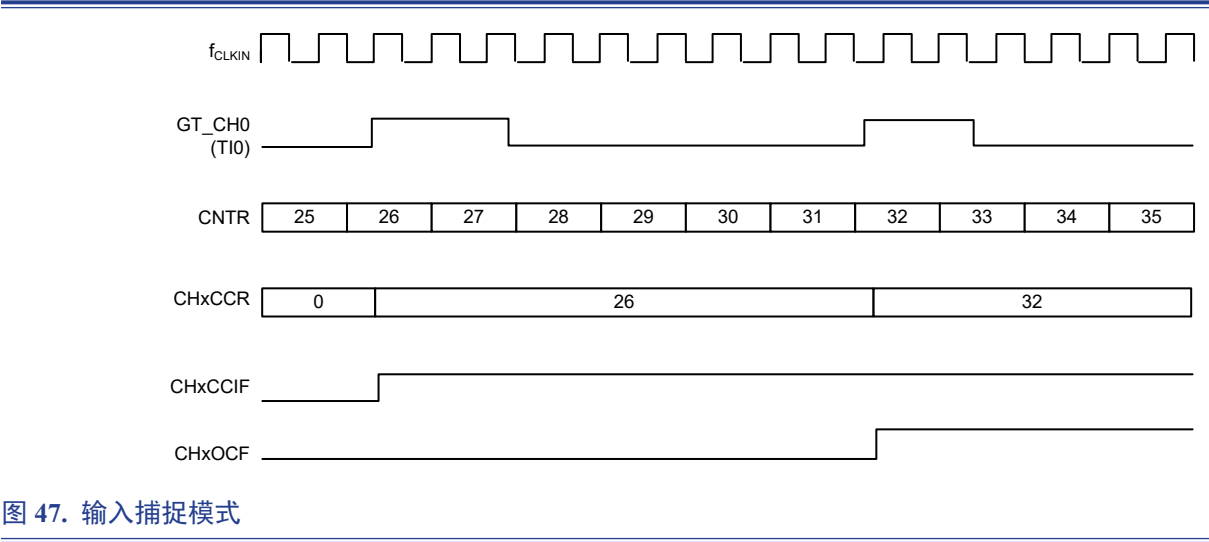


图 47. 输入捕捉模式

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 位域置为 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 CH0CCR 寄存器中，输入信号周期将被捕捉到 CH1CCR 寄存器中。

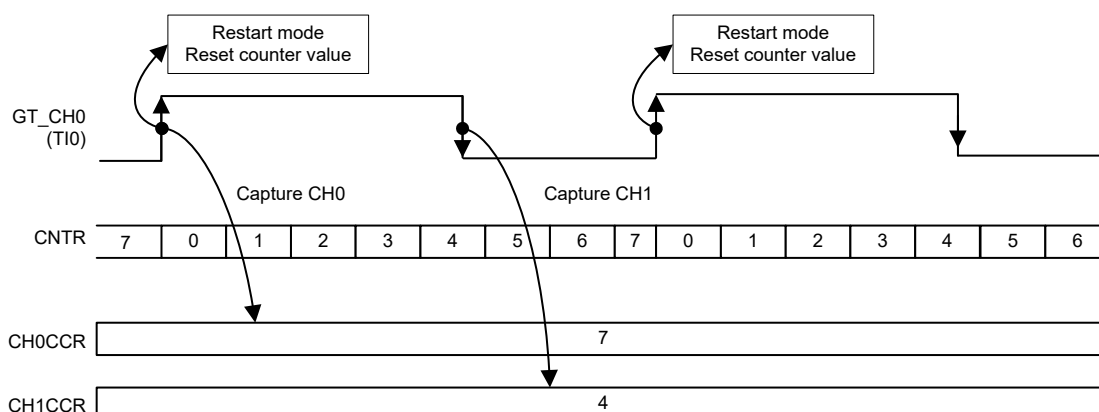


图 48. PWM 脉冲宽度测量范例

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 TIxSyED 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配置寄存器 CHxICFR 中的 CHxPSC 位域设置。

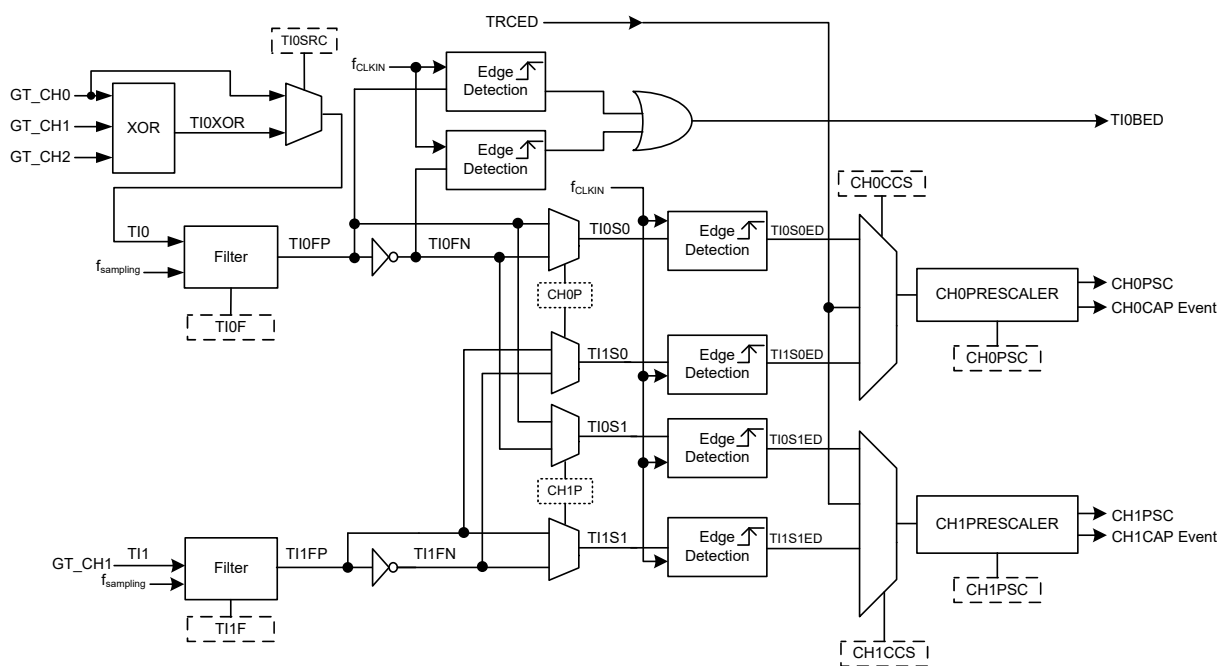


图 49. 通道 0 和通道 1 输入级

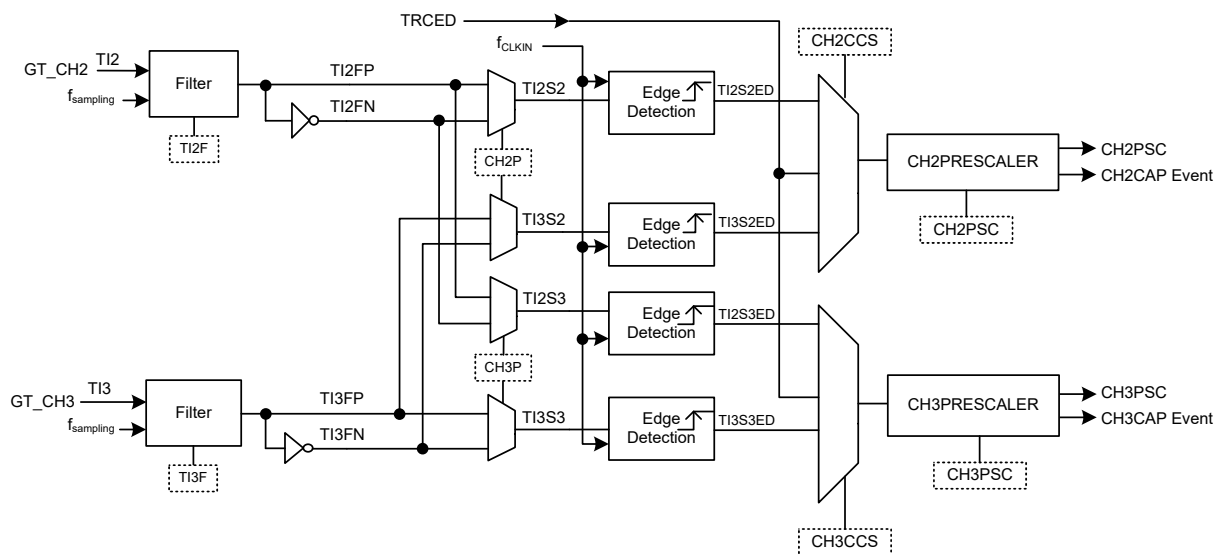


图 50. 通道 2 和通道 3 输入级

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，可通过 CHxICFR 寄存器中的 TIxF 位域进行设置。

Digital Filter (N=2)

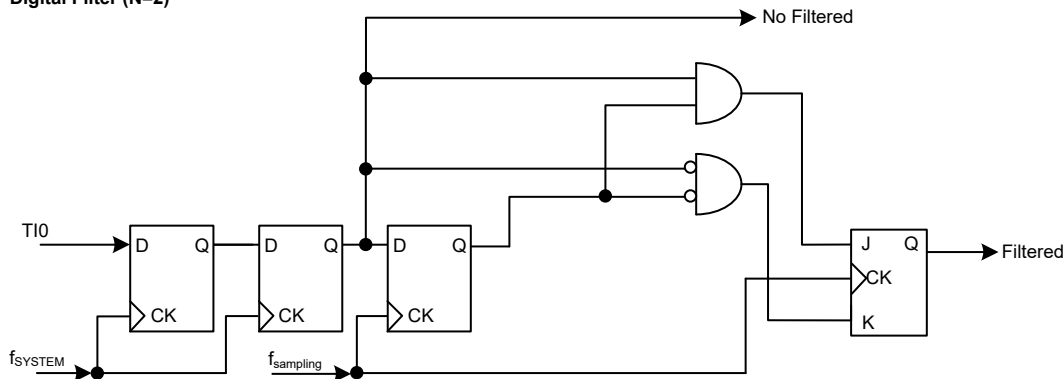


图 51. TI0 数字滤波器方框图 (N = 2)

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 位域为 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。

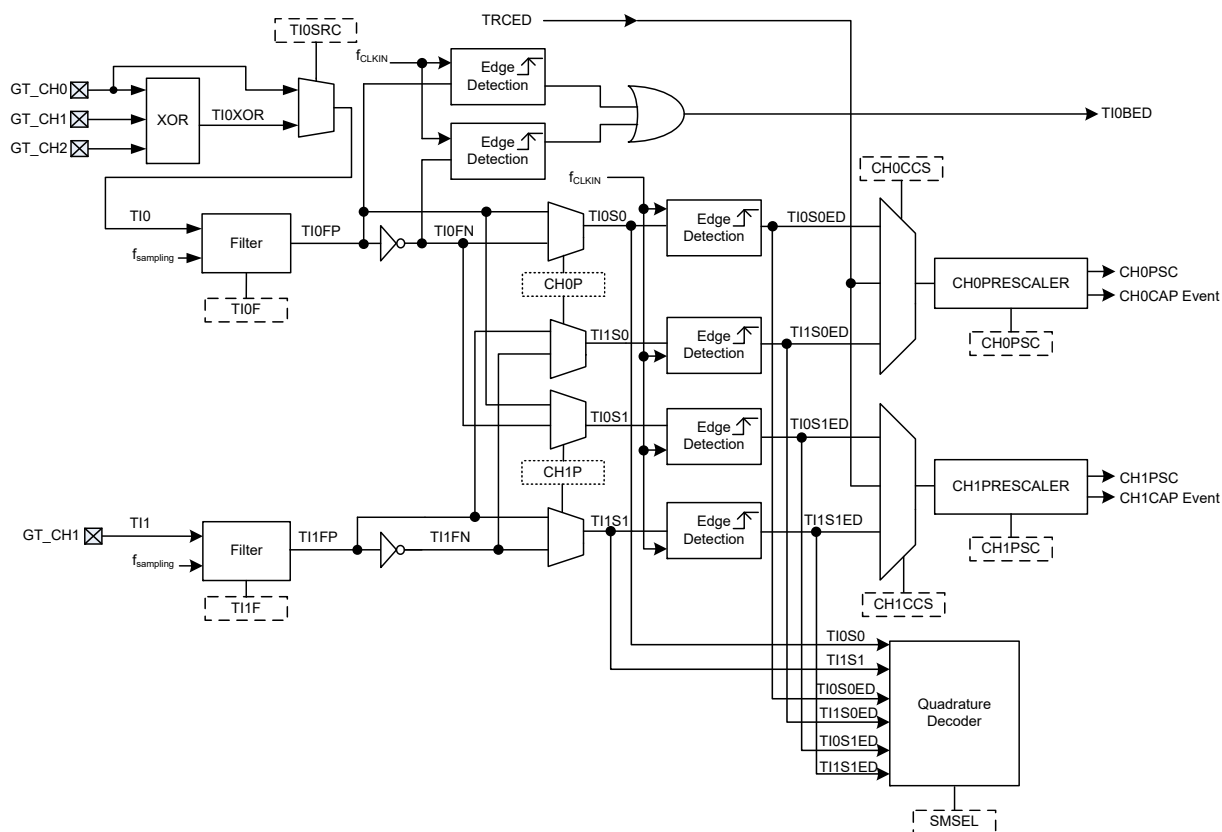


图 52. 输入级和正交解码器方框图

表 31. 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

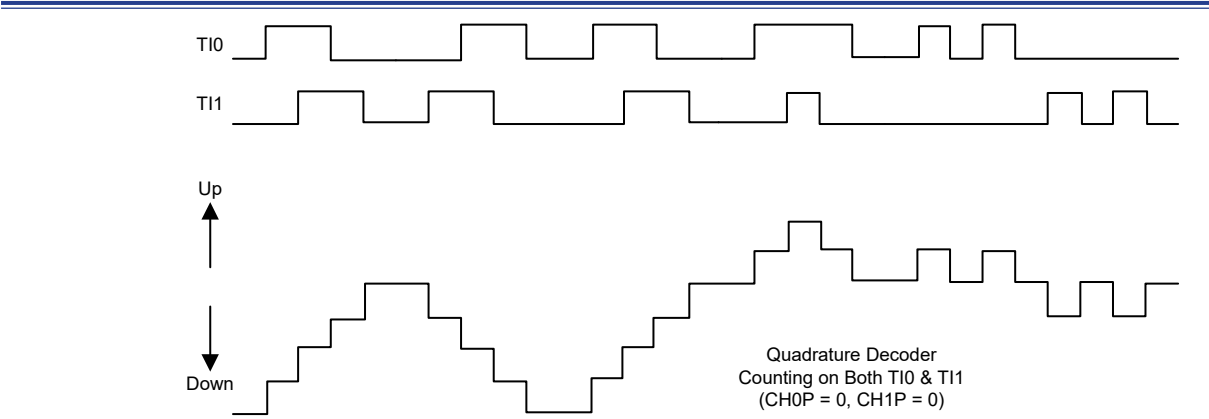


图 53. TI0 和 TI1 正交解码器计数

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 CHxOCFR、CHPOLR 和 CHCTR 寄存器分别对应的 CHxOM、CHxP 和 CHxE 位控制。

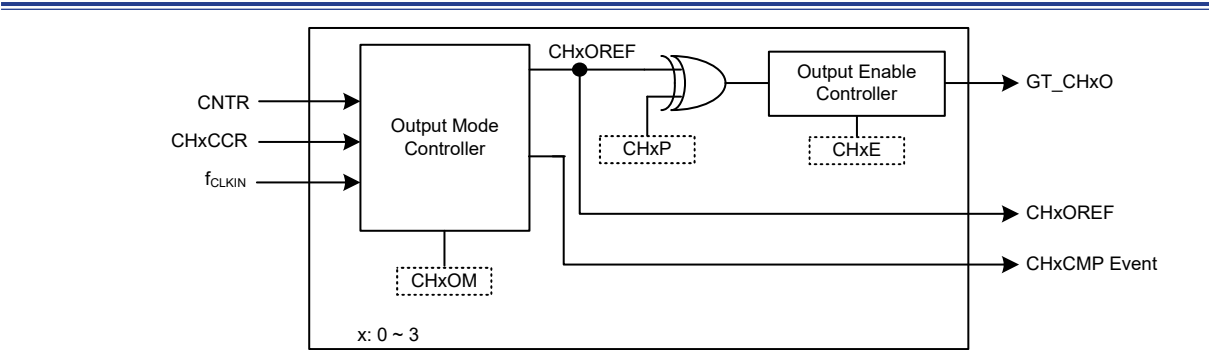


图 54. 输出级方框图

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 位来定义。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。输出类型设置如下表所示。

表 32. 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

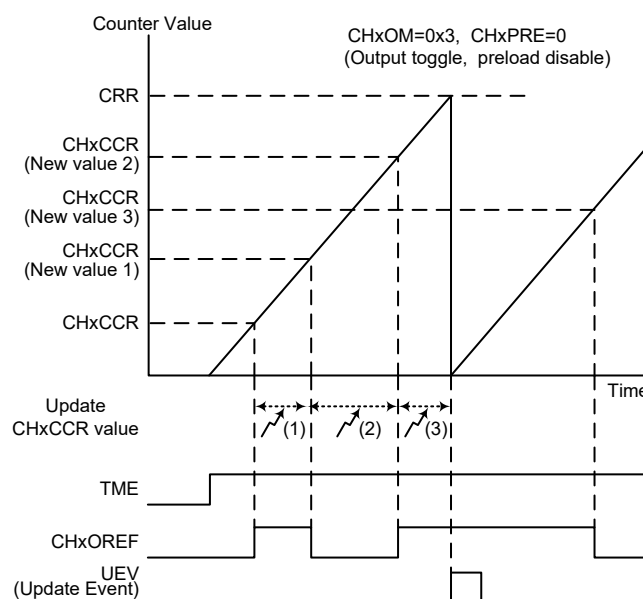


图 55. 翻转模式通道输出参考信号 – CHxPRE = 0

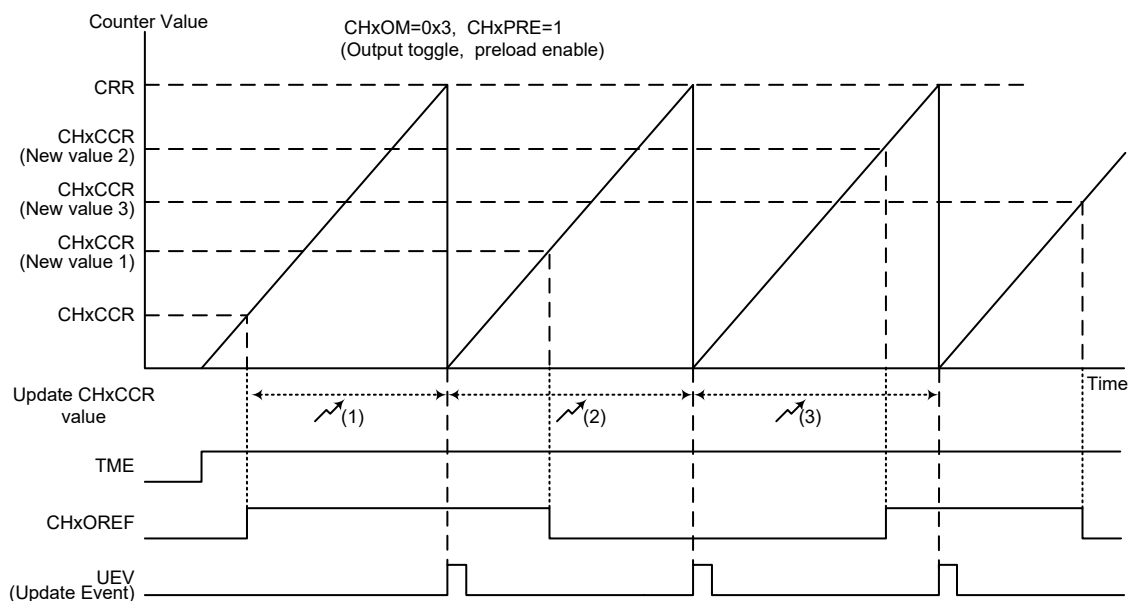


图 56. 翻转模式通道输出参考信号 – $CHxPRE = 1$

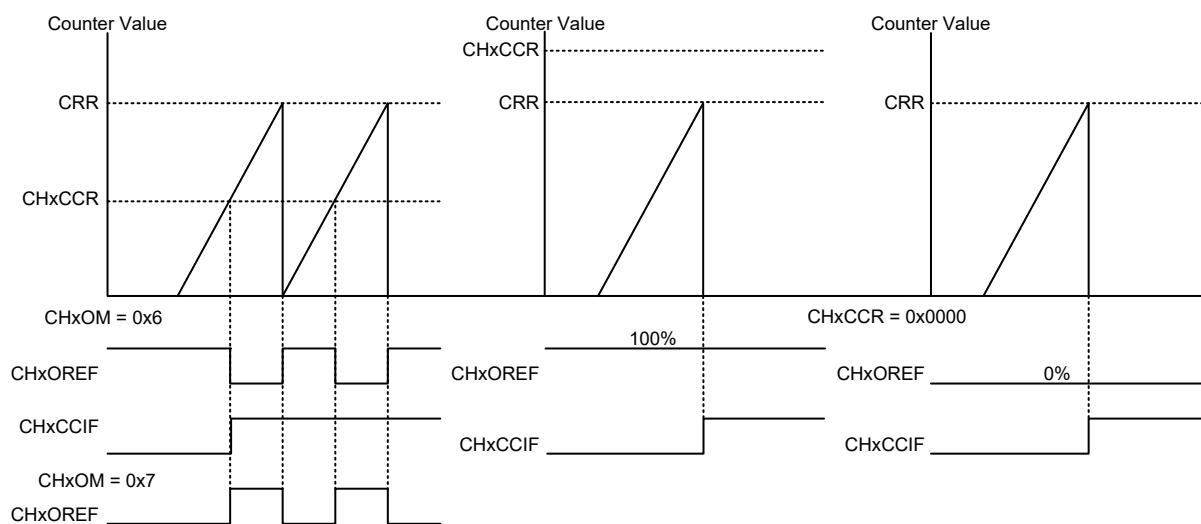


图 57. PWM 模式通道输出参考信号和计数器向上计数模式

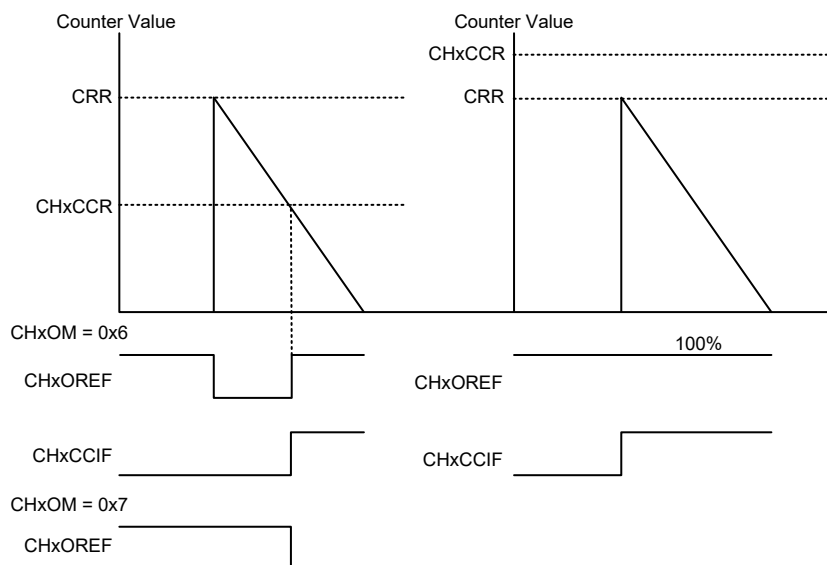


图 58. PWM 模式通道输出参考信号和计数器向下计数模式

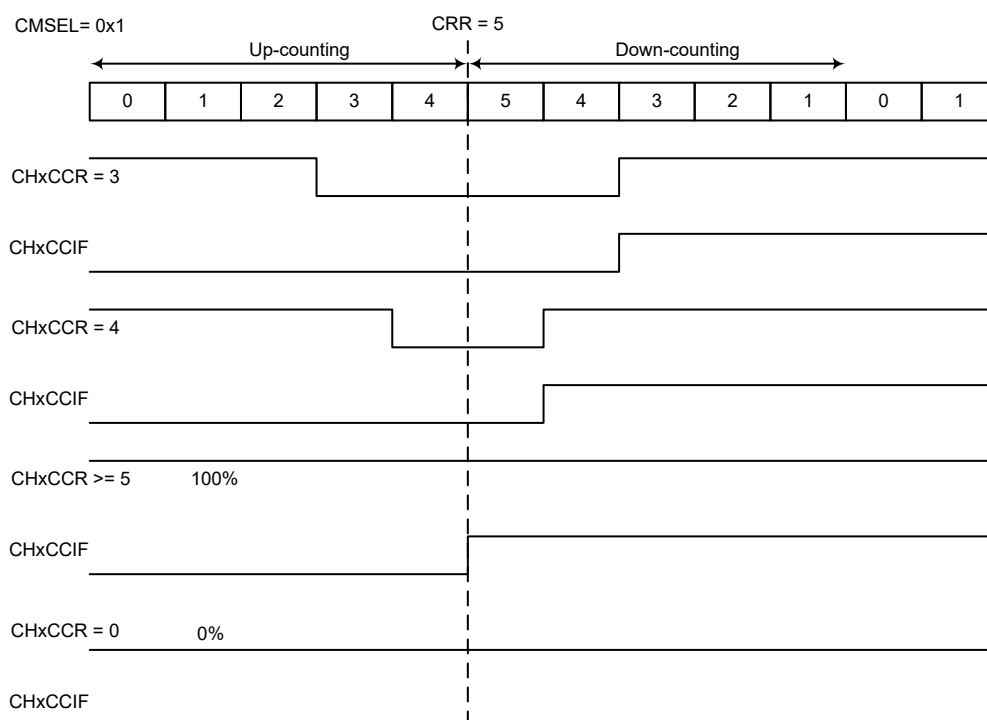


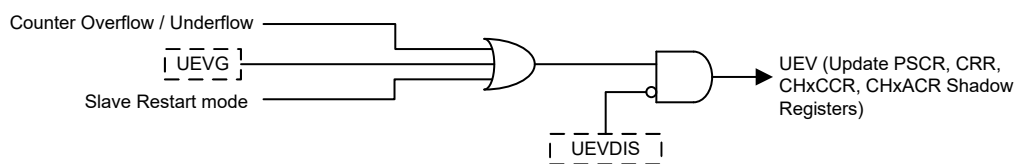
图 59. PWM 模式通道输出参考信号和计数器中心对齐计数模式

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

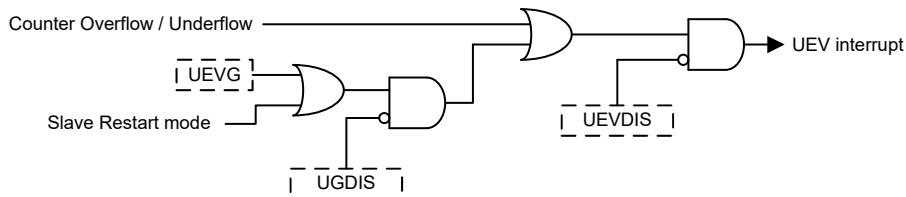


图 60. 更新事件设置方框图

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 位是因更新事件由硬件自动清零, 计数器将被重新初始化。

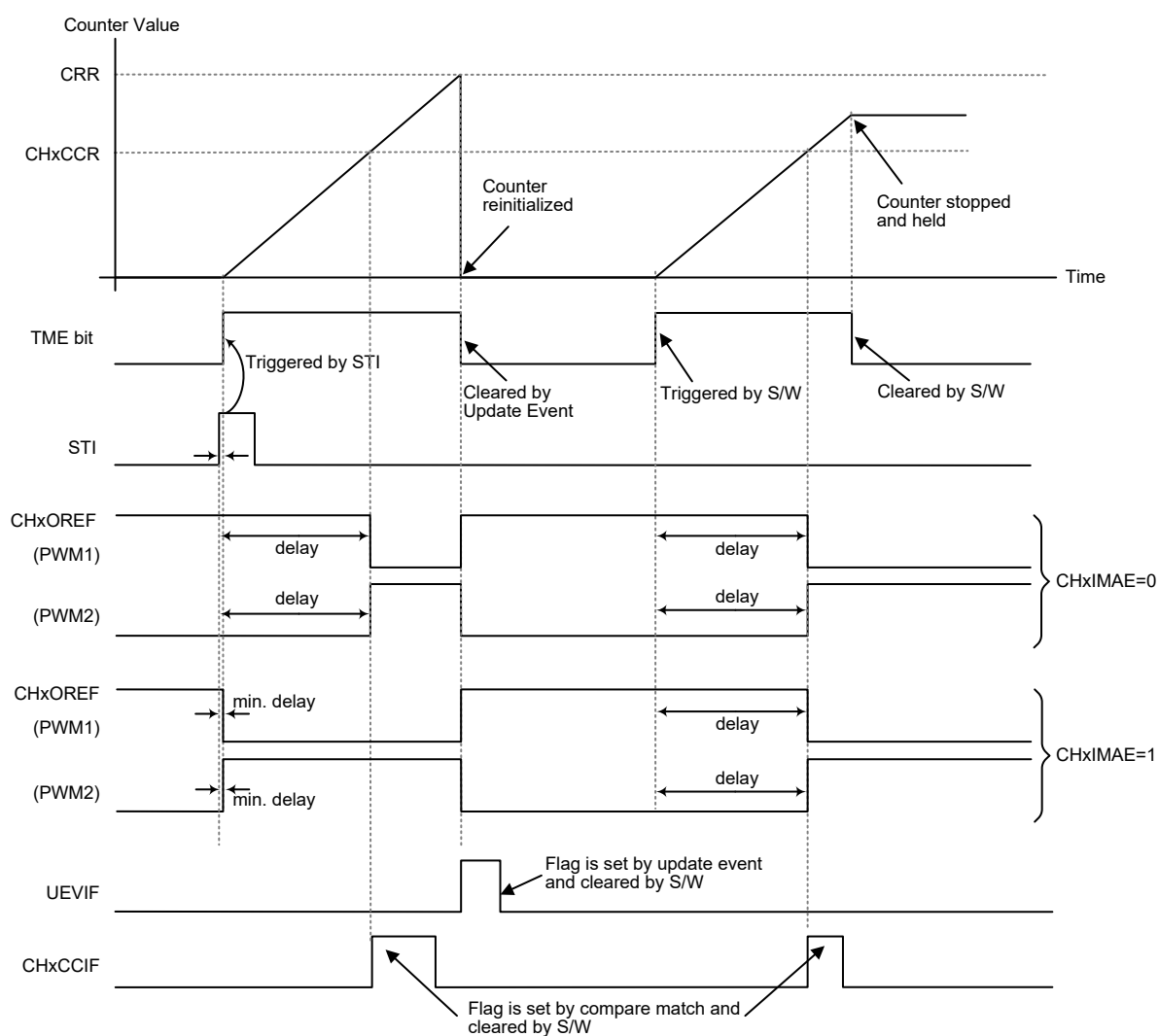


图 61. 单脉冲模式

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 PWM1 或 PWM2 模式下且触发源来自于 STI 信号时, CHxIMAE 位才可用。

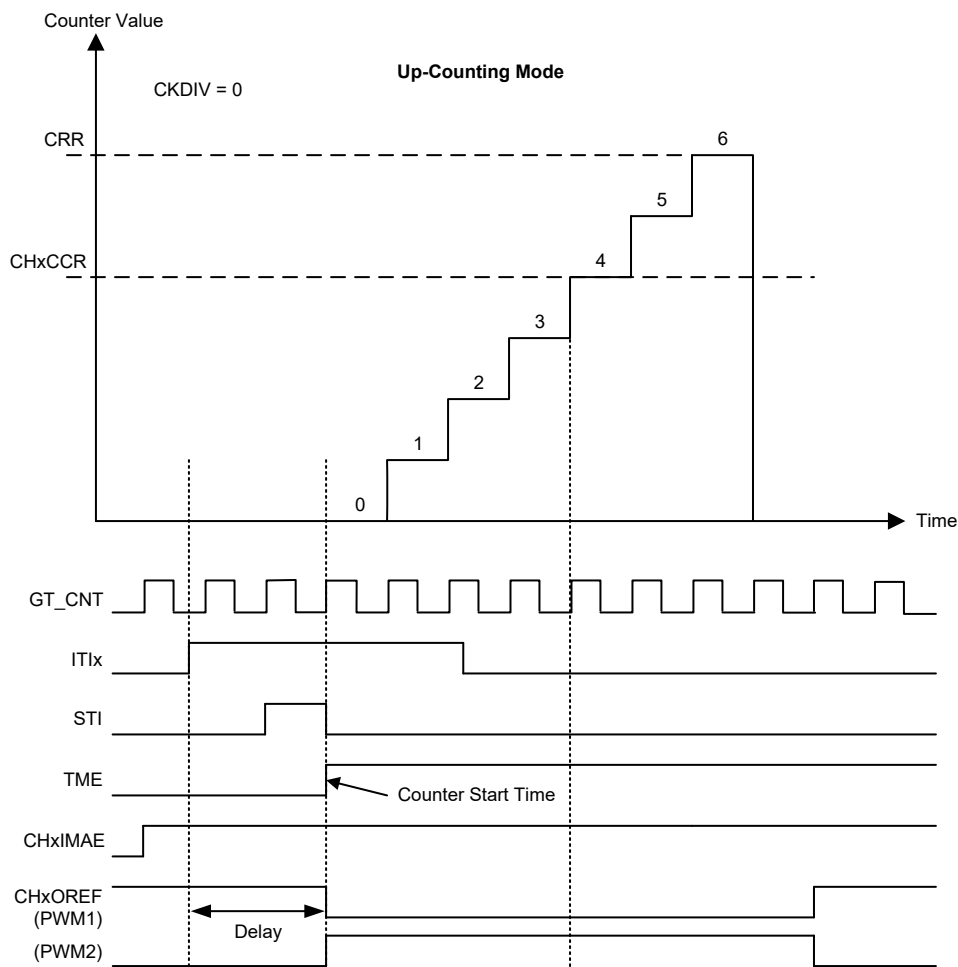


图 62. 立即有效模式的延迟

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 (SMSEL = 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

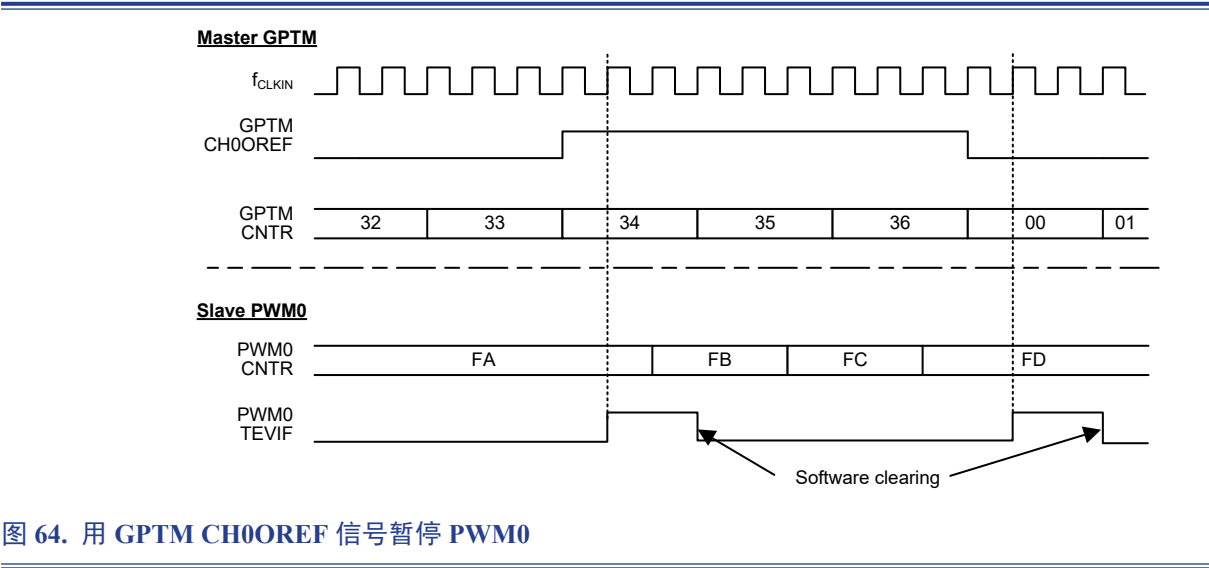


图 64. 用 GPTM CH0OREF 信号暂停 PWM0

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

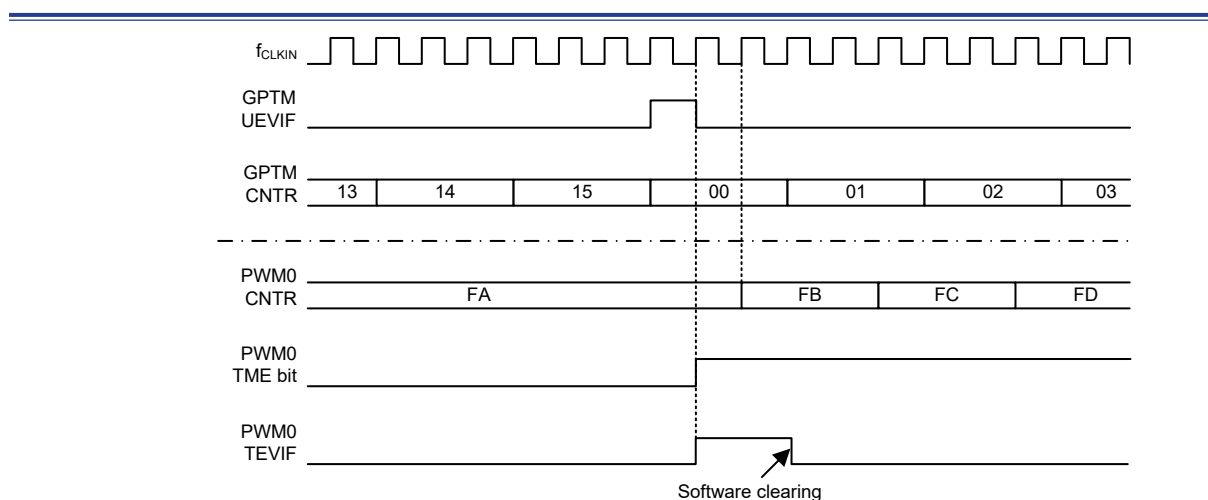
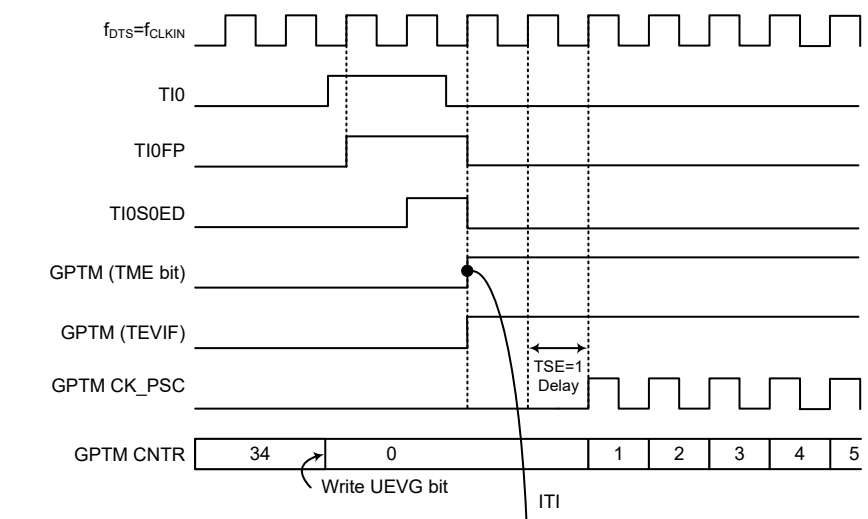


图 65. 用 GPTM 更新事件触发 PWM0

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)

Master GPTM



Slave PWM0

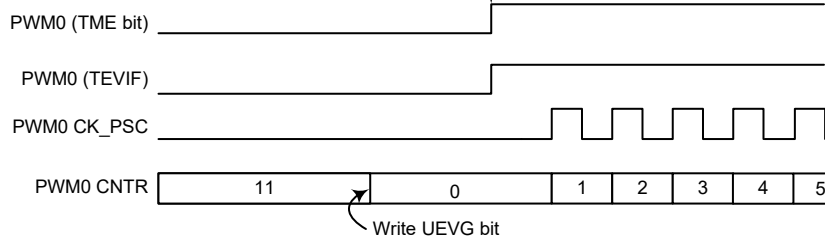


图 66. 用 GPTM CH0 输入信号触发 GPTM 和 PWM0

触发其它外设功能启动

为了与其它外设功能 (如 ADC、定时器等) 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) 作为外设输入触发信号, 取决于 MCU 的规格。

PDMA 请求

GPTM 提供了 PDMA 数据传输接口。GPTM 更新事件, 触发事件和通道捕捉 / 比较事件, 可产生 PDMA 请求, 如果其相应的使能控制位置 1 则可使能 PDMA 存取。当 GPTM 通道产生 PDMA 请求时, 可通过通道 PDMA 选择位 CHCCDS 选择是来自通道捕捉 / 比较事件还是 GPTM 更新事件。更多 PDMA 配置信息, 请参考 PDMA 章节。

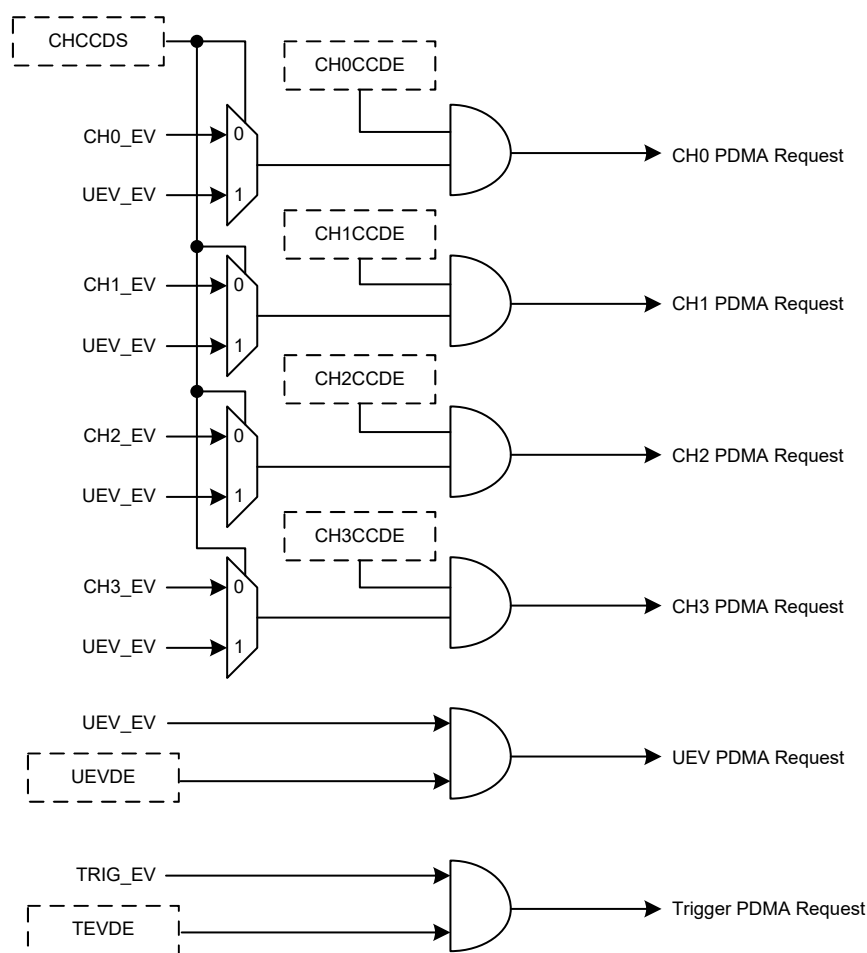


图 67. GPTM PDMA 映射图

寄存器列表

下表显示了 GPTM 寄存器及其复位值。

表 33. 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 PDMA / 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

位	域	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 位为 1 – 通过从机模式产生更新 1: 除能更新事件 (如果 UEVG 位被置位或在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	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	域	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

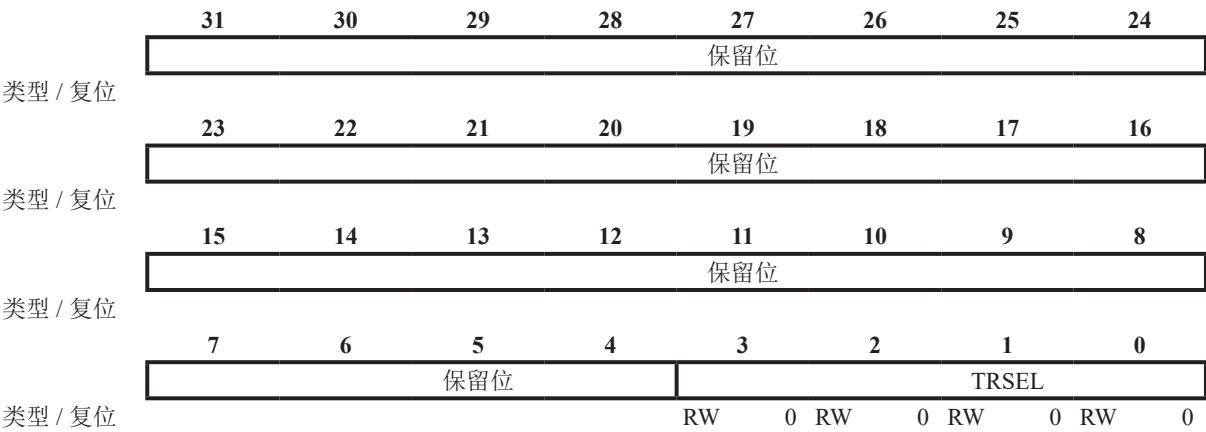
位	域	描述																											
[18:16]	MMSEL	<p>主机模式选项</p> <p>主机模式选项用来选择与其它从机定时器同步的 MTO 信号源。</p> <table> <tr> <th>MMSEL[2:0]</th><th>模式</th><th>描述</th></tr> <tr> <td>000</td><td>复位模式</td><td>复位模式下的 MTO 输出信号由以下条件之一产生： 1. 软件设置 UEVG 位为 1 2. 当定时器用于从机重启模式，STI 触发输入信号被输出到 MTO 信号线</td></tr> <tr> <td>001</td><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 位为 1 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> </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 位为 1 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 位为 1 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 位为 1 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 位为 1 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> <tr> <th>SMSEL[2:0]</th><th>模式</th><th>描述</th></tr> <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>STI 触发信号上升沿处，计数器从当前值开始计数。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> </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	STI 触发信号上升沿处，计数器从当前值开始计数。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	STI 触发信号上升沿处，计数器从当前值开始计数。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	域	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 位域为 0x0 除能从机模式时, 这些位才能被更新。

表 34. 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)、CRR 缓冲器使能位 (CRBE) 和通道 PDMA 选择位 (CHCCDS)。

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							CHCCDS	
类型 / 复位								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						CRBE	TME	
类型 / 复位							RW	0	RW 0

位	域	描述
[16]	CHCCDS	通道 PDMA 事件选择 0: 通道 PDMA 请求来自通道捕捉 / 比较事件 1: 通道 PDMA 请求来自更新事件
[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
	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				TI0F			
类型 / 复位					RW	0	RW	0

位	域	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 位域。

位	域	描述
[3:0]	TI0F	通道 0 输入源 TI0 滤波器设置 这些位定义了用来采样 TI0 信号的采样频率。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 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							
类型 / 复位							
23	22	21	20	19	18	17	16
保留位				CH1PSC		CH1CCS	
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				TI1F			
类型 / 复位				RW	0	RW	0

位	域	描述
[19:18]	CH1PSC	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

位	域	描述
[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 位域。
[3:0]	TI1F	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。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							
类型 / 复位							
23	22	21	20	19	18	17	16
保留位				CH2PSC		CH2CCS	
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				TI2F			
类型 / 复位				RW	0	RW	0

位	域	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 位域。
[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。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: $f_{\text{sampling}} = f_{\text{CLKIN}}$, $N = 2$ 0010: $f_{\text{sampling}} = f_{\text{CLKIN}}$, $N = 4$ 0011: $f_{\text{sampling}} = f_{\text{CLKIN}}$, $N = 8$ 0100: $f_{\text{sampling}} = f_{\text{DTS}} / 2$, $N = 6$ 0101: $f_{\text{sampling}} = f_{\text{DTS}} / 2$, $N = 8$ 0110: $f_{\text{sampling}} = f_{\text{DTS}} / 4$, $N = 6$ 0111: $f_{\text{sampling}} = f_{\text{DTS}} / 4$, $N = 8$ 1000: $f_{\text{sampling}} = f_{\text{DTS}} / 8$, $N = 6$ 1001: $f_{\text{sampling}} = f_{\text{DTS}} / 8$, $N = 8$ 1010: $f_{\text{sampling}} = f_{\text{DTS}} / 16$, $N = 5$ 1011: $f_{\text{sampling}} = f_{\text{DTS}} / 16$, $N = 6$ 1100: $f_{\text{sampling}} = f_{\text{DTS}} / 16$, $N = 8$ 1101: $f_{\text{sampling}} = f_{\text{DTS}} / 32$, $N = 5$ 1110: $f_{\text{sampling}} = f_{\text{DTS}} / 32$, $N = 6$ 1111: $f_{\text{sampling}} = f_{\text{DTS}} / 32$, $N = 8$

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

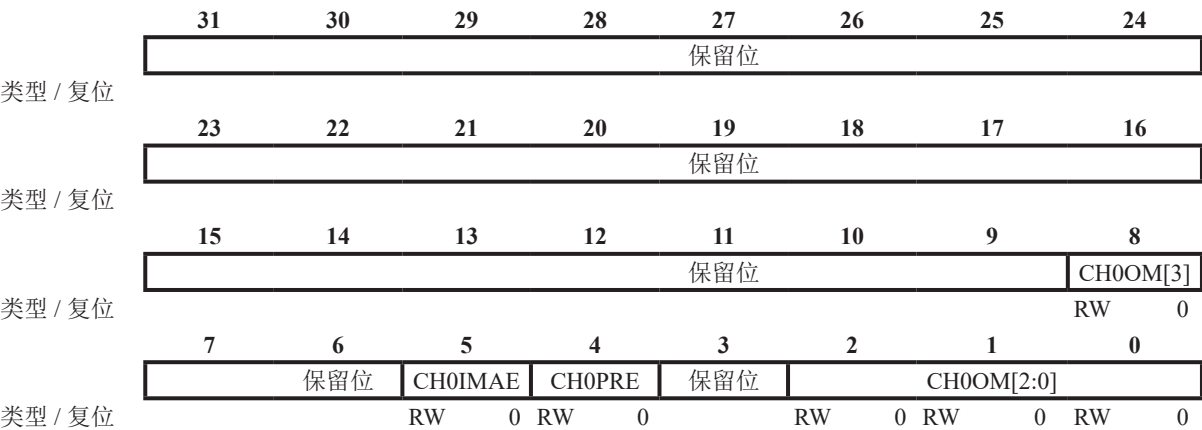
类型 / 复位	31	30	29	28	27	26	25	24
	保留位							
类型 / 复位	23	22	21	20	19	18	17	16
	保留位				CH3PSC		CH3CCS	
					RW	0	RW	0
类型 / 复位	15	14	13	12	11	10	9	8
	保留位							
类型 / 复位	7	6	5	4	3	2	1	0
	保留位				TI3F			
					RW	0	RW	0

位	域	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 位域。
[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。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： $f_{\text{sampling}} = f_{\text{CLKIN}}$, $N = 2$ 0010： $f_{\text{sampling}} = f_{\text{CLKIN}}$, $N = 4$ 0011： $f_{\text{sampling}} = f_{\text{CLKIN}}$, $N = 8$ 0100： $f_{\text{sampling}} = f_{\text{DTS}} / 2$, $N = 6$ 0101： $f_{\text{sampling}} = f_{\text{DTS}} / 2$, $N = 8$ 0110： $f_{\text{sampling}} = f_{\text{DTS}} / 4$, $N = 6$ 0111： $f_{\text{sampling}} = f_{\text{DTS}} / 4$, $N = 8$ 1000： $f_{\text{sampling}} = f_{\text{DTS}} / 8$, $N = 6$ 1001： $f_{\text{sampling}} = f_{\text{DTS}} / 8$, $N = 8$ 1010： $f_{\text{sampling}} = f_{\text{DTS}} / 16$, $N = 5$ 1011： $f_{\text{sampling}} = f_{\text{DTS}} / 16$, $N = 6$ 1100： $f_{\text{sampling}} = f_{\text{DTS}} / 16$, $N = 8$ 1101： $f_{\text{sampling}} = f_{\text{DTS}} / 32$, $N = 5$ 1110： $f_{\text{sampling}} = f_{\text{DTS}} / 32$, $N = 6$ 1111： $f_{\text{sampling}} = f_{\text{DTS}} / 32$, $N = 8$

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



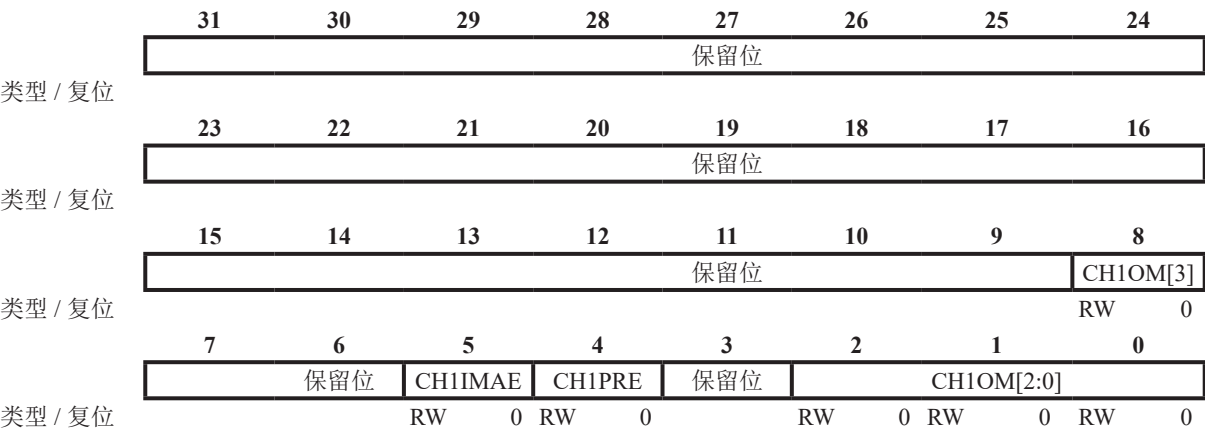
位	域	描述
[5]	CH0IMAE	通道 0 立即有效使能位 0: 无动作 1: 单脉冲立即有效模式使能 无论 CNTR 和 CH0CCR 值的比较结果如何，在一个有效触发事件发生后，CH0OREF 会立即强制为比较匹配电平。 比较匹配电平将维持到下一个上溢或下溢事件的到来。 注：只有当通道 0 被配置工作在 PWM 模式 1 或 PWM 模式 2 时，CH0IMAE 位可用。
[4]	CH0PRE	通道 0 捕捉 / 比较寄存器 (CH0CCR) 预载使能位 0: CH0CCR 预载功能除能 当 CH0PRE 位清零，CH0CCR 寄存器将立即被更新为新的值，且立即可用。 1: CH0CCR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



位	域	描述
[5]	CH1MAE	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，CH1MAE 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	域	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 < 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 > CH1CCR, 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 > CH1CCR, 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 – 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

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	保留位	RW 0	RW 0	RW 0

位	域	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	域	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 < 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									
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							CH3OM[3]		
类型 / 复位								RW	0	
	7	6	5	4	3	2	1	0		
	保留位		CH3IMAE	CH3PRE	保留位	CH3OM[2:0]				
类型 / 复位			RW	0	RW	0	RW	0	RW	0

位	域	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	域	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

	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	CH3E	保留位	CH2E	保留位	CH1E	保留位	CH0E
类型 / 复位	RW	0	RW	0	RW	0	RW	0

位	域	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 = 0x01 / 0x02 / 0x03) 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							
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	CH3P	保留位	CH2P	保留位	CH1P	保留位	CH0P
	RW 0		RW 0		RW 0		RW 0
类型 / 复位							

位	域	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 / 0x3) 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 PDMA / 中断控制寄存器 – DICTR

该寄存器包含了定时器 PDMA 和中断使能控制位。

偏移量: 0x074

复位值: 0x0000_0000

类型 / 复位	31	30	29	28	27	26	25	24
	保留位					TEVDE	保留位	UEVDE
						RW 0		RW 0
类型 / 复位	23	22	21	20	19	18	17	16
	保留位				CH3CCDE	CH2CCDE	CH1CCDE	CH0CCDE
					RW 0	RW 0	RW 0	RW 0
类型 / 复位	15	14	13	12	11	10	9	8
	保留位					TEVIE	保留位	UEVIE
						RW 0		RW 0
类型 / 复位	7	6	5	4	3	2	1	0
	保留位				CH3CCIE	CH2CCIE	CH1CCIE	CH0CCIE
					RW 0	RW 0	RW 0	RW 0

位	域	描述
[26]	TEVDE	触发事件 PDMA 请求使能位 0: 除能触发 PDMA 请求 1: 使能触发 PDMA 请求
[24]	UEVDE	更新事件 PDMA 请求使能位 0: 除能更新事件 PDMA 请求 1: 使能更新事件 PDMA 请求
[19]	CH3CCDE	通道 3 捕捉 / 比较 PDMA 请求使能位 0: 除能通道 3 PDMA 请求 1: 使能通道 3 PDMA 请求
[18]	CH2CCDE	通道 2 捕捉 / 比较 PDMA 请求使能位 0: 除能通道 2 PDMA 请求 1: 使能通道 2 PDMA 请求
[17]	CH1CCDE	通道 1 捕捉 / 比较 PDMA 请求使能位 0: 除能通道 1 PDMA 请求 1: 使能通道 1 PDMA 请求
[16]	CH0CCDE	通道 0 捕捉 / 比较 PDMA 请求使能位 0: 除能通道 0 PDMA 请求 1: 使能通道 0 PDMA 请求
[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

位	域	描述
[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									
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					TEVG	保留位	UEVG		
类型 / 复位						WO	0	WO	0	
	7	6	5	4	3	2	1	0		
	保留位				CH3CCG	CH2CCG	CH1CCG	CH0CCG		
类型 / 复位					WO	0	WO	0	WO	0

位	域	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。

位	域	描述
[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。
[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

	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					TEVIF	保留位	UEVIF	
						W0C 0	W0C 0		
	7	6	5	4	3	2	1	0	
类型 / 复位	CH3OCF	CH2OCF	CH1OCF	CH0OCF	CH3CCIF	CH2CCIF	CH1CCIF	CH0CCIF	
	W0C 0	W0C 0	W0C 0	W0C 0	W0C 0	W0C 0	W0C 0	W0C 0	

位	域	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

位	域	描述
[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：更新事件可在以下情况下发生： – 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。
[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。

位	域	描述
[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
	7	6	5	4	3	2	1	0
类型 / 复位	CNTV							
	RW	0	RW	0	RW	0	RW	0

位	域	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
	7	6	5	4	3	2	1	0	
类型 / 复位	PSCV								
	RW	0	RW	0	RW	0	RW	0	RW

位	域	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
	7	6	5	4	3	2	1	0
	CRV							
类型 / 复位	RW	1	RW	1	RW	1	RW	1

位	域	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	
	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	
	CH0CCV								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
	CH0CCV								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	域	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	
	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	
	CH1CCV								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
	CH1CCV								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	域	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	
	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	
	CH2CCV								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
	CH2CCV								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	域	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	
	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	
	CH3CCV								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
	CH3CCV								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	域	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	
	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
	7	6	5	4	3	2	1	0	
	CH0ACV								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	域	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
	7	6	5	4	3	2	1	0	
	CH1ACV								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	域	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
	7	6	5	4	3	2	1	0	
	CH2ACV								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	域	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
	7	6	5	4	3	2	1	0	
	CH3ACV								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	域	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 脉冲宽度调制定时器 (PWM)

简介

脉冲宽度调制定时器由一个 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。该定时器可用于多种用途, 包括通用计时或产生输出波形, 如单脉冲或 PWM 输出。

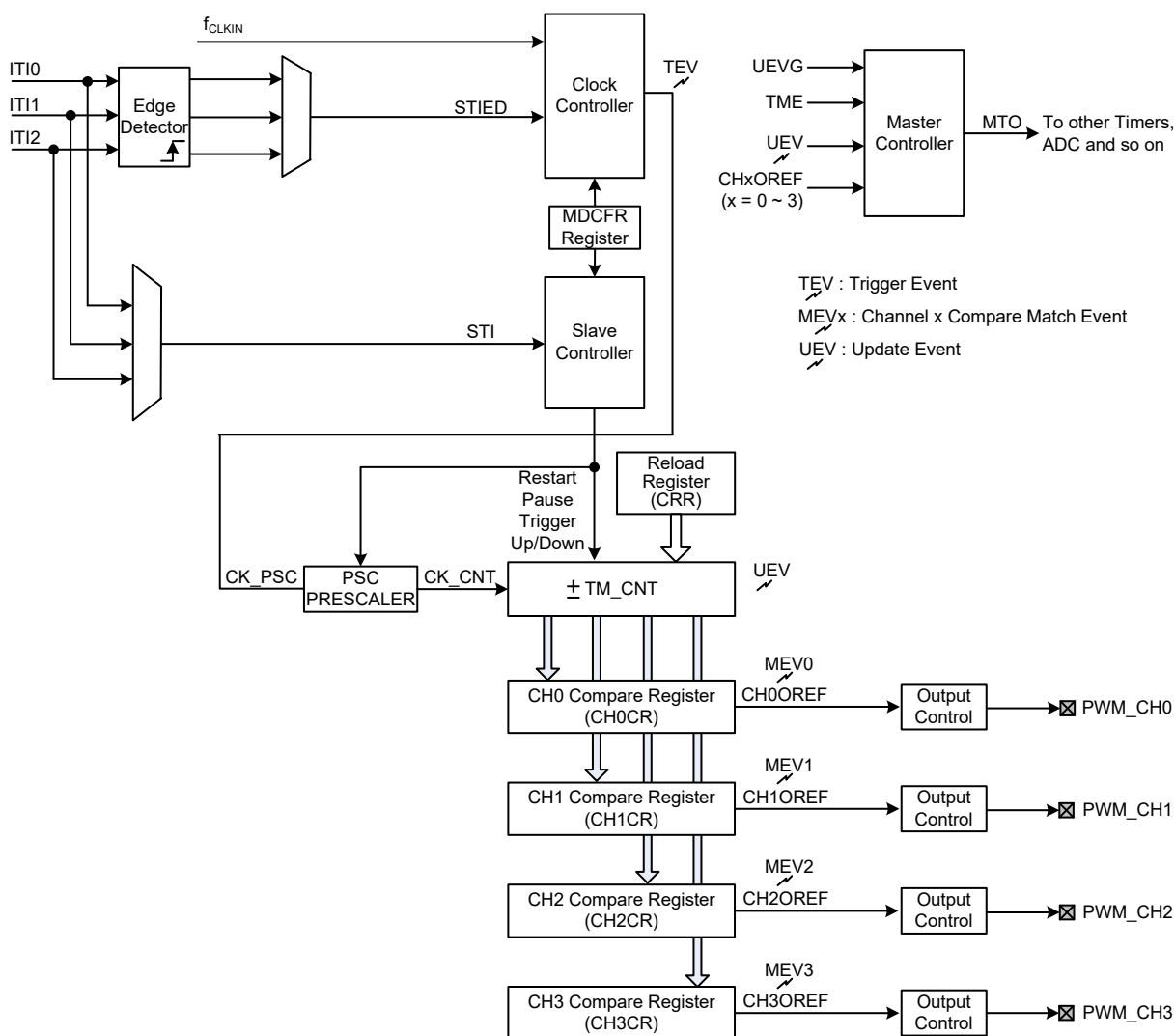


图 68. PWM 方框图

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
- 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 / PDMA 请求：
 - 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。

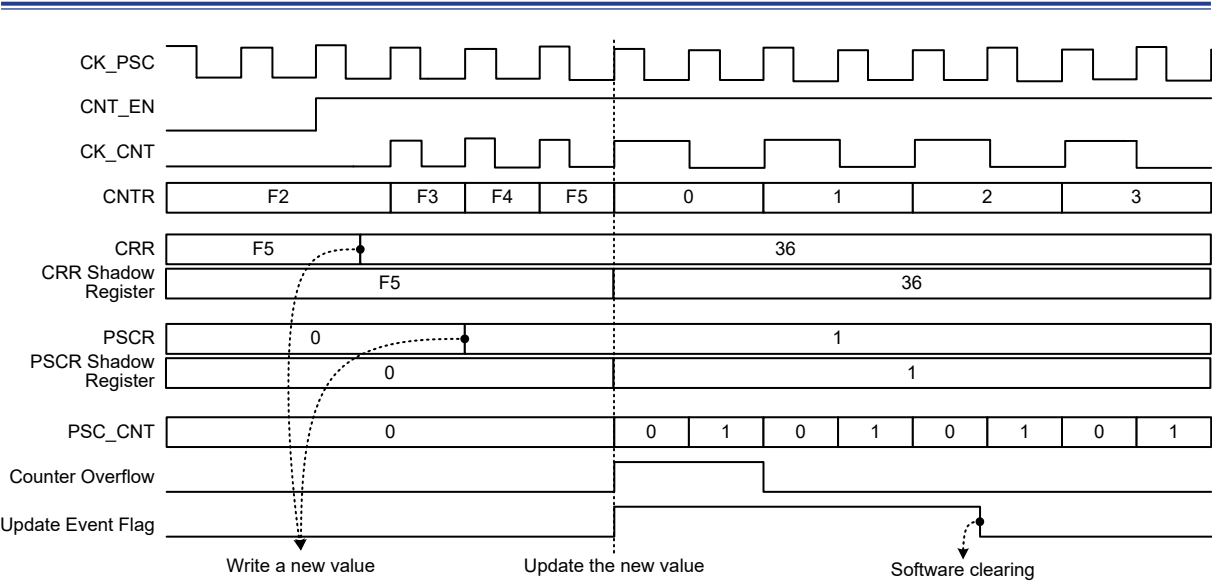


图 69. 向上计数范例

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。

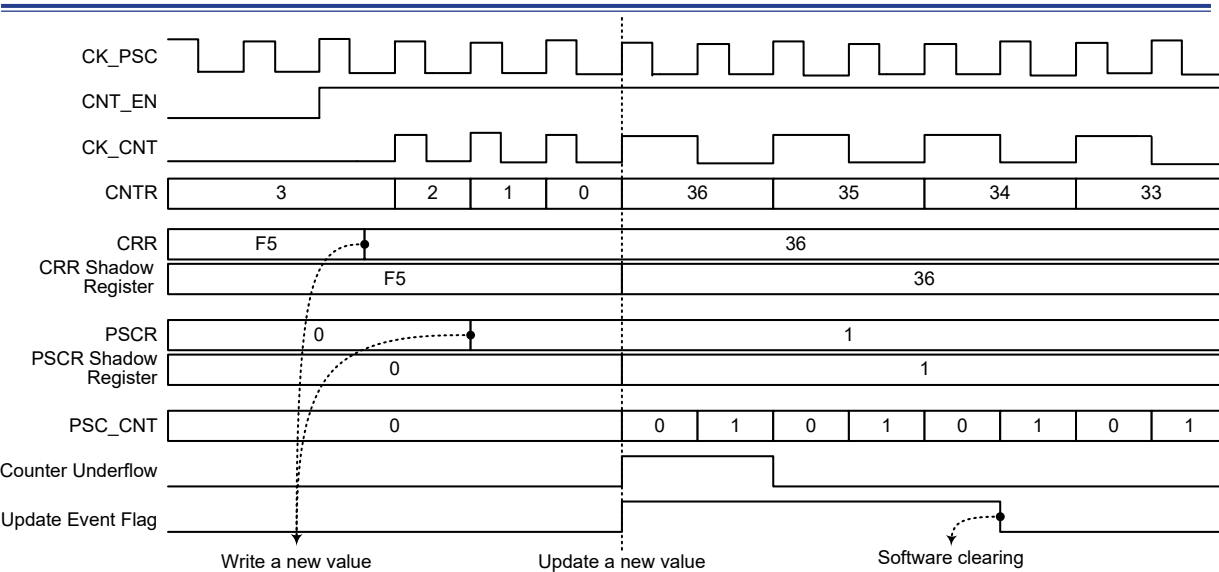


图 70. 向下计数范例

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。

当发生上溢事件或下溢事件，INTSR 寄存器中的更新事件中断标志位 UEVIF 将被置为 1。

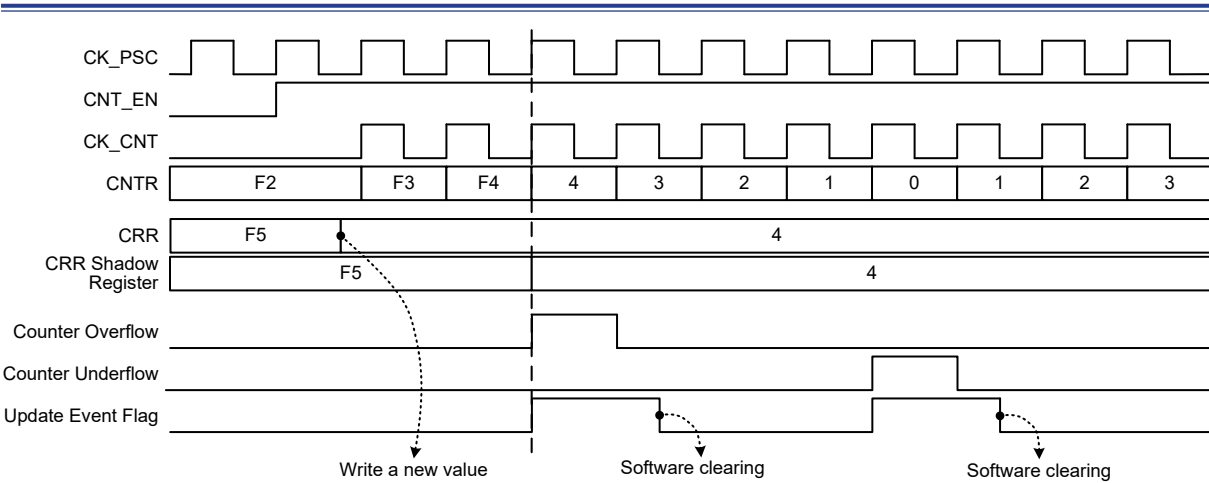


图 71. 中心对齐计数范例

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 信号的上升沿计数。此模式可通过将 MDCFR 寄存器中的 SMSEL 位域设为 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 位域设成除 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 位域设成 0x0 来选择软件 UEVG 位作为触发源，那么当 SMSEL 位域设成 0x7 时，计数器将会被更新而非计数。

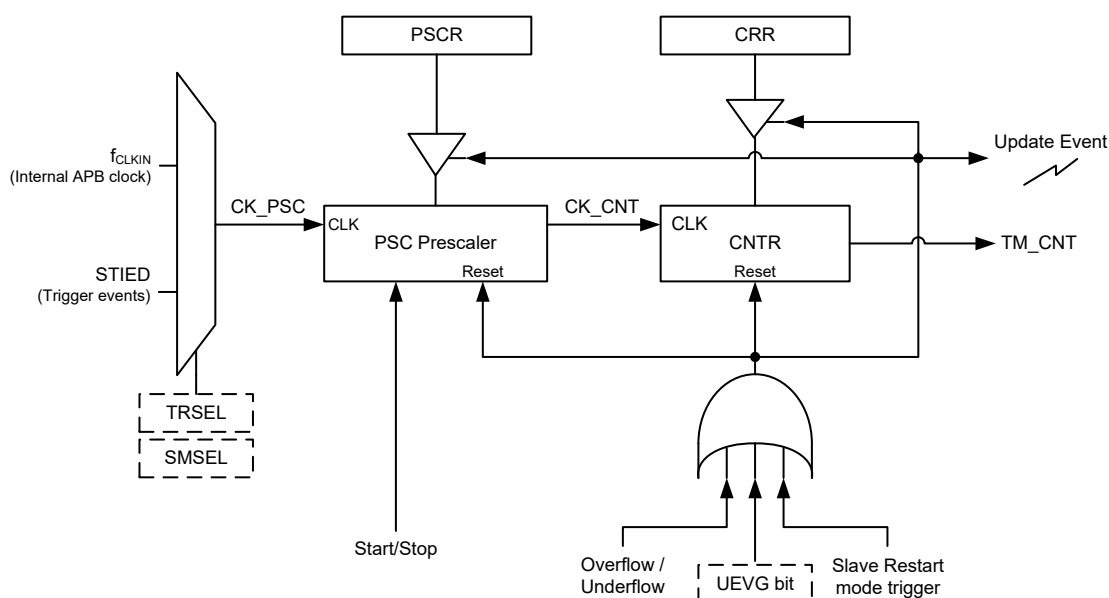


图 72. PWM 时钟源选择

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

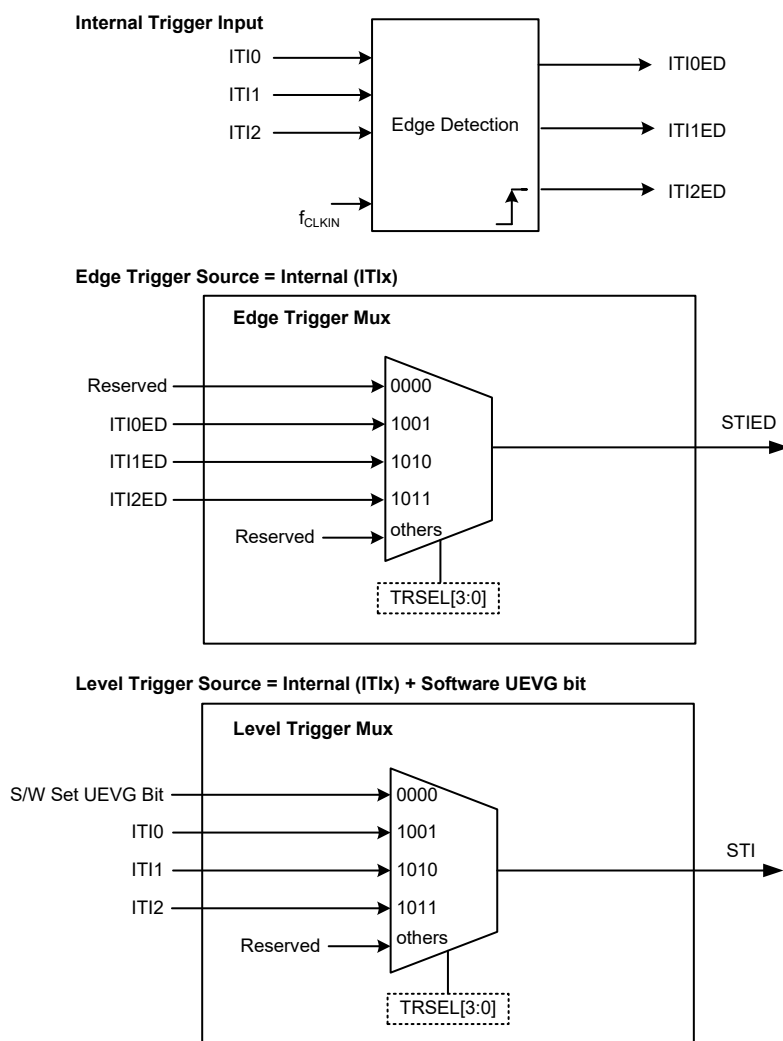


图 73. 触发控制器方框图

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 位域选择的。这些模式的触发输入来自于 STI 信号，通过 TRCFR 寄存器中的 TRSEL 位域选择。从机控制器中的工作模式在相关章节中有所描述。

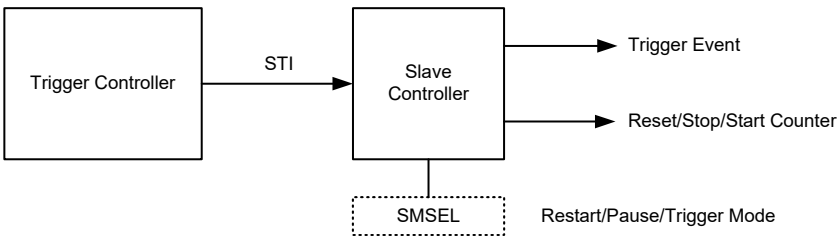


图 74. 从机控制器方框图

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。

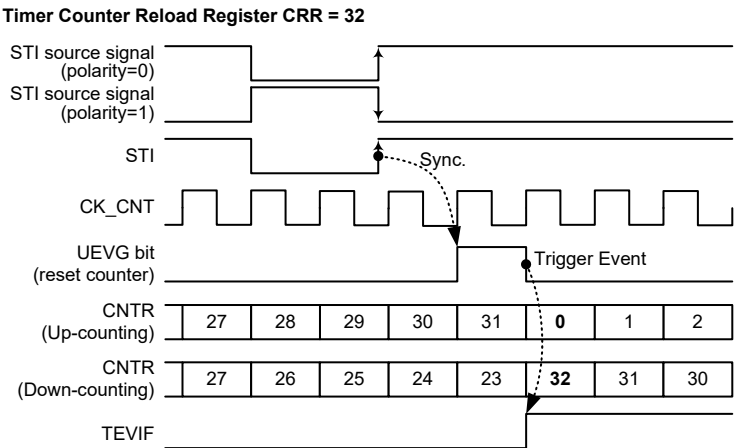


图 75. 重启模式下的 PWM

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，不会被复位。计数器的开始 / 停止操作是由 STI 电平控制的。

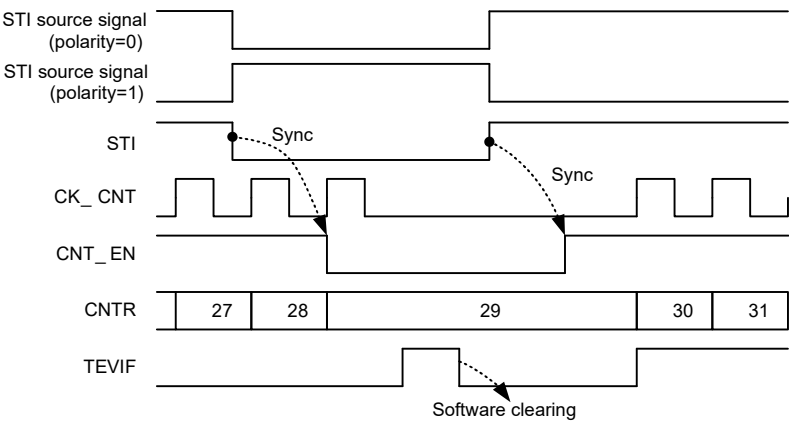


图 76. 暂停模式下的 PWM

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 信号选择来自于 UEVG 位软件触发，计数器不会继续计数。当 STI 源信号通过 UEVG 位选择为软件触发时，不会产生使计数器继续计数的时钟脉冲。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。

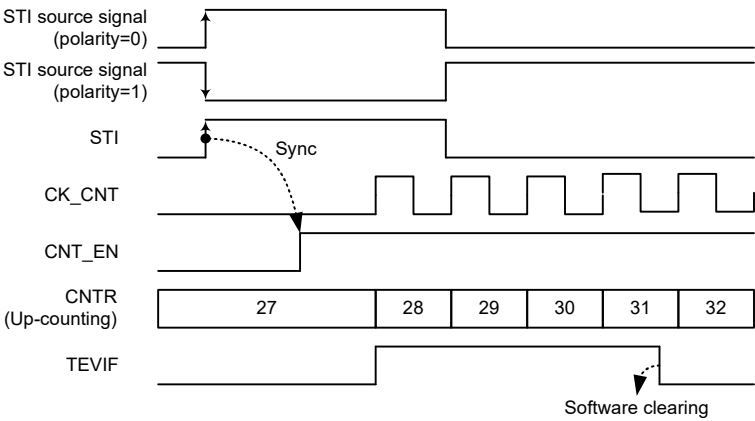


图 77. 触发模式下的 PWM

主机控制器

PWM 和其它 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 位域选择可以触发或驱动处于从机模式下的其它 PWM 或 TM 定时器复位、启动、停止或为其提供时钟源。

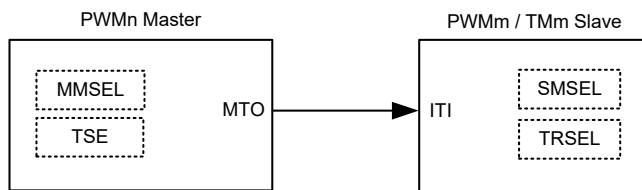


图 78. 主机 PWMn 和从机 PWMm / TMm 相连接

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

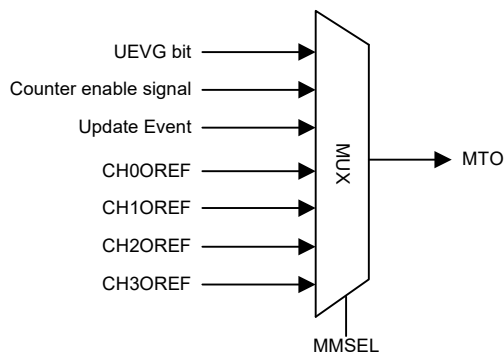


图 79. MTO 选项

例如, 把 MMSEL 位域设为 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 位域定义。

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。

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。

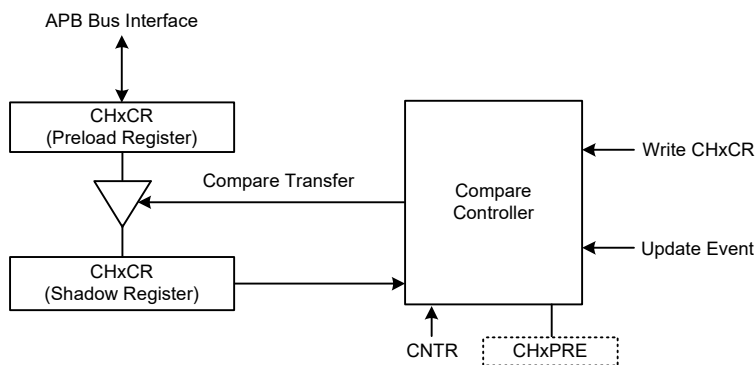


图 80. 比较方框图

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。

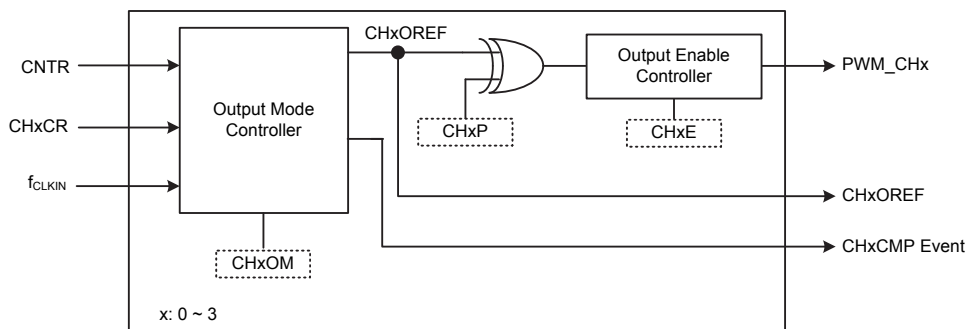


图 81. 输出级方框图

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。CHxOREF 信号有多种输出功能类型，即当计数器的值与 CHxCR 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 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。更多详细说明请参考相应位的定义。输出类型设置如下表所示。

表 35. 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
0xE	非对称 PWM 模式 1
0xF	非对称 PWM 模式 2

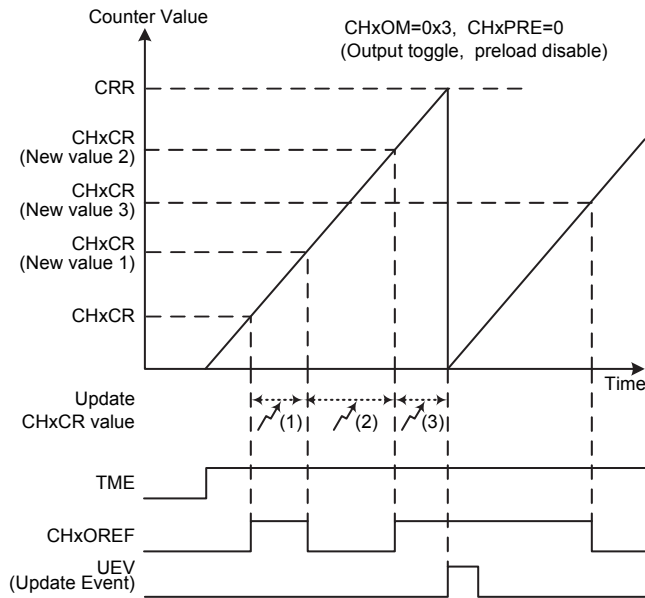


图 82. 翻转模式通道输出参考信号 – CHxPRE = 0

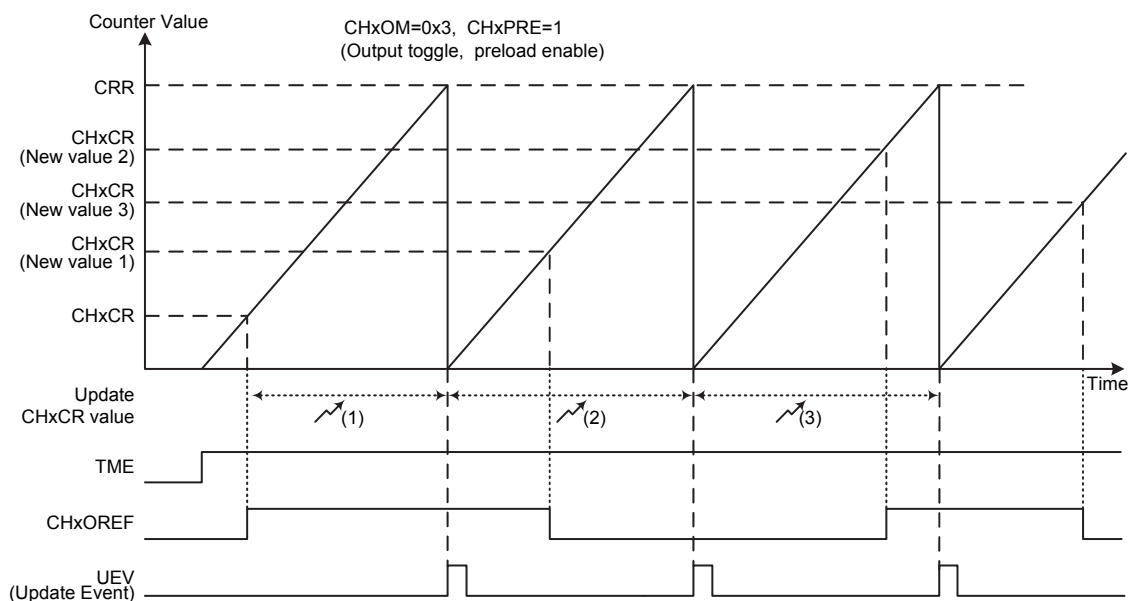


图 83. 翻转模式通道输出参考信号 – CHxPRE = 1

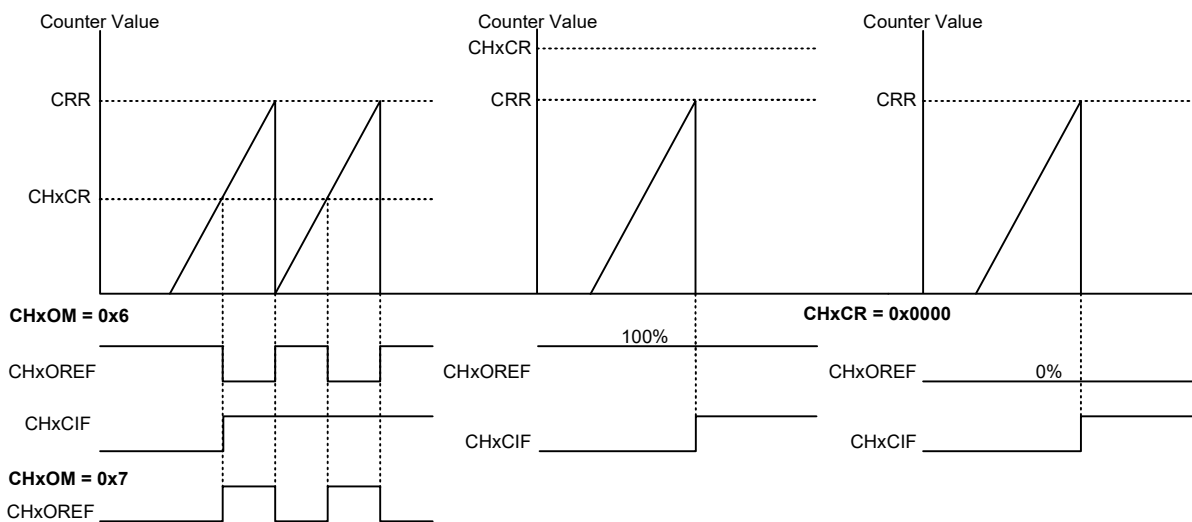


图 84. PWM 模式通道输出参考信号和计数器向上计数模式

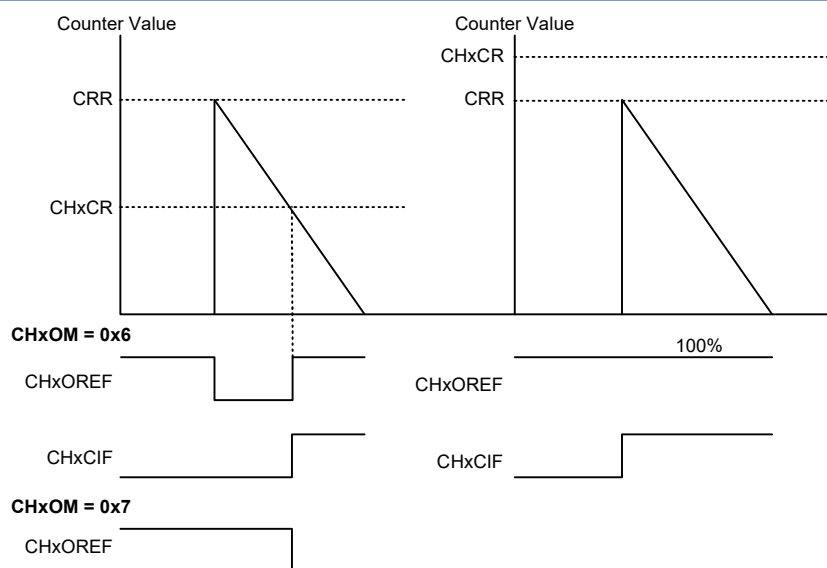


图 85. PWM 模式通道输出参考信号和计数器向下计数模式

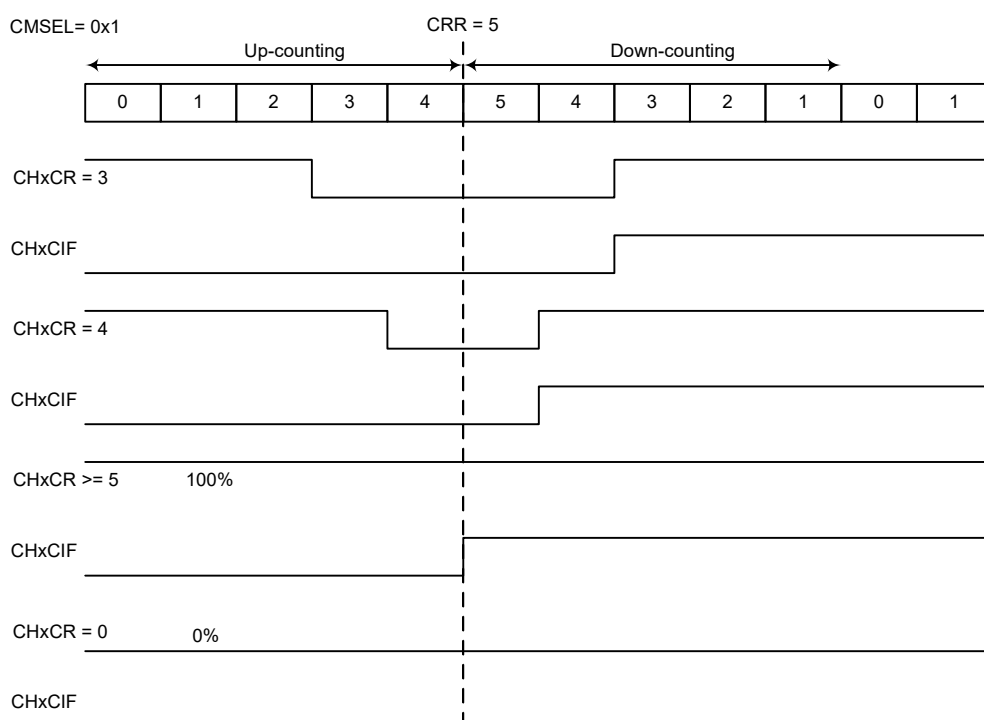


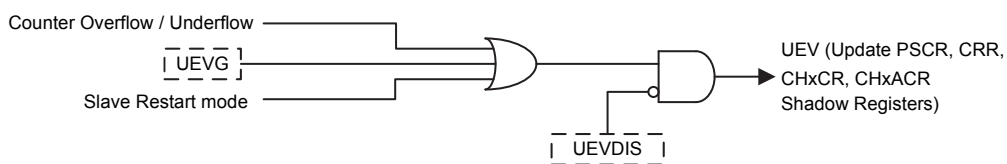
图 86. PWM 模式通道输出参考信号和计数器中心对齐计数模式

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。

Update Event Management



Update Event Interrupt Management

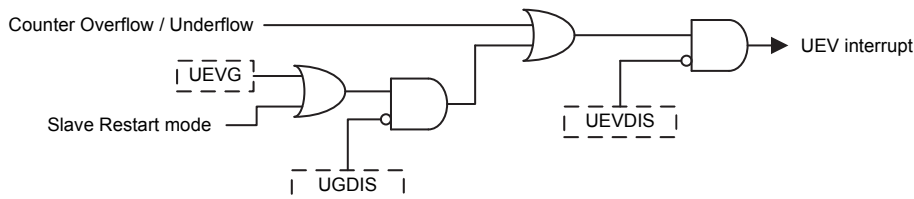


图 87. 更新事件设置方框图

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。

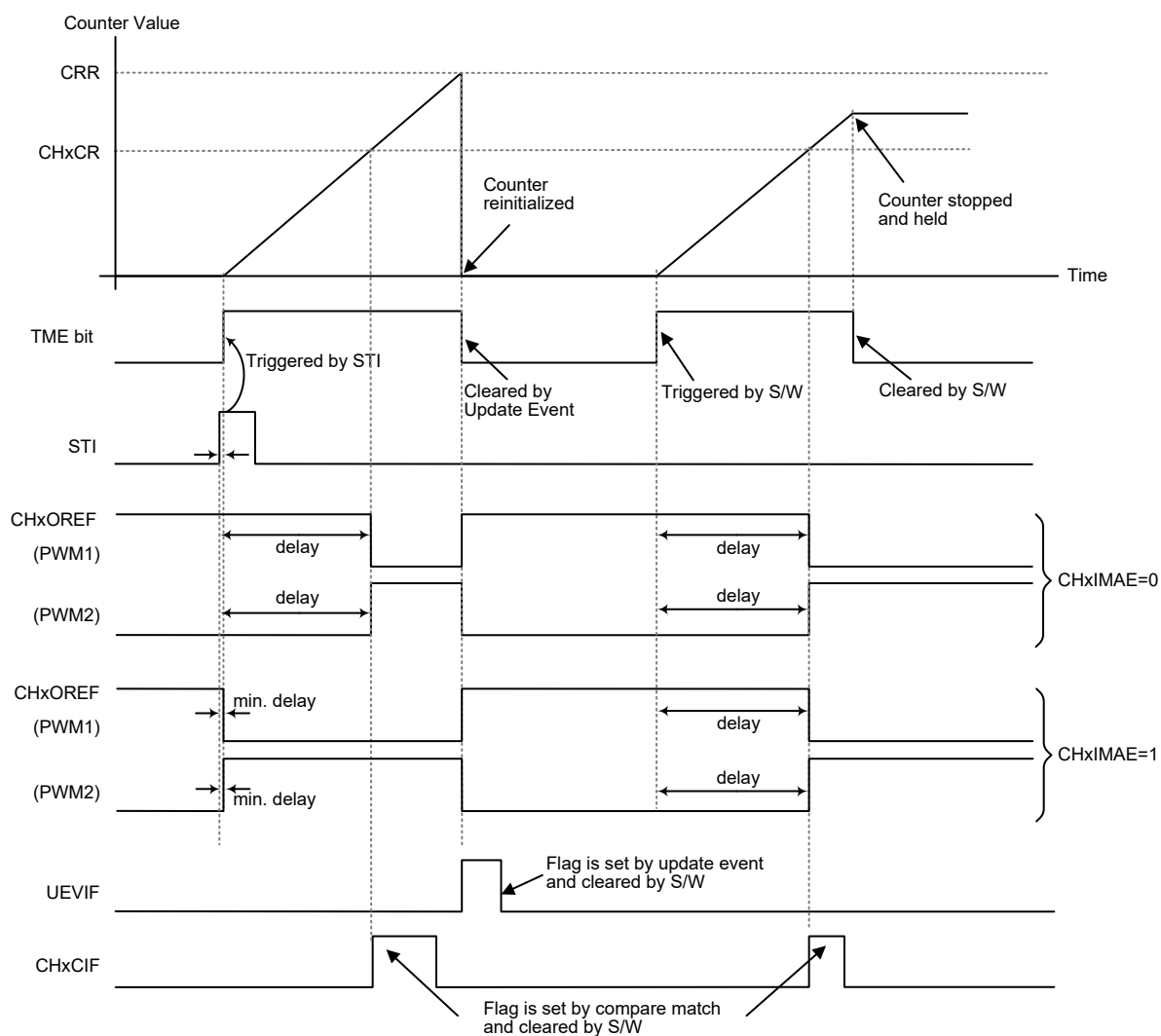


图 88. 单脉冲模式

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 PWM1 或 PWM2 模式下且触发源来自于 STI 信号时, CHxIMAE 位才可用。

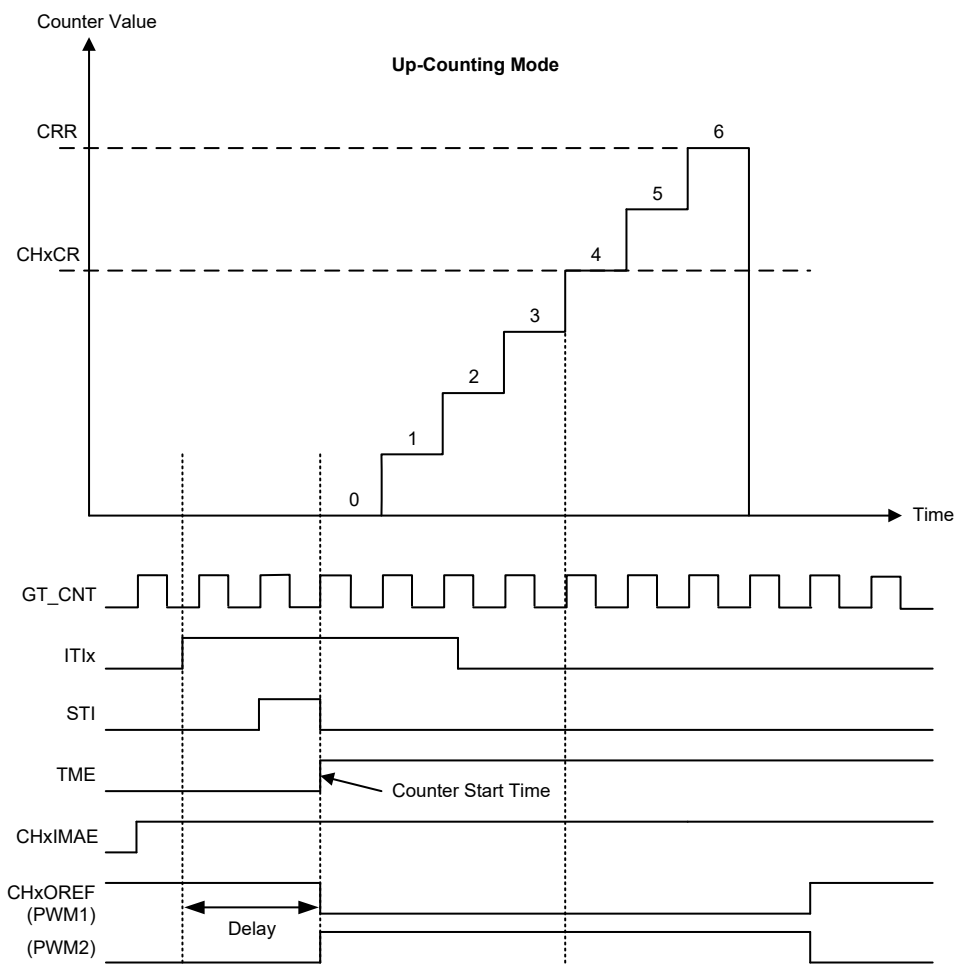


图 89. 立即有效模式的延迟

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。下图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。

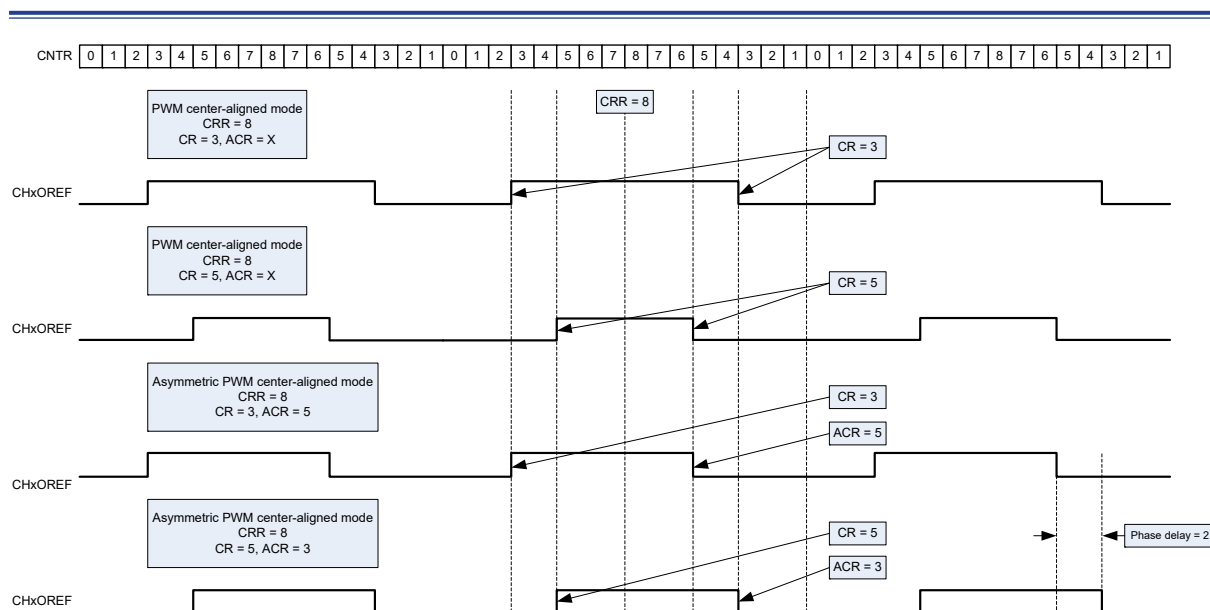


图 90. 非对称 PWM 模式与中心对齐计数模式

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。

使用一个定时器控制另外一个定时器使其开始或停止计数
(主定时器 = PWM0; 从定时器 = PWM1)

- 配置主定时器工作在主机模式，发送通道 0 输出参考信号 CH0OREF 作为触发输出 (MMSEL = 0x4)
- 配置主定时器 CH0OREF 波形
- 配置从定时器，使其接收的输入触发源来自于主定时器触发器输出信号 (TRSEL = 0x9)
- 配置从定时器工作在暂停模式下 (SMSEL = 0x5)
- 通过向从定时器 TME 位写入 1 来使能从定时器
- 通过向主定时器 TME 位写入 1 来使能主定时器

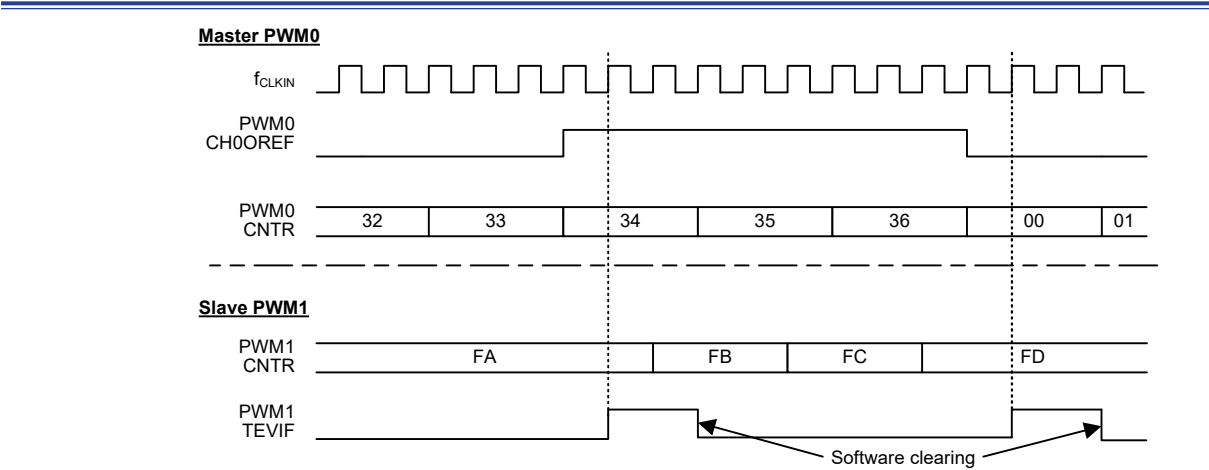


图 91. 用 PWM0 CH0OREF 信号暂停 PWM1

使用一个定时器触发另一个定时器开始计数 (主定时器 = PWM0; 从定时器 = PWM1)

- 配置主定时器工作在主机模式，发送更新事件 (UEV) 作为触发输出 (MMSEL = 0x2)
- 通过设置 CRR 寄存器配置主定时器周期
- 配置从定时器，使其接收的输入触发源来自主定时器触发输出信号 (TRSEL = 0x9)
- 配置从定时器工作在从机触发模式 (SMSEL = 0x6)
- 向主定时器 TME 位写入 1 启动主定时器

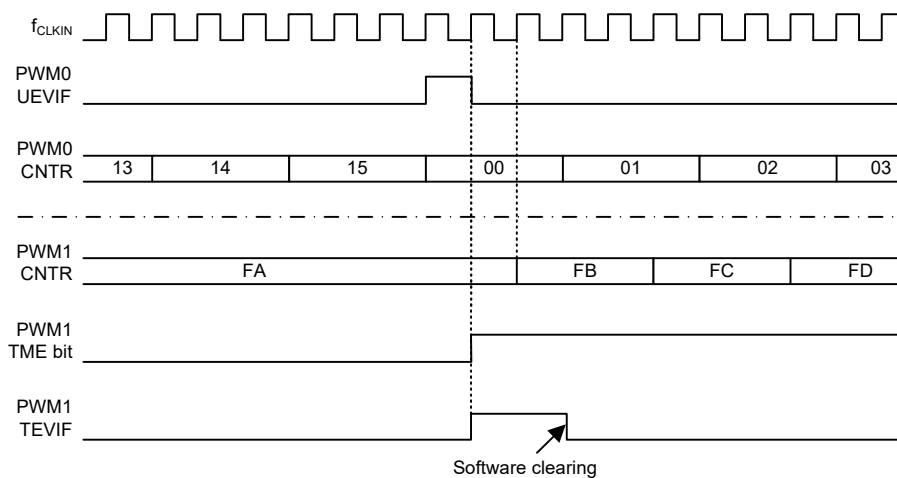


图 92. 用 PWM0 更新事件触发 PWM1

通过主定时器使能信号触发同时启动两个定时器 (主定时器 = PWM0; 从定时器 = PWM1)

- 配置主定时器工作在主机模式下，发送使能信号作为触发输出信号 (MMSEL = 0x1)
- 设置 MDCFR 寄存器中的 TSE 位为 1，使能主定时器主机定时器同步功能使其与从机定时器同步
- 配置从定时器，使其接收的输入触发源来自于主定时器的触发输出信号 (TRSEL = 0x9)
- 配置从定时器工作在从机触发模式下 (SMSEL = 0x6)
- 向主定时器 TME 位写入 1，启动主定时器

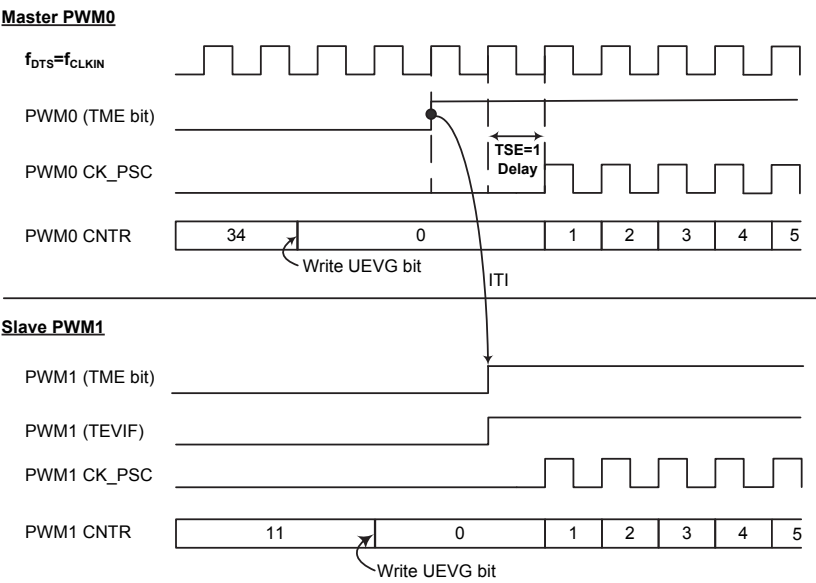


图 93. 用 PWM0 使能信号触发 PWM0 和 PWM1

触发其它外设功能启动

为了与其它外设功能 (如 ADC、TM 等) 相连接, PWM 可以输出 MTO 信号或通道比较匹配输出信号 CHxOREF (x = 0 ~ 3) 作为外设的输入触发信号, 取决于 MCU 的规格。

PDMA 请求

PWM 提供了 PDMA 数据传输接口。PWM 更新事件, 触发事件和通道比较事件, 可产生 PDMA 请求, 如果其相应的使能控制位置 1 则可使能 PDMA 存取。当 PWM 通道产生 PDMA 请求时, 可通过通道 PDMA 选择位 CHCDS 选择是来自通道比较事件还是 PWM 更新事件。更多 PDMA 配置信息, 请参考 PDMA 章节。

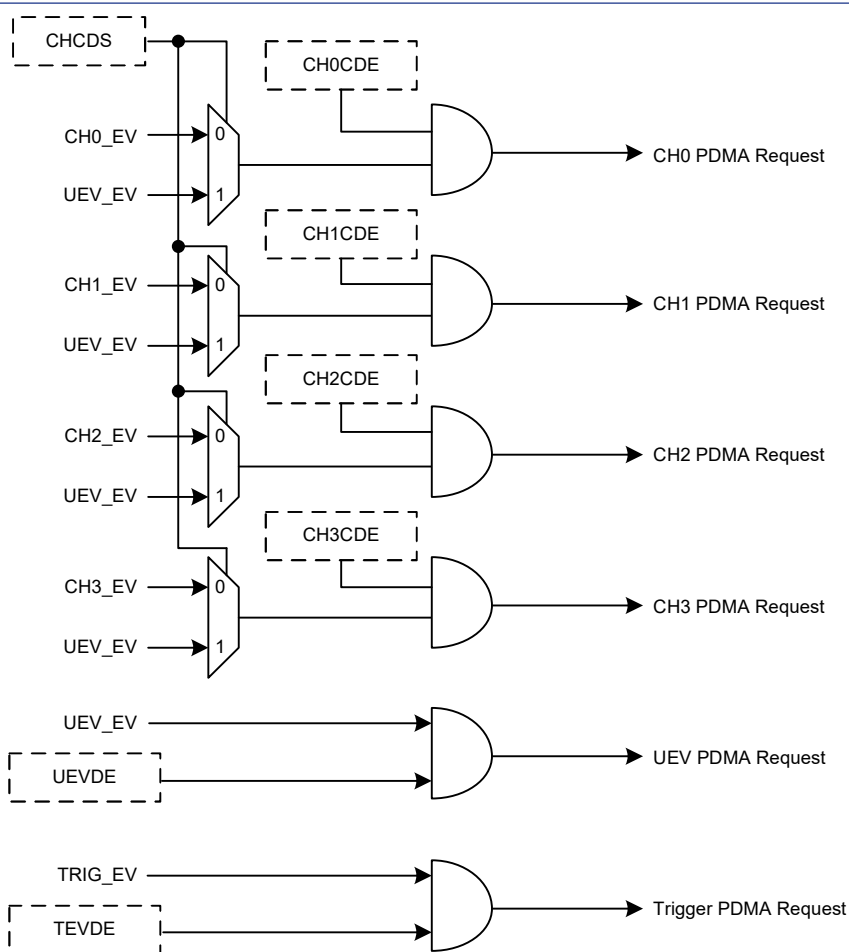


图 94. PWM PDMA 映射图

寄存器列表

下表显示了 PWM 寄存器及其复位值。

表 36. 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 PDMA / 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	域	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，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 PDMA 请求或中断 – 计数器上溢 / 下溢 – 设置 UEVG 位为 1 – 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 PDMA 请求或中断
[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 位为 1 – 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 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
	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	域	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 主机模式选项用来选择与其它从机定时器同步的 MTO 信号源。 <table><tr><th>MMSEL[2:0]</th><th>模式</th><th>描述</th></tr><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>—</td><td>保留</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></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	—	保留	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	—	保留																											
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。																											

位

域

描述

[10:8]

SMSEL

从机模式选项

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	—	保留
010	—	保留
011	—	保留
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，计数器从当前值开始计数。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

定时器同步使能

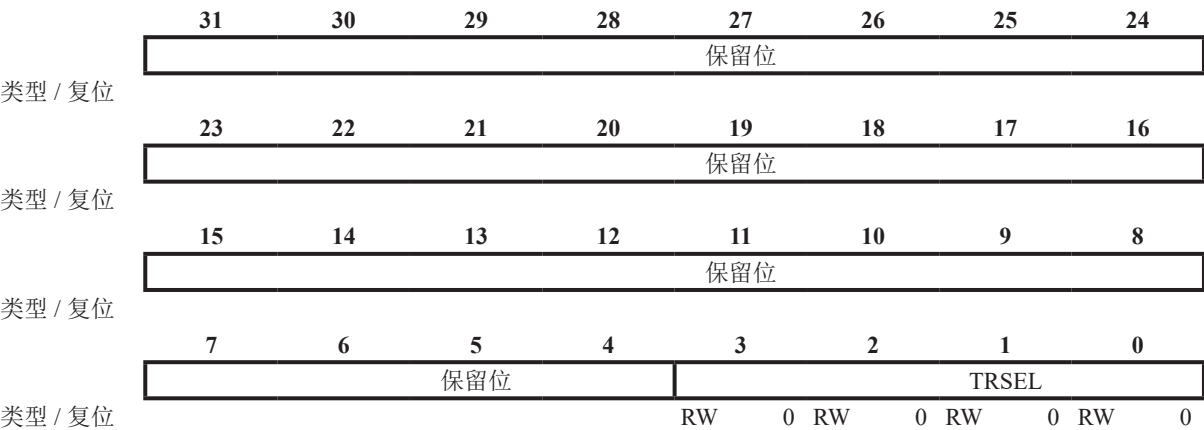
0: 无动作

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	域	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) 其它值: 保留 注: 当 SMSEL 位域为 0x0 除能从机模式时, 这些位才能被更新。

表 37. 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)、CRR 缓冲器使能位 (CRBE) 和通道 PDMA 选择位 (CHCDS)。

偏移量: 0x010
复位值: 0x0000_0000

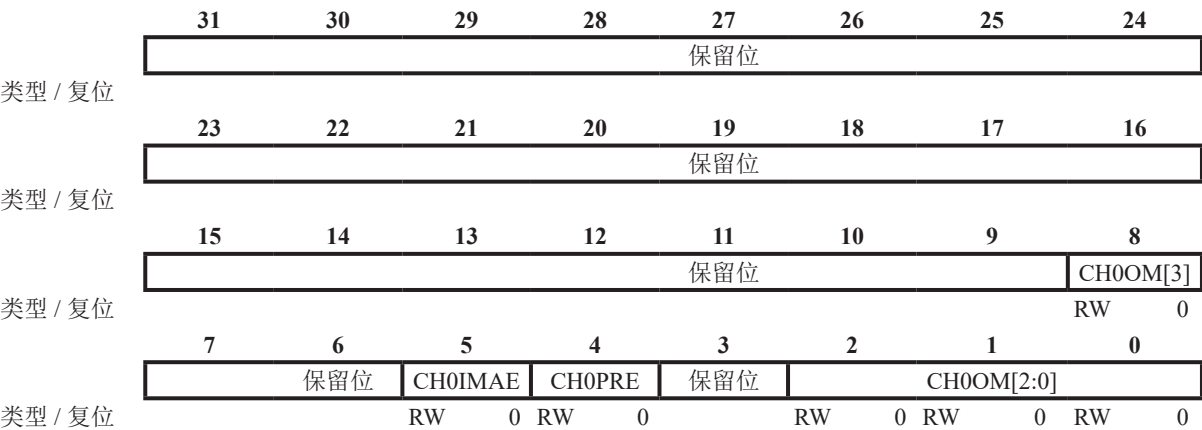
类型 / 复位	31	30	29	28	27	26	25	24
	保留位							
类型 / 复位	23	22	21	20	19	18	17	16
	保留位							CHCDS
类型 / 复位	15	14	13	12	11	10	9	8
	保留位							
类型 / 复位	7	6	5	4	3	2	1	0
	保留位						CRBE	TME
							RW	0
							RW	0

位	域	描述
[16]	CHCDS	通道 PDMA 事件选择 0: 通道 PDMA 请求来自通道比较事件 1: 通道 PDMA 请求来自更新事件
[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



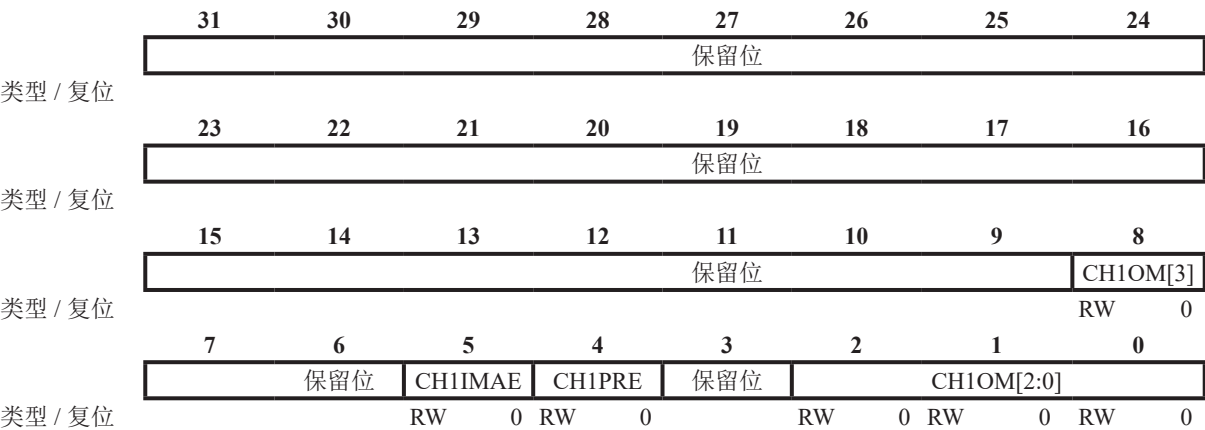
位	域	描述
[5]	CH0IMAE	通道 0 立即有效使能位 0: 无动作 1: 单脉冲立即有效模式使能 无论 CNTR 和 CH0CR 值的比较结果如何，在一个有效触发事件发生后，CH0OREF 会立即强制为比较匹配电平。 比较匹配电平将维持到下一个上溢或下溢事件的到来。 注：只有当通道 0 被配置工作在 PWM 模式 1 或 PWM 模式 2 时，CH0IMAE 位可用。
[4]	CH0PRE	通道 0 比较寄存器 (CH0CR) 预载使能位 0: CH0CR 预载功能除能 当 CH0PRE 位清零，CH0CR 寄存器将立即被更新为新的值，且立即可用。 1: CH0CR 预载功能使能 直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

位	域	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



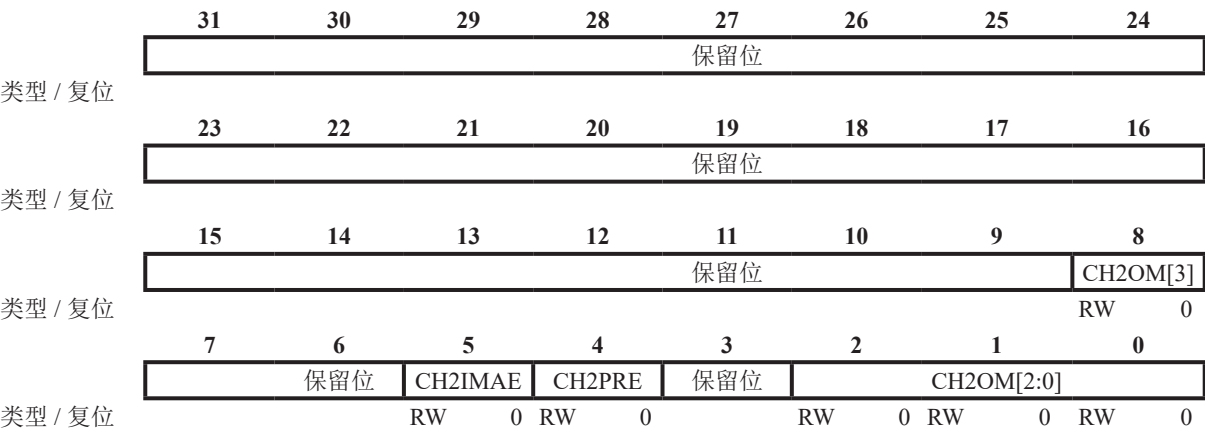
位	域	描述
[5]	CH1IMAE	通道 1 立即有效使能位 0: 无动作 1: 单脉冲立即有效模式使能 无论 CNTR 和 CH1CR 值的比较结果如何，在一个有效触发事件发生后，CH1OREF 会立即强制为比较匹配电平。 比较匹配电平将维持到下一个上溢或下溢事件的到来。 注：只有当通道 1 被配置工作在 PWM 模式 1 或 PWM 模式 2 时，CH1IMAE 位可用。
[4]	CH1PRE	通道 1 比较寄存器 (CH1CR) 预载使能位 0: CH1CR 预载功能除能 当 CH1PRE 位清零，CH1CR 寄存器将立即被更新为新的值，且立即可用。 1: CH1CR 预载功能使能 直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

位	域	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 输出配置寄存器 – 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



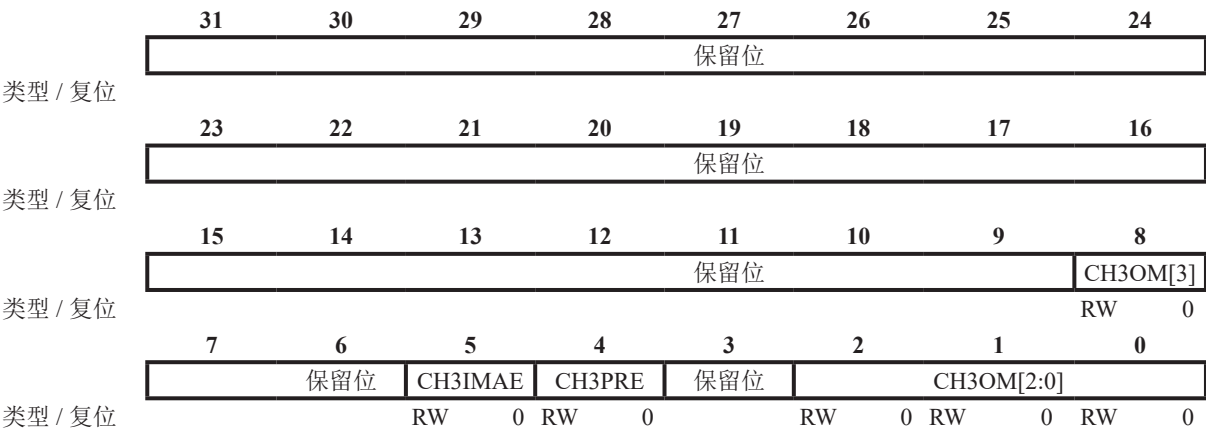
位	域	描述
[5]	CH2IMAE	通道 2 立即有效使能位 0: 无动作 1: 单脉冲立即有效模式使能 无论 CNTR 和 CH2CR 值的比较结果如何，在一个有效触发事件发生后，CH2OREF 会立即强制为比较匹配电平。 比较匹配电平将维持到下一个上溢或下溢事件的到来。 注：只有当通道 2 被配置工作在 PWM 模式 1 或 PWM 模式 2 时，CH2IMAE 位可用。
[4]	CH2PRE	通道 2 比较寄存器 (CH2CR) 预载使能位 0: CH2CR 预载功能除能 当 CH2PRE 位清零，CH2CR 寄存器将立即被更新为新的值，且立即可用。 1: CH2CR 预载功能使能 直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

位	域	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



位	域	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	域	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

类型 / 复位	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	CH3E	保留位	CH2E	保留位	CH1E	保留位	CH0E
	RW	0	RW	0	RW	0	RW	0

位	域	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

类型 / 复位	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	CH3P	保留位	CH2P	保留位	CH1P	保留位	CH0P
	RW	0	RW	0	RW	0	RW	0

位	域	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 PDMA / 中断控制寄存器 – DICTR

该寄存器包含了定时器 PDMA 和中断使能控制位。

偏移量: 0x074

复位值: 0x0000_0000

类型 / 复位	31	30	29	28	27	26	25	24
	保留位					TEVDE	保留位	UEVDE
						RW 0		RW 0
类型 / 复位	23	22	21	20	19	18	17	16
	保留位				CH3CDE	CH2CDE	CH1CDE	CH0CDE
					RW 0	RW 0	RW 0	RW 0
类型 / 复位	15	14	13	12	11	10	9	8
	保留位					TEVIE	保留位	UEVIE
						RW 0		RW 0
类型 / 复位	7	6	5	4	3	2	1	0
	保留位				CH3CIE	CH2CIE	CH1CIE	CH0CIE
					RW 0	RW 0	RW 0	RW 0

位	域	描述
[26]	TEVDE	触发事件 PDMA 请求使能位 0: 除能触发 PDMA 请求 1: 使能触发 PDMA 请求
[24]	UEVDE	更新事件 PDMA 请求使能位 0: 除能更新事件 PDMA 请求 1: 使能更新事件 PDMA 请求
[19]	CH3CDE	通道 3 比较 PDMA 请求使能位 0: 除能通道 3 PDMA 请求 1: 使能通道 3 PDMA 请求
[18]	CH2CDE	通道 2 比较 PDMA 请求使能位 0: 除能通道 2 PDMA 请求 1: 使能通道 2 PDMA 请求
[17]	CH1CDE	通道 1 比较 PDMA 请求使能位 0: 除能通道 1 PDMA 请求 1: 使能通道 1 PDMA 请求
[16]	CH0CDE	通道 0 比较 PDMA 请求使能位 0: 除能通道 0 PDMA 请求 1: 使能通道 0 PDMA 请求
[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

位	域	描述
[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									
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					TEVG	保留位	UEVG		
类型 / 复位						WO	0	WO	0	
	7	6	5	4	3	2	1	0		
	保留位				CH3CG	CH2CG	CH1CG	CH0CG		
类型 / 复位					WO	0	WO	0	WO	0

位	域	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]	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

位	域	描述
[1]	CH1CG	通道 1 比较发生 通道 1 捕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							
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					TEVIF	保留位	UEVIF
类型 / 复位					W0C 0	保留位	W0C 0
7	6	5	4	3	2	1	0
保留位				CH3CIF	CH2CIF	CH1CIF	CH0CIF
类型 / 复位				W0C 0	W0C 0	W0C 0	W0C 0

位	域	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
[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。

位	域	描述
[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。
[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

	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	域	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
	7	6	5	4	3	2	1	0
	PSCV							
类型 / 复位	RW	0	RW	0	RW	0	RW	0

位	域	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
	7	6	5	4	3	2	1	0
	CRV							
类型 / 复位	RW	1	RW	1	RW	1	RW	1

位	域	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 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
	7	6	5	4	3	2	1	0	
	CH0CV								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	域	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
	7	6	5	4	3	2	1	0	
	CH1CV								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	域	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

该寄存器定义了定时器通道 2 比较值。

偏移量: 0x098
复位值: 0x0000_0000

类型 / 复位	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
	CH2CV							
类型 / 复位	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

位	域	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

该寄存器定义了定时器通道 3 比较值。

偏移量: 0x09C
复位值: 0x0000_0000

类型 / 复位	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
	CH3CV							
类型 / 复位	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

位	域	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	
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	域	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
	7	6	5	4	3	2	1	0	
	CH1ACV								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	域	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
	7	6	5	4	3	2	1	0	
	CH2ACV								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	域	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
	7	6	5	4	3	2	1	0	
	CH3ACV								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	域	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 单通道定时器 (SCTM)

简介

单通道定时器由一个 16-bit 向上计数器、一个 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。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 PWM 输出。

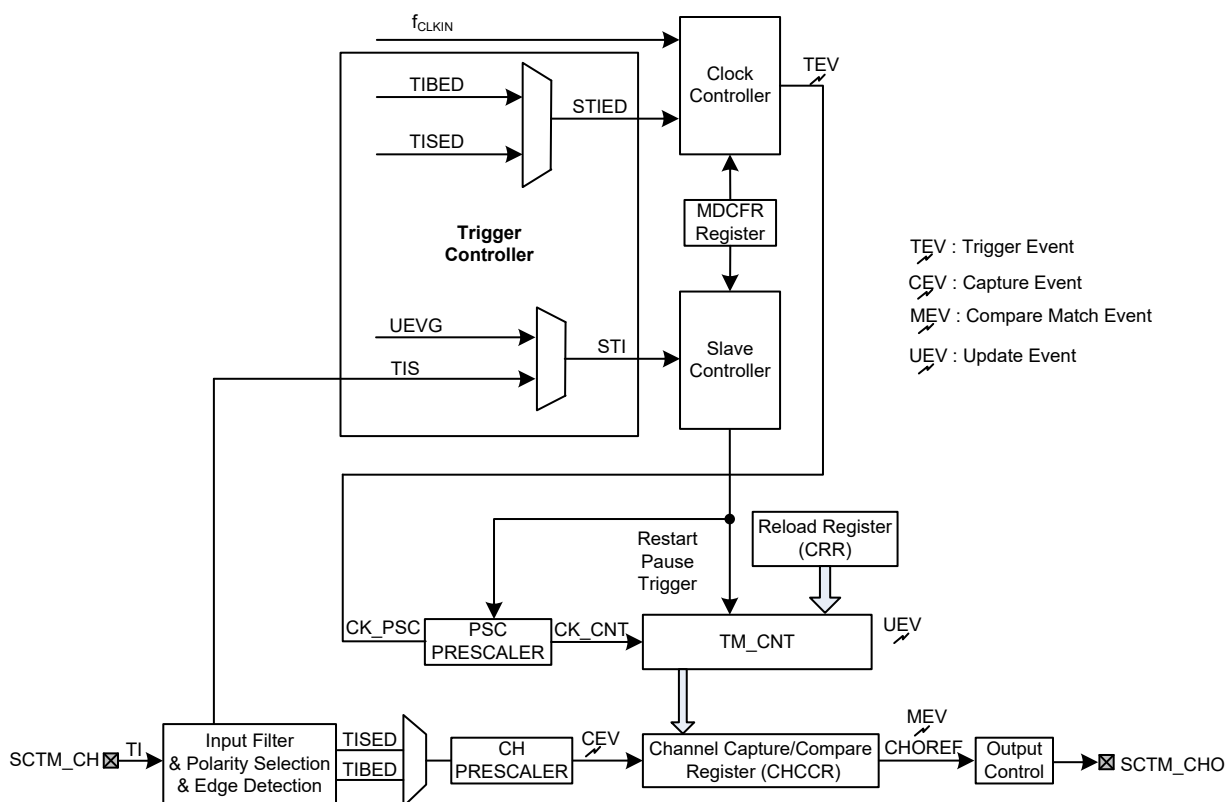


图 95. SCTM 方框图

特性

- 16-bit 自动重载向上计数器
- 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
- 单通道用于：
 - 输入捕捉功能
 - 比较匹配输出
 - PWM 波形生成
- 下列事件发生时将产生中断：
 - 更新事件
 - 触发事件
 - 输入捕捉事件
 - 输出比较匹配事件

功能描述

计数器模式

向上计数

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，接着从 0 开始重新计数并产生一个溢出事件。这一动作会反复执行。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。

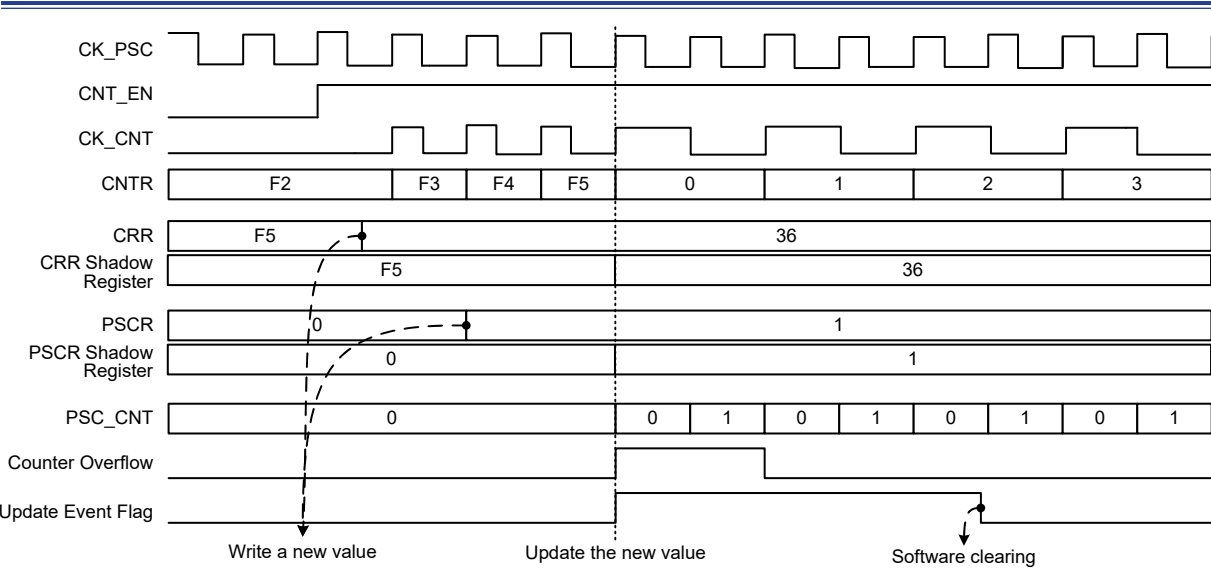


图 96. 向上计数范例

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。如果通过设置 MDCFR 寄存器中的 SMSEL 位域为 0x7 来使能从机模式控制器，那么预分频器的时钟将由 TRCFR 寄存器中的 TRSEL 位域设置为其它时钟源，具体描述如下。

■ 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 位域设为 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 位域设成除 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 位域设成 0x0 来选择软件 UEVG 位作为触发源，那么当 SMSEL 位域设成 0x7 时，计数器将会被更新而非计数。

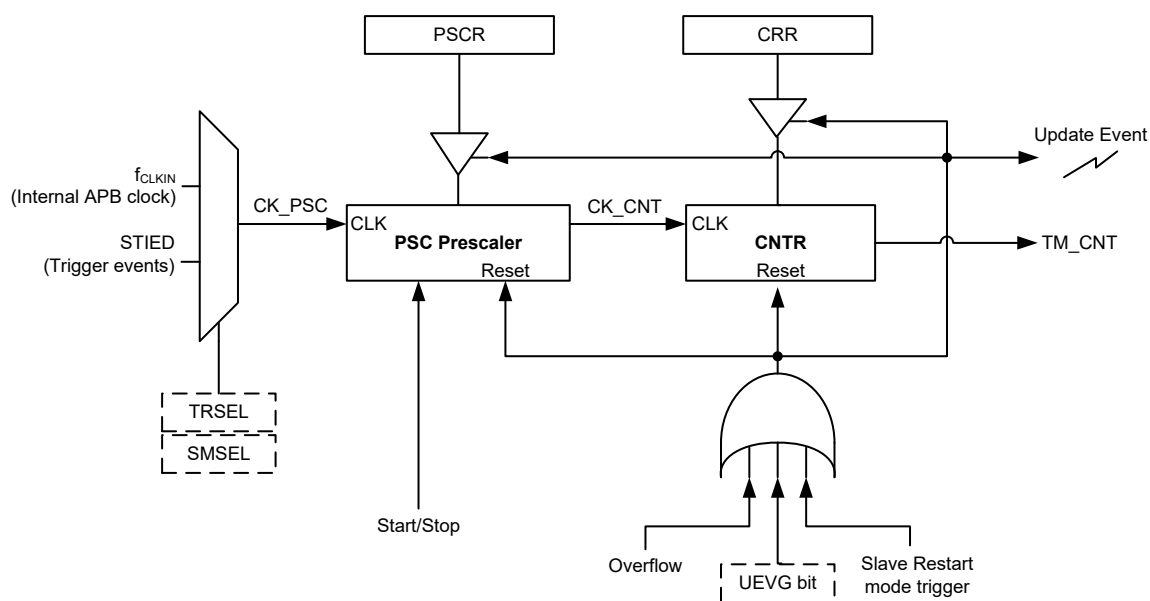
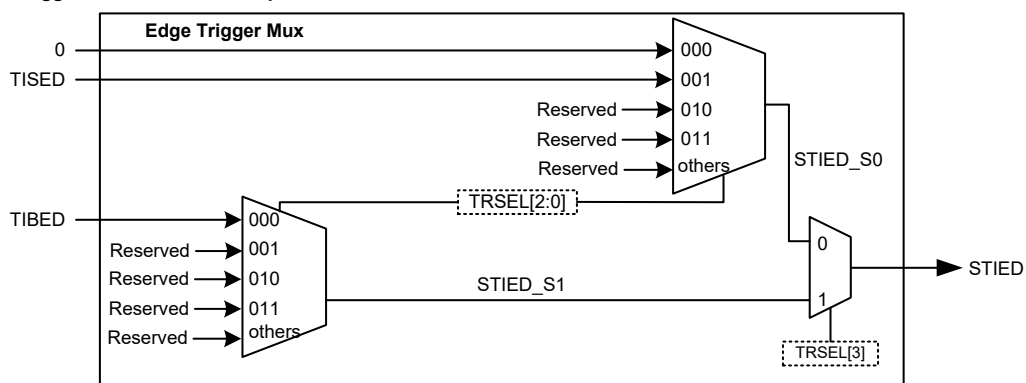


图 97. SCTM 时钟源选择

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 SCTM 功能。

Edge Trigger Source = Channel input



Level Trigger Source = Channel input + Software UEVG bit

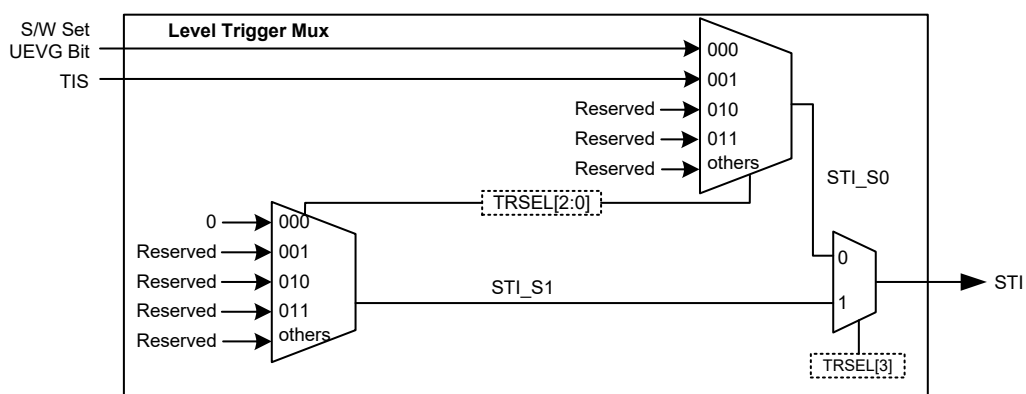


图 98. 触发控制器方框图

从机控制器

在几种模式下，SCTM 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 位域选择的。这些模式的触发输入来自于 STI 信号，通过 TRCFR 寄存器中的 TRSEL 位域选择。从机控制器中的工作模式在相关章节中有所描述。

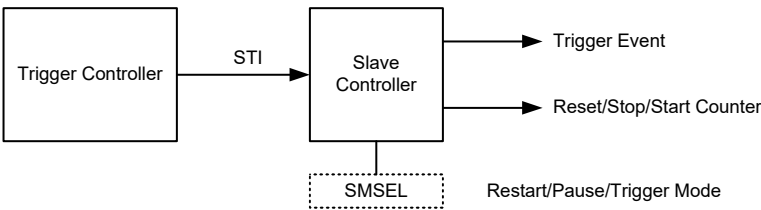


图 99. 从机控制器方框图

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。

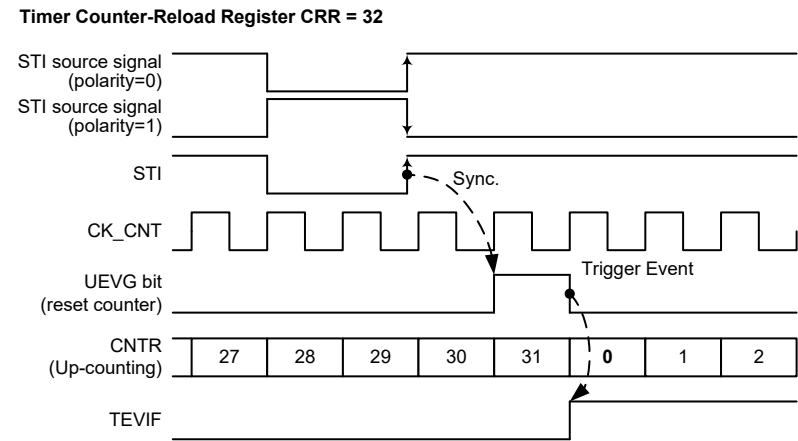


图 100. 重启模式下的 SCTM

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。由于计数器的开始 / 停止操作是由 STI 电平控制的，所以所选择的 STI 信号不能来自于 TIBED。

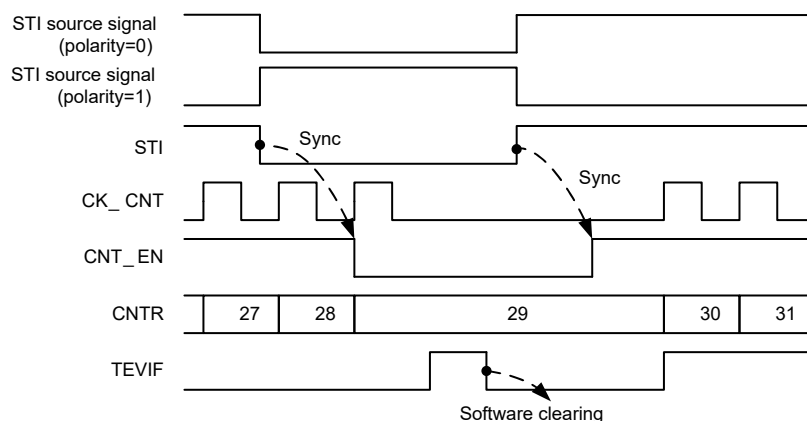


图 101. 暂停模式下的 SCTM

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，计数器不会恢复计数。当 STI 源信号通过 UEVG 位选择为软件触发时，不会产生使计数器恢复计数的时钟脉冲。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。

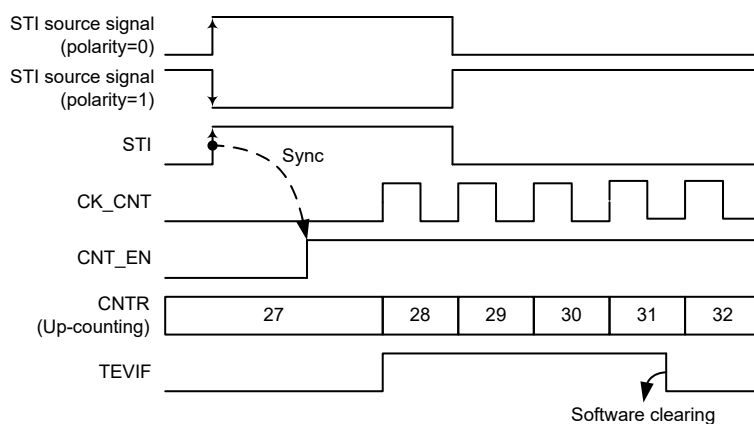


图 102. 触发模式下的 SCTM

通道控制器

SCTM 通道可以用来选择作为捕捉输入或比较匹配输出。捕捉输入或比较匹配输出通道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 CHCCR 影子寄存器中，其值再被传送到 CHCCR 预载寄存器中。

当工作在比较匹配输出模式，CHCCR 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。

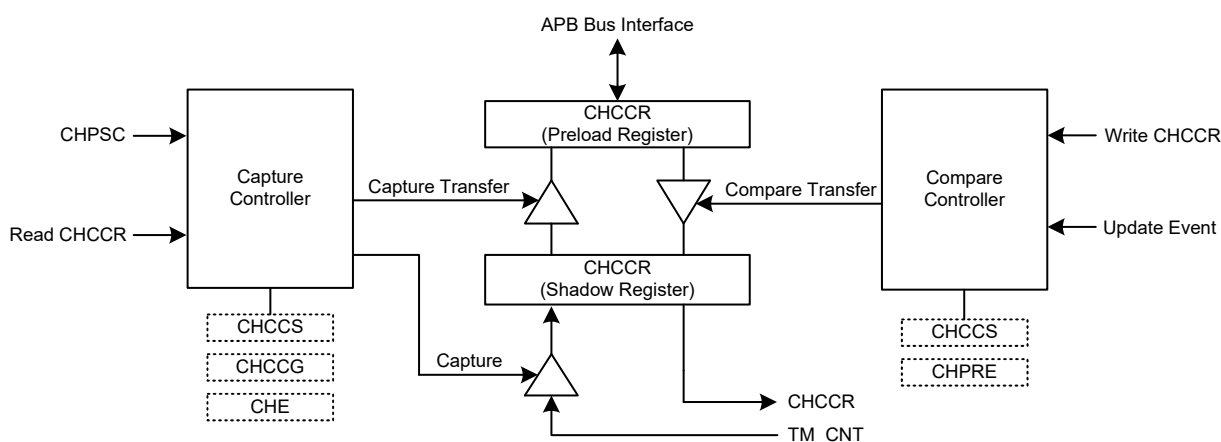


图 103. 捕捉 / 比较方框图

捕捉计数器值传送到 CHCCR

当通道用作捕捉输入时，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 (CHCCR) 中。一旦捕捉事件发生，INTSR 寄存器中的 CHCCIF 标志位会相应的被置位。如果 CHCCIF 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 CHOCF 将被置位。

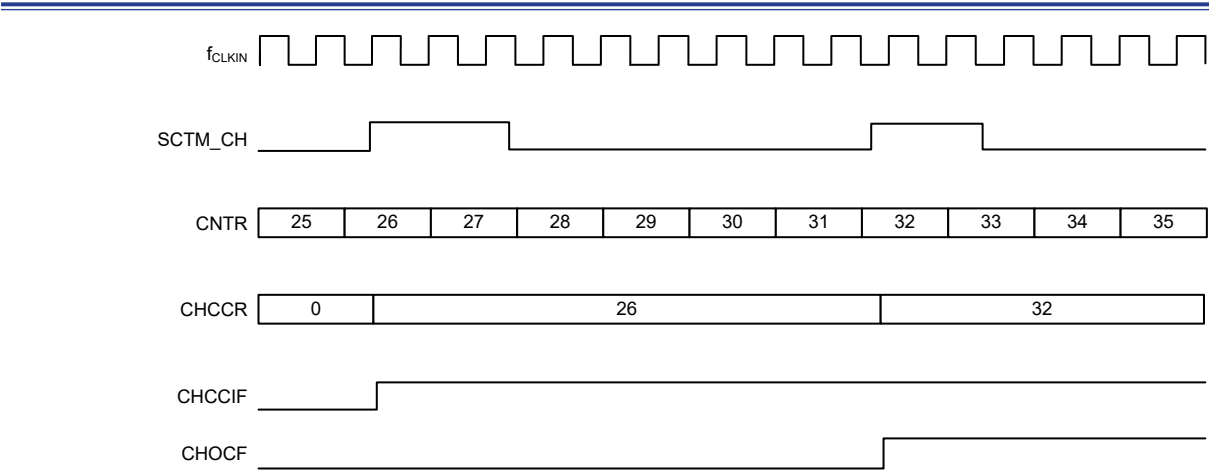


图 104. 输入捕捉模式

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。通道输入信号 TI 被一个数字滤波器采样，产生一个滤波输入信号 TIFP。通道极性和边沿检测模块可以产生一个 TISED 信号提供给输入捕捉功能。有效输入事件数量可通过通道捕捉输入源配置寄存器 CHICFR 的 CHPSC[19:18] 位域设置。

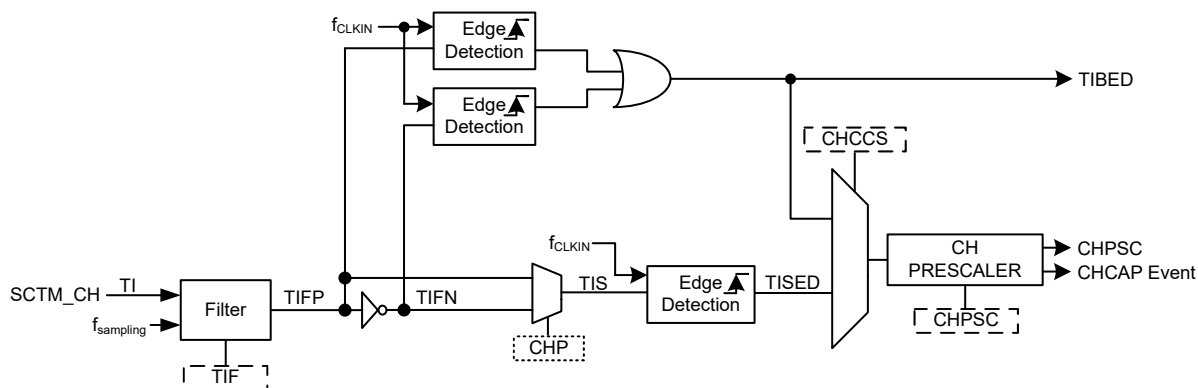


图 105. 通道输入级

数字滤波器

数字滤波器嵌入在通道输入级中。SCTM 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 个事件后会产生一个输出的跳变。根据该滤波器的用户选择，N 的值可以是 0、2、4、5、6 或 8。

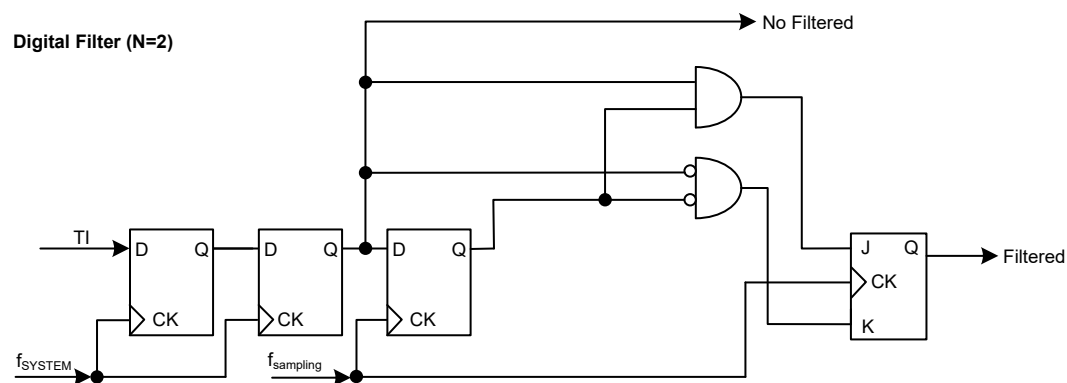


图 106. TI 数字滤波器方框图 (N = 2)

输出级

SCTM 输出具有比较匹配或 PWM 输出功能。通道输出 SCTM_CHO 由 CHOCFR、CHPOLR 和 CHCTR 寄存器分别对应的 CHOM、CHP 和 CHE 位控制。

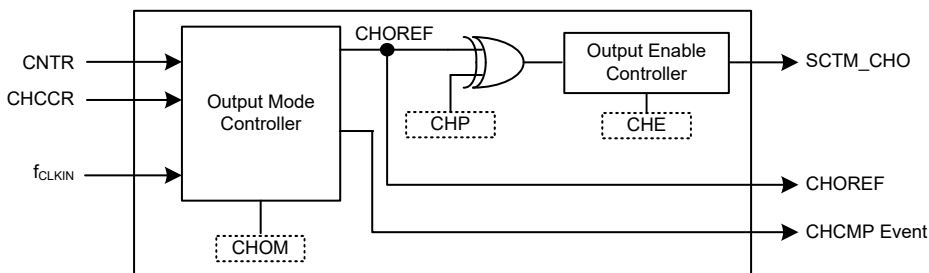


图 107. 输出级方框图

通道输出参考信号

当 SCTM 用在比较匹配输出模式时，CHOREF 信号（通道输出参考信号）通过设置 CHOM 位来定义。当计数器的值与 CHCCR 寄存器的内容匹配时 CHOREF 信号有几种输出功能类型，这些类型包括，CHOREF 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，CHOREF 信号的电平都是根据计数方向以及计数器值与 CHCCR 内容的关系而改变。还有两种模式，不论计数器和 CHCCR 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。输出类型设置如下表所示。

表 38. 比较匹配输出设置

CHOM 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

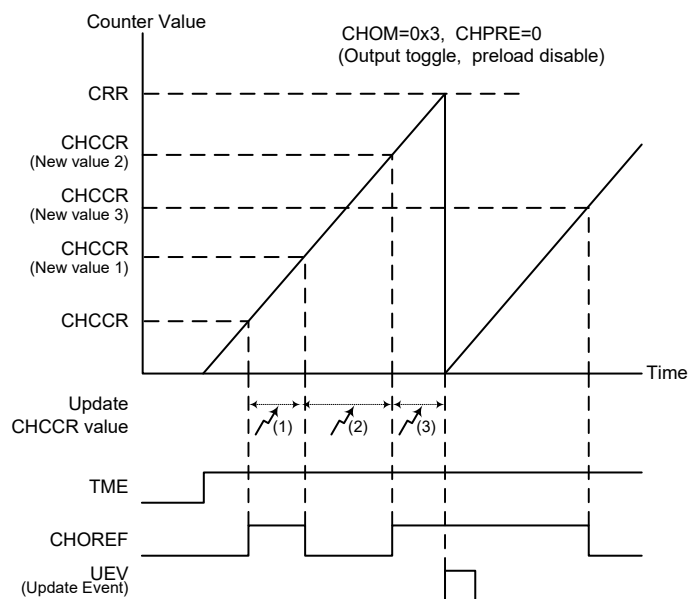


图 108. 翻转模式通道输出参考信号 – CHPRE = 0

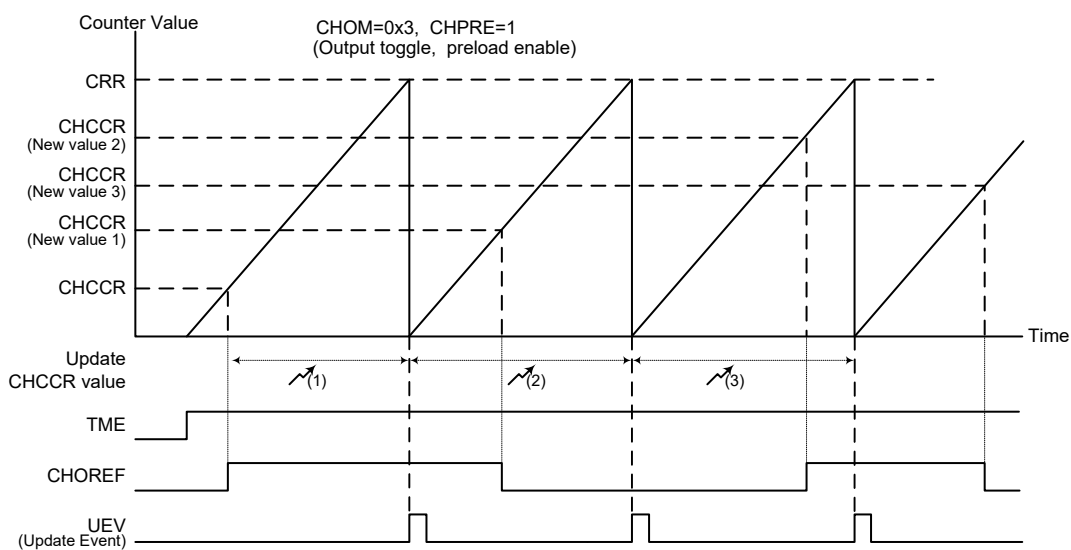


图 109. 翻转模式通道输出参考信号 – CHPRE = 1

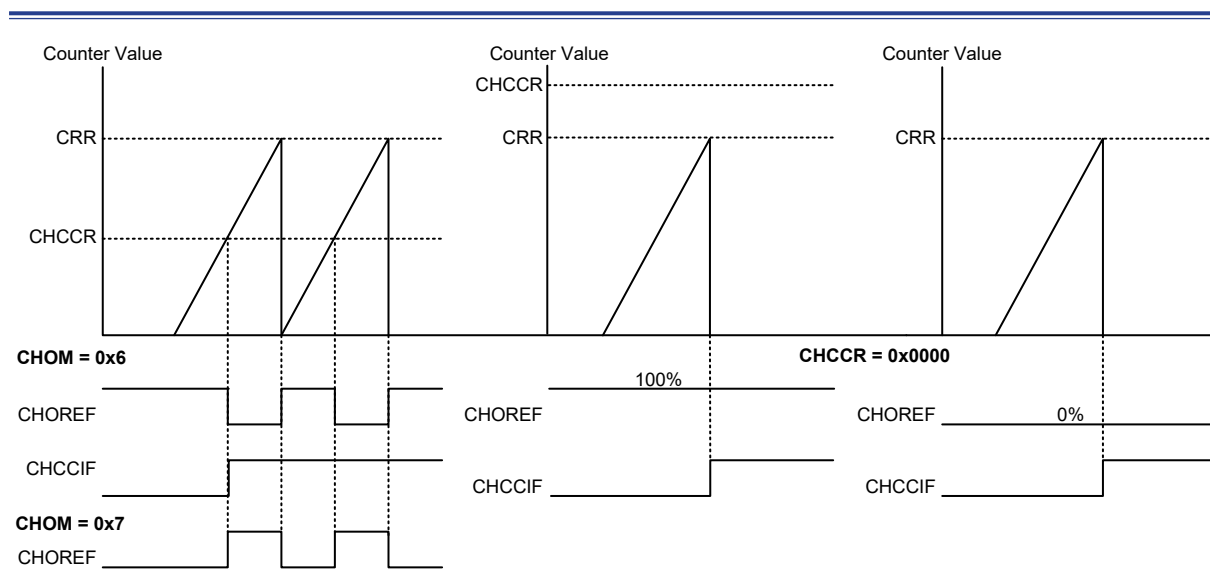


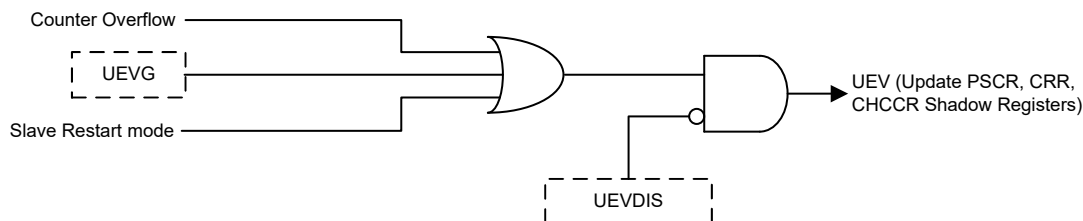
图 110. PWM 模式通道输出参考信号

更新管理

更新事件用来把实际寄存器中 CRR、PSCR 和 CHCCR 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、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。

更新事件能否发生由 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

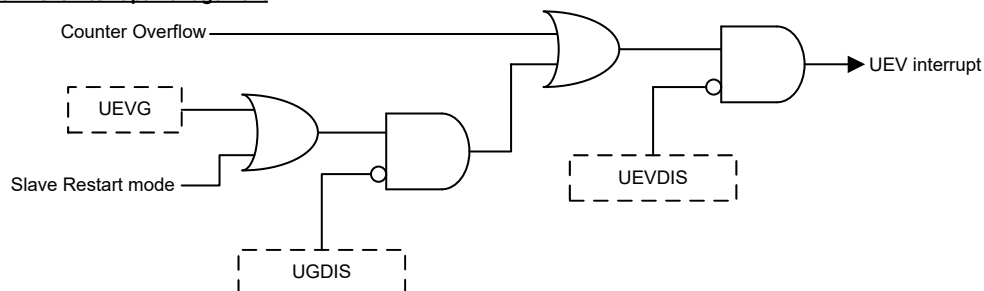


图 111. 更新事件设置方框图

寄存器列表

下表显示了 SCTM 寄存器及其复位值。

表 39. SCTM 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
CHICFR	0x020	通道输入配置寄存器	0x0000_0000
CHOCFR	0x040	通道输出配置寄存器	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
CHCCR	0x090	通道捕捉 / 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 SCTM 计数器配置。

偏移量: 0x000
复位值: 0x0000_0000

类型 / 复位	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						CKDIV	
							RW	0 RW 0
类型 / 复位	7	6	5	4	3	2	1	0
	保留位						UGDIS	UEVDIS
							RW	0 RW 0

位	域	描述
[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 – 设置 UEVG 位为 1 – 通过从机模式产生更新 1: 只在计数器上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使能更新事件请求 – 计数器上溢 – 设置 UEVG 位为 1 – 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 SCTM 从机模式选项。

偏移量： 0x004
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					SMSEL		
类型 / 复位					RW	0	RW	0
	7	6	5	4	3	2	1	0
	保留位							
类型 / 复位								

位	域	描述		
[10:8]	SMSEL	从机模式选项		
		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。
		100	重启模式	接收到 STI 信号上升沿信号时，计数器从 0 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，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，计数器从当前值开始计数。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。
		其它	保留	

定时器触发配置寄存器 – TRCFR

该寄存器定义了 SCTM 的触发源选项。

偏移量： 0x008

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				TRSEL			
类型 / 复位					RW	0	RW	0
						RW	0	RW
							RW	0
								RW
								0

位	域	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: 通道滤波输入 (TIS) 1000: 通道双边沿检测器 (TIBED) 其它: 保留 注: 当 SMSEL 位域为 0x0 除能从机模式时, 这些位才能被更新。

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	域	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: SCTM 关闭 1: SCTM 开启 当 TME 位被清零, 计数器停止计数且 SCTM 无功耗 (从机触发模式除外)。在从机触发模式中, TME 位可通过硬件自动置 1, 允许所有的 SCTM 寄存器正常工作。

通道输入配置寄存器 – CHICFR

该寄存器定义了通道输入模式配置。

偏移量： 0x020
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				CHPSC		CHCCS	
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				TIF			
类型 / 复位				RW	0	RW	0

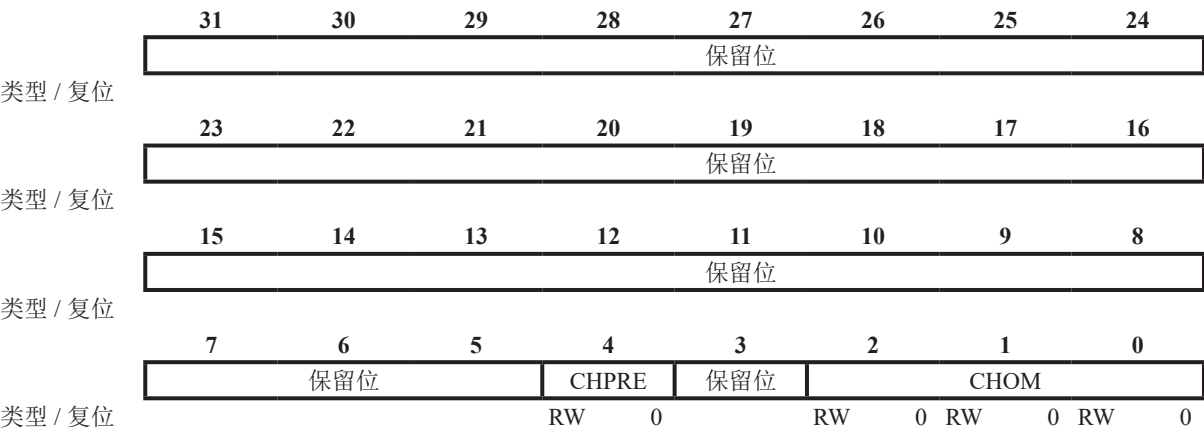
位	域	描述
[19:18]	CHPSC	通道捕捉输入源预分频器设置 这些位定义了通道捕捉输入的有效事件。注意，一旦通道控制寄存器 CHCTR 中的通道捕捉 / 比较使能位 CHE 位被清零，则预分频器将复位。 00: 无分频，每个有效事件为一个通道捕捉输入信号 01: 每 2 个事件为一个通道捕捉输入信号 10: 每 4 个事件为一个通道捕捉输入信号 11: 每 8 个事件为一个通道捕捉输入信号
[17:16]	CHCCS	通道捕捉 / 比较选择 00: 通道配置为输出 01: 通道配置为来自 TI 信号的一个输入 10: 保留 11: 通道配置为来自 TIBED 信号的一个输入 注：仅当 CHE 位清零时，才可以访问 CHCCS 位域。
[3:0]	TIF	通道输入源 TI 滤波器设置 这些位定义了用来采样 TI 信号的分频比。SCTM 中的数字滤波器是一个 N 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: $f_{\text{sampling}} = f_{\text{CLKIN}}$, $N = 2$ 0010: $f_{\text{sampling}} = f_{\text{CLKIN}}$, $N = 4$ 0011: $f_{\text{sampling}} = f_{\text{CLKIN}}$, $N = 8$ 0100: $f_{\text{sampling}} = f_{\text{DTS}} / 2$, $N = 6$ 0101: $f_{\text{sampling}} = f_{\text{DTS}} / 2$, $N = 8$ 0110: $f_{\text{sampling}} = f_{\text{DTS}} / 4$, $N = 6$ 0111: $f_{\text{sampling}} = f_{\text{DTS}} / 4$, $N = 8$ 1000: $f_{\text{sampling}} = f_{\text{DTS}} / 8$, $N = 6$ 1001: $f_{\text{sampling}} = f_{\text{DTS}} / 8$, $N = 8$ 1010: $f_{\text{sampling}} = f_{\text{DTS}} / 16$, $N = 5$ 1011: $f_{\text{sampling}} = f_{\text{DTS}} / 16$, $N = 6$ 1100: $f_{\text{sampling}} = f_{\text{DTS}} / 16$, $N = 8$ 1101: $f_{\text{sampling}} = f_{\text{DTS}} / 32$, $N = 5$ 1110: $f_{\text{sampling}} = f_{\text{DTS}} / 32$, $N = 6$ 1111: $f_{\text{sampling}} = f_{\text{DTS}} / 32$, $N = 8$

通道输出配置寄存器 – CHOCFR

该寄存器定义了通道输出模式配置。

偏移量：0x040

复位值：0x0000_0000



位	域	描述
[4]	CHPRE	通道捕捉 / 比较寄存器 (CHCCR) 预载使能位 0: CHCCR 预载功能除能 当 CHPRE 位清零, CHCCR 寄存器将立即被更新为新的值, 且立即可用。 1: CHCCR 预载功能使能 直到更新事件发生后, 新的 CHCCR 值才会被传送到影子寄存器中。
[2:0]	CHOM	通道输出模式设置 这些位定义了输出参考信号 CHOREF 的功能类型 000: 无变化 001: 比较匹配时输出 0 010: 比较匹配时输出 1 011: 比较匹配时输出翻转 100: 强制无效 – CHOREF 强制为 0 101: 强制有效 – CHOREF 强制为 1 110: PWM 模式 1 – 在向上计数期间, 当 CNTR < CHCCR, 通道处于有效电平, 否则将处于无效电平。 111: PWM 模式 2 – 在向上计数期间, 当 CNTR < CHCCR, 通道处于无效电平, 否则将处于有效电平。

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

	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							CHE
类型 / 复位								RW 0

位	域	描述
[0]	CHE	通道捕捉 / 比较使能位 – 通道被配置为输入 (CHCCS = 0x1 / 0x3) 0: 输入捕捉模式除能 1: 输入捕捉模式使能 – 通道被配置为输出 (CHCCS = 0x0) 0: 关闭 – 通道输出信号 CHO 无作用 1: 开启 – 通道输出信号 CHO 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							
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							CHP
类型 / 复位								RW 0

位	域	描述
[0]	CHP	通道捕捉 / 比较极性 – 通道被配置为输入时 (CHCCS = 0x1 / 0x3) 0: 捕捉事件发生在通道上升沿 1: 捕捉事件发生在通道下降沿 – 通道被配置为输出时 (CHCCS = 0x0) 0: 通道输出高电平有效 1: 通道输出低电平有效

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	
类型 / 复位						RW	0	RW	0
	7	6	5	4	3	2	1	0	
	保留位							CHCCIE	
类型 / 复位								RW	0

位	域	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
[0]	CHCCIE	通道捕捉 / 比较中断使能位 0: 除能通道中断 1: 使能通道中断

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								
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					TEVG	保留位	UEVG	
类型 / 复位						WO 0		WO 0	
	7	6	5	4	3	2	1	0	
	保留位							CHCCG	
类型 / 复位								WO 0	

位	域	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。相关寄存器也会被更新。详细描述请参考相关章节。
[0]	CHCCG	通道捕捉 / 比较发生 通道捕捉 / 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: 通道发生捕捉 / 比较事件 如果通道被配置为输入，计数器的值将被捕捉到 CHCCR 寄存器，接着 CHCCIF 位被置位。如果通道被配置为输出，则 CHCCIF 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

	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					W0C	0	W0C
	7	6	5	4	3	2	1	0
类型 / 复位	保留位			CHOCF	保留位			CHCCIF
				W0C				W0C

位	域	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 – UEVG 位被置位 – 来自从机触发输入的重启触发事件发生
[4]	CHOCF	通道过度捕捉标志位 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 CHCCIF 位已被置位且还没有被软件清零时，捕捉事件再次发生。
[0]	CHCCIF	通道捕捉 / 比较中断标志位 – 通道配置为输出时： 0：无匹配事件发生 1：计数器 CNTR 内容与 CHCCR 寄存器内容匹配 当计数器的值与 CHCCR 的值匹配时，此位被硬件置位。通过软件清零。 – 通道配置为输入时： 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 CHCCR 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	
	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	
	CNTV								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
	CNTV								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	域	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
	7	6	5	4	3	2	1	0	
	PSCV								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	域	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
	7	6	5	4	3	2	1	0		
	CRV									
类型 / 复位	RW	1	RW	1	RW	1	RW	1	RW	1

位	域	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。

通道捕捉 / 比较寄存器 – CHCCR

该寄存器定义了定时器通道捕捉 / 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	
	CHCCV								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
	CHCCV								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

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

16 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，用来测量时间间隔并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。重复模式在内部比较器产生比较匹配事件时，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。

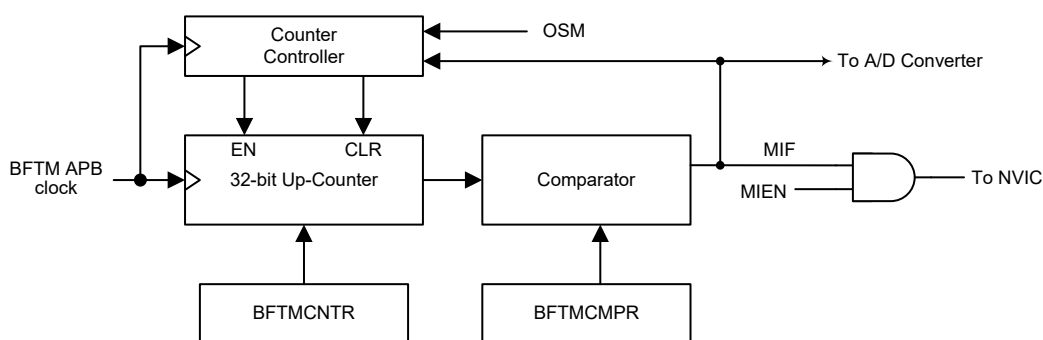


图 112. BFTM 方框图

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。

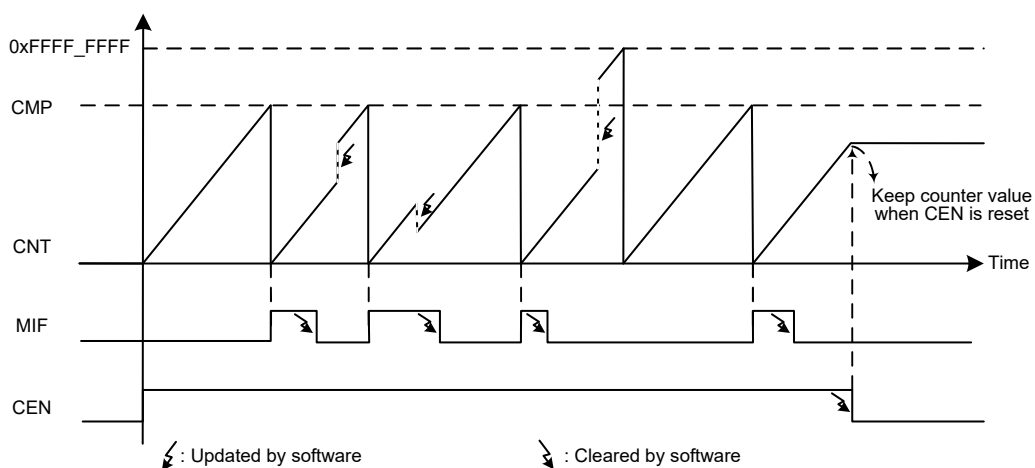


图 113. BFTM – 重复模式

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。

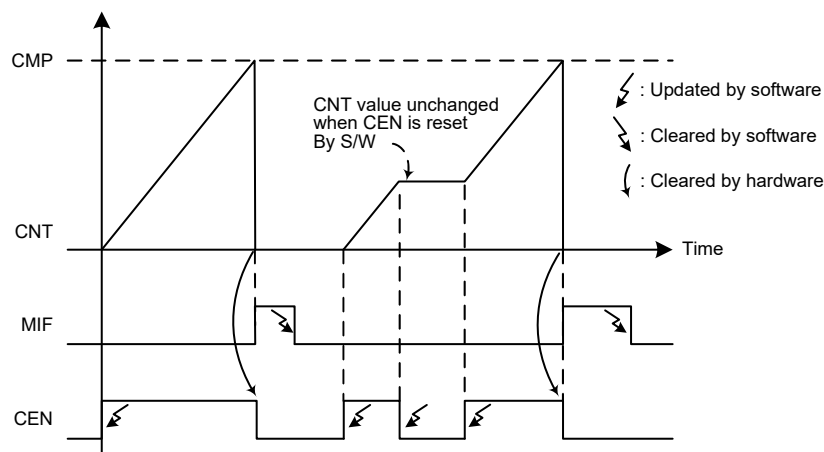


图 114. BFTM – 单次模式

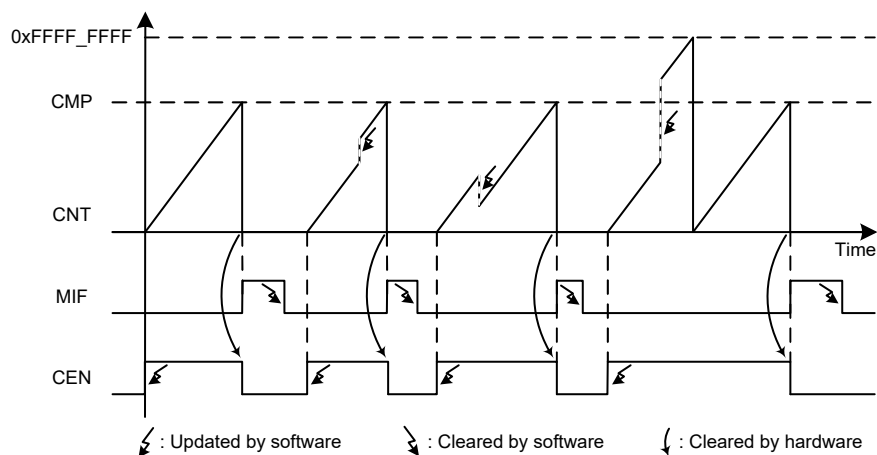


图 115. BFTM – 单次模式计数器更新

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 40. 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 控制寄存器	0x0000_0000
BFTMCNTR	0x008	BFTM 计数器值寄存器	0x0000_0000
BFTMCMPR	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
类型 / 复位	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 RW 0 RW 0 RW 0

位	域	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

	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							MIF
类型 / 复位								W0C 0

位	域	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 将被置位。MIF 位可通过软件写“0”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
	23	22	21	20	19	18	17	16	
	CNT								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	15	14	13	12	11	10	9	8	
	CNT								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
	CNT								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	域	描述
[31:0]	CNT	BFTM 计数器值 该位域用来存储 32-bit BFTM 计数器的值，在计数器运行过程中，都可读取或写入。

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	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	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	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	RW	1	RW	1

位	域	描述
[31:0]	CMP	BFTM 比较值 该寄存器用于指定与 BFTM 计数器值作比较的 32-bit BFTM 的比较值。

功能描述

RTC 相关寄存器复位

RTC 寄存器只能由 V_{DD} 域上电复位 (POR), 或通过设置 PWRCR 寄存器中的 PWRST 位使 V_{DD} 域软件复位的方式来复位。其它复位事件对清除 RTC 寄存器没有影响。

读取 RTC 寄存器

RTC 控制逻辑电路和相关的寄存器由 V_{DD} 电源电压供电。因此, RTC 电路在 V_{CORE} 电源关闭的暂停模式下仍可运行。V_{CORE} 电源关闭时, 只有位于 V_{CORE} 域的 APB 总线, 通过电平转换电路与位于 V_{DD} 域的 RTC 电路相连接并在 V_{CORE} 电源关闭时通过 ISO 信号隔离。

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 32768 Hz 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。分频器由 RPRE[3:0] 位域设置 CK_RTC 时钟分频比, 分频范围是 2⁰ ~ 2¹⁵。例如, 当 CK_RTC 时钟频率是 32768 Hz 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 41. 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 μA	500 ms 以上
快速启动	1	3.5 μA	300 ms 以下

@ V_{DD} = 3.3 V 和 LSE 时钟 = 32768 Hz; 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。这一操作用于定义一个时间间隔的预设值。当 RTCCNT 寄存器的值等于 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。对 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

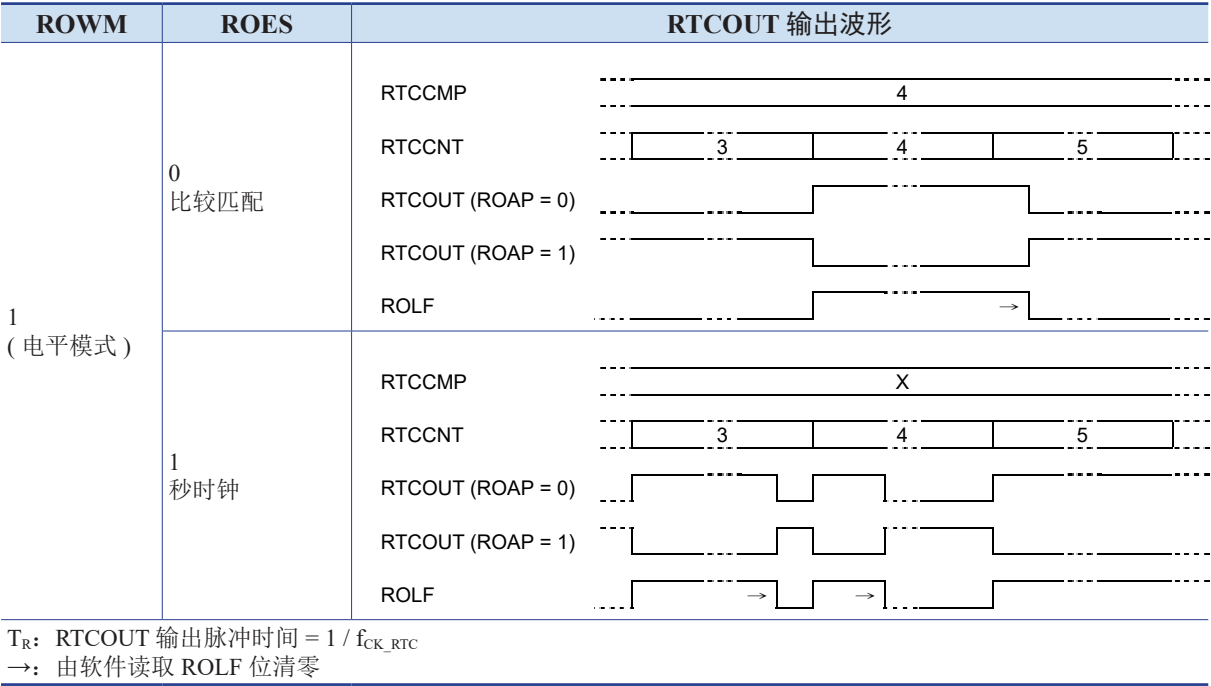
CK_SECOND 时钟下降沿会引起 RTCSR 寄存器中的 CSECFLAG 位置位, 如果 RTCIWEN 寄存器中的相关中断使能位 CSECIEN 置位, 还会产生中断。当相应的唤醒使能位 CSECWEN 置位时, 将产生唤醒事件来唤醒 HSI / HSE 振荡器、PLL 电路、LDO 以及 CPU 内核。当 RTC 计数器溢出或比较匹配事件发生时, 将产生中断或唤醒事件, 这取决于 RTCIWEN 寄存器中相应的中断使能控制位或唤醒使能控制位, OVIEN / OVWEN 或 CMIEN / CMWEN。欲知详细信息, 请参考相关的寄存器定义。

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 42. 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)	-----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)	-----T _R -----T _R -----T _R -----
		ROLF	-----



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_{DD} 电源域中。

表 43. 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_0F04
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。

偏移量: 0x000
复位值: 0x0000_0000 (仅由 V_{DD} 电源域复位)

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

位	域	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，RTCCNT 寄存器的值将被更新。该寄存器在以下情况被复位： – V _{DD} 域软件复位 – 在 PWRCCR 寄存器中设置 PWRST 位 – V _{DD} 域上电复位 – POR – 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。

偏移量： 0x004
复位值： 0x0000_0000 (仅由 V_{DD} 电源域复位来复位)

	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	域	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 RTCCR 寄存器中的 CMPCLR 位清零且匹配条件发生时，RTCSR 寄存器中的 CMFLAG 位被置位，而 RTCCNT 寄存器的值不受影响，计数器将继续计数直到溢出为止。当 RTCCR 寄存器中的 CMPCLR 位置 1 且匹配条件发生时，RTCSR 寄存器中的 CMFLAG 位被置位，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 电路控制位的范围。

偏移量： 0x008
复位值： 0x0000_0F04 (仅由 V_{DD} 电源域复位来复位)

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			ROLF	ROAP	ROWM	ROES	ROEN					
类型 / 复位												
			RC	0	RW	0	RW	0	RW	0	RW	0
15	14	13	12	11	10	9	8					
保留位				RPRE								
类型 / 复位												
				RW	1	RW	1	RW	1	RW	1	
7	6	5	4	3	2	1	0					
保留位		LSESM	CMPCLR	LSEEN	保留位	RTCSRC	RTCEN					
类型 / 复位												
		RW	0	RW	0	RW	0	RW	0			

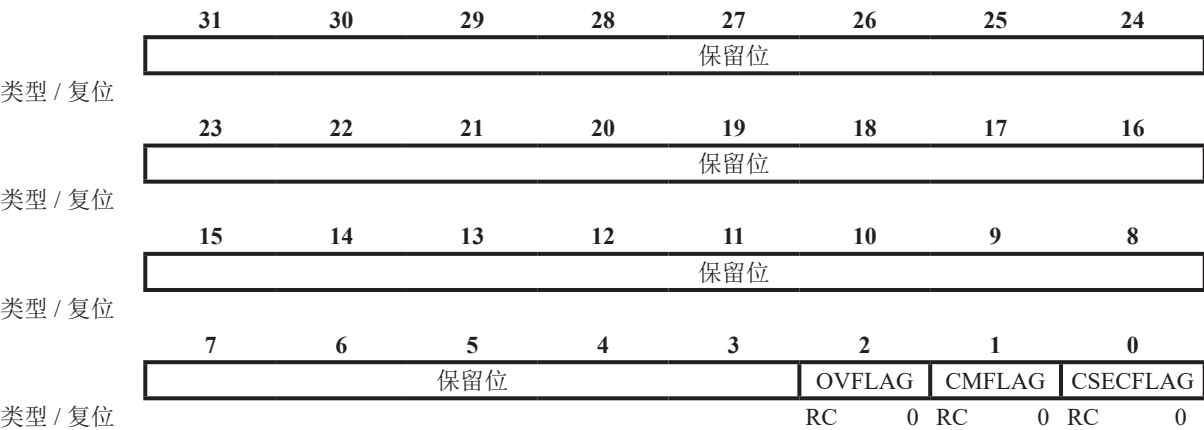
位	域	描述
[20]	ROLF	RTCOUNT 电平模式标志位 0: RTCOUNT 输出无效 1: RTCOUNT 输出保持在有效电平 当处于电平模式 (ROWM = 1) 且一个 RTCOUNT 输出事件发生时，此位被硬件置位。 软件读取此位会使其清零。软件读取此位后，RTCOUNT 信号将回到无效电平。
[19]	ROAP	RTCOUNT 输出有效极性 0: 高电平有效 1: 低电平有效
[18]	ROWM	RTCOUNT 输出波形模式 0: 脉冲模式 输出脉冲时间是一个 RTC 时钟 (CK_RTC) 周期。 1: 电平模式 在软件读取 ROLF 位使其清零之前，RTCOUNT 信号将保持在有效电平。
[17]	ROES	RTCOUNT 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 此位用于选择当 RTC 比较匹配事件还是 RTC 秒时钟 (CK_SECOND) 事件发生时，RTCOUNT 引脚输出 RTCOUNT 信号。
[16]	ROEN	RTCOUNT 输出引脚使能位 0: 除能 RTCOUNT 输出引脚 1: 使能 RTCOUNT 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，RTCOUNT 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，RTCOUNT 引脚将处于浮空状态。

位	域	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 24-bit RTC 计数器无影响 1: 比较匹配条件发生时, 24-bit 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。

偏移量： 0x00C
复位值： 0x0000_0000 (由 V_{DD} 电源域复位来复位且 RTCEN 位从 1 变为 0)



位	域	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 0xFFFF_FFFF 变为 0x0000_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 处理器中读取此位，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。

偏移量： 0x010

复位值： 0x0000_0000 (仅由 V_{DD} 电源域复位来复位)

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					OVWEN	CMWEN	CSECWEN
类型 / 复位					RW 0	RW 0	RW 0
7	6	5	4	3	2	1	0
保留位					OVIEN	CMIEN	CSECIEN
类型 / 复位					RW 0	RW 0	RW 0

位	域	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

18 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。这意味着看门狗定时器要防止软件死锁不断地被触发，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三种休眠模式时，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的非预期的突然改变。

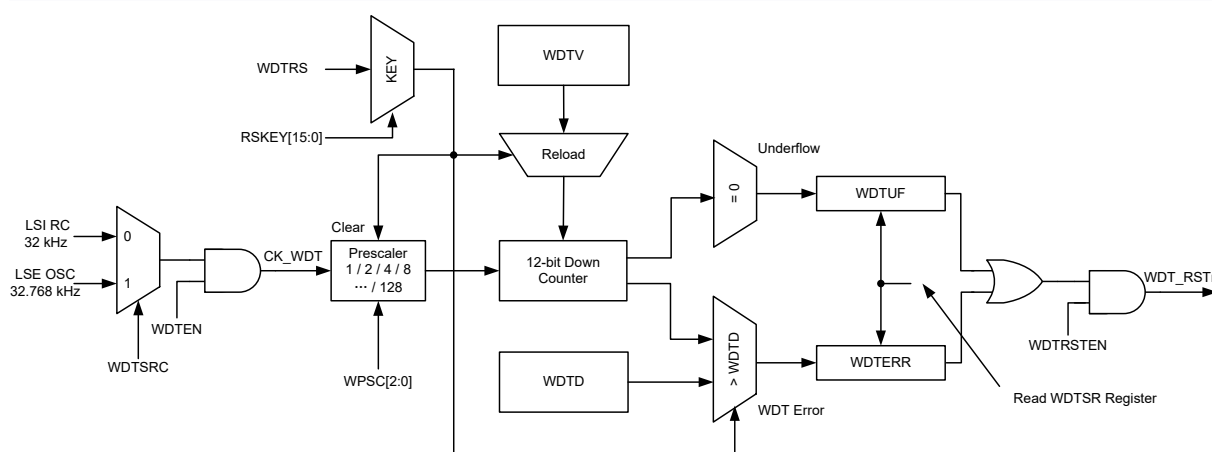


图 117. 看门狗定时器方框图

特性

- 时钟源来自于内部 32 kHz RC 振荡器 (LSI) 或外部 32768 Hz 振荡器 (LSE)
- 进入休眠模式或深度休眠模式 1 时，可以独立设置为继续运行或停止
- 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，可以定制看门狗定时器的重载时间
- 当处理器在调试模式时，看门狗定时器可能会停止
- 重载锁定键可以防止意外操作
- 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 LSE 或 LSI 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。为防止这些配置被写入不可预期的值，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 和设置锁定键位为 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检测到。为了避免这一情况的发生，当看门狗定时器计数器的值在增量值 (WDTD) 以内时必须执行重载操作。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“1”操作会清除 WDTERR 标志位和 WDTUF 标志位。

看门狗定时器使用两个时钟：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。而在系统进入深度休眠模式 2 时，看门狗定时器计数器将始终停止。看门狗定时器停止计数时，计数值会被保留，使得在系统从这三种休眠模式中被唤醒后可以继续计数。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) 和复位选项。
- 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重新载入增量值 (WDTD) 以内的重载值。

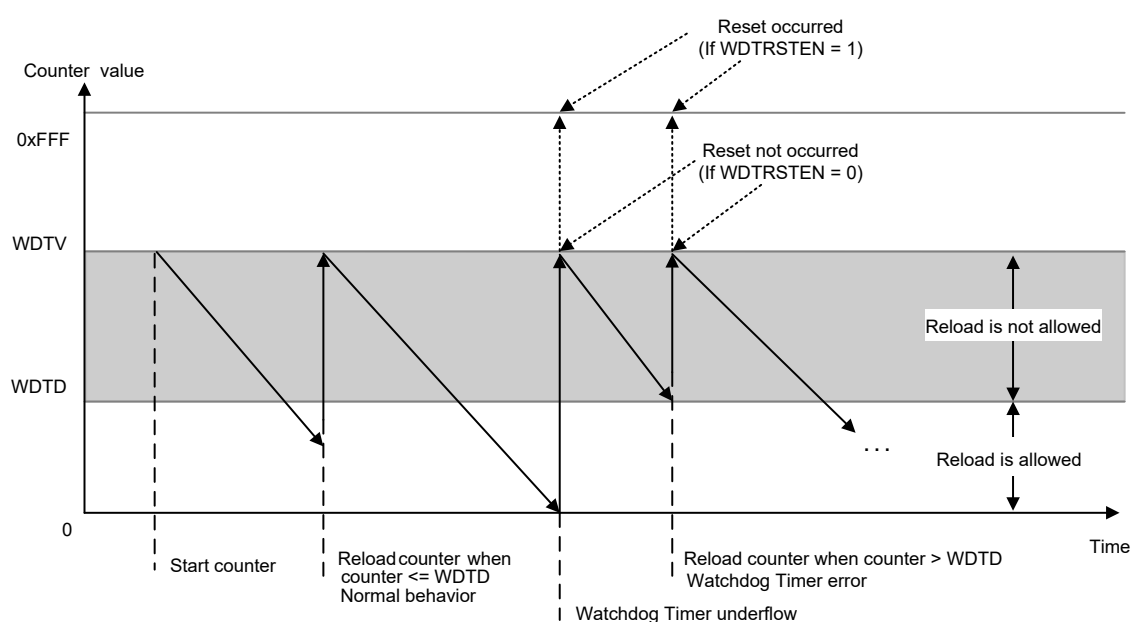


图 118. 看门狗定时器的运转

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 44. 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_7FFF
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	域	描述
[31:16]	RSKEY	看门狗定时器重载锁定键 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		
	保留位									
类型 / 复位										
	23	22	21	20	19	18	17	16		
	保留位							WDTEN		
类型 / 复位								RW	0	
	15	14	13	12	11	10	9	8		
	WDTSHLT		WDTRSTEN	保留位	WDTV					
类型 / 复位	RW	0	RW	0	RW	1	RW	1	RW	1
	7	6	5	4	3	2	1	0		
	WDTV									
类型 / 复位	RW	1	RW	1	RW	1	RW	1	RW	1

位	域	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 1x: 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。适当配置 WDTSHLT 位域可使看门狗定时器在休眠模式或深度休眠模式 1 下停止计数。当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中唤醒后继续计数。若在休眠模式或深度休眠模式 1 中发生看门狗定时器复位将唤醒单片机。
[13]	WDTRSTEN	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_7FFF

	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	WPSC				WDTD			
类型 / 复位		RW	1	RW	1	RW	1	RW	1
	7	6	5	4	3	2	1	0	
	WDTD								
类型 / 复位	RW	1	RW	1	RW	1	RW	1	RW

位	域	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中的值所允许的范围。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 WDTCSR 寄存器中写入 WDTRIS = 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 WDTCSR 寄存器中写入 WDTRIS = 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

	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						WDTERR	WDTUF
类型 / 复位							WC 0	WC 0

位	域	描述
[1]	WDTERR	看门狗定时器错误 0: 无看门狗定时器错误发生 1: 有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: 无看门狗定时器下溢发生 1: 有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键的配置。

偏移量：0x010

复位值：0x0000_0000

类型 / 复位	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							
	PROTECT							
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							
类型 / 复位	PROTECT							
	RW 0 RW 0 RW 0 RW 0 RW 0 RW 0 RW 0							

位	域	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 其它值：使能看门狗定时器寄存器写保护功能 读操作： 0x0000：除能看门狗定时器寄存器写保护功能 0x0001：使能看门狗定时器寄存器写保护功能 该寄存器用来除能 / 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							
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			WDTLOCK	保留位			WDTSRC
类型 / 复位				RW 0				RW 0

位	域	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。

19 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，而外部 I²C 接口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。

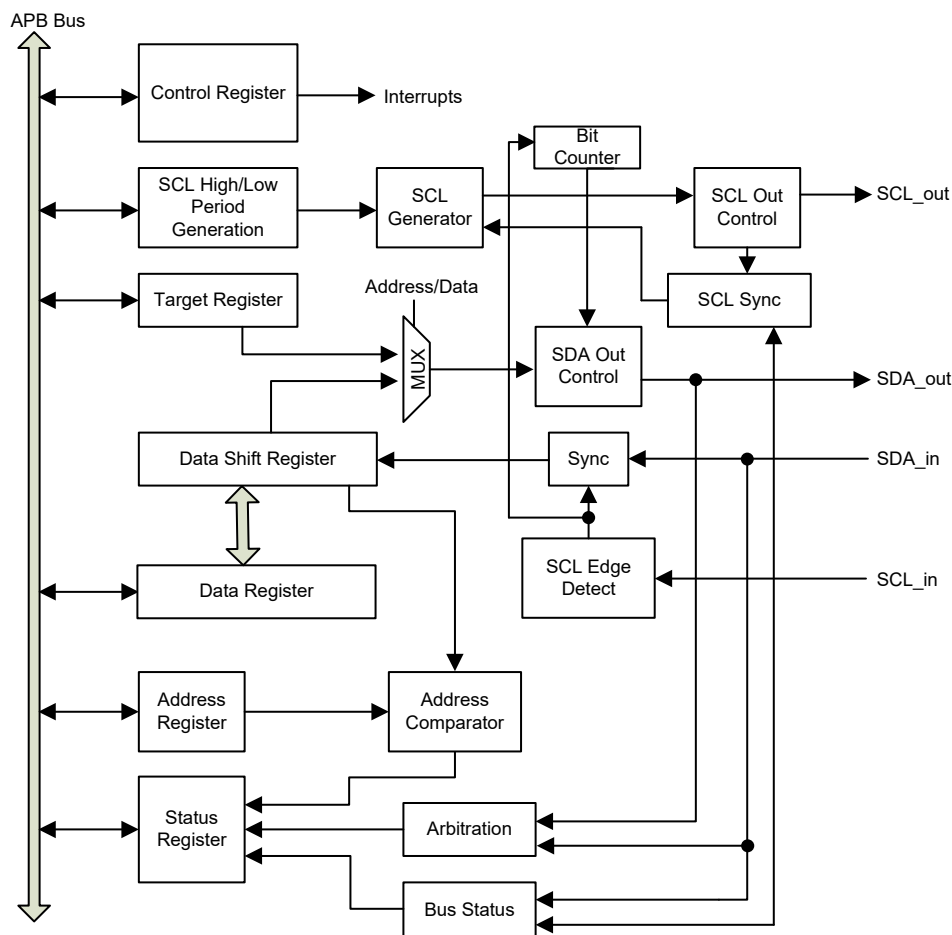


图 119. I²C 模块方框图

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
 - 同一个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
 - 允许不同比特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，可支持多个从机地址寻址
- 超时功能
- 支持 PDMA 接口

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的情况下与另一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。

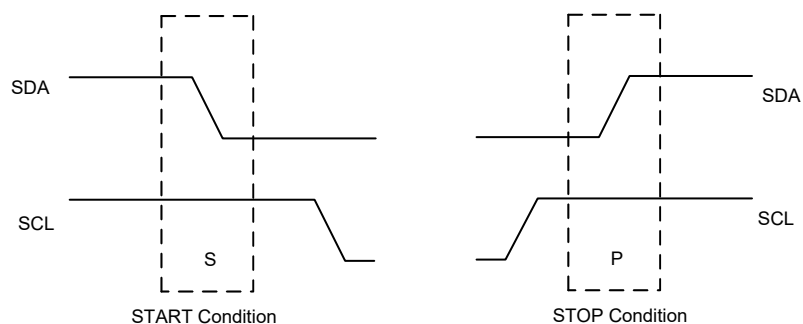


图 120. START 和 STOP 条件

数据有效性

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。

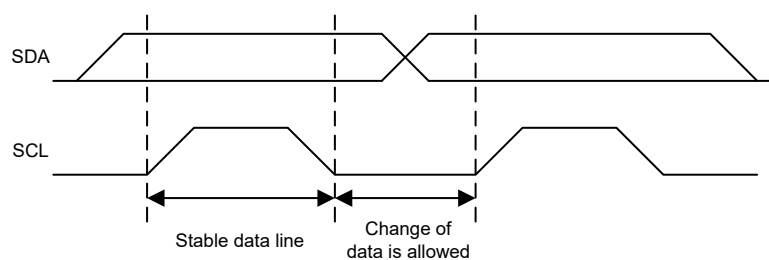


图 121. 数据有效性

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。I2CCR 寄存器中的寻址模式选择位 ADRM 应该被设置来选择 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 位域分配。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。

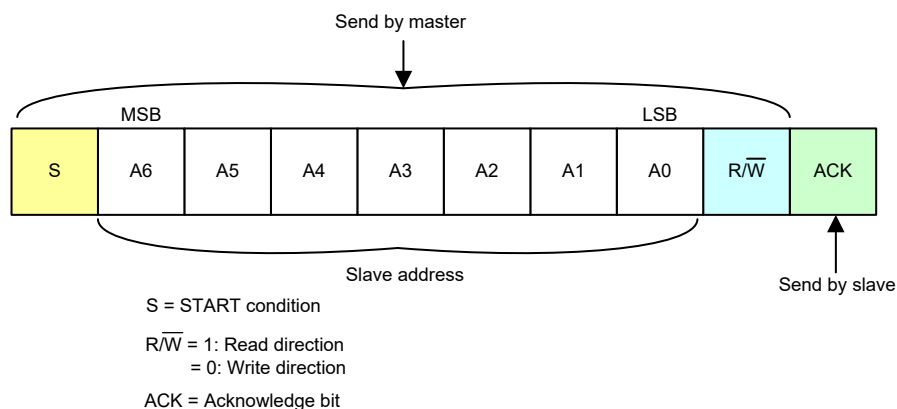


图 122. 7-bit 寻址模式

10-bit 地址格式

因 7-bit 地址范围有限, 为了防止地址冲突, 这里介绍一个 10-bit 地址格式的新方案。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。

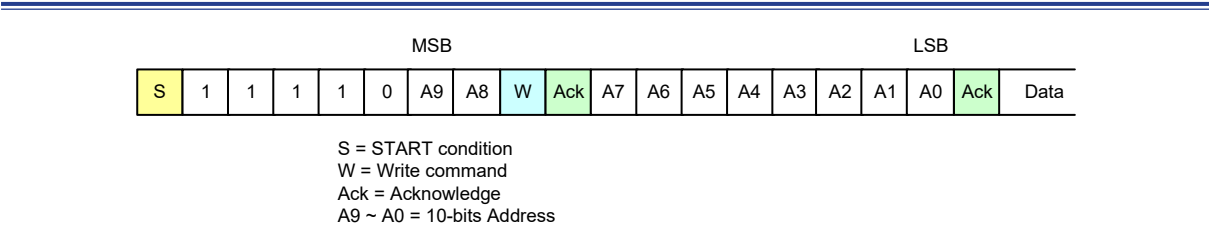


图 123. 10-bit 寻址写发送模式

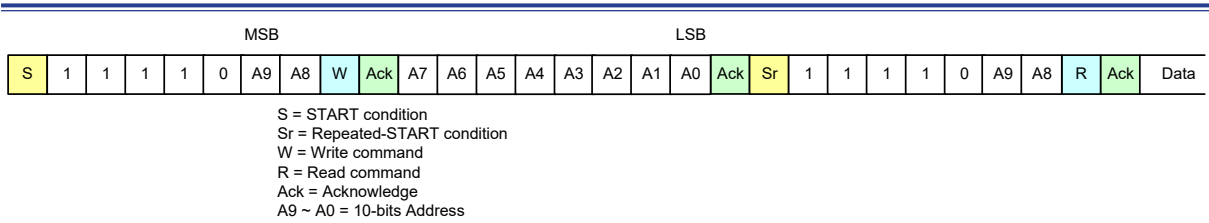


图 124. 10-bit 寻址读接收模式

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。

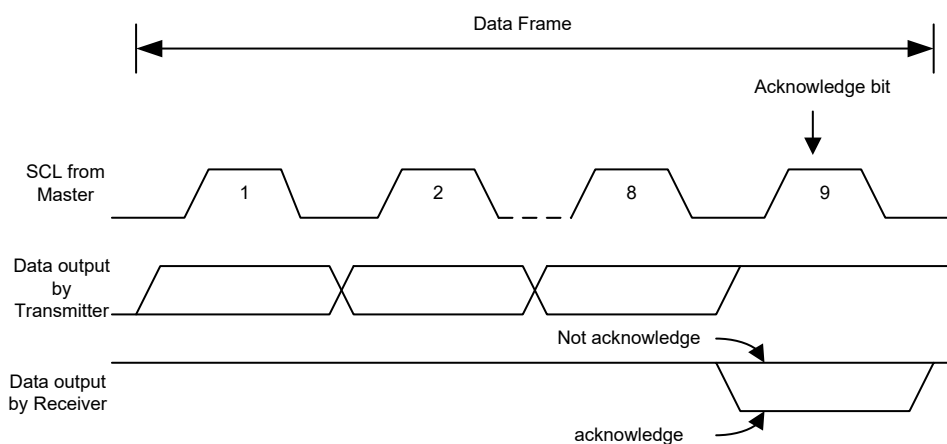


图 125. I²C 总线确认

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, 时钟应该同步以便数据输出可以比较。时钟同步可通过 I²C 接口到 SCL 线的线与连接来执行。

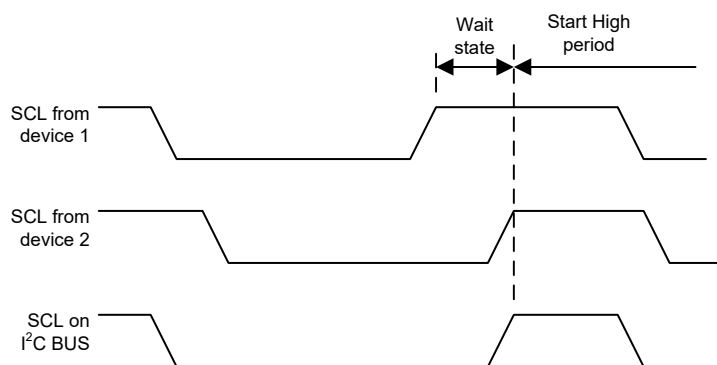


图 126. 仲裁期间时钟同步

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) 数据的主机将输掉仲裁。只要一个主机输掉仲裁, 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, 输掉仲裁的主机允许再次访问总线。

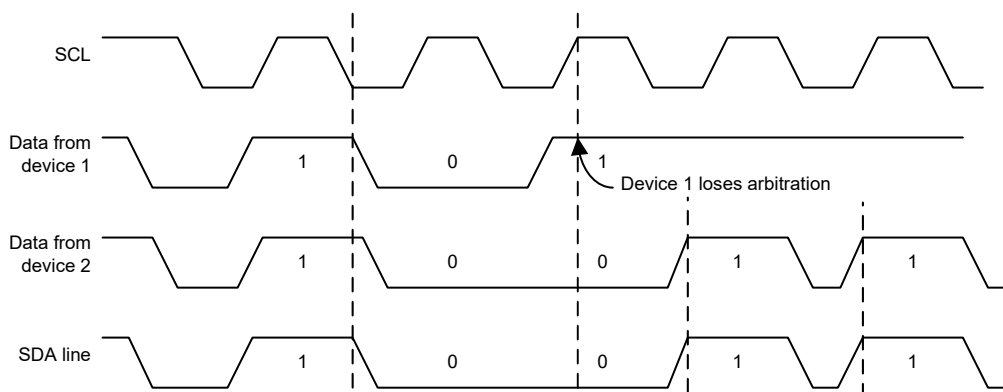


图 127. 两个主机仲裁程序

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，主机可通过设置 I2CTAR 寄存器中的 TAR 位域值为零且向 RWD 位写入 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 标志位将通过写 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中的哪一位。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 0x05 和 I2CADDR 寄存器的值为 0x55，这意味着如果总线上 I²C 主机发送的地址等于 0x50、0x51、0x54 或 0x55，该 I²C 从机地址都将被认为是匹配的，且 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
- 主机接收
- 从机发送
- 从机接收

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, 则必须使其清零。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, 则必须使其清零。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要先写入 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。新数据必须被传送到 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。

停止 / 继续发送

最后的数据字节发送完成后, I2CCR 寄存器中的 STOP 位需置位来结束发送或通过配置 I2CTAR 寄存器重新分配另外一个从机以开始新的传输。

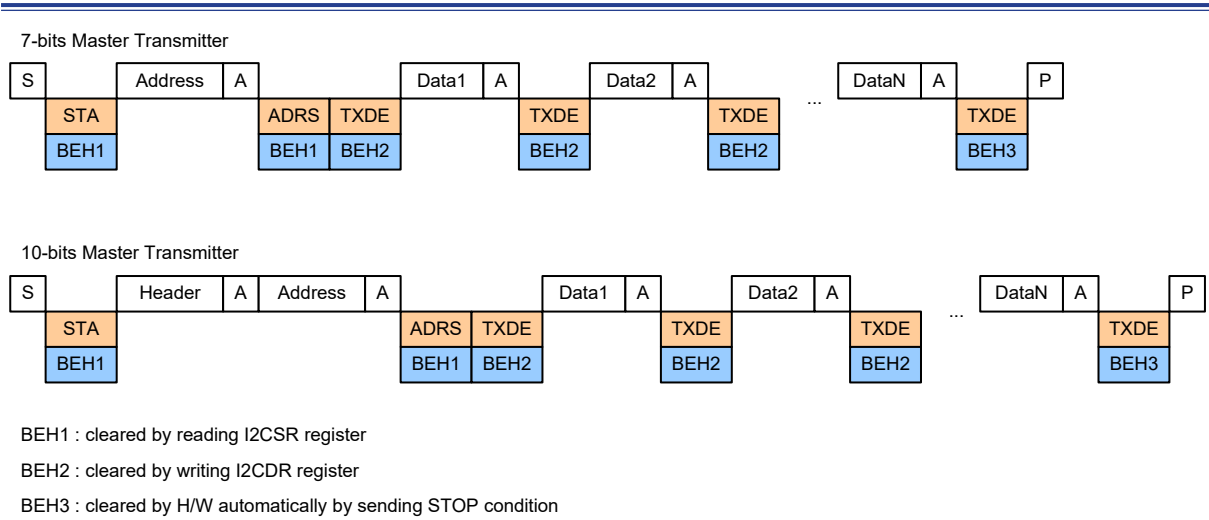


图 128. 主机发送时序图

主机接收模式

开始条件

目标从机地址和通信方向被写入 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, 则必须使其清零。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, 则必须使其清零。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 地址字节被发送且接收到来自于从机的确认信号时, ADRS 位被置位。第二次是当头字节再一次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, 则必须使其清零。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。

停止 / 继续发送

在最后一个数据字节传送完成前，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 位需置位来结束数据传送或重新配置 I2CTAR 寄存器以开始新的传输。

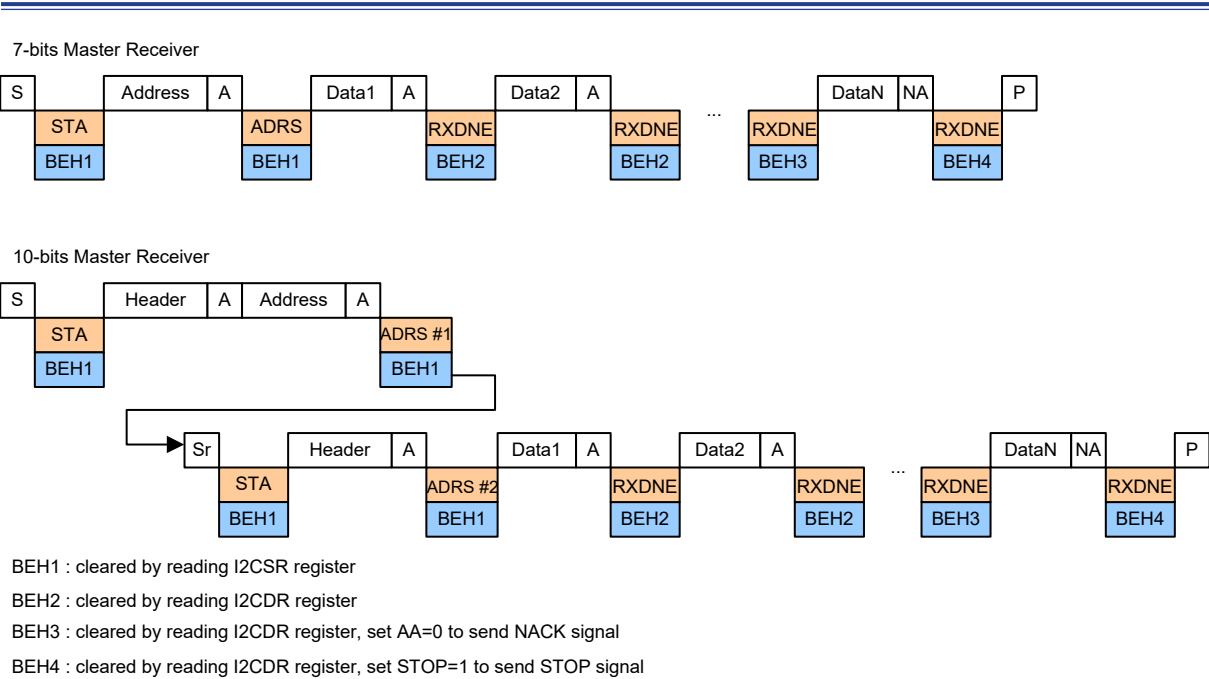


图 129. 主机接收时序图

从机发送模式

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 位第一次被置位。应注意当第二个头字节再一次匹配的时候，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。

接收未确认信号

当从机接收到一个未确认信号时，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。

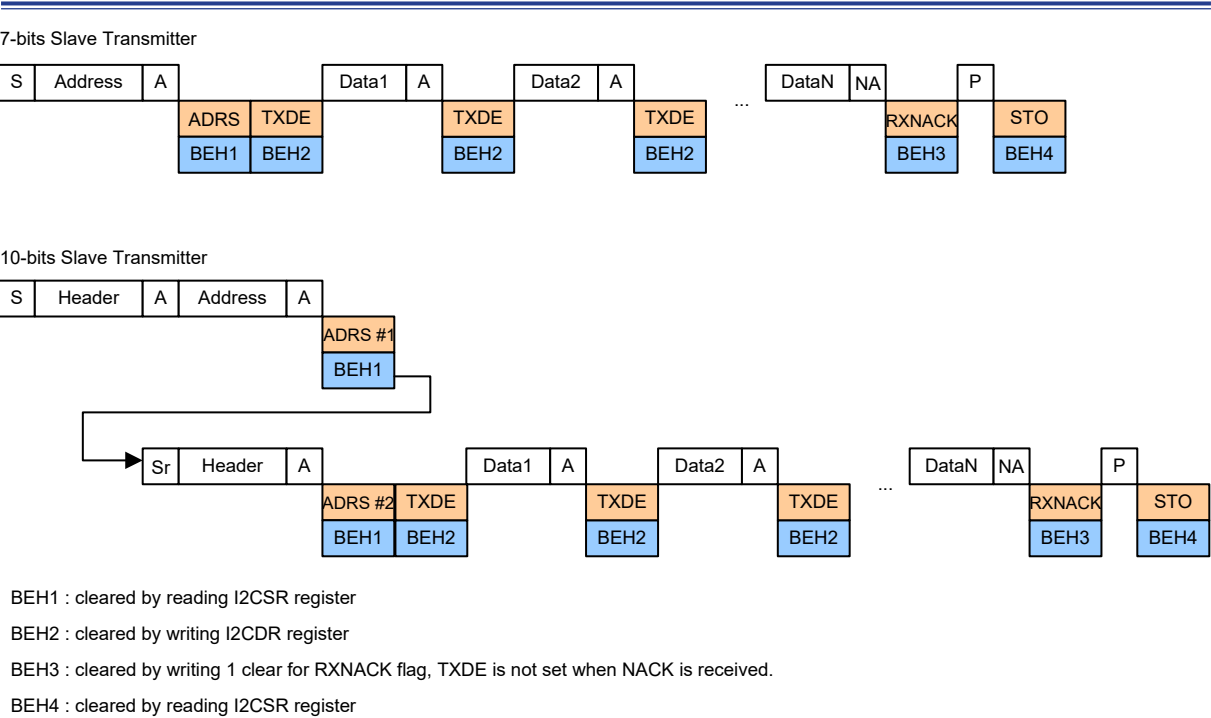


图 130. 从机发送时序图

从机接收模式

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 后必须使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，数据由主机发送。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。

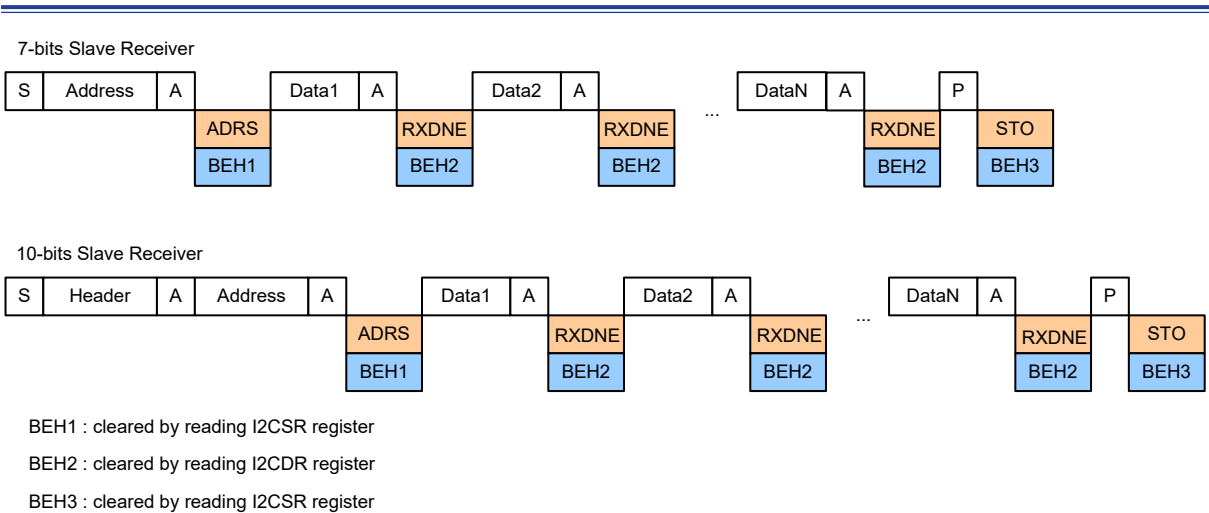


图 131. 从机接收时序图

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 45. 保持 SCL 线状态的条件

类型	条件	描述	条件消除后
标志位	TXDE	I ² C 模块用在发送模式，I2CDR 寄存器需要发送数据。 (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 设置 TAR 设置 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 128 和图 129) 从机： I ² C 模块作为从机被寻址 (注：请参考图 130 和图 131)	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。	设置 TAR 设置 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 129。此情况下，RXNACK 标志将不会被置位。)	设置 TAR 设置 STOP

I²C 超时功能

超时功能可减少由于接收错误时钟源造成 I²C 被锁住的问题。如果在超时周期里没有接收到 I²C 总线时钟源，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位域定义的。I2CTOUT 寄存器的 TOUT 位域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。

PDMA 接口

PDMA 接口集成于 I²C 模块。在发送或接收模式下可分别通过设置 TXDMAE 或 RxDMAE 位为 1 来使能 PDMA 功能。在发送器模式下, 若数据寄存器为空, TXDMAE 位置 1, PDMA 功能将被激活用于把数据从某一个存储器位置移到 I²C 数据寄存器。类似地, 在接收器模式下, 若数据寄存器不为空且 RxDMAE 位置 1, PDMA 功能将被激活用于把数据从 I²C 数据寄存器移到指定的存储器位置。

当 I²C 模块工作于主机接收器模式且 PDMA 功能使能时, DMA 模式 NACK 控制位 DMANACK 用于确认 NACK 信号是否发送。如果 DMANACK 位被置 1, 并且数据已全部通过 PDMA 接口被接收和移动, NACK 信号将自动发出以正确终止数据传输。

PDMA 配置模式的详细描述, 请参考 PDMA 章节。

寄存器列表

下表所示为 I²C 寄存器及其复位值。

表 46. 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
复位值: 0x0000_2000

类型 / 复位	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
	SEQFILTER	COMBFILTEREN	ENTOUT	保留位	DMANACK	RXDMAE	TXDMAE	
类型 / 复位	RW	0	RW	0	RW	1	RW	0
类型 / 复位	7	6	5	4	3	2	1	0
	ADRM	保留位			I2CEN	GCEN	STOP	AA
类型 / 复位	RW	0			RW	0	RW	0

位	域	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
[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。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 位域。
[10]	DMANACK	DMA 模式 NACK 控制位 0: 无操作 1: 在 DMA 模式下, I ² C 主机接收器模块在接收到从机发送器的最后一个字节之后会自动发送一个 NACK 信号
[9]	RXDMAE	DMA 模式 RX 请求使能控制位 0: RX DMA 请求除能 1: RX DMA 请求使能 在接收器模式下, 如果数据寄存器不为空, RXDMAE 位被置 1, 相关 PDMA 通道将被激活用于把数据从数据寄存器移到特定的位置, 该位置由相关 PDMA 寄存器定义。

位	域	描述
[8]	TXDMAE	DMA 模式 TX 请求使能控制位 0: TX DMA 请求除能 1: TX DMA 请求使能 在发送器模式下，如果数据寄存器不为空，TXDMAE 位被置 1，相关 PDMA 通道将被激活用于把数据从特定的位置移到数据寄存器，该位置由相关 PDMA 寄存器定义。
[7]	ADRM	寻址模式 0: 7-bit 寻址模式 1: 10-bit 寻址模式 当 I²C 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。
[3]	I2CEN	I²C 接口使能位 0: I²C 接口除能 1: I²C 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 I²C 接口将作为一个从机且 I2CSR 寄存器中的 GCS 位将被置 1。
[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: 在一个字节接收后发送一个未确认信号 (NACK) 1: 在一个字节接收后发送一个确认信号 (ACK)

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			
类型 / 复位	保留位										
	23	22	21	20	19	18	17	16			
类型 / 复位	保留位					RXBFIE	TXDEIE	RXDNEIE			
						RW	0	RW	0	RW	0
	15	14	13	12	11	10	9	8			
类型 / 复位	保留位					TOUTIE	BUSERRIE	RXNACKIE	ARBLOSIE		
						RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0			
类型 / 复位	保留位					GCSIE	ADRSIE	STOIE	STAIE		
						RW	0	RW	0	RW	0

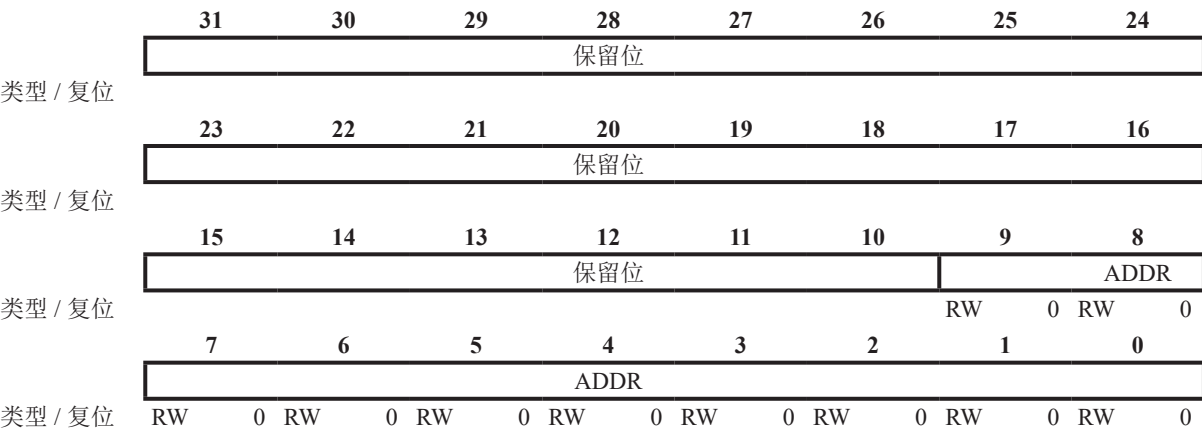
位	域	描述
[18]	RXBFIE	RX 缓冲器已满中断使能位 0: 中断除能 1: 中断使能
[17]	TXDEIE	发送器模式下数据寄存器空中断使能位 0: 中断除能 1: 中断使能
[16]	RXDNEIE	接收器模式下数据寄存器非空中断使能位 0: 中断除能 1: 中断使能
[11]	TOUTIE	超时中断使能位 0: 中断除能 1: 中断使能
[10]	BUSERRIE	总线错误中断使能位 0: 中断除能 1: 中断使能
[9]	RXNACKIE	接收未确认信号中断使能位 0: 中断除能 1: 中断使能
[8]	ARBLOSIE	I ² C 多主机模式下仲裁丢失中断使能位 0: 中断除能 1: 中断使能
[3]	GCSIE	广播呼叫从机中断使能位 0: 中断除能 1: 中断使能
[2]	ADRSIE	从机地址匹配中断使能位 0: 中断除能 1: 中断使能

位	域	描述
[1]	STOIE	STOP 条件检测中断使能位 0: 中断除能 1: 中断使能 此位仅用于 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 此位仅用于 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	域	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							
类型 / 复位							
23	22	21	20	19	18	17	16
保留位		TXNRX	MASTER	BUSBUSY	RXBF	TXDE	RXDNE
类型 / 复位		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
类型 / 复位				WC	0 WC	0 WC	0 WC
7	6	5	4	3	2	1	0
保留位				GCS	ADRS	STO	STA
类型 / 复位				RC	0 RC	0 RC	0 RC

位	域	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 总线错误来软件除能 I ² C 总线时，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。

位	域	描述
[17]	TXDE	<p>发送器模式下数据寄存器为空</p> <p>0: 数据寄存器 I2CDR 非空</p> <p>1: 数据寄存器 I2CDR 为空</p> <p>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置位, 告知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; 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。</p>
[16]	RXDNE	<p>接收器模式下数据寄存器非空</p> <p>0: 数据寄存器 I2CDR 为空</p> <p>1: 数据寄存器 I2CDR 非空</p> <p>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。</p>
[11]	TOUTF	<p>超时计数器下溢标志位</p> <p>0: 无超时计数器下溢发生</p> <p>1: 超时计数器下溢发生</p> <p>此位写入 1 将清零 TOUTF 标志位。</p>
[10]	BUSERR	<p>总线错误标志位</p> <p>0: 无总线错误发生</p> <p>1: 总线错误发生</p> <p>在传输过程中, 当 I²C 接口检测到一个错误的 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。</p> <p>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。</p> <p>在从机模式下: 一旦从机检测到一个错误的 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。</p>
[9]	RXNACK	<p>接收未确认信号标志位</p> <p>0: 从接收器返回确认信号</p> <p>1: 从接收器返回未确认信号</p> <p>RXNACK 位表明在主机或从机发送器模式下接收到未确认信号。向此位写入 1 来清除 RXNACK 标志位。</p>
[8]	ARBLOS	<p>仲裁丢失标志位</p> <p>0: 无仲裁丢失被检测到</p> <p>1: 位仲裁丢失被检测到</p> <p>在地址或数据帧发送过程中, 当 I²C 接口在主机模式下于当下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。</p>
[3]	GCS	<p>广播呼叫从机标志位</p> <p>0: 无广播呼叫从机发生</p> <p>1: I²C 接口通过广播呼叫命令被寻址</p> <p>在 7-bit 寻址模式或 10-bit 寻址模式下, I²C 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。</p>

位	域	描述
[2]	ADRS	<p>地址发送 (主机模式) / 地址接收 (从机模式) 标志位</p> <p>主机模式下地址发送</p> <p>0: 地址帧没被发送</p> <p>1: 地址帧已被发送</p> <p>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位后被置位。对于 10-bit 寻址模式, 此位在分别接收到第一个头字节和第二个地址字节的确认位后被置位。应注意若第二个头字节也确认, 此位也会被置位。</p> <p>从机模式下地址匹配</p> <p>0: I²C 接口没有被寻址</p> <p>1: I²C 接口作为从机被寻址</p> <p>当 I²C 接口接收到与 I2CADDR 寄存器里定义的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。</p>
[1]	STO	<p>STOP 条件检测标志位</p> <p>0: 无 STOP 条件被检测</p> <p>1: 从机模式下 STOP 条件被检测</p> <p>此位只在从机模式下可用, 且在读取 I2CSR 寄存器后自动清零。</p>
[0]	STA	<p>START 条件发送标志位</p> <p>0: 无 START 条件被发送</p> <p>1: 在主机模式 START 条件被发送</p> <p>此位只在主机模式下可用, 且在读取 I2CSR 寄存器后自动清零。</p>

I²C SCL 高电平周期发生寄存器 – I2CSHPGR

该寄存器定义了 I²C SCL 时钟高电平周期间隔。

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
	7	6	5	4	3	2	1	0	
类型 / 复位	SHPG								
	RW	0	RW	0	RW	0	RW	0	RW

位	域	描述
[15:0]	SHPG	SCL 时钟高电平周期产生 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 时钟低电平周期间隔。

偏移量: 0x014
复位值: 0x0000_0000

类型 / 复位	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
	7	6	5	4	3	2	1	0
类型 / 复位	SLPG							
	RW	0	RW	0	RW	0	RW	0

位	域	描述
[15:0]	SLPG	SCL 时钟低电平周期产生 低电平周期持续时间设置 $SCL_{LOW} = T_{PCLK} \times (SLPG + d)$ 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。 若 SEQFILTER = 00, d = 6 若 SEQFILTER = 01, d = 8 若 SEQFILTER = 10 或 11, d = 9

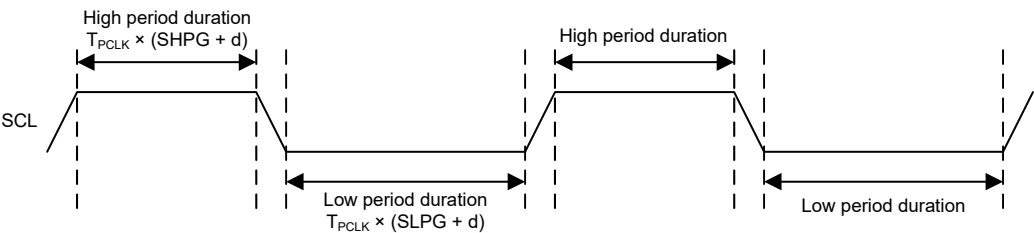


图 132. SCL 时序图

表 47. I²C 时钟设置范例

I²C 时钟	$T_{SCL} = T_{PCLK} \times [(SHPG + d) + (SLPG + d)]$ (d = 6) PCLK 时钟下 SHPG + SLPG 的值			
	8 MHz	24 MHz	48 MHz	72 MHz
100 kHz (标准模式)	68	228	468	708
400 kHz (快速模式)	8	48	108	168
1 MHz (高速模式)	x	12	36	60

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

	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
	DATA							
类型 / 复位	RW	0	RW	0	RW	0	RW	0

位	域	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分配给这些位。如果软件分配新的数据给 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。一旦发送了确认位，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							
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					RWD	TAR	
类型 / 复位						RW	0	RW
	7	6	5	4	3	2	1	0
	TAR							
类型 / 复位	RW	0	RW	0	RW	0	RW	0

位	域	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 接口将在第一个头帧中的字节初始化为 11110XX0b，接着硬件自动发送值为 11110XX1b 的字节在第二个头帧。
[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 – I2CADDRMR

该寄存器定义了 I²C 地址哪个位被屏蔽且无需与接收到的地址帧对应位作比较。

偏移量: 0x020
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						ADDRMR	
类型 / 复位							RW	0 RW 0
	7	6	5	4	3	2	1	0
	ADDRMR							
类型 / 复位	RW	0 RW	0 RW	0 RW	0 RW	0 RW	0 RW	0

位	域	描述
[9:0]	ADDRMR	地址屏蔽控制位 ADDRMR[i] 用于定义 I2CADDR 寄存器 ADDR 位域的第 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 位被屏蔽且没有与 I ² C 总线上的地址帧作比较

I²C 地址捕获寄存器 – I2CADDRSR

该寄存器用于描述 I²C 总线上的地址帧值。

偏移量： 0x024
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						ADDRSR	
类型 / 复位							RO	0
	7	6	5	4	3	2	1	0
	ADDRSR							
类型 / 复位	RO	0	RO	0	RO	0	RO	0

位	域	描述
[9:0]	ADDRSR	地址捕获 一旦 I2CEN 位使能，I²C 总线上的呼叫地址值将被自动载入到该 ADDSR 位域。

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

类型 / 复位	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					PSC		
类型 / 复位	15	14	13	12	11	10	9	8
	TOUT							
类型 / 复位	7	6	5	4	3	2	1	0
	TOUT							

位	域	描述
[18:16]	PSC	I ² C 超时计数器预分频选项 该 PSC 位域用于定义 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 位域用于定义计数器预载值。 下列情况发生时，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。

20 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 通信协议的数据发送与接收功能。SPI 接口使用 4 个引脚，其中有串行数据输入和输出线 SPI_MISO 和 SPI_MOSI，时钟线 SPI_SCK 和从机选择线 SPI_SEL。SPI 通信中作为主机的设备，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。接收数据时，数据帧长度由 SPICR1 寄存器的 DFL 位域定义为 1 位到 16 位，数据在特定的时钟边沿时被锁存并存储到数据寄存器或 RX FIFO。发送数据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。

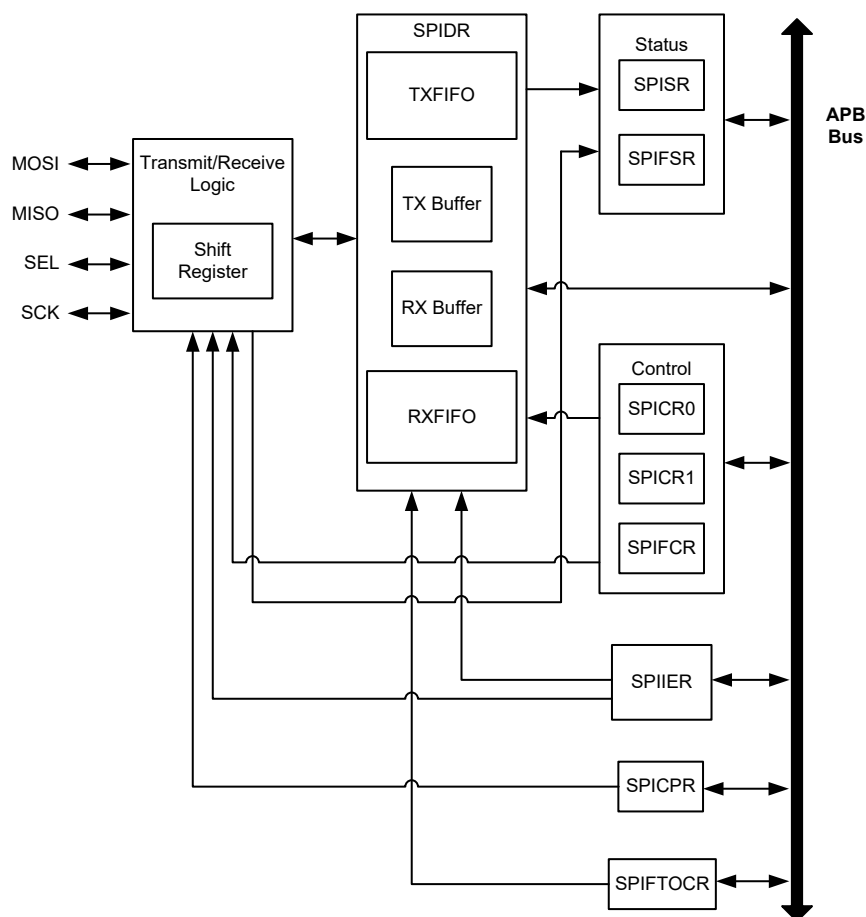


图 133. SPI 方框图

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
- 支持 PDMA 接口

功能描述

主机模式

每个数据帧的长度范围是 1 ~ 16 位。被发送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 SPI_SCK 引脚产生串行时钟。在串行时钟边沿，移位寄存器中的数据被发送到 SPI_MOSI 引脚。在数据传输的整个过程中，SPI_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 SPI_SEL 引脚为高电平有效。当 SPICR1 寄存器中的 SELM 位被置位，SPI_SEL 引脚将由硬件自动驱动，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，SPI_SCK 引脚作为输入脚且串行时钟来自于外部主机。SPI_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 信号为低电平有效。当 SELAP 位被置为 1，在完整数据流接收期间，SEL 信号为高电平有效。

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。

- 时钟极性位 – CPOL
当时钟极性位为零，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。
- 时钟相位位 – CPHA
当时钟相位位为零，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。下表显示了如何通过 SPICR1 寄存器中的 FORMAT 位域设置这些格式。

表 48. SPI 接口格式设置

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

CPOL = 0, CPHA = 0

在此格式下，接收的数据在 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。下图显示了此格式下单个字节数据传输时序图。

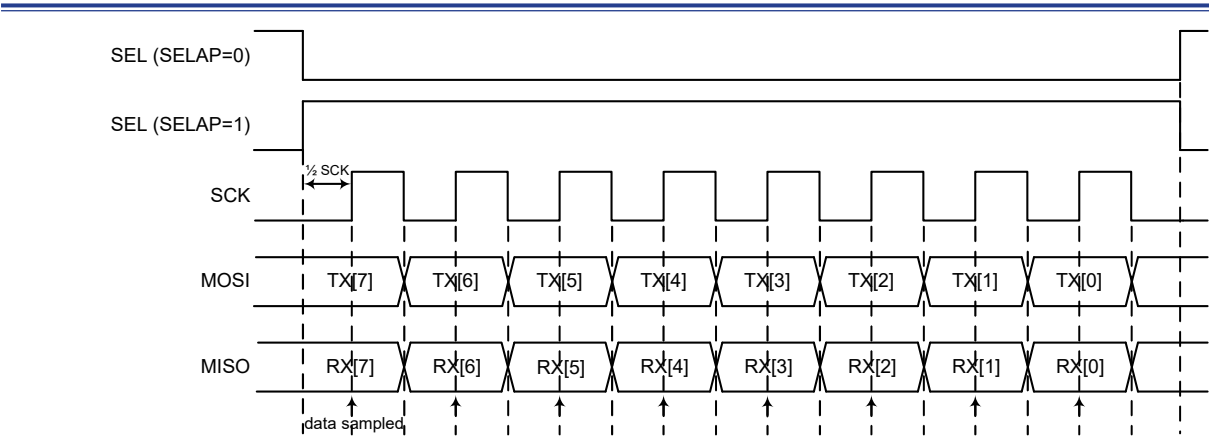


图 134. SPI 单个字节传输时序图 – CPOL = 0, CPHA = 0

下图为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。

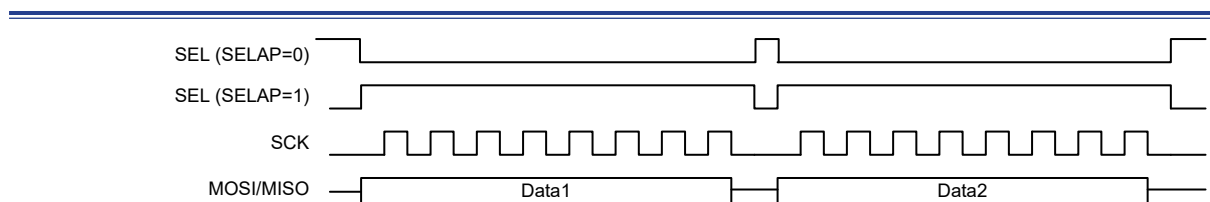


图 135. SPI 连续传输时序图 – CPOL = 0, CPHA = 0

CPOL = 0, CPHA = 1

在此格式下，接收的数据在 SCK 信号下降沿被采样，而发送的数据在 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。下图显示了单个数据字节传输的时序。

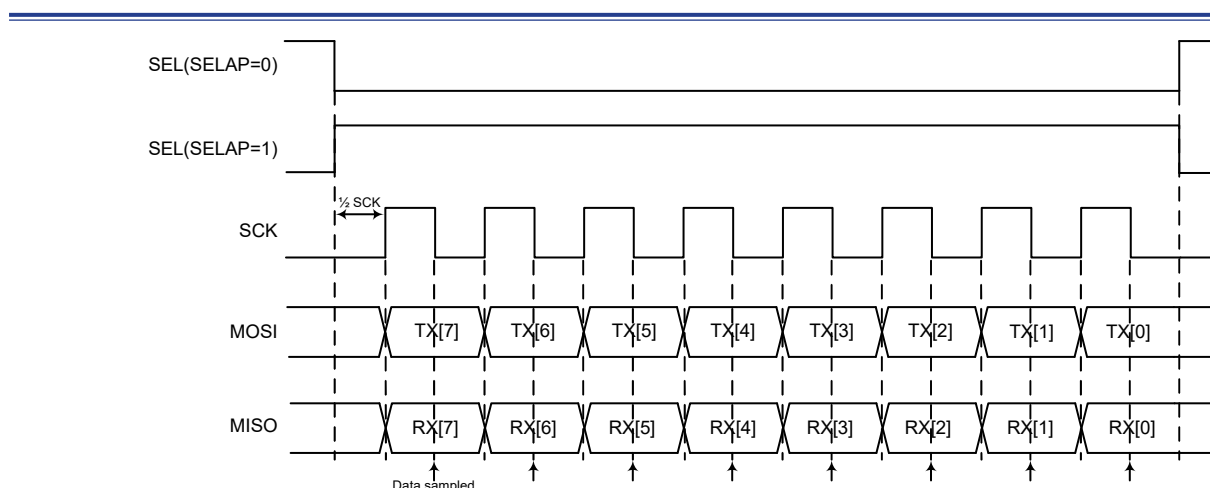


图 136. SPI 单个字节传输时序图 – CPOL = 0, CPHA = 1

下图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。

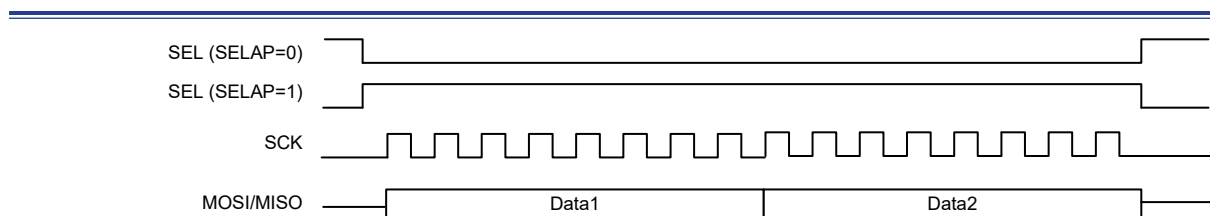


图 137. SPI 连续传输时序图 – CPOL = 0, CPHA = 1

CPOL = 1, CPHA = 0

在此格式下，接收的数据在 SCK 信号下降沿被采样，而发送的数据在 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。下图显示了此格式下单个字节数据传输的时序图。

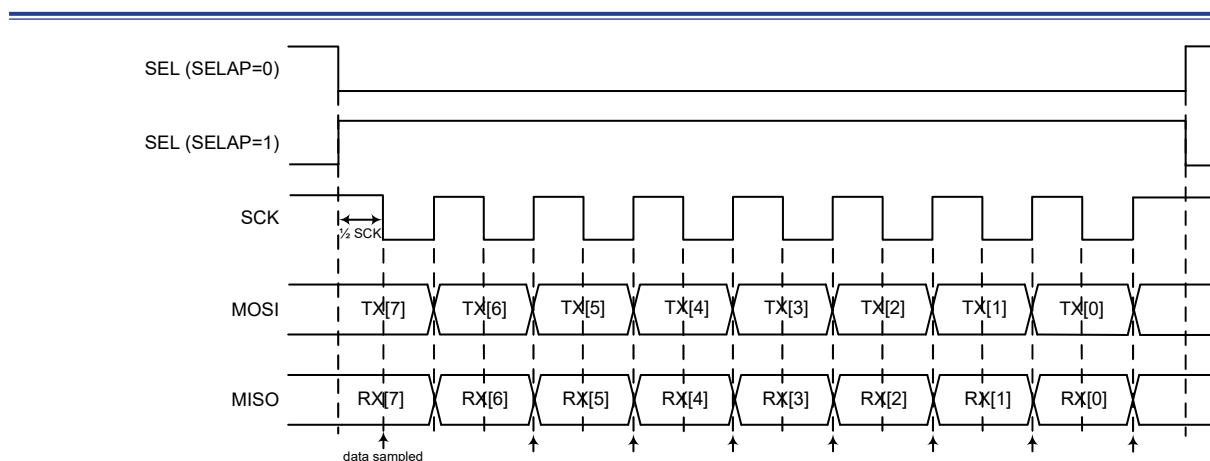


图 138. SPI 单个字节传输时序图 – CPOL = 1, CPHA = 0

下图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。

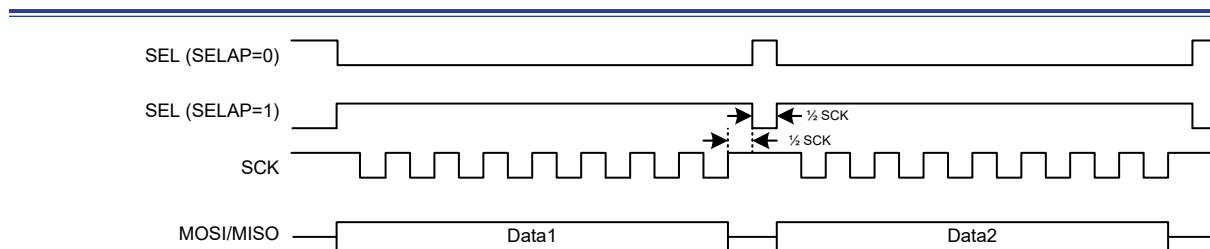


图 139. SPI 连续数据传输时序图 – CPOL = 1, CPHA = 0

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。下图显示了此格式下单个字节数据传输的时序图。

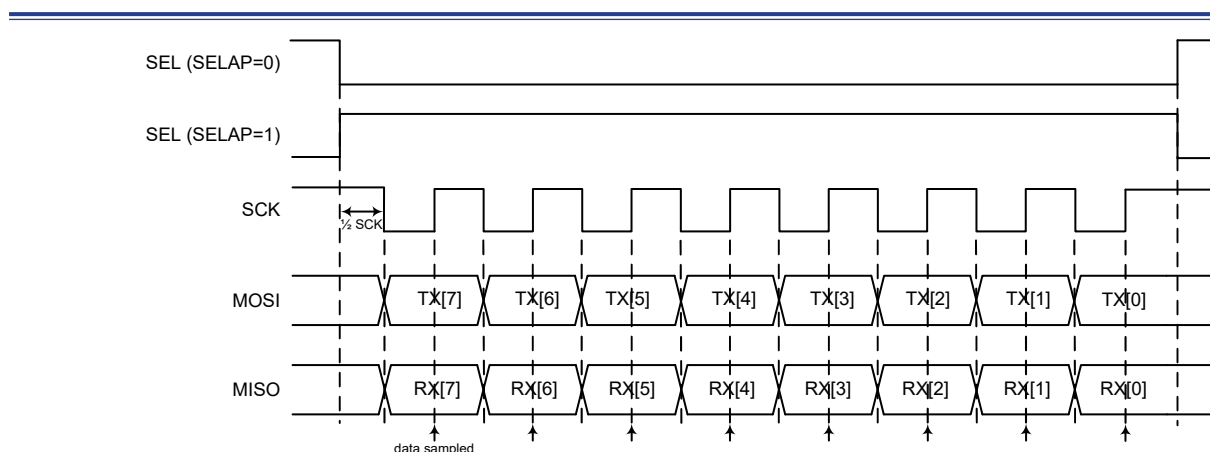


图 140. SPI 单个字节传输时序图 – CPOL = 1, CPHA = 1

下图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。

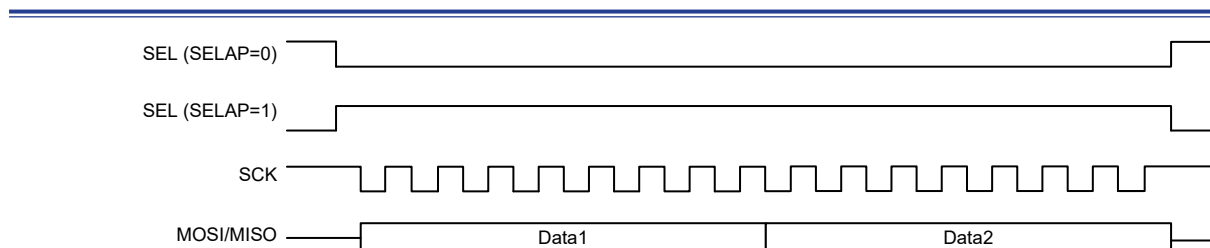


图 141. SPI 连续数据传输时序图 – CPOL = 1, CPHA = 1

SPI 双线读取模式

SPI 接口在主机模式下, 可以配置为双线读取模式。此模式搭配上文提到的四种格式, 能够进行更高效的数据传输。在双线读取模式下, 数据传输只支持输入方向, SPI_MOSI 引脚从输出模式切换为输入模式, 配合 SPI_MISO 引脚, 以两线的方式从外部设备同步读取数据, 支持的数据长度为 16-bit (需设置 DFL = 0x8)。双线读取模式通常用于外部串行 SPI Flash 读取。下图为 SPI 双线读取模式所支持传输格式的时序。

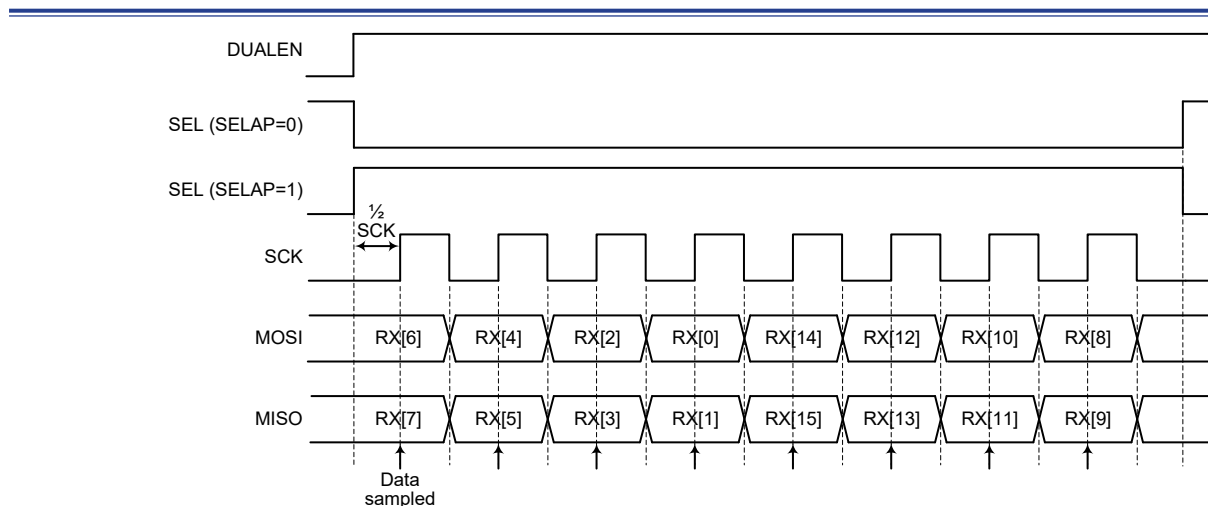


图 142. SPI 双线读取模式时序图 – CPOL = 0, CPHA = 0, DFL = 0x8 (16-bit), MSB 先传

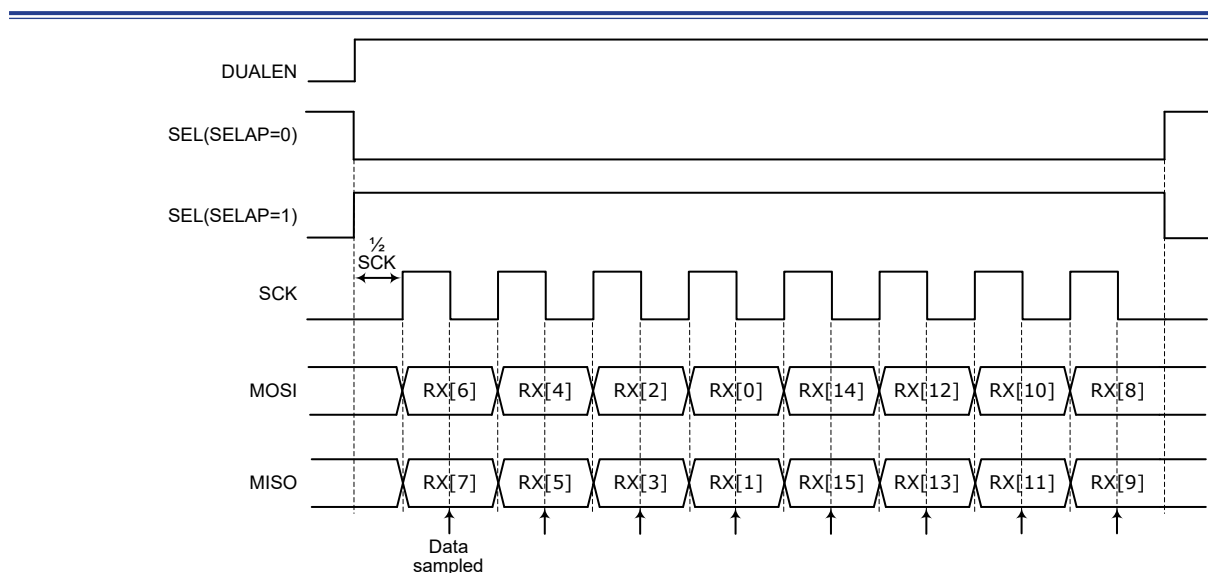


图 143. SPI 双线读取模式时序图 – CPOL = 0, CPHA = 1, DFL = 0x8 (16-bit), MSB 先传

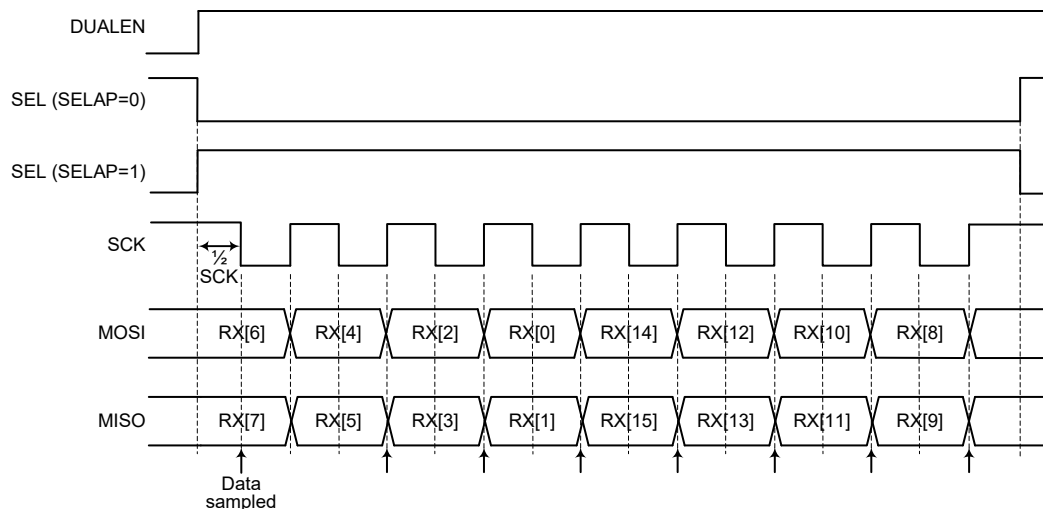


图 144. SPI 双线读取模式时序图 – CPOL = 1, CPHA = 0, DFL = 0x8 (16-bit), MSB 先传

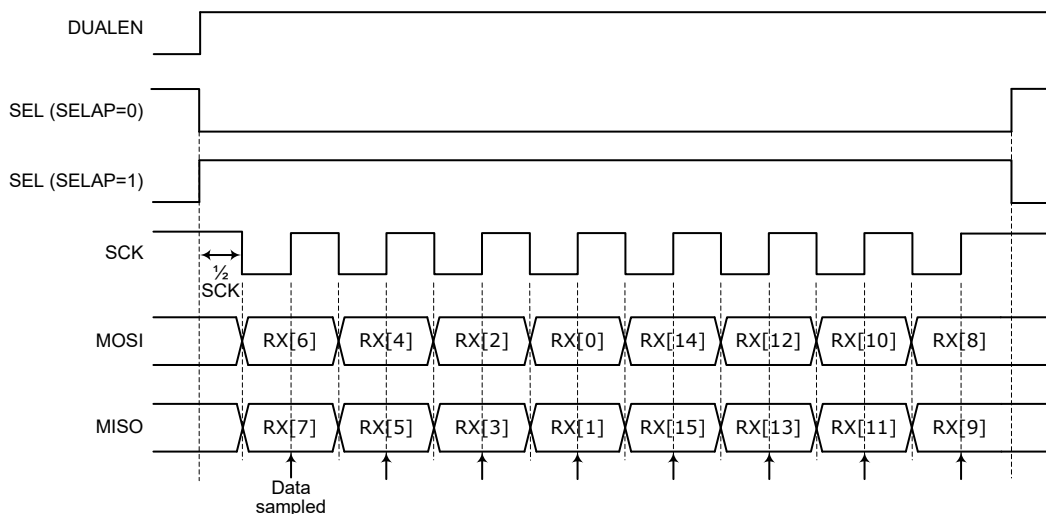


图 145. SPI 双线读取模式时序图 – CPOL = 1, CPHA = 1, DFL = 0x8 (16-bit), MSB 先传

下图为使用 SPI 双线读取模式从外部串行 SPI Flash 读取数据的时序图。

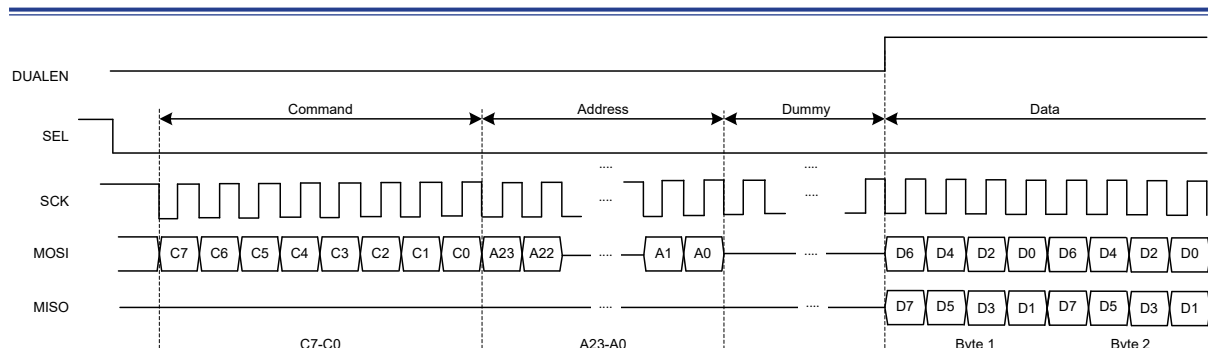


图 146. SPI 双线模式读取数据范例 – CPOL = 1, CPHA = 1

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 位域定义的 TX FIFO 阈值时，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 位域定义的 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 位域定义的 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 位域设置的 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，通过写 0 到超时计数器寄存器的 TOC 位域可除能此功能。当超时功能使能，一旦 SPI RX FIFO 为非空，超时计数器开始计数，当 SPIDR 寄存器数据被读出，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 位域指定的值时，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。当此 SPI_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，需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。

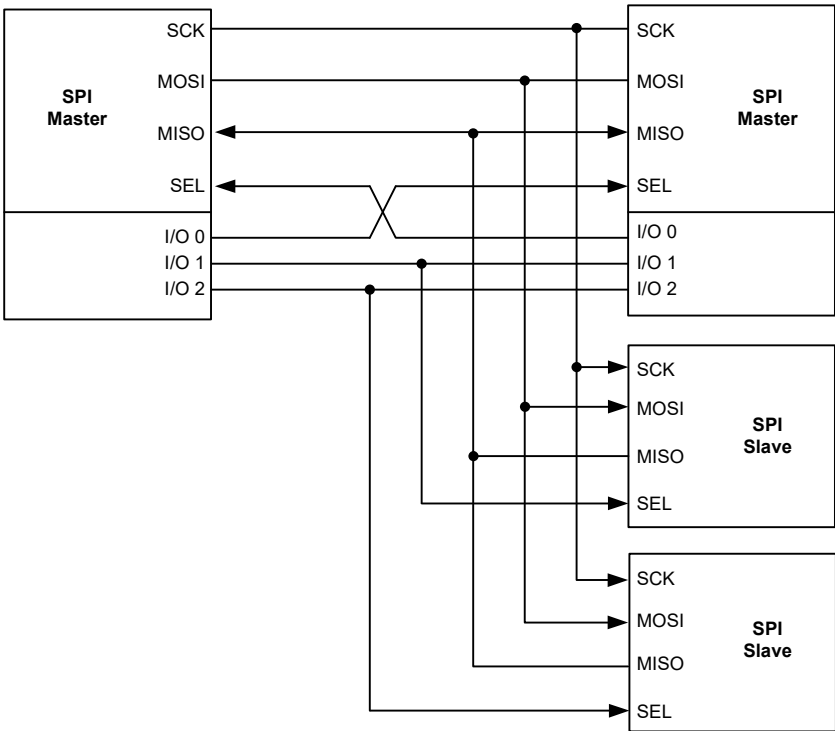


图 147. SPI 多主机从机环境

表 49. SPI 模式故障触发条件

模式故障	描述
触发条件	1. SPI 主机模式。 2. SPICR0 寄存器的 SELOEN = 0 – SPI_SEL 引脚被配置为输入模式。 3. SEL 信号被外部 SPI 主机驱动且转换为有效电平。
SPI 动作	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 50. SPI 主机模式 SPI_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 SPI_SEL 引脚功能		SPI_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	情况 1	情况 2	情况 1	情况 2
	不支持	支持	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 信号才会变为无效。
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，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 位域定义。

PDMA 接口

PDMA 接口集成于 SPI 模块。在发送和接收模式下可分别通过设置 TXDMAE 或 RXDMAE 位为 1 来使能 PDMA 功能。在 TX 缓存器空标志位 TXBE 被置 1 且 TXDMAE 位置 1，PDMA 功能将被激活用于把数据从用户指定的存储器位置移到 SPI 数据寄存器或 TX FIFO 中，直到 TXBE 位被清零。在非 FIFO 模式下 TX 缓存器为空时，或在 FIFO 模式下当 TX FIFO 包含的数据长度等于或小于 SPIFCR 寄存器中 TXFTLS 位域定义的 TX FIFO 阈值时，TXBE 标志位被置位。

同样地，在 RX 缓存器非空标志位 RXBNE 被置 1 且 RXDMAE 位置 1，PDMA 功能被激活用于把数据从 SPI 数据寄存器或 RX FIFO 移到用户指定的存储器位置，直到 RXBNE 标志位被清零。在非 FIFO 模式下 RX 缓存器中有有效数据，或在 SPI FIFO 模式下 RX FIFO 包含的数据长度等于或大于 SPIFCR 寄存器中 RXFTLS 位域定义的 RX FIFO 阈值时，RXBNE 标志位被置位。

PDMA 配置详细描述可参考相应的 PDMA 章节。

寄存器列表

下表显示了 SPI 寄存器及其复位值。

表 51. SPI 寄存器列表

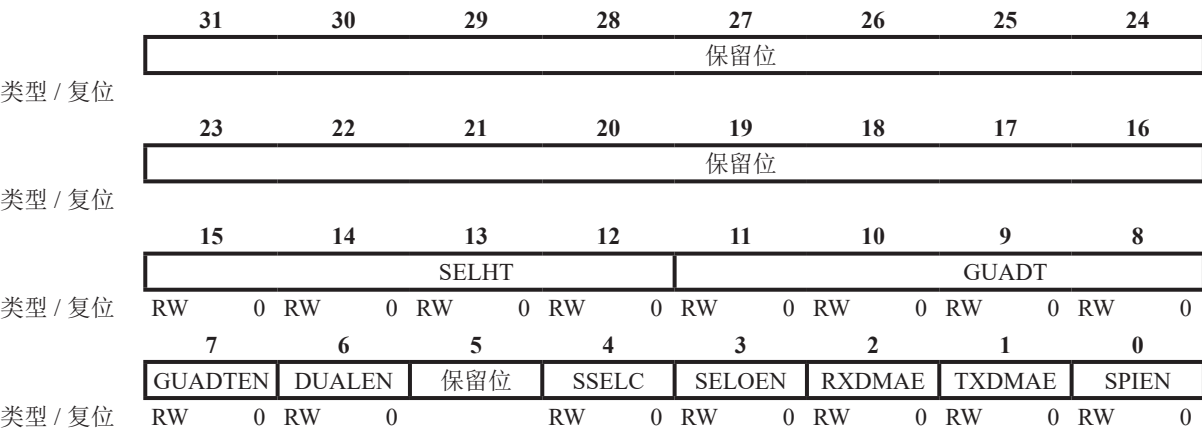
寄存器	偏移量	描述	复位值
SPICR0	0x000	SPI 控制寄存器 0	0x0000_0000
SPICR1	0x004	SPI 控制寄存器 1	0x0000_0000
SPIIER	0x008	SPI 中断使能寄存器	0x0000_0000
SPICPR	0x00C	SPI 时钟预分频器寄存器	0x0000_0000
SPIDR	0x010	SPI 数据寄存器	0x0000_0000
SPISR	0x014	SPI 状态寄存器	0x0000_0003
SPIFCR	0x018	SPI FIFO 控制寄存器	0x0000_0000
SPIFSR	0x01C	SPI FIFO 状态寄存器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



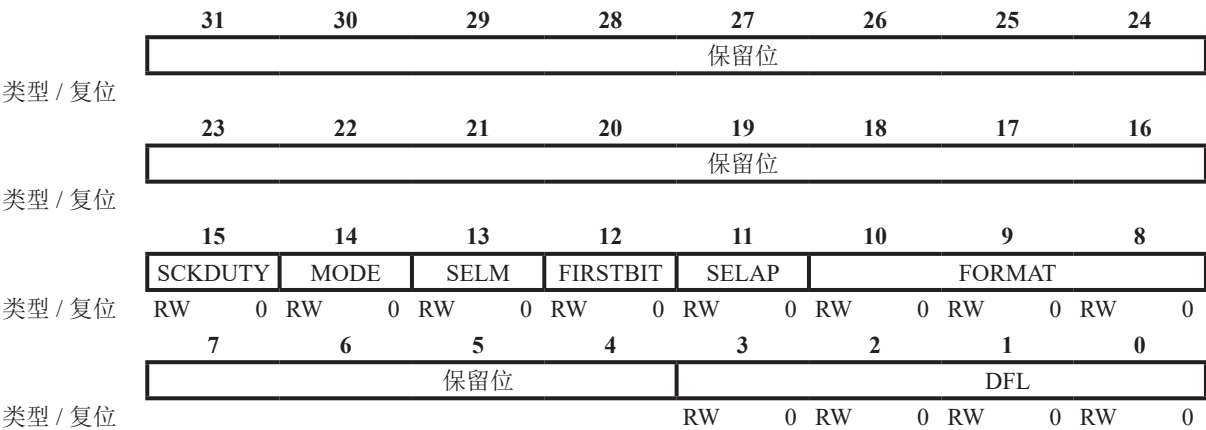
位	域	描述
[15:12]	SELHT	芯片选择保持时间控制 0x0: 1/2 SCK 0x1: 1 SCK 0x2: 3/2 SCK 0x3: 2 SCK 注意，SELHT 位选择仅主机模式有效。

位	域	描述
[11:8]	GUADT	保护时间控制 GUADTEN = 1 0x0: 1 SCK 0x1: 2 SCK 0x2: 3 SCK ... 注意，GUADT 位选择仅在 SPI 主机模式有效。
[7]	GUADTEN	保护时间使能位 0: 保护时间为 1/2 SCK 1: 保护时间由 GUADT 位控制 注意，GUADTEN 位仅在 SPI 主机模式有效。
[6]	DUALEN	双线模式使能位 0: 双线模式除能 1: 双线模式使能 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 控制位仅在 SPI 主机模式有效。
[4]	SSELC	软件从机选择控制位 0: 设置 SEL 输出为无效电平 1: 设置 SEL 输出为有效电平 软件可以通过设置 SSELC 位使 SEL 输出为一个有效或无效电平。有效电平极性通过 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。
[2]	RXDMAE	RX PDMA 请求使能位 0: SPI RX 通道 PDMA 请求除能 1: SPI RX 通道 PDMA 请求使能
[1]	TXDMAE	TX PDMA 请求使能位 0: SPI TX 通道 PDMA 请求除能 1: SPI TX 通道 PDMA 请求使能
[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	域	描述
[15]	SCKDUTY	SCK 脉冲高 / 低占空比 0: High-Duty 比 Low-Duty 长 1: High-Duty 比 Low-Duty 短 SCK 的频率由 fCLK/n 计算。当 n 为奇数时，SCK 高 / 低占空比不是 50%。在这种情况下，可以通过该控制位选择高 / 低占空比的比率。
[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

位 域 描述

[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
其它值	保留位	

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

[3:0] DFL 数据帧长度
该位域用于选择数据传输的数据帧长度。

DFL[3:0]	SPI 串行模式	SPI 双线模式
0x1	1 位	—
0x2	2 位	—
0x3	3 位	—
0x4	4 位	—
0x5	5 位	—
0x6	6 位	—
0x7	7 位	—
0x8	8 位	16 位
0x9	9 位	—
...	...	—
0xF	15 位	—
0x0	16 位	—

注: 1. 数据传输时一帧包含的位数由 DFL 位域以及 SPI 传输模式共同决定。

2. 以 16-bit 数据传输为例, DFL 位域的设置如下所示:

对于 SPI 串行模式: 数据帧长度 = $16/1 = 16$, DFL = 0x0;

对于 SPI 双线模式: 数据帧长度 = $16/2 = 8$, DFL = 0x8;

3. SPI 双线模式仅支持 16-bit 数据长度。

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

类型 / 复位	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
	TOIEN	SAIEN	MFIEN	ROIEN	WCIEN	RXBNEIEN	TXEIEN	TXBEIEN
类型 / 复位	RW	0	RW	0	RW	0	RW	0

位	域	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
[2]	RXBNEIEN	RX 缓存器非空中断使能位 0: 除能 1: 使能 若 RXBNEIEN 位为高，当 RXBNE 标志位被置位时，会产生 RX 缓存器非空中断请求。 在 FIFO 模式下，中断请求的产生取决于 RX FIFO 触发级别的设置。
[1]	TXEIEN	传输寄存器空中断使能位 0: 除能 1: 使能 若 TXEIEN 位为高，当 TXE 标志位被置位时，将产生传输寄存器空中断请求。
[0]	TXBEIEN	TX 缓存器空中断使能位 0: 除能 1: 使能 若 TXBEIEN 位为高，当 TXBE 标志位被置位时，会产生 TX 缓存器空中断请求。在 FIFO 模式下，中断请求的产生取决于 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
	7	6	5	4	3	2	1	0	
类型 / 复位	CP								
	RW	0	RW	0	RW	0	RW	0	RW

位	域	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： $f_{SCK} = f_{PCLK} / (CP + 1)$ ，CP 范围：1 ~ 65535 注：在 SPI 主机模式下，APB 时钟 (f_{PCLK}) 不得小于 SPI SCK 输出频率的 2 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		
	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		
	DR									
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0		
	DR									
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW	0

位	域	描述
[15:0]	DR	数据寄存器 SPI 数据寄存器用来储存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返回接收缓存器即 RX 缓存器中的数据。

该寄存器包含了相关的 SPI 状态。

复位值: 0x0000_0003

	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							BUSY	
									RO 0
	7	6	5	4	3	2	1	0	
类型 / 复位	TO	SA	MF	RO	WC	RXBNE	TXE	TXBE	
	WC	0 WC	0 WC	0 WC	0 WC	0 RO	0 RO	1 RO	1

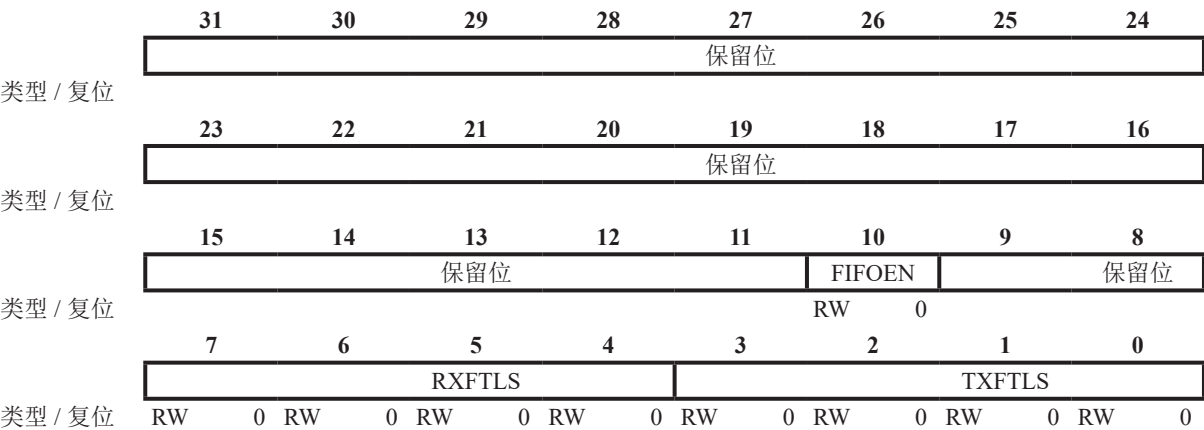
位	域	描述
[8]	BUSY	<p>SPI 忙标志位</p> <p>0: SPI 不忙</p> <p>1: SPI 忙</p> <p>在主机模式，当 TX 缓存器和 TX 移位寄存器都为空时，此位被复位。当 TX 缓存器或 TX 移位寄存器不为空时，此位被置位。</p> <p>在从机模式，当 SEL 信号转变为有效电平时，此位被置位。当 SEL 信号转变为无效电平时，此位被复位。</p>
[7]	TO	<p>超时标志位</p> <p>0: 无 RX FIFO 超时</p> <p>1: RX FIFO 超时发生</p> <p>一旦超时计数器的值等于 SPIFTOCR 寄存器中 TOC 位域的设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

位	域	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 位域定义的触发级别时，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，此位置位表明存放在 TX FIFO 的数据量等于或小于 SPIFCR 寄存器中 TXFTLS 位域定义的触发级别。

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	域	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。
[7:4]	RXFTLS	RX FIFO 触发级别选择 0000: 触发级别为 0 0001: 触发级别为 1 ... 1000: 触发级别为 8 其它值: 保留 RXFTLS 位域用来定义 RX FIFO 触发级别。当 RX FIFO 中的数据量等于或大于 RXFTLS 位域定义的触发级别时，RXBNE 标志将被置位。

位	域	描述
[3:0]	TXFTLS	TX FIFO 触发级别选择 0000: 触发级别为 0 0001: 触发级别为 1 ... 1000: 触发级别为 8 其它值: 保留 TXFTLS 位域用来定义 TX FIFO 触发级别。当 TX FIFO 中的数据量等于或小于 TXFTLS 位域定义的触发级别时, TXBE 标志位被置位。

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
	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
	RXFS TXFS
类型 / 复位	RO 0 RO 0 RO 0 RO 0 RO 0 RO 0 RO 0 RO 0

位	域	描述
[7:4]	RXFS	RX FIFO 状态 0000: RX FIFO 为空 0001: RX FIFO 包含 1 个数据 ... 1000: RX FIFO 包含 8 个数据 其它值: 保留
[3:0]	TXFS	TX FIFO 状态 0000: TX FIFO 为空 0001: TX FIFO 包含 1 个数据 ... 1000: TX FIFO 包含 8 个数据 其它值: 保留

SPI FIFO 超时计数器寄存器 – SPIFCR

该寄存器存储了 SPI RX FIFO 超时计数器比较值。

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	域	描述
[15:0]	TOC	超时计数器比较值 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没有通过软件从 SPIDR 寄存器被读取数据，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 位域，SPI FIFO 超时功能可除能。超时计数器由系统 APB 时钟 f _{CLK} 驱动。

21 通用同步异步收发器 (USART)

简介

通用同步异步收发器 USART，提供了一种灵活的同步或异步传输的全双工数据交换方式。USART 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。USART 外设功能支持多种类型的中断。

USART 模块包含一个 8 级的发送器 FIFO (TX FIFO) 和一个 8 级的接收器 FIFO (RX FIFO)。通过读取 USART 状态 & 中断标志位寄存器 USRSIFR，软件可检测 USART 的错误状态。这些状态包括传输操作运行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。

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

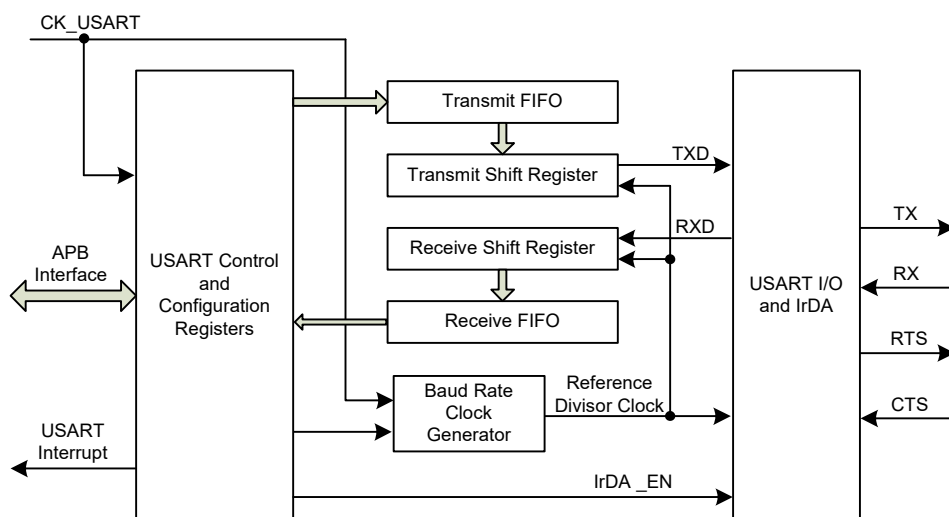


图 148. USART 方框图

特性

- 支持异步和时钟同步串行通信模式
- 全双工通信能力
- 可编程波特率：异步模式下高达 ($f_{PCLK} / 16$) MHz，同步模式下高达 ($f_{PCLK} / 8$) MHz。
- IrDA SIR 编码器和解码器
 - 支持正常 3/16 位持续时间和低功耗 (1.41 ~ 2.23 μ s) 持续时间
- 支持带有输出使能的 RS485 模式
- 自动硬件流控制模式 – RTS, CTS
- 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
- FIFO：
 - 接收器 FIFO：8 级 (最多 9 个数据位)
 - 发送器 FIFO：8 级 (最多 9 个数据位)
- 支持 PDMA 接口

功能描述

串行数据格式

USART 模块对写入在发送器 FIFO 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 或 MSB 优先的数据位，可选的奇偶校验位和最后 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，并在下一个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。

USART 模块也对从接收器 FIFO 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，然后寻找停止位。如果停止位没有找到，USART 模块会认为整个字的传输失败并以帧错误作为响应。

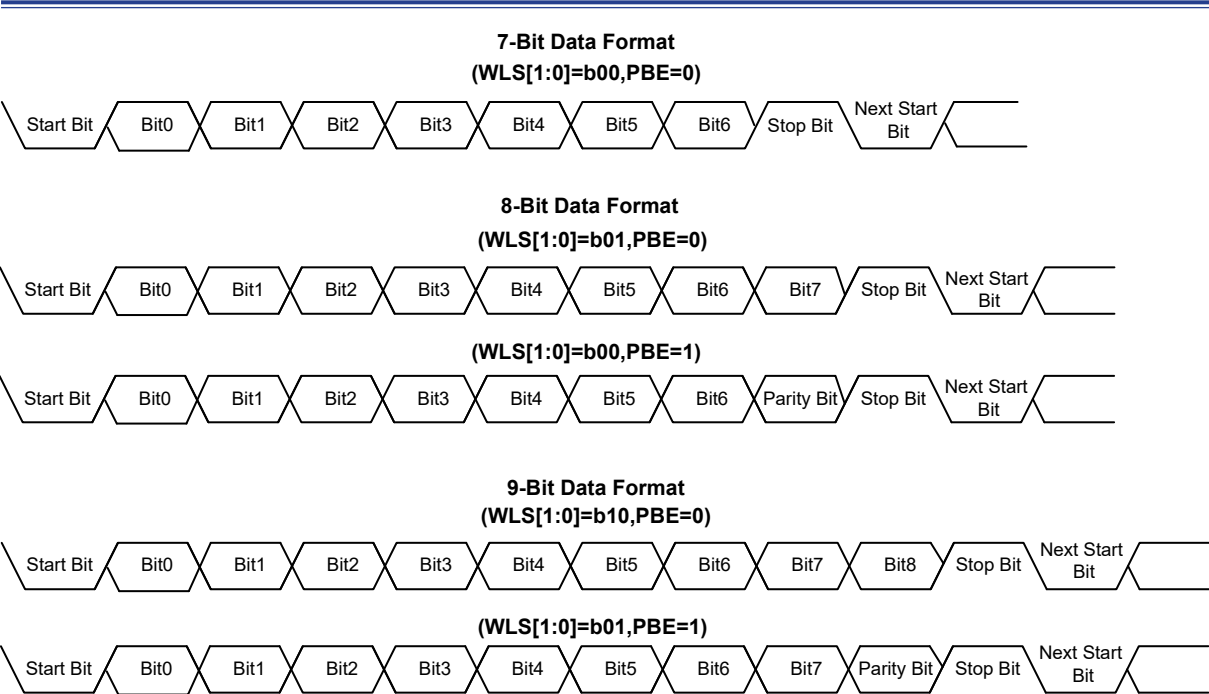


图 149. USART 串行数据格式

波特率发生器

USART 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 USART 时钟 CK_USART 的关系如下。

波特率时钟 = CK_USART / BRD

CK_USART 时钟是连接到 USART 模块的 APB 时钟, 异步模式中 BRD 的范围是 16 ~ 65535, 在同步模式中 BRD 的范围则为 8 ~ 65535。

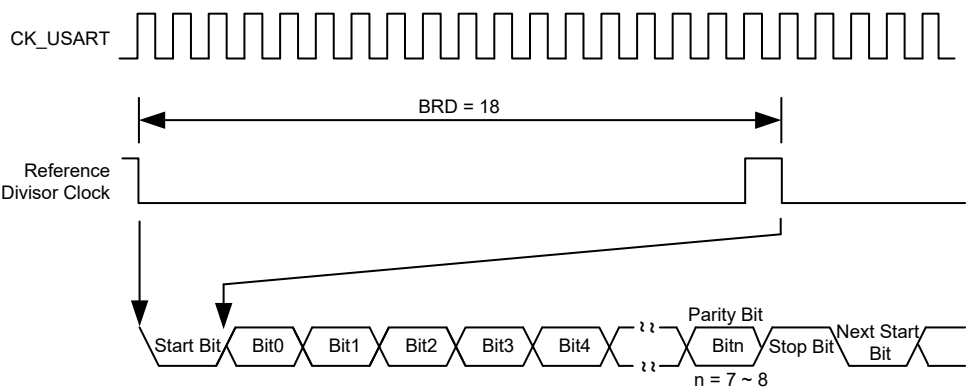


图 150. USART 时钟 CK_USART 和数据帧时序

表 52. 波特率误差计算 – CK_USART = 40 MHz

波特率		CK_USART = 40 MHz		
No.	Kbps	实际值	BRD	误差率
1	2.4	2.4	16667	0.00%
2	9.6	9.6	4167	-0.01%
3	19.2	19.2	2083	0.02%
4	57.6	57.6	694	0.06%
5	115.2	115.3	347	0.06%
6	230.4	229.9	174	-0.22%
7	460.8	459.8	87	-0.22%
8	921.6	930.2	43	0.94%
9	2250	2222.2	18	-1.23%
10	3000	—	—	—

表 53. 波特率误差计算 – CK_USART = 48 MHz

波特率		CK_USART = 48 MHz		
No.	Kbps	实际值	BRD	误差率
1	2.4	2.4	20000	0.00%
2	9.6	9.6	5000	0.00%
3	19.2	19.2	2500	0.00%
4	57.6	57.6	833	0.04%
5	115.2	115.1	417	-0.08%
6	230.4	230.8	208	0.16%
7	460.8	461.5	104	0.16%
8	921.6	923.1	52	0.16%
9	2250	2285.7	21	1.59%
10	3000	3000	16	0.00%

表 54. 波特率误差计算 – CK_USART = 60 MHz

波特率		CK_USART = 60 MHz		
No.	Kbps	实际值	BRD	误差率
1	2.4	2.4	25000	0.00%
2	9.6	9.6	6250	0.00%
3	19.2	19.2	3125	0.00%
4	57.6	57.6	1042	-0.03%
5	115.2	115.1	521	-0.03%
6	230.4	230.8	260	0.16%
7	460.8	461.5	130	0.16%
8	921.6	923.1	65	0.16%
9	2250	2285.7	27	-1.23%
10	3000	3000	20	0.00%

硬件流控制

USART 支持硬件流控制功能，通过 USRCR 寄存器中的 HFCEN 位置 1 使能。可通过 CTS 输入和 RTS 输出控制 2 个 USART 设备之间的串行数据流。下图显示了该模式下的连接图。硬件流控制功能分为两种类型，一个是 RTS 流控制功能，一个是 CTS 流控制功能。

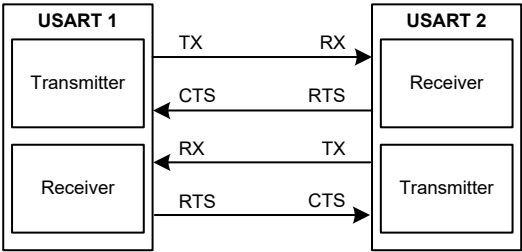


图 151. 两个 USART 设备之间的硬件流控制

RTS 流控制功能

在 RTS 流控制中, 当接收数据寄存器为空时, USART RTS 引脚会输出有效电平 (低电平)。也就是说接收器做好接收新数据的准备。当 RX FIFO 达到由 USRFCR 寄存器中 RFTL 位域定义的触发级别时, USART RTS 引脚会输出无效电平 (高电平)。下图显示了一个 RTS 流控制的范例。

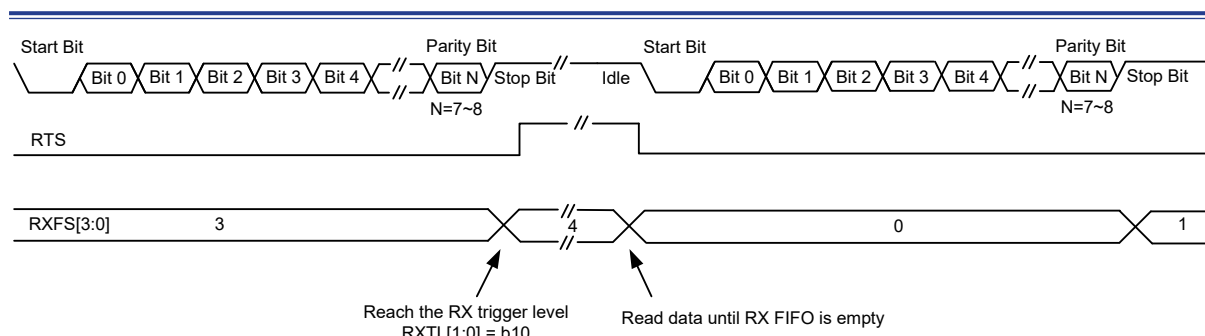


图 152. USART RTS 流控制

CTS 流控制功能

如果硬件流控制功能使能, USRCR 寄存器中的 URTXEN 位将由 USART CTS 输入信号来控制。如果 USART CTS 引脚被拉成有效电平 (低电平), 则 URTXEN 位将自动置 1 使能数据发送。但是, 如果 USART CTS 引脚被拉成无效电平 (高电平), 则 URTXEN 位将清零除能数据发送。

若在 USART 数据发送过程中, 将 CTS 引脚拉成高电平, 则当前数据将继续发送直到停止位发送完成。下图显示了一个 CTS 流控制通信的范例。

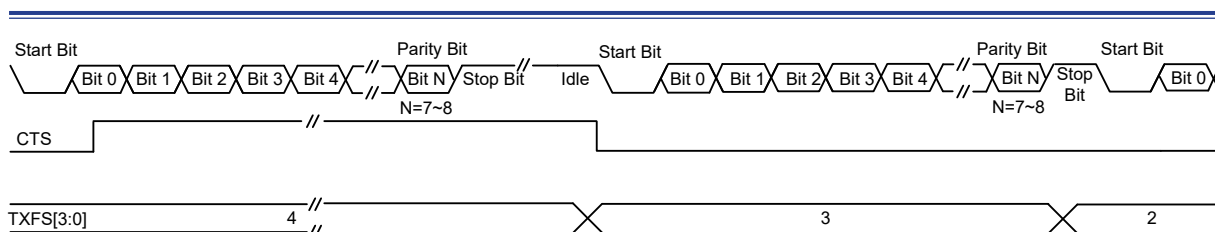


图 153. USART CTS 流控制

IrDA 模式

USART IrDA 模式提供半双工点对点无线通信。

USART 模块包括一个集成的调制器和解调器，它允许使用红外线收发器进行无线通信。在 IrDA 模式中，发送器把一个逻辑数据“0”作为一个“高”脉冲发送，一个逻辑数据“1”作为一个“低”电平发送，而接收器把一个“低脉冲”解读为逻辑数据“0”，把一个“高”电平解读为逻辑数据“1”。

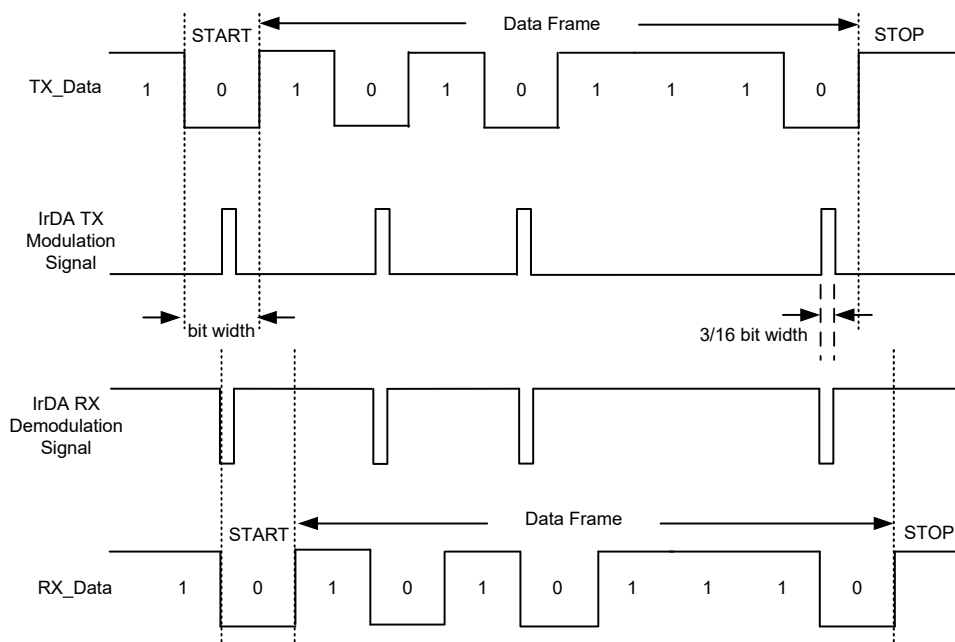


图 154. IrDA 调制与解调

IrDA 模式提供两种工作模式，一种是正常模式，另一种是低功耗模式。

IrDA 正常模式

在 IrDA 正常模式里，由发送调制器生成的每个发射脉冲宽度规定为波特率时钟周期的 3/16。IrDA 接收器解调器接收到的脉冲宽度取决于 IrDA 接收去抖滤波器，它是由一个 8-bit 向下计数的计数器定义的。去抖滤波器的计数器的值由 IrDACR 寄存器的 IrDAPSC 位域定义。当在接收器引脚检测到一个下降沿时，去抖滤波器计数器在 CK_USART 时钟驱动下开始向下计数。当接收器引脚上检测到一个上升沿时，计数器停止计数，并重新载入 IrDAPSC 值。当在接收器引脚检测到一个低脉冲下降沿，且在去抖滤波器计数到零之前检测到上升沿时，则此低脉冲将被视为干扰噪声，被丢弃。当在接收器引脚检测到一个低脉冲下降沿，但在去抖滤波器计数到零之前没有检测到上升沿时，则在此位的持续时间内，USART 接收器引脚上的输入信号将被视为一个有效数据“0”。IrDAPSC 值必须设置为大于或等于 0x01，这样 IrDA 接收器解调操作才可以正常进行。可以调整 IrDAPSC 值以符合 USART 波特率的设置，从而将 IrDA 接收到的宽度小于预分频器设置时间的干扰噪声过滤掉。

IrDA 低功耗模式

在 IrDA 低功耗模式里,发送器调制器产生的传输脉冲宽度不再持续 3/16 个波特率时钟周期,而是一个固定值,这个值可由下列公式计算。设置 IrDAPSC 位域来调整发送的脉冲宽度以符合外部 IrDA 接收器的最低脉冲宽度规格。

$$T_{\text{IrDA_L}} = 3 \times \text{IrDAPSC} / \text{CK_USART}$$

注: $T_{\text{IrDA_L}}$ 是在低功耗模式下传输的 IrDA 脉冲宽度。

IrDA 控制寄存器 IrDACR 中的 IrDAPSC 位域是 IrDA 分频器的值。

IrDA 低功耗模式接收的去抖操作类似于 IrDA 正常模式下的操作。为了滤除尖峰干扰脉冲,应滤除宽度小于 $1 \times (\text{IrDAPSC} / \text{CK_USART})$ 的低脉冲,只有低电平持续时间大于 $2 \times (\text{IrDAPSC} / \text{CK_USART})$ 的脉冲信号才被接收作为有效数据。

IrDA 物理层规范规定发送和接收之间最小要有 10 ms 的延时 (IrDA 是一个半双工协议), IrDA 接收器的建立时间需通过软件进行设置。

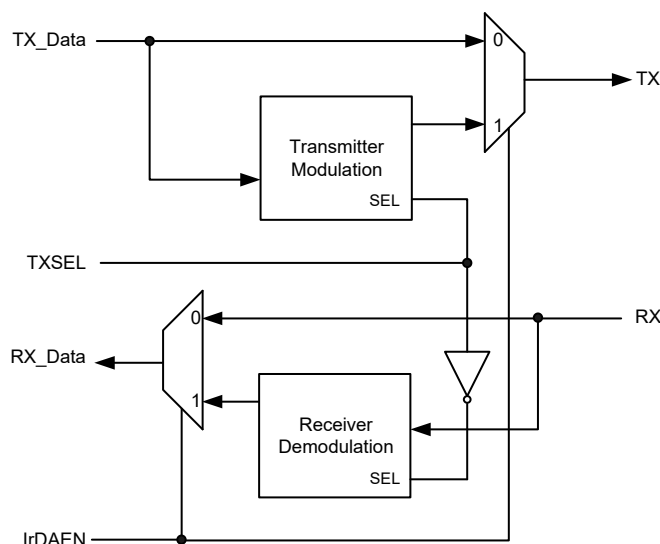


图 155. USART I/O 和 IrDA 方框图

RS485 模式

在 USART 的 RS485 模式中，接口上的数据通过 2 线式双绞线传输。RS485 收发器通过参照第三方共模参考电压值来解析差分信号的电压准位。如果没有这个共模参考，收发器可能会错误地解析差分信号。这增强了 RS485 接口的抗干扰能力。当 USART 模块工作在 RS485 模式下时，USART RTS 引脚用来控制外部 RS485 收发器，此引脚使能时的极性可以由 RS485 控制寄存器 RS485CR 中的 TXENP 位来配置。

RS485 自动方向控制模式 – AUD

当 RS485 模式被配置为主机发送器时，其将工作在自动方向控制模式，即 AUD 模式。在 RS485 的 AUD 模式下，USART RTS 引脚的极性根据 RS485 控制寄存器的 TXENP 位来配置。此引脚用来控制外部 RS485 收发器的发送器使能。

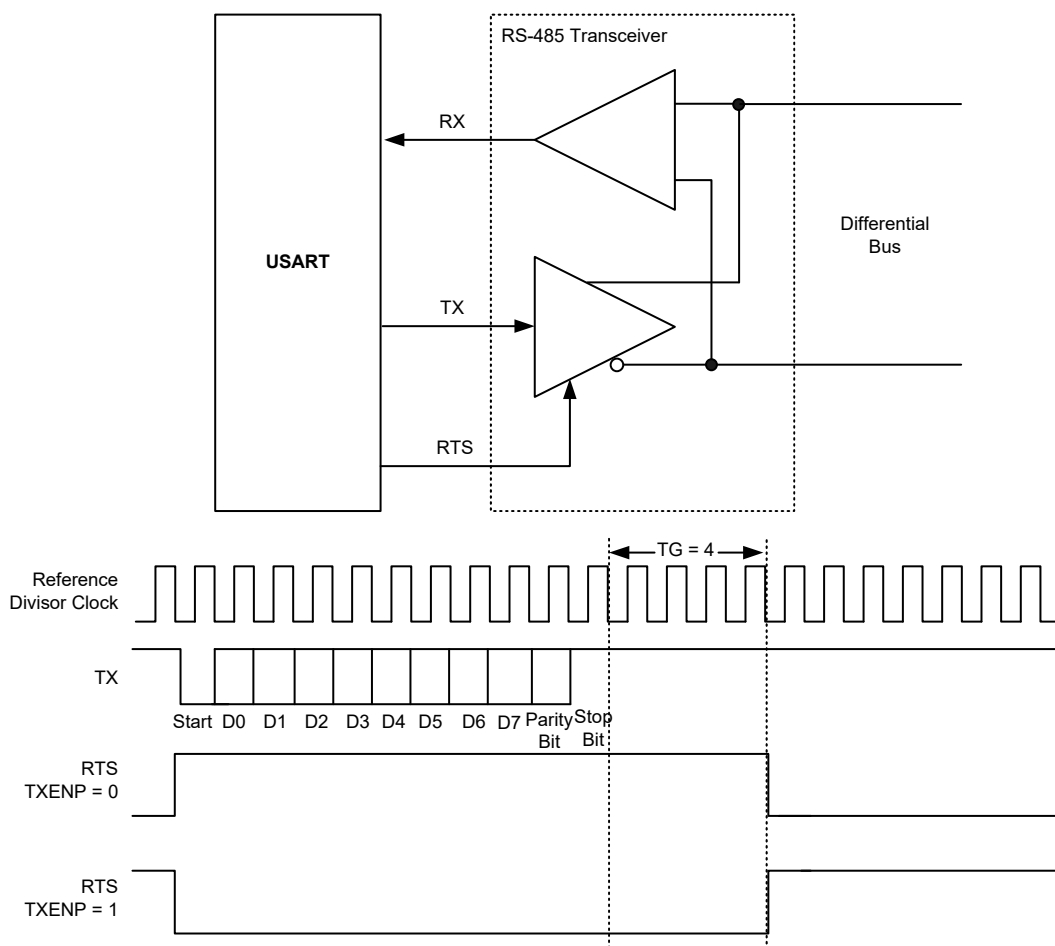


图 156. RS485 接口和波形

RS485 正常多点通信工作模式 – NMM

当 RS485 接口配置为可寻址从机时,可工作在正常多点通信工作模式,即 NMM 模式。当 RS485CR 寄存器中的 RSNMM 位被置位时,此模式使能。无论 USRCR 寄存器中的 URRXEN 位为何值,所有接收到的校验位为“0”的数据都将被忽略,直到检测到第一个校验位为“1”的地址字节,这时,接收到的地址字节将存入到 RX FIFO 中。一旦检测到第一个地址字节数据并存入 RX FIFO 时,USRSIFR 寄存器中的 RSADD 标志位将被置位,如果 USRIER 寄存器中的 RSADDIE 位置 1,将产生中断。应用程序通过配置 URRXEN 位决定接收器是否使能来接收后面的数据。当 URRXEN 位置 1 使能接收器,所有接收到的数据都将被存入 RX FIFO,若是 URRXEN 位清零除能接收器时,这些数据将被忽略。

RS485 自动地址检测工作模式-AAD

当 RS485 接口配置为可寻址从机时,除了正常多点通信工作模式外,RS485 接口还可以工作在自动地址检测工作模式,即 AAD 模式。此模式由 RS485CR 寄存器中的 RSAAD 位置 1 使能。接收器检测到带有校验位“1”的地址帧后,将此地址数据和 RS485CR 寄存器中 ADDMATCH 位域定义的 8 位地址值相比较。若匹配,则将数据存储在 RX FIFO 中且硬件自动置位 URRXEN 以使能接收器。当接收器使能,所有接收到的数据都将存储在 RX FIFO 中,直到地址帧和 ADDMATCH 值不匹配的情况发生为止,这时,接收器将自动除能。接收器使能后,软件可通过将 URRXEN 位清为“0”,来除能接收器。

同步主机模式

USART 同步主机模式中, 用全双工方式传输数据, 即数据发送和接收同时发生, 且只支持主机模式。USART CTS 引脚用于同步 USART 发送器的时钟输出。在这种模式下, 起始位、校验位和停止位期间内, 没有时钟脉冲被送到 CTS 引脚。同步控制寄存器 SYNCR 中的 CPS 位可以用于决定是在第一个还是第二个时钟脉冲边沿捕捉数据。SYNCR 寄存器中的 CPO 位用来配置 USART 同步模式空闲状态时时钟的极性。具体的时序信息如下图所示。

在 USART 同步模式下, USART CTS / SCK 时钟输出引脚仅在发送数据到从机时使用。如果发送数据寄存器 USRDR 被写入有效的数据, USART 同步模式下会在相应时钟作用下发送数据, USART 接收器会通过 RX 引脚接收数据。否则, 若没有要发送的数据, 接收器将无法获得同步数据。

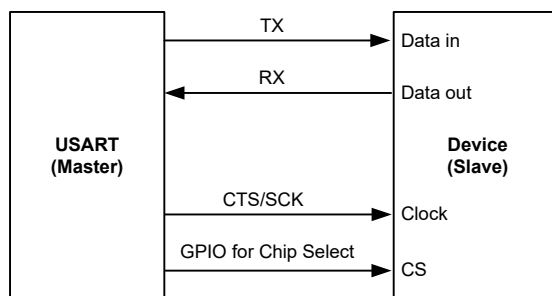


图 157. USART 同步传输范例

注：USART 仅支持同步主机模式。它不能在输入时钟作用下接收或发送数据。USART CTS / SCK 时钟始终是输出信号。

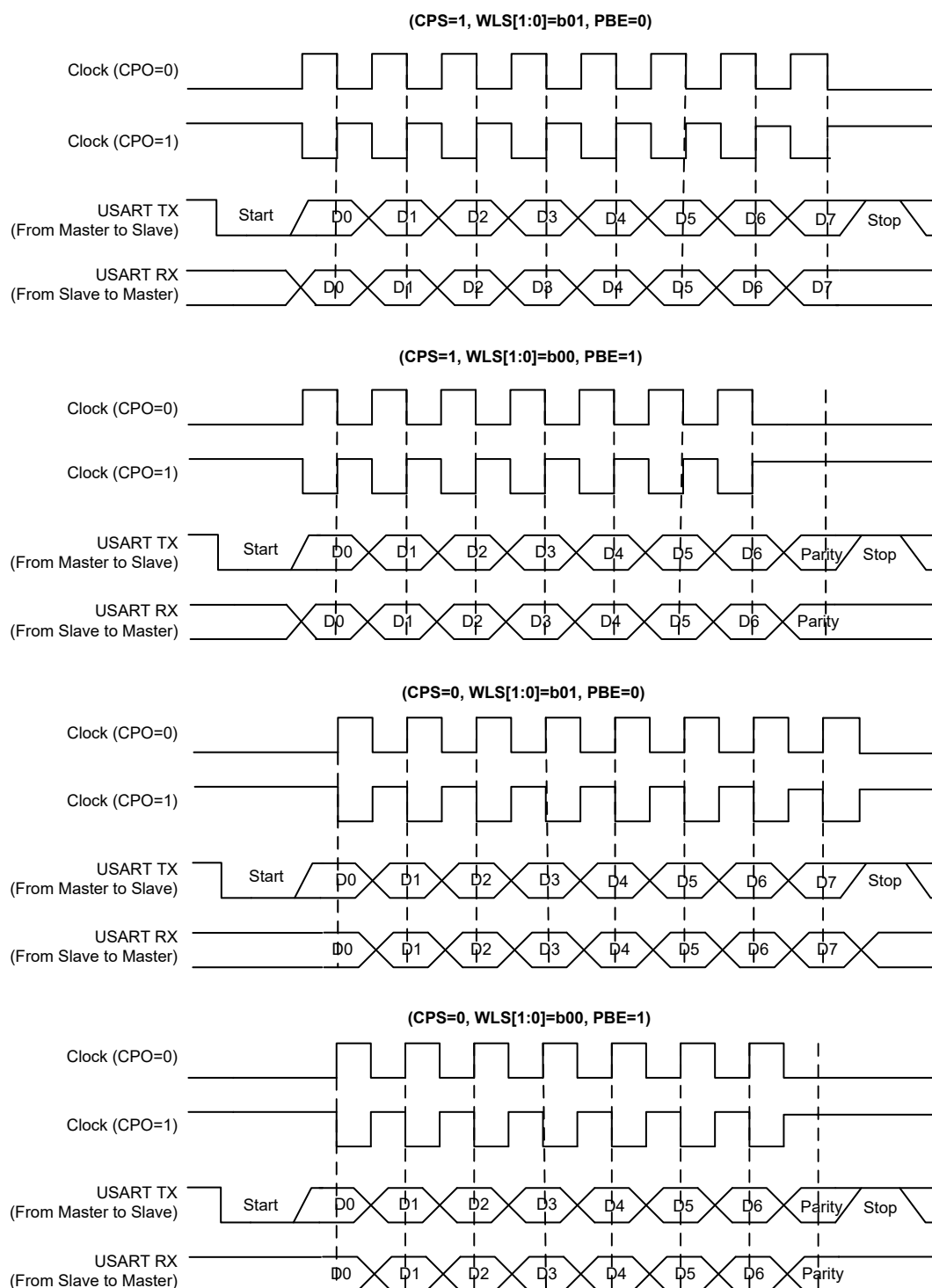


图 158. 8-bit 格式 USART 同步波形

中断和状态

下列事件发生时，若相应的中断使能位被置位，USART 模块可产生一个中断：

- 接收器 FIFO 超时中断：USART 接收器 FIFO 非空且在指定的超时时间内未收到新的数据包。
- 接收器线路状态中断：USART 溢出错误、奇偶错误、帧错误或线中止事件发生。
- 发送器 FIFO 阈值级别中断：USART 发送器 FIFO 中要传输的数据量少于指定的阈值级别。
- 发送完成中断：发送 FIFO 为空且发送移位寄存器 TSR 的内容已全部移出。
- 接收器 FIFO 阈值级别中断：FIFO 接收的数据量已达到指定的阈值级别。

PDMA 接口

PDMA 接口集成于 USART 模块。在发送或接收模式下可分别通过设置 USRCR 寄存器的 TXDMAEN 或 RXDMAEN 位为 1 来使能 USART PDMA 功能。当 USART 发送器 FIFO 中要发送的数据少于 USRFCR 寄存器中 TXTL 位域定义的 TX FIFO 阈值级别，则当 TXDMAEN 位置 1 时，PDMA 功能使能，将指定地址上的数据移到 USART TX FIFO 中。

同样的，若接收器 FIFO 中接收的数据量等于 USRFCR 寄存器 RXTL 位域定义的 RX FIFO 阈值级别，则当 RXDMAEN 位置 1 时，PDMA 功能使能，将数据从 USART RX FIFO 移到指定的目标地址中。

PDMA 配置详细描述可参考相应的 PDMA 章节。

寄存器列表

下表显示了 USART 寄存器及其复位值。

表 55. USART 寄存器列表

寄存器	偏移量	描述	复位值
USRDR	0x000	USART 数据寄存器	0x0000_0000
USRCR	0x004	USART 控制寄存器	0x0000_0000
USRFCR	0x008	USART FIFO 控制寄存器	0x0000_0000
USRIER	0x00C	USART 中断使能寄存器	0x0000_0000
USRSIFR	0x010	USART 状态 & 中断标志位寄存器	0x0000_0980
USRTPR	0x014	USART 时序参数寄存器	0x0000_0000
IrDACR	0x018	USART IrDA 控制寄存器	0x0000_0000
RS485CR	0x01C	USART RS485 控制寄存器	0x0000_0000
SYNCR	0x020	USART 同步控制寄存器	0x0000_0000
USRDLR	0x024	USART 分频器锁存寄存器	0x0000_0010
USRTSTR	0x028	USART 测试寄存器	0x0000_0000

寄存器描述

USART 数据寄存器 – USRDR

该寄存器用来存取 USART TX 与 RX FIFO 数据。

偏移量: 0x000
复位值: 0x0000_0000

类型 / 复位	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							DB
								RW 0
类型 / 复位	7	6	5	4	3	2	1	0
	DB							
	RW	0	RW	0	RW	0	RW	0

位	域	描述
[8:0]	DB	通过此接收缓冲寄存器读取数据，将返回接收器 FIFO 中的数据。接收器 FIFO 最大容量为 8 × 9 位。通过读取该寄存器，USART 将返回 7、8 或 9 位接收到的数据。DB[8] 位只有在 9-bit 模式下有效，在 8-bit 模式下此位固定为 0。在 7-bit 模式下，DB[6:0] 位为有效位。 写数据到此缓冲寄存器将会把数据加载到发送器 FIFO 中。发送器 FIFO 的最大容量为 8 × 9 位。通过写入数据到该寄存器，USART 将送出 7、8 或 9 位发送的数据。DB[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，DB[6:0] 位为有效位。

USART 控制寄存器 – USRCR

该寄存器用于定义串口参数，如 USART 的数据长度，奇偶和停止位，也包含 USART 使能控制位，USART 模式及数据传输模式选择位。

偏移量： 0x004
复位值： 0x0000_0000

类型 / 复位	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
	RTS	BCB	SPE	EPE	PBE	NSB	WLS	
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0
类型 / 复位	RXDMAEN	TXDMAEN	URRXEN	URTXEN	HFCEN	TRSM	MODE	
	RW	0	RW	0	RW	0	RW	0

位	域	描述
[15]	RTS	RTS Request-To-Send 信号 0: 驱动 USART RTS 引脚为逻辑 1 1: 驱动 USART RTS 引脚为逻辑 0 注意，当 HFCEN 位被复位，该位可用于控制 USART RTS 引脚状态。 当 HFCEN 位被置位时，RTS 位为只读位，用来指示 RTS 引脚在硬件流控功能控制下的引脚状态。
[14]	BCB	线中止控制位 当此位被设置为 1，USART 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 USART 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: 强制奇偶校验使能 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 位域设置为“10”来选择 9-bit 数据格式时，对 PBE 位的设置无效。
[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

位	域	描述
[9:8]	WLS	字长选择 00: 7 位 01: 8 位 10: 9 位 11: 保留
[7]	RXDMAEN	USART RX DMA 使能位 0: 除能 1: 使能
[6]	TXDMAEN	USART TX DMA 使能位 0: 除能 1: 使能
[5]	URRXEN	USART RX 使能 0: 除能 1: 使能
[4]	URTXEN	USART TX 使能 0: 除能 1: 使能
[3]	HFCEN	硬件流控制功能使能位 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
[1:0]	MODE	USART 模式选择 00: 正常模式 01: IrDA 模式 10: RS485 模式 11: 同步主机模式

USART FIFO 控制寄存器 – USRFCR

此寄存器定义了包括阈值级别和复位功能等 USART FIFO 控制与配置以及 RX 和 TX FIFO 容量状态位。

偏移量： 0x008
复位值： 0x0000_0000

	31	30	29	28	27	26	25	24
	保留位				RXFS			
类型 / 复位					RO	0	RO	0
	23	22	21	20	19	18	17	16
	保留位				TXFS			
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
	RXTL		TXTL		保留位		RXR	TXR
类型 / 复位	RW	0	RW	0	RW	0	WO	0

位	域	描述
[27:24]	RXFS	RX FIFO 状态 RXFS 位域显示了在 RX FIFO 中当前数据的总数。 0000: RX FIFO 为空 0001: RX FIFO 包含 1 个数据 ... 1000: RX FIFO 包含 8 个数据 其它: 保留
[19:16]	TXFS	TX FIFO 状态 TXFS 位域显示了在 TX FIFO 中当前数据的总数。 0000: TX FIFO 为空 0001: TX FIFO 包含 1 个数据 ... 1000: TX FIFO 包含 8 个数据 其它: 保留
[7:6]	RXTL	RX FIFO 阈值级别设定 00: 1 个数据 01: 2 个数据 10: 4 个数据 11: 6 个数据 RXTL 位域定义 RX FIFO 触发级别。
[5:4]	TXTL	TX FIFO 阈值级别设定 00: 0 个数据 01: 2 个数据 10: 4 个数据 11: 6 个数据 TXTL 位域定义 TX FIFO 触发级别。
[1]	RXR	RX FIFO 复位 此位置位将产生一个复位脉冲以复位 RX FIFO, RX FIFO 将被清空, 即在复位信号之后 RX 指针将复位至 0。在复位脉冲产生后此位由硬件自动清 0。

位	域	描述
[0]	TXR	TX FIFO 复位 此位置位将产生一个复位脉冲以复位 TX FIFO，TX FIFO 将被清空，即在复位信号之后 TX 指针将复位至 0。在复位脉冲产生后此位由硬件自动清 0。

USART 中断使能寄存器 – USRIER

此寄存器用于使能相关 USART 中断功能。当相应的事件发生且相应的中断使能位置位时，USART 模块产生中断。

偏移量： 0x00C
复位值： 0x0000_0000

类型 / 复位	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						CTSIE	RXTOIE
							RW 0	RW 0
类型 / 复位	7	6	5	4	3	2	1	0
	RSADDIE	BIE	FEIE	PEIE	OEIE	TXCIE	TXDEIE	RXDRIE
	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	域	描述
[9]	CTSIE	CTS – Clear-To-Send 中断使能 0: 除能 1: 使能 若此位置位，当 USRSIFR 寄存器中的 CTSC 位置位时将产生中断。
[8]	RXTOIE	接收器 FIFO 超时中断使能 0: 除能 1: 使能 接收器 FIFO 超时中断意味着在由 RXTOC 位域所定义的超时时间内接收器 FIFO 未空。 若此位置位，当 USRSIFR 寄存器中的 RXTOF 位置位时将产生中断。
[7]	RSADDIE	RS485 地址检测中断使能 0: 除能 1: 使能 若此位置位，当 USRSIFR 寄存器中的 RSADD 位置位时将产生中断。
[6]	BIE	线中止中断使能 0: 除能 1: 使能 若此位置位，当 USRSIFR 寄存器中的 BII 位置位时将产生中断。
[5]	FEIE	帧错误中断使能 0: 除能 1: 使能 若此位置位，当 USRSIFR 寄存器中的 FEI 位置位时将产生中断。

位	域	描述
[4]	PEIE	奇偶错误中断使能 0: 除能 1: 使能 若此位置位，当 USRSIFR 寄存器中的 PEI 位置位时将产生中断。
[3]	OEIE	溢出错误中断使能 0: 除能 1: 使能 若此位置位，当 USRSIFR 寄存器中的 OEI 位置位时将产生中断。
[2]	TXCIE	发送完成中断使能 0: 除能 1: 使能 若此位置位，当 USRSIFR 寄存器中的 TXC 位置位时将产生中断。
[1]	TXDEIE	发送数据空中断使能 0: 除能 1: 使能 若此位置位，当 USRSIFR 寄存器中的 TXDE 位置位时将产生中断。
[0]	RXDRIE	接收数据就绪中断使能 0: 除能 1: 使能 若此位置位，当 USRSIFR 寄存器中的 RXDR 位置位时将产生中断。

USART 状态 & 中断标志位寄存器 – USRSIFR

该寄存器包含了相应的 USART 状态。

偏移量: 0x010

复位值: 0x0000_0980

	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				CTSS	CTSC	RSADD	TXC	
类型 / 复位					RO	1 WC	0 WC	0 RO	1
	7	6	5	4	3	2	1	0	
	TXDE	RXTOF	RXDR	BII	FEI	PEI	OEI	RXDNE	
类型 / 复位	RO	1 WC	0 RO	0 WC	0 WC	0 WC	0 WC	0 RO	0

位	域	描述
[11]	CTSS	CTS – Clear-To-Send 状态 0: CTS 引脚无效 1: CTS 引脚有效且保持在逻辑低状态
[10]	CTSC	CTS 状态改变标志位 CTS 输入引脚状态一旦改变，此位将被置位。若 USRIER 寄存器中的 CTSIE = 1，则将产生中断。对此位写 1 将清除此标志位。

位	域	描述
[9]	RSADD	RS485 地址检测标志位 0: 未检测到地址帧 1: 已检测到地址帧 当接收器检测到地址帧时此位置位。若 USRIER 寄存器中的 RSADDIE = 1, 则将产生中断。对此位写 1 将清除此标志位。 注: 此位仅用于 RS485 模式, 模式选择通过 USRCR 寄存器中的 MODE[1:0] 设置。
[8]	TXC	发送完成标志位 0: 发送器 FIFO (TX FIFO) 或发送移位寄存器 (TSR) 为非空 1: TX FIFO 与 TSR 寄存器均为空 若 USRIER 寄存器中的 TXCIE = 1, 则将产生中断。向 USRDR 寄存器写入一个新数据将清除此标志位。
[7]	TXDE	发送器数据 FIFO 为空标志位 0: TX FIFO 级别高于阈值 1: TX FIFO 级别等于或低于阈值 当发送器 FIFO 级别等于或低于阈值级别时 TXDE 位将置位, 阈值由 USRFCR 寄存器中的 TXTL 位域设置。当新数据写入 USRDR 寄存器且 TX FIFO 级别高于阈值设定时此位清零。
[6]	RXTOF	接收器 FIFO 超时标志位 0: RX FIFO 超时未发生 1: RX FIFO 超时发生 当 RX FIFO 为非空且在 RXTOC 位域定义的超时周期内 RX FIFO 无操作发生, RXTOF 位将被置位。在读取 RX FIFO 之前需先将此位清零。对此位写 1 将清除此标志位。
[5]	RXDR	接收器 FIFO 就绪标志位 0: RX FIFO 级别低于阈值 1: RX FIFO 级别等于或高于阈值 当 FIFO 接收数据量达到指定的阈值级别时 RXDR 位将置位, 阈值由 USRFCR 寄存器中的 RXTL 位域设置。当从 USRDR 寄存器中读取数据且 RX FIFO 级别低于阈值时此位清零。
[4]	BII	线中止中断指示位 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。
[3]	FEI	帧错误指示位 接收字符没有有效停止位, 即检测到最后一个数据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。
[2]	PEI	奇偶错误指示位 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。
[1]	OEI	溢出错误指示位 在 RX FIFO 已满后, 若 RX 移位寄存器又完整接收了一个新的字符, 此时会发生溢出错误。溢出错误事件发生后, 若 RX 移位寄存器又接收了一个新的数据, RX 移位寄存器中原本的数据会被覆盖, 但 RX FIFO 中的数据不会受影响。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。
[0]	RXDNE	RX FIFO 数据非空 0: RX FIFO 为空 1: RX FIFO 中至少包含一个接收到的数据字

USART 时序参数寄存器 – USRTPR

该寄存器包含了 USART 的时序参数，包括发送器时间保护参数和接收器 FIFO 超时值以及 RX FIFO 超时功能使能控制。

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	TG							
	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0
类型 / 复位	RXTOEN	RXTOC						
	RW	0	RW	0	RW	0	RW	0

位	域	描述
[15:8]	TG	发送器时间保护 发送器时间保护计数器由波特率时钟驱动。当 TX FIFO 发送数据时，计数器被复位，完整的数据帧传送完后开始计数。只有当计数器的内容等于 TG 值时，才允许执行下一个字发送。
[7]	RXTOEN	TX FIFO 超时计数器使能 0: TX FIFO 超时计数器除能 1: TX FIFO 超时计数器使能
[6:0]	RXTOC	TX FIFO 超时计数器比较值 RX FIFO 超时计数器由波特率时钟驱动。当 RX FIFO 接收到新数据时，计数器被复位，然后开始计数。一旦计数器的内容等于超时计数器比较值 RXTOC，且 USRIER 寄存器的 RXTOIE 位为 1 时，将产生 TX FIFO 超时中断 RXTOI。接收到新数据或空的 RX FIFO 被读取后，将清除 RX FIFO 超时计数器。

USART IrDA 控制寄存器 – IrDACR

该寄存器用于控制 USART 的 IrDA 模式。

偏移量: 0x018

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	
	IrDAPSC								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
	保留位		RXINV	TXINV	LB	TXSEL	IrDALP	IrDAEN	
类型 / 复位			RW	0	RW	0	RW	0	RW

位	域	描述
[15:8]	IrDAPSC	IrDA 预分频值 此位域包含 8-bit 去抖预分频值。 去抖向下计数器由 USART 时钟 CK_USART 驱动。计数周期由 IrDAPSC 位域定义。 IrDAPSC 位域必须设置为等于或大于 0x01 的值，使去抖计数器正常工作。如果脉冲宽度小于 IrDAPSC 位域定义的时间段，则脉冲将被视为干扰噪声并丢弃。 00000000: 保留 – 不可用 00000001: CK_USART 时钟 / 1 00000010: CK_USART 时钟 / 2 00000011: CK_USART 时钟 / 3 ...
[5]	RXINV	RX 信号反相控制 0: 不反相 1: RX 输入信号反相
[4]	TXINV	TX 信号反相控制 0: 不反相 1: TX 输出信号反相
[3]	LB	IrDA 回送模式 0: 除能 IrDA 回送模式 1: 使能 IrDA 回送模式用于自检
[2]	TXSEL	收发器选择 0: 使能 IrDA 接收器 1: 使能 IrDA 发送器
[1]	IrDALP	IrDA 低功耗模式 选择 IrDA 工作模式。 0: 正常模式 1: IrDA 低功耗模式
[0]	IrDAEN	IrDA 使能控制 0: 除能 IrDA 模式 1: 使能 IrDA 模式

USART RS485 控制寄存器 – RS485CR

该寄存器用于控制 USART 的 RS485 模式。

偏移量：0x01C

复位值：0x0000_0000

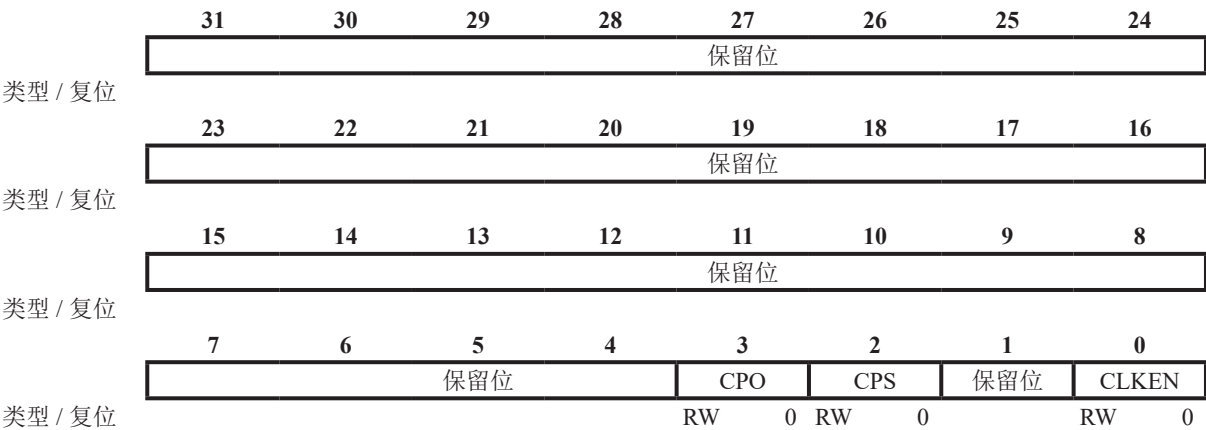
	31	30	29	28	27	26	25	24	
	保留位								
类型 / 复位									
	23	22	21	20	19	18	17	16	
	保留位								
类型 / 复位									
	15	14	13	12	11	10	9	8	
	ADDMATCH								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	7	6	5	4	3	2	1	0	
	保留位					RSAAD	RSNMM	TXENP	
类型 / 复位						RW	0	RW	0

位	域	描述
[15:8]	ADDMATCH	RS485 自动地址匹配值 此位域是用于存储 RS485 自动地址检测工作模式的地址匹配值。
[2]	RSAAD	RS485 自动地址检测工作模式控制 0: 除能 1: 使能
[1]	RSNMM	RS485 正常多点通信工作模式控制 0: 除能 1: 使能
[0]	TXENP	USART RTS / TXE 引脚极性 0: 在 RS485 发送模式下 RTS / TXE 高电平有效 1: 在 RS485 发送模式下 RTS / TXE 低电平有效

USART 同步控制寄存器 – SYNCR

该寄存器用于控制 USART 的同步模式。

偏移量： 0x020
复位值： 0x0000_0000



位	域	描述
[3]	CPO	时钟极性 0: CTS / SCK 引脚空闲状态为低 1: CTS / SCK 引脚空闲状态为高 在同步模式下选择 USART CTS / SCK 引脚上的时钟输出极性。与 CPS 位共同定义所需的时钟空闲状态。
[2]	CPS	时钟相位 0: 数据在第一个时钟边沿被捕捉 1: 数据在第二个时钟边沿被捕捉 在同步模式下该位允许用户选择 USART CTS / SCK 引脚上的时钟输出相位。与 CPO 位共同定义数据捕捉边沿。
[0]	CLKEN	时钟使能 0: CTS / SCK 引脚除能 1: CTS / SCK 引脚使能 使能 / 除能 USART CTS / SCK 引脚。

USART 分频器锁存寄存器 – USRDLR

该寄存器用来定义 USART 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								
	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								
	BRD															
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								
	BRD															
类型 / 复位	RW	0	RW	0	RW	0	RW	1	RW	0	RW	0	RW	0	RW	0

位	域	描述
[15:0]	BRD	波特率分频器 这 16 位定义了 USART 时钟分频率。 波特率 = CK_USART / BRD CK_USART 时钟是连接到 USART 模块的时钟。 在异步模式中 BRD = 16 ~ 65535；在同步模式中 BRD = 8 ~ 65535。

USART 测试寄存器 – USRTSTR

该寄存器控制 USART 调试模式。

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	保留位						LBM	
							RW	0 RW 0

位	域	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

22 通用异步收发器 (UART)

简介

通用异步收发器 UART, 提供了一种灵活的异步传输的全双工数据交换方式。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。

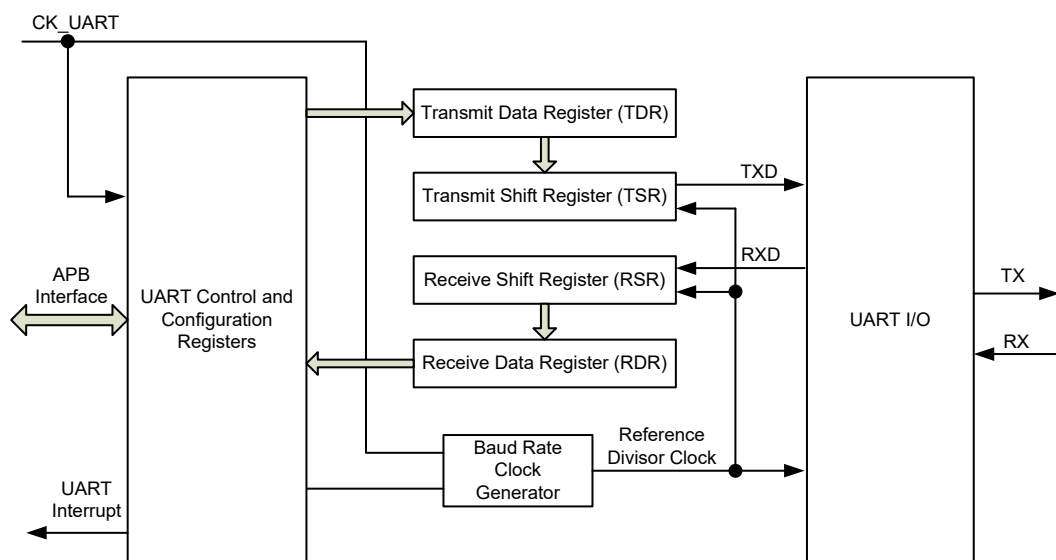


图 159. UART 方框图

特性

- 支持异步串行通信模式
- 全双工通信能力
- 可编程波特率高达 (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
- 支持 PDMA 接口

功能描述

串行数据格式

UART 模块对写入在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，可选的奇偶校验位和最后 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，并在下一个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，然后寻找停止位。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。

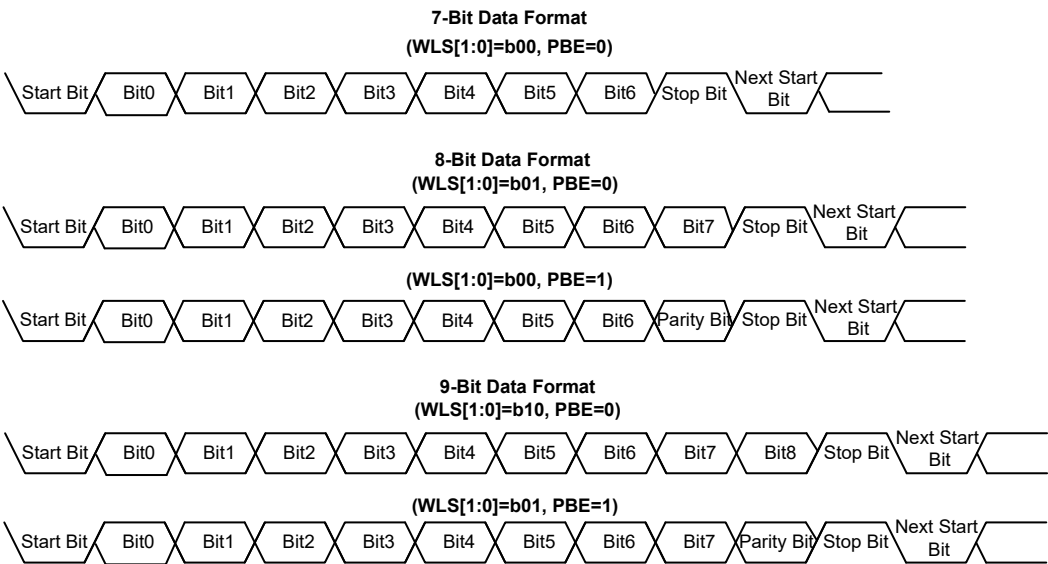


图 160. UART 串行数据格式

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。

波特率时钟 = CK_UART / BRD

CK_UART 时钟是连接到 UART 模块的 APB 时钟, BRD 的范围是 16 ~ 65535。

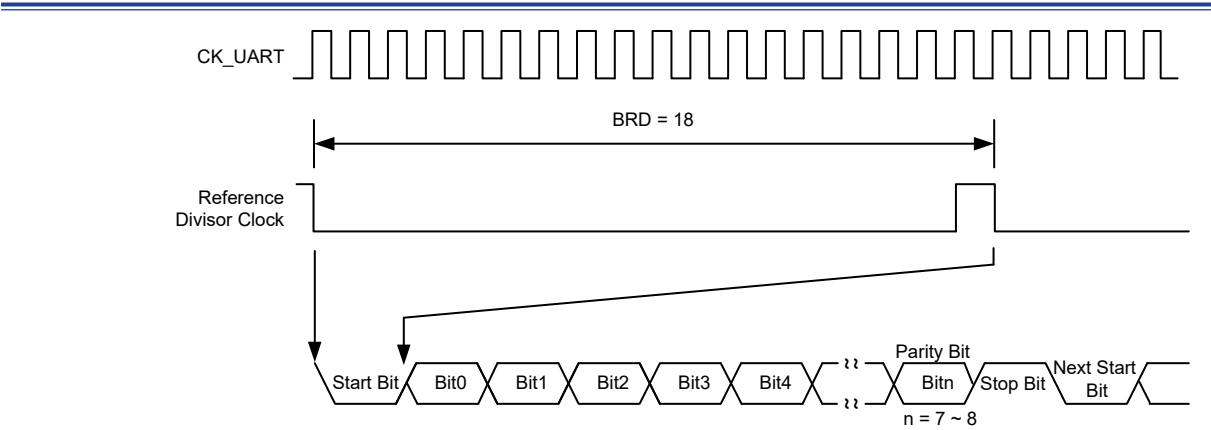


图 161. UART 时钟 CK_UART 和数据帧时序

表 56. 波特率误差计算 – CK_UART = 40 MHz

波特率		CK_UART = 40 MHz		
No.	Kbps	实际值	BRD	误差率
1	2.4	2.4	16667	0.00%
2	9.6	9.6	4167	-0.01%
3	19.2	19.2	2083	0.02%
4	57.6	57.6	694	0.06%
5	115.2	115.3	347	0.06%
6	230.4	229.9	174	-0.22%
7	460.8	459.8	87	-0.22%
8	921.6	930.2	43	0.94%
9	2250	2222.2	18	-1.23%
10	3000	—	—	—

表 57. 波特率误差计算 – CK_UART = 48 MHz

波特率		CK_UART = 48 MHz		
No.	Kbps	实际值	BRD	误差率
1	2.4	2.4	20000	0.00%
2	9.6	9.6	5000	0.00%
3	19.2	19.2	2500	0.00%
4	57.6	57.6	833	0.04%
5	115.2	115.1	417	-0.08%
6	230.4	230.8	208	0.16%
7	460.8	461.5	104	0.16%
8	921.6	923.1	52	0.16%
9	2250	2285.7	21	1.59%
10	3000	3000	16	0.00%

表 58. 波特率误差计算 – CK_UART = 60 MHz

波特率		CK_UART = 60 MHz		
No.	Kbps	实际值	BRD	误差率
1	2.4	2.4	25000	0.00%
2	9.6	9.6	6250	0.00%
3	19.2	19.2	3125	0.00%
4	57.6	57.6	1042	-0.03%
5	115.2	115.1	521	-0.03%
6	230.4	230.8	260	0.16%
7	460.8	461.5	130	0.16%
8	921.6	923.1	65	0.16%
9	2250	2285.7	27	-1.23%
10	3000	3000	20	0.00%

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 寄存器等待被读取。

PDMA 接口

PDMA 接口集成于 UART 模块。在发送和接收模式下可分别通过设置 URCR 寄存器的 TXDMAEN 或 RXDMAEN 位为 1 来使能 PDMA 功能。当 UART 发送数据寄存器 TDR 为空，则当 TXDMAEN 位置 1 时，PDMA 功能使能，将指定地址上的数据移到 UART 发送数据寄存器 TDR 中。

同样的，若接收数据寄存器 RDR 中已接收有数据，则当 RXDMAEN 位置 1 时，PDMA 功能使能，可将数据从接收数据寄存器 RDR 移到指定的目标地址中。

PDMA 配置详细描述可参考相应的 PDMA 章节。

寄存器列表

下表显示了 UART 寄存器及其复位值。

表 59. UART 寄存器列表

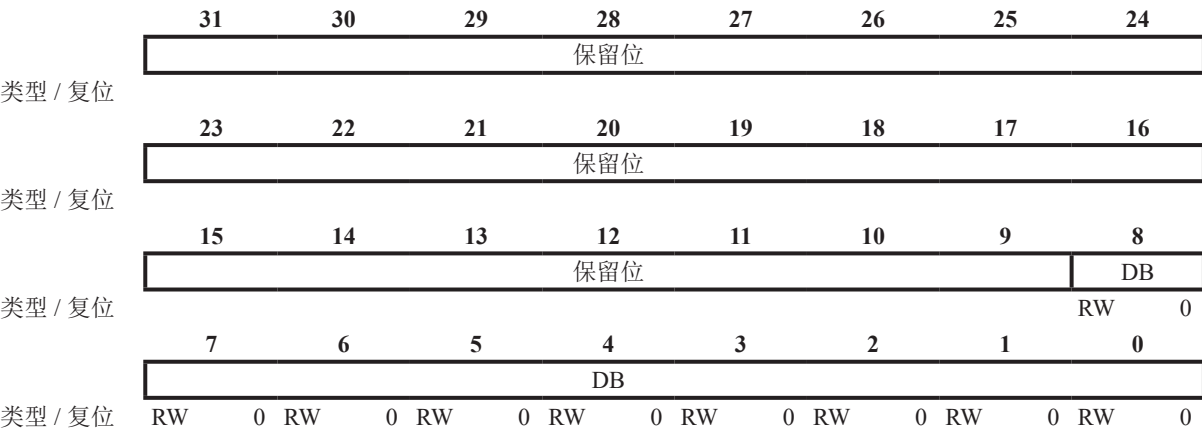
寄存器	偏移量	描述	复位值
URDR	0x000	UART 数据寄存器	0x0000_0000
URCR	0x004	UART 控制寄存器	0x0000_0000
URIER	0x00C	UART 中断使能寄存器	0x0000_0000
URSIFR	0x010	UART 状态 & 中断标志位寄存器	0x0000_0180
URDLR	0x024	UART 分频器锁存寄存器	0x0000_0010
URTSTR	0x028	UART 测试寄存器	0x0000_0000

寄存器描述

UART 数据寄存器 – URDR

该寄存器用来存取 UART 发送或接收到的数据。

偏移量: 0x000
复位值: 0x0000_0000



位	域	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[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，寄存器的 DB[6:0] 为可用位。通过写入该寄存器，UART 将送出 7、8 或 9 位发送的数据。DB[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，寄存器 DB[6:0] 为可用位。

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

类型 / 复位	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	BCB	SPE	EPE	PBE	NSB	WLS	
类型 / 复位	7	6	5	4	3	2	1	0
	RXDMAEN	TXDMAEN	URRXEN	URTXEN	保留位	TRSM	保留位	
类型 / 复位	RW	0	RW	0	RW	0	RW	0

位	域	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：强制奇偶校验使能 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 位域设置为“10”来选择 9-bit 数据格式时，对 PBE 位的设置无效。
[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：保留位
[7]	RXDMAEN	UART RX DMA 使能位 0：除能 1：使能

位	域	描述
[6]	TXDMAEN	UART TX DMA 使能位 0: 除能 1: 使能
[5]	URRXEN	UART RX 使能 0: 除能 1: 使能
[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	
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	BIE	FEIE	PEIE	OEIE	TXCIE	TXDEIE	RXDRIE	
	RW	0	RW	0	RW	0	RW	0	RW

位	域	描述
[6]	BIE	线中止中断使能 0: 除能 1: 使能 若此位置位, 当 URSIFR 寄存器中的 BII 位置位时将产生中断。
[5]	FEIE	帧错误中断使能 0: 除能 1: 使能 若此位置位, 当 URSIFR 寄存器中的 FEI 位置位时将产生中断。
[4]	PEIE	奇偶错误中断使能 0: 除能 1: 使能 若此位置位, 当 URSIFR 寄存器中的 PEI 位置位时将产生中断。

位	域	描述
[3]	OEIE	溢出错误中断使能 0: 除能 1: 使能 若此位置位，当 URSIFR 寄存器中的 OEI 位置位时将产生中断。
[2]	TXCIE	发送完成中断使能 0: 除能 1: 使能 若此位置位，当 URSIFR 寄存器中的 TXC 位置位时将产生中断。
[1]	TXDEIE	发送数据寄存器空中断使能 0: 除能 1: 使能 若此位置位，当 URSIFR 寄存器中的 TXDE 位置位时将产生中断。
[0]	RXDRIE	接收数据就绪中断使能 0: 除能 1: 使能 若此位置位，当 URSIFR 寄存器中的 RXDR 位置位时将产生中断。

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

	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							TXC	
									RO 1
	7	6	5	4	3	2	1	0	
类型 / 复位	TXDE	保留位	RXDR	BII	FEI	PEI	OEI	保留位	
	RO 1		RO 0	WC 0	WC 0	WC 0	WC 0	0	

位	域	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 为非空 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	域	描述
[5]	RXDR	RX 数据就绪标志位 0: 接收数据寄存器为空 1: 接收数据寄存器中有待读取数据 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。
[4]	BII	线中止中断指示位 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。
[3]	FEI	帧错误指示位 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。
[2]	PEI	奇偶错误指示位 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。
[1]	OEI	溢出错误指示位 在接收数据寄存器已满后, 若接收移位寄存器完整接收了一个新的字符, 此时会发生溢出错误。溢出错误事件发生后, 若接收移位寄存器又接收了一个新的数据, 接收移位寄存器中原本的数据不会转移到接收数据寄存器中, 而是会被覆盖掉。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。

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
	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
	BRD							
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0
	BRD							
类型 / 复位	RW	0	RW	0	RW	1	RW	0

位	域	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 APB 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

	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						LBM	
							RW	0 RW 0

位	域	描述
[1:0]	LBM	回送测试模式选择 00: 正常工作模式 01: 保留位 10: 自动回应模式 11: 回送模式

23 智能卡接口 (SCI)

简介

该智能卡接口 (SCI) 符合 ISO 7816-3 标准。此接口包括卡插入 / 移除检测、SCI 数据传输控制逻辑和数据缓冲器、内部定时 / 计数器和控制逻辑电路来完成所需的智能卡操作。智能卡接口充当一个智能卡读卡器, 以便与外部智能卡通信。智能卡接口的所有功能是由一系列控制寄存器和状态寄存器以及一些相关中断控制的。中断可使单片机注意到 SCI 的传输状态。

由于 ISO 7816-3 标准数据协议的复杂性, 无法在此提供全面的规范, 因此, 读者应该查阅其它外部信息以详细了解本标准。

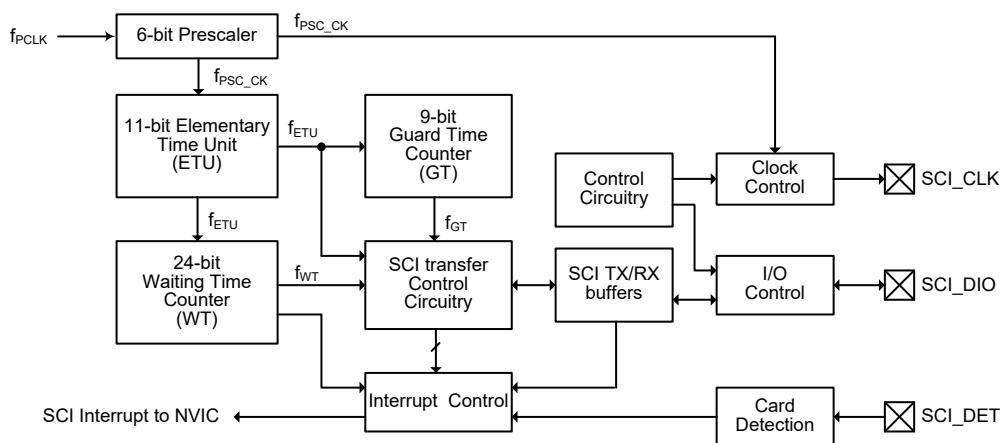


图 162. SCI 方框图

特性

- 符合 ISO 7816-3 标准
- 字符传输模式
- 一个发送缓冲器和一个接收缓冲器
- 11-bit 基本时间单元 (ETU) 计数器
- 9-bit 保护时间计数器
- 24-bit 通用等待时间计数器
- 奇偶检验位产生和检测
- 发送和接收模式下检测到奇偶检验错误时自动重发字符
- 发送或接收完成时支持 PDMA 访问

功能描述

内部智能卡接口通过 SCI_CLK、SCI_DIO 和 SCI_DET 这几个外部引脚，实现与外部智能卡通信。SCI_CLK 是时钟输出信号引脚，与串行数据引脚 SCI_DIO 一起用来与外部智能卡通信。SCI_CLK 和 SCI_DIO 的工作模式可以选择为 SCI 数据传输模式，由 SCI 控制电路自动驱动，或手动模式，由应用程序分别设置内部 CLK 和 DIO 寄存器位来控制。SCI_DET 是外部智能卡检测的输入引脚，当外部智能卡插入或移除被检测到时，如果相应的中断控制位使能，会产生一个中断信号发送到单片机。

为实现正确的数据传输，在智能卡接口开始与外部智能卡进行通信之前，必须先执行一些与时序相关的程序。智能卡接口数据传输操作中时序相关功能会使用到三个计数器，分别为基本时间单元计数器 (ETU)、保护时间计数器 (GT) 和等待时间计数器 (WT)。

基本时间单元计数器

基本时间单元 (ETU) 是一个 11-bit 向上计数型计数器，产生的时钟 f_{ETU} 可以作为 SCI 数据发送和接收的工作频率来源。ETU 的时钟源来自智能卡时钟 f_{PSC_CK} ，此时钟源自一个 6-bit 分频器产生。SCI 的数据传输是一个基于字符帧的协议，基本上包括一个起始位、8 位数据和一个奇偶校验位。ETU 产生的时间周期 t_{ETU} ($1/f_{ETU}$)，是一个字符位的时间单元。有一个与基本时间单元相关的寄存器 ETUR，用来存储 ETU 的预期内容。每次 ETUR 寄存器被写入，ETU 电路都将重新加载新写入的值，并重新开始计数。基本时间单元 t_{ETU} 由如下公式取得。波特率的定义遵循 ISO 7816-3 标准。

$$1etu = t_{ETU} = \frac{F_i}{D_i} \times \frac{1}{f}$$

其中：

- etu 是由接口提供给智能卡的 SCI_DIO 数据位的额定时间
- Di 是比特率调整因数
- Fi 是时钟频率转换因数
- f 是由接口提供给智能卡的时钟信号 SCI_CLK 的频率值

Di 是基于 4-bit DI 位域编码的十进制值，如下表所示。

表 60. 基于 DI 位域编码的 Di 十进制值

DI 位域	0001	0010	0011	0100	0101	0110	1000	1001
Di (十进制)	1	2	4	8	16	32	12	20

Fi 是基于 4-bit FI 位域编码的十进制值，如下表所示。

表 61. 基于 FI 位域编码的 Fi 十进制值

FI 位域	0000	0001	0010	0011	0100	0101	0110	1001	1010	1011	1100	1101
Fi (十进制)	372	372	558	744	1116	1488	1860	512	768	1024	1536	2048

上表中的 FI 和 DI 的值，在首次插入外部智能卡时，将从外部智能卡发送到智能卡接口的复位应答包中得到。当智能卡接口收到 FI 和 DI 信息，Fi 和 Di 的值可以通过查询上面两个表得到。在获得 Fi 和 Di 值时，写入到 ETUR 寄存器的值可以由 Fi/Di 计算。下表显示了由 Fi/Di 比例计算得到的 ETU 可能值。

表 62. 由 Fi/Di 比例计算得到的 ETU 可能值

$\frac{F_i}{D_i}$	372	558	774	1116	1488	1860	512	768	1024	1536	2048
1	372	558	744	1116	1488	1860	512	768	1024	1536	2048
2	186	279	372	558	744	930	256	384	512	768	1024
4	93	139.5	186	279	372	465	128	192	256	384	512
8	46.5	69.75	93	139.5	186	232.5	64	96	128	192	256
16	23.25	34.87	46.5	69.75	93	116.2	32	48	64	96	128
32	11.62	17.43	23.25	34.87	46.5	58.13	16	24	32	48	64
12	31	46.5	62	93	124	155	42.66	64	85.33	128	170.6
20	18.6	27.9	37.2	55.8	74.4	93	25.6	38.4	51.2	76.8	102.4

补偿模式

由于 ETUR 寄存器的值是由上面的公式得到的, 计算结果可能不是一个整数。如果计算的结果不是一个整数, 小于整数 n 但大于整数 (n-1), 应写入 ETUR 寄存器的是整数 n 还是 (n-1) 取决于结果是接近 n 还是 (n-1)。这里所说的整数 n 是十进制数。

如果计算的结果接近 (n-0.5), 为了成功的数据传输, 应设置 ETUR 寄存器中的补偿使能控制位 COMP 为 1 以使能补偿模式。当结果接近 (n-0.5), 补偿模式使能时, 写入 ETUR 寄存器的值应是 n。然后 ETU 时间单元序列会由 n 个时钟周期和接着 (n-1) 个时钟周期这样交替产生。因此产生了 (n-0.5) 个时钟周期的平均时间单元, 并允许 ETU 时间间隔减小为半个时钟周期。注意, 当在 SCI 数据传输模式下出现起始位时, ETU 将重新载入 ETUR 寄存器的值并重新开始计数。

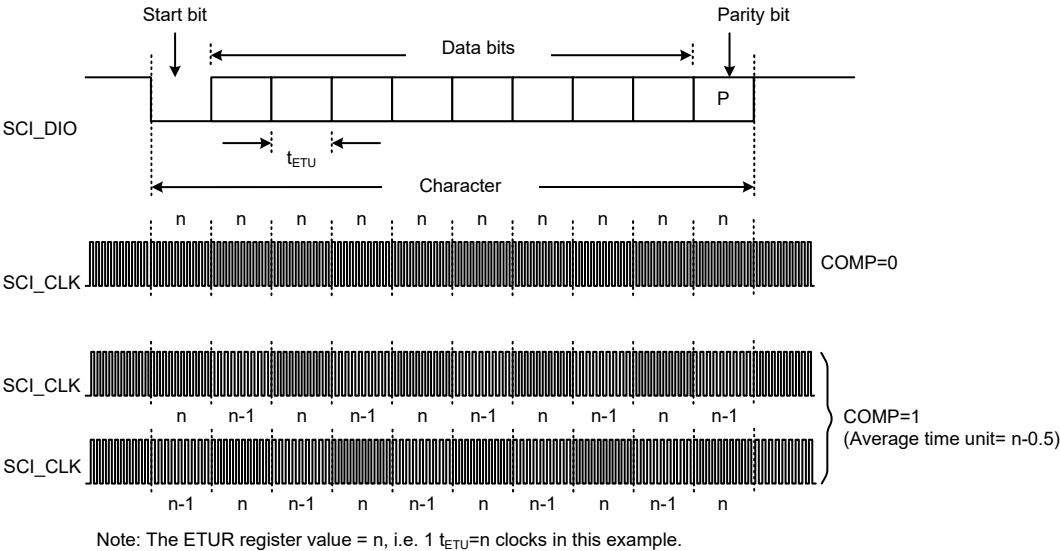


图 163. 字符帧和补偿模式

保护时间计数器

保护时间计数器 (GT) 是一个 9-bit 向上计数型计数器, 在 SCI 数据传输的两个连续字符的前沿之间产生一个很短的时间长度, 即字符帧 t_{GT} 。保护时间计数器的时钟源来自于 ETU, 即方框图中的 f_{ETU} 。有一个与保护时间计数器相关的寄存器 GTR, 用来存储保护时间计数器的预期值。保护时间值将在当前保护时间周期结束时重新加载。注意, 智能卡接口电路从接收最后一个字符到发送下一个字符之间的保护时间应该通过应用程序合理设置。因为第一个字符被发送时不插入保护时间。

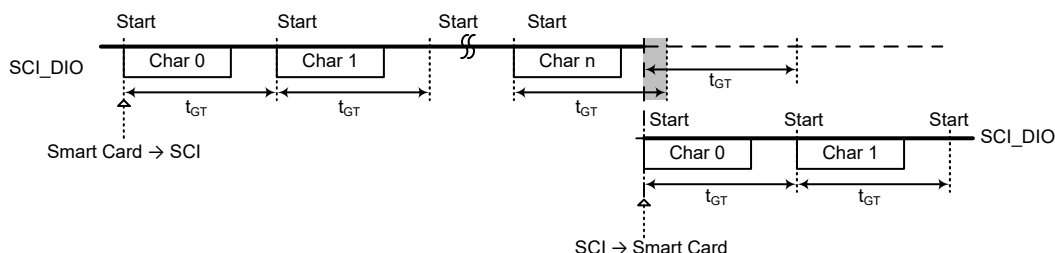


图 164. 保护时间

等待时间计数器

等待时间计数器 WT 是一个 24-bit 向下计数型计数器, 生成数据传输的最大持续时间为 t_{WT} , 等待时间计数器的时钟源来自于 ETU, 即 f_{ETU} 。

有一个用于等待时间计数器的寄存器 WTR, 用来存储等待时间计数器的预期值。等待时间计数器可用于 SCI 数据传输模式和手动模式, 可以在特定条件重新加载。等待时间计数器的功能由 CR 寄存器中的 WTEN 位控制。当 SCI 设置在 SCI 数据传输模式且 WTEN 位置 1 使等待时间计数器使能时, 如果检测到起始位, 则更新的 WTR 寄存器值将被加载到等待时间计数器。注意, 在 SCI 数据传输模式, 在外部智能卡插入之后才可置位 WTEN 使能等待时间计数器功能。

如果 SCI 被配置为手动模式, 等待时间计数器可作为普通定时器使用, 定时器可通过置位或清零 WTEN 位来控制使能或除能。如果等待时间计数器被使能, 则更新的 WTR 寄存器值将不会被载入等待时间计数器中。当设置 WTEN 位为 0, 等待时间计数器功能除能, 此时对 WTR 寄存器写入新的值, 该值会立即载入到等待时间计数器, 之后当 WTEN 位再次置 1 时, 计数器从此新的值开始计数。

在传输过程中, 软件可以改变等待时间的值。例如, 在 $T = 1$ 模式, 块等待时间 t_{BWT} 应该在最后的传输字符的起始位出现前加载到 WTR 寄存器。当最后一个字符传输完成时, 软件应把字符等待时间值 t_{CWT} , 写入到 WTR 寄存器中。

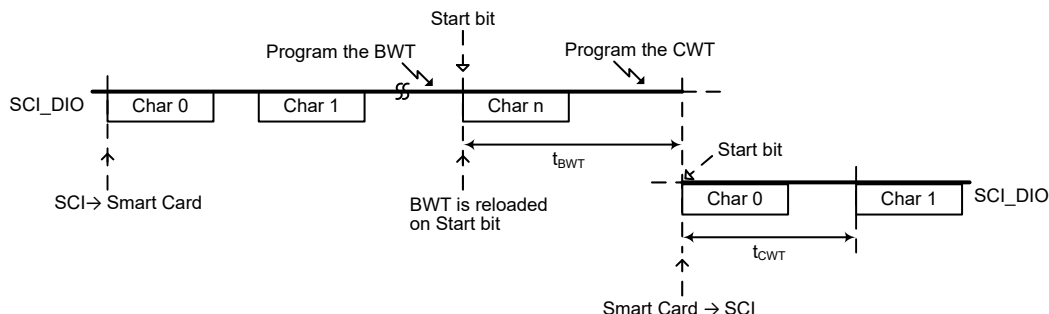


图 165. 字符和块等待时间 – CWT 和 BWT

智能卡时钟和数据选择

SCI 通过一系列的外部引脚与外部智能卡通信。这些引脚是串行数据引脚 SCI_DIO、外部时钟引脚 SCI_CLK 和智能卡检测输入引脚 SCI_DET。

SCI 串行数据引脚 SCI_DIO, 可选择由 SCI 硬件电路或软件通过相关的控制位来控制, 这取决于 SCI 是工作在 SCI 传输模式还是工作在手动模式。工作模式通过 CR 寄存器中的 SCIM 位进行选择。当清零 CR 寄存器的 SCIM 位时, SCI 工作在手动模式, SCI_DIO 引脚状态由 CCR 寄存器中的 CDIO 位控制。但当 SCI 工作在 SCI 传输模式时, SCI_DIO 引脚状态由 SCI 传输电路控制。

SCI 时钟输出引脚 SCI_CLK 可选择由 6-bit SCI 预分频器或软件位进行控制, 取决于 CCR 寄存器中的 CLKSEL 位的情况。当 CLKSEL 位清零时, SCI_CLK 引脚状态由 CCR 寄存器中的 CCLK 位控制。但当 CLKSEL 位置 1 时, SCI_CLK 信号来自于 6-bit 预分频器的输出信号。预分频器的分频比由 PSCR 寄存器中的 PSC 位域控制。

智能卡检测

当有外部智能卡插入时, 内部卡检测器可以检测到这个插入动作, 如果 IER 寄存器中对应的中断控制位 CARDIRE 置 1, 将产生卡插入中断。类似地, 如果卡被移除, 内部卡检测器同样会检测到移除动作, 如果 IER 寄存器中的中断控制位 CARDIRE 置 1, 相关中断使能, 将产生卡移除中断。

卡检测器可支持两种检测开关机制。一种是没有卡时开关常 OFF 机制, 另一种是没有卡时开关常 ON 机制。在注意到哪个卡检测机制类型被选择后, 应设置 CR 寄存器中的 DETCNF 位来选择开关机制使其正确检测卡是否存在。当卡实际存在于 SCI_DET 引脚时, 无论由 DETCNF 位设置为哪个卡开关类型, SR 寄存器中卡插入 / 移除标志位 CPREF 都将被置 1。注意, 卡检测器没有硬件去抖电路。SCI_DET 引脚电平的任何变化都将引起 CPREF 位发生变化。所需的去抖时间由应用程序进行处理。

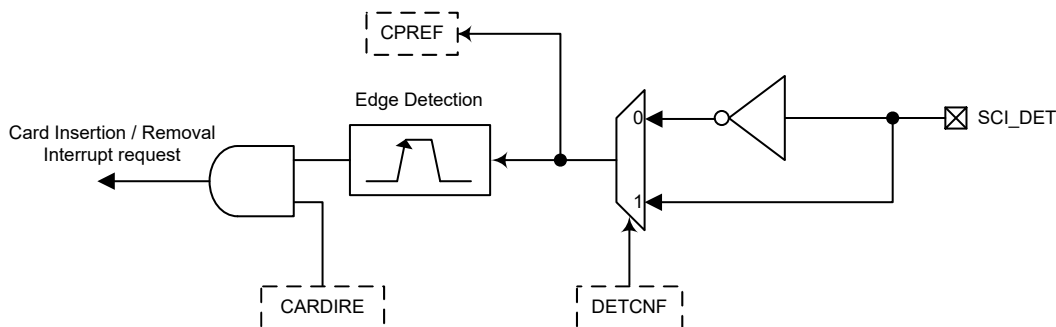


图 166. SCI 卡检测方框图

SCI 数据传输模式

与外部智能卡进行 SCI 数据传输有两个工作模式。一个是 SCI 模式而另一个是手动模式。数据传输模式通过 CR 寄存器中的 SCI 模式选择位 SCIM 选择。当 SCIM 位设置为 1 时，SCI 模式使能，数据将自动由 SCI 传输电路进行传输。否则，如果 SCIM 位设置为 0，数据传输在手动模式中运行。SCI 传输接口是半双工接口，通过 SCI_CLK 和 SCI_DIO 引脚与外部智能卡进行通信。复位后，SCI 传输接口处于接收模式，SCI 传输被除能。当选择了 SCI 模式，数据传输通过 SCI_CLK 和 SCI_DIO 引脚由 SCI 传输电路自动驱动。

有两个与数据发送和接收相关的数据寄存器，TXB 和 RXB，分别存储发送和接收的数据。如果在 SCI 传输模式下，一个字符被写入到 TXB 寄存器，复位后 SCI 传输接口将自动从接收模式切换到发送模式。当 SCI 发送或接收完成后，SR 寄存器中相应的请求标志 TXCF 或 RXCF 将被设置为 1。如果发送缓冲器是空的，SR 寄存器中的发送缓冲器空标志 TXBEF 将被设置为 1。

奇偶校验功能

SCI 传输接口支持奇偶发生器和奇偶校验功能。当在数据传输过程中发生奇偶校验错误，SR 寄存器相应的请求标志 PARF 将被设置为 1。一旦 PARF 位被设置为 1，如果 IER 寄存器中的相关中断控制位 PARE 使能，则 IPR 寄存器中的奇偶校验错误挂起标志 PARP 也将被设置为 1。

如果 SCI 发送的数据被智能卡接收器所接收，且在发送过程中没有奇偶校验错误，SCI 接口发送请求标志 TXCF 将被设置为 1，SCI 奇偶错误请求标志位 PARF 位将保持为 0。如果由外部智能卡发送的数据被 SCI 接口所接收，且在接收过程中无奇偶校验错误，SCI 接口接收请求标志 RXCF 将被设置为 1，PARF 位将保持为零。

重发功能

当奇偶校验错误发生时，SCI 传输电路支持字符重发功能。字符重发功能由 CR 寄存器中的 CREP 位置 1 使能。在数据传输过程中，如果发生奇偶错误，则重发功能将使能。重发次数可由 CR 寄存器中的 RETRY 位选择为 4 或 5。

当 CREP 位置 1，字符重发功能将使能。以 4 次重发为例，当 CREP 位置 1 且 RETRY 位也置 1 时，在发送模式下，如果产生奇偶校验错误信号，则 SCI 将重复发送数据，最多重发 4 次。如果 SCI 接口被告知在 4 次发送都存在奇偶校验错误，SCI 接口奇偶校验错误标志 PARF 将在相同的数据发送 4 次后被设置为 1，但 TXCF 将不会被置位。此时在发送缓冲器中的数据将被载入到发送移位寄存器，发送缓冲器为空，将使 TXBEF 标志位设为 1。

同样地, 如果 SCI 接口处于接收模式, 当字符重发功能使能, 它会告知外部智能卡有一个奇偶校验错误, 最多 4 次。如果 SCI 告知外部智能卡在 4 次发送过程中一直存在奇偶校验错误, 奇偶错误标志位 PARF 和接收请求标志位 RXCF 都将被置 1。

如果 CREP 位清零, 字符重发功能将除能。当 SCI 工作在接收模式时, 若接收到的数据奇偶校验错误, PARF 和 RXCF 位都将置 1。如果在发送模式下 SCI 被告知有奇偶错误, PARF 位将被置 1, 但 TXCF 位不会被置位。

手动数据传输模式

当设置 SCIM 位为 0, 选择在手动模式下传输数据。在手动模式下, 数据是由 CCR 寄存器中的控制位 CDIO 控制。在手动模式下, CDIO 位的值将立即反映在 SCI_DIO 引脚上。注意, 在手动模式下的字符重发功能以及相关的标志位都不能使用, 所有的数据传输是由应用程序处理。SCI_CLK 引脚输出的用于驱动外部智能卡的时钟, 通过 CCR 寄存器的 CLKSEL 位选择, 由内部时钟源, 即 6-bit 预分频器的输出 f_{PSC_CK} , 或 CCR 寄存器中的控制位 CCLK 提供。当 CLKSEL 被设置 1, 选择 6-bit 预分频器的输出 f_{PSC_CK} 作为智能卡的时钟源。如果用户想手动来管理时钟, 应先设置 CLKSEL 位为 0, 然后 CCLK 位的值将出现在 SCI_CLK 引脚上。

数据传输方向协议

如果智能卡所使用的方向协议与智能卡接口的协议相同, 当接收中断使能, SCI 将产生接收中断但不会置位奇偶校验错误标志位。否则, SCI 产生接收中断且奇偶校验错误标志将被置 1。通过检查奇偶校验错误标志, SCI 可以知道数据方向协议正确与否。

中断发生器

SCI 产生 SCI 中断有几个条件。当这些条件存在, 将会产生一个中断脉冲引起单片机的注意。这些条件包括智能卡插入 / 移除、等待时间计数器下溢、奇偶校验错误、字符发送或接收结束和发送缓冲器为空。当智能卡中断由这些条件的任何一个产生, 如果 NVIC 中 SCI 总中断和 SCI 相应的中断控制同时使能, 程序将在返回到主程序之前跳转到相应的中断向量。

对于 SCI 的中断事件, 相应的挂起标志可以由相应的中断使能控制位屏蔽。当相关的中断使能控制除能, 相应的中断挂起标志不受请求标志影响, 不会产生中断。如果相关的中断使能控制位使能, 相应的中断挂起标志将受中断请求标志影响, 并产生中断。挂起标志寄存器 IPR 是只读的, 一旦挂起标志由应用程序读取, 它会自动清零, 而相关的请求标志由应用程序手动清零。

为了响应 SCI 中断, 除了 SCI 相应的中断使能控制位被置位, NVIC 中 SCI 总中断使能控制位也必须被置位。如果 SCI 总中断控制位没有被置位, 将不会响应中断。

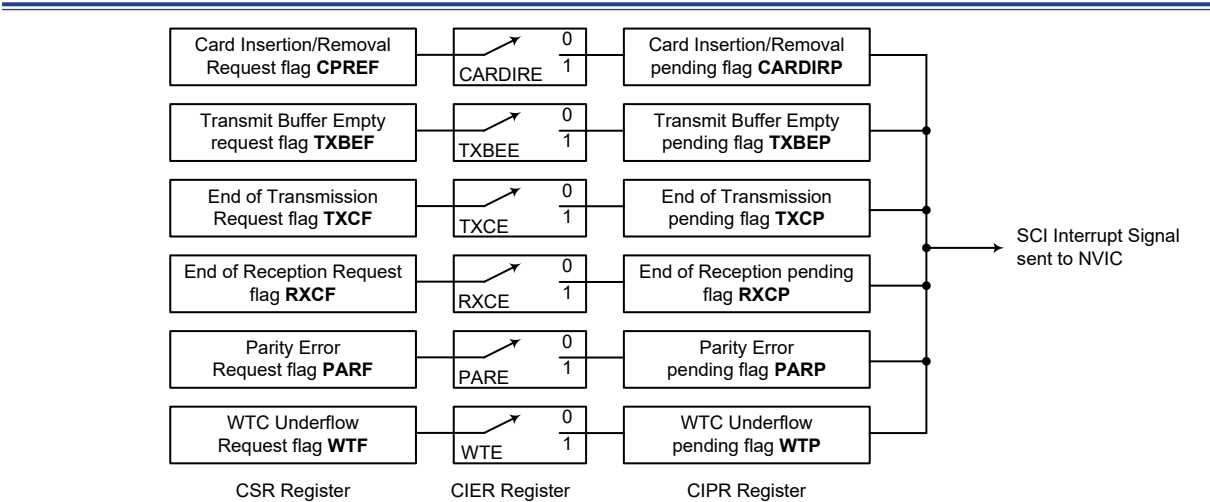


图 167. SCI 中断结构

PDMA 接口

PDMA 接口集成在 SCI 模块中。在发送或接收模式下通过对应设置 CR 寄存器中的 TXDMA 或 RXDMA 位为 1 可使能 PDMA 功能。当发送缓冲器为空从而置位发送缓冲器空标志位 TXBEF, 且 TXDMA 位被置 1 时, PDMA 功能使能, 将数据从特定的存储器地址移到 SCI 发送缓冲器中。同样地, 当接收到数据从而置位接收完成标志位 RXCF, 且 RXDMA 位被置 1 时, PDMA 功能使能, 将数据从 SCI 接收缓冲器移到特定的存储器地址中。

详细的 PDMA 配置信息, 请参考 PDMA 章节。

寄存器列表

有几个与智能卡功能相关的寄存器。一些寄存器控制 SCI 所有功能以及中断, 而一些寄存器包含状态位, 用来表示智能卡的数据传输情况和错误条件。另外有两个 SCI 发送和接收相关的寄存器, 分别用于存放将要发送到外部智能卡的数据, 或从外部智能卡收到的数据。下表显示了 SCI 相关寄存器和复位值。

表 63. SCI 寄存器列表

寄存器	偏移量	描述	复位值
CR	0x000	SCI 控制寄存器	0x0000_0000
SR	0x004	SCI 状态寄存器	0x0000_0080
CCR	0x008	SCI 通信控制寄存器	0x0000_0000
ETUR	0x00C	SCI 基本时间单元寄存器	0x0000_0174
GTR	0x010	SCI 保护时间寄存器	0x0000_000C
WTR	0x014	SCI 等待时间寄存器	0x0000_2580
IER	0x018	SCI 中断使能寄存器	0x0000_0000
IPR	0x01C	SCI 中断挂起寄存器	0x0000_0000
TXB	0x020	SCI 发送缓冲器	0x0000_0000
RXB	0x024	SCI 接收缓冲器	0x0000_0000
PSCR	0x028	SCI 预分频寄存器	0x0000_0000

寄存器描述

SCI 控制寄存器 – CR

该寄存器包含 SCI 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						RXDMA	TXDMA
类型 / 复位						RW 0	RW 0
7	6	5	4	3	2	1	0
保留位	DETCNF	ENSCI	RETRY	SCIM	WTEN	CREP	CONV
类型 / 复位	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	域	描述															
[9]	RXDMA	SCI RX DMA 使能位 0: 除能 1: 使能															
[8]	TXDMA	SCI TX DMA 使能位 0: 除能 1: 使能															
[6]	DETCNF	卡开关类型选择位 0: 如果没有卡，开关是常断开 1: 如果没有卡，开关是常闭合 <table><tr><th>DETCNF</th><th>SCI_DET 引脚</th><th>状态</th></tr><tr><td>0</td><td>1</td><td>无卡插入</td></tr><tr><td>0</td><td>0</td><td>卡插入</td></tr><tr><td>1</td><td>1</td><td>卡插入</td></tr><tr><td>1</td><td>0</td><td>无卡插入</td></tr></table>	DETCNF	SCI_DET 引脚	状态	0	1	无卡插入	0	0	卡插入	1	1	卡插入	1	0	无卡插入
DETCNF	SCI_DET 引脚	状态															
0	1	无卡插入															
0	0	卡插入															
1	1	卡插入															
1	0	无卡插入															
[5]	ENSCI	此位由应用程序置位和清零，以设置智能卡检测器的开关类型。 SCI 有限状态机制 (FSM) 使能位 0: SCI FSM 除能并强制进入初始状态 1: SCI FSM 使能															
[4]	RETRY	一个奇偶错检验误发生时字符传输重发次数选择 0: 奇偶检验错误发生时数据最多重发 5 次 1: 奇偶检验错误发生时数据最多重发 4 次 此位仅当 CREP 被置 1 时可用。当此位被置 1，一旦奇偶校验错误发生，数据将被发送或接收最多 4 次。如果此位为 0，一旦奇偶校验错误发生，数据将被传输最多 5 次。															

位	域	描述
[3]	SCIM	<p>SCI 模式选择位</p> <p>0: 手动模式下 SCI 数据传输 1: SCI 模式下 SCI 数据传输</p> <p>此位由应用程序置位或清零用于选择 SCI 数据传输模式。如果此位清零, SCI_DIO 引脚状态与 CCR 寄存器中的 CDIO 位相同。如果置 1, SCI_DIO 引脚由内部 SCI 控制电路驱动。在数据传输类型从手动模式切换到 SCI 模式前, CDIO 位必须被置 1 来避免 SCI 故障。</p>
[2]	WTEN	<p>等待时间计数器使能控制位</p> <p>0: 停止计数 1: 开始计数</p> <p>此位由应用程序置位和清零。当 WTEN 位清零, 对 WTR 寄存器写入数值, 此值将载入到等待时间计数器中。如果此位置 1, 等待时间计数器将使能, 在每次起始位出现时自动重新加载。</p>
[1]	CREP	<p>奇偶校验错误条件下的字符自动重发使能控制位</p> <p>0: 奇偶校验错误时不重发 1: 奇偶校验错误时自动重发</p> <p>CREP 位由应用程序置位和清零。当 CREP 位被清除为 0, 在接收模式中收到的数据发生奇偶校验错误后, RXCF 和 PARF 标志将被置位, 而在发送模式下 PARF 被置位, TXCF 不会被置位。如果 CREP 位被置 1, 字符传输将自动被重试 4 或 5 次取决于 RETRY 位的值。在发送模式, 如果发生奇偶校验错误, 则字符将会被重复发送。在第 4 或第 5 次发送结束时奇偶校验错误标志 PARF 将被置位, 但 TXCF 不会被置位。在接收模式中, 如果接收到的数据有一个奇偶校验错误, SCI 将通知外部智能卡重复发送 4 或 5 次, 然后在第 4 或第 5 次接收结束时 PARF 和 RXCF 标志都将被置位。</p>
[0]	CONV	<p>数据方向协议选择位</p> <p>0: LSB 先发送; SCI_DIO 引脚上逻辑高电平代表数据“1”, 奇偶校验位跟随在 MSB 后 1: MSB 先发送; SCI_DIO 引脚上逻辑低电平代表数据“1”, 奇偶校验位跟随在 LSB 后</p> <p>该位由应用程序置位和清零, 以选择数据是 LSB 首先发送还是 MSB 首先发送。当协议的方向与外部智能卡指定的方向相同时, 只有 RXCF 被设置为 1, 无奇偶校验错误。否则, 在收到数据后 RXCF 和 PARF 都将被设置为 1。</p>

SCI 状态寄存器 – SR

该寄存器包含 SCI 状态位。

偏移量： 0x004

复位值： 0x0000_0080

类型 / 复位	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
	TXBEF	CPREF	保留位		WTF	TXCF	RXCF	PARF
类型 / 复位	RO	1 RO	0		RO	0 W0C	0 RO	0 W0C

位	域	描述
[7]	TXBEF	发送缓冲器空请求标志位 0：发送缓冲器不为空 1：发送缓冲器空 此位用来表示发送缓冲器是否为空，由硬件自动置位和清零。
[6]	CPREF	卡存在请求标志位 0：没有卡 1：有卡 此位用来表示是否有卡，由硬件自动置位和清零。ENSCI 位置位时，卡存在检测功能使能。
[3]	WTF	等待时间计数器下溢请求标志位 0：等待时间计数器无下溢 1：等待时间计数器下溢 此位由应用程序置位和清零，表示等待时间计数器是否下溢。
[2]	TXCF	字符发送请求标志位 0：无字符发送请求 1：字符已发送 此位由硬件置位，由应用程序写入 0 清零。
[1]	RXCF	字符接收请求标志位 0：无字符接收请求 1：字符已接收 此位由硬件置位，由应用程序读取 RXB 寄存器后清零。不管奇偶校验的结果为何，当接收到字符，RXCF 位将设置为 1。 当字符已收到，存储在 RXB 寄存器的已接收到的数据应被转移到应用程序指定的数据存储单元。如果在下一个字符移入结束前 RXB 寄存器的内容没被读取，存储在 RXB 寄存器的数据将被覆盖。

位	域	描述
[0]	PARF	奇偶校验错误请求标志位 0: 无奇偶校验错误请求 1: 奇偶校验错误已发生 此位由硬件置位，由应用程序清零。当接收到字符，奇偶校验电路检查校验正确与否。如果奇偶校验的结果是不正确的，奇偶校验错误请求标志 PARF 将被设置为 1。否则，PARF 位将保持为零。在发送模式，当外部智能卡告知 SCI 发送的字符有奇偶校验错误，PARF 位也会被置位。

SCI 通信控制寄存器 – CCR

该寄存器指定了 SCI 引脚的设置和时钟选择。

偏移量:	0x008
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	CLKSEL 保留位 CDIO CCLK 保留位
	RW 0 RW 1 RW 0

位	域	描述
[7]	CLKSEL	智能卡时钟选择位 0: CCLK 位的内容为外部 SCI_CLK 引脚的状态 1: 外部 SCI_CLK 引脚的时钟输出来自 f _{PSC_CLK} 时钟 此位用来选择外部 SCI_CLK 引脚的时钟源。它由应用程序置位和清零。建议在 CLKSEL 位由 1 切换到 0 之前，写入一个已知值到 CCLK 位使时钟激活后处于一个已知电平。
[3]	CDIO	SCI_DIO 引脚控制位 0: SCI_DIO 引脚是逻辑电平 0 1: SCI_DIO 引脚处于开漏状态 仅当 CR 寄存器中 SCIM 位清零用来配置 SCI 工作在手动模式时，此位才可用。手动模式下，由应用程序置位和清零以控制外部 SCI_DIO 引脚状态。读取此位将返回 SCI_DIO 引脚的当前状态。
[2]	CCLK	SCI_CLK 引脚控制位 0: SCI_CLK 引脚为逻辑电平 0 1: SCI_CLK 引脚为逻辑电平 1 此位由应用程序置位和清零以控制外部 SCI_CLK 引脚状态。读取此位将返回寄存器中此位的当前值而非 SCI_CLK 引脚的当前状态。在 CLKSEL 位由 1 切换到 0 之前，写入某一已知值到 CCLK 位以确保时钟保持在一个已知电平值。

SCI 基本时间单元寄存器 – ETUR

该寄存器定义的 ETU 值按基本时间单元计数器章节所述公式计算。它还包括 ETU 时间间隔补偿功能的控制位。

偏移量: 0x00C
复位值: 0x0000_0174

	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				
	COMP	保留位				ETU						
类型 / 复位	RW	0				RW	0	RW	0	RW	1	
	7	6	5	4	3	2	1	0				
	ETU											
类型 / 复位	RW	0	RW	1	RW	1	RW	1	RW	0	RW	0

位	域	描述
[15]	COMP	基本时间单元补偿模式使能控制位 0: 补偿模式除能 1: 补偿模式使能 此位由应用程序置位和清零，用来控制 ETU 补偿功能。有关补偿功能的细节请参照基本时间单元章节。
[10:0]	ETU	一个字符数据位的 ETU 值 该位域由应用程序配置用于修改 ETU 的值。注意，ETU 的值必须在 0x00C ~ 0x7FF 范围内。为获得十进制的最大 ETU 值 2048，需向此位写入 0x000 值。

SCI 保护时间寄存器 – GTR

该寄存器定义了保护时间值。该保护时间值可从复位应答包获取，详细参考保护时间计数器章节。

偏移量： 0x010
复位值： 0x0000_000C

	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							GT	
类型 / 复位								RW	0
	7	6	5	4	3	2	1	0	
	GT								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	域	描述
[8:0]	GT	字符保护时间值 此位域由应用程序配置用于修改保护时间。在当前保护时间周期结束时，更新的 GT 值将被载入到 GT 计数器中。注意，GT 值必须在 0x00C ~ 0x1FF 范围内。

SCI 等待时间寄存器 – WTR

该寄存器定义了等待时间值。该等待时间值可从复位应答包获取，详细参考等待时间计数器章节。

偏移量： 0x014
复位值： 0x0000_2580

	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	WT								
	RW	0	RW	0	RW	0	RW	0	RW
	15	14	13	12	11	10	9	8	
类型 / 复位	WT								
	RW	0	RW	0	RW	1	RW	0	RW
	7	6	5	4	3	2	1	0	
类型 / 复位	WT								
	RW	1	RW	0	RW	0	RW	0	RW

位	域	描述
[23:0]	WT	字符等待时间值通过 ETU 表示 (0/16777215)。 此位域由应用程序配置用于修改等待时间。等待时间计数器更新值的重载条件已经在等待时间计数器章节描述过，更多的细节请参考等待时间计数器章节。注意，WT 的值在 0x00_0000 ~ 0xFF_FFFF 之间。

SCI 中断使能寄存器 – IER

该寄存器定义了 SCI 中控制所有中断事件的中断控制使能位。

偏移量： 0x018

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
TXBEE	CARDIRE	保留位		WTE	TXCE	RXCE	PARE
RW	0	RW	0	RW	0	RW	0
类型 / 复位							

位	域	描述
[7]	TXBEE	发送缓冲器空中断使能控制位 0: 除能 1: 使能 此位由应用程序置位和清零来控制发送缓冲器空中断。如果此位置 1，当发送缓冲器为空时，将产生发送缓冲器空中断。
[6]	CARDIRE	卡插入 / 移除中断使能控制位 0: 除能 1: 使能 此位由应用程序置位和清零来控制卡插入 / 移除中断。如果此位置 1，当外部智能卡被插入或移除时，将产生卡插入 / 移除中断。
[3]	WTE	产生等待时间计数器下溢中断使能控制位 0: 除能 1: 使能 此位由应用程序置位和清零来控制产生等待时间计数器下溢中断。如果此位置 1，当等待时间计数器下溢时，将产生等待时间计数器下溢中断。
[2]	TXCE	字符发送完成中断使能控制位 0: 除能 1: 使能 此位由应用程序置位和清零来控制字符发送完成中断。如果此位置 1，在字符发送完成时将产生字符发送完成中断。
[1]	RXCE	字符接收完成中断使能控制位 0: 除能 1: 使能 此位由应用程序置位和清零来控制字符接收完成中断。如果此位置 1，在字符接收完成时将产生字符接收完成中断。
[0]	PARE	奇偶校验错误中断使能控制位 0: 除能 1: 使能 此位由应用程序置位和清零来控制奇偶校验错误中断。如果此位置 1，在奇偶校验错误发生时将产生奇偶校验错误中断。

SCI 中断挂起寄存器 – IPR

此寄存器包含 SCI 所有中断事件的中断挂起标志位。这些标志位可以通过相应的中断使能控制位屏蔽。

偏移量: 0x01C
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
TXBEP	CARDIRP	保留位		WTP	TXCP	RXCP	PARP
RC	0	RC	0	RC	0	RC	0
类型 / 复位							

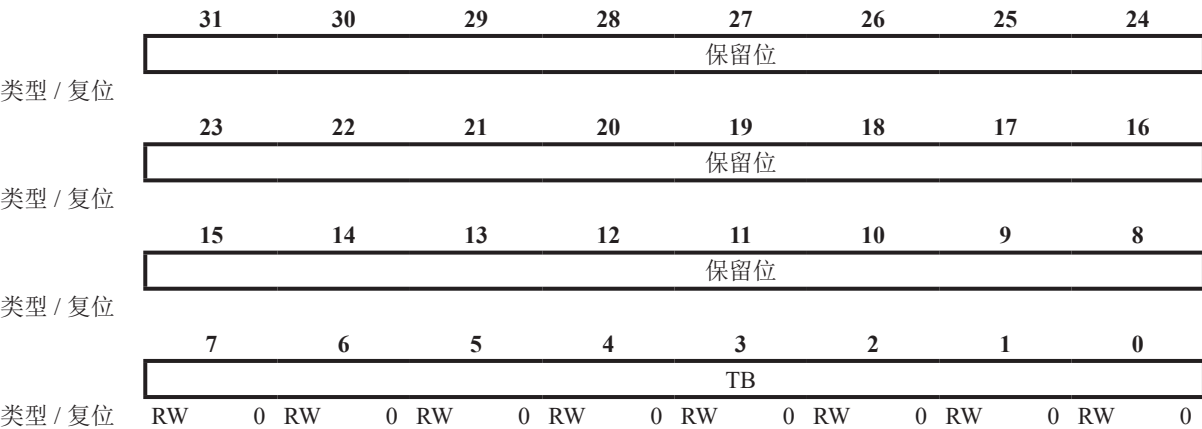
位	域	描述
[7]	TXBEP	发送缓冲器空中断挂起标志位 0: 无中断挂起 1: 中断挂起 此位由硬件置位，通过使用应用程序读访问该寄存器时清零。用来表示发送缓冲器空中断挂起与否。如果相应的中断使能控制位被设置为 1 且发送缓冲器为空，该位会被置为 1，表示发送缓冲器空中断正在等待处理。
[6]	CARDIRP	卡插入 / 移除中断挂起标志位 0: 无中断挂起 1: 中断挂起 此位由硬件置位，通过使用应用程序读访问该寄存器时清零。用来表示外部智能卡插入 / 移除中断挂起与否。如果相应的中断使能控制位被设置为 1 且有卡插入或移除动作发生，该位会被置为 1，表示卡插入 / 移除中断正在等待处理。
[3]	WTP	等待时间计数器下溢中断挂起标志位 0: 无中断挂起 1: 中断挂起 此位由硬件置位，通过使用应用程序读访问该寄存器时清零。用来表示等待时间计数器下溢中断挂起与否。如果相应的中断使能控制位被设置为 1 且有等待时间计数器下溢发生，该位会被置为 1，表示等待时间计数器下溢中断正在等待处理。
[2]	TXCP	字符发送完成中断挂起标志位 0: 无中断挂起 1: 中断挂起 此位由硬件置位，通过使用应用程序读访问该寄存器时清零。用来表示字符发送完成中断挂起与否。如果相应的中断使能控制位被设置为 1 且字符发送完成，该位会被置为 1，表示字符发送完成中断正在等待处理。
[1]	RXCP	字符接收完成中断挂起标志位 0: 无中断挂起 1: 中断挂起 此位由硬件置位，通过使用应用程序读访问该寄存器时清零。用来表示字符接收完成中断挂起与否。如果相应的中断使能控制位被设置为 1 且字符接收完成，该位会被置为 1，表示字符接收完成中断正在等待处理。

位	域	描述
[0]	PARP	奇偶校验错误中断挂起标志位 0: 无中断挂起 1: 中断挂起 此位由硬件置位，通过使用应用程序读访问该寄存器时清零。用来表示奇偶校验错误中断挂起与否。如果相应的中断使能控制位被设置为 1 且奇偶校验错误发生时，该位会被置为 1，表示奇偶校验错误中断正在等待处理。

SCI 发送缓冲器 – TXB

该寄存器用来存储要发送的 SCI 数据。

偏移量: 0x020
复位值: 0x0000_0000



位	域	描述
[7:0]	TB	要发送的 SCI 数据字节

SCI 接收缓冲器 – RXB

该寄存器用来存储接收到的 SCI 数据。

偏移量： 0x024
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
	RB							
类型 / 复位	RO	0	RO	0	RO	0	RO	0

位	域	描述
[7:0]	RB	接收到的 SCI 数据字节

SCI 预分频器寄存器 – PSCR

该寄存器定义了 SCI 内部时钟的预分频器的分频比。

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
	保留位							
类型 / 复位								
	7	6	5	4	3	2	1	0
	保留位		PSC					
类型 / 复位			RW	0	RW	0	RW	0

位	域	描述
[5:0]	PSC	SCI 预分频器分频比 0: $f_{PSC_CK} = f_{PCLK}$ 1~63: $f_{PSC_CK} = \frac{f_{PCLK}}{2 \times PSC}$

24 USB 设备控制器 (USB)

简介

该 USB 设备控制器符合 USB 2.0 全速规范。其具有一个控制端点 (端点 0) 和 7 个可配置的端点 (EP1 ~ EP7)。一个 1024-byte 的 EP_SRAM 用作端点缓冲器。每个端点缓冲器的大小可通过设置相应寄存器来得到, 为各种应用提供较大的灵活性。内置的 USB 全速收发器减小了整个系统的复杂性并且降低了成本。USB 包含了挂起和恢复功能, 满足了低功耗产品的需求。下图显示了 USB 方框图。

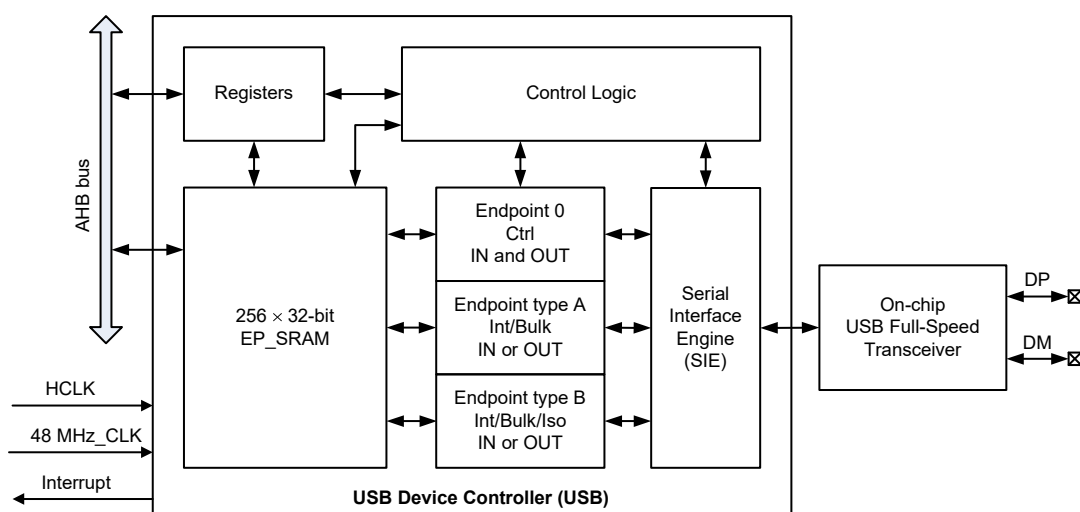


图 168. USB 方框图

特性

- 符合 USB 2.0 全速 (12 Mbps) 设备规范
- 完全集成的 USB 全速收发器
- 1 个控制端点 (EP0) 用于控制传输
- 3 个单缓冲端点 (EP1 ~ EP3) 用于批量和中断传输
- 4 个双缓冲端点 (EP4 ~ EP7) 用于批量、中断和等时传输
- 1024 字节 EP_SRAM 用作端点数据缓冲器

功能描述

端点

USB 端点 0 是唯一一个用于 USB 控制传输的双向端点。该设备也包含 7 个单向的端点，用于其它 USB 传输类型。有三个端点 (EP1 ~ EP3) 支持单缓冲功能用于批量和中断 IN 或 OUT 数据传输。有另外四个端点 (EP4 ~ EP7) 支持单缓冲或双缓冲功能用于批量、中断和等时 IN 或 OUT 数据传输。7 个单向端点 (EP1 ~ EP7) 的地址可由应用软件设置。下表列出了端点特性。

表 64. 端点特性

端点编号	端点地址	传输类型	方向	缓冲器类型
0	固定	控制	IN 和 OUT	单缓冲
1 ~ 3	可设置	中断 / 批量	IN 或 OUT	单缓冲
4 ~ 7	可设置	中断 / 批量 / 等时	IN 或 OUT	单或双缓冲

EP_SRAM

USB 控制器包含一个专用的存储器空间 EP_SRAM，用作 USB 端点缓冲器。EP_SRAM 与 AHB 总线相连，可由 MCU 和 PDMA 访问。EP_SRAM 基址是 0x400A_A000，偏移量范围是 0x000 ~ 0x3FF。EP_SRAM 的前两个字被预留用于端点 0 临时存储 8-byte SETUP 数据。因此，端点缓冲器的有效起始地址应从 0x400A_A008 开始并且对齐到 4-byte 边界。每个端点缓冲器容量都是可编程的。下表列出了符合 USB 2.0 全速设备规范的 USB 端点缓冲器的最大容量。

表 65. USB 数据类型和缓冲器容量

传输类型	方向	支持的缓冲器容量	带宽	CRC	重试
控制	双向	8, 16, 32, 64 字节	不确定	Yes	自动
批量	单向	8, 16, 32, 64 字节	不确定	Yes	Yes
中断	单向	≤ 64 字节	不确定	Yes	Yes
等时	单向	< 512 字节	确定	Yes	No

在下面端点缓冲器分配范例中，端点“4”设置作为双缓冲的批量 IN 端点，而端点“5”设置作为双缓冲的批量 OUT 端点。每个端点缓冲器的容量是 64 字节。

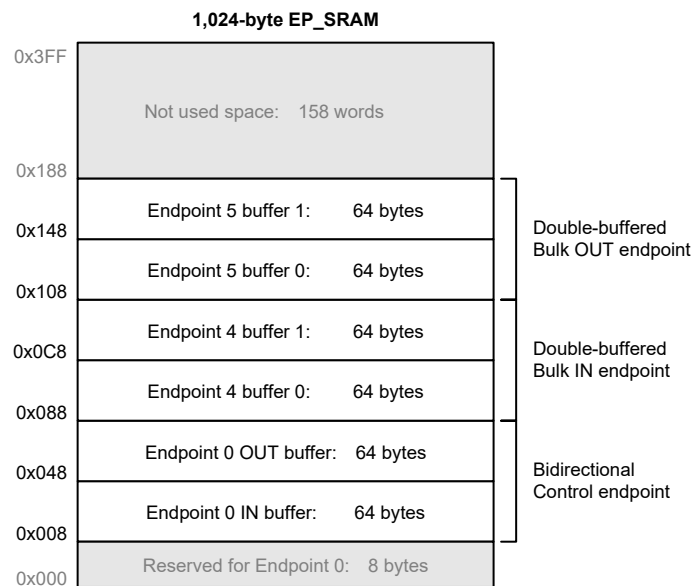


图 169. 端点缓冲器分配范例

串行接口引擎 – SIE

串行接口引擎 SIE 与 USB 全速收发器和内部 USB 控制电路相连，为发送和接收数据提供一个临时缓冲器。当 USB 模块接收数据、发送数据或发送恢复信号用于远程控制时，SIE 也可解码 SE0 信号、SE1 信号、J-state、K-state、USB RESET 事件和数据包结束事件信号 EOP。SIE 会检测 SOF 包的数量并对 USB 控制电路产生 SOF 中断信号，SIE 包含从并行到串行或串行到并行的数据格式转换功能。SIE 还包含 CRC 检测和产生、PID 验证 / 产生、bit-stuffing 和 debit-stuffing 功能。

双缓冲

当要进行等时传输或大批量传输时，建议使用有双缓冲功能的端点并使能其双缓冲功能。在 OUT 事务处理期间，双缓冲中的一个缓冲器用来存储之前由 USB 主机发送的数据用于单片机处理；并且在另一个缓冲器中，硬件会确保端点继续接收当前数据包，进行 IN 事务处理时亦然。使用双缓冲器功能可达到数据传输的最大速率。关于双缓冲功能的用法请参考 USBEPnCSR 寄存器中的 UDBTG 和 MDBTG 控制位的相关描述，其中 n 的范围是 4 ~ 7。

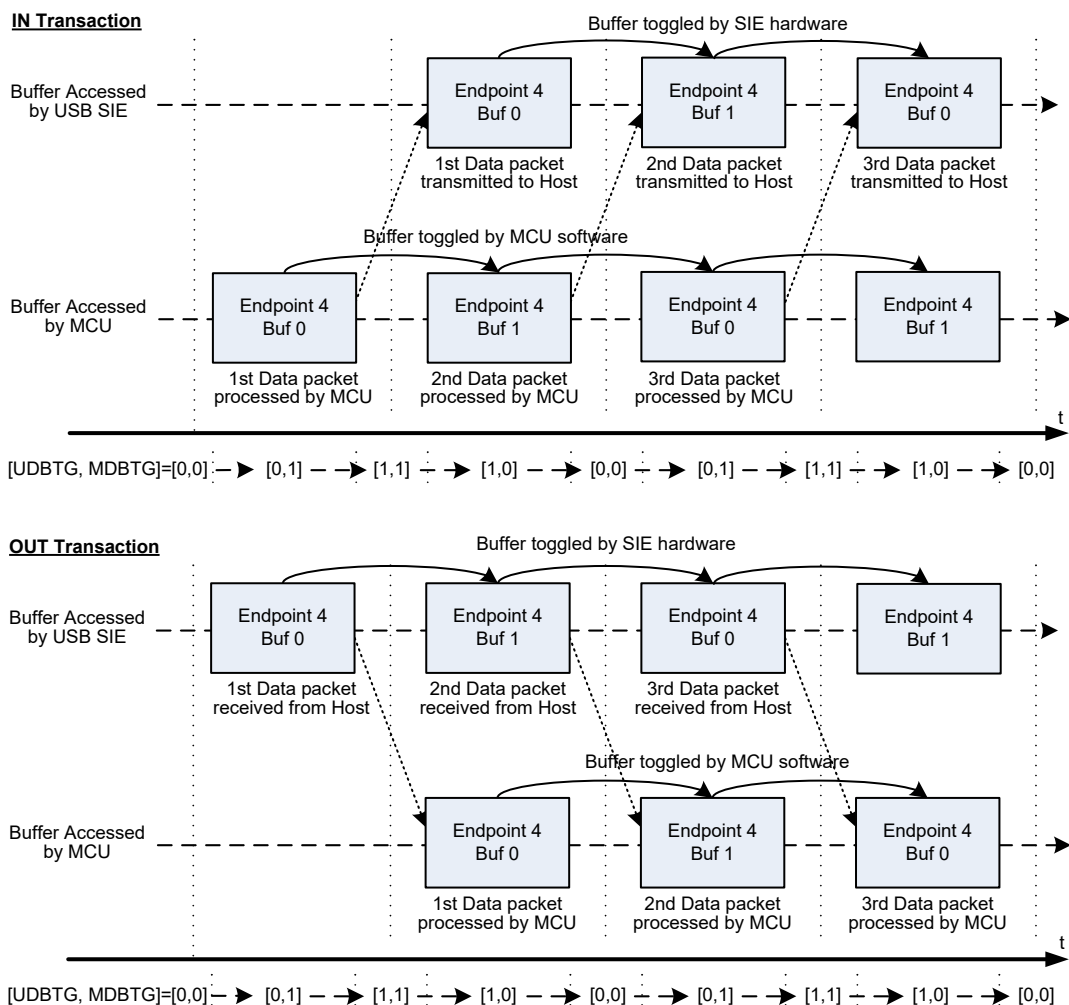


图 170. 双缓冲工作范例

挂起模式和唤醒

根据 USB 的规格, USB 设备必须在经过 3 ms 总线空闲时间后进入挂起模式。当 USB 设备进入挂起模式, USB 总线的电流消耗不能大于 500 μ A, 以满足挂起模式下电流规格要求。如果总线空闲状态持续 3 ms, USB 控制电路将产生挂起中断。此时, USBCSR 寄存器中的 LPMODE 和 PDWN 位必须由软件置 1。通过 LPMODE 位决定 USB 控制器是否进入低功耗模式 (将 USB 总线保持在复位状态), 而 PDWN 位用来决定内置的 USB 全速收发器是否关闭。

有两种方式使 USB 主机唤醒 USB 设备。一种是发送一个 USB 复位信号 SE0, 另一种是发送一个 USB 设备恢复信号 K-state。唤醒信号产生后, 不论是检测到 SE0 信号还是检测到 K-state 信号, USB 设备都将被唤醒。

远程唤醒

USB 设备具有远程唤醒功能, 可以通过置 USBCSR 寄存器中的 GENRSM 位为 1 来发送一个恢复请求信号以唤醒 USB 主机。一旦 USB 主机接收到来自 USB 设备的远程唤醒信号, 将发送一个恢复信号给 USB 设备。

寄存器列表

下表显示了 USB 寄存器及其复位值。

表 66. USB 寄存器列表

寄存器	偏移量	描述	复位值
USBCSR	0x000	USB 控制和状态寄存器	0x0000_00X6
USBIER	0x004	USB 中断使能寄存器	0x0000_0000
USBISR	0x008	USB 中断状态寄存器	0x0000_0000
USBFCR	0x00C	USB 帧计数寄存器	0x0000_0000
USBDEVAR	0x010	USB 设备地址寄存器	0x0000_0000
USBEP0CSR	0x014	USB 端点 0 控制和状态寄存器	0x0000_0002
USBEP0IER	0x018	USB 端点 0 中断使能寄存器	0x0000_0000
USBEP0ISR	0x01C	USB 端点 0 中断状态寄存器	0x0000_0000
USBEP0TCR	0x020	USB 端点 0 传输计数寄存器	0x0000_0000
USBEP0CFGR	0x024	USB 端点 0 配置寄存器	0x8000_0008
USBEP1CSR	0x028	USB 端点 1 控制和状态寄存器	0x0000_0002
USBEP1IER	0x02C	USB 端点 1 中断使能寄存器	0x0000_0000
USBEP1ISR	0x030	USB 端点 1 中断状态寄存器	0x0000_0000
USBEP1TCR	0x034	USB 端点 1 传输计数寄存器	0x0000_0000
USBEP1CFGR	0x038	USB 端点 1 配置寄存器	0x1000_03FF
USBEP2CSR	0x03C	USB 端点 2 控制和状态寄存器	0x0000_0002
USBEP2IER	0x040	USB 端点 2 中断使能寄存器	0x0000_0000
USBEP2ISR	0x044	USB 端点 2 中断状态寄存器	0x0000_0000
USBEP2TCR	0x048	USB 端点 2 传输计数寄存器	0x0000_0000
USBEP2CFGR	0x04C	USB 端点 2 配置寄存器	0x1000_03FF
USBEP3CSR	0x050	USB 端点 3 控制和状态寄存器	0x0000_0002
USBEP3IER	0x054	USB 端点 3 中断使能寄存器	0x0000_0000
USBEP3ISR	0x058	USB 端点 3 中断状态寄存器	0x0000_0000
USBEP3TCR	0x05C	USB 端点 3 传输计数寄存器	0x0000_0000

寄存器	偏移量	描述	复位值
USBEP3CFGR	0x060	USB 端点 3 配置寄存器	0x1000_03FF
USBEP4CSR	0x064	USB 端点 4 控制和状态寄存器	0x0000_0002
USBEP4IER	0x068	USB 端点 4 中断使能寄存器	0x0000_0000
USBEP4ISR	0x06C	USB 端点 4 中断状态寄存器	0x0000_0000
USBEP4TCR	0x070	USB 端点 4 传输计数寄存器	0x0000_0000
USBEP4CFGR	0x074	USB 端点 4 配置寄存器	0x1000_03FF
USBEP5CSR	0x078	USB 端点 5 控制和状态寄存器	0x0000_0002
USBEP5IER	0x07C	USB 端点 5 中断使能寄存器	0x0000_0000
USBEP5ISR	0x080	USB 端点 5 中断状态寄存器	0x0000_0000
USBEP5TCR	0x084	USB 端点 5 传输计数寄存器	0x0000_0000
USBEP5CFGR	0x088	USB 端点 5 配置寄存器	0x1000_03FF
USBEP6CSR	0x08C	USB 端点 6 控制和状态寄存器	0x0000_0002
USBEP6IER	0x090	USB 端点 6 中断使能寄存器	0x0000_0000
USBEP6ISR	0x094	USB 端点 6 中断状态寄存器	0x0000_0000
USBEP6TCR	0x098	USB 端点 6 传输计数寄存器	0x0000_0000
USBEP6CFGR	0x09C	USB 端点 6 配置寄存器	0x1000_03FF
USBEP7CSR	0x0A0	USB 端点 7 控制和状态寄存器	0x0000_0002
USBEP7IER	0x0A4	USB 端点 7 中断使能寄存器	0x0000_0000
USBEP7ISR	0x0A8	USB 端点 7 中断状态寄存器	0x0000_0000
USBEP7TCR	0x0AC	USB 端点 7 传输计数寄存器	0x0000_0000
USBEP7CFGR	0x0B0	USB 端点 7 配置寄存器	0x1000_03FF

寄存器描述

USB 控制和状态寄存器 – USBCSR

该寄存器定义了 USB 控制位和 USB 数据线状态。

偏移量: 0x000
复位值: 0x0000_00X6

类型 / 复位	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			FREST	DPWKEN	DPPUEN	SRAMRSTC	ADRSET
类型 / 复位	7	6	5	4	3	2	1	0
	RXDM	RXDP	GENRSM	保留位	LPMODE	PDWN	FRES	保留位
类型 / 复位	RO	X RO	X RW	0	RW	0 RW	1 RW	1

位	域	描述
[12]	FREST	强制 USB 复位控制时序 此位用来决定 USB 复位状态解除的时序 0: 当 FRES 位清零, 立即解除 USB 复位状态 1: 当 FRES 位置高, 检测到 SE0 后再解除 USB 复位状态
[11]	DPWKEN	DP 唤醒使能位 0: DP 唤醒除能 1: DP 唤醒使能
[10]	DPPUEN	DP 上拉使能位 0: DP 上拉除能 1: DP 上拉使能
[9]	SRAMRSTC	EP_SRAM 复位条件 0: 当 (DP, DM) = (0, 0) 时, 复位 EP_SRAM 1: 无论 (DP, DM) 为何状态, 用户都可访问 EP_SRAM
[8]	ADRSET	设备地址控制位 此位用来控制 USB SIE 何时将设备地址更新为 USBDEVAR 寄存器的值。 0: 当有一个地址被写入 USBDEVAR 寄存器之后, SIE 立即更新设备地址。 1: USB 主机通过 IN 操作成功从设备读取数据之后 SIE 更新设备地址。 此位在设备地址更新后由 SIE 清零。
[7]	RXDM	接收的 DM 线状态 在挂起程序结束时, 此位用来观察 DM 数据线的状态以判断唤醒事件是否已经发生。
[6]	RXDP	接收的 DP 线状态 在挂起程序结束时, 此位用来观察 DP 数据线的状态以判断唤醒事件是否已经发生。
[5]	GENRSM	恢复请求发生控制位 通过向此位写 1 来产生一个恢复请求信号给 USB 主机。USB 远程唤醒功能总是使能的。USB 主机发送的恢复信号被接收时, 此位清零。

位	域	描述
[3]	LPMODE	低功耗模式控制位 此位用来决定 USB 工作模式。此位置位 1 将使 USB 强制进入低功耗模式。当 USB 总线唤醒事件被硬件检测到时，此位应由软件清零。 0: 退出低功耗模式 1: 进入低功耗模式
[2]	PDWN	暂停模式控制位 此位用来关闭 USB 总线功能。此位置位将暂停全速 USB PHY 收发器，这将断开 USB PHY 收发器与 USB 总线的连接。 0: 退出暂停模式 1: 进入暂停模式
[1]	FRES	强制 USB 复位控制 此位用来复位 USB 电路。置位此位将强制 USB 进入复位状态直到软件清零此位。如果 USBIER 寄存器中的相关中断使能位置 1，则将产生 USB 复位中断。所有相关的 USB 寄存器都将复位到默认值。 0: 解除 USB 复位 1: 强制 USB 复位

表 67. 恢复事件检测

[RXDP, RXDM] 状态	唤醒事件	恢复后软件需进行的动作
00	Root 复位	无
10	无 (总线噪声)	返回到挂起模式
01	Root 恢复	无
11	不允许 (总线噪声)	返回到挂起模式

USB 中断使能寄存器 – USBIER

该寄存器定义了 USB 中断使能控制。

偏移量： 0x004
复位值： 0x0000_0000

类型 / 复位	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
	EP7IE	EP6IE	EP5IE	EP4IE	EP3IE	EP2IE	EP1IE	EP0IE
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0
类型 / 复位	保留位	FRESIE	ESOFIE	SUSPIE	RSMIE	URSTIE	SOFIE	UGIE
		RW	0	RW	0	RW	0	RW

位	域	描述
[15:8]	EPnIE	端点 n 中断使能控制 (n = 0 ~ 7) 0: 除能 1: 使能
[6]	FRESIE	强制 USB 复位 (FRES) 中断使能控制位 0: 除能 1: 使能
[5]	ESOFIE	预期帧开始 (ESOF) 中断使能控制位 0: 除能 1: 使能
[4]	SUSPIE	挂起中断使能控制位 0: 除能 1: 使能
[3]	RSMIE	恢复中断使能控制位 0: 除能 1: 使能
[2]	URSTIE	USB 复位中断使能控制位 0: 除能 1: 使能
[1]	SOFIE	帧开始 (SOF) 中断使能控制位 0: 除能 1: 使能
[0]	UGIE	USB 总中断使能控制位 0: 除能 1: 使能 此位必须置 1 来使能相关 USB 中断功能。如果此位清零, 相关中断将不会产生, 然而, 相关中断标志位仍然会置起。

USB 中断状态寄存器 – USBISR

该寄存器定义了 USB 中断状态。

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	EP7IF	EP6IF	EP5IF	EP4IF	EP3IF	EP2IF	EP1IF	EP0IF
	WC	0	WC	0	WC	0	WC	0
	7	6	5	4	3	2	1	0
类型 / 复位	保留位	FRESIF	ESOFIF	SUSPIF	RSMIF	URSTIF	SOFIF	保留位
		WC	0	RW	0	WC	0	WC

位	域	描述
[15:8]	EPnIF	端点 n 中断标志位 (n = 0 ~ 7) 此位由硬件置位，表示相关端点中断产生。 向此位写 1 清零。注意，仅当 USBEPnISR 寄存器中的端点中断状态位是 0 时，此中断标志位才可清零。
[6]	FRESIF	强制 USB 复位 (FRES) 中断标志位 此位通过写 1 清零。此位由硬件置位，表示强制 USB 复位 (FRES) 完成。此位通过写 1 清零。
[5]	ESOFIF	预期帧开始 (ESOF) 中断标志位 当一个 SOF 包预期被接收时，此位由硬件置位。USB 主机每毫秒发送一个 SOF 包。如果 USB 设备硬件没有正确接收 SOF 包，当 USBIER 寄存器中的 ESOFIE 位置 1 时，将产生 ESOF 中断。如果产生三个连续的 ESOF 中断，也就是说 SOF 包已经被丢失 3 次，那么 SUSPIF 位将被置 1。如果定时器还没有被锁，也会导致 SOF 包发生丢失，此位会被置 1。 此位可读写。但只可写入 0，写 1 将无效。
[4]	SUSPIF	挂起中断标志位 当超过 3 ms 没有数据传输时，此位将由硬件置位，说明 USB 主机已经发送了挂起请求。USB 复位后挂起条件检测功能立刻使能。 此位通过写 1 清零。
[3]	RSMIF	恢复中断标志位 此位由硬件置位，表示设备恢复已经发生。 此位通过写 1 清零。

位	域	描述
[2]	URSTIF	USB 复位中断标志位 当检测到 USB 复位时，此位由硬件置位。当发生 USB 复位，内部协议状态机将复位，当 USBIER 寄存器中的 URSTIE 位置 1 时，将产生 USB 复位中断。数据接收和发送都将除能直到 URSTIF 位清零。除了 USB 设备地址 (USBDEVAR) 之外，USB 设置相关的寄存器 (USBCSR、USBIER、USBISR、USBFCR 和 USBDEVAR) 都不会因 USB 复位事件而复位，这确保 USB 复位中断能安全执行，并且让 USB 复位发生之前的任何数据处理可以完全由软件访问。因此，单片机必须合理地复位这些寄存器。USB 端点相关的寄存器 (USBEPnCSR、USBEPnISR 和 USBEPnTCR) 也因复位事件而复位，然而，端点配置 (USBEPnCFGR) 及中断使能 (USBEPnIER) 寄存器则不受 USB 复位的影响，仍然保持不变。 此位通过写 1 清零。
[1]	SOFIF	帧开始 SOF 中断标志位 当 SOF 包被接收时，此位由硬件置位。 此位通过写 1 清零。

USB 帧计数寄存器 – USBFCR

该寄存器定义了丢失 SOF (Start-of-Frame) 的数量和 USB 帧计数。

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					LSOF		SOFLCK			
类型 / 复位						RO	0	RO	0	RO	0
	15	14	13	12	11	10	9	8			
	保留位					FRNUM					
类型 / 复位						RO	0	RO	0	RO	0
	7	6	5	4	3	2	1	0			
	FRNUM										
类型 / 复位	RO	0	RO	0	RO	0	RO	0	RO	0	

位	域	描述
[18:17]	LSOF	丢失 SOF 包的数量 ESOFIF 位每置位一次，这些位由硬件写入且加 1。这些位用来计算丢失的 SOF 包的数量。一旦此位域中的值达到 3，即记录到 SOF 包已经被丢失三次，SUSPIF 标志位将被置位。当 SOF 包被接收到时，这些位被清零。
[16]	SOFLCK	SOF 锁存标志位 当 SOF 包在帧定时器超时之前被接收，此位由硬件置位。一旦此标志位置 1，由 USB 主机发送的帧数量将被载入到 USBFCR 寄存器中的帧数量位域 FRNUM。如果在 1 ms 帧时间段内，没有 SOF 包被接收，那么此位将清零。
[10:0]	FRNUM	帧数量 此位域存储由 USB 主机发送的帧数量。

USB 设备地址寄存器 – USBDEVAR

该寄存器定义了 USB 设备地址。

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	DEVA											
类型 / 复位		RW	0	RW	0	RW	0	RW	0	RW	0	RW	0

位	域	描述
[6:0]	DEVA	设备地址 此位域用来定义 USB 设备地址。当 USB 复位事件发生时，此位域清零。

USB 端点 0 控制和状态寄存器 – USBEP0CSR

该寄存器定义了端点 0 控制和状态位。

偏移量：0x014

复位值：0x0000_0002

类型 / 复位	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	STLRX	NAKRX	DTGRX	STLTX	NAKTX	DTGTX
			RW	0	RW	0	RW	0

位	域	描述
[5]	STLRX	接收 (OUT) 传输的 STALL 状态 当功能错误被检测到，此位由应用程序置 1，在 OUT 事务处理的握手阶段将返回一个 STALL 信号。也就是说，来自于 USB 主机的控制请求不被 USB 设备支持。当 SETUP 令牌被接收时，STALL 状态由硬件电路清零。 此位可读写，仅当写 1 时状态切换。
[4]	NAKRX	接收 (OUT) 传输的 NAK 状态 在 ACK 信号被发送给主机之后，此位由硬件电路从 0 切换到 1，在 OUT 事务处理的握手阶段将产生一个 NAK 信号。这意味着 USB 设备将暂时不能接收来自 USB 主机的数据。因此，会需要更多的时间来处理所接收的数据。 此位可读写，仅当写 1 时状态切换。
[3]	DTGRX	接收 (OUT) 传输的数据切换状态 此位包含数据切换位的预期值 (0 = DATA0, 1 = DATA1)，用于下一个要接收的数据包。当当前有效数据包被接收，且 USB 设备已发送给 USB 主机相应的 ACK 信号，硬件将切换此位，设备将准备接收下一个数据包。对于端点 0，在 SETUP 令牌已被接收且端点 0 已被寻址后，硬件会将此位切换到 1。此位也可由软件对其初始化以适用于特定的应用。 此位可读写，仅当写 1 时状态切换。
[2]	STLTX	发送 (IN) 传输的 STALL 状态 当功能错误被检测到，此位由应用程序置 1，将返回一个 STALL 信号以响应 IN 令牌。这意味着 USB 设备不能发送数据。当新的 SETUP 令牌被接收到时，STALL 状态由硬件电路清除。 此位可读写，仅当写 1 时状态切换。
[1]	NAKTX	发送 (IN) 传输的 NAK 状态 在 ACK 信号被设备接收之后，此位由硬件电路从 0 切换到 1，在 IN 事务处理的握手阶段将产生一个 NAK 信号。这意味着 USB 设备暂时不能向 USB 主机发送数据。因此，应用软件有更多的时间为发送数据作准备。 此位可读写，仅当写 1 时状态切换。

位	域	描述
[0]	DTGTX	发送 (IN) 传输的数据切换状态 此位包含数据切换位所需要的值 (0 = DATA0, 1 = DATA1)，用于下一个要发送的数据包。当当前数据包被 USB 设备发送，且由 USB 主机发送的 ACK 信号被接收，硬件将切换此位，并将发送下一个数据包。对于端点 0，在 SETUP 令牌已被接收且端点 0 已被寻址后，硬件会将此位切换到 1。在特定条件下，此位也可由软件对其初始化。 此位可读写，仅当写 1 时状态切换。

USB 端点 0 中断使能控制寄存器 – USBEP0IER

该寄存器定义了端点 0 中断使能控制位。

偏移量:	0x018																												
复位值:	0x0000_0000																												
	<table><tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td></tr><tr><td colspan="8">保留位</td></tr></table>	31	30	29	28	27	26	25	24	保留位																			
31	30	29	28	27	26	25	24																						
保留位																													
类型 / 复位																													
	<table><tr><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr><tr><td colspan="8">保留位</td></tr></table>	23	22	21	20	19	18	17	16	保留位																			
23	22	21	20	19	18	17	16																						
保留位																													
类型 / 复位																													
	<table><tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td></tr><tr><td colspan="4">保留位</td><td>ZLRXIE</td><td>SDERIE</td><td>SDRXIE</td><td>STRXIE</td></tr><tr><td colspan="4"></td><td>RW</td><td>0</td><td>RW</td><td>0</td><td>RW</td><td>0</td><td>RW</td><td>0</td></tr></table>	15	14	13	12	11	10	9	8	保留位				ZLRXIE	SDERIE	SDRXIE	STRXIE					RW	0	RW	0	RW	0	RW	0
15	14	13	12	11	10	9	8																						
保留位				ZLRXIE	SDERIE	SDRXIE	STRXIE																						
				RW	0	RW	0	RW	0	RW	0																		
类型 / 复位																													
	<table><tr><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr><tr><td>UERIE</td><td>STLIE</td><td>NAKIE</td><td>IDTXIE</td><td>ITRXIE</td><td>ODOVIE</td><td>ODRXIE</td><td>OTRXIE</td></tr><tr><td>RW</td><td>0</td><td>RW</td><td>0</td><td>RW</td><td>0</td><td>RW</td><td>0</td><td>RW</td><td>0</td><td>RW</td><td>0</td></tr></table>	7	6	5	4	3	2	1	0	UERIE	STLIE	NAKIE	IDTXIE	ITRXIE	ODOVIE	ODRXIE	OTRXIE	RW	0	RW	0	RW	0	RW	0	RW	0	RW	0
7	6	5	4	3	2	1	0																						
UERIE	STLIE	NAKIE	IDTXIE	ITRXIE	ODOVIE	ODRXIE	OTRXIE																						
RW	0	RW	0	RW	0	RW	0	RW	0	RW	0																		
类型 / 复位																													

位	域	描述
[11]	ZLRXIE	零长度数据接收中断使能控制位 0: 除能 1: 使能
[10]	SDERIE	SETUP 数据错误中断使能控制位 0: 除能 1: 使能
[9]	SDRXIE	SETUP 数据接收中断使能控制位 0: 除能 1: 使能
[8]	STRXIE	SETUP 令牌接收中断使能控制位 0: 除能 1: 使能
[7]	UERIE	USB 错误中断使能控制位 0: 除能 1: 使能
[6]	STLIE	STALL 发送中断使能控制位 0: 除能 1: 使能

位	域	描述
[5]	NAKIE	NAK 发送中断使能控制位 0: 除能 1: 使能
[4]	IDTXIE	IN 数据发送中断使能控制位 0: 除能 1: 使能
[3]	ITRXIE	IN 令牌接收中断使能控制位 0: 除能 1: 使能
[2]	ODOVIE	OUT 数据缓冲器溢出中断使能控制位 0: 除能 1: 使能
[1]	ODRXIE	OUT 数据接收中断使能控制位 0: 除能 1: 使能
[0]	OTRXIE	OUT 令牌接收中断使能控制位 0: 除能 1: 使能

USB 端点 0 中断状态寄存器 – USBEP0ISR

该寄存器定义了端点 0 中断状态。

偏移量: 0x01C
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				ZLRXIF	SDERIF	SDRXIF	STRXIF
类型 / 复位				WC	0	WC	0
WC				0	WC	0	WC
7	6	5	4	3	2	1	0
UERIF	STLIF	NAKIF	IDTXIF	ITRXIF	ODOVIF	ODRXIF	OTRXIF
类型 / 复位							
WC	0	WC	0	WC	0	WC	0
WC	0	WC	0	WC	0	WC	0

位	域	描述
[11]	ZLRXIF	零长度数据接收中断标志位 当接收到一个零长度数据包时，此位由硬件置位。 当 SETUP 令牌被接收时此位由硬件清零，或者通过写 1 清零。
[10]	SDERIF	SETUP 数据错误中断标志位 当 SETUP 数据包的长度不是 8 字节时，此位由硬件置位。 当 SETUP 令牌被接收时此位由硬件清零，或者通过写 1 清零。

位	域	描述
[9]	SDRXIF	SETUP 数据接收中断标志位 当接收到来自于 USB 主机的一个 SETUP 数据包时，此位被硬件置位。当 SETUP 令牌被接收时此位由硬件清零，或者通过写 1 清零。在下一个 SETUP 数据包被接收前，如果已接收到的 SETUP 数据没有被应用软件访问，则 SETUP 数据缓冲器将被覆盖。
[8]	STRXIF	SETUP 令牌接收中断标志位 当 SETUP 令牌被接收时，此位由硬件置位。写 1 时清零。
[7]	UERIF	USB 错误中断标志位 在端点 0 事务处理过程中，当有错误发生时，此位由硬件置位。 当 SETUP 令牌被接收时此位由硬件清零，或者通过写 1 清零。
[6]	STLIF	STALL 发送中断标志位 当 STALL 信号被发送以响应 IN 或 OUT 事务处理时，此位由硬件置位。 当 SETUP 令牌被接收时此位由硬件清零，或者通过写 1 清零。
[5]	NAKIF	NAK 发送中断标志位 当 NAK 信号被发送以响应 IN 或 OUT 事务处理时，此位由硬件置位。 当 SETUP 令牌被接收时此位由硬件清零，或者通过写 1 清零。
[4]	IDTXIF	IN 数据发送中断标志位 当数据包被发送到 USB 主机且从 USB 主机发送的 ACK 信号被接收时，此位由硬件置位。 当 SETUP 令牌被接收时此位由硬件清零，或者通过写 1 清零。
[3]	ITRXIF	IN 令牌接收中断标志位 当接收到来自于 USB 主机的 IN 令牌时，此位由硬件置位。 当 SETUP 令牌被接收时此位由硬件清零，或者通过写 1 清零。
[2]	ODOVIF	OUT 数据缓冲器溢出中断标志位 当接收到数据字节的数量大于端点缓冲器容量时，此位由硬件置位。 当 SETUP 令牌被接收时此位由硬件清零，或者通过写 1 清零。
[1]	ODRXIF	OUT 数据接收中断标志位 当成功接收到来自于 USB 主机的数据包且 ACK 信号被发送到 USB 主机时，此位由硬件置位。 当 SETUP 令牌被接收时此位由硬件清零，或者通过写 1 清零。
[0]	OTRXIF	OUT 令牌接收中断标志位 当接收到来自于 USB 主机的 OUT 令牌时，此位由硬件置位。 当 SETUP 令牌被接收时此位由硬件清零，或者通过写 1 清零。

USB 端点 0 传输计数寄存器 – USBEP0TCR

该寄存器定义了端点 0 数据传输字节数。

偏移量： 0x020
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	RXCNT							
类型 / 复位		RO	0	RO	0	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	TXCNT							
类型 / 复位		RW	0	RW	0	RW	0	RW	0

位	域	描述
[22:16]	RXCNT	接收字节数 这个位域包含了在之前的 SETUP 事务处理中被端点 0 接收的数据字节总数。
[6:0]	TXCNT	发送字节总数 这个位域包含了在下个 IN 令牌中端点 0 要发送的数据字节总数。如果此位域为 0，表示将发送 0 长度数据包。

USB 端点 0 配置寄存器 – USBEP0CFGR

该寄存器定义了端点 0 的配置选项。

偏移量： 0x024
复位值： 0x8000_0008

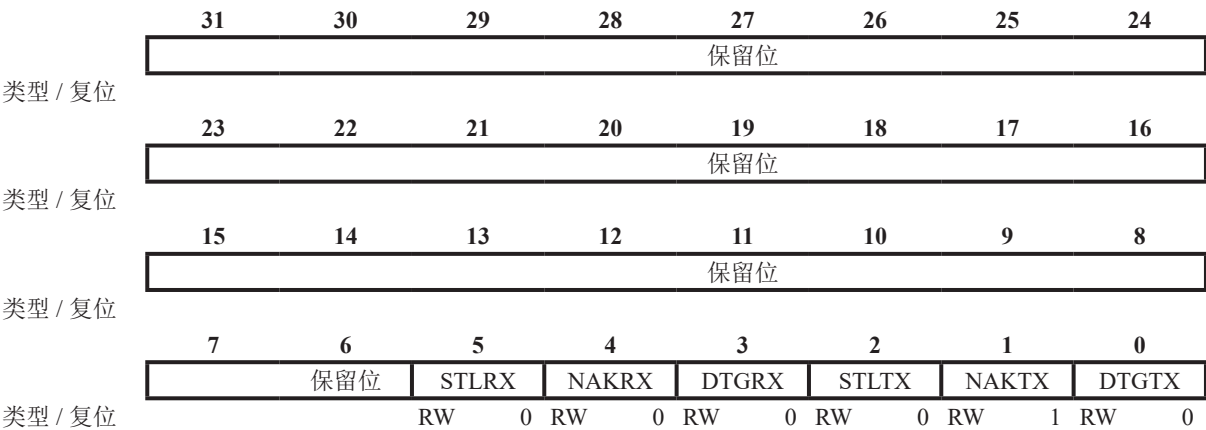
	31	30	29	28	27	26	25	24
	EPEN		保留位			EPADR		
类型 / 复位	RO	1			RO	0	RO	0
	23	22	21	20	19	18	17	16
	保留位							EPLEN
类型 / 复位								RW
	15	14	13	12	11	10	9	8
	EPLEN						EPBUFA	
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0
	EPBUFA							
类型 / 复位	RW	0	RW	0	RW	0	RW	0

位	域	描述
[31]	EPEN	端点使能控制位 此位由硬件始终置 1，使端点 0 一直使能。
[27:24]	EPADR	端点地址 该位域由硬件始终清 0。
[16:10]	EPLEN	端点缓冲器长度 该位域用来指定控制传输包的大小，在 USB 全速标准规格中，可设置为 8、16、32、或 64 个字节。
[9:0]	EPBUFA	端点缓冲器开始地址 此位域用来指定在 EP_SRAM 中的端点 0 缓冲器的起始地址。起始地址从 0x008 开始并 对齐到 4-byte 边界。 端点 0 IN 缓冲器起始地址 = EPBUFA 端点 0 OUT 缓冲器起始地址 = EPBUFA + EPLEN

USB 端点 1 ~ 3 控制和状态寄存器 – USBEPnCSR, n = 1 ~ 3

该寄存器定义了端点 1 ~ 3 控制和状态位。

偏移量: 0x028 (n = 1), 0x03C (n = 2), 0x050 (n = 3)
复位值: 0x0000_0002



位	域	描述
[5]	STLRX	接收传输的 STALL 位 当功能错误被检测到，此位由应用程序置 1。 此位可读写，仅当写 1 时状态切换。此位也可由软件对其初始化以适用于特定的应用。
[4]	NAKRX	接收传输的 NAK 位 当已接收到一个数据包且在 OUT 事务处理的握手阶段已发送一个 ACK 信号到主机后，此位由硬件电路从 0 切换到 1 从而将产生一个 NAK 信号。这意味着 USB 设备需要更多的时间来处理已经接收的数据，将暂时不能接收来自 USB 主机的数据。 此位可读写，仅当写 1 时状态切换。
[3]	DTGRX	接收传输的数据切换位 此位包含数据切换位的预期值 (0 = DATA0, 1 = DATA1)，用于下一个要接收的数据包。 当当前有效数据包被接收且 USB 设备发送给 USB 主机相应的 ACK 信号后，硬件将切换此位，设备将准备接收下一个数据包。 此位可读写，仅当写 1 时状态切换。在特定条件下，此位也可由软件对其初始化。
[2]	STLTX	发送传输的 STALL 位 当功能错误被检测到，此位由应用程序置 1。 此位可读写，仅当写 1 时状态切换。在特定情况下，此位也可由软件对其初始化。
[1]	NAKTX	发送传输的 NAK 位 当一个数据包已被发送且来自于主机 ACK 信号在 IN 事务处理的握手阶段已被接收，此位由硬件电路从 0 切换到 1 从而产生一个 NAK 信号。这意味着 USB 设备暂时不能向 USB 主机发送数据直到应用软件准备好要发送的数据为止。 此位可读写，仅当写 1 时状态切换。
[0]	DTGTX	发送传输的数据切换位 此位包含数据切换位所需要的值 (0 = DATA0, 1 = DATA1)，用于下一个要发送的数据包。 当当前数据包被 USB 设备发送，由 USB 主机发送的 ACK 信号被接收后，硬件将切换此位，并将发送下一个数据包。 此位可读写，仅当写 1 时状态切换。在特定情况下，此位也可由软件对其初始化。

USB 端点 1 ~ 3 中断使能寄存器 – USBEPnIER, n = 1 ~ 3

该寄存器定义了端点 1 ~ 3 中断使能控制位。

偏移量: 0x02C (n = 1), 0x040 (n = 2), 0x054 (n = 3)

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
UERIE	STLIE	NAKIE	IDTXIE	ITRXIE	ODOVIE	ODRXIE	OTRXIE
RW	0 RW	0 RW	0 RW	0 RW	0 RW	0 RW	0 RW

位	域	描述
[7]	UERIE	USB 错误中断使能控制位 0: 除能 1: 使能
[6]	STLIE	STALL 发送中断使能控制位 0: 除能 1: 使能
[5]	NAKIE	NAK 发送中断使能控制位 0: 除能 1: 使能
[4]	IDTXIE	IN 数据发送中断使能控制位 0: 除能 1: 使能
[3]	ITRXIE	IN 令牌接收中断使能控制位 0: 除能 1: 使能
[2]	ODOVIE	OUT 数据缓冲器溢出中断使能控制位 0: 除能 1: 使能
[1]	ODRXIE	OUT 数据接收中断使能控制位 0: 除能 1: 使能
[0]	OTRXIE	OUT 令牌接收中断使能控制位 0: 除能 1: 使能

USB 端点 1 ~ 3 中断状态寄存器 – USBEPnISR, n = 1 ~ 3

该寄存器定义了端点 1 ~ 3 点中断状态。

偏移量: 0x030 (n = 1), 0x044 (n = 2), 0x058 (n = 3)
复位值: 0x0000_0000

类型 / 复位	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
	UERIF	STLIF	NAKIF	IDTXIF	ITRXIF	ODOVIF	ODRXIF	OTRXIF
类型 / 复位	WC	0	WC	0	WC	0	WC	0

位	域	描述
[7]	UERIF	USB 错误中断标志位 当在事务处理期间有错误发生时，此位由硬件置位。通过写 1 清零。
[6]	STLIF	STALL 发送中断标志位 当 STALL 令牌被发送来响应 IN 或 OUT 令牌时，此位由硬件置位。通过写 1 清零。
[5]	NAKIF	NAK 发送中断标志位 当 NAK 令牌被发送来响应 IN 或 OUT 令牌时，此位由硬件置位。通过写 1 清零。
[4]	IDTXIF	IN 数据发送中断标志位 当数据包被成功发送给主机以作为对 IN 令牌的响应且 ACK 令牌被端点接收到时，此位由硬件置位。通过 1 时清零。
[3]	ITRXIF	IN 令牌接收中断标志位 当端点接收到来自于主机的 IN 令牌时，此位由硬件置位。通过写 1 清零。
[2]	ODOVIF	OUT 数据缓冲器溢出中断标志位 当接收到的数据字节数大于相关的端点 OUT 数据缓冲器的容量时。此位由硬件置位。通过写 1 清零。
[1]	ODRXIF	OUT 数据接收中断标志位 对于 OUT 令牌，当成功接收到主机发送的数据包和当端点 n ACK 信号被发送给主机时，此位由硬件电路置位。通过写 1 清零。
[0]	OTRXIF	OUT 令牌接收中断标志位 当端点接收到来自主机的 OUT 令牌时，此位被置位。通过写 1 清零。

USB 端点 1 ~ 3 传输计数寄存器 – USBEPnTCR, n = 1 ~ 3

该寄存器定义了端点 1 ~ 3 传输字节数。

偏移量: 0x034 (n = 1), 0x048 (n = 2), 0x05C (n = 3)

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							TCNT	
类型 / 复位								RW	0
	7	6	5	4	3	2	1	0	
	TCNT								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	域	描述
[8:0]	TCNT	传输字节数 该位域包含了在之前的 OUT 事务处理中由端点 n 接收到的数据字节数，或在下一个 IN 事务处理中由端点 n 发送的数据字节数。

USB 端点 1 ~ 3 配置寄存器 – USBEPnCFGR, n = 1 ~ 3

该寄存器定义了端点 1 ~ 3 的配置。

偏移量: 0x038 (n = 1), 0x04C (n = 2), 0x060 (n = 3)

复位值: 0x1000_03FF

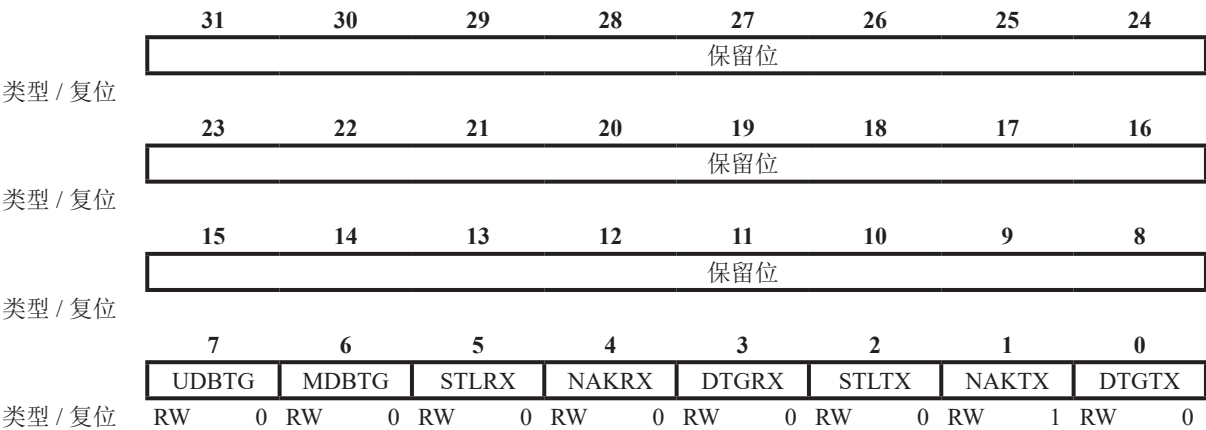
	31	30	29	28	27	26	25	24
	EPEN	保留位	EPTYPE	EPDIR	EPADR			
类型 / 复位	RW 0		RW 0	RW 1	RW 0	RW 0	RW 0	RW 0
	23	22	21	20	19	18	17	16
	保留位						EPLEN	
类型 / 复位							RW 0	
	15	14	13	12	11	10	9	8
	EPLEN						EPBUFA	
类型 / 复位	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 1	RW 1
	7	6	5	4	3	2	1	0
	EPBUFA							
类型 / 复位	RW 1	RW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

位	域	描述
[31]	EPEN	端点使能控制位 0: 除能端点 n 1: 使能端点 n
[29]	EPTYPE	端点传输类型 此位由硬件清零，用来定义端点 n 传输类型为中断或批量传输类型。
[28]	EPDIR	端点传输方向 0: OUT 1: IN
[27:24]	EPADR	端点地址 EPADR 位域由应用软件分配用来指定端点 n 地址。需注意的是，EPADR 位域不能清为 0，否则，端点将被除能。
[16:10]	EPLEN	端点缓冲器长度 此位域用来定义端点 n 数据包大小。此位域定义的值必须是 4 的倍数。此位域最多可达 64 字节，这也是 USB 全速标准规格定义的最大容载量。注意，EPLEN 值不能清除为 0，否则将除能端点。
[9:0]	EPBUFA	端点缓冲器起始地址 此位域用来定义端点 n 数据缓冲器的起始地址。在容量为 1024 字节的 EP_SRAM 中，起始地址范围是 0x008 ~ 0x3FC，且该位域值必须是 4 的倍数。

USB 端点 4 ~ 7 控制和状态寄存器 – USBEPnCSR, n = 4 ~ 7

该寄存器定义了端点 4 ~ 7 控制和状态位。

偏移量: 0x064 (n = 4), 0x078 (n = 5), 0x08C (n = 6), 0x0A0 (n = 7)
复位值: 0x0000_0002



位	域	描述
---	---	----

[7] UDBTG USB 双缓冲器切换位

双缓冲器功能使能时，UDBTG 和 MDBTG 位用来指定哪个数据缓冲器被 USB SIE 硬件访问，哪个数据缓冲器被 MCU 软件访问。在当前缓冲器工作完成时，UDBTG 位将由 SIE 硬件电路切换。此位被 SIE 切换后，硬件电路会自动发送一个 NAK 信号给 USB 主机。因此，数据传输将暂时停止直到另一个缓冲器内的数据被正确设置，在这之后 MDBTG 位通过 MCU 应用软件切换。

下表显示了双缓冲器的操作和在 IN 或 OUT 事务处理时 UDBTG 和 MDBTG 位的状态。

事务处理类型	UDBTG	MDBTG	SIE 读缓冲器	MCU 写缓冲器
IN	0	0	无 *	EP_BUF0
	0	1	EP_BUF0	EP_BUF1
	1	0	EP_BUF1	EP_BUF0
	1	1	无 *	EP_BUF1

事务处理类型	UDBTG	MDBTG	SIE 写缓冲器	MCU 读缓冲器
OUT	0	0	无 *	EP_BUF0
	0	1	EP_BUF0	EP_BUF1
	1	0	EP_BUF1	EP_BUF0
	1	1	无 *	EP_BUF1

* 表示 USB 设备通过硬件电路发送一个 NAK 信号给 USB 主机。

用于双缓冲功能的 UDBTG 和 MDBTG 位的设置步骤如下例所示：
[UDBTG, MDBTG] = [0, 0]→[0, 1]→[1, 1]→[1, 0]→[0, 0]→[0, 1]→[1, 1]→[1, 0]→...

[6] MDBTG MCU 双缓冲器切换位

如果双缓冲功能使能，MDBTG 位用来指定哪个数据缓冲器被 MCU 访问。在被 MCU 访问的当前缓冲器中的数据正确设置之后，此位可通过 MCU 应用软件改变状态来切换到另一个缓冲器。双缓冲功能操作与 UDBTG 和 MDBTG 位的关系请参考前面 UDBTG 位定义里的表格。

位	域	描述
[5]	STLRX	接收传输的 STALL 位 当功能错误被检测到，此位由应用程序置 1。 此位可读写，仅当写 1 时状态切换。在特定条件下，此位也可由软件对其初始化。
[4]	NAKRX	接收传输的 NAK 位 当已接收到一个数据包且在 OUT 事务处理的握手阶段已发送一个 ACK 信号到主机后，此位由硬件电路从 0 切换到 1 从而将产生一个 NAK 信号。这意味着 USB 设备需要更多的时间来处理已经接收的数据，将暂时不能接收来自 USB 主机的数据。如果端点被定义成等时传输类型，此位将不可用。经过完整的事务处理之后，硬件不会改变 NAKRX 位的状态。此位可读写，仅当写 1 时状态切换。
[3]	DTGRX	接收传输的数据切换位 如果端点不用于等时传输，此位可用。此位包含数据切换位的预期值 (0 = DATA0, 1 = DATA1)，用于下一个要接收的数据包。当当前有效数据包被接收且 USB 设备已发送给 USB 主机相应的 ACK 信号后，硬件将切换此位，设备将准备接收下一个数据包。如果端点设置为等时传输类型，因为无数据切换，所以此位不可用。只有 DATA0 包在正常等时传输器中可以传输。 此位可读写，仅当写 1 时状态切换。在特定条件下，此位也可由软件对其初始化。
[2]	STLTX	发送传输的 STALL 位 当功能错误被检测到，此位由应用程序置 1。 此位可读写，仅当写 1 时状态切换。在特定情况下，此位也可由软件对其初始化。
[1]	NAKTX	发送传输的 NAK 位 当一个数据包已被发送，且来自于主机的 ACK 信号在 IN 事务处理对此端点寻址的握手阶段已被接收，此位由硬件电路从 0 切换到 1 从而将产生一个 NAK 信号。这意味着 USB 设备暂时不能向 USB 主机发送数据直到应用软件准备好要发送的数据为止。如果端点定义为等时传输类型，则此位不可用。在一个完整的事务处理之后，硬件不会改变 NAKTX 位的状态。 此位可读写，仅当写 1 时状态切换。在特定情况下，此位也可由软件对其初始化。
[0]	DTGTX	发送传输的数据切换位 如果端点不用于等时传输，此位可用。此位包含数据切换位所需要的值 (0 = DATA0, 1 = DATA1)，用于下一个要发送的数据包。当当前数据包已被 USB 设备发送，且由 USB 主机发送的 ACK 信号已被接收，硬件将切换此位，并将发送下一个数据包。如果端点设置为等时传输类型，因为无数据切换，所以此位不可用。在正常等时传输中只有 DATA0 包可以传输。 此位可读写，仅当写 1 时状态切换。在特定情况下，此位也可由软件对其初始化。

USB 端点 4 ~ 7 中断使能寄存器 – USBEPnIER, n = 4 ~ 7

该寄存器定义了端点 4 ~ 7 中断使能控制位。

偏移量: 0x068 (n = 4), 0x07C (n = 5), 0x090 (n = 6), 0x0A4 (n = 7)

复位值: 0x0000_0000

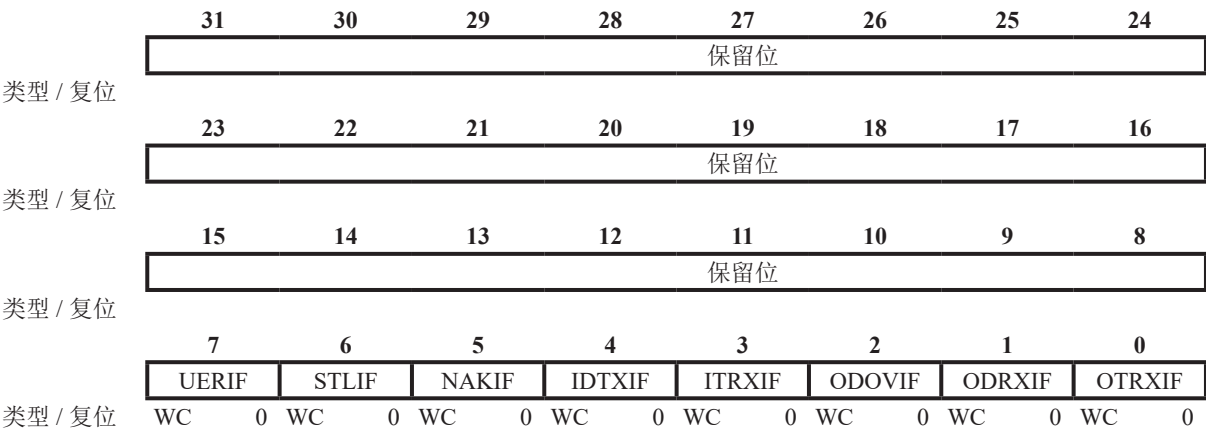
31	30	29	28	27	26	25	24
保留位							
23	22	21	20	19	18	17	16
保留位							
15	14	13	12	11	10	9	8
保留位							
7	6	5	4	3	2	1	0
UERIE	STLIE	NAKIE	IDTXIE	ITRXIE	ODOVIE	ODRXIE	OTRXIE
RW	0 RW	0 RW	0 RW	0 RW	0 RW	0 RW	0 RW

位	域	描述
[7]	UERIE	USB 错误中断使能控制位 0: 除能 1: 使能
[6]	STLIE	STALL 发送中断使能控制位 0: 除能 1: 使能
[5]	NAKIE	NAK 发送中断使能控制位 0: 除能 1: 使能
[4]	IDTXIE	IN 数据发送中断使能控制位 0: 除能 1: 使能
[3]	ITRXIE	IN 令牌接收中断使能控制位 0: 除能 1: 使能
[2]	ODOVIE	OUT 数据缓冲器溢出中断使能控制位 0: 除能 1: 使能
[1]	ODRXIE	OUT 数据接收中断使能控制位 0: 除能 1: 使能
[0]	OTRXIE	OUT 令牌接收中断使能控制位 0: 除能 1: 使能

USB 端点 4 ~ 7 中断状态寄存器 – USBEPnISR, n = 4 ~ 7

该寄存器定义了端点 4 ~ 7 端点中断状态。

偏移量: 0x06C (n = 4), 0x080 (n = 5), 0x094 (n = 6), 0x0A8 (n = 7)
复位值: 0x0000_0000



位	域	描述
[7]	UERIF	USB 错误中断标志位 当在事务处理期间有错误发生时，此位由硬件置位。 通过写 1 清零。
[6]	STLIF	STALL 发送中断标志位 当 STALL 令牌被发送来响应 IN 或 OUT 令牌时，此位由硬件置位。 通过写 1 清零。
[5]	NAKIF	NAK 发送中断标志位 当 NAK 令牌被发送来响应 IN 或 OUT 令牌时，此位由硬件置位。 通过写 1 清零。
[4]	IDTXIF	IN 数据发送中断标志位 当数据包已被成功发送给主机作为对 IN 令牌的响应且 ACK 令牌被端点接收到时，此位由硬件置位。通过写 1 清零。
[3]	ITRXIF	IN 令牌接收中断标志位 当端点接收到来自主机的 IN 令牌时，此位由硬件置位。 通过写 1 清零。
[2]	ODOVIF	OUT 数据缓冲器溢出中断标志位 当接收到的数据字节数大于相关的端点 OUT 数据缓冲器的大小时，此位由硬件置位。 通过写 1 清零。
[1]	ODRXIF	OUT 数据接收中断标志位 对于 OUT 令牌，当成功接收到主机发送的数据包且 ACK 信号已发送给主机时，此位由硬件电路置位。通过写 1 清零。
[0]	OTRXIF	OUT 令牌接收中断标志位 当端点接收到来自主机的 OUT 令牌时，此位由硬件置位。 通过写 1 清零。

USB 端点 4 ~ 7 传输计数寄存器 – USBEPnTCR, n = 4 ~ 7

该寄存器定义了端点 4 ~ 7 端点传输字节数。

偏移量: 0x070 (n = 4), 0x084 (n = 5), 0x098 (n = 6), 0x0AC (n = 7)
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24
	保留位						TCNT1	
类型 / 复位							RW	0
	23	22	21	20	19	18	17	16
	TCNT1							
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	15	14	13	12	11	10	9	8
	保留位						TCNT0	
类型 / 复位							RW	0
	7	6	5	4	3	2	1	0
	TCNT0							
类型 / 复位	RW	0	RW	0	RW	0	RW	0

位	域	描述
[25:16]	TCNT1	缓冲器 1 传输字节数 该位域包含了在之前的 OUT 事务处理中由端点 n 缓冲器 1 接收到的数据字节数，或在下一个 IN 事务处理中由端点 n 缓冲器 1 要发送的数据字节数。
[9:0]	TCNT0	缓冲器 0 传输字节数 该位域包含了在之前的 OUT 事务处理中由端点 n 缓冲器 0 接收到的数据字节数，或在下一个 IN 事务处理中由端点 n 缓冲器 0 要发送的数据字节数。当端点配置为单缓冲传输类型时，仅 TCNT0 位域用于端点数据传输计数。

USB 端点 4 ~ 7 配置寄存器 – USBEPnCFGR, n = 4 ~ 7

该寄存器定义了端点 4 ~ 7 端点的配置。

偏移量: 0x074 (n = 4), 0x088 (n = 5), 0x09C (n = 6), 0x0B0 (n = 7)
复位值: 0x1000_03FF

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

位	域	描述
[31]	EPEN	端点使能控制位 0: 除能端点 n 1: 使能端点 n
[29]	EPTYPE	端点传输类型 0: 中断或批量传输类型 1: 等时传输类型
[28]	EPDIR	端点传输方向 0: OUT 1: IN
[27:24]	EPADR	端点地址 EPADR 位域由应用软件分配用来指定端点 n 的地址。需注意的是，EPADR 位域不能设为 0，否则，端点 n 将被除能。
[23]	SDBS	单缓冲或双缓冲选择 0: 单缓冲 1: 双缓冲 如果 SDBS 位被置为 1，端点缓冲器大小是 EPLEN 值的两倍： – 端点缓冲器 0 起始地址是 EPBUFA – 端点缓冲器 1 起始地址是 (EPBUFA + EPLEN)
[19:10]	EPLEN	端点缓冲器长度 此位域用来定义端点 n 数据包大小。此位域定义的值必须是 4 的倍数。注意，如果 EPLEN 值为 0，则端点将除能。
[9:0]	EPBUFA	端点缓冲器起始地址 此位域用来定义端点 n 数据缓冲器的起始地址。在容量为 1024 字节的 EP_SRAM 中，起始地址范围是 0x008 ~ 0x3FC，且该位域值必须是 4 的倍数。

25 外设直接存储器访问 (PDMA)

简介

外设直接存储器访问电路 PDMA 提供了 6 个单向通道用于专用外设执行“外设 → 存储器”和“存储器 → 外设”的数据传输。应用程序同时也支持和要求如 FLASH→SRAM 或 SRAM→SRAM 类型的“存储器 → 存储器”数据传输。每个 PDMA 通道配置都是独立的。PDMA 通道传输分为多个块处理，且每个块的大小等于块的长度乘以数据的宽度。

特性

- 6 个单向 PDMA 通道
- 支持存储器 → 外设、外设 → 存储器和存储器 → 存储器类型的数据传输
- 8-bit、16-bit 和 32-bit 宽度数据传输
- 带有可配置通道优先级的软硬件需求的数据传输
- 线性地址、环形地址和固定地址模式
- 4 个传输事件标志：传输完成、半传输、块结束和传输错误
- 自动重载功能

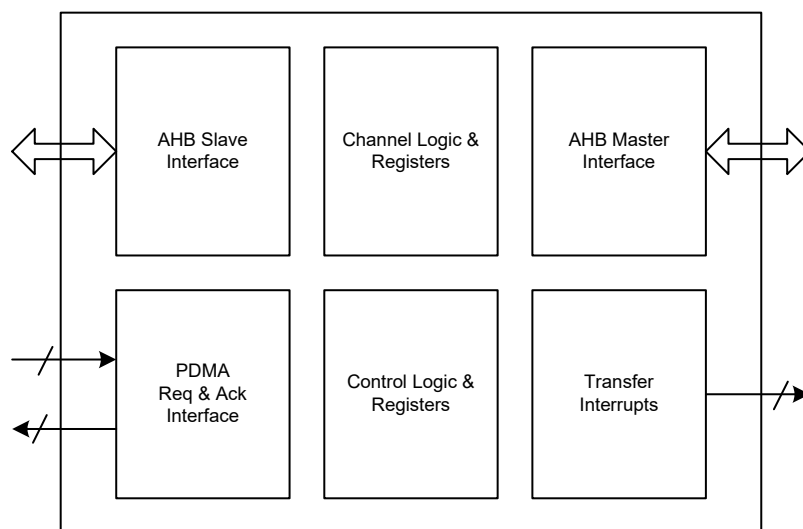


图 171. PDMA 方框图

功能描述

AHB 主机

PDMA 是一个 AHB 主机，通过总线矩阵与其它 AHB 外设相连，如 FLASH 存储器、SRAM 存储器和 AHB-to-APB 总线桥。CPU 和 PDMA 通过总线矩阵可以同时访问不同的 AHB 从机。

PDMA 通道

有 6 个单向 PDMA 通道用于外设和存储器之间的数据传输。每个 PDMA 通道的配置和操作都是独立的。对于一个双向数据传输应用，需要两个 PDMA 通道。每个 PDMA 使用相同的寄存器来处理不同外设的数据传输请求。因此，一个 PDMA 通道在同一时间内只能服务一个外设。PDMA 通道的相关寄存器被限制只能由 32-bit 操作访问，否则，将发生系统硬故障。

PDMA 请求映射

外设 (ADC、SPI、I2C 和 USART 等) 的请求简单地进行逻辑与运算后进入 PDMA。这就意味着同一时间只允许一个请求进入 PDMA 通道。请参考下图 PDMA 请求映射架构。详细的外设 IP 请求映射列表如下表所示。外设的 DMA 请求可通过设置相应外设寄存器中的 DMA 控制位被单独开启或关闭。

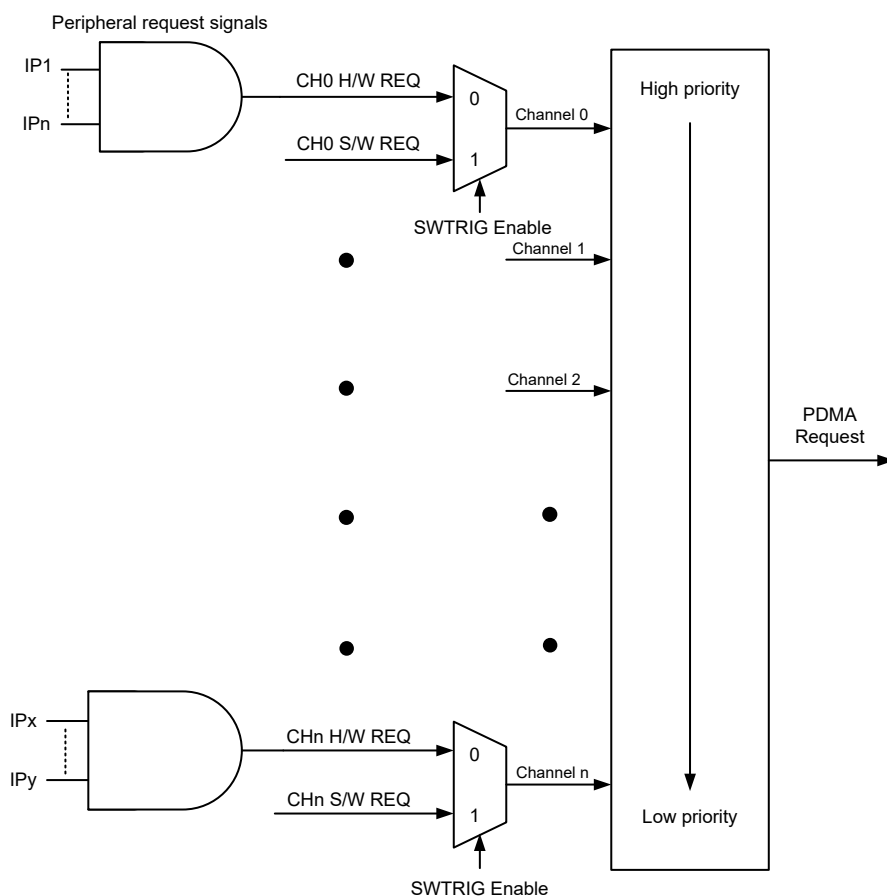


图 172. PDMA 请求映射架构

表 68. PDMA 通道配置

IP (x = 0, 1)	PDMA 通道编号					
	CH0	CH1	CH2	CH3	CH4	CH5
ADC	ADC					
SPIx	SPI0_RX	SPI0_TX			SPI1_RX	SPI1_TX
USART	USR0_RX	USR0_TX				
UARTx			UR0_RX	UR0_TX	UR1_RX	UR1_TX
SCI					SCI_RX	SCI_TX
I2Cx			I2C0_RX	I2C1_RX	I2C0_TX	I2C1_TX
GPTM	GT_CH1 GT_CH3	GT_CH2 GT_UEV	GT_CH0 GT_TRIG			
PWM				PWM_CH1 PWM_CH3	PWM_CH2 PWM_UEV	PWM_CH0 PWM_TRIG
AES					AES_OUT	AES_IN

通道传输

PDMA 通道传输被分成多个块处理，在每个块处理结束后，PDMA 仲裁发生。虽然这些通道传输都可以被激活，但一次只能通过总线进行一个块处理传输。通道传输顺序取决于每个 PDMA 通道的优先级设置。总的传输大小由块处理数和块大小计算而得。块大小等于块长度和数据位宽度的乘积。对于一个有效的传输，建议块长度为 4 的倍数。

总传输数据大小可由下列等式算出：

PDMA 通道总传输数据大小 = 块处理数 × (块长度 × 数据宽度)

通道优先级

PDMA 优先级分为四个等级，即非常高、高、中、低，可由应用软件设置。PDMA 提供了两种方式来决定通道优先级。一种由应用软件配置选项进行设置，另一种由固定硬件编号决定。在请求 PDMA 提供数据传输服务时，PDMA 仲裁处理器先检查软件配置的通道优先级等级。如果多个通道有相同的优先级，则经过仲裁后，编号较小的通道比编号大的通道具有高优先级。

注意，最高优先级通道不会一直占用 PDMA 服务。当其它低优先级通道请求挂起时，最高优先级通道在完成一个块处理后，将被跳过一个块处理的时间，执行由第二优先级通道请求的块处理。在第二优先级通道完成当前块处理后，第二优先级通道将被排除在外，PDMA 仲裁处理器将重新判断除第二优先级通道外的其它请求通道的优先级。因此，较高优先级通道的块数据处理将被服务，且在当前块处理完成时，此通道将从优先级仲裁中排除。PDMA 将用上述的方法持续传输数据直到所有请求的通道数据传输完成。下图的例子显示了 PDMA 通道仲裁和安排。

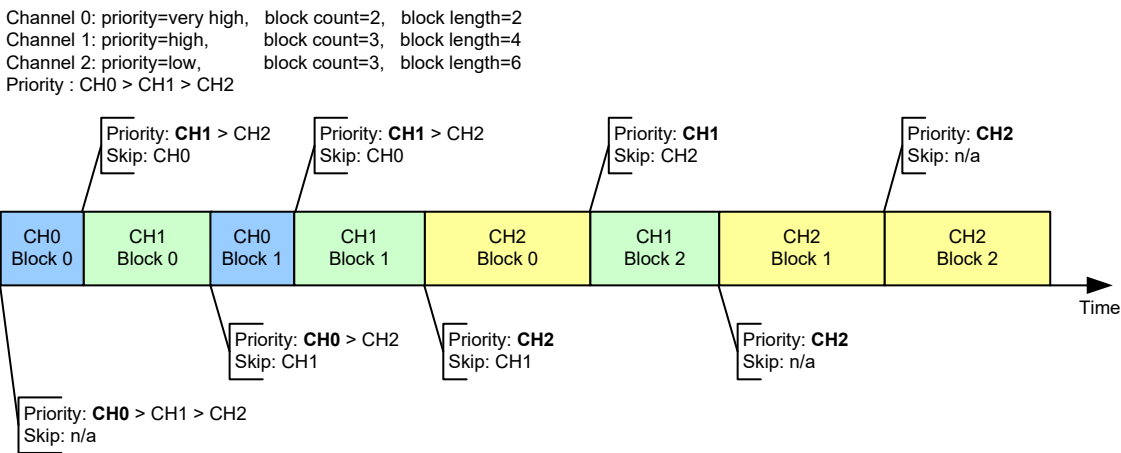


图 173. PDMA 通道仲裁和安排范例

传输请求

对于“外设 → 存储器”或“存储器 → 外设”传输，一个外设硬件请求将触发一个专用 PDMA 通道的块处理。但当软件请求发生时，相关专用 PDMA 通道的一个完整数据传输会被触发。应“存储器 → 存储器”数据复制应用的软件需求，建议 PDMA 通道被配置为较低的优先级和较小的块长度。

地址模式

PDMA 提供了三种地址模式，即线性地址、环形地址和固定地址模式。这些不同的地址模式用来支持不同种类的源地址和目标地址配置。下表显示了详细的地址模式组合。

表 69. PDMA 地址模式

源地址模式	目标地址模式
线性递增 / 递减地址	线性递增 / 递减地址
线性递增 / 递减地址	环形递增 / 递减地址
线性递增 / 递减地址	固定地址
环形递增 / 递减地址	线性递增 / 递减地址
环形递增 / 递减地址	环形递增 / 递减地址
固定地址	线性递增 / 递减地址
固定地址	固定地址

线性地址模式

在数据被传输后，当前地址将以 1、2 或 4 为间隔递增或递减，取决于数据位的宽度设置。

环形地址模式

在数据被传输后，当前地址将以 1、2 或 4 为间隔递增或递减，取决于数据位的宽度设置。当块处理完成，所设置的起始地址将加载到当前地址中。

固定地址模式

数据传输后，当前地址仍然不变。

自动重载

当 PDMA 通道 n 控制寄存器 PDMACHnCR 中的自动重载控制位 AUTORLn 被置位时, 在当前 PDMA 通道数据传输完全完成时, 通道 n 当前地址和通道 n 当前传输大小将分别自动载入对应的初始值。通道 n 仍然有效且下一个相关的 PDMA 请求在不需要软件重新配置的情况下可以被服务。

传输中断

每个 PDMA 通道都有 5 种传输事件, 当传输事件发生时可以产生相应中断。这些传输事件分别为块处理结束 (BE)、半传输 (HT)、传输完成 (TC)、传输错误 (TE) 和总传输事件 (GE)。在 PDMA 中断使能寄存器 PDMAIER 中设置相关控制位可使能相关中断事件。如果 BE、HT、TC 或 TE 四个中断事件中有任意一个发生时, 将产生总中断事件 GE。清零 BE、HT、TC 或 TE 事件标志位也会清零 GE 标志位。清零 GE 标志位将自动清零所有其它事件标志位。当 PDMA 访问一个系统预留地址空间或 PDMA 接收一个请求但相应传输大小设置为 0 时, 将产生 TE 中断事件。

寄存器列表

下表显示了 PDMA 寄存器和复位值。

表 70. PDMA 寄存器列表

寄存器	偏移量	描述	复位值
PDMA 通道 0 寄存器			
PDMACH0CR	0x000	PDMA 通道 0 控制寄存器	0x0000_0000
PDMACH0SADR	0x004	PDMA 通道 0 源地址寄存器	0x0000_0000
PDMACH0DADR	0x008	PDMA 通道 0 目标地址寄存器	0x0000_0000
PDMACH0TSR	0x010	PDMA 通道 0 传输大小寄存器	0x0000_0000
PDMACH0CTSR	0x014	PDMA 通道 0 当前传输大小寄存器	0x0000_0000
PDMA 通道 1 寄存器			
PDMACH1CR	0x018	PDMA 通道 1 控制寄存器	0x0000_0000
PDMACH1SADR	0x01C	PDMA 通道 1 源地址寄存器	0x0000_0000
PDMACH1DADR	0x020	PDMA 通道 1 目标地址寄存器	0x0000_0000
PDMACH1TSR	0x028	PDMA 通道 1 传输大小寄存器	0x0000_0000
PDMACH1CTSR	0x02C	PDMA 通道 1 当前传输大小寄存器	0x0000_0000
PDMA 通道 2 寄存器			
PDMACH2CR	0x030	PDMA 通道 2 控制寄存器	0x0000_0000
PDMACH2SADR	0x034	PDMA 通道 2 源地址寄存器	0x0000_0000
PDMACH2DADR	0x038	PDMA 通道 2 目标地址寄存器	0x0000_0000
PDMACH2TSR	0x040	PDMA 通道 2 传输大小寄存器	0x0000_0000
PDMACH2CTSR	0x044	PDMA 通道 2 当前传输大小寄存器	0x0000_0000
PDMA 通道 3 寄存器			
PDMACH3CR	0x048	PDMA 通道 3 控制寄存器	0x0000_0000
PDMACH3SADR	0x04C	PDMA 通道 3 源地址寄存器	0x0000_0000
PDMACH3DADR	0x050	PDMA 通道 3 目标地址寄存器	0x0000_0000
PDMACH3TSR	0x058	PDMA 通道 3 传输大小寄存器	0x0000_0000
PDMACH3CTSR	0x05C	PDMA 通道 3 当前传输大小寄存器	0x0000_0000
PDMA 通道 4 寄存器			
PDMACH4CR	0x060	PDMA 通道 4 控制寄存器	0x0000_0000

寄存器	偏移量	描述	复位值
PDMACH4SADR	0x064	PDMA 通道 4 源地址寄存器	0x0000_0000
PDMACH4DADR	0x068	PDMA 通道 4 目标地址寄存器	0x0000_0000
PDMACH4TSR	0x070	PDMA 通道 4 传输大小寄存器	0x0000_0000
PDMACH4CTSR	0x074	PDMA 通道 4 当前传输大小寄存器	0x0000_0000
PDMA 通道 5 寄存器			
PDMACH5CR	0x078	PDMA 通道 5 控制寄存器	0x0000_0000
PDMACH5SADR	0x07C	PDMA 通道 5 源地址寄存器	0x0000_0000
PDMACH5DADR	0x080	PDMA 通道 5 目标地址寄存器	0x0000_0000
PDMACH5TSR	0x088	PDMA 通道 5 传输大小寄存器	0x0000_0000
PDMACH5CTSR	0x08C	PDMA 通道 5 当前传输大小寄存器	0x0000_0000
PDMA 总寄存器			
PDMAISR	0x120	PDMA 中断状态寄存器	0x0000_0000
PDMAISCR	0x128	PDMA 中断状态清除寄存器	0x0000_0000
PDMAIER	0x130	PDMA 中断使能寄存器	0x0000_0000

寄存器描述

PDMA 通道 n 控制寄存器 – PDMACHnCR, n = 0 ~ 5

该寄存器定义了 PDMA 通道 n 数据传输配置。

偏移量: 0x000 (0), 0x018 (1), 0x030 (2), 0x048 (3), 0x060 (4), 0x078 (5)
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				AUTORLn	FIXAENn	CHnPRI	
类型 / 复位				RW	0 RW	0 RW	0 RW
7	6	5	4	3	2	1	0
SRCAMODn	SRCAINCn	DSTAMODn	DSTAINCn	DWIDTHn		SWTRIGn	CHnEN
RW	0 RW	0 RW	0 RW	0 RW	0 RW	0 RW	0

位	域	描述
[11]	AUTORLn	通道 n 自动重载使能控制位 0: 除能自动重载功能 1: 使能自动重载功能 该位设为 1 将使能自动重载功能, 当传输完成, 通道 n 当前地址和当前传输大小将重新载入对应的起始值且 PDMA 通道 n 仍有效。如果该位设为 0, 在传输完成后, 通道 n 当前地址和当前传输大小的值将保持不变且 PDMA 通道 n 将除能。

位	域	描述
[10]	FIXAENn	通道 n 固定地址使能控制位 0: 在环形地址模式下除能固定地址功能 1: 在环形地址模式下使能固定地址功能 注意, 该位仅当源地址或目标地址处于环形地址模式下时才有效。例如, 源地址模式设置为线性地址模式, 目标地址模式设置为环形模式。如果该位置位使能固定地址功能, 则源地址模式仍会处于线性地址模式, 但是目标地址将会处于固定地址模式而不是环形地址模式。
[9:8]	CHnPRI	通道 n 优先级 00: 低 01: 中 10: 高 11: 非常高 CHnPRI 位域通过应用程序来配置通道的优先级。如果有多个通道软件优先等级相同, 则更小编号的通道将在仲裁后优先传输一个数据块。
[7]	SRCAMODn	通道 n 源地址模式选择位 0: 线性地址模式 1: 环形地址模式 在线性地址模式下, 在一个完整的数据传输过程中当前源地址的值根据 SRCAINCn 位的值可以递增或递减。在环形地址模式下, 当前源地址的值在数据传输过程中也是根据 SRCAINCn 位的值来决定是递增或递减, 当一块的数据传输完成时 PDMACHnSADR 寄存器的低 16-bit 的值将会被加载为当前源地址的值。
[6]	SRCAINCn	通道 n 源地址增减控制位 0: 递增 1: 递减 该位用于定义在线性地址模式下的一个完整的数据传输过程中或在环形地址模式下的一个数据块传输过程中当前源地址是递增还是递减。
[5]	DSTAMODn	通道 n 目标地址模式选择位 0: 线性地址模式 1: 环形地址模式 在线性地址模式下, 当前目标地址的值在一个完整的数据传输过程中根据 DSTAINCn 位的值可以递增或递减。在环形地址模式下, 当前地址在一个数据块传输过程中也是根据 DSTAINCn 位的值来决定是递增或递减, 当一块的数据传输完成时 PDMACHnDADR 寄存器的低 16-bit 的值将会被加载为当前目标地址值。
[4]	DSTAINCn	通道 n 目标地址增减控制位 0: 递增 1: 递减 该位用于定义在线性地址模式下的一个完整的数据传输过程中或环形地址模式下的一块数据传输过程中当前目标地址是递增还是递减。
[3:2]	DWIDTHn	数据位宽度选择位 00: 8-bit 01: 16-bit 10: 32-bit 11: 保留 该位用来选择 PDMA 通道 n 的数据位宽度。
[1]	SWTRIGn	软件触发控制位 0: 无操作 1: 软件触发传输请求 置位该位将会在 PDMA 通道 n 产生一个“存储器 → 存储器”软件传输请求。当传输完成该位会自动清零。

位	域	描述
[0]	CHnEN	通道 n 使能控制位 0: 除能 PDMA 通道 n 1: 使能 PDMA 通道 n 置位该位将会在 PDMA 通道 n 上使能一个软件或硬件传输请求。当传输完成且自动重载功能除能时，该位将自动由硬件清零。然而，如果 AUTORLn 位设为 1，使能自动重载功能，在传输完成后该位将保持高以使能 PDMA 通道 n 功能用于下一个传输请求而不会由硬件自动清零。

PDMA 通道 n 源地址寄存器 – PDMACHnSADR, n = 0 ~ 5

该寄存器定义了 PDMA 通道 n 的源地址。

偏移量:	0x004 (0), 0x01C (1), 0x034 (2), 0x04C (3), 0x064 (4), 0x07C (5)
复位值:	0x0000_0000

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

位	域	描述
[31:0]	SADRn	通道 n 源地址 该寄存器用于定义 PDMA 通道 n 的 32-bit 源地址。

PDMA 通道 n 目标地址寄存器 – PDMACHnDADR, n = 0 ~ 5

该寄存器定义了 PDMA 通道 n 的目标地址。

偏移量: 0x008 (0), 0x020 (1), 0x038 (2), 0x050 (3), 0x068 (4), 0x080 (5)

复位值: 0x0000_0000

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

位	域	描述
[31:0]	DADRn	通道 n 目标地址 该寄存器用于定义 PDMA 通道 n 的 32-bit 目标地址。

PDMA 通道 n 传输大小寄存器 – PDMACHnTSR, n = 0 ~ 5

该寄存器用于定义块处理的数量和长度。

偏移量: 0x010 (0), 0x028 (1), 0x040 (2), 0x058 (3), 0x070 (4), 0x088 (5)
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
	BLKCNTn								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW
	23	22	21	20	19	18	17	16	
	BLKCNTn								
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	
	BLKLENn								
类型 / 复位	RW	0	RW	0	RW	0	RW	0	RW

位	域	描述
[31:16]	BLKCNTn	通道 n 块处理数量 BLKCNTn 代表通道 n 完整传输的块处理的数量。一个完整传输的容量是由 BLKCNTn × BLKLENn 的值计算得到。BLKCNTn 最大值为 65535。
[7:0]	BLKLENn	通道 n 块长度 BLKLENn 代表数据块的长度。数据宽度由 PDMACHnCR 寄存器中的 DWIDTHn 位域来定义。BLKLENn 最大值为 255。

PDMA 通道 n 当前传输大小寄存器 – PDMACHnCTSR, n = 0 ~ 5

该寄存器用于定义当前块处理数量。

偏移量: 0x014 (0), 0x02C (1), 0x044 (2), 0x05C (3), 0x074 (4), 0x08C (5)
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24								
	CBLKCNTn															
类型 / 复位	RO	0	RO	0	RO	0	RO	0	RO	0	RO	0	RO	0	RO	0
	23	22	21	20	19	18	17	16								
	CBLKCNTn															
类型 / 复位	RO	0	RO	0	RO	0	RO	0	RO	0	RO	0	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															
类型 / 复位																

位	域	描述
[31:16]	CBLKCNTn	通道 n 当前块数量 CBLKCNTn 是一个 16-bit 的只读值，它定义了待传输的数据块的数量。一个数据块传输完成后，CBLKCNTn 值将会减 1。给 PDMACHnTSR 寄存器中的 BLKCNTn 写一个新值将会更新 CBLKCNTn 的值。

PDMA 中断状态寄存器 – PDMAISR

该寄存器用于定义 PDMA 通道 0 ~ 5 相关中断状态。

偏移量: 0x120

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24	
	保留位		TEISTA5	TCISTA5	HTISTA5	BEISTA5	GEISTA5	TEISTA4	
类型 / 复位			RO	0	RO	0	RO	0	RO
	23	22	21	20	19	18	17	16	
	TCISTA4	HTISTA4	BEISTA4	GEISTA4	TEISTA3	TCISTA3	HTISTA3	BEISTA3	
类型 / 复位	RO	0	RO	0	RO	0	RO	0	RO
	15	14	13	12	11	10	9	8	
	GEISTA3	TEISTA2	TCISTA2	HTISTA2	BEISTA2	GEISTA2	TEISTA1	TCISTA1	
类型 / 复位	RO	0	RO	0	RO	0	RO	0	RO
	7	6	5	4	3	2	1	0	
	HTISTA1	BEISTA1	GEISTA1	TEISTA0	TCISTA0	HTISTA0	BEISTA0	GEISTA0	
类型 / 复位	RO	0	RO	0	RO	0	RO	0	RO

位	域	描述
[29], [24], [19], [14], [9], [4]	TEISTAn	通道 n 传输错误中断状态位 (n = 0 ~ 5) 0: 传输错误未发生 1: 传输错误发生 该位由硬件置位, 写 1 到 PDMAISR 寄存器中的相关中断状态清零位将会清零该位。 当 PDMA 访问系统预留地址空间或 PDMA 接收到一个请求但是对应的传输容量为 0 时, 将会发生传输错误。
[28], [23], [18], [13], [8], [3]	TCISTAn	通道 n 传输完成中断状态位 (n = 0 ~ 5) 0: 传输完成未发生 1: 传输完成发生 该位由硬件置位, 写 1 到 PDMAISR 寄存器的相关中断状态清零位将会清零该位。 当 PDMA 完成数据传输任务时将发生传输完成事件。
[27], [22], [17], [12], [7], [2]	HTISTAn	通道 n 半传输中断状态位 (n = 0 ~ 5) 0: 半传输事件未发生 1: 半传输事件发生 该位由硬件置位, 写 1 到 PDMAISR 寄存器的相关中断状态清零位将会清零该位。 当 PDMA 完成一半的数据传输任务时将发生半传输完成事件。
[26], [21], [16], [11], [6], [1]	BEISTAn	通道 n 块处理结束中断状态位 (n = 0 ~ 5) 0: 块处理结束事件未发生 1: 块处理结束事件发生 该位由硬件置位, 写 1 到 PDMAISR 寄存器的相关中断状态清零位将会清零该位。 当 PDMA 完成数据块处理任务时将发生块处理结束事件。
[25], [20], [15], [10], [5], [0]	GEISTAn	通道 n 总传输中断状态位 (n = 0 ~ 5) 0: TE、TC、HT 或 BE 事件未发生 1: TE、TC、HT 或 BE 事件发生 该位由硬件置位, 写 1 到 PDMAISR 寄存器的相关中断状态清零位 GEICLRn 将会清零该位。如果 BE、HT、TC 或 TE 中的任意一个事件发生将发生一个总传输事件。清零 BE、HT、TC 或 TE 中的任意一个事件中断标志位会将 GE 中断标志位清零。注意, 如果写 1 到 PDMAISR 寄存器中的 GEICLRn 位, GE 中断状态标志位和 BE、HT、TC 或 TE 事件中断标志位都将清零。

PDMA 中断状态清除寄存器 – PDMAISR

该寄存器用来将 PDMAISR 寄存器中的相关中断状态位清零。

偏移量： 0x128

复位值： 0x0000_0000

	31	30	29	28	27	26	25	24
	保留位		TEICLR5	TCICLR5	HTICLR5	BEICLR5	GEICLR5	TEICLR4
类型 / 复位			WC	0	WC	0	WC	0
	23	22	21	20	19	18	17	16
	TCICLR4	HTICLR4	BEICLR4	GEICLR4	TEICLR3	TCICLR3	HTICLR3	BEICLR3
类型 / 复位	WC	0	WC	0	WC	0	WC	0
	15	14	13	12	11	10	9	8
	GEICLR3	TEICLR2	TCICLR2	HTICLR2	BEICLR2	GEICLR2	TEICLR1	TCICLR1
类型 / 复位	WC	0	WC	0	WC	0	WC	0
	7	6	5	4	3	2	1	0
	HTICLR1	BEICLR1	GEICLR1	TEICLR0	TCICLR0	HTICLR0	BEICLR0	GEICLR0
类型 / 复位	WC	0	WC	0	WC	0	WC	0

位	域	描述
[29], [24], [19], [14], [9], [4]	TEICLRn	通道 n 传输错误中断状态清除 (n = 0 ~ 5) 0: 无操作 1: 将 PDMAISR 寄存器中对应的 TEISTAn 位清零 写 1 到 TEICLRn 位会将 PDMAISR 寄存器中的 TEISTAn 位清零。写 1 之后该位会自动清零。
[28], [23], [18], [13], [8], [3]	TCICLRn	通道 n 传输完成中断状态清除 (n = 0 ~ 5) 0: 无操作 1: 将 PDMAISR 寄存器中对应的 TCISTAn 位清零 写 1 到 TCICLRn 位会将 PDMAISR 寄存器中的 TCISTAn 位清零。写 1 之后该位会自动清零。
[27], [22], [17], [12], [7], [2]	HTRICLRn	通道 n 半传输中断状态清除 (n = 0 ~ 5) 0: 无操作 1: 将 PDMAISR 寄存器中对应的 HTISTAn 位清零 写 1 到 HTRICLRn 位会将 PDMAISR 寄存器中的 HTISTAn 位清零。写 1 之后该位会自动清零。
[26], [21], [16], [11], [6], [1]	BEICLRn	通道 n 块处理结束中断状态清除 (n = 0 ~ 5) 0: 无操作 1: 将 PDMAISR 寄存器中对应的 BEISTAn 位清零 写 1 到 BEICLRn 位会将 PDMAISR 寄存器中的 BEISTAn 位清零。写 1 之后该位会自动清零。
[25], [20], [15], [10], [5], [0]	GEICLRn	通道 n 总传输事件中断状态清除 (n = 0 ~ 5) 0: 无操作 1: 将 PDMAISR 寄存器中对应的 TEISTAn、TCISTAn、HTISTAn、BEISTAn 和 GEISTAn 位清零 写 1 到 GEICLRn 位会将 PDMAISR 寄存器中的 GEISTAn、TEISTAn、TCISTAn、HTISTAn 和 BEISTAn 位清零。写 1 之后该位会自动清零。

PDMA 中断使能寄存器 – PDMAIER

该寄存器用于使能或除能 PDMA 通道 0 ~ 5 相关中断。

偏移量: 0x130

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24
	保留位		TEIE5	TCIE5	HTIE5	BEIE5	GEIE5	TEIE4
类型 / 复位			RW	0	RW	0	RW	0
	23	22	21	20	19	18	17	16
	TCIE4	HTIE4	BEIE4	GEIE4	TEIE3	TCIE3	HTIE3	BEIE3
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	15	14	13	12	11	10	9	8
	GEIE3	TEIE2	TCIE2	HTIE2	BEIE2	GEIE2	TEIE1	TCIE1
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0
	HTIE1	BEIE1	GEIE1	TEIE0	TCIE0	HTIE0	BEIE0	GEIE0
类型 / 复位	RW	0	RW	0	RW	0	RW	0

位	域	描述
[29], [24], [19], [14], [9], [4]	TEIE _n	通道 n 传输错误中断使能控制位 (n = 0 ~ 5) 0: 传输错误中断除能 1: 传输错误中断使能 该位通过软件置位和清零。
[28], [23], [18], [13], [8], [3]	TCIE _n	通道 n 传输完成中断使能控制位 (n = 0 ~ 5) 0: 传输完成中断除能 1: 传输完成中断使能 该位通过软件置位和清零。
[27], [22], [17], [12], [7], [2]	HTIE _n	通道 n 半传输中断使能控制位 (n = 0 ~ 5) 0: 半传输中断除能 1: 半传输中断使能 该位通过软件置位和清零。
[26], [21], [16], [11], [6], [1]	BEIE _n	通道 n 块处理结束中断使能控制位 (n = 0 ~ 5) 0: 块处理结束中断除能 1: 块处理结束中断使能 该位通过软件置位和清零。
[25], [20], [15], [10], [5], [0]	GEIE _n	通道 n 总传输事件中断使能控制位 (n = 0 ~ 5) 0: 总传输事件中断除能 1: 总传输事件中断使能 该位通过软件置位和清零。

26 外部总线接口 (EBI)

简介

外部总线接口可用于访问外部并行接口设备，如 SRAM、Flash 和 LCD 模块。该接口存储映射于 CPU 内部地址总线。数据和地址线可多路复用，减少了连接外部设备所需引脚数量。总线读 / 写时序可调整，以符合外部设备时序规格。注意，该接口只支持异步 8-bit 或 16-bit 总线接口。

特性

- 可编程接口，适用于多种存储器类型
 - 异步静态随机访问存储器 – SRAM
 - 只读存储器 – ROM
 - NOR Flash 存储器
 - 8-bit 或 16-bit 并行总线 CPU 接口设备
- 将 AHB 事务转换为适当的外部设备协议
- 4 个存储器 Bank 且每个存储器 Bank 有独立的片选控制
- 可编程时序，支持多种设备
 - 可编程等待状态
 - 可编程总线周转周期
 - 可编程输出使能以及写使能周期延长
 - 高有效或低有效接口控制信号设置
- 当 AHB 事务宽度和外部设备宽度不同时，支持自动转换
- 写缓冲器可减少因 AHB 写突发事件而停滞的状况
- 支持复用和非复用的地址线和数据线配置
 - 多达 21 条地址线
 - 8-bit 或 16-bit 总线宽度

功能描述

EBI 模块方框图如下图所示。EBI 允许内部 CPU 和其它总线矩阵主机外设访问外部存储器或设备。EBI 可自动将 AHB 事务转换为外部设备协议。特别是当选择的外部存储器为 16-bit 或 8-bit 宽度时, AHB 的 32-bit 事务将自动分割成连续的 16-bit 或 8-bit 事务进行访问。

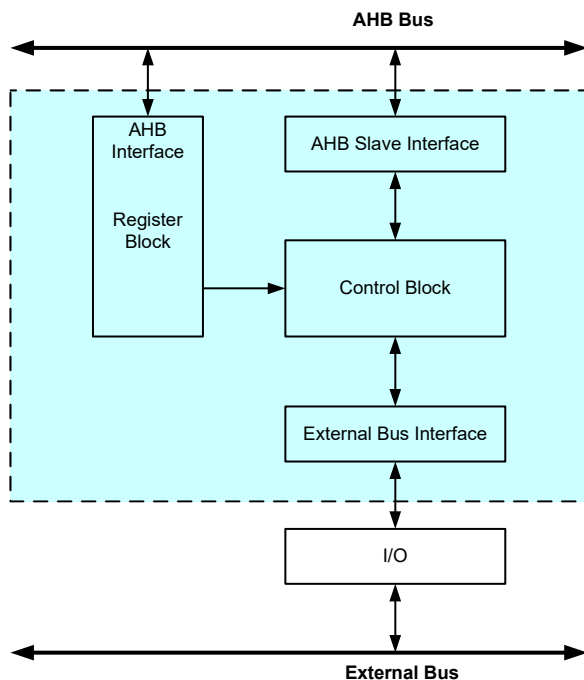


图 174. EBI 方框图

EBI 支持复用和非复用寻址模式。非复用寻址模式工作效率更高且速度更快, 但引脚需求量较大。复用寻址模式工作速度较慢, 需要一个外部地址锁存设备, 但引脚需求量较小。16 个 EBI_AD 引脚的功能取决于使用何种复用寻址模式。在复用模式下, 这些引脚既可以用作地址线也可用作数据线。在非复用 8-bit 地址模式下, 这 16 个 EBI_AD 引脚包含地址线和数据线。在复用寻址模式下, 如果需要更多地址位和数据位时, 借助一个外部锁存器仅使用 16 个 EBI_AD 引脚便可实现多达 20-bit 的地址位或 16-bit 的数据位。此外, 无论在何种寻址模式下, 有多达 21 条非复用地址线在 EBI_A 上可能用于地址连接。所有支持的模式的细节操作将在接下来的部分介绍。AHB 时钟 (HCLK) 为 EBI 参考时钟。

非复用 8-bit 数据 8-bit 地址模式

此模式支持 8-bit 地址和 8-bit 数据。地址位于 EBI_AD 线的高 8 位，数据位于低 8 位。可通过设置 EBICR 寄存器的 MODE 位域为 D8A8 来选择此模式。8-bit 模式读和写时序分别如图 175 和图 176 所示。

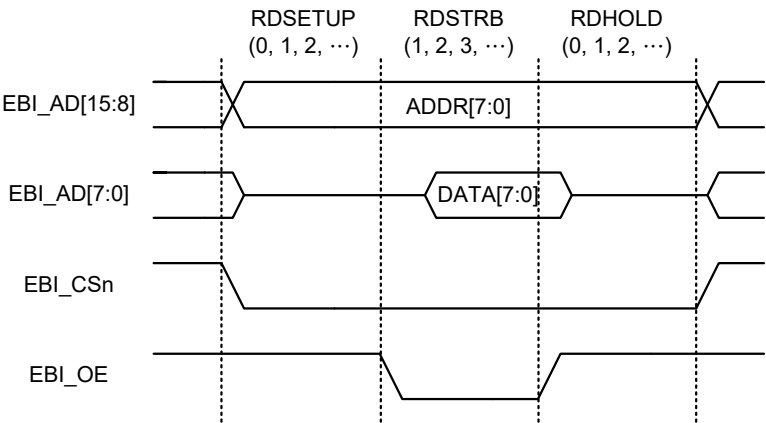


图 175. EBI 非复用 8-bit 数据 8-bit 地址读操作

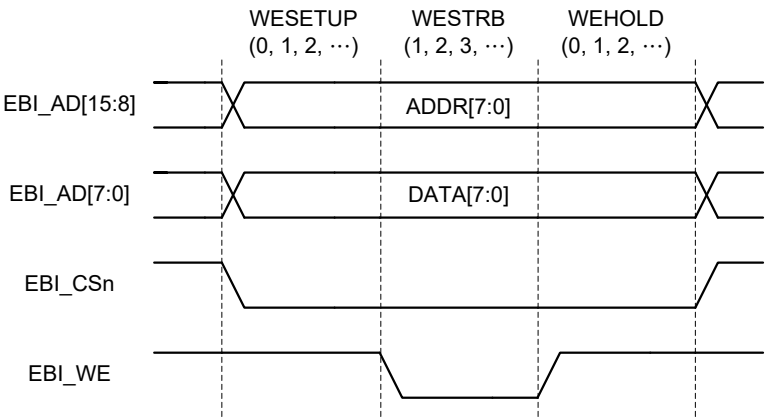


图 176. EBI 非复用 8-bit 数据 8-bit 地址写操作

非复用 16-bit 数据 N-bit 地址模式

此模式下, 16-bit 数据由 16 个 EBI_AD 线提供, 地址由 EBI_A 线提供。可通过设置 EBICR 寄存器的 MODE 位域为 D16 来选择此模式。当 EBI_A 上的 N 个地址线使能时, 读和写信号分别如图 177 和图 178 所示。

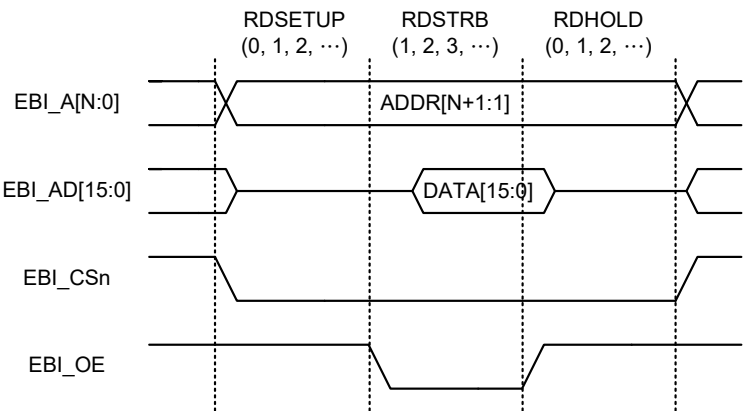


图 177. EBI 非复用 16-bit 数据 N-bit 地址读操作

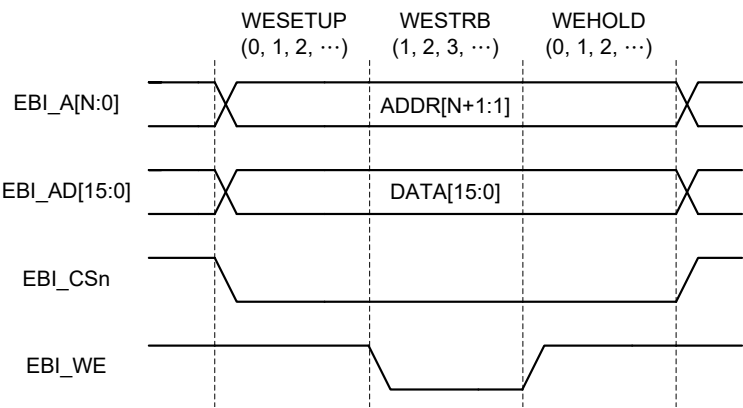


图 178. EBI 非复用 16-bit 数据 N-bit 地址写操作

由于内部 AHB 地址 (HADDR) 为字节地址 (8-bit), 而 16-bit 宽度的外部设备是按字寻址 (16-bit), 因此实际发给外部设备的地址会根据数据宽度的不同而不同, 如下表所示。

存储器宽度	发给 EBI 的数据地址
8-bit	HADDR[N:0] → EBI_A[N:0]
16-bit	HADDR[N+1:1] → EBI_A[N:0]

对于 16-bit 的外部设备宽度, EBI 在内部使用 HADDR[N+1:1] 产生地址 EBI_A[N:0] 给外部设备。无论外部设备存储器宽度为 16-bit 还是 8-bit, EBI_A[0] 必须连接到外部设备地址的 A[0]。

复用 16-bit 数据 16-bit 地址模式

此模式支持 16-bit 地址和 16-bit 数据,但需要使用一个外部锁存器以及一个额外的 EBI_ALE 信号。16-bit 地址和 16-bit 数据通过 EBI_AD 引脚复用。EBI 地址锁存器设置方框图如下图所示。可通过设置 EBICR 寄存器的 MODE 位域为 DI6A16ALE 来选择此模式。

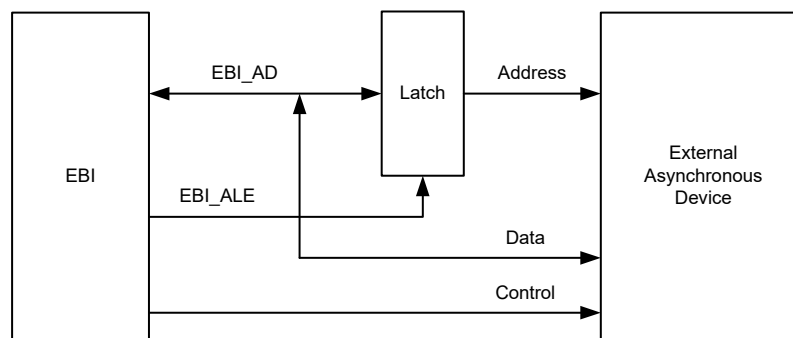


图 179. EBI 地址锁存器设置方框图

事务开始时地址被输出至 EBI_AD 线。外部地址锁存器由 EBI_ALE 信号控制并存储地址。接着根据所需操作数据可被读取或写入。读和写信号分别如图 180 和图 181 所示。

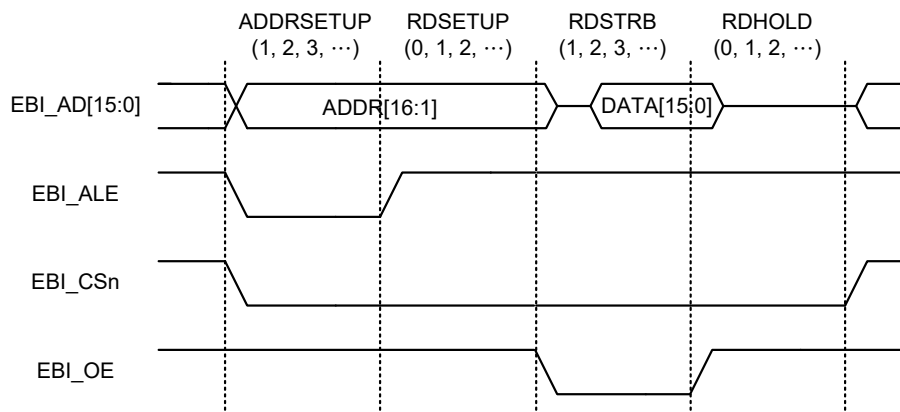


图 180. EBI 复用 16-bit 数据 16-bit 地址读操作

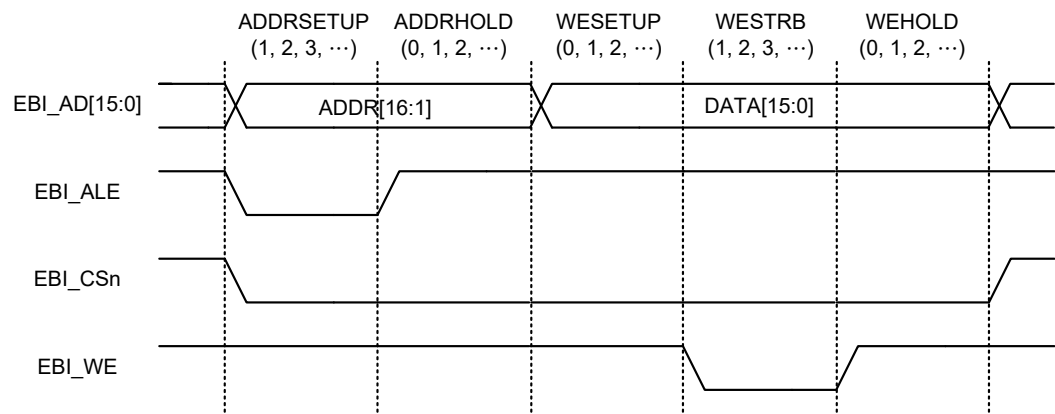


图 181. EBI 复用 16-bit 数据 16-bit 地址写操作

复用 8-bit 数据 20-bit 地址模式

此模式允许 20-bit 地址和 8-bit 数据通过 EBI_AD[15:0] 线复用以减少引脚使用量, 并使用 EBI_ALE 信号解码 8-bit 数据和 20-bit 地址。EBI_AD 线的高 8 位, 即 EBI_AD[15:8], 依次用作地址的最高 4 位和最低 8 位。EBI_AD 线的低 8 位, 即 EBI_AD[7:0], 用于中间 8 位的地址和 8 位的数据。可通过设置 EBICR 寄存器的 MODE 位域为 D8A24ALE 来选择此模式。读和写信号分别如图 182 和图 183 所示。

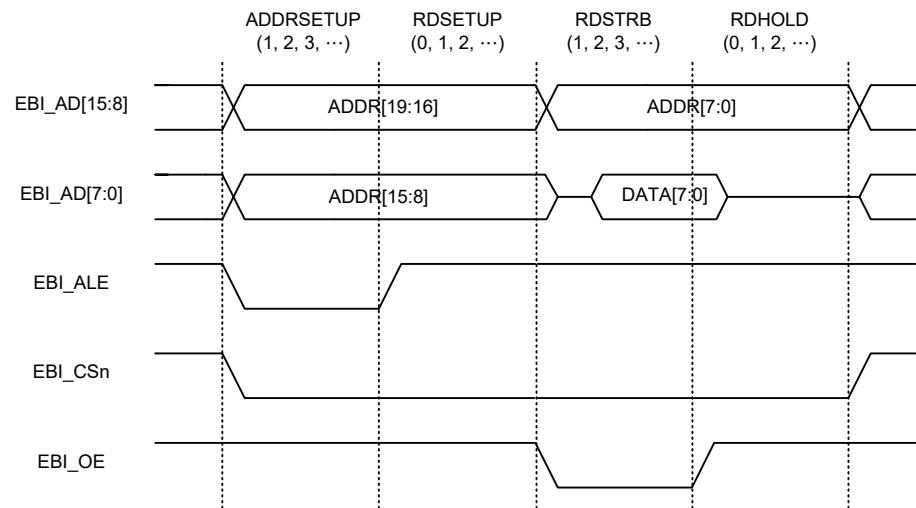


图 182. EBI 复用 8-bit 数据 20-bit 地址读操作

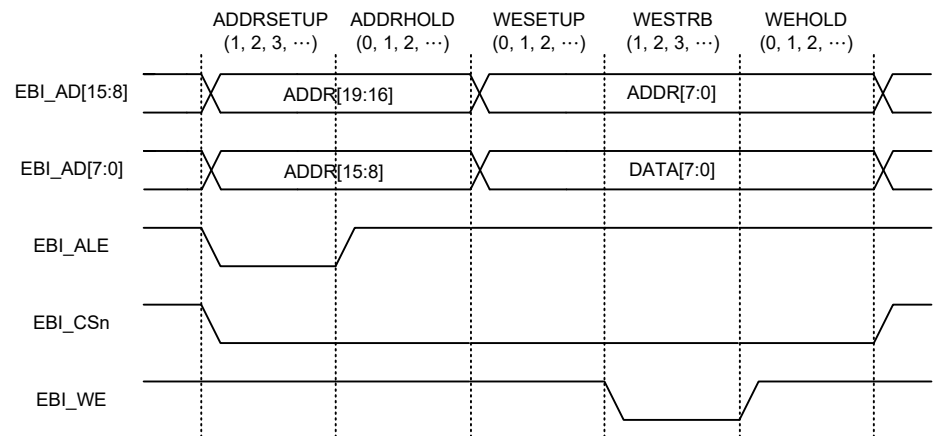


图 183. EBI 复用 8-bit 数据 20-bit 地址写操作

写缓冲器和 EBI 状态

此 EBI 有一个 32-bit 宽的写缓冲器。此缓冲器可用于限制 CPU 或 PDMA 访问一个潜在低速的外部设备的 AHB 写突发事务造成的停滞。

EBISR 寄存器的 EBIBUSY 状态位用于指示 EBI 的 AHB 事务是否仍保持有效。当执行一个 AHB 读或写操作时，EBIBUSY 位保持为 1，直到与外部设备进行的此事务已结束。

总线周转和空闲周期

EBI_AD 线可由 EBI 或外部设备驱动，这取决于 EBI 总线的周期状态。RDHOLD 时序参数用于定义总线周转时间，设置此参数时应确保有足够的时间实现外部设备各电气特性。EBI 默认设置是在访问同一个 bank 的各个 EBI 事务之间插入一个 IDLE 周期。图 184 显示了在两个连续的读事务之间插入一个 IDLE 周期。在 IDLE 状态也可提供所需总线周转时间的情况下，RDHOLD 参数可设置为 0。为了增强 EBI 访问性能，可通过设置 EBICR 寄存器的 NOIDLEn (n = 0 ~ 3) 位为 1 来除能 IDLE 状态自动插入功能。图 185 显示了非复用地址模式下两个连续的读操作范例。

以下情况时会自动插入一个 IDLE 周期：

- 当 NOIDLE 位为 0 时，访问同一个 bank 的两个外部设备事务之间。
- 访问不同 bank 的两个外部设备事务之间。
- EBI_AD 线上一个读操作和一个紧接着的写操作之间。
- EBI 无外部事务请求时。

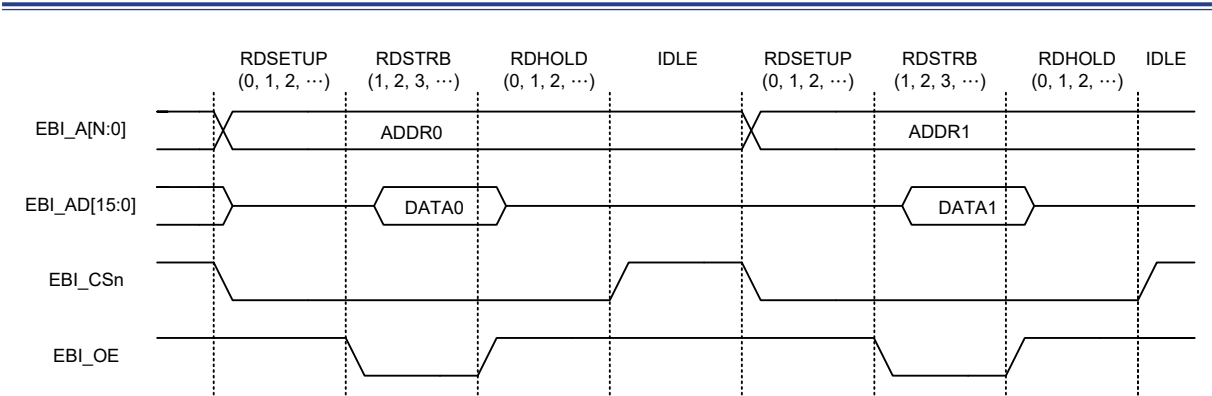


图 184. EBI 在同一个 bank 的各个事务之间插入一个 IDLE 周期 (NOIDLE = 0)

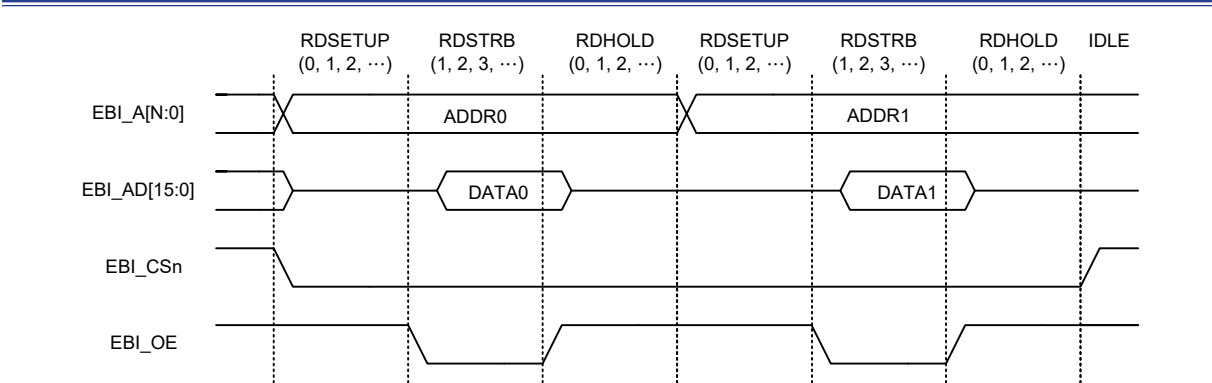


图 185. EBI 在同一个 bank 的各个事务之间不插入一个 IDLE 周期 (NOIDLE = 1)

AHB 事务宽度转换

AHB 事务与外部设备的映射取决于外部设备的数据宽度, 该单片机支持外部设备字节通道。表 71 显示了 AHB 事务与外部设备的 EBI 映射。EBI 可自动将不同的 AHB 事务宽度转换为匹配外部设备总线容量的外部设备事务。

- 若 AHB 主机 (CPU 或 PDMA) 事务宽度大于外部总线事务宽度, EBI 会将 AHB 事务分割且转换为连续多个外部事务, 这些外部事务有连续递增的地址并且从 AHB 事务的最低位开始。
- 若 AHB 主机 (CPU 或 PDMA) 事务宽度小于外部总线事务宽度, EBI 将根据外部设备的数据总线宽度读取, 忽略多余的数据。写操作时 EBI 自动执行读 - 修改 - 写时序。

表 71. AHB 事务宽度与外部设备事务的 EBI 映射

AHB 事务	8-bit 外部设备事务	16-bit 外部设备事务
8-bit 读操作	1 × 8-bit 读操作	1 × 16-bit 读操作 (EBI 忽略多余的数据)
16-bit 读操作	2 × 8-bit 读操作	1 × 16-bit 读操作
32-bit 读操作	4 × 8-bit 读操作	2 × 16-bit 读操作
8-bit 写操作	1 × 8-bit 写操作	1 × 16-bit 读操作 ; 1 × 16-bit 写操作 (EBI 执行读 - 修改 - 写时序)
16-bit 写操作	2 × 8-bit 写操作	1 × 16-bit 写操作
32-bit 写操作	4 × 8-bit 写操作	2 × 16-bit 写操作

表 72. AHB 事务宽度与外部设备事务宽度的 EBI 映射

外部总线 宽度	来自 AHB 主机的访问		访问外部总线接口 (EBI)		
	访问类型	地址 HADDR[1:0] 注	访问分割	EBI_A[1:0] 输出值	EBI_AD[15:0] 有效数据
8-bit	字节 (8-bit)	0b00	无分割	0b00	EBI_AD[7:0]
		0b01	无分割	0b01	
		0b10	无分割	0b10	
		0b11	无分割	0b11	
	半字 (16-bit)	0b00	1/2 访问	0b00	
			2/2 访问	0b01	
	半字 (16-bit)	0b10	1/2 访问	0b10	
			2/2 访问	0b11	
	字 (32-bit)	0b00	1/4 访问	0b00	
			2/4 访问	0b01	
			3/4 访问	0b10	
			4/4 访问	0b11	
16-bit	字 (8-bit)	0b00	无分割	0bx0	EBI_AD[7:0]
		0b01	无分割	0bx0	EBI_AD[15:8]
		0b10	无分割	0bx1	EBI_AD[7:0]
		0b11	无分割	0bx1	EBI_AD[15:8]
	半字 (16-bit)	0b00	无分割	0bx0	EBI_AD[15:0]
		0b10	无分割	0bx1	EBI_AD[15:0]
	字 (32-bit)	0b00	1/2 access	0bx0	EBI_AD[15:0]
			2/2 access	0bx1	EBI_AD[15:0]

注: HADDR 为 AHB 总线地址输入。

EBI Bank 访问

EBI 被分为 4 个不同的地址区域 (Bank), 每一个区域有各自单独的 EBI_CS_n 线。当访问其中一个区域时, 对应的 EBI_CS_n 线置位。这样就有多达 4 个独立的设备可共用 EBI 线且分别通过 EBI_CS_n 线识别。每个 Bank 可通过 EBICR 寄存器单独使能或除能。每个 Bank 可访问的数据空间高达 4 MB, 如下图所示。

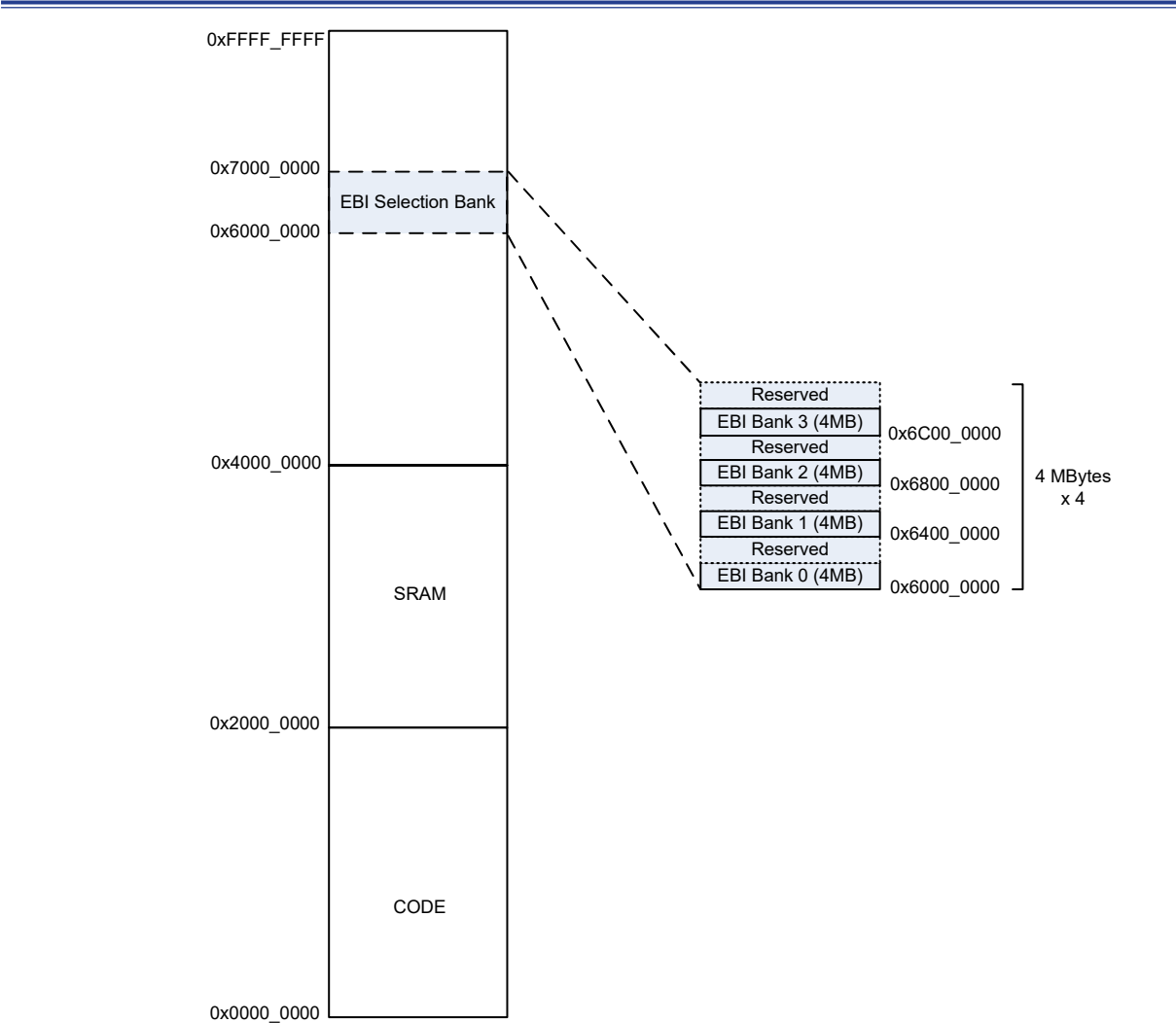


图 186. EBI Bank 存储器映射

PDMA 请求

EBI 只支持软件触发 PDMA 服务。

寄存器列表

下表显示了 EBI 寄存器和复位值。

表 73. EBI 寄存器列表

寄存器	偏移量	描述	复位值
EBICR	0x000	EBI 控制寄存器	0x0000_0000
EBISR	0x008	EBI 状态寄存器	0x0000_0000
EBIATR	0x010	EBI 地址时序寄存器	0x0000_0707
EBIRTR	0x014	EBI 读时序寄存器	0x0007_1F07
EBIWTR	0x018	EBI 写时序寄存器	0x0007_1F07
EBIPR	0x01C	EBI 极性寄存器	0x0000_0000

寄存器描述

EBI 控制寄存器 – EBICR

该寄存器定义了 EBI bank 的控制设置。

偏移量:	0x000
复位值:	0x0000_0000
	31 30 29 28 27 26 25 24
	IDLET 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
	NOIDLE3 NOIDLE2 NOIDLE1 NOIDLE0 BANKEN3 BANKEN2 BANKEN1 BANKEN0
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
	MODE3 MODE2 MODE1 MODE0
类型 / 复位	RW 0 RW 0 RW 0 RW 0 RW 0 RW 0 RW 0

位	域	描述
[31:28]	IDLET	IDLE 时间 该位设置了 EBI 事务之间的周期数。若该位为 0，硬件插入一个周期，该周期以 HCLK 时钟周期为单位。
[15]	NOIDLE3	无 IDLE 3 0：使能 IDLE 状态插入 1：除能 IDLE 状态插入 使能或除能 Bank 3 各事务之间空闲状态插入功能。

位	域	描述
[14]	NOIDLE2	无 IDLE 2 0: 使能 IDLE 状态插入 1: 除能 IDLE 状态插入 使能或除能 Bank 2 各事务之间空闲状态插入功能。
[13]	NOIDLE1	无 IDLE 1 0: 使能 IDLE 状态插入 1: 除能 IDLE 状态插入 使能或除能 Bank 1 各事务之间空闲状态插入功能。
[12]	NOIDLE0	无 IDLE 0 0: 使能 IDLE 状态插入 1: 除能 IDLE 状态插入 使能或除能 Bank 0 各事务之间空闲状态插入功能。
[11]	BANKEN3	Bank 3 使能 0: 除能 1: 使能 该位使能或除能 Bank 3。
[10]	BANKEN2	Bank 2 使能 0: 除能 1: 使能 该位使能或除能 Bank 2。
[9]	BANKEN1	Bank 1 使能 0: 除能 1: 使能 该位使能或除能 Bank 1。
[8]	BANKEN0	Bank 0 使能 0: 除能 1: 使能 该位使能或除能 Bank 0。
[7:6]	MODE3	设置 EBI Bank 3 访问模式 00: D8A8 01: D16A16ALE 10: D8A24ALE 11: D16
[5:4]	MODE2	设置 EBI Bank 2 访问模式 00: D8A8 01: D16A16ALE 10: D8A24ALE 11: D16
[3:2]	MODE1	设置 EBI Bank 1 访问模式 00: D8A8 01: D16A16ALE 10: D8A24ALE 11: D16
[1:0]	MODE0	设置 EBI Bank 0 访问模式 00: D8A8 01: D16A16ALE 10: D8A24ALE 11: D16

EBI 状态寄存器 – EBISR

该寄存器定义了 EBI 状态。

偏移量： 0x008

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							EBISMRST	
类型 / 复位								RW	0
	7	6	5	4	3	2	1	0	
	保留位							EBIBUSY	
类型 / 复位								RO	0

位	域	描述
[8]	EBISMRST	EBI 状态机复位 0: 正常 1: 复位 EBI 状态机 该位写 1 将使 EBI 内部状态机复位到初始状态且保持初始的寄存器设置。
[0]	EBIBUSY	EBI 忙碌 0: EBI 空闲 1: EBI 忙碌 该位置位用于指示 EBI 忙于 AHB 事务。

EBI 地址时序寄存器 – EBIATR

该寄存器定义了每个 Bank 的地址时序。

偏移量： 0x010
复位值： 0x0000_0707

	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					ADDRHOLD		
类型 / 复位						RW	1	RW
	7	6	5	4	3	2	1	0
	保留位					ADDRSETUP		
类型 / 复位						RW	1	RW

位	域	描述
[10:8]	ADDRHOLD	地址保持时间 设置 EBI_ALE 置位后 EBI_AD 总线上地址保持的周期数。该位域可设为 0。
[2:0]	ADDRSETUP	地址设置时间 设置 EBI_ALE 置位前 EBI_AD 总线上地址设置的周期数。若设为 0，硬件插入一个周期，该周期以 HCLK 时钟周期为单位。

EBI 读时序寄存器 – EBIRTR

该寄存器定义了每个 Bank 的读时序设置。

偏移量： 0x014
复位值： 0x0007_1F07

	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				RDHOLD			
类型 / 复位					RW 1 RW 1 RW 1			
	15	14	13	12	11	10	9	8
	保留位			RDSTRB				
类型 / 复位				RW 1 RW 1 RW 1 RW 1				
	7	6	5	4	3	2	1	0
	保留位					RDSETUP		
类型 / 复位						RW 1 RW 1 RW 1		

位	域	描述
[18:16]	RDHOLD	读保持时间 该位用于设置EBI_OE信号无效后EBI_CS _n 保持有效的周期数。该时间用于总线周转。
[12:8]	RDSTRB	读选通时间 设置EBI_OE保持有效的周期数。经过此特定的时间后，数据被读取。若该位域设为0，硬件插入一个周期，该周期以HCLK时钟周期为单位。
[2:0]	RDSETUP	读设置时间 设置EBI_OE有效前地址设置所用的周期数，该周期以HCLK时钟周期为单位。

EBI 写时序寄存器 – EBIWTR

该寄存器定义了每个 bank 的写时序设置。

偏移量： 0x018
复位值： 0x0007_1F07

	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				WRHOLD			
类型 / 复位					RW 1 RW 1 RW 1			
	15	14	13	12	11	10	9	8
	保留位			WRSTRB				
类型 / 复位				RW 1 RW 1 RW 1 RW 1				
	7	6	5	4	3	2	1	0
	保留位					WRSETUP		
类型 / 复位						RW 1 RW 1 RW 1		

位	域	描述
[18:16]	WEHOLD	写保持时间 设置 EBI_WE 信号无效后 EBI_CS _n 保持有效的周期数。
[12:8]	WESTRB	写选通时间 设置 EBI_WE 保持有效的周期数。若该位域设为 0，硬件插入一个周期，该周期以 HCLK 时钟周期为单位。
[2:0]	WESETUP	写设置时间 设置 EBI_WE 有效前地址设置所用的周期数，该周期以 HCLK 时钟周期为单位。

EBI 极性寄存器 – EBIPR

该寄存器定义了每个 Bank EBI 控制信号的极性。

偏移量：0x01C

复位值：0x0000_0000

类型 / 复位	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				ALEPOL	WEPOL	OEPOL	CSPOL
					RW	0	RW	0

位	域	描述
[3]	ALEPOL	地址锁存极性 0: EBI_ALE 低有效 1: EBI_ALE 高有效 该位设置 EBI_ALE 引脚的极性。
[2]	WEPOL	写使能极性 0: EBI_WE 低有效 1: EBI_WE 高有效 该位设置 EBI_WE 引脚的极性。
[1]	OEPOL	输出使能极性 0: EBI_OE 低有效 1: EBI_OE 高有效 该位设置 EBI_OE 引脚的极性。
[0]	CSPOL	片选极性 0: EBI_CS 低有效 1: EBI_CS 高有效 该位设置 EBI_CS _n 引脚的极性。

27 循环冗余校验 (CRC)

简介

循环冗余校验 (CRC) 计算单元是一种错误检测技术测试算法，用于验证数据传输或存储数据的正确性。CRC 计算将数据流或数据块作为输入，并生成一个 16-bit 或 32-bit 的输出余数。通常情况下，一个数据流带有 CRC 后缀码，当被发送或存储时该数据流可用作校验和。因此，被接收或重新存储的数据流都是通过上述相同的生成多项式计算得到的。若新的 CRC 码结果与先前计算的不匹配，这意味着数据流出错。

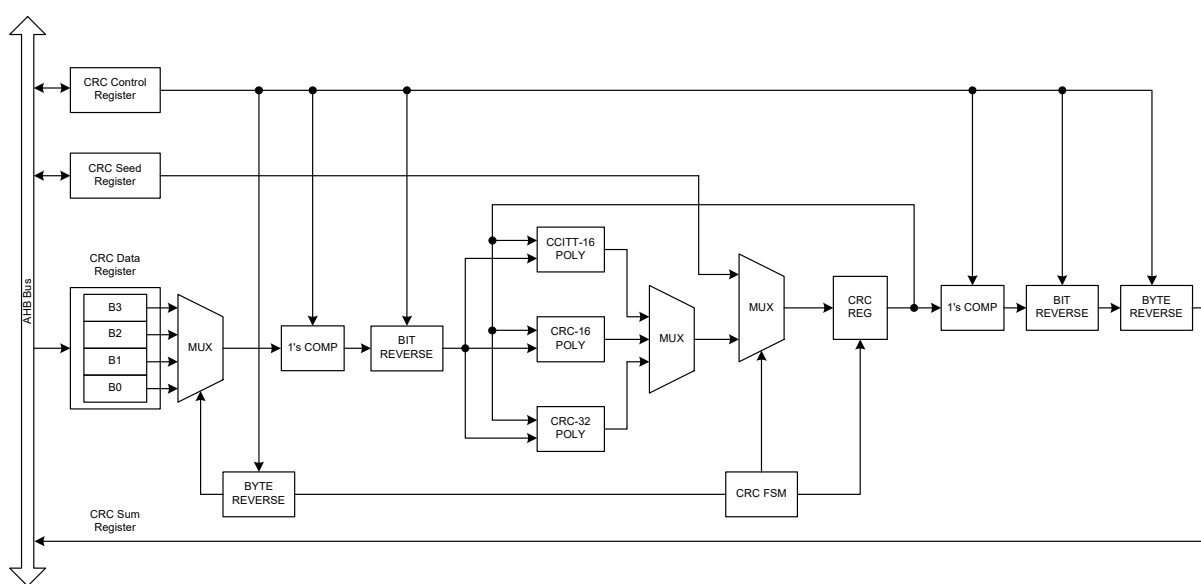


图 187. CRC 方框图

特性

- 支持 CRC16 多项式: $0x8005$, $X^{16} + X^{15} + X^2 + 1$
- 支持 CCITT CRC16 多项式: $0x1021$, $X^{16} + X^{12} + X^5 + 1$
- 支持 IEEE-802.3 CRC32 多项式: $0x04C11DB7$,
 $X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$
- 支持对数据和校验和进行反码、按字节反序以及按位反序操作
- 支持字节、半字和字数据大小
- 可编程 CRC 初始种子值
- 对 8-bit 数据执行 CRC 计算需要 1 个 AHB 时钟周期, 32-bit 数据需要 4 个 AHB 时钟周期
- 支持 PDMA 对指定存储器区块进行 CRC 计算

功能描述

此 CRC 单元仅允许 CRC16、CCITT CRC16 和 IEEE-802.3 CRC32 多项式的计算。在此单元内, 生成多项式有各自固定的二进制值。因此, 基于其他生成多项式的 CRC 值不可计算。

CRC 计算

CRC 计算单元具有一个 32-bit 的写 CRC 数据寄存器 (CRCDR) 和读 CRC 校验和寄存器 CRCCSR。CRCDR 寄存器用于输入新数据 (写访问), 而 CRCCSR 寄存器用于保持先前的 CRC 计算的结果 (读访问)。每次对 CRC 数据寄存器 CRCDR 进行写操作, 都会将存储在 CRCCSR 中的前个 CRC 值和新的 CRC 值结合起来。CRC 方框图如图 187 所示。CRC 单元计算 CRC 数据寄存器值是按字节进行的, 默认的字节和位的顺序为大端模式。CRCDR 寄存器可以通过字、右对齐半字节和右对齐字节的方式进行写访问。对于其它寄存器仅允许以 32-bit 格式进行访问。计算持续时间取决于数据宽度:

- 输入数据为 32-bit: 需要 4 个 AHB 时钟周期
- 输入数据为 16-bit: 需要 2 个 AHB 时钟周期
- 输入数据为 8-bit: 需要 1 个 AHB 时钟周期

CRC 计算按字节反序与位反序

字节重新排序和字节内位反序操作可出现在数据用于 CRC 计算之前或 CRC 校验和输出后, 可通过 CRCCR 寄存器相应的设定位域配置。这些操作发生在字或半字节访问时。字节写访问时硬件会忽略 CRCCR 寄存器中的 DATBYRV 位, 但在字节写入时出现按位反序仍可置位 DATBIRV 位。下图为按字节和位反序操作范例。

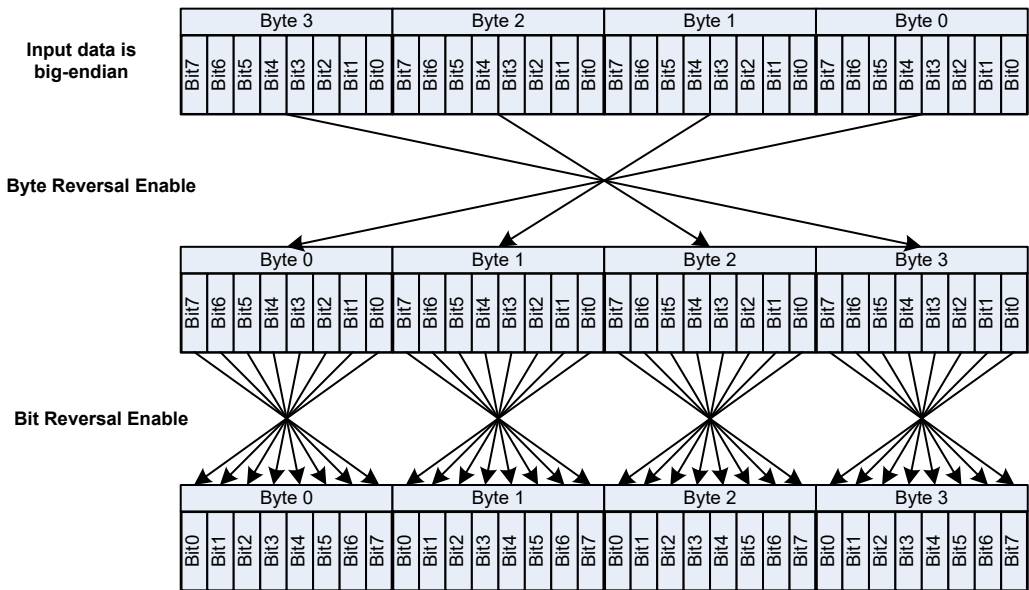


图 188. CRC 数据按字节反序和按位反序范例

CRC 与 PDMA 搭配工作

可使用软件触发 PDMA 通道可用于将数据传入 CRC 单元。若需计算大量的数据块，建议先使用 PDMA 来传输其容量允许的数据量，再使用软件写操作来传输剩余的字节。要将数据写入 CRC 单元，PDMA 应按字访问的方式将数据从存储器源地址写入 CRC 数据寄存器 CRCDR (固定地址模式)。通过软件可将剩余字节写入 CRC 数据寄存器 CRCDR 且可从 CRC 校验和寄存器 CRCCSR 中读取 CRC 计算结果的值。

寄存器列表

下面表格显示了 CRC 的寄存器和复位值。

表 74. CRC 寄存器列表

寄存器	偏移量	描述	复位值
CRCCR	0x000	CRC 控制寄存器	0x0000_0000
CRCSDR	0x004	CRC 种子值寄存器	0x0000_0000
CRCCSR	0x008	CRC 校验和寄存器	0x0000_0000
CRCDR	0x00C	CRC 数据寄存器	0x0000_0000

寄存器描述

CRC 控制寄存器 – CRCCR

该寄存器定义了相关的 CRC 功能使能控制。

偏移量: 0x000
复位值: 0x0000_0000

类型 / 复位	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
	SUMCMPL	SUMBYRV	SUMBIRV	DATCMPL	DATBYRV	DATBIRV	POLY	
类型 / 复位	RW	0	RW	0	RW	0	RW	0

位	域	描述
[7]	SUMCMPL	对校验和进行反码 0: 除能 1: 使能
[6]	SUMBYRV	对校验和进行字节反序 0: 除能 1: 使能
[5]	SUMBIRV	对校验和进行位反序 0: 除能 1: 使能
[4]	DATCMPL	对数据进行反码 0: 除能 1: 使能
[3]	DATBYRV	对数据进行字节反序 0: 除能 1: 使能
[2]	DATBIRV	对数据进行位反序 0: 除能 1: 使能
[1:0]	POLY	CRC 多项式 00: CRC-CCITT (0x1021) 01: CRC-16 (0x8005) 1X: CRC-32 (0x04C11DB7)

CRC 种子值寄存器 – CRCSDR

该寄存器定义了 CRC 种子值。

偏移量: 0x004
复位值: 0x0000_0000

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

位	域	描述
[31:0]	SEED	CRC 种子值 当所有数据已写入 CRCDR 寄存器，可通过此寄存器获得依 CRCCR 寄存器里多项式的设置所算出的 16-bit 或 32-bit 校验和。

CRC 校验和寄存器 – CRCCSR

此寄存器包含了 CRC 校验和输出。

偏移量: 0x008
复位值: 0x0000_0000

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

位	域	描述
[31:0]	CHKSUM	CRC 校验和数据 当所有数据已写入 CRCDR 寄存器，根据 CRCCR 寄存器里多项式的设置通过此寄存器获得 16/32-bit 校验和。

CRC 数据寄存器 – CRCDR

该寄存器定义 CRC 输入数据。

偏移量: 0x00C
复位值: 0x0000_0000

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

位	域	描述
[31:0]	CRCDATA	CRC 输入数据 允许使用字节、半字或字写入方式。根据写入方式选择反码、按字节反序或按位反序操作。

28 AES 加密 / 解密接口 (AES)

简介

AES 核心提供加密和解密功能。它支持 128 位输入数据，但需要注意的是，若输入数据不足 128 位，硬件不会对数据位进行填补，因此用户需在最开始通过软件进行补足。

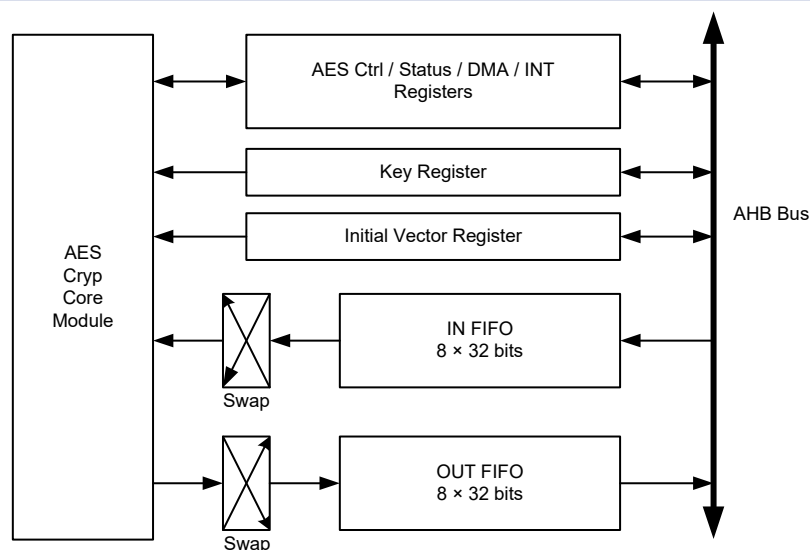


图 189. AES 方框图

特性

- 支持 AES 加密 / 解密功能
- 支持 AES ECB / CBC / CTR 三种模式
- 支持 128、192 和 256 位密钥长度
- 支持 4 字长度的初始矢量用于 CBC 和 CTR 模式
- 8 × 32 位 (每个输入 / 输出 FIFO 容量) 用于 2 个 AES 数据块
- 支持按字的数据交换功能
- 支持 PDMA 接口

功能描述

AES 模式

AES 电码本模式 (AES-ECB)

ECB 模式加密, 输入 FIFO 中的 128 位明文数据, 经过字交换处理后, 发送到 AES 核心进行加密操作。AES 核心使用 128、192 或 256 位密钥进行加密处理。加密后产生的密文再次经过字交换处理后, 存入输出 FIFO。

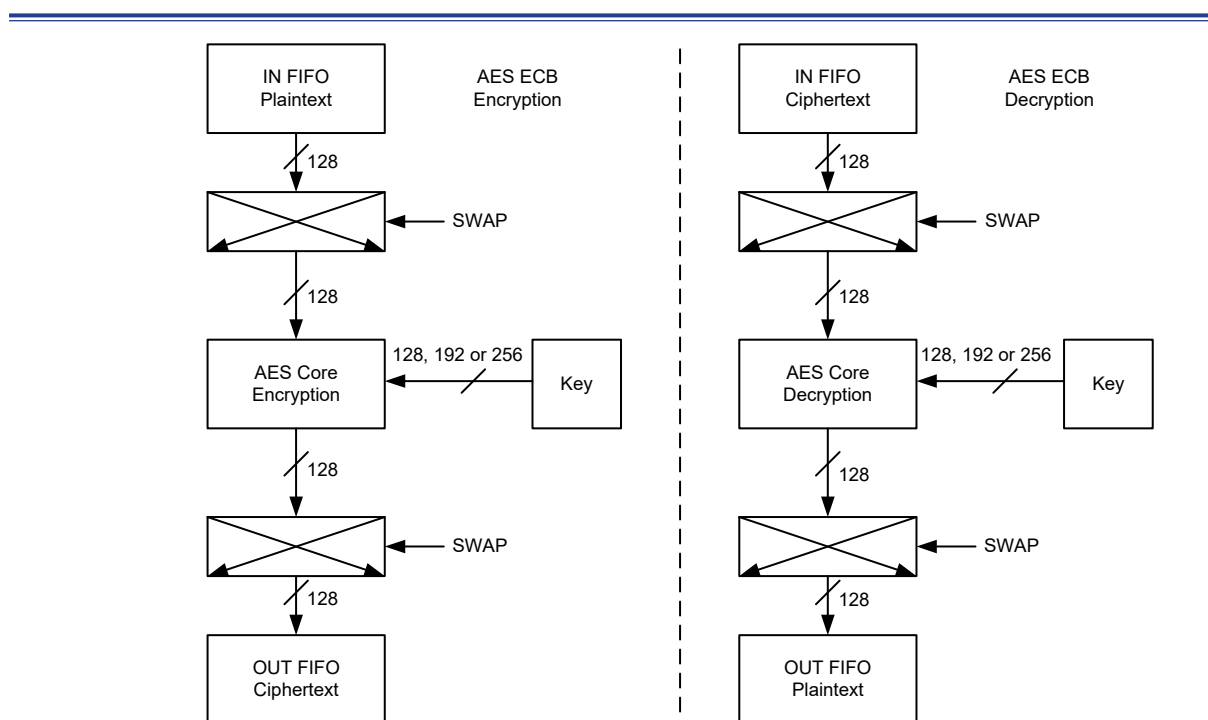


图 190. AES-ECB 模式

AES 密码分组链接模式 (AES-CBC)

CBC 模式加密，明文切分成若干小段，每一个明文段与上一个的密文段异或运算后，其结果再与密钥进行加密。通过第一次加密运算得到第一个初始矢量起始值。明文经过字交换处理后与初始矢量进行异或运算，其结果与密钥进行加密运算。当生成的密文数据存入输出 FIFO 时，同时初始矢量值会由输出密文数据更新。

CBC 模式解密，每一个明文段经过解密后与上一个的密文段进行异或运算。通过第一次解密运算得到第一个初始矢量起始值。密文经过字交换处理及解密之后，与初始矢量进行异或运算。当异或后的解密数据存至输出 FIFO 时，同时初始矢量值会由解密输入数据更新，用于下一轮的密文解密。

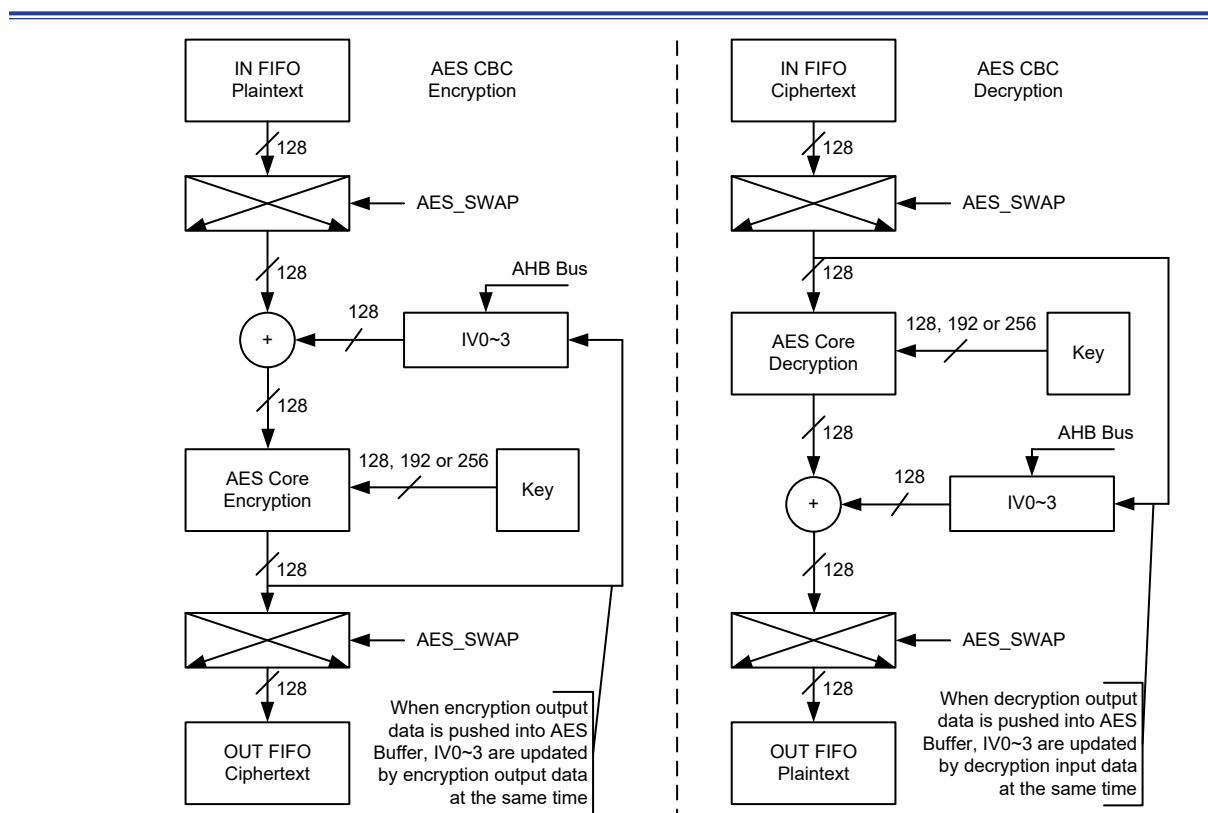


图 191. AES-CBC 模式

AES 计算器模式 (AES-CTR)

在 CTR 模式中，初始矢量计数器值在增 1 之后，被发送至 AES 核心进行加密得到密文。在加密和解密时，AES 核心使用相同的方向设置。

CTR 模式加密和解密，输入 FIFO 中的数据进行字交换处理后与密文执行异或，异或的结果再经过字交换后发送到输出 FIFO。同时初始矢量值增 1 用于产生下一轮密文。

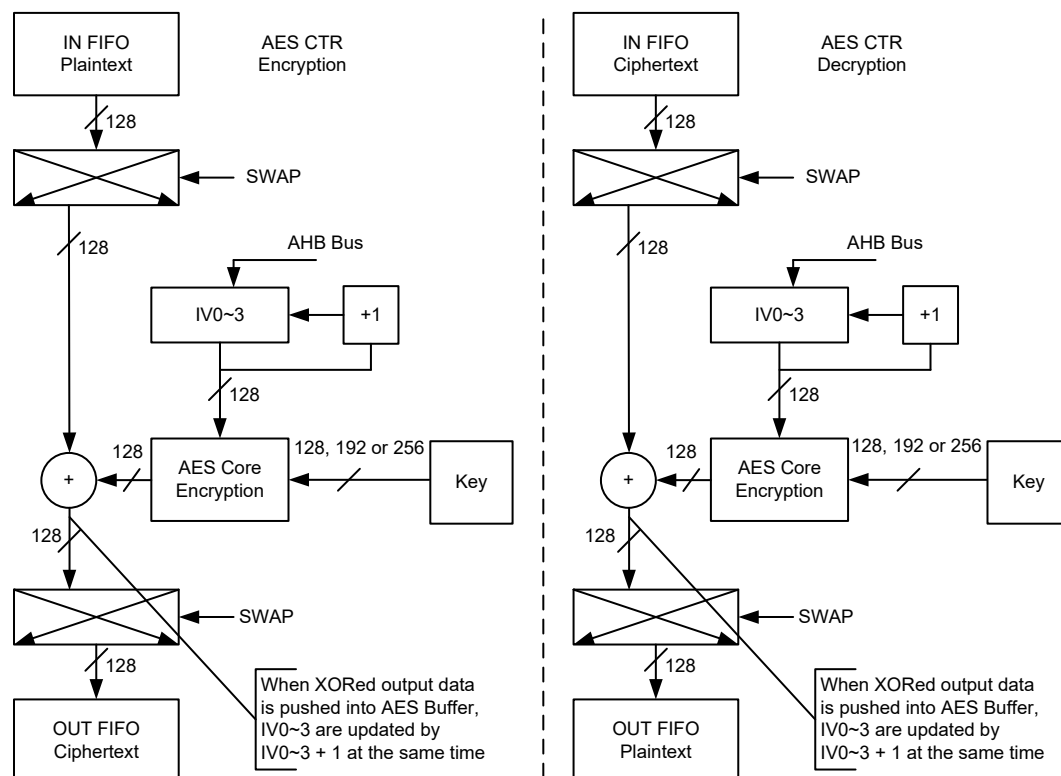


图 192. AES-CTR 模式

AES 状态

提供了五种状态标志位可供用户监测 AES 的工作情况。当输入 FIFO 为空时, IFEMPTY 位被置位; 当输入 FIFO 未滿时, IFNFULL 位会被置位。当输出 FIFO 中有数据时, OFNEMPTY 位被置位; 当输出 FIFO 存滿数据时, OFFULL 位会被置位。当 AES 核心在进行加密, 解密或者当在进行密钥扩展时, BUSY 位被置位。

AES PDMA 接口

AES 支持 32 位 PDMA 数据传输。当输入 FIFO 为空, AES 将发送一个输入 FIFO 请求到 PDMA。当输出 FIFO 已滿, AES 将发送一个输出 FIFO 请求到 PDMA。

AES 中断

当输入 FIFO 小于或等于一个 AES 数据块 (4 × 32 位), 产生输入 FIFO 中断 (IFINT) 请求。当输出 FIFO 中有数据时, 产生输出 FIFO 中断 (OFINT) 请求。当 IFINT 被置位, 若 IFINT 中断使能且此时 AES 为使能状态 (AESEN = 1), 才会产生 AES 中断; 当 OFINT 被置位, 若 OFINT 中断使能, 即可产生 AES 中断, 这种情况下不需考虑 AES 是否为使能状态。

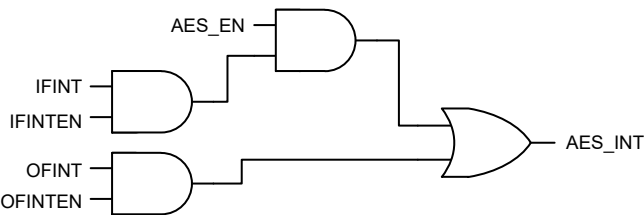


图 193. AES 中断

AES 初始矢量

ECB 模式时, 不会用到初始矢量 (IV0 ~ 3)。CBC 和 CTR 模式时, 初始矢量会在第一个输入数据块时被初始化。完成第一个输入数据块的操作后, 初始矢量通过硬件自动更新作为下一轮数据块的初始矢量。CTR 模式的初始矢量包括随机数、初始矢量和计数器。在每一个 AES 数据块处理完成后, 计数器值加一。

Nonce (32 bits)	IV0
Initial Vector (64 bits)	IV1, 2
Counter (32 bits)	IV3

图 194. CTR 模式初始矢量

AES 字交换

AES 支持字交换功能。交换动作发生在输入 FIFO 和 AES 块数据之间以及输出 FIFO 和 AES 块数据之间。若需要使用字交换功能，需设置 AESCR 寄存器中的 SWAP 位为高以使能字交换功能。

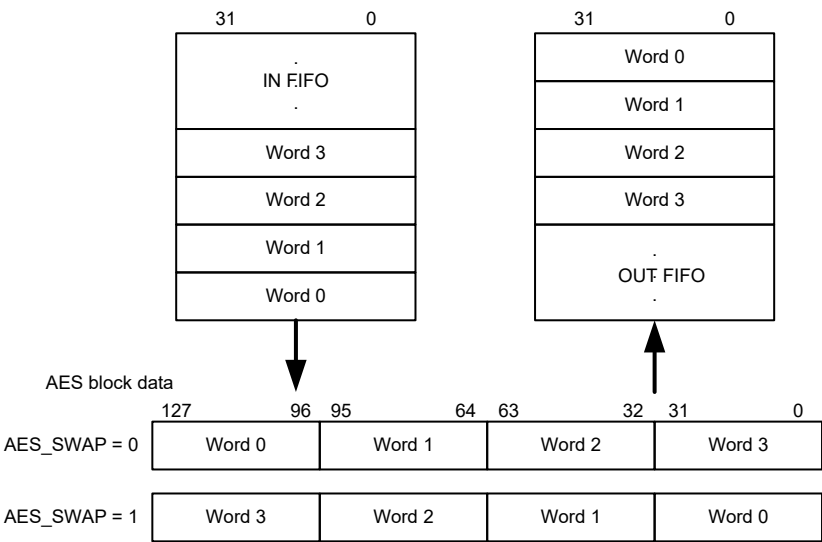


图 195. AES 字交换功能

寄存器列表

下面表格显示了 AES 的寄存器和复位值。

表 75. AES 寄存器列表

寄存器	偏移量	描述	复位值
AESCR	0x000	AES 控制寄存器	0x0000_0200
AESSR	0x004	AES 状态寄存器	0x0000_0003
AESDMAR	0x008	AES DMA 寄存器	0x0000_0000
AESISR	0x00C	AES 中断状态寄存器	0x0000_0001
AESIER	0x010	AES 中断使能寄存器	0x0000_0000
AESDINR	0x014	AES 数据输入寄存器	0x0000_0000
AESDOUTR	0x018	AES 数据输出寄存器	0x0000_0000
AESKEY0	0x01C	AES 密钥寄存器 0	0x0000_0000
AESKEY1	0x020	AES 密钥寄存器 1	0x0000_0000
AESKEY2	0x024	AES 密钥寄存器 2	0x0000_0000
AESKEY3	0x028	AES 密钥寄存器 3	0x0000_0000
AESKEY4	0x02C	AES 密钥寄存器 4	0x0000_0000
AESKEY5	0x030	AES 密钥寄存器 5	0x0000_0000
AESKEY6	0x034	AES 密钥寄存器 6	0x0000_0000
AESKEY7	0x038	AES 密钥寄存器 7	0x0000_0000
AESIVR0	0x03C	AES 初始矢量寄存器 0	0x0000_0000

寄存器	偏移量	描述	复位值
AESIVR1	0x040	AES 初始矢量寄存器 1	0x0000_0000
AESIVR2	0x044	AES 初始矢量寄存器 2	0x0000_0000
AESIVR3	0x048	AES 初始矢量寄存器 3	0x0000_0000

寄存器描述

AES 控制寄存器 – AESCR

该寄存器定义了 AES 控制设置。

偏移量: 0x000
复位值: 0x0000_0200

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					FFLUSH	ENDIAN	SWAP
类型 / 复位					RW 0	RW 1	RW 0
7	6	5	4	3	2	1	0
保留位	KEYSIZE		KEYSTART	MODE		DIR	AESEN
类型 / 复位	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	域	描述
[10]	FFLUSH	AES 输入 / 输出 FIFO 清理 0: 无动作 1: 清理 FIFO
[9]	ENDIAN	此位由硬件自动清零。该位只能在 AES 除能模式下被置位。 字节存储机制选择 0: 大端方式 1: 小端方式
[8]	SWAP	该设定用于输入 / 输出 FIFO 数据，密钥以及初始向量数据。 AES 数据交换功能 0: 不交换 1: 字交换
[6:5]	KEYSIZE	该设定用于输入 / 输出 FIFO 数据。 AES 密钥长度 00: 128 位 01: 192 位 1x: 256 位
[4]	KEYSTART	AES 密钥开始 0: 密钥未开始 1: 密钥开始 此位由硬件自动清零。该位只有在 AES 使能模式才有效。

位	域	描述
[3:2]	MODE	AES 功能模式 00: ECB 模式 01: CBC 模式 1x: CTR 模式
[1]	DIR	AES 方向 0: 加密 1: 解密
[0]	AESEN	AES 使能 0: AES 除能 1: AES 使能

AES 状态寄存器 – AESSR

该寄存器指示 AES 状态。

偏移量: 0x004
复位值: 0x0000_0003

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			BUSY	OFFULL	OFNEMPTY	IFNFULL	IFEMPTY
类型 / 复位			RO	0 RO	0 RO	0 RO	1 RO 1

位	域	描述
[4]	BUSY	忙标志位 0: AES 空闲 1: AES 忙 当 AES 在进行加密、解密及密钥扩展时, AES 为忙状态。
[3]	OFFULL	输出 FIFO 满标志位 0: 输出 FIFO 未满 1: 输出 FIFO 已满
[2]	OFNEMPTY	输出 FIFO 非空标志位 0: 输出 FIFO 空 1: 输出 FIFO 非空
[1]	IFNFULL	输入 FIFO 未满标志位 0: 输入 FIFO 已满 1: 输入 FIFO 未满

位	域	描述
[0]	IFEMPTY	输入 FIFO 空标志位 0: 输入 FIFO 非空 1: 输入 FIFO 空

AES DMA 寄存器 – AESDMAR

该寄存器用于 DMA 设置。

偏移量: 0x008

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						OFDMAEN	IFDMAEN
类型 / 复位						RW 0	RW 0

位	域	描述
[1]	OFDMAEN	输出 FIFO DMA 使能 0: DMA 除能 1: DMA 使能
[0]	IFDMAEN	输入 FIFO DMA 使能 0: DMA 除能 1: DMA 使能

AES 中断状态寄存器 – AESISR

该寄存器用于储存 AES 的中断状态位。

偏移量： 0x00C

复位值： 0x0000_0001

	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						OFINT	IFINT
类型 / 复位							RO 0 RO	1

位	域	描述
[1]	OFINT	输出 FIFO 中断状态 0: 未发生输出 FIFO 中断 1: 发生输出 FIFO 中断
[0]	IFINT	输入 FIFO 中断状态 0: 未发生输入 FIFO 中断 1: 发生输入 FIFO 中断

AES 中断使能寄存器 – AESIER

该寄存器用于设置 AES 中断。

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						OFINTEN	IFINTEN
类型 / 复位							RW 0	RW 0

位	域	描述
[1]	OFINTEN	输出 FIFO 中断使能位 0: 中断除能 1: 中断使能
[0]	IFINTEN	输入 FIFO 中断使能位 0: 中断除能 1: 中断使能

AES 数据输入寄存器 – AESDINR

该寄存器用于定义数据输入值。

偏移量: 0x014

复位值: 0x0000_0000

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

位	域	描述
[31:0]	DIN	AES 数据输入 0x0000_0000 ~ 0xFFFF_FFFF

AES 数据输出寄存器 – AESDOUTR

该寄存器用于储存 AES 数据输出。

偏移量: 0x018

复位值: 0x0000_0000

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

位	域	描述
[31:0]	DOUT	AES 数据输出 0x0000_0000 ~ 0xFFFF_FFFF

AES 密钥寄存器 n – AESKEYRn, n = 0 ~ 7

该寄存器用于定义密钥数据字 n。

偏移量: 0x01C ~ 0x038

复位值: 0x0000_0000

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

位	域	描述
[31:0]	KeyData	密钥数据 0x0000_0000 ~ 0xFFFF_FFFF

AES 初始矢量寄存器 n – AESIVRn, n = 0 ~ 3

该寄存器用于定义初始矢量数据字 n。

偏移量: 0x03C ~ 0x048

复位值: 0x0000_0000

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

位	域	描述
[31:0]	IVData	初始矢量数据 0x0000_0000 ~ 0xFFFF_FFFF

Copyright© 2021 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/>.